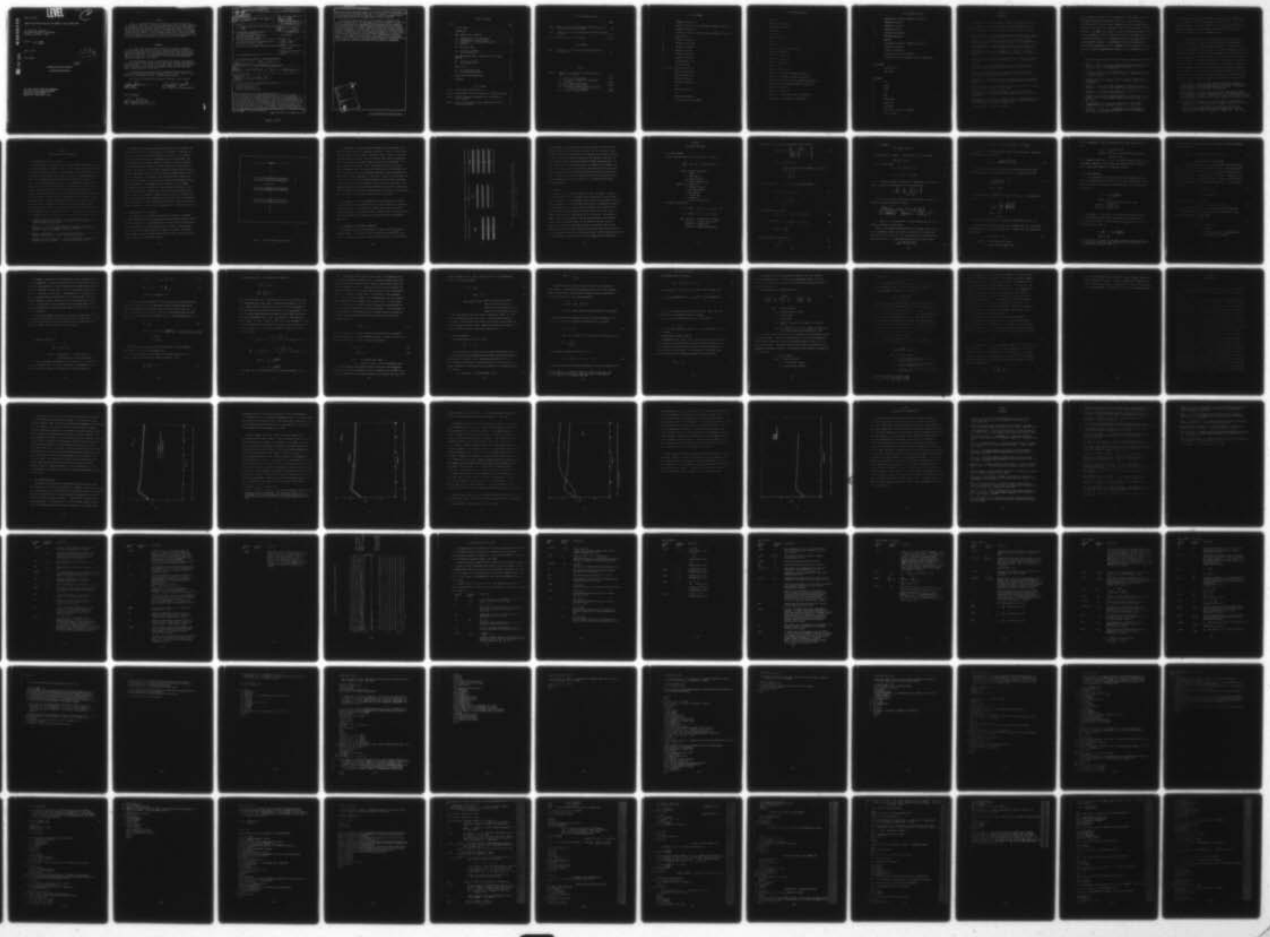
AD-A060 045 JET PROPULSION LAB PASADENA CALIF F/G 19/1
COMBUSTION RESPONSE MODELING FOR COMPOSITE SOLID PROPELLANTS.(U)
JUN 78 N S COHEN, J M BOWYER F04611-76-X-0050
UNCLASSIFIED JPL-PUB-78-59 AFRPL-TR-78-39 NL

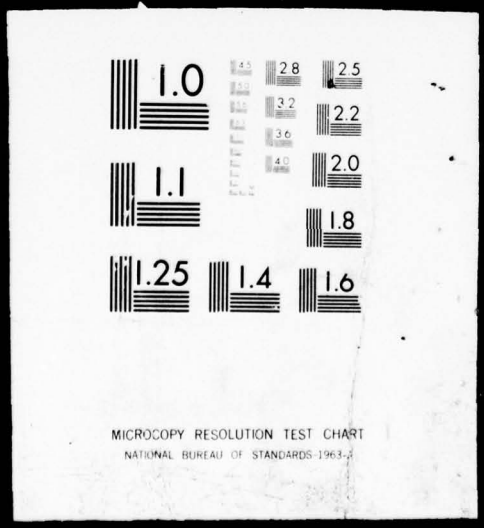
1 OF 2
AD
A060045



IFIED

1 OF 2

AD
AO 60045



AD A060045

DDC FILE COPY

LEVEL II

AFRPL-TR-78-39

COMBUSTION RESPONSE MODELING FOR COMPOSITE SOLID PROPELLANTS

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California 91103

Authors: N. S. Cohen
J. M. Bowyer

June 1, 1978

Final Report

DDC
RECEIVED
OCT 17 1978
F

APPROVED FOR PUBLIC RELEASE,

DISTRIBUTION UNLIMITED

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

78 09 28 044

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 be 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 may in any way be related thereto.

FOREWARD

This report was prepared by the Jet Propulsion Laboratory, Pasadena, California 91103, under Air Force Rocket Propulsion Laboratory MIPR No. F04611-76-X-0050 as administered by the National Aeronautics and Space Administration under Contract NAS7-100. This investigation is entitled, "Combustion Response Modeling for Composite Solid Propellants," and was technically monitored by Captain Jack Donn and Mr. Jay Levine.

The Program Manager for this program was Mr. Leon Strand. Contributions to the technical effort were made by Drs. James Bowyer, Norman Cohen, Fred Culick and Kumar Ramohalli. The program was conducted within the Advanced Technology Group of the Solid Propulsion and Environmental Systems Section, under the general supervision of Dr. Giulio Varsi.

This report has been reviewed by the Information Office/XOJ and is releasable to the National Technical Information Service (NTIS). At NTIS it will be available to the general public, including foreign nations.

This report is unclassified and suitable for general public release.

JAY N. LEVINE
Project Manager

W. C. ANDREPONT
Chief, Combustion Technology Section

FOR THE COMMANDER

JOHN I. WASHBURN, Maj, USAF
Chief, Propulsion Analysis Division

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFRPL-TR-78-39	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): COMBUSTION RESPONSE MODELING FOR COMPOSITE SOLID PROPELLANTS	5. TYPE OF REPORT & PERIOD COVERED: FINAL REPORT
6. PERFORMING ORG. REPORT NUMBER: JPL PUBLICATION 78-59	
7. AUTHOR(s): N. S. Cohen, Norman Cohen Professional Services; J. M. Bowyer	8. CONTRACT OR GRANT NUMBER(s): FO-4611-76-X-0050
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California 91103	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: JON 573010AG
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Rocket Propulsion Laboratory/AFSC, Edwards AFB, California 93523	12. REPORT DATE: June 1, 1978
	13. NUMBER OF PAGES: 135
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	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

1 Jun 78 | 137 p.

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

Norman S. Cohen James M. Bowyer

18. SUPPLEMENTARY NOTES

JPL-PUB-78-59

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

Composite Solid Propellants
Combustion Instability
Combustion Response Modeling

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

The objective of this program was to develop a computerized mathematical model of the combustion response function of composite solid propellants, with particular attention to the contributions of the solid phase heterogeneity. A one-dimensional model was developed which treats the solid phase as alternating layers of AP and binder, with an exothermic melt layer at the surface. Solution of the Fourier heat equation in the solid provides temperature and heat flux distributions with space and time. The problem is solved by equating the heat flux at the surface to that produced by a suitable model of the gas phase. An

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

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

491 750

approximation of the BDP flame model is utilized to represent the gas phase. By the use of several reasonable assumptions, it is found that a significant portion of the problem can be solved in closed form. A method is presented by which the model can be applied to tetramodal particle size distributions.

A computerized steady-state version of the model was completed, which served to validate the various approximations and lay a foundation for the combustion response modeling. The combustion response modeling was completed in a form which does not require an iterative solution, and some preliminary results were acquired. The model satisfactorily describes the steady state combustion properties, but is deficient in describing the transient combustion response. Although an effect of solid phase heterogenity on transient combustion response is predicted by the current model, the effect is so small as to allow it to be neglected in future work. It is concluded that some other mechanism associated with the propellant heterogeneity must be incorporated into the theory to account for observed behavior. Potential deficiencies of the current model are identified, and areas for future work are recommended.



TABLE OF CONTENTS

SECTION		PAGE
	LIST OF SYMBOLS	3
1	INTRODUCTION	6
2	MODEL PREMISES AND ASSUMPTIONS	11
	2.1 Representation of the Solid Phase	11
	2.2 Representation of the Gas Phase	12
	2.3 Representation of Multimodal Propellants	14
	2.4 Approach	16
3	THE STEADY-STATE MODEL	
	3.1 Solid Phase Equations	17
	3.2 Gas Phase Equations	21
4	THE TIME-DEPENDENT MODEL: CALCULATION OF THE RESPONSE FUNCTION	24
	4.1 Solid Phase Equations	24
	4.2 Gas Phase Equations	29
5	MODEL RESULTS	36
	5.1 The Steady-State Model	36
	5.2 The Time-Dependent Model	39
6	CONCLUSIONS AND RECOMMENDATIONS	47
7	REFERENCES	48

LIST OF FIGURES

Figure 1	Physical Model Representation	13
Figure 2	Layered Model of the Solid Phase for a Trimodal Propellant	15
Figure 3	Response Function Calculations for a 2 μAP Analog of A-13 Propellant	40
Figure 4	Effect of AP Particle size at Constant Burning Rate for A-13 Type Propellant	42

LIST OF FIGURES (cont'd)

PAGE

Figure 5 Effect of Burning Rate at Constant AP Particle Size for the 2μ Analog of A-13 Propellant 44

Figure 6 Comparison of Theory and Experiment for SP-540 Propellant at 500 psi 46

Figure A-1 FORFLO Chart of Overall Program A-121

LIST OF TABLES

Table I Comparisons of Burning Rate and Other Quantities for A-13 Propellant. 37

APPENDIX

Appendix A COMPOSITE SOLID PROPELLANT COMBUSTION RESPONSE MODEL COMPUTER PROGRAM . . . . . A- 51

A-1 Definitions of Input Data. . . . . A- 52
A-2 Input Deck Corresponding to the Example Problem. . . . . A- 56
A-3 Definitions of Output Data . . . . . A- 57
A-4 Card Image Listing of Program. . . . . A- 66
A-5 Descriptions of Program Logic, Subroutines and Use of Text Equations A-120
A-6 Solutions to Example Problem A-124

LIST OF SYMBOLS

a	defined in Equation 15
A	kinetics prefactor; also a solid phase parameter in Equation 51
b	defined in Equation 15
B	defined in Equation 3; also a gas phase parameter in Equation 51
c	heat capacity
C	defined in Equation 4.
C_F	defined in Equation 24
C_M	defined in Equation 13
C_1, C_2	arbitrary constants
d	defined in Equation 16
D	defined in Equation 4
D_1	oxidizer particle size
E	activation energy
$f_1(x)$, $f_2(x)$	denotes functions of x
F	defined in Equation 22
g	spatial gradient in τ
H	defined in Equation 3
H_f	defined in Equation 43
i	$\sqrt{-1}$
k	thermal conductivity
K	C/D
K_2	defined by Equation 49
n	burning rate pressure exponent

p	pressure
Q	heat of decomposition
Q_f	heat release in the gas
r	burning rate
R	gas constant
R	response function
t	time
T	temperature
u	gas velocity normal to the surface
V_3	defined in Equation 50
V_5	defined in Equation 32
V_{6A}	defined in Equation 50
V_{6B}	defined in Equation 50
W	weight fraction
x	distance into the solid
x^*	effective flame height, defined by Equation 24
y	dimensionless distance x, defined in Equation 3
y_S	time-dependent surface position, defined in Equation 29
y^*	dimensionless flame height, defined in Equation 20
z	defined in Equation 16
Z	defined in Equation 5b
Δy	thickness of a particular layer, see Equation 18
ϵ	an extremely small number, see Equation 38b

ζ	dimensionless distance x, defined in Equation 19
η	defined by Equation 7
θ	Defined in Equation 32
κ	thermal diffusivity
λ_1, λ_2	defined in Equation 36
ξ	defined by Equation 6
ρ	density
τ	normalized temperature T, defined in Equation 3
τ_F	defined in Equation 41
χ	defined in Equation 32
ω	angular frequency of oscillations
Ω	dimensionless angular frequency, defined in Equation 26

Superscripts

'	perturbed value
-	mean value

Subscripts

a	AP
b	binder
f	flame
g	gas
m	melt
N	Nth AP layer
o	deep solid
s	propellant
Top	upper side of boundry or interface
w	wall or surface

SECTION I

INTRODUCTION

Experimental data have established that ammonium perchlorate (AP) particle size has a significant effect upon the pressure-coupled response function of composite solid propellants (1-14). Moreover, the effect cannot be attributed

---

1. Green, L.G., "Effects of Oxidizer Concentration and Particle Size on Resonance Burning of Composite Solid Propellants", Jet Prop. 28, 159-164 (Mar., 1958).

2. Strand, L. D., "Low Pressure L* Instability and Extinction", 3rd ICRPG Combustion Conference (CPIA Publication 138, Vol. I, Feb., 1967) pp. 195-207.

3. "Experimental Studies on the Oscillatory Combustion of Solid Propellants", Report NWC-TP-4393, U.S. Naval Weapons Center, China Lake, CA (Mar., 1969).

4. Beckstead, M.W., Boggs, T.L. and Madden, O.H., "The effect of Oxidizer Particle Size and Binder Type on Nonacoustic Instability", AIAA Paper 69-175 (1969).

5. Boggs, T.L. and Beckstead, M.W., "Failure of Existing Theories to Correlate Experimental Nonacoustic Combustion Instability Data", J. AIAA 8, 626-631 (Apr.1970).

6. Crump, J.E., "Combustion Instability in a Series of AP-HTPB Smokeless Propellants", 8th JANNAF Combustion Meeting (CPIA Publication 220, Vol. II, Dec.1971) pp. 81-90.

7. Crump, J.E.,"Combustion Instability Studies on Non-metallized AP-HTPB Propellants", 9th JANNAF Combustion Meeting (CPIA Publication 231, Vol. III, Dec., 1972) pp 123-134.

8. Wendelken, C.P., "Combustion Stability Characteristics of Solid Propellants", AFRPL-TR-73-63, Air Force Rocket Propulsion Laboratory, Edwards, CA (Oct., 1973).

9. "Aluminum Behaviour in Solid Propellant Combustion", AFRPL-TR-74-13, Lockheed Propulsion Company, Redlands, CA (May, 1974).

10. Micheli, P.L., "Stabilization of Smokeless Propellants with Additives", 11th JANNAF Combustion Meeting (CPIA Publication 261, Vol. III, Dec., 1974) pp 123-136.

11. Anderson, F.A. and Kumar, R.N., "Feasibility Study of Propellants and Igniters for Shuttle Solid Rocket Booster Separation Motors", Report 900-710, Jet Propulsion Laboratory, Pasadena, CA (June, 1975).

(See next page for remainder of references.)

simply to changes in burning rate or formulation; the effect appears to involve the composite propellant heterogeneity as well (15). Classical theories of combustion driving (16) have assumed a homogeneous propellant and are therefore inadequate to fully explain the combustion instability characteristics of composite propellants. The community has come to rely upon experimental measurement of the combustion response in T-Burners, and work in recent years has been devoted largely to improving the method (17-19). Although experimental measurement serves several important purposes, interest in the theoretical work has revived because the acquisition and interpretation of full complements of data continues to be expensive and does not furnish a phenomenological mechanism for the guidance of propellant R & D.

Viewing the combustion zone as the region between the thermal wave penetration

---

12. Cohen, N.S., et al., "Design of a Smokeless Solid Rocket Motor Emphasizing Combustion Stability", 12th JANNAF Combustion Meeting (CPIA Publication 273, Vol. II, Dec. 1975) pp 205-220.

13. Horton, M.D. and Rice, D. W., "The effects of Compositional Variables Upon Oscillatory Combustion of Solid Rocket Propellants", Combustion and Flame 8, 21-28 (Mar. 1964).

14. "Control of Solids Distribution in HTPB Propellants", Contract F04611-76-C-0006, Hercules, Inc., Allegheny Ballistics Laboratory, Cumberland, MD. (in progress).

15. Cohen, N. S., "Report of Workshop on Combustion Instability of Smokeless Propellants", Proceedings of 14th JANNAF Combustion Meeting (to be published).

16. Culick, F.E.C., "A Review of Calculations for Unsteady Burning of a Solid Propellant", J. AIAA 6, 2241-2254 (Dec. 1968).

17. "T-Burner Testing of Metallized Solid Propellants", AFRPL-TR-74-28, edited by F.E.C. Culick, Air Force Rocket Propulsion Laboratory, Edwards, Calif. (Oct. 1974).

18. "T-Burner Motor Verification Program Final Report", AFRPL-TR-74-71, Aerojet Solid Propulsion Company, Sacramento, Calif. (Jan. 1975).

19. Lovine, R.L. and Linfor, J.J., "Measurement of Propellant Response at High Frequency", 13th JANNAF Combustion Meeting (CPIA Publication 281, Vol. II, Dec. 1976) pp 95-112.

in the solid and the location of the flames in the gas, there are several ways in which the composite propellant heterogeneity can manifest itself. Two schools of thought have arisen: one which emphasizes the solid phase, treating the gas as a homogeneous source of propellant heating; and one which emphasizes the gas, continuing to treat the solid as a homogeneous medium.

The solid phase proponents may be represented by Lengelle & Williams (20), Kumar (21) and Cohen (12). Lengelle & Williams performed a one-dimensional analysis of a solid having sinusoidal thermal properties. Although the model was too idealized for direct practical application, it made the essential point that the heterogeneity augments the combustion response depending upon the periodicity of the thermal properties (and, therefore, particle size and spacing). Kumar introduced a surface melt layer, purportedly representative of the AP surface, in an otherwise homogeneous solid. The most significant result of this model was a mechanism by which zero-exponent propellants could exhibit a positive combustion response and pressure effects[1]. Since Kumar did not treat in-depth hetergeneity, particle size effects appeared in the solid only through the effect of burn rate on melt layer thickness[2]. Cohen postulated two characteristic parameters for the solid phase,

---

1. Classical theories produce a response function proportional to pressure exponent, yet it is well known that plateau (zero exponent) and mesa (negative exponent) propellants have exhibited combustion instability. The AP melt may be analogous to the foam zone of such propellants.

2. An effect of burn rate on melt layer thickness would also appear in double-base propellants, so it cannot be the sole basis for the role of AP particle size in composite propellants. Furthermore, AP size effects persist at constant burning rate when catalysts are used or solids loading or distribution are varied.

---

20. Williams, F.A., and Lengelle, G., "Simplified Model for Effect of Solid Heterogeneity on Oscillatory Combustion", Astronautica Acta 14, 97-118 (1968).

21. Kumar, R.N., "Some Considerations in the Combustion of AP/Composite Propellants", Report under Contract NAS7-100, Guggenheim Jet Propulsion Center, California Institute of Technology, Pasadena, Calif. (Aug. 1972).

each dependent upon particle size. One was a measure of frequency response, the other of thermal response. He assumed that all response function curves could be determined by the pressure exponent plus these two parameters, and used experimental data for a standard propellant as calibration. The essential result is that decreasing particle size increases the peak response and shifts the peak to higher frequency. He further assumed that all multimodal propellants could be treated by linear superposition of the constituent results, and predicted multi-peaked response functions. Although the method is not founded upon a formal analysis, it is being used to guide propellant tailoring (15,22). Extensive applications have revealed some qualitative and quantitative deficiencies, for example, a tendency to over-emphasize the effect of fine sizes (22).

The gas phase proponents may be represented by Hamann (23), Glick and Condon (24), and Beckstead (25). All utilize some form of the "BDP" model of steady-state combustion (26) to represent the gas phase details, and none consider the solid to be heterogeneous. Hamann performed a perturbation analysis upon the entire BDP model, but did not report any results. Beckstead used the BDP model to calculate values for the parameters which are called for by the homogeneous theory of Denison & Baum (27). This approach of combining unrelated models is

---

22. Glick, R.L., Private Communications, Thiokol Corp., Huntsville, Ala. (1977).

23. Hamann, R.J., "Three Solid Propellant Combustion Models, A Comparison and Some Application to Non-Steady Cases", Memo. M-215, Delft University of Technology, Delft, Netherlands (Apr. 1974).

24. Condon, J. A., Osborn, J.R. and Glick, R.L., "Statistical Analysis of Polydisperse, Heterogeneious Propellant Combustion: Nonsteady-state", 13th JANNAF Combustion Meeting (CPIA Pub. 281, Vol. II, Dec. 1976) pp 209-223.

25. Beckstead, M. W., "Combustion Calculations for Composite Solid Propellants", ibid., 299-312.

26. Beckstead, M.W., Derr, R.L. and Price, C.F., "Model of Solid Propellant Combustion Based on Multiple Flames", J. AIAA 8, 2200-2207 (Dec. 1970).

(See next page for remainder of references.)

open to question (15), and a review of the technique reveals several deficiencies[3]. Glick and Condon employed a similar approach, but used a modified BDP model (28) for polydisperse particle size distributions and the method of Zeldovich and Novozhilov (29) as an alternative to the theory of Denison & Baum. Comparison of results with experimental data was disappointing. Considerable improvement was noted, however, when the Cohen postulates were incorporated into the method to position the peak response and peak response frequency (24).

The following concensus appears to emerge from this background. First, there is a need to account for the melt layer and the in-depth solid phase heterogeneity of composite propellants. Second, there is a need to provide an analytical basis to test, confirm or modify the Cohen postulates. Third, the representation of the gas phase also should address the heterogeneity of composite propellants by embodying some form of the BDP model rather than the homogeneity of the classical theories. Accordingly, it was the objective of this program to develop an analytical model of the combustion response of composite solid propellants with particular attention to these contributions of the propellant heterogeneity. Primary emphasis was placed on the modeling of heterogeneity in the solid phase.

---

3 For example, calculation of the "A" and "B" parameters of the homogeneous theory by this method reveals much too small a pressure effect to account for measured changes in the response function curve with pressure. As with Kumar, particle size effects appear only through changes in steady-state burning rate properties.

---

27. Denison, M. R. and Baum, E., "A Simplified Model of Unstable Burning in Solid Propellants", J. ARS 31, 1112-1122 (Aug. 1961).

28. Glick, R.L. and Condon, J.A., "Statistical Analysis of Polydisperse, Heterogeneous Propellant Combustion: Steady-State", 13th JANNAF Combustion Meeting, (CPIA Pub. 281, Vol. II, Dec. 1976) pp 313-345.

29. Novozhilov, B. V., "Nonstationary Combustion of Solid Rocket Fuels", Nauka, Moscow (1973).

SECTION 2

MODEL PREMISES AND ASSUMPTIONS

2.1 REPRESENTATION OF THE SOLID PHASE

The solid phase is represented by a "sideways sandwich", following the concept of Lengelle & Williams, as shown in Figure 1. This picture, really, does nothing more than state that the analysis is one-dimensional, so its dissimilarity to real propellants is of no greater concern than is the use of a one-dimensional treatment. Such a treatment assumes that the lateral processes are negligible in comparison to the normal processes. The solid is considered to be semi-infinite, having alternating layers of AP and binder. The thickness of the AP layers is nominally equal to the particle size, with exceptions to be noted later. The thickness of the binder layers is equal to the interstitial spacing as determined from the statistical geometry (26, 30). The surface AP layer contains a thin melt layer, which is justified experimentally (31-33), and follows the model of Kumar as a region of decomposition reactions in accordance with an Arrhenius law. The melt layer is "thin" in that the melting point of AP approximates the surface temperature during deflagration (32, 33). The analysis is linearized for small harmonic pressure perturbations, and is concerned with pressure-coupling only.

---

30. Cohen,N.S., Price, C.F. and Strand L.D., "Analytical Model of the Combustion of Multicomponent Solid Propellants", AIAA Paper 77-927, AIAA/SAE 13th Propulsion Conference (July, 1977).

31. Boggs, T.L., "The Decomposition, Pyrolysis and Deflagration of Ammonium Perchlorate", et seq., 7th JANNAF Combustion Conference (CPIA Publication 204, Vol. I, 1971) pp 113-138.

32. Guirao, C. and Williams, F.A., "A Model for Ammonium Perchlorate Deflagration Between 20 and 100 atm", J. AIAA 9, 1345-1356 (July 1971).

33. Beckstead, M.W. and Hightower, J.D., "Surface Temperature of Deflagrating Ammonium Perchlorate Crystals", J. AIAA 5, 1785-1790 (Oct. 1967).

Although the model contains the convective heating term to represent the regression of the material at the mean rate $\bar{r}$, the geometry of the layers is taken to be fixed with the AP layer always at the surface. This assumption is similar to the statistically fixed geometry used in the BDP model. One consequence of this assumption is that the "pulsation" mechanism associated with the sequential burn through of individual crystals and binder layers is excluded. The existence of a "pulsation" mechanism, coherent over microscopic regions of propellant surface, is open to question, although Lengelle & Williams (20) offer an argument in its favor. Thus, the present model examines only the effect of heterogeneity in relation to the thermal wave propagation through the solid propellant. A consequence of this assumption is that in-depth heterogeneity will not be important in those cases where the particle size is larger than the thermal wave (generally, coarse particles and high burn rate). Whether or not this consequence is unduly restrictive remains to be seen, but the fact that catalyzed coarse propellants are more stable than fine propellants would seem to permit it. The statistically fixed geometry requires the surface AP layer (including the melt) to have a thickness $\sqrt{2/3}$ of the particle size (26).

2.2 REPRESENTATION OF THE GAS PHASE

The function of the gas phase in the analytical scheme is to transform the oscillating pressure into an oscillating heat flux boundary condition at the surface of the AP melt. An approximate form of the BDP model has been selected to represent the gas phase, and the gas phase is assumed to be quasi-steady (viz, the gas phase responds instantly to changes in pressure). The approximation assumes a single flame above the propellant surface, where all gas phase reactions occur, as illustrated by Figure 1.

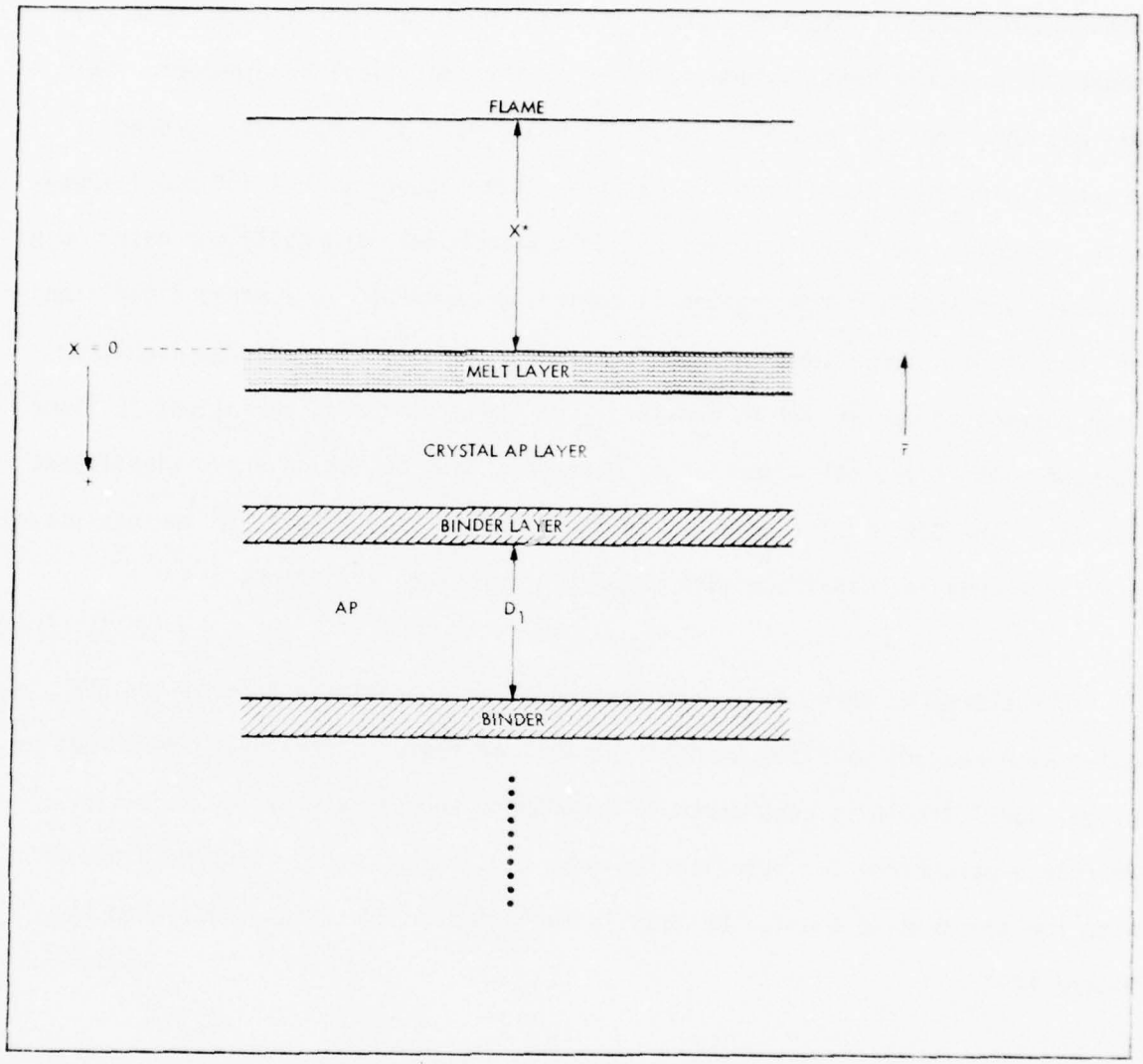


Figure 1. Physical Model Representation

Presuming that the condensed phase heterogeneity has the dominant effect upon the response function of composite propellants, it would not seem to matter what particular model were chosen to represent the gas so long as it provides a reasonable boundary condition. Thus, the Denison & Baum model could have been chosen to preserve some systematic order to the analytical development. This was not done for two reasons: First, it would have presumptously ignored those developments addressing the gas phase heterogeneity in a BDP model framework. Second, the fact that the Denison & Baum model is heavily dependent upon fluctuations in flame temperature is coming to be viewed as a serious deficiency of that model. Variations in flame temperature are fourth order in magnitude with respect to variations in pressure. On the other hand, variations in flame standoff, which are not addressed by Denison & Baum but which are a significant aspect of the BDP model (and a key to the particle size effects in the gas phase), are first order of magnitude with respect to pressure variations.

The approach, then, is a perturbation of an approximate form of the BDP model with respect to flame standoff as well as flame temperature. Perturbation of the model itself is considered to be proper, whereas use of the model to calculate parameters for substitution into a different model is open to question. With the approximate model, the task is much simpler than that undertaken by Hamann (23).

2.3 REPRESENTATION OF MULTIMODAL PROPELLANTS

Multimodal propellants are represented by adjacent columns of layers, as shown by Figure 2. Each particle size, corrected by the factor $\sqrt{2/3}$, tops a column. The remaining AP layers consist of the finest size in the distribution.

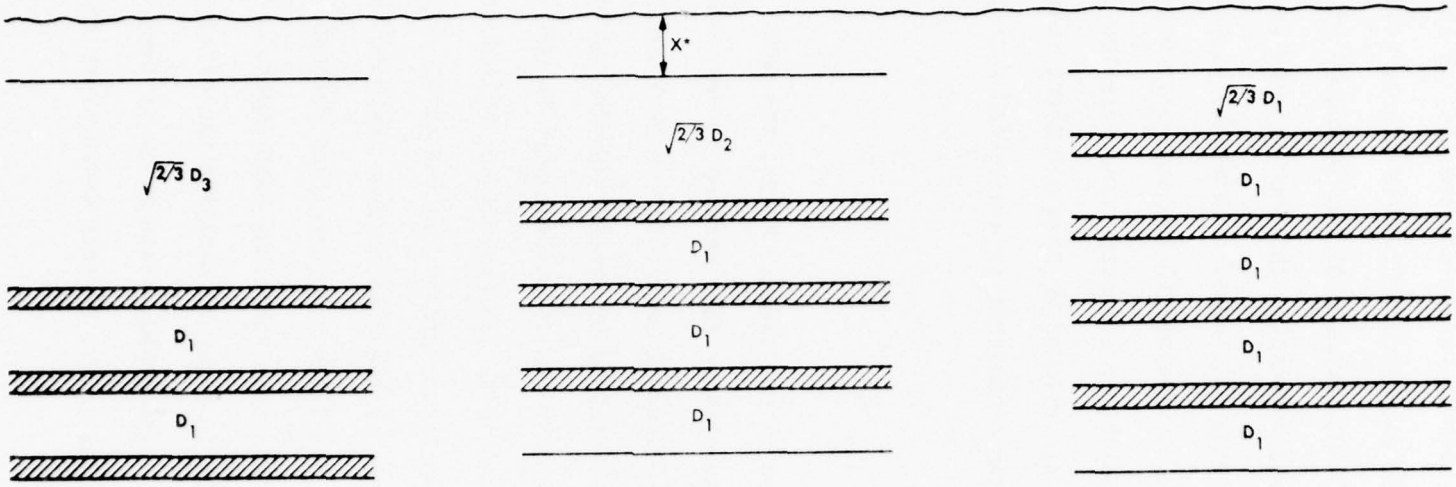


Figure 2. Layered Model of the Solid Phase for a Trimodal Propellant: Sizes D_1, D_2, D_3. Shaded Layers are Binder. The Top of the Surface AP Layer is a Thin Melt

This assumption is justified by the fact that the finest size will fill the smallest interstices between coarser particles, and will be valid so long as the thermal wave does not penetrate to a subsequent coarse sublayer. Validity is likely in practical propellants because of the influence of the fine size in raising the burning rate and decreasing the thermal wave penetration. The binder layers are all of the same thickness, as computed by the method of Ref. (30). All of the columns are constrained to the same mean burning rate, which implies one effective flame height or characteristic size for the diffusion flame. However, the columns are allowed to have different response functions and the aggregate value for the propellant is a weighted average from the constituent columns.

2.4 APPROACH

The analysis is performed in two parts. First, the model is derived in its steady-state version. The steady-state version serves to calculate mean values required for the time-dependent model, verify that the boundary conditions will be reasonably accurate as measued by the ability to reproduce experimental (or formal BDP) results, establish the zero-frequency (no oscillation) limit, verify some of the model assumptions, and provide an initial check on the method of solving for the temperature profile in the solid. The steady-state model is one of the subroutines of the computerized response function model. Secondly, the time-dependent model for the combustion response function is derived. In calculating the response function, experimental values of mean burning rate are used as input in order to minimize the effects of uncertainties in the steady-state modeling. The steady-state model is then limited to calculation of other steady-state quantities (e.g., surface temperature and flame height).

SECTION 3

THE STEADY-STATE MODEL

3.1 SOLID PHASE EQUATIONS

The heat conduction equation in the melt layer is written as:

$$k_a \frac{d^2T}{dx^2} + \rho_a c_a r \frac{dT}{dx} = W_a \rho_a Q_a A \exp(-E/RT) \tag{1}$$

where k = thermal conductivity
ρ = density
c = heat capacity
W = weight fraction
Q = heat of decomposition
subscript a = denotes AP
A = kinetics prefactor
E = activation energy
R = gas constant
r = burning rate
T = temperature
x = distance into the solid

The boundary conditions are:

$$x = 0, \; -k_a \frac{dT}{dx} = \rho_s c_s r(T_w - T_o) + W_a \rho_s r (Q_a + Q_m) \tag{2a}$$

$$x = x_m, \; -k_a \frac{dT}{dx} = \rho_s c_s r (T_m - T_o) + W_a \rho_s r Q_m \tag{2b}$$

where subscript w = denotes wall or surface
subscript s = denotes mean propellant
subscript m = denotes melt
subscript o = denotes the deep solid

Equation (1) may be written in dimensionless form as follows:

$$\frac{d^2\tau}{dy^2} + \frac{d\tau}{dy} = B \exp\left[\frac{-E/RT_w}{\left(\frac{T_w-T_o}{T_w}\right)(\tau-1)+1}\right] \tag{3}$$

where $\tau = (T-T_o)/T_w-T_o)$

$y = \frac{r}{\kappa} x$

κ = thermal diffusivity of the medium, κ_a for Eq. (3)

$B = A \frac{\kappa_a}{r^2} H_a \frac{c_s}{c_a}$

$H = \frac{WQ}{c_s(T_w-T_o)}$

In the thin melt layer, $\tau \sim 1$, so Eq. (3) may be approximated as:

$$\frac{d^2\tau}{dy^2} + \frac{d\tau}{dy} = C \exp\left[-D(1-\tau)\right] \tag{4}$$

where $C = B \exp(-E/RT_w)$

$D = \frac{E}{RT_w^2}(T_w-T_o)$

The boundary conditions become:

$$\text{surface:} \quad y = 0, \ \frac{d\tau}{dy} = -Z_a (1+H_a + H_m) \tag{5a}$$

$$\text{melt/crystal:} \quad y = y_m, \ \frac{d\tau}{dy} = -Z_a (\tau_m + H_m) \tag{5b}$$

where $Z_a = \frac{\rho_s c_s}{\rho_a c_a}$

Making the following transformation:

$$\xi = \tau + \frac{d\tau}{dy} \tag{6}$$

$$\eta = D(1-\tau) \tag{7}$$

Eq. (4) becomes:

$$(\xi - 1 + \eta/D)\ \frac{d\xi}{d\eta} = \frac{-C}{D}\ \exp\ (-\eta) \tag{8}$$

Recognizing $\eta/D << 1$ where $\tau \sim 1$, and letting $K = \frac{C}{D}$, Eq. (8) becomes:

$$-\xi\ \frac{d\xi}{d\eta} + \frac{d\xi}{d\eta} = K\ \exp(-\eta) \tag{9}$$

Eqs. (5a & b) become:

$$\eta = 0,\ \xi = \xi_w = 1 - Z_a(1 + H_a + H_m) \tag{10a}$$

$$\eta = \eta_m,\ \xi = \xi_m = \tau_m - Z_a\ (\tau_m + H_m) \tag{10b}$$

Eq. (9) may be integrated in closed form. Applying Eqs.(10) and some algebra yields an expression for burning rate in the following form:

$$K = \frac{\xi_w\left(\frac{\xi_w}{2} - 1\right) - \xi_m\left(\frac{\xi_m}{2} - 1\right)}{1 - \exp\ (-\eta_m)} \tag{11}$$

Making all substitutions, burning rate is expressed as a function of wall temperature and other constants as follows:

$$r^2 = \frac{\frac{2\kappa_a A W_a Q_a}{C_a(T_w - T_o)} \frac{RT_w^2}{E(T_w - T_o)} \exp(-\frac{E}{RT_w})\ \left[1 - \exp(-\frac{E}{RT_w}\ \frac{T_w - T_m}{T_w})\right]}{\left\{Z_a\left[1 + \frac{W_a(Q_a + Q_m)}{c_s(T_w - T_o)}\right]\right\}^2 - \left\{(\frac{T_m - T_o}{T_w - T_o})(1 - Z_a) - Z_a\ \frac{W_a Q_m}{c_s(T_w - T_o)} - 1\right\}^2} \tag{12}$$

This relation is to be matched with a relation between burn rate and wall temperature from the gas phase model.

Analysis of the steady-state problem and, as will appear later, solution of the time-dependent problem, also require a description of the steady-state temperature profile in the solid. The temperature profile in the melt layer is determined from Eq. (9). Integration and application of Eq.(10a) yields:

$$\xi = 1 \pm \sqrt{1 + 2[K \exp(-\eta) - C_M]} \tag{13}$$

$$\text{where } C_M = K - \frac{1}{2}\left[1 - Z_a(1+H_a+H_m)\right]^2 + 1 - Z(1+H_a+H_m)$$

The physics of the temperature decay requires the negative root. Substituting the definitions for ξ and τ yields:

$$\frac{d\eta}{dy} = -\eta + D\sqrt{1+2\left[K\exp(-\eta) - C_M\right]} \tag{14}$$

An order of magnitude analysis shows that the square root term will always be much larger than η in the melt layer. Thus, Eq. (14) reduces to the form:

$$\frac{d\eta}{\sqrt{a + b\exp(-\eta)}} = D\,dy \tag{15}$$

$$\text{where } a = 1 - 2C_M$$

$$b = 2K$$

This equation can be integrated in closed form for $y(\eta)$; it is transcendental as $\eta = \eta(y)$.

$$y = \frac{-1}{D\sqrt{a}} \ln\left[\frac{1+z - \sqrt{z^2 + 2z}}{1+d - \sqrt{d^2+2d}}\right] \tag{16}$$

$$\text{where } d = 2a/b$$

$$z = d\exp(\eta)$$

The melt layer thickness may be calculated by evaluating Eq. (16) at $\eta=\eta_m$.

For all layers beneath the melt layer, the right-hand-side of Eq. (3) vanishes. The gradient condition at the top of each layer is similar to Eq. (5b), without the heat of fusion.

$$y = y_{Top},\ \frac{d\tau}{dy} = -Z\,\tau_{Top} \tag{17}$$

$$\text{where } Z = Z_a \text{ when entering an AP layer}$$

$$Z = Z_b \text{ when entering a binder layer}$$

Thus the temperatures in each layer beneath the melt layer follow the recurring form:

$$\tau = \tau_{Top} - Z\,\tau_{Top}\,[1-\exp(-\Delta y)] \qquad (18)$$

where Δy = thickness of the particular layer, and uses the appropriate κ.

For a homogeneous propellant, Z=1, so Eq. (18) properly reduces to the result for a homogeneous propellant. Note also that the temperature and the gradient will properly tend to zero together, as y approaches infinity.

3.2 GAS PHASE EQUATIONS

If it is assumed that all reactions occur at the flame height, the heat conduction equation has the form of Eq. (1) with a zero right-hand-side. Taking the flame to be at x=0 and the wall at the flame height (x=x*), the temperature distribution is:

$$T = T_f - (T_f - T_w)\frac{1-\exp(-\zeta)}{1-\exp(-\zeta^*)} \qquad (19)$$

where $\zeta = ux/\kappa_g$

u = gas velocity normal to the surface

subscript g = denotes gas

subscript f = denotes flame

For convenience, ζ is set equal to y by employing the continuity relation, $\rho_g u = \rho_s r$, and assuming that the ratio of heat capacity to thermal conductivity is equal for gas and solid[4]. Then the gradient at the wall may be written, from differentiation of Eq. (16), as:

$$y=y^*;\ \frac{dT}{dy} = -(T_f - T_w)\,\frac{\exp(-y^*)}{1-\exp(-y^*)} \qquad (20)$$

where $y^* = \dfrac{rx^*}{\kappa_s}$

4 This assumption is comparable to the general assumption that these properties are temperature-insensitive. The conductivities of propellant gases may be calculated, but are not well-known.

At the same time, the gradient at the wall must satisfy the energy requirements of the condensed phase:

$$-k_g \frac{dT}{dx} = \rho_s c_s r(T_w - T_o) + W_a \rho_s r (Q_a + Q_m) + W_b \rho_s r Q_b \tag{21}$$

where subscript b = denotes binder.

Eq. (21), when compared to Eq. (2a), states that the binder heat of decomposition is being positioned on the gas side of the wall. The binder does not appear at the surface in the model of Fig. 1, but does in reality exist over portions of the surface. The model surface is the AP melt. Therefore, in the framework of this model, the decomposing binder is external to the melt layer so may be represented by a heat absorption on the gas side. Eq. (21) may be re-written as:

$$y = y^*, \; -\frac{dT}{dy} = (T_w - T_o) + F \tag{22}$$

$$\text{where } F = (H_a + H_m + H_b)(T_w - T_o)$$

Equating Eq. (20) and Eq. (22) yields:

$$T_w = (T_f + F - T_o) \exp(-y^*) - (F - T_o) \tag{23}$$

which is the required matching relation in burn rate and wall temperature.

The remaining unknown, x^*, is determined by an approximate fit of the effective flame height from the BDP model:

$$x^* = C_F \frac{r D_1}{p} \tag{24}$$

where D_1 = particle size

p = pressure

C_F = 24.6 for r in cm/sec, p in atmospheres, D_1 in microns and x^* in microns.

This form does not take into consideration the relative importance of the components of the BDP multiple flame structure and their differing dependencies upon the parameters of Eq. (21). However, it is a reasonable representation selected for mathematical convenience and with the perturbation analysis in mind.

After substitution of Eq. (21) into Eq. (18), Eqs. (12) and (20) are two equations for the unknowns r and T_w. These equations are solved by iteration. Once r and T_w are known, it is possible to calculate the melt layer thickness and the thermal profile in the solid.

The steady-state model is not intended to be used to calculate burning rates for multimodal propellants. That would require some definition of an effective particle size or flame height for use of Eq. (4). Rather, experimental values of burning rate are used to determine the wall temperature from the solid phase model, whence the effective flame height and particle size may be determined from the gas phase model. This is the general method for determining the mean values for use in the response function model. An option is provided to predict these values for unimodal propellants, but is used in this work only to validate aspects of the modeling. Those results are discussed in Section 5.

Note that the effects of particle size appear only as gas phase effects in the steady-state model. This conforms with a generally-accepted view of steady-state combustion (34). Although Eq. (12) (from the solid phase) may influence the magnitude of the particle size effect, particle size does not appear in that equation. The thermal profile in the solid does not enter into the calculation for mean burning rate. However, this does not preclude the importance of solid phase effects in determining the role of particle size in the time-dependent model for the response function.

34. Derr, R.L., "Review of Workshop on Steady State Combustion Modeling of Composite Solid Propellants", 7th JANNAF Combustion Meeting (CPIA Publication 204, Vol. I, 1971) pp 1-8.

SECTION 4

THE TIME-DEPENDENT MODEL: CALCULATION OF THE RESPONSE FUNCTION

4.1 SOLID PHASE EQUATIONS

Using dimensionless quantities as previously defined, the time-dependent heat conduction equation in the thin melt layer is written as:

$$\frac{\partial^2 \tau}{\partial y^2} + \frac{\partial \tau}{\partial y} - \frac{\kappa a}{r^2} \frac{\partial \tau}{\partial t} = C \exp[-D(1-\tau)] \tag{25}$$

Eq. (25) is the time-dependent form of Eq. (4). Denoting mean values as barred and perturbed values as primed, and employing Eq. (4) to describe the mean portion, Eq. (25) may be written as:

$$\frac{\partial^2 \tau'}{\partial y^2} + \frac{\partial \tau'}{\partial y} - \Omega \frac{\partial \tau'}{\partial (\omega t)} = C \exp(\bar{\eta}) \; [\exp(D\tau')-1] \tag{26}$$

where ω = frequency of oscillations

$$\Omega = \frac{\kappa}{r^2} \omega$$

For harmonic perturbations, $\tau' = \exp(i\omega t)$, the time-dependent term of Eq. (26) may be re-written to provide an ordinary differential equation:

$$\frac{d^2 \tau'}{dy^2} + \frac{d\tau'}{dy} - i\Omega\tau' = C \exp(-\bar{\eta}) \; [\exp(D\tau')-1] \tag{27}$$

If τ' is of second order of magnitude and Ω sufficiently small, Eq. (27) may be approximated as:

$$\frac{d^2 \tau'}{dy^2} + \frac{d\tau'}{dy} = CD\tau' \exp(-\bar{\eta}) \tag{28}$$

The problem is linearized to small perturbations, and is therefore restricted to linear instability. Variations in burning rate are of the order of variations in pressure, but variations in surface temperature are second or third order with respect to variations in burning rate. Thus, for second-order pressure perturbations, τ' is at most of third order. The exponential in $\bar{n}$ is approximately unity in the melt layer, and the product CD is of the order 10^2. This would permit frequencies as high as 10KHz to satisfy the approximation for cases of interest. Since the quasi-steady assumption for the gas-phase model will also restrict the frequencies to values less than 10KHz, use of Eq. (28) is consistent with that assumption.

In applying the boundary conditions at the surface, one must recognize that the surface is fluctuating relative to the mean surface (y=0) position. Since the fluctuating burning rate will also be of the form, $\exp(i\omega t)$, in the linearized problem, it follows that the surface position is given by:

$$y_S = -\frac{i}{\Omega}\frac{r'}{\bar{r}} \tag{29}$$

The boundary conditions are:

$$y = y_S, \qquad \tau' = \tau_w' \tag{30a}$$

$$\frac{d\tau'}{dy} = Z_a\left(g_w' + \frac{r'}{\bar{r}} H_b\right) \tag{30b}$$

where g_w' = the gradient in τ' at the surface, on the negative (gas) side of the boundary.

Eq. (30b) is derived by substituting Eq. (2a) into Eq. (21), and perturbing in the non-dimensional form. For small perturbations, the quantities at the actual surface may be related to the quantities at the mean surface as:

$$y = y_S, \qquad \tau' = \tau'|_{y=0} - \frac{i}{\Omega}\frac{r'}{\bar{r}}\,\bar{g}|_{y=0} \tag{31a}$$

$$\frac{d\tau'}{dy} = g'|_{y=0} - \frac{i}{\Omega}\frac{r'}{\bar{r}}\frac{d\bar{g}}{dy}|_{y=0} \tag{31b}$$

where $\bar{g}$ = the gradient in $\bar{\tau}$

Eq. (28) is integrated numerically. The functions $\exp(-\bar{\eta})$ and $\bar{g}$ are available in terms of y from Eq. (16). The integration provides the mean surface values, which are then converted to the actual surface values by means of Eqs. (31). The actual surface values must satisfy Eqs. (30); g'_w is determined from the gas phase model, and τ'_w is related to r' by perturbing Eq. (12). The perturbation of Eq. (12) yields:

$$\frac{r'}{\bar{r}} = V_5\tau'_w \tag{32}$$

$$\text{where } V_5 = \left(1+\frac{\theta}{2}\right)\chi + \frac{D}{2}\,\frac{1-2\chi(1-\tau_m)}{\exp[D(1-\tau_m)]-1} - \frac{1}{Z_a(1+H_a+H_m)-\tau_m(1-Z_a)+Z_aH_m+1}$$

$$\theta = E/(R\bar{T}_w)$$

$$\chi = (\bar{T}_w - T_o)/\bar{T}_w$$

Although the solution appears to require iteration, it will be shown subsequently that it does not (cf. Subsection 4.3).

For all layers beneath the melt layer, the right-hand-side of Eq. (27) vanishes so the time-derivative term cannot be neglected. Thus:

$$\frac{d^2\tau'}{dy^2} + \frac{d\tau'}{dy} - i\Omega\tau' = 0 \tag{33}$$

The boundary conditions at each interface are generally:

$$y=y_{Top},\ \tau'=\tau'_{Top} \tag{34a}$$

$$\frac{d\tau'}{dy} = \frac{Z_{a,b}}{Z_{b,a}}\ g'_{Top} \tag{34b}$$

where the first Z subscript is used when entering an AP layer and the second is used when entering a binder layer. Assuming that the product ρc for the melt is equal to that for the solid AP, the ratio of Z does not appear for that first layer of solid AP which joins the melt. The g'_{Top} refers to that value of the perturbed gradient on the negative (upper) side of the boundary, which drives the behavior below. The τ'_{Top}, however, is preserved on the positive side. Eq. (34b) is similar in form to Eq. (30b), but there is no phase change heat beneath the melt layer and the sublayers do not oscillate relative to the mean surface. Eq. (33) may be integrated to produce the following recurring formulas:

$$\tau'=\tau'_{Top}\exp(\lambda_1\Delta y) - \frac{\lambda_1\tau'_{Top} - \frac{Z_{a,b}}{Z_{b,a}}\ g'_{Top}}{\lambda_2-\lambda_1}\left[\exp(\lambda_2\Delta y)-\exp(\lambda_1\Delta y)\right] \tag{35}$$

$$\frac{d\tau'}{dy} = \lambda_1\tau'_{Top}\exp(\lambda_1\Delta y) - \frac{\lambda_1\tau'_{Top} - \frac{Z_{a,b}}{Z_{b,a}}\ g'_{Top}}{\lambda_2-\lambda_1}\left[\lambda_2\exp(\lambda_2\Delta y)-\lambda_1\exp(\lambda_1\Delta y)\right] \tag{36}$$

where $\lambda_1 = -\frac{1}{2} - \frac{1}{2}\sqrt{1+4i\Omega}$

$$\lambda_2 = -\frac{1}{2} + \frac{1}{2}\sqrt{1+4i\Omega}$$

Each layer uses the appropriate ratio of Z, and the appropriate κ for Δy, λ_1 and λ_2.

Eqs. (35) and (36) properly reduce to the results for a homogeneous solid ($Z=1$, and the difference between g' and $\lambda_1\tau'$ vanishes throughout, leaving only the first term on the right-hand-side). However, these equations retain the growing exponential term (in λ_2, of positive real part) for the layered solid because the formulation of the problem does not explicitly impose the boundary condition at infinity upon each layer. This condition is the requirement that τ' and $d\tau'/dy$ vanish together. For the homogeneous solid, this condition would immediately set the exponential in λ_2 equal to zero. For the layered solid, it can only be said that a layer will eventually be reached where this condition may be approximated for all practical purposes. Since the effect of the heterogeneity on the perturbations also disappears when the perturbations disappear, this approximate condition may be expressed in the form of the homogeneous solid:

$$\tau' \to 0, \quad \frac{d\tau'}{dy} \to \lambda_1\tau' \tag{37}$$

In other words, there is a depth below which the propellant can be treated as homogeneous for purposes of the perturbation problem. Consider that this occurs below the $N^{\underline{th}}$ AP layer. Then Eq. (37) may be expressed as:

$$y=y_N, \quad \tau' = \varepsilon \tag{38a}$$

$$\frac{d\tau'}{dy} = \lambda_{1_S}\,\varepsilon \tag{38b}$$

where ε = an extremely small number

λ_{1_S} = value of λ_1 using κ_S for the homogeneous region.

Eq. (38b) represents the perturbed gradient at the top of the homogeneous region, on the positive (homogeneous) side of the boundary with the $N^{\underline{th}}$ AP layer. This may be converted to the perturbed gradient on the negative (AP) side of the

boundary through use of Eq. (34b), recognizing that Z=1 in the homogeneous region. Thus, Eqs. (38) become:

$$y = y_N, \quad \tau'_{TopN} = \varepsilon \tag{39a}$$

$$g'_{TopN} = Z_a \lambda_{1s} \varepsilon \tag{39b}$$

where subscript TopN = denotes the bottom of the Nth AP layer, which would be the "top" condition for the succeeding homogeneous region were the calculation to continue.

Thus, the problem of the solid phase is now properly closed. Eqs. (30) define the conditions at the "top", and Eqs. (39) define the conditions at the "bottom". The conditions at the bottom are now known, but the conditions at the top depend upon the gas phase. Additional discussion with respect to the solution for the solid phase is deferred to Subsection 4.3.

4.2 GAS PHASE EQUATIONS

The perturbed form of Eq. (24) is simply:

$$\frac{x^{*'}}{\bar{x}^*} = \frac{r'}{\bar{r}} - \frac{p'}{\bar{p}} \tag{40}$$

Since the analysis will take into account flame temperature perturbations, a relation between the flame temperature perturbations and the wall temperature perturbations is required. This is obtained by perturbation of Eq. (23). Using dimensionless quantities as previously defined, Eq. (40) for the flame standoff perturbations, and linearizing the exponential in perturbed quantities, there results:

$$\tau'_f = \tau'_w \exp(\bar{y}^*) - (1+\tau_F)\bar{y}^* \exp(\bar{y}^*)\left[\frac{p'}{\bar{p}} - 2\,\frac{r'}{\bar{r}}\right] \tag{41}$$

where $\tau_F = \dfrac{F}{\bar{T}_w - T_o}$

The perturbed gradient on the gas side of the wall is derived from an energy balance. The energy being transmitted into the solid is the difference between the heat release in the gas and the energy required to raise the gas from the wall temperature to the flame temperature:

$$-k_g \frac{dT}{dx} = -\rho_s r Q_f - c_g(T_f - T_w) \tag{42}$$

where Q_f = heat release in the gas (negative for an exotherm).

Using the same relation between gas and solid thermal properties as led to Eq. (20), Eq. (42) may be written in dimensionless form as follows:

$$\bar{g}_w = H_f + (\bar{\tau}_f - 1) \tag{43}$$

where $\bar{g}_w$ = gradient in $\bar{\tau}$ at the wall on the negative (gas) side

$$H_f = \frac{Q_f}{c_g(\bar{T}_w - T_o)}$$

The perturbed dimensionless form of Eq. (42) is:

$$g_w' = \frac{r'}{\bar{r}} [H_f + (\bar{\tau}_f - 1)] + (\tau_f' - \tau_w') \tag{44}$$

H_f may be eliminated by combining the relations in Eqs. (43), (5a) and (21)[5].

5 For this purpose, Eq. (21) may be written as: $d\bar{\tau}/dy = Z(\bar{g}_w + H_b)$. This is the steady-state analog of Eq. (30b), and comes from substituting Eq. (2a) into Eq. (21) in dimensionless form.

$$H_f = -(\bar{\tau}_f - 1) - (1+\tau_F) \tag{45}$$

Substitution into Eq. (44) yields:

$$g_w' = -\frac{r'}{\bar{r}}(1+\tau_F) + (\tau_f' - \tau_w') \tag{46}$$

Substitution of Eq. (41) into Eq. (46) yields, after combining terms:

$$g_w' = \frac{r'}{\bar{r}}(1+\tau_F)(2\bar{y}^*\exp(\bar{y}^*)-1) + \tau_w'(\exp(\bar{y}^*)-1) - \frac{p'}{\bar{p}}(1+\tau_F)\bar{y}^*\exp(\bar{y}^*) \tag{47}$$

Eq. (47) is the necessary matching relation for Eq. (30b). Thus, the formulation of the time-dependent problem is complete.

Eq. (45) also may be substituted into Eq. (43) to yield:

$$\bar{g}_w = -(1+\tau_F) \tag{48}$$

Eq. (48) is the dimensionless form of Eq. (22), so consistency is verified in this respect.

4.3 SOLUTION FOR THE RESPONSE FUNCTION

By combination of Eqs. (30b), (31a), (31b), (32) and (47), it is possible to derive an explicit expression for the response function in terms of solid phase constants, gas phase constants and one key parameter characteristic of the solution for the solid phase. This key parameter is the ratio of the perturbed gradient to the perturbed temperature at the mean surface:

$$\left.\frac{g'}{\tau'}\right|_{y=0} \equiv K_2 \tag{49}$$

The components of this ratio appear individually in Eqs. (31), and the ratio is defined here as K_2. In the classical homogeneous theory, this ratio is always λ_1. In this work, the ratio will depend upon the modeled heterogeneity and λ_2 as well as on λ_1.

The expression for the response function is:

$$R = \frac{\Omega_a V_{6B}}{\Omega_a \left[\frac{V_{6A}}{V_5} + 2V_{6B} - \frac{V_3}{Z_a} \right] + i \frac{C}{Z_a} - \frac{K_2}{Z_a} \left[\frac{\Omega_a}{V_5} - i V_3 \right]} \tag{50}$$

Where R = response function

$V_3 = Z_a(1+H_a+H_m)$ (see Eq. [5a])

V_5 = (see Eq. [32])

C = (see Eq. [4])

V_{6A} = $(\exp(\bar{y}^*)-1)$ (see Eq. [47], appears as coefficient of τ_w')

V_{6B} = $(1+\tau_F)\bar{y}^*\exp(\bar{y}^*)$ (see Eq. [47], appears as coefficient of $p'/\bar{p}$ and as part of the coefficient of $r'/\bar{r}$).

If the ratio K_2 depended upon g_w' and τ_w' separately, the problem would require an iterative solution. However, it turns out that K_2 is an intrinsic property of the solid phase, independent of the surface boundary condition, just as is λ_1 for the homogeneous solid. The reason is that the solid phase is described by linear homogeneous differential equations. All solutions of such equations of second order are of the form:

$$f = C_1 f_1(x) + C_2 f_2(x)$$

where f = denotes functions

C_1 = first arbitrary constant

C_2 = second arbitrary constant

The gradient is:

$$g = C_1 f_1'(x) + C_2 f_2'(x)$$

It is required that g/f go to λ_1 in the deep solid. This constitutes one of the two boundary conditions that can be imposed. Therefore, C_1 and C_2 cannot be independent. Solving for C_2 in terms of C_1:

$$C_2 = C_1 \frac{f_1'(\infty) - \lambda_1 f_1(\infty)}{\lambda_1 f_2(\infty) - f_2'(\infty)}$$

Substitution of this expression for C_2 into the ratio, g/f, followed by algebraic simplification, leads to the result that g/f depends upon the ratio C_2/C_1 and not upon C_1 and C_2 independently. Of course, g/f will vary from case to case and will vary with x for a given case; the important point is that it depends only on the ratio C_2/C_1 and x. This property was first observed numerically in the course of a prototypal computer program which was based upon an iterative scheme. Accordingly, it suffices to go through Eqs. (35) and (36) in the layers, beginning with Eqs. (38)[6], and then solve numerically through the melt layer, just one time for K_2[7]. Knowing K_2, Eq. (50) for R is solved by complex arithemtic[8].

It is of interest to examine the form of Eq. (50) in comparison to the form obtained from the homogeneous theory (Ref. 16). The latter can be written as:

$$R = \frac{\Omega n A B}{\Omega[AB-(1+A)] + iA - \lambda_1[\Omega - iA]} \quad (51)$$

where n = pressure exponent

A = a solid phase parameter, not to be confused with A as defined in Eq. (1)

B = a gas phase parameter, not to be confused with B as defined in Eq. (3)

λ_1 = as defined in Eqs. (35, 36); it is of opposite sign in Ref. (16).

6 Subroutine LAYRSP of the computer program.
7 Subroutine MLTLRP of the computer program.
8 Subroutine GLTFP of the computer program.

It is noted that Eqs. (50) and (51) are identical in form and, to some extent, they are similar in substance. The parameter V_{6B} combines condensed phase and gas phase terms, as does (nAB) of Eq. (51). The parameters C and V_3, although different, are condensed phase terms as is A of Eq. (51). The parameter V_{6A} is a gas phase term, as is B of Eq. (51). The parameter V_5 is related to C, and, therefore, a part of the analogy to A; but important differences from A derive from the finite melt layer. The parameter K_2 is the heterogeneous analog of λ_1. The parameter Z_a is purely a consequence of the heterogeneity, so there would be no analog for it in Eq. (51). The identity in form, and the similarity in substance, suggests that the heterogenity as described by this model will not produce radical changes in the qualitative behavior of the response function.

The zero-frequency limit of Eq. (51) is the pressure-exponent, n. Pressure exponent does not appear explicitly as such in the steady-state model described herein. Nevertheless, it is of interest to examine the zero-frequency limit of Eq. (50). It is readily apparent that a non-zero response function at zero frequency requires that the following relationship be satisfied:

$$\lim_{\Omega_a \to 0} K_2 = -\frac{C}{V_3} \tag{52}$$

The satisfaction of Eq. (52) is verified by combining Eqs. (31), (32), (5a), the derivative of Eq. (15) applied at y=0, and a perturbation of Eq. (2a). In general, K_2 must be solved numerically, but at zero frequency it is possible to derive an expression which reduces to $-C/V_3$. Since Eq. (52) is satisfied, the indeterminate form of Eq. (50) that results may be evaluated to yield the non-zero response function:

$$\lim_{\Omega_a \to 0} R = \frac{V_{6B}}{2V_{6B} + \frac{1}{V_5}(1+V_{6A})} \tag{53}$$

Eq. (53) may be thought of as an "effective" pressure exponent, extracting an implicit property of the model. Note that it depends upon condensed phase terms as well as gas phase terms. Numerically, it is found to be consistent with pressure-dependence as calculated from results of the steady-state model.

SECTION 5

MODEL RESULTS

5.1 THE STEADY-STATE MODEL

Results of the steady-state model have been obtained in order to evaluate some of the important model premises. The essential results are tabulated in Table I.

Table I presents various results, compared with data and with BDP model results, for A-13 propellant used as a standard case. The first set of results compares burning rate as a function of pressure. The model results compare very well with the data. It should be emphasized, however, that these results should not be construed to imply that this model is "better than" the BDP model. The second set of results compares surface temperature. Experimental values of surface temperature are reportedly in the neighborhood of 850°K (26). It is observed that this model produces higher surface temperatures and a somewhat greater sensitivity to pressure than does the BDP model. However, the results are reasonable. The results also confirm the assumption that variations in surface temperature are second order (or smaller) with respect to variations in burning rate. The third set of results compares flame standoff distance. This model uses one flame. The BDP results are for the primary flame (sum of diffusion and reaction heights), and the values to the right of the slash are for the AP monopropellant flame when that flame moves closer to the surface than the primary flame. When that happens, the BDP model employs an energy partitioning which may be thought of as some single flame having an effective height between the two shown. On that basis, the flame heights from this model are roughly a factor of 3 greater than from the BDP model, but the qualitative behavior with pressure is the same. The value of C_F in Eq. (24) was adjusted to achieve good agreement with the burning rate data; values of other constants

TABLE I

COMPARISONS OF BURNING RATE AND OTHER QUANTITIES FOR A-13 PROPELLANT

BURNING RATE

PRESSURE (PSIA)	DATA (cm/sec)	FROM BDP MODEL	FROM THIS MODEL
100	0.27	0.29	0.28
300	0.41	0.39	0.47
500	0.52	0.47	0.55
700	0.62	0.53	0.65
900	0.73	0.60	0.74
1100	0.85	0.65	0.83

WALL TEMPERATURE

100	---	$824^{o}K$	864
300	---	836	900
500	---	844	937
700	---	851	973
900	---	855	982
1100	---	859	986

FLAME HEIGHT

100	---	25.6μm	82.9
300	---	20.7	50.8
500	---	19.9/18.3	35.7
700	---	19.5/5.7	30.2
900	---	19.4/3.1	25.8
1100	---	19.3/3.1	23.6

are the same as used in the BDP model. It is concluded that, for purposes of the time-dependent analysis, the model conforms reasonably well with steady-state reality.

The ability to reproduce measured effects of AP particle size on burning rate was tested with a series of propellants analogous to A-13. These propellants were the subject of a low pressure L* instability study performed by Ramohalli (35). The comparison of burning rates at 100 psia is as follows:

PARTICLE SIZE (μm)	DATA (cm/sec)	MODEL (cm/sec)
40	0.41	0.38
90	0.27	0.28
200	0.23	0.19
360	0.19	0.13

Again, the agreement is reasonable.

There are two other aspects of the steady-state model results which merit discussion: The melt layer and the heterogeneity in relation to the thermal wave.

The melt layer thickness is computed to be of the order of microns or less, which is consistent with experimental observation and the thin melt layer assumption. Its dependence upon heating rate involves a tradeoff between surface temperature and the steepness of the thermal gradient. Theoretically, it will disappear at such low burning rate that the surface temperature does not reach the AP melting point, and also will approach zero at very high burning rate where the gradient is very steep. Although the layer is thin, it was considered improper to neglect it for mathematical convenience because the characteristic time of its dimension corresponds to the high frequencies of interest.

---

35. Kumar, R. N. and McNamara, R. P., "Some Experiments Related to L-Star Instability in Rocket Motors", AIAA Paper 73-1300 (Nov., 1973).

For particle sizes in excess of 40μm, the thermal wave will not penetrate the first AP layer under conditions of interest. The implication is that, except for the melt layer, the solid can be considered homogeneous in determining its role. However, this is not true for the fine sizes which are generally utilized in practical propellants. An estimate for a 2μm AP propellant reveals that, at 1000 psi, the temperature does not fall to within 10 percent of the bulk temperature until about 5 pairs of AP-binder layers are traversed. Further, if the 2μm AP is a component of a multimodal propellant, the burn rate will be lower such that the thermal wave will penetrate more layers of the column consisting of the 2μm AP. As a result, it appears that the role of solid phase heterogeneity will be limited to melt layer heterogeneity in the intermediate-coarse size regime, but that in-depth heterogeneity can be important in the fine size regime. This distinction is one consequence of the present fixed-geometry model; were the layers permitted to move to evoke the pulsation mechanism, then the in-depth heterogeneity could be important for all sizes. The distinction was considered significant in view of the experimental importance of fine AP (15).

5.2 THE TIME-DEPENDENT MODEL

The effects of the solid phase heterogeneities are most likely to appear at combinations of fine AP and low burning rate. Therefore, a test case consisting of a 2μm AP propellant at a burning rate of 0.47 cm/sec was selected for evaluation. Except for the particle size, this test case would correspond to A-13 propellant at 300 psi. Results are shown in Figure 3. The solid line is for the heterogeneous propellant. The long-dash line is for AP and binder thermal properties equal to mean propellant thermal properties; therefore, it is for melt layer

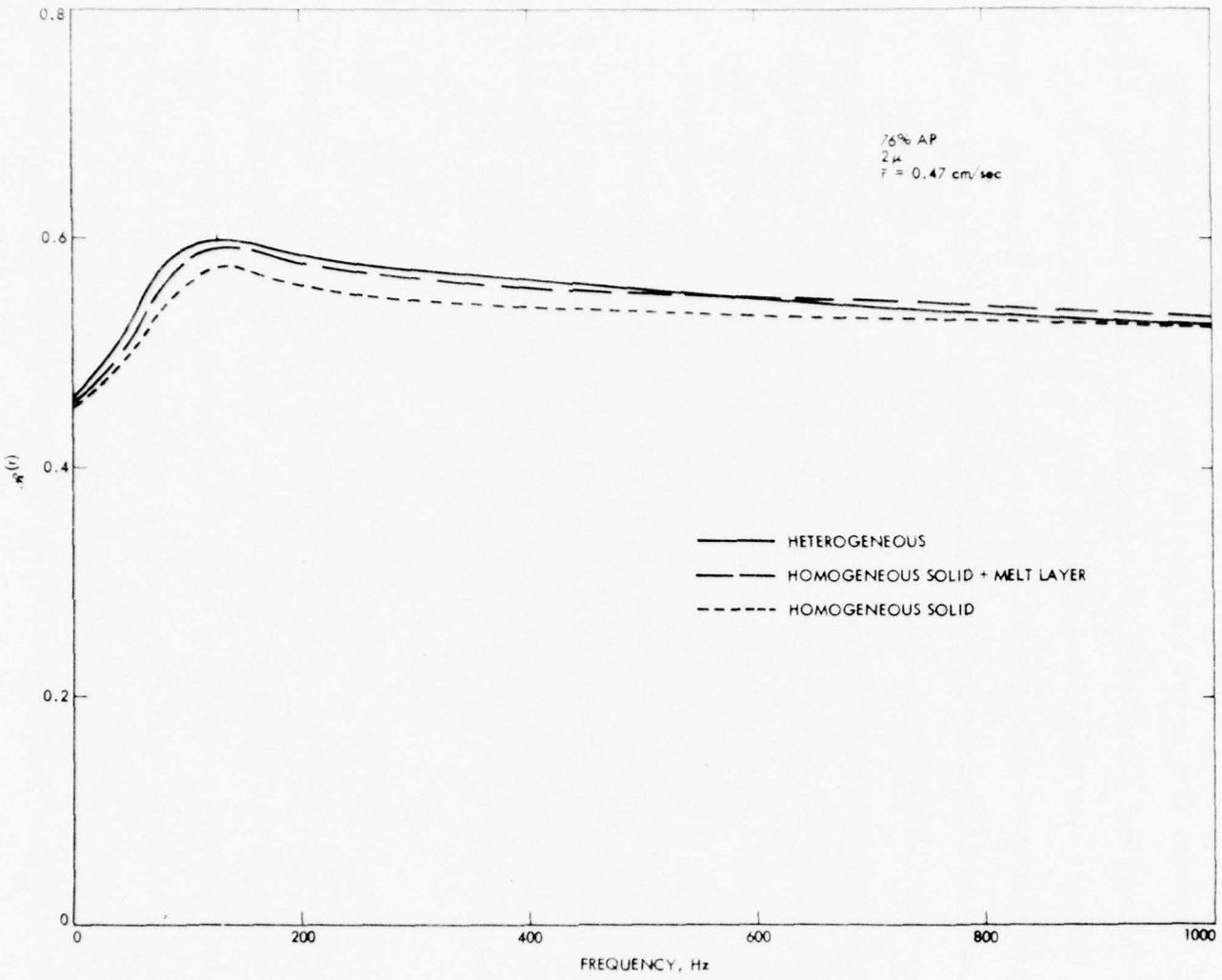


Figure 3. Response Function Calculations for a 2 μAP Analog of A-13 Propellant

heterogeneity only, the propellant below the melt layer being homogeneous[9]. The short-dash line is for a completely homogeneous solid; i.e., the distributed heat release in the finite melt layer is now concentrated at the surface only and the entire region beneath the surface is homogeneous[10]. It is observed that the effects of the heterogeneities are small.

Figure 4 compares results for a 90μm AP propellant, which is A-13 propellant, with the Figure 3 results. Thus, the effect of particle size at a constant burning rate is shown. In the framework of this model, a constant burning rate implies a constant wall temperature and constant dimensionless flame properties; thus, any difference is due to solid phase heterogeneities. An effect of the heterogeneity does appear, but again, it is small. It is noted that the results for A-13 are virtually identical to the homogeneous solution displayed in Figure 3. In the case of A-13, the thermal wave does not penetrate the surface AP layer and the melt layer thickness is about 1 percent of the particle size; thus, the solid is homogeneous for all practical purposes. In the case of the 2μm propellant the thermal wave penetrates 15 AP layers and the melt layer thickness is about one-third (1/3) of the particle size; thus, the solid is heterogeneous, but the effect of the heterogeneity appears to be small. Significantly, the effect is small with respect to peak response

---

9 $k_a = k_b = k_s$; $\rho_a = \rho_b = \rho_s$; $c_a = c_b = c_s$; $Z = 1$.

10 $V_3 = C = Z = 1$; $K_2 = \lambda_1$; $V_5 = \Theta \cdot \chi$. This would correspond to the Denison and Baum model except for differences in the modeling of the gas phase, and differences in the values of combustion constants due to the use of this model (including the finite melt layer) to reproduce steady-state burning rates.

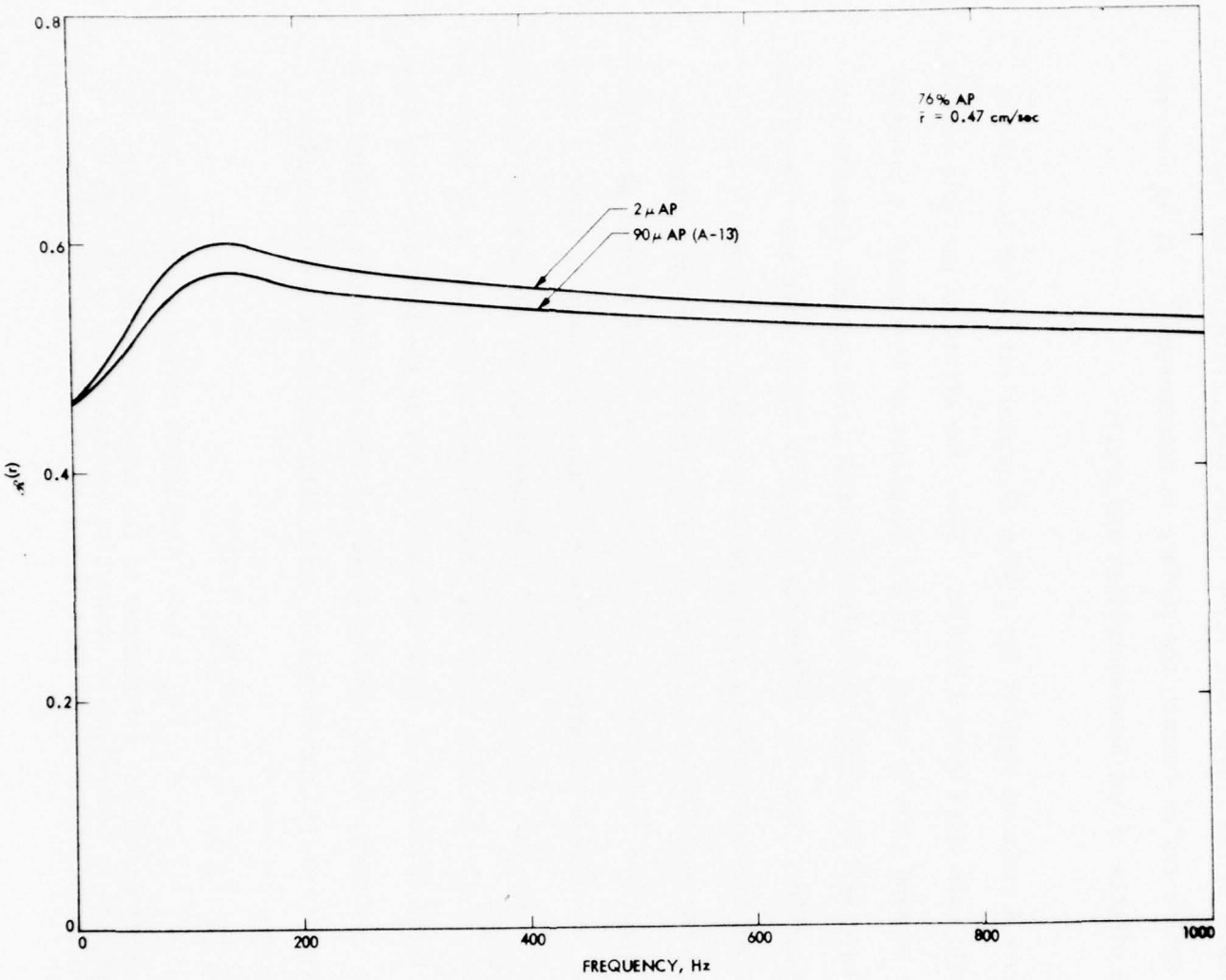


Figure 4. Effect of AP Particle Size at Constant Burning Rate for A-13 Type Propellant

frequency as well as magnitude. Thus, it can be concluded that the expected effects of the heterogeneity are not being represented by this model.

The theoretical results for the A-13 propellant, moreover, do not agree with experimental data. The A-13 is a JANNAF standard propellant, and has been well-characterized in T-burner testing (Ref. 36). The experimental peak response is about 4 and occurs at roughly 300 Hz. Thereafter, the response declines to a value of approximately 1.5 at 1000 Hz. The theoretical results presented in Figure 4 show a peak response of 0.58 at 125 Hz, and a value of 0.51 at 1000 Hz. Therefore, the theory shows a relatively slight peak and at too low a frequency. Such theoretical results are a consequence of the combustion paramters, analogous to the "A" and "B" (or "α") parameters of the Denison & Baum model. Presumably, the agreement with data could be improved by selection of a different set of parameters. However, a ground rule of this study was that the parameters would be predetermined by considerations of a credible steady-state model. Given that model, it is inappropriate to change the constants arbitrarily. According to Ref. (27), a relatively large value of the "B" (or "α") parameter will constrain the time-dependent model to the prediction of small peaks, and a relatively small value of the "A" parameter will produce low peak response frequencies. By analogy, that is the situation here. Since it would be inappropriate to juggle parameters, it must be concluded that there is a mechanistic deficiency in the time-dependent model that has been postulated here.

Figure 5 presents theoretical results showing the effect of burning rate for a constant particle size. The 2μm AP propellant was selected for this illustration.

---

36. "T-Burner Manual", CPIA Publication 191 (Nov., 1969).

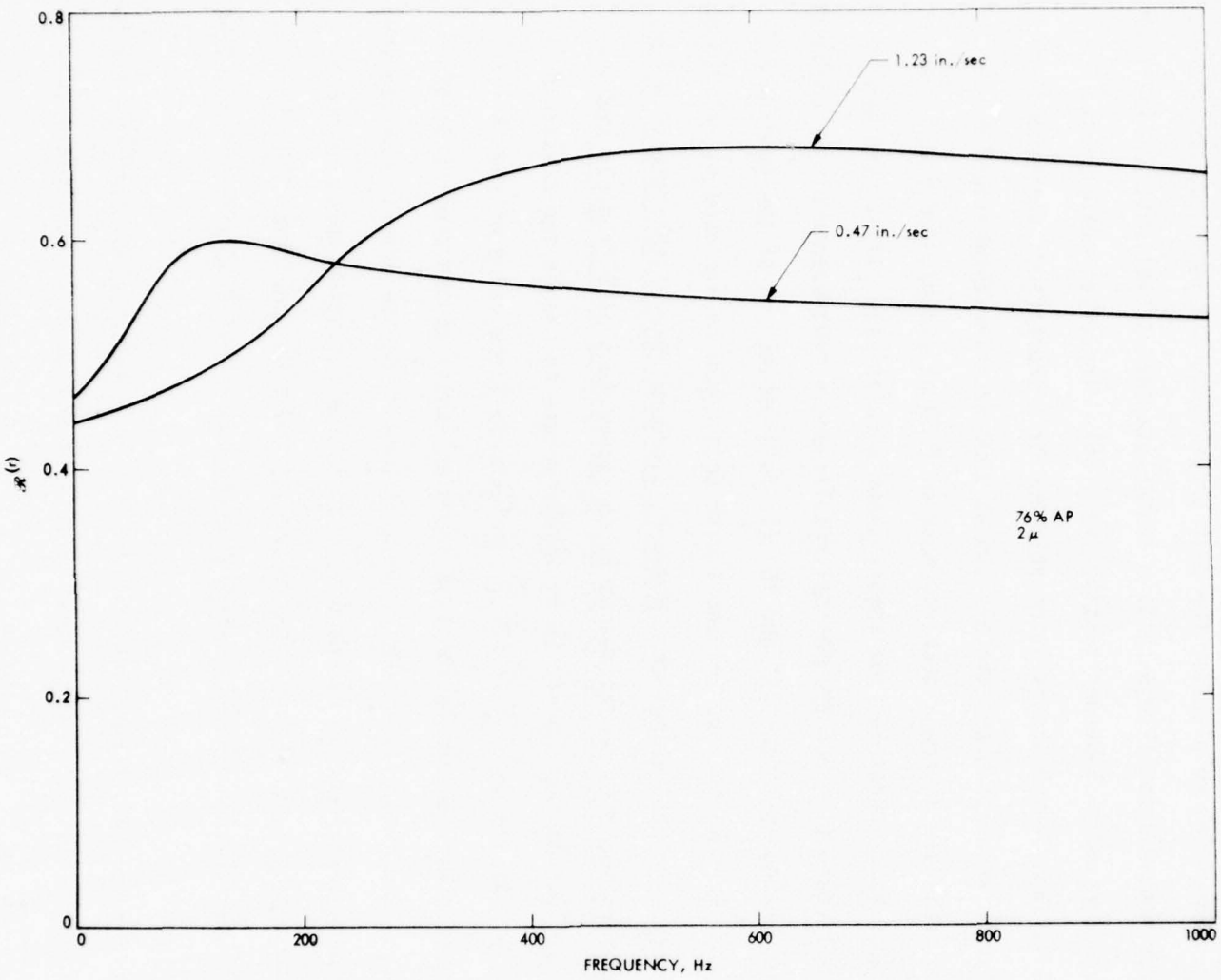


Figure 5. Effect of Burning Rate at Constant AP Particle Size for the 2μ Analog of A-13 Propellant

The higher burning rate is representative of this propellant; the lower burning rate may be thought of as a suppression for purposes of Figures 3 and 4. The effect on peak response frequency demonstrates further that the modeled heterogeneities are of little consequence. It is observed that the peak response frequency varies nearly with the square of the burning rate, which is the result for homogeneous propellants. On the other hand, the Figure 3 and 4 results show that the effect of the heterogeneity on peak response frequency is about 10 percent. According to the Cohen postulates for the effect of heterogeneity, the peak response frequency in Figure 5 should vary with the first power of burning rate, and in Figure 4 it should vary inversely with particle size. These effects are not being produced by this model.

Figure 6 compares theoretical results with experimental data for a bimodal propellant. Although the shape of the theoretical curve is reasonable, the peak response magnitude and frequency are again underpredicted. Also, the zero frequency limit is overpredicted, reflecting a deviation from the measured steady-state pressure exponent. It appears that further work is necessary in order to implement a proper mechanism for the combustion response.

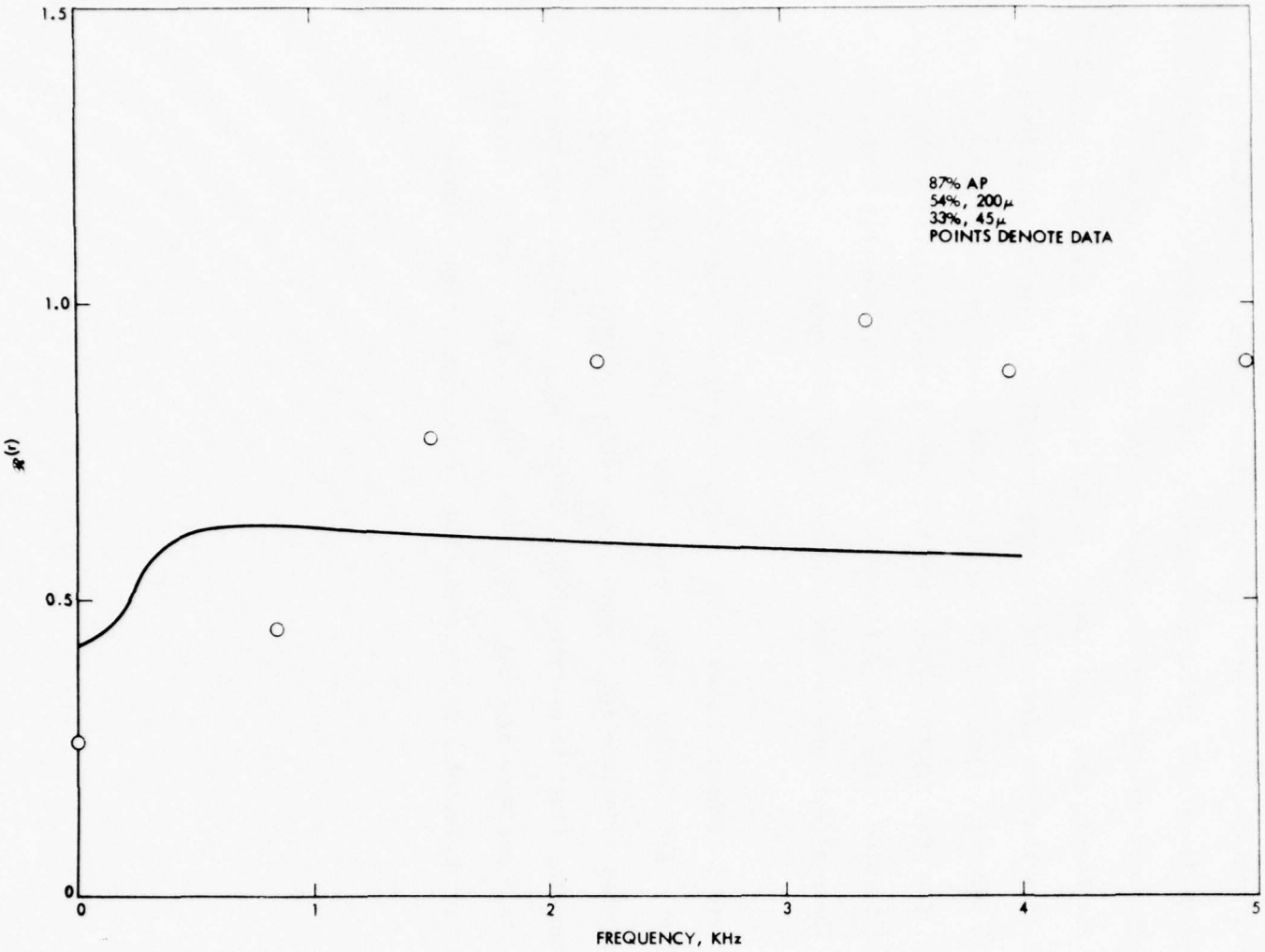


Figure 6. Comparison of Theory and Experiment for SP-540 Propellant at 500 psi

SECTION 6

CONCLUSIONS AND RECOMMENDATIONS

An analytical model has been developed which incorporates mechanisms of solid phase and gas phase heterogeneities into the calculation of steady-state and linear time-dependent combustion properties of composite solid propellants. Although the model satisfactorily describes the steady-state combustion properties, it is deficient in describing the time-dependent combustion response characteristics in several respects. Use of a consistent set of combustion constants produces peak response magnitudes and frequencies which are too low in comparison to experimental data and which are not significantly affected by the AP particle size per se. Although an effect of the solid phase heterogeneities is predicted by this model, the effect is quantitatively so small as to allow it to be neglected in future work. Therefore, the role of AP cannot be attributed to the solid phase alone unless some other mechanism is incorporated into the theory. It is recommended that the concept of moving layers be re-examined, including justification for the coherence of such a mechanism. It is further recommended that the perturbed BDP model be examined as a potentially useful way in which to represent the heterogeneity of the gas phase. It is desired not only to achieve the effects of the heterogeneity, but also to justify a set of values of the combustion constants that will properly position the response function curve. It appears necessary to modify both the solid phase and gas phase models in order to achieve those purposes in a consistent manner.

SECTION 7

REFERENCES

1. Green, L.G., "Effects of Oxidizer Concentration and Particle Size on Resonance Burning of Composite Solid Propellants", Jet Prop. 28, 159-164 (Mar., 1958).

2. Strand, L.D., "Low Pressure L* Instability and Extinction", 3rd ICRPG Combustion Conference (CPIA Publication 138, Vol. I, Feb., 1967) pp. 195-207.

3. "Experimental Studies on the Oscillatory Combustion of Solid Propellants", Report NWC-TP-4393, U.S. Naval Weapons Center, China Lake, CA (Mar., 1969).

4. Beckstead, M.W., Boggs, T.L. and Madden, O.H., "The Effect of Oxidizer Particle Size and Binder Type on Nonacoustic Instability", AIAA Paper 69-175 (1969).

5. Boggs, T.L. and Beckstead, M.W., "Failure of Existing Theories to Correlate Experimental Nonacoustic Combustion Instability Data", J. AIAA 8, 626-631 (Apr., 1970).

6. Crump, J.E., "Combustion Instability in a Series of AP-HTPB Smokeless Propellants", 8th JANNAF Combustion Meeting (CPIA Publication 220, Vol. II, Dec. 1971) pp. 81-90.

7. Crump, J.E., "Combustion Instability Studies on Non-metallized AP-HTPB Propellants", 9th JANNAF Combustion Meeting (CPIA Publication 231, Vol. III, Dec., 1972) pp 123-134.

8. Wendelken, C.P., "Combustion Stability Characteristics of Solid Propellants", AFRPL-TR-73-63, Air Force Rocket Propulsion Laboratory, Edwards, CA. (Oct., 1973).

9. "Aluminum Behavior in Solid Propellant Combustion", AFRPL-TR-74-13, Lockheed Propulsion Company, Redlands, CA (May, 1974).

10. Micheli, P.L., "Stabilization of Smokeless Propellants with Additives", 11th JANNAF Combustion Meeting (CPIA Publication 261, Vol. III, Dec., 1974) pp 123-136.

11. Anderson, F.A. and Kumar, R.N., "Feasibility Study of Propellants and Igniters for Shuttle Solid Rocket Booster Separation Motors", Report 900-710, Jet Propulsion Laboratory, Pasadena, CA (June, 1975).

12. Cohen, N.S., et al., "Design of a Smokeless Solid Rocket Motor Emphasizing Combustion Stability", 12th JANNAF Combustion Meeting (CPIA Publication 273, Vol. II, Dec. 1975) pp. 205-220.

13. Horton, M.D. and Rice, D.W., "The Effects of Compositional Variables Upon Oscillatory Combustion of Solid Rocket Propellants", Combustion and Flame 8, 21-28 (Mar. 1964).

14. "Control of Solids Distribution in HTPB Propellants", Contract F04611-76-C-0006, Hercules Inc., Allegheny Ballistics Laboratory, Cumberland, MD (in progress).

15. Cohen, N.S., "Report of Workshop on Combustion Instability of Smokeless Propellants", Proceedings of 14th JANNAF Combustion Meeting (to be published).

16. Culick, F.E.C., "A Review of Calculations for Unsteady Burning of a Solid Propellant", J. AIAA 6, 2241-2254 (Dec., 1968).

17. "T-Burner Testing of Metallized Solid Propellants", AFRPL-TR-74-28, edited by F.E.C. Culick, Air Force Rocket Propulsion Laboratory, Edwards, CA (Oct., 1974).

18. "T-Burner Motor Verification Program Final Report", AFRPL-TR-74-71, Aerojet Solid Propulsion Company, Sacramento, CA (Jan., 1975).

19. Lovine, R.L. and Linfor, J.J., "Measurement of Propellant Response at High Frequency", 13th JANNAF Combustion Meeting (CPIA Publication 281, Vol. II, Dec. 1976) pp 95-112.

20. Williams, F.A. and Lengelle, G., "Simplified Model for Effect of Solid Heterogeneity on Oscillatory Combustion", Astronautica Acta 14, 97-118 (1968).

21. Kumar, R.N., "Some Considerations in the Combustion of AP/Composite Propellants", Report under Contract NAS 7-100, Guggenheim Jet Propulsion Center, California Institute of Technology, Pasadena, CA (Aug., 1972).

22. Glick, R.L., Private Communications, Thiokol Corp., Huntsville, AL (1977).

23. Hamann, R.J., "Three Solid Propellant Combination Models, A Comparison and Some Application to Non-Steady Cases", Memo. M-215, Delft University of Technology, Delft, Netherlands (Apr., 1974).

24. Condon, J.A., Osborn, J.R. and Glick, R.L., "Statistical Analysis of Polydisperse, Heterogeneous Propellant Combustion: Nonsteady-state", 13th JANNAF Combustion Meeting (CPIA Publication 281, Vol. II, Dec., 1976) pp 209-223.

25. Beckstead, M.W., "Combustion Calculations for Composite Solid Propellants", ibid., 299-312.

26. Beckstead, M.W., Derr, R.L. and Price, C.F., "A Model of Solid Propellant Combustion Based on Multiple Flames", J. AIAA 8, 2200-2207 (Dec., 1970).

27. Denison, M.R. and Baum, E., "A Simplified Model of Unstable Burning in Solid Propellants", J. ARS 31, 1112-1122 (Aug., 1961).

28. Glick, R.L. and Condon, J.A., "Statistical Analysis of Polydisperse, Heterogeneous Propellant Combustion: Steady-State", 13th JANNAF Combustion Meeting (CPIA Publication 281, Vol. II, Dec., 1976) pp 313-345.

29. Novozhilov, B.V., "Nonstationary Combustion of Solid Rocket Fuels", Nauka, Moscow (1973).

30. Cohen, N.S., Price, C.F. and Strand, L.D., "Analytical Model of the Combustion of Multicomponent Solid Propellants", AIAA Paper 77-927, AIAA/SAE 13th Propulsion Conference (July, 1977).

31. Boggs, T.L., "The Decomposition, Pyrolysis and Deflagration of Ammonium Perchlorate", et seq., 7th JANNAF Combustion Conference (CPIA Publication 204, Vol. I, 1971) pp 113-138.

32. Guirao, C. and Williams, F.A., "A Model for Ammonium Perchlorate Deflagration Between 20 and 100 atm", J. AIAA 9, 1345-1356 (July, 1971).

33. Beckstead, M.W. and Hightower, J.D., "Surface Temperature of Deflagrating Ammonium Perchlorate Crystals", J. AIAA 5, 1785-1790 (Oct., 1967).

34. Derr, R.L., "Review of Workshop on Steady State Combustion Modeling of Composite Solid Propellants", 7th JANNAF Combustion Meeting (CPIA Publication 204, Vol. I, 1971) pp 1-8.

35. Kumar, R.N. and McNamara, R.P., "Some Experiments Related to L-Star Instability in Rocket Motors", AIAA Paper 73-1300 (Nov., 1973).

36. "T-Burner Manual", CPIA Publication 191 (Nov., 1969).

APPENDIX A

COMPOSITE SOLID PROPELLANT COMBUSTION RESPONSE MODEL COMPUTER PROGRAM

This appendix presents information essential to the utilization of the composite Solid Propellant Combustion Response Model (CSP CRM) computer program. This information is presented under the following headings and in the order listed:

1. Definitions of Input Data
2. Input Deck Corresponding to the Example Problem
3. Definitions of Output Data
4. Card Image Listing of Program.
5. Description of Program Logic, Subroutines and Use of Text Equations
6. Solutions to Example Problem

A-1. DEFINITIONS OF INPUT DATA

All input to the Composite Solid Propellant Combustion Response Model Computer Program is made by means of a NAMELIST/INPUT statement and an associated read statement, both of which appear in the DATAIN subroutine. The FØRTRAN names of the input variables and parameters, together with their corresponding algebraic symbols and brief definitions are presented below in the exact order that they appear in the INPUT list:

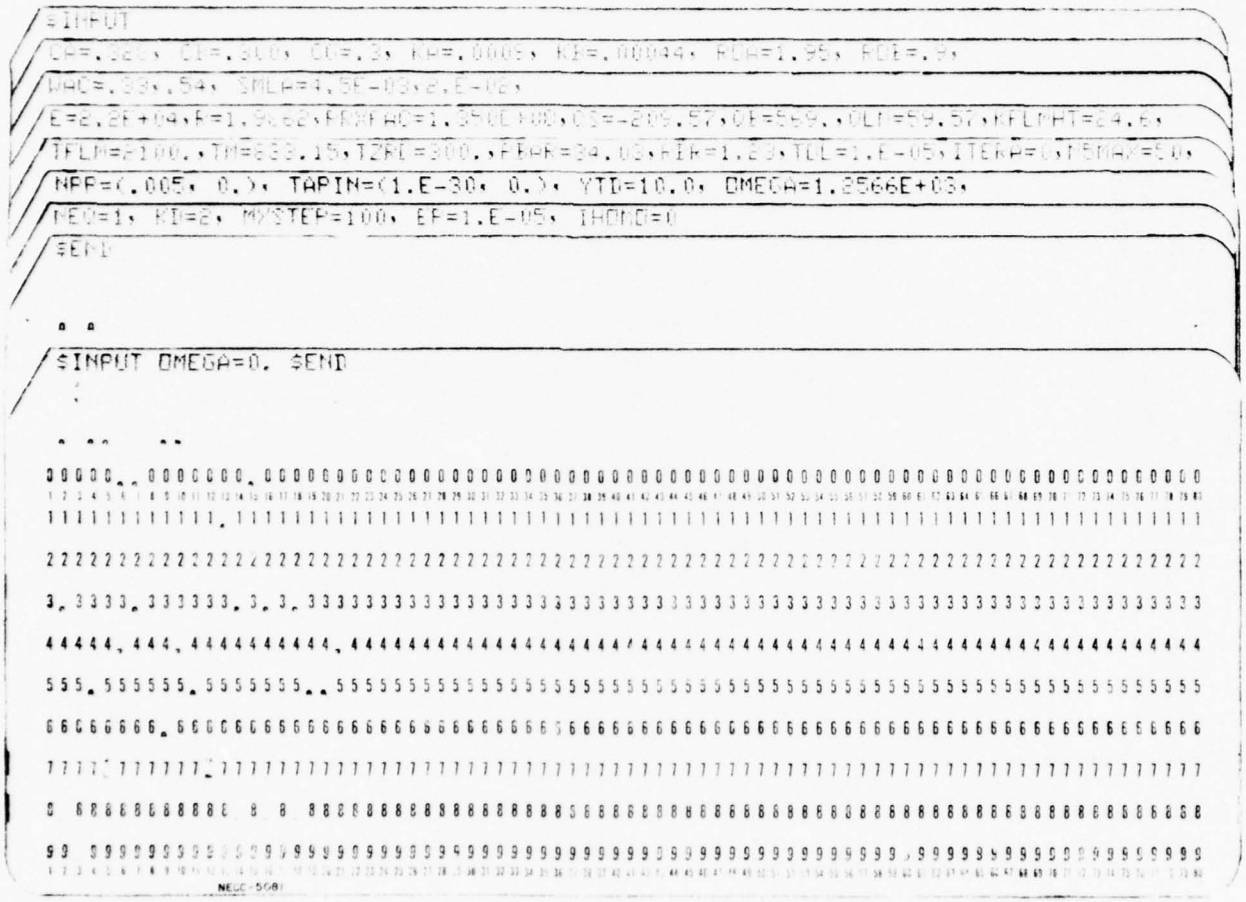
FØRTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
CA	C_a	specific heat at constant pressure for the oxidizer (cal g^{-1} $°K^{-1}$, real)
CB	C_b	specific heat at constant pressure for the binder (cal g^{-1} $°K^{-1}$, real)
CG	C_g	specific heat at constant pressure for the combustion gas mixture that exists between the gas-condensed phase interface and the flame front (cal g^{-1} $°K^{-1}$, real)
KA	k_a	thermal conductivity of the oxidizer (cal sec^{-1} cm^{-1} $°K^{-1}$, real)
KB	k_b	thermal conductivity of the binder (cal sec^{-1} cm^{-1} $°K^{-1}$, real)
RØA	ρ_a	density of the oxidizer (g cm^{-3}, real)
RØB	ρ_b	density of the binder (g cm^{-3}, real)
WAC(J)	$W_a(j)$	fraction of composite solid propellant mass attributable to the j th size component of the oxidizer ($j \leq 4$) (dimensionless, real)
SMLA(J)	a_j or D_j	diameter of j th size component oxidizer particles ($j \leq 4$) (cm, real) Note: Must be listed in order of increasing particle size.
E	E	activation energy for decomposition of the oxidizer (cal. g-$mole^{-1}$, real)
R	R	universal gas constant (cal. g-$mole^{-1}$ $°K^{-1}$, real)

FORTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
PRXFAC	A	prefactor in the Arrhenius expression for oxidizer decomposition (sec^{-1}, real)
QS	Q_a	gross heat of decomposition of the oxidizer (equals net heat of decomposition of the oxidizer minus latent heat of melting of the oxidizer) cal g^{-1}, real)
QB	Q_b	heat of decomposition of the binder (cal g^{-1}, real)
QLM	Q_m	latent heat of melting of the oxidizer (cal g^{-1}, real)
KFLMHT	CF	CF = $\bar{\rho}x^*/(\bar{r}D_1)$ constant of proportionality in the rational algebraic approximation to BDP effective flame height (atmos sec μm^{-1}, real)
TFLM	T_f	adiabatic temperature of the gas-phase combustion flame sheet (°K, real)
TM	T_m	melting temperature of the oxidizer (°K, real)
TZRØ	T_o	temperature in the composite solid propellant far from the gas-condensed phase interface (°K, real)
PBAR	$\bar{p}$	temporal mean pressure in the gas phase--- assumed to be uniform (atmos, real)
RBR	$\bar{r}$	temporal mean rate of regression of the composite solid propellant; no input if ITERA=1 (cm sec^{-1}, real)
TØL	-	arbitrary tolerance employed in the steady-state part of the computer program to evaluate convergence of the steady-state solution (dimensionless, real, usually $10^{-3} \leq TØL \leq 10^{-5}$)
ITERA	-	specifies the mode of operation of the computer program: If ITERA=0, the program determines the combustion response parameters for a CSP whose temporal mean regression rate is specified; if ITERA=1, the program determines the temporal mean regression rate for a propellant whose basic physical and chemical characteristics have been specified (dimensionless, integer).

FØRTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
N5MAX	-	if ITERA=1 and if a satisfactory value for RBR has not been achieved after N5MAX-1 one-percent increments or decrements of RBR, the solution will be terminated and an appropriate comment will be printed in the computer output (dimensionless, integer, $5 \leq N5MAX \leq 20$) Note: This mode of program termination indicates an injudicious initial choice for RBR.
NPP	$p'/\bar{p}$	normalized pressure perturbation (dimensionless, polar complex) Note: This is the exciter of the perturbation; as such, its phase is arbitrarily set equal to zero
TAPIN	$\tau'_{y>>o}$	normalized temperature perturbation applied in the solid phase at a distance from the gas-condensed phase interface that is several times the characteristic transient thermal depth within the solid. (dimensionless polar complex) Note: Since the ratio, $(g'/\tau')_{y=o}$, obtained from the solid phase solution is independent of the argument assumed for τ', the argument is ordinarily arbitrarily set equal to zero.
YTD		factor which, when multipled by SQRT (KAPS/ØMEGA), provides a value of x>o, viz., XTD (dimensionless, usually assumed to be an integer ≥3) Note: SQRT (KAPS/ØMEGA) is a measure of the depth of penetration of the perturbation into the CSP
ØMEGA	ω	assumed angular frequency of the perturbation (radians sec $^{-1}$, real)
NEQ	-	number of differential equations solved in subroutine SVDQ---always equal to one in this computer program (dimensionless integer)
KD	-	order of the differential equations solved in subroutine SVDQ--- always equal to two in this computer program (dimensionless, integer)
MXSTEP	-	maximum number of steps allowed between output points in subroutine SVDQ----usually set equal to one hundred in this computer program (dimensionless, integer)
EP	-	parameter used to control local error in subroutine SVDQ---refer to description of subroutine for guidance in setting the value of this parameter (dimensionless, real)

FØRTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
IHØMØ	-	When IHOMO=0, certain pertinent physical and thermal properties of the propellant binder, oxidizer, and gaseous products specified in namelist/input/are generally distinct; however, when IHOMO=1, these properties are set equal to corresponding properties of the oxidizer. Thus, specifying IHOMO=1 implies a homogeneous propellant (dimensionless, integer)

A-2. INPUT DECK CORRESPONDING TO THE EXAMPLE COMPUTER RUN



Example Problem
WAC = .33, .54
SMLA = 4.5E-03,
2.E-02,---
OMEGA = 1.2566E+03

OMEGA = 0. causes the computer run to be terminated.

A-3. DEFINITIONS OF OUTPUT DATA

All permanent output of the Composite Solid Propellant Combustion Response Model (CSP CRM) Computer Program is made by means of NAMELIST/ØUTPUT statements and corresponding associated write statements. All of the NAMELIST/ØUTPUT statements and associated write statements appear in the DTAØUT.

Although some temporary NAMELIST/TPØUT statements and associated write statements are located in various intermediate subprograms of the CSP CRM Computer Program, the FØRTRAN names therein are, with rare and inconsequential exception, reprinted in one or the other of the write statements associated with NAMELIST/ØUTPUT statements. For this reason, only FØRTRAN names associated with permanent output are presented here.

Each NAMELIST/ØUTPT and, within each list, each FØRTRAN name appearing in DTAØUT are presented below in the exact order of appearance in the resulting print-out.

NAMELIST/ØUTPT 1

FORTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
JMAX	j_{max}	maximum number of size components of the oxidizer ($j_{max} \leq 4$) (dimensionless, integer)
WA	W_a	$W_a = \Sigma W_a(j)$ fraction of solid propellant mass attributable to all size components of the oxidizer (dimensionless, real)
DNM	-	denominator in formula for SMLB in subroutine DATAIN $\Sigma[W_a(j)/a_j]$ (cm^{-1}, real)
WB	W_b	$W_b = 1.-W_a$ fraction of solid propellant mass attributable to the binder (dimensionless, real)
VFA	ν	volume of the oxidizer expressed as a fraction of solid propellant volume (dimensionless, real)
SMLATP	a_{top}	$a_{top} = \sqrt{\frac{2}{3}}\ A_{j_m}$ Hermance equivalent mean size of the oxidizer size component assumed to exist in the topmost lamina of the solid propellant (j_m) (cm, real)

FØRTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
DLTYTP	$\Delta\bar{y}_{top}$	$\Delta\bar{y}_{top} = a_{top}\,\bar{r}/\kappa_a$ thickness of any oxidizer lamina other than the topmost (dimensionless, real)
SMLB	b	$b=W_a\{[\pi/6\nu)]^{1/3} - \sqrt{2/3}\}/\Sigma[W_a(j)/a_j]$ idealized thickness of a binder lamina calculated on the basis of oxidizer packing factor (cm, real)
DLTYBB	$\Delta\bar{y}_b$	$\Delta\bar{y}_b=b\bar{r}/\kappa_b$ thickness of any binder lamina (dimensionless, real)
KS	k_s	bulk equivalent thermal conductivity of the CSP (cal sec^{-1} cm^{-1} $°K^{-1}$, real)
RØS	ρ_s	bulk mean density of the CSP (g cm^{-3}, real)
CS	C_s	bulk equivalent specific heat at constant pressure of the CSP (cal g^{-1} $°K^{-1}$, real)
KAPA	κ_a	$\kappa_a=k_a/\rho_a C_a)$ thermal diffusivity of the oxidizer (cm^2 sec^{-1}, real)
KAPB	κ_b	$\kappa_b=k_s/(\rho_s C_s)$ bulk equivalent thermal diffusivity of the CSP (cm^2 sec^{-1}, real)
KAPS	κ_s	$\kappa_s=k_s/(\rho_s C_s)$ bulk equivalent thermal diffusivity of the CSP (cm^2 sec^{-1}, real)
Ƶ	Ƶ	$Ƶ=\rho_s C_s/(\rho_a C_a)$ ratio of thermal capacity per unit volume of the bulk CSP to that of the oxidizer (dimensionless, real)
Ƶ'	Ƶ'	$Ƶ'=\rho_s C_s/(\rho_b C_b)$ ratio of thermal capacity per unit volume of the bulk CSP to that of the binder (dimensionless, real)

NAMELIST/ØUTPT 2

FORTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
THETAA	θ_a	$\theta_a = E/(R\bar{T}_w)$ (dimensionless, real)
CHI	χ	$\chi = \frac{\bar{T}_w - T_o}{\bar{T}_w}$ (dimensionless, real)
D	D	$D = \theta_a \cdot \chi$ (dimensionless, real)
QSDMLS	H_a	$H_a = W_a Q_a / [C_a(\bar{T}_w - T_o)]$ (dimensionless, real)
QMDMLS	H_m	$H_m = W_a Q_m / [C_a(\bar{T}_w - T_o)]$ (dimensionless, real)
FDMLS	τ_f	$\tau_f = \bar{r}/(\bar{T}_w - T_o)$ (dimensionless, real)
B	B	$B = \kappa_a A W_a Q_a / [C_a(\bar{T}_w - T_o)\bar{r}^2]$ (dimensionless, real)
KMLTLR	K	$K = B \exp(-\theta_a)/D$ (dimensionless, real)

NAMELIST/ØUTPT 3

FØRTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
TWBAR	$\bar{T}_w$	mean temperature of the gas-liquid interface at the mean position of this interface, i.e., at $\bar{y}$=o (°K, real)
YSSTBR	$\bar{y}_s^* = \frac{\bar{x}^*\bar{r}}{\kappa_s}$	dimensionless flame mean stand-off distance (dimensionless, real)
XSTBAR	$\bar{x}^*$	flame mean stand-off distance (cm, real)
AMM	a_m	equivalent mean particle diameter (cm, real)
TAUMLT(M-1)	$\bar{\tau}_{m-1}$	dimensionless_mean temperature within the melt layer Note: T_o=1.0, and $\bar{T}_{100}=T_{melt}$, the dimensionless melting temperature of the oxidizer. (dimensionless, real)
XMLT(M-1)	x_{m-1}	distance from the mean position of the gas-liquid interface to the location at which $\bar{T}=\bar{T}_{m-1}$ (cm, real)
M	m	index of the increments in $\bar{T}$ and x that are calculated and tabulated through the melt layer 2≤M≤101 (dimensionless, integer)
N	n	index of laminae from the gas-liquid interface (n=0), through the liquid-solid interface (n=1), through the first oxidizer-to-binder interface (n=2) through the first binder-to-oxidizer interface (n=3), and so on to as far into the CSP as n=101, if necessary (dimensionless, integer)
RBR	$\bar{r}$	temporal mean rate of regression of the composite solid propellant (cm sec^{-1}, real)
ITERA	-	specifies the mode of operation of the computer program: If ITERA=0, the program determines the combustion response parameters for a CSP whose temporal mean regression rate is specified; if ITERA=1, the program determines the temporal mean regression rate for a propellant whose basic physical and chemical characteristics have been specified (dimensionless, integer)
N5	-	when ITERA=1, N5 is the number of the iteration in the iterative attempt to determine RBR (1≤N5≤N5MAX) (dimensionless, integer)
N5MAX	-	if ITERA=1 and if a satisfactory value for RBR has not been achieved after N5MAX-1 one-percent increments or decrements of RBR, the solution will be terminated and an appropriate comment will be printed in the computer output (dimensionless, integer) Note: This mode of program temination indicates an injudicious initial choice for RBR.

NAMELIST/ØUTPT 3 (continued)

FØRTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
LFLAG	-	LFLAG is set to zero each time SUBRØUTINE SLDFAZ is entered in a normal manner. If ITERA=0, LFLAG is then immediately reset to 1 and control of the program is returned to SUBRØUTINE CØNTRL, the main program, as soon as explicit values of certain required variables are determined in SUBRØUTINES SLDFAZ, GASFAZ, and AMULTI. If ITERA=1, LFLAG is only reset to 1 and program control is only returned to the main program when a suitable value has been found for RBR or when N5=N5MAX (dimensionless, integer)
GWBRS	$\bar{g}_{w_s}$	$\bar{g}_{w_s} = (\frac{d\bar{\tau}}{dy_s})_{y_s=0}$ (dimensionless, real)
DGWBRS	$(\frac{d\bar{g}}{dy_s})_{y_s=0}$	$(\frac{d\bar{g}}{dy_s})_{y_s=0} = (\frac{d^2\bar{\tau}}{dy_s^2})_{y_s=0}$ (dimensionless, real)
QLDMLS	H_l	sometimes set equal to H_s; at other times, calculated as non-dimensionalized sum of H_m and the actual heat absorbed as a result of oxidizer degradation as it is transported through the melt layer. Note: The two methods of determining H_l ordinarily lead to virtually identical results. (dimensionless, real)

NAMELIST/ØUTPT 4

FØRTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
TAULR(N)	$\bar{\tau}_n$	dimensionless mean temperature at distance SLR(N) from the mean gas-liquid interface (dimensionless, real)
DTAULR(N)	$[\frac{d\bar{\tau}}{dy\{\substack{a\\b}\}}]_{n-}$	dimensionless mean temperature gradient at distance, XLR(N), from the mean gas-liquid interface and on the side of the N,N+1 interface nearest the mean gas-liquid interface. Note: For n even, $y=y_a$; for n odd, $y=y_b$ (dimensionless, real)
XLR (N)	x_n	distance from the mean position of the gas-liquid interface to the N,N+1 interface (cm, real)
DSCRIM(N)	$\bar{\tau}_n+(\frac{d\bar{\tau}}{dy})_{n-}$	because, deep in the CSP, the exponentially decaying nature of the solution requires that DSCRIM(N) approach zero, DSCRIM(N) furnishes an indication as to whether the solution in this region has proceeded to a depth sufficient to provide a satisfactory steady-state solution (dimensionless, real)
N	n	index of laminae from the gas-liquid interface (n=0), through the liquid-solid interface (n=1), through the first oxidizer-to-binder interface (n=2) through the first binder-to-oxidizer interface (n=3), and so on to as far into the CSP as n=101, if necessary (dimensionless, integer)
XTD	-	SQRT (KAPS/ØMEGA)*YTD (cm, real)
CAPØA	Ω_a	$\Omega_a = \frac{\omega\kappa_a}{\bar{r}^2}$ (dimensionless, real)
CAPØB	Ω_b	$\Omega_b = \frac{\omega\kappa_b}{\bar{r}^2}$ (dimensionless, real)
CAPØS	Ω_s	$\Omega_s = \frac{\omega\kappa_s}{\bar{r}^2}$ (dimensionless, real)

NAMELIST/ØUTPT 5

FØRTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
N2	$n2$	index of the laminae from the deep-solid side of the first oxidizer laminae for which XLR (N2=N2MXP1) > XTD, through the adjacent overlying binder lamina for which XLR (N2=N2MXP1-1) > XTD, through successive laminae of oxidizer and binder and, finally, of melt layer to N2=0. Note: XLR(N2=N) ≡ XLR(N) The index N2 is used in the step-by-step calculation of the perturbed solution, whereas N is used in the step-by-step calculation of the steady-state solution. (dimensionless, integer)
N2MXP1	$n2_{max}+1$	maximum value of N2 determined as indicated in the definition of N2 (dimensionless, integer)
TAPIN	$\tau'_{n2_{max}+1}$	normalized temperature perturbation applied in the solid phase at a distance from the gas-condensed phase interface that is several times the characteristic transient thermal depth within the solid. (dimensionless, complex)
DTAPIN	$g'_{n2_{max}+1}$	$g'_{n2_{max}+1} = (\frac{d\tau'}{dy_a})_{n2_{max}+1}$ (dimensionless, complex)
XLR(N2)	x_{n2}	distance from the mean position of the gas-liquid interface to the N2, N2+1 interface (cm, real)
TAPLR(N2)	τ'_{n2}	dimensionless perturbed temperature at the N2, N2+1 interface (dimensionless, complex)
DTAPLR(N2)	g'_{n2}	first derivative of dimensionless perturbed temperature with respect to y_a when N2 is even, or with respect to y_b when N2 is odd, at the N2, N2+1 interface (dimensionless, complex)
TAUOPS	$\tau'_{y=o}$	dimensionless perturbed temperature at the mean position of the gas-liquid interface, i.e., at y=o (dimensionless, complex)
GOPPS	$g'_{y=o}$	first derivative of dimensionless perturbed temperature with respect to y_a at y=0+ (dimensionless, complex)
DGOPPS	$g'_{y=o}$	second derivative of dimensionless perturbed temperature with respect to y_a at y=0+ (dimensionless, complex)
RF	R	$R = (\frac{r'/\bar{r}}{p'/\bar{p}})$, the response function (dimensionless, complex)

NAMELIST/ØUTPT 5 (continued)

FORTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
NRP	$\frac{p'}{\bar{p}}$	normalized perturbed pressure in the gas phase (quasi-static gas-phase behavior assumed) (dimensionless, complex)
YSWP	y'_{s_w}	dimensionless displacement of the instantaneous gas-liquid interface with respect to the mean gas-liquid interface, i.e., with respect to y=0; displacement non-dimensionalized on the basis of bulk CSP properties (dimensionless, complex)
XWP	x'_w	$x'_w = \frac{\kappa_s y'_{s_w}}{\bar{r}}$ (cm, complex)
NXSTP	$\frac{x^{*'}}{\bar{x}^*}$	perturbed displacement of the flame front with respect to the mean position of the flame front; normalized with respect to the mean position of the flame front (dimensionless, complex)
XSTP	$x^{*'}$	$x^{*'} = \bar{x}^* \cdot \frac{x^{*'}}{\bar{x}^*}$ (cm, complex)
NYSSTP	$\frac{y_s^{*'}}{\bar{y}_s^*}$	$\frac{y_s^{*'}}{\bar{y}_s^*} = \frac{x^{*'}}{\bar{x}^*}$ (dimensionless, complex)
YSSTP	$y_s^{*'}$	$y_s^{*'} = \bar{y}_s^* \cdot \frac{x^{*'}}{\bar{x}^*}$ (dimensionless, complex)
RFNMLT	R_{NML}	response function under the assumption that no melt layer (NML) exists in the oxidizer lamina, that is contiguous with the gas phase (dimensionless, complex)
NRPNM	$(\frac{p'}{\bar{p}})_{NML}$	normalized perturbed pressure in the gas phase under the assumption that no melt layer exists (dimensionless, complex)
YSWPNM	$y'_{s_{w_{NML}}}$	y'_{s_w} under the assumption that no melt layer exists (dimensionless, complex)
XWPNM	$x'_{w_{NML}}$	x'_w in the absence of a melt layer (cm, complex)

NAMELIST/ØUTPT 5 (continued)

FØRTRAN NAME	ALGEBRAIC SYMBOL	DEFINITION
NXSTPN	$\left(\frac{x^{*\prime}}{\bar{x}^{*}}\right)_{NML}$	$\frac{x^{*\prime}}{\bar{x}^{*}}$ in the absence of a melt layer (dimensionless, complex)
XSTPNM	$x^{*\prime}_{NML}$	$x^{*\prime}$ in the absence of a melt layer (cm, complex)
YSSTPN	$y^{*\prime}_{s_{NML}}$	$y^{*\prime}_{s}$ in the absence of a melt layer (dimensionless, complex)

A-4. CARD IMAGE LISTING OF PROGRAM

```
C
C
CSTEAD PROC
C
C
C
C      THIS RUN HAS BEEN MADE WITH PROGRAM DECK NO. 9A
C
C
       IMPLICIT REAL (K)
       COMMON/STEADY/ AMM,B,C,CA,CB,CG,CHI,CM,CS,D,DLTYAB,DLTYBB,DLTYTP,
      1DELTAT,DELTAU,DNM,DNOM,DTAULR,DSCRIM,E,ETA,FDMLS,JMAX,KA,KAPA,
      2KAPB,KAPS,KB,KFLMHT,KMLTLR,KS,PBAR,PRXFAC,QB,QBDMLS,QLM,QMDMLS,
      3QS,QSDMLS,R,RBR,ROA,ROB,ROS,SMLA,SMLATP,SMLB,TAULR,BUFFR1,TAUMLT,
      4TFLM,THETAA,TM,TOL,TWBAR,TZRO,VFA,WA,WAC,WB,XLR,BUFFR2,XMLT,
      5XSTBAR,YSSTBR,BUFFR3,YABRML,Z,ZI,ZPR,RBROLD,DLTRBR,ITERA,N5,N5MAX,
      6LFLAG,CAPOA,CAPOB,CAPOS,GWBRS,DGWBRS,QLDMLS,IHOMO
C
C      BUFFR1, BUFFR2, AND BUFFR3 ARE REQUIRED TO BUFFER THE STORAGE
C        LOCATIONS JUST PRECEDING TAUMLT(1), XMLT(1), AND YABRML(1),
C        RESPECTIVELY. THIS PROCEDURE IS MADE NECESSARY BY MAKING TAUT(2)
C        EQUIVALENT TAUMLT(1), XMLTT(2), TO XMLT(1), AND YABRT(2), TO
C        YABRML(1).
C
       DIMENSION DSCRIM(110),DTAULR(110),SMLA(4),TAULR(110),TAUMLT(110),
      1TAUT(110),WAC(4),XLR(110),XMLT(110),XMLTT(110),YABRML(110),
      2YABRT(110)
       EQUIVALENCE (TAUT(2),TAUMLT(1)),(XMLTT(2),XMLT(1)),
      1(YABRT(2),YABRML(1))
       EQUIVALENCE  (GWBRS,GOBRS),(DGWBRS,DGOBRS)
 END
```

```
C
C
C UNSTD PROC
C
C
      COMMON/UNSTDY/NPP,TAPIN,XTD,YTD,OMEGA,NEQ,KD,MXSTEP,KQ,EP,
     1DTAPIN,TAPLR,DTAPLR,N2MXP1,RF,NRP,YSWP,XWP,NXSTP,XSTP,NYSSTP,
     2YSSTP,D2TAPR,D2TAPI,TAUOPS,G0PPS,DG0PPS,
     3RFNMLT,NRPNM,YSWPNM,XWPNM,NXSTPN,XSTPNM,YSSTPN
C
      COMPLEX NPP,TAPIN,DTAPIN,TAPLR,DTAPLR,RF,NRP,YSWP,XWP,NXSTP,XSTP,
     1NYSSTP,YSSTP,TAUOPS,G0PPS,DG0PPS,
     2RFNMLT,NRPNM,YSWPNM,XWPNM,NXSTPN,XSTPNM,YSSTPN
C
      DIMENSION TAPLR(110),DTAPLR(110)
 END
```

```
C
C
C     CONTRL PROVIDES THE SEQUENCE OF CALLS TO
C       SUBROUTINES THAT IS REQUIRED TO OBTAIN THE DESIRED SOLUTION
C       OF THE COMBUSTION RESPONSE PROBLEM.
C
C
      INCLUDE CSTEAD, LIST
      INCLUDE CUNSTD, LIST
C
C
   10 CONTINUE
      CALL OVRFLW(3)
      CALL DATAIN
      IF(ITERA .EQ. 0 .AND. OMEGA .EQ. 0.) GO TO 70
      CALL SLDFAZ
      CALL MELTLR
      CALL LAYERS
      IF(ITERA .NE. 0) GO TO 60
      CALL LAYRSP
      CALL MLTLRP
      CALL GLIFP
   60 CALL DTAOUT
      IF(ITERA .EQ. 0 .AND. OMEGA .NE. 0.) GO TO 10
   70 CONTINUE
      STOP
      END
```

```
C
C
      SUBROUTINE DATAIN
C
C     SUBROUTINE DATAIN LOADS THE INPUT DATA AND DERIVES CERTAIN WIDELY
C       USED PARAMETERS FROM THESE DATA.
C
      INCLUDE CSTEAD, LIST
      INCLUDE CUNSTD, LIST
      INTEGER KD,KQ
      DIMENSION TNPP(2),TTAPIN(2)
      EQUIVALENCE (TNPP,NPP),(TTAPIN,NRP)
C
C
C     CAUTION***NAMELIST/INPUT/ REQUIRES THE ENTRY OF NPP AND TAPIN IN
C       TERMS OF THE NUMBER PAIR, MODULUS (AS A DECIMAL FRACTION) AND
C       ARGUMENT (AS A PHASE ANGLE IN DEGREES). NPP AND TAPIN ARE THEN
C       REDEFINED BY THEIR CARTESIAN REAL AND IMAGINARY COMPONENTS IN
C       SUBROUTINE DATAIN.
C
C
      NAMELIST/INPUT/CA,CB,CG,KA,KB,ROA,ROB,WAC,SMLA,E,R,PRXFAC,QS,QB,
     1QLM,KFLMHT,TFLM,TM,TZRO,PBAR,RBR,TOL,ITERA,N5MAX,NPP,TAPIN,YTD,
     2OMEGA,NEQ,KD,MXSTEP,EP,IHOMO
      READ(5,INPUT)
      IF(OMEGA .EQ. 0.) RETURN
      DO 10 I=1,101
      TAPLR(I)=0.0
      DTAPLR(I)=0.0
   10 CONTINUE
      IF(IDATA .EQ. 1) GO TO 106
      IDATA=1
      WRITE(6,INPUT)
      WA=0.
      DNM=0.
      JMAX=0
      IF(WAC(1) .GT. 0.) JMAX=1
      IF(WAC(2) .GT. 0.) JMAX=2
      IF(WAC(3) .GT. 0.) JMAX=3
      IF(WAC(4) .GT. 0.) JMAX=4
      IF(JMAX .EQ. 0) WRITE(6,150)
  150 FORMAT(//T6,-NO AP PARTICLES OF ANY FINITE DIAMETER HAVE BEEN INCL
     1UDED IN THE INPUT DATA-//)
      IF(JMAX .EQ. 0) STOP
      DO 105 J=1,JMAX
      WA=WA+WAC(J)
      DNM=DNM+WAC(J)/SMLA(J)
  105 CONTINUE
  106 IF(IHOMO .EQ. 0) GO TO 107
C
C     WHEN IHOMO=0, CERTAIN PERTINENT PHYSICAL AND THERMAL PROPERTIES OF
C       THE PROPELLANT BINDER, OXIDIZER, AND GASEOUS PRODUCTS SPECIFIED
C       IN NAMELIST/INPUT/ ARE GENERALLY DISTINCT; HOWEVER, WHEN
C       IHOMO=1, THESE PROPERTIES OF THE OXIDIZER AND GASEOUS PRODUCTS
C       ARE SET EQUAL TO CORRESPONDING PROPERTIES OF THE OXIDIZER.
C       THUS, SPECIFYING IHOMO=1 IMPLIES A HOMOGENEOUS PROPELLANT.
C
      CB=CA
      CG=CA
```

```
      KB=KA
      ROB=ROA
  107 CONTINUE
      WB=1.-WA
      VFA=WA/ROA/(WA/ROA+WB/ROB)
      SMLATP=SQRT(2./3.)*SMLA(JMAX)
      SMLB=WA*VFA/DNM
      KS=KA
      ROS=ROA*ROB/(ROA*WB+ROB*WA)
      CS=CA*WA+CB*WB
      KAPA=KA/ROA/CA
      KAPB=KB/ROB/CB
      KAPS=KS/ROS/CS
      DLTYTP=SMLATP*RBR/KAPA
      DLTYAB=RBR*SMLA(1)/KAPA
      DLTYBB=RBR*SMLB/KAPB
      Z=ROS/ROA*CS/CA
      ZPR=ROS/ROB*CS/CB
      TNPP(1)=REAL(NPP)*COS(AIMAG(NPP/57.29578))
      TNPP(2)=REAL(NPP)*SIN(AIMAG(NPP/57.29578))
      TTAPIN(1)=REAL(TAPIN)*COS(AIMAG(TAPIN)/57.29578)
      TTAPIN(2)=REAL(TAPIN)*SIN(AIMAG(TAPIN)/57.29578)
  110 CONTINUE
      CAPOA=OMEGA*KAPA/RBR**2
      CAPOB=OMEGA*KAPB/RBR**2
      CAPOS=OMEGA*KAPS/RBR**2
      XTD=YTD*SQRT(KAPS/OMEGA)
      RETURN
      END
```

```
C
C
      FUNCTION TAU(TW,TZRO,T)
C
C     FUNCTION TAU DETERMINES A DIMENSIONLESS TEMPERATURE THAT IS WIDELY
C       USED THROUGHOUT THIS PROGAM.
C
      TAU=(T-TZRO)/(TW-TZRO)
      RETURN
      END
```

```
C
C
      SUBROUTINE SLDFAZ
C
C     FOR A GIVEN VALUE OF MEAN BURNING RATE, RBR, SUBROUTINE SLDFAZ
C       CALCULATES MEAN WALL TEMPERATURE, TWBAR.
C
      INCLUDE CSTEAD, LIST
      INCLUDE CUNSTD, LIST
C
      NAMELIST/TPOUT1/TWBAR,QSDMLS,QMDMLS,QBDMLS,THETAA,CHI,D,B,KMLTLR,
     1QLDMLS,FDMLS,GWBRS,DGWBRS
C
C
      LFLAG=0
      N5=1
      DAM=0.
  202 IF(LFLAG .EQ. 1) RETURN
      IF(ITERA .EQ. 0 .OR. N5 .EQ. N5MAX) LFLAG=1
      RBROLD=RBR
      TWBAR=TM
      TRLRBR=0.
      DO 205 I=1,100
      OLDTWB=TWBAR
      OLDRBR=TRLRBR
      DELTAT=.01*(TFLM-TM)
      TWBAR=OLDTWB+DELTAT
      QSDMLS=WA*QS/CA/(TWBAR-TZRO)
      QMDMLS=WA*QLM/CA/(TWBAR-TZRO)
      QLDMLS=QSDMLS
      THETAA=E/R/TWBAR
      CHI=(TWBAR-TZRO)/TWBAR
      TRLRBR=SQRT(2.*KAPA*PRXFAC*QSDMLS*EXP(-THETAA)*
     1(1.-EXP(-THETAA*CHI*(1.-TAU(TWBAR,TZRO,TM))))
     2/(THETAA*CHI*((Z*(1.+(QLDMLS+QMDMLS)*CA/CS))**2
     3-((TAU(TWBAR,TZRO,TM)*(1.-Z)-Z*QMDMLS*CA/CS)-1.)**2)))
      IF(TRLRBR .GT. RBR) GO TO 210
  205 CONTINUE
      WRITE(6,208)
  208 FORMAT(///T6,-NUMBER OF ITERATIONS FOR RBR HAS REACHED 100 IN SUBR
     1OUTINE SLDFAZ-///)
      STOP
  210 TWBAR=OLDTWB+(TWBAR-OLDTWB)*(RBR-OLDRBR)/(TRLRBR-OLDRBR)
      QSDMLS=WA*QS/CA/(TWBAR-TZRO)
      QMDMLS=WA*QLM/CA/(TWBAR-TZRO)
      QLDMLS=QSDMLS
      QBDMLS=WB*QB/CB/(TWBAR-TZRO)
      THETAA=E/R/TWBAR
      CHI=(TWBAR-TZRO)/TWBAR
      D=THETAA*CHI
      B=KAPA*PRXFAC*QSDMLS/RBR/RBR
      KMLTLR=B*EXP(-THETAA)/D
      FDMLS=(QLDMLS+QMDMLS)*CA/CS+QBDMLS*CB/CS
      GWBRS=-Z*(1.+(QLDMLS+QMDMLS)*CA/CS)
      DGWBRS=-GWBRS+B*EXP(-THETAA)
      WRITE(6,TPOUT1)
      CALL GASFAZ(*202)
      RETURN
      END
```

```
C
C
      SUBROUTINE GASFAZ($)
C
C     GIVEN RBR AND TWBAR, SUBROUTINE GASFAZ CALCULATES FLAME STAND-OFF
C       DISTANCES, YSSTBR AND XSTBAR.
C
      INCLUDE CSTEAD, LIST
      INCLUDE CUNST), LIST
  300 YSSTBR=ALOG((TAU(TWBAR,TZRO,TFLM)+FDMLS)/(1.+FDMLS))
      XSTBAR=YSSTBR*KAPS/RBR
      CALL AMULTI ($310)
  310 RETURN1
      END
```

```
C
C
      SUBROUTINE AMULTI(*)
C
C     SUBROUTINE AMULTI CALCULATES A VIRTUAL UNIMODAL AP PARTICLE
C       DIAMETER CORRESPONDING TO RBR AND YSSTBR.
C
      INCLUDE CSTEAD, LIST
  500 AMM=KAPS*PBAR*YSSTBR/(KELMHT*RBR*RBR)
      IF(LFLAG .EQ. 1) RETURN
      DAMOLD=DAM
      DAM=AMM-SMLA(JMAX)
      IF(ABS(DAM-DAMOLD) .GT. ABS(ABS(DAM)-ABS(DAMOLD))) GO TO 550
      DLTRBR=.01*RBR
      IF(DAM)510,520,530
  510 RBR=RBR-DLTRBR
      GO TO 540
  520 GO TO 540
  530 RBR=RBR+DLTRBR
      GO TO 540
  540 N5=N5+1
      RETURN1
  550 RBR=RBROLD-(RBR-RBROLD)*DAMOLD/(DAM-DAMOLD)
      LFLAG=1
      RETURN
      END
```

```
C
C
      SUBROUTINE MELTLR
C
C     SUBROUTINE MELTLR DETERMINES TEMPORAL MEAN TEMPERATURE AND THE
C       DERIVATIVE OF THIS TEMPERATURE WITH RESPECT TO DIMENSIONLESS
C       DISTANCE AS FUNCTIONS OF DIMENSIONAL DISTANCE INTO THE LIQUID
C       PROPELLANT (MEASURED FROM THE TEMPORAL MEAN POSITION OF THE
C       MELTING SURFACE OF THE PROPELLANT).
C
      INCLUDE CSTEAD, LIST
      INCLUDE CUNSTD, LIST
      COMMON/M/ M
      COMMON/N/ N
      COMMON/MLTINT/ P,NDEG
C
C
      DELTAU=.01*(1.-TAU(TWBAR,TZRO,TM))
      TAUT(1)=1.
      YABRT(1)=0.
      DO 505 I=2,101
      TAUT(I)=TAUT(I-1)-DELTAU
      ETA=D*(1.-TAUT(I))
      CM=0.5+KMLTLR-0.5*(Z*(1.+(QLDMLS+QMDMLS)*CA/CS))**2
      C=(1.-2.*CM)/KMLTLR
      ZI=C*EXP(ETA)
      DNOM=1.+C-SQRT(C**2+2.*C)
      YABRT(I)=ALOG((1.+ZI-SQRT(ZI**2+2.*ZI))/DNOM)/(D*SQRT(1.-2.*CM))
      XMLTT(I)=KAPA*YABRT(I)/RBR
      M=I
  505 CONTINUE
      DIMENSION P(10),W(200)
      CALL PFIT(M,YABRT,TAUT,-1.,6.,.TRUE.,.TRUE.,.TRUE.,P,NDEG,
     1SIGFAC,W)
      NP3=NDEG+3
      WRITE(6,510) NDEG,SIGFAC,(P(I),I=1,NP3)
  510 FORMAT(//T5,-NDEG=-,I2,T17,-SIGFAC=-,F6.4/T5,-P(1),...,P(NDEG+3)=-
     1,T24,3F15.8/(T24,3F15.8))
      DO 530 I=1,M
      TAUFIT=SCPVAL(P,NDEG,YABRT(I))
      TAUERR=TAUT(I)-TAUFIT
  530 CONTINUE
      TAULR(1)=TAUMLT(100)
      DTAULR(1)=-Z*(TAUMLT(100)+QMDMLS*CA/CS)
      XLR(1)=YABRML(100)*KAPA/RBR
      DSCRIM(1)=TAULR(1)+DTAULR(1)
      N=2
      RETURN
      END
```

```
C
C
      SUBROUTINE LAYERS
C
C     SUBROUTINE LAYERS DETERMINES TEMPORAL MEAN TEMPERATURE AND THE
C       DERIVATIVE OF THIS TEMPERATURE WITH RESPECT TO DIMENSIONLESS
C       DISTANCE AS FUNCTIONS OF DIMENSIONAL DISTANCE INTO THE SOLID
C       PROPELLANT (MEASURED FROM THE TEMPORAL MEAN POSITION OF THE
C       MELTING SURFACE OF THE PROPELLANT).
C
      INCLUDE CSTEAD, LIST
      INCLUDE CUNSTD, LIST
      COMMON/N/ N
  601 IF(N .EQ. 2) GO TO 610
  602 IF(MOD(N,2) .EQ. 0) GO TO 615
  605 DELTAY=DLTYBB
      DELTAX=SMLB
      C1=(1.-ZPR)*TAULR(N-1)
      GO TO 630
  610 DELTAY=DLTYTP
      DELTAX=SMLATP
      C1=(1.-Z)*TAULR(1)
      GO TO 630
  615 DELTAY=DLTYAB
      DELTAX=SMLA(1)
      C1=(1.-Z)*TAULR(N-1)
  630 TAULR(N)=C1-(C1-TAULR(N-1))*EXP(-(DELTAY))
      DTAULR(N)=C1-TAULR(N)
      XLR(N)=XLR(N-1)+DELTAX
      DSCRIM(N)=TAULR(N)+DTAULR(N)
      IF(MOD(N,2) .EQ. 0) GO TO 6540
      ITOL=0
      IN=0
      IXTD=0
 6400 IF(ABS(ABS(DSCRIM(N))-ABS(DSCRIM(N-2))) .EQ. ABS(DSCRIM(N)-
     1DSCRIM(N-2)) .AND. ABS(DSCRIM(N)) .LT. TOL .AND. ABS(TAULR(N))
     2.LT. TOL) GO TO 6410
      GO TO 6420
 6410 ITOL=1
      WRITE(6,6415)
 6415 FORMAT(//6X,-TOLERANCE TESTS FOR A SATISFACTORY STEADY-STATE SOLUT
     1ION HAVE BEEN MET.-//)
 6420 IF(N .LE. 100) GO TO 6430
      WRITE(6,6425)
 6425 FORMAT(//6X,-MORE THAN FIFTY PAIR OF OXIDIZER AND BINDER LAYERS HA
     1VE BEEN TRAVERSED.-//)
      GO TO 6440
 6430 IN=1
 6440 IF(XLR(N-1) .GT. XTD) GO TO 6450
      WRITE(6,6445)
 6445 FORMAT(//6X,-NUMBER OF LAYERS TRAVERSED IS INSUFFICIENT TO INSURE
     1SATISFACTORY ACCURACY OF THE PERTURBED SOLUTION.-//)
      GO TO 6460
 6450 IXTD=1
 6460 CONTINUE
 6500 IF(ITOL .EQ. 1) GO TO 6510
      IF(IN .EQ. 1) GO TO 6530
      IF(IXTD .EQ. 1) GO TO 6550
 6510 IF(IN .EQ. 1) GO TO 6520
```

```
      GO TO 6550
 6520 IF(IXTD .EQ. 1) GO TO 6560
 6530 GO TO 6540
 6540 N=N+1
      GO TO 602
 6550 IF(ABS(ABS(DSCRIM(N))-ABS(DSCRIM(N-2))) .EQ. ABS(DSCRIM(N)-
     1DSCRIM(N-2))) WRITE(6,660)
  660 FORMAT(/6X,-DISCRIMINANTS N AND N-2 HAVE THE SAME ALGEBAIC SIGN.-)
      IF(ABS(DSCRIM(N)) .LT. TOL) WRITE(6,665)
  665 FORMAT(/6X,-ABS(DSCRIM(N)) MEETS REQUIREMENTS FOR A SATISFACTORY S
     1OLUTION.-)
      IF(ABS(TAULR(N)) .LT. TOL) WRITE(6,670)
  670 FORMAT(/6X,-ABS(TAULR(N)) MEETS REQUIREMENTS FOR A SATISFACTORY SO
     1LUTION.-)
      WRITE(6,6555)
 6555 FORMAT(//6X,-THE ACCURACY OF THIS STEADY-STATE SOLUTION MAY BE SAT
     1ISFACTORY$-/6X,-HOWEVER, IT HAS NOT MET ALL CONVERGENCE CRITERIA.-
     2/6X,-REFER TO THE PRECEDING DIAGNOSTIC REMARKS TO DETERMINE WHICH
     3CRITERIA-/6X,-HAVE BEEN MET AND WHICH HAVE NOT.-//)
      RETURN
 6560 WRITE(6,6565)
 6565 FORMAT(//6X,-A SATISFACTORY STEADY-STATE SOLUTION HAS BEEN OBTAINE
     1D.-//)
      RETURN
      END
```

```
C
C
      SUBROUTINE LAYRSP
C
C     IN SUBROUTINE LAYRSP, AN INITIAL DEPTH WITHIN THE SOLID, ADEQUATE
C       TO EXCEED SOME CHOSEN MULTIPLE OF THERMAL DEPTH AND TERMINATING
C       ON THAT SURFACE OF A BINDER LAYER WHICH IS FARTHEST FROM THE
C       BURNING SURFACE, IS DETERMINED. INTIAL VALUES OF DIMENSIONLESS
C       TEMPERATE AND ITS DERIVATIVE WITH RESPECT TO DIMENSIONLESS
C       DISTANCE ARE THEN SPECIFIED ON THE FAR SIDE OF THE NEXT UNDERLY-
C       ING OXIDIZER LAYER.  DIMENSIONLESS TEMPERATURE IS ARBITRARILY
C       ASSUMED. ITS DERIVATE IS APPROXIMATED FROM A REQUIREMENT THAT IS
C       STRICTLY VALID ONLY FOR A HOMOGENEOUS SEMI-INFINITE SLAB.
C
C
C
      INCLUDE CSTEAD, LIST
      INCLUDE CUNSTD, LIST
C
      COMMON/N2/N2
C
      COMPLEX LAM1, LAM2, C1, C2, TAP, GPM
C
C
      N2=3
 1601 IF(XLR(N2) .GT. XTD) GO TO 1605
      N2=N2+2
      GO TO 1601
 1605 N2MXP1=N2+1
      CAPO=CAPOA
      A1=SQRT((SQRT(1.+16.*CAPO**2)+1.)/2)
      B1=SQRT((SQRT(1.+16.*CAPO**2)-1.)/2)
      LAM1=CMPLX(-0.5*(1.+A1),-0.5*B1)
      DTAPIN=LAM1*TAPIN
      N2=N2MXP1
      DELTAY=-DLTYAB
      TAPLR(N2)=TAPIN
      TAP=TAPIN
      DTAPLR(N2)=DTAPIN
      GPM=DTAPIN
      GO TO 1655
 1607 IF(N2 .EQ. 2) GO TO 1630
 1609 IF(MOD(N2,2) .EQ. 0) GO TO 1620
 1610 DELTAY=-DLTYBB
      CAPO=CAPOB
      TAP=TAPLR(N2)
      GPM=ZPR/Z*DTAPLR(N2)
      GO TO 1655
 1620 DELTAY=-DLTYAB
      CAPO=CAPOA
      TAP=TAPLR(N2)
      GPM=Z/ZPR*DTAPLR(N2)
      GO TO 1655
 1630 DELTAY=-DLTYTP
      CAPO=CAPOA
      TAP=TAPLR(2)
      GPM=Z/ZPR*DTAPLR(2)
 1655 A1=SQRT((SQRT(1.+16.*CAPO**2)+1.)/2)
      B1=SQRT((SQRT(1.+16.*CAPO**2)-1.)/2)
```

```
      LAM1=CMPLX(-0.5*(1.+A1),-0.5*B1)
      LAM2=CMPLX(-0.5*(1.-A1),+0.5*B1)
      C1=(LAM2*TAP-GPM)/(LAM2-LAM1)
      C2=-(LAM1*TAP-GPM)/(LAM2-LAM1)
      TAPLR(N2-1)=C1*EXP(LAM1*DELTAY)+C2*EXP(LAM2*DELTAY)
      DTAPLR(N2-1)=C1*LAM1*EXP(LAM1*DELTAY)+C2*LAM2*EXP(LAM2*DELTAY)
 1665 N2=N2-1
      IF(N2 .EQ. 1) RETURN
      GO TO 1607
      END
```

```
C
C
      SUBROUTINE MLTLRP
C
C     GIVEN TAULR(1) AND DTAULR(1), SUBROUTINE MLTLRP INTEGRATES
C       STEP-BY-STEP FROM THE LIQUID-SOLID INTERFACE IN THE TOPMOST
C       OXIDIZER LAYER TO THE GAS-LIQUID INTERFACE THAT IS THE BURNING
C       SURFACE.  TAUOPS AND GOPPS ARE THUS DETERMINED FOR THE LIQUID
C       SIDE OF THE BURNING SURFACE.
C
      INCLUDE CSTEAD, LIST
      INCLUDE CUNSTD, LIST
C
      COMMON/M/ M
      COMMON/MLTINT/ P,NDEG
      COMMON/N2/N2
C
      DIMENSION P(10)
C
C
 1500 INTEGER NEQ,KD,IFLAG,MXSTEP,KSTEP,KEMAX,KQ
      REAL EP,HMINA,HMAXA,EMAX
      REAL D2TAP,DT(10,1)
      REAL YABRV,DEPVAR(2)
      H=-.01*YABRT(M)
      HMINA=.1*H
      HMAXA=10.*H
      YABRVF=0.
      DELT=10.*H
 1501 L=1
      YABRV=YABRT(M)
      DEPVAR(1)=REAL(TAPLR(1))
      DEPVAR(2)=REAL(DTAPLR(1))
      WRITE(6,1504)
 1504 FORMAT(//T11,-YABRV-,T26,-TAPR-,T41,-DTAPR-,T55,-D2TAPR-/)
      GO TO 1507
 1502 L=2
      YABRV=YABRT(M)
      DEPVAR(1)=AIMAG(TAPLR(1))
      DEPVAR(2)=AIMAG(DTAPLR(1))
      WRITE(6,1506)
 1506 FORMAT(//T11,-YABRV-,T26,-TAPI-,T41,-DTAPI-,T55,-D2TAPI-/)
 1507 CALL SVDQ(NEQ,YABRV,DEPVAR,D2TAP,KD,EP,IFLAG,H,HMINA,HMAXA,DELT,
     1YABRVF,MXSTEP,KSTEP,KEMAX,EMAX,KQ,YN,DT)
      GO TO 20
   10 CALL SVDQ1
   20 GO TO (30,30,40,40,50,60,70,70), IFLAG
   30 TAUFIT=SCPVAL(P,NDEG,YABRV)
      T3=KMLTLR*D*D*EXP(-D*(1.-TAUFIT))*DEPVAR(1)
      D2TAP=-DEPVAR(2)+T3
      GO TO 10
   40 WRITE(6,1540) IFLAG, H
 1540 FORMAT(/T6,-IFLAG=-,I2,T21,-H=-,E15.6)
   45 WRITE(6,1545) YABRV,DEPVAR(1),DEPVAR(2),D2TAP
 1545 FORMAT(/T6,4E15.6)
      IF(IFLAG .EQ. 3) GO TO 10
      IF(L .EQ. 1) GO TO 1552
      IF(L .EQ. 2)  GO TO 1554
   50 WRITE(6,1540) IFLAG, H
```

```
      GO TO 10
   60 EP=32.*EMAX*EP
      WRITE(6,1560) YABRV,EP
 1560 FORMAT(/T6,-OLD EP WAS TOO SMALL. NEW EP HAS BEEN ESTABLISHED AT Y
     1ABRV-/T8,-YABRV=-,E15.6,T36,-EP=-,E15.6)
      GO TO 10
   70 WRITE(6,1540) IFLAG, H
      GO TO 10
 1552 TAPR=DEPVAR(1)
      DTAPR=DEPVAR(2)
      D2TAPR=D2TAP
      GO TO 1502
 1554 TAPI=DEPVAR(1)
      DTAPI=DEPVAR(2)
      D2TAPI=D2TAP
      TAUOPS=CMPLX(TAPR,TAPI)
      GOPPS=CMPLX(DTAPR,DTAPI)
      DGOPPS=CMPLX(D2TAPR,D2TAPI)
      N2=0
      RETURN
      END
```

```
C
C
      SUBROUTINE GLIFP
C
C     SUBROUTINE GLIFP DETERMINES THE RESPONSE FUNCTION AND OTHER
C       DEPENDENT VARIABLES ASSOCIATED WITH THE PERTURBED SOLUTION.
C       THE EFFECT OF THE CONDENSED-PHASE ON THESE RESULTS IS DETERMINED
C       SOLELY BY THE RATIO, GOPPS/TAUOPS, WHICH APPEARS IN THE FORMULA
C       FOR RESPONSE FUNCTION,RF.
C
C
      INCLUDE CSTEAD,LIST
      INCLUDE CUNSTD,LIST
C
C
C
      COMPLEX K2
C
      NAMELIST/TPOUT3/K2,V3,V5,V6A,V6B,V7,RF0,K2NM,V5NM
C
      K2=GOPPS/TAUOPS
      V3=Z*(1.+(QMDMLS+QLDMLS)*CA/CS)
      V5=(1.+THETAA/2.)*CHI
     1+(D/2.)*(1.-2.*(1.-TAU(TWBAR,TZRO,TM))*CHI)
     2/(EXP(D*(1.-TAU(TWBAR,TZRO,TM)))-1.)
     3-1./(V3-(TAU(TWBAR,TZRO,TM)*(1.-Z)-Z*QMDMLS*CA/CS-1.))
      V6A=EXP(YSSTBR)-1.
      V6B=(1.+FDMLS)*YSSTBR*EXP(YSSTBR)
      V7=Z*QBDMLS*CB/CS
      RF0=V6B/(2.*V6B+V6A/V5+1./V5)
      RF=CAPOA*V6B/(CAPOA*(2.*V6B+V6A/V5-V3/Z)+(0.,1.)*KMLTLR*D/Z
     1-(CAPOA/V5-(0.,1.)*V3)*K2/Z)
      NRP=RF*NPP
      YSWP=(0.,-1.)*NRP/CAPOS
      XWP=KAPS*YSWP/RBR
      NXSTP=(1.-CMPLX(0.,-1./(CAPOS*YSSTBR)))*NRP-NPP
      XSTP=NXSTP*XSTBAR
      NYSSTP=NXSTP
      YSSTP=NYSSTP*YSSTBR
      IF(IHOMO .EQ. 0) GO TO 1650
      K2NM=DTAPIN/TAPIN
      CAPCNM=1.
      V3NM=1.
      V5NM=THETAA*CHI
      RFNMLT=CAPOA*V6B/(CAPOA*(2.*V6B+V6A/V5NM-V3NM/Z)+(0.,1.)*CAPCNM/Z
     1-(CAPOA/V5NM-(0.,1.)*V3NM)*K2NM/Z)
      NRPNM=RFNMLT*NPP
      YSWPNM=(0.,1.)*NRPNM/CAPOS
      XWPNM=KAPS*YSWPNM/RBR
      NXSTPN=(1.-CMPLX(0.,-1./(CAPOS*YSSTBR)))*NRPNM-NPP
      XSTPNM=NXSTPN*XSTBAR
      YSSTPN=NXSTPN*YSSTBR
 1650 CONTINUE
      WRITE(6,TPOUT3)
      RETURN
      END
```

```
C
C
      SUBROUTINE DTAOUT
C
C     SUBROUTINE DTAOUT PROVIDES A PRINTED RECORD OF THE RESULTS THAT
C       HAVE ACCUMULATED DURING THE COURSE OF THE RUN.
C
      INCLUDE CSTEAD, LIST
      INCLUDE CUNSTD, LIST
C
      INTEGER KD,KQ
C
      COMMON/M/ M
      COMMON/N/ N
      COMMON/N2/N2
C
C
      NAMELIST/INPUT/CA,CB,CG,KA,KB,ROA,ROB,WAC,SMLA,E,R,PRXFAC,QS,QB,
     1QLM,KFLMHT,TFLM,TM,TZRO,PBAR,RBR,TOL,ITERA,N5MAX,NPP,TAPIN,YTD,
     2OMEGA,NEQ,KD,MXSTEP,EP,IHOMO
      NAMELIST/OUTPT1/ JMAX,WA,DNM,WB,VFA,SMLATP,DLTYTP,DLTYAB,SMLB,
     1DLTYBB,KS,ROS,CS,KAPA,KAPB,KAPS,Z,ZPR
      NAMELIST/OUTPT2/ THETAA,CHI,D,QSDMLS,QBDMLS,QMDMLS,FDMLS,B,KMLTLR
      NAMELIST/OUTPT3/TWBAR,YSSTBR,XSTBAR,AMM,TAUMLT,XMLT,M,N,RBR,ITERA,
     1N5,N5MAX,LFLAG,GWBRS,DGWBRS,QLDMLS
      NAMELIST/OUTPT4/ TAULR,DTAULR,XLR,DSCRIM,N,XTD,CAPOA,CAPOB,CAPOS
      NAMELIST/OUTPT5/N2,N2MXP1,TAPIN,DTAPIN,XLR,TAPLR,DTAPLR,TAUOPS,
     1GOPPS,DGOPPS,RF,NRP,YSWP,XWP,NXSTP,XSTP,NYSSTP,YSSTP,
     2RFNMLT,NRPNM,YSWPNM,XWPNM,NXSTPN,XSTPNM,YSSTPN
      WRITE(6,INPUT)
      WRITE(6,OUTPT1)
      WRITE(6,OUTPT2)
      WRITE(6,OUTPT3)
      WRITE(6,OUTPT4)
      IF(ITERA .NE. 0) RETURN
      WRITE(6,OUTPT5)
      RETURN
      END
```

```
      SUBROUTINE PFIT (M,X,Y,SIG,NMAX,SEEKN,COMTRN,CHBBAS,P,NFIT,SIGFAC,PFT00100
     1W)                                                                PFT00200
C        C.L.LAWSON, JPL, 1969 DEC 10                                   PFT00300
C     C.L.LAWSON, JPL, 1970 JAN 12      CALLING SEQUENCE CHANGED        PFT00400
C        LEAST SQUARES POLYNOMIAL FIT TO DISCRETE DATA.                 PFT00500
C     M              NO. OF DATA POINTS                                 PFT00600
C                                                                       PFT00700
C     (X(I),I=1,M)  ABCISSAS OF DATA                                    PFT00800
C                                                                       PFT00900
C     (Y(I),I=1,M)  ORDINATES OF DATA                                   PFT01000
C                                                                       PFT01100
C     (SIG(I),I=1,M)  STANDARD DEVIATIONS OF DATA Y()                   PFT01200
C                     IF SIG(1) .LT. 0., THE SUBR WILL FUNCTION AS      PFT01300
C                     THOUGH ALL SIG(I) ARE EQUAL TO ABS(SIG(1))        PFT01400
C                                                                       PFT01500
C     NMAX            NMAX      SPECIFIES HIGHEST DEGREE POLYNOMIAL TO  PFT01600
C                     BE CONSIDERED.                                    PFT01700
C                                                                       PFT01800
C     SEEKN           IF .TRUE.       THE SUBR WILL DETERMINE OPTIMUM   PFT01900
C                     NFIT NOT EXCEEDING NMAX.                          PFT02000
C                     IF .FALSE.      THE SUBR WILL SET NFIT = NMAX     PFT02100
C                     UNLESS THIS PRODUCES A NEAR-SINGULAR PROBLEM, IN  PFT02200
C                     WHICH CASE NFIT WILL BE REDUCED.                  PFT02300
C                                                                       PFT02400
C     COMTRN  = .TRUE.   SUBR WILL COMPUTE TRANSFORMATION PARAMETERS,   PFT02500
C                     P(1) AND P(2) SO THAT THE TRANSFORMED VARIABLE    PFT02600
C                     RANGES FROM -1. TO +1.                            PFT02700
C             = .FALSE.  SUBR WILL USE P(1) AND P(2) AS SET BY USER.    PFT02800
C                                                                       PFT02900
C     CHBBAS     = .TRUE. MEANS USE CHEBYSHEV BASIS                     PFT03000
C                = .FALSE. MEANS USE MONOMIAL  BASIS                    PFT03100
C                                                                       PFT03200
C     (P(J),J=1,NMAX+3)  P(1) AND P(2) DEFINE A TRANSFORMATION OF THE   PFT03300
C                        INDEPENDENT VARIABLE AS FOLLOWS..              PFT03400
C                                                                       PFT03500
C                             S = ( X - P(1) ) / P(2)                   PFT03600
C                                                                       PFT03700
C                        (P(I+3),I=0,...,NMAX) ARE POLY COEFFS COMPUTED PFT03800
C                        BY THE SUBR.   P(I+3) IS THE COEFF OF S**I     PFT03900
C                        IF MONOMIAL BASIS IS USED AND OF THE I-TH      PFT04000
C                        DEGREE CHEBY POLY IF THE CHEBY BASIS IS USED.  PFT04100
C                                                                       PFT04200
C                        IF NFIT .LT. NMAX THE COEFFS P(I+3) FOR        PFT04300
C                        I .GT. NFIT WILL BE SET TO ZERO.               PFT04400
C                                                                       PFT04500
C                                                                       PFT04600
C     NFIT            DEGREE OF POLY AS DETERMINED BY SUBROUTINE.       PFT04700
C                                                                       PFT04800
C     SIGFAC             FACTOR BY WHICH THE GIVEN SIG() VALUES SHOULD  PFT04900
C                        BE MULTIPLIED TO IMPROVE CONSISTENCY WITH THE  PFT05000
C                        FIT.      SIGFAC IS THE SQUARE ROOT OF THE     PFT05100
C                        RATIO OF(THE SUM OF SQUARES OF RESIDUALS)      PFT05200
C                              TO(M-NFIT-1).                            PFT05300
C                                                                       PFT05400
C     W()                WORKING SPACE. MUST BE DIMENSIONED AT LEAST    PFT05500
C                        ( MAX(2*NMAX,20) )*(NMAX+2)                    PFT05600
C                                                                       PFT05700
      LOGICAL           SEEKN, COMTRN, CHBBAS                           PFT05800
      REAL              X(M),Y(M),SIG(M),P(1),W(1)                      PFT05900
```

```
C                       EPS = 2**(-27)                                  PFT06000
      DATA              EPS/ .745E-8/                                   PFT06100
      DATA              TOL/1.E-7/                                      PFT06200
      DATA      ONE,TWO,HALF/1.E0,2.E0,.5E0/, ZERO/0.E0/                PFT06300
C                                                                       PFT06400
C         DEFINE A(,) TO BE A FUNCTION                                  PFT06500
C     A(I,J)=W(I+(J-1)*IDIM)                                            PFT06600
      A(I,J)=W(-IDIM+I+J*IDIM)                                          PFT06700
C                                                                       PFT06800
      N=NMAX                                                            PFT06900
      NP1=N+1                                                           PFT07000
      NP2=NP1+1                                                         PFT07100
      IF (N.LT.0.OR.M.LE.0) GO TO 320                                   PFT07200
      IDIM=MAX0(2*NMAX,20)                                              PFT07300
      SIGMA=ABS(SIG(1))                                                 PFT07400
C                   ZERO  FIRST N+2 LOCATIONS OF COLUMN N+2             PFT07500
C                   THIS IS DONE TO CLEAN UP OUTPUT IN CASES IN         PFT07600
C                   WHICH M IS LESS THAN N+2.  IT IS NOT                PFT07700
C                   NECESSARY FOR THE COMPUTATION.                      PFT07800
      DO 10 I=1,NP2                                                     PFT07900
   10 A(I,NP2)=ZERO                                                     PFT08000
C                                       COMPUTE P(1),P(2) IF REQUESTED  PFT08100
C                                                                       PFT08200
C     CHANGE OF INDEPENDENT VARIABLE IS GIVEN BY  S=(X-P(1))/P(2)       PFT08300
C                                             OR  X=P(1) + P(2)*S       PFT08400
C                                                                       PFT08500
      IF (COMTRN) GO TO 20                                              PFT08600
      IF (P(2)) 60,320,60                                               PFT08700
   20 CONTINUE                                                          PFT08800
      XMIN=X(1)                                                         PFT08900
      XMAX=XMIN                                                         PFT09000
      IF (M.EQ.1) GO TO 40                                              PFT09100
      DO 30 I=2,M                                                       PFT09200
      XMIN=AMIN1(XMIN,X(I))                                             PFT09300
   30 XMAX=AMAX1(XMAX,X(I))                                             PFT09400
   40 CONTINUE                                                          PFT09500
      P(1)=(XMAX+XMIN)*HALF                                             PFT09600
      P(2)=(XMAX-XMIN)*HALF                                             PFT09700
      IF (P(2)) 60,50,60                                                PFT09800
   50 P(2)=ONE                                                          PFT09900
C                                                                       PFT10000
C                                                                       PFT10100
   60 CONTINUE                                                          PFT10200
C                                 INITIALIZE FOR ACCUMULATION           PFT10300
      CALL BHSLR1 (W,IDIM,NP1,A(1,NP2),IDIM,1,IR)                       PFT10400
      I=1                                                               PFT10500
C                                                                       PFT10600
C                                    ACCUMULATION LOOP BEGINS HERE      PFT10700
   70 MEQ=MIN0(IDIM-IR,M-I)+1                                           PFT10800
      IF (MEQ.LE.0) GO TO 150                                           PFT10900
      KMAX=IR+MEQ-1                                                     PFT11000
      DO 140 K=IR,KMAX                                                  PFT11100
      S=(X(I)-P(1))/P(2)                                                PFT11200
      IF (SIG(1)) 90,320,80                                             PFT11300
   80 SIGMA=SIG(I)                                                      PFT11400
      IF (SIGMA) 320,320,90                                             PFT11500
   90 CONTINUE                                                          PFT11600
      A(K,1)=ONE/SIGMA                                                  PFT11700
      A(K,NP2)=Y(I)/SIGMA                                               PFT11800
```

```
      IF (N.LE.0) GO TO 130                                              PFT11900
      IF (CHBBAS) GO TO 110                                              PFT12000
C                                                  MONOMIAL BASIS         PFT12100
      DO 100 J=2,NP1                                                     PFT12200
  100 A(K,J)=S*A(K,J-1)                                                  PFT12300
      GO TO 130                                                          PFT12400
C                                                  CHEBYSHEV BASIS        PFT12500
  110 CONTINUE                                                           PFT12600
      A(K,2)=S/SIGMA                                                     PFT12700
      IF (N.EQ.1) GO TO 130                                              PFT12800
      FAC=TWO*S                                                          PFT12900
      DO 120 J=3,NP1                                                     PFT13000
  120 A(K,J)=FAC*A(K,J-1)-A(K,J-2)                                       PFT13100
C                                                                        PFT13200
C                                                                        PFT13300
C                                                                        PFT13400
  130 CONTINUE                                                           PFT13500
      I=I+1                                                              PFT13600
  140 CONTINUE                                                           PFT13700
      CALL BHSLR2 (MEQ,IR)                                               PFT13800
      GO TO 70                                                           PFT13900
  150 CONTINUE                                                           PFT14000
C                                                                        PFT14100
C                                          END OF ACCUMULATION LOOP      PFT14200
C                                                                        PFT14300
C          TEMPORARILY COPY RT-SIDE VECTOR INTO P()                      PFT14400
C                                                                        PFT14500
      DO 160 I=1,NP1                                                     PFT14600
  160 P(I+2)=A(I,NP2)                                                    PFT14700
C                                                                        PFT14800
C     USUALLY IR WILL EQUAL NMAX+3.  IF M .LE. NMAX+1,OR POSSIBLY FOR    PFT14900
C     OTHER REASONS, IR WILL BE LESS THAN NMAX+3.  IN THIS CASE SET ROW  PFT15000
C     IR TO ZERO AND REPLACE IR BY IR+1.     THIS SIMPLIFIES LATER       PFT15100
C     LOGIC BY ASSURING THAT NFIT WILL BE .LE. IR-3.                     PFT15200
C                                                                        PFT15300
      IF (IR.GE.NMAX+3) GO TO 180                                        PFT15400
      DO 170 J=1,NP2                                                     PFT15500
  170 A(IR,J)=ZERO                                                       PFT15600
      IR=IR+1                                                            PFT15700
  180 CONTINUE                                                           PFT15800
C                          COMPUTE NORMS OF SUCCESSIVE RESIDUAL VECTORS  PFT15900
C                                                                        PFT16000
      K=IR-2                                                             PFT16100
      A(K+1,NP2)=ABS(A(K+1,NP2))                                         PFT16200
  190 IF (K.LE.0) GO TO 200                                              PFT16300
      CALL SL2NRM (2,A(K,NP2),A(K,NP2))                                  PFT16400
      K=K-1                                                              PFT16500
      GO TO 190                                                          PFT16600
  200 CONTINUE                                                           PFT16700
C                                                                        PFT16800
C          DIVIDE NORM IN ROW I        BY SQRT(M - (I-1) )               PFT16900
C          AND LOCATE SMALLEST RESULT                                    PFT17000
      IRM1=IR-1                                                          PFT17100
      IMIN=1                                                             PFT17200
      TEMP=M                                                             PFT17300
      AMIN=A(1,NP2)                                                      PFT17400
      BIAS=AMIN*EPS                                                      PFT17500
      DO 210 I=1,IRM1                                                    PFT17600
      A(I,NP2)=A(I,NP2)/SQRT(TEMP)                                       PFT17700
```

```
      TEMP=AMAX1(TEMP-ONE,ONE)                                          PFT17800
      IF (A(I,NP2).GE.AMIN) GO TO 210                                   PFT17900
      AMIN=A(I,NP2)-BIAS                                                PFT18000
      IMIN=I                                                            PFT18100
  210 CONTINUE                                                          PFT18200
C                                                                       PFT18300
C               OPTION TO USE BEST N OR MAXIMUM N                       PFT18400
C                                                                       PFT18500
      IF (SEEKN) GO TO 220                                              PFT18600
      NFT=MIN0(NMAX,IR-3)                                               PFT18700
      GO TO 230                                                         PFT18800
  220 NFT=IMIN-2                                                        PFT18900
  230 CONTINUE                                                          PFT19000
      NFT=MAX0(NFT,0)                                                   PFT19100
C                                                                       PFT19200
C           TEST THE FIRST NFT+1 DIAG ELTS FOR NEAR-SINGULARITY         PFT19300
C                                                                       PFT19400
C                                                                       PFT19500
C                                                                       PFT19600
C                                                                       PFT19700
      N2=NFT+1                                                          PFT19800
      IF (N2.LT.2) GO TO 260                                            PFT19900
      DO 240 I=2,N2                                                     PFT20000
      CALL SL2NRM (I-1,A(1,I),T)                                        PFT20100
      IF (ABS(A(I,I)).LE.TOL*T) GO TO 250                               PFT20200
  240 CONTINUE                                                          PFT20300
      GO TO 260                                                         PFT20400
  250 NFT=I-2                                                           PFT20500
  260 CONTINUE                                                          PFT20600
      NFIT=NFT                                                          PFT20700
C                                                                       PFT20800
C                          SOLVE FOR COEFFS FOR DEGREE NFT              PFT20900
C                                                                       PFT21000
      SIGFAC=A(NFT+2,NP2)                                               PFT21100
      NFP1=NFT+1                                                        PFT21200
      DO 290 I=NFP1,1,-1                                                PFT21300
      IP1=I+1                                                           PFT21400
      T=P(I+2)                                                          PFT21500
      IF (I.EQ.NFP1) GO TO 280                                          PFT21600
      DO 270 J=IP1,NFP1                                                 PFT21700
  270 T=T-A(I,J)*P(J+2)                                                 PFT21800
  280 P(I+2)=T/A(I,I)                                                   PFT21900
  290 CONTINUE                                                          PFT22000
      IF (NFT.EQ.NMAX) GO TO 310                                        PFT22100
      N1=NFT+2                                                          PFT22200
      N2=NMAX+1                                                         PFT22300
      DO 300 I=N1,N2                                                    PFT22400
  300 P(I+2)=ZERO                                                       PFT22500
  310 CONTINUE                                                          PFT22600
      RETURN                                                            PFT22700
C                          ****** ERROR TERMINATION ******              PFT22800
  320 CONTINUE                                                          PFT22900
      WRITE (6,330) M,SIG(1),N,COMTRN,P(2),SIGMA                        PFT23000
      NFIT=-1                                                           PFT23100
      RETURN                                                            PFT23200
  330 FORMAT (48HOBAD VALUES INPUT TO PFIT.  NO FIT WILL BE DONE.,12X,1HPFT23300
     1M,7X,6HSIG(1),9X,4HNMAX,7X,6HCOMTRN,9X,4HP(2),8X,5HSIGMA/48X,I3,EPFT23400
     213.5,I3,L13,2E13.5/)                                              PFT23500
      END                                                               PFT23600
```

```
CIBFTC BHSLR.  LIST BASIC SEQ. LEAST SQUARES WITH H.H. TRANS.  30 APR 68BHSL  10
C       BHSLR.  S.P. BASIC SEQ. LEAST SQUARES WITH H.H. TRANS.  30 APR 68BHSL  20
C                    R. HANSON, J.P.L.                                  BHSL  30
C                                                                       BHSL  40
C      ENTRY POINTS  BHSLR1,BHSLR2..                                    BHSL  50
C                                                                       BHSL  60
C      ****                                                             BHSL  70
       SUBROUTINE BHSLR1 (A,NDA,N,B,NDB,NB,RI)                          BHSL  80
C      ****                                                             BHSL  90
       INTEGER SMI,RI,NDA,N,NDB,NB                                      BHSL 100
C                                                                       BHSL 110
C      START PLACING EQUS. OF COND. AND RT. HAND SIDES IN ROW INDEXED   BHSL 120
C      WITH RI AND PROCEED DOWNWARD FOR MEQ EQUS.                       BHSL 130
C                                                                       BHSL 140
C      THE SPACE  A(I,J),(I = 1,...,RI-1 ,J=1,...,I-1) IS FREE AND      BHSL 150
C      NEVER REFERENCED AGAIN BY THIS PROGRAM AS RI INCREASES TOWARD N+2.BHSL 160
C                                                                       BHSL 170
C      NOW USE  THE  SECOND ENTRY BHSLR2.                               BHSL 180
C      ..                                                               BHSL 190
       REAL A(NDA,N),B(NDB,NB),UP,ZERO                                  BHSL 200
       DATA ZERO/0.E0/                                                  BHSL 210
       RI=1                                                             BHSL 220
       SMI=0                                                            BHSL 230
       RETURN                                                           BHSL 240
C                                                                       BHSL 250
C      PRIMARY ENTRY TO PACK ARRAY TO UPPER  TRIANGULAR FORM.           BHSL 260
C      ..                                                               BHSL 270
C      ****                                                             BHSL 280
       ENTRY BHSLR2(MEQ,RI)                                             BHSL 290
C      ****                                                             BHSL 300
C      ..                                                               BHSL 310
C      MEQ NEW EQUS. AND RT. HAND SIDES ENTER.                          BHSL 320
C                                                                       BHSL 330
       IF (MEQ.LE.0) RETURN                                             BHSL 340
C      ..                                                               BHSL 350
C      NO TRANSFORMATIONS NEEDED.                                       BHSL 360
C                                                                       BHSL 370
       SMI=SMI+MEQ                                                      BHSL 380
       M=RI+MEQ-1                                                       BHSL 390
       NT=MIN0(M,N)                                                     BHSL 400
       DO 20 IP=1,NT                                                    BHSL 410
       IF (IP.EQ.M) GO TO 20                                            BHSL 420
       L=MAX0(0,RI-IP-1)                                                BHSL 430
       CALL AHLR1 (A(1,IP),1,UP,IP-1,L,M,A(1,IP+1),1,NDA,N-IP)          BHSL 440
C      ..                                                               BHSL 450
C      COMPUTE TRANSFORMATION AND APPLY TO THE REMAINING COLUMNS.       BHSL 460
C                                                                       BHSL 470
       CALL AHLR3 (A(1,IP),1,UP,IP-1,L,M,B,1,NDB,NB)                    BHSL 480
C      ..                                                               BHSL 490
C      APPLY TRANSFORMATIONS TO B,IF PRESENT.                           BHSL 500
C                                                                       BHSL 510
       L=L+IP+1                                                         BHSL 520
       DO 10 I=L,M                                                      BHSL 530
   10  A(I,IP)=ZERO                                                     BHSL 540
C      ..                                                               BHSL 550
C      CLEAR AREA JUST MADE IMPLICITLY ZERO.                            BHSL 560
C                                                                       BHSL 570
   20  CONTINUE                                                         BHSL 580
       RI=MIN0(SMI,N+1)+1                                               BHSL 590
```

```
      IF (M.LE.N) RETURN                                                BHSL 600
      IF (NB.LE.0) RETURN                                               BHSL 610
      DO 30 J=1,NB                                                      BHSL 620
   30 CALL SL2NRM (M-N,B(N+1,J),B(N+1,J))                               BHSL 630
C     ..                                                                BHSL 640
C     PACK LENGTHS OF RT. HAND SIDES OUTSIDE THE COL. SPACE OF A        BHSL 650
C     TO SINGLE LOCATIONS EACH.                                         BHSL 660
C                                                                       BHSL 670
C     ..                                                                BHSL 680
C     ADVANCE NEW ORIGIN OF ROW TO START EQUS. OF CONDITION AND RT.SIDESBHSL 690
C                                                                       BHSL 700
      L=N+2                                                             BHSL 710
      IF(L.GT.M) RETURN                                                 BHSL 720
      DO 40 J=1,NB                                                      BHSL 730
      DO 40 I=L,M                                                       BHSL 740
   40 B(I,J)=ZERO                                                       BHSL 750
      RETURN                                                            BHSL 760
C     ..                                                                BHSL 770
C     IF A SOLUTION IS DESIRED WITH FURTHER ACCUMULATION INTENDED,      BHSL 780
C     SAVE THE UPPER TRIANGULAR PART OF THE A ARRAY AND THE FIRST       BHSL 790
C     N+1 ROWS OF THE B ARRAY. FOLLOWING CALCULATION OF THE SOLUTION,   BHSL 800
C     RESTORE THESE DATA AND ZERO THE PART OF A BELOW THE MAIN          BHSL 810
C     DIAGONAL TOGETHER WITH THE (N+1)ST ROW. SEQ. ACCUMULATION MAY THENBHSL 820
C     CONTINUE AS BEFORE.   THE NUMBER OF ROWS IN THE PROCESSED MATRIX  BHSL 830
C     TO BE SOLVED WILL BE RI-1.                                        BHSL 840
C                                                                       BHSL 850
      END                                                               BHSL 860
```

```
C       BHTFLR  LIST BASIC H.H. TRANS. FROM THE LEFT.  S.P.      10 JUN 68HTR00100
C       ENTRY POINTS..                                                    HTR00200
C                                                                         HTR00300
C       AHLR1, AHLR2, AHLR3                                               HTR00400
C               11 FEB 1970                                               HTR00600
C                                                                         HTR00700
C      **                                                                 HTR00800
       SUBROUTINE AHLR1 (U,NDU,UP,L1,L2,M,P,IRA,ICA,NCOLS)                HTR00900
C      **                                                                 HTR01000
       REAL U(NDU,M),UP,P(1),CL,ZERO                                      HTR01100
       DOUBLE PRECISION SM,DZERO,BETA                                     HTR01200
       LOGICAL SRCLAV,N1GTM,CMPU                                          HTR01300
       DATA DZERO/0.D0/                                                   HTR01400
       DATA ZERO/0.E0/                                                    HTR01500
C      ..                                                                 HTR01600
C      DEFINE DATA FOR LITERALLY NAMED PROGRAM CONSTANTS.                 HTR01700
C                                                                         HTR01800
       REAL AV,SR,AM,SGN                                                  HTR01900
       AV(DUM)=ABS(DUM)                                                   HTR02000
       SR(DUM)=SQRT(DUM)                                                  HTR02100
       AM(D1,D2)=AMAX1(D1,D2)                                             HTR02200
       SGN(D1,D2)=SIGN(D1,D2)                                             HTR02300
C      ..                                                                 HTR02400
C      DFINE FUNCTIONS USED IN THIS SUBROUTINE.                           HTR02500
C                                                                         HTR02600
       CMPU=.TRUE.                                                        HTR02700
       SRCLAV=.FALSE.                                                     HTR02800
C      ..                                                                 HTR02900
C      ENTIRE RAW U VECTOR IN THE FIRST ROW OF U.                         HTR03000
C                                                                         HTR03100
       GO TO 10                                                           HTR03200
C      **                                                                 HTR03300
       ENTRY AHLR2(U,NDU,UP,SRCL,L1,L2,M,P,IRA,ICA,NCOLS)                 HTR03400
C      **                                                                 HTR03500
       REAL SRCL                                                          HTR03600
       CMPU=.TRUE.                                                        HTR03700
       SRCLAV=.TRUE.                                                      HTR03800
C      ..                                                                 HTR03900
C      RAW U VECTOR IN THE FIRST ROW OF U.  LENGTH OF U VECTOR IS         HTR04000
C      AVAILABLE IN SRCL.                                                 HTR04100
C                                                                         HTR04200
       GO TO 10                                                           HTR04300
C      **                                                                 HTR04400
       ENTRY AHLR3(U,NDU,UP,L1,L2,M,P,IRA,ICA,NCOLS)                      HTR04500
C      **                                                                 HTR04600
       CMPU=.FALSE.                                                       HTR04700
       SRCLAV=.FALSE.                                                     HTR04800
C      ..                                                                 HTR04900
C      U TRANSFORMATION CALCULATED AND IN U.  PIVOT ELEMENT IS IN UP NOW.HTR05000
C                                                                         HTR05100
   10  IF (L1.LT.0.OR.L2.LT.0.OR.M.LE.0) RETURN                           HTR05200
C      ..                                                                 HTR05300
C      RETURN IF TRANSFORMATION IS ABSENT OR NOT WELL-DEFINED.            HTR05400
C                                                                         HTR05500
       N1=L1+L2+2                                                         HTR05600
       N1GTM=N1.GT.M                                                      HTR05700
       L1P1=L1+1                                                          HTR05800
       IF (.NOT.CMPU) GO TO 50                                            HTR05900
       CL=SRCL                                                            HTR06000
```

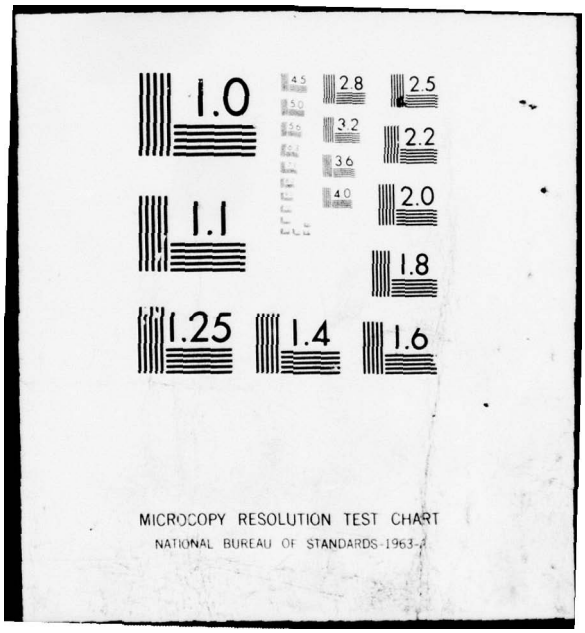
```
      IF (SRCLAV) GO TO 40                                                  HTR06100
      CL=U(1,L1P1)                                                          HTR06200
      IF (CL.LT.ZERO) CL=-CL                                                HTR06300
      IF (N1GTM) GO TO 40                                                   HTR06400
      DO 20 J=N1,M                                                          HTR06500
   20 CL=AM(AV(U(1,J)),CL)                                                  HTR06600
      IF (CL.LE.ZERO) RETURN                                                HTR06700
      SM=(U(1,L1P1)/CL)**2                                                  HTR06800
      DO 30 J=N1,M                                                          HTR06900
   30 SM=SM+(U(1,J)/CL)**2                                                  HTR07000
      CL=CL*SR(SM)                                                          HTR07100
C     ..                                                                    HTR07200
C     COMPUTE LENGTH OF U VECTOR IF SRCLAV = .FALSE..                       HTR07300
C                                                                           HTR07400
   40 IF (CL.LE.ZERO) RETURN                                                HTR07500
      CL=-SGN(CL,U(1,L1P1))                                                 HTR07600
      UP=U(1,L1P1)-CL                                                       HTR07700
      U(1,L1P1)=CL                                                          HTR07800
   50 IF (NCOLS.LE.0) RETURN                                                HTR07900
C     ..                                                                    HTR08000
C     RETURN IF MATRIX P IS ABSENT.                                         HTR08100
C                                                                           HTR08200
      BETA=-U(1,L1P1)*UP                                                    HTR08300
      IF (BETA.EQ.DZERO) RETURN                                             HTR08400
C     ..                                                                    HTR08500
C     IF BETA.EQ.ZERO, NO TRANSFORMATION IS NECESSARY.                      HTR08600
C                                                                           HTR08700
      DO 90 J=1,NCOLS                                                       HTR08800
C                                                                           HTR08900
C     APPLY HOUSEHOLDER TRANSFORMATION  (I - 2*U*U**(T)) TO  P              HTR09000
C     FROM THE LEFT.                                                        HTR09100
C                                                                           HTR09200
C                                                                           HTR09300
C     IRA = NUMBER OF ROWS CELLS OF P ARE APART.                            HTR09400
C                                                                           HTR09500
C     ICA = NUMBER OF CELLS COLUMNS OF P ARE APART.                         HTR09600
C                                                                           HTR09700
C     U  = (0,...,(L1 ZEROS), UP,0,...,(L2 ZEROS), U(L1+L2+2),...,U(M))    HTR09800
C     L1 .GE. 0,  L2 .GE. 0.                                                HTR09900
C     ..                                                                    HTR10000
      I1=ICA*(J-1)                                                          HTR10100
      I2=I1+L1*IRA+1                                                        HTR10200
      I4=I1+(N1-1)*IRA+1                                                    HTR10300
      SM=P(I2)*UP                                                           HTR10400
      IF (N1GTM) GO TO 70                                                   HTR10500
      I3=I4                                                                 HTR10600
      DO 60 I=N1,M                                                          HTR10700
      SM=SM+P(I3)*U(1,I)                                                    HTR10800
   60 I3=I3+IRA                                                             HTR10900
   70 IF (SM.EQ.DZERO) GO TO 90                                             HTR11000
C     ..                                                                    HTR11100
C     TRANSFORMATION LEAVES THAT COLUMN OF P  ALONE.                        HTR11200
      SM=SM/BETA                                                            HTR11300
      P(I2)=P(I2)-UP*SM                                                     HTR11400
      IF (N1GTM) GO TO 90                                                   HTR11500
      I3=I4                                                                 HTR11600
      DO 80 I=N1,M                                                          HTR11700
      P(I3)=P(I3)-SM*U(1,I)                                                 HTR11800
   80 I3=I3+IRA                                                             HTR11900
```

AD-A060 045 JET PROPULSION LAB PASADENA CALIF F/G 19/1
COMBUSTION RESPONSE MODELING FOR COMPOSITE SOLID PROPELLANTS.(U)
JUN 78 N S COHEN, J M BOWYER F04611-76-X-0050
UNCLASSIFIED JPL-PUB-78-59 AFRPL-TR-78-39 NL

2 OF 2
AD
A060045



END
DATE
FILMED
12-78
DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
C     ..                                                              HTR12000
C     BASIC STEP OF HOUSEHOLDER TRANSFORMATION.                       HTR12100
C                                                                     HTR12200
   90 CONTINUE                                                        HTR12300
C     ..                                                              HTR12400
C     RETURN WITH TRANSFORMATION APPLIED FROM THE LEFT.               HTR12500
C                                                                     HTR12600
      RETURN                                                          HTR12700
      END                                                             HTR12800
```

```
      SUBROUTINE SL2NRM (N,U,G)                                          L2N00100
C     MODIFIED BY C J DEVINE 3/21/72 TO ALLOW ADDRESS OF .GT. 16 BITS FOR ARGS
C     IN CALLING SEQ                                  DUMMY DIMENSION ARRAY  G(1),
C     WAS DEFINED  .   THIS ALLOWS PROGRAM TO REFERENCE LOCATIONS IN EXT. CORE
C      ENR2.    S.P. AND  D.P. L2 NORM FOR ARRAY BASED VECTORS 22 APR 68L2N00200
C         11 FEB 1970                                                     L2N00300
      REAL     U(1),     G(1)                                            L2N00400
      F1=0.E0                                                            L2N00500
      F2=0.E0                                                            L2N00600
      IF (N.LE.0) GO TO 20                                               L2N00700
      DO 10 J=1,N                                                        L2N00800
   10 F1=AMAX1(F1,ABS(U(J)))                                             L2N00900
      IF (F1.NE.0.) GO TO 30                                             L2N01000
   20 G(1)=0.0E0                                                         L2N01100
      RETURN                                                             L2N01200
   30 DO 40 J=1,N                                                        L2N01300
      T=U(J)/F1                                                          L2N01400
   40 F2=F2+T*T                                                          L2N01500
      G(1)=F1*SQRT(F2)                                                   L2N01600
      RETURN                                                             L2N01700
      END                                                                L2N01800
```

```
      FUNCTION SCPVAL (P,NDEGP,X)
C
C     C.L.LAWSON,JPL, 1969 DEC 17   MODIFIED 1973 JULY 24
C
C     MODIFIED 1974 NOV 19
C
C     EVALUATE A POLYNOMIAL OF DEGREE NDEGP GIVEN TRANSFORMATION
C     PARAMETERS, P(1) AND P(2), AND COEFFICIENTS RELATIVE TO THE
C     CHEBYSHEV BASIS.
C
C     NDEGP                  DEGREE OF POLYNOMIAL
C     (P(I),I=1,NDEGP+3)  PARAMETERS DEFINING THE POLYNOMIAL
C     X                     INPUT ARGUMENT
C             THE POLYNOMIAL-S VALUE AT X IS DEFINED AS FOLLOWS.
C
C                                S = ( X - P(1) ) / P(2)
C
C                SCPVAL=SUM OF P(I+3)*T(I,S) FOR I=0,1,...NDEGP
C
C                                WHERE T(I,S) DENOTES THE CHEBYSHEV
C                                POLYNOMIAL OF DEGREE I EVALUATED AT S .
C
      REAL               P(1),W(3),S,S2,X
      W(1)=0.
      W(2)=0.
C                                TRANSFORM X TO S
      S=(X-P(1))/P(2)
      S2=S+S
      J=NDEGP+3
C
C                                EVALUATE POLYNOMIAL USING RECURSION
C
   10     IF(J .LE. 3) GO TO 20
          W(3)=W(2)
          W(2)=W(1)
          W(1)=(S2*W(2)-W(3))+P(J)
          J = J - 1
          GO TO 10
   20 SCPVAL=(S*W(1)-W(2))+P(3)
      RETURN
      END
```

```
      SUBROUTINE SVDQ (NEQ,T,Y,F,KD,EP,IFLAG,H,HMINA,HMAXA,DELT,          SVDQ0001
     1  TFINAL,MXSTEP,KSTEP,KEMAX,EMAX,KQ,YN,DT)                         SVDQ0002
C                                                                        SVDQ0003
C     VARIABLE ORDER INTEGRATION SUBROUTINE FOR THE                      SVDQ0004
C     SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS                        SVDQ0005
C                                                                        SVDQ0006
C     ANALYSIS AND CODING BY FRED T. KROGH,  AT THE JET PROPULSION       SVDQ0007
C     LABORATORY, PASADENA, CALIF.  APRIL 1, 1969.                       SVDQ0008
C     THIS SUBROUTINE DESIGNED FOR THE IBM 7094 AND THE UNIVAC 1108.     SVDQ0009
C                                                                        SVDQ0010
C     AT THE END OF THIS LISTING INSTRUCTIONS ARE GIVEN FOR REMOVING     SVDQ0011
C     SOME FEATURES AND FOR ADDING OTHERS. THE GSTOP FEATURE IS          SVDQ0012
C     EXPLAINED NEAR THE END OF THE LISTING.                             SVDQ0013
C                                                                        SVDQ0014
C     VARIABLES IN THE CALLING SEQUENCE HAVE THE FOLLOWING TYPES.        SVDQ0015
      INTEGER NEQ,KD(1),IFLAG,MXSTEP,KSTEP,KEMAX,KQ(1)                   SVDQ0016
      REAL    F(1),EP(1),HMINA,HMAXA,EMAX,DT(10,1)                       SVDQ0017
      REAL     T,Y(1),H,DELT,TFINAL,YN(1)                                SVDQ0018
C                                                                        SVDQ0019
C     PARAMETERS WHICH MUST BE ASSIGNED VALUES BEFORE CALLING            SVDQ0020
C     SVDQ ARE    NEQ, T, Y, KD, H, HMINA, HMAXA, DELT,                  SVDQ0021
C                 TFINAL, AND MXSTEP.                                    SVDQ0022
C     SVDQ IS USED ONLY ON THE INITIAL ENTRY. ALL OTHER                  SVDQ0023
C     ENTRIES ARE MADE BY CALLING SVDQ1. IN ADDITION TO                  SVDQ0024
C     THE PARAMETERS MENTIONED ABOVE THE USER MUST ASSIGN                SVDQ0025
C     VALUES TO F (ONCE PER STEP INITIALLY, AND TWICE PER STEP           SVDQ0026
C     AFTER GETTING STARTED) AND EP (EITHER INITIALLY, OR DURING         SVDQ0027
C     THE INTEGRATION IF A RELATIVE ERROR TEST IS USED).                 SVDQ0028
C     THE FOLLOWING PARAMETERS GIVE ADDITIONAL INFORMATION ABOUT THE     SVDQ0029
C     INTEGRATION AND ARE USED FOR STORAGE. THEY SHOULD NOT BE           SVDQ0030
C     CHANGED BY THE USER.  IFLAG,KSTEP,KEMAX,EMAX,KQ,YN, AND DT.       SVDQ0031
C                                                                        SVDQ0032
C                                                                        SVDQ0033
C                                                                        SVDQ0034
C     AN EXAMPLE OF HOW ONE MIGHT SET UP THE CALLS TO SVDQ IS GIVEN      SVDQ0035
C     BELOW.                                                             SVDQ0036
C     CALL SVDQ(NEQ,....,DT)                                             SVDQ0037
C     GO TO YYY                                                          SVDQ0038
C XXX CALL SVDQ1                                                         SVDQ0039
C YYY GO TO (N1,N2,....,N8), IFLAG                                       SVDQ0040
C     AFTER THE INITIAL CALL, RETURN TO SVDQ WITH                        SVDQ0041
C     GO TO XXX                                                          SVDQ0042
C     IF NO ERROR HAS BEEN MADE, IFLAG WILL EQUAL 1 AFTER THE INITIAL    SVDQ0043
C     CALL INDICATING THE DERIVATIVES ARE TO BE COMPUTED.                SVDQ0044
C                                                                        SVDQ0045
C                                                                        SVDQ0046
C     THE USAGE OF THE VARIABLES IS GIVEN BELOW.                         SVDQ0047
C                                                                        SVDQ0048
C     NEQ=NUMBER OF EQUATIONS (INPUT)                                    SVDQ0049
C                                                                        SVDQ0050
C     T=INDEPENDENT VARIABLE (INITIAL VALUE SUPPLIED BY THE USER)        SVDQ0051
C                                                                        SVDQ0052
C     Y=CURRENT VALUE OF DEPENDENT VARIABLE. THE INITIAL                 SVDQ0053
C       VALUE OF Y MUST BE SPECIFIED BY THE USER BEFORE                  SVDQ0054
C       THE FIRST ENTRY. THE DIMENSION OF Y MUST BE                      SVDQ0055
C       AT LEAST AS GREAT AS THE SUM OF THE ORDERS OF                    SVDQ0056
C       THE DIFFERENTIAL EQUATIONS WHICH ARE BEING                       SVDQ0057
C       INTEGRATED. IF WE LET KD(I) DENOTE THE ORDER                     SVDQ0058
C       OF THE I-TH DIFFERENTIAL EQUATION, THEN Y(J)                     SVDQ0059
```

```
C       IS THE K-TH DERIVATIVE OF THE L-TH COMPONENT.                    SVDQ0060
C       WHERE L IS THE SMALLEST INTEGER FOR WHICH                        SVDQ0061
C       KD(1)+KD(2)+...+KD(L).GE.J  AND K=KD(L)+J-1-(KD(1)               SVDQ0062
C       +KD(2)+...+KD(L)), J=1,2,...,(KD(1)+KD(2)+...+KD(NEQ)).          SVDQ0063
C       (FOR EXAMPLE, FOR THE SYSTEM  F(1)=UPP, F(2)=VPP. WHERE P        SVDQ0064
C       DENOTES A PRIME, Y(1)=U, Y(2)=UP, Y(3)=V, Y(4)=VP.)              SVDQ0065
C                                                                        SVDQ0066
C     F(I)=KD(I)-TH DERIVATIVE OF THE I-TH COMPONENT WITH RESPECT        SVDQ0067
C       TO T, I=1,2,...,NEQ. THE USER MUST PROVIDE                       SVDQ0068
C       THE CODE WHICH COMPUTES F GIVEN Y AND T.                         SVDQ0069
C                                                                        SVDQ0070
C     KD GIVES THE ORDER OF THE DIFFERENTIAL EQUATIONS IN THE            SVDQ0071
C        SYSTEM. KD MUST BE LESS THAN OR EQUAL TO 4.                     SVDQ0072
C        (FOR DIFFERENTIAL EQUATIONS WITH DIFFERENT ORDERS SET           SVDQ0073
C        KD.LT.0. IF THIS IS DONE IT IS ASSUMED THAT KD IS A VECTOR      SVDQ0074
C        AND THAT ABS(KD(I)) GIVES THE ORDER OF THE I-TH EQUATION.)      SVDQ0075
C                                                                        SVDQ0076
C     EP IS A PARAMETER USED TO CONTROL THE LOCAL ERROR.                 SVDQ0077
C     IF EP IS POSITIVE THE LOCAL ERROR IS KEPT LESS                     SVDQ0078
C     THAN EP IN ALL COMPONENTS OF THE DIFF. EQ.                         SVDQ0079
C     (THE ESTIMATED LOCAL ERROR IS KEPT LESS THAN EP IN                 SVDQ0080
C     THE (KD(I)-1)-ST DERIVATIVE OF THE I-TH COMPONENT. THUS            SVDQ0081
C     FOR EQUATIONS WITH ORDER GREATER THAN ONE, THE ERROR               SVDQ0082
C     IN A DERIVATIVE IS ESTIMATED. IN THIS CASE THE VALUE OF            SVDQ0083
C     EP REQUIRED TO OBTAIN A GIVEN ACCURACY IN THE DEPENDENT            SVDQ0084
C     VARIABLE DEPENDS ON THE SCALING.)                                  SVDQ0085
C     IF EP.LT.0, THEN IT IS ASSUMED THAT EP                             SVDQ0086
C     IS A VECTOR. LET K BE THE SMALLEST VALUE                           SVDQ0087
C     OF I FOR WHICH EP(I).GE.0.  FOR I.LT.K                             SVDQ0088
C     THE LOCAL ERROR CONTROL IS BASED ON                                SVDQ0089
C     ABS(EP(I)), AND FOR I.GE.K IT IS BASED ON                          SVDQ0090
C     EP(K). IF ONE WANTS A RELATIVE ERROR TEST--                        SVDQ0091
C     FOR EXAMPLE, THE LOCAL ERROR IS TO BE KEPT                         SVDQ0092
C     LESS THAN C*P WHERE C IS A CONSTANT                                SVDQ0093
C     AND P IS A POSITIVE FUNCTION OF T AND Y,                           SVDQ0094
C     THEN ONE SHOULD SET EP=C*P WHEN IFLAG=1.                           SVDQ0095
C     IF EP=0 AND HMAXA.NE.0, IFLAG IS SET EQUAL 8. IF EP=0 AND HMAXA=0.SVDQ0096
C     NO ERROR TESTS ARE PERFORMED AND THE ORDER(S) AND STEPSIZE ARE     SVDQ0097
C     NOT CHANGED. THIS OPTION SHOULD NOT BE USED IF KQ(I)=1 FOR ANY I.  SVDQ0098
C                                                                        SVDQ0099
C     IFLAG IS USED FOR COMMUNICATION BETWEEN THE INTEGRATOR             SVDQ0100
C           AND THE PROGRAM WHICH CALLS IT.  THE VALUE                   SVDQ0101
C           OF IFLAG SHOULD NOT BE CHANGED BY THE USER.                  SVDQ0102
C        THE FOLLOWING VALUES OF IFLAG HAVE THE FOLLOWING MEANINGS.      SVDQ0103
C     =1  THE VALUE OF Y FOR THE CURRENT STEP HAS BEEN                   SVDQ0104
C         PREDICTED. THE USER SHOULD COMPUTE F AND CALL SVDQ1.           SVDQ0105
C         IF A RELATIVE ERROR TEST IS USED THE NEW VALUE                 SVDQ0106
C         OF EP SHOULD ALSO BE COMPUTED HERE.                            SVDQ0107
C     =2  THE VALUE OF Y FOR THE CURRENT STEP HAS BEEN                   SVDQ0108
C         CORRECTED. THE USER SHOULD COMPUTE F AND CALL SVDQ1.           SVDQ0109
C     =3  AN OUTPUT POINT HAS BEEN REACHED (SEE DESCRIPTION              SVDQ0110
C         OF DELT), PRINT RESULTS AND CALL SVDQ1.                        SVDQ0111
C     =4  T=TFINAL    IF SVDQ1 IS CALLED WITH T=TFINAL AND               SVDQ0112
C         IFLAG=4, IFLAG IS SET EQUAL TO 8. IF THE VALUE OF              SVDQ0113
C         TFINAL IS CHANGED THE INTEGRATION WILL CONTINUE.               SVDQ0114
C     =5  KSTEP=KSOUT   (SEE THE DESCRIPTION OF MXSTEP).                SVDQ0115
C     =6  EMAX.GT..1 AND IT APPEARS TO THE SUBROUTINE THAT               SVDQ0116
C         REDUCING H WILL NOT HELP BECAUSE OF ROUND-OFF ERROR.           SVDQ0117
C         IF THIS OCCURS A LARGER VALUE OF EP (OR OF ABS(EP(KEMAX)) IF   SVDQ0118
```

```
C        EP IS A VECTOR) SHOULD PROBABLY BE USED. IF EP IS NOT          SVDQ0119
C        INCREASED, TOO SMALL A STEPSIZE IS LIABLE TO BE USED. (WE HAVESVDQ0120
C        FOUND THAT REPLACING EP WITH 32.*EMAX*EP WORKS QUITE WELL.)    SVDQ0121
C        INCREASING EP IN THIS WAY WILL NOT DEGRADE THE ACCURACY,       SVDQ0122
C        HOWEVER IF THE NATURE OF THE PROBLEM CHANGES IT MAY PAY TO     SVDQ0123
C        USE A SMALLER VALUE OF EP LATER IN THE INTEGRATION.            SVDQ0124
C     =7 ABS(H).LT.HMINA.  TO CONTINUE WITH THE CURRENT                 SVDQ0125
C        VALUE OF H, SET HMINA.LE.ABS(H) AND CALL SVDQ1.                SVDQ0126
C        IF THE INTEGRATOR HAS JUST HALVED H ONE MAY CONTINUE           SVDQ0127
C        WITH TWICE THE STEPSIZE BY SIMPLY CALLING SVDQ1. (SUCH         SVDQ0128
C        AN ACTION IS RISKY WITHOUT A CAREFUL ANALYSIS OF THE           SVDQ0129
C        SITUATION.) IF THE STEPSIZE HAS NOT JUST BEEN HALVED           SVDQ0130
C        (ABS(H).LT.HMINA MAY BE DUE TO THE USER INCREASING THE         SVDQ0131
C        VALUE OF HMINA OR TO HAVING TOO SMALL AN H AT THE END          SVDQ0132
C        OF THE STARTING PHASE.) THE INTEGRATION WILL CONTINUE          SVDQ0133
C        WITH THE CURRENT VALUE OF H AND A RETURN TO THE USER WITH      SVDQ0134
C        IFLAG=7 WILL BE MADE ON EVERY STEP UNTIL ABS(H).GE.HMINA.      SVDQ0135
C     =8 ILLEGAL PARAMETER IN THE CALLING SEQUENCE. IF SVDQ1            SVDQ0136
C        IS CALLED WITH IFLAG=8 THE PROGRAM IS STOPPED.                 SVDQ0137
C                                                                       SVDQ0138
C     H=CURRENT VALUE OF THE STEPSIZE. IN SELECTING THE INITIAL         SVDQ0139
C       VALUE FOR H, THE USER SHOULD REMEMBER THE FOLLOWING--           SVDQ0140
C        1. THE INTEGRATOR IS CAPABLE OF CHANGING H QUITE QUICKLY AND   SVDQ0141
C           THUS THE INITIAL CHOICE IS NOT CRITICAL.                    SVDQ0142
C        2. IF IT DOES NOT LEAD TO PROBLEMS IN COMPUTING THE DERIVATIVES SVDQ0143
C           (E.G. BECAUSE OF OVERFLOW OR TRYING TO EXTRACT THE SQUARE   SVDQ0144
C           ROOT OF A NEGATIVE NUMBER), IT IS BETTER TO CHOOSE H MUCH   SVDQ0145
C           TOO LARGE THAN MUCH TOO SMALL.                              SVDQ0146
C        3. IF H*DELT.LE.0 INITIALLY, AN IMMEDIATE RETURN IS MADE       SVDQ0147
C           WITH IFLAG=8. THE SIGN OF H IS WHAT DETERMINES THE          SVDQ0148
C           DIRECTION OF INTEGRATION.                                   SVDQ0149
C        4. IF DELT=H*(2**K) K A NONNEGATIVE INTEGER THEN OUTPUT        SVDQ0150
C           VALUES WILL BE OBTAINED WITHOUT DOING AN INTERPOLATION.     SVDQ0151
C                                                                       SVDQ0152
C     HMINA   AFTER GETTING STARTED, AND WHENEVER H                     SVDQ0153
C             IS HALVED, ABS(H) IS COMPARED WITH HMINA.                 SVDQ0154
C             IF ABS(H).LT.HMINA CONTROL IS RETURNED TO                 SVDQ0155
C             THE USER WITH IFLAG=7.                                    SVDQ0156
C                                                                       SVDQ0157
C     HMAXA   THE STEPSIZE IS NOT DOUBLED IF                            SVDQ0158
C             DOING SO WOULD MAKE ABS(H).GT.HMAXA                       SVDQ0159
C                                                                       SVDQ0160
C     DELT  ENABLES THE USER TO SPECIFY THE POINTS WHERE                SVDQ0161
C       OUTPUT IS DESIRED. LET TOUT=DELT + THE VALUE OF T THE LAST      SVDQ0162
C       TIME CONTROL WAS RETURNED TO THE USER WITH IFLAG=3. (INITIALLY  SVDQ0163
C       TOUT=THE INITIAL VALUE OF T.) CONTROL IS RETURNED TO THE        SVDQ0164
C       USER WITH IFLAG=3 WHENEVER T=TOUT. IF TOUT DOES NOT FALL        SVDQ0165
C       ON AN INTEGRATION STEP, OUTPUT VALUES ARE OBTAINED BY           SVDQ0166
C       INTERPOLATION ON THE FIRST STEP THAT (T-TOUT)*H.GT.0.           SVDQ0167
C       INTERPOLATED VALUES FOR BOTH Y AND F ARE COMPUTED.              SVDQ0168
C       (NOTE THAT A RETURN WITH IFLAG=3 IS ALWAYS MADE                 SVDQ0169
C       BEFORE TAKING THE FIRST STEP.)                                  SVDQ0170
C                                                                       SVDQ0171
C     TFINAL   CONTROL IS RETURNED TO THE USER WITH IFLAG=4 WHEN        SVDQ0172
C       T REACHES TFINAL. IF TFINAL DOES NOT FALL ON AN INTEGRATION     SVDQ0173
C       STEP VALUES AT TFINAL ARE OBTAINED BY EXTRAPOLATION.            SVDQ0174
C                                                                       SVDQ0175
C     MXSTEP   ON THE INITIAL ENTRY, AND ON ENTRIES                     SVDQ0176
C       WITH  2.LT.IFLAG.LT.6  KSOUT IS SET EQUAL TO                    SVDQ0177
```

```
C       KSTEP+MXSTEP. AT THE END OF EACH STEP KSTEP IS INCREMENTED      SVDQ0178
C       AND COMPARED WITH KSOUT. IF KSTEP.GE.KSOUT CONTROL IS          SVDQ0179
C       RETURNED TO THE USER WITH IFLAG=5. (THUS IF DELT IS            SVDQ0180
C       SUFFICIENTLY LARGE, CONTROL WILL BE RETURNED TO THE USER       SVDQ0181
C       WITH IFLAG=5 EVERY MXSTEP STEPS.)                              SVDQ0182
C                                                                      SVDQ0183
C     KSTEP=NUMBER OF INTEGRATION STEPS TAKEN (COMPUTED                SVDQ0184
C       BY THE INTEGRATOR.)                                            SVDQ0185
C                                                                      SVDQ0186
C     KEMAX=INDEX OF COMPONENT RESPONSIBLE FOR THE                     SVDQ0187
C           VALUE OF EMAX (SEE BELOW).                                 SVDQ0188
C                                                                      SVDQ0189
C     EMAX=LARGEST VALUE IN ANY COMPONENT OF (ESTIMATED ERROR)/EP      SVDQ0190
C       ORDINARILY THE STEPSIZE IS HALVED IF EMAX.GT..1. WITH A        SVDQ0191
C       RECENT HISTORY OF LOCAL ROUND-OFF PROBLEMS VALUES OF EMAX AS   SVDQ0192
C       LARGE AS 1 ARE PERMITTED. THE STEPSIZE IS NOT HALVED ON ANY    SVDQ0193
C       STEP THAT ROUND OFF ERROR APPEARS TO BE LIMITING THE PRECISION. SVDQ0194
C                                                                      SVDQ0195
C     KQ(I)=HIGHEST ORDER DIFFERENCE USED IN INTEGRATING               SVDQ0196
C       THE I-TH EQUATION. (COMPUTED BY THE INTEGRATOR)                SVDQ0197
C                                                                      SVDQ0198
C     YN=A VECTOR WITH THE DIMENSION OF Y USED TO STORE                SVDQ0199
C       THE VALUE OF Y AT THE END OF EACH INTEGRATION STEP.            SVDQ0200
C                                                                      SVDQ0201
C     DT=AN ARRAY WITH DIMENSION DT(10,NEQ) USED TO                    SVDQ0202
C       STORE THE DIFFERENCE TABLE.                                    SVDQ0203
C                                                                      SVDQ0204
C                                                                      SVDQ0205
      REAL     TOUT,TL,TPD,TPD1,TPD2,HH,FAC                          SVDQ0206
      DIMENSION DD(12),D(11),GAM(10,4),GAS(10),PT(11),FAC(3)         SVDQ0207
      EQUIVALENCE(DD(2),D(1))                                        SVDQ0208
C                                                                    SVDQ0209
      DATA KMAXO/4/                                                  SVDQ0210
C     KMAXO IS THE MAXIMUM ORDER DIFFERENTIAL EQUATION               SVDQ0211
C     THIS SUBROUTINE WILL INTEGRATE.                                SVDQ0212
C                                                                    SVDQ0213
      DATA FAC/1.E0,.5E0,.16666667E0/                                SVDQ0214
C     FAC(J)=1/(FACTORIAL J), J=1,2,...,MAX(2,KMAXO-1)               SVDQ0215
C                                                                    SVDQ0216
      DATA KQMAX/9/                                                  SVDQ0217
C     KQMAX GIVES THE MAXIMUM ORDER.                                 SVDQ0218
C     THERE IS LITTLE POINT IN HAVING KQMAX MUCH BIGGER THAN THE NUMBER SVDQ0219
C     OF DECIMAL DIGITS IN THE MANTISSA.                             SVDQ0220
C     IF KQMAX IS SET LESS THAN 6, DT, D, AND PT SHOULD BE DIMENSIONED SVDQ0221
C     AS IF KQMAX=6.                                                 SVDQ0222
C                                                                    SVDQ0223
      DATA RND,KBIT2/6.0E-8,56/                                      SVDQ0224
C     RND IS APPROXIMATELY 2**(3-B) WHERE B IS                       SVDQ0225
C     THE NUMBER OF BITS IN THE MANTISSA.                            SVDQ0226
C     KBIT2=2*B+2 WHERE B IS THE NUMBER OF BITS IN THE MANTISSA.     SVDQ0227
C     IF THE DERIVATIVES ARE NOT COMPUTED TO THE ACCURACY EXPECTED   SVDQ0228
C     FROM THE WORD LENGTH OF THE COMPUTER (FOR EXAMPLE BECAUSE OF   SVDQ0229
C     CANCELLATION PROBLEMS OR TABULAR DATA), THEN THESE CONSTANTS   SVDQ0230
C     CAN BE CHANGED TO REFLECT THE NUMBER OF BITS WHICH ARE         SVDQ0231
C     SIGNIFICANT IN THE COMPUTED DERIVATIVES. (THIS IS NOT NECESSARY. SVDQ0232
C     BUT IS WISE IF THE ACCURACY REQUESTED IS DIFFICULT TO OBTAIN   SVDQ0233
C     BECAUSE THE DERIVATIVES HAVE SO FEW SIGNIFICANT DIGITS.)       SVDQ0234
C                                                                    SVDQ0235
      DATA P1,P01,P25,P3E1/.1,.01,.25,3./                            SVDQ0236
```

```
C     THE ABOVE DATA STATEMENT CONTAINS VARIOUS CONSTANTS                SVDQ0237
C     USED IN THE SUBROUTINE.                                            SVDQ0238
C                                                                        SVDQ0239
      DATA PT/1.,2.,4.,8.,16.,32.,64.,128.,256.,512.,1024./              SVDQ0240
C     PT(J)=2**(J-1), J=1,2,...,KQMAX+2                                  SVDQ0241
      EQUIVALENCE (PT(1),PTS1),(PT(2),PTS2),(PT(3),PTS3),(PT(4),PTS4),   SVDQ0242
     1  (PT(5),PTS5),(GAM(2,1),P5)                                       SVDQ0243
C                                                                        SVDQ0244
      DATA GAS/1.,-.5,-8.33333333E-02,-4.16666667E-02,                   SVDQ0245
     1 -2.63888889E-02,-1.875E-02,-1.42691799E-02,                       SVDQ0246
     2 -1.13673942E-02,-9.35653660E-03,                                  SVDQ0247
     3 -7.89255401E-03/                                                  SVDQ0248
C     GAS(I) GIVES THE I-TH ADAMS-MOULTON CORRECTOR                      SVDQ0249
C     COEFFICIENT, I=1,2,...,KQMAX+1.                                    SVDQ0250
C                                                                        SVDQ0251
      DATA GAM/1.,.5,.416666667,.375,.348611111,.329861111,              SVDQ0252
     1 .315591931 ,.304224537,.294868000,.286975446,                     SVDQ0253
     2 .5,.166666667,.125,.105555556,9.375E-02,                          SVDQ0254
     3 8.56150794E-02,7.95717593E-02,7.48522928E-02,                     SVDQ0255
     4 7.10329861E-02,6.78584998E-02,                                    SVDQ0256
     5 .166666667,4.16666667E-02,2.91666667E-02,                         SVDQ0257
     6 2.36111111E-02,2.03373016E-02,1.81299603E-02,                     SVDQ0258
     7 1.65181327E-02,1.52772266E-02,1.42851882E-02,                     SVDQ0259
     8 1.34693966E-02,                                                   SVDQ0260
     9 4.16666667E-02,8.33333333E-03,5.55555556E-03,                     SVDQ0261
     $ 4.36507937E-03,3.67890212E-03,3.22365520E-03,                     SVDQ0262
     $ 2.89544753E-03,2.64543584E-03,2.44737491E-03,                     SVDQ0263
     $ 2.28579544E-03/                                                   SVDQ0264
C     GAM(I,J) GIVES THE I-TH ADAMS-FALKNER PREDICTOR                    SVDQ0265
C     COEFFICIENT FOR INTEGRATING J-TH ORDER DIFFERENTIAL                SVDQ0266
C     EQUATIONS, I=1,2,...,KQMAX+1,  J=1,2,...,KMAXO.                    SVDQ0267
C                                                                        SVDQ0268
      DIMENSION ETA(8,8)                                                 SVDQ0269
      DATA (ETA(I, 1), I=1, 8)/  3.33333330E-01, 2.50000000E-01,         SVDQ0270
     1  1.13636360E-01, 6.73076930E-02, 4.60526330E-02, 3.43749980E-02,  SVDQ0271
     2  2.71381590E-02, 2.22547310E-02/                                  SVDQ0272
      DATA (ETA(I, 2), I=1, 8)/  2.00000000E-01, 4.00000000E-01,         SVDQ0273
     1  3.40909090E-01, 2.01923080E-01, 1.38157900E-01, 1.03124990E-01,  SVDQ0274
     2  8.14144780E-02, 6.67641930E-02/                                  SVDQ0275
      DATA (ETA(I, 3), I=1, 8)/  1.42857140E-01, 2.85714280E-01,         SVDQ0276
     1  3.42857140E-01, 3.46153840E-01, 2.45614040E-01, 1.87500000E-01,  SVDQ0277
     2  1.50303650E-01, 1.24626490E-01/                                  SVDQ0278
      DATA (ETA(I, 4), I=1, 8)/  1.11111110E-01, 2.22222220E-01,         SVDQ0279
     1  2.85714280E-01, 2.53968250E-01, 3.07017540E-01, 2.50000000E-01,  SVDQ0280
     2  2.08755060E-01, 1.78037850E-01/                                  SVDQ0281
      DATA (ETA(I, 5), I=1, 8)/  9.09090910E-02, 1.81818180E-01,         SVDQ0282
     1  2.42424240E-01, 2.42424240E-01, 1.73160170E-01, 2.50000000E-01,  SVDQ0283
     2  2.27732800E-01, 2.05428290E-01/                                  SVDQ0284
      DATA (ETA(I, 6), I=1, 8)/  7.69230760E-02, 1.53846150E-01,         SVDQ0285
     1  2.09790210E-01, 2.23776220E-01, 1.86480190E-01, 1.11888110E-01,  SVDQ0286
     2  1.91295550E-01, 1.91733070E-01/                                  SVDQ0287
      DATA (ETA(I, 7), I=1, 8)/  6.66666660E-02, 1.33333330E-01,         SVDQ0288
     1  1.84615380E-01, 2.05128200E-01, 1.86480190E-01, 1.34265730E-01,  SVDQ0289
     2  6.96192690E-02, 1.39442230E-01/                                  SVDQ0290
      DATA (ETA(I, 8), I=1, 8)/  5.88235290E-02, 1.17647060E-01,         SVDQ0291
     1  1.64705880E-01, 1.88235290E-01, 1.80995470E-01, 1.44796380E-01,  SVDQ0292
     2  9.21431500E-02, 4.21225830E-02/                                  SVDQ0293
C     ETA(I,J) I=1,2,...,J IS USED IN THE FIRST MODIFICATION OF THE      SVDQ0294
C     I-TH DIFFERENCE OF A J-TH ORDER METHOD AFTER THE STEPSIZE IS       SVDQ0295
```

```
C     HALVED.                                                          SVDQ0296
C     ETA(I,J) J=1,2,...,I-1 IS USED IN THE SECOND MODIFICATION OF     SVDQ0297
C     THE (J+1)-ST DIFFERENCE OF AN I-TH ORDER METHOD.                 SVDQ0298
C     THE TWO MODIFICATIONS OF THE DIFFERENCE TABLE AFTER HALVING THE  SVDQ0299
C     STEPSIZE REMOVES MOST OF THE INSTABILITY INHERENT IN THE METHOD  SVDQ0300
C     USED HERE FOR HALVING THE STEPSIZE.                              SVDQ0301
C                                                                      SVDQ0302
C                                                                      SVDQ0303
C     IF THE GSTOP FEATURE IS ELIMINATED, REMOVE THE FOLLOWING CARD.   SVDQ0304
      DATA LGSS,LGSD,LGSE/0,0,0/                                       SVDQ0305
C                                                                      SVDQ0306
C****
C     THE STATEMENTS BETWEEN THE LINE ABOVE AND A SIMILAR LINE BELOW
C     ARE INSERTED IN THE JPL VERSION OF THIS SUBROUTINE TO INSURE
C     THAT THE CODE IS COMPILED CORRECTLY. THESE STATEMENTS THEMSELVES
C     ARE NOT ACTUALLY COMPILED.
C
      NEQ=NEQ
      HMINA=HMINA
      HMAXA=HMAXA
      DELT=DELT
      TFINAL=TFINAL
      MXSTEP=MXSTEP
C****
C                                                                      SVDQ0307
C     INITIALIZE                                                       SVDQ0308
      KSTEP=-1                                                         SVDQ0309
      NE=NEQ                                                           SVDQ0310
      IF (NE.LE.0) GO TO 1190                                          SVDQ0311
      HH=H                                                             SVDQ0312
      NV=0                                                             SVDQ0313
      KDMAX=0                                                          SVDQ0314
      KDD=KD(1)                                                        SVDQ0315
      KDS=KDD                                                          SVDQ0316
      DO 10 I=1,NE                                                     SVDQ0317
      KQ(I)=1                                                          SVDQ0318
      DT(1,I)=0.                                                       SVDQ0319
      IF (KDS.LE.0) KDD=IABS(KD(I))                                    SVDQ0320
      IF ((KDD.EQ.0).OR.(KDD.GT.KMAXO)) HH=0.E0                        SVDQ0321
      IF (KDD.GT.KDMAX) KDMAX=KDD                                      SVDQ0322
   10 NV=NV+KDD                                                        SVDQ0323
C                                                                      SVDQ0324
      IF ((DELT*HH).LE.0.E0) GO TO 1190                                SVDQ0325
      ERRMX=P1                                                         SVDQ0326
      EMAX=ERND                                                        SVDQ0327
      RNDC=RND*P25                                                     SVDQ0328
      LDOUB=0                                                          SVDQ0329
      E2HFAC=P25                                                       SVDQ0330
      LSC=8                                                            SVDQ0331
      LSTC=4                                                           SVDQ0332
C     LSC AND LSTC ARE USED IN COMBINATION AS FOLLOWS                  SVDQ0333
C       LSTC=4,  LSC=4     FIRST TIME THROUGH THE FIRST STEP           SVDQ0334
C       LSTC=3,  LSC=4     SECOND TIME THROUGH THE FIRST STEP          SVDQ0335
C                          (NECESSARY TO CHECK STABILITY)              SVDQ0336
C       LSTC=2,  LSC=4     THIRD TIME THROUGH THE FIRST STEP           SVDQ0337
C                          (ONLY OCCURS IF INSTABILITY POSSIBLE)       SVDQ0338
C       LSTC=2,  LSC=2     SECOND STEP  (IF KQ(I)=2 , I=1,...,NEQ)     SVDQ0339
C       LSTC=1,  LSC=0     STARTING, ONE DERIVATIVE EVAL. PER STEP.    SVDQ0340
C       LSTC=1,  LSC.GT.0  SET WHEN STARTING TWO DERIV. EVAL. PER STEP SVDQ0341
```

```
C       LSTC=-1  LSC.LT.0  SET WHEN HALVING THE STEPSIZE                    SVDQ0342
C      IN THE LAST TWO CASES LSC IS SET EQUAL TO LSTC*(MAXIMUM KQ(I)        SVDQ0343
C      +1). AT THE END OF EACH STEP IF LSC.NE.0 IT IS REPLACED BY           SVDQ0344
C      LSC-LSTC UNTIL LSC=0, AT WHICH TIME LSTC IS SET EQUAL TO 0.          SVDQ0345
C       WHEN DOUBLING H, LSTC IS SET EQUAL TO -1 AND LSC TO -3.             SVDQ0346
C      UNDER CERTAIN CONDITIONS WHEN KQ(I)=1, LSTC IS SET =-1 AND LSC=-5    SVDQ0347
C                                                                           SVDQ0348
      KSOUT=MXSTEP                                                         SVDQ0349
      TOUT=T                                                               SVDQ0350
      IFL=13                                                               SVDQ0351
   20 IFLAG=1                                                              SVDQ0352
      GO TO 315                                                            SVDQ0353
C     END OF INITIALIZATION                                                SVDQ0354
C                                                                          SVDQ0355
C                                                                          SVDQ0356
      ENTRY SVDQ1                                                          SVDQ0357
C                                                                          SVDQ0358
C                                                                          SVDQ0359
C     TO OUTPUT VARIABLES IN THE CALLING SEQUENCE REMOVE THE C-S           SVDQ0360
C     IN COLUMN ONE OF THE FOLLOWING CARDS UNTIL REACHING THE COMMENT      SVDQ0361
C     END OF CODE FOR PRINTING VARIABLES IN CALLING SEQUENCE.              SVDQ0362
C     IF (NEQ.NE.0) GO TO 28                                               SVDQ0363
C     NEQ=1                                                                SVDQ0364
C  22 WRITE(6,5000) T,DELT,HMINA,HMAXA,KEMAX,EMAX,IFLAG,TFINAL,MXSTEP     SVDQ0365
C5000 FORMAT(3HOT=1PE15.8,7H  DELT=E12.5,8H  HMINA=,E10.3,8H  HMAXA=,      SVDQ0366
C    1  E10.3,8H  KEMAX=,I2,7H  EMAX=E10.3,8H  IFLAG=,I2/                  SVDQ0367
C    2  9H  I KQ KD,7X,4HF(I),9X,1HJ,8X,4HY(J),13X,5HYN(J),                SVDQ0368
C    3  10X,7HTFINAL=1E15.8,9H  MXSTEP=I4)                                 SVDQ0369
C     J=1                                                                  SVDQ0370
C     DO 24 I=1,NE                                                         SVDQ0371
C     IF (KDS.LT.0) KDD=IABS(KD(I))                                        SVDQ0372
C     K=KDD                                                                SVDQ0373
C     WRITE(6,5001) I,KQ(I),KDD,F(I),J,Y(J),YN(J)                          SVDQ0374
C5001 FORMAT(1H ,I2,2I3,1PE17.8,I4,2E17.8)                                 SVDQ0375
C  23 J=J+1                                                                SVDQ0376
C     K=K-1                                                                SVDQ0377
C     IF (K.EQ.0) GO TO 24                                                 SVDQ0378
C     WRITE(6,5002) J,Y(J),YN(J)                                           SVDQ0379
C5002 FORMAT(26X,I4,1P2E17.8)                                              SVDQ0380
C     GO TO 23                                                             SVDQ0381
C  24 CONTINUE                                                             SVDQ0382
C     WRITE(6,5003)                                                        SVDQ0383
C5003 FORMAT(3HO I,15X,16HDIFFERENCE TABLE)                                SVDQ0384
C     DO 27 I=1,NE                                                         SVDQ0385
C     KQQ=KQ(I)+1                                                          SVDQ0386
C     K=MIN0(KQQ,7)                                                        SVDQ0387
C     WRITE(6,5004) I,(DT(IO,I),IO=1,K)                                    SVDQ0388
C5004 FORMAT(1H ,I2,1PE19.8,6E16.7)                                        SVDQ0389
C                                                                          SVDQ0390
C     IF (K.EQ.KQQ) GO TO 27                                               SVDQ0391
C     K=K+1                                                                SVDQ0392
C     WRITE(6,5005) (DT(IO,I),IO=K,KQQ)                                    SVDQ0393
C5005 FORMAT(1H ,1PE21.5,7E14.5)                                           SVDQ0394
C  27 CONTINUE                                                             SVDQ0395
C     IF (NEQ.EQ.0) RETURN                                                 SVDQ0396
C     NEQ=0                                                                SVDQ0397
C  28 CONTINUE                                                             SVDQ0398
C     END OF CODE FOR PRINTING VARIABLES IN CALLING SEQUENCE.              SVDQ0399
C                                                                          SVDQ0400
```

```
      IF (2-IFL) 30,60,320                                              SVDQ0401
   30 IF (IFL.GT.5) GO TO 1180                                          SVDQ0402
C                                                                       SVDQ0403
C     SET STEP STOP                                                     SVDQ0404
      KSOUT=KSTEP+MXSTEP                                                SVDQ0405
      IF (IFL-4) 40,1210,210                                            SVDQ0406
C                                                                       SVDQ0407
C                                                                       SVDQ0408
C     SET PRINT STOP                                                    SVDQ0409
   40 TOUT=T+DELT                                                       SVDQ0410
C                                                                       SVDQ0411
   50 TPS1=ABS(AMOD((TOUT-TL)/HH,PTS2)-PTS1)                            SVDQ0412
      LFD=-1                                                            SVDQ0413
      IF (TPS1.GE.P5) LFD=1                                             SVDQ0414
C                                                                       SVDQ0415
C     LFD IS USED TO INDICATE WHETHER DOUBLING H IS PERMITTED.          SVDQ0416
C     IF LFD.LT.0 AT THE END OF A STEP THEN DOUBLING H IS               SVDQ0417
C     NOT PERMITTED. THE SIGN OF LFD IS CHANGED JUST BEFORE THE         SVDQ0418
C     END OF EACH STEP. IF DELT=H*(POWER OF 2) THEN                     SVDQ0419
C     OUTPUT VALUES WILL BE OBTAINED WITHOUT INTERPOLATION.             SVDQ0420
C                                                                       SVDQ0421
      GO TO 200                                                         SVDQ0422
C                                                                       SVDQ0423
C                                                                       SVDQ0424
C     ENTRY WITH IFLAG=2                                                SVDQ0425
C                                                                       SVDQ0426
C     UPDATE DIFFERENCE TABLE                                           SVDQ0427
C     AND COMPUTE KQM=MAXIMUM VALUE OF KQ(I), I=1,2,...,NEQ.            SVDQ0428
C                                                                       SVDQ0429
   60 KQM=0                                                             SVDQ0430
      DO 80 I=1,NE                                                      SVDQ0431
      KQQ=KQ(I)                                                         SVDQ0432
      IF (KQQ.GT.KQM) KQM=KQQ                                           SVDQ0433
      D(1)=F(I)                                                         SVDQ0434
      DO 70 K=1,KQQ                                                     SVDQ0435
      D(K+1)=D(K)-DT(K,I)                                               SVDQ0436
   70 DT(K,I)=D(K)                                                      SVDQ0437
      DT(KQQ+1,I)=D(KQQ+1)                                              SVDQ0438
   80 CONTINUE                                                          SVDQ0439
C     END OF UPDATING DIFFERENCE TABLE                                  SVDQ0440
C                                                                       SVDQ0441
C     STORE Y(J) IN YN(J)                                               SVDQ0442
      DO 90 J=1,NV                                                      SVDQ0443
   90 YN(J)=Y(J)                                                        SVDQ0444
C                                                                       SVDQ0445
      LFD=-LFD                                                          SVDQ0446
      TL=T                                                              SVDQ0447
      KSTEP=KSTEP+1                                                     SVDQ0448
C                                                                       SVDQ0449
C     IF THE GSTOP FEATURE IS ELIMINATED, REMOVE THE 2 FOLLOWING CARDS. SVDQ0450
      IF (LGSS) 1430,110,1510                                           SVDQ0451
  100 IFLAG=2                                                           SVDQ0452
  110 IF (LSC.EQ.0) GO TO 140                                           SVDQ0453
      LSC=LSC-LSTC                                                      SVDQ0454
      IF (LSC.EQ.0) GO TO 130                                           SVDQ0455
      IF (LSTC.NE.(-1)) GO TO 140                                       SVDQ0456
      IF (LDOUB.LT.0) RNDC=RND*P1                                       SVDQ0457
  120 E2HAVE=E2HMAX                                                     SVDQ0458
      TPS1=PTS1                                                         SVDQ0459
```

```
      GO TO 190                                                          SVDQ0460
  130 IF (ABS(HH).LT.HMINA) GO TO 1000                                   SVDQ0461
      LSTC=0                                                             SVDQ0462
  140 IF (LDOUB.NE.1) GO TO 150                                          SVDQ0463
      IF ((LFD.GT.0).AND.(ABS(HH+HH).LE.HMAXA)) GO TO 1030               SVDQ0464
      GO TO 200                                                          SVDQ0465
  150 RQMAX=PTS1/FLOAT(KQM+3)                                            SVDQ0466
      IF (LSTC.NE.0.OR.E2HAVE.EQ.0.E0)        GO TO 120                     FK
      TPS1=E2HMAX/E2HAVE                                                 SVDQ0468
      IF (TPS1-PTS1) 160,190,170                                         SVDQ0469
  160 E2HFAC=AMAX1(.075E0,E2HFAC-RQMAX,E2HFAC*TPS1)                      SVDQ0470
      GO TO 180                                                          SVDQ0471
  170 TPS1=TPS1*TPS1                                                     SVDQ0472
      E2HFAC=AMIN1(PTS1,E2HFAC*TPS1)                                     SVDQ0473
  180 RNDC=(1.1-E2HFAC)*RND                                              SVDQ0474
      E2HAVE=P5*(E2HMAX+E2HAVE)                                          SVDQ0475
  190 ERRMX=AMAX1(P1,ERRMX-RQMAX*TPS1)                                   SVDQ0476
C     E2HFAC IS A FACTOR WHICH IS TAKEN TIMES AN INITIAL ESTIMATE OF     SVDQ0477
C            E2H TO GET A FINAL VALUE OF E2H. (E2H=ESTIMATE OF WHAT      SVDQ0478
C            (ESTIMATED ERROR)/(REQUESTED ERROR) WOULD BE IF H WERE      SVDQ0479
C            DOUBLED.)                                                   SVDQ0480
C     E2HMAX IS THE MAXIMUM VALUE OF THE INITIAL ESTIMATE OF E2H OVER    SVDQ0481
C            ALL COMPONENTS WITH KQ(I).GT.1.                             SVDQ0482
C     E2HAVE IS A WEIGHTED AVERAGE OF PAST VALUES OF E2HMAX.             SVDQ0483
C     THE VALUE OF E2HFAC TENDS TO BE SMALLER WHEN E2HMAX IS             SVDQ0484
C     CONSISTANTLY SMALLER THAN E2HAVE.                                  SVDQ0485
C                                                                        SVDQ0486
C                                                                        SVDQ0487
C     CHECK FOR PRINT STOP AND FOR T REACHING TFINAL                     SVDQ0488
  200 TPD=(TOUT-TL)/HH                                                   SVDQ0489
      TPD1=(TFINAL-TL)/HH                                                SVDQ0490
C                                                                        SVDQ0491
C     IF THE GSTOP FEATURE IS ELIMINATED, REMOVE THE FOLLOWING CARD.     SVDQ0492
      IF (LGSE.LT.0) GO TO 1780                                          SVDQ0493
      IF (TPD1.LT.FAC(1)) GO TO 1220                                     SVDQ0494
      IF (TPD.LE.0.E0) GO TO 1280                                        SVDQ0495
C                                                                        SVDQ0496
C     CHECK FOR STEP STOP                                                SVDQ0497
      IF (KSOUT.GT.KSTEP) GO TO 210                                      SVDQ0498
C                                                                        SVDQ0499
      IFL=5                                                              SVDQ0500
      GO TO 310                                                          SVDQ0501
C                                                                        SVDQ0502
C     CHECK TO SEE IF ROUND-OFF ERROR IS PROMINENT                       SVDQ0503
  210 IF (EMAX.EQ.ERND) GO TO 220                                        SVDQ0504
C     IT IS                                                              SVDQ0505
      IFL=6                                                              SVDQ0506
      IF (EMAX.GE.P1) GO TO 310                                          SVDQ0507
      IF ((LSTC.GE.0).OR.(LDOUB.EQ.1)) ERRMX=PTS1                        SVDQ0508
C                                                                        SVDQ0509
  220 IFL=1                                                              SVDQ0510
  230 T=TL+HH                                                            SVDQ0511
C                                                                        SVDQ0512
C     START A NEW STEP                                                   SVDQ0513
C                                                                        SVDQ0514
C     PREDICT                                                            SVDQ0515
  240 J=0                                                                SVDQ0516
      DO 290 I=1,NE                                                      SVDQ0517
      IF (KDS.LE.0) KDD=IABS(KD(I))                                      SVDQ0518
```

```
      KDC=KDD                                                          SVDQ0519
  250 KQQ=KQ(I)                                                        SVDQ0520
      TPD=0.E0                                                         SVDQ0521
      K=KDC                                                            SVDQ0522
  260 TPD=TPD+DT(KQQ,I)*GAM(KQQ,KDC)                                   SVDQ0523
      KQQ=KQQ-1                                                        SVDQ0524
      IF (KQQ.GT.0) GO TO 260                                          SVDQ0525
  270 K=K-1                                                            SVDQ0526
      IF (K.LE.0) GO TO 280                                            SVDQ0527
      L=J+K                                                            SVDQ0528
      TPD=YN(L+1)*FAC(K)+HH*TPD                                        SVDQ0529
      GO TO 270                                                        SVDQ0530
  280 J=J+1                                                            SVDQ0531
      Y(J)=YN(J)+HH*TPD                                                SVDQ0532
      KDC=KDC-1                                                        SVDQ0533
      IF (KDC.GT.0) GO TO 250                                          SVDQ0534
  290 CONTINUE                                                         SVDQ0535
C     END OF PREDICT                                                   SVDQ0536
C                                                                      SVDQ0537
C     IF THE GSTOP FEATURE IS ELIMINATED, REMOVE THE C  IN COLUMN ONE  SVDQ0538
C     OF THE 2 FOLLOWING CARDS                                         SVDQ0539
C     IF (IFL) 20,320,300                                              SVDQ0540
C 300 CONTINUE                                                         SVDQ0541
C     AND THEN REMOVE THE 2 FOLLOWING CARDS.                           SVDQ0542
      IF (IFL) 1240,320,300                                            SVDQ0543
  300 IF (LGSD.NE.0) GO TO 1520                                        SVDQ0544
C                                                                      SVDQ0545
  310 IFLAG=IFL                                                        SVDQ0546
  315 CONTINUE                                                         SVDQ0547
C                                                                      SVDQ0548
C     TO OUTPUT VARIABLES IN THE CALLING SEQUENCE REMOVE THE C IN      SVDQ0549
C     COLUMN ONE OF THE FOLLOWING CARD.                                SVDQ0550
C     IF (NEQ.EQ.0) GO TO 22                                           SVDQ0551
C                                                                      SVDQ0552
      RETURN                                                           SVDQ0553
C                                                                      SVDQ0554
C                                                                      SVDQ0555
C     ENTRY WITH IFLAG=1                                               SVDQ0556
  320 EPS=EP(1)                                                        SVDQ0557
      ERND=0.                                                          SVDQ0558
      EMAX=0.                                                          SVDQ0559
      E2HMAX=0.                                                        SVDQ0560
      J=0                                                              SVDQ0561
      IF (LDOUB.GE.0) LDOUB=1                                          SVDQ0562
C                                                                      SVDQ0563
C     LDOUB IS SET IN THE LOOP BELOW AS FOLLOWS                        SVDQ0564
C     LDOUB=0   HALVE                                                  SVDQ0565
C     LDOUB=1   DOUBLE                                                 SVDQ0566
C     LDOUB=2   DO NOT DOUBLE                                          SVDQ0567
C                                                                      SVDQ0568
C     LDOUB.LT.0 AT THE BEGINNING OF THE LOOP INDICATES THE FOLLOWING  SVDQ0569
C       =-3  STEPSIZE HAS JUST BEEN HALVED. IF A DISCONTINUITY IS      SVDQ0570
C            NOT INDICATED MODIFY THE DIFFERENCE TABLE AND REPEAT      SVDQ0571
C            THE STEP.                                                 SVDQ0572
C       =-2  STEP AFTER LDOUB=-3. PROCEED AS USUAL (ORDER IS NOT       SVDQ0573
C            CHANGED)                                                  SVDQ0574
C       =-1  STEP AFTER LDOUB=-2. MODIFY THE DIFFERENCE TABLE ONCE     SVDQ0575
C            AGAIN AND REPEAT THE STEP.                                SVDQ0576
C     IF LDOUB IS SET EQUAL TO -4 THE ORDER IN AT LEAST ONE COMPONENT  SVDQ0577
```

```
C     HAS BEEN GREATLY REDUCED AND THE STEP IS REPEATED.                    SVDQ0578
C                                                                           SVDQ0579
C                                                                           SVDQ0580
C     IF THE OUTPUT OPTION IS ELIMINATED, REMOVE THE 4 FOLLOWING CARDS.     SVDQ0581
      IF (NEQ.LE.0) WRITE (6,5020) LSC,LFD,LSTC,KSTEP,E2HFAC,ERRMX,HH       SVDQ0582
 5020 FORMAT (19H0 I  KQQ LRND LDOUB,5X,1HE,9X,3HE2H,                       SVDQ0583
     1  8X,3HEPS,3X,4HLSC=,I3,6H  LFD=,I2,7H  LSTC=,I2,8H  KSTEP=,I4,       SVDQ0584
     2  9H  E2HFAC=,F4.2,8H  ERRMX=,F4.2,4H  H=,1PE9.2)                     SVDQ0585
C                                                                           SVDQ0586
C                                                                           SVDQ0587
C     BEGINNING OF LOOP FOR CORRECTING, ESTIMATING THE ERROR,               SVDQ0588
C     AND ADJUSTING THE NUMBER OF DIFFERENCES USED                          SVDQ0589
C                                                                           SVDQ0590
      DO 790 I=1,NE                                                         SVDQ0591
      IF (KDS.LE.0) KDD=IABS(KD(I))                                         SVDQ0592
      KQQ=KQ(I)                                                             SVDQ0593
C     KQQ GIVES THE ORDER OF THE PREDICTOR FORMULA AND KQQ+1 THE            SVDQ0594
C     ORDER OF THE CORRECTOR FORMULA.                                       SVDQ0595
C                                                                           SVDQ0596
      KQ1=KQQ+1                                                             SVDQ0597
      D(1)=F(I)                                                             SVDQ0598
C     FORM THE DIFFERENCE TABLE FROM PREDICTED DERIVATIVE VALUES.           SVDQ0599
      DO 330 K=1,KQ1                                                        SVDQ0600
      D(K+1)=D(K)-DT(K,I)                                                   SVDQ0601
  330 CONTINUE                                                              SVDQ0602
C     D(K) GIVES THE (K-1)-ST DIFFERENCE FORMED FROM PREDICTED              SVDQ0603
C     DERIVATIVE VALUES                                                     SVDQ0604
      TPS3=ABS(D(KQQ+1))                                                    SVDQ0605
      IF (LDOUB.LT.0) GO TO 720                                             SVDQ0606
C                                                                           SVDQ0607
  340 IF (KQQ.NE.1) GO TO 520                                               SVDQ0608
C                                                                           SVDQ0609
C     KQ(I)=1  IS TREATED AS A SPECIAL CASE                                 SVDQ0610
      E2H=PTS2                                                              SVDQ0611
      TPS5=DT(3,I)                                                          SVDQ0612
      IF (LSTC.LT.2) GO TO 370                                              SVDQ0613
C     FIRST STEP OF INTEGRATION                                             SVDQ0614
      IF (LSTC.NE.4) GO TO 350                                              SVDQ0615
      TPS4=0.                                                               SVDQ0616
      IF (KDD.GT.1) TPS3=AMAX1(TPS3,ABS(HH*D(1)))                           SVDQ0617
      TPS3=TPS3*P1                                                          SVDQ0618
  350 DT(2,I)=D(2)                                                          SVDQ0620
      D(2)=D(1)-DT(5,I)                                                     SVDQ0621
      TPS2=-D(2)                                                            SVDQ0622
      TPS3=PTS5*ABS(TPS2)                                                   SVDQ0623
C     FIRST STEP THAT KQ(I)=1                                               SVDQ0624
  360 DT(7,I)=PT(4)                                                         SVDQ0625
      IF (LSTC-2) 420,380,380                                               SVDQ0626
  370 IF (TPS5.EQ.0.) GO TO 360                                             SVDQ0627
      IF (DT(6,I).EQ.0.) GO TO 400                                          SVDQ0628
      TPS2=DT(5,I)-DT(1,I)                                                  SVDQ0629
  380 TPS4=DT(4,I)                                                          SVDQ0630
      TPS1=ABS(TPS4)                                                        SVDQ0631
      TPS4=TPS2*SIGN(PTS2,TPS4)-TPS5*TPS1                                   SVDQ0632
      IF (TPS4.GT.(-TPS1)) GO TO 410                                        SVDQ0633
  390 TPS6=-PTS1                                                            SVDQ0634
      GO TO 450                                                             SVDQ0635
C     FIRST STEP AFTER THE STEPSIZE HAS BEEN CHANGED                        SVDQ0636
  400 DT(6,I)=PT(1)                                                         SVDQ0637
```

```
      TPS6=0.                                                          SVDQ0638
      GO TO 450                                                        SVDQ0639
  410 IF (TPS4.LT.TPS1) GO TO 440                                      SVDQ0640
      IF (TPS1.EQ.0.) GO TO 390                                        SVDQ0641
  420 TPS6=PTS1                                                        SVDQ0642
      GO TO 450                                                        SVDQ0643
  430 KQ(I)=2                                                          SVDQ0644
      IF (2-LSTC) 510,510,520                                          SVDQ0645
  440 TPS6=TPS4/TPS1                                                   SVDQ0646
  450 TPS4=TPS5+TPS6                                                   SVDQ0647
      IF (TPS4.LT.P25) GO TO 430                                       SVDQ0648
C     INCREASE E2H IF (-S).GT..25                                      SVDQ0649
      E2H=PTS4*TPS4                                                    SVDQ0650
      IF (2-LSTC) 460,470,480                                          SVDQ0651
  460 LSC=0                                                            SVDQ0652
      GO TO 510                                                        SVDQ0653
  470 IF (TPS5-P25) 430,460,460                                        SVDQ0654
  480 IF (TPS4.GT.PTS2) GO TO 490                                      SVDQ0655
      IF (TPS4.GT.P5) D(2)=D(2)*GAM(2,1)                               SVDQ0656
      GO TO 510                                                        SVDQ0657
  490 IF (TPS4.LT.PTS4) GO TO 500                                      SVDQ0658
      TPS4=PTS4                                                        SVDQ0659
      D(2)=D(2)/PT(3)                                                  SVDQ0660
C     THE ESTIMATE OF E (AND HENCE OF E2H) IS INCREASED IF (-S).GE.8.  SVDQ0661
      TPS3=TPS3*DT(7,I)                                                SVDQ0662
      GO TO 510                                                        SVDQ0663
  500 D(2)=D(2)*PTS2*(TPS4-PTS1)/(TPS4*TPS4)                           SVDQ0664
      IF (TPS4.GE.P3E1) E2H=E2H*DT(7,I)                                SVDQ0665
C     STORE D(1)=PREDICTED DERIVATIVE  AND  D(2)=2*(CORRECTED Y -      SVDQ0666
C     PREDICTED Y)/H    D(1) AND D(2) ARE USED TO COMPUTE (-S) ON      SVDQ0667
C     THE NEXT STEP.                                                   SVDQ0668
  510 DT(5,I)=D(1)                                                     SVDQ0669
      DT(4,I)=D(2)                                                     SVDQ0670
      D(4)=TPS4                                                        SVDQ0671
C     STORE D(4)= CURRENT ESTIMATE OF (-S).  (-S).GT.3 IS AN INDICATION SVDQ0672
C     THAT THE STEPSIZE SHOULD BE LIMITED BECAUSE OF STABILITY PROBLEMS.SVDQ0673
C     S=H*(ESTIMATE OF EIGENVALUE OF F)=H*(DIFFERENCE BETWEEN PREDICTED SVDQ0674
C     AND CORRECTED DERIVATIVE VALUES)/(DIFFERENCE BETWEEN PREDICTED   SVDQ0675
C     AND CORRECTED INTEGRALS OF THE DERIVATIVE VALUES)                SVDQ0676
C     THE TREATMENT OF THE CASE KQ(I)=1 COULD BE IMPROVED BY USING A   SVDQ0677
C     SPECIAL METHOD FOR STIFF EQUATIONS WHEN (-S).GT.3  (MAYBE).      SVDQ0678
C     (THE ENTIRE TREATMENT OF THE CASE KQ(I)=1 IS FAR FROM IDEAL.)    SVDQ0679
      DT(3,I)=D(4)                                                     SVDQ0680
C                                                                      SVDQ0681
C     CORRECT                                                          SVDQ0682
  520 KDC=0                                                            SVDQ0683
      TPD=D(KQ1)                                                       SVDQ0684
      J=J+KDD                                                          SVDQ0685
      K=J                                                              SVDQ0686
  530 TPD=HH*TPD                                                       SVDQ0687
      KDC=KDC+1                                                        SVDQ0688
      Y(K)=Y(K)+GAM(KQQ+1,KDC)*TPD                                     SVDQ0689
      K=K-1                                                            SVDQ0690
      IF (KDC.LT.KDD) GO TO 530                                        SVDQ0691
C     END OF CORRECT                                                   SVDQ0692
C                                                                      SVDQ0693
      IF (EPS) 540,550,560                                             SVDQ0694
  540 EPS=EP(I)                                                        SVDQ0695
      IF (EPS.NE.0.) GO TO 560                                         SVDQ0696
```

```
  550 IF (HMAXA) 1190,780,1190                                          SVDQ0697
  560 TPS4=ABS(D(KQQ+2))                                                SVDQ0698
      TPS2=ABS(D(KQQ))                                                  SVDQ0699
      TPS6=HH/EPS                                                       SVDQ0700
C                                                                       SVDQ0701
      E=ABS(GAS(KQQ+1)*TPS3*TPS6)                                       SVDQ0702
C     E GIVES  ABS((ESTIMATED ERROR)/EPS)                               SVDQ0703
C                                                                       SVDQ0704
      LRND=1                                                            SVDQ0705
C                                                                       SVDQ0706
C     LRND= 1  MEANS NO ROUND-OFF ERROR                                 SVDQ0707
C         = 0  MEANS SOME ROUND-OFF ERROR                               SVDQ0708
C         =-1  MEANS EXTREME ROUND-OFF ERROR                            SVDQ0709
C                                                                       SVDQ0710
      FRND=PT(KQQ+2)*RNDC*ABS(D(1))                                     SVDQ0711
C     CHECK TO SEE IF ROUND OFF ERROR IS DOMINANT                       SVDQ0712
      IF ((TPS3+TPS4).GT.FRND) GO TO 570                                SVDQ0713
      LRND=0                                                            SVDQ0714
      IF ((PTS4*TPS2).LT.FRND) LRND=-1                                  SVDQ0715
C                                                                       SVDQ0716
  570 IF (E.LE.ERND) GO TO 590                                          SVDQ0717
      IF (E.LE.EMAX) GO TO 580                                          SVDQ0718
      EMAX=E                                                            SVDQ0719
      KEMAX=I                                                           SVDQ0720
  580 IF (LRND.LE.0) GO TO 590                                          SVDQ0721
      ERND=E                                                            SVDQ0722
      IF (ERND.GT.ERRMX) LDOUB=0                                        SVDQ0723
  590 IF (LDOUB.LE.0) GO TO 780                                         SVDQ0724
      TPS1=ABS(DD(KQQ))                                                 SVDQ0725
      TPS5=TPS1                                                         SVDQ0726
      IF (KQQ-2) 600,610,620                                            SVDQ0727
  600 E2H=E*E2H                                                         SVDQ0728
      IF (E2H.LT.P01) GO TO 780                                         SVDQ0729
      IF (D(4).LT.P3E1) GO TO 770                                       SVDQ0730
      LSTC=-1                                                           SVDQ0731
      LSC=-5                                                            SVDQ0732
      GO TO 770                                                         SVDQ0733
  610 TPS1=TPS2                                                         SVDQ0734
      IF (LSTC.NE.2) GO TO 620                                          SVDQ0735
      KQ(I)=3                                                           SVDQ0736
      TPS2=0.                                                           SVDQ0737
      TPS4=0.                                                           SVDQ0738
      LRND=0                                                            SVDQ0739
  620 E2H=TPS2+TPS3+TPS4                                                SVDQ0740
      E2H=ABS(GAS(KQQ-1)*PT(KQQ+1)*E2H*TPS6)                            SVDQ0741
C     E2H IS USED AS AN ESTIMATE OF WHAT THE VALUE OF E WOULD BE        SVDQ0742
C     IF H WERE DOUBLED. THE ESTIMATE IS CONSERVATIVELY LARGE.          SVDQ0743
      IF (E2H.GT.E2HMAX) E2HMAX=E2H                                     SVDQ0744
C                                                                       SVDQ0745
      IF (LRND) 630,640,660                                             SVDQ0746
C     EXTREME ROUND-OFF ERROR--REDUCE E2H                               SVDQ0747
  630 K=(KBIT2/KQQ)-4                                                   SVDQ0748
      IF (K.LE.3) GO TO 640                                             SVDQ0749
      IF (K.GT.KQMAX) K=KQMAX                                           SVDQ0750
      E2H=E2H/PT(K+1)                                                   SVDQ0751
      GO TO 650                                                         SVDQ0752
  640 E2H=AMIN1(E2H,E2H*P3E1*E2HFAC)                                    SVDQ0753
  650 E2H=E2H*P1                                                        SVDQ0754
      TPS6=PTS4                                                         SVDQ0755
```

```
      GO TO 670                                                          SVDQ0756
C                                                                        SVDQ0757
  660 E2H=E2H*E2HFAC                                                     SVDQ0758
      TPS6=FLOAT(KQQ+2)                                                  SVDQ0759
C     TEST TO SEE IF DIFFERENCES DECREASE MORE RAPIDLY THAN NECESSARY    SVDQ0760
C                                                                        SVDQ0761
  670 IF (TPS5.LT.(TPS3*TPS6)) GO TO 680                                 SVDQ0762
      IF (TPS2.LE.(TPS4*TPS6)) GO TO 760                                 SVDQ0763
C     THEY DO   INCREASE KQ(I)                                           SVDQ0764
      IF (KQQ.NE.KQMAX) KQ(I)=KQ1                                        SVDQ0765
      GO TO 760                                                          SVDQ0766
C                                                                        SVDQ0767
C     TEST TO SEE IF DIFFERENCES DECREASE TOO SLOWLY                     SVDQ0768
  680 TPS6=TPS6*P25                                                      SVDQ0769
      IF ((TPS1.GT.(TPS3*TPS6)).OR.(TPS2.GT.(TPS4*TPS6))) GO TO 760      SVDQ0770
C     THEY DO                                                            SVDQ0771
      IF (LSTC.LE.0) GO TO 750                                           SVDQ0772
      IF (E2H.LT.P01) GO TO 750                                          SVDQ0773
      IF (LSC-LSTC) 690,750,770                                          SVDQ0774
  690 IF (KSTEP-4) 750,700,710                                           SVDQ0775
  700 KQ1=LSTC                                                           SVDQ0776
  710 LSC=KQ1                                                            SVDQ0777
C     END OF ONE DERIVATIVE EVALUATION PER STEP                          SVDQ0778
      GO TO 770                                                          SVDQ0779
C                                                                        SVDQ0780
C     AFTER HALVING H. REDUCE KQ(I) IF A DISCONTINUITY HAS OCCURRED.     SVDQ0781
  720 IF (LDOUB.EQ.(-2)) GO TO 340                                       SVDQ0782
      DT(KQQ+1,I)=D(KQQ+1)                                               SVDQ0783
      IF (LDOUB.EQ.(-1)) DT(KQQ+1,I)=D(KQQ+2)                            SVDQ0784
      K=KQQ                                                              SVDQ0785
  730 IF (K.EQ.1) GO TO 740                                              SVDQ0786
      IF ((ABS(D(K-1)).GT.(PT(2)*ABS(D(K+1)))).OR.                       SVDQ0787
     1  (ABS(D(K)).GT.(PT(2)*ABS(D(K+2))))) GO TO 740                    SVDQ0788
      K=K-1                                                              SVDQ0789
      GO TO 730                                                          SVDQ0790
  740 IF ((K+K).GE.KQQ) GO TO 780                                        SVDQ0791
      LDOUB=-4                                                           SVDQ0792
      E2H=0.                                                             SVDQ0793
      KQQ=K+1                                                            SVDQ0794
C                                                                        SVDQ0795
C                                                                        SVDQ0796
C     DIFFERENCES DECREASE TOO SLOWLY REDUCE KQ(I).                      SVDQ0797
  750 KQ(I)=KQQ-1                                                        SVDQ0798
      IF (KQQ.EQ.2) DT(3,I)=0.                                           SVDQ0799
  760 IF (E2H.LT.P01) GO TO 780                                          SVDQ0800
  770 LDOUB=2                                                            SVDQ0801
  780 CONTINUE                                                           SVDQ0802
C                                                                        SVDQ0803
C     IF THE OUTPUT OPTION IS ELIMINATED, REMOVE THE 6 FOLLOWING CARDS.  SVDQ0804
      IF (NEQ.GT.0) GO TO 790                                            SVDQ0805
      IO2=MAXO(1,(KQQ-1))                                                SVDQ0806
      IO3=IO2+3                                                          SVDQ0807
      WRITE (6,5021) I,KQQ,LRND,LDOUB,E,E2H,EPS,                         SVDQ0808
     1  (IO1,D(IO1),IO1=IO2,IO3)                                         SVDQ0809
 5021 FORMAT (1H I2,I4,2I5,1PE13.3,2E11.3,4(3H  (,I2,1H),E10.3))        SVDQ0810
C                                                                        SVDQ0811
  790 CONTINUE                                                           SVDQ0812
C                                                                        SVDQ0813
C     END OF LOOP FOR CORRECTING, ESTIMATING THE ERROR, ETC.             SVDQ0814
```

```
C                                                                       SVDQ0815
C                                                                       SVDQ0816
C     IF THE INTERPOLATION CAPABILITY IS ELIMINATED REMOVE THE          SVDQ0817
C     FOLLOWING CARD.                                                   SVDQ0818
      IF (IFL.LT.0) GO TO 1250                                          SVDQ0819
C     TEST FOR HALVING H                                                SVDQ0820
      IF (LDOUB) 800,950,870                                            SVDQ0821
  800 LDOUB=LDOUB+1                                                     SVDQ0822
      IF (LDOUB+1) 810,870,820                                          SVDQ0823
  810 IF (LDOUB.EQ.(-2)) GO TO 820                                      SVDQ0824
C     ORDER IN AT LEAST ONE COMPONENT HAS BEEN GREATLY REDUCED          SVDQ0825
      LDOUB=0                                                           SVDQ0826
      GO TO 220                                                         SVDQ0827
  820 DO 860 I=1,NE                                                     SVDQ0828
      KQQ=KG(I)                                                         SVDQ0829
      TP=DT(KQQ+1,I)                                                    SVDQ0830
      IF (KQQ.LE.3) GO TO 860                                           SVDQ0831
      IF (LDOUB.NE.0) GO TO 840                                         SVDQ0832
      DO 830 K=3,KQQ                                                    SVDQ0833
C     SECOND MODIFICATION OF DIFFERENCE TABLE AFTER HALVING H           SVDQ0834
  830 DT(K,I)=DT(K,I)+ETA(KQQ-1,K-2)*TP                                 SVDQ0835
      GO TO 860                                                         SVDQ0836
  840 DO 850 K=2,KQQ                                                    SVDQ0837
C     FIRST MODIFICATION OF DIFFERENCE TABLE AFTER HALVING H            SVDQ0838
  850 DT(K,I)=DT(K,I)+ETA(K-1,KQQ-1)*TP                                 SVDQ0839
  860 CONTINUE                                                          SVDQ0840
      IFL=0                                                             SVDQ0841
      GO TO 240                                                         SVDQ0842
C                                                                       SVDQ0843
  870 IFL=2                                                             SVDQ0844
      IF (LSTC.LE.0) GO TO 300                                          SVDQ0845
      IF (2-LSTC) 880,900,940                                           SVDQ0846
  880 LSTC=LSTC-1                                                       SVDQ0847
      IF (LSTC.EQ.3) GO TO 890                                          SVDQ0848
      IF (LSC) 920,960,920                                              SVDQ0849
  890 IFL=1                                                             SVDQ0850
      GO TO 300                                                         SVDQ0851
  900 IF (LSC-2) 910,930,920                                            SVDQ0852
  910 LSTC=0                                                            SVDQ0853
  920 LDOUB=2                                                           SVDQ0854
      GO TO 60                                                          SVDQ0855
  930 LSTC=1                                                            SVDQ0856
      LSC=0                                                             SVDQ0857
      GO TO 60                                                          SVDQ0858
  940 IF (LSC) 300,60,300                                               SVDQ0859
C                                                                       SVDQ0860
C     HALVE H                                                           SVDQ0861
  950 HH=FAC(2)*HH                                                      SVDQ0862
      IF (LSTC.LT.2) GO TO 990                                          SVDQ0863
      ERND=P25*ERND                                                     SVDQ0864
C     IN LOOP TO FIND A NEW INITIAL STEPSIZE                            SVDQ0865
      IF (ERND.GE.P1) GO TO 950                                         SVDQ0866
      EMAX = ERND
      LSTC=4                                                            SVDQ0867
  960 LSC=4                                                             SVDQ0868
      DO 970 I=1,NE                                                     SVDQ0869
  970 KQ(I)=1                                                           SVDQ0870
      IF (LSTC-3) 890,890,1170                                          SVDQ0871
C                                                                       SVDQ0872
```

```
C     ENTRY AFTER IFLAG=7                                              SVDQ0873
  980 IF (LDOUB.EQ.0) GO TO 990                                        SVDQ0874
      LSC=1                                                            SVDQ0875
      LSTC=1                                                           SVDQ0876
      GO TO 140                                                        SVDQ0877
C     TEST TO SEE IF H IS TOO SMALL FOR HALVING                        SVDQ0878
  990 IF (ABS(HH).GE.HMINA) GO TO 1040                                 SVDQ0879
      IF (IFL.EQ.7) GO TO 1010                                         SVDQ0880
 1000 IFL=7                                                            SVDQ0881
      GO TO 1020                                                       SVDQ0882
C                                                                      SVDQ0883
 1010 HH=HH+HH                                                         SVDQ0884
      IFL=2                                                            SVDQ0885
 1020 H=HH                                                             SVDQ0886
      GO TO 310                                                        SVDQ0887
C                                                                      SVDQ0888
C                                                                      SVDQ0889
C     ERROR CRITERIA PERMIT DOUBLING                                   SVDQ0890
 1030 HH=HH+HH                                                         SVDQ0891
      IF (LSTC.EQ.1) GO TO 1050                                        SVDQ0892
      LSC=-3                                                           SVDQ0893
 1040 LSTC=-1                                                          SVDQ0894
C                                                                      SVDQ0895
C     CHANGE THE STEPSIZE                                              SVDQ0896
 1050 DO 1160 I=1,NE                                                   SVDQ0897
      KQQ=KQ(I)                                                        SVDQ0898
      IF (KQQ.NE.1) GO TO 1070                                         SVDQ0899
      DT(6,I)=0.                                                       SVDQ0900
      D(3)=DT(3,I)*PT(2)                                               SVDQ0901
      IF (D(3).GT.PT(3)) LSC=-6                                        SVDQ0902
      IF (LDOUB.NE.0) GO TO 1060                                       SVDQ0903
      KQM=8                                                            SVDQ0904
      IF (D(3).GE.PT(5)) DT(7,I)=DT(7,I)*PT(2)                         SVDQ0905
      D(3)=D(3)/PT(3)                                                  SVDQ0906
 1060 DT(3,I)=D(3)                                                     SVDQ0907
      GO TO 1160                                                       SVDQ0908
C                                                                      SVDQ0909
C     BEGINNING OF LOOP FOR CHANGING DIFFERENCE TABLE TO               SVDQ0910
C     CORRESPOND TO NEW VALUE OF H                                     SVDQ0911
 1070 DO 1080 K=1,KQQ                                                  SVDQ0912
      D(K)=DT(K,I)/PT(K)                                               SVDQ0913
 1080 IF (LDOUB.EQ.0) D(K)=D(K)/PT(K)                                  SVDQ0914
      KQQ2=KQQ-2                                                       SVDQ0915
      IF (KQQ2) 1160,1140,1090                                         SVDQ0916
 1090 DO 1130 J=1,KQQ2                                                 SVDQ0917
      IF (LDOUB.NE.0) GO TO 1110                                       SVDQ0918
C                                                                      SVDQ0919
C     HALVE                                                            SVDQ0920
      K=KQQ                                                            SVDQ0921
 1100 D(K-1)=D(K-1)+D(K)                                               SVDQ0922
      K=K-1                                                            SVDQ0923
      IF (K+J-KQQ) 1130,1130,1100                                      SVDQ0924
C                                                                      SVDQ0925
C     DOUBLE                                                           SVDQ0926
 1110 DO 1120 K=J,KQQ2                                                 SVDQ0927
 1120 D(K+1)=D(K+1)-D(K+2)                                             SVDQ0928
 1130 CONTINUE                                                         SVDQ0929
C                                                                      SVDQ0930
 1140 DO 1150 K=2,KQQ                                                  SVDQ0931
```

```
      IF (LDOUB.NE.0) D(K)=D(K)*PT(K)                                    SVDQ0932
      DT(K,I)=D(K)*PT(K)                                                 SVDQ0933
 1150 CONTINUE                                                           SVDQ0934
C     DIFFERENCE TABLE NOW CORRESPONDS TO NEW VALUE OF H                 SVDQ0935
C                                                                        SVDQ0936
 1160 CONTINUE                                                           SVDQ0937
 1170 H=HH                                                               SVDQ0938
      IF (LDOUB.NE.0) GO TO 50                                           SVDQ0939
      LFD=1                                                              SVDQ0940
      IF (LSTC.GE.0) GO TO 220                                           SVDQ0941
      LDOUB=-3                                                           SVDQ0942
      LSC=LSTC-KQM                                                       SVDQ0943
      GO TO 220                                                          SVDQ0944
C     END OF CHANGING STEPSIZE                                           SVDQ0945
C                                                                        SVDQ0946
C                                                                        SVDQ0947
C     IF THE GSTOP FEATURE IS ELIMINATED, REMOVE THE C  IN COLUMN ONE    SVDQ0948
C     OF THE 2 FOLLOWING CARDS                                           SVDQ0949
C1180 IF (7-IFL) 1181,980,220                                            SVDQ0950
C1181 IF (IFL-8) 60,1200,60                                              SVDQ0951
C     AND THEN REMOVE THE 2 FOLLOWING CARDS.                             SVDQ0952
 1180 K=IFL-5                                                            SVDQ0953
      GO TO (220,980,1200,1570,1570,1720,1720,60,1480,1450,1630,1570), KSVDQ0954
C                                                                        SVDQ0955
C     ILLEGAL VALUE OF PARAMETER   INTEGRATION CAN NOT PROCEED           SVDQ0956
 1190 IFL=8                                                              SVDQ0957
      GO TO 310                                                          SVDQ0958
 1200 WRITE (6,4000)                                                     SVDQ0959
 4000 FORMAT (26HOIFLAG=8 IN CALL TO DVDQ1.)                             SVDQ0960
      STOP                                                               SVDQ0961
C                                                                        SVDQ0962
C                                                                        SVDQ0963
 1210 IF (T-TFINAL) 200,1190,200                                         SVDQ0964
C                                                                        SVDQ0965
C     IF ONE DOES NOT WANT THE INTERPOLATION FEATURE, REMOVE ALL CARDS   SVDQ0966
C     BELOW THIS POINT (EXCEPT FOR THE END STATEMENT), AND ADD THE       SVDQ0967
C     FIVE FOLLOWING STATEMENTS.                                         SVDQ0968
C1220 IFL=4                                                              SVDQ0969
C     IF (TPD1.GT.TPD) GO TO 1280                                        SVDQ0970
C     GO TO 310                                                          SVDQ0971
C1280 IFL=3                                                              SVDQ0972
C     GO TO 310                                                          SVDQ0973
C                                                                        SVDQ0974
 1220 IFL=4                                                              SVDQ0975
      IF (KSTEP.NE.0) GO TO 1270                                         SVDQ0976
      TPD2=TPD                                                           SVDQ0977
C     ESTIMATE ERROR WHEN EXTRAPOLATION FROM INITIAL POINT IS REQUESTED  SVDQ0978
 1230 HH=HH*TPD1*.75E0                                                   SVDQ0979
C                                                                        SVDQ0980
C     IF THE GSTOP FEATURE IS ELIMINATED, REMOVE THE FOLLOWING CARD.     SVDQ0981
      IFLS=IFL                                                           SVDQ0982
      IFL=-1                                                             SVDQ0983
      GO TO 230                                                          SVDQ0984
C                                                                        SVDQ0985
C     IF THE GSTOP FEATURE IS ELIMINATED, REMOVE THE 4 FOLLOWING CARDS.  SVDQ0986
 1240 IF ((LGSD.EQ.0).OR.(IFLS.NE.4)) GO TO 20                           SVDQ0987
      LGSE=-1                                                            SVDQ0988
      TPD=FAC(1)                                                         SVDQ0989
      GO TO 1820                                                         SVDQ0990
```

```
 1250 HH=H                                                              SVDQ0991
      IF (EMAX.LT.P01) GO TO 1260                                       SVDQ0992
C     ERROR IS TOO LARGE, REDUCE H AND REPEAT THE FIRST STEP            SVDQ0993
      IF (TPD1.LT.C.E0) GO TO 1190                                      SVDQ0994
      LDOUB=1
      ERND=FAC(1)/TPD1
      ERND=ERND*ERND*P25
C     SET IFLAG SO INTERPOLATION IS DONE
      IFLAG = 3
      GO TO 950                                                         SVDQ0998
C                                                                       SVDQ0999
C     IF THE GSTOP FEATURE IS ELIMINATED, REMOVE THE C IN COLUMN ONE    SVDQ1000
C     OF THE FOLLOWING CARD                                             SVDQ1001
C1260 IFL=4                                                             SVDQ1002
C     AND THEN REMOVE THE 2 FOLLOWING CARDS.                            SVDQ1003
 1260 IFL=IFLS                                                          SVDQ1004
      IF (IFL.NE.4) GO TO 1790                                          SVDQ1005
      TPD=TPD2                                                          SVDQ1006
      IFLAG=3                                                           SVDQ1007
 1270 IF (TPD1.GT.TPD) GO TO 1280                                       SVDQ1008
      T=TFINAL                                                          SVDQ1009
      TPD=TPD1                                                          SVDQ1010
      GO TO 1290                                                        SVDQ1011
 1280 T=TOUT                                                            SVDQ1012
      IFL=3                                                             SVDQ1013
 1290 IF ((TPD.EQ.C.E0).AND.(IFLAG.LE.2)) GO TO 310                     SVDQ1014
C                                                                       SVDQ1015
C     INTERPOLATE FOR OUTPUT                                            SVDQ1016
 1300 TP=TPD                                                            SVDQ1017
      D(2)=TP                                                           SVDQ1018
      KQQ2=0                                                            SVDQ1019
      KDC=0                                                             SVDQ1020
      D(1)=PT(1)                                                        SVDQ1021
      DD(1)=PT(1)                                                       SVDQ1022
      DO 1310 K=2,KQM                                                   SVDQ1023
      DD(1)=DD(1)+PT(1)                                                 SVDQ1024
      TP=TP+PT(1)                                                       SVDQ1025
 1310 D(K+1)=(D(K)*TP)/DD(1)                                            SVDQ1026
      GO TO 1350                                                        SVDQ1027
C                                                                       SVDQ1028
C     COMPUTE THE INTERPOLATING INTEGRATION COEFFICIENTS                SVDQ1029
 1320 KQQ2=1                                                            SVDQ1030
      L=KQM-KDC                                                         SVDQ1031
      KDC=KDC+1                                                         SVDQ1032
 1330 IF (L.LE.0) GO TO 1350                                            SVDQ1033
      TP=0.                                                             SVDQ1034
      K=L                                                               SVDQ1035
      J=L+KDC                                                           SVDQ1036
 1340 JS=J-K                                                            SVDQ1037
      TP=TP+GAS(K)*D(JS+1)                                              SVDQ1038
      K=K-1                                                             SVDQ1039
      IF (K.GT.0) GO TO 1340                                            SVDQ1040
      D(J)=TP                                                           SVDQ1041
C                                                                       SVDQ1042
C     D(J) IS THE INTEGRATION COEFFICIENT FOR THE INTERPOLATION WHICH   SVDQ1043
C     CORRESPONDS TO GAM(J-KDC,KDC).                                    SVDQ1044
C                                                                       SVDQ1045
      L=L-1                                                             SVDQ1046
      GO TO 1330                                                        SVDQ1047
```

```
C     END OF COMPUTING INTEGRATION COEFFICIENTS                          SVDQ1048
C                                                                        SVDQ1049
C     PERFORM THE PARTIAL STEP INTEGRATION                               SVDQ1050
 1350 J=0                                                                SVDQ1051
      DO 1420 I=1,NE                                                     SVDQ1052
      IF (KDS.LE.0) KDD=IABS(KD(I))                                      SVDQ1053
      IF (KDC.GT.KDD) GO TO 1410                                         SVDQ1054
      TP=0.                                                              SVDQ1055
      KQQ=KQ(I)+KQQ2                                                     SVDQ1056
 1360 L=KQQ-KDC                                                          SVDQ1057
      IF (L.LE.0) GO TO 1370                                             SVDQ1058
      TP=TP+D(KQQ)*DT(L,I)                                               SVDQ1059
      KQQ=KQQ-1                                                          SVDQ1060
      IF (KQQ) 1390,1390,1360                                            SVDQ1061
 1370 K=J+KDD                                                            SVDQ1062
      L=KDC                                                              SVDQ1063
 1380 L=L-1                                                              SVDQ1064
      IF (L.EQ.0) GO TO 1400                                             SVDQ1065
      TP=TP*HH+YN(K)*FAC(L)*TPD**L                                       SVDQ1066
      K=K-1                                                              SVDQ1067
      GO TO 1380                                                         SVDQ1068
 1390 F(I)=TP                                                            SVDQ1069
      GO TO 1420                                                         SVDQ1070
 1400 Y(K)=YN(K)+HH*TP                                                   SVDQ1071
 1410 J=J+KDD                                                            SVDQ1072
 1420 CONTINUE                                                           SVDQ1073
      IF (KDC.NE.KDMAX) GO TO 1320                                       SVDQ1074
C     END OF PARTIAL STEP INTEGRATION                                    SVDQ1075
C                                                                        SVDQ1076
C     IF THE GSTOP FEATURE IS ELIMINATED, REMOVE THE C IN COLUMN ONE     SVDQ1077
C     OF THE FOLLOWING CARD                                              SVDQ1078
C     GO TO 310                                                          SVDQ1079
C     ALL STATEMENTS BELOW THIS POINT SHOULD THEN BE REMOVED (EXCEPT     SVDQ1080
C     FOR THE END STATEMENT)                                             SVDQ1081
      IF (LGSE) 1800,310,1810                                            SVDQ1082
C                                                                        SVDQ1083
C                                                                        SVDQ1084
C     SECTION FOR COMPUTING GSTOPS                                       SVDQ1085
C                                                                        SVDQ1086
      ENTRY SVDQG(NG,NSTOP,G,GT)                                         SVDQ1087
C                                                                        SVDQ1088
C     VARIABLES IN THE CALLING SEQUENCE HAVE THE FOLLOWING TYPES.        SVDQ1089
      INTEGER NG,NSTOP                                                   SVDQ1090
      REAL    G(1),GT(1)                                                 SVDQ1091
C                                                                        SVDQ1092
C     A GSTOP IS DEFINED AS A RETURN WHICH IS MADE TO THE USER WHEN A    SVDQ1093
C     USER SPECIFIED FUNCTION G PASSES THROUGH ZERO. THE USER MAY        SVDQ1094
C     SPECIFY ANY NUMBER OF FUNCTIONS G OF TWO TYPES. ZEROS OF THE FIRSTSVDQ1095
C     TYPE ARE LOCATED WITHOUT REQUIRING A DERIVATIVE EVALUATION         SVDQ1096
C     BEYOND THE ZERO. THIS TYPE OF GSTOP REQUIRES THAT G BE EVALUATED  SVDQ1097
C     BEFORE EACH DERIVATIVE EVALUATION. ZEROS OF THE SECOND TYPE ARE    SVDQ1098
C     LOCATED USING INTERPOLATION, WHICH IS MORE ACCURATE THAN THE       SVDQ1099
C     EXTRAPOLATION USED IN THE PRECEDING CASE AND ONLY REQUIRES ONE     SVDQ1100
C     EVALUATION OF G PER STEP. THUS ONE SHOULD USE THE SECOND TYPE OF   SVDQ1101
C     GSTOP IF POSSIBLE. USERS NOT USING THE GSTOP FEATURE NEED READ     SVDQ1102
C     NO FURTHER.                                                        SVDQ1103
C                                                                        SVDQ1104
C     SVDQG IS USED AS A SET UP CALL TO INDICATE A CHANGE IN THE NUMBER  SVDQ1105
C     OR TYPES OF GSTOPS. SVDQG SHOULD BE CALLED JUST BEFORE OR JUST     SVDQ1106
```

```
C       AFTER CALLING SVDQ IF                                                   SVDQ1107
C       1. ONE WANTS TO TEST FOR GSTOPS BEGINNING WITH THE FIRST STEP.          SVDQ1108
C       2. A JOB IS BEING RUN AFTER ANOTHER JOB THAT USES THE GSTOP             SVDQ1109
C          FEATURE. SVDQG MUST BE CALLED EVEN IF ALL THE VARIABLES IN           SVDQ1110
C          THE NEW JOB ARE THE SAME.                                            SVDQ1111
C       IN ADDITION SVDQG MAY BE CALLED AT ANY TIME IN THE INTEGRATION          SVDQ1112
C       TO CHANGE THE NUMBER OR TYPE OF GSTOPS.                                 SVDQ1113
C                                                                               SVDQ1114
C       THE USAGE OF THE VARIABLES IS GIVEN BELOW.                              SVDQ1115
C       NG=    THE NUMBER OF COMPONENTS IN G TO BE EXAMINED FOR A ZERO.         SVDQ1116
C              IF SVDQG IS CALLED AFTER THE FIRST STEP OF THE INTEGRATION,      SVDQ1117
C              THEN G IS EVALUATED FOR THE FIRST TIME AT THE END OF THE         SVDQ1118
C              NEXT STEP AND THUS A GSTOP IS NOT DETECTED IF G CHANGES          SVDQ1119
C              SIGN ON THE CURRENT STEP. IF IT IS IMPORTANT THAT G BE           SVDQ1120
C              EVALUATED IMMEDIATELY SET NG EQUAL TO THE NEGATIVE OF THE        SVDQ1121
C              NUMBER OF COMPONENTS TO BE TESTED FOR A ZERO. SETTING NG         SVDQ1122
C              LESS THAN ZERO WHEN CALLING SVDQG BEFORE THE FIRST STEP IS       SVDQ1123
C              NOT NECESSARY AND IS LIABLE TO BE DISASTEROUS. IF SVDQG IS       SVDQ1124
C              CALLED DURING THE INTEGRATION THE FOLLOWING STATEMENT SHOULDSVDQ1125
C              BE A GO TO (THE COMPUTED GO TO FOLLOWING THE CALL TO SVDQ1).SVDQ1126
C                                                                               SVDQ1127
C       NSTOP=THE NUMBER OF COMPONENTS OF G THAT MUST BE EXAMINED FOR           SVDQ1128
C              A ZERO BEFORE COMPUTING THE DERIVATIVES (FIRST TYPE OF           SVDQ1129
C              GSTOP). IF NSTOP.LT.0 OR NSTOP.GT.ABS(NG) IFLAG IS SET           SVDQ1130
C              EQUAL 8 AND AN IMMEDIATE RETURN IS MADE. IF NSTOP.GT.0,          SVDQ1131
C              G(1),G(2),...,G(NSTOP) ARE EXAMINED FOR A ZERO BEFORE EACH       SVDQ1132
C              DERIVATIVE EVALUATION, THE REMAINING COMPONENTS (IF ANY)         SVDQ1133
C              ARE EXAMINED AT THE END OF EACH STEP. WHEN A GSTOP IS FOUND      SVDQ1134
C              THE SUBROUTINE SETS NSTOP EQUAL TO THE INDEX OF THE              SVDQ1135
C              COMPONENT OF G RESPONSIBLE FOR THE STOP.                         SVDQ1136
C                                                                               SVDQ1137
C       G=     A VECTOR CONTAINING THE CURRENT VALUES OF THE FUNCTIONS          SVDQ1138
C              WHOSE ZEROS ARE TO BE DETERMINED.                                SVDQ1139
C                                                                               SVDQ1140
C       GT=    A VECTOR WITH THE SAME DIMENSION AS G USED BY THE                SVDQ1141
C              SUBROUTINE FOR TEMPORARY STORAGE.                                SVDQ1142
C                                                                               SVDQ1143
C       RETURNS FROM CALLING SVDQ1 WITH IFLAG.GT.8 SHOULD BE INTERPETED         SVDQ1144
C       AS FOLLOWS. (WE USE NSTOPI TO DENOTE THE INITIAL VALUE OF NSTOP.)       SVDQ1145
C       IFLAG                                                                   SVDQ1146
C       = 9 COMPUTE G(NSTOPI+1),...,G(ABS(NG)) (THE COMPONENTS OF G WITH        SVDQ1147
C           ZEROS TO BE LOCATED USING INTERPOLATION). THEN CALL SVDQ1.          SVDQ1148
C           NO RETURN IS MADE WITH IFLAG=9 IF NSTOPI=ABS(NG).                   SVDQ1149
C       =10 COMPUTE G(1),G(2),...,G(NSTOPI) (THE COMPONENTS OF G WITH           SVDQ1150
C           ZEROS TO BE LOCATED USING EXTRAPOLATION). THEN CALL SVDQ1.          SVDQ1151
C           NO RETURN IS MADE WITH IFLAG=10 IF NSTOPI=0.                        SVDQ1152
C       -11 G(NSTOP) IS APPROXIMATELY ZERO. IF THERE ARE NO                     SVDQ1153
C           DISCONTINUITIES SIMPLY CALL SVDQ1 TO CONTINUE THE INTEGRATION.SVDQ1154
C       =12 G(NSTOP) CHANGES SIGN, BUT THERE IS DIFFICULTY IN CONVERGING        SVDQ1155
C           TO A ZERO. THE USER MAY WISH TO MAKE A SPECIAL CHECK TO BE          SVDQ1156
C           CERTAIN THAT EVERYTHING IS ALL RIGHT. TO CONTINUE THE               SVDQ1157
C           INTEGRATION CALL SVDQ1.                                             SVDQ1158
C                                                                               SVDQ1159
      REAL     RG                                                               SVDQ1160
      DIMENSION GI(2),RG(3)                                                     SVDQ1161
C                                                                               SVDQ1162
C     INITIALIZE FOR GSTOPS                                                     SVDQ1163
      NGA=IABS(NG)                                                              SVDQ1164
      LGSS=-NGA                                                                 SVDQ1165
```

```
      LGSD=0                                                            SVDQ1166
      LGSE=0                                                            SVDQ1167
      IFLG=-20                                                          SVDQ1168
      IF (NG.GE.0) RETURN                                               SVDQ1169
      IFLG=-IFL                                                         SVDQ1170
 1430 LGSD=NSTOP                                                        SVDQ1171
      IF (LGSD) 1190,1450,1440                                          SVDQ1172
 1440 IFL=15                                                            SVDQ1173
      GO TO 1470                                                        SVDQ1174
C     ENTRY WITH IFL=15                                                 SVDQ1175
 1450 LGSS=0                                                            SVDQ1176
      IF (LGSD-NGA) 1460,1480,1190                                      SVDQ1177
 1460 LGSS=LGSD+1                                                       SVDQ1178
      IFL=14                                                            SVDQ1179
 1470 IFLAG=IFL-5                                                       SVDQ1180
      GO TO 315                                                         SVDQ1181
C     ENTRY WITH IFL=14                                                 SVDQ1182
 1480 DO 1490 I=1,NGA                                                   SVDQ1183
 1490 GT(I)=G(I)                                                        SVDQ1184
      GO TO 1730                                                        SVDQ1185
C     END OF INITIALIZATION FOR GSTOPS                                  SVDQ1186
C                                                                       SVDQ1187
C     ENTRY TO EVALUATE G AT THE END OF THE STEP                        SVDQ1188
 1500 LGSE=1                                                            SVDQ1189
 1510 IGK=LGSS                                                          SVDQ1190
      IFLG=0                                                            SVDQ1191
      IFL=9                                                             SVDQ1192
      GO TO 310                                                         SVDQ1193
C     ENTRY TO EVALUATE G BEFORE EVALUATING THE DERIVATIVES             SVDQ1194
 1520 IFLG=IFL                                                          SVDQ1195
      IFL=10                                                            SVDQ1196
 1530 IFLAG=10                                                          SVDQ1197
      IGKM=LGSD                                                         SVDQ1198
 1540 IGK=1                                                             SVDQ1199
 1550 GO TO 315                                                         SVDQ1200
 1560 IGK=IGK+1                                                         SVDQ1201
      IF (IGK.GT.IGKM) GO TO 1650                                       SVDQ1202
C     ENTRY WITH IFL=9,10, AND 17                                       SVDQ1203
C     TEST FOR G CHANGING SIGN                                          SVDQ1204
 1570 IF (G(IGK)*GT(IGK)) 1600,1580,1590                                SVDQ1205
 1580 IF (GT(IGK).NE.0.) GO TO 1600                                     SVDQ1206
      IF (TL.EQ.TG) GO TO 1560                                          SVDQ1207
 1590 IF (LGSE.GT.0) GT(IGK)=G(IGK)                                     SVDQ1208
      GO TO 1560                                                        SVDQ1209
C     G CHANGES SIGN -- PREPARE FOR ITERATION TO FIND ZERO              SVDQ1210
 1600 NSTOP=IGK                                                         SVDQ1211
      NSTOPI=IGK                                                        SVDQ1212
      IFLGS=IFL                                                         SVDQ1213
C     COMPUTE INITIAL VALUE FOR RG (=RATIO OF PARTIAL STEPSIZE WHERE    SVDQ1214
C     G IS KNOWN/THE INTEGRATION STEPSIZE)                              SVDQ1215
      IF (IFLG.EQ.0) GO TO 1610                                         SVDQ1216
      RG(3)=FAC(1)                                                      SVDQ1217
      RG(2)=0.E0                                                        SVDQ1218
      IF ((IFLG.EQ.2).AND.(IGK.LT.LGSS)) RG(2)=FAC(1)                   SVDQ1219
      GO TO 1620                                                        SVDQ1220
 1610 RG(3)=0.E0                                                        SVDQ1221
      RG(2)=-FAC(1)                                                     SVDQ1222
 1620 IF (LGSE.LT.0) RG(3)=IPD                                          SVDQ1223
      LGSE=-3                                                           SVDQ1224
```

```
      GI(2)=GT(IGK)                                                          SVDQ1225
      EPSGS=RND                                                              SVDQ1226
      IFL=16                                                                 SVDQ1227
      K=1                                                                    SVDQ1228
      GO TO 1640                                                             SVDQ1229
C     END OF PREPARATION TO BEGIN THE ITERATION                              SVDQ1230
C                                                                            SVDQ1231
C     ENTRY WITH IFL=16                                                      SVDQ1232
C     ITERATE TO FIND GSTOP                                                  SVDQ1233
 1630 K=1                                                                    SVDQ1234
      IF ((GI(2)*G(IGK)).GT.0.) K=2                                          SVDQ1235
      IF (ABS(GI(K)).GT.ABS(G(IGK))) GO TO 1640                              SVDQ1236
C     CONVERGENCE PROBLEMS                                                   SVDQ1237
      LGSE=LGSE-1                                                            SVDQ1238
      IF (LGSE.EQ.(-5)) EPSGS=PTS1                                           SVDQ1239
      EPSGS=EPSGS*PTS4                                                       SVDQ1240
 1640 GI(K)=G(IGK)                                                           SVDQ1241
      RG(K)=RG(3)                                                            SVDQ1242
C     SECANT ITERATION (GIVES NEW PARTIAL STEPSIZE/H)                        SVDQ1243
      TPD=RG(1)-(GI(1)*(RG(2)-RG(1)))/(GI(2)-GI(1))                          SVDQ1244
      T=TL+TPD*HH                                                            SVDQ1245
C     TEST FOR CONVERGENCE OF ITERATION                                      SVDQ1246
      IF (ABS(TPD-RG(3)).LE.EPSGS) GO TO 1560                                SVDQ1247
      RG(3)=TPD                                                              SVDQ1248
      GO TO 1300                                                             SVDQ1249
 1650 IF (10-IFL) 1660,1700,100                                              SVDQ1250
 1660 IF (IGKM.NE.NGA) GO TO 1710                                            SVDQ1251
      IF (LGSE.GT.(-3)) GO TO 1690                                           SVDQ1252
      IF (LSTC.NE.4) GO TO 1670                                              SVDQ1253
C     ESTIMATE ERROR -- GSTOP IS THE RESULT OF EXTRAPOLATING FROM            SVDQ1254
C     THE INITIAL POINT                                                      SVDQ1255
      TPD1=TPD                                                               SVDQ1256
      RG(3)=TPD                                                              SVDQ1257
      GO TO 1230                                                             SVDQ1258
 1670 IFL=11                                                                 SVDQ1259
      IF (LGSE.LT.(-4)) IFL=12                                               SVDQ1260
 1680 IFLAG=IFL                                                              SVDQ1261
C     TEST TO SEE IF GSTOP IS PRECEDED BY ANOTHER STOP                       SVDQ1262
      IF (((T-TOUT)*HH.LE.0.E0).AND.((T-TFINAL)*HH.LE.0.E0)) GO TO 1300      SVDQ1263
C     IT IS                                                                  SVDQ1264
      RG(3)=TPD                                                              SVDQ1265
      IFLS=IFL                                                               SVDQ1266
      GO TO 200                                                              SVDQ1267
 1690 LGSE=1                                                                 SVDQ1268
      IFL=IFLG                                                               SVDQ1269
      IF (IFL.LT.0) GO TO 20                                                 SVDQ1270
 1700 IGKM=NGA                                                               SVDQ1271
      IFL=IFLG                                                               SVDQ1272
      GO TO 310                                                              SVDQ1273
 1710 IFL=17                                                                 SVDQ1274
      IFLAG=9                                                                SVDQ1275
      IGKM=NGA                                                               SVDQ1276
      GO TO 315                                                              SVDQ1277
C     ENTRY WITH IFL=11 AND 12                                               SVDQ1278
C     SET PARAMETERS TO INDICATE A GSTOP HAS BEEN FOUND                      SVDQ1279
 1720 GT(NSTOPI)=0.                                                          SVDQ1280
 1730 LGSE=1                                                                 SVDQ1281
      IGKM=NGA                                                               SVDQ1282
      TG=TL                                                                  SVDQ1283
```

```
      IF (IFLG) 1740,1760,1770                                          SVDQ1284
 1740 IF (IFL.LT.13) GO TO 1750                                         SVDQ1285
      IF (IFLG.EQ.(-20)) GO TO 100                                      SVDQ1286
      IFL=-IFLG                                                         SVDQ1287
      GO TO 310                                                         SVDQ1288
 1750 HH=H                                                              SVDQ1289
      GO TO 200                                                         SVDQ1290
 1760 TPD=0.E0                                                          SVDQ1291
      T=TL                                                              SVDQ1292
      LGSE=-2                                                           SVDQ1293
      GO TO 1300                                                        SVDQ1294
 1770 IF (IFLG-3) 220,200,200                                           SVDQ1295
 1780 IF (LGSE.EQ.(-1)) GO TO 1790                                      SVDQ1296
      LGSE=-1                                                           SVDQ1297
      GO TO 1220                                                        SVDQ1298
 1790 TPD=RG(3)                                                         SVDQ1299
      T=TL+TPD*HH                                                       SVDQ1300
      IF (LGSE.NE.(-1)) GO TO 1670                                      SVDQ1301
      IFL=IFLS                                                          SVDQ1302
      LGSE=-3                                                           SVDQ1303
      GO TO 1680                                                        SVDQ1304
 1800 IF (LGSE+2) 1550,1500,310                                         SVDQ1305
 1810 IF (TPD.LE.0.E0) GO TO 310                                        SVDQ1306
      LGSE=-2                                                           SVDQ1307
 1820 IFLG=IFL                                                          SVDQ1308
      IFL=17                                                            SVDQ1309
      IFLAG=9                                                           SVDQ1310
      IF (LGSD) 1540,1540,1530                                          SVDQ1311
C     END OF SECTION FOR COMPUTING GSTOPS                               SVDQ1312
C                                                                       SVDQ1313
C                                                                       SVDQ1314
C     IN SOME APPLICATIONS, FOR EXAMPLE MULTIPLE QUADRATURE, MORE THAN  SVDQ1315
C     ONE INTEGRATION SUBROUTINE IS REQUIRED. THIS IS NOT NECESSARY IF  SVDQ1316
C     ALL OF THE VARIABLES ASSOCIATED WITH ONE INTEGRATION ARE SAVED    SVDQ1317
C     OUTSIDE OF THE INTEGRATOR WHILE DOING OTHER INTEGRATIONS, AND     SVDQ1318
C     THEN RESTORING THEM WHEN NEEDED. THESE VARIABLES CAN BE RESTORED  SVDQ1319
C     BY CALLING AN ENTRY WHICH CONTAINS ALL OF THE VARIABLES IN THE    SVDQ1320
C     CALLING SEQUENCE AND ALL OF THE VARIABLES THAT MUST BE SAVED      SVDQ1321
C     WHENEVER CHANGING TO A DIFFERENT INTEGRATION. (THIS ENTRY MUST    SVDQ1322
C     BE ADDED BY THE USER AND SHOULD BE FOLLOWED BY A RETURN STATEMENT.SVDQ1323
C     AFTER CALLING THIS ENTRY EITHER SVDQ OR SVDQ1 SHOULD BE CALLED    SVDQ1324
C     DEPENDING ON WHETHER THE INTEGRATION IS BEING STARTED OR NOT.)    SVDQ1325
C     THE VARIABLES WHICH MUST BE SAVED ARE-                            SVDQ1326
C     NE,NV,KDS,KDMAX,KSOUT,LDOUB,LFD,LSC,LSTC,IFL,IFLS,KQM   (INTEGERS)SVDQ1327
C     ERND,QDEC,ERRMX,E2HAVE,E2HFAC,E2HMAX,RNDC   (REAL)                SVDQ1328
C     HH,TOUT,TL   (REAL)                                               SVDQ1329
C     IF THE GSTOP FEATURE IS USED, THE EVALUATION OF G REQUIRES AN     SVDQ1330
C     INTEGRATION, AND THIS INTEGRATION MAY RESULT IN ANOTHER GSTOP,    SVDQ1331
C     THEN SOME ADDITIONAL VARIABLES MUST BE SAVED.                     SVDQ1332
C     IN MANY APPLICATIONS NE(=NEQ),NV(=SUM OF ORDERS OF THE            SVDQ1333
C     DIFFERENTIAL EQUATIONS),KDS(=KD),AND KDMAX(=MAXIMUM ORDER OF ANY  SVDQ1334
C     DIFFERENTIAL EQUATION) WILL BE THE SAME FOR EVERY INTEGRATION,    SVDQ1335
C     AND HENCE NEED NOT BE SAVED.                                      SVDQ1336
C                                                                       SVDQ1337
C     INSTRUCTIONS FOR MAKING CERTAIN CHANGES IN THIS SUBROUTINE ARE    SVDQ1338
C     GIVEN THROUGHOUT THE LISTING. TO FIND THESE INSTRUCTIONS, SEE     SVDQ1339
C     BELOW.                                                            SVDQ1340
C                                                                       SVDQ1341
C     TO ELIMINATE THE GSTOP CAPABILITY, SEE JUST BELOW CARDS SEQUENCED SVDQ1342
```

```
C     303,449,491,537,947,980,985,999, AND 1076.                        SVDQ1343
C     THIS MAKES THE SUBROUTINE SHORTER AND REDUCES OVERHEAD A LITTLE.  SVDQ1344
C                                                                       SVDQ1345
C     TO REMOVE THE INTERPOLATION CAPABILITY, SEE JUST BELOW CARDS      SVDQ1346
C     SEQUENCED 816 AND 965.                                            SVDQ1347
C     THE GSTOP FEATURE MUST ALSO BE ELIMINATED SINCE IT REQUIRES THE   SVDQ1348
C     INTERPOLATION CAPABILITY. IF OUTPUT POINTS ARE NOT HIT EXACTLY    SVDQ1349
C     (THEY ARE HIT EXACTLY IF HMAXA.LE.ABS(DELT), AND INITIAL H=       SVDQ1350
C     DELT*(2**(-K)), K=0,1,2...), THEN IFLAG=3 ON THE FIRST STEP THAT  SVDQ1351
C     (T-TOUT)*H.GT.0 (SEE THE USAGE OF DELT). IFLAG IS SET EQUAL TO 4  SVDQ1352
C     ON THE LAST STEP THAT (T-TFINAL)*H.LE.0.                          SVDQ1353
C                                                                       SVDQ1354
C     THE OUTPUT OPTION GIVES OUTPUT OF VARIABLES USED IN THE           SVDQ1355
C     INTEGRATION ON EVERY STEP THAT NEQ.LE.0. (WHICH OF COURSE MUST    SVDQ1356
C     BE SET AFTER THE INITIAL CALL TO THE INTEGRATOR)  TO ELIMINATE    SVDQ1357
C     THIS OPTION, SEE JUST BELOW CARDS SEQUENCED 580 AND 803.          SVDQ1358
C                                                                       SVDQ1359
C     THE CHECK OPTION WHEN ADDED TO THE OUTPUT OPTION OUTPUTS EVERY    SVDQ1360
C     VARIABLE IN THE CALLING SEQUENCE JUST AFTER ENTERING AND JUST     SVDQ1361
C     BEFORE LEAVING THE INTEGRATOR WHEN NEQ=0. THIS OUTPUT IS          SVDQ1362
C     SOMETIMES USEFUL IN DEBUGGING A PROGRAM. TO INCLUDE THIS OPTION   SVDQ1363
C     SEE JUST BELOW CARDS SEQUENCED 359 AND 548.                       SVDQ1364
C                                                                       SVDQ1365
      END                                                               SVDQ1366
```

A-5. DESCRIPTIONS OF PROGRAM LOGIC, SUBROUTINES AND USE OF TEXT EQUATIONS

A FORFLO chart of the overall program is shown in Figure A-1. The purpose of each subroutine was indicated by comment cards in the listing presented in Appendix A-4. A brief summary indicating the usage of the text equations is presented as follows.

DATAIN loads input and calculates various constants (in the nature of input) which are used throughout the program. For example, DATAIN calculates thermal diffusivity and dimensionless frequency from constituent inputs. DATAIN also equalizes all thermal properties to represent a homogeneous solid if IHOMO=1.

Function TAU defines the dimensionless temperature.

SLDFAZ solves Eq. (12) for wall temperature. If burn rate is specified, ITERA = 0, only one pass is made to calculate the wall temperature associated with that rate. If ITERA = 1, burn rate and wall temperature are solved by iteration in association with subroutine GASFAZ. SLDFAZ also calculates certain constants that are associated with Eqs. (4), (5) and (22) and are needed to calculate the thermal profiles.

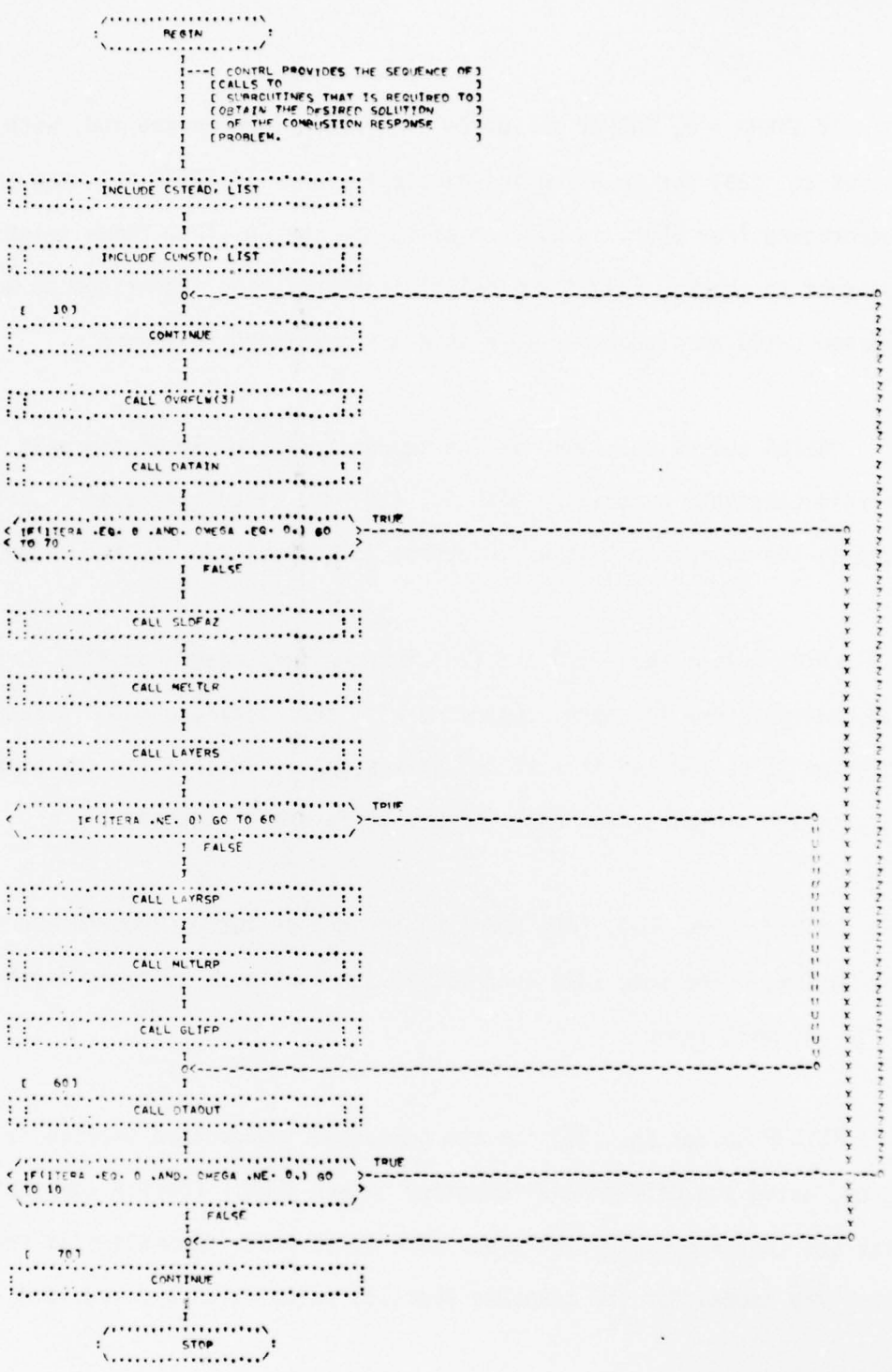


FIGURE A-1. FORFLO Chart of Overall Program

If ITERA = 0, GASFAZ solves Eq. (23) for flame height and, with AMULTI, solves Eq. (24) for an effective particle size. If ITERA = 1, the trial wall temperature from SLDFZ is used to calculate the Eq. (23) flame height. This is compared to the Eq. (24) flame height in AMULTI, and iterations on wall temperature proceed until the two agree to within a preassigned tolerance.

MELTLR solves Eq. (16) for the temperature profile in the melt layer. Certain constants associated with Eq. (16) are calculated also. Surface temperature is known, and melt layer thickness is determined from the known melting point.

LAYRS solves Eqs. (17) and (18) for the temperature profile in the succeeding layers in depth, beginning with the melting point. A satisfactory solution is tested in terms of the dimensionless temperature and gradient tending to zero at some adequate depth (number of layers traversed).

LAYRSP solves Eqs. (35) and (36) for the perturbed temperature profile in the layers, using Eqs. (34) and (38) to start at some adequate depth and working up to the melt layer.

MLTLRP solves Eq. (28) for the perturbed temperature profile in the melt layer, using the differential equation solver SVDQ. Certain constants associated with Eq. (28) are calculated also. The solid phase parameters at the mean surface which are needed for the response function calculation are provided.

GLIFP calculates the response function, Eq. (50), and the zero frequency limit, Eq. (53). Using Eqs. (29) - (31), the solid phase parameters at the mean surface are converted to values at the oscillating surface. Necessary gas phase parameters derived from Eq. (47) are calculated also.

DTAOUT prints the input and output.

Auxiliary JPL library routines which are used to solve the equations in the melt layer are subroutine PFIT, function SCPVAL and subroutine SVDQ. These were included in the listing to provide a self-contained program. PFIT and SCPVAL are called by MELTLR (steady-state solution); SVDQ and SCPVAL are called by MLTLRP (perturbed, complex solution). PFIT is a polynominal least squares curve fitting routine and SCPVAL evaluates the polynominal at a specified argument Subroutines BHSLR, AHLR and SL2NRM are orthogonal transformation subroutines which are called by PFIT to solve the least squares problem. SVDQ computes the numerical solution of the ordinary differential equation describing the perturbed melt layer.

A-6. SOLUTIONS TO SAMPLE PROBLEM

COMPOSITE SOLID PROPELLANT COMBUSTION RESPONSE MODEL

```
NDEG= 6      SIGFAC= .0000
P(1),...,P(NDEG+3)=      .19436020      .19436020      .84546567
                        -.17224529     -.01451589      .00289440
                        -.00031804     -.00000467      .00000915
```

TOLERANCE TESTS FOR A SATISFACTORY STEADY-STATE SOLUTION HAVE BEEN MET.

A SATISFACTORY STEADY-STATE SOLUTION HAS BEEN OBTAINED.

YABRV	TAPR	DTAPR	D2TAPR
IFLAG= 3 H= -.388720-02			
.388720+00	-.479468-16	.446635-16	.699864-16
IFLAG= 3 H= -.388720-02			
.349848+00	-.496136-16	.408895-16	.119785-15
IFLAG= 3 H= -.388720-02			
.310976+00	-.510967-16	.349742-16	.188050-15
IFLAG= 3 H= -.388720-02			
.272104+00	-.522927-16	.259677-16	.279606-15
IFLAG= 3 H= -.388720-02			
.233232+00	-.530627-16	.128803-16	.398577-15
IFLAG= 3 H= -.388720-02			
.194360+00	-.532267-16	-.539890-17	.546672-15
IFLAG= 3 H= -.388720-02			
.155488+00	-.525615-16	-.299537-16	.720195-15
IFLAG= 3 H= -.388720-02			
.116616+00	-.508064-16	-.615565-16	.906141-15
IFLAG= 3 H= -.388720-02			
.777441-01	-.476837-16	-.100234-15	.107872-14
IFLAG= 3 H= -.388720-02			
.388721-01	-.429374-16	-.144748-15	.119919-14
IFLAG= 3 H= -.388720-02			
.144355-07	-.363915-16	-.192204-15	.122273-14
IFLAG= 4 H= -.388720-02			
.000000	-.363915-16	-.192204-15	.122273-14

YABRV	TAPI	DTAPI	D2TAPI

```
 IFLAG= 3        H=    -.388720-02

      .388720+00     -.291908-16      .723704-16     -.256974-17
 IFLAG= 3        H=    -.388720-02

      .349848+00     -.319983-16      .718517-16      .317756-16
 IFLAG= 3        H=    -.388720-02

      .310976+00     -.347558-16      .696997-16      .820005-16
 IFLAG= 3        H=    -.388720-02

      .272104+00     -.373868-16      .652042-16      .153267-15
 IFLAG= 3        H=    -.388720-02

      .233232+00     -.397828-16      .574403-16      .251042-15
 IFLAG= 3        H=    -.388720-02

      .194360+00     -.417956-16      .452850-16      .379742-15
 IFLAG= 3        H=    -.388720-02

      .155488+00     -.432305-16      .275029-16      .540203-15
 IFLAG= 3        H=    -.388720-02

      .116616+00     -.438459-16      .295507-17      .725921-15
 IFLAG= 3        H=    -.388720-02

      .777441-01     -.433634-16     -.290331-16      .918866-15
 IFLAG= 3        H=    -.388720-02

      .388721-01     -.414955-16     -.681674-16      .108720-14
 IFLAG= 3        H=    -.388720-02

      .144356-07     -.379932-16     -.112688-15      .118857-14
 IFLAG= 4        H=    -.388720-02

      .000000        -.379932-16     -.112688-15      .118857-14
$TPOUT3
K2      =    ( .40739436E+01, -.11566977E+01)
V3      =      .43017409E+00
V5      =      .38842347E+01
V6A     =      .15725228E+01
V6B     =      .19052972E+01
V7      =      .24288042E+00
RF0     =      .42596531E+00
```

```
K2NM    =      .00000000E+00
V5NM    =      .00000000E+00
$END
$INPUT
CA      =      .32800000E+00
CB      =      .30000000E+00
CG      =      .30000000E+00
KA      =      .90000000E-03
KB      =      .44000000E-03
ROA     =      .19500000E+01
ROB     =      .90000000E+00
WAC     =      .33000000E+00,    .54000000E+00,    .00000000E+00,    .00000000E+00
SMLA    =      .45000000E-02,    .20000000E-01,    .00000000E+00,    .00000000E+00
E       =      .22000000E+05
R       =      .19862000E+01
PRXFAC  =      .13500000E+09
QS      =     -.20957000E+03
QB      =      .56900000E+03
QLM     =      .59570000E+02
QFLMHT  =      .24600000E+02
TFLM    =      .21000000E+04
TM      =      .83315000E+03
TZRO    =      .30000000E+03
PBAR    =      .34030000E+02
RBR     =      .12300000E+01
TOL     =      .10000000E-04
ITERA   =                +0
NSMAX   =               +50
NPP     =    ( .50000000E-02,   .00000000E+00)
TAPIN   =    ( .10000000E-29,   .00000000E+00)
YTD     =      .10000000E+02
OMEGA   =      .12566000E+04
NEQ     =                +1
KD      =                +2
MXSTEP  =              +100
EP      =      .10000000E-04
IHOMO   =                +0
$END
$OUTPT1
JMAX    =                +2
WA      =      .37000000E+00
DNM     =      .10033333E+03
WB      =      .13000001E+00
VFA     =      .75542691E+00
SMLAFP  =      .16329932E-01
DLTYTP  =      .14274320E+02
DLTYAB  =      .39335400E+01
SMLB    =      .65503794E-02
DLTYBB  =      .49440478E+01
KS      =      .90000000E-03
ROS     =      .16931982E+01
CS      =      .32436000E+00
KAPA    =      .14071294E-02
KAPB    =      .16296296E-02
KAPS    =      .16387300E-02
Z       =      .85867067E+00
ZPR     =      .20340954E+01
```

```
$END
$OUTPT2
THETAA  =      .10012712E+02
CHI     =      .72881025E+00
D       =      .72973667E+01
QSDMLS  =     -.68946471E+00
QBDMLS  =      .30582425E+00
QMDMLS  =      .19597945E+00
FDMLS   =     -.21616689E+00
B       =     -.86570440E+05
KMLTLR  =     -.53178746E+00
$END
$OUTPT3
TWBAR   =      .11062365E+04
YSSTBR  =      .94488705E+00
XSTBAR  =      .12588738E-02
AMM     =      .14158066E-02
TAUMLT  =      .99661283E+00,   .99322566E+00,   .98983848E+00,   .98645131E+00,
               .98306413E+00,   .97967695E+00,   .97628978E+00,   .97290260E+00,
               .96951543E+00,   .96612826E+00,   .96274108E+00,   .95935390E+00,
               .95596673E+00,   .95257955E+00,   .94919237E+00,   .94580520E+00,
               .94241802E+00,   .93903085E+00,   .93564367E+00,   .93225650E+00,
               .92886932E+00,   .92548215E+00,   .92209497E+00,   .91870780E+00,
               .91532062E+00,   .91193344E+00,   .90854627E+00,   .90515909E+00,
               .90177192E+00,   .89838474E+00,   .89499757E+00,   .89161039E+00,
               .88822322E+00,   .88483604E+00,   .88144886E+00,   .87806169E+00,
               .87467451E+00,   .87128734E+00,   .86790016E+00,   .86451299E+00,
               .86112581E+00,   .85773864E+00,   .85435146E+00,   .85096429E+00,
               .84757711E+00,   .84418993E+00,   .84080276E+00,   .83741558E+00,
               .83402841E+00,   .83064123E+00,   .82725406E+00,   .82386688E+00,
               .82047971E+00,   .81709253E+00,   .81370535E+00,   .81031818E+00,
               .80693100E+00,   .80354383E+00,   .80015665E+00,   .79676948E+00,
               .79338230E+00,   .78999513E+00,   .78660795E+00,   .78322078E+00,
               .77983360E+00,   .77644642E+00,   .77305925E+00,   .76967207E+00,
               .76628490E+00,   .76289772E+00,   .75951055E+00,   .75612337E+00,
               .75273620E+00,   .74934902E+00,   .74596184E+00,   .74257467E+00,
               .73918749E+00,   .73580032E+00,   .73241314E+00,   .72902597E+00,
               .72563879E+00,   .72225162E+00,   .71886444E+00,   .71547727E+00,
               .71209009E+00,   .70870291E+00,   .70531574E+00,   .70192856E+00,
               .69854139E+00,   .69515421E+00,   .69176704E+00,   .68837986E+00,
               .68499269E+00,   .68160551E+00,   .67821833E+00,   .67483116E+00,
               .67144398E+00,   .66805681E+00,   .66466963E+00,   .66128246E+00,
               .00000000E+00,   .00000000E+00,   .00000000E+00,   .00000000E+00,
               .00000000E+00,   .00000000E+00,   .00000000E+00,   .00000000E+00,
               .00000000E+00,   .00000000E+00
XMLT    =      .87110704E-05,   .16906565E-04,   .24678197E-04,   .32093999E-04,
               .39205886E-04,   .46054691E-04,   .52673046E-04,   .59087671E-04,
               .65320610E-04,   .71390445E-04,   .77312846E-04,   .83101363E-04,
               .88767592E-04,   .94321742E-04,   .99772671E-04,   .10512831E-03,
               .11039556E-03,   .11558078E-03,   .12068946E-03,   .12572670E-03,
               .13069701E-03,   .13560458E-03,   .14045313E-03,   .14524615E-03,
               .14998675E-03,   .15467788E-03,   .15932217E-03,   .16392212E-03,
               .16847998E-03,   .17299791E-03,   .17747784E-03,   .18192166E-03,
               .18633102E-03,   .19070757E-03,   .19505277E-03,   .19936806E-03,
               .20365472E-03,   .20791404E-03,   .21214713E-03,   .21635514E-03,
               .22053907E-03,   .22469994E-03,   .22883863E-03,   .23295607E-03,
               .23705304E-03,   .24113038E-03,   .24518879E-03,   .24922903E-03,
```

```
                .25325174E-03,   .25725760E-03,   .26124719E-03,   .26522113E-03,
                .26917995E-03,   .27312420E-03,   .27705438E-03,   .28097099E-03,
                .28487448E-03,   .28876531E-03,   .29264389E-03,   .29651067E-03,
                .30036598E-03,   .30421025E-03,   .30804382E-03,   .31186704E-03,
                .31568024E-03,   .31948374E-03,   .32327785E-03,   .32706287E-03,
                .33083907E-03,   .33460675E-03,   .33836614E-03,   .34211753E-03,
                .34586112E-03,   .34959721E-03,   .35332597E-03,   .35704764E-03,
                .36076244E-03,   .36447057E-03,   .36817221E-03,   .37186758E-03,
                .37555683E-03,   .37924018E-03,   .38291777E-03,   .38658978E-03,
                .39025635E-03,   .39391768E-03,   .39757387E-03,   .40122510E-03,
                .40487149E-03,   .40851320E-03,   .41215033E-03,   .41578304E-03,
                .41941143E-03,   .42303563E-03,   .42665576E-03,   .43027192E-03,
                .43388423E-03,   .43749280E-03,   .44109772E-03,   .44469911E-03,
                .00000000E+00,   .00000000E+00,   .00000000E+00,   .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,   .00000000E+00,
                .00000000E+00,   .00000000E+00
M               =          +101
N               =           +99
RBR             =  .12300000E+01
ITERA           =            +0
NS              =            +1
NSMAX           =           +50
LFLAG           =            +1
GWBRS           =  -.43017409E+00
DGWBRS          =  -.34504740E+01
QLDMLS          =  -.68946471E+00
$END
$OUTPT4
TAULR           =   .66128246E+00,   .93458967E-01,  -.95290865E-01,  -.15069028E-01,
                .15364397E-01,   .24296824E-02,  -.24773068E-02,  -.39175429E-03,
                .39943310E-03,   .63165220E-04,  -.64403327E-04,  -.10184560E-04,
                .10384188E-04,   .16421262E-05,  -.16743137E-05,  -.26477123E-06,
                .26996103E-06,   .42690874E-07,  -.43527661E-07,  -.68833413E-08,
                .70182622E-08,   .11098481E-08,  -.11316024E-08,  -.17894840E-09,
                .18245598E-09,   .28853073E-10,  -.29418624E-10,  -.46521781E-11,
                .47433658E-11,   .75010245E-12,  -.76480528E-12,  -.12094414E-12,
                .12331478E-12,   .19500651E-13,  -.19882885E-13,  -.31442231E-14,
                .32058533E-14,   .50696458E-15,  -.51690164E-15,  -.81741363E-16,
                .83343583E-16,   .13179719E-16,  -.13438055E-16,  -.21250560E-17,
                .21667095E-17,   .34263731E-18,  -.34935337E-18,  -.55245753E-19,
                .56328630E-19,   .89076502E-20,  -.90822499E-20,  -.14362413E-20,
                .14643932E-20,   .23157500E-21,  -.23611412E-21,  -.37338420E-22,
                .38070294E-22,   .60203286E-23,  -.61383336E-23,  -.97069871E-24,
                .98972545E-24,   .15651238E-24,  -.15958020E-24,  -.25235561E-25,
                .25730206E-25,   .40689019E-26,  -.41486568E-26,  -.65605685E-27,
                .66891628E-27,   .10578053E-27,  -.10785394E-27,  -.17055717E-28,
                .17390028E-28,   .27500098E-29,  -.28039130E-29,  -.44340287E-30,
                .45209405E-30,   .71492874E-31,  -.72894211E-31,  -.11527284E-31,
                .11753232E-31,   .18586228E-32,  -.18950538E-32,  -.29967845E-33,
                .30555248E-33,   .48319205E-34,  -.49266314E-34,  -.77908356E-35,
                .79435445E-35,   .12561697E-35,  -.12807920E-35,  -.20254082E-36,
                .20651084E-36,   .32657040E-37,  -.33770496E-37,  -.47727617E-38,
                .49354910E-38,   .00000000E+00,   .00000000E+00,   .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,   .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,   .00000000E+00,
                .00000000E+00,   .00000000E+00
DTAULR          =  -.73799413E+00,  -.35855919E-06,  -.13546264E-02,   .16016336E-02,
```

```
                .21841563E-03,  -.25824233E-03,  -.35216653E-04,  .41638174E-04,
                .56782192E-05,  -.67136070E-05,  -.91553829E-06,  .10824808E-05,
                .14761649E-06,  -.17453577E-06,  -.23801533E-07,  .28141596E-07,
                .38376839E-08,  -.45374615E-08,  -.61877614E-09,  .73160594E-09,
                .99769526E-10,  -.11796182E-09,  -.16086521E-10,  .19019791E-10,
                .25937395E-11,  -.30666908E-11,  -.41820627E-12,  .49446352E-12,
                .67430274E-13,  -.79725729E-13,  -.10872244E-13,  .12854724E-13,
                .17530058E-14,  -.20726551E-14,  -.28264916E-15,  .33418833E-15,
                .45573443E-16,  -.53883465E-16,  -.73481167E-17,  .86879996E-17,
                .11847873E-17,  -.14008256E-17,  -.19103137E-18,  .22586466E-18,
                .30801285E-19,  -.36417706E-19,  -.49663019E-20,  .58718749E-20,
                .80075041E-21,  -.94676254E-21,  -.12911055E-21,  .15265300E-21,
                .20817370E-22,  -.24613287E-22,  -.33565274E-23,  .39685683E-23,
                .54119566E-24,  -.63987940E-24,  -.87260687E-25,  .10317212E-24,
                .14069642E-25,  -.16635146E-25,  -.22685436E-26,  .26821983E-26,
                .36577262E-27,  -.43246916E-27,  -.58976067E-28,  .69729955E-28,
                .95091106E-29,  -.11243037E-28,  -.15332193E-29,  .18127917E-29,
                .24721135E-30,  -.29228879E-30,  -.39859579E-31,  .47127720E-31,
                .64268332E-32,  -.75987239E-32,  -.10362424E-32,  .12251942E-32,
                .16708045E-33,  -.19754642E-33,  -.26939507E-34,  .31851761E-34,
                .43436410E-35,  -.51356770E-35,  -.70035471E-36,  .82806031E-36,
                .11292304E-36,  -.13351382E-36,  -.18207351E-37,  .21527347E-37,
                .29356951E-38,  -.34710009E-38,   .00000000E+00,  .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,  .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,  .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,  .00000000E+00,
                .00000000E+00,   .00000000E+00
XLR         =   .44469911E-03,   .16774631E-01,   .23325010E-01,  .27625010E-01,
                .34375389E-01,   .38875389E-01,   .45425768E-01,  .49925768E-01,
                .56476147E-01,   .60976147E-01,   .67526526E-01,  .72026526E-01,
                .78576905E-01,   .83076905E-01,   .89627284E-01,  .94127283E-01,
                .10067766E+00,   .10517766E+00,   .11172804E+00,  .11622804E+00,
                .12277842E+00,   .12727842E+00,   .13382880E+00,  .13832880E+00,
                .14487918E+00,   .14937918E+00,   .15592956E+00,  .16042956E+00,
                .16697994E+00,   .17147993E+00,   .17803031E+00,  .18253031E+00,
                .18908069E+00,   .19358069E+00,   .20013107E+00,  .20463107E+00,
                .21118145E+00,   .21568145E+00,   .22223183E+00,  .22673183E+00,
                .23328221E+00,   .23778221E+00,   .24433259E+00,  .24883259E+00,
                .25538297E+00,   .25988296E+00,   .26643334E+00,  .27093334E+00,
                .27748372E+00,   .28198372E+00,   .28853410E+00,  .29303410E+00,
                .29958447E+00,   .30408447E+00,   .31063485E+00,  .31513485E+00,
                .32168523E+00,   .32618523E+00,   .33273561E+00,  .33723560E+00,
                .34378598E+00,   .34828598E+00,   .35483636E+00,  .35933636E+00,
                .36588674E+00,   .37038673E+00,   .37693711E+00,  .38143711E+00,
                .38798749E+00,   .39248749E+00,   .39903787E+00,  .40353787E+00,
                .41008824E+00,   .41458824E+00,   .42113862E+00,  .42563862E+00,
                .43218900E+00,   .43668900E+00,   .44323938E+00,  .44773937E+00,
                .45428975E+00,   .45878975E+00,   .46534013E+00,  .46984013E+00,
                .47639051E+00,   .48089051E+00,   .48744088E+00,  .49194088E+00,
                .49849126E+00,   .50299126E+00,   .50954164E+00,  .51404163E+00,
                .52059201E+00,   .52509201E+00,   .53164238E+00,  .53614238E+00,
                .54269276E+00,   .54719275E+00,   .55374313E+00,  .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,  .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,  .00000000E+00,
                .00000000E+00,   .00000000E+00
DSCRIM      =  -.76711670E-01,   .93458609E-01,  -.96645491E-01,  -.13467394E-01,
                .15582813E-01,   .21714400E-02,  -.25125234E-02,  -.35011611E-03,
```

```
                .40511132E-03,   .56451613E-04,  -.65318865E-04,  -.91020792E-05,
                .10531807E-04,   .14675904E-05,  -.16981152E-05,  -.23662963E-06,
                .27379872E-06,   .38153412E-07,  -.44146438E-07,  -.61517353E-08,
                .71180318E-08,   .99188632E-09,  -.11476889E-08,  -.15992861E-09,
                .18504972E-09,   .25786382E-10,  -.29836830E-10,  -.41577145E-11,
                .48107961E-11,   .67037672E-12,  -.77567752E-12,  -.10808942E-12,
                .12506779E-12,   .17427996E-13,  -.20165534E-13,  -.28100348E-14,
                .32514268E-14,   .45308111E-15,  -.52424975E-15,  -.73053363E-16,
                .84528370E-16,   .11778893E-16,  -.13629087E-16,  -.18991914E-17,
                .21975107E-17,   .30621960E-18,  -.35431967E-18,  -.49373878E-19,
                .57129381E-19,   .79608877E-20,  -.92113604E-20,  -.12835883E-20,
                .14852106E-20,   .20696171E-21,  -.23947065E-21,  -.33369851E-22,
                .38611489E-22,   .53804492E-23,  -.62255943E-23,  -.86752658E-24,
                .10037951E-23,   .13987724E-24,  -.16184874E-24,  -.22553363E-25,
                .26095978E-25,   .36364328E-26,  -.42076329E-26,  -.58632690E-27,
                .67842539E-27,   .94537491E-28,  -.10938716E-27,  -.15242925E-28,
                .17637239E-28,   .24577210E-29,  -.28437726E-29,  -.39627515E-30,
                .45852088E-30,   .63894150E-31,  -.73930454E-31,  -.10302090E-31,
                .11920312E-31,   .16610764E-32,  -.19219933E-32,  -.26782669E-33,
                .30989812E-33,   .43183528E-34,  -.49966669E-34,  -.69627753E-35,
                .80564675E-35,   .11226558E-35,  -.12989993E-35,  -.18101347E-36,
                .20944654E-36,   .29186039E-37,  -.33770496E-37,  -.47727617E-38,
                .49354910E-38,   .00000000E+00,   .00000000E+00,   .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,   .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,   .00000000E+00,
                .00000000E+00,   .00000000E+00
N               =                +99
XTD             =     .11419713E-01
CAPOA           =     .11687480E+01
CAPOB           =     .13535545E+01
CAPOS           =     .13611132E+01
$END
$OUTPTS
N2              =                +0
N2MXP1          =                +4
TAPIN           =   ( .10000000E-29,  .00000000E+00)
DTAPIN          =   (-.13500550E-29, -.68745435E-30)
XLR             =     .44469911E-03,   .16774631E-01,   .23325010E-01,   .27825010E-01,
                      .34375389E-01,   .38875389E-01,   .45425768E-01,   .49925768E-01,
                      .56476147E-01,   .60976147E-01,   .67526526E-01,   .72026526E-01,
                      .78576905E-01,   .83076905E-01,   .89627284E-01,   .94127283E-01,
                      .10067766E+00,   .10517766E+00,   .11172804E+00,   .11622804E+00,
                      .12277842E+00,   .12727842E+00,   .13382880E+00,   .13832880E+00,
                      .14487918E+00,   .14937918E+00,   .15592956E+00,   .16042956E+00,
                      .16697994E+00,   .17147993E+00,   .17803031E+00,   .18253031E+00,
                      .18908069E+00,   .19358069E+00,   .20013107E+00,   .20463107E+00,
                      .21118145E+00,   .21568145E+00,   .22223183E+00,   .22673183E+00,
                      .23328221E+00,   .23778221E+00,   .24433259E+00,   .24883259E+00,
                      .25538297E+00,   .25988296E+00,   .26643334E+00,   .27093334E+00,
                      .27748372E+00,   .28198372E+00,   .28853410E+00,   .29303410E+00,
                      .29958447E+00,   .30408447E+00,   .31063485E+00,   .31513485E+00,
                      .32168523E+00,   .32618523E+00,   .33273561E+00,   .33723560E+00,
                      .34378598E+00,   .34828598E+00,   .35483636E+00,   .35933636E+00,
                      .36588674E+00,   .37038673E+00,   .37693711E+00,   .38143711E+00,
                      .38798749E+00,   .39248749E+00,   .39903787E+00,   .40353787E+00,
                      .41008824E+00,   .41458824E+00,   .42113862E+00,   .42563862E+00,
                      .43218900E+00,   .43668900E+00,   .44323938E+00,   .44773937E+00,
```

```
                .45428975E+00,   .45878975E+00,   .46534013E+00,   .46984013E+00,
                .47639051E+00,   .48089051E+00,   .48744088E+00,   .49194088E+00,
                .49849126E+00,   .50299126E+00,   .50954164E+00,   .51404163E+00,
                .52059201E+00,   .52509201E+00,   .53164238E+00,   .53614238E+00,
                .54269276E+00,   .54719275E+00,   .55374313E+00,   .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,   .00000000E+00,
                .00000000E+00,   .00000000E+00,   .00000000E+00,   .00000000E+00,
                .00000000E+00,   .00000000E+00
TAPLR    =   (-.47946830E-16, -.29190791E-16), ( .38077123E-24,  .81146070E-26),
             (-.18338554E-27,  .85766757E-28), ( .10000000E-29,  .00000000E+00),
             ( .00000000E+00,  .00000000E+00), ( .00000000E+00
```

```
                ( .00000000E+00,  .00000000E+00), ( .00000000E+00,  .00000000E+00),
                ( .00000000E+00,  .00000000E+00), ( .00000000E+00,  .00000000E+00),
                ( .00000000E+00,  .00000000E+00), ( .00000000E+00,  .00000000E+00),
                ( .00000000E+00,  .00000000E+00), ( .00000000E+00,  .00000000E+00),
                ( .00000000E+00,  .00000000E+00), ( .00000000E+00,  .00000000E+00),
                ( .00000000E+00,  .00000000E+00), ( .00000000E+00,  .00000000E+00)
DTAPLR      =   ( .44663520E-16,  .72370430E-16), (-.52771847E-24, -.29710804E-24),
                ( .30654129E-27,  .10279349E-28), (-.13500550E-29, -.68745435E-30),
                ( .00000000E+00,  .00000000E+00), ( .00000000E+00,  .00000000E+00
```

```
                ( .00000000E+00,  .00000000E+00), ( .00000000E+00,  .00000000E+00),
                ( .00000000E+00,  .00000000E+00), ( .00000000E+00,  .00000000E+00),
                ( .00000000E+00,  .00000000E+00), ( .00000000E+00,  .00000000E+00),
                ( .00000000E+00,  .00000000E+00), ( .00000000E+00,  .00000000E+00)
TAUOPS    =   (-.36391481E-16, -.37993231E-16)
GOPPS     =   (-.19220352E-15, -.11268834E-15)
DGOPPS    =   ( .12227293E-14,  .11885721E-14)
RF        =   ( .47143511E+00,  .27978254E+00)
NRP       =   ( .23571755E-02,  .13989127E-02)
YSWP      =   ( .10277711E-02, -.17317998E-02)
XWP       =   ( .13693002E-05, -.23072783E-05)
NXSTP     =   (-.37305429E-02,  .32317242E-02)
XSTP      =   (-.46962826E-05,  .40683328E-05)
NYSSTP    =   (-.37305429E-02,  .32317242E-02)
YSSTP     =   (-.35249417E-02,  .30536143E-02)
RFNMLT    =   ( .00000000E+00,  .00000000E+00)
NRPNM     =   ( .00000000E+00,  .00000000E+00)
YSWPNM    =   ( .00000000E+00,  .00000000E+00)
XWPNM     =   ( .00000000E+00,  .00000000E+00)
NXSTPN    =   ( .00000000E+00,  .00000000E+00)
XSTPNM    =   ( .00000000E+00,  .00000000E+00)
YSSTPN    =   ( .00000000E+00,  .00000000E+00)
$END
$TPOUT1
TWBAR     =     .11062365E+04
QSDMLS    =    -.68946471E+00
QMDMLS    =     .19597945E+00
QBDMLS    =     .30582425E+00
THETAA    =     .10012712E+02
CHI       =     .72881025E+00
D         =     .72973667E+01
B         =    -.86570440E+05
KMLTLR    =    -.53178746E+00
QLDMLS    =    -.68946471E+00
FDMLS     =    -.21616689E+00
GWBRS     =    -.43017409E+00
DGWBRS    =    -.34504740E+01
$END
```