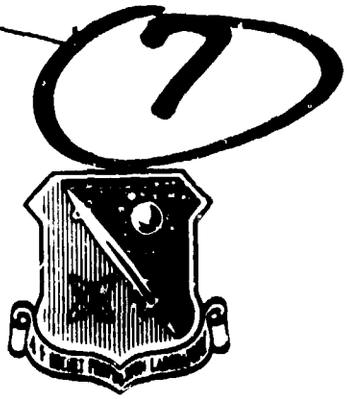
LEVEL II



AD A108224

AFRPL-TR-81-70
Thiokol Report U-81-4457A

USER'S MANUAL FOR SOLID PROPULSION OPTIMIZATION CODE (SPOC)

Volume I - Technical Description

Author: G. P. Roys

Thiokol Corporation
Huntsville Division
Huntsville, AL 35807

August 1981

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

The AFRPL Technical Services Office has reviewed this report, and it is releasable to the National Technical Information Service, where it will be available to the general public, including foreign nationals.

DTIC FILE COPY

Prepared for

AIR FORCE ROCKET PROPULSION LABORATORY
DIRECTOR OF SCIENCE AND TECHNOLOGY
AIR FORCE SYSTEMS COMMAND
EDWARDS AFB, CALIFORNIA 93523

81 12 08 217

NOTICES

When U. S. Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise, or in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that way in any way be related thereto.

FOREWORD

This report was submitted by Thiokol Corporation/Huntsville Division, Huntsville AL 35807, under Contract F04611-80-C-0016, Job Order No. 314809VG with the Air Force Rocket Propulsion Laboratory, Edwards AFB, CA 93523. This Technical Report is approved for release and distribution in accordance with the distribution statement on the cover and on the DD Form 1473.

JOHN H. COSTELLO, JR, 1Lt, USAF
Project Manager

LEE G. MEYER, Chief
Air-Launched Missile
Propulsion Branch

FOR THE COMMANDER

CHARLES R. COOKE
Director, Solid Rocket Division

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

(12)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFRPL-TR-81-70	2. GOVT ACCESSION NO.: AD-A108 224
3. RECIPIENT'S CATALOG NUMBER:	
4. TITLE (and Subtitle): User's Manual for Solid Propulsion Optimization Code (SPOC) Volume I - Technical Description	5. TYPE OF REPORT & PERIOD COVERED: User's Guide 28 Mar 80 - 21 Aug 81
	6. PERFORMING ORG. REPORT NUMBER: U-81-4457A
7. AUTHOR(s): G. P. Roys	8. CONTRACT OR GRANT NUMBER(s): AFRPL F04611-80-C-0016
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Thiokol Corporation Huntsville Division Huntsville, AL 35807	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: JON: 314809VG
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Rocket Propulsion Laboratory/MKAS Edwards Air Force Base, CA 93523	12. REPORT DATE: August 1981
	13. NUMBER OF PAGES: 240
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): N/A	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE: N/A

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)

Approved for Public Release. Distribution Unlimited.

18. SUPPLEMENTARY NOTES

Principal Investigator for Thiokol was G. P. Roys. W. I. Dale, Jr. was Program Manager. Significant technical contributions were made by W. C. Aycock, D. D. Maser, P. R. Renfroe, and R. O. Hessler

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

Solid propellant rocket motor, mathematical modeling, numerical non-linear optimization, computer code, preliminary design.

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

This report is Volume I of a three-volume user's manual for a computer code that performs detailed preliminary designs of solid propellant rocket motors. All major components and performance of a motor are mathematically determined using source dimensions and characteristics. A direct pattern search non-linear optimization scheme based on the Hooke and Jeeves algorithm is employed to establish motor characteristics that optimize any one of several performance parameters. Decision variables during

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

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

401444

20. optimization are propellant formulation, propellant burn rate, propellant grain dimensions, nozzle dimensions, and pressure vessel dimensions. Provisions are made for easily inserted user-defined models of several characteristics. Constraints imposed during the optimization process are performance requirements, design constraints, and operating limits.

TABLE OF CONTENTS

Table of Contents (Continued)

Table of Contents (Continued)

LIST OF FIGURES

List of Figures (Continued)

List of Figures (Continued)

List of Figures (Continued)

LIST OF TABLES

VOLUME I

TECHNICAL DESCRIPTION

INTRODUCTION

The Solid Propulsion Optimization Code (SPOC) performs detailed preliminary designs of a large variety of solid propellant rocket motors. Dimensions of the propellant grain, nozzle, and pressure vessel are adjusted by the code, along with propellant formulation and burn rate, to produce a motor design that meets performance requirements and satisfies design constraints and operating limits, and that has been optimized with respect to a performance parameter selected by the user from a menu.

This volume of the User's Manual - Volume I (Technical Description) - gives the basis for the code computations, analytical developments, logic flow charts used in verification checks, and error messages. Volume II (User's Guide) contains the input and output dictionaries and their accompanying illustrations, along with other input instructions needed to execute the code. Volume III (Program Description) contains the subroutine descriptions and flow charts, cross-indices of common statements, subroutines and call statements.

SPOC was prepared for use by a motor designer. The user/designer controls the direction taken by the search through the inputs. Information used in the code must be provided by the designer, but no more is required than what must already be accumulated in order to prepare a detailed preliminary design---which is what this code will produce. It is not intended that this code replace final detailed stress, thermal, and combustion stability analyses; it will monitor certain stress, thermal and stability parameters in the search for an optimized design so that the final arrangement is more likely to pass detailed analyses. SPOC will do no more, nor will it do any less, than a good designer will do; but the code will do it much faster, thus enabling the designer to examine more approaches and more combinations than previously possible.

The code is operational on an IBM 4341 and a CDC 6600 computer, and the two versions are almost identical. Double precision statements required on the IBM computer have been deactivated in the CDC version, but they remain in the code. Arc sin and Arc cos functions are ARSIN and ARCOS in the IBM version, but are ASIN and ACOS in the CDC version; however, the latter can be kept for an IBM version if the H-extended compiler is employed. All coding is in FORTRAN IV language.

There are 115 subroutines. No external devices (other than a printer) are used. There is no overlaying of the code structure. Core storage requirements are 118K decimal words on the Thiokol IBM 4341 computer and 270K octal words on the AFRPL CDC 6600 computer.

Execution time depends on the user and the problem, and so there is no single representative time. About 8 seconds are required for a single pass through the analysis routine for a minimum option short burn time motor . About 19 seconds are needed for a problem that includes propellant formulation adjustments, cost and propellant structural analyses, impulse efficiency calculation and ballistic simulation at two temperatures. Typically, 300 to 500 calls to the analysis routine are needed to achieve an optimum solution.

Five AFRPL-supplied sample problems have been successfully solved with SPOC. These problems demonstrated many of the features of the code.

Another aspect of the demonstration phase of this project was a briefing for representives of propulsion and prime contractors held on 16 June 1981. This final version of the User's Manual (three volumes) supersedes the draft version distributed at that briefing.

Revisions made to the manual in the future will consist of added or replacement pages, identified with revision level and effective date; the original page numbering system will be preserved. A revision cover sheet will serve to transmit the changed/added pages and as a record of the changes. Revisions to the code will be transmitted in the same manner if the changes are not too extensive; a listing of the changed/added coding will be distributed. More extensive changes will require electronic transmittal through the AFRPL computer center.

MOTOR AND PROBLEM DEFINITION

SPOC includes models for five propellant grain configurations, three forward closure and two aft closure arrangements, and six nozzle configurations. Any combination of grain, closure, and nozzle may be selected except that a Type 4 grain (conocyl) may be used only with the Type 1 forward closure (ellipsoidal).

The analyses performed by SPOC are

Thermochemistry	Pressure vessel structural
Ballistic	Nozzle thermal and structural
Propellant structural	Trajectory
Weight	Combustion stability
Cost	Impulse efficiency

Flexibility has been provided for the user/designer so that the code may be tailored to enable varied problems to be solved. These choices are described in the following paragraphs.

Propellant Grain: choose one from those illustrated in Figure 1.

Type 1:	Star
Type 2:	Double web wagon wheel
Type 3:	Finocyl (slots in forward end)
Type 4:	Conocyl
Type 5:	Cylindrically perforated (CP)

Nozzle: choose one from those illustrated in Figure 2.

Type 1:	Thin shell, composite structure as the insulating ablative and support structure.
Type 2:	Thin shell support structure with insert and ablative insulator.
Type 3:	One-piece ablative; supersonic blast tube; constant diameter support structure.
Type 4:	One-piece ablative; supersonic blast tube; reduced diameter aft section.
Type 5:	Subsonic blast tube; without expansion cone.
Type 6:	Subsonic blast tube; with expansion cone.

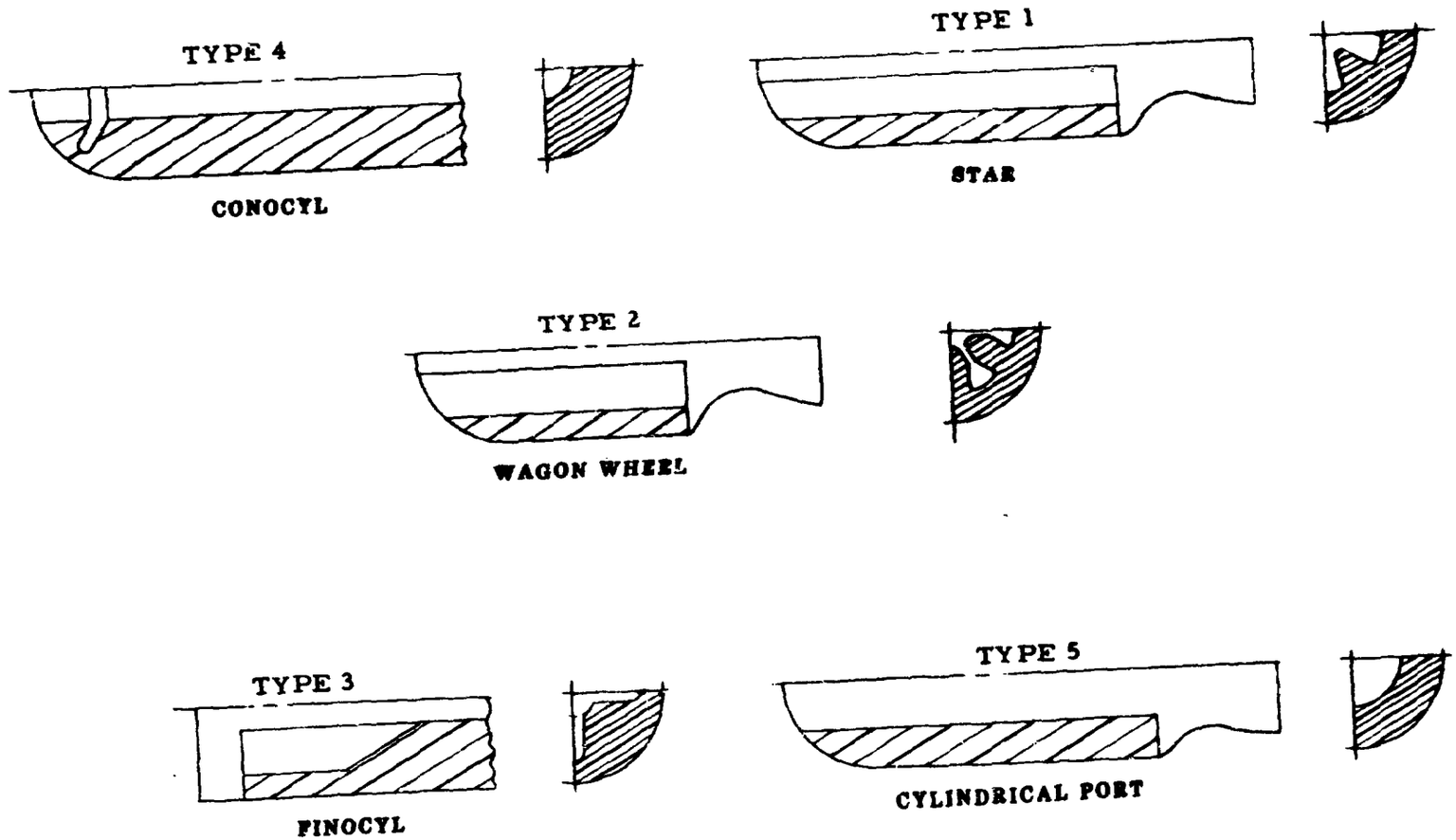
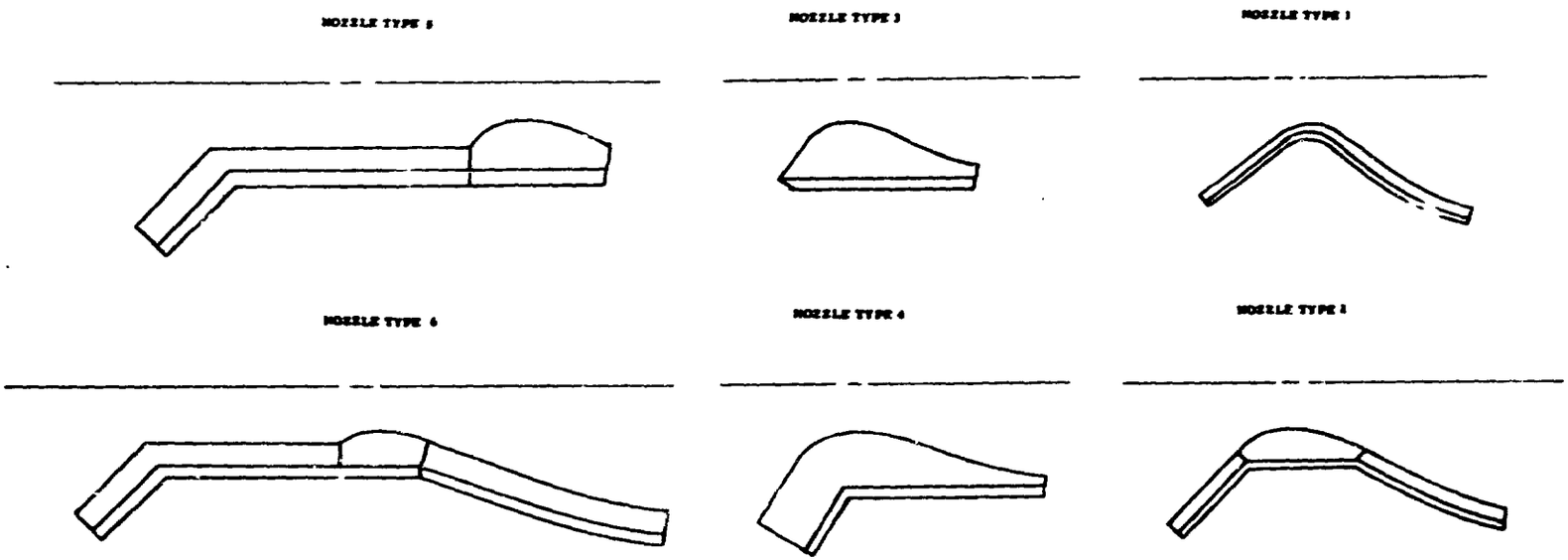


Figure 1. Propellant Grain Geometries Available in SPOC



All Exit Sections May be Conical or Contoured

Figure 2. Nozzle Configurations Available in SPOC

Forward Closure: choose one from those illustrated in Figure 3.

Type 1: Ellipsoidal

Type 2: Flat plate with closure secured with retaining ring

Type 3: Flat plate with closure integral with case

Aft Closure: choose one from those illustrated in Figure 4.

Type 1: Ellipsoidal

Type 2: None (aft closure formed by nozzle entrance section)

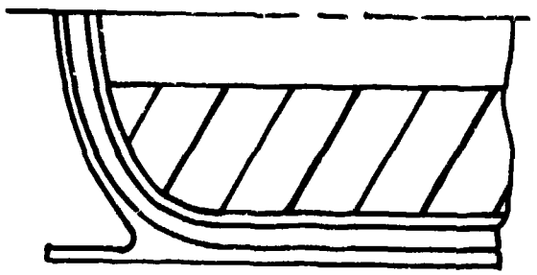
Other choices that must be made to define the problem are:

1. A propellant formulation may be input and adjusted as part of the optimization (FORMAD=T), in which case the thermochemistry routines are entered every time the design is evaluated (except for some internal by-passes to reduce execution time). Another option is to input a formulation but not adjust it (FORMIN=T), in which case the thermochemistry routines are entered only for the first evaluation in order to obtain basic propellant characteristics for the ballistic simulation. The third option is for the user to input the appropriate ballistic parameters rather than having the thermochemistry routines calculate them from a formulation (PROPIN=T). The proper combination of these three inputs is shown below (all default to F).

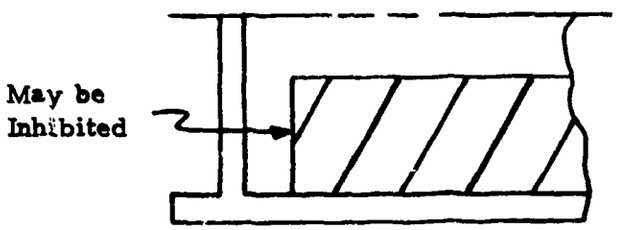
	MODE	FORMAD	FORMIN	PROPIN
(1)	Formulation input and adjusted during optimization	T	F	F
(2)	Formulation input, but not adjusted	F	T	F
(3)	User supplies required propellant characteristics	F	F	T

2. Impulse efficiency may be input by the user, calculated internally with the AFRPL SPP "empirical model", or calculated with a user-supplied model which must be installed in subroutine USEREF. EFMDL=T is the flag to show a user model has been supplied. SPPETA=T is the flag to specify the SPP model.

3. Propellant burn rate is calculated internally with the Vielle model, or with a user-supplied model which he must install in subroutine USERRB. RBMDL=T is the flag to show a user model has been supplied.

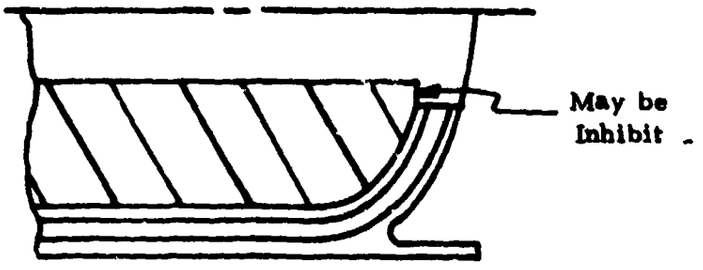


Ellipsoidal - Type 1

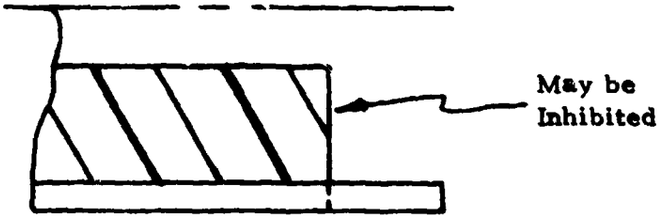


Flat Plate - Type 2 (Retaining Ring)
- Type 3 (Integral with Case)

Figure 3. Forward Closure Configurations Available in SPOC



Ellipsoidal - Type 1



None - Type 2 (Formed by Nozzle)

Figure 4. Aft Closure Configurations Available in SPOC

4. The propellant face on the forward end of a grain with a Type 2 or Type 3 forward closure and on the aft face of a grain with either a Type 1 or Type 2 aft closure may be inhibited through use of FWDINH=T or AFTINH=T, respectively.

5. Ballistic simulations will be performed at both the low temperature and high temperature conditions if different values are input for THI and TLO. Propellant structural analysis is performed at a different temperature (TPROP) than is the low temperature ballistic simulation. Pressure vessel structural analysis is performed at the high temperature condition. If THI is input equal to TLO, only one ballistic simulation is performed; propellant structural analysis is still performed at TPROP, and pressure vessel structural analysis is performed with the results of the single-temperature ballistic simulation (i.e., pressure not adjusted to some high temperature condition).

6. The optimization routine will adjust user-specified parameters in order to meet all performance requirements and satisfy all design constraints. In addition, the user may specify another parameter to be optimized by setting ICHOZE to one of the following.

0: None (default value)
1: Minimize cost
2: Minimize total motor weight
3: Maximize total impulse
4: Maximize total impulse-to-total weight ratio
5: Maximize burnout velocity

7. There are 36 parameters (not all on one problem) whose values can be adjusted by the optimization routine PATSH to achieve an optimum design (Table 1). Each of these must be specified by the user as "T" (maintain at input value) or "F" (do not maintain at input value, but adjust during pattern search). Default value is T (do not adjust).

8. A trajectory simulation (point mass, flat earth, ballistic trajectory) will be performed if specified by the user (FTRAJ=T). If ballistic simulations are performed at two temperatures (TLO and THI), then trajectory simulations are performed with each of the resultant thrust-time histories. In addition, the user must select a trajectory termination option.

9. Motor cost will be calculated with the Tri-Services cost model or with a user-supplied model. FCOST=T is the flag to specify the Tri-Services model; CSTMDL is the flag to show a user-supplied model has been provided.

10. Either a contoured or conical nozzle expansion section may be specified (CONTUR=T or CONTUR=F, respectively). If a conical exit section is selected, the initial half-angle of the expansion section (ALFA) must be input equal to the exit half-angle (ALFAEX).

TABLE 1

ADJUSTABLE VARIABLES AVAILABLE IN SPOC

Variable Number	Value of Variable: Variable Name	Value of Variable: Input Source(1)	Search Control: Control Name	Search Control: Input Source(1)	Definition
1	BIND	INGAMT	FBIND	INGFIX	Weight fraction of binder
2	FUEL	INGAMT	FFUEL	INGFIX	Weight fraction of fuel
3	OXA(1)	INGAMT	FOXA(1)	INGFIX	Weight fraction of oxidizer A, Size (1)
4	OXA(2)	INGAMT	FOXA(2)	INGFIX	Weight fraction of oxidizer A, Size (2)
5	OXA(3)	INGAMT	FOXA(3)	INGFIX	Weight fraction of oxidizer A, Size (3)
6	OXB(1)	INGAMT	FOXB(1)	INGFIX	Weight fraction of oxidizer B, Size (1)
7	OXB(2)	INGAMT	FOXB(2)	INGFIX	Weight fraction of oxidizer B, Size (2)
8	OXB(3)	INGAMT	FOXB(3)	INGFIX	Weight fraction of oxidizer B, Size (3)
9	RCATL	INGAMT	FRCL	INGFIX	Weight fraction of liquid rate catalyst
10	RCATS	INGAMT	FRCS	INGFIX	Weight fraction of solid rate catalyst
11	STAB	INGAMT	FSTAB	INGFIX	Weight fraction of combustion stabilizer
12	RE	NOZGEO	FDE	NOZGEO	Nozzle exit radius (Note 2)
13	RT	NOZGEO	FDTI	NOZGEO	Nozzle throat radius (Note 2)
14	R2A1	GRAIN3	FR2A1	GRAIN3	Propellant port radius at Plane 1 (Grain 3), and star point tip radius at Plane 1 (Grain 2)
15	R2A14	GRAINi	FR2A14	GRAINi	Propellant port radius at Plane 14 (Grains 3, 4 and 5)
16	R4A1	GRAIN3	FR4A1	GRAIN3	Propellant slot fillet radius at Plane 1 (Grain 3)
17	R5A1	GRAINi	FR5A1	GRAINi	Propellant slot depth radius at Plane 1 (Grain 3) and fillet radius between propellant tip and web (Grains 1 and 2)

(1) Namelist designation
(2) Input as appropriate diameter

Table 1

ADJUSTABLE VARIABLES AVAILABLE IN SPOC (contd.)

Variable Number	Value of Variable: Variable Name	Value of Variable: Input Source (1)	Search Control: Control Name	Search Control: Input Source (1)	Definition
18	ALPHA1	GRAINi	FALPA1	GRAINi	Angle on side of propellant slot (Grain 3), and included half-angle of start tip (Grain 1)
19	LSLOT	GRAIN3	FLSLOT	GRAIN3	Length of propellant slot (Grain 3)
20	LCP	GRAINi	FLCP	GRAINi	Length of cylindrically perforated grain section (Grains 3, 4, 5)
21	LCONE	GRAINi	FLCONE	GRAINi	Length of aft coned grain section (Grains 3, 4, 5)
22	ALFAEX	NOZGEO	FALFAX	NOZGEO	Nozzle exit half angle
23	TCASE	GRAINi	FTCASE	GRAINi	Case wall thickness, cylindrical section
24	RB70	BALLST	FRB70	BALLST	Propellant burn rate at 70°F, 1000 psia
25	XN	BALLST	FXN	BALLST	Pressure exponent in burn rate model
26	LCONEF	GRAIN5	FLCONF	GRAIN5	Length of forward coned grain section (Grain 5)
27	RMOTOR	GRAINi	FDMTR	GRAINi	Motor outside radius (Note 2)
28	R2A3	GRAIN4	FR2A3	GRAIN4	Propellant port radius at Plane 3 (Grain 4)
29	RTIP	GRAIN4	FRTIP	GRAIN4	Outboard radius of propellant slot (Grain 4)
30	ZED	GRAIN4	FZED	GRAIN4	Angle of slot with centerline (Grain 4)
31	LH	GRAIN4	FLH	GRAIN4	Length of forward propellant segment (Grain 4)
32	TAUW1	GRAINi	FTAUW1	GRAINi	Propellant web thickness (Grains 1 and 2)
33	LSA1	GRAINi	FLSA1	GRAINi	Star tip height at Plane 1 (Grains 1 and 2)
34	LSA14	GRAINi	FLSA14	GRAINi	Star tip height at Plane 14 (Grains 1 and 2)
35	LFWD	GRAINi	FLFWD	GRAINi	Length of forward untapered propellant section (Grains 1 and 2)
36	LTAPER	GRAINi	FLTAPR	GRAINi	Length of aft tapered propellant sections (Grains 1 and 2)
37	R2A5	GRAIN5	FR2A5	GRAIN5	Propellant port radius at Plane 5 (Grain 5)

11. Several analyses are by-passed completely unless the user specifies otherwise.

(a) Propellant structural analysis (PSTRUC=T)
(b) Combustion stability (FSTAB=T)
(c) Trajectory simulation (FTRAJ=T)
(d) SPP impulse efficiency (SPPETA=T)
(e) Thermochemistry (FORMAD=T or FORMIN=T)
(f) Cost (FCOST=T)

12. The user may provide models for certain parameters that are used in the analyses. A flag is set to show a user model has been loaded into a specified subroutine (T= model has been supplied).

Flag	Load in Subroutine	Parameter to Be Supplied
RBMDL	USERRB	Propellant burn rate, RATE (in/sec)
SEMDL	USERSE	Propellant nominal strain endurance, SENOM (in/in)
EOMMDL	USERRH	Propellant rheological property to be defined by user, EOM (units by user)
CSTMDL	USERCS	Motor cost, COST ($ or $/unit)
EFMDL	USEREF	Impulse efficiency, ETAISP (% x 0.01)
*	RSPNSE	Combustion response

*IRSPNS = 6 in namelist STABIN

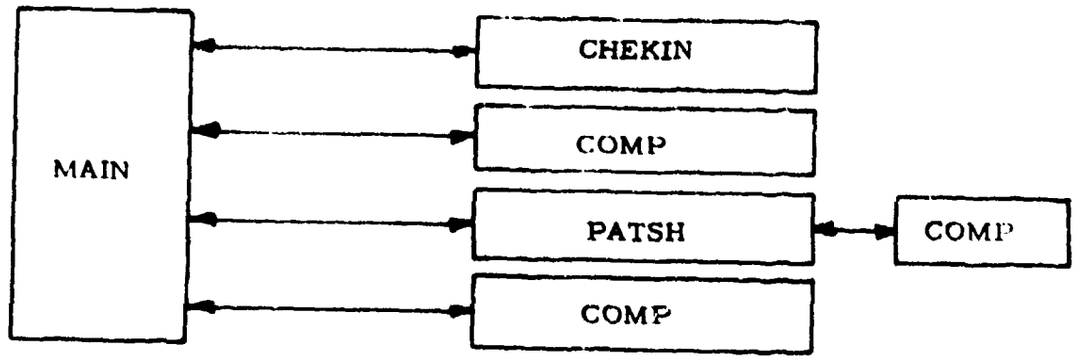
13. If a combustion stability analysis (Reference 7) is desired, the user can select one of five combustion response models (one is user-supplied) and can specify at how many modes stability margin is to be calculated.

COMPUTER CODE ARRANGEMENT

The computer code has an overall organization similar to that shown in Figure 5. There are three major subprograms (MAIN, COMP, and PATSH), whose functions are listed on Figure 5. Flow through the program will then be that of Figure 6. MAIN first reads and initializes various parameters and prints a narrative summary of the problem as defined by the user in subroutine CHECKIN. A call is then made to COMP for the first time in order to calculate performance of the motor with user-supplied initial values. Initial values of penalties are also calculated (in COMP) and all output is printed, after which the PRINT flag is turned off. MAIN then calls PATSH to adjust specified parameters in order to minimize the payoff parameter and penalties (described in detail in the PATSH module description in another report section). Each time PATSH adjust one or more of the specified parameters, COMP is called to calculate motor performance, payoff and associated penalties. PATSH builds a pattern and makes adjustments to minimize the OBJ function. When there is no further decrease in the payoff and penalties, the PRINT flag is turned on, COMP calculates the performance with the last set of adjusted parameters, and results are printed.

The executive subroutine COMP sets up the user- or PATSH-supplied inputs for the various analyses and simulations and passes the results of early analyses to later calculations when they are needed (Figure 7). For the first pass through COMP, where all analysis inputs are furnished by the user, the inputs are read in the specific subroutine to which the data applies. On all subsequent passes through COMP, the input data are either constant at the user-input value or are updated according to the PATSH adjustments. Write commands are also given within the individual subroutines.

The first call by COMP is to one of the grain dimension verification and setup subroutines (SETUP1 for Grain Type 1, SETUP2 for Grain Type 2, etc.); these subroutines verify the geometric validity of the incoming dimension set and calculates other dimensions needed by the ballistic simulation module. Subroutine NOZINP is called to perform the same function for the nozzle. If the problem involves a propellant formulation, subroutine TCHEM is called next to perform thermochemical analyses; results of the calculations are used in IMPEFF (impulse efficiency), SEC2SB (ballistic simulation), NOZL (nozzle thermal and structural analysis), and E488M2 (combustion stability). Subroutine IMPEFF is called next to furnish a value for impulse efficiency, if specified by the user.



MAIN: Reads control inputs; initializes some parameters; controls printout; calls search routine

COMP: Executive subroutine passes information between subroutines; calculates some penalties and overall objective function (OBJ); provides printout

PATSH: Adjusts specified parameters; evaluates changes in objective function (OBJ)

CHEKIN: Checks compatibility of problem definition. Prints narrative description of problem.

Figure 5. Overall Code Organization

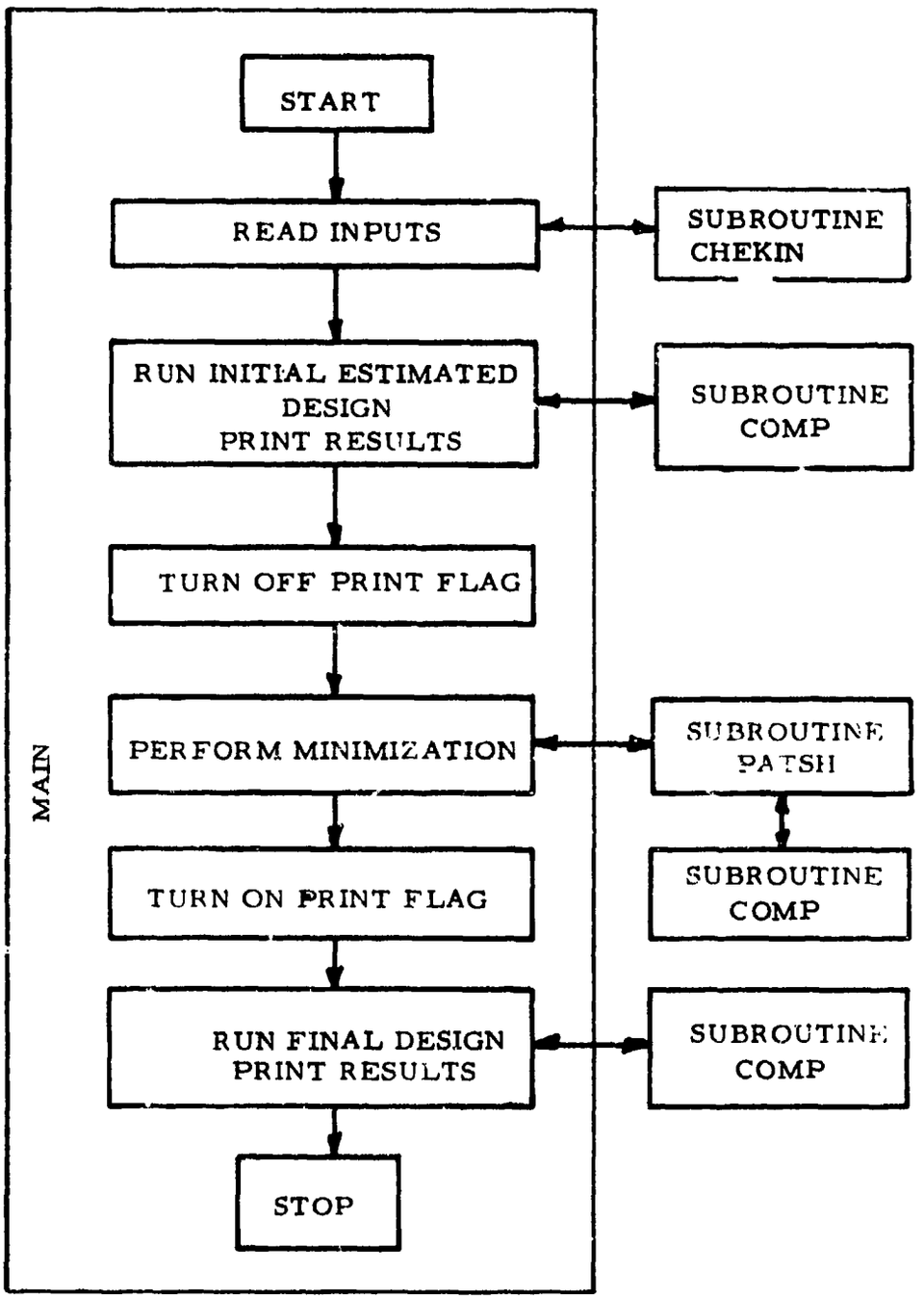


Figure 6. Overall Code Flowchart

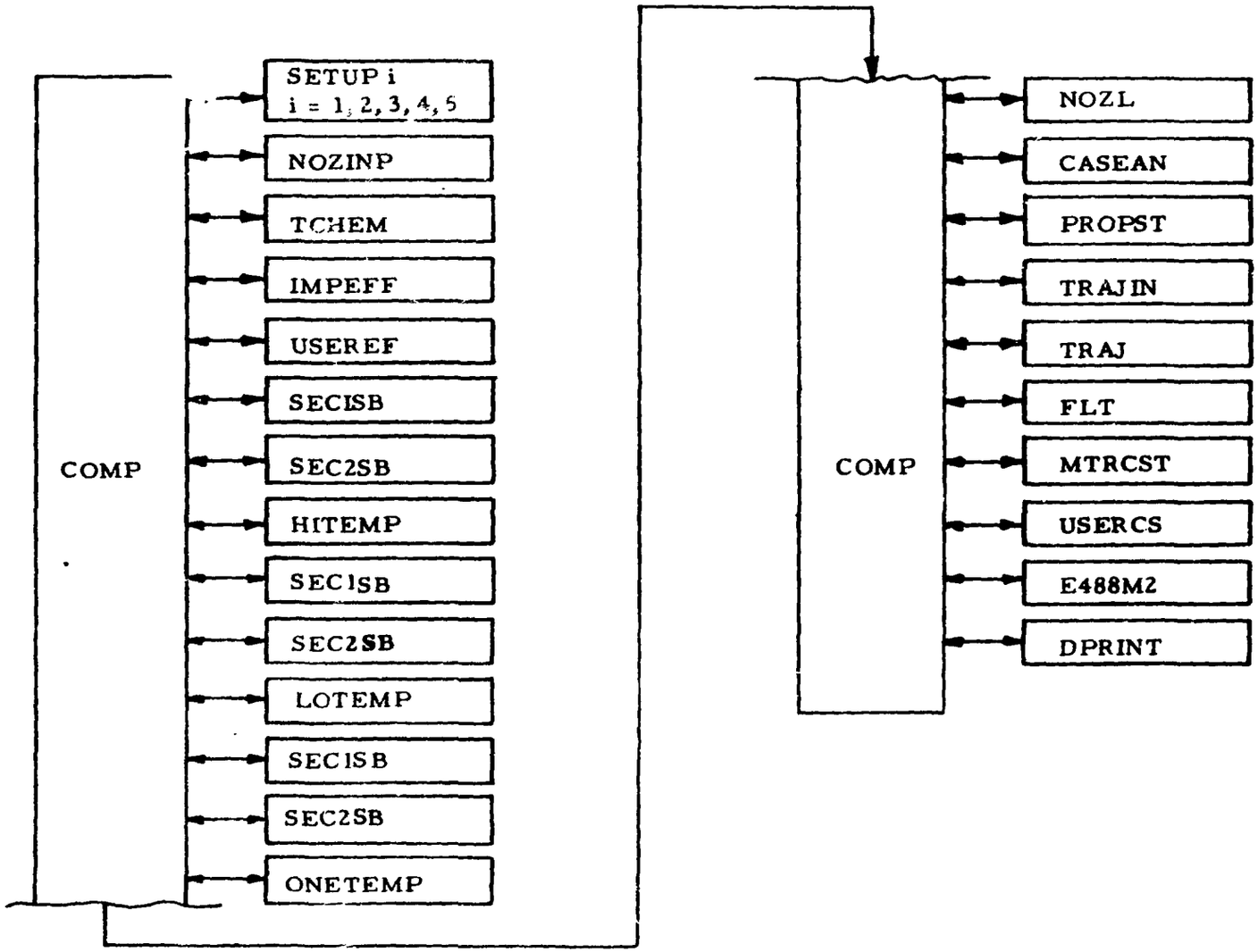


Figure 7. Organization of Executive Subroutine COMP.

Subroutines SEC1SB and SEC2SB make up the ballistic simulation module. The first time they are called, the input ballistic parameters have been set up (in COMP) for a grain conditioned to high temperature conditions. When the ballistic simulation is completed, subroutine HITEMP uses the results to calculate certain performance parameters and operating conditions associated with high temperature motor operation (e.g., design pressures, minimum burn time, etc). The predicted values are compared with user input limits and appropriate penalties are calculated. Next COMP sets up ballistic parameters for a simulation with the grain conditioned to low temperature, and then SEC1SB and SEC2SB are called again. Results of the low temperature simulation are analyzed in subroutine LOTEMP for performance parameters and operating conditions associated with low temperature motor operation. If the user wants only to study a problem via ballistic simulation at a single temperature, the second simulation is skipped and results of the first are analyzed (by making THI = TLO) in subroutine ONETMP (that combines the calculations of HITEMP and LOTEMP).

Once the results of the ballistic simulations(s) are available, nozzle thermal and structural analyses are performed in subroutine NOZL, pressure vessel structural analyses are performed in subroutine CASEAN, and (if specified by the user) propellant structural analyses are performed in subroutine PROPST. The user may also command a trajectory simulation. Subroutine TRAJIN acts as a mini-executive subroutine to control the trajectory simulations for a one-or two-temperature problem. Motor cost is calculated in subroutine COST, and combustion stability characteristics are determined in subroutine E488M2, if specified by the user.

OPTIMIZATION PROCESS

SPOC combines computer models for solid rocket motor performance prediction and design analyses with a numerical parameter optimization technique. As stated in Reference 1, this combination requires an understanding of both areas. The following discussion was taken from Reference 1 because approaches taken in the TACMOP and SPOC codes are very similar, even though the codes have different end objectives.

In order to eliminate misinterpretation, several terms used throughout the remainder of the discussion are defined below:

Performance requirement - A measure of acceptable system operation in accomplishing its intended purpose. For solid-propellant rocket motors, performance requirements typically include such items as range, velocity, or payload delivered to a specified end condition. In SPOC, performance requirements are expressed as total impulse, impulse-to-weight ratio, etc., as well as the ultimate end-item requirements listed above; however, the trajectory simulation in SPOC is not intended for complex maneuvering trajectories, and so SPOC should be used in conjunction with more sophisticated trajectory simulations.

Design parameter - A length, angle, or material property used in describing a particular design, such as propellant grain length, case diameter, nozzle half angle, or propellant burning rate.

Design constraint - A limit imposed directly or indirectly on the allowable values of a design parameter, such as maximum length, maximum nozzle divergence angle, maximum propellant web fraction, or minimum port-to-throat area ratio.

Operating limit - A maximum or minimum acceptable level for a condition produced by motor operation, such as maximum acceleration, minimum pressure, or maximum velocity.

Payoff - The quantity selected as the maximized or minimized variable during the optimization process, such as maximum range. In SPOC, corresponding payoffs are total impulse, motor weight, cost, etc.

Penalty function - A function corresponding to a particular performance requirement, design constraint, or operating limit, having zero value when the requirement, constraint, or limit is satisfied by the design being evaluated, and having a non-zero value proportional to the amount of violation of the particular requirement when it is not satisfied.

Objective function - A single-valued function for a particular design representing both the payoff value and any non-zero penalty function values associated with that design.

The design problem consists of finding a set of design parameter values that produce a system with maximum (or minimum) payoff, subject to meeting all performance requirements, design constraints, and operating limits (i.e., all penalties non-zero).

Parameter Optimization Scheme

The optimization routine used in SPOC is the PATSH (Pattern Search) subroutine developed by D. E. Whitney at the Massachusetts Institute of Technology (Reference 2). This subroutine performs an unconstrained non-linear optimization with the direct pattern search algorithm of Hooke and Jeeves (Reference 3) in a highly compact FORTRAN subroutine. This particular scheme has delivered good performance when compared with other methods (References 4 and 5). Direct search methods operate on the basis of always saving the most optimum point encountered as the new "base point", or point about which further searches are made.

The Hooke and Jeeves direct search is unconstrained in itself; however as applied here the problem is constrained through the manner in which the single-valued objective function is calculated. Limits on the magnitude of the decision variables, as well as analytical relationships between the decision variables, are imposed through the use of individual penalties.

PATSH operates by "moving" (adjusting) the decision variables

$$\underline{X}^{i+1} = \underline{X}^{i} + (0.05)(DEL)\,\underline{X}^{i}$$

where $\underline{X}^{i}$ = current decision variable set

$\underline{X}^{i+1}$ = new decision variable set

DEL = step size multiplier

These are two results of moves. A successful move produces a reduction in the objective function OBJ. A move is a failure when there is no reduction in OBJ. Moves can be accomplished in one of two ways. An exploratory move consists of changing the value of only one decision variable and evaluating OBJ. A pattern move occurs when values of all decision variables are changed simultaneously according to the information derived from exploratory moves. During a pattern move, each variable is changed by an amount proportional to the difference between its value at the current base point and its value at the immediately preceeding base point.

The logic flow of PATSH is presented in Figure 8. PATSH begins the search by calling the computational program (subroutine COMP) with the initial user-supplied parameter set to establish the initial base point; this produces an analysis identical to the first call to COMP by MAIN. In the call to the computational package, PATSH sends the current parameter set to the package and receives back the objective function value corresponding to that parameter set. After evaluation of the initial base point, PATSH begins a series of exploratory moves, varying the value of each parameter in the following systematic manner:

(1) Vary the parameter in the positive direction by five percent and evaluate the objective function. If the objective function decreased in value from the base point, keep the parameter change, save the current total parameter set as the new base point, and go to the next parameter.

(2) If the positive variation of the parameter did not result in a reduction of the objective function, decrease the original value of the parameter by five percent and evaluate the objective function. If the objective function decreases, a new base point is established; if not, reset the parameter to its original value and go on to the next parameter.

If the preceding exploratory move for this parameter did not produce a reduction in objective function when the parameter was varied positively, but did when it was varied negatively, then the next exploratory move tries the negative direction first (and then the positive if no improvement is seen).

(3) When all parameters have been varied one at a time, either a new base point will have been established, or the original base point will be retained if none of the exploratory moves resulted in an improvement. If an improvement has been achieved, the exploratory moves have established a pattern - change the first parameter positive, do not change the second

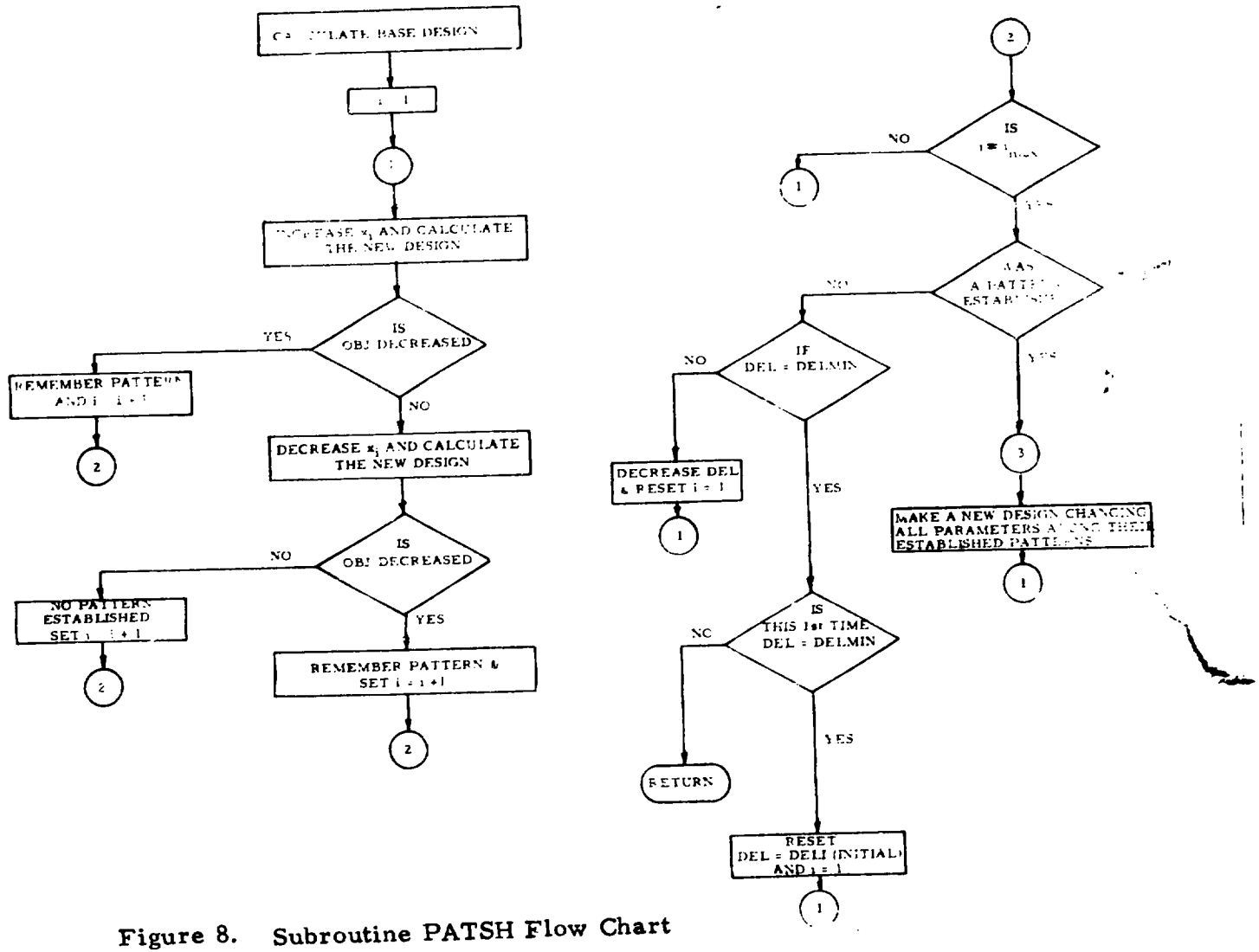


Figure 8. Subroutine PATSH Flow Chart

parameter, change the third parameter negative, etc. - from which a pattern move can be taken. A pattern step is one in which all parameters producing an improvement during the exploratory moves are varied simultaneously. If no improvement was obtained during the exploratory moves (i.e., the previous base point has been retained), the step size is reduced to one-half its current value and the exploratory moves are repeated.

The pattern step may or may not produce a decrease in the objective function over the current base point. PATSH does not immediately reject a pattern move that results in an increase in the objective function. Each pattern move is followed by another set of exploratory moves, using the pattern move parameter set as the "base" point. If none of these exploratory moves provides a lower objective function value than the base point value prior to the pattern move, the previous point is retained, and a set of exploratory moves is made about it. If this set does not produce a reduction in OBJ, the step size is reduced for a new set of exploratory moves about the current base point. An improvement in the objective function by any means (exploratory move or pattern move) is always retained as the new base point. The search is assumed to be converged when, through repeated efforts to obtain improvements, the step size is reduced from its original value to the minimum value specified by user input (DELMIN). Such a process may appear to be succeeding by failing to achieve any better point; however, the final set of exploratory moves clearly demonstrates no improvement in the objective function by perturbing all of the parameters in either direction. This is similar to evaluating through a finite difference method, the first-order partial derivatives of the objective function with respect to the design parameters. Any error in obtaining an optimum would be contained within the minimum step size used for the final exploratory moves.

When using numerical optimization techniques, there is always concern over whether the true, or global, optimum has been reached, or whether a local optimum is the result. No guarantee exists that the solution is not a local optimum. The only way to gain a feeling of confidence in the solution (if it is in doubt) is to use different starting points (i.e., different initial (user-supplied) parameter sets), and to determine whether or not the same solution is reached each time. The possibility of local optima is a function of the problem to be solved. Some problems with highly complex constraints may have a number of local optima while many problems have only one global optimum. Keep in mind that, even though the solution may be suspected to be a local optimum, if all penalties are zero, then the solution is a valid design; some improvement in the payoff parameter may be realized, and that can be determined only through starting the search with a different input set.

Performance Requirement, Design Constraint, and Operating Limit Satisfaction

The optimization routine, PATSH, operates by minimizing a single-valued objective function. This single value must reflect the pay-off quantity (which is multiplied by -1 if maximization is desired) and the effectiveness of the design in meeting the performance requirements, design constraints, and operating limits. This has been accomplished by incorporating a penalty function scheme such that

$$OBJ = P + \sum_{i=1}^{n} F_i$$

where OBJ is the single-valued objective function minimized by PATSH, P is the payoff quantity, and the F_i are individual penalty functions for each of the performance requirements, design constraints, and operating limits (all of which are considered as constraints on the optimization process, and will be referred to as such for the remainder of this discussion). Two basic types of constraints exist, inequality constraints and equality constraints. Each constraint can be expressed in one of the following forms:

$$g_i \leq 0,$$

$$g_i \geq 0,$$

or

$$g_i = 0.$$

For example, a maximum acceleration constraint would be expressed as

$$g_a = a_{max} - a_{max,\ req},$$

where a_{max} is the maximum acceleration produced by the design being evaluated and $a_{max,\ req}$ is the required maximum. Clearly, the constraint is met when $g_a \leq 0$. Similarly, a minimum burnout velocity constraint is expressed as

$$g_V = V_{BO} - V_{BO,\ req}$$

This constraint is met when $g_V \geq 0$. Finally, a requirement for an exact burn time would be expressed as

$$g_{t_B} = t_B - t_{B,\ req}$$

and the constraint would be satisfied when $g_{t_B} = 0$.

Penalty function values for the three examples given above would be calculated as

$$F_a = \begin{cases} 0 & , \ g_a \leq 0 \\ g_a^2 \ S_a, & g_a > 0 \end{cases}$$

$$F_V = \begin{cases} 0 & , \ g_V \geq 0 \\ g_V^2 \ S_V, & g_V < 0 \end{cases}$$

$$F_{t_B} = \begin{cases} 0 & , \ g_{t_B} = 0 \\ g_{t_B}^2 \ S_{t_B}, & g_{t_B} \neq 0 \end{cases}$$

The S_a, S_V, and S_{t_B} are scale factors used to normalize constraint violation penalties to an appropriate level with respect to the payoff quantity. The choice of this form for the penalty functions provides a penalty value that can be scaled to relatively small values for minor violations with rapidly increasing (second order) value for larger violations. Constraint enforcement in this manner can be thought of as a "soft" constraint (i. e., minor violations are not totally excluded from the solution). Certain limits on design parameter values are enforced as "hard" constraints. An attempt by the optimizer routine to specify a design parameter value which violates a "hard" constraint results in the specified value being overridden with the limiting value and the generation of a penalty function proportional to the attempted violation. An example of a "soft" constraint is the upper limit on propellant web fraction, because a web fraction slightly greater than the limit may be acceptable if it produces greater improvements elsewhere. An example of a "hard" constraint is the length of one part of the motor, because a length of less-than-zero is physically meaningless (and can be computationally misleading).

Adjustable Variables

There are 36 variables in SPOC which may be adjusted by PATSH to obtain an optimum design (Table 1). However, not all of the decision variables can be adjusted in any one given problem because some are peculiar to certain grain geometries. The decision variables fall into these categories

- Propellant grain cross-section dimensions
- Propellant grain lengths

- o Propellant ingredient relative weights
- o Propellant ballistic characteristics (burn rate and performance level, the latter as influenced by ingredient amounts)
- o Nozzle dimensions
- o Miscellaneous (motor diameter, case cylindrical wall thickness)

PERFORMANCE REQUIREMENTS

The following performance parameters are driven toward user-input requirements. Penalties are calculated for not meeting each requirements. Default values provided in the code prevent the penalties from being activated unless the user chooses to enforce the requirement. The accompanying parenthetical expressions give the appropriate limit.

- Total impulse (lower three-sigma value at low temperature)
- Total motor weight (maximum nominal)
- Ignition thrust (lower three-sigma value at low temperature)
- Ignition thrust (upper three-sigma value at high temperature)
- Burn time (lower three-sigma value at high temperature)
- Burn time (upper three-sigma value at low temperature)
- Axial acceleration (maximum nominal at high temperature)
- Change in velocity (minimum nominal value at low temperature)
- Time-to-target (maximum nominal value at low temperature)
- Impact (or termination) velocity (minimum nominal value at low temperature)

Those requirements that are shown above to apply to a particular grain conditioning temperature condition can also be enforced with a one-temperature problem.

DESIGN CONSTRAINTS AND OPERATING LIMITS

Design constraints and operating limits that are enforced in the SPOC are:

- Case, closure and nozzle support thickness (sufficient for maximum expected operating pressure plus safety factor)
- Case and nozzle structure wall thickness (≥ manufacturing limit)
- Nozzle ablative thickness (≥ that required for char, ablation and thermal protection)
- Propellant strain margin of safety during low temperature storage in both CP and valley sections of grain (≥ 0)
- Propellant strain at low temperature ignition pressurization (≤ input maximum)
- Propellant web fraction (≤ maximum based on design experience)
- Propellant thickness under propellant valley (≥ manufacturing limit)
- Propellant total solids (between maximum and minimum limits)
- Propellant burn rate and pressure exponent (between maximum and minimum limits)
- Burn rate catalyst and fuel contents (≤ maximum based on experience)
- Combustion gas Mach number in port at low temperature (nominal ≤ maximum based on experience)
- Chamber pressure at high temperature (nominal ≤ maximum based on experience)
- Geometrically valid (compatible) propellant grain cross-section dimensions
- Lengths and thicknesses greater than zero

- Motor dimensions (length ≤ maximum, nozzle exit diameter ≤ maximum, case aft opening radius = nozzle entrance radius, nozzle blast tube length and diameter = requirement)
- Geometrically valid (compatible) nozzle dimensions
- Longitudinal combustion stability

PAYOFF PARAMETERS

The PAYOFF parameters from which the user can select one to be minimized during any given machine submission are

- None
- Total motor cost (minimize)
- Total motor weight (minimize)
- Total impulse (maximize)(1)
- Total impulse-to-total motor weight ratio (maximize)(1)
- Burnout Velocity (maximize)(1)

(1) PATSH will minimize the product of minus one times the value of this parameter, which produces a maximization of the parameters.

LIMITATIONS AND ACCURACY

The purpose of this discussion is to summarize some general limitations of the code and to provide estimates of the accuracy of the results.

LIMITATIONS

Some limitations on the use of SPOC are inherent in the assumptions employed during original development of the analysis and simulation modules; these assumptions are given in the discussions of the individual modules and their impact on a given problem solution is best left to the user.

Basically, there are no restrictions on the size of the motor which may be analyzed with SPOC. Small motors operating at high pressure could possibly enter the regime where thin-wall pressure vessel equations should be replaced by thick-wall relationships; it is up to the user to recognize this situation. The cylindrical section of a motor employing elliptical closures (forward or aft, or both) cannot be reduced to zero length because of how the grain geometry is described to the ballistic simulation module; the minimum length attainable is between one and two grain web thicknesses. As for large motors, there are no restrictions.

All volumes and concomitant weights are calculated from exact geometric relationships; there are no internal empiricisms to estimate weights. Weights not amenable to direct calculation in an optimization code (e.g., igniter, safe-and-arm device, wings, etc.) are user-supplied values.

Of necessity, some of the analysis routines are somewhat simplified, as would be expected when operating in a preliminary design mode; however, all analysis routines are industry-accepted methods.

(a) Propellant strain is calculated under plane-strain conditions. Thus end-effects and three-dimensional effects during abrupt configuration changes are not acoounted for.

(b) Membrane stresses in the ellipsoidal pressure vessel closures (Type 1) are calculated at the motor centerline which provides a satisfactory estimate of the required closure thickness elsewhere. Bending stresses at the closure-to-cylindrical shell junction are not considered.

(c) Bending at the closure-to-cylindrical shell junction is considered for the Type 3 forward closure (that features a flat plate closure integral with the cylindrical shell) as long as material response is elastic. Transitions between the cylindrical shell and the integral flat plate (i.e., radii or gradually increasing cylindrical wall thickness in the vicinity of the closure) are not included in stress estimates or volume calculations.

(d) The user must input a heat-transfer coefficient for each of the three nozzle ablative materials, which means that the coefficient is constant for all flow conditions to which a particular material is exposed.

There are dimensional mismatches at case-to-closure tangent points and case-to-nozzle joints in order to allow the user complete flexibility in choosing his motor arrangement and to make the computations more manageable; however, the results of these mismatches on predicted ballistic performance and weights is thought to be minimal. Figure 9 shows the potential mismatch between the liner inner surface at the closure-to-case cylindrical section interface; there are two ways that this mismatch can occur, and both are considered when the grain outer dimensions are established for the ballistic simulation. Figure 10 shows the potential mismatch of the pressure vessel outer surface at the closure-to-case cylindrical section interface. The inner surfaces of the closure and cylindrical section exactly match at the tangent point. Then the required closure thickness (TCLOF) is calculated after the ballistic simulation, and the cylindrical section thickness (TCASE) is a PATSH-adjusted parameter that eventually is satisfactory for the maximum pressure. Thus the outer surface of the pressure vessel could have a discontinuity at the tangent point. The thrust skirt is also shown in Figure 10 to show that its mating surface is the cylindrical section outer surface. Obviously, the degrees of mismatch shown in Figures 9 and 10 are greatly exaggerated for clarity; their effects on weights is negligible.

Another mismatch that always occurs is shown in Figure 11. The case opening radius (RNOZEN) always (eventually) is equal to the nozzle entrance radius, so there is no mismatch there. However, the nozzle ablative and structural support calculations are performed normal to the internal surface, so that part of the nozzle coincides with the case as shown by the shaded area in Figure 11; this "duplication" of volume provides an allowance for the nozzle attachment flange.

The trajectory simulation employs a point-mass missile flying a two-dimensional path in the altitude-range plane over a flat earth. Forces modeled are restricted to thrust, drag and weight (i.e., lift is always zero), and angle of attack is always zero. The trajectory simulation is intended as a supplementary evaluation tool (unless, of course, this model accurately describes the problem under consideration).

A two-dimensional plane-strain model is used to calculate propellant strain due to low-temperature storage and ignition pressurization. Such a model accurately describes the propellant behavior at a point mid-way along the grain length when the grain length-to-diameter ratio (L/D) is equal to or greater than about seven. For $L/D<7$, or for locations near the grain terminations, the plane-strain models give very conservative predictions because the end effects (three-dimensional) that relieve the strain are not accounted for in SPOC. Strains predicted for a propellant valley or slot will also be conservative near the ends or for short slots.

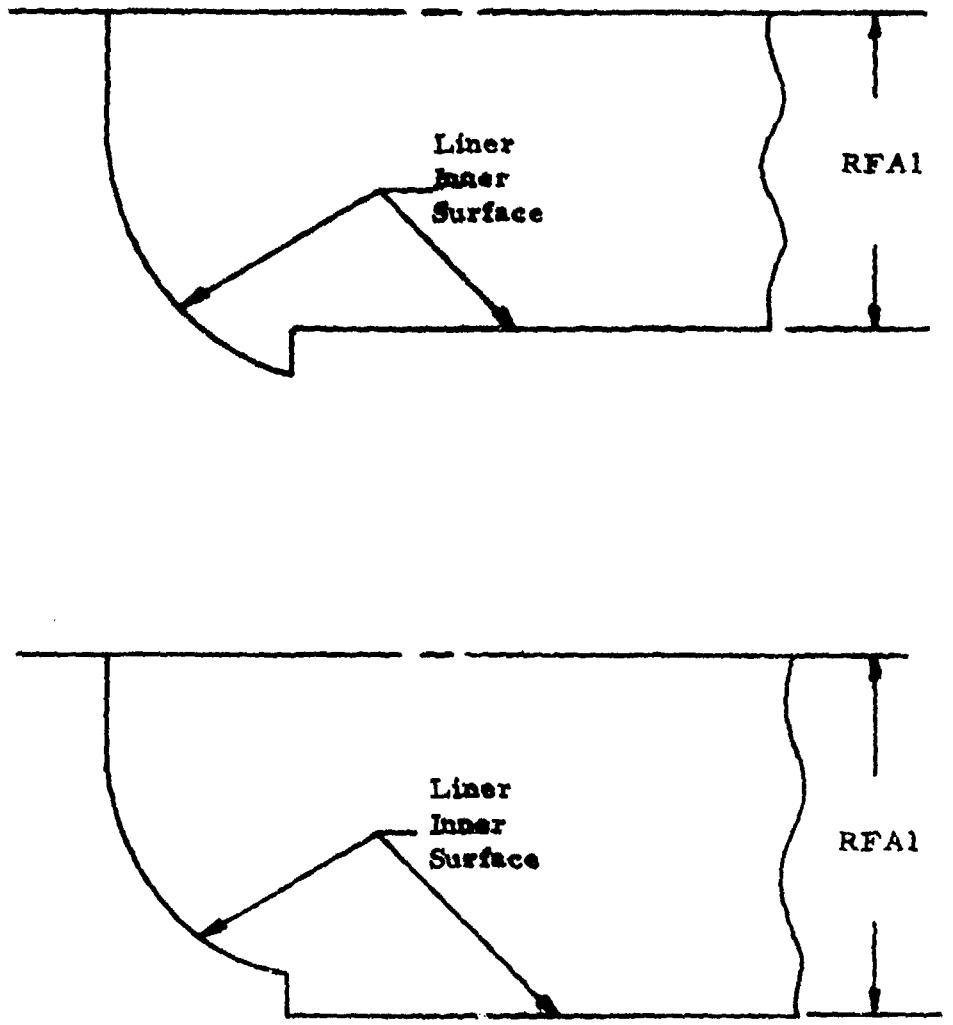


Figure 9. Potential Dimensional Mis-match of Liner Inner Surface

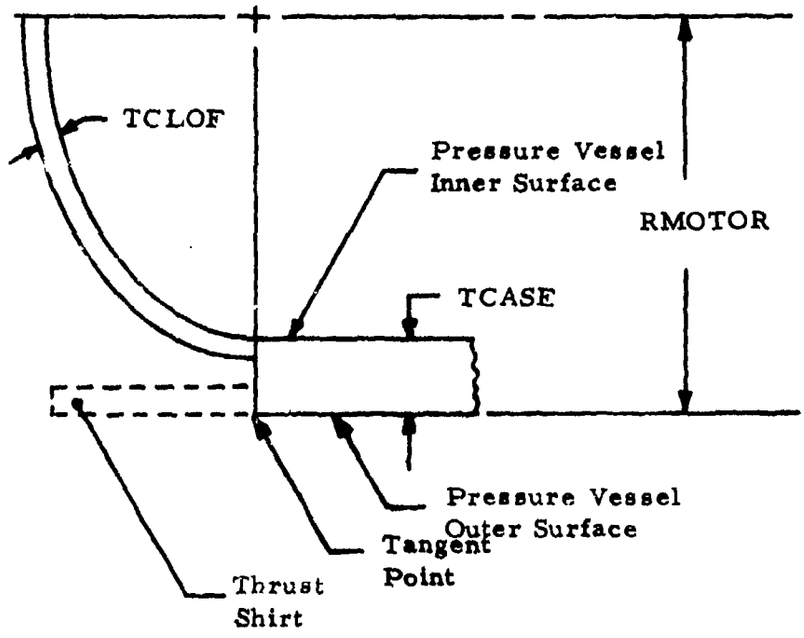


Figure 10. Potential Dimensional Mis-match of Pressure Vessel Outer Surface

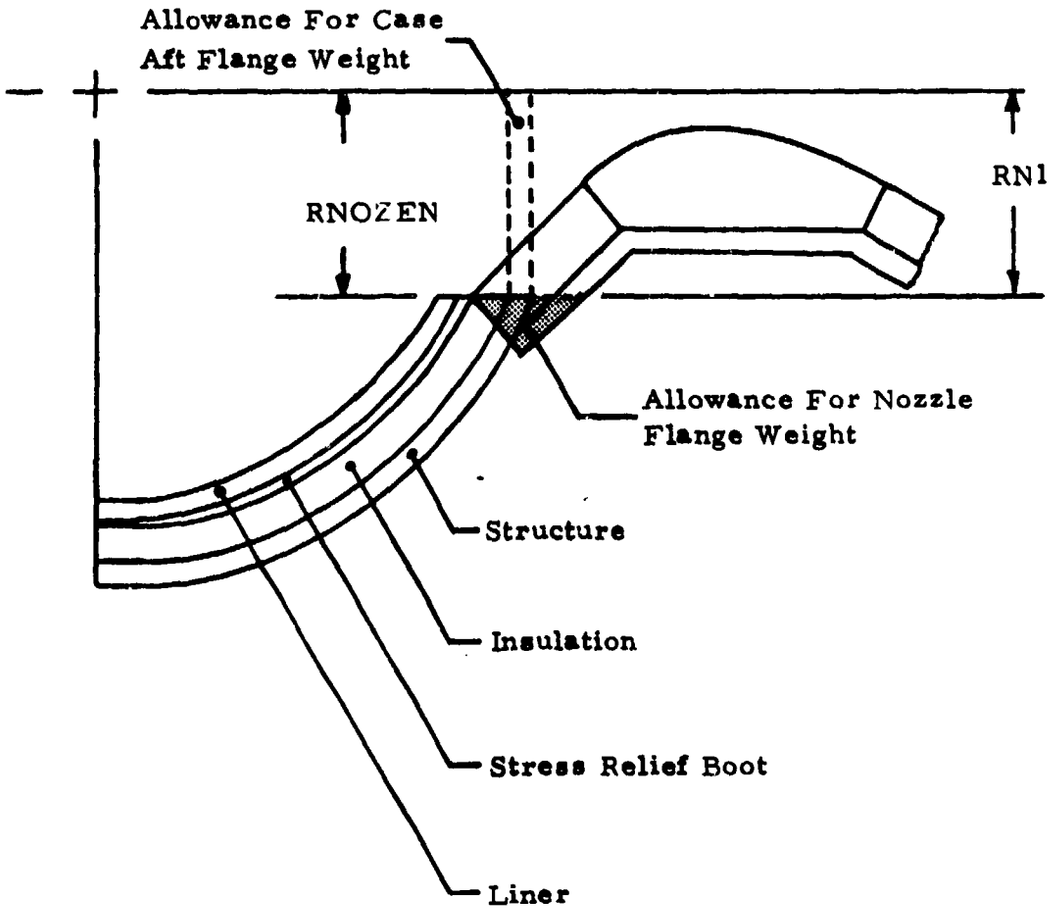


Figure 11. Dimensional Mismatch at Case-to-Nozzle Interface

The propellant structural analysis is not conservative at the hinge points of stress relief flaps and at the transition between propellant slots and CP regions. Both of these areas represent highly three-dimensional conditions that are not amenable to preliminary design calculations used in SPOC. Consequently, there is the inherent assumption that the bore conditions are the critical locations. Provisions have been made to include volume and weight allowances for stress relief boots in ellipsoidal closures, even though their final configuration is dependent on more detailed analyses. The transition section between slots and cylindrical port may require a special configuration to limit imposed strains; another way to achieve the same results is to specify about 7 degrees as the angle on the side of the slot (ALPHA1) of a finocyl grain (Type 3).

Thermal strain in the propellant due to low-temperature storage is compared with design strain endurance (nominal strain endurance reduced for mix-to-mix variations and aging degradation). Strain induced by ignition pressurization is compared with a user-input maximum limit. This latter limit should be derived from tests that measure strain capability at rapid strain rate (to simulate ignition pressurization) on test specimens conditioned to the design low temperature and already strained to the level that will be induced by low temperature storage.

ACCURACY OF CODE

There are three levels of accuracy to consider in the evaluation of a computer code. First, the user must decide how well the mathematical equations model the reality of a particular problem. Second is the computational accuracy, or how faithfully the programmer has carried out the mathematical manipulations. Finally, and totally under the control of the user, is the accuracy of the input data. Only the first two levels will be discussed here.

Accuracy of the mathematical models is paramount in the overall accuracy of a code. The several analysis and simulation modules are discussed separately in the following list:

Module	Estimated Accuracy of Model
Ballistic simulation	±3% total impulse ±5% maximum pressure General qualitative assessment based on experience
Weight estimates	±2% General qualitative assessment based on experience
Propellant theoretical characteristics	Essentially error free. Uses NASA-Lewis thermochemical analysis (Reference 11).
Combustion stability	Based on AFRPL Standard Stability Code (Reference 19).
Combustion efficiency	Based on AFRPL Solid Propellant Prediction Code (Reference 12).
Motor costs	Based on Tri-Services Rocket Motor Trade-off Study for steel cases (Reference 18).
Trajectory simulation	Estimated to be very high, provided the problem is adequately described by the model. See discussion above.
Propellant structural analysis	Strain calculation "very accurate" in center of motor with LD > 7 (probably within 10%). For location near ends of long motor or for L/D < 7, calculated strains are conservative, with degree of conservatism depending on problem.
Pressure vessel structural analysis	Estimated to be conservative by approximately 15%.

The computational accuracy of the code is extremely high. Iteration schemes in the ballistics simulation and grain subroutines require convergence to within 0.0 or less. The trajectory simulation uses an industry-accepted technique. Thus it is felt that the mathematical models have been faithfully computed.

MOTOR GEOMETRY

This section of Volume I provides more detailed discussion of the motor geometry aspects of SPOC, as differentiated from descriptions of the various analysis routines. As with much of the manual, some information given here applies to other parts of the code, and so there is some duplication.

The user may choose one from each of three forward closure arrangements, two aft closure arrangements and six nozzle configurations to describe the basic motor geometry. All of these, in any combination, may be used with any of the five propellant grain configurations, except that a conocyl grain can be employed only with an ellipsoidal (Type 1) forward closure (simply because of the definition of a conocyl grain).

FORWARD CLOSURE

There are two basic forward closure arrangements: an ellipsoid and a flat plate. The ellipsoidal closure is Type 1. The flat plate closure provides two types; one has the closure secured in the case by a separate retaining ring (Type 2) and the other has the closure integral with the cylindrical portion of the case (Type 3).

Type 1 Forward Closure

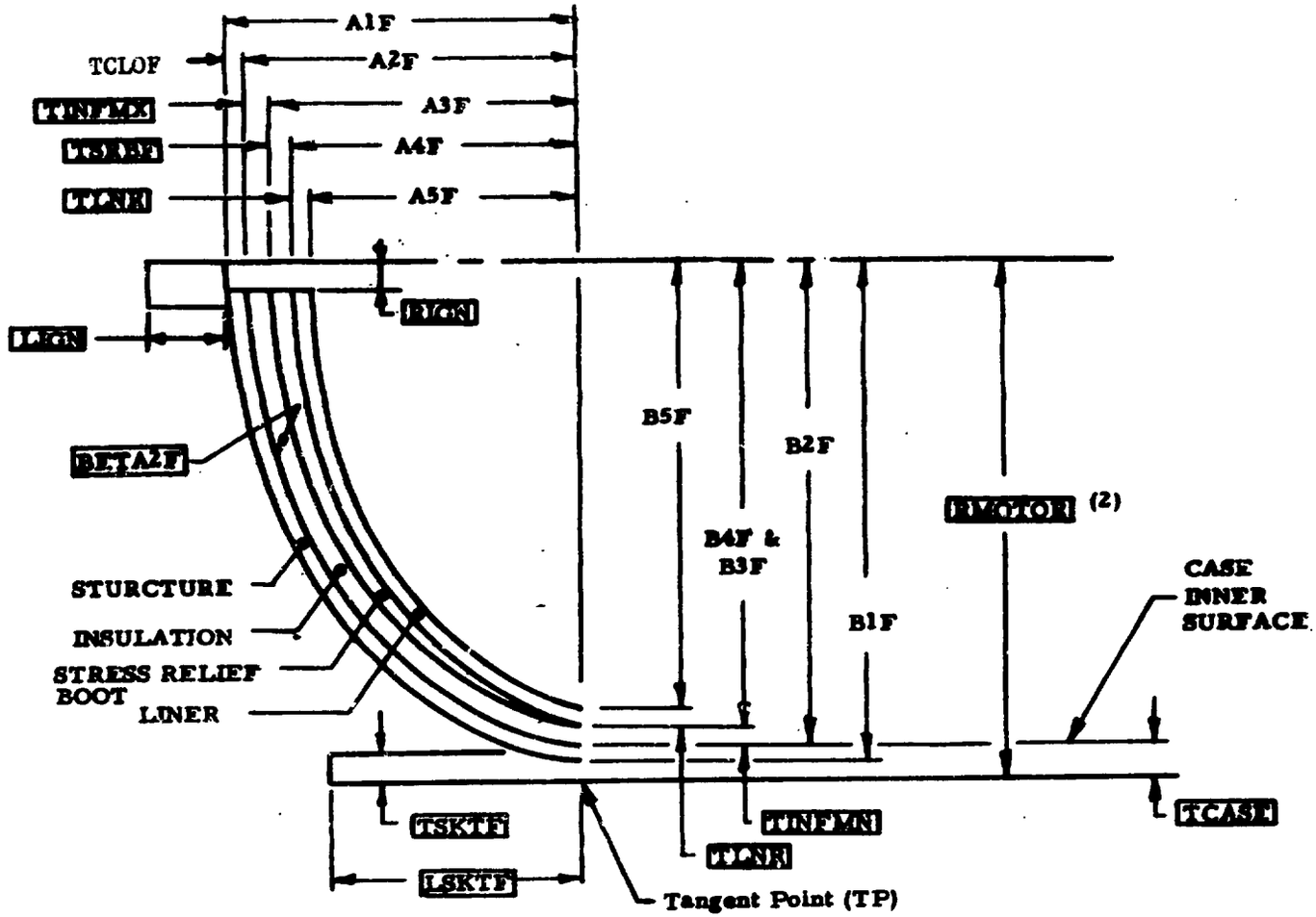
The controlling interface between the Type 1 forward closure and the cylindrical portion of the case is at the tangent point of the ellipsoid and the cylinder on the inner surface of the pressure vessel (Figure 12); all else derives from this. The semi-major axis is found from

$$B2F = RMOTOR - TCASE \tag{1}$$

Then using the input ellipse ratio, the semi-minor axis of the controlling interface surface is

$$A2F = B2F/BETA2F \tag{2}$$

These two parameters then are used to define the semi-major and semi-minor axes of other ellipsoids using input and calculated material thickness (see Figure 12).



(1) Dimensions shown in blocks are input; others are output
(2) Input as appropriate diameter

Figure 12. Forward Closure Type 1 (Ellipsoid)

$$A1F = A2F + TCLOF \quad (3)$$
$$A3F = A2F - TINFMX \quad (4)$$
$$A4F = A3F - TSRBF \quad (5)$$
$$A5F = A4F - TLNR \quad (6)$$

$$B1F = B2F + TCLOF \quad (7)$$
$$B3F = B2F - TINFMN \quad (8)$$
$$B4F = B3F$$
$$B5F = B4F - TLNR \quad (10)$$

It is assumed that the subsequent bodies thus described are ellipsoids, which is not so mathematically, but in reality the difference is very small for any situation where the thicknesses used in Eqs. (3) - (10) are small compared with the semi-minor axis.

The thicknesses TINFMX, TSRBF, and TLNR are defined at the igniter opening RIGN; however, to establish the ellipse parameters, these thicknesses are assumed to act at the motor centerline. This approach does not introduce significant error because the igniter opening is usually small with respect to the motor sizes.

The closure thickness, TCLOF, is calculated after the ballistic simulation is performed so that the stress analysis may use the design pressure for the current pass through COMP. The other ellipsoidal surfaces are defined prior to the ballistic simulation because they are needed to describe the propellant outer surface in the closure for the ballistic simulation routine. Radii (RECH) and concomitant lengths (HECH) measured forward from the first plane that describes the propellant grain are internally calculated (Figure 13). This first plane is positioned aft of the closure/case tangent point a distance of TAUMXF, which is the maximum distance burned in any given propellant grain.

It can be noted on Figure 13 that there is no attempt to achieve a perfect blend at the tangent point between identical materials located in the closure and in the cylindrical portion of the case. The liner inner surface does not necessarily match at the tangent point, etc. The error in propellant and insulation volumes should be small for any reasonable combination of input thicknesses, and the computation routine is greatly simplified. Because of some rigidly enforced geometric rules in the ballistic simulation routine, this mismatch of the liner inner surface makes a difference where the first coordinate (RECH(1), HECH(1)) is placed (Figure 14).

The innermost surface of the propellant (e.g., the bore of a CP grain) must intersect the liner inner surface on a Type 1 closure. There are no provisions for a forward opening larger than the propellant inner surface. Thus the control input to inhibit burning surface in the forward closure (FWDINH) has no meaning for a Type 1 closure, and so it is bypassed.

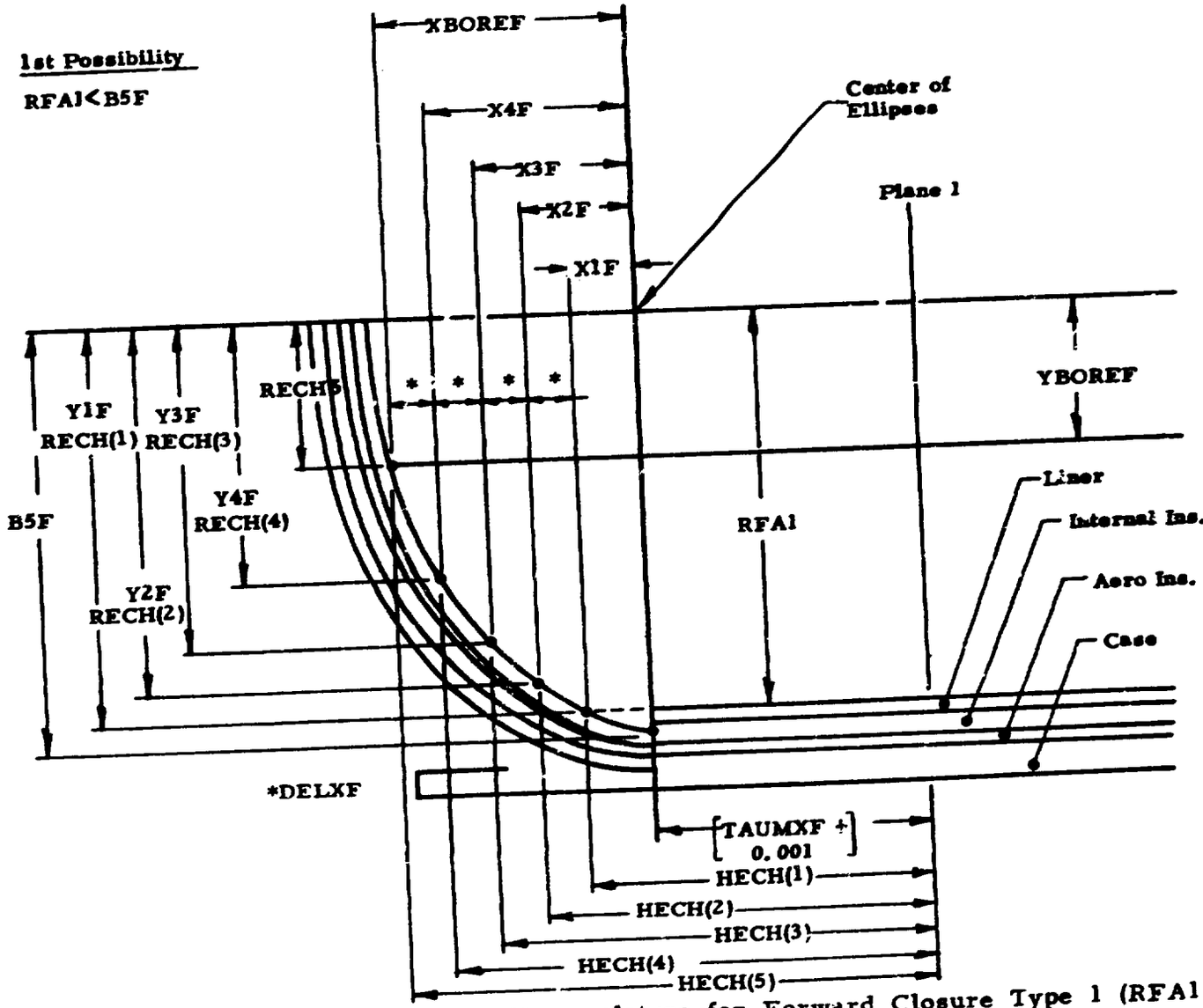


Figure 13. Internal and Output Nomenclature for Forward Closure Type 1 (RFA1 < B5F)

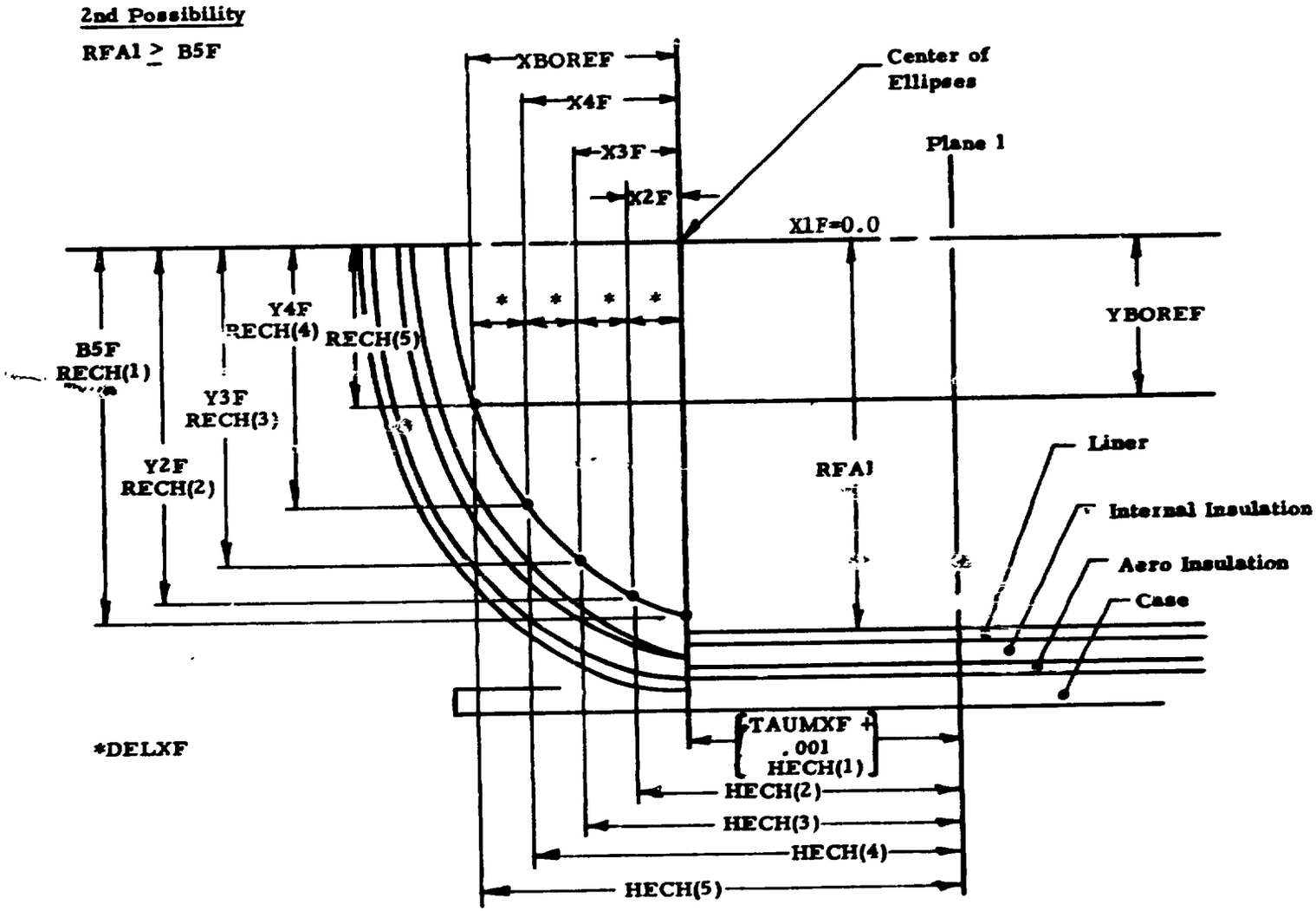


Figure 14. Internal and Output Nomenclature for Forward Closure Type 1 (RFA1 ≥ B5F)

Type 2 and Type 3 Forward Closures

Type 2 and Type 3 closures (Figure 15) are identical with regard to the propellant-closure arrangement; they are differentiated only in how the closure is secured in the case to form the pressure vessel. The controlling interface between the Type 2 and Type 3 closures and the cylindrical portion of the case is the outer edge of the flat plate, the inner surface of the case, and the aft-facing surface of the flat plate.

The forward face of the propellant grain is always perpendicular to the motor centerline; it can be inhibited (FWDINH = T) or not (FWDINH = F). A space can be provided between the closure insulation and propellant with the input LGAPF. The inputs to simulate a grain cast against and bonded to the closure are LGAPF = 0 and FWDINH = T.

Although not shown in Figure 15, the liner and insulation under the grain extends forward across LGAPF until it contacts the closure insulation.

The first plane describing the grain for the ballistic simulation is positioned a distance TAUMXF aft of the forward propellant surface. Only two radii and lengths are needed to describe the propellant forward of Plane 1. HECH(1) and HECH(2) are equal to TAUMXF. RECH(1) is equal to the propellant outside radius at Plane 1. If the forward propellant face is inhibited, RECH(2) is zero; if it is not inhibited, RECH(2) is equal to RECH(1).

AFT CLOSURE

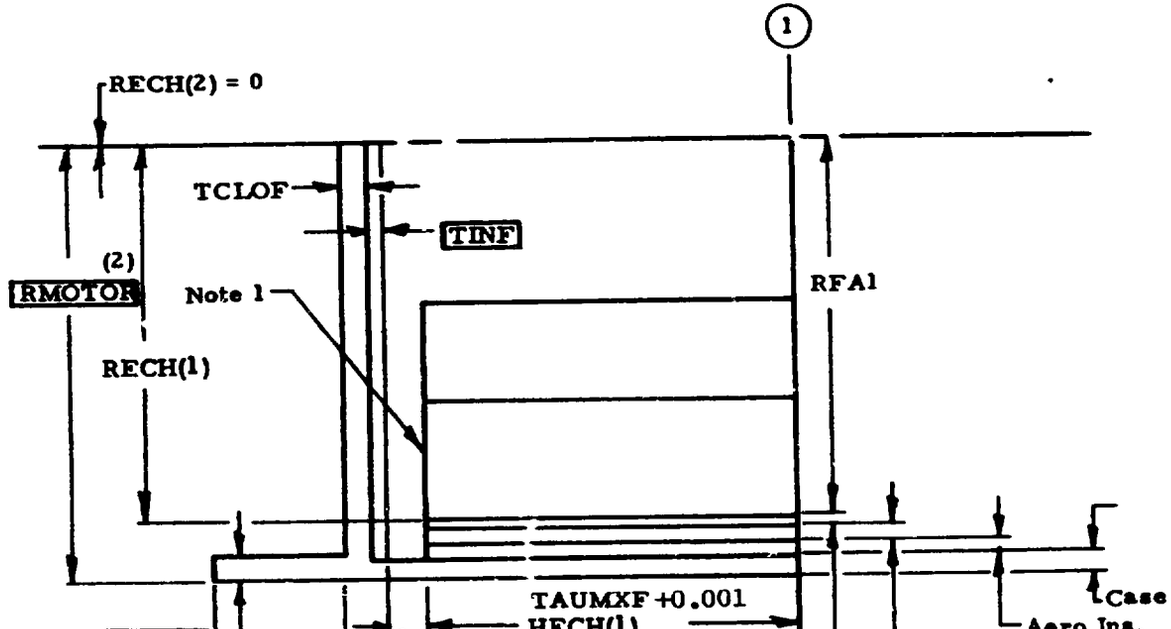
There are two basic aft closure arrangements. Type 1 is an ellipsoid with an opening for nozzle attachment; Type 2 causes the closure to be formed by the entrance section of the nozzle.

Type 1 Aft Closure

The controlling interface between the Type 1 aft closure and the cylindrical portion of the case is at the tangent point of the ellipsoid and the cylinder on the inner surface of the pressure vessel (Figure 16. Definition of ellipse parameters proceeds as with the ellipsoid forward closure with one significant exception. The thicknesses TINAMX, TSRBA, and TLNR are measured normal to the local surface at the nozzle-to-case interface, RNOZEN; these are then used to establish the ellipses.

The closure thickness, TCLOA, is calculated after the ballistic simulation is performed. The other ellipsoid surfaces are defined prior to the ballistic simulation. Radii (RECN) and concomitant lengths (HECN) are measured aft from the last plane that describes the propellant grain. This

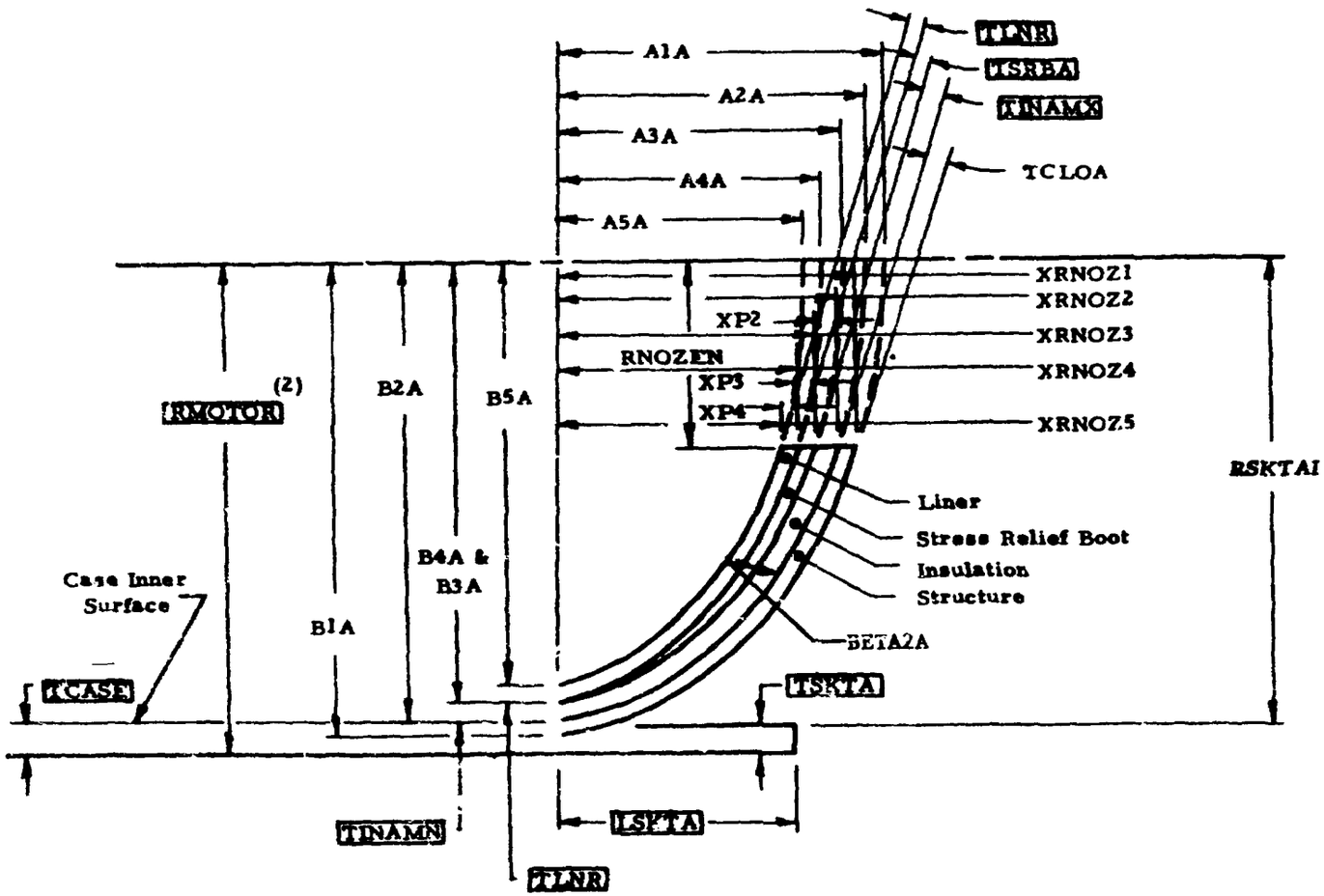
HEADEND TYPES 2 AND 3



(1) This face inhibited if FWDINH = T, not inhibited if FWDINH = F
(2) Input as appropriate diameter
(3) Dimensions shown in blocks are input; others are output

Figure 15. Forward Closure Type 2 and Type 3 (Flat Plate)

AFT CLOSURE TYPE 1



( ) Dimensions shown in blocks are input; others are output
(2) Input as appropriate diameter

Figure 16. Aft Closure Type 1 (Ellipsoid)

last plane is positioned forward of the aftmost propellant surface a distance of TAUMXA (Figure 17).

Note on Figure 17 that the axial length of the aft closure, LCLOA, extends aft from the case-to-closure tangent point to the interface of the stress relief boot and the closure insulation. This point also serves as the interface between the case and nozzle (at the radius RNOZEN).

As for the Type 1 forward closure, the mismatch between liner inner surfaces of the case and closure is considered through the internal generation of appropriate radii and lengths (Figures 17 and 18).

Aft-facing burning surface perpendicular to the motor centerline is formed when the aft case opening, RNOZEN, is larger than the innermost surface of the propellant (e.g., the bore of a CP grain). This perpendicular surface can be inhibited (AFTINH = T) or not (AFTINH = F).

Type 2 Aft Closure

An aft closure Type 2 is, in actuality, not a closure per se (Figure 19); instead the pressure vessel is closed by the entrance section of the nozzle. There is no specific controlling interface between the Type 2 aft closure and the case. The controlling interface with the nozzle is at the outside propellant radius at Plane 14 (RFA14). In other words, the case "opening" for the nozzle, RNOZEN, is set equal to RFA14. Length of the aft closure is zero, which means that the nozzle is positioned at the propellant aft face.

The aft face of the grain is always perpendicular to the motor centerline; it can be inhibited (AFTINH = T) or not (AFTINH = F). The last plane describing the propellant grain for the ballistic simulation is positioned a distance TAUMXA forward of the aft propellant surface. Only two radii and lengths are needed to describe the propellant aft of Plane 14. HECN(1) and HECN(2) are equal to TAUMXA. RECN(1) is equal to the propellant outside radius at Plane 14. If the aft propellant face is inhibited, RECN(2) is zero; if there is no inhibiting, RECN(2) is equal to RECN(1).

Thickness of the aft skirt is TCASE for a Type 2 aft closure (for purposes of skirt weight calculations). It was reasoned that for most applications with this aft end arrangement the skirt would be an extension of the case.

NOZZLE

Six nozzle configurations were identified for user selection (Figures 20 through 25). A review of recent designs (References 6 and 7) and

1st Possibility

RFA14 < B5A

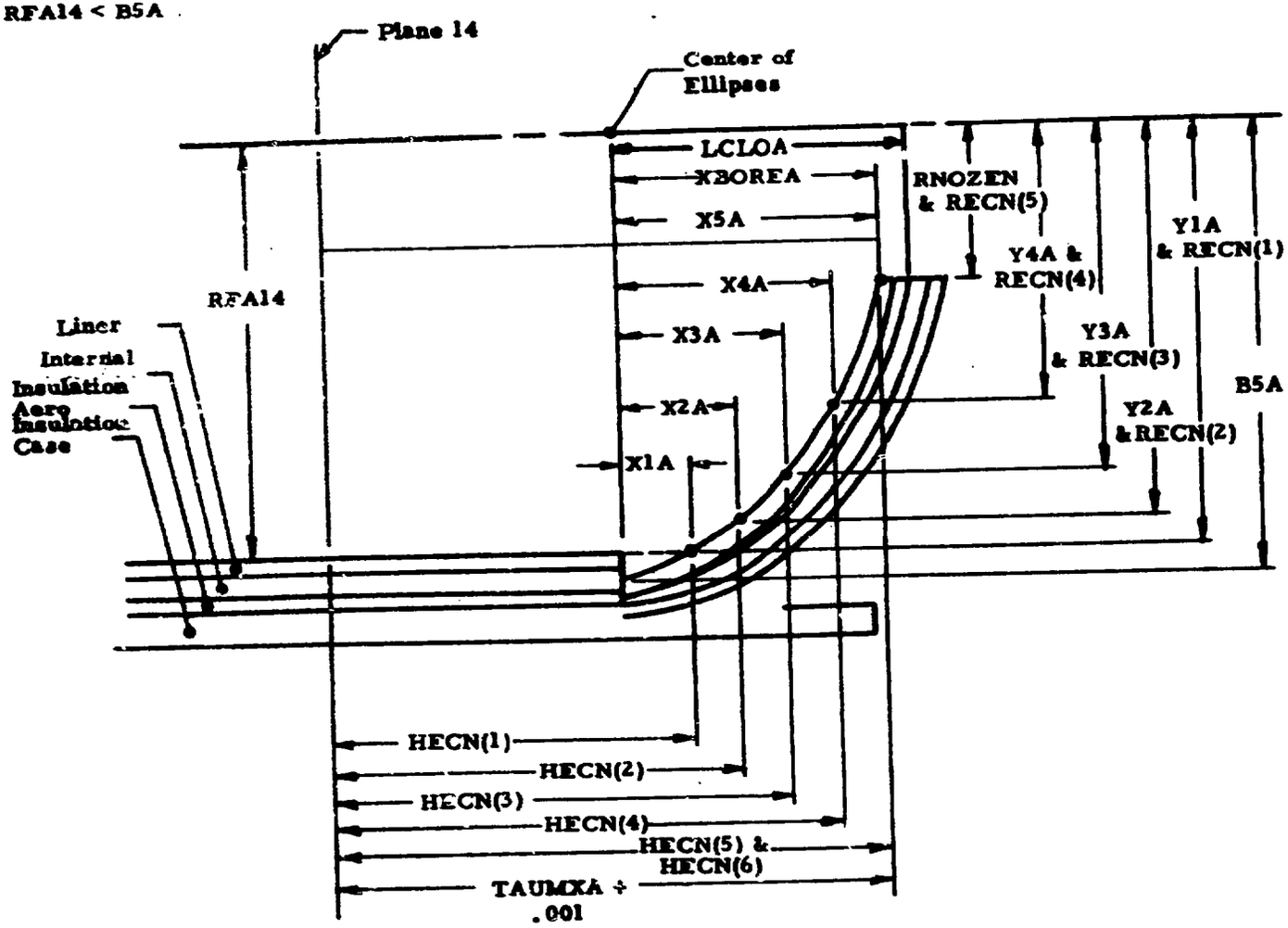


Figure 17. Internal and Output Nomenclature for Aft Closure Type 1 (RFA14 < B5A)

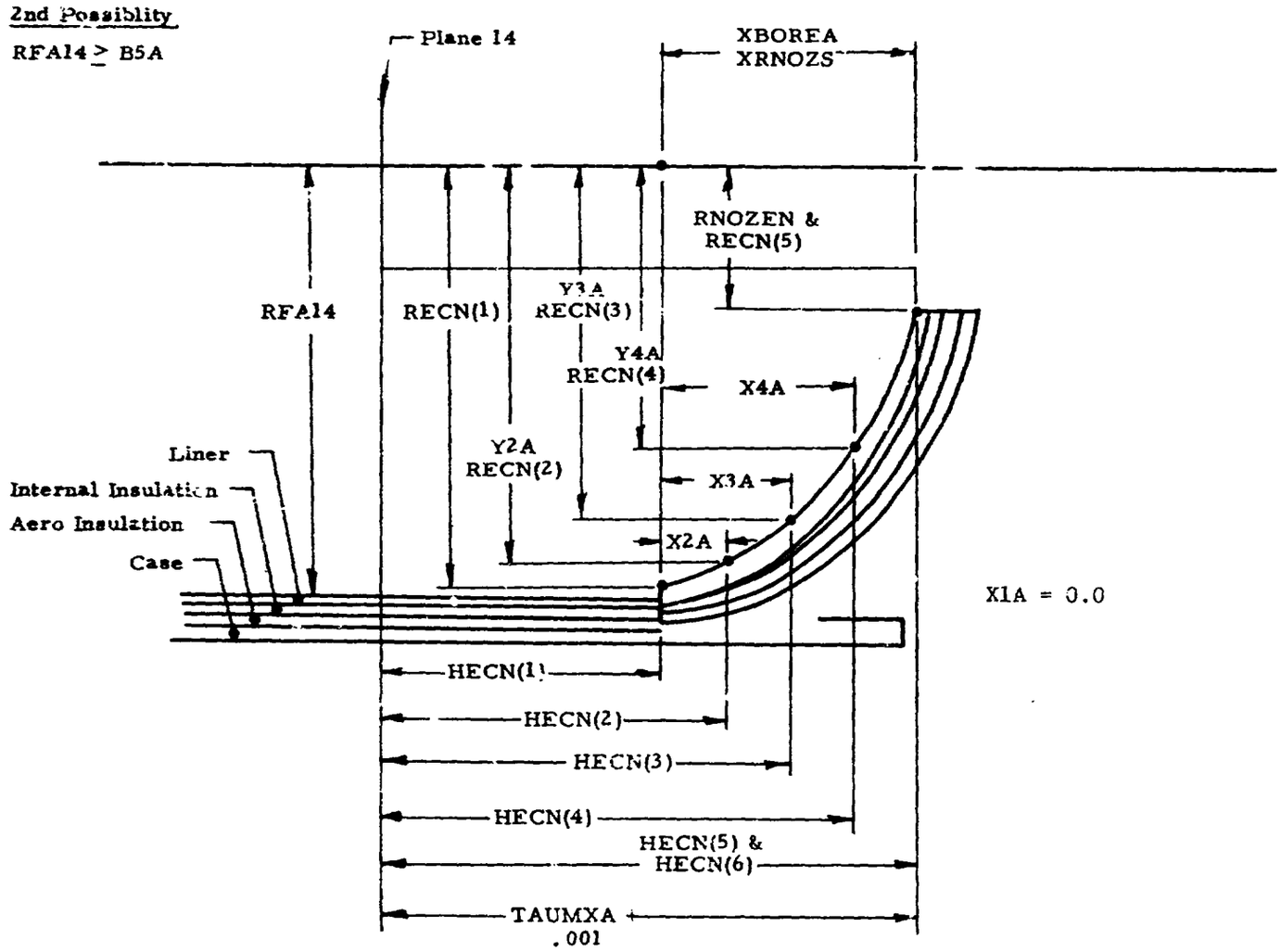
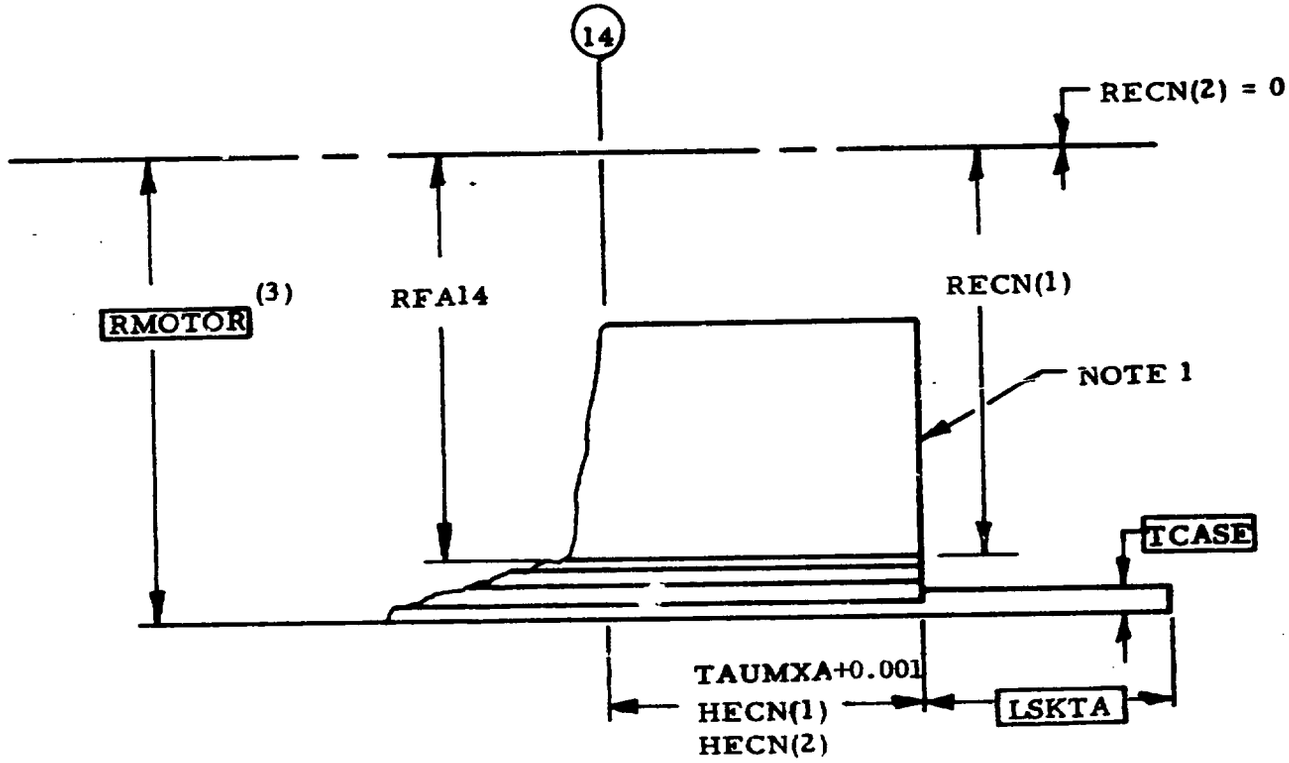


Figure 18. Internal and Output Nomenclature for Aft Closure Type 1 (RFA14 ≥ B5A)

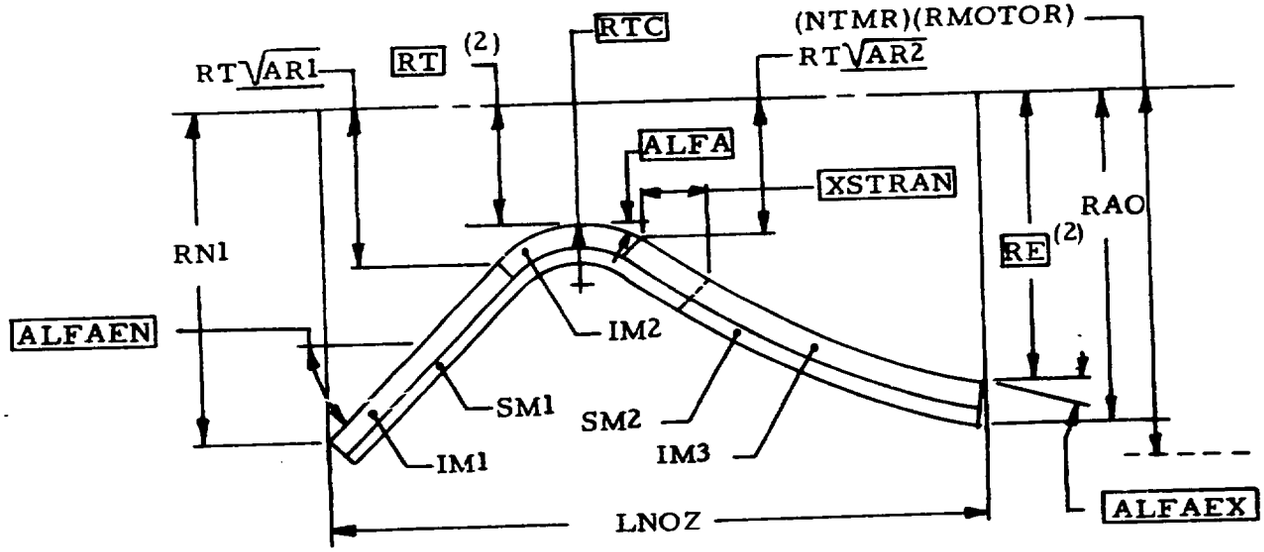
AFT CLOSURE TYPE 2



(1) This face inhibited if AFTINH = T, not inhibited if AFTINH = F
(2) Dimensions shown in blocks are input; others are output
(3) Input as appropriate diameter

Figure 19. Nomenclature for Aft Closure Type 2

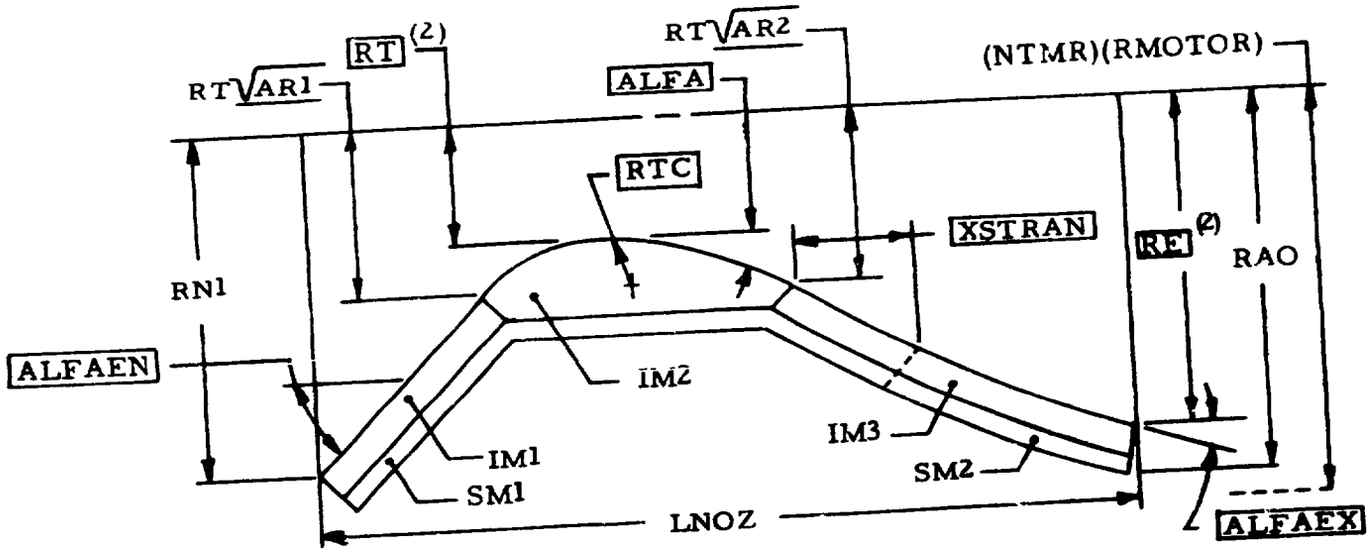
NOZZLE TYPE 1



(1) Dimensions shown in blocks or underlined are input; others are output
(2) Input as appropriate diameters
(3) IM1 = Insulating Material No. 1
IM2 = Insulating Material No. 2
IM3 = Insulating Material No. 3
SM1 = Structural Material No. 1
SM2 = Structural Material No. 2

Figure 20. Dimensional Inputs for Nozzle Type 1

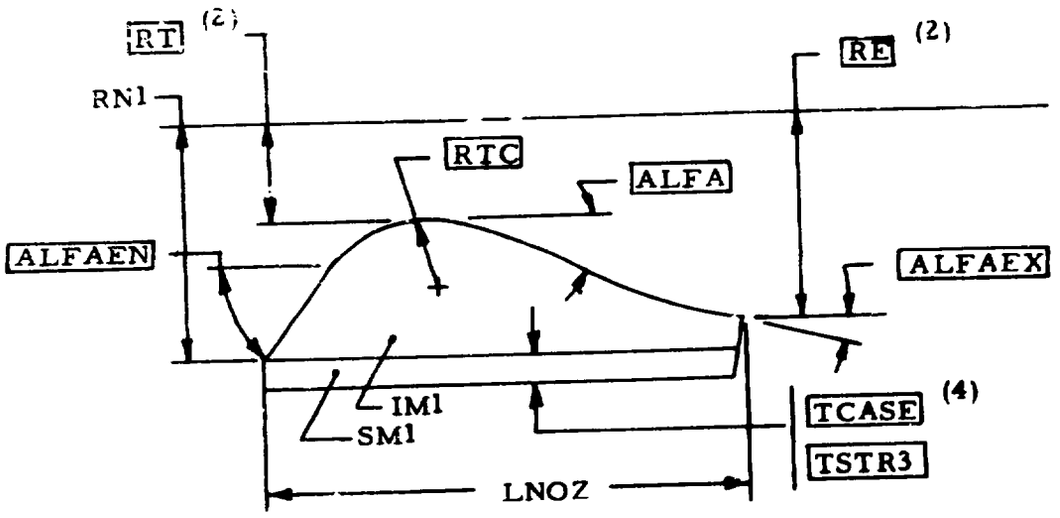
NOZZLE TYPE 2



(1) Dimensions shown in blocks or underlined are input; others are output
(2) Input as appropriate diameters
(3) IM1 = Insulating Material No. 1
IM2 = Insulating Material No. 2
IM3 = Insulating Material No. 3
SM1 = Structural Material No. 1
SM2 = Structural Material No. 2

Figure 21. Dimensional Inputs for Nozzle Type

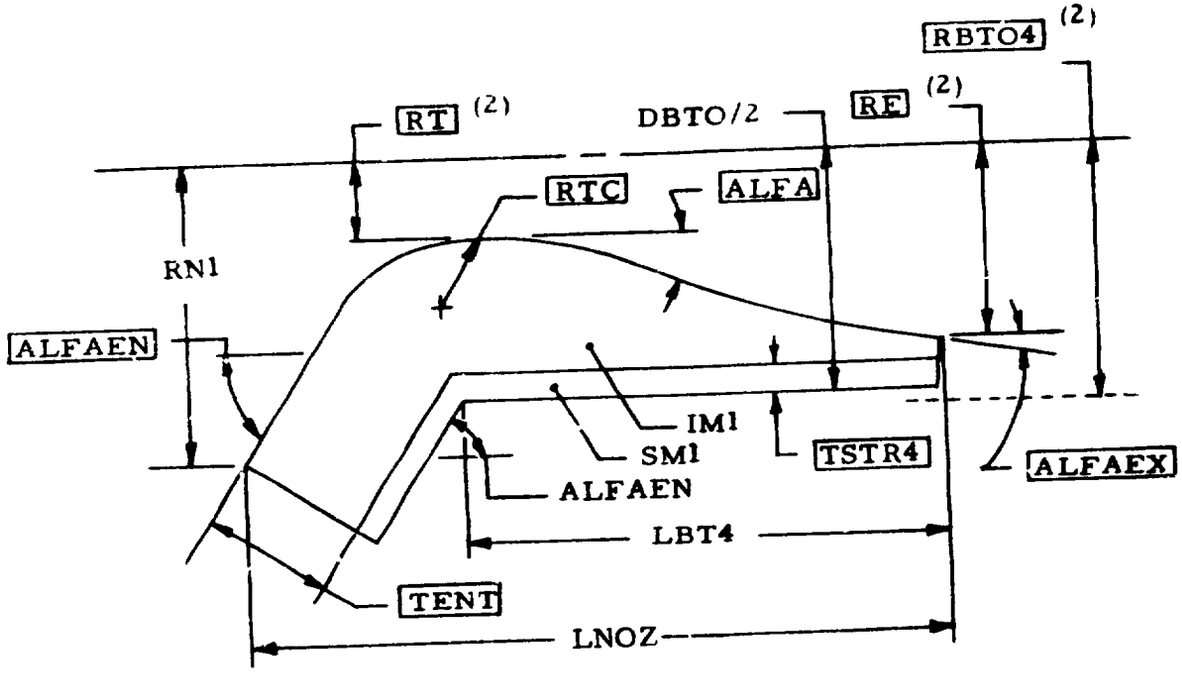
NOZZLE TYPE 3



(1) Dimensions shown in blocks are input; others are output
(2) Input as appropriate diameters
(3) IM1 = Insulating Material No. 1
SM1 = Structural Material No. 1
(4) TSTR3 is user-input for Aft Closure Type 1 (IAFTCL=1); if IAFTCL=2, TSTR3 is internally set to TCASE.

Figure 22. Dimensional Inputs for Nozzle Type 3

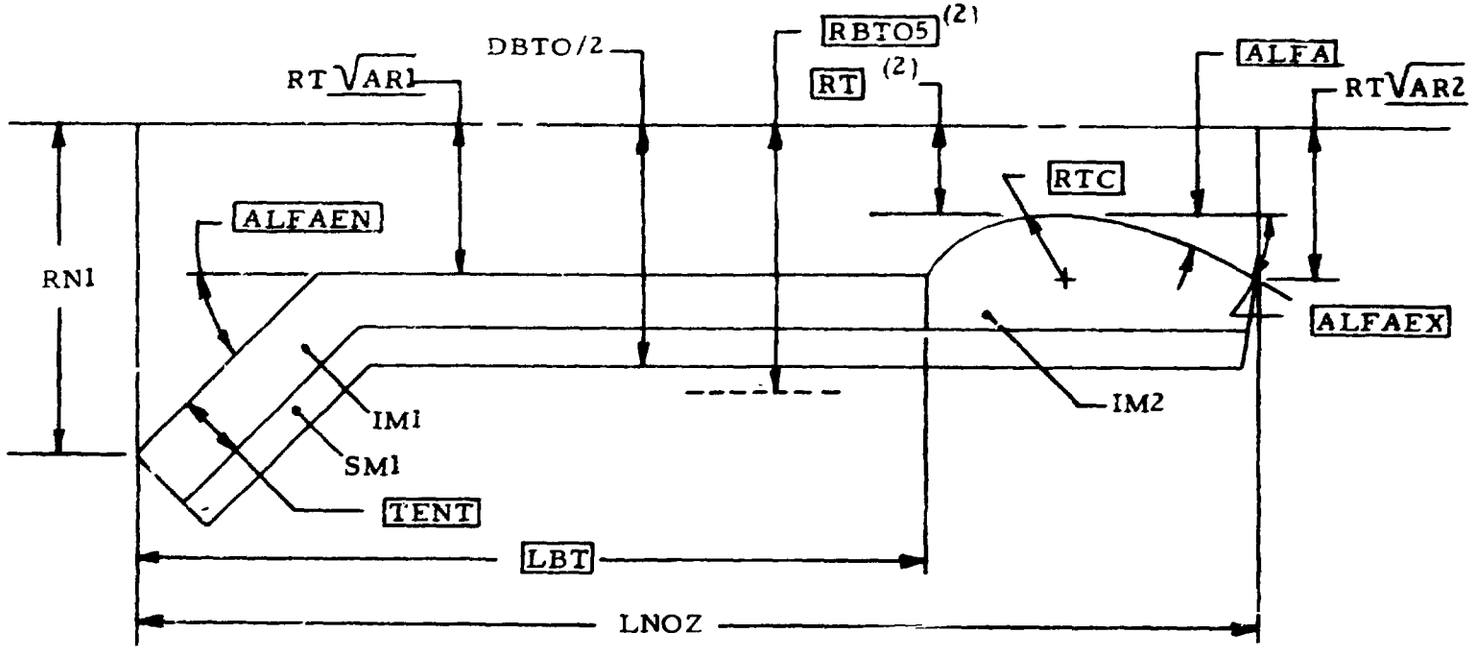
NOZZLE TYPE 4



(1) Dimensions shown in blocks are input; others are output
(2) Input as appropriate diameters
(3) IM1 = Insulating Material No. 1
SM1 = Structural Material No. 1
(4) RBTO4 = DBTO4/2

Figure 23. Dimensional Inputs for Nozzle Type 4

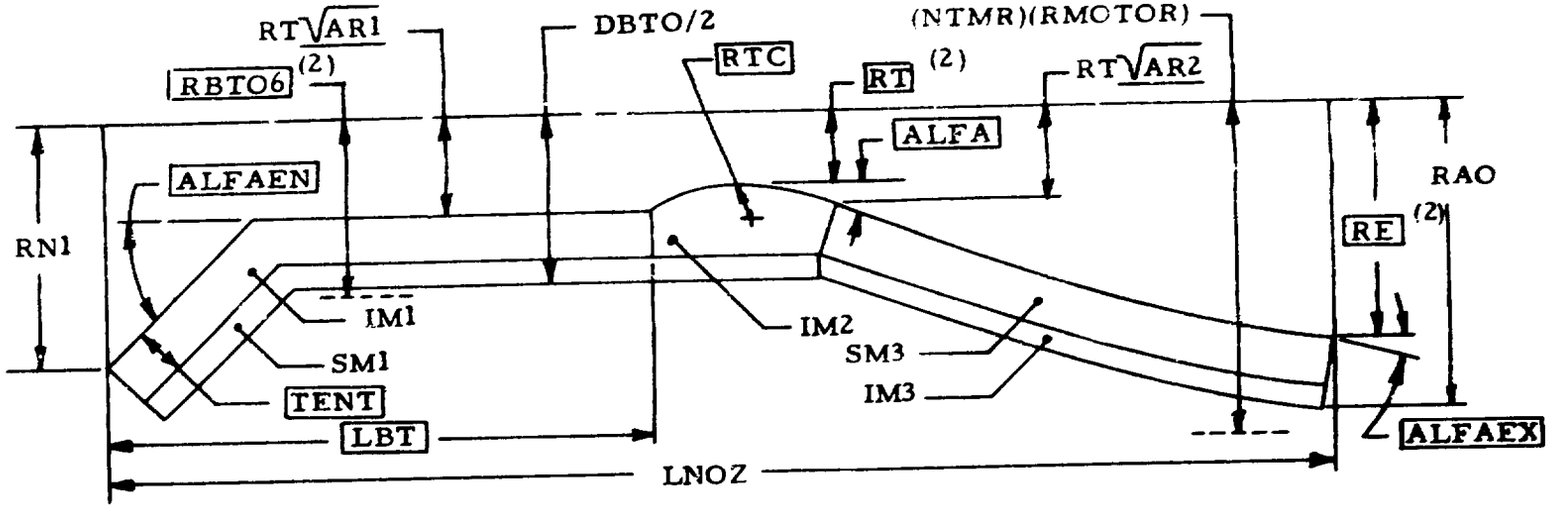
NOZZLE TYPE 5



(1) Dimensions shown in blocks or underlined are input; others are output.
(2) Input as appropriate diameters
(3) IM1 = Insulating Material No. 1
IM2 = Insulating Material No. 2
SM1 = Structural Material No. 1

Figure 24. Dimensional Inputs for Nozzle Type 5

NOZZLE TYPE 6



(1) Dimensions shown in blocks or underlined are input; all others are output
(2) Input as appropriate diameters
(3) IM1 = Insulating Material No. 1
IM2 = Insulating Material No. 2
IM3 = Insulating Material No. 3
SM1 = Structural Material No. 1
SM2 = Structural Material No. 2

Figure 25. Dimensional Inputs for Nozzle Type 6

the CPIA Manual (Reference 8) indicates these configurations should describe most applications for which SPOC is intended.

Nozzle Type 1 and Nozzle Type 2 (Figures 20 and 21) are basically the same, except Type 2 provides for a one-piece throat insert in a conical seat. Type 1 is more akin to all-plastic nozzles where low weight is important, although the computation routine allows any material to be used for structural support. Nozzle Type 3 (Figure 22) is derived from smaller tactical motors where the support structure is an extension of the case. However, it is allowable to attach the Type 3 nozzle to Type 1 aft closure. Nozzle Type 4 (Figure 23) was established to provide missile equipment volume around the outside of the nozzle and to have its throat located at the forward end of the nozzle section. An identical external envelope is found with Nozzle Type 5, but the nozzle throat is located at the aft end of the blast tube. Nozzle Type 6 still provides equipment volume at the aft end of the motor, but now the envelope allows an exit cone to be attached to the arrangement of the Type 5 nozzle.

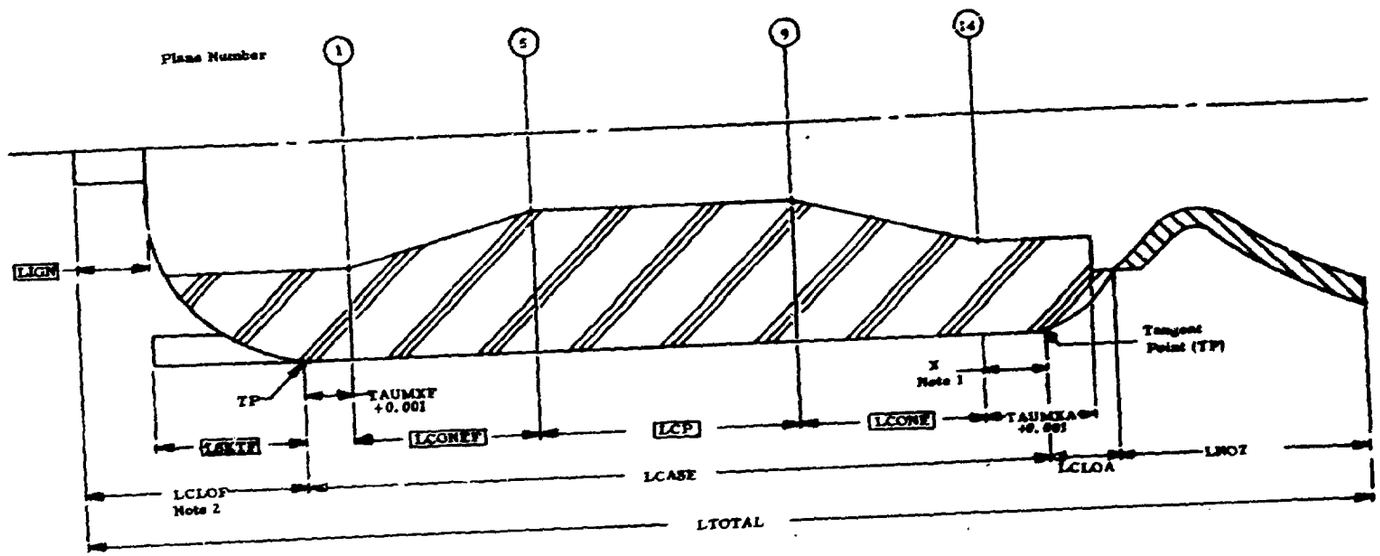
All nozzle types have the option of either conical or contoured expansion sections. In the former, inputs of ALFA = ALFAEX describes the conical section to the nozzle routine, and a flag is set by the user (CONTUR = F) as part of the motor definition inputs. The user specifies a contoured expansion section by CONTUR = T and ALFAEX < ALFA. When a contoured section is specified, the code internally chooses either an elliptical, hyperbolic or parabolic profile on the basis of minimum length. Neither of these contours will be identical to one determined by precise gas dynamic analyses, but the estimates of weights and lengths are sufficiently accurate for the purposes of SPOC.

As described earlier, the nozzle entrance radius, RN1, is eventually made equal to the aft case opening, RNOZEN; at the beginning of an optimization problem there may be points where RN1 and RNOZEN are not equal, but an appropriate penalty is calculated to force them together.

LENGTHS

Motor length is summed as shown in Figures 26 and 27. These two illustrations show a Type 5 (CP) grain, but the technique is consistent for all grains. Individual lengths between Plane 1 and Plane 14 are unique for a particular grain configuration. Length components forward of Plane 1 and aft of Plane 14 are identical for all grain configurations, as is the definition of LCASE.

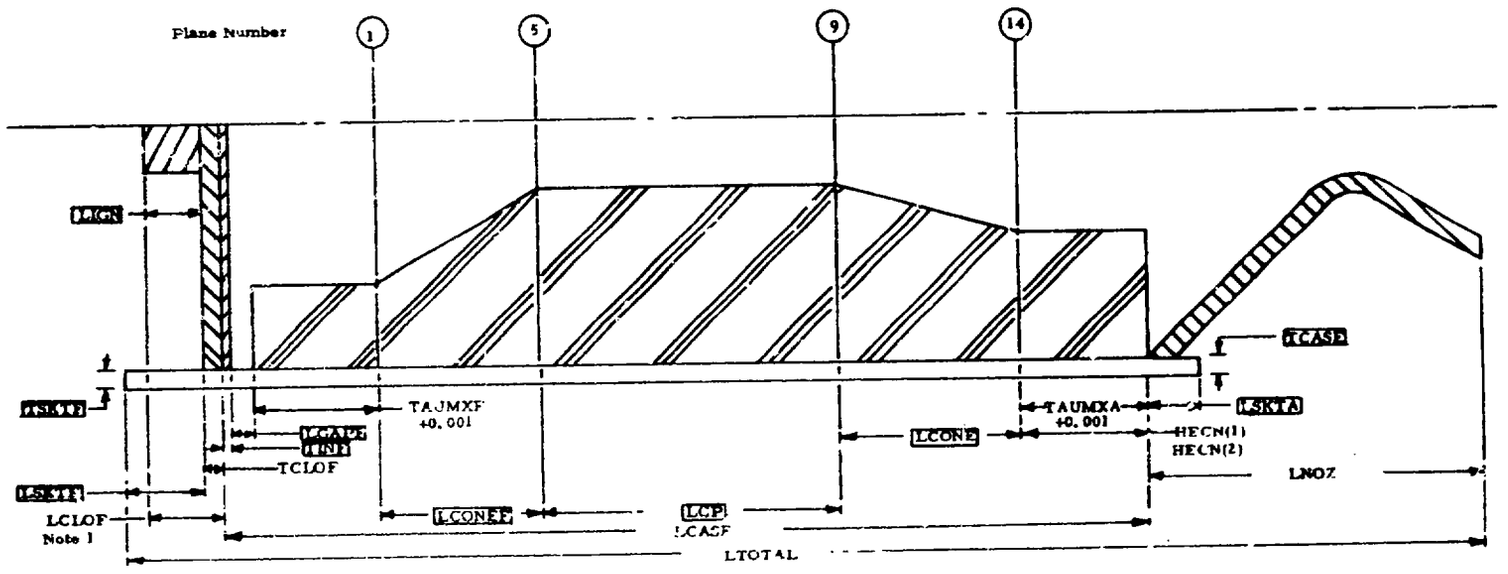
Figure 28 illustrates the details of the case-nozzle interface for a Type 1 aft closure and how resultant lengths are defined. The nozzle entrance is always placed at the junction of the stress relief boot and closure internal insulation. Thus LCLOA is measured from that point forward to the case-closure tangent point, and LNOZ is measured from that point aft to the nozzle exit plane.



NOTES

(1) For Aft Closure Type 1, this dimension is X = (HECN(1)-X1A) if RFA14 < B5A, or X = HECN(1) if RFA14≥B5A
(2) LSFTF used to calculate LTOTAL if LSKTF > LCLOF
(3) Dimensions shown in blocks are input; others are output

Figure 26. Typical Length Summation for Forward Closure Type 1 and Aft Closure Type 1



NOTE

(1) LCLOF used to calculate LTOTAL if LCLOF > LSKTF
(2) Dimensions shown in blocks are input; others are output

Figure 27. Typical Length Summation for Forward Closure Type 2 or 3 and Aft Closure Type 2

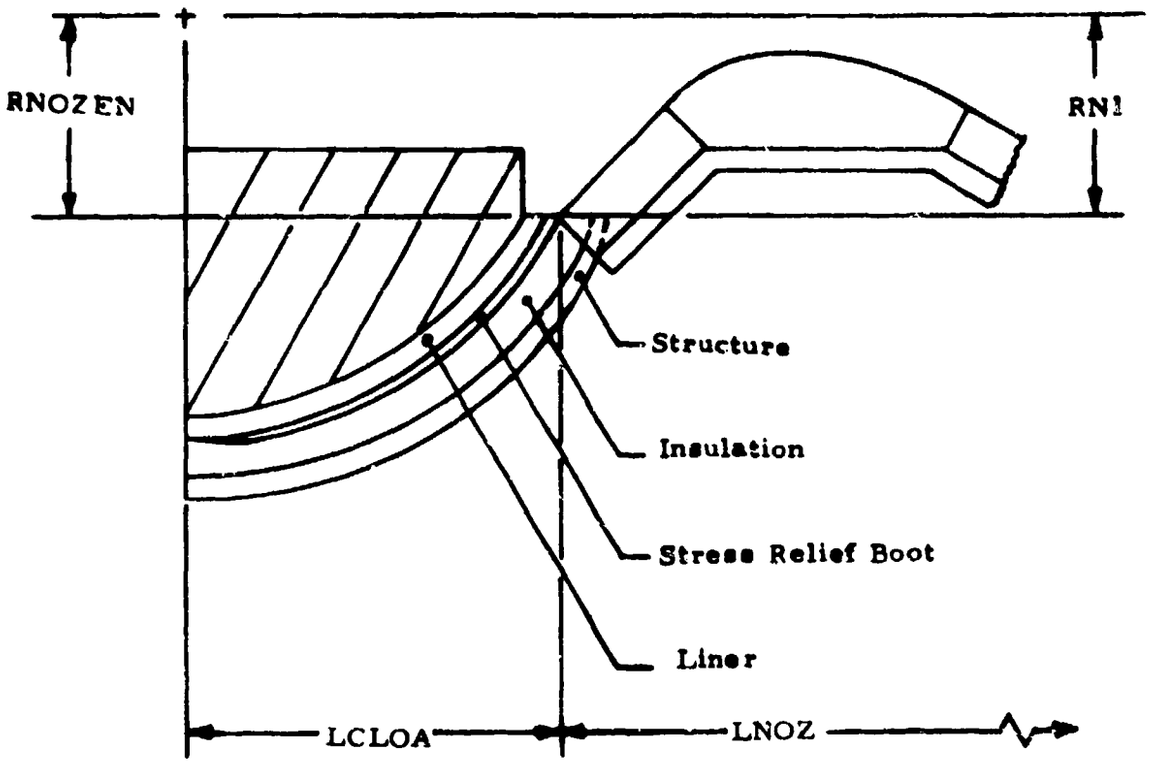


Figure 28. Case-Nozzle Interface for Type 1 Aft Closure

ANALYSIS ROUTINES

This section of Volume I describes the analyses performed within SPOC to determine how the design currently being proposed by PATSH satisfies performance requirements, design constraints and operating limits. Propellant grain configuration dimensions are checked for geometric validity and are converted into the language of the ballistic simulation subprogram. Propellant ingredient weight fractions are checked for compatibility with interacting limits, normalized to 100% total weight fraction and then sent to the thermochemical subprogram. There ballistic characteristics and parameters for impulse efficiency and combustion stability predictions are calculated. Impulse efficiency is predicted with the AFRPL Solid Propulsion Prediction (SPP) code (Reference 12) and combustion stability with the AFRPL Standard Stability Prediction (SSP) code (Reference 19). A ballistic simulation is performed, including the effects of erosive burning and mass addition and using a geometrically rigorous two-dimensional grain regression technique. Once the ballistic simulation is completed, results from it are used for pressure vessel and propellant structural analyses, a trajectory simulation, cost estimate, and weight calculations, and combustion stability prediction.

Some analyses are not performed unless specified by the user: thermochemical, impulse efficiency, propellant structural, trajectory, cost, and combustion stability. Even when the thermochemical analyses is called, it is bypassed unless the pressure or the nozzle expansion has changed at least 5% between the two previous analyses, or if propellant formulation has changed.

Equations describing the various analyses are numbered within each report section. All references are listed at the end of this volume.

BALLISTIC SIMULATION

The ballistic simulation routine requires inputs of the initial propellant configuration, propellant ballistic and burning rate properties, nozzle geometry and various control parameters. The primary computed outputs are chamber pressure, thrust, mass flow rate, pressure-time integral, impulse, and propellant weight at selected values of burning time.

The initial propellant geometry is described at a series of planes (called "direct input" planes) positioned along the grain in accordance with a pre-programmed scheme. Fourteen planes are used for all grain configurations; they are located as shown in the illustrations describing each configuration. Planes not shown on the illustrations are called "interpolated input" planes because they are positioned equidistance between and their dimensions are internally derived from the two adjacent "direct input" planes.

The gas dynamic solution is based on the assumption of equilibrium (i.e., rate of mass stored is negligible) at each time point. Other assumptions are:

a. Combustion products behave as perfect gases.

b. Flow processes are isentropic.

c. Propellant burning surface regresses only in a radial direction, except on the forward-facing and aft-facing surfaces at the forward and aft grain terminations, respectively.

d. Burning rate is calculated at each plane and varies linearly between planes.

e. Ends of propellant grain do not experience erosive burning.

f. All interpolations are linear.

g. Chamber is already filled and all of the initial burning surface is ignited at the initial (time = zero) calculation.

h. Contribution of igniter mass flow to initial pressure is ignored.

i. Two-phase mixture effects and other losses are accounted for through an empirical impulse efficiency factor.

Operation is initiated with a solution for incremental burning surface areas and propellant volumes throughout the motor. Then a trial value of stagnation chamber pressure at the head-end of the motor is assumed. Based on the assumed pressure, incremental values of burning rate, gas flow rate, temperature, velocity, specific weight, and Mach number are

calculated at the adjacent station in the downstream direction. These calculations include the effects of both mass addition pressure drop and erosive burning effects on propellant burning rate. This solution is repeated for each successive station along the grain until a solution is obtained for gas properties at the aft end of the grain. Gas discharge rate through the nozzle is computed from the nozzle end stagnation pressure, using the appropriate solution for either sonic or subsonic flow. The gas discharge rate is then compared to the gas generation rate. If these two values do not agree within a prescribed tolerance, the original trial value of head-end pressure is adjusted to a new value and the entire solution is repeated until the required agreement is obtained. Motor thrust is then computed, and all inputs are printed. Burning time is incrementally advanced and the thickness burned at each plane is calculated, based on the respective burning rate values previously calculated. New incremental values of burning surface and volume are computed, and the entire ballistic analysis is repeated. This process is continued until all of the propellant has been consumed, or until a specified time or pressure level is obtained.

Ballistic simulations can be performed at either one or two grain soak temperatures. If simulations are run at two temperatures, results at the high temperature are used to calculate

a. Upper three-sigma ignition thrust,
b. Lower three-sigma burn time,
c. Maximum head-end pressure,
d. Pressure vessel design pressures (MEOP, yield, ultimate),

and results at the low temperature are used to calculate

a. Lower three-sigma total impulse
b. Lower three-sigma ignition thrust
c. Upper three-sigma burn time
d. Maximum port Mach number
e. Upper three-sigma ignition thrust for propellant structural analysis
f. Burn time for nozzle thermal analysis

If a simulation is run at only one temperature, all of the above are calculated at that single temperature.

Each grain configuration has its own individual subroutine (SETUP1, SETUP2, etc.) that generates the initial propellant geometry description from the fewest possible inputs. The different configurations are described in detail in the following sections. All grain subroutines use another subroutine (CLOS) that performs the same function for the different pressure

vessel closure configurations that may be selected. In addition, CLOS calculates the weights of inert components associated with the closures, as does each of the "SETUP" routines for their respective grain configurations. Propellant surface regression and internal ballistics are calculated at four stations in the closures, as shown on Figures 29 and 30.

Burning rate is calculated with the Vielle relationship

$$RATE = (BRSF)\,(A)\,P^{XN} \tag{1}$$

The user supplies RB70 (the rate at 1000 psia, 70°F) and XN. The code uses conventional temperature sensitivity coefficients to adjust the coefficient "A" to temperature extremes.

Erosive burning can be considered through selection of one of the combinations shown on Table 2 (References 9 and 10).

When either the second or third options of Table 2 are selected, the burn rate becomes

$$RATE = (BRSF)(A)P^{XN}\,(M/MCRIT)^{XM} \tag{2}$$

or,

$$RATE = (MPCOEF)(MP)^{MPEXP} \tag{3}$$

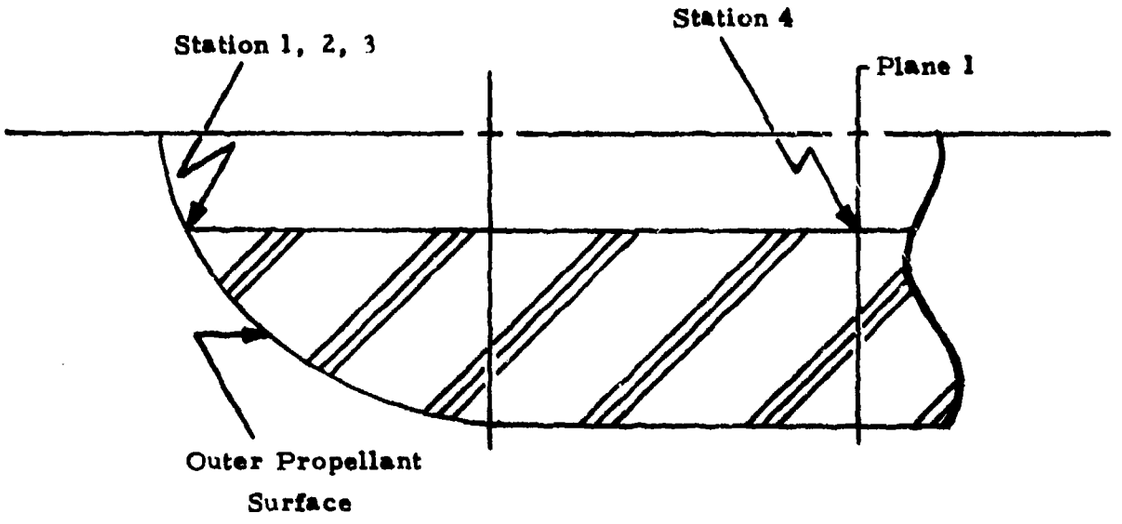
where MP is the product of the local Mach number and static pressure. The relation producing the largest burn rate at a given location and time is used to calculate the internal gas dynamics (either Eq (1), (2), or (3)).

Nozzle throat ablation is modeled by

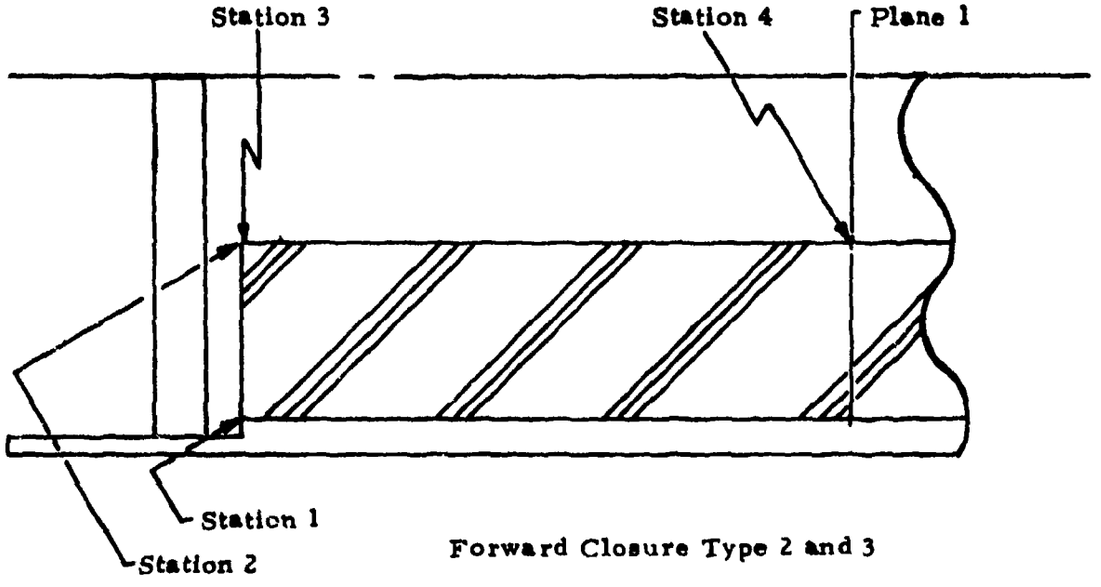
$$RE = (KRE1)(PON)^{KRE2} \tag{4}$$

where PON is the nozzle-end stagnation pressure. The nozzle thermal analysis subroutine also calculates an ablation profile for the entire nozzle internal surface, but that is not used in the ballistic simulation.

Impulse efficiency is a user input, or it may be estimated by the subroutine IMPEFF which uses the AFRPL SPP "empirical" model (Reference 12). Certain inputs to the SPP impulse efficiency calculation must come from a thermochemistry analysis of a particular propellant formulation; if a formulation is not input to the code, the user must supply values for the specified parameters.

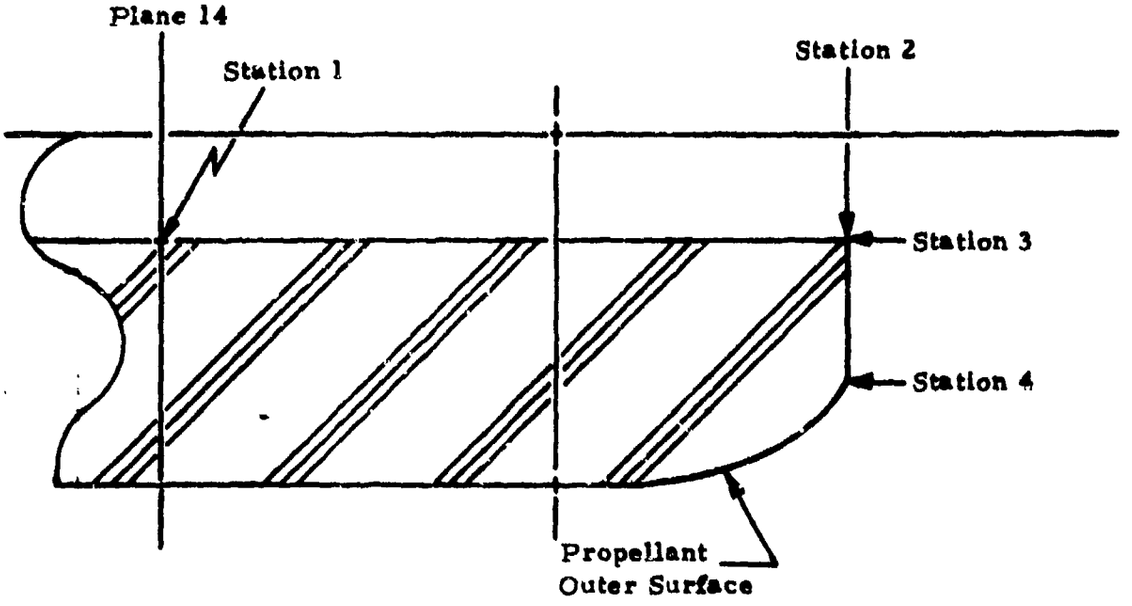


Forward Closure Type 1

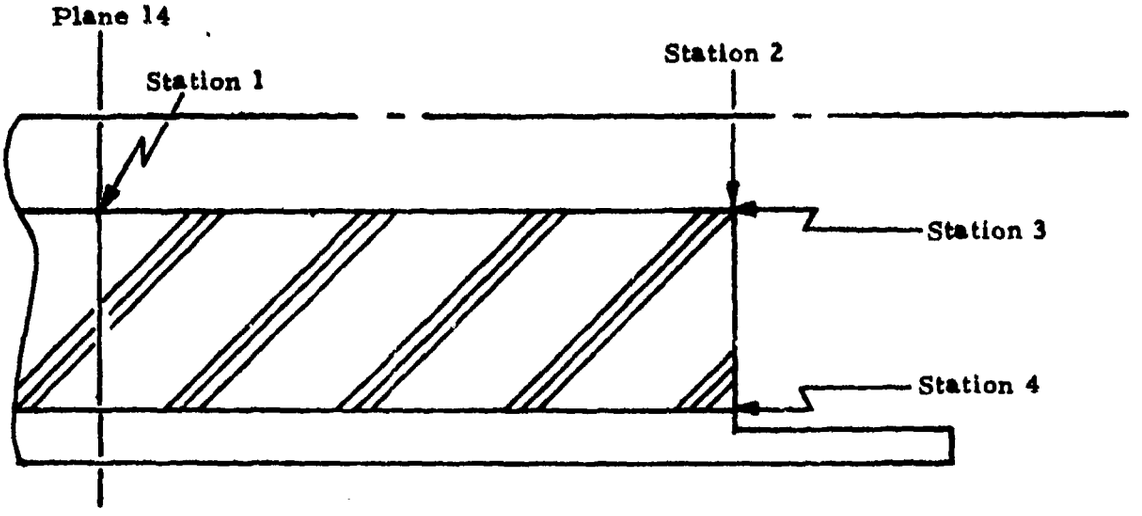


Forward Closure Type 2 and 3

Figure 29. Location of Stations for Ballistic Simulation in Head-end of Grain



Aft Closure Type 1



Aft Closure Type 2

Figure 30. Location of Stations for Ballistic Simulation in Nozzle-end Of Grain

TABLE 2

EROSIVE BURNING RATE COMBINATIONS

AUTOEB	MCRIT	XM	MPCOEF	MPEXP	Description
0.0	0.0	0.0	0.0	0.0	No erosive burning
0.0	X[(a)]	X	X	X	Erosive burning with user values for MCRIT, XM, MPCOEF and MPEXP
1.0	0.0	0.0	X	X	Erosive burning with internally calculated values for MCRIT and XM (see Note (b)) and user values for MPCOEF and MPEXP (see Note (c))

(a) "X" indicates user-input or default value.

(b) Saderholm model, Reference 9 and 10. MCRIT = Mach number corresponding to 250 fps. XM calculated from the relationship.

$$XM = \ln\left[0.06768\left(\frac{P^{0.74}}{RATE}\right)^{0.4948}\right]$$

which is a curve fit to Saderholm data (XM $\geq$ 0.0 and RATE = burn rate without erosive burning, P = local static pressure, when local Mach number $>$ MCRIT.

(c) Default values MPCOEF = 0.0093 and MPEXP = 0.71.

Nozzle divergence loss is calculated according to

$$LAMDA = 1 - COS\left[\frac{ALFA + ALFAEX}{2}\right] \quad (5)$$

where

ALFA = Half-angle (deg) at entrance to nozzle expansion section

ALFAEX = Half-angle (deg) at exit of nozzle expansion section

This term LAMDA is included in the calculation of thrust coefficient within the ballistic simulation module, unless the SPP impulse efficiency option is selected. Because SPP efficiency includes divergence losses, LAMDA per Eq (5) is internally set equal to 1.0.

Certain other gas dynamic parameters may be input by the user or calculated internally if a propellant formulation is supplied.

The ballistic simulation uses inputs from the following namelists:

a. BALLST
b. GRAIN x (x=1, . . . 5)
c. NOZGEO
d. INGAMT (if thermochemical analysis performed)
e. Card that names ingredients

PROPELLANT GRAIN CONFIGURATIONS

Basic describing dimensions for the propellant grains are furnished by the user, who can then specify that certain of them be adjusted by PATSH during the optimization process. Regression of the burning surface is mathematically exact, and that means that the dimensions furnished to the ballistic simulation module must obey strict rules. If the rules are not obeyed, the burning surface regression cannot proceed and the run is aborted. Abnormal terminations are unwanted in a pattern search optimization process.

Therefore, a subroutine was formulated for each of the different grain configurations to:

(a) check the geometric validity of the incoming dimension set,
(b) adjust certain dimensions to obtain geometric validity (if required), and calculate associated penalties
(c) derive from the incoming dimensions those other dimensions required by the ballistic simulation module
(d) compare the dimensions with design constraints
(e) calculate volume and weights of inert components associated with the cylindrical portion of the pressure vessel
(f) calculate grain dimensions that are needed in subsequent analyses

The grain geometries are described in the ballistic simulation routine by a series of planes oriented perpendicular to the motor centerline and positioned along the length of the grain at appropriate locations. Fourteen planes are used to describe all grains. Some are located where there are changes in either the internal or external configuration of the propellant; these are known as "direct input" planes, and it is their dimension sets that are treated in the aforementioned geometric validation analysis. The remaining planes are located equidistance between adjacent "direct input" planes; they are known as "interpolated inputs" because their dimensions are derived from linear interpolations from the adjacent direct input planes. The planes that are indicated on the illustrations accompanying this discussion are the direct input planes established for each of the grain configurations.

Each grain type has its own particular arrangement of internal insulation. Details of the insulation are given in Volume II of this Manual and will not be repeated here.

Dimensions that can be adjusted during the optimization process are discussed in detail in subsequent paragraphs. The capabilities for adjustment are not the same for all the grains.

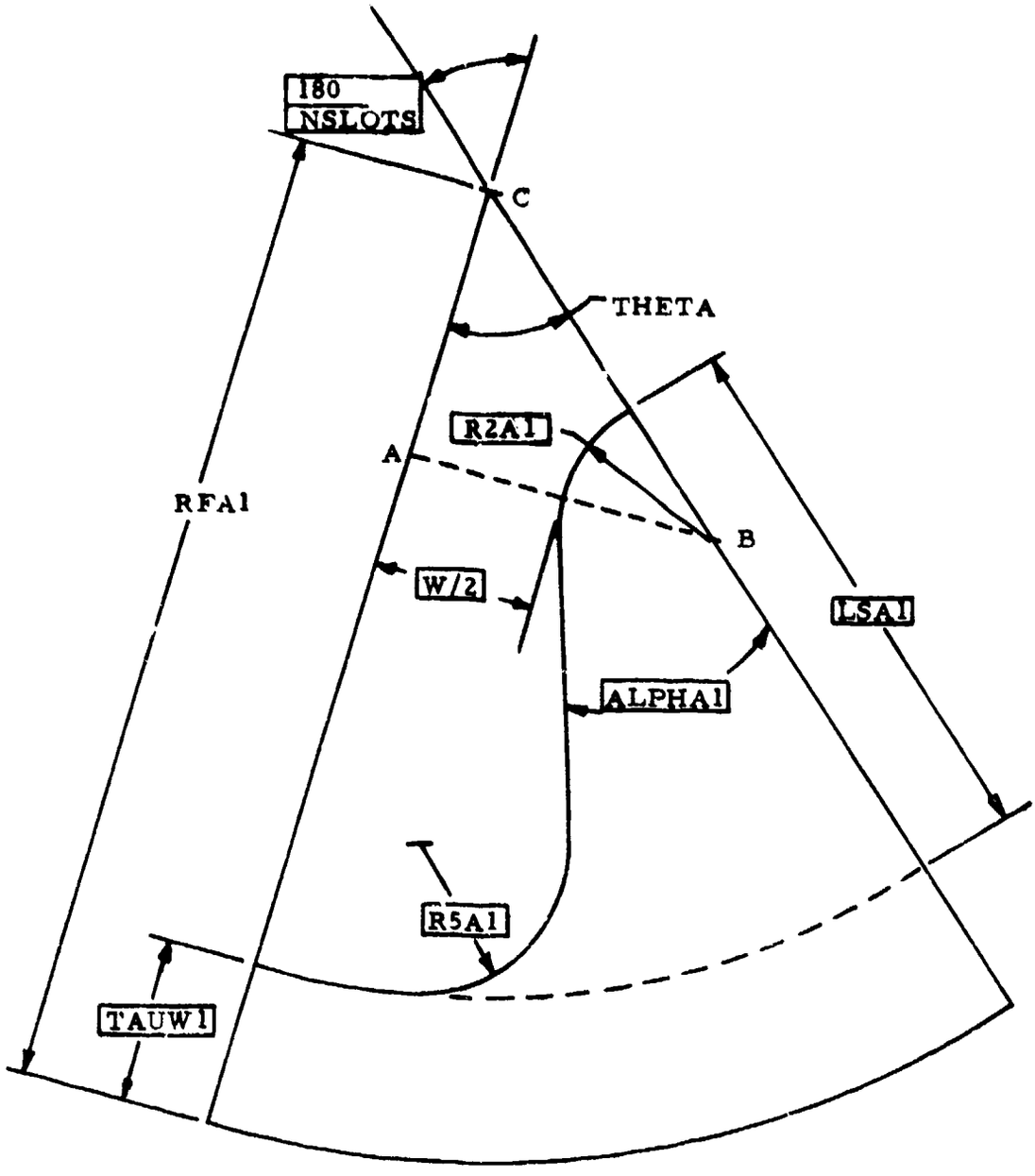
(a) Lengths of the various grain segments can be adjusted for all grains during the optimization process.
(b) For the Type 1 (Star) grain, all major dimensions describing the cross-section are either held constant at the initial input, or they are all varied during the optimization process.
(c) For the Type 2 (Wagon Wheel) grain, all dimensions describing the cross-section are either held constant at the initial input, or they are all varied during the optimization process.
(d) For the Type 3 (finocyl) grain, the major dimensions describing the cross-section can be individually held constant or allowed to vary during the optimization process.
(e) For the Type 4 (conocyl) grain, the major dimensions describing the cross-section can be individually held constant or allowed to vary during the optimization process.
(f) For the Type 5 (CP) grain, the dimensions describing the cross-section can be individually held constant or allowed to vary during the optimization process.

Basic arrangements of the closures are ellipsoidal and flat plate. Either of these arrangements can be used with any of the five grain configurations except that the Type 4 (conocyl) grain can be used only with the Type 1 (ellipsoidal) forward closure; this is because the definition of a conocyl grain positions the slot adjacent to the forward closure and the geometric grain regression routine was established for this arrangement.

Otherwise, the treatment of the closures is the same for all grains. The illustrations giving the plane locations show that any variation in grain dimensions along the length of the motor (e.g. port radius, star height) does not continue forward of Plane 1 or aft of Plane 14. In other words, the configuration that exists at Plane 14 is "projected" into the aft closure, even though in the actual motor the variation in port radius (for example) would probably continue until the end of the grain in the aft closure. This technique slightly overpredicts the volume of propellant. However, Plane 1 and Plane 14 are positioned only a short distance of TAUMXF and TAUMXA, respectively, from the grain ends where TAUMXF and TAUMXA are the maximum distances burned at Plane 1 and Plane 14, respectively.

Type 1 (Star) Grain

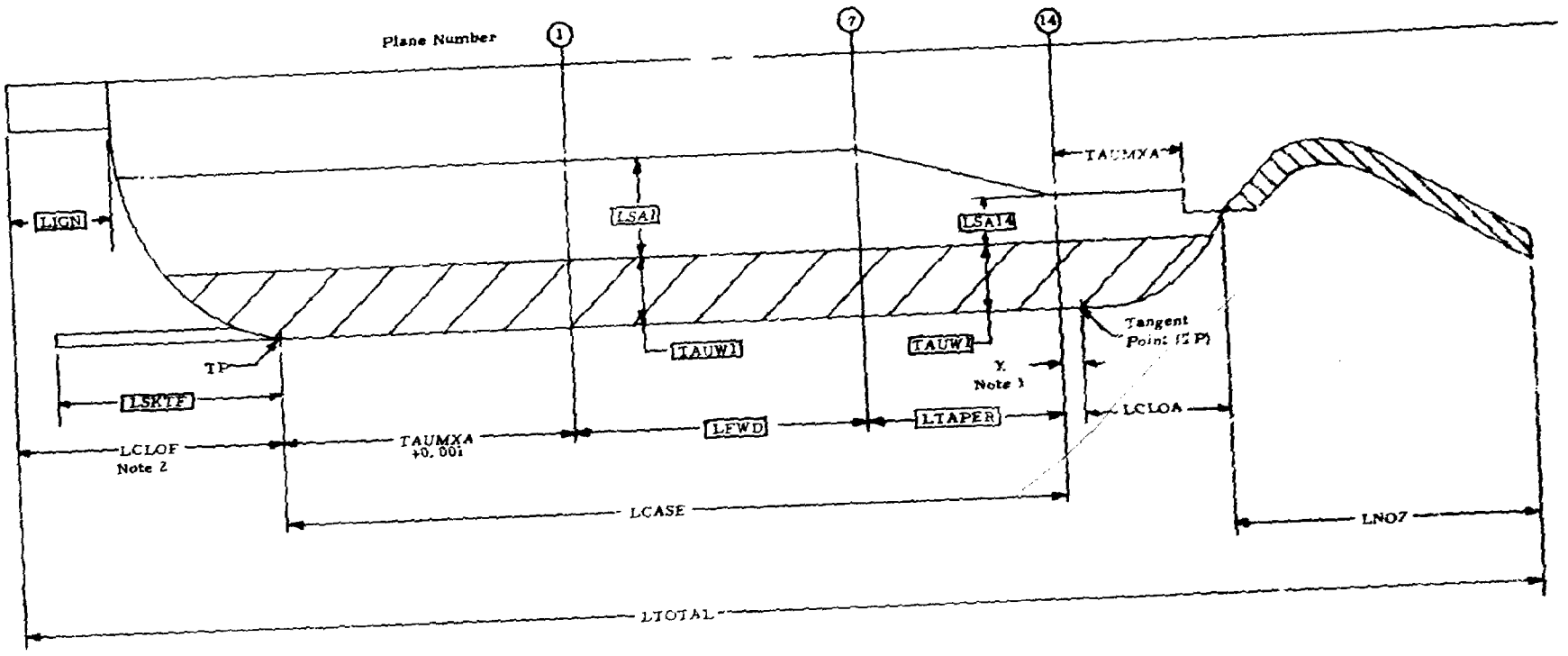
Type 1 grain is a standard star configuration (Figure 31). Locations of direct input planes are shown in Figures 32 and 33 for the two closure types. Dimensions shown in blocks on Figure 31 are for Plane 1, which are held constant to Plane 7; then the height of the star tip tapers from LSA1 at Plane 7 to LSA14 at Plane 14. All other dimensions are constant from Plane 7 to Plane 14.



Notes

(1) Line AB is perpendicular to Line AC.
(2) Dimensions in blocks are input; others are output

Figure 31. Type 1 (Star) Grain Configuration



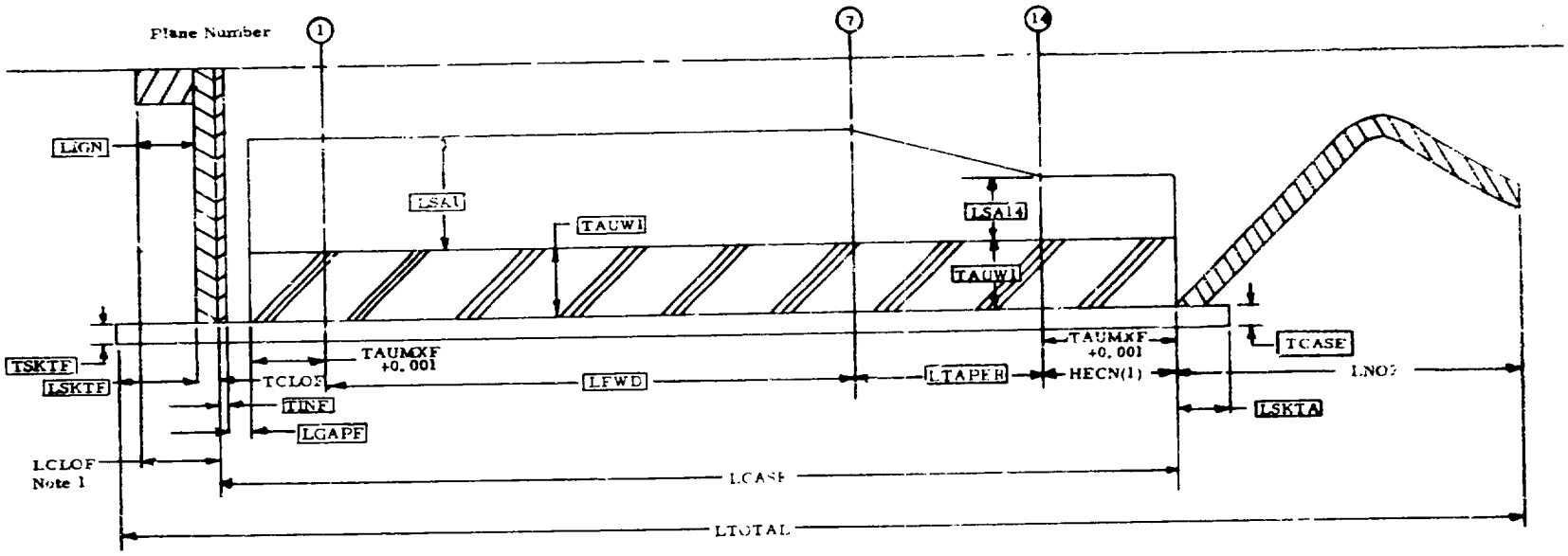
NOTES

(1) For Alt Closure Type 1, X = DHECN(1)-X1A) if RFA14 <B5A, or is X = HECN(1) if RFA14≥ B5A

(2) LSKTF used to calculate *LTOTAL if LSKTF* > LCLOF

(3) Dimensions shown in blocks are *input; others are output*

Figure 32. Location of Direct Input Planes for Type 1 (Star) Grain with Type 1 Closures



NOTES

(1) LCLOF used to calculate LTOTAL if LCLOF > LSKTF

(2) Dimensions shown in blocks as input; others are output

Figure 33. Location of Direct Input Planes for Type 1 (Star) Grain with Type 2 Closures

Subroutine SETUP1 performs the validation checks described above for the Type 1 grain. The logic flow for the cross-section dimension checking is shown in Figure 34.

There are two categories of defining variables. First are those that are fixed (insofar as SETUP1 is concerned) and are never adjusted. If these data are found to be unacceptable, the run is terminated with an appropriate message.

RFA1	Propellant outside radius. Is actually adjusted through changes in motor diameter and case thickness, but SETUP1 has no control over these.
R2A1	User-selected input
THETA	180/NSLOTS. See Figure 31.
W	Minimum clearance between adjacent star tips
WFLIM	User-selected input of maximum allowable web fraction

Second are those dimensions that are varied by PATSH during the optimization process. If these data are found to be unacceptable, they are adjusted in accordance with a hierarchy described below.

TAUW1	Web thickness
LSA1	Star point height above web
R5A1	Fillet radius between star point and web
ALPHA1	Included half-angle of star point

The <u>first group</u> of tests involve the fixed inputs.

$$RFA1 > 0 \quad (1)$$

$$R2A1 \geq 0 \quad (2)$$

$$THETA > 0 \quad (3)$$

$$W \geq 0$$

$$(R2A1 + W/2) < (RFA1)\mathrm{Sin}(THETA) \quad (4)$$

$$0 < WFLIM < 1.0 \quad (5)$$

Failure of any of the above tests results in a run termination.

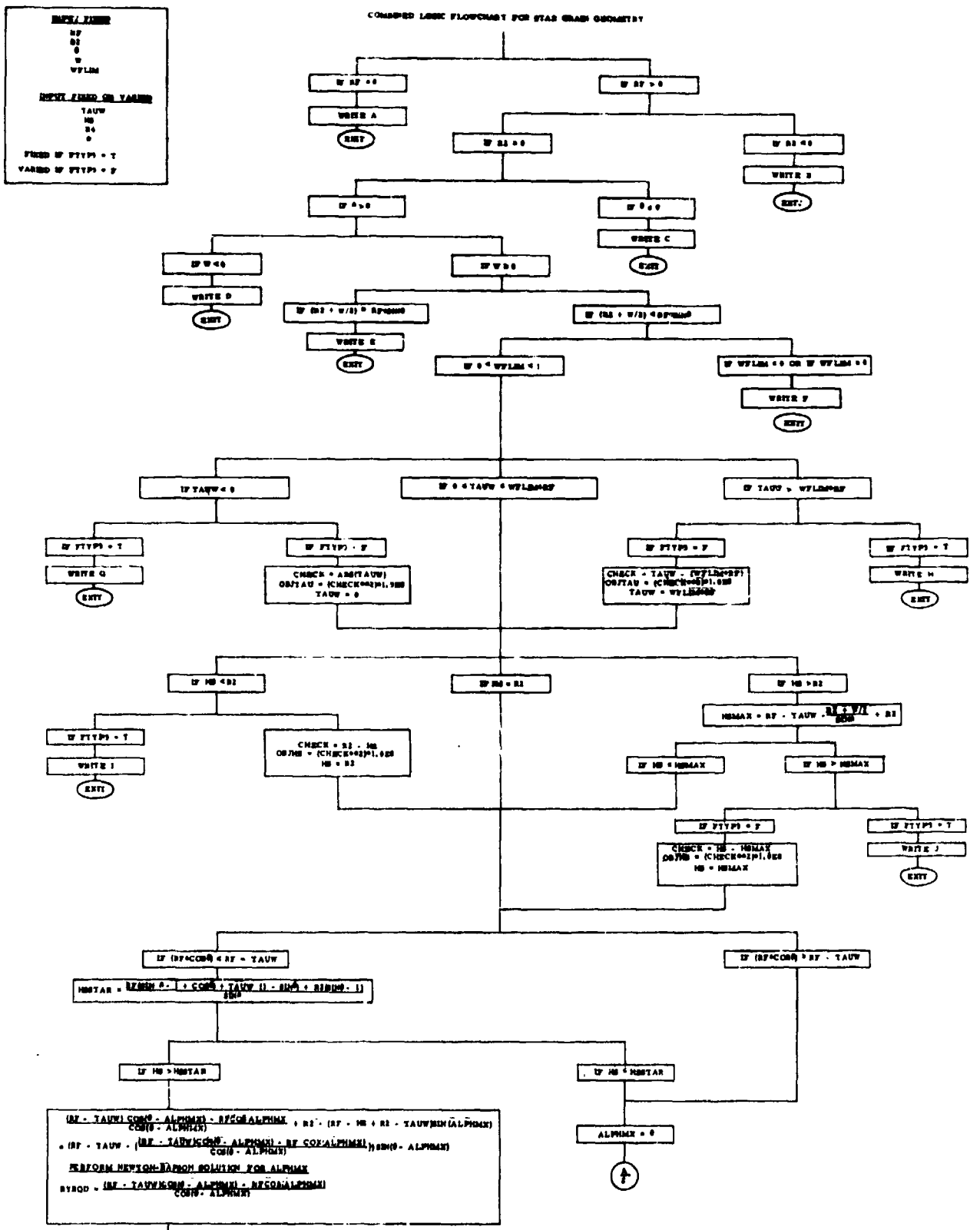


Figure 34. Logic Flow Chart for Type 1 (Star) Grain

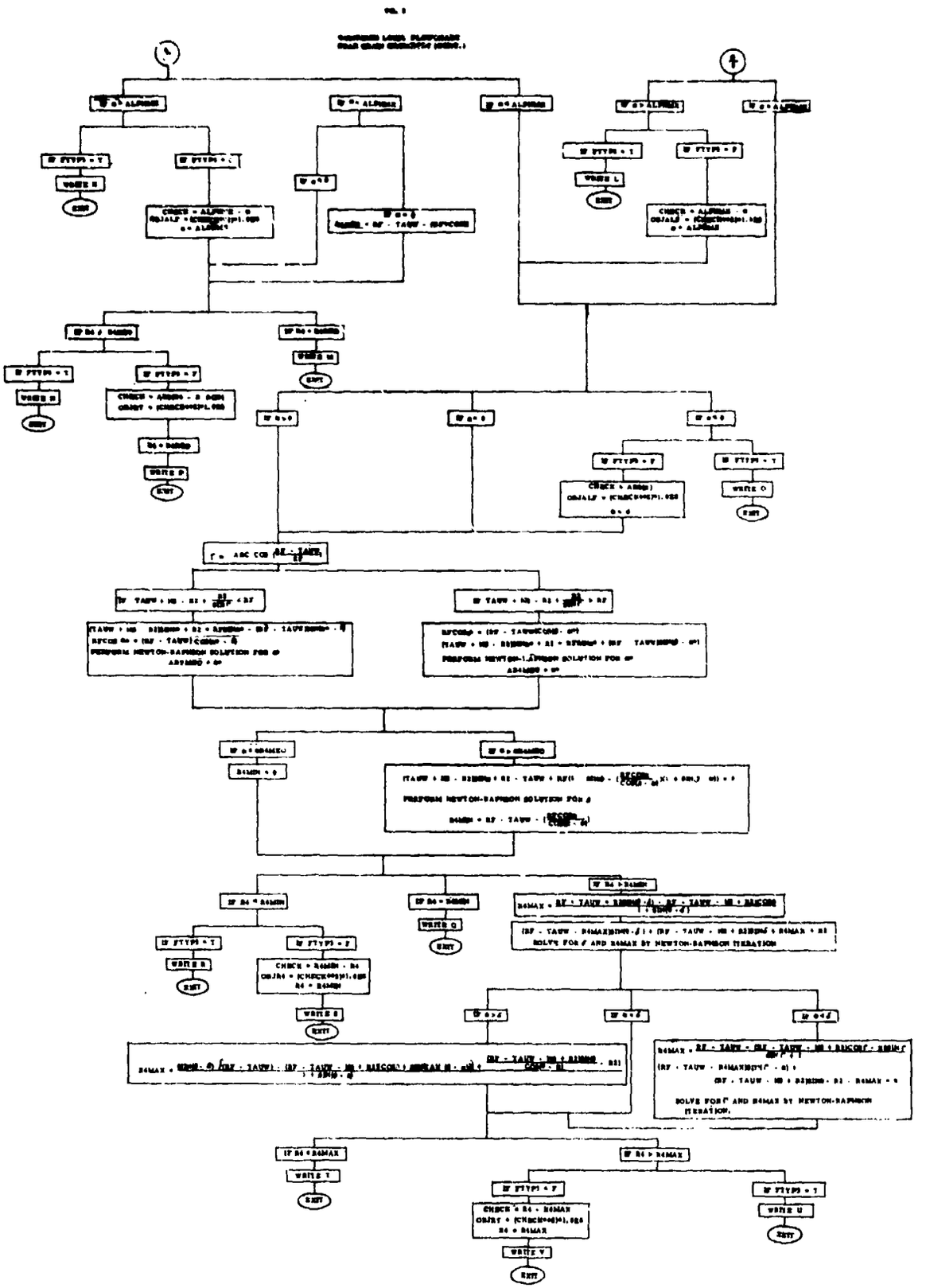


Figure 34 Continued (Part 2 of 2)

The second group of tests are:

$$0 \leq TAUW1 \leq (WFLIM)(RFA1) \quad (6)$$

$$R2A1 \leq LSA1 \leq LSMAX \quad (7)$$

where $$LSMAX = RFA1 - TAUW1 + R2A1 - (R2A1 + W/2)/Sin(THETA) \quad (8)$$

Failure of Eq. (6) causes TAUW1 to be set to the limit. Failure of Eq. (7) causes LSA1 to be set to the limit. Penalties are calculated in both instances.

At this point, only ALPHA1 and R5A1 of the four adjustable variables remain to be tested. The third group of tests is concerned with ALPHA1. The decision logic is shown on Figure 35. The variable LSSTAR is the maximum star tip height possible when ALPHA1 = THETA, for the situation where RFA1 Cos (THETA) < (RFA1 - TAUW1) and RFA1, TAUW1, R2A1 and THETA have been previously accepted (Figure 36). The following possibilities can exist:

If LSA1 > LSSTAR	Maximum ALPHA1 must be less than THETA
If LSA1 = LSSTAR	Maximum ALPHA1 equals THETA and R5A1 is unique at input value
If LSA1 < LSSTAR	Maximum ALPHA1 equals THETA and multiple solutions exist for RFA1

When LSA1 is greater than LSSTAR (the only situation where maximum ALPHA1 ≠ THETA), maximum ALPHA1 is determined from the criteria shown on Figure 37. Note from Figure 35 that minimum ALPHA1 is always zero. If ALPHA1 in the current dimension set is less than zero or greater than maximum ALPHA1, it is set to the limit and an appropriate penalty calculated.

The fourth group of tests is concerned with R5A1, the last dimension needed to define the cross-section geometry. A portion of the R5A1 logic diagram is given in Figure 38. The first step is to solve for the angle AFSTAR (Figure 39), where R5A1 can be zero, subject to the criteria shown on Figure 39. Thus,

ALPHA1 ≤ AFSTAR	Minimum R5A1 = 0
ALPHA1 > AFSTAR	Minimum R5A1 > 0

Therefore, the next step is to determine minimum R5A1 when ALPHA1 > AFSTAR, and it is shown in Figure 40. The final step of this group of tests is to determine maximum R5A1. The solution for maximum R5A1 depends on the regime in which ALPHA1 is located; this regime is defined by the angle DELTA, which is shown on Figure 41. DELTA

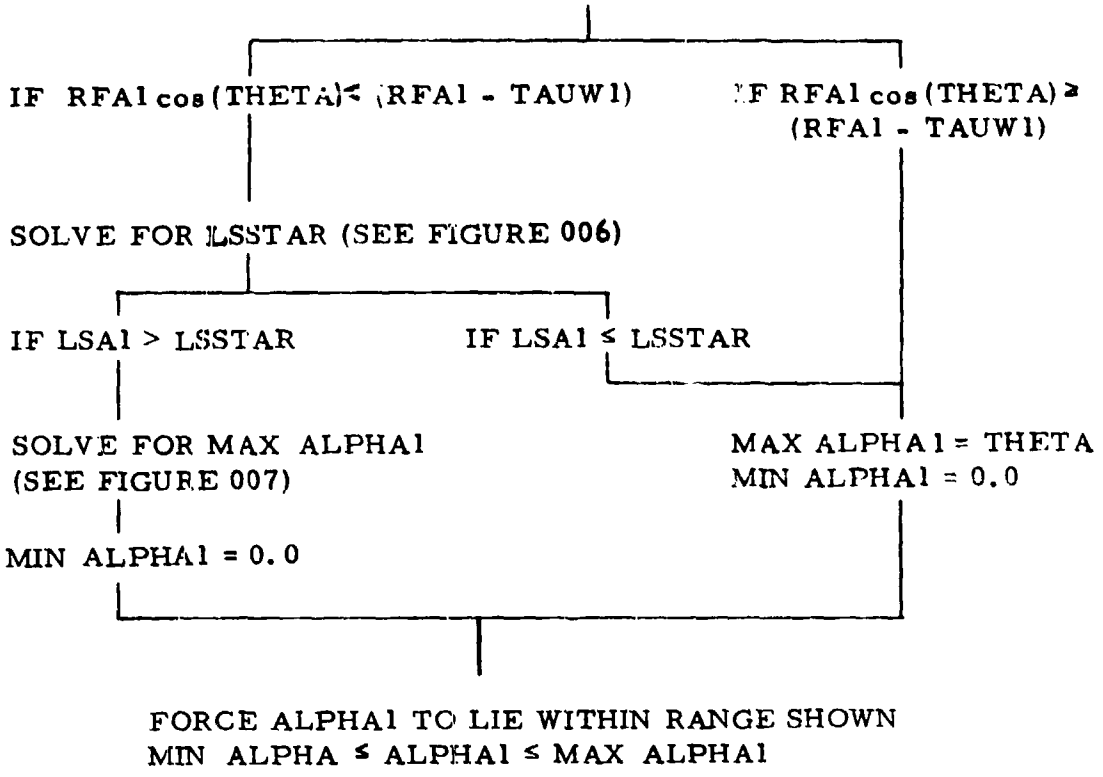


Figure 35. Testing of ALPHA1 for Type 1 Grain

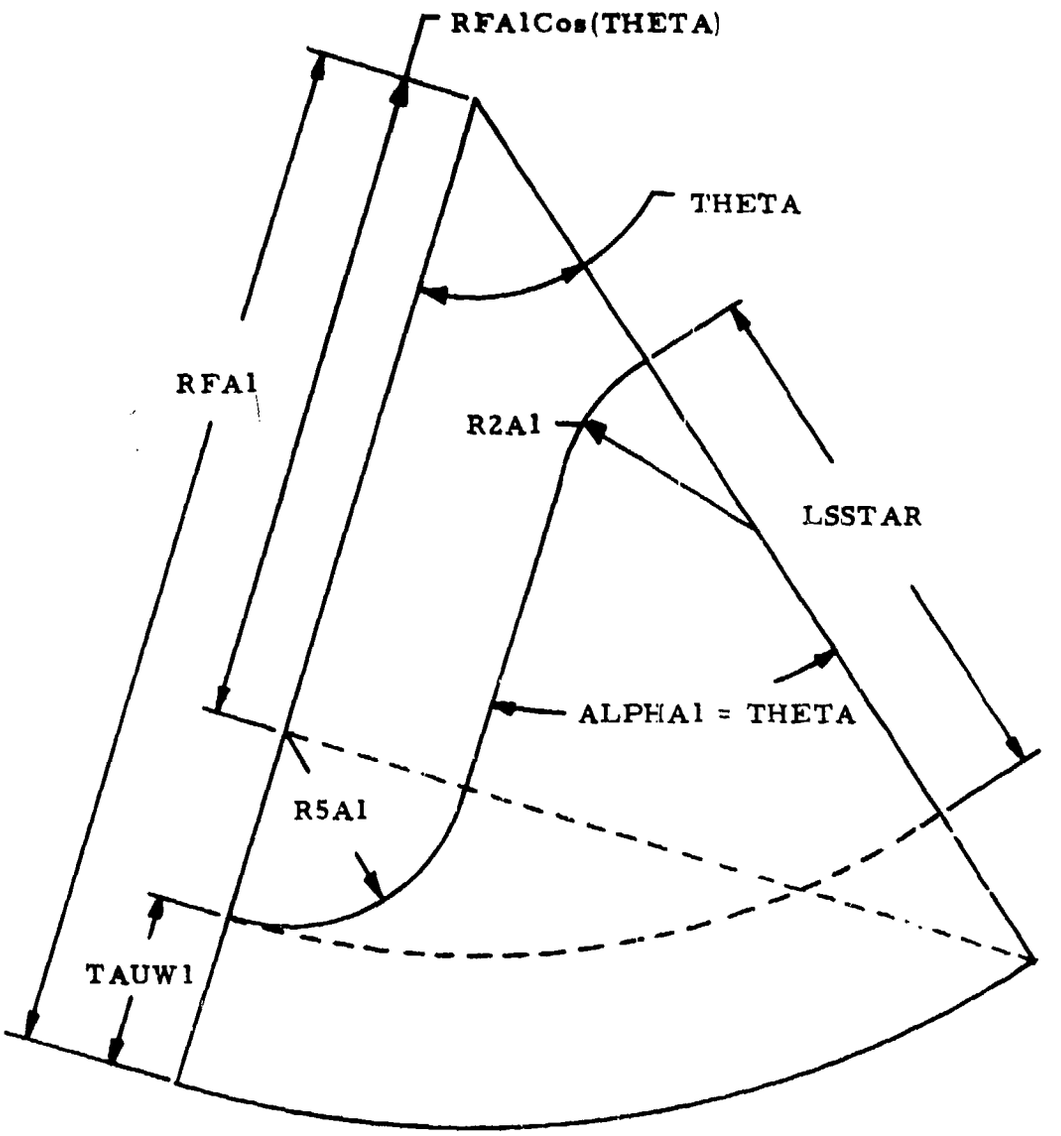


Figure 36. Definition of LSSTAR, Type I Grain

Limiting Criteria: A line from the center of R5A1 (which is Point A) and perpendicular to line BD must not intersect line CE at a point further from Point C than Point E.

Subject to: RFA1, TAUW1, THETA, R2A1, LSA1 already accepted

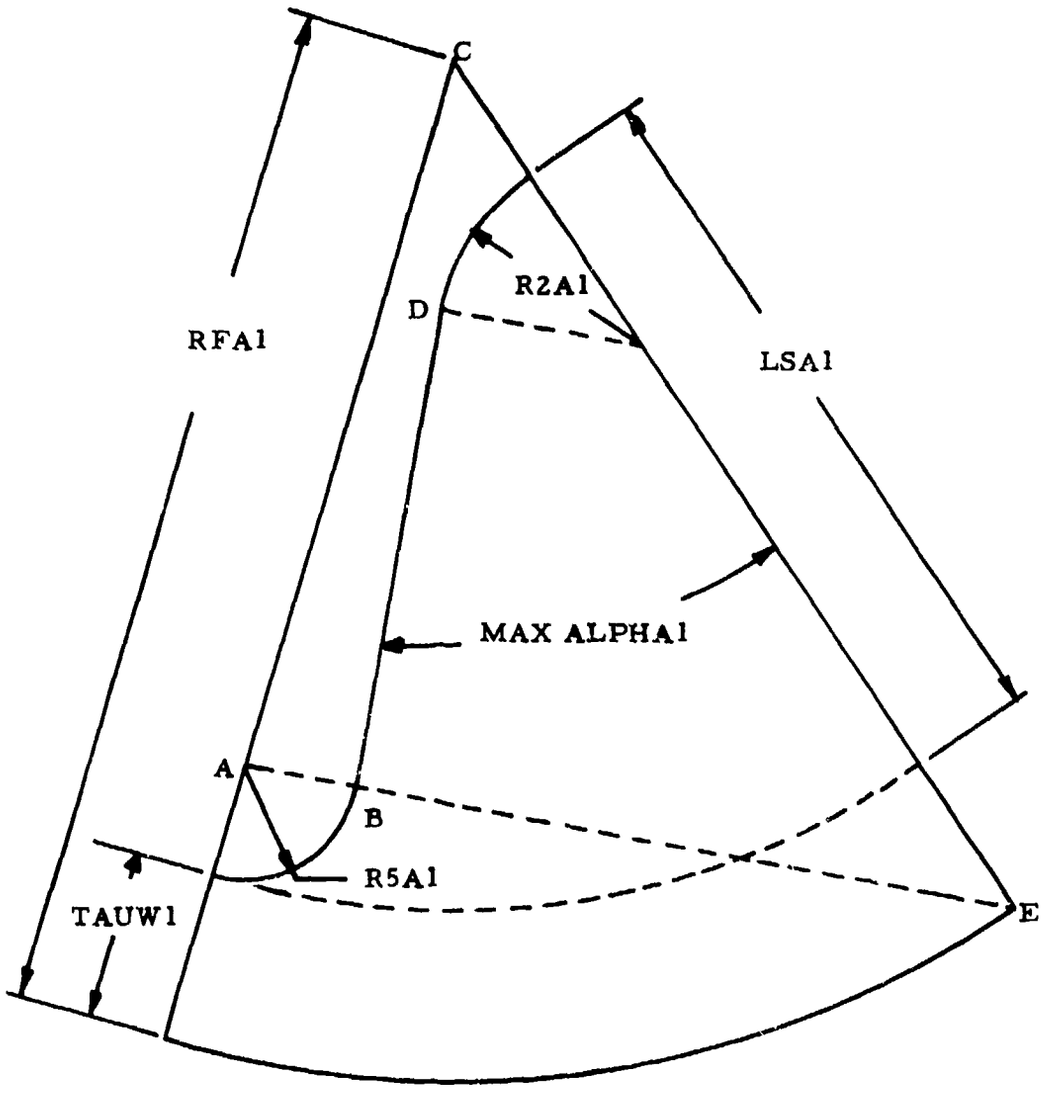


Figure 37. Definition of Maximum ALPHA1 when LSA1 > LSSTAR, Type 1 Grain

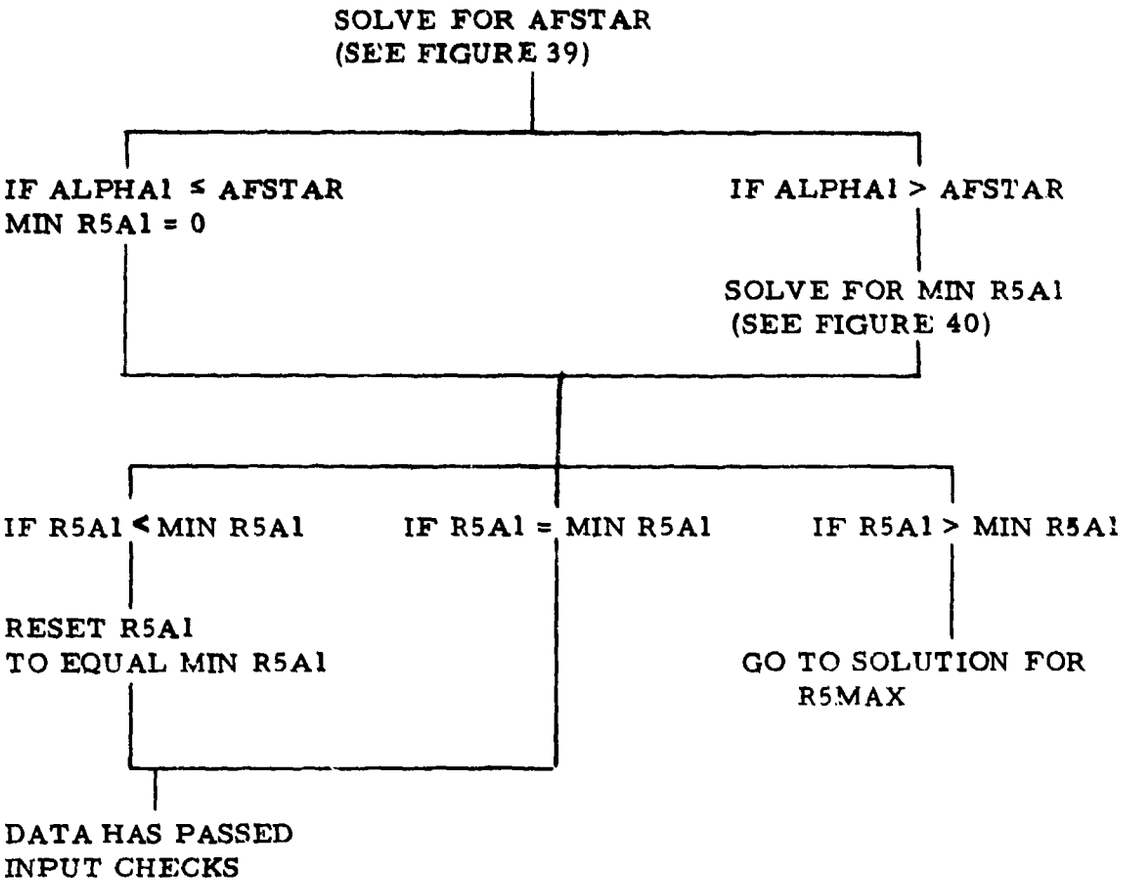


Figure 38. Logic Diagram for AFSTAR, Type 1 Grain

Limiting Criteria: A line from Point B and perpendicular to line AB cannot intersect line CAD at a point further from Point C than Point D

Subject to: RFA1, TAUW1, THETA, R2A1, LSA1 already accepted

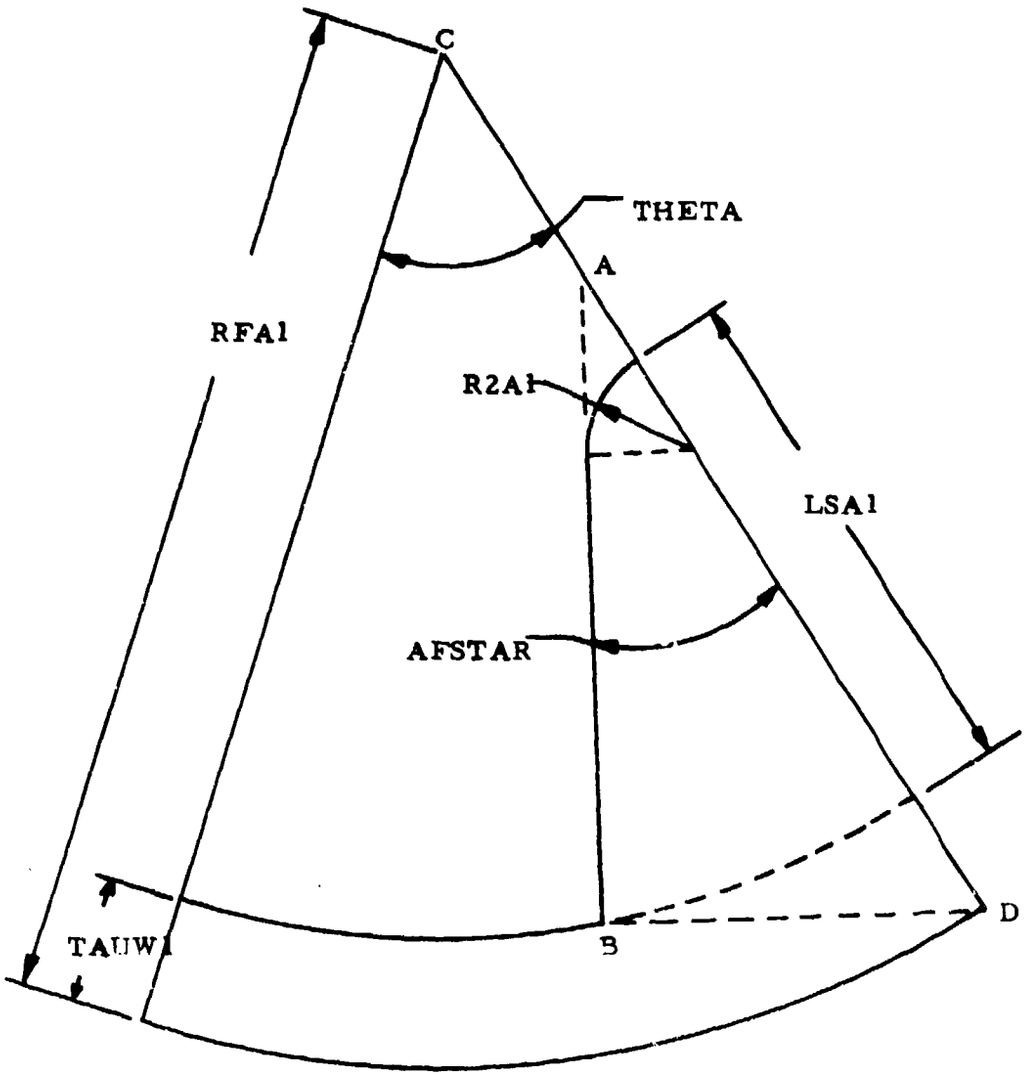


Figure 39. Definition of AFSTAR, Type 1 Grain

Limiting Criteria: **A line from the center of R5A1 (which is Point A) and perpendicular to line BE must not intersect line CD at a point further from Point C than Point D.**

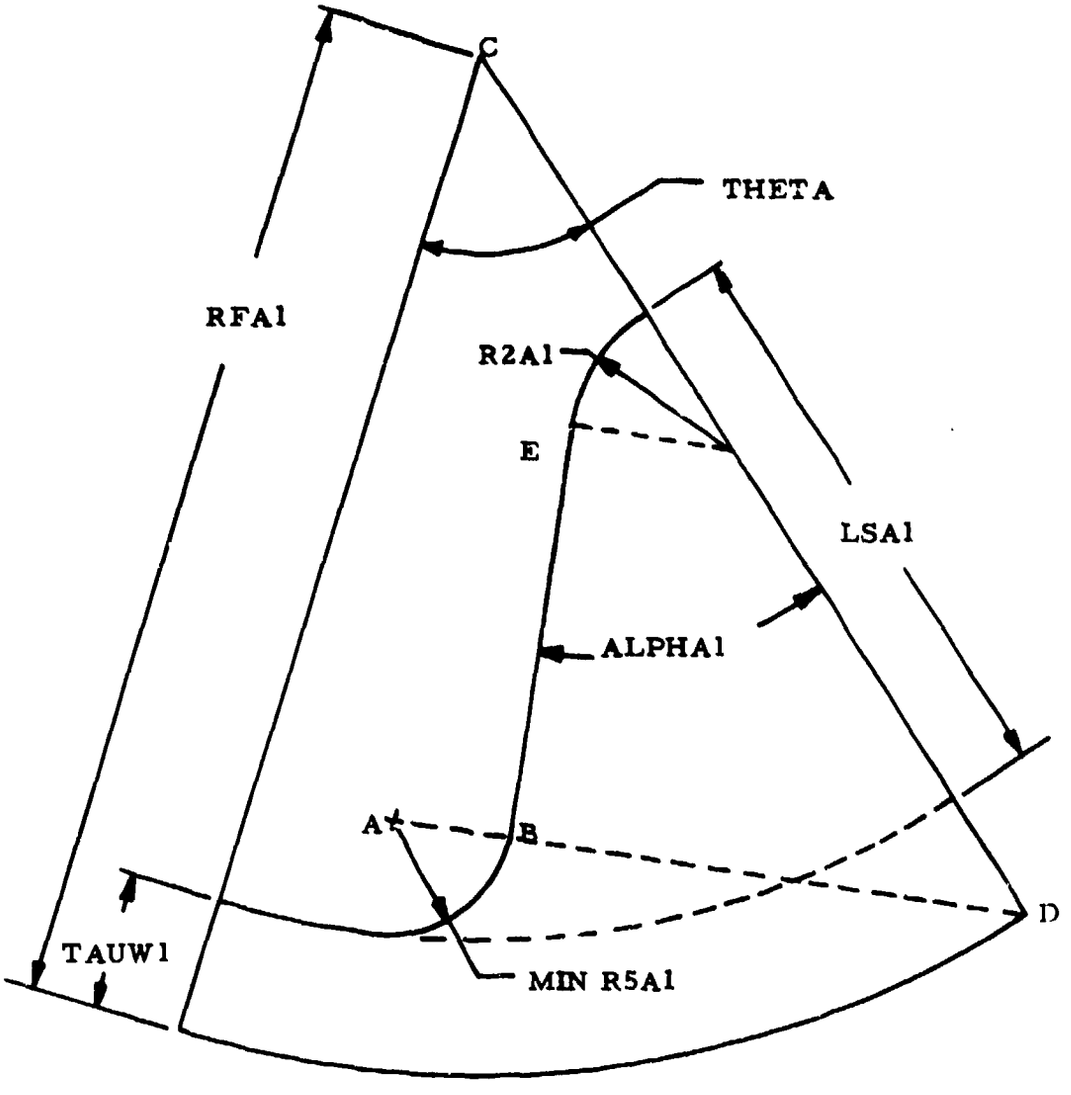


Figure 40. **Minimum RFA1 when ALPHA1 > AFSTAR, Type 1 Grain**

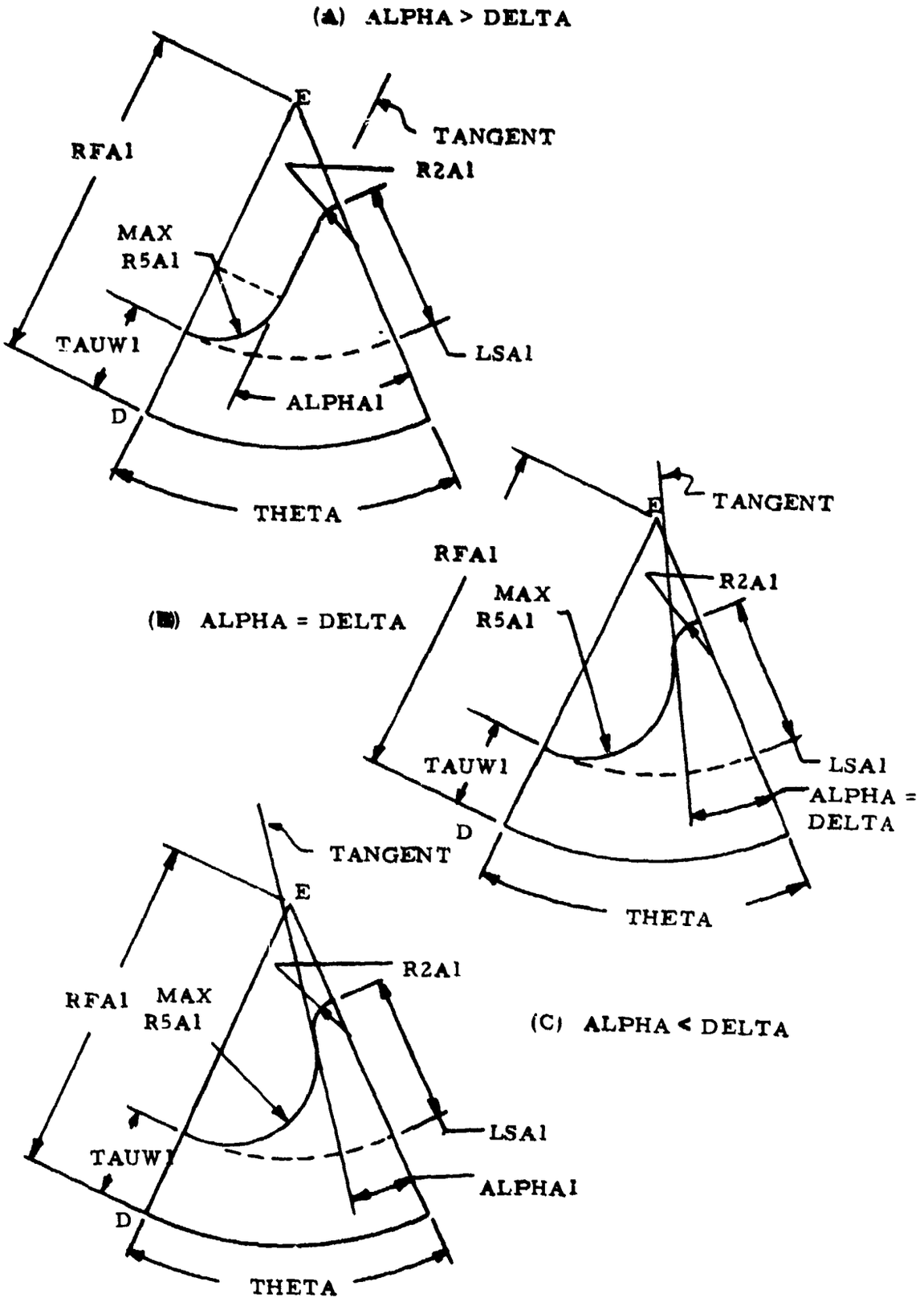


Figure 41. Definition of DELTA, Type 1 Grain

(illustrated in part B of Figure 41) is the unique value of ALPHA1 at which the center of R5A1 radius lies on the line DE and a line tangent to the R2A1 and R5A1 arcs is perpendicular to a line connecting their centers. For ALPHA1 > DELTA, (part A of Figure 41), the center of R5A1 radius lies on the line DE, but a line connecting the centers of R2A1 and R5A1 is not perpendicular to a line tangent to the R2A1 and R5A1 arcs. For ALPHA < DELTA (part C of Figure 41), the center of R5A1 radius lies inside line DE, and a line tangent to the R2A1 and R5A1 arcs is perpendicular to a line connecting their centers. A test is made to determine in which regime ALPHA1 is located, and appropriate geometric relationships are solved to find maximum R5A1. Then, the final test is to insure that R5MIN ≤ R5A1 ≤ R5MAX. If R5A1 is outside the limits, it is set to the limits and a penalty is calculated.

Now that all dimensions for Plane 1 are acceptable, star point height at Plane 14 (LSA14) is tested to insure R2A1 ≤ LSA14 ≤ LSA1, where R2A1 is the same minimum height requirement at Plane 1.

The final tasks of subroutine SETUP1 are to insure that grain lengths are greater than zero, calculate inert weights, calculate data for use in propellant structural analysis, translate the grain dimensions into language required by the ballistic simulation module, and other miscellaneous tasks. The dimension L3 (Figure 42) is derived for use in propellant structural analysis, through the following expression:

$$LITD = (L3 + R5)(2) \tag{9}$$

so that LITD becomes the distance across the propellant valley. The distance $T_{w_{max}}$ shown on Figure 42 is the maximum distance burned for the star grain and is used to establish the positions of the first and last planes (that describe the grain to the ballistic simulation module) with respect to the forward and aft closures (e.g., see Figure 26).

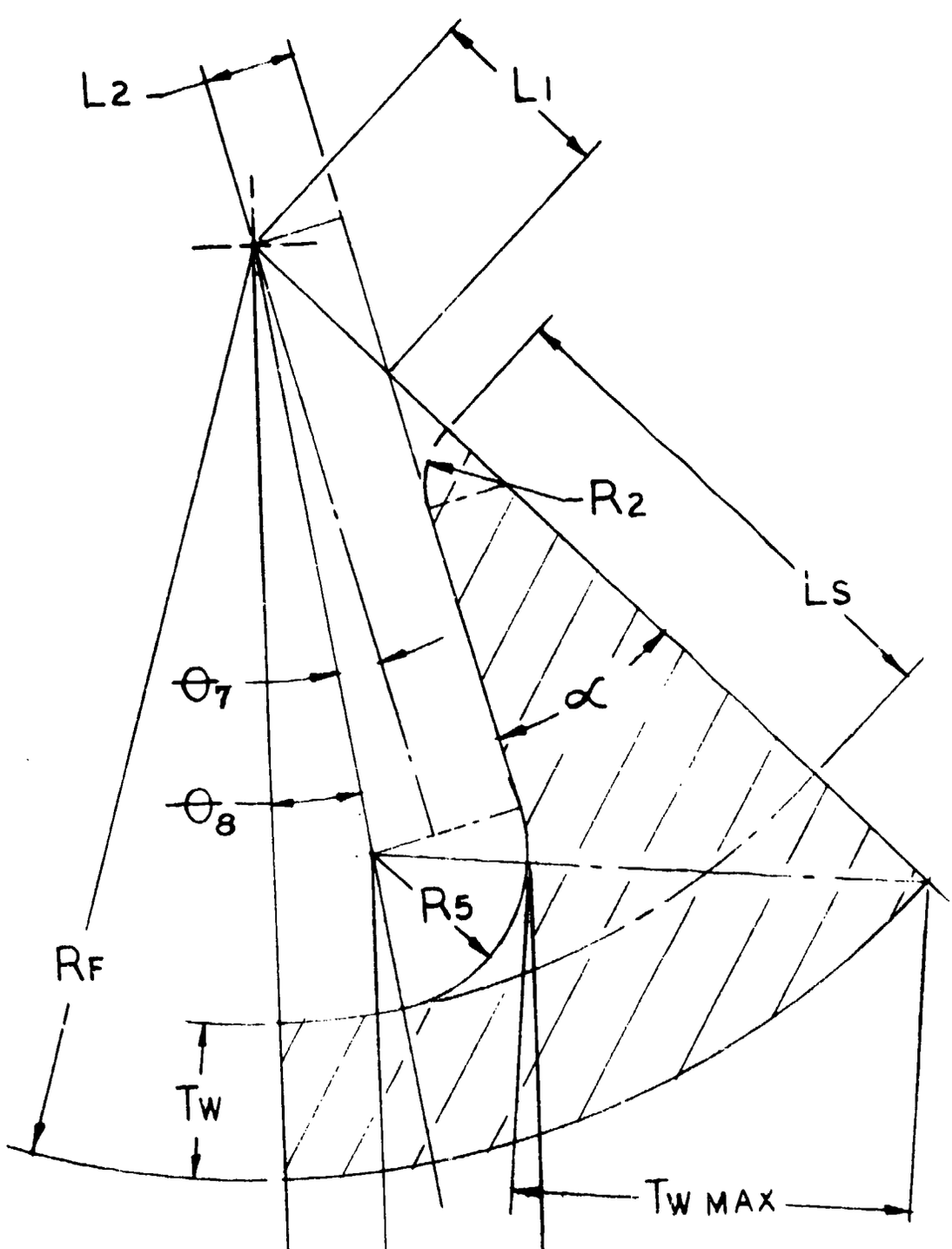


Figure 42. Dimensions for Propellant Structural Analysis, Type 1 Grain

Type 2 (Wagon Wheel) Grain

Type 2 grain is a double-web wagon-wheel configuration (Figure 43). Locations of direct input planes are shown in Figures 44 and 45 for the two closure types. Dimensions shown in blocks on Figure 43 are for Plane 1, which are held constant to Plane 7; then, the height of the star point can taper from LSA1 at Plane 7 to LSA14 at Plane 14. All other dimensions are constant from Plane 7 to Plane 14 as long as they are geometrically compatible with the reduced LSA14. Basic dimensions describing the grain cross-section (TAUW1, LSA1, R5A1) can all be varied during the optimization process, or they all may be held constant at the initial input.

Subroutine SETUP2 performs the validation checks described above for the Type 2 grain. The logic flow for the cross-section dimension checking is shown in Figure 46 for all variables being adjusted and in Figure 47 for all variables held constant. Although two different flow charts are shown, the two routines have been merged in the code.

There are four categories of defining variables. First are those that are fixed (insofar as SETUP2 is concerned) and are never adjusted. If these data are found unacceptable, the run is terminated with an appropriate message.

RFA1 > 0	Propellant outside radius. Is actually adjusted through motor diameter and case thickness, but SETUP2 has no control over these.
THETA > 0	180/NSLOTS
WO2MIN $\geq$ 0	Half the minimum clearance between adjacent star points

Second are those input data that are accepted if possible, but SETUP2 will adjust them if required to avoid a run termination.

WFLIM	User-selected input of maximum allowable web fraction.

This input is accepted if possible. If not, it is adjusted to be compatible with the remaining dimensions in order to prevent run termination. It is reset to the appropriate values with no penalties being calculated. A geometrically maximum possible web fraction (independent of the user input) exists when TAUW1 increases to the point where R5A1 is driven to zero and WO2 is equal to WO2MIN (Figure 48); this condition results when the length LC (shown in Figure 43) is zero. At this condition, web thickness TAUW1 is the largest possible and

(1) TAUMAX shown in Figure 48 is not the maximum distance burned used to position ballistic planes; the latter has the same definition as in Figure 42.

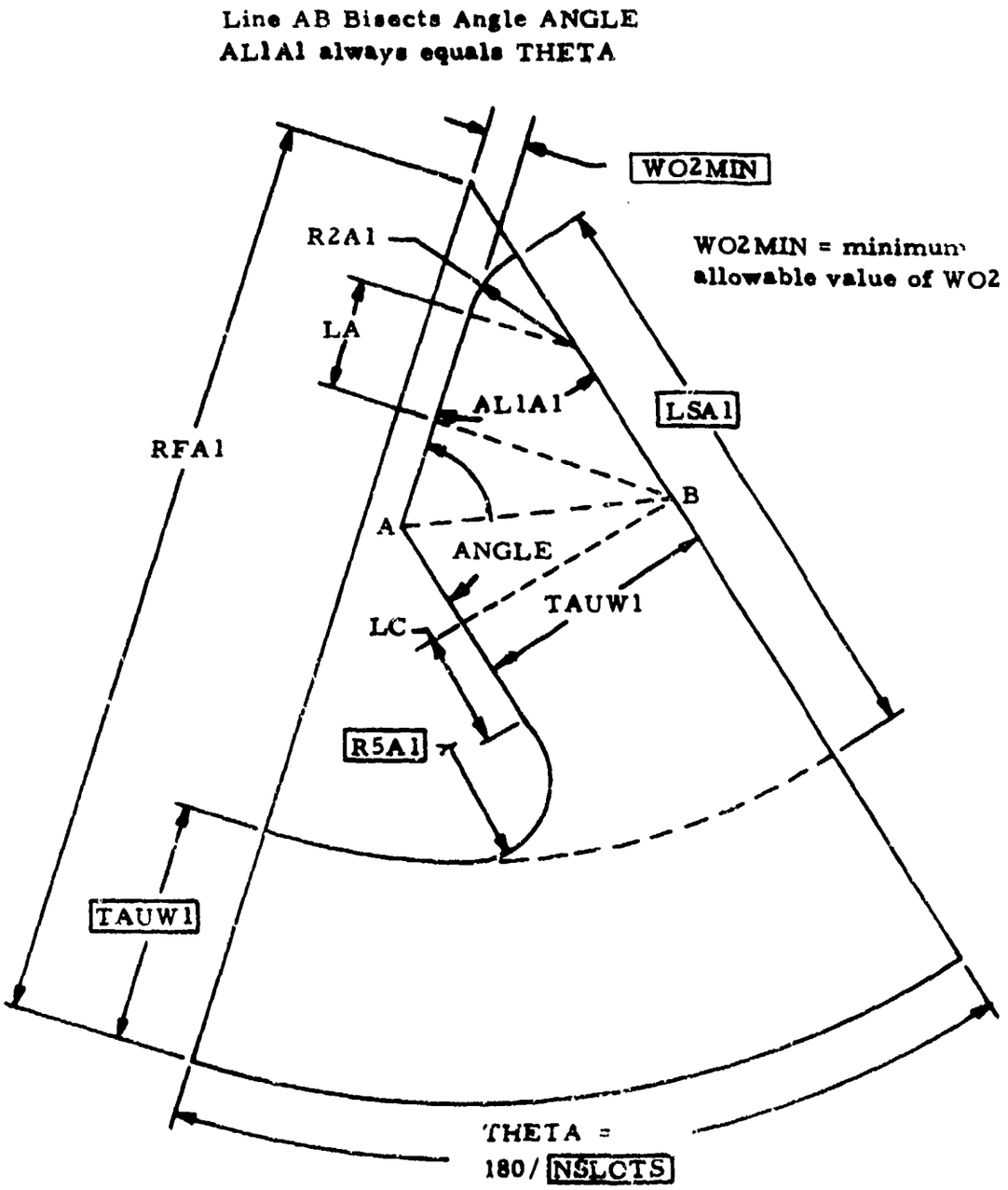
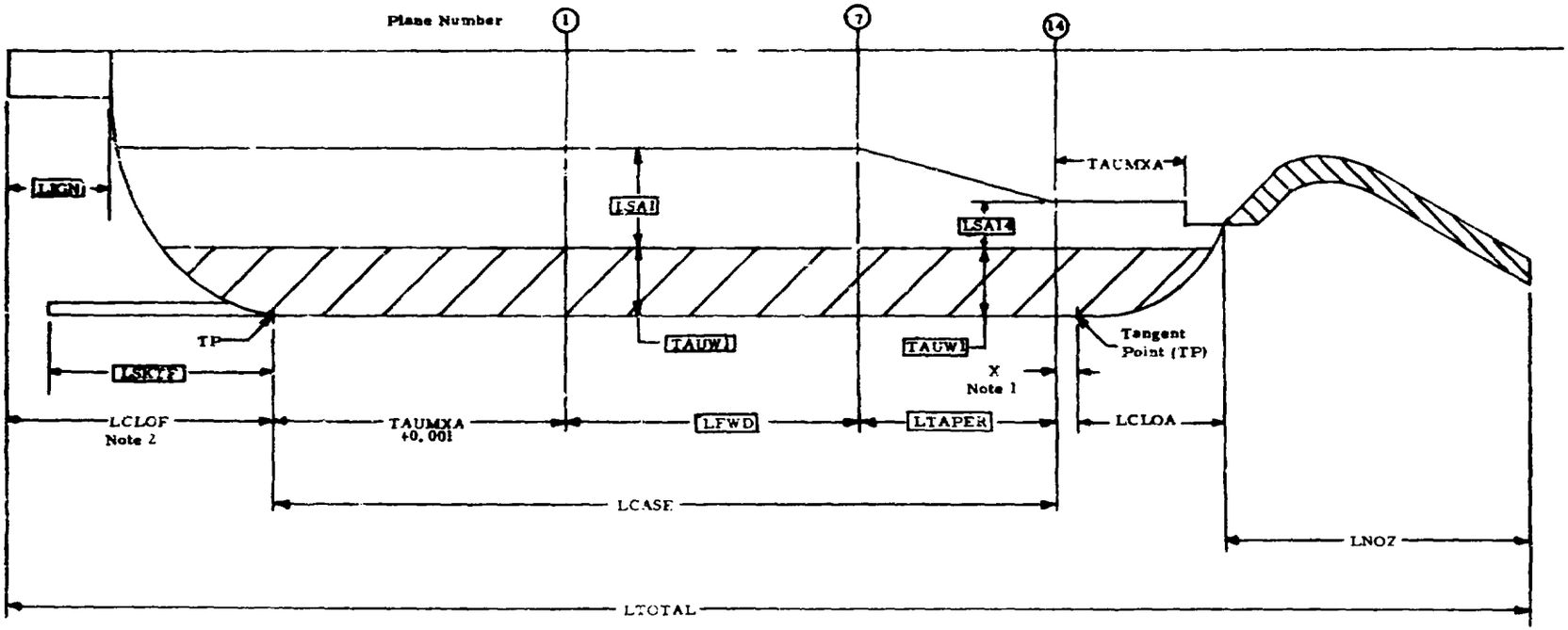


Figure 43. Type 2 (Wagon Wheel) Grain Configuration



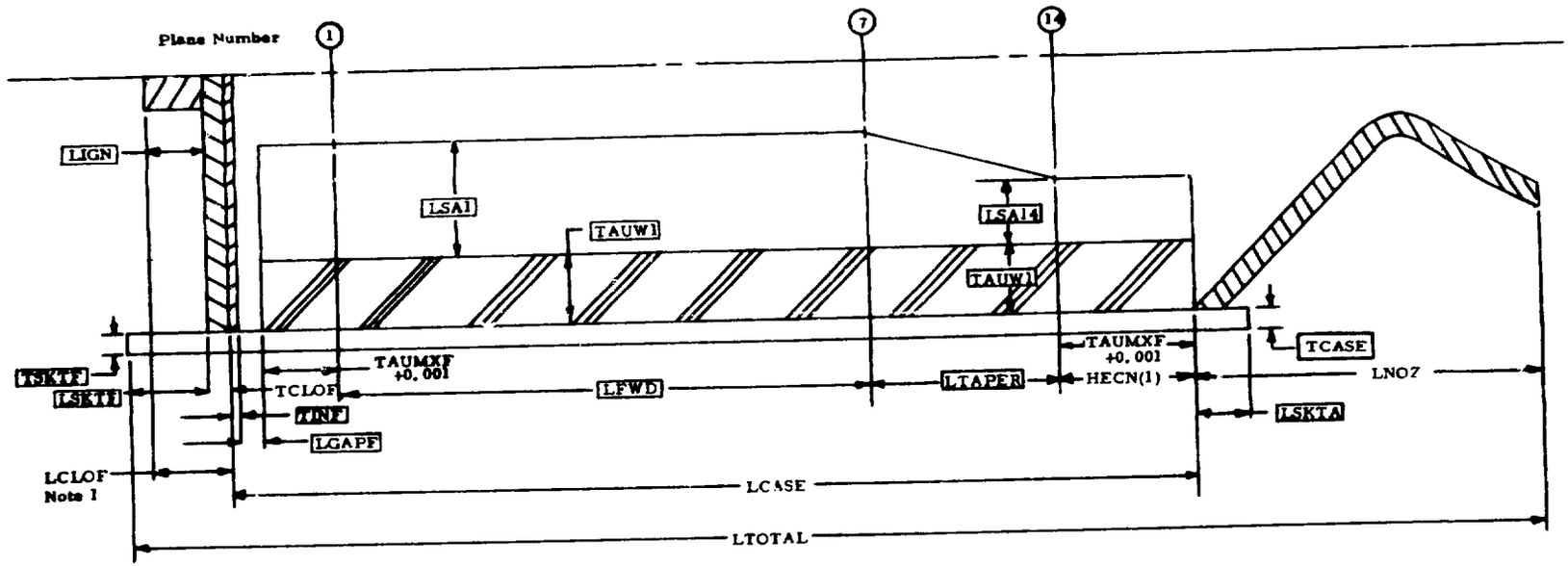
NOTES

(1) For Aft Closure Type 1, X = (HECN(1)-X1A) if RFA14 <B5A, or is X = HECN(1) if RFA14 ≥ B5A

(2) LSKTF used to calculate LTOTAL if LSKTF > LCLOF

(3) Dimensions shown in blocks are input; others are output

Figure 44. Location of Direct Input Planes, Type 2 Grain



NOTES

(1) LCLOF used to calculate LTOTAL if LCLOF > LSKTF

(2) Dimensions shown in blocks as input; others are output

Figure 45. Location of Direct Input Planes, Type 2 Grain

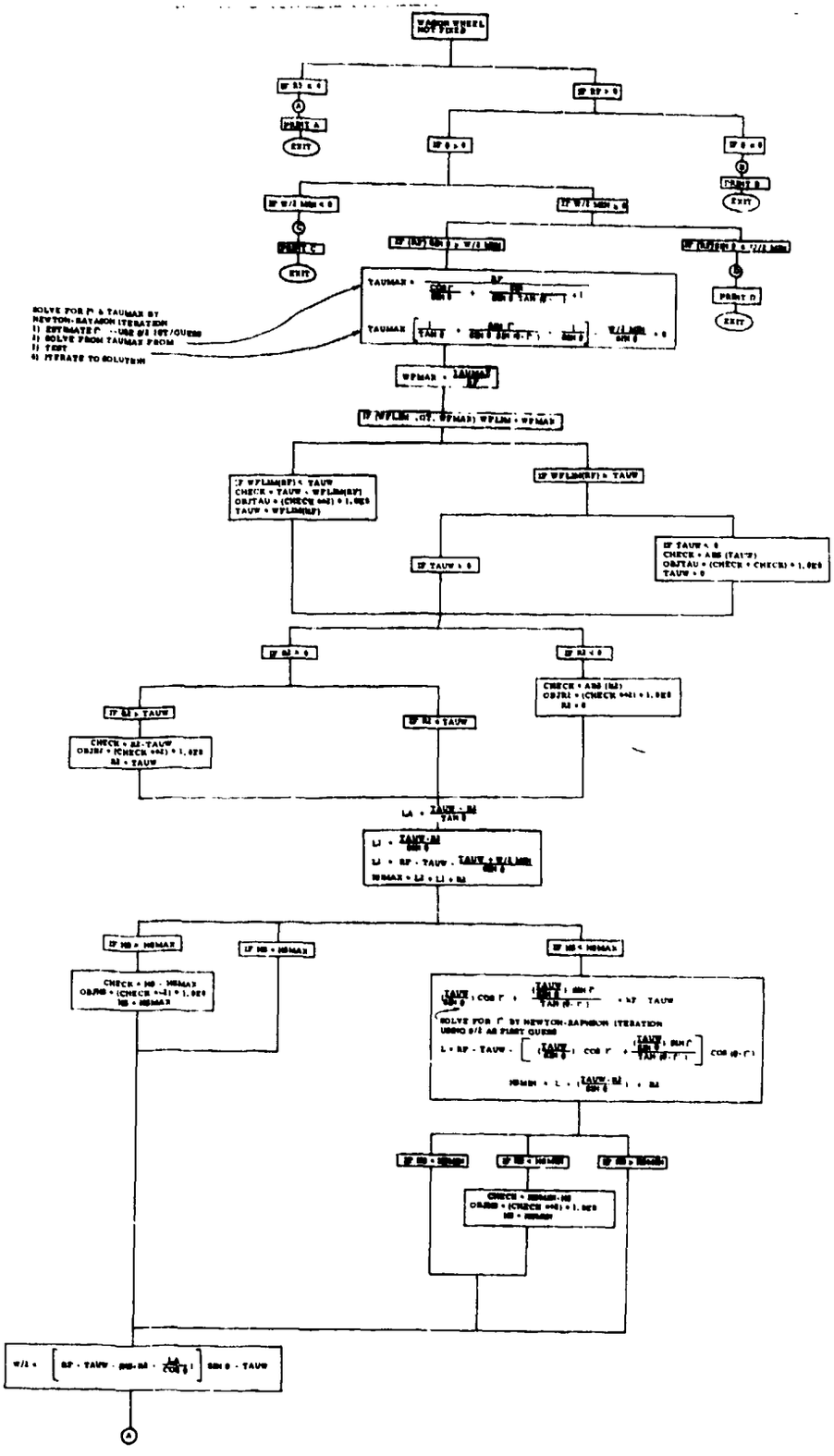


Figure 46. Logic Flow Chart For Type 2 (Wagon Wheel) Grain (All Variables Adjusted)

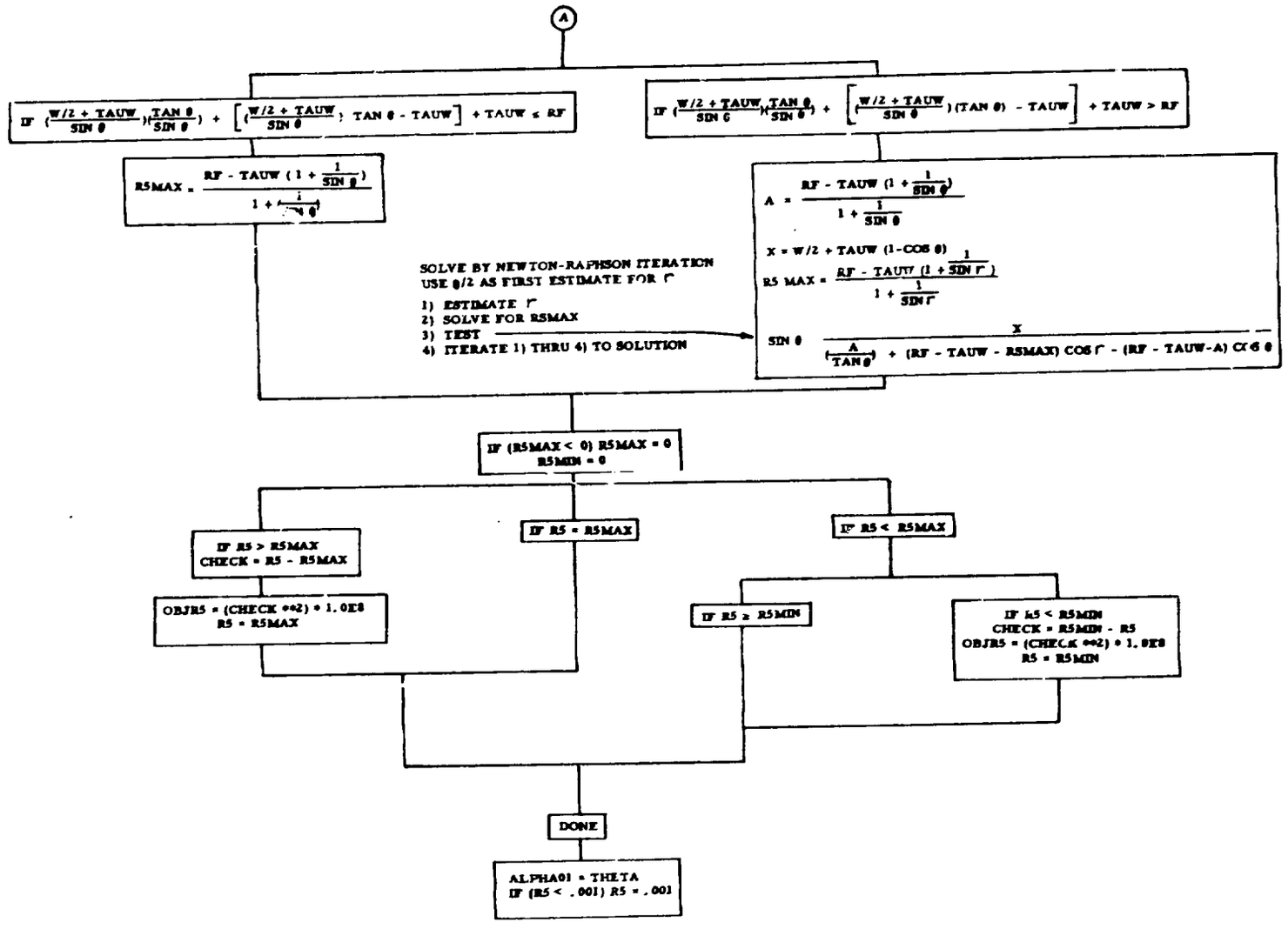


Figure 46. (Continued)

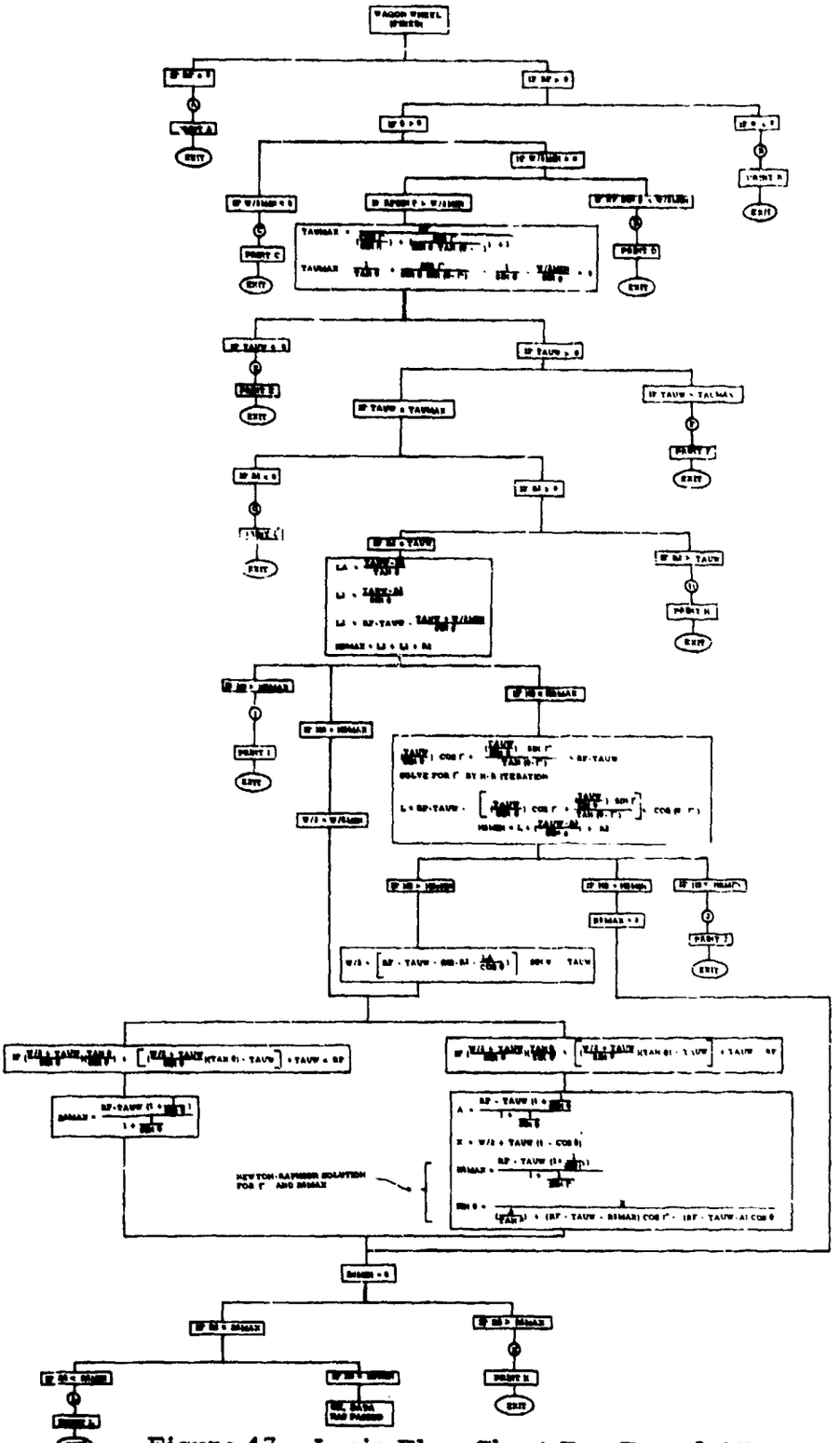


Figure 47. Logic Flow Chart For Type 2 (Wagon Wheel) Grain (All Variables Not Adjusted)

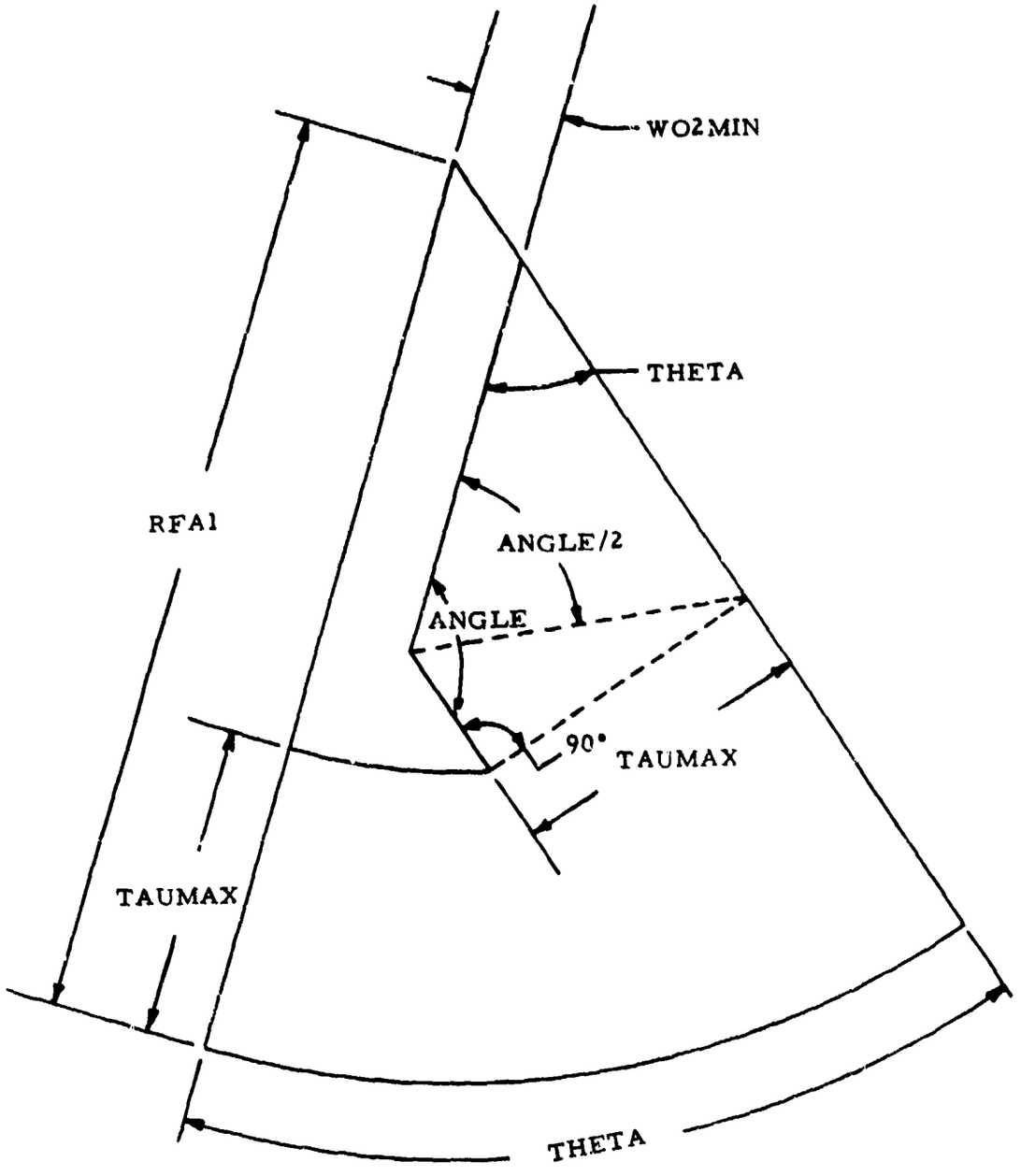


Figure 48. Maximum Web Thickness TAUMAX For Wagon Wheel (Type 2) Grain

$$WFMAX = \frac{TAUMAX}{RFA1} \tag{9}$$

WFLIM is reset to WFMAX, or held at its input value, whichever is smaller.

Third are the variables adjusted by PATSH during the optimization process. These data are accepted as input, if possible; but, if they are not geometrically acceptable, they are reset to appropriate limits and penalties are calculated.

TAUW1	Web thickness at Plane 1
LSA1	Star-point height above web at Plane 1
R5A1	Fillet radius between star point and web at Plane 1
R2A1	Corner radius at top of star point

The fourth and final category contains those dimensions that are calculated in SETUP2 for transmittal to the ballistic simulation module. They are based on all other dimensions.

LA	Length of star point (See Figure 43)
AL1A1	Included half-angle on star point (See Figure 43) at Plane 1

The first step in the dimension check is to test RFA1, THETA, and WO2MIN as shown above. If these tests are passed, WO2 is set equal to WO2MIN and TAUMAX (as defined in Figure 48) is found. Knowing TAUMAX, the maximum web fraction WFMAX is found from Eq (9) and compared to the input WFLIM. WFLIM is set equal to WFMAX if the latter is smaller than the input value. Next, web thickness TAUW1 is forced to be within the range

$$0 \leq TAUW1 \leq (WFLIM)(RFA1)$$

If PATSH has adjusted TAUW1 outside these limits, TAUW1 is reset to the limit and a penalty calculated.

The maximum possible corner radius R2A1 is TAUW1 in order to maintain the star point with the same web thickness as TAUW1. So the next test is

$$0 \leq R2A1 \leq TAUW1$$

If PATSH has adjusted R2A1 outside these limits, R2A1 is reset to the limit and a penalty calculated.

The fourth step is to solve for the length LA (shown in Figure 43).

$$LA = \frac{TAUW1 - R2A1}{\tan(THETA)} \tag{10}$$

Note that when R2A1 = TAUW1, LA = 0.

Now that R2A1 and TAUW1 are acceptable (or have been reset to acceptable values), WO2 is made equal to WO2MIN so that the maximum possible star point height (LSMAX) can be calculated (see Figure 49). The minimum possible star point height (LSMIN) is also calculated (see Figure 50); LSMIN occurs for a given TAUW1 when R5A1 and LC are zero. Then the incoming star point height (LSA1) is forced to be within the range

$$LSMIN \leq LSA1 \leq LSMAX$$

If PATSH has adjusted LSA1 outside these limits, LSA1 is reset to the limit and a penalty is calculated.

The final test is to assure that the fillet radius R5A1 is between maximum and minimum limits that are determined from the particular dimensional combination. Minimum R5A1 is zero; although small radii are critical, strains calculated in the propellant structural analysis will serve to limit the minimum fillet radius. Maximum R5A1 can be established by one of two situations. The center of the radius R5 must lie within the pie-shaped segment of symmetry or on its boundary. In the first situation (Figure 51), R5MAX occurs when the length LC equals zero and the center of R5 is within the segment of symmetry. In the second situation (Figure 52), the center of R5 is on the boundary of the segment of symmetry and LC $\neq$ 0. Whichever situation prevails, R5A1 is forced to lie within the range

$$0 \leq R5A1 \leq R5MAX$$

If PATSH has adjusted R5A1 outside these limits, R5A1 is reset to the limit and a penalty is calculated.

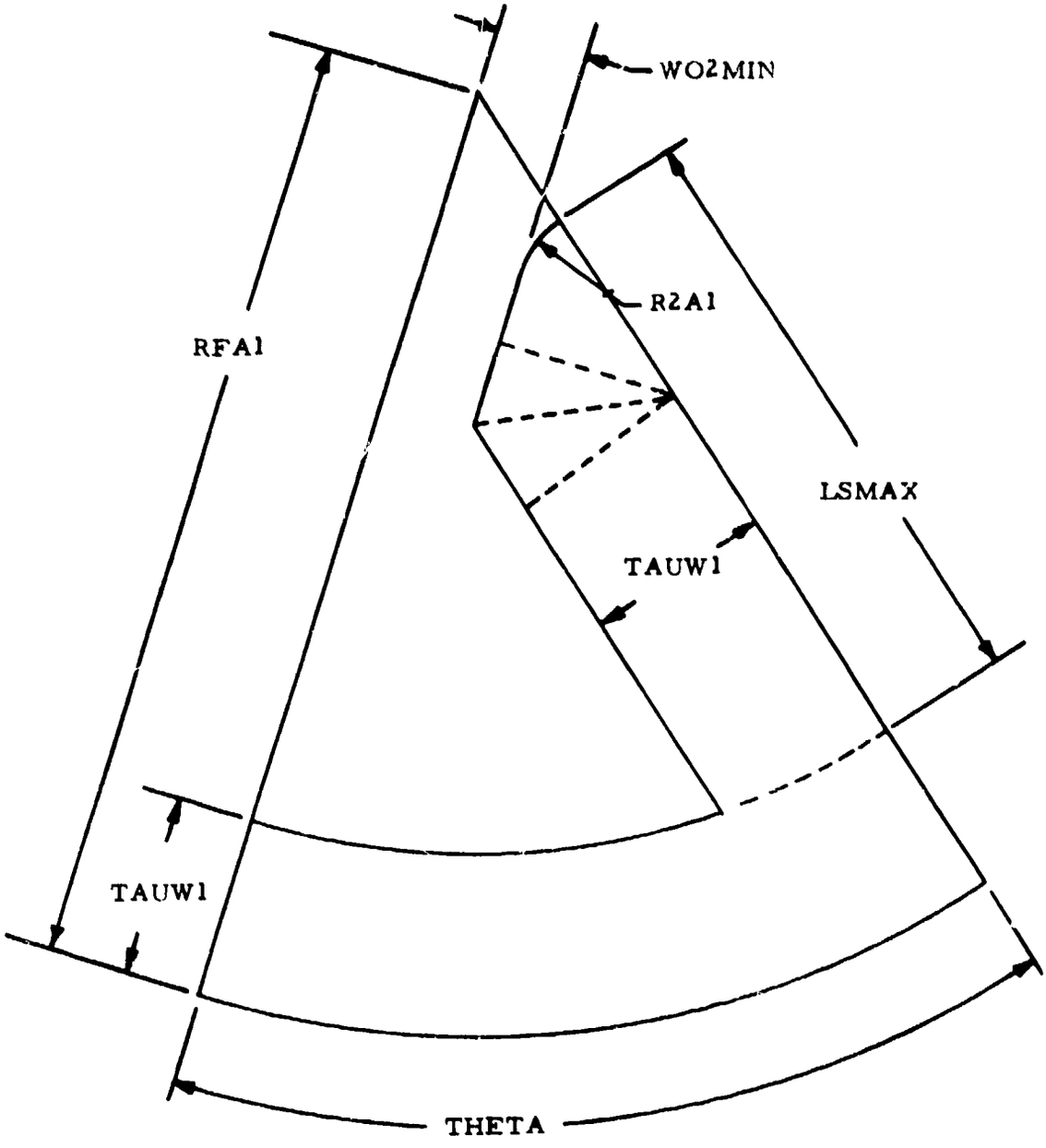


Figure 49. Maximum Star Point Height For Wagon Wheel (Type 2) Grain

R5A1 = 0.0
LC = 0.0

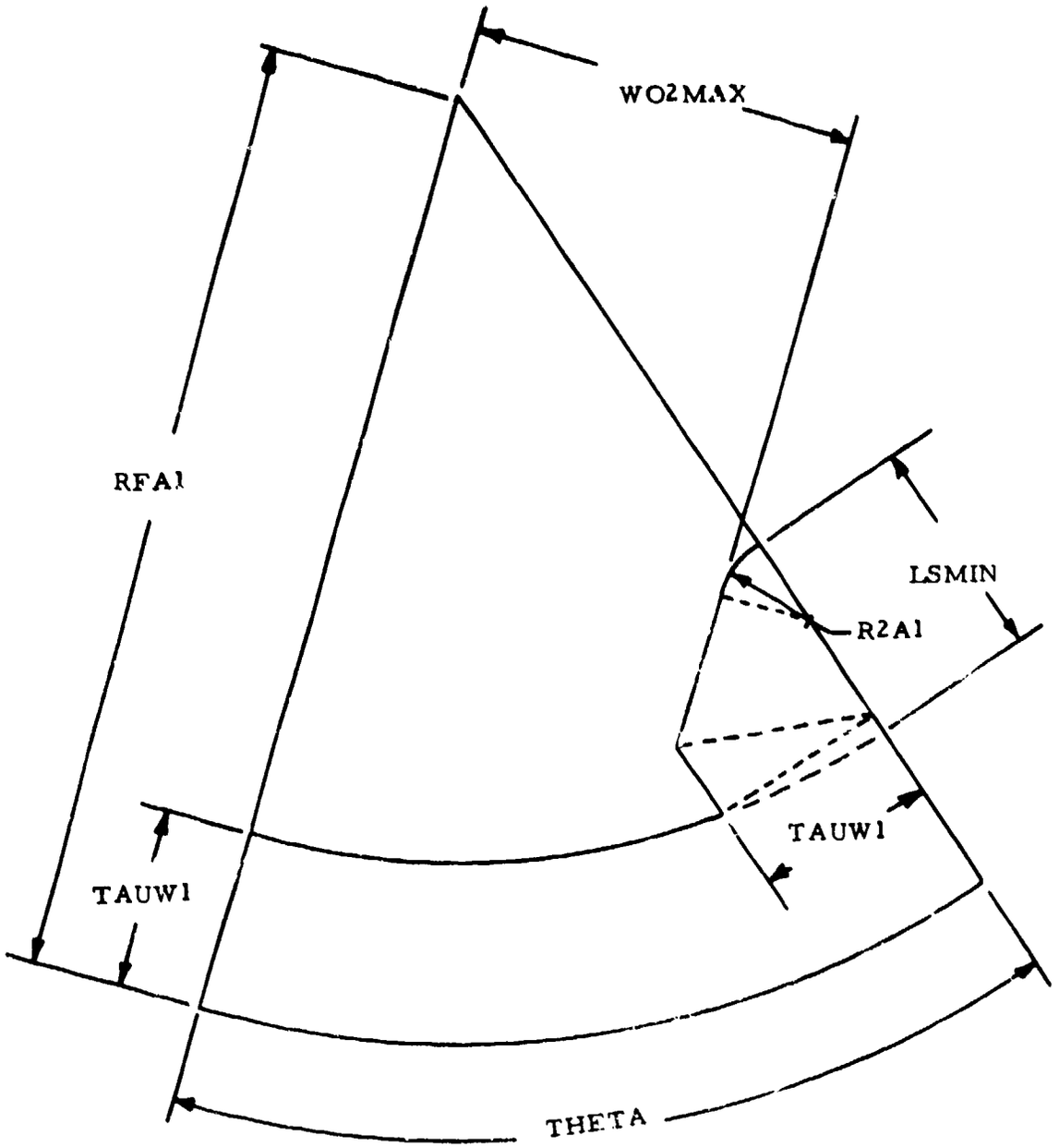


Figure 50. Minimum Star Point Height for Wagon Wheel (Type 2) Grain

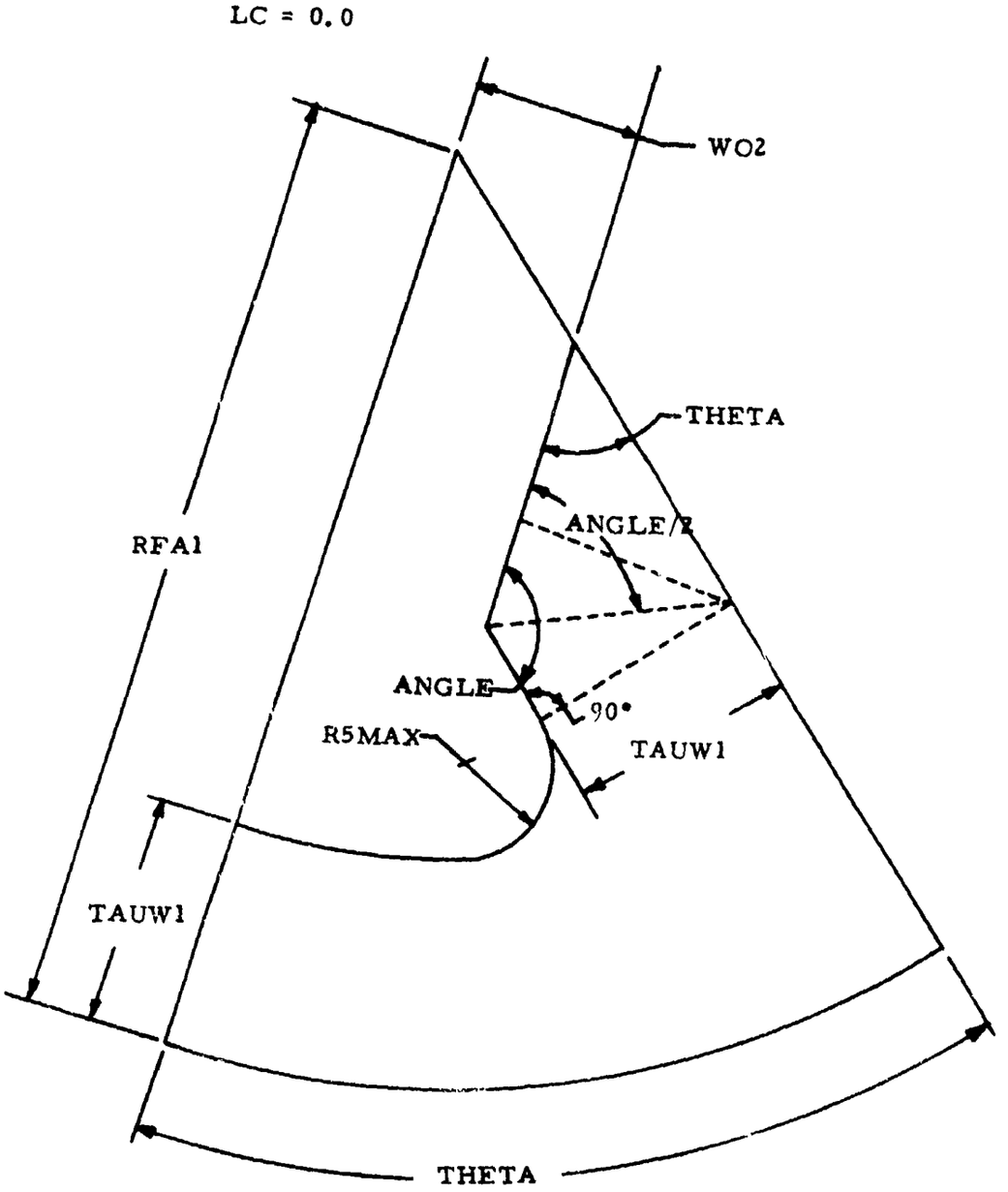


Figure 51. R5MAX Within Symmetry Segment (Type 2 Grain)

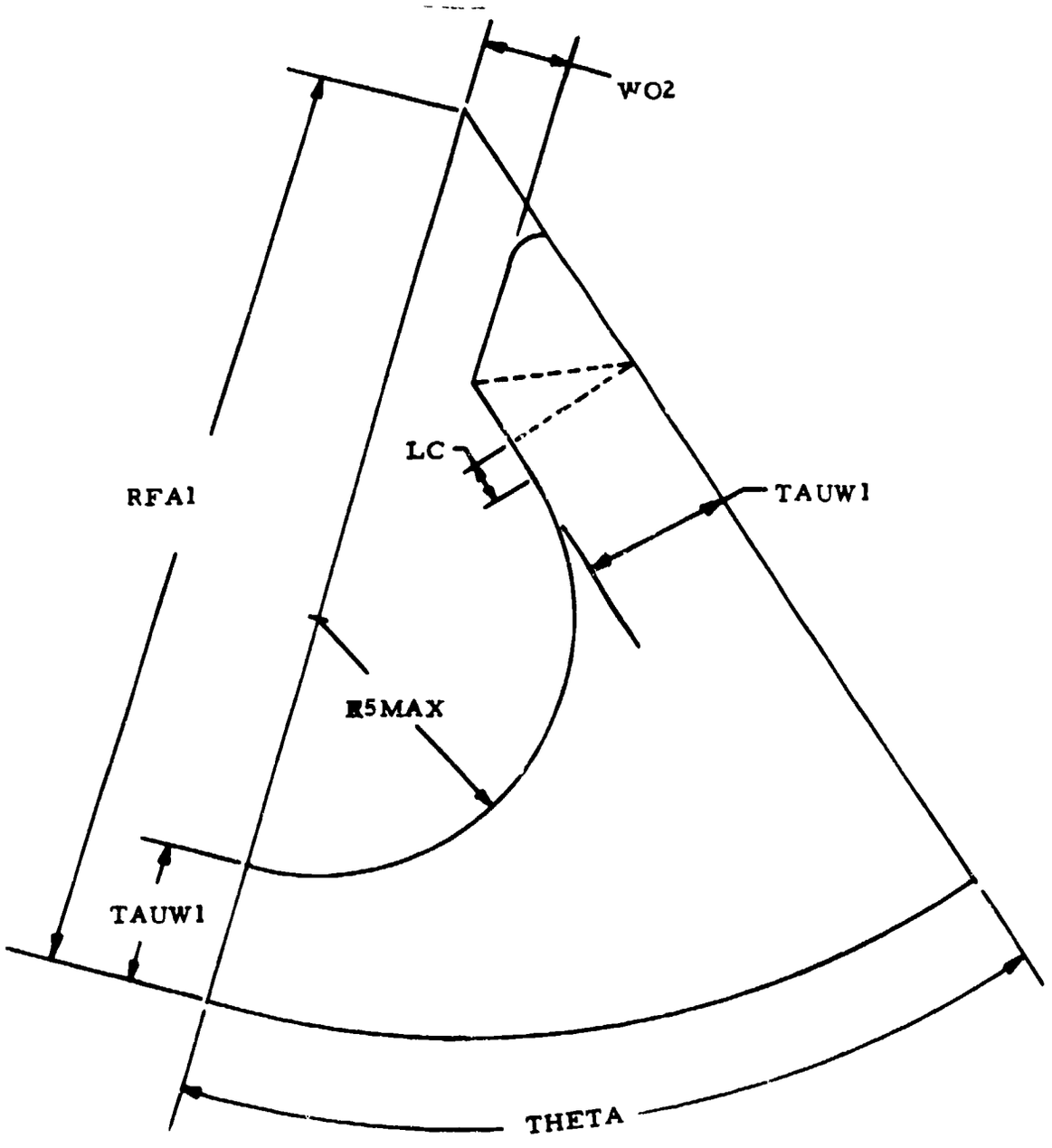


Figure 52. R5MAX On Boundary of Symmetry Segment (Type 2 Grain)

Now that all dimensions for Plane 1 are acceptable, star-point height at Plane 14 (LSA14) is tested to insure LSMIN ≤ LSA14 ≤ LSA1. If LSA14 is anything other than LSA1, the checks described above for LSA1 are performed again for LSA14, and R5A14 (different from R5A1) is calculated, if necessary.

The final tasks of subroutine SETUP2 are to insure that grain lengths are greater than zero, calculate inert weights, calculate data for use in propellant structural analysis, translate the grain dimensions into language required by the ballistic simulation module, and perform other miscellaneous tasks.

Type 3 (Finocyl) Grain

Type 3 grain is a finocyl configuration (Figure 53) with the longitudinal slots located in the forward end of the grain. Locations of direct input planes are shown in Figures 54 and 55 for the two closure types. Dimensions shown in blocks on Figure 53 are for Plane 1, which are held constant to Plane 4. Between Plane 4 and Plane 7, the slot depth radius R5 decreases until at Plane 7 it equals the bore radius R2. Slot fillet radii at Planes 5 and 6 (R4A5 and R4A6) are calculated internally to be tangent to the slot sides formed by angle ALPHA1. The circular port radius (R21) is held constant between Planes 7/8 and 10/11, and then it can expand at Plane 14 to control gas velocity.

The dimensions varied by PATSH during the optimization process are R5A1, R4A1, R2A1, and R2A14. Subroutine SETUP3 performs the validation checks. The logic flow for the cross-section dimension checking is shown in Figure 56.

A restriction on the minimum size of the bore radius R2A1 is a user-input of RIGN, that provides a clearance for the igniter and gas flow passage around it. The slot fillet radius R4A1 has a lower limit R4MIN also supplied by the user. An indirect restriction on the minimum R4A1 is the strain level calculated in the propellant structural analysis (which is then compared to an allowable strain). The user also supplies a MINWEB dimension which sets the maximum R4A1 for a given RFA1. The minimum R5A1 is R2A1 plus R4A1. The slot side angle ALPHA1 is constant at the user input value, but it is compared with ALPHMX for every incoming dimension set; if ALPHA1 > ALPHMX, it is set equal to ALPHMX for the current dimension set. It is then reset back to input ALPHA1 for the next dimension set.

Type 4 (Conocyl) Grain

Type 4 grain is a conocyl configuration with the transverse slot located in a Type 1 (ellipsoidal) forward closure (Figure 57). Locations of the direct input planes are given in Figures 58 and 59 for the two closure types. Dimensions shown in blocks on Figure 57 are input and result in a description of the conocyl grain.

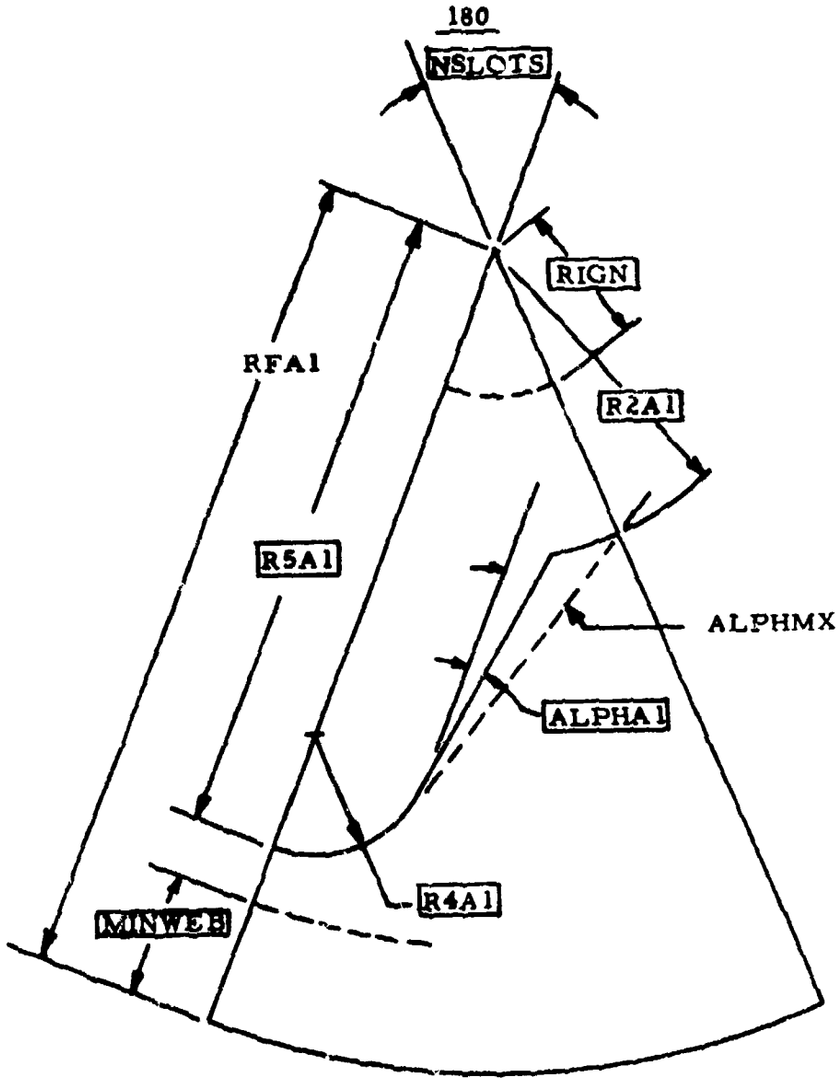
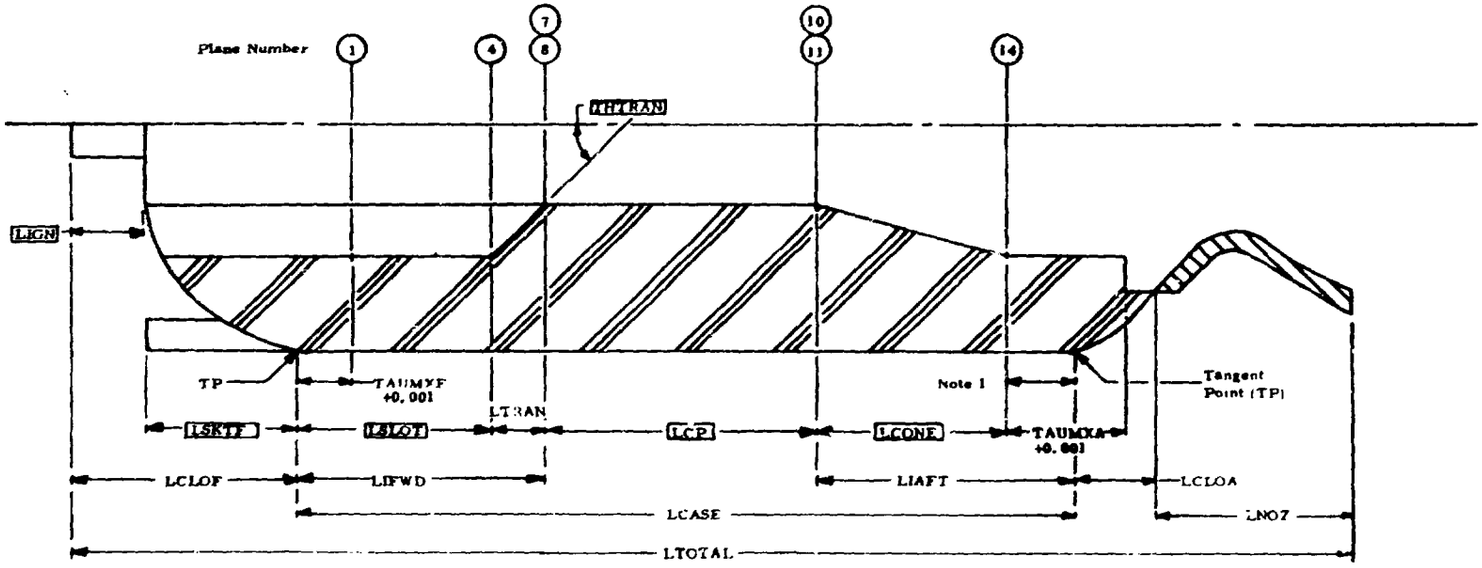


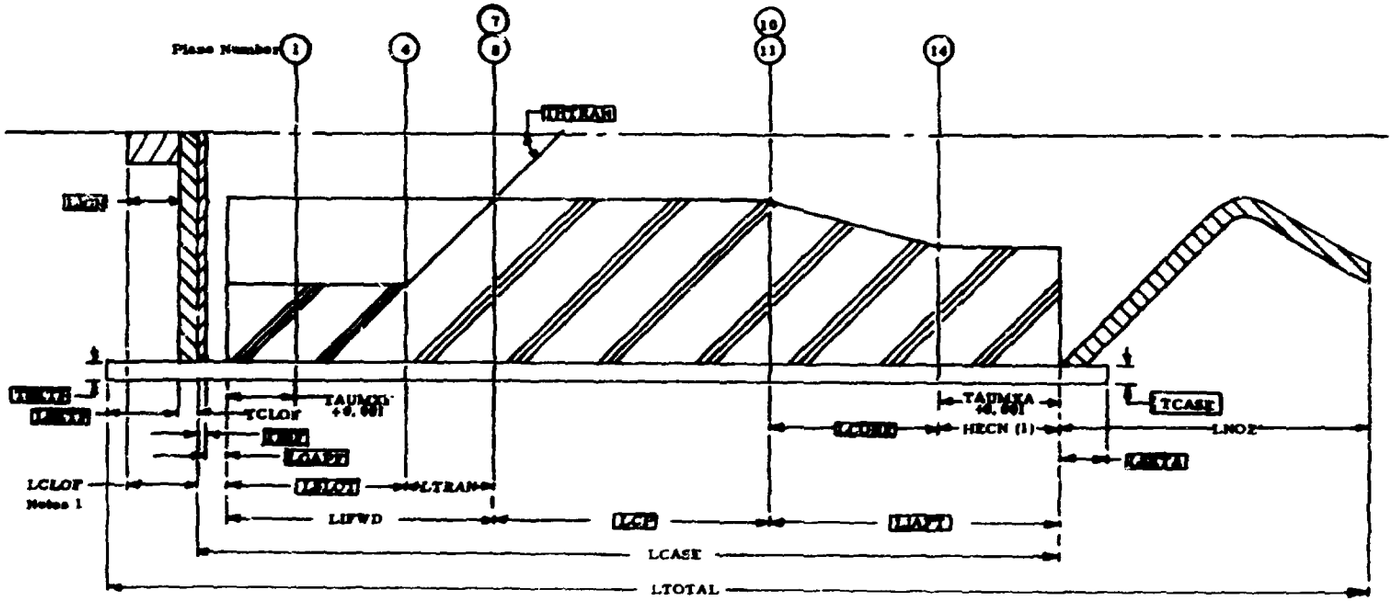
Figure 53. Type 3 (Finocyl) Grain Configuration



NOTES

(1) For Aft Closure Type 1, this dimension is (HECN(1)-X1A) if RFA14 < B5A, or is HECN (1) if RFA14 ≥ LCLOF
(2) LSKTF used to calculate LTOTAL if LSKTF > LCLOF
(3) Dimensions shown in blocks are input; others are output

Figure 54. Direct Input Planes for Type 3 Grain, Closure Type 1



NOTES

(1) LCLOF used to calculate LTOTAL if LCLOF > LSKTF
(2) Dimensions shown in blocks as input; others are output

Figure 55. Direct Input Planes for Grain Type 3, Closure Type 1

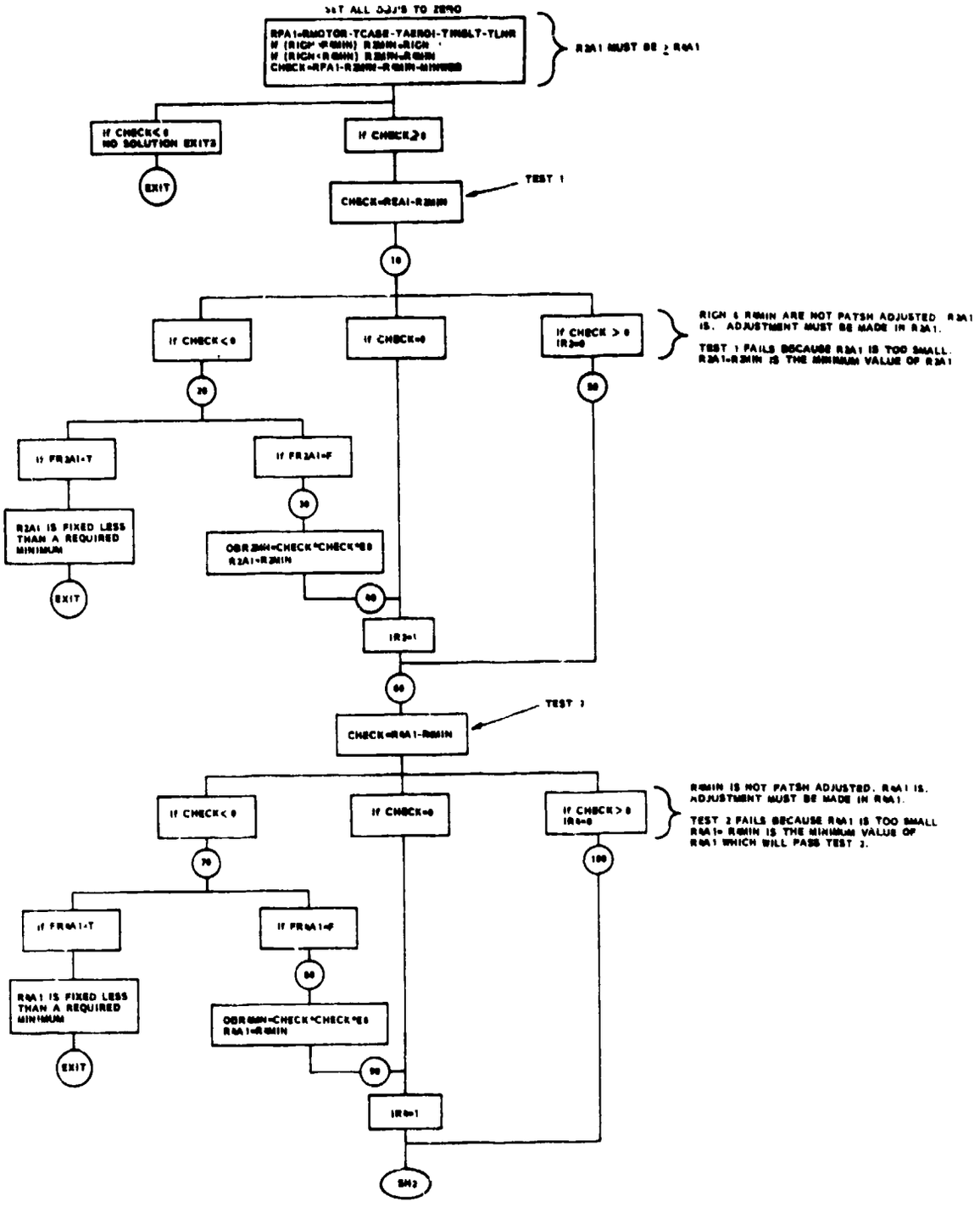


Figure 56. Logic Flow Diagram, Grain Type 3 (Part 1 of 6)

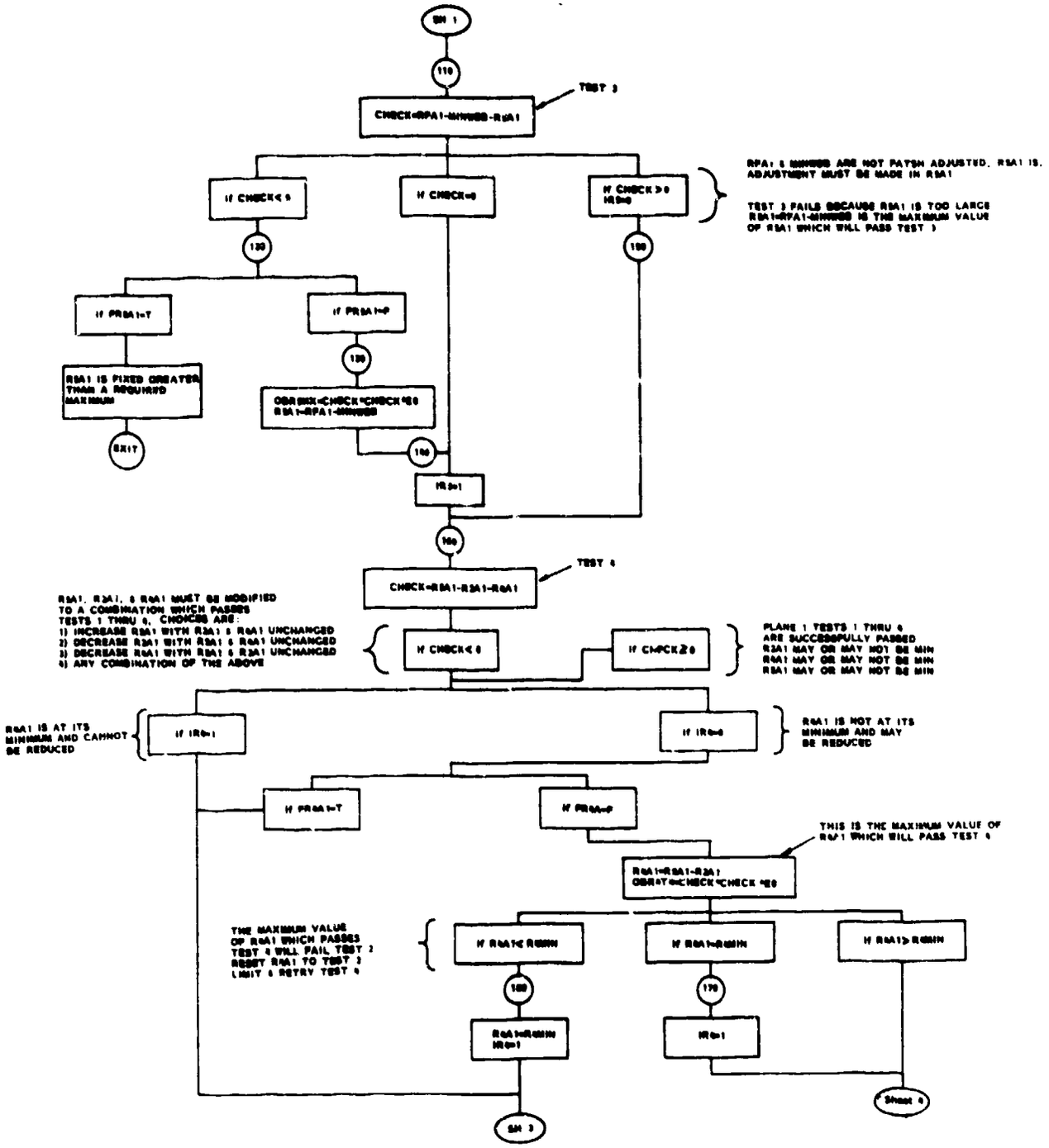


Figure 56. Continued (Part 2 of 6)

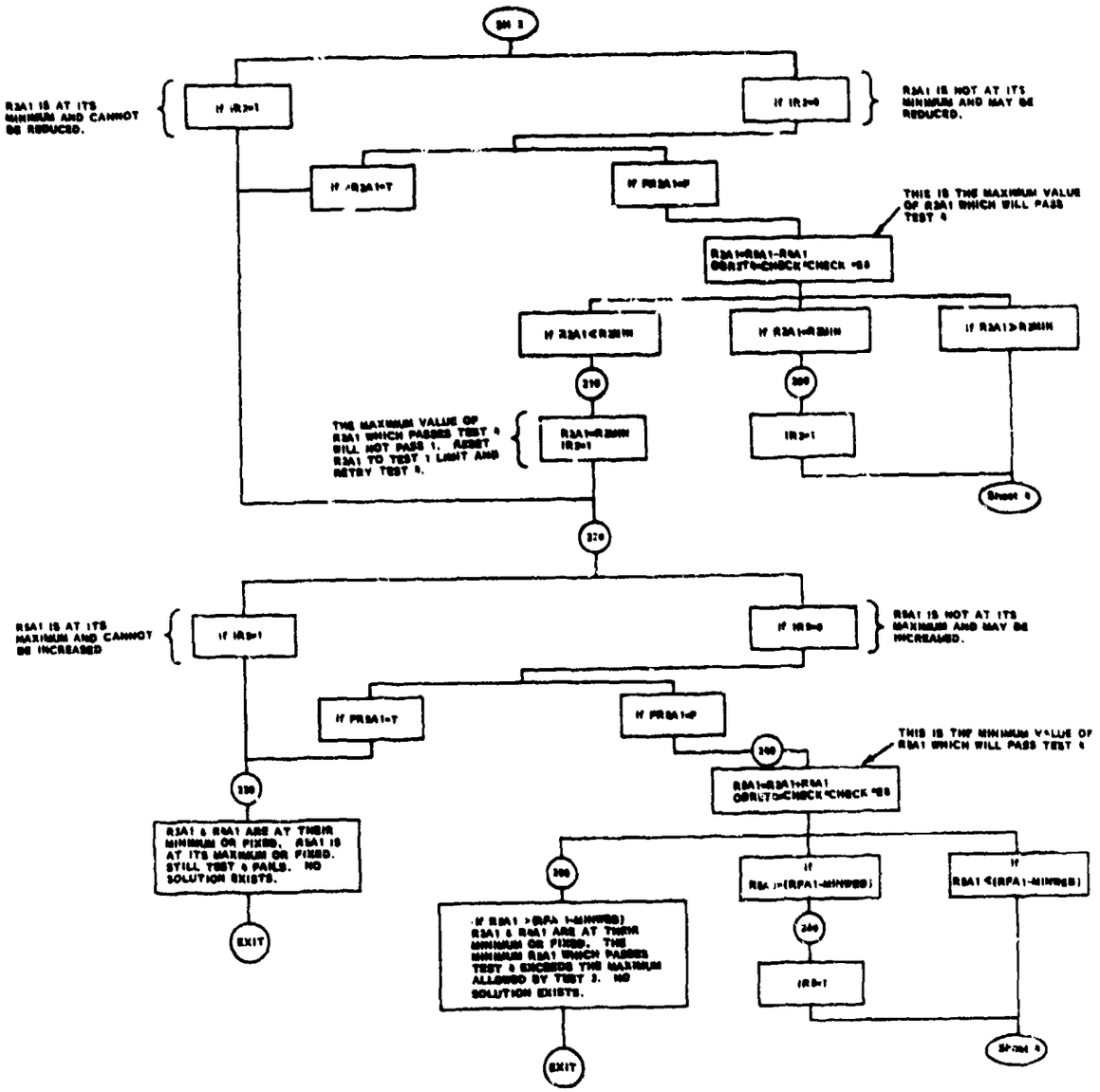


Figure 56. Continued (Part 3 of 6)

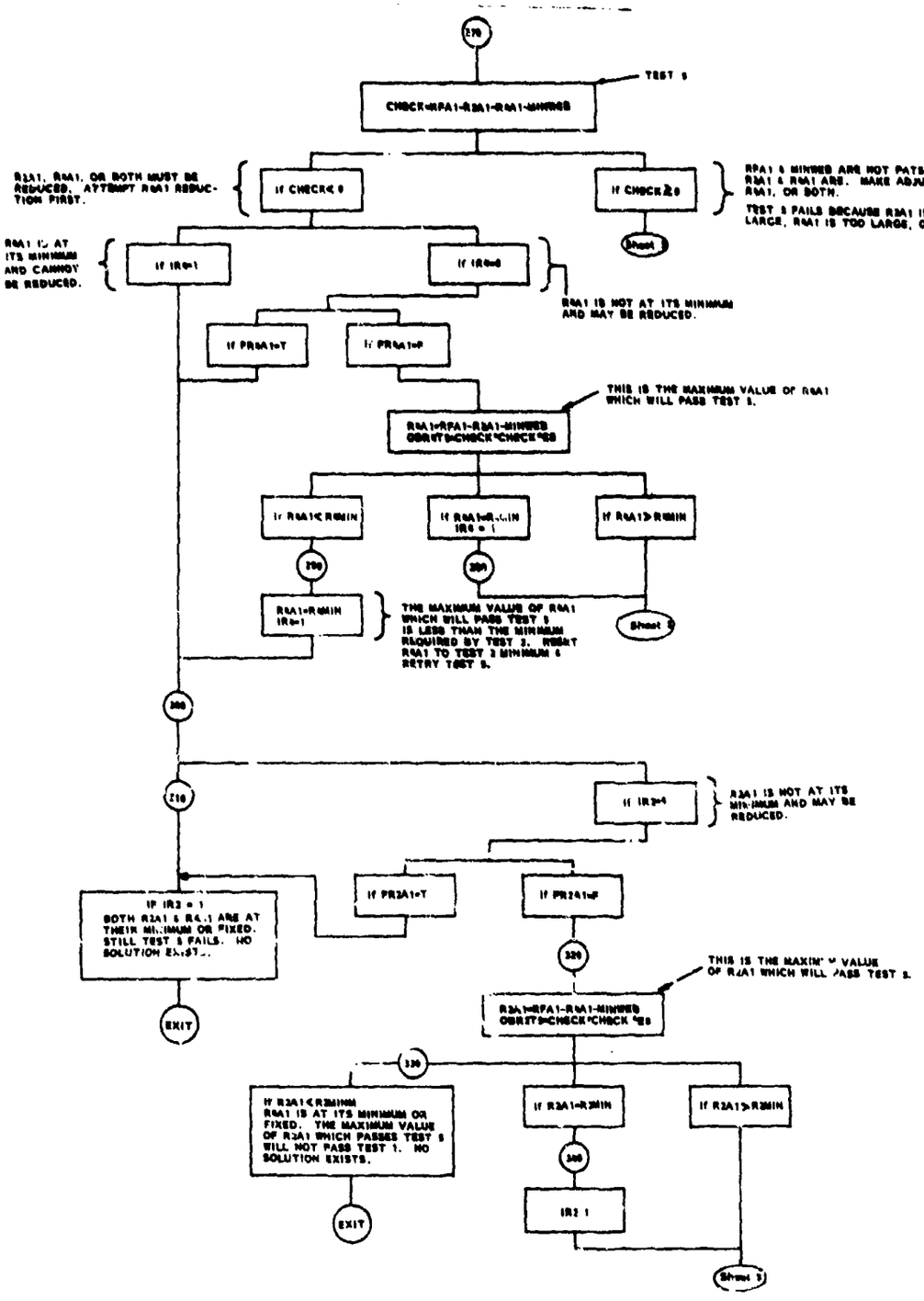


Figure 56. Continued (Part 4 of 6)

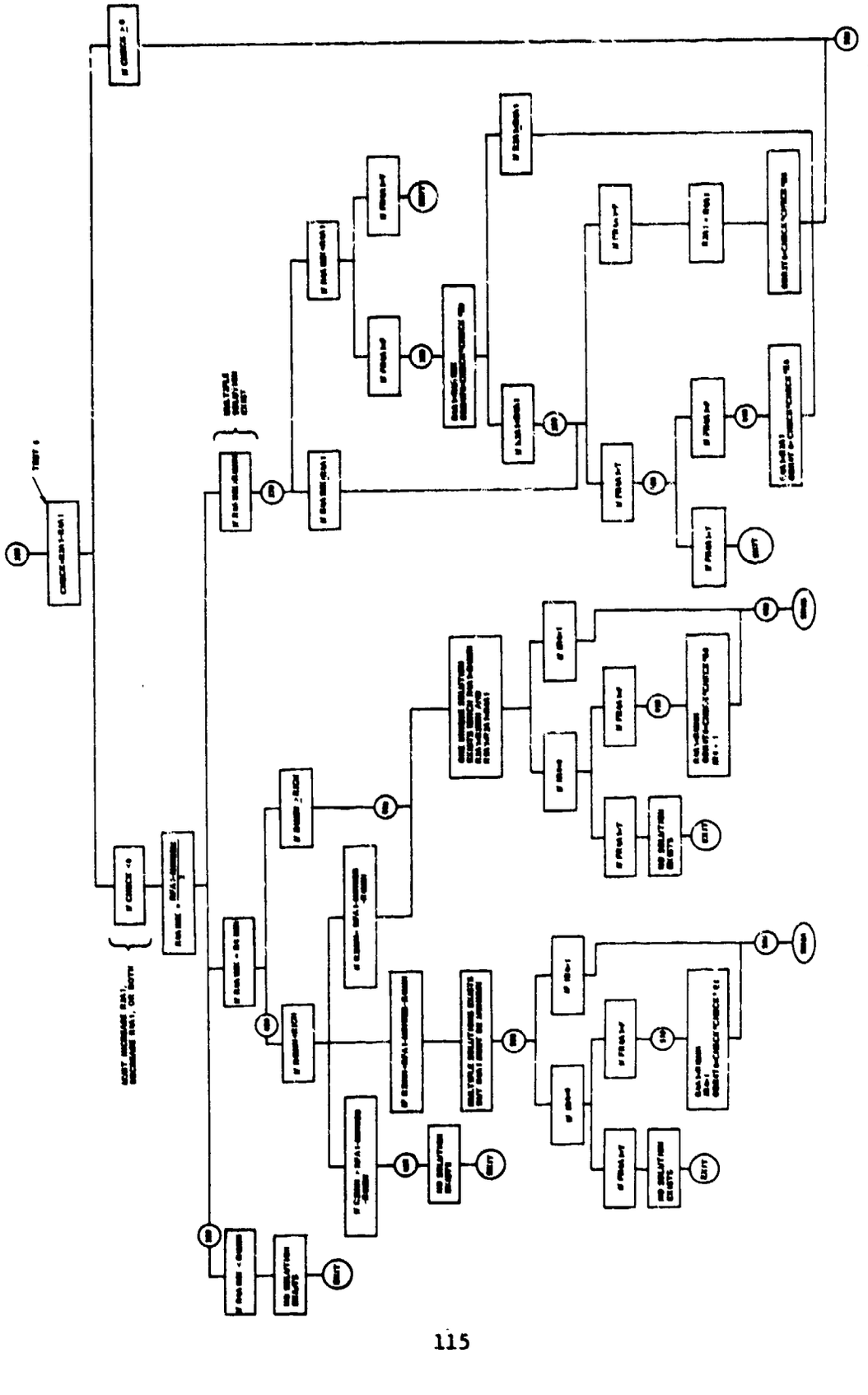


Figure 56. Continued (Part 5 of 6)

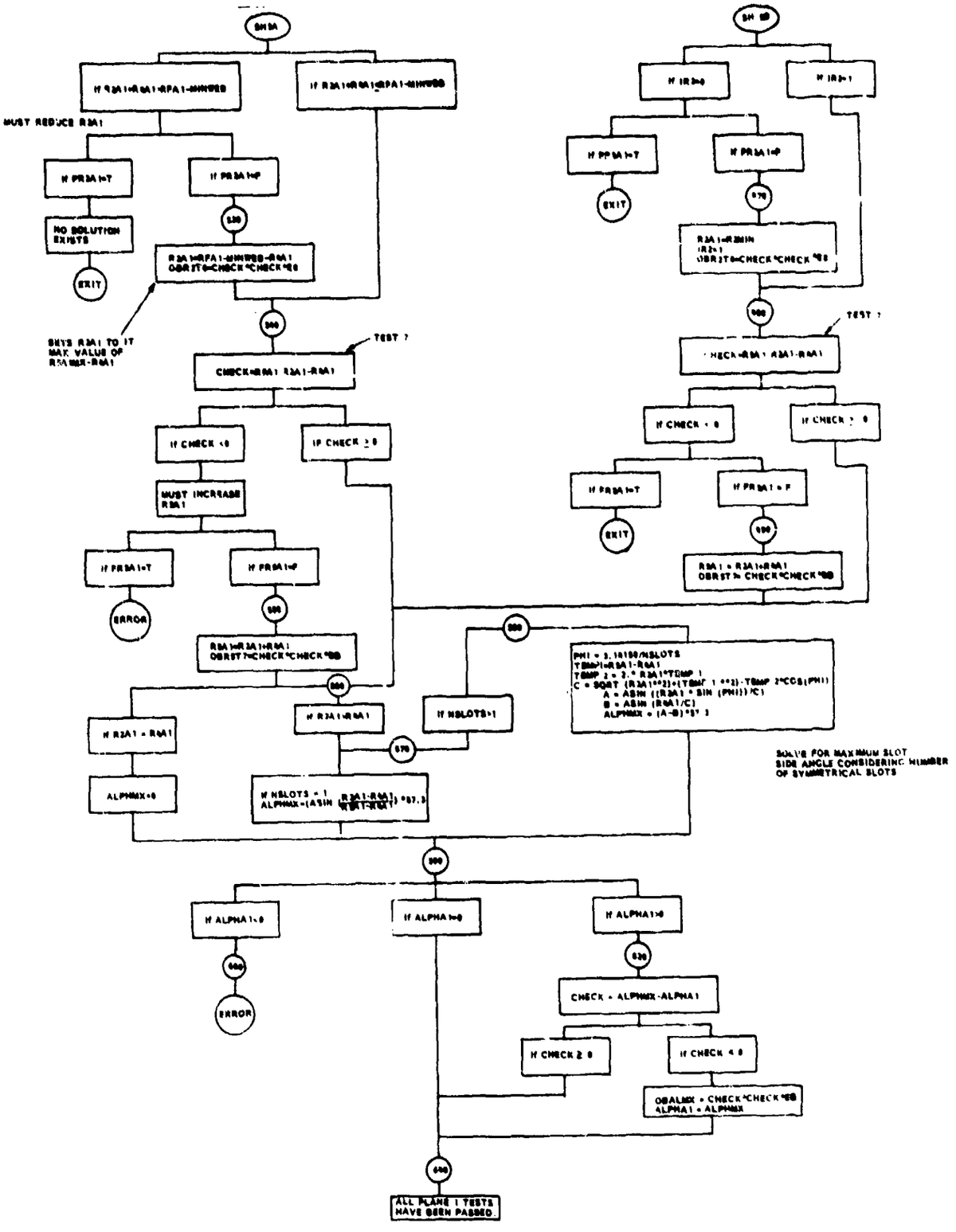
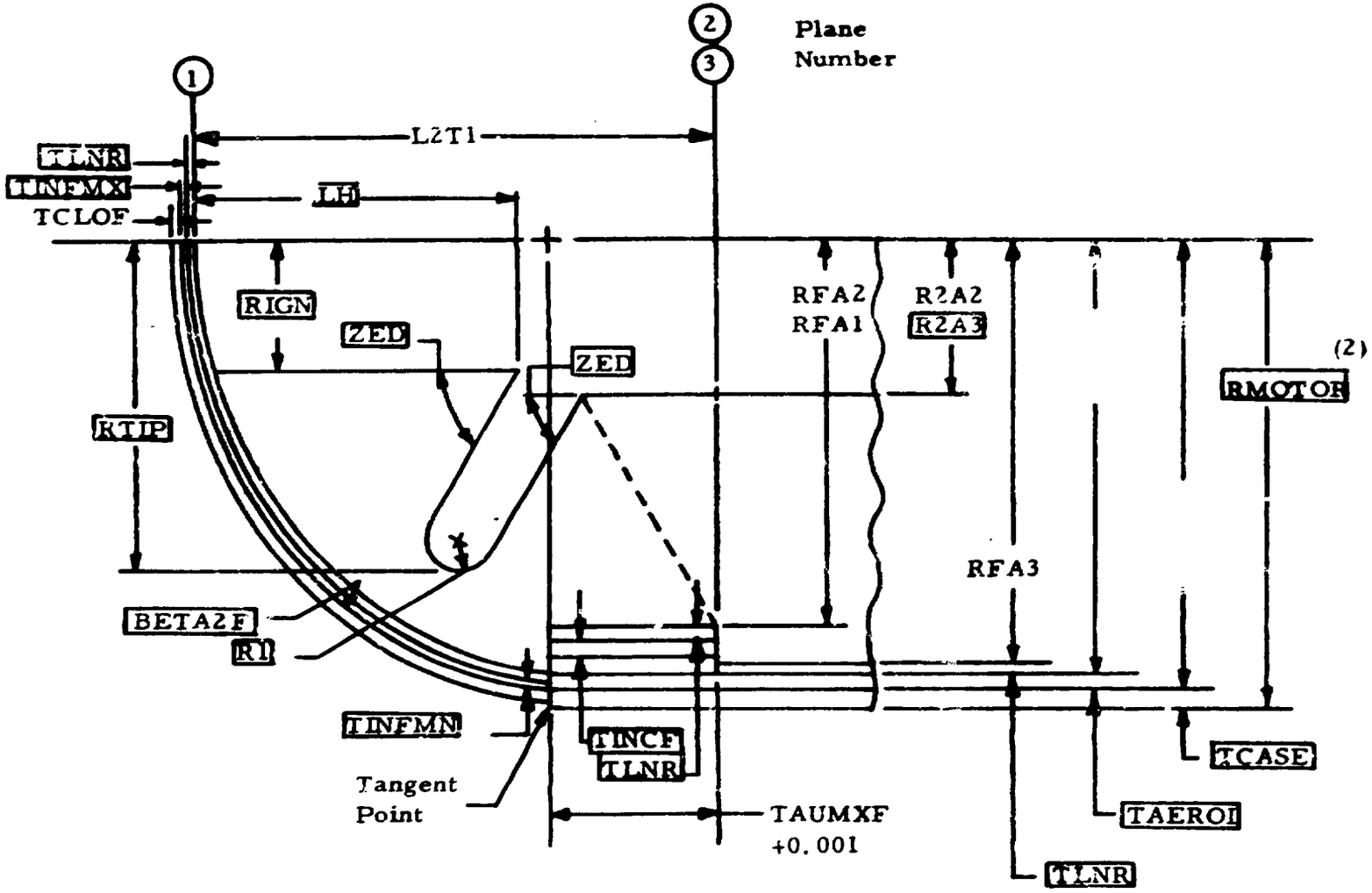
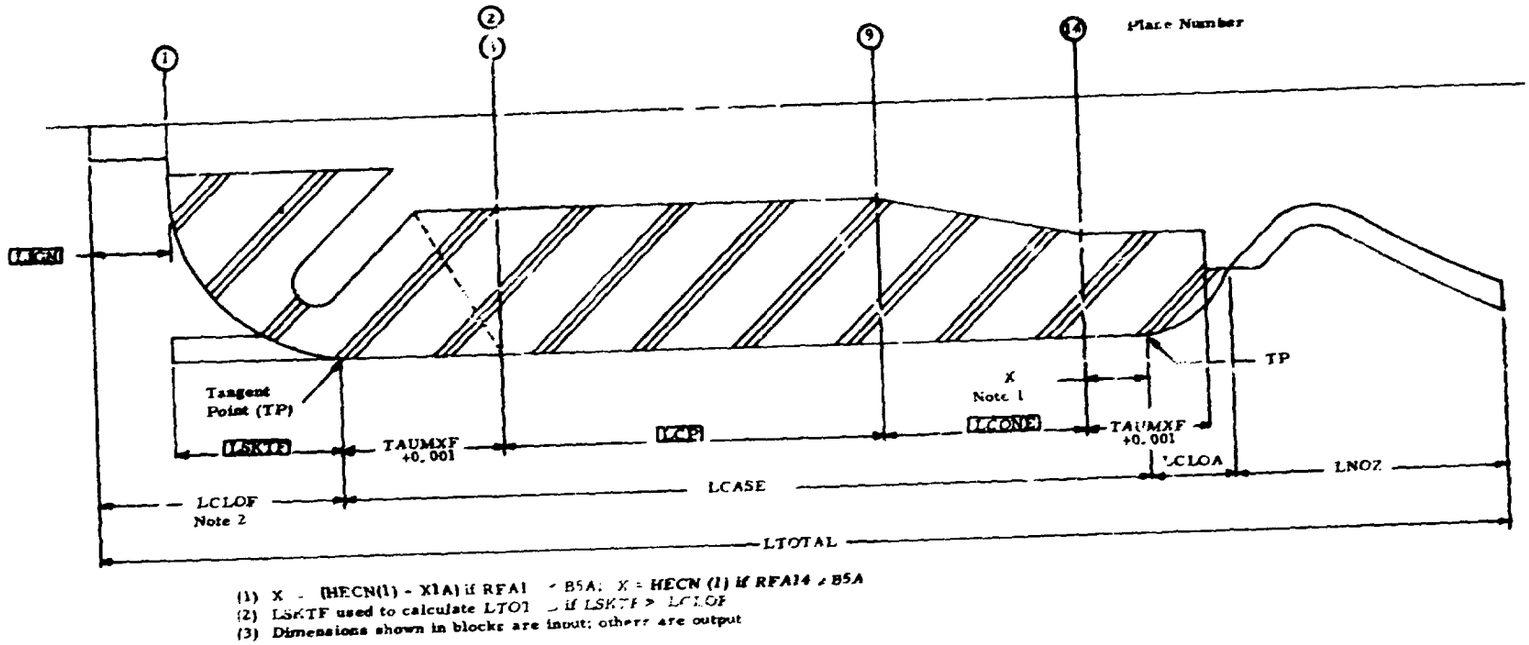


Figure 56. Continued (Part 6 of 6)



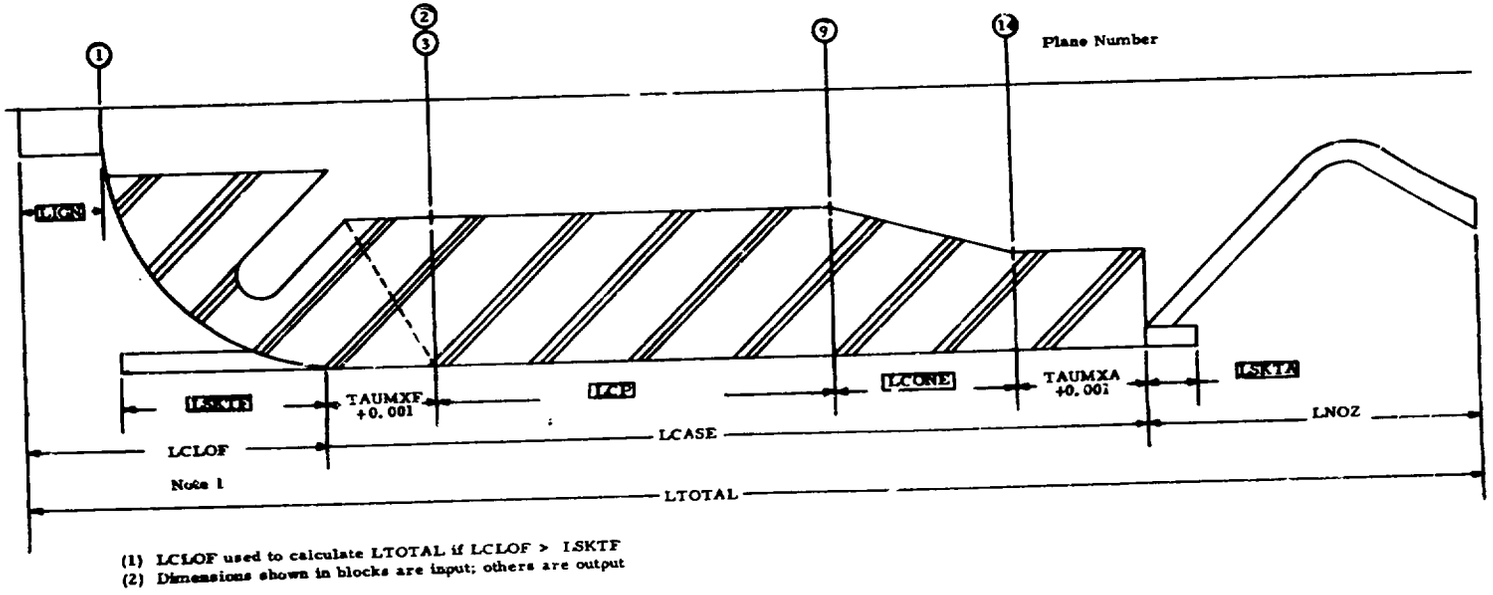
(1) Dimension shown in blocks are input; others are output
(2) Input as appropriate diameter

Figure 57. Nomenclature for Head End of Grain Type 4



(1) X = (HECN(1) - X1A) if RFA1 ≠ B5A; X = HECN (1) if RFA14 ≠ B5A
(2) LSKTF used to calculate LTO1 if LSKTF > LCLOF
(3) Dimensions shown in blocks are input; others are output

Figure 58. Direct Input Planes for Grain Type 4, Aft Closure Type 1



(1) LCLOF used to calculate LTOTAL if LCLOF > LSKTF
(2) Dimensions shown in blocks are input; others are output

Figure 59. Direct Input Planes for Grain Type 4, Aft Closure Type 2

A geometrically exact burn surface regression model for the forward portion of the grain produces a history of burning surface and port area (the latter at Plane 2/3) versus distance burned. This burning surface is converted to a perimeter versus distance burned (needed by ballistic simulation module) by dividing the surface area by the length L2T1, where L2T1 = TAUMXF + A5F, and A5F is the semi-minor axis of the liner inside surface (see Figure 12). The bore radius R2 is held constant to Plane 9, and then it can expand at Plane 14 to control gas velocity.

Subroutine SETUP4 performs the validation checks for the Type 4 grain. The logic flow for the forward segment checking is shown in Figure 60. These checks are illustrated on Figure 61. At the forward end, the slot tip is tested to assure it is located within the bounds shown in Figure 61, Part K, and that the slot angle ZED is $45^{\circ} < ZED < 90^{\circ}$ (Part E). Other checks are performed at the aft end so that R2A9<R2A14<RFA14 (Part F and Part H) and that the nozzle entrance radius RNOZEN is properly established (Part G). The other checks assure RIGN is dimensionally compatible with the remainder of the problem and that port radius R2A3 is within limits.

The final tasks of subroutine SETUP4 are to insure that grain lengths are greater than zero, calculate inert weights, translate the grain dimensions into language required by the ballistic simulation module, and perform other miscellaneous tasks.

Type 5 (CP) Grain

Type 5 grain has a cylindrical port (CP) for its entire length. Direct input planes are shown in Figures 62 and 63 for the two closure types. Cross-sectional dimensions adjusted in the optimization process are the port radii at Planes 1, 5 and 14 (R2A1, R2A5, R2A14, respectively). The dimensional checks are relatively simple.

$$R2A1 \geq R2A5$$
$$R2A14 \geq R2A5$$

Other checks determine that $R2MIN < R2 < RF$, where R2MIN is established by a web fraction limit.

Subroutine SETUP5 performs these verifications. As with the other grains, SETUP5 also insures that lengths are greater than zero, calculates inert weights associated with the cylindrical portion of the pressure vessel, translates the grain dimensions into language required by the ballistic simulation module, and performs other miscellaneous tasks.

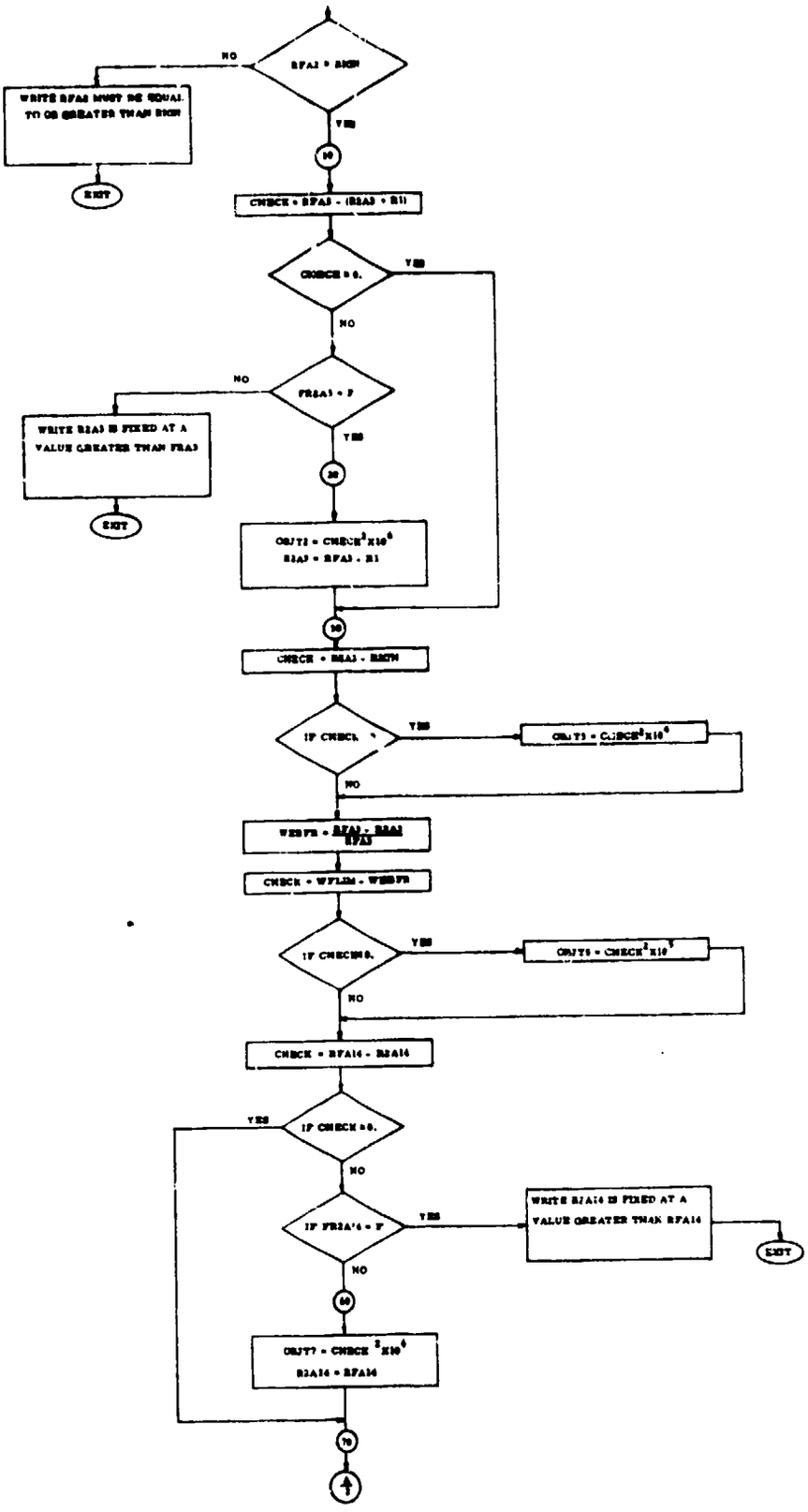


Figure 60. Logic Flow Chart for Type 4 (Conocyl) Grain

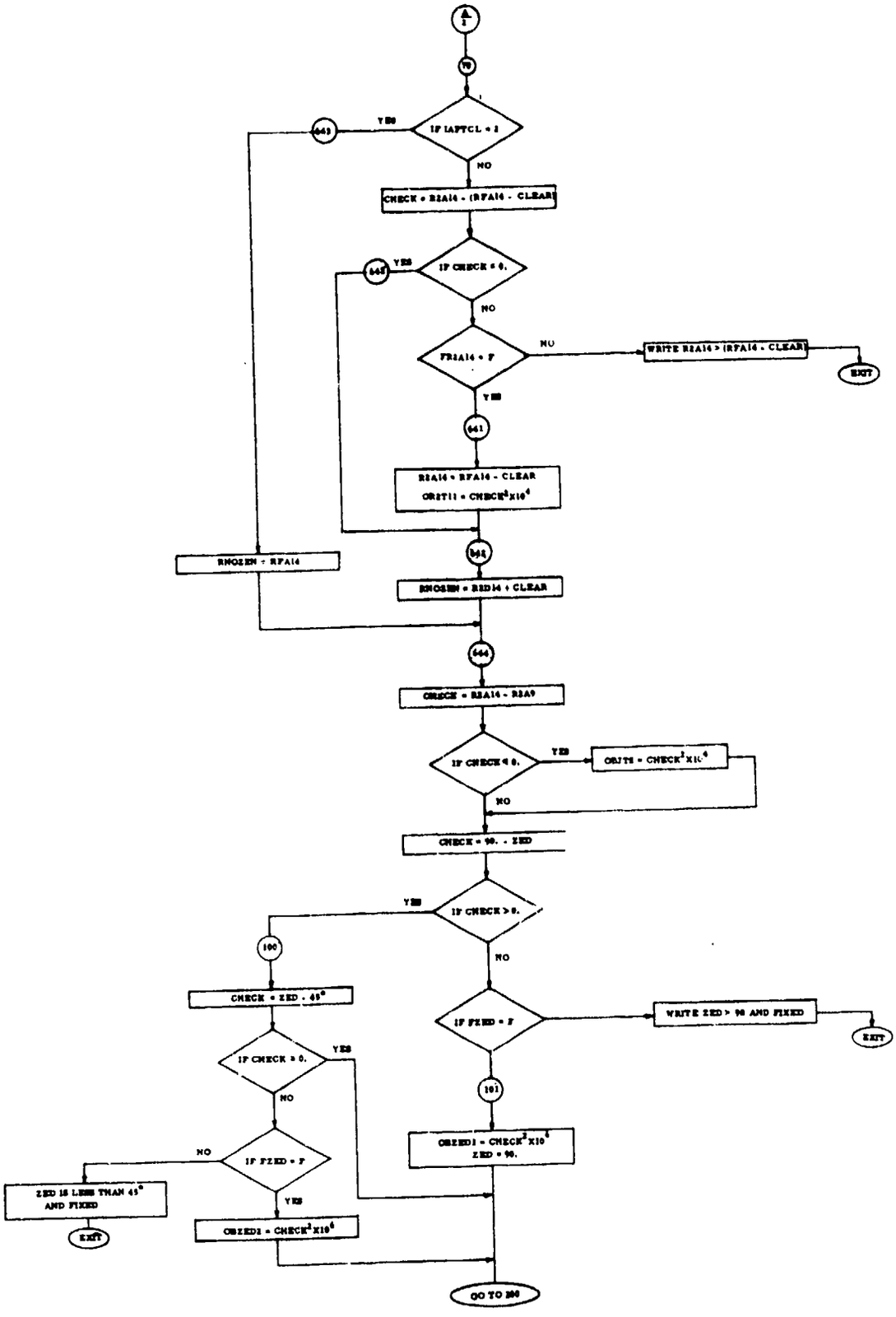


Figure 60 Continued (Part 2 of 4)

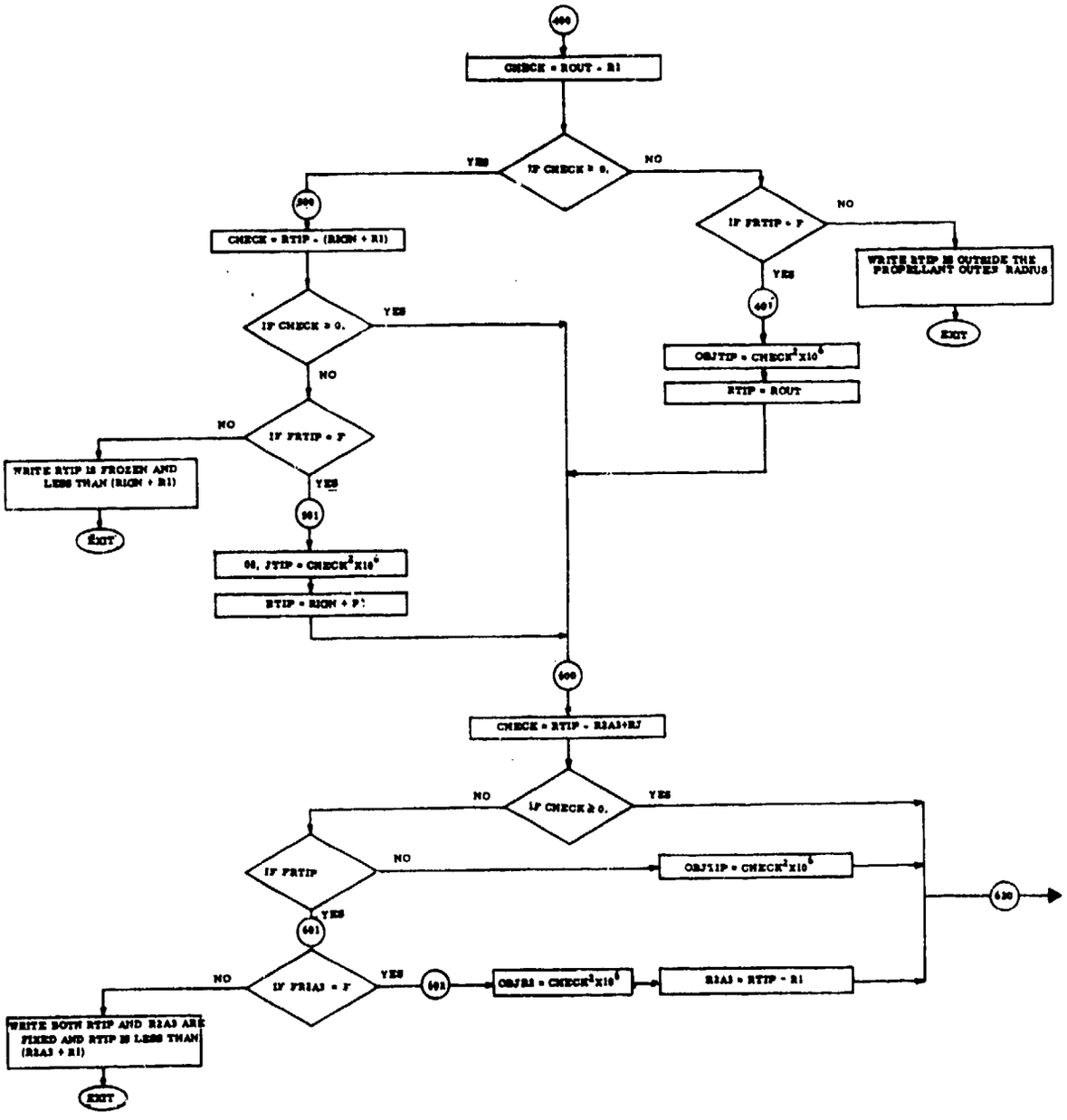


Figure 60 Continued (Part 3 of 4)

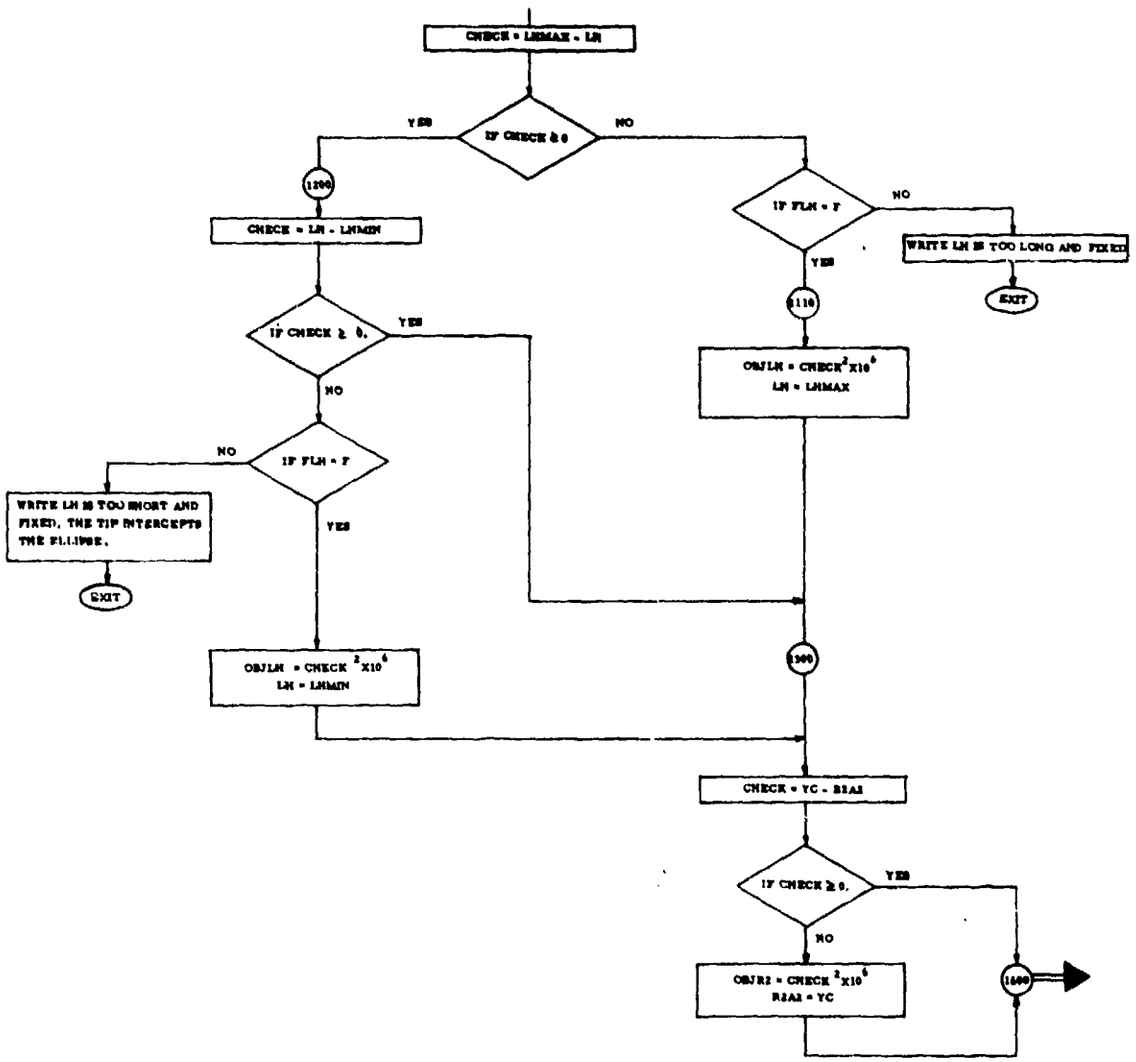


Figure 60 Continued (Part 4 of 4)

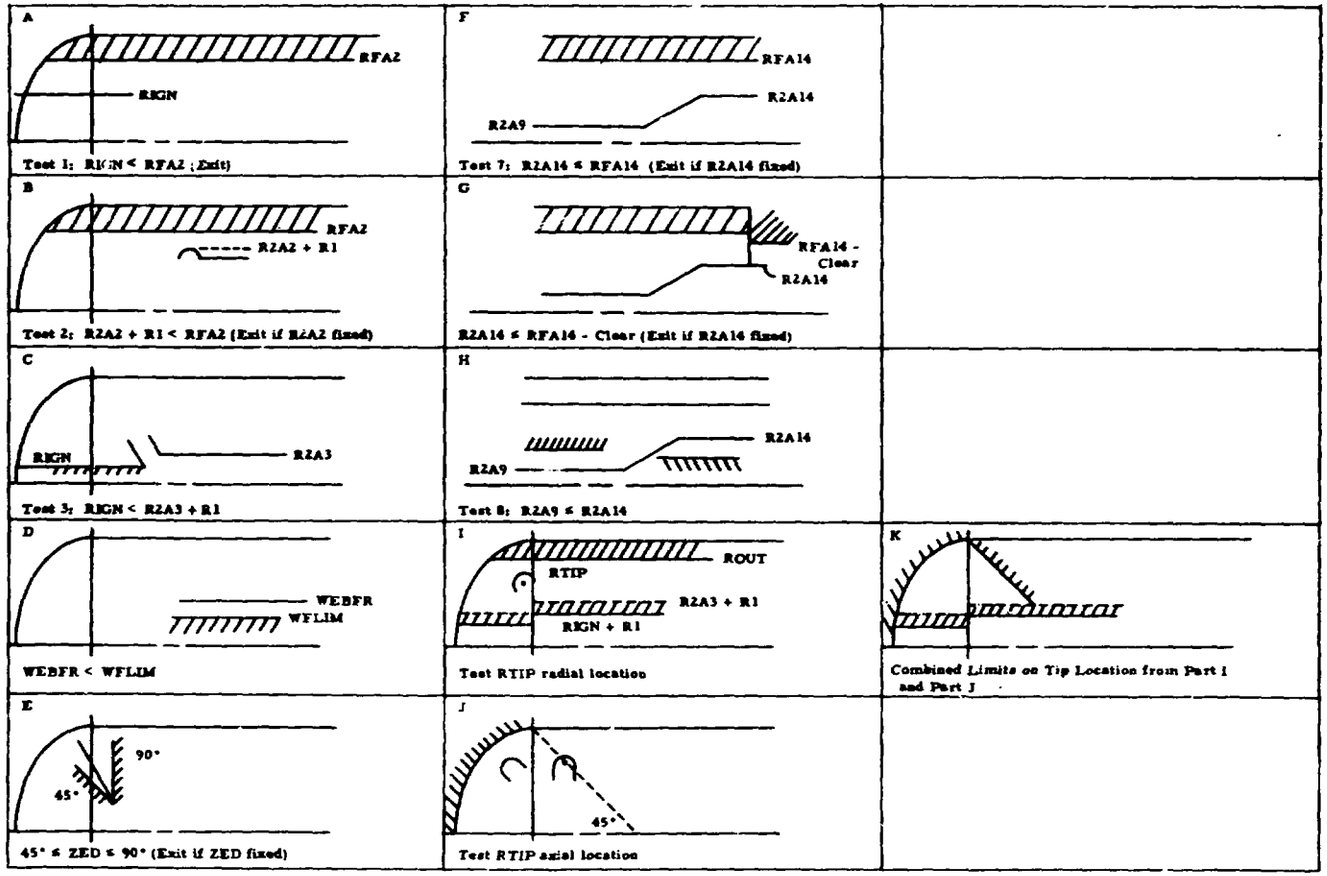
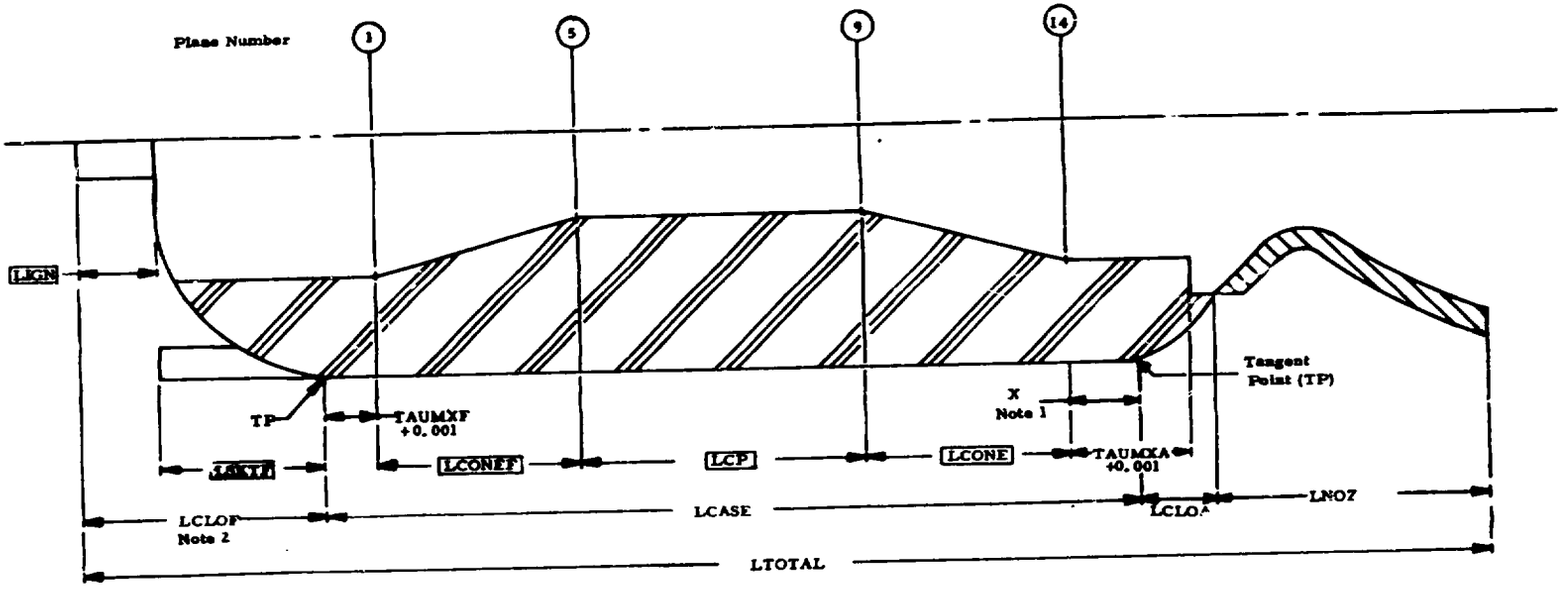


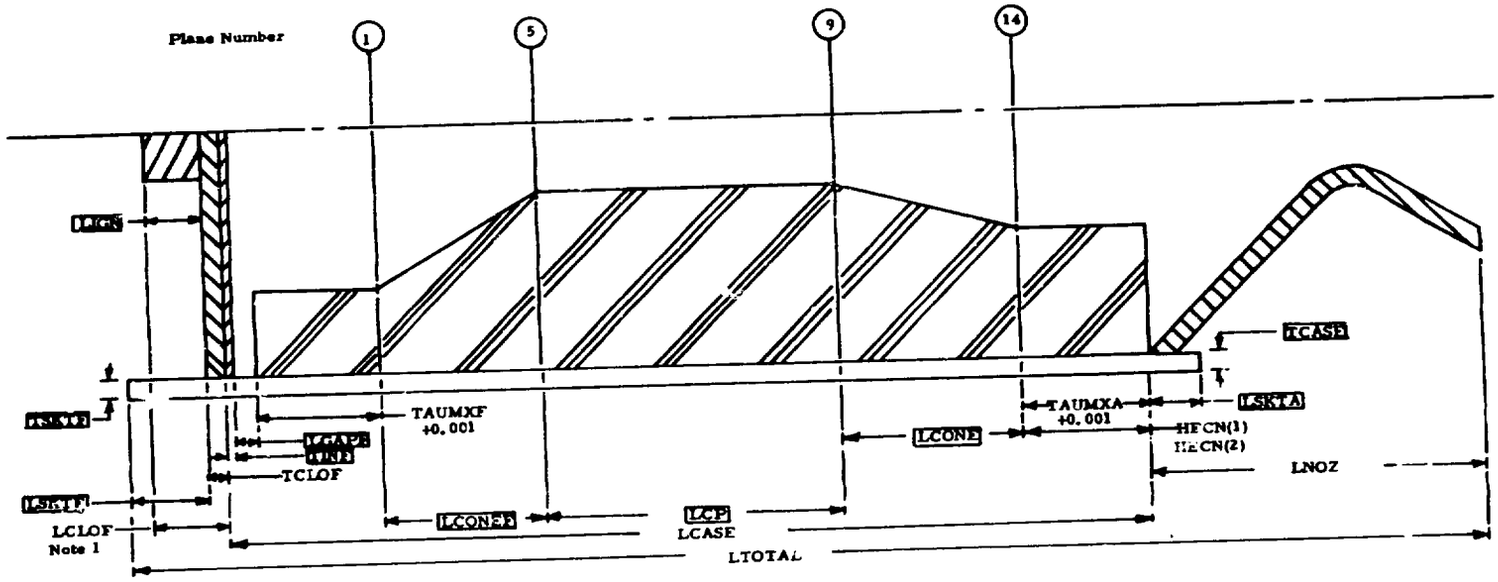
Figure 61. Dimensional Checks Performed on Type 4 (Conocyl) Grain



NOTES

(1) For Aft Closure Type 1, this dimension is X = (HECN(1)-X1A) if RFA14 ≠ B5A, or X = HECN(1) if RFA14≠B5A
(2) LSFTF used to calculate LTOTAL if LSKTF > LCLOF
(3) Dimensions shown in blocks are input: others are output

Figure 62. Direct Input Planes for Type 5 Grain, Closure Type 1



NOTE

(1) LCLOF used to calculate LTOTAL if LCLOF > LSKTF
(2) Dimensions shown in blocks are input; others are output

Figure 63. Direct Input Planes for Type 5 Grain, Closure Type 2

PROPELLANT

Propellant ballistic and gas dynamic properties can be supplied to SPOC by several means.

Option 1 allows evaluation of a single propellant whose characteristics are known. It would be employed when the user wants to design a motor that contains a specific propellant. The flag to set for Option 1 is PROPIN = T and two other flags (described below) are allowed to default to FORMAD = F and FORMIN = F. The user must furnish propellant ballistic characteristics:

CSTR70	Characteristic velocity (nozzle end) at 70°F (ft/sec)
DELP	Propellant cured density (lbm/cu in)
GAMAC	Ratio of specific heats in chamber
RGAS	Gas donstant of combustion products in chamber (ft-lbf/lbm-°R)
RB70	Propellant burn rate at 70°F, 1000 psia (in/sec)
XN	Pressure exponent in the rate model RATE = A*P** N

If the problem requires ballistic simulation at two grain temperatures, additional user-supplied inputs are:

PIK	Temperature coefficient of pressure (per °F)
MC	Temperature coefficient of characteristic velocity (per °F)

If the SPP impulse efficiency model (Reference 12) is specified by the user (SPPETA = T), the user must supply additional information for this propellant option:

IVAC	Vacuum specific impulse, shifting equilibrium at motor pressure and expansion ratio (lbm-sec/lbm)
IVACF	Vacuum specific impulse, frozen equilibrium at motor pressure and expansion ratio (lbf-sec/lbm)
MOLCND	Mole fraction of condensed species (moles per 100 gms of mixture)

If a combustion stability analysis is to be performed (FSTAB = T), the user must supply more information for this option:

SONVEL	Sonic velocity in chamber, stagnation conditions (ft/sec)

Option 2 will permit the user to supply the formulation of a given propellant and SPOC will calculate much of the information listed above. This option is also employed when the user wants to design a motor that contains a specific propellant, as did Option 1. The flag to set for Option 2 is FORMIN = T, and two other flags are allowed to default to FORMAD = F (described below) and PROPIN = F. In this option, the thermochemistry module (TCHEM), is called only on the first pass through COMP and these results are used throughout the optimization process. Chamber pressure and nozzle expansion ratio are needed in TCHEM; on the first pass through COMP the chamber pressure that is used is an input value, PC (defaults to 1000 psia), and nozzle expansion ratio is derived from initial nozzle dimensions. Data that must still be supplied by the user are RB70, XN, PIK, and MC. All other data are produced by the thermochemistry module, provided another condition is satisfied: IVAC, IVACF and MOLCND are not calculated unless SPPETA = T.

Option 3 is employed when the propellant ingredient weight fractions are adjusted as part of the optimization process. The flag to set for Option 3 is FORMAD = T, and two other flags are allowed to default to FORMIN = F and PROPIN = F. In this option, the thermochemistry module (TCHEM) is entered on every pass through COMP (subject to the conditions discussed below) to provide the data listed above. Chamber pressure and nozzle expansion ratio are needed in TCHEM. On the first pass through COMP, the chamber pressure that is used is an input value, PC, and nozzle expansion ratio is derived from the initial nozzle dimensions. On subsequent passes through COMP, the chamber pressure is the average pressure from the preceeding simulation, and the expansion ratio is still based on the current nozzle dimension set; if the problem being solved calls for ballistic simulation at two temperatures, the average pressure used is from the high-temperature simulation. As the optimum design is approached, the difference between the pressures from the current simulation and the just completed simulation will disappear. The thermochemistry module is not called for a new analysis unless the chamber pressure or the nozzle expansion ratio on the last pass through COMP is at least 5% different from that on the next-to-last pass or the ingredient weight fractions have been adjusted.

TCHEM is the executive subroutine for the entire thermochemistry module. It calls subroutines LIQUID and NORMAL (described below) and then MAINCO, which is the executive subroutine for the thermochemistry analysis itself (once a valid formulation has been provided to it).

Theoretical density calculated by the thermochemical analysis routine is based on ingredient density and relative amounts. This value is multiplied by 0.985 to obtain a "cured" density to account for polymerization and cool-down from cure temperature.

Propellant Formulation

The propellant formulation is adjusted as part of the optimization process through changes in weight fractions of related ingredients. Two steps are required in this process. First, the current formulation must be

checked against user-supplied and physical limits and then normalized so that the total weight fraction is 1.0. Second, this verified data set is sent to the thermochemistry subroutine to have the various properties calculated.

The first task performed by subroutine TCHEM is (through subroutine LIQUID) to verify that the incoming formulation is physically possible and that it adheres to all limits (Figure 64). In this effort, it checks the compatibility of liquid constituents and total solids content. For example, if all liquid ingredients have their weight fractions fixed by the user at input values, that sets the total solids level; if that exceeds a user-supplied maximum total solids content, a basic incompatibility exist; and subroutine LIQUID so informs the user.

The second task performed by subroutine TCHEM is (through subroutine NORMAL) to normalize the incoming propellant ingredient weight fractions so that the total is 1.0 (Figure 65). One or more weight fractions can be changed by PATSH during the optimization process. Subroutine NORMAL takes these quantities, (knowing the ingredients to be adjusted by PATSH and the limits calculated by subroutine LIQUID) to specify a formulation whose weight fractions total 1.0 and that are within user- or internal-generated limits. It calculates penalties when incoming weight fractions must be changed to conform to limits. Only those ingredients being adjusted by PATSH are changed in subroutine NORMAL to normalize the formulation. The normalization process is an iterative one (Figure 65).

Within the thermochemistry analysis module itself is another normalization routine that will adjust all ingredient weight fractions to total 1.0, regardless of t. : intent of the user. This situation should not be encountered unless the formulation is input with the weight fraction total not equal to 1.0 and flags set to hold all ingredients at their input weight fractions (FORMIN = T). Then, the second normalization process will change all amounts so that the total is equal to 1.0.

The next task performed by subroutine TCHEM is to set up the propellant ingredient data prior to calling MAINCO (the entry to the thermochemical analysis). Up to this point, oxidizer has been considered as up to two independent materials with up to three independent particle sizes of each; now, these are combined into total amounts of each of the two oxidizers.

Thermochemistry

Propellant ballistic and gas dynamic characteristics are calculated internally in SPOC (when so specified) using a version of the NASA-Lewis code TRAN72 (Reference 01). In its original form, this code calculates thermodynamic and transport properties of complex mixtures, chemical equilibrium for assigned thermodynamic states, and theoretical rocket performance for both frozen and equilibrium compositions during expansion for condensed and gaseous species. Ingredient and specie thermodynamic and transport properties are obtained from JANNAF tables that are periodically

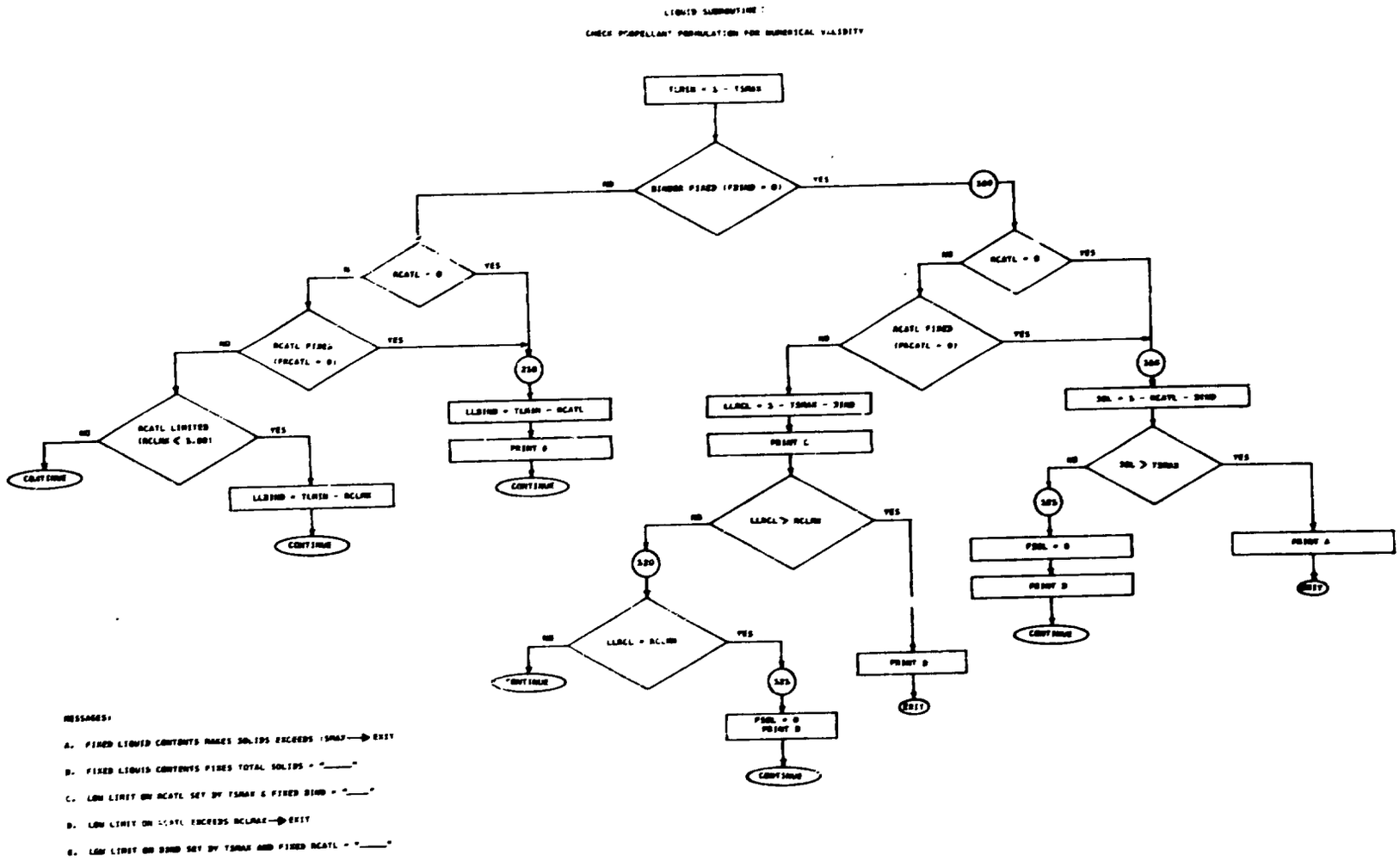


Figure 64. Logic Flow Chart for Subroutine LIQUID

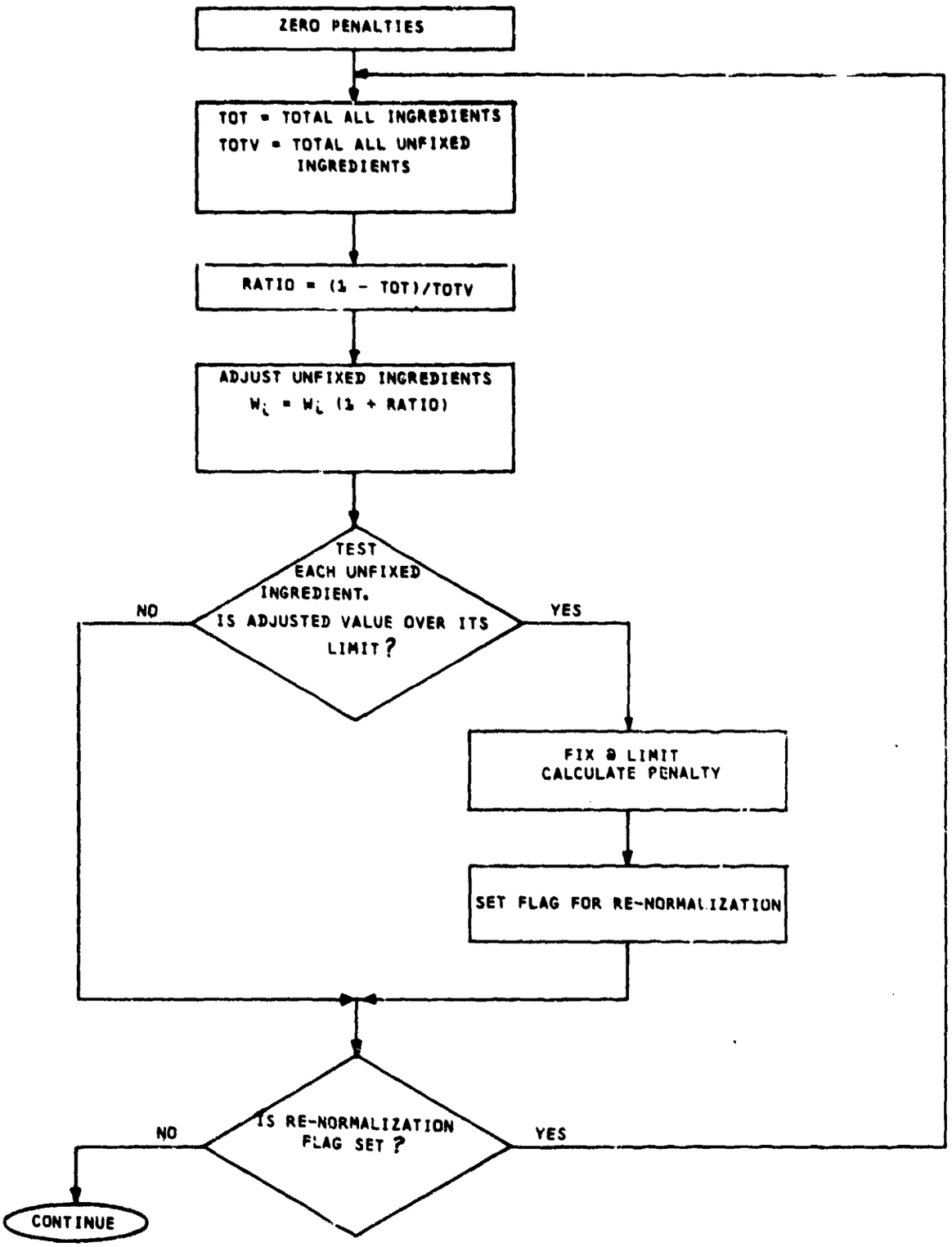


Figure 65. Logic Flow Chart for Subroutine NORMAL

updated. Execution time was reduced by streamlining the code so that it performs only those calculations needed by SPOC, produces no printout, and considers only those combustion products that will be produced by a limited group of propellant constituents. This latter consideration is one of the more productive changes in reducing execution time, because it greatly reduces the time needed to search the product table.

In addition, SPOC stores all the appropriate thermodynamic data internal to the code, rather than on external tapes as done in the original NASA code. Propellant ingredients stored in the code are:

Ingredient	Ingredient Name	Choice of Ingredients
Binder	BIND	HTPB -- Mixture consisting of HTPB polymer and typical cure agent, plasticizer and bond agent
Fuel	FUEL	C -- Carbon ZR -- Zirconium AL -- Aluminum
Oxidizer	OXA, OXB	AP -- Ammonium perchlorate HMX -- Cyclotetramethylene-tetranitramine RDX -- Cyclotrimethylene-trinitramine
Rate Catalyst (Solid)	RCATS	Fe_2O -- Iron Oxide FCH -- Ferrocene
Rate Catalyst (Liquid)	RCATL	None available at the present
Combustion Stabilizer	STAB	ZRC -- Zirconium carbide ALOX -- Aluminum oxide ZR -- Zirconium C -- Carbon

The pertinent properties for HTPB binder are:

Formula: C(7.133) H(11.150) O(0.135) N(0.067)
Enthalpy (cal/gram formula weight): -6047.0
Density (gm/cu cm): 0.903

Input Information

Manipulations within the code to adjust, verify and normalize the propellant formulation use generic nomenclature so that the user can have a choice of ingredients for any one constituent class. For example, two oxidizers may be employed (identified as OXA and OXB) and either one of

them may be "named" as AP, or RDX, or HMX. Furthermore, each of the oxidizers may be indexed to designate that they have up to three different particle sizes, to use, for example, in propellant burn rate models that combine amount and size. Particle size can be input for the AP oxidizer (DIAAP(I), I = 1, 2, 3) for use in the combustion stability module.

Up to four other ingredients may be defined by the user to the code to provide additional propellant formulation combinations. The user must furnish the chemical formula, enthalpy, and density, along with their identifying name. Then, this name can be employed, along with those already stored in the code, to describe the propellant. Table 3 lists the species which can be employed in the chemical formula of a new ingredient.

TABLE 3

SPECIES CONTAINED IN THERMOCHEMICAL ANALYSIS MODULE

FORMULA	TEMP RANGE (K)		PHASE
AL1	300.	6000.	G
AL1	932.	4000.	L
AL1	300.	932.	S
AL1 CL1	300.	6000.	G
AL1 CL1 O 1	300.	6000.	G
AL1 CL2	300.	6000.	G
AL1 CL3	300.	6000.	G
AL1 CL3	466.	1500.	L
AL1 H 1	300.	6000.	G
AL1 H 1 O 1	300.	6000.	G
AL1 N 1	300.	6000.	G
AL1 O 1	300.	6000.	G
AL1 O 1 H 1	300.	6000.	G
AL1 O 2	300.	6000.	G
AL2 O 1	300.	6000.	G
AL2 O 2	300.	6000.	G
AL2 O 3	2327.	4000.	L
AL2 O 3	300.	2327.	S
C 1	300.	6000.	G
C 1	300.	6000.	S
C 1 CL1 O 1	300.	6000.	G
C 1 CL4	300.	6000.	G
C 1 H 1 N 1	300.	6000.	G
C 1 H 1 N 1 O 1	300.	6000.	G
C 1 H 1 O 1	300.	6000.	G
C 1 H 2	300.	6000.	G
C 1 H 2 O 1	300.	6000.	G
C 1 H 3	300.	6000.	G
C 1 H 3 CL1	300.	6000.	G
C 1 H 4	300.	6000.	G
C 1 N 1	300.	6000.	G
C 1 N 1 O 1	300.	6000.	G
C 1 O 1	300.	6000.	G
C 1 O 2	300.	6000.	G
C 1 ZR1	300.	3805.	S
C 2	300.	6000.	G
C 2 H 1	300.	6000.	G
C 2 H 1 CL1	300.	6000.	G
C 2 H 2	300.	6000.	G
C 2 H 4	300.	6000.	G
C 2 N 2	300.	6000.	G
CL1	300.	6000.	G
CL1 FE1	300.	6000.	G
CL1 H 1	300.	6000.	G
CL1 H 1 O 1	300.	6000.	G
CL1 N 1 O 1	300.	6000.	G
CL1 O 1	300.	6000.	G
CL1 ZR1	300.	6000.	G
CL2	300.	6000.	G
CL2 FE1	300.	6000.	G

Table 3

Species Contained in Thermochemical Analysis Module (Cont'd)

FORMULA	TEMP RANGE (K)		PHASE
CL2 FE1	950.	3000.	L
CL2 FE1	300.	950.	S
CL2 ZR1	300.	6000.	G
CL2 ZR1	1000.	2000.	L
CL2 ZR1	300.	1000.	S
CL3 FE1	300.	6000.	G
CL3 FE1	577.	1500.	L
CL3 ZR1	300.	6000.	G
CL3 ZR1	300.	2000.	S
CL4 FE2	300.	6000.	G
CL4 ZR1	300.	6000.	G
FE1	300.	6000.	G
FE1	1809.	4500.	L
FE1	300.	1809.	S
FE1 H 2 O 2	300.	6000.	G
FE1 H 3 O 3	300.	1500.	S
FE1 O 1	300.	6000.	G
FE1 O 1	1650.	5000.	L
FE1 O 1	300.	1650.	S
H 1	300.	6000.	G
H 1 N 1	300.	6000.	G
H 1 N 1 O 1	300.	6000.	G
H 1 N 1 O 2	300.	6000.	G
H 1 N 1 O 3	300.	6000.	G
H 1 O 1	300.	6000.	G
H 1 O 2	300.	6000.	G
H 1 ZR1	300.	6000.	G
H 2	300.	6000.	G
H 2 N 1	300.	6000.	G
H 2 O 1	300.	6000.	G
H 2 O 1	273.	373.	L
H 3 N 1	300.	6000.	G
N 1	300.	6000.	G
N 1 O 1	300.	6000.	G
N 1 O 2	300.	6000.	G
N 1 ZR1	300.	6000.	G
N 1 ZR1	3225.	6000.	L
N 1 ZR1	300.	3225.	S
N 2	300.	6000.	G
N 2 O 1	300.	6000.	G
N 2 O 3	300.	6000.	G
O 1	300.	6000.	G
O 1 ZR1	300.	6000.	G
O 2	300.	6000.	G
O 2 ZR1	300.	6000.	G
O 2 ZR1	2950.	6000.	L
O 2 ZR1	300.	2950.	S
ZR1	300.	6000.	G
ZR1	2125.	5500.	L
ZR1	300.	1500.	S

IMPULSE EFFICIENCY

The rocket motor impulse efficiency is a fixed user input, determined from the empiricisms defined for the SPP(Reference 12)computer code, or calculated from a user-supplied model.

As employed in the SPOC ballistic simulation module, impulse efficiency is the ratio of delivered vacuum specific impulse at motor pressure and expansion ratio to theoretical specific impulse at identical conditions. Divergence losses are not included in the efficiency factor, but are accounted for through the thrust coefficient calculated by the code. In actual practice, the impulse efficiency is treated as a "thrust efficiency" during ballistic simulation. However, the impulse efficiency predicted in SPP already includes a divergence loss term. Therefore, when the SPP model is selected by the user (SPPETA=T), the divergence loss calculation in the ballistic module is by-passed by internally setting the effective nozzle exit half-angle to zero.

Impulse efficiency calculated by the SPP empiricisms is as follows (where code nomenclature is also shown)

$$ETAISP = (ETACSR)(ETACF)$$

where

$$ETACF = 1 - (ETABL)(ETADIV)(ETAKIN)(ETASUB)(ETATP)/100$$

and

ETATP	is impulse loss effect, in %, due to two-phase flow
ETADIV	is impulse loss effect, in %, due to nozzle divergence
ETAKIN	is impulse loss effect, in %, due to finite rate reaction kinetics
ETABL	is impulse loss effect, in %, due to boundary layer buildup in the nozzle
ETACSR	is impulse loss effect, in %, due to C* efficiency (i.e., same as combustion efficiency)
ETASUB	is impulse loss effect, in %, due to nozzle submergence. Always equal to one because submerged nozzles are not provided in SPOC.

The various losses are defined by empiricisms as described below.

ETATP

The two-phase flow loss is given by the empiricism

$$ETATP = C_3 \left\{ \frac{\xi^{C_5} C_4 D_P}{P^{0.15} \epsilon_R^{0.08} D_t C_6} \right\}$$

where

ξ is mol fraction of condensed species (at average motor chamber pressure) expressed in mols of condensed species per 100 grams of mixture

D_P is condensed specie particle size in microns and is given by

$$D_P = 0.454\, P^{1/3}\, \xi^{1/3} \left[1 - e^{-.004L^*}\right] \left[1 + .045\, D_t\right]$$

P is average motor chamber pressure in psia (PBAR)

ϵ_R is nozzle expansion ratio at ignition conditions (ERI)

D_t is nozzle throat diameter in inches at ignition (DTI)

L^* is motor L^* in inches at ignition (LSTRI)

e is the Naperian base 2.71828...

The coefficients C_3, C_4, C_5, and C_6 are determined as follows:

if $\xi < 0.09$

$C_4 = 1.0$

if $D_t < 1$; $C_3 = 30$, $C_5 = 1$, $C_6 = 1$

if $1 \leq D_t \leq 2$; $C_3 = 30$, $C_5 = 1$, $C_6 = 0.8$

if $D_t > 2$ & $D_P < 4$; $C_3 = 44.6$, $C_5 = 0.8$, $C_6 = 0.8$

if $D_t > 2$ & $D_P \leq 8$; $C_3 = 34$, $C_5 = 0.8$, $C_6 = 0.4$

if $D_t > 2$ & $D_P > 8$; $C_3 = 25.2$, $C_5 = 0.8$, $C_6 = 0.33$

$$\underline{\text{if } \xi \geq 0.09}$$

$$C_4 = 0.5$$

if $D_t < 1$; $C_3 = 9$, $C_5 = 1$, $C_6 = 1$

if $1 \leq D_t \leq 2$; $C_3 = 9$, $C_5 = 1$, $C_6 = 0.8$

if $D_t > 2$ & $D_P < 4$; $C_3 = 13.4$, $C_5 = 0.8$, $C_6 = 0.8$

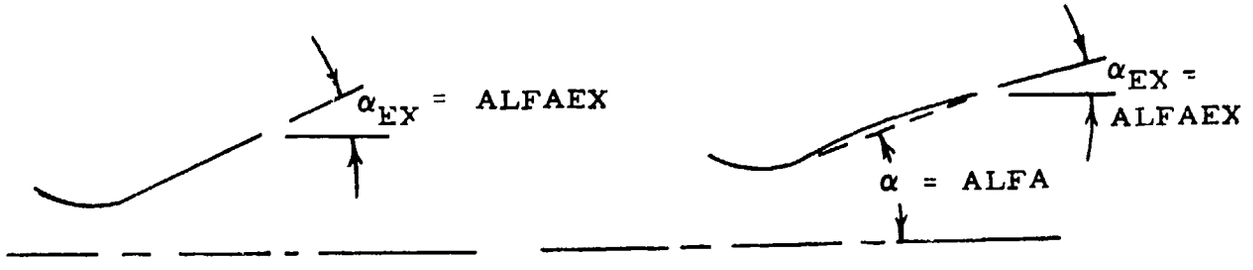
if $D_t > 2$ & $4 \leq D_P$; $C_3 = 10.2$, $C_5 = 0.8$, $C_6 = 0.4$

if $D_t > 2$ & $D_P > 8$; $C_3 = 7.58$, $C_5 = 0.8$, $C_6 = 0.33$

ETADIV

The nozzle divergence loss is given by

$$\text{ETADIV} = 50 \left[1 - \cos\left(\frac{\alpha + \alpha_{EX}}{2}\right)\right]$$



Conical Nozzle

Contoured Nozzle

ETAKIN

The reaction kinetics loss is given by

$$\text{ETAKIN} = \left[\frac{100}{3}\right] \left[1 - \frac{I_{sp_{V_{TF}}}}{I_{sp_{V_{TS}}}}\right] \left[A\right]$$

where

$I_{sp_{V_{TF}}}$ is theoretical vacuum I_{sp} computed at ignition nozzle geometry and assuming frozen equilibrium thermochemistry (IVACF, FISP)

$I_{sp_{V_{TS}}}$ is theoretical vacuum I_{sp} computed at ignition nozzle geometry and assuming shifting equilibrium thermochemistry (IVAC, SISP)

if $P \leq 200$, $A = 1$

if $P > 200$, $A = \frac{200}{P}$

P is average motor chamber pressure in psia (PBAR)

ETABL

The boundary layer loss is given by

$$ETABL = C_1 \left[\frac{P^{0.8}}{D_t^{0.2}}\right] \left[1 + 2\, e^{\left(-C_2\, P^{0.8}\, \left(\frac{t}{D_t^{0.2}}\right)\right)}\right] \left[1 + .016(\epsilon_R - 9)\right]$$

where

C_1 is 0.00365

C_2 is 0.000937

P is average motor chamber pressure in psia (PBAR)

D_t is nozzle throat diameter in inches at ignition (DTI)

e is the Naperian base 2.71828...

t is motor burn time in seconds (TB)

ϵ_R is nozzle expansion ratio at ignition conditions (ERI)

ETACSR

The c* efficiency loss is given by

$$ETACSR = \left[K + \left(\frac{10-d}{10}\right)(100-K)\right] \quad b\,(c)$$

if $a \leq 10$, $d = a$

if $a > 10$, $d = 10$

where

a is the weight percent of aluminum in the propellant formulation

b = 1.0

c = 1.003

The parameter K is a function of burning rate and is determined by the following table. Burning (RBBAR) is evaluated at motor average pressure.

Burning Rate (in/sec)	K	Burning Rate (in/sec)	K
<0.11	91.4	0.50	98.6
0.11	91.4	0.60	98.9
0.12	92.1	0.70	99.1
0.13	94.0	0.80	99.2
0.14	94.6	0.90	99.3
0.15	95.1	1.00	99.4
0.16	95.6	1.20	99.6
0.17	96.0	1.40	99.7
0.18	96.4	1.60	99.8
0.19	96.7	1.80	99.9
0.20	97.0	2.00	100.0
0.30	97.7	>2.00	100.0
0.40	98.2		

Values for the parameters used to calculate the various loss factors are generated at numerous places in the code. Table 4 lists the parameters (according to both the calling and called arguments) and their sources. The primary influence was the need to use data from the just completed evaluation (the preceding pass through COMP) in order to not have an iteration within one COMP evaluation. For example, the proper chamber pressure to use in estimating ETAISP is the one at which the motor will operate; but since pressure and thrust simulations are performed concurrently, the average pressure is not available when the estimate for ETAISP is needed. Thus the technique of using pressure from the just completed simulation was adopted. The error usually should be small (at least not excessive), and as the optimum design is approached the small step-size by PATSH will produce only small changes in basic parameters. Therefore the conditions for the just completed evaluation should be essentially identical to the current evaluation. As seen in Table 4, there are other parameters treated in the same manner; unless otherwise stated, the data come from the current evaluation.

TABLE 4

SOURCES OF DATA USED IN SPP PREDICTION OF IMPULSE EFFICIENCY

Calling Argument	Called Argument	Definition and Units
TB	TB	Burn time (sec) from zero to time at which 99.5% of the propellant has been consumed. Taken from preceding ballistic simulation at low temperature (if a two-temperature problem), otherwise from the single temperature simulation. One second on initial pass.
P	PBAR	Average pressure (psia) over TB, taken at same conditions as TB, from preceding pass through COMP. Input PC (default to 1000 psia) on initial pass.
ALFA	ALFA	Half-angle (deg) at entrance to exit cone in contoured nozzle. Equal to ALFAEX in conical exit cone.
ALFAEX	ALFAEX	Half-angle (deg) at exit of exit cone.
NOZER	ERI	Nozzle initial expansion ratio.
RATE	RBBAR	Burning rate (in/sec) at pressure PBAR. Burn rate subroutine called with pressure P to obtain this value.
LSTRI	LSTRI	Initial L* (in). Ratio of initial port volume from preceding evaluation to initial throat area. Port volume includes only that part of chamber occupied by propellant. Estimate for first pass through COMP is $(\pi)(\text{RMOTOR})^2(\text{LMOTMX})/3$.
DTI	DTI	Initial throat diameter (in).
IVACF	FISP	Theoretical specific impulse (lbf-sec/lbm) at frozen equilibrium, current nozzle expansion ratio, preceding chamber pressure; from subroutine TCHEM.
IVAC	SISP	Theoretical specific impulse (lbf-sec/lbm) at shifting equilibrium, current nozzle expansion ratio; at preceding chamber pressure; from subroutine TCHEM.

Table 4 (Continued)

SOURCES OF DATA USED IN SPP PREDICTION OF IMPULSE EFFICIENCY

Calling Argument	Called Argument	Definition and Units
MOLCND	MOLFR	Mole fraction of condensed species (moles per 100 gm of mixture). From subroutine TCHEM at preceding chamber pressure.
PCAL	PCAL	Percent aluminum (%) in propellant.

NOZZLE STRUCTURAL AND ABLATIVE THICKNESS

Each machine access will require the user to specify a nozzle type to be considered. The user may choose one from among six nozzle types shown by Figures 20 through 25.

All nozzle types consist of an inner layer of insulating material supported by an outer structural member. For Nozzle Types 1 and 2 and the exit cone of Type 6, the thickness of the insulating material is established by the erosion, char and thermal penetration depths calculated in the code. For Nozzle Types 3, 4, 5 and the entrance/blast tube of Type 6, the thickness of the insulating material is established by user inputs that define the outer contour of the structural material, which in turn defines the outer contour of the insulating material. Calculations of erosion, char and thermal penetration depths are still made for the latter nozzles, and the results are used to determine if sufficient insulation material has been provided.

Thickness requirements of the structural members are calculated in the code for all the nozzles except for Type 3 and the supersonic portion of Type 4.

Up to three different insulating materials can be specified for Nozzle Types 1, 2 and 6. The boundary between materials is defined by a user-input area ratio (AR1 in the entrance section, AR2 in the exit section). By definition, Nozzle Type 3 and 4 have only one insulating material. Again, by definition, the boundary between insulating materials No. 1 and No. 2 occurs at the aft end of the blast tube on Nozzle Type 5 and Type 6. The boundary between insulating material No. 2 and No. 3 in Nozzle Types 1, 2 and 6 will occur at the AR2 input by the user; however the conical ramp supporting the insert of Nozzle Type 2 will be positioned to connect the two boundaries regardless of the relative magnitudes of AR1 and AR2.

Two structural support materials can be specified. Material No. 1 is for the entrance and throat regions; No. 2 is for the exit region. The boundary between Structural Materials No. 1 and 2 occurs a distance XSTRAN downstream of the boundary between Insulating Materials No. 2 and 3.

The stagnation pressure from which the local static pressure is calculated is MEOP (maximum expected operating pressure) determined from the high temperature ballistic simulation; thus all pressure-dependent analyses in the nozzle subroutine include an inherent degree of conservatism by the use of the MEOP, which is the upper three-sigma maximum pressure. Additional conservatism is included in motors with either a progressive or regressive pressure history which have an average pressure less than the maximum on which MEOP is based.

Burn time used in the nozzle analyses is the nominal burn time determined from the low temperature ballistic simulation.

Establishing Internal Contour

The program uses the parameters shown in Figure 66 to describe the internal contour of the nozzle in terms of an X-R coordinate system with its origin at the nozzle entrance.

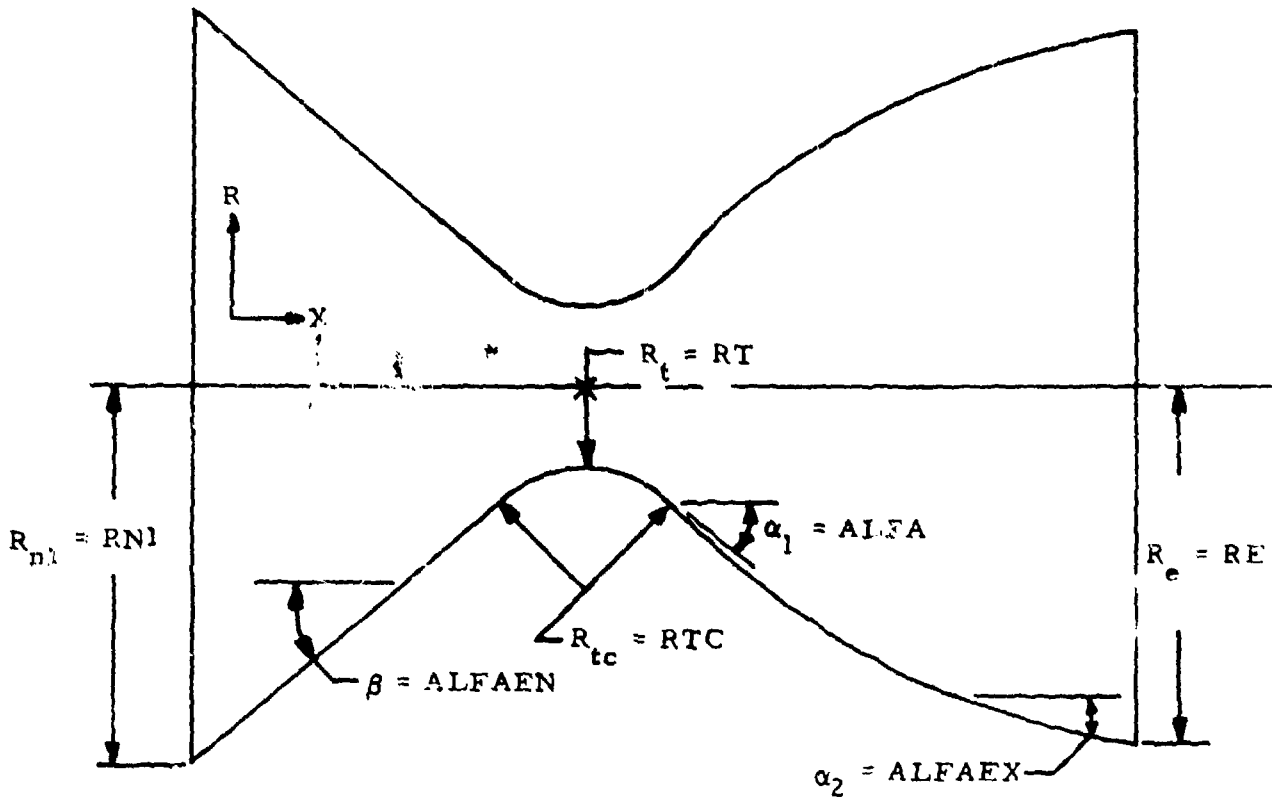


Figure 66. Nozzle Descriptive Nomenclature for Internal Contour

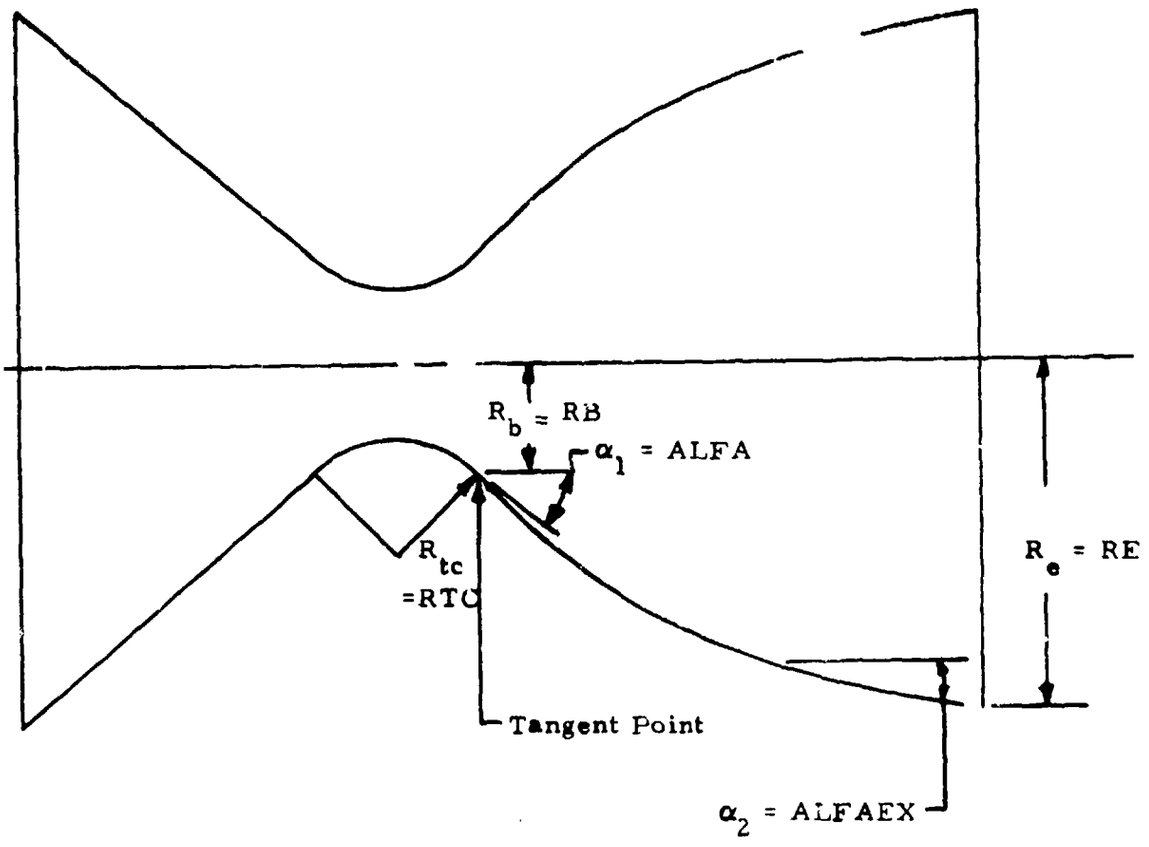


Figure 67. Conditions for Contoured Expansion Section

The parameters shown in Figure 67 are used to determine whether the nozzle is to have either a conical, elliptic, parabolic or hyperbolic exit cone. This decision is made in the following way.

Condition	Decision
$\alpha_1 = \alpha_2$ (input)	Conical Exit Section
$\left[\frac{\text{Tan } \alpha_1}{\text{Tan } \alpha_2}\right]^2 > \left[\frac{R_e}{R_b}\right]^2$	Elliptic Exit Section
$\left[\frac{\text{Tan } \alpha_1}{\text{Tan } \alpha_2}\right]^2 = \left[\frac{R_e}{R_b}\right]^2$	Parabolic Exit Section
$\left[\frac{\text{Tan } \alpha_1}{\text{Tan } \alpha_2}\right]^2 < \left[\frac{R_e}{R_b}\right]^2$	Hyperbolic Exit Section

The inner contour is described by an X-R array, where the X-coordinates are established by a user-input incremental X. The procedure to establish the inner contour is the same for all six nozzle types. The only difference comes in defining the blast tube length for Types 5 and 6; there a radius equal to DTI $\sqrt{ARI}$ is held constant until a length LBT is reached. Figure 66 shows general contour information.

Establishing Insulation Thickness

After describing the internal contour, the insulation thickness is determined for each point in the nozzle by considering the amount needed for erosion, char, thermal protection and safety factor.

The thickness of insulation needed due to erosion is determined at each X-R coordinate from a mathematical model

$$r_e = C_1 P^{C_4} M (1 + C_2 \sin \alpha) + C_3 P^{C_5} + C_6 \quad (1)$$

$$\tau_e = (r_e)(t_b) \quad (2)$$

where

r_e = local erosion rate (in/sec)

τ_e = local erosion thickness (in)

t_b = motor burn time (sec)

P = local static gas pressure (psia)

M = local gas Mach number

α = local angle between nozzle internal contour and nozzle centerline (deg)

C_1, C_2---C_6 = parameters that are a function of the insulating material and its location in the nozzle

Values for P and M are calculated from the isentropic flow relations. Local values for α come from the contour routines. C_1, C_2---C_6 are determined from a statistical curve fit of erosion data for the material under consideration, input by user.

The thickness of insulation needed due to char is determined at each X-R coordinate from the following relation found in Reference 13 for an ablating surface

$$\tau_c = \left[\frac{\alpha_c}{r_e}\right] \ln \left[\frac{T_{vapor} - T_{amb}}{T_{char} - T_{amb}}\right] \quad (3)$$

(Nozzle Type 1 Shown)

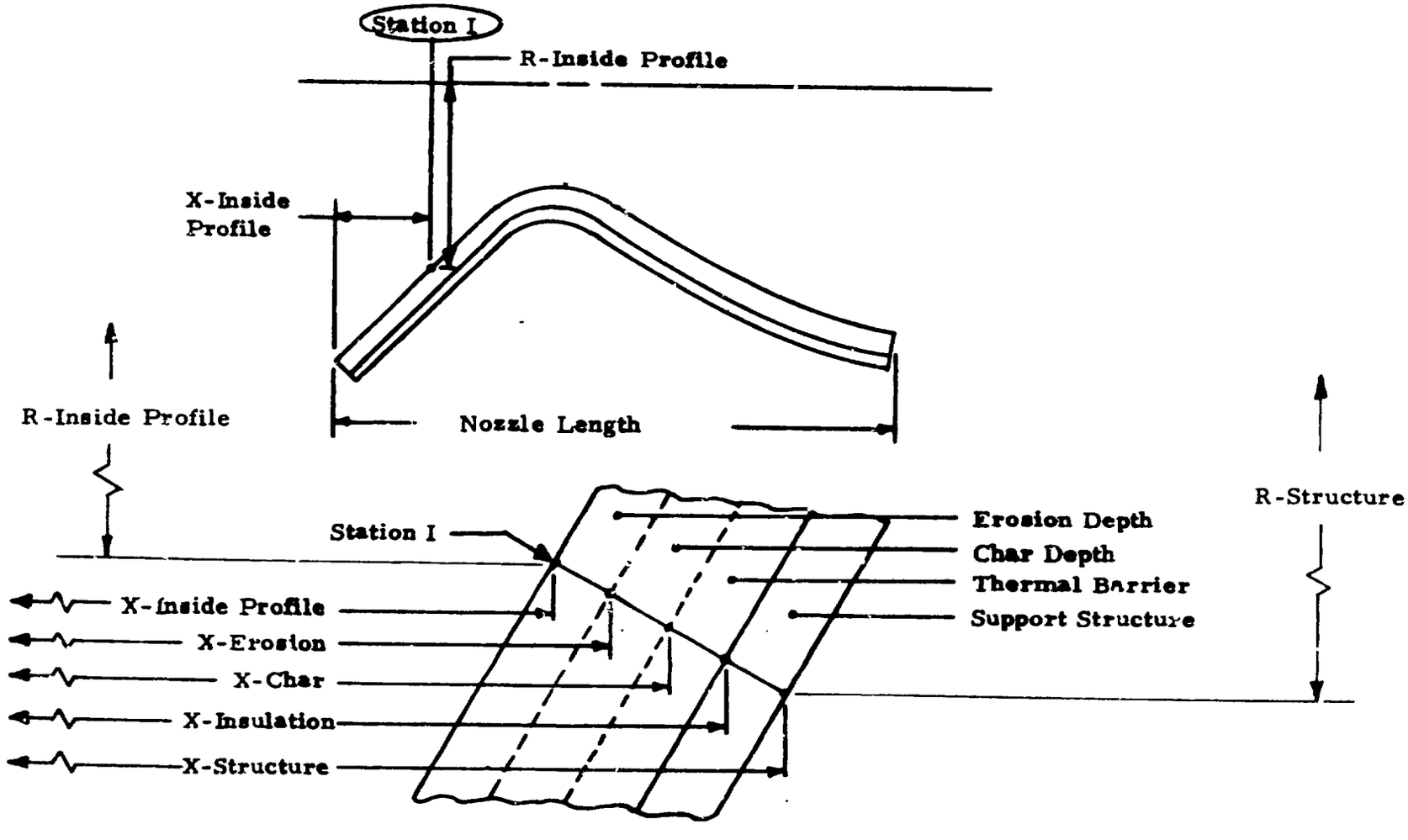


Figure 68. **Nozzle Geometry Showing X-R Coordinates Described in Output of Nozzle Analysis Routine**

τ_c is limited to a maximum of:

$$\tau_c = 0.79 \left\{ \frac{\alpha_c t_b}{-\ln\left[1 - \frac{T_{char} - T_{amb}}{T_{vapor} - T_{amb}}\right]} + 0.574\left[\frac{k_c}{h}\right]^2 \right\}^{\frac{1}{2}} - 0.6\left[\frac{k_c}{h}\right] \qquad (4)$$

where

τ_c	=	char thickness (in)
α_c	=	char thermal diffusivity (sq in/sec)
k_c	=	char thermal conductivity (BTU/in-sec-°F)
T_{char}	=	material char temperature (°F)
T_{vapor}	=	material vaporization temperature (°F)
T_{amb}	=	ambient (initial) temperature of material (°F). Set to 70°F in code.
h	=	surface heat transfer coefficient ($h \to \infty$ for extreme conservatism) (BTU/sq in-sec-°F)

The limiting value is obtained from a curve fit of the Hottel chart for the midplane temperature of a large slab (found in Reference 14).

The thickness of insulation required as a thermal barrier for the nozzle structural material is determined from the following relationship

$$\tau_b = -\left[\frac{\alpha_b t_b}{\tau_e + \tau_c}\right] \ln\left[1 - \frac{T_{alow} - T_{amb}}{T_{vapor} - T_{amb}}\right] C_b + (C_b - 1.0)\tau_e \qquad (5)$$

where τ_b if limited to a maximum of

$$\tau_b = C_b \left\{ 0.79 \left\{ \frac{\alpha_c t_b}{\ln\left[1 - \frac{T_{alow} - T_{amb}}{T_{char} - T_{amb}}\right]} + 0.574\left[\frac{k_b}{h}\right]^2 \right\}^{\frac{1}{2}} - 0.6\left[\frac{k_b}{h}\right] \right\} + (C_b - 1.0)\tau_e \qquad (6)$$

and where

r_b = thermal barrier thickness (in)

α_b = barrier thermal diffusivity (sq in/sec)

k_b = barrier thermal conductivity (BTU/in-sec-°F)

C_b = thermal barrier safety factor (provided by user)

T_{alow} = allowable temperature (°F) of structural material

The limiting value is also obtained from a curve fit of the Hottel chart in Reference 14.

The total insulation thickness is

$$r_i = r_e + r_c + r_b \tag{7}$$

noting that a safety factor is included in the thermal barrier increment.

Separate data sets of input parameters must be furnished for each of the insulating materials inherent with a particular nozzle type (three for Type 1, one for Type 3, one for Type 4, etc.).

$C_1, C_2 \ldots C_6$	k_b
C_b	k_c
T_{vapor}	h
T_{char}	α_c
	α_b

The thickness of the insulation in the entrance section of Nozzle Types 4, 5 and 6 is input by the user (TENT); however, erosion, char and thermal barrier thickness requirements are also calculated. The final thickness is the greater of the two (input or calculated).

Establishing Structure Thickness

Nozzle Type 1 and Type 2

Structural requirements for nozzle Type 1, Type 2 and the exit cone of Type 6 are found with the following analysis. The longitudinal and radial forces the nozzle must support at any point are calculated. (See Figure 69.)

F_x is the longitudinal force found by summing the pressure forces from the exit plane. F_y is the hoop load caused by the local static pressure.

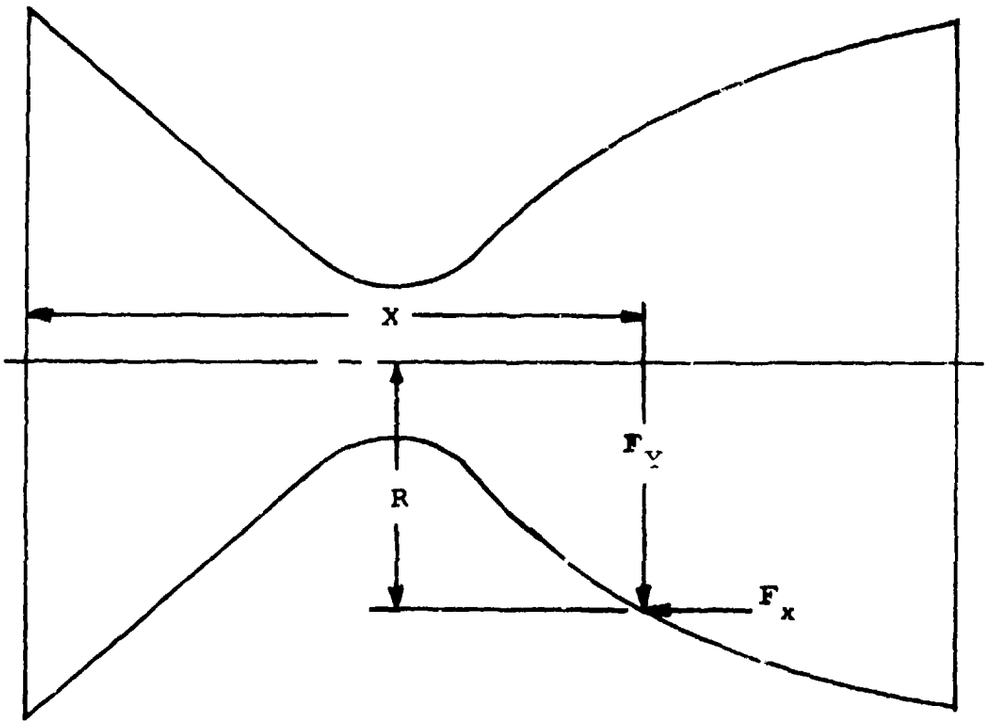


Figure 69. Forces Acting on Nozzle

The structural thickness needed to withstand these forces is calculated from the following relations. Load carrying capability of the insulating material is ignored.

(1) Longitudinal Stress

$$\tau_1 = \frac{F_x}{2 \pi R \cos \alpha \, \delta_c} \tag{8}$$

where

τ_1 = thickness (in) required by longitudinal stress

F_x = longitudinal force (lbf) at individual station, summed from exit plane

R = local radius (in)

α = angle (deg) between nozzle surface and nozzle centerline

δ_c = compressive yield strength (psi) of structural material

(2) Buckling Stress

$$\tau_2 = \sqrt{\frac{5 F_x}{2 \pi \cos \alpha E}} \tag{9}$$

where

τ_2 = thickness (in) required by buckling stress

E = modulus of elasticity (psi) of the structural material

(3) Shear Stress

$$\tau_3 = \frac{F_x \sin \alpha}{2 \pi R \delta_s} \tag{10}$$

where

τ_3 = thickness (in) required by shear stress

δ_s = shear yield strength (psi) of structural material

(4) Hoop Stress

$$\tau_4 = \frac{PR}{\delta_t} \tag{11}$$

where

τ_4 = thickness (in) required by hoop stress

P = local gas pressure (psia)

δ_t = tensile yield strength (psi) of structural material

The thickness of the structural material is the maximum of τ_1, τ_2, τ_3, or τ_4. The input structural safety factor is then applied to the maximum thickness.

Nozzle Type 3

A structural analysis is not performed on Type 3 nozzles because the ablative adds significantly to the load-carrying capability of the assembly. Generally speaking, the capacity of the structure is not a critical item. When a Type 3 nozzle is employed with an Aft Closure Type 2, the structure thickness is made equal to the case thickness (TCASE). When it is combined with an Aft Closure Type 1, the support structure thickness is a user input TSTR3.

Nozzle Type 4

A structural analysis is not performed on the reduced-diameter aft section of Type 4 nozzles; its thickness is a user input TSTR4. Thickness of the structure that forms the entrance section, TENTS, (Figure 70) is estimated through the thin-wall pressure vessel relationship

$$TENTS = \frac{(P)(RENT)}{(FTY)(FSTRUS)} \tag{12}$$

$$RENT = RN1/\sin(90\text{-}ALFAEN) \tag{13}$$

where

P = MEOP (psia)

RENT = Effective radius (in); See Figure 70

FTY = Structural material No. 1 tensile yield strength (psi)

FSTRUS = Structural material No. 1 safety factor

Nozzle Type 5 and 6

Thickness of the structure that forms the entrance to the blast tube is calculated with Eqs (12) and (13). Structure in the blast tube is calculated with

$$TBTS = \frac{P(KUP) * R3(KUP)}{FTY * FSTRUS} \tag{14}$$

where TBTS = Thickness (in) of blast tube structure

P(KUP) = Static pressure (psia) at entrance to throat insert (Station KUP)

R3(KUP) = Outside radius (in) of insulation material along blast tube (Station KUP)

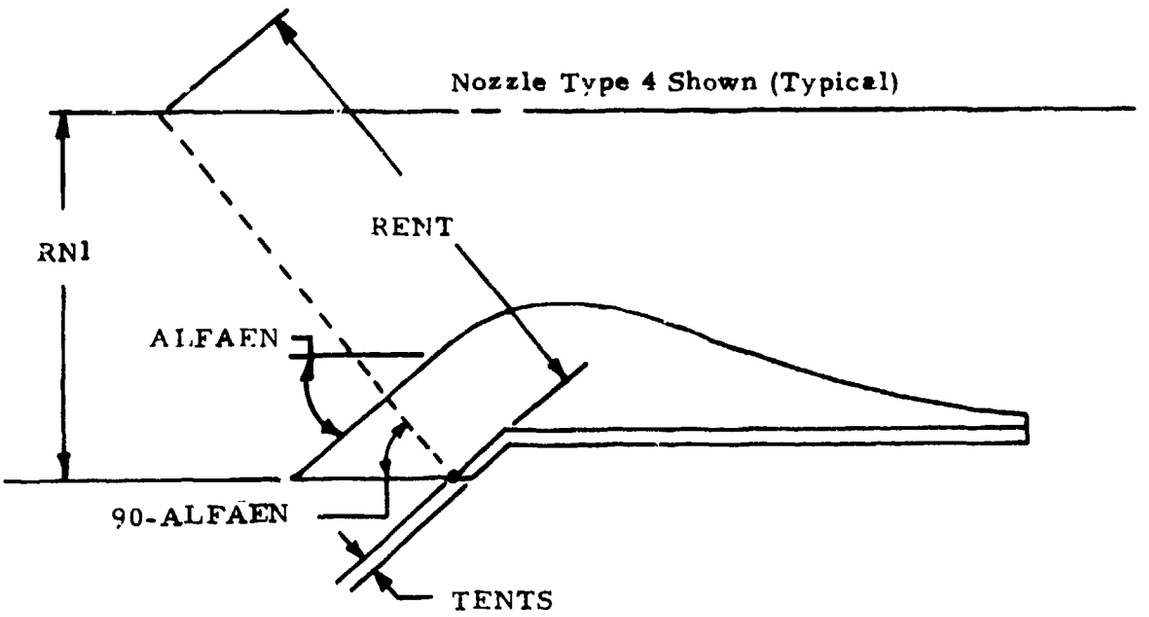


Figure 70. Basis for Structural Analysis of Entrance Section of Nozzle Types 4, 5 and 6

Structure thickness for the exit cone of Nozzle Type 6 is determined with Eqs (8) - (11), as were Types 1 and 2.

Interfaces with Motor and Geometric Verification

The outermost contour of Nozzle Type 1, Type 2 and the exit cone of Type 6 are a result of separate calculations for erosion depth, char depth, thermal barrier and structural thickness, all performed at a number of nodes (located by X-R coordinates) along the internal surface of the motor. Calculations at each node are independent of one another, and so there is some "waviness" in the outer contour (except for behind the throat of Type 2 nozzle). It is recognized that a nozzle would not actually be built with this contour, but the purpose of these analyses is to obtain a reasonably accurate estimate of the nozzle size, and to that extent, the analyses are appropriate.

As described above, the outer contours of Nozzles Type 3, Type 4, Type 5 and the blast tube of Type 6 are established by various user inputs. How the interface between the user-established outer contour and the analysis-established inner contour is controlled is described in later paragraphs.

The primary interface between the motor and nozzle is at the aft case opening where the case opening radius RNOZEN must mate with the nozzle entrance radius RN1. For any given problem, a value of RNOZEN is calculated at every design evaluation (every pass through COMP), whatever the combination of propellant grain configuration and aft closure type might be. The definition of the various possibilities of RNOZEN are illustrated in the figures included as part of the grain configuration discussion.

A series of geometric validations are made prior to ballistic simulation to achieve the proper motor/nozzle interface and to assure compatibility of other nozzle dimensions (Figures 71, 72 and 73). These comparison guarantee:

(1) Radius of boundary between Insulation Materials No. 1 and No. 2 ($RT\sqrt{AR1}$) is less than RNOZEN (Nozzle Types 1, 2, 5 and 6). Note that for Types 5 and 6, this material boundary radius corresponds to the inside radius of the blast tube.

(2) Exit radius (RE) is greater than the radius of the boundary between Insulation Material No. 2 and Insulation Material No. 3 ($RT\sqrt{AR2}$) for Nozzle Types 1, 2 and 6.

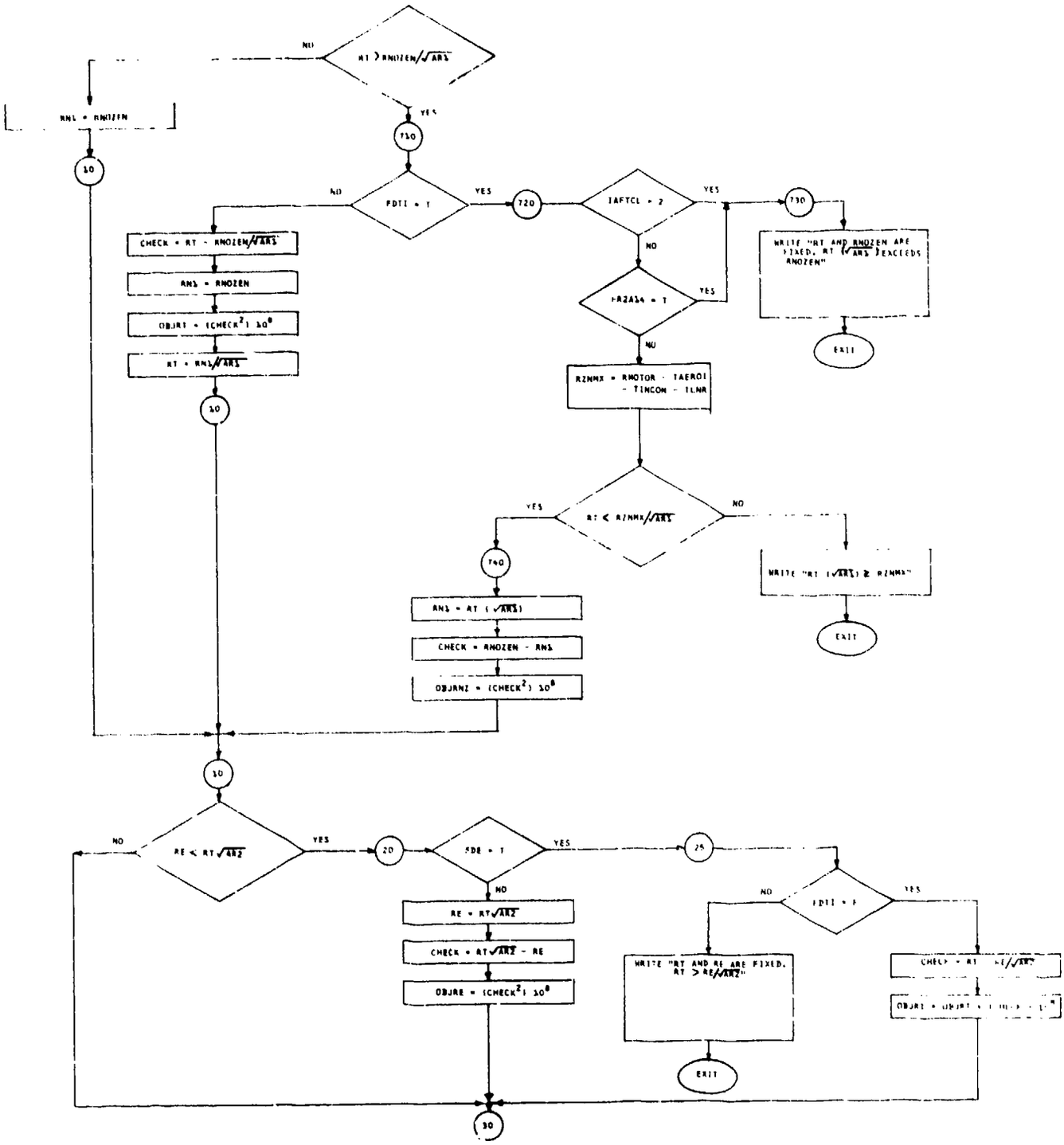


Figure 71. Geometric Verification of Nozzle Types 1, 2 and 6

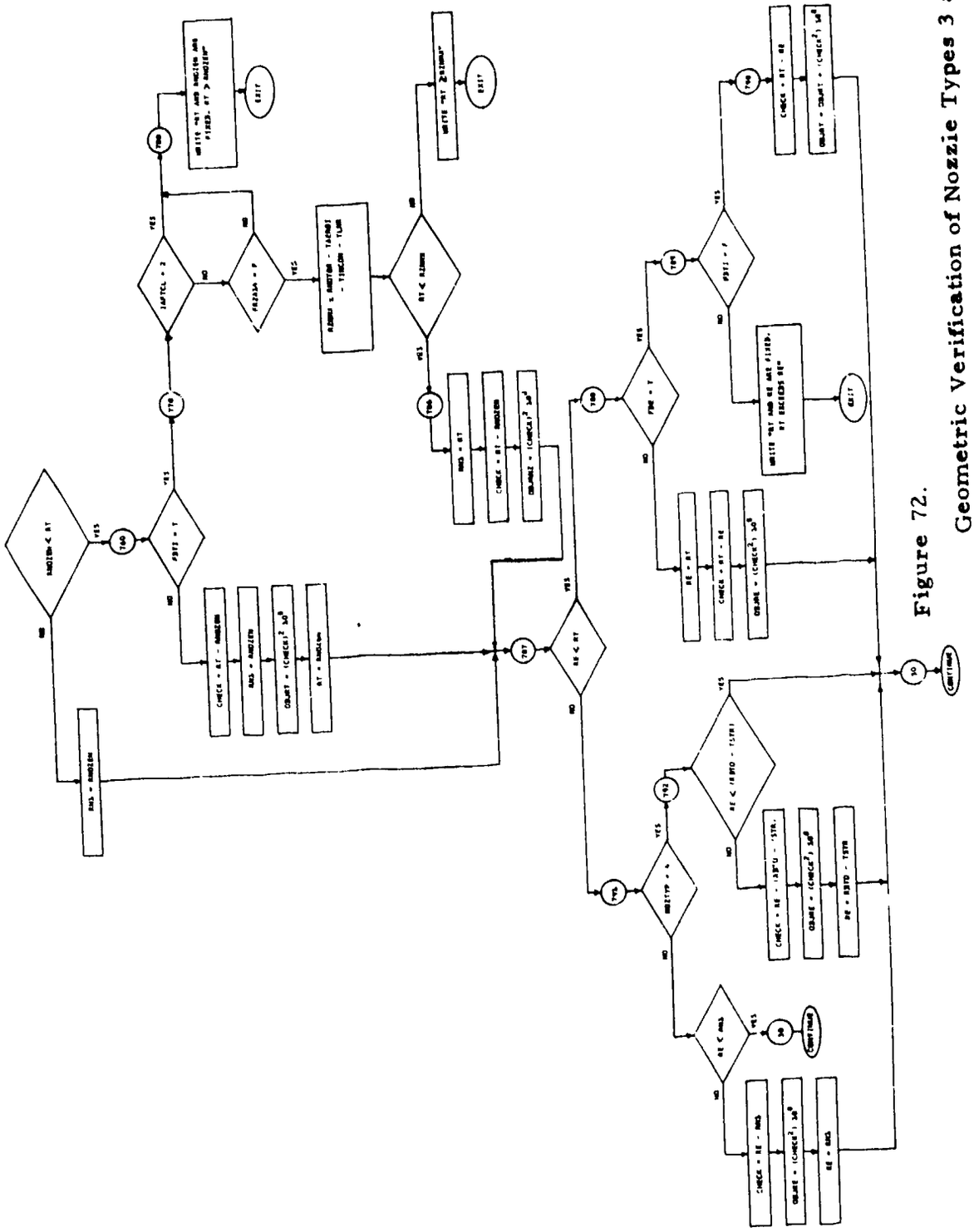


Figure 72. Geometric Verification of Nozzle Types 3 and 4

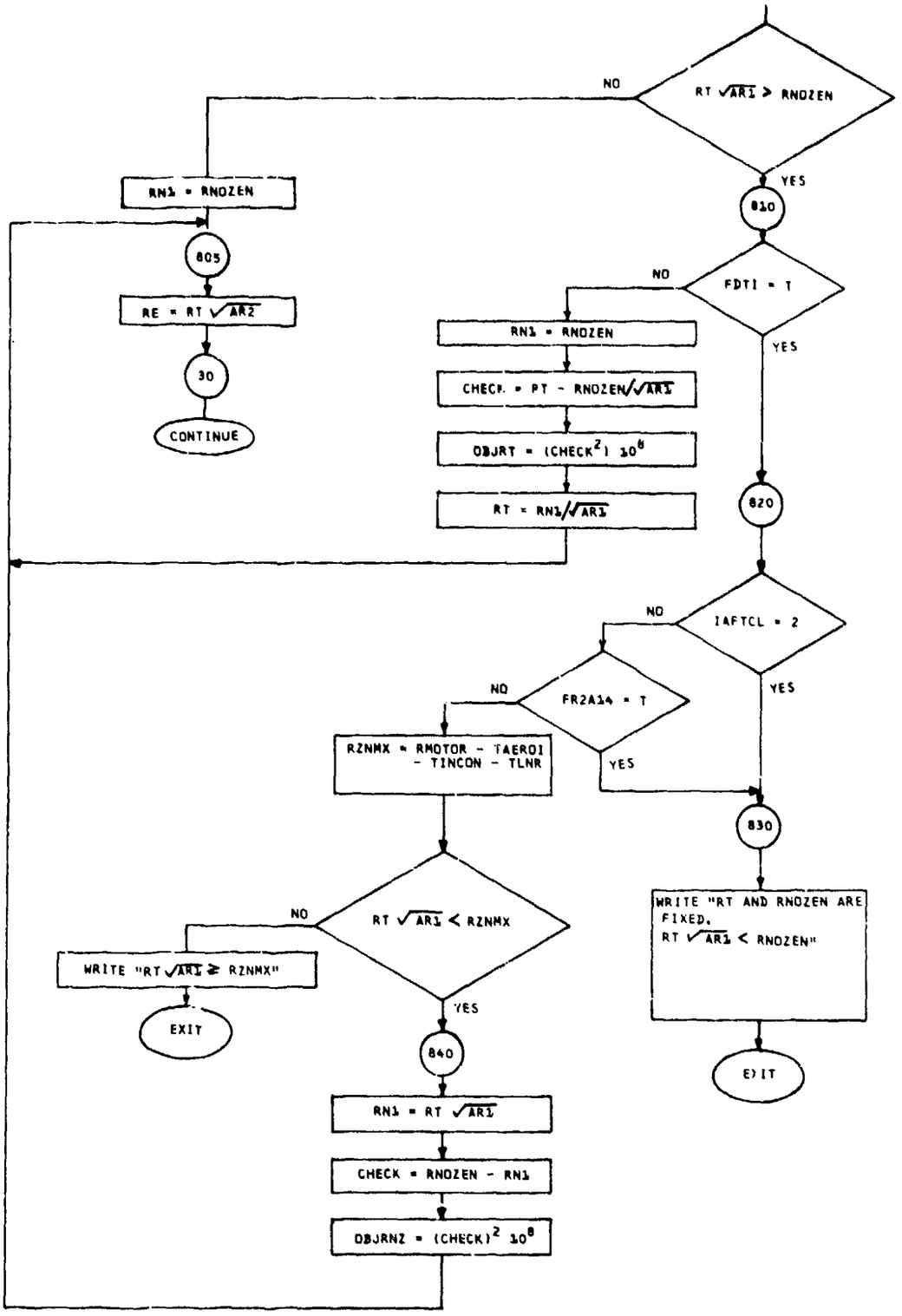


Figure 73. Geometric Verification of Nozzle Type 5

(3) Throat radius (RT) is less than aft case opening (RNOZEN) for Nozzle Types 3 and 4.

(4) Exit radius (RE) is greater than throat radius (RT) for Nozzle Types 3 and 4.

(5) Exit radius (RE) is less than the nozzle entrance radius (RNOZEN) for Nozzle Type 3.

(6) Exit radius (RE) is less than the inside radius of the blast tube support structure (RBTO-TSTR4) for Nozzle Type 4.

(7) Nozzle entrance radius (RN1) is equal to the aft case opening (RNOZEN) for Nozzle Types 1, 2, 5 and 6.

(8) Exit half-angle (ALFAEX) is less than the expansion section entrance angle (ALFA) for contoured nozzles.

When the ballistic simulation is completed, the nozzle thermal and structural analyses are performed. These results are used to determine dimensional compatibility between insulation and support structure in Nozzle Types 3, 4, 5 and 6. Figure 74 illustrates the check made at the exit plane of Nozzle Types 3, 4 and 5; a penalty, OBEXIM, is calculated if the margin EXINSM is less than zero. For the blast tube of Nozzle Types 5 and 6, the outside structure radius is found by summing the required insulation thickness (erosion, char, thermal barrier) and the required support structure thickness with the inside radius; if this total is greater than the user-input RBTO5 (or RBTO6), a penalty is calculated (OBBTO5 or OBBTO6).

The outside radius of the exit section (RAO) is also determined for Nozzle Types 1, 2 and 6 after the nozzle analyses are performed. If RAO is greater than a user limit, a penalty is calculated (OBJDEO). The outside exit radius limit is input as the ratio of nozzle diameter to motor diameter (NTMR). Thus the nozzle exit size can be adjusted in concert with the motor diameter if the latter is one of the adjusted parameters in an optimization problem.

Another check is on the length of the blast tube (i.e., the reduced diameter aft section) of a Type 4 nozzle. If the length calculated in the analysis (LBT4) is not equal to the required length LBT4RQ, a penalty is calculated (OBLBT4).

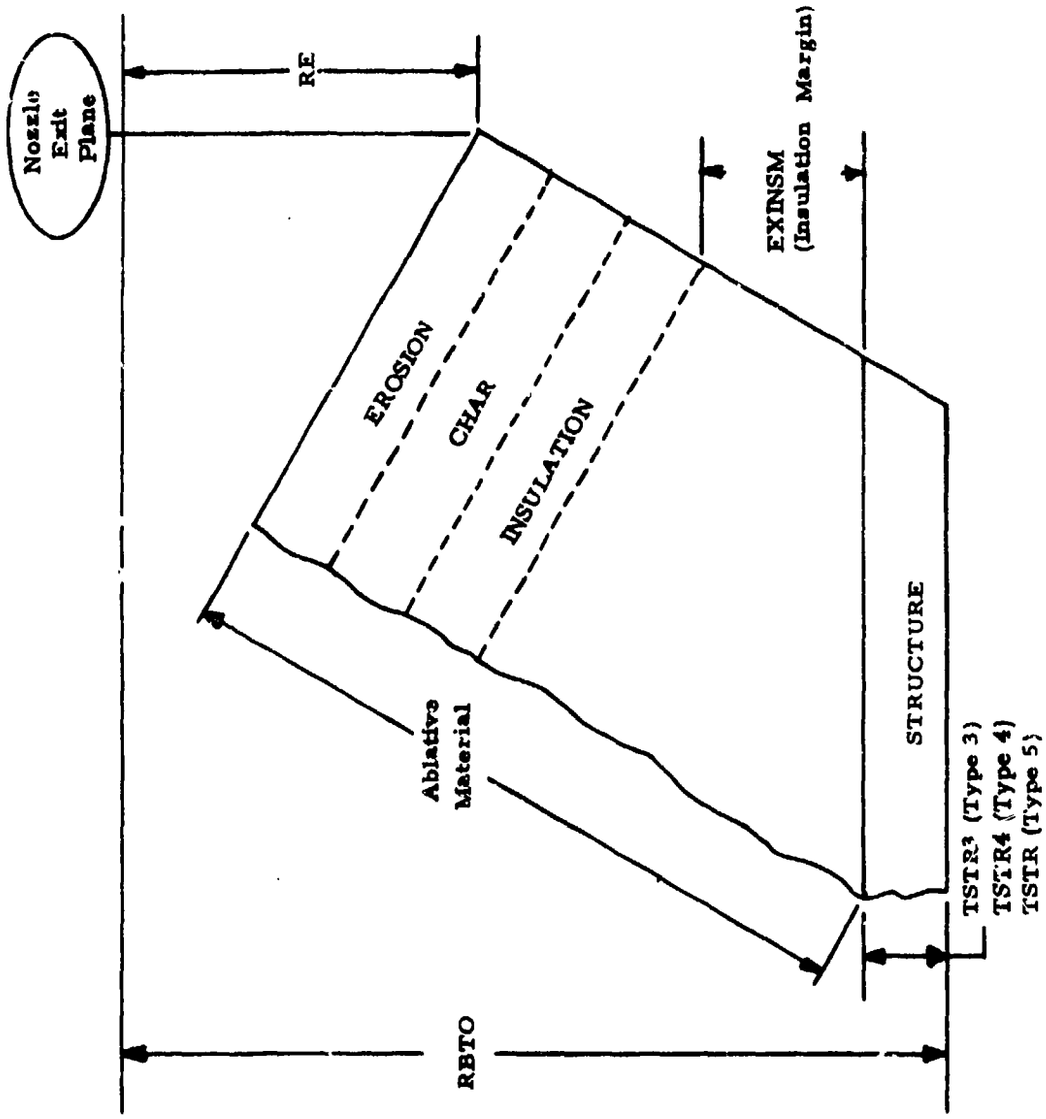


Figure 7-4. Insulation Margin at Exit Plane of Nozzle Types 3, 4 and 5

PRESSURE VESSEL STRUCTURAL ANALYSIS

Thicknesses of structural material required to withstand maximum expected operating pressure (MEOP), plus a safety factor, are calculated with conventional relationships (Reference 15). To account for the several case/closure arrangements, the basic stress relationships are employed for:

- o Flat plate unrestrained on outer edge, such as when closure is held in place by retaining ring (Forward Closure Type 2)
- o Flat plate restrained on outer edge, such as when closure is integral with tubular portion of case (Forward Closure Type 3)
- o Elliptical domes for both forward and aft closures (Type 1)

These calculations are performed in the subroutine CASEAN.

Two structural design pressures are calculated immediately after the ballistic simulation; one is for use with yield tensile properties and the other is for use with ultimate tensile properties.

$$PYIELD = (FSYLD)(MEOP) \quad (1)$$

$$PULT = (FSULT)(MEOP) \quad (2)$$

where FSYLD = Factor of safety for yield conditions
FSULT = Factor of safety for ultimate conditions
MEOP = Maximum expected operating pressure (psia); upper three-sigma maximum pressure at high temperature firing

A test is made to determine whether the yield condition or the ultimate condition is the more critical; the decision depends on the relationship between the two safety factors and the material yield and ultimate strengths

$$F = \frac{(FSULT)(FTYC)}{(FSYLD)(FTUC)} \quad (3)$$

where FTYC = Case structural material tensile yield strength (psi)
FTUC = Case structural material tensile ultimate strength (psi)

If $F > 1$, the ultimate condition is more critical (i.e., when designing to ultimate condition such that the ultimate factor of safety is FSULT, the resultant yield factor of safety will be greater than that required, FSYLD). If $F < 1$, the reverse situation prevails. Note that this test uses case material properties and the decision is applied even to the forward closure that is a separate part (Type 2).

Ellipsoidal Dome Closures, Forward and Aft (Type 1)

Thickness of an ellipsoidal closure is calculated from Case 5, Table XIII (Reference 15), considering conditions at the centerline (Figure 75).

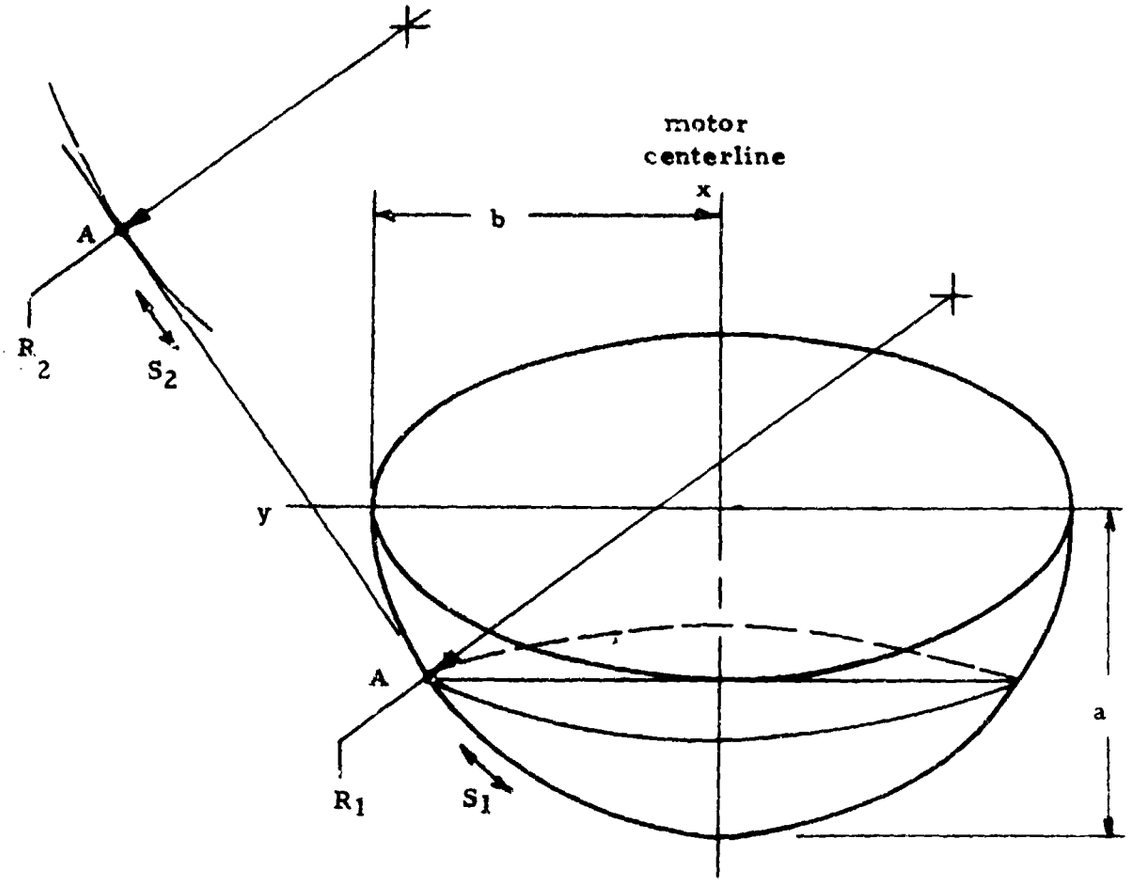


Figure 75. Ellipsoidal Closure Stress Analysis

The radius R_2 can be shown to be

$$R_2 = a \left[(x/a)^2 (1-K^2) + K^2 \right]^{\frac{1}{2}} \quad (1)$$

Where K = a/b. When x = zero, $R_2 = a^2/b = aK$. Thus in terminology used in the code, R_2 becomes

$$R = (RCI)(BETA2F) \quad (2)$$

where RCI = case inside radius (in)
BETA2F = ellipse ratio of dome internal surface = a/b

From the above quoted Reference 15 case, hoop and meridional stress are equal at the center, so that the required closure wall thickness (TCLOF or TCLOA) is

$$\text{TCLOF (or TCLOA)} = \frac{(P)(R)}{(2)(STRESS)} \tag{3}$$

where P = critical pressure (psia), ultimate or yield, selected as described above

R = radius (in) from Eq (2)

STRESS = case material strength (psi), ultimate or yield, selected as described above.

If the thickness calculated with Eq (3) is less than the thickness of the case cylindrical wall (TCASE), it is set equal to TCASE. Mathematically, the condition of TCLOF < TCASE will occur when BETA2F<2; however, manufacturing experience has shown that it usually involves extra expense to provide the thinner closure.

Flat Plate Forward Closure Not Integrally Attached (Type 2)

For a flat plate loaded as shown in Figure 76, the required thickness is related to radial stress at the center by (Case 1, Table X, Ref. 15)

$$f_r = \left[\frac{3P\pi r^2}{8\pi m t^2}\right]\left[3m + 1\right] = \left[\frac{3Pr^2\nu}{8t^2}\right]\left[3m + 1\right] \tag{4}$$

from which

$$t = \sqrt{\frac{3\,Pr^2\,\nu\,(3m + 1)}{8\,F_{t_x}}} = \text{TCLOF} \tag{5}$$

Nomenclature is in Table 5.

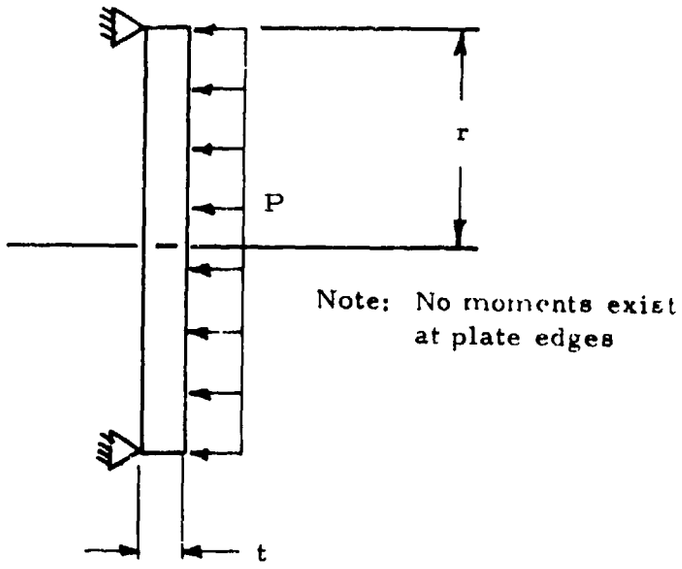


Figure 76. Unrestrained Flat Plate Forward Closure

This type of forward closure stress analysis applies when the flat plate is keyed (or similarly attached) to the case wall. Case strength levels are used to calculate this closure thickness.

Flat Plate Forward Closure, Integral with Case (Type 3)

Stress in this closure is found by superimposing radial stresses, S_r, as defined in Reference 15 (radial at center of closure).

Pressure (Case 1, Table X)

$$S_r = -\frac{3\,PR^2\,(3m+1)}{8\,m\,t_1^2} \tag{6}$$

Moment (Case 12, Table X)

$$S_r = -\frac{6M}{t^2} \tag{7}$$

Shear

$$S_r = \frac{V}{t} \tag{8}$$

where M and V are found from Case 24, Table XIII.

Superposition and then combining terms results in

TABLE 5

NOMENCLATURE FOR PRESSURE VESSEL STRUCTURAL ANALYSIS

P:	motor pressure in psia, yield or ultimate conditions, PULT, PYIELD
r:	plate radius in inches (same as motor case inside radius), RCI
m:	reciprocal of Poisson's ratio for case material, XM
E:	modulus of elasticity for case material, MODCAS
t:	forward closure thickness in inches, TCLOF
ν:	Poisson's ratio for case or closure material, PRCAS, PRCLO
R:	radius (in inches) from motor centerline to center of case thickness, CAPR
t_c:	case wall thickness in inches, TCASE
M:	moment applied to case lip in inch-pounds per inch of circumference
V:	radial shear load applied at case lip in pounds per inch of circumference
F_{tx}:	tensile strength of case material---either ultimate or yield, FTUC, FTYC
R_m:	case outside radius in inches, RMOTOR
a:	ellipse semi-minor diameter in inches, A1A or A1F
b:	ellipse semi-major diameter in inches, B1A or B1F
f_r:	radial stress in psi (tension)

$$S_r = x_1 - \frac{6}{t^2}\left[(x_2 + x_3)/(x_4 - x_5)\right] + \frac{1}{t}\left(x_4\left[(x_2 + x_3)/(x_4 - x_5)\right] - x_2\right) \quad (9)$$

where

$$x_1 = \frac{3\,PR^2\,(3m + 1)}{8mt^2} \quad (10)$$

$$x_2 = \frac{12\,PR^3\lambda^2\,D\,(1-\nu^2)}{4\,(1 + \nu)\,Et^3} \quad (11)$$

$$x_3 = \frac{2\,PR^2\lambda^3\,t\,D}{t_c\,(1 - \nu/2)\,\left[Et + 2RD\lambda^3\,(1-\nu)\right]} \quad (12)$$

$$x_4 = 2\lambda + \frac{24R\lambda^2\,D\,(1-\nu^2)}{Et^3\,(1 + \nu)} \quad (13)$$

$$x_5 = \frac{\lambda\,Et}{Et + 2D\lambda^3\,R\,(1-\nu)} \quad (14)$$

$$\lambda = \left[\frac{3\,(1-\nu^2)}{R^2\,t_c^2}\right]^{0.25} \quad (15)$$

$$D = \frac{Et_c^3}{12\,(1-\nu^2)} \quad (16)$$

Eq (9) is solved iteratively for t (i.e., TCLOF and TCLOA) until the calculated radial stress S_r is within 0.1% of the critical strength (either FTUC or FTYC).

Case Cylindrical Section

The case thickness required to withstand the predicted design pressure (TCREQ) is calculated by

$$TCREQ = \frac{(P)(R_m)}{F_{tx}} \quad (17)$$

This is compared with the current value of case thickness (TCASE) that was used to determine the propellant external dimensions. If TCASE< TCREQ, a penalty is calculated.

PROPELLANT STRUCTURAL ANALYSIS

Strain imposed on the propellant grain due to low-temperature storage and low-temperature ignition is calculated with "plane-strain" relations (Reference 16) for the cylindrically perforated cavity. Strain in the valley of a star or slotted configuration is found by application of strain concentration factors (Reference 17) to strain calculated for equivalent cylindrically perforated ports.

It was decided to use plane strain analyses of circular port geometries to determine these strains and to modify these with appropriate concentration factors when the port is not circular. Such analyses should be conservative for storage condition loading, in that plane strain analyses predict higher hoop strains (and stresses) than do three-dimensional analyses. Also, "lobes" of propellant protruding into the bore restrict hoop strain somewhat, but this effect is ignored. For ignition conditions, plane strain analyses are expected to be accurate except at transitions between irregular ports and circular ports. At the transitions the actual hoop stress and hoop strain are bounded from above by plane stress values and from below by plane strain results.

For total hoop strain imposed by low temperature storage (Table I, Reference 15):

$$\epsilon_{\vartheta t} = -(1+\nu)(\alpha)(\Delta T)\left[\left[1-\frac{\alpha_c(1+\nu_c)}{\alpha(1+\nu)\Omega}\right]\left[1-2\nu+\left(\frac{a}{r}\right)^2\left(\frac{b}{a}\right)^2\right]-1\right] \quad (1)$$

where

ν = Poisson's ratio of propellant (in/in), PRP

ν_c = Poisson's ratio of case (in/in), PRCAS

α = Coefficient of thermal expansion of propellant (in/in/°F), ALPHAP

α_c = Coefficient of thermal expansion of case (in/in/°F), ALPHAC

ΔT = Conditioned temperature minus strain-free temperature (°F), DELT = TLO-SFTEMP

a = Bore radius (in), R2

b = Propellant outside radius (in), RF

r = Radius at which calculations are being made = a when bore strains are being calculated

Ω_t = Intermediate function defined in Eq (2) for thermal loading of the grain, SIGTHM

$$\Omega_t = \left[(1 - 2\nu)\left(\frac{b}{a}\right)^2 + 1\right] + \frac{\left[\left(\frac{b}{a}\right)^2 - 1\right]\left[1 - \nu_c^2\right]}{(1 + \nu)(h)(E_c)}(b)(E_{pt}) \quad (2)$$

E_{pt} = Equilibrium modulus of propellant for use in thermal loading (psi), MODPT

E_c = Tensile modulus of case (psi), MODCAS

h = Case wall thickness (in), TCASE

Total strain found by Eq (1) is composed of mechanical (E_m) and thermal (E_t) components; the mechanical component is compared to strain endurance capability of the propellant to determine the structural margin of safety

$$\epsilon_m = \epsilon_{\theta t} - \epsilon_t = \epsilon_{\theta t} - \alpha(\Delta T) \quad (3)$$

where ϵ_m = Mechanical hoop strain imposed on propellant due to low temperature storage (in/in), EPT

$\epsilon_{\theta t}$ = Total hoop strain imposed on propellant due to low temperature storage (in/in)

ϵ_t = Thermal strain (in/in)

For hoop strain imposed by ignition pressurization at low temperature (Table II, Reference 1), which are superimposed on those strains due to thermal shrinkage

$$\epsilon_{\theta p} = \left[\frac{(1 + \nu)(P)}{\left[\left(\frac{b}{a}\right)^2 - 1\right] E_{pp}}\right]\left[\left[1 - 2\nu + \left(\frac{b}{a}\right)^2\right] - \frac{2(1 - \nu)}{\Omega_p}\left[(1 - 2\nu)\left(\frac{b}{a}\right)^2 + \left(\frac{b}{r}\right)^2\right]\right] \quad (4)$$

where $\epsilon_{\theta p}$ = Hoop strain due to pressurization (in/in), EPP

P = Ignition pressure (psia), PIGN

E_{pp} = Modulus of propellant appropriate for ignition pressurization rate and temperature (psi), MODPP

Ω_p = Intermediate function defined in Eq (5) for pressurization loading of the grain, SIGPR

and all other nomenclature is the same as before.

$$\Omega_p = (1-2\nu)\left(\frac{b}{a}\right)^2 + 1 + \frac{\left[\left(\frac{b}{a}\right)^2 - 1\right]\left[1-\nu_c^2\right](b)(E_{pp})}{(1+\nu)(h)(E_c)} \quad (5)$$

A complete development of these relations is given in Reference 17.

When calculating strains in the CP portion of a grain, a = R2 = radius of CP port, and imposed hoop strains will be found directly from Eqs (3) and (4) for thermal (EPT) and pressurization (EPP) conditions, respectively.

When calculating strains in the valley of star or slotted portions of the grain, a = radius to "bottom" of valley for use in Eqs (3) and (4). Then imposed strains are found by

EPTS = Thermal strain in valley = (K) (EPT) (6)

EPPS = Pressurization strain in valley = (K)(EPP) (7)

where K = concentration factor. Figure 77 shows a section of a typical star geometry. The general form of the equation for the concentration factor is

$$K = \left[\frac{\lambda^2 - 1}{2\lambda^2}\right] H \quad (8)$$

in which λ = b/a and H depends on the geometry of the star. When the angle β is zero, H is given by

$$H = N^{-1/3}\left[\sqrt{\frac{a+b}{b-a}}\right]\left[1 + 2\sqrt{\frac{a}{p}}\right] \quad (9)$$

where N = Number of star points ($2 \leq N \leq 8$), NSLOTS

p = Fillet radius (in) between star point and web, R4

β = Included angle (deg) of valley in star or finocyl grain, BETA

There are three basic types of "star" configurations of interest here. They are the finocyl, the star and the wagon wheel, shown in Figures 78, 79 and 80, respectively. They will now be discussed individually.

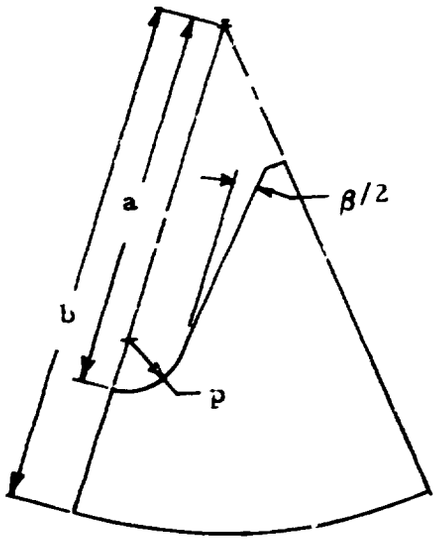


Figure 77. Generalized Configuration

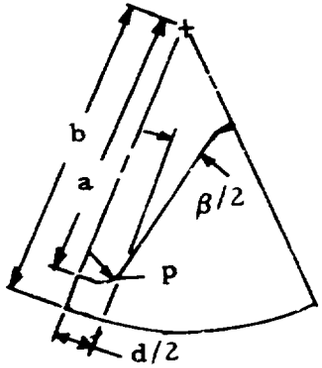


Figure 79. Star Configuration

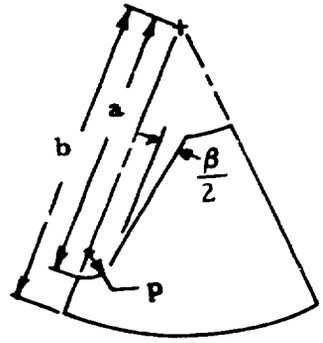


Figure 78. Slotted Tube Configuration

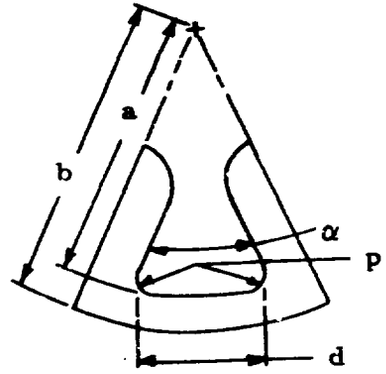


Figure 80. Forked Wagon-Wheel Configuration

Finocyl

The angle β of the slotted tube section in Figure 78 will always be less than 40° because of design practice. Figure 81 shows the dependance of H on β. For β less than 40° the effect on H is small, so for this configuration Eq (9) will be used to calculate H.

As an example, consider a finocyl geometry (Figure 78) with a = 4, b = 5, d = 2p = 1, N = 5 and β = 21°. In this case H is given by Eq (7), H = 11.6789. Then λ = 1.25 and K = 2.10 from Eq (8).

Star

In the star configuration of Figure 79 the angle β will not necessarily be less than 40°. Furthermore, the width of the end of the star valley, d, may be larger than twice the corner radius, p. Modifications to Eq (9) are in order for either or both situations. Consider first the case in which β is larger than 40° and d = 2p. In this case H is given by

$$H = H^* \left(\frac{N}{4}\right)^{-1/3} \left[\frac{1 + 2\sqrt{a/p}}{1 + 2\sqrt{12}} \right] \tag{10}$$

in which H* is selected from the graphs in Figure 81 for the proper β and a/b ratio.

In the case of $d > 2p$ and $\beta \leq 40°$, the H* is selected from Figure 82 which gives the dependance of H on the d/2p ratio, and Eq (10) is used to calculate H.

When $\beta > 40°$ and $d/2p > 1$ concurrently, H* must reflect both facts. Define a factor, F, as

$$F = H^*(d/2p)/H^*(1) \tag{11}$$

where H* (d/2p) = Photoelastic parameter obtained from Figure 82 at the current value of d/2p, HSTAR2

H*(1) = Photoelastic parameter obtained from Figure 82 at d/2p = 1, HSTAR1

Then H is calculated using

$$H = FH^* \left(\frac{N}{4}\right)^{-2/3} \left[\frac{1 + 2\sqrt{a/p}}{1 + 2\sqrt{12}} \right]^2 \tag{12}$$

From: CPIA Publication 214, pg. 3.68

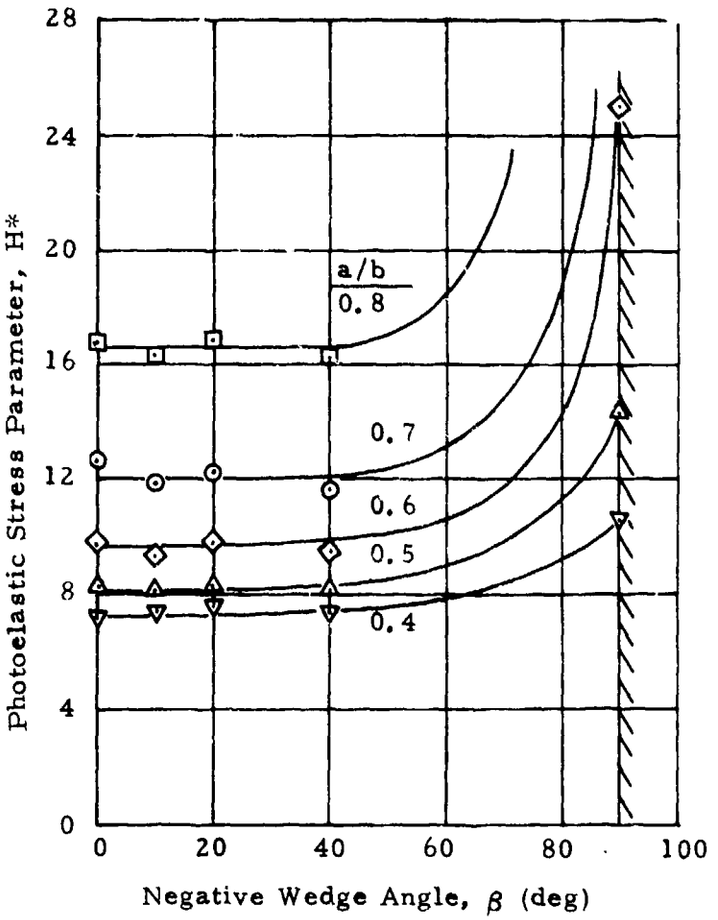


Figure 81. Negative Wedge Angle Test Results, N = 4, a/p = 12

From: CPIA Publication 214, pg. 3.63

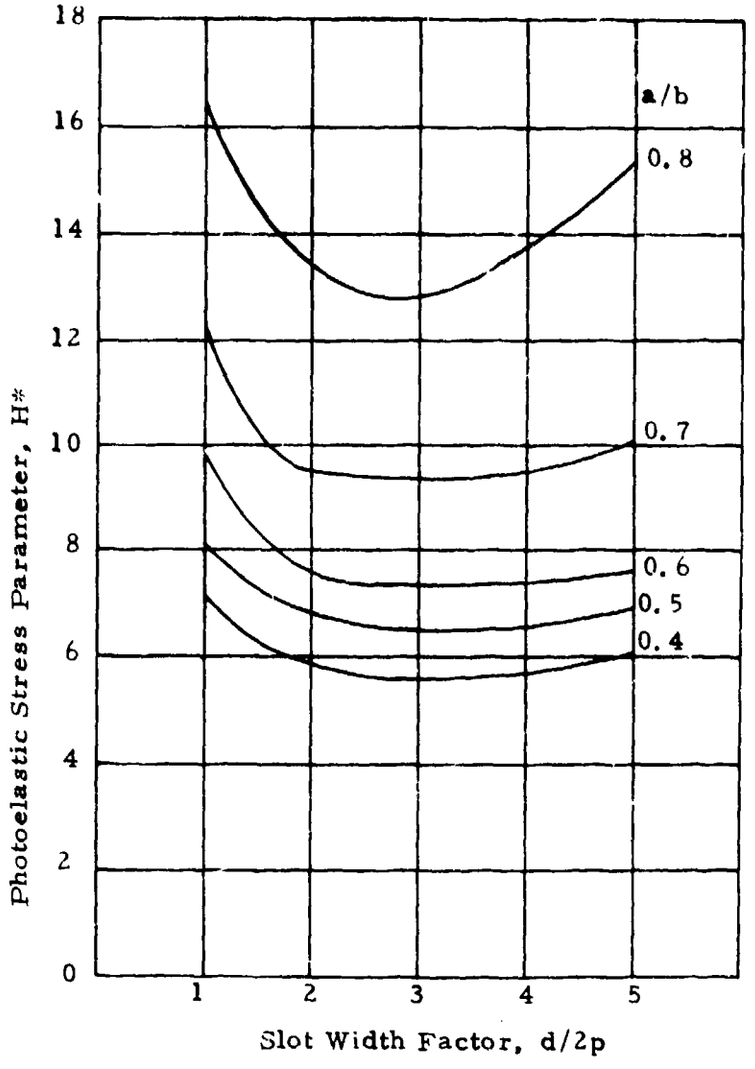


Figure 82. Variation of Parameter H* With Slot Width Factor d/2p, N = 4, a/p = 12

where H* = photoelastic stress parameter, HSTAR, and is obtained from Figure 81 using the current value of β.

Consider another example with d = 1.6, p = 0.5 and the other dimensions the same as in the previous example. H is calculated using Eq (10) with H* obtained from Figure 82. H* = 14.2, H = 11.0683 and K = 1.99. Notice that widening the star width, d, lowers the concentration factor, K.

Wagon Wheel

The effect of the wagon wheel (Figure 80) included angle α on H is shown to be pronounced in Figure 83. Furthermore, d/2p will generally be greater than one, so the factor F in Eq (11) must be utilized with H* given in Figure 83 for the α of the slot to determine H from

$$H = FH^* \left(\frac{N}{4}\right)^{-2/3} \left[\frac{1 + 2\sqrt{a/p}}{1 + 2\sqrt{12}}\right]\left[\frac{1 + 2\sqrt{a/p}}{1 + 2\sqrt{8}}\right] \quad (13)$$

Comparison with Propellant Capability

Nominal strain endurance is furnished by the user as a fixed value or can be calculated with a user-supplied model (see User Models section of this volume). Design strain endurance is derived from the nominal value by accounting for statistical variations in the nominal and the degradation due to aging

$$SEDES = SENOM\left[1 - (3)(CVPS)\right]\left[1 - AGE\right] \quad (14)$$

where SEDES = Design strain endurance (in/in)

SENOM = Nominal strain endurance (in/in)

CVPS = Coefficient of variation of strain endurance (% x 0.01)

AGE = Fraction of propellant strain endurance lost as a result of aging.

After applying a factor of safety to the predicted strain, appropriate margins of safety are calculated

$$MSP = \frac{SEDES}{(FSPS)(EPT)} - 1 \quad (15)$$

$$MSPS = \frac{SEDES}{(FSPS)(EPTS)} - 1 \quad (16)$$

From: CPIA Publication 214, pg. 3.66

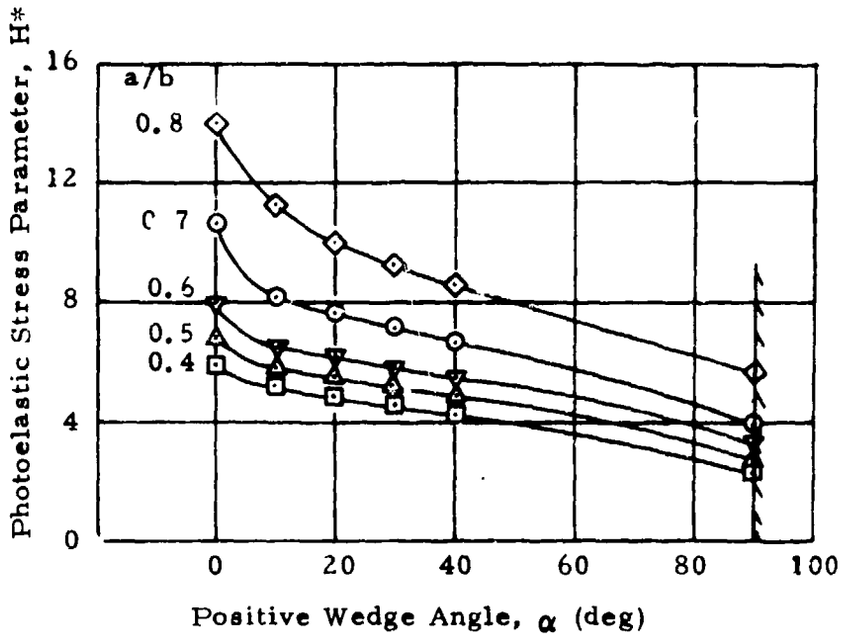


Figure 83. Positive Wedge Angle Test Results, N = 4, a/ρ = 8

where MSP = Margin of safety for storage thermal strain in CP portion of grain

MSPS = Margin of safety for storage thermal strain in slotted portion of grain

FSPS = Factor of safety for propellant thermal strain

If the margins of safety are less than zero, penalties are calculated and summed into the overall penalty function

$$\text{If MSP} < 0.0,\ \text{OBJPS} = (\text{MSP})^2\ 10^5 \tag{17}$$

$$\text{If MSPS} < 0.0,\ \text{OBJPSS} = (\text{MSPS})^2\ 10^5 \tag{18}$$

Strains due to ignition pressurization are compared with a user-supplied maximum limit (EPPMAX) and appropriate penalties calculated

$$\text{If EPP} > \text{EPPMAX},\ \text{OBEPP} = (\text{EPP}-\text{EPPMAX})^2\ 10^5 \tag{19}$$

$$\text{If EPPS} > \text{EPPMAX},\ \text{OBEPPS} = (\text{EPPS}-\text{EPPMAX})^2\ 10^5 \tag{20}$$

Algorithm Summary

The following outlines give the algorithms used to evaluate propellant structural integrity, referring to the equations numbered as above and the accompanying figures, and employing the nomenclature used in the individual grain setup subroutines (see Volume II for definitions). Also shown is the internal code nomenclature from the ballistic simulation module.

Grain Type 1 (Star)

R2 = RFA1-TAUW1 = PLANE (1,4) - PLANE (1,6)
R4 = R5A1 = PLANE (1,10)
RF = RFA1 = PLANE (1,4)
DOVERP = LITD/(2)(R4), where LITD calculated in SETUP1 subroutine

If BETA ≤ 40. and DOVERP = 1.0

SIGTHM	from Eq (2)
SIGPR	from Eq (5)
EPT	from Eq (3)
EPP	from Eq (4)
K	from Eq (8)
EPTS	from Eq (6)
EPPS	from Eq (7)

If BETA ≤ 40 and DOVERP > 1.0

HSTAR	from Figure 82 at DOVERP and R2/RF
SIGTHM	from Eq (2)
SIGPR	from Eq (5)
EPT	from Eq (3)
EPP	from Eq (4)
H	from Eq (10)
K	from Eq (8)
EPTS	from Eq (6)
EPPS	from Eq (7)

If BETA > 40 and DOVERP = 1.0

HSTAR	from Figure 81 at BETA and R2/RF
SIGTHM	from Eq (2)
SIGPR	from Eq (5)
EPT	from Eq (3)
EPP	from Eq (4)
H	from Eq (10)
K	from Eq (8)
EPTS	from Eq (6)
EPPS	from Eq (7)

If BETA > 40 and DOVERP > 1.0

HSTAR2	from Figure 82 at DOVERP and R2/RF
HSTAR1	from Figure 82 at DOVERP = 1 and R2/RF
F	from Eq (11)
HSTAR	from Figure 81 at BETA and R2/RF
H	from Eq (12)
SIGTHM	from Eq (2)
SIGPR	from Eq (5)
EPT	from Eq (3)
EPP	from Eq (4)
K	from Eq (8)
EPTS	from Eq (6)
EPPS	from Eq (7)

Grain Type 2 (Wagon Wheel)

R2 = RFA1-TAUW1 = PLANE (1,4) - PLANE (1,6)
R4 = R5A1 = PLANE (1,10)
RF = RFA1 = PLANE (1,4)
DOVERP = LITD/(2)(R4), where LITD calculated in SETUP2 subroutine
HSTAR2 from Figure 82 at DOVERP and R2/RF
HSTAR 1 from Figure 82 at DOVERP = 1 and R2/RF

F from Eq (11)
HSTAR from Figure 83 at ALPHA and R2/RF
H from Eq (13)
SIGTHM from Eq (2)
SIGPR from Eq (5)
EPT from Eq (3)
EPP from Eq (4)
K from Eq (8)
EPTS from Eq (6)
EPPS from Eq (7)

Grain Type 3 (Finocyl)

For Slotted Region

R2 = R5A1 = PLANE (4, 9)
RF = RFA1 = PLANE (1, 9)
R4 = R4A1 = PLANE (1, 8)
SIGTHM from Eq (2)
SIGPR from Eq (5)
EPT from Eq (3)
EPP from Eq (4)
H from Eq (9)
K from Eq (8)
EPTS from Eq (6)
EPPS from Eq (7)

For CP Region

R2 = R2A10 = PLANE (10, 6)
RF = RFA3 = PLANE (3, 4)
SIGTHM from Eq (2)
SIGPR from Eq (5)
EPT from Eq (3)
EPP from Eq (4)

Grain Type 4 (Conocyl)

R2 = R2A3 = PLANE (3, 6)
RF = RFA3 = PLANE (3, 4)
SIGTHM from Eq (2)
SIGPR from Eq (5)
EPP from Eq (4)

Grain Type 5 (CP)

R2 = R2A5 = PLANE (5, 6)
RF = RFA5 = PLANE (5, 4)
SIGTHM from Eq (2)
SIGPR from Eq (5)
EPT from Eq (3)
EPP from Eq (4)

Figures 81, 82 and 83 are described in the code as Tables, each as an individual subroutine; these subroutines are named FIG5, FIG6, FIG7, respectively, to correspond to a figure numbering system of the Reference 25 report.

A two-dimensional plane-strain model is used to calculate propellant strain due to low-temperature storage and ignition pressurization. Such a model accurately describes the propellant behavior at a point mid-way along the grain length when the grain length-to-diameter ratio (L/D) is equal to or greater than about seven. For L/D < 7, or for locations near the grain terminations, the plane-strain models give very conservative predictions because the end effects (three-dimensional) that relieve the strain are not accounted for in SPOC. Strain predicted for a propellant valley or slot will also be conservative near the ends or for short slots.

The propellant structural analysis is not conservative at the hinge points of stress relief flaps and at the transition between propellant slots and CP regions. Both of these areas represent highly three-dimensional conditions that are not amenable to preliminary design calculations used in SPOC. Consequently, there is the inherent assumption that the bore conditions are the critical locations. Provisions have been made to include volume and weight allowances for stress relief boots ellipsoidal closures, even though their final configuration is dependent on more detailed analyses. The transition section between slots and cylindrical port may require a special configuration to limit imposed strains; another way to achieve the same results is to specify about 7 degrees as the angle on the side of the slot (ALPHA1) of a finocyl grain (Type 3).

Thermal strain in the propellant due to low-temperature storage is compared with design strain endurance (nominal strain endurance reduced for mix-to-mix variations and aging degradation). Strain induced by ignition pressurization is compared with a user-input maximum limit. This latter limit should be derived from tests that measure strain capability at rapid strain rate (to simulate ignition pressurization on test specimens conditional to the design low-temperature and already strained to the level that will be induced by low temperature storage.

TRAJECTORY SIMULATION

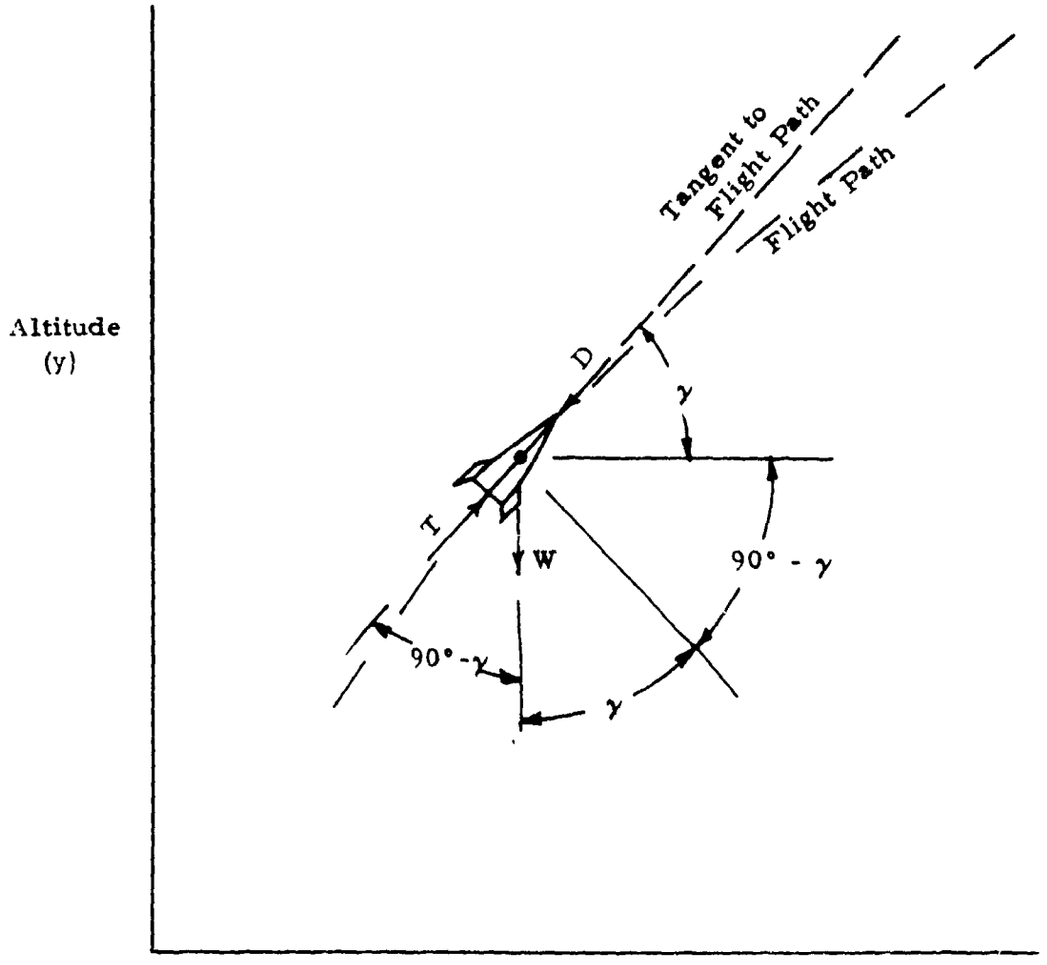
Analytical Relationships

The trajectory simulation is based upon a mathematical model of the flight dynamics of a point-mass missile flying a two-dimensional path in the altitude/range plane over a flat earth. Forces modeled are restricted to thrust, drag, and weight. The ballistic trajectory restriction assumes missile orientation such that lift is always zero. A symmetrical missile is assumed, resulting in angle of attack always being zero. At zero angle of attack, drag and axial aerodynamic force are equal. There is, therefore, no need to differentiate between the two common force-accounting systems (body oriented or flight-path oriented).

The time-dependency of thrust and propellant weight is included via the output of the motor ballistic simulation subroutine. Instantaneous missile weight is taken as launch weight less the integral of motor weight discharge rate. Variation of drag with Mach number is described through a user-generated input table. Provision is made for separate aerodynamic data to be input for power-on and power-off phases. The variation in atmospheric properties with altitude is modeled from the 1959 ARDC STD Atmosphere and the MIL-STD-210A Tropical, Polar, Hot, and Cold Atmospheres. The user shall choose one from among these choices. Required integration of time-dependent parameters is accomplished using a fourth-order Runge-Kutta procedure.

It is assumed that the missile will be air-launched and that no on-launcher kinematics will be included. Launch conditions are specified by altitude, Mach number, and flight-path angle. A pre-boost glide phase is not included, i.e., boost ignition occurs at the instant of launch. Provision is included for two flight phases; (1) rocket thrusting and (2) post-burnout glide. The user has the capability of terminating the trajectory simulation by his command. He may specify a termination upon achieving a selected value for (1) time of flight from launch, (2) time of flight after boost burnout, (3) slant range, (4) horizontal range, (5) altitude (either approaching from above or below), (6) missile Mach number, (7) missile velocity, (8) flight path angle, (9) missile acceleration along the flight path, or (10) range along the flight path. Also termination may be commanded upon ground impact or boost burnout. The boost phase is always completed unless ground impact occurs first. The termination commands apply only at or after boost burnout. Termination upon ground impact will be automatic should this occur before user-commanded termination.

The trajectory analytics are shown on the following pages.



$$Ay = \frac{dVy}{dt} = \left(\frac{g}{w}\right)(T\cos(90-\gamma) - W - D\sin\gamma) \quad (1)$$

$$A_x = \frac{dVx}{dt} = \left(\frac{g}{w}\right)(T\sin(90-\gamma) - D\cos\gamma) \quad (2)$$

$$Vy = \int Ay\,dt + Vy_L \qquad \gamma = \arctan\left(\frac{Vy}{Vx}\right) \quad (3)$$

$$Vx = \int Ax\,dt + Vx_L$$

$$V = \sqrt{Vx^2 + Vy^2} \quad (4)$$

$$Vx_L = V_L\cos\gamma_L \qquad W = W_L - \int \dot{W}_p\,dt \quad (5)$$

$$V_{Y_L} = V_L \sin \gamma_L \qquad T = T_V - PA_{ex} \tag{6}$$

$$SR = \sqrt{(X-X_L)^2 + (Y-Y_L)^2}$$

$$Y = \int Vy\, dt + Y_L \qquad D = C_D\, q\, A \tag{7}$$

$$X = \int Vx\, dt + X_L \qquad q = (\tfrac{1}{2})\, \rho\, V^2 \tag{8}$$

$$M = \frac{V}{C} \tag{9}$$

NOMENCLATURE

A	Aerodynamic reference area for the missile in ft^2.
C	Ambient sonic velocity in ft/sec.
D	Missile drag in pounds.
g	Gravitational constant in ft/sec^2.
M	Missile flight Mach number.
P	Ambient pressure in $pounds/in^2$.
q	Freestream dynamic pressure in lbs/ft^2
t	Time in seconds.
T	Thrust in pounds.
V	Missile velocity in ft/sec.
W	Missile weight in pounds.
X	Range in feet.
Y	Altitude in feet.
ρ	Ambient air density in $\frac{lb\text{-}sec^2}{ft^4}$.
γ	Flight path angle in degrees.

A_y	Vertical acceleration in ft/sec^2.
A_x	Horizontal acceleration in ft/sec^2.
A_{ex}	Nozzle exit area in in^2.
C_D	Missile drag coefficient.
T_v	Vacuum thrust in pounds.
V_L	Missile velocity at launch in ft/sec.
V_X	Horizontal component of missile velocity in ft/sec.
V_Y	Vertical component of missile velocity in ft/sec.
W_L	Missile weight at launch in pounds.
X_L	Missile range at launch in feet.
Y_L	Missile altitude at launch in feet.
γ_L	Missile flight path angle at launch in degrees.
$\dot{W}_p$	Propellant weight flowrate in pounds/sec.
SR	Slant range from launch in feet.
V_{X_L}	Horizontal component of missile velocity at launch in ft/sec.
V_{Y_L}	Vertical component of missile velocity at launch in ft/sec.

Execution Logic

If the user selects FTRAJ=T in namelist CONTRL, a trajectory simulation will be performed. Subroutine TRAJIN reads the user inputs and digests the input data and subroutine TRAJ performs the simulation. TRAJ is called by TRAJIN. If the user is analyzing a two-temperature problem, and his inputs require running the trajectory simulation at both low and high temperatures, then TRAJIN will be called only on the low temperature trajectory simulation (i.e., using the low-temperature thrust history). The variable DELTAV (change in velocity) is used as a flag, and a non-zero value indicates that TRAJIN (and the subsequent low-temperature trajectory simulation) have been run. Figure 84 shows the trajectory decision logic.

If any of the requirements associated with low-temperature trajectory analysis are different from their default values (DELVRQ > 1, TTTRQ < .99E6, or VTRQ > 1), TRAJIN will be called with the low-temperature thrust history tables. If this test falls, NOLO will be set to 1, which indicates no low-temperature requirement has been specified. If the requirement associated with high-temperature trajectory analysis is different from its default value (ACLIM < .99E6), and the ballistics simulation was performed at only one temperature, the flag DELTAV indicates that TRAJIN should be called. If ACLIM < .99E6 and ballistics were run at both low and high temperatures, then the high-temperature thrust tables are used to run the trajectory simulation. DELTAV indicates whether TRAJIN or TRAJ should be called. If the test fails, then NOHI is set equal to 1, which indicates no high temperature requirement.

If the user inputs no low or no high temperature requirements, a trajectory simulation will be performed only on the first and last passes through the program if FTRAJ=T is selected by the user.

Ideal drag-free burnout velocity and axial acceleration are calculated in subroutine FLT if FTRAJ=F (its default value). Velocity at launch is assumed equal to zero. Burnout velocity is calculated at both temperatures in a two-temperature problem, but the low temperature value is compared with the requirement (DELVRQ). Axial acceleration is calculated at high-temperature for comparison with its requirement (ACLIM).

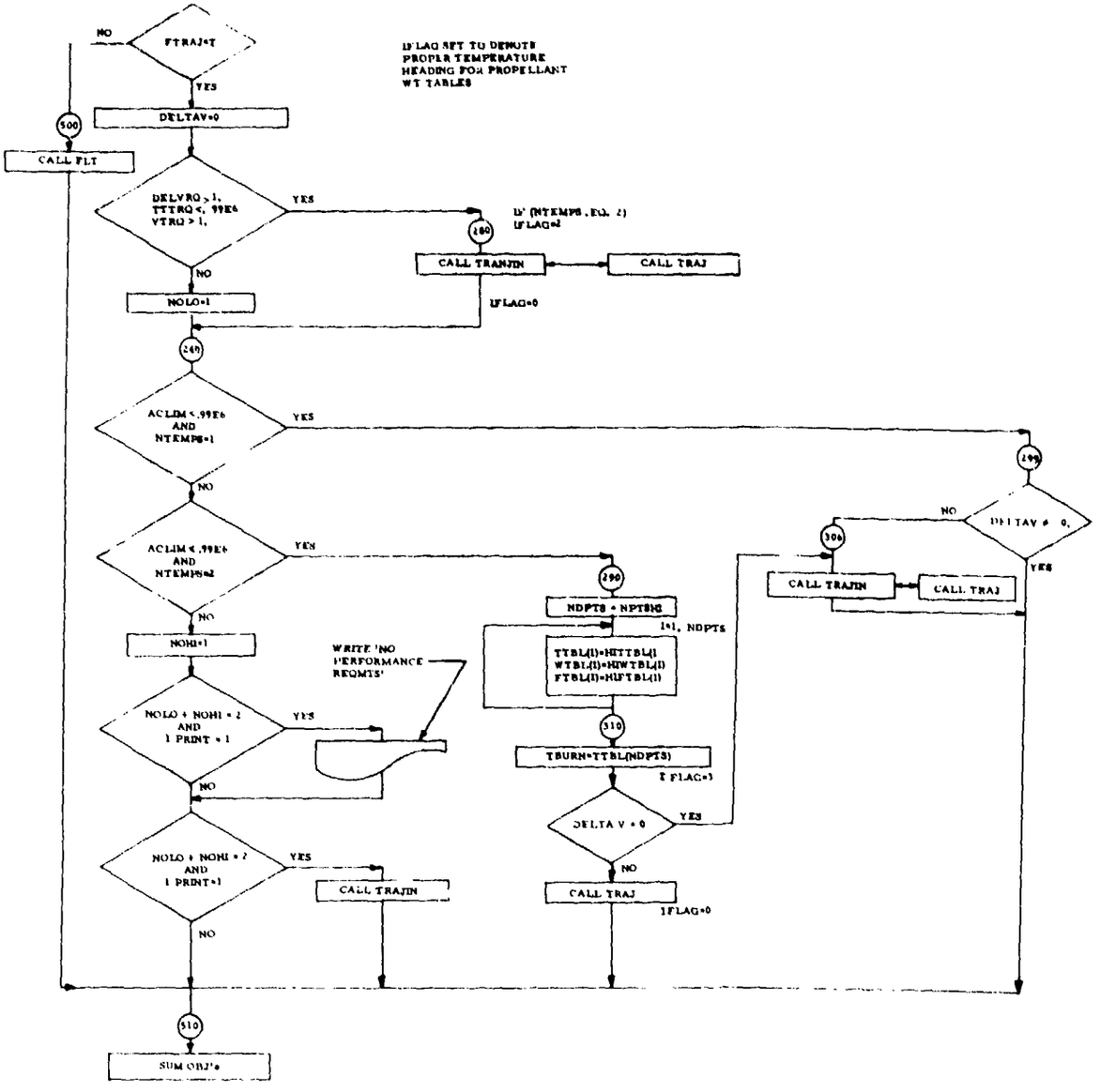


Figure 84. Trajectory Decision Logic

Integration Technique

For those instances in which it will be necessary to integrate time-dependent variables, a fourth-order Runge-Kutta* numerical integration procedure will be employed. This method is widely used because of its many desirable characteristics, including: (1) it is a single-step method and is therefore self-starting, (2) its accuracy is great relative to the independent variable step-size, (3) independent variable step-size may be changed at any time without affecting previous computations, and (4) the method is readily adapted to systems of simultaneous equations where integration in parallel is required.

The necessity for numerical integration in the SPOC computer code arises as a result of the need to perform trajectory simulations. The rocket motor weight discharge rate must be integrated to determine instantaneous missile weight. The missile acceleration must be integrated once to determine velocity and again to determine displacement. The nature of these integrations differ substantially and therefore the application of the Runge-Kutta integration procedure to each will be treated separately.

Rocket Motor Weight Discharge Rate

The rocket motor internal ballistics subroutine will produce a schedule of rocket motor weight discharge rate as a function of time. This complete schedule of motor weight discharge rate with time is known prior to any trajectory integration being performed. The form of these data is a simple stored table of motor discharge rate versus time. The fact that the rocket motor weight discharge rate is independent of missile flight dynamics allows these data to be integrated separately from (i. e., not parallel to) missile flight dynamics and therefore special simplifications apply to the general Runge-Kutta numerical integration procedure. In this special case where the derivative of rocket motor weight discharged with respect to time is a function of time only, the general fourth-order Runge-Kutta numerical integration procedure defaults to the same scheme known as Simpson's Rule. For this special case,

$$\frac{dw_d}{dt} = \dot{w}_d = f(t) \qquad (10)$$

where $\dot{w}_d$ is the time-derivative of rocket motor weight discharged and is the function to be integrated. The result of this integration is the decrease in rocket motor weight (also the decrease in missile weight) to a given point in

*A Basic Course In Numerial Methods; Ralph E. Ekstrom; reprinted from Machine Design; October 26, 1967 through June 20, 1968; Penton Publishing Co.; Cleveland, Ohio 44113

time. In stepping through the numerical integration process from time = t to time = t + Δt,

$$w_{d_{t+\Delta t}} = w_{d_t} + (\Delta t)(\dot{\overline{w}}_d) \tag{11}$$

where

$w_{d_{t+\Delta t}}$ is the rocket motor weight discharged to time = t + Δt

w_{d_t} is the rocket motor weight discharged to time = t

Δt is the numerical integration step size

$\dot{\overline{w}}_d$ is the average value of $\dot{w}_d$ between time = t and time = t + Δt

The numerical integration procedure is concerned with the evaluation of the term $(\Delta t)(\dot{\overline{w}}d)$ in equation (11). For this special case, the fourth-order Runge-Kutta method estimates this term to be

$$(\Delta t)(\dot{\overline{w}}d) = (1/6)(k_0 + 4k_1 + k_2) \tag{12}$$

$$k_0 = (\Delta t)\ \dot{w}_{d_t} \tag{13}$$

$$k_1 = (\Delta t)\ \dot{w}_{d_{t+\frac{\Delta t}{2}}} \tag{14}$$

$$k_2 = (\Delta t)\ \dot{w}_{d_{t+\Delta t}} \tag{15}$$

where

$\dot{w}_{d_t}$ is the rocket motor weight discharge rate at time = t

$\dot{w}_{d_{t+\frac{\Delta t}{2}}}$ is the rocket motor weight discharge rate at time = t + $\frac{\Delta t}{2}$

$\dot{w}_{d_{t+\Delta t}}$ is the rocket motor weight discharge rate at time = t + Δt

The mechanics of the numerical integration procedure consists of a cyclic repetition of the following steps:

(1) The results from the rocket motor internal ballistics subroutine make known the value of $\dot{w}_{d_t}$ at time = t. Select a value for Δt and compute k_0 from equation (13).

(2) The results from the rocket motor internal ballistics subroutine makes known the value of $\dot{w}_{d_{t+\frac{\Delta t}{2}}}$ at time = $t + \frac{\Delta t}{2}$. Then compute k_1 from equation (14).

(3) The results from the rocket motor internal ballistics subroutine makes known the value of $\dot{w}_{d_{t+\Delta t}}$ at time = $t + \Delta t$. Then compute k_2 from equation (15).

(4) Knowing k_0, k_1, and k_2, solve for $w_{d_{t+\Delta t}}$ from equations (11) and (12).

(5) Reset time t to $t + \Delta t$ and return to Step (1). Continue cycling until time becomes equal to motor burn time determined from the rocket motor internal ballistics subroutine.

The above procedure defines rocket motor weight discharged as a function of time from ignition to burnout.

Missile Acceleration

The rocket motor internal ballistics subroutine will produce a schedule of vacuum thrust as a function of time. This complete history of vacuum thrust is known prior to any missile flight dynamics integration being performed. The complete history of motor weight discharged during the burn is also known, this information having been determined as a result of the previously described integration. It follows that the missile weight is known at any instant of time. The integration of missile acceleration is then performed in the following manner:

$$a_x = \frac{dV_x}{dt} = f(t, V_x, V_y) \quad (16) \qquad a_y = \frac{dV_y}{dt} = j(t, V_x, V_y) \quad (17)$$

where

a_x is the missile horizontal acceleration

a_y is the missile vertical acceleration

V_x is the missile horizontal velocity

V_y is the missile vertical velocity

f ( ) is the function defining missile horizontal acceleration and is the system of equations numbered (2), (3), (4), (5), (6), (7), (8), and (9) in the TRAJECTORY section.

j ( ) is the function defining missile vertical acceleration and is the system of equations numbered (1), (3), (4), (5), (6), (7), (8), and (9) in the TRAJECTORY section.

t is the independent variable time

In stepping through the numerical integration process from time = t to time = t + Δt,

$$V_{x_{t+\Delta t}} = V_{x_t} + (\Delta t)(\bar{a}_x) \quad (18) \qquad V_{y_{t+\Delta t}} = V_{y_t} + (\Delta t)(\bar{a}_y) \quad (19)$$

where

$V_{x_{t+\Delta t}}$ is missile horizontal velocity at time = t + Δt

$V_{y_{t+\Delta t}}$ is missile vertical velocity at time = t + Δt

V_{x_t} is missile horizontal velocity at time = t

V_{y_t} is missile vertical velocity at time = t

Δt is the numerical integration step size

$\bar{a}_x$ is the average of a_x between time = t and time = t + Δt

The numerical integration procedure is concerned with the evaluation of the terms $(\Delta t)(\bar{a}_x)$ and $(\Delta t)(\bar{a}_y)$ in equations (18) and (19) respectively. The fourth-order Runge-Kutta numerical integration procedure estimates these terms to be

$$(\Delta t)(\bar{a}_x) = (1/6)(k_0 + 2k_1 + 2k_2 + k_3) \quad (20) \qquad (\Delta t)(\bar{a}_y) = (1/6)(L_0 + 2L_1 + 2L_2 + L3) \quad (21)$$

$$k_0 = (\Delta t)\, f\,(t,\ V_{x_t},\ V_{y_t}) \quad (22) \qquad L_0 = (\Delta t)\, j\,(t,\ V_{x_t},\ V_{y_t}) \quad (23)$$

$$k_1 = (\Delta t)\, f\left(t + \frac{\Delta t}{2},\ V_{x_t} + \frac{k_0}{2},\ V_{y_t} + \frac{L_0}{2}\right) \tag{24}$$

$$L_1 = (\Delta t)\, j\left(t + \frac{\Delta t}{2},\ V_{x_t} + \frac{k_0}{2},\ V_{y_t} + \frac{L_0}{2}\right) \tag{25}$$

$$k_2 = (\Delta t)\, f\left(t + \frac{\Delta t}{2},\ V_{x_t} + \frac{k_1}{2},\ V_{y_t} + \frac{L_1}{2}\right) \tag{26}$$

$$L_2 = (\Delta t)\, j\left(t + \frac{\Delta t}{2},\ V_{x_t} + \frac{k_1}{2},\ V_{y_t} + \frac{L_1}{2}\right) \tag{27}$$

$$k_3 = (\Delta t)\, f\left(t + \Delta t,\ V_{x_t} + k_2,\ V_{y_t} + L_2\right) \tag{28}$$

$$L_3 = (\Delta t)\, j\left(t + \Delta t,\ V_{x_t} + k_2,\ V_{y_t} + L_2\right) \tag{29}$$

where

$f(t, V_{x_t}, V_{y_t})$ is the function f ( ) evaluated at time = t, missile horizontal velocity = V_{x_t}, and missile vertical velocity = V_{y_t}

$j(t, V_{x_t}, V_{y_t})$ is the function j ( ) evaluated at time = t, missile horizontal velocity = V_{x_t}, and missile verticle velocity = V_{y_t}

$f\left(t + \frac{\Delta t}{2},\ V_{x_t} + \frac{k_0}{2},\ V_{y_t} + \frac{L_0}{2}\right)$ is the function f ( ) evaluated at time = $t + \frac{\Delta t}{2}$, missile horizontal velocity = $V_{x_t} + \frac{k_0}{2}$, and missile vertical velocity = $V_{y_t} + \frac{L_0}{2}$

$j\left(t + \frac{\Delta t}{2}, V_{x_t} + \frac{k_0}{2}, V_{y_t} + \frac{L_0}{2}\right)$ is the function j ( ) evaluated at time = $t + \frac{\Delta t}{2}$, missile horizontal velocity = $V_{x_t} + \frac{k_0}{2}$, and missile vertical velocity = $V_{y_t} + \frac{L_0}{2}$

$f\left(t + \frac{\Delta t}{2}, V_{x_t} + \frac{k_1}{2}, V_{y_t} + \frac{L_1}{2}\right)$ is the function f ( ) evaluated at time = $t + \frac{\Delta t}{2}$, missile horizontal velocity = $V_{x_t} + \frac{k_1}{2}$, and missile vertical velocity = $V_{y_t} + \frac{L_1}{2}$

$j\left(t + \frac{\Delta t}{2}, V_{x_t} + \frac{k_1}{2}, V_{y_t} + \frac{L_1}{2}\right)$ is the function j ( ) evaluated at time $t + \frac{\Delta t}{2}$, missile horizontal velocity = $V_{x_t} + \frac{k_1}{2}$, and missile vertical velocity = $V_{y_t} + \frac{L_1}{2}$

$f(t + \Delta t, V_{x_t} + k_2, V_{y_t} + L_2)$ is the function f ( ) evaluated at time = $t + \Delta t$, missile horizontal velocity = $V_{x_t} + k_2$, and missile vertical velocity = $V_{y_t} + L_2$

$j(t + \Delta t, V_{x_t} + K_2, V_{y_t} + L_2)$ is the function j ( ) evaluated at time = $t + \Delta t$, missile horizontal velocity = $V_{x_t} + K_2$, and missile vertical velocity = $V_{y_t} + L_2$

The mechanics of the numerical integration procedure consists of a cyclic repetition of the following steps:

(1) The results of the rocket motor internal ballistics subroutine determine the value of vacuum thrust at any point in time. The previously performed integration of motor weight discharge rate determines missile weight at any instant in time. Selection of a value for Δt will now permit the determination of k_0 and L_0 from equations (22) and (23) respectively.

(2) Knowing k_0 and L_0, compute k_1 and L_1 from equations (24) and (25).

(3) Knowing k_1 and L_1, compute k_2 and L_2 from equations (26) and (27).

(4) Knowing k_2 and L_2, compute k_3 and L_3 from equations (28) and (29).

(5) Knowing k_0, k_1, k_2, k_3, L_0, L_1, L_2, and L_3 compute $(\Delta t)(\bar{a}_x)$ from equation (20) and $(\Delta t)(\bar{a}_y)$ from equatior (21). Knowing $(\Delta t)(\bar{a}_x)$ and $(\Delta t)(\bar{a}_y)$, solve for $V_{x_{t+\Delta t}}$ and $V_{y_{t+\Delta t}}$ from equations (18) and (19).

(6) Reset t to $t + \Delta t$ and return to step (1). Continue cycling until a trajectory termination command is encountered. The result will be a history of horizontal and vertical velocity throughout the missile time of flight.

Missile Velocity

Once the integration of missile acceleration has been performed, a history of missile velocity versus time is known throughout the entire flight. Thus missile velocity is a function of time only and its integration will be a special case similar to that previously described for motor weight discharge rate. As was noted previously, this special case of the fourth-order Runge-Kutta procedure defaults to Simpson's Rule. The integration of missile velocity may be performed either (1) in series with the integration of missile acceleration (that is one after the other) or (2) in parallel with the integration of missile acceleration (that is both at the same time). SPOC performs acceleration and velocity integration in parallel. The integration of missile velocity will define missile translation and is performed in the following manner:

$$\frac{dx}{dt} = V_x = h(t) \qquad (30) \qquad\qquad \frac{dy}{dt} = V_y = b(t) \qquad (31)$$

where

V_x is the missile horizontal velocity

V_y is the missile vertical velocity

h (t) is the fur ction defining missile horizontal velocity as a function of time and is the result of the previously described integration of missile acceleration.

b (t) is the function defining missile vertical velocity as a function of time and is the result of the previously described integration of missile acceleration.

In stepping through the numerical integration process from time = t to time = $t + \Delta t$,

$$X_{t+\Delta t} = X_t + (\Delta t)(\bar{V}_x) \quad (32) \qquad Y_{t+\Delta t} = Y_t + (\Delta t)(\bar{V}_y) \quad (33)$$

where

$X_{t+\Delta t}$ is missile horizontal range at time = $t + \Delta t$

$Y_{t+\Delta t}$ is missile altitude at time = $t + \Delta t$

X_t is missile horizontal range at time = t

Y_t is the missile altitude at time = t

Δt is the numerical integration step size

$\bar{V}_x$ is the average missile horizontal velocity between time = t and time = $t + \Delta t$

$\bar{V}_y$ is the average missile vertical velocity between time = t and time = $t + \Delta t$

The numerical integration process is concerned with the evaluation of the terms $(\Delta t)(\bar{V}_x)$ and $(\Delta t)(\bar{V}_y)$ in equations (32) and (33) respectively. For this special case, the fourth-order Runge-Kutta method estimates these terms to be

$$(\Delta t)(\bar{V}_x) = (1/6)(K_0 + 4K_1 + K_2) \quad (34) \qquad (\Delta t)(\bar{V}_y) = (1/6)(N_0 + 4N_1 + N_2) \quad (35)$$

$$K_0 = (\Delta t)(V_{x_t}) \quad (36) \qquad N_0 = (\Delta t)(V_{y_t}) \quad (37)$$

$$K_1 = (\Delta t)(V_{x_t + \frac{\Delta t}{2}}) \quad (38) \qquad N_1 = (\Delta t)(V_{y_t + \frac{\Delta t}{2}}) \quad (39)$$

$$K_2 = (\Delta t)(V_{x_t + \Delta t}) \quad (40) \qquad N_2 = (\Delta t)(V_{y_t + \Delta t}) \quad (41)$$

where

V_{x_t} is the missile horizontal velocity at time = t (obtained from previous integration of acceleration)

V_{y_t} is the missile vertical velocity at time = t (obtained from previous integration of acceleration)

$V_{x_{t+\frac{\Delta t}{2}}}$ is the missile horizontal velocity at time = $t + \frac{\Delta t}{2}$ (obtained from previous integration of acceleration)

$V_{y_{t+\frac{\Delta t}{2}}}$ is the missile vertical velocity at time = $t + \frac{\Delta t}{2}$ (obtained from previous integration of acceleration)

$V_{x_{t+\Delta t}}$ is the missile horizontal velocity at time = $t + \Delta t$ (obtained from previous integration of acceleration)

$V_{y_{t+\Delta t}}$ is the missile vertical velocity at time = $t + \Delta t$ (obtained from previous integration of acceleration)

The mechanics of the numerical integration procedure consists of a cyclic repetition of the following steps:

(1) The results of the previous integration of missile acceleration determines missile horizontal and vertical velocity at any instant of time. Select a starting time and Δt. Solve for K_0 and N_0 from equations (36) and (37).

(2) Solve for K_1 and N_1 from equations (38) and (39).

(3) Solve for K_2 and N_2 from equations (40) and (41).

(4) Knowing K_0, K_1, K_2, N_0, N_1, and N_2, solve for $(\Delta t)(\overline{V}_x)$ from equation (34) and for $(\Delta t)(\overline{V}_y)$ from equation

(5) Solve for $X_{t+\Delta t}$ from equation (32) and for $Y_{t+\Delta t}$ from equation (33).

(6) Reset time t to $t + \Delta t$ and return to step (1). Continue cycling until a trajectory termination command is encountered. The result will be a complete history of altitude and horizontal range throughout the missile flight.

Integration Step Size Determination

In using any numerical integration procedure, the allowable error at the end of each step determines the interval length. If the interval is smaller than necessary, the number of computational cycles will be unnecessarily great and excessive computer run time will result. If the interval is too large, computational accuracy will suffer. The desired compromise is to select an interval sufficiently large to just avoid exceeding needed accuracy. The needed accuracy now becomes a judgement criterion and must be either user stated or implied within the code. Probably a separate accuracy criteria would be needed for each integrated parameter. The substantial increase in complexity required of variable step size integration logic was not deemed

to be justified. SPOC has been written to use fixed step-size integration logic during powered flight with the step size being a user-input fraction of motor burn time. The integration step-size post-burnout coast flight is also fixed, but at a new value equal to a user-input multiple of the boost powered flight trajectory step size.

COST

Two options are available to estimate costs. One uses the Tri-Services Cost Study results; the other employs a model supplied by the user (see User Model section of this volume). Either option is initiated by setting FCOST = T in the namelist CONTRL. If the Tri-Services model is desired, nothing more is required. If a user model is supplied, CSTMDL = T must also be included in CONTRL.

Tri-Services Cost Model

This option employs the general cost relations for steel-case motors developed in the Tri-Services Cost Study (Reference 18). First unit production cost (BFUCST) of the basic motor is found from

$$BFUCST = (4.493)(WMOTOR)^{0.1306}\ (ISP)^{0.711}\ (1-MF)^{-1.828} \qquad (1)$$

where

WMOTOR	=	Total motor weight (lbm)
ISP	=	Delivered specific impulse, 70°F (lbf-sec/lbm)
MF	=	Motor mass fraction

If a single-temperature problem is being run, ISP70 calculated for that simulation becomes ISP. If a two-temperature problem is being run, ISP is estimated by

$$\ln (ISP) = \ln (ISPHI) - \left[\frac{THI-70}{THI-TLO}\right] \ln (ISPHI/ISPLO) \qquad (2)$$

Then BFUCST is adjusted for the components such as igniters, blast tubes, etc., using the factors in Table 6. The code user must select from Table 6 those multiplicative factors which apply to his particular problem; the product of these individual values are input as MULFAC. Two components are additive factors: igniter and safe-and-arm device; an appropriate sum is input as ADDFAC. Thus, the motor first unit production cost (FUPCST) is

$$FUPCST = (MULFAC)(BFUCST) + ADDFAC \qquad (3)$$

The expected level of production is used to adjust the first unit cost to the average unit cost. Two adjustments are needed: production rate (PRATE) and production quantity (PQUAN). Learning curves of 96% for production rate and 94% for production quantity are employed for this adjustment

$$PRATEF = 1.314\left[(PRATE)^{-0.0589} - (1/PRATE)\right] \qquad (4)$$

$$PQUANF = 1.098\left[(PQUAN)^{-0.0893} - (1/PQUAN)\right] \qquad (5)$$

TABLE 6

PRODUCTION COST FACTORS

Adjustment	Factor
o Case Attachments	
Forward attachment to missile	1.11
Launch lugs, aft of pressure vessel	1.07
Launch lugs, on pressure vessel, integral	1.13
Launch lugs, on pressure vessel, strap-on	1.07
Fin attachment, dovetail, untapered	1.20
Fin attachment, dovetail, tapered	1.30
Fin attachment, folding	1.25
Fin clips, fixed	1.07
o Blast tube	1.09
o Canted nozzle	1.05
o Grain	
Dual thrust, single grain	1.04
Composite smoky	1.00
Composite reduced smoke	0.98
Double base smoky	1.28
Double base minimum smoke	1.44
Dual thrust, dual grains	1.12
High burn rate, greater than 1.5 in/sec	1.12
High burn rate, greater than 3.0 in/sec	1.30
Free standing grain, internal burning	0.80
Free standing grain, internal/external burning	0.80
o Inert slivers	1.07
o External insulation	1.05
o Thrust vector control	
Liquid injection TVC	1.30
Flexible nozzle	1.40
Hot gas bleed	1.40
Warm injection and jet interaction	1.30
Jet vanes	1.35
Jet tabs	1.37

Table 6 (Continued)

PRODUCTION COST FACTORS

Adjustment		Factor
o Boost/Sustain		
	2:1 through 5:1	1.04
	6:1	1.06
	7:1	1.08
	8:1	1.10
	9:1	1.14
	10:1	1.20
o Pulse Mode		
One	One pulse (two grains)	1.33
	Two pulse (three grains)	1.64
	Four pulse (five grains)	1.80
o Thermal cookoff		1.04
o Thrust termination		
	Propellant extinguishment	1.09
	Thrust reversal	1.13
o Wire harness		1.09
o RI filter		1.04
o Igniter (additive)		$336
o Safe/arm (additive)		
	Manual	$462
	Remote	$2217

Thus the average unit production cost is

$$AUPCST=(PRATEF)(PQUANF)(FUPCST) \quad (6)$$

and the total production cost becomes

$$TPCST = (AUPCST)(PQUAN) \quad (7)$$

Costs for development (DEVCST), pre-flight readiness testing (PFRTCS) and qualification (QUALCS) are estimated by

$$DEVCST = ((0.00379)(ISP)(BFUCST) + 261.)\ 1000 \quad (8)$$
$$QUALCS = ((0.000736)(ISP)(BFUCST) + 51.)\ 1000 \quad (9)$$
$$PFRTCS = ((0.000736)(ISP)(BFUCST) + 51)\ 1000 \quad (10)$$

Eq (9) and (10) are invoked only if specified by the user (QUAL = T and PFRT = T, respectively).

The total project cost (COST) is

$$COST = DEVCST + PFRTCS + QUALCS + TPCST \quad (11)$$

This parameter can be used merely as other data by which the user evaluates the design, or by selecting ICHOSE = 1, COST becomes the parameter to be minimized by the optimizer.

User Model

As described in the User Model section, the user builds his own subroutine, including whatever common blocks that are necessary to provide his model with the necessary input from other parts of the code. The parameter COST must be supplied for optimization.

COMBUSTION STABILITY ANALYSIS

Recognition that propulsion system mission failure or degradation can result from the effects of combustion instability has led to increasing emphasis on combustion stability as a design parameter. Accordingly, motor stability was deemed a necessary optimization parameter. The one-dimensional longitudinal Standardized Stability Prediction (SSP) (Ref. 19) was selected for use in this program due to its general acceptance in the combustion community and general agreement with experience (Ref. 20). The stability analysis module is a version of SSP that has been modified to reduce execution time and to include various desirable options and features. The code is tailored for use as an optimization tool and to add two additional combustion response models.

Except as noted in the following discussion, the same philosophy, theory and general coding logic are used as in SSP and will not be repeated in this report. The vast majority of variable names are the same as in SSP, so the coding will largely be familiar to persons experienced with the coding in SSP.

Figure 85 is a diagram of the general organization of the stability analysis code block. Entry to this stability analysis is accomplished by calling subroutine E488M2. The majority of the required data is transferred by way of Common. User input has been minimized by the internal data transfer and by selecting the options considered most appropriate to tactical rocket motors. The only direct user input namelist is STABIN, in which the user may specify the number of modes to be analyzed and the combustion response model to be used. Default values are provided, so even STABIN inputs are not required.

Figure 86 shows the locations of the twenty "sections" used to describe the motor cavity for stability analyses, and their relationships to the four "stations" in the head-end and nozzle-end portions of the grain and to the fourteen "planes" used to define the center portion of the grain. For closure Type 1: (1) Section 1 is the closure itself prior to ignition; (2) Section 2 is essentially non-existant at ignition, but grows in length as propellant is consumed (Figure 86 shows it at an intermediate position); (3) End burning surface at the aft end is part of Section 18. For closure Type 2: (1) Section 1 length does not change during burning; (2) End-burning surfaces are part of Section 2 and 18 at the forward and aft ends, respectively; (3) Sections 2 and 18 increase in length and Sections 3 and 17 decrease in length as propellant is consumed if the grain ends are not inhibited (otherwise their length stays constant). Table 7 gives the sources of all data used in the stability analysis.

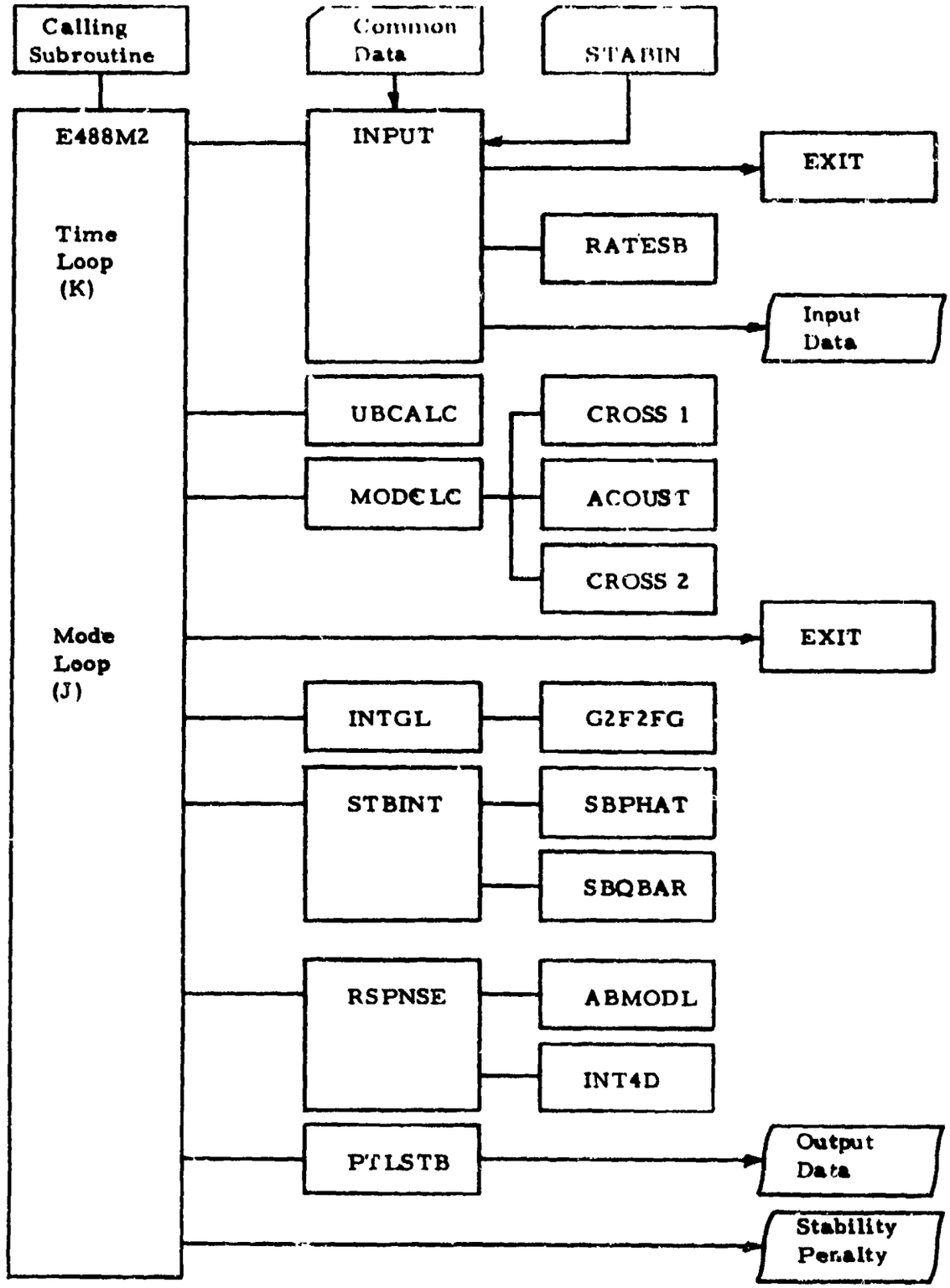
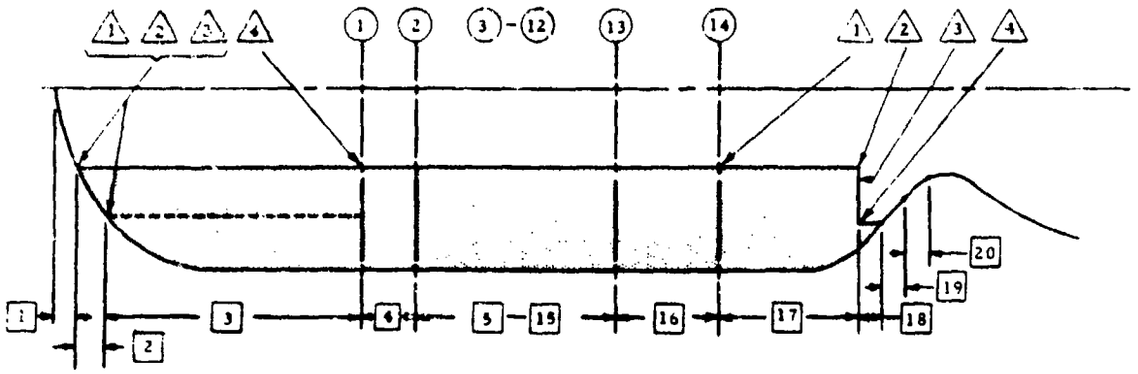
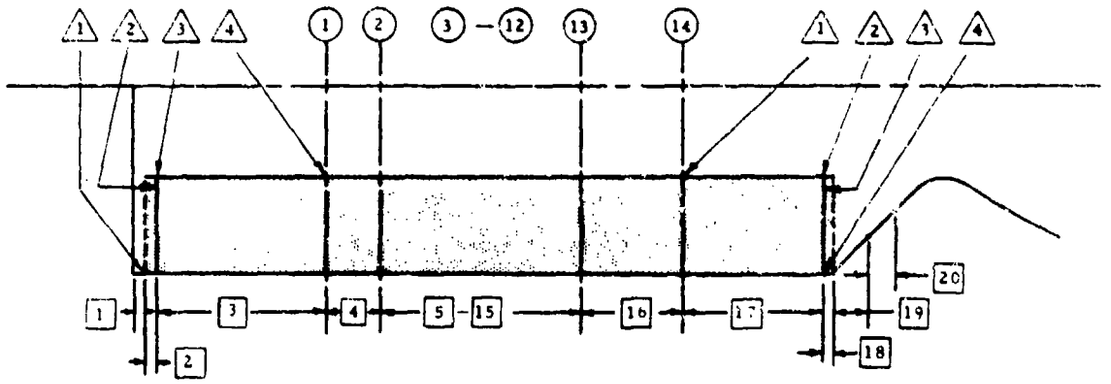


Figure 85. Block Diagram of Combustion Stability Subprogram.

Type 1 Closures



Type 2 (or 3) Closures



○ Ballistic Plane Numbers (I=1, NOPLNS; NOPLNS = 14)
□ Stability Section Numbers (I=1, NSEC; NSEC = 20)
△ Ballistic Station Numbers for End Closures

Section 19 to Mach 0.2 in nozzle
Section 20 to Mach 0.5 in nozzle

Figure 86. Location of Stability Section and Ballistic Planes

TABLE 7

SOURCE OF STABILITY ANALYSIS DATA

Parameter	Closure Type	NSEC 1	2	3	4 - 16	17	18	19	20
TPLOCL	1,2,3	PI'	PH	(1)	(2)	(3)	NZP4	(55)	(55)
TXL	1	0.001	(4)	(5)	(6)	(7)	(8)	(57)	(57)
	2,3	LGAPF	(9)	(10)	(6)	(11)	(12)	(57)	(57)
TS1	1	APHED	APHED	APHED	(13)	PLANE(14, 73)	APNOZ	(14)	(55)
	2,3	(15)	(15)	APHED	(13)	PLANE(14, 73)	(16)	(16)	(55)
TS2	1	APHED	APHED	APHED	(17)	PLANE(14, 73)	(16)	(55)	(55)
	2,3	(15)	(15)	APHED	(17)	PLANE(14, 73)	(16)	(16)	(55)
TSBL	1	0.0	0.0	(54)	(18)	ASINTN	0.0	0.0	0.0
	2,3	0.0	0.0	(54)	(18)	ASINTN	0.0	0.0	0.0
TSBE1	1	0.0	0.0	0.0	0.0	0.0	(19)	0.0	0.0
	2,3	0.0	0.0	0.0	0.0	0.0	(19)	0.0	0.0
TSBE2	1	0.0	HDASE	0.0	0.0	0.0	0.0	0.0	0.0
	2,3	0.0	HDASE	0.0	0.0	0.0	0.0	0.0	0.0
TQ1	1	0.0	0.0	PLANE(1, 72)	(20)	PLANE(14, 72)	0.0	0.0	0.0
	2,3	0.0	0.0	PLANE(1, 72)	(20)	PLANE(14, 72)	0.0	0.0	0.0
TQ2	1	0.0	0.0	PLANE(1, 72)	(22)	PLANE(14, 72)	0.0	0.0	0.0
	2,3	0.0	0.0	PLANE(1, 72)	(22)	PLANE(14, 72)	0.0	0.0	0.0
TA	1,2,3	(24)	(24)	(25)	(26)	(27)	(28)	(55)	(55)
TRLOCL	1	0.0	0.0	(29)	(30)	(31)	(33)	0.0	0.0
	2,3	0.0	(32)	(29)	(30)	(31)	(33)	0.0	0.0
TMACH	1,2,3	0.0	0.0	(34)	(35)	(36)	(37)	(55)	(55)
TRHO	1,2,3	(38)	(39)	(40)	(41)	(42)	(43)	(55)	(55)
TXLL	1	0.001	(44)	(45)	(46)	(47)	(48)	(57)	(57)
	2,3	LGAPF	(49)	(50)	(51)	(52)	(53)	(57)	(57)

Table 7 (contd.)

Source of Stability Analysis Data

NOTES

(1) $\sqrt{PH*HDP4}$

(2) $\sqrt{PLANE(I,77) * PLANE(I-1,77)}$, I = 2, NOPLUS (NOPLNS = 14)

(3) $\sqrt{NZP1 * NZP2}$

(4) Initial = 0.001; Intermediate = TAUHED; Final = XBOREF

(5) (54)/(21)

(6) PLANE(I,3) - PLANE(I-1,3), I = 1,14

(7) Initial = TAUMXA; Intermediate = TAUMXA-TAUNOZ; Final = 0.001

(8) Initial = XRNOZ3-XRNOZ5; Intermediate = XRNOZ3-XRNOZ5 + TAUNOZ
Final = XRNOZ3-XRNOZ5+TAUMXA

(9) Initial = 0.001; Intermediate = 0.001 + TAUHED; Final = TAUMXF. See Note 56

(10) Initial = TAUMXF; Intermediate = TAUMXF - TAUHED; Final = 0.001. See Note 56

(11) Initial = TAUMXA; Intermediate = TAUMXA - TAUNOZ; Final 0.001. See Note 56

(12) Initial = 0.001; Intermediate = 0.001 + TAUNOZ; Final = TAUMXA. See Note 56

(13) PLANE(I,73), I = 1,13

(14) # - RN1**2

(15) # * RFA1**2

(16) # * RFA14**2

(17) PLANE(I+1,73), I = 1,13

Table 7 (Contd.)

Source of Stability Analysis Data

Notes (contd.)

(18) $((PLANE(I, 72) + PLANE(I+1, 72))/2.0) * (PLANE(I, 3) - PLANE(I-1, 3)),\ I = 1, 13$

(19) ASN-ASINTN

(20) $PLANE(I, 72),\ I = 1, 13$

(21) Not used

(22) $PLANE(I+1, 72),\ I = 1, 13$

(23) Not used

(24) $\sqrt{386*GAMAC*RGAS*HDT1} = CA \sqrt{HDT2}$

(25) $CA \sqrt[4]{HDT2 * HDT4}$

(26) $CA \sqrt[4]{PLANE(I, 78) * PLANE (I+1, 78)}\ ,\ I = 1, 13$

(27) $CA \sqrt[4]{NZT1 * NZT2}$

(28) $CA \sqrt{NZT4}$

(29) $\sqrt{HDRT2*HDRT4}$

(30) $\sqrt{PLANE(I, 83) * PLANE (I+1, 83)}\ ,\ I = 1, 13$

(31) $\sqrt{NZRT1 * NZRT2}$

(32) $\sqrt{HDRT1 * HDRT2}$

(33) $\sqrt{NZRT3 * NZRT4}$

(34) $0.5 * (HDU2 + HDU4)/(25)$

(35) $\sqrt{PLANE(I, 80) * PLANE(I+1, 80)}/(26)\ ,\ I = 1, 13$

(36) $\sqrt{PLANE(14, 80) * NZU2}/(27)$

Table 7 (contd.)

Source of Stability Analysis Data

Notes (contd.)

(37) NZU4/(28)

(38) HDDLT1/386.0

(39) $\sqrt{\text{HDDLT1} * \text{HDDLT2}}/386.0$

(40) $\sqrt{\text{HDDLT2} * \text{HDDLT4}}/386.0$

(41) $\sqrt{\text{PLANE}(I, 79) * \text{PLANE}(I+1, 79)}/386.0$, I = 1,13

(42) $\sqrt{\text{NZDLT2} * \text{NZDLT2}}/386.0$

(43) NZDLT4/386.0

(44) 0.001 + (4)

(45) (44) + (5)

(46) (45) + (6)

(47) (46) + (7)

(48) (47) + (8)

(49) LGAPF + (9)

(50) (49) + (10)

(51) (50) + (6)

(52) (51) + (11)

(53) (52) + (12)

(54) ASH-HDASE

(55) Derived from isentropic relationships at the first nozzle station downstream of Mach = 0.2 (Section 19) and Mach = 0.5 (Section 20)

(56) If FWDINH = F, TXL varies with time as shown. If FWDINH=T, TXL is constant at initial values shown.

(57) TXL (and hence, TXLL) determined from spacing of nozzle analysis stations.

Table 7 (contd.)

Source of Stability Analysis Data

DEFINITIONS

ASH	Total cumulative burning surface area (sq in) on head-end portion of grain (Station 1 through 4).
ASINTN	Lateral burning surface area (sq in) between ballistic stations 1 and 2 on nozzle-end of grain.
ASN	Total cumulative burning surface area (sq in) on nozzle-end portion of grain Stations 1 through 4).
GAMAC	Ratio of specific heats in chamber.
HDASE	End burning surface area (sq in) in forward portion of grain (Station 1 through 2).
HDDLT1	Combustion gas density (lbm/cu in) at Station 1 in head-end portion of grain.
HDDLT2	Combustion gas density (lbm/cu in) at Station 2 in head-end portion of grain.
HDDLT4	Combustion gas density (lbm/cu in) at Station 4 in head-end portion of grain.
HDP4	Static pressure (psia) at head-end Station 4.
HDRT1	Instantaneous propellant burn rate (in/sec) at Station 1 of head-end portion of grain.
HDRT2	Instantaneous propellant burn rate (in/sec) at Station 2/3 of head-end portion of grain.
HDRT4	Instantaneous propellant burn rate (in/sec) at Station 4 of head-end portion of grain.
HDT1	Combustion gas static temperature (°R) at Station 1 of head-end portion of grain.
HDT2	Combustion gas static temperature (°R) at Station 2/3 of head-end portion of grain.
HDT4	Combustion gas static temperature (°R) at Station 4 of head-end portion of grain.
HDU2	Combustion gas velocity (ft/sec) at Stations 2/3 in head-end portion of grain.
HDU4	Combustion gas velocity (ft/sec) at Station 4 in head-end portion of grain.
LGAPF	Length (in) between forward face of grain and aft face of insulation on Forward Closure Types 2 or 3. See Figure 15.

Table 7 (contd.)

Source of Stability Analysis Data

Definitions (contd.)

NOPLNS	Number of planes describing propellant grain (14).
NZDLT1	Combustion gas density (lbm/cu in) at Station 1 in nozzle-end portion of grain.
NZDLT2	Combustion gas density (lbm/cu in) at Station 2 in nozzle-end portion of grain.
NZDLT4	Combustion gas density (lbm/cu in) at Station 4 in nozzle-end portion of grain.
NZP1	Static pressure (psia) at nozzle-end Station 1.
NZP2	Static pressure (psia) at nozzle-end Station 2.
NZP4	Static pressure (psia) at nozzle-end Station 4.
NZRT1	Instantaneous propellant burn rate (in/sec) at Station 1 of nozzle-end portion of grain.
NZRT2	Instantaneous propellant burn rate (in/sec) at Station 2/3 of nozzle-end portion of grain.
NZRT4	Instantaneous propellant burn rate (in/sec) at Station 4 of nozzle-end portion of grain.
NZT1	Combustion gas static temperature (°R) at Station 1 of nozzle-end portion of grain.
NZT2	Combustion gas static temperature (°R) at Station 2/3 of nozzle-end portion of grain.
NZT4	Combustion gas static temperature (°R) at Station 4 of nozzle-end portion of grain.
NZU2	Combustion gas velocity (ft/sec) at Station 2 in nozzle-end portion of grain.
NZU4	Combustion gas velocity (ft/sec) at Station 4 in nozzle-end portion of grain.
PH	Head-end stagnation pressure (psia).
PLANE(I,3)	Distance (in) from Plane 1 to Plane I (I=1,14).
PLANE(I,72)	Perimeter (in) of port at Plane (I), I = 1,14.
PLANE(I,73)	Port area (sq in) at particular plane (I = 1,14).
PLANE(I,77)	Static pressure (psia) at Plane I (I=1,14).
PLANE(I,78)	Combustion gas static temperature (°R) at Plane I, I = 1,14
PLANE(I,79)	Combustion gas density (lbm/cu in) at Plane I, I = 1,14.

Table 7 (contd.)

Source of Stability Analysis Data

Definitions (contd.)

PLANE(I, 80)	Combustion gas velocity (ft/sec) at Plane I, I = 1, 14.
PLANE(I, 83)	Instantaneous propellant burn rate (in/sec) at given plane, I = 1, 14.
RFA1	Outside propellant radius (in) at Plane 1.
RFA14	Outside propellant radius (in) at Plane 14.
RGAS	Gas constant of combustion products in chamber (ft-lbf/lbm-°R)
RNOZEN	Radius (in) of aft case opening. See Figure 17.
RN1	Radius (in) at entrance to nozzle. Equals RNOZEN.
TA	Acoustic velocity (in/sec).
TAUHED	Distance (in) burned on head-end region of grain (forward of Plane 1).
TAUMXA	Maximum distance burned (in) at Plane 14. See Figures 18 and 19.
TAUMXF	Maximum distance burned (in) at Plane 1. See Figures 14 and 15.
TAUNOZ	Distance (in) burned on nozzle-end region of grain (aft of Plane 14).
TMACH	Average Mach number in section.
TPLOCL	Local static pressure (psia).
TQ1	Perimeter (in) of lateral burning surface area at forward end of section.
TQ2	Perimeter (in) of lateral burning surface area at aft end of section.
TRHO	Average density (lbf-sec^2/in^4) in section.
TRLOCL	Average burn rate (in/sec) in section.
TSBE1	End burning surface area (sq in) on forward end of section.
TSBE2	End burning surface area (sq in) on aft end of section.
TSBL	Lateral burning surface area (sq in) of section.

Table 7 (contd.)

Source of Stability Analysis Data

Definitions (contd.)

TS1	Port area (sq in) at upstream end of particular section.
TS2	Port area (sq in) at downstream end of particular section.
TXL	Length (in) of section.
TXLL	Cumulative total of section lengths (TXL) measured to aft end of section (in).
XBOREF	Axial distance (in) from tangent point of Type 1 forward closure and case forward to where propellant intersects forward closure. See Figure 14.
XRNOZ3	See Figure 16 (in).
XRNOZ5	See Figure 16 (in).

E488M2

This is the control subroutine. It was derived from SSP/MAIN. SSP capabilities and subroutines deleted (because not needed) were

SSP/TABUL	Mode shape data print
SSP/IPD	Particle damping
SSP/IWD	Wall damping
SSP/IDC	Distributed combustion
SSP/GFTERM	Flandro term
SSP/HYMOK	Effect of high Mach numbers
SSP/NLVC	Nonlinear velocity coupling
SSP/PTNLVC	Nonlinear velocity coupling print
SSP/MODCUP	Mode coupling

Although SSP/TABUL was deleted, UBAR is printed as MACH in INPUT and the effects of mode shape may be examined by examination of the partial integrals DAPL, DAPE, DALV and DAFT added and printed in PTLSTB. Particle damping and distributed combustion were deleted on the arguments that most current tactical rocket motors employ reduced or minimum smoke propellants with small concentrations of particulates. Wall damping was also deleted for small effect; most tactical motors have minimal non-burning surfaces. Mode coupling, high-Mach effects and the Flandro correction were dropped because of their controversial status. Nonlinear velocity coupling was deleted on the basis that the nonlinear effects would be negligible for motors with optimized (high) stability. However, in connection with this last decision, the linear coupling (now calculated in STBINT) was changed to calculate non-zero partial integrals only for sections with average velocity greater than the erosive burning threshold velocity specified in the ballistic analysis[1].

The logic flow of E488M2 differs little from SSP/MAIN. The remaining stability integrals are all calculated in STBINT. The response calculations have been removed from PTLSTB and now constitute subroutine RSPNSE.

The job stacking capability of SSP has been replaced with a formal time loop (K = 1, NTIMES). The internal variable NTIMES is selected by the calling program. For the initial and final stability calculations with full print-out, NTIMES = 20. The times are internally selected to furnish the data required for stability analysis (principally via common/TIMDAT/) at five-percent intervals of propellant weight burned, from 0% through 95%.

---

[1] Erosive burning threshold velocity determined from critical Mach number (MCRIT), which is user input or internally calculated. MCRIT corresponds to code internal designation of KR3.

The use of weight burned avoids the requirement for pre-knowledge of burn times and thereby evaluates all designs on a more consistent basis. This tactic provides thorough mapping of stability parameters for any pressure-time history without requiring user input. For PATSH usage, NTIMES is reduced to 4, with the times selected to furnish the internal stability inputs at weight-burned fractions of 5%, 25%, 55%, and 95%. The uneven spacing provides even coverage for either level-thrust or boost-sustain designs. This spacing also matches the theoretical trends of the individual stability integrals due to the fractional change ratio of internal volume, port area or Mach number by shifting the time points earlier in burn where the change rates are greatest.

The usual error-squared type of penalty function usually associated with PATSH has been replaced with the one-sided penalty function OBSTAB, shown in Figure 87. The argument HMIN is calculated in PTLSTB as the minimum value of H for all modes analyzed at all times during burn. where H =- ALPHA/(2*PI*FREQUENCY). In this context, H positive is good (stable) and H negative is bad (unstable). Recent work on flow-driven oscillations uses H as a measure of resonant gain (Ref 21), and therefore, of the amplitudes of oscillations as well as of stability. The penalty function OBSTAB is divided into three regions: unstable (HMIN < 0.00001), extremely stable (HMIN > 0.2) and the region in between where optimization is useful. The unstable region is clearly unacceptable and large values of OBSTAB will drive the design rapidly toward stability.

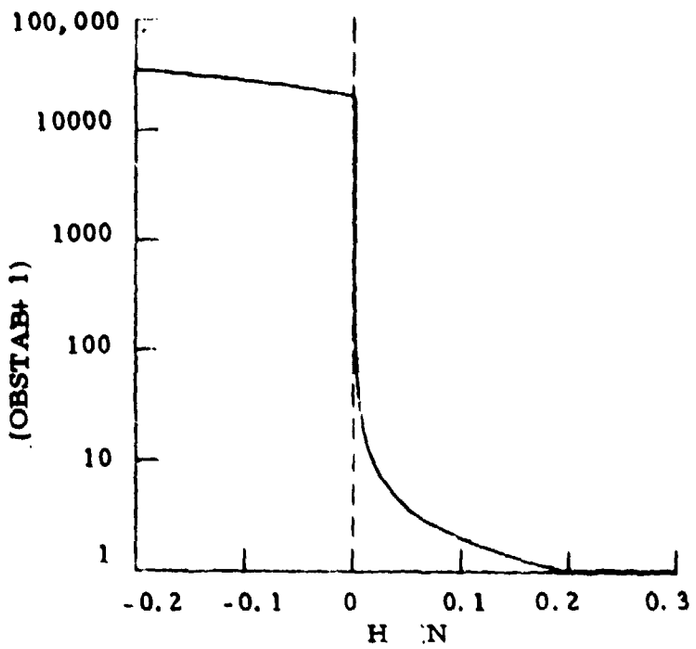
In the extremely stable region, improved stability would only require compromise of the other parameters, so OBSTAB remains constant at zero. A hyperbolic function in between rapidly drives the design to more than neutral stability, but with rapidly decreasing stress on other parameters as stability approaches "rock stable".

INPUT(K)

The input subroutine performs five functions

- o Read/write user inputs via namelist STABIN
- o Default and/or diagnose inconsistent STABIN inputs
- o Transcribe the internally transferred/TIMDAT/data to SSP data arrays without the time subscript
- o Perturb local pressure and Mach number to calculate pressure and velocity exponents
- o Write the input data set used for each time point

The read and write functions are controlled by IREAD and IPRINT transferred from the calling program. Namelist STABIN permits the user to specify the number of modes to be analyzed (NMODE) and to specify response functions. The default value of NMODE is four . This value is



HMIN ≥ 0.2 OBSTAB = 0.0

10^{-6} <HMIN<0.2 $OBSTAB = \frac{0.2}{HMIN} - 1.0$

HMIN<10^{-6} OBSTAB = 19999(HMIN + 1.00001)

HMIN = Minimum H at any time for any mode during simulation

$H = \frac{-ALPHA}{(2)(\pi)(FREQ)}$

ALPHA = Sum of gains and losses

FREQ = Mode frequency at given time

Figure 87. Stability Penalty Function (OBSTAB)

consistent with the general observation that longitudinal stability problems, if any, will usually arise only in the lower modes. The value also exceeds the longitudinal stability analysis requirements usually found in AFRPL procurements. Although occasional higher mode oscillation problems have been encountered, it is expected that NMODE = 4 will be adequate for all but extreme cases.

The response function inputs and defaults are discussed in subroutine RSPNSE. The pressure and velocity exponents are discussed in the section on subroutine RSPNSE.

Although the formats have been changed and additional data printed, the printout from INPUT closely follows SSP.

UBCALC

This subroutine calculates coefficients of burning perimeter and port area variation with length and calculates the volume current (CAPQ) at the upstream end of each element. Except for deleting the wall and particle damping sections and streamlining, the subroutine is virtually identical to SSP/UBCALC.

Usage of SSP for submerged nozzles, with NSEC = 20 (and NNOZ <20) is believed to have resulted in numerical error due to the (J+1) subscripts in the 'DO 80-- 'loop in UBCALC. That problem has not been addressed in the present program, in that submerged nozzles are not available in SPOC.

MODCLC, CROSS1, CROSS2, ACOUST

These subroutines solve the acoustic flow equations for the standing-wave mode frequencies using an iterative search method. They are derived from SSP/MODCLC, SSP/CRSVDX, SSP/CRSVD2 and SSP/ACOUST, respectively. Particle damping and the SSP print option NDUMP1 were deleted. The call arguments were changed by adding commons. This was the only section of the stability analysis in which appreciable difficulty arose when single precision arithmetic (24-bit floating-point mantissa) was used. Occasionally, the search would locate a mode frequency, but be unable to satisfy the nozzle boundary condition of U=0 and finally skip on to the next mode. The actual test argument (UAZERO) was 1×10^{-7}. When the test was relaxed to 1×10^{-5}, no more difficulties were encountered. However, errors in the fifth significant figure of stability integrals were noted in changing from 56-bit to 24-bit mantissas. Consequently, the user of computers with more than 24-bit mantissas should examine the effect of restoring UAZERO to its SSP value. UAZERO is specified in a data statement at the beginning of CROSS2 and is accompanied by appropriate comments.

INTGL, G2F2FG

These subroutines calculate the mode amplitude integrals required in STBINT to calculate the stability integrals. INTGL was derived from SSP/INTGRL by deletion of wall and particle damping integrals and calculating several SSP function subprograms in the code stream. The remaining SSP function subprograms were combined into a single subroutine G2F2FG.

STBINT, SBPHAT, SBQBAR

Subroutine STBINT combines the SSP stability integral subroutines SSP/IPCLAT, SSP/IPCEND, SSP/ILNVC, SSP/IFT and SSP/IND. Subroutines SBPHAT and SBQBAR are very mild revisions of SSP/SBPHAT and SSP/SBUBAR, respectively. As a group, these subroutines calculate and store each required stability integral over the length of each geometry section. This tactic permits variation of the combustion response functions along the length of the motor (in subroutine RSPNSE) and also permits examination of the contribution of each section to the overall stability margin. The partial integrals are printed in subroutine PTLSTB.

RSPNSE, ABMODL, INT4D

Subroutine RSPNSE was created to calculate combustion response in each section of the motor. It replaces the tabular input/interpolation method included in SSP/PTLSTB. Four combustion response models may be specified by the response option IRSPNS in namelist STABIN:

IRSPNS =	1	Tabular input as in SSP
	2	Analytical model due to Culick (Ref 22)
	3	Empirical model due to Cohen (Ref 23)
	4	Empirical model due to Hessler (Ref 24)

Provisions have been made for insertion of a user-defined response model; instructions are included in comments at the end of RSPNSE. The user's model would be inserted into RSPNSE and the permissible IRSPNS test changed in INPUT.

The tabular input option assumes that response is constant in all sections of the motor with non-zero RBAR. The tabular input in namelist STABIN is used to interpolate values of RPLAT, RPEND and RV at the mode frequency with FRES as abscissa. This was the only option in SSP. The interpolation subroutine SSP/INT4D was replaced with ITERP1 to provide extrapolation. This was necessary to avoid error when the user-supplied FRES did not span all calculated mode frequencies.

Response options 2, 3, and 4 assume that the normalized combustion response (CRPOVN) varies as a function of burning rate (RBAR) along the length of the grain. Although only pressure-coupled response was modeled in References 22 and 23, the models were extended to velocity-coupled response using the methods of Reference 24.

The essential assumption for the extension is that the combustion response to heat-flux variations is largely controlled by the solid propellant. Representing the combustion response as a complex variable, it follows that the response normalized by the appropriate exponent is constant, regardless of the source of the heat-flux variations. Reference 24 proposed that the appropriate exponents satisfy the equation.

$$r = a(P)^{n_p}(U)^{n_u}$$

and could consequently be evaluated from erosive burning expressions. This notion was implemented in INPUT by perturbation of the burning rate subroutine (RATESB or USERRB) used in the ballistic analysis for the pressure/velocity field in each section of the motor. The resulting exponents were added to the printout in PTLSTB as ENP, ENU and (for end-burning surfaces) EN0.

It should be noted that the perturbation necessarily defines the existence and magnitude of a velocity coupling threshold (VT) if the burning rate subroutine includes a threshold (either explicit or implicit) for erosive burning. Subroutine RATESB, for example, contains two thresholds. One threshold is explicit (KR3) and one implicit, depending on the relationship between local pressure (P), acoustic velocity (A), and the burn rate inputs KR1, KR2, KR5, and KR6[1]

```
VT = A*((KR1/KR5)**(1/KR6))*(P**((KR2/KR6)-1)).
```

The lower of the two threshold conditions controls.

Reference 24 points out that the velocity response defined in that fashion (RU) differs from the velocity response defined for the SSP stability analysis (RV) by a factor of the Mach number:

$$RV = RU/XMACH$$

Although turbulence effects on velocity coupling are not explicitly included, this method does include turbulence effects implicit in the erosive burning representation in the burning rate subroutine.

The final step in extending the models from pressure coupling to velocity coupling is use of the observation that the pressure-coupled response function required by the SSP analysis mathematics is the real part of the complex combustion response due to pressure fluctuations. Similarly, the velocity-coupled response function required is the imaginary part of the

---

[1] KR1 = A70, AHI, or AL0, depending on grain temperature; KR2 = XN; KR5 = MPCOEF; KR6 = MPEXP

complex combustion response due to velocity fluctuations. The combined representation in RSPNSE, consequently predicts interactive effects between pressure coupling and velocity coupling due to the variation of burning rate, pressure and velocity along the length of the motor.

Response option 2 uses the two-parameter analytical combustion response model form:

$$Rp/n = AB/(\lambda + (A/\lambda) - (1+A) + AB)$$

derived by Culick from the work of several modelers (Ref 22). Lambda (λ) is a complex variable function of the dimensionless frequency (OMEGA, Ω) determined by frequency (F), thermal diffusivity (DFUSVT) and burning rate (RBAR):

```
OMEGA = 2. * PI * F * DFUSVT/(RBAR * RBAR).
```

The model is implemented in subroutine ABMODL using complex variables. For user computers with Fortran compilers that may not support complex arithmetic, the needed real-variable coding is also supplied in comment cards in ABMODL with comment notes for implementation. Response option 2 requires specification of the parameters (APARAM and BPARAM) and DFUSVT in namelist STABIN.

Response options 3 and 4 each assume that the magnitude and frequency of peak response are related to ammonium perchlorate (AP) oxidizer sizes (DIAAP), total AP concentration (CONCAP) and concentration of each size (CONCD):

```
MPEAK = FACPK * CONCAP * DIAAP ** PWR OF D
FPEAK = FCONST * RBAR/DIAAP.
```

It is also assumed that the relation between F and OMEGA is constant, so

```
OMEGA = OMEGPK * F/FPEAK.
```

The summation method used in Reference 24 for multiple oxidizer modes has been used for both models to assure proper response at both zero and infinite frequency. As both models were based on pressure-coupled T-burner data, the real part of the response (RPOVNR) is used for scaling, for example,

```
ARPLI = ARPLI + TEMPR * ENP, where
TEMPR = CONCD * MPEAK ** XY, and the exponent
XY = FACXY * ln (RPOVNR)
```

Reference 24 arbitrarily set APARAM = 14, and selected BPARAM to fit the required curve shape. Using this method, two values of BPARAM were required to fit the curve shape of Reference 23. The resulting empirical

constants for the two models are listed in Table 8.

PTLSTB (J, K)

This subroutine performs the same functions as SSP/PTLSTB. The code was rewritten deleting particle damping, wall damping, distributed combustion and the Flandro term. In its present form, PTLSTB sums the products of combustion response and the appropriate partial integral along the length of the motor, sums all calculated gains and losses to determine the total alpha (AL), calculates the fraction of critical damping (H) and HMIN and writes the output for that mode.

TABLE 8

EMPIRICAL CONSTANTS IN COMBUSTION RESPONSE MODELS

BASED ON AP CONTENT

	Cohen Model		Hessler Model
IRSPNS =	3		4
APARAM	14.		14.
FCONST	6.0		2.25
PWROFD	-1.0		-0.1
FACPK	1056.0		9.5
	(F < FPEAK)	(F > FPEAK)	
BPARAM	0.9	1.3	1.4
OMEGPK	12.069	12.621	12.706
FACXY	0.764676	1.42049	1.57824

WEIGHTS

All weights are calculated directly from dimensions that are user input or are internally generated, except for the following (which do not vary during a given problem):

WOTHER (components credited to the propulsion unit)

- Environmental closure
- Safe-and-arm device
- Launch lugs
- Wing lugs
- Igniter
- Miscellaneous

WNP (missile components not credited to propulsion unit)

- Warhead
- Guidance and Control
- Wings

These two weights must be determined by the user and supplied as fixed inputs.

Thrust skirt weights also do not change during a given problem, but the forward skirt length is needed to calculate overall motor length, and so the code will calculate forward and aft skirt weights from input lengths and thicknesses. An alternate approach is to include skirt weights as part of WOTHER, input skirt thicknesses as zero and input an appropriate value for forward skirt length; thus the calculated skirt weight will be zero but the length is available for motor length calculations.

Provisions are made to include weight allowances for joint flanges, even though there is no direct joint design and weight estimate made in the code. For the pressure vessel, this allowance is achieved by adhering to the designer's rule-of-thumb that the volume of metal removed to provide an opening in the closure should be replaced in the flange surrounding that opening for attachment of the mating component. Thus for the aft ellipsoidal closure (Aft Closure Type 1), the weight of the center segment shown in Figure 88 is allocated to the aft flange on the case and is separately identified as WFLGA. An allowance for the weight of the flange on the nozzle is provided by virtue of the nozzle computation scheme that measures all nozzle thicknesses normal to the interior surface. Therefore, there is a portion of the nozzle that can rightly be designated as "flange" (Figure 88), although it is not identified as such in the code output. At the forward end, the closure weights are calculated as if there were no opening, thus preserving the above quoted rule-of-thumb, but as with the nozzle, there is no special identification of the forward flange weights.

There is a small volume of material accounted for twice, once in the skirt volume and once in the ellipsoidal closure volume (Figure 89). There is always a fillet between the skirt and closure and the duplicated volume makes at least some allowance for this extra material.

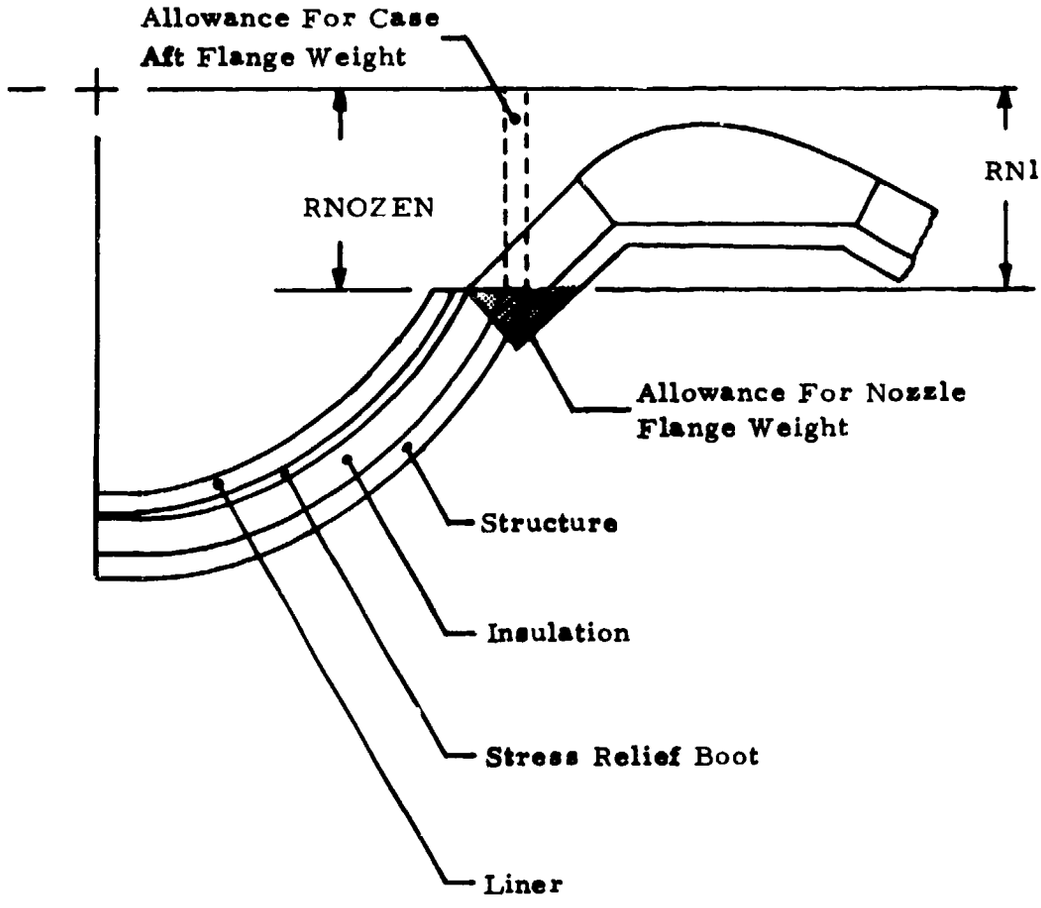


Figure 88. Weight Considerations at Aft Case Opening

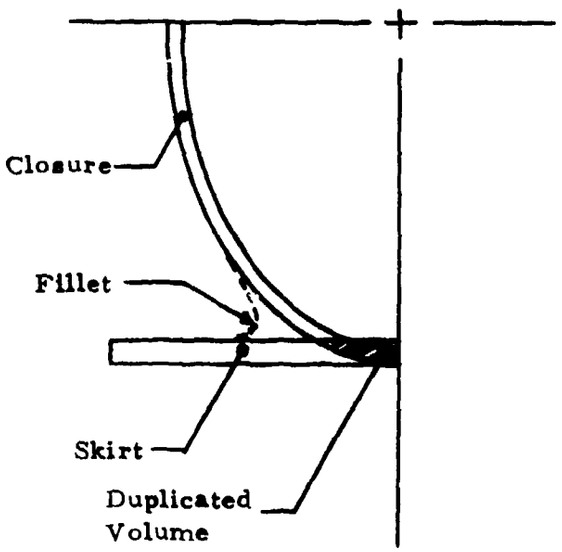


Figure 89. Weight Allowance for Skirt-to-Closure Fillet

MISCELLANEOUS

Liner and Insulation

Liner is described to SPOC as a single material having a constant thickness everywhere in the motor.

Two classes of insulation may be defined in the code; one is allocated specifically to protect the grain against the effects of external aerodynamic heating and the other is to protect the case against internal heating by the combustion products.

Aerodynamic heating insulation is specified through input of a thickness (TAEROI) and density (DELAI). This insulation has a constant thickness over the entire cylindrical portion of the case; there are no provisions for aerodynamic heating insulation in the closures. The thickness cannot be adjusted during a given machine submission.

Internal heating insulation is specified through input of a number of thicknesses and a single value of density (DELINS). This insulation has a constant thickness around the periphery at any given longitudinal station, but the thickness can vary along the length of the motor in accordance with the input features provided for the different grain types. In particular, this means that the internal insulation cannot be contoured to match a star or wagon wheel grain; instead, the user must select an average thickness with which to account for the volume displaced by the insulation. Internal insulation thickness is not adjusted during a given machine submission; the user must make a selection for the initial inputs at the different locations in the motor based on his knowledge of what the heating conditions are expected to be. After reaching a full or partial solution to the problem, these thickness choices are reviewed, based on the latest definition of heating conditions, and adjustments to the thicknesses are made if deemed appropriate (or if dictated by thermal analyses made external to the optimization problem); this is the basic preliminary design approach.

Insulation in the Type 1 (ellipsoidal) forward and aft closures is described with a minimum thickness at the case-closure tangent points and a maximum thickness at the case openings. Insulation on the flat plate forward closure Type 2 or Type 3 has a separate thickness input. All internal heating insulation has the same density; a different density may be input for the aerodynamic heating insulation.

Stress-Relief Boots

Stress relief boots may be specified through input of a thickness at the case openings and a separate density value. The boot tapers to zero thickness at the case-closure tangent points. Stress relief boots (or flaps) cannot be defined for closure Types 2 or 3, but allowances can be made for them by adjusting the insulation thicknesses.

USER MODELS

Provisions have been made for the user to supply special mathematical models to describe certain characteristics of the motor. They are:

(1) Propellant burn rate
(2) Propellant nominal strain endurance
(3) Propellant rheological property
(4) Motor costs
(5) Impulse efficiency
(6) Combustion response

The procedure for employing a user-supplied model is as follows:

(1) In the subroutines that are furnished (Table 9.)
    (a) Code the mathematical model, making sure the dependent variable has the nomenclature given in Table 9.
    (b) Add any common statements required to furnish the independent variables required by the model.
    (c) Furnish any WRITE commands that are needed to print information from the model. See further discussion below.
(2) Compile the subroutine and link with the remainder of the code.
(3) Set the flag in the namelists shown in Table 9 to show that a given user model is being furnished.

The internal flag of IPRINT is used to control when computation results are printed.

IPRINT = 1	All write statements are executed
IPRINT = 0	Only PATSH output is printed

Therefore, WRITE commands in the user-model subroutines should be structured such that they are executed when IPRINT = 1 and are branched around when IPRINT = 0.

If data must be supplied to the model by the user, a READ command must be included for a namelist defined by the user. Common TRIGR contains the flag IPRINT (discussed above) and a similar flag to read data (IREAD).

IREAD = 1	To read data
IREAD = 0	To bypass read command

IREAD = 1 only for first pass through COMP.
IPRINT = 1 for both the first and last passes through COMP.

TABLE 9

USER-SUPPLIED MODELS

Parameter to be Supplied	Flag to Call User Model: Name	Flag to Call User Model: Input NL	Load Model In Subroutine
Propellant burn rate, RATE (in/sec)	RBMDL(1)	CONTRL	USERRB
Propellant nominal strain endurance, SENOM (in/in)	SEMDL(1)	STPROP	USERSE
Propellant rheological property, EOM (units by user)	EOMMDL(1)	INGLM	USERRH
Motor cost, COST ($ or $/unit)	CSTMDL(1)	CONTRL	USERCS
Impulse efficiency, ETAISP (% x 0.01)	EFMDL(1)	CONTRL	USEREF
Combustion response	IRSPNS(2)	STABIN	RSPNSE

(1) T = user model is furnished; F = internal model will be used.
(2) 5 = user model is furnished.

PROPELLANT BURNING RATE

The user-supplied propellant burn rate model must be installed in subroutine USERRB. The flag RBMDL = T must be input in namelist CONTRL so that USERRB will be called whenever burn rate must be calculated. The common statement /IB/ must be employed in subroutine USERRB, to furnish values for pressure (P), Mach number (MACH), and rate scale factor (SF) and to pass back the calculated burn rate (RATE). If the user wishes to identify which burn rate equation out of several possibilities is being employed in subroutine USERRB, the common statement /SEC3/ must be included to furnish the variable RATEQU; values for RATEQU will be printed out at time zero in the ballistic simulation for each plane along the grain.

The burn rate subroutine is called from subroutines HITEMP, LOTEMP and ONETMP with a pressure averaged over the ballistic simulations at high temperature, low temperature and single temperature, respectively. The purpose of the calculations at this point in the code is to obtain an average burn rate without cross flow effects to send to the SPP impulse efficiency model. Therefore, Mach number is not supplied from these subroutines, and a flag is set (IFLAG = 1) to mark where the call to the rate subroutine originated. Then, in the rate subroutine, IFLAG = 1 triggers a return to the calling routine after a non-erosive rate has been calculated. The user-supplied subroutine USERRB must contain this same response.

Other commons that already exist may be included in USERRB in order to furnish the data needed for the user model.

Temperature effects on burn rate can be included through the use of TMPUR as temperature in the new model. The variable TMPUR, which is contained in the common statement MODELS, is equated to THI just prior to the high temperature ballistic simulation; it is then equated to TLO just before the low temperature simulation. Thus, TMPUR has the appropriate value at the time USERRB is called in the ballistic simulation.

A typical USERRB is given in Table 10.

PROPELLANT STRAIN ENDURANCE

The user-supplied propellant strain endurance model must be installed in subroutine USERSE. The flag SEMDL = T must be input in namelist STPROP so that USERSE will be called out of subroutine PROPST. The model must furnish the nominal strain endurance (SENOM) for comparison with thermally induced strains. The nominal value is devalued for mix-to-mix variations and aging degradation in PROPST. SENOM is returned from USERSE through the calling argument.

If the model is not furnished, SEMON is a constant user-supplied input.

TABLE 10

TYPICAL USER-SUPPLIED

BURN RATE SUBROUTINE (USERRB)

```
ISN 0002          SUBROUTINE USERRB
ISN 0003          COMMON/TRIGR/IREAD,IPRINT,IQUIT,IQUITI,IFLAG                    COM00120
                  C                                                               COM00160
ISN 0004          COMMON /DLIST/ DI(75),KSWITCH(75),OBJ,WINERT,LTOTAL             COM00170
                  C
ISN 0005          COMMON /IB/APHIB, APNIB, SFHIB, SFNIB, CDRHIB, CDFNIB, DLASIB,  COM00240
                 .          PHIB, THIB, DLTHIB, UHIB, WDTHIB, RTEHIB, P, T, DELTA, COM00250
                 .          U, WDOT, MACH, SF, CDRATF, RATE, SGNCDF                COM00260
                  C                                                               COM00270
ISN 0006          COMMON/MODELS/TMPUP
                  C
ISN 0007          COMMON/REQIR/FDT,PHMAX,MACMAX,AE,PATMI,CVF,CVTH,CVI,CVPI,EMIN,  COM00330
                 .ITREQ,TRMXLM,TRMNLM,MACLIM,IB,PIGN,CVP,FSULT,FSYLD,FMAX,WND,ACLIMCOM00340
                 .PMXLIM,PULT,PYIELD,TLC,MCDCAS,PRCAS,DELVRQ,TTTRQ,VTRQ,VBO,ISP70, COM00350
                 .ISPHI,ISPLO,NTERPS,ACLMAX,WCTHER,THI,ITMIN                      COM00360
                  C                                                               COM00370
                  C
ISN 0008          COMMON/APDATA/CONCAP,IMODE,DIAAP(3),CONCD(3)
                  C
ISN 0009          COMMON /CHEMIN/OIND,FUEL,OXA(3),OXB(3),RCATL,RCATS,STAB,IOIND,
                 .IFUEL,IOXA(3),IOXB(3),IRCL,IRCS,ISCL,ISTAB,FUELMX,RCLMX,RCSMX,
                 .ISMAX,LLR,LLL
                  C                                                               00000330
ISN 0010          COMMON /SEC1/DELF, GAMMA, RGAS, TS, KR1, KR2, KR3, KR4, KR5,    00000340
                 .KR6,USONIC,CIR1,CIR2,CIR3,RATEQU                                00000270
                  C
ISN 0011          REAL MACH,KR1,KR2,KR3,KR4,KR5,KR6
ISN 0012          REAL MC,LOBAR1,LOBAR3,LOGA70,LPR70
                  C
                  C     USER MUST INPUT VALUES FOR MAX AND MIN BURN RATE CONSTRAINTS,
                  C     TEMPERATURE COEFFICIENT OF CSTAR (MC),
                  C     TEMPERATURE COEFFICIENT OF PRESSURE (PIK),
                  C     BEFORE COMPILING THIS SUBROUTINE
                  C
ISN 0013          RBMAX = 100.0
ISN 0014          RBMIN = 0.0
ISN 0015          PIK = 0.0015
ISN 0016          MC = 0.0005
                  C
                  C     CONVERT INPUTS TO UNITS NEEDED FOR RATE MODEL (CHANGE WEIGHT
                  C     FRACTION TO WEIGHT PERCENT)
                  C
ISN 0017          AP1 = OXA(1) * 100.
ISN 0018          AP2 = OXA(2) * 100.
ISN 0019          AP3 = OXA(3) * 100.
ISN 0020          TOTAP = (OXA(1) + OXA(2) + OXA(3)) * 100.
ISN 0021          AL = FUEL * 100.
ISN 0022          FEC = RCATS * 100.
                  C
                  C     CALCULATE CONSTANT TERMS
                  C
ISN 0023          DBAR1 = (AP1*DIAAP(1) + AP2*DIAAP(2) + AP3*DIAAP(3))/TOTAP
ISN 0024          LOBAR1 = ALOG10(10.*DBAR1)
ISN 0025          LOBAR3 = (AP1*ALOG10(10.*DIAAP(1))+ AP2*ALOG10(10.*DIAAP(2))+
                 .          AP3*ALOG10(10.*DIAAP(3)))/TOTAP
ISN 0026          TEMP1 = 0.587*LOBAR3
```

Table 10 (Continued)

Typical User-Supplied Burn Rate Subroutine (USERRB)

```
ISN 0027          TEMP2=-0.00371*TOTAP*LOBAR1
ISN 0028          TEMP3 = 0.0466*FEO*LOBAR1
ISN 0029          TEMP4 = -0.0221*AL*FEO
                  C
                  C     CALCULATE BURN RATE PARAMETERS AT 70 DEG. RATE IN MODEL IS
                  C     SCALED TO 10 TIMES TRUE RATE
                  C
ISN 0030          LOGA70 = TEMP1 + TEMP2 + TEMP3 + TEMP4 - 1.25
ISN 0031          A70 = (10.** LOGA70)/10.
ISN 0032          XN = 0.0136*TOTAP + 0.000142*FEO*TOTAP*AL - 0.0298*FEO*FEO -
                 .     0.255*LOBAR3 + 0.000266*AL*AL
                  C
                  C     CHECK BURN RATE AT 70 DEG, 1000 PSIA AGAINST CONSTRAINTS
                  C
ISN 0033          LRB70 = LOGA70 + XN*3.0
ISN 0034          RB70 =10.**(LRB70 - 1.0)
ISN 0035          IF(RB70.GT.RBMAX) OBRBMX = ((RB70 - RBMAX)**2)*1.0E4
ISN 0037          IF(RB70.LT.RBMIN) OBRBMN = ((RB70 - RBMIN)**2)*1.0E4
                  C
                  C     UPDATE PENALTY SUMMATION
ISN 0039          OBJ = OBJ + OBRBMX + OBRBMN
                  C
                  C     CALCULATE PRESSURE COEFFICIENT AT TEMPERATURE EXTREMES
                  C
ISN 0040          IF(NTEMPS.NE.2)TMPUR = 70.
ISN 0042          ATMPUR=A70*EXP((PIK*(1.0-XN)-HC) * (TMPUR - 70.0))
                  C
                  C     WRITE BASIC DATA FROM RATE MODEL
                  C
                  C     PRINT ON FIRST CALL TO USERRB DURING FIRST PASS THROUGH COMP
ISN 0043          IF((IPRINT.EQ.1.AND.IREAD.EQ.1.AND.IFIRST.NE.1)GO TO 20
                  C
                  C     PRINT ON FIRST CALL TO USERRB DURING LAST PASS THROUGH COMP
ISN 0045          IF((IPRINT.EQ.1.AND.IREAD.EQ.0.AND.ILAST.NE.1)GO TO 20
                  C
ISN 0047          GO TO 30
                  C
ISN 0048       20 WRITE(6,1000)
ISN 0049          WRITE(6,1001)RB70,XN,A70,OBRBMX,OBRBMN
ISN 0050          WRITE(6,1002)AP1,AP2,AP3,TOTAP,FEO,AL
ISN 0051          WRITE(6,1003)OBAR1,LOBAR1,LOBAR3
ISN 0052          WRITE(6,1004)LOGA70,ATMPUR,TMPUR
ISN 0053          IF((IPRINT.EQ.1.AND.IREAD.EQ.1.AND.IFIRST.NE.1))IFIRST=1
ISN 0055          IF((IPRINT.EQ.1.AND.IREAD.EQ.0.AND.ILAST.NE.1))ILAST=1
                  C
                  C     CALCULATE BURN RATE AT CURRENT CONDITIONS
                  C
ISN 0057       30 RATE = SF * ATMPUR*(P**XN)
ISN 0058          RATEQU = 1.0
                  C
                  C     IF IFLAG=1,RATE SUBROUTINE HAS BEEN CALLED FROM SUBROUTINE HITEMP,
                  C     LOTEMP,OR ONETMP TO CALCULATE RATE AT AN AVERAGE PRESSURE WITHOUT
                  C     EROSIVE BURNING. MUST HAVE THIS LINE OF CODE IN ANY RATE SUBROUTINE
                  C
ISN 0059          IF((IFLAG.EQ.1) RETURN
                  C
                  C     CORATE=1.0 FOR NO EROSIVE BURING, =2.0 FOR EROSIVE BURNING.  M IS
                  C     USED TO OBTAIN INTEGER TO TEST.  M=1 CAUSES RE USN WITHOUT
```

Table 10 (Continued)
Typical User-Supplied Burn Rate
Subroutine (USERRB)

```
                C     CALCULATING EROSIVE BURNING EFFECTS.  MUST HAVE THESE TWO LINES
                C     OF CODE.
                C
ISN 0061              M = CDRATE + 0.5
ISN 0062              IF(M.EQ.1) RETURN
                C
                C     KR3=MCRIT,KR4=XM=X,MACH=LOCAL MACH NUMBER.  CAUSES SADERHOLM MODEL
                C     TO BE SKIPPED IF KR3 LESS THAN ZERO OR LOCAL MACH NUMBER LESS THAN
                C     MCRIT.
                C
ISN 0064              IF(KR3.LE.0.0) GO TO 10
ISN 0066              IF(MACH.LE.KR3) GO TO 10
                C
                C     EROSIVE BURING WITH SADERHOLM MODEL
                C
ISN 0068              X = ALOG(0.06768*(P**0.74/RATE)**0.4948)
ISN 0069              IF(X.LE.0.0) X=0.0
ISN 0071              IF(KR4.GT.0.001) X = KR4
ISN 0073              RATE2 = RATE*(MACH/KR3)**X
ISN 0074              IF(RATE2.GT.RATE)RATEQU=2.0
ISN 0076              IF(RATE2.GT.RATE) RATE=RATE2
                C
                C     EROSIVE BURING WITH MACH-PRESSURE PRODUCT MODEL
                C
ISN 0078           10 IF(MACH * P.LE.0.0) RETURN
ISN 0080              RATE3 = KR5*(MACH*P)**KR6
ISN 0081              IF(RATE3.GT.RATE)RATEQU=3.0
ISN 0083              IF (RATE3.GT.RATE)RATE=RATE3
                C
ISN 0085            1000 FORMAT(1H1,41H     BASIC DATA FROM USER BURN RATE MODEL)
ISN 0086            1001 FORMAT(1HO,5HRB70=,E12.4,3X,3HXN=,E12.4,3X,4HA70=,E12.4,
                    .3X,7HOBRBMX=,E12.4,3X,7HOBRBMN=,E12.4)
ISN 0087            1002 FORMAT(1HO,4HAP1=,E12.4,3X,4HAP2=,E12.4,3X,4HAP3=,E12.4,
                    .3X,6HTOTAP=,E12.4,3X,4HFEO=,E12.4,3X,3HAL=,E12.4)
ISN 0088            1003 FORMAT(1HO,7HORAR1=,E12.4,3X,7HLOBAP1=,E12.4,3X,7HLOBAR3=,E12.4)
ISN 0089            1004 FORMAT(1HO,7HLOGA70=,E12.4,3X,7HATMPUR=,E12.4,3X,7HITMPUR=,E12.4)
ISN 0090              RETURN
ISN 0091              END
```

PROPELLANT RHEOLOGICAL PROPERTY

The user-supplied propellant rheology model must be installed in subroutine USERRH. The flag EOMMDL = T must be input in namelist INGLIM so that USERRH will be called out of subroutine TCHEM.

This parameter (EOM) must be defined by the user; it can be end-of-mix viscosity (hence EOM), shear stress, or any other measure of rheological characteristics. The intent is that this model be used when propellant ingredient concentrations are being adjusted, so that some control can be exercised on the propellant processibility. If the flag has been set (EOMMDL = T), the output of the model (EOM) is compared with an input maximum limit, and a penalty (OBJEOM) is calculated if the limit is exceeded. EOM is returned from USERRH through the calling argument.

MOTOR COST

The user-supplied motor cost model must be installed in subroutine USERCS. The flag CSTMDL = T must be input in namelist CONTRL so that USERCS will be called out of subroutine COMP. The model must furnish the parameter COST, which is used only as one of the payoff parameters. The units can be either total project cost or unit cost. COST is returned from USERCS through the common statement MISL.

IMPULSE EFFICIENCY

The user-supplied impulse efficiency model must be installed in subroutine USEREF. The flag EFMDL = T must be input in namelist CONTRL so that USEREF will be called out of subroutine COMP. If a user efficiency model is called, then the SPP model cannot be called (i.e., SPPETA = F is required). The model-furnished impulse efficiency then is used in the ballistic simulation. Efficiency is returned from USEREF through the calling argument.

COMBUSTION RESPONSE

The user-supplied combustion response model must be installed in subroutine RSPNSE, which is the subroutine where all internal models are located. Thus many sources of input data are already available. Entry to the model is statement number 500. The flag IRSPNS = 5 will cause the user-supplied model to be called.

Combustion response is returned through the common ALF already furnished.

ERROR MESSAGES

Error messages are printed whenever code execution is terminated under abnormal conditions. Other messages are provided the user to show conditions under which the code is operating. The messages that originate from subroutines that were developed for SPOC are self-explanatory and do not require a complete listing in the user's manual (except as discussed below). Generally these messages explain what caused the termination and provide data to show the abnormal condition; if the solution is not self-evident, instructions are given.

There is a large group of termination messages that originate from the subroutines comprising the ballistic simulation module. None of these conditions should be encountered because the purpose of the SETUP subroutine is to prevent invalid conditions from occurring during the optimization process. Nevertheless, the ballistic module error messages are given in Table II in the event that a combination of inputs and optimization searches produces an invalid situation.

The messages are listed alphabetically. Those that begin with asterisks are listed alphabetically at the end of the table. With the message is the reason that message was triggered. The last column of the table is the format statement number and the subroutine in which that statement is found. Note that some of the messages speak of a "Type 1" or "Type 2" grain; these are designations internal to the ballistic simulation module and have no direct relation to the grain types that can be selected by the user of SPOC. For information purposes, the correspondence is

SPOC Grain Type	Ballistic Module Grain Type
1 (Star)	2
2 (Wagon Wheel)	2
3 (Finocyl)	3
4 (Conocyl)	1 (Fwd. segment) 3 (Remaining)
5 (CP)	3

Incompatible inputs for the trajectory simulation are detected and identified with the flag ISTOP Definitions of ISTOP are given in Table 12.

TABLE 11

ERROR MESSAGES IN BALLISTIC SIMULATION MODULE

ERROR MESSAGE AS IT APPEARS IN THE PRINTOUT (ALPHABETICAL ORDER)	THE PROGRAM WILL TERMINATE AND THE RESPECTIVE ERROR MESSAGE WILL BE PRINTED IF:	LOCATION CODE
B 71 MAX NEGATIVE	The input parameter α_3 of type 2 configurations is so large that a line perpendicular to L_C at its end tangent to R5 crosses the outside edge of propellant before it crosses the centerline of that star point (See Figure 2a).	2076 PC2SUB
ERROR - A (F + SL) GREATER THAN A-LIMIT	The propellant plus inert sliver cross sectional area is greater than the cylindrical cross sectional area formed by RF for a type 1 configuration.	1006 PC1SUB
ERROR - CLOSURE LENGTH LESS THAN MAXIMUM THICKNESS OF ADJACENT PLANE	The total length of the end section is shorter than the maximum thickness of the adjacent cylindrical section plane.	9003 ECSUB
ERROR - END CLOSURE CAN NOT BE USED WITH NON-ZERO VALUE OF LS1 IN ADJACENT PLANE	An end section is called for and LS_1 of the adjacent plane of type 3 configuration is not equal to zero.	9005 ECSUB
ERROR: IMAGINARY SOLUTION FOR LA	In the calculation of L_A (type 3 configurations), a negative value appears under the radical. (This is an intermediate calculation. Check the value of $\alpha 3$.)	512 PC3SUB
ERROR: IMAGINARY SOLUTION FOR TAUIN	In the calculation of τ_{in} (type 3 configurations), a negative value appears under the radical. (This is an intermediate calculation)	514 PC3SUB
ERROR: IMAGINARY SOLUTION FOR T4MAX	In the calculation of $\tau_{4_{max}}$ (type 3 configurations), a negative value appears under the radical. This is an intermediate calculation)	513 PC3SUB
ERROR - INITIAL INPUT LENGTH = 0.0	The initial end closure length is equal to zero. The plane adjacent to the end closure is already in the calculation so inputting it again caused division by zero length.	9004 ECSUB
ERROR - INITIAL THICKNESS NOT EQUAL TO ZERO	The initial thickness burned in type 1 configuration is not zero.	1007 PC1SUB
ERROR - INVALID CRITICAL MACH NUMBER INPUT	The input parameter M (MCRIT) is less than zero. The automatic erosive burning option cannot get this message.	1005 RTCKSB
ERROR - INVALID DELTA -TIME INPUT	The burning time increment is less than or equal to zero.	1034 SEC1SB
ERROR - INVALID EXIT DIAMETER INPUT	The nozzle throat area is greater than the nozzle exit area at time = 0.	1007 NOZZSB
ERROR - INVALID LENGTH INPUT	The length defining one plane is less than that defining the previous plane.	1009 SEC1SB

Table 11 (Cont'd)
Error Messages in Ballistic Simulation Module

ERROR MESSAGE AS IT APPEARS IN THE PRINTOUT (ALPHABETICAL ORDER)	THE PROGRAM WILL TERMINATE AND THE RESPECTIVE ERROR MESSAGE WILL BE PRINTED IF:	LOCATION CODE
ERROR - INVALID PERIMETER	Any perimeter of type 1 configuration is less than zero.	1009 PC1SUB
ERROR - INVALID PRESSURE COEFFICIENT INPUT	The input parameter a_T (A) is less than or equal to zero.	1003 RTCKSB
ERROR - INVALID PRESSURE EXPONENT INPUT	The input parameter n (N) is greater than or equal to 1 and KR7 (GNRL (7)) is not used	1004 RTCKSB
ERROR - INVALID RF	The input parameter RF of type 1 configuration is less than or equal to zero.	1004 PC1SUB
ERROR - INVALID SCALE FACTOR INPUT	Any input burn rate scale factor is less than or equal to zero.	1027 SEC1SB
ERROR - INVALID THICKNESS	Any thickness of type 1 configuration is less than the preceding thickness in the same plane.	1008 PC1SUB
ERROR - INVALID THROAT DIAMETER INPUT	The nozzle throat diameter is equal to or less than zero at any time.	1006 NOZZSB
ERROR - INVALID TIME INPUT	A time for a burning time increment is less than that for the preceding burning time increment. This means that the times and their corresponding burning time increments must be input in chronological order. OR A burning time increment is not defined for time = 0.	1033 SEC1SB
ERROR: NEGATIVE LA	The calculated parameter L_A in type 3 configurations is negative. This can happen only if the sum of the input values R2 and R3 is too large.	515 PC3SUB
ERROR - PORT AREA GREATER THAN A-LIMIT	Any port area of type 1 configuration is greater than the cylindrical cross sectional area formed by RF.	1011 PC1SUB
ERROR - PORT AREA NOT GREATER THAN ZERO	Any port area of type 1 configuration is less than or equal to zero.	1010 PC1SUB
LC NEGATIVE	The parameter L_C (see Figure 89) of type 2 configurations is less than zero.	2074 PC2SUB
NEGATIVE X03	The calculated parameter X03 in type 3 configurations is less than zero. This can happen if the combination of inputs is such that the origin of R3 is outside the section 180°/NO. Solve the third occurrence by decreasing α 01.	516 PC3SUB

Table 11 (Cont'd)
Error Messages in Ballistic Simulation Module

ERROR MESSAGE AS IT APPEARS IN THE PRINTOUT (ALPHABETICAL ORDER)	THE PROGRAM WILL TERMINATE AND THE RESPECTIVE ERROR MESSAGE WILL BE PRINTED IF:	LOCATION CODE
PROGRAM TERMINATED BECAUSE REFERENCE PRESSURE OR RATE IS OUT OF RANGE	The automatic erosive burning option is called for and the pressure or burn rate used in the table-look-up for Mach number exponent (x) is out of table limits. Pressure must be between 250 psia and 3000 psia and burn rate must be between 0.2 in/sec and 1.0 in/sec. These values are taken from a trial run using no erosive burning, which is done automatically.	1028 SEC3SB
PROGRAM TERMINATED DUE TO ITERATION FAILURE PMAX = XXXX PMIN = XXXX	The program has not converged on an equilibrium pressure in 100 or less iterations.	1002 SEC3SB
PROGRAM TERMINATED THROAT GREATER THAN EXIT AREA TIME AT AE XXXX XXXX XXXX	The nozzle throat area is greater than the nozzle exit area at any time.	1000 ATAESB
R03 TOO LARGE	The sum of the input radii R2 and k3 of type 3 configurations (Figure 90) is too large	518 PC3SUB
R1 NEGATIVE	The parameter R1 (Figure 89) of type 2 configurations is is less than zero.	2070 PC2SUB
SQUARE ROOT OR TRIG FUNCTION ERROR IN PCGSUB	A trigonometry function is negative or a function that will be raised to the .5 power is negative in PCGSUB. This subroutine is only called for geometry calculations in type 2 configurations.	1000 PCGSUB
THROAT AREA > MAX ALLOWABLE THROAT AREA THROAT AREA = XXXX MAXIMUM ALLOWABLE THROAT AREA = XXXX	The mass generated and mass discharged have not converged within 1/1000 but P_{max} and P_{min} have converged to within P_{max} / 100,000. This is generally met if the throat area is greater than the port area. This can also occur if a propellant with unstable characteristics is used.	1001 SEC3SB
** ERROR - NO INPUTS HAVE BEEN STORED IN PLANE NO. 1	No inputs are stored in plane number 1.	1003 SEC1SB
** ILLEGAL INTERPOLATION ERROR - PLANE (XXX) IS TYPE (XX) PLANE (XXX) IS TYPE (XX)	Interpolation is attempted between two different types of geometry. Geometry types can be changed if there is no interpolation between the two plane inputs defining the change.	1004 SEC1SB
*** INVALID ALPHA 01 ***	The input parameter α_{01} of type 3 configurations is not between 0° and 90° inclusive (see Figure 90).	504 PC3SUB

Table 11 (Cont'd)
Error Messages in Ballistic Simulation Module

ERROR MESSAGE AS IT APPEARS IN THE PRINTOUT (ALPHABETICAL ORDER)	THE PROGRAM WILL TERMINATE AND THE RESPECTIVE ERROR MESSAGE WILL BE PRINTED IF:	LOCATION CODE
*** INVALID INPUT INSTRUCTION *** CARD NO. XXX INSTRUCTION NO. XXX INSTRUCTION XXX	A storage location is other than the ones described in the input explanations.	9002 INPTSB
*** INVALID NO ***	The number of star points (NO) of type 2 configurations is less than or equal to zero.	9006 PC2SUB
*** INVALID NO ***	The number of sections (NO - See Figure 90) of type 3 configurations is less than 2.	508 PC3SUB
*** INVALID RF ***	The input parameter R5 (Figure 90) of type 3 configurations is greater than or equal to the input parameter RF.	507 PC3SUB
*** INVALID R2 ***	The input parameter R2 (Figure 90) of type 3 configurations less than or equal to zero.	506 PC3SUB
*** INVALID R3 ***	The input parameter R3 (Figure 90) of type 3 configurations is less than zero.	517 PC3SUB
*** INVALID R4 ***	The input parameter R4 (Figure 90) of type 3 configurations is used and is greater than R5.	505 PC3SUB
*** INVALID R5 ***	The input parameter R5 of type 3 configurations is smaller than the input parameter R2 (See Figure 90).	503 PC3SUB

TABLE 12

EXITS FROM TRAJECTORY SIMULATION FOR INCOMPATIBLE INPUTS USING ISTOP

ISTOP	Explanation
1	Both launch velocity and launch Mach number are input
2	ITERM < 1, or ITERM > 12
3	FPAAL < -90.0, or FPAAL > 90.0
4	AREF < 0.0
5	IATMOS < 0, or IATMOS > 4
6	ALTAL < 0.0
7	AEX < 0.0
8	RNGAL < 0.0
9	WNP < 0.0
10	Failure to pass the test (IPRDEG ≠ 0 and IPRDET ≠ 1)
11	RGFPAL < 0.0
12	WMI < 0.0
13	FDELT1 < 0.0
14	FDELT2 < 0.0
16	ITERM = 1 and TERTIM < 0.0
17	ITERM = 2 and TPHANE < 0.0
18	ITERM = 3 and SPTERM < 0.0
19	ITERM = 4 and RGTERM < 0.0
20	ITERM = 6 and MTERM = 0.0
21	ITERM = 7 and VLTERM = 0.0
22	ITERM = 8 and FPAAL < -90.0
23	ITERM = 8 and FPAAL < FPATRM
24	ITERM = 8 and FPAAL > 90.0
25	ITERM = 9 and ACTERM $\geq$ 0.0
26	ITERM = 11 and RGFPTM $\leq$ 0.0
27	ALTERM < 0.0, and ALTAL < |ALTERM|, and FPAAL > 0.0
28	ALTERM < 0.0, and ALTAL < ALTERM, and FPAAL $\leq$ 0.0

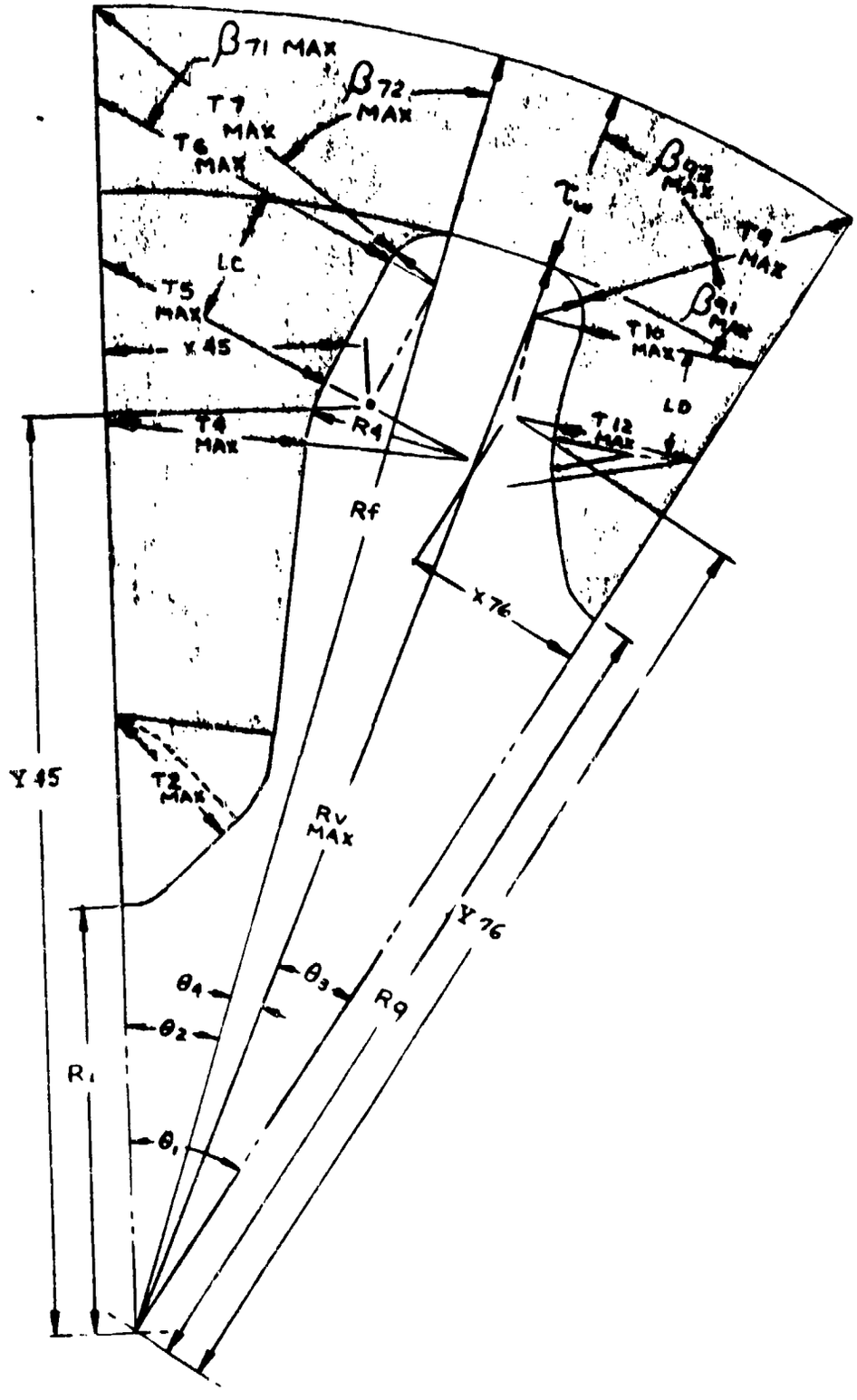


Figure 90. Internally Calculated Dimensions for Type 2 Grain

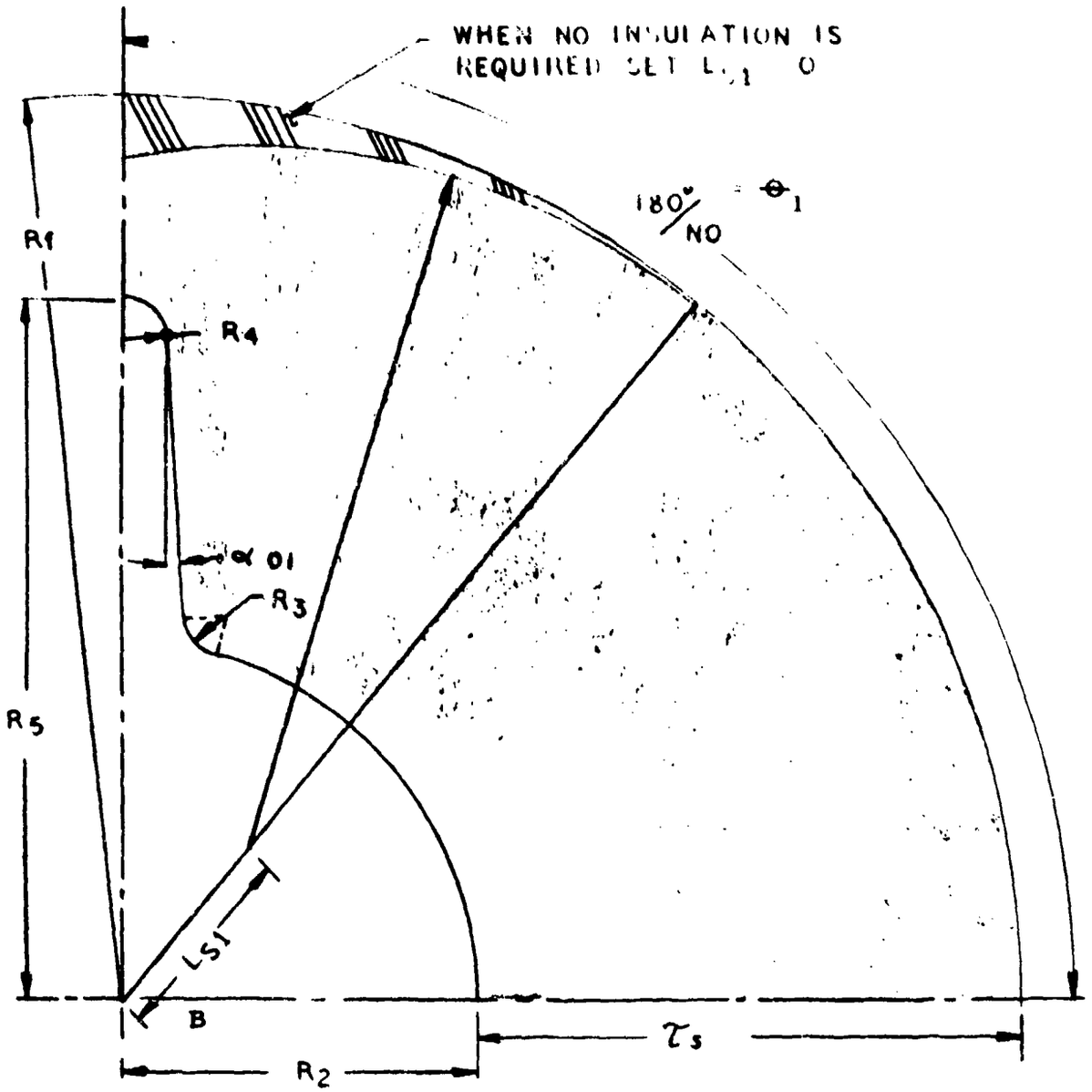


Figure 91. Inputs for Finocyl Configuration

REFERENCES

1. AO 142/79/A239, "Tactical Missile Optimization Program TACMOP", Final Report, M. D. Harsh, Hercules/ABL, October 1979.

2. Whitney, D. E. Personal Communication. Joint Mechanical and Civil Engineering Computing Facility, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, Massachusetts 02139.

3. Hooke, R., and T. A. Jeeves. "'Direct Search' Solution of Numerical and Statistical Problems." Journal of the Association for Computing Machinery, Vol. 8, No. 2. pp 212-229.

4. Afimiwala, K. A., and R. W. Mayne. "Evaluation of Optimization Techniques for Applications in Engineering Design." Journal of Spacecraft and Rockets, Vol. II, No. 10. pp 673-674.

5. Eason, E. D., and Fenton, R. G., "A Comparison of Numerical Optimization Methods for Engineering Design", Transactions of the ASME, Ser. B: Journal of Engineering for Industry, Vol. 96, Feb 1974, pp 196-200.

6. Internal Review of Recent Designs, Circa 1980, Thiokol/Huntsville.

7. Papers presented at 1979 JANNAF Propulsion Meeting, 6-8 March 1979, Anaheim, CA.

8. CPIA/M1 Rocket Motor Manual.

9. Saderholm, C. A., "A Characterization of Erosive Burning for Composite H-Series Propellants," AIAA Solid Propellant Rocket Conference, January 29-31, 1964.

10. Saderholm, C. A., "Special Report--Erosive Burning Study of TP-H8041 Propellant", Thiokol/Huntsville, Report No. C-A-61-176A.

11. Svehla, R. A., and McBride, B. J., "FORTRAN IV Computer Program for Calculation of Thermodynamic and Transport Properties of Complex Chemical Systems", NASA TN D-7056, Lewis Research Center, January 1973.

12. AFRPL-TR-75-36, "A Computer Program for the Prediction of Solid Rocket Motor Performance; Vol. I"; Ultrasystems Inc.; Irvine, Calif. 92664; July 1975.

13. Heat, Mass, and Momentum Transfer, Rohsenow, W. M., and Choi, H. Y., Chapter 6, pp 122-125, 1961.

References (Continued)

14. Heat Transmission, McAdams, W. H., Chapter 3, pp 36-38, 1954.

15. Formulas for Stress and Strain, Raymond J. Roark, McGraw-Hill Book Co., New York, N.Y., 1954.

16. NASA SP-8073, Solid Propellant Grain Structural Integrity Analysis, NASA Space Vehicle Design Criteria (Chemical Propulsion), National Aeronautics and Space Administration, June 1973.

17. CPIA Publication 214, Handbook for the Engineering Structural Analysis of Solid Propellant, May 1971.

18. CPIA Publication 322, "Tri-Services Rocket Motor Trade-off Study", Booz, Allen Applied Research, Contract N00123-76-C-0732, May 1980.

19. Lovine, R. L., "Standardized Stability Prediction Method for Solid Rocket Motors", AFRPL-TR-76-32, May 1976.

20. Butcher, A. G., "A Comparison of Predicted Acoustic Stability with Motor Test Data", AFRPL-TR-79-53, November 1979.

21. Hessler, R. O., "Prediction of Finite Pressure Oscillations in Stable Rocket Motors," 17th JANNAF Combustion Meeting, September 1980.

22. Culick, F. E. C., "A Review of Calculations for Unsteady Burning of a Solid Propellant", AIAA Journal, Volume 6 (12), pp. 2241-2255, December 1968.

23. Cohen, N. S., et al, "Design of a Smokeless Solid Rocket Motor Emphasizing Combustion Stability", CPIA 273, Vol II, pp 205-220, December 1975.

24. Hessler, R. O., "An Empirical Propellant Response Function for Combustion Instability Predictions," 17th JANNAF Combustion Meeting, September 1980.