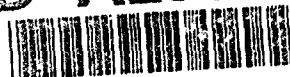
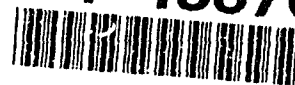


NAVY POSTGRADUATE SCHOOL
Monterey, California

AD-A280 395



94-18870



THESIS

Anode Sheath Contributions in Plasma Thrusters

by

John Forrest Riggs

March, 1994

Thesis Advisor:

Oscar Biblarz

Approved for public release; distribution is unlimited.

DTIC QUALITY INSPECTED 2

94 6 17 027

DTIC
ELECTE
JUN 20 1994
S B D

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 1994	3. REPORT TYPE AND DATES COVERED Engineer's and Master's Thesis	
4. TITLE AND SUBTITLE ANODE SHEATH CONTRIBUTIONS IN PLASMA THRUSTERS			5. FUNDING NUMBERS	
6. AUTHOR(S) Riggs, John Forrest				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE A	
13. ABSTRACT (maximum 200 words) Contributions of the anode to Magnetoplasmadynamic (MPD) thruster performance are considered. High energy losses at this electrode, surface erosion, and sheath/ionization effects must be controlled in designs of practical interest. Current constriction or spotting at the anode, evolving into localized surface damage and considerable throat erosion, is shown to be related to the electron temperature's (T_e) rise above the gas temperature (T_0). An elementary one-dimensional description of a collisional sheath which highlights the role of T_e is presented. Computations to model the one-dimensional sheath are attempted using a set of five coupled first-order, nonlinear differential equations describing the electric field, as well as the species current and number densities. For a large temperature nonequilibrium (i.e., $T_e \gg T_0$), the one-dimensional approach fails to give reasonable answers and a multidimensional description is deemed necessary. Thus, anode spotting may be precipitated by the elevation of T_e among other factors. A review of transpiration cooling as a means of recouping some anode power is included. Active anode cooling via transpiration cooling would result in (1) quenching T_e , (2) adding "hot" propellant to exhaust, and (3) reducing the local electron Hall parameter. However, significant technical problems remain.				
			15. NUMBER OF PAGES 106	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

NSN 7540-01-280-3500

Standard Form 298 (Rev. 2-89)

Prescribed by ANSI Std. Z39-18

Approved for public release; distribution is unlimited.

Anode Sheath Contributions in Plasma Thrusters

by

John F. Riggs

Lieutenant Commander, United States Navy

B.A., University of Kansas, 1982

**Submitted in partial fulfillment
of the requirements for the degrees of**

AERONAUTICAL & ASTRONAUTICAL ENGINEER

and

MASTER OF SCIENCE IN ASTRONAUTICAL ENGINEERING

from the


NAVAL POSTGRADUATE SCHOOL

March 1994


Author:


John F. Riggs

Approved by:


Oscar Biblarz, Thesis Advisor


Fred Schwirzke, Second Reader


D.J. Collins, Chairman

Department of Aeronautical Engineering


Richard S. Elster, Dean of Instruction

ABSTRACT

Contributions of the anode to Magnetoplasmadynamic (MPD) thruster performance are considered. High energy losses at this electrode, surface erosion, and sheath/ionization effects must be controlled in designs of practical interest. Current constriction or spotting at the anode, evolving into localized surface damage and considerable throat erosion, is shown to be related to the electron temperature's (T_e) rise above the gas temperature (T_0). An elementary one-dimensional description of a collisional sheath which highlights the role of T_e is presented. Computations to model the one-dimensional sheath are attempted using a set of five coupled first-order, nonlinear differential equations describing the electric field, as well as the species current and number densities. For a large temperature nonequilibrium (i.e., $T_e \gg T_0$), the one-dimensional approach fails to give reasonable answers and a multidimensional description is deemed necessary. Thus, anode spotting may be precipitated by the elevation of T_e among other factors. A review of transpiration cooling as a means of recouping some anode power is included. Active anode cooling via transpiration cooling would result in (1) quenching T_e , (2) adding "hot" propellant to exhaust, and (3) reducing the local electron Hall parameter. However, significant technical problems remain.

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/ _____	
Availability Codes	
Dist	Avail and/or Special
A-1	

TABLE OF CONTENTS

I. INTRODUCTION	1
II. LITERATURE REVIEW	5
III. ANODE DESCRIPTION	7
A. THRUSTER GEOMETRY DESCRIPTION	7
B. ELEMENTARY SHEATH FORMULAE DESCRIPTION	11
1. Discussion	11
2. Simplified Formulation	14
3. Approximate Formulation	21
a. Effects of Temperature on Anode Constriction	22
(1) Case I: $T_e = T_i = T_o$ (Equilibrium)	23
(2) Case II: $T_e \gg T_i = T_o$ (Two-Temperature)	25
b. Similarity to Vacuum Arc Phenomena	27
C. COMPUTER CODE	29
D. COMPUTATIONAL RESULTS	30
E. ANODE FALL VOLTAGE	36
IV. TRANSPIRATION COOLING	38
V. CONCLUSIONS AND RECOMMENDATIONS	43

APPENDIX - APPLICABLE FORTRAN PROGRAMS	46
LIST OF REFERENCES	92
INITIAL DISTRIBUTION LIST	96

LIST OF TABLES

I. NOMENCLATURE	15
-----------------------	----

LIST OF FIGURES

1. Magnetoplasmadynamic (MPD) Thruster	8
2. Space Plasma Thruster	10
3. Electric Field Between Two Electrodes	11
4. Electric Field and First Two Space Derivatives	19
5. Electric Field and Species Approximation	24
6. Two-dimensional Model of Current Paths	26
7. Anode Discharge Modes	28
8. Ionization Coefficient ν as a Function of E/N	30
9. Species Number Density Plots for Individual Computer Run	32
10. Electric Field as a Function of \bar{y}	34
11. Species Number Density as a Function of \bar{y}	35

ACKNOWLEDGMENT

I dedicate this work to my dear wife Lin, for her support and understanding during my absence as a geographical bachelor of almost three years.

My sincere thanks to my parents for encouraging me to pursue my ambitions from childhood on, and to Oscar Biblarz, Ph.D., for patiently helping me to understand plasma physics.

I. INTRODUCTION

Several types of space flight propulsion systems have been developed over the years. These include chemical, nuclear, electric and solar propulsion. The majority of space thrusters to date have been chemical rockets. Although the Chinese used rockets over 800 years ago, true development of rocket propulsion took place during this century [Ref. 1]. Chemical thrusters give high thrust-to-weight ratios, larger than unity, and have been fully developed in the form of space launch vehicles and attitude control thrusters. In contrast, other propulsion systems have been developed only to the proof-of-concept stage, and essentially remain at this stage of development. Nuclear propulsion was studied with the NERVA (Nuclear Engine for Rocket Vehicle Application) thruster in the 1960's, and abandoned [Ref. 2:pp. 517-519]. Electric propulsion flights during the 1960's included the U.S. SERT-1 (Space Electric Rocket Test) in 1964 and the U.S.S.R. Yantari-1 rocket in 1966. Solar-electric propulsion was demonstrated via the SERT-2 rocket in 1970, powering the electric thruster from power generated by solar cells. Further electric propulsion research flights in the 1980's included the U.S. Navy's NOVA-1 satellite in 1981, and Japan's MS-T4 satellite, launched from the Space Shuttle. Beyond this, nonchemical thrusters have only been used in auxiliary roles, such as station-keeping and attitude

control on geosynchronous satellites. NASA's Project PATHFINDER in the mid-1980's proposed the use of a megawatt-level electric plasma thruster for a manned Mars mission. However, development of this project was never funded.

In comparing the different propulsion schemes, a primary performance indicator is specific impulse, defined as the ratio of thrust to the rate of propellant usage, or alternately, propellant effective exhaust velocity (u_e), divided by the sea-level gravitational constant, (g_0).

$$I_{sp} = \frac{\dot{m} u_e}{\dot{m} g_0} = \frac{u_e}{g_0} \quad \text{sec} \quad (1)$$

Chemical rockets are inherently limited in performance by the total energy available in the fuel/oxidizer combustion process, so that the total enthalpy available for conversion into exhaust kinetic energy is limited. Exhaust velocity is also limited by material heating limitations of the combustion chamber and nozzle throat, and "frozen flow Losses" (unrecoverable energy deposition in internal modes of the gas) [Ref. 3:pp. 4-5]. Peak specific impulse for liquid chemical propellants is presently on the order of 450 seconds. This capability is completely sufficient for the tasks of launch to low earth orbit (LEO), attitude control, station keeping, and such missions. However, for missions such as manned interplanetary exploration, chemical propulsion can be shown to be clearly inadequate. A comparative analysis of a Mars

mission using chemical and electric propulsion systems shows the large difference in mass payload ratio (final mass/initial launch mass) for the two systems. A chemical system using a high impulse Hohman ellipse trajectory delivers a maximum of approximately 10% to 18% of the launch mass to the Martian surface [Ref. 4:p. 115]. In comparison, an electric system using a low impulse spiral trajectory could deliver from 20% to 60% of the launch mass, depending on the desired transit time. Each mission assumes transit from low Earth orbit to Mars orbit. An electric propulsion system would still need a high thrust propulsion system to reach the Martian surface [Ref. 5:pp. 344-346]. The large difference in payload ratio is due to the much larger exhaust velocity and more efficient use of fuel by electric propulsion. Thus, some form of electric or hybrid electric thruster would seem to be in order for such interplanetary missions. However, due to the low thrust-to-weight ratio of electric thrusters, they must be launched into orbit by other means. Their usefulness is restricted to space thrusters, not to launch systems.

With specific impulses of as high as 10,000 seconds, electric propulsion offers the performance envelope needed for manned interplanetary missions. Electric propulsion is divided into three types of thrusters: electrostatic, electrothermal, and electromagnetic. The type relevant to this work is the magnetoplasmadynamic (MPD) thruster, an electromagnetic propulsion system that utilizes the Lorentz force created by an electric current together with its induced magnetic field to propel a gas that has been heated to the plasma state. According to electromagnetic theory, a conductor carrying a current produces an induced magnetic force perpendicular to

the current. The applied electric field and its induced magnetic field interact to produce the Lorentz force ($\vec{F} = \vec{j} \times \vec{B}$) perpendicular to both fields on the conductors. This briefly summarizes the concept behind the "self-field" MPD accelerator [Ref. 2:pp. 485-486]. MPD performance is enhanced by adding magnetic coils to the thruster, thus strengthening the magnetic field and, as a consequence, the Lorentz force and thrust. This thruster is appropriately called an "applied-field" MPD thruster. MPD thrusters have shown specific impulses of up to 7,000 seconds and efficiencies as high as 70% [Ref. 6:pp. 2-3]. Performance of MPD thrusters is limited by several factors, including electrode erosion, current spotting, frozen flow losses, and electrode power deposition. Specifically, anode power deposition is the single largest power loss mechanism in MPD thrusters operating at submegawatt power levels [Ref. 7]. In the following work, we review and analytically model the MPD anode, including the sheath and anode potential drop.

II. LITERATURE REVIEW

Anode losses significantly limit magnetoplasmadynamic (MPD) thruster performance. Much effort has been placed on characterizing these losses and on the nature of power deposition in the anode [Refs. 8-14]. As much as 80% of thruster total power may end up being deposited in the anode at sub-megawatt power levels [Refs. 8,15]. This power deposition together with current constriction at the anode surface present serious problems to thruster cooling and performance, as well as to anode lifetime. Before any practical design can be achieved, a more thorough understanding of the phenomena at the anode, particularly the anode sheath, must be gained. Studies have shown that the anode power fraction depends on thruster power, current, mass flow rate, and the parameter J^2/\dot{m} [Refs. 8,12,13,16]. It has also been shown that the anode fall voltage is inversely proportional to anode current density [Refs. 13,16]. It is believed that a better understanding of the role of an elevated electron temperature, of current flow dimensionality, and of current unsteadiness are prerequisites for the evolution of any practical MPD thruster design.

Computer codes that accurately describe observed data from steady-state MPD thrusters have been developed [Refs. 17-19]. However, these codes do not adequately describe observed data from quasi-steady thruster experiments. It has been suggested that the lack of proper electrode modelling (i.e., sheaths and fall potentials) in these

codes may explain this discrepancy [Ref. 6]. Limited analytical work has been done in modelling the sheath and ambipolar regions at the anode, influenced perhaps by the difficult set of coupled, nonlinear partial differential equations involved. Hugel [Ref. 12] and Subramaniam [Ref. 20] address the influence of the sheath region, but do not model the electric field, temperature, or sheath fall voltage.

Given the minuscule extent of the sheath versus thruster anode curvature, the problem at first appears one-dimensional in nature. A one-dimensional, collisional, equilibrium solution can satisfactorily reproduce the observed electric field and charge density distributions for the entire sheath and ambipolar regions for a sheath where the electron temperature equals that of the heavy species [Ref. 21]. However, this model cannot describe any decrease in current density away from the surface, or current constriction, at the anode surface which might be necessary in nonequilibrium. A two-dimensional model, developed by Biblarz and Dolson [Ref. 14], represents these phenomena and predicts the voltage drop in the region. It is shown that the sheath must account for a majority of the anode voltage drop, and that the sheath extent must be greater than the Debye length [Refs. 14,21]. Thus, a combination of one- and two-dimensional approaches appears to better describe sheath behavior. Incorporation of modelling of this sort may improve the ability of the computer codes cited above to properly describe quasi-steady thrusters.

Next, a description of the anode region is presented in order to delineate some of the possible effects of temperature.

III. ANODE DESCRIPTION

A. THRUSTER GEOMETRY DESCRIPTION

The majority of plasma thrusters to date have consisted of a central cathode rod surrounded by an annular shell anode, as shown in Figure 1 [Ref. 23]. The thruster illustrated is sufficient to produce needed thrust at current levels above one kiloamp. Below this level, an external magnetic field produced by an annular magnet is needed to ensure sufficient Lorentz force on the plasma propellant to meet thrust requirements. [Ref. 8]. As illustrated in Figure 1, the $\vec{j} \times \vec{B}$ body force simplifies into an axial $(j_z B_\theta)$ body force, which provides direct electromagnetic thrust ("blowing"), and a radial $(-j_r B_\theta)$ body force, which provides electromagnetic compression of the plasma and a subsequent pressure force along the cathode surface ("pumping"). [Ref. 6]

A notable exception to this geometry is the Stationary Plasma Thruster (SPT), a design from the former Soviet Union. The SPT is an example of a plasma propulsion system known as a Hall Current Plasma accelerator. An electric field is applied axially to a stream of flowing plasma, in addition to a magnetic field with a strong radial component, which is applied by an external electromagnet. When the axial electric field is applied and a current flows through the plasma, an azimuthal component of current is induced, i.e., the "Hall" current.

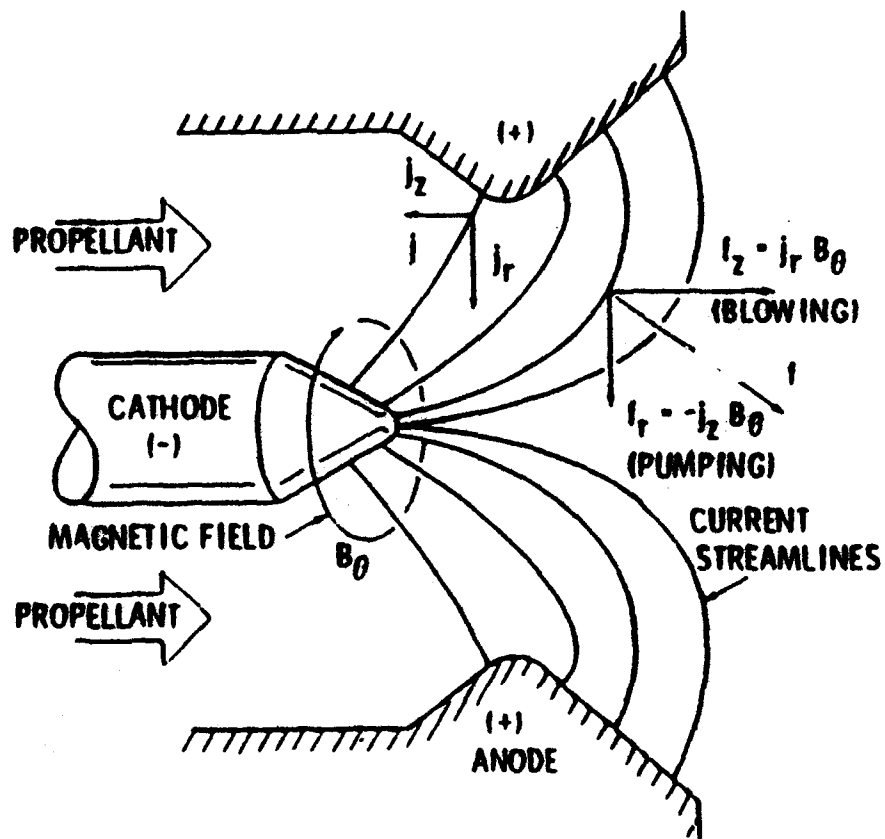


Figure 1 - Magnetoplasmadynamic (MPD) Thruster, with Axial and Radial Forces on Plasma Indicated. [Ref. 23]

Thrust is produced by electrostatically accelerating the ions in the plasma, as well as through the induced Lorentz force mentioned above. A strong radial magnetic field is applied to the plasma, whose properties are controlled to make the electron Larmor radius¹ small compared to the mean free path², while the ion Larmor radius is comparatively large. As a consequence, electron mobility in the axial direction is greatly reduced. Thus, the electric field energy is given mainly to the ions, producing axial ion acceleration. Collisions with neutral particles serve to accelerate the entire neutral plasma. [Ref. 24]

A pair of the final prototype design developed, the SPT-100, have been acquired by NASA recently from Fakel Enterprise in Kaliningrad, Russia, and are undergoing performance evaluation at the Jet Propulsion Laboratory. Designed at the Kurchatov Institute of Atomic Energy (IAE) in Moscow, USSR in the 1960's, smaller versions of the SPT-100 (SPT-50 and SPT-70) were flown beginning in 1972³. A specific impulse of 1,600 seconds and 50% efficiency, as well as space flights of fifty similar thrusters is claimed. The specific operational characteristics of the thruster are not well understood presently. Bohm diffusion of electrons and a phenomenon called "near-wall conductivity" have been proposed to explain the thruster's operation. This thruster is shown in Figure 2. [Ref. 25]

¹ Larmor radius is the radius of the helix traversed by a charged particle moving in a magnetic field.

² Mean free path is the distance traveled by a particle before making a collision.

³The suffix (i.e., "-70") indicates the characteristic diameter of the thruster in millimeters.

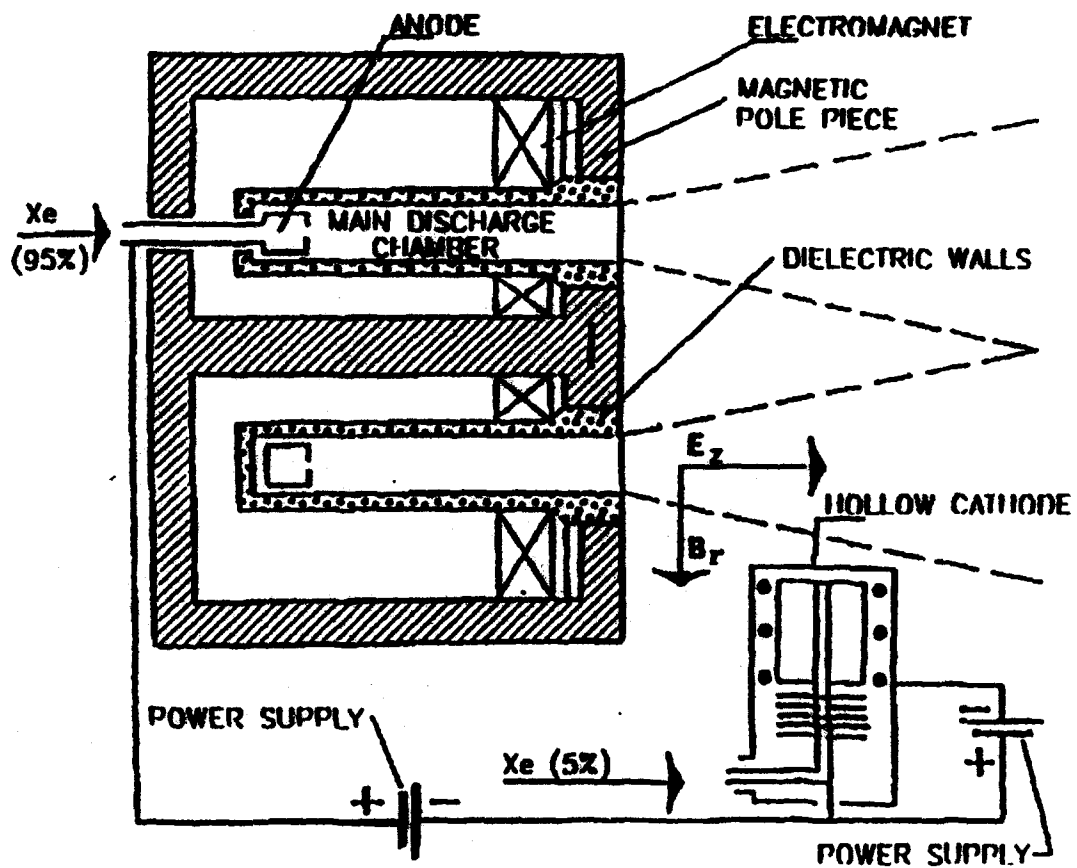


Figure 2 - Stationary Plasma Thruster [Ref. 25]

B. ELEMENTARY SHEATH FORMULAE DESCRIPTION

1. Discussion

Voltage losses and anode power deposition account for most of the inefficiency of plasma thrusters. In order to understand these losses, the anode region must be understood and related phenomena explained and modelled. As shown in Figure 3, a substantial drop in voltage occurs in a short distance from the anode surface.

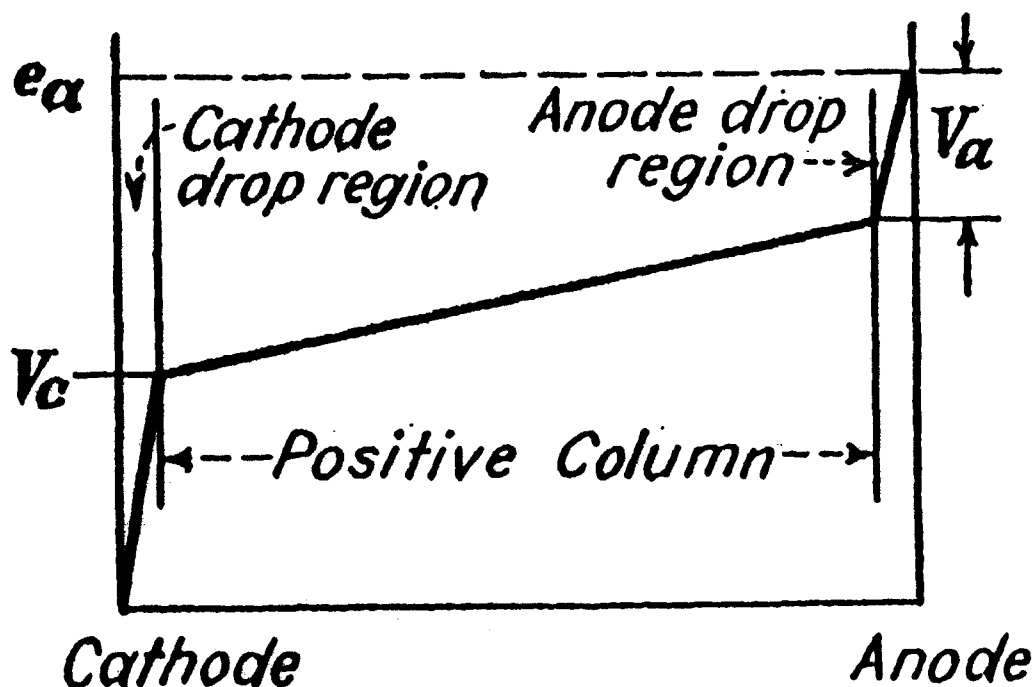


Figure 3 - Electric Field Between Two Electrodes, Including "Positive Column". [Ref. 27]

The anode fall region may be divided into two parts, the sheath and ambipolar regions. The plasma attempts to adjust itself near electrodes so as to shield the main body of the plasma from the electric field [Ref. 26]. The sheath is the region closest

to the anode surface within which the ion and electron number densities are unequal, with the electrons dominating the region. A high electron space charge exists at the anode surface. This is caused by the anode collecting incoming electrons in completing the arc current with the cathode. Positive ions are produced within the sheath by electron impact of neutral gas molecules, and the ions are repelled toward the cathode. At the cathode end of the anode drop region, the density of positive ions is high enough to almost neutralize the electron space charge, thus forming the positive column or core plasma. The essential positive ion current is created in this way near the anode. A more complete description of this process may be found in Cobine [Ref. 27] and von Engel [Ref. 28]. A fundamental characteristic of plasma behavior is its tendency toward electrical neutrality. Whenever local charge concentrations arise or external potentials are introduced into a system, these are shielded out in a distance known as the "Debye length". This distance must be much smaller than the system dimension for the ionized gas to be considered a plasma [Ref. 29]. Equation (2) gives the Debye length [Ref. 26].

$$\lambda_d = \sqrt{\frac{\epsilon_0 kT}{n_e e^2}} = 69.0 \sqrt{\frac{T}{n_e}} \quad (\text{m}) \quad (2)$$

*The Debye length effectively describes the radius of a shell around a charged particle outside of which the potential of the particle is not seen.

Another distance of interest is the electron mean free path, or distance traveled by a particle before making a collision. Equation (3) is from a derivation of Lin, Resler, and Kantrowitz [Refs. 30,31] giving the mean free path, with λ_s being the approximate sheath length.

$$\lambda = 0.12 \left(\frac{1}{n_e (e^2/3kT)^2 \ln(\lambda_s e^2/3kT)} \right) \quad (3)$$

Since the sheath extends at most a few mean-free lengths from the anode surface, curvature of the anode does not affect the governing equations in high pressure discharges. Thus, the region may be described in one dimension, the distance "y" from the anode surface. While the Debye length is sometimes assumed as the sheath extent, Reference 22 showed that the sheath thickness is a function of the anode fall voltage and the electron temperature. Equation (4) gives the appropriate form.

$$\lambda_s = \sqrt{\frac{\epsilon_0 \Phi_s}{en_e}} = \lambda_D \sqrt{\frac{e\Phi_s}{kT_e}} \quad (4)$$

An example case with a fall voltage of 100 volts gives a sheath extent of $\lambda_s = 2.352 \times 10^{-5} \text{ m}$. This compares to a computed Debye length of $\lambda_D = 1.690 \times 10^{-6} \text{ m}$. Therefore, the sheath can be an order of magnitude larger than the Debye length.

Nasser [Ref. 32] discusses an elementary theoretical approach to the glow discharge problem. He suggests a set of four one-dimensional ordinary differential equations, including the electron and ion current and number density equations, in addition to Poisson's equation. Most solution attempts have failed, with the boundary conditions being identified as the culprit. A similar attempt for the plasma thruster is discussed below.

2. Simplified Formulation

The steady probe equations are first written [Ref. 21] in their simplest form. The anode is assumed to operate as a heavily biased probe, which is true for low enough currents when the anode is not a source of ions. Whenever the temperature can be considered fixed, the energy equations are implicitly satisfied and, since ion inertia is neglected, the resulting set consists only of two species continuity equations and Poisson's equation. These equations are written in terms of y , which is the coordinate outward from the planar positive surface. Constants and variables are listed in Table 1.

TABLE 1 - NOMENCLATURE

a ...characteristic length of plasma	n_{∞} ...species number density at core plasma
$D_{i,e}$...species diffusion coefficient	N ...total number density
e ...elementary charge constant	T ...temperature
E ...electric field	T_0 ...neutral species temperature
E_0 ...electric field at anode surface	α ...two-body recombination coefficient
E_{∞} ...electric field at core plasma	ϵ_0 ...permittivity constant
$j_{i,e}$...species current density	ν ...ionization coefficient
J ...total current	$\mu_{i,e}$...species mobility coefficient
k ...Boltzmann's constant	Φ_a ...anode fall potential
K ...current parameter	λ ...mean-free distance
$n_{i,e}$...species number density	λ_d ...Debye length
\dot{n}_e ...time rate-of-change of n_e	λ_s ...Sheath thickness

Note: Species subscripts denote ions (i) and electrons (e).

$$j_i = e\mu_i n_i E - (eD_i) \frac{dn_i}{dy} \quad (5)$$

$$j_e = e\mu_e n_e E + (eD_e) \frac{dn_e}{dy} \quad (6)$$

$$\frac{dE}{dy} = \frac{e}{\epsilon_0} (n_i - n_e) \quad (7)$$

$$J = j_i + j_e \quad (8)$$

$$\mu_{i,e} = \frac{eD_{i,e}}{kT_{i,e}} \quad (9)$$

Here, the j 's are species contributions to the total current density. The existence of negative charges as free electrons is pivotal in the formulation. Next, the Einstein relation, equation (9), is introduced to write the mobilities in terms of the diffusion coefficients. We assume that the diffusion coefficients remain constant in the problem.

Equations (5) and (6) are next solved for $dn_{i,e}/dy$. The species current density equations are found from the net reaction rate of the plasma. Equations (10) and (11) combine to produce space derivatives for species current density.

$$\dot{n}_e = v_i n_i - \alpha n_i n_e \quad (10)$$

$$\frac{dj_i}{dy} = \frac{-dj_e}{dy} = e \dot{n}_e \quad (11)$$

Combining equations (5)-(11) produces a set of five coupled, non-linear differential equations describing the sheath. These are nondimensionalized to adjust all variables to the first order, and are rewritten below as equations (12)-(16), with nondimensionalized variables denoted by "x". Nondimensionalization can be accomplished as follows: The species number densities n_i, n_e are divided by their values at infinity to produce output from the anode surface to unity at the ambipolar boundary. The current densities j_i, j_e are divided by the total current, allowing the output to show the "mirror behavior" of the two currents. The electric field is divided by the initial anode value to give output starting from unity at the surface and decreasing to the final core field value. The variable "y" is divided by the characteristic length³ of the plasma, "a", producing \bar{y} . These corrections allow all output to vary in the range from zero to one, as a function of distance from the anode.

³The characteristic length is defined so as to cancel the multiplying factor in the electric field equation, (14), ($a = 1.107 \times 10^4$). This allows a physical interpretation of the ion/electron number densities, as well as the decay rate of the electric field.

$$\frac{d\tilde{n}_i}{dy} = \left(\frac{aeE_0}{kT_0} \right) \tilde{n}_i \tilde{E} - \left(\frac{aeE_0}{kT_0} \right) \tilde{j}_i \quad (12)$$

$$\frac{d\tilde{n}_e}{dy} = - \left(\frac{aeE_0}{kT_0} \right) \tilde{n}_e \tilde{E} + \left(\frac{aeE_0}{kT_0} \right) \tilde{j}_e \quad (13)$$

$$\frac{d\tilde{E}}{dy} = \left(\frac{aen_0}{E_0 \epsilon_0} \right) (\tilde{n}_i - \tilde{n}_e) \quad (14)$$

$$\frac{d\tilde{j}_e}{dy} = - \left(\frac{akT_0 v_i}{eE_0 D_i} \right) \tilde{n}_e (\tilde{v}_i - \alpha \tilde{n}_i) \quad (15)$$

$$\frac{d\tilde{j}_i}{dy} = \left(\frac{akT_0 v_i}{eE_0 D_i} \right) \tilde{n}_i (\tilde{v}_i - \alpha \tilde{n}_i) \quad (16)$$

Attempts to solve this equation set using the computer code discussed below shows the set to be extremely sensitive to initial conditions. The computer code solver uses a "marching" scheme from the anode to the undisturbed plasma. The initial conditions are chosen to produce the electric field potential drop observed in actual thrusters. First and second space derivatives of the electric field are used as diagnostic checks to ensure reasonable output values and indicate instability of the integration process. Figure 4 shows the required resulting curves for the electric field and its first and second derivatives.

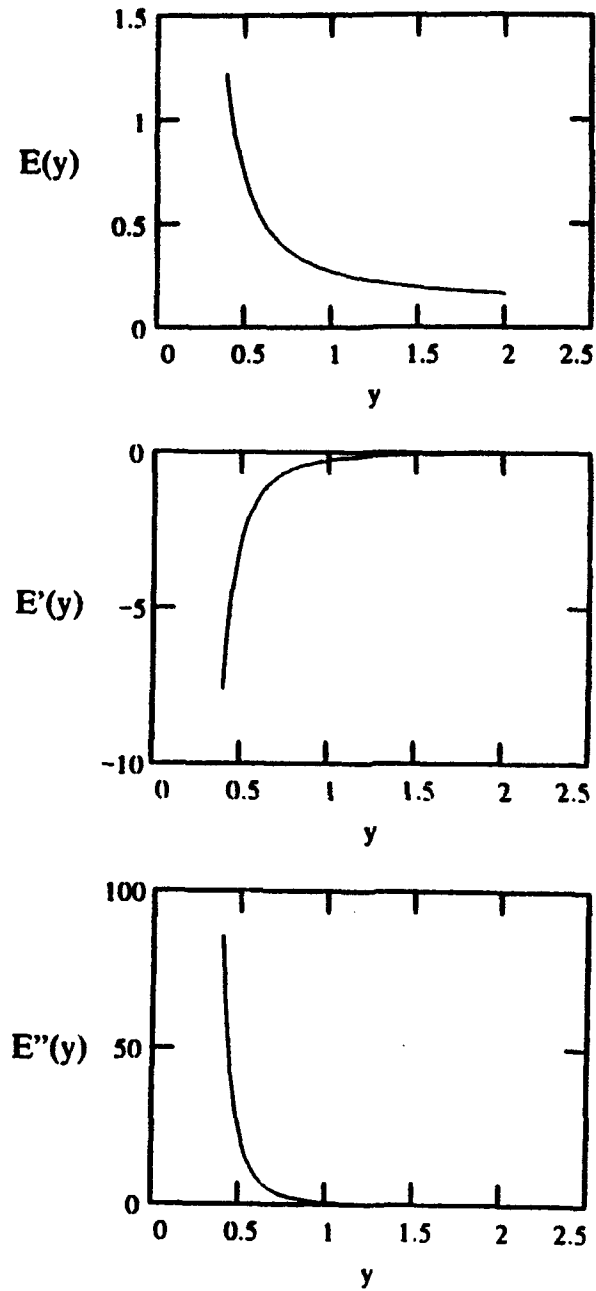


Figure 4 - Electric Field and First Two Space Derivatives Used as Diagnostic Checks for Integrator Output. (Plotted for a Generic Exponential Function).

Ecker characterizes the plasma at the anode as a double sheath, with the inner section called the "inertia sheath", and an outer section called the "energy loss section". The inner section shows a potential rise of the order of one volt, with the outer section showing the exponential potential drop shown in Figure 4. While this double sheath may in fact describe the actual sheath region, the formulation above only models the potential drop portion of the sheath, and does not attempt to produce the potential rise of the inner sheath. In addition, Ecker's current constrictions are of a "macroscopic" nature, whereas those of Reference 14 and this work are "microscopic" [Ref. 33].

Data for a 6,000°K Nitrogen plasma were used to test the equation set [Ref. 21]. Producing a proper solution required adjusting the initial conditions to force the curve shapes discussed above. Using Equations (2), (3), and (4), the mean free path, Debye length, and approximate sheath extent are calculated as $\lambda = 9.352 \times 10^{-3}$ m, $\lambda_D = 1.690 \times 10^{-6}$ m, and $\lambda_s = 2.352 \times 10^{-5}$ m (this assumes a drop voltage of 100 volts).

3. Approximate Formulation

Reference 21 explores the above equation set by taking advantage of the symmetry among the equations, and introducing two parameters, K^+ and K^- , shown below.

$$K^+ \equiv \frac{j_i}{eD_i} + \frac{j_e}{eD_e} \quad (17)$$

$$-K^- \equiv \frac{j_i}{eD_i} - \frac{j_e}{eD_e} \quad (18)$$

It can be shown that the resulting equations can be manipulated to yield a single, ordinary differential equation for the K 's in terms of the electric field. The resulting equation can be scrutinized for two distinct temperature regimes. Note that while the total current density, J , is constant in a steady, one-dimensional case, the K 's can vary and will in turn also depend on the degree of reactivity of the plasma (n_e), i.e.,

$$\frac{dj_i}{dy} = \frac{-dj_e}{dy} = en_e \quad (19)$$

Because ion diffusion is much slower than electron diffusion, it can be shown that the K 's are related by

$$K^+ \approx -K^- + \frac{2J}{eD_e} \quad (20)$$

As will be evident, at the electrode surface the K 's are equal to each other and at the undisturbed plasma, $K^- = 0$. The total current density may be evaluated from

$$J = en_e v_{e\infty} \quad (21)$$

where $v_{e\infty}$ is the electron drift velocity beyond the ambipolar region which is strictly a function of E_∞/N , (i.e., of the ratio of undisturbed electric field to the total number density).

a. Effects of Temperature on Anode Constriction

It is useful to investigate the overall effects of temperature. Since temperature will be considered constant, it comes in as a parameter in this formulation whereas charge density and electric field remain as variables. Intuitive arguments will be introduced which suggest that the electron and ion/neutral temperatures play a rather singular role in determining the intrinsic dimensionality of the problem, (i.e., there are cases when the geometry of the current lines is not necessarily impressed by the electrode geometry). Since the problem is described by moderate pressure, largely collisional sheaths, the ion and neutral temperatures are anticipated to remain reasonably equal. Depending on the gas, the electron temperature, on the other hand, can be elevated from the gas temperature at the anode where actual magnitudes depend on the local value of E/N . In order to get a perspective on the effects of temperature, we shall consider two extremes, namely, the case where the electron and ion temperature are the same (the equilibrium case) and the case where the electron temperature is substantially elevated from that of the ions/neutrals (the two-temperature case).

(1) *Case I: $T_e = T_i = T_0$ (Equilibrium)*

The charge densities can be eliminated by combining equations (5)-(9), (17) and (18). The resulting equation can be shown to be

$$\frac{kT_0}{e} \left(\frac{K^*}{E} \right)' + K^* = \frac{2J}{eD_e} - \left(\frac{kT_0 \epsilon_0}{e^2} \right) \frac{1}{E^2} \left[EE'' - (E')^2 - \frac{1}{4} \left(\frac{e}{kT_0} \right)^2 E^4 \right] \quad (22)$$

If the electric field decreases monotonically from the wall to the undisturbed plasma (i.e., from $E_0 \rightarrow E_\infty$), then as $y \rightarrow \infty$, $E \rightarrow E_\infty$, $E' \rightarrow 0$, $E'' \rightarrow 0$.

So that in equation (22) above the "outer solution" becomes:

$$K^* = \frac{2J}{eD_e} \quad (23)$$

Now this represents an acceptable solution from a physical point of view. Moreover, as $y \rightarrow \infty$,

$$\dot{n}_{e\infty} \approx D_e (K^*)' \approx 0 \quad (24)$$

which is also acceptable for an equilibrium situation at the undisturbed plasma. Results [Ref. 21] are shown in Figure 5 for the case of nitrogen at 6000°K using an approximate electric field distribution.

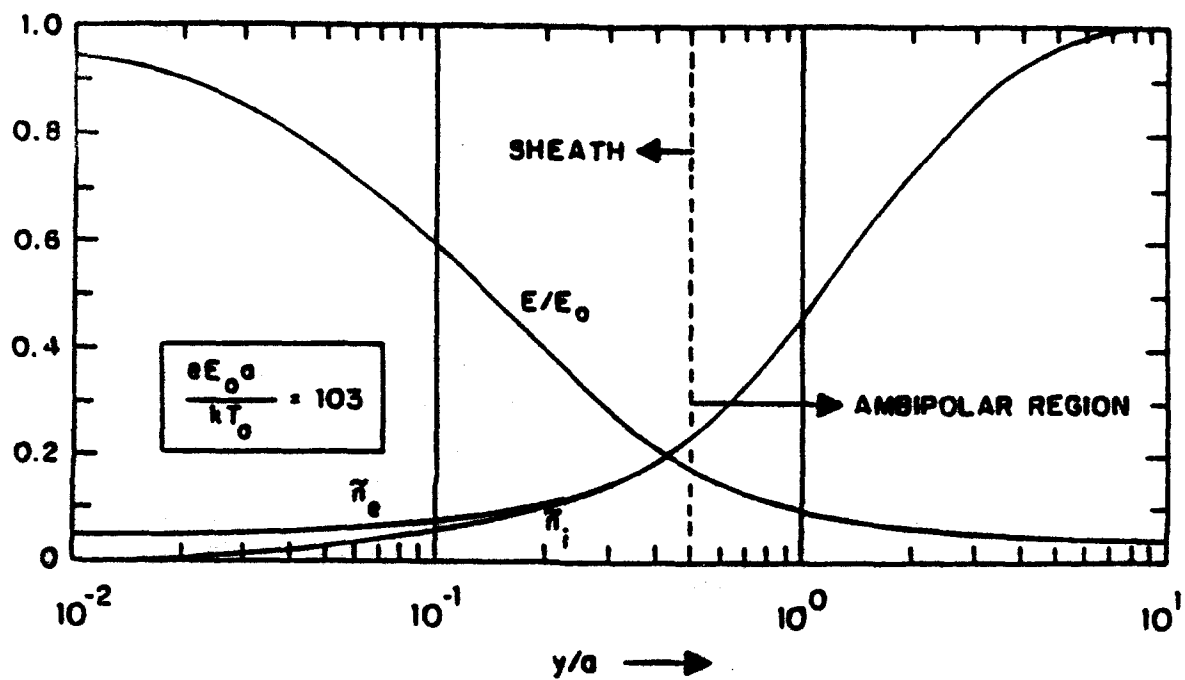


Figure 5 - Electric Field and Species as a Function of y , Distance From Anode. An Approximation Using a Shaped Electric Field and Isothermal Plasma [Ref. 21].

(2) *Case II: $T_e \gg T_i = T_0$ (Two-Temperature)*

In this case the same procedure as before yields the following equation where terms divided by T_i have been dropped when compared to their counterparts divided by T_0 .

$$(K^*)' - \frac{K^*}{E} E' = \frac{2EJ}{kT_0 D_e} + \frac{2\epsilon_0}{kT_0} EE'' + \frac{\epsilon_0}{eE} E''E' - \frac{\epsilon_0}{e} E''' \quad (25)$$

Assuming the same monotonic decrease as before for the electric field from the wall to the plasma proper, as $y \rightarrow \infty$, $E \rightarrow E_\infty$, $E' \rightarrow 0$, $E'' \rightarrow 0$.

Then the outer solution becomes

$$\frac{dK^*}{dy} = \frac{2eEJ}{kT_0 e D_e} \quad \text{with } n_{e\infty} > 0 \quad (26)$$

Or, $K^* \rightarrow (\text{constant}) y + \text{constant}$, and $n_{e\infty}$ keeps increasing with y .

This is not the proper outer solution for the one-dimensional, equilibrium plasma that we seek because the net ionization rate continues to increase well inside the plasma proper where conditions should saturate, yielding a constant electric field. Therefore, as formulated, Case II is not amenable to a one-dimensional solution. References 14 and 21 show how this case can be analyzed under a multidimensional approach. These references also discuss a method for describing the electron temperature as a function of E/N , then how to couple a simplified energy relation which satisfactorily describes a two-temperature plasma. The necessary ingredient to make equation (26) approach zero beyond the decrease of E to E_∞ is to allow J

to fan out as indicated in Figure 6. Thus, in equation (26), the product "EJ" can bring down the charge production rate to arbitrarily low values. Alternatively, it is possible to explore techniques of bringing the electron temperature down to be in closer equilibrium with the ions and neutrals. Transpiration cooling is one such means.

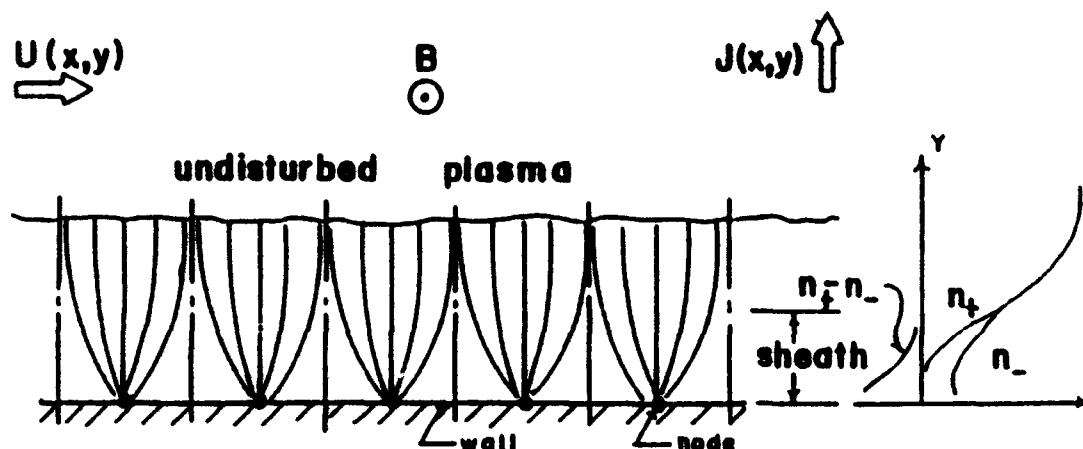


Figure 6 - Two-Dimensional Model of Current Paths Showing Periodic Structure. Thermal Instabilities and Inhomogeneities Would Favor One Site Over Others and a Single Macroscopic Constriction May Then Be Produced. [Refs. 14, 34]

b. Similarity to Vacuum Arc Phenomena

Instability phenomena observed in vacuum arcs [Ref. 35] are very similar to those observed in self-field thrusters [Ref. 12]. After the establishment of the current, the anode region operates in a vapor that issues from the electrodes. In vacuum arcs, Miller characterizes the anode region as operating in one of five distinct modes, ranging from a passive, low current mode to a high current, fully developed spot mode [Ref. 36]. Given the similarities mentioned above, vacuum arc anode research should be helpful in the understanding of MPD thruster transition to the anode spot mode. Existence diagrams after Miller [Ref. 36] are shown in Figure 7, which divide operating modes into regions as a function of anode current versus electrode geometry. Figure 7 shows the transition from glow to spot mode.

Anode spot formation at high currents is clearly a factor in limiting anode lifetime. Various phenomena have been related to anode spotting. Hugel [Ref. 12] relates the transition to spotting mode to an increase in J^2/η above a critical level. A separate factor connected with the spot mode is surface temperature of the anode. Rich, et.al., [Ref. 37] show that anode spotting is preceded by a luminous "footpoint" and followed by local melting prior to spot formation. Separately, Schuocker [Ref. 38] finds a connection between spotting initiation and the factors of anode evaporation and magnetic constriction in vacuum arcs with high currents. Experimental investigations must be performed to see if the above-mentioned vacuum arc criteria apply to self-field thrusters.

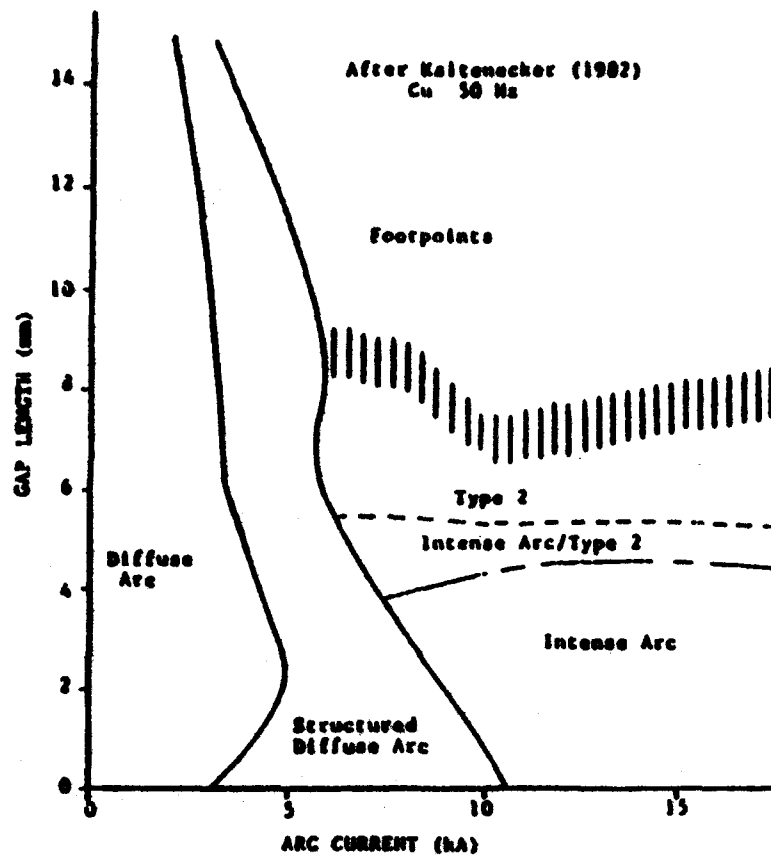


Figure 7 - Anode Discharge Modes as a Function of Current and Gap Length. [Ref. 36]

C. COMPUTER CODE

Rather than using linear approximations to equations (12)-(16), the nonlinear set was used, with initial conditions adjusted in an attempt to produce observed electric fields from probe data. First and second space derivatives of the electric field were used as diagnostic checks to ensure computed output was reasonable. Initial conditions computed from the approximate formulae in Reference 21 were used. The equation set above presents a difficult problem for two reasons, nonlinearity and multiple time constants. The species number density equations, (12) and (13), both contain a nonlinear term, each with a time constant of its own. In addition, the electric field equation, (14), adds a possible third time constant. This constitutes a "stiff" set of equations. Attempts were made to solve the set with the data discussed above, using Gear's method of backward differentiation, in hopes that the variables would change slowly enough with each iteration to render a convergent iterative process. As described in Reference 39, if some reactions are slow and others fast among a set of coupled equations, the fast ones will control the stability of the method. This is addressed in the DGEAR program available from the International Mathematical & Statistical Library (IMSL). The latter software contains an Adams predictor-corrector method, as well as Gear's method, which is well known for its success at solving stiff equation sets. The DGEAR software allows for a choice of functional or chord iteration methods, as well as a choice of Jacobian matrices. A more detailed discussion of this software can be found in Reference 39 and in the IMSL library. [Ref. 39]

D. COMPUTATIONAL RESULTS

Numerous computer runs were completed using the initial conditions taken from Reference 21. In addition, data for the ionization coefficient ν , Figure 8, was taken from References 40 and 41.

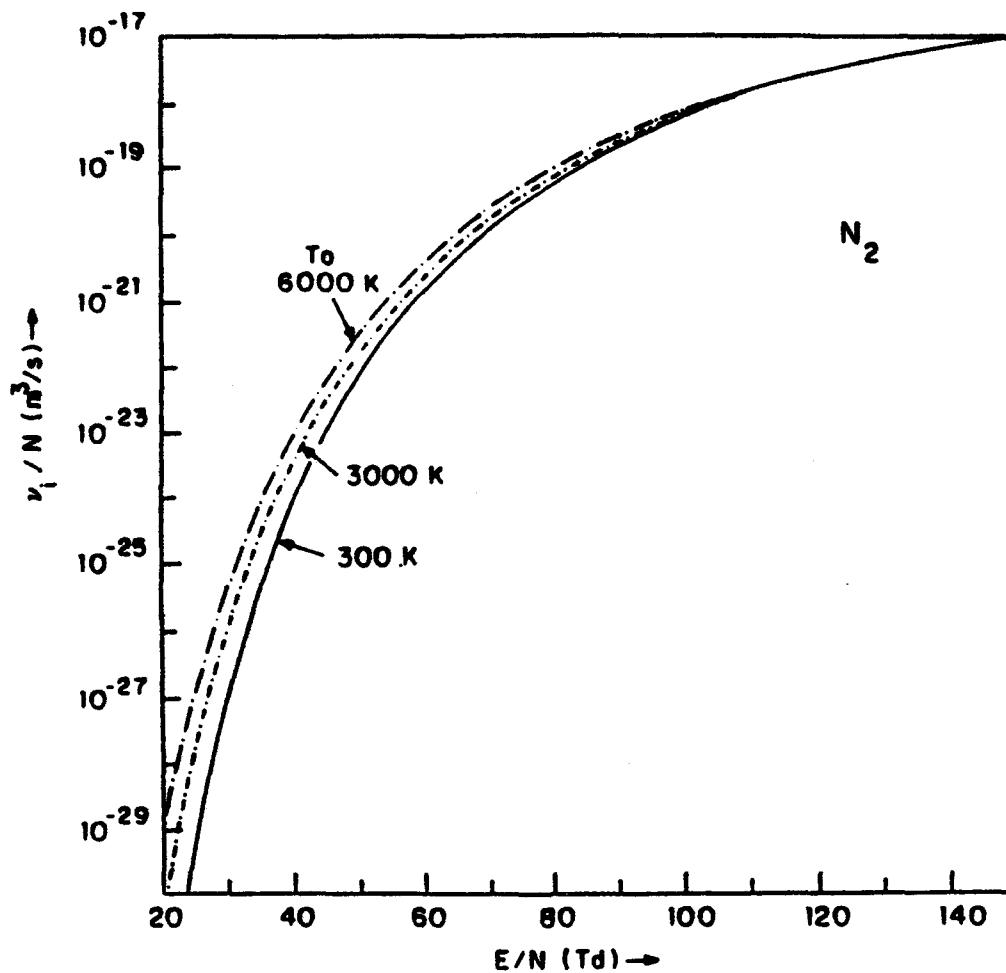


Figure 8. Ionization Coefficient ν as a Function of E/N for Nitrogen for Various Vibrational Temperatures (Refs. 40,41).

Various combinations of initial conditions and ionization coefficient were used. As mentioned above, the electric field and its space derivatives were used as

diagnostic/reasonability checks on the output. Individual, as well as multiple computer runs were attempted to model the sheath region. Nonlinearities in the equation set are clearly seen in Figure 9. The ion number density does not reach that of the electrons, and the latter population growth rate continues to grow without bound. The shape of the electron population curve is very sensitive to its initial value. As shown in Figure 9, the latter population has too high a growth rate when compared to the ion population, and the latter does not "catch up". Increasing the initial value of n_e flattens out this curve to a reasonable shape. Above an initial value of approximately 0.06, however, the plot of n_e "dips" after a certain distance and then continues to increase as expected. This gives an approximate upper value for this initial value. To avoid instabilities like this, small "slices" were taken of the output after a small number of integration steps and multiple runs were used to form a "cut and paste" plot of the region. When a reasonable plot shape was produced, the value of ionization coefficient was varied in the "slices" to attempt to produce the required end values for electric field and species population. Both multiple and equilibrium values for the ionization coefficient were used. When the data showed signs of instability and failure to follow the required forms of Figure 4, a "slice" was made in the data stream, and the data points from this point used to start a new computer run. This approach was taken in the hope of avoiding singularities in the integration from anode surface to ambipolar region. In addition to the diagnostic checks shown in Figure 4, an additional data check is provided by the transition from the sheath to the ambipolar region.

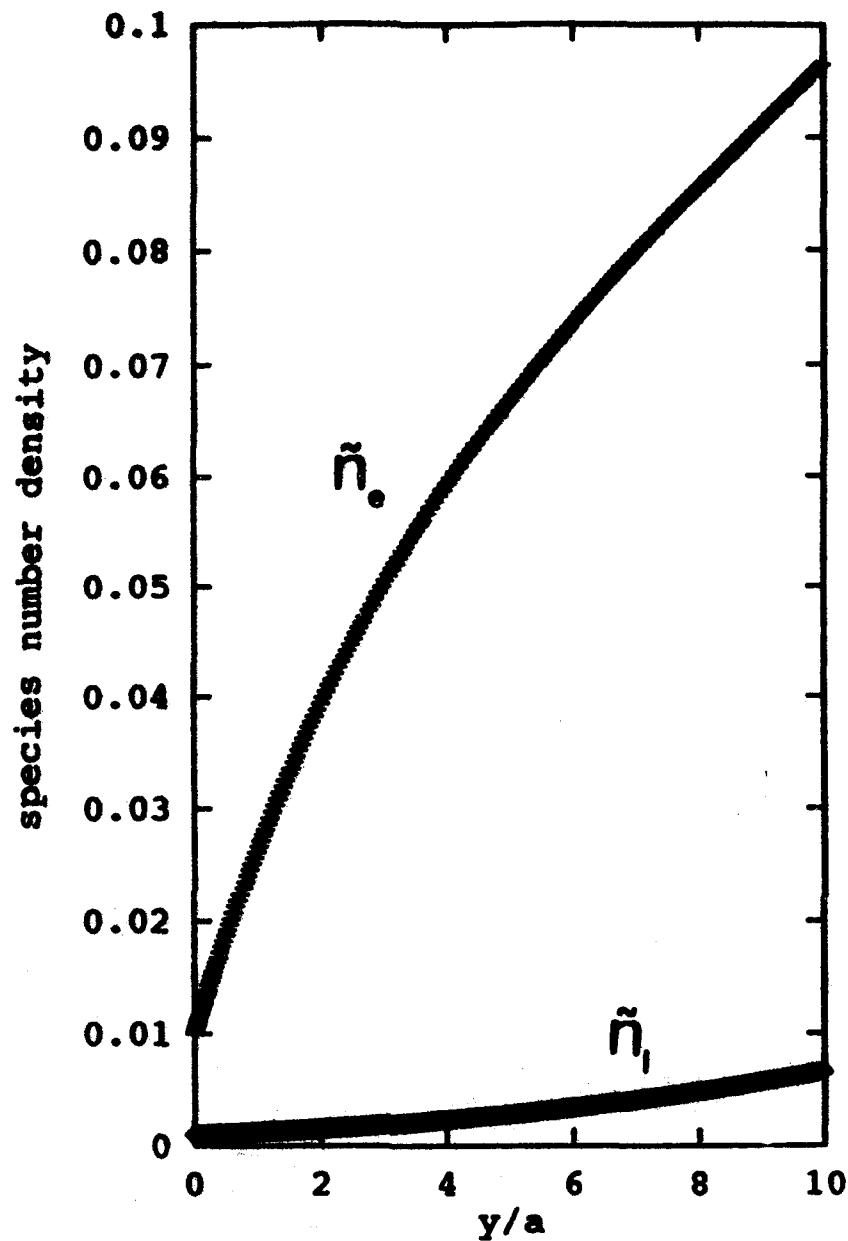


Figure 9. Species Number Density plots for Individual Computer Run, Showing Divergent Tracks for Ion and Electron Populations, and Effect of Nonlinearity.

As shown in Figure 5, the species number densities are equivalent in this region, as are their change rate. Thus, setting Equations (12) and (15) equal to each other and solving for \tilde{n} yields a value of 0.5 in the ambipolar region. As indicated in Figures

10-11, the output produces the desired plot slopes for electric field and species number density. However, the number density plots cross long before approaching the required value of 0.5. In addition, neither electric field nor species number density approaches an equilibrium value or shows sign of levelling off. Apparently, the multiple time constants and nonlinear portions of the number density equations combine to create a seemingly intractable system. Solutions for this system may be possible for specific, individual initial condition sets, but the problem does not appear amenable to this approach in general. A one-dimensional system such as this may be better described through the approach of boundary layer theory or nonlinear dynamics and chaos. Given the effort and difficulty involved in the latter, a one-dimensional approach such as that modelled above does not appear useful. A combination of one- and two-dimensional modelling would appear to be more useful, as discussed in Reference 14. A one-dimensional model may be useful, but only in an approximation approach, with a shaped electric field, such as that used in Reference 21.

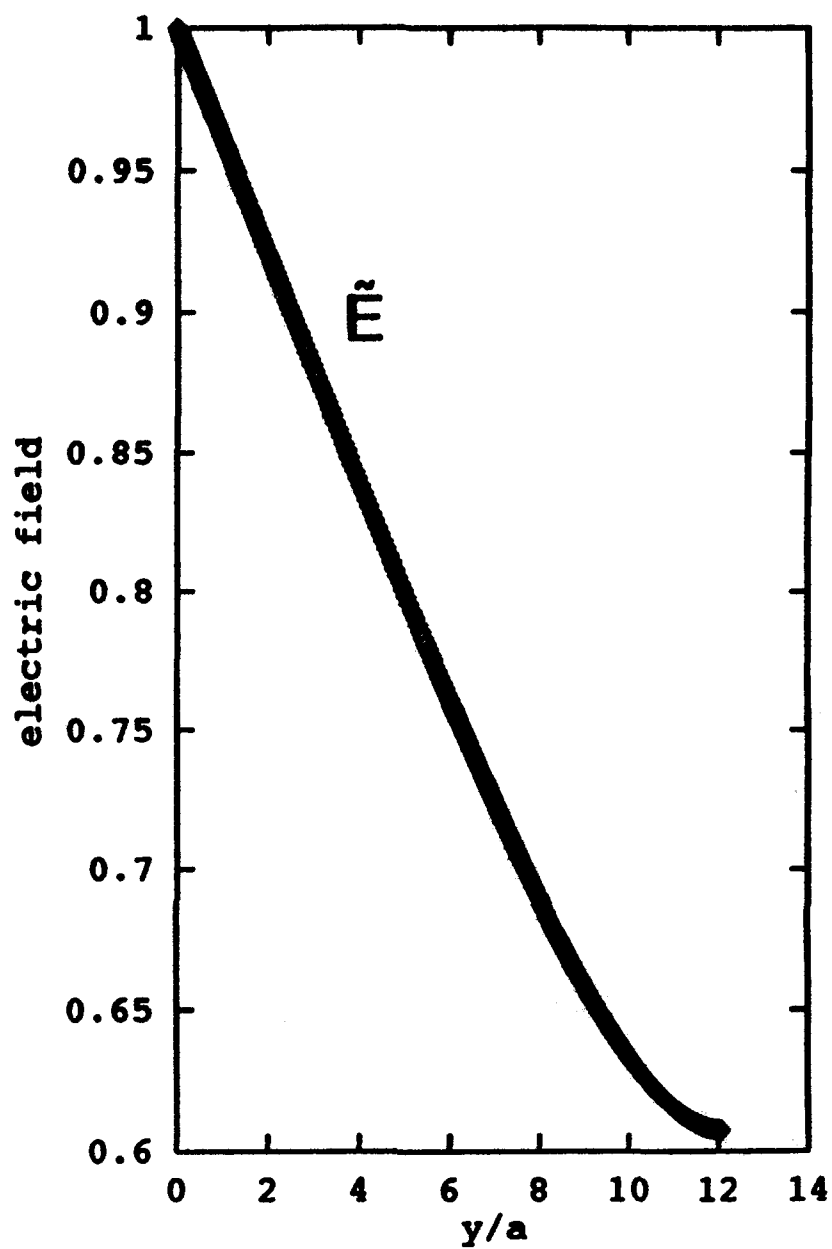


Figure 10. Electric Field as a Function of \bar{y} , Distance From Anode, Using Equations (12)-(16).

ionization (with some help from the tail of the Maxwellian distribution of electron energies), what results is a breakdown voltage appreciably below the ionization potential. This then could be an explanation for the low voltage breakdown observations [Ref. 42]. Clearly, gases with low ionization potentials and lots of atomic electron energy levels are preferred (such as cesium and barium) but low-voltage breakdown has been observed with most gases.

The increase of the anode fall voltage above the ionization potential has been related to the electron Hall parameter, since a reduction of this parameter decreases that voltage and corresponding losses. Control of the local magnetic field through the use of an array of permanent magnets as well as implementation of transpiration cooling (which increases the electron collision frequency) have both yielded some encouraging results. Because the anode fall also scales up with J^2/η , it is conceivable that current inhomogeneities and plasma instabilities which are reflected in this parameter are in the picture as well. [Ref. 44]

In summary, any possible reduction of the anode fall voltage will hinge on a thorough understanding of the anode region, with its associated sheath and ambipolar regions, where electron temperature effects, ionization effects, and magnetic field effects play a pivotal role. If transpiration cooling is present, then additional phenomena of fluid-dynamic nature may come into play. Experimental observations with atmospheric discharges indicate the possible presence of convective effects at the anode. [Ref. 45]

IV. TRANSPIRATION COOLING

Transpiration cooling of the anode has often been promoted as an attractive means of recovering a large portion of the power deposited there. Additionally, the onset of melting may be minimized or even avoided by active anode cooling. Rich, et.al., related high anode temperatures to anode spotting [Ref. 37]. Similarly, Park and Choi showed that low thermal diffusion leads to erosion and, consequently, anode damage [Ref. 46]. Active anode cooling via transpiration is one means of ensuring high thermal diffusion and extending anode lifetime. Early work by Schoeck, et. al., [Ref. 47] showed that up to 80% of the energy deposited in the anode is recoverable via transpiration cooling. While this study used non-convective, high-intensity argon arcs, it is reasonable to assume that this effect would apply in MPD arcs using other propellants. Although this cooling method has not been studied for incorporation in plasma thrusters for some time, it has been recently considered as a means of cooling the fuselage of the National Aerospace Plane (NASP). Plasma thruster designs could undoubtedly gain from this database, and due consideration should be given to this cooling approach for the anode.

For a given mass transfer flow rate, the heat flux reduction to a surface is inversely proportional to the molecular weight of the injected gas. Use of the propellant as coolant as well as fuel would eliminate the need for additional tankage and pumps, simplifying the design considerably. Lithium has been considered to be

the propellant of choice, primarily because of its low molecular weight, its favorable ionization potential, and its low-volume tankage properties. It has a relatively low first ionization potential of 5.4 eV and a high second ionization potential of 75.6 eV. This single ionization potential range of over 70 eV compares to approximately 20 eV for Cesium and 27 eV for Potassium [Ref. 48]. This provides a broad temperature range within which only single ionization will occur. Large temperatures must be reached within the gas before double ionization occurs. As the gas temperature is increased several thousand degrees Kelvin, it undergoes ionization and disassociation. Thermal energy deposited can be recovered through nozzle expansion at the exit. However, residence time of the gas is not long enough to ensure recombination. Thus, the energy invested in ionization and disassociation will be lost. [Ref. 49] Lithium has been shown to produce specific impulse figures in excess of 7,000 seconds at 70% efficiency in steady-state thrusters, [Ref. 50] whereas all other propellants have been limited to less than 3,000 seconds specific impulse at less than 40% efficiency [Ref. 6]. Subramaniam has concluded that:

...regenerative cooling of anodes (at the specific impulse values in the MPD regime) is possible only with hydrogen or with alkali-metal propellants, notably lithium. In the latter case, the ideal anode operating mode would be evaporation and ionization of the propellant on the porous or wetted anode surface, resulting in increased ion current fraction, reduced anode voltage fall and utilization of part of the anode loss energy [Ref. 51].

Liquid coolants, as well as reducing storage requirements, offer the advantage of providing latent heat of vaporization for energy disposal. However, design problems can occur if the liquid is allowed to vaporize within the porous structure. Problems

arise due to the abrupt increase in pressure gradient as the coolant vaporizes. Since coolant flow generally have three-dimensional characteristics, the flow will be diverted around the vapor bubbles and hot spots often develop. The technical practicality of using molten lithium to cool a porous tungsten anode would seem to be beyond current technology. On the other hand, the products of decomposition of hydrazine (gaseous hydrogen and ammonia) have proven to be efficient and practical coolants [Ref. 52].

Given the performance figures above, using an auxiliary coolant gas even with high molecular weight (e.g., NH_3 , N_2 , CH_4 , etc.) which could serve as a propellant once released from a porous hot tungsten anode surface would seem to more practical, vice dealing with molten lithium. Experimental studies would be needed to compare the approaches. Kuriki and Suzuki performed experiments with a quasi-steady MPD thruster to study the effect of anode gas injection (Argon). At high currents of up to 10 kA, increases in thrust, specific impulse, and flow discharge stability were observed [Ref. 53].

There is some question as to the likelihood of current constriction resulting from anode gas injection. In such a case, swirling or circulating the propellant gas would help to move any footpoints that developed around the anode surface and prevent them from becoming fully-developed spots. Additionally, an applied magnetic field could serve to circulate the footpoints as well. The unique advantage of transpiration cooling hinges on providing effective anode cooling while supplying

hot propellant, but the real benefit will depend on how small the amount of coolant required really will be.

Transpiration cooling has proven to be as desirable as it is challenging. It is complicated to implement, with associated reliability problems and difficulty of analytical predictions. While the production of thicker boundary layers is largely ineffective against the electron flux heating, the cooling itself is most efficient and a substantial fraction of the energy transferred to the anode is recoverable. The arguments of Chapter Three indicate that a reduction of the electron temperature in the anode would have the desirable effect of reducing the initial current spotting which can be conjectured to be the path that leads to anode arc spots. This electron temperature reduction can be done most effectively by polyatomic gases (which have a high δ -loss factor) emanating from the anode surface [Ref. 54].

The arguments relating to transpiration cooling might be summarized as follows:

Favorable Outcomes

- No separate cooling mechanism for anode required,
- Adds "hot" propellant to exhaust "recovering" most of the electrical power loss to the anode,
- Quenches T_e , thus likely to postpone anode spotting and reduce the heating associated with the electron thermal energy ($5kT_e/2e$),
- Reduces bulk convective heating,
- Reduces the local electron Hall parameter by increasing the collision frequency,

Favorable Outcomes (cont'd.)

- Allows for some radiation cooling from the hot tungsten surface (about 120 watts/cm² at 2800°K [Ref. 51]),
- Hydrogen/ammonia gases flowing through hot porous (sintered) tungsten represent a compatible, proven technology.

Unfavorable Outcomes

- May decrease the electrical conductivity in the anode region,
- May destabilize the ionization processes in the sheath and bring about significant fluctuations in the current,
- Disrupts "cathode jet" in front of the anode with unpredictable consequences,
- Introduces propellant which may not be hot enough, not ionized enough, or not in the proper place for $\vec{j} \times \vec{B}$ acceleration,
- Transpiration cooling through a porous (tungsten) anode is a difficult design problem.

V. CONCLUSIONS AND RECOMMENDATIONS

Plasma thrusters offer distinct advantages in terms of payload delivered for interplanetary missions, as well as for orbital transfer. A recent comparison completed by Choueiri, Kelly, and Jahn shows a mass savings of 65 tons for an orbital transfer from low Earth orbit to geosynchronous Earth orbit using a quasisteady MPD thruster as opposed to an advanced chemical thruster.⁶ This superior performance comes at the expense of low thrust-to-weight ratio and long transit time. However, given the large cargo/logistic requirements of a manned interplanetary mission, delivery of payload must be maximized. Thus, further work to characterize more fully thruster behavior and anode contributions in particular are certainly warranted. [Ref. 55]

The "cut-and-paste" method used to generate Figures 10 and 11 is not of practical use as a modelling method, due to the large effort involved. It did produce the expected electric field and species number and current density plots near the anode, but failed to produce the entire sheath out to the ambipolar region. The nonlinearity of the equation set led to a quickly deteriorating solution. A more practical approach using nonlinear dynamics and/or chaos must be developed to model the sheath numerically.

⁶This assumes a specific impulse of 2,000 seconds, 600 kW of input power, and a 270-day transit time.

Depending on the propellant mass fraction used for cooling, the transpiration scheme discussed above presents some rather unique advantages. A hot anode which uses only a small amount of propellant for cooling need not be penalized for any lost thrust. If in addition, we increase anode lifetime by delaying the formation of anode arc spots, then the scheme is all the more desirable. A decrease of the electron temperature in the vicinity of the anode may bring about a more homogeneous flow of current and a reduction in the heating effect associated with the high electron kinetic energy. Recovery of the heat deposited at the anode would be most important if the propellant fraction is high. In such case, nozzle expansion of the hot-propellant/coolant-gas might be implemented.

Means of limiting anode losses through decreasing anode fall voltages were discussed, including the control of the local Hall parameter and the implementation of thermionic arc breakdown. The electrical conductivity (of a nonreacting plasma) could possibly decrease as a result of transpiration cooling and this might increase the anode fall voltage.

Additional work needs to be done in the following areas:

- Investigate effectiveness of nonlinear dynamics and chaos in solving sheath equation set,
- Incorporate adequate one- or two-dimensional sheath modelling in quasisteady MPD numerical codes,

- Investigate the role that fluid dynamic effects play in MPD thruster anode discharges,
- Investigate the effect of transpiration cooling on current and plasma stability, as well as on thruster performance and lifetime,
- Determine effectiveness of transpiration cooling's increase of the collision frequency parameter,
- Compare performance of gaseous propellant/coolants versus hybrid designs with lithium propellant/gaseous coolant,
- Determine if required percentage of propellant gas as coolant is practical (e.g., less than 10%),
- Investigate effect of surface imperfections as focal points for current constrictions and as precursors to anode spotting.

APPENDIX A

The following software includes the calling program, SHEATH, its two subroutines, FCNJ and YDOT, and the DGEAR integrator. The latter is quite extensive in length and includes ten subroutines, including the following: DGRST, DGRCS, DGRPS, DGRIN, LUDATF, LUELMF, LEQT1B, UERTST, UGETIO, and USPKD. A detailed discussion may be found in IMSL literature or Reference 39.

```

*****
      Program Sheath                                000010
C                                           000020
C-----Calling program for DGEAR integrator. Initial conditions are 000030
C input via READ statements and keyboard entry. Output is to data 000040
C files via the DGRST subroutine. Diagnostic check of output via 000050
C Figure 4 printed to data file from DGRST subroutine. Consult 000060
C DGEAR comments for variable descriptions not listed below. 000061
C----- 000062
      REAL E,K,EPS,TI,EF,EFINF,DI,DE,NUINF,C1,K1,A 000070
      INTEGER N,METH,MITER,INDEX,IWK(1),IER,STEP 000080
      REAL*8 X,H,Y(5),XEND,TOL,WK 000090
      EXTERNAL YDOT, FCNJ, DGEAR 000100
      COMMON/CONST/E,K,EPS,TI,EF,EFINF,NINF,DI,DE,VINF,C1,K1,A 000110
C                                           000120
C-----Constants 000121
C C1 and K1 are constants describing the ionization coefficient. 000122
C They are taken from the data plotted in Figure 8. The 000123
C coefficient is equal to the nondimensionalized electric potential 000124
C raised to the K1 power and then multiplied by C1. 000125
C In this way, the ionization coefficient is allowed to vary in 000126
C proportion to the strength of the electric potential. 000127
C----- 000128
      WRITE(*,*)'Input value for C1 (format 6E3):' 000129
      READ(*,*)C1 000130
      WRITE(*,*)C1 000131
      WRITE(*,*)'Input value for K1 (format 6E3):' 000132
      READ(*,*)K1 000133
      WRITE(*,*)K1 000134
C Initial conditions for species number density, electric potential 000135
C and species current density are now input (ni,ne,E,je,ji). 000136
C----- 000137
      WRITE(*,*)'Input values for y(1) through y(5) (format 5(6E3)):' 000138
      READ(*,*)y(1),y(2),y(3),y(4),y(5) 000139
      WRITE(*,*)y(1),y(2),y(3),y(4),y(5) 000140
C Following constants are for plasma described in Reference 21 000141
C (6,000 K, Init E=20,000 V/m, Final E=1,200 V/m) 000142
      E=1.6E-19 000143
      K=1.38E-23 000150
      EPS=8.854E-12 000160
      TI=6E3 000170
      EF=2E5 000180
      EFINF=1.2E4 000190
      DI=1.724E-4 000200
      DE=1.724E-1 000210
      VIO = 2.E6 000215
      VINF = 4.93E-7 000220
C                                           000230

```

```

C      A is plasma characteristic length which shows potential drop. 000240
C      A = ((EPS*EF)/(E*NINF)) = 1.107E-6 000250
      A = 1.107E-5 000270
      X = 0.01 000280
      XEND = 10. 000290
      H = 1e-6 000300
      TOL = 1E-6 000305
      METH = 2 000310
      MITER = 1 000320
      INDEX = 1 000330
      N=5 000340
      IWK(1) = 5 000350
      WK = 18000. 000360
      IER = 0 000370
      OPEN(UNIT=8, FILE='SHEATH.DAT', STATUS='UNKNOWN') 000380
      CALL DGEAR2(N, YDOT, FCNJ, X, H, Y, XEND, TOL, METH, MITER, INDEX, IWK, WK,
+IER, STEP) 000390
      DO 3 I=0, N 000400
DO 2 J=0, 100 000410
      WRITE(*,*) J, Y(I) 000420
      WRITE(8,1) J, Y(I) 000430
1  FORMAT(T2, F5.1, 5(5X, D9.2)) 000440
2  CONTINUE 000450
3  CONTINUE 000460
      WRITE(*,*) 'Total Steps = ', STEP, 'Final Step Size = ', H,
+ 'Error Code = ', IER 000470
      CLOSE(UNIT=8) 000480
      STOP 000500
      END 000510

C*****
C DUMMY SUBROUTINE FCNJ
C*****
      SUBROUTINE FCNJ(N,X,Y,PD) 1
      INTEGER N 2
      REAL Y(N), PD(N,N), X 3
      RETURN 4
      END 5

C*****
C SUBROUTINE YDOT
C*****
      SUBROUTINE YDOT(N,X,Y,YPRIME,eprime,eprime2)
      REAL*8 X, Y(5), YPRIME(5), NUI, eprime, eprime2
      REAL E,K,EPS,TI,EF,EFINF,NINF,DI,DE,VINF,C1,K1,A,B1,B2,B3,B4
      COMMON/CONST/E,K,EPS,TI,EF,EFINF,NINF,DI,DE,VIO,VINF,C1,K1,A
      VI = C1 * (Y(3)**K1)
      VIT = VI / VIO
C      Following constants are the bracketed values in Equations 12-16.
C      A is left as a variable.
C-----
C      B1 = ((E*EPS)/(K*TI)) * A
      B1= 3.86E5 * A
C      B2 = ((E*EFINF)/(K*TI)) * A
      B2 = 2.32E4 * A
C      B3 = ((E*NINF)/(EF*EPS)) * A
      B3 = 9.04E5 * A
C      B4 = ((VINF*K*TI)/(E*DE*EFINF)) * A
      B4 = 2.62E-21 * A
C      Alpha = 2-body recombination coefficient (fm. Laser Kinetics
C      Handbook (AFWL-TR-74-216, 1974)) (cm3/sec)
      Alpha = 9.e-8

```

```

C-----
C FIVE FIRST ORDER EQUATIONS - Equations 12-16
C-----
C      Ni
C      YPRIME(1) = (B * Y(1) * Y(3)) - Y(5)
C      No
C      YPRIME(2) = -(B * Y(2) * Y(3)) + Y(4)
C      E
C      YPRIME(3) = B3 * (Y(1) - Y(2))
C      je
C      YPRIME(4) = -B4 * Y(2) * (VIT - (ALPHA * Y(1)))
C      ji
C      YPRIME(5) = B4 * Y(2) * (VIT - (ALPHA * Y(1)))
C
C---Diagnostic Check of first,second derivatives-----
C
C      eprime = y(1) - y(2)
C      eprime2 = yprime(1) - yprime(2)
C
C      RETURN
C      END

```

C	IMSL ROUTINE NAME	- DGEAR	DGEA0010
C			DGEA0020
C	-modified to return # of steps via variable "step" in subroutine call +		
C			DGEA0040
C	COMPUTER	- IBM/DOUBLE	DGEA0050
C			DGEA0060
C	LATEST REVISION	- NOVEMBER 1, 1984	DGEA0070
C			DGEA0080
C	PURPOSE	- DIFFERENTIAL EQUATION SOLVER - VARIABLE ORDER	DGEA0090
C		ADAMS PREDICTOR CORRECTOR METHOD OR	DGEA0100
C		GEARS METHOD	DGEA0110
C			DGEA0120
C	USAGE	- CALL DGEAR (N,FCN,FCNJ,X,H,Y,XEND,TOL,METH,	DGEA0130
C		MITER,INDEX,IWK,WK,IER)	DGEA0140
C			DGEA0150
C	ARGUMENTS	N - INPUT NUMBER OF FIRST-ORDER DIFFERENTIAL	DGEA0160
C		EQUATIONS.	DGEA0170
C	FCN	- NAME OF SUBROUTINE FOR EVALUATING FUNCTIONS.	DGEA0180
C		(INPUT)	DGEA0190
C		THE SUBROUTINE ITSELF MUST ALSO BE PROVIDED	DGEA0200
C		BY THE USER AND IT SHOULD BE OF THE	DGEA0210
C		FOLLOWING FORM	DGEA0220
C		SUBROUTINE FCN (N,X,Y,YPRIME)	DGEA0230
C		REAL X,Y(N),YPRIME(N)	DGEA0240
C		.	DGEA0250
C		.	DGEA0260
C		.	DGEA0270
C		FCN SHOULD EVALUATE YPRIME(1),...,YPRIME(N)	DGEA0280
C		GIVEN N,X, AND Y(1),...,Y(N). YPRIME(I)	DGEA0290
C		IS THE FIRST DERIVATIVE OF Y(I) WITH	DGEA0300
C		RESPECT TO X.	DGEA0310
C		FCN MUST APPEAR IN AN EXTERNAL STATEMENT IN	DGEA0320
C		THE CALLING PROGRAM AND N,X,Y(1),...,Y(N)	DGEA0330
C		MUST NOT BE ALTERED BY FCN.	DGEA0340
C	FCNJ	- NAME OF THE SUBROUTINE FOR COMPUTING THE	DGEA0350
C		JACOBIAN MATRIX OF PARTIAL DERIVATIVES.	DGEA0360
C		(INPUT)	DGEA0370
C		THE SUBROUTINE ITSELF MUST ALSO BE PROVIDED	DGEA0380
C		BY THE USER.	DGEA0390
C		IF MITER=1 IT SHOULD BE OF THE FOLLOWING	DGEA0400
C		FORM	DGEA0410
C		SUBROUTINE FCNJ (N,X,Y,PD)	DGEA0420
C		REAL X,Y(N),PD(N,N)	DGEA0430
C		.	DGEA0440
C		.	DGEA0450
C		FCNJ MUST EVALUATE PD(I,J), THE PARTIAL	DGEA0460
C		DERIVATIVE OF YPRIME(I) WITH RESPECT TO	DGEA0470
C		Y(J), FOR I=1,N AND J=1,N.	DGEA0480
C		IF MITER= -1 IT SHOULD BE OF THE FOLLOWING	DGEA0490
C		FORM	DGEA0500
C		SUBROUTINE FCNJ (N,X,Y,PD)	DGEA0510
C		REAL X,Y(N),PD(1)	DGEA0520
C		.	DGEA0530
C		.	DGEA0540
C		FCNJ MUST EVALUATE PD IN BAND STORAGE MODE.	DGEA0550
C		THAT IS, PD(N*(J-I+NLC)+I) IS THE PARTIAL	DGEA0560
C		DERIVATIVE OF YPRIME(I) WITH RESPECT TO	DGEA0570
C		Y(J). NLC IS THE NUMBER OF LOWER	DGEA0580
C		CODIAGONALS FOR THE BAND MATRIX.	DGEA0590
C		FCNJ MUST APPEAR IN AN EXTERNAL STATEMENT	INDGEA0600
C		THE CALLING PROGRAM AND N,X,Y(1),...,Y(N)	DGEA0610

	MUST NOT BE ALTERED BY FCNJ.	DGEA0620
	FCNJ IS USED ONLY IF MITER IS EQUAL TO	DGEA0630
	1 OR -1. OTHERWISE A DUMMY ROUTINE CAN	DGEA0640
	BE SUBSTITUTED. SEE REMARK 1.	DGEA0650
X	- INDEPENDENT VARIABLE. (INPUT AND OUTPUT)	DGEA0660
	ON INPUT, X SUPPLIES THE INITIAL VALUE	DGEA0670
	AND IS USED ONLY ON THE FIRST CALL.	DGEA0680
	ON OUTPUT, X IS REPLACED WITH THE CURRENT	DGEA0690
	VALUE OF THE INDEPENDENT VARIABLE AT WHICH	DGEA0700
	INTEGRATION HAS BEEN COMPLETED.	DGEA0710
H	- INPUT/OUTPUT.	DGEA0720
	ON INPUT, H CONTAINS THE NEXT STEP SIZE IN	DGEA0730
	X. H IS USED ONLY ON THE FIRST CALL.	DGEA0740
	ON OUTPUT, H CONTAINS THE STEP SIZE USED	DGEA0750
	LAST, WHETHER SUCCESSFULLY OR NOT.	DGEA0760
Y	- DEPENDENT VARIABLES, VECTOR OF LENGTH N.	DGEA0770
	(INPUT AND OUTPUT)	DGEA0780
	ON INPUT, Y(1), ..., Y(N) SUPPLY INITIAL	DGEA0790
	VALUES.	DGEA0800
	ON OUTPUT, Y(1), ..., Y(N) ARE REPLACED WITH	DGEA0810
	A COMPUTED VALUE AT XEND.	DGEA0820
XEND	- INPUT VALUE OF X AT WHICH SOLUTION IS DESIRED	DGEA0830
	NEXT. INTEGRATION WILL NORMALLY GO	DGEA0840
	BEYOND XEND AND THE ROUTINE WILL INTERPOLATE	DGEA0850
	TO X = XEND.	DGEA0860
	NOTE THAT (X-XEND)*H MUST BE LESS THAN	DGEA0870
	ZERO (X AND H AS SPECIFIED ON INPUT).	DGEA0880
TOL	- INPUT RELATIVE ERROR BOUND. TOL MUST BE	DGEA0890
	GREATER THAN ZERO. TOL IS USED ONLY ON THE	DGEA0900
	FIRST CALL UNLESS INDEX IS EQUAL TO -1.	DGEA0910
	TOL SHOULD BE AT LEAST AN ORDER OF	DGEA0920
	MAGNITUDE LARGER THAN THE UNIT ROUNDOFF	DGEA0930
	BUT GENERALLY NOT LARGER THAN .001.	DGEA0940
	SINGLE STEP ERROR ESTIMATES DIVIDED BY	DGEA0950
	YMAX(I) WILL BE KEPT LESS THAN TOL IN	DGEA0960
	ROOT-MEAN-SQUARE NORM (EUCLIDEAN NORM	DGEA0970
	DIVIDED BY \sqrt{N}). THE VECTOR YMAX OF	DGEA0980
	WEIGHTS IS COMPUTED INTERNALLY AND STORED	DGEA0990
	IN WORK VECTOR WK. INITIALLY YMAX(I) IS	DGEA1000
	THE ABSOLUTE VALUE OF Y(I), WITH A DEFAULT	DGEA1010
	VALUE OF ONE IF Y(I) IS EQUAL TO ZERO.	DGEA1020
	THEREAFTER, YMAX(I) IS THE LARGEST VALUE	DGEA1030
	OF THE ABSOLUTE VALUE OF Y(I) SEEN SO FAR,	DGEA1040
	OR THE INITIAL VALUE OF YMAX(I) IF THAT IS	DGEA1050
	LARGER.	DGEA1060
METH	- INPUT BASIC METHOD INDICATOR.	DGEA1070
	USED ONLY ON THE FIRST CALL UNLESS INDEX IS	DGEA1080
	EQUAL TO -1.	DGEA1090
	METH = 1, IMPLIES THAT THE ADAMS METHOD IS	DGEA1100
	TO BE USED.	DGEA1110
	METH = 2, IMPLIES THAT THE STIFF METHODS OF	DGEA1120
	GEAR, OR THE BACKWARD DIFFERENTIATION	DGEA1130
	FORMULAE ARE TO BE USED.	DGEA1140
MITER	- INPUT ITERATION METHOD INDICATOR.	DGEA1150
	MITER = 0, IMPLIES THAT FUNCTIONAL	DGEA1160
	ITERATION IS USED. NO PARTIAL	DGEA1170
	DERIVATIVES ARE NEEDED. A DUMMY FCNJ	DGEA1180
	CAN BE USED.	DGEA1190
	MITER = 1, IMPLIES THAT THE CHORD METHOD	DGEA1200
	IS USED WITH AN ANALYTIC JACOBIAN. FOR	DGEA1210
	THIS METHOD, THE USER SUPPLIES	DGEA1220
	SUBROUTINE FCNJ.	DGEA1230

	MITER = 2, IMPLIES THAT THE CHORD METHOD	DGEA1240
	IS USED WITH THE JACOBIAN CALCULATED	DGEA1250
	INTERIALLY BY FINITE DIFFERENCES.	DGEA1260
	A DUMMY FCNJ CAN BE USED.	DGEA1270
	MITER = 3, IMPLIES THAT THE CHORD METHOD	DGEA1280
	IS USED WITH THE JACOBIAN REPLACED BY	DGEA1290
	A DIAGONAL APPROXIMATION BASED ON A	DGEA1300
	DIRECTIONAL DERIVATIVE.	DGEA1310
	A DUMMY FCNJ CAN BE USED.	DGEA1320
	MITER = -1 OR -2, IMPLIES USE THE SAME	DGEA1330
	METHOD AS FOR MITER= 1 OR 2, RESPECTIVELY,	DGEA1340
	BUT USING A Banded JACOBIAN MATRIX. IN	DGEA1350
	THESE TWO CASES BANDWIDTH INFORMATION	DGEA1360
	MUST BE PASSED TO DGEAR THROUGH THE	DGEA1370
	COMMON BLOCK	DGEA1380
	COMMON /DBAND/ NLC,NUC	DGEA1390
	WHERE NLC=NUMBER OF LOWER CODIAGONALS	DGEA1400
	NUC=NUMBER OF UPPER CODIAGONALS	DGEA1410
INDEX	- INPUT AND OUTPUT PARAMETER USED TO INDICATE	DGEA1420
	THE TYPE OF CALL TO THE SUBROUTINE. ON	DGEA1430
	OUTPUT INDEX IS RESET TO 0 IF INTEGRATION	DGEA1440
	WAS SUCCESSFUL. OTHERWISE, THE VALUE OF	DGEA1450
	INDEX IS UNCHANGED.	DGEA1460
	ON INPUT, INDEX = 1, IMPLIES THAT THIS IS THE	DGEA1470
	FIRST CALL FOR THIS PROBLEM.	DGEA1480
	ON INPUT, INDEX = 0, IMPLIES THAT THIS IS NOT	DGEA1490
	THE FIRST CALL FOR THIS PROBLEM.	DGEA1500
	ON INPUT, INDEX = -1, IMPLIES THAT THIS IS NOT	DGEA1510
	THE FIRST CALL FOR THIS PROBLEM, AND THE	DGEA1520
	USER HAS RESET TOL.	DGEA1530
	ON INPUT, INDEX = 2, IMPLIES THAT THIS IS NOT	DGEA1540
	THE FIRST CALL FOR THIS PROBLEM. INTEGRATION	DGEA1550
	IS TO CONTINUE AND XEND IS TO BE HIT EXACTLY	DGEA1560
	(NO INTERPOLATION IS DONE). THIS VALUE OF	DGEA1570
	INDEX ASSUMES THAT XEND IS BEYOND THE	DGEA1580
	CURRENT VALUE OF X.	DGEA1590
	ON INPUT, INDEX = 3, IMPLIES THAT THIS IS NOT	DGEA1600
	THE FIRST CALL FOR THIS PROBLEM. INTEGRATION	DGEA1610
	IS TO CONTINUE AND CONTROL IS TO BE RETURNED	DGEA1620
	TO THE CALLING PROGRAM AFTER ONE STEP. XEND	DGEA1630
	IS IGNORED.	DGEA1640
IWK	- INTEGER WORK VECTOR OF LENGTH N. USED ONLY IF	DGEA1650
	MITER = 1 OR 2	DGEA1660
WK	- REAL WORK VECTOR OF LENGTH 4*N+NMETH+NMITER.	DGEA1670
	THE VALUE OF NMETH DEPENDS ON THE VALUE OF	DGEA1680
	METH.	DGEA1690
	IF METH IS EQUAL TO 1,	DGEA1700
	NMETH IS EQUAL TO N*13.	DGEA1710
	IF METH IS EQUAL TO 2,	DGEA1720
	NMETH IS EQUAL TO N*6.	DGEA1730
	THE VALUE OF NMITER DEPENDS ON THE VALUE OF	DGEA1740
	MITER.	DGEA1750
	IF MITER IS EQUAL TO 1 OR 2,	DGEA1760
	NMITER IS EQUAL TO N*(N+1)	DGEA1770
	IF MITER IS EQUAL TO -1 OR -2,	DGEA1780
	NMITER IS EQUAL TO (2*NLC+NUC+3)*N	DGEA1790
	WHERE NLC=NUMBER OF LOWER CODIAGONALS	DGEA1800
	NUC=NUMBER OF UPPER CODIAGONALS	DGEA1810
	IF MITER IS EQUAL TO 3,	DGEA1820
	NMITER IS EQUAL TO N.	DGEA1830
	IF MITER IS EQUAL TO 0,	DGEA1840
	NMITER IS EQUAL TO 1.	DGEA1850

COMPARISON OF THE JACOBIAN WITH THAT GENERATED WITH
 ABS(MITER)=2. IF THE JACOBIAN MATRIX IS SIGNIFICANTLY
 DIAGONALLY DOMINANT, THEN THE OPTION MITER = 3 IS
 LIKELY TO BE NEARLY AS EFFECTIVE AS ABS(MITER)=1 OR 2,
 AND WILL SAVE CONSIDERABLE STORAGE AND RUN TIME.
 IT IS POSSIBLE, AND POTENTIALLY QUITE DESIRABLE, TO
 USE DIFFERENT VALUES OF METH AND MITER IN DIFFERENT
 SUBINTERVALS OF THE PROBLEM. FOR EXAMPLE, IF THE
 PROBLEM IS NON-STIFF INITIALLY AND STIFF LATER,
 METH = 1 AND MITER = 0 MIGHT BE SET INITIALLY, AND
 METH = 2 AND MITER = 1 LATER.

5. THE INITIAL VALUE OF THE STEP SIZE, H, SHOULD BE
 CHOSEN CONSIDERABLY SMALLER THAN THE AVERAGE VALUE
 EXPECTED FOR THE PROBLEM, AS THE FIRST-ORDER METHOD
 WITH WHICH DGEAR BEGINS IS NOT GENERALLY THE MOST
 EFFICIENT ONE. HOWEVER, FOR THE FIRST STEP, AS FOR
 EVERY STEP, DGEAR TESTS FOR THE POSSIBILITY THAT
 THE STEP SIZE WAS TOO LARGE TO PASS THE ERROR TEST
 (BASED ON TOL), AND IF SO ADJUSTS THE STEP SIZE
 DOWN AUTOMATICALLY. THIS DOWNWARD ADJUSTMENT, IF
 ANY, IS NOTED BY IER HAVING THE VALUES 66 OR 67,
 AND SUBSEQUENT RUNS ON THE SAME OR SIMILAR PROBLEM
 SHOULD BE STARTED WITH AN APPROPRIATELY SMALLER
 VALUE OF H.

6. SOME OF THE VALUES OF INTEREST LOCATED IN THE
 COMMON BLOCK /GEAR/ ARE

- A. HUSED, THE STEP SIZE H LAST USED SUCCESSFULLY
 (DUMMY(8))
- B. NUSED, THE ORDER LAST USED SUCCESSFULLY
 (IDUMMY(6))
- C. NSTEP, THE CUMULATIVE NUMBER OF STEPS TAKEN
 (IDUMMY(7))
- D. NFE, THE CUMULATIVE NUMBER OF FCN EVALUATIONS
 (IDUMMY(8))
- E. NJE, THE CUMULATIVE NUMBER OF JACOBIAN
 EVALUATIONS, AND HENCE ALSO OF MATRIX LU
 DECOMPOSITIONS (IDUMMY(9))

7. THE NORMAL USAGE OF DGEAR MAY BE SUMMARIZED AS FOLLOWS

- A. SET THE INITIAL VALUES IN Y.
- B. SET N, X, H, TOL, METH, AND MITER.
- C. SET XEND TO THE FIRST OUTPUT POINT, AND INDEX TO 1.
- D. CALL DGEAR
- E. EXIT IF IER IS GREATER THAN 128.
- F. OTHERWISE, DO DESIRED OUTPUT OF Y.
- G. EXIT IF THE PROBLEM IS FINISHED.
- H. OTHERWISE, RESET XEND TO THE NEXT OUTPUT POINT, AND
 RETURN TO STEP D.

8. THE ERROR WHICH IS CONTROLLED BY WAY OF THE PARAMETER
 TOL IS AN ESTIMATE OF THE LOCAL TRUNCATION ERROR, THAT
 IS, THE ERROR COMMITTED ON TAKING A SINGLE STEP WITH
 THE METHOD, STARTING WITH DATA REGARDED AS EXACT. THIS
 IS TO BE DISTINGUISHED FROM THE GLOBAL TRUNCATION
 ERROR, WHICH IS THE ERROR IN ANY GIVEN COMPUTED VALUE
 $\frac{1}{2} Y(X)$ AS A RESULT OF THE LOCAL TRUNCATION ERRORS
 FROM ALL STEPS TAKEN TO OBTAIN $Y(X)$. THE LATTER ERROR
 ACCUMULATES IN A NON-TRIVIAL WAY FROM THE LOCAL
 ERRORS, AND IS NEITHER ESTIMATED NOR CONTROLLED BY
 THE ROUTINE. SINCE IT IS USUALLY THE GLOBAL ERROR THAT
 A USER WANTS TO HAVE UNDER CONTROL, SOME
 EXPERIMENTATION MAY BE NECESSARY TO GET THE RIGHT
 VALUE OF TOL TO ACHIEVE THE USER'S NEEDS. IF THE
 PROBLEM IS MATHEMATICALLY STABLE, AND THE METHOD USED

DGEA3090
 DGEA3100
 DGEA3110
 DGEA3120
 DGEA3130
 DGEA3140
 DGEA3150
 DGEA3160
 DGEA3170
 DGEA3180
 DGEA3190
 DGEA3200
 DGEA3210
 DGEA3220
 DGEA3230
 DGEA3240
 DGEA3250
 DGEA3260
 DGEA3270
 DGEA3280
 DGEA3290
 DGEA3300
 DGEA3310
 DGEA3320
 DGEA3330
 DGEA3340
 DGEA3350
 DGEA3360
 DGEA3370
 DGEA3380
 DGEA3390
 DGEA3400
 DGEA3410
 DGEA3420
 DGEA3430
 DGEA3440
 DGEA3450
 DGEA3460
 DGEA3470
 DGEA3480
 DGEA3490
 DGEA3500
 DGEA3510
 DGEA3520
 DGEA3530
 DGEA3540
 DGEA3550
 DGEA3560
 DGEA3570
 DGEA3580
 DGEA3590
 DGEA3600
 DGEA3610
 DGEA3620
 DGEA3630
 DGEA3640
 DGEA3650
 DGEA3660
 DGEA3670
 DGEA3680
 DGEA3690
 DGEA3700

DGRA3710
DGRA3720
DGRA3730

- DGEA3740
DGEA3750
DGEA3760
DGEA3770
DGEA3780
DGEA3790
DGEA3800
DGEA3810
DGEA3820
DGEA3830
DGEA3840
DGEA3850
DGEA3860
DGEA3870
DGEA3880
DGEA3890
DGEA3900
DGEA3910
DGEA3920
DGEA3930
DGEA3940
DGEA3950
DGEA3960
DGEA3970
DGEA3980

- 1984 BY IMSL, INC. ALL RIGHTS RESERVED.

- IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN APPLIED TO THIS CODE. NO OTHER WARRANTY, EXPRESSED OR IMPLIED, IS APPLICABLE.

DGEA4000
DGEA4010
DGEA4020
DGEA4030
DGEA4040
DGEA4050
- - DGEA4060
DGEA4070
DGEA4080
+
DGEA4100
+
DGEA4120
DGEA4130
DGEA4140
D, DGEA4150
DGEA4160
DGEA4170
DGEA4180
DGEA4190
DGEA4200
DGEA4210
DGEA4220
DGEA4230
, DGEA4240
DGEA4250
DGEA4260
DGEA4270
DGEA4280
DGEA4290
DGEA4300
DGEA4310
DGEA4320

COMPUTE WORK VECTOR INDICIES

C	25 IF ((T+HH).EQ.T) KER = 33	DGEA4950
	write(*,*) 'error code = ',ker	DGEA4960
C	30 NN = NO	+ DGEA4970
	step = step + 1	DGEA4980
	write(*,*) 'step = ',step	+
	CALL DGRST (FCN,FCNJ,WK(NY+1),WK,WK(MERROR+1),WK(NSAVE1+1),	+ DGEA4990
	1 WK(NSAVE2+1),WK(NPW+1),WK(NEQUIL+1),IWK,NN,step)	+
C	KGO = 1-KFLAG	DGEA5010
	GO TO (35,55,70,80), KGO	DGEA5020
C	KFLAG = 0, -1, -2, -3	DGEA5030
C	35 CONTINUE	DGEA5040
C		DGEA5050
C		DGEA5060
C		DGEA5070
C		DGEA5080
C		DGEA5090
C		DGEA5100
C		DGEA5110
C		DGEA5120
C		DGEA5130
C		DGEA5140
C		DGEA5150
C		DGEA5160
C		DGEA5170
C		DGEA5180
C		DGEA5190
C		DGEA5200
C		DGEA5210
C		DGEA5220
C		DGEA5230
C		DGEA5240
C		DGEA5250
C		DGEA5260
C		DGEA5270
C		DGEA5280
C		DGEA5290
	D = 0.D0	DGEA5300
	DO 40 I=1,N	DGEA5310
	AYI = DABS(WK(NY+I))	DGEA5320
	WK(I) = DMAX1(WK(I),AYI)	DGEA5330
40	D = D+(AYI/WK(I))**2	DGEA5340
	D = D*(UROUND/TOL)**2	DGEA5350
	DN = N	DGEA5360
	IF (D.GT.DN) GO TO 75	DGEA5370
	IF (INDEX.EQ.3) GO TO 95	DGEA5380
	IF (INDEX.EQ.2) GO TO 50	DGEA5390
45	IF ((T-XEND)*HH.LT.0.D0) GO TO 25	DGEA5400
	NN = NO	DGEA5410
	CALL DGRIN (XEND,WK(NY+1),NN,Y)	DGEA5420
	X = XEND	DGEA5430
	GO TO 105	DGEA5440
50	IF (((T+HH)-XEND)*HH.LE.0.D0) GO TO 25	DGEA5450
	IF (DABS(T-XEND).LE.UROUND*DMAX1(10.D0*DABS(T),HMAX)) GO TO 95	DGEA5460
	IF ((T-XEND)*HH.GE.0.D0) GO TO 95	DGEA5470
	HH = (XEND-T)*(1.D0-4.D0*UROUND)	DGEA5480
	JSTART = -1	DGEA5490
	GO TO 25	DGEA5500
C		DGEA5510
C		DGEA5520
C		DGEA5530
	ON AN ERROR RETURN FROM INTEGRATOR,	
	AN IMMEDIATE RETURN OCCURS IF	
	KFLAG = -2, AND RECOVERY ATTEMPTS	

C	ARE MADE OTHERWISE. TO RECOVER, HH	DGEA5540
C	AND HMIN ARE REDUCED BY A FACTOR	DGEA5550
C	OF .1 UP TO 10 TIMES BEFORE GIVING	DGEA5560
C	UP.	DGEA5570
	55 JER = 66	DGEA5580
	60 IF (NHCUT.EQ.10) GO TO 65	DGEA5590
	NHCUT = NHCUT+1	DGEA5600
	HMIN = HMIN*.1D0	DGEA5610
	HH = HH*.1D0	DGEA5620
	JSTART = -1	DGEA5630
	GO TO 25	DGEA5640
C		DGEA5650
	65 IF (JER.EQ.66) JER = 132	DGEA5660
	IF (JER.EQ.67) JER = 133	DGEA5670
	GO TO 95	DGEA5680
C		DGEA5690
	70 JER = 134	DGEA5700
	GO TO 95	DGEA5710
C		DGEA5720
	75 JER = 134	DGEA5730
	KFLAG = -2	DGEA5740
	GO TO 95	DGEA5750
C		DGEA5760
	80 JER = 67	DGEA5770
	GO TO 60	DGEA5780
C		DGEA5790
	85 JER = 135	DGEA5800
	GO TO 110	DGEA5810
C		DGEA5820
	90 JER = 136	DGEA5830
	NN = NO	DGEA5840
	CALL DGRIN (XEND,WK(NY+1),NN,Y)	DGEA5850
	X = XEND	DGEA5860
	GO TO 110	DGEA5870
C		DGEA5880
	95 X = T	DGEA5890
	DO 100 I=1,N	DGEA5900
	100 Y(I) = WK(NY+I)	DGEA5910
	105 IF (JER.LT.128) INDEX = KFLAG	DGEA5920
	TOUTP = X	DGEA5930
	IF (KFLAG.EQ.0) H = HUSED	DGEA5940
	IF (KFLAG.NE.0) H = HH	DGEA5950
	110 IER = MAX0(KER,JER)	DGEA5960
	9000 CONTINUE	DGEA5970
	IF (KER.NE.0.AND.JER.LT.128) CALL UERTST (KER,6HDGEAR)	DGEA5980
	IF (JER.NE.0) CALL UERTST (JER,6HDGEAR)	DGEA5990
	9005 RETURN	DGEA6000
	END	DGEA6010

C	IMSL ROUTINE NAME	- DGRST	DGRS0010
C			DGRS0020
C	-modified to print sheath and diagnostic output to files "sheatha.dat" +		
C	and "diag.dat"		+
C	COMPUTER	- IBM/DOUBLE	DGRS0050
C			DGRS0060
C	LATEST REVISION	- JUNE 1, 1982	DGRS0070
C			DGRS0080
C	PURPOSE	- NUCLEUS CALLED ONLY BY IMSL SUBROUTINE DGEAR	DGRS0090
C			DGRS0100
C	PRECISION/HARDWARE	- SINGLE AND DOUBLE/H32	DGRS0110
C		- SINGLE/H36,H48,H60	DGRS0120
C			DGRS0130
C	REQD. IMSL ROUTINES	- DGRCS,DGRPS,LUDATF,LUELMF,LEQT1B,UERTST, UGETIO	DGRS0140
C			DGRS0150
C			DGRS0160
C	NOTATION	- INFORMATION ON SPECIAL NOTATION AND CONVENTIONS IS AVAILABLE IN THE MANUAL INTRODUCTION OR THROUGH IMSL ROUTINE UHELP	DGRS0170
C			DGRS0180
C			DGRS0190
C			DGRS0200
C	COPYRIGHT	- 1982 BY IMSL, INC. ALL RIGHTS RESERVED.	DGRS0210
C			DGRS0220
C	WARRANTY	- IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN APPLIED TO THIS CODE. NO OTHER WARRANTY, EXPRESSED OR IMPLIED, IS APPLICABLE.	DGRS0230
C			DGRS0240
C			DGRS0250
C			DGRS0260
C	-----		DGRS0270
C	SUBROUTINE DGRST	(FCN,FCNJ,Y,YMAX,ERROR,SAVE1,SAVE2,PW,EQUIL, 1 IPIV,N0,step)	DGRS0280
C			DGRS0290
C		SPECIFICATIONS FOR ARGUMENTS	+
C	INTEGER	IPIV(1),N0	DGRS0310
C	DOUBLE PRECISION	Y(N0,1),YMAX(1),ERROR(1),SAVE1(1),SAVE2(1), 1 PW(1),EQUIL(1),eprime,eprime(2)	DGRS0320
C			DGRS0330
C		SPECIFICATIONS FOR LOCAL VARIABLES	+
C	INTEGER	N,MF,KFLAG,JSTART,NQUSED,NSTEP,NFE,NJE,NSQ, 1 I,METH,MITER,NQ,L,IDOUB,MFOLD,NOLD,IRET,MEO, 2 MIO,IWEVAL,MAXDER,LMAX,IREDO,J,NSTEPJ,J1,J2, 3 M,IER,NEWQ,NPW,NERROR,NSAVE1,NSAVE2,NEQUIL,NY, 4 MITER1,IDUMMY(2),NLC,NUC,NWK,JER	DGRS0350
C			DGRS0360
C	REAL	TQ(4)	DGRS0370
C	DOUBLE PRECISION	T,H,HMIN,HMAX,EPS,UROUND,HUSED,EL(13),OLDLO, 1 TOLD,RMAX,RC,CRATE,EPSOLD,HOLD,FN,EDN,E,EUP, 2 BND,RH,R1,CON,R,HL0,R0,D,PHL0,PR3,D1,ENQ3,ENQ2, 3 PR2,PR1,ENQ1,EPSJ,DUMMY,tcum	DGRS0380
C			DGRS0390
C	EXTERNAL	FCN,FCNJ	DGRS0400
C	COMMON /DBAND/	NLC,NUC	DGRS0410
C	COMMON /GEAR/	T,H,HMIN,HMAX,EPS,UROUND,EPSJ,HUSED, 1 EL,OLDLO,TOLD,RMAX,RC,CRATE,EPSOLD,HOLD,FN, 2 EDN,E,EUP,BND,RH,R1,R,HL0,R0,D,PHL0,PR3,D1, 3 ENQ3,ENQ2,PR2,PR1,ENQ1,DUMMY,TQ, 4 N,MF,KFLAG,JSTART,NSQ,NQUSED,NSTEP,NFE,NJE, 5 NPW,NERROR,NSAVE1,NSAVE2,NEQUIL,NY, 6 I,METH,MITER,NQ,L,IDOUB,MFOLD,NOLD,IRET,MEO, 7 MIO,IWEVAL,MAXDER,LMAX,IREDO,J,NSTEPJ,J1,J2, 8 M,NEWQ,IDUMMY	DGRS0420
C			DGRS0430
C			DGRS0440
C			+
C			DGRS0460
C			DGRS0470
C			DGRS0480
C			DGRS0490
C			DGRS0500
C			DGRS0510
C			DGRS0520
C			DGRS0530
C			DGRS0540
C			DGRS0550
C			DGRS0560
C		FIRST EXECUTABLE STATEMENT	DGRS0570
C	open(unit=8,file='sheatha.dat',status='unknown')		+
C	open(unit=9,file='diag.dat',status='unknown')		+
C	KFLAG = 0		DGRS0580
C	TO: D = T		DGRS0590
C		THIS ROUTINE PERFORMS ONE STEP OF	DGRS0600

C
C
C

IF (JSTART.GT.0) GO TO 50
IF (JSTART.NE.0) GO TO 10

C
C
C
C
C
C
C
C
C
C

THE INTEGRATION OF AN INITIAL
VALUE PROBLEM FOR A SYSTEM OF
ORDINARY DIFFERENTIAL EQUATIONS.

ON THE FIRST CALL, THE ORDER IS SET
TO 1 AND THE INITIAL YDOT IS
CALCULATED. RMAX IS THE MAXIMUM
RATIO BY WHICH H CAN BE INCREASED
IN A SINGLE STEP. IT IS INITIALLY
1.E4 TO COMPENSATE FOR THE SMALL
INITIAL H, BUT THEN IS NORMALLY
EQUAL TO 10. IF A FAILURE OCCURS
(IN CORRECTOR CONVERGENCE OR ERROR
TEST), RMAX IS SET AT 2 FOR THE
NEXT INCREASE.

DGRS0610
DGRS0620
DGRS0630
DGRS0640
DGRS0650
DGRS0660
DGRS0670
DGRS0680
DGRS0690
DGRS0700
DGRS0710
DGRS0720
DGRS0730
DGRS0740
DGRS0750
DGRS0760

CALL FCN (N,T,Y,SAVE1,eprime,eprime2)
DO 5 I=1,N
5 Y(I,2) = H*SAVE1(I)
METH = MF/10
MITER = MF-10*METH
NQ = 1
L = 2
IDOUN = 3
RMAX = 1.D4
RC = 0.D0
CRATE = 1.D0
HOLD = H
MFOLD = MF
NSTEP = 0
NSTEPJ = 0
NFE = 1
NJE = 0
IRET = 3
GO TO 15

+
DGRS0780
DGRS0790
DGRS0800
DGRS0810
DGRS0820
DGRS0830
DGRS0840
DGRS0850
DGRS0860
DGRS0870
DGRS0880
DGRS0890
DGRS0900
DGRS0910
DGRS0920
DGRS0930
DGRS0940
DGRS0950

C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C

IF THE CALLER HAS CHANGED METH,
DGRCS IS CALLED TO SET THE
COEFFICIENTS OF THE METHOD. IF THE
CALLER HAS CHANGED N, EPS, OR
METH, THE CONSTANTS E, EDN, EUP,
AND BND MUST BE RESET. E IS A
COMPARISON FOR ERRORS OF THE
CURRENT ORDER NQ. EUP IS TO TEST
FOR INCREASING THE ORDER, EDN FOR
DECREASING THE ORDER. BND IS USED
TO TEST FOR CONVERGENCE OF THE
CORRECTOR ITERATES. IF THE CALLER
HAS CHANGED H, Y MUST BE RESCALED.
IF H OR METH HAS BEEN CHANGED,
IDOUN IS RESET TO L + 1 TO PREVENT
FURTHER CHANGES IN H FOR THAT MANY
STEPS.

DGRS0960
DGRS0970
DGRS0980
DGRS0990
DGRS1000
DGRS1010
DGRS1020
DGRS1030
DGRS1040
DGRS1050
DGRS1060
DGRS1070
DGRS1080
DGRS1090
DGRS1100
DGRS1110
DGRS1120
DGRS1130
DGRS1140
DGRS1150
DGRS1160
DGRS1170
DGRS1180
DGRS1190
DGRS1200
DGRS1210
DGRS1220

10 IF (MF.EQ.MFOLD) GO TO 25
MEO = METH
MIO = MITER
METH = MF/10
MITER = MF-10*METH
MFOLD = MF
IF (MITER.NE.MIO) IWEVAL = MITER
IF (METH.EQ.MEO) GO TO 25
IDOUN = L+1
IRET = 1

15	CALL DGRCS (METH,NQ,EL,TQ,MAXDER)	DGRS1230
	LMAX = MAXDER+1	DGRS1240
	RC = RC*EL(1)/OLDLO	DGRS1250
	OLDLO = EL(1)	DGRS1260
20	FN = N	DGRS1270
	EDN = FN*(TQ(1)*EPS)**2	DGRS1280
	E = FN*(TQ(2)*EPS)**2	DGRS1290
	EUP = FN*(TQ(3)*EPS)**2	DGRS1300
	BND = FN*(TQ(4)*EPS)**2	DGRS1310
	EPSOLD = EPS	DGRS1320
	NOLD = N	DGRS1330
	GO TO (30,35,50), IRET	DGRS1340
25	IF ((EPS.EQ.EPSOLD).AND.(N.EQ.NOLD)) GO TO 30	DGRS1350
	IF (N.EQ.NOLD) IWEVAL = MITER	DGRS1360
	IRET = 1	DGRS1370
	GO TO 20	DGRS1380
30	IF (H.EQ.HOLD) GO TO 50	DGRS1390
	RH = H/HOLD	DGRS1400
	H = HOLD	DGRS1410
	IREDO = 3	DGRS1420
	GO TO 40	DGRS1430
35	RH = DMAX1(RH,HMIN/DABS(H))	DGRS1440
40	RH = DMIN1(RH,HMAX/DABS(H),RMAX)	DGRS1450
	R1 = 1.D0	DGRS1460
	DO 45 J=2,L	DGRS1470
	R1 = R1*RH	DGRS1480
	DO 45 I=1,N	DGRS1490
45	Y(I,J) = Y(I,J)*R1	DGRS1500
	H = H*RH	DGRS1510
	RC = RC*RH	DGRS1520
	IDOUB = L+1	DGRS1530
	IF (IREDO.EQ.0) GO TO 285	DGRS1540
		DGRS1550
		DGRS1560
		DGRS1570
		DGRS1580
		DGRS1590
		DGRS1600
		DGRS1610
		DGRS1620
		DGRS1630
		DGRS1640
		DGRS1650
		DGRS1660
		DGRS1670
50	IF (DABS(RC-1.D0).GT.0.3D0) IWEVAL = MITER	DGRS1680
	IF (NSTEP.GE.NSTEPJ+20) IWEVAL = MITER	DGRS1690
	T = T+H	DGRS1700
	DO 55 J1=1,NQ	DGRS1710
	DO 55 J2=J1,NQ	DGRS1720
	J = (NQ+J1)-J2	DGRS1730
	DO 55 I=1,N	DGRS1740
55	Y(I,J) = Y(I,J)+Y(I,J+1)	DGRS1750
		DGRS1760
		DGRS1770
		DGRS1780
		DGRS1790
		DGRS1800
		DGRS1810
		DGRS1820
		DGRS1830
		DGRS1840

THIS SECTION COMPUTES THE PREDICTED VALUES BY EFFECTIVELY MULTIPLYING THE Y ARRAY BY THE PASCAL TRIANGLE MATRIX. RC IS THE RATIO OF NEW TO OLD VALUES OF THE COEFFICIENT H*EL(1). WHEN RC DIFFERS FROM 1 BY MORE THAN 30 PERCENT, OR THE CALLER HAS CHANGED MITER, IWEVAL IS SET TO MITER TO FORCE THE PARTIALS TO BE UPDATED, IF PARTIALS ARE USED. IN ANY CASE, THE PARTIALS ARE UPDATED AT LEAST EVERY 20-TH STEP.

UP TO 3 CORRECTOR ITERATIONS ARE TAKEN. A CONVERGENCE TEST IS MADE ON THE R.M.S. NORM OF EACH CORRECTION, USING BND, WHICH IS DEPENDENT ON EPS. THE SUM OF THE CORRECTIONS IS ACCUMULATED IN THE VECTOR ERROR(I). THE Y ARRAY IS NOT ALTERED IN THE CORRECTOR LOOP. THE UPDATED Y VECTOR IS STORED

C		TEMPORARILY IN SAVE1.	DGRS1850
	60 DO 65 I=1,N		DGRS1860
	65 ERROR(I) = 0.D0		DGRS1870
	M = 0		DGRS1880
	CALL FCN (N,T,Y,SAVE2,eprime,eprime2)		+
	NFE = NFE+1		DGRS1900
	IF (IWEVAL.LE.0) GO TO 95		DGRS1910
C		IF INDICATED, THE MATRIX P = I -	DGRS1920
C		H*EL(1)*J IS REEVALUATED BEFORE	DGRS1930
C		STARTING THE CORRECTOR ITERATION.	DGRS1940
C		IWEVAL IS SET TO 0 AS AN INDICATOR	DGRS1950
C		THAT THIS HAS BEEN DONE. IF MITER	DGRS1960
C		= 1 OR 2, P IS COMPUTED AND	DGRS1970
C		PROCESSED IN PSET. IF MITER = 3,	DGRS1980
C		THE MATRIX USED IS P = I -	DGRS1990
C		H*EL(1)*D, WHERE D IS A DIAGONAL	DGRS2000
C		MATRIX.	DGRS2010
	IWEVAL = 0		DGRS2020
	RC = 1.D0		DGRS2030
	NJE = NJE+1		DGRS2040
	NSTEPJ = NSTEP		DGRS2050
	GO TO (75,70,80), MITER		DGRS2060
	70 NFE = NFE+N		DGRS2070
	75 CON = -H*EL(1)		DGRS2080
	MITER1 = MITER		DGRS2090
	CALL DGRPS (FCN,FCNJ,Y,N0,CON,MITER1,YMAX,SAVE1,SAVE2,PW,EQUIL,		DGRS2100
	1 IPIV,IER)		DGRS2110
	IF (IER.NE.0) GO TO 155		DGRS2120
	GO TO 125		DGRS2130
	80 R = EL(1)*.1D0		DGRS2140
	DO 85 I=1,N		DGRS2150
	85 PW(I) = Y(I,1)+R*(H*SAVE2(I)-Y(I,2))		DGRS2160
	CALL FCN (N,T,PW,SAVE1,eprime,eprime2)		+
	NFE = NFE+1		DGRS2180
	HL0 = H*EL(1)		DGRS2190
	DO 90 I=1,N		DGRS2200
	R0 = H*SAVE2(I)-Y(I,2)		DGRS2210
	PW(I) = 1.D0		DGRS2220
	D = .1D0*R0-H*(SAVE1(I)-SAVE2(I))		DGRS2230
	SAVE1(I) = 0.D0		DGRS2240
	IF (DABS(R0).LT.UROUND*YMAX(I)) GO TO 90		DGRS2250
	IF (DABS(D).EQ.0.D0) GO TO 155		DGRS2260
	PW(I) = .1D0*R0/D		DGRS2270
	SAVE1(I) = PW(I)*R0		DGRS2280
	90 CONTINUE		DGRS2290
	GO TO 135		DGRS2300
	95 IF (MITER.NE.0) GO TO (125,125,105), MITER		DGRS2310
C			DGRS2320
C		IN THE CASE OF FUNCTIONAL ITERATION,	DGRS2330
C		UPDATE Y DIRECTLY FROM THE RESULT	DGRS2340
C		OF THE LAST FCN CALL.	DGRS2350
	D = 0.D0		DGRS2360
	DO 100 I=1,N		DGRS2370
	R = H*SAVE2(I)-Y(I,2)		DGRS2380
	D = D+((R-ERROR(I))/YMAX(I))**2		DGRS2390
	SAVE1(I) = Y(I,1)+EL(1)*R		DGRS2400
	100 ERROR(I) = R		DGRS2410
	GO TO 145		DGRS2420
C		IN THE CASE OF THE CHORD METHOD,	DGRS2430
C		COMPUTE THE CORRECTOR ERROR, F SUB	DGRS2440
C		(M), AND SOLVE THE LINEAR SYSTEM	DGRS2450
C		WITH THAT AS RIGHT-HAND SIDE AND P	DGRS2460

C		AS COEFFICIENT MATRIX, USING THE	DGRS2470
C		LU DECOMPOSITION IF MITER = 1 OR	DGRS2480
C		2. IF MITER = 3, THE COEFFICIENT	DGRS2490
C		H*EL(1) IN P IS UPDATED.	DGRS2500
	105	PHL0 = HLO	DGRS2510
		HLO = H*EL(1)	DGRS2520
		IF (HLO.EQ.PHL0) GO TO 115	DGRS2530
		R = HLO/PHL0	DGRS2540
		DO 110 I=1,N	DGRS2550
		D = 1.D0-R*(1.D0-1.D0/PW(I))	DGRS2560
		IF (DABS(D).EQ.0.D0) GO TO 165	DGRS2570
	110	PW(I) = 1.D0/D	DGRS2580
	115	DO 120 I=1,N	DGRS2590
	120	SAVE1(I) = PW(I)*(H*SAVE2(I) - (Y(I,2)+ERROR(I)))	DGRS2600
		GO TO 135	DGRS2610
	125	DO 130 I=1,N	DGRS2620
	130	SAVE1(I) = H*SAVE2(I) - (Y(I,2)+ERROR(I))	DGRS2630
		IF (NLC.EQ. -1) GO TO 131	DGRS2640
		NWK = (NLC+NUC+1)*NO+1	DGRS2650
		CALL LEQT1B(PW,N,NLC,NUC,NO,SAVE1,1,NO,2,PW(NWK),JER)	DGRS2660
		GO TO 135	DGRS2670
	131	CALL LUELMF (PW,SAVE1,IPIV,N,NO,SAVE1)	DGRS2680
	135	D = 0.D0	DGRS2690
		DO 140 I=1,N	DGRS2700
		ERROR(I) = ERROR(I)+SAVE1(I)	DGRS2710
		D = D+(SAVE1(I)/YMAX(I))**2	DGRS2720
	140	SAVE1(I) = Y(I,1)+EL(1)*ERROR(I)	DGRS2730
C		TEST FOR CONVERGENCE. IF M.GT.0, THE	DGRS2740
C		SQUARE OF THE CONVERGENCE RATE	DGRS2750
C		CONSTANT IS ESTIMATED AS CRATE,	DGRS2760
C		AND THIS IS USED IN THE TEST.	DGRS2770
	145	IF (M.NE.0) CRATE = DMAX1(.9D0*CRATE,D/D1)	DGRS2780
		IF ((D*DMIN1(1.D0,2.D0*CRATE)).LE.BND) GO TO 170	DGRS2790
		D1 = D	DGRS2800
		M = M+1	DGRS2810
		IF (M.EQ.3) GO TO 150	DGRS2820
		CALL FCN (N,T,SAVE1,SAVE2,eprime,eprime2)	+
		GO TO 95	DGRS2840
C		THE CORRECTOR ITERATION FAILED TO	DGRS2850
C		CONVERGE IN 3 TRIES. IF PARTIALS	DGRS2860
C		ARE INVOLVED BUT ARE NOT UP TO	DGRS2870
C		DATE, THEY ARE REEVALUATED FOR THE	DGRS2880
C		NEXT TRY. OTHERWISE THE Y ARRAY IS	DGRS2890
C		RETRACTED TO ITS VALUES BEFORE	DGRS2900
C		PREDICTION, AND H IS REDUCED, IF	DGRS2910
C		POSSIBLE. IF NOT, A NO-CONVERGENCE	DGRS2920
C		EXIT IS TAKEN.	DGRS2930
	150	NFE = NFE+2	DGRS2940
		IF (IWEVAL.EQ.-1) GO TO 165	DGRS2950
	155	T = TOLD	DGRS2960
		RMAX = 2.D0	DGRS2970
		DO 160 J1=1,NQ	DGRS2980
		DO 160 J2=J1,NQ	DGRS2990
		J = (NQ+J1)-J2	DGRS3000
		DO 160 I=1,N	DGRS3010
	160	Y(I,J) = Y(I,J) - Y(I,J+1)	DGRS3020
		IF (DABS(H).LE.HMIN*1.00001D0) GO TO 280	DGRS3030
		RH = .25D0	DGRS3040
		IREDO = 1	DGRS3050
		GO TO 35	DGRS3060
	165	IWEVAL = MITER	DGRS3070
		GO TO 60	DGRS3080

C		THE CORRECTOR HAS CONVERGED. IWEVAL	DGRS3090
C		IS SET TO -1 IF PARTIAL	DGRS3100
C		DERIVATIVES WERE USED, TO SIGNAL	DGRS3110
C		THAT THEY MAY NEED UPDATING ON	DGRS3120
C		SUBSEQUENT STEPS. THE ERROR TEST	DGRS3130
C		IS MADE AND CONTROL PASSES TO	DGRS3140
C		STATEMENT 190 IF IT FAILS.	DGRS3150
	170 IF (MITER.NE.0) IWEVAL = -1		DGRS3160
	NFE = NFE+M		DGRS3170
	D = 0.D0		DGRS3180
	DO 175 I=1,N		DGRS3190
	175 D = D+(ERROR(I)/YMAX(I))**2		DGRS3200
	IF (D.GT.E) GO TO 190		DGRS3210
C		AFTER A SUCCESSFUL STEP, UPDATE THE	DGRS3220
C		Y ARRAY. CONSIDER CHANGING H IF	DGRS3230
C		IDOUB = 1. OTHERWISE DECREASE	DGRS3240
C		IDOUB BY 1. IF IDOUB IS THEN 1 AND	DGRS3250
C		NQ .LT. MAXDER, THEN ERROR IS	DGRS3260
C		SAVED FOR USE IN A POSSIBLE ORDER	DGRS3270
C		INCREASE ON THE NEXT STEP. IF A	DGRS3280
C		CHANGE IN H IS CONSIDERED, AN	DGRS3290
C		INCREASE OR DECREASE IN ORDER BY	DGRS3300
C		ONE IS CONSIDERED ALSO. A CHANGE	DGRS3310
C		IN H IS MADE ONLY IF IT IS BY A	DGRS3320
C		FACTOR OF AT LEAST 1.1. IF NOT,	DGRS3330
C		IDOUB IS SET TO 10 TO PREVENT	DGRS3340
C		TESTING FOR THAT MANY STEPS.	DGRS3350
	KFLAG = 0		DGRS3360
	IREDO = 0		DGRS3370
	NSTEP = NSTEP+1		DGRS3380
	HUSED = H		DGRS3390
	NQUSED = NQ		DGRS3400
	DO 180 J=1,L		DGRS3410
	DO 180 I=1,N		DGRS3420
	180 Y(I,J) = Y(I,J)+EL(J)*ERROR(I)		DGRS3430
	IF (IDOUB.EQ.1) GO TO 200		DGRS3440
	IDOUB = IDOUB-1		DGRS3450
	IF (IDOUB.GT.1) GO TO 290		DGRS3460
	IF (L.EQ.LMAX) GO TO 290		DGRS3470
	DO 185 I=1,N		DGRS3480
	185 Y(I,LMAX) = ERROR(I)		DGRS3490
	GO TO 290		DGRS3500
C		THE ERROR TEST FAILED. KFLAG KEEPS	DGRS3510
C		TRACK OF MULTIPLE FAILURES.	DGRS3520
C		RESTORE T AND THE Y ARRAY TO THEIR	DGRS3530
C		PREVIOUS VALUES, AND PREPARE TO	DGRS3540
C		TRY THE STEP AGAIN. COMPUTE THE	DGRS3550
C		OPTIMUM STEP SIZE FOR THIS OR ONE	DGRS3560
C		LOWER ORDER.	DGRS3570
	190 KFLAG = KFLAG-1		DGRS3580
	T = TOLD		DGRS3590
	DO 195 J1=1,NQ		DGRS3600
	DO 195 J2=J1,NQ		DGRS3610
	J = (NQ+J1)-J2		DGRS3620
	DO 195 I=1,N		DGRS3630
	195 Y(I,J) = Y(I,J)-Y(I,J+1)		DGRS3640
	RMAX = 2.D0		DGRS3650
	IF (DABS(H).LE.HMIN*1.00001D0) GO TO 270		DGRS3660
	IF (KFLAG.LE.-3) GO TO 260		DGRS3670
	IREDO = 2		DGRS3680
	PR3 = 1.D+20		DGRS3690
	GO TO 210		DGRS3700

C
C
C
C
C
C
C
C
C
C
C
C

REGARDLESS OF THE SUCCESS OR FAILURE OF THE STEP, FACTORS PR1, PR2, AND PR3 ARE COMPUTED, BY WHICH H COULD BE DIVIDED AT ORDER NQ - 1, ORDER NQ, OR ORDER NQ + 1, RESPECTIVELY. IN THE CASE OF FAILURE, PR3 = 1.E20 TO AVOID AN ORDER INCREASE. THE SMALLEST OF THESE IS DETERMINED AND THE NEW ORDER CHOSEN ACCORDINGLY. IF THE ORDER IS TO BE INCREASED, WE COMPUTE ONE ADDITIONAL SCALED DERIVATIVE.

200 PR3 = 1.D+20
IF (L.EQ.LMAX) GO TO 210
D1 = 0.D0
DO 205 I=1,N
205 D1 = D1+((ERROR(I)-Y(I,LMAX))/YMAX(I))**2
ENQ3 = .5D0/(L+1)
PR3 = ((D1/EUP)**ENQ3)*1.4D0+1.4D-6
210 ENQ2 = .5D0/L
PR2 = ((D/E)**ENQ2)*1.2D0+1.2D-6
PR1 = 1.D+20
IF (NQ.EQ.1) GO TO 220
D = 0.D0
DO 215 I=1,N
215 D = D+(Y(I,L)/YMAX(I))**2
ENQ1 = .5D0/NQ
PR1 = ((D/EDN)**ENQ1)*1.3D0+1.3D-6
220 IF (PR2.LE.PR3) GO TO 225
IF (PR3.LT.PR1) GO TO 235
GO TO 230
225 IF (PR2.GT.PR1) GO TO 230
NEWQ = NQ
RH = 1.D0/PR2
GO TO 250
230 NEWQ = NQ-1
RH = 1.D0/PR1
IF (KFLAG.NE.0.AND.RH.GT.1.D0) RH = 1.D0
GO TO 250
235 NEWQ = L
RH = 1.D0/PR3
IF (RH.LT.1.1D0) GO TO 245
DO 240 I=1,N
240 Y(I,NEWQ+1) = ERROR(I)*EL(L)/L
GO TO 255
245 IDOUB = 10
GO TO 290
250 IF ((KFLAG.EQ.0).AND.(RH.LT.1.1D0)) GO TO 245

C
C
C
C
C
C
C

IF THERE IS A CHANGE OF ORDER, RESET NQ, L, AND THE COEFFICIENTS. IN ANY CASE H IS RESET ACCORDING TO RH AND THE Y ARRAY IS RESCALED. THEN EXIT FROM 285 IF THE STEP WAS OK, OR REDO THE STEP OTHERWISE.

IF (NEWQ.EQ.NQ) GO TO 35
255 NQ = NEWQ
L = NQ+1
IRET = 2
GO TO 15

C
C

CONTROL REACHES THIS SECTION IF 3 OR MORE FAILURES HAVE OCCURED. IT IS

C
C
C
C
C
C
C
C
C
C

ASSUMED THAT THE DERIVATIVES THAT
HAVE ACCUMULATED IN THE Y ARRAY
HAVE ERRORS OF THE WRONG ORDER.
HENCE THE FIRST DERIVATIVE IS
RECOMPUTED, AND THE ORDER IS SET
TO 1. THEN H IS REDUCED BY A
FACTOR OF 10, AND THE STEP IS
RETRIED. AFTER A TOTAL OF 7
FAILURES, AN EXIT IS TAKEN WITH
KFLAG = -2.

DGRS4330
DGRS4340
DGRS4350
DGRS4360
DGRS4370
DGRS4380
DGRS4390
DGRS4400
DGRS4410
DGRS4420
DGRS4430
DGRS4440
DGRS4450
DGRS4460
+
DGRS4480
DGRS4490
DGRS4500
DGRS4510
DGRS4520
DGRS4530
DGRS4540
DGRS4550
DGRS4560
DGRS4570
DGRS4580
DGRS4590
DGRS4600
DGRS4610
DGRS4620
DGRS4630
DGRS4640
DGRS4650
DGRS4660
DGRS4670
DGRS4680
DGRS4690
DGRS4700
+
+
+
+
+
+
DGRS4710
DGRS4720

260 IF (KFLAG.EQ.-7) GO TO 275
RH = .1D0
RH = DMAX1(HMIN/DABS(H), RH)
H = H*RH
CALL FCN (N,T,Y,SAVE1,eprime,eprime2)
NFE = NFE+1
DO 265 I=1,N
265 Y(I,2) = H*SAVE1(I)
INEVAL = MITER
IDCUB = 10
IF (NQ.EQ.1) GO TO 50
NQ = 1
L = 2
IRFT = 3
GO TO 15

ALL RETURNS ARE MADE THROUGH THIS
SECTION. H IS SAVED IN HOLD TO
ALLOW THE CALLER TO CHANGE H ON
THE NEXT STEP.

C
C
C
C

270 KFLAG = -1
GO TO 290
275 KFLAG = -2
GO TO 290
280 KFLAG = -3
GO TO 290
285 RMAX = 10.D0
290 HOLD = H
JSTART = NQ
C--Diagnostic Check of first and second derivatives of E
if(tcum.eq.told)go to 310
write(8,300)tcum,step,y(1,1),y(2,1),y(3,1),y(4,1),y(5,1)
300 format(1x,e11.4,1x,i5,5(1x,e11.4))
write(9,305)step,eprime,eprime2
305 format(1x,i5,2(1x,e20.13))
RETURN
END


```

C   IMSL ROUTINE NAME      - DGRCS                                DGRC0010
C                                                                    DGRC0020
C-----DGRC0030
C   COMPUTER                - IBM/DOUBLE                          DGRC0040
C                                                                    DGRC0050
C   LATEST REVISION         - JANUARY 1, 1978                      DGRC0060
C                                                                    DGRC0070
C   PURPOSE                 - NUCLEUS CALLED ONLY BY IMSL SUBROUTINE DGEAR DGRC0080
C                                                                    DGRC0090
C   PRECISION/HARDWARE      - SINGLE AND DOUBLE/H32                DGRC0100
C                                                                    DGRC0110
C                           - SINGLE/H36,H48,H60                    DGRC0120
C                                                                    DGRC0130
C   REQD. IMSL ROUTINES    - NONE REQUIRED                          DGRC0140
C                                                                    DGRC0150
C   NOTATION                - INFORMATION ON SPECIAL NOTATION AND  DGRC0160
C                           CONVENTIONS IS AVAILABLE IN THE MANUAL DGRC0170
C                           INTRODUCTION OR THROUGH IMSL ROUTINE UHELP DGRC0180
C                                                                    DGRC0190
C   COPYRIGHT              - 1978 BY IMSL, INC. ALL RIGHTS RESERVED. DGRC0200
C                                                                    DGRC0210
C   WARRANTY               - IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN DGRC0220
C                           APPLIED TO THIS CODE. NO OTHER WARRANTY, DGRC0230
C                           EXPRESSED OR IMPLIED, IS APPLICABLE.     DGRC0240
C                                                                    DGRC0250
C-----DGRC0260
C   SUBROUTINE DGRCS      (METH,NQ,EL,TQ,MAXDER)                  DGRC0270
C                                                                    DGRC0280
C                           SPECIFICATIONS FOR ARGUMENTS            DGRC0290
C   INTEGER               METH,NQ,MAXDER                          DGRC0300
C   REAL                  TQ(1)                                    DGRC0310
C   DOUBLE PRECISION      EL(1)                                    DGRC0320
C                                                                    DGRC0330
C                           SPECIFICATIONS FOR LOCAL VARIABLES      DGRC0340
C   INTEGER               K                                        DGRC0350
C   REAL                  PERTST(12,2,3)                          DGRC0360
C   DATA                 PERTST/1.,1.,2.,1.,.3158,.7407E-1,      DGRC0370
C   1                      .1391E-1,.2182E-2,.2945E-3,.3492E-4,    DGRC0380
C   2                      .3692E-5,.3524E-6,1.,1.,.5,.1667,       DGRC0390
C   3                      .4167E-1,7*1.,2.,12.,24.,37.89,         DGRC0400
C   4                      53.33,70.08,87.97,106.9,126.7,          DGRC0410
C   5                      147.4,168.8,191.0,2.0,4.5,7.333,         DGRC0420
C   6                      10.42,13.7,7*1.,12.0,24.0,37.89,        DGRC0430
C   7                      53.33,70.08,87.97,106.9,126.7,          DGRC0440
C   8                      147.4,168.8,191.0,1.,3.0,6.0,           DGRC0450
C   9                      9.167,12.5,8*1./                          DGRC0460
C                                                                    DGRC0470
C                           FIRST EXECUTABLE STATEMENT              DGRC0480
C   GO TO (5,10), METH                                           DGRC0490
C   5 MAXDER = 12                                                  DGRC0500
C   GO TO (15,20,25,30,35,40,45,50,55,60,65,70), NQ             DGRC0510
C   10 MAXDER = 5                                                  DGRC0520
C   GO TO (75,80,85,90,95), NQ                                    DGRC0530
C                                                                    DGRC0540
C   THE FOLLOWING COEFFICIENTS SHOULD BE DEFINED TO MACHINE ACCURACY. FOR A DGRC0550
C   GIVEN ORDER NQ, THEY CAN BE CALCULATED BY USE OF THE          DGRC0560
C   GENERATING POLYNOMIAL L(T), WHOSE COEFFICIENTS ARE EL(I) .. L(T) = DGRC0570
C   EL(1) + EL(2)*T + ... + EL(NQ+1)*T**NQ. FOR THE IMPLICIT      DGRC0580
C   ADAMS METHODS, L(T) IS GIVEN BY DL/DT = (T+1)*(T+2)* ...      DGRC0590
C   *(T+NQ-1)/K, L(-1) = 0, WHERE K = DGRC0600
C                                                                    DGRC0610
C                                                                    DGRC0620

```

C
C
C
C
C
C
C
C
C

FACTORIAL(MQ-1). FOR THE GEAR
METHODS, $L(T) = (T+1)*(T+2)* \dots$
 $*(T+MQ)/K$, WHERE $K =$
 $FACTORIAL(MQ)*(1 + 1/2 + \dots +$
 $1/MQ)$. THE ORDER IN WHICH THE
GROUPS APPEAR BELOW IS.. IMPLICIT
ADAMS METHODS OF ORDERS 1 TO 12,
BACKWARD DIFFERENTIATION METHODS
OF ORDERS 1 TO 5.

15 EL(1) = 1.0D0
GO TO 100
20 EL(1) = 0.5D0
EL(3) = 0.5D0
GO TO 100
25 EL(1) = 4.166666666666667D-01
EL(3) = 0.75D0
EL(4) = 1.666666666666667D-01
GO TO 100
30 EL(1) = 0.375D0
EL(3) = 9.166666666666667D-01
EL(4) = 3.333333333333333D-01
EL(5) = 4.166666666666667D-02
GO TO 100
35 EL(1) = 3.486111111111111D-01
EL(3) = 1.041666666666667D0
EL(4) = 4.861111111111111D-01
EL(5) = 1.041666666666667D-01
EL(6) = 8.333333333333333D-03
GO TO 100
40 EL(1) = 3.298611111111111D-01
EL(3) = 1.141666666666667D+00
EL(4) = 0.625D+00
EL(5) = 1.770833333333333D-01
EL(6) = 0.025D+00
EL(7) = 1.388888888888889D-03
GO TO 100
45 EL(1) = 3.155919312169312D-01
EL(3) = 1.225D+00
EL(4) = 7.518518518518519D-01
EL(5) = 2.552083333333333D-01
EL(6) = 4.861111111111111D-02
EL(7) = 4.861111111111111D-03
EL(8) = 1.984126984126984D-04
GO TO 100
50 EL(1) = 3.042245370370370D-01
EL(3) = 1.296428571428571D+00
EL(4) = 8.685185185185185D-01
EL(5) = 3.357638888888889D-01
EL(6) = 7.777777777777778D-02
EL(7) = 1.064814814814815D-02
EL(8) = 7.936507936507937D-04
EL(9) = 2.480158730158730D-05
GO TO 100
55 EL(1) = 2.948680004409171D-01
EL(3) = 1.358928571428571D+00
EL(4) = 9.765542328042328D-01
EL(5) = 4.171875D-01
EL(6) = 1.113541666666667D-01
EL(7) = 0.01875D+00
EL(8) = 1.934523809523810D-03
EL(9) = 1.116071428571429D-04
EL(10) = 2.755731922398589D-06

DGRC0630
DGRC0640
DGRC0650
DGRC0660
DGRC0670
DGRC0680
DGRC0690
DGRC0700
DGRC0710
DGRC0720
DGRC0730
DGRC0740
DGRC0750
DGRC0760
DGRC0770
DGRC0780
DGRC0790
DGRC0800
DGRC0810
DGRC0820
DGRC0830
DGRC0840
DGRC0850
DGRC0860
DGRC0870
DGRC0880
DGRC0890
DGRC0900
DGRC0910
DGRC0920
DGRC0930
DGRC0940
DGRC0950
DGRC0960
DGRC0970
DGRC0980
DGRC0990
DGRC1000
DGRC1010
DGRC1020
DGRC1030
DGRC1040
DGRC1050
DGRC1060
DGRC1070
DGRC1080
DGRC1090
DGRC1100
DGRC1110
DGRC1120
DGRC1130
DGRC1140
DGRC1150
DGRC1160
DGRC1170
DGRC1180
DGRC1190
DGRC1200
DGRC1210
DGRC1220
DGRC1230
DGRC1240

GO TO 100	DGRC1250
60 EL(1) = 2.869754464285714D-01	DGRC1260
EL(3) = 1.414484126984127D+00	DGRC1270
EL(4) = 1.077215608465609D+00	DGRC1280
EL(5) = 4.985670194003527D-01	DGRC1290
EL(6) = 1.484375D-01	DGRC1300
EL(7) = 2.906057098765432D-02	DGRC1310
EL(8) = 3.720238095238095D-03	DGRC1320
EL(9) = 2.996858465608466D-04	DGRC1330
EL(10) = 1.377865961199295D-05	DGRC1340
EL(11) = 2.755731922398589D-07	DGRC1350
GO TO 100	DGRC1360
65 EL(1) = 2.801895964439367D-01	DGRC1370
EL(3) = 1.464484126984127D+00	DGRC1380
EL(4) = 1.171514550264550D+00	DGRC1390
EL(5) = 5.793581900352734D-01	DGRC1400
EL(6) = 1.883228615520282D-01	DGRC1410
EL(7) = 4.143036265432099D-02	DGRC1420
EL(8) = 6.211144179894180D-03	DGRC1430
EL(9) = 6.252066798941799D-04	DGRC1440
EL(10) = 4.041740152851264D-05	DGRC1450
EL(11) = 1.515652557319224D-06	DGRC1460
EL(12) = 2.505210838544172D-08	DGRC1470
GO TO 100	DGRC1480
70 EL(1) = 2.742655400315991D-01	DGRC1490
EL(3) = 1.509938672438672D+00	DGRC1500
EL(4) = 1.260271164021164D+00	DGRC1510
EL(5) = 6.592341820987654D-01	DGRC1520
EL(6) = 2.304580026455027D-01	DGRC1530
EL(7) = 5.569724610523222D-02	DGRC1540
EL(8) = 9.439484126984127D-03	DGRC1550
EL(9) = 1.119274966931217D-03	DGRC1560
EL(10) = 9.093915343915344D-05	DGRC1570
EL(11) = 4.822530864197531D-06	DGRC1580
EL(12) = 1.503126503126503D-07	DGRC1590
EL(13) = 2.087675698786810D-09	DGRC1600
GO TO 100	DGRC1610
C	DGRC1620
75 EL(1) = 1.0D+00	DGRC1630
GO TO 100	DGRC1640
80 EL(1) = 6.666666666666667D-01	DGRC1650
EL(3) = 3.333333333333333D-01	DGRC1660
GO TO 100	DGRC1670
85 EL(1) = 5.454545454545455D-01	DGRC1680
EL(3) = EL(1)	DGRC1690
EL(4) = 9.090909090909091D-02	DGRC1700
GO TO 100	DGRC1710
90 EL(1) = 0.48D+00	DGRC1720
EL(3) = 0.7D+00	DGRC1730
EL(4) = 0.2D+00	DGRC1740
EL(5) = 0.02D+00	DGRC1750
GO TO 100	DGRC1760
95 EL(1) = 4.379562043795620D-01	DGRC1770
EL(3) = 8.211678832116788D-01	DGRC1780
EL(4) = 3.102189781021898D-01	DGRC1790
EL(5) = 5.474452554744526D-02	DGRC1800
EL(6) = 3.649635036496350D-03	DGRC1810
C	DGRC1820
100 DO 105 K=1,3	DGRC1830
TQ(K) = PERTST(NQ,METH,K)	DGRC1840
105 CONTINUE	DGRC1850
TQ(4) = .5D0*TQ(2)/(NQ+2)	DGRC1860

RETURN
END

DGRC1870
DGRC1880

C	IMSL ROUTINE NAME	- DGRPS	DGRP0010
C			DGRP0020
C			DGRP0030
C	COMPUTER	- IBM/DOUBLE	DGRP0040
C			DGRP0050
C	LATEST REVISION	- NOVEMBER 1, 1984	DGRP0060
C			DGRP0070
C	PURPOSE	- NUCLEUS CALLED ONLY BY IMSL SUBROUTINE DGEAR	DGRP0080
C			DGRP0090
C	PRECISION/HARDWARE	- SINGLE AND DOUBLE/H32	DGRP0100
C		- SINGLE/H36,H48,H60	DGRP0110
C			DGRP0120
C	REQD. IMSL ROUTINES	- LUDATF,LEQT1B,UERTST,UGETIO	DGRP0130
C			DGRP0140
C	NOTATION	- INFORMATION ON SPECIAL NOTATION AND	DGRP0150
C		CONVENTIONS IS AVAILABLE IN THE MANUAL	DGRP0160
C		INTRODUCTION OR THROUGH IMSL ROUTINE UHELP	DGRP0170
C			DGRP0180
C	COPYRIGHT	- 1984 BY IMSL, INC. ALL RIGHTS RESERVED.	DGRP0190
C			DGRP0200
C	WARRANTY	- IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN	DGRP0210
C		APPLIED TO THIS CODE. NO OTHER WARRANTY,	DGRP0220
C		EXPRESSED OR IMPLIED, IS APPLICABLE.	DGRP0230
C			DGRP0240
C			DGRP0250
C			DGRP0260
C	SUBROUTINE DGRPS	(FCN,FCNJ,Y,N0,CON,MITER,YMAX,SAVE1,SAVE2,PW,	DGRP0270
C	*	EQUIL,IPIV,IER)	DGRP0280
C		SPECIFICATIONS FOR ARGUMENTS	DGRP0290
C	INTEGER	N0,MITER,IPIV(1),IER	DGRP0300
C	DOUBLE PRECISION	Y(N0,1),CON,YMAX(1),SAVE1(1),SAVE2(1),PW(1),	DGRP0310
C	*	EQUIL(1)	DGRP0320
C		SPECIFICATIONS FOR LOCAL VARIABLES	DGRP0330
C			DGRP0340
C	INTEGER	NC,MFC,KFLAG,JSTART,NQUSED,NSTEP,NFE,NJE,NPW,	DGRP0350
C	*	NSQ,I,J1,J,NERROR,NSAVE1,NSAVE2,NEQUIL,NY,	DGRP0360
C	*	IDUMMY(23),NLIM,II,IJ,LIM1,LIM2,NB,NLC,NUC,NWK	DGRP0370
C	REAL	SDUMMY(4)	DGRP0380
C	DOUBLE PRECISION	T,H,HMIN,HMAX,EPSC,URound,EPSJ,HUSED,D,R0,YJ,R,	DGRP0390
C	*	D1,D2,WA,DUMMY(40)	DGRP0400
C	COMMON /DBAND/	NLC,NUC	DGRP0410
C	COMMON /GEAR/	T,H,HMIN,HMAX,EPSC,URound,EPSJ,HUSED,DUMMY,	DGRP0420
C	*	SDUMMY,NC,MFC,KFLAG,JSTART,NSQ,NQUSED,NSTEP,	DGRP0430
C	*	NFE,NJE,NPW,NERROR,NSAVE1,NSAVE2,NEQUIL,NY,	DGRP0440
C	*	IDUMMY	DGRP0450
C		THIS ROUTINE IS CALLED BY DGRST TO	DGRP0460
C		COMPUTE AND PROCESS THE MATRIX P =	DGRP0470
C		I - H*EL(1)*J , WHERE J IS AN	DGRP0480
C		APPROXIMATION TO THE JACOBIAN. J	DGRP0490
C		IS COMPUTED, EITHER BY THE USER-	DGRP0500
C		SUPPLIED ROUTINE FCNJ IF MITER =	DGRP0510
C		1, OR BY FINITE DIFFERENCING IF	DGRP0520
C		MITER = 2. J IS STORED IN PW AND	DGRP0530
C		REPLACED BY P, USING CON =	DGRP0540
C		-H*EL(1). THEN P IS SUBJECTED TO	DGRP0550
C		LU DECOMPOSITION IN PREPARATION	DGRP0560
C		FOR LATER SOLUTION OF LINEAR	DGRP0570
C		SYSTEMS WITH P AS COEFFICIENT	DGRP0580
C		MATRIX. IN ADDITION TO VARIABLES	DGRP0590
C		DESCRIBED PREVIOUSLY,	DGRP0600
C		COMMUNICATION WITH DGRPS USES THE	DGRP0610
C			DGRP0620

C		FOLLOWING EPSJ = DSQRT(UROUND),	DGRP0630
C		USED IN THE NUMERICAL JACOBIAN	DGRP0640
C		INCREMENTS.	DGRP0650
C		FIRST EXECUTABLE STATEMENT	DGRP0660
C	IF (NLC.EQ.-1) GO TO 45		DGRP0670
C		BANDED JACOBIAN CASE	DGRP0680
	NB = NLC+NUC+1		DGRP0690
	NWK = NB*N0+1		DGRP0700
	IF (MITER.EQ.2) GO TO 15		DGRP0710
C		MITER = 1	DGRP0720
	NLIM = NB*N0		DGRP0730
	DO 5 I=1,NLIM		DGRP0740
	PW(I) = 0.0D0		DGRP0750
	5 CONTINUE		DGRP0760
	CALL FCNJ(NC,T,Y,PW)		DGRP0770
	DO 10 I=1,NLIM		DGRP0780
	PW(I) = PW(I)*CON		DGRP0790
	10 CONTINUE		DGRP0800
	GO TO 35		DGRP0810
C		MITER = 2	DGRP0820
	15 D = 0.0D0		DGRP0830
	DO 20 I=1,NC		DGRP0840
	20 D = D+SAVE2(I)**2		DGRP0850
	R0 = DABS(H)*DSQRT(D)*1.0D+03*UROUND		DGRP0860
	DO 30 J=1,NC		DGRP0870
	YJ = Y(J,1)		DGRP0880
	R = EPSJ*YMAX(J)		DGRP0890
	R = DMAX1(R,R0)		DGRP0900
	Y(J,1) = Y(J,1)+R		DGRP0910
	D = CON/R		DGRP0920
	CALL FCN(NC,T,Y,SAVE1)		DGRP0930
	LIM1 = MAX0(1,J-NUC)		DGRP0940
	LIM2 = MIN0(N0,J+NLC)		DGRP0950
	DO 25 I=LIM1,LIM2		DGRP0960
	IJ = (J-I+NLC)*N0+I		DGRP0970
	PW(IJ) = (SAVE1(I)-SAVE2(I))*D		DGRP0980
	25 CONTINUE		DGRP0990
	Y(J,1) = YJ		DGRP1000
	30 CONTINUE		DGRP1010
C		ADD IDENTITY MATRIX.	DGRP1020
	35 DO 40 I=1,NC		DGRP1030
	II = NLC*N0+I		DGRP1040
	PW(II) = PW(II)+1.0D0		DGRP1050
	40 CONTINUE		DGRP1060
C		DO LU DECOMPOSITION ON P	DGRP1070
C			DGRP1080
	CALL LEQT1B(PW,NC,NLC,NUC,N0,EQUIL,1,N0,1,PW(NWK),IER)		DGRP1090
	RETURN		DGRP1100
C		FULL JACOBIAN CASE	DGRP1110
C	45 IF (MITER.EQ.2) GO TO 55		DGRP1120
		MITER = 1	DGRP1130
	CALL FCNJ(NC,T,Y,PW)		DGRP1140
	DO 50 I=1,NSQ		DGRP1150
	50 PW(I) = PW(I)*CON		DGRP1160
	GO TO 75		DGRP1170
C		MITER = 2	DGRP1180
	55 D = 0.0D0		DGRP1190
	DO 60 I=1,NC		DGRP1200
	60 D = D+SAVE2(I)**2		DGRP1210
	R0 = DABS(H)*DSQRT(D)*1.0D+03*UROUND		DGRP1220
	J1 = 0		DGRP1230
			DGRP1240

DO 70 J=1,NC		DGRP1250
YJ = Y(J,1)		DGRP1260
R = EPSJ*YMAX(J)		DGRP1270
R = DMAX1(R,R0)		DGRP1280
Y(J,1) = Y(J,1)+R		DGRP1290
D = CON/R		DGRP1300
CALL FCN(NC,T,Y,SAVE1)		DGRP1310
DO 65 I=1,NC		DGRP1320
65 PW(I+J1) = (SAVE1(I) - SAVE2(I)) * D		DGRP1330
Y(J,1) = YJ		DGRP1340
J1 = J1+N0		DGRP1350
70 CONTINUE		DGRP1360
C ADD IDENTITY MATRIX.		DGRP1370
75 J = 1		DGRP1380
DO 80 I=1,NC		DGRP1390
PW(J) = PW(J)+1.0D0		DGRP1400
J = J+(N0+1)		DGRP1410
80 CONTINUE		DGRP1420
C DO LU DECOMPOSITION ON P.		DGRP1430
C		DGRP1440
CALL LUDATF(PW,PW,NC,N0,0,D1,D2,IPIV,EQUIL,WA,IER)		DGRP1450
RETURN		DGRP1460
END		DGRP1470

C	IMSL ROUTINE NAME	- DGRIN	DGRI0010
C			DGRI0020
C	-----		DGRI0030
C	COMPUTER	- IBM/DOUBLE	DGRI0040
C			DGRI0050
C	LATEST REVISION	- JANUARY 1, 1978	DGRI0060
C			DGRI0070
C	PURPOSE	- NUCLEUS CALLED ONLY BY IMSL SUBROUTINE DGEAR	DGRI0080
C			DGRI0090
C	PRECISION/HARDWARE	- SINGLE AND DOUBLE/H32	DGRI0100
C		- SINGLE/H36,H48,H60	DGRI0110
C			DGRI0120
C	REQD. IMSL ROUTINES	- NONE REQUIRED	DGRI0130
C			DGRI0140
C	NOTATION	- INFORMATION ON SPECIAL NOTATION AND	DGRI0150
C		CONVENTIONS IS AVAILABLE IN THE MANUAL	DGRI0160
C		INTRODUCTION OR THROUGH IMSL ROUTINE UHELP	DGRI0170
C			DGRI0180
C	COPYRIGHT	- 1978 BY IMSL, INC. ALL RIGHTS RESERVED.	DGRI0190
C			DGRI0200
C	WARRANTY	- IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN	DGRI0210
C		APPLIED TO THIS CODE. NO OTHER WARRANTY,	DGRI0220
C		EXPRESSED OR IMPLIED, IS APPLICABLE.	DGRI0230
C			DGRI0240
C	-----		DGRI0250
C	SUBROUTINE DGRIN	(TOUT,Y,N0,Y0)	DGRI0260
C			DGRI0270
C		SPECIFICATIONS FOR ARGUMENTS	DGRI0280
C	INTEGER	N0	DGRI0290
C	DOUBLE PRECISION	TOUT,Y0(N0),Y(N0,1)	DGRI0300
C			DGRI0310
C		SPECIFICATIONS FOR LOCAL VARIABLES	DGRI0320
C	INTEGER	NC,MFC,KFLAG,I,L,J,JSTART,NSQ,NQUSED,NSTEP,	DGRI0330
C	1	NFE,NJE,NPW,NERROR,NSAVE1,NSAVE2,NEQUIL,NY,	DGRI0340
C	2	IDUMMY(23)	DGRI0350
C	REAL	SDUMMY(4)	DGRI0360
C	DOUBLE PRECISION	T,H,HMIN,HMAX,EPSC,URound,EPSJ,HUSED,S,S1,	DGRI0370
C	1	DUMMY(40)	DGRI0380
C	COMMON /GEAR/	T,H,HMIN,HMAX,EPSC,URound,EPSJ,HUSED,DUMMY,	DGRI0390
C	1	SDUMMY,NC,MFC,KFLAG,JSTART,NSQ,NQUSED,NSTEP,	DGRI0400
C	2	NFE,NJE,NPW,NERROR,NSAVE1,NSAVE2,NEQUIL,NY,	DGRI0410
C	3	IDUMMY	DGRI0420
C		FIRST EXECUTABLE STATEMENT	DGRI0430
C	DO 5 I = 1,NC		DGRI0440
C	Y0(I) = Y(I,1)		DGRI0450
C	5 CONTINUE		DGRI0460
C		THIS SUBROUTINE COMPUTES INTERPOLATED	DGRI0470
C		VALUES OF THE DEPENDENT VARIABLE	DGRI0480
C		Y AND STORES THEM IN Y0. THE	DGRI0490
C		INTERPOLATION IS TO THE	DGRI0500
C		POINT T = TOUT, AND USES THE	DGRI0510
C		NORDSIECK HISTORY ARRAY Y, AS	DGRI0520
C		FOLLOWS..	DGRI0530
C		NQ	DGRI0540
C		Y0(I) = SUM Y(I,J+1)*S**J ,	DGRI0550
C		J=0	DGRI0560
C		WHERE S = - (T-TOUT)/H.	DGRI0570
C			DGRI0580
C	L = JSTART + 1		DGRI0590
C	S = (TOUT - T)/H		DGRI0600
C	S1 = 1.0D0		DGRI0610
C	DO 15 J = 2,L		DGRI0620
C	S1 = S1*S		


```
      DO 10 I = 1,NC  
        Y0(I) = Y0(I) + S1*Y(I,J)  
10    CONTINUE  
15    CONTINUE  
      RETURN  
      END
```

```
DGRI0630  
DGRI0640  
DGRI0650  
DGRI0660  
DGRI0670  
DGRI0680
```

C	IMSL ROUTINE NAME	- LUDATF	LUDA0010
C			LUDA0020
C			LUDA0030
C	COMPUTER	- IBM/DOUBLE	LUDA0040
C			LUDA0050
C	LATEST REVISION	- JANUARY 1, 1978	LUDA0060
C			LUDA0070
C	PURPOSE	- L-U DECOMPOSITION BY THE CROUT ALGORITHM WITH OPTIONAL ACCURACY TEST.	LUDA0080
C			LUDA0090
C			LUDA0100
C	USAGE	- CALL LUDATF (A, LU, N, IA, IDGT, D1, D2, IPVT, EQUIL, WA, IER)	LUDA0110
C			LUDA0120
C			LUDA0130
C			LUDA0140
C	ARGUMENTS	A	LUDA0150
C		- INPUT MATRIX OF DIMENSION N BY N CONTAINING THE MATRIX TO BE DECOMPOSED.	LUDA0160
C		LU	LUDA0170
C		- REAL OUTPUT MATRIX OF DIMENSION N BY N CONTAINING THE L-U DECOMPOSITION OF A ROWWISE PERMUTATION OF THE INPUT MATRIX. FOR A DESCRIPTION OF THE FORMAT OF LU, SEE EXAMPLE.	LUDA0180
C			LUDA0190
C			LUDA0200
C		N	LUDA0210
C		- INPUT SCALAR CONTAINING THE ORDER OF THE MATRIX A.	LUDA0220
C		IA	LUDA0230
C		- INPUT SCALAR CONTAINING THE ROW DIMENSION OF MATRICES A AND LU EXACTLY AS SPECIFIED IN THE CALLING PROGRAM.	LUDA0240
C			LUDA0250
C		IDGT	LUDA0260
C		- INPUT OPTION.	LUDA0270
C		IF IDGT IS GREATER THAN ZERO, THE NON-ZERO ELEMENTS OF A ARE ASSUMED TO BE CORRECT TO IDGT DECIMAL PLACES. LUDATF PERFORMS AN ACCURACY TEST TO DETERMINE IF THE COMPUTED DECOMPOSITION IS THE EXACT DECOMPOSITION OF A MATRIX WHICH DIFFERS FROM THE GIVEN ONE BY LESS THAN ITS UNCERTAINTY.	LUDA0280
C		IF IDGT IS EQUAL TO ZERO, THE ACCURACY TEST IS BYPASSED.	LUDA0290
C		D1	LUDA0300
C		- OUTPUT SCALAR CONTAINING ONE OF THE TWO COMPONENTS OF THE DETERMINANT. SEE DESCRIPTION OF PARAMETER D2, BELOW.	LUDA0310
C			LUDA0320
C		D2	LUDA0330
C		- OUTPUT SCALAR CONTAINING ONE OF THE TWO COMPONENTS OF THE DETERMINANT. THE DETERMINANT MAY BE EVALUATED AS (D1) (2**D2).	LUDA0340
C			LUDA0350
C		IPVT	LUDA0360
C		- OUTPUT VECTOR OF LENGTH N CONTAINING THE PERMUTATION INDICES. SEE DOCUMENT (ALGORITHM).	LUDA0370
C			LUDA0380
C		EQUIL	LUDA0390
C		- OUTPUT VECTOR OF LENGTH N CONTAINING RECIPROCAL OF THE ABSOLUTE VALUES OF THE LARGEST (IN ABSOLUTE VALUE) ELEMENT IN EACH ROW.	LUDA0400
C			LUDA0410
C		WA	LUDA0420
C		- ACCURACY TEST PARAMETER, OUTPUT ONLY IF IDGT IS GREATER THAN ZERO. SEE ELEMENT DOCUMENTATION FOR DETAILS.	LUDA0430
C			LUDA0440
C		IER	LUDA0450
C		- ERROR PARAMETER. (OUTPUT) TERMINAL ERROR	LUDA0460
C		IER = 129 INDICATES THAT MATRIX A IS ALGORITHMICALLY SINGULAR. (SEE THE CHAPTER I PRELUDE).	LUDA0470
C			LUDA0480
C		WARNING ERROR	LUDA0490
C		IER = 34 INDICATES THAT THE ACCURACY TEST FAILED. THE COMPUTED SOLUTION MAY BE IN ERROR BY MORE THAN CAN BE ACCOUNTED FOR BY THE UNCERTAINTY OF THE DATA. THIS	LUDA0500
C			LUDA0510
C			LUDA0520
C			LUDA0530
C			LUDA0540
C			LUDA0550
C			LUDA0560
C			LUDA0570
C			LUDA0580
C			LUDA0590
C			LUDA0600
C			LUDA0610
C			LUDA0620

C		WARNING CAN BE PRODUCED ONLY IF IDGT IS	LUDA0630
C		GREATER THAN 0 ON INPUT. SEE CHAPTER L	LUDA0640
C		PRELUDE FOR FURTHER DISCUSSION.	LUDA0650
C			LUDA0660
C	PRECISION/HARDWARE	- SINGLE AND DOUBLE/H32	LUDA0670
C		- SINGLE/H36,H48,H60	LUDA0680
C			LUDA0690
C	REQD. IMSL ROUTINES	- UERTST,UGETIO	LUDA0700
C			LUDA0710
C	NOTATION	- INFORMATION ON SPECIAL NOTATION AND	LUDA0720
C		CONVENTIONS IS AVAILABLE IN THE MANUAL	LUDA0730
C		INTRODUCTION OR THROUGH IMSL ROUTINE UHELP	LUDA0740
C			LUDA0750
C	REMARKS	A TEST FOR SINGULARITY IS MADE AT TWO LEVELS:	LUDA0760
C		1. A ROW OF THE ORIGINAL MATRIX A IS NULL.	LUDA0770
C		2. A COLUMN BECOMES NULL IN THE FACTORIZATION PROCESS.	LUDA0780
C			LUDA0790
C	COPYRIGHT	- 1978 BY IMSL, INC. ALL RIGHTS RESERVED.	LUDA0800
C			LUDA0810
C	WARRANTY	- IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN	LUDA0820
C		APPLIED TO THIS CODE. NO OTHER WARRANTY,	LUDA0830
C		EXPRESSED OR IMPLIED, IS APPLICABLE.	LUDA0840
C			LUDA0850
C			LUDA0860
C			LUDA0870
C	SUBROUTINE LUDATF	(A,LU,N,IA,IDGT,D1,D2,IPVT,EQUIL,WA,IER)	LUDA0880
C			LUDA0890
C	DIMENSION	A(IA,1),LU(IA,1),IPVT(1),EQUIL(1)	LUDA0900
C	DOUBLE PRECISION	A,LU,D1,D2,EQUIL,WA,ZERO,ONE,FOUR,SIXTN,SIXTH,	LUDA0910
C	*	RN,WREL,BIGA,BIG,P,SUM,AI,WI,T,TEST,Q	LUDA0920
C	DATA	ZERO,ONE,FOUR,SIXTN,SIXTH/0.D0,1.D0,4.D0,	LUDA0930
C	*	16.D0,.0625D0/	LUDA0940
C		FIRST EXECUTABLE STATEMENT	LUDA0950
C		INITIALIZATION	LUDA0960
C			LUDA0970
C	IER = 0		LUDA0980
C	RN = N		LUDA0990
C	WREL = ZERO		LUDA1000
C	D1 = ONE		LUDA1010
C	D2 = ZERO		LUDA1020
C	BIGA = ZERO		LUDA1030
C	DO 10 I=1,N		LUDA1040
C	BIG = ZERO		LUDA1050
C	DO 5 J=1,N		LUDA1060
C	P = A(I,J)		LUDA1070
C	LU(I,J) = P		LUDA1080
C	P = DABS(P)		LUDA1090
C	IF (P .GT. BIG) BIG = P		LUDA1100
C	5 CONTINUE		LUDA1110
C	IF (BIG .GT. BIGA) BIGA = BIG		LUDA1120
C	IF (BIG .EQ. ZERO) GO TO 110		LUDA1130
C	EQUIL(I) = ONE/BIG		LUDA1140
C	10 CONTINUE		LUDA1150
C	DO 105 J=1,N		LUDA1160
C	JM1 = J-1		LUDA1170
C	IF (JM1 .LT. 1) GO TO 40		LUDA1180
C		COMPUTE U(I,J), I=1,...,J-1	LUDA1190
C	DO 35 I=1,JM1		LUDA1200
C	SUM = LU(I,J)		LUDA1210
C	IM1 = I-1		LUDA1220
C	IF (IDGT .EQ. 0) GO TO 25		LUDA1230
C		WITH ACCURACY TEST	LUDA1240
C	AI = DABS(SUM)		

	WI = ZERO	LUDA1250
	IF (IM1 .LT. 1) GO TO 20	LUDA1260
	DO 15 K=1,IM1	LUDA1270
	T = LU(I,K)*LU(K,J)	LUDA1280
	SUM = SUM-T	LUDA1290
	WI = WI+DABS(T)	LUDA1300
15	CONTINUE	LUDA1310
	LU(I,J) = SUM	LUDA1320
20	WI = WI+DABS(SUM)	LUDA1330
	IF (AI .EQ. ZERO) AI = BIGA	LUDA1340
	TEST = WI/AI	LUDA1350
	IF (TEST .GT. WREL) WREL = TEST	LUDA1360
	GO TO 35	LUDA1370
C	WITHOUT ACCURACY	LUDA1380
25	IF (IM1 .LT. 1) GO TO 35	LUDA1390
	DO 30 K=1,IM1	LUDA1400
	SUM = SUM-LU(I,K)*LU(K,J)	LUDA1410
30	CONTINUE	LUDA1420
	LU(I,J) = SUM	LUDA1430
35	CONTINUE	LUDA1440
40	P = ZERO	LUDA1450
C	COMPUTE U(J,J) AND L(I,J), I=J+1,...	LUDA1460
	DO 70 I=J,N	LUDA1470
	SUM = LU(I,J)	LUDA1480
	IF (IDGT .EQ. 0) GO TO 55	LUDA1490
C	WITH ACCURACY TEST	LUDA1500
	AI = DABS(SUM)	LUDA1510
	WI = ZERO	LUDA1520
	IF (JMI .LT. 1) GO TO 50	LUDA1530
	DO 45 K=1,JMI	LUDA1540
	T = LU(I,K)*LU(K,J)	LUDA1550
	SUM = SUM-T	LUDA1560
	WI = WI+DABS(T)	LUDA1570
45	CONTINUE	LUDA1580
	LU(I,J) = SUM	LUDA1590
50	WI = WI+DABS(SUM)	LUDA1600
	IF (AI .EQ. ZERO) AI = BIGA	LUDA1610
	TEST = WI/AI	LUDA1620
	IF (TEST .GT. WREL) WREL = TEST	LUDA1630
	GO TO 65	LUDA1640
C	WITHOUT ACCURACY TEST	LUDA1650
55	IF (JMI .LT. 1) GO TO 65	LUDA1660
	DO 60 K=1,JMI	LUDA1670
	SUM = SUM-LU(I,K)*LU(K,J)	LUDA1680
60	CONTINUE	LUDA1690
	LU(I,J) = SUM	LUDA1700
65	Q = EQUIL(I)*DABS(SUM)	LUDA1710
	IF (P .GE. Q) GO TO 70	LUDA1720
	P = Q	LUDA1730
	IMAX = I	LUDA1740
70	CONTINUE	LUDA1750
C	TEST FOR ALGORITHMIC SINGULARITY	LUDA1760
	IF (RN+P .EQ. RN) GO TO 110	LUDA1770
	IF (J .EQ. IMAX) GO TO 80	LUDA1780
C	INTERCHANGE ROWS J AND IMAX	LUDA1790
	D1 = -D1	LUDA1800
	DO 75 K=1,N	LUDA1810
	P = LU(IMAX,K)	LUDA1820
	LU(IMAX,K) = LU(J,K)	LUDA1830
	LU(J,K) = P	LUDA1840
75	CONTINUE	LUDA1850
	EQUIL(IMAX) = EQUIL(J)	LUDA1860

80	IPVT(J) = IMAX	LUDA1870
	D1 = D1*LU(J,J)	LUDA1880
85	IF (DABS(D1) .LE. ONE) GO TO 90	LUDA1890
	D1 = D1*SIXTH	LUDA1900
	D2 = D2+FOUR	LUDA1910
	GO TO 85	LUDA1920
90	IF (DABS(D1) .GE. SIXTH) GO TO 95	LUDA1930
	D1 = D1*SIXTH	LUDA1940
	D2 = D2-FOUR	LUDA1950
	GO TO 90	LUDA1960
95	CONTINUE	LUDA1970
	JP1 = J+1	LUDA1980
	IF (JP1 .GT. N) GO TO 105	LUDA1990
C		LUDA2000
	DIVIDE BY PIVOT ELEMENT U(J,J)	LUDA2010
	P = LU(J,J)	LUDA2020
	DO 100 I=JP1,N	LUDA2030
	LU(I,J) = LU(I,J)/P	LUDA2040
100	CONTINUE	LUDA2050
105	CONTINUE	LUDA2060
C		LUDA2070
	PERFORM ACCURACY TEST	LUDA2080
	IF (IDGT .EQ. 0) GO TO 9005	LUDA2090
	P = 3*N+3	LUDA2100
	WA = P*WREL	LUDA2110
	IF (WA+10.D0**(-IDGT) .NE. WA) GO TO 9005	LUDA2120
	IER = 34	LUDA2130
	GO TO 9000	LUDA2140
C		LUDA2150
	ALGORITHMIC SINGULARITY	LUDA2160
110	IER = 129	LUDA2170
	D1 = ZERO	LUDA2180
	D2 = ZERO	LUDA2190
9000	CONTINUE	LUDA2200
C		LUDA2210
	PRINT ERROR	
	CALL UERTST(IER,6H LUDATF)	
9005	RETURN	
	END	

C	IMSL ROUTINE NAME	- LUELMF	LUEF0010
C			LUEF0020
C	-----		LUEF0030
C	COMPUTER	- IBM/DOUBLE	LUEF0040
C			LUEF0050
C	LATEST REVISION	- JANUARY 1, 1978	LUEF0060
C			LUEF0070
C	PURPOSE	- ELIMINATION PART OF SOLUTION OF AX=B	LUEF0080
C		(FULL STORAGE MODE)	LUEF0090
C			LUEF0100
C	USAGE	- CALL LUELMF (A,B,IPVT,N,IA,X)	LUEF0110
C			LUEF0120
C	ARGUMENTS	A	LUEF0130
C		- A = LU (THE RESULT COMPUTED IN THE IMSL	LUEF0140
C		ROUTINE LUDATF) WHERE L IS A LOWER	LUEF0150
C		TRIANGULAR MATRIX WITH ONES ON THE MAIN	LUEF0160
C		DIAGONAL. U IS UPPER TRIANGULAR. L AND U	LUEF0170
C		ARE STORED AS A SINGLE MATRIX A AND THE	LUEF0180
C		UNIT DIAGONAL OF L IS NOT STORED. (INPUT)	LUEF0190
C		B	LUEF0200
C		- B IS A VECTOR OF LENGTH N ON THE RIGHT HAND	LUEF0210
C		SIDE OF THE EQUATION AX=B. (INPUT)	LUEF0220
C		IPVT	LUEF0230
C		- THE PERMUTATION MATRIX RETURNED FROM THE	LUEF0240
C		IMSL ROUTINE LUDATF, STORED AS AN N LENGTH	LUEF0250
C		VECTOR. (INPUT)	LUEF0260
C		N	LUEF0270
C		- ORDER OF A AND NUMBER OF ROWS IN B. (INPUT)	LUEF0280
C		IA	LUEF0290
C		- ROW DIMENSION OF A EXACTLY AS SPECIFIED IN	LUEF0300
C		THE DIMENSION STATEMENT IN THE CALLING	LUEF0310
C		PROGRAM. (INPUT)	LUEF0320
C		X	LUEF0330
C		- THE RESULT X. (OUTPUT)	LUEF0340
C	PRECISION/HARDWARE	- SINGLE AND DOUBLE/H32	LUEF0350
C		- SINGLE/H36,H48,H60	LUEF0360
C			LUEF0370
C	REQD. IMSL ROUTINES	- NONE REQUIRED	LUEF0380
C			LUEF0390
C	NOTATION	- INFORMATION ON SPECIAL NOTATION AND	LUEF0400
C		CONVENTIONS IS AVAILABLE IN THE MANUAL	LUEF0410
C		INTRODUCTION OR THROUGH IMSL ROUTINE UHELP	LUEF0420
C			LUEF0430
C	COPYRIGHT	- 1978 BY IMSL, INC. ALL RIGHTS RESERVED.	LUEF0440
C			LUEF0450
C	WARRANTY	- IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN	LUEF0460
C		APPLIED TO THIS CODE. NO OTHER WARRANTY,	LUEF0470
C		EXPRESSED OR IMPLIED, IS APPLICABLE.	LUEF0480
C	-----		LUEF0490
C	SUBROUTINE LUELMF (A,B,IPVT,N,IA,X)		LUEF0500
C	DIMENSION	A(IA,1),B(1),IPVT(1),X(1)	LUEF0510
C	DOUBLE PRECISION	A,B,X,SUM	LUEF0520
C		FIRST EXECUTABLE STATEMENT	LUEF0530
C		SOLVE LY = B FOR Y	LUEF0540
	DO 5 I=1,N		LUEF0550
	5 X(I) = B(I)		LUEF0560
	IW = 0		LUEF0570
	DO 20 I=1,N		LUEF0580
	IP = IPVT(I)		LUEF0590
	SUM = X(IP)		LUEF0600
	X(IP) = X(I)		LUEF0610
	IF (IW .EQ. 0) GO TO 15		LUEF0620
	IM1 = I-1		

```

        DO 10 J=IW,IM1
          SUM = SUM-A(I,J)*X(J)
10      CONTINUE
        GO TO 20
15      IF (SUM .NE. 0.D0) IW = I
20      X(I) = SUM
C
        DO 30 IB=1,N
          I = N+1-IB
          IP1 = I+1
          SUM = X(I)
          IF (IP1 .GT. N) GO TO 30
          DO 25 J=IP1,N
            SUM = SUM-A(I,J)*X(J)
25      CONTINUE
30      X(I) = SUM/A(I,I)
        RETURN
      END

```

```

LUEF0630
LUEF0640
LUEF0650
LUEF0660
LUEF0670
LUEF0680
LUEF0690
LUEF0700
LUEF0710
LUEF0720
LUEF0730
LUEF0740
LUEF0750
LUEF0760
LUEF0770
LUEF0780
LUEF0790
LUEF0800

```

C	IMSL ROUTINE NAME	- LEQT1B	LE1B0010
C			LE1B0020
C	-----		LE1B0030
C	COMPUTER	- IBM/DOUBLE	LE1B0040
C			LE1B0050
C	LATEST REVISION	- JANUARY 1, 1978	LE1B0060
C			LE1B0070
C	PURPOSE	- LINEAR EQUATION SOLUTION - BAND STORAGE	LE1B0080
C		MODE - SPACE ECONOMIZER SOLUTION	LE1B0090
C			LE1B0100
C	USAGE	- CALL LEQT1B (A,N,NLC,NUC,IA,B,M,IB,IJOB,XL,	LE1B0110
C		IER)	LE1B0120
C			LE1B0130
C	ARGUMENTS	A	LE1B0140
C		- INPUT/OUTPUT MATRIX OF DIMENSION N BY	LE1B0150
C		(NUC+NLC+1). SEE PARAMETER IJOB.	LE1B0160
C		N	LE1B0170
C		- ORDER OF MATRIX A AND THE NUMBER OF ROWS IN	LE1B0180
C		B. (INPUT)	LE1B0190
C		NLC	LE1B0200
C		- NUMBER OF LOWER CODIAGONALS IN MATRIX A.	LE1B0210
C		(INPUT)	LE1B0220
C		NUC	LE1B0230
C		- NUMBER OF UPPER CODIAGONALS IN MATRIX A.	LE1B0240
C		(INPUT)	LE1B0250
C		IA	LE1B0260
C		- ROW DIMENSION OF MATRIX A EXACTLY AS	LE1B0270
C		SPECIFIED IN THE DIMENSION STATEMENT IN THE	LE1B0280
C		CALLING PROGRAM. (INPUT)	LE1B0290
C		B	LE1B0300
C		- INPUT/OUTPUT MATRIX OF DIMENSION N BY M.	LE1B0310
C		ON INPUT, B CONTAINS THE M RIGHT-HAND SIDES	LE1B0320
C		OF THE EQUATION $AX = B$. ON OUTPUT, THE	LE1B0330
C		SOLUTION MATRIX X REPLACES B. IF IJOB = 1,	LE1B0340
C		B IS NOT USED.	LE1B0350
C		M	LE1B0360
C		- NUMBER OF RIGHT HAND SIDES (COLUMNS IN B).	LE1B0370
C		(INPUT)	LE1B0380
C		IB	LE1B0390
C		- ROW DIMENSION OF MATRIX B EXACTLY AS	LE1B0400
C		SPECIFIED IN THE DIMENSION STATEMENT IN THE	LE1B0410
C		CALLING PROGRAM. (INPUT)	LE1B0420
C		IJOB	LE1B0430
C		- INPUT OPTION PARAMETER. IJOB = I IMPLIES WHEN	LE1B0440
C		I = 0, FACTOR THE MATRIX A AND SOLVE THE	LE1B0450
C		EQUATION $AX = B$. ON INPUT, A CONTAINS THE	LE1B0460
C		COEFFICIENT MATRIX OF THE EQUATION $AX = B$,	LE1B0470
C		WHERE A IS ASSUMED TO BE AN N BY N BAND	LE1B0480
C		MATRIX. A IS STORED IN BAND STORAGE MODE	LE1B0490
C		AND THEREFORE HAS DIMENSION N BY	LE1B0500
C		(NLC+NUC+1). ON OUTPUT, A IS REPLACED	LE1B0510
C		BY THE U MATRIX OF THE L-U DECOMPOSITION	LE1B0520
C		OF A ROWWISE PERMUTATION OF MATRIX A. U	LE1B0530
C		IS STORED IN BAND STORAGE MODE.	LE1B0540
C		I = 1, FACTOR THE MATRIX A. A CONTAINS THE	LE1B0550
C		SAME INPUT/OUTPUT INFORMATION AS IF	LE1B0560
C		IJOB = 0.	LE1B0570
C		I = 2, SOLVE THE EQUATION $AX = B$. THIS	LE1B0580
C		OPTION IMPLIES THAT LEQT1B HAS ALREADY	LE1B0590
C		BEEN CALLED USING IJOB = 0 OR 1 SO THAT	LE1B0600
C		THE MATRIX A HAS ALREADY BEEN FACTORED.	LE1B0610
C		IN THIS CASE, OUTPUT MATRICES A AND XL	LE1B0620
C		MUST HAVE BEEN SAVED FOR REUSE IN THE	
C		CALL TO LEQT1B.	
C		XL	
C		- WORK AREA OF DIMENSION $N*(NLC+1)$. THE FIRST	
C		$NLC*N$ LOCATIONS OF XL CONTAIN COMPONENTS OF	
C		THE L MATRIX OF THE L-U DECOMPOSITION OF A	
C		ROWWISE PERMUTATION OF A. THE LAST N	
C		LOCATIONS CONTAIN THE PIVOT INDICES.	
C		IER	
C		- ERROR PARAMETER. (OUTPUT)	

C	TERMINAL ERROR	LE1B0630
C	IER = 129 INDICATES THAT MATRIX A IS	LE1B0640
C	ALGORITHMICALLY SINGULAR. (SEE THE	LE1B0650
C	CHAPTER L PRELUDE).	LE1B0660
C	PRECISION/HARDWARE - SINGLE AND DOUBLE/H32	LE1B0670
C	- SINGLE/H36,H48,H60	LE1B0680
C	REQD. IMSL ROUTINES - UERTST,UGETIO	LE1B0690
C		LE1B0700
C	NOTATION	LE1B0710
C	- INFORMATION ON SPECIAL NOTATION AND	LE1B0720
C	CONVENTIONS IS AVAILABLE IN THE MANUAL	LE1B0730
C	INTRODUCTION OR THROUGH IMSL ROUTINE UHELP	LE1B0740
C		LE1B0750
C	COPYRIGHT	LE1B0760
C	- 1978 BY IMSL, INC. ALL RIGHTS RESERVED.	LE1B0770
C		LE1B0780
C	WARRANTY	LE1B0790
C	- IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN	LE1B0800
C	APPLIED TO THIS CODE. NO OTHER WARRANTY,	LE1B0810
C	EXPRESSED OR IMPLIED, IS APPLICABLE.	LE1B0820
C	-----	LE1B0830
C	SUBROUTINE LEQT1B (A,N,NLC,NUC,IA,B,M,IB,IJOB,XL,IER)	LE1B0840
C		LE1B0850
C	DIMENSION A(IA,1),XL(N,1),B(IB,1)	LE1B0860
C	DOUBLE PRECISION A,XL,B,P,Q,ZERO,ONE,RN	LE1B0870
C	DATA ZERO/0.D0/,ONE/1.0D0/	LE1B0880
C		LE1B0890
C	FIRST EXECUTABLE STATEMENT	LE1B0900
C	IER = 0	LE1B0910
C	JBEG = NLC+1	LE1B0920
C	NLC1 = JBEG	LE1B0930
C	IF (IJOB .EQ. 2) GO TO 80	LE1B0940
C	RN = N	LE1B0950
C		LE1B0960
C	RESTRUCTURE THE MATRIX	LE1B0970
C	FIND RECIPROCAL OF THE LARGEST	LE1B0980
C	ABSOLUTE VALUE IN ROW I	LE1B0990
C		LE1B1000
C	I = 1	LE1B1010
C	NC = JBEG+NUC	LE1B1020
C	NN = NC	LE1B1030
C	JEND = NC	LE1B1040
C	IF (N .EQ. 1 .OR. NLC .EQ. 0) GO TO 25	LE1B1050
C	5 K = 1	LE1B1060
C	P = ZERO	LE1B1070
C	DO 10 J = JBEG,JEND	LE1B1080
C	A(I,K) = A(I,J)	LE1B1090
C	Q = DABS(A(I,K))	LE1B1100
C	IF (Q .GT. P) P = Q	LE1B1110
C	K = K+1	LE1B1120
C	10 CONTINUE	LE1B1130
C	IF (P .EQ. ZERO) GO TO 135	LE1B1140
C	XL(I,NLC1) = ONE/P	LE1B1150
C	IF (K .GT. NC) GO TO 20	LE1B1160
C	DO 15 J = K,NC	LE1B1170
C	A(I,J) = ZERO	LE1B1180
C	15 CONTINUE	LE1B1190
C	20 I = I+1	LE1B1200
C	JBEG = JBEG-1	LE1B1210
C	IF (JEND-JBEG .EQ. N) JEND = JEND-1	LE1B1220
C	IF (I .LE. NLC) GO TO 5	LE1B1230
C	JBEG = I	LE1B1240
C	NN = JEND	
C	25 JEND = N-NUC	

DO 40 I = JBEG,N	LE1B1250
P = ZERO	LE1B1260
DO 30 J = 1,NN	LE1B1270
Q = DABS(A(I,J))	LE1B1280
IF (Q .GT. P) P = Q	LE1B1290
30 CONTINUE	LE1B1300
IF (P .EQ. ZERO) GO TO 135	LE1B1310
XL(I,NLC1) = ONE/P	LE1B1320
IF (I .EQ. JEND) GO TO 37	LE1B1330
IF (I .LT. JEND) GO TO 40	LE1B1340
K = NN+1	LE1B1350
DO 35 J = K,NC	LE1B1360
A(I,J) = ZERO	LE1B1370
35 CONTINUE	LE1B1380
37 NN = NN-1	LE1B1390
40 CONTINUE	LE1B1400
L = NLC	LE1B1410
C L-U DECOMPOSITION	LE1B1420
DO 75 K = 1,N	LE1B1430
P = DABS(A(K,1))*XL(K,NLC1)	LE1B1440
I = K	LE1B1450
IF (L .LT. N) L = L+1	LE1B1460
K1 = K+1	LE1B1470
IF (K1 .GT. L) GO TO 50	LE1B1480
DO 45 J = K1,L	LE1B1490
Q = DABS(A(J,1))*XL(J,NLC1)	LE1B1500
IF (Q .LE. P) GO TO 45	LE1B1510
P = Q	LE1B1520
I = J	LE1B1530
45 CONTINUE	LE1B1540
50 XL(I,NLC1) = XL(K,NLC1)	LE1B1550
XL(K,NLC1) = I	LE1B1560
C SINGULARITY FOUND	LE1B1570
Q = RN+P	LE1B1580
IF (Q .EQ. RN) GO TO 135	LE1B1590
C INTERCHANGE ROWS I AND K	LE1B1600
IF (K .EQ. I) GO TO 60	LE1B1610
DO 55 J = 1,NC	LE1B1620
P = A(K,J)	LE1B1630
A(K,J) = A(I,J)	LE1B1640
A(I,J) = P	LE1B1650
55 CONTINUE	LE1B1660
60 IF (K1 .GT. L) GO TO 75	LE1B1670
DO 70 I = K1,L	LE1B1680
P = A(I,1)/A(K,1)	LE1B1690
IK = I-K	LE1B1700
XL(K1,IK) = P	LE1B1710
DO 65 J = 2,NC	LE1B1720
A(I,J-1) = A(I,J) - P*A(K,J)	LE1B1730
65 CONTINUE	LE1B1740
A(I,NC) = ZERO	LE1B1750
70 CONTINUE	LE1B1760
75 CONTINUE	LE1B1770
IF (IJOB .EQ. 1) GO TO 9005	LE1B1780
C FORWARD SUBSTITUTION	LE1B1790
80 L = NLC	LE1B1800
DO 105 K = 1,N	LE1B1810
I = XL(K,NLC1)	LE1B1820
IF (I .EQ. K) GO TO 90	LE1B1830
DO 85 J = 1,M	LE1B1840
P = B(K,J)	LE1B1850
B(K,J) = B(I,J)	LE1B1860

	B(I,J) = P	LE1B1870
85	CONTINUE	LE1B1880
90	IF (L .LT. N) L = L+1	LE1B1890
	K1 = K+1	LE1B1900
	IF (K1 .GT. L) GO TO 105	LE1B1910
	DO 100 I = K1,L	LE1B1920
	IK = I-K	LE1B1930
	P = XL(K1,IK)	LE1B1940
	DO 95 J = 1,M	LE1B1950
	B(I,J) = B(I,J) - P*B(K,J)	LE1B1960
95	CONTINUE	LE1B1970
100	CONTINUE	LE1B1980
105	CONTINUE	LE1B1990
C		LE1B2000
	JBEG = NUC+NLC	LE1B2010
	DO 125 J = 1,M	LE1B2020
	L = 1	LE1B2030
	K1 = N+1	LE1B2040
	DO 120 I = 1,N	LE1B2050
	K = K1-I	LE1B2060
	P = B(K,J)	LE1B2070
	IF (L .EQ. 1) GO TO 115	LE1B2080
	DO 110 KK = 2,L	LE1B2090
	IK = KK+K	LE1B2100
	P = P-A(K,KK)*B(IK-1,J)	LE1B2110
110	CONTINUE	LE1B2120
115	B(K,J) = P/A(K,1)	LE1B2130
	IF (L .LE. JBEG) L = L+1	LE1B2140
120	CONTINUE	LE1B2150
125	CONTINUE	LE1B2160
	GO TO 9005	LE1B2170
135	IER = 129	LE1B2180
9000	CONTINUE	LE1B2190
	CALL UERTST(IER,6HLEQT1B)	LE1B2200
9005	RETURN	LE1B2210
	END	LE1B2220

BACKWARD SUBSTITUTION

C	IMSL ROUTINE NAME	- UERTST	UERT0010
C			UERT0020
C			UERT0030
C	COMPUTER	- IBM/SINGLE	UERT0040
C			UERT0050
C	LATEST REVISION	- JUNE 1, 1982	UERT0060
C			UERT0070
C	PURPOSE	- PRINT A MESSAGE REFLECTING AN ERROR CONDITION	UERT0080
C			UERT0090
C	USAGE	- CALL UERTST (IER,NAME)	UERT0100
C			UERT0110
C			UERT0120
C	ARGUMENTS	IER - ERROR PARAMETER. (INPUT)	UERT0130
C		IER = I+J WHERE	UERT0140
C		I = 128 IMPLIES TERMINAL ERROR MESSAGE,	UERT0150
C		I = 64 IMPLIES WARNING WITH FIX MESSAGE,	UERT0160
C		I = 32 IMPLIES WARNING MESSAGE.	UERT0170
C		J = ERROR CODE RELEVANT TO CALLING	UERT0180
C		ROUTINE.	UERT0190
C		NAME - A CHARACTER STRING OF LENGTH SIX PROVIDING	UERT0200
C		THE NAME OF THE CALLING ROUTINE. (INPUT)	UERT0210
C			UERT0220
C	PRECISION/HARDWARE	- SINGLE/ALL	UERT0230
C			UERT0240
C	REQD. IMSL ROUTINES	- UGETIO,USPKD	UERT0250
C			UERT0260
C	NOTATION	- INFORMATION ON SPECIAL NOTATION AND	UERT0270
C		CONVENTIONS IS AVAILABLE IN THE MANUAL	UERT0280
C		INTRODUCTION OR THROUGH IMSL ROUTINE UHELP	UERT0290
C			UERT0300
C	REMARKS	THE ERROR MESSAGE PRODUCED BY UERTST IS WRITTEN	UERT0310
C		TO THE STANDARD OUTPUT UNIT. THE OUTPUT UNIT	UERT0320
C		NUMBER CAN BE DETERMINED BY CALLING UGETIO AS	UERT0330
C		FOLLOWS.. CALL UGETIO(1,NIN,NOUT).	UERT0340
C		THE OUTPUT UNIT NUMBER CAN BE CHANGED BY CALLING	UERT0350
C		UGETIO AS FOLLOWS..	UERT0360
C		NIN = 0	UERT0370
C		NOUT = NEW OUTPUT UNIT NUMBER	UERT0380
C		CALL UGETIO(3,NIN,NOUT)	UERT0390
C		SEE THE UGETIO DOCUMENT FOR MORE DETAILS.	UERT0400
C			UERT0410
C	COPYRIGHT	- 1982 BY IMSL, INC. ALL RIGHTS RESERVED.	UERT0420
C			UERT0430
C	WARRANTY	- IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN	UERT0440
C		APPLIED TO THIS CODE. NO OTHER WARRANTY,	UERT0450
C		EXPRESSED OR IMPLIED, IS APPLICABLE.	UERT0460
C			UERT0470
C			UERT0480
C			UERT0490
C	SUBROUTINE UERTST (IER,NAME)		UERT0500
C		SPECIFICATIONS FOR ARGUMENTS	UERT0510
C	INTEGER	IER	UERT0520
C	INTEGER	NAME(1)	UERT0530
C		SPECIFICATIONS FOR LOCAL VARIABLES	UERT0540
C	INTEGER	I, IEQ, IEQDF, IOUNIT, LEVEL, LEVOLD, NAMEQ(6),	UERT0550
C	*	NAMSET(6), NAMUPK(6), NIN, NMTB	UERT0560
C	DATA	NAMSET/1HU, 1HE, 1HR, 1HS, 1HE, 1HT/	UERT0570
C	DATA	NAMEQ/6*1H /	UERT0580
C	DATA	LEVEL/4/, IEQDF/0/, IEQ/1H= /	UERT0590
C		UNPACK NAME INTO NAMUPK	UERT0600
C		FIRST EXECUTABLE STATEMENT	UERT0610
C	CALL USPKD (NAME,6,NAMUPK,NMTB)		UERT0620

C	CALL UGETIO(1,NIN,IOUNIT)	GET OUTPUT UNIT NUMBER	UERT0630
C		CHECK IER	UERT0640
	IF (IER.GT.999) GO TO 25		UERT0650
	IF (IER.LT.-32) GO TO 55		UERT0660
	IF (IER.LE.128) GO TO 5		UERT0670
	IF (LEVEL.LT.1) GO TO 30		UERT0680
C		PRINT TERMINAL MESSAGE	UERT0690
	IF (IEQDF.EQ.1) WRITE(IOUNIT,35) IER,NAMEQ,IEQ,NAMUPK		UERT0700
	IF (IEQDF.EQ.0) WRITE(IOUNIT,35) IER,NAMUPK		UERT0710
	GO TO 30		UERT0720
5	IF (IER.LE.64) GO TO 10		UERT0730
	IF (LEVEL.LT.2) GO TO 30		UERT0740
C		PRINT WARNING WITH FIX MESSAGE	UERT0750
	IF (IEQDF.EQ.1) WRITE(IOUNIT,40) IER,NAMEQ,IEQ,NAMUPK		UERT0760
	IF (IEQDF.EQ.0) WRITE(IOUNIT,40) IER,NAMUPK		UERT0770
	GO TO 30		UERT0780
10	IF (IER.LE.32) GO TO 15		UERT0790
C		PRINT WARNING MESSAGE	UERT0800
	IF (LEVEL.LT.3) GO TO 30		UERT0810
	IF (IEQDF.EQ.1) WRITE(IOUNIT,45) IER,NAMEQ,IEQ,NAMUPK		UERT0820
	IF (IEQDF.EQ.0) WRITE(IOUNIT,45) IER,NAMUPK		UERT0830
	GO TO 30		UERT0840
15	CONTINUE		UERT0850
C		CHECK FOR UERSET CALL	UERT0860
	DO 20 I=1,6		UERT0870
	IF (NAMUPK(I).NE.NAMSET(I)) GO TO 25		UERT0880
20	CONTINUE		UERT0890
	LEVOLD = LEVEL		UERT0900
	LEVEL = IER		UERT0910
	IER = LEVOLD		UERT0920
	IF (LEVEL.LT.0) LEVEL = 4		UERT0930
	IF (LEVEL.GT.4) LEVEL = 4		UERT0940
	GO TO 30		UERT0950
25	CONTINUE		UERT0960
	IF (LEVEL.LT.4) GO TO 30		UERT0970
C		PRINT NON-DEFINED MESSAGE	UERT0980
	IF (IEQDF.EQ.1) WRITE(IOUNIT,50) IER,NAMEQ,IEQ,NAMUPK		UERT0990
	IF (IEQDF.EQ.0) WRITE(IOUNIT,50) IER,NAMUPK		UERT1000
30	IEQDF = 0		UERT1010
	RETURN		UERT1020
35	FORMAT(19H *** TERMINAL ERROR,10X,7H(IER = ,I3,		UERT1030
1	20H) FROM IMSL ROUTINE ,6A1,A1,6A1)		UERT1040
40	FORMAT(27H *** WARNING WITH FIX ERROR,2X,7H(IER = ,I3,		UERT1050
1	20H) FROM IMSL ROUTINE ,6A1,A1,6A1)		UERT1060
45	FORMAT(18H *** WARNING ERROR,11X,7H(IER = ,I3,		UERT1070
1	20H) FROM IMSL ROUTINE ,6A1,A1,6A1)		UERT1080
50	FORMAT(20H *** UNDEFINED ERROR,9X,7H(IER = ,I5,		UERT1090
1	20H) FROM IMSL ROUTINE ,6A1,A1,6A1)		UERT1100
C		SAVE P FOR P = R CASE	UERT1110
C		P IS THE PAGE NAMUPK	UERT1120
C		R IS THE ROUTINE NAMUPK	UERT1130
55	IEQDF = 1		UERT1140
	DO 60 I=1,6		UERT1150
60	NAMEQ(I) = NAMUPK(I)		UERT1160
65	RETURN		UERT1170
	END		UERT1180
			UERT1190
			UERT1200

C	IMSL ROUTINE NAME	- UGETIO	UGET0010
C			UGET0020
C	-----		UGET0030
C	COMPUTER	- IBM/SINGLE	UGET0040
C			UGET0050
C	LATEST REVISION	- JUNE 1, 1981	UGET0060
C			UGET0070
C	PURPOSE	- TO RETRIEVE CURRENT VALUES AND TO SET NEW VALUES FOR INPUT AND OUTPUT UNIT IDENTIFIERS.	UGET0080
C			UGET0090
C	USAGE	- CALL UGETIO(IOPT,NIN,NOUT)	UGET0100
C			UGET0110
C	ARGUMENTS	IOPT - OPTION PARAMETER. (INPUT)	UGET0120
C		IF IOPT=1, THE CURRENT INPUT AND OUTPUT UNIT IDENTIFIER VALUES ARE RETURNED IN NIN AND NOUT, RESPECTIVELY.	UGET0130
C		IF IOPT=2, THE INTERNAL VALUE OF NIN IS RESET FOR SUBSEQUENT USE.	UGET0140
C		IF IOPT=3, THE INTERNAL VALUE OF NOUT IS RESET FOR SUBSEQUENT USE.	UGET0150
C		NIN - INPUT UNIT IDENTIFIER.	UGET0160
C		OUTPUT IF IOPT=1, INPUT IF IOPT=2.	UGET0170
C		NOUT - OUTPUT UNIT IDENTIFIER.	UGET0180
C		OUTPUT IF IOPT=1, INPUT IF IOPT=3.	UGET0190
C			UGET0200
C	PRECISION/HARDWARE	- SINGLE/ALL	UGET0210
C			UGET0220
C	REQD. IMSL ROUTINES	- NONE REQUIRED	UGET0230
C			UGET0240
C	NOTATION	- INFORMATION ON SPECIAL NOTATION AND CONVENTIONS IS AVAILABLE IN THE MANUAL INTRODUCTION OR THROUGH IMSL ROUTINE UHELP	UGET0250
C			UGET0260
C	REMARKS	EACH IMSL ROUTINE THAT PERFORMS INPUT AND/OR OUTPUT OPERATIONS CALLS UGETIO TO OBTAIN THE CURRENT U IDENTIFIER VALUES. IF UGETIO IS CALLED WITH IOPT=1 OR IOPT=3, NEW UNIT IDENTIFIER VALUES ARE ESTABLISHED. SUBSEQUENT INPUT/OUTPUT IS PERFORMED ON THE NEW UNITS.	UGET0270
C			UGET0280
C	COPYRIGHT	- 1978 BY IMSL, INC. ALL RIGHTS RESERVED.	UGET0290
C			UGET0300
C	WARRANTY	- IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN APPLIED TO THIS CODE. NO OTHER WARRANTY, EXPRESSED OR IMPLIED, IS APPLICABLE.	UGET0310
C			UGET0320
C	-----		UGET0330
C	SUBROUTINE UGETIO(IOPT,NIN,NOUT)		UGET0340
C		SPECIFICATIONS FOR ARGUMENTS	UGET0350
C	INTEGER	IOPT,NIN,NOUT	UGET0360
C		SPECIFICATIONS FOR LOCAL VARIABLES	UGET0370
C	INTEGER	NIND,NOUTD	UGET0380
C	DATA	NIND/5/,NOUTD/6/	UGET0390
C		FIRST EXECUTABLE STATEMENT	UGET0400
	IF (IOPT.EQ.3) GO TO 10		UGET0410
	IF (IOPT.EQ.2) GO TO 5		UGET0420
	IF (IOPT.NE.1) GO TO 9005		UGET0430
	NIN = NIND		UGET0440
	NOUT = NOUTD		UGET0450
	GO TO 9005		UGET0460
			UGET0470
			UGET0480
			UGET0490
			UGET0500
			UGET0510
			UGET0520
			UGET0530
			UGET0540
			UGET0550
			UGET0560
			UGET0570
			UGET0580
			UGET0590
			UGET0600
			UGET0610
			UGET0620

5 NIND = NIN
GO TO 9005
10 NOUTD = NOUT
9005 RETURN
END

UGET0630
UGET0640
UGET0650
UGET0660
UGET0670

C	IMSL ROUTINE NAME	- USPDK	USPK0010
C			USPK0020
C	-----		USPK0030
C	COMPUTER	- IBM/SINGLE	USPK0040
C			USPK0050
C	LATEST REVISION	- NOVEMBER 1, 1984	USPK0060
C			USPK0070
C	PURPOSE	- NUCLEUS CALLED BY IMSL ROUTINES THAT HAVE CHARACTER STRING ARGUMENTS	USPK0080
C			USPK0090
C	USAGE	- CALL USPDK (PACKED, NCHARS, UNPAKD, NCHMTB)	USPK0100
C			USPK0110
C	ARGUMENTS	PACKED - CHARACTER STRING TO BE UNPACKED. (INPUT)	USPK0120
C		NCHARS - LENGTH OF PACKED. (INPUT) SEE REMARKS.	USPK0130
C		UNPAKD - INTEGER ARRAY TO RECEIVE THE UNPACKED REPRESENTATION OF THE STRING. (OUTPUT)	USPK0140
C		. NCHMTB - NCHARS MINUS TRAILING BLANKS. (OUTPUT)	USPK0150
C			USPK0160
C			USPK0170
C			USPK0180
C	PRECISION/HARDWARE	- SINGLE/ALL	USPK0190
C			USPK0200
C	REQD. IMSL ROUTINES	- NONE	USPK0210
C			USPK0220
C	REMARKS	1. USPDK UNPACKS A CHARACTER STRING INTO AN INTEGER ARRAY IN (A1) FORMAT.	USPK0230
C		2. UP TO 129 CHARACTERS MAY BE USED. ANY IN EXCESS OF THAT ARE IGNORED.	USPK0240
C			USPK0250
C			USPK0260
C			USPK0270
C	COPYRIGHT	- 1984 BY IMSL, INC. ALL RIGHTS RESERVED.	USPK0280
C			USPK0290
C	WARRANTY	- IMSL WARRANTS ONLY THAT IMSL TESTING HAS BEEN APPLIED TO THIS CODE. NO OTHER WARRANTY, EXPRESSED OR IMPLIED, IS APPLICABLE.	USPK0300
C			USPK0310
C			USPK0320
C			USPK0330
C			USPK0340
C	-----		USPK0350
C	SUBROUTINE USPDK	(PACKED, NCHARS, UNPAKD, NCHMTB)	USPK0360
C		SPECIFICATIONS FOR ARGUMENTS	USPK0370
C	INTEGER	NC, NCHARS, NCHMTB	USPK0380
C			USPK0390
C	LOGICAL*1	UNPAKD(1), PACKED(1), LBYTE, LBLANK	USPK0400
C	INTEGER*2	IBYTE, IBLANK	USPK0410
C	EQUIVALENCE	(LBYTE, IBYTE)	USPK0420
C	DATA	LBLANK /1H /	USPK0430
C	DATA	IBYTE /1H /	USPK0440
C	DATA	IBLANK /1H /	USPK0450
C		INITIALIZE NCHMTB	USPK0460
C	NCHMTB = 0		USPK0470
C		RETURN IF NCHARS IS LE ZERO	USPK0480
C	IF (NCHARS.LE.0) RETURN		USPK0490
C		SET NC=NUMBER OF CHARS TO BE DECODED	USPK0500
C	NC = MIN0 (129, NCHARS)		USPK0510
C	NWORDS = NC*4		USPK0520
C	J = 1		USPK0530
C	DO 110 I = 1, NWORDS, 4		USPK0540
C	UNPAKD(I) = PACKED(J)		USPK0550
C	UNPAKD(I+1) = LBLANK		USPK0560
C	UNPAKD(I+2) = LBLANK		USPK0570
C	UNPAKD(I+3) = LBLANK		USPK0580
C	110 J = J+1		USPK0590
C		CHECK UNPAKD ARRAY AND SET NCHMTB	USPK0600
C		BASED ON TRAILING BLANKS FOUND	USPK0610
C	DO 200 N = 1, NWORDS, 4		USPK0620


```
      NN = NWORDS - N - 2
      LBYTE = UNPAKD(NN)
      IF (LBYTE .NE. IBLANK) GO TO 210
200  CONTINUE
      NN = 0
210  NCHMTB = (NN + 3) / 4
      RETURN
      END
```

```
USPK0630
USPK0640
USPK0650
USPK0660
USPK0670
USPK0680
USPK0690
USPK0700
```

LIST OF REFERENCES

1. von Braun, W. and Ordway, F., *History of Rocketry and Space Travel*, 3d ed., T. Crowell Co., 1974.
2. Sutton, G.P., *Rocket Propulsion Elements*, 4th ed., John Wiley & Sons, 1976.
3. Jahn, R.G., *Physics of Electric Propulsion*, McGraw-Hill, 1968.
4. Sutton, G.P., *Rocket Propulsion Elements*, 5th ed., John Wiley & Sons, 1986.
5. Stuhlinger, E., *Ion Propulsion for Space Flight*, McGraw-Hill, 1964.
6. Myers, R.M., Mantenicks, M.A., and LaPointe, M.R., *MPD Thruster Technology*, NASA-TM-105242, AIAA-91-3568, Sept., 1968.
7. Myers, R.M., *Applied-Field MPD Thruster Geometry Effects*, AIAA-91-2342, June, 1991.
8. Saber, A.J., "Anode Power in the Quasi-Steady MPD (Magnetoplasdynamic) Thruster," Ph.D. Thesis, Princeton Univ., NJ, 1974.
9. Shih, K.T., Pfender, E., Ibele, W.E., and Eckert, E.R.G., "Experimental Anode Heat-Transfer Studies in a Coaxial Arc Configuration," *AIAA Journal*, V. 6, No. 8, pp.1482-1487, August, 1968.
10. Sanders, N.A., and Pfender, E., "Measurement of the Anode Falls and Anode Heat Transfer in Atmospheric Pressure, High Intensity Arcs," *J. Appl. Phys.*, V. 55, No. 3, pp. 714-722, Feb., 1984.
11. Vainberg, L.I., Lyubimov, G.A., and Smolin, G.G., "High Current Discharge Effects and Anode Damage in an End-Fire Plasma Accelerator," *Sov. Physics, Tech. Phys.*, V. 23, No. 4, pp. 439-443, April, 1978.
12. Hugel, H., "Effects of Self-Magnetic Forces on the Anode Mechanism of a High Current Discharge," *IEEE Trans. Plasma Sci.*, V. PS-8, No. 4, pp.437-442, Dec. 1980.
13. Gallimore, A.D., Kelly, A.J., and Jahn, R.G., "Anode Power Deposition in Quasisteady MPD Thrusters," *J. Propulsion & Pwr.*, V. 8, No. 6, pp. 1224-1231, Dec., 1992.

14. Dolson, R.C., and Biblarz, O., "Analysis of the Voltage Drop Arising from a Collision-dominated Sheath," *J. Appl. Phys.*, V. 47, No. 12, pp. 5280-5287, Dec., 1976.
15. Myers, R.M., "Energy Deposition in Low Power Coaxial Plasma Thrusters," Ph.D. Thesis, Princeton Univ., NJ, June, 1989.
16. Oberth, R.C., and Jahn, R.G., "Anode Phenomena in High-Current Accelerators," *AIAA Journal*, V. 10, No. 1, pp. 86-91, Jan., 1972.
17. Slezione, P.C., Auweter-Kurtz, M., and Schrade, H.O., "Numerical Codes for Cylindrical MPD Thrusters," IEPC 88-038, 20th Int'l. Elec. Prop. Conf., Garmisch-Partenkirchen, W. Germany, Oct., 1988.
18. Slezione, P.C., Auweter-Kurtz, M., and Schrade, H.O., "Numerical Evaluation of MPD Thrusters," *AIAA 90-2602*, July, 1990.
19. LaPointe, M.R., "Numerical Simulation of Self-Field MPD Thrusters," *AIAA-91-2341*, NASA-CR-187168, July, 1991.
20. Subramaniam, V.V., and Lawless, J.L., "Thermal Instabilities of the Anode in a Magnetoplasma Dynamic Thruster," *J. Propulsion & Pwr.*, V. 6, No. 2, pp. 221-224, Mar., 1990.
21. Biblarz, O., "Approximate Sheath Solutions for a Planar Plasma Anode," *IEEE Trans. Plasma Sci.*, V. 19, No. 6, pp. 1235-1243, Dec., 1991.
22. Biblarz, O., Dolson, R.G., and Shorb, R.C., "Anode Phenomena in a Collision-dominated Plasma," *J. Appl. Phys.*, V. 46, No. 8, pp. 3342-3346, Aug., 1975.
23. Rudolph, L.K. and Pawlik, E.V., "The MPD Thruster Development Program," AIAA Technical Paper 79-2050, in *Progress in Aeronautics and Astronautics*, Vol. 79, Amer. Inst of Aer. & Astro., 1981.
24. Cann, G.L., and Marlotte, G.L., "Hall Current Plasma Accelerator," *AIAA Journal*, V. 2, No. 7, pp. 1234-1241, Jul., 1964.
25. Brophy, J., *Stationary Plasma Thruster Evaluation in Russia, Summary Report*, Jet Propulsion Laboratory (JPL) Publication 92-4, March 15, 1992.
26. Mitchner, M., and Kruger, C.H., *Partially Ionized Gases*, pp. 128-134, Wiley, 1973.
27. Cobine, J.D., *Gaseous Conductors, Theory and Engineering Applications*, McGraw-Hill, 1941.

28. von Engel, A., *Ionized Gases*, Clarendon Press, 1965.
29. Chen, F.F., *Introduction to Plasma Physics and Controlled Fusion*, 2nd ed., p.10, Plenum, 1984.
30. Lin, S., Resler, E., and Kantrowitz, A., "Electrical Conductivity of Highly Ionized Argon Produced by Shock Waves," *J. Appl. Phys.*, V. 26, p. 95, Jan., 1955.
31. Campbell, A., *Plasma Physics and Magnetofluidmechanics*, p. 161, McGraw-Hill, 1963.
32. Nasser, E., *Fundamentals of Gaseous Ionization and Plasma Electronics*, p.412, Wiley, 1971.
33. Ecker, G., "Anode Spot Instability. I. The Homogeneous Short Gap Instability," *IEEE Trans. Plasma Sci.*, V. PS-2, No. 3, Sept., 1974.
34. Biblarz, O. and Riggs, J.F., "Anode Sheath Contributions in Plasma Thrusters," *AIAA* 93-2495, Jun., 1993.
35. Miller, H.C., "Vacuum Arc Anode Phenomena," *IEEE Trans. Plasma Sci.*, V. PS-11, No. 2, June, 1983.
36. Miller, H.C., "Discharge Modes at the Anode of a Vacuum Arc," *IEEE Trans. Plasma Sci.*, V. PS-11, No. 3, pp. 122-127, Sep., 1983.
37. Rich, J.A., Prescott, L.E., and Cobine, J.D., "Anode Phenomena in Metal-Vapor Arcs at High Currents," *J. Appl. Phys.*, V. 12, No. 12, pp.587-601, Feb., 1971.
38. Schuocker, D., "Improved Model for Anode Spot Formation in Vacuum Arcs," *IEEE Trans. Plasma Sci.*, V. PS-7, No. 4, pp. 209-216, Dec., 1979.
39. Gear, C.W., *Numerical Initial Value Problems in Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1971.
40. Burnet, H., Vincent, P., and Rocca Serra, J., "Ionization Mechanism in a Nitrogen Glow Discharge," *J. Appl. Phys.*, V. 54, No. 9, pp. 4951-4957, 1983.
41. Phelps, A.P. and Pitchford, L.C., "Anisotropic Scattering Electrons by N_2 and its Effect on Electron Transport," *Phys. Rev. A*, V. 31, pp. 2932-2949, 1985.
42. Myers, R.M., Kelly, A.J. and Jahn, R.G., "Energy Deposition in Low-Power Coaxial Thrusters," *J. Propulsion*, V. 7, No. 5, pp. 732-739, Sep./Oct., 1991.

43. Biblarz, O., "Thermionic Arc Initiation", 1992 *IEEE International Conference on Plasma Science, Tampa, FL*, June 1992.
44. Gallimore, A.D., "Anode Power Deposition in Coaxial MPD Thrusters," Ph.D. Thesis, Princeton Univ., NJ, Oct., 1992.
45. Biblarz, O. and Barto, J.L., "Fluid-Dynamic Effects, Including Turbulence, on a High Pressure Discharge". *Gas Flow and Chemical Lasers*, 6th Int. Symposium (S. Rosenwaks, Ed.), pp. 34-39, Springer-Verlag, Berlin, 1987.
46. Park, W. & Choi, D., "Numerical Analysis of MPD Arcs for Plasma Acceleration," *IEEE Trans. Plasma Sci.*, V. PS-15, No. 5, pp. 618-624, Oct., 1987.
47. Schoeck, P. A., Eckert, E.R.G., and Wutzke, S.A., "An Investigation of the Anode Losses in Argon Arcs and their Reduction by Transpiration Cooling," *ARL 62-341, DTIC AD-278570*, April, 1962.
48. Brady, J.E., *General Chemistry, Principles & Structure*, 5th ed., Wiley, 1990, p. 223.
49. Janes, G.S., "Magnetohydrodynamic Propulsion," in *Advanced Propulsion Techniques*, AGARD Proceedings, Aug., 1960, pp. 151-154, Pergamon, 1961.
50. Connolly, D.J., Sovie, R.J., Michels, C.J., and Burkhart, J.A., "Low Environmental Pressure MPD Arc Tests," *AIAA Journal*, V. 6, No. 7, pp. 1271-1276, July, 1968.
51. Subramaniam, V.V., "Fundamental Studies on Erosion in MPD Thrusters," *AFOSR 87-0360*, April, 1992.
52. Hurwics, H. and Rogan, J. E., "High Temperature Thermal Protection Systems", Section 19, , *Handbook of Heat Transfer* (Rohsenow, W.M., and Hartnett, J.P., Eds) McGraw-Hill, 1973.
53. Kuriki, K. and Suzuki, H., "Quasisteady MPD Arcjet with Anode Gas Injection," *AIAA 79-2058, 14th International Electric Propulsion Conference, Princeton, NJ*, Oct., 1979.
54. Sutton, G.W. and Sherman A., *Engineering Magnetohydrodynamics*, p. 148, McGraw-Hill, 1965.
55. Choueiri, E.Y., Kelly, A.J., and Jahn, R.G., "Mass Savings Domain of Plasma Propulsion for LEO to GEO Transfer," *J. Spacecraft and Rockets*, V. 30, No. 6, pp. 749-754, Nov./Dec., 1993.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
Cameron Station
Alexandria, Virginia 22304-6145
2. Library, Code 52 2
Naval Postgraduate School
Monterey, California 93943-5002
3. Chairman 1
Department of Aeronautics & Astronautics, Code AA
Naval Postgraduate School
Monterey, California 93943-5000
4. Professor Oscar Biblarz 3
Department of Aeronautics & Astronautics, Code AA/Bi
Naval Postgraduate School
Monterey, California 93943-5000
5. Professor Fred Schwirzke 1
Department of Physics, Code PH/Sw
Naval Postgraduate School
Monterey, California 93943-5000
6. Commander, Space & Naval Warfare Systems Command 1
Space Technology Directorate (SPAWAR-40)
2451 Crystal Drive
Arlington, Virginia 22245-5200
7. Dr. Roger Myers 1
Lewis Research Center
M.S. SPTD-1
Cleveland, Ohio 44135-3191
8. Dr. James S. Sovey 1
SPTD
NASA Lewis Research Center
21000 Brookpark Rd.
Cleveland, Ohio 44135

- | | |
|---|---|
| 9. Dr. Tom Pivrotto
Jet Propulsion Laboratory
4800 Oak Grove Dr.
Pasadena, California 91109
M.S.125-224 | 1 |
| 10. Dr. John Brophy
Jet Propulsion Laboratory
4800 Oak Grove Dr.
Pasadena, California 91109
M.S.125-224 | 1 |
| 11. Dr. Jay Polk
Jet Propulsion Laboratory
4800 Oak Grove Dr.
Pasadena, California 91109
M.S.125-224 | 1 |
| 12. Dr. Arnold J. Kelly
Dept. of Mechanical & Aerospace Engineering
Princeton University
Princeton, New Jersey 08544 | 1 |
| 13. Dr. Mitat A. Birkan
AFOSR NA
110 Duncan Avenue, Suite B115
Bolling AFB, D.C. 20352-0001 | 1 |
| 14. Dr. Ed Weiler
Code SZB
NASA Headquarters
Washington, D.C. 20546-0001 | 1 |
| 15. LCDR John Riggs
10215 Centinella Dr.
La Mesa, California 91941 | 5 |