CR128596

GEORGIA INSTITUTE OF TECHNOLOGY
School of Mechanical Engineering
Atlanta, Georgia

SIMPLIFIED ANALYSIS AND OPTIMIZATION OF SPACE BASE AND SPACE SHUTTLE HEAT REJECTION SYSTEMS

Contract No. NAS 9-10415

by

Wolfgang Wulff

Sponsored by the
Power Generation Branch
Manned Spacecraft Center
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Houston, Texas

April 1972

(NASA-CR-128596) SIMPLIFIED ANALYSIS AND OPTIMIZATION OF SPACE BASE AND SPACE SHUTTLE HEAT REJECTION SYSTEMS W. Wulff (Georgia Inst. of Tech.) Apr. 1972 158 p CSCL 22B G3/31

N73-10892

Unclas
45726

GEORGIA INSTITUTE OF TECHNOLOGY

SCHOOL OF MECHANICAL ENGINEERING

ATLANTA, GEORGIA

SIMPLIFIED ANALYSIS AND OPTIMIZATION

OF

SPACE BASE AND SPACE SHUTTLE HEAT

REJECTION SYSTEMS

Contract No. NAS 9-10415

by

Wolfgang Wulff

Sponsored by the

Power Generation Branch, Manned Space Craft Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Houston, Texas

April 1972

I

SIMPLIFIED ANALYSIS AND OPTIMIZATION

OF

SPACE BASE AND SPACE SHUTTLE HEAT

REJECTION SYSTEMS

by

Wolfgang Wulff

SCHOOL OF MECHANICAL ENGINEERING

GEORGIA INSTITUTE OF TECHNOLOGY

ATLANTA, GEORGIA 30332

April 1972

Sponsored by

Power Generation Branch, Manned Space Craft Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Houston, Texas

Wolfgang Wulff

Wolfgang Wulff, Ph.D.
Associate Professor

Stothe P. Kezios

Stothe P. Kezios, Ph.D.
Director, School of Mechanical Engineering

FOREWORD

This report covers part of the work performed during the second year phase of a two year research project, at the School of Mechanical Engineering at the Georgia Institute of Technology in Atlanta, Georgia, for the NASA Manned Spacecraft Center, Houston, Texas. The contract designation is NAS 9-10415 and the project title is "Study of Design Parameters of Space Base and Shuttle Heat Rejection Systems." The project was monitored by Dr. W.E. Simon of the Power Generation Branch of NASA MSC, Houston, Texas, and was performed by Dr. W.Z. Black and Dr. W. Wulff as Co-Investigators. The project resulted in one Annual Report [2]* and two Final Reports, the first one of which covers a detailed and rigorous space radiator simulation analysis [1] and includes a computer program users manual. The second Final Report is this report.

Although the study of system parameters and the system optimization were originally conceived to be performed on the basis of the rigorous radiator system simulation [1], the growing complexity of this simulation soon gave rise to the need for a simplified analysis. The simplified system simulation was later expanded into systematic optimization procedures. Both the simplified simulation and the optimization procedures serve to supplement and support the originally intended system parameter study [1].

The work presented here was supported by the contributions of computer coding by Mr. Richard J. Huntley and Mr. Wallace W. Carr, both Graduate Research Assistants and M.S. Candidates.

---

*Numbers in brackets refer to the Bibliography at the end of this report.

SUMMARY

A simplified radiator system analysis was performed to predict steady-state radiator system performance. The system performance was found to be describable in terms of five non-dimensional system parameters. The governing differential equations are integrated numerically to yield the enthalpy rejection for the coolant fluid.

The simplified analysis was extended to produce firstly the derivatives of the coolant exit temperature with respect to the governing system parameters and secondly a procedure to find the optimum set of system parameters which yields the lowest possible coolant exit temperature for either a given projected area or a given total mass. The process can be inverted to yield either the minimum area or the minimum mass, together with the optimum geometry, for a specified heat rejection rate.

The major accomplishments of the simplified radiator system analysis are:

(1) the reduction of the number of necessary systems parameters from twelve or more to six,

(2) the graphical representation of system performance in terms of non-dimensional groups, suitable to aid in the design of radiative heat rejection systems, and

(3) an efficient computer code suitable for preliminary performance prediciton.

The accomplishments of the systematic optimization analysis are two computer codes which perform iterative optimizations processes leading to the maximum heat rejection for either a given projected fin-plus-tube area or a given total system mass.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

NOMENCLATURE

A^*	Normalized projected radiator area, Eq. 38
$\underset{\approx}{A}$	Coefficient matrix in Eq. 49
A_{ij}	Elements of coefficient matrix $\underset{\approx}{A}$
a_i	Coefficients of power polynomial
$B_{ij}=\partial^2\Theta_b/\partial x_i \partial x_j$	Second-order derivatives of Θ_b
c	Constant in Nusselt number relation, Eq. 35
c_p	Coolant fluid specific heat (Btu/(lbm R))
d	Tube diameter (ft)
$d^*=d/L_o$	Normalized tube diameter
F_{fs}	Fin to sink view factor, Eq. 15
f	Blockage coefficient, defined by Eq. 16
H	Fin height, from base to tip (ft)
$H^*=H/L_o$	Normalized fin height
$\bar{h}_c$	Average convective film coefficient for convection from fluid to tube (Btu/(hr ft^2 R))
i	Number of radiatively active sides
k	Thermal conductivity (Btu/(hr ft R)) of structural material
k_f	Thermal conductivity (Btu/(hr ft R)) of coolant fluid
L	Tube length (ft)
L_o	Reference length (ft), defined by Eq. 37
$L^*=L/L_o$	Normalized tube length
M	Parameter, defined by Eq. 12 (viewfactor augmentation)
M	Structural mass, per fin-tube element (lbm)
$M_o=\rho L_o^3$	Reference mass (lbm)

M_f	Coolant fluid mass (lbm)
$M^*=M/(M_o\phi_1)$	Normalized structural mass
$M_f^*=M_f/(M_o\phi_1)$	Normalized coolant mass
$M_{tot}^*=M_f^* + M^*$	Normalized total mass
m	Exponent on N_{Re} in Nusselt number relation, Eq. 35
$\dot{m}$	Coolant mass flow rate (lbm/hr)
N	Tube to sink view factor, Eq. 17
$N_c = \bar{N}_c\Theta_b^3$	Conduction parameter, Eq. 14
$\bar{N}_c, N_c^*$	Reference conduction parameter, defined by Eqs. 20 and 58*
N_{Gr}, N_{Gr}^*	Graetz number, Eqs. 23 and 55
N_{Nu}, N_{Nu}^*	Nusselt number, Eqs. 8, 9, 10, 35 and 57*
N_{Pr}	Prandtl number, Eq. 7
N_{Re}, N_{Re}^*	Reynolds number, Eqs. 6 and 56*
n	Number of coolant channels
n	Exponent on N_{Pr} in Nusselt number relation, Eq. 35
p	Exponent on d/L-ratio in Nusselt number relation, Eq. 35
q''	Incident radiant heat flux (Btu/(hr ft^2))
r	Number of constraints
T	Absolute temperature (R)
t	Fin panel thickness (ft)
$t^* = t/L_o$	Normalized fin panel thickness
U	System parameter, defined by Eq. 22
V	System parameter, ratio of convective to radiative resistances, Eq. 21
$\underset{\sim}{X}$	Vector whose components are the system parameters x_i

*Superscripted stars on symbol N indicate non-dimensional groups evaluated by replacing all dimensions L, H, d and t by L_o.

x_i	General system parameters, $x_1 = U$, $x_2 = V$, $x_3 = \bar{N}_c$, $x_4 = \lambda$
$\Delta \underset{\sim}{X}$	Computed changes of $\underset{\sim}{X}$, in iteration process
Y	Generalized distance from optimum, defined by Eqs. 49 and 50
y_i	Components of $\underset{\sim}{Y}$, defined by Eqs. 50
z	Distance along the tube (ft)

GREEK SYMBOLS

α_s	Solar absorptance
$\underset{\sim}{\delta}$	Incremental component vector defining a neighborhood about potential optimum, Eq. 47
ε	Surface emittance
$\zeta = Z/L_o$	Normalized axial distance
η	Effectiveness of unobstructed fin
$\bar{\eta}$	Combined fin and tube effectiveness
$\theta = T/T_o$	Normalized temperature
$\lambda = 4H/d$	
σ	Stefan Beltzmann constant $\sigma = 0.1714 \times 10^{-8}$ Btu/(hr ft^2 R^4)
μ	Dynamic viscosity (lbm/(sec ft))
ρ	Structural material density (lbm/ft^3)
ρ_f	Fluid density (lbm/ft^3)
ϕ_1	Parameter, defined by Eq. 42
ϕ_2	Parameter, defined by Eq. 44
ϕ_3	Parameter, defined by Eq. 63
φ_i	Constraints, Eqs. 39, 40, 41 and 46

SUBSCRIPTS

b	Fin base
c	Conduction

e	Channel exit
f	Fluid, bulk property
i,j	Component subscripts
k	Iteration step counting index
m	at optimum
s	sink
w	Fluid, at tube wall
o	Channel inlet

I. INTRODUCTION

A large-scale, complete and rigorous computer simulation of space radiator systems was developed under the same contract as the work presented here [1].* The computer code consists of over fifty program units and is capable of simulating transient as well as steady-state radiator system performances under prescribed time-dependent operational and environmental conditions. The program accommodates both gaseous and liquid coolant fluids with any consistently prescribed set of thermodynamic and transport properties. In principle, this large-scale computer program could serve to not only predict radiator system performance but also to optimize certain design paramters via enumeration of performance characteristics, associated with selected parameter combinations

Any large computer simulation must be tested during its development; the greater the number of independent verification modes the greater will be the confidence in the program performance. In addition to the tests described in Reference [1], a simplified radiator system analysis was developed to verify the large-scale computer program performance. This simplified analysis served later (1) to extend the rigorous analysis to a wider class of radiator geometries and operating conditions [1] than would have been possible otherwise and (2) to develop systematic optimization procedures in support of system parameter studies on the basis of the rigorous computer simulation.

The essence of the simplified analysis lies in the recognition of the dominant performance characteristics and in the parameter reduction through normalization of the governing differential equations. The optimization is based on standard requirements of extrema subject to suitable constraints. These requirements are imposed on the governing differential equations prior to their numerical integration.

Chapter II below is a discussion of the simplified analysis and is followed by the presentation of the optimization analysis in Chapter III.

---

*Numbers in brackets refer to the Bibliography.

II. SIMPLIFIED RADIATOR SYSTEM ANALYSIS

A. Objective and Background

The purpose of this analysis is to provide an efficient process of least complexity which serves to describe the steady-state performance of a space radiator system. The radiator system is described in detail in the following chapter.

The analysis is intended to accommodate the essential features of the radiator system performance but to require less computational effort than the rigorous system simulation described in Reference [1]. Finally, the simplified analysis should serve as the basis for systematic optimization of the radiator system.

A first simplified analysis was developed during the first year of the contract period [2], also in support of the rigorous computer analysis, and for the purpose of treating approximately some radiator system geometries which deviate from the basic geometry underlying the rigorous system analysis, such as cylindrical shell radiator structures, asymmetrically loaded coolant channels etc. [1]. This analysis was based on two global energy balances, one for the coolant fluid and one for the radiator panel, and included the two thermal resistances associated with the energy transfer from the fluid to the tube and from the radiator into space. The <u>mean</u> tube wall temperature (and with it, the fin base temperature) was obtained from a single, transcendental equation (Eq. D.11, [2]) and the cooling capacity was found explicitly in terms of the mean tube wall temperature (Eq. 1.1 or 3, [2]). Agreement between this simplified analysis and the full-scale computer simulation was found to be approximately 4%.

The search for an optimum radiator geometry on the basis of the first simplified analysis lead to the conclusion that for optimum radiator geometries neighboring coolant channels should be expected to be close enough to each other so as to obstruct each other's, and the fin panel's, view of space. Moreover, the rigorous computer simulation indicated that the tube wall temperature changes significantly in the direction of the flow even though conduction parallel to the flow direction, both in the fluid, in the tube wall and in the fin, is negligible. These effects are not accounted for in the first simplified analysis.

The new simplified analysis presented here, for the prediction of steady-state radiator heat rejection performance, takes into account

(i) temperature variations within the fluid, tube and fin, along the direction of the flow,

(ii) temperature variation within the fin, normal to the direction of the flow,

(iii) direct radiative interaction between fin panel and space, tube and space, fin and tubes, tube and neighboring tubes (by approximation),

but does not take into account

(i) thermal conduction in the direction of the coolant flow, both in the fin-tube structure and the fluid

(ii) temperature variation of thermal properties,

(iii) end effects at the coolant fluid manifold,

(iv) cross-sectional changes in the fin panel,

(v) secondary reflections of radiant heat.

The analysis reduces to an initial value problem consisting of two ordinary, first-order, coupled differential equations, with one of its initial conditions given through a transcendental equation. The new formulation is suitable for optimization.

B. Description of Radiator System

The radiator system considered here consists of a given number of parallel coolant channels, equally spaced in one plane, and connected by rectangular fin panels which have constant thickness and are symmetric with respect to the plane through the tube areas.

The coolant channels are taken to be thin-walled tubes of circular cross-section, with diameter d and length L. Substitution of the hydraulic diameter for the diameter permits readily the inclusion of channels with non-circular cross-sections, except for possible effects from radiative interaction between tube, fin and environment.

Let the distance between tube centers be designated by $d + 2H$ and the fin panel thickness by t. Then the radiator can be considered to consist of the given number n of identical fin elements as shown in Figure 1.

The coolant fluid enters the channel at $z = 0$ with time-invariant temperature $T_f = T_o$. Both the fluid temperature $T_f(z)$ and the fin base temperature $T_b(z)$ decrease along the tube provided the equivalent sink temperature T_s is less than T_o. The coolant fluid emerges from the radiator at $z = L$ with the exit temperature T_e.

The objective of the analysis is to predict the rate of heat rejection $\dot{m}\, c_p\, (T_o - T_e)$ per tube for a given mass flow rate $\dot{m}$ per tube and given fluid properties, namely density ρ, specific heat c_p, thermal conductivity k_f, dynamic viscosity μ, and given properties of the fin, that is, surface emittance ε, solar absorptance α_s, and thermal conductivity k.

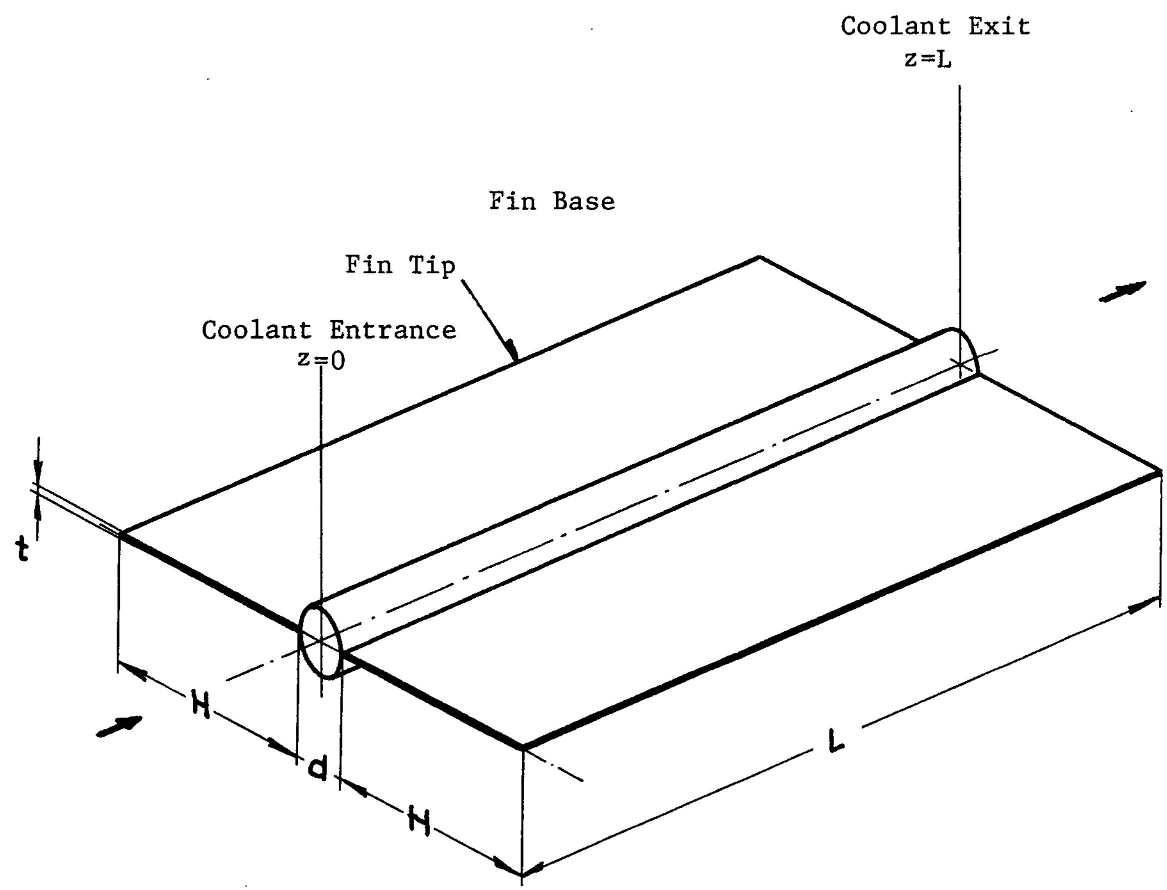

Figure 1. Geometry of Fin System Element

C. Governing Equations

1. Conservation of Energy

Setting the local change of the fluid enthalpy flux first equal to the local heat exchange per unit tube length between fluid and tube wall, and then equal to the radiative heat rejection per unit tube length one obtains, for i effective sides of the radiator:

$$- \dot{m}\, c_p \frac{dT_f}{dz} = \pi d \bar{h}_c (T_f - T_b) \qquad (1)$$

$$\pi d \bar{h}_c (T_f - T_b) = 2iH\varepsilon\sigma\bar{\eta}(T_b^4 - T_s^4) \qquad (2)$$

where $\bar{h}_c$ and σ represent, respectively, the convective film coefficient and the Stefan-Boltzmann constant. The symbol $\bar{\eta}$ represents the overall tube-and-fin effectiveness and is evaluated in Section 3. The convective film coefficient is computed from the Nusselt number N_{Nu}

$$\bar{h}_c = N_{Nu} \frac{k_f}{d} \qquad (3)$$

the evaluation of which is deferred to Section 2. The equivalent sink temperature T_s is computed from the known incident normal radiant heat flux q'' by

$$T_s = \sqrt[4]{\frac{\alpha_s}{\varepsilon} \frac{q''}{\sigma}} \qquad (4)$$

The first-order differential equation, Eq. 1, is subject to the initial condition at z = 0

$$T_f(0) = T_o \qquad (5)$$

Equations 1 and 2 determine the two dependent variables $T_f(z)$ and $T_b(z)$. Integration of Eq. 1 from $z = 0$ to $z = L$ gives the unknown fluid exit temperature T_e.

2. Convective Film Coefficient

The Nusselt number N_{Nu} in Eq. 3 depends on the Reynolds number N_{Re}, the Prandtl number N_{Pr} and the d/L ratio.

$$N_{Re} = \frac{4\dot{m}}{\pi d \mu} \quad (6)$$

$$N_{Pr} = \frac{\mu c_p}{k_f} \quad (7)$$

For laminar flow, $N_{Re} \leq 2300$, of non-metallic fluids, $N_{Pr} \geq 0.1$, Hausen [3] established the relation

$$N_{Nu} = \left[3.65 + \frac{0.0668\, N_{Re} N_{Pr} \frac{d}{L}}{1 + 0.045 (N_{Re} N_{Pr} \frac{d}{L})^{2/3}} \right] \left(\frac{\mu_f}{\mu_w} \right)^{0.14} \quad (8)$$

The nomogram in Figure 2, taken from the VDI Waermeatlas,* facilitates the estimate of the Nusselt number in accordance with Eq. 8.

For turbulent flow, $N_{Re} > 2300$, of non-metallic fluids, $N_{Pr} \geq 0.1$, similarity considerations lead to

$$N_{Nu} = 0.116 \left[1 + \left(\frac{d}{L} \right)^{2/3} \right] (N_{Re}^{2/3} - 125) N_{Pr}^{1/3} \left(\frac{\mu_f}{\mu_w} \right)^{0.14} \quad (9)$$

which is also presented in a nomogram in Figure 3.

Liquid metal convective heat transfer in tubes, $N_{Pr} < 0.1$, may be represented by the result of the work by Seban and Shimazaki [4] which is

*VDI-Verlag GMBH, Düsseldorf, West Germany valid for $(N_{Re}\ N_{Pr}) > 100$ and $L/d > 60$

$$N_{Nu} = 5.0 + 0.025\ (N_{Re} N_{Pr})^{0.8} \tag{10}$$

Other Nusselt number relationships may be more suitable in a particular case. Their selection does not affect the ensuing analysis because the Nusselt number calculation is part of the input data preparation, prior to the numerical evaluation of the solution.

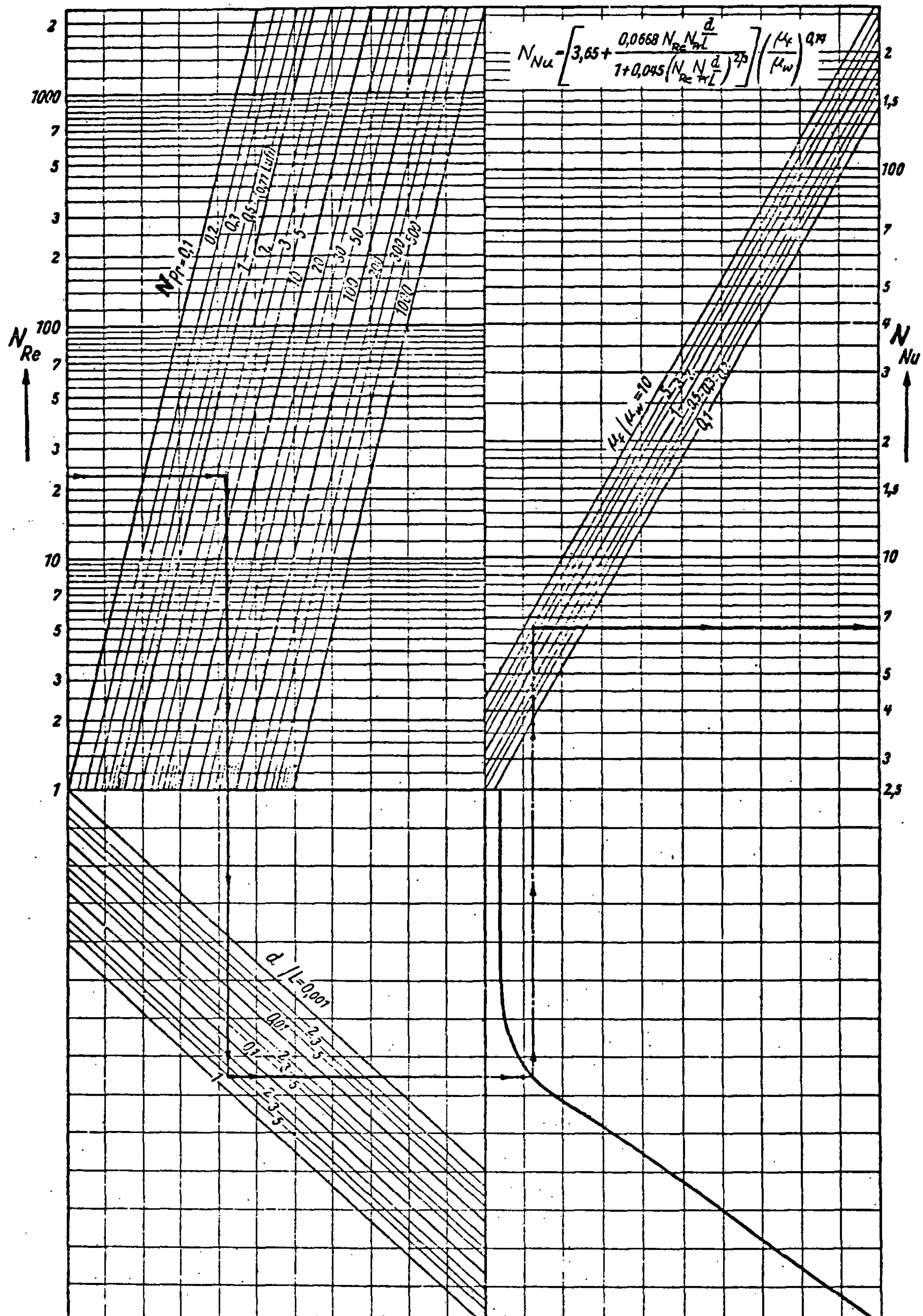

Figure 2. Nussett Number Nomogram for Laminar Flow of Non-Metallic Coolant Fluids, Eq. 8. (VDI Waermeatlas)

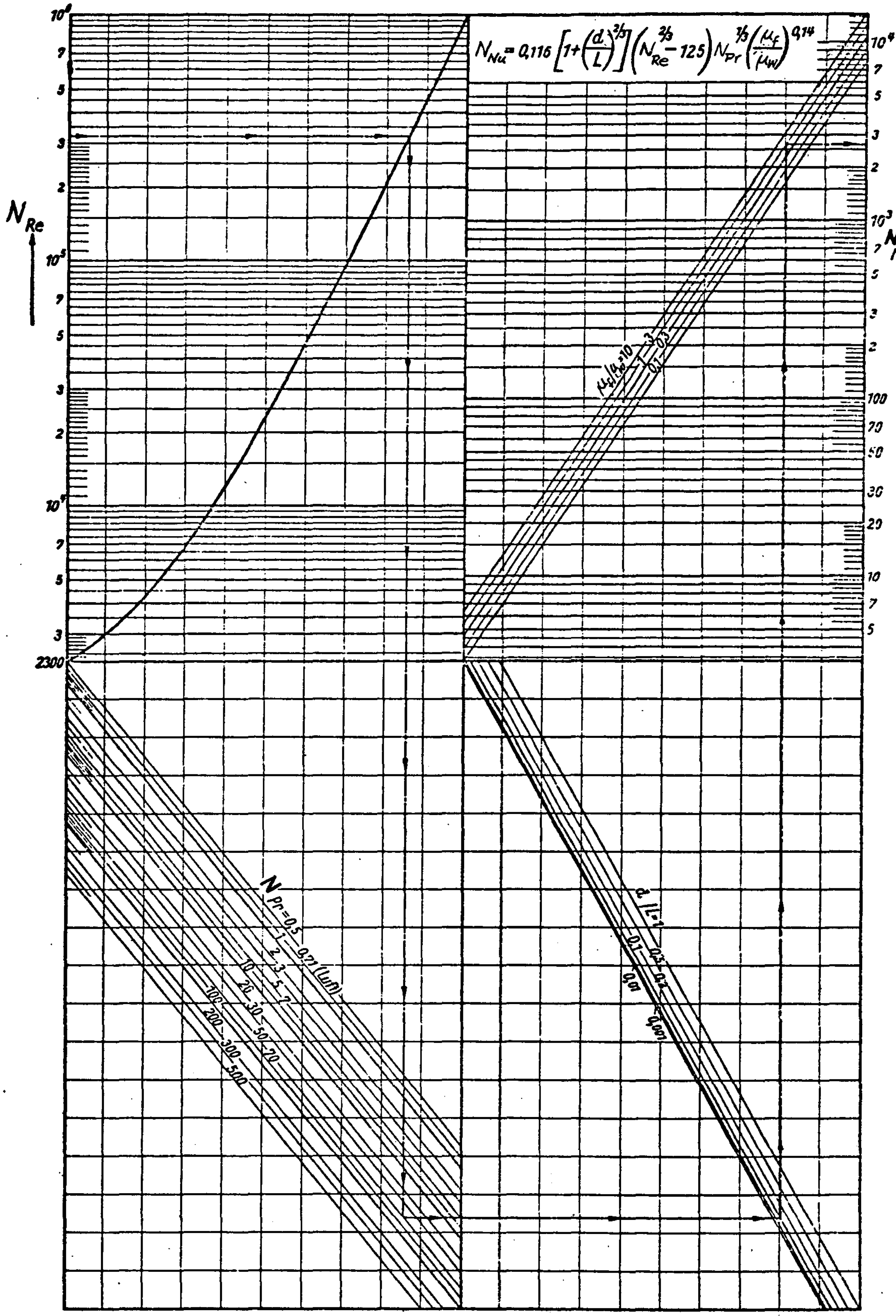

Figure 3. Nussett Number Nomogram for Turbulent Flow of Non-Metallic Coolant Fluids, Eq. 9. (VDI Waermeatlas)

3. Total Fin-Tube Effectiveness

The symbol $\bar{\eta}$ in Eq. 2 denotes the overall radiative fin-and-tube effectiveness and accounts not only for the non-uniform temperature distribution within the fin and normal to the flow direction, but also for the effects on the radiative heat transfer from the tube. These effects are due to the additional radiating tube area and the partial blockage of emission from the fin and the tubes by the tubes of the neighboring fin elements. Adding the contributions from the tube to that of the fin, one obtains

$$\bar{\eta}(z) = M(z)\cdot\eta(z) + N \tag{11}$$

Here, $\eta(z)$ is the radiative fin effectiveness of the unobstructed fin, Eqs. 32, [5,6,7] and varies with the fin base temperature $T_b(z)$. The factor M is the product of the shape factor of the fin with respect to the sink, F_{fs}, and the corrective term f on η for the temperature profile distortion resulting from the tube-fin interaction [8]:

$$M = F_{fs}(\lambda)\cdot f(\lambda,N_c) \tag{12}$$

where

$$\lambda = \frac{4H}{d} \tag{13}$$

and

$$N_c = \frac{2\varepsilon\sigma H^2}{kt} \quad [T_b(z)]^3 \tag{14}$$

which is the well-known conduction parameter. Hottel's crossed-string method yields

$$F_{fs}(\lambda) = \left(\sqrt{\lambda(\lambda + 2)} + \arcsin \frac{1}{1 + \lambda} - \pi/2 \right) / \lambda \quad (15)$$

and the curve fit of data in Reference [8] resulted in

$$f(\lambda, N_c) = 1 - (N_c/\lambda)(0.1460\ N_c - 0.02866)\ . \quad (16)$$

Finally, the symbol N in Eq. 11 represents the radiative heat rejection contributed by the tubes and is the shape factor of the tube with respect to the sink, multiplied by the tube-to-fin area ratio

$$N(\lambda) = \pi \left[1/2 + \left[\lambda + 2 - \sqrt{\lambda(\lambda + 2)} - \arcsin \frac{1}{\lambda + 1} \right] / \pi \right] / \lambda \quad (17)$$

This completes the description of the overall fin-and-tube effectiveness. In summary it should be emphasized that axial conduction, radial tube wall temperature variation, kinetic and potential coolant energies, and end effects are ignored, that radiative interaction between tube and fin and between tubes is only approximate and that the representation of Eq. 16 is limited to $T_s < 0.8\ T_b$ (but could be extended, in principle).

D. Scaling Parameters

Effective analysis, numerical integration, graphical representation of numerical results and, ultimately, the system optimization makes the reduction of parameters mandatory. Introduce the non-dimensional

axial distance $$\zeta = \frac{z}{L} \tag{18}$$

temperature $$\theta = T/T_o \tag{19}$$

reference conduction parameter $$\bar{N}_c = \frac{2\varepsilon\sigma T_o^3 H^2}{kt} \tag{20}$$

ratio of convective to radiative resistance $$V = \frac{i}{\pi}\frac{t}{H}\frac{k}{k_f}\frac{\bar{N}_c}{N_{Nu}} \tag{21}$$

convection number $$U = \pi N_{Nu}/N_{Gz} \tag{22}$$

where the Graetz number N_{Gz} is defined by

$$N_{Gz} = (c_p\,\dot{m})/(k_f L) = \frac{\pi}{4}\frac{d}{L} N_{Re}\, N_{Pr}\ . \tag{23}$$

With these parameters and the λ defined by Eq. 13 one may recast the problem as previously established by Eqs. 1,2 and 5 as given by Eqs. 24 through 26:

$$\frac{d\theta_f}{d\zeta} = -\, U(\theta_f - \theta_b) \tag{24}$$

$$\theta_f - \theta_b = V\bar{\eta}(\theta_b^4 - \theta_s^4) \tag{25}$$

subject to the initial condition at $\zeta = 0$

$$\theta_f(0) = 1 \ . \tag{26}$$

Integration of Eq. 24 subject to Eqs. 25 and 26 yields the coolant fluid exit temperature θ_e sought

$$\theta_e = \theta_f(1) \tag{27}$$

and, consequently, this coolant fluid exit temperature is a function of five parameters

$$\theta_e = \theta_e(U, V, \bar{N}_c, \lambda; \theta_s) \ . \tag{28}$$

The first four parameters correspond to the geometrical dimensions L, H, t and d of the radiator system and θ_s represents the environment.

E. Solution

Equation 25 is transcendental because of $\bar{\eta}(\theta_b)$ and would require iterative solutions at every step of the numerical integration of Eq. 24. It is more economical to derive a second differential equation for θ_b by differentiating Eq. 25

$$\frac{d\theta_b}{d\zeta} = \frac{-U}{\dfrac{1}{\theta_f - \theta_b} + \dfrac{1}{\bar{\eta}}\dfrac{d\bar{\eta}}{d\theta_b} + \dfrac{4\theta_b^{\,3}}{\theta_b^{\,4} - \theta_s^{\,4}}} \tag{29}$$

and to obtain the initial condition for Eq. 29 through a single iterative process (Newton Raphson method) from

$$V\bar{\eta}(\theta_b^{\,4} - \theta_s^{\,4}) + \theta_b = 1 \quad \text{at} \quad \zeta = 0 \tag{30}$$

The derivative $d\bar{\eta}/d\theta_b$ is obtained from Eqs. 11 and 16

$$\begin{aligned} \frac{d\bar{\eta}}{d\theta_b} &= \frac{d\bar{\eta}}{dN_c}\frac{dN_c}{d\theta_b} = 3\,\bar{N}_c\theta_b^{\,2}\,\frac{\partial}{\partial N_c}\,(M\eta) \\ &= 3\bar{N}_c\theta_b^{\,2}\left\{ \eta F_{fs}\left[0.02866 - 0.2920\,\bar{N}_c\theta_b^{\,3}\right]/\lambda \right. \\ &\qquad \left. + M\frac{d\eta}{dN_c} \right\} \end{aligned} \tag{31}$$

which can be evaluated once the fin effectiveness $\eta(N_c)$, that is the ratio of the actual power loss from an unobstructed (tubless) fin panel to the power loss from an ideal, unobstructed fin of infinite thermal conductivity, is represented by a power polynomial:

$$\eta(N_c) = \sum_{i=0}^{6} a_i (N_c)^i \qquad (32a)$$

where

$a_0 = 1.000\ 000$
$a_1 = -1.163\ 143$
$a_2 = 1.478\ 836$
$a_3 = -1.267\ 550$
$a_4 = 0.632\ 522\ 3$
$a_5 = -0.162\ 706\ 7$
$a_6 = 0.016\ 542\ 23$

Equation 32 is valid for $0 \leq N_c \leq 2.5$. For greater values, $N_c > 2.5$

$$\eta(N_c) = 0.686\ 609\ 5\ e^{-0.229\ 771\ 8} \qquad (32b)$$

Equations 24 and 29 have been integrated by a fourth-order Runge-Kutta procedure [2], subject to initial conditions given by Eqs. 26 and 30. The result of the integration yields primarily Θ_e in the form of Eq. 28.

F. Results

A typical computer print-out of the results is shown in Figure 4. First are presented the five parameters $\{\theta_s; U, V, \bar{N}_c, \lambda\}$ of Eq. 28, as read-in. Following that is a table in which are listed, as functions of axial distance measured from the coolant fluid inlet, the fluid temperature θ_f = THETAF, the fin base temperature θ_b = THETAB, next the local conduction parameter N_c = NC, the local total fin-plus-tube effectiveness $\bar{\eta}$ = ETABAR, the local fin effectiveness η = ETA, the nondimensional axial position ζ = ZETA and, finally, the number of integration steps required to reach the particular axial position. The number of integration steps depends on the chosen error limits and the rates of variable changes.

The time rate of heat rejection per fin element (tube) is to be computed from

$$\dot{m}\, c_p\, (T_o - T_e) = \dot{m}\, c_p\, T_o\, (1 - \theta_e) \tag{32}$$

Parameter studies are presented in Figures 5 through 9. The non-dimensional fluid exit temperature is plotted versus non-dimensional fin panel area with the temperature θ_s (Fig. 5), the non-dimensional tube spacing λ (Figs. 6 and 7) and the reference conduction parameter $\bar{N}_c$ (Figs. 8 and 9) as parameters.

THETA-S = .800000
U = .100000+01
V = .200000+01
ENCBAR = .200000+00
LAMBDA = .250000+02

THETAF	THETAB	NC	ETABAR	ETA	ZETA	STEP NO
1.000000	.838159	.117763	.964240	.881541	.00	1
.98446[illegible]	.835315	.116568	.965251	.882615	.10	2
.970137	.832677	.115468	.966185	.883569	.20	4
.956937	.830231	.114453	.967048	.884452	.30	5
.944763	.827964	.113518	.967846	.885267	.40	6
.93355[illegible]	.825862	.112656	.968583	.886020	.50	7
.923266	.823916	.111861	.969263	.886716	.60	8
.913669	.822113	.111128	.969892	.887359	.70	9
.904875	.820443	.110453	.970473	.887953	.80	10
.896764	.818898	.109830	.971009	.888501	.90	11
.889284	.817467	.109255	.971505	.889008	1.00	12

FIGURE 4. TYPICAL COMPUTER PRINT-OUT FOR RADIATOR SYSTEM PERFORMANCE

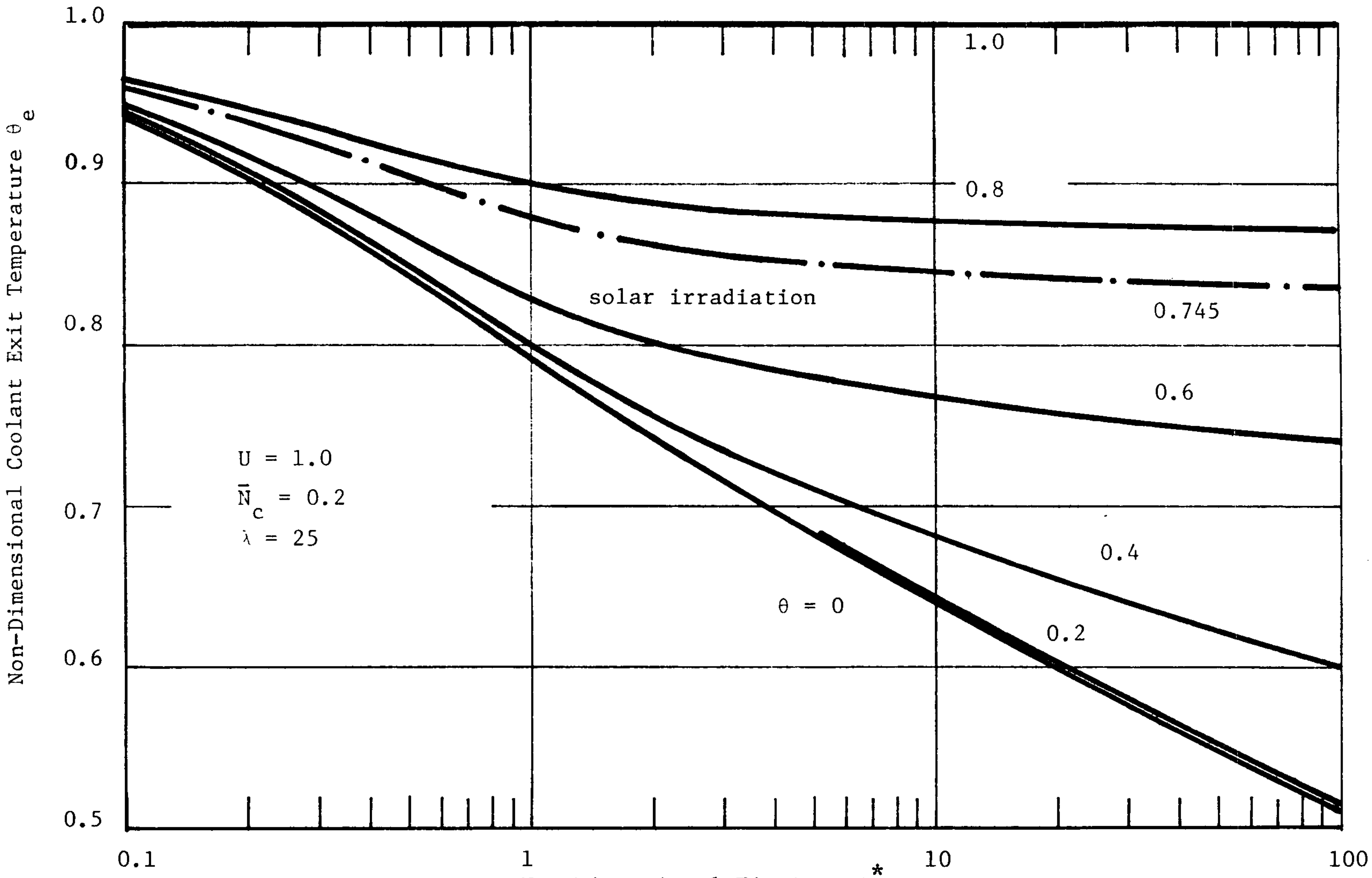

Figure 5. Non-Dimensional Coolant Exit Temperature θ vs. Non-Dimensional Fin Panel Area A^* with Sink Temperature θ_s As Parameter.

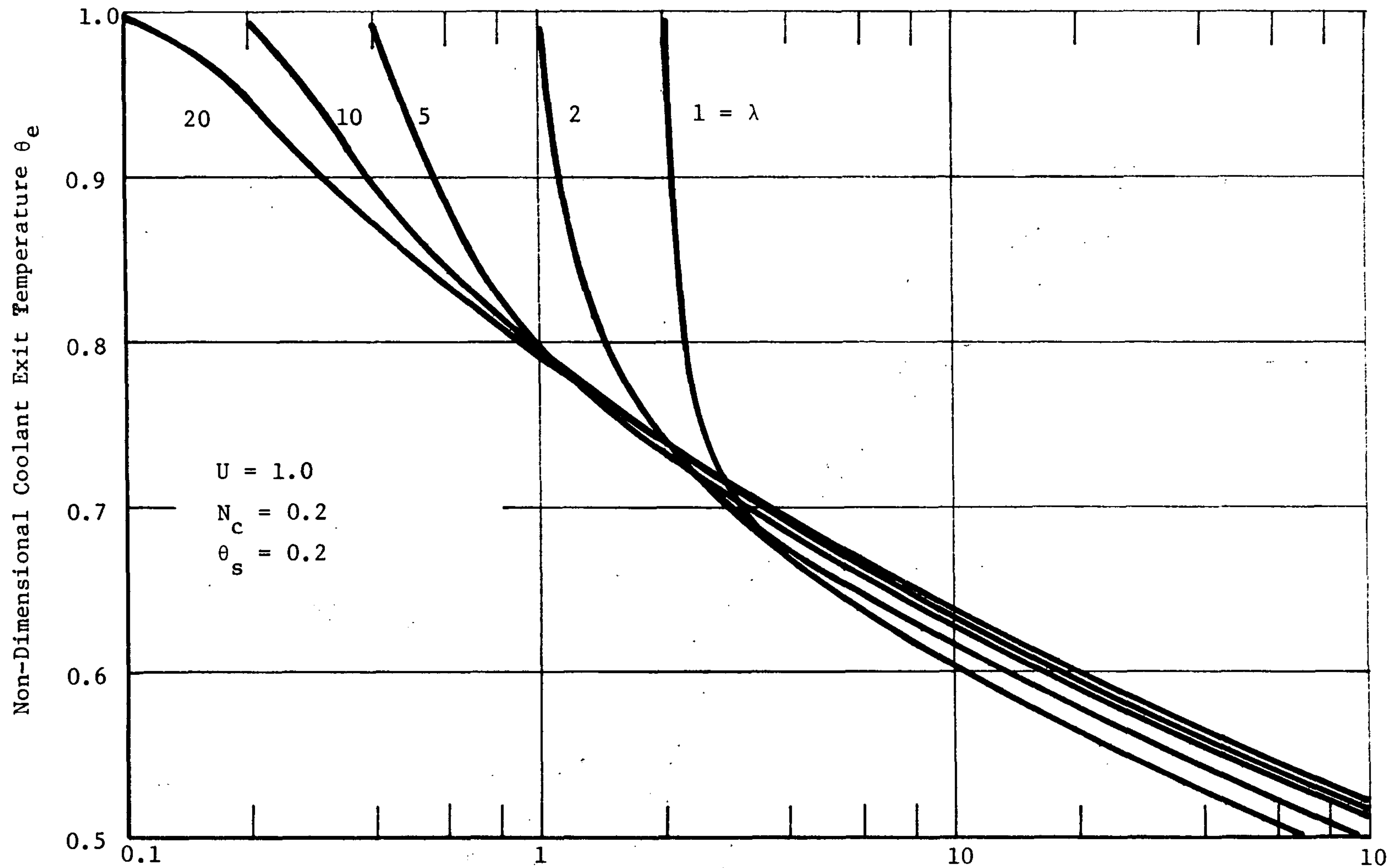

Figure 6. Non-Dimensional Coolant Exit Temperature vs. Non-Dimensional Fin Area For Constant U and Varying λ

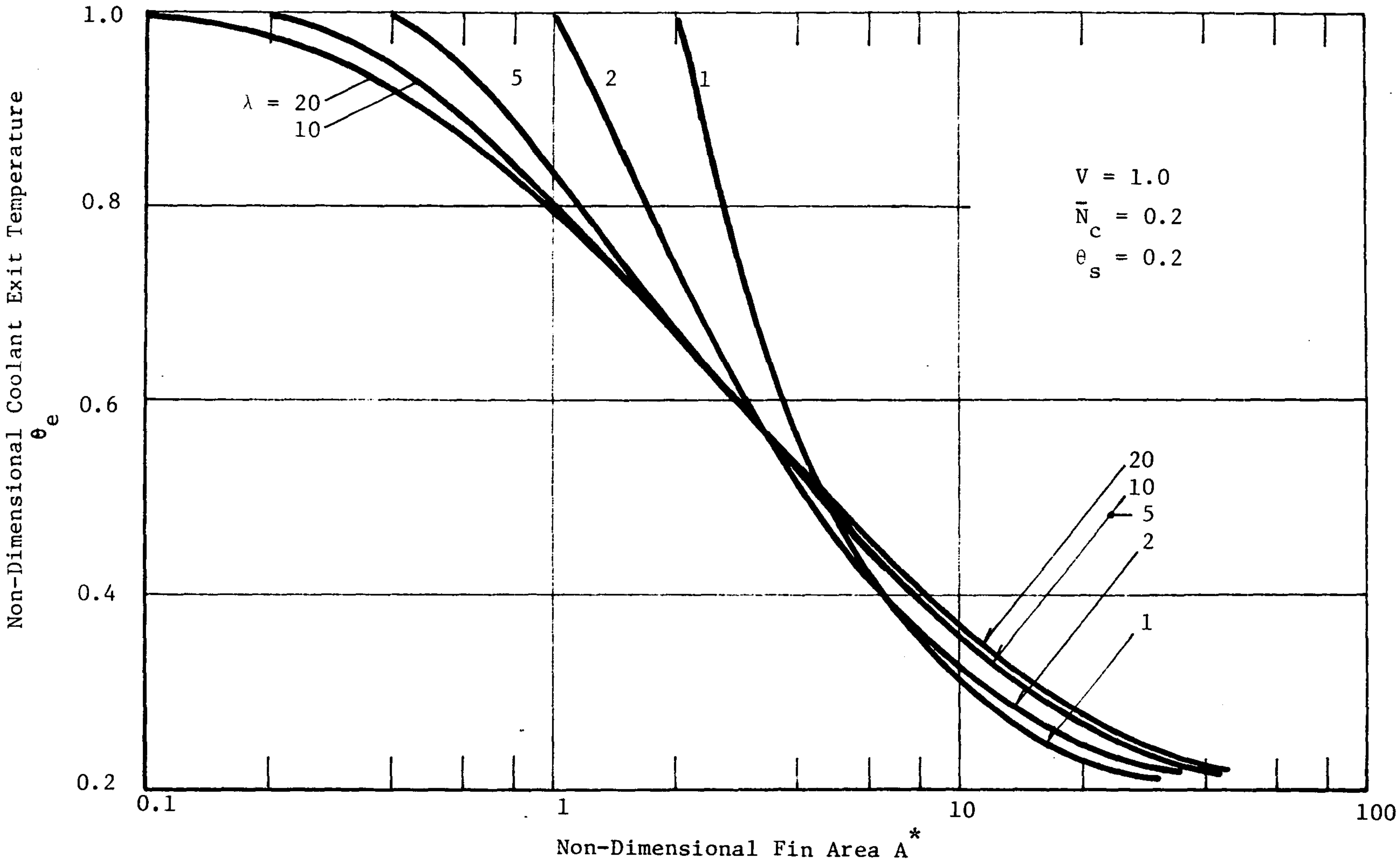

Figure 7. Non-Dimensional Coolant Exit Temperature vs. Non-Dimenstional Fin Area For Constant V and Varying λ

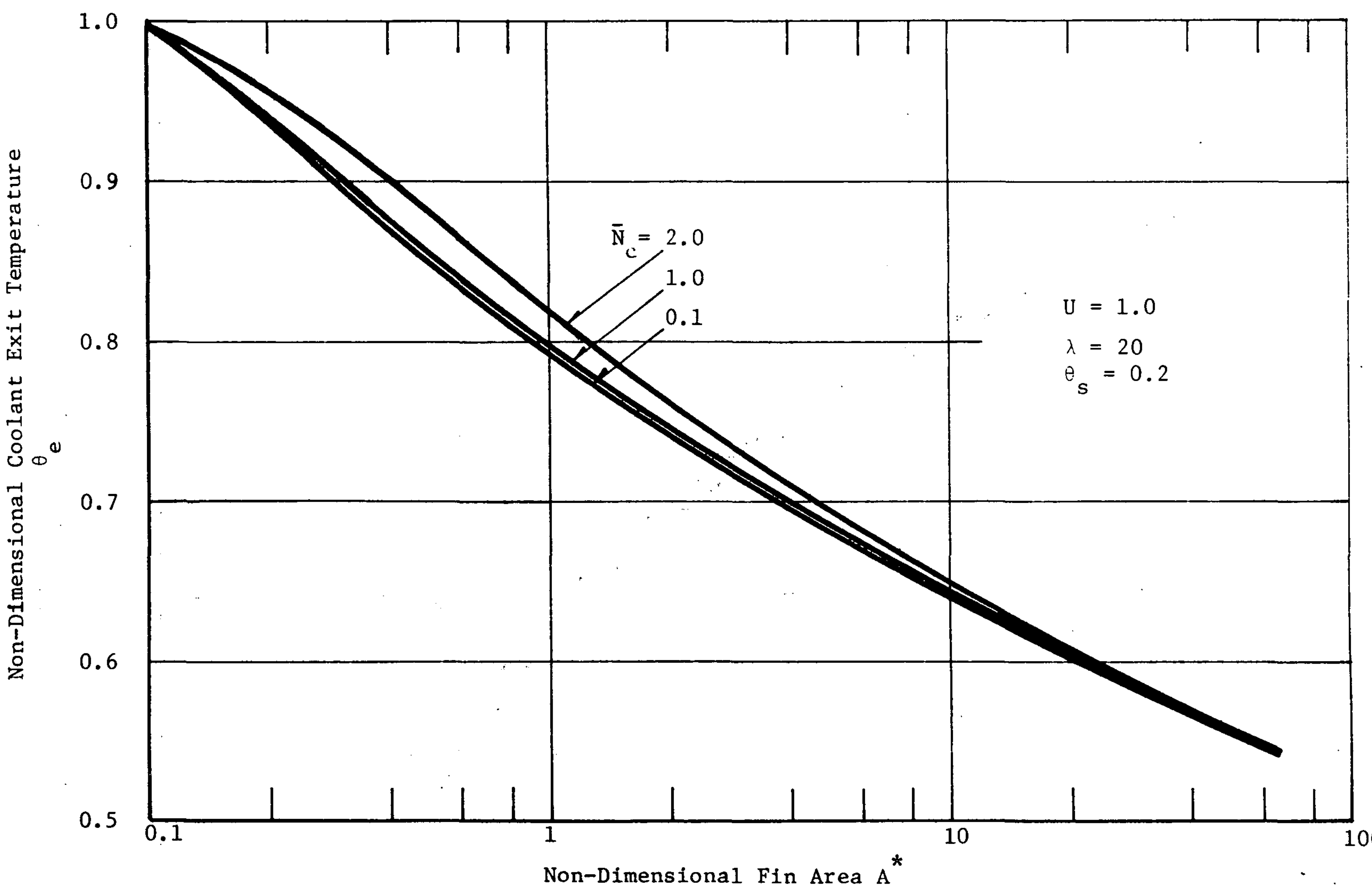

Figure 8. Non-Dimensional Coolant Exit Temperature VS. Non-Dimensional Fin Area For Varying Conduction Parameter and Long Fins.

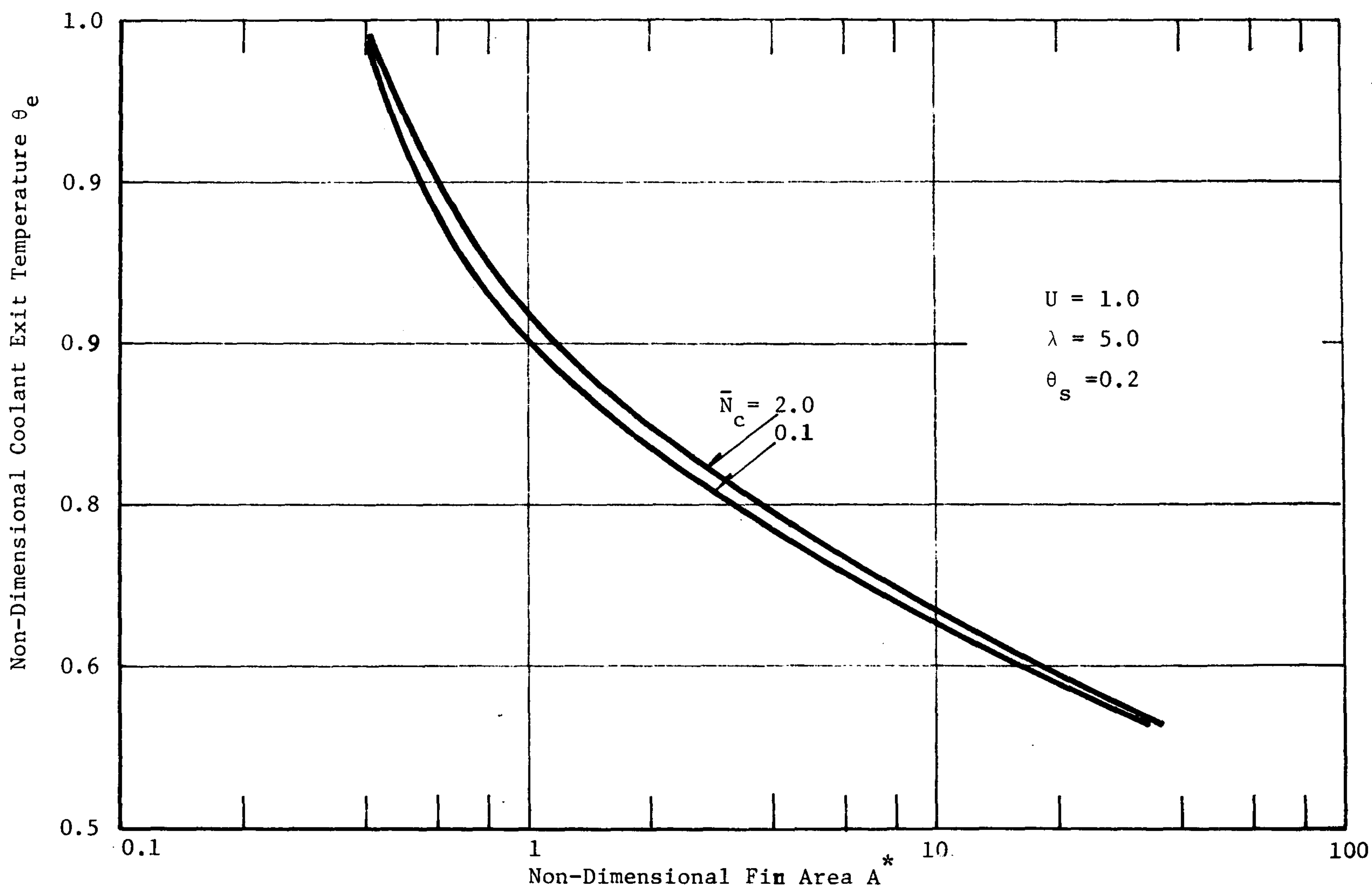

Figure 9. Non-Dimensional Coolant Exit Temperature VS. Non-Dimensional Fin Area For Varying Conduction Parameter and Short Fins

III. OPTIMIZATION OF THE RADIATOR SYSTEM

A. Purpose

In the design of the radiator system all parameters describing the system objectives and the system environment are prescribed beforehand while, at some occasions and within certain limits, the geometry of the system may be selected in the process of the design. This possible freedom of choice leads to the optimization of the radiator system geometry.

The primary purpose of the radiator system is to lower the coolant fluid temperature. For a given set of constraints the radiator with the lowest fluid exit temperature is considered here to be an optimum radiator system.

The set of constraints to be imposed on the optimum radiator system must be selected for a particular set of circumstances. In this analysis two sets out of many possible constraints were selected, that is, weight (or volume) and projected area of the radiator are considered to be prescribed.

The process of optimization proposed here, namely to find the minimum coolant fluid exit temperature for either a fixed radiator area or a fixed mass may be inverted to yield the minimum radiator area or weight for a prescribed cooling rate.

The optimization process can be outlined in the following steps:

(i) select, for given coolant exit temperature

$$\theta_e = \theta_e \{U, V, \bar{N}_c, \lambda; \theta_s\} \tag{28}$$

that optimum set $\{U, V, \bar{N}_c, \lambda\}$ of geometric parameters which constitutes, under selected constraints, the system geometry for the lowest possible coolant exit temperature,

(ii) transform the non-dimensional groups U, V and $\bar{N}_c$ into suitable, non-dimensional radiator length L^*, fin height H^* and panel thickness t^*,

(iii) calculate the physical dimensions L, H, t and d of the optimum radiator system.

If all fin elements operate under optimum conditions then the radiator as a whole operates under optimum conditions.

B. Problem Formulation

1. General

Of the five parameters U, V, $\bar{N}_c$, λ and θ_s which define θ_e only the first four can be considered independent while θ_s is determined by the incident radiant flux. For simplicity let us introduce the vector $\underset{\sim}{X}$ with components $x_1 = U$, $x_2 = V$, $x_3 = \bar{N}_c$ and $x_4 = \lambda$. The minimum coolant fluid exit temperature $\theta_e(\underset{\sim}{X})$, subject to $r \leq 3$ constraints may be obtained from the necessary conditions for a relative extremum

$$\frac{\partial \theta_e}{\partial x_i} = 0 \qquad i = 1, \ldots, (4 - r) \tag{33}$$

and the r constraints

$$\varphi_j(\underset{\sim}{X}) = 0 \qquad j = 1, \ldots, r \ . \tag{34}$$

This problem formulation implies that the r constraint equations can be solved explicitly for r components of $\underset{\sim}{X}$ which can be substituted into Eqs. 33 to yield (4 - r) equations for the remaining (4 - r) components. This was in effect achieved by replacing the Nusselt number relationships given in Eqs. 8 and 9 by a simpler relation of the form

$$N_{Nu} = c \ N_{Re}{}^{m} \ N_{Pr}{}^{n} \ (d/L)^{p} \tag{35}$$

with appropriate constants c = 0.023, m = 0.8, n = 1/3, p = 0, for $N_{Re} \geq 2{,}300$; and c = 1.86, m = n = p = 1/3 for $N_{Re} < 2{,}300$. The method of Lagrangian multipliers, however, permits the retention of the original constraints at the expense of additional, computational efforts in the case of fixed-mass constraints, since Eqs. 34 need not to be solved explicitely for $\underset{\sim}{X}$.

Two sets of constraints have been considered. Both sets have in common four sets of bounds

$$\left.\begin{aligned} U_{min} &\leq x_1 \leq U_{max} \\ V_{min} &\leq x_2 \leq V_{max} \\ (\bar{N}_c)_{min} &\leq x_3 \leq (\bar{N}_c)_{max} \\ \lambda_{min} &\leq x_4 \leq \lambda_{max} \end{aligned}\right\} \qquad (36)$$

which represent practical limits of the design parameters x_i, $i = 1, \ldots, 4$. Should the search procedure lead to the intersection of any of the above limits, then their appropriate equalities would enter the set of constraints. The other two constraints, namely of fixed volume or of fixed area, are discussed below.

2. Area Constraint

Introduce the characteristic length of a single fin element

$$L_o = \sqrt{\frac{\dot{m}\, c_p}{i\varepsilon\sigma T_o^3}} \qquad (37)$$

which may be interpreted as the length of a square fin element whose effectiveness is unity and which reduces the fluid temperature to zero. Dividing the physical dimensions L, H, t and d by the reference length L_o one obtains the non-dimensional geometric quantities L^*, H^*, t^* and d^*, respectively. The normalized projected fin element area is (for $t \ll d$)

$$A^* = A/L_o^2 = [2HL + dL]/L_o^2 \quad . \qquad (38)$$

Consequently, the constraint of "fixed area," the first of Eq. 34, becomes

$$\varphi_1 = 2H^*L^*[1 + 2/\lambda] - A^* = 0 \qquad (39)$$

or

$$\varphi_1(x_1,x_2,x_4) = x_1x_2(1 + 2/x_4) - A^* = 0 \qquad (40)$$

This constraint equation can be solved explicitly for any one of its arguments, regardless of applicable Nusselt number relationship (see comments following Eq. 34).

The constant area constraint may be supplemented by an additional constraint reflecting the preference for a particular fin panel thickness, arising from manufacturing considerations. A fixed panel thickness t, or equivalently t^* would require

$$\varphi_2(\underset{\sim}{X}) = t^* - \phi_1 x_2^2/x_3 \cdot N_{Nu}^2\,(\underset{\sim}{X}) = 0 \qquad (41)$$

where ϕ_1 is known beforehand

$$\phi_1 = (1/2i)\ (\pi k_f/k)^2\ k^2/(i\dot{m}\ C_p\ \varepsilon\sigma\ T_o^3) \tag{42}$$

Equation 41 depends on the Nusselt number relationship used. It becomes a <u>necessary</u> constraint when the optimization tends toward panel thicknesses which approach the tube diameter.

3. Volume Constraint

The volume constraint is stipulated by the desire to design a radiator system of minimum mass. The coolant mass may (liquids) or may not (gases) contribute significantly to the total mass.

The mass of one fin element is, under the assumption of a tube wall thickness, equal to the fin panel thickness

$$M^* = \frac{M}{\phi_1 M_o} = L^* t^* \, [2H^* + \pi(d^* - t^*)] \approx \frac{UV^3}{\bar{N}_c} \left[1 + \frac{2\pi}{\lambda}\right] N_{Nu}^2 \tag{43}$$

provided $t^* << d^*$. Here $M_o = \rho L_o^3$ is the reference mass and M the physical mass of the structure. With

$$\phi_2 = 2i(\rho_f/\rho)(k/k_f) \tag{44}$$

the coolant fluid mass may be expressed as

$$M_f^* = \frac{M_f}{\phi_1 M_o} = \phi_2 \, N_{Nu} \left(\frac{V}{\lambda}\right)^2 U \quad . \tag{45}$$

Consequently, the "volume constraint" becomes

$$\varphi_1(\underset{\sim}{X}) = M_{tot}^* - x_1 x_2^2 \, N_{Nu}(\underset{\sim}{X}) \left\{ x_2 \, [1 + 2\pi/x_4] N_{Nu}/x_3 + \phi_2/x_4^2 \right\} = 0 \tag{46}$$

where $M_{tot}^* = M^* + M_f^*$. Equation 46 reveals that while $\phi_2 > 1$ the fluid mass contributes to the total mass only when x_4 becomes small. For the expected cases of $x_4 \approx 10$ Eq. 46 reduces to a volume constraint as the influence of the density ratio ρ/ρ_f vanishes.

4. Sufficiency Requirements

An optimum radiator is found at $\underset{\sim}{X}_m$ when $\theta_e(\underset{\sim}{X}_m)$ is a minimum, that is, when in the vicinity of $\underset{\sim}{X}_m$

$$\theta_e(\underset{\sim}{X}_m + \underset{\sim}{\delta}) > \theta_e(\underset{\sim}{X}_m) \tag{47}$$

where $\underset{\sim}{\delta}$ is a small vector, with components $\{\Delta x_1, \ldots, \Delta x_4\}$, and with its endpoint on the hypersurface defined by Eqs. 34.

There exist analytical expressions for the sufficiency conditions in terms of second-order derivatives $\partial^2\theta_e/\partial x_i \partial x_j$ which are applicable when either all constraints can be solved explicitly as implied by Eqs. 33 and 34 or the number of constraints is at least two. Since, however, θ_e is obtained through numerical integration of Eqs. 24 and 29 and the sufficiency test is to be performed only once the potential optimum is found, it appears economical to evaluate Eq. 47 directly. Further developments concerning the sufficiency criteria are necessary at this time.

C. Solution

1. The Optimum

The remaining task is to solve the system of Eqs. 33 as they are obtained after substitution of Eqs. 34. The solution is obtained through an iterative process based on the Newton-Raphson procedure. Starting with an estimated set of parameters $\underset{\sim}{X}_1$ the iteration is carried on according to

$$\underset{\sim}{X}_{k+1} = \underset{\sim}{X}_k + \Delta\underset{\sim}{X}_k \ , \ k = 1,2, \ldots \tag{48}$$

where $\underset{\sim}{X}_k$ represents the current, $\underset{\sim}{X}_{k+1}$ the future parameter set and the increments $\Delta\underset{\sim}{X}_k$ are the solution to the system of linear algebraic equations

$$\underset{\sim}{Y}_k = (\underset{\approx}{A})_k(\Delta\underset{\sim}{X})_k \ . \tag{49}$$

The current components $(y_i)_k$ of $\underset{\sim}{Y}_k$ are the current values of the derivatives

$$y_i = - \frac{\partial\theta_e}{\partial x_i} \ , \ i = 1, \ldots, (4 - r) \ . \tag{50}$$

The current elements $\{A_{ij}\}_k$ of the square matrix $(\underset{\approx}{A})_k$ are the second-order derivatives

$$A_{ij} = \frac{\partial^2\theta_e}{\partial x_i \partial x_j} \ ; \ i,j = 1, \ldots, (4 - r) \ . \tag{51}$$

It is obvious that the matrix A_{ij} is symmetric and one needs to compute only $(5 - r)(4 - r)/2$ independent, second-order derivatives.

The derivatives in Eq. 50 and 51 are obtained from Eqs. 24 and 25 by first differentiating with respect to x_i, $i = 1, \ldots, (4 - r)$ and then interchanging the order of differentiation. This leads first to $2(4 - r)$ first-order, ordinary, non-linear differential equations for

$$\left.\begin{aligned} -\frac{dy_i}{d\zeta} &= \frac{d}{d\zeta}\left(\frac{\partial\theta_f}{\partial x_i}\right) \; , \; i = 1, \ldots, (4 - r) \\ -\frac{dy_i}{d\zeta} &= \frac{d}{d\zeta}\left(\frac{\partial\theta_b}{\partial x_i}\right) \; , \; i = (5 - r), \ldots, (8 - 2r) \end{aligned}\right. \tag{52}$$

and then to (5 - r)(4 - r) first-order, ordinary non-linear differential equations for

$$\left.\begin{aligned} \frac{dA_{ij}}{d\zeta} &= \frac{d}{d\zeta}\left(\frac{\partial^2\theta_f}{\partial x_i \partial x_j}\right) \\ \frac{dB_{ij}}{d\zeta} &= \frac{d}{d\zeta}\left(\frac{\partial^2\theta_b}{\partial x_i \partial x_j}\right). \end{aligned}\right\} \tag{53}$$

The second sets of equations in Eqs. 52 and 53 are necessary because of the dependence of θ_e on θ_b. All initial conditions, at $\zeta = 0$, for Eqs. 52 and 53 can be derived from Eqs. 26 and 30

$$y_i(0) = (y_o)_i \; , \; A_{ij}(0) = (A_o)_{ij} \; , \; B_{ij}(0) = (B_o)_{ij} \; . \tag{54}$$

In summary, one needs to integrate, together with Eqs. 24 and 29, the Eqs. 52 and 53, subject to the initial conditions given by Eqs. 26, 30 and 54. The integration is carried out from $\zeta = 0$ to $\zeta = 1$ where it yields not only θ_e and $\theta_b(1)$ but also all derivatives in Eqs. 49. This system is solved for the components of $\Delta\underset{\sim}{X}$, then a new set of parameters $\underset{\sim}{X}$ is computed from Eqs. 48 and the iterative cycle repeated. The repetitions are continued until either the first (4 - r) y_i's are small or one of the inequalities in Eqs. 36 is violated. Should that happen then an additional constraint is introduced, r is incremented by one, and the cycle is continued until either all limits in Eqs. 36 are reached or the remaining y_i's are sufficiently small. The result may or may not be an optimum for the initially chosen A^* or M^*. Finally, the potential optimum is tested in accordance with Eq. 47.

2. Parameter Transformation and Physical Dimension of Optimum Radiator System.

Once the optimum set $\left\{x_1,\ldots,x_4\right\}_m$ is found it is necessary to compute from this the physical dimensions L, H, t and d which define the geometry of the radiator system. To accomplish this task the set $\left\{x_1,\ldots,x_4\right\}_m$ is first transformed into the previously introduced set of normalized dimensions $\left\{L^*, H^*, t^*, d^*\right\}$. This transformation is, in general, not possible in explicit form. However, when the simplified Nusselt number relation of the form given by Eq. 35 is substituted for the more general expressions in Eqs. 8 and 9 then L^*, H^*, t^* and d^* can be expressed explicitely in terms of the parameters $\left\{x_1,\ldots,x_4\right\}$.

Define first the starred quantities.

$$N^*_{Gz} = \frac{c_p \dot{m}}{k_f L_o} \tag{55}$$

$$N^*_{Re} = \frac{4\dot{m}}{\pi\mu L_o} \tag{56}$$

$$N^*_{Nu} = c(N^*_{Re})m(N_{Pr})n \tag{57}$$

$$N^*_{c} = \frac{2\varepsilon\sigma T_o{}^3 L_o}{k} \tag{58}$$

which are essentially the unstarred quantities evaluated with the reference length instead of any other dimension and which are all known from system objectives and environmental conditions.

Next one needs to distinguish between laminar and turbulent coolant flow as the constants c, m, n and p are different for the two regimes.

For <u>laminar</u> flow, c = 1.86, m = n = p = 1/3.

$$L^* = \frac{N^*_{Gz}}{2\pi\sqrt{c^3}} \quad \sqrt{x_1{}^3} \tag{59}$$

$$H^* = \pi\sqrt{c^3}/N^*_{Gz} \frac{x_2}{\sqrt{x_1}} \tag{60}$$

$$t^* = \pi^2 c N^*_c/(N^*_{Gz})^2 \frac{x_2^{\;2}}{x_1 x_3} \tag{61}$$

and

$$d^* = 4\pi\sqrt{c^3}\ /N^*_{Gz} \frac{x_2}{x_4\sqrt{x_1}} \tag{62}$$

For turbulent flow, $c = 0.023$, $m = 4/5$, $n = 1/3$, $p = 0$.

With

$$\phi_3 = \pi N^*_{Nu}/N^*_{Gz} \tag{63}$$

one obtains

$$L^* = 1/[2^5\phi_3^{\;14}]^{1/9} x_1 \left[\frac{x_2}{x_4}\right]^{4/9} \tag{64}$$

$$H^* = 1/2[2^5\phi_3^{\;14}]^{1/9}\ (x_2^{\;5}x_4^{\;4})^{1/9} \tag{65}$$

$$t^* = N^*_c/4\cdot[2^5\phi_3^{\;14}]^{2/9} \frac{(x_2^{\;5}x_4^{\;4})^{2/9}}{x_3} \tag{66}$$

$$d^* = \left[2\phi_3 \frac{x_2}{x_4}\right]^{5/9} \tag{67}$$

Finally, multiplication of the starred quantities L^*, H^*, t^* and d^* by the reference length L_o gives the dimension L, H, t and d of the fin radiator system with optimum performance.

D. Results

The ultimate presentation of the results from the radiator system optimization is the plot of the optimum non-dimensional geometric parameters L*, H*, t*, and d* versus the non-dimensional enthalpy rejection $(1 - \Theta_e)$. These graphs would include three or more parameters, namely the sink temperature Θ_s and the parameters occurring in the constraint equations. However, the optimum geometry is not expected to depend strongly on all parameters because of physical considerations, and convenient graphical presentation should be possible.

As indicated in the Introduction, the systematic optimization based on analytical extrema search techniques was developed in addition to the original program objectives but could not be completed within the contract period. The graphical presentation of the optimum geometry in terms of intended enthalpy rejection is therefore not included in this report.

However, computer codes were written on the basis of the solution discussed in Section C. These codes produce the optimum radiator parameters L*, H*, t* and d* based on the Nusselt number relationship in Eq. 35, for any particular set of input parameters computed from mission requirements and environmental conditions. These results are described in detail in Chapter IV, Sections C. 4 and D. 4.

IV. COMPUTER CODES

A. Introduction

Three separate FØRTRAN codes were developed, one for the simplified radiator system simulation, one for the minimum area optimization and one for the minimum mass system optimization.

The three codes have several subprograms in common and could be united into a single code to avoid duplication. The simulation program, however, is more efficient as a single program and will remain a tool by itself. The optimization programs, on the other hand, are means to develop a tool, namely suitable working charts from which to read optimum system parameters. Once these diagrams are obtained, the codes are no longer needed.

The three FØRTRAN codes are discussed separately in the following three sections but reference is made in the description of subprograms which have previously been discussed.

B. Radiator System Simulation

1. Objective

The purpose of the Simplified Radiator System Simulation (SRSS) is to evaluate the analysis developed in Chapter II for determining the coolant exit temperature

$$\Theta_e(U,V,\bar{N}_c,\lambda;\Theta_s)$$

in terms of the five governing groups U, V, $\bar{N}_c$, λ and Θ_s defined by Eqs. 13, 19 through 22.

The coolant exit temperature and also the fin base temperature are obtained by integrating simultaneously Eqs. 24 and 29, subject to the initial conditions given by Eqs. 27 and 30. The integration is performed by the Runge Kutta integration procedure described in Reference [1], under variable step size mode.

The individual program units are discussed in the subsequent sections to the extent deemed necessary for the proper utilization of the code. Following the program description are presented the input and output specifications.

2. Deck Assembly

The Simplified Radiator System Simulation (SRSS) code consists of:

one		main program	MAIN
three	SUBRØUTINE	subprograms	RKS SDRV SCNTL
three	FUNCTIØN	subprograms	ETA PØLY

A block diagram is shown in Figure 7 below.

a. The MAIN Program accepts data input and lists the accepted input data. Following the input data management, the initial conditions for integration are set, an initial step size is computed and control variables are set to control the integration procedure. Then the Runge Kutta procedure RKS is called which performs the integration and indirectly the listing of results. After return from the integration, control is transferred to the start of the program for arbitrarily many repetitions of the program execution. When all input data are exhausted then control is transferred to a normal exit.

The input data preparation is discussed in Section 3 below.

The initial conditions are given through Eqs. 27 and 30. The dependent variables Θ_f and Θ_b are, during integration, placed in the array Y; Y(1) = Θ_f and Y(2) = Θ_b. Equation 27 leads to the statement Y(1) = 1.0 and Eq. 30 is solved for Θ_b by the Newton-Raphson iteration which follows the listing of accepted input data. The iteration process leads to the statement Y(2) = THETAB (= Θ_b).

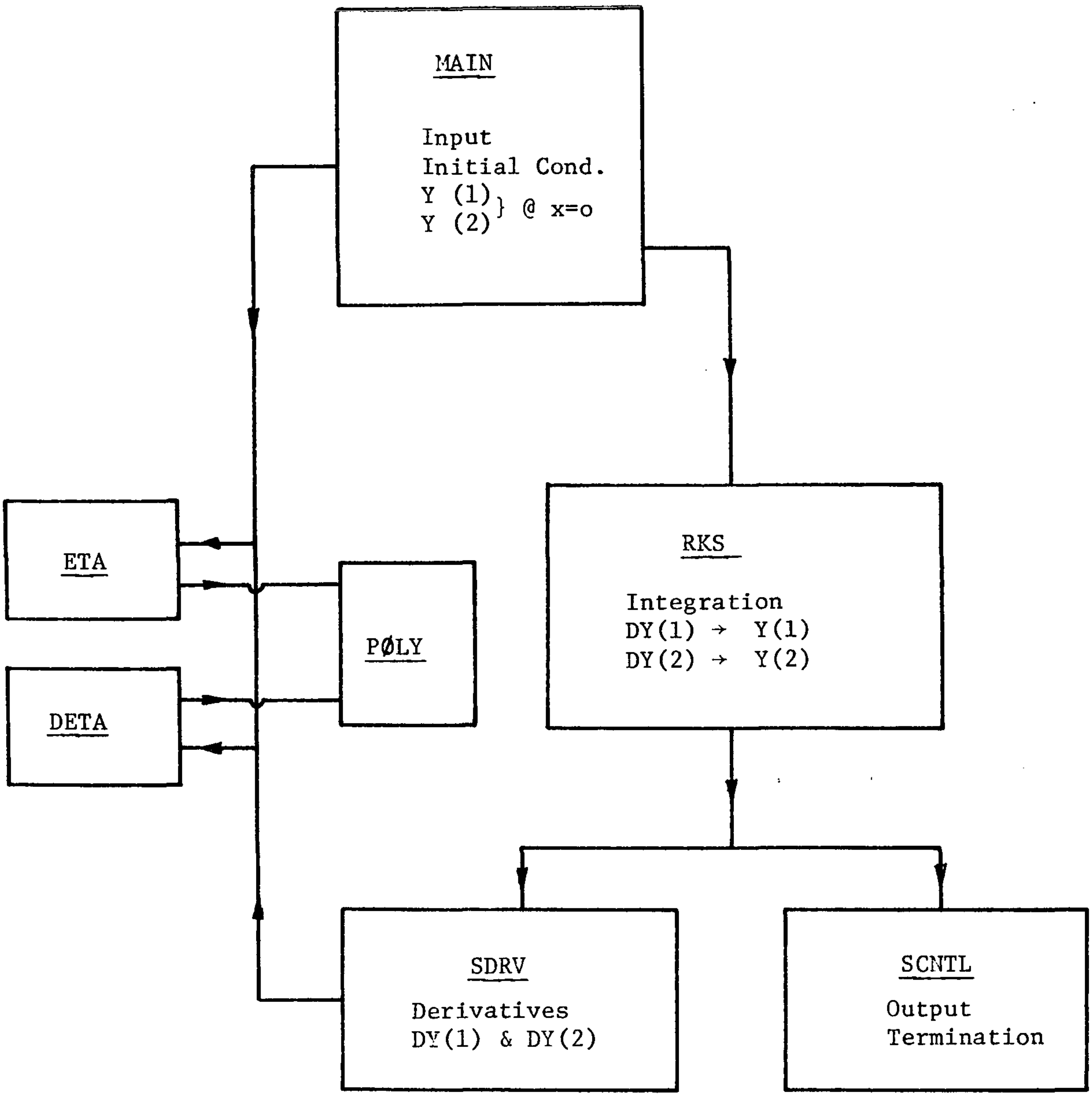

Figure 10. Block Diagram For Simplified Radiator System Simulation Code

Control variables which direct the Runge Kutta Integration RKS are:

ZZZ = 0.0	initial value for independent variable
DX = $1/(U(1 - \Theta_b))$	initial step size
IFVD = 0	variable step size selection
N = 2	two equations to be integrated
IBKP = 1	cut interval as required
NTRY = 1	normal integration mode
IERR = 0	normal error condition (DX $\neq$ 0)
A(I) = 5×10^{-5}; I = 1,2	absolute error per step
R(I) = 5×10^{-5}	relative error per step

Control variables which direct output during the integration are:

DLMT = DXWRT/500	error limit for output interval
XWRT = 0.0	first value of independent variable at first output listing
LSTEP = 0	controls change to fixed step, LSTEP = 1 once integration step exceeds write interval DXWRT
LCNT = 0	senses cut-back (LCNT = 1) of step size prior to listing of results; used to save uncut time step
ICNT = 0	integration step counter

The CALL statement for the Runge-Kutta SUB-RØUTINE is explained in Reference [1] together with the specification of necessary DIMENSIØN and EXTERNAL declarations. SDRV and SCNTL must be declared EXTERNAL. The necessary array allocations can be read from the listing of the RKS code: all arrays with variable dimension (N) must be declared in MAIN for N = 2 elements. Only the elements of Y, A and R must be specified prior to the CALL RKS statement.

b. The Runge-Kutta SUBRØUTINE RKS is described in detail in Reference [1]. The user is not expected to alter this code in any way and needs to know only the control features discussed in paragraph a above and the fact that RKS calls two subroutines SDRV and SCNTL.

c. The SUBRØUTINE SDRV serves to evaluate the derivatives $d\Theta_f/d\zeta$ = dY(1) and $d\Theta_b/d\zeta$ = dY(2) in accordance to Eqs. 24 and 29. The fin effectiveness $\eta\ (N_c)$ and its derivative $d\eta/dN_c$ are evaluated externally by the FUNCTIØN subprograms ETA and DETA.

SDRV is called from RKS and supplies the derivatives, via argument list, as functions of the independent variable X and dependent variables Y(1) and Y(2), to the integrating procedure RKS. SDRV is called four times per integration step.

d. The SUBRØUTINE SCNTL serves to provide

(i) output listing at prescribed intervals DXWRT,

(ii) transfer to fixed-step integration mode once the automatically selected (by RKS) step size DX exceeds the print-out interval DXWRT,

(iii) termination of the integration procedure and RETURN to the MAIN program (via RKS). The same program, with different WRITE commands is used in the Minimum Area and Minimum Mass Optimization codes.

In the argument list are available

(i) the dependent variables Y(1) and Y(2)

(ii) their derivatives DY(1) and DY(2)

(iii) the dependent variable X

(iv) the step size DX which may be altered for repeat of last step

(v) NTRY = 1 normal continuation of RKS
2 return from RKS to MAIN

	3	repeat with new DX
	4	restart
(vi)	IFVD = 0	variable integration step
	1	fixed integration step

e. The FUNCTION ETA Subprogram computes the fin effectiveness $\eta(N_c)$ for the unobstructed fin in accordance to Eqs. 32a and b. The power polynomial is evaluated by calling the FUNCTIØN PØLY subprogram [1].

FUNCTIØN ETA (X) receives the independent variable $X = N_c$ and returns the fin effectiveness ETA. The two functions represented by Eqs. 32 a and b apply strictly to zero sink temperature, but the effectiveness is insensitive to the sink temperature and the equations hold sufficiently well for $\Theta_s < 0.8\ \Theta_b$.

ETA (X) is called from MAIN and SDRV.

f. The FUNCTIØN DETA Subprogram computes the derivative $d\eta/dN_c$ of the fin effectiveness $\eta(N_c)$ for the unobstructed fin. The derivative is derived from Eqs. 32 a and b, and evaluated by calling the FUNCTIØN PØLY subprogram [1].

FUNCTIØN DETA (X) receives the independent variable $X = N_c$ and returns DETA = $d\eta/dN_c$. DETA (X) is called from MAIN and SDRV.

g. The FUNCTIØN PØLY Subprogram [1] serves to evaluate power polynomials f(x) of any degree (N - 1).

$$f(x) = \sum_{i=1}^{N} A_i X^i$$

by computing the recurrence expression

$$f_i = A_N$$

$$f_{k+1} = f_k X + A_{N-k}, k = 1,\dots(N-1)$$

(N-1) times. The argument list of FUNCTIØN PØLY (N, A, X) transfers to the subprogram the number N of coefficients A_i, the one-dimensional array A(N) and the independent variable X; it returns PØLY = f(x). PØLY is called from ETA and DETA.

3. Input Data Preparation

The non-dimensional groups

THETAS = Θ_s = T_s/T_o (19)

U = $\pi N_{Nu}/N_{Gz}$ (22)

V = $(i/\pi)(t/H)(k/k_f)(\bar{N}_c/N_{Nu})$ (21)

FNCBR = $\bar{N}_c$ = $2\varepsilon\sigma T_o^3 H^2/(kt)$ (20)

RHØD = λ = $4H/d$ (13)

DXWRT = $\Delta z/L$, output intervals

are computed and punched, in the above sequence, on two cards in

FØRMAT (5F16.8)

There may be arbitrarily many pairs of cards as specified above, one pair for each desired simulation. They will be executed in succession. The End-of-Job card will cause normal exit from the program.

The expected run time on the UNIVAC 1108 is 0.24 seconds per simulation plus necessary compiling and collecting times. There were 32 runs executed in 14.8 seconds.

4. Output Presentation

For each simulation are printed, on a separate page, first the governing parameters as read in:

THETAS = Θ_s ,	defined by	Eqs. 4 & 19
U ,	"	Eq. 22
V ,	"	Eq. 21
FNCBAR = $\bar{N}_c$,	"	Eq. 20
LAMBDA = λ ,	"	Eq. 13

and then a table consisting of seven columns. In the first two columns are listed, as functions of axial distance ZETA = ζ at the selected intervals

$\Delta\zeta$ = DXWRT, the coolant fluid temperature THETAF = θ_f and the fin base temperature THETAB = θ_b. The positions ZETA = ζ are listed in the sixth column.

The conduction parameter NC = N_c, the combined fin-plus-tube effectiveness ETABAR = $\bar{\eta}$, defined by Eq. 11, and the fin effectiveness (unobstructed fin, Eqs. 32 a and b) ETA = $\bar{\eta}$, are listed, respectively, in the third, fourth and fifth columns. The last columns indicate the integration steps required to reach the respective axial position ζ. The desired coolant fluid exit temperature θ_e $(U,V,\bar{N}_c,\lambda;\theta_s)$ is found in the last entry of the first column.

Figure 11 shows a typical output listing

C. Minimum Area Optimization

1. Objective

The Minimum Area Optimization (MAO) code serves to find the minimum coolant fluid exit temperature for a given projected fin panel area, on the basis of the optimization analysis developed in Chapter III and the constraints as discussed in Section III-2. From the results, evaluated for a number of selected areas, one can determine the minimum area required for a chosen heat rejection rate. The optimum parameter set $\underset{\sim}{X}$ is sought which satisfies Eqs. 33, 36, 40 and 41 by performing the iterations specified in Eqs. 48 through 53 until Eq. 33 is satisfied.

From the optimum parameter set $\underset{\sim}{X}$ one computes the normalized system dimensions L*, H*, t* and d* in accordance to Eqs. 59 through 62 or Eqs. 64 through 67, depending on the flow regime in the coolant channels.

The necessary derivatives occurring in Eqs. 50 and 51 are obtained by Runge-Kutta Integration of the first-order, ordinary, non-linear differential equations given by Eqs. 52 and 53.

The individual program units are discussed in the following sections. Reference is made to program description in Section 3 of this Chapter and to Reference [1].

THETA-S =	.200000	
U =	.100000+01	
V =	.100000+01	
FNCBAR =	.200000+00	
LAMBDA =	.250000+02	

THETAF	THETAB	NC	ETABAR	ETA	ZETA	STEP NO
1.000000	.724992	.076213	1.001229	.919403	.00	1
.973292	.713938	.072780	1.004461	.922709	.10	2
.948092	.703261	.069563	1.007515	.925832	.20	4
.924291	.692944	.066546	1.010401	.928785	.30	5
.901791	.682975	.063715	1.013130	.931576	.40	6
.880499	.673338	.061056	1.015711	.934216	.50	7
.860331	.664019	.058556	1.018154	.936714	.60	8
.841212	.655007	.056204	1.020465	.939079	.70	9
.823068	.646288	.053989	1.022655	.941319	.80	10
.805835	.637851	.051902	1.024729	.943441	.90	11
.789452	.629684	.049934	1.026696	.945453	1.00	12

FIGURE 11. SAMPLE RESULTS OF SIMPLIFIED RADIATOR SYSTEM SIMULATION

2. Deck Assembly

In its current state the Minimum Area Optimization (MAO) code consists of

1		main program	MAIN	
14	SUBRØUTINE	subprograms	RKS SDRV SCNTL	integration
			BNDCND RESBND REST	restraints
			NDERV MDER NCDERV MIXDER DERV	partial derivatives
			INITIAL INTMIX	initial conditions
			FMINV	matrix inversion
5	FUNCTIØN	subprograms	ETA DETA DDETA DDDETA PØLY	η and its derivatives

The basic concept of the program is as described in Chapter B before and the block diagram in Figure 7 applies in principle. The exceptions are

(i) there are 20 simultaneous differential equations to be solved:

2 for Θ_f and Θ_b

6 for the derivatives of Θ_f and Θ_b with respect to U, V and $\bar{N}_c$

12 for the second derivatives of Θ_f and Θ_b with respect to U, V, and $\bar{N}_c$,

(ii) the initial conditions on the last 18 differential equations are obtained in subroutines,

(iii) partial derivatives occurring in the differential equations for the last 18 of the above derivatives are computed by subroutines,

(iv) at the end of each integration, the system of Eq. 49 is solved by FMINV and the integration repeated until the optimum is reached.

a. The MAIN Program performs the same functions as described in Chapter IV, Section B.2.a. and, in addition, calls INITIAL and INTMIX which compute the remaining 18 initial conditions for the partial derivatives as defined through Eq. 54. Moreover, instead of returning to the next input data set after completion of an integration, new system parameters $\underset{\sim}{X}$ are computed by solving, via FMINV, the system of Eq. 49. The results are tested, through SUBRØUTINE BNDCND, to satisfy the Inequalities 36. If necessary, the newly computed parameters are adjusted to fall within the above limits, and the integration is repeated.

b. The SUBRØUTINES RKS, SDRV and SCNTL perform the integration as explained in Sections B.2.b, c and d. There are, however, 18 additional dependent variables Y(3), ..., Y(20) to be integrated by RKS and, consequently, 18 additional derivatives to be evaluated in accordance with Eqs. 52 and 53. These derivatives are evaluated by calling DERV for $\partial\Theta_f/\partial X_i$, $\partial\Theta_b/\partial X_i$, $\partial^2\Theta_f/\partial X_i{}^2$ and $\partial^2\Theta_b/\partial X_i{}^2$ and by calling MIXDER for $\partial^2\Theta_f/\partial X_i\partial X_j$ and $\partial^2\Theta_b/\partial X_i\partial X_j$, $i \neq j$. SCNTL contains a slightly altered output specification.

c. The Restraint SUBROUTINES BNDCND, RESBND and REST serve to, respectively,

(i) test newly computed components $\underset{\sim}{X}$ for compliance with Inequalities 36.

(ii) solve Eqs. 40 and 41 for x_3 and x_4.

When newly computed components $\underset{\sim}{X}$ do not comply with Inequalities 36 then they are set to meet the minimum and maximum conditions and the system of Eq. 49 is solved again, all from within BNDCND.

SUBRØUTINE REST is the only particular program which differentiates the two optimization modes, minimum area and minimum mass. It may be replaced to accommodate other constraints. BNDCND and REST are called from MAIN.

d. The SUBRØUTINES NDERV, MDER, NCDER, MIXDER and DERV are used to compute, respectively,

(i) N and its derivatives $dN/d\lambda$, $d^2N/d\lambda^2$ from Eq. 17,

(ii) M and its derivatives, up to third order, with respect to its arguments λ and N_c, in accordance to Eqs. 12, 15 and 16,

(iii) N_c and its derivatives $\partial N_c/\partial\Theta_b$, $\partial^2 N_c/\partial\Theta_b{}^2$ and $\partial^2 N_c/\partial T\partial\bar{N}_c$,

(iv) the mixed derivatives defined by Eq. 53 for $i \neq j$,

(v) the derivatives defined by Eqs. 52 and 53.

e. The SUBRØUTINES INITIAL and INTMIX serve to compute the initial conditions for the dependent variable Y(3), ..., Y(20), at the channel inlet $\zeta = 0$ which represent simple and mixed derivatives $\partial^2\Theta_f/\partial X_i\partial X_j$ and $\partial^2\Theta_b/\partial X_i\partial X_j$ in Eq. 53.

f. The SUBRØUTINE FMINV is capable of performing two related tasks:

(i) to invert a square, invertible matrix $\underset{\sim}{A}$

(ii) to solve a non-trivial system of equations $\underset{\sim}{A}\underset{\sim}{Z} = \underset{\sim}{X}$.

The solutions are obtained through a sequence of elementary row operations which lead from the properly augmented coefficient matrix to the row-reduced echelon matrix, a standard procedure described in most elementary introductions to linear algebra [1].

SUBROUTINE FMINV (A, X, N, M) accepts, through its argument list, the square, invertible matrix $\underset{\sim}{A}$ of rank N, and if task (i) above is intended, M is set equal to M = 2N. Then, upon return from FMINV there will be the inverted matrix $\underset{\sim}{A}^{-1}$ placed in XMAT(I,J) with I = 1, ..., N, J = N + 1, N + 2, ..., 2N. The two-dimensional array XMAT is transferred to the calling program via a CØMMØN declaration. When task (ii) above is intended, M is set equal to M = N + 1 and, upon return from FMINV, the unknown vector $\underset{\sim}{Z}$ is placed in the one-dimensional array $\underset{\sim}{X}$.

FMINV is used to solve Eq. 49 and is called from MAIN and from BNDCND.

50

g. The FUNCTIØN subprograms ETA, DETA and PØLY are described in Sections B.2.e, f, and g of this chapter.

h. The FUNCTIØN subprograms DDETA and DDDETA are used to compute the second and third-order derivatives $d^2\eta/dN_c^2$ and $d^3\eta/dN_c^3$ as derived from Eqs. 32 a and b. The derivatives are evaluated via FUNCTIØN PØLY subprogram.

3. Input Data Preparation

Each optimization run is carried out for a selected ASTAR = A^* (Eq. 38) and TSTAR = t^* (Eq. 41) as the independent variables. The necessity of specifying t^* is not yet completely established at this time but included in the program for additional flexibility and to prevent the search from reading excessively large values of t^*.

The optimization produces

(i) the optimum coolant exit temperature

(ii) the optimum parameters $\underset{\sim}{X} = \{x_1, \ldots, x_4\}$ from which to compute L^*, H^*, d^* and λ (Eqs. 59-67).

For each optimization one computes

THETAS = Θ_s	,	defined	by	Eqs. 19 & 4
U, starting value	,	"	"	Eq. 22
V, starting value	,	"	"	Eq. 21
FNCBR = $\bar{N}_c$ starting value	,	"	"	Eq. 20
ASTAR = A*	,	"	"	Eq. 40

and selects

DXWRT = $\Delta\zeta$ the axial interval for which results are to be printed.

The first five values are punched on one card in

FØRMAT (5F16.8).

The last value is punched on a second card in

FORMAT(F16.8)

There may be arbitrarily many pairs of cards as specified above, one pair for each optimization run. They will be carried out in succession. The End-of-Job card will cause normal exit from the program.

The expected run time per optimization is approximately 15 seconds plus necessary compilation and collection times.

4. Presentation of Results

Each iteration produces one page of output, first the three parameters THETAS, ASTAR and TSTARI as read in. Next the current values of the system parameters U, V, FNCBAR = $\bar{N}_c$, LAMBDA = λ and the identification of the flow regime, followed by the iteration count for establishing initial conditions

The table lists, as functions of $\zeta = Z/L$, in this order

THETAF = $\theta_f(\zeta)$ coolant temperature

THETAB = $\theta_b(\zeta)$ fin base temperature

NC = $N_c(\zeta) = \bar{N}_c \theta_b^3$, Eq. 14

ETABAR = $\bar{\eta}$, defined by Eq. 11

ETA = η, defined by Eq. 32

M defined by Eq. 12

N defined by Eq. 17

X = ζ, the non-dimensional distance along the channel.

Following the table is a list of the three system parameters U, V, $\bar{N}_c$ and the first and second derivatives of Θ_f with respect to these parameters. Finally are listed the newly computed changes of the above parameters.

A typical listing of the results is shown in Figure 9.

02

THETA-S =	.70300000
ASTAR =	.675000
U =	250.000000
V =	.000050
ENCBR =	.000100
LAMBDA =	.037736

NUMBER OF ITERATIONS = 14

THETAF	THETAB	NC	ETABAR	ETA	M	N	ZETA
1.0000000	.9979813	.0000894	53.9999905	.9998844	.1801294	53.8198819	.0000000
.9211307	.9180655	.0000776	53.9999919	.9999098	.1801264	53.8198819	.2000000
.8677819	.8669164	.0000652	53.9999928	.9999242	.1801247	53.8198819	.4000000
.8309971	.8303729	.0000573	53.9999933	.9999334	.1801236	53.8198819	.6000000
.8039970	.8035309	.0000519	53.9999938	.9999397	.1801229	53.8198819	.8000000
.7835871	.7832305	.0000480	53.9999938	.9999441	.1801224	53.8198819	1.0000000

PARAMETERS AT END OF TUBE

THETAF = .783587

U =	.250000+03	V =	.500000-04	NCBAR =	.100000-03
DTFDU =	-.247174-05	DTFDV =	.117912+00	DTFNCBAR =	.771548-04
D2TFD2U =	.197315-07	D2TFD2V =	.979100+01	D2TFD2NCBAR =	.844942-03
D2TFDUDV =	.530919-05	D2TFDUDNCBAR =	.105256-05	D2TFDVDNCBAR =	.525590+01
DELTA U =	.126478+03	DELTA V =	-.363925-04	DELTA NCBAR =	-.224943-01

OPTIMUM REACHED

FIGURE 12. RESULTS LISTING OF MINIMUM AREA OPTIMIZATION

D. Minimum Mass Optimization

1. Objective

The objective of the Minimum Mass Optimization (MMO) code is identical to that discussed in Section C above, except that the constraints are replaced by the constraints developed in Section III-3 of this report; the minimum mass is thought. Equation 46 replaces Eqs. 40 and 41.

2. Deck Assembly

The MMO code is assembled identically to the MAO code described in Section C.2 above except the SUBRØUTINE REST is replaced by the SUBRØUTINE CØNST which serves to compute $x_4 = \lambda$ from Eq. 46, together with all the necessary derivatives of λ with respect to $x_1 = U$, $x_2 = V$ and $x_3 = \bar{N}_c$. CØNST is called from MAIN. RESBND is not required.

3. Input Data Preparation

Each optimization is carried out for a selected M^*_{tot} as the independent variable and for the two parameters N^*_{Nu} and ϕ_2, defined by Eqs. 46, 42 and 57, respectively. The optimization produces

(u) the optimum (minimum) coolant exit temperature

(ii) the optimum system parameters $\underset{\sim}{X}$ from which to compute $\{L^*, H^*, t^*, d^*\}$ through Eqs. 59-67.

For each optimization are computed

THETAS = Θ_s	defined by Eqs. 4 & 19	
U } starting values		22
V }		21
FNCBAR = $\bar{N}_c$		20
TØTMAS = M^*_{tot} , selected		46
PHI2 = ϕ_2		44

ALPHA = $\alpha = N^*_{Nu}\sqrt{c}\ \sqrt[3]{2\pi/N^*_{Gz}}$

and selected

DXWRT = $\Delta\zeta$ the axial interval for which results are to be printed.

DA, DR absolute and relative integration step errors.

The data values are punched on the data card in NAMELIST format. The NAMELIST is called INPUT.

There may be arbitrarily many additional input data cards as specified above, or with only one datum, one NAMELIST for each optimization run. They will be carried out in succession until an End-of-Job card causes normal exit from the program.

The expected run time per optimization is approximately 20 seconds plus necessary machine preparation times.

4. Representation of Results

Results are presented in the format identical to that described in section C.4 above, except that instead of the single TSTAR1 are listed MSTAR = M*, PHI2 = ϕ_2 and FNUSTR = N^*_{Nu}, as read in. The representative output example is shown in Figure 10.

```
        THETA-S =          .60000000
              U =        13.363489
              V =         2.792685
          FNCBR =         5.731686
         TOTMAS =        95.000000
           PHI2 =      2000.000000
          ALPHA =         5.000000
         LAMBDA =  31297273.250000
```

NUMBER OF ITERATIONS = 8

THETAF	THETAB	NC	ETABAR	ETA	M	N	X
1.0000000	.7686288	2.6027493	.3775592	.3775592	1.0000000	.0000001	.0000000
.7134382	.6486320	1.0316816	.5284999	.5284998	1.0000000	.0000001	.2000000
.6322974	.6138584	1.1674339	.5047780	.5047780	1.0000000	.0000001	.4000000
.6088749	.6038113	1.2169663	.4969006	.4969006	1.0000000	.0000001	.6000000
.6024210	.6010400	1.2322024	.4945601	.4945600	1.0000000	.0000001	.8000000
.6006592	.6002832	1.2364472	.4939149	.4939148	1.0000000	.0000001	1.0000000

PARAMETERS AT END OF TUBE

THETAF = .600659

	U =	.133635+02	V =	.279269+01	NCBAR =	.573169+01
	DTFDU =	-.496030-04	DTFDV =	-.131301-03	DTFNCBAR =	-.384840-04
	D2TFD2U =	.105462-03	D2TFD2V =	.455954+04	D2TFD2NCBAR =	.120270+03
	D2TFDUDV =	-.229946-02	D2TFDUDNCBAR =	.294970-03	D2TFDVDNCBAR =	-.740525+03
INCR. SOUGHT	DELTA U =	.100314+00	DELTA V =	-.807482-01	DELTA NCBAR =	-.497181+00
INCR. USED	DELTA U =	.100314+00	DELTA V =	-.832025-01	DELTA NCBAR =	-.497181+00

INSIGNIFICANT IMPROVEMENT OVER LAST STEP

FIGURE 13. RESULTS LISTING OF MINIMUM MASS OPTIMIZATION

V. CONCLUSIONS

The work presented herein resulted in a simplified radiator system analysis and a systematic optimization procedure.

Comparison of the simplified with the rigorous [1] system analyses indicated that the agreement between the two analyses can be expected to be within approximately 5%.

The optimization procedures carried out lead frequently to an optimum on the boundaries of the parameter domain. Mass optimization tends toward widley spaced tubes between thin fins.

Optimization was originally intended to be achieved through the parametric study of trends rather than through analytical procedures. Within the resources provided by this contract two analytical iterative optimization procedures were developed beyond the original goal of work. These procedures lead to the maximum heat rejection for given system area or system mass. Future work should be aimed toward the development of

(1) a parameter domain within which relative extrema exist,

(2) suitable working diagrams, through repeated applications of the developed codes, which depict the optimum geometric system parameters and produce either the least area or the least weight requirements for a given heat rejection rate.

BIBLIOGRAPHY

1. Black, W. Z. and Wulff, W., "Study of Design Parameters of Space Base and Space Shuttle Heat Rejection Systems," Final Report, Contract No. NAS 9-10415, Georgia Institute of Technology, Atlanta; April 1972.

2. Black, W. Z. and Wulff, W., "Study of Design Parameters of Space Base and Space Shuttle Heat Rejection Systems," Initial Phases I, II and III, Contract No. NAS 9-10415, Georgia Institute of Technology, Atlanta; February 1971.

3. Hausen, H., VDI Beihefte, Verfahrenstechnik, 1943, p. 91.

4. Seban, R. A. and Shimazaki, T. T., "Heat Transfer to a Fluid Flowing Turbulently in a Smooth Pipe with Walls at Constant Temperature," Trans. ASME, Vol. 73, p. 803, 1951.

5. Lieblein, S., "Analysis of Temperature Distribution and Radiant Heat Transfer Along a Rectangular Fin." NASA TN D-196, 1959.

6. Bartas, J. G. and Sellers, W. H., "Radiation Fin Effectiveness," J. Heat Transfer, C82, 73-75, 1960.

7. Sparrow, E. M. and Cess, R. D., Radiation Heat Transfer, Brooks/Cole Publishing Company, p. 173, Fig. 6-9, 1966.

8. dto, p. 175, Fig. 6-11.

APPENDIX A

COMPUTER CODE FOR SIMPLIFIED RADIATOR SYSTEM SIMULATION

```
@FOR,IS MAIN
FOR S9A-07/25/72-22:59:39 (,0)

MAIN PROGRAM

STORAGE USED: CODE(1) 000464; DATA(0) 000355; BLANK COMMON(2) 000000

COMMON BLOCKS:

0003   SDR    000014
0004   MCT    000006

EXTERNAL REFERENCES (BLOCK, NAME)

0005   SDRV
0006   SCNTL
0007   ETA
0010   DETA
0011   RKS
0012   NINTR$
0013   NWDU$
0014   NIO2$
0015   NRDU$
0016   SQRT
0017   ASIN
0020   NSTOP$

STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

0001   000150 100   0001   000327 105L   0001   000345 110L   0000   000265 120F   0001   000405 232G
0001   000007 45L   0000   000233 5F     0000   000234 50F    0000   000236 55F    0001   000131 85L
0001   000145 90L   0000   000304 900F   0000   000311 905F   0001   000452 920L   0000   000313 925F
0001   000147 95L   0001   000460 950L   0000 R 000030 A      0000 R 000213 AYY    0000 R 000205 AI
0000 R 000214 BYY   0010 R 000000 DETA   0000 R 000220 DETZ   0003 R 000007 DFM    0003   000010 DFMX
0003 R 000011 DFNX  0004 R 000000 DLMT   0000 R 000060 DLY    0003   000012 DROU   0003   000013 DROV
0000 R 000224 DX    0004 R 000003 DXWRT  0000 R 000014 DY     0000 R 000140 DYST   0000 R 000221 DYY
0007 R 000000 ETA   0000 R 000216 ETY    0000 R 000215 ETYY   0003 R 000006 FM     0003 R 000005 FN
0000 R 000212 FNC   0003 R 000002 FNCBR  0000 I 000232 I      0000 I 000227 IBKP   0004 I 000001 ICNT
0000 I 000231 IERR  0000 I 000225 IFVD   0004 I 000005 LCNT   0004 I 000004 LSTEP  0000 I 000226 N
0000 I 000207 NCT   0000 I 000230 NTRY   0000 R 000074 PD     0000 R 000044 R      0003 R 000004 RHOD
0006 R 000000 SCNTL 0000 R 000110 SD     0005 R 000000 SDRV   0000 R 000206 THETAB 0000 R 000204 THETAS
0003 R 000003 TH4   0000 R 000210 T2     0000 R 000211 T3     0003 R 000000 U      0003 R 000001 V
0004 R 000002 XWRT  0000 R 000000 Y      0000 R 000124 YS     0000 R 000170 YSIMP  0000 R 000154 YST
0000 R 000217 YY    0000 R 000222 ZZ     0000 R 000223 ZZZ


00100      1*     C
00100      2*     C     SIMPLIFIED RADIATOR SYSTEM SIMULATION
00100      3*     C     ***************************************
00100      4*     C
00100      5*     C           THETAS,U,V,FNCBR, AND RHOD ARE SYSTEM PARAMETERS
00100      6*     C           DXWRT IS THE INCREMENT OF AXIAL DISTANCE FOR OUTPUT LISTING
```

```
C                  FORMAT (5F16.8)
C
      EXTERNAL SDRV,SCNTL
      DIMENSION Y(12),DY(12),A(12),R(12),DLY(12),PD(12),SD(12),YS(12),
     1          DYST(12),YST(12),YSIMP(12)
      COMMON /SDR/ U,V,FNCBR,TH4,RHOD,FN,FM,DFM,DFMX,DFNX,DROU,DROV/MCT/
     1             DLMT,ICNT,XWRT,DXWRT,LSTEP,LCNT
C
C
      FNN(X)   = (X+3.570796-SQRT(X*(X+2.0))-ASIN(1.0/(1.0+X)))/X
      FFS(X)   = (SQRT(X*(X+2.0))+ASIN(1.0/(X+1.0))-1.570796)/X
      FFN(X,Y)= 1.0-Y*(0.1460*Y-0.02866)/X
      DFFN(X,Y)=(-0.2920*Y+0.02866)/X
      DFFNX(X,Y)= Y*(0.1460*Y-0.02866)/(X*X)
      DFFS(X) = -SQRT((X+2.0)/X)/(X*(X+1.))+(1.5708-ASIN(1.0/(X+1.0)))/
     1          (X*X)
C

      WRITE(6,5)
    5 FORMAT(1H1)
C
   45 READ(5,50,END=950) THETAS,U,V,FNCBR,RHOD,DXWRT
   50 FORMAT(5F16.8)
      WRITE(6,55) THETAS,U,V,FNCBR,RHOD
   55 FORMAT(1H0,10X,10HTHETA-S = ,F19.6,/,
     1          11X,10H      U = ,E20.6,/,
     2          11X,10H      V = ,E20.6,/,
     3          11X,10H FNCBAR = ,E20.6,/,
     4          11X,10H LAMBDA = ,E20.6,///)
C
      IF(THETAS.GE. 1.0) GO TO 920
      FN       = FNN(RHOD)
      DFNX     = (-FN+1.0-SQRT(RHOD*(RHOD+2.0)))/RHOD
      TH4      = THETAS**4
      IF(V.GT.0.2) GO TO 90
      A1       = 2.0+1.0/(V*ETA(FNCBR))
      IF(A1.GE.2.45) GO TO 85
      THETAB   = 1.0-1.0/(2.0*A1)
      GO TO 95
   85 THETAB   = 1.0-(A1-SQRT(A1**2-6.0))/6.0
      GO TO 95
   90 THETAB   = 0.9
   95 NCT      = 0
  100 T2       = THETAB**2
      T3       = T2*THETAB
      FNC      = FNCBR*T3
      FM       = FFN(RHOD,FNC)*FFS(RHOD)
      DFM      = DFFN(RHOD,FNC)*FFS(RHOD)
      AYY      = T3*THETAB-TH4
      BYY      = 1.0-THETAB
      ETYY     = ETA(FNC)
      ETY      = ETYY*FM+FN
      YY       = 1.0/V-ETY*AYY/BYY
      DETZ     = (DFM*ETYY+FM*DETA(FNC))*3.0*T2*FNCBR
      DYY      = -AYY/BYY*(DETZ+ETY/BYY)-4.0*ETY*T3/BYY
      ZZ       = THETAB-YY/DYY
      IF(NCT.GT.20) WRITE(6,900)
```

```
00177     64*            IF(ZZ.LT.1.0) GO TO 105
00201     65*            THETAB  = (THETAB+1.0)/2.0
00202     66*            NCT     = NCT+1
00203     67*            GO TO 100
00204     68*        105 IF(ABS(ZZ-THETAB)/ZZ.LT.1.0E-06) GO TO 110
00206     69*            NCT     = NCT+1
00207     70*            THETAB  = ZZ
00210     71*            GO TO 100
00210     72*      C
00211     73*        110 Y(1)    = 1.0
00212     74*            Y(2)    = THETAB
00213     75*            ZZZ     = 0.0
00214     76*            DX      = 1.0/(U*(1.0-THETAB))
00215     77*            IFVD    = 0
00216     78*            IF(DX.GT.DXWRT) DX = DXWRT
00220     79*            N       = 2
00221     80*            IBKP    = 1
00222     81*            NTRY    = 1
00223     82*            IERR    = 0
00223     83*      C
00224     84*            DLMT    = DXWRT/500.0
00225     85*            XWRT    = 0.0
00226     86*            LSTEP   = 0
00227     87*            LCNT    = 0
00230     88*            ICNT    = 0
00231     89*            DO 115 I=1,12
00234     90*            A(I)    = 5.0E-05
00235     91*        115 R(I)    = 5.0E-05
00235     92*      C
00237     93*            WRITE(6,120)
00241     94*        120 FORMAT(1H0,13X,6HTHETAF,14X,6HTHETAB,16X,2HNC,17X,6HETABAR,15X,
00241     95*          1        3HETA,,8X,4HZETA,6X,7HSTEP NO,/)
00241     96*      C
00242     97*            CALL RKS(SDRV,SCNTL,Y,DY,A,R,ZZZ,DX,N,IFVD,IBKP,NTRY,IERR,
00242     98*          1          DLY,PD,SD,YS,YST,DYST,YSIMP)
00243     99*            WRITE(6,905)
00245    100*            GO TO 45
00246    101*        900 FORMAT(1H0,20HNEWTON-RAPHSON FAILS)
00247    102*        905 FORMAT(1H ,/////)
00250    103*        920 WRITE(6,925)
00252    104*        925 FORMAT(1H0,37HRADIATIVE HEATING CANNOT BE SIMULATED)
00253    105*            GO TO 45
00254    106*        950 STOP
00255    107*            END

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS SUB1
FOR S9A-07/25/72-23:00:14 (,0)

   SUBROUTINE SDRV      ENTRY POINT 000150

   STORAGE USED: CODE(1) 000155; DATA(0) 000037; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   SDR    000014
    0004   SOC    000003

   EXTERNAL REFERENCES (BLOCK, NAME)

    0005   ETA
    0006   DETA
    0007   ASIN
    0010   SQRT
    0011   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001    000136 10L       0001    000103 5L        0000 R 000005 A0        0000 R 000006 A1        0000 R 000007 A2
    0000 R 000010 A3        0000 R 000011 A4        0000 R 000012 A5        0006 R 000000 DETA      0003 R 000007 DFM
    0003    000010 DFMX      0003    000011 DFNX      0003    000012 DROU      0003    000013 DROV      0005 R 000000 ETA
    0004 R 000001 ETY       0004 R 000002 ETZ       0000 R 000003 FFN       0000 R 000004 FFS       0003 R 000006 FM
    0003 R 000005 FN        0003 R 000002 FNCBR     0000    000024 INJP$     0003 R 000004 RHOD      0003 R 000003 TH4
    0003 R 000000 U         0003    000001 V         0000 R 000000 Y2        0000 R 000001 Y3        0000 R 000002 Y4
    0004 R 000000 Z


00101      1*           SUBROUTINE SDRV(Y,DY,X)
00101      2*      C
00101      3*      C    COMPUTES DERIVATIVES
00101      4*      C
00103      5*           DIMENSION Y(12),DY(12)
00104      6*           COMMON /SDR/ U,V,FNCBR,TH4,RHOD,FN,FM,DFM,DFMX,DFNX,DROU,DROV/SOC/
00104      7*          1             Z,ETY,ETZ
00104      8*      C
00105      9*           Y2       = Y(2)**2
00106     10*           Y3       = Y(2)*Y2
00107     11*           Y4       = Y2*Y2
00110     12*           Z        = FNCBR*Y3
00111     13*           ETY      = ETA(Z)
00112     14*           FFN      = 1.-Z*(0.1460*Z-0.02866)/RHOD
00113     15*           FFS      = (SQRT(RHOD*(RHOD+2.))+ASIN(1./(RHOD+1.))-1.5708)/RHOD
00114     16*           FM       = FFN*FFS
00115     17*           DFM      = (-0.2920*Z+0.02866)/RHOD*FFS
00116     18*           ETZ      = ETY*FM+FN
00117     19*           DY(1)    = -U*(Y(1)-Y(2))
00120     20*           A0       = Y4-TH4
```

```
00121      21*            IF(A0.GT.1.0E-08) GO TO 5
00123      22*            DY(2)    = 0.0
00124      23*            GO TO 10
00125      24*          5 A1       = Y(1)-Y(2)
00126      25*            A2       = (ETY*DFM+DETA(Z)*FM)
00127      26*            A3       = 3.*A2*Y2*FNCBR/ETZ
00130      27*            A4       = 4.0*Y3/A0
00131      28*            A5       = 1.0+A1*(A3+A4)
00132      29*            DY(2)    = DY(1)/A5
00132      30*      C
00133      31*         10 CONTINUE
00134      32*            RETURN
00135      33*            END

       END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS SUB2
FOR S9A-07/25/72-23:01:21 (,0)

     SUBROUTINE SCNTL     ENTRY POINT 000146

     STORAGE USED: CODE(1) 000173; DATA(0) 000016; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   SDC    000003
      0004   MCT    000005

     EXTERNAL REFERENCES (BLOCK, NAME)

      0005   NWDU$
      0006   NIO2$
      0007   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000054 5L        0001   000061 50L       0000   000002 55F       0001   000134 60L       0004 R 000000 DLMT
      0000 R 000000 DXSTR     0004 R 000003 DXWRT     0003 R 000001 ETY       0003 R 000002 ETZ       0004 I 000001 ICNT
      0000   000010 INJP$     0000 I 000001 LCNT      0004 I 000004 LSTEP     0004 R 000002 XWRT      0003 R 000000 Z


 00101      1*          SUBROUTINE SCNTL(Y,DY,DX,X,NTRY,IFVD)
 00101      2*     C
 00101      3*     C    CONTROLS INTEGRATION
 00101      4*     C
 00103      5*          DIMENSION Y(12),DY(12)
 00104      6*          COMMON /SDC/ Z,ETY,ETZ /MCT/ DLMT,ICNT,XWRT,DXWRT,LSTEP
 00105      7*          ICNT     = ICNT+1
 00106      8*          IF(DX.GE.DXWRT.AND.ICNT.GT.1) LSTEP = 1
 00110      9*          IF(ABS(X-XWRT).LT.DLMT) GO TO 50
 00112     10*          IF(XWRT.GT.X) GO TO 5
 00112     11*     C
 00114     12*          DXSTR    = DX
 00115     13*          DX       = DX+XWRT-X
 00116     14*          LCNT     = 1
 00117     15*          NTRY     = 3
 00120     16*          RETURN
 00120     17*     C
 00121     18*        5 NTRY     = 1
 00122     19*          RETURN
 00122     20*     C
 00123     21*       50 WRITE(6,55) Y(1),Y(2),Z,ETZ,ETY,X,ICNT
 00134     22*       55 FORMAT(1H ,5F20.6,F10.2,I10)
 00135     23*          IF(LCNT.EQ.1) DX = DXSTR
 00137     24*          LCNT     = 0
 00140     25*          IF(ABS(1.0-XWRT).LE.DLMT) GO TO 60
 00142     26*          XWRT     = XWRT+DXWRT
```

```
00143     27*           NTRY     = 1
00144     28*           IF(LSTEP.EQ.0) RETURN
00146     29*           DX       = DXWRT
00147     30*           IFVD     = 1
00150     31*           RETURN
00150     32*     C
00151     33*        60 NTRY     = 2
00152     34*           RETURN
00153     35*           END

      END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
@FOR,IS SUB3
FOR S9A-07/25/72-23:01:28 (,0)

     FUNCTION ETA          ENTRY POINT 000036

     STORAGE USED: CODE(1) 000044; DATA(0) 000022; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003   POLY
      0004   EXP
      0005   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000016 1L       0000 R 000001 A        0000 R 000010 B        0000 R 000000 ETA       0000    000014 INJP$
      0003 R 000000 POLY


00101      1*              FUNCTION ETA(X)
00101      2*      C
00101      3*      C       COMPUTES FIN EFFECTIVENESS ETA
00101      4*      C
00103      5*              DIMENSION A(7), B(2)
00104      6*              DATA A(1),A(2),A(3),A(4),A(5),A(6),A(7)/0.10E+01, -0.1163143E+01,
00104      7*             1     0.1478836E+01, -0.1267550E+01, 0.6325223E+00, -0.1627067E+00,
00104      8*             2     0.1654223E-01/ B(1),B(2)/0.6866095E+00, -0.2297718E+00/
00116      9*              IF(X.GT.2.5) GO TO 1
00120     10*              ETA   = POLY(7,A,X)
00121     11*              RETURN
00122     12*            1 ETA   = B(1)*EXP(B(2)*X)
00123     13*              RETURN
00124     14*              END

        END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS SUB4
FOR S9A-07/25/72-23:01:35 (,0)

   FUNCTION DETA        ENTRY POINT 000036

   STORAGE USED: CODE(1) 000044; DATA(0) 000021; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   POLY
    0004   EXP
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000016 1L        0000 R 000001 A         0000 R 000007 B         0000 R 000000 DETA       0000    000013 INJP$
    0003 R 000000 POLY


00101       1*             FUNCTION DETA(X)
00101       2*      C
00101       3*      C      COMPUTES FIRST DERIVATIVE     DETA/DNC
00101       4*      C
00103       5*             DIMENSION A(6), B(2)
00104       6*             DATA A(1),A(2),A(3),A(4),A(5),A(6)/-0.1163143E+01, 0.2957672E+01,
00104       7*            1     -0.3802650E+01, 0.2530089E+01, -0.8135335E+00, 0.9925338E-01/
00104       8*            2     B(1),B(2)/-0.1577635E+00, -0.2297718E+00/
00115       9*             IF(X.GT.2.5) GO TO 1
00117      10*             DETA  = POLY(6,A,X)
00120      11*             RETURN
00121      12*           1 DETA  = B(1)*EXP(B(2)*X)
00122      13*             RETURN
00123      14*             END

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS SUB5
FOR S9A-07/25/72-23:01:39 (,0)

   FUNCTION POLY        ENTRY POINT 000036

   STORAGE USED: CODE(1) 000044; DATA(0) 000015; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000012 1075     0000   000003 INJP$     0000 I 000002 K          0000 I 000001 L          0000 R 000000 POLY


00101       1*              FUNCTION POLY(N,A,X)
00101       2*      C
00101       3*      C       EVALUATES POLYNOMIALS
00101       4*      C
00103       5*              DIMENSION A(N)
00104       6*              POLY = 0.
00105       7*              L = N
00106       8*              DO 1 K=1,N
00111       9*              POLY = POLY*X+A(L)
00112      10*            1 L = L-1
00114      11*              RETURN
00115      12*              END
       END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS SUB6
FOR S9A-07/25/72-23:01:48 (,0)

   SUBROUTINE RKS        ENTRY POINT 000643

   STORAGE USED: CODE(1) 001040; DATA(0) 000064; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NERR2$
    0004   NEXP5$
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000010 10L      0001   000313 110L     0001   000333 120L     0001   000045 126G     0001   000343 130L
    0001   000071 140G     0001   000355 140L     0001   000105 146G     0001   000130 156G     0001   000417 160L
    0001   000150 164G     0001   000177 174G     0001   000500 185L     0001   000510 190L     0001   000013 20L
    0001   000232 205G     0001   000270 217G     0001   000524 220L     0001   000530 230L     0001   000543 240L
    0001   000374 243G     0001   000032 25L      0001   000552 250L     0001   000554 251L     0001   000572 257L
    0001   000604 259L     0001   000623 270L     0001   000456 300L     0001   000615 336G     0001   000054 40L
    0001   000060 45L      0001   000006 5L       0001   000076 50L      0001   000123 70L      0001   000135 80L
    0000 R 000014 AM       0000 R 000007 AMAX     0000 R 000011 C        0000 R 000010 D        0000 R 000001 DDT
    0000 R 000003 DELT     0000 R 000012 E        0000 R 000000 FR10     0000 I 000004 I        0000 I 000005 IFLAG
    0000   000030 INJP$    0000 I 000002 ISYMP    0000 I 000013 J        0000 R 000006 S


  00101      1*          SUBROUTINE RKS(DERIV,CNTRL,Y,DY,A,R,T,DEL,N,IFVD,IBKP,NTRY,      DG000100
  00101      2*         1IERR,DELY,PD,SD,YS,YST,DYST,YSIMP)                                DG000200
  00101      3*     C
  00101      4*     C    INTEGRATION
  00101      5*     C
  00103      6*          DIMENSION Y(N),DY(N),A(N),R(N),DELY(N),                          DG0  3
  00103      7*         1PD(N),SD(N),YS(N),DYST(N),YST(N),YSIMP(N)                        DG00040
  00104      8*          EXTERNAL  DERIV, CNTRL                                           DG000500
  00104      9*     C    FR10 IS  FIFTH ROOT OF TEN                                       DG00 60
  00105     10*          FR10=1.5848932                                                   DG000700
  00106     11*          IERR=0                                                           DG000800
  00106     12*     C      YS  CONTAINS Y VALUES AT LEFT END POINT OF INTEGRATION INTERVAL DG00090
  00106     13*     C                                                                     DG001000
  00106     14*     C      YSIMP CONTAINS Y FOR SIMPSONS RULE CHECK  CHECK NOT MADE FOR   DG001100
  00106     15*     C          FIXED STEP MODE    ISYMP IS CONTROL PARAMETER =1,FIXED,2 VARDG00120
  00106     16*     C                                                                     DG001300
  00106     17*     C      IF FIXED STEP SIZE GO ONE INTERVAL OF LENGTH DELT AND RETURN TO DG00140
  00106     18*     C         CNTRL, IF VAR GO TWO INTERVALS BEFORE RETURN TO CONTRL      DG00150
  00106     19*     C                                                                     DG001600
  00106     20*     C      IFVD  = 0     VARIABLE INTERVAL                                DG00170
  00106     21*     C            = 1     FIXED                                            DG001800
  00106     22*     C      IBKP  = 0     CUT INTERVAL ONCE BEFORE REPEAT  (UNDER IFVD=0 ) DG00190
  00106     23*     C            = 1     CUT AS REQUIRED                                  DG0 200
  00106     24*     C      NTRY  = 1     CONTINUE INTEGRATING                             DG002100
```

```
00106   C             2     RETURN FROM RKS                                       DG0 220
00106   C             3     STEP REPEATED WITH NEW DELT                           DG002300
00106   C             4     RESTART                                               DG00240
00106   C      IERR  = 0    NORMAL                                                DG002500
00106   C            -1     DELT=0, RETURN FROM RKS                               DG002600
00106   C             1     A(I)+ R(I)*ABS(Y(I)) = 0. , RETURN FROM RKS           DG002700
00107        5 IF(DEL) 20,10,20                                                   DG002800
00112       10 IERR=-1                                                            DG002900
00113          GO TO 270                                                          D7003000
00114       20 CALL DERIV(Y,DY,T)                                                 DG003100
00115          NTRY=1                                                             DG003200
00116          CALL CNTRL(Y,DY,DEL,T,NTRY,IFVD)                                   DG003300
00117       25 DDT=DEL                                                            DG003400
00120          IF(IFVD) 40,30,40                                                  DG003500
00123       30 ISYMP=2                                                            DG003600
00124          DELT=DEL/2.                                                        DG00370
00125          DO 31 I=1,N                                                        D7003800
00130       31 SD(I)=0.0                                                          DG00390
00132          IFLAG=1                                                            DG00400
00133          S=1.                                                               DG004100
00134          GO TO 45                                                           DG004200
00135       40 ISYMP=1                                                            DG004300
00136          DELT=DEL                                                           DG00440
00137       45 DO 46 I=1,N                                                        DG004500
00142          YST(I)=Y(I)                                                        DG00460
00143       46 DYST(I)=DY(I)                                                      DG0 470
00145       50 DO 60 I=1,N                                                        DG004800
00150          DELY(I)=DELT*DY(I)                                                 DG004900
00151          PD(I)=DELY(I)                                                      DG00500
00152       60 CONTINUE                                                           DG005100
00154          GO TO (80,70),ISYMP                                                DG005200
00155       70 DO 71 I=1,N                                                        DG005300
00160       71 SD(I)=SD(I)+S*DY(I)                                                DG005400
00162       80 T=T+DELT/2.                                                        DG005500
00163          DO 85 I=1,N                                                        DG005600
00166          YS(I)=Y(I)                                                         DG00570
00167          Y(I)=YS(I)+DELY(I)/2.                                              DG005800
00170       85 CONTINUE                                                           DG005900
00172          CALL DERIV(Y,DY,T)                                                 DG006000
00173          DO 90 I=1,N                                                        DG006100
00176          DELY(I)=DELT*DY(I)                                                 DG00620
00177          PD(I)=PD(I)+2.*DELY(I)                                             DG006300
00200          Y(I)=YS(I)+DELY(I)/2.                                              DG00640
00201       90 CONTINUE                                                           DG006500
00203          CALL DERIV(Y,DY,T)                                                 DG006600
00204          DO 95 I=1,N                                                        DG00670
00207          DELY(I)=DELT*DY(I)                                                 DG00680
00210          PD(I)=PD(I)+2.*DELY(I)                                             DG006900
00211          Y(I)=YS(I)+DELY(I)                                                 DG00700
00212       95 CONTINUE                                                           DG007100
00214          T=T+DELT/2.                                                        DG007200
00215          CALL DERIV(Y,DY,T)                                                 DG00730
00216          DO 100 I=1,N                                                       DG007400
00221          DELY(I)=DELT*DY(I)                                                 DG007500
00222          PD(I)=PD(I)+DELY(I)                                                DG00760
00223          Y(I)=YS(I)+PD(I)/6.                                                DG00770
00224      100 CONTINUE                                                           DG007800
```

```
00226            GO TO (110,120),ISYMP                           DG007900
00227        110 NTRY=1                                          DG008000
00230            CALL DERIV(Y,DY,T)                              DG008100
00231            CALL CNTRL(Y,DY,DEL,T,NTRY,IFVD)                DG00820
00232            GO TO 300                                       DG008300
00233        120 GO TO (130,140),IFLAG                           DG008400
00234        130 S=4.                                            DG008500
00235            IFLAG=2                                         DG008600
00236            CALL DERIV(Y,DY,T)                              DG008700
00237            GO TO 50                                        D7008800
00240        140 CALL DERIV(Y,DY,T)                              DG008900
00241            AMAX =0.0                                       DG009000
00242            DO 180 I=1,N                                    DG009100
00245            SD(I)=SD(I)+DY(I)                               DG009200
00246            YSIMP(I)=YST(I)+DELT*SD(I)/3.                   DG009300
00247            D   =ABS(Y(I)-YSIMP(I))                         DG00940
00250            C   =A(I)+R(I)*ABS(Y(I))                        DG009500
00251            IF(C    ) 160,150,160                           DG00960
00254        150 IERR=1                                          DG009700
00255            GO TO 270                                       DG009800
00256        160 E   =ABS(D   /C    )                            DG009900
00257            AMAX=AMAX1(AMAX,E)                              DG01000
00260        180 CONTINUE                                        D7010100
00262            IF(AMAX-1.) 215,215,230                         DG01020
00265        215 NTRY= 1                                         DG010300
00266            CALL CNTRL(Y,DY,DEL,T,NTRY,IFVD)                DG010400
00267        300 IF(NTRY-1) 185,185,310                          DG01050
00272        310 IF(NTRY-2) 270,270,330                          DG01060
00275        330 IF(NTRY-3) 340,340,5                            DG01070
00300        340 T=T-DDT                                         DG01080
00301            IF(DEL) 259,10,259                              DG010900
00304        185 GO TO (40,190),ISYMP                            DG01100
00305        190 IF(AMAX-.75) 200,25,220                         DG011100
00310        200 IF(AMAX-.075) 210,25,25                         DG01120
00313        210 DEL=DEL*FR10                                    DG011300
00314            GO TO 25                                        DG011400
00315        220 DEL=DEL/FR10                                    DG011500
00316            GO TO 25                                        DG011600
00317        230 I =1+ IBKP                                      DG011700
00320            GO TO (240,250),I                               DG011800
00321        240 T=T-DEL                                         DG011900
00322            DEL=DEL/FR10                                    DG012000
00323            GO TO  259                                      DG01210
00324        250 J=1                                             DG012200
00325        251 AM=AMAX/10.**J                                  DG012300
00326            IF(1.-AM) 255,257,257                           DG01240
00331        255 J=J+1                                           DG012500
00332            GO TO 251                                       DG01260
00333        257 T=T-DEL                                         DG01270
00334            DEL=DEL/(FR10**J)                               DG01280
00335        259 DO 245 I=1,N                                    DG01290
00340            DY(I)=DYST(I)                                   DG013000
00341        245 Y(I)=YST(I)                                     DG013100
00343            GO TO 25                                        DG013200
00344        270 RETURN                                          DG013300
00345            END                                             DG013400
```

```
END OF COMPILATION:        NO  DIAGNOSTICS.
2
```

```
@ XQT
MAP 0023-07/25-23:02

ADDRESS LIMITS     001000 013206      040000 044714
STARTING ADDRESS  012523

WORDS DECIMAL       5255 IBANK         2509 OBANK

              SEGMENT MAIN             001000 013206      040000 044714

NSWTC$/FOR                1   001000 001021
NRBLK$/FOR                1   001022 001044
NRWND$/FOR                1   001045 001124      2   040000 040011
NWEF$/FOR                 1   001125 001326      2   040012 040031
NFTCH$/FOR                1   001327 001617      2   040032 040067
NBDCV$/FOR                1   001620 001752      2   040070 040125
NFTV$/FOR                 1   001753 001775
NCNVT$/FOR                1   001776 002222      2   040126 040215
NCLOS$/FOR                1   002223 002371      2   040216 040247
NWBLK$/FOR                1   002372 002513
NBSBL$/FOR                1   002514 002550
NUPDA$/FOR                1   002551 002603
NBFOO$/FOR                                       2   040250 042451
NININ$/FOR                1   002604 003014      2   042452 042463
NINPT$/FOR                1   003015 003657      2   042464 042503
NOTIN$/FOR                1   003660 004173      2   042504 042507
NOUT$/FOR                 1   004174 005162      2   042510 042534
NFMT$/FOR                 1   005163 006041      2   042535 042611
NIOER$/FOR                1   006042 006214      2   042612 042716
NFCHK$/FOR                1   006215 007076      2   042717 043055
                                                 4   043056 043127
NTAB$/FOR                                        2   043130 043166
ERU$/MISC
NIBJF$/FOR                1   007077 007141      2   043167 043167
TIR$/TECH                 1   007142 007626      0   043170 043220
                                                 2   043221 043500
SQRT$/FOR                 1   007627 007667      2   043501 043512
ASINCOS$/FOR              1   007670 010104      0   043513 043540
NIER$/FOR                 1   010105 010166      2   043541 043674
NOBJF$/FOR                1   010167 010233
EXP$/FOR                  1   010234 010323      2   043675 043715
NEXP5$/FOR                1   010324 010411      2   043716 043725
NERR$/FOR                 1   010412 010736      2   043726 044071
SDR (COMMON BLOCK)                                   044072 044105
MCT (COMMON BLOCK)                                   044106 044113
SDC (COMMON BLOCK)                                   044114 044116
BLANK$COMMON (COMMON BLOCK)
SUB6                      1   010737 011776      0   044117 044202
                                                 2   BLANK$COMMON
```

```
SUB5        1  011777 012042    0  044203 044217
                                2  BLANK$COMMON
SUB4        1  012043 012106    0  044220 044240
                                2  BLANK$COMMON
SUB3        1  012107 012152    0  044241 044262
                                2  BLANK$COMMON
SUB2        1  012153 012345    0  044263 044300
            3  SDC              2  BLANK$COMMON
                                4  MCT
SUB1        1  012346 012522    0  044301 044337
            3  SDR              2  BLANK$COMMON
                                4  SDC
MAIN        1  012523 013206    0  044340 044714
            3  SDR              2  BLANK$COMMON
                                4  MCT

SYS$*RLIB$. LEVEL 63
END OF COLLECTION - TIME 1.677 SECONDS
```

APPENDIX B

COMPUTER CODE FOR MINIMUM

AREA OPTIMIZATION

```
2   @FOR,IS MAIN
    FOR S9A-07/27/72-17:43:15 (,0)

    MAIN PROGRAM

    STORAGE USED: CODE(1) 001045; DATA(0) 000677; BLANK COMMON(2) 000000

     COMMON BLOCKS:

     0003   BLKI    000004
     0004   BLKD    000002
     0005   PCT     000005
     0006   BLKR    000002
     0007   BLK3    000002

    EXTERNAL REFERENCES (BLOCK, NAME)

     0010   SERV
     0011   SCNTL
     0012   REST
     0013   NDERV
     0014   ETA
     0015   DETA
     0016   ACDERV
     0017   VDER
     0020   INTIAL
     0021   INTMIX
     0022   RKS
     0023   FMINV
     0024   BLDCND
     0025   PESOND
     0026   NIMTR$
     0027   NRDU$
     0030   NIO2$
     0031   NWDU$
     0032   ASIN
     0033   SQRT
     0034   NIO1$
     0035   NSTOP$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0001   000002 1L        0001   000163 100L      0000   000421 1000F     0001   000333 105L      0001   000351 108L
     0001   000354 109L      0001   000361 110L      0000   000423 2000F     0000   000454 2010F     0000   000467 2040F
     0000   000505 205F      0000   000622 2050F     0000   000616 2070F     0001   001033 2080L     0000   000631 2085F
     0000   000514 210F      0000   000521 215F      0000   000602 220F      0000   000533 230F      0001   000516 233G
     0000   000547 240F      0000   000564 250F      0001   001024 300L      0001   001041 3000L     0001   000725 340G
     0001   000020 50L       0001   000160 90L       0000   000462 900F      0000 R 000050 A         0000 R 000337 AM
     0000 R 000353 ASTA      0000 R 000362 AYY       0004   000003 A1        0004   000004 A2        0004   000005 A3
     0004   000006 A4        0004   000007 A5        0004   000010 A6        0000 R 000363 BYY       0015 R 000000 DETA
     0000 R 000367 DETZ      0003   000023 DE1DL     0003   000021 DE4DTB    0003 R 000013 DFM       0003 R 000012 DFMX
     0003 R 000010 DFNX      0006 R 000006 DLDNCB    0006 R 000001 DLDU      0006 R 000002 DLDV      0005 R 000000 DLMT
     0000 R 000120 DLY       0007   000000 DNCBDU    0007   000001 DNCBDV    0003 R 000007 DNCDT1    0000 R 000405 DX
     0005 R 000003 DXWRT     0000 R 000024 DY        0000 R 000240 DYST      0000 R 000370 DYY       0000 R 000373 D2CD2T
```

```
0000 R 000374 D2CTNB   0000 R 000402 D2E4LN   0000 R 000376 D2E4LU   0000 R 000400 D2E4LV   0000 R 000401 D2E4TN
0000 R 000375 D2E4TU   0000 R 000377 D2E4TV   0003 R 000016 D2FMLL   0006 R 000003 D2LDUU   0006 R 000004 D2LDUV
0006 R 000005 D2LDVV   0006 R 000007 D2LNCB   0006 R 000010 D2LUNB   0006 R 000011 D2LVNB   0003 R 000015 D2MDNC
0003 R 000014 D2MLNC   0003 R 000011 D2NDLL   0004 R 000001 D3M2LN   0004 R 000002 D3M2NL   0004   000011 EB
0014 R 000000 ETA      0000 R 000365 ETY      0000 R 000364 ETYY     0003   000022 E3      0003 R 000002 FM
0003 R 000003 FN       0003 R 000005 FNC      0003 R 000004 FNCBR    0000 I 000403 I        0000 I 000410 IRKP
0005 I 000001 ICNT     0000 I 000412 IERR     0000 I 000406 IFVD     0000 I 000413 LCNT     0005 I 000004 LSTEP
0000 I 000407 N        0000 I 000357 NCT      0000 I 000354 NIT      0000 I 000414 NITMAX   0000 I 000411 NTRY
0000 R 000144 PD       0000 R 000074 R        0000 R 000355 RHOD     0006 R 000000 RLAM     0011 R 000000 SCNTL
0000 R 000170 SD       0010 R 000000 SDRV     0003 R 000000 THETAB   0000 R 000372 THETAF   0003 R 000001 THETAS
0000 R 000356 TH4      0000 R 000360 T2       0000 R 000361 T3       0004 R 000000 U        0003 R 000006 V
0000 R 000334 XM       0005 R 000002 XWRT     0000 R 000000 Y        0000 R 000214 YS       0000 R 000310 YSIMP
0000 R 000264 YST      0000 R 000366 YY       0003 R 000017 Y1       0003 R 000020 Y2       0000 R 000371 ZZ
0000 R 000404 ZZZ      0000 R 000415 Z1       0000 R 000416 Z2       0000 R 000417 Z3       0000 R 000420 Z5

00100        C
00100        C     RADIATOR SYSTEMS OPTIMIZATION FOR MINIMUM AREA
00100        C
00101              EXTERNAL SDRV,SCNTL
00103              DIMENSION Y(20),DY(20),A(20),R(20),DLY(20),PD(20),SD(20),
00103             1          YS(20),DYST(20),YST(20),YSIMP(20),XM(3),AM(3,4)
00104              COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
00104             1          DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DE1DL/
00104             2     BLKD/U,D3M2LN,D3M2NL,A1,A2,A3,A4,A5,A6,EB
00104             3      /MCT/DLMT,ICNT,XWRT,DXWRT,LSTEP
00104             4     /BLKP/RLAM,DLDU,DLDV,D2LDUU,D2LDUV,D2LDVV,DLDNCB,
00104             5          D2LNCB,D2LUNB,D2LVNB
00104             6     /BLKB/DNCBDU,DNCBDV
00105              FFS(X)   = (SQRT(X*(X+2.0))+ASIN(1.0/(X+1.0))-1.570796)/X
00106              FFN(X,Y)= 1.0-Y*(0.1460*Y-0.02866)/X
00107              DFFN(X,Y)=(-0.2920*Y+0.02866)/X
00110            1 READ(5,1000,END=3000) THETAS,U,V,FNCBR,ASTAR,DXWRT
00120        100 FORMAT(5F16.8)
00121              NIT    =  1
00122          5   IF (U.LT.4.00E-06.OR.U.GT.250.1) GO TO 1
00124              IF (V.LT.4.00E-05.OR.V.GT.50.01) GO TO 1
00126              IF(FNCBR.LT.0.00009.OR.FNCBR.GT.4.01) GO TO 1
00130          5   CALL REST(U,V,ASTAR,RLAM,DLDU,DLDV,D2LDUU,D2LDUV,D2LDVV,
00130             1DLDNCB,D2LNCB,D2LUNB,D2LVNB)
00131              CALL NDERV(RLAM,FN,DFNX,D2NDLL)
00132              RHOD     = RLAM
00133              WRITE(6,2000) THETAS,ASTAR,U,V,FNCBR,RLAM
00143        200   FORMAT (1H1,10X,10HTHETA-S = F20.8/
00143             4             11X,10H  ASTAR = F20.6//
00143             1             11X,10H      U = F20.6/
00143             2             11X,10H      V = F20.6/
00143             3             11X,10H  FNCBR = F20.6/
00143             5             11X,10H LAMBDA = F20.6///)
00144              WRITE(6,2010) NIT
00147        201   FORMAT (5X*NUMBER OF ITERATIONS = *I2)
00150              TH4      = THETAS**4
00151              IF(V.GT.0.2) GO TO 90
00153              IF(V.LE.1.0E-10) GO TO 108
00155          9   THETAB   = 0.9
```

```
   95 NCT      = 0
  100 T2       = THETAB**2
      T3       = T2*THETAB
      FNC      = FNCBR*T3
      FM       = FFN(RHOD,FNC)*FFS(RHOD)
      DFM      = DFFN(RHOD,FNC)*FFS(RHOD)
      AYY      = T3*THETAB-TH4
      BYY      = 1.0-THETAB
      ETYY     = ETA(FNC)
      ETY      = ETYY*FM+FN
      YY       = 1.0/V-ETY+AYY/BYY
      DETZ     = (DFM*ETYY+FM*DETA(FNC))*3.0*T2*FNCBR
      DYY      = -AYY/BYY+(DETZ+ETY/BYY)-4.0*ETY*T3/BYY
      ZZ       = THETAB-YY/DYY
      IF(NCT.GT.20) GO TO 109
      IF(ZZ.LT.1.0) GO TO 105
      THETAB   = (THETAB+1.0)/2.0
      NCT      = NCT+1
      GO TO 100
  105 IF(ABS(ZZ-THETAB)/ZZ.LT.1.0E-06) GO TO 110
      NCT            = NCT+1
      THETAB   = ZZ
      GO TO 100
  108 THETAB         = 1.0
      GO TO 110
  109 WRITE(6,900)
C
  110 Y(1)     = 1.0
      Y(2)     = THETAB
  900 FORMAT(1H0,20HNEWTON-RAPHSON FAILS)
      THETAF   = Y(1)
      Y1       = Y(1)
      Y2       = Y(2)
       CALL NCDERV(FNCBR,THETAB,FNC,DNCDT1,D2CD2T,D2CTNB)
      CALL MDER(RLAM,FNC,FM,DFMX,DFM,D2FMLL,D2MDNC,D2MLNC,D3M2LN,D3M2NL)
      CALL INTIAL(DLDU,D2LDUU,Y(3),Y(4),Y(5),Y(6),0.,1.,D2E4TU,D2E4LU)
      CALL INTIAL(DLDV,D2LDVV,Y(7),Y(8),Y(9),Y(10),0.,0.,D2E4TV,D2E4LV)
      CALL INTIAL(DLDNCB,D2LNCB,Y(11),Y(12),Y(13),Y(14),1.,1.,D2E4TN,
     1                D2E4LN)
      CALL INTMIX(D2E4TV,Y(4),DLDU,D2E4LV,D2LDUV,Y(16),Y(15))
      CALL INTMIX(D2E4TN,Y(4),DLDU,D2E4LN,D2LUNB,Y(18),Y(17))
      CALL INTMIX(D2E4TN,Y(8),DLDV,D2E4LN,D2LVNB,Y(20),Y(19))
      DO 115 I=1,20
      A(I)     = 1.0E-05
  115 R(I)     = 1.0E-05
C
      WRITE(6,2040)
 2040 FORMAT(/7X,6HTHETAF,9X,6HTHETAB,9X,2HNC,13X,6HETABAR,
     1           9X,3HETA,13X,1HM,14X,1HN,14X,4HZETA/)
C
      ZZZ      = 0.0
      DX       = 1.0/(U*(1.0-THETAB))
      IFVD     = 0
      IF(DX.GT.DXWRT) DX = DXWRT
      N        = 20
      IBKP     = 1
      NTRY     = 1
```

```
00253     97*          IERR     = 0
00253     98*     C
00254     99*          DLMT     = DXWRT/500.0
00255    100*          XWRT     = 0.0
00256    101*          LSTEP    = 0
00257    102*          LCNT     = 0
00260    103*          ICNT     = 0
00261    104*          CALL RKS(SDRV,SCNTL,Y,DY,A,R,ZZZ,DX,N,IFVD,IBKP,NTRY,IERR,
00261    105*         1          DLY,PD,SD,YS,YST,DYST,YSIMP)
00262    106*          XM(1)    =  -Y(3)
00263    107*          XM(2)    =  -Y(7)
00264    108*          XM(3)    = -Y(11)
00265    109*          AM(1,1) =  Y(5)
00266    110*          AM(1,2) =  Y(15)
00267    111*          AM(1,3) =  Y(17)
00270    112*          AM(2,1) =  Y(15)
00271    113*          AM(2,2) =  Y(9)
00272    114*          AM(2,3) =  Y(19)
00273    115*          AM(3,1) = Y(17)
00274    116*          AM(3,2) = Y(19)
00275    117*          AM(3,3) = Y(13)
00276    118*          CALL FMINV(AM,XM,3,4)
00277    119*          WRITE (6,205)
00301    120*      205 FORMAT (///45X,27H PARAMETERS AT END OF TUBE //)
00302    121*          WRITE (6,210) Y(1)
00305    122*      210 FORMAT (1X,51X,8HTHETAE =,F10.6,/)
00306    123*          WRITE (6,215) U,V,FNCBR
00313    124*      215 FORMAT (1X,23X,3HU =,E14.6,16X,3HV =,E14.6,12X,7HNCBAR =,E14.6,/)
00314    125*          WRITE (6,230) Y(3),Y(7),Y(11)
00321    126*      230 FORMAT (1X,19X,7HDTFDU =,E14.6,12X,7HDTFDV =,E14.6,9X,
00321    127*         1        10HDTFNCBAR =,E14.6,/)
00322    128*          WRITE (6,240) Y(5),Y(9),Y(13)
00327    129*      240 FORMAT (1X,17X,9HD2TFD2U =,E14.6,10X,9HD2TFD2V =,E14.6,6X,
00327    130*         1        13HD2TFD2NCBAR =,E14.6,/)
00330    131*          WRITE (6,250) Y(15),Y(17),Y(19)
00335    132*      250 FORMAT (1X,16X,10HD2TFDUDV =,E14.6,5X,14HD2TFDUDNCBAR =,E14.6,
00335    133*         1        5X,14HD2TFDVDNCBAR =,E14.6,/)
00336    134*          WRITE(6,220) (XM(I),I=1,3)
00344    135*      220 FORMAT (/18X9HDELTA U =,E14.6,10X,9HDELTA V =,E14.6,6X,
00344    136*         1         13HDELTA NCBAR =,E14.6/)
00345    137*          NITMAX = 20
00346    138*      200 IF (NIT.GT.NITMAX) GO TO 300
00350    139*          NIT      =  NIT+1
00351    140*          Z1       = U
00352    141*          Z2       = V
00353    142*          Z3       = FNCBR
00354    143*          CALL BNDCND(Y,U,V,FNCBR,YM)
00355    144*          CALL RESBND(U,V,XM,ASTAR)
00356    145*          Z5       = Y(1)-THETAS
00357    146*          IF(Z5.LT.0.0001) GO TO 2080
00361    147*          IF (ABS(XM(1))/Z1.GT.0.0001) GO TO 50
00363    148*          IF (ABS(XM(2))/Z2.GT.0.0001) GO TO 50
00365    149*          IF(ABS(XM(3))/Z3.GT.0.0001) GO TO 50
00367    150*          WRITE(6,2070)
00371    151*     2070 FORMAT(1H0,15HOPTIMUM REACHED)
00372    152*          GO TO 1
00373    153*      300 WRITE(6,2050) NITMAX
```

```
00376      2050 FORMAT (1X,30H NUMBER OF ITERATIONS EXCEEDS ,I2)
00377           GO TO 1
00400      2080 WRITE(6,2085)
00402      2085 FORMAT(1H0,24HVANISHING HEAT REJECTION)
00403           GO TO 1
00404      3000 STOP
C0405           END

     END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
@FOR,IS RKS
FOR S9A-07/27/72-17:43:28 (,0)

    SUBROUTINE RKS        ENTRY POINT 000643

    STORAGE USED: CODE(1) 001040; DATA(0) 000064; BLANK COMMON(2) 000000

    EXTERNAL REFERENCES (BLOCK, NAME)

     0003   NERR2$
     0004   NEXP5$
     0005   NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0001     000010 10L      0001     000313 110L     0001     000333 120L     0001     000045 126G     0001     000343 130L
     0001     000071 146      0001     000355 140L     0001     000105 146G     0001     000130 156G     0001     000417 160L
     0001     000150 164      0001     000177 174G     0001     000500 185L     0001     000510 190L     0001     000013 20L
     0001     000232 205      0001     000270 217G     0001     000524 220L     0001     000530 230L     0001     000543 240L
     0001     000374 243      0001     000032 25L      0001     000552 250L     0001     000554 251L     0001     000572 257L
     0001     000604 259      0001     000623 270L     0001     000456 300L     0001     000615 336G     0001     000054 40L
     0001     000060 45L      0001     000006 5L       0001     000076 50L      0001     000123 70L      0001     000135 80L
     0000 R 000014 AM         0000 R 000007 AMAX       0000 R 000011 C          0000 R 000010 D          0000 R 000001 DDT
     0000 R 000003 DELT       0000 R 000012 F          0000 R 000000 FR10       0000 I 000004 I          0000 I 000005 IFLAG
     0000   000030 INJP$      0000 I 000002 ISYMP      0000 I 000013 J          0000 R 000006 S


 00101      1*          SUBROUTINE RKS(DERIV,CNTPL,Y,DY,A,R,T,DEL,N,IFVD,IRKP,NTRY,
 00101      2*         1IERR,DELY,PD,SD,YS,YST,DYST,YSIMP)                              DG00020
 00103      3*          DIMENSION Y(N),DY(N),A(N),R(N),DELY(N),                         DG0  3
 00103      4*         1PD(N),SD(N),YS(N),DYST(N),YST(N),YSIMP(N)                       DG00040
 00104      5*          EXTERNAL  DERIV, CNTRL                                          DG000500
 00104      6*     C    FR10 IS  FIFTH ROOT OF TEN                                      DG00 60
 00105      7*          FR10=1.5848932                                                  DG000700
 00106      8*          IERR=0                                                          DG000800
 00106      9*     C      YS  CONTAINS Y VALUES AT LEFT END POINT OF INTEGRATION INTERVAL DG00090
 00106     10*     C                                                                    DG001000
 00106     11*     C      YSIMP CONTAINS Y FOR SIMPSON'S RULE CHECK  CHECK NOT MADE FOR  #G001100
 00106     12*     C          FIXED STEP MODE    ISYMP IS CONTROL PARAMETER =1,FIXED,2 VARDG00120
 00106     13*     C                                                                    DG001300
 00106     14*     C      IF FIXED STEP SIZE GO ONE INTERVAL OF LENGTH DELT AND RETURN TO DG00140
 00106     15*     C         CNTRL, IF VAR GO TWO INTERVALS BEFORE RETURN TO CONTRL     DG00150
 00106     16*     C                                                                    DG001600
 00106     17*     C      IFVD   = 0    VARIABLE INTERVAL                               DG00170
 00106     18*     C             = 1    FIXED                                           DG001800
 00106     19*     C      IRKP   = 0    CUT INTERVAL ONCE BEFORE REPEAT  (UNDER IFVD=0 ) DG00190
 00106     20*     C             = 1    CUT AS REQUIRED                                 DG0 200
 00106     21*     C      NTRY   = 1    CONTINUE INTEGRATING                            DG002100
 00106     22*     C               2    RETURN FROM RKS                                 DG0 220
 00106     23*     C               3    STEP REPEATED WITH NEW DELT                     DG002300
 00106     24*     C               4    RESTART                                         DG00240
```

```
00106   25*   C      IERR  = 0    NORMAL                                                      DG002500
00106   26*   C            -1     DELT=0, RETURN FROM RKS                                     DG002600
00106   27*   C             1     A(I)+ R(I)*ABS(Y(I)).= 0., RETURN FROM RKS                  DG002700
00107   28*        5  IF(DEL) 20,10,20                                                        DG002800
00112   29*        1  IERR=-1                                                                 DG002900
00113   30*           GO TO 270                                                               DG003000
00114   31*        2  CALL DERIV(Y,DY,T)                                                      DG003100
00115   32*           NTRY=1                                                                  DG003200
00116   33*           CALL CNTRL(Y,DY,DEL,T,NTRY,IFVD)                                        DG003300
00117   34*        2  DDT=DEL                                                                 DG003400
00120   35*           IF(IFVD) 40,30,40                                                       DG003500
00123   36*        3  ISYMP=2                                                                 DG003600
00124   37*           DELT=DEL/2.                                                             DG00370
00125   38*           DO 31 I=1,N                                                             DG003800
00130   39*        3  SD(I)=0.0                                                               DG00390
00132   40*           IFLAG=1                                                                 DG00400
00133   41*           S=1.                                                                    DG004100
00134   42*           GO TO 45                                                                DG004200
00135   43*        4  ISYMP=1                                                                 DG004300
00136   44*           DELT=DEL                                                                DG00440
00137   45*        4  DO 46 I=1,N                                                             DG004500
00142   46*           YST(I)=Y(I)                                                             DG00460
00143   47*        4  DYST(I)=DY(I)                                                           DG0 470
00145   48*        5  DO 60 I=1,N                                                             DG004800
00150   49*           DELY(I)=DELT*DY(I)                                                      DG004900
00151   50*           PD(I)=DELY(I)                                                           DG00500
00152   51*        6  CONTINUE                                                                DG005100
00154   52*           GO TO (80,70),ISYMP                                                     DG005200
00155   53*        7  DO 71 I=1,N                                                             DG005300
00160   54*        7  SD(I)=SD(I)+S*DY(I)                                                     DG005400
00162   55*        8  T=T+DELT/2.                                                             DG005500
00163   56*           DO 85 I=1,N                                                             DG005600
00166   57*           YS(I)=Y(I)                                                              DG00570
00167   58*           Y(I)=YS(I)+DELY(I)/2.                                                   DG005800
00170   59*        8  CONTINUE                                                                DG005900
00172   60*           CALL DERIV(Y,DY,T)                                                      DG006000
00173   61*           DO 90 I=1,N                                                             DG006100
00176   62*           DELY(I)=DELT*DY(I)                                                      DG00620
00177   63*           PD(I)=PD(I)+2.*DELY(I)                                                  DG006300
00200   64*           Y(I)=YS(I)+DELY(I)/2.                                                   DG00640
00201   65*        9  CONTINUE                                                                DG006500
00203   66*           CALL DERIV(Y,DY,T)                                                      DG006600
00204   67*           DO 95 I=1,N                                                             DG00670
00207   68*           DELY(I)=DELT*DY(I)                                                      DG00680
00210   69*           PD(I)=PD(I)+2.*DELY(I)                                                  DG006900
00211   70*           Y(I)=YS(I)+DELY(I)                                                      DG00700
00212   71*        9  CONTINUE                                                                DG007100
00214   72*           T=T+DELT/2.                                                             DG007200
00215   73*           CALL DERIV(Y,DY,T)                                                      DG00730
00216   74*           DO 100 I=1,N                                                            DG007400
00221   75*           DELY(I)=DELT*DY(I)                                                      DG007500
00222   76*           PD(I)=PD(I)+DELY(I)                                                     DG00760
00223   77*           Y(I)=YS(I)+PD(I)/6.                                                     DG00770
00224   78*       10  CONTINUE                                                                DG007800
00226   79*           GO TO (110,120),ISYMP                                                   DG007900
00227   80*       11  NTRY=1                                                                  DG008000
00230   81*           CALL DERIV(Y,DY,T)                                                      DG008100
```

```
00231      82*          CALL CNTRL(Y,DY,DEL,T,NTRY,IFV)                    DG00820
00232      83*          GO TO 300                                          DG008300
00233      84*     12)  GO TO (130,140),IFLAG                              DG008400
00234      85*     13)  S=4.                                               DG008500
00235      86*          IFLAG=2                                            DG008600
00236      87*          CALL DERIV(Y,DY,T)                                 DG008700
00237      88*          GO TO 50                                           D7008800
00240      89*     14)  CALL DERIV(Y,DY,T)                                 DG008900
00241      90*          AMAX =0.0                                          DG009000
00242      91*          DO 180 I=1,N                                       DG009100
00245      92*          SD(I)=SD(I)+DY(I)                                  DG009200
00246      93*          YSIMP(I)=YST(I)+DELT*SD(I)/3.                      DG009300
00247      94*          D    =ABS(Y(I)-YSIMP(I))                           DG00940
00250      95*          C    =A(I)+R(I)*ABS(Y(I))                          DG009500
00251      96*          IF(C     ) 160,150,160                             DG00960
00254      97*     15)  IERR=1                                             DG009700
00255      98*          GO TO 270                                          DG009800
00256      99*     16)  E    =ABS(D    /C    )                             DG009900
00257     100*          AMAX=AMAX1(AMAX,E)                                 DG01000
00260     101*     18)  CONTINUE                                           D7010100
00262     102*          IF(AMAX-1.) 215,215,230                            DG01020
00265     103*     21)  NTRY= 1                                            DG010300
00266     104*          CALL CNTRL(Y,DY,DEL,T,NTRY,IFV)                    DG010400
00267     105*     30)  IF(NTRY-1) 185,185,310                             DG01050
00272     106*     31)  IF(NTRY-2) 270,270,330                             DG01060
00275     107*     33)  IF(NTRY-3) 340,340,5                               DG01070
00300     108*     34)  T=T-DOT                                            DG01080
00301     109*          IF(DEL) 259,10,259                                 DG010900
00304     110*     18)  GO TO (40,190),ISYMP                               DG01100
00305     111*     19)  IF(AMAX-.75) 200,25,220                            DG011100
00310     112*     20)  IF(AMAX-.075) 210,25,25                            DG01120
00313     113*     21)  DEL=DEL*FR10                                       DG011300
00314     114*          GO TO 25                                           DG011400
00315     115*     22)  DEL=DEL/FR10                                       DG011500
00316     116*          GO TO 25                                           DG011600
00317     117*     23)  I =1+ I9KP                                         DG011700
00320     118*          GO TO (240,250),I                                  DG011800
00321     119*     24)  T=T-DEL                                            DG011900
00322     120*          DEL=DEL/FR10                                       DG012000
00323     121*          GO TO  259                                         DG01210
00324     122*     25)  J=1                                                DG012200
00325     123*     25)  AM=AMAX/10.**J                                     DG012300
00326     124*          IF(1.-AM) 255,257,257                              DG01240
00331     125*     25)  J=J+1                                              DG012500
00332     126*          GO TO 251                                          DG01260
00333     127*     257  T=T-DEL                                            DG01270
00334     128*          DEL=DEL/(FR10**J)                                  DG01280
00335     129*     259  DO 245 I=1,N                                       DG01290
00340     130*          DY(I)=DYST(I)                                      DG013000
00341     131*     245  Y(I)=YST(I)                                        DG013100
00343     132*          GO TO 25                                           DG013200
00344     133*     270  RETURN                                             DG013300
00345     134*          END                                                DG013400

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS SDRV
FOR S9A-07/27/72-17:43:39 (,0)

    SUBROUTINE SDRV        ENTRY POINT 000601

    STORAGE USED: CODE(1) 000605; DATA(0) 000066; BLANK COMMON(2) 000000

     COMMON BLOCKS:

     0003   BLKI    00002
     0004   BLKD    00001
     0005   BLKR    00001
     0006   VAB     00000

    EXTERNAL REFERENCES (BLOCK, NAME)

     0007   ETA
     0010   NCDERV
     0011   NDER
     0012   DETA
     0013   DERV
     0014   MIXDER
     0015   NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0000 R 000003 AAA      0004   000003 A1        0004   000004 A2        0004   000005 A3        0004   000006 A4
     0004   000007 A5       0004   000010 A6        0000 R 000005 BBB       0000 R 000037 DA1DN     0000 R 000007 DA1DU
     0000 R 000023 DA1DV    0000 R 000040 DA2DN     0000 R 000010 DA2DU     0000 R 000024 DA2DV     0000 R 000041 DA3DN
     0000 R 000011 DA3DU    0000 R 000025 DA3DV     0000 R 000042 DA4DN     0000 R 000012 DA4DU     0000 R 000026 DA4DV
     0000 R 000043 DA5DN    0000 R 000013 DA5DU     0000 R 000027 DA5DV     0000 R 000044 DA6DN     0000 R 000014 DA6DU
     0000 R 000030 DA6DV    0000 R 000045 DEBDN     0000 R 000004 DEBDTB    0000 R 000015 DEBDU     0000 R 000031 DEBDV
     0000 R 000046 DEN      0000 R 000047 DEPN      0000 R 000017 DEPU      0000 R 000033 DEPV      0012 R 000000 DETA
     0000 R 000016 DEU      0000 R 000032 DEV       0003   000023 DF1DL     0003   000021 DF4DTB    0003 R 000013 DFM
     0003 R 000012 DFMX     0003   000010 DFNX      0005 R 000006 DLDNCB    0005 R 000001 DLDU      0005 R 000002 DLDV
     0000 R 000050 DMDN     0000 R 000020 DMDU      0000 R 000034 DMDV      0000 R 000036 DNCDN     0003 R 000007 DNCDT1
     0000 R 000006 DNCDU    0000 R 000022 DNCDV     0000 R 000001 D2CD2T    0000 R 000002 D2CTNB    0003 R 000016 D2FMLL
     0005 R 000003 D2LDUU   0005 R 000004 D2LDUV    0005 R 000005 D2LDVV    0005 R 000007 D2LNCB    0005 R 000010 D2LUNB
     0005 R 000011 D2LVNL   0003 R 000015 D2MDNC    0003 R 000014 D2MLNC    0000 R 000051 D2MNCN    0000 R 000021 D2MNCU
     0000 R 000035 D2MNCL   0003   000011 D2NDLL    0004 R 000001 D3M2LN    0004 R 000002 D3M2NL    0004 R 000011 EB
     0006 R 000002 ET       0007 R 000000 ETA       0003   000022 E3        0003 R 000002 FM        0003 R 000003 FN
     0003 R 000005 FNC      0003 R 000004 FNCBR     0000   000056 INJP$     0005 R 000000 RLAM      0003 R 000000 THETAB
     0000 R 000000 THETAF   0003 R 000001 THETAS    0004 R 000000 U         0003   000006 V         0003 R 000017 Y1
     0003 R 000020 Y2       0006 R 000000 Z1        0006 R 000001 Z2


00101       1*          SUBROUTINE SDRV(Y,DY,X)
00101       2*    C
00101       3*    C     CALCULATION OF DERIVATIVES FOR THE INTEGRATION SUBROUTINE
00101       4*    C
00103       5*          DIMENSION Y(20),DY(20)
```

```
00104      COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
00104     1           DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DE1DL/
00104     2      BLKO/U,D3M2LN,D3M2NL,A1,A2,A3,A4,A5,A6,EB
00104     3     /BLKR/RLAM,DLDU,DLDV,D2LDUU,D2LDUV,D2LDVV,DLDNCB,
00104     4           D2LNCB,D2LUNB,D2LVNB
00104     5     /VAB/ Z1,Z2,ET
00105      Z1       = FNC
00106      Z2       = FM*ETA(FNC)+FN
00107      ET       = ETA(FNC)
00110      THETAE   = Y(1)
00111      THETAB   = Y(2)
00112      Y1       = Y(1)
00113      Y2       = Y(2)
00114       CALL NCDERV(FNCBR,THETAB,FNC,DNCDT1,D2CD2T,D2CTNB)
00115      CALL MDER(RLAM,FNC,FM,DFMX,DFM,D2FMLL,D2MDNC,D2MLNC,D3M2LN,D3M2NL)
00116      AAA      = THETAE-THETAB
00117      DY(1)    = -U*AAA
00120      EB       = FM*ETA(FNC)+FN
00121      DEBDTB   = 3*THETAB**2*FNCBR*(FM*DETA(FNC)+ETA(FNC)*DFM)
00122      BBB      = 4*THETAB**3/(THETAE**4-THETAS**4)
00123      DY(2)    = -U/(1/AAA+DEBDTB/EB+BBB)
00124      CALL DERV(Y(3),Y(4),Y(5),Y(6),DLDU,D2LDUU,DY(3),DY(4),DY(5),
00124     1           DY(6),0.,1.,DNCDU,DA1DU,DA2DU,DA3DU,DA4DU,DA5DU,
00124     2           DA6DU,DEBDU,DEU,DEPU,DMDU,D2MNCU)
00125      CALL DERV(Y(7),Y(8),Y(9),Y(10),DLDV,D2LDVV,DY(7),DY(8),DY(9),
00125     1           DY(10),0.,0.,DNCDV,DA1DV,DA2DV,DA3DV,DA4DV,DA5DV,
00125     2           DA6DV,DEBDV,DEV,DEPV,DMDV,D2MNCV)
00126      CALL DERV(Y(11),Y(12),Y(13),Y(14),DLDNCB,D2LNCB,DY(11),DY(12),
00126     1           DY(13),DY(14),1.,0.,DNCDN,DA1DN,DA2DN,DA3DN,DA4DN,
00126     2           DA5DN,DA6DN,DEBDN,DEN,DEPN,DMDN,D2MNCN)
00127      CALL MIXDER(Y(4),Y(8),Y(7),Y(16),Y(15),DNCDU,DNCDV,DY(4),DA1DU,
00127     1             DA1DV,DA2DU,DA2DV,DA3DU,DA3DV,DA4DV,DA5DU,DA5DV,
00127     2             DA6DU,DA6DV,DLDU,DLDV,D2LDUV,DEBDV,DEBDU,DEU,DEV,
00127     3             DEPU,DEPV,DMDU,DMDV,D2MNCU,D2MNCV,1.,0.,DY(16),
00127     4             DY(15))
00130      CALL MIXDER(Y(4),Y(12),Y(11),Y(18),Y(17),DNCDU,DNCDN,DY(4),
00130     1             DA1DU,DA1DN,DA2DU,DA2DN,DA3DU,DA3DN,DA4DN,DA5DU,
00130     2             DA5DN,DA6DU,DA6DN,DLDU,DLDNCB,D2LUNB,DEBDN,DEBDU,
00130     3             DEU,DEN,DEPU,DEPN,DMDU,DMDN,D2MNCU,D2MNCN,1.,1.,
00130     4             DY(18),DY(17))
00131      CALL MIXDER(Y(8),Y(12),Y(11),Y(20),Y(19),DNCDV,DNCDN,DY(8),
00131     1             DA1DV,DA1DN,DA2DV,DA2DN,DA3DV,DA3DN,DA4DN,DA5DV,
00131     2             DA5DN,DA6DV,DA6DN,DLDV,DLDNCB,D2LVNB,DEBDN,DEBDV,
00131     3             DEV,DEN,DEPV,DEPN,DMDV,DMDN,D2MNCV,D2MNCN,0.,1.,
00131     4             DY(20),DY(19))
00132       RETURN
00133      END

     END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS SCNTL
FOR S9A-07/27/72-17:43:50 (,0)

   SUBROUTINE SCNTL      ENTRY POINT 000147

   STORAGE USED: CODE(1) 000174; DATA(0) 000014; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   MCT     000005
    0004   VAB     000003
    0005   BLKI    000024

   EXTERNAL REFERENCES (BLOCK, NAME)

    0006   NWDU$
    0007   NI02$
    0010   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000   000002 2000F      0001   000054 5L        0001   000061 50L       0001   000135 60L       0003 R 000000 DLMT
    0000 R 000000 DXSTR      0003 R 000003 DXWRT     0004 R 000002 ET        0003 I 000001 ICNT      0000   000006 INJP$
    0000 I 000001 LCNT       0003 I 000004 LSTEP     0005 R 000000 VAL       0003 R 000002 XWRT      0004 R 000000 Z1
    0004 R 000001 Z2


00101      1*           SUBROUTINE SCNTL(Y,DY,DX,X,NTRY,IFVD)
00101      2*     C
00101      3*     C     INTEGRATION CONTROL PROGRAM
00101      4*     C
00103      5*           DIMENSION Y(20),DY(20)
00104      6*           COMMON /MCT/ DLMT,ICNT,XWRT,DXWRT,LSTEP
00104      7*          1        /VAB/ Z1,Z2,ET
00104      8*          2        /BLKI/VAL(20)
00105      9*           ICNT    = ICNT+1
00106     10*           IF(DX.GE.DXWRT.AND.ICNT.GT.1) LSTEP = 1
00110     11*           IF(ABS(X-XWRT).LT.DLMT) GO TO 50
00112     12*           IF(XWRT.GT.X) GO TO 5
00112     13*     C
00114     14*           DXSTR   = DX
00115     15*           DX      = DX+XWRT-X
00116     16*           LCNT    = 1
00117     17*           NTRY    = 3
00120     18*           RETURN
00120     19*     C
00121     20*         5 NTRY    = 1
00122     21*           RETURN
00122     22*     C
00123     23*        50 WRITE(6,2000) Y(1),Y(2),Z1,Z2,ET,VAL(3),VAL(4),X
00135     24*      2000 FORMAT (8F15.7)
```

```
00136        IF(LCNT.EQ.1) DX = DXSTR
00140        LCNT     = 0
00141        IF(ABS(1.0-XWRT).LE.DLMT) GO TO 60
00143        XWRT     = XWRT+DXWRT
00144        NTRY     = 1
00145        IF(LSTEP.EQ.0) RETURN
00147        DX       = DXWRT
00150        IFVD     = 1
00151        RETURN
00151  C
00152     60 NTRY     = 2
00153        RETURN
00154        END

   END OF COMPILATION:        NO DIAGNOSTICS.
```

```
@FOR,IS BNDCND
FOR S9A-07/27/72-17:43:54 (,0)

 SUBROUTINE BNDCND     ENTRY POINT 000350

 STORAGE USED: CODE(1) 000411; DATA(0) 000065; BLANK COMMON(2) 000000

  COMMON BLOCKS:

  0003    BLK3    000002

 EXTERNAL REFERENCES (BLOCK, NAME)

  0004    FMINV
  0005    NERR3$

 STORAGE ASSIGNMENT   (BLOCK, TYPE, RELATIVE LOCATION, NAME)

  0001    000115 600L       0001    000130 601L       0001    000134 602L       0001    000137 603L       0001    000167 610L
  0001    000202 611L       0001    000206 612L       0001    000211 613L       0001    000241 620L       0001    000254 621L
  0001    000260 622L       0001    000263 623L       0001    000313 630L       0001    000040 700L       0000 R 000000 AN
  0003    000000 DNCBD      0003    000001 DNCBDV     0000 I 000017 I         0000    000044 INJP$      0000 I 000020 J
  0000 I 000021 K         0000 R 000014 XN        0000 R 000022 Z1        0000 R 000023 Z2        0000 R 000024 Z3


00101      1*          SUBROUTINE BNDCND(Y,U,V,FNCBR,XM)
00101      2*     C
00101      3*     C    BNDCND INSURES U,V, AND NCBAR ARE WITHIN THEIR BOUNDARY CONDITIONS
00101      4*     C
00103      5*          DIMENSION AN(3,4),XN(3),XM(3),Y(20)
00104      6*          COMMON /BLKB/DNCBDU,DNCBDV
00105      7*          I        = 0
00106      8*          J        = 0
00107      9*          K        = 0
00110     10*          Z1       = U
00111     11*          Z2       = V
00112     12*          Z3       = FNCBR
00113     13*          U        = Z1+XM(1)
00114     14*          V        = Z2+XM(2)
00115     15*          FNCBR    = Z3+XM(3)
00116     16*          AN(1,1) =  Y(5)
00117     17*          AN(1,2) =  Y(15)
00120     18*          AN(1,3) =  Y(17)
00121     19*          AN(2,1) =  Y(15)
00122     20*          AN(2,2) =  Y(9)
00123     21*          AN(2,3) =  Y(19)
00124     22*          AN(3,1) = Y(17)
00125     23*          AN(3,2) = Y(19)
00126     24*          AN(3,3) = Y(13)
00127     25*      700 IF (U.LT.4.00E-06.OR.U.GT.250.1) GO TO 600
00131     26*          IF (V.LT.4.00E-05.OR.V.GT.50.01) GO TO 610
```

```
00133                IF (FNCBR.LT.0.00009.OR.FNCBR.GT.4.01) GO TO 620
00135                RETURN
00136         60:    IF (U.LE.5.0E-06) GO TO 601
00140                IF (U.GE.250.) GO TO 602
00142                GO TO 610
00143         60     XN(1)    = 5.0E-06-Z1
00144                GO TO 603
00145         60:    XN(1)    = 250.-Z1
00146         60:    AN(1,1) = 1.
00147                AN(1,2) = 0.
00150                AN(1,3) = 0.
00151                I        = 1
00152                IF(I.EQ.0) XN(1) = -Y(3)
00154                IF(J.EQ.0) XN(2) = -Y(7)
00156                IF(K.EQ.0) XN(3) = -Y(11)
00160                CALL FMINV(AN,XN,3,4)
00161         61     IF (V.LE.5.0E-05) GO TO 611
00163                IF (V.GE.50.) GO TO 612
00165                GO TO 620
00166         61:    XN(2)    = 5.0E-05-Z2
00167                GO TO 613
00170         61;    XN(2)    = 50.-Z2
00171         61     AN(2,1) = 0.
00172                AN(2,2) = 1.
00173                AN(2,3) = 0.
00174                J        = 1
00175                IF(I.EQ.0) XN(1) = -Y(3)
00177                IF(J.EQ.0) XN(2) = -Y(7)
00201                IF(K.EQ.0) XN(3) = -Y(11)
00203                CALL FMINV(AN,XN,3,4)
00204         62.    IF (FNCBR.LE.1.0E-04) GO TO 62
00206                IF (FNCBR.GE.4.00) GO TO 622
00210                GO TO 630
00211         62     XN(3)    = 1.0E-04-Z3
00212                GO TO 623
00213         62     XN(3)    = 4.00-Z3
00214         62     AN(3,1) = 0.
00215                AN(3,2) = 0.
00216                AN(3,3) = 1.
00217                K        = 1
00220                IF(I.EQ.0) XN(1) = -Y(3)
00222                IF(J.EQ.0) XN(2) = -Y(7)
00224                IF(K.EQ.0) XN(3) = -Y(11)
00226                CALL FMINV(AN,XN,3,4)
00227         63     XM(1)    = XN(1)
00230                XM(2)    = XN(2)
00231                XM(3)    = XN(3)
00232                U        = Z1+XM(1)
00233                V        = Z2+XM(2)
00234                FNCBR    = Z3+XM(3)
00235         66     GO TO 700
00236                END

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
2   @FOR,IS RESBND
    FOR S9A-07/27/72-17:44:04 (,0)

      SUBROUTINE RESBND    ENTRY POINT 000203

      STORAGE USED: CODE( ),000232; DATA(0) 000053; BLANK COMMON(2) 000000

      EXTERNAL REFERENCES (BLOCK, NAME)

       0003   SQRT
       0004   NWDU$
       0005   NIO2$
       0006   NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0001   000026 1L         0001   000126 2L         0000   000007 2000F      0001   000144 3L         0001   000161 4L
       0000 R 000003 A          0000 R 000004 B          0000 R 000005 C          0000 R 000006 EPS        0000   000037 INJP$
       0000 R 000000 U2         0000 R 000001 V2         0000 R 000002 Z


   00101      1*             SUBROUTINE RESBND(U,V,XM,ASTAR)
   00101      2*        C
   00101      3*        C    RESBND CHECKS THE BOUNDARY OF LAMBDA AND KEEPS ASTAR
   00101      4*        C    GREATER THAN U*V
   00101      5*        C
   00103      6*             DIMENSION XM(3)
   00104      7*             U2       = U-XM(1)
   00105      8*             V2       = V-XM(2)
   00106      9*             Z        = ASTAR-U*V
   00107     10*             IF(Z.LT.2.*U*V*1.0E-06) GO TO 1
   00111     11*             RETURN
   00111     12*        C
   00112     13*           1 A        = XM(1)*XM(2)
   00113     14*             B        = U2*XM(2)+V2*XM(1)
   00114     15*             C        = U2*V2-0.99*ASTAR
   00115     16*             IF(ABS(C).LE.1.0E-05.AND.XM(2).LT.0.) GO TO 2
   00117     17*             IF(ABS(C).LE.1.0E-05.AND.XM(1).LT.0.) GO TO 3
   00117     18*        C
   00121     19*             EPS      = (-B+SQRT(B**2.-4.*A*C))/2./A
   00122     20*             XM(1)    = EPS*XM(1)
   00123     21*             XM(2)    = EPS*XM(2)
   00124     22*             U        = U2+XM(1)
   00125     23*             V        = V2+XM(2)
   00126     24*             GO TO 4
   00127     25*           2 EPS      = (0.99*ASTAR-U2*V)/(V*XM(1))
   00130     26*             XM(1)    = EPS*XM(1)
   00131     27*             U        = U2+XM(1)
   00132     28*             GO TO 4
   00133     29*           3 EPS      = (0.99*ASTAR-U*V2)/(U*XM(2))
   00134     30*             XM(2)    = EPS*XM(2)
```

```
00135      31*              V        = V2+XM(2)
00135      32*        C
00136      33*            4 WRITE(6,2000) XM(1),XM(2),EPS
00143      34*         2000 FORMAT (//5X'EPSILON CALCULATED        DELTAU ='E14.6/
00143      35*            1           31X'    DELTAV ='E14.6/31X'EPSILON =     'E14.6)
00143      36*        C
00144      37*              RETURN
00145      38*              END

       END OF COMPILATION:        NO  DIAGNOSTICS.
```


158669

```
02   @FOR,IS REST
     FOR S9A-07/27/72-17:44:10 (,0)

     SUBROUTINE REST      ENTRY POINT 000066

     STORAGE USED: CODE(1) 000120; DATA(0) 000023; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0000 R 000000 A1        0000   000003 INJP$


 00101       1*            SUBROUTINE REST(U,V,ASTAR,RLAM,DLDU,DLDV,D2LDUU,D2LDUV,D2LDVV,
 00101       2*           1DLDNCB,D2LNCB,D2LUNB,D2LVNB)
 00101       3*       C
 00101       4*       C    CONSTANT AREA RESTRAINT
 00101       5*       C
 00103       6*            A1      = ASTAR-U*V
 00104       7*            RLAM    = 2*U*V/A1
 00105       8*            DLDU    = 2.*ASTAR*V/A1**2.
 00106       9*            DLDV    = 2.*ASTAR*U/A1**2.
 00107      10*            D2LDUU  = 4.*ASTAR*V**2./A1**3.
 00110      11*            D2LDUV  = 2.*ASTAR*(ASTAR+U*V)/A1**3.
 00111      12*            D2LDVV  = 4.*ASTAR*U**2./A1**3.
 00112      13*            DLDNCB  = 0.
 00113      14*            D2LNCB  = 0.
 00114      15*            D2LUNB  = 0.
 00115      16*            D2LVNB  = 0.
 00116      17*            RETURN
 00117      18*            END

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS NDERV
FOR S9A-07/27/72-17:44:13 (,0)

   SUBROUTINE NDERV      ENTRY POINT 000056

   STORAGE USED: CODE(1) 000070; DATA(0) 000015; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   SQRT
    0004   ASIN
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000 R 000000 C4        0000 R 000001 C5        0000 R 000002 C6        0000    000006 INJP$

00101       1*            SUBROUTINE NDERV(RLAM,FN,DFNX,D2NDLL)
00101       2*      C
00101       3*      C     N AND ITS DERIVATIVES
00101       4*      C
00103       5*            C4       = RLAM+1.0
00104       6*            C5       = RLAM+2.0
00105       7*            C6       = SQRT(RLAM*C5)
00106       8*            FN       = (1.5707963+C5-C6-ASIN(1.0/C4))/RLAM
00107       9*            DFNX     = -(FN+C6/C4-1.0)/RLAM
00110      10*            D2NDLL   = -(2.0*DFNX+1.0/(C4*C4*C6))/RLAM
00111      11*            RETURN
00112      12*            END

     END OF COMPILATION:        NO  DIAGNOSTICS.
```

158671

```
@FOR,IS MDER
FOR S9A-07/27/72-17:44:16 (,0)

    SUBROUTINE MDER        ENTRY POINT 000251

    STORAGE USED: CODE(1) 000304; DATA(0) 000066; BLANK COMMON(2) 000000

    EXTERNAL REFERENCES (BLOCK, NAME)

     0003   SQRT
     0004   ASIN
     0005   NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0000 R 000011 CFN     0000 R 000000 C1      0000 R 000017 C10     0000 R 000020 C11     0000 R 000022 C12
     0000 R 000001 C2      0000 R 000003 C3      0000 R 000002 C4      0000 R 000004 C5      0000 R 000005 C6
     0000 R 000006 C7      0000 R 000007 C8      0000 R 000010 C9      0000 R 000015 DCFDNC  0000 R 000013 DCFNDL
     0000 R 000014 DFFSDL  0000 R 000016 D2CFLL  0000 R 000025 D2CFNC  0000 R 000026 D2CLNC  0000 R 000023 D2C2LL
     0000 R 000021 D2C3LL  0000 R 000024 D2FSLL  0000 R 000027 D3C2LN  0000 R 000030 D3C2NL  0000 R 000012 FFS
     0000   000040 INJP$


 00101      1*          SUBROUTINE MDER(RLAM,FNC,FM,DFMX,DFM,D2FMLL,D2MDNC,D2MLNC,D3M2LN,
 00101      2*         1 3M2NL)
 00101      3*     C
 00101      4*     C    AND ITS DERIVATIVES
 00101      5*     C
 00103      6*          C1      = 0.1460*FNC-0.02866
 00104      7*          C2      = SQRT(RLAM*(2.+RLAM))
 00105      8*          C4      = 1.0/(RLAM+1.0)
 00106      9*          C3      = ASIN(C4)
 00107     10*          C5      = 1/(C4*C2)
 00110     11*          C6      = (C4**2)/SQRT(1-C4**2)
 00111     12*          C7      = 1/(RLAM**2)
 00112     13*          C8      = FNC/RLAM
 00113     14*          C9      = 2.*C7*C8
 00114     15*          CFN     = 1.0-C1*C8
 00115     16*          FFS     = (C2+C3-1.57079635)/RLAM
 00116     17*          FM      = CFN+FFS
 00117     18*          DCFNDL  = C1*C7*FNC
 00120     19*          DFFSDL  = (C5-C6-FFS)/RLAM
 00121     20*          DFMX    = CFN*DFFSDL+FFS*DCFNDL
 00122     21*          DCFDNC  = -(0.1460*FNC+C1)/RLAM
 00123     22*          DFM     = FFS*DCFDNC
 00124     23*          D2CFLL  = -C1*C9
 00125     24*          C10     = 1-C4**2
 00126     25*          C11     = +1.+0.5/C10*C4**2
 00127     26*          D2C3LL  = 2*(C4**3)*C11/SQRT(C10)
 00130     27*          C12     = 1/C2-C2*(C4**2)
 00131     28*          D2C2LL  = -C12/(C4*C2)**2
```

```
00132    29*        D2FSLL  = (D2C2LL+D2C3LL-2.*DFFSDL)/RLAM
00133    30*        D2FMLL  = CFN*D2FSLL+2.*DFFSDL*DCFNDL+FFS*D2CFLL
00134    31*        D2CFNC  = -0.2920/RLAM
00135    32*        D2MDNC  = FFS*D2CFNC
00136    33*        D2CLNC  = (0.146*FNC+C1)*C7
00137    34*        D2MLNC  = DCFDNC*DFFSDL+D2CLNC*FFS
00140    35*        D3C2LN  = -2.*D2CLNC/RLAM
00141    36*        D3M2LN  = DCFDNC*D2FSLL+2.*DFFSDL*D2CLNC+FFS*D3C2LN
00142    37*        D3C2NL  = 0.2920*C7
00143    38*        D3M2NL  = FFS*D3C2NL+DFFSDL*D2CFNC
00144    39*        RETURN
00145    40*        END

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS NCDERV
FOR S9A-07/27/72-17:44:19 (,0)

   SUBROUTINE NCDERV     ENTRY POINT 000031

   STORAGE USED: CODE(1) 000044; DATA(0) 000013; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003    NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000   C00004 INJP$      0000 R 000000 T2          0000 R 000001 T3


00101       1*              SUBROUTINE NCDERV(FNCBR,THETAB,FNC,DNCDT1,D2CD2T,D2CTNB)
00101       2*       C
00101       3*       C      NC AND ITS DERIVATIVES
00101       4*       C
00103       5*              T2       = THETAB**2
00104       6*              T3       = T2*THETAB
00105       7*              FNC      = FNCBR*T3
00106       8*              DNCDT1   = 3.*FNCBR*T2
00107       9*              D2CD2T   = 6.*FNCBR*THETAB
00110      10*              D2CTNB   = 3.*T2
00111      11*              RETURN
00112      12*              END

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS MIXDER
FOR S9A-07/27/72-17:44:49 (,0)

SUBROUTINE MIXDER     ENTRY POINT 000347

STORAGE USED: CODE(1) 000454; DATA(0) 000104; BLANK COMMON(2) 000000

 COMMON BLOCKS:

 0003   BLKI    000024
 0004   BLKD    000012

EXTERNAL REFERENCES (BLOCK, NAME)

 0005   DDETA
 0006   DETA
 0007   DDDETA
 0010   ETA
 0011   NERR3$

STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

 0004 R 000003 A1        0004 R 000004 A2        0004 R 000005 A3        0004 R 000006 A4        0004 R 000007 A5
 0004 R 000010 A6        0007 R 000000 DDDETA    0005 R 000000 DDETA     0006 R 000000 DETA      0003   000023 DE1DL
 0003   000021 DE4DT     0003 R 000013 DFM       0003 R 000012 DFMX      0003 R 000010 DFNX      0003   000007 DNCDT1
 0000 R 000006 DNCDX     0000 R 000003 D2A1XY    0000 R 000020 D2A2XY    0000 R 000021 D2A3XY    0000 R 000022 D2A4XY
 0000 R 000015 D2A5X     0000 R 000004 D2A6XY    0000 R 000017 D2EBXY    0000 R 000012 D2EPXY    0000 R 000011 D2EXY
 0003 R 000016 D2FML     0000 R 000013 D2MDLY    0003 R 000015 D2MDNC    0000 R 000014 D2MDXY    0003 R 000014 D2MLNC
 0003 R 000011 D2NDL     0000 R 000016 D2NDXY    0000 R 000007 D3MNLY    0000 R 000010 D3MNXY    0004 R 000001 D3M2LN
 0004 R 000002 D3M2N     0000 R 000005 D3M2NY    0000 R 000011 EB        0010 R 000000 ETA       0003   000022 E3
 0003 R 000002 FM        0003   000003 FN        0003 R 000005 FNC       0003 R 000004 FNCBR     0000   000026 INJP$
 0003   000000 THETAB    0003   000001 THETAS    0004 R 000000 U         0003   000006 V         0000 R 000000 Y
 0003   000017 Y1        0003 R 000020 Y2        0000 R 000002 Z

00101      1*          SUBROUTINE MIXDER(DTBDX,DTBDY,DTFDY,D2TBXY,D2TFXY,DNCDX,DNCDY,
00101      2*         1                    D2TBYE,DA1DX,DA1DY,DA2DX,DA2DY,DA3DX,DA3DY,
00101      3*         2                    DA4DY,DA5DX,DA5DY,DA6DX,DA6DY,DLDX,DLDY,
00101      4*         3                    D2LDXY,DEBDY,DEBDX,DEX,DEY,DEPX,DEPY,DMDX,
00101      5*         4                    DMDY,D2MNCX,D2MNCY,B,C,D3TXYE,DTFDXY)
00101      6*    C
00101      7*    C     FOR MIXED DERIVATIVES OF U AND V      SET B=1. AND C= 0.
00101      8*    C     FOR MIXED DERIVATIVES OF U AND FNCBR SET B=1. AND C= 1.
00101      9*    C     FOR MIXED DERIVATIVES OF V AND FNCBR SET B=0. AND C= 1.
00101     10*    C
00103     11*          DIMENSION Y(2)
00104     12*          COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
00104     13*         1          DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DE1DL/
00104     14*         2     BLKD/U,D3M2LN,D3M2NL,A1,A2,A3,A4,A5,A6,EB
00105     15*          Y(2)      = Y2
00106     16*          Z         = FNC
```

```
00107    17*       DTFDXY  = U*(D2TBXY-D2TFXY)+B*(DTBDY-DTFDY)
00110    18*       D2A1XY  = 2*DA1DX*DA1DY/A1-A1**2*(D2TFXY-D2TBXY)
00111    19*       D2A6XY  = 6*FNCBR*DTBDX*DTBDY+6*Y(2)*FNCBR*D2TBXY+C*6*Y(2)*DTBDX
00112    20*       D3M2NY  = DLDY*D3M2NL
00113    21*       DNCDXY  = 6*FNCBR*Y(2)*DTBDX*DTBDY+3*FNCBR*(Y(2)**2)*D2TBXY+
00113    22*      1           C*3*(Y(2)**2)*DTBDX
00114    23*       D3MNLY  = DLDY+D3M2LN+DNCDY*D3M2NL
00115    24*       D3MNXY  = D3MNLY*DLDX+D2MLNC*D2LDXY+D3M2NY*DNCDX+D2MDNC*DNCDXY
00116    25*       D2EXY   = DDETA(Z)*DNCDX*DNCDY+DETA(Z)*DNCDXY
00117    26*       D2EPXY  = DDDETA(Z)*DNCDX*DNCDY+DDETA(Z)*DNCDXY
00120    27*       D2MDLY  = DLDY+D2FMLL+DNCDY*D2MLNC
00121    28*       D2MDXY  = DLDX*D2MDLY+DFMX*D2LDXY+D2MNCY*DNCDX+DFM*DNCDXY
00122    29*       D2A5XY  = FM*D2EPXY+DEPX*DMDY+DEPY*DMDX+DETA(Z)*D2MDXY+DEY*D2MNCX+
00122    30*      1           ETA(Z)*D3MNXY+DEX*D2MNCY+DFM*D2EXY
00123    31*       D2NDXY  = DFNX+D2LDYY+DLDX*DLDY+D2NDLL
00124    32*       D2EBXY  = FM*D2EXY+DMDY*DEX+ETA(Z)*D2MDXY+DEY*DMDX+D2NDXY
00125    33*       D2A2XY  = -(DEBDY*DA2DX+DEBDX*DA2DY+A2*D2EBXY-(A6*D2A5XY+
00125    34*      1           DA6DY*DA5DX+DA5DY*DA6DX+A5*D2A6XY))/EB
00126    35*       D2A3XY  = DA3DX*DA3DY/A3+D2TBXY*DA3DX/DTBDX+
00126    36*      1           A3*DTBDX*(-3*DTBDY/Y(2)**2-DA3DY)
00127    37*       D2A4XY  = D2A1XY+D2A2XY+D2A3XY
00130    38*       D3TXYE  = U*D2A4XY/A4**2-DA4DY*(2*A4*D2TBXE+B)/A4**2
00131    39*       RETURN
00132    40*       END

        END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS DERV
FOR S9A-07/27/72-17:44:54 (,0)

      SUBROUTINE DERV      ENTRY POINT 000604

      STORAGE USED: CODE(1) 000721; DATA(0) 000130; BLANK COMMON(2) 000000

       COMMON BLOCKS:

       0003   BLKI   000024
       0004   BLKD   000012

      EXTERNAL REFERENCES (BLOCK, NAME)

       0005   ETA
       0006   DETA
       0007   DDETA
       0010   DDDETA
       0011   NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0004 R 000003 A1        0004 R 000004 A2        0004 R 000005 A3        0004 R 000006 A4        0004 R 000007 A5
       0004 R 000010 A6        0000 R 000051 A7        0000 R 000063 DA7DU     0010 R 000000 DDDETA    0007 R 000000 DDETA
       0006 R 000000 DETA      0003   000023 DE1DL     0003   000021 DE4DTB    0003 R 000013 DFM       0003 R 000012 DFMX
       0003 R 000010 DFNX      0003   000007 DNCDT1    0000 R 000052 DNDU      0000 R 000024 DY        0000 R 000053 D2A1UU
       0000 R 000066 D2A2UU    0000 R 000067 D2A3UU    0000 R 000070 D2A4UU    0000 R 000061 D2A5UU    0000 R 000062 D2A6UU
       0000 R 000065 D2EBUU    0000 R 000055 D2EPUU    0000 R 000060 D2EUU     0003 R 000016 D2FMLL    0003 R 000015 D2MDNC
       0003 R 000014 D2MLNC    0000 R 000056 D2MUU     0000 R 000054 D2NCUU    0003 R 000011 D2NDLL    0000 R 000064 D2NUU
       0000 R 000057 D3MNUU    0004 R 000001 D3M2LN    0004 R 000002 D3M2NL    0004 R 000011 EB        0005 R 000000 ETA
       0003   000022 E3        0003 R 000002 FM        0003 R 000003 FN        0003 R 000005 FNC       0003 R 000004 FNCBR
       0000   000077 INJP$     0003   000000 THETAB    0003 R 000001 THETAS    0004 R 000000 U         0003   000006 V
       0000 R 000000 Y         0003 R 000017 Y1        0003 R 000020 Y2        0000 R 000050 Z


 00101      1*            SUBROUTINE DERV(Y3,Y4,Y5,Y6,DLDU,D2LDUU,DY3,DY4,DY5,DY6,B,C,
 00101      2*           1           DNCDU,DA1DU,DA2DU,DA3DU,DA4DU,DA5DU,DA6DU,
 00101      3*           2           DEPDU,DEU,DEPU,DMDU,D2MNCU)
 00101      4*      C
 00101      5*      C     DERIVATIVES
 00101      6*      C
 00101      7*      C     FOR FIRST AND SECOND DERIVATIVES OF      U SET C=1 AND B=0
 00101      8*      C     FOR FIRST AND SECOND DERIVATIVES OF      V SET C=0 AND B=0
 00101      9*      C     FOR FIRST AND SECOND DERIVATIVES OF FNCBR SET C=0 AND B=1
 00101     10*      C
 00103     11*            DIMENSION Y(20),DY(20)
 00104     12*            COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
 00104     13*           1      DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DE1DL/
 00104     14*           2      BLKD/U,D3M2LN,D3M2NL,A1,A2,A3,A4,A5,A6,EB
 00105     15*            Y(1)     = Y1
 00106     16*            Y(2)     = Y2
```

```
00107     17*        Y(3)     = Y3
00110     18*        Y(4)     = Y4
00111     19*        Y(5)     = Y5
00112     20*        Y(6)     = Y6
00113     21*        Z        = FNC
00114     22*        E3       = FM*ETA(Z)+FN
00115     23*        A1       = 1./(Y(1)-Y(2))
00116     24*        A5       = FM*DETA(Z)+ETA(Z)*DFM
00117     25*        A6       = -3*Y(2)**2*FNCBR
00120     26*        A2       = A6*A5/EB
00121     27*        A3       = 4*Y(2)**3/(Y(2)**4-THETAS**4)
00122     28*        A4       = A1+A2+A3
00123     29*        DA1DU    = -A1**2*(Y(3)-Y(4))
00124     30*        DNCDU    = A6*Y(4)+B*Y(2)**3
00125     31*        DEPU     = DDETA(Z)*DNCDU
00126     32*        DMDU     = DFMX*DLDU+DFM*DNCDU
00127     33*        D2MNCU   = D2MLNC*DLDU+D2MDNC*DNCDU
00130     34*        DEU      = DETA(Z)*DNCDU
00131     35*        DA5DU    = FM*DEPU+DETA(Z)*DMDU+ETA(Z)*D2MNCU+DFM*DEU
00132     36*        DA6DU    = 6*Y(2)*FNCBR*Y(4)+B*3*Y(2)**2
00133     37*        A7       = A6*DA5DU+A5*DA6DU
00134     38*        DNDU     = DFNX*DLDU
00135     39*        DEBDU    = FM*DEU+ETA(Z)*DMDU+DNDU
00136     40*        DA2DU    = (A7-A2*DEBDU)/EB
00137     41*        DA3DU    = A3*Y(4)*(3/Y(2)-A3)
00140     42*        DA4DU    = DA1DU+DA2DU+DA3DU
00141     43*        DY(4)    = -C/A4+U/A4**2*DA4DU
00142     44*        D2A1UU   = -A1**2*(Y(5)-Y(6))-2*A1*DA1DU*(Y(3)-Y(4))
00143     45*        D2NCUU   = A6*Y(5)+DA6DU*Y(4)+B*3*Y(4)*Y(2)**2
00144     46*        D2EPUU   = DDETA(Z)*D2NCUU+DNCDU**2*DDDETA(Z)
00145     47*        D2MUU    = DFMX*D2LDUU+DLDU*(D2FMLL*DLDU+D2MLNC*DNCDU)+DFM*D2NCUU
00145     48*      1            +DNCDU*(D2MLNC*DLDU+D2MDNC*DNCDU)
00146     49*        D3MNUU   = D2MLNC*D2LDUU+DLDU*(D3M2LN*DLDU+D3M2NL*DNCDU)
00146     50*      1            +D2MDNC*D2NCUU+DNCDU*D3M2NL*DLDU
00147     51*        D2EUU    = DETA(Z)*D2NCUU+DDETA(Z)*DNCDU**2
00150     52*        D2A5UU   = FM*D2EPUU+2*DEPU*DMDU+DETA(Z)*D2MUU+ETA(Z)*D3MNUU
00150     53*      1            +2*DEU*D2MNCU+DFM*D2EUU
00151     54*        D2A6U    = 6*Y(2)*FNCBR*Y(6)+6*FNCBR*Y(4)**2+B*12*Y(2)*Y(4)
00152     55*        DA7DU    = A6*D2A5UU+DA5DU*DA6DU+A5*D2A6U+DA6DU*DA5DU
00153     56*        D2NUU    = DFNX*D2LDUU+D2NDLL*DLDU**2
00154     57*        D2EBUU   = FM*D2EUU+2*DEU*DMDU+ETA(Z)*D2MUU+D2NUU
00155     58*        D2A2UU   = (DA7DU-A2*D2EBUU-2*DEBDU*DA2DU)/EB
00156     59*        D2A3UU   = A3*((3/Y(2)-A3)*Y(6)+Y(4)*(-3/Y(2)**2*Y(4)-DA3DU)
00156     60*      1            +(1/A3*DA3DU)**2)
00157     61*        D2A4UU   = D2A1UU+D2A2UU+D2A3UU
00160     62*        DY(6)    = -2/A4*DY(4)*DA4DU+U/A4**2*D2A4UU
00161     63*        DY(3)    = U*(Y(4)-Y(3))+(Y(2)-Y(1))*C
00162     64*        DY(5)    = U*(Y(6)-Y(5))+2*(Y(4)-Y(3))*C
00163     65*        DY3      = DY(3)
00164     66*        DY4      = DY(4)
00165     67*        DY5      = DY(5)
00166     68*        DY6      = DY(6)
00167     69*        RETURN
00170     70*        END

      END OF COMPILATION:        NO DIAGNOSTICS.
```

```
@FOR,IS INTIAL
FOR S9A-07/27/72-17:44:59 (,0)

   SUBROUTINE INTIAL     ENTRY POINT 000470

   STORAGE USED: CODE(1) 000522; DATA(0) 000123; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   BLKI    00002

   EXTERNAL REFERENCES (BLOCK, NAME)

    0004   ETA
    0005   DETA
    0006   DDETA
    0007   NERR3$

   STORAGE ASSIGNMENT (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000 R 000037 A1        0000 R 000040 A2        0000 R 000041 A3        0000 R 000053 B1        0000 R 000054 B2
    0006 R 000000 DDETA     0000 R 000051 DDETDU    0005 R 000000 DETA      0000 R 000046 DETADU    0003 R 000023 DE1DL
    0000 R 000032 DE1DN     0000 R 000027 DE1DTB    0000 R 000047 DE1DU     0000 R 000030 DE2DTB    0000 R 000034 DE2DU
    0000 R 000031 DE3DT     0000 R 000035 DE3DU     0003 R 000021 DE4DTB    0000 R 000033 DE4DU     0000 R 000044 DECNDU
    0003 R 000013 DFM       0000 R 000045 DFMDU     0003 R 000012 DFMX      0000 R 000043 DFNDU     0003 R 000010 DFNX
    0003 R 000007 DNCDT     0000 R 000061 D1        0000 R 000062 D2        0000 R 000065 D2E1CB    0000 R 000060 D2E1LU
    0000 R 000055 D2E1T     0000 R 000036 D2E2TU    0000 R 000042 D2E3TU    0000 R 000066 D2E4CB    0000 R 000071 D2E4DU
    0003 R 000016 D2FML     0000 R 000056 D2MDLU    0003 R 000015 D2MDNC    0003 R 000014 D2MLNC    0000 R 000052 D2MNCU
    0000 R 000067 D2NCN     0000 R 000050 D2NCTU    0003 R 000011 D2NDLL    0000 R 000057 D2NDLU    0000 R 000063 D3
    0000 R 000064 D4        0004 R 000000 ETA       0000 R 000025 E1        0000 R 000026 E2        0003 R 000022 E3
    0003 R 000002 FM        0003 R 000003 FN        0003 R 000005 FNC       0003 R 000004 FNCBR     0000   000100 INJP$
    0000 R 000070 TERMN     0003 R 000000 THETAB    0003 R 000001 THETAS    0003 R 000006 V         0000 R 000000 Y
    0003 R 000017 Y1        0003 R 000020 Y2        0000 R 000024 Z

 00101       1*            SUBROUTINE INTIAL(DLDU,D2LDUU,Y3,Y4,Y5,Y6,B,C,D2E4TU,D2E4LU)
 00101       2*     C
 00101       3*     C      INTIAL CONDITIONS
 00101       4*     C
 00101       5*     C      FOR FIRST AND SECOND DERIVATIVES OF      U SET C=1 AND B=0
 00101       6*     C      FOR FIRST AND SECOND DERIVATIVES OF      V SET C=0 AND B=0
 00101       7*     C      FOR FIRST AND SECOND DERIVATIVES OF FNCBR SET C=1 AND B=1
 00101       8*     C
 00103       9*            DIMENSION Y(20)
 00104      10*            COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
 00104      11*           1            DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DE1DL
 00105      12*            Y(1)     = Y1
 00106      13*            Y(2)     = Y2
 00107      14*            Z        = FNC
 00110      15*            E1       = FM*ETA(Z)+FN
 00111      16*            E2       = THETAB**4-THETAS**4
```

```
00112   E3     = E2/(1-THETAB)
00113   Y(3)   = 0.
00114   DE1DTB = DNCDT1*(DFM*ETA(Z)+FM*DETA(Z))
00115   DE2DTB = 4.*(THETAB**3)
00116   DE3DTB = E3*(DE2DTB+E3)/E2
00117   DE4DTB = E1*DE3DTB+E3*DE1DTB
00120   DE1DL  = ETA(Z)*DFMX+DFNX
00121   DE1DNC = ETA(Z)*DFM+DETA(Z)*FM
00122   DE4DU  = E3*DE1DL*DLDU+B*E3*DE1DNC*THETAB**3
00123   Y(4)   = -((1-C)/V**2+DE4DU)/DE4DTB
00124   Y(5)   = 0.
00125   DE2DU  = DE2DTB*Y(4)
00126   DE3DU  = E3*(E3*Y(4)+DE2DU)/E2
00127   D2E2TU = 12.*Y(4)*(THETAB**2)
00130   A1     = E3*(D2E2TU+DE3DU)/E2
00131   A2     = DE2DTB+E3
00132   A3     = -E3*DE2DU/(E2**2)+DE3DU/E2
00133   D2E3TU = A1+A2*A3
00134   DFNDU  = DFNX*DLDU
00135   DFCNDU = DNCDT1*Y(4)+B*Y(2)**3
00136   DFMDU  = DFMX*DLDU+DFM*DFCNDU
00137   DETADU = DETA(Z)*DFCNDU
00140   DE1DU  = FM*DETADU+ETA(Z)*DFMDU+DFNDU
00141   D2NCTU = 6*ENCBR*Y(2)*Y(4)+B*3*Y(2)**2
00142   DDETDU = DDETA(Z)*DFCNDU
00143   D2MNCU = D2MLNC*DLDU+D2MDNC*DFCNDU
00144   B1     = DFM*DETADU+ETA(Z)*D2MNCU+FM*DDETDU+DETA(Z)*DFMDU
00145   B2     = DE1DTB*D2NCTU/DNCDT1
00146   D2E1TU = DNCDT1*B1+B2
00147   D2MDLU = D2FMLL*DLDU+D2MLNC*DFCNDU
00150   D2NDLU = D2NDLL*DLDU
00151   D2E1LU = ETA(Z)*D2MDLU+DETADU*DFMX+D2NDLU
00152   D2E4LU = E3*D2E1LU+DE1DL*DE3DU
00153   D1     = DETA(Z)*DFCNDU*DFM
00154   D2     = ETA(Z)*D2MNCU
00155   D3     = DDETA(Z)*FM*DFCNDU
00156   D4     = DETA(Z)*DFMDU
00157   D2E1CB = D1+D2+D3+D4
00160   D2E4CB = DE3DU*DE1DNC+E3*D2E1CB
00161   D2NCNB = 3*Y(4)*Y(2)**2
00162   TERMNB = B*(E3*DE1DNC*D2NCNB+Y(2)**3*D2E4CB)
00163   D2E4UU = E3*DE1DL*D2LDUU+DLDU*D2E4LU+TERMNB
00164   D2E4TU = E1*D2E3TU+DE3DTB*DE1DU+E3*D2E1TU+DE1DTB*DE3DU
00165   Y(6)   = -(Y(4)*D2E4TU+D2E4UU+2*(C-1)/V**3)/DE4DTB
00166   Y3     = Y(3)
00167   Y4     = Y(4)
00170   Y5     = Y(5)
00171   Y6     = Y(6)
00172   RETURN
00173   END

        END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
@FOR,IS INTMIX
FOR S9A-07/27/72-17:45:03 (,0)

    SUBROUTINE INTMIX    ENTRY POINT 000024

    STORAGE USED: CODE(1) 000032; DATA(0) 000007; BLANK COMMON(2) 000000

     COMMON BLOCKS:

     0003   BLKI   000024

    EXTERNAL REFERENCES (BLOCK, NAME)

     0004   NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0003 R 000023 DE1DL      0000 R 000000 DE4DL      0003 R 000021 DE4DTB     0000 R 000001 DE4DXY     0003   000013 DFM
     0003   000012 DFMX       0003   000010 DFNX       0003   000007 DNCDT1     0003   000016 D2FMLL     0003   000015 D2MDNC
     0003   000014 D2MLNC     0003   000011 D2NDLL     0003 R 000022 E3         0003   000002 FM         0003   000003 FN
     0003   000005 FNC        0003   000004 FNCBR      0000   000002 INJP$      0003   000000 THETAB     0003   000001 THETAS
     0003   000006 V          0003   000017 Y1         0003   000020 Y2


 00101      1*          SUBROUTINE INTMIX(D2E4TY,DTBDX,DLDX,D2E4LY,D2LDXY,D2TBXY,D2TFXY)
 00101      2*     C
 00101      3*     C    INITIAL MIXED DERIVATIVES
 00101      4*     C
 00103      5*          COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
 00103      6*         1          DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DE1DL
 00104      7*          D2TFXY  = 0.
 00105      8*          DE4DL   = E3*DE1DL
 00106      9*          DE4DXY  = D2E4LY+DLDX+DE4DL*D2LDXY
 00107     10*          D2TBXY  = -(DE4DXY+DTBDX*D2E4TY)/DE4DTB
 00110     11*          RETURN
 00111     12*          END

       END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
@FOR,IS FMINV
FOR S9A-07/27/72-17:45:07 (,0)

   SUBROUTINE FMINV     ENTRY POINT 000254

   STORAGE USED: CODE(1) 000306; DATA(0) 001260; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000046 105G      0001   000151 11L       0001   000051 111G      0001   000071 117G      0001   000104 123G
    0001   000125 133G      0001   000130 137G      0001   000172 150G      0001   000175 154G      0001   000227 164G
    0001   000212 20L       0000 R 001214 AA        0000 R 001217 B         0000 I 001212 I         0000   001224 INJP$
    0000 I 001215 I1        0000 I 001220 I2        0000 I 001213 J         0000 I 001216 K         0000 R 000000 XMAT


00101      1*            SUBROUTINE FMINV (A,X,N,M)                                      1
00101      2*      C
00101      3*      C     MATRIX INVERSION AND SOLUTION OF SIMULTANEOUS EQUATIONS
00101      4*      C
00103      5*            DIMENSION A(N,N),X(N),XMAT(25,26)                               2
00104      6*            DO 1 I=1,N                                                      3
00107      7*            XMAT(I,M) = X(I)                                                4
00110      8*            DO 1 J=1,N                                                      5
00113      9*          1 XMAT(I,J) = A(I,J)                                              6
00116     10*            DO 20 I=1,N                                                     7
00121     11*            AA = XMAT(I,I)                                                  8
00122     12*            DO 5 J=1,M                                                      9
00125     13*          5 XMAT(I,J) = XMAT(I,J)/AA                                        10
00127     14*            IF (I.EQ.1) GO TO 11                                            11
00131     15*            I1 = I-1                                                        12
00132     16*            DO 10 K=1,I1                                                    13
00135     17*            B = XMAT(K,I)                                                   14
00136     18*            DO 10 J=1,M                                                     15
00141     19*         10 XMAT(K,J) = XMAT(K,J) - XMAT(I,J) * B                           16
00144     20*            IF (I.EQ.N) GO TO 20                                            17
00146     21*         11 I2 = I+1                                                        18
00147     22*            DO 15 K=I2,N                                                    1
00152     23*            B = XMAT(K,I)                                                   20
00153     24*            DO 15 J=1,M                                                     21
00156     25*         15 XMAT(K,J) = XMAT(K,J) - XMAT(I,J) * B                           22
00161     26*         20 CONTINUE                                                        23
00163     27*            DO 25 I=1,N                                                     24
00166     28*         25 X(I) = XMAT(I,M)                                                25
00170     29*            RETURN                                                          26
00171     30*            END

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
2   @FOR,IS ETA
    FOR S9A-07/27/72-17:45:11 (,0)

      FUNCTION ETA          ENTRY POINT 000036

      STORAGE USED: CODE(1) 000044; DATA(0) 000022; BLANK COMMON(2) 000000

      EXTERNAL REFERENCES (BLOCK, NAME)

       0003   POLY
       0004   EXP
       0005   NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0001   000015 1L        0000 R 000001 A        0000 R 000010 B        0000 R 000000 ETA        0000   000014 INJP$
       0003 R 000000 POLY

    00101      1*           FUNCTION ETA(X)
    00101      2*     C
    00101      3*     C     FIN EFFECTIVENESS OF THE UNOBSTRUCTED FIN
    00101      4*     C
    00103      5*           DIMENSION A(7), B(2)
    00104      6*           DATA A(1),A(2),A(3),A(4),A(5),A(6),A(7)/0.10E+01, -0.1163143E+01,
    00104      7*          1     0.1478836E+01, -0.1267550E+01, 0.6325223E+00, -0.1627067E+00,
    00104      8*          2     0.1654223E-01/ B(1),B(2)/0.6866095E+00, -0.2297718E+00/
    00116      9*           IF(X.GT.2.5) GO TO 1
    00120     10*           ETA    = POLY(7,A,X)
    00121     11*           RETURN
    00122     12*         1 ETA    = B(1)*EXP(B(2)*X)
    00123     13*           RETURN
    00124     14*           END

        END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS DETA
FOR S9A-07/27/72-17:45:15 (,0)

   FUNCTION DETA        ENTRY POINT 000036

   STORAGE USED: CODE(1) 000044; DATA(0) 000021; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   POLY
    0004   EXP
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000016 1L        0000 R 000001 A        0000 R 000007 B        0000 R 000000 DETA      0000   000013 INJP$
    0003 R 000000 POLY


00101      1*           FUNCTION DETA(X)
00101      2*      C
00101      3*      C    DETA/DNC
00101      4*      C
00103      5*           DIMENSION A(6), B(2)
00104      6*           DATA A(1),A(2),A(3),A(4),A(5),A(6)/-0.1163143E+01, 0.2957672E+01,
00104      7*          1     -0.3802550E+01, 0.2530089E+01, -0.8135335E+00, 0.9925338E-01/
00104      8*          2     B(1),B(2)/-0.1577635E+00, -0.2297718E+00/
00115      9*           IF(X.GT.2.5) GO TO 1
00117     10*           DETA  = POLY(6,A,X)
00120     11*           RETURN
00121     12*         1 DETA  = B(1)*EXP(B(2)*X)
00122     13*           RETURN
00123     14*           END

      END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
@FOR,IS DDETA
FOR S9A-07/27/72-17:45:23 (,0)

   FUNCTION DDETA        ENTRY POINT 000036

   STORAGE USED: CODE(1) 000044; DATA(0) 000020; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   POLY
    0004   EXP
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000016 1L        0000 R 000001 A          0000 R 000006 B          0000 R 000000 DDETA      0000   000012 INJP$
    0003 R 000000 POLY

 00101       1*             FUNCTION  DDETA(X)
 00101       2*      C
 00101       3*      C      D2ETA/D2NC
 00101       4*      C
 00103       5*             DIMENSION  A(5), B(2)
 00104       6*             DATA  A(1),A(2),A(3),A(4),A(5)/0.2957672E+01, -0.7605300E+01,
 00104       7*            1       0.7590257E+01, -0.3254134E+01, 0.4962669E+00/ B(1),B(2)/
 00104       8*            2       0.3624960E-01, -0.2297718E+00/
 00114       9*             IF(X.GT.2.5) GO TO 1
 00116      10*             DDETA  = POLY(5,A,X)
 00117      11*             RETURN
 00120      12*           1 DDETA  = B(1)*EXP(B(2)*X)
 00121      13*             RETURN
 00122      14*             END

     END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS DDDETA
FOR S9A-07/27/72-17:45:35 (,0)

    FUNCTION DDDETA         ENTRY POINT 000036

    STORAGE USED: CODE(1) 000044; DATA(0) 000017; BLANK COMMON(2) 000000

    EXTERNAL REFERENCES (BLOCK, NAME)

     0003   POLY
     0004   EXP
     0005   NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0001   000016 1L        0000 R 000001 A          0000 R 000005 B          0000 R 000000 DDDETA     0000    000011 INJP$
     0003 R 000000 POLY


00101       1*              FUNCTION  DDDETA(X)
00101       2*       C
00101       3*       C      DDETA/D3NC
00101       4*       C
00103       5*              DIMENSION  A(4), B(2)
00104       6*              DATA  A(1),A(2),A(3),A(4)/-0.7405300E+01,0.1518053E+02,
00104       7*             1-0.9762402E+01,0.1985068E+01/
00104       8*             2 B(1),B(2)/-0.8329136E-02,-0.2297718E+00/
00113       9*              IF(X.GT.2.5) GO TO 1
00115      10*              DDDETA  = POLY(4,A,X)
00116      11*              RETURN
00117      12*            1 DDDETA  = B(1)*EXP(B(2)*X)
00120      13*              RETURN
00121      14*              END

         END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS POLY
FOR S9A-07/27/72-17:45:42 (,0)

   FUNCTION POLY        ENTRY POINT 000036

   STORAGE USED: CODE(1) 000044; DATA(0) 000015; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000012 1076G     0000   000003 INJP$     0000 I 000002 K        0000 I 000001 L        0000 R 000000 POLY


00101      1*              FUNCTION POLY(N,A,X)
00101      2*      C
00101      3*      C       POLYNOMIAL EVALUATION   Y = A(N)+X*A(N-1)+X**2*A(N-2)+...
00101      4*      C
00103      5*              DIMENSION A(N)
00104      6*              POLY = 0.
00105      7*              L = N
00106      8*              DO 1 K=1,N
00111      9*              POLY = POLY*X+A(L)
00112     10*            1 L = L-1
00114     11*              RETURN
00115     12*              END

     END OF COMPILATION:      NO  DIAGNOSTICS.
```


158687

```
02  @XQT
    MAP 0023-07/27-17:45

    ADDRESS LIMITS     001000 020432      040000 047526
    STARTING ADDRESS   017366

    WORDS DECIMAL        7363 IBANK        3927 DBANK

                     SEGMENT MAIN          001000 020432      040000 047526

    NSWTC$/FOR              1    001000 001021
    NRELK$/FOR              1    001022 001044
    NRWN$/FOR               1    001045 001124     2    040000 040011
    NWEF$/FOR               1    001125 001326     2    040012 040031
    NFTCH$/FOR              1    001327 001617     2    040032 040067
    NBDCL$/FOR              1    001620 001752     2    040070 040125
    NFTV$/FOR               1    001753 001775
    NCNVT$/FOR              1    001776 002222     2    040126 040215
    NCLOS$/FOR              1    002223 002371     2    040216 040247
    NWELK$/FOR              1    002372 002513
    NBSBL$/FOR              1    002514 002550
    NUPDA$/FOR              1    002551 002603
    NBFOR$/FOR                                     2    040250 042451
    NININ$/FOR              1    002604 003014     2    042452 042463
    NINPT$/FOR              1    003015 003657     2    042464 042503
    NOTIN$/FOR              1    003660 004173     2    042504 042507
    NOUT$/FOR               1    004174 005162     2    042510 042534
    NFMT$/FOR               1    005163 006041     2    042535 042611
    NIOER$/FOR              1    006042 006214     2    042612 042716
    NFCHK$/FOR              1    006215 007076     2    042717 043055
                                                   4    043056 043127
    NTAB$/FOR                                      2    043130 043166
    ERU$/MISC
    NIBUF$/FOR              1    007077 007141     2    043167 043167
    TIR$/TECH               1    007142 007626     0    043170 043220
                                                   2    043221 043500
    NEXP6$/FOR              1    007627 007714     2    043501 043510
    NIER$/FOR               1    007715 007776     2    043511 043644
    NOBUF$/FOR              1    007777 010043
    ASINCOS$/FOR            1    010044 010260     0    043645 043672
    SQRT$/FOR               1    010261 010321     2    043673 043704
    EXP$/FOR                1    010322 010411     2    043705 043725
    NERR$/FOR               1    010412 010736     2    043726 044071
    BLKR (COMMON BLOCK)                                 044072 044103
    VAB (COMMON BLOCK)                                  044104 044106
    MCT (COMMON BLOCK)                                  044107 044113
    BLKB (COMMON BLOCK)                                 044114 044115
    BLKD (COMMON BLOCK)                                 044116 044127
    BLKI (COMMON BLOCK)                                 044130 044153
```

BLANK$COMMON (COMMON BLOCK)

POLY	1	010737 011002	0	044154 044170
			2	BLANK$COMMON
DDDETA	1	011003 011046	0	044171 044207
			2	BLANK$COMMON
DDETA	1	011047 011112	0	044210 044227
			2	BLANK$COMMON
DETA.	1	011113 011156	0	044230 044250
			2	BLANK$COMMON
ETA	1	011157 011222	0	044251 044272
			2	BLANK$COMMON
EMINV	1	011223 011530	0	044273 045552
			2	BLANK$COMMON
INIMIX	1	011531 011562	0	045553 045561
	3	BLKI	2	BLANK$COMMON
INTIAL	1	011563 012304	0	045562 045704
	3	BLKI	2	BLANK$COMMON
DERV	1	012305 013225	0	045705 046034
	3	BLKI	2	BLANK$COMMON
			4	BLKD
MIXDER	1	013226 013701	0	046035 046140
	3	BLKI	2	BLANK$COMMON
			4	BLKD
NCDERV	1	013702 013745	0	046141 046153
			2	BLANK$COMMON
MDER	1	013746 014251	0	046154 046241
			2	BLANK$COMMON
NDERV	1	014252 014341	0	046242 046256
			2	BLANK$COMMON
REST	1	014342 014461	0	046257 046301
			2	BLANK$COMMON
RESBND	1	014462 014713	0	046302 046354
			2	BLANK$COMMON
BNDCND	1	014714 015324	0	046355 046441
	3	BLK3	2	BLANK$COMMON
SCNTL	1	015325 015520	0	046442 046455
	3	MCT	2	BLANK$COMMON
	5	BLKI	4	VAB
SDRV	1	015521 016325	0	046456 046543
	3	BLKI	2	BLANK$COMMON
	5	BLKR	4	BLKD
			6	VAB
RKS	1	016326 017365	0	046544 046627
			2	BLANK$COMMON
MAIN	1	017366 020432	0	046630 047526
	3	BLKI	2	BLANK$COMMON
	5	MCT	4	BLKD
	7	BLKB	6	BLKR

SYS$*RLIB$. LEVEL 63
END OF COLLECTION - TIME 2.084 SECONDS

APPENDIX C

COMPUTER CODE FOR MINIMUM MASS OPTIMIZATION

```
@FOR,IS MAIN
FOR S9A-07/27/72-19:37:17 (,0)

    MAIN PROGRAM

    STORAGE USED: CODE(1) 001035; DATA(0) 000776; BLANK COMMON(2) 000000

     COMMON BLOCKS:

     0003   BLKI    000024
     0004   BLKD    000012
     0005   MCT     000005
     0006   BLKR    000012

    EXTERNAL REFERENCES (BLOCK, NAME)

     0007   SDRV
     0010   SCNTL
     0011   CONST
     0012   NDERV
     0013   ETA
     0014   DETA
     0015   NCDERV
     0016   MDER
     0017   INTIAL
     0020   INTMIX
     0021   RKS
     0022   FMINV
     0023   BNDCND
     0024   NINTR$
     0025   NRNL$
     0026   NWDU$
     0027   NIO2$
     0030   SQRT
     0031   ASIN
     0032   NIO1$
     0033   NSTOP$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0001   000002 1L       0001   000122 100L     0001   000273 105L     0001   000311 108L     0001   000314 109L
     0001   000321 110L     0000   000466 2000F    0000   000526 2010F    0000   000541 2040F    0000   000557 205F
     0000   000716 2050F    0000   000712 2060F    0001   001015 2070L    0000   000725 2075F    0001   001023 2090L
     0000   000733 2095F    0000   000566 210F     0000   000573 215F     0000   000654 220F     0000   000673 221F
     0000   000605 230F     0001   000462 233G     0000   000621 240F     0000   000636 250F     0001   001007 300L
     0001   001031 3000L    0001   000703 346G     0001   000725 355G     0001   000024 50L      0001   000117 90L
     0000   000534 900F     0000 R 000050 A        0000 R 000355 ALPHA    0000 R 000337 AM       0000 R 000372 AYY
     0004   000003 A1       0004   000004 A2       0004   000005 A3       0004   000006 A4       0004   000007 A5
     0004   000010 A6       0000 R 000373 BYY      0000 R 000356 DA       0014 R 000000 DETA     0000 R 000377 DETZ
     0003   000023 DE1DL    0003   000021 DE4DTB   0003 R 000013 DFM      0003 R 000012 DFMX     0003 R 000010 DFNX
     0006 R 000006 DLDNCB   0006 R 000001 DLDU     0006 R 000002 DLDV     0005 R 000000 DLMT     0000 R 000120 DLY
     0003 R 000007 DNCDT1   0000 R 000357 DR       0000 R 000414 DX       0005 R 000003 DXWRT    0000 R 000024 DY
     0000 R 000240 DYST     0000 R 000400 DYY      0000 R 000403 D2CD2T   0000 R 000404 D2CTNB   0000 R 000412 D2E4LN
     0000 R 000406 D2E4LU   0000 R 000410 D2E4LV   0000 R 000411 D2E4TN   0000 R 000405 D2E4TU   0000 R 000407 D2E4TV
```

```
0003 R 000016 D2FMLL   0006 R 000003 D2LDUU   0006 R 000004 D2LDUV   0006 R 000005 D2LDVV   0006 R 000007 D2LNCB
0006 R 000010 D2LUNB   0006 R 000011 D2LVNB   0003 R 000015 D2MDNC   0003 R 000014 D2MLNC   0003 R 000011 D2NDLL
0004 R 000001 D3M2LN   0004 R 000002 D3M2NL   0004   000011 EB       0013 R 000000 ETA      0000 R 000375 ETY
0000 R 000374 ETYY     0003   000022 E3       0003 R 000002 FM       0003 R 000003 FN       0003 R 000005 FNC
0003 R 000004 FNCBR    0000 I 000415 I        0000 I 000420 IBKP     0005 I 000001 ICNT     0000 I 000422 IERR
0000 I 000416 IFVD     0000   000426 INPUT    0000 I 000424 LCNT     0005   000004 LSTEP    0000 I 000423 LOTEP
0000 I 000417 N        0000 I 000367 NCT      0000 I 000360 NIT      0000 I 000421 NTRY     0000 R 000144 PD
0000 R 000354 PHI2     0000 R 000074 R        0000 R 000365 RHOD     0006 R 000000 RLAM     0010 R 000000 SCNTL
0000 R 000170 SD       0007 R 000000 SDRV     0000 R 000364 TFLEXT   0003 R 000000 THETAB   0000 R 000402 THETAF
0003 R 000001 THETAS   0000 R 000366 TH4      0000 R 000353 TOTMAS   0000 R 000370 T2       0000 R 000371 T3
0004 R 000000 U        0003 R 000006 V        0000 R 000334 XM       0005 R 000002 XWRT     0000 R 000000 Y
0000 R 000214 YS       0000 R 000310 YSIMP    0000 R 000264 YST      0000 R 000376 YY       0003 R 000017 Y1
0003 R 000020 Y2       0000 R 000425 ZS       0000 R 000401 ZZ       0000 R 000413 ZZZ      0000 R 000361 Z1
0000 R 000362 Z2       0000 R 000363 Z3

00100     C
00100     C
00100     C     RADIATOR SYSTEMS OPTIMIZATION FOR MINIMUM MASS
00100     C
00100     C
00101           EXTERNAL SDRV,SCNTL
00103           DIMENSION Y(20),DY(20),A(20),R(20),DLY(20),PD(20),SD(20),
00103          1         YS(20),DYST(20),YST(20),YSIMP(20),XM(3),AM(3,4)
00104           COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
00104          1            DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DE1DL/
00104          2       BLKD/U,D3M2LN,D3M2NL,A1,A2,A3,A4,A5,A6,EB
00104          3       /MCT/DLMT,ICNT,XWRT,DXWRT,LSTEP
00104          4       /BLKR/RLAM,DLDU,DLDV,D2LDUU,D2LDUV,D2LDVV,DLDNCB,
00104          5            D2LNCB,D2LUNB,D2LVNB
00105           NAMELIST /INPUT/ THETAS,U,V,FNCBR,TOTMAS,PHI2,ALPHA,DXWRT,DA,DR
00105     C
00106           FFS(X)   = (SQRT(X*(X+2.0))+ASIN(1.0/(X+1.0))-1.570796)/X
00107           FFN(X,Y)= 1.0-Y*(0.1460*Y-0.02866)/X
00110           DFFN(X,Y)=(-0.2920*Y+0.02866)/X
00111         1 READ(5,INPUT,END=3000)
00114           NIT      =  1
00115           Z1       = U
00116           Z2       = V
00117           Z3       = FNCBR
00120           XM(1)    = 0.
00121           XM(2)    = 0.
00122           XM(3)    = 0.
00123           TFLEXT   = 2.0
00124        50 CALL CONST(TOTMAS,PHI2,ALPHA,U,V,FNCBR,RLAM,DLDU,DLDV,DLDNCB,
00124          1          D2LDUU,D2LDUV,D2LUNB,D2LDVV,D2LVNB,D2LNCB)
00125           CALL NDERV(RLAM,FN,DFNX,D2NDLL)
00126           RHOD     = RLAM
00127           WRITE(6,2000) THETAS,U,V,FNCBR,TOTMAS,PHI2,ALPHA,RLAM
00141      2000 FORMAT (1H1,10X,10HTHETA-S = F20.8/
00141          1          11X,10H       U = F20.6/
00141          2          11X,10H       V = F20.6/
00141          3          11X,10H   FNCBR = F20.6/
00141          4          11X,10H  TOTMAS = F20.6/
00141          5          11X,10H    PHI2 = F20.6/
```

```
     6              11X,10H  ALPHA = F20.6/
     9              11X,10H LAMBDA = F20.6///)
      WRITE(6,2010) NIT
 2010 FORMAT (5X'NUMBER OF ITERATIONS = 'I2)
      TH4      = THETAS**4
      IF(V.GT.0.2) GO TO 90
      IF(V.LE.1.0E-10) GO TO 108
   90 THETAB   = 0.9
   95 NCT      = 0
  100 T2       = THETAB**2
      T3       = T2*THETAB
      FNC      = FNCBR*T3
      FM       = FFN(RHOD,FNC)*FFS(RHOD)
      DFM      = DFFN(RHOD,FNC)*FFS(RHOD)
      AYY      = T3*THETAB-TH4
      BYY      = 1.0-THETAB
      ETYY     = ETA(FNC)
      ETY      = ETYY*FM+FN
      YY       = 1.0/V-ETY*AYY/BYY
      DETZ     = (DFM*ETYY+FM*DETA(FNC))*3.0*T2*FNCBR
      DYY      = -AYY/BYY*(DETZ+ETY/BYY)-4.0*ETY*T3/BYY
      ZZ       = THETAB-YY/DYY
      IF(NCT.GT.20) GO TO 109
      IF(ZZ.LT.1.0) GO TO 105
      THETAB   = (THETAB+1.0)/2.0
      NCT      = NCT+1
      GO TO 100
  105 IF(ABS(ZZ-THETAB)/ZZ.LT.1.0E-06) GO TO 110
      NCT               = NCT+1
      THETAB   = ZZ
      GO TO 100
  108 THETAB            = 1.0
      GO TO 110
  109 WRITE(6,900)
C
  110 Y(1)     = 1.0
      Y(2)     = THETAB
  900 FORMAT(1H0,20HNEWTON-RAPHSON FAILS)
      THETAF   = Y(1)
      Y1       = Y(1)
      Y2       = Y(2)
        CALL NCDERV(FNCBR,THETAB,FNC,DNCDT1,D2CD2T,D2CTNR)
      CALL MDER(RLAM,FNC,FM,DFMX,DFM,D2FMLL,D2MDNC,D2MLNC,D3M2LN,D3M2NL)
      CALL INTIAL(DLDU,D2LDUU,Y(3),Y(4),Y(5),Y(6),0.,1.,D2E4TU,D2E4LU)
      CALL INTIAL(DLDV,D2LDVV,Y(7),Y(8),Y(9),Y(10),0.,0.,D2E4TV,D2E4LV)
      CALL INTIAL(DLDNCB,D2LNCB,Y(11),Y(12),Y(13),Y(14),1.,1.,D2E4TN,
     1                D2E4LN)
      CALL INTMIX(D2E4TV,Y(4),DLDU,D2E4LV,D2LDUV,Y(16),Y(15))
      CALL INTMIX(D2E4TN,Y(4),DLDU,D2E4LN,D2LUNB,Y(18),Y(17))
      CALL INTMIX(D2E4TN,Y(8),DLDV,D2E4LN,D2LVNB,Y(20),Y(19))
      ZZZ      = 0
      DLMT     = DX/500.
      DO 115 I=1,20
      A(I)     = DA
  115 R(I)     = DR
C
      WRITE(6,2040)
```

```
00242   2040 FORMAT(/7X,6HTHETAF,10X,6HTHETAB,10X,2HNC,13X,6HETABAR,
00242      1        11X,3HETA,13X,1HM,14X,1HN,14X,1HX/)
00242   C
00243        ZZZ     = 0.0
00244        DX      = 1.0/(U*(1.0-THETAB))
00245        IFVD    = 0
00246        IF(DX.GT.DXWRT) DX = DXWRT
00250        N       = 20
00251        IBKP    = 1
00252        NTRY    = 1
00253        IERR    = 0
00253   C
00254        DLMT    = DXWRT/500.0
00255        XWRT    = 0.0
00256        LOTEP   = 0
00257        LCNT    = 0
00260        ICNT    = 0
00261        CALL RKS(SDRV,SCNTL,Y,DY,A,R,ZZZ,DX,N,IFVD,IBKP,NTRY,IERR,
00261      1         DLY,PD,SD,YS,YST,DYST,YSIMP)
00262        XM(1)   =  -Y(3)
00263        XM(2)   =  -Y(7)
00264        XM(3)   =  -Y(11)
00265        AM(1,1) =  Y(5)
00266        AM(1,2) =  Y(15)
00267        AM(1,3) =  Y(17)
00270        AM(2,1) =  Y(15)
00271        AM(2,2) =  Y(9)
00272        AM(2,3) =  Y(19)
00273        AM(3,1) =  Y(17)
00274        AM(3,2) =  Y(19)
00275        AM(3,3) =  Y(13)
00276        CALL FMINV(AM,XM,3,4)
00277    205 FORMAT (///45X,27H PARAMETERS AT END OF TUBE //)
00300    210 FORMAT (1X,51X,8HTHETAF =,F10.6,/)
00301    215 FORMAT (1X,23X,3HU =,E14.6,16X,3HV =,E14.6,12X,7HNCBAR =,E14.6,/)
00302    230 FORMAT (1X,19X,7HDTFDU =,E14.6,12X,7HDTFDV =,E14.6,8X,
00302      1        11HDTFNCBAR =,E14.6,/)
00303    240 FORMAT (1X,17X,9HD2TFD2U =,E14.6,10X,9HD2TFD2V =,E14.6,6X,
00303      1        13HD2TFD2NCBAR =,E14.6,/)
00304    250 FORMAT (1X,16X,10HD2TFDUDV =,E14.6,5X,14HD2TFDUDNCBAR =,E14.6,
00304      1        5X,14HD2TFDVDNCBAR =,E14.6,/)
00305    220 FORMAT (/,3X,24HINCR. SOUGHT   DELTA U =,E14.6,10X,9HDELTA V =,
00305      1          E14.6,6X,13HDELTA NCBAR =,E14.6/)
00306    221 FORMAT (/,3X,24HINCR. USED     DELTA U =,E14.6,10X,9HDELTA V =,
00306      1          E14.6,6X,13HDELTA NCBAR =,E14.6/)
00307        NIT     =  NIT+1
00310        Z1      = U
00311        Z2      = V
00312        Z3      = FNCBR
00313        WRITE (6,205)
00315        WRITE (6,210) Y(1)
00320        WRITE (6,215) Z1,Z2,Z3
00325        WRITE (6,230) Y(3),Y(7),Y(11)
00332        WRITE (6,240) Y(5),Y(9),Y(13)
00337        WRITE (6,250) Y(15),Y(17),Y(19)
00344        WRITE(6,220) (XM(I),I=1,3)
00352        CALL BNDCND(U,V,FNCBR,XM,TOTMAS,ALPHA)
```

```
00353     154*          WRITE(6,221) (XM(I),I=1,3)
00361     155*      200 IF (NIT .GT. 20) GO TO 300
00363     156*          ZS       = Y(1)-THETAS
00364     157*          IF(ZS.LT.0.0001) GO TO 2070
00366     158*          IF(ABS(Y(1)-TFLEXT)/Y(1) .LE. 5.0E-04) GO TO 2090
00370     159*          TFLEXT  = Y(1)
00371     160*          IF (ABS(XM(1))/Z1.GT.0.0001) GO TO 50
00373     161*          IF (ABS(XM(2))/Z2.GT.0.0001) GO TO 50
00375     162*          IF (ABS(XM(3))/Z3.GT.0.0001) GO TO 50
00377     163*          WRITE(6,2060)
00401     164*     2060 FORMAT(1H0,15HOPTIMUM REACHED)
00402     165*          GO TO 1
00403     166*      300 WRITE (6,2050)
00405     167*     2050 FORMAT (1X,32H NUMBER OF ITERATIONS EXCEEDS 20 )
00406     168*          GO TO 1
00407     169*     2070 WRITE(6,2075)
00411     170*     2075 FORMAT(1H0,24HVANISHING HEAT REJECTION)
00412     171*          GO TO 1
00413     172*     2090 WRITE(6,2095)
00415     173*     2095 FORMAT(1H0,40HINSIGNIFICANT IMPROVEMENT OVER LAST STEP)
00416     174*          GO TO 1
00417     175*     3000 STOP
00420     176*          END

          END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
QFOR,IS CONST
FOR S9A=07/27/72-19:37:44 (,0)

   SUBROUTINE CONST      ENTRY POINT 000373

   STORAGE USED: CODE(1) 000446; DATA(0) 000071; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   SQRT
    0004   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000 R 000003 A       0000 R 000002 AZ      0000 R 000004 B       0000 R 000005 C       0000 R 000006 D
    0000 R 000010 DAN     0000 R 000007 DAV     0000 R 000014 DBN     0000 R 000013 DBV     0000 R 000017 DCU
    0000 R 000020 DCV     0000 R 000026 DDN     0000 R 000024 DDU     0000 R 000025 DDV     0000 R 000023 DZ
    0000 R 000012 D2ANN   0000 R 000031 D2AVN   0000 R 000011 D2AVV   0000 R 000016 D2BNN   0000 R 000032 D2BVN
    0000 R 000015 D2BVV   0000 R 000021 D2CUU   0000 R 000033 D2CUV   0000 R 000022 D2CVV   0000 R 000030 D2DNN
    0000 R 000034 D2DVN   0000 R 000027 D2DVV   0000   000043 INJP$   0000 R 000000 PI      0000 R 000001 Z


 00101      1*             SUBROUTINE CONST(TOTMAS,PHI2,ALPHA,U,V,FNCBR,HOD,DLDU,DLDV,DLDNCB,
 00101      2*            1                 D2LDUU,D2LDUV,D2LUNB,D2LDVV,D2LVNB,D2LNCB)
 00101      3*       C
 00103      4*             PI      = 3.1415926
 00104      5*             Z       = V**3/FNCBR
 00105      6*             AZ      = ALPHA*ALPHA
 00106      7*             A       = TOTMAS-AZ*Z
 00107      8*             IF (A .LT. 1.0E-06)  A = 1.0E-06
 00107      9*       C
 00111     10*          50 B       = PI*ALPHA*Z
 00112     11*             C       = PHI2*ALPHA*V**2*SQRT(U)
 00113     12*             D       = SQRT(B*B+A*C)
 00114     13*             HOD     = (B+D)/A
 00114     14*       C
 00115     15*             DAV     = -3.0*B*ALPHA/(PI*V)
 00116     16*             DAN     = AZ*Z/FNCBR
 00117     17*             D2AVV   = -6.0*AZ*V/FNCBR
 00120     18*             D2ANN   = -2.0*DAN/FNCBR
 00120     19*       C
 00121     20*             DBV     = 3.0*B/V
 00122     21*             DBN     = -B/FNCBR
 00123     22*             D2BVV   = 6.0*B/(V*V)
 00124     23*             D2BNN   = 2.0*B/(FNCBR*FNCBR)
 00124     24*       C
 00125     25*             DCU     = C/(2.0*U)
 00126     26*             DCV     = 2.0*C/V
 00127     27*             D2CUU   = -DCU/(2.0*U)
 00130     28*             D2CVV   = DCV/V
 00130     29*       C
```

```
   00131     30*              DZ      = 2.0*D
   00132     31*              DDU     = A*DCU/DZ
   00133     32*              DDV     = (2.0*B*DBV+A*DCV+C*DAV)/DZ
   00134     33*              DDN     = (2.0*B*DBN+C*DAN)/DZ
   00135     34*              D2DVV   = (-DDV*DDV+DBV*DBV+B*D2BVV+DAV*DCV+(D2CVV*A+D2AVV*C)/2.0)
   00135     35*             1         /D
   00136     36*              D2DNN   = (-DDN*DDN+DBN*DBN+B*D2BNN+C*D2ANN/2.0)/D
   00136     37*        C
   00137     38*              D2AVN   = -3.0*DBN*ALPHA/(PI*V)
   00140     39*              D2BVN   = 3.0*DBN/V
   00141     40*              D2CUV   = C/(U*V)
   00142     41*              D2DVN   = (-DDN*DDV+DBV*DBN+B*D2BVN+(DCV*DAN+C*D2AVN)/2.0)/D
   00142     42*        C
   00143     43*              DLDU    = DCU/DZ
   00144     44*              DLDV    = (-HOD*DAV+DBV+DDV)/A
   00145     45*              DLDNCB  = (-HOD*DAN+DBN+DDN)/A
   00145     46*        C
   00146     47*              D2LDUU  = (D2CUU-DDU*DCU/D)/DZ
   00147     48*              D2LDVV  = (-2.0*DLDV*DAV-HOD*D2AVV+D2BVV+D2DVV)/A
   00150     49*              D2LNCB  = (-2.0*DLDNCB*DAN-HOD*D2ANN+D2BNN+D2DNN)/A
   00150     50*        C
   00151     51*              D2LDUV  = (D2CUV-2.0*DLDU*DDV)/DZ
   00152     52*              D2LUNB  = -DLDU*DDN/D
   00153     53*              D2LVNB  = (-DLDV*DAN-DLDNCB*DAV-HOD*D2AVN+D2BVN+D2DVN)/A
   00154     54*              RETURN
   00155     55*              END

        END OF COMPILATION:         NO  DIAGNOSTICS.
```

177938

```
@FOR,IS BNDCND
FOR S9A-07/27/72-19:37:47 (,0)

     SUBROUTINE BNDCND     ENTRY POINT 000154

     STORAGE USED: CODE(1) 000207; DATA(0) 000027; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003   NEXP6$
      0004   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000011 500L       0000   000014 INJP$     0000 I 000000 L          0000 R 000004 X1         0000 R 000005 X2
      0000 R 000001 Z1         0000 R 000002 Z2         0000 R 000003 Z3


 00101      1*          SUBROUTINE BNDCND(U,V,FNCBR,XM,TOTMAS,ALPHA)
 00101      2*     C
 00101      3*     C    BNDCND INSURES U,V, AND NCBAR ARE WITHIN THEIR BOUNDARY CONDITIONS
 00101      4*     C
 00103      5*          DIMENSION XM(3)
 00104      6*          L       = 1
 00105      7*          Z1      = U
 00106      8*          Z2      = V
 00107      9*          Z3      = FNCBR
 00110     10*      500 U       = Z1+XM(1)
 00111     11*          V       = Z2+XM(2)
 00112     12*          FNCBR   = Z3+XM(3)
 00113     13*          IF(L.EQ.2) RETURN
 00113     14*     C
 00115     15*          IF(U.LT.5.0E-06)       XM(1) = 5.0E-06-Z1
 00117     16*          IF(U.GT.250.0)         XM(1) = 250.0-Z1
 00121     17*          IF(V.LT.5.0E-06)       XM(2) = 5.0E-06-Z2
 00123     18*          IF(V.GT. 50.0)         XM(2) =  50.0-Z2
 00125     19*          IF(FNCBR.LT.1.0E-04)   XM(3) = 1.0E-04-Z3
 00127     20*          IF(FNCBR.GT.6.0)       XM(3) = 6.0-Z3
 00131     21*          X1      = TOTMAS*FNCBR
 00132     22*          X2      = ALPHA**2
 00133     23*          IF(X1-X2*V**3.LT.0.0)  XM(2) = (X1/X2)**(1.0/3.0)-Z2
 00135     24*          L       = 2
 00136     25*          GO TO 500
 00137     26*          END

       END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
QFOR,IS NDERV
FOR S9A-07/27/72-19:37:49 (,0)

   SUBROUTINE NDERV     ENTRY POINT 000056

   STORAGE USED: CODE(1) 000070; DATA(0) 000015; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   SQRT
    0004   ASIN
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000 R 000000 C4        0000 R 000001 C5        0000 R 000002 C6        0000   000006 INJP$


00101      1*             SUBROUTINE NDERV(RLAM,FN,DFNX,D2NDLL)
00101      2*     C
00101      3*     C       N AND ITS DERIVATIVES
00101      4*     C
00103      5*             C4      = RLAM+1.0
00104      6*             C5      = RLAM+2.0
00105      7*             C6      = SQRT(RLAM*C5)
00106      8*             FN      = (1.5707963+C5-C6-ASIN(1.0/C4))/RLAM
00107      9*             DFNX    = -(FN+C6/C4-1.0)/RLAM
00110     10*             D2NDLL  = -(2.0*DFNX+1.0/(C4*C4*C6))/RLAM
00111     11*             RETURN
00112     12*             END

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

177937

```
@FOR,IS MDER
FOR S9A-07/27/72-19:37:57 (,0)

     SUBROUTINE MDER       ENTRY POINT 000251

     STORAGE USED: CODE(1) 000304; DATA(0) 000066; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003   SQRT
      0004   ASIN
      0005   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0000 R 000011 CFN      0000 R 000000 C1       0000 R 000017 C10      0000 R 000020 C11      0000 R 000022 C12
      0000 R 000001 C2       0000 R 000003 C3       0000 R 000002 C4       0000 R 000004 C5       0000 R 000005 C6
      0000 R 000006 C7       0000 R 000007 C8       0000 R 000010 C9       0000 R 000015 DCFDNC   0000 R 000013 DCFNDL
      0000 R 000014 DFFSDL   0000 R 000016 D2CFLL   0000 R 000025 D2CFNC   0000 R 000026 D2CLNC   0000 R 000023 D2C2LL
      0000 R 000021 D2C3LL   0000 R 000024 D2FSLL   0000 R 000027 D3C2LN   0000 R 000030 D3C2NL   0000 R 000012 FFS
      0000   000040 INJPS


 00101      1*           SUBROUTINE MDER(RLAM,FNC,FM,DFMX,DFM,D2FMLL,D2MDNC,D2MLNC,D3M2LN,
 00101      2*          1D3M2NL)
 00101      3*      C
 00101      4*      C    M AND ITS DERIVATIVES
 00101      5*      C
 00103      6*           C1       = 0.1460*FNC-0.02866
 00104      7*           C2       = SQRT(RLAM*(2.+RLAM))
 00105      8*           C4       = 1.0/(RLAM+1.0)
 00106      9*           C3       = ASIN(C4)
 00107     10*           C5       = 1/(C4*C2)
 00110     11*           C6       = (C4**2)/SQRT(1-C4**2)
 00111     12*           C7       = 1/(RLAM**2)
 00112     13*           C8       = FNC/RLAM
 00113     14*           C9       = 2.*C7*C8
 00114     15*           CFN      = 1.0-C1*C8
 00115     16*           FFS      = (C2+C3-1.57079635)/RLAM
 00116     17*           FM       = CFN*FFS
 00117     18*           DCFNDL   = C1*C7*FNC
 00120     19*           DFFSDL   = (C5-C6-FFS)/RLAM
 00121     20*           DFMX     = CFN*DFFSDL+FFS*DCFNDL
 00122     21*           DCFDNC   = -(0.1460*FNC+C1)/RLAM
 00123     22*           DFM      = FFS*DCFDNC
 00124     23*           D2CFLL   = -C1*C9
 00125     24*           C10      = 1-C4**2
 00126     25*           C11      = +1.+0.5/C10*C4**2
 00127     26*           D2C3LL   = 2*(C4**3)*C11/SQRT(C10)
 00130     27*           C12      = 1/C2-C2*(C4**2)
 00131     28*           D2C2LL   = -C12/(C4*C2)**2
```

```
      D2FSLL  = (D2C2LL+D2C3LL-2.*DFFSDL)/RLAM
      D2FMLL  = CFN*D2FSLL+2.*DFFSDL*DCFNDL+FFS*D2CFLL
      D2CFNC  = -0.2920/RLAM
      D2MDNC  = FFS*D2CFNC
      D2CLNC  = (0.146*FNC+C1)*C7
      D2MLNC  = DCFDNC*DFFSDL+D2CLNC*FFS
      D3C2LN  = -2.*D2CLNC/RLAM
      D3M2LN  = DCFDNC*D2FSLL+2.*DFFSDL*D2CLNC+FFS*D3C2LN
      D3C2NL  = 0.2920*C7
      D3M2NL  = FFS*D3C2NL+DFFSDL*D2CFNC
      RETURN
      END

END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
SUBROUTINE INTIAL    ENTRY POINT 000470

STORAGE USED: CODE(1) 000522; DATA(0) 000123; BLANK COMMON(2) 000000

 COMMON BLOCKS:

 0003   BLKI   000024

EXTERNAL REFERENCES (BLOCK, NAME)

 0004   ETA
 0005   DETA
 0006   DDETA
 0007   NERR3$

STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

 0000 R 000037 A1         0000 R 000040 A2         0000 R 000041 A3         0000 R 000053 B1         0000 R 000054 B2
 0006 R 000000 DDETA      0000 R 000051 DDETDU     0005 R 000000 DETA       0000 R 000046 DETADU     0003 R 000023 DE1DL
 0000 R 000032 DE1DNC     0000 R 000027 DE1DTB     0000 R 000047 DE1DU      0000 R 000030 DE2DTB     0000 R 000034 DE2DU
 0000 R 000031 DE3DTB     0000 R 000035 DE3DU      0003 R 000021 DE4DTB     0000 R 000033 DE4DU      0000 R 000044 DFCNDU
 0007 R 000013 DFM        0000 R 000045 DFMDU      0003 R 000012 DFMX       0000 R 000043 DFNDU      0003 R 000010 DFNX
 0003 R 000007 DNCDT1     0000 R 000061 D1         0000 R 000062 D2         0000 R 000065 D2E1CB     0000 R 000060 D2E1LU
 0000 R 000055 D2E1TU     0000 R 000036 D2E2TU     0000 R 000042 D2E3TU     0000 R 000066 D2E4CB     0000 R 000071 D2E4UU
 0003 R 000016 D2FMLL     0000 R 000056 D2MDLU     0003 R 000015 D2MDNC     0003 R 000014 D2MLNC     0000 R 000052 D2MNCU
 0000 R 000067 D2NCHB     0000 R 000050 D2NCTU     0003 R 000011 D2NDLL     0000 R 000057 D2NDLU     0000 R 000063 D3
 0000 R 000064 D4         0004 R 000000 ETA        0000 R 000025 E1         0000 R 000026 E2         0003 R 000022 E3
 0003 R 000002 FM         0003 R 000003 FN         0003 R 000005 FNC        0003 R 000004 FNCBR      0000   000100 INJP$
 0000 R 000070 TERMHB     0003 R 000000 THETAB     0003 R 000001 THETAS     0003 R 000006 V          0000 R 000000 Y
 0003 R 000017 Y1         0003 R 000020 Y2         0000 R 000024 Z


 00101      1*          SUBROUTINE INTIAL(DLDU,D2LDUU,Y3,Y4,Y5,Y6,B,C,D2E4TU,D2E4LU)
 00101      2*     C
 00101      3*     C    INTIAL CONDITIONS
 00101      4*     C
 00101      5*     C    FOR FIRST AND SECOND DERIVATIVES OF      U SET C=1 AND B=0
 00101      6*     C    FOR FIRST AND SECOND DERIVATIVES OF      V SET C=0 AND B=0
 00101      7*     C    FOR FIRST AND SECOND DERIVATIVES OF FNCBR SET C=1 AND B=1
 00101      8*     C
 00103      9*          DIMENSION Y(20)
 00104     10*          COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
 00104     11*         1             DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DEIDL
 00105     12*          Y(1)    = Y1
 00106     13*          Y(2)    = Y2
 00107     14*          Z       = FNC
 00110     15*          E1      = FM*ETA(Z)+FN
 00111     16*          E2      = THETAB**4-THETAS**4
```

```
00112     17*          E3      = E2/(1-THETAB)
00113     18*          Y(3)    = 0.
00114     19*          DE1DTB  = DNCDT1*(DFM*ETA(Z)+FM*DETA(Z))
00115     20*          DE2DTB  = 4.*(THETAB**3)
00116     21*          DE3DTB  = E3*(DE2DTB+E3)/E2
00117     22*          DE4DTB  = E1*DE3DTB+E3*DE1DTB
00120     23*          DE1DL   = ETA(Z)*DFMX+DFNX
00121     24*          DE1DNC  = ETA(Z)*DFM+DETA(Z)*FM
00122     25*          DE4DU   = E3*DE1DL*DLDU+B*E3*DE1DNC*THETAB**3
00123     26*          Y(4)    = -((1-C)/V**2+DE4DU)/DE4DTB
00124     27*          Y(5)    = 0.
00125     28*          DE2DU   = DE2DTB*Y(4)
00126     29*          DE3DU   = E3*(E3*Y(4)+DE2DU)/E2
00127     30*          D2E2TU  = 12.*Y(4)*(THETAB**2)
00130     31*          A1      = E3*(D2E2TU+DE3DU)/E2
00131     32*          A2      = DE2DTB+E3
00132     33*          A3      = -E3*DE2DU/(E2**2)+DE3DU/E2
00133     34*          D2E3TU  = A1+A2*A3
00134     35*          DFNDU   = DFNX*DLDU
00135     36*          DFCNDU  = DNCDT1*Y(4)+B*Y(2)**3
00136     37*          DFMDU   = DFMX*DLDU+DFM*DFCNDU
00137     38*          DETADU  = DETA(Z)*DFCNDU
00140     39*          DE1DU   = FM*DETADU+ETA(Z)*DFMDU+DFNDU
00141     40*          D2NCTU  = 6*FNCBR*Y(2)*Y(4)+B*3*Y(2)**2
00142     41*          DDETDU  = DDETA(Z)*DFCNDU
00143     42*          D2MNCU  = D2MLNC*DLDU+D2MDNC*DFCNDU
00144     43*          B1      = DFM*DETADU+ETA(Z)*D2MNCU+FM*DDETDU+DETA(Z)*DFMDU
00145     44*          B2      = DE1DTB*D2NCTU/DNCDT1
00146     45*          D2E1TU  = DNCDT1*B1+B2
00147     46*          D2MDLU  = D2FMLL*DLDU+D2MLNC*DFCNDU
00150     47*          D2NDLU  = D2NDLL*DLDU
00151     48*          D2E1LU  = ETA(Z)*D2MDLU+DETADU*DFMX+D2NDLU
00152     49*          D2E4LU  = E3*D2E1LU+DE1DL*DE3DU
00153     50*          D1      = DETA(Z)*DFCNDU*DFM
00154     51*          D2      = ETA(Z)*D2MNCU
00155     52*          D3      = DDETA(Z)*FM*DFCNDU
00156     53*          D4      = DETA(Z)*DFMDU
00157     54*          D2E1CB  = D1+D2+D3+D4
00160     55*          D2E4CB  = DE3DU*DE1DNC+E3*D2E1CB
00161     56*          D2NCNB  = 3*Y(4)*Y(2)**2
00162     57*          TERMNB  = B*(E3*DE1DNC*D2NCNB+Y(2)**3*D2E4CB)
00163     58*          D2E4UU  = E3*DE1DL*D2LDUU+DLDU*D2E4LU+TERMNB
00164     59*          D2E4TU  = E1*D2E3TU+DE3DTB*DE1DU+E3*D2E1TU+DE1DTB*DE3DU
00165     60*          Y(6)    = -(Y(4)*D2E4TU+D2E4UU+2*(C-1)/V**3)/DE4DTB
00166     61*          Y3      = Y(3)
00167     62*          Y4      = Y(4)
00170     63*          Y5      = Y(5)
00171     64*          Y6      = Y(6)
00172     65*          RETURN
00173     66*          END

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS DERV
FOR S9A-07/27/72-19:38:07 (,0)

      SUBROUTINE DERV        ENTRY POINT 000604

      STORAGE USED: CODE(1) 000721; DATA(0) 000130; BLANK COMMON(2) 000000

       COMMON BLOCKS:

       0003    BLKI     000024
       0004    BLKD     000012

      EXTERNAL REFERENCES (BLOCK, NAME)

       0005    ETA
       0006    DETA
       0007    DDETA
       0010    DDDETA
       0011    NERR3$

      STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

       0004 R 000003 A1        0004 R 000004 A2        0004 R 000005 A3        0004 R 000006 A4        0004 R 000007 A5
       0004 R 000010 A6        0000 R 000051 A7        0000 R 000063 DA7DU     0010 R 000000 DDDETA    0007 R 000000 DDETA
       0006 R 000000 DETA      0003   000023 DE1DL     0003   000021 DE4DTB    0003 R 000013 DFM       0003 R 000012 DFMX
       0003 R 000010 DFNX      0003   000007 DNCDT1    0000 R 000052 DNDU      0000 R 000024 DY        0000 R 000053 D2A1UU
       0000 R 000066 D2A2UU    0000 R 000067 D2A3UU    0000 R 000070 D2A4UU    0000 R 000061 D2A5UU    0000 R 000062 D2A6U
       0000 R 000065 D2EEUU    0000 R 000055 D2EPUU    0000 R 000060 D2EUU     0003 R 000016 D2FMLL    0003 R 000015 D2MDNC
       0003 R 000014 D2MLNC    0000 R 000056 D2MUU     0000 R 000054 D2NCUU    0003 R 000011 D2NDLL    0000 R 000064 D2NUU
       0000 R 000057 D3MNUU    0004 R 000001 D3M2LN    0004 R 000002 D3M2NL    0004 R 000011 EB        0005 R 000000 ETA
       0003   000022 E3        0003 R 000002 FM        0003 R 000003 FN        0003 R 000005 FNC       0003 R 000004 FNCBR
       0000   000077 INJP$     0003   000000 THETAB    0003 R 000001 THETAS    0004 R 000000 U         0003   000006 V
       0000 R 000000 Y         0003 R 000017 Y1        0003 R 000020 Y2        0000 R 000050 Z


    00101      1*            SUBROUTINE DERV(Y3,Y4,Y5,Y6,DLDU,D2LDUU,DY3,DY4,DY5,DY6,B,C,
    00101      2*           1               DNCDU,DA1DU,DA2DU,DA3DU,DA4DU,DA5DU,DA6DU,
    00101      3*           2               DEBDU,DEU,DEPU,DMDU,D2MNCU)
    00101      4*      C
    00101      5*      C     DERIVATIVES
    00101      6*      C
    00101      7*      C     FOR FIRST AND SECOND DERIVATIVES OF      U SET C=1 AND B=0
    00101      8*      C     FOR FIRST AND SECOND DERIVATIVES OF      V SET C=0 AND B=0
    00101      9*      C     FOR FIRST AND SECOND DERIVATIVES OF FNCBR SET C=0 AND B=1
    00101     10*      C
    00103     11*            DIMENSION Y(20),DY(20)
    00104     12*            COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
    00104     13*           1          DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DE1DL/
    00104     14*           2      BLKD/U,D3M2LN,D3M2NL,A1,A2,A3,A4,A5,A6,EB
    00105     15*            Y(1)    = Y1
    00106     16*            Y(2)    = Y2
```

```
00107    17*        Y(3)   = Y3
00110    18*        Y(4)   = Y4
00111    19*        Y(5)   = Y5
00112    20*        Y(6)   = Y6
00113    21*        Z      = FNC
00114    22*        EB     = FM*ETA(Z)+FN
00115    23*        A1     = 1./(Y(1)-Y(2))
00116    24*        A5     = FM*DETA(Z)+ETA(Z)*DFM
00117    25*        A6     = 3*Y(2)**2*FNCBR
00120    26*        A2     = A6*A5/EB
00121    27*        A3     = 4*Y(2)**3/(Y(2)**4-THETAS**4)
00122    28*        A4     = A1+A2+A3
00123    29*        DA1DU  = -A1**2*(Y(3)-Y(4))
00124    30*        DNCDU  = A6*Y(4)+B*Y(2)**3
00125    31*        DEPU   = DDETA(Z)*DNCDU
00126    32*        DMDU   = DFMX*DLDU+DFM*DNCDU
00127    33*        D2MNCU = D2MLNC*DLDU+D2MDNC*DNCDU
00130    34*        DEU    = DETA(Z)*DNCDU
00131    35*        DA5DU  = FM*DEPU+DETA(Z)*DMDU+ETA(Z)*D2MNCU+DFM*DEU
00132    36*        DA6DU  = 6*Y(2)*FNCBR*Y(4)+B*3*Y(2)**2
00133    37*        A7     = A6*DA5DU+A5*DA6DU
00134    38*        DNDU   = DFNX*DLDU
00135    39*        DEBDU  = FM*DEU+ETA(Z)*DMDU+DNDU
00136    40*        DA2DU  = (A7-A2*DEBDU)/EB
00137    41*        DA3DU  = A3*Y(4)*(3/Y(2)-A3)
00140    42*        DA4DU  = DA1DU+DA2DU+DA3DU
00141    43*        DY(4)  = -C/A4+U/A4**2*DA4DU
00142    44*        D2A1UU = -A1**2*(Y(5)-Y(6))-2*A1*DA1DU*(Y(3)-Y(4))
00143    45*        D2NCUU = A6*Y(6)+DA6DU*Y(4)+B*3*Y(4)*Y(2)**2
00144    46*        D2EPUU = DDETA(Z)*D2NCUU+DNCDU**2*DDDETA(Z)
00145    47*        D2MUU  = DFMX*D2LDUU+DLDU*(D2FMLL*DLDU+D2MLNC*DNCDU)+DFM*D2NCUU
00145    48*     1           +DNCDU*(D2MLNC*DLDU+D2MDNC*DNCDU)
00146    49*        D3MNUU = D2MLNC*D2LDUU+DLDU*(D3M2LN*DLDU+D3M2NL*DNCDU)
00146    50*     1           +D2MDNC*D2NCUU+DNCDU*D3M2NL*DLDU
00147    51*        D2EUU  = DETA(Z)*D2NCUU+DDETA(Z)*DNCDU**2
00150    52*        D2A5UU = FM*D2EPUU+2*DEPU*DMDU+DETA(Z)*D2MUU+ETA(Z)*D3MNUU
00150    53*     1           +2*DEU*D2MNCU+DFM*D2EUU
00151    54*        D2A6U  = 6*Y(2)*FNCBR*Y(6)+6*FNCBR*Y(4)**2+B*12*Y(2)*Y(4)
00152    55*        DA7DU  = A6*D2A5UU+DA5DU*DA6DU+A5*D2A6U+DA6DU*DA5DU
00153    56*        D2NUU  = DFNX*D2LDUU+D2NDLL*DLDU**2
00154    57*        D2EBUU = FM*D2EUU+2*DEU*DMDU+ETA(Z)*D2MUU+D2NUU
00155    58*        D2A2UU = (DA7DU-A2*D2EBUU-2*DEBDU*DA2DU)/EB
00156    59*        D2A3UU = A3*((3/Y(2)-A3)*Y(6)+Y(4)*(-3/Y(2)**2*Y(4)-DA3DU)
00156    60*     1           +(1/A3*DA3DU)**2)
00157    61*        D2A4UU = D2A1UU+D2A2UU+D2A3UU
00160    62*        DY(6)  = -2/A4*DY(4)*DA4DU+U/A4**2*D2A4UU
00161    63*        DY(3)  = U*(Y(4)-Y(3))+(Y(2)-Y(1))*C
00162    64*        DY(5)  = U*(Y(6)-Y(5))+2*(Y(4)-Y(3))*C
00163    65*        DY3    = DY(3)
00164    66*        DY4    = DY(4)
00165    67*        DY5    = DY(5)
00166    68*        DY6    = DY(6)
00167    69*        RETURN
00170    70*        END

       END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS INTMIX
FOR S9A-07/27/72-19:38:11 (,0)

     SUBROUTINE INTMIX     ENTRY POINT 000024

     STORAGE USED: CODE(1) 000032; DATA(0) 000007; BLANK COMMON(2) 000000

      COMMON BLOCKS:

      0003   BLKI   000024

     EXTERNAL REFERENCES (BLOCK, NAME)

      0004   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0003 R 000023 DE1DL     0000 R 000000 DE4DL     0003 R 000021 DE4DTB    0000 R 000001 DE4DXY    0003   000013 DFM
      0003   000012 DFMX      0003   000010 DFNX      0003   000007 DNCDT1    0003   000016 D2FMLL    0003   000015 D2MDNC
      0003   000014 D2MLNC    0003   000011 D2NDLL    0003 R 000022 E3        0003   000002 FM        0003   000003 FN
      0003   000005 FNC       0003   000004 FNCBR     0000   000002 INJP$     0003   000000 THETAB    0003   000001 THETAS
      0003   000006 V         0003   000017 Y1        0003   000020 Y2


00101      1*          SUBROUTINE INTMIX(D2E4TY,DTBDX,DLDX,D2E4LY,D2LDXY,D2TBXY,D2TFXY)
00103      2*          COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
00103      3*         1         DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DE1DL
00104      4*          D2TFXY  = 0.
00105      5*          DE4DL   = E3*DE1DL
00106      6*          DE4DXY  = D2E4LY*DLDX+DE4DL*D2LDXY
00107      7*          D2TBXY  = -(DE4DXY+DTBDX*D2E4TY)/DE4DTB
00110      8*          RETURN
00111      9*          END

        END OF COMPILATION:        NO  DIAGNOSTICS.
```

177945

```
@FOR,IS MIXDER
FOR S9A-07/27/72-19:38:18 (,0)

   SUBROUTINE MIXDER    ENTRY POINT 000347

   STORAGE USED: CODE(1) 000454; DATA(0) 000104; BLANK COMMON(2) 000000

    COMMON BLOCKS:

    0003   BLKI   000024
    0004   BLKD   000012

   EXTERNAL REFERENCES (BLOCK, NAME)

    0005   DDETA
    0006   DETA
    0007   DDDETA
    0010   ETA
    0011   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0004 R 000003 A1        0004 R 000004 A2        0004 R 000005 A3        0004 R 000006 A4        0004 R 000007 A5
    0004 R 000010 A6        0007 R 000000 DDDETA    0005 R 000000 DDETA     0006 R 000000 DETA      0003   000023 DE1DL
    0003   000021 DE4DTB    0003 R 000013 DFM       0003 R 000012 DFMX      0003 R 000010 DFNX      0003   000007 DNCDT1
    0000 R 000006 DNCDXY    0000 R 000003 D2A1XY    0000 R 000020 D2A2XY    0000 R 000021 D2A3XY    0000 R 000022 D2A4XY
    0000 R 000015 D2A5XY    0000 R 000004 D2A6XY    0000 R 000017 D2EBXY    0000 R 000012 D2EPXY    0000 R 000011 D2EXY
    0003 R 000016 D2FMLL    0000 R 000013 D2MDLY    0003 R 000015 D2MDNC    0000 R 000014 D2MDXY    0003 R 000014 D2MLNC
    0003 R 000011 D2NDLL    0000 R 000016 D2NDXY    0000 R 000007 D3MNLY    0000 R 000010 D3MNXY    0004 R 000001 D3M2LN
    0004 R 000002 D3M2NL    0000 R 000005 D3M2NY    0004 R 000011 EB        0010 R 000000 ETA       0003   000022 E3
    0003 R 000002 FM        0003   000003 FN        0003 R 000005 FNC       0003 R 000004 FNCBR     0000   000026 INJP$
    0003   000000 THETAB    0003   000001 THETAS    0004 R 000000 U         0003   000006 V         0000 R 000000 Y
    0003   000017 Y1        0003 R 000020 Y2        0000 R 000002 Z


  00101      1*           SUBROUTINE MIXDER(DTBDX,DTBDY,DTFDY,D2TBXY,D2TFXY,DNCDX,DNCDY,
  00101      2*           1                    D2TBXE,DA1DX,DA1DY,DA2DX,DA2DY,DA3DX,DA3DY,
  00101      3*           2                    DA4DY,DA5DX,DA5DY,DA6DX,DA6DY,DLDX,DLDY,
  00101      4*           3                    D2LDXY,DEBDY,DEBDX,DEX,DEY,DEPX,DEPY,DMDX,
  00101      5*           4                    DMDY,D2MNCX,D2MNCY,B,C,D3TXYE,DTFDXY)
  00101      6*      C
  00101      7*      C    FOR MIXED DERIVATIVES OF U AND V     SET B=1. AND C= 0.
  00101      8*      C    FOR MIXED DERIVATIVES OF U AND FNCBR SET B=1. AND C= 1.
  00101      9*      C    FOR MIXED DERIVATIVES OF V AND FNCBR SET B=0. AND C= 1.
  00101     10*      C
  00103     11*           DIMENSION Y(2)
  00104     12*           COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
  00104     13*           1             DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DE1DL/
  00104     14*           2        BLKD/U,D3M2LN,D3M2NL,A1,A2,A3,A4,A5,A6,EB
  00105     15*           Y(2)     = Y2
  00106     16*           Z        = FNC
```

```
00107    17*         DTFDXY  = U*(D2TBXY-D2TFXY)+B*(DTBDY-DTFDY)
00110    18*         D2A1XY  = 2*DA1DX*DA1DY/A1-A1**2*(D2TFXY-D2TBXY)
00111    19*         D2A6XY  = 6*FNCBR*DTBDX*DTBDY+6*Y(2)*FNCBR*D2TBXY+C*6*Y(2)*DTBDX
00112    20*         D3M2NY  = DLDY*D3M2NL
00113    21*         DNCDXY  = 6*FNCBR*Y(2)*DTBDX*DTBDY+3*FNCBR*(Y(2)**2)*D2TBXY+
00113    22*        1          C*3*(Y(2)**2)*DTBDX
00114    23*         D3MNLY  = DLDY*D3M2LN+DNCDY*D3M2NL
00115    24*         D3MNXY  = D3MNLY*DLDX+D2MLNC*D2LDXY+D3M2NY*DNCDX+D2MDNC*DNCDXY
00116    25*         D2EXY   = DDETA(Z)*DNCDX*DNCDY+DETA(Z)*DNCDXY
00117    26*         D2EPXY  = DDDETA(Z)*DNCDX*DNCDY+DDETA(Z)*DNCDXY
00120    27*         D2MDLY  = DLDY*D2FMLL+DNCDY*D2MLNC
00121    28*         D2MDXY  = DLDX*D2MDLY+DFMX*D2LDXY+D2MNCY*DNCDX+DFM*DNCDXY
00122    29*         D2A5XY  = FM*D2EPXY+DEPX*DMDY+DEPY*DMDX+DETA(Z)*D2MDXY+DEY*D2MNCX+
00122    30*        1          ETA(Z)*D3MNXY+DEX*D2MNCY+DFM*D2EXY
00123    31*         D2NDXY  = DFNX*D2LDXY+DLDX*DLDY*D2NDLL
00124    32*         D2EBXY  = FM*D2EXY+DMDY*DEX+ETA(Z)*D2MDXY+DEY*DMDX+D2NDXY
00125    33*         D2A2XY  = -(DEBDY*DA2DX+DEBDX*DA2DY+A2*D2EBXY-(A6*D2A5XY+
00125    34*        1          DA6DY*DA5DX+DA5DY*DA6DX+A5*D2A6XY))/EB
00126    35*         D2A3XY  = DA3DX*DA3DY/A3+D2TBXY*DA3DX/DTBDX+
00126    36*        1          A3*DTBDX*(-3*DTBDY/Y(2)**2-DA3DY)
00127    37*         D2A4XY  = D2A1XY+D2A2XY+D2A3XY
00130    38*         D3TXYE  = U*D2A4XY/A4**2-DA4DY*(2*A4*D2TBXE+B)/A4**2
00131    39*         RETURN
00132    40*         END

         END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS ETA
FOR S9A-07/27/72-19:38:27 (,0)

     FUNCTION ETA          ENTRY POINT 000036

     STORAGE USED: CODE(1) 000044; DATA(0) 000022; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003   POLY
      0004   EXP
      0005   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000016 1L        0000 R 000001 A          0000 R 000010 B          0000 R 000000 ETA        0000    000014 INJP$
      0003 R 000000 POLY


00101       1*             FUNCTION ETA(X)
00103       2*             DIMENSION A(7), B(2)
00104       3*             DATA A(1),A(2),A(3),A(4),A(5),A(6),A(7)/0.10E+01, -0.1163143E+01,
00104       4*            1     0.1478836E+01, -0.1267550E+01, 0.6325223E+00, -0.1627067E+00,
00104       5*            2     0.1654223E-01/ B(1),B(2)/0.6866095E+00, -0.2297718E+00/
00116       6*             IF(X.GT.2.5) GO TO 1
00120       7*             ETA   = POLY(7,A,X)
00121       8*             RETURN
00122       9*           1 ETA   = B(1)*EXP(B(2)*X)
00123      10*             RETURN
00124      11*             END

        END OF COMPILATION:        NO  DIAGNOSTICS.
```


177947

```
@FOR,IS DETA
FOR S9A-07/27/72-19:38:44 (,0)

    FUNCTION DETA        ENTRY POINT 000036

    STORAGE USED: CODE(1) 000044; DATA(0) 000021; BLANK COMMON(2) 000000

    EXTERNAL REFERENCES (BLOCK, NAME)

     0003   POLY
     0004   EXP
     0005   NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0001   000016 1L         0000 R 000001 A          0000 R 000007 B          0000 R 000000 DETA       0000    000013 INJP$
     0003 R 000000 POLY


00101       1*           FUNCTION DETA(X)
00103       2*           DIMENSION A(6), B(2)
00104       3*           DATA A(1),A(2),A(3),A(4),A(5),A(6)/-0.1163143E+01, 0.2957672E+01,
00104       4*          1     -0.3802650E+01, 0.2530089E+01, -0.8135335E+00, 0.9925338E-01/
00104       5*          2     B(1),B(2)/-0.1577635E+00, -0.2297718E+00/
00115       6*           IF(X.GT.2.5) GO TO 1
00117       7*           DETA  = POLY(6,A,X)
00120       8*           RETURN
00121       9*         1 DETA  = B(1)*EXP(B(2)*X)
00122      10*           RETURN
00123      11*           END

      END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
@FOR,IS DDETA
FOR S9A-07/27/72-19:39:00 (,0)

   FUNCTION DDETA       ENTRY POINT 000036

   STORAGE USED: CODE(1) 000044; DATA(0) 000020; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   POLY
    0004   EXP
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000016 1L        0000 R 000001 A          0000 R 000006 B          0000 R 000000 DDETA      0000    000012 INJP$
    0003 R 000000 POLY


00101       1*            FUNCTION  DDETA(X)
00103       2*            DIMENSION  A(5), B(2)
00104       3*            DATA  A(1),A(2),A(3),A(4),A(5)/0.2957672E+01, -0.7605300E+01,
00104       4*            1       0.7590267E+01, -0.3254134E+01, 0.4962669E+00/ B(1),B(2)/
00104       5*            2       0.3624960E-01, -0.2297718E+00/
00114       6*            IF(X.GT.2.5) GO TO 1
00116       7*            DDETA  = POLY(5,A,X)
00117       8*            RETURN
00120       9*          1 DDETA  = B(1)*EXP(B(2)*X)
00121      10*            RETURN
00122      11*            END
       END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS DDDETA
FOR S9A-07/27/72-19:39:17 (,0)

   FUNCTION DDDETA       ENTRY POINT 000036

   STORAGE USED: CODE(1) 000044; DATA(0) 000017; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   POLY
    0004   EXP
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000016 1L         0000 R 000001 A          0000 R 000005 B          0000 R 000000 DDDETA     0000    000011 INJP$
    0003 R 000000 POLY


00101       1*             FUNCTION  DDDETA(X)
00103       2*             DIMENSION  A(4), B(2)
00104       3*             DATA  A(1),A(2),A(3),A(4)/-0.7605300E+01,0.1518053E+02,
00104       4*            1-0.9762402E+01,0.1985068E+01/
00104       5*            2 B(1),B(2)/-0.8329136E-02,-0.2297718E+00/
00113       6*             IF(X.GT.2.5) GO TO 1
00115       7*             DDDETA  = POLY(4,A,X)
00116       8*             RETURN
00117       9*           1 DDDETA  = B(1)*EXP(B(2)*X)
00120      10*             RETURN
00121      11*             END

       END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
@FOR,IS POLY
FOR S9A-07/27/72-19:39:21 (,0)

     FUNCTION POLY        ENTRY POINT 000036

     STORAGE USED: CODE(1) 000044; DATA(0) 000015; BLANK COMMON(2) 000000

     EXTERNAL REFERENCES (BLOCK, NAME)

      0003   NERR3$

     STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

      0001   000012 107G      0000   000003 INJP$      0000 I 000002 K        0000 I 000001 L        0000 R 000000 POLY


00101        1*             FUNCTION POLY(N,A,X)
00103        2*             DIMENSION A(N)
00104        3*             POLY = 0.
00105        4*             L = N
00106        5*             DO 1 K=1,N
00111        6*             POLY = POLY*X+A(L)
00112        7*           1 L = L-1
00114        8*             RETURN
00115        9*             END

        END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS SDRV
FOR S9A-07/27/72-19:39:24 (,0)

        SUBROUTINE SDRV       ENTRY POINT 000601

        STORAGE USED: CODE(1) 000605; DATA(0) 000066; BLANK COMMON(2) 000000

         COMMON BLOCKS:

         0003    BLKI     000024
         0004    BLKD     000012
         0005    BLKR     000012
         0006    VAB      000003

        EXTERNAL REFERENCES (BLOCK, NAME)

         0007    ETA
         0010    NCDERV
         0011    MDER
         0012    DETA
         0013    DERV
         0014    MIXDER
         0015    NERR3$

        STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

         0000 R 000003 /AA      0004   000003 A1      0004   000004 A2      0004   000005 A3      0004   000006 A4
         0004   000007 /5       0004   000010 A6      0000 R 000005 BBB     0000 R 000037 DA1DN   0000 R 000007 DA1DU
         0000 R 000023 DA1DV    0000 R 000040 DA2DN   0000 R 000010 DA2DU   0000 R 000024 DA2DV   0000 R 000041 DA3DN
         0000 R 000011 DA3DU    0000 R 000025 DA3DV   0000 R 000042 DA4DN   0000 R 000012 DA4DU   0000 R 000026 DA4DV
         0000 R 000043 DA5DN    0000 R 000013 DA5DU   0000 R 000027 DA5DV   0000 R 000044 DA6DN   0000 R 000014 DA6DU
         0000 R 000030 DA6DV    0000 R 000045 DEBDN   0000 R 000004 DEBDTB  0000 R 000015 DEBDU   0000 R 000031 DEBDV
         0000 R 000046 DEN      0000 R 000047 DEPN    0000 R 000017 DEPU    0000 R 000033 DEPV    0012 R 000000 DETA
         0000 R 000016 DEU      0000 R 000032 DEV     0003   000023 DE1DL   0003   000021 DE4DTB  0003 R 000013 DFM
         0003 R 000012 DFMX     0003   000010 DFNX    0005 R 000006 DLDNCB  0005 R 000001 DLDU    0005 R 000002 DLDV
         0000 R 000050 DMDN     0000 R 000020 DMDU    0000 R 000034 DMDV    0000 R 000036 DNCDN   0003 R 000007 DNCDT1
         0000 R 000006 DNCDU    0000 R 000022 DNCDV   0000 R 000001 D2CD2T  0000 R 000002 D2CTNB  0003 R 000016 D2FMLL
         0005 R 000003 D2LDUU   0005 R 000004 D2LDUV  0005 R 000005 D2LDVV  0005 R 000007 D2LNCB  0005 R 000010 D2LUNB
         0005 R 000011 D2LVNB   0003 R 000015 D2MDNC  0003 R 000014 D2MLNC  0000 R 000051 D2MNCN  0000 R 000021 D2MNCU
         0000 R 000035 D2MNCV   0003   000011 D2NDLL  0004 R 000001 D3M2LN  0004 R 000002 D3M2NL  0004 R 000011 EB
         0006 R 000002 ET       0007 R 000000 ETA     0003   000022 E3      0003 R 000002 FM      0003 R 000003 FN
         0003 R 000005 FNC      0003 R 000004 FNCBR   0000   000056 INJP$   0005 R 000000 RLAM    0003 R 000000 THETAB
         0000 R 000000 THETAF   0003 R 000001 THETAS  0004 R 000000 U       0003   000006 V       0003 R 000017 Y1
         0003 R 000020 Y2       0006 R 000000 Z1      0006 R 000001 Z2


00101      1*          SUBROUTINE SDRV(Y,DY,X)
00103      2*          DIMENSION Y(20),DY(20)
00104      3*          COMMON /BLKI/THETAB,THETAS,FM,FN,FNCBR,FNC,V,DNCDT1,DFNX,D2NDLL,
00104      4*        1      DFMX,DFM,D2MLNC,D2MDNC,D2FMLL,Y1,Y2,DE4DTB,E3,DE1DL/
00104      5*        2      BLKD/U,D3M2LN,D3M2NL,A1,A2,A3,A4,A5,A6,EB
```

```
00104         3       /BLKR/RLAM,DLDU,DLDV,D2LDUU,D2LDUV,D2LDVV,DLDNCB,
00104         4             D2LNCB,D2LUNB,D2LVNB
00104         5       /VAB/ Z1,Z2,ET
00105           Z1       = FNC
00106           Z2       = FM*ETA(FNC)+FN
00107           ET       = ETA(FNC)
00110           THETAF   = Y(1)
00111           THETAB   = Y(2)
00112           Y1       = Y(1)
00113           Y2       = Y(2)
00114            CALL NCDERV(FNCBR,THETAB,FNC,DNCDT1,D2CD2T,D2CTNB)
00115           CALL MDER(RLAM,FNC,FM,DFMX,DFM,D2FMLL,D2MDNC,D2MLNC,D3M2LN,D3M2NL)
00116           AAA      = THETAF-THETAB
00117           DY(1)    = -U*AAA
00120           EB       = FM*ETA(FNC)+FN
00121           DEBDTB   = 3*THETAB**2*FNCBR*(FM*DETA(FNC)+ETA(FNC)*DFM)
00122           BBB      = 4*THETAB**3/(THETAB**4-THETAS**4)
00123           DY(2)    = -U/(1/AAA+DEBDTB/EB+BBB)
00124           CALL DERV(Y(3),Y(4),Y(5),Y(6),DLDU,D2LDUU,DY(3),DY(4),DY(5),
00124         1           DY(6),0.,1.,DNCDU,DA1DU,DA2DU,DA3DU,DA4DU,DA5DU,
00124         2           DA6DU,DEBDU,DEU,DEPU,DMDU,D2MNCU)
00125           CALL DERV(Y(7),Y(8),Y(9),Y(10),DLDV,D2LDVV,DY(7),DY(8),DY(9),
00125         1           DY(10),0.,0.,DNCDV,DA1DV,DA2DV,DA3DV,DA4DV,DA5DV,
00125         2           DA6DV,DEBDV,DEV,DEPV,DMDV,D2MNCV)
00126           CALL DERV(Y(11),Y(12),Y(13),Y(14),DLDNCB,D2LNCB,DY(11),DY(12),
00126         1           DY(13),DY(14),1.,0.,DNCDN,DA1DN,DA2DN,DA3DN,DA4DN,
00126         2           DA5DN,DA6DN,DEBDN,DEN,DEPN,DMDN,D2MNCN)
00127           CALL MIXDER(Y(4),Y(8),Y(7),Y(16),Y(15),DNCDU,DNCDV,DY(4),DA1DU,
00127         1             DA1DV,DA2DU,DA2DV,DA3DU,DA3DV,DA4DV,DA5DU,DA5DV,
00127         2             DA6DU,DA6DV,DLDU,DLDV,D2LDUV,DEBDV,DEBDU,DEU,DEV,
00127         3             DEPU,DEPV,DMDU,DMDV,D2MNCU,D2MNCV,1.,0.,DY(16),
00127         4             DY(15))
00130           CALL MIXDER(Y(4),Y(12),Y(11),Y(18),Y(17),DNCDU,DNCDN,DY(4),
00130         1             DA1DU,DA1DN,DA2DU,DA2DN,DA3DU,DA3DN,DA4DN,DA5DU,
00130         2             DA5DN,DA6DU,DA6DN,DLDU,DLDNCB,D2LUNB,DEBDN,DEBDU,
00130         3             DEU,DEN,DEPU,DEPN,DMDU,DMDN,D2MNCU,D2MNCN,1.,1.,
00130         4             DY(18),DY(17))
00131           CALL MIXDER(Y(8),Y(12),Y(11),Y(20),Y(19),DNCDV,DNCDN,DY(8),
00131         1             DA1DV,DA1DN,DA2DV,DA2DN,DA3DV,DA3DN,DA4DN,DA5DV,
00131         2             DA5DN,DA6DV,DA6DN,DLDV,DLDNCB,D2LVNB,DEBDN,DEBDV,
00131         3             DEV,DEN,DEPV,DEPN,DMDV,DMDN,D2MNCV,D2MNCN,0.,1.,
00131         4             DY(20),DY(19))
00132           RETURN
00133           END

     END OF COMPILATION:        NO  DIAGNOSTICS.
```


177953

```
@FOR,IS SCNTL
FOR S9A-07/27/72-19:40:52 (,0)

  SUBROUTINE SCNTL      ENTRY POINT 000147

  STORAGE USED: CODE(1) 000174; DATA(0) 000014; BLANK COMMON(2) 000000

   COMMON BLOCKS:

   0003   MCT     000005
   0004   VAB     000003
   0005   BLKI    000024

  EXTERNAL REFERENCES (BLOCK, NAME)

   0006   NWDU$
   0007   NI02$
   0010   NERR3$

  STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

   0000   000002 2000F     0001   000054 5L        0001   000061 50L       0001   000135 60L       0003 R 000000 DLMT
   0000 R 000000 DXSTR     0003 R 000003 DXWRT     0004 R 000002 ET        0003 I 000001 ICNT      0000   000006 INJP$
   0003 I 000001 LCNT      0003 I 000004 LSTEP     0005 R 000000 VAL       0003 R 000002 XWRT      0004 R 000000 Z1
   0004 R 000001 Z2


00101      1*              SUBROUTINE SCNTL(Y,DY,DX,X,NTRY,IFVD)
00103      2*              DIMENSION Y(20),DY(20)
00104      3*              COMMON /MCT/ DLMT,ICNT,XWRT,DXWRT,LSTEP
00104      4*             1        /VAB/ Z1,Z2,ET
00104      5*             2        /BLKI/VAL(20)
00105      6*              ICNT     = ICNT+1
00106      7*              IF(DX.GE.DXWRT.AND.ICNT.GT.1) LSTEP = 1
00110      8*              IF(ABS(X-XWRT).LT.DLMT) GO TO 50
00112      9*              IF(XWRT.GT.X) GO TO 5
00112     10*       C
00114     11*              DXSTR    = DX
00115     12*              DX       = DX+XWRT-X
00116     13*              LCNT     = 1
00117     14*              NTRY     = 3
00120     15*              RETURN
00120     16*       C
00121     17*            5 NTRY     = 1
00122     18*              RETURN
00122     19*       C
00123     20*           50  WRITE(6,2000) Y(1),Y(2),Z1,Z2,ET,VAL(3),VAL(4),X
00135     21*         2000 FORMAT (8F15.7)
00136     22*              IF(LCNT.EQ.1) DX = DXSTR
00140     23*              LCNT     = 0
00141     24*              IF(ABS(1.0-XWRT).LE.DLMT) GO TO 60
```

177958

```
00143     25*            XWRT     = XWRT+DXWRT
00144     26*            NTRY     = 1
00145     27*            IF(LSTEP.EQ.0) RETURN
00147     28*            DX       = DXWRT
00150     29*            IFVD     = 1
00151     30*            RETURN
00151     31*      C
00152     32*         60 NTRY     = 2
00153     33*            RETURN
00154     34*            END

       END OF COMPILATION:       NO  DIAGNOSTICS.
```

```
@FOR,IS RKS
FOR S9A-07/27/72-19:41:32 (,0)

   SUBROUTINE RKS        ENTRY POINT 000643

   STORAGE USED: CODE(1) 001040; DATA(0) 000064; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NERR2$
    0004   NEXP5$
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000010 10L      0001   000313 110L     0001   000333 120L     0001   000045 126G     0001   000343 130L
    0001   000071 140G     0001   000355 140L     0001   000105 146G     0001   000130 156G     0001   000417 160L
    0001   000150 164G     0001   000177 174G     0001   000500 185L     0001   000510 190L     0001   000013 20L
    0001   000232 205G     0001   000270 217G     0001   000524 220L     0001   000530 230L     0001   000543 240L
    0001   000374 243G     0001   000032 25L      0001   000552 250L     0001   000554 251L     0001   000572 257L
    0001   000604 259L     0001   000623 270L     0001   000456 300L     0001   000615 336G     0001   000054 40L
    0001   000060 45L      0001   000006 5L       0001   000076 50L      0001   000123 70L      0001   000135 80L
    0000 R 000014 AM       0000 R 000007 AMAX     0000 R 000011 C        0000 R 000010 D        0000 R 000001 DDT
    0000 R 000003 DELT     0000 R 000012 E        0000 R 000000 FR10     0000 I 000004 I        0000 I 000005 IFLAG
    0000   000030 INJP$    0000 I 000002 ISYMP    0000 I 000013 J        0000 R 000006 S


 00101      1*          SUBROUTINE RKS(DERIV,CNTRL,Y,DY,A,R,T,DEL,N,IFVD,IBKP,NTRY,
 00101      2*         1IERR,DELY,PD,SD,YS,YST,DYST,YSIMP)                                  DG000200
 00103      3*          DIMENSION Y(N),DY(N),A(N),R(N),DELY(N),                             DG0   3
 00103      4*         1PD(N),SD(N),YS(N),DYST(N),YST(N),YSIMP(N)                           DG00040
 00104      5*          EXTERNAL  DERIV, CNTRL                                              DG000500
 00104      6*     C    FR10 IS  FIFTH ROOT OF TEN                                          DG00 60
 00105      7*          FR10=1.5848932                                                      DG000700
 00106      8*          IERR=0                                                              DG000800
 00106      9*     C      YS  CONTAINS Y VALUES AT LEFT END POINT OF INTEGRATION INTERVAL   DG00090
 00106     10*     C                                                                        DG001000
 00106     11*     C      YSIMP CONTAINS Y FOR SIMPSONS RULE CHECK  CHECK NOT MADE FOR      #G001100
 00106     12*     C          FIXED STEP MODE    ISYMP IS CONTROL PARAMETER =1,FIXED,2 VARDG00120
 00106     13*     C                                                                        DG001300
 00106     14*     C      IF FIXED STEP SIZE GO ONE INTERVAL OF LENGTH DELT AND RETURN TO   DG00140
 00106     15*     C         CNTRL, IF VAR GO TWO INTERVALS BEFORE RETURN TO CONTRL         DG00150
 00106     16*     C                                                                        DG001600
 00106     17*     C      IFVD   = 0    VARIABLE INTERVAL                                   DG00170
 00106     18*     C             = 1    FIXED                                               DG001800
 00106     19*     C      IBKP   = 0    CUT INTERVAL ONCE BEFORE REPEAT  (UNDER IFVD=0 )    DG00190
 00106     20*     C             = 1    CUT AS REQUIRED                                     DG0 200
 00106     21*     C      NTRY   = 1    CONTINUE INTEGRATING                                DG002100
 00106     22*     C               2    RETURN FROM RKS                                     DG0 220
 00106     23*     C               3    STEP REPEATED WITH NEW DELT                         DG002300
 00106     24*     C               4    RESTART                                             DG00240
```

```
00106   C        IERR  = 0     NORMAL                                       DG002500
00106   C               =1     DELT=0, RETURN FROM RKS                      DG002600
00106   C                1     A(I)+ R(I)*ABS(Y(I)) = 0. , RETURN FROM RKS  DG002700
00107         5 IF(DEL) 20,10,20                                            DG002800
00112        10 IERR=-1                                                     DG002900
00113           GO TO 270                                                   D7003000
00114        20 CALL DERIV(Y,DY,T)                                          DG003100
00115           NTRY=1                                                      DG003200
00116           CALL CNTRL(Y,DY,DEL,T,NTRY,IFVD)                            DG003300
00117        25 DDT=DEL                                                     DG003400
00120           IF(IFVD) 40,30,40                                           DG003500
00123        30 ISYMP=2                                                     DG003600
00124           DELT=DEL/2.                                                 DG00370
00125           DO 31 I=1,N                                                 D7003800
00130        31 SD(I)=0.0                                                   DG00390
00132           IFLAG=1                                                     DG00400
00133           S=1.                                                        DG004100
00134           GO TO 45                                                    DG004200
00135        40 ISYMP=1                                                     DG004300
00136           DELT=DEL                                                    DG00440
00137        45 DO 46 I=1,N                                                 DG004500
00142           YST(I)=Y(I)                                                 DG00460
00143        46 DYST(I)=DY(I)                                               DG0 470
00145        50 DO 60 I=1,N                                                 DG004800
00150           DELY(I)=DELT*DY(I)                                          DG004900
00151           PD(I)=DELY(I)                                               DG00500
00152        60 CONTINUE                                                    DG005100
00154           GO TO (80,70),ISYMP                                         DG005200
00155        70 DO 71 I=1,N                                                 DG005300
00160        71 SD(I)=SD(I)+S*DY(I)                                         DG005400
00162        80 T=T+DELT/2.                                                 DG005500
00163           DO 85 I=1,N                                                 DG005600
00166           YS(I)=Y(I)                                                  DG00570
00167           Y(I)=YS(I)+DELY(I)/2.                                       DG005800
00170        85 CONTINUE                                                    DG005900
00172           CALL DERIV(Y,DY,T)                                          DG006000
00173           DO 90 I=1,N                                                 DG006100
00176           DELY(I)=DELT*DY(I)                                          DG00620
00177           PD(I)=PD(I)+2.*DELY(I)                                      DG006300
00200           Y(I)=YS(I)+DELY(I)/2.                                       DG00640
00201        90 CONTINUE                                                    DG006500
00203           CALL DERIV(Y,DY,T)                                          DG006600
00204           DO 95 I=1,N                                                 DG00670
00207           DELY(I)=DELT*DY(I)                                          DG00680
00210           PD(I)=PD(I)+2.*DELY(I)                                      DG006900
00211           Y(I)=YS(I)+DELY(I)                                          DG00700
00212        95 CONTINUE                                                    DG007100
00214           T=T+DELT/2.                                                 DG007200
00215           CALL DERIV(Y,DY,T)                                          DG00730
00216           DO 100 I=1,N                                                DG007400
00221           DELY(I)=DELT*DY(I)                                          DG007500
00222           PD(I)=PD(I)+DELY(I)                                         DG00760
00223           Y(I)=YS(I)+PD(I)/6.                                         DG00770
00224       100 CONTINUE                                                    DG007800
00226           GO TO (110,120),ISYMP                                       DG007900
00227       110 NTRY=1                                                      DG008000
00230           CALL DERIV(Y,DY,T)                                          DG008100
```


177957

```
00231          CALL CNTRL(Y,DY,DEL,T,NTRY,IFVD)                DG00820
00232          GO TO 300                                       DG008300
00233      120 GO TO (130,140),IFLAG                           DG008400
00234      130 S=4.                                            DG008500
00235          IFLAG=2                                         DG008600
00236          CALL DERIV(Y,DY,T)                              DG008700
00237          GO TO 50                                        D7008800
00240      140 CALL DERIV(Y,DY,T)                              DG008900
00241          AMAX =0.0                                       DG009000
00242          DO 180 I=1,N                                    DG009100
00245          SD(I)=SD(I)+DY(I)                               DG009200
00246          YSIMP(I)=YST(I)+DELT*SD(I)/3.                   DG009300
00247          D   =ABS(Y(I)-YSIMP(I))                         DG00940
00250          C   =A(I)+R(I)*ABS(Y(I))                        DG009500
00251          IF(C    ) 160,150,160                           DG00960
00254      150 IERR=1                                          DG009700
00255          GO TO 270                                       DG009800
00256      160 E   =ABS(D    /C    )                           DG009900
00257          AMAX=AMAX1(AMAX,E)                              DG01000
00260      180 CONTINUE                                        D7010100
00262          IF(AMAX-1.) 215,215,230                         DG01020
00265      215 NTRY= 1                                         DG010300
00266          CALL CNTRL(Y,DY,DEL,T,NTRY,IFVD)                DG010400
00267      300 IF(NTRY-1) 185,185,310                          DG01050
00272      310 IF(NTRY-2) 270,270,330                          DG01060
00275      330 IF(NTRY-3) 340,340,5                            DG01070
00300      340 T=T-DDT                                         DG01080
00301          IF(DEL) 259,10,259                              DG010900
00304      185 GO TO (40,190),ISYMP                            DG01100
00305      190 IF(AMAX-.75) 200,25,220                         DG011100
00310      200 IF(AMAX-.075) 210,25,25                         DG01120
00313      210 DEL=DEL*FR10                                    DG011300
00314          GO TO 25                                        DG011400
00315      220 DEL=DEL/FR10                                    DG011500
00316          GO TO 25                                        DG011600
00317      230 I =1+ IBKP                                      DG011700
00320          GO TO (240,250),I                               DG011800
00321      240 T=T-DEL                                         DG011900
00322          DEL=DEL/FR10                                    DG012000
00323          GO TO  259                                      DG01210
00324      250 J=1                                             DG012200
00325      251 AM=AMAX/10.**J                                  DG012300
00326          IF(1.-AM) 255,257,257                           DG01240
00331      255 J=J+1                                           DG012500
00332          GO TO 251                                       DG01260
00333      257 T=T-DEL                                         DG01270
00334          DEL=DEL/(FR10**J)                               DG01280
00335      259 DO 245 I=1,N                                    DG01290
00340          DY(I)=DYST(I)                                   DG013000
00341      245 Y(I)=YST(I)                                     DG013100
00343          GO TO 25                                        DG013200
00344      270 RETURN                                          DG013300
00345          END                                             DG013400

    END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS FMINV
FOR S9A-07/27/72-19:41:58 (,0)

   SUBROUTINE FMINV      ENTRY POINT 000254

   STORAGE USED: CODE(1) 000306; DATA(0) 001260; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000046 105G     0001   000151 11L     0001   000051 111G     0001   000071 117G     0001   000104 123G
    0001   000125 133G     0001   000130 137G    0001   000172 150G     0001   000175 154G     0001   000227 164G
    0001   000212 20L      0000 R 001214 AA      0000 R 001217 B         0000 I 001212 I        0000   001224 INJP$
    0000 I 001215 I1       0000 I 001220 I2      0000 I 001213 J         0000 I 001216 K        0000 R 000000 XMAT


 00101     1*          SUBROUTINE FMINV (A,X,N,M)                                    1
 00103     2*          DIMENSION A(N,N),X(N),XMAT(25,26)                             2
 00104     3*          DO 1 I=1,N                                                    3
 00107     4*          XMAT(I,M) = X(I)                                              4
 00110     5*          DO 1 J=1,N                                                    5
 00113     6*        1 XMAT(I,J) = A(I,J)                                            6
 00116     7*          DO 20 I=1,N                                                   7
 00121     8*          AA = XMAT(I,I)                                                8
 00122     9*          DO 5 J=1,M                                                    9
 00125    10*        5 XMAT(I,J) = XMAT(I,J)/AA                                      10
 00127    11*          IF (I.EQ.1) GO TO 11                                          11
 00131    12*          I1 = I-1                                                      12
 00132    13*          DO 10 K=1,I1                                                  13
 00135    14*          B = XMAT(K,I)                                                 14
 00136    15*          DO 10 J=1,M                                                   15
 00141    16*       10 XMAT(K,J) = XMAT(K,J) - XMAT(I,J) * B                         16
 00144    17*          IF (I.EQ.N) GO TO 20                                          17
 00146    18*       11 I2 = I+1                                                      18
 00147    19*          DO 15 K=I2,N                                                  1
 00152    20*          B = XMAT(K,I)                                                 20
 00153    21*          DO 15 J=1,M                                                   21
 00156    22*       15 XMAT(K,J) = XMAT(K,J) - XMAT(I,J) * B                         22
 00161    23*       20 CONTINUE                                                      23
 00163    24*          DO 25 I=1,N                                                   24
 00166    25*       25 X(I) = XMAT(I,M)                                              25
 00170    26*          RETURN                                                        26
 00171    27*          END

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

177959

```
@FOR,IS NCDERV
FOR S9A-07/27/72-19:42:07 (,0)

    SUBROUTINE NCDERV    ENTRY POINT 000031

    STORAGE USED: CODE(1) 000044; DATA(0) 000013; BLANK COMMON(2) 000000

    EXTERNAL REFERENCES (BLOCK, NAME)

     0003   NERR3$

    STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

     0000   000004 1IJPS      0000 R 000000 T2        0000 R 000001 T3


00101      1*              SUBROUTINE NCDERV(FNCBR,THETAB,FNC,DNCDT1,D2CD2T,D2CTNB)
00101      2*      C
00101      3*      C       NC AND ITS DERIVATIVES
00101      4*      C
00103      5*              T2       = THETAB**2
00104      6*              T3       = T2*THETAB
00105      7*              FNC      = FNCBR*T3
00106      8*              DNCDT1   = 3.*FNCBR*T2
00107      9*              D2CD2T   = 6.*FNCBR*THETAB
00110     10*              D2CTNB   = 3.*T2
00111     11*              RETURN
00112     12*              END

       END OF COMPILATION:        NO  DIAGNOSTICS.
```

177961

QXQT
MAP 0023-07/27-19:42

ADDRESS LIMITS 001000 021317 040000 047773
STARTING ADDRESS 020263

WORDS DECIMAL 8400 IBANK 4092 DBANK

Segment		IBANK limits		DBANK limits
SEGMENT MAIN		001000 021317		040000 047773
NSWTC$/FOR	1	001000 001021		
NRBLK$/FOR	1	001022 001044		
NRWND$/FOR	1	001045 001124	2	040000 040011
NWEF$/FOR	1	001125 001326	2	040012 040031
NFTCH$/FOR	1	001327 001617	2	040032 040067
NBDCV$/FOR	1	001620 001752	2	040070 040125
NFTV$/FOR	1	001753 001775		
NCLOS$/FOR	1	001776 002144	2	040126 040157
NWBLK$/FOR	1	002145 002266		
NBSBL$/FOR	1	002267 002323		
NUPDA$/FOR	1	002324 002356		
NBFOU$/FOR			2	040160 042361
NCNVT$/FOR	1	002357 002603	2	042362 042451
NIVIN$/FOR	1	002604 003014	2	042452 042463
NOTIN$/FOR	1	003015 003330	2	042464 042467
NOUT$/FOR	1	003331 004317	2	042470 042514
NFMT$/FOR	1	004320 005176	2	042515 042571
NIOER$/FOR	1	005177 005351	2	042572 042676
NFCHK$/FOR	1	005352 006233	2	042677 043035
			4	043036 043107
NTAB$/FOR			2	043110 043146
ERU$/MISC				
NLINP$/FOR	1	006234 007747	2	043147 043330
TIR$/TECH	1	007750 010434	0	043331 043361
			2	043362 043641
NEXP6$/FOR	1	010435 010631	2	043642 043713
ASINCOS$/FOR	1	010632 011046	0	043714 043741
SQRT$/FOR	1	011047 011107	2	043742 043753
EXP$/FOR	1	011110 011177	2	043754 043774
NIER$/FOR	1	011200 011261	2	043775 044130
NOBJF$/FOR	1	011262 011326		
NEXP5$/FOR	1	011327 011414	2	044131 044140
NERR$/FOR	1	011415 011741	2	044141 044304
BLKR (COMMON BLOCK)				044305 044316
BLKD (COMMON BLOCK)				044317 044330
BLKI (COMMON BLOCK)				044331 044354
VAB (COMMON BLOCK)				044355 044357
MCT (COMMON BLOCK)				044360 044364
BLANK$COMMON (COMMON BLOCK)				

```
NCDERV       1   011742 012005     0   044365 044377
                                   2   BLANK$COMMON
FMINV        1   012006 012313     0   044400 045657
                                   2   BLANK$COMMON
RKS          1   012314 013353     0   045660 045743
                                   2   BLANK$COMMON
SCNTL        1   013354 013547     0   045744 045757
             3   MCT               2   BLANK$COMMON
             5   BLKI              4   VAB
SDRV         1   013550 014354     0   045760 046045
             3   BLKI              2   BLANK$COMMON
             5   BLKR              4   BLKD
                                   6   VAB
POLY         1   014355 014420     0   046046 046062
                                   2   BLANK$COMMON
DDDETA       1   014421 014464     0   046063 046101
                                   2   BLANK$COMMON
DDETA        1   014465 014530     0   046102 046121
                                   2   BLANK$COMMON
DETA         1   014531 014574     0   046122 046142
                                   2   BLANK$COMMON
ETA          1   014575 014640     0   046143 046164
                                   2   BLANK$COMMON
MIXDER       1   014641 015314     0   046165 046270
             3   BLKI              2   BLANK$COMMON
                                   4   BLKD
INTMIX       1   015315 015346     0   046271 046277
             3   BLKI              2   BLANK$COMMON
DERV         1   015347 016267     0   046300 046427
             3   BLKI              2   BLANK$COMMON
                                   4   BLKD
INTIAL       1   016270 017011     0   046430 046552
             3   BLKI              2   BLANK$COMMON
MDER         1   017012 017315     0   046553 046640
                                   2   BLANK$COMMON
NDERV        1   017316 017405     0   046641 046655
                                   2   BLANK$COMMON
BNDCND       1   017406 017614     0   046656 046704
                                   2   BLANK$COMMON
CONST        1   017615 020262     0   046705 046775
                                   2   BLANK$COMMON
MAIN         1   020263 021317     0   046776 047773
             3   BLKI              2   BLANK$COMMON
             5   MCT               4   BLKD
                                   6   BLKR

SYS$*RLIB$. LEVEL 63
END OF COLLECTION - TIME 1.955 SECONDS
```