MATHEMATICAL SIMULATION OF THE HUMAN CIRCULATORY SYSTEM

A Dissertation Presented to the Faculty of the Department of Physics University of Houston

In Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy

> By Humphrey Hill Hardy August, 1978

•

MATHEMATICAL SIMULATION OF THE HUMAN CIRCULATORY SYSTEM

.

An Abstract of A Dissertation Presented to the Faculty of the Department of Physics University of Houston

In Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy

> By Humphrey Hill Hardy August, 1978

ABSTRACT

A digital model of the human circulatory system has been developed which simulates pulsatile blood flow and gas transport and exchange. The model was designed specifically to study short term G stress encountered in modern aerial combat manuevers and incorporates a realistic representation of the nonlinear elastic characteristics of circulatory elements and the related pressure dependent flow resistance characteristic of these elements. One form for the pressure-volume relationship used in the model was shown to fit in vivo data for arteries, veins, and left atrium. The resistance-pressure relationship which follows using the Poiseuille-Hagen formula was shown to fit lung data. The oxygen saturation curve and the carbon dioxide dissociation curve were represented by published equations. The systemic circulation is partitioned into four zones: head, upper torso, lower torso, and legs; while the pulmonary circulation is partitioned into six zones with a corresponding distribution of ventilation.

This model has been shown to properly simulate experimental human data for passive breathing in a prone subject. The computed carbon dioxide and oxygen partial pressures vary realistically around measured average partial pressures from human subjects. The computed blood pressure-time, volume-time, and flow-time curves match corresponding curves for human data. Under sinusoidal G_z variations, the model predicts realistic variations of body segment volumes, flows and pressures.

TABLE OF CONTENTS

CHAPTER																						PAGE
I	IN	ΙTF ΕΣ	ROI (1 S	OUC ST I	CT] [N([0] G 1	N A 401	ANI DEI) I LS	REV •	VII •	EW •	01 •		•	•	•	•	•	•	•	1
II	Τŀ	ΙE	MI	ECF	IAN	1I(CS	Oł	7]	BLO	201) I	FLO	WC	•	•	•	•	•	•	•	12
III	p-	-V AS	RI SS I	ELA [GN	AT I IMI	[0] EN [νSŀ Γ.	+11 •	PS •	Al •	ND •	₽ <i>4</i> •	\R/ •	AMI •	ETI	ER •	•	•	•	•	•	25
IV	GA	١S	ΤI	RAN	ISI	201	RT	•	•	•	•	•	•	•	•	•	•	•	•	•	•	46
V	RE	ESU	JL]	٢S	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	59
VI	СС)N(CLU	JSI	[0]	١.	•	•		•	•	•		•	•	•		•	•	•	•	75
APPENDIX	1	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	77
APPENDIX	2	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	86
APPENDIX	3	•		•		•	•		•	•		•	•	•	•	•	•	•	•	•	•	96
APPENDIX	4	•	•		•	•	•	•		•	•	•	•	•	•		•	•		•	•	100
REFERENCE	ES	•				•	•							•	•	•	•	•	•	•		159

•

CHAPTER I

INTRODUCTION AND REVIEW OF EXISTING MODELS

A comprehensive mathematical model of the circulatory system is developed. In order to fully understand the details of its development, we must approach it within a larger framework. It is one facet of a multiphase, long range project for the U. S. Air Force. The goal of the project is to develop a complete cardio-pulmonary mathematical model to study the effects of G stress on man beyond the limits of human centrifuge experiments in the hopes of better providing protection for short term G stresses encountered in modern day aerial combat. The development phase of this project consisted of a three pronged approach: development of a lung model (Ray Calvert, University of Houston, 1976-1978), a circulatory model (this model), and a feedback control system (H.I. Modell, University of Washington, 1977-present). All have incorporated parameters with anatomical correspondences exhibiting nonlinear properties. Each portion of the development required a more comprehensive and detailed approach to the mathematics of human physiology than has been previously attempted. The development and testing of the circulatory model is presented in this thesis.

Due to the nature of the long range project, the circulatory model must meet a number of conditions in its It must incorporate gas transport and a pumping design. heart. It must be complete with verifiable nonlinearities and parameters with physiological correspondences. Tt. must be able to exchange gases with the existing lung model and transport these gases to tissue sites representing different portions of the body. It must exchange gases between lung alveoli and blood, and tissue segments and blood with a physiologically verifiable rate. It must transfer blood, modeling physiologic pressure, volume and flow versus time. Finally, it must provide the ability to change G as a function of time, to incorporate feedback mechanisms, and be able to model a number of seconds of real time with stability, and accomplish all of this in a reasonable amount of computer time.

In order to keep computer time within a workable framework, it is equally important to recognize what the model need not be required to describe. It need not describe the details of the ventricular pulse and arterial wave propagation. It need not provide the data for a detailed study of the uptake of gases by blood plasma and hemoglobin. Finally, it need not model the functioning of the human circulatory system for periods of hours or days.

A number of models of the circulatory system in man have been developed. In the late fifties and sixties,

great strides were made by many researchers in an effort to model the circulatory system. The analog computer was well developed and this was the modeler's medium. The primary hope was that an accurate modeling of the arterial pulse would produce a correlation between pulse shape and heart dynamics, which could then be used as a diagnostic This work reached a peak in 1962 when Circulatory aid. Analog Computers (1) was published from the proceedings of the symposium on the applications of analog computers to the study of the mammalian circulatory system held in the Netherlands on April 19 and 20, 1962. Following this, study turned from the circulation itself to the regulation of the circulation. In 1966, a conference was held on the regulation of the circulation and metabolite transport resulting in the book, Physical Bases of Circulatory Transport: Regulation and Exchange.⁽²⁾ Since that time research in all three areas (arterial pulse propagation, regulation, and gas exchange) has continued independently.

A summary of some of the circulatory models that have been developed is given in Table I (this model is included in the table for comparison). The table indicates the author, date of publication, to what detail the systemic and pulmonary systems have been divided, to what extent physiologic data has been linearized, and the studies performed with the models. From the table, it can be seen that existing models of the circulatory system can be

							TABLE	I			
MODEL AUTHOR	Date(s) of Publication	Only arterial tree	Only systemic circulation	Number of systemic body divisions	Total number of of systemic chambers	Number of pulmonary divisions	Linear P-V or P-d	Resistance Independent of Volume	No gas Transport	Constant blood flow assumed	
Beneken ⁽³⁾	63			1	2	1	x	x	x		Presented only
Noordergraaf ⁽⁴⁻⁷⁾	62- 63	x		0	113	0	x	x	x		Ventricular and arterial pulses, flows, and ballistocardiograms modeled
Robinson ⁽⁸⁾	63			1	2	1	x	x	x		Feedback
Defares ⁽⁹⁾	63			1	2	1	x	x	x		Work of heart, stroke volume and cardiac output
Beneken ⁽¹⁰⁻¹²⁾	64–65 67	i —		4	13	1	x	x	x		Detail of heart contraction, ventricular and arterial pulses and flows, feedback, blood volume changes, valsalvic maneuvers
Grodins ⁽¹³⁾	67	x		1	2	0	x	х	x		Ventricular and arterial pulses, work of heart
Hwang ⁽¹⁴⁾	71	x		1	10	0	x	x	x		Arterial pressure pulse and some feedback
Croston ⁽¹⁵⁾	73			4	23	1	x	x	x		Cardiac output, blood pressures, stroke volume and work rate under exercise conditions
Attinger ⁽¹⁶⁾	68	x		14	49	0		x	x		Ventricular and arterial pulses and flows
Boyers ⁽¹⁷⁾	72			4	4	1	x	х	x		g-stress of 1 g; change of blood volume and some feedback as it affects blood volume and pressure; No heart included

MODEL AUTHOR	Date(s) of Publication	Only arterial tree	Only systemic circulation	Number of systemic body divisions	Total number of of systemic chambers	Number of pulmonary divisions	Linear P-V or P-d	Resistance Independent of Volume	No gas Transport	Constant blood flow assumed	Studies Performed
Green ⁽¹⁸⁾	73		x	1	2	0	x	x	x		g-stress but without a pumping heart
Grodins ⁽¹⁹⁾	67									x	Response to pulsed P_{O_2} and P_{CO_2} . Recovery from O_2 for CO_2 . Hypoxia. Constant blood flow.
Emery ⁽²⁰⁾	71			1	1	1				x	Hyperventilation, and breath holding. Constant blood flow.
Saidel ⁽²¹⁾	71			0	0	1				x	Nitrogen washout and CO uptake. Constant blood flow.
Saidel ⁽²²⁾	72			0	0	2				x	Sinusoidal breathing with and without obstructive lung disease.
This Model	78			4	12	6					

divided into three distinct categories:

1. Those models containing a pumping heart - these are all analog in nature and include Dick and Rideout,⁽²³⁾ Noordergraaf,⁽⁴⁻⁷⁾ Beneken,⁽¹⁰⁻¹²⁾ Hwang,⁽¹⁴⁾ and Croston.⁽¹⁵⁾ These have accurately modeled blood flow under physiologic conditions, usually with a major emphasis on arterial wave pulse propagation. All but one of these analog models consider a constant resistance, unaffected by vessel geometry. Most assume constant pressure-volume relationships. Most also ignore gravitational effects. These existing models also place no emphasis on lung geometry and are therefore inadequate for studies of lung perfusion.

2. Those models addressing the effect of acceleratory stress - only two are known to the author; that of Green⁽¹⁸⁾ and that of Boyers.⁽¹⁷⁾ Boyers' model treats only one g and Green assumes a linear pressure-volume relationship.

3. Those models containing gas transport and diffusion many lung models have some form of circulation. Usually this consists of only an influx of venous blood and an efflux of arterial blood. These are not considered. Those models with a more complete circulatory system include that of Saidel, (21,22) Grodins, (19) and Emery. (20) Of these only Grodins' model has more than one capillary vessel. All assume steady state.

No model, known to the author, falls within any two of these categories. The model developed here addresses all three categories and therefore is the most comprehensive and detailed model known to exist.

In addition, only one of the models has assumed that flow resistances are dependent on the vessel geometry as the well-established Poiseuille-Hagen relationship⁽²⁴⁾ states. All but two of the models assume linear pressurevolume or pressure-diameter relationships, yet all in-vivo data and modern in-vitro data (holding vessel length constant) indicate a logarithmic relationship (see Figures 6, 7, and 8).

The one model with variable resistance is only an arterial model and is therefore not useful to study the entire circulatory system. The assumptions made in the remaining models restrict their reliability to small pressure changes and make them of highly questionable value in studying g-stress. In fact, the question of finding a proper pressure-volume relationship is so important that a digression is in order to review the literature on existing empirical relationships.

An empirical pressure-volume relation, satisfactory for the purpose of a model, must contain relatively few parameters that can be determined from known physiological data, as it is impossible to obtain data for all pressures on every vessel and body segment. Gessner⁽²⁵⁾ proposed the differential relationship;

$$\frac{dV}{dP} = K_0 (1 + a\overline{P} + b\overline{P}^2 + c\overline{P}^3) \quad (1) \quad V = blood \ volume$$
$$K_0, \ a, \ b, \ c = constants$$

where $\overline{P} = P - P_{Ave}$ P_{Ave} = mean transmural blood pressure P = instantaneous transmural blood pressure Another form proposed, by Attinger, ⁽¹⁶⁾ was

P = transmural blood pressure

a, b, c, d, t, g = constants

Both provide reasonably good fits, but the parameters vary widely from vessel to vessel with no physiological basis for their choice. Similarly, the equation proposed by Cope,⁽²⁶⁾

$$\frac{V_{m}}{N} = \frac{T_{s}}{R}(P_{s} - P_{v}) + a(P_{s}^{2} - P_{d}^{2}) + b(P_{s} - Pd)$$
(3)

where P_s = systolic arterial pressure P_d = diastolic arterial pressure P_v = Veneous pressure a,b = constants of the aortic pressure value curve T_s = duration of systole T_d = duration of diastole V_m = cardiac output = minute volume
R = total peripheral resistance
N = heart rate

is not useful as it applies only as a time average over a heart beat and only to the aorta. More recently, $Gaasch^{(27)}$ proposed the relationship

$$P = be^{KV}$$

where P = transmural pressure

V = end-diastolic ventricular volume

b,k = constants

for an end-diastolic pressure-volume relationship in the ventricle, but as pointed out by Glantz, ⁽²⁸⁾ it has serious difficulties. The model predicts that volume becomes zero for positive transmural pressure, and the fit to data becomes poorer as the accuracy of data is improved.

None of these relationships is satisfactory. This thesis presents a satisfactory empirical pressure-volume relationship. This relationship is also used with the Poiseuille-Hagen relationship in the model to give a resulting flow resistance that is dependent on transmural pressure and agrees well with physiological measurements.

The model presented here will provide the best pressure-volume and resistance-volume relationships of any complete circulatory model now available. In addition, it will include the exchange and transport of gases with a

pumping heart. The model includes the following anatomical details to further simulate the human circulatory system:

1. Venous return volume has been separated from the capillary volume so that venous storage without significant O_2 and CO_2 exchanges with tissue is possible.

2. Each lung and body compartment has venous return to provide for the gravitational effect of different body orientations and venous storage.

3. The leg return compartment and values simulate control of circulation and muscle action on circulation within in the legs.

4. The chest cavity compartment allows changes in blood pressure and flow in response to changes of pleural and alveolar pressure due to breathing.

5. Multiple vascular compartments in the lung will simulate distribution of perfusion in the lung.

6. External pressure, P', and resistances at each compartment are variable to simulate short term neural control, g suit effects, and breathing maneuvers.

7. Each portion of the body has a separate control of resistance and compliance allowing vasoconstriction and dilatation in the skeletal muscles while allowing no vasoconstriction in cranial arteries and neither vasoconstriction nor vasodilatation in the lungs. 8. A detailed simulation of heart function which permits studies of such abnormalities as a defective heart valve, high blood pressure, or enlarged heart.

9. The variable resistance of the distensible tubes of the vascular system is adequately modeled.

10. The dependence of compartment pressures on gravity and whole body acceleration is included for all body orientations (e.g., horizontal, reclining, legs up, head down, etc.).

The differential equations of this model have been programmed for numerical integration using Hamming's fourth order predictor-corrector integration with a fourth order Runge-Kutta method starting procedure.⁽²⁹⁾ It consists of 84 coupled, nonlinear, simultaneous, differential equations in its present form (including the transport of O_2 and CO_2 The program is able to include any number of gases only). with the number of equations increasing by 24 for each additional gas. These equations are derived and the determination of model parameters is discussed in the following four chapters. A listing of these equations and parameters is found in Appendix 1; the program itself is in Appendix 4. Appendix 2 consists of a brief discussion of the chemistry of gas transport in the blood, and Appendix 3 lists the empirical saturation and dissociation relationships used in the model.

CHAPTER II

THE MECHANICS OF BLOOD FLOW

This chapter will address the mechanics of blood circulation in the model. The derivation of the blood flow equations will be divided into two parts. Part A will discuss the longitudinal motion of the blood through the blood vessels, and Part B will discuss the transverse motion of blood, vessel wall, and surrounding tissue. Throughout the model, the basic assumption is that many branching blood vessels may be approximated by a single vessel of uniform size. Due to the enormous geometric complexity and number of blood vessels in the body, this assumption is essential if a model is to be developed. The single vessel in turn is mathematically equivalent to a volume element and a resistive element (Figure 1).

These model elements are linked together to form the entire circulatory model (Figures 2, 3 and 4). Only three types of capacitive elements are employed in the entire model. These are enlarged and shown in greater detail in Figure 5. Type a) consists of a blood volume element able to exchange gases with an alveolar space in the lungs; b) is a blood volume able to exchange gas with a tissue volume element; and c) is a blood volume element having no



many branching blood vessels







FIGURE 1



model element



SYSTEMIC CIRCULATION FIGURE 2



Veins



15

•

Key to Chamber Numbers

- Large veins including superior vena cava 1 2 Right Atrium 3 Right Ventricle 2 Left Atrium 3 Left Ventricle Pulmonary arteries 4 5-10 Pulmonary capillaries Pulmonary veins 11 Right Atrium 12 13 Right Ventricle Systemic arteries 14 Systemic capillaries 15-18 19-22 Small veins 23 Veins of the leg Large veins including inferior vena cava 24 The valves are: Tricuspid valve (a) (b) Pulmonary valve (c) Mitral valve (d) Aortic valve
 - (e) Leg valve



٠.

, ·

exchange of gas with exterior elements. Types a) and b) will be referred to as the capillary elements.

Valves are indicated in the model within the heart and leg chambers. The flow of blood through the leg, mitral, and tricuspid valves is modeled by

$$v_{19} \neq 0$$
 if $P_{20} > P_{19}$, value open (5)
 $v_{19} = 0$ if $P_{20} \le P_{19}$, value closed

where v_{19} is the volumetric flow rate of blood from chamber 19 into 20, with pressures P_{19} and P_{20} respectively (see Figure 2, legs and leg return). This allows only one-way flow through the leg values.

The aortic and pulmonary valves also allow only one-way flow, but are modeled in a slightly different way due to the inertia of the blood being included (see Part A of Chapter I). If the aortic or pulmonary valve is closed (v=0), the same conditions apply as in Equation 5. If the valve is open, then the flow is calculated and its sign is checked. If it is positive, calculation continues, if it is negative, the flow is set equal to zero and the valve is closed.

The conservation of blood volume in each chamber is expressed as;

$$v^{\text{in}} - v^{\text{out}} = \frac{dV}{dt} \tag{6}$$

where V is the chamber volume and $v^{in}(v^{out})$ is the volumetric rate of blood flow into (out of) the chamber.

Part A

The Navier-Stokes Equation (30) for incompressible flow can be written

$$\rho \frac{\partial \vec{r}}{\partial t} + \rho \vec{\Omega} \vec{X} \vec{r} + \frac{\rho}{2} \vec{\nabla} r^{2} = - \vec{\nabla} \underline{P} - \rho \vec{\nabla} \phi + \vec{f}_{visc}$$
(7)

where $\vec{\Omega} = \vec{\nabla} X \vec{r}$ (8) and \vec{r} = velocity of blood flow (cm/sec) ρ = density of blood ϕ = potential term \underline{P} = blood pressure \vec{f}_{visc} = viscous force per unit volume of blood

Assuming the circulation and viscous terms $(\vec{f}_{visc} - \rho \vec{\alpha} \vec{X} \vec{r})$ can be written as $\vec{f} v^2$, where \vec{f} is approximately a constant plus a constant divided by the flow rate (f reduces to a constant divided by the flow rate for small Reynolds number), ⁽³¹⁾ the axial component of blood flow becomes

$$\rho \frac{\partial \mathbf{r}_{z}}{\partial t} + \frac{\rho}{2} \frac{\partial}{\partial z} (\mathbf{r}_{x}^{2} + \mathbf{r}_{y}^{2} + \mathbf{r}_{z}^{2}) = -\frac{\partial P}{\partial z} - \rho \frac{\partial \phi}{\partial z} + \overline{f} \mathbf{v}_{z}$$
(9)

The vessel sections are assumed to change size uniformly, so that

$$\frac{\partial \mathbf{r}_{\mathbf{X}}}{\partial z} = \frac{\partial \mathbf{r}_{\mathbf{y}}}{\partial z} = 0 \tag{10}$$

$$\rho \frac{\partial \mathbf{r}_{z}}{\partial \mathbf{t}} = -\frac{\rho}{2} \frac{\partial \mathbf{r}_{z}}{\partial z} - \frac{\partial \underline{P}}{\partial z} - \rho \frac{\partial \phi}{\partial z} + \overline{\mathbf{f}} \mathbf{r}_{z}^{2}$$
(11)

Gravitational forces (or equivalently, centrifugal forces) are the only body forces, so

$$\phi = g_{\chi} \tag{12}$$

where g is the z component of the acceleration (gravitational and/or centrifugal), and ρ is assumed constant (incompressible blood). The derivative $\frac{\partial}{\partial z}$ is approximated as a ratio of finite difference $\frac{\Delta}{\Delta z}$ for the model elements of length Δz . Assuming that the vessel segment is of uniform cross section, inlet and outlet velocities are equal;

$$\mathbf{r}_{z_1} = \mathbf{r}_{z_2} \tag{13}$$

and Equation 11 becomes

$$(\rho \Delta z) \frac{\partial \mathbf{r}_{z}}{\partial t} = (P_{1} + \rho gh_{1}) - (P_{2} + \rho gh_{2}) + Rr_{z} + R'r_{z}^{2} \quad (14)$$

where $fr_{z}^{2} = Rr_{z} + R'r_{z}^{2}$

and
$$P_1(P_2)$$
 = pressure of the blood at inlet (outlet)
 $r_{z_1}(r_{z_2})$ = rate of the blood entering (exiting)
vessel

 $h_1(h_2)$ = height above the heart (chosen as h=0) along the direction of z at inlet (outlet) Δz = vessel length This is the equation used in the model segment connecting the left ventricle and aorta (or arterial chamber) and the right ventricle and pulmonary arterial chamber (see Chapter IV).

The Reynolds number is appreciable only between the ventricles and the arteries at the peak of systole.⁽³²⁾ Throughout the rest of the circulatory system, it is small - the turbulence term, $R'r_z^2$, is therefore dropped for the rest of the model segments.

An additional assumption,

$$(\rho \Delta z) \frac{\partial r_z}{\partial t} = 0, \qquad (16)$$

is made for chambers outside the ventricles and arteries. This assumption is justifiable since the change of flow rate outside these chambers is small compared to the rate of flow under normal conditions. With rapidly changing g, this assumption must be reassessed, but not enough information is available for a conclusion at this time. These assumptions yield the general relationship of axial flow for these vessels:

$$(P_1 + \rho gh_1) - (P_2 + \rho gh_2) = \overline{R}_1 r_z$$
(17)

where

Within the program, the velocity, r_z , is expressed in terms of the volumetric rate of flow, v. This is related to v by

$$Ar_{7} = v \tag{18}$$

where A = uniform cross section of the segment. This expresses the blood flow in ml/sec and because of mass conservation for the incompressible fluid, the change of chamber volume is given by

$$\frac{\mathrm{d}V}{\mathrm{d}t} = v^{\mathrm{in}} - v^{\mathrm{out}}, \qquad (19)$$

where $v^{in} = \sum_{i=1}^{N} v_i$ and $v^{out} = \sum_{i=1}^{M} v_j$ (20)

and

Part B

The assumption is made that the vessel wall and surrounding tissue contributes more to the vessel size and transverse motion than does the transverse blood flow; thus, the equation of transverse blood flow is not discussed. The equation of motion of the vessel wall and surrounding tissue is Newton's second law expressed in terms of pressures,

$$\frac{M}{A}a_{r} = P - P' - \tilde{P} - Rv$$
(21)

where

- M = mass of tissue in motion
- A = cross sectional area of the vessel wall (internal and external areas of the vessel wall assumed approximately equal)
- P = blood pressure inside the vessel

a_r = average transverse acceleration of the vessel wall and surrounding tissue

An additional assumption of instantaneous equilibrium was made by giving

$$P = P' + \tilde{P} + R \frac{dV}{dt}$$
(22)

for the transverse motion, $\frac{dV}{dt}$, of the arterial walls. In order to simplify calculations in the rest of the chambers, v was assumed zero. The justification for the removal of this term, as with the removal of the a_r term, can only be established by operating the model under various conditions and comparing the results to physiologic data. Other models in which the Rv_r term was included have demonstrated that R is very small, but final acceptance of this approximation will await many computer experiments on the model. With these assumptions, the transverse motion of all other chambers is represented as

$$P = P' + \tilde{P}.$$
 (23)

Within the model, v is expressed in terms of the rate of change of the volume chamber associated with the arterial chamber as follows;

$$v = \frac{dV}{dt}$$
(24)

where V = the chamber blood volume in ml.

Equations 14 and 17-24 are adapted for each model chamber as just described in terms of $\frac{dV}{dt}$ and $\frac{d^2V}{dt^2}$. (A complete listing is found in Appendix 1). From initial values of volumes throughout the model and flows through the ventricular chambers, resistances and compliance pressures are found as described in the next chapter. From these, $\frac{dV}{dt}$ and $\frac{d^2V}{dt^2}$ are calculated. A fourth order Haming integration technique with a Runge-Kutta starting procedure is used to integrate the coupled equations sumulatneously to produce new volumes and flows.

CHAPTER III

P-V RELATIONSHIPS AND PARAMETER ASSIGNMENT

The assignment of blood flow-pressure-volume relationships and parameters to be used in the model will be addressed in this chapter. Nonlinear blood pressure-volume and pressure-resistance relationships are found in a unique manner. Parameter assignment, external to the heart, is discussed briefly, and a novel method of setting heart parameters, as used for this model; is presented.

The Pressure-Volume Relationship

for Circulatory Elements

Many measurements of pressure-volume relationships, both in vivo and in vitro, for both venous and arterial elements have been reported in the literature. $(^{33-37})$ Bergel $(^{38})$ has pointed out that in vivo and much in vitro data differ because the in vitro data were taken without holding vessel length constant; here only in vivo data is considered. Using in vivo data from the literature for dogs and man, the author has found that the P-V data for <u>any</u> circulatory element can be fitted very accurately by an equation of the form;

$$V - V_0 = \tau (1 - e^{-k(P - P_0)})$$
(25)

where P_0 , V_0 correspond to an arbitrarily selected pressure-volume point on the curve, k is a constant characterizing the elastic property of the tissue associated with the circulatory element and τ is a constant proportional to the tissue volume associated with the circulatory element.

In this equation, P is transmural pressure and V is blood volume in the element. For blood vessel data reported in terms of vessel diameter, d, we assume that the length, ℓ , of the vessel remains fixed in vivo and hence V is expressed as $\pi d^2 \ell/4$. For atrial data reported in terms of diameter, d, a spherical geometry is assumed so that V is expressed as $\frac{1}{6}\pi d^3$.

Figure 6 exhibits typical fits of this equation to data at two sites for the unexposed femoral vein in a dog while Figure 7 exhibits fits for arteries in man.

It is from fits to data as in Figure 7 that the dependence of the pressure-volume relationship on associated tissue volume (which is proportional to τ) is demonstrated. In this figure the three distinct pressure-volume curves for three subjects become <u>one</u> curve when plotted as normalized volume, $(V-V_0)/\tau$, versus P. In this instance the circulatory element is the arterial system of the leg and the associated tissue volume is the volume of the leg. This dependence on associated tissue volume is also evident



FIGURE 6



in Figure 6 in that the two curves correspond to the same elastance, but different tissue volumes, τ 's, associated with different portions of the femoral vein.

This equation has also been shown to be a good form for representing the observed pressure-volume data for heart chambers as well as arteries and veins; Figure 8 shows this equation fitted to data for the atrium of two dogs. The difference in these two curves is attributed solely to the difference in size of the two animals, again through different τ 's, but this could not be verified directly because the sizes were not reported in the paper⁽³⁷⁾ from which these data were taken.

Finally, using the elastic constant found from Figure 7 and changing the tissue volume to the tissue volume of the whole body, Figure 9 was arrived at for an estimate of the pressure-volume curve of the entire arterial tree. As indicated in the figure, a proper stroke volume is accounted for as a result of the pressure changes observed in the arteries.

It is really not surprising that the pressure-volume relationship for a circulatory element contains a scaling factor, τ , proportional to the tissue volume of the element. The literature⁽³⁹⁾ reveals many cross-species studies showing direct proportionality of the sizes of body organs, for example, to total body volume, and it is this kind of scaling which is revealed here. In particular, Dahn⁽³³⁾



FIGURE 8





FIGURE 9
made note of this scaling by plotting arterial pressurevolume data as in Figure 7.

It is conjectured that the elastic constant k has the same numerical value for all arterial elements of all individuals of all mammalian species and similarly k for all venous elements is the same, but not enough data are available for confirmation. If this is indeed true, then a very important fact has been found for the further application of this circulatory model. For purposes of fixing parameters in the model, this is assumed true.

Assignment of Compliance Parameters

Having established the general form for the pressurevolume relationship, and appropriate values for the elastic constants, k, for venous, arterial and atrial chambers of the circulatory system by curve fits to human and dog data it remained only to establish suitable values for P_0 , V_0 and τ appropriate to each element of the model. P_0 was assigned as the average pressure in an element and V_0 as average blood volume from general anatomical data. The value of tissue volume, τ , was also estimated from general anatomical data.

Flow Resistance

The P-V relationship just discussed is essential to setting flow resistances in the model due to dependence on vessel size. Once the P-V relationship was found, the dependence of resistance on transmural pressure was set using the Poiseuille-Hagen formula

$$R = \frac{\lambda'}{r^4}$$
(26)

with λ^\prime a constant. Assuming cylindrical vessel geometry leads to

$$R = \frac{\lambda}{V^2}$$
(27)

with λ a new constant since vessel length is assumed fixed. Since V(P) is known from the P-V relationship, R(P) follows immediately. The resulting equation for flow resistance as a function of transmural pressure has been fitted to data on a dog's lung shown in Figure 10. This provides excellent validation for the form of this equation and additional confirmation of the fundamental P-V relationship.

Assignment of Resistance Parameters

The resistance parameters outside the heart were assigned for the prone position in the manner indicated in Figure 11. This figure indicates the logical flow of established and estimated information necessary to yield the flow resistance parameters. Once these parameters are set and verified, they will not change under the action of g forces. The setting of resistance parameters inside the heart is discussed in the following section.



FIGURE 10

Fixing Resistance Parameters



Figure 11

Assignment of Heart Parameters

After parameters in other parts of the cardiopulmonary model had been chosen and verified, the heart and aorta posed the most difficult problem due to the type of data available and the number of parameters to be determined. Parameters had to be chosen with very little physiological basis yet these had to make the model produce proper flows and pressures. The force developed by the heart muscle as a function of time must also be specified. A number of physiologic measurements have been made on isolated heart muscle, but this data is not useful in determining the total force of all heart muscle since all muscle tissue does not contract simultaneously within each chamber.

A unique approach to this dual problem of heart muscle force and parameter assignment was found. A model consisting of only the ventricle and arteries was developed to study systole.

A study conducted with this model of the relative importance of the inertial term vs. the resistive term showed that the inertial term significantly affects only the dicrotic notch. That is, with the inertial term the notch is present, without it it is absent, but with negligible effect (~5 mm Hg) on the rest of the arterial pressure-time curve. However, without the resistive term it is impossible to produce an arterial pressure-time curve and a reasonable stroke volume with the same set of parameters. This sub-model was constructed as follows.

A fit to experimental data of ventricular pressure (Figure 12) and derived arterial compliance (Figure 9) was used in the model with different heart resistance, inertial, and turbulence parameters to produce arterial pressure-time and stroke volume data which could be compared to experimental data. Only systole (the time during which the aortic (pulmonary) valve is open) was modeled so that arterial and venous pressures were always greater than 60 mmHg; the mitral (tricuspid) valve was closed; and the change in volume of the ventricle was equal in magnitude to the volumetric flow rate between the ventricle and arterial chambers.

The equations programmed were derived from equations 14, 17 and 21 as follows. Equation 14, written in terms of the volumetric flow rate (Chapter II), was used for the segment connecting the ventricle and arteries

$$(\rho \Delta z) \frac{\partial}{\partial t} \left(\frac{v_1}{A_1} \right) = (P_1 + \rho g h_1) - (P_2 + \rho g h_2) + \overline{R}_1 v_1 + \overline{R}_1 v_1^2 \qquad (28)$$

where

 $v_1 = Ar_z = volumetric flow rate between ventricular chamber$

P₁(P₂) = ventricular (arterial) pressure h₁(h₂) = height above the center of the heart of the ventricular (arterial) chamber

$$R_1 = \overline{R}_1 / A_1 \tag{29}$$







$$R'_{1} = \overline{R}'_{1}/A_{1} \tag{30}$$

The opening between the ventricle and arteries is approximately constant in size, and the cross sectional area of this opening, A, is therefore constant. In addition, for the duration of execution of this model,

$$v_1 = \frac{dV_1}{dt}$$
(31)

where V_1 = ventricular volume in ml In this model, it is assumed that $h_1 = h_2$, simulating a prone subject, giving

$$L_{1} \frac{d_{2}V_{1}}{dt^{2}} = P_{1} - P_{2} + R_{1} \frac{dV_{1}}{dt} + R_{1}' \left(\frac{dV_{1}}{dt}\right)^{2}$$
(32)

where $L_1 = \rho \Delta z / A$

This is the equation programmed for the segment connecting the ventricle and arterial chambers.

Equation 17 leads to the following result with the same assumptions for the segment connecting the arterial and final chambers:

$$P_2 - P_3 = R_2 V_2$$
(33)

where $R_2 = \overline{R}_2/A_2$ $v_2 = A_2r_2$ = volumetric flow rate between arterial and final chambers $h_2 = h_3 = 0$ $P_2(P_3)$ = pressure in the arterial (final) chamber For the chambers themselves, equation 21 gives the following for the arterial chamber, including the inertial term,

$$L_{2} \frac{d^{2}V_{2}}{dt^{2}} = P_{2}^{*} + \tilde{P}_{2}^{-} - P_{2}^{*} + R_{2}^{\frac{dV_{2}}{dt}}$$
(34)
$$L_{2} = \frac{M}{\Lambda^{2}}$$

P'_2 = external pressure P_2 = compliance pressure, Figure 9, (equation and constants given in Appendix 1) P_2 = pressure in the arteries R_2 = resistance to the transverse motion of the arterial wall

Finally, conservation of blood mass (Equation 6) gives

$$v_1 = \frac{dV_1}{dt}$$
(35)

$$v_2 = \frac{dV_2}{dt}$$
(36)

and

where

$$v_3 = \frac{dv_1}{dt} - \frac{dv_2}{dt}$$
(37)

Equations 32 through 37 are coupled, nonlinear differential equations to be solved by a Runga-Kutta integration procedure. The followingare needed as input:

 $v_1(t=0)$ = initial volume of ventricle, from Guyton

 R_3 = average systemic resistance calculated from known average flow rate (Guyton)⁽⁴⁸⁾ and the assumption P_3 =0 (this is approximately the pressure in the right atrium at the end of the systemic circuit).

 $\tilde{P}_2(V_2)$ = derived arterial compliance, Figure 9 (equation and constants given in Appendix 1).

Using these resulting equations and input data, the parameters in the model (R_1 , R'_1 , R_2 , L_1 , L_2) were systematically varied over a range of values. Based on our findings, the inertial term, L_2 , was dropped from the main program resulting in the loss of the dicrotic notch. With some diffuculty and a large amount of computer time it would be possible to include both the resistive and inertial terms and find a set of parameters to match physiologic data including the dicrotic notch, but since the details of heart mechanics are not the main object of our study, only the resistive term was kept with the resulting loss of dicrotic notch. A set of parameters was found from this study which provided a match to the pressure-time curves for aortic pressure and gave a proper stroke volume (as shown by the dotted curve in Figure 13).



;

Only the forcing function remained to be determined. This was done by using the full model with all the results thus far discussed included. Only slight changes in the simplified model parameters were necessary in the full scale model. The resulting parameters and forcing function in the final full scale model are given in Appendix 1.

Due to the findings made in the simplified model, the final form of the ventricle equation (as indicated in Appendix I) is as follows:

a. Ventricle valve open, atrial valve closed.

$$L_{i} \frac{d^{2}V_{i}}{dt^{2}} = PC_{i} - PC_{i+1} + R_{i}'v_{i}^{2} - R_{i} v_{i}$$
(38)

where v_i = flow out of ventricle = rate of change of ventricular volume.

 $PC_i = P_i + \rho gh_i$.

- R_i = resistance of blood to longitudinal motion of ventricle wall.
- V_i = volume of ventricle.
- P_i = pressure in chamber.
- pgh; = hydrostatic pressure.

R! = turbulence constant.

 L_i = inertia of blood exiting ventricle.

b. Atrial valve open, ventricular valve closed.

$$R_{i}\frac{dV_{i}}{dt} = PC_{i}-PC_{i-1}$$
(39)

where
$$R_i$$
 = resistance of blood flow into ventricle.

$$\frac{dV_i}{dt} = \text{rate of change of ventricular volume.}$$

$$PC_i = P_i + \rho gh_i.$$

$$P_i = \text{pressure in chamber.}$$

$$\rho gh_i = \text{hydrostatic pressure.}$$
c. Both valves closed.

$$P = P' + \tilde{P}$$
(40)
where P' = external ventricular pressure.

 \tilde{P} = pressure due to nonlinear compliance.

The atrial and aortic equations, like those of chambers without inertia, become

$$P_{i} + \rho g h_{i} - (P_{i} - 1 + \rho g h_{i-1}) = R_{i-1} V_{i-1}$$
(41)

for the equation of blood flow between chambers.

$$P'_{i} + \tilde{P}_{i} - P_{i} = R^{in}_{i} \frac{dV_{i}}{dt}$$

is the equation of transverse motion of blood and surrounding tissue for atrial, aortic, and ventricular chambers where

$$V_{i-1} - V_i = \frac{dV_i}{dt} .$$

Here P_i = pressure in chamber.

pgh; = hydrostatic pressure.

 R_{i-1} = resistance to flow between chambers.

$$v_{i-1}, v_i$$
 = volumetric flow between chambers.
 $\frac{dV_i}{dt}$ = rate of change of volume of chamber.
 R_i^{in} = resistance to transverse motion of
ventricle wall.

All other chambers have the same form of blood flow equations as the atrial and aortic chambers if R_i^{in} is set equal to zero (See Appendix 1).

CHAPTER IV

GAS TRANSPORT

In addition to the equations and parameters of blood transfer in the body discussed in the previous two chapters, equations of gas transport are also necessary. This chapter will discuss these gas transport equations and the assignment of gas transport parameters.

Gases enter or leave the circulatory system by diffusion to the alveoli, which lead to the outside air through the airways of the lung, or by diffusion into the capillary beds of body tissues where a loss of oxygen and a gain of Carbon dioxide occurs due to the tissue metabolism. Throughout the rest of the circulatory system the gases are merely transported, the total number of moles of each gas being conserved. Within each chamber gas may be gained or lost to other chambers by two processes: blood flow or diffusion. Within all but the capillary chambers, gas may be exchanged only by the influx or efflux of blood containing that gas.

Oxygen is carried in the blood dissolved in plasma and bound to hemoglobin molecules within red blood cells. For the purpose of discussion here, these will be lumped

into a single concentration of Oxygen per unit volume of whole blood and labeled C_{O_2} . Carbon dioxide is carried in the blood as dissolved in the plasma, combined with water to form $HCO_{\overline{3}}$, and bound to the hemoglobin molecules within red blood cells. These will be lumped into a single concentration of carbon dioxide per unit volume of whole blood (C_{CO_2}). These concentrations may be represented in number of molecules of gas per blood volume, mass of gas per blood volume, or volume of equivalent gas at STP per blood volume. The author has chosen the first of these alternatives, using the units of micromoles of gas per milliliter of blood. Most physiologists use the third representation expressing their units as volume percent (the number of milliliters of gas in one milliliter of blood would occupy at STP multiplied by 100). There is a simple conversion factor of 22.41 $\frac{\mu \text{ moles/ml}}{\text{Vol}}$ relating these two units for normal body temperature and pressure.

Perfect mixing in every blood and tissue chamber of this model is assumed. As a consequence, the concentration of each gas in whole blood leaving a chamber is always equal to the concentration of the gas in the chamber. Instantaneous chemical equilibrium is assumed between gases in solution in plasma and hemoglobin within the blood. Instantaneous equilibrium is also assumed in the tissue chambers and a uniform gradient is assumed within the vessel walls through which the gases diffuse. This is

not the case in the real system of course and some studies have addressed this problem.⁽⁴¹⁾ As with the instantaneous equilibrium of the blood, these approximations are assumed to have negligible affect on the system as a whole.

Using these assumptions and Figure 14, it can be seen from the conservation of mass of a particular gas that the rate of change of the number of moles of each gas within a chamber can be expressed as

$$\frac{d}{dt}N_{g} = [Cv]^{in} - [Cv]^{out} - D(P - \overline{P})$$

where \overline{P} = the partial pressure of the gas in the tissue or alveoli (in mmHg) adjacent to capillary chambers

D = the diffusion coefficient in μ moles/sec·mmHg

$$N_g$$
 = the number of moles of gas in each chamber in μ moles

In this equation, $D(P-\overline{P})$ is the rate of diffusion from the chamber into the tissue. In Appendix 2, D is shown to be related to the diffusion constant, S, as follows:

$$D = \frac{\alpha AS}{\ell}$$
(43)

Conservation of mass of gas in vessel chambers





- where α = the ratio of concentration to partial pressure within the tissue.
 - A = the cross sectional area through which diffusion occurs.
 - l = the distance over which the diffusion occurs.

The tissue chambers have an associated metabolic rate (M) for oxygen and carbon dioxide. The tissue chamber equations are therefore;

$$\frac{d\overline{N}}{dt} = D(P - \overline{P}) + M$$
(44)

where \overline{N}_{g} = number of moles of gas in the tissue element in

M = metabolic rate of appearance of O_2 or CO_2 in μ moles/sec. For all other gases, M is zero.

In Equations 42 and 44, D will be zero in all chambers except the capillary chambers.

The gases in the blood can be divided into two categories: those that will combine with hemoglobin and those that will not. The first category consists of oxygen, carbon dioxide, and carbon monoxide. The second consists of all other gases. Due to the fact that oxygen exists primarily only in two states in the blood (bound to hemoglobin and dissolved in the plasma), whereas carbon dioxide exists primarily in three states (bound to hemoglobin, dissolved in plasma, and as HCO_3^-), the gas concentration of whole blood will be expressed differently. In addition, O_2 concentration, CO_2 concentrations, and pH are dependent on each other and must be found simultaneously. The following describes the method.

The volumetric blood flow into and out of every chamber is provided by the portion of the model discussed in Chapter II. The number of moles of O_2 in each chamber is calculated as the sum of the number of moles of dissolved O_2 in the plasma plus the number of moles of O_2 bound to the hemoglobin. The number of moles of O_2 dissolved in the plasma is directly proportional to the partial pressure of O_2 and the volume of plasma. This can be expressed in terms of the hematocrit, H, as follows:

$$N_{O_{2}}^{plasma} = \alpha_{O_{2}}(1-H)VP_{O_{2}}$$
(45)
where $N_{O_{2}}^{plasma}$ = number of moles of O_{2} in the plasma
 H = hematocrit = $\frac{V_{RBC}}{V_{plasma}+V_{RBC}}$
 V_{RBC} = volume of red blood cells
 V_{plasma} = volume of plasma
 $P_{O_{2}}$ = partial pressure of O_{2}
 $\alpha_{O_{2}}$ = solubility factor (related to the Bunsen
solubility coefficient in Appendix III).
 V = volume of whole blood = $V_{RBC}+V_{plasma}$

The number of moles of 0_2 in the plasma and hemoglobin is expressed in terms of the hematocrit since a normal hematocrit is known to be about 0.45.

The number of moles of O_2 bound to the hemoglobin in each chamber is expressed in terms of the saturation of hemoglobin, S_{O_2} . This is the fraction of total O_2 saturation of hemoglobin. Since 100% saturated blood contains .201 volume % of O_2 (8.98 μ moles of O_2 per ml blood), we have

$$N_{O_{2}}^{hemoglobin} = S_{O_{2}}\beta V$$
(46)
where $N_{O_{2}}^{hemoglobin}$ = number of moles of O_{2} bound to hemoglobin
in a chamber of blood volume V
 $\beta = 8.98$ moles O_{2}/ml blood
 $S_{O_{2}}$ = fraction of total O_{2} saturation of hemoglobin
This gives for the total number of moles of O_{2} in each
chamber, $N_{O_{2}}$,

$$N_{O_2} = \alpha_{O_2} (1-H) V P_{O_2} + S_{O_2} \beta V$$
(47)

and for the concentration of $\boldsymbol{O}_2^{}\text{,}$

$$C_{0_2} = \alpha_{0_2} (1-H) P_{0_2} + S_{0_2} \beta$$
(48)

Equation 42 becomes for O_2 ,

$$a \frac{dP_{O_2}}{dt} + b \frac{dP_{CO_2}}{dt} = K_{O_2}$$
(49)

where
$$a = \alpha_{O_2} (1-H)V + \beta V \frac{\partial S_{O_2}}{\partial P_{O_2}}$$
 (50)
 $b = \beta \frac{\partial S_{O_2}}{\partial P_{CO_2}}V$ (51)

and

$$K_{0_{2}} = [C_{0_{2}}v]^{in} - [C_{0_{2}}v]^{out} - D(P_{0_{2}} - \overline{P}_{0_{2}}) - C_{0_{2}}\frac{dV}{dt}$$
(52)

The expression to be used for S_{O_2} is the one by Gomez⁽⁴²⁾ (see Appendix 2). In the process of this calculation, an equation relating pH to the partial pressure of O_2 and CO_2 was necessary and the one used by Kelman⁽⁴³⁾ and West⁽⁴⁴⁾ was chosen (see Appendix 2). Figure 15 shows a comparison of the calculated and measured S_{O_2} for two extreme P_{CO_2} values as a function of P_{O_2} .

For CO_2 , the total number of moles of CO_2 in blood is written in terms of the concentration of CO_2 which is a function of P_{O_2} and P_{CO_2} . Figure 16 shows the comparison of the calculated and measured concentration of CO_2 in volumes % for two extreme P_{O_2} values as a function of P_{CO_2} . The equation used to compute the concentration of CO_2 is adapted from Kelman⁽⁴³⁾ and West.⁽⁴⁴⁾ (See Appendix 2).

For CO_2 , equation 42 becomes

$$c \frac{dP_{O_2}}{dt} + d \frac{dP_{CO_2}}{dt} = K_{CO_2}$$
(53)

where $c = \frac{\partial C_{CO_2}}{\partial P_{O_2}} V$ (54)



FIGURE 15



$$d = \frac{\partial^{C} CO_{2}}{\partial^{P} CO_{2}}$$
(55)

and

$$K_{CO_2} = [C_{CO_2}v]^{in} - [C_{CO_2}v]^{out} - C_{CO_2}\frac{dV}{dt}$$
(56)

Equations 49 and 53 are solved simultaneously, yeilding

$$\frac{dP_{O_2}}{dt} = \frac{dK_{O_2} - bK_{CO_2}}{ad - bc}$$
(57)

$$\frac{dP_{CO_2}}{dt} = \frac{aK_{CO_2} - cK_{O_2}}{ad - bc}$$
(58)

These are solved simultaneously with the saturation function of O_2 , the concentration of CO_2 , and the pH equation using either a Hamming or Runge-Kutta integration procedure.

Equations 44, 57, and 58 are adapted for each model chamber as shown in Appendix 1. From initial values of the partial pressures in all the model chambers, the concentrations of O_2 and CO_2 are calculated for each chamber as just described. These partial pressures and concentrations together with the flows provided by the equations described in Chapter I are used to calculate $\frac{dPO_2}{dt}$ and $\frac{dPCO_2}{dt}$. A fourth order Hamming predictor-corrector integration algorithm is used with a Runge-Kutta starting procedure to integrate these evaluations simultaneously to produce new partial pressures for all of the model chambers.

The subroutines which calculate the concentrations of CO and gases not bound to hemoglobin have not yet been executed within the program. The number of moles of CO bound to hemoglobin will be modeled by a curve fit to the CO saturation curve as was O_2 . The number of moles of CO in the plasma is a constant times its partial pressure. The concentrations of all other gases within the blood are just a constant times the partial pressure of each gas.

Assigning Gas Transport Parameters

The constants for gas flow and diffusion are estimated as shown in Figure 17. This figure shows the logical development of the gas transport parameters. As indicated, some of the parameters are preliminary and further physiologic verification. Steady state and perfect mixing have been assumed so that the mass conservation equations used to assign the parameters are

$$C^{1n}v - C^{out}v = D(P - \overline{P})$$
(59)

and

$$C^{\text{in}} v - C^{\text{out}} v = +M \tag{60}$$

where M = the metabolic rate of disappearance of 0_2 or appearance of CO_2 in tissue.

> v = the volumetric rate of blood flow into or out of the chamber (same under the steady state assumption).

Establishing Concentration Parameters



- (1) Used as a temporary assumption. From literature searches, physiologic measurements of these values are to be found.
- (2) Temporary data from model by Grodins.⁽¹⁵⁾

CHAPTER V

RESULTS

The model described in the last three chapters is really a combination of three models: the circulatory model of the blood alone, the gas transport model, and the lung model. Each of these may be operated separately or together. This was done so that different portions of the complete model may be run separately, thereby saving computer time for specific studies. This chapter on results will address the following model combinations: the circulatory model of the blood alone, the gas transport model with constant alveolar pressures, and finally the complete cardiopulmonary model.

Blood Flow Alone

This model represents only blood transport. The model parameters were selected to simulate a prone resting condition. Figure 18 compares physiologic measurements to model results. The model was run for 5 seconds of real time and then the results were plotted for the following second of real time. Only the artifacts (0, C, 0, C', a) of valvular movements are absent in the model results. This is a direct consequence of the omission of tissue inertia

Pulsatile Blood Flow



FIGURE 18

as discussed in Chapter III. As it is not our purpose to model every detail of heart mechanics, the results given are considered to be in excellent agreement with physiologic measurements.

The differences in stroke volume shown in the figure 18 are due to a heart rate of 70 beats/min in the model whereas these physiological measurements are taken at 75 beats/min.

Figure 19 enlarges the left ventricular pressure and includes the forcing functions used as they occur in the model. Left atrial pressure has also been plotted so that the effect of the forcing function can be seen.

This model was then subjected to a sinusoidal variation of g on the body axis with amplitude 0.5g, with positive acceleration being directed head-to-foot. The effect on capillary blood volumes is shown in Figure 20. The capillary blood volumes resulting from the model run without this variation of g are also plotted for comparison. Notice that the increases in capillary volumes are small, but are clearly the result of the ventricular pressure There is a phase shift between the heart pulse pulses. at the ventricle and the volume change at the capillaries. The symbol, +, is the figure notes the time at which the peak of systole is reached. A similar phase shift can be seen in the variation of capillary volumes due to the applied g. The variations of capillary volume resulting from the model are a result of both g and ventricular





pulses. The 180° phase shift of the head and leg capillary volume was to be expected because gravitational forces tend to reduce (increase) the volume of blood in the head at the same time they tend to increase (reduce) the volume of blood in the legs.

Gas Transport Model

This model is executed with constant blood flow corresponding to the average measured flow of 5 liters/min. The constants used are listed in Appendix 1. Because this model contains all of the qualities of the models reviewed in Table I plus more capillary and systemic chambers, it is by itself an improvement over existing gas transport models with closed circulations.

This model was first executed with constant alveolar partial pressures. Simulation of 30 seconds of real time indicated that equilibrium had been reached in the model. The equilibrium values are compared to physiologic measurements in Table II. All model values seem to be in agreement with physiologic values.

A sinusoidal variation of alveolar pressures was applied to simulate breathing with all initial conditions matching the equilibrium results of Table II. O_2 and CO_2 were varied 180° out of phase. The amplitude of 20 mmHg for both curves was chosen somewhat arbitrarily as this has not been directly measured. The selection of this

TABLE II

Partial Pressures of Oxygen (mmHg)

	Gordon ⁽⁴⁷⁾	Guyton ⁽⁴⁹⁾	Model (<u>Steady State</u>)	Model (passive <u>breathing</u>)
alveoli	103	104	104	109
end-capillary blood	103	104	97.7	103
arterial blood	83	95	93.1	94
venous blood	40	40	40.0	40
tissue	<30	6	6	6

Partial Pressures of Carbon Dioxide (mmHg)

	Gordon ⁽⁴⁷⁾	<u>Guyton</u> (49)	Model (<u>Steady State</u>)	Model (passive <u>breathing</u>)
alveoli	40	40	40	37.7
end-capillary blood	40	40	40.2	37.5
arterial blood	40	40	40.3	39.0
venous blood	47	45	45.0	45.0
tissue	>50	46	46	46.0

value was based rather loosely upon Guyton's difference between inspired and average alveolar $P_{O_2}(P_{CO_2})$ as 45 mmHg (37.7 mmHg) and the difference between expired and average alveolar $P_{O_2}(P_{CO_2})$ as 16 mmHg (13 mm Hg).

Figure 21 shows the model results in selected chambers. First, note that the partial pressure of CO_2 in the capillaries is almost identical to the partial pressure of the alveoli whereas the partial pressure of O_2 in the capillaries is about 7 mmHg below the partial pressure of the alveoli. This is a result of the diffusing capacity of CO_2 being 20 times that of O_2 because of the different molecular weights of the molecules.

Secondly, there is a phase shift between alveoli and capillary partial pressures of both 0_2 and $C0_2$. This is because it takes some time for the diffusion processes to occur across the vessel walls. The partial pressures in the capillaries are not exactly sinusoidal, as there is a larger phase shift in the lower portion of the partial pressure This is a reflection of the shape of the 0_2 and curves. CO_2 dissociation curves as shown in Figure 22. These figures indicate that a sinusoidal variation of the partial pressures of 0_2 and $C0_2$ results in a nonsymmetrical variation of the concentrations. In particular the concentrations corresponding to the lowest partial pressure are further below equilibrium than the concentrations corresponding to the highest partial pressures are above



67

FIGURE 21


equilibrium. In short, more gas molecules flow out than in and thus a longer time is required to reach the lower values of partial pressure.

Finally, the sinusoidal fluctuation in the partial pressures of the alveoli are damped dramatically as the blood moves from one chamber to the next as shown in Figure 21. This effect is due to the assumption of perfect mixing in the model. In fact, by the time that the blood reaches the arteries, there is almost no fluctuation in partial pressures. This effect is more pronounced in the model than the real system because perfect mixing does not occur in the real system.

These studies have demonstrated the proper functioning of the gas transport model.

Gas Transport Plus the Circulatory Model

The next step in the validation of the model was to couple the circulatory and gas transport models together. Blood flow conditions were the same as in the previous discussion since the gas transport has no effect on the blood flow without feedback. The partial pressures produced from this model are shown in Figure 23.

The large, periodic drops in the partial pressure of O_2 in the capillaries shown in the figure are a result of the sudden influx of blood resulting from ventricular contractions. This would be reflected in the body as the



variation in distance capillary blood must travel (assuming plug flow) before it becomes fully oxygenated. This is much less pronounced in P_{CO_2} due to the greater diffusion rate.

Damping is apparent from chamber to chamber as it was in the gas transport model alone. And finally, a slight drift is noticeable in the partial pressure of O_2 in the pulmonary veins. This is due to the nonlinear system approaching a slightly different equilibrium under pulsed flow. This model was not executed for a longer period of time to find this new equilibrium since it is merely an intermediate model.

The Cardio-Pulmonary Model

Once the testing of the intermediate model was completed, the complete model was tested. Due to the volume of data resulting, only the results for a few of the chambers are included.

Figures 24 and 25 show the partial pressures of 0₂ and CO₂ in selected chambers of the lung and circulatory models resulting from the complete model simulating passive breathing. Blood flows, pressure, and volumes are not shown and discussed as they are almost the same as those previously discussed. Breathing does affect blood flows through changes in the transmural blood pressure within the chest cavity. This effect is included in the





FIGURE 25

model, but the variations of the transmural pressures due to passive breathing are small. This effect will become more predominant in forced breathing.

All the observations previously noted in this chapter can be applied to the full model results.

Other than the interesting visual effect of all the variations superimposed, only one new observation should be made. That is that the drift of values has apparently stopped at new, dynamic equilibrium values by the end of 20 seconds of real time. These values are still within the range of physiologic measurements shown in Table II.

CHAPTER VI

CONCLUSION

The results presented in the preceding chapter demonstrate that all the objectives of the circulatory model have been fulfilled. The model incorporates gas transport and a pumping heart. It exchanges gas with the existing lung model and transports these gases. That model equilibrium values match physiologic values indicates that the gases are exchanged between lung alveoli and blood, and tissue segments and blood at a rate which is physiologically realistic. The heart function simulated by the model agrees with physiologic measurements except for valve artifacts. Variations of alveolar gases are reflected in the blood gases in a reasonable way. The model has been tested under a time-varying G stress. The complete model simulation requires about 25 seconds of cpu time on a Honeywell 6660 to simulate one second of real time.

The model has been designed so that it is easily adapted to incorporate feedback methanisms, external body forces, arbitrary breathing maneuvers and gas concentration changes.

As was mentioned in the introduction, this work is part of a long-term project. Studies with the model that are presently being considered include sinusoidal g variation of larger amplitude and a variety of frequencies for comparison to experimental data of Dr. C. F. Knapp (48)at the University of Kentucky: linear increases of g to large sustained values for simulted normal subjects and subjects with borderline abnormalities of the cardiopulmonary system; and, ultimately, simulation of g variation encountered in aerial combat maneuvers. Currently a study of alteration of the ventilation-perfusion distribution in the lungs due to elevated g values is underway. These model studies will complement experimental studies using the human centrifuge at Brooks AFB being carried out by Major James E. Whinnery, Ph.D., M.D.

Not only can the model be used for these studies, but it also can be used as a tool to study disease. For example, arteriosclerosis, emphysema and hypertension are only a few of the possibilities. In fact, any physical disease of the pulmonary and/or circulatory system can be modeled to some degree.

APPENDIX 1

This appendix contains the equations and constants used in the model.

<u>Chamber Equations</u> (all chamber numbers refer to Figures 2, 3 and 4). For all chambers except 2, 3, 4, 12, 13 and 14 the pressures are related as follows:

$$P = P' + \tilde{P}$$
(1-1-a)

where P = blood pressure in the chamber

P' = external pressure

 \tilde{P} = compliance presure (see <u>Compliance Equation</u>)

For chambers 2, 3, 4, 12, 13 and 14 the pressures are related as

$$P = P' + \tilde{P} + R \frac{dV}{dt}$$
(1-1-b)

The remaining chamber equations can be written in the following form:

$$\frac{dV_{m}}{dt} = v_{in} - v_{out}$$
(1-2)

where
$$v_{in} = \prod_{i=1}^{n} v_{I_i}$$

 $v^{out} = \prod_{i=1}^{q} v_{O_i}$
 $\left. \frac{d^P_{O_2}}{dt} \right|_m = \frac{dK_{O_2}^{m} \cdot bK_{O_2}^m}{(ad \cdot cb)}$ (1-3)
 $\left. \frac{d^P_{CO_2}}{dt} \right|_m = \frac{dK_{CO_2}^{m} \cdot bK_{CO_2}}{(ad \cdot cb)}$ (1-4)
where $a = \alpha + \beta \cdot \frac{\partial S_{O_2}}{\partial P_{O_2}}$
 $b = \beta \cdot \frac{\partial S_{O_2}}{\partial P_{O_2}}$, $P_{CO_2} \cdot \gamma$
 $d = \frac{\partial \overline{S}_{CO_2}}{\partial P_{CO_2}} \cdot P_{CO_2} \cdot \gamma + \overline{S}_{CO_2} \cdot \gamma$
 $K_{O_2}^m = (Cv)^{in} \cdot (Cv)^{out} \cdot C_m \frac{dV_m}{dt}$
where $(CCFL)_{I_i} = C_{I_i} v_{I_i} \quad if \quad v_{I_i} \ge 0$
 $C_m v_{I_i} \quad if \quad v_{I_i} < 0$

and
$$(Cv)^{out} = \sum_{i=1}^{q} (CCFL)_{O_i}$$

where $(CCFL)_{O_i} = \frac{C_{O_i}v_{O_i} \text{ if } v_{O_i} \ge 0}{C_m V_{O_i} \text{ if } v_{O_i} < 0}$
 $C_{O_2} = \alpha \cdot P_{O_2} + S_{O_2}\beta$
 $C_{CO_2} = \overline{S}_{CO_2} \cdot \gamma \cdot P_{CO_2}$
 $\alpha = 1.339 \times 10^{-3} \frac{\mu \text{ moles}}{(\text{ml}) \cdot (\text{mmHg})}$
 $\beta = 8.98 \ \mu \text{ moles/ml}$
 $\gamma = 1.00 \frac{\mu \text{ moles}}{(\text{ml}) (\text{mmHg})}$

The subscripts are defined in Table III.

TA]	BLE	Ι	I	Ι

Chamber Number	Number of Connecting links <u>in</u>	Number of Connecting links <u>out</u>			Cham	Conn ber Nu	ectin; umber	g "in"			Cha	Con amber	nnect: Numbe	ing er "o	ut"		
m	n	8	I,	1 ₂	I ₃	I ₄	I ₅	I ₆	I ₇	01	0,	03	04	05	06	07	
1	2	1	21	22	5	-	2	Ŭ	•	2	-	5	-	2	Ū	,	
2	2	1	1	24						3							
3	1	1	2	- •						4							
4	1	7	3							5	6	7	8	9	10	11	
5	1	1	4							11		-		-			
6	1	1	4							11							
7	1	1	4							11							
8	1	1	4							11							
9	1	1	4							11							
10	1	1	4							11							
11	7	1	4	5	6	7	8	9	10	12							
12	1	1	11							13							
13	1	1	12							14							
14	1	4	13							15	16	17	18				
15	1	1	14							19							
16	1	1	14							20							
17	1	1	14							21							
18	1	1	14							22							
19	1	1	15							23							
20	1	1	16							24							
21	1	1	17							1							
22	1	1	18							1							
23	1	1	19							24							
24	2	1	20	23						2							

Segment Equations

For all segments except 3 and 13,

$$R_{i}^{M}v_{i} = P_{i} + \rho gh_{i} - (P_{j} + \rho gh_{j})$$

where

M =	= in	M 🗲	in		
i	<u>i</u>	<u>i</u>	j	i	j
5 6 7 9 10 11 15 16 17 18	4 4 4 4 4 14 14 14 14	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	2 3 4 11 11 11 11 11 11 11 11 11 11 12 13 14 - 19 20	17 18 19 20 21 22 23 24	21 22 23 24 1 24 2

For segments 3 and 13,

$$\frac{M}{A^{2}}\Big|_{i} \frac{dV_{i}}{dt} = (P_{i} + \rho gh_{i}) - (P_{i+1} + \rho gh_{i+1}) - R_{i}v_{i} - R_{i}v_{i}^{2}$$
(NOTE: $\frac{M}{A^{2}}\Big|_{i} = L_{i}$ used in Chapter III)

<u>Tissue Equations</u> (same for all four tissue chambers)

$$\frac{dP_{0_2}^{t}}{dt} = D_{0_2}(P_{0_2} - \lambda_{0_2}P_{0_2}^{t}) - M_{0_2}$$

$$\frac{dP_{CO_2}^t}{dt} = D_{CO_2}(P_{CO_2} - \lambda_{CO_2}P_{CO_2}) + M_{CO_2}$$
$$\frac{dP_{CO}}{dt} = D_{CO}(P_{CO} - \lambda_{CO}P_{CO})$$
$$\frac{dP_x}{dt} = D_x(P_x - \lambda_x P_x)$$

Compliance Equation

Presently
$$P_1 = P_0 + 1/k \ln (1 - \frac{V - V_0}{.224V})$$

for chambers 1, 2, 4, 11, 12, 14, 19, 20, 21, 22, 23, 24 and $P_1 = k(V-V_0)$ for chambers 3, 5, 6, 7, 8, 9, 10, 13, 15, 16, 17, 18.

Segment No. (i)	R(<u>mmHg</u>)				
1 2 3 4 5	.148 .00963 .0024 0. .294	λ is calc in the pr	ulated from ogram using R·V ²	n initial valu g	es of R and V
5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 5 in 6 in 7 in 8 9 in 10 in 11 11 12 13 14 15 16 17 18 19 20 21 22 23 24 5 in 10 11 11 15 16 17 18 19 20 21 22 23 24 5 in 10 11 15 16 17 17 18 19 20 21 22 23 24 5 in 10 11 15 16 17 17 18 19 20 21 22 23 24 5 10 11 15 16 17 11 15 16 17 17 18 19 20 21 22 23 24 5 10 11 12 17 11 15 16 17 11 15 16 17 11 17 18 19 20 21 22 23 24 5 10 11 12 17 11 12 17 10 11 11 12 17 18 19 20 21 22 23 24 5 10 11 11 12 15 16 17 18 19 20 21 22 23 24 5 10 11 11 12 15 16 17 17 18 19 20 21 22 23 24 5 10 11 11 12 22 23 24 5 10 11 11 12 12 12 21 22 12 21 22 11 22 12 21 12 11 11	.294 .294 .294 .294 .294 .033 .019 .012 0. 1.31 .24 1.11 .9 .327 .12 .553 .45 .327 .0608 .588 .588 .588 .588 .588 .588 .588 .5	SEG. NO. (1) 2 3 4 12 13 14	(<u>mmHg</u> (<u>m1/sec</u>) - 1667. - - 333. -	$\frac{\frac{mHg}{(m1/sec)^2}}{3.0x10^{-7}}$	(<u>mmHg</u>) (<u>m1/sec</u>) 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
16 in 17 in 18 in	1.76 8.12 6.62				

Forcing Function Used in Simplified Model (Chapter III)

$$FF = [A/.0219 + c^{-b}] \cdot \frac{a}{2}$$

where

A =
$$\left[\frac{.85(t-t_0)}{a}\right]^2 \cdot \left[1.0 - \frac{.85(t-t_0)}{a}\right]^4$$

b = 749. $\left(\frac{t-t_0-a}{a}\right)^8/a^8$

where a = pulse duration = 0.4 seconds

 t_0 = starting time of the pulse = -.057 seconds

Forcing Function Used in the Complete Model

 $FF = \alpha \cdot c^{-A}$

where α = amplitude of the pulse A = 1/2(t-t₀-a/2)⁴/ σ^2 σ = (a/4)² a = pulse duration t₀ = starting time of the pulse

		t	a	
	1	(sec)	(sec)	<u>(mmHg)</u>
Rt	Atrium	.26	.35	5.0
Rt	Ventricle	.43	.53	20.0
Lt	Atrium	.26	. 35	15.0
Lt	Ventricle	.43	.53	110.0
t _o	reset after a	time of $1/\beta$	where $\beta = pulse$	rate = 70/sec

P _{CO2}	P ^t O2	P ^t CO2	DOO	D _{CO}
<u>mmHg</u>)	(mmHg)	$(\underline{\mathrm{mmHg}})$	(µ moleś/mmHg)	(<u>µ_moles7mmHg</u>)
45.0				
45.0				
45.0				
45.0				
40.2				
40.2	104.	40.0	4.96	99.3
40.2	104.	40.0	4.96	99.3
40.2	104.	40.0	4.96	99.3
40.2	104.	40.0	4.96	99.3
40.2	104.	40.0	4.96	99.3
40.3	104.	40.0	4.96	99.3
40.3				
40.3				
40.3				
45.0	36.8	50.2	6.44	3.17
45.0	36.8	50.2	35.2	17.3
45.0	36.8	50.2	7.63	3.75
45.0	36.8	50.2	9.38	4.62
45.0				
45.0				
45.0				
45.0				
45.0				
45.0				

Chamber	k	P'	τ	PO	v	vo	POn
Number	(1/mmHg)	(mmHg)	(m1)	(mmHg)	<u>(m1)</u>	(m1)	(mmHg)
1	.552	0	358.	4.0	941.0	945.0	40.0
2	.243	-4.	373.	0.	73.39	30.0	40.0
3	.056	-4.	0.	0.	125.0	125.0	40.0
4	.092	-4.	647.	18.0	290.6	265.0	40.0
5	1.00	-4.	0.	0.	23.70	11.70	97.7
6	1.00	-4.	0.	0.	23.70	11.70	97.7
7	1.00	-4.	0.	0.	23.70	11.70	97.7
8	1.00	-4.	0.	0.	23.70	11.70	97.7
9	1.00	-4.	0.	0.	23.70	11.70	97.7
10	.368	-4.	0.	0.	23.70	11.70	97.7
11	.061	-4.	647.	8.0	262.9	256.0	93.1
12	.150	-4.	373.	0.	62.24	30.00	93.1
13	.023	-4.	0.	0.	125.0	125.0	93.1
14	1.00	-4.	644.	8.0	778.9	750.0	93.1
15	1.00	0.	0.	0.	55.20	35.20	40.0
16	1.00	0.	0.	0.	56.20	36.20	40.0
17	1.00	0.	0.	0.	125.2	105.2	40.0
18	.552	0.	0.	0.	20.00	0.	40.0
19	.552	0.	143.	10.0	277.2	277.0	40.0
20	.552	0.	143.	10.0	282.7	278.0	40.0
21	.552	0.	322.	10.0	624.1	625.0	40.0
22	.552	0.	36.	10.0	69.80	70.00	40.0
23	.552	0.	143.	7.0	378.3	377.0	40.0
24	.552	0.	143.	4.0	376.8	378.0	40.0

TABLE IV Constants and Initial Values

APPENDIX 2

This appendix is not essential to the material presented in this thesis, but provides some background to the equations chosen for gas concentrations. This appendix consists of three parts. Part A will discuss briefly the chemistry of oxygen and carbon dioxide transport by the blood. Part B will relate the diffusion coefficient used in the model to the diffusion constant. Part C will relate the Bunsen solubility coefficient to the solubility constant used in the thesis.

Part A

Oxygen is carried in the blood in two states, either dissolved in the blood plasma or bound to the hemoglobin molecule within the red blood cells. Only a small portion of the oxygen carried by the blood is dissolved in the plasma. That which is carried in this form is directly proportional to the partial pressure of the oxygen. The portion of oxygen that is carried by the hemoglobin is bound to the hemoglobin molecule at one of four binding sites. For each molecule of oxygen added to the hemoglobin molecule, the reaction rate is greater than for the preceding molecule. This cooperative binding process

accounts for the characteristic sigmoidal shape of the Oxygen saturation curve (Figure 15) which relates the percent of filled sites versus partial pressure of Oxygen in plasma.

This curve has been found experimentally to vary also as a function of temperature and pH. As this model considers only normal body temperature (37°C), the effect of temperature on the Oxygen saturation curve will not be discussed. The pH of the blood does change within the body, however, because it is directly affected by the amount of Carbon Dioxide dissolved in the blood.

When the blood is exposed to Carbon Dioxide, the following reactions occur:

$$CO_2 + H_2O \rightleftharpoons H_2CO_3$$
 (2-1)

$$H_2CO_3 \longrightarrow H^+ + HCO_3^-$$
(2-2)

$$H^{\dagger} + HbO_2 \rightleftharpoons HHb^{\dagger} + O_2$$
 (2-3)

$$CO_2 + HbO_2 \rightleftharpoons HbCO_2 + O_2$$
 (2-4)

These reactions indicate that Carbon Dioxide can be found in the blood in four states: dissolved, or combined to form H_2CO_3 , HCO_3^- , or $HbCO_2$. Reaction 2-2 goes so far to the right, however, that only a trace of H_2CO_3 remains and essentially all of the Carbon Dioxide can be found in one of three remaining states; normally 7% dissolved, 10% H_2CO_3 , and 60-90% HbCO₂).⁽⁴⁹⁾ The uptake of total CO₂ in all forms by whole blood is found experimentally as shown in the CO₂ dissociation curve (Figure 16) which plots the concentration of CO₂ in whole blood versus the partial pressure of CO₂ in the blood plasma.

These reactions also indicate that two processes tend to cause the loss of O_2 by the hemoglobin molecule in the presence of Carbon Dioxide. One is the buffering of Reaction 2-2 by the hemoglobin (Reaction 2-3) and the other is the displacement of O_2 by CO_2 in the hemoglobin molecule This accounts for the dependence of the (Reaction 4). Oxygen saturation curve on pH and indicates a further dependence (directly, through replacement and indirectly through reaction with water) of the oxygen saturation curve of CO₂. Since all of these reactions are reversible, the CO_2 dissociation is dependent in turn on pH and O_2 concentration in the blood. One could write $C_{0_2}(P_{0_2}, P_{C0_2},$ These functional dependences are incorporated into pH). the equation used to calculate O_2 and CO_2 concentrations within the model.

pH is related to P_{O_2} and P_{CO_2} in the following manner. From Reaction 202,

$$\begin{bmatrix} H^{+} \end{bmatrix} \begin{bmatrix} HCO_{-} \end{bmatrix}^{-} = K'$$
(2-5)

and reaction 2-1

$$\frac{[H_2O] \cdot [CO_2]}{[H_2CO_3]} = K''$$
(2-6)

follows

$$\frac{[H^+][HCO_3]}{[CO_2]} = K$$
(2-7)

assuming the concentration of water to be constant. Rearranging

$$-\log[H^{+}] = \log K + \log([HCO_{3}]/[CO_{2}])$$
 (2-8)

or

$$pH = pK + \log \frac{[HCO_3]}{[CO_2]}$$
 (2-9)

Since the dissolved CO_2 is proportional to the partial pressure of CO_2 , P_{CO_2} equation 2-9 becomes

$$pH = pK + \log \frac{[HCO_3]}{\alpha P_{CO_2}}$$
(2-10)

This relationship is plotted as shown in Figure 26a as P_{CO_2} isobars.

Reaction 2-3 indicates that oxygenated hemoglobin can act as a buffer. The buffering action depends, of course, on the amount of hemoglobin present. Figure 26b shows that experimentally the pH and HCO_3^- are linearly dependent on one another in the presence of hemoglobin. The slope is dependent on the amount of hemoglobin present, as shown in









Figure 26b. Finally, the buffering action of hemoglobin depends on the number of oxygen molecules bound to it. Figure 26c shows the relationship between HCO_3^- and pH for fully oxygenated and fully reduced plasma in contact with hemoglobin. As indicated, the linear relationship remains, but is translated by the absence of O_2 bound to the hemoglobin. These relationships can be expressed mathematically as

$$pH = m[HCO_3] + b \tag{2-11}$$

- where m = slope of lines in Figure 26b and is a function of the hemoglobin concentration in the blood.
 - b = intercept of lines in Figure 26c and is a function of the oxygen saturation of hemoglobin in the blood.

Equations 2-10 and 2-11 provide a set of transcendental equations relating pH to the partial pressure of CO_2 , hemoglobin concentration, and oxygen saturation of hemoglobin. This relationship as approximated by Kelman⁽⁴³⁾ is used in the model.

Part B

The diffusion constant, S, is the ratio of the number of moles of gas per second, $\frac{dn}{dt}$, to $A\frac{dC}{dx}$, where the gas is diffusion through a cross sectional area, A, across a distance dx due to a concentration difference dC.

$$\frac{\mathrm{dn}}{\mathrm{dt}} = -\mathrm{AS}\frac{\mathrm{dC}}{\mathrm{dx}} \tag{2-12}$$

where S is independent of concentration assuming the diffusion molecules are very similar to the molecules through which diffusion occurs. This equation applies to the blood-alveoli and blood-tissue interface. If a uniform concentration gradient in the capillary wall is assumed, the rate of gas flow in or out of the blood is

$$\frac{dn}{dt} = \frac{-AS}{\ell} (C_{w_1} - C_{w_2})$$
(2-13)
where $C_{w_2} (C_{w_1}) =$ concentration of gas in capillary wall
adjacent to the blood (alveoli or tissue)
 $\ell =$ thickness of capillary wall.

This can be written in terms of the partial pressures in the vessel wall by introducing the coefficient α as

$$\frac{\mathrm{dn}}{\mathrm{dt}} = \frac{-\mathrm{AS}}{\ell} (\alpha P_{\mathrm{W}_{1}} - \alpha P_{\mathrm{W}_{2}})$$
(2-14)

where P_{W_2} and P_{W_1} are the partial pressures of gas in the capillary walls and α is defined as the ratio of concentration to partial pressure within the tissue.

Assuming instantaneous equilibrium at the wall faces, it follows that (see Figure 27)

$$P_a = P_{w_1}$$
(2-15)

Concentration





$$P_a = P_{w_2}$$
 (2-16)

and

$$\frac{\mathrm{dn}}{\mathrm{dt}} = \frac{\alpha \mathrm{AS}}{\ell} (\mathrm{P}_{\mathrm{a}} - \mathrm{P}_{\mathrm{b}}) = \mathrm{D} (\mathrm{P}_{\mathrm{a}} - \mathrm{P}_{\mathrm{b}})$$
(2-17)

where P_b and P_a are the partial pressure of gas in the blood and alveoli, or tissue, respectively. Here, the quantity

$$D = \frac{\alpha AS}{\ell}$$
 (2-18)

is the diffusion coefficient used in our model.

Part C

. ..

The Bunsen solubility coefficient, β , is defined as follows

$$\beta = \frac{V_o}{V_s} \cdot \frac{1}{P}$$
(2-19)

where V_0 = volume of gas at STP dissolved in a volume V_S of solute

P = partial pressure of the gas.

This can be related to the ratio of partial pressure to the number of moles of gas per solute used in the thesis with the ideal gas law:

$$PV = nRT$$
 (2-20)

where R = ideal gas constant and V is the volume of n moles of gas at partial pressure P at temperature T.

At STP,

 $P_{O}V_{O} = nRT_{O}$ (2-21)

where
$$P_0 = 760 \text{ mmHg}$$

 $T_0 = 273 \text{ K}$
 $n = \text{number of moles of gas occupying a volume V}_0$
Substituting into Equation ,

$$\frac{{}^{\beta}P}{RT_{o}}P = \frac{n}{V_{s}}$$
(2-22)

Since concentration, C, is defined as the number of moles per volume of solvent, Equation 2-22 becomes

$$\frac{\beta P_{o}}{RT_{o}}P = C = \alpha P \qquad (2-23)$$

$$\alpha = \frac{\beta P_{o}}{RT_{o}} \qquad (2-24)$$

Thus

APPENDIX 3

This appendix consists of the empirical Oxygen saturation curve, Carbon Dioxide dissociation curve, and their derivatives with respect to P_{O_2} and P_{CO_2} . Note that definitions of symbols carry throughout this appendix.

set y = 0.0
calculate pH

Concentration of Carbon Dioxide C_{CO2}

$$C_{CO_2} = S_{CO_2} \cdot P_{CO_2}$$

where
$$S_{CO_2} = (1-H) \cdot C_p + H \cdot d_1 \cdot C_p$$

 $H = hematocrit = .45$
 $C_p = 0.0307 \cdot (1+Y')$
 $Y' = exp[2.3961 \cdot pH - 14.7162]$
 $d_1 = d_2 + (d_3 - d_2) \cdot (1 - S_{O_2})$
 $d_2 = .59 + .2913(7.4 - pH)$
 $d_3 = .664 + .2275(7.4 - pH)$

Rate of Change of the Oxy-Hemoglobin Saturation function with Respect to the Partial Pressure of Oxygen

 $\frac{\partial S_{O_2}}{\partial P_{O_2}}$

$$\frac{\partial S_{O_2}}{\partial P_{O_2}} = \frac{A_1 C}{(1+V)^2 + A_2}$$

$$A_2 = .045A_3$$

$$A_3 = A_1 \cdot P_{O_2} \cdot (.010377) \cdot z$$

$$A_1 = .925 + 5.6V + 90V^2$$

Rate of change of the Oxy-Hemoglobin Saturation	٥S
Function with Respect to the Partial Pressure	⁰ 2
of Carbon Dioxide,	⁹ P _{CO}
	^{CO} 2

$$\frac{\partial SO_2}{\partial P_{CO_2}} = \frac{-A_3 \cdot .2741}{(1+V)^2 + A_3 \cdot .045} \frac{1}{P_{CO_2}}$$

Rate of Change of the Concentration	of ^{°C} co
Carbon Dioxide with Respect to the	2
Partial Pressure of Oxygen,	⁹ P0
	⁰ 2

$$\frac{{}^{\partial C}CO_2}{{}^{\partial P}O_2} = \frac{{}^{\partial S}CO_2}{{}^{\partial P}O_2} P_{CO_2}$$

where
$$\frac{\partial S_{CO_2}}{\partial P_{O_2}} = (1-H)C_P^{+}+Hd_1^{+}C_P^{+}+Hd_1^{+}C_P^{+}$$

 $C_P^{+} = .0307\overline{Y}^{+}$
 $\overline{Y} = 2.396 \cdot 6^{+} \cdot Y^{+}$
 $d^{+} = d_2^{+} + (d_3^{+} - d_2^{+})(1-S_{O_2}^{-}) + d_2^{-}d_1)(-S_{O_2}^{+})$
 $d_2^{+} = -.2913b^{+}$
 $d_3^{+} = -.2275b^{+}$
 $b^{+} = -.045\frac{dS_{O_2}}{dP_{O_2}}$

Rate of Change of the Concentration of	⁹ C _{CO}
Carbon Dioxide with Respect to the	^{CO} 2
Partial Pressure of Carbon Dioxide	⁹ P _C O
	202

$$\frac{{}^{\partial C}CO_2}{{}^{\partial P}CO_2} = S_{CO_2} + \frac{{}^{\partial S}CO_2}{{}^{\partial P}CO_2} P_{CO_2}$$

where
$$\frac{\partial S_{CO_2}}{\partial P_{CO_2}} = (1-H)\overline{C}_P + H \cdot \overline{d}_1 \cdot C_P + H \cdot \overline{d}_1 \cdot \overline{C}_P$$

 $\overline{C}_P = .0307f$
 $f = 2.3961 \cdot Y' \cdot \overline{D}$
 $\overline{d}_1 = \overline{d}_2 + (\overline{d}_3 - \overline{d}_2)(1 - S_{O_2}) - (d_3 - d_2)\frac{\partial S_{O_2}}{\partial P_{CO_2}}$
 $\overline{d}_2 = -.2913\overline{D}$
 $d_3 = .2275\overline{D}$
 $\overline{b} = -.045\frac{\partial S_{O_2}}{\partial P_{CO_2}} - .2741/P_{CO_2}$

APPENDIX 4

This appendix lists the subroutines and their purposes followed by a complete computer listing of the programs. The outlining of the subroutines will not include the details of the lung program. A flow chart describing the order of calculation is shown in Figure 27. BMAIN--program of pumping heart and blood flow only

(circulatory portion of the model only)

BMAIN--calling routine controlling storage, subroutines, time incrementation, and I.O. Subroutines of BMAIN BDRIVS--calculates flows, derivatives of volumes, and controls the leg valve. RESIST--calculates resistance of all segments. Options available are LOPR=neg makes $R = k/(V^2+V)$. This is to model non-Newtonian flow in the capillaries if desired. Not presently in use.

LOPR = 0 makes R = constant

LOPR = pos. makes $R = d/V^2$ in accordance with

the Poiseuille-Hagen law.

PHEART--Calculates the forcing function of the heart. PCHART--calculates pressures and valve dispositions within the heart.





Figure 27

SKON--calculates compliance pressures. Options
available are LOPP1 = 0 means compliance constant.
LOPP1 ≠ 0 makes compliance a function of volume.
(See Appendix 1)
HAMING AND RUNGE--calculates advancement of all
independent variables in time.

Calling structure:

BMAIN

PHEART RESIST SKON BDRIVS HAMING RUNGE SKON PCHART

BCC--program of gas exchange with constant blood flow

BCC--(Same as BMAIN) Subroutines of BCC BCDRIV--calculates derivatives of tissue concentrations. Options are; LOPGS = 1 execute for only the gases O_2 and CO_2 LOPGS = 2 execute for only the gases O_2 , CO_2 , and X LOPGS = 3 execute for only the gases X LOPGS = 4 execute for only the gases O_2 , CO_2 , and CO
LOPGS = 5 execute for all the gases O_2 , CO_2 , CO, and X where X is one or more gases that do not bind to hemoglobin O2DRIV--calculates the $\frac{dPO_2}{dt}$, $\frac{dPCO_2}{dt}$, and $\frac{dPCO}{dt}$ for all chambers. ($\frac{dP_{CO}}{dt}$ is not calculated if LOPGS.LT.4). XDRIV--calculates the $\frac{dx}{dt}$ for all chambers (is not called unless LOPGS = 2, 3, or 5) SATUR--calculates S_{O_2} , C_{CO_2} , $\frac{dSO_2}{dPO_2}$, $\frac{dC_{CO_2}}{dPO_2}$, and $\frac{dC_{CO_2}}{dPO_2}$. GFLTST--calculates K_{O_2} and K_{CO_2} for all chambers. FLTEST--determines the direction of flow and (Cv)ⁱⁿ and (Cv)^{Out} for each chamber

Calling Structure

BCC

BCDRIV

Subroutines of BMAIN

O2DRIV XDRIV

SATUR GFLTST GFLTST

FLTEST FLTEST

<u>BCMAIN</u>--program of gas exchange with pumping heart (circulatory model with gas transport) BCMN--same as BMAIN. Subroutines of BCMAIN consist of all those in BMAIN plus all those in BCC.

MAIN--complete cardio-pulmonary model to date.

MAIN--controls the execution of the blood program (BCMAIN) and lung program (LMAIN) so that they may be executed with different time steps and durations. Subroutines in MAIN.

BCMAIN--circulatory model with gas transport with various entry points added.

LMAIN--complete lung program with various entry points added

BLCONV--calculates the rate of removal of each gas from each alveoli chamber

LBCONV--calculates external pressures on blood chambers using alveoli pressures and pleural pressures from the lung program and converts each gas concentration to each gas partial pressure in each alveoli. Calling Structure;

MAIN

BCMAIN

LMAIN

BLCONV

LBCONV

```
10C
       SUBROUTINE MAIN
        INTEGER HAMING, OPT1, OPT2, FBOPT
20
30
        IMPLICIT REAL*3 (A-H.O-Z)
40
        REAL*d LAMBO2.LAMBC2.LAMBCO,LAMBX,LCO2,LCC2,LCC0,LCx,METC2,METC2
50
               N. NW
       s.
        DIMENSION R(24), FKDA(24), PP(24), V(24), VO(24), H(24), DV(24).
bЭ
70
       1VFLOW(24), VFLIN(18), RIN(18), PC(24), P(24)
80
        DIMENSION FCTE(00), DFCTB(d6), FCTSB(86,4), DFCT3B(66,3), TEB(05)
90
        DIMENSION GPO2(24), GPC2(24), GPCO(48), GPX(1,24), CTO2(4), CTC2(4),
100
        1CTCO(4),CTX(1,4),DIF02(10),DIFC2(10),D1FCO(10),D1FX(10),CPh(24),
        2LCO2(6),LCC2(0),LCCO(0),LCX(1,0),CSATO2(24),CSATC2(24),D3P32(24),
110
120
        3DGPC2(24), DGPCC(45), DGPX(1,24), AX(1)
150
         DIMENSION DCT02(4).DCTC2(4).DCTC0(4).DCTX(1.4).HeI02(4).HeIC2(4)
140C
       THIS DIMENSION STATEMENT IS ONLY IN EMAIN
150
         DIMENSION TE(62), FCT(62), FCTS(62,4), DFCT(62), \nuFCTS(62,5)
160
         COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
170
        &
             F(12), G, GS, NG, DNT(12), NW(12), DNW(12), PHD(12), PLS,
             DPES, PELALV(6), DPELAL(6), PELDA, DPELDA, PELIX, DPELIX,
180
        Če .
190
        æ
             PG(6), DPG(6), PGAE, DPGAB, PPL, DPS(6), DPS2,
             S(4), T, DT, IN(12), TOTN(12), VHD(12), VD, VLUNG.
200
        X-
210
        è
             SCT(12), L2(12), HDT
         COMMON /KN1/KN16
220
230
         COMMON /BCM/ 1E.TMAXB
240
         COMMON /LINN/ KNT
250
         COMMON /CMM1/V, VP2, VP3, VP4, VP12, VP13, VP14, GP02, CT02,
200
        &GPC2.C1C2
270
         COMMON /CHANIT/GPCO, CTCO, GPX, CTX
260
         COMPON /CMA2/DV.DVP2.DVP3.DVP4.DVP12.DVP13.DVP14.DGP02.DCT02.
290
        1DuPC2, DUTC2
300
         CONTION /CMN2T/DGPCO.DCTCO.DGPX.DCTX
310
         COMMON /BLK1/ N(12,3), VTX, VALV(6), PALV(6), SW(12), VSPR
         CUMMUN /BLK2/ DN(12,3), DVTX, DVALV(6), DPALV(0), DSW(12), DVER
320
         EQUIVALENCE (V, FCIE, FCTSE), (DV, DFCTE, DFCTSE)
350
340
        δċ.
                     ,(N(1), FCT(1), FCTS(1)), (DN(1), DFCI(1), DFCI:(1))
      FCTE(V=24, VP=0, C=24, CT=4)
350C
300
         READ (5,2) KNTTST, CONV, RIC
570
       2 FORMAT (15.2E10.3)
300
         CALL BCHAIN(LDUMMY)
390
         CALL LMAIN(LDUMMY)
400
         CALL LECONV(RTC)
410
         IF (KNTTST.LT.O) GO TO 100
420
      10 CALL BCMN1(LDUMMY)
450
         CALL ELCONV(CUNV)
440
         CALL LENT(LDUMEY)
450
         CALL LECONV(RIC)
400
      20 CONTINUE
470
         1F (T-TB.LT.1.E-04) GO TO 35
         IF (KNTB.GT.3) CALL BCMN3(LDUMMY)
460
490
         IF (KNTB.LE.5) CALL BCMN2(LDUMMY)
500
      30 CALL BLCONV(CONV)
```

```
510
         IF (18-T.LT.1.E-04) GO TO 45
520
      35 CONTINUE
530
         IF (KNT.GT.3) CALL LMN3(LDUMMY)
540
         IF (KNT.LE.3) CALL LMN2(LDUMMY)
550
      40 CALL LBCONV(RTC)
560
      45 IF (TB-TMAXB) 20,50,50
570
      50 CONTINUE
580
         CALL LMN4(LDUMMY)
590
     100 CONTINUE
600C
       REVERSE GRDER OF CALL BCMN AND LMN'S AND ADJUST KNTTST
610C
         STATEMENTS FOR NEG. KNTTST.
620
         STOP
630
         Ê ND
640
         SUBROUTINE LECONV(RTC)
650
         IMPLICIT REAL*d (A-H.O-Z)
660
         REAL*5 LAMEO2.LAMECO.LAMEX.LCO2.LCC2.LCCO.LCX.METO2.METC2
670
               N, NW
        Š.
000
         COMMON /AVE/ PPAVE
         COMMON /BLK1/ N(12,3), VTX, VALV(6), PALV(6), Sw(12), VSPR
690
700
         COMMON C(12,4), DC(12,4), CT(12), CW(12), DCw(12), DCwDP(12),
             F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PES,
710
        &
720
        š
             DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
730
        &
             PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
740
        å
             S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
750
        R.
             SQT(12), DP(12), HOT
760
         COMMON /BCMBLK/ R.FKDA.PP.VO.A2DM2.A2DM3.A2DM4.A2DM12.A2DM15.
770
        1A2DM14.H.RHO.GHOLD.RIN.PHOLD.PC
780
         COMMON /BCONLY/LCO2,LCC2,LCC0,LCX
790
         DIMENSION R(24), FKDA(24), PP(24), VO(24), H(24), DV(24),
800
        1VFLOW(24), VFLIN(18), RIN(18), PC(24)
810
         DIMENSION LCO2(6), LCC2(6), LCCO(6), LCX(6)
820
         DIMENSION PHOLD(24)
       WARNING...THIS DIM IS IN THIS SUBROUTINE ONLY
5300
840
         DU 10 I=1,0
850
         LCO2(1)=C(1+6,1)*RTC
         LCC2(1)=C(I+6.2)*RTC
660
870
         PP(1+4) = PALV(1) - 700.
086
      10 CONTINUE
040
         PPAVE=PPL+(PG(3)+PG(4))/2.-760.
900
         RETURN
910
         END
920
         SUBROUTINE BLCONV(CONV)
930
         IMPLICIT REAL*3 (A-H.O-Z)
940
         REAL*8 LAMBO2, LAMBC2, LAMBCO, LAMBX, LCO2, LCC2, LCCO, LCX, METO2, METC2
950
         COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GP02,CT02,GPC2,CTC2
960
         CUMMON /CMN1T/GPCO,CTCO,GPX,CTX
970
         COMMON /BC/LAMBO2, LAMBC2, LAMBCO, LAMBX, AO2, AC2, ACO, AX, HEMAT, D1F02,
         1D1FC2, D1FC0, D1FX, CPH, CSAT02, CSATC2, MET02, METC2
980
990
         COMMON /BCONLY/LCU2,LCC2,LCC0,LCX
1000
           COMMON /BLK12/ DIFF(6,3)
```

1010	DIMENSION V(24), METO2(4), METC2(4)
1020	DIMENSION GPO2(24), GPC2(24), GPCO(48), GPX(1,24), CTO2(4), CTC2(4),
1030	1CTCO(4), CTX(1,4), D1FO2(10), D1FC2(10), D1FCO(10), D1FX(10), CPH(24),
1040	2LC02(0),LCC2(6),LCC0(6),LCX(1,6),CSAT02(24),CSATC2(24),DGP02(24),
1050	3DGPC2(24), DGPCU(48), DGPX(1,24), AX(1)
10o0	DO 10 N=1,0
1070	DIFF(N,1)=DIFO2(N)*(LCO2(N)-LAMBO2*GPO2(N+4))*CUNV
1050	D1FF(N,2)=D1FC2(N)*(LCC2(N)-LAMEC2*GPC2(N+4))*CONV
1090	10 CONTINUE
1100	RETURN
1110	ĒND

¥

.

.

SUBROUTINE LMAIN(LDUMMY)
INTEGER HAMING, OPI1, OPT2, FBOPT
IMPLICIT REAL*5(A-H, O-Z)
REAL*O N, NW
REAL*4 A, B, W, TIN, TEX, Y1, Y2, FRCP, VMAX, VMIN, PDAG,
& FCTN, FCTN2, Y3, Y4, TB, PB, PDIN, PDEX, G1, G2, G5, G4,
& PTXG, PTXIN, PTXEX, PCALC, CTCALC, VLDIF, PRCT, ASUL, PV,
& RES, PPLC, ASUMI, AWSUM, AWSUMI, RTOI, 10, 12, TTESI
DIMENSION TE(62), FCT(62), FCTS(62,4), DFCT(62), DFCTS(62,5),
& PRC1(12,4), ASUM(4), PV(12), RES(12), PPLC(5), PSAVE(6)
DIMENSION ASUM1(4)
COMMON /LMN/ KNT
COMMON /BLK1/ N(12,3), VTX, VALV(6), PALV(6), SW(12), VSPR
COMMON /BLK2/ DN(12,3), DVTX, DVALV(6), DPALV(6), D3W(12), DVSPR
COMMON /BLK3/ B(0,12), FS(12), RA(12), RE(12), W(12), DIM
COMMON /BLK4/ TIN, TEX, PDIN, TB, Y1, Y2, Y3, Y4, PIXIN, PTXEX,
& PDEX, PDAG, PTXG, TO, T2, PB, FCTST, FSPRS, 11EST
COMMON /BLK5/ $A(\gamma, 8)$
COMMON / BLK // PCALC(12), CTCALC(12), VLD1F
COMMON /BLK6/ PH1, GF, GR, GT, GD, HP, P1E, ANG1, ANG2, PH11, PH12
COMMON /BLK10/ PGAV, DPGAV, PGAVB, DPGAVB
COMMON /BLK11/ PTX, DPTX, PD, DPD, G1, G2, G3, G4, FBOPT
COMMGN /BLK12/ DIFF(6,3)
COMMON C(12,4), DC(12,4), CT(12), CW(12), DCw(12), DCWDP(12),
& F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PES,
& DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
& PG(0), DPG(0), PGAB, DPGAB, PPL, DPS(6), DPS2,
& S(4), T, DT, IN(12), TOIN(12), V(12), VD, VLUNG,
& SQT(12), DP(12), hDT
EQUIVALENCE (N(1), FCT(1), FCTS(1)), (DN(1), DFCT(1), DFCTS(1))

```
310
         DATA PIE. HP / 3.141592654 DO. 1.570796327 DO /
320
         REWIND 12
330
         UNSTAB = 0.
340
         NF = 62
350
         NG = 3
360
         TO = 0.
370
         DO 50 K= 1,4
300
         DO 50 J= 1.NF
         1F(K-4)49.50.50
390
400
      49 DFCTS(J,K) = 0.
410
      50 \text{ FCTS}(J.K) = 0.
420
         READ(5,2008) OPT2, KPRINT
430 2008 FORMAT(214)
         READ(5,2000) T, DT, TMAX, TIN, TEX, INT, UPT1, TMP, TMPS
440
450 2000 FORMAT ( D10.5, D10.8, D10.5, 2F10.5, 2I3, 2D6.1 )
460
         DTM = DT
470
         1NTS = INT
450
         READ(5,2003) PPL, PDEX, PTXEX, RG, VD, Y1, Y2
490
         READ(5,2004) PTXG, DPTXG,
500
             PDAG, DPDAG, VTX, VMAX, FCTN
        ጲ
         READ(5,2001) VMIN, FCTN2, Y3, Y4, TB, PE
510
520 2001 FORMAT ( 7F10.4 )
530 2003 FORMAT ( 5D10.4, 2F10.4 )
540 2004 FORMAT ( F10.4, D10.4, F10.4, 2D10.4, 2F10.4 )
550
         READ(5.2010) V
500
         READ(5,2010) ( P(J), J=1,12 )
570
         READ(5,2002) ( ( PRCT(1,J), I = 1, 12 ), J = 1, 4 )
550 2002 FORMAT ( 6F10.4 )
         READ(5,2001) ( ( E(1,J), I=1,7), J=2,12 )
590
600
         READ(5,2001) A
610
         DU 5 1= 1.0
620
         A(4,1) = A(4,1) - .1D-0
ó30
         A(3,1) = A(3,1) + .1D-0
640
       5 A(o,1) = A(1,1) * HP
650 2005 FURMAT ( 3D10.0, 13 )
         READ(5,2006) (W(J), J=1,12)
660
670
         READ(5,2009) (FS(J), J=1,12), F(1)
680 2006 FORMAT ( 6E10.3 )
         READ(5,2005) PHI1, PHI2, GF, FBOPT
690
700 2010 FORMAT ( 6D10.4 )
710 2009 FORMAT ( 7D10.4 )
720
         WR1TE(6, 2007)
730 2007 FORMAT(1H1)
740
         WRITE(6,4001) T,DT,TIN,TEX,INT,OPT1,TMP,TMPS,PPL,PDEX,PTXEX,RG,
750
        &
              (J, J=1, 6), VD, Y1, Y2, PTXG, (V(J), J=1, 6), (J, J=1, 12), (P(J), J=1, 12),
760
              (J, J=1, 12), (J, (PRC1(1, J), I=1, 12), J=1, 4), (J, J=1, 7),
        Ł
770
        ĉ.
              ((E(I,J),I=1,7),J=2,12),(J,J=1,7),A,
700
              (W(J), J=2,12), PH11, PH12, GF, Y3, Y4, TB, VMAX, FCTN,
        å
790
              VMIN, FCIN2
        &
600 4001 FORMAT(1H ,10X,'T',5X,'DT',7X,'TIN',7X,'TEX',7X,'INT',7X,'OPT1',
```

```
810
                            6X, 'TMP', 7X, 'TMPS', 6X, 'PPL', 6X, 'PDEX', 5X, 'PTXEX', 6X, 'RG', //,
                 &
820
                 r
k
                            6x, 2E10.3, 2(F7.3.3x), 2(2X.15.5X),
                            6(F7.3.3X),///,9X,'VD',8X,'Y1',8X,'Y2',7X,'PTXG',7X,
830
                 &
                 &
840
                            6('V(',11,')',6X),//.6X,10(F7.3,3X),///,8X,12('P(',12,')',5X),
                            //,oX,12(F7.3,3X),///,2X,'PRCT',3X,12(12,6X),//,4(3X,11,2X,
850
                 &
860
                 å
                            12(F7.5.3X), //), /, 5X, 'B', 8X, 7(11, 9X), //, 5X, '(12)', 4X, 7(F7.3), 12(F7.5, 3X), //), 12(F7.5, 3X), 12(F7
870
                 &
                            3X),//,3X,'(23)',4X,7(F7.3,3X),//,3X,'(24)',4X,7(F7.3,3X),//,
                 &
                             3x, '(35)', 4x, 7(F7.3, 3x), //, 3x, '(46)', 4x, 7(F7.3, 5x), //, 3x, '(3/)
860
                 &
890
                             ',7(F7.3,3x),//,3x,'(4o)',4x,7(F7.3,3x),//,3x,'(59)',4x,7(F7.3
                            (3x), //, 2x, (610), 4x, 7(F7.3, 3x), //, 2x, (511), 4x, 7(F7.3, 3x),
900
                 &
                            //.2X.'(b12)'.4X.7(F7.3.3X).///.7X.7('A('.11.')'.6X).//.0(5X.
910
                 &
920
                 &
                            7(Fo.4,2X),//),/,7X,'W12',0X,'W23',0X,'W24',0X,'W35',0X,'W46',
                 &
                            JX, 'W37', JX, 'W48', JX, 'W59', 7X, 'W610', 7X, 'W511', 7X, 'W612', //,
930
94ú
                 &
                            5X,11(E10.4,1X),///,7X,'PHI1',7X,'PH12',,7X,'GF',0X,'Y3',0X,
                             'Y4', 6X, 'TB', 6X, 'VMAX', 6X, 'FCTN', 6X, 'VMIN', 6X, 'FCTN2'//
950
                 &
960
                 &
                            5x, F7.3, 5X, F7.3, 3X, F7.2, 7(3X, F7.4), /// )
970
             51 CONTINUE
900
                    D0.55 1 = 1.4
990
             55 S(1) = 0.0
1000
                      TMAX1 = TMAX + DT / 2.
1010
                      G = TMP * RG
1020
                      GS = TMPS * RG
1030
                      G1 = 0.84791 / (G*G)
1040
                      G2 = 0.010015 / G
1050
                      G_3 = 47.067/G
1050
                      G4 = 1./G
                      PHI1 = PHI1 * PIE / 180.
1070
1000
                      PH12 = PHI2 * P1E / 180.
10900
                      CALL PGRAV ( PES, PGAB, PG, DPBS, DPGAB, DPG )
1100
1110
                      PTX = PTXG * GT
                      PD = PDAG * GD
1120
11300
1140
                      FRCP = FRC2 (VA, VTX, PD, PTX)
1150C
                      IF ( ABS( VMIN ). LT. 0.0001 ) GO TO 57
1100
1170C
1100
                      CALL PMAX ( FRCP, VMIN, PG, PES, GT, GD, PGAE,
1190
                    &
                                                 FCTN2, VTX, VD, PPL, VA )
12000
1210
                      PTXEXS = PTXIN
1220
                      PDEXS = PD1N
1230
                57 CONTINUE
1240C
1250
                      CALL PMAX ( FRCP, VMAX, PG, PBS, GT, GD, PGAb,
1200
                    &
                                                 FCTN, VTX, VD, PPL, VA )
12/00
1200
                      IF ( TIN. LE. O. ) CALL SUB3( V )
1290
                      VSPR = V(1)
1300
                      D0.60 J = 1,6
ready
```

¥

```
1310
          K = J+6
1320
          PSAVE(J) = P(J)
1330
          PALV(J) = P(K)
1340
       60 \text{ VALV}(J) = V(K)
          PPLC(1) = PPL + PG(1)
1350
          PPLC(2) = PPL + PG(3)
1560
13/0
          PPLC(3) = PPL + PG(5)
          WRITE (6,995) GR, PPL, PPLC(2), VLUNG, VTX, VD, VALV
1300
      998 FURMAT(1H ,4X,'GR', OX, 'PPL', 7X, 'PPLAVG', 6X, 'VLUNG', OX,
1390
              'VIX', 9X, 'VD', 8X, 'VALV', //,
1400
         ¢.
1410
         å
              1X, F5.1, 1X, 2F12.6, 9F11.7, /// )
1420
          VLDIF = VLUNG - VTX - VD
1430C
           IF ( FBOPT. EQ. 11 ) CALL PCHK( VMIN, VMAX, FRCP, FCIN,
1440C
14500
               FCTN2, VLDIF, PGAB )
          &
1460C
           IF ( FBOPT. EQ. 10 ) CALL FCHK( T, FSPR, DT )
1470C
1480
          TOTN(1) = P(1) * V(1)/GS
          C_{W}(1) = 31.0 * D_{ZXP}(0.018 * (P(1) - 760.) / GS) / GS
1490
1500
          NW(1) = CW(1) * V(1)
1510
          TN(1) = IOTN(1) - NW(1)
1520
          DO 70 1=1, NG
          N(1,1) = TN(1) * PRCT(1,1)
1530
       70 C(1,1) = N(1,1) / V(1)
1540
1550
          PV(1) = P(1) * V(1) / (TOTN(1) * GS)
1500
          DO 80 J=2,12
1570
          TOTN(J) = P(J) * V(J)/G
1500
          Cw(J) = G3 * DEXP(G2 * (P(J)-760.))
1590
          NW(J) = CW(J) * V(J)
          TN(J) = TOTN(J) - Nw(J)
1600
1010
          PV(J) = P(J) * V(J) / (TOTN(J) * G)
1620
          DU 60 I=1, NG
          N(J,1) = TN(J) * PRCT(J,1)
1630
       80 C(J,1) = N(J,1)/V(J)
1640
1050 3011 FORMAT( //,1H ,4(5X, 6(E17.11), //) )
1660
          DO 1500 1 = 1, NG
1670 1580 \text{ ASUM1(I)} = 0.
1600
          AWSUM1 = 0.
          DO 1585 J= 1,12
1690
1700
          AWSUM1 = AWSUM1 + NW(J)
1710
          DO 1585 1= 1.NG
1720 1585 ASUM1(I) = ASUM1(I) + N(J,I)
1730
          HDT = DT/4.
          CALL FLOW (1, 2)
1740
          CALL FLOW (2, 3)
1750
          CALL FLOW (2, 4)
1700
1770
          CALL FLOW (3, 5)
1700
          CALL FLOW (4, 6)
          CALL FLOW (3, 7)
1790
1800
          CALL FLOW (4, 8)
```

×

```
CALL FLOW (5.9)
1810
1820
          CALL FLOW ( b, 10 )
1830
          CALL FLOW (5, 11)
1040
          CALL FLOW ( b. 12 )
1850
       90 KNT = 0.
1600
          DO 100 K = 1, NF
          FCTS(K, 4) = FCT(K)
1070
1800
      100 \text{ TE}(K) = 0.
1890
          WRITE(6.2007)
1900
          RETURN
1910
          ENTRY LMN1(LDUMMY)
19200
1930C
          $55 RUNGE-KUTTA INTEGRATION SECTION $$$
19400
1950
          M = 1
1960
      110 CALL DERIVS( OPT1, 1, PSAVE
                                         )
1970
          IF( T. LT. 1.D-6 ) GO 10 151
1900
      111 IF( M. EQ. 5 ) GO TO 130
      120 CALL RUNGE( NF, FCT, DFCT, T, DT, M, HDT )
1990
2000
          DTM = HDT * 2.
          CALL PARMS2( VA )
2010
2020
          M = M + 1
2030
          GO TO 110
      130 \text{ KNT} = \text{KNT} + 1
2040
2050
          \dot{M} = 1
2050
          K = 4 - KNT
2010
          HDT = DT/2.
2080
          1F( K. EQ. 1 ) GO TO 150
2090
          DO 140 J= 1.NF
          FCTS(J,K) = FCT(J)
2100
2110
      140 DFCTS(J,K) = DFCT(J)
2120
      150 DU 155 J= 2,12
      155 FS(J) = F(J)
2150
2140
          DU 154 JPS = 2,6
      154 \text{ PSAVE(JPS)} = P(JPS)
2150
2160
          1NT = 1NTS
          1F ( T. LT. .15 ) INT = INTS / 10
21700
2160
          1F (1NT. EQ. 0) 1NT = 1
2190
      152 IF ( KNT/INT * INT .NE. KNT ) GO TO 190
2200
      151 CONTINUE
2210
          CALL SUE2( ASUM, ASUM1, AWSUM, AWSUM1, RES, PV, RTOT, PPLC )
2220
          IF( KPRINT. NE. 1 ) GO TO 153
2230
          WRITE(6,3006) KNT
2240
          wRITE(6,3000) T, VTX, DVTX, VD, VSPR, DVSPR
2250
          WRITE(6,3001) ((N(J,I),J=1,12), (DN(J,I),J=1,12), (C(J,I),J=1,12),
2260
               (DC(J,I),J=1,6),I=1,NG )
         X.
          WRITE(6,3002) NW, DNW, CW, (DCW(J), J=1,6), (PCALC(J), J=1, 0),
2270
2200
               P, DP, VALV, DVALV, F, DCWDP
         &
2290
          WRITE(6, 1) PELALV, S, PELDA, PELTX, PTX,
2300
         &
               PD, PG, PGAB, PPL, PPLC, VLUNG, RTOT, ASUM(1), ASUM(2), ALSUM
```

111

ready

*

```
2310
          WRITE(6.3009) DNT. CTCALC
2320
      153 CONTINUE
          WRITE(12,3100) T, VSPR, PPL, PPLC, RTOT, VLUNG, VIX, VD,
2350
2340
         &
                         PTX. PD. VALV. PELALV. PELTX. PELDA
2350 3100 FORMAT( 10E14.7 )
2560
          WRITE(12,3100) P, F, RES, RA, NW, CW, CT,
2370
                          ((N(J,I),J=1,12), (C(J,I),J=1,12), I=1,NG)
         6
2380 3000 FORMAT( 1H , 4X, 'T=', 6E13.6 )
2390 3001 FURMAT( 1H ,
                         3(2(6E13.6./), )6E13.6)
2400 5002 FORMAT( 1H,
                         7(2(6E13.6, /)), 2(12E10.4, /)
                                                                  )
2410 3009 FORMAT( 1H , 2( /, 5x, 6E13.6 ) )
2420 3005 FORMAT( 1X, 'KNT = ', 16
                                        )
2430
        1 FORMAT( 1H , 10E13.6 )
2440
          1F( T. LT. 1.D-0 ) GO TO 111
2450
      190 CONTINUE
2460
          RETURN
2470
          ENTRY LMN2(LDUMMY)
2400
          IF ( KNT. LT. 3 ) GO TO 120
2490
          IF(OPT2. LT. 1 ) GO TO 1000
2500C
25100
          $55 CALL HAMING PREDICTOR OR CORRECTOR $$$
25200
2530
          TTESTO = TTEST
2540
          DTM = HDT \times 2.
2550
          MTEST = HAMNG1( NF, FCTS, DFCTS, T, DT, TE )
      500 MTEST = HAMING( NF, FCTS, DFCTS, T, DT, TE )
2500
2570
          CALL PARMS2( VA )
          IF ( T. GT. TIN - .001 ) PDEX = PDEXS
2580
2590
          IF ( T. GT. TIN - .001 ) PTXEX = PTXEXS
          CALL DERIVS( OPT1, MTEST, PSAVE )
2000
2010
          IF( MTEST. EQ. 1 ) GO TO 500
2620
          KNT = KNT + 1
2000
          DU 500 J = 2.12
2640
      506 FS(J) = F(J)
2650
          FSPRS = FS(2)
2000
          DO 501 JPS = 2.6
2670
      501 \text{ PSAVe}(\text{JPS}) = P(\text{JPS})
2000
          IF ( P(9). LT. 550.0. OR . P(9). GT. 850.0 ) GO TO 502
2090
          IF ( P(4). LT. 650.0. OR . P(4). GT. 850.0 ) GO TO 502
2700
          GO TO 503
2710
      502 END FILE 12
2720
          UNSTAB = 10.
2750
          KPRINT = 1
2740
          GO TO 510
2750
      503 CONTINUE
2760
          INT = INTS
27700
          1F ( T. GT. TO. AND. T. LT. TO + .15 ) INT = 1NTS / 10
          1F ( T. GT. T2. AND. T. LT. T2 + .15 ) INT = INTS / 10
2700C
2790
          1F ( INT. EQ. 0 ) INT = 1
          IF (KNT.LT.1000) GO TO 600
2800
```

×

```
505 IF ( KNT/INT * 1NT . NE. KNT ) GO TO 600
2810
2820
     510 CONTINUE
2030
          CALL SUB2( ASUM, ASUM1, AWSUM, AWSUM1, RES, PV, RTOT, PPLC )
2840
          IF( KPRINT. NE. 1 ) GO TO 515
2850
          WRITE(6.3006) KNT
          WRITE(6,3000) T. VTX, DVTX, VD, VSPR, DVSPR
2000
2870
          WRITE(6,3001) ((N(J,I),J=1,12), (DN(J,I),J=1,12), (C(J,I),J=1,12),
              (DC(J,I),J=1,v),I=1,NG )
2890
         æ
          WRITE(6,3002) NW, DNW, CW, (DCW(J), J=1,6), (PCALC(J), J=1,0),
2890
              P, DP, VALV, DVALV, F, DCWDP
2900
         å
2910
          WRITE(6, 1) PELALV, S, PELDA, PELTX, PTX,
2920
              PD, PG, PGAE, PPL, PPLC, VLUNG, RTOT, ASUM(1), ASUM(2), AWSUM
         Š.
          WRITE(6,3009) DNT, CTCALC
2930
2940
          IF( UNSTAB. GT. 5. ) STOP 2
2950
      515 CONTINUE
2960
          WRITE(12,3100) T, VSPR, PPL, PPLC, RTOT, VLUNG, VIX, VD,
2970
                         PTX, PD, VALV, PELALV, PELTX, PELDA
         å
          WRITE(12,3100) P, F, RES, RA, NW, CW, CT,
2900
                         ((N(J,I),J=1,12), (C(J,I),J=1,12), I=1,NG)
2990
         Å.
3000
      600 CONTINUE
3010
      625 CONTINUE
3020
          TTESTO = TTEST
3030
          RETURN
3040
          ENTRY LMN3(LDUMMY)
          IF (T = TEAX1) 500, 1000, 1000
3050
3050 1000 CONTINUE
3070
          RETURN
3020
          ENTRY LMN4(LDUMMY)
3090
          T = 0.0
3100
          WRITE(12,3100) (T,1=1,10)
3110
          END FILE 12
3120
          WRITE (6,9999) T
3130 9999 FORMAT (' LUNG PROGRAM HAS ENDED AT TIME=', F10.5)
3140
          RETURN
5150
          END
```

```
ready
```

¥

```
40
        REAL*5 LAMBO2.LAMBC2.LAMBC0.LAMBX.LCO2.LCC2.LCC0.LCX.METO2.METC2
50
        COMMON /BCM/ TB.TMAXB
60
        COMMON /BCMBLK/ R.FKDA.PP.VO.A2DM2.A2DM3.A2DM4.A2DM12.A2DN15.
70
       1A2DM14.H.RHO.G.RIN.P.PC
80
        COMMON /RP/ RP3.RP13
90
        COMMON /SK/ TAU.PO.P1
         COMMON /CMN1/V, VP2, VP3, VP4, VP12, VP13, VP14, GP02, CT02, GPC2, CTC2
100
110
         COMMON /CMN1T/GPCO, CTCO, GPX, CTX
120
         COMMON /CMN2/DV, DVP2, DVP3, DVP4, DVP12, DVP13, DVP14, DGPU2, DCTU2,
130
        1DGPC2.DCTC2
140
         COMMON /CHN2T/DGPCO.DCICO.DGPX.DCTX
150
         COMMON /A1/LVALV2.LVALV3.LVLV12.LVLV13.LVLV23.
160
        1PST2.PST12.TBS2.1bS1
170
         COMMON /KNT/KNTB
180
         COMMON /PHT/KTB,KTDEL
190
         COMMON /VFLT/VFLOW,VFLIN
200
         COMMON / bC/LAMEC2, LAMBC2, LAMBC0, LAMEX, AO2, AC2, AC0, AX, HEMAT, D1F02.
210
        1D1FC2, D1FC0, D1FX, CPH, CSATO2, CSATC2, METC2, METC2
220
         COMMON /BCONLY/LCO2.LCC2.LCC0.LCX
230
         EQUIVALENCE (V.FCTB.FCTSB).(DV.DFCTB.DFCTSB)
240
         DIMENSION R(24), FKDA(24), PP(24), V(24), VO(24), H(24), DV(24),
250
        1VFLOW(24), VFLIN(15), RIN(15), PC(24), P(24)
260
         DIMENSION RK(24), RINK(18)
270
         DIMENSION 1AU(24), PO(24), P1(24)
200
         DIMENSION FCTB(06), DFCTB(86), FCTSB(86,4), DFCTSB(86,5), TEB(d6)
         DIMENSION GPO2(24), GPC2(24), GPCO(48), GPX(1,24), CTO2(4), CTO2(4),
290
        1CTCO(4).CTX(1.4).DIFU2(10).DIFC2(10).DIFCO(10).DIFX(10).CPH(24).
300
310
        2LC02(6),LCC2(6),LCC0(6),LCX(1,6),CSAT02(24),CSAT02(24),DGP02(24),
320
        3DGPC2(24), DGPCO(43), DGPX(1,24), AX(1)
         DIMENSION DCTO2(4), DCTC2(4), DCTCO(4), DCTX(1,4), METO2(4), METC2(4)
330
340C
       THIS DIMENSION STATEMENT IS ONLY IN BMAIN
3500
      FCTB(V=24, VP=6, C=24, CT=4)
3600
      ADD 20 LOCATIONS FOR EAC NEW GAS TRANSPORTED*. THIS PROGRAM IS
      PRESENTLY SET UP FOR 24 CHAMBERS AND 1 GAS..NFE=NC+6+NG(NC+4)=55
3700
3006
       *EXCEPT FOR CO. WHICH REQUIRES TWICE AS MANY LOCATIONS (2*20=56).
         NFB=06
390
400
         DO 50 K=1.4
410
         DO 50 J=1,NFB
420
         IF (K-4) 49,50,50
430
      49 DFCTSB(J,K)=0
      50 FCTSE(J,K)=0
440
450
         READ(5,1010) (R(N), N=1,24), (FKDA(N), N=1,24), (PP(N), N=1,24)
460
         READ(5,1010) (TAU(N),N=1,24),(PO(N),N=1,24)
4/0 1010 FORMAT(8(E8.1.1X))
         READ(5,1020) (V(N),N=1,24),(VO(N),N=1,24),A2DM2,A2DM3,A2DM4,A2DM12
480
490
         1, A2DM13, A2DM14
```

```
500 1020 FORMAT(8(F8.3.1X))
```

10

20

30

SUBROUTINE BCMAIN(LDUMMY)

IMPLICIT REAL*S (A-H.O-Z)

INTEGER HAMING

```
510
         READ(5,1030) (H(N),N=1,24),RHO,G
520 1030 FORMAT(2(12(F5.1.1X)/),F3.1.F5.1)
         READ(5,1040) (RIN(N),N=5,11),(RIN(N),N=15,1°),VP2,VP3,VP4,VP12,
530
540
        1VP13.VP14
550 1040 FORMAT(7E9.1/4E9.1/6E9.1)
560
         READ (5,201) GP02,GPC2,GPC0,GPX
570
     201 FORMAT (8E9.2/8E9.2/8E9.2)
580
         READ(5.201) LAMBO2.LAMBC2.LAMBC0.LAMBX.AO2.AC2.ACC.AX.CTU2.CTU2.
590
        1CTCO.CTX
000C
       CT STAND FOR THE PARTIAL PRESSUR IN THE TISSUE, NOT THE CONC.
610
         READ(5.203) LCO2.LCC2.LCC0.LCX
620
         READ (5,204) DIFC2, DIFC2, DIFC0, DIFX
6.40
         READ (5,205) METO2, METC2
     204 FORMAT (SE9.2/2E9.2)
640
650
     203 FORMAT (6E9.2)
660
     205 FORMAT (4E9.2)
610
         READ(5,1050) DTB.TMAXB
680 1050 FORMAT (Fd.1.1X.Fo.1)
690
         READ (5,1200) KNTP,LOPPR,LOPP1,LOPR,LOPGS
700 1200 FORMAT (12,1X,11,1X,11,12,12)
710C
      LOPPR=1 IF RUNGE IS TO BE PRINTED, LOPPR=0 IF NOT
7200
      KNTP TELLS HOW OFTEN (NO. OF TIME STEPS) TO PRINT
7300
       LOPP1 CONTROLES LIN(0) OR NONLINEAR COMPLIANCE
740C
       SEE SUB RESIST FOR EXPLAINATION OF LOPR
       SEE SUB ECDRIVS FOR EXPLAINATION OF LOPGS
750C
700
         WRITE(0,2010)
770 2010 FORMAT(10X, 12H(INPUT DATA)/)
780
         WRITE (6,2001) DTB, TMAXB
790 2001 FORMAT(10H DTB, TEAXE, 5X, 2F8.4)
800
         WRITE(6,2040) (RIN(N), N=1,18)
810 2040 FURMAT(10H
                     RIN(N) ,5X,0(E10.3,2X)/15X,8(E10.5,2X)/15X,2(E10.5,
820
        12X)/)
0,0
         WRITE(0,2020) R,FKDA,PP
                         R(N) ,5X,8(E10.3,2X)/2(15X,o(E10.3,2X)/)///
640 2020 FURMAT(10H
850
        110H
              FKDA(N) ,5X, d(E10.3,2X)/2(15X, d(E10.3,2X)/)///
                PP(N) ,5x,o(E10.3,2X)/2(15x,8(E10.3,2X)/)///)
860
        210H
870
         WRITE(6,2021) TAU, PO
                       TAU(N) .5X.8(E10.3.2X)/2(15X.8(E10.3.2X)/)///
800 2021 FORMAT(10H
890
        110H
                PO(N) ,5X,8(E10.3,2X)/2(15X,8(E10.3,2X)/)///)
900
         WRITE(6,2030) V,VO,A2DM2,A2DM3,A2DM4,A2DM12,A2DM13,A2DM14
910 2030 FORMAT(10H
                         V(N) ,5X,8(F8.3,2X)/2(15X,8(F8.3,2X)/)///
                VO(N) ,5X,8(F8.3,2X)/2(15X,8(F8.3,2X)/)///
920
        110H
930
                  A2DM ,5X,o(Fd.3,2X)///)
        210H
940
         WRITE (6,2031) H
950 2031 FORMAT (10H
                       H(N)
                               ,5X,8E12.3/2(15X,8E12.3/)///)
         WRITE (6,2032) RHO,G
960
970 2032 FORMAT (' RHO,G =
                               ',2F8.3)
         WRITE (6,2033) VP2,VP3,VP4,VP12,VP13,VP14
980
990 2033 FORMAT ('
                      ٧P
                             ',6E12.3///)
1000
          WRITE (6,2041) KNTP,LOPPR,LOPP1,LOPR,LOPGS
```

```
1010 2041 FORMAT (' OPTIONS ',5X,12,1X,12,1X,12,1X,12,1X,12)
1020
          WRITE (6.208) GPO2.GPC2.GPC0.GPX
1030
     208 FORMAT (15H GP02.C2.C0.PX .8(E10.3.2X)/14(15X.8E12.3/))
1040C
        WARNING...FORMAT MUST BE CHANGED IF NGX.GT.1
1050
          WRITE (6.209) LAMBO2.LAMBC2.LAMBCO.LAMBX.AO2.AC2.ACO.AX
1060
      209 FORMAT (' LAMB.A...02.C2.C0.X './.15X.8E12.3/)
1070
          WRITE (6,210) CTO2,CTC2,CTCO,CTX
1080
      210 FORMAT (' CTO2, C2, CO, X ', /, 2(15X, 8E12.3/))
1090
          WRITE (6,211) LCO2,LCC2,LCC0,LCX
1100
      211 FORMAT (' LCO2.C2.CO.X './4(15X.6E12.3/))
1110
          WRITE (6,212) DIF02, DIFC2, DIFC0, DIFX
1120
      212 FORMAT (' DIF02,C2,C0,X ',/,4(10E11.3/))
          WRITE (6,216) METO2, METC2
1130
1140
      216 FORMAT (' METO2.METC2 '.4E12.3.10X.4E12.3)
1150
          RETURN
1160
          ENTRY BCMN1(LDUMMY)
1170
          WRITE(6.2050)
1180 2050 FORMAT(10X.13H(OUTPUT DATA)/)
1190
          WRITE (6.3002) PP.LC02.LCC2
1200
          SUMVI=0
1210
          DO 444 N=1.24
1220
      444 SUMVI=SUMVI+V(N)
1230
          T\hat{B}=0
1240
          KNTE=0
1250
          MTEST=0
1260
          RIN(4) = .0136
1270
          RIN(14) = .068
1280
          RP3=3.E-07
1290
          RP13=1.5E-06
1300
          HEMAT = .45
        INITIAL VALUES FOR PHEART SET HERE
1310C
1320
          PST01=.26
1330
          PST02=.43
1340
          PST03=.26
1350
          PST04=.43
1360
          PSDUR1=.35
1370
          PSDUR2=.53
1380
          PSDUR3=.35
1390
          PSDUR4=.53
1400
          PSAMP1=5.
1410
          PSAMP2=20.
1420
          PSAMP3=15.
1430
          PSAMP4=110.
1440
          PSRT=70
1450
          CALL CHEART (PSDUR1, PSAMP1, PSRT, Z1, PSEND1, PSTO1)
1460
          CALL CHEART(PSDUR2, PSAMP2, PSRT, Z2, PSEND2, PSTON2, PSTO2)
1470
          CALL CHEART (PSDUR3, PSAMP3, PSRT, Z3, PSEND3, PSTON3, PSTO3)
1480
          CALL CHEART(PSDUR4, PSAMP4, PSRT, Z4, PSEND4, PSTON4, PSTO4)
1490C
1500
          CALL PHEART(PSDUR1, PSAMP1, PSRT, TB, PP(2), PSTO1, Z1, PSEND1, PSTON1)
```

```
1510
          CALL PHEART(PSDUR2.PSAMP2.PSRT.TB.PP(3).PSTO2.22,PSEND2.PSTON2)
1520
          CALL PHEART(PSDUR3.PSAMP3.PSRT.TB.PP(12).PSTU3.Z3.PSEND3.PSTON3)
1530
          CALL PHEART(PSDUR4.PSAMP4.PSRT.TB.PP(13).PSTO4.24.PSEND4.PSTON4)
1540
        2 CONTINUE
1550C
1550
          CALL CRES(LOPR.RK.RINK)
          CALL RESIST(LOPR.RK.RINK)
1570
1580C
        THE FOLLOWING SETS PC FOR THE HEART. IF ANY VALVE IS OPEN, INITIAL
15900
              CONDITIONS SHOULD GIVE THE CONNECTED CHAMBERS THE SAME
16000
              INITIAL PRESSURE FOR NON-INERTIAL FLUID
1010
          CALL SKON(LOPP1.TB)
1o20
          P(2)=PP(2)+P1(2)
1050
          P(3)=PP(3)+P1(3)
1640
          P(4) = PP(4) + P1(4)
1650
          P(12)=PP(12)+P1(12)
1000
          P(15)=PP(15)+P1(13)
1670
          P(14)=PP(14)+P1(14)
1000
          CALL BDRIVS(LUPP1.0.TE)
1690
          CALL ECDRIV(LOPUS)
1/000
1710
          IF (LOPP1.EQ.2) GO TO 998
1720C
1730
          DO 60 K=1.NFB
          FCTSB(K, 4) = FCTB(K)
1740
1/50
       00 \text{ TEB}(K) = 0
       ó5 M=1
1760
1770C THE LAST VARIABLE IN RUNGE IS DUMMY
1700
       6b CALL RUNGE (NFB, FCTB, DFCTB, TB, DTB, M, DUM)
          CALL PHEART(PSDUR1, PSAMP1, PSRT, TB, PP(2), PSTO1, Z1, PSEND1, PSTON1)
1790
1000
          CALL PHEARI(PSDUR2, PSAMP2, PSRT, TB, PP(3), PST02, 22, PSEND2, PST0N2)
1010
           CALL PHEART(PSDUR3, PSAMP3, PSRT, TB, PP(12), PST03, Z3, PSEND3, PST0N5)
1820
           CALL PHEART(PSDUR4, PSAMP4, PSRT, TE, PP(15), PSTO4, Z4, PSEND4, PSTON4)
1030
       22 CONTINUE
       SEE ABOVE
1040C
1850
           CALL RESIST(LOPR, KK, RINK)
1000
           CALL EDRIVS(LCPP1.0.TE)
1670
           CALL ECDRIV(LOPGS)
1880
       13 CONTINUE
1090
           M = M + 1
1900
           1F (M-5) 56.67.67
1910
       67 KNTB=KNTB+1
1920
           SUMV=0
1930
           DO 155 N=1,24
1940
      155 SUMV=SUMV+V(N)
1950
           DELV=SUMV-SUMVI
1960
           DO 153 N=1,24
1970
      156 1F (V(N)) 157,158,158
1930
      157 WRITE (6,3009) N,V(N)
                      CHAMBER NO. ', I2, ' HAS A NEG VOLUME = ', E12.6)
1990 3009 FORMAT ('
2000 158 CONTINUE
```

```
20100
        THE FOLLOWING IS A PRINT OPTION
          IF (KNTH/KNTP*KNTP.EO.KNTB) GO TO 998
2020
2030
       68 CONTINUE
2040
          RETHRN
2050
          ENTRY BCMN2(LDUMMY)
2050
          1F (KNTB-3) 70.72.72
2070
       70 K= 4-KNTB
          DO 71 J=1.NFB
2030
2040
          FCTSE(J,K) = FCTE(J)
2100
       71 DFCTSB(J.K)=DFCTB(J)
2110
          GU TO DS
2120
       72 CONTINUE
2130
          MTEST=HAMNG1(NFb,FCTSB,DFCTSB,TB,DTB,TEB)
2140
       89 CONTINUE
2150
       90 MTEST=HAMING(NFB,FCTSB,DFCTSB,TB,DTB,TEB)
          IF(MTEST.EQ.1) GO TO 32
2160
2170
          CALL PHEART(PSDUR1, PSAMP1, PSRT, TB, PP(2), PSTO1, Z1, PSEND1, PSTON1)
2100
          CALL PHEART(PSDUR2, PSAMP2, PSRT, TB, PP(3), PSTO2, Z2, PSEND2, PSTON2)
2190
          CALL PHEART (PSDUR3, PSAMP3, PSRT, TB, PP(12), PSTO<sub>2</sub>, 23, PSEND3, PSTON3)
2200
          CALL PHEART(PSDUR4, PSAMP4, PSRT, TB, PP(13), PSF04, Z4, PSEND4, PS10N4)
2210
       32 CUNTINUE
2220C
       SEE ABOVE
2250
          CALL RESIST(LOPR, RK, RINK)
2240
          CALL BDRIVS(LOPP1.MTEST.TB)
2250
          CALL BCDRIV(LOPGS)
2260
          IF (MTEST.EQ.1) GU TO 89
2270
          SUMV=0
2200
          DU 555 N=1,24
      555 SUMV=SUMV+V(N)
2290
2300
          DELV=SUMV-SUMVI
2310
          DO 558 N=1.24
2320
      556 IF (V(N)) 557,558,558
2330
      557 WRITE (0,3002) N,V(N)
2340
      558 CONTINUE
2350
           KNTE=KNTE+1
23000
       THIS IS THE PLACE TO TEST RESULTS TO SEE IF YOU WISH TO CONTINUE
2370
           1F (KNTB.LT.1000) GU TO 91
2380
           IF (KNTE/KNTP*KNTP.EQ.KNTE) GO TO 999
2390
       91 CONTINUE
2400
           RETURN
2410
           ENTRY BCMN3(LDUMMY)
2420
           IF (TE-TMAXE) 90, 100,100
2430
      998 CONTINUE
2440
           wRITE (4,3000) KNTB,KNTP,TB,DTB,LOPPR,LOPP1,LOPR,LOPGS,PP(2),
2450
          1PP(3), PP(12), PP(13), LVALV2, LVALV3, LVLV12, LVLV13, LVLV23
2460
           WRITE (4,3002) VP2,VP3,VP4,VP12,VP15,VP14,DVP2,DVP3,DVP4,DVP12,
24/0
          1DVP15, DVP14
           WRITE (4,3002) P,V,DV,VFLOW
2480
2490
           WRITE (4,3002) VFLIN,RIN,R
2500
           WRITE (6,3000) KNTB, KNTP, TB, DTB, LOPPR, LOPP1, LOPR, LOPGS, PP(2),
```

```
2510
          1PP(3), PP(12), PP(13), LVALV2, LVALV3, LVLV12, LVLV13, LVLV23
2520
           WRILE (6.3002) VP2.VP3.VP4.VP12.VP13.VP14.DVP2.DVP3.DVP4.DVP12.
          1DVP13.DVP14
2530
2540
           WRITE (6.3002) P.V.DV.VFLOW
           WRITE (6.3002) VELIN.RIN.R
2550
2500
           WRITE (0.3002) GP02.GPC2.CSAT02.CPH.DGP02.DGPC2
2570
           WRITE (6.3002) CTO2.DCTO2.CTC2.DCTC2
2580
           WRITE (4,3002) GP02,GPC2,CSAT02,CPH,DGP02,DGPC2
           WRITE (4,3002) CT02, DCT02, CTC2, DCTC2
2550
2600 3000 FORMAT (1X,216,2F10.4,416,4F10.4,513)
2610 3002 FORMAT (1x.12F10.4)
26200
 2630
           IF (LOPP1.EQ.2) GO TO 100
2640C
 2650
           GO TO 68
 2660 999 CUNTINUE
 2070
           WRITE (4,3000) KNTB, KNTP, TB, DTB, LOPPR, LOPP1, LOPR, LOPGS, PP(2),
 2650
          1PP(3).PP(12).PP(15).LVALV2.LVALV3.LVLV12.LVLV13.LVLV23
 2690
           WRITE (4,3002) VP2.VP3.VP4.VP12.VP13.VP14.DVP2.DVP5.DVP4.DVP12.
 2700
          1DVP13.DVP14
 2710
           WRITE (4.3002) P.V.DV.VFLOW
 2720
           WRITE (4.3002) VFLIN.RIN.R
           WRITE (6,5000) KNTB, KNTP, TB, DTB, LOPPR, LOPP1, LOPR, LOPGS, PP(2),
 2730
 2740
          1PP(3), PP(12), PP(13), LVALV2, LVALV3, LVLV12, LVLV13, LVLV23
 2750
           WRITE (6,3002) VP2,VP3,VP4,VP12,VP13,VP14,DVP2,DVP3,DVP4,DVP12,
 2700
          1DVP13, DVP14
 2770
           WRITE (6,3002) P.V.DV.VFLOW
 2780
           WRITE (6,3002) VFLIN.RIN.R
 2790
           WRITE (6,3002) GP02,GPC2,CSAT02,CPH,DGP02,DGPC2
           WRITE (6,3002) CT02, DCT02, CTC2, DCTC2
 2800
 2510
           WRITE (4,3002) GP02,GPC2,CSAT02,CPH,DGP02,DGPC2
 2820
           WRITE (4,3002) CTO2, DCTO2, CTC2, DCTC2
 2050
           GO 10 91
 2640
       100 CUNTINUÉ
 2050
           WRITE (6,9999) TB
 2860 9999 FORMAT (' ... BLOOD PROGRAM HAS ENDED AT TIME=', F10.5)
2070
           RETURN
 2800
           END
 2890
           SUBROUTINE BDRIVS (LOPP1.MTEST.TB)
 2900
           IMPLICIT REAL*3 (A-H,O-2)
 29100
        NEW FOR ABMINRC
 2920
           COMMON /RP/RP3, RP13
 29300
 2940
           COMMON /BCMBLK/ R,FKDA,PP,VO,A2DM2,A2DM3,A2DM4,A2DM12,A2DM13,
 2950
          1A2DM14, H, RHO, G, RIN, P, PC
 2960
           COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GP02,CTU2,GPC2,CTC2
 2970
           COMMON /CMN1T/GPCO,CTCO,GPX,CTX
           COMMON /CMN2/DV, DVP2, DVP3, DVP4, DVP12, DVP13, DVP14, DGP02, DCT02.
 2980
          1DGPC2.DCTC2
 2990
```

3000 COMMON /CMN2T/DGPCO, DCTCO, DGPX, DCTX

```
3010
          COMMON /A1/LVALV2.LVALV3.LVLV12.LVLV13.LVLV23,PS12,PS12,TBS2,TBS12
3020
          COMMON /SK/ TAU, PO, P1
5030
          COMMON /VFLT/VFLOW.VFLIN
          DIMENSION R(24).FKDA(24).PP(24).V(24),VO(24).H(24),DV(24),
3040
3050
         1VFLOW(24).VFLIN(18).RIN(18).PC(24).P(24)
          DIMENSION TAU(24), PO(24), P1(24)
3060
          DIMENSION GPO2(24).GPC2(24),GPCO(48),GPX(1,24),CTO2(4),CTC2(4),
3070
3080
         1CTCO(4), CTX(1,4), D1FO2(10), D1FC2(10), D1FCO(10), D1FX(10), CPH(24).
3090
         2LCO2(6).LCC2(6).LCCO(6).LCX(1.6).CSATO2(24).CSATC2(24).DGPO2(24).
         3DGPC2(24), DGPCU(45), DGPX(1,24), AX(1)
3100
          DIMENSION DC102(4), DCTC2(4), DCTC0(4), DCTX(1,4), METO2(4), METC2(4)
3110
3120
          CALL SKON(LOPP1.1B)
31:0
          P(1)=PP(1)+P1(1)
5140
          DO 10 N=5.11
5150
       10 P(N) = PP(N) + P1(N)
3160
          DO 12 N=15.24
3170
       12 P(N) = PP(N) + P1(N)
3100
          DO 20 N=1,24
3190
       20 PC(N)=P(N)+RHO*G*H(N)
          CALL PCHART(2.LVALV2,LVALV3,MTEST,PST2, 1BS2, 1b)
3200
3210
          CALL PCHART (12,LVLV12,LVLV13,MTEST,PST12,TBS12,TB)
3220
          VFLOW(1) = (PC(1) - PC(2))/R(1)
3230
          IF (LVALV2) 113,111,112
3240
      111 VFLOW(2) = (PC(2) - PC(3))/R(2)
3250
          GU TO 120
      112 VFLOW(2)=0
3200
3270
          GU TU 120
3200
     113 WRITE (6,1113)
3290 1113 FORMAT ('LVALV2 1S NEG .. NOT POSSIBLE')
3300
      120 IF (LVALV3) 123,121,122
      121 VFLOW(3) = -VP3
3310
3320
          DVP3=(-PC(3)+PC(4)+RP3*VP3**2-RIN(3)*VP3)*A2DN3
3350
          DV(3)=VPS
3340
          GU TU 130
3350
      122 VFLOW(3)=0
          VP3=0.0
0025
          DVP3=0.0
3370
3300
          DV(3) = VFLOW(2)
3390
          GO TO 130
3400 123 WRITE (6,1123)
3410 1123 FORMAT ('LVALV3 IS NEG .. NOT POSSIBLE')
3420
      150 VFLOW(4)=0
3430
          DÚ 13 N=5.11
3440
          VFLIN(N) = (PC(4) - PC(N))/RIN(N)
3450
          VFLOW(N) = (PC(N) - PC(11))/R(N)
3460
       13 VFLOW(4)=VFLOW(4)+VFLIN(N)
34/00
       THE FOLOWING IS A CORRECTION TO THE ZERO CALC. IN THE DO FOR
3450C
       VFLOW(11) ABOVE
3490
          VFLOW(11) = (PC(11) - PC(12))/R(11)
3500
          1F (LVLV12) 133,131,132
```

```
131 VFLOW(12)=(PC(12)-PC(13))/R(12)
3510
3520
          GO TO 140
3550
      132 VFLOW(12)=0
3540
          GO TO 140
3550
     133 WRITE (6.1133)
3500 1133 FORMAT ('LVLV12 IS NEG .. NOT POSSIBLE')
      140 IF (LVLV13) 143,141,142
3570
3580
      141 VELOW(13) = -VP13
          DVP13=(-PC(13)+PC(14)+RP13*VP13**2-RIN(13)*VP13)*A2DM13
3590
3600
          DV(13)=VP13
3610
          GO TO 150
3620
      142 VFLOW(15)=0
3630
          VP13=0.0
3640
          DVP13=0.0
          DV(13) = VFLOW(12)
3050
          GO 10 150
3600
      143 WRITE (0.1143)
3670
3000 1143 FORMAT ('LVLV13 1S NEG .. NOT POSSIBLE')
3690
      150 VELOW(14)=0
3700
          DU 14 N=15.10
          VFLIN(N) = (PC(14) - PC(N))/RIN(N)
3710
3720
          VFLOW(N) = (PC(N) - PC(N+4))/R(N)
3730
       14 VFLOW(14)=VFLOW(14)+VFLIN(N)
3/40C THE FOLLOWING TESTS VALVE 23
3750
          1F (MTEST.E0.1) GO TO 153
          IF (PC(1y)-PC(23)) 152,151,151
3700
3110
      151 VFLOw(19)=(PC(19)-PC(23))/R(19)
3700
          LVLV23=0
          GO TO 160
3790
3800
      152 VFLOW(19)=0
3010
          LVLV23=1
5820
          GU TO 160
        VALVE TEST FROM LAST BDRIVS EXECUTED IS USED
36300
3840 153 IF (LVLV23) 154,155,150
      154 WR11E (0,1144)
3050
3000 1144 FURMAT ('LVLV23 IS NEG .. NOT POSSIBLE')
3070
      155 VFLOW(19) = (PC(19) - PC(23))/R(19)
3000
          GO TO 157
3890
      156 VFLOW(19)=0
      157 CUNTINUE
3900
3910
      160 VFLOW(20) = (PC(20) - PC(24))/R(20)
           VFLOW(21) = (PC(21) - PC(1))/R(21)
3920
           VFLOW(22) = (PC(22) - PC(1))/R(22)
3950
3940
           VFLOW(23) = (PC(23) - PC(24))/R(23)
3450
           VFLOW(24) = (PC(24) - PC(2))/R(24)
3950
           DV(1) = VFLOW(22) + VFLOW(21) - VFLOW(1)
3970
           DV(2) = VFLOW(24) + VFLOW(1) - VFLOW(2)
3930
          DV(4) = VFLOW(5) - VFLOW(4)
3990
      223 CONTINUE
4000
           DO 15 N=5,10
```

¥

```
15 DV(N) = VFLIN(N) - VFLOW(N)
4010
4020
          DV(11) = VFLIN(11) + VFLOW(5) + VFLOW(6) + VFLOW(7) + VFLOW(8) + VFLOW(9) +
4030
         1VFLOW(10) - VFLOW(11)
4040
          DV(12)=VFLOW(11)-VFLOW(12)
          DV(14)=VFLOW(13)-VFLOW(14)
4050
4060
      253 CONTINUE
          DV(15)=VFLIN(15)-VFLOW(15)
4070
4080
          DV(10) = VFLIN(10) - VFLOW(16)
4090
          DV(17)=VFLIN(1/)-VFLOW(17)
4100
          DV(1b) = VFLIN(1b) - VFLOW(1b)
          DV(19)=VFLOW(15)-VFLOW(19)
4110
4120
          DV(20) = VFLOW(15) - VFLOW(20)
          DV(21) = VFLOW(17) - VFLOW(21)
41.30
4140
          DV(22) = VFLOW(10) - VFLOW(22)
4150
          DV(23) = VFLOW(19) - VFLOW(23)
4160
          DV(24) = VFLOV(25) + VFLOW(20) - VFLOW(24)
41/0
          RETURN
4180
          END
4190
          SUBROUTINE RESIST(LOPR.RK.RINK)
4200
          IMPLICIT REAL*8 (A-H.O-Z)
          COMMON /BCMELK/ R.FKDA.PP,VO,A2DM2,A2DM3,A2DM4,A2DM12,A2DM15,
4210
4220
         1A2DM14, H, RHO, G, RIN, P, PC
4250
          COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GP02,CT02,GPC2,CTC2
4240
          COMMON /CEN11/GPCO.CTCU.GPX.CTX
4250
          COMMON /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGP02,DCT02,
4260
         1DGPC2.DCTC2
4270
          COMMON /CRIN2T/DGPCO,DCTCO,DGPX,DCTX
4200
          COMMON /VFLT/VFLOW.VFLIN
4290
          DIMENSION R(24), FKDA(24), PP(24), V(24), VO(24), H(24), DV(24),
4300
         1VFLOW(24), VFLIN(18), RIN(18), PC(24), P(24)
4310
          DIMENSION RK(24), RINK(10)
4320
          DIMENSION GP02(24), GPC2(24), GPC0(48), GPX(1,24), CTC2(4), CTC2(4),
4550
         1CTCO(4), CTX(1,4), D1FO2(10), D1FC2(10), D1FCO(10), D1FX(10), CPH(24).
         2LCO2(6), LCC2(6), LCCO(6), LCX(1,6), CSATO2(24), CSATO2(24), DGPO2(24).
4340
4350
         3DGPC2(24), DGPCU(45), DGPX(1,24), AX(1)
       LOPR=NEG MEANS R=K/V**2+K/V
45600
4370C
        LOPR=O MEANS R=CONST=RK
4380C
         LOPR=POS MEANS R=RK/V**2 WHERE INPUT R IS USED FOR RK
4390
           IF (LOPR) 1.2.3
4400
        1 WRITE (6,2000)
4410 2000 FORMAT (' LOPR 1S NEG, NOT POSSIBLE YET')
       THIS WILL EVENTUALLY INCLUDE RESISTANCES OF THE TYPE R=K/V**2+K/V
4420C
4430
           RETURN
4440
        2 CONTINUE
4450C
       THIS WILL LEAVE R UNCHANGED FROM THE GIVEN VALUES (I.E. K=CONST)
4450
           RETURN
4470
        3 CONTINUE
           R(1)=RK(1)/(V(1)*V(1))
4480
4490
           DO 41 N=4.11
4500
       41 R(N) = RK(N) / (V(N) * V(N))
```

```
ready
```

*

```
4510
          DO 42 N=14.24
4520
       42 R(N) = RK(N) / (V(N) * V(N))
4530
          DO 32 N=5.11
4540
       32 RIN(N)=RINK(N)/(V(4)*V(4))
4550
          DO 33 N=15.18
       53 RIN(N)=RINK(N)/(V(14)*V(14))
4560
4570C
       THIS CALCULATES NEW VALUES OF R AND RIN USING R=RK/V**2
4500
          RETURN
4590
          ENTRY CRES(LOPR.RK.RINK)
          IF (LOPR.LE.O) GO TO 20
4600
4610
          RK(1) = R(1) \times V(1) \times V(1)
4020
          DO 10 N=4.11
       18 RK(N) = R(N) = V(N) = V(N)
4630
4640
          DO 19 N=14.24
4650
       19 RK(N) = R(N) * V(N) * V(N)
4660
          DO 16 N=5.11
4670
       16 RINK(N) = RIN(N) * V(4) * V(4)
4000
          DO 17 N=15,18
       17 RINK(N) = RIN(N) * V(14) * V(14)
4690
4700
       20 CONTINUE
4710
          RETURN
4720
          END
4730
          SUBROUTINE PHEART(PSDUR, PSAMP, PSRT, TB, FF, PSTO, Z, PSEND, PSTON)
4740
          IMPLICIT REAL*3 (A-H.O-Z)
4/50
          COMMON /AVE/ PPAVE
          IF (TB.LT.PSTO) GO TO 1
4760
4770
          PSTON=PSTO
4700
          PSTU=PSTO+60./PSRT
4790
          Z=b0./PSRT-PSDUR
4000
          PSEND=PSTU-Z
4810
          GO TO 2
4020
        1 lF(Tb.LT.PSEND) GO TO 2
4330
          FF=PPAVE
4840
           RETURN
4550
        2 CONTINUE
           SIGMA=(PSDUR/4.)**2
4000
           A=-(TE-PSTON-PSDUR/2.)**4/(2.*(SIGMA)**2)
4870
4800
           FF=PSAMP*EXP(A)+PPAVE
4890
      102 CONTINUE
4400
           RETURN
4910
           ENTRY CHEART (PSDUR, PSAMP, PSRT, Z, PSEND, PSTO)
4920
           PSTUN=PSTU-60./PSRT
4930
           Z=60./PSHT-PSDUR
4940
           PSEND=PSTO-Z
4950
           RETURN
4900
           END
4970
           SUBROUTINE PCHART(I,LVALV2,LVALV3,MTEST,PST,TES,TB)
4980
           IMPLICIT REAL*3 (A-H.O-Z)
4990
           COMMON /BCMBLK/ R,FKDA, PP, VO, A2DM2, A2DM3, A2DM4, A2DM12, A2DM13,
5000
          1A2DM14, H, RHO, G, RIN, P, PC
```

*

```
5010
                     COMMON /CMN1/V, VP2, VP3, VP4, VP12, VP13, VP14, GP02, CT02, GPC2, CTC2
5020
                     COMMON /CEN1T/GPCO.CTCO.GPX.CTX
5030
                     COMMON /CHN2/DV.DVP2.DVP3.DVP4.DVP12.DVP13.DVP14.DGP02.DCT02.
5040
                   1DGPC2.DCTC2
5050
                     COMMON /CMN2T/DGPCO.DCTCO.DGPX.DCTX
5060
                     COMMON /KNT/KN1B
5070
                     COMMON /SK/ TAU.PO.P1
5080
                     COMMON /VELT/VELOW.VELIN
                     DIMENSION R(24).FKDA(24).PP(24).V(24).VO(24).H(24).DV(24).
5090
5100
                   1VFLOW(24), VFLIN(10), RIN(10), PC(24), P(24), VP(14)
5110C
              WARNING...
                                         THIS SUBROUTINE HAS A DIFFERENT CIMENSION. . VP(14) ADDED
5120
                     DIMENSION TAU(24), PO(24), P1(24)
                     DIMENSION GPU2(24), GPC2(24), GPCO(48), GPX(1,24), CTU2(4), CTU2(4),
5130
5140
                   1CTCU(4).C1X(1,4).D1F02(10).D1FC2(10).D1FCU(10).D1FX(10).CPH(24).
5150
                   2LCO2(6), LCC2(b), LCCO(6), LCX(1,6), CSATO2(24), CSATO2(24), LCCO(6), LCX(1,6), CSATO2(24), CSATO2(24), LCCO(6), LCCO(6), LCX(1,6), CSATO2(24), CSATO2(24), LCCO(24), L
5160
                   3DGPC2(24), DGPCU(45), DGPX(1,24), AX(1)
5170
                     DIMENSION DCTO2(4), DCTO2(4), DCTO(4), DCTX(1,4), PETO2(4), PETO2(4)
5160
                     VP(3) = VP3
5190
                     VP(13)=VP13
5200
                     1F(1-2)1.2.3
5210
                1 WRITE (6.111)
5220
            111 FORMAT('SUBROUTINE PCHART HAS BEEN CALLED IN ERROR')
5230
                     RETURN
5240
                2 SINUR=1/R(1)+1/R(24)
5250
                     SINPDR=PC(1)/R(1)+PC(24)/R(24)
5200
                                                  1/RIN(5)+1/RIN(6)+1/RIN(7)+1/RIN(6)+1/RIN(9)+1/
                     SOUTDR=
5270
                   1RIN(10) + 1/RIN(11)
5280
                     SUUTPR=PC(5)/RIN(5)+PC(\delta)/RIN(\delta)+PC(\gamma)/RIN(\gamma)+PC(\sigma)/RIN(\sigma)+
5290
                   1PC(9)/RIN(9)+PC(10)/RIN(10)+PC(11)/RIN(11)
5300
                    GO TO 10
5310
                3 SINDR=1/R(11)
5320
                     SINPDR=PC(11)/R(11)
5330
                     SOUTDR=
                                                    1/RIN(15)+1/RIN(16)+1/RIN(17)+1/RIN(13)
5540
                     SOUTPR=PC(15)/Rin(15)+PC(16)/RIN(16)+PC(17)/Rin(17)+PC(10)/Rin(10)
5350
               10 J = 1 + 1
5360
                     K = 1 + 2
5370
                     IF (TB.LT.0.00.0R.TB.GT.0.05) GO TO 201
5300
                     WRITE (6,1000) TB. MTEST, PC(1), PC(J), PC(K)
5390
                   1,PP(I),PP(J),PP(K),P1(I),P1(I),P1(K),Rin(I),Rin(J),Rin(K),
5400
                   2H(1),H(J),H(K),SINPDR,SINDR,VP(J),SOUTDR,SOUTPR,KHO,G
5410 1000 FORMAT (' TB,MfEST,PC(2,3,4)= ',F10.4,I2,3F10.4
5420
                   1, /, 1X, 12F10.4, /, 1X, 12F10.4)
5430
            201 CONTINUE
                     1F (TB.LE.O.O) GO TO 33
5440
5450
                     1F (MTEST.EQ.1) GO TO 9
5460
                     IF (LVALV3.EQ.1) GO TO 310
5470
                     VALTSJ = -VP(J)
5480
                     GO TO 320
5490
            310 VALTSJ=PC(J)-PC(K)
5500
            320 CONTINUE
```

5510	IF (PC(1)-PC(J).LT.O.O.AND.VALTSJ.LT.O.O) GO TO 46
5520	IF (PC(I)-PC(J).LT.0.0) GO TO 21
5530	IF (VALTSJ.LT.O.O) GO TO 33
5540	GO TO 11
5550	9 IF (LVALV2.NE.O.AND.LVALV3.NE.O) GO TO 46
5560	IF (LVALV2.NE.O) GO TO 21
5570	IF (LVALV5.NE.O) GO TO 33
5500	11 CONTINUE
5590	WRITE (6,1035) TB,I,J,K
5600	1035 FORMAT (' BOTH VALVES ARE OPENNOT WORKING NOW,
5010	1TB,1,J,K='F20.5,514)
5620	STOP
5030	21 P(1)=(PP(1)+P1(1)-R1N(1)*S1NPDR+RIN(1)*S1NDR*
5540	1RHO*G*H(1))/(1.0+RIN(1)*SINDR)
5650	P(J)=PP(J)+P1(J)-RIN(J)*(-VP(J))
5650	P(K) = (PP(K) + P1(K) + RIN(K) * (-VP(J))
5670	1-RIN(K)*RHO*G*H(K)*SOUTDR+RIN(K)*SOUTPR)/(1.0+RIN(K)*
5000	2SOUTDR)
5690	PC(1)=P(1)+RHO*G*H(1)
5700	PC(J) = P(J) + RHU # G # H(J)
5710	PC(K) = P(K) + RHO * G * H(K)
5720	LVALV2=1
5730	LVALV3=0
5740	GO TO 47
5750	33 $P(K) = (PP(K) + P1(K) - RIN(K) * SOUTPR + RIN(K) *$
5700	1SOUTDR*RHO*G*H(K))/(1.0+RIN(K)*SOUTDR)
5770	PC(K) = P(K) + RhO*G*H(K)
5700	P(I) = (PP(I) + P1(I) + RIN(I) * SINPDR - RIN(I) *
5790	1RHO*G*H(1)*SINDR+(RIN(1)/(R(1)+RIN(J)))*(PP(J)+
5800	2P1(J)+RHO*G*H(J)-RHO*G*H(1)))/(1.0+
5510	3RIN(1)*SINDR+RIN(1)/(R(1)+RIN(J)))
5820	PC(1)=P(1)+RHO*G*H(1)
5030	P(J) = R(I) / (RIN(J) + R(I)) * (P1(J) + RIN(J) / R(I) * PC(I) + A) = A + A + A + A + A + A + A + A + A + A
5840	$\frac{1}{2} \frac{1}{2} \frac{1}$
5050	PU(J)=P(J)+KHU*G*H(J)
5000	
5010	
5000	
5090	40 CUNIINUE D(I) (D)(I) 04(I) DIN(I) BOINDED DIN(I) BOINDEA
5900	P(I) = (PP(I) + P(I) + RIN(I) * SINPDR - RIN(I) * SINDR *
5910	
5920	$P(K) = (PP(K) + P(K) - RIN(K) \circ SUUIPR + RIN(K) \circ$
5930	$\frac{15001DR*RH0*G*H(K)}{(1.0+RIN(K)*S001DR)}$
2940 EDEO	$\frac{1}{10} \frac{1}{10} \frac$
5950	IELLVALVZ.EW.I.AND.LVALVJ.EW.IJ GO IO 140 Tegete
5900	100=10 TR(INAIND RO 1) CO TO 105
2310	14'\UVRUV2,EQ.) GU 10 4") DSm=1 (2#D(1) D3(1) D1(Τ)
5000	101-1.2"F(1)-FF(0)-FF(0) G() TO 146
7220	
2000	

.

.

*

•

```
146 TF (TB-TBS.GT.1.0) GO TO 147
6010
          P(J)=PST*EXP(-40.*(TB-TBS))+PP(J)+P1(J)
6020
6030
          GO TO 148
6040
      147 P(J) = PP(J) + P1(J)
6050
      148 CONTINUE
          PC(I)=P(I)+RHO*G*H(I)
6060
6070
          PC(J)=P(J)+RHO*G*H(J)
6080
          PC(K) = P(K) + RHO * G * H(K)
6090
          LVALV2=1
6100
          LVALV3=1
6110
       47 CONTINUE
6120
          RETURN
6130
          END
6140
           SUBROUTINE SKON(LOPP1.TB)
6150
          IMPLICIT REAL*8 (A-H.O-Z)
6160
          COMMON /BCMBLK/ R.FKDA, PP, VO, A2DM2, A2DM3, A2DM4, A2DM12, A2DM13,
6170
         1A2DM14.H.RHO.G.RIN.P.PC
6180
          COMMON /CMN1/V.VP2.VP3.VP4.VP12.VP13.VP14.GP02.CT02.GPC2.CTC2
6190
          COMMON /CMN1T/GPCO,CTCO,GPX,CTX
6200
          COMMON /CMN2/DV, DVP2, DVP3, DVP4, DVP12, DVP13, DVP14, DGP02, DCT02,
6210
          1DGPC2.DCTC2
6220
          COMMON /CMN2T/DGPCO.DCTCO.DGPX.DCTX
6230
          COMMON /SK/ TAU.PO.P1
6240
          COMMON /VFLT/VFLOW,VFLIN
          DIMENSION R(24), FKDA(24), PP(24), V(24), VO(24). H(24). DV(24).
6250
6260
          1VFLOW(24), VFLIN(18), RIN(18), PC(24), P(24)
6270
          DIMENSION TAU(24), PO(24), P1(24)
          DIMENSION GP02(24), GPC2(24), GPC0(48), GPX(1,24), CT02(4), CT02(4),
6280
6290
          1CTCO(4), CTX(1,4), DIFO2(10), DIFC2(10), DIFCO(10), DIFX(10), CPH(24),
6300
         2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24),DGPO2(24),
6310
          3DGPC2(24), DGPCO(48), DGPX(1,24), AX(1)
           DIMENSION DCTO2(4), DCTC2(4), DCTCO(4), DCTX(1,4), METO2(4), METC2(4).
6320
6330
           IF (LOPP1.EQ.0) GO TO 100
6340
          DO 9 N=1.24
6350
          IF (TAU(N).LT.1.D-02) GO TO 9
6360
           VMAX = VO(N) + .224 * TAU(N)
6370
           IF (V(N).GT.VMAX) GO TO 8
           GO TO 9
6380
6390
        8 WRITE (6,1001) TB,N,V(N),VMAX
6400 1001 FORMAT (' TB,N,V(N),VMAX=',F10.4,I6,2F10.4,'....V TOO BIG.STOP.')
6410
           STOP
6420
         9 CONTINUE
6430
           P1(1)=PO(1)-ALOG(1.0-(V(1)-VO(1))/(.224*TAU(1)))/FKDA(1)
6440
           P1(2)=PO(2)-ALOG(1.0-(V(2)-VO(2))/(.224*TAU(2)))/FKDA(2)
           P1(4)=PO(4)-ALOG(1.0-(V(4)-VO(4))/(.224*TAU(4)))/FKDA(4)
6450
6460
           P1(11)=P0(11)-ALOG(1.0-(V(11)-V0(11))/(.224*TAU(11)))/FKDA(11)
6470
           P1(12)=P0(12)-ALOG(1.0-(V(12)-V0(12))/(.224*TAU(12)))/FKDA(12)
6480
           P1(14)=PO(14)-ALOG(1.0-(V(14)-VO(14))/(.224*TAU(14)))/FKDA(14)
6490
           DO 10 N=19,24
           P1(N) = PO(N) - ALOG(1.0 - (V(N) - VO(N)) / (.224 + TAU(N))) / FKDA(N)
6500
```

10 CONTINUE 6510 6520 P1(3) = FKDA(3) * (V(3) - VO(3))6530 P1(13) = FKDA(13) * (V(13) - VO(13))6540 DO 15 N=5.10 6550 P1(N) = FKDA(N) * (V(N) - VO(N))6500 15 CONTINUE DO 20 N=15.18 6570 P1(N) = FKDA(N) * (V(N) - VO(N))6560 6590 20 CONTINUE 6500 RETURN 0610 100 CONTONUE 6520 DO 110 N=1.24 P1(N) = FKDA(N) * (V(N) - VO(N))6630 6640 **110 CONTINUE** 6650 RETURN 6660 END 6670 SUBROUTINE BCDRTV(LOPGS) 6650 IMPLICIT REAL*3 (A-H.O-Z) 6690 LAMBO2, LANBC2, LAMBCO, LAMBX, LCO2, LCC2, LCC0, LCX, METO2, METC2 REAL*O 6700C WHEN ADDINT THIS ROUTINE TO BMAIN, REMOVE THE SECOND DIM. STATEMENT AND REPLACE 11 WITH THE ONE IN BMAIN CONTAINING THESE COANTAILES 6710C 6720 DIMENSION GPU2(24), GPC2(24), GPCU(48), GPX(1,24), CTU2(4), CTU2(4), 6730 1CTCO(4), CTX(1,4), DIFO2(10), DIFC2(10), DIFCO(10), DIFX(10), CPH(24), 6740 2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24),DGPO2(24), 6750 3DGPC2(24), DGPCU(43), DGPX(1.24), AX(1)0700 DIMENSION VFLIN(16), VFLOW(24), DV(24), V(24) 6770 DIMENSION DCTO2(4), DCTC2(4), DCTCO(4), DCTX(1,4), METO2(4), METC2(4) 6700 DIMENSION CTLO2(18), CTLC2(18), CTLCO(18), CTLX(1,18) COMMON / LCONLY/LCO2, LCC2, LCC0, LCX 6790 6800 COMMON /BC/LAMBO2,LAMBC2,LAMBCO,LAMBX,AO2,AC2,AC0,AX,HEMAT,DIFO2, 6610 1D1FC2.D1FC0.D1Fx.CPH.CSAT02.CSATC2.MET02.METC2 o820 COMMON /CMN1/V, VP2, VP3, VP4, VP12, VP13, VP14, GP02, CT02, GPC2, CTC2 6830 CUMMON /CMN1T/GPCO, CTCO, GPX, CTX 6840 COMPLON /CMN2/DV, DVP2, DVP3, DVP4, DVP12, DVP13, DVP14, DGP02, DC102, 6850 1DGPC2,DCTC2 0860 COMMON /CMN2T/DGPCO,DCTCO,DGPX,DCTX 0070 COMMON /BCOVER/CTL02, CTLC2, CTLC0, CTLX 0600 COMMON /VFL1/VFLOW,VFLIN 6890 DU 101 N=1.6 6900 CTLO2(N+4) = LCO2(N)6910 CTLC2(N+4) = LCC2(N)6920 CTLCO(N+4) = LCCO(N)6930 CTLX(1.N+4)=LCX(1.N)101 CONTINUE 6940 6950 DO 102 N=1.4 6960 CTLO2(N+14)=CTO2(N)CILC2(N+14)=CTC2(N)**6970** 6980 CTLCO(N+14)=CTCO(N)6990 CTLX(1, N+14) = CTX(1, N)7000 **102 CONTINUE**

ready

×

```
7010
          GO TO (1.2.3.4.5).LOPGS
7020C
        LOPGS = 1....02.02
        LOPGS = 2....02, C2, X
70's0C
7040C
        LOPGS = 3....X
        LOPGS = 4....02.CI.CO
70500
70500
        LOPGS = 5....02.C2.C0.X
7070
        1 CALL O2DRIV(LOPGS)
           DO 10 N=1.4
7050
7090
           1F (CTO2(N).LE.O.O) GO TO 201
           DCTO2(N) = -DIFO2(N+\delta)*(CTO2(N) - LAMEO2*GPO2(N+14)) - METO2(N)
7100
7110
           GO TU 202
7120 201 DCTO2(N)=-DIFO2(N+6)*(CTO2(N)-LAMBO2*GPO2(N+14))
7130
      202 CONTINUE
           1F (CTC2(N).LE.O.O) GO TO 203
7140
7150
           DCTC2(N) = -DIFC2(N+D) \times (CTC2(N) - LAMbC2 \times GPC2(N+14)) + METC2(N)
7160
           GO TO 204
7170 203 DCTC2(N)=-D1FC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))
7100
     204 CONTINUE
7190
       10 CONTINUE
7200
           GO TO 500
7210
        2 CALL G2DR1V(LOPGS)
           CALL XDRIV(LOPGS)
7220
7230
           DU 11 N=1.4
7240
           IF (CTO2(N).LE.O.O) GO TO 301
           DCTO2(N) = -D1FO2(N+6)*(CTO2(N)-LAMBO2*GPO2(N+14))-METU2(N)
7250
7260
           GO TU 302
     301 \text{ DCTU2}(N) = -D1F02(N+6)*(CT02(N) - LAMB02*GP02(N+14))
7270
72öu
      302 CONTINUE
7290
           IF (CTC2(N).LE.O.O) GO TO 303
           DCTC2(N) = -D1FC2(N+\delta)*(CTC2(N)-LAMBC2*GPC2(N+14))+METC2(N)
7300
7310
           GO TO 304
      303 \text{ DCTC2(N)} = -\text{DIFC2(N+6)} (\text{CTC2(N)} - \text{LAMBC2} + \text{GPC2(N+14)})
7320
7530
      304 CONTINUE
           DCTX(1,N) = -D1FX(-N+6)*(CTX(1,N)-LAMBX*GPX(1,N+14))
7340
7350
        11 CONTINUE
7300
           GO TO 600
7370
         3 CALL XDRIV(LOPGS)
           DO 12 N=1.4
7300
           DCTX(1,N) = -D1FX(N+\delta) * (CTX(1,N) - LAMBX * GPX(1,N+14))
7390
7400
        12 CONTINUE
7410
           GO TO 600
7420
         4 CALL O2DRIV(LOPGS)
7430
           DU 13 N=1.4
           IF (CTO2(N).LE.O.O) GO TO 401
7440
           DCTO2(N) = -DIFO2(N+\delta)*(CTO2(N) - LAMBO2*GPO2(N+14)) - METO2(N)
7450
7460
           GO 10 402
7470
       401 DCTO2(N) = -DIFO2(N+6) * (CTO2(N) - LAMBO2*GPO2(N+14))
7460
       402 CONTINUE
           1F (CTC2(N).LE.O.O) GO TO 403
7490
           DCTC2(N) = -D1FC2(N+6)*(CTC2(N) - LAMBC2*GPC2(N+14)) + METC2(N)
7500
```

122

ready

×

7510		GO TO 404
7520	403	<pre>DCTC2(N)=-DIFC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))</pre>
7530	404	CONTINUE
7540		DCTCO(N) = -D1FCO(N+6) * (CTCO(N) - LAMBCO * GPCO(N+14))
7550	13	CONTINUE
7560		GO TO 500
7570	5	CONTINUE
7580		CALL O2DRIV(LOPGS)
7590		CALL XDRIV(LOPGS)
7600		DO 14 N=1,4
7610		IF (CTO2(N).LE.O.O) GO TO 701
7020		DCTO2(N)=-D1FO2(N+o)*(CTO2(N)-LAMBO2*GPO2(N+14))-METO2(N)
7030		GU TO 702
7040	701	DCT02(N)=-D1F02(N+6)*(CT02(N)-LAMB02*GP02(N+14))
7650	702	CONTINUE
7000		1F (CTC2(N).LE.O.O) GO TO 703
7070		DCTC2(N)=-D1rC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))+METC2(N)
7680		GO TO (04
7690	703	DCTC2(N)=-D1FC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))
7100	704	CONTINUE
7/10		DCTCU(N)=-D1FCO(N+6)*(CTCO(N)-LAMBCO*GPCO(N+14))
1720		DCTX(1,N)=-D1FX(N+6)*(CTX(1,N)-LAMBX*GPX(1,N+14))
7730	14	CONTINUE
7740	600	CONTINUE
7750		DO 501 N=1,6
7700		LCO2(N) = CTLO2(N+4)
7770		LCC2(N)=C1LC2(N+4)
7780		LCCO(N) = CTLCO(N+4)
7790		LCX(1,N)=CTLX(1,N+4)
7000	50 1	CONTINUE
7810		DO 502 N=1,4
7820		CTO2(N)=CTLO2(N+14)
7830		CTC2(N)=CTLC2(N+14)
7040		CTCO(N) = CTLCO(N+14)
7850		CTX(1,N) = CTLX(1,N+14)
7860	502	CONTINUE
7010		RETORN
7880		END
7090		SUBROUTINE O2DRIV(LOPGS)
7900		IMPLICIT REAL*o (A-H,C-Z)
7910		REAL*3 LAMBU2, LAMBC2, LAMBCO, LAMBX, LCO2, LCC2, LCCU, LCX, METU2, METC2
7920		DIMENSION CTLO2(18), CTLC2(18), CTLCO(18), CTLX(1,18)
7930		DIMENSION GPO2(24), GPC2(24), GPCO(46), GPX(1,24), CTC2(4), CTC2(4),
7940		1CTCO(4),CTX(1,4),DIFO2(10),DIFC2(10),DIFCO(10),DIFX(10),CPH(24),
7950		2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24),DGPO2(24),
7900		3DGPC2(24),DGPCO(48),DGPX(1,24),AX(1)
7970		DIMENSION VFLIN(10), VFLOW(24), DV(24), V(24)
7930		DIMENSION SS(2),S(2,2),EKO2(24),EKC2(24),EKCO(24),CO2(24),
7990		1002(24),000(24),HMSP(24)
8000		DIMENSION DCTO2(4),DCTC2(4),DCTCO(4),DCTX(1,4),METU2(4),HETC2(4)

•

×

.

ייו

.

b010 DIMENSIGN SCRIPE(24), SCRIPC(24), SCRIPD(24), DENOM(b020 COMMOW /ECUTATVELOW, VFLIN b030 COMMOW /ECUTATVELOW, VFLIN b040 COMMOW /ECUTATVELOW, VFLIN b040 COMMOW /ECUTATVELOW, VFLIN b040 COMMOW /ECUTATVELOW, CTC2, CTC2, CTC2, CTC2, CTC2, CTC2, DIFC0, DIFYC2, UF2, VF3, VF4, VF12, VF13, VF14, GF02, CTC2, GFC2, CTC2 b050 COMMON /CMM17/QFC0, CTC0, GFX, CTX b060 COMMON /CMM17/GFC0, CTC0, GFX, GTX b100 COMMON /CMM17/GFC0, CTC0, GFX, GTX b101 DEFC2, DCTC2 b103 COMMON /CMM27/DUFC0, DCTC0, DGFX, DCTX b104 D0 12 Ast 1, 24 b105 PC2=GFC2(N) b110 CFH(N) = PM1 b130 CSALC2(N) = SS(1) b140 D1 / F(LOPGS, LT, 4) GO TO 11 b200 CSATC2(N) = SS(1) b170 CALL SATUR(PPC2, PC2, HEMAT, SS, S, S1, PH1) b170 CALL SATUR(PO2, PC2, HEMAT, SS, S, S1, PH1) b171 CALL SATUR(PO2, PC2, HEMAT, SS, S, S1, PH1) b172 CALL SATUR(PO2, PC2, HEMAT, SS, S, S1, PH1) b1730 CSALC2(N) = SS(1) b1740 CFH1 SCATC2(N) = SGTATPA </th <th></th> <th></th>		
b020 COARGOM /VELT/VELOW, VELIN 8050 CONMON /ECVLANEO2, CTLC2, CTLC0, CTLX 8050 COMMON /ECVLANEO2, LAMEC2, LAMECO, LAMEX, AO2, AC2, ACU, AX, HEHAT, D 8050 COMMON /ECVLANEO2, LAMEC2, LAMECO, LAMEX, AO2, AC2, ACU, AX, HEHAT, D 8050 COMMON /ECVLANEO2, LAMEC2, LAMECO, LAMEX, AO2, AC2, ACU, AX, HEHAT, D 8050 COMMON /CMN1T/GPCO, CTCO, GPX, CTX 8050 COMMON /CMN1T/GPCO, CTCO, GPX, CTX 8050 COMMON /CMN2T/GPCO, DCTCO, DGPX, DCTX 8110 ALPHA=1.339±-03 9120 ETLA=3, 93 9130 GAMME=1.00 9140 DG 9150 CSAT02(N)=SS(1) 8160 CP2=GP02(N) 8170 CSAT02(N)=SS(1) 8180 CP2=GP02(N) 8180 CP10(Q)=S15(1) 8190 CSAT02(N)=S15(2) 8190 CSAT02(N)=S15(1)	8010	DIMENSION SCRIPA(24),SCRIPB(24),SCRIPC(24),SCRIPD(24),DENOA(24)
d0j0 COMMON /ECOVER/CTL02 CTLC2, CTLC2, CTLC3, CTLC3, AC2, AC3, AX, HEWAT, D d0i0 COMMON /EC/LANEO2, LAMEC2, LAMEC0, LAMEX, A02, AC2, AC3, AX, HEWAT, D d0i0 1D1FC2, D1FC0, D1FX, CFH, CSAT02, CSAT02, MET02, MET02 d0i0 COMMON /Cam1/V, VP2, VP3, VP12, VP13, VP14, GP02, CT02, GPC2, CT02 d0i0 COMMON /Cam2/DV, DVP2, DVP3, DVP12, DVP13, DVP14, DGP02, DCT02 d0i0 COMMON /Cam2/DV, DVP2, DVP3, DVP4, DVP12, DVP13, DVP14, DGP02, DCT02 d0i0 COMMON /Cam2/DV, DVP2, DVP3, DVP4, DVP12, DVP13, DVP14, DGP02, DCT02 d0i0 COMMON /Cam2/DV, DVP2, DVP3, DVP4, DVP12, DVP13, DVP14, DGP02, DCT02 d0i0 COMMON /Cam2/DV, DVP2, DVP3, DVP4, DVP12, DVP13, DVP14, DGP02, DCT02 d0i0 COMMON /Cam2/DV, DVP2, DVP3, DVP4, DVP12, DVP13, DVP14, DGP02, DCT02 d0i0 COMMON /Cam2/DV, DVP2, DVP3, DVP4, DVP12, DVP13, DVP14, DGP02, DCT02 d10 CP4421, DC0 d10 CAM2/CAM2/DVP00, DVP2, DVP3, DVP4, DVP12, DVP13, DVP14, DGP02, DCT02 d10 CAM2/CAM2/DVP14, DVP13, DVP14, DGP02, DCT02 d10 CAM2/CAM2/DVP14, DVP14, DVP13, DVP14, DGP02, DCT02 d10 DC2=CP2(2) SC1141, D0 d10 DC2=CP2(2) d10 CAL2, AS1, PC2, PC2, HEMAT, SS, S, S1, PH1) <	6020	COMMON /VFLT/VFLOW,VFLIN
6040 CGMARON /EC/LAMEC2, LAMEC2, LAMEC2, LAMEC2, LAMEC2, METC2 8050 1D1FC2, D1FC0, D1FX, CPH, CSAT02, CSAT02, NETC2, METC2 8050 COMMON /CMN1/V, VP2, VP3, VP4, VP12, VP15, VP14, GP02, CT02, GPC2, CT02 8070 COMMON /CMN1/V, VP2, VP3, VP4, VP12, VP15, VP14, GP02, CT02, GPC2, CT02 8070 COMMON /CMN1/V, VP2, VP3, VP4, VP12, DVP13, DVP14, GP02, CT02, GPC2, CT02 8070 COMMON /CMN1/V, VP2, VP3, VP4, VP12, DVP13, DVP14, GP02, CT02, DCT02 8070 COMMON /CMN1/V, VP2, VP3, VP4, VP12, VP13, DVP13, DVP14, GP02, CT02 8070 COMMON /CMN1/V, VP2, VP3, VP4, VP12, DVP13, DVP14, GP02, CT02 8070 COMMON /CMN1/V, VP2, VP3, VP4, VP12, DVP13, DVP14, GP02, CT02 8070 Common / CMN1/V, VP2, VP3, VP4, DVP12, DVP13, DVP14, GP02, CT02 8070 Common / CMN1/V, VP2, VP3, VP4, DVP12, DVP13, DVP14, GP02, LC02 8170 CAMAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA	8020	COMMON /BCOVER/CTLO2,CTLC2,CTLCO,CTLX
0050 D1FC2,D1FC0,D1FX,CPH,CSATU2,CSATU2,KETU2,KETU2,KETU2 0050 COMMON /CMM17/PC0,CTC0,GFX,CTX 0050 COMMON /CMM27/DGPC0,DCTC0,DGPX,DVP1,DVP13,DVP14,DGP02,DCTU2 0050 COMMON /CMM27/DGPC0,DCTC0,DGPX,DCTX 0110 ALPHAE1.339E-03 0120 COMMON /CMM27/DGPC0,DCTC0,DGPX,DCTX 0110 COMMON /CMM27/DGPC0,DCTC0,DGPX,DCTX 0110 COMMON /CMM27/DGPC0,DCTC0,DGPX,DCTX 0111 ALPHAE1.339E-03 0120 COMMON /CMM27/DGPC0,DCTC0,DGPX,DCTX 0110 CALPAL Statu 0110 CALPAL Statu Statu 0120 CALPAL Statu Statu 0130 CSATU2(N)=SS(1) Statu Statu 0200 CSATU2(N)=SS(2) Statu Statu 0210 If (LOPGS,LT.4) GO TO 11 Statu Statu	6040	COMMON /BC/LAMBO2,LAMBC2,LAMBCO,LAMBX,AO2,AC2,ACO,AX,HEMAT,D1FC2,
δ050 COMMON / CHMITY, VP2, VP3, VP4, VP12, VP12, VP13, DVP14, DGP02, CIC2, GPC2, CIC δ070 COMMON / CHMITY/GPC0, CTC0, GPX, CTX δ070 COMMON / CHMITY/GPC0, DVP3, DVP4, DVP12, DVP13, DVP14, DGP02, DCIC2 δ070 COMMON / CHM2T/DGPC0, DCTC0, DGPX, DCTX δ170 COMMON / CHM2T/DGPC0, DCTC0, DGPX, DCTX δ170 CAMADA / CAM2T/DGPC0, DCTC0, DGPX, DCTX δ170 CALMATA / S39E-03 δ170 GAMAL=1.00 5170 GAMAL=1.00 5170 CALL SATUR(PO2, PC2, HEMAT, SS, S, S1, PH1) 6170 CALL SATUR(PO2, PC2, HEMAT, SS, S, S1, PH1) 6170 CALL SATUR(PO2, PC2, HEMAT, SS, S, S1, PH1) 6170 CALL SATUR(PO2, PC2, HEMAT, SS, S, S1, PH1) 6170 CALL SATUR(PO2, PC2, HEMAT, SS, S, S1, PH1) 6170 CALL SATUR(PO2, PC2, HEMAT, SS, S, S1, PH1) 6170 CSATU2(N) = SS(2) 6170 CSATU2(N) = SS(2) 6170 CSATU2(N) = SS(2) 6270 MAMAMK=1.0 62820 SATUPA(N=ALPHA HEMAT*NEWHAMAS1* GPC0(N)/A02 62820 SCATPE(N) = BATA*S(1, 1) 6290 SCATPE(N) = SCATA*S(1, 1) </td <td>8050</td> <td>1D1FC2,D1FC0,D1FX,CPH,CSAT02,CSATC2,MET02,METC2</td>	8050	1D1FC2,D1FC0,D1FX,CPH,CSAT02,CSATC2,MET02,METC2
6070 COMMON /CMNT/GPC0,CTC0,GPX,CTX 8080 CONMEN /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGP02,DCT02 8090 LDGPC2,DCTC2 8100 CCMMEND /CMN2/DGPC0,DCTC0,DGPX,DCTX 8110 ALPHA=1.339E-03 8120 B2TA=5.93 8130 GAMMA=1.00 8140 DG 12 N=1,24 8150 PC2=GPC2(N) 8140 CPH(N)=PH1 8140 CPH(N)=PH1 8140 CPH(N)=PH1 8140 CSATU2(N)=SS(1) 8200 CSATU2(N)=SS(2) 8210 IF (LOPGS,LT:4) GO TO 11 82200 CSATU2(N)=SS(2) 8210 IF (LOPGS,LT:4) GO TO 11 82201 DGPCU(N+24)=ACC*MEMMEM*S1* GPCO(N)/AO2 8240 DGPCU(N)=ALPHA+BETA*S1*GPCO(N) 8250 SCATPA(N)=ALPHA+BETA*S(1,1) 8260 SCATPA(N)=ALPHA+BETA*S(1,1) 8270 SCATPA(N)=ALPHA+BETA*S(1,1) 8280 SCATPA(N)=ALPHA+PO2+SS(1)*BETA 8300 SCATPA(N)=SCRP2(D,V,LAMBO2,DIFO2,CO2,GF02,EKO2) 8300 CALL GFLTST(CTLC2,DV,LAMBC2,DIFO2,CC2,G	8060	COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP15,VP14,GP02,CT02,GPC2,C1C2
8080 COMMON /CHM2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGP02,DCT02 6090 1DGP02,DCT02 6090 LDGP02,DCT02 6090 LDGP02,DCT02 8110 ALPHA=1.339E-03 6120 CCMHUN /CHM2T/DGP00,DCTC0,DGPX,DCTX 8111 ALPHA=1.339E-03 6120 DCHAMA=1.00 6140 DG 12 N=1.24 6150 GANMA=1.00 6140 DG 12 N=1.24 6150 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) 6160 CSAT02(N)=SS(1) 6200 CSAT02(N)=SS(2) 6210 IF (LDPG3,LT.4) GO TO 11 62200 CSAT02(N)=SS(2) 6210 IF (LDPG3,LT.4) GO TO 11 62200 CSAT02(N)=SS(2) 6210 DCPC0(N+24)=ACC*MMMAM*S1* GPC0(N)+AC2 6250 MMPAMK=1.0 6240 DCPC0(N+24)=ACC*MMMAM*S1* GPC0(N)+AC2 6250 CCO(N)=(1hEMAT)*ACC*GPC0(N)+HEMAT*GPC0(N+24) 6270 CCO(N)=(1hEMAT*S(1,2) 8280 SCRIPE(N)=SC(2,1)*PC2*GAMMA 8290 SCRIPE(N)=SC(2,1)*PC2*GAMMA	6070	COMMON /CMN1T/GPCO,CTCO,GPX,CTX
b090 1D0PC2,DCTC2 b100 CGMHUN /CMN2T/DGPC0,DCTC0,DGPX,DCTX b110 ALPHA=1,339b=03 b120 BETA=3.93 b130 GALPHA=1,339b=03 b140 D0 12 k=1,24 b150 PC2=GPC2(N) b170 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) b170 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) b170 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) b170 CALU2(N)=SS(1) b200 CSATC2(N)=SS(2) b210 IF (L0PGS.LT.4) GO TO 11 b220 MAPAMM41.0 b210 IF (L0PGS.LT.4) GO TO 11 b220 MAPAMM41.0 b220 MAPAMM41.0 b220 MAPAMM41.0 b210 IF (L0PGS.LT.4) GO TO C(N) / AO2 d520 SCRIPE(N)=BETA*S(1,2) b220 MAPAMM451*GPC0(N)+AO2 d520 SCRIPE(N)=SC1,1*PC2*GAMMA b310 SCRIPE(N)=SC2,2)*PC2*GAMMA b320 SCRIPE(N)=SC2,2)*GAMMA*PC2 b320 CC2(N)=SC3/*GAMMA*PC2 b320 CC2(N)=SCRIPC(N	8080	COMMON /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGP02,DCTU2,
b100 CGenow /CdM2T/DGPC0, DCTC0, DGPX, DCTX 8110 ALPHA=1.339b=03 8120 BZTA=3.93 b130 GAMMA=1.00 b140 DC 12 N=1,24 b150 PO2=GPC2(N) 8160 PC2=OPC2(N) b170 CALL SATUR(PC2, PC2, HEMAT, SS, S, S1, PH1) b170 CALL SATUR(PC2, PC2, HEMAT, SS, S, S1, PH1) b170 CALL SATUR(PC2, PC2, HEMAT, SS, S, S1, PH1) b170 CALL SATUR(PC2, PC2, HEMAT, SS, S, S1, PH1) b170 CALL SATUR(PC2, PC2, HEMAT, SS, S, S1, PH1) b170 CALL SATUR(PC2, PC2, HEMAT, SS, S, S1, PH1) b170 CALL SATUR(PC2, PC2, HEMAT, SS, S, S1, PH1) b170 CALL SATUR(PC2, PC2, HEMAT, SS, S, S1, PH1) b170 CALL SATUR(PC2, PC2, HEMAT, SS, S, S1, PH1) b200 CSATC2(N)=SS(2) b210 IF (LOPGS, LT.4) GO TO 11 b220 MAMAMAM=1.0 b220 MAMAMAM=1.0 b220 COC(N)=(N=TA)*ACC*APCO(N)+HEMAT*GPCO(N)/AO2 b210 DCWOTINE: ACC*APCO(N)+HEMAT*GPCO(N)+424) b220 SCRIPE(N)==ACC*APCO*APAMAS b	6090	1DGPC2,DCTC2
<pre>8110 ALPHA=1.339E-03 8120 BETA=5.93 8130 GAA:HA=1.00 8140 D0 12 N=1,24 8150 PO2=GP02(N) 8160 PC2=GP02(N) 8160 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) 8170 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) 8180 CPH(N)=PH1 8180 CPH(N)=S11 8180 CPH(N)=S11 8180 CPH(N)=S11 8200 CSATC2(N)=SS(1) 8210 IF (LOPGS,LT,4) GO TO 11 8220C AC2=.030/ EY THE PROGRAM SATUR 8250 M:Abademinet S1* GPC0(N)/A02 8250 D:GPC0(N+24)=AC0*PMEMMEM*S1* GPC0(N)/A02 8250 D:GPC0(N+24)=AC0*PMEMMEM*S1* GPC0(N)/A02 8250 D:GPC0(N)=H:AHA*HA#AH#A#S1* GPC0(N)+HEMAT*GPC0(N+24) 8260 CC0(N)=H:AHA*HA*ST* GPC0(N)+HEMAT*GPC0(N+24) 8270 11 CCWTINUE 8280 SCRIPA(N)=ALPHA*ESTA*S(1,1) 8290 SCRIPA(N)=ALPHA*ESTA*S(1,1) 8290 SCRIPA(N)=S(2,1)*PC2*GAMMA 8510 SCRIPC(N)=S(2,2)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA*DC2 8540 12 CCWTINUE 8550 CALL GPLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8570 CALL GPLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8570 CALL GPLTST(CTLC2,DV,LAMBC2,DIFC0,CC0,GPC0,EKC0) 8560 D: 50 N=1,24 8590 DENOM(N)=(SCRIPA(N)*SCRIPD(N)=SCRIPC(N)*SCRIPE(N))*V(N) 8410 D:GPC2(N)=(SCRIPA(N)*SCRIPD(N)=SCRIPC(N)*EKC2(N))/DEMOM(N) 8410 D:GPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DEMOM(N) 8410 D:GPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DEMOM(N) 8410 D:GPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DEMOM(N) 8410 D:GPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DEMOM(N) 8410 D:GPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DEMOM(N) 8410 D:GPC2(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 CONTINUE 8450 RETURN 8450 SUEROUTINE XDRIV(LOPGS) 8450 IMPLICIT HEAL*6 (A=H,0-2) 8450 IMPLICIT HEAL*6 (A=H,0-2) 8450 IMPLICIT HEAL*6 (A=H,0-2) 8450 IMPLICIT HEAL*6 (A=H,0-2) 8450 DIMENSION GPO2(24),GPC2(24),GPCC(HB),GPC(HB),GPC(H),CTC2(H),CTC2(H),CTC2(H),CTC2(H),CTC2(H),CTC2(</pre>	ს1 00	COMMON /CMN2T/DGPCO, DCTCO, DGPX, DCTX
<pre>b120 BCTA=3.93 b130 GANMA=1.00 b140 D0 12 N=1,24 b150 P02=GP02(N) b170 CALL SATUR(P02,P02,HEMAT,SS,S,S1,PH1) b170 CALL SATUR(P02,P02,HEMAT,SS,S,S1,PH1) b170 CALUSATUR(N)=PH1 b190 CSATU2(N)=SS(1) b210 IF (LOPGS,LT.4) GO TO 11 b220C AC2=.030/ LY THE PROGRAM SATUR b250 MAMAWA=1.0 b240 DJP0C(N+24)=ACO*PRIMAMA*S1* GPC0(N)/A02 b250 HabP(N)=HEMAT*MAMAMA*S1*GPC0(N) b250 CC0(N)=(1hEAT)*ACC*JPC0(N)+HEMAT*GPC0(N+24) b250 SCATPA(N)=ALDHA+BETA*S(1,1) b250 SCATPA(N)=ALDHA+BETA*S(1,1) b250 SCATPA(N)=SL2,1)*PC2*GAMMA b310 SCRIPD(N)=SL2,1)*PC2*GAMMA b310 SCRIPD(N)=SL2,1)*PC2*GAMMA b310 SCRIPD(N)=SL2,1)*PC2*GAMMA b310 SCRIPD(N)=SL2,1)*PC2*GAMMA b310 SCRIPD(N)=SL2,1)*PC2*GAMMA b310 SCRIPD(N)=SL2,2)*PC2*GAMMA b310 SCRIPD(N)=SL2,1)*PC2*GAMMA b310 SCRIPD(N)=SL2,2)*C2*GAMMA+SL2,2)*GAMMA b310 SCRIPD(N)=SL2,2)*C2*GAMMA+SL2,2)*GAMMA b310 SCRIPD(N)=SL2,2)*C2*GAMMA+SL2,2)*GAMMA b310 SCRIPD(N)=SL2,2)*C2*GAMMA+SL2,2)*GAMMA b310 SCRIPD(N)=SL2,2)*C2*GAMMA+SL2,2)*GAMMA b310 SCRIPD(N)=SL2,2)*C2*GAMMA+SL2,2)*GAMMA b310 SCRIPD(N)=SL2,2)*C2*GAMMA+SL2,2)*GAMMA b310 SCRIPD(N)=SCRIPA(N)*SCRIPC,0)*CC0,GPC2,EK02) b330 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,C02,GPC2,EK02) b330 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,C02,GPC2,EK02) b250 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,C02,GPC2,EK02) b250 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) b350 DENOM(N)=(SCRIPA(N)*EK02(N)-SCRIPC(N)*EK02(N)/DENOM(N) b420 LF(LOPGS,LT.4) GO TO 50 DGPC2(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) b440 50 COMTINUE b450 RETURN b440 SUEROUTINE XDRIV(LOPGS) b450 IMPLICIT REAL*6 (A+H,0-2) b450 REAL*3 LAMBC2,LAMBC2,LAMBC0,LAMBX,LC02,LCC2,LCC0,LCX,MET02, D1MENSION GPO2(2),GPC2(2),GPC2(2),GPC2(4),GPC2(4),GTC2(4),C</pre>	8110	ALPHA=1.3396-03
b130 GAMMA=1.00 b140 DG 12 N=1,24 b155 PO2=GP02(N) b170 CALL SATUR(PO2,PC2,HEMAT,SS,S,S1,PH1) b200 CALL SATUR(PO2,PC2,PC2,HEMAT,S1*GPC0(N)/AO2 b200 SCRIPC(N)=BETA*S(1,2) b200 SCRIPC(N)=ALPHA+BETA*S(1,1) b200 SCRIPC(N)=S(2,1)*PC2*GAMMA b310 SCRIPC(N)=S(2,1)*PC2*GAMMA b320 SCRIPC(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA b320 SCRIPC(N)=SCRIPA(N)=SCRIPC b340 SCCU(N)=SCRIPA(N)*EKC2(N)-SCRIPC(N)*SCRIPE(N)*EKC2) b340 12 CALL GFLTST(CTLC2, DV,LAMBC2,DIFC2,CC2,GPC2,EKC2)	o120	BETA=3.93
b140 DG 12 N=1,24 b150 PO2=GPO2(N) b170 CALL SATUR(PO2,PC2,HEMAT,SS,S,S1,PH1) b200 CSATU2(N)=SS(2) b211 F (LOPGS,LT.4) GO TO 11 b220 M=M=MAT,MEMMHM*S1*GPC0(N)/AO2 b231 M=M=MAT,MEMMHM*S1*GPC0(N)/AO2 b240 D_GCO(N)=1.4.PAC*GPC0(N)+HEMAT*GPC0(N+24) b241 CO(N)=1.1.PAC*GPC0(N)+HEMAT*GPC0(N+24) b340 SCRIPC(N)=SC2,1)*PC2*GAMAA b340 SCRIPC(N)=SC2,1)*PC2*GAMAA b350 SCRIPC(N)=SC2,1)*PC2*GAMAA b350 CC2(N)=SC2)*GAMAA*PC2 b340 12 CONTINUE b350 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) b350 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GP	8130	GAMMA=1.00
<pre>8153 P02=GP02(N) 8160 PC2=GPC2(N) 8160 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) 8170 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) 8170 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) 8170 CSATU2(N)=SS(1) 8280 CSATU2(N)=SS(1) 8290 CSATU2(N)=SS(2) 8210 IF (LOPGS.LT.4) GO TO 11 8220C AC2=.0307 bY THE PROGRAN SATUR 8250 MATAMAK=1.0 8240 DGPCU(N+24)=ACO*PARIMAMA*S1* GPCO(N)/AO2 8250 HASP(N)=HEMAT*NAMAMA*S1*GPCO(N) 8260 CCO(N)=(1hEAT*NAMAMA*S1*GPCO(N)+HEMAT*GPCO(N+24) 8270 11 CONTINUE 8280 SCRIPE(N)=BETA*S(1,1) 8290 SCRIPE(N)=BETA*S(1,2) 8300 SCRIPC(N)=S(2,2)*PC2*GAMMA 8310 SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA 8350 CC2(N)=ALPHA*P02+SS(1)*EETA 8350 CC2(N)=SS(2)*GAMMA*PC2 8360 C2(N)=ALPHA*PC2 8370 CALL GFTTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8360 CALL GFTTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8370 CALL GFTTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8360 DD 50 N=1,24 8390 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 8410 DGPC2(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPE(N)*CKC2(N)/DEMOM(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPE(N)*EKC2(N)/DEMOM(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPE(N)*EKC2(N)/DEMOM(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPE(N)*EKC2(N)/DEMOM(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPE(N)*EKC2(N)/DEMOM(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPE(N)*EKC2(N)/DEMOM(N) 8410 DGPC2(N)=(EKCU(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 S0 CONTINUE 8450 RETURN 8440 SUBROUTINE XDRIV(LOPGS) 8447 SUBROUTINE XDRIV(LOPGS) 8449 REAL*3 LAMBC2,LAMBC2,LAMBC2,LAMBC2,LCC2,LCC2,LCC0,LCX,METU2, DIMENSION GPO2(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT22(4),GPC2(44),GPC2</pre>	o 1 40	DO 12 N=1,24
<pre>8160 PC2=GPC2(N) 6170 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) 8140 CPH(N)=PH1 6190 CSATU2(N)=SS(1) 6200 CSATU2(N)=SS(2) 8210 IF (LOPGS.LT.4) GO TO 11 6220C AC2=.050/ bY THE PROGRAN SATUR 6230 MAPAMME1.0 8240 DGPC0(N+24)=ACO*AMAMMM*S1* GPC0(N)/A02 8250 HASP(N)=HEMAT*NAMAMM*S1*GPC0(N) 8260 CCO(N)=(1hEMAT)*ACC*GOPC0(N)+HEMAT*GPC0(N+24) 6270 11 CONTINUE 8260 SCRIPA(H)=ALPHA+BETA*S(1,1) 8290 SCRIPA(H)=ALPHA+BETA*S(1,1) 8290 SCRIPA(H)=BLPA*S(1,2) 8500 SCRIPC(N)=S(2,1)*PC2*GAMMA 8510 SCRIPD(N)=SC2,1)*PC2*GAMMA 8510 SCRIPD(N)=SC2,1)*PC2*GAMMA 8520 CC2(N)=SS(2)*GAMMA*PC2 8540 12 CONTINUE 8550 CC2(N)=SS(2)*GAMMA*PC2 8540 12 CONTINUE 8550 CALL GFLTST(CTLC2,DV,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTLC2,DV,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTLC2,DV,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTLC0,DV,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTLC0,DV,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTLC0,DV,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTLC0,DV,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTLC0,DV,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTL00,V,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTL00,V,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTL00,V,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTL00,V,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTL00,V,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTL00,V,LAMB02,DIF02,C02,GP02,EKO2) 8370 CALL GFLTST(CTL00,V,LAMB02,DIF00,SCRIPE(N))*V(N) 8470 DGP02(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPE(N)*EKC2(N))/DEMOM(N) 8440 DGP02(N)=(SCRIPA(N)*EKC2(N)-SCRIPE(N)*EKC2(N))/DEMOM(N) 8440 DGP02(N)=(SCRIPA(N)*EKC2(N)-SCRIPE(N)*EKC2(N))/DEMOM(N) 8440 DGP02(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 S0 CONTINUE 8450 RETORN 8440 SUBROUTINE XDRIV(LOPGS) 8440 SUBROUTINE XDRIV(LOPGS) 8440 SUBROUTINE XDRIV(LOPG2) 8440 HEAL*6 LAMB02,LAMBC2,LAMBC2,LAMBC,LC02,LCC2,LCC0,LCX,MET02, B440 HEAL*6 LAMB02,LAMBC2,LAMBC2,LAMBC3,LC02,LCC2,LCC0,LCX,MET02, B440 DIMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT22(4),GPC3</pre>	8 1 50	P02=GP02(N)
 b170 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) b170 CALL SATUR(P02,PC2,HEMAT,SS,S,S1,PH1) b170 CSATU2(N)=SS(1) b170 CSATU2(N)=SS(2) b210 IF (LOPGS.LT.4) GO TO 11 b2200 AC2=.050/ bY THE PROGRAN SATUR b230 MMPAMK=1.0 b240 DUPCO(N+24)=AC0*PEMAMMA*S1* GPCO(N)/AO2 b250 HASP(N)=HEMAT*MAMMA*S1*GPCO(N) b260 CCO(N)=(1,-hEMAT)*ACC*GPCO(N)+HEMAT*GPCO(N+24) b270 I1 CONTINUE b270 SCRIPE(N)=BETA*S(1,1) b270 SCRIPE(N)=BETA*S(1,2) b300 SCRIPE(N)=S(2,1)*PC2*GAMMA b310 SCRIPE(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA b320 CC2(N)=ALPAA*PC2+SS(1)*BETA b320 CC2(N)=ALPAA*PC2+SS(1)*BETA b320 CC2(N)=SS(2)*GAMMA*PC2 b340 12 CONTINUE cC4LL GFLTST(CTLO2,DV,LAMBO2,DIFO2,CO2,GP02,EKO2) cC4LL GFLTST(CTLO2,DV,LAMBO2,DIFO2,CC2,GP02,EKO2) cC4LL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) b350 D2 50 N=1,24 b50 D400(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) d400 D602(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*EKC2(N))/DEA0M(N) d400 D602(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DEA0M(N) b410 D4PC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DEA0M(N) b420 1F (LOPGS.LT.4) GO TO 50 d430 D50 CONTINUE b440 50 CONTINUE b440 50 CONTINUE b440 50 CONTINUE b440 SUBROUTINE XDRIV(LOPGS) b440 MPLICIT REAL*0 (A-H,O-Z) b440 MELLCIT REAL*0 (A-H,O-Z) 	8160	PC2=GPC2(N)
 8140 CPH(N)=PH1 8190 CSAT02(N)=SS(1) 8200 CSATC2(N)=SS(2) 8210 IF (LOPGS.LT.4) GO TO 11 8220C AC2=.030/ bY THE PROGRAN SATUR 8240 DGPC0(N+24)=AC0*PEMEMENTS1* GPC0(N)/A02 8250 HABP(N)=HEMAT*MAMMENTS1*GPC0(N) 8250 CC0(N)=(1EEAT)*ACC*GPC0(N)+HEMAT*GPC0(N+24) 8270 11 CONTINUE 8290 SCRIPE(N)=BETA*S(1,1) 8290 SCRIPE(N)=S(2,1)*PC2*GAMMA 8510 SCRIPP(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA 8510 SCRIPP(N)=S(2,1)*PC2*GAMMA+SS(2)*GAMMA 8520 CO2(N)=ALPHA*PC2*S(1)*BETA 8520 CO2(N)=ALPHA*PC2+SS(1)*BETA 8530 CC2(N)=SS(2)*GAMMA*PC2 8540 12 CONTINUE 8550 CALL GFLTST(CTLC2,DV,LAMBC2,DIFO2,C02,GP02,EK02) 8360 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GP02,EK02) 8370 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC0,CC0,GPC0,EKC0) 8360 DC 50 N=1,24 8390 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N)/DEAOM(N) 8420 IF (LOPGS.LT.4) GO TO 50 8440 S0 CONTINUE 8440 S0 CONTINUE XDRIV(LOPGS) 8440 S0 CONTINUE XDRIV(LOPGS) 8440 S0 CONTINUE XDRIV(LOPGS) 8440 S0 MOPLOTINE XDRIV(LOPGS) 8440 S0 CONTINUE XDRIV(LO	8170	CALL SATUR(PO2,PC2,HEMAT,SS,S,S1,PH1)
<pre>6190 CSAT02(N)=SS(1) 6200 CSAT02(N)=SS(2) 8210 IF (LOPGS.LT.4) GO TO 11 8220C AC2=.050/ bY THE PROGRAN SATUR 8230 MAPAMEME1.0 8240 D0PC0(N+24)=AC0*PAMAMM*S1* GPC0(N)/A02 8250 HASP(N)=HEMAT*MAMAMM*S1*GPC0(N) 8260 CC0(N)=HEMAT*MAMAMM*S1*GPC0(N)+HEMAT*GPC0(N+24) 8270 11 CCNTINUE 8260 SCRIPA(N)=ALPHA+BETA*S(1,1) 8290 SCRIPA(N)=BETA*S(1,2) 8500 SCRIPD(N)=BETA*S(1,2) 8500 SCRIPD(N)=S(2,1)*PC2*GAMMA 8520 C02(N)=ALPHA*P02*SS(1)*BETA 8530 CC2(N)=SS(2)*GAMMA*PC2 8540 12 CONTINUE 8550 C4LL GFLTST(CTL02,DV,LAMB02,DIF02,C02,GP02,EK02) 8540 C4LL GFLTST(CTL02,DV,LAMB02,DIF02,C02,GP02,EK02) 8540 C4LL GFLTST(CTL02,DV,LAMB02,DIF02,C02,GP02,EK02) 8540 C4LL GFLTST(CTL02,DV,LAMB02,DIF02,C02,GP02,EK02) 8540 C4LL GFLTST(CTL02,DV,LAMB02,DIF02,C02,GP02,EK02) 8540 C4LL GFLTST(CTL02,DV,LAMB02,DIF00,CC0,GPC0,EK00) 8540 D2 50 N=1,24 8590 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 0400 DGP02(N)=(SCRIPA(N)*EK02(N)-SCRIPC(N)*EK02(N))/DEA00m(N) 8410 DGPC2(N)=(SCRIPA(N)*EK02(N)-SCRIPC(N)*EK02(N))/DEA00m(N) 8420 IF (LOPGS.LT.4) GO TO 50 0430 DGP00(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 CONTINUE 8450 RETURN 8450 EKD 8470 SUBRODTINE XDRIV(LOPGS) 8450 INPLICIT REAL*8 (A-H,0-Z) 8450 INPLICIT REAL*8 (A-H,0-Z) 8450 EKD</pre>	8 1 30	CPH(N)=PH1
 6200 CSATC2(N)=SS(2) 6210 IF (LOPOS.LT.4) GO TO 11 6220C AC2=.030/ bY THE PROGRAN SATUR 62200 McMadMatt. 62200 DGPCO(N+24)=ACO*DEMEMENT*GPCO(N)/AO2 6250 Hosp(N)=HEMAT*MAMEMENT*GPCO(N) 6250 CCO(N)=(1hEMAT)*ACC*GPCO(N)+HEMAT*GPCO(N+24) 6270 11 CONTINUE 6270 11 CONTINUE 6280 SCRIPE(N)=BETA*S(1,1) 6290 SCRIPE(N)=BETA*S(1,2) 6300 SCRIPC(N)=S(2,1)*PC2*GAMMA 6310 SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA 6320 CO2(N)=ALPHA*PO2+SS(1)*BETA 6350 CC2(N)=SS(2)*GAMMA*PC2 6350 CALL GFLTST(CTLC2,DV,LAMEO2,DIFO2,CO2,GPU2,EKO2) 6350 CALL GFLTST(CTLC2,DV,LAMEC2,DIFC2,CC2,GPU2,EKO2) 6350 CALL GFLTST(CTLC2,DV,LAMEC2,DIFC2,CC2,GPU2,EKO2) 6350 CALL GFLTST(CTLC0,DV,LAMEC0,DIFC0,CC0,GPU2,EKO2) 6350 CALL GFLTST(CTLC0,DV,LAMEC0,DIFC0,CC0,GPU2,EKO2) 6350 CALL GFLTST(CTLC0,DV,LAMEC0,DIFC0,CC0,GPU2,EKO2) 6350 DD 50 N=1,24 6350 DD (SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 6400 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DEMOM(N) 6410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DEMOM(N) 6420 IF (LOPOS,LT.4) GO TO 50 6450 GETORN 6450 RETORN 6450 RETORN 6450 RETORN 6450 RETORN 6450 IMPLICIT REAL*6 (A-H,0-2) 760 REAL*6 LAME02,LAMEC2,LAMEC0,LAMEX,LC02,LCC2,LCC2,LCC0,LCX,METU2, 6500 DIAENDIN GPO2(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CTC2 	o190	CSATU2(N)=SS(1)
<pre>8210 IF (LOPGS.LT.4) GO TO 11 82200 AC2=.0307 bY THE PROGRAM SATUR 8250 MeMMedmed.0 8240 DGPCU(N+24)=ACO*meMedmed*S1* GPCO(N)/AO2 8250 Beap(N)=HemA1*MeMedmed*S1*GPCO(N) 8260 CCO(N)=(1hemA1)*AcC*GPCO(N)+HEMAT*GPCO(N+24) 8270 I1 CCNTINUE 8280 SCRIPA(N)=ALPHA+BETA*S(1,1) 8290 SCRIPB(N)=BETA*S(1,2) 8500 SCRIPC(N)=S(2,1)*PC2*GAMMA 8510 SCRIPC(N)=S(2,2)*PC2*GAMMA 8520 CO2(N)=ALPHA*PO2+SS(1)*BETA 8530 CC2(N)=SS(2)*GAMMA*PC2 8540 I2 CONTINUE 8550 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8360 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8360 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8360 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8360 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8360 D0 50 N=1,24 8590 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N)/DEMOM(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N)/DEMOM(N) 8420 IF (LOPGS.LT.4) GO TO 50 9450 DGPC0(N)=(EKCU(N)/(ACO*V(N))-HMSP(N))/(1.+HeMAT) 8440 50 CONTINUE 8450 RETURN 8440 SUBROUTINE XDRIV(LOPGS) 8450 IMPLICIT REAL*6 (A-H,O-Z) 8450 REAL*5 LAMBC2,LAMBC2,LAMBC,LAMBX,LCO2,LCC2,LCC0,LCX,METU2, 8500 DEMOM(S)(2(24),GPC2(24),GPC0(48),GPX(1,24),CTO2(4),CTC2)</pre>	8200	CSATC2(N)=SS(2)
 b220C AC2=.030/ bY THE PROGRAN SATUR b230 MARMAMET1.0 b240 DGPC0(N+24)=AC0*MAMMAMET1* GPCO(N)/AO2 b3260 BASP(N)=HEMAT*MAMMAMET1*GPCO(N) b250 CC0(N)=(1hEMAT*MAMMAMET1*GPCO(N)+HEMAT*GPCO(N+24) b270 11 CCNTINUE b280 SCRIPE(N)=BETA*S(1,1) b290 SCRIPE(N)=BETA*S(1,2) b300 SCRIPE(N)=S(2,1)*PC2*GAMMA b310 SCRIPO(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA b320 CO2(N)=ALPHA*PO2*SS(1)*BETA b320 CO2(N)=ALPHA*PO2*SS(1)*BETA b320 CO2(N)=ALPHA*PO2*SS(1)*BETA b320 CO2(N)=SS(2)*GAMMA*PC2 b340 12 CONTINUE b350 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) b350 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) b350 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC0,CC0,GPC0,EKC0) b50 DC 50 N=1,24 b590 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) b3420 DGP02(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N)/DENOM(N) b410 DGP02(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N)/DENOM(N) b420 IF (LOPGS.LT.4) GO TO 50 bGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) b6450 RÉTURN b440 SUBROUTINE XDRIV(LOPGS) b4470 SUBROUTINE XDRIV(LOPGS) b450 IMPLICIT REAL*8 (A-H,0-Z) b450 REAL*8 LANBO2,LAMEC2,LAMBCA,LCO2,LCC2,LCC0,LCX,METO2, b500 DENOM GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT02 	8210	IF (LOPGS.LT.4) GO TO 11
 b230 MARAMET1.0 b240 DGPC0(N+24)=ACO*MARAMET*S1* GPCO(N)/AO2 b350 HADSP(N)=HEMAT*MARAMET*S1*GPCO(N) b260 CCO(N)=(1HEMAT)*ACO*GPCO(N)+HEMAT*GPCO(N+24) b270 11 CONTINUE b280 SCRIPB(N)=BETA*S(1,2) b290 SCRIPB(N)=BETA*S(1,2) b300 SCRIPC(N)=S(2,1)*PC2*GAMMA b310 SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA b320 CO2(N)=ALPHA*PC2*GAMMA+SS(2)*GAMMA b320 CO2(N)=ALPHA*PC2 b340 12 CONTINUE cALL GFLTST(CTLC2,DV,LAMEC2,DIFC2,CC2,GPC2,EKC2) cALL GFLTST(CTLC2,DV,LAMEC2,DIFC2,CC2,GPC2,EKC2) cALL GFLTST(CTLC2,DV,LAMEC2,DIFC2,CC2,GPC2,EKC2) cALL GFLTST(CTLC2,DV,LAMEC0,DIFC0,CC0,GPC0,EKC0) b350 D0 50 N=1,24 b390 DENOM(N)=(SCRIPD(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) dGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DENOM(N) b410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DENOM(N) b420 IF (LOPGS.IT.4) G0 TO 50 c450 RETURN 	8220C	AC2=.0307 by The Program Satur
8240 DJPCO(N+24)=ACO*PriM.MM/#S1* GPCO(N)/AO2 8250 HASP(N)=HEMAT*MAMM/#S1*GPCO(N) 8260 CCO(N)=(1HEMAT*MAMM/#S1*GPCO(N)+HEMAT*GPCO(N+24) 6270 11 CONTINUE 8260 SCRIPA(H)=ALPHA+BETA*S(1,1) 8290 SCRIPB(N)=BETA*S(1,2) 8500 SCRIPC(N)=S(2,1)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA 8520 CO2(N)=ALPHA*PO2+SS(1)*BETA 8530 CC2(N)=SS(2)*GAMMA*PC2 8540 12 CONTINUE 8540 12 CONTINUE 8540 CALL GFLTST(CTLC2,DV,LAMBC2,DIFO2,CO2,GP02,EKO2) CALL GFLTST(CTLC2,DV,LAMBC2,DIFO2,CC2,GP02,EKO2) 8540 CALL GFLTST(CTLC2,DV,LAMBC2,DIFO2,CC2,GP02,EKO2) CALL GFLTST(CTLC2,DV,LAMBC2,DIFC0,CC0,GPC0,EKCO) 8540 CALL GFLTST(CTLC0,NV,LAMBC0,DIFC0,CC0,GPC0,EKCO) DG 0 8540 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 8430 DG 02(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DEAOM(N) 8440 DG 0CO(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DEAOM(N) 8440 DG 0CON)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 DG 0	8200	Michanik=1.0
 8250 HMSP(N)=HEMAT*MAMMM*S1*GPCO(N) 8250 CCO(N)=(1HEMAT)*ACC*GPCO(N)+HEMAT*GPCO(N+24) 6270 11 CONTINUE 8260 SCRIPB(N)=BETA*S(1,2) 8290 SCRIPE(N)=BETA*S(1,2) 8300 SCRIPC(N)=S(2,1)*PC2*GAMMA 8310 SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA 8320 CO2(N)=ALPHA*PO2+SS(1)*BETA 8350 CC2(N)=SS(2)*GAMMA*PC2 8340 12 CONTINUE 8350 CALL GFLTST(CTLO2,DV,LAMBO2,DIFO2,CO2,GPO2,EKO2) 8360 CALL GFLTST(CTLC2,DV,LAMBO2,DIFO2,CC2,GPC2,EKC2) 8370 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8370 CALL GFLTST(CTLC0,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8370 CALL GFLTST(CTLC0,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8370 CALL GFLTST(CTLC0,NV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8370 CALL GFLTST(CTLC0,NV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8370 CALL GFLTST(CTLC0,NV,LAMEC0,DIFC0,CC0,GPC0,EKC0) 9450 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 8440 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKO2(N)/DENOM(N) 8440 50 CONTINUE 8440 S0 DGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 CONTINUE 8440 S0 DGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 CONTINUE 8440 S0 DGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 CONTINUE 8450 RETURN 8440 S0 CONTINUE XDRIV(LOPGS) 8470 SUBROUTINE XDRIV(LOPGS) 8470 MEAL*8 LAMB02,LAMBC2,LAMBC0,LAMBX,LC02,LCC2,LCC0,LCX,MET02, 8500 DMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT02(4) 	8240	DGPCO(N+24)=ACO*MMMMMM*S1* GPCO(N)/AO2
 6260 CCO(N)=(1hEMAT)*ACG*GPCO(N)+HEMAT*GPCO(N+24) 6270 11 CCMTINUE 8260 SCRIPA(N)=ALPHA+BETA*S(1,1) 8290 SCRIPE(N)=BETA*S(1,2) 8500 SCRIPD(N)=S(2,1)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA 8520 CO2(N)=ALPHA*PO2+SS(1)*BETA 8530 CC2(N)=SS(2)*GAMMA*PC2 8540 12 CONTINUE 8550 CALL GFLTST(CTLC2,DV,LAMBO2,DIFO2,CO2,GPO2,EKO2) 8370 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8370 CALL GFLTST(CTLC0,DV,LAMEC0,DIFC0,CC0,GPC0,EKC0) 90 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 8450 DENOM(N)=(SCRIPA(N)*EKO2(N)-SCRIPC(N)*EKO2(N)/DEMOM(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKO2(N)/DEMOM(N) 8420 IF (LOPGs.LT.4) GO TO 50 9450 DGPC0(N)=(EKCO(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 CONTINUE 8440 S0 CONTINUE 8470 SUBROUTINE XDRIV(LOPGS) 8470 SUBROUTINE XDRIV(LOPGS) 8490 REAL*3 LANBO2,LAMBC2,LAMBC0,LAMBX,LC02,LCC2,LCC0,LCX,MET02, 8490 REAL*3 LANBO2(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT02(4) 	8250	HMSP(N)=HEMAT*MAMMAM*S1*GPCO(N)
<pre>4270 11 CONTINUE 8250 SCRIPA(H)=ALPHA+BETA*S(1,1) 8250 SCRIPB(N)=BETA*S(1,2) 8500 SCRIPC(N)=S(2,1)*PC2*GAMMA 8510 SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA 6320 CO2(N)=ALPHA*PO2+SS(1)*BETA 6550 CO2(N)=S(2)*GAMMA*PC2 8540 12 CONTINUE 8550 CALL GFLTST(CTLC2,DV,LAMBO2,DIFO2,CO2,GPO2,EKO2) 8350 CALL GFLTST(CTLC2,DV,LAMBC2,DIFO2,CC2,GPC2,EKC2) 8370 CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) 8370 CALL GFLTST(CTLC0,DV,LAMBC0,DIFC0,CC0,GPC0,EKC0) 8360 D0 50 N=1,24 8590 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPE(N)*EKC2(N))/DENOM(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DENOM(N) 8420 IF (LOPGS,LT.4) GO TO 50 0450 DGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 CONTINUE 8450 RETURN 8440 SUBROUTINE XDRIV(LOPGS) 6450 IMPLICIT REAL*8 (A-H,O-Z) 8490 REAL*3 LAMBC2,LAMBC2,LAMBC0,LAMBX,LCO2,LCC2,LCC0,LCX,METO2, 9500 DIMENSION GPO2(24),GPC2(24),GPC0(48),GPX(1,24),CTO2(4),CTC2(4)</pre>	o200	CCO(N)=(1hemat)*ACO*GPCO(N)+HEMAT*GPCO(N+24)
 SCRIPA(H)=ALPHA+BETA*S(1,1) SCRIPB(N)=BETA*S(1,2) SCRIPC(N)=S(2,1)*PC2*GAMMA SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA CO2(N)=ALPHA*PO2+SS(1)*BETA CO2(N)=ALPHA*PO2+SS(1)*BETA CO2(N)=S(2)*GAMMA*PC2 CALL GFLTST(CTLC2,DV,LAMBO2,DIFO2,CO2,GPO2,EKO2) CALL GFLTST(CTLC2,DV,LAMBC2,DIFO2,CC2,GPC2,EKC2) CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) CALL GFLTST(CTLC0,DV,LAMBC0,DIFC0,CC0,GPC0,EKC0) CALL GFLTST(CTLC0,DV,LAMBC0,DIFC0,CC0,GPC0,EKC0) CALL GFLTST(CTLC0,DV,LAMBC0,DIFC0,CC0,GPC0,EKC0) DO 50 N=1,24 DO 50 N=1,24 DO 50 N=1,24 DO 050 N=1,24 DO 050 N=1,24 CALD GFLTST(CTLC0,N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) DGP02(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DENOM(N) B410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DENOM(N) B420 IF (LOPGS.LT.4) GO TO 50 OGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) B440 50 CONTINUE CALD B440 SUEROUTINE XDRIV(LOPGS) CALTA LAMBO2,LAMBC2,LAMBC2,LAMBX,LCO2,LCC2,LCC0,LCX,METO2, DIMENSION GPO2(24),GPC2(24),GPC0(48),GPX(1,24),CTO2(4),CTC2(4) 	07.10	11 CONTINUE
 SCRIPB(N)=BETA*S(1,2) SCRIPC(N)=S(2,1)*PC2*GAMMA SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA CO2(N)=ALPHA*PO2+SS(1)*BETA CO2(N)=SS(2)*GAMMA*PC2 CO2(N)=SS(2)*GAMMA*PC2 CONTINUE CALL GFLTST(CTLC2,DV,LAMEO2,DIFO2,CO2,GPO2,EKO2) CALL GFLTST(CTLC2,DV,LAMEC2,DIFC2,CC2,GPC2,EKC2) CALL GFLTST(CTLC0,DV,LAMEC0,DIFC0,CC0,GPC0,EKC0) DC 50 N=1,24 DO DO 50 N=1,24 DO DO 50 N=1,24 DO DOPO2(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) DGPO2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N)/DENOM(N) 4410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N)/DENOM(N) B420 IF (LOPGS.LT.4) GO TO 50 DGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) B440 50 CONTINUE CMETURN SUBROUTINE XDRIV(LOPGS) IMPLICIT REAL*S (A-H,O-Z) REAL*5 LAMB02,LAMEC2,LAMEC0,LAMBX,LCO2,LCC2,LCC0,LCX,METU2, DIMENSION GPO2(24),GPC2(24),GPC0(48),GPX(1,24),CTO2(4),CTC2(4) 	8260	SCRIPA(N)=ALPHA+BETA*S(1,1)
 SCRIPC(N)=S(2,1)*PC2*GAMMA SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA CO2(N)=ALPHA*PO2+SS(1)*BETA CC2(N)=SS(2)*GAMMA*PC2 CC1L GFLTST(CTLO2,DV,LAMBO2,DIFO2,CO2,GPO2,EKO2) CALL GFLTST(CTLC2,DV,LAMBO2,DIFO2,CC2,GPC2,EKC2) CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) CALL GFLTST(CTLC0,DV,LAMBC0,DIFC0,CC0,GPC0,EKC0) CALL GFLTST(CTLC0,DV,LAMBC0,DIFC0,CC0,GPC0,EKC0) DO 50 N=1,24 O 00 50 N=1,24 O DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) DGPO2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DENOM(N) B410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DENOM(N) B420 IF (LOPGS.LT.4) GO TO 50 DGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) B440 50 CONTINUE CALT REAL*8 (A-H,O-Z) B470 SUBROUTINE XDRIV(LOPGS) IMPLICIT REAL*8 (A-H,O-Z) REAL*8 LAMBO2,LAMBC2,LAMEC0,LAMBX,LC02,LCC2,LCC0,LCX,METU2, DIMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT02(4) 	8290	SCRIPB(N) = BETA*S(1,2)
 SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA SCRIPD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA CO2(N)=ALPHA*PO2+SS(1)*BETA CO2(N)=SS(2)*GAMMA*PC2 CALL GFLTST(CTLO2,DV,LAMBO2,DIFO2,CO2,GPU2,EKO2) CALL GFLTST(CTLC2,DV,LAMBO2,DIFO2,CC2,GPC2,EKC2) CALL GFLTST(CTLC0,DV,LAMBC0,DIFC0,CC0,GPC0,EKC0) DC 50 N=1,24 DO DC 50 N=1,24 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) DGPO2(N)=(SCRIPA(N)*EKO2(N)-SCRIPE(N)/DEMOM(N) B410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKO2(N)/DEMOM(N) B420 IF (LOPGS.LT.4) GO TO 50 DGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) B440 50 CONTINUE B450 RETURN B460 END SUBROUTINE XDRIV(LOPGS) IMPLICIT REAL*S (A-H,O-Z) B490 REAL*S LAMBC2,LAMEC2,LAMBC0,LAMBX,LCO2,LCC2,LCC0,LCX,METU2, DIMENSION GPO2(24),GPC2(24),GPC0(48),GPX(1,24),CTO2(4),CTC2(4) 	8200	SCRIPC(N)=S(2,1)*PC2*GAMMA
 CO2(N)=ALPHA*PO2+SS(1)*BETA CC2(N)=SS(2)*GAMMA*PC2 CALL GFLTST(CTLO2,DV,LAMBO2,DIFO2,CO2,GPO2,EKO2) CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2) CALL GFLTST(CTLC0,DV,LAMBC0,DIFC0,CC0,GPC0,EKC0) CALL GFLTST(CTLC0,DV,LAMBC0,DIFC0,CC0,GPC0,EKC0) DO 50 N=1,24 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) DGPO2(N)=(SCRIPA(N)*EKC2(N)-SCRIPE(N)*EKC2(N)/DENOM(N) GGR1PA(N)*EKC2(N)-SCRIPE(N)*EKC2(N)/DENOM(N) DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N)/DENOM(N) CONTINUE CONTINUE CONTINUE CONTINUE CONTINUE CONTINUE CONTINUE SUBROUTINE XDRIV(LOPGS) MPLICIT REAL*& (A-H,O-Z) REAL*& LAMBO2,LAMBC2,LAMBC0,LAMBX,LCO2,LCC2,LCC0,LCX,METU2, DIMENSION GPO2(24),GPC2(24),GPC0(48),GPX(1,24),CTO2(4),CTC2(4) 	8310	SCR1PD(N)=S(2,2)*PC2*GAMMA+SS(2)*GAMMA
 CC2(N)=SS(2)*GAMMA*PC2 CONTINUE CALL GFLTST(CTLO2, DV, LAMBO2, DIFO2, CO2, GP02, EKO2) CALL GFLTST(CTLC2, DV, LAMBO2, DIFO2, CC2, GP02, EKO2) CALL GFLTST(CTLC0, DV, LAMBC0, DIFC0, CC0, GPC0, EKCO) CALL GFLTST(CTLC0, DV, LAMBC0, DIFC0, CC0, GPC0, EKCO) DO 50 N=1,24 DO 50 N=1,24 DO 50 N=1,24 DGP02(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) DGP02(N)=(SCRIPA(N)*EKO2(N)-SCRIPC(N)*EKO2(N)/DENOM(N) B410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKO2(N))/DENOM(N) B420 IF (LOPGS.LT.4) GO TO 50 DGPC0(N)=(EKCO(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) DGPC0(N)=(EKCO(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) CONTINUE B440 50 CONTINUE B450 RETURN B460 END B470 SUBROUTINE XDRIV(LOPGS) IMPLICIT REAL*8 (A-H,0-Z) REAL*8 LAMBO2, LAMBC2, LAMBC0, LAMBX, LCO2, LCC2, LCC0, LCX, METU2, DIMENSION GP02(24), GPC2(24), GPC0(48), GPX(1,24), CTO2(4), CTC2(4) 	6320	CO2(N) = ALPHA*PO2+SS(1)*BETA
 CONTINUE CALL GFLTST(CTLO2, DV, LAMBO2, DIFO2, CO2, GPO2, EKO2) CALL GFLTST(CTLC2, DV, LAMBC2, DIFC2, CC2, GPC2, EKC2) CALL GFLTST(CTLC0, DV, LAMBC0, DIFC0, CC0, GPC0, EKCO) CALL GFLTST(CTLC0, DV, LAMBC0, DIFC0, CC0, GPC0, EKCO) DG 50 N=1,24 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) DGPO2(N)=(SCRIPD(N)*EKO2(N)-SCRIPC(N)*EKC2(N))/DENOM(N) DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKO2(N))/DENOM(N) DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKO2(N))/DENOM(N) CONTINUE CONTINUE CONTINUE CONTINUE CONTINUE CONTINUE CONTINUE SUBROUTINE XDRIV(LOPGS) IMPLICIT REAL*& (A-H, 0-Z) REAL*& LAMBO2, LAMBC2, LAMBC0, LAMBX, LCO2, LCC2, LCC0, LCX, METU2, DIMENSION GPO2(24), GPC2(24), GPC0(48), GPX(1,24), CTO2(4), CTC2(4) 	8330	CC2(N) = SS(2) * GAMMA* PC2
8350 CALL GFLTST(CTL02,DV,LAMB02,D1F02,C02,GP02,EK02) 8300 CALL GFLTST(CTL02,DV,LAMB02,D1F02,C02,GP02,EK02) 8370 CALL GFLTST(CTL00,DV,LAMB02,D1F00,CC0,GP00,EK00) 8300 D0 50 N=1,24 8590 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 8400 DGP02(N)=(SCRIPA(N)*EK02(N)-SCRIPD(N)*EK02(N))/DENOM(N) 8410 DGP02(N)=(SCRIPA(N)*EK02(N)-SCRIPC(N)*EK02(N))/DENOM(N) 8420 1F (L0PGS.LT.4) GO TO 50 0430 DGP00(N)=(EK00(N)/(AC0*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 50 CONTINUE 8440 50 6450 RETURN 8440 SUBROUTINE XDRIV(LOPGS) 8470 SUBROUTINE XDRIV(LOPGS) 8470 SUBROUTINE XDRIV(LOPGS) 8480 IMPLICIT REAL*8 (A-H,O-Z) 8490 REAL*8 LAMB02,LAMBC2,LAMBC0,LAMBX,LC02,LCC2,LCC2,LCC0,LCX,MET02, 8500 DIMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4)	8340	12 CONTINUE
8350 CALL GFLTST(CTLC2, DV, LAMBC2, DIFC2, CC2, GPC2, EKC2) 8370 CALL GFLTST(CTLC2, DV, LAMBC0, DIFC0, CC0, GPC0, EKC0) 8350 D0 50 N=1, 24 8590 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 8400 DGP02(N)=(SCRIPA(N)*EK02(N)-SCRIPE(N)*EK02(N))/DENOM(N) 8410 DGPC2(N)=(SCRIPA(N)*EK02(N)-SCRIPC(N)*EK02(N))/DENOM(N) 8420 IF (LOPGS.LT.4) GO TO 50 8430 DGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 50 CONTINUE 8440 50 8440 50 8440 50 8440 50 8440 50 8450 RETURN 8440 50 8450 RETURN 8460 END 8470 SUBROUTINE XDRIV(LOPGS) 8480 IMPLICIT REAL*8 (A-H, 0-Z) 8490 REAL*8 LAMB02, LAMBC2, LAMBC0, LAMBX, LC02, LCC2, LCC2, LCC0, LCX, MET02, 5500 DIMENSION GP02(24), GPC2(24), GPC0(48), GPX(1, 24), CT02(4), CT02(4), CT02(4)	8350	CALL GFLTST(CTLO2, DV, LAMBO2, D1F02, CO2, GP02, EKO2)
8370 CALL GFLTST(CTLCO, DV, LAMBCO, DIFCO, CCO, GPCO, EKCO) 8300 D0 50 N=1,24 8390 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 8400 DGPO2(N)=(SCRIPD(N)*EKO2(N)-SCRIPC(N)*EKO2(N))/DENOM(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKO2(N))/DENOM(N) 8420 IF (LOPGS.LT.4) GO TO 50 8440 50 50 CONTINUE 8440 50 50 CONTINUE 8440 S0 50 CONTINUE 8440 S0 50 CONTINUE 8440 S0 50 CONTINUE 8440 S0 50 CONTINUE 8450 RETURN 8460 END 8470 SUBROUTINE XDRIV(LOPGS) 8480 IMPLICIT REAL*8 (A-H, 0-Z) 8490 REAL*8 LAMB02, LAMBC2, LAMBC0, LAMBX, LC02, LCC2, LCC2, LCC0, LCX, MET02, 0 8500 DIMENSION GP02(24), GPC2(24), GPC0(48), GPX(1,24), CT02(4),	0020	CALL GFLTST(CTLC2, DV, LAMBC2, DIFC2, CC2, GPC2, EKC2)
0500 D0 50 N=1,24 0590 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 0400 DGP02(N)=(SCRIPD(N)*EK02(N)-SCRIPE(N)*EK02(N))/DENOM(N) 05410 DGP02(N)=(SCRIPA(N)*EK02(N)-SCRIPC(N)*EK02(N))/DENOM(N) 05420 1F (LOPGS.LT.4) GO TO 50 05450 DGP00(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 05450 CONTINUE 05450 RETURN 05450 RETURN 05450 SUBROUTINE XDRIV(LOPGS) 05450 IMPLICIT REAL*8 (A-H,0-Z) 05450 REAL*8 LAMB02,LAMBC2,LAMBC0,LAMBX,LC02,LCC2,LCC0,LCX,MET02, 0500 DIMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT02(4),CT02(4),CT02(4)	8370	CALL GFLTST(CTLCO, DV, LAMBCO, DIFCO, CCO, GPCO, EKCO)
0590 DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPE(N))*V(N) 0400 DGP02(N)=(SCRIPD(N)*EK02(N)-SCRIPE(N)*EK02(N))/DENOM(N) 05410 DGPC2(N)=(SCRIPA(N)*EK02(N)-SCRIPC(N)*EK02(N))/DENOM(N) 05420 IF (LOPGS.LT.4) GO TO 50 05430 DGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 05440 50 050 CONTINUE 05450 RETURN 05450 END 05450 SUBROUTINE XDRIV(LOPGS) 05450 IMPLICIT REAL*8 (A-H,O-Z) 05450 REAL*8 LAMB02,LAMBC2,LAMBC0,LAMBX,LC02,LCC2,LCC0,LCX,MET02, 0500 DIMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT02(4)	0300	DU = 50 N = 1,24
8455 DGP02(N)=(SCRIPD(N)*EK02(N)-SCRIPD(N)*EKC2(N))/DENOM(N) 8410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DENOM(N) 8420 1F (LOPGS.LT.4) GO TO 50 0430 DGPC0(N)=(EKC0(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 CONTINUE 8450 RETURN 8460 END 8470 SUBROUTINE XDRIV(LOPGS) 8480 IMPLICIT REAL*8 (A-H,O-Z) 8490 REAL*8 LAMB02,LAMEC2,LAMEC0,LAMBX,LC02,LCC2,LCC2,LCC0,LCX,MET02, 8500 DIMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT02	8590	DENOM(N) = (SCRIPA(N) * SCRIPD(N) - SCRIPC(N) * SCRIPE(N)) * V(N)
0410 DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKC2(N))/DENOM(N) 8420 1F (LOPGS.LT.4) GO TO 50 0450 DGPCO(N)=(EKCO(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 50 CONTINUE 8440 50 8450 RETURN 0450 END 8470 SUBROUTINE XDRIV(LOPGS) 8480 IMPLICIT REAL*8 (A-H,O-Z) 8490 REAL*8 LAMBO2,LAMBC2,LAMBC0,LAMBX,LC02,LCC2,LCC0,LCX,MET02, 0500 DIMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT02(4)	8400	DGPO2(N) = (SCRIPD(N) * EKO2(N) - SCRIPB(N) * EKC2(N)) / DENOM(N)
0420 IF (LOPGS.LI.4) GO TO 50 0450 DGPCO(N)=(EKCO(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT) 8440 50 50 CONTINUE 8450 RETURN 0450 END 8470 SUBROUTINE XDRIV(LOPGS) 6480 IMPLICIT REAL*8 (A-H,O-Z) 8490 REAL*8 LAMB02,LAMBC2,LAMBC0,LAMBX,LC02,LCC2,LCC0,LCX,METO2, 8500 DIMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT02(4)	0410	DGPC2(N) = (SCRIPA(N) * EKC2(N) - SCRIPC(N) * EKC2(N)) / DENOM(N)
8430 DGPCO(N)=(ERCO(N)/(ACO*V(N)))-HMSP(N))/(1.+HEMA1) 8440 50 8450 RETURN 8460 END 8470 SUBROUTINE XDRIV(LOPGS) 8480 IMPLICIT REAL*8 (A-H,O-Z) 8490 REAL*8 LAMBO2,LAMBC2,LAMBC0,LAMBX,LC02,LCC2,LCC0,LCX,METU2, 8500 DIMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT02(4)	0420	IF (LUPGS.LT.4) GU TU 5U $P_{2}P_{2}P_{3}P_{4}$
 b) CONTINUE c) Continue <lic) continue<="" li=""> c) Continue <lic) continue<="" li=""> c)</lic)></lic)>	0450	DGFCU(N)=(EKCU(N)/(ACU*V(N))-HMSP(N))/(1.+HEMA1)
0450RETURN0450END8470SUBROUTINE XDRIV(LOPGS)0480IMPLICIT REAL*8 (A-H,O-Z)8490REAL*8 LAMBO2,LAMBC2,LAMBC0,LAMBX,LC02,LCC2,LCC0,LCX,MET02,8500DIMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT0	0440	DU CUNTINUE
 8470 END 8470 SUBROUTINE XDRIV(LOPGS) 6480 IMPLICIT REAL*8 (A-H,O-Z) 8490 REAL*8 LAMBO2,LAMBC2,LAMBCO,LAMBX,LCO2,LCC2,LCC0,LCX,MET02, 8500 DIMENSION GPO2(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4)	0450	
0470SUBRUUTINE XDRIV(LUPGS)0480IMPLICIT REAL*8 (A-H,O-Z)8490REAL*8 LAMBO2,LAMBC2,LAMBCO,LAMBX,LCO2,LCC2,LCC0,LCX,METU2,8500DIMENSION GPO2(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),CT	040U Ω1170	END CHERODAULNE VIDIN(LODCC)
0400IMPLICIT REAL*0 (A=n,0-2)8490REAL*5LAMB02,LAMBC2,LAMBC0,LAMBX,LC02,LCC2,LCC0,LCX,MET02,8500DIMENSION GP02(24),GPC2(24),GPC0(48),GPX(1,24),CT02(4),	0470	
6490 REAL"S LAMBC2, LAMBC2, LAMBC0, LAMBX, LCO2, LCC2, LCC0, LCX, METO2, 8500 DIMENSION GPO2(24), GPC2(24), GPC0(48), GPX(1,24), CTO2(4), CTC2(4), CTC2(4)	0400	THEFTOTI REFERENCE (N-T)-C)
$0000 \qquad DIMENSION Groz(24), Groz(24), Groz(40), Gra(1, 24), Groz(4), Groz(24), Groz(2$	8600	TEAL - LANDUZ, LANDUZ, LANDUU, LANDX, LUUZ, LUUZ, LUU, LUX, METUZ, METUZ
	0000	DIFENSION OFO2(24), OFO2(24), OFO0(40), OFX(1,24), OFO2(4), OFO2

8510 1CTCO(4).CTX(1.4).DIFO2(10).DIFC2(10).DIFCO(10).DIFX(10).CPH(24). 6520 2LCO2(6).LCC2(6).LCCO(6).LCX(1.6).CSATO2(24).CSATC2(24).DGPO2(24).6530 3DGPC2(24), DGPCO(45), DGPX(1,24), AX(1)DIMENSION CILO2(18), CILC2(18), CILCO(18), CILX(1,13) 8540 8550 DIMENSION VFLIN(18), VFLOW(24), DV(24), V(24), C(24) DIMENSION DCTO2(4), DCTC2(4), DCTCO(4), DCTX(1,4), METO2(4), METC2(4) 8560 8570 DIMENSION EKX1(1.24) o580 COMMON /VFLT/VFLOW.VFLIN 8590 COMMON /BCOVER/CTLO2.CTLC2.CTLC0.CTLX 8600 COMMON /EC/LAMBG2.LAMEC2.LAMEC0.LAMEX.AO2.AC2.AC0.AX.HEMA1.DIF02. 8610 1D1FC2.D1FC0.D1FX.CPH.CSAT02.CSATC2.MET02.METC2 COMMON /CMN1/V, VP2, VP3, VP4, VP12, VP13, VP14, GP02, CT02, GPC2, CTC2 0020 8630 COMMON /CMN1T/GPCO.CTCO.GPX.CTX 0040 COMMON /CMN2/DV.DVP2.DVP3.DVP4.DVP12.DVP13.DVP14.DGP02.DCT02. 8650 1DGPC2.DCTC2 COMMON /CMN2T/DGPCO.DCTCO.DGPX.DCTX 0003 8670 DO 1 N=1.24 0606 C(N) = AX(1) * GPX(1.N)6690 CALL GFLTST(CTLX, DV, LAMBX, DIFX, C, GPX, EKX1) 8700 DGPX(1,N) = EKX1(1,N)/(AX(1)*V(N))8710 **1** CONTINUE 0/20 RETURA 8730 END 8740 SUBROUTINE SATUR (PO2.PCO2.HEMAT.SS.S.S1.PH) 8150 IMPLICIT REAL*3 (A-H.O-Z) 8700 DIMENSION SS(2), S(2,2)0770 Y=0.0 3700 1PC02=1.00 8793 DO 2 N=1.2 8300 IF (PCU2.GE.0.001) GO TO 1 0010 TPCG2=PCO2 8520 PC02=0.001 <u>აძ</u>30 1 PH=7.59+Y-0.2741*ALOG(PC02/20.0) Z=EXP((PH-7.4)*1.812)dð40 8850 C=5.727E-3*Z+4.273E-3 ნძსე V=C*P02 8910 1F (V.GT.1.0E-11) GO TO 11 8330 V = 0.08040 11 CONTINUE U=0.925*V+2.8*V*V+30.0*V*V*V 0900 S1=0/(1+0)6910 8920 2 Y=0.045*(1.0-S1) 8930 SS(1) = S18940 A = (.925 + 5.6 * V + 90 * V * V)8950 AA=A*PO2*(.010377)*Z A2=AA*.045 8400 S11=A*C/((1.+U)*(1.+U)+A2)8970 6960 S(1,1)=S118990 S12=-AA*.2741/((1.+U)*(1.+U)+AA*.045)/PCO2 9000 S(1,2)=S12

131

9010C	
9020	DOX=0.59+0.2913*(7.4-PH)
9030	DR=0.664+0.2275*(7.4-PH)
9040	DDD=DOX+(DR-DOX)*(1.0-S1)
9050	YY=EXP(2.3961*PH-14.7162)
9000	CP=0.0307 *(1.0+YY)
9070	SS(2)=(1.0-HEMAT)*CP+HEMAT*DDD*CP
9050C	
9090	DPH02=045*S11 ·
9100	DDR=2275*DPh02
9110	DDOX=2913*DPh02
9120	DDDD=DDUX+(DDR-DDOX)*(1,-S1)+(DR-DOX)*(-S11)
9150	DYYG2=2.396*DPHO2*YY
y140	DCP = .0307*(+DYY02)
915ü	S(2,1)=(1,-heMAT)*DCP+HEMAT*DDDD*CP+HEMAT*DDDD*DCP
9100	DYC2=045*S12
9170	DPHC2=DYC22741/PCO2
9130	DDRC2=2275*DPHC2
9190	DDOXC2=2913*DPHC2
9200	DDDDC2=DDOXC2+(DDRC2-DDOXC2)*(1,-S1)-(DR-DOX)*S12
9210	DYYC2=2.3961*YY*DPHC2
9220	DCPC2 = .030 (*0) YC2
9230	S(2,2)=(1-HEMAT)*DCPC2+HEMAT*DDDDC2*CP+HEMAT*DDD*DCPC2
9240	IF (TPC02.GE.0.001) GU TO 3
9250	PCO2=TPCO2
9260	3 CONTINUE
9270	RETURN
9200	END
9290	SUEROUTINE GELTST(CT.VP.LAMBDA.D.C.GP.EK)
4300	$IMP[JC1T] REAL*\sigma (A-H, O-Z)$
9310	REAL#d LAUBDA
4320	DIMENSION VFLOW(24).VFLIN(18).C(24).CCFL(24).CCFL(18).VP(24)
9350	$1 \text{ EK}(24) \cdot D(10) \cdot CT(13) \cdot GP(24)$
4340	COMMON /VFLT/VFLOW, VFLTN
9350	CÚMEON /CELT/CCEL.CCELIN
9300	CALL FUEST $(1,2,1,1,c)$
9370	$CCFL(2) = C(2) \times VFLOW(2)$
9560	CCF((X) = C(X) * VFLOW(X)
4340	CCFL(4)=0.0
4400	$DO_{11} N=5.11$
9410	1=N
9420	CALL FLTEST(1+24, 1, 4, 1+24, C)
9430	11 CCEL(4) = CCEL(4) + CCELTN(N)
9440	$D_{\rm U} = 12 \text{N} = 5.10$
9450	1=N
GHDO	$12 \text{ CAUL FLTEST}(1, 1), T \in C)$
9470	CALL FLTEST(11, 12, 11, 11, C)
9480	CCPL(12)=C(12)*VFLOW(12)
9490	CCFL(13)=C(13)*VFLOW(13)
9500	CCFL(14)=0.0

1.5

•

ready

.

```
9510
          DO 13 N=15.10
9520
          l = N
9550
          CALL FLTEST(1+24.1.14.1+24.C)
9540
       13 CCFL(14)=CCFL(14)+CCFLIN(N)
          DO 14 N=15.20
9550
          I = N
9560
9570
       14 CALL FLTEST(I.I+4.I.I.C)
9500
          CALL FLTEST(21.1.21.21.C)
9590
          CALLFLTEST(22,1,22,22,C)
9600
          CALL FLTEST(23,24,23,23,C)
          CALL FLIEST(24,2,24,24,C)
9ó10
6020
          EK(1)
                  =CCFL(21)+CCFL(22)-CCFL(1)-C(1)*VP(1)
9530
          EK(2)
                  =CCFL(1)+CCFL(24)-CCFL(2)-C(2)*VP(2)
9640
          EK(3)
                  = CCFL(2) - CCFL(3) - C(3) * VP(3)
9050
          EK(4) = CCFL(3) - CCFL(4) - C(4) * VP(4)
9050
          DU 100 N=5.10
9670
      100 EK(N)=CCFLIN(N)-CCFL(N)-C(N)*VP(N)+D(N-4)*(CT(N)-LAMEDA*GP(N))
9080
          EK(11)=CCFL(5)+CCFL(6)+CCFL(7)+CCFL(8)+CCFL(9)+CCFL(10)+CCFLIN(11)
5090
          1 - CCrL(11) - C(11) \times VP(11)
          EK(12)=CCFL(11)-CCFL(12)-C(12)*VP(12)
9700
9/10
          EK(15)=CCFL(12)=CCFL(15)=C(15)*VP(15)
9720
          EK(14) = CCFL(15) = CCFL(14) = VP(14) * C(14)
9730
          DO 112 N=15.10
      112 EK(N)=CCFLIN(N)-CCFL(N)-VP(N)*C(N)+D(N-d)*(CT(N)-LAMEDA*GP(N))
9740
          DO 115 N=19,25
9750
      113 EK(N)=CCFL(N-4)-CCFL(N)-VP(N)*C(N)
9760
          EK(24) = CCFL(23) + CCFL(20) - CCFL(24) - VP(24) + C(24)
9710
9700
          DO 131 N=1.24
9790
          IF (DABS(EK(N)).GT.1.0E-13) GO TO 130
9000
          EK(N)=0.0
      130 CONTINUE
9810
9820
      151 CONTINUE
9330
          RETURN
9840
          END
y350
          SUBROUTINE FLTEST(11,1N,1P,1V,C)
           IMPLICIT REAL*3 (A-H,O-Z)
9860
9570
           DIMENSION FL(42), C(24), CC(42)
9000
           COMMON /VELT/FL
9890
           COMMON /CFLT/CC
9900
           IF(FL(11)) 1.1.2
9910
        1 CC(1V) = C(1N) * FL(11)
9920
           RETURN
9930
        2 CC(IV) = C(1P)*FL(11)
9940
           RETURN
9950
           END
```

122

```
ready
```

¥

```
SUBROUTINE DERIVS( OPT1, MTEST, PSAVE )
        IMPLICIT REAL*o( A-H, O-Z )
        REAL#8 N. NW
        REAL*4 G1. G2. G3. G4
        INTEGER OPT1. FEOPT
        DIMENSION X(b), Z(b), ALPHA(6), BETA(6), PS(b), DDCW(12).
            DDNW(12). PSAVE(12)
       1
        COMMON / LLK1/ N(12,3), VTX, VALV(6), PALV(6), SW(12), VSPR
90
        COMMON /BLK2/ DN(12,3), DVTX, DVALV(6), DPALV(6), DSW(12), DVSPR
100
         COMMON /BLX10/ PGAV. DPGAV. PGAVB. DPGAVE
         COMMON /ELK11/ PTX, DPTX, PD, DPD, G1, G2, G3, G4, FEGPT
110
120
         COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
             F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PBS,
150
        1
140
        2
             DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
150
        3
             PG(6), DPG(6), PGAE, DPGAB, PPL, DPS(6), DPS2,
160
        4
             S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
170
             SQT(12), DP(12), HDT
        5
180
         DATA TTTS / -1.0 DO /
190C
200
         TTT = T
210
         IF ( OPT1. EQ. 10. AND. TTT. EQ. TTTS ) GO TO 10
220C
250
       5 \text{ FSPR} = F(2)
240
         IF ( MTEST. EQ. 1 ) CALL FORCE ( TTT. FSPR. DT )
250
         IF ( MTEST. NE. 1 . AND . FBOPT. GE. 9 ) CALL FORCE ( ITT, FSPR,
260
             DT)
        8
2700
200
         GO TO 15
290
      10 CONTINUE
300
         DPT\lambda = 0.
310
         DPD = 0.
320
      15 CONTINUE
         TITS = TTT
330
340
         PELIX = PELAST (0, 7, VTX)
         DPELTX = PELAST (-1, \gamma, VTX)
350
300
         PELDA = PELAST (0, 0, VD)
         DPELDA = PELAST (-1, 8, VD)
370
300
         D0 20 I = 1, o
390
         100MY = 1
400
         PELALV(I) = PELAST (0, IDUMY, VALV(1))
410
      20 DPELAL(1) = PELAST (-1, 1) DUMY, VALV(I) )
4200
         IF ( OPT1. NE. O ) CALL PGRAV ( PBS, PGAB, PG, DPES, DPGAB, DPG )
430
440C
450
         DPS2 = DPTX - DPD + DPGAB + DPGAVB - DPGAV
460
         DUM = DPTX + DPBS - DPGAV
470
         DUM2 = PTX + PBS - PGAV
480
         DO 30 i = 1, 6
```

490

500

DPS(1) = DUM + DPG(1)

30 PS(1) = DUM2 + PG(1)

×

10

20

30

40

50

óΟ

70

80

```
510C
520
          CALL AIRWAY ( DN, DDCW )
5300
540
          DO 50 J = 2.0
          DCWDP(J) = G1 * DEXP(G2 * (P(J) - 760.))
550
500
          DP(J) = DNT(J) * G / (V(J) * (1. - G * DCWDP(J)))
570
          DP(J) = (P(J) - PSAVE(J)) / DT
580
          DCW(J) = DCWDP(J) * DP(J)
          D_{NW}(J) = V(J) = DCW(J)
590
      50 DSW(J) = DNW(J) - DDCW(J)
600
610
          DNT(1) = 0.
620
          IF ( F(2) ) 60. 70. 80
530
      60 DO 65 I= 1. NG
          DN(1,I) = -F(2) * (C(2,I) + HDT * DC(2,I)) + S(1)
640
650
      55 \text{ DNT}(1) = \text{DNT}(1) + \text{DN}(1,1)
660
          DU_{M}Y = -F(2) * (CW(2) + HDT * DCW(2))
570
          GO TO 90
000
      70 DO 75 1= 1. NG
640
      75 DN(1.1) = 0.
700
          DUMY = 0.
710
          GO TO 90
720
      00 DO 85 1= 1,NG
730
          DN(1,I) = -C(1,I) * F(2) + S(I)
740
      85 \text{ DNT}(1) = \text{DNT}(1) + \text{DN}(1,1)
750
          DU_{CY} = -F(2) * CW(2)
      90 DVSPR = DNT(1) / ( P(1) / GS - CW(1) )
760
770
          DNW(1) = DVSPR * CW(1)
700
          DSW(1) = DNW(1) - DUAY
740
          DO 100 I= 1, NG
     100 DC(1,1) = (DN(1,1) - DVSPR * C(1,1)) / V(1)
000
010C
950
          CALL MOLES (F(7), DnT(7), NG, HDT, C, DC, DN, DDNW(7), CW, DDCW, 3, ()
830
          CALL MOLES(F(\delta), DAT(\delta), NG, HDT, C, DC, DN, DDNW(\delta), CW, DDCW, 4, \hat{\beta})
640
          CALL MOLES( F(9), DNT(9), NG, HDT, C, DC, DN, DDNW(9), CW, DDCW, 5, 9)
850
          CALL MOLES( F(10), DNT(10), NG, HDI, C, DC, DN, DDNh(10), CW, DDCW, 6, 10)
043
          CALL MOLES( F(11), DNT(11), NG, HDT, C, DC, DN, DDNN(11), CW, DDCW, 5, 11)
870
          CALL MOLES( F(12), DNT(12), NG, HDT, C, DC, DN, DDNW(12), CW, DDCW, 6, 12)
850C
690
          DU 250 J = 7, 12
900
          CT(J) = 0.
910
          DÚ 250 1= 1, NG
920
     250 \text{ CT}(J) = \text{CT}(J) + \text{C}(J,I)
930
          ALPHAT = 0.
          betat = 0.
940
          DO 300 J = 7, 12
950
960
          K = J - \dot{o}
          DCwDP(J) = G1 * DEXP(G2 * (P(J) - 760.))
970
980
          X(K) = G * DCwDP(J)
          Z(K) = 1. - X(K)
990
           DEY = P(J) + DPELAL(K) * VALV(K) * Z(K) - G * CW(J)
1000
```

```
1010
          ALPHA(K) = (DNT(J) * G - DPS(K) * V(J) * Z(K)) / DMY
1020
          bela(K) = DPELTX * VALV(K) * Z(K) / DAY
          ALPHAT = ALPHAT + ALPHA(K)
1050
1040
          BETAT = BETAT + BETA(K)
1050
      500 CONTINUE
1050
          DVTX = (ALPHAT * DPELDA + DPS2) / (DPELTX + DPELDA * (1.-
1070
                 BETAT ) )
         1
          DO 400 J= 7, 12
1000
1090
          K = J - o
1100
          DVALV(K) = ALPHA(K) + BETA(K) * DVTX
          DP(J) = (DNI(J)*G - (P(J) - G*CW(J))*DVALV(K)) / (V(J)*Z(K))
1110
1120
          DPALV(K) = DP(J)
          DSW(J) = DCWDP(J) * DP(J) * V(J) + CW(J) * DVALV(K) - DUNK(J)
1130
1140
      400 CONTINUE
1150 1001 FURMAT( 3X,F5.3, 3X, F12.8, 3X, F12.8,
              3X, F12.7, 2(3X, E10.4)
1100
         1
1170 1000 RETURN
1160
          E.ND
1190
          SUBROUTINE PARMS2( VA )
1200
          REAL*O N. NW
          REAL*4 PCALC, CTCALC, VLDIF, G1, G2, G3, G4
1210
1220
          IMPLICII REAL*3( A-H, O-Z )
1230
          INTEGER FBOPT
1240
          DIMENSION DELP(6), X(5), B(5), KNT2(5), FACT(5)
1250
          COMMON /ELK1/ N(12,3), VTX, VALV(6), PALV(6), SW(12), VSPR
          COMMON /ELK2/ DN(12,3), DVTX, DVALV(6), DPALV(6), DSw(12), DVSPR
1200
          CUMMON / ELK6 / A(5,5), FO(6), FI(6), SQO(6), SQI(6)
1270
1280
          COMMINIAN /ELK // PCALC(12), CTCALC(12), VLDIE
          COMMON /ELK10/ PGAV, DPGAV, PGAVE, DPGAVE
1290
          CUMMON /ELK11/ PTX, DPTX, PD, DPD, G1, G2, G3, G4, FEOPT
1300
          CUMMUN C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12).
1510
1320
         1
              F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PBS,
              UPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELIX, DPELIX,
1550
         2
              PG(0), DPG(0), PGAE, DPGAB, PPL, DPS(0), DPS2,
1540
         3
              S(4), T, DT, TN(12), 10TN(12), V(12), VD, VLUNG,
1550
         4
1300
         5
              SQT(12), DP(12), HDT
          DATA EPS / 1.0D-5 /
1570
1300
          VLUNG = VA
1390
          V(1) = VSPR
1400
          DO 25
                 J=7,12
1410
          K = J - 6
          P(J) = PALV(K)
1420
          V(J) = VALV(K)
1430
1440
       25 VLUNG = VLUNG+V(J)
1450
          TN(1) = 0.
1460
          DO 75 1= 1,NG
          C(1,1) = N(1,1) / V(1)
1470
1400
       75 \text{ TN}(1) = \text{TN}(1) + \text{N}(1,1)
          CT(1) = P(1) / GS - CW(1)
1490
1500
          DO 90 J=7,12
```

135

```
ready
```

```
1510
          C_{W}(J) = G_{X} * DEXP(G_{2} * (P(J) - 760.))
          DCWDP(J) = G1 * DEXP(G2 * (P(J) - 760.))
1520
          CT(J) = P(J) / G - CW(J)
1530
          TN(J) = 0.
1540
          DO 65 1= 1,NG
1550
          C(J,1) = N(J,I)/V(J)
1560
       85 \text{ TN}(J) = \text{TN}(J) + \text{N}(J,I)
1570
          CT(J) = P(J) / G - CW(J)
1580
1590
       90 CONTINUE
          DG 100 J= 2,6
1600
          C_{W}(J) = G_3 * DEXP(G_2 * (P(J) - 760.))
1610
1620
          DCWDP(J) = G1 * DEXP(G2 * (P(J) - 760.))
16.0
          CT(J) = P(J) / G - Cw(J)
1640
          TN(J) = 0.
          DO 100 1= 1,NG
1650
1650
          TN(J) = TN(J) + N(J,I)
      100 C(J,1) = N(J,1) / V(J)
1070
          PPL = PTX + PBS - PELTX - PGAV
1000
1690
          VD = VLONG - VIX - VLDIF
          DO 130 J= 1,5
1700
1710
          FACT(J) = .6
1720
          KNT2(J) = 0
1730
          DO 150 1= 1,5
      130 A(I,J) = 0.
1740
          DU 210 KNT = 1, 50
1750
1760
          IF ( KNT. EQ. 1 ) GO TO 155
17/0
          CALL FLOW ( 1, 2 )
          CALL FLOW (2, 3)
1700
          CALL FLOW (2,4)
1790
1000
          CALL FLOW (3,5)
1810
          CALL FLOW (4, 6)
      155 CONTINUE
1020
1050
          CALL FLOW (3,7)
          CALL FLUW (4, 8)
1040
          CALL FLOW ( 5, 9 )
1050
          CALL FLOW ( 6, 10 )
1000
          CALL FLOW ( 5, 11 )
1070
1000
          CALL FLOW (v, 12)
          DO 100 J= 2,6
1690
1900
          FO(J) = 0.
1910
          F1(J) = 0.
1920
          SQO(J) = 0.
1930
          SQI(J) = 0.
      160 \text{ CT}(J) = P(J) / G - CW(J)
1940
          CALL SUB1 ( 1, 2 )
1950
          CALL SUB1 (2, 3)
1900
          CALL SUB1 (2, 4)
1970
1900
          CALL SUB1 ( , 5 )
          CALL SUB1 ( 4, 6 )
1990
2000
          CALL SUE1 ( 3, 7 )
```

×

```
2010
          CALL SUE1 ( 4, 8 )
2020
          CALL SUB1 (5, 9)
          CALL SUB1 ( 6, 10 )
2030
2040
          CALL SUB1 ( 5, 11 )
          CALL SUB1 ( 6, 12 )
2050
2060
          SE = 0.
2070
          DO 200 J= 2.6
          K = J - 1
2080
          KNT2(K) = KNT2(K) + 1
2090
2100
          A(K,K) = -(SQI(J) + (G4 - DCWDP(J)) * FO(J) + CT(J) * SQO(J))
          IF( DABS( A(K,K) ). LT. .1D-12 ) A(K,K) = -.1D-12
2110
          B(K) = -FI(J) + CT(J) * FO(J)
2120
2130
      200 \text{ SB} = \text{SB} + \text{DABS}(B(K))
          CALL BAPM ( A, X, B )
2140
2150
          SDP = 0.
2160
          DO 203 K= 1,5
      203 \text{ SDP} = \text{SDP} + \text{Abs}(X(K))
2170
2180
          IF ( KNT. GT. 25) WRITE(6,1234) T,SB,(P(J),J=2,6)
2190 1234 FORMAT ( 1X, F5.3, 1X, 1PE9.2, 5(1X, F9.5))
          IF ( SB. LT. EPS ) GO TO 215
2200
2210
          DU 210 J= 2,6
          K = J - 1
2220
          IF ( KNT2(K). EQ. 1 ) GO TO 206
2230
2240
          1F ( DELP(J). EQ. 0.. OR. X(K). EQ. 0. ) GO TO 206
2250
          1F ( DELP(J) / X(K) ) 205, 205, 204
      204 IF ( KNT2(K). LT. 3 ) GO TO 206
2260
2270
          FACT(K) = 1.0
          GO TO 206
2260
2290
      205 \text{ FACT}(K) = 0.6
          KNT2(K) = 0.
2300
2310
      206 X(K) = FACT(K) * X(K)
2320
          DELP(J) = X(K)
           P(J) = P(J) + DELP(J)
2330
           CW(J) = G3 * DEXP(G2 * (P(J)-760.))
2340
2350
      210 \text{ DCWDP}(J) = G1 * \text{DEXP}(G2 * (P(J) - 760.))
2360
           STOP 1234
2370
      215 CONTINUE
2380C
            WRITE(6,1111) KNT, SB
2390 1111 FORMAT(1X,'KNT, SB = ', I5, E14.3 )
2400
           DO 220 J = 1, 12
2410
           NW(J) = CW(J) * V(J)
2420
      220 \text{ TOTN}(J) = \text{TN}(J) + \text{NW}(J)
2430
           DO 230 J= 2,6
2440
           Cw(J) = G3 * DEXP(G2 * (P(J)-760.))
2450
           CTCALC(J) = CT(J)
2460
      230 PCALC(J) = TOTN(J) * G / V(J)
2470
          DO 235 J= 1,12
2400
      235 CT(J) = TOTN(J) / V(J)
2490
           RETURN
2500
           END
```

X
```
SUBROUTINE PGRAV ( PBS, PGAB, PG, DPBS, DPGAB, DPG )
 10
 20
         IMPLICIT REAL#8( A-H. O-Z )
 30
         DIMENSION PG(6), DPG(6)
 40
         COMMON /BLK8/ PHI, GF, GR, GT, GD, HP, PIE, ANG1, ANG2, PHI1, PHI2
          COMMON /BLK10/ PGAV, DPGAV, PGAVB, DPGAVB
 50
 60
         GR = 1. + GF
 70
          PHI = PHI2
 80
          COSPHI = DCOS(PHI)
 90
          SINPHI = DSIN(PHI)
           GD = GR * COSPHI
 100
 110
          GT = GR * DCOS(PHI - .2618)
 120
           GCON = GR \neq 0.73557 / 2.
           H = 8.67 * COSPH1
 130
 140
           PBS = 760.
           PGAB = GCON * 8.* DCOS( 2.*PHI ) + 2. * GR * S1NPHI
 150
 100
           PG(1) = GCON * (13.0 - H)
 170
           PG(2) = GCON * (13.0 - H)
           PG(3) = GCON * 13.0
 160
 190
           PG(4) = GCON * 13.0
 200
           PG(5) = GCON \neq (13.0 + H)
 210
           PG(6) = GCON * (13.0 + H)
 220
           DPBS = 0.
 230
           DPGAB = 0.
 240
           DPG(1) = 0.
 250
           DPG(2) = 0.
 260
           DPG(3) = 0.
 270
           DPG(4) = 0.
           DPG(5) = 0.
 280
 290
           DPG(6) = 0.
 300
           PGAVB = GCON * 13.* (1. + COSPHI)
. 310
           DPGAVB = 0.
 320
           SPG = 0.
 330
           SDPG = 0.
 340
           DO 10 J=1.6
           SPG = SPG + PG(J)
 350
 360
        10 \text{ SDPG} = \text{SDPG} + \text{DPG}(J)
 370
           PGAV = SPG / 6.
           DPGAV = SDPG / 6.
 390
 390
           WR1TE(0,98)
        98 FORMAT( //. ' PGRAV SUBROUTINE ',//,
 400
              PGAV, DPGAV, PGAVB, DPGAVB, PGAB, DPGAB',/,
 410
          1
 420
               ' PG, / , DPG', / )
          2
 430
           WRITE(6,99) PGAV, DPGAV, PGAVB, DPGAVB, PGAB, DPGAB, PG, DPG
 440
        99 FORMAT(1X,6E12.5)
 450
           WRITE(6, 96)
 460
        96 FORMAT( /, 1X, ' GR, PHI, COSPHI, GD, GT, H', / )
           WRITE(6,99) GR, PHI, COSPHI, GD, GT, H
 470
 480
       101 CONTINUE
 490
           RETURN
 500
           END
```

ready

```
510
         FUNCTION PELAST ( KTEST. I. V )
520
         REAL#8 V. HP. PELAST
         COMMON /BLK5/ A(7.8)
530
540
         DATA HP / 1.570796327 DO /
550
         IF (V - A(2.1)) 1.1.2
560
       1 \text{ VD1F} = A(2,1) - A(4,1)
570
         GO TO 3
580
       2 \text{ VDIF} = A(3,I) - A(2,I)
       3 IF ( KTEST ) 5,4,4
590
600
       4 \text{ DUM} = \text{HP} * (V - A(2.1)) / VDIF
610
         DTAN = DSIN(DUM) / DCOS(DUM)
620
         PELAST = VDIF \neq DTAN / A(6,I) + A(5,I)
630
         RETURN
640C
       5 PELAST = 1./ ( A(1,I) * ( DCOS( HP * ( V-A(2,1) ) / VDIF ) )**2 )
650
660
         RETURN
670
         END
680
         SUBROUTINE FLOW ( 1, J )
690
         IMPLICIT
                  REAL*3(A-H, O-Z)
700
         REAL*8 NW
710
         REAL*4 B. W
         COMMON /BLK3/ B(8.12), FS(12), RA(12), RB(12), W(12), DTM
720
         COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
730
             F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PBS,
740
        1
             DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
750
        2
             PG(6), DPG(6), PGAB; DPGAB, PPL, DPS(6), DPS2,
760
        3
        4
             S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
770
780
        5
             SQT(12), DP(12), HDT
790C
008
         PDIF = P(I) - P(J)
810
         IF ( 2-I ) 1.2.2
820
       1 K = J
830
         1F ( J. EQ. 5. OR. J. EQ. 6 ) K = J + 2
         DELP = (P(I) + P(J))/2. - P(K)
840
850
         VAVG = (V(I) + V(K)) / 2.
         RA(J) = (B(1,J) + B(2,J) * DELP) / (1. + B(3,J) * VAVG)
860
         RB(J) = (B(4,J) + B(5,J) * DELP + B(6,J) * (2, * PPL - DELP)) /
870
880
               (1. + B(7.J) * VAVG)
        1
890
         GO TO 3
900
       2 RA(J) = B(1,J)
910
         RB(J) = B(4,J)
920
       3 SIGN = 1.
930
         IF( PDIF + W(J)  * FS(J) / DTM. LT. 0. ) SIGN = -1.
940
         DUMY =( RA(J) + W(J) / DTM )**2 + 4. * SIGN * RB(J) *
950
                  (PDIF + W(J) * FS(J) / DTM)
         1
         IF ( DUMY. GE. .0 ) GO TO 11
960
         WRITE (6,12) T, I, J, DUMY, RA(J), RB(J), PDIF, FS(J), SIGN
970
      12 FORMAT(2X, 'SQRT FAILURE IN FLOW, T, I, J, DUMY, RA, RB,
980
990
         &
              PDIF, FS(J), SIGN =',/,2X, F10.5, 2I3, 5E12.4, F5.1 )
1000
          STOP 13
```

ready

```
1010
        11 \text{ SOT}(J) = D \text{SORT}(D U MY)
1020
           IF (.1D-12 - ABS(PDIF)) 6, 5, 5
         5 F(J) = 0.
1030
1040
           RETURN
1050C
1060
         6 F(J) = (-RA(J) - W(J) / DTM + SQT(J)) / (2. * RB(J) * SIGN)
1070
           RETURN
1080
           END
           SUBROUTINE MOLES( F, DNT, NG, HDT, C, DC, DN, DDNW, CW, DCW, J. K)
1090
1100
           IMPLICIT REAL*8( A-H, O-Z )
           DIMENSION C(12.4), DC(12.4), CW(12), DCW(12), DN(12.4)
1110
1120
           COMMON /BLK12/ DIFF(6,3)
           DNT = 0.
1130
1140
           IF( F ) 5, 10, 15
         5 DO 6 I= 1, NG
1150
           DN(K,I) = F * C(K,I) - DIFF(K-6,I)
1160
         6 \text{ DNT} = \text{DNT} + \text{DN}(K, I)
1170
1160
           DDNW = F * CW(K)
1190
           RETURN
 1200
        10 DO 11 I= 1, NG
        11 DN(K,I) = 0.-DIFF(K-6,I)
 1210
           DDNW = 0.
 1220
 1230
           RETURN
 1240
        15 DO 16 I= 1, NG
           DN(K,I) = F * (C(J,I) + HDT * DC(J,I)) - DIFF(K-6,I)
 1250
 1260
        16 DNT = DNT + DN(K,I)
           DDNW = F * (CW(J) + HDT * DCW(J))
 1270
 1280
           RETURN
 1290
           END
ready
 ¥
 10
         SUBROUTINE FORCE ( T. FSPR, DT )
         REAL*3 T, PTX, PD, DPTX, DPD, PHI, GF, GR, GT, GD, HP, PIE,
 20
 30
        1
                ANG1, ANG2, PHI1, PHI2, FSPR, FSPRS, FCT, DT
 40
         INTEGER FBOPT
 50
         DIMENSION AA(8,2)
 60
         COMMON /BLK4/ TIN, TEX, PDIN, TB, Y1, Y2, Y3, Y4, PTXIN, PTXEX,
                        PDEX, PDAG, PTXG, TO, T2, PB, FCT, FSPRS, TTEST
 70
        1
 80
         COMMON /BLK8/ PnI, GF, GR, GT, GD, HP, PIE, ANG1, ANG2, PHI1, PHI2
         COMMON / BLK11/ PTX, DPTX, PD, DPD, G1, G2, G3, G4, FEOPT
 90
 100
          DATA TO, TTEST, TBRTH, ITST, CONST, NFRC / 3* 0.0, 1, 1., 0 /,
              AA / .39, 4., .50, .25, .11, .30, 2.1, 3.,
 110
         &
                    .43, 4., .23, .25, .36, .15, 1.5, 3. /
 120
          DATA A, B, C, ALP, BET, GAM, DEL, EPS / .40, .60, 114.0, .25,
 130
 140
         &
               80., 110., 1., 2. /, FMX / 1.0 /
 150C
 160
          IF ( NFRC. EQ. 1 ) GO TO 105
```

ready

```
142
```

```
180
         1F ( 1TST. EQ. 3 ) 1TST = 1
190
         IF( ITST. EQ. FBOPT ) NFRC = 1
200
         1F ( NFRC. EQ. 1 ) GO TO 100
210
         ITST = I1ST + 1
220
         B1 = ABS(PTXIN - PTXEX) / PB
230
         B3 = ABS (PDIN - PDEX) / PB
240
         ASQR1 = TB \neq TB
250
         TES = TB
260
         515 = 51
         DU 1 J= 1,50
270
         1F ( 1TST. EQ. 2 ) DPTX = ( PTXIN + B1S - PTXEX ) * Y1 / ( 1.-
2ò0
290
             EXP(-Y1 " (TIN - TbP))
        1
300
         IF ( 1TST. EQ. 3 ) DPTX = Y2 * ( PTXIN + B1S - PIXEX) / ( 1.-
             DEXP(-Y2 * (TEX - TBP))
310
        1
320
         TBP = ABS( DPTX ) * ASQR1 / SQRT( B1*B1 + ASQR1 * DPTX*DPTX )
350
         ASQR = TEP * TBP
         DUMY = SQRT( 1.- ASQR / ASQR1 )
340
350
         ETX = B1 - B1 + DUMY
         1F ( ABS( ETX - E1S ). LT. 0.001 ) GO TO 2
360
370
         B1S = ETX
300
       1 \text{ TbS} = \text{TbP}
       2 bda = 83 - 83 * DUMY
390
         C1 = PTXEX - ETX
400
410
         C2 = PTX1N + bTX
420
         C_3 = PDEX - BDA
430
         C4 = PDIN + BDA
440
         EA1 = BTX / ASQR
450
         BA3 = BDA / ASQR
460
         PTXDM = PTXIN + BTX
470
         PDD = PD1N + BDA
         1F ( 1TST - 3 ) 0, 7, 7
450
       5 P1N2 = ( PTXDM - PTXEX ) / ( 1.- DEXP( -Y1 * ( TIN - TEP ) ) )
490
500
         PIN_{5} = (PDDM - PDEX) / (1.- EXP(-Y3 * (TIN - TEP)))
510
         TTEST = TTEST + TIN
520
         TO = TO + TERTH
530
         T1 = T0 + TbP
540
         T2 = T0 + T1N
550
         T3 = T2 + TBP
560
         TBRTH = TEX + TIN
570
         10P = T0 - .0001
         T1P = T1 - .0001
580
590
         T2P = T2 - .0001
600
         T3P = 13 - .0001
610
         GO TO 8
620
       7 IF ( FEOPT. LT. 10 ) GO TO 71
630
         T3P = T2P
640
         T3 = T2
650
         TBP = 0.
060
         PTXDM = PTXIN
```

IF (T. LT. TIEST - .0001) GO TO 5

170

```
670
         PDDM = PDIN
        PEX2 = 1./(.455*(1.-EXP(-1.95 * TEX)) + .335*(1. -
650
                EXP(-.1 * TEX**2)) + .21* (1.- EXP(-.11* TEX**3)))
690
        &
700
         PEX3 = (PDDM - PDEX) * PEX2
         PEX2 = ( PTXDM - PTXEX ) * PEX2
710
720
         WRITE (6,1221) PEX2, PEX3, PTXDM, PDDM
730 1221 FORMAT( 1X, 'PEX2, PEX3 =', 4E15.4 )
740
         GO TO 72
      71 \text{ PEX2} = (\text{ PTXDM} - \text{PTXEX}) / (1.- \text{DEXP}(-\text{Y2} * (\text{TEX} - \text{TBP})))
750
700
         PEX3 = (PDDM - PDEX) / (1.- DEXP(-Y4 * (TEX - TBP))).
110
      72 TTEST = TTES1 + TEX
700
       8 CONTINUE
790C
800
       5 CONTINUE
310
         IF ( NFRC. EQ. 1 ) GO TO 100
v20
         1F ( T. GE. 13P ) GO TO 75
830
         IF ( T. GE. T2P ) GO TO 50
840
         IF ( T. GE. T1P ) 30 TO 25
850C
00dd
      INSPIRATION: PART 1
8700
800
         TDIF = T - TO
         TSQR = TD1F ➡ TD1F
090
900
         D1 = SORT(1. - TSOR / ASOR1)
910
         DUM = TDIF / (D1 \neq ASQR1)
920
         PTX = PTXEX - B1 + B1 * D1 + PTXG * GT
930
         DPIX = -BI = DUM
940
         PD = PDEX - B3 + B3 * D1 + PDAG * GD
950
         DPD = -B3 * DUM
930
         RETURN
970C
900C
      INSPIRATION: PART 2
9900
1000
       25 D1 = EXP(-Y1 * (T - T1))
1010
          D3 = EXP(-Y3 * (T - T1))
1020
          PTX = C1 + P1N2 * (1.- D1) + PTXG * GT
1050
          DPTX = PIN2 * Y1 * D1
1040
          PD = C3 + PIN3 * (1.- D3) + PDAG * GD
1050
          DPD = PIN3 * Y3 * D3
1060
          RETURN
1070C
1000C
      EXPIRATION: PAR1 1
10900 ,
1100
       50 IF ( FEUPT, EQ. 10 ) GU TO 200
1110
          TD1F = T - T2
1120
         TSQR = TD1F * TD1F
          D1 = SQRT( 1.- TSQR / ASQR1 )
1130
          DOM = TDIF / (D1 * ASQR1)
1140
1150
          PTX = PTXIN + B1 - B1 * D1 + PTXG * GT
1100
          DPTX = B1 * DUM
```

*

```
PD = PD1N + B3 - E3 * D1 + PDAG * GD
1170
1100
         DPD = b3 * DUM
1190
         RETURN
12300
1210C
      EXPIRATION: PART 2
12200
      75 1F ( FEOPT. EQ. 10 ) GO TO 200
1230
          D1 = EXP(-Y2 * (T - T3))
1240
1250
          D_3 = EXP(-Y4 = (T - 13))
          PTX = PTXDM - PEX2 * (1.- D1) + PTXG * GT
1200
1270
          DPTX = -PEX2 = Y2 = D1
1250
          PD = PDDM - PEX3 * (1.- D3) + PDAG * GD
          DPD = -PEX3 + Y4 + D3
1290
1300
          RETURN
13100
     100 CONTINUE
1320
1330
          IF ( T. LT. TTEST - .0001) GO TO 105
          TW = TTEST
1340
1350
          TIEST = TTEST + TEX
          COR = A * (1. - EXP(-ALP * EXP(EPS * ALOG(TEX))) +
1360
                B = (1. - EXP(-BET * TEX)) +
13/0
         å
               C * TEX * EXP( -GAM * EXP( DEL * ALOG( TEX ) ) )
1300
         λe –
          PTXD1F = ( PTXEX - PTXIN ) / COR
1390
1400
          PDD1F = (PDEX - PDIN) / COR
1410C
1420 ' 105 CONTINUE
1430
          TT = T - TN
1440
          ALNT = ALOG(TT)
1450
          TEPSM= EXP( ( EPS -1. ) * ALNT )
          TEPS = EXP( EPS * ALNT )
1400
          TDEL = EXP(DEL * ALNT)
1470
          EXP1 = EXP( -ALP * FEPS )
1400
1490
          EXP2 = EXP( -BET * TT )
          EXP3 = EXP(-GAM * TDEL)
1500
          FCT = A * ( 1.- EXP1 ) + B * ( 1.- EXP2 ) + C * TT * EXP3
1510
          DFCT = A * ALP * EPS * TEPSM * EXP1 + B * BET * EXP2 +
1520
1530
                 C * EXP5 * (1.- DEL * GAM * TDEL)
         ů
1540
          PTX = PTXG * GT + PTXIN + PTXDIF * FCT
          PD = PDAG * GD + PDIN + PDDIF * FCT
1550
          DPTX= DFCT * PTXDIF
1500
1570
          DPD = DFCT * PDD1F
1500
          RETURN
1590
      200 CONTINUE
          TDIF = T - T2
1600
1610
          TT = 1D1F - 2.10
1ó20
          PTX = .455 * ( 1.- EXP( -1.95 * TD1F ) ) + .535 * ( 1.- EXP( -.1 *
1630
              TD1F**2 ) )
         å
1640
          DPTX = .80725 * EXP ( -1.95 * TDIF ) + .067 * TDIF * EXP (-.1
1650
             * TD1F*TD1F )
         &
1600
          Ir (TT) 90,90,85
```

ready

```
35 PTX = PTX + .20 * ( 1. - EXP(-.11 * TT**3 ) )
1670
1630
          DPTD = DPTX + .065 * TT*TT * EXP(-.11 * TT**5)
1690
       90 PD = PDDM - PEX3 * PTX + PDAG * GD
          PTX = PTXDM - PEX2 * PTX + PTXG * GT
1/00
1710
          DPD = -DPTX * PEX3
          DPTX = -PEX2 * DPTX
1720
           IF(T.GE.1.55) WRITE(6,1222) T, PTX, DPTX, PD, DPD
17500
1740
          RETURN
1/50 1222 FORMAT(1X, 'T, PTX, DPTX, PD, DPD =' ,F5.2, F7.2, E10.2, F7.2,
1700
         č.
              £10.2)
1770
          END
          SUBHOUTINE AIRWAY ( DN. DDCW )
1700
1790
          IMPLICIT REAL*8( A-H, O-Z )
1800
          REAL*3 NW
          DIMENSION X(5), DDCW(12), FI(6,4), FO(6), A(5,5), FIW(6), E(5),
1810
1820
         1
              DN(12,4)
          COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
1030
1040
         1
              F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PES,
1850
         2
              DPES, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
1000
         З
              PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
         Ц
              S(4), 1, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
10/0
1800
         5
              SQT(12), DP(12), HDT
1390
          DO 10 J = 1.5/
       10 \ Flw(J) = 0.
1903
1910
          DO 20 J= 2.0
          DNT(J) = 0.
1920
1930
          DÜ 20 1=1,NG
1940
       20 \ fl(J,I) = 0.
          1F(F(2)) 30, 50, 40
1950
       30 FO(2) = -F(2)
1960
1970
          30 10 50
1980
       40 DO 45 I= 1, NG
1990
       45 \text{ F1}(2,1) = F(2) * C(1,1)
2000
          FIW(2) = F(2) * CW(1)
       50 DO 80 J= 3,0
2010
2020
          K_3 = J - 1
2030
          K = J - 3
2040
          1F(J. EQ. 3) K = 1
2050
          X2 = K + 1
2060
          IF(F(J)) 60, 80, 70
2070
       60 F \hat{U}(J) = -F(J)
2030
          A(K,K3) = -F(J)
2090
          DO 65 I= 1, NG
2100
       65 \ FI(K2,I) = FI(K2,I) - F(J) = C(J,I)
2110
          FIW(K2) = FIW(K2) - F(J) * CW(J)
2120
          GO TO 30
2130
       70 FO(K2) = FO(K2) + F(J)
2140
           A(K3,K) = F(J)
2150
          DO 75 I= 1, NG
2160
       75 \text{ F1}(J,I) = F(J) * C(K2,I)
```

×

```
FIW(J) = F(J) * CW(K2)
2170
2100
       30 CONTINUE
      170 DO 230 J= 7,12
2190
2200
          K = J - 4
          1F ( J. GE. 11 ) K = J - 6
2210
2220
          IF( F(J) ) 210, 230, 220
2230
      210 DO 215 I= 1, NG
      215 FI(K,I) = FI(K,I) - F(J) * C(J,I)
2240
2250
          FIW(K) = FIW(K) - F(J) * CW(J)
2260
          40 10 230
2270
      220 FO(K) = FO(K) + F(J)
2200
      230 CONTINUE
2290
      350 DU 360 K= 1, 5
2300
      360 A(K,K) = -(V(K+1) / HDT + FO(K+1))
2310
          DO 360 I= 1, NG
2320
          DO 370 K=2, 6
      370 B(K-1) = (C(K,I) * FO(K) - FI(K,I)) / HDT
2330
2340 3003 FORMAT ( /, 1H , 5(1X, 5E12.6, 4X, E12.6, //) )
2350
          CALL BAPM( A, X, E )
2300
          DO 375 J=2.6
2370
          DC(J,1) = X(J-1)
          DN(J,1) = DC(J,1) * V(J)
2330
2390
      575 \text{ DNT}(J) = \text{DNT}(J) + \text{DN}(J,I)
2400 3004 FORMAT(5X, 5E15.6 )
2410
      300 CONTINUE
2420
          DO 390 J= 2, 6
2430
      390 B(K-1) = (CW(K) * FO(K) - FIW(K)) / HDT
2440
          CALL BAPM( A. X. B )
2450
          DO 400 J= 2, o
2460
      400 \text{ DDCW}(x) = x(x-1)
2470
          RETURN
2480
          END
          SUBROUTINE PMAX ( FRCP, VMAX, PG, PBS, GT, GD, PGAE,
2490
                      FCIN, VIX, VD, PPL, VA )
2500
         1
          REAL*5 PG, P6S, GT, GD, PGAV, DPGAV, PGAVB, DPGAVB, FCTST,
2510
                  V, VTX, VD, VDM, VTXM, PELAST, PGAB, PPL, VA, FSPRS
2520
         1
2530
          REAL*O DP1(0), DUMYS
          DIMENSION PG(6), A(7), B(7), C(7), XP(7), DELE(7), FACT(7),
2540
2550
         1
                     KNT2(\gamma)
          COMMON /BLK4/ TIN, TEX, PDIN, TB, Y1, Y2, Y2, Y4, PTXIN, PTXEX,
2560
2570
                         PDEX, PDAG, PTXG, TO, T2, PB, FCTST, FSPRS, TTEST
          1
2550
          COMMON /BLK5/ AA(7, \delta)
2590
          COMMON /BLK10/ PGAV, DPGAV, PGAVB, DPGAVB
2000C
2610
          VLG = FRCP + VeiAX - VA
2620
          PPLM = PPL
2630
          XP(7) = PPLM
2640
          DO 25 J=1,6
          XP(J) = VLG / \acute{o}.
2650
2050
          C(J) = PG(J) - PBS
```

ready

```
20/0
       25 CONTINUE
2680
          C(\gamma) = VLG
          DO 35 J= 1, 7
2690
2700
          FACT(J) = .5
       35 \text{ KNT2}(J) = 0
2710
27200
2730
          DO 1000 KNT = 1, 30
2740
          SB = 0.
2750
          SV = 0.
          SA = 0.
2700
2110
          SAW = 0.
          DO 50 J= 1,6
2700
2790
          JDUMY = J
2800
          V = XP(J)
          B(J) = (PELAST(0, JDUMY, V) + XP(7) + C(J))
2810
2620
          A(J) = PELAST(-1, JDUAY, V)
2000
          SE = SE + AES(E(J))
          SV = SV + \lambda P(J)
2040
          SA = SA + 1./A(J)
2050
          SAW = SAW + b(J) / A(J)
2000
2070
       50 CONTINUE
2000
          B(7) = (SV - C(7))
          SB = SB + ABS(-b(7))
2090
29000
          IF ( SB. LT. 0.01 ) GO TO 1005
2910
2920C
2930
          B(7) = (B(7) - SAW) / SA
2940
          Dù 55 J= 1,0
2950
       55 B(J) = -(B(J) + B(7)) / A(J)
29500
2970
          DO 60 J = 1,7
2930
          IF ( KNT2(J). EQ. 1 ) GO TO 206
          1F ( DELB(J). EQ. 0.. GR. B(J). EQ. 0. ) GU TU 200
2990
          1F ( DELB(J) / B(J) ) 205, 205, 204
3000
3010
      204 IF ( KNT2(J). LT. 3 ) GO TO 200
          FACT(J) = 1.0
3020
3030
          GO TO 206
      205 \text{ FACT}(J) = 0.5
3040
          KN12(J) = 0.0
3050
3050
      206 B(J) = FACT(J) * B(J)
          D \subseteq LB(J) = B(J)
3070
3000
          1F ( J. EQ. 7 ) GO TO 60
      209 DUMY = ABS( DELB(J) )
5090
          lr ( DUMY. Gr. 0.2 ) DELB(J) = 0.2 * DUMY / DELE(J)
3100
5110
       60 XP(J) = XP(J) + DELE(J)
          DO 70 J= 1,6
3120
3130
       70 IF ( XP(J). GE. AA(3,J) ) XP(J) = AA(3,J) - .01
3140 1000 CONTINUE
          WRITE(6, 98) XP
3150
3100
          WRITE(6,99) KNT, SB
```

```
99 FORMAT( 5X. 'KNT ='.I5, 5X. 'SB ='. E10.3 )
3170
       93 FURMAT(5X, 9E11.4)
3150
31900
3200
          STOP 3
32100
3220 1005 CONTINUE
      111 FORMAT( 5X. 7E15.8 )
3230
3240
          PPLM = XP(7)
3250
          C(1) = PLS + PTXG * GT - PGAV - PPLM
3250
          C(2) = PBS + PDAG * GD - PGAVB - PPLM - PGAB
3270
          C(3) = FCTN
          C(4) = VLG + .005
3200
3290
          ADUM1 = AA(5, 7) - .001
          ADUM2 = AA(4,7) + .001
3300
3310
          ADUM3 = AA(3,8) - .001
3320
          ADUM4 = AA(4.3) + .001
          VTXM = VTX + .5 * VMAX
5550
3340
          IF ( VTXM. GE. ADUM1 )
                                  VTXM = ADUM1
3350
          IF ( VTXM. LE. ADUM2 ) VTXM = ADUM2
          VDM = C(4) - VTXH
3300
3370
          IF ( VDM - ADUM3 ) 1205, 1201, 1201
3300 1201 VDM = ADUM3
3390
          GU TO 1209
3400 1205 IF ( VDM - ADUMA ) 1205, 1206, 1210
3410 1200 VDM = ADUM4
3420 \ 1209 \ VTXH = C(4) - VDM
          IF ( VTXM. LT. ADUM2. OR . VTXM. GT. ADUM1 ) GO TO 2001
3430
3440 1210 CUNTINUE
34500
          DO 2000 KNT = 1,50
3400
3470
          VTXMN = VTXM + (C(3) * (PELAST(0, 8, VDM) - C(2)) -
              PELAST(0, \gamma, VTXM) + C(1)) / (C(3) \sim PELAST(-1, \sigma, VDM)
3400
         1
              + PELAST(-1, 7, VTXM))
3490
         2
          VDM = C(4) - VIXMN
3500
3510
          IF( ( VTXER. LE. ADUM2. OR. VTXMN. GE. ADUM1 ). AND .
3520
               ( VDM. LE. ADUH4. OR. VDM. GE. ADUM3 ) ) GO TO 2001
          1
3530
          IF ( ABS( VIXMN - VIXM ). LT. 0.0005 ) GO TO 2005
          VTXH = VTXMN
3540
3550 2000 CONTINUE
35600
3570 2001 WRITE(6,101) VDM, VTX
      101 FORMAT(//5X, '******* VDM = ',F10.5, ' VTX = ',F10.5,
3500
3590
          1
               'ONE OF THE ABOVE IS TO LARGE, CHECK A(7,8) AND FOIN')
3600
          STOP 4
.3610 2005 CUNTINUE
3620
          WRITE(6,99) KNT
3630
          VTXM = VTXMN
3640
          DUM1 = PELAST(0, 7, VTXM)
          DUM2 = PELAST(0, 8, VDM)
3650
3660
          DU_{M3} = PELAST(0, 7, VTX)
```

Ż

```
3670
          DUM4 = PELAST(0.8, VD)
          CTXM = 1.7 PALAST(-1, 7, VTXM)
3680
3640
          CDAM = 1./ PELAST(-1, 8, VDM)
      988 FORMAT(2X. 5E12.6)
3700
          PDIN = DUM2 - C(2)
3710
3720
          PTXIN = DUM1 - C(1)
          WRITE(6.939) DUM1. PPLM. PGAV. PTXG. GT.
3730
              DUM2, PGAB, PDAG, GD, PGAVB
3740
         1
      989 FORMAT( 2X, 'PELTXM, PPLM, PGAV, PTXG, GT, PELDAM, PGAB, PDAG, GD,
3750
              PGAVB = ',/,3X, 5E12.6, / 5E12.6, // )
5700
         1
3770
          WRITE( 6,97 ) PTXIN, PDIN, VTXM, VDM, CIXM, CDAM
       97 FORMAT( //, 7x, 'PTXIN
                                                                           VDM'.
3700
                                           PDIN
                                                           VTXM
                                                  CDAM'.//.1X.6E15.8. //)
3140
                                  CTXM
         1
3000
      997 FORMAT( 5F10.5 )
3010
          DU 7/11 J=1.6
          JD = J
3020
3830
          DUMYS = XP(J)
5040
          DP1(J) = PELAST(0, JD, DUMYS)
3050 7711 CONTINUE
          WRITE (6,7712) (DP1(J),J=1,6)
3860
3670 7712 FORMAT(//,1x,'PELAST =', 6F9.3 ,//)
3000
          RETURN
          END
3840
3900
          FUNCTION FRC2( VA. VTX, PD, PTX )
3910
          IMPLICIT REAL*5 ( C-H, O-V )
3920
          REAL*3 NW
3930
          COMMON /BLK5/ AA(7, b)
          COMMON /BLK10/ PGAV, DPGAV, PGAVB, DPGAVB
3940
          CUMMON C(12,4), LC(12,4), CT(12), Cw(12), DCw(12), DCwDP(12),
3950
               F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PES,
3960
         1
3970
         2
               DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
3900
         3
               PG(6), DPG(6), PGAE, DPGAB, PPL, DPS(6), DPS2,
         4
               S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLU<sub>NG</sub>,
3990
               SQT(12), DP(12), HDT
4000
         5
          DIMENSION A(9), E(9), XP(9), ZPK(9), KNT2(9),
4010
4020
               DELE(9), FACT(9)
          1
40.50
          VA = 0.
4040
          DO 15 1= 2.0
4050
       15 VA = VA + V(1)
4000
          FRC2 = VA
4070
          XP(9) = PPL
4000
          XP(\gamma) = VTX
          \lambda P(8) = VD
4090
4100
          DO 30 J= 7,12
4110
           XP(J-b) = V(J)
4120
           2PK(J-5) = P(J) - PG(J-5)
      901 FORMAT( 5X, 'V, P, PG, ZPK', 12, '=', 4E15.7 )
4130
4140
       30 CONTINUE
4150
           ZPK(7) = -PGAV + PTX + PBS
           ZPK(8) = PD + PBS - PGAB - PGAVB
4100
```

Ŕ

```
4170
          ZPK(9) = 0.005
4100 210 FURMAT (3X, 9E11.5)
          DU 35 J= 1,9
4190
4200
          FACT(J) = .5
4210
       35 \text{ KNT2}(J) = 0.
4220C
4230
          DO 1000 KNT = 1, 100
4240
          SA = 0.
4250
          SAW = 0.
4230
          VLUKG = 0.0
4270
          DU 40 J = 1,6
4200
          JDUMY = J
4290
          VLUNG = VLUNG + XP(J)
4300
          A(J) = PELAST(-1, JDUMY, V(J+6))
4510
          PTEST = PELAST(1, JDUMY, V(J+6))
4320
          B(J) = (XP(9) - ZPK(J) + PTEST)
4350
          SA = SA + 1./A(J)
4340
          SAW = SAW + E(J) / A(J)
4350
      902 FORMAT( 1X, 'A, B, XP1, ZPK, PEL', 12, '=', 5E17.10 )
4360
       40 CONTINUE
4370
          A(7) = PELAST(-1, 7, VTX)
4300
          A(8) = PELAST(-1, 8, VD)
          b(7) = (XP(9) - 2PK(7) + PELAST(1, 7, VTX))
4390
4400
          B(\delta) = (\lambda P(y) - ZPK(\delta) + PELAST(1, \delta, VD))
4410
          B(9) = (VIX + VD - ZPK(9) - VLUNG)
4420
          Sb = 0.
4430
          DU 50 J= 1,9
4440
       50 SE = SE + AES( b(J) )
4450
          IF ( KNT. GE. 100 ) WRITE (6,98) B
4400
          IF ( SB. LT. 0.0003 ) GO TO 1001
4470C
4430
          B(5) = (B(9) - B(7) / A(7) - B(5) / A(8) + SAW) / (1./ A(7))
4490
                 + 1./ A(d) - SA)
         1
4500
          DO 55 J=1,8
4510
       55 \ B(J) = -( \ E(J) + B(9) ) / A(J)
4520C
          lf ( KNT. GE. 100 ) Walte(6,93) B
4530
4540
          DO 60 J = 1,9
4550
          IF ( KNT2(J). EQ. 1 ) GO TO 206
4560
          IF ( DELB(J). EQ. 0.. OR. B(J). EQ. 0. ) GO TO 206
4570
          1F ( DELB(J) / B(J) ) 205, 205, 204
4580
      204 1F ( KNT2(J). LT. 5 ) GO TO 206
4590
          FACT(J) = 1.0
4600
          GO TO 206
4010
      205 FACT(J) = 0.5
4620
          KN12(J) = 0.0
4630
      205 B(J) = FACT(J) * B(J)
4640
          DELE(J) = E(J)
4650
          1F ( J. EQ. 9 ) GO TO 60
4060
          1F ( J-7 ) 20/, 208, 208
```

```
4670
       207 \text{ TEST} = 0.2
 4600
            GO TO 209
 4690
       208 \text{ TEST} = 0.5
 4700
       209 DUMY = ABS(DELB(J))
            1F ( DUMY. GT. TEST ) DELB(J) = TEST * DUMY / DELB(J)
 4710
 4720
         60 \text{ XP}(J) = \text{XP}(J) + \text{DELB}(J)
 4730
            DO 70 J = 1.6
 4740
            1F ( XP(J). LE. AA(4,J) ) XP(J) = AA(4,J)
 4750
            IF ( XP(J). GE. AA(3,J) ) XP(J) = AA(3,J)
        70 V(J+b) = XP(J)
 4760
 4770
            PFC = XF(A)
            IF ( XP(7). LE. AA(4,7) ) XP(7) = AA(4,7)
 4700
 4790
            1F ( XP(7). GE. AA(5,7) ) XP(7) = AA(3,7)
 4000
            VTX = XP(7)
            IF (XP(6), LE. AA(4,8)) XP(8) = AA(4,8) + .0002
 4810
 4820
            IF ( XP(\delta). GE. AA(3,8) ) XP(8) = AA(3,3) - .0002
 4850
            VD = XP(3)
        111 FORMAT(1X, 'DELB =', 5E17.10, /7X, 4E17.10, /' XP =', 5E17.10, /
 4840
                .7X.4E1/.10,/4X,'V = '.6E17.10)
 4850
           1
 4800
         99 FORMAT( 5x, 'KNT =', I3, 5X, 'SB =', E10.3 )
 4870
         90 FORMAT(5X, 9E11.4 )
 4080 1000 CONTINUE
 4090
            WRITE(6,99) KNT, SB
 4900
            WRITE(6, 98) E, XP
 4910C
 4920
            STOP 1000
 4930 1001 CONTINUE
 4940
            PELTX = PELAST(1, 7, VTX)
 4950
            PELDA = PELAST(1, 0, VD)
 4900
            WRITE(0,99) KNT, SE
 4970
            FRC2 = FRC2 + VLUNG
            VLUNG = FRC2
 4900
 4990
            PPL = XP(9)
            VTX = XP(7)
 5000
            VD = XP(o)
 501ú
            RETURN
 5020
· 5030
            END
```

```
ready
```

```
SUBROUTINE SUB1 ( I, J )
10
20
        REAL*8 NW
30
        IMPLICIT REAL*8( A-H, O-Z )
        COFAMON /BLK6/ A(5,5), FO(6), FI(6), SQO(6), SQ1(6)
40
50
        COMMGN C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
             F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PES.
60
       1
70
             DPES, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
       2
80
             PG(6), DPG(6), PGAE, DPGAB, PPL, DPS(6), DPS2,
       3
90
       4
            S(4), T. LT. TN(12), TOTN(12), V(12), VD, VLUNG,
100
        5
             SQT(12), DP(12), HDT
110C
120
         IF( J. GT. 2 ) GO TO 50
130
         IF(F(J)) 10, 20, 30
140
      10 FO(2) = -F(2)
150
         SOU(2) = +1./SOT(2)
160
      20 RETURN
170
      30 \text{ FI}(2) = CT(1) * F(2)
180
         SQ1(2) = CT(1) / SQT(2)
         RETURN
190
200
      50 IF( J. GT. 6 ) GO TO 100
210C
      *** ALRWAYS
220
         IF(F(J)) = 60.70.80
230
      \delta 0 F \tilde{U}(J) = F U(J) - F(J)
240
         FI(1) = FI(1) - F(J) * CT(J)
250
         SQG(J) = SQG(J) + 1./ SQT(J)
         SQI(1) = SQI(\overline{1}) + CT(\overline{J}) / SQT(\overline{J})
260
270
         A(I-1,J-1) = -F(J) * (1./G - DCWDP(J)) + CT(J) / SQT(J)
200
         A(J-1, 1-1) = -CT(J) / SQT(J)
      70 RETURN
290
300
      80 \text{ F1}(J) = F1(J) + F(J) * CT(1)
310
         FO(1) = FO(1) + F(J)
         SQI(J) = SQI(J) + CT(I) / SQT(J)
320
         SQO(1) = SQO(1) + 1./ SQT(J)
330
340
          A(1-1, J-1) = CT(1) / SQT(J)
         A(J-1, I-1) = F(J) \times (1./G - DCWDP(I)) + CT(I) / SQT(J)
350
         RETURN
360
370C *** LUNG CHAMBERS
     100 IF( F(J) ) 110, 120, 130
300
     110 FI(I) = FI(I) - F(J) * CT(J)
390
400
          SQI(1) = SQI(1) + CT(I) / SQT(J)
410
     120 RETURN
420
     130 FO(I) = FO(I) + F(J)
          SQO(1) = SQO(1) + 1./ SQT(J)
430
440
          RETURN
450
          END
460
          SUBROUTINE SUB2( ASUM, ASUM1, AWSUM, AWSUM1, RES, PV, RTOT, PPLC )
470
          IMPLICIT REAL*O( A-H, O-Z )
400
          REAL *O N, NW
          REAL*4 B, W, ASUM, ASUM1, AWSUM, AWSUM1, RES, PV, RTOT, PPLC
490
500
          DIMENSION ASUM1(4), ASUM(4), PV(12), RES(12), PPLC(3)
```

```
ready
```

```
COMMON /BLK1/ N(12.3), VTX, VALV(6), PALV(6), SW(12), VSPR
510
         CUMMON /BLK3/ B(0,12), FS(12), RA(12), RE(12), h(12), DTM
520
         COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCwDP(1∠),
530
             F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PES,
540
        1
             DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX.
550
        2
560
        3
             PG(o), DPG(o), PGAE, DPGAE, PPL, DPS(o), DPS2,
570
        4
             S(4), T. DT. TN(12), TOTN(12), V(12), VD. VLUNG,
530
        5
             SQT(12), DP(12), HDT
590
         D0 100 I = 1.NG
600
     160 \text{ ASUM}(1) = -ASUM1(1)
610
         AVSUM = -AVSUM1
620
         DO 170 J= 1.12
630
         AWSUM = AWSUM + NW(J)
640
         DO 170 I = 1.ivG
     170 \text{ ASUM}(1) = \text{ASUM}(1) + \text{N}(J, I)
650
660
         D0 100 1 = 2.12
670
         R \le S(1) = RA(1) + RE(1) * ABS(F(1))
ნაა
     130 PV(I) = P(I) = V(I) / (TOTN(I) + G) - 1.
690
         PV(1) = P(1) * V(1) / (TOTN(1) * GS) - 1.
700
         RR1 = ReS(b) + ReS(10) * ReS(12) / (RES(10) + ReS(12))
         RL1 = RES(5) + RES(9) * RES(11) / (RES(9) + RES(11))
710
720
         RR = RES(4) + RES(8) * RR1
                                           / ( \text{ReS}( 8) + \text{RR1} )
730
         RL = RES(3) + RES(7) * RL1
                                            /(RES(7) + RL1)
740
         RTOT = ReS(2) + RR * RL / (RR + RL)
150
         PPLC(1) = PPL + PG(1)
760
         PPLC(2) = PPL + PG(3)
         PPLC(3) = PPL + PG(5)
770
700
         RETURN
790
         END
800
         SUBROUTINE SUB3( V )
01υ
         REAL*8 V
ŏ20
         DIMENSION V(12)
630
         V(7) = 1.050116
840
         V(b) = V(7)
         V(9) = 0.95276
050
000
         V(10) = V(9)
610
         V(11) = 0.837573
იიი
         V(12) = V(11)
090
         RETURN
900
         END
```

```
ready
```

```
10
        SUBROUTINE BAPM( A, X, B )
20
        10PLICIT REAL*3( A-H, L, M, O-Z )
        DIMENSION A(5,5), X(5), B(5), D(5), L(5), G(5)
30
40
        D(1) = A(1,2) / A(1,1)
50
        L(1) = A(1,3) / A(1,1)
60
        G(1) = E(1) / A(1,1)
        M = A(2,2) - A(2,1) * D(1)
70
        D(2) = -A(2,1) * L(1) / M
80
        L(2) = A(2,4) / M
90
100
         G(2) = (B(2) - A(2,1) \neq G(1)) / M
110
         Z = -A(3,1) = D(1)
         M = A(3,5) - Z * D(2) - A(3,1) * L(1)
120
         D(3) = -Z * L(2) / M
150
         L(3) = A(3,5) / M
140
150
         G(3) = (B(3) - Z * G(2) - A(3,1) * G(1)) / M
160
         Z = -A(4,2) * D(2)
173
         M = A(4,4) - 2 \neq D(3) - A(4,2) \neq L(2)
100
         D(4) = -2 * L(3) / 14
190
         G(4) = (B(4) + 2 * G(3) - A(4,2) * G(2)) / K
         Z = -A(5,3) * D(5)
200
         M = A(5,5) - Z * D(4) - A(5,3) * L(3)
210
         X(5) = (E(5) - Z * G(4) - A(5,3) * G(3)) / H
220
230
         X(4) = G(4) - D(4) * X(5)
240
         X(3) = G(5) - D(3) * X(4) - L(3) * X(5)
250
         X(2) = G(2) - D(2) * X(3) - L(2) * X(4)
2ó0
         X(1) = G(1) - D(1) * X(2) - L(1) * X(3)
270
         RETURN
200
         END
290
         SUBROUTINE RUNGE( N, Y, F, X, H, M, HDT )
300
         IMPLICIT REAL*3(A-H, O-Z)
310
         DIMENSION PHI(100), SAVEY(100), Y(N), F(N)
320
         GO 10 (1,2,3,4), M
330
       1 \text{ DO } 11 \text{ J} = 1, \text{N}
340
         SAVEY(J) = Y(J)
         PHI(J) = F(J)
350
300
      11 Y(J) = SAVEY(J) + 0.5*H*F(J)
310
         X = X + 0.5^{*}H
300
         RETURN
390
       2 DO 22 J= 1,N
400
         PHI(J) = PHI(J) + 2.0 F(J)
      22 Y(J) = SAVEY(J) + 0.5 H + F(J)
410
420
         HDT = H/2.
430
         RETURN
440
       3 DO 33 J= 1,N
450
         PH1(J) = PH1(J) + 2.0*F(J)
460
      33 Y(J) = SAVEY(J) + H^*F(J)
470
         X = X + 0.5  H
400
         RETURN
       4 DO 44 J= 1,N
490
500
      44 Y(J) = SAVEY(J) + (PH1(J) + F(J))*H/6.0
```

×

```
510
         HDT = H/4.
520
         RETORN
530
         END
540
         FUNCTION HAMING( N, Y, F, X, H, TE )
550
         IMPLICIT REAL*8(A-H, O-Z)
         INTEGER HAMING
550
570
         LOGICAL PRED
         DIMENSION YPRED(100), TE(N), Y(N,4), F(N,3)
500
590
         DATA PRED / .TRUE. /
         TF (.NOT.PRED) GO TO 4
600
         ..... PREDICTOR SECTION OF HAMING .....
610C
620
         DO 1 J = 1.N
       1 YPRED(J) = \dot{Y}(J,4) + C1^{*}(2.* (F(J,1) + F(J,5)) - F(J,2))
650
640
         DO 2 K5= 1.3
650
         k = 5 - k5
         KM1 = K - 1
560
670
         DO 2 J = 1.N
6d0
         Y(J,K) = Y(J,KM1)
690
       2 IF (K.LT.4) F(J.K)= F(J.KE1)
700
         DO 5 J=1.N
       3 Y(J,1) = YPRED(J) + C2 * TE(J)
710
         X = X + H
720
730
         PRED= .FALSE.
         HAMING = 1
740
750
         RETURN
         ..... CORRECTOR SECTION OF HAMING .....
700C
770
       4 DO 5 J= 1.N
         Y(J,1) = (9.*Y(J,2) - Y(J,4) + C4 * (F(J,1) + 2.*F(J,2) -
700
790
        1
                  F(J,j))/ J.
         TE(J) = C5 \neq (Y(J,1) - YPRED(J))
600
810
       5 Y(J,1) = Y(J,1) - TE(J)
020
         PRED= .TRUE.
         EAchlinG = 2
830
o40
         RETURN
650C
      THIS SECTION CALCULATES CONSTANTS USED IN HAMING AND MUST BE CALLED
9099
870C
      ONLY ONCE BEFORE THE ABOVE SECTION IS USED. IT NEED NOT BE CALLED
3006
      AGAIN UNLESS THE INTEGRATION STEP IS CHANGED.
890C
900
         ENTRY HAMNG1( N, Y, F, X, H, TE )
910
         C1 = 4.* H / 3.
         C2 = 112./ 9.
920
930
         C4 = 3.* H
940
         C5 = 9./121.
950
         HAMING = 0
960
         RETURN
970
         END
ready
```

×

0010# 1 1.0E-06 1.934E+04

•

ready

*

10∦0	.148E+	.903E-02	.0024E+	0.000E+	0.294E+	0.294E+ ().294E+ ().294E+
20#0	.294E+	0.294E+	0.033E+	0.019E+	0.012E+	0.000E+ '	1.309E+ (0.240E+
30#1	.106E+	0.900E+	0.327E+	0.120E+	0.553E+	0.450E+ (0.527E+	.U600E+
40%0	•552E+	0.2436+	0.055E+	0.092E+	1.000E+	1.000E+ '	1.000E+ '	1.000E+
50#1	.000E+	1.000E+	0.368E+	0.051E+	0.150E+	0.0236+ 1	1.000E+ 1	1.000E+
o0#1	.000E+	1.000E+	0.552E+	0.552E+	0.552E+	0.5526+ ().552E+ ().552E+
70	Ė+	-4.E+	-4.E+	-4.E+	-0.E+	-0.L+	-0.Ė+	-0. ± +
60	-0.E+	-0.E+	-4.c+	-4.E+	-4.E+	-4.E+	L+	6+
90	E+	Ľ+	Ľ +	上 +	E+	ビ +	E+	E+
100#	350.0	373.	0.0.	0 647.0	0.0	0.0	0,Ò	0.0
110#	0.0	0.	0 647.	0 373.0	0.0	644.0	0.0	0.0
120#	0.0	0.	0 143.	0 143.0) 322.0	36.0	143.0	143.0
130#	4.0	0.	ο ο.	0 18.0	0.0	0.0	0.0	0.0
140 <i>ii</i>	J.U	0.	ο.	0.0	0.0	0. Co	0.0	0.0
150 <i>#</i>	0.0	0.	ບ 10.	0 10.0	0 10.0	10.0	7.0	4.0
160 <i>#</i>	941.0	83.3	9 125.	0 290.6	ó 23 . 70	25.70	23.70	23.70
1'/0#	23.70	23.7	0 202.	9 102.24	i 125.0	(10.9	55.20	56.20
180#	125.2	20.0	0 277.	2 282.	7 624.1	09. δ	370.3	370.0
190#	945.	40.	125.	265.	11.7	11.7	11.7	11.7
200 <i>i</i> ł	11.7	11.7	256.	70.	125.	750.	35.2	30.2
210 <i>i</i> r	105.2	0.0	277.0	278.	625.0	70.0	377.0	310.
220	0.00	1007.0	000.	00.0	333. 0	.0		
230						•		
240								
250#	1.0 .738	1						
2ó0#	0.500E+	0.550E+	0.50dE+	- 0.588E+	0.500E+	0.500E+	(.190E+	
270#	9.600E+	1.700E+	3.120E+	- 6.620E+				
200+	1.0E+	+0.00£+	-03.3E+	+ 1.0E+	+0.00E+	-83.3±+		
290	4.00E+01	4.00E+0	1 4.00E+C	1 4.00E+0	1 9.77E+01	9.7/E+01	9.77E+01	9.77E+01
300	9.7/E+01	9.77E+0	1 9.31E+0	1 9.31E+0	1 9.31E+01	9.31E+01	4.00E+01	4.00E+01
310	4.00E+01	4.00E+0	1 4.00E+C	1 4.00E+0	1 4.00E+01	4.00E+01	4.00E+01	4.00E+01
320	4.502+01	4.50E+0	1 4.50E+C	1 4.50E+0	1 4.02E+01	4.02E+01	4.02E+01	4.02E+01
330	4.02E+01	4.02E+0	1 4.0 ₀ E+0	1 4.03E+0	1 4.03E+01	4.038+01	4.50E+01	4.50E+01
340					A 1	N - 0 - 0 -	1	1 600.01
210	4.50E+01	4.50些+0	1 4.50E+C)1 4.50E+0	1 4.50E+01	4.50E+01	4.508+01	4.505+01
350	4.508+01	4 . 50≟+0	1 4.50E+C	1 4.50E+0	1 4.50E+01	4.50E+01	4.501-01	4.506+01
350 360	4.50E+01	4 . 50≟+0	1 4.50E+C	1 4.50E+0	1 4.50E+01	4.50E+01	4.508+01	4.505+01
350 360 370	4.50£+01	4 . 50≟+0	1 4.50E+C	01 4.50E+0	1 4.50E+01	4.50E+01	4.502+01	4.502+01

ready

ź

390 400 410 420 430 440 1.00E+00 1.00E+00 1.0dE-06 2.57E-05 450 36.8E+00 30.8E+00 36.8E+00 36.8E+00 5.02E+01 5.02E+01 5.02E+01 5.02E+01 460 470 1.04E+02 1.04E+02 1.04E+02 1.04E+02 1.04E+02 1.04E+02 480 40.0E+00 40.0E+00 40.0E+00 40.0E+00 40.0E+00 40.0E+00 490 500 510 4.96E+00 4.96E+00 4.96E+00 4.96E+00 4.96E+00 4.96E+00 6.44E+00 35.2E+00 520 7.63E+00 9.30E+00 530 99.3E+00 99.3E+00 99.3E+00 99.3E+00 99.3E+00 99.3E+00 3.1/E+00 17.5E+00 540 5.75E+00 4.o2E+00 550 560 570 500 590 20.6E+00 1.13E+02 24.4E+00 30.0E+00 500 10.5E+00 90.0E+00 19.5E+00 24.0E+00 510 0.01 20.0 620#10 1 1 1.1 readv فغة 10 50 1 20 .0100 20.00 2.000 0.0 3.000 10 0 310. 298. 62.36 2.00 30 153.2 00.00 υ.0 0.95 2.00 •0 40 6.2117 0.0 00.0 1.2 0.00 1.00 50 0.0 1.00 2.00 2.00 .10 б. 60 10.0 0.050 0.025 0.025 0.050 0.050 70 .52 0.52 0.30 0.33 0.27 0.27 ່ຽວ 760.0 700.0 700.0 760.0 760.0 700.0 760.0 90 760.0 760.0 760.0 760.0 760.0 100# .2094 .1459 .1459 .1459 .1459 .1459 110# .1459 .1459 .1459 .1459 .1459 .1459 120# .0034 .0501 .0561 .0561 .0561 .0561 130# .0561 .0561 .0561 .0561 .0501 .0561 140# .7902 .7900 .7960 .7980 .7930 .7900 ready

157

*

150#	.7980	.7980	.7980	.7980	.7980	.7980	
100							
170							
100.	0.009	0.0	0.0	0.100	0.		0.0
190	0.300	0.0		0.790	0.		
200	0.300	0.0		0.790	0.		
210	0.275	-0.006	1.0	2.250	-0.005		2.0
220	0.275	-0.006	1.0	2.250	-0.006		2.0
230	2.080	-0.05	1.0	4.380	-0.05		3.0
240	2.030	-0.05	1.0	4.380	-0.05		3.0
250	0.730	-0.05	1.0	1.970	-0.05		3.0
260	0.730	-0.05	1.0	1.970	-0.05		3.0
270	0.700	-0.05	1.0	1.780	-0.05		3.0
280	0.700	-0.05	1.0	1.780	-0.05		3.0
290	0.046	0.44	1.90	0.070	3.50		_
300	0.046	0.44	1.90	0.070	3.50		
310	0.046	0.44	1.90	0.070	3.50		
320	0.046	0.44	1.90	0.070	3.50		
330	0.046	0.44	1.90	0.070	3.50		
340	0.046	0.44	1.90	0.070	3.50		
350	-0.140	1.4	4.38	0.50	8.00		
360	-0.140	1.0	2.45	0.30	-6.00		
370							
380							
390							
400							
410-90.0		90.0	0.00	5			

.

.

.

ready

¥

158

REFERENCES

- (1) Noordergraaf, A., Jager, G. N., and Westerhof, N. <u>Circulatory Analog Computers</u>. North-Holland Publishing <u>Co.</u>, Amsterdam, 1963.
- Reeve, E. G. and Guyton, A. C. <u>Physical Bases of</u> <u>Circulatory Transport: Regulation and Exchange.</u> W. B. Saunders Co., Philadelphia, Pennsylvania, 1967.
- (3) Beneken, J.E.W. Investigation of the regulatory system of the blood circulation, <u>Circulatory Analog Computers</u>, pp. 16-28, Ed. by Noordergraaf, A., et al., North-Holland Publishing Co., Amsterdam, 1963.
- (4) Noordergraaf, A. Boom, H.B.K., and Verdouw, P. D. A human systemic circulatory analog computer, <u>Proc.</u> First Congr. Soc. B.C.G. Res., Zeist, 1960.
- (5) Noordergraaf, A. Verdouw, P. D., and Boom, H.B.K. The use of an analog computer in a circulation model, Progr. Cardiovase. Dis. 5: 419-439, 1963.
- (6) Noordergraaf, A. Development of an analog computer of the human circulatory system. In <u>Circulatory Analog</u> <u>Computers</u>, Ed. by Noordergraaf, Amsterdam, North Hall, Publ, 1963.
- (7) Noordergraaf, A., Verdouw, P. D., van Brummelen, A.G.W., and Wiegel, F. W. Analog of the arterial bed, <u>Pusatile</u> <u>Blood Flow</u>, Ed. by Attinger, E. O., McGraw-Hill, <u>New York</u>, 1964.
- (8) Robinson, D. A. Ventricular dynamics and the cardiac representation problem, <u>Circulatory Analog Computers</u> pp. 56-81, Ed. by Noordergraaf, A., et al., North-Holland Publishing Co., Amsterdam, 1963.
- (9) Defores, J. G., Hara, H. H., Osborn, J. J., and McLeod, J. Theoretical analysis and computer simulation of the circulation with special reference to the Starling properties of ventricles, <u>Circulatory Analog Computers</u>, pp. 91-122, Ed by Noordergraaf, A., et al., North-Holland Publishing Co., Amsterdam, 1963.
- (10) Beneken, J.E. W. Electronic analog computer model of the human blood circulation. In <u>Pulsatile Blood Flow</u>. Ed. by E. O. Attinger, New York, McGraw-Hill, 1964, pp. 423-432.

- (11) Beneken, J.E.W., and Grupping J.C. M. Electronic analog compouter model of the human circulatory system. In <u>Medical Electronics</u>. Ed by F. H. Bostem, Bruxelles, Desoer, 1965.
- (12) Beneken, J.E.W. and DeWitt B. A physical approach to hemodynamic aspects of the human cardiovascular system. In <u>Physical Bases of Circulatory Transport:</u> <u>Regulation and Exchange</u>. Ed. by Reeve, E. G. and <u>Guyton, Arthur C. W. B. Saunders, 1967, pp. 1-45.</u>
- (13) Grodins, F. S. and Buoncristiani, J. F. General formulation of the cardiovascular control problemmathematical models of the mechanical system, <u>Physical</u> <u>Bases of Circulatory Transport: Regulation and Exchange</u> pp. 61-76. Ed. by Reeve, E. B., et al., W. B. Saunders Co., Philadelphia, Penn., 1967.
- (14) Hwang, N.H.C. and Chao, J. C. An integral form of the hydraulic transmission line equation as applied to circulatory systems, <u>Ninth Annual Sym. on Biomath.</u> Comp. Sci. in Life Series, 1971.
- (15) Croston, R. C., Rummel, J. A. and Kay, F. J. Computer model of cardiovascular control system responses to exercise, <u>Journal of Dynamic Systems</u>, <u>Measurement</u>, and Controls, <u>pp. 301-307</u>, 1973.
- (16) Attinger, E., Anne, A., Mikami, T., and Sugawara, H. Modeling of pressure-flow relations in arteries and veins, <u>Hemerheology</u>, Ed. by A. L. Copley, Pergamon Press, 1968.
- (17) Boyers, D. G., Cuthbertson, J. G., and Luetscher, J. A. Simulation of the human cardiovascular system: a model with normal responses to change of posture, blood loss, transfusion, and autonomic blockage, <u>Simulation</u>, 18:197-205, 1972.
- (18) Green, J. F. and Miller, N. C. A model describing the response of the circulatory system to acceleration stress, <u>Annals of Biomedical Engineering</u>, 1:455 - 467, 1973.
- (19) Grodins, F. S., Buell, J., and Bart, A. J. Mathematical analysis and digital simulation of the respiratory control system, <u>J. Appl. Physiol</u>., 22(2):260-276, 1967.

- (20) Energy B., and Pack, A. I. An experimentally verified model of the gas exchanging properties of the lung. <u>Conference on Computers for Analysis and Control in</u> <u>Med. and Biological Research, pp. 130-134, 1971.</u>
- (21) Saidel, G. M., Militano, T. C., and Chester, E. H. Pulmonary gas transport characterized by a dynamic model, Resp. Physiol., 12:305 (1971).
- (22) Saidel, G. M., Militano, T. C., and Chester, E. H. Mass-balance model of pulmonary oxygen transport, IEEETrans., (BioMed. Engr.) ME-19: 205 (1972).
- (23) Dick, D. E. and Rideout, V. C. Analog simulation of left heart and arterial dynamics, <u>Proc. 18th ACEMB</u>, Philadelphia, 1965.
- (24) Skalak, R. Wave propagation in blood flow, <u>Biomechanics</u>, Ed. by Y. C. Fung, ASME, 20-46, 1966.
- (25) Gessner, U.R.S. Wave propagation in Arteries with nonlinear distensibility, <u>Circulatory Analog Computers</u>, pp. 82-90, Ed. by Noordergraaf, A., et al., North-Holland Publishing Co., Amsterdam, 1963.
- (26) Cope, F. W. A modified Windkessle (elastic reservoir) theory of the human systemic arterial system using modern data on aortic elasticity so as to yield computational accuracy sufficient for clinical use, <u>Circulatory Analog Compouters</u>, pp. 45-55, Ed. by Noordergraaf, A., et al., North-Holland Pub. Co., Amsterdam, 1963.
- (27) Gaasch, W. H., Battle, W. D. Oboler, A. A., Banas, J.S., and Levine, H. J. Left ventricular streess and compliance in man, Circulation, XLV:746-762, 1972.
- (28) Glantz, S. A. Ventricular pressure-voluem curve indices change with end-diostolic pressure, <u>Circulation</u> <u>Research</u>, 39:772-778, 1976.
- (29) Carnahan, B., Luther, H.A. and Wilkes, J. O. <u>Applied</u> Numerical Methods, John Wiley & Sons, Inc., 1969.
- (30) Sommerfeld, A. <u>Mechanics of Deformable Bodies</u>. Academic Press, <u>New York</u>, 1950.
- (31) Perry, R. H. and Chilton C. H. <u>Chemical Engineers</u> <u>Handbook</u>. McGraw-Hill, New York, 1973.

V

Y

- Ruch, T. C. and Patton, H. D. <u>Physiology and</u> <u>Biophysics</u>. W. B. Saunders Co., Philadelphia, Pennsylvania, 1974.
- (33) Dahn, I., Jonson, B., and Nilson, R. Plethysmographic in vivo determination of the elastic properties of arteries in man, J. Appl. Physiol., 28:328-332, 1970.
- (34) Morris, T. W., Abbrecht, P. H., and Leverett, S. D. Diameter-Pressure Relationships in the Unexposed Femoral Vein, Am. J. Physiol., 227:782-788, 1974.
- (35) Borst, H. G., McGregor, M., Whittengerger, J. L., and Berglund, E. Influence of pulmonary arterial and left atrial pressures on pulmonary vascular resistance, Circulations Res.: 393-399, 1956.
- (36) Patel, D. J., Greenfield, J. C., and Fry, D. L. in Vivo pressure length-radius relationship of certain blood vessels in man and dog, <u>Pulsatile</u> <u>Blood Flow</u>, pp. 293-305, Ed. by Attinger, E. O., <u>McGraw-Hill</u>, N. Y., 1964.
- (37) Payne, R. M., Stone, H. L., and Engelken E. J. Atrial Function During Volume Loading, <u>J. Appl.</u> Physiol., 31:326-331, 1971.
- (38) Bergel, E. H. <u>Pulsatile Blood Flow</u>. pp. 275-292, Ed. by Attinger, E. O., McGraw-Hill, N.Y., 1964.
- (39) Rashevsky, N. <u>Mathematical Biophysics</u>. The Univ. of Chicago Press, Chicago, Illinois, 1948.
- (40) Ganong, W. F. <u>Review of Medical Physiology</u>. LANGE Medical Publications, Los Altos, Ca., 1973.
- (41) Bailey, H. R. Oxygen exchange between capillary and tissue: some equations describing countercurrent and nonlinear transport, <u>Physical Bases of Circulatory</u> <u>Transport: Regulation and Exchange, Ed. by Reeve,</u> <u>E. B. et al., pp. 353-366, W. B. Saunders Co.,</u> <u>Philadelphia, Penn, 1967.</u>
- (42) Gomez, D. M. Considerations of Oxygen-Hemoglobin Equilibrium in the Physiological State, <u>Am. J.</u> <u>Physiol.</u>, <u>49</u>: 105-124, 1965.
- (43) Kelman, G. R. Digital computer procedure for the conversion of P_{CO} into Blood CO₂ Content, <u>Respir.</u> <u>Physiol.</u>, <u>3</u>: 111-115, 1967.
- (44) West, J. G. <u>Bioengineering Aspects of the Lung</u>. Marcel Dekker, Inc., N.Y., 1977.

- (45) Dill, D. B., Edwards, H. T., and Consolazio, W. V. Blood as a Physiochemical System: XI. Man at Rest., J. Biol. Chem., 118: 635-648, 1937.
- (46) Davenport, H. W. <u>The ABC of Acid-Base Chemistry</u>. The Univ. of Chicago Press, Chicago, 11., 1969.
- (47) Gordon B. L., et al., ed. <u>Clinical Cardiopulmonary</u> Physiology. Grune & Sratton, Inc., N.Y., 1969.
- (48) Knapp, C. F. Response to the cardiovascular system to vibration and combined stress, <u>Progress Report to</u> Air Force Office of Scientific Research, 1977.
- (49) Guyton, A. C. <u>Medical Physiology</u>. W. B. Saunders Co., London, 1971.
- (50) Dill, D. B., Edwards, H. T., and Consolazio, W. V. Blood as a physicochemical system: XI. Man at rest, J. Biol. Chem., 118: 635-648, 1937.
- (51) Davenport, Horace W. <u>The ABC of Acid-Base Chemistry</u>. The University of Chicago Press, Chicago, 1969.