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 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 design. It must incorporate gas transport and a pumping heart. It must be complete with verifiable nonlinearities and parameters with physiological correspondences. It 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 aid. This work reached a peak in 1962 when Circulatory 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			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	x	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	x	x		g-stress of 1 g_z; 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 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 pressure-volume 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;

$$\frac{dV}{dP} = K_0(1+a\overline{P}+b\overline{P}^2+c\overline{P}^3) \qquad (1)$$

V=blood volume

K_o, 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

$$r = a(P-b)^c+d \text{ (for low pressures)} \qquad (2)$$

$$r = t\log P+g \quad \text{(for high pressures)}$$

where r = vessel radius

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]

$$\frac{V_m}{N} = \frac{T_s}{R}(P_s-P_v)+a(P_s^2-P_d^2)+b(P_s-Pd) \qquad (3)$$

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 valves 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 vaso-constriction 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 only). The program is able to include any number of gases 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

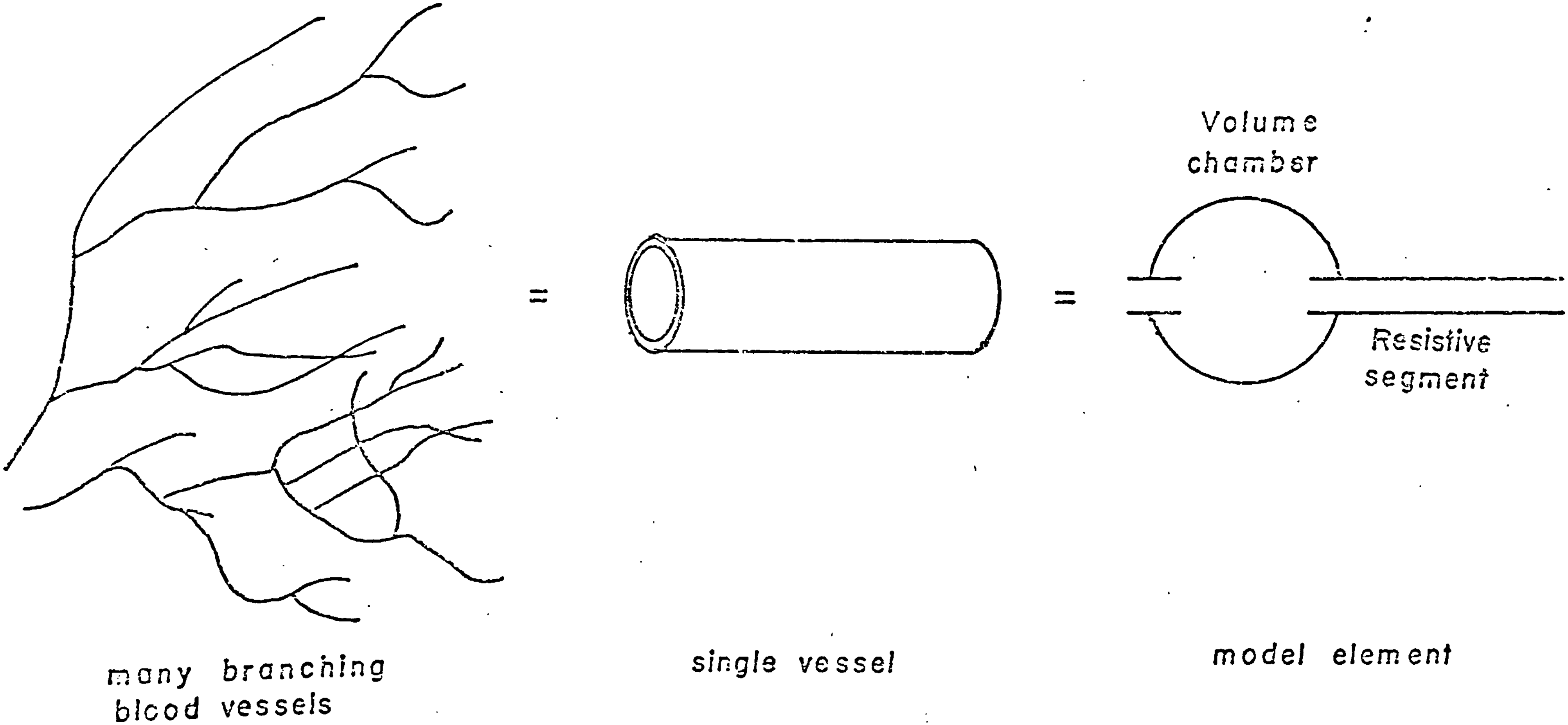
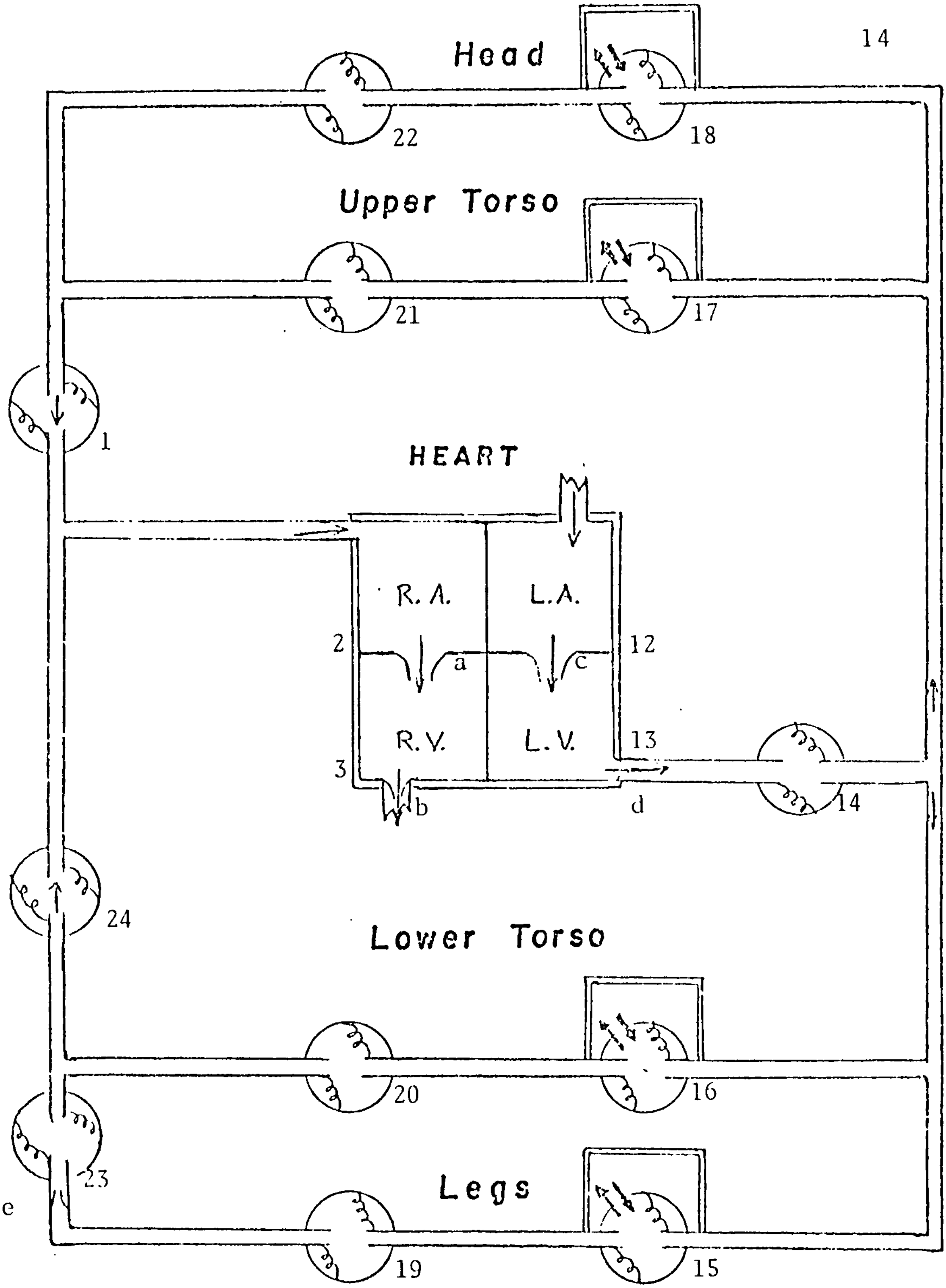
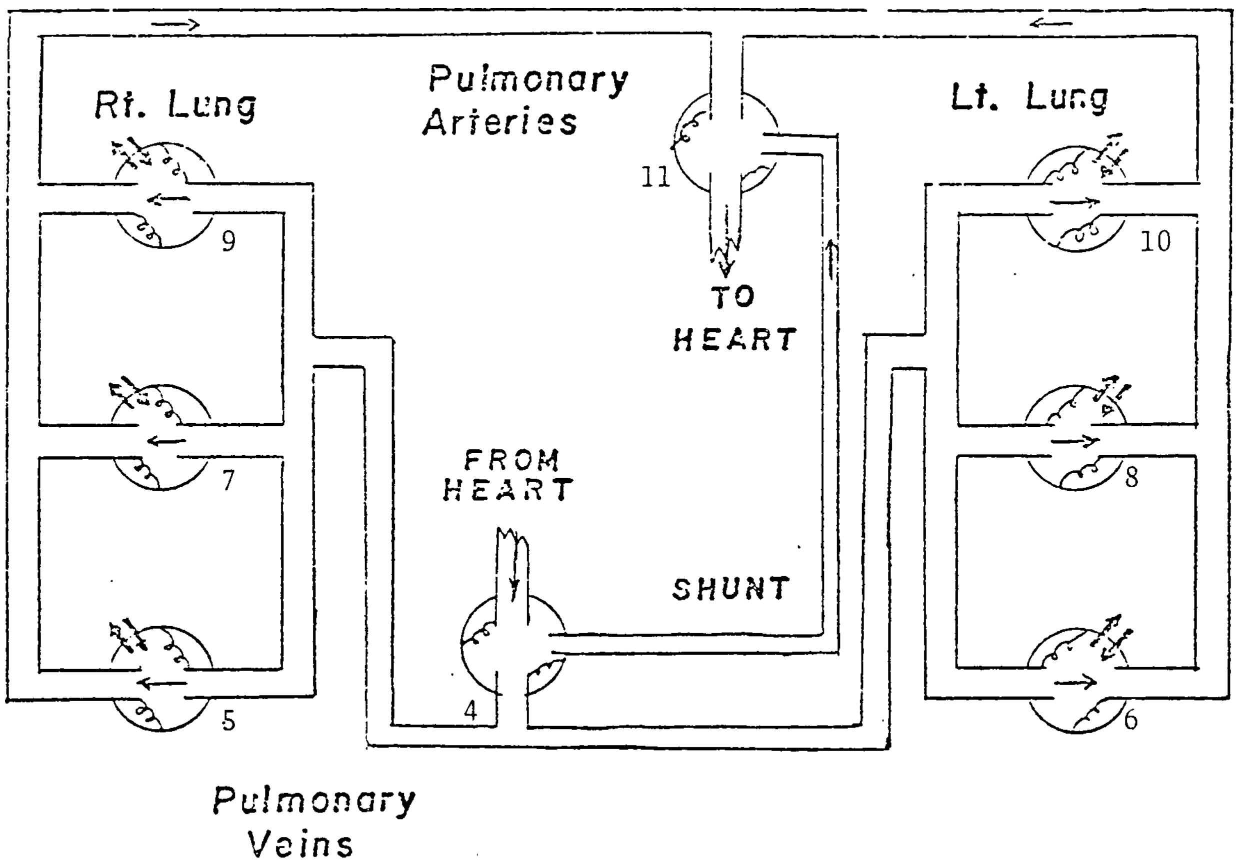

many branching blood vessels
=
single vessel
=
Volume chamber
Resistive segment
model element

FIGURE 1



SYSTEMIC CIRCULATION
FIGURE 2



PULMONARY CIRCULATION

FIGURE 3

Key to Chamber Numbers

1	Large veins including superior vena cava
2	Right Atrium
3	Right Ventricle
2	Left Atrium
3	Left Ventricle
4	Pulmonary arteries
5-10	Pulmonary capillaries
11	Pulmonary veins
12	Right Atrium
13	Right Ventricle
14	Systemic arteries
15-18	Systemic capillaries
19-22	Small veins
23	Veins of the leg
24	Large veins including inferior vena cava

The valves are:

(a) Tricuspid valve

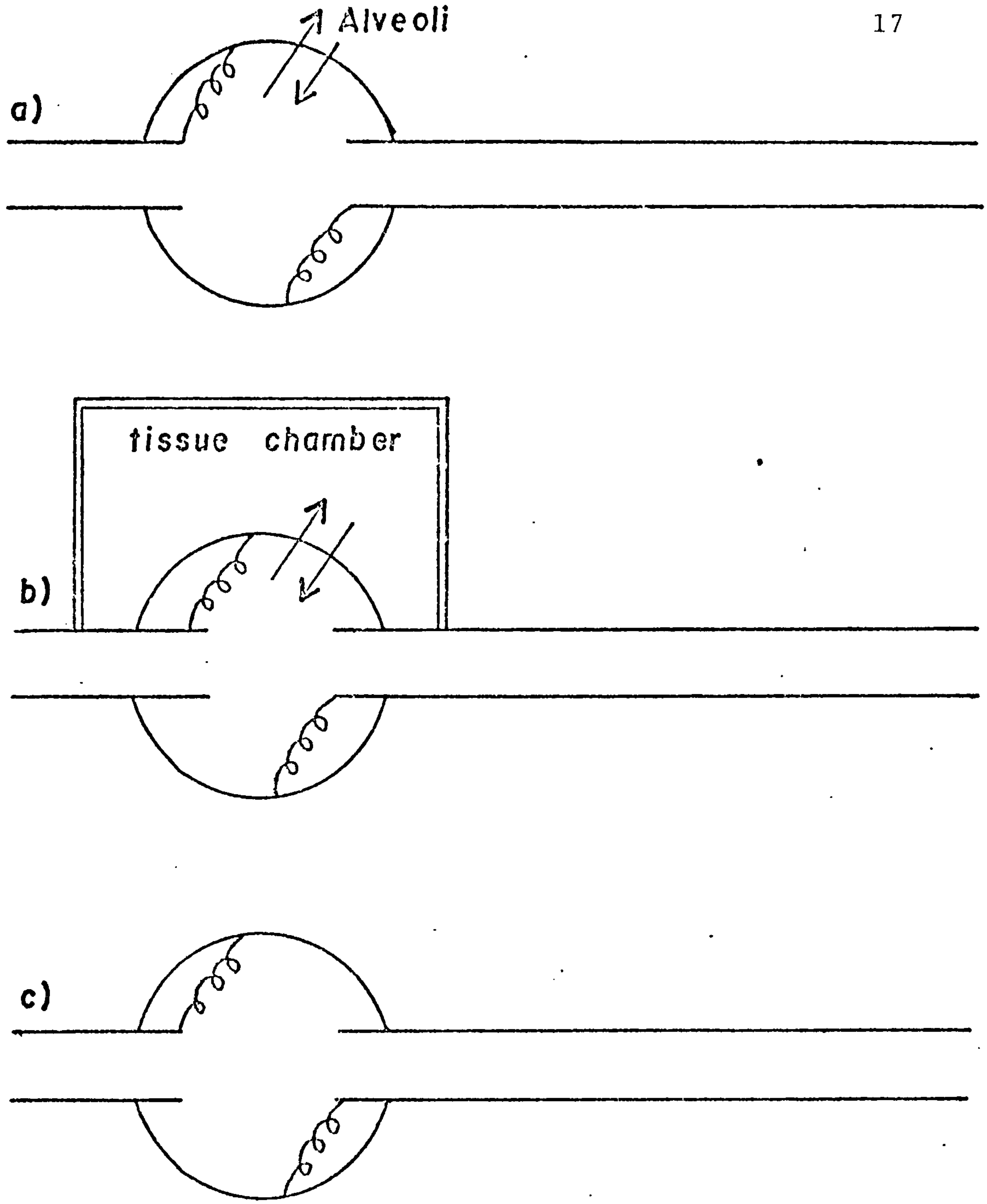
(b) Pulmonary valve

(c) Mitral valve

(d) Aortic valve

(e) Leg valve

Figure 4



TYPES OF MODEL ELEMENTS

FIGURE 5

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

$$v_{19} \neq 0 \text{ if } P_{20} > P_{19}, \text{ valve open} \qquad (5)$$
$$v_{19} = 0 \text{ if } P_{20} \leq P_{19}, \text{ valve closed}$$

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 valves.

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;

$$v^{in} - v^{out} = \frac{dV}{dt} \qquad (6)$$

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

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

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 $(\vec{f}_{visc}-\rho\vec{\Omega}X\vec{r})$ can be written as $\bar{f}v^2$, where $\bar{f}$ 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

$$\rho\frac{\partial r_z}{\partial t}+\frac{\rho}{2}\frac{\partial}{\partial z}(r_x^2+r_y^2+r_z^2) = -\frac{\partial\underline{P}}{\partial z}-\rho\frac{\partial\phi}{\partial z}+\bar{f}v_z \qquad (9)$$

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

$$\frac{\partial r_x}{\partial z} = \frac{\partial r_y}{\partial z} = 0 \qquad (10)$$

$$\rho\frac{\partial r_z}{\partial t} = -\frac{\rho}{2}\frac{\partial r_z}{\partial z} - \frac{\partial P}{\partial z} - \rho\frac{\partial \phi}{\partial z} + \overline{f}r_z^2 \qquad (11)$$

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

$$\phi = g_z \qquad (12)$$

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;

$$r_{z_1} = r_{z_2} \qquad (13)$$

and Equation 11 becomes

$$(\rho\Delta z)\frac{\partial r_z}{\partial t} = (P_1+\rho gh_1)-(P_2+\rho gh_2)+Rr_z+R'r_z^2 \qquad (14)$$

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:

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

where

$\bar{R}_1$ = flow resistance in the segment connecting chambers 1 and 2

r_2 = velocity out of chamber 1 (cm/sec)

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

$$Ar_z = v \tag{18}$$

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

$$\frac{dV}{dt} = v^{in} - v^{out}, \tag{19}$$

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

and

$v_i(v_j)$ = volumetric flow rate into (out of) the chamber through segment i (j)

V = blood volume of the chamber

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,

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

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

P' = external pressure on the vessel and surrounding tissue

$\tilde{P}$ = elastance pressure of the muscular tissue surrounding vessel (function of vessel radius)

Rv_r = resistance of the vessel wall to transverse motion

R = resistive parameter

v = rate of change of the vessel volume with respect to time

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

An additional assumption of instantaneous equilibrium was made by giving

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

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

$$P = P' + \tilde{P}. \tag{23}$$

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

$$v = \frac{dV}{dt} \tag{24}$$

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 any circulatory element can be fitted very accurately by an equation of the form;

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

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 one 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

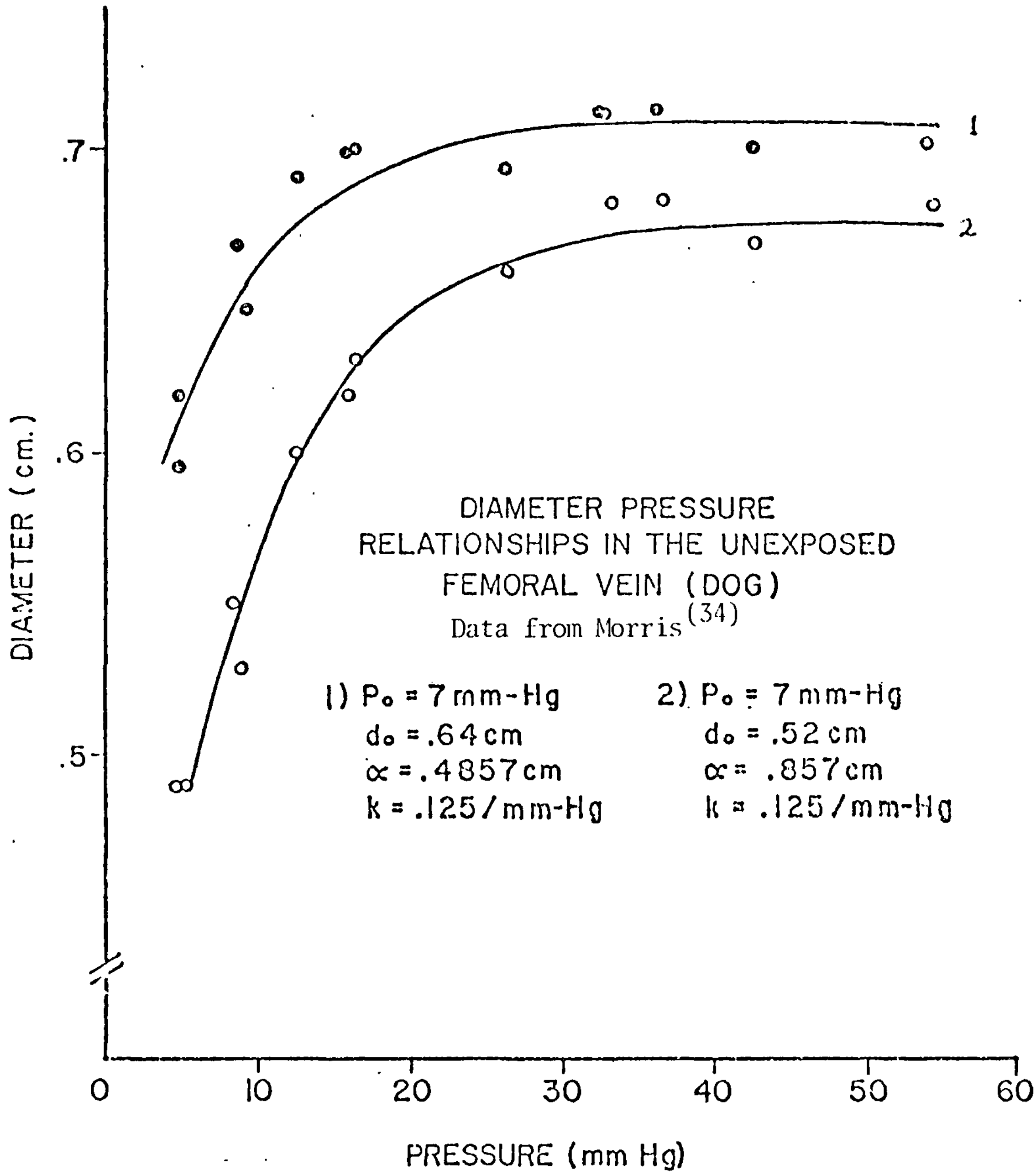
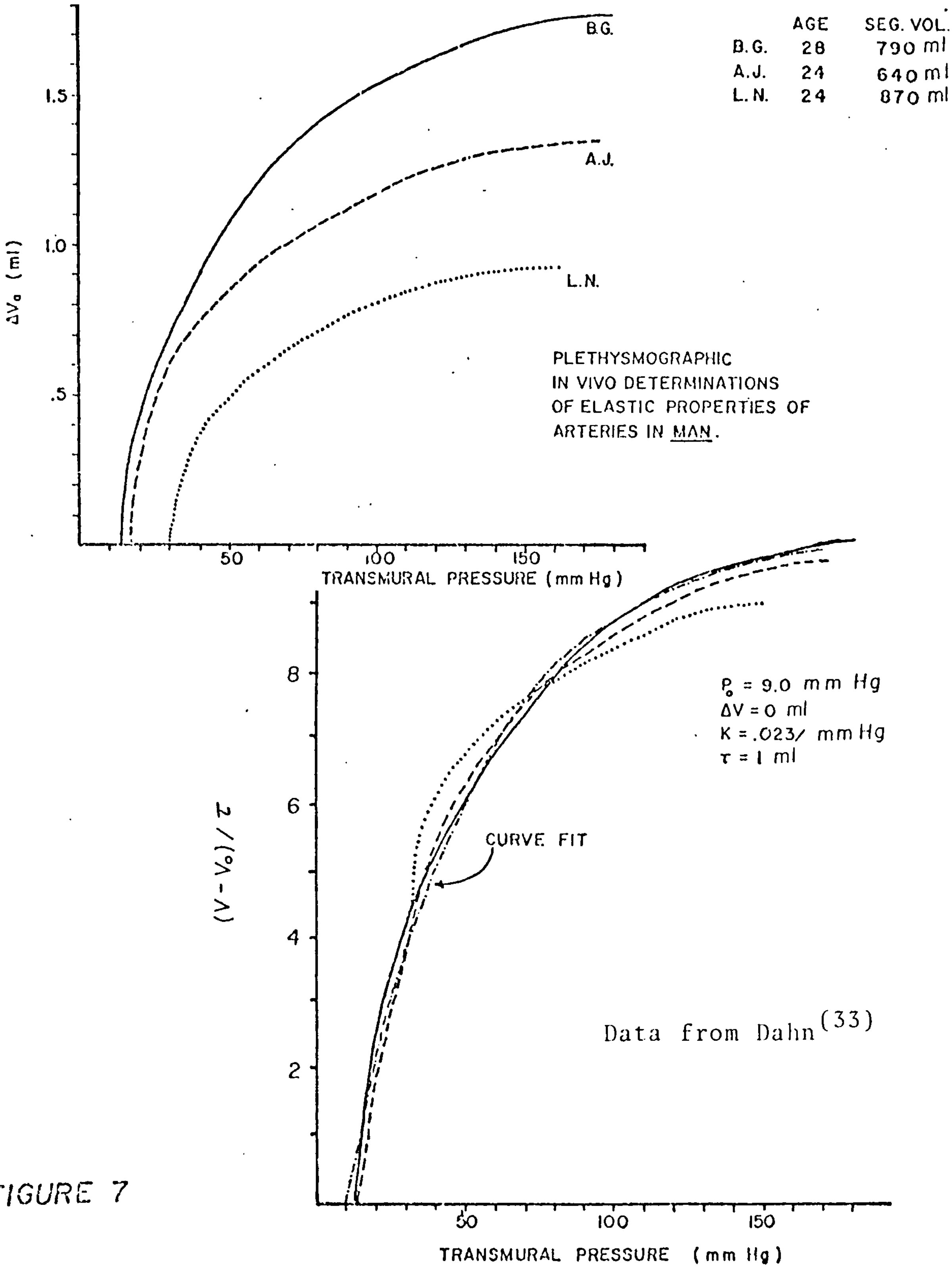

DIAMETER PRESSURE
RELATIONSHIPS IN THE UNEXPOSED
FEMORAL VEIN (DOG)
Data from Morris(34)
1) Po = 7 mm-Hg
do = .64 cm
α = .4857 cm
k = .125/mm-Hg
2) Po = 7 mm-Hg
do = .52 cm
α = .857 cm
k = .125/mm-Hg
1
2
.7
.6
.5
DIAMETER (cm.)
0
10
20
30
40
50
60
PRESSURE (mm Hg)

FIGURE 6



AGE SEG. VOL.
B.G. 28 790 ml
A.J. 24 640 ml
L.N. 24 870 ml
B.G.
A.J.
L.N.
1.5
1.0
.5
ΔVa (ml)
50
100
150
TRANSMURAL PRESSURE (mm Hg)
PLETHYSMOGRAPHIC
IN VIVO DETERMINATIONS
OF ELASTIC PROPERTIES OF
ARTERIES IN MAN.
8
6
4
2
(V-Vo)/τ
P₀ = 9.0 mm Hg
ΔV = 0 ml
K = .023/ mm Hg
τ = 1 ml
CURVE FIT
Data from Dahn(33)
50
100
150
TRANSMURAL PRESSURE (mm Hg)

FIGURE 7

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)

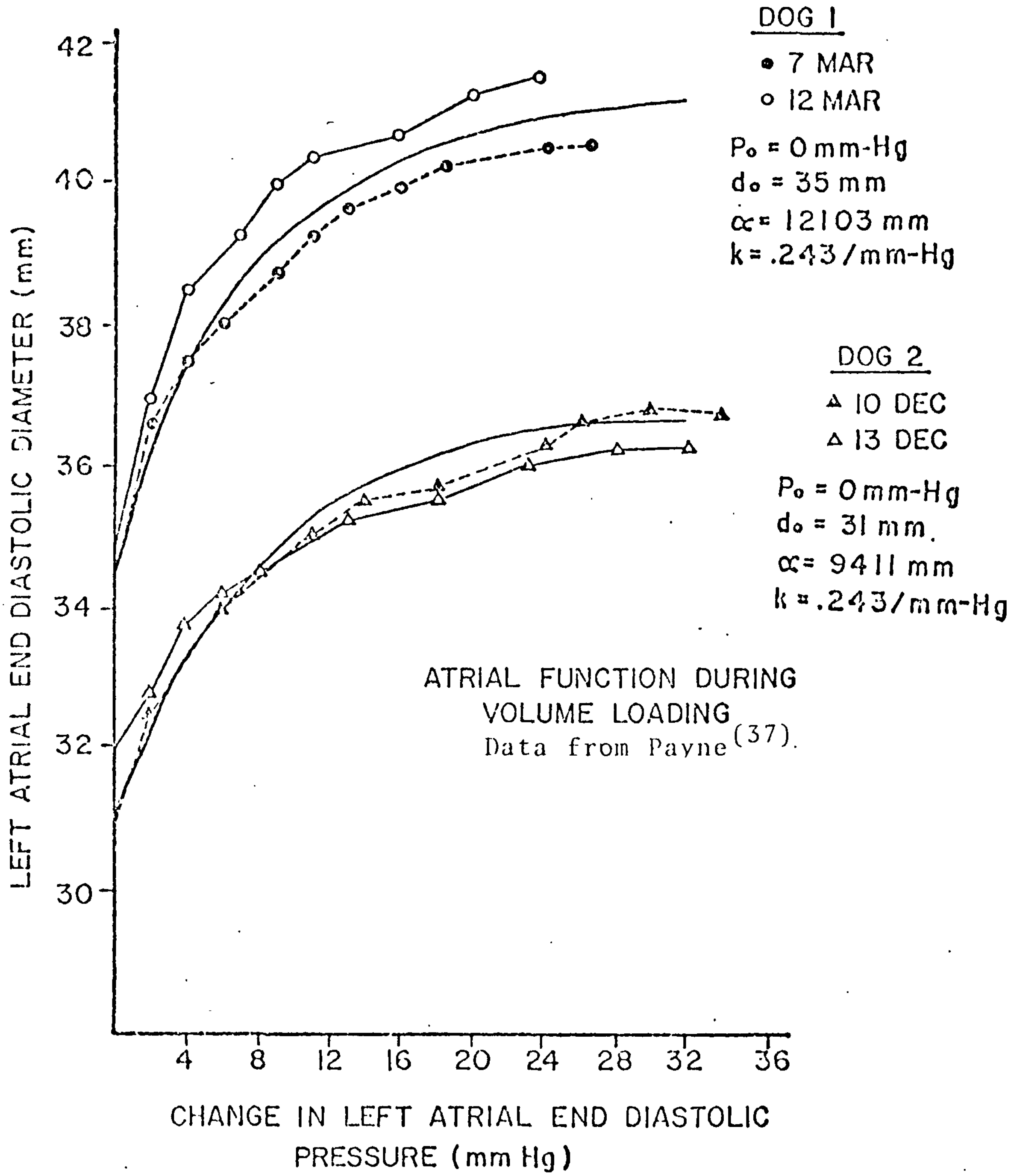
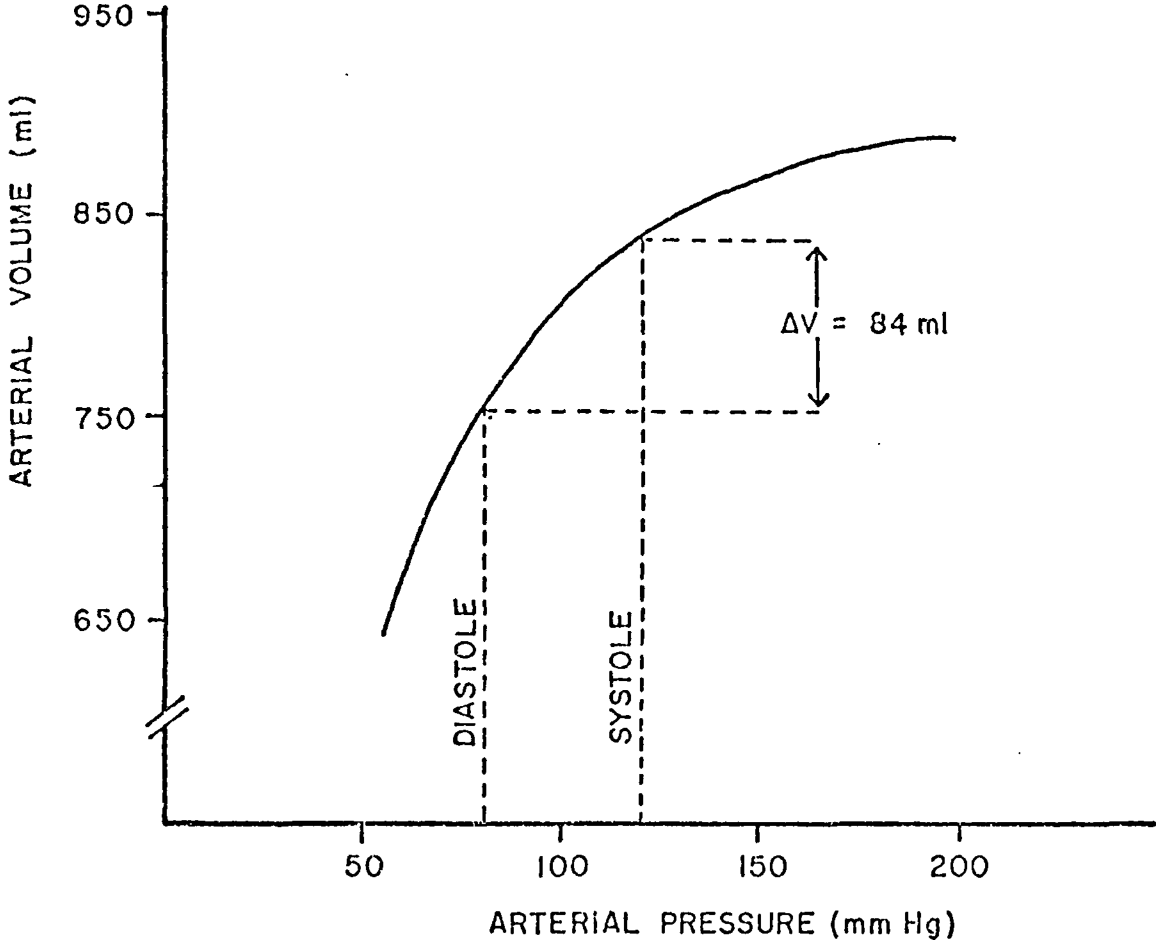

DOG I
• 7 MAR
○ 12 MAR
Po = 0 mm-Hg
do = 35 mm
α = 12103 mm
k = .243/mm-Hg
DOG 2
▲ 10 DEC
△ 13 DEC
Po = 0 mm-Hg
do = 31 mm.
α = 9411 mm
k = .243/mm-Hg
ATRIAL FUNCTION DURING
VOLUME LOADING
Data from Payne(37).
LEFT ATRIAL END DIASTOLIC DIAMETER (mm)
42
40
38
36
34
32
30
4
8
12
16
20
24
28
32
36
CHANGE IN LEFT ATRIAL END DIASTOLIC
PRESSURE (mm Hg)

FIGURE 8



DERIVED COMPLIANCE CURVE OF HUMAN ARTERIAL CHAMBER

FIGURE 9

made note of this scaling by plotting arterial pressure-volume 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 pressure-volume 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

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

with λ' a constant. Assuming cylindrical vessel geometry leads to

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

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.

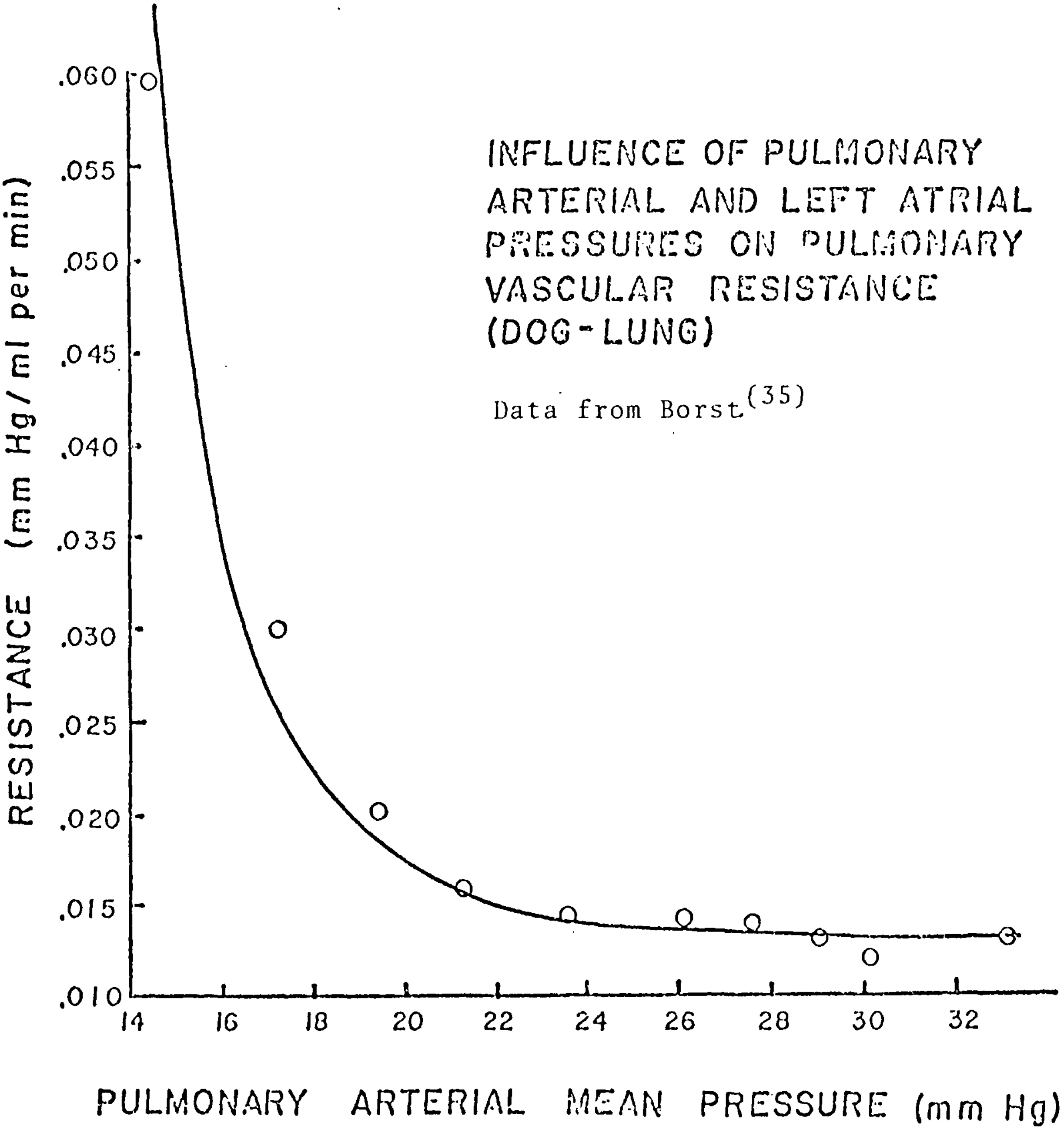

INFLUENCE OF PULMONARY ARTERIAL AND LEFT ATRIAL PRESSURES ON PULMONARY VASCULAR RESISTANCE (DOG-LUNG)
Data from Borst(35)
RESISTANCE (mm Hg / ml per min)
.060
.055
.050
.045
.040
.035
.030
.025
.020
.015
.010
14
16
18
20
22
24
26
28
30
32
PULMONARY ARTERIAL MEAN PRESSURE (mm Hg)

FIGURE 10

Fixing Resistance Parameters

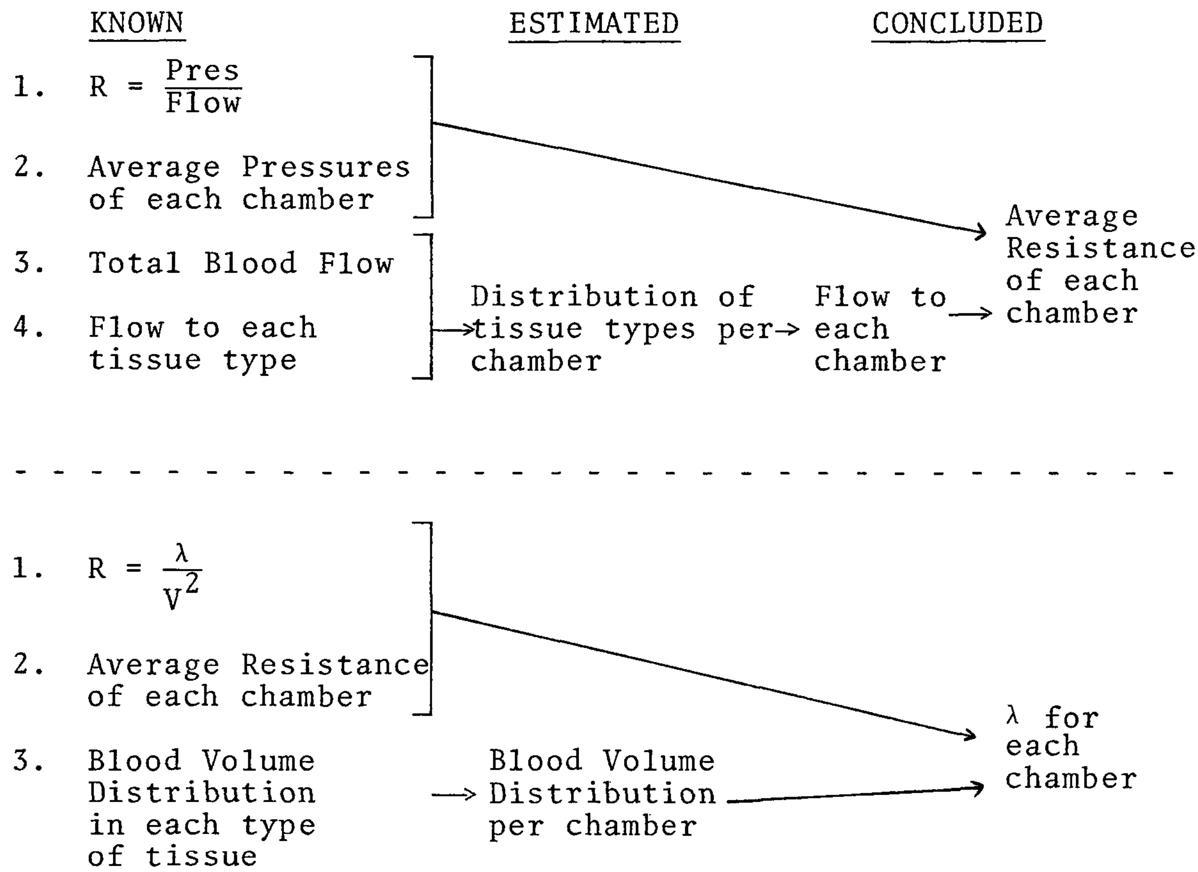


Figure 11

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

$$(\rho\ \Delta\ z)\frac{\partial}{\partial t}\left(\frac{v_1}{A_1}\right) = (P_1+\rho g h_1) - (P_2+\rho g h_2)+\bar{R}_1 v_1+\bar{R}_1' v_1^2 \qquad (28)$$

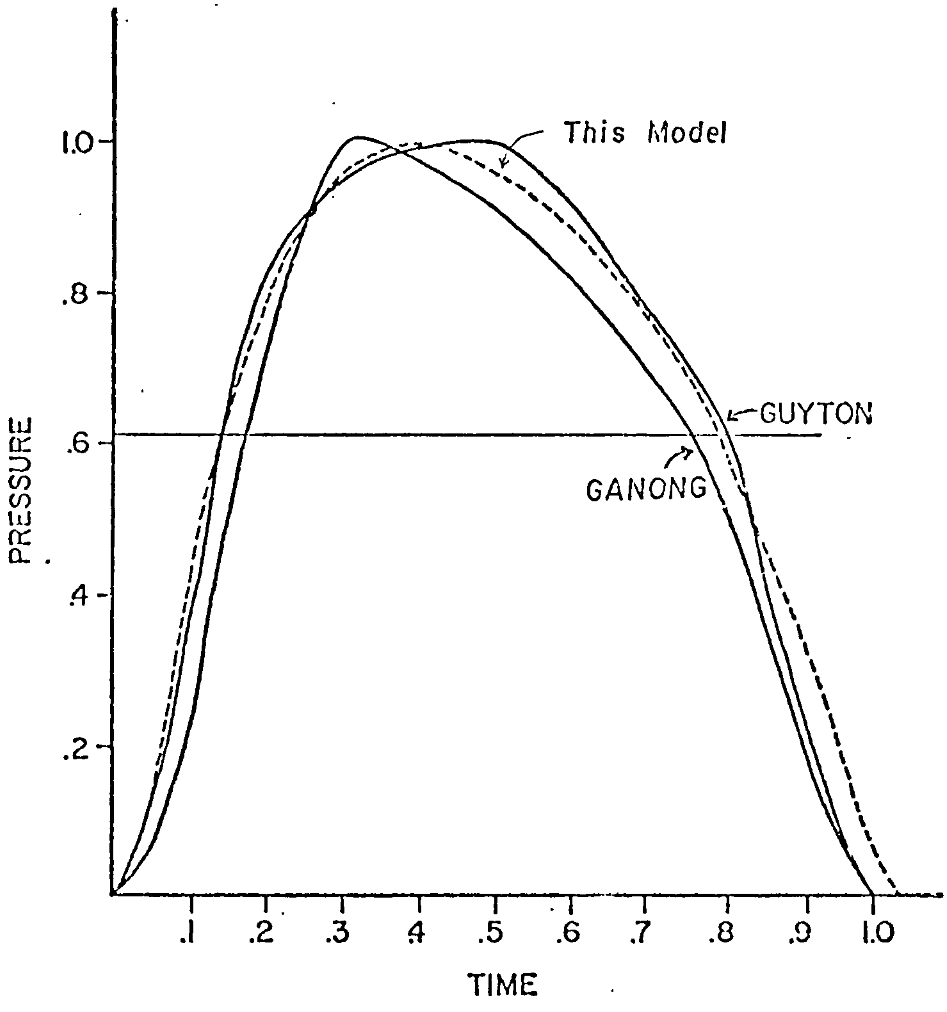
where

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

$P_1(P_2)$ = ventricular (arterial) pressure

$h_1(h_2)$ = height above the center of the heart of the ventricular (arterial) chamber

$$R_1 = \bar{R}_1/A_1 \qquad (29)$$



VENTRICULAR PRESSURE
PULSE

FIGURE 12

$$R_1' = \overline{R}_1'/A_1 \tag{30}$$

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,

$$v_1 = \frac{dV_1}{dt} \tag{31}$$

where V_1 = ventricular volume in ml

In this model, it is assumed that $h_1 = h_2$, simulating a prone subject, giving

$$L_1\frac{d_2V_1}{dt^2} = P_1-P_2+R_1\frac{dV_1}{dt}+R_1'(\frac{dV_1}{dt})^2 \tag{32}$$

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:

$$P_2-P_3 = R_2v_2 \tag{33}$$

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,

$$L_2\frac{d^2V_2}{dt^2} = P_2'+\tilde{P}_2-P_2+R_2\frac{dV_2}{dt} \quad (34)$$

where $L = \frac{M}{A^2}$

P_2' = external pressure

$\tilde{P}_2$ = 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

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

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

and $$v_3 = \frac{dV_1}{dt}-\frac{dV_2}{dt} \quad (37)$$

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

$P_1(t)$ = ventricular pressure simulated as shown in Figure 12 (equation and constants are given in Appendix 1).

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

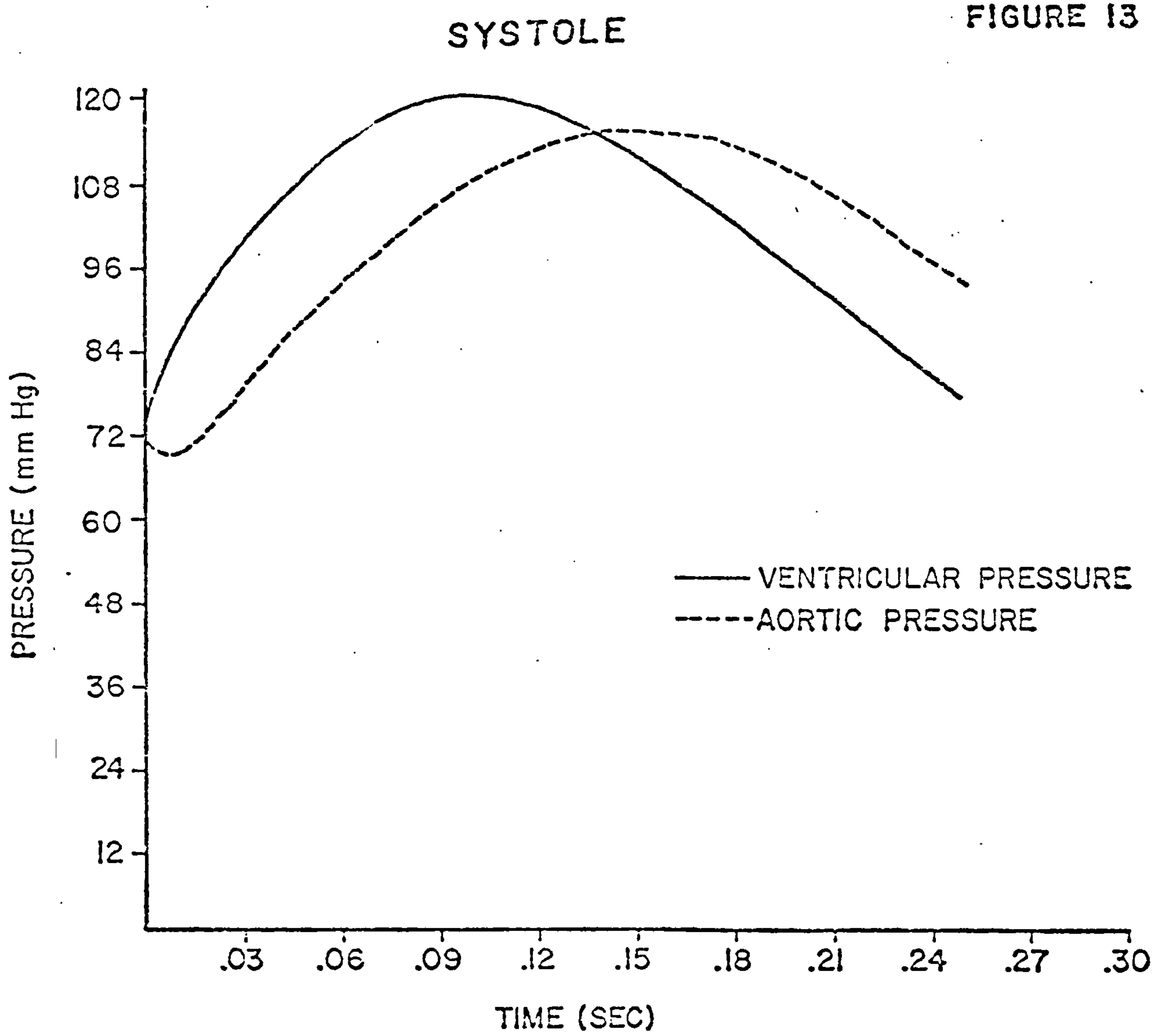
$v_2(t=0)$ = initial volume of atrium, from Guyton.[48]

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).

FIGURE 13


SYSTOLE
120
108
96
84
72
60
48
36
24
12
PRESSURE (mm Hg)
.03
.06
.09
.12
.15
.18
.21
.24
.27
.30
TIME (SEC)
VENTRICULAR PRESSURE
AORTIC PRESSURE

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.

$$L_i \frac{d^2V_i}{dt^2} = PC_i - PC_{i+1} + R_i' v_i^2 - R_i \ v_i \qquad (38)$$

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

PC_i = $P_i + \rho g h_i$.

R_i = resistance of blood to longitudinal motion of ventricle wall.

V_i = volume of ventricle.

P_i = pressure in chamber.

$\rho g h_i$ = hydrostatic pressure.

R_i' = turbulence constant.

L_i = inertia of blood exiting ventricle.

b. Atrial valve open, ventricular valve closed.

$$R_i \frac{dV_i}{dt} = PC_i - PC_{i-1} \qquad (39)$$

where R_i = resistance of blood flow into ventricle.

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

$PC_i = P_i + \rho g h_i$.

P_i = pressure in chamber.

$\rho g h_i$ = hydrostatic pressure.

c. Both valves closed.

$$P = P' + \tilde{P} \tag{40}$$

where P' = external ventricular pressure.

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

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

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

for the equation of blood flow between chambers.

$$P_i' + \tilde{P}_i - P_i = R_i^{in} \frac{dV_i}{dt}$$

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.

$\rho g h_i$ = 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_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

$[Cv]^{in}$ ($[Cv]^{out}$) = transfer rate of gas into (out of) chamber by blood flow (μ moles/sec)

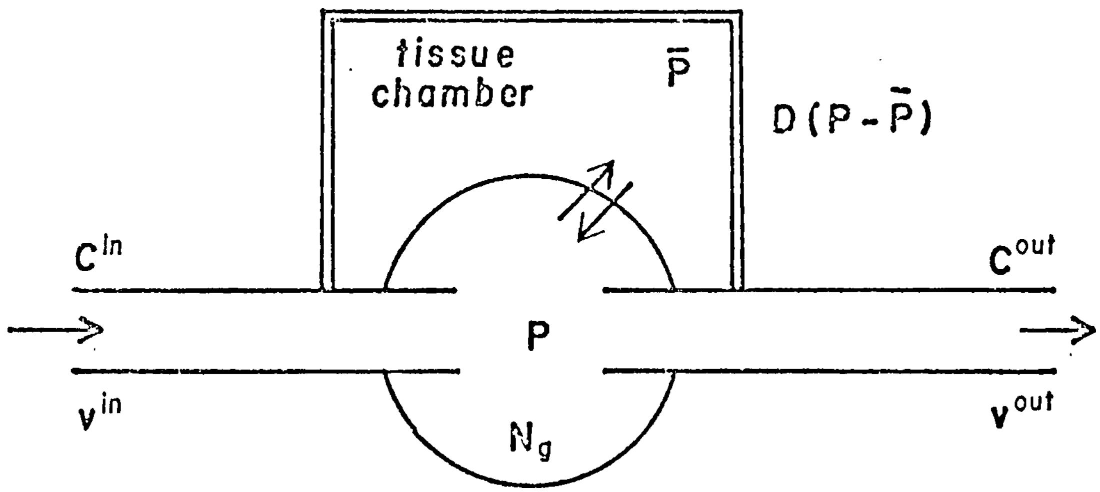
D = the diffusion coefficient in μ moles/sec·mmHg

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:

$$D = \frac{\alpha A S}{\ell} \qquad (43)$$

Conservation of mass of gas in vessel chambers



$$\frac{dN_g}{dt} = C^{in} v^{in} - C^{out} v^{out} - D(P - \bar{P})$$

FIGURE 14

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;

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

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:

$$N_{O_2}^{plasma} = \alpha_{O_2}(1-H)VP_{O_2} \qquad (45)$$

where $N_{O_2}^{plasma}$ = number of moles of O_2 in the plasma

H = hematocrit = $\dfrac{V_{RBC}}{V_{plasma}+V_{RBC}}$

V_{RBC} = volume of red blood cells

V_{plasma} = volume of plasma

P_{O_2} = partial pressure of O_2

α_{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 O_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

$$N_{O_2}^{hemoglobin} = S_{O_2} \beta V \qquad (46)$$

where $N_{O_2}^{hemoglobin}$ = number of moles of O_2 bound to hemoglobin in a chamber of blood volume V

β = 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},

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

and for the concentration of O_2,

$$C_{O_2} = \alpha_{O_2}(1-H)P_{O_2}+S_{O_2}\beta \qquad (48)$$

Equation 42 becomes for O_2,

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

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

$$b = \beta\frac{\partial S_{O_2}}{\partial P_{CO_2}} V \tag{51}$$

and

$$K_{O_2} = [C_{O_2}v]^{in}-[C_{O_2}v]^{out}-D(P_{O_2}-\overline{P}_{O_2})-C_{O_2}\frac{dV}{dt} \tag{52}$$

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

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

$$\text{where } c = \frac{\partial C_{CO_2}}{\partial P_{O_2}} V \tag{54}$$

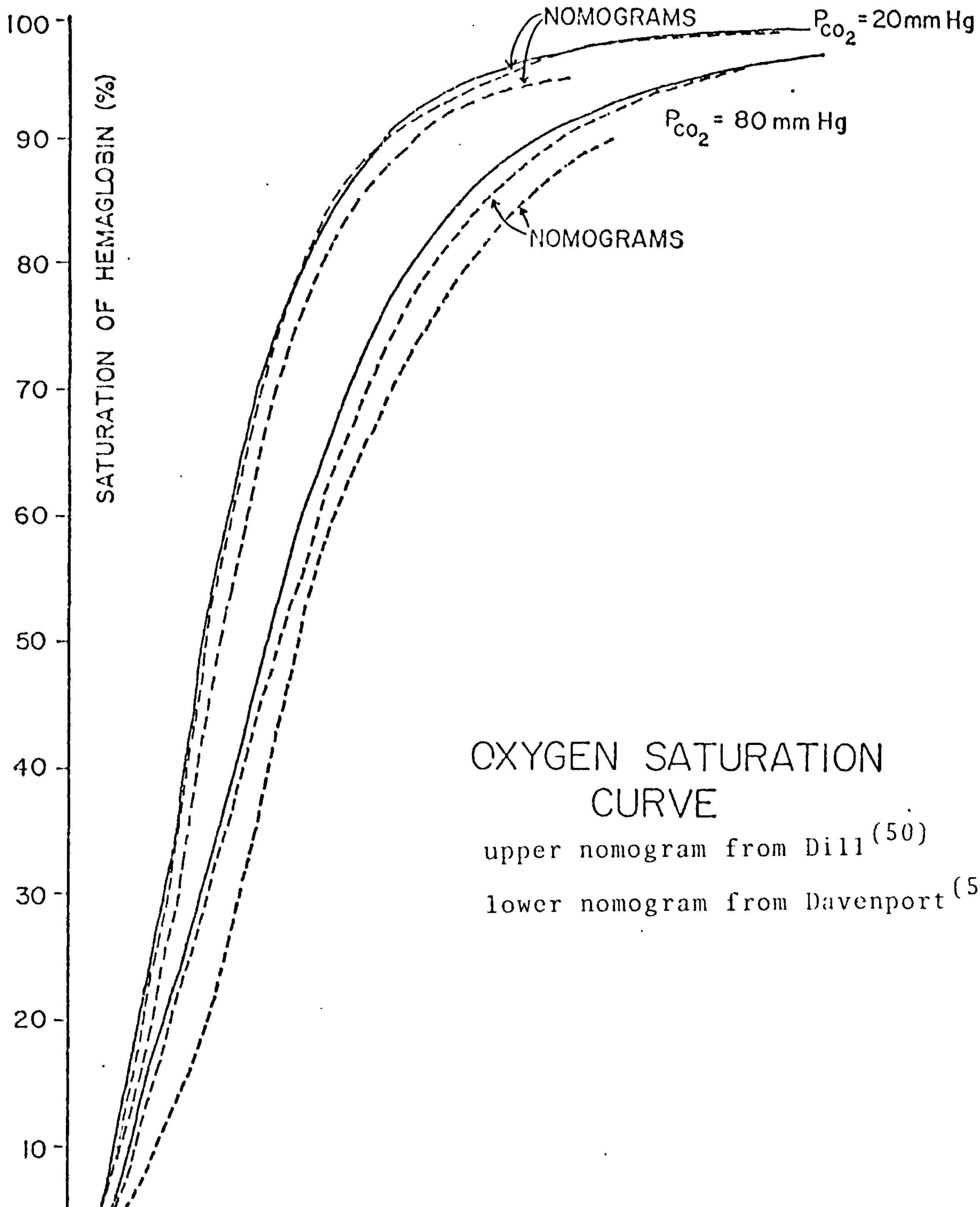

NOMOGRAMS
P_{CO_2} = 20mm Hg
P_{CO_2} = 80 mm Hg
NOMOGRAMS
SATURATION OF HEMAGLOBIN (%)
100
90
80
70
60
50
40
30
20
10
OXYGEN SATURATION CURVE
upper nomogram from Dill (50)
lower nomogram from Davenport (51)
O_2 PRESSURE (mm Hg)
10 20 30 40 50 60 70 80 90 100 110 120

FIGURE 15

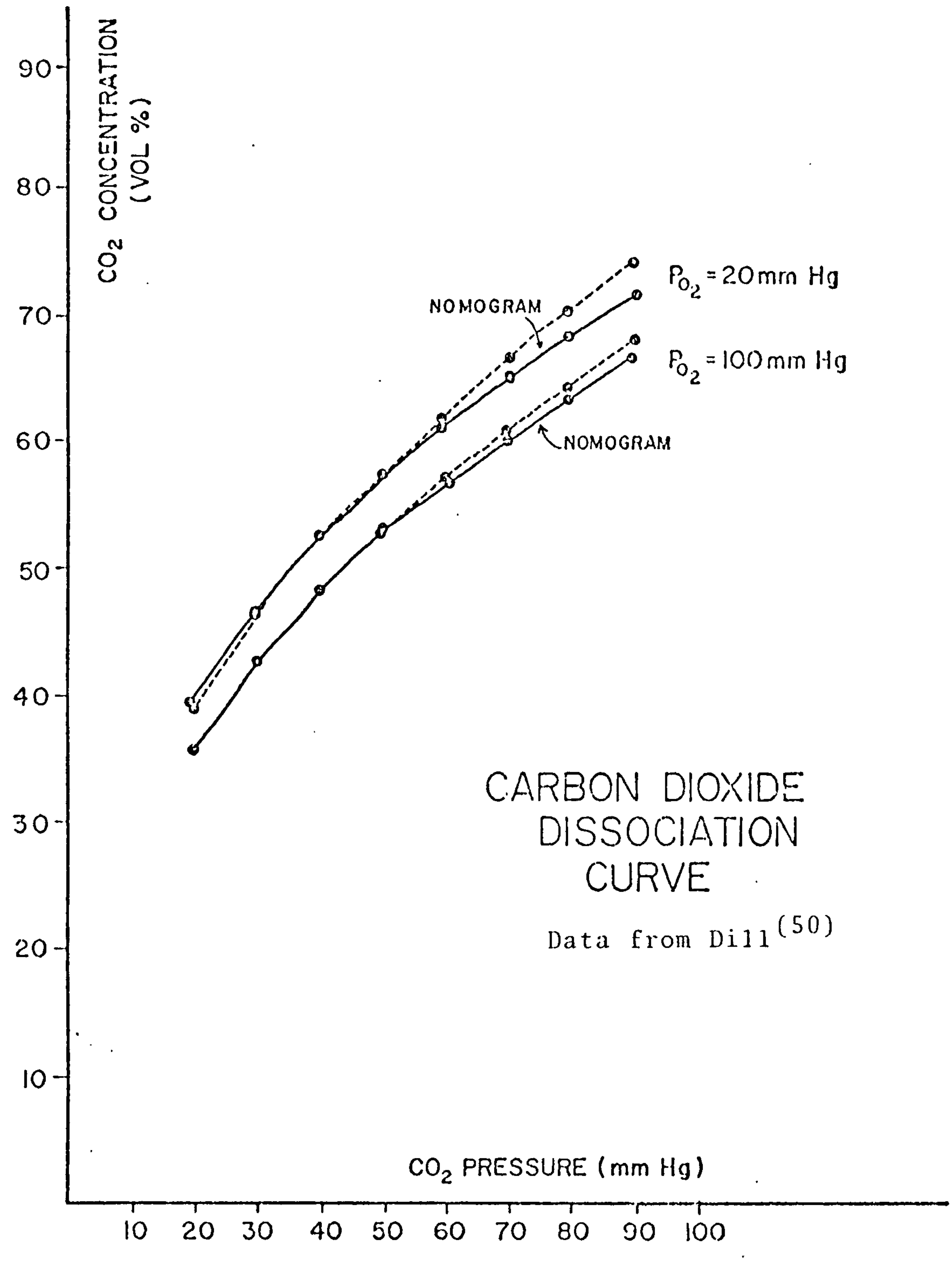

90
80
70
60
50
40
30
20
10
CO2 CONCENTRATION (VOL %)
NOMOGRAM
Po2 = 20mm Hg
Po2 = 100mm Hg
NOMOGRAM
CARBON DIOXIDE DISSOCIATION CURVE
Data from Dill(50)
CO2 PRESSURE (mm Hg)
10 20 30 40 50 60 70 80 90 100

FIGURE 16

$$d = \frac{\partial C_{CO_2}}{\partial P_{CO_2}} \tag{55}$$

and

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

Equations 49 and 53 are solved simultaneously, yeilding

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

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

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 $\frac{dP_{O_2}}{dt}$ and $\frac{dP_{CO_2}}{dt}$. 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

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

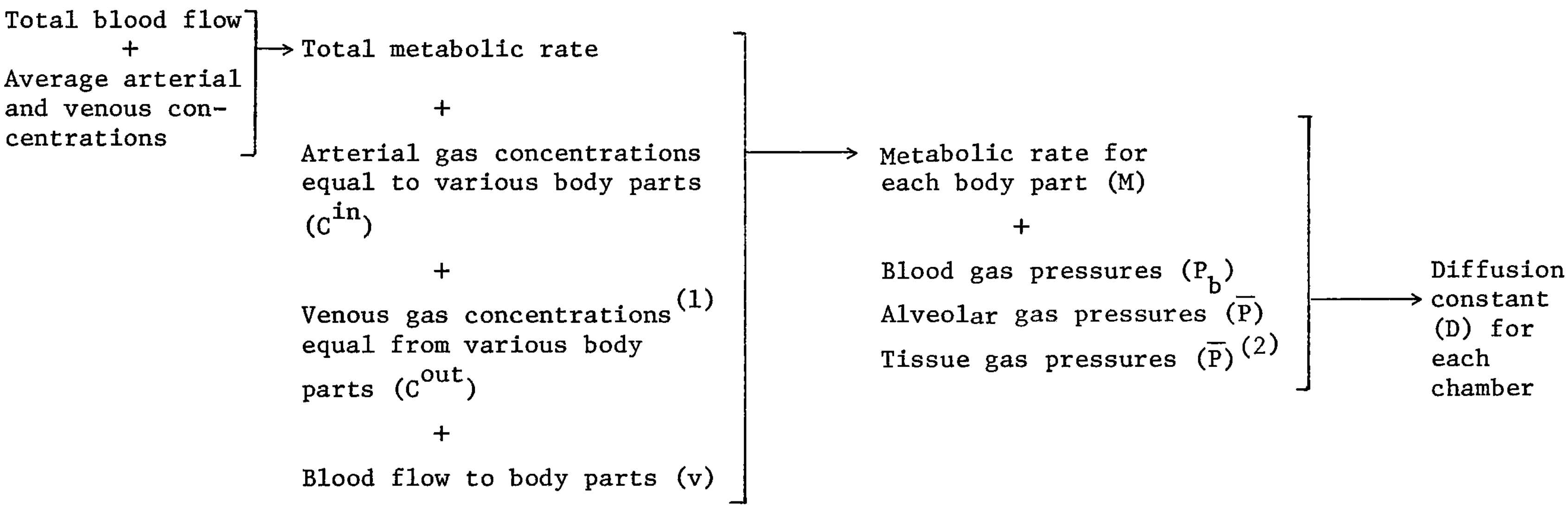
and

$$C^{in}v - C^{out}v = +M \qquad (60)$$

where M = the metabolic rate of disappearance of O_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]

FIGURE 17

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 (O, C, O , 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

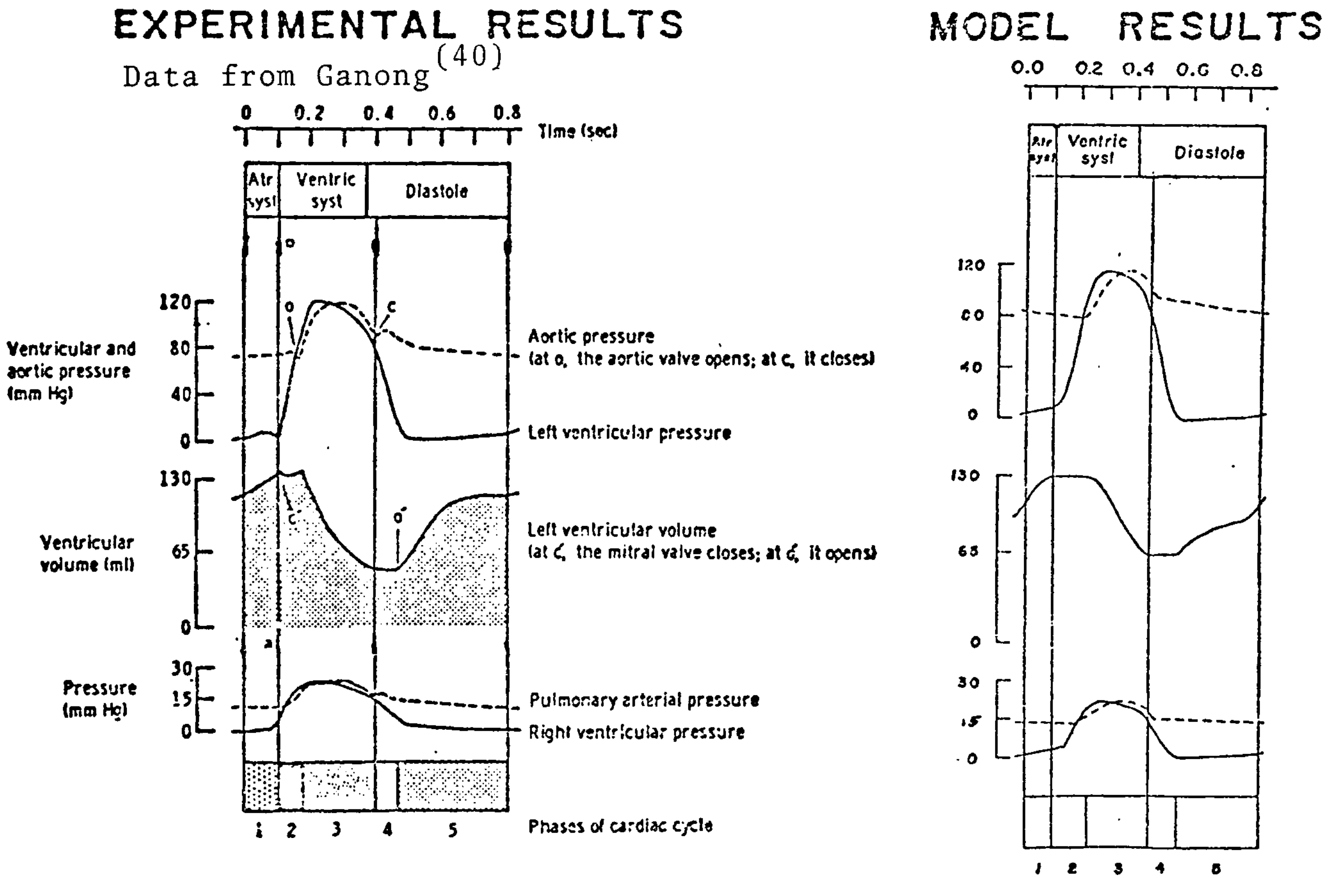


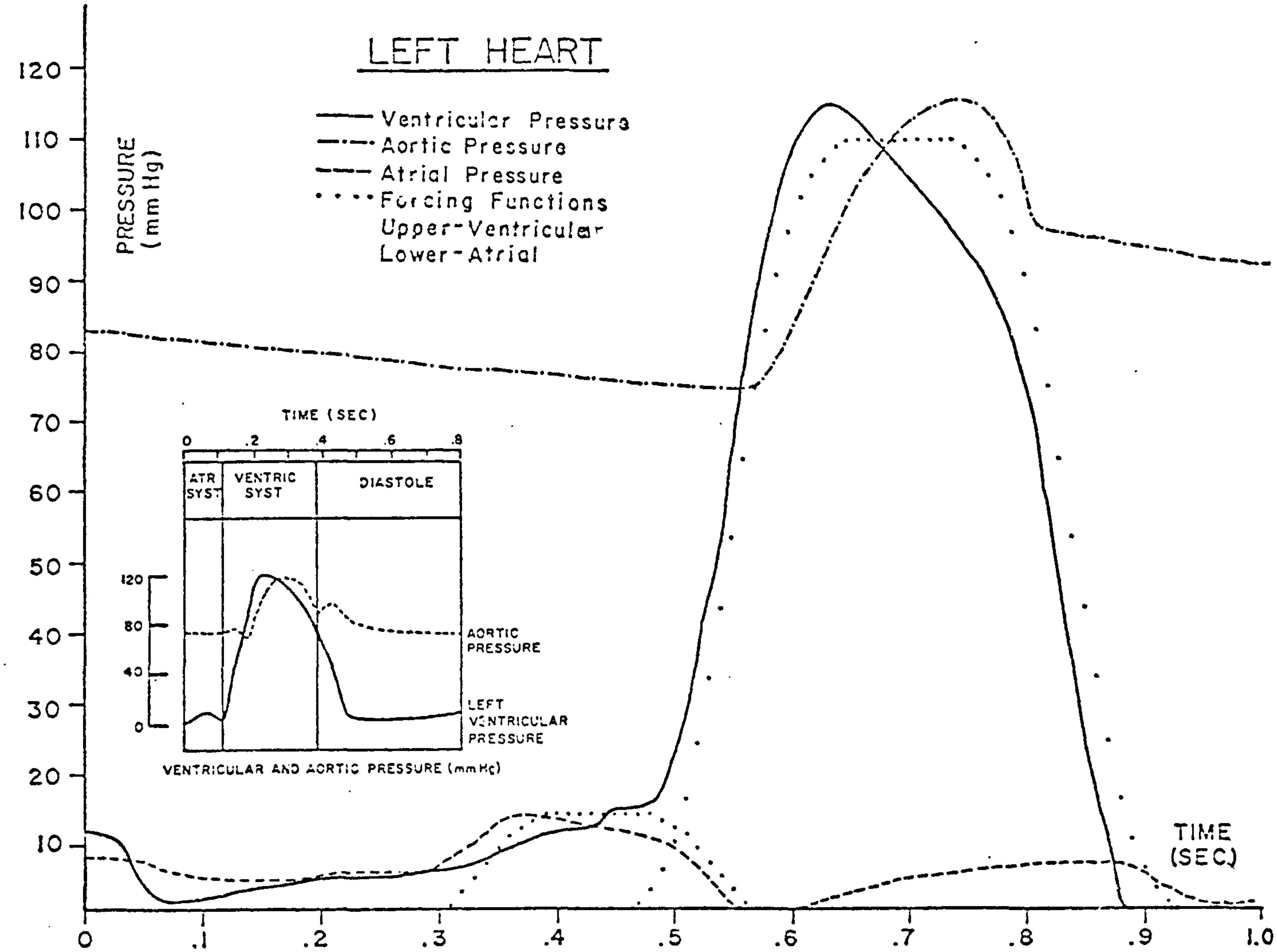
FIGURE 18

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.

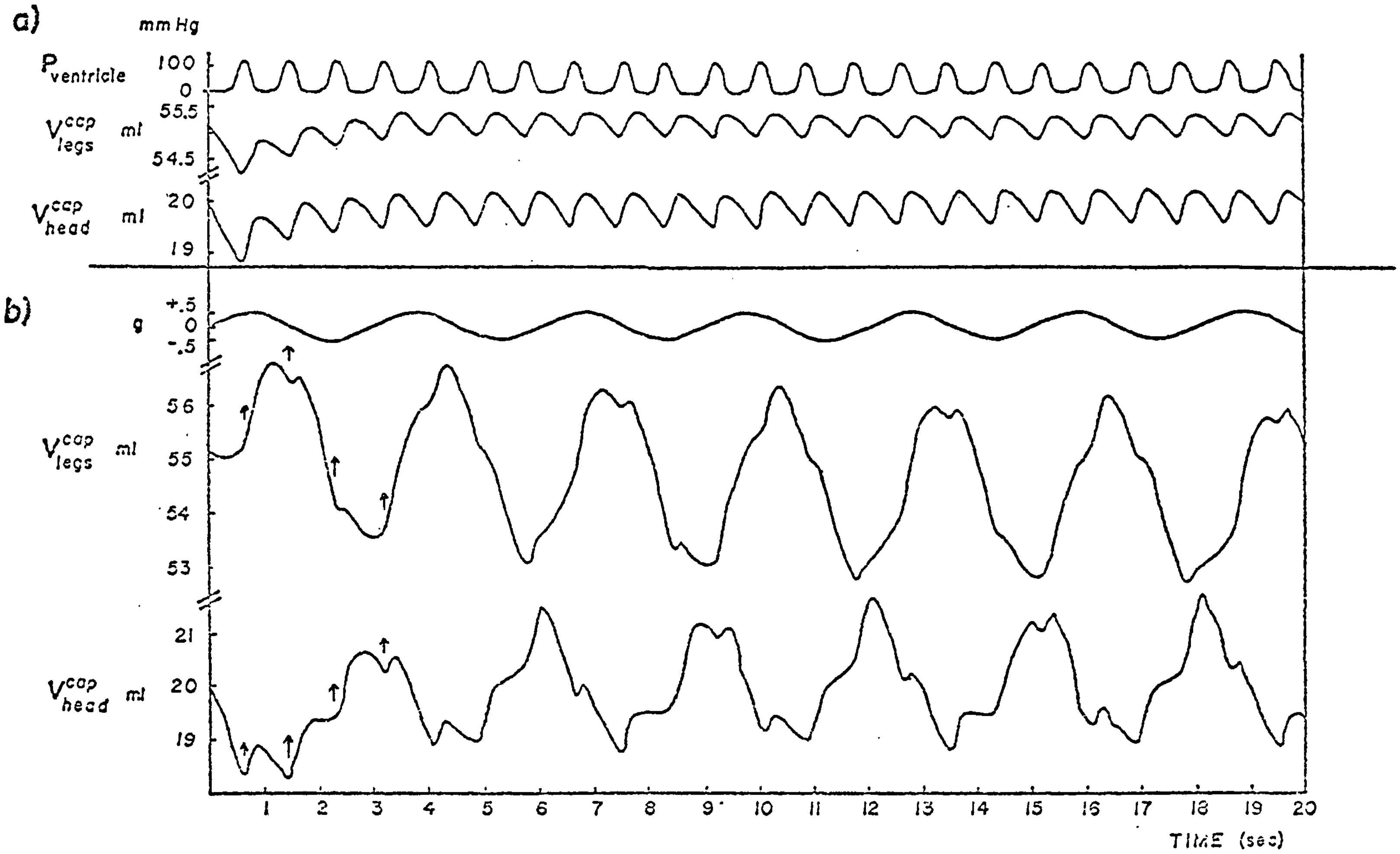
Figure 19 enlarges the left ventricular pressure and includes the forcing functions used as they occur in the model. Left atrial pressure has also been plotted so that the effect of the forcing function can be seen.

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 pulses. There is a phase shift between the heart pulse 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



MODEL RESULTS WITH FORCING FUNCTIONS

FIGURE 19



Capillary head and leg volumes for a) prone subject and b) sinusoidal variation of g_z.

FIGURE 20

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 (Steady State)	Model (passive breathing)
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

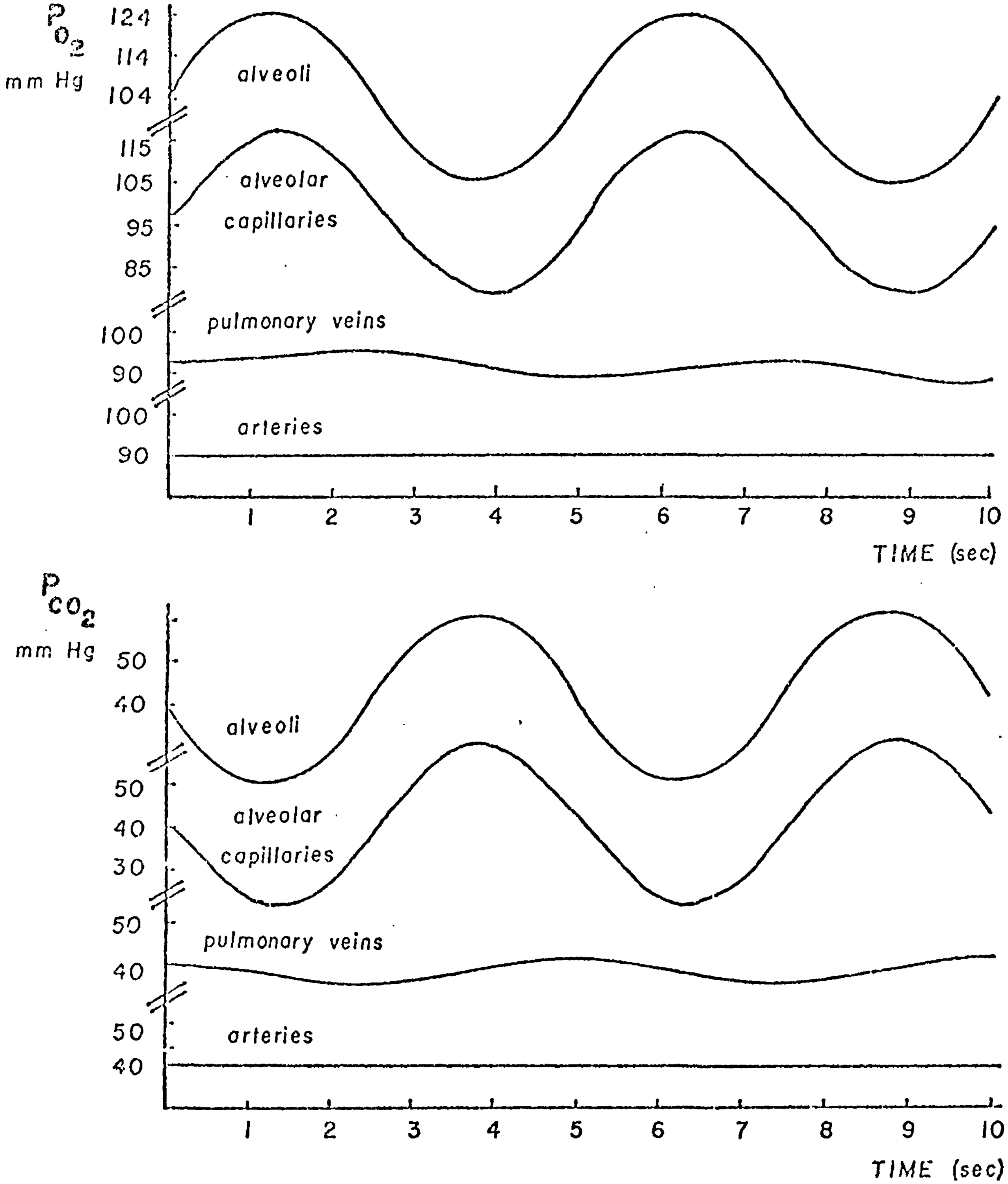
Partial Pressures of Carbon Dioxide (mmHg)

	Gordon(47)	Guyton(49)	Model (Steady State)	Model (passive breathing)
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).

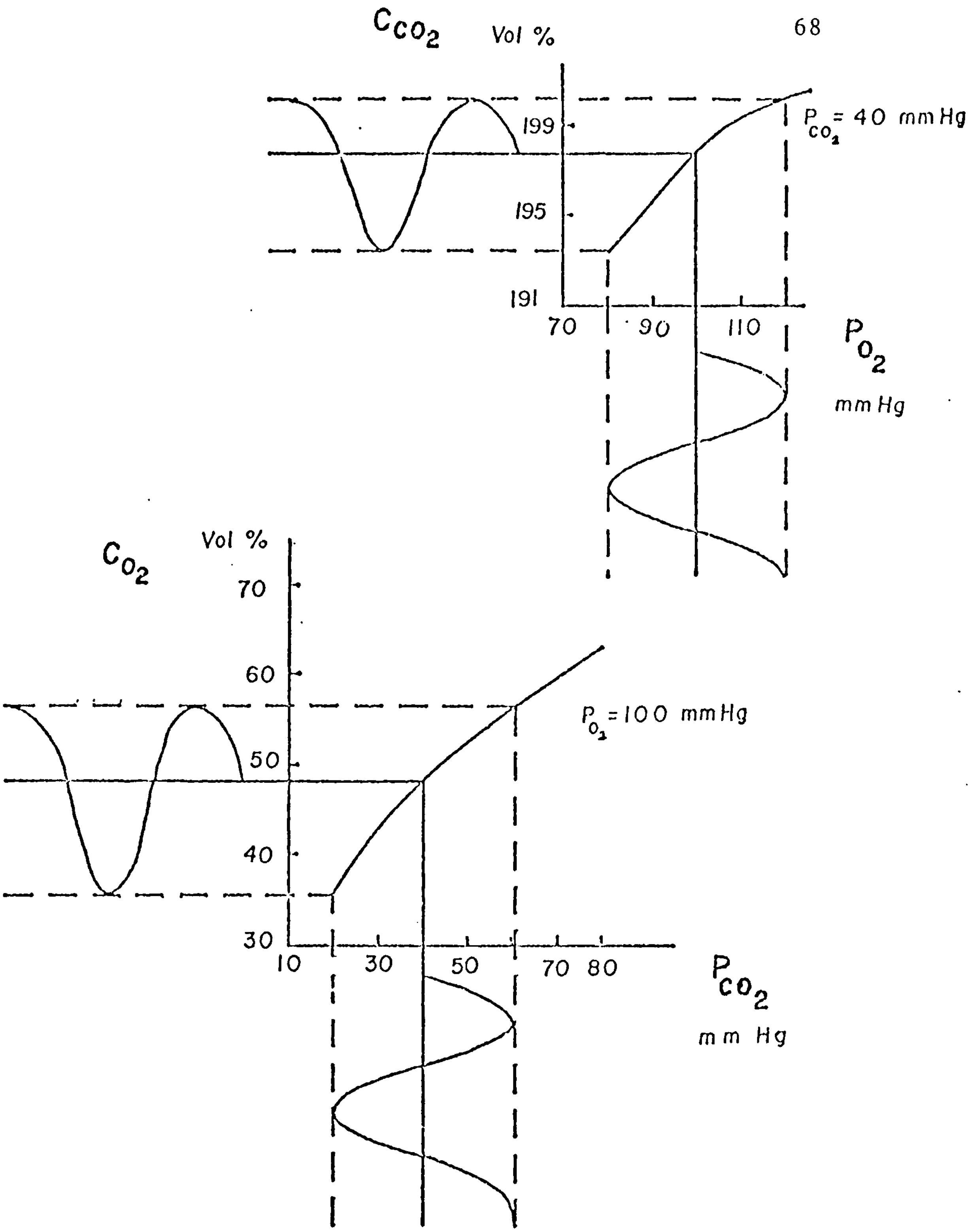
Figure 21 shows the model results in selected chambers. First, note that the partial pressure of CO_2 in the capillaries is almost identical to the partial pressure of the alveoli whereas the partial pressure of O_2 in the capillaries is about 7 mmHg below the partial pressure of the alveoli. This is 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 O_2 and CO_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 curves. This is a reflection of the shape of the O_2 and CO_2 dissociation curves as shown in Figure 22. These figures indicate that a sinusoidal variation of the partial pressures of O_2 and CO_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



Partial pressures in selected model chambers with constant blood flow in response to sinusoidal alveolar variations.

FIGURE 21



Non-sinusoidal variations of the concentrations of O_2 & CO_2 resulting from sinusoidal variation of the partial pressures of O_2 & CO_2.

FIGURE 22

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.

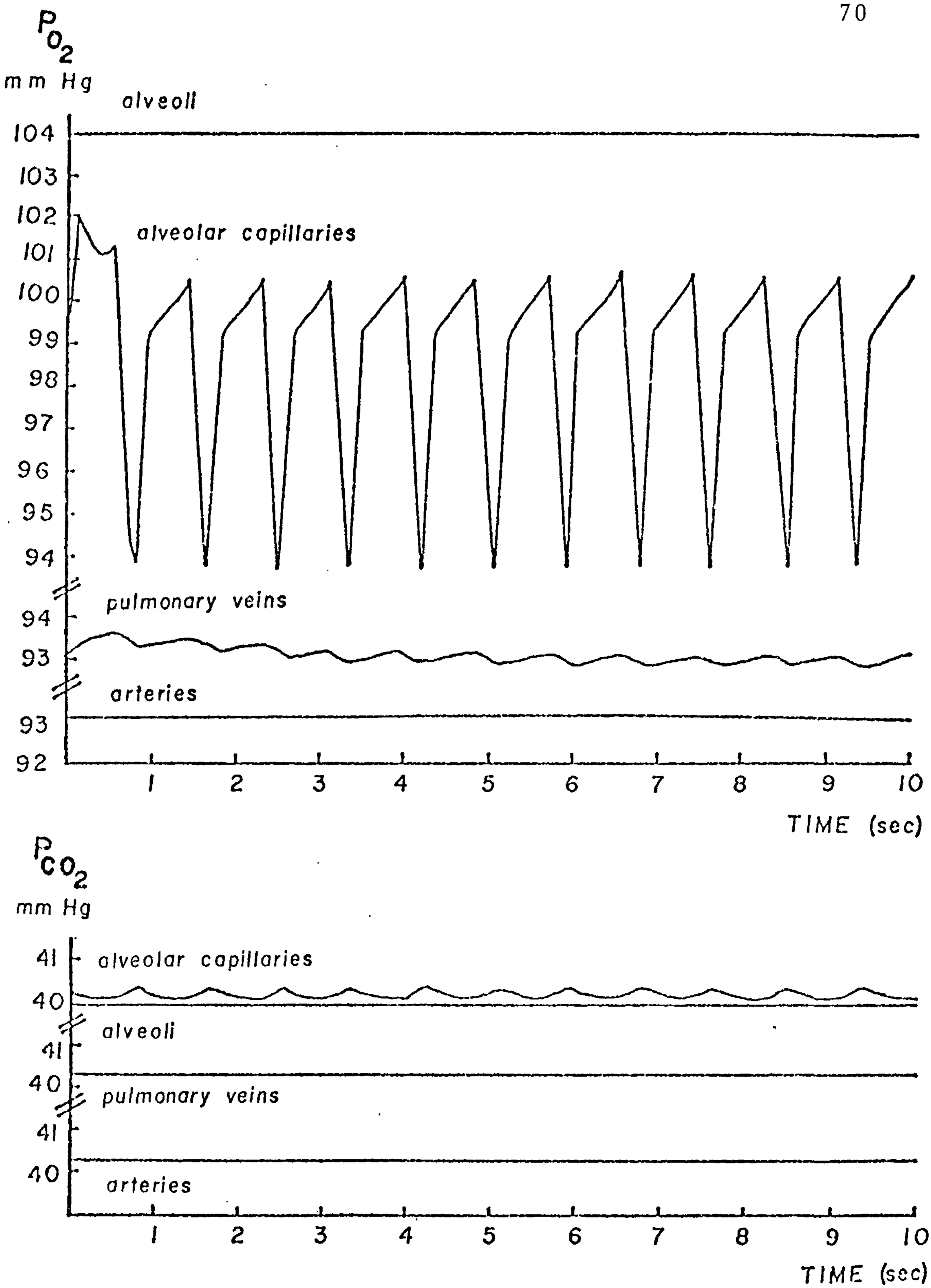
Finally, the sinusoidal fluctuation in the partial pressures of the alveoli are damped dramatically as the blood moves from one chamber to the next as shown in Figure 21. This effect is due to the assumption of perfect mixing in the model. In fact, by the time that the blood reaches the arteries, there 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



Partial pressures in selected model chambers with pulsatile blood flow and constant alveolar pressures.

FIGURE 23

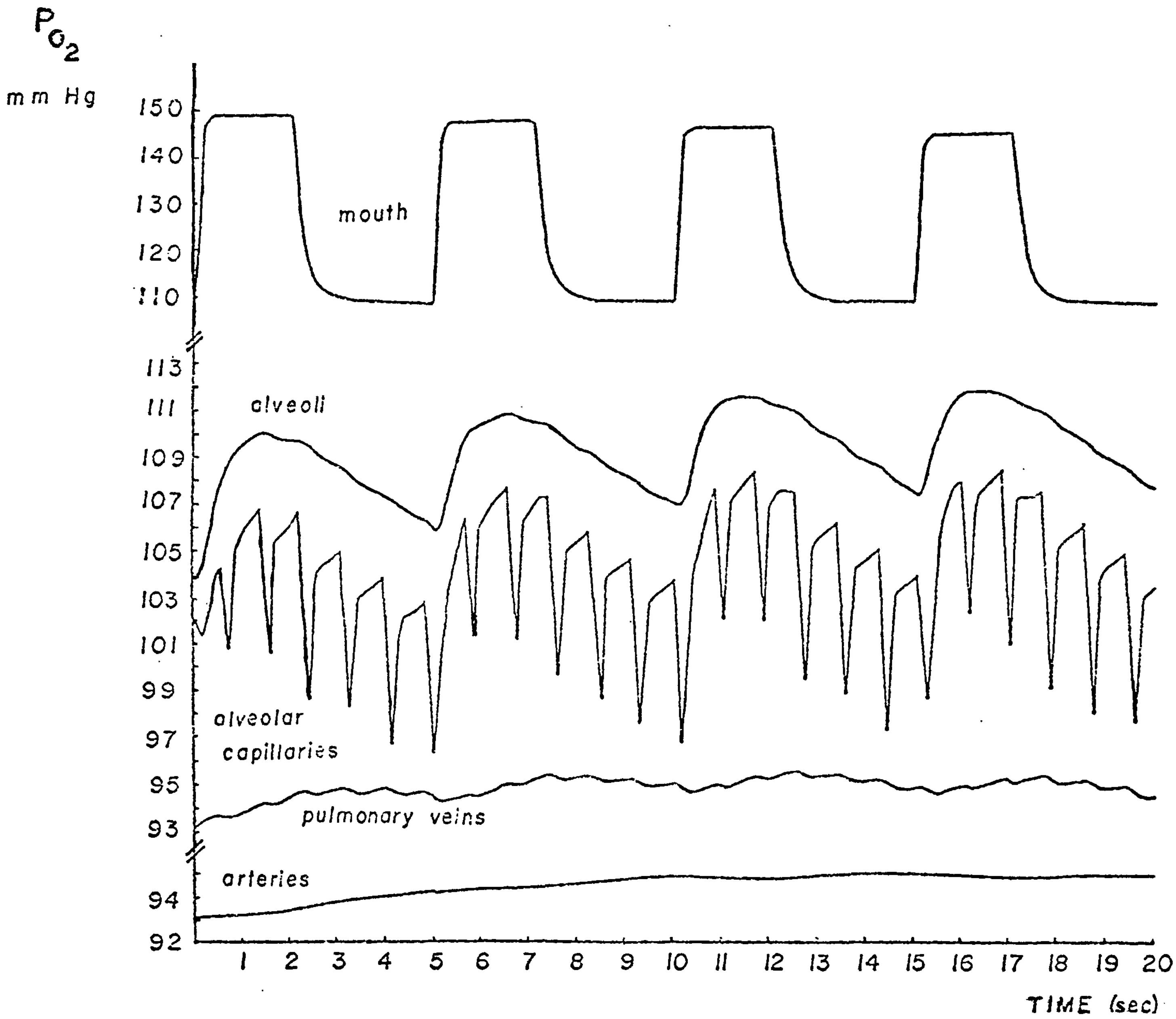
variation in distance capillary blood must travel (assuming plug flow) before it becomes fully oxygenated. This is much less pronounced in P_{CO_2} due to the greater diffusion rate.

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

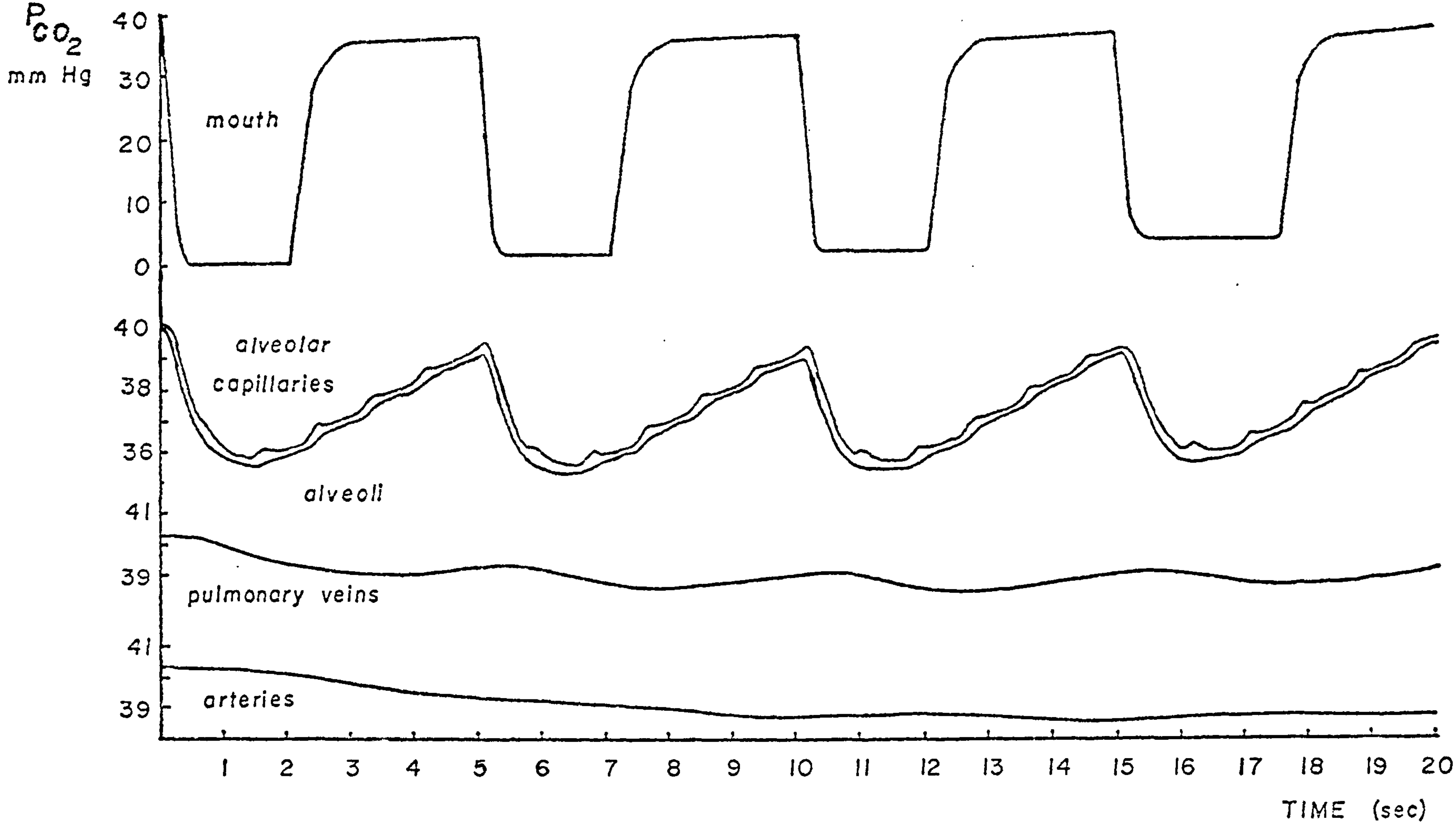
Once the testing of the intermediate model was completed, the complete model was tested. Due to the volume of data resulting, only the results for a few of the chambers are included.

Figures 24 and 25 show the partial pressures of O_2 and CO_2 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



Partial pressures of Oxygen in selected model chambers during passive breathing

FIGURE 24



Partial pressures of Carbon Dioxide in selected model chambers during passive breathing

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 cardio-pulmonary 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.

Chamber Equations (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:

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

where P = blood pressure in the chamber

P' = external pressure

$\tilde{P}$ = compliance presure (see Compliance Equation)

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

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

where R = resistance of blood, vessel wall, and tissue in transverse motion

V = volume of chamber

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

$$\frac{dV_m}{dt} = v_{in} - v_{out} \tag{1-2}$$

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

$$v^{out} = \sum_{i=1}^{q} v_{O_i}$$

$$\left.\frac{dP_{O_2}}{dt}\right|_m = \frac{dK_{O_2}^m - bK_{O_2}^m}{(ad-cb)} \tag{1-3}$$

$$\left.\frac{dP_{CO_2}}{dt}\right|_m = \frac{dK_{CO_2}^m - bK_{CO_2}^m}{(ad-cb)} \tag{1-4}$$

where $a = \alpha+\beta\cdot\frac{\partial S_{O_2}}{\partial P_{O_2}}$

$$b = \beta\cdot\frac{\partial S_{O_2}}{\partial P_{CO_2}}$$

$$c = \frac{\partial \overline{S}_{CO_2}}{\partial P_{O_2}}\cdot P_{CO_2}\cdot\gamma$$

$$d = \frac{\partial \overline{S}_{CO_2}}{\partial P_{CO_2}}\cdot P_{CO_2}\cdot\gamma+\overline{S}_{CO_2}\cdot\gamma$$

$$K_{O_2}^m = (Cv)^{in} - (Cv)^{out} - C_m\frac{dV_m}{dt}$$

where $(Cv)^{in} = \sum_{i=1}^{n} (CCFL)_{I_i}$

where $(CCFL)_{I_i} = C_{I_i} v_{I_i}$ if $v_{I_i} \geq 0$

$C_m v_{I_i}$ if $v_{I_i} < 0$

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

where $(CCFL)_{O_i} = \begin{cases} C_{O_i} v_{O_i} & \text{if } v_{O_i} \geq 0 \\ C_m V_{O_i} & \text{if } v_{O_i} < 0 \end{cases}$

$$C_{O_2} = \alpha \cdot P_{O_2} + S_{O_2} \beta$$

$$C_{CO_2} = \overline{S}_{CO_2} \cdot \gamma \cdot P_{CO_2}$$

$$\alpha = 1.339 \times 10^{-3} \frac{\mu \text{ moles}}{(\text{ml}) \cdot (\text{mmHg})}$$

$$\beta = 8.98 \ \mu \text{ moles/ml}$$

$$\gamma = 1.00 \frac{\mu \text{ moles}}{(\text{ml})(\text{mmHg})}$$

The subscripts are defined in Table III.

TABLE III

Chamber Number	Number of Connecting links in	Number of Connecting links out	Connecting Chamber Number "in"							Connecting Chamber Number "out"						
m	n	8	I_1	I_2	I_3	I_4	I_5	I_6	I_7	O_1	O_2	O_3	O_4	O_5	O_6	O_7
1	2	1	21	22						2						
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,

$$R_i^M v_i = P_i + \rho g h_i - (P_j + \rho g h_j)$$

where

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

For segments 3 and 13,

$$\left.\frac{M}{A^2}\right|_i \frac{dV_i}{dt} = (P_i + \rho g h_i) - (P_{i+1} + \rho g h_{i+1}) - R_i v_i - R_i' v_i^2$$

(NOTE: $\left.\frac{M}{A^2}\right|_i = L_i$ used in Chapter III)

Tissue Equations (same for all four tissue chambers)

$$\frac{dP_{O_2}^t}{dt} = D_{O_2}(P_{O_2} - \lambda_{O_2} P_{O_2}^t) - M_{O_2}$$

$$\frac{dP^t_{CO_2}}{dt} = D_{CO_2}(P_{CO_2} - \lambda_{CO_2} P_{CO_2}) + M_{CO_2}$$

$$\frac{dP_{CO}}{dt} = D_{CO}(P_{CO} - \lambda_{CO} P_{CO})$$

$$\frac{dP_x}{dt} = D_x(P_x - \lambda_x P_x)$$

Compliance Equation

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

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)	$R(\frac{mmHg}{ml/sec})$
1	.148
2	.00963
3	.0024
4	0.
5	.294
6	.294
7	.294
8	.294
9	.294
10	.294
11	.033
12	.019
13	.012
14	0.
15	1.31
16	.24
17	1.11
18	.9
19	.327
20	.12
21	.553
22	.45
23	.327
24	.0608
5 in	.588
6 in	.588
7 in	.588
8 in	.588
9 in	.588
10 in	.588
11 in	7.19
15 in	9.60
16 in	1.76
17 in	8.12
18 in	6.62

λ is calculated from initial values of R and V in the program using

$$= R \cdot V^2$$

SEG. NO. (i)	M/A^2 $(\frac{mmHg}{ml/sec})$	R' $(\frac{mmHg}{(ml/sec)^2})$	$\bar{R}$ $(\frac{mmHg}{ml/sec})$
2	-	-	0.
3	1667.	3.0×10^{-7}	0.
4	-	-	.0136
12	-	-	0.
13	333.	1.5×10^{-6}	0.
14	-	-	.068

Forcing Function Used in Simplified Model (Chapter III)

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

where

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

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

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

$A = 1/2(t-t_0-a/2)^4/\sigma^2$

$\sigma = (a/4)^2$

a = pulse duration

t_0 = starting time of the pulse

	t_0 (sec)	a (sec)	(mmHg)
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 β = pulse rate = 70/sec

P_{CO_2} (mmHg)	$P^t_{O_2}$ (mmHg)	$P^t_{CO_2}$ (mmHg)	D_{O_2} (μ moles/mmHg)	D_{CO_2} (μ moles/mmHg)
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				

TABLE IV

Constants and Initial Values

Chamber Number	k (1/mmHg)	P' (mmHg)	τ (ml)	P_0 (mmHg)	V (ml)	V_0 (ml)	P_{O_2} (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

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:

$$CO_2+H_2O \rightleftharpoons H_2CO_3 \qquad (2\text{-}1)$$

$$H_2CO_3 \rightleftharpoons H^++HCO_3^- \qquad (2\text{-}2)$$

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

$$CO_2+HbO_2 \rightleftharpoons HbCO_2+O_2 \qquad (2\text{-}4)$$

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% $HbCO_2$).[49] The uptake of total CO_2 in all forms by whole blood is found experimentally as shown in the CO_2 dissociation curve (Figure 16) which plots the concentration of CO_2 in whole blood versus the partial pressure of CO_2 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 (Reaction 4). This accounts for the dependence of the 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 CO_2. 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 $C_{O_2}(P_{O_2}, P_{CO_2}$, pH). These functional dependences are incorporated into 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,

$$\frac{[H^+][HCO_3^-]}{[H_2CO_3]} = K' \qquad (2\text{-}5)$$

and reaction 2-1

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

follows

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

assuming the concentration of water to be constant. Rearranging

$$-\log[H^+] = \log K + \log([HCO_3^-]/[CO_2]) \qquad (2\text{-}8)$$

or

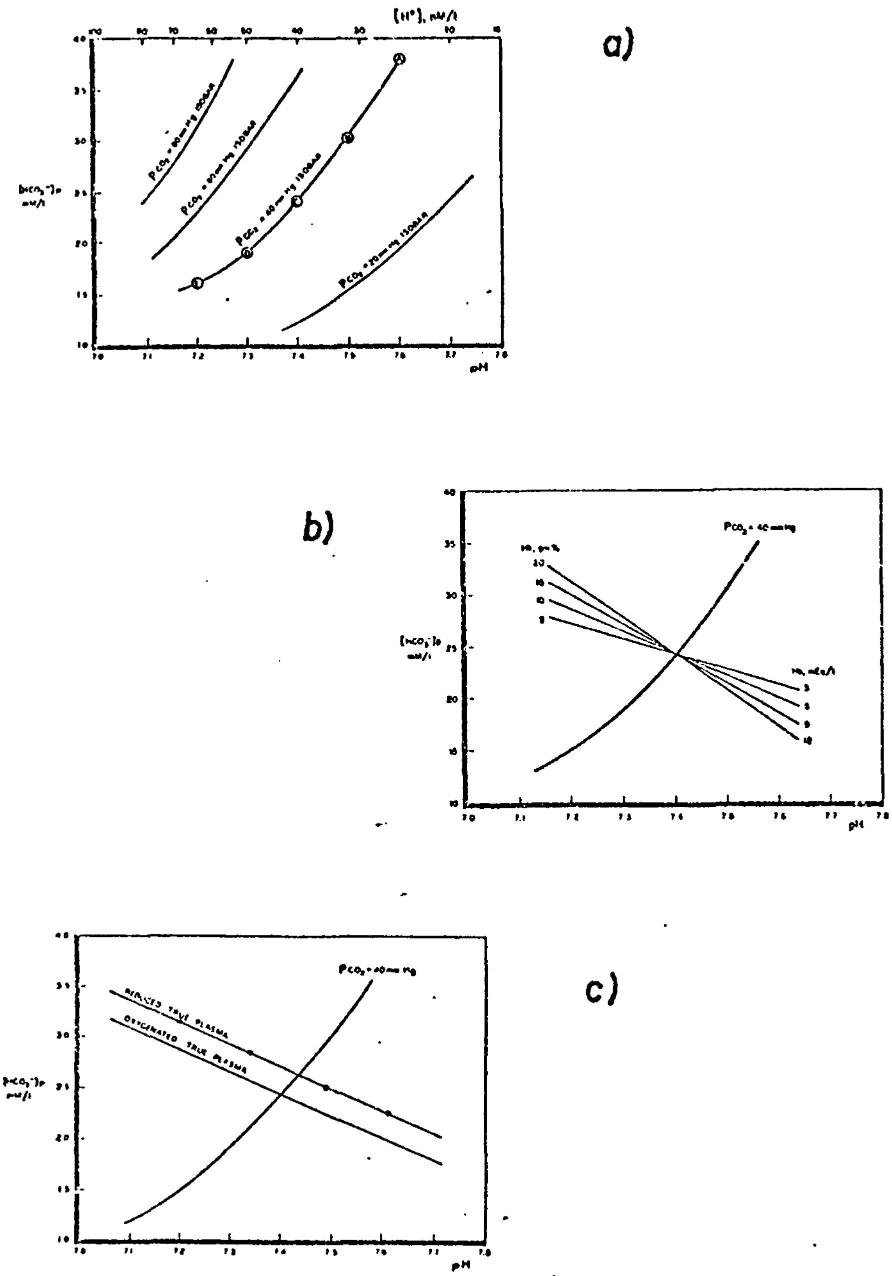
$$pH = pK + \log\frac{[HCO_3^-]}{[CO_2]} \qquad (2\text{-}9)$$

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

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

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



The concentration of HCO_3^- versus pH

FIGURE 26

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

$$pH = m[HCO_3^-]+b \qquad (2\text{-}11)$$

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.

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

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

$$\frac{dn}{dt} = \frac{-AS}{\ell}(C_{w_1} - C_{w_2}) \tag{2-13}$$

where $C_{w_2}(C_{w_1})$ = concentration of gas in capillary wall adjacent to the blood (alveoli or tissue)

ℓ = thickness of capillary wall.

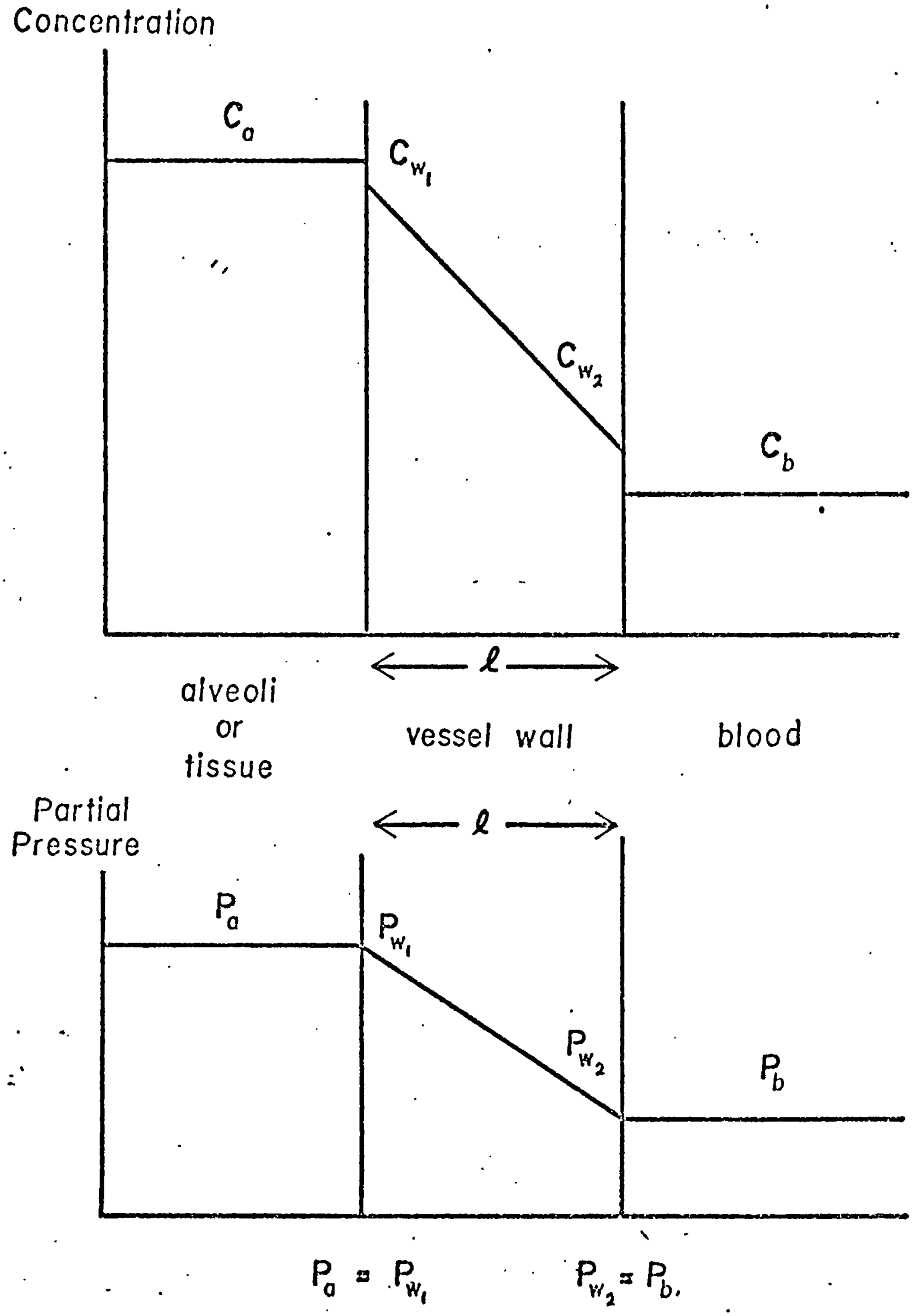
This can be written in terms of the partial pressures in the vessel wall by introducing the coefficient α as

$$\frac{dn}{dt} = \frac{-AS}{\ell}(\alpha P_{w_1} - \alpha P_{w_2}) \tag{2-14}$$

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)

$$P_a = P_{w_1} \tag{2-15}$$



Diffusion of gasses through vessel walls.

FIGURE 27

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

and

$$\frac{dn}{dt} = \frac{\alpha AS}{\ell}(P_a - P_b) = D(P_a - P_b) \qquad (2\text{-}17)$$

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

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

is the diffusion coefficient used in our model.

Part C

The Bunsen solubility coefficient, β, is defined as follows

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

where V_o = 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:

$$PV = nRT \qquad (2\text{-}20)$$

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

At STP,

$$P_oV_o = nRT_o \tag{2-21}$$

where P_o = 760 mmHg

T_o = 273 K

n = number of moles of gas occupying a volume V_o

Substituting into Equation ,

$$\frac{\beta P_o}{RT_o}P = \frac{n}{V_s} \tag{2-22}$$

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

$$\frac{\beta P_o}{RT_o}P = C = \alpha P \tag{2-23}$$

Thus $$\alpha = \frac{\beta P_o}{RT_o} . \tag{2-24}$$

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.

Oxyhemoglobin Saturation, S_{O_2}

$$S_{O_2} = \frac{U}{1+V}$$

where $U = .925V+2.8V^2+30V^3$

$$V = CP_{O_2}$$

$$C = 5.727x10^{-3}z+4.273x10^{-3}$$

$$z = \exp[(pH-7.4)\cdot 1.812]$$

and $pH = 7.59+y-.2741 \ln P_{CO_2}/20.0$

where $y = .045\cdot(1-S_{O_2})$

S_{O_2} and pH are dependent on another and are calculated with P_{O_2} & P_{CO_2} as inputs by the two step iterative procedure given below;

set y = 0.0

calculate pH

calculate S_{O_2}

recalculate y

calculate pH

calculate S_{O_2}

<u>Concentration of Carbon Dioxide</u> C_{CO_2}

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

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

H = hematocrit = .45

$C_p = 0.0307 \cdot (1+Y')$

$Y' = \exp[2.3961 \cdot pH - 14.7162]$

$d_1 = d_2 + (d_3 - d_2) \cdot (1 - S_{O_2})$

$d_2 = .59 + .2913(7.4 - pH$

$d_3 = .664 + .2275(7.4 - pH)$

<u>Rate of Change of the Oxy-Hemoglobin Saturation function with Respect to the Partial Pressure of Oxygen</u>

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

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

$A_2 = .045 A_3$

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

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

<u>Rate of change of the Oxy-Hemoglobin Saturation Function with Respect to the Partial Pressure of Carbon Dioxide,</u> $\frac{\partial S_{O_2}}{\partial P_{CO_2}}$

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

<u>Rate of Change of the Concentration of Carbon Dioxide with Respect to the Partial Pressure of Oxygen,</u> $\frac{\partial C_{CO_2}}{\partial P_{O_2}}$

$$\frac{\partial C_{CO_2}}{\partial P_{O_2}} = \frac{\partial S_{CO_2}}{\partial P_{O_2}} P_{CO_2}$$

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 Carbon Dioxide with Respect to the Partial Pressure of Carbon Dioxide $\frac{\partial C_{CO_2}}{\partial P_{CO_2}}$

$$\frac{\partial C_{CO_2}}{\partial P_{CO_2}} = S_{CO_2} + \frac{\partial S_{CO_2}}{\partial P_{CO_2}} P_{CO_2}$$

where $$\frac{\partial S_{CO_2}}{\partial P_{CO_2}} = (1-H)\overline{C}_P + H\cdot\overline{d}_1\cdot C_P + H\cdot d_1\cdot\overline{C}_P$$

$$\overline{C}_P = .0307f$$

$$f = 2.3961\cdot Y'\cdot\overline{b}$$

$$\overline{d}_1 = \overline{d}_2 + (\overline{d}_3 - \overline{d}_2)(1 - S_{O_2}) - (d_3 - d_2)\frac{\partial S_{O_2}}{\partial P_{CO_2}}$$

$$\overline{d}_2 = -.2913\overline{b}$$

$$d_3 = .2275\overline{b}$$

$$\overline{b} = -.045\frac{\partial S_{O_2}}{\partial P_{CO_2}} - .2741/P_{CO_2}$$

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.

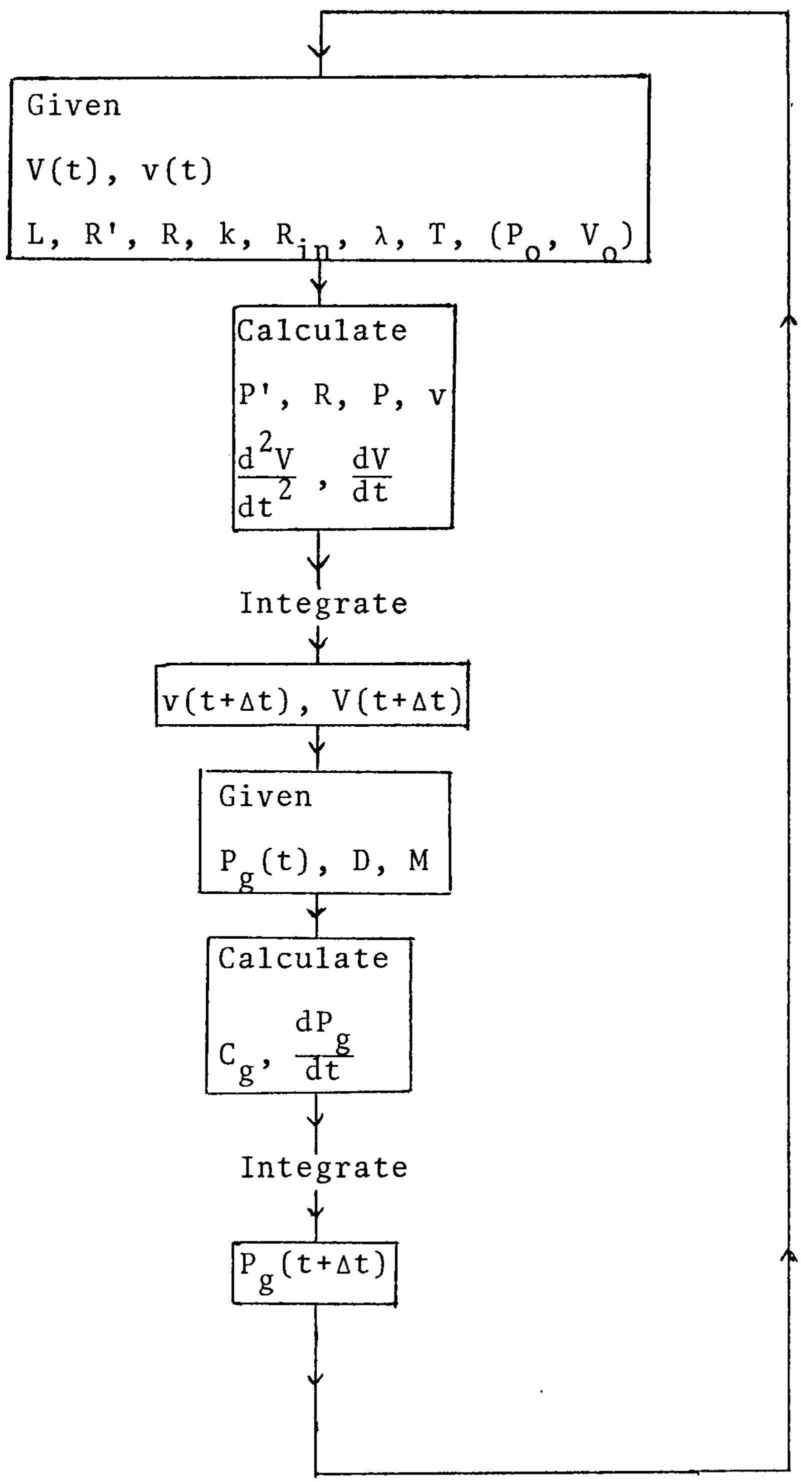
RESIST--calculates resistance of all segments. Options available are LOPR=neg makes $R = k/(V^2+V)$. This is to model non-Newtonian flow in the capillaries if desired. Not presently in use.

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.



The order of calculation within the program.

Figure 27

SKON--calculates compliance pressures. Options available are LOPP1 = 0 means compliance constant. LOPP1 $\neq$ 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 $\frac{dP_{O_2}}{dt}$, $\frac{dP_{CO_2}}{dt}$, and $\frac{dP_{CO}}{dt}$ 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}, $\frac{dS_{O_2}}{dP_{O_2}}$, $\frac{dS_{O_2}}{dP_{CO_2}}$, $\frac{dC_{CO_2}}{dP_{O_2}}$, and $\frac{dC_{CO_2}}{dP_{CO_2}}$.

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

<u>Calling Structure</u>

BCC

BCDRIV | Subroutines of BMAIN

O2DRIV | XDRIV

SATUR | GFLTST | GFLTST

FLTEST | FLTEST

BCMAIN--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

```
C    SUBROUTINE MAIN
      INTEGER  HAMING, OPT1, OPT2, FBOPT
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8  LAMBO2,LAMBC2,LAMBCO,LAMBX,LCO2,LCC2,LCCO,LCX,METO2,METC2
     &       ,N, NW
      DIMENSION R(24),FKDA(24),PP(24),V(24),VO(24),H(24),DV(24),
     1VFLOW(24),VFLIN(18),RIN(18),PC(24),P(24)
      DIMENSION FCTB(86),DFCTB(86),FCTSB(86,4),DFCTSB(86,3),TEB(86)
      DIMENSION GPO2(24),GPC2(24),GPCO(48),GPX(1,24),CTO2(4),CTC2(4),
     1CTCO(4),CTX(1,4),DIFO2(10),DIFC2(10),DIFCO(10),DIFX(10),CPH(24),
     2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24 ),DGPO2(24),
     3DGPC2(24),DGPCO(48),DGPX(1,24),AX(1)
       DIMENSION DCTO2(4),DCTC2(4),DCTCO(4),DCTX(1,4),METO2(4),METC2(4)
C    THIS DIMENSION STATEMENT IS ONLY IN BMAIN
       DIMENSION TE(62), FCT(62), FCTS(62,4), DFCT(62), DFCTS(62,3)
       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),PHD(12), PBS,
     &    DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
     &    PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
     &    S(4), T, DT, TN(12), TOTN(12),VHD(12), VD, VLUNG,
     &    SQT(12), DP(12), HDT
       COMMON /KN1/KNTB
       COMMON /BCM/ TB,TMAXB
       COMMON /LMN/ KNT
       COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GPO2,CTO2,
     &GPC2,CTC2
       COMMON /CMN1T/GPCO,CTCO,GPX,CTX
       COMMON /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGPO2,DCTO2,
     1DGPC2,DCTC2
       COMMON /CMN2T/DGPCO,DCTCO,DGPX,DCTX
       COMMON /BLK1/ N(12,3), VTX, VALV(6), PALV(6), SW(12), VSPR
       COMMON /BLK2/ DN(12,3), DVTX, DVALV(6), DPALV(6), DSW(12), DVSPR
       EQUIVALENCE (V,FCTB,FCTSB),(DV,DFCTB,DFCTSB)
     &            ,(N(1), FCT(1), FCTS(1) ), (DN(1), DFCT(1), DFCTS(1) )
C   FCTB(V=24,VP=6,C=24,CT=4)
       READ (5,2) KNTTST,CONV,RTC
     2 FORMAT (I5,2E10.3)
       CALL BCMAIN(LDUMMY)
       CALL LMAIN(LDUMMY)
       CALL LBCONV(RTC)
       IF (KNTTST.LT.0) GO TO 100
    10 CALL BCMN1(LDUMMY)
       CALL BLCONV(CONV)
       CALL LMN1(LDUMMY)
       CALL LBCONV(RTC)
    20 CONTINUE
       IF (T-TB.LT.1.E-04) GO TO 35
       IF (KNTB.GT.3) CALL BCMN3(LDUMMY)
       IF (KNTB.LE.3) CALL BCMN2(LDUMMY)
    30 CALL BLCONV(CONV)
```

ready

*

```
      IF (TB-T.LT.1.E-04) GO TO 45
   35 CONTINUE
      IF (KNT.GT.3) CALL LMN3(LDUMMY)
      IF (KNT.LE.3) CALL LMN2(LDUMMY)
   40 CALL LBCONV(RTC)
   45 IF (TB-TMAXB) 20,50,50
   50 CONTINUE
      CALL LMN4(LDUMMY)
  100 CONTINUE
C   REVERSE ORDER OF CALL BCMN AND LMN'S AND ADJUST KNTTST
C     STATEMENTS FOR NEG. KNTTST.
      STOP
      END
      SUBROUTINE LBCONV(RTC)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 LAMBO2,LAMBCO,LAMBX,LCO2,LCC2,LCCO,LCX,METO2,METC2
     &      ,N, NW
      COMMON /AVE/ PPAVE
      COMMON /BLK1/ N(12,3), VTX, VALV(6), PALV(6), SW(12), VSPR
      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,
     &    DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
     &    PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
     &    S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
     &    SQT(12), DP(12), HDT
      COMMON /BCMBLK/ R,FKDA,PP,VO,A2DM2,A2DM3,A2DM4,A2DM12,A2DM13,
     1A2DM14,H,RHO,GHOLD,RIN,PHOLD,PC
      COMMON /BCONLY/LCO2,LCC2,LCCO,LCX
      DIMENSION R(24),FKDA(24),PP(24),VO(24),H(24),DV(24),
     1VFLOW(24),VFLIN(18),RIN(18),PC(24)
      DIMENSION LCO2(6),LCC2(6),LCCO(6),LCX(6)
      DIMENSION PHOLD(24)
C   WARNING...THIS DIM IS IN THIS SUBROUTINE ONLY
      DO 10 I=1,6
      LCO2(I)=C(I+6,1)*RTC
      LCC2(I)=C(I+6,2)*RTC
      PP(I+4)=PALV(I)-760.
   10 CONTINUE
      PPAVE=PPL+(PG(3)+PG(4))/2.-760.
      RETURN
      END
      SUBROUTINE BLCONV(CONV)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8  LAMBO2,LAMBC2,LAMBCO,LAMBX,LCO2,LCC2,LCCO,LCX,METO2,METC2
      COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GPO2,CTO2,GPC2,CTC2
      COMMON /CMN1T/GPCO,CTCO,GPX,CTX
      COMMON /BC/LAMBO2,LAMBC2,LAMBCO,LAMBX,AO2,AC2,ACO,AX,HEMAT,DIFO2,
     1DIFC2,DIFCO,DIFX,CPH,CSATO2,CSATC2,METO2,METC2
      COMMON /BCONLY/LCO2,LCC2,LCCO,LCX
        COMMON /BLK12/ DIFF(6,3)

ready

*
```

```
      DIMENSION V(24),METO2(4),METC2(4)
      DIMENSION GPO2(24),GPC2(24),GPCO(48),GPX(1,24),CTO2(4),CTC2(4),
     1CTCO(4),CTX(1,4),DIFO2(10),DIFC2(10),DIFCO(10),DIFX(10),CPH(24),
     2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24 ),DGPO2(24),
     3DGPC2(24),DGPCO(48),DGPX(1,24),AX(1)
      DO 10 N=1,6
      DIFF(N,1)=DIFO2(N)*(LCO2(N)-LAMBO2*GPO2(N+4))*CONV
      DIFF(N,2)=DIFC2(N)*(LCC2(N)-LAMBC2*GPC2(N+4))*CONV
   10 CONTINUE
      RETURN
      END
```

ready

*

```
      SUBROUTINE LMAIN(LDUMMY)
      INTEGER  HAMING, OPT1, OPT2, FBOPT
      IMPLICIT  REAL*8( A-H, O-Z )
      REAL*8 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, G3, G4,
     &       PTXG, PTXIN, PTXEX, PCALC, CTCALC, VLDIF, PRCT, ASUM, PV,
     &       RES, PPLC, ASUM1, AWSUM, AWSUM1, RTOT, T0, T2, TTEST
      DIMENSION TE(62), FCT(62), FCTS(62,4), DFCT(62), DFCTS(62,3),
     &    PRCT(12,4), ASUM(4), PV(12), RES(12), PPLC(3), 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), DSW(12), DVSPR
      COMMON /BLK3/ B(8,12), FS(12), RA(12), RE(12), W(12), DTM
      COMMON /BLK4/ TIN, TEX, PDIN, TB, Y1, Y2, Y3, Y4, PTXIN, PTXEX,
     &               PDEX, PDAG, PTXG, T0, T2, PB, FCTST, FSPRS, TTEST
      COMMON /BLK5/ A(7,8)
      COMMON /BLK7/ PCALC(12), CTCALC(12), VLDIF
      COMMON /BLK8/ PHI, GF, GR, GT, GD, HP, PIE, ANG1, ANG2, PHI1, PHI2
      COMMON /BLK10/ PGAV, DPGAV, PGAVB, DPGAVB
      COMMON /BLK11/ PTX, DPTX, PD, DPD, G1, G2, G3, G4, FBOPT
      COMMON /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), PBS,
     &    DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
     &    PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
     &    S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLONG,
     &    SQT(12), DP(12), HDT
      EQUIVALENCE (N(1), FCT(1), FCTS(1) ), (DN(1), DFCT(1), DFCTS(1) )
```

ready

*

```
      DATA PIE, HP / 3.141592654 D0, 1.570796327 D0 /
      REWIND 12
      UNSTAB = 0.
      NF = 62
      NG = 3
      TO = 0.
      DO 50  K= 1,4
      DO 50  J= 1,NF
      IF ( K - 4 ) 49, 50, 50
   49 DFCTS(J,K) = 0.
   50 FCTS(J,K) = 0.
      READ(5,2008) OPT2, KPRINT
 2008 FORMAT(2I4)
      READ(5,2000) T, DT, TMAX, TIN, TEX, INT, OPT1, TMP, TMPS
 2000 FORMAT ( D10.5, D10.8, D10.5, 2F10.5, 2I3, 2D6.1 )
      DTM = DT
      INTS = INT
      READ(5,2003) PPL, PDEX, PTXEX, RG, VD, Y1, Y2
      READ(5,2004) PTXG, DPTXG,
     &    PDAG, DPDAG, VTX, VMAX, FCTN
      READ(5,2001)  VMIN, FCTN2, Y3, Y4, TB, PB
 2001 FORMAT ( 7F10.4 )
 2003 FORMAT ( 5D10.4, 2F10.4 )
 2004 FORMAT ( F10.4, D10.4, F10.4, 2D10.4, 2F10.4 )
      READ(5,2010) V
      READ(5,2010) ( P(J),  J= 1,12 )
      READ(5,2002) ( ( PRCT(I,J), I = 1, 12 ), J = 1, 4 )
 2002 FORMAT ( 6F10.4 )
      READ(5,2001) ( ( B(I,J), I=1,7), J= 2,12 )
      READ(5,2001) A
      DO 5  I= 1,8
      A(4,I) = A(4,I) - .1D-8
      A(3,I) = A(3,I) + .1D-8
    5 A(6,I) = A(1,I) * HP
 2005 FORMAT ( 3D10.6, I3 )
      READ(5,2006) (W(J), J=1,12)
      READ(5,2009) (FS(J), J=1,12), F(1)
 2006 FORMAT ( 6E10.3 )
      READ(5,2005) PHI1, PHI2, GF, FBOPT
 2010 FORMAT ( 6D10.4 )
 2009 FORMAT ( 7D10.4 )
      WRITE(6,2007)
 2007 FORMAT(1H1)
      WRITE(6,4001) T,DT,TIN,TEX,INT,OPT1,TMP,TMPS,PPL,PDEX,PTXEX,RG,
     &    (J,J=1,6),VD,Y1,Y2,PTXG,(V(J),J=1,6),(J,J=1,12),(P(J),J=1,12),
     &    (J,J=1,12),(J,(PRCT(I,J),I=1,12),J=1,4),(J,J=1,7),
     &    ((B(I,J),I=1,7),J=2,12),(J,J=1,7),A,
     &    (W(J), J=2,12), PHI1, PHI2, GF, Y3, Y4, TB, VMAX, FCTN,
     &    VMIN, FCTN2
 4001 FORMAT(1H ,10X,'T',8X,'DT',7X,'TIN',7X,'TEX',7X,'INT',7X,'OPT1',
```

ready

*

```
     &     6X,'TMP',7X,'TMPS',6X,'PPL',6X,'PDEX ',5X,'PTXEX',6X,'RG',//,
     &     6X, 2E10.3, 2(F7.3,3X),2(2X,I5,3X),
     &     6(F7.3,3X),///,9X,'VD',8X,'Y1',8X,'Y2',7X,'PTXG',7X,
     &     6('V(',I1,')',6X),//,6X,10(F7.3,3X),///,8X,12('P(',I2,')',5X),
     &     //,8X,12(F7.3,3X),///,2X,'PRCT',3X,12(I2,8X),//,4(3X,I1,2X,
     &     12(F7.5,3X),//),/,5X,'B',8X,7(I1,9X),//,3X,'(12)',4X,7(F7.3,
     &     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(F7.3,3X),//,3X,'(37)
     &     ',7(F7.3,3X),//,3X,'(48)',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),
     &     //,2X,'(612)',4X,7(F7.3,3X),///,7X,7('A(',I1,')',6X),//,8(5X,
     &     7(F8.4,2X),//),/,7X,'W12',8X,'W23',8X,'W24',8X,'W35',8X,'W46',
     &     8X,'W37',8X,'W48',8X,'W59',7X,'W610',7X,'W511',7X,'W612',//,
     &     5X,11(E10.4,1X),///,7X,'PHI1',7X,'PHI2',,7X,'GF',8X,'Y3',8X,
     &     'Y4',8X,'TB',6X,'VMAX',6X,'FCTN',6X,'VMIN',6X,'FCTN2'//
     &     5X,F7.3,3X,F7.3,3X,F7.2,7(3X,F7.4),/// )
  51 CONTINUE
      DO 55  I = 1,4
  55 S(I) = 0.0
       TMAX1 = TMAX + DT / 2.
       G = TMP * RG
       GS = TMPS * RG
       G1 = 0.84791 / ( G*G )
       G2 = 0.018015 / G
       G3 = 47.067/ G
       G4 = 1./ G
       PHI1 = PHI1 * PIE / 180.
       PHI2 = PHI2 * PIE / 180.
C
       CALL PGRAV ( PBS, PGAB, PG, DPBS, DPGAB, DPG )
       PTX = PTXG * GT
       PD = PDAG * GD
C
       FRCP = FRC2 ( VA, VTX, PD, PTX )
C
       IF ( ABS( VMIN ). LT. 0.0001 ) GO TO 57
C
       CALL PMAX ( FRCP, VMIN, PG, PBS, GT, GD, PGAB,
      &             FCTN2, VTX, VD, PPL, VA )
C
       PTXEXS = PTXIN
       PDEXS = PDIN
   57 CONTINUE
C
       CALL PMAX ( FRCP, VMAX, PG, PBS, GT, GD, PGAB,
      &             FCTN, VTX, VD, PPL, VA )
C
       IF ( TIN. LE. 0. )  CALL SUB3( V )
       VSPR = V(1)
       DO 60  J = 1,6
```

ready

*

```
C          K = J+6
      PSAVE(J) = P(J)
      PALV(J) = P(K)
   60 VALV(J) = V(K)
      PPLC(1) = PPL + PG(1)
      PPLC(2) = PPL + PG(3)
      PPLC(3) = PPL + PG(5)
      WRITE (6,998) GR, PPL, PPLC(2), VLUNG, VTX, VD, VALV
  998 FORMAT(1H ,4X,'GR', 6X, 'PPL', 7X, 'PPLAVG', 6X, 'VLUNG', 8X,
     &    'VTX', 9X, 'VD', 8X, 'VALV', //,
     &    1X, F5.1, 1X, 2F12.6, 9F11.7, /// )
      VLDIF = VLUNG - VTX - VD
C
C      IF ( FBOPT. EQ. 11 ) CALL PCHK( VMIN, VMAX, FRCP, FCTN,
C     &    FCTN2, VLDIF, PGAB )
C      IF ( FBOPT. EQ. 10 ) CALL FCHK( T, FSPR, DT )
C
      TOTN(1) = P(1)*V(1)/GS
      CW(1) = 31.8 * DEXP( 0.018 * ( P(1) -760. ) / GS) / GS
      NW(1) = CW(1)*V(1)
      TN(1) = TOTN(1)-NW(1)
      DO 70 I=1, NG
      N(1,I) = TN(1) * PRCT(1,I)
   70 C(1,I) = N(1,I) / V(1)
      PV(1) = P(1) * V(1) / ( TOTN(1) * GS)
      DO 80 J=2,12
      TOTN(J) = P(J)*V(J)/G
      CW(J) = G3 * DEXP( G2 * (P(J)-760.) )
      NW(J) =CW(J)*V(J)
      TN(J) = TOTN(J)-NW(J)
      PV(J) = P(J) * V(J) / ( TOTN(J) * G )
      DO 80 I=1, NG
      N(J,I) = TN(J)*PRCT(J,I)
   80 C(J,I) = N(J,I)/V(J)
 3011 FORMAT( //,1H ,4(5X, 6(E17.11), //) )
      DO 1580  I = 1,NG
 1580 ASUM1(I) = 0.
      AWSUM1 = 0.
      DO 1585  J= 1,12
      AWSUM1 = AWSUM1 + NW(J)
      DO 1585  I= 1,NG
 1585 ASUM1(I) = ASUM1(I) + N(J,I)
      HDT = DT/4.
      CALL FLOW ( 1, 2 )
      CALL FLOW ( 2, 3 )
      CALL FLOW ( 2, 4 )
      CALL FLOW ( 3, 5 )
      CALL FLOW ( 4, 6 )
      CALL FLOW ( 3, 7 )
      CALL FLOW ( 4, 8 )

ready

*
```

```
      CALL FLOW ( 5, 9 )
      CALL FLOW ( 6, 10 )
      CALL FLOW ( 5, 11 )
      CALL FLOW ( 6, 12 )
   90 KNT = 0.
      DO 100  K = 1, NF
      FCTS(K,4) = FCT(K)
  100 TE(K) = 0.
      WRITE(6,2007)
      RETURN
      ENTRY LMN1(LDUMMY)
C
C     $$$ RUNGE-KUTTA INTEGRATION SECTION $$$
C
      M = 1
  110 CALL DERIVS( OPT1, 1, PSAVE  )
      IF( T. LT. 1.D-6 ) GO TO 151
  111 IF( M. EQ. 5 ) GO TO 130
  120 CALL RUNGE( NF, FCT, DFCT, T, DT, M, HDT )
      DTM = HDT * 2.
      CALL PARMS2( VA )
      M = M + 1
      GO TO 110
  130 KNT = KNT + 1
      M = 1
      K = 4 - KNT
      HDT = DT/2.
      IF( K. EQ. 1 ) GO TO 150
      DO 140  J= 1,NF
      FCTS(J,K) = FCT(J)
  140 DFCTS(J,K) = DFCT(J)
  150 DO 155  J= 2,12
  155 FS(J) = F(J)
      DO 154 JPS = 2,6
  154 PSAVE(JPS) = P(JPS)
      INT = INTS
C     IF ( T. LT. .15 ) INT = INTS / 10
      IF ( INT. EQ. 0 ) INT = 1
  152 IF ( KNT/INT * INT .NE. KNT )  GO TO 190
  151 CONTINUE
      CALL SUB2( ASUM, ASUM1, AWSUM, AWSUM1, RES, PV, RTOT, PPLC )
      IF( KPRINT. NE. 1 ) GO TO 153
      WRITE(6,3006) KNT
      WRITE(6,3000) T, VTX, DVTX, VD, VSPR, DVSPR
      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=1,6),I=1,NG )
      WRITE(6,3002) NW, DNW, CW, (DCW(J),J=1,6), (PCALC(J),J=1,6),
     &    P, DP, VALV, DVALV, F, DCWDP
      WRITE(6, 1) PELALV, S, PELDA, PELTX, PTX,
     &    PD, PG, PGAB, PPL, PPLC, VLUNG, RTOT, ASUM(1), ASUM(2), AWSUM

ready

*
```

```
      WRITE(6,3009) DNT, CTCALC
  153 CONTINUE
      WRITE(12,3100) T, VSPR, PPL, PPLC, RTOT, VLUNG, VTX, VD,
     &                   PTX, PD, VALV, PELALV, PELTX, PELDA
 3100 FORMAT( 10E14.7 )
      WRITE(12,3100) P, F, RES, RA, NW, CW, CT,
     &                  ( (N(J,I),J=1,12), (C(J,I),J=1,12), I=1,NG )
 3000 FORMAT( 1H , 4X, 'T=', 6E13.6 )
 3001 FORMAT( 1H ,    3( 2( 6E13.6,/),   ) 6E13.6 )
 3002 FORMAT( 1H,    7( 2( 6E13.6, /)   ), 2( 12E10.4, /)   )
 3009 FORMAT( 1H , 2( /, 5X, 6E13.6 ) )
 3005 FORMAT(  1X, 'KNT = ', I6     )
    1 FORMAT( 1H , 10E13.6 )
      IF( T. LT. 1.D-6 )  GO TO 111
  190 CONTINUE
      RETURN
      ENTRY LMN2(LDUMMY)
      IF ( KNT. LT. 3 ) GO TO 120
      IF(OPT2. LT. 1 )  GO TO 1000
C
C     $$$ CALL HAMING PREDICTOR OR CORRECTOR $$$
C
      TTESTO = TTEST
      DTM = HDT * 2.
      MTEST = HAMNG1( NF, FCTS, DFCTS, T, DT, TE )
  500 MTEST = HAMING( NF, FCTS, DFCTS, T, DT, TE )
      CALL PARMS2( VA )
      IF ( T. GT. TIN - .001 )  PDEX = PDEXS
      IF ( T. GT. TIN - .001 )  PTXEX = PTXEXS
      CALL DERIVS( OPT1, MTEST, PSAVE )
      IF( MTEST. EQ. 1 )  GO TO 500
      KNT = KNT + 1
      DO 506  J= 2,12
  506 FS(J) = F(J)
      FSPRS = FS(2)
      DO 501 JPS = 2, 6
  501 PSAVE(JPS) = P(JPS)
      IF ( P(9). LT. 650.0. OR . P(9). GT. 850.0 ) GO TO 502
      IF ( P(4). LT. 650.0. OR . P(4). GT. 850.0 ) GO TO 502
      GO TO 503
  502 END FILE 12
      UNSTAB = 10.
      KPRINT = 1
      GO TO 510
  503 CONTINUE
      INT = INTS
C     IF ( T. GT. T0. AND. T. LT. T0 + .15 ) INT = INTS / 10
C     IF ( T. GT. T2. AND. T. LT. T2 + .15 ) INT = INTS / 10
      IF ( INT. EQ. 0 ) INT = 1
      IF (KNT.LT.1000) GO TO 600

ready

*
```

```
  505 IF ( KNT/INT * INT . NE. KNT )  GO TO 600
  510 CONTINUE
      CALL SUB2( ASUM, ASUM1, AWSUM, AWSUM1, RES, PV, RTOT, PPLC )
      IF( KPRINT. NE. 1 ) GO TO 515
      WRITE(6,3006) KNT
      WRITE(6,3000) T, VTX, DVTX, VD, VSPR, DVSPR
      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=1,6),I=1,NG )
      WRITE(6,3002) NW, DNW, CW, (DCW(J),J=1,6), (PCALC(J),J=1,6),
     &     P, DP, VALV, DVALV, F, DCWDP
      WRITE(6, 1) PELALV, S, PELDA, PELTX, PTX,
     &     PD, PG, PGAB, PPL, PPLC, VLUNG, RTOT, ASUM(1), ASUM(2), AWSUM
      WRITE(6,3009) DNT, CTCALC
      IF( UNSTAB. GT. 5. ) STOP 2
  515 CONTINUE
      WRITE(12,3100) T, VSPR, PPL, PPLC, RTOT, VLUNG, VTX, VD,
     &                   PTX, PD, VALV, PELALV, PELTX, PELDA
      WRITE(12,3100) P, F, RES, RA, NW, CW, CT,
     &                  ( (N(J,I),J=1,12), (C(J,I),J=1,12), I=1,NG )
  600 CONTINUE
  625 CONTINUE
      TTESTO = TTEST
      RETURN
      ENTRY LMN3(LDUMMY)
      IF ( T - TMAX1 )  500, 1000, 1000
 1000 CONTINUE
      RETURN
      ENTRY LMN4(LDUMMY)
      T = 0.0
      WRITE(12,3100) (T,I=1,10)
      END FILE 12
      WRITE (6,9999) T
 9999 FORMAT (' LUNG PROGRAM HAS ENDED AT TIME=',F10.5)
      RETURN
      END

ready

*
```

```
      SUBROUTINE BCMAIN(LDUMMY)
      INTEGER HAMING
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8  LAMBO2,LAMBC2,LAMBCO,LAMBX,LCO2,LCC2,LCCO,LCX,METO2,METC2
      COMMON /BCM/ TB,TMAXB
      COMMON /BCMBLK/ R,FKDA,PP,VO,A2DM2,A2DM3,A2DM4,A2DM12,A2DM13,
     1A2DM14,H,RHO,G,RIN,P,PC
      COMMON /RP/ RP3,RP13
      COMMON /SK/ TAU,PO,P1
      COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GPO2,CTO2,GPC2,CTC2
      COMMON /CMN1T/GPCO,CTCO,GPX,CTX
      COMMON /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGPO2,DCTO2,
     1DGPC2,DCTC2
      COMMON /CMN2T/DGPCO,DCTCO,DGPX,DCTX
      COMMON /A1/LVALV2,LVALV3,LVLV12,LVLV13,LVLV23,
     1PST2,PST12,TBS2,TBS1
      COMMON /KNT/KNTB
      COMMON /PHT/KTB,KTDEL
      COMMON /VFLT/VFLOW,VFLIN
      COMMON /BC/LAMBO2,LAMBC2,LAMBCO,LAMBX,AO2,AC2,ACO,AX,HEMAT,DIFO2,
     1DIFC2,DIFCO,DIFX,CPH,CSATO2,CSATC2,METO2,METC2
      COMMON /BCONLY/LCO2,LCC2,LCCO,LCX
      EQUIVALENCE (V,FCTB,FCTSB),(DV,DFCTB,DFCTSB)
      DIMENSION R(24),FKDA(24),PP(24),V(24),VO(24),H(24),DV(24),
     1VFLOW(24),VFLIN(18),RIN(18),PC(24),P(24)
      DIMENSION RK(24),RINK(18)
      DIMENSION TAU(24),PO(24),P1(24)
      DIMENSION FCTB(86),DFCTB(86),FCTSB(86,4),DFCTSB(86,3),TEB(86)
      DIMENSION GPO2(24),GPC2(24),GPCO(48),GPX(1,24),CTO2(4),CTC2(4),
     1CTCO(4),CTX(1,4),DIFO2(10),DIFC2(10),DIFCO(10),DIFX(10),CPH(24),
     2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24 ),DGPO2(24),
     3DGPC2(24),DGPCO(48),DGPX(1,24),AX(1)
      DIMENSION DCTO2(4),DCTC2(4),DCTCO(4),DCTX(1,4),METO2(4),METC2(4)
C  THIS DIMENSION STATEMENT IS ONLY IN BMAIN
C FCTB(V=24,VP=6,C=24,CT=4)
C ADD 28 LOCATIONS FOR EAC NEW GAS TRANSPORTED*.  THIS PROGRAM IS
C PRESENTLY SET UP FOR 24 CHAMBERS AND 1 GAS..NFB=NC+6+NG(NC+4)=58
C  *EXCEPT FOR CO, WHICH REQUIRES TWICE AS MANY LOCATIONS (2*28=56).
      NFB=86
      DO 50 K=1,4
      DO 50 J=1,NFB
      IF (K-4) 49,50,50
   49 DFCTSB(J,K)=0
   50 FCTSB(J,K)=0
      READ(5,1010) (R(N),N=1,24),(FKDA(N),N=1,24),(PP(N),N=1,24)
      READ(5,1010) (TAU(N),N=1,24),(PO(N),N=1,24)
 1010 FORMAT(8(E8.1,1X))
      READ(5,1020) (V(N),N=1,24),(VO(N),N=1,24),A2DM2,A2DM3,A2DM4,A2DM12
     1,A2DM13,A2DM14
 1020 FORMAT(8(F8.3,1X))

ready

*
```

```
      READ(5,1030) (H(N),N=1,24),RHO,G
 1030 FORMAT(2(12(F5.1,1X)/),F3.1,F5.1)
      READ(5,1040) (RIN(N),N=5,11),(RIN(N),N=15,18),VP2,VP3,VP4,VP12,
     1VP13,VP14
 1040 FORMAT(7E9.1/4E9.1/6E9.1)
      READ (5,201) GPO2,GPC2,GPCO,GPX
  201 FORMAT (8E9.2/8E9.2/8E9.2)
      READ(5,201) LAMBO2,LAMBC2,LAMBCO,LAMBX,AO2,AC2,ACO,AX,CTO2,CTC2,
     1CTCO,CTX
C   CT STAND FOR THE PARTIAL PRESSUR IN THE TISSUE, NOT THE CONC.
      READ(5,203) LCO2,LCC2,LCCO,LCX
      READ (5,204) DIFO2,DIFC2,DIFCO,DIFX
      READ (5,205)  METO2,METC2
  204 FORMAT (6E9.2/2E9.2)
  203 FORMAT (6E9.2)
  205 FORMAT (4E9.2)
      READ(5,1050) DTB,TMAXB
 1050 FORMAT (F8.1,1X,F6.1)
      READ (5,1200)  KNTP,LOPPR,LOPP1,LOPR,LOPGS
 1200 FORMAT (I2,1X,I1,1X,I1,I2,I2)
C  LOPPR=1 IF RUNGE IS TO BE PRINTED,LOPPR=0 IF NOT
C  KNTP TELLS HOW OFTEN (NO. OF TIME STEPS) TO PRINT
C   LOPP1 CONTROLES LIN(O) OR NONLINEAR COMPLIANCE
C   SEE SUB RESIST FOR EXPLAINATION OF LOPR
C   SEE SUB ECDRIVS FOR EXPLAINATION OF LOPGS
      WRITE(6,2010)
 2010 FORMAT(10X,12H(INPUT DATA)/)
      WRITE (6,2001) DTB,TMAXB
 2001 FORMAT(10H DTB,TMAXB,5X,2F8.4)
      WRITE(6,2040) (RIN(N),N=1,18)
 2040 FORMAT(10H   RIN(N) ,5X,8(E10.3,2X)/15X,8(E10.3,2X)/15X,2(E10.3,
     12X)/)
      WRITE(6,2020) R,FKDA,PP
 2020 FORMAT(10H     R(N) ,5X,8(E10.3,2X)/2(15X,8(E10.3,2X)/)///
     110H  FKDA(N) ,5X,8(E10.3,2X)/2(15X,8(E10.3,2X)/)///
     210H    PP(N) ,5X,8(E10.3,2X)/2(15X,8(E10.3,2X)/)///)
      WRITE(6,2021) TAU,PO
 2021 FORMAT(10H   TAU(N) ,5X,8(E10.3,2X)/2(15X,8(E10.3,2X)/)///
     110H    PO(N) ,5X,8(E10.3,2X)/2(15X,8(E10.3,2X)/)///)
      WRITE(6,2030) V,VO,A2DM2,A2DM3,A2DM4,A2DM12,A2DM13,A2DM14
 2030 FORMAT(10H     V(N) ,5X,8(F8.3,2X)/2(15X,8(F8.3,2X)/)///
     110H    VO(N) ,5X,8(F8.3,2X)/2(15X,8(F8.3,2X)/)///
     210H     A2DM ,5X,8(F8.3,2X)///)
      WRITE (6,2031) H
 2031 FORMAT (10H   H(N)   ,5X,8E12.3/2(15X,8E12.3/)///)
      WRITE (6,2032) RHO,G
 2032 FORMAT ('  RHO,G  =  ',2F8.3)
      WRITE (6,2033) VP2,VP3,VP4,VP12,VP13,VP14
 2033 FORMAT ('    VP    ',6E12.3///)
       WRITE (6,2041) KNTP,LOPPR,LOPP1,LOPR,LOPGS
```

ready

*

```
 2041 FORMAT ('  OPTIONS ',5X,I2,1X,I2,1X,I2,1X,I2,1X,I2)
      WRITE (6,208) GPO2,GPC2,GPCO,GPX
  208 FORMAT (15H GPO2,C2,CO,PX  ,8(E10.3,2X)/14(15X,8E12.3/))
C   WARNING...FORMAT MUST BE CHANGED IF NGX.GT.1
      WRITE (6,209) LAMBO2,LAMBC2,LAMBCO,LAMBX,AO2,AC2,ACO,AX
  209 FORMAT (' LAMB,A...O2,C2,CO,X ',/,15X,8E12.3/)
      WRITE (6,210) CTO2,CTC2,CTCO,CTX
  210 FORMAT ('  CTO2,C2,CO,X ',/,2(15X,8E12.3/))
      WRITE (6,211) LCO2,LCC2,LCCO,LCX
  211 FORMAT ('  LCO2,C2,CO,X ',/4(15X,6E12.3/))
      WRITE (6,212) DIFO2,DIFC2,DIFCO,DIFX
  212 FORMAT (' DIFO2,C2,CO,X ',/,4(10E11.3/))
      WRITE (6,216) METO2,METC2
  216 FORMAT (' METO2,METC2 ',4E12.3,10X,4E12.3)
      RETURN
      ENTRY BCMN1(LDUMMY)
      WRITE(6,2050)
 2050 FORMAT(10X,13H(OUTPUT DATA)/)
      WRITE (6,3002) PP,LCO2,LCC2
      SUMVI=0
      DO 444 N=1,24
  444 SUMVI=SUMVI+V(N)
      TB=0
      KNTB=0
      MTEST=0
      RIN(4)=.0136
      RIN(14)=.068
      RP3=3.E-07
      RP13=1.5E-06
      HEMAT=.45
C   INITIAL VALUES FOR PHEART SET HERE
      PSTO1=.26
      PSTO2=.43
      PSTO3=.26
      PSTO4=.43
      PSDUR1=.35
      PSDUR2=.53
      PSDUR3=.35
      PSDUR4=.53
      PSAMP1=5.
      PSAMP2=20.
      PSAMP3=15.
      PSAMP4=110.
      PSRT=70
      CALL CHEART(PSDUR1,PSAMP1,PSRT,Z1,PSEND1,PSTON1,PSTO1)
      CALL CHEART(PSDUR2,PSAMP2,PSRT,Z2,PSEND2,PSTON2,PSTO2)
      CALL CHEART(PSDUR3,PSAMP3,PSRT,Z3,PSEND3,PSTON3,PSTO3)
      CALL CHEART(PSDUR4,PSAMP4,PSRT,Z4,PSEND4,PSTON4,PSTO4)
C
      CALL PHEART(PSDUR1,PSAMP1,PSRT,TB,PP(2),PSTO1,Z1,PSEND1,PSTON1)

ready

*
```

```
      CALL PHEART(PSDUR2,PSAMP2,PSRT,TB,PP(3),PSTO2,Z2,PSEND2,PSTON2)
      CALL PHEART(PSDUR3,PSAMP3,PSRT,TB,PP(12),PSTO3,Z3,PSEND3,PSTON3)
      CALL PHEART(PSDUR4,PSAMP4,PSRT,TB,PP(13),PSTO4,Z4,PSEND4,PSTON4)
    2 CONTINUE
C
      CALL CRES(LOPR,RK,RINK)
      CALL RESIST(LOPR,RK,RINK)
C   THE FOLLOWING SETS PC FOR THE HEART.  IF ANY VALVE IS OPEN,INITIAL
C          CONDITIONS SHOULD GIVE THE CONNECTED CHAMBERS THE SAME
C          INITIAL PRESSURE FOR NON-INERTIAL FLUID
      CALL SKON(LOPP1,TB)
      P(2)=PP(2)+P1(2)
      P(3)=PP(3)+P1(3)
      P(4)=PP(4)+P1(4)
      P(12)=PP(12)+P1(12)
      P(13)=PP(13)+P1(13)
      P(14)=PP(14)+P1(14)
      CALL BDRIVS(LOPP1,0,TB)
      CALL BCDRIV(LOPGS)
C
      IF (LOPP1.EQ.2) GO TO 998
C
      DO 60 K=1,NFB
      FCTSB(K,4)=FCTB(K)
   60 TEB(K)=0
   65 M=1
C THE LAST VARIABLE IN RUNGE IS DUMMY
   66 CALL RUNGE (NFB,FCTB,DFCTB,TB,DTB,M,DUM)
      CALL PHEART(PSDUR1,PSAMP1,PSRT,TB,PP(2),PSTO1,Z1,PSEND1,PSTON1)
      CALL PHEART(PSDUR2,PSAMP2,PSRT,TB,PP(3),PSTO2,Z2,PSEND2,PSTON2)
      CALL PHEART(PSDUR3,PSAMP3,PSRT,TB,PP(12),PSTO3,Z3,PSEND3,PSTON3)
      CALL PHEART(PSDUR4,PSAMP4,PSRT,TB,PP(13),PSTO4,Z4,PSEND4,PSTON4)
   22 CONTINUE
C  SEE ABOVE
      CALL RESIST(LOPR,RK,RINK)
      CALL BDRIVS(LOPP1,0,TB)
      CALL BCDRIV(LOPGS)
   13 CONTINUE
      M=M+1
      IF (M-5) 66,67,67
   67 KNTB=KNTB+1
      SUMV=0
      DO 155 N=1,24
  155 SUMV=SUMV+V(N)
      DELV=SUMV-SUMVI
      DO 158 N=1,24
  156 IF (V(N)) 157,158,158
  157 WRITE (6,3009) N,V(N)
 3009 FORMAT ('   CHAMBER NO. ',I2,' HAS A  NEG VOLUME =  ',E12.6)
  158 CONTINUE
```

ready

*

```
C   THE FOLLOWING IS A PRINT OPTION
       IF (KNTB/KNTP*KNTP.EQ.KNTB) GO TO 998
    68 CONTINUE
       RETURN
       ENTRY BCMN2(LDUMMY)
       IF (KNTB-3) 70,72,72
    70 K= 4-KNTB
       DO 71 J=1,NFB
       FCTSB(J,K)=FCTB(J)
    71 DFCTSB(J,K)=DFCTB(J)
       GO TO 65
    72 CONTINUE
       MTEST=HAMNG1(NFB,FCTSB,DFCTSB,TB,DTB,TEB)
    89 CONTINUE
    90 MTEST=HAMING(NFB,FCTSB,DFCTSB,TB,DTB,TEB)
       IF(MTEST.EQ.1) GO TO 32
       CALL PHEART(PSDUR1,PSAMP1,PSRT,TB,PP(2),PSTO1,Z1,PSEND1,PSTON1)
       CALL PHEART(PSDUR2,PSAMP2,PSRT,TB,PP(3),PSTO2,Z2,PSEND2,PSTON2)
       CALL PHEART(PSDUR3,PSAMP3,PSRT,TB,PP(12),PSTO3,Z3,PSEND3,PSTON3)
       CALL PHEART(PSDUR4,PSAMP4,PSRT,TB,PP(13),PSTO4,Z4,PSEND4,PSTON4)
    32 CONTINUE
C   SEE ABOVE
       CALL RESIST(LOPR,RK,RINK)
       CALL BDRIVS(LOPP1,MTEST,TB)
       CALL BCDRIV(LOPGS)
       IF (MTEST.EQ.1) GO TO 89
       SUMV=0
       DO 555 N=1,24
   555 SUMV=SUMV+V(N)
       DELV=SUMV-SUMVI
       DO 558 N=1,24
   556 IF (V(N)) 557,558,558
   557 WRITE (6,3002) N,V(N)
   558 CONTINUE
       KNTB=KNTB+1
C   THIS IS THE PLACE TO TEST RESULTS TO SEE IF YOU WISH TO CONTINUE
       IF (KNTB.LT.1000) GO TO 91
       IF (KNTB/KNTP*KNTP.EQ.KNTB) GO TO 999
    91 CONTINUE
       RETURN
       ENTRY BCMN3(LDUMMY)
       IF (TB-TMAXB) 90, 100,100
   998 CONTINUE
       WRITE (4,3000) KNTB,KNTP,TB,DTB,LOPPR,LOPP1,LOPR,LOPGS,PP(2),
      1PP(3),PP(12),PP(13),LVALV2,LVALV3,LVLV12,LVLV13,LVLV23
       WRITE (4,3002) VP2,VP3,VP4,VP12,VP13,VP14,DVP2,DVP3,DVP4,DVP12,
      1DVP13,DVP14
       WRITE (4,3002) P,V,DV,VFLOW
       WRITE (4,3002) VFLIN,RIN,R
       WRITE (6,3000) KNTB,KNTP,TB,DTB,LOPPR,LOPP1,LOPR,LOPGS,PP(2),
```

ready

*

```
     1PP(3),PP(12),PP(13),LVALV2,LVALV3,LVLV12,LVLV13,LVLV23
      WRITE (6,3002) VP2,VP3,VP4,VP12,VP13,VP14,DVP2,DVP3,DVP4,DVP12,
     1DVP13,DVP14
      WRITE (6,3002) P,V,DV,VFLOW
      WRITE (6,3002) VFLIN,RIN,R
      WRITE (6,3002) GPO2,GPC2,CSATO2,CPH,DGPO2,DGPC2
      WRITE (6,3002) CTO2,DCTO2,CTC2,DCTC2
      WRITE (4,3002) GPO2,GPC2,CSATO2,CPH,DGPO2,DGPC2
      WRITE (4,3002) CTO2,DCTO2,CTC2,DCTC2
 3000 FORMAT (1X,2I6,2F10.4,4I6,4F10.4,5I3)
 3002 FORMAT (1X,12F10.4)
C
      IF (LOPP1.EQ.2) GO TO 100
C
      GO TO 68
  999 CONTINUE
      WRITE (4,3000) KNTB,KNTP,TB,DTB,LOPPR,LOPP1,LOPR,LOPGS,PP(2),
     1PP(3),PP(12),PP(13),LVALV2,LVALV3,LVLV12,LVLV13,LVLV23
      WRITE (4,3002) VP2,VP3,VP4,VP12,VP13,VP14,DVP2,DVP3,DVP4,DVP12,
     1DVP13,DVP14
      WRITE (4,3002) P,V,DV,VFLOW
      WRITE (4,3002) VFLIN,RIN,R
      WRITE (6,3000) KNTB,KNTP,TB,DTB,LOPPR,LOPP1,LOPR,LOPGS,PP(2),
     1PP(3),PP(12),PP(13),LVALV2,LVALV3,LVLV12,LVLV13,LVLV23
      WRITE (6,3002) VP2,VP3,VP4,VP12,VP13,VP14,DVP2,DVP3,DVP4,DVP12,
     1DVP13,DVP14
      WRITE (6,3002) P,V,DV,VFLOW
      WRITE (6,3002) VFLIN,RIN,R
      WRITE (6,3002) GPO2,GPC2,CSATO2,CPH,DGPO2,DGPC2
      WRITE (6,3002) CTO2,DCTO2,CTC2,DCTC2
      WRITE (4,3002) GPO2,GPC2,CSATO2,CPH,DGPO2,DGPC2
      WRITE (4,3002) CTO2,DCTO2,CTC2,DCTC2
      GO TO 91
  100 CONTINUE
      WRITE (6,9999) TB
 9999 FORMAT (' ...BLOOD PROGRAM HAS ENDED AT TIME=',F10.5)
      RETURN
      END
      SUBROUTINE BDRIVS (LOPP1,MTEST,TB)
      IMPLICIT REAL*8 (A-H,O-Z)
C  NEW FOR ABMNRC
      COMMON /RP/RP3,RP13
C
      COMMON /BCMBLK/ R,FKDA,PP,VO,A2DM2,A2DM3,A2DM4,A2DM12,A2DM13,
     1A2DM14,H,RHO,G,RIN,P,PC
      COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GPO2,CTO2,GPC2,CTC2
      COMMON /CMN1T/GPCO,CTCO,GPX,CTX
      COMMON /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGPO2,DCTO2,
     1DGPC2,DCTC2
      COMMON /CMN2T/DGPCO,DCTCO,DGPX,DCTX
```

ready

*

```
      COMMON /A1/LVALV2,LVALV3,LVLV12,LVLV13,LVLV23,PS12,PST12,TBS2,TBS12
      COMMON /SK/ TAU,PO,P1
      COMMON /VFLT/VFLOW,VFLIN
      DIMENSION R(24),FKDA(24),PP(24),V(24),VO(24),H(24),DV(24),
     1VFLOW(24),VFLIN(18),RIN(18),PC(24),P(24)
      DIMENSION TAU(24),PO(24),P1(24)
      DIMENSION GPO2(24),GPC2(24),GPCO(48),GPX(1,24),CTO2(4),CTC2(4),
     1CTCO(4),CTX(1,4),DIFO2(10),DIFC2(10),DIFCO(10),DIFX(10),CPH(24),
     2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24 ),DGPO2(24),
     3DGPC2(24),DGPCO(48),DGPX(1,24),AX(1)
      DIMENSION DCTO2(4),DCTC2(4),DCTCO(4),DCTX(1,4),METO2(4),METC2(4)
      CALL SKON(LOPP1,TB)
      P(1)=PP(1)+P1(1)
      DO 10 N=5,11
   10 P(N)=PP(N)+P1(N)
      DO 12 N=15,24
   12 P(N)=PP(N)+P1(N)
      DO 20 N=1,24
   20 PC(N)=P(N)+RHO*G*H(N)
      CALL PCHART(2,LVALV2,LVALV3,MTEST,PST2,TBS2,TB)
      CALL PCHART (12,LVLV12,LVLV13,MTEST,PST12,TBS12,TB)
      VFLOW(1)=(PC(1)-PC(2))/R(1)
      IF (LVALV2) 113,111,112
  111 VFLOW(2)=(PC(2)-PC(3))/R(2)
      GO TO 120
  112 VFLOW(2)=0
      GO TO 120
  113 WRITE (6,1113)
 1113 FORMAT ('LVALV2 IS NEG .. NOT POSSIBLE')
  120 IF (LVALV3) 123,121,122
  121 VFLOW(3)=-VP3
      DVP3=(-PC(3)+PC(4)+RP3*VP3**2-RIN(3)*VP3)*A2DM3
      DV(3)=VP3
      GO TO 130
  122 VFLOW(3)=0
      VP3=0.0
      DVP3=0.0
      DV(3)=VFLOW(2)
      GO TO 130
  123 WRITE (6,1123)
 1123 FORMAT ('LVALV3 IS NEG .. NOT POSSIBLE')
  130 VFLOW(4)=0
      DO 13 N=5,11
      VFLIN(N)=(PC(4)-PC(N))/RIN(N)
      VFLOW(N)=(PC(N)-PC(11))/R(N)
   13 VFLOW(4)=VFLOW(4)+VFLIN(N)
C  THE FOLOWING IS A CORRECTION TO THE ZERO CALC. IN THE DO FOR
C  VFLOW(11) ABOVE
      VFLOW(11)=(PC(11)-PC(12))/R(11)
      IF (LVLV12) 133,131,132

ready

*
```

```
 131 VFLOW(12)=(PC(12)-PC(13))/R(12)
     GO TO 140
 132 VFLOW(12)=0
     GO TO 140
 133 WRITE (6,1133)
1133 FORMAT ('LVLV12 IS NEG .. NOT POSSIBLE')
 140 IF (LVLV13) 143,141,142
 141 VFLOW(13)=-VP13
     DVP13=(-PC(13)+PC(14)+RP13*VP13**2-RIN(13)*VP13)*A2DM13
     DV(13)=VP13
     GO TO 150
 142 VFLOW(13)=0
     VP13=0.0
     DVP13=0.0
     DV(13)=VFLOW(12)
     GO TO 150
 143 WRITE (6,1143)
1143 FORMAT ('LVLV13 IS NEG .. NOT POSSIBLE')
 150 VFLOW(14)=0
     DO 14 N=15,16
     VFLIN(N)=(PC(14)-PC(N))/RIN(N)
     VFLOW(N)=(PC(N)-PC(N+4))/R(N)
  14 VFLOW(14)=VFLOW(14)+VFLIN(N)
C THE FOLLOWING TESTS VALVE 23
     IF (MTEST.EQ.1) GO TO 153
     IF (PC(19)-PC(23)) 152,151,151
 151 VFLOW(19)=(PC(19)-PC(23))/R(19)
     LVLV23=0
     GO TO 160
 152 VFLOW(19)=0
     LVLV23=1
     GO TO 160
C   VALVE TEST FROM LAST BDRIVS EXECUTED IS USED
 153 IF (LVLV23) 154,155,156
 154 WRITE (6,1144)
1144 FORMAT ('LVLV23 IS NEG .. NOT POSSIBLE')
 155 VFLOW(19)=(PC(19)-PC(23))/R(19)
     GO TO 157
 156 VFLOW(19)=0
 157 CONTINUE
 160 VFLOW(20)=(PC(20)-PC(24))/R(20)
     VFLOW(21)=(PC(21)-PC(1))/R(21)
     VFLOW(22)=(PC(22)-PC(1))/R(22)
     VFLOW(23)=(PC(23)-PC(24))/R(23)
     VFLOW(24)=(PC(24)-PC(2))/R(24)
     DV(1)=VFLOW(22)+VFLOW(21)-VFLOW(1)
     DV(2)=VFLOW(24)+VFLOW(1)-VFLOW(2)
     DV(4)=VFLOW(3)-VFLOW(4)
 223 CONTINUE
     DO 15 N=5,10

ready

*
```

```
   15 DV(N)=VFLIN(N)-VFLOW(N)
      DV(11)=VFLIN(11)+VFLOW(5)+VFLOW(6)+VFLOW(7)+VFLOW(8)+VFLOW(9)+
     1VFLOW(10)-VFLOW(11)
      DV(12)=VFLOW(11)-VFLOW(12)
      DV(14)=VFLOW(13)-VFLOW(14)
  233 CONTINUE
      DV(15)=VFLIN(15)-VFLOW(15)
      DV(16)=VFLIN(16)-VFLOW(16)
      DV(17)=VFLIN(17)-VFLOW(17)
      DV(18)=VFLIN(18)-VFLOW(18)
      DV(19)=VFLOW(15)-VFLOW(19)
      DV(20)=VFLOW(16)-VFLOW(20)
      DV(21)=VFLOW(17)-VFLOW(21)
      DV(22)=VFLOW(18)-VFLOW(22)
      DV(23)=VFLOW(19)-VFLOW(23)
      DV(24)=VFLOW(23)+VFLOW(20)-VFLOW(24)
      RETURN
      END
      SUBROUTINE RESIST(LOPR,RK,RINK)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /BCMBLK/ R,FKDA,PP,VO,A2DM2,A2DM3,A2DM4,A2DM12,A2DM13,
     1A2DM14,H,RHO,G,RIN,P,PC
      COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GPO2,CTO2,GPC2,CTC2
      COMMON /CMN11/GPCO,CTCO,GPX,CTX
      COMMON /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGPO2,DCTO2,
     1DGPC2,DCTC2
      COMMON /CMN2T/DGPCO,DCTCO,DGPX,DCTX
      COMMON /VFLT/VFLOW,VFLIN
      DIMENSION R(24),FKDA(24),PP(24),V(24),VO(24),H(24),DV(24),
     1VFLOW(24),VFLIN(18),RIN(18),PC(24),P(24)
      DIMENSION RK(24),RINK(18)
      DIMENSION GPO2(24),GPC2(24),GPCO(48),GPX(1,24),CTO2(4),CTC2(4),
     1CTCO(4),CTX(1,4),DIFO2(10),DIFC2(10),DIFCO(10),DIFX(10),CPH(24),
     2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24 ),DGPO2(24),
     3DGPC2(24),DGPCO(48),DGPX(1,24),AX(1)
C  LOPR=NEG MEANS R=K/V**2+K/V
C   LOPR=0  MEANS R=CONST=RK
C    LOPR=POS MEANS R=RK/V**2 WHERE INPUT R IS USED FOR RK
      IF (LOPR) 1,2,3
    1 WRITE (6,2000)
 2000 FORMAT (' LOPR IS NEG, NOT POSSIBLE YET')
C  THIS WILL EVENTUALLY INCLUDE RESISTANCES OF THE TYPE R=K/V**2+K/V
      RETURN
    2 CONTINUE
C  THIS WILL LEAVE R UNCHANGED FROM THE GIVEN VALUES (I.E. R=CONST)
      RETURN
    3 CONTINUE
      R(1)=RK(1)/(V(1)*V(1))
      DO 41 N=4,11
   41 R(N)=RK(N)/(V(N)*V(N))
```

ready

*

```
      DO 42 N=14,24
   42 R(N)=RK(N)/(V(N)*V(N))
      DO 32  N=5,11
   32 R1N(N)=R1NK(N)/(V(4)*V(4))
      DO 33 N=15,18
   33 R1N(N)=R1NK(N)/(V(14)*V(14))
C  THIS CALCULATES NEW VALUES OF R AND R1N USING R=RK/V**2
      RETURN
      ENTRY CRES(LOPR,RK,RINK)
      IF (LOPR.LE.0) GO TO 20
      RK(1)=R(1)*V(1)*V(1)
      DO 18 N=4,11
   18 RK(N)=R(N)*V(N)*V(N)
      DO 19 N=14,24
   19 RK(N)=R(N)*V(N)*V(N)
      DO 16 N=5,11
   16 R1NK(N)=R1N(N)*V(4)*V(4)
      DO 17 N=15,18
   17 R1NK(N)=R1N(N)*V(14)*V(14)
   20 CONTINUE
      RETURN
      END
      SUBROUTINE PHEART(PSDUR,PSAMP,PSRT,TB,FF,PSTO,Z,PSEND,PSTON)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /AVE/ PPAVE
      IF (TB.LT.PSTO) GO TO 1
      PSTON=PSTO
      PSTO=PSTO+60./PSRT
      Z=60./PSRT-PSDUR
      PSEND=PSTO-Z
      GO TO 2
    1 IF(TB.LT.PSEND) GO TO 2
      FF=PPAVE
      RETURN
    2 CONTINUE
      SIGMA=(PSDUR/4.)**2
      A=-(TB-PSTON-PSDUR/2.)**4/(2.*(SIGMA)**2)
      FF=PSAMP*EXP(A)+PPAVE
  102 CONTINUE
      RETURN
      ENTRY CHEART(PSDUR,PSAMP,PSRT,Z,PSEND,PSTON,PSTO)
      PSTON=PSTO-60./PSRT
      Z=60./PSRT-PSDUR
      PSEND=PSTO-Z
      RETURN
      END
      SUBROUTINE PCHART(I,LVALV2,LVALV3,MTEST,PST,TBS,TB)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /BCMBLK/ R,FKDA,PP,VO,A2DM2,A2DM3,A2DM4,A2DM12,A2DM13,
     1A2DM14,H,RHO,G,RIN,P,PC

ready

*
```

```
      COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GPO2,CTO2,GPC2,CTC2
      COMMON /CMN1T/GPCO,CTCO,GPX,CTX
      COMMON /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGPO2,DCTO2,
     1DGPC2,DCTC2
      COMMON /CMN2T/DGPCO,DCTCO,DGPX,DCTX
      COMMON /KNT/KN1B
      COMMON /SK/ TAU,PO,P1
      COMMON /VFLT/VFLOW,VFLIN
      DIMENSION R(24),FKDA(24),PP(24),V(24),VO(24),H(24),DV(24),
     1VFLOW(24),VFLIN(18),RIN(18),PC(24),P(24),VP(14)
C WARNING...   THIS SUBROUTINE HAS A DIFFERENT DIMENSION..VP(14) ADDED
      DIMENSION TAU(24),PO(24),P1(24)
      DIMENSION GPO2(24),GPC2(24),GPCO(48),GPX(1,24),CTO2(4),CTC2(4),
     1CTCO(4),CTX(1,4),DIFO2(10),DIFC2(10),DIFCO(10),DIFX(10),CPH(24),
     2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24 ),DGPO2(24),
     3DGPC2(24),DGPCO(48),DGPX(1,24),AX(1)
      DIMENSION DCTO2(4),DCTC2(4),DCTCO(4),DCTX(1,4),METO2(4),METC2(4)
      VP(3)=VP3
      VP(13)=VP13
      IF (I-2) 1,2,3
    1 WRITE  (6,111)
  111 FORMAT('SUBROUTINE PCHART HAS BEEN CALLED IN ERROR')
      RETURN
    2 SINDR=1/R(1)+1/R(24)
      SINPDR=PC(1)/R(1)+PC(24)/R(24)
      SOUTDR=        1/RIN(5)+1/RIN(6)+1/RIN(7)+1/RIN(8)+1/RIN(9)+1/
     1RIN(10)+1/RIN(11)
      SOUTPR=PC(5)/RIN(5)+PC(6)/RIN(6)+PC(7)/RIN(7)+PC(8)/RIN(8)+
     1PC(9)/RIN(9)+PC(10)/RIN(10)+PC(11)/RIN(11)
      GO TO 10
    3 SINDR=1/R(11)
      SINPDR=PC(11)/R(11)
      SOUTDR=         1/RIN(15)+1/RIN(16)+1/RIN(17)+1/RIN(18)
      SOUTPR=PC(15)/RIN(15)+PC(16)/RIN(16)+PC(17)/RIN(17)+PC(18)/RIN(18)
   10 J=I+1
      K=I+2
      IF (TB.LT.0.00.OR.TB.GT.0.05) GO TO 201
      WRITE (6,1000) TB,MTEST,PC(I),PC(J),PC(K)
     1,PP(I),PP(J),PP(K),P1(I),P1(I),P1(K),RIN(I),RIN(J),RIN(K),
     2H(I),H(J),H(K),SINPDR,SINDR,VP(J),SOUTDR,SOUTPR,RHO,G
 1000 FORMAT (' TB,MTEST,PC(2,3,4)= ',F10.4,I2,3F10.4
     1,/,1X,12F10.4,/,1X,12F10.4)
  201 CONTINUE
      IF (TB.LE.0.0) GO TO 33
      IF (MTEST.EQ.1) GO TO 9
      IF (LVALV3.EQ.1) GO TO 310
      VALTSJ=-VP(J)
      GO TO 320
  310 VALTSJ=PC(J)-PC(K)
  320 CONTINUE
```

ready

*

```
      IF (PC(I)-PC(J).LT.0.0.AND.VALTSJ.LT.0.0) GO TO 46
      IF (PC(I)-PC(J).LT.0.0) GO TO 21
      IF (VALTSJ.LT.0.0) GO TO 33
      GO TO 11
    9 IF (LVALV2.NE.0.AND.LVALV3.NE.0) GO TO 46
      IF (LVALV2.NE.0) GO TO 21
      IF (LVALV3.NE.0) GO TO 33
   11 CONTINUE
      WRITE (6,1035) TB,I,J,K
 1035 FORMAT (' BOTH VALVES ARE OPEN...NOT WORKING NOW...,
     1TB,I,J,K='F20.5,3I4)
      STOP
   21 P(I)=(PP(I)+P1(I)-RIN(I)*SINPDR+RIN(I)*SINDR*
     1RHO*G*H(I))/(1.0+RIN(I)*SINDR)
      P(J)=PP(J)+P1(J)-RIN(J)*(-VP(J))
      P(K)=(PP(K)+P1(K)+RIN(K)*(-VP(J))
     1-RIN(K)*RHO*G*H(K)*SOUTDR+RIN(K)*SOUTPR)/(1.0+RIN(K)*
     2SOUTDR)
      PC(I)=P(I)+RHO*G*H(I)
      PC(J)=P(J)+RHO*G*H(J)
      PC(K)=P(K)+RHO*G*H(K)
      LVALV2=1
      LVALV3=0
      GO TO 47
   33 P(K)=(PP(K)+P1(K)-RIN(K)*SOUTPR+RIN(K)*
     1SOUTDR*RHO*G*H(K))/(1.0+RIN(K)*SOUTDR)
      PC(K)=P(K)+RHO*G*H(K)
      P(I)=(PP(I)+P1(I)+RIN(I)*SINPDR-RIN(I)*
     1RHO*G*H(I)*SINDR+(RIN(I)/(R(I)+RIN(J)))*(PP(J)+
     2P1(J)+RHO*G*H(J)-RHO*G*H(I)))/(1.0+
     3RIN(I)*SINDR+RIN(I)/(R(I)+RIN(J)))
      PC(I)=P(I)+RHO*G*H(I)
      P(J)=R(I)/(RIN(J)+R(I))*(P1(J)+RIN(J)/R(I)*PC(I)+
     1PP(J)-RIN(J)/R(I)*RHO*G*H(J))
      PC(J)=P(J)+RHO*G*H(J)
      LVALV2=0
      LVALV3=1
      GO TO 47
   46 CONTINUE
      P(I)=(PP(I)+P1(I)+RIN(I)*SINPDR-RIN(I)*SINDR*
     1RHO*G*H(I))/(1.0+RIN(I)*SINDR)
      P(K)=(PP(K)+P1(K)-RIN(K)*SOUTPR+RIN(K)*
     1SOUTDR*RHO*G*H(K))/(1.0+RIN(K)*SOUTDR)
      IF(TB.LE.0.) GO TO 147
      IF(LVALV2.EQ.1.AND.LVALV3.EQ.1) GO TO 146
      TBS=TB
      IF(LVALV2.EQ.1) GO TO 145
      PST=1.2*P(I)-PP(J)-P1(J)
      GO TO 146
  145 PST=0.0

ready

*
```

```
146 IF (TB-TBS.GT.1.0) GO TO 147
    P(J)=PST*EXP(-40.*(TB-TBS))+PP(J)+P1(J)
    GO TO 148
147 P(J)=PP(J)+P1(J)
148 CONTINUE
    PC(I)=P(I)+RHO*G*H(I)
    PC(J)=P(J)+RHO*G*H(J)
    PC(K)=P(K)+RHO*G*H(K)
    LVALV2=1
    LVALV3=1
 47 CONTINUE
    RETURN
    END
     SUBROUTINE SKON(LOPP1,TB)
    IMPLICIT REAL*8 (A-H,O-Z)
    COMMON /BCMBLK/ R,FKDA,PP,VO,A2DM2,A2DM3,A2DM4,A2DM12,A2DM13,
   1A2DM14,H,RHO,G,RIN,P,PC
    COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GPO2,CTO2,GPC2,CTC2
    COMMON /CMN1T/GPCO,CTCO,GPX,CTX
    COMMON /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGPO2,DCTO2,
   1DGPC2,DCTC2
    COMMON /CMN2T/DGPCO,DCTCO,DGPX,DCTX
    COMMON /SK/ TAU,PO,P1
    COMMON /VFLT/VFLOW,VFLIN
    DIMENSION R(24),FKDA(24),PP(24),V(24),VO(24),H(24),DV(24),
   1VFLOW(24),VFLIN(18),RIN(18),PC(24),P(24)
    DIMENSION TAU(24),PO(24),P1(24)
    DIMENSION GPO2(24),GPC2(24),GPCO(48),GPX(1,24),CTO2(4),CTC2(4),
   1CTCO(4),CTX(1,4),DIFO2(10),DIFC2(10),DIFCO(10),DIFX(10),CPH(24),
   2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24 ),DGPO2(24),
   3DGPC2(24),DGPCO(48),DGPX(1,24),AX(1)
    DIMENSION DCTO2(4),DCTC2(4),DCTCO(4),DCTX(1,4),METO2(4),METC2(4)
    IF (LOPP1.EQ.0) GO TO 100
    DO 9 N=1,24
    IF (TAU(N).LT.1.D-02) GO TO 9
    VMAX=VO(N)+.224*TAU(N)
    IF (V(N).GT.VMAX) GO TO 8
    GO TO 9
  8 WRITE (6,1001) TB,N,V(N),VMAX
1001 FORMAT (' TB,N,V(N),VMAX=',F10.4,I6,2F10.4,'....V TOO BIG.STOP.')
    STOP
  9 CONTINUE
    P1(1)=PO(1)-ALOG(1.0-(V(1)-VO(1))/(.224*TAU(1)))/FKDA(1)
    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)
    P1(11)=PO(11)-ALOG(1.0-(V(11)-VO(11))/(.224*TAU(11)))/FKDA(11)
    P1(12)=PO(12)-ALOG(1.0-(V(12)-VO(12))/(.224*TAU(12)))/FKDA(12)
    P1(14)=PO(14)-ALOG(1.0-(V(14)-VO(14))/(.224*TAU(14)))/FKDA(14)
    DO 10 N=19,24
    P1(N)=PO(N)-ALOG(1.0-(V(N)-VO(N))/(.224*TAU(N)))/FKDA(N)
```

ready

*

```
   10 CONTINUE
      P1(3)=FKDA(3)*(V(3)-VO(3))
      P1(13)=FKDA(13)*(V(13)-VO(13))
      DO 15 N=5,10
      P1(N)=FKDA(N)*(V(N)-VO(N))
   15 CONTINUE
      DO 20 N=15,18
      P1(N)=FKDA(N)*(V(N)-VO(N))
   20 CONTINUE
      RETURN
  100 CONTINUE
      DO 110 N=1,24
      P1(N)=FKDA(N)*(V(N)-VO(N))
  110 CONTINUE
      RETURN
      END
      SUBROUTINE BCDRIV(LOPGS)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8  LAMBO2,LAMBC2,LAMBCO,LAMBX,LCO2,LCC2,LCCO,LCX,METO2,METC2
C   WHEN ADDINT THIS ROUTINE TO BMAIN,REMOVE THE SECOND DIM. STATEMENT
C          AND REPLACE IT WITH THE ONE IN BMAIN CONTAINING THESE QUANTATIES
      DIMENSION GPO2(24),GPC2(24),GPCO(48),GPX(1,24),CTO2(4),CTC2(4),
     1CTCO(4),CTX(1,4),DIFO2(10),DIFC2(10),DIFCO(10),DIFX(10),CPH(24),
     2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24 ),DGPO2(24),
     3DGPC2(24),DGPCO(48),DGPX(1,24),AX(1)
      DIMENSION VFLIN(18),VFLOW(24),DV(24),V(24)
      DIMENSION DCTO2(4),DCTC2(4),DCTCO(4),DCTX(1,4),METO2(4),METC2(4)
      DIMENSION CTLO2(18), CTLC2(18),CTLCO(18),CTLX(1,18)
      COMMON /BCONLY/LCO2,LCC2,LCCO,LCX
      COMMON /BC/LAMBO2,LAMBC2,LAMBCO,LAMBX,AO2,AC2,ACO,AX,HEMAT,DIFO2,
     1DIFC2,DIFCO,DIFX,CPH,CSATO2,CSATC2,METO2,METC2
      COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GPO2,CTO2,GPC2,CTC2
      COMMON /CMN1T/GPCO,CTCO,GPX,CTX
      COMMON /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGPO2,DCTO2,
     1DGPC2,DCTC2
      COMMON /CMN2T/DGPCO,DCTCO,DGPX,DCTX
      COMMON /BCOVER/CTLO2,CTLC2,CTLCO,CTLX
      COMMON /VFL1/VFLOW,VFLIN
      DO 101 N=1,6
      CTLO2(N+4)=LCO2(N)
      CTLC2(N+4)=LCC2(N)
      CTLCO(N+4)=LCCO(N)
      CTLX(1,N+4)=LCX(1,N)
  101 CONTINUE
      DO 102 N=1,4
      CTLO2(N+14)=CTO2(N)
      CTLC2(N+14)=CTC2(N)
      CTLCO(N+14)=CTCO(N)
      CTLX(1,N+14)=CTX(1,N)
  102 CONTINUE

ready

*
```

```
      GO TO (1,2,3,4,5),LOPGS
C    LOPGS = 1......O2,C2
C    LOPGS = 2......O2,C2,X
C    LOPGS = 3......X
C    LOPGS = 4......O2,CI,CO
C    LOPGS = 5......O2,C2,CO,X
    1 CALL O2DRIV(LOPGS)
      DO 10 N=1,4
      IF (CTO2(N).LE.0.0) GO TO 201
      DCTO2(N)=-DIFO2(N+6)*(CTO2(N)-LAMBO2*GPO2(N+14))-METO2(N)
      GO TO 202
  201 DCTO2(N)=-DIFO2(N+6)*(CTO2(N)-LAMBO2*GPO2(N+14))
  202 CONTINUE
      IF (CTC2(N).LE.0.0) GO TO 203
      DCTC2(N)=-DIFC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))+METC2(N)
      GO TO 204
  203 DCTC2(N)=-DIFC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))
  204 CONTINUE
   10 CONTINUE
      GO TO 600
    2 CALL O2DRIV(LOPGS)
      CALL XDRIV(LOPGS)
      DO 11 N=1,4
      IF (CTO2(N).LE.0.0) GO TO 301
      DCTO2(N)=-DIFO2(N+6)*(CTO2(N)-LAMBO2*GPO2(N+14))-METO2(N)
      GO TO 302
  301 DCTO2(N)=-DIFO2(N+6)*(CTO2(N)-LAMBO2*GPO2(N+14))
  302 CONTINUE
      IF (CTC2(N).LE.0.0) GO TO 303
      DCTC2(N)=-DIFC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))+METC2(N)
      GO TO 304
  303 DCTC2(N)=-DIFC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))
  304 CONTINUE
      DCTX(1,N)=-DIFX(   N+6)*(CTX(1,N)-LAMBX*GPX(1,N+14))
   11 CONTINUE
      GO TO 600
    3 CALL XDRIV(LOPGS)
      DO 12 N=1,4
      DCTX(1,N)=-DIFX(   N+6)*(CTX(1,N)-LAMBX*GPX(1,N+14))
   12 CONTINUE
      GO TO 600
    4 CALL O2DRIV(LOPGS)
      DO 13 N=1,4
      IF (CTO2(N).LE.0.0) GO TO 401
      DCTO2(N)=-DIFO2(N+6)*(CTO2(N)-LAMBO2*GPO2(N+14))-METO2(N)
      GO TO 402
  401 DCTO2(N)=-DIFO2(N+6)*(CTO2(N)-LAMBO2*GPO2(N+14))
  402 CONTINUE
      IF (CTC2(N).LE.0.0) GO TO 403
      DCTC2(N)=-DIFC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))+METC2(N)

ready

*
```

```
      GO TO 404
  403 DCTC2(N)=-DIFC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))
  404 CONTINUE
      DCTCO(N)=-DIFCO(N+6)*(CTCO(N)-LAMBCO*GPCO(N+14))
   13 CONTINUE
      GO TO 600
    5 CONTINUE
      CALL O2DRIV(LOPGS)
      CALL XDRIV(LOPGS)
      DO 14 N=1,4
      IF (CTO2(N).LE.0.0) GO TO 701
      DCTO2(N)=-DIFO2(N+6)*(CTO2(N)-LAMBO2*GPO2(N+14))-METO2(N)
      GO TO 702
  701 DCTO2(N)=-DIFO2(N+6)*(CTO2(N)-LAMBO2*GPO2(N+14))
  702 CONTINUE
      IF (CTC2(N).LE.0.0) GO TO 703
      DCTC2(N)=-DIFC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))+METC2(N)
      GO TO 704
  703 DCTC2(N)=-DIFC2(N+6)*(CTC2(N)-LAMBC2*GPC2(N+14))
  704 CONTINUE
      DCTCO(N)=-DIFCO(N+6)*(CTCO(N)-LAMBCO*GPCO(N+14))
      DCTX(1,N)=-DIFX(  N+6)*(CTX(1,N)-LAMBX*GPX(1,N+14))
   14 CONTINUE
  600 CONTINUE
      DO 501 N=1,6
      LCO2(N)=CTLO2(N+4)
      LCC2(N)=CTLC2(N+4)
      LCCO(N)=CTLCO(N+4)
      LCX(1,N)=CTLX(1,N+4)
  501 CONTINUE
      DO 502 N=1,4
      CTO2(N)=CTLO2(N+14)
      CTC2(N)=CTLC2(N+14)
      CTCO(N)=CTLCO(N+14)
      CTX(1,N)=CTLX(1,N+14)
  502 CONTINUE
      RETURN
      END
      SUBROUTINE O2DRIV(LOPGS)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8  LAMBO2,LAMBC2,LAMBCO,LAMBX,LCO2,LCC2,LCCO,LCX,METO2,METC2
      DIMENSION CTLO2(18), CTLC2(18),CTLCO(18),CTLX(1,18)
      DIMENSION GPO2(24),GPC2(24),GPCO(48),GPX(1,24),CTO2(4),CTC2(4),
     1CTCO(4),CTX(1,4),DIFO2(10),DIFC2(10),DIFCO(10),DIFX(10),CPH(24),
     2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24 ),DGPO2(24),
     3DGPC2(24),DGPCO(48),DGPX(1,24),AX(1)
      DIMENSION VFLIN(18),VFLOW(24),DV(24),V(24)
      DIMENSION SS(2),S(2,2),EKO2(24),EKC2(24),EKCO(24),CO2(24),
     1CC2(24),CCO(24),HMSP(24)
      DIMENSION DCTO2(4),DCTC2(4),DCTCO(4),DCTX(1,4),METO2(4),METC2(4)
```

ready

*

```
      DIMENSION SCRIPA(24),SCRIPB(24),SCRIPC(24),SCRIPD(24),DENOM(24)
      COMMON /VFLT/VFLOW,VFLIN
      COMMON /BCOVER/CTLO2,CTLC2,CTLCO,CTLX
      COMMON /BC/LAMBO2,LAMBC2,LAMBCO,LAMBX,AO2,AC2,ACO,AX,HEMAT,DIFO2,
     1DIFC2,DIFCO,DIFX,CPH,CSATO2,CSATC2,METO2,METC2
      COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GPO2,CTO2,GPC2,CTC2
      COMMON /CMN1T/GPCO,CTCO,GPX,CTX
      COMMON /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGPO2,DCTO2,
     1DGPC2,DCTC2
      COMMON /CMN2T/DGPCO,DCTCO,DGPX,DCTX
      ALPHA=1.339E-03
      BETA=8.98
      GAMMA=1.00
      DO 12 N=1,24
      PO2=GPO2(N)
      PC2=GPC2(N)
      CALL SATUR(PO2,PC2,HEMAT,SS,S,S1,PH1)
      CPH(N)=PH1
      CSATO2(N)=SS(1)
      CSATC2(N)=SS(2)
      IF (LOPGS.LT.4) GO TO 11
C   AC2=.030/ BY THE PROGRAN SATUR
      MMMMMM=1.0
      DGPCO(N+24)=ACO*MMMMMM*S1* GPCO(N)/AO2
      HMSP(N)=HEMAT*MMMMMM*S1*GPCO(N)
      CCO(N)=(1.-HEMAT)*ACO*GPCO(N)+HEMAT*GPCO(N+24)
   11 CONTINUE
      SCRIPA(N)=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
      CC2(N)=SS(2)*GAMMA*PC2
   12 CONTINUE
      CALL GFLTST(CTLO2,DV,LAMBO2,DIFO2,CO2,GPO2,EKO2)
      CALL GFLTST(CTLC2,DV,LAMBC2,DIFC2,CC2,GPC2,EKC2)
      CALL GFLTST(CTLCO,DV,LAMBCO,DIFCO,CCO,GPCO,EKCO)
      DO 50 N=1,24
      DENOM(N)=(SCRIPA(N)*SCRIPD(N)-SCRIPC(N)*SCRIPB(N))*V(N)
      DGPO2(N)=(SCRIPD(N)*EKO2(N)-SCRIPB(N)*EKC2(N))/DENOM(N)
      DGPC2(N)=(SCRIPA(N)*EKC2(N)-SCRIPC(N)*EKO2(N))/DENOM(N)
      IF (LOPGS.LT.4) GO TO 50
      DGPCO(N)=(EKCO(N)/(ACO*V(N))-HMSP(N))/(1.+HEMAT)
   50 CONTINUE
      RETURN
      END
      SUBROUTINE XDRIV(LOPGS)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8  LAMBO2,LAMBC2,LAMBCO,LAMBX,LCO2,LCC2,LCCO,LCX,METO2,METC2
      DIMENSION GPO2(24),GPC2(24),GPCO(48),GPX(1,24),CTO2(4),CTC2(4),

ready

*
```

```
     1CTCO(4),CTX(1,4),DIFO2(10),DIFC2(10),DIFCO(10),DIFX(10),CPH(24),
     2LCO2(6),LCC2(6),LCCO(6),LCX(1,6),CSATO2(24),CSATC2(24 ),DGPO2(24),
     3DGPC2(24),DGPCO(48),DGPX(1,24),AX(1)
      DIMENSION CTLO2(18), CTLC2(18),CTLCO(18),CTLX(1,18)
      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)
      DIMENSION EKX1(1,24)
      COMMON /VFLT/VFLOW,VFLIN
      COMMON /BCOVER/CTLO2,CTLC2,CTLCO,CTLX
      COMMON /BC/LAMBO2,LAMBC2,LAMBCO,LAMBX,AO2,AC2,ACO,AX,HEMAT,DIFO2,
     1DIFC2,DIFCO,DIFX,CPH,CSATO2,CSATC2,METO2,METC2
      COMMON /CMN1/V,VP2,VP3,VP4,VP12,VP13,VP14,GPO2,CTO2,GPC2,CTC2
      COMMON /CMN1T/GPCO,CTCO,GPX,CTX
      COMMON /CMN2/DV,DVP2,DVP3,DVP4,DVP12,DVP13,DVP14,DGPO2,DCTO2,
     1DGPC2,DCTC2
      COMMON /CMN2T/DGPCO,DCTCO,DGPX,DCTX
      DO 1 N=1,24
      C(N)=AX(1)*GPX(1,N)
      CALL GFLTST(CTLX,DV,LAMBX,DIFX,C,GPX,EKX1)
      DGPX(1,N)=EKX1(1,N)/(AX(1)*V(N))
    1 CONTINUE
      RETURN
      END
      SUBROUTINE SATUR (PO2,PCO2,HEMAT,SS,S,S1,PH)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION SS(2),S(2,2)
      Y=0.0
      TPCO2=1.00
      DO 2 N=1,2
      IF (PCO2.GE.0.001) GO TO 1
      TPCO2=PCO2
      PCO2=0.001
    1 PH=7.59+Y-0.2741*ALOG(PCO2/20.0)
      Z=EXP((PH-7.4)*1.812)
      C=5.727E-3*Z+4.273E-3
      V=C*PO2
      IF (V.GT.1.0E-11) GO TO 11
      V=0.0
   11 CONTINUE
      U=0.925*V+2.8*V*V+30.0*V*V*V
      S1=U/(1+U)
    2 Y=0.045*(1.0-S1)
      SS(1)=S1
      A=(.925+5.6*V+90*V*V)
      AA=A*PO2*(.010377)*Z
      A2=AA*.045
      S11=A*C/((1.+U)*(1.+U)+A2)
      S(1,1)=S11
      S12=-AA*.2741/((1.+U)*(1.+U)+AA*.045)/PCO2
      S(1,2)=S12
```

ready

*

```
C
      DOX=0.59+0.2913*(7.4-PH)
      DR=0.664+0.2275*(7.4-PH)
      DDD=DOX+(DR-DOX)*(1.0-S1)
      YY=EXP(2.3961*PH-14.7162)
      CP=0.0307 *(1.0+YY)
      SS(2)=(1.0-HEMAT)*CP+HEMAT*DDD*CP
C
      DPHO2=-.045*S11
      DDR=-.2275*DPHO2
      DDOX=-.2913*DPHO2
      DDDD=DDOX+(DDR-DDOX)*(1.-S1)+(DR-DOX)*(-S11)
      DYYO2=2.396*DPHO2*YY
      DCP=.0307*(+DYYO2)
      S(2,1)=(1.-HEMAT)*DCP+HEMAT*DDDD*CP+HEMAT*DDDD*DCP
      DYC2=-.045*S12
      DPHC2=DYC2-.2741/PCO2
      DDRC2=-.2275*DPHC2
      DDOXC2=-.2913*DPHC2
      DDDDC2=DDOXC2+(DDRC2-DDOXC2)*(1.-S1)-(DR-DOX)*S12
      DYYC2=2.3961*YY*DPHC2
      DCPC2=.0307*DYYC2
      S(2,2)=(1-HEMAT)*DCPC2+HEMAT*DDDDC2*CP+HEMAT*DDD*DCPC2
      IF (TPCO2.GE.0.001) GO TO 3
      PCO2=TPCO2
    3 CONTINUE
      RETURN
      END
      SUBROUTINE GFLTST(CT,VP,LAMBDA,D,C,GP,EK)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 LAMBDA
      DIMENSION VFLOW(24),VFLIN(18),C(24),CCFL(24),CCFLIN(18),VP(24),
     1EK(24),D(10),CT(18),GP(24)
      COMMON /VFLT/VFLOW,VFLIN
      COMMON /CFLT/CCFL,CCFLIN
      CALL FLTEST(1,2,1,1,C)
      CCFL(2)=C(2)*VFLOW(2)
      CCFL(3)=C(3)*VFLOW(3)
      CCFL(4)=0.0
      DO 11 N=5,11
      I=N
      CALL FLTEST(I+24,1,4,I+24,C)
   11 CCFL(4)=CCFL(4)+CCFLIN(N)
      DO 12 N=5,10
      I=N
   12 CALL FLTEST(1,11,I,I,C)
      CALL FLTEST(11,12,11,11,C)
      CCFL(12)=C(12)*VFLOW(12)
      CCFL(13)=C(13)*VFLOW(13)
      CCFL(14)=0.0
```

ready

*

```
      DO 13 N=15,18
      I=N
      CALL FLTEST(I+24,I,14,I+24,C)
   13 CCFL(14)=CCFL(14)+CCFLIN(N)
      DO 14 N=15,20
      I=N
   14 CALL FLTEST(I,I+4,I,I,C)
      CALL FLTEST(21,1,21,21,C)
      CALLFLTEST(22,1,22,22,C)
      CALL FLTEST(23,24,23,23,C)
      CALL FLTEST(24,2,24,24,C)
      EK(1)  =CCFL(21)+CCFL(22)-CCFL(1)-C(1)*VP(1)
      EK(2)  =CCFL(1)+CCFL(24)-CCFL(2)-C(2)*VP(2)
      EK(3)  =CCFL(2)-CCFL(3)-C(3)*VP(3)
      EK(4)  =CCFL(3)-CCFL(4)-C(4)*VP(4)
      DO 100 N=5,10
  100 EK(N)=CCFLIN(N)-CCFL(N)-C(N)*VP(N)+D(N-4)*(CT(N)-LAMBDA*GP(N))
      EK(11)=CCFL(5)+CCFL(6)+CCFL(7)+CCFL(8)+CCFL(9)+CCFL(10)+CCFLIN(11)
     1-CCFL(11)-C(11)*VP(11)
      EK(12)=CCFL(11)-CCFL(12)-C(12)*VP(12)
      EK(13)=CCFL(12)-CCFL(13)-C(13)*VP(13)
      EK(14)=CCFL(13)-CCFL(14)-VP(14)*C(14)
      DO 112 N=15,18
  112 EK(N)=CCFLIN(N)-CCFL(N)-VP(N)*C(N)+D(N-8)*(CT(N)-LAMBDA*GP(N))
      DO 113 N=19,23
  113 EK(N)=CCFL(N-4)-CCFL(N)-VP(N)*C(N)
      EK(24)=CCFL(23)+CCFL(20)-CCFL(24)-VP(24)*C(24)
      DO 131 N=1,24
      IF (DABS(EK(N)).GT.1.0E-13) GO TO 130
      EK(N)=0.0
  130 CONTINUE
  131 CONTINUE
      RETURN
      END
      SUBROUTINE FLTEST(I1,IN,IP,IV,C)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FL(42),C(24),CC(42)
      COMMON /VFLT/FL
      COMMON /CFLT/CC
      IF(FL(I1)) 1,1,2
    1 CC(IV)= C(IN)*FL(I1)
      RETURN
    2 CC(IV)= C(IP)*FL(I1)
      RETURN
      END
```

ready

*

```
      SUBROUTINE DERIVS( OPT1, MTEST, PSAVE )
      IMPLICIT  REAL*8( A-H, O-Z )
      REAL*8 N, NW
      REAL*4 G1, G2, G3, G4
      INTEGER  OPT1, FBOPT
      DIMENSION X(6), Z(6), ALPHA(6), BETA(6), PS(6), DDCW(12),
     1    DDNW(12), PSAVE(12)
      COMMON /BLK1/ N(12,3), VTX, VALV(6), PALV(6), SW(12), VSPR
      COMMON /BLK2/ DN(12,3), DVTX, DVALV(6), DPALV(6), DSW(12), DVSPR
       COMMON /BLK10/ PGAV, DPGAV, PGAVB, DPGAVB
       COMMON /BLK11/ PTX, DPTX, PD, DPD, G1, G2, G3, G4, FBOPT
       COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
     1    F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PBS,
     2    DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
     3    PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
     4    S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
     5    SQT(12), DP(12), HDT
       DATA TTTS / -1.0 D0 /
C
       TTT = T
       IF ( OPT1. EQ. 10. AND. TTT. EQ. TTTS )  GO TO 10
C
     5 FSPR = F(2)
       IF ( MTEST. EQ. 1 ) CALL FORCE ( TTT, FSPR, DT )
       IF ( MTEST. NE. 1 . AND . FBOPT. GE. 9 ) CALL FORCE ( TTT, FSPR,
     &    DT )
C
       GO TO 15
    10 CONTINUE
       DPTX = 0.
       DPD = 0.
    15 CONTINUE
       TTTS = TTT
       PELTX =  PELAST ( 0, 7, VTX )
       DPELTX = PELAST (-1, 7, VTX )
       PELDA =  PELAST ( 0, 8, VD )
       DPELDA = PELAST (-1, 8, VD )
       DO 20 I = 1, 6
       IDUMY = 1
       PELALV(I) = PELAST ( 0, IDUMY, VALV(I) )
    20 DPELAL(I) = PELAST (-1, IDUMY, VALV(I) )
C
       IF ( OPT1. NE. 0 )  CALL PGRAV ( PBS, PGAB, PG, DPBS, DPGAB, DPG )
C
       DPS2 = DPTX - DPD + DPGAB + DPGAVB - DPGAV
       DUM = DPTX + DPBS - DPGAV
       DUM2 = PTX + PBS - PGAV
       DO 30  I= 1, 6
       DPS(I) = DUM + DPG(I)
    30 PS(I) = DUM2 + PG(I)

ready

*
```

```
C
      CALL AIRWAY ( DN, DDCW )
C
      DO 50  J= 2, 6
      DCWDP(J) = G1 * DEXP( G2 * ( P(J) - 760. ) )
      DP(J)= DNT(J) * G / ( V(J) * ( 1.- G * DCWDP(J) ) )
      DP(J) = ( P(J) - PSAVE(J) ) / DT
      DCW(J) = DCWDP(J) * DP(J)
      DNW(J)= V(J) * DCW(J)
   50 DSW(J) = DNW(J) - DDCW(J)
      DNT(1) = 0.
      IF ( F(2) ) 60, 70, 80
   60 DO 65  I= 1, NG
      DN(1,I) = - F(2) * ( C(2,I) + HDT * DC(2,I) ) + S(I)
   65 DNT(1) = DNT(1) + DN(1,I)
      DUMY = -F(2) * ( CW(2) + HDT * DCW(2) )
      GO TO 90
   70 DO 75  I= 1, NG
   75 DN(1,I) = 0.
      DUMY = 0.
      GO TO 90
   80 DO 85  I= 1,NG
      DN(1,I) = -C(1,I) * F(2) + S(I)
   85 DNT(1) = DNT(1) + DN(1,I)
      DUMY = -F(2) * CW(2)
   90 DVSPR = DNT(1) / ( P(1) / GS - CW(1) )
      DNW(1) = DVSPR * CW(1)
      DSW(1) = DNW(1) - DUMY
      DO 100  I= 1, NG
  100 DC(1,I) = ( DN(1,I) - DVSPR * C(1,I) ) / V(1)
C
      CALL MOLES( F(7),DNT(7),NG,HDT,C,DC,DN,DDNW(7),CW,DDCW,3,7 )
      CALL MOLES( F(8),DNT(8),NG,HDT,C,DC,DN,DDNW(8),CW,DDCW,4,8 )
      CALL MOLES( F(9),DNT(9),NG,HDT,C,DC,DN,DDNW(9),CW,DDCW,5,9 )
      CALL MOLES( F(10),DNT(10),NG,HDT,C,DC,DN,DDNW(10),CW,DDCW,6,10 )
      CALL MOLES( F(11),DNT(11),NG,HDT,C,DC,DN,DDNW(11),CW,DDCW,5,11 )
      CALL MOLES( F(12),DNT(12),NG,HDT,C,DC,DN,DDNW(12),CW,DDCW,6,12 )
C
      DO 250  J = 7, 12
      CT(J) = 0.
      DO 250  I= 1, NG
  250 CT(J) = CT(J) + C(J,I)
      ALPHAT = 0.
      BETAT = 0.
      DO 300  J = 7, 12
      K = J - 6
      DCWDP(J) = G1 * DEXP( G2 * ( P(J) - 760. ) )
      X(K) = G * DCWDP(J)
      Z(K) = 1. - X(K)
       DMY = P(J) + DPELAL(K) * VALV(K) * Z(K) - G * CW(J)

ready

*
```

```
      ALPHA(K) = ( DNT(J) * G - DPS(K) * V(J) * Z(K) ) / DMY
      BETA(K) = DPELTX * VALV(K) * Z(K) / DMY
      ALPHAT = ALPHAT + ALPHA(K)
      BETAT = BETAT + BETA(K)
  300 CONTINUE
      DVTX = ( ALPHAT * DPELDA + DPS2 ) / ( DPELTX + DPELDA * ( 1.-
     1         BETAT ) )
      DO 400  J= 7, 12
      K = J - 6
      DVALV(K) = ALPHA(K) + BETA(K) * DVTX
      DP(J) = ( DNT(J)*G - ( P(J) - G*CW(J) )*DVALV(K) ) / ( V(J)*Z(K) )
      DPALV(K) = DP(J)
      DSW(J) = DCWDP(J) * DP(J) * V(J) + CW(J) * DVALV(K) - DDNW(J)
  400 CONTINUE
 1001 FORMAT( 3X,F5.3, 3X, F12.8, 3X, F12.8,
     1     3X, F12.7, 2( 3X, E10.4 ) )
 1000 RETURN
      END
      SUBROUTINE PARMS2( VA )
      REAL*8 N, NW
      REAL*4 PCALC, CTCALC, VLDIF, G1, G2, G3, G4
      IMPLICIT  REAL*8( A-H, O-Z )
      INTEGER FBOPT
      DIMENSION DELP(6), X(5), B(5), KNT2(5), FACT(5)
      COMMON /BLK1/ N(12,3), VTX, VALV(6), PALV(6), SW(12), VSPR
      COMMON /BLK2/ DN(12,3), DVTX, DVALV(6), DPALV(6), DSW(12), DVSPR
      COMMON /BLK6/ A(5,5), FO(6), FI(6), SQO(6), SQI(6)
      COMMON /BLK7/ PCALC(12), CTCALC(12), VLDIF
      COMMON /BLK10/ PGAV, DPGAV, PGAVB, DPGAVB
      COMMON /BLK11/ PTX, DPTX, PD, DPD, G1, G2, G3, G4, FBOPT
      COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
     1    F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PBS,
     2    DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
     3    PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
     4    S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
     5    SQT(12), DP(12), HDT
      DATA EPS / 1.0D-5 /
      VLUNG = VA
      V(1) = VSPR
      DO 25  J=7,12
      K= J-6
      P(J) = PALV(K)
      V(J) = VALV(K)
   25 VLUNG = VLUNG+V(J)
      TN(1) = 0.
      DO 75 I= 1,NG
      C(1,I) = N(1,I) / V(1)
   75 TN(1) = TN(1) + N(1,I)
      CT(1) = P(1) / GS - CW(1)
      DO 90 J=7,12

ready

*
```

```
      CW(J) = G3 * DEXP( G2 *(P(J)-760.) )
      DCWDP(J) = G1 * DEXP( G2 * ( P(J) - 760. ) )
      CT(J) = P(J) / G - CW(J)
      TN(J) = 0.
      DO 85  I= 1,NG
      C(J,I) = N(J,I)/V(J)
   85 TN(J) = TN(J) + N(J,I)
      CT(J) = P(J) / G - CW(J)
   90 CONTINUE
      DO 100 J= 2,6
      CW(J) = G3 * DEXP( G2 *(P(J)-760.) )
      DCWDP(J) = G1 * DEXP( G2 * ( P(J) - 760. ) )
      CT(J) = P(J) / G - CW(J)
      TN(J) = 0.
      DO 100  I= 1,NG
      TN(J) = TN(J) + N(J,I)
  100 C(J,I) = N(J,I) / V(J)
      PPL = PTX + PBS - PELTX - PGAV
      VD = VLUNG - VTX - VLDIF
      DO 130  J= 1,5
      FACT(J) = .6
      KNT2(J) = 0
      DO 130  I= 1,5
  130 A(I,J) = 0.
      DO 210 KNT = 1, 50
      IF ( KNT. EQ. 1 ) GO TO 155
      CALL FLOW ( 1, 2 )
      CALL FLOW ( 2, 3 )
      CALL FLOW ( 2, 4 )
      CALL FLOW ( 3, 5 )
      CALL FLOW ( 4, 6 )
  155 CONTINUE
      CALL FLOW ( 3, 7 )
      CALL FLOW ( 4, 8 )
      CALL FLOW ( 5, 9 )
      CALL FLOW ( 6, 10 )
      CALL FLOW ( 5, 11 )
      CALL FLOW ( 6, 12 )
      DO 160  J= 2,6
      FO(J) = 0.
      FI(J) = 0.
      SQO(J) = 0.
      SQI(J) = 0.
  160 CT(J) = P(J) / G - CW(J)
      CALL SUB1 ( 1, 2 )
      CALL SUB1 ( 2, 3 )
      CALL SUB1 ( 2, 4 )
      CALL SUB1 ( 3, 5 )
      CALL SUB1 ( 4, 6 )
      CALL SUB1 ( 3, 7 )
```

ready

*

```
      CALL SUB1 ( 4, 8 )
      CALL SUB1 ( 5, 9 )
      CALL SUB1 ( 6, 10 )
      CALL SUB1 ( 5, 11 )
      CALL SUB1 ( 6, 12 )
      SB = 0.
      DO 200  J= 2,6
      K= J - 1
      KNT2(K) = KNT2(K) + 1
      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
      B(K) = -FI(J) + CT(J) * FO(J)
  200 SB = SB + DABS( B(K) )
      CALL BAPM ( A, X, B )
      SDP = 0.
      DO 203  K= 1,5
  203 SDP = SDP + ABS( X(K) )
      IF ( KNT. GT. 25)  WRITE(6,1234) T,SB,(P(J),J=2,6)
 1234 FORMAT ( 1X, F5.3,1X,1PE9.2, 5(1X,F9.5))
      IF ( SB. LT. EPS )  GO TO 215
      DO 210  J= 2,6
      K = J-1
      IF ( KNT2(K). EQ. 1 )  GO TO 206
      IF ( DELP(J). EQ. 0.. OR. X(K). EQ. 0. )  GO TO 206
      IF ( DELP(J) / X(K) )  205, 205, 204
  204 IF ( KNT2(K). LT. 3 ) GO TO 206
      FACT(K) = 1.0
      GO TO 206
  205 FACT(K) = 0.6
      KNT2(K) = 0.
  206 X(K) = FACT(K) * X(K)
      DELP(J) = X(K)
      P(J) = P(J) + DELP(J)
      CW(J) = G3 * DEXP( G2 *(P(J)-760.) )
  210 DCWDP(J) = G1 * DEXP( G2 * ( P(J) - 760. ) )
      STOP 1234
  215 CONTINUE
C       WRITE(6,1111) KNT, SB
 1111 FORMAT(1X,'KNT, SB = ', I5, E14.3 )
      DO 220  J= 1,12
      NW(J) = CW(J) * V(J)
  220 TOTN(J) = TN(J) + NW(J)
      DO 230  J= 2,6
      CW(J) = G3 * DEXP( G2 *(P(J)-760.) )
      CTCALC(J) = CT(J)
  230 PCALC(J) = TOTN(J) * G / V(J)
      DO 235  J= 1,12
  235 CT(J) = TOTN(J) / V(J)
      RETURN
      END

ready

*
```

```
      SUBROUTINE PGRAV ( PBS, PGAB, PG, DPBS, DPGAB, DPG )
      IMPLICIT  REAL*8( A-H, O-Z )
      DIMENSION PG(6), DPG(6)
      COMMON /BLK8/ PHI, GF, GR, GT, GD, HP, PIE, ANG1, ANG2, PHI1, PHI2
      COMMON /BLK10/ PGAV, DPGAV, PGAVB, DPGAVB
      GR = 1. + GF
      PHI = PHI2
      COSPHI = DCOS( PHI )
      SINPHI = DSIN( PHI )
       GD = GR * COSPHI
       GT = GR * DCOS( PHI - .2618 )
       GCON = GR * 0.73557 / 2.
       H = 8.67 * COSPHI
       PBS = 760.
       PGAB = GCON * 8.* DCOS( 2.*PHI ) + 2. * GR * SINPHI
       PG(1) = GCON * (13.0 - H )
       PG(2) = GCON * (13.0 - H )
       PG(3) = GCON * 13.0
       PG(4) = GCON * 13.0
       PG(5) = GCON * ( 13.0 + H )
       PG(6) = GCON * ( 13.0 + H )
       DPBS = 0.
       DPGAB = 0.
       DPG(1) = 0.
       DPG(2) = 0.
       DPG(3) = 0.
       DPG(4) = 0.
       DPG(5) = 0.
       DPG(6) = 0.
       PGAVB = GCON * 13.* ( 1. + COSPHI )
       DPGAVB = 0.
       SPG = 0.
       SDPG = 0.
       DO 10  J=1,6
       SPG = SPG + PG(J)
    10 SDPG = SDPG + DPG(J)
       PGAV = SPG / 6.
       DPGAV = SDPG / 6.
       WRITE(6,98)
    98 FORMAT( //, '  PGRAV SUBROUTINE ',//,
     1    '  PGAV, DPGAV, PGAVB, DPGAVB, PGAB, DPGAB',/,
     2    '  PG, / , DPG', / )
       WRITE(6,99) PGAV, DPGAV, PGAVB, DPGAVB, PGAB, DPGAB, PG, DPG
    99 FORMAT(1X,6E12.5)
       WRITE(6,96)
    96 FORMAT( /, 1X, ' GR, PHI, COSPHI, GD, GT, H', / )
       WRITE(6,99) GR, PHI, COSPHI, GD, GT, H
   101 CONTINUE
       RETURN
       END
```

ready

*

```
      FUNCTION PELAST ( KTEST, I, V )
      REAL*8 V, HP, PELAST
      COMMON /BLK5/ A(7,8)
      DATA HP / 1.570796327 D0 /
      IF ( V - A(2,I) ) 1,1,2
    1 VDIF = A(2,I) - A(4,I)
      GO TO 3
    2 VDIF = A(3,I) - A(2,I)
    3 IF ( KTEST ) 5,4,4
    4 DUM = HP * ( V - A(2,I) ) / VDIF
      DTAN = DSIN(DUM) / DCOS(DUM)
      PELAST = VDIF * DTAN / A(6,I) + A(5,I)
      RETURN
C
    5 PELAST = 1./ ( A(1,I) * ( DCOS( HP * ( V-A(2,I) ) / VDIF ) )**2 )
      RETURN
      END
      SUBROUTINE FLOW ( I, J )
      IMPLICIT  REAL*8( A-H, O-Z )
      REAL*8 NW
      REAL*4 B, W
      COMMON /BLK3/ B(8,12), FS(12), RA(12), RB(12), W(12), DTM
      COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
     1    F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PBS,
     2    DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
     3    PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
     4    S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
     5    SQT(12), DP(12), HDT
C
      PDIF = P(I) - P(J)
      IF ( 2-I ) 1,2,2
    1 K = J
      IF ( J. EQ. 5. OR. J. EQ. 6 ) K = J + 2
      DELP = ( P(I) + P(J) )/2. - P(K)
      VAVG = ( V(I) + V(K) ) / 2.
      RA(J) = ( B(1,J) + B(2,J) * DELP ) / ( 1. + B(3,J) * VAVG )
      RB(J) = ( B(4,J) + B(5,J) * DELP + B(6,J) * (2. * PPL - DELP ) ) /
     1     (1. + B(7,J) * VAVG )
      GO TO 3
    2 RA(J) = B(1,J)
      RB(J) = B(4,J)
    3 SIGN = 1.
      IF( PDIF + W(J) * FS(J) / DTM. LT. 0. ) SIGN = -1.
      DUMY =( RA(J) + W(J) / DTM )**2 + 4. * SIGN * RB(J) *
     1         ( PDIF + W(J) * FS(J) / DTM )
      IF ( DUMY. GE. .0 ) GO TO 11
      WRITE (6,12) T, I, J, DUMY, RA(J), RB(J), PDIF, FS(J), SIGN
   12 FORMAT(2X, 'SQRT FAILURE IN FLOW,  T, I, J, DUMY, RA, RB,
     &    PDIF, FS(J), SIGN =',/,2X, F10.5, 2I3, 5E12.4, F5.1 )
        STOP 13

ready

*
```

```
   11 SQT(J) = DSQRT( DUMY )
      IF ( .1D-12 - ABS(PDIF) ) 6, 5, 5
    5 F(J) = 0.
      RETURN
C
    6 F(J) = ( -RA(J) - W(J) / DTM + SQT(J) ) / ( 2. * RB(J) * SIGN )
      RETURN
      END
      SUBROUTINE MOLES( F, DNT, NG, HDT, C, DC, DN, DDNW, CW, DCW, J, K)
      IMPLICIT  REAL*8( A-H, O-Z )
      DIMENSION  C(12,4), DC(12,4), CW(12), DCW(12), DN(12,4)
      COMMON /BLK12/ DIFF(6,3)
      DNT = 0.
      IF( F ) 5, 10, 15
    5 DO 6  I= 1, NG
      DN(K,I) = F * C(K,I)-DIFF(K-6,I)
    6 DNT = DNT + DN(K,I)
      DDNW = F * CW(K)
      RETURN
   10 DO 11  I= 1, NG
   11 DN(K,I) = 0.-DIFF(K-6,I)
      DDNW = 0.
      RETURN
   15 DO 16  I= 1, NG
      DN(K,I) = F * ( C(J,I) + HDT * DC(J,I) )-DIFF(K-6,I)
   16 DNT = DNT + DN(K,I)
      DDNW = F * ( CW(J) + HDT * DCW(J) )
      RETURN
      END

ready

*
```

```
      SUBROUTINE FORCE ( T, FSPR, DT )
      REAL*8 T, PTX, PD, DPTX, DPD, PHI, GF, GR, GT, GD, HP, PIE,
     1       ANG1, ANG2, PHI1, PHI2, FSPR, FSPRS, FCT, DT
      INTEGER FBOPT
      DIMENSION AA(8,2)
      COMMON /BLK4/ TIN, TEX, PDIN, TB, Y1, Y2, Y3, Y4, PTXIN, PTXEX,
     1              PDEX, PDAG, PTXG, TO, T2, PB, FCT, FSPRS, TTEST
      COMMON /BLK8/ PHI, GF, GR, GT, GD, HP, PIE, ANG1, ANG2, PHI1, PHI2
      COMMON /BLK11/ PTX, DPTX, PD, DPD, G1, G2, G3, G4, FBOPT
       DATA TO, TTEST, TBRTH, ITST, CONST, NFRC / 3* 0.0, 1, 1., 0 /,
     &    AA / .39, 4., .50, .25, .11, .30, 2.1, 3.,
     &         .43, 4., .23, .25, .36, .15, 1.5, 3. /
       DATA A, B, C, ALP, BET, GAM, DEL, EPS / .40, .60, 114.0, .25,
     &     80., 110., 1., 2. /, FMX / 1.0 /
C
       IF ( NFRC. EQ. 1 ) GO TO 105

ready

*
```

```
      IF ( T. LT. TTEST - .0001 )  GO TO 5
      IF ( ITST. EQ. 3 )  ITST = 1
      IF( ITST. EQ. FBOPT ) NFRC = 1
      IF ( NFRC. EQ. 1 )  GO TO 100
      ITST = ITST + 1
      B1 = ABS( PTXIN - PTXEX ) / PB
      B3 = ABS ( PDIN - PDEX ) / PB
      ASQR1 = TB * TB
      TBS = TB
      B1S = B1
      DO 1  J= 1,50
      IF ( ITST. EQ. 2 ) DPTX = ( PTXIN + B1S - PTXEX ) * Y1 / ( 1.-
     1    EXP( -Y1 * ( TIN - TBP ) )  )
      IF ( ITST. EQ. 3 ) DPTX = Y2 * ( PTXIN + B1S - PTXEX) / ( 1.-
     1    DEXP( -Y2 * ( TEX - TBP ) )  )
      TBP = ABS( DPTX ) * ASQR1 / SQRT( B1*B1 + ASQR1 * DPTX*DPTX )
      ASQR = TBP * TBP
      DUMY = SQRT( 1.- ASQR / ASQR1 )
      BTX = B1 - B1 * DUMY
      IF ( ABS( BTX - B1S ). LT. 0.001 ) GO TO 2
      B1S = BTX
    1 TBS = TBP
    2 BDA = B3 - B3 * DUMY
      C1 = PTXEX - BTX
      C2 = PTXIN + BTX
      C3 = PDEX - BDA
      C4 = PDIN + BDA
      BA1 = BTX / ASQR
      BA3 = BDA / ASQR
      PTXDM = PTXIN + BTX
      PDDM = PDIN + BDA
      IF ( ITST - 3 ) 6, 7, 7
    6 PIN2 = ( PTXDM - PTXEX ) / ( 1.- DEXP( -Y1 * ( TIN - TBP )  ) )
      PIN3 = ( PDDM - PDEX ) / ( 1.- EXP( -Y3 * ( TIN - TBP ) )  )
      TTEST = TTEST + TIN
      TO = TO + TBRTH
      T1 = TO + TBP
      T2 = TO + TIN
      T3 = T2 + TBP
      TBRTH = TEX + TIN
      TOP = TO - .0001
      T1P = T1 - .0001
      T2P = T2 - .0001
      T3P = T3 - .0001
      GO TO 8
    7 IF ( FBOPT. LT. 10 ) GO TO 71
      T3P = T2P
      T3 = T2
      TBP = 0.
      PTXDM = PTXIN

ready

*
```

```
      PDDM = PDIN
      PEX2 = 1./ (.455* ( 1.-EXP( -1.95 * TEX ) ) + .335* ( 1. -
     &       EXP( -.1 * TEX**2 ) ) + .21* ( 1.- EXP( -.11* TEX**3) ) )
      PEX3 = ( PDDM - PDEX ) * PEX2
      PEX2 = ( PTXDM - PTXEX ) * PEX2
      WRITE (6,1221) PEX2,PEX3,PTXDM,PDDM
 1221 FORMAT( 1X, 'PEX2, PEX3 =', 4E15.4 )
      GO TO 72
   71 PEX2 = ( PTXDM - PTXEX ) / ( 1.- DEXP( -Y2 * ( TEX - TBP ) ) )
      PEX3 = ( PDDM  - PDEX  ) / ( 1.- DEXP( -Y4 * ( TEX - TBP ) ) )
   72 TTEST = TTEST + TEX
    8 CONTINUE
C
    5 CONTINUE
      IF ( NFRC. EQ. 1 )  GO TO 100
      IF ( T. GE. T3P ) GO TO 75
      IF ( T. GE. T2P ) GO TO 50
      IF ( T. GE. T1P ) GO TO 25
C
C  INSPIRATION:  PART 1
C
      TDIF = T - TO
      TSQR = TDIF * TDIF
      D1 = SQRT( 1. - TSQR / ASQR1 )
      DUM = TDIF / ( D1 * ASQR1 )
      PTX = PTXEX - B1 + B1 * D1 + PTXG * GT
      DPTX = -B1 * DUM
      PD = PDEX - B3 + B3 * D1 + PDAG * GD
      DPD = -B3 * DUM
      RETURN
C
C  INSPIRATION:  PART 2
C
   25 D1 = EXP( -Y1 * ( T - T1 ) )
      D3 = EXP( -Y3 * ( T - T1 ) )
      PTX = C1 + PIN2 * ( 1.- D1 ) + PTXG * GT
      DPTX = PIN2 * Y1 * D1
      PD = C3 + PIN3 * ( 1.- D3 ) + PDAG * GD
      DPD = PIN3 * Y3 * D3
      RETURN
C
C  EXPIRATION: PART 1
C
   50 IF ( FBOPT. EQ. 10 )  GO TO 200
      TDIF = T - T2
      TSQR = TDIF * TDIF
      D1 = SQRT( 1.- TSQR / ASQR1 )
      DUM = TDIF / ( D1 * ASQR1 )
      PTX = PTXIN + B1 - B1 * D1 + PTXG * GT
      DPTX = B1 * DUM
```

ready

*

```
      PD = PDIN + B3 - B3 * D1 + PDAG * GD
      DPD = B3 * DUM
      RETURN
C
C EXPIRATION:  PART 2
C
 75 IF ( FBOPT. EQ. 10 )  GO TO 200
      D1 = EXP( -Y2 * ( T - T3 ) )
      D3 = EXP( -Y4 * ( T - T3 ) )
      PTX = PTXDM - PEX2 * ( 1.- D1 ) + PTXG * GT
      DPTX = -PEX2 * Y2 * D1
      PD = PDDM - PEX3 * ( 1.- D3 ) + PDAG * GD
      DPD = -PEX3 * Y4 * D3
      RETURN
C
100 CONTINUE
      IF ( T. LT. TTEST - .0001) GO TO 105
      TN = TTEST
      TTEST = TTEST + TEX
      COR = A * ( 1. - EXP( -ALP * EXP( EPS * ALOG( TEX ) ) ) ) +
     &       B * ( 1. - EXP( -BET * TEX ) ) +
     &       C * TEX * EXP( -GAM * EXP( DEL * ALOG( TEX ) ) )
      PTXDIF = ( PTXEX - PTXIN ) / COR
      PDDIF  = ( PDEX - PDIN ) / COR
C
105 CONTINUE
      TT = T - TN
      ALNT = ALOG( TT )
      TEPSM= EXP( ( EPS -1. ) * ALNT )
      TEPS = EXP( EPS * ALNT )
      TDEL = EXP( DEL * ALNT )
      EXP1 = EXP( -ALP * TEPS )
      EXP2 = EXP( -BET * TT )
      EXP3 = EXP( -GAM * TDEL )
      FCT = A * ( 1.- EXP1 ) + B * ( 1.- EXP2 ) + C * TT * EXP3
      DFCT = A * ALP * EPS * TEPSM * EXP1 + B * BET * EXP2 +
     &        C * EXP3 * ( 1.- DEL * GAM * TDEL )
      PTX = PTXG * GT + PTXIN + PTXDIF * FCT
      PD  = PDAG * GD + PDIN + PDDIF * FCT
      DPTX= DFCT * PTXDIF
      DPD = DFCT * PDDIF
      RETURN
200 CONTINUE
      TDIF = T - T2
      TT = TDIF - 2.10
      PTX = .455 * ( 1.- EXP( -1.95 * TDIF ) ) + .335 * ( 1.- EXP( -.1 *
     &  TDIF**2 ) )
      DPTX = .88725 * EXP ( -1.95 * TDIF ) + .067 * TDIF * EXP (-.1
     &   * TDIF*TDIF )
      IF (TT) 90,90,85

ready

*
```

```
   85 PTX = PTX + .20 * ( 1. - EXP(-.11 * TT**3 ) )
      DPTD = DPTX + .066 * TT*TT * EXP(-.11 * TT**3 )
   90 PD = PDDM -PEX3 * PTX + PDAG * GD
      PTX = PTXDM - PEX2 * PTX + PTXG * GT
      DPD = -DPTX * PEX3
      DPTX = -PEX2 * DPTX
C      IF(T.GE.1.55) WRITE(6,1222) T, PTX, DPTX, PD, DPD
      RETURN
 1222 FORMAT(1X, 'T, PTX, DPTX, PD, DPD =' ,F5.2, F7.2, E10.2, F7.2,
     &     E10.2 )
      END
      SUBROUTINE AIRWAY ( DN, DDCW )
      IMPLICIT REAL*8( A-H, O-Z )
      REAL*8 NW
      DIMENSION X(5), DDCW(12), FI(6,4), FO(6), A(5,5), FIW(6), E(5),
     1     DN(12,4)
      COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
     1     F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PES,
     2     DPES, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
     3     PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
     4     S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
     5     SQT(12), DP(12), HDT
      DO 10  J= 1,37
   10 FIW(J) = 0.
      DO 20  J= 2,6
      DNT(J) = 0.
      DO 20  I=1,NG
   20 FI(J,I) = 0.
      IF( F(2) ) 30, 50, 40
   30 FO(2) = -F(2)
      GO TO 50
   40 DO 45  I= 1, NG
   45 FI(2,I) = F(2) * C(1,I)
      FIW(2) = F(2) * CW(1)
   50 DO 80  J= 3,6
      K3 = J-1
      K = J-3
      IF ( J. EQ. 3 )  K= 1
      K2 = K + 1
      IF( F(J) ) 60, 80, 70
   60 FO(J) = -F(J)
      A(K,K3) = -F(J)
      DO 65  I= 1, NG
   65 FI(K2,I) = FI(K2,I) - F(J) * C(J,I)
      FIW(K2) = FIW(K2) - F(J) * CW(J)
      GO TO 80
   70 FO(K2) = FO(K2) + F(J)
      A(K3,K) = F(J)
      DO 75 I= 1, NG
   75 FI(J,I) = F(J) * C(K2,I)
```

ready

*

```
      FIW(J) = F(J) * CW(K2)
   80 CONTINUE
  170 DO 230  J= 7,12
      K = J-4
      IF ( J. GE. 11 )  K= J-6
      IF( F(J) ) 210, 230, 220
  210 DO 215  I= 1, NG
  215 FI(K,I) = FI(K,I) - F(J) * C(J,I)
      FIW(K) = FIW(K) - F(J) * CW(J)
      GO TO 230
  220 FO(K) = FO(K) + F(J)
  230 CONTINUE
  350 DO 360  K= 1, 5
  360 A(K,K) = -( V(K+1) / HDT + FO(K+1) )
      DO 380  I= 1, NG
      DO 370  K= 2, 6
  370 B(K-1) = ( C(K,I) * FO(K) - FI(K,I) ) / HDT
 3003 FORMAT ( /, 1H , 5(1X, 5E12.6, 4X, E12.6, //) )
      CALL BAPM( A, X, B )
      DO 375  J= 2, 6
      DC(J,I) = X(J-1)
      DN(J,I) = DC(J,I) * V(J)
  375 DNT(J) = DNT(J) + DN(J,I)
 3004 FORMAT(5X, 5E15.6 )
  380 CONTINUE
      DO 390  J= 2, 6
  390 B(K-1) = ( CW(K) * FO(K) - FIW(K) ) / HDT
      CALL BAPM( A, X, B )
      DO 400  J= 2, 6
  400 DDCW(K) = X(K-1)
      RETURN
      END
      SUBROUTINE PMAX ( FRCP, VMAX, PG, PBS, GT, GD, PGAB,
     1              FCTN, VTX, VD, PPL, VA )
      REAL*8 PG, PBS, GT, GD, PGAV, DPGAV, PGAVB, DPGAVB, FCTST,
     1        V, VTX, VD, VDM, VTXM, PELAST, PGAB, PPL, VA, FSPRS
      REAL*8 DP1(6),DUMYS
      DIMENSION PG(6), A(7), B(7), C(7), XP(7), DELB(7), FACT(7),
     1           KNT2(7)
      COMMON /BLK4/ TIN, TEX, PDIN, TB, Y1, Y2, Y3, Y4, PTXIN, PTXEX,
     1                PDEX, PDAG, PTXG, TO, T2, PB, FCTST, FSPRS, TTEST
      COMMON /BLK5/ AA(7,8)
      COMMON /BLK10/ PGAV, DPGAV, PGAVB, DPGAVB
C
      VLG = FRCP + VMAX - VA
      PPLM = PPL
      XP(7) = PPLM
      DO 25  J=1,6
      XP(J) = VLG / 6.
      C(J) = PG(J) - PBS
```

ready

*

```
   25 CONTINUE
      C(7) = VLG
      DO 35  J= 1, 7
      FACT(J) = .5
   35 KNT2(J) = 0
C
      DO 1000  KNT = 1, 30
      SB = 0.
      SV = 0.
      SA = 0.
      SAW = 0.
      DO 50  J= 1,6
      JDUMY = J
      V = XP(J)
      B(J) = ( PELAST( 0, JDUMY, V ) + XP(7) + C(J) )
      A(J) = PELAST( -1, JDUMY, V )
      SB = SB + ABS( B(J) )
      SV = SV + XP(J)
      SA = SA + 1./ A(J)
      SAW = SAW + B(J) / A(J)
   50 CONTINUE
      B(7) = ( SV - C(7) )
      SB = SB + ABS( B(7) )
C
      IF ( SB. LT. 0.01 )  GO TO 1005
C
      B(7) = ( B(7) - SAW ) / SA
      DO 55  J= 1,6
   55 B(J) = -( B(J) + B(7) ) / A(J)
C
      DO 60  J= 1,7
      IF ( KNT2(J). EQ. 1 ) GO TO 206
      IF ( DELB(J). EQ. 0.. OR. B(J). EQ. 0. )  GO TO 206
      IF ( DELB(J) / B(J) )  205, 205, 204
  204 IF ( KNT2(J). LT. 3 ) GO TO 206
      FACT(J) = 1.0
      GO TO 206
  205 FACT(J) = 0.5
      KNT2(J) = 0.0
  206 B(J) = FACT(J) * B(J)
      DELB(J) = B(J)
      IF ( J. EQ. 7 ) GO TO 60
  209 DUMY = ABS( DELB(J) )
      IF ( DUMY. GT. 0.2 )  DELB(J) = 0.2 * DUMY / DELB(J)
   60 XP(J) = XP(J) + DELB(J)
      DO 70  J= 1,6
   70 IF ( XP(J). GE. AA(3,J) ) XP(J) = AA(3,J) - .01
 1000 CONTINUE
      WRITE(6,98) XP
      WRITE(6,99) KNT, SB
```

ready

*

```
   99 FORMAT( 5X, 'KNT =',I3, 5X, 'SB =', E10.3 )
   98 FORMAT(5X, 9E11.4 )
C
      STOP 3
C
 1005 CONTINUE
  111 FORMAT( 5X, 7E15.8 )
      PPLM = XP(7)
      C(1) = PBS + PTXG * GT - PGAV - PPLM
      C(2) = PBS + PDAG * GD - PGAVB - PPLM - PGAB
      C(3) = FCTN
      C(4) = VLG + .005
      ADUM1 = AA(3,7) - .001
      ADUM2 = AA(4,7) + .001
      ADUM3 = AA(3,8) - .001
      ADUM4 = AA(4,8) + .001
      VTXM = VTX + .5 * VMAX
      IF ( VTXM. GE. ADUM1 )  VTXM = ADUM1
      IF ( VTXM. LE. ADUM2 )  VTXM = ADUM2
      VDM = C(4) - VTXM
      IF ( VDM - ADUM3 ) 1205, 1201, 1201
 1201 VDM = ADUM3
      GO TO 1209
 1205 IF ( VDM - ADUM4 )  1206, 1206, 1210
 1206 VDM = ADUM4
 1209 VTXM = C(4) - VDM
      IF ( VTXM. LT. ADUM2. OR . VTXM. GT. ADUM1 )  GO TO 2001
 1210 CONTINUE
C
      DO 2000 KNT = 1,50
      VTXMN = VTXM + ( C(3) * ( PELAST( 0, 8, VDM ) - C(2) ) -
     1    PELAST( 0, 7, VTXM ) + C(1) ) / ( C(3) * PELAST( -1, 8, VDM )
     2    + PELAST( -1, 7, VTXM ) )
      VDM = C(4) - VTXMN
      IF( ( VTXMN. LE. ADUM2. OR. VTXMN. GE. ADUM1 ). AND .
     1    ( VDM. LE. ADUM4. OR. VDM. GE. ADUM3 ) )  GO TO 2001
      IF ( ABS( VTXMN - VTXM ). LT. 0.0005 ) GO TO 2005
      VTXM = VTXMN
 2000 CONTINUE
C
 2001 WRITE(6,101) VDM, VTX
  101 FORMAT(//5X, '******** VDM = ',F10.5, '   VTX = ',F10.5,
     1    'ONE OF THE ABOVE IS TO LARGE, CHECK A(7,8) AND FCTN')
      STOP 4
 2005 CONTINUE
      WRITE(6,99) KNT
      VTXM = VTXMN
      DUM1 = PELAST( 0, 7, VTXM )
      DUM2 = PELAST( 0, 8, VDM )
      DUM3 = PELAST( 0, 7, VTX )

ready

*
```

```
      DUM4 = PELAST( 0, 8, VD )
      CTXM = 1./ PELAST( -1, 7, VTXM )
      CDAM = 1./ PELAST( -1, 8, VDM )
  988 FORMAT(2X, 5E12.6 )
      PDIN = DUM2 - C(2)
      PTXIN = DUM1 - C(1)
      WRITE(6,989) DUM1, PPLM, PGAV, PTXG, GT,
     1    DUM2, PGAB, PDAG, GD, PGAVB
  989 FORMAT( 2X, 'PELTXM, PPLM, PGAV, PTXG, GT, PELDAM, PGAB, PDAG, GD,
     1    PGAVB = ',/,3X, 5E12.6, / 5E12.6, // )
      WRITE( 6,97 ) PTXIN, PDIN, VTXM, VDM, CTXM, CDAM
   97 FORMAT( //, 7X, 'PTXIN          PDIN              VTXM              VDM',
     1        '             CTXM            CDAM',//,1X,6E15.8, //)
  997 FORMAT( 5F10.5 )
      DO 7711 J=1,6
      JD = J
      DUMYS = XP(J)
      DP1(J) = PELAST(0,JD,DUMYS)
 7711 CONTINUE
      WRITE (6,7712) (DP1(J),J=1,6)
 7712 FORMAT(//,1X,'PELAST =', 6F9.3 ,//)
      RETURN
      END
      FUNCTION FRC2( VA, VTX, PD, PTX )
      IMPLICIT REAL*8 ( C-H, O-V )
      REAL*8 NW
      COMMON /BLK5/ AA(7,8)
      COMMON /BLK10/ PGAV, DPGAV, PGAVB, DPGAVB
      COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
     1    F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PBS,
     2    DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
     3    PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
     4    S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
     5    SQT(12), DP(12), HDT
      DIMENSION A(9), B(9), XP(9), ZPK(9), KNT2(9),
     1    DELB(9), FACT(9)
      VA = 0.
      DO 15  I= 2,6
   15 VA = VA + V(I)
      FRC2 = VA
      XP(9) = PPL
      XP(7) = VTX
      XP(8) = VD
      DO 30  J= 7,12
      XP(J-6) = V(J)
      ZPK(J-6) = P(J) - PG(J-6)
  901 FORMAT( 5X, 'V,P,PG,ZPK',I2,'=',4E15.7 )
   30 CONTINUE
      ZPK(7) = -PGAV + PTX + PBS
      ZPK(8) = PD + PBS - PGAB - PGAVB

ready

*
```

```
      ZPK(9) = 0.005
  210 FORMAT (3X, 9E11.5 )
      DO 35  J= 1,9
      FACT(J) = .5
   35 KNT2(J) = 0.
C
      DO 1000  KNT = 1, 100
      SA = 0.
      SAW = 0.
      VLUNG = 0.0
      DO 40  J= 1,6
      JDUMY = J
      VLUNG = VLUNG + XP(J)
      A(J) = PELAST( -1, JDUMY, V(J+6) )
      PTEST = PELAST(1, JDUMY, V(J+6) )
      B(J) = ( XP(9) - ZPK(J) + PTEST )
      SA = SA + 1./ A(J)
      SAW = SAW + B(J) / A(J)
  902 FORMAT( 1X, 'A,B,XP1,ZPK,PEL',12,'=',5E17.10 )
   40 CONTINUE
      A(7) = PELAST( -1, 7, VTX )
      A(8) = PELAST( -1, 8, VD )
      B(7) = ( XP(9) - ZPK(7) + PELAST( 1, 7, VTX ) )
      B(8) = ( XP(9) - ZPK(8) + PELAST( 1, 8, VD ) )
      B(9) = ( VTX + VD - ZPK(9) - VLUNG )
      SB = 0.
      DO 50  J= 1,9
   50 SB = SB + ABS( B(J) )
      IF ( KNT. GE. 100 ) WRITE (6,98) B
      IF ( SB. LT. 0.0003 )  GO TO 1001
C
      B(9) = ( B(9) - B(7) / A(7) - B(8) / A(8) + SAW ) / ( 1./ A(7)
     1         + 1./ A(8) - SA )
      DO 55  J=1,8
   55 B(J) = -( B(J) + B(9) ) / A(J)
C
      IF ( KNT. GE. 100 )  WRITE(6,98) B
      DO 60  J= 1,9
      IF ( KNT2(J). EQ. 1 ) GO TO 206
      IF ( DELB(J). EQ. 0.. OR. B(J). EQ. 0. )  GO TO 206
      IF ( DELB(J) / B(J) )  205, 205, 204
  204 IF ( KNT2(J). LT. 3 ) GO TO 206
      FACT(J) = 1.0
      GO TO 206
  205 FACT(J) = 0.5
      KNT2(J) = 0.0
  206 B(J) = FACT(J) * B(J)
      DELB(J) = B(J)
      IF ( J. EQ. 9 ) GO TO 60
      IF ( J-7 ) 207, 208, 208
```

ready

*

```
  207 TEST = 0.2
      GO TO 209
  208 TEST = 0.5
  209 DUMY = ABS( DELB(J) )
      IF ( DUMY. GT. TEST )  DELB(J) = TEST * DUMY / DELB(J)
   60 XP(J) = XP(J) + DELB(J)
      DO 70 J= 1,6
      IF ( XP(J). LE. AA(4,J) )  XP(J) = AA(4,J)
      IF ( XP(J). GE. AA(3,J) )  XP(J) = AA(3,J)
   70 V(J+6) = XP(J)
      PPL = XP(9)
      IF ( XP(7). LE. AA(4,7) )  XP(7) = AA(4,7)
      IF ( XP(7). GE. AA(3,7) )  XP(7) = AA(3,7)
      VTX = XP(7)
      IF ( XP(8). LE. AA(4,8) )  XP(8) = AA(4,8) + .0002
      IF ( XP(8). GE. AA(3,8) )  XP(8) = AA(3,8) - .0002
      VD= XP(8)
  111 FORMAT(1X,'DELB =',5E17.10,/7X,4E17.10,/'   XP =',5E17.10,/
     1     ,7X,4E17.10,/4X,'V =',6E17.10 )
   99 FORMAT( 5X, 'KNT =',I3, 5X, 'SB =', E10.3 )
   98 FORMAT(5X, 9E11.4 )
 1000 CONTINUE
      WRITE(6,99) KNT, SB
      WRITE(6,98) B, XP
C
      STOP 1000
 1001 CONTINUE
      PELTX = PELAST(1, 7, VTX )
      PELDA = PELAST(1, 8, VD )
      WRITE(6,99) KNT, SB
      FRC2 = FRC2 + VLUNG
      VLUNG = FRC2
      PPL = XP(9)
      VTX = XP(7)
      VD = XP(8)
      RETURN
      END

ready

*
```

```
      SUBROUTINE SUB1 ( I, J )
      REAL*8 NW
      IMPLICIT  REAL*8( A-H, O-Z )
      COMMON /BLK6/ A(5,5), FO(6), FI(6), SQO(6), SQI(6)
      COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
     1    F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PBS,
     2    DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
     3    PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
     4    S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
     5     SQT(12), DP(12), HDT
C
       IF( J. GT. 2 ) GO TO 50
       IF( F(J) )  10, 20, 30
   10 FO(2) = -F(2)
       SQO(2) = +1./ SQT(2)
   20 RETURN
   30 FI(2) = CT(1) * F(2)
       SQI(2)= CT(1) / SQT(2)
       RETURN
   50 IF( J. GT. 6 ) GO TO 100
C  *** AIRWAYS
       IF( F(J) ) 60, 70, 80
   60 FO(J) = FO(J) - F(J)
       FI(I) = FI(I) - F(J) * CT(J)
       SQO(J)= SQO(J) + 1./ SQT(J)
       SQI(I)= SQI(I) + CT(J) / SQT(J)
       A(I-1,J-1) = -F(J) * ( 1./G - DCWDP(J) ) + CT(J) / SQT(J)
       A(J-1,I-1) = -CT(J) / SQT(J)
   70 RETURN
   80 FI(J) = FI(J) + F(J) * CT(I)
       FO(I) = FO(I) + F(J)
       SQI(J) = SQI(J) + CT(I) / SQT(J)
       SQO(I) = SQO(I) + 1./ SQT(J)
       A(I-1,J-1) = CT(I) / SQT(J)
       A(J-1,I-1) = F(J) * ( 1./G - DCWDP(I) ) + CT(I) / SQT(J)
       RETURN
C *** LUNG CHAMBERS
  100 IF( F(J) ) 110, 120, 130
  110 FI(I) = FI(I) - F(J) * CT(J)
       SQI(I) = SQI(I) + CT(I) / SQT(J)
  120 RETURN
  130 FO(I) = FO(I) + F(J)
       SQO(I) = SQO(I) + 1./ SQT(J)
       RETURN
       END
       SUBROUTINE SUB2( ASUM, ASUM1, AWSUM, AWSUM1, RES, PV, RTOT, PPLC )
       IMPLICIT  REAL*8( A-H, O-Z )
       REAL *8 N, NW
       REAL*4 B, W, ASUM, ASUM1, AWSUM, AWSUM1, RES, PV, RTOT, PPLC
       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
      COMMON /BLK3/ B(8,12), FS(12), RA(12), RB(12), W(12), DTM
      COMMON C(12,4), DC(12,4), CT(12), CW(12), DCW(12), DCWDP(12),
     1    F(12), G, GS, NG, DNT(12), NW(12), DNW(12), P(12), PBS,
     2    DPBS, PELALV(6), DPELAL(6), PELDA, DPELDA, PELTX, DPELTX,
     3    PG(6), DPG(6), PGAB, DPGAB, PPL, DPS(6), DPS2,
     4    S(4), T, DT, TN(12), TOTN(12), V(12), VD, VLUNG,
     5    SQT(12), DP(12), HDT
      DO  160  I = 1,NG
  160 ASUM(I) = -ASUM1(I)
      AWSUM = -AWSUM1
      DO  170  J= 1,12
      AWSUM = AWSUM + NW(J)
      DO  170  I= 1,NG
  170 ASUM(I) = ASUM(I) + N(J,I)
      DO 180  I = 2,12
      RES(I) = RA(I) + RB(I) * ABS( F(I) )
  180 PV(I) = P(I) * V(I) / ( TOTN(I) * G ) - 1.
      PV(1) = P(1) * V(1) / ( TOTN(1) * GS) - 1.
      RR1 = RES(6) + RES(10) * RES(12) / ( RES(10) + RES(12) )
      RL1 = RES(5) + RES( 9) * RES(11) / ( RES( 9) + RES(11) )
      RR  = RES(4) + RES( 8) * RR1     / ( RES( 8) + RR1 )
      RL  = RES(3) + RES( 7) * RL1     / ( RES( 7) + RL1 )
      RTOT  = RES(2) + RR * RL / ( RR + RL )
      PPLC(1) = PPL + PG(1)
      PPLC(2) = PPL + PG(3)
      PPLC(3) = PPL + PG(5)
      RETURN
      END
      SUBROUTINE SUB3( V )
      REAL*8 V
      DIMENSION V(12)
      V(7) = 1.030116
      V(8) = V(7)
      V(9) = 0.95278
      V(10) = V(9)
      V(11) = 0.887373
      V(12) = V(11)
      RETURN
      END
```

ready

*

```
      SUBROUTINE BAPM( A, X, B )
      IMPLICIT REAL*8( A-H, L, M, O-Z )
      DIMENSION A(5,5), X(5), B(5), D(5), L(5), G(5)
      D(1) = A(1,2) / A(1,1)
      L(1) = A(1,3) / A(1,1)
      G(1) = B(1) / A(1,1)
      M = A(2,2) - A(2,1) * D(1)
      D(2) = -A(2,1) * L(1) / M
      L(2) = A(2,4) / M
       G(2) = ( B(2) - A(2,1) * G(1) ) / M
       Z = -A(3,1) * D(1)
       M = A(3,3) - Z * D(2) - A(3,1) * L(1)
       D(3) = -Z * L(2) / M
       L(3) = A(3,5) / M
       G(3) = ( B(3) - Z * G(2) - A(3,1) * G(1) ) / M
       Z = -A(4,2) * D(2)
       M = A(4,4) - Z * D(3) - A(4,2) * L(2)
       D(4) = -Z * L(3) / M
       G(4) = ( B(4) - Z * G(3) - A(4,2) * G(2) ) / M
       Z = -A(5,3) * D(3)
       M = A(5,5) - Z * D(4) - A(5,3) * L(3)
       X(5) = ( B(5) - Z * G(4) - A(5,3) * G(3) ) / M
       X(4) = G(4) - D(4) * X(5)
       X(3) = G(3) - D(3) * X(4) - L(3) * X(5)
       X(2) = G(2) - D(2) * X(3) - L(2) * X(4)
       X(1) = G(1) - D(1) * X(2) - L(1) * X(3)
       RETURN
       END
       SUBROUTINE RUNGE( N, Y, F, X, H, M, HDT )
       IMPLICIT  REAL*8(A-H, O-Z)
       DIMENSION PHI(100), SAVEY(100), Y(N), F(N)
       GO TO (1,2,3,4), M
     1 DO 11  J= 1,N
       SAVEY(J)= Y(J)
       PHI(J)= F(J)
    11 Y(J)= SAVEY(J) + 0.5*H*F(J)
       X= X + 0.5*H
       RETURN
     2 DO 22 J= 1,N
       PHI(J)= PHI(J) + 2.0*F(J)
    22 Y(J)= SAVEY(J) + 0.5*H*F(J)
       HDT = H/2.
       RETURN
     3 DO 33  J= 1,N
       PHI(J)= PHI(J) + 2.0*F(J)
    33 Y(J)= SAVEY(J) + H*F(J)
       X= X + 0.5*H
       RETURN
     4 DO 44  J= 1,N
    44 Y(J)= SAVEY(J) + (PHI(J) + F(J))*H/6.0

ready

*
```

```
      HDT = H/4.
      RETURN
      END
      FUNCTION HAMING( N, Y, F, X, H, TE )
      IMPLICIT  REAL*8(A-H, O-Z)
      INTEGER HAMING
      LOGICAL  PRED
      DIMENSION YPRED(100), TE(N), Y(N,4), F(N,3)
      DATA  PRED / .TRUE. /
      IF (.NOT.PRED)  GO TO 4
C     ..... PREDICTOR SECTION OF HAMING .....
      DO 1  J= 1,N
    1 YPRED(J)= Y(J,4) + C1*( 2.* ( F(J,1) + F(J,3) ) - F(J,2) )
      DO 2  K5= 1,3
      K= 5 - K5
      KM1 = K - 1
      DO 2  J= 1,N
      Y(J,K)= Y(J,KM1)
    2 IF (K.LT.4) F(J,K)= F(J,KM1)
      DO 3  J=1,N
    3 Y(J,1)= YPRED(J) + C2 * TE(J)
      X= X+ H
      PRED= .FALSE.
      HAMING= 1
      RETURN
C     ..... CORRECTOR SECTION OF HAMING .....
    4 DO 5  J= 1,N
      Y(J,1) = ( 9.* Y(J,2) - Y(J,4) + C4 * ( F(J,1) + 2.* F(J,2) -
     1           F(J,3) ) ) / 8.
      TE(J)= C5 * (Y(J,1) - YPRED(J))
    5 Y(J,1) = Y(J,1) - TE(J)
      PRED= .TRUE.
      HAMING = 2
      RETURN
C
C  THIS SECTION CALCULATES CONSTANTS USED IN HAMING AND MUST BE CALLED
C  ONLY ONCE BEFORE THE ABOVE SECTION IS USED.  IT NEED NOT BE CALLED
C  AGAIN UNLESS THE INTEGRATION STEP IS CHANGED.
C
      ENTRY HAMNG1( N, Y, F, X, H, TE )
      C1 = 4.* H / 3.
      C2 = 112./ 9.
      C4 = 3.* H
      C5 = 9./ 121.
      HAMING = 0
      RETURN
      END
```

ready

*

THE FOLLOWING ARE THE DATA FILES:

```
0010#    1    1.0E-06 1.934E+04

ready

*

10#0.148E+  .963E-02 .0024E+  0.000E+  0.294E+  0.294E+  0.294E+  0.294E+
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+  0.327E+  .0606E+
40#0.552E+  0.243E+  0.056E+  0.092E+  1.000E+  1.000E+  1.000E+  1.000E+
50#1.000E+  1.000E+  0.368E+  0.081E+  0.150E+  0.023E+  1.000E+  1.000E+
60#1.000E+  1.000E+  0.552E+  0.552E+  0.552E+  0.552E+  0.552E+  0.552E+
70     E+     -4.E+    -4.E+    -4.E+    -0.E+    -0.E+    -0.E+    -0.E+
80  -0.E+     -0.E+    -4.E+    -4.E+    -4.E+    -4.E+      E+       E+
90     E+       E+       E+       E+       E+       E+       E+       E+
100#   358.0    373.0      0.0    647.0      0.0      0.0      0.0      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      0.0      0.0     18.0      0.0      0.0      0.0      0.0
140#     0.0      0.0      6.0      0.0      0.0     60.0      0.0      0.0
150#     0.0      0.0     10.0     10.0     10.0     10.0      7.0      4.0
160#   941.0    83.39    125.0    290.6    23.70    23.70    23.70    23.70
170#   23.70    23.70    262.9   102.24    125.0    778.9    55.20    56.20
180#   125.2    20.00    277.2    282.7    624.1     69.8    378.3    378.8
190#945.       40.     125.     285.     11.7     11.7     11.7     11.7
200#11.7       11.7    256.      70.     125.     750.      35.2     36.2
210#105.2      0.0     277.0    278.     625.0     70.0    377.0    378.
220  0.00    1667.0   000.      00.0     333.     0.0
230
240
250#1.0 .738
260#0.588E+  0.588E+  0.588E+  0.588E+  0.588E+  0.588E+  7.190E+
270#9.600E+  1.760E+  8.120E+  6.620E+
280+ 1.0E+  +0.00E+  -83.3E+  + 1.0E+  +0.00E+  -83.3E+
290 4.00E+01 4.00E+01 4.00E+01 4.00E+01 9.77E+01 9.77E+01 9.77E+01 9.77E+01
300 9.77E+01 9.77E+01 9.31E+01 9.31E+01 9.31E+01 9.31E+01 4.00E+01 4.00E+01
310 4.00E+01 4.00E+01 4.00E+01 4.00E+01 4.00E+01 4.00E+01 4.00E+01 4.00E+01
320 4.50E+01 4.50E+01 4.50E+01 4.50E+01 4.02E+01 4.02E+01 4.02E+01 4.02E+01
330 4.02E+01 4.02E+01 4.03E+01 4.03E+01 4.03E+01 4.03E+01 4.50E+01 4.50E+01
340 4.50E+01 4.50E+01 4.50E+01 4.50E+01 4.50E+01 4.50E+01 4.50E+01 4.50E+01
350
360
370
380

ready

*
```

```
390
400
410
420
430
440 1.00E+00 1.00E+00                           1.08E-06 2.57E-05
450 36.8E+00 36.8E+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 9.38E+00
530 99.3E+00 99.3E+00 99.3E+00 99.3E+00 99.3E+00 99.3E+00 3.17E+00 17.3E+00
540 3.75E+00 4.02E+00
550
560
570
580
590 20.6E+00 1.13E+02 24.4E+00 30.0E+00
600 16.5E+00 90.0E+00 19.5E+00 24.0E+00
610 0.01     20.0
620#10 1 1 1.1

ready

*


10  50   1
20   0.0     .0100       20.00      2.000     3.000     10 0 310.  298.
30  753.2      00.0        0.0      62.36       0.95       2.00       2.00
40    6.2117    0.0      00.0        .0        1.2        0.60       1.00
50    0.0        1.00      2.00       2.00       .10       6.
60   10.0        0.050     0.025      0.025      0.050      0.050
70     .52       0.52      0.36       0.38       0.27       0.27
80  760.0      760.0     760.0      760.0      760.0      760.0
90  760.0      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#     .0004      .0561      .0561      .0561      .0561      .0561
130#     .0561      .0561      .0561      .0561      .0561      .0561
140#     .7902      .7980      .7980      .7980      .7980      .7980

ready

*
```

```
150#    .7980     .7980     .7980     .7980     .7980     .7980
160
170
180.   0.009     0.0       0.0       0.160     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    -0.006               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    2.080    -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

*
```

REFERENCES

(1) Noordergraaf, A., Jager, G. N., and Westerhof, N. Circulatory Analog Computers. North-Holland Publishing Co., Amsterdam, 1963.

(2) Reeve, E. G. and Guyton, A. C. Physical Bases of Circulatory Transport: Regulation and Exchange. W. B. Saunders Co., Philadelphia, Pennsylvania, 1967.

(3) Beneken, J.E.W. Investigation of the regulatory system of the blood circulation, Circulatory Analog Computers, 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, Proc. 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. Cardiovasc. Dis. 5: 419-439, 1963.

(6) Noordergraaf, A. Development of an analog computer of the human circulatory system. In Circulatory Analog Computers, 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, Pusatile Blood Flow, Ed. by Attinger, E. O., McGraw-Hill, New York, 1964.

(8) Robinson, D. A. Ventricular dynamics and the cardiac representation problem, Circulatory Analog Computers 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, Circulatory Analog Computers, 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 Pulsatile Blood Flow. 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 Medical Electronics. 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 Physical Bases of Circulatory Transport: Regulation and Exchange. Ed. by Reeve, E. G. and Guyton, Arthur C. W. B. Saunders, 1967, pp. 1-45.

(13) Grodins, F. S. and Buoncristiani, J. F. General formulation of the cardiovascular control problem-mathematical models of the mechanical system, Physical Bases of Circulatory Transport: Regulation and Exchange 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, Ninth Annual Sym. on Biomath. 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, Journal of Dynamic Systems, Measurement, and Controls, pp. 301-307, 1973.

(16) Attinger, E., Anne, A., Mikami, T., and Sugawara, H. Modeling of pressure-flow relations in arteries and veins, Hemerheology, 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, Simulation, 18:197-205, 1972.

(18) Green, J. F. and Miller, N. C. A model describing the response of the circulatory system to acceleration stress, Annals of Biomedical Engineering, 1:455 - 467, 1973.

(19) Grodins, F. S., Buell, J., and Bart, A. J. Mathematical analysis and digital simulation of the respiratory control system, J. Appl. Physiol., 22(2):260-276, 1967.

(20) Energy B., and Pack, A. I. An experimentally verified model of the gas exchanging properties of the lung. Conference on Computers for Analysis and Control in Med. and Biological Research, pp. 130-134, 1971.

(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, Proc. 18th ACEMB, Philadelphia, 1965.

(24) Skalak, R. Wave propagation in blood flow, Biomechanics, Ed. by Y. C. Fung, ASME, 20-46, 1966.

(25) Gessner, U.R.S. Wave propagation in Arteries with nonlinear distensibility, Circulatory Analog Computers, 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, Circulatory Analog Compouters, 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, Circulation Research, 39:772-778, 1976.

(29) Carnahan, B., Luther, H.A. and Wilkes, J. O. Applied Numerical Methods, John Wiley & Sons, Inc., 1969.

(30) Sommerfeld, A. Mechanics of Deformable Bodies. Academic Press, New York, 1950.

(31) Perry, R. H. and Chilton C. H. Chemical Engineers Handbook. McGraw-Hill, New York, 1973.

(32) Ruch, T. C. and Patton, H. D. Physiology and Biophysics. 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, Pulsatile Blood Flow, pp. 293-305, Ed. by Attinger, E. O., McGraw-Hill, N. Y., 1964.

(37) Payne, R. M., Stone, H. L., and Engelken E. J. Atrial Function During Volume Loading, J. Appl. Physiol., 31:326-331, 1971.

(38) Bergel, E. H. Pulsatile Blood Flow. pp. 275-292, Ed. by Attinger, E. O., McGraw-Hill, N.Y., 1964.

(39) Rashevsky, N. Mathematical Biophysics. The Univ. of Chicago Press, Chicago, Illinois, 1948.

(40) Ganong, W. F. Review of Medical Physiology. LANGE Medical Publications, Los Altos, Ca., 1973.

(41) Bailey, H. R. Oxygen exchange between capillary and tissue: some equations describing countercurrent and nonlinear transport, Physical Bases of Circulatory Transport: Regulation and Exchange, Ed. by Reeve, E. B. et al., pp. 353-366, W. B. Saunders Co., Philadelphia, Penn, 1967.

(42) Gomez, D. M. Considerations of Oxygen-Hemoglobin Equilibrium in the Physiological State, Am. J. Physiol., 49: 105-124, 1965.

(43) Kelman, G. R. Digital computer procedure for the conversion of P_{CO_2} into Blood CO_2 Content, Respir. Physiol., 3: 111-115, 1967.

(44) West, J. G. Bioengineering Aspects of the Lung. 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. The ABC of Acid-Base Chemistry. The Univ. of Chicago Press, Chicago, Il., 1969.

(47) Gordon B. L., et al., ed. Clinical Cardiopulmonary Physiology. Grune & Sratton, Inc., N.Y., 1969.

(48) Knapp, C. F. Response to the cardiovascular system to vibration and combined stress, Progress Report to Air Force Office of Scientific Research, 1977.

(49) Guyton, A. C. Medical Physiology. 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. The ABC of Acid-Base Chemistry. The University of Chicago Press, Chicago, 1969.