



NASA CR-  
160215

**GENERAL ELECTRIC**

HOUSTON, TEXAS

TECHNICAL INFORMATION RELEASE

TIR 741-MED-4004

FROM G. T. Archer TO J. A. Rummel, Ph.D./DB6

DATE 2/28/74 WORK ORDER REF: DM-110T WORK STATEMENT PARA: NAS9-12932 REFERENCE:

SUBJECT User's Instructions for the Guyton Circulatory Dynamics Model Using the Univac 1110 Batch and Demand Processing (with Graphic Capabilities)

(NASA-CR-160215) USER'S INSTRUCTIONS FOR THE GUYTON CIRCULATORY DYNAMICS MODEL USING THE UNIVAC 1110 BATCH AND DEMAND PROCESSING (WITH GRAPHIC CAPABILITIES) (General Electric Co.) 96 p HC A05/MF A01 . CSCL 06P G3/52 N79-25731 Unclas 22233

This model presents a systems analysis of a human circulatory regulation based almost entirely on experimental data and cumulative present knowledge of the many facets of the circulatory system. The model itself consists of 18 different major systems that enter into circulatory control. These systems are grouped into 16 distinct sub-programs that are melded together to form the total model.

The model develops circulatory regulation and fluid regulation in a simultaneous manner. Thus, the effects of hormonal and autonomic control, electrolyte regulation, and excretory dynamics are all important and are all included in the model. The model does not treat respiration or thermal regulation.

*G. T. Archer*  
G. T. Archer

Attachment  
/db

CONCURRENCES  
Counterpart: Medical Projects Unit Manager: *R.C. Croston* Engrg. & Advanced Programs Subsection Mgr. *CWFulcher* 3/8/74

DISTRIBUTION GE/AGS: Central Product File NASA/JSC: Retha Shirkey/JM6 (1979 distribution)  
D. G. Fitzjerrell  
D. J. Grounds  
R. F. Hassell  
J. I. Leonard V. J. Marks

Page No.  
1 of 1

PROGRAM DESCRIPTION GUIDE

A. IDENTIFICATION

Program Name - Guyton  
Programmer's Names - Guyton, White, Marks, and Archer  
Programmer Contact - G. T. Archer, GE/AGS, Houston  
Date of Issue - February 28, 1974



B. GENERAL DESCRIPTION

This model presents a systems analysis of human circulatory regulation based almost entirely on experimental data and cumulative present knowledge of the many facets of the circulatory system. The model itself consists of 18 different major systems that enter into circulatory control. These systems are grouped into 16 distinct sub-programs that are melded together to form the total model.

In spite of the fact that the total model contains almost 100 independent variables and over 350 mathematical relations of various types, each major system is modeled in a relatively crude way only with emphasis placed on gross correctness instead of fine details. It has been found that the systems analysis thus developed is successful in predicting the outcome of many varied stress experiments. This is only possible because of the extreme stability and many built-in compensations of the actual circulatory system. Without this inherent stability, each system would have to have been modeled in a much more detailed fashion to produce the requisite correlation with experiment.

The model develops circulatory regulation and fluid regulation in a simultaneous manner. Thus, the effects of hormonal and autonomic control, electrolyte regulation, and excretory dynamics are all important and are all included in the model. The model does not treat respiration or thermal regulation.

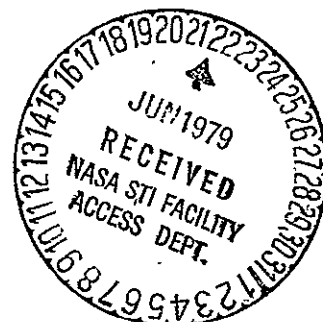
C. USAGE AND RESTRICTIONS

Machine, Operating System, and Compiler Required - Univac 1110, EXEC 8, Fortran  
Peripheral Equipment Required - Printer, Card Reader, Tape Unit, Remote Terminal (Tektronix if graph output wanted)  
Approximate Memory Required 20000<sub>10</sub>

## PROGRAM DESCRIPTION GUIDE

### A. IDENTIFICATION

Program Name - Guyton  
Programmer's Names - Guyton, White, Marks, and Archer  
Programmer Contact - G. T. Archer, GE/AGS, Houston  
Date of Issue - February 28, 1974



### B. GENERAL DESCRIPTION

This model presents a systems analysis of human circulatory regulation based almost entirely on experimental data and cumulative present knowledge of the many facets of the circulatory system. The model itself consists of 18 different major systems that enter into circulatory control. These systems are grouped into 16 distinct sub-programs that are melded together to form the total model.

In spite of the fact that the total model contains almost 100 independent variables and over 350 mathematical relations of various types, each major system is modeled in a relatively crude way only with emphasis placed on gross correctness instead of fine details. It has been found that the systems analysis thus developed is successful in predicting the outcome of many varied stress experiments. This is only possible because of the extreme stability and many built-in compensations of the actual circulatory system. Without this inherent stability, each system would have to have been modeled in a much more detailed fashion to produce the requisite correlation with experiment.

The model develops circulatory regulation and fluid regulation in a simultaneous manner. Thus, the effects of hormonal and autonomic control, electrolyte regulation, and excretory dynamics are all important and are all included in the model. The model does not treat respiration or thermal regulation.

### C. USAGE AND RESTRICTIONS

Machine, Operating System, and  
Compiler Required - Univac 1110, EXEC 8, Fortran

Peripheral Equipment Required - Printer, Card Reader, Tape Unit,  
Remote Terminal (Tektronix if graph  
output wanted)

Approximate Memory Required - 20000<sub>10</sub>

D. PARTICULAR DESCRIPTION

Equations used - See the following references:

- 1) Guyton, A.C., Coleman, T.G., and Granger, H.I., "Circulation: Overall Regulation," Annual Review of Physiology, V. 34: 13-46, 1972.
- 2) GE TIR No. 741-MED-3042

Definitions of Terms - Appendix A

Values of Variables - Appendix B

E. DESCRIPTION OF INPUT

A Univac 1110 file (GUYTON.) contains the source and relocatables of all the subroutines, the executable program, base data files (GUYTON.GUYDAT and GUYTON.TTYDAT), and Remote Batch Run stream file (GUYTON.RUNGUY). Remote Batch may require the base data and/or the Remote Batch run stream file to be modified. Since GUYTON. is not protected, the user should copy GUYTON. into another file, then make his modifications to this other file. Examples to follow should explain.

1. Batch -

The print and tape output will be generated at the 1110 (onsite).

a. Onsite Batch

The run will be in card deck form, and initiated at the 1110 (onsite). A JSC Form 588A should be submitted with the card deck.

(1) Job Stream Cards (Start Col. 1)

@RUN,/N BA629C,7007-Q509-C,DB6-G03432,TT,PPP  
(TT=mins.run time,PPP=pages output,NAME=user name)

Col. 61  
↓  
NAME SKYLAB

@ASG,T 7,8C9,TAPENØ  
(omit above card if output tape not wanted,TAPENØ=tape no.)

@ASG,AX GUYTON.  
@CØPY GUYTON.,TPF\$  
@FREE GUYTON.  
@XQT  
data cards  
@FIN

(2) Data Cards

<u>Card No.</u>	<u>Column</u>	<u>Format</u>	<u>Description</u>
1	1-4	A4	Nbbb
2	1-4	A4	Nbbb if no output tape wanted. Ybbb if output tape wanted.
3-N			Cards 3-N are initializing cards for up to 400 variables (see Appendix B for list).
	1-13	E-13.6	Variable value.
	16-20	I5	Array location (1-400).
	23-26	A4	Variable name (left justified).
N+1			Blank card.
N+2	1-4	I4	Experiment no. for printout.
	5-76	18A4	Alpha heading for printout.
N+3	1-80	20A4	Up to 20 variable names (left justified) wanted for printout. ALLb coded in cols. 1-4 will result in all 400 variables being output, and no output tape will be generated.
N+4	1-6	I6	Time to stop first time step.
	7-10	A4	Units of time (DAYS,HOUR,MINS,SECS) to which the no. in cols. 1-6 pertains. Printout will also occur at this unit of time.

## (N+5)-Last Data Card

The remaining data cards are combinations of 'variable value change' and 'time to stop time step' cards.

Variable Value	11-14	A4	Alpha name of variable to change (left justified).
	15-27	E-13.6	New value for variable.
Time to Stop	1-6	I6	Time to stop next time step.
Time Step Card	7-10	A4	Units of time (DAYS,HOUR,MINS,SECS) to which the no. in cols. 1-6 pertains. Printout will also occur at this unit of time.

As many of the above two types of cards as needed can be input at (N+5) - Last Data Card.

Last Card

Blank

b. Remote Batch

The run is initiated from a remote terminal. The run stream and the data are in Univac 1110 files GUYTON.RUNGUY and GUYTON.GUYDAT. The data file is the same format as the Onsite Batch data cards. The following example shows copying GUYTON. into another file (NEWFIL.), modifying the run stream and data files, copying the run stream into another file (RUNFIL.), then executing the run via the remote terminal.

<u>TTY Input</u>	<u>Comments</u>
@ASG,UP NEWFIL.,F2	Catalog a new file named NEWFIL.. This need not be done if NEWFIL. already catalogued and hasn't been deleted.
@ASG,UP RUNFIL.,F2	Catalog a new file to put run stream (run stream must be in separate file).
@COPY GUYTON.,TPF\$.	Copy GUYTON. into temporary work file.
@ED,L RUNGUY,RUNGUY	To edit run stream file.

The run stream is the following format:

Col. 61

```

@RUN,/N BA629C,7007-Q509-C,DB6-G03432,TT,PPP      NØDECK
(TT=mins.run time, PPP=pages output.

@ASG,T 7,8C9,SBMDXX
  Omit above if output tape not wanted.
  (XX=01-20 for tape no.)

@ASG,AX NEWFIL.
@COPY NEWFIL.,TPF$.
@FREE NEWFIL.
@XQT
@ADD GUYDAT

```

The user might want to change the run time or output pages, the output tape no., or completely delete the tape assign statement if tape output not wanted.

@ED,L GUYDAT,GUYDAT                      To edit data file.

The data file is same format as the Onsite Batch data cards. The user can modify (change, delete, or add to) this base data file from card 2 on for his particular run.

@COPY TPF\$,NEWFIL.	Copy temporary work file into NEWFIL.
@COPY,S TPF\$,RUNGUY,RUNFIL.	Copy run stream in RUNFIL.
@MSG,Ø STARTING SBMDXX PROCESSING	Omit this input if an output tape not wanted. XX=01=20 for tape no.
@FREE RUNFIL.	
@START RUNFIL.RUNGUY	Initialize Remote Batch execution.

NOTE: After above input, NEWFIL. contains the latest run stream and data files. The next time the user wants to modify these files he may substitute the first three TTY inputs with @COPY NEWFIL.,TPF\$.

2. Demand (Time-Share)

The run will be initiated from a remote terminal. The print and/or graphic output will be to the terminal that initiated the run. Since output tapes are made so results can be plotted on a plotter, the option to make a tape is not used in the Demand mode because with the Tektronix terminal the user may generate graphic output. The user inputs his data via responding to questions asked by the program. Options are available so a user can input all his data before the program executes, or intermediate results may be requested allowing the user to determine what variables to change (and to what values) and what variables to output and/or graph before continued execution. Numerous options are available for the generation of graphic output. Note the following example: (Appendix D also contains an example).

<u>TTY Input/Output</u>	<u>Comments</u>
@COPY GUYTON., TPF\$.	Copy program into work file.
@XQT	Execute program.
CONVERSATIONAL INPUT (A4)Y,N... Y (user answer)	Question from program.
WANT OUTPUT TAPE (A4)Y,N... N (user answer)	Question from program.
ADD DATA FILE (@ADD TTYDAT),... @ADD TTYDAT (user answer)	Question from program. This adds initial data for variables in program.
INPUT NØ.AND NAME FØR EXPERIMENT (I4,15A4)... 1313 REDUCE CV TEST 2/21/74 (user answer)	Question from program.  This will be output on number and graph outputs.
TYPE RETURN...	Question from program. User answers with Carriage Return.
ØUTPUT WANTED ØR SAME, STØP (6A4)... YRC VP QØ2 PRA HM I (user answer)	Question from program. 1-6 variable names may be entered here. These will be the variables printed out and/or the variables available for graphing. If this is not the 1st 'Run Step', SAME may be entered meaning same variables as preceding 'Run Step'. STØP entered will stop execution of program.
INPUT WANTED CHGED.(A4,2X,F10.4) DØNE=NØ MØRE CHGS... SYMB VALUE... DØNE (user answer)	Opportunity for user to change the value of any variables.

<u>TTY Input/Output</u> (cont'd)	<u>Comments</u>
<pre> TIME STEP (A4,1X,F6.0,F6.0)... UNIT PRINT TIME (UNIT=DAYS,HOUR,MIN, SECS, STEP)... HOUR: 1. 5. (user answer) </pre>	<p>Question from program. Program wants to know how long to run this 'Time Step', and what times to printout results (in case the user is going to exercise the option of outputting numbers). STEP entered will cause the program to printout the beginning, ending, and total time of the 'Run Step' (total of all 'Time Steps') input at this time</p>
<pre> EXEC PARAM.(A4) (MORE,RUN,PLOT)... MORE (user answer) </pre>	<p>Question from program. MORE means the user wants to enter more 'Time Steps' before executing the 'Run Step'. RUN means to execute the 'Time Steps' (the total of these 'Time Steps' constitutes 'Run Step') printing the results in number form. Each time the screen is filled with numbers, the program gives the user the opportunity to make hard copies before erasing the screen and continuing the number output. At the conclusion of the number output for the 'Run Step', the user is then given the opportunity to graph any of the output variables. PLOT entered will execute the 'Run Step', while plotting on graphs as it is being executed (no number output).</p>
<pre> INPUT WANTED CHGED.(A4,2X,F10.4) DONE= NO MORE CHGS... SYMB VALUE... CV .06 (user answer) CV .0600 SYMB VALUE... DONE (user answer) </pre>	<p>Program echos user change. Program asks for more changes</p>
<pre> TIME STEP (A4,1X,F6.0,F6.0)... UNIT PRINT TIME (UNIT=DAYS,HOUR,MIN,SECS,STEP)... HOUR 2. 20. (user answer) </pre>	
<pre> EXEC PARAM.(A4) (MORE,RUN,PLOT)... PLOT (user answer) </pre>	
<pre> GRAPH OUTPUT WANTED (Y,N,S);NO INTERVALS OF TIME (A4,F5.0)... Y 5. (user answer) </pre>	<p>Program wants to know if graph output wanted, and if it is, how many divisions wanted in time (at present this can only be 3.,4.,or 5.).</p>

(cont'd next page)



TTY Input/OutputComments

The total 'Run Step' time (which is in mins.) will be divided by the ~~hrs~~ of divisions of time. This quotient will be factored to nearest unit of time (like multiplied by 60 if quotient is < 1 for secs., or divided by 60 if  $\geq 60$  and  $\leq 1440$  for hour units, etc.) The user can correlate his 'Run Step' time with the  $N\emptyset$ .INTERVALS OF TIME he inputs so the X scale will be some meaningful increment of time (like if 'Run Step' started at 0 and went to 40 mins., a good  $N\emptyset$ .INTERVALS OF TIME would be 4.50 so that each time division would be incremented by 10 minutes).

S means to use the same graph commands as used in previous graphs made. N means do not graph.

```
VRC          Y SCALE (A4,6X,F4.0,2F6.0)
PLOT(Y,N,S) LOC  HIGH  LOW  ...
Y            1.    2.5  0.  (user answer)
```

When a Y is entered in response to 'GRAPH OUTPUT WANTED' question, the program will give the user the opportunity to provide graph commands for each of the variables available for graphing. The screen can be divided into 1,2,3,4,5, or 6 different graphs, with the variables available for graphing put into the graphs any way wanted. LOC is the graph command that conveys to program how many graphs, and which graph the variable is to be plotted on. Example: If the screen were to be divided into two graphs, all the variables wanted in the top graph would have 1. entered in LOC, and all variables wanted in the bottom graph would have 2. entered in LOC. Variables plotted on same graph will be distinguishable by different types of lines. HIGH and LOW is the maximum and minimum values wanted plotted of the variable. S entered means plot the variable with the same graph commands as used in previous graph output. N entered means this variable is not to be plotted.

<u>TTY Input/Output</u>	<u>Comments</u>
VP	Y SCALE (A4,6X,F4.0,2F6.0)
PLOT(Y,N,S) LOC HIGH LOW...	Etc. for all the variables entered
N (user answer)	in 'OUTPUT WANTED' question.

When a user enters graph commands for the last variable, the program plots the graph(s) on the screen. The program then gives the user opportunities to make hard copies. When the user is finished making hard copies (a N is entered to hard copy question), the program cycles back to 'GRAPH OUTPUT WANTED' question, giving an opportunity to re-graph this same 'Run Step' again (maybe with completely different graph commands). This sequence continues until the user enters N to 'GRAPH OUTPUT WANTED' question. At that time the program cycles back to 'OUTPUT WANTED' question so the user can build a new 'Run Step' beginning at the time where previous 'Run Step' ended.

#### F. DESCRIPTION OF OUTPUT

See Appendix C for example of Batch output.

See Appendix D for example of Demand input/output.

#### G. INTERNAL CHECKS AND EXITS

1. Curve limits are checked with a diagnostic message printed out if they are exceeded.
2. Demand mode checks input data for invalid input, allowing the user to resubmit his input if wrong.
3. Demand mode will buffer up to 200 input changes, and 100 'Time Steps' per 'Run Step'.

#### H. INDEPENDENT SUBROUTINES

See Appendix E for listing of all subroutines except Tektronix plot software.

#### I. SYSTEM SUBROUTINES

No special system subroutines.

#### J. COMPLETION OR FINAL CHECKOUT DATE

2/21/74

APPENDIX A  
DEFINITION OF TERMS

The following list includes all variables used in the model together with the normal values of these variables. Independent variables (never calculated by the program) are indicated by \*. Units used are: volume in liters, mass in grams, time in minutes, chemical units in milliequivalents, pressure in millimeters of mercury, and control factors as ratio to normal.

- AAR- afferent arteriolar resistance (31.7)
- AGK\*- constant concerned with effect of renin on angiotensin formation (0.20)
- AH- antidiuretic hormone secretion rate (3.0)
- AHC- antidiuretic hormone concentration (1.0)
- AHK\*- constant used in calculating antidiuretic hormone concentration (7.0)
- AHM- antidiuretic hormone multiplier (1.0)
- AHY- adapted effect of right atrial pressure on antidiuretic hormone secretion rate (0.0)
- AHZ- basic effect of right atrial pressure on antidiuretic hormone secretion rate (0.0)
- AH8- effect of autonomic stimulation on antidiuretic hormone secretion rate (0.0)
- ALO\*- maximum aortic arterial oxygen saturation (1.0)
- AM- aldosterone multiplier (1.0)
- AMC- aldosterone concentration (1.0)
- AMM= muscle vascular constriction caused by local tissue control, ratio to resting state (1.0)
- AMP- effect of arterial pressure on rate of aldosterone secretion (1.0)
- AMR- effect of sodium to potassium ratio on rate of aldosterone secretion (1.0)
- AMT\*- time constant of aldosterone accumulation and destruction (60)
- AM1- rate of aldosterone secretion (1.0)

- ANC- angiotensin concentration (1.0)
- ANM- angiotensin multiplier effect on vascular resistance, ratio to normal (1.0)
- ANP- effect of renal blood flow on angiotensin formation (1.0)
- ANR- effect of glomerular filtration and sodium concentration on renin formation with consequent effect on angiotensin formation (1.0)
- ANT\*- time constant of angiotensin accumulation and destruction (15.0)
- ANU- non-renal effect of angiotensin (1.0)
- ANV\*- constant used in calculating effect of renin formation on angiotensin formation (0.0003)
- ANW- partial effect of renin on angiotensin formation (0.0)
- ANY\*- constant used to calculate angiotensin effect on venous volume (-0.2)
- ANZ\*- constant used to calculate angiotensin effect on venous resistance (0.4)
- AN1 - rate of angiotensin formation (1.0)
- AOM- autonomic effect on tissue oxygen utilization (1.0)
- APD- afferent arteriolar pressure drop (38.0)
- ARF\*- intensity of sympathetic effects on renal function (1.5)
- ARM- vasoconstrictor effect of all types of autoregulation (1.0)
- AR1- vasoconstrictor effect of rapid autoregulation (1.0)
- AR2- vasoconstrictor effect of intermediate autoregulation (1.0)
- AR3- vasoconstrictor effect of long-term autoregulation (1.0)
- AU- overall activity of autonomic system (1.0)
- AUB- effect of baroreceptors on autoregulation (1.0)
- AUC- effect of chemoreceptors on autonomic stimulation (0.0)
- AUH- autonomic stimulation of heart (1.0)

- AUJ- basic overall autonomic stimulation (1.0)
- AUK\*- time constant of baroreceptor adaptation (0.0005)
- AUL\*- sensitivity of sympathetic control of vascular capacitance (0.21)
- AUM- sympathetic vasoconstrictor effect on arteries (1.0)
- AUN- effect of CNS ischemic reflex on autoregulation (0.0)
- AUO- fractional departure of overall activity of autonomic system from normal (0.0)
- AUP- autonomic stimulation of peripheral circulatory sensitivity (1.0)
- AUQ\*- sensitivity of sympathetic control of peripheral circulation (1.0)
- AUR- autonomic stimulation for heart rate (1.0)
- AUS\*- sensitivity of sympathetic control of heart rate (1.0)
- AUV\*- sensitivity of sympathetic control on heart function (0.3)
- AUX\*- sensitivity of baroreceptors (3.0)
- AUY\*- sensitivity of sympathetic control of veins (0.25)
- AUZ\*- overall sensitivity of autonomic control (1.0)
- AU4- degree of adjustment of baroreceptor response (0.0)
- AU6- adapted baroreceptor response (1.0)
- AU8- rate of adaptation of baroreceptors (0.0)
- AVE- effect of autonomic stimulation on venous resistance (1.0)
- A1B- sensitivity parameter for baroreceptor drive (1.0)
- A1K\*- time constant of rapid autoregulation
- A2K\*- time constant of intermediate autoregulation (20.0)
- A3K\*- time constant of long-term autoregulation (11520.0)
- A4K\*- time constant for muscle local vascular response to metabolic activity (1.0)
- BFM- muscle blood flow (1.0)

- BFN- blood flow in non-muscle, non-renal tissues (3.0)
- CCD- concentration gradient across cell membrane (0.0)
- CFC\*- capillary filtration coefficient (0.007)
- CHY- concentration of hyaluronic acid in tissue fluids (5.0)
- CKE- extracellular potassium concentration (5.0)
- CKI- intracellular potassium concentration (142.0)
- CNA- extracellular sodium concentration (142.0)
- CNB- difference between extracellular sodium concentration and set point used to calculate antidiuretic hormone secretion rate (3.0).
- CNR\*- reference sodium concentration used in determining effect of sodium on anti-diuretic hormone secretion rate (139.0).
- CNE- sodium concentration abnormality causing third factor effect (10.0)
- CNX\* constant used in calculation of renal excretion rate of sodium (2.5)
- CNY\*- constant used in calculation of renal excretion rate of sodium (6.0)
- CNZ\*- sensitivity of antidiuretic hormone production rate to extracellular sodium concentration (1.0)
- CN2\* constant used in calculation of venous resistance (0.0212)
- CN3- dummy variable used in calculation of the effect of capillary pressure on venous resistance (0.366)
- CN7\*- constant used in calculation of venous resistance (0.2).
- CPF\*- sensitivity of rate of transfer of fluid across pulmonary capillaries to pressure gradient (0.0003)
- CPG- concentration of protein in tissue gel (12.5)
- CPI- concentration of protein in free interstitial fluid (16.5)
- CPK\*- rate constant used in determining loss of plasma protein through systemic capillaries ( $1.6 \times 10^{-7}$ )

CPN-	concentration of protein in pulmonary fluids (30.0)
CPP-	plasma protein concentration (70.0)
CPR*-	reference plasma protein concentration governing protein production by liver (85.0)
CV*-	venous capacitance (0.0925)
DAS-	rate of volume increase of systemic arteries (0.0)
DAU-	autonomic stimulation drive (1.0)
DFP-	rate of increase in pulmonary free fluid (0.0)
DHM-	rate of cardiac deterioration caused by hypoxia (0.0)
DLA-	rate of volume increase in pulmonary veins and left atrium (0.0)
DLP-	rate of formation of plasma protein by liver (0.007)
DLZ-	undamped plasma protein concentration differential causing protein production by liver (0.007)
DOB-	rate of oxygen delivery to non-muscle cells (180.0)
DPA-	rate of increase in pulmonary volume (0.0)
DPC-	rate of loss of plasma proteins through systemic capillaries (0.05)
DPI-	rate of change of protein in free interstitial fluid (0.0)
DPL-	rate of systemic lymphatic return of protein (0.05)
DPO*-	rate of loss of plasma protein (0.007)
DRA-	rate of increase in right atrial volume (0.0)
DVS-	rate of increase in venous vascular volume (0.0)
EXC*	exercise activity, ratio to normal at rest (1.0)
EXE-	exercise effect on autonomic stimulation (0.0)
EXI*-	constant concerned with effect of muscle cell $P_{O_2}$ on autonomic stimulation during exercise (3.0)



FIS*-	fistula parameter (0.0)
GBL*-	Goldblatt hypertension parameter (0.0)
GFN-	glomerular filtration rate of undamaged kidney (0.125)
GFR-	glomerular filtration rate (0.125)
GF1 -	value of GFN on previous iteration (0.125)
GF2*-	constant used in calculation of glomerular filtration rate (0.05)
GF3-	degree of autoregulatory feedback at macular densa (1.0)
GF4*-	constant controlling the feedback loop for GF3 (5.0)
GLP-	glomerular pressure (62.0)
GPD-	rate of increase of protein in gel (0.0)
GPR-	total protein in gel (143.0)
HKM*-	constant used in calculation of portion of blood viscosity caused by red blood cells (0.53)
HM-	hematocrit (41.0)
HMD-	cardiac depressant effect of hypoxia (1.0)
HMK*-	constant used in calculation of portion of blood viscosity caused by red blood cells (90.0)
HPL-	hypertrophy effect on left ventricle (1.0)
HPR-	hypertrophy effect on right ventricle (1.0)
HR-	heart rate (72.0)
HSL*-	basic left ventricular strenght (1.0)
HSR*-	basic right ventricular strenght (1.0)
HYL*-	quantity of hyaluronic acid in tissues (57.0)
I-	integration step size (0.73)
IFP-	interstitial fluid protein (9.1)
II-	variable integration step size utilized on stable asymptote

I2*-	normal increment on time (0.003)
I3*-	maximum time increment for stable asymptote (20.0)
KCD-	rate of change of intracellular potassium concentration (0.0)
KE-	total extracellular fluid potassium (75.0)
KED-	rate of change of extracellular potassium concentration (0.0)
KI-	total intracellular potassium concentration (3550.0)
KID*	rate of potassium intake (0.0028)
KIE-	excess potassium concentration causing change in intracellular potassium level (0.0)
KIR-	total expected level of potassium in the intracellular fluid under equilibrium conditions (3550.0)
KOD-	rate of renal loss of potassium (0.0028)
LPK* -	rate constant for plasma protein production by liver (0.00047)
LVM-	effect of aortic pressure on left ventricular output (1.0)
MMO-	rate of oxygen utilization by muscle cells (60.0)
MO2-	rate of oxygen utilization by non-muscle cells (180)
NAE-	total extracellular sodium (2136.0)
NED-	rate of change of sodium in extracellular fluids (0.0)
NID*-	rate of sodium intake (0.1)
NOD -	rate of renal excretion of sodium (0.1)
NOZ-	effect of urinary output, aldosterone, and sodium level on renal excretion rate for sodium (0.1)
OMM*-	muscle oxygen utilization at rest (60.0)
OSA-	aortic oxygen saturation (1.0)
OSV-	non-muscle venous oxygen saturation (0.7)

OVA-	oxygen volume in aortic blood (203.0)
OVS-	muscle venous oxygen saturation (0.7)
O2A*-	sensitivity of the effect of autonomic stimulation on metabolism (1.5)
O2M*-	basic oxygen utilization in non-muscle body tissues (180.0)
PA-	aortic pressure (100.0)
PAM-	effect of arterial pressure in distending arteries, ratio to normal (1.0)
PAR-	renal arterial pressure (100.0)
PA1-	effective pressure drive on autonomic system (100.0)
PA2-	effective arterial pressure on left ventricle (100.0)
PC-	capillary pressure (18.4)
PCD-	net pressure gradient across capillary membrane (0.45)
PCE*-	capillary pressure exponent (3.0)
PCP-	pulmonary capillary pressure (7.0)
PDO-	difference between muscle venous oxygen $P_{O_2}$ and normal venous oxygen $P_{O_2}$ (0.0)
PFI-	rate of transfer of fluid across pulmonary capillaries (0.0)
PFL-	renal filtration pressure (16.0)
PGC-	colloid osmotic pressure of tissue gel (6.1)
PGH-	absorbency effect of gel caused by recoil of gel reticulum (-4.0)
PGL-	pressure gradient in lungs (15.2)
PGP-	colloid osmotic pressure of tissue gel caused by entrapped protein (4.13)
PGR-	colloid osmotic pressure of interstitial gel caused by Donnan equilibrium (2.0)
PGS-	pressure difference between arteries and veins (96.0)
PGV-	venous pressure gradient (3.7)

- PGX- activity factor for protein in the interstitial fluid (16.5)
- PIF- interstitial fluid pressure (-6.0)
- PK1\*- constant used in calculating muscle cell  $P_{O_2}$  from total volume of oxygen in muscle cells (2500.0)
- PK2\*- constant used in calculating muscle cell  $P_{O_2}$  from total volume of oxygen in muscle cells (800.0)
- PK3\*- constant used in calculating rate of oxygen transport to muscle cells (2.0)
- PLA- left atrial pressure (0.0)
- PLD- pressure gradient to cause lymphatic flow (0.8)
- PLF- pulmonary lymphatic flow (0.0003)
- PMC- mean circulatory pressure (6.9)
- PMO- muscle cell  $P_{O_2}$  (8.0)
- PMP- mean pulmonary pressure (4.6)
- PMS- mean systemic pressure (7.25)
- PM1- effective muscle cell  $P_{O_2}$  (8.0)
- PM3\*- minimum value allowed for PM1 (0.001)
- PM4\*- constant used in calculating rate of oxygen transport to muscle cells (-1.0)
- PM5\*- constant used in calculating rate of oxygen transport to muscle cells (122.0)
- POA- rate of change of intermediate autoregulation vasoconstrictor effect (1.0)
- POB- rate of change of rapid autoregulation vasoconstrictor effect (1.0)
- POC- rate of change of long-term autoregulation vasoconstrictor effect (1.0)
- POD- non-muscle venous  $P_{O_2}$  minus normal value (0.0)
- POE- sensitivity control for oxygen feedback control loop (1.0)
- POK\*- sensitivity of rapid system of autoregulation (0.06)

POM*-	sensitivity of oxygen feedback control loop (0.08)
PON*-	sensitivity of intermediate autoregulation (0.3)
POQ-	effective non-muscle cell $P_{O_2}$ (8.0)
POR*-	reference value of capillary $P_{O_2}$ in non-muscle tissue (40.0)
POS-	pulmonary interstitial fluid colloid osmotic pressure (12.0)
POT-	non-muscle cell $P_{O_2}$ (8.2)
POV-	non-muscle venous $P_{O_2}$ (40.0)
POY*-	sensitivity of red cell production (0.0000464)
POZ*-	sensitivity of long-term autoregulation (0.3)
PO1*-	constant used in determining oxygen deficit factor causing red cell production (8.25)
PO2-	oxygen deficit factor causing red cell production (0.25)
PPA-	pulmonary arterial pressure (15.4)
PPC-	plasma colloid osmotic pressure (28.0)
PPD-	rate of change of protein in pulmonary fluids (0.0)
PPI-	pulmonary interstitial fluid pressure (-10.0)
PPN-	rate of pulmonary capillary protein loss (0.0)
PPO-	pulmonary lymph protein flow (0.009)
PPR-	total protein in pulmonary fluids (0.38)
PPI-	variable used to empirically relate pulmonary arterial pressure and pulmonary arterial resistance (0.4)
PP2-	effective pulmonary arterial pressure (15.5)
PRA-	right atrial pressure (0.0)
PRM-	pressure caused by compression of interstitial fluid gel reticulum (-5.0)

PRP-	total plasma protein (208.0)
PR1-	effective right atrial pressure (0.0)
PTC-	interstitial fluid colloid osmotic pressure (4.1)
PTS-	solid tissue pressure (7.0)
PTT-	total tissue pressure (1.0)
PVG-	venous pressure gradient (14.6)
PVO-	muscle venous $P_{O_2}$ (40.0)
PVS-	average venous pressure (3.8)
PIO-	tissue $P_{O_2}$ effective in oxygen utilization (8.0)
P2O-	muscle cell $P_{O_2}$ effective in depressing rate of metabolism (8.0)
QAO-	blood flow in the systemic arterial system (5.0)
QLN-	basic left ventricular output (5.0)
QLO-	output of left ventricle (cardiac output) (5.0)
QOM-	total volume of oxygen in muscle cells (2400.0)
QO2-	non-muscle total cellular oxygen (2400.0)
QPO-	rate of blood flow into pulmonary veins and left atrium (5.0)
QRF*-	feedback effect of left ventricular function on right ventricular function (0.6)
QRN-	basic right ventricular output (5.0)
QRO-	actual right ventricular output (5.0)
QVO-	rate of blood flow from veins into right atrium (5.0)
RAM*-	basic vascular resistance of muscles (96.3)
RAR*-	basic resistance of non-muscular and non-renal arteries (30.52)
RBF-	renal blood flow (1.2)

RCD-	rate of change of red cell mass (0.0)
RC1-	red cell production rate (0.000011)
RC2-	red cell destruction rate (0.000011)
RDO-	resistance of diffusion of oxygen from capillaries to cells (555.0)
REK*-	fraction of normal renal function (1.0)
RFN-	renal blood flow if kidney is not damaged (1.2)
RKC*-	rate constant for red cell destruction ( $5.8 \times 10^{-6}$ )
RMO-	rate of oxygen utilization by tissues (60.0)
RPA-	pulmonary arterial resistance (1.6)
RPT-	pulmonary vascular resistance (3.0)
RPV-	pulmonary venous resistance (1.4)
RR-	renal resistance (84.0)
RSM-	vascular resistance in muscle (96.5)
RSN-	vascular resistance in non-muscle, non-renal tissues (32.5)
RTP-	total peripheral resistance (19.4)
RVG-	resistance from veins to right atrium (0.72)
RVM-	depressing effect of pulmonary arterial pressure on right ventricle (1.0)
RVS-	venous resistance (2.8)
SR*-	intensity factor for stress relaxation (0.5)
SRK*-	time constant for stress relaxation (33.0)
STA*-	overriding value of overall activity of autonomic system AU (0.0)
STH-	effect of tissue hypoxia on salt and water intake (1.0)
SVO-	stroke volume output (0.07)

T-	total time elapsed
TRR-	tubular reabsorption rate (0.124)
TVD-	rate of drinking (0.001)
TVZ-	combined effect of tissue ischemia and central nervous stimulation on thirst and drinking (0.001)
T1-	total time elapsed on previous step
U*-	damping factor for QPO (4.0)
VAE-	excess volume in systemic arteries that causes stretch of arterial walls (0.354)
VAS-	volume in systemic arteries (0.85)
VB-	blood volume (5.0)
VBD-	volume correction factor added to systemic circulation to allow for updating blood volume (0.0)
VEC-	extracellular fluid volume (15.0)
VG-	volume of interstitial fluid gel (11.5)
VGD-	rate of change of tissue gel volume (0.0)
VIB-	blood viscosity, ratio to that of water (3.0)
VIC-	cell volume (25.0)
VID-	rate of fluid transfer between interstitial fluid and cells (0.0)
VIE-	portion of blood viscosity caused by red blood cells (1.5)
VIF-	volume of free interstitial fluid (0.55)
VIM-	blood viscosity, ratio to normal (1.0)
VLA-	volume in left atrium (0.40)
VLE-	excess volume in left atrium causing stretch of left atrium and pulmonary veins (0.0)
VP-	plasma volume (3.0)



VPA-	volume in pulmonary arteries (0.38)
VPD-	rate of change of plasma volume (0.0)
VPE-	excess volume in right atrium causing stretching of the right atrium (0.07)
VPF-	pulmonary free fluid volume (0.012)
VRA	right atrial volume (0.1)
VRC-	volume of red blood cells (2.0)
VRE-	excess volume in right atrium causing stretching of the right atrium (0.0)
VTC-	rate of fluid transfer across systemic capillary membranes (0.0)
VTD-	rate of volume change in total interstitial fluid (0.0)
VTL-	rate of systemic lymph flow (0.003)
VTs-	total interstitial fluid volume (12.0)
VTW-	total body water (40.0)
VUD-	rate of urinary output (0.001)
VVE-	excess venous vascular volume before stress relaxation correction (0.33)
VVR-	volume of blood in veins at zero venous pressure (2.95)
VVS-	venous vascular volume (3.0)
VV6-	rate of change of vascular stress relaxation effect (0.0)
VV7-	increased vascular volume caused by stress relaxation (0.0)
VV8-	excess volume of blood in the systemic veins after stress relaxation correction (0.31)
VV9*-	reference venous vascular volume (3.159)
V2D*-	resistance factor which converts pressure drop to rate of change of tissue gel volume (0.02)
X*-	damping factor for QVO (10.0)

- Y\*- damping factor for DAU (1.0)
- Z'- damping factor for AH, DAU, DFP, DLP, DPC, DPL, GFN, GPD, KCD, NOD, POA, POB, PPD, TVD, VID, VTC, VTL, VUD, VV6 (1.0)
- Z1\*- damping factor for VPD (1.0)
- Z3\*- damping factor for VP (4.0)
- Z4\*- time constant used to calculate non-muscle cell total cellular oxygen (10.0)
- Z5\*- time constant used to calculate volume of oxygen in muscle cells (10.0)
- Z6\*- damping factor for OVS (5.0)
- Z7\*- damping factor for OSV (5.0)
- Z8\*- time constant of autonomic response (1.0)
- Z10\*- constant used to calculate effect of tissue hypoxia on salt and water intake (8.25)
- Z11\*- constant used to calculate effect of tissue hypoxia on salt and water intake (4.0)
- Z12\*- constant that converts exercise activity to autonomic stimulation (1.24)
- Z13\*- constant used in calculating heart hypertrophy (0.625)

GLOSSARY OF SOME OF I/O VARIABLES USED IN DEMAND MODE

- IEXECI - last EXEC parameter for a 'Run Step' ('RUNb' or 'PLOT').  
 IEXECN - subroutine TTYOUT will set this to 1 when finished 'Run Step'.  
 IIDAS(6,2) - 1st subscript is 6 possible variables that may be plotted  
                   with dashed lines.  
                   - 2nd subscript is dashed image on, and no.rasters left in image.  
 IPLTPT - pointer into PLOTBF (101,X).  
 ISTEP - pointer into JTSTEP (100,X)  
 ITIME - (flg) 1= 1st.time TTYOUT called.  
                   2= 1st.time TTYOUT called after called TTYIN (for  
                   2nd-on 'Run Steps').  
                   0= all other times.  
 IVC - pointer into VCHGS (200,X)  
 JTSTEP (100,5)- 1st. subscript is 100 possible 'Time Steps' for a 'Run Step'.  
                   2nd. subscript: 1=Ending time for 'Time Step'.  
                   2=Index of time unit.  
                   3=No. chgs. in VCHGS(200,2) associated with  
                   this 'Time Step'.  
                   4=EXEC parameter.  
                   5=No. time units for printout  
 KSTOPP - (flg) 1=TTYOUT finished a 'Run Step' (used by PLOT33 when  
                   running the PLOT option.  
 PLOTBF (101,6)- 1st subscript is buffer to save plot data.  
                   2nd subscript is the 6 variables available for plotting.  
 PLOTPT - Increment of time (in mins) for buffering into PLOTBF(101,6)..  
 PLOTTM - Next time (in mins) to put data into PLOTBF(101,6) and TLOTBF(101).  
 PPARS (6,3) - Graph commands.  
                   1st subscript is the 6 possible variables for graphing.  
                   2nd subscript: 1=which graph to plot on.  
                   2=High Y limit.  
                   3=Low Y limit.  
 RUNSTP - Total mins. in 'Run Step'.  
 TIMEC - Next time to compute to (mins.).  
 TIMEP - Next time to printout (mins.).  
 TLOTBF (101) - The mins. when data was put into PLOTBF(101,6).  
 VCHGS(200,2) - 1st. subscript is up to 200 chgs. possible for a 'Run Step'.  
                   2nd subscript is the subscript of variable to be changed,  
                   and the new value.

APPENDIX B  
INITIALIZING CARDS FOR  
INPUT VARIABLES

	Card	Variable	Location	Variable
	Col. 1	Value	In Array A(400)	Symbol
000001	000	.0	1	T
000002	000	.726606E 00	2	I
000003	000	.709991E-03	3	VBD
000004	000	.327092E 01	4	VVS
000005	000	.379952E 00	5	VPA
000006	000	.850641E 00	6	VAS
000007	000	.400835E 00	7	VLA
000008	000	.100342E 00	8	VRA
000009	000	.355641E 00	9	VAE
000010	000	.100181E 03	10	PA
000011	000	.998198E 00	11	PAM
000012	000	.990992E 00	12	LVM
000013	000	.341885E-03	13	VRE
000014	000	.683770E-01	14	PRA
000015	000	.516410E 01	15	QRN
000016	000	.737016E-01	16	VPE
000017	000	.153545E 02	17	PPA
000018	000	.399217E 00	18	
000019	000	.158269E 01	20	RPA
000020	000	.990897E 00	21	RVM
000021	000	.834793E-03	22	VLE
000022	000	.834793E-01	23	PLA
000023	000	.517104E 01	24	QLN
000024	000	.996877E 00	26	A1B
000025	000	.139474E 01	27	RPV
000026	000	.297743E 01	28	RPT
000027	000	.152710E 02	29	PGL
000028	000	.513240E 01	30	QPO
000029	000	.322544E 00	32	VVE
000030	000	.311873E 00	33	
000031	000	.378027E 01	34	PVS
000032	000	.724286E 00	36	RVG
000033	000	.513705E 01	37	QVO
000034	000	.999924E 00	38	AVE
000035	000	.212000E-01	39	CN2
000036	000	.366763E 00	40	CN3
000037	000	.278334E 01	41	RVS
000038	000	.964003E 02	42	
000039	000	.195263E 02	43	RTP
000040	000	.512704E 01	44	QAO
000041	000	.513840E 01	45	QRO
000042	000	.513356E 01	46	QLO
000043	000	-.100126E-01	47	DVS
000044	000	.599641E-02	48	OPA
000045	000	.652361E-02	49	DAS
000046	000	-.115794E-02	50	DLA
000047	000	-.134945E-02	51	DRA
000048	000	.100074E 03	52	
000049	000	.000000	53	AUC
000050	000	.998959E 00	54	AUB
000051	000	.000000	55	AUN
000052	000	.998431E 00	56	
000053	000	-.784352E-06	58	AUB
000054	000	.998431E 00	59	DAU

Card	Variable	Location	Variable
Col. 1	Value	In Array A(400)	Symbol
000055	000	.999622E 00	60 AUJ
000056	000	.999622E 00	61 AU
000057	000	-.378400E-03	62 AUO
000058	000	.999886E 00	63 AUH
000059	000	.999678E 00	66 AUM
000060	000	-.155708E-02	67 AU4
000061	000	.555132E 00	68 VIF
000062	000	.825000E 01	69 PO1
000063	000	.101296E 01	70 PTT
000064	000	.697301E 01	71 PTS
000065	000	-.596005E 01	72 PIF
000066	000	.164017E 02	73 CPI
000067	000	.410042E 01	74
000068	000	.701829E 02	75 CPP
000069	000	.280732E 02	76 PPC
000070	000	.147303E 02	77 PVG
000071	000	.185106E 02	78 PC
000072	000	.488332E 00	79 PCD
000073	000	.341832E-02	80 VTC
000074	000	.833411E 00	81 PLO
000075	000	.333364E-02	82 VTL
000076	000	-.142810E-03	83 VTD
000077	000	-.825464E-03	84 VPD
000078	000	.546352E-01	85 DPL
000079	000	.545903E-01	87 DPC
000080	000	-.448585E-04	88 DPI
000081	000	.696403E-02	90 DLP
000082	000	.494707E 01	92 CHY
000083	000	-.498773E 01	93 PRM
000084	000	.197883E 01	94 PGR
000085	000	.123660E 02	95 CPG
000086	000	.409933E 01	96 PGP
000087	000	.163973E 02	98 PGX
000088	000	.607816E 01	99 PGC
000089	000	-.397476E 01	100 PGH
000090	000	-.151085E-03	102 VGD
000091	000	.115218E 02	103 VG
000092	000	-.477273E-04	107 GPD
000093	000	.322779E 02	108
000094	000	.849375E 02	109 RR
000095	000	.117946E 01	110
000096	000	.380705E 02	111 APD
000097	000	.621100E 02	112 GLP
000098	000	.160369E 02	113 PFL
000099	000	.125020E 00	114 GFR
000100	000	.124014E 00	115 TRR
000101	000	.100609E-02	116 VUD
000102	000	.100000E 01	117 REK
000103	000	.102388E 00	118 NOD
000104	000	-.163419E-03	119 NED
000105	000	.213803E 04	120 NAE
000106	000	.150460E 02	121 VEC
000107	000	.499240E 01	122 CKE
000108	000	.278510E-02	123 KOD
000109	000	.354894E 04	125 KIR
000110	000	.149017E 00	126 KIE
000111	000	.193723E-02	127 KCD

	Card	Variable	Location	Variable
	Col. 1	Value	In Array A(400)	Symbol
000112	000	-.192232E-02	128	KED
000113	000	.142073E 03	129	CKI
000114	000	.142100E 03	130	CNA
000115	000	-.261745E-01	131	CCD
000116	000	-.261745E-03	132	VID
000117	000	.751092E 02	133	KE
000118	000	.354879E 04	134	KI
000119	000	.249777E 02	135	VIC
000120	000	.336342E 01	136	II
000121	000	.100000E 01	138	Z
000122	000	.115908E-02	141	TVZ
000123	000	.100000E 02	144	X
000124	000	.300000E-02	145	I2
000125	000	.683770E-01	146	
000126	000	.120771E 02	147	VTS
000127	000	.295841E 01	148	VP
000128	000	.207716E 03	149	PRP
000129	000	.910511E 01	150	IFP
000130	000	.142480E 03	151	GPR
000131	000	.980345E 00	154	AMR
000132	000	.997442E 00	155	AMP
000133	000	.981239E 00	156	AM1
000134	000	.981906E 00	157	AMC
000135	000	.984764E 00	161	AM
000136	000	.994573E 01	162	CNE
000137	000	.200000E 00	163	AGK
000138	000	.998409E 00	164	ANP
000139	000	.998409E 00	165	AN1
000140	000	.997770E 00	166	ANC
000141	000	.100352E 01	170	ANM
000142	000	.500269E 01	171	VB
000143	000	.408497E 02	173	HM
000144	000	.155836E 01	175	VIE
000145	000	.305836E 01	176	VIB
000146	000	.101935E 01	177	VIM
000147	000	.118528E-04	178	RC2
000148	000	.255528E 00	179	PO2
000149	000	.580000E-05	180	RKC
000150	000	.118565E-04	181	RC1
000151	000	.370494E-08	182	RCD
000152	000	.204359E 01	183	VRC
000153	000	.326051E 02	184	RSN
000154	000	.202801E 03	185	OVA
000155	000	.295660E 01	186	BFN
000156	000	.179952E 03	187	DOB
000157	000	.999943E 00	188	ADM
000158	000	.799444E 01	189	P10
000159	000	.695447E 00	190	OSV
000160	000	.799444E 01	191	POT
000161	000	-.262149E 00	192	
000162	000	.984271E 00	193	POB
000163	000	.984296E 00	194	AR1
000164	000	.923107E 00	195	AR2
000165	000	.974047E 00	196	
000166	000	.975063E 00	197	AR3
000167	000	.846225E 00	198	ARM
000168	000	.305427E 01	199	

	Card	Variable	Location	Variable
	Col. 1	Value	In Array A(400)	Symbol
000169	000	.175020E 00	200	GFN
000170	000	.305320E 01	203	AH
000171	000	.102326E 01	204	AHC
000172	000	.101340E 01	208	AHM
000173	000	.600000E 01	209	CNY
000174	000	.250000E 01	210	CNX
000175	000	.100638E-03	214	VV6
000176	000	.106801E-01	215	VV7
000177	000	.115908E-02	216	TVD
000178	000	.400294E 02	217	VTW
000179	000	.100000E 01	218	HSR
000180	000	.100000E 01	219	HSL
000181	000	.100000E 00	220	NID
000182	000	.500000E 00	221	SR
000183	000	.294908E 01	222	VVR
000184	000	.305200E 02	223	RAR
000185	000	.825000E-01	224	CV
000186	000	.200000E 00	225	CN7
000187	000	.300000E 01	226	AUX
000188	000	.500000E-03	227	AUK
000189	000	.100000E 01	228	AUZ
000190	000	.100000E 01	229	Y
000191	000	.700000E-02	230	CFC
000192	000	.160000E-06	231	CPK
000193	000	.300000E 01	232	PCE
000194	000	.850000E 02	233	CPR
000195	000	.470000E-03	234	LPK
000196	000	.700000E-02	235	DPO
000197	000	.570000E 02	236	HYL
000198	000	.280000E-02	237	KID
000199	000	.600000E 02	238	AMT
000200	000	.150000E 02	239	ANT
000201	000	.600000E-01	240	POK
000202	000	.300000E 00	241	PON
000203	000	.100000E 01	242	AIK
000204	000	.200000E 02	243	A2K
000205	000	.115200E 05	244	A3K
000206	000	.139000E 03	245	CNR
000207	000	.100000E 01	246	CNZ
000208	000	.700000E 01	247	AHK
000209	000	.330000E 02	248	SRK
000210	000	.301000E 00	249	V9
000211	000	.200000E-01	250	V2D
000212	000	.100000E 01	251	Z1
000213	000	.400000E 01	253	Z3
000214	000	.100000E 02	254	Z4
000215	000	.100000E 02	255	Z5
000216	000	.500000E 01	256	Z6
000217	000	.500000E 01	257	Z7
000218	000	.100000E 01	258	Z8
000219	000	.900000E 02	259	HMK
000220	000	.533330E 00	260	HKM
000221	000	.397379E 02	261	POV
000222	000	.300000E 00	262	POZ
000223	000	.510939E 03	263	
000224	000	.240073E 04	264	Q02
000225	000	.117946E 01	265	RBF

ALL INFORMATION CONTAINED  
 HEREIN IS UNCLASSIFIED  
 DATE 08/19/01 BY 60322/UC/STP



Card	Variable	Location	Variable
Col. 1	Value	In Array A(400)	Symbol
000226	000	.179990E 03	266 MO2
000227	000	.921355E 00	267 POA
000228	000	.464000E-04	268 POY
000229	000	.100348E 01	269 ANU
000230	000	.400000E 02	270 POR
000231	000	.500000E-01	271 GF2
000232	000	.100000E 01	272 HMD
000233	000	.498610E-02	273 DHM
000234	000	.799447E 01	274
000235	000	.100000E 02	275 I3
000236	000	.400000E 01	276 U
000237	000	.100000E-01	277 VP1
000238	000	.00	278 T1
000239	000	.100033E-01	279
000240	000	.500000E 01	280 GF4
000241	000	.999622E 00	281 AUP
000242	000	.300000E 00	282 AUV
000243	000	.250000E 00	284 AUY
000244	000	.300000E 01	285 GUT
000245	000	.000000	286 DSP
000246	000	.136754E-01	287 AHZ
000247	000	.126076E-01	288 AHY
000248	000	.992914E 00	289 OSA
000249	000	.858498E 01	290
000250	000	.282523E 02	291 CPN
000251	000	.113009E 02	292 POS
000252	000	.724506E-03	293 PLF
000253	000	.204689E-01	294 PPO
000254	000	.943440E-02	295 PPN
000255	000	.110345E-01	296 PPD
000256	000	.369553E-03	297 PFI
000257	000	.109406E-02	298 DFP
000258	000	.104912E-01	299 VPF
000259	000	.363250E 00	300 PPR
000260	000	.684603E 01	301 PMC
000261	000	.723762E 01	302 PMS
000262	000	.458685E 01	303 PMP
000263	000	.721216E 02	304 HR
000264	000	.300000E-03	305 CPF
000265	000	.695544E 01	306 PCP
000266	000	.100000E 01	307 DA1
000267	000	.696403E-02	308 DLZ
000268	000	.102388E 00	312 NOZ
000269	000	.100771E 01	315 HPR
000270	000	.100188E 01	316 HPL
000271	000	.102224E 01	317 STH
000272	000	.100000E 01	318 ALO
000273	000	.100000E 01	319 EXC
000274	000	.180000E 03	320 O2M
000275	000	.100191E 03	321
000276	000	.153561E 02	322
000277	000	.711792E-01	323 SVO
000278	000	.210000E 00	324 AUL
000279	000	.315900E 01	325 VV9
000280	000	.150000E 00	326 O2A
000281	000	.370620E-01	328 EXE
000282	000	.150000E 01	329 ARF

ORIGINAL PAGE IS  
OF POOR QUALITY

Card	Variable	Location	Variable
Col. 1	Value	In Array A(400)	Symbol
000283	000	.600000E 00	330 QRF
000284	000	.971946E 02	331 RSM
000285	000	.991828E 00	332 BFM
000286	000	.963000E 02	333 RAM
000287	000	.697390E 00	334 OVS
000288	000	.398489E 02	335 PVO
000289	000	.599833E 02	336 RMO
000290	000	.239984E 04	337 QOM
000291	000	.798737E 01	338 PMO
000292	000	.798737E 01	339 P20
000293	000	.599966E 02	340 MMO
000294	000	-.151147E 00	341 PDO
000295	000	.987908E 00	342
000296	000	.987945E 00	343 AMM
000297	000	.100000E 01	344 .A4K
000298	000	.800000E-01	345 POM
000299	000	.600000E 02	346 QMM
000300	000	.798737E 01	347 PM1
000301	000	.100000E-02	348 PM3
000302	000	-.100000E 01	349 PM4
000303	000	.300000E 01	350 EX1
000304	000	.122000E 03	353 PM5
000305	000	.250000E 04	354 PK1
000306	000	.825000E 01	356 Z10
000307	000	.400000E 01	357 Z11
000308	000	.124000E 01	358 Z12
000309	000	.625000E 00	359 Z13
000310	000	.800000E 03	363 PK2
000311	000	.200000E 01	364 PK3
000312	000	.000000	365 FIS
000313	000	.000000	366 STA
000314	000	.100181E 03	367 PAR
000315	000	.000000	368 GBL
000316	000	-.200000E 00	369 ANY
000317	000	.400000E 00	370 ANZ
000318	000	.300000E-03	372 ANV
000319	000	.000000	373 ANW
000320	000	.100000E 01	375 AUQ
000321	000	.999622E 00	376 AUR
000322	000	.100000E 01	377 AUS
000323	000	.400000E 02	379 H1
000324	000	.900000E 01	380 A2
000325	000	.352000E-02	381 A3
000326	000		

END  
 END IGNORED - IN CONTROL MODE

COPY TPFS, GUYTON  
 FURPUR 0025-12/05-13:32  
 27 BLOCKS COPIED

APPENDIX C  
EXAMPLE  
OF  
BATCH OUTPUT

DAYS	VRC	VP	Q02	PRA	HM	PVS	POT	VTW	VEC	VIC
0	2.0436	2.9584	2400.7300	.0684	40.8497	3.7803	7.9944	40.0294	15.0460	24.9777
1	2.0436	2.9575	2400.4644	.0612	40.8604	3.7767	7.9935	40.0424	15.0458	24.9922
2	2.0437	2.9574	2400.4953	.0610	40.8621	3.7765	7.9936	40.0425	15.0463	24.9923
3	2.0437	2.9575	2400.5126	.0612	40.8634	3.7762	7.9937	40.0423	15.0483	24.9934
4	2.0438	2.9576	2400.5329	.0575	40.8673	3.7747	7.9938	40.0389	15.0525	24.9942
5	2.0438	2.9572	2400.5452	.0605	40.8662	3.7759	7.9938	40.0427	15.0469	24.9929
6	2.0439	2.9575	2400.5611	.0610	40.8661	3.7751	7.9939	40.0430	15.0497	24.9937
7	2.0439	2.9570	2400.5419	.0603	40.8685	3.7757	7.9938	40.0426	15.0458	24.9923
8	2.0439	2.9571	2400.5340	.0593	40.8703	3.7750	7.9938	40.0409	15.0484	24.9930
9	2.0440	2.9574	2400.6804	.0569	40.8714	3.7744	7.9943	40.0386	15.0524	24.9942
10	2.0440	2.9577	2400.5498	.0554	40.8736	3.7732	7.9938	40.0429	15.0479	24.9955
11	2.0441	2.9572	2400.5318	.0616	40.8709	3.7761	7.9938	40.0423	15.0486	24.9940
12	2.0441	2.9568	2400.5764	.0592	40.8719	3.7747	7.9939	40.0438	15.0464	24.9931
13	2.0441	2.9567	2400.5334	.0591	40.8762	3.7748	7.9938	40.0394	15.0477	24.9922
14	2.0442	2.9569	2400.6169	.0579	40.8749	3.7736	7.9941	40.0407	15.0491	24.9929
15	2.0442	2.9568	2400.5577	.0598	40.8748	3.7755	7.9939	40.0448	15.0483	24.9960
16	2.0442	2.9563	2400.5063	.0580	40.8781	3.7741	7.9937	40.0416	15.0435	24.9897
17	2.0443	2.9566	2400.6139	.0592	40.8774	3.7741	7.9940	40.0388	15.0469	24.9902
18	2.0443	2.9567	2400.5073	.0595	40.8760	3.7745	7.9937	40.0440	15.0484	24.9949
19	2.0443	2.9568	2400.5922	.0582	40.8778	3.7743	7.9940	40.0422	15.0487	24.9945
20	2.0443	2.9563	2400.5973	.0601	40.8771	3.7749	7.9940	40.0477	15.0466	24.9949
21	2.0444	2.9567	2400.6230	.0581	40.8784	3.7735	7.9941	40.0411	15.0482	24.9922
22	2.0444	2.9561	2400.5155	.0600	40.8793	3.7747	7.9937	40.0472	15.0463	24.9946
23	2.0444	2.9570	2400.5920	.0549	40.8824	3.7727	7.9940	40.0394	15.0509	24.9948
24	2.0444	2.9571	2400.7300	.0607	40.8758	3.7754	7.9944	40.0441	15.0490	24.9954
25	2.0445	2.9565	2400.6492	.0583	40.8796	3.7743	7.9942	40.0431	15.0461	24.9935
26	2.0445	2.9562	2400.5698	.0583	40.8846	3.7745	7.9939	40.0415	15.0475	24.9941
27	2.0445	2.9561	2400.6052	.0601	40.8807	3.7749	7.9940	40.0473	15.0462	24.9949
28	2.0445	2.9567	2400.5714	.0604	40.8798	3.7752	7.9939	40.0438	15.0486	24.9956

AT 28 DAYS CV CHANGED FROM .082 TO .060

DAYS	VRC	VP	Q02	PRA	HM	PVS	POT	VTW	VEC	VIC
29	2.0446	2.7650	2398.4294	-.0380	42.5172	3.6624	7.9868	39.7815	14.7918	24.9921
30	2.0446	2.7710	2401.2461	-.0106	42.4508	3.6796	7.9961	39.8287	14.8376	24.9858
31	2.0446	2.7729	2401.3871	-.0084	42.4320	3.6828	7.9966	39.8316	14.8409	24.9858
32	2.0444	2.7738	2401.3397	-.0099	42.4267	3.6834	7.9965	39.8307	14.8419	24.9854
33	2.0442	2.7738	2401.2824	-.0077	42.4230	3.6847	7.9963	39.8329	14.8418	24.9858
34	2.0441	2.7744	2401.2430	-.0203	42.4307	3.6766	7.9961	39.8290	14.8461	24.9860
35	2.0440	2.7746	2401.2108	-.0152	42.4253	3.6800	7.9960	39.8292	14.8478	24.9863
36	2.0439	2.7736	2401.1769	-.0079	42.4185	3.6842	7.9959	39.8343	14.8445	24.9857
37	2.0439	2.7740	2401.1496	-.0099	42.4187	3.6833	7.9958	39.8335	14.8447	24.9858
38	2.0438	2.7740	2401.1284	-.0073	42.4156	3.6846	7.9958	39.8357	14.8444	24.9857
39	2.0437	2.7737	2401.1156	-.0081	42.4151	3.6837	7.9957	39.8357	14.8458	24.9858
40	2.0436	2.7741	2401.1017	-.0097	42.4153	3.6831	7.9957	39.8349	14.8460	24.9857
41	2.0436	2.7741	2401.0834	-.0070	42.4125	3.6844	7.9956	39.8368	14.8455	24.9857
42	2.0435	2.7737	2401.0838	-.0084	42.4124	3.6832	7.9956	39.8369	14.8469	24.9859
43	2.0435	2.7746	2401.0823	-.0208	42.4209	3.6755	7.9956	39.8332	14.8497	24.9863
44	2.0434	2.7748	2401.0663	-.0151	42.4160	3.6790	7.9956	39.8332	14.8512	24.9866
45	2.0433	2.7741	2401.0573	-.0071	42.4089	3.6836	7.9955	39.8380	14.8465	24.9857
46	2.0433	2.7748	2401.0620	-.0148	42.4140	3.6785	7.9955	39.8328	14.8521	24.9864

47	2.0432	2.7747	2401.0565	-.0189	42.4162	3.6757	7.9955	39.8343	14.8512	24.9865
48	2.0432	2.7741	2401.0461	-.0083	42.4077	3.6823	7.9955	39.8382	14.8486	24.9862
49	2.0431	2.7750	2401.0450	-.0156	42.4125	3.6774	7.9955	39.8352	14.8522	24.9868
50	2.0431	2.7743	2401.0367	-.0059	42.4045	3.6834	7.9955	39.8393	14.8483	24.9861
51	2.0430	2.7744	2401.0277	-.0060	42.4041	3.6834	7.9954	39.8392	14.8488	24.9861
52	2.0430	2.7751	2401.0406	-.0125	42.4083	3.6791	7.9955	39.8353	14.8537	24.9868
53	2.0429	2.7743	2401.0222	-.0051	42.4022	3.6838	7.9954	39.8405	14.8489	24.9861
54	2.0429	2.7751	2401.0322	-.0130	42.4075	3.6788	7.9954	39.8351	14.8537	24.9866
55	2.0429	2.7740	2401.0280	-.0062	42.4016	3.6829	7.9954	39.8405	14.8502	24.9862
56	2.0428	2.7740	2401.0351	-.0068	42.4012	3.6824	7.9954	39.8407	14.8504	24.9862

AT 56 DAYS CV CHANGED FROM .060 TO .082

DAYS	VRC	VP	QOZ	PRA	HM	PVS	POT	VTW	VEC	VIC
57	2.0461	2.9582	2404.2603	.0777	40.8837	3.7807	8.0062	40.1096	15.1161	24.9871
58	2.0458	2.9542	2401.1099	.0634	40.9140	3.7716	7.9957	40.0511	15.0569	24.9937
59	2.0457	2.9546	2400.9280	.0626	40.9101	3.7724	7.9951	40.0506	15.0545	24.9955
60	2.0457	2.9544	2400.8087	.0606	40.9125	3.7714	7.9947	40.0477	15.0538	24.9944
61	2.0457	2.9541	2400.7109	.0629	40.9113	3.7724	7.9944	40.0528	15.0513	24.9948
62	2.0456	2.9544	2400.7105	.0611	40.9109	3.7720	7.9944	40.0491	15.0519	24.9942
63	2.0457	2.9549	2400.6525	.0575	40.9128	3.7701	7.9942	40.0492	15.0566	24.9975
64	2.0457	2.9544	2400.6599	.0596	40.9121	3.7714	7.9942	40.0485	15.0524	24.9952
65	2.0457	2.9546	2400.6045	.0612	40.9103	3.7723	7.9940	40.0482	15.0524	24.9952
66	2.0457	2.9550	2400.5542	.0558	40.9154	3.7706	7.9938	40.0462	15.0516	24.9955
67	2.0457	2.9540	2400.5358	.0620	40.9119	3.7731	7.9938	40.0522	15.0500	24.9960
68	2.0457	2.9548	2400.7829	.0660	40.9045	3.7747	7.9946	40.0524	15.0513	24.9967
69	2.0457	2.9547	2400.5780	.0640	40.9095	3.7740	7.9939	40.0463	15.0513	24.9950
70	2.0457	2.9551	2400.5578	.0567	40.9149	3.7708	7.9939	40.0446	15.0503	24.9953
71	2.0457	2.9546	2400.5894	.0610	40.9110	3.7726	7.9940	40.0461	15.0508	24.9948
72	2.0458	2.9546	2400.6541	.0615	40.9123	3.7729	7.9942	40.0437	15.0504	24.9931
73	2.0458	2.9545	2400.6017	.0601	40.9113	3.7725	7.9940	40.0473	15.0491	24.9948
74	2.0458	2.9548	2400.5500	.0523	40.9181	3.7686	7.9938	40.0438	15.0507	24.9943
75	2.0458	2.9546	2400.5148	.0597	40.9133	3.7724	7.9937	40.0454	15.0509	24.9954
76	2.0458	2.9550	2400.6627	.0538	40.9161	3.7701	7.9942	40.0470	15.0505	24.9976
77	2.0458	2.9552	2400.5535	.0673	40.9080	3.7759	7.9938	40.0452	15.0497	24.9966
78	2.0458	2.9550	2400.7300	.0578	40.9128	3.7719	7.9944	40.0430	15.0529	24.9964
79	2.0459	2.9544	2400.6122	.0592	40.9126	3.7722	7.9940	40.0465	15.0472	24.9949
80	2.0459	2.9544	2400.5106	.0589	40.9145	3.7722	7.9937	40.0450	15.0493	24.9956
81	2.0459	2.9550	2400.5413	.0547	40.9175	3.7707	7.9938	40.0433	15.0483	24.9959
82	2.0459	2.9542	2400.5168	.0587	40.9149	3.7719	7.9937	40.0449	15.0470	24.9946
83	2.0459	2.9543	2400.6868	.0614	40.9115	3.7737	7.9943	40.0490	15.0472	24.9958
84	2.0459	2.9543	2400.5334	.0596	40.9162	3.7726	7.9938	40.0411	15.0477	24.9932

OFIN

APPENDIX D  
EXAMPLE OF CONVERSATIONAL DEMAND  
INPUT/OUTPUT

@XQT

GUYTON MODEL FROM WHITE  
REFER TO GE-ASG USER GUIDE TIR 741-MED-4004

CONVERSATIONAL INPUT (A4) Y,N...  
>Y

WANT OUTPUT TAPE (A4) Y,N...  
>N

ADD DATA FILE(@ADD TTYDAT)...  
>@ADD TTYDAT

331DATA RECORDS INPUT.  
INPUT NO. AND NAME FOR EXPERIMENT (I4,15A4)...  
>1313 TEST FOR TIO....

TYPE SHIFT-OUT (SO) AND RETURN-->

ORIGINAL PAGE 15  
OF FOUR QUALITY

```

OUTPUT WANTED OR SAME, STOP (6A4)...
>URC UP Q02 PRA HM I
INPUT WANTED CHGED.(A4,2X,F10.4) DONE= NO MORE CHGES...
SYMB VALUE...
>DONE
TIME STEP (A4,1X,F6.0,F6.0)...
UNIT PRINT TIME (UNIT=DAYS,HOUR,MINS,SECS,STEP)...
-HOUR 1. 4.
EXEC PARAM.(A4) (MORE,RUN,PLOT)...
>MORE
INPUT WANTED CHGED.(A4,2X,F10.4) DONE= NO MORE CHGES...
SYMB VALUE...
>CU .06
CU .0600
SYMB VALUE...
>DONE
TIME STEP (A4,1X,F6.0,F6.0)...
UNIT PRINT TIME (UNIT=DAYS,HOUR,MINS,SECS,STEP)...
-HOUR 2. 10.
EXEC PARAM.(A4) (MORE,RUN,PLOT)...
>MORE
INPUT WANTED CHGED.(A4,2X,F10.4) DONE= NO MORE CHGES...
SYMB VALUE...
>CU .08
CU .0800
SYMB VALUE...
>DONE
TIME STEP (A4,1X,F6.0,F6.0)...
UNIT PRINT TIME (UNIT=DAYS,HOUR,MINS,SECS,STEP)...
-HOUR 1. 6.
EXEC PARAM.(A4) (MORE,RUN,PLOT)...
>RUN

```



1313 TEST FOR TIO....

DAY	HR	MN	SE	URC	UP	Q02	PRA	HM	I
0	0	0	0	2.0436	2.9584	2400.7300	.0684	40.8497	.727
0	1	3	48	2.0436	2.9596	2400.7849	.0697	40.8460	10.003
0	2	0	47	2.0436	2.9602	2401.1457	.0588	40.8457	2.666
0	3	1	55	2.0436	2.9604	2401.2867	.0611	40.8460	2.213
0	4	3	23	2.0436	2.9598	2401.3124	.0673	40.8396	3.928
AT	0	DAY, 4	HR, 3 MIN, 23 SEC			CU FROM	.0825	TO	.0800
0	6	2	2	2.0435	2.9075	2434.1181	.3031	41.2693	3.132
0	8	2	45	2.0434	2.8767	2436.0443	.2456	41.5253	2.824
0	10	1	9	2.0433	2.8405	2426.1819	.1585	41.8306	2.846
0	12	2	11	2.0432	2.8078	2414.0373	.0758	42.1147	2.847
0	14	1	38	2.0431	2.7846	2404.9205	.0160	42.3198	2.829
AT	0	DAY, 14	HR, 1 MIN, 38 SEC			CU FROM	.0600	TO	.0800
0	15	3	55	2.0432	2.7987	2382.2391	-.2512	42.1992	10.003
0	16	2	23	2.0434	2.7936	2364.8690	-.2545	42.2404	3.479
0	17	0	41	2.0438	2.7773	2348.8313	-.2752	42.3863	2.710
0	18	0	14	2.0444	2.7614	2335.4866	-.3027	42.5329	2.381
0	19	2	16	2.0450	2.7533	2327.2643	-.3150	42.6129	2.355
0	20	2	25	2.0457	2.7557	2325.5790	-.3209	42.6856	2.494

HARD COPY WANTED(Y,N)...

GRAPH OUTPUT WANTED(Y,N,S), TIME INTERVALS(A4,F2.0)...

```

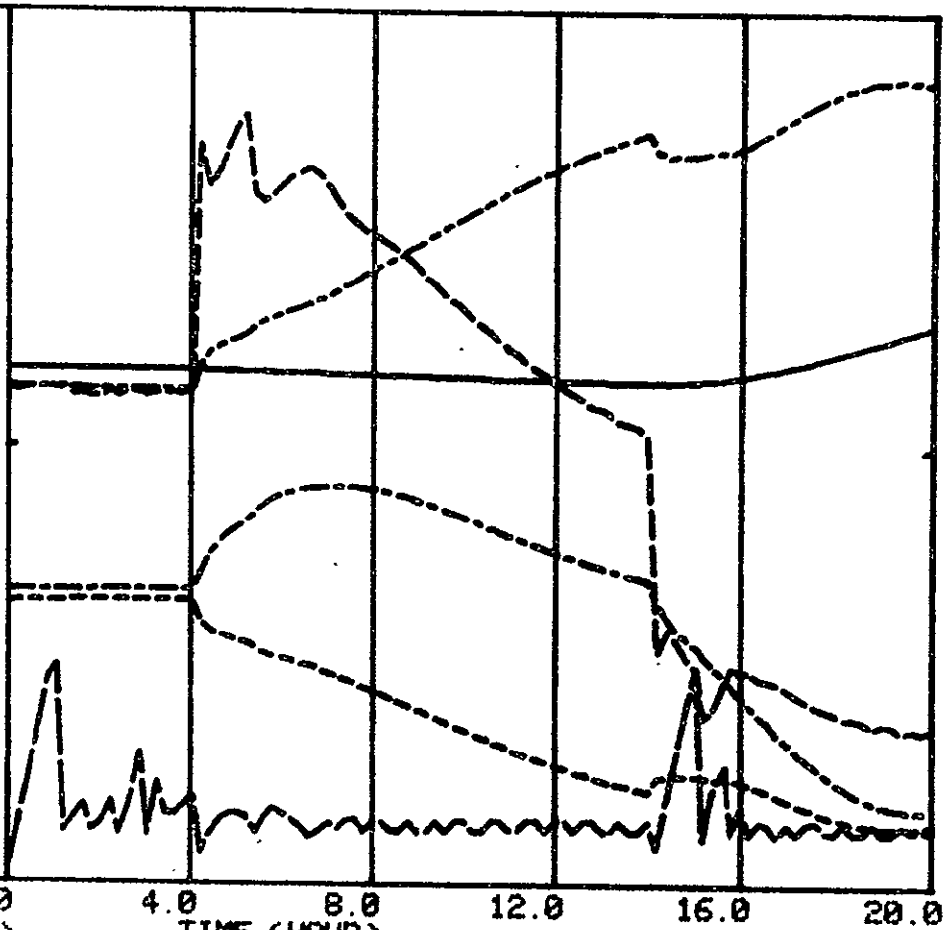
>Y 5.
  URC          Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          1.  2.06  2.02
  UP          Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          1.  3.5   2.7
  Q02        Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          1. 2600. 2300.
  FRA        Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          1.  .5   - .5
  HM         Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          1.  43.  38.
  I         Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          1.  40.  0.

```

1313 TEST FOR TIO....

-----	URC	2.06
-----	UP	3.50
-----	Q02	2600.00
-----	PRA	.50
-----	HM	43.00
-----	I	40.00

-----	I	.00
-----	HM	38.00
-----	PRA	-.50
-----	Q02	2300.00
-----	UP	2.70
-----	URC	2.02



HARD COPY WANTED(Y,N)...

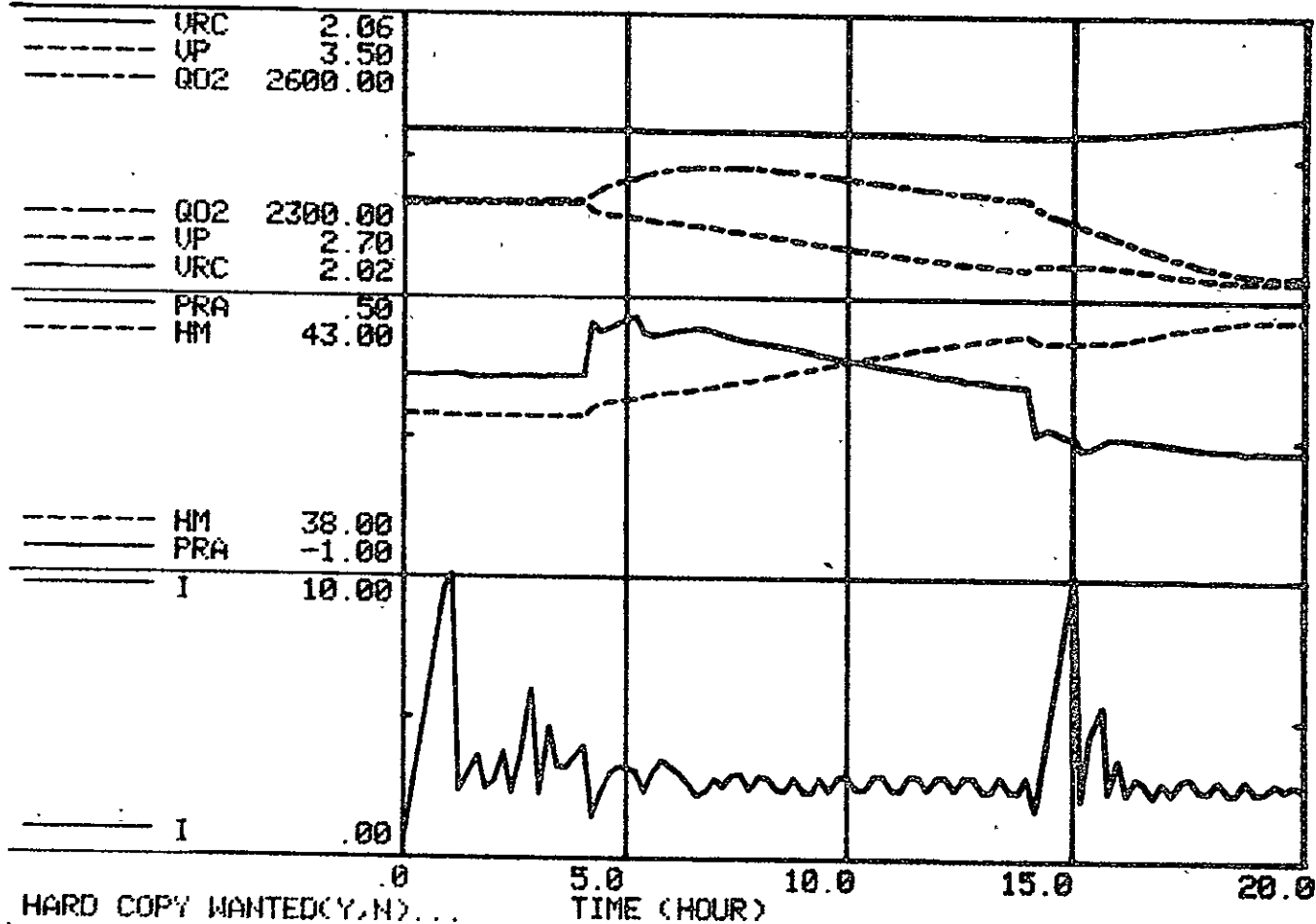
TIME (HOUR)

```

GRAPH OUTPUT WANTED(Y,N,S), TIME INTERVALS(A4,F2.0)...
>Y 4.
  URC          Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>S
  UP          Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>S
  Q02        Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>S
  PRA        Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          2.  .5  -1.
  HM        Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          2.  43.  38.
  I        Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          3.  10.  0.

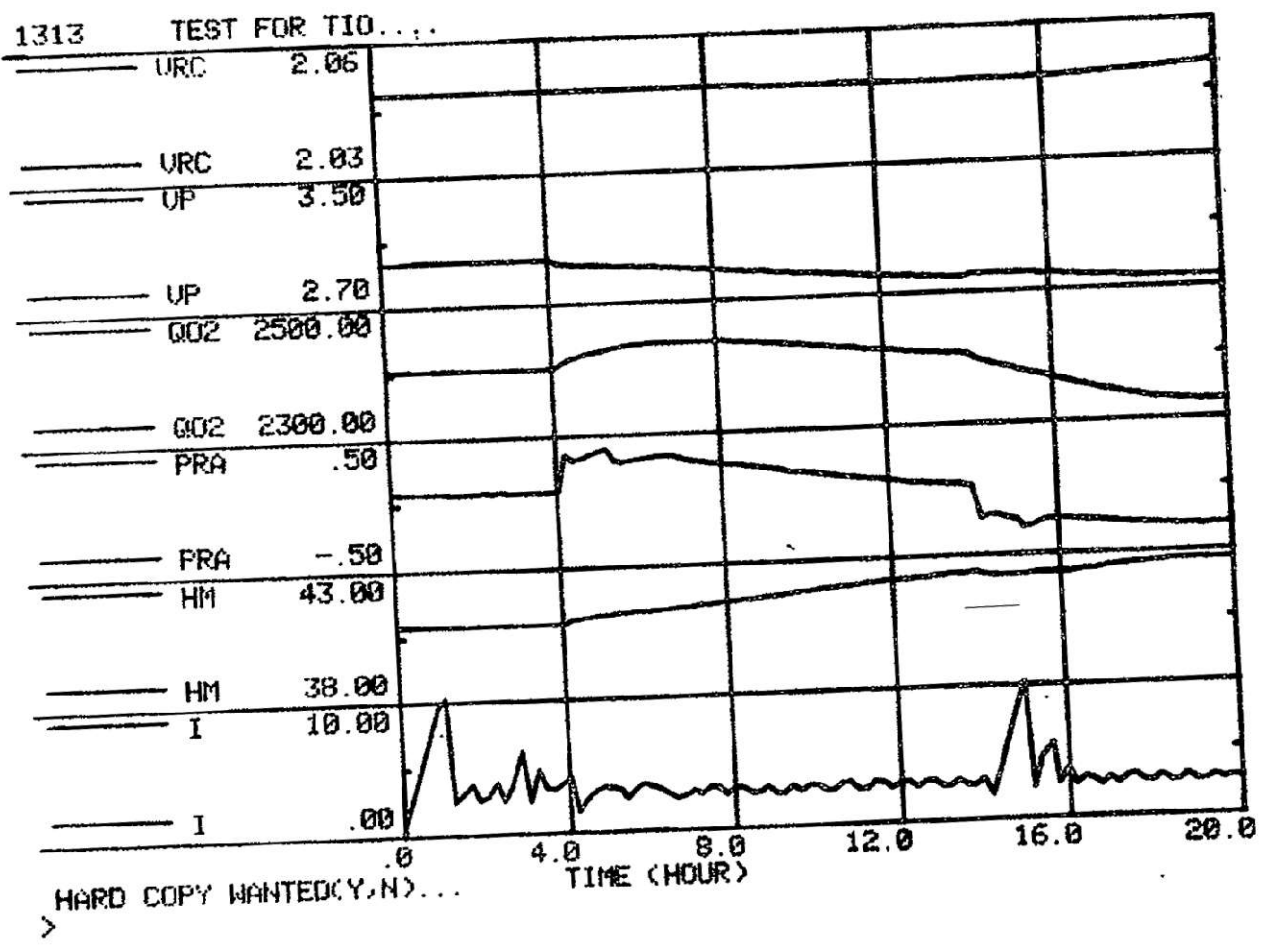
```

1313 TEST FOR TIO...



GRAPH OUTPUT WANTED(Y,N,S), TIME INTERVALS(A4,F2.0)...

>Y	5.								
URC				Y SCALE	(A4,8X,F4.0,2F6.0)				
PLOT(Y,N,S)		LOC	HIGH	LOW	...				
>Y		1.	2.055	2.025					
UP				Y SCALE	(A4,8X,F4.0,2F6.0)				
PLOT(Y,N,S)		LOC	HIGH	LOW	...				
>Y		2.	3.5	2.7					
002				Y SCALE	(A4,8X,F4.0,2F6.0)				
PLOT(Y,N,S)		LOC	HIGH	LOW	...				
>Y		3.	2500.	2300.					
PRA				Y SCALE	(A4,8X,F4.0,2F6.0)				
PLOT(Y,N,S)		LOC	HIGH	LOW	...				
>Y		4.	.5	-.5					
HM				Y SCALE	(A4,8X,F4.0,2F6.0)				
PLOT(Y,N,S)		LOC	HIGH	LOW	...				
>Y		5.	43.	38.					
I				Y SCALE	(A4,8X,F4.0,2F6.0)				
PLOT(Y,N,S)		LOC	HIGH	LOW	...				
>Y		6.	10.	0.					



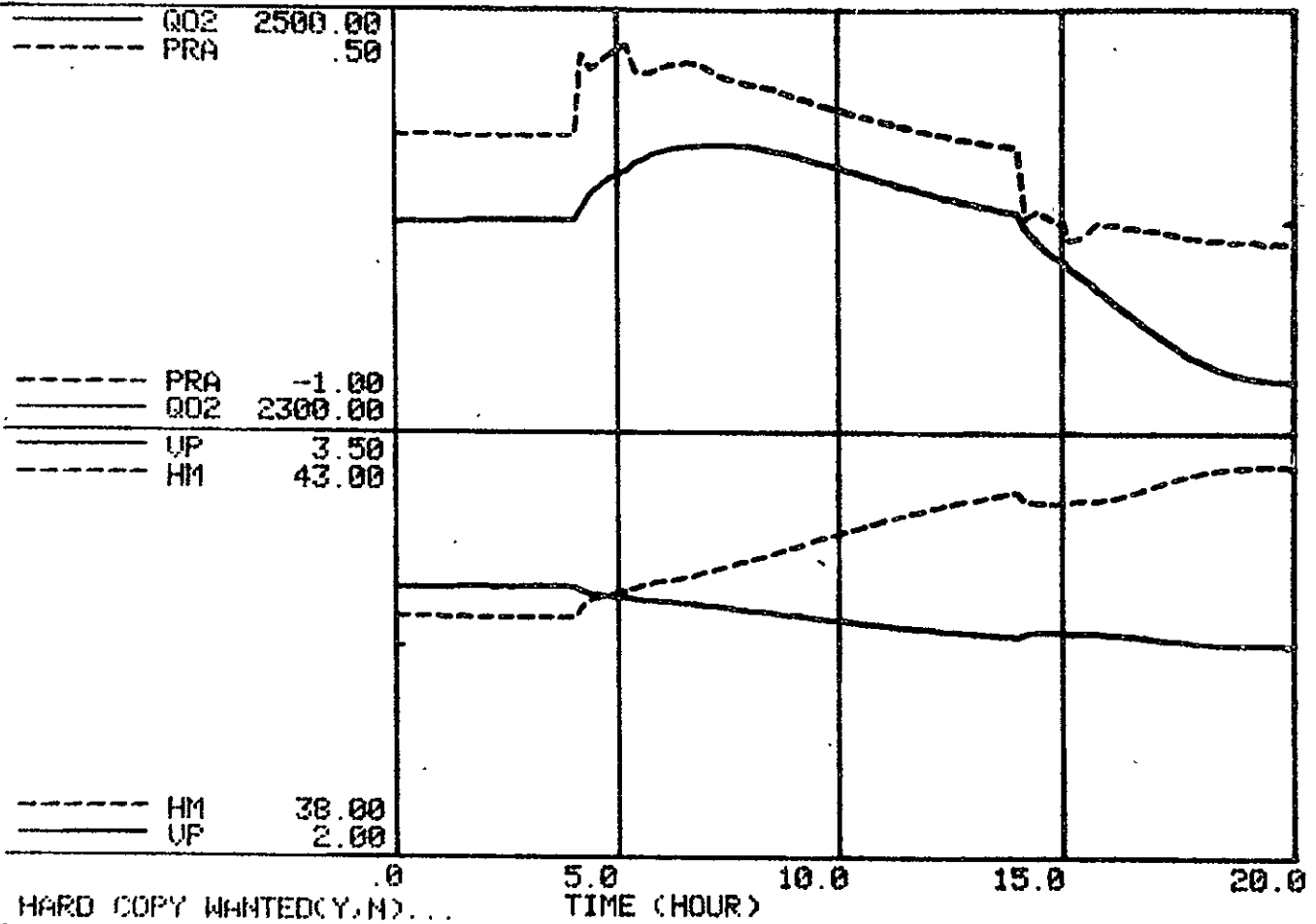
```

GRAPH OUTPUT WANTED(Y,N,S), TIME INTERVALS(A4,F2.0).
>Y 4.
  URC          Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>N
  UP          Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          2. 3.5 2.
  QO2        Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          1. 2500. 2300.
  PRA        Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          1. .5 -1.
  HM         Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>Y          2. 43. 38.
  I         Y SCALE (A4,8X,F4.0,2F6.0)
  PLOT(Y,N,S) LOC HIGH LOW ...
>N

```



1313 TEST FOR T10...



GRAPH OUTPUT WANTED(Y,N,S), TIME INTERVALS(A4,F2.0)...  
>N

OUTPUT WANTED OR SAME, STOP (6A4)...  
>STOP

APPENDIX E

LISTINGS OF SUBROUTINES

```

00100 1* C R67093.8 LARRY NEAL--MATH: NASA FILE
00100 2* C
00100 3* C PROGRAM GUYTON
00100 4* C CIRCULATORY DYNAMICS - CIRCE
00100 5* C CIRCE1
00101 6* REAL LVM,I,IFP,LPD,KE,KE1,KOD,KIR,KIE,KI,KCD,KED,KN1,KN3
00103 7* REAL NAE,NED,NID,NOD,I1,LPK,KID,MO2,NOZ,KCZ,HPI,HPR,I2,I3,MMO
00104 8* DIMENSION FUN1(14),FUN2(14),FUN3(14),FUN4(14),FUN6(14),FUN7(14)
00105 9* COMMON/ARRAY/T,I,VBD,VVS,VPA,VAS,VLA,VRA,VAE,PA,PAM,LVM,
00105 10* * VRE,PRA,QRN,VPE,PPA,PP1,CPA,RPA,RVM,VLE,PLA,QLN,PL1,
00105 11* * A1B,RPV,RPT,PGL,QPD,Q5,VVE,VV8,PVS,PGV,RVG,QVO,AVE
00106 12* COMMON/ARRAY/CN2,CN3,RVS,PGS,RTP,QAO,QRO,QLO,DVS,DPA,DAS,DLA,DRA,
00106 13* * PA1,AUC,AUB,AUN,AU6,AU2,AUB,DAU,AUJ,AU,AUO,AUH,VV4,
00106 14* * AU9,AUM,AU4,VIF,PO1,PTT,PTS,PIF,CPI,PTC,CPP,PPC,PVG
00107 15* COMMON/ARRAY/PC,PCD,VTC,PLD,VIL,VTD,VPD,DPL,CPI,DPC,DPI,LPD,DLP,
00107 16* * DPP,CHY,PRM,PGR,CPG,PGP,GF1,PGX,PGC,PGH,PG2,VDG,VG,
00107 17* * EPH,GPI,GP2,GPD,AAR,RR,RFN,APD,GLP,PEL,GFR,TRR,VUD
00110 18* COMMON/ARRAY/REK,NOD,NED,NAE,VEC,CKE,KOD,KE1,KIR,KIE,KCD,KED,CKI,
00110 19* * CNA,CCD,VID,KE,KI,VIC,I1,VTY,Z,VTZ,VUZ,TVZ,PPZ,
00110 20* * DFZ,X,I2,PR1,VTS,VP,PRP,IFP,GPR,KN3,KN1,AMR,AMP
00111 21* COMMON/ARRAY/AM1,AMC,AM2,AM3,AM5,AM,CNE,AGK,ANP,AN1,ANC,AN2,AN3,
00111 22* * AN5,ANM,VB,HM1,HM,BI,VIE,VIB,VIM,RC2,PO2,RKC,RC1,
00111 23* * RCD,VRC,RSN,QVA,BEN,DOB,AOM,PI0,OSV,POT,POD,POB,AR1
00112 24* COMMON/ARRAY/AR2,POC,AR3,ARM,CNB,GFN,AH7,AH8,AH,AHC,AH1,AH2,AH4,
00112 25* * AHM,CNY,CNX,VV1,VV2,VV5,VV6,VV7,IVD,VIW,HSR,NSI,NID,
00112 26* * SR,VVR,RAR,CV,CN7,AUX,AUK,AUZ,Y,CFC,CPK,PCE,CPR
00113 27* COMMON/ARRAY/LPK,DPO,HYL,KID,AMT,ANT,POK,PON,A1K,A2K,A3K,CNR,CNZ,

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

00113 28*      *      AHK,SRK,V9,V2D,Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8,HMK,
00113 29*      *      HKM,POV,POZ,RDO,Q02,RBF,M02,POA,POY,ANU,POR,GF2,HMD
00114 30*      *      COMMON/ARRAY/DHM,POQ,I3,U,VP1,T1,GF3,GF4,AUB,AUY,RV1,AUY,OUT,
00114 31*      *      DSP,AHZ,AHY,OSA,PPI,CPN,POS,PLF,PP0,PPN,PPD,PFI,DFP,
00114 32*      *      VPF,PPR,PMC,PMS,PMP,HR,CPF,PCP,DA1,DLZ,DPY,DPZ,GPZ
00115 33*      *      COMMON/ARRAY/NOZ,KCZ,VIZ,HPR,HPL,STH,ALO,EXC,O2M,PA2,PP2,SVO,AUL,
00115 34*      *      VV9,O2A,Q1,EXE,ARE,GRE,RSM,BEM,RAM,QVS,PVO,RMO,QDM,
00115 35*      *      PM0,P20,MM0,PDC,POE,AMM,A4K,POH,OMM,PM1,PM3,PM4,EX1
00116 36*      *      COMMON/ARRAY/Q2,Q3,PMS,PK1,Z9,Z10,Z11,Z12,Z13,Z14,Z15,Z16,PK2,
00116 37*      *      PK3,FIS,STA,PAR,GBL,ANY,ANZ,ANX,ANV,ANW,ANR,AUQ,AUR,
00116 38*      *      AUS,A378,H1,A2,A3,DUMMY(19),TITLE(400),DUMMY(40)
00117 39*      *      COMMON/NUMERO/K,NO(20),NTIMEC,UNITS,NZ,NTIMEP,NN:MAXNO,NTIME
00120 40*      *      COMMON/STORE/NG1,NG2,NG3,NG4,NG5,NG6,NG7,NG8,NG9,DT,TLP,TNP,ND,
00120 41*      *      TM,TMM,NFIRST,ZZ(15),OLY(9),OBY(9),YMIN(10),YMAX(10),
00120 42*      *      N,PT(18),BETA(10),NGRAPH(10),GRAPH(10),HEAD(19),NOEXP
00120 43*      *      ,DTMAX
00121 44*      *      COMMON/TAPE/TOTAL
00122 45*      *      DATA FUN1(1),FUN1(2),FUN1(3),FUN1(4),FUN1(5),FUN1(6),FUN1(7),
00122 46*      *      *FUN1(8),FUN1(9),FUN1(10),FUN1(11),FUN1(12),FUN1(13),FUN1(14)/
00122 47*      *      *0.,1,04,60.,1,025,125.,.97,160.,.88,200.,.59,240.,0.,240.,0./
00141 48*      *      DATA FUN2(1),FUN2(2),FUN2(3),FUN2(4),FUN2(5),FUN2(6),FUN2(7),
00141 49*      *      *FUN2(8),FUN2(9),FUN2(10),FUN2(11),FUN2(12),FUN2(13),FUN2(14)/
00141 50*      *      *-100.,0,0,-6.,0,0,-3.,.75,-1.,2.6,2.,.9,8,8.,13,5,1000.,13,5/
00160 51*      *      DATA FUN3(1),FUN3(2),FUN3(3),FUN3(4),FUN3(5),FUN3(6),FUN3(7),
00160 52*      *      *FUN3(8),FUN3(9),FUN3(10),FUN3(11),FUN3(12),FUN3(13),FUN3(14)/
00160 53*      *      *0.0,1.06,20.,.97,24.,.93,30.,.8,38.,.46,45.,0.,45.,0./
00177 54*      *      DATA FUN4(1),FUN4(2),FUN4(3),FUN4(4),FUN4(5),FUN4(6),FUN4(7),
00177 55*      *      *FUN4(8),FUN4(9),FUN4(10),FUN4(11),FUN4(12),FUN4(13),FUN4(14)/
00177 56*      *      *-100.,0.,.4.,0.,-1.,3.6,3.,.9,4.6.,11,6,10.,13,5,1000.,13,5/
00216 57*      *      DATA FUN6(1),FUN6(2),FUN6(3),FUN6(4),FUN6(5),FUN6(6),FUN6(7),
00216 58*      *      *FUN6(8),FUN6(9),FUN6(10),FUN6(11),FUN6(12),FUN6(13),FUN6(14)/
00216 59*      *      *-100.,10000.,0.,70.,.4,9,3.,.8,3,3,1,2,1,3,1,6.,.43,100.,0./
00235 60*      *      DATA FUN7(1),FUN7(2),FUN7(3),FUN7(4),FUN7(5),FUN7(6),FUN7(7),
00235 61*      *      *FUN7(8),FUN7(9),FUN7(10),FUN7(11),FUN7(12),FUN7(13),FUN7(14)/
00235 62*      *      *0.,7.,30.,.8,25,60.,.3.,100.,.1.,160.,.15,400.,.05,400.,.05/
00254 63*      *      COMMON/DEMAND/ITAPEO,ITME,ISEXCN,ICONVI
00255 64*      *      DATA IN000/*N /*/,IY000/*Y /*/
00255 65*      C
00260 66*      WRITE(6,724)
00262 67*      724 FORMAT('IGUYTON MODEL FROM WHITE'/
00262 68*      1 ' REFER TO GF-ASG USER GUIDE ',
00262 69*      2 'TIR 741-MED-4004')
00263 70*      90 CALL SEGID (4)
00263 71*      C SEE IF CONVERSATIONAL INPUT.
00264 72*      19 WRITE(6,20)
00266 73*      20 FORMAT('CONVERSATIONAL INPUT (A4) Y,N...')
00267 74*      READ(5,21,ERR=19) ICONVI
00272 75*      21 FORMAT(A4)
00273 76*      IF(ICONVI.EQ.IN000) GO TO 25
00275 77*      IF(ICONVI.NE.IY000) GO TO 19
00275 78*      C SEE IF TO OUTPUT TAPE.
00277 79*      25 WRITE(6,26)
00301 80*      26 FORMAT('WANT OUTPUT TAPE (A4) Y,N...')
00302 81*      READ(5,21,ERR=25) ITAPEO
00305 82*      IF(ITAPEO.EQ.IN000) GO TO 30
00307 83*      IF(ITAPEO.NE.IY000) GO TO 25
00307 84*      C

```

```

00311 85* 30 IF(ICONV1 .EQ. 1Y000) GO TO 40
00311 86* C HERE IF NOT CONVERSATIONAL MODE.
00313 87* WRITE(6,32)
00315 88* 32 FORMAT('DADD DATA FILE...')
00316 89* CALL PUTIN
00317 90* GO TO 38
00317 91* C HERE IF IN CONVERSATIONAL MODE.
00320 92* 40 WRITE(6,721)
00322 93* 721 FORMAT(' ADD DATA FILE(DADD TTYDAT)...')
00323 94* CALL PUTIN
00324 95* CALL TTYIN
00325 96* ITME = 1
00326 97* CALL TTYOUT
00326 98* C
00327 99* 38. IF(1 .GT. 0.5) I=0.5
00331 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00331 100* 100 IF(OUT.NE.31) GO TO 101
00333 101* CALL SEGLD (5)
00334 102* IF(ICONV1 .EQ. 1Y000) GO TO 50
00336 103* CALL PUTOUT
00337 104* GO TO 101
00337 105* C HERE IF CONVERSATIONAL MODE.
00340 106* 50 IEXECN = 0
00341 107* ITME = 0
00342 108* CALL TTYOUT
00343 109* IF(IEXECN .EQ. 0) GO TO 101
00343 110* C HERE WHEN FINISHED ALL TIME STEPS INPUT.
00345 111* CALL TTYIN
00346 112* ITME = 2
00347 113* CALL TTYOUT
00347 114* C
00347 115* C100 IF(OUT .EQ. 3.) CALL OUTPUT
00347 116* C IF(DSP .EQ. 3.) CALL DISPLAY
00350 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00350 117* 101 IF(DSP.NE.3.) GO TO 102
00352 118* CALL SEGLD (4)
00353 119* CALL DISPLAY
00353 120* C
00354 121* 102 I=T+12
00354 122* C
00355 123* CALL SEGLD.(1)
00356 124* CALL HEMO (FUN1,FUN2,FUN3,FUN4)
00356 125* C CALL HEMO (AMM,ANM,ANU,ANY,ANZ,ARM,AUH,AUM,AUY,AVE,BEM,BFN,
00356 126* C * CN2,CN3,CN7,CV ,DAS,DLA,DPA,DRA,DVS,FIS,HMD,HPL,
00356 127* C * HPR,HSL,HSR,I2 ,LVM,PA ,BA1,PA2,BC ,PGL,PGS,PLA,
00356 128* C * PPA,PP1,PP2,PRA,PR1,PVS,QA0,QLN,QL0,QPO,QRF,QRN,
00356 129* C * QRO,QVO,RAM,RAR,RBF,RPA,RPT,RPV,RSM,RSN,RVG,RVM,
00356 130* C * RVS,U ,VAE,VAS,VBD,VIM,VLA,VLE,VP ,VPA,VPE,VRA,
00356 131* C * VRC,VRE,VVE,VVR,VVS,VV7,VV8,X ,FUN1,FUN2,FUN3,
00356 132* C * FUN4)
00356 133* C
00357 134* CALL AUTO (AU ,AUB,AUC,AUH,AUJ,AUK,AUL,AUM,AUN,AUO,AUP,AUQ,
00357 135* * AUR,AUS,AUV,AUX,AUZ,AU4,AU6,AUB,A1B,DAU,EXC,EXE,
00357 136* * EX1,I2 ,PA ,PA1,POQ,POT,P20,STA,VVR,VV9,Y ,Z,
00357 137* * Z8 ,Z1,Z1)
00357 138* C
00360 139* IF(13.LE.12)GO TO 168

```

ORIGINAL PAGE IS OF POOR QUALITY

```

00362 140* IF (ABS(QAU-AUJ).GT,DA1) GO TO 100
00364 141* 100 IF (ABS(QAO-QLO).GT.,2) GO TO 100
00366 142* IF (ABS(QAO-QPO).GT.,2) GO TO 100
00370 143* IF (ABS(QAO-QRO).GT.,4) GO TO 100
00370 144* C
00372 145* 168 CALL SEGLD (2)
00373 146* CALL HORMON (AH,AMC,AMP,AMR,AMT,AMI,ANM,CKE,PA,Z,FUN7,
00373 147* AGK,ANC,ANP,ANR,ANT,ANV,ANW,ANI,CNA,CNE,GFN,
00373 148* I,REK,A2,A3)
00373 149* C
00374 150* CALL BLOOD (HKM,HM,HMK,I,POT,POY,PO1,PO2,RC1,RC2,RCO,RKC,
00374 151* VB,VIB,VIE,VIM,VP,VRC)
00374 152* C
00375 153* CALL MUSCLE (ALO,AMM,AOM,AUP,A4K,BFM,EXC,HM,I,MMO,OMM,OSA,
00375 154* OVA,OVS,O2A,POD,PK1,PK2,PK3,PMO,PM1,PM3,PM4,PM5,
00375 155* POE,POM,PVO,P20,QOM,RMO,VPF,Z5,Z6)
00375 156* C
00376 157* CALL AUTORG (AOM,ARM,AR1,AR2,AR3,A1K,A2K,A3K,BFN,DOB,HM,I,
00376 158* MO2,OSV,OYA,O2M,POA,POB,POC,POD,POK,PON,POR,POT,
00376 159* POV,POZ,P10,QO2,RDO,Z,Z4,Z7)
00376 160* C
00377 161* CALL ADH (AH,AHC,AHK,AHM,AHY,AHZ,AH7,AH8,AUP,CNA,CNB,CNR,
00377 162* CNZ,I,PRA,Z)
00377 163* C
00400 164* CALL MISC1 (AHM,AU4,AUB,I,SR,SRK,STH,TVD,TVZ,VEC,VIC,VTW,
00400 165* VVE,VV6,VV7,Z,V9)
00400 166* C
00401 167* CALL HEART (AUR,DHM,HMD,HR,I,PA,PMC,PMP,PMS,POT,PRA,QAO,
00401 168* QLO,RTP,SVO,VAE,VLE,VPE,VRE,VVE,HI)
00401 169* C
00402 170* CALL CAPMBD (BEN,CFC,CPI,CPP,DEB,I,IEP,PC,PCD,PIE,PLD,PPC,
00402 171* PRP,PTC,PTS,PTT,PVG,PVS,RVS,TVD,VG,VID,VIF,VP,
00402 172* VPD,VTC,VTD,VTL,VTS,VUD,Z,Z1,FUN6)
00402 173* C
00403 174* I=I+1.2+T-T1
00404 175* I1=ABS(VP1/VPD/I)
00405 176* IF(I1.LT,I) I=I1
00407 177* IF(I3+T-T1.LT,I) I=I3+T-T1
00411 178* T=I+T1
00412 179* T1=T
00412 180* C
00413 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00413 181* IF(OUT.NE.4.) GO TO 199
00415 182* CALL SEGLD (5)
00416 183* CALL PUTOUT
00416 184* C
00416 185* C IF(OUT.EQ.4.) CALL OUTPUT
00416 186* C IF(DSP.EQ.4.) CALL DSPLAY
00417 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00417 187* 199 IF(DSP.NE.4.) GO TO 200
00421 188* CALL SEGLD (4)
00422 189* CALL DSPLAY
00422 190* C
00423 191* 200 CALL SEGLD (3)
00424 192* CALL PULMON (CRF,CRP,CRN,DFR,I,PCP,PE1,PLA,PLF,POS,PRA,PPC,
00424 193* PPD,PPI,PPN,PPO,PPR,VP,VPD,VPF,Z,Z3)
00424 194* C

```

00425	195*		CALL MISC2	(HPL,HPR,HSL,HSR,I,PA,PPA,ROI,SIH,ZIQ,ZII,ZI3)
00425	196*	C		
00426	197*		CALL PROTEN	(CHY,CPG,CPI,CPK,CPP,CPR,CBI,DLP,DLZ,DEC,DEI,DPL,
00426	198*		*	DPO,DPY,GPD,GPR,I,IFP,LPK,PC,PCE,PGX,PRP,VG,
00426	199*		*	VTL,Z,PPD)
00426	200*	C		
00427	201*		CALL KIDNEY	(AAR,AHM,AM,APD,ARE,AUM,CNE,CNX,CNY,GBL,GEN,GFR,
00427	202*		*	GF2,GF3,GF4,GLP,I,NAE,NED,NID,NOD,NOZ,PA,PAR,
00427	203*		*	PFL,PPC,RBF,REK,RFN,RR,SIH,TRR,VIM,VUD,Z)
00427	204*	C		
00430	205*		CALL IONS	(AM,CCD,CKE,CKI,CNA,I,KCD,KE,KED,KI,KID,KIE,
00430	206*		*	KIR,KOD,NAE,REK,VEC,VIC,VID,VP,VPP,VTS,Z)
00430	207*	C		
00431	208*		CALL GELFLO	(CHY,CPG,CPI,GPR,HYL,IFP,PGC,PGH,PGP,PGR,PGX,PIF,
00431	209*		*	PRM,PTC,PTS,PTT,VG,VGD,VIF,VRS,VTS,V2D,FUN6)
00431	210*	C		
00432	211*		GO TO 100	
00433	212*		END	

END OF COMPILATION: 4 DIAGNOSTICS.



```

00101      1*      SUBROUTINE ADH (AH ,AHC,AHK,AHM,AHY,AHZ,AH7,AH8,AUP,CNA,CNB,CNR,
00101      2*      *      CNZ,I ,PRA,Z)
00103      3*      REAL I
00103      4*      C
00103      5*      C      ANTIIDIURETIC HORMONE
00103      6*      C
00104      7*      CNB=CNA-CNR
00105      8*      AHZ=.2*PRA
00106      9*      AHY=AHY+(AHZ-AHY)*.0007*I
00107     10*      AH8=AUP-.1.
00110     11*      IF(AH8.LT.0.)AH8=0.
00112     12*      IF(CNB.LT.0.)CNB=0.
00114     13*      AH=AH+(CNZ*CNB+AH8-AHZ+AHY-AH)/Z
00115     14*      IF(AH.LT.0.)AH=0.
00117     15*      AHC=AHC+(.3333*AH-AHC)*(1.-EXP(-I/AHK))
00120     16*      AHM=6.*{1.-EXP(-0.1808*AHC)}
00121     17*      IF(AHM.LT..3)AHM=.3
00123     18*      RETURN
00124     19*      END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101 1* SUBROUTINE AUTO (AU ,AUB,AUC,AUH,AUJ,AUK,AUL,AUM,AUN,AUO,AUP,AUQ,
00101 2* * AUR,AUS,AUV,AUX,AUZ,AU4,AU6,AUB,AIB,DAU,EXC,EXE,
00101 3* * EXI,I2 ,PA ,PAI,POQ,POT,P20,STA,VVR,VV9,Y ,Z,
00101 4* * ZB ,Z12)
00103 5* REAL I2
00103 6* C
00103 7* C AUTONOMIC CONTROL BLOCK
00103 8* C
00104 9* 120 EXE=(8,-P20)*EXI+(EXC-1.)*Z12
00105 10* POQ=POT
00106 11* IF (POQ.GT.8.)POQ=8.
00110 12* IF (POQ.LT.9.)POQ=9.
00112 13* PAI=PA*POQ/8.-EXE
00113 14* AUC=0.
00114 15* IF (PAI.LT.80.)AUC=.03*(80.-PAI)
00116 16* IF (PAI.LT.40.)AUC=1.2
00120 17* AUB=0.
00121 18* IF (PAI.LT.170.)AUB=.014286*(170.-PAI)
00123 19* IF (PAI.LT.40.)AUB=1.83
00125 20* 123 AIB=(AUB-1.)*AUX+1.
00126 21* 124 AUN=0
00127 22* IF (PAI.LT.50.)AUN=.2*(50.-PAI)
00131 23* IF (PAI.LT.20.)AUN=6.0
00133 24* AU6=AIB-AU4
00134 25* AUB=AUK*(AU6-1.)
00135 26* DAU=DAU+(AUC+AU6+AUN-DAU)/Z/Y
00136 27* AUJ=AUJ+(DAU-AUJ)*I2*6./ZB
00137 28* IF (AUJ.LT.0.)AUJ=0.
00141 29* IF (AUJ-1.)126,127,127
00144 30* 126 AU=AUJ**AUZ
00145 31* GO TO 128
00146 32* 127 AU=(AUJ-1.)*AUZ+1.
00147 33* 128 IF (STA.GT.,00001)AU=STA
00151 34* AUB=AU-1.

```

```
00152 35* AUP=AUD*ADQ+1.
00153 36* AUH=AUD*ADV+1.
00154 37* AUR=AUD*AUS+1.
00155 38* VVR=VV9-AUL*AUP
00156 39* AQM=.15+.85*AUP
00157 40* RETURN
00160 41* END
```

END OF COMPILATION: NO DIAGNOSTICS.

ORIGINAL PAGE IS  
OF POOR QUALITY

```

00101 1* SUBROUTINE AUTORG(AOM,ARM,ARI,AR2,AR3,A1K,A2K,A3K,BFN,DOB,HM,I,
00101 2* * MQ2,OSV,OVA,Q2M,POA,POB,POC,PQD,PQK,PON,POR,POI,
00101 3* * POV,POZ,P10,Q02,RD0,Z,Z4,Z7)
00103 4* REAL I,MQ2
00103 5* C
00103 6* C NON-MUSCLE OXYGEN DELIVERY BLOCK
00103 7* C AND NON-MUSCLE LOCAL BLOOD FLOW CONTROL BLOCK
00103 8* C -----
00103 9* C AUTOREGULATION,RAPID
00103 10* C -----
00104 11* OSV=OSV+((BFN*OVA-DOB)/HM/5./BFN-OSV)/Z7
00105 12* POV=OSV*57.14
00106 13* RD0=POT**3.
00107 14* IF(RD0.LT.50.)RD0=50.
00111 15* DOB=(POV-POT)*2896.5/RD0
00112 16* MQ2=AOM*Q2M*(1.-(8.0001=P10.)**3./512.)
00113 17* Q02=Q02+(DOB-MQ2)*(1.-EXP(-1/Z4))
00114 18* POT=Q02*.00333
00115 19* P10=POT
00116 20* IF(POT.GT.8.)P10=8.
00120 21* POD=POV-POR
00121 22* POB=POB+(PQK*POD+1.-POB)/Z
00122 23* IF(POB.LT..2)POB=.2
00124 24* ARI=ARI+(POB-ARI)*(1.-EXP(-1/A1K))
00125 25* ARM=ARI*AR2*AR3
00125 26* C -----
00125 27* C AUTOREGULATION,INTERMEDIATE
00125 28* C -----
00126 29* POA=POA+(PON*POD+1.-POA)/Z
00127 30* IF(POA.LT..5)POA=.5
00131 31* AR2=AR2+(POA-AR2)*(1.-EXP(-1/A2K))
00131 32* C -----
00131 33* C AUTOREGULATION,LONG-TERM
00131 34* C -----

```

```
00132 35* IF (POD) 194, 192, 192
00135 36* 192 POC=POZ*POD+1.
00136 37* GO TO 196
00137 38* 194 POC=POZ*POD*.33+1.
00140 39* 196 IF (POC*LT.3) POC=.3
00142 40* AR3=AR3+(POC-AR3)*I/A3K
00143 41* RETURN
00144 42* END
```

END OF COMPILATION: NO. DIAGNOSTICS.

ORIGINAL FILED  
OF ROOM QUALITY

```

00101 1* SUBROUTINE BLOOD (HKM, HM, HMK, I, POT, POY, PO1, PO2, RC1, RC2, RCD, RKC,
00101 2* * VB, VIB, VIE, VIM, VP, VRC)
00103 3* REAL I
00103 4* C
00103 5* C RED CELLS AND VISCOSITY BLOCK
00103 6* C-----
00103 7* C BLOOD VISCOSITY
00103 8* C-----
00104 9* 170 VB=VP+VRC
00105 10* HM=100.*VRC/VB
00106 11* VIE=HM/(HMK-HM)/HKM
00107 12* VIB=VIE+1.5
00110 13* VIM=.3333*VIB
00110 14* C-----
00110 15* C RED BLOOD CELLS
00110 16* C-----
00111 17* RC2=RKC*VRC
00112 18* PO2=PO1=POT
00113 19* IF(PO2.LT..2375)PO2=.2375
00115 20* RC1=POY*PO2
00116 21* RCD=RC1-RC2
00117 22* VRC=VRC+RCD*I
00120 23* RETURN
00121 24* END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101      1*      SUBROUTINE CAPMBD(BFN,CFC,CPI,CRP,DFP,I ,IFP,PC ,PCD,PIF,PLD,PPC,
00101      2*      *          PRP,PTC,PTS,PTT,PVG,PVS,RVS,TVD,VG ,VID,VIF,VP,
00101      3*      *          VPD,VTC,VTD,VTL,VTS,VUD,Z ,Z1,FUN6)
00103      4*      DIMENSION FUN6(14)
00104      5*      REAL I,IFP
00104      6*      C
00104      7*      C      CAPILLARY MEMBRANE DYNAMICS BLOCK
00104      8*      C
00105      9*      130 PTF=(VTS/12.)*2.
00106     10*      VIF=VTS-VG
00107     11*      CALL FUNCTN (VIF,PTS,FUN6)
00110     12*      PIF=PTT-PTS
00111     13*      CPI=IFP/VIF
00112     14*      PTC=.25*CPI
00113     15*      CPP=PRP/VP
00114     16*      PPC=.4*CPP
00115     17*      PVG=RVS*1.79*BFN
00116     18*      PC=PVG+PVS
00117     19*      PCD=PC+PTC-PPC-PIF
00120     20*      VTC=VTC+(CFC*PCD-VTC)/Z
00121     21*      PLD=7.8+PIF-PTT
00122     22*      VTL=VTL+(.004*PLD-VTL)/Z
00123     23*      IF(VTL.LT.0.)VTL=0.
00125     24*      VTD=VTC-VTL-VID
00126     25*      VTS=VTS+VTD*I
00127     26*      VPD=VPD+(VTD-VTC+VTL-VUD-DFP-VPD)/Z1
00130     27*      RETURN
00131     28*      END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101 1*      SUBROUTINE DSHARC(X2,Y2,L,I,N,IDUP)
00101 2*      C THIS ROUTINE WILL DRAW ONE OF 5 TYPES OF DASHED LINES
00101 3*      C FROM THE CURRENT IMAGINARY BEAM TO VIRTUAL COORS X2,Y2,
00101 4*      C L=DASH WANTED(1-5), I=WHICH IMAGE OF DASH THIS PROG.AT,
00101 5*      C N=NO. RASTERS REMAINING ON CURRENT IMAGE..IDUP=WHEN TO DUMP BUFFER.
00101 6*      C I SHOULD BE INITIALIZED TO 0 AT START OF A LINE.
00101 7*      C THIS ROUTINE WILL UPDATE I,N,AND IDUP.
00103 8*      COMMON/TKTRNX/KBAUDR,KERROR,KGRAFL,KHOMEY,KKMODE,
00103 9*      1 KHORSZ,KVERSZ,KITALC,KSIZEF,KLMRGN,KRMRGN,
00103 10*     2 KTBSZ,KHORZT(10),KVERTT(10),
00103 11*     3 KBEAMX,KBEAMY,KMOVFF,KPCHAR(4),
00103 12*     4 KMINSX,KMINSY,KMAXSX,KMAXSY,TMINVX,TMINVY,TMAXVX,TMAXVY,
00103 13*     5 TREALX,TREALY,TIMAGX,TIMAGY,TRCOSF,TRSINF,TRSCAL

```



```

00104 14*      DIMENSION ID(7,5)
00105 15*      DATA ID/3,10,7,4*0.5,20,7,5,7,2*0.3,20,7,4*0,
00105 16*      1 7,20,7,5,7,5,7,3,40,7,4*0/
00107 17*      RN = N
00110 18*      LAST = 0
00110 19*      C COMPUTE VIRTUAL AND SCREEN FACTORS TO APPLY TO X2,Y2.
00111 20*      X = (KMAXSX - KMINSX) / (IMAXVX - TMINVX)
00112 21*      Y = (KMAXSY - KMINSY) / (TMAXVY - TMINVY)
00112 22*      C COMPUTE ANGLE BETWEEN THIS POINT AND PREVIOUS POINT.
00113 23*      X1 = ABS(X2-TIMAGX) * X
00114 24*      Y1 = ABS(Y2-TIMAGY) * Y
00115 25*      IF(Y1 .LE. 0.) Y1 = .0000001
00117 26*      A = ATAN(X1/Y1)
00120 27*      XS = TIMAGX
00121 28*      YS = TIMAGY
00121 29*      C COMPUTE INTERMEDIATE POINTS FOR DASHES AND SKIPS(XS,YS ARE
00121 30*      C START AND XE,YE ARE END OF A COMPUTED DASH OR SKIP).
00121 31*      C RN=NO,RASTERS FOR A DASH/SKIP.
00122 32*      IE(I,GE, 2) GO TO 6
00124 33*      I = 2
00125 34*      5 RN = ID(I,L)
00125 35*      C SEE IF ON SKIP OR DASH.
00126 36*      6 ISKP = 0
00127 37*      Z = FLOAT(I) / 2.
00130 38*      Z1 = I / 2
00131 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00131 39*      IE(Z,NE,Z1) ISKP = 1
00131 40*      C COMPUTE VIRTUAL X FOR END OF IMAGE(XE).
00133 41*      Z = SIN(A) * RN / X
00134 42*      IF(TIMAGX .GT. X2) GO TO 10
00134 43*      C HERE IE X GETTING LARGER.
00136 44*      XE = XS + Z
00137 45*      IF(XE .LT. X2) GO TO 74
00137 46*      C HERE IF INTERMEDIATE POINT WENT BEYOND ACTUAL POINT.
00141 47*      GO TO 15
00141 48*      C HERE IF X GETTING SMALLER.
00142 49*      10 XE = XS - Z
00143 50*      IF(XE .GT. X2) GO TO 74
00143 51*      C HERE IF INTERMEDIATE POINT WENT BEYOND ACTUAL POINT.
00145 52*      15 XE = X2
00146 53*      YE = Y2
00147 54*      X = (XE - XS) * X
00150 55*      Y = (YE - YS) * Y
00151 56*      W = SQRT((X**2) + (Y**2))
00152 57*      N = RN * W
00153 58*      IF(N .LE. 0) N = 1
00155 59*      LAST = 1
00156 60*      GO TO 80
00156 61*      C HERE IF STILL BETWEEN 2 ACTUAL POINTS.
00156 62*      C COMPUTE VIRTUAL Y FOR END OF IMAGE (YE).
00157 63*      74 Z = COS(A) * RN / Y
00160 64*      IF(TIMAGY .GT. Y2) Z = Z * (-1.)
00162 65*      YE = YS + Z
00163 66*      80 IF(ISKP .EQ. 1) GO TO 82
00165 67*      CALL DRAWA(XE,YE)
00166 68*      GO TO 90
00167 69*      82 CALL MOVEA(XF,YE)

```

```
00170 70* 90 IDUP = IDUP + 1
00171 71* IF (IDUP .LT. 10) GO TO 95
00173 72* CALL DMPBUF
00174 73* IDUP = 1
00175 74* 95 IF (LAST .EQ. 1) RETURN
00177 75* I = I + 1
00200 76* JF (I .GT. 10(L)) I = 2
00202 77* XS = XE
00203 78* YS = YE
00204 79* GO TO 5
00205 80* END
```

END OF COMPILATION:

1 DIAGNOSTICS.

```

00101 1* SUBROUTINE DSPLAY
00103 2* COMMON/ARRAY/A(400),TITLE(400),COL(20),ALPHA(20)
00104 3* COMMON/NUMERO/K,N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,
00104 4* * N11,N12,N13,N14,N15,N16,N17,N18,N19,N20,
00104 5* * NTIMEC,UNITS,NZ,NTIMEP,NN,MAXNO,NTIME
00105 6* COMMON/STORE/NG1,NG2,NG3,NG4,NG5,NG6,NG7,NG8,NG9,DT,TLP,TNP,ND,
00105 7* * TM,TMM,NFIRST,ZZ(15),QLY(9),OBY(9),YMIN(10),YMAX(10),
00105 8* * N,PT(18),BETA(10),NGRAPH(10),GRAPH(10),HEAD(19),NOEXP
00105 9* * ,DTMAX
00106 10* IF(NFIRST) 200,100,200
00111 11* 100 READ(5,101) TOTAL
00114 12* 101 FORMAT(F10.1)
00115 13* NFIRST=1
00116 14* GO TO 2
00117 15* 200 WRITE(7) A(1),A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),
00117 16* * A(N8),A(N9),A(N10),A(N11),A(N12),A(N13),A(N14),A(N15)
00141 17* IF(A(1)-TOTAL) 2,3,3
00144 18* 2 RETURN
00145 19* 3 CALL NTRAN(7,9)
00146 20* STOP
00147 21* END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101      1*      SUBROUTINE FUNCTN(TH,POL,TAB)
00103      2*      DIMENSION TAB(14)
00104      3*      N=14
00105      4*      DO 110 I=1,N,2
00110      5*      IF(TAB(I).EQ.TH) 110,120,110
00113      6*      110 CONTINUE
00115      7*      GO TO 140
00116      8*      120 POL=TAB(I+1)
00117      9*      130 RETURN
00120     10*      140 NN=N-2
00121     11*      DO 150 I=1,NN,2
00124     12*      150 IF(TAB(I) .LT. TH .AND. TAB(I+2) .GT. TH) GO TO 160
00127     13*      WRITE(6,100) TH
00132     14*      100 FORMAT(5X,' ***** CURVE LIMITS EXCEEDED ***** ',G12.6//)
00133     15*      IF(TH .LT. TAB(I)) POL=TAB(I+2)
00135     16*      IF(TH .GT. TAB(N-1)) POL=TAB(N)
00137     17*      GO TO 130
00140     18*      160 POL=TAB(I+1)+(TAB(I+3)-TAB(I+1))*((TH-TAB(I))/(TAB(I+2)-TAB(I)))
00141     19*      GO TO 130
00142     20*      END

```

END OF COMPILATION: NO DIAGNOSTICS.

ORIGINAL PAGE IS  
OF POOR QUALITY

```

00101      1*      SUBROUTINE GELFLD(CHY,CPG,CPI,GPR,HYL,IFP,PGC,PGH,PGP,PGR,PGX,PIF,
00101      2*      *      PRM,PTC,PTS,PTJ,VG ,VGD,VIF,VRS,VIS,V2D,EUN6)
00103      3*      DIMENSION FUN6(14)
00104      4*      REAL IFP
00104      5*      C
00104      6*      C      GEL FLUID DYNAMICS
00105      7*      140  CHY=HYL/VG
00106      8*      PRM=-5.9*CHY+24.2
00107      9*      PGR=.4*CHY
00110     10*      CPG=GPR/VG
00111     11*      PGP=.25*PGX
00112     12*      PGC=PGP+PGR
00113     13*      VIF=VTS-VG
00114     14*      CALL FUNCIN (VIF,PTS,FUN6)
00115     15*      PIF=PTT-PTS
00116     16*      CPI=IFP/VIF
00117     17*      PTC=.25*CPI
00120     18*      PGH=PIF+PTS+PRM
00121     19*      VGD=V2D*(PIF+PGC-PTC-PGH)
00122     20*      VG=VG+VGD
00123     21*      IF(VG.LT.0.)VG=0.
00125     22*      IF(.012.LT.ABS(VGD)). GO. TO 140
00127     23*      RETURN
00130     24*      END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101 1*      SUBROUTINE GOTOCH(I,J)
00101 2*      C THIS ROUTINE MOVES THE A/N CURSOR TO LINE I COL J.
00103 3*      IY = 767 - ((I-1) * 21)
00104 4*      IX = J * 12
00105 5*      CALL MOVABS(IX,IY)
00106 6*      CALL ANMODE
00107 7*      RETURN
00110 8*      END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101      1*      SUBROUTINE HEART (AUR,DHM,HMD,HR ,I ,PA ,PMC,PMP,PMS,POT,PRA,QAQ,
00101      2*      *      QLO,RTP,SVO,VAE,VLE,VPE,VRE,VVE,H1)
00103      3*      REAL I
00103      4*      C
00103      5*      C HEART HYPERTROPHY OR DETERIORATION BLOCK
00103      6*      C
00103      7*      C
00103      8*      C HEART VICIOUS CYCLE
00103      9*      C
00104      10*     DHM=(POT-6.)*.0025
00105      11*     HMD=HMD+DHM*I
00106      12*     IF (HMD.GT.1.)HMD=1.
00106      13*     C
00106      14*     C MEAN CIRCULATORY PRESSURES
00106      15*     C
00110      16*     PMC=(VAE+VVE+VRE+VPE+VLE)/.11
00111      17*     PMS=(VAE+VVE+VRE)/.09375
00112      18*     PMP=(VPE+VLE)/.01625
00112      19*     C*****
00112      20*     C
00112      21*     C HEART RATE AND STROKE VOLUME BLOCK AND TOTAL PERIPHERAL RESISTANCE
00112      22*     C
00112      23*     C*****
00113      24*     HR=(32.+H1 *AUR+PRA*2.)*((HMD-1.)*.5+1.)
00114      25*     RIP=(PA=PRA)/QAQ
00115      26*     SVO=QLO/HR
00116      27*     RETURN
00117      28*     END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101 1* SUBROUTINE HEMO (FUN1,FUN2,FUN3,FUN4)
00103 2* DIMENSION FUN1(14),FUN2(14),FUN3(14),FUN4(14)
00103 3* C SUBROUTINE HEMO (AMM,ANM,ANU,ANY,ANZ,ARM,AUH,AUM,AUY,AVE,BFM,BFN,
00103 4* C CNZ,CN3,CN7,CV,DAS,DLA,DPA,DRA,DVS,FIS,HMD,HPL,
00103 5* C HPR,HSL,HSR,I2,LVM,PA,PAM,PAZ,PC,PGL,PGS,PLA,
00103 6* C PPA,PP1,PP2,PRA,PR1,PVS,QAO,QLN,QLO,QPO,QRF,QRN,
00103 7* C QRO,QVO,RAM,RAR,RBF,RPA,RPT,RPV,RSM,RSN,RVG,RVM,
00103 8* C RVS,U,VAE,VAS,VBD,VIM,VLA,VLE,VP,VPA,VPE,VRA,
00103 9* C VRC,VRE,VVE,VVR,VVS,VV7,VV8,X,FUN1,FUN2,FUN3,
00103 10* C FUN4)
00103 11* C REAL I2,LVM
00103 12* C DIMENSION FUN1(14),FUN2(14),FUN3(14),FUN4(14)
00104 13* REAL LVM,I,IFP,LPD,KE,KE1,KOD,KIR,KIE,KI,KCD,KED,KN1,KN3
00105 14* REAL NAE,NED,NID,NOD,I1,LPK,KID,MOZ,NOZ,KCZ,HPL,HPR,I2,I3,MMO
00106 15* COMMON/ARRAY/T,I,VBD,VVS,VPA,VAS,VLA,VRA,VAE,PA,PAM,LVM,
00106 16* VRE,PRA,QRN,VPE,PPA,PP1,CPA,RPA,RVM,VLE,PLA,QLN,PL1,
00106 17* AIB,RPV,RPT,PGL,QPO,Q5,VVE,VV8,PVS,PGV,RVG,QVO,AVE
00107 18* COMMON/ARRAY/CN2,CN3,RVS,PGS,RTF,QAO,QRO,QLO,DVS,DPA,DAS,DLA,DRA,
00107 19* PA1,AUC,AUB,AUN,AU6,AU2,AUB,DAU,AUJ,AU, AUO,AUH,VV4,
00107 20* AU9,AUH,AU4,VIF,PO1,PTT,PTS,PIF,CPI,PTC,CPP,PPC,PVG
00110 21* COMMON/ARRAY/PC,PCD,VTC,PLD,VIL,VTD,VPD,DPL,CP1,DPC,DR1,LRD,DLP,
00110 22* DPP,CHY,PRM,PGR,CPG,PGP,GF1,PGX,PGC,PGH,PG2,VGD,VG,
00110 23* EPH,GPI,GP2,GPD,AAR,RR,RFN,APD,GLP,RFL,GER,TRR,VUD
00111 24* COMMON/ARRAY/REK,NOD,NED,NAE,VEC,CKE,KOD,KE1,KIR,KIE,KCD,KED,CKI,
00111 25* CNA,CCD,VID,KE,KI,VIC,I1,VTY,Z,VTZ,VUZ,VZ,PPZ,
00111 26* DFZ,X,I2,PR1,VTS,VP,PRP,IFP,GPR,KN3,KN1,AMR,AMP
00112 27* COMMON/ARRAY/AM1,AMC,AM2,AM3,AM5,AM,CNE,AGK,ANP,AN1,ANC,AN2,AN3,
00112 28* AN5,ANM,VB,HM1,HM,B1,VIE,VIB,VIM,RC2,PO2,RKC,RC1,
00112 29* RCD,VRC,RSN,OVA,BFN,DOB,AQM,P10,OSV,ROT,POD,POB,ARI
00113 30* COMMON/ARRAY/AR2,POC,AR3,ARM,CNR,GFN,AH7,AH8,AH,AHC,AH1,AH2,AH4,
00113 31* AHM,CNY,CNX,VV1,VV2,VV5,VV6,VV7,TVD,VTW,HSR,HSL,NID,
00113 32* SR,VVR,RAR,CV,CN7,AUX,AUK,AUZ,Y,CFC,CPK,PCE,CPR
00114 33* COMMON/ARRAY/LPK,DDO,HYL,KID,AMT,ANT,POK,PON,A1K,A2K,A3K,CNR,CNZ,
00114 34* AHK,SRK,V9,V2D,Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8,HMK,
00114 35* HKM,PQV,PQZ,RDO,QD2,RBE,MQ2,POA,POY,ANU,POR,GF2,HMD
00115 36* COMMON/ARRAY/DHM,POQ,I3,U,VP1,T1,GF3,GF4,AUP,AUV,RV1,AUY,OUT,
00115 37* DSP,AHZ,AHY,OSA,PP1,CPN,POS,PLF,PPQ,PPN,RPD,RFI,DFE,
00115 38* VPF,PPR,PMC,PMS,PMP,HR,CPF,PCP,DA1,DLZ,DPY,DPZ,GPZ
00116 39* COMMON/ARRAY/NOZ,KCZ,VIZ,HPR,HPL,STH,ALO,EXC,OZM,RA2,RR2,SVO,AUL,
00116 40* VV9,O2A,Q1,EXE,ARF,QRF,RSM,BFM,AM,OV5,PVO,RMO,QOM,
00116 41* RMO,P20,MMO,PDO,PDE,AMM,A4K,POM,QMM,PM1,PM3,PM4,EX1
00117 42* COMMON/ARRAY/Q2,Q3,PM5,PK1,Z9,Z10,Z11,Z12,Z13,Z14,Z15,Z16,PK2,
00117 43* PK3,FIS,SIA,PAR,GBL,ANY,ANZ,ANX,ANV,ANW,ANR,AUQ,AUR,
00117 44* AUS,A378,H1,A2,A3,DUMMY(19),TITLE(400),DUMNY(40)
00120 45* COMMON/NUMERO/K,NO(2D),NTIMEC,UNITS,NZ,NTIMEP,NN,MAXNO,NTIME
00121 46* COMMON/STORE/NG1,NG2,NG3,NG4,NG5,NG6,NG7,NG8,NG9,DT,TLF,TNP,ND,
00121 47* TM,TMM,NFIRST,ZZ(15),OLY(9),OBY(9),YMIN(10),YMAX(10),
00121 48* N,PT(18),BETA(10),NGRAPH(10),GRAPH(10),HEAD(19),NOEXP
00121 49* DTMAX
00121 50* C
00121 51* C CIRCULATORY DYNAMICS BLOCK
00121 52* C HEMODYNAMICS
00121 53* C
00122 54* VBD=VP+VRC-VVS-VAS-VLA-VPA-VRA
00123 55* VVS=VVS+DVS*I2+VBD*.3986
00124 56* VPA=VPA+DPA*I2+VBD*.155
00125 57* VAS=VAS+DAS*I2+VBD*.261

```



```

00126 58* VLA=VLA+DLA*I2+VBD*.128
00127 59* VRA=VRA+DRA*I2+VBD*.0574
00130 60* VAE=VAS-.495
00131 61* PA=VAE/.00355
00132 62* IF (PA.LT.0.) PAF=.0001
00134 63* PAM=100./PA
00135 64* PA2=PA/AUH
00136 65* CALL FUNCTN(PA2,LVM,FUN1)
00137 66* VRE=VRA-.1
00140 67* PRA=VRE/.005
00141 68* CALL FUNCTN(PRA,QRN,FUN2)
00142 69* VPE=VPA-.30625
00143 70* PPA=VPE/.0048
00144 71* PPI=.026*PPA
00145 72* IF (PPI.LT.0.) PPI=10.**(-12)
00147 73* RPA=PPI*(-.5)
00150 74* PR2=PRA/AUH
00151 75* IF (PR2.LE.0.) PR2=.0001
00153 76* CALL FUNCTN(PR2,RVM,FUN3)
00154 77* VLE=VLA-.4
00155 78* PLA=VLE/.01
00156 79* CALL FUNCTN(PLA,QLN,FUN4)
00157 80* RPV=1./ (PLA+20.)/.0357
00160 81* RPT=RPV+RPA
00161 82* PGL=PPA-PLA
00162 83* QPO=PGL/RPT
00163 84* ANU=ANM
00164 85* IF (ANU.LT..8) ANU=.8
00166 86* VVE=VVS-VVR-(ANU-1.)*ANY
00167 87* VV8=VVE-VV7
00170 88* IF (VV8.LT..0001) VV8=.0001
00172 89* PVS=VV8/CV
00173 90* PRI=ERA
00174 91* IF (PRA.LT.0.) PRI=0.
00176 92* RVG=2.738/PVS
00177 93* QVO=(PVS-PRI)/RVG
00200 94* CN3=CN3+((IPC-17.)*CN7+17.)*CN2-CN3)*.1
00201 95* AVE=(AUM-1.)*AUY+1.
00202 96* RVS=AVE*(1./CN3)*VIM*(ANU-1.)*ANZ+1.
00203 97* PGS=PA-PVS
00204 98* RSN=RAR*ARM*ANU*AUM*PAM*VIM+RVS*1.79
00205 99* BFN=PGS/RSN
00206 100* RSM=ANU*VIM*PAM*AUM*AMM*RAM
00207 101* BFM=PGS/RSM
00210 102* QAO=BFN+BEM+RBF*(EA=ERA)*FIS
00211 103* QLO=LVM*QLN*AUH*HSL*HMD*HPL
00212 104* QRO=QRN*((1.-QRF)*AUH+RVM*HSR*HMD*HPR+QRF*QLO/QLN)
00213 105* QPO=QLO+(QPO-QLO)/U
00214 106* QVO=QRO+(QVO-QRO)/X
00215 107* DVS=QAO-QVO
00216 108* DPA=QRO-QPO
00217 109* DAS=QLO-QAO
00220 110* QLA=QPO-QLO
00221 111* DRA=QVO-QRO
00222 112* RETURN
00223 113* END

```

```

00101 1* SUBROUTINE HORMON(A,AMC,AMP,AMR,AMT,AMI,ANM,CKE,PA,Z,FUN7,
00101 2* AGK,ANC,ANP,ANR,ANT,ANW,ANI,CNA,CNE,GFN,
00101 3* I,REK,A2,A3)
00103 4* DIMENSION FUN7(14)
00104 5* REAL I
00104 6* C
00104 7* C*****
00104 8* C
00104 9* C ALDOSTERONE CONTROL BLOCK
00104 10* C
00104 11* C*****
00105 12* 168 AMR=CKE/CNA/A3-A2
00106 13* IF (AMR.LT.0.)AMR=0.
00110 14* CALL FUNCTN (PA,AMP,FUN7)
00111 15* AMI=AMI+(ANM*AMP*AMR-AMI)/Z
00112 16* AMC=AMC+(AMI-AMC)*(1.-EXP(-I/AMT))
00113 17* AM=20.039-19.8*EXP(-.0391*AMC)
00113 18* C*****
00113 19* C
00113 20* C ANGIOTENSIN CONTROL BLOCK
00113 21* C
00113 22* C*****
00114 23* CNE=152.-CNA
00115 24* IF (CNE.LT.1.)CNE=1.
00117 25* ANR=(117.75-GEN*(CNA))*AGK+1.*REK
00120 26* ANW=ANW+((ANR-1.)*10.-ANW)*ANV*I
00121 27* IF (ANW.LT.0.)ANW=0.
00123 28* ANP=ANR*ANW
00124 29* IF (ANP.GT.100.)ANP=100.
00126 30* IF (ANP.LT.0.)ANP=0.
00130 31* ANI=ANI+(ANP-ANI)/Z
00131 32* ANC=ANC+(ANI-ANC)*(1.-EXP(-I/ANT))
00132 33* ANM=4.0-3.3*EXP(-.0967*ANC)

00133 34* IF (ANM.LT.7)ANM=7
00135 35* RETURN
00136 36* END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101 1* SUBROUTINE IONS (AM, CCD, CKE, CKI, CNA, I, KCD, KE, KED, KI, KID, KIE,
00101 2* * KIR, KOD, NAE, REK, VEC, VIC, VID, VP, VPF, VTS, Z)
00103 3* REAL I, KCD, KE, KED, KI, KID, KIE, KIR, KOD, NAE
00103 4* C
00103 5* C ELECTROLYTES AND CELL WATER BLOCK
00103 6* C
00104 7* 160 VEC=VTS+VP+VPF
00105 8* CKE=KE/VEC
00106 9* KOD=(.00042*CKE+.00014*AM*CKE)*REK
00107 10* KIR=2850.+140.*CKE
00110 11* KIE=KIR*KI
00111 12* KCD=KCD+(KIE*.013-KCD)/Z
00112 13* KI=KI+KCD*I
00113 14* KED=KID-KCD-KOD
00114 15* KE=KE+KED*I
00115 16* CKI=KI/VIC
00116 17* CNA=NAE/VEC
00117 18* CCD=CKI-CNA
00120 19* VID=VID+(.01*CCD-VID)/Z
00121 20* VIC=VIC+VID*I
00122 21* RETURN
00123 22* END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101 1* SUBROUTINE KIDNEY(AAR,AHM,AM,APD,ARF,AUM,CNE,CNX,CNY,GBL,GFN,GFR,
00101 2* GF2,GF3,GF4,GLP,I,NAE,NED,NID,NOD,NOZ,PA,PAR,
00101 3* PFL,PPC,RBF,REK,RFN,RR,STH,TRR,VIM,VUD,Z)
00103 4* REAL I,NAE,NED,NID,NOD,NOZ
00103 5* C
00103 6* C KIDNEY DYNAMICS AND EXCRETION BLOCK
00103 7* C
00104 8* 142 GF3=((GFN/.125-1.)*GF4)+1.
00105 9* IF(GF3.GT.15.)GF3=15.
00107 10* IF(GF3.LT..4)GF3=.4
00111 11* AAR=31.67*VIM*(AUM*ARF+1.-ARF)*GF3
00112 12* RR=AAR+51.66*VIM
00113 13* PAR=PA-GBL
00114 14* RFN=PAR/RR
00115 15* RBF=REK*RFN
00116 16* 150 APD=AAR*RFN
00117 17* GLP=PAR-APD
00120 18* PFL=GLP-PPC-18.
00121 19* GF1=GFN
00122 20* GFN=GFN+(PFL*.00781-GFN)*GF2/Z
00123 21* IF (ABS(GFN-GF1).GT..002)GO TO 142
00125 22* GER=GFN*REK
00126 23* TRR=.8*GFR+.025*REK-.001*REK/AM/AHM
00127 24* VUD=VUD+(GFR-TRR-VUD)/Z
00130 25* IF(VUD.LT..0002)VUD=.0002
00130 26* C-----
00130 27* C KIDNEY SALT OUTPUT AND SALT INTAKE
00130 28* C (SEE ALSO ELECTROLYTES AND CELL WATER BLOCK)
00130 29* C-----
00132 30* NOZ=1000.*VUD/AM/(CNE/CNX+CNY)
00133 31* NOD=NOD+(NOZ-NOD)/Z
00134 32* NED=NID*STH-NOD
00135 33* NAE=NAE+NED*I
00136 34* RETURN
00137 35* END

```

```

00101 1* SUBROUTINE MISCL (AHM,AU4,AU8,I ,SR ,SRK,STH,TVD,TVZ,VEC,VIC,VTW,
00101 2* * VVE,VV6,VV7,Z ,V9)
00103 3* REAL I
00103 4* C
00103 5* C*****
00103 6* C
00103 7* C VASCULAR STRESS RELAXATION BLOCK
00103 8* C
00103 9* C*****
00104 10* VV6=VV6+(SR*(VVE-V9 )-VV7-VV6)/Z
00105 11* VV7=VV7+VV6*(1.-EXP(-L/SRK))
00105 12* C*****
00105 13* C
00105 14* C THIRST AND DRINKING BLOCK
00105 15* C*****
00105 16* C
00106 17* TVZ=(.01*AHM-.009)*STH
00107 18* TVD=TVD+(TVZ-TVD)/Z
00110 19* IF(TVD.LT.0.)TVZ=0.
00112 20* VTW=VIC+VEC
00112 21* C*****
00112 22* C
00112 23* C .AUTONOMIC CONTROL BLOCK
00112 24* C ADAPTATION OF BARORECEPTORS
00112 25* C
00112 26* C*****
00113 27* AU4=AU4+AU8*I
00114 28* RETURN
00115 29* END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101      1*      SUBROUTINE MISC2 (HPL,HPR,HSL,HSR,I,PA,PPA,POT,STH,Z10,Z11,Z13)
00103      2*      REAL I
00103      3*      C
00103      4*      C*****
00103      5*      C
00103      6*      C      HEART HYPERTROPHY OR DETERIORATION BLOCK
00103      7*      C
00103      8*      C*****
00104      9*      HPL=HPL+(((PA/100./HSL)**Z13)-HPL)*1/57600.
00105     10*      HPR=HPR+(((PPA/15./HSR)**Z13)-HPR)*1/57600.
00105     11*      C*****
00105     12*      C
00105     13*      C      TISSUE EFFECT ON THIRST AND SALT INTAKE
00105     14*      C
00105     15*      C*****
00106     16*      STH=(Z10-POT)*Z11
00107     17*      IF(STH.LT.1.)STH=1.
00111     18*      IF(STH.GT.8.)STH=8.
00113     19*      RETURN
00114     20*      END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101 1* SUBROUTINE MUSCLE(ALO,AMM,ADM,AUP,ASK,BFM,EXC,HM,I,MMO,OMM,OSA,
00101 2* OVA,OVS,OZA,PDO,PK1,PK2,PK3,PMO,PM1,PM3,PM4,PM5,
00101 3* POE,POM,PVO,PZO,QOM,RMO,VFF,Z5,Z6)
00103 4* REAL I,MMO
00103 5* C
00103 6* C MUSCLE BLOOD FLOW CONTROL AND P02 BLOCK
00103 7* C
00104 8* 180 OSA=ALO-VFF*.5
00105 9* OVA=OSA*HM*.5.
00106 10* OVS=OVS+((BFM*OVA-RMO)/HM/5./BFM-OVS)/Z6
00107 11* PVO=57.14*OVS
00110 12* RMO=(PVO-PMO)*PM5/(PM1**PK3-PM4)
00111 13* QOM=QOM+(RMO-MMO)*(1.-EXP(-I/Z5))
00112 14* PMO=PK2/(PK1-QOM)
00113 15* PM1=PMO
00114 16* IF(PM1.LT.PM3)PM1=PM3
00116 17* PZO=PMO
00117 18* IF(PZO.GT.8.)PZO=8.
00121 19* ADM=(AUP-1.)*OZA+1.
00122 20* MMO=ADM*OMM*EXC*(1.-(8.0001-PZO)**3./512.)
00123 21* PDO=PVO-40.
00124 22* POE=POM*PDO+1.
00125 23* IF(POE.LT..005)POE=.005
00127 24* AMM=AMM+(POE-AMM)*(1.-EXP(-I/A4K))
00130 25* RETURN
00131 26* END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101 1* SUBROUTINE PAGE3
00101 2* C THIS ROUTINE ASKS FOR HARD COPY UNTILL N IS ENTERED.
00101 3* C THE PAGE IS THEN ERASED AND CURSOR MOVED TO HOME.
00103 4* DATA INO/'N '/,IYO/'Y '/'
00106 5* 5 CALL GOTOCH(34,1)
00106 6* C
00107 7* CALL BELL
00110 8* WRITE(6,11)
00112 9* 11 FORMAT(' HARD COPY WANTED(Y,N) . . .')
00113 10* 8 READ(5,12,ERR=8) I
00116 11* 12 FORMAT(A4)
00117 12* IF(I .EQ. INO) GO TO 25
00121 13* IF(I .NE. IYO) GO TO 5
00123 14* CALL HD COPY
00124 15* GO TO 5
00124 16* C
00125 17* 25 CALL NEWPAG
00126 18* RETURN
00127 19* END

```

END OF COMPILATION: NO DIAGNOSTICS.



```

00101 1* SUBROUTINE PLOT33
00103 2* COMMON/NUMERO/K,NO(20)
00104 3* COMMON/PPPPP/PLOTBF(101,6),RUNSTP,IEXECI,IPLTPT
00104 4* 1 TLOTBE(101),KSTOPP
00105 5* DIMENSION TSTEPX(4),TSTALP(4)
00106 6* DATA TSTEPX/.01666667,1.26D.0,1440./
00110 7* DATA TSTALP/'SECS','MINS','HOUR','DAYS'/
00112 8* COMMON/ARRAY/A(400),TITLE(400),COL(20),ALPHA(20)
00113 9* COMMON/STORE/JUNK(118),HEAD(19),NOEXP
00114 10* DIMENSION IIDAS(6,2),PPARS(6,3)
00115 11* DIMENSION XNO(5)
00116 12* DATA ISA/'S'/'IN/IN'/'Y/Y'/'IPL/PLT'/
00123 13* IS = IPLTPT - 2
00124 14* IF(IPLTPT.LE.2) GO TO 1
00126 15* IF(IEXECI.EQ.IPLL.AND.KSTOPP.EQ.0)
00126 16* GO TO 28
00130 17* IF(IEXECI.EQ.IPLL.AND.KSTOPP.EQ.1)
00130 18* GO TO 120
00132 19* 1 IS = 1
00133 20* 11 WRITE(6,2)
00135 21* 2 FORMAT(' GRAPH OUTPUT WANTED(Y,N,S),TIME INTERVALS(A4,F2.0'
00135 22* ' )')
00136 23* 4 READ(5,3,ERR=4) I,TSPLIT
00142 24* 3 EFORMAT(A4,F2.0)
00143 25* IF(I.EQ.ISA) GO TO 10
00145 26* IF(I.EQ.IN) RETURN
00147 27* IF(I.NE.IY) GO TO 11
00151 28* LE(TSPLIT,LT,3,OR,TSPLIT,GT,5) GO TO 11
00151 29* C HERE TO BUILD PLOT PARAMETERS.
00153 30* TISPT = TSPLIT
00154 31* NOP = 0
00155 32* DO 20 I = 1,K
00160 33* 17 WRITE(6,5) ALPHA(I)
00163 34* 5 FORMAT(' ',A4,13X,'Y SCALE (A4,8X,F4.0,2F6.0)')
00163 35* 1 ' PLOT(Y,N,S) LOC HIGH LOW ...')
00164 36* 7 READ(5,6,ERR=7) I1,XP1,XP2,XP3
00172 37* 6 FORMAT(A4,8X,F4.0,2F6.0)
00173 38* IF(I1.EQ.IY) GO TO 18
00175 39* IF(I1.EQ.ISA) GO TO 19
00177 40* IF(I1.NE.IN) GO TO 17
00201 41* PPARS(1,1) = 0.
00202 42* GO TO 20
00203 43* 18 IF(XP1.LE.0) GO TO 17
00205 44* IF(IFIX(XP1).GT.6) GO TO 17
00207 45* IF(XP3.GE.XP2) GO TO 17
00211 46* PPARS(1,1) = XP1
00212 47* PPARS(1,2) = XP2
00213 48* PPARS(1,3) = XP3

```

ORIGINAL PAGE IS  
 OF POOR QUALITY

```

00214 49*      19 IF(IFIX(PPARS(I,1)) .GT. NOP) NOP = PPARS(I,1)
00216 50*      IF(PPARS(I,1) .LT. 1.) GO TO 17
00220 51*      20 CONTINUE
00220 52*      C BUILD WHOLE PAGE GRAPH.
00222 53*      10 CALL NEWPAG
00223 54*      CALL MOVABS(0,780)
00224 55*      CALL DMPBUF
00225 56*      WRITE(6,21) NOEXP,(HEAD(J),J=1,15)
00234 57*      21 FORMAT(' ',14,2X,15A4)
00235 58*      CALL MOVABS(0,100)
00236 59*      CALL DRWABS(1000,100)
00237 60*      CALL DRWABS(1000,750)
00240 61*      CALL DRWABS(0,750)
00241 62*      CALL MOVABS(300,750)
00242 63*      CALL DRWABS(300,100)
00243 64*      CALL DMPBUF
00244 65*      11 = TTSRT = 1.
00245 66*      12 = 300
00246 67*      13 = 700 / (11 + 1)
00247 68*      DO 25 I = 1,11
00252 69*      12 = 12 + 13
00253 70*      CALL MOVABS(12,100)
00254 71*      CALL DRWABS(12,750)
00255 72*      25 CONTINUE
00257 73*      CALL DMPBUF
00257 74*      C DRAW DIVISIONS FOR DIFFERENT GRAPHS.
00260 75*      13Y = 650 / NOP
00261 76*      14Y = 13Y / 2
00262 77*      11 = NOP - 1
00263 78*      12 = 100
00264 79*      IF(11 .LT. 1) GO TO 280
00266 80*      DO 26 I = 1,11
00271 81*      14 = 12 + 14Y
00272 82*      CALL MOVABS(300,14)
00273 83*      CALL DRWABS(305,14)
00274 84*      CALL MOVABS(995,14)
00275 85*      CALL DRWABS(1000,14)
00276 86*      12 = 12 + 13Y
00277 87*      CALL MOVABS(0,12)
00300 88*      CALL DRWABS(1000,12)
00301 89*      CALL DMPBUF
00302 90*      26 CONTINUE
00304 91*      280 14 = 12 + 14Y
00305 92*      CALL MOVABS(300,14)
00306 93*      CALL DRWABS(305,14)
00307 94*      CALL MOVABS(995,14)
00310 95*      CALL DRWABS(1000,14)
00311 96*      CALL DMPBUF
00311 97*      C ADD ALPHA.
00312 98*      11 = 750
00313 99*      10UP = 1
00314 100*      CALL VWINDO(0.,1023.,0.,780.)
00315 101*      CALL SWINDO(0.,1023.,0.,780)
00316 102*      DO 70 I = 1,NOP
00321 103*      111 = 11
00322 104*      122 = 11 - 13Y + 30
00323 105*      10 = 0

```

```

00324 106*      DO 60 J = 1,K
00327 107*      IF(IFIX(PPARS(J,1)) .NE. 1) GO TO 60
00331 108*      CALL MOVABS(0,I11)
00332 109*      CALL ANMODE
00333 110*      CALL DMPBUF
00334 111*      WRITE(6,55) ALPHA(J),PPARS(J,2)
00340 112*      55 FORMAT(' ',8X,A4,F8.2)
00341 113*      Y1 = I11 - 11.
00342 114*      CALL MOVEA(10.,Y1)
00343 115*      IF(ID .EQ. 0) GO TO 410
00345 116*      ID1 = 0
00346 117*      ID2 = 0
00347 118*      CALL DSHARC(104.,Y1, ID, ID1, ID2, IDUP)
00350 119*      GO TO 411
00351 120*      410 CALL DRAWA(104.,Y1)
00352 121*      411 CALL MOVABS(0,I22)
00353 122*      CALL ANMODE
00354 123*      CALL DMPBUF
00355 124*      WRITE(6,55) ALPHA(J),PPARS(J,3)
00361 125*      Y1 = I22 - 13.
00362 126*      CALL MOVEA(10.,Y1)
00363 127*      IF(ID .EQ. 0) GO TO 413
00365 128*      ID1 = 0
00366 129*      ID2 = 0
00367 130*      CALL DSHARC(104.,Y1, ID, ID1, ID2, IDUP)
00370 131*      GO TO 415
00371 132*      413 CALL DRAWA(104.,Y1)
00372 133*      415 ID = ID + 1
00373 134*      I11 = I11 + 21
00374 135*      I22 = I22 + 21
00375 136*      60 CONTINUE
00377 137*      I1 = I1 - 13Y
00400 138*      70 CONTINUE
00402 139*      X = (RUNSTP/TTSPT) + .000001
00403 140*      I = 2
00404 141*      IF(X .LT. 1.) I = 1
00406 142*      IF(X .GE. 6D.) I = 3
00410 143*      IF(X .GE. 144D.) I = 4
00412 144*      X = X / ISTEPX(I)
00413 145*      X2 = TLOTBF(I) / TSTEPX(I)
00414 146*      X1 = X2 + X
00415 147*      I1 = TTSP
00416 148*      DO 75 J = 1,I1
00421 149*      XNO(J) = X1
00422 150*      X1 = X1 + X
00423 151*      75 CONTINUE
00425 152*      CALL MOVABS(0,I00)
00426 153*      CALL ANMODE
00427 154*      CALL DMPBUF
00430 155*      IF(TTSPT .GT. 4.) GO TO 80
00432 156*      IF(TTSPT .GT. 3.) GO TO 79
00434 157*      WRITE(6,93) X2,(XNO(J),J=1,3)
00443 158*      93 FORMAT(' ',15X,F6.1,11X,F6.1,11X,F6.1,10X,F6.1)
00444 159*      GO TO 85
00445 160*      79 WRITE(6,77) X2,(XNO(J),J=1,4)
00454 161*      77 FORMAT(' ',15X,F6.1,6X,F6.1,7X,F6.1,6X,F6.1,7X,F6.1)
00455 162*      GO TO 85

```

```

00456 163*      80 WRITE(6,81) X2,(XNO(J),J=1,5)
00465 164*      81 FORMAT(' ',11X,6(4X,F6,1))
00466 165*      85 WRITE(6,86) TSTALP(I)
00471 166*      86 FORMAT(' ',30X,'TIME (' ,A4,')')
00471 167*      C INITIALIZE DASH INFORMATION.
00472 168*          DO 90 J = 1,6
00475 169*          IIDAS(J,1) = 0
00476 170*          90 CONTINUE
00500 171*          IF(IEXEC1 .EQ. IPLL) RETURN
00500 172*      C PLOT VARIABLES...
00502 173*          28 IL = 750
00503 174*          IS2 = IS + 1
00504 175*          I2 = IPLTPT - 1
00505 176*          DO 40 I = 1,NOP
00510 177*          IL = IL - 13Y
00511 178*          IDAS = -1
00512 179*          IDUP = 1
00513 180*          DO 30 II = 1,K
00516 181*          IF(LIEIX(PPARS(II,1)) .NE. 1) GO TO 30
00520 182*          IDAS = IDAS + 1
00521 183*          X = TLOTBF(1)
00522 184*          XL = RUNSTP
00523 185*          Y = PPARS(II,3)
00524 186*          YL = PPARS(II,2) - Y
00525 187*          CALL VWINDO(X,XL,Y,YL)
00526 188*          CALL SWINDO(300,700,IL,13Y)
00527 189*          X = TLOTBF(IS)
00530 190*          Y = PLOTBF(IS,II)
00531 191*          IF(IEXEC1 .NE. IPLL) GO TO 97
00533 192*          IF(Y .LE. PPARS(II,2)) GO TO 105
00535 193*          IF(PLOTBF(IS2,II) .GE. PPARS(II,2)) GO TO 30
00537 194*          YY = PPARS(II,2)
00540 195*          GO TO 96
00541 196*          105 IF(Y .GE. PPARS(II,3)) GO TO 97
00543 197*          IF(PLOTBF(IS2,II) .LT. PPARS(II,3)) GO TO 30
00545 198*          YY = PPARS(II,3)
00546 199*          96 CALL MOVEA(X,YY)
00547 200*          IDUP = IDUP + 1
00550 201*          97 IDUP = IDUP + 1
00551 202*          CALL MOVEA(X,Y)
00552 203*          DO 35 III = IS2,I2
00555 204*          X = TLOTBF(III)
00556 205*          Y = PLOTBF(III,II)
00557 206*          IF(IDAS .GT. 0) GO TO 33
00561 207*          IDUP = IDUP + 1
00562 208*          CALL DRAWA(X,Y)
00563 209*          GO TO 34
00564 210*          33 L = IIDAS(II,1)
00565 211*          N = IIDAS(II,2)
00566 212*          CALL DSHARC(X,Y,IDAS,L,N,IDUP)
00567 213*          IIDAS(II,1) = L
00570 214*          IIDAS(II,2) = N
00571 215*          34 IF(IDUP .LT. 13) GO TO 35
00573 216*          CALL DMPBUF
00574 217*          IDUP = 1
00575 218*          35 CONTINUE
00577 219*          30 CONTINUE

```

```
00601 220* 40 CONTINUE
00603 221* IF(IDUP .GT. 1) CALL DMPBUF
00605 222* IF(KSTOPP .EQ. 0) RETURN
00607 223* 120 CALL PAGE3
00610 224* IEXECI = 0
00611 225* GO TO 1
00612 226* END
```

END OF COMPILATION: NO DIAGNOSTICS.

ORIGINAL PAGE IS  
OF POOR QUALITY

```

00101      1*      SUBROUTINE PROTEN(CHY,CPG,CPI,CPK,CPP,CPR,CPI,DLP,DLZ,DPC,DPI,DPL,
00101      2*      *      DPO,DPY,GPD,GPR,I ,IFP,LPK,PC ,PCE,PGX,PRP,VG
00101      3*      *      VTL,Z ,PPD)
00103      4*      REAL I,IFP,LPK
00103      5*      C
00103      6*      C      TISSUE FLUIDS,PRESSURES AND GEL BLOCK
00103      7*      C
00103      8*      C-----
00103      9*      C      PLASMA AND TISSUE FLUID PROTEIN
00103     10*      C-----
00104     11*      135 DPL=DPL+(VTL*CPI-DPL)/Z
00105     12*      IF (PC.LT.0.)PC=0.
00107     13*      DPC=DPC+(CPK*(CPP-CPI)*PC**PCE-DPC)/Z
00110     14*      DPI=DPC-DPL
00111     15*      DLZ=LPK*(CPR-CPP)
00112     16*      IF (CPP.GT.CPR)DLZ=4.*DLZ
00114     17*      DLP=DLP+(DLZ-DLP)/Z
00115     18*      PRP=PRP+(DLP-DPO+DPL-DPC=PPD)*I
00115     19*      C-----
00115     20*      C      GEL PROTEIN DYNAMICS
00115     21*      C-----
00116     22*      141 PGX=CHY**2*.D1332*CPG+CPG
00117     23*      GPD=GPD+(.0005*(CPI-PGX)*VG-GPD)/Z
00120     24*      GPR=GPR+GPD*I
00121     25*      IFP=IFP+(DPI-GPD)*I
00122     26*      RETURN
00123     27*      END

```

END OF COMPILATION; NO DIAGNOSTICS.

```

00101 1* SUBROUTINE PULMON(CPF, CPP, CPN, DEP, L, PCP, PFI, PLA, PLF, POS, PPA, PPC,
00101 2* * PPD, PPI, PPN, PPO, PPR, VP, VPD, VPF, Z, Z3)
00103 3* REAL I
00103 4* C
00103 5* C PULMONARY DYNAMICS AND FLUIDS BLOCK
00103 6* C
00104 7* VP=VP+(VPD*I)/Z3
00104 8* C
00105 9* 200 PCP=.45*PPA+.55*PLA
00106 10* PPI=2.-.150/VPF
00107 11* CPN=PPR/VPF
00110 12* POS=CPN*.4
00111 13* PLF=(PPI+1.)*.0003
00112 14* PPO=PLF*CPN
00113 15* PPN=(CPP-CPN)*.000225
00114 16* PPD=PPD+(PPN-PPO-PPD)/Z
00115 17* IF (PPR+PPD*I-.025.LT.0.) PPD=(.025-PPR)/I
00117 18* PFI=(PCP-PPI+POS-PPC)*CPF
00120 19* DFP=DFP+(PFI-PLF-DEP)/Z
00121 20* IF (VPF+DFP*I-.001.LT.0.) DFP=(.001-VPF)/I
00123 21* VPF=VPF+DFP*I
00124 22* PPR=PPR+PPD*I
00125 23* RETURN
00126 24* END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101 1* SUBROUTINE PUTIN
00101 2* C
00103 3* COMMON/ARRAY/A(400),TITLE(400),COL(20),ALPHA(20)
00104 4* COMMON/NUMERO/K,NO(20),NTIMEC,UNITS,NZ,NTIMEP,NN,MAXNO,NTIME

```

```

00105 5* COMMON/STORE/NG1,NG2,NG3,NG4,NG5,NG6,NG7,NG8,NG9,DT,TL,LP,INP,ND,
00105 6* TM,TMM,NFIRST,ZZ(15),OLY(9),OBY(9),YMIN(10),YMAX(10),
00105 7* N,PT(18),BEJA(10),NGRAPH(10),GRAPH(10),HEAD(19),NOEXP
00105 8* ,DTMAX
00106 9* COMMON/DEMAND/LTAPEO,ITME,LEXECN,ICONVI
00107 10* DATA INOOD/'N '/'
00111 11* DATA ALL/'ALL','BLANK/' '%,SAME/'!SAME!/'
00115 12* DO 1 J=1,400
00120 13* A(J)=0.
00121 14* 1 TITLE(J)=BLANK
00123 15* NZ=0
00124 16* NTIMEP=1
00125 17* NN=1
00126 18* MAXNO=1
00127 19* MIXX = 1
00130 20* 2 READ(5,100) VALUE,NUMBRO,SYMBOL
00135 21* 100 FORMAT (F13.6,2X,15,2X,A4)
00136 22* IF(MAXNO.LT.NUMBRO) MAXNO=NUMBRO
00140 23* IF(NUMBRO.EQ.0) GO TO 3
00142 24* A(NUMBRO)=VALUE
00143 25* TITLE(NUMBRO)=SYMBOL
00144 26* MIXX = MIXX + 1
00145 27* GO TO 2
00146 28* 3 DSP=A(286)
00147 29* WRITE(6,731) MIXX
00152 30* 731 FORMAT('0',I6,'DATA RECORDS INPUT.')
00153 31* 579 WRITE(6,580)
00155 32* 580 FORMAT(' INPUT NO. AND NAME FOR EXPERIMENT (I4,15A4)...')
00156 33* READ(5,101,ERR=579) NOEXP,(HEAD(J),J=1,15)
00165 34* 101 FORMAT(I4,15A4)
00165 35* C GET OUT IF CONVERSATIONAL MODE.
00166 36* IF(ICONVI.NE. INOOD) GO TO 31
00170 37* IF(DSP) 37,38,37
00173 38* 37 NFIRST=0.
00174 39* CALL DSPLAY.
00174 40* C IF(N=10) 201,38,201
00175 41* 38 READ(5,200) (ALPHA(J),J=1,20)
00203 42* 200 FORMAT (20A4)
00204 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00204 43* 201 IF(ALPHA(1).EQ.SAME) GO TO 32
00206 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00206 44* IF(ALPHA(1).NE.ALL) GO TO 4
00210 45* READ(5,300) NTIMEC,UNITS
00214 46* WRITE(6,102) NOEXP,(HEAD(J),J=1,18)
00223 47* WRITE(6,711) UNITS,(TITLE(J),A(J),J=1,MAXNO)
00233 48* GO TO 31
00234 49* 4 DO 5 K=1,20
00237 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00237 50* IF(ALPHA(K).EQ.BLANK) GO TO 6
00241 51* 5 CONTINUE
00243 52* KR21
00244 53* 6 K=K-1
00245 54* DO 10 J=1,K
00250 55* L=1
00251 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00251 56* 7 IF(ALPHA(J).EQ.TITLE(L)) GO TO 9
00253 57* L=L+1

```



```

00254 58*      IF(L.LT.MAXNO+1) GO TO 7
00254 59*      C      WRITE(2,530) ALPHA(J)
00254 60*      C 530 FORMAT(/' THE VARIABLE ',A4,' IS NOT AVAILABLE TO THE PRINTER,!'
00254 61*      C      *      ' CHOOSE ANOTHER ONE TO TAKE ITS PLACE,!'
00254 62*      C      *      ' ABC ')
00254 63*      C      READ(2,531) ALPHA(J)
00254 64*      C 531 FORMAT(A4)
00254 65*      C      L=L+1
00254 66*      C      GO TO 7
00256 67*      9 COL(J)=A(L)
00257 68*      NO(J)=L
00260 69*      10 CONTINUE
00262 70*      GO TO 34
00263 71*      32 DO 33 J=1,K
00266 72*      ALPHA(J)=BETA(J)
00267 73*      NO(J)=NGRAPH(J)
00270 74*      33 COL(J)=GRAPH(J)
00272 75*      34 READ(5,300) NTIMEC,UNITS
00276 76*      300 FORMAT(16,A4)
00277 77*      WRITE(6,102) NOEXP,(HEAD(J),J=1,18)
00306 78*      102 FORMAT(1H1,2X,'EXP ',14Z',',18A4///)
00307 79*      IF(1TAPEO.EQ.INOOO) GO TO 13
00311 80*      CALL NTRAN(7,1,400,A,J)
00312 81*      1234 IF (J.LT.O) GO TO 1234
00314 82*      13 IF(K.GT.10) GO TO 70
00316 83*      WRITE(6,21) UNITS,(ALPHA(J),J=1,K)
00325 84*      21 FORMAT(10,'A4,10(6X,A4,1X))
00326 85*      WRITE(6,22) (COL(J),J=1,K)
00334 86*      22 FORMAT(' ',5X,'0',2X,F10.4,9(1X,F10.4))
00335 87*      GO TO 31
00336 88*      70 WRITE(6,71) UNITS,(ALPHA(J),COL(J),J=1,K)
00346 89*      71 FORMAT(60X,2H0 ,A4//5(4X,A4,' = 'F10.4,4X))
00347 90*      31 RETURN
00350 91*      END

```

END OF COMPILATION; 4 DIAGNOSTICS.

```

00101 1* SUBROUTINE PUTOUT
00101 2* C
00103 3* COMMON/ARRAY/A(400),TITLE(400),COL(20),ALPHA(20)
00104 4* COMMON/NUMERO/K,NO(20),
00104 5* NTIMEC,UNITS,NZ,NTIMEP,NN,MAXNO,NTIME
00105 6* COMMON/STORE/NG1,NG2,NG3,NG4,NG5,NG6,NG7,NG8,NG9,DT,TLP,TNP,ND,
00105 7* TM,TMM,NEIRST,ZZ(15),OLY(9),OBY(9),YMIN(10),YMAX(10),
00105 8* N,PT(18),BETA(10),NGRAPH(10),GRAPH(10),HEAD(19),NOEXP
00105 9* DIMAX
00106 10* COMMON/TAPE/TOTAL
00107 11* COMMON/DEMAND/LTAPEO,ITME,IEXECN
00110 12* DATA IN000/'N '/'
00112 13* DATA SECS/'SECS'/,TMIN/'MINS'/,HOUR/'HOUR'/,DAYS/'DAYS'/
00117 14* DATA ALL/'ALL '/,BLANK/' '/
00117 15* C EQUIVALENCE(A(1),T)
00122 16* T=A(1)
00123 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00123 17* 1 IF(UNITS.EQ.SECS) GO TO 2
00125 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00125 18* IF(UNITS.EQ.TMIN) GO TO 3
00127 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00127 19* IF(UNITS.EQ.HOUR) GO TO 4
00131 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00131 20* IF(UNITS.EQ.DAYS) GO TO 5
00131 21* C WRITE(2,501) UNITS
00131 22* C 501 FORMAT(/' YOU CANNOT ASK FOR TIME UNITS OF ',A4,'.'/
00131 23* C * 44H TYPE @SECS@,@MINS@,@HOUR@, OR @DAYS@ BELOW /
00131 24* C * ' UNIT')
00131 25* C READ(2,500) UNITS
00131 26* C GO TO 1
00133 27* 2 NTIME=T*60
00134 28* IF(NTIME.LT.NTIMEP) GO TO 65
00136 29* IF(NTIME.LT.(NZ+1)*60) GO TO 6
00140 30* NZ=NZ+1
00141 31* GROSSU=TMIN
00142 32* GO TO 6
00143 33* 3 NTIME=T
00144 34* IF(NTIME.LT.NTIMEP) GO TO 65
00146 35* IF(NTIME.LT.(NZ+1)*60) GO TO 6
00150 36* NZ=NZ+1
00151 37* GROSSU=HOUR
00152 38* GO TO 6
00153 39* 4 NTIME=T/60
00154 40* IF(NTIME.LT.NTIMEP) GO TO 65
00156 41* IF(NTIME.LT.(NZ+1)*24) GO TO 6
00160 42* NZ=NZ+1
00161 43* GROSSU=DAYS
00162 44* GO TO 6
00163 45* 5 NTIME=T/1440
00164 46* IF(NTIME.LT.NTIMEP) GO TO 65
00166 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00166 47* 6 IF(ALPHA(1).NE.ALL) GO TO 7
00170 48* WRITE(6,71) NTIME,UNITS,(TITLE(J),A(1),J=1,MAXNO)
00201 49* GO TO 51
00202 50* 7 DO 20 I=1,K
00205 51* I1=NO(I)
00206 52* COL(I1)=A(I1)
00207 53* 20 CONTINUE
00211 54* IE(LTAPEO.EQ.IN000) GO TO 34

```

```

00213 55* CALL NTRAN(7,1,400,A,J)
00214 56* 1234 IF (J.LT.0) GO TO 1234
00216 57* 34 IF(K=10) 75,75,70
00221 58* 75 WRITE(6,31) NTIME,(COL(J),J=1,K)
00230 59* 31 FORMAT(' ',16,2X,E10.4,2(1X,F10.4))
00231 60* GO TO 51
00232 61* 70 WRITE(6,71) NTIME,UNITS,(ALPHA(J),COL(J),J=1,K)
00243 62* 71 FORMAT(///56X,15,1X,A4//5(4X,A4,' = 'F10.4,4X))
00244 63* 51 NTIME=NTIME+1
00245 64* IF(NZ.LT.NN) GO TO 53
00247 65* WRITE(6,52) NZ,GROSSU
00253 66* 52 FORMAT(14,1X,A4)
00254 67* NN=NN+1
00255 68* 53 IF(NTIME.LT.NTIMEC) GO TO 65
00257 69* 54 READ(5,400) NTIMEC,CUNITS,SYMBOL,CVALUE
00265 70* 400 FORMAT(16,A4,A4,E13.6)
00266 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00266 71* IF(SYMBOL.EQ.CUNITS) GO TO 66
00270 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00270 72* IF(CUNITS.NE.BLANK) GO TO 59
00270 73* C IF=.5
00272 74* IF(A(2).GT.=5) A(2)=.5
00272 75* C DT=DTMAX/5
00272 76* C TNP=T+DT
00274 77* 450 DO 55 MN=1,MAXNO
00277 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00277 78* IF(SYMBOL.EQ.TITLE(MN)) GO TO 57
00301 79* 55 CONTINUE
00301 80* C WRITE(2,56) SYMBOL
00301 81* C 56 FORMAT(' ' I DO NOT RECALL ',A4,' BEING READ IN WITH THE '
00301 82* C * ' INITIALIZING DECK. TRY AGAIN TO CHANGE ITS VALUE '
00301 83* C * ' BY TYPING BELOW AS FOLLOWS,'
00301 84* C * ' ABC = XXX,XXX'
00301 85* C READ(2,500) SYMBOL,CVALUE
00301 86* C 500 FORMAT(A4,1X,F8.3)
00301 87* C GO TO 450
00303 88* 57 WRITE(6,58) NTIME,UNITS,SYMBOL,A(MN),CVALUE
00312 89* 58 FORMAT(' ',A4,15,1X,A4,
00312 90* | 1X,A4,' CHANGED FROM ',F10.3,' TO ',F10.3)
00313 91* A(MN)=CVALUE
00314 92* GO TO 54
00315 93* 59 IF(K.GT.10) GO TO 82
00317 94* WRITE(6,86) CUNITS,(ALPHA(J),J=1,K)
00326 95* 86 FORMAT('D ',A4,10(6X,A4,1X))
00327 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00327 96* 82 IF(UNITS.EQ.CUNITS) GO TO 65
00331 97* IF(K.LE.10) GO TO 83
00333 98* WRITE(6,80) UNITS,CUNITS
00337 99* 60 FORMAT(' OUTPUT HAS BEEN CHANGED FROM ',A4,' TO '
00337 100* * A4,' ')
00340 101* 83 UNITS=CUNITS
00341 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00341 102* IF(UNITS.EQ.SECONDS) GO TO 61
00343 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00343 103* IF(UNITS.EQ.TMIN) GO TO 62
00345 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL
00345 104* IF(UNITS.EQ.HOUR) GO TO 63

```

```

00347 105* NTIMEP=T/1440.*+1.
00350 106* NZ=T/1440.*7.
00351 107* GO TO 64
00352 108* 61 NTIMEP=T*60.*+1.
00353 109* NZ=T
00354 110* GO TO 64
00355 111* 62 NTIMEP=T+1.
00356 112* NZ=T/60.
00357 113* GO TO 64
00360 114* 63 NTIMEP=T/60.*+1.
00361 115* NZ=T/1440.
00362 116* 64 NN=NZ+1
00363 117* 65 RETURN
00364 118* 66 IF(ITAPEO.EQ.INOOO) GO TO 35
00366 119* CALL NJRAN(7,9)
00367 120* 35 STOP
00370 121* END

```

END OF COMPILATION: 12 DIAGNOSTICS.

```

00101 1* SUBROUTINE SEGLD(INUTS)
00103 2* X=1.
00104 3* RETURN
00105 4* END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101 1* SUBROUTINE TIMPERT(T,IDI,IHI,IMI,ISI)
00101 2* C THIS ROUTINE COMPUTES TIME FOR PRINTOUT.
00101 3* C INPUT T (TIME IN MINS.)
00101 4* C OUTPUT IDI=DAYS,IHI=HOURS,IMI=MINS,ISI=SECS.
00103 5* IDI = T / 1440.
00104 6* XI = T - (FLOAT(IDI) * 1440.)
00105 7* IHI = XI / 60.
00106 8* XI = XI - (FLOAT(IHI) * 60.)
00107 9* IMI = XI
00110 10* ISI = (XI - FLOAT(IMI)) * 60.
00111 11* RETURN
00112 12* END

```

END OF COMPILATION: NO DIAGNOSTICS.

```

00101 1* SUBROUTINE TTYIN
00103 2* COMMON/ARRAY/A(400),TITLE(400),COL(20),ALPHA(20)
00104 3* COMMON/NUMERO/K,NO(20)
00104 4* I NTIMEC,UNITS,NZ,NTIMEP,NN,MAXNO,NTIME
00105 5* COMMON/TIYIOB/VCHGS(200,2),JTSTEP(100,5),PLDPT,PLOTM
00106 6* COMMON/PPPPP/RLOTBF(101,6),RUNSTP,IEXECI,IPLTPT
00106 7* I TLOTFE(101),KSTOPP
00107 8* COMMON/DEMAND/ITAPEO,ITME,IEXECN,ICONVI
00110 9* DATA STOP/'STOP',IPLL/'PLOT'/
00113 10* DATA BLANK/' ',DONE/'DONE'/
00116 11* DATA LISTEP/'STEP'/
00120 12* DATA SAMEE/'SAME'/
00122 13* DATA JMORE/'MORE',JRUN/'RUN'/
00125 14* DIMENSION ITSYM(4),ALPHA2(6)
00126 15* DATA ITSYM/'SECS','MINS','HOUR','DAYS'/
00130 16* DIMENSION TDELTA(4)
00131 17* DATA TDELTA/.01666667,1.60,1440./
00133 18* TDELTA(1) = 1. / 60.
00134 19* RUNSTP = 0.
00135 20* PLOTM = A(1)
00136 21* IPLTPT = 1
00136 22* C
00136 23* C
00137 24* MAXNO = 400
00140 25* CALL INITT(300)
00140 26* C
00140 27* C READ INPUT VARIABLES WANTED.
00141 28* 20 WRITE(6,21)
00143 29* 21 FORMAT(' OUTPUT WANTED OR SAME, STOP (A4) ')
00144 30* READ(5,22,ERR=20) (ALPHA2(J),J=1,6)
00152 31* 22 FORMAT(A4)
00153 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00153 32* IF(ALPHA2(1) .EQ. STOP) STOP
00155 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00155 33* IF(ALPHA2(1) .EQ. SAMEE .AND. ITME .NE. 1) GO TO 160
00157 34* K = 0
00160 35* DO 25 J = 1,6
00163 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00163 36* IF(ALPHA2(J) .EQ. BLANK) GO TO 27
00165 37* K = J
00166 38* 25 CONTINUE
00170 39* 27 IF(K .EQ. 0) GO TO 20
00172 40* IF(K .GT. 6) K = 6
00174 41* DO 35 I = 1,K
00177 42* DO 34 J = 1,MAXNO
00202 43* J1 = J
00203 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00203 44* IF(ALPHA2(I) .EQ. TITLE(J)) GO TO 31
00205 45* 34 CONTINUE
00205 46* C HERE IF AN OUTPUT SYMBOL BAD.
00207 47* WRITE(6,29) ALPHA2(I)
00212 48* 29 FORMAT(' SYMBOL ',A4,' WRONG, TRY AGAIN.')
00213 49* GO TO 20
00213 50* C HERE IF A GOOD SYMBOL.
00214 51* 31 NO(I) = J1
00215 52* ALPHA(I) = ALPHA2(I)
00216 53* 35 CONTINUE

```

```

00216 54* C
00216 55* C
00216 56* C
00220 57* 160 ISTEP = 1
00221 58* IVC = 1
00221 59* C
00221 60* C START OF INPUT FOR A TIME STEP.
00221 61* C
00221 62* C READ ANY CHGS WANTED.
00222 63* 39 IVC = 0
00223 64* WRITE(6,40)
00225 65* 40 FORMAT(' INPUT WANTED CHGED.(A4,2X,F10.4)',
00225 66* 1 ' DONE= NO MORE CHGES...')
00226 67* 42 WRITE(6,41)
00230 68* 41 FORMAT(' SYMB VALUE...')
00231 69* READ(5,43,ERR=42) SYMBOL,CVALUE
00235 70* 43 FORMAT(A4,2X,F10.4)
00236 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00236 71* IF(SYMBOL.EQ.DONE) GO TO 50
00240 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00240 72* IF(SYMBOL.EQ.BLANK) GO TO 42
00242 73* DO 45 I = 1,MAXNO
00245 74* I1 = 1
00246 *DIAGNOSTIC* THE TEST FOR EQUALITY BETWEEN NON-INTEGERS MAY NOT BE MEANINGFUL.
00246 75* IF(SYMBOL.EQ.I1) GO TO 46
00250 76* 45 CONTINUE
00250 77* C HERE WHEN A SYMBOL NOT RECOGNIZED.
00252 78* WRITE(6,49)
00254 79* 49 FORMAT(' SYMBOL NOT RECOGNIZED. TRY AGAIN.')
00255 80* GO TO 42
00255 81* C
00256 82* 46 VCHGS(IVC,1) = I1
00257 83* VCHGS(IVC,2) = CVALUE
00260 84* WRITE(6,150) SYMBOL,CVALUE
00264 85* 150 FORMAT(' ',A4,2X,F10.4)
00265 86* IF(IVC.LT.200) GO TO 47
00267 87* WRITE(6,48)
00271 88* 48 FORMAT(' BUFFER FOR CHGS FULL.')
00272 89* GO TO 50
00273 90* 47 IVC = IVC + 1
00274 91* I1VC = I1VC + 1
00275 92* GO TO 42
00275 93* C
00275 94* C
00275 95* C READ TIME STEP CARD.
00276 96* 50 WRITE(6,55)
00300 97* 55 FORMAT(' TIME STEP (A4,1X,F6.0,F6.0)...')
00300 98* 1 ' UNIT PRINT TIME (UNIT=DAYS, HOUR, MINS, SECS, STEP)...')
00301 99* READ(5,56,ERR=50) IUNITS,XP,XZ
00306 100* 56 FORMAT(A4,1X,2F6.0)
00307 101* 1 = XZ
00310 102* 12 = XP
00311 103* IF(IUNITS.NE.I1STEP) GO TO 200
00311 104* C #ILL CALCULATE AND OUTPUT RUN STEP.
00313 105* CALL TIMPRT(PLOTTH, ID1, IH1, IM1, IS1)
00314 106* TT = PLOTTH + PUNSTP
00315 107* CALL TIMPRT(TT, ID2, IH2, IM2, IS2)

```

```

00316 108* CALL IMPRT(RUNSTEP, ID3, IH3, IM3, IS3)
00317 109* WRITE(6,108) ID1, IH1, IM1, IS1, ID2, IH2, IM2,
00317 110* 1 IS2, ID3, IH3, IM3, IS3
00335 111* 106 FORMAT(' RUNSTEP ', 7(' '), ' BEGIN',
00335 112* 1 13X, 7(' '), ' END', 8X, 7(' '), ' INTERVAL', /
00335 113* 2 ' ', 13, ' DAY', 13, ' HR', 13, ' MN', 13, ' SE',
00335 114* 3 2(13X, 13, ' DAY', 13, ' HR', 13, ' MN', 13, ' SE', ))
00336 115* 200 DO 58 I1 = 1, 4
00341 116* ITSY = 11
00342 117* IF(IUNITS .EQ. ITSYM(I1)) GO TO 62
00344 118* 58 CONTINUE
00344 119* C HERE WHEN ERROR IN TIME UNIT.
00346 120* GO TO 50
00346 121* C
00347 122* 62. IF(I1 .LE. 0) GO TO 50
00351 123* IF(I2 .LE. 0) GO TO 50
00353 124* JTSTEP(ISTEP, 1) = 1
00354 125* JTSTEP(ISTEP, 2) = ITSY
00355 126* JTSTEP(ISTEP, 3) = 11VC
00356 127* JTSTEP(ISTEP, 5) = 12
00356 128* C RUNSTEP = MINS FOR RUN STEP.
00357 129* RUNSTP = RUNSTP + (FLOAT(I1)*TDELTA(ITSY))
00360 130* PLOTPT = RUNSTP / 100.
00360 131* C
00360 132* C
00360 133* C READ EXEC PARAMETER.
00361 134* 69 WRITE(6, 70)
00363 135* 70 FORMAT(' EXEC PARAM. (A4) (MORE, RUN, PLOT)...')
00364 136* 73 READ(5, 72, ERR=69) IEXECI
00367 137* 72 FORMAT(A4)
00370 138* IF(IEXECI .EQ. JMORE) GO TO 80
00372 139* IF(IEXECI .EQ. JRUN) GO TO 80
00374 140* IF(IEXECI .EQ. IPLL) GO TO 80
00376 141* WRITE(6, 75)
00400 142* 75 FORMAT(' DO NOT RECOGNIZE EXEC PAR. TRY AGAIN. ')
00401 143* GO TO 73
00401 144* C
00402 145* 80 JTSTEP(ISTEP, 4) = IEXECI
00403 146* IF(IEXECI .NE. JMORE) GO TO 90
00403 147* C
00405 148* ISTEP = ISTEP + 1
00406 149* IF(ISTEP .LE. 100) GO TO 39
00406 150* C
00410 151* WRITE(6, 85)
00412 152* 85 FORMAT(' BUFFER FULL FOR TIME STEPS. ')
00413 153* JTSTEP(50, 4) = JRUN
00414 154* 90 RETURN
00415 155* END

```

ORIGINAL PAGE IS  
 OF POOR QUALITY

END OF COMPILATION: 7 DIAGNOSTICS.

```

00101 1* SUBROUTINE TTYOUT
00103 2* COMMON/ARRAY/ALPH(20),TITLE(400),COL(20),ALPHA(20)
00104 3* COMMON/NUMERO/K,NO(20)
00104 4* .L. NTIMEC,UNITS,NZ,NTIMEP,NN,MAXNO,NTIME.

00105 5* COMMON/TTYJOB/VCHGS(200,2),JTSTEP(100,5),PLOTPT,PLOTTM
00106 6* COMMON/DEMAND/ITAPE0,ITME,IEXECN,ICONVI
00107 7* COMMON/PPPPPP/PLOTBF(101,6),RUNSTP,IEXECI,IPLTPT
00107 8* I ,TLOTBF(101),KSTOPP
00110 9* COMMON/STORE/JUNK(118),HEAD(19),NDEXP
00111 10* DATA INOOO/'N '/'
00113 11* DATA JMORE/'MORE',JRUN/'RUN '/'
00116 12* DATA IPLL/IPLPT/'
00120 13* DIMENSION TDELTA(4)
00121 14* DATA TDELTA/.01666667,1.,60.,1440./
00121 15* C
00121 16* C
00123 17* KSTOPP = 0
00124 18* TDELTA(1) = 1./60.
00125 19* T = A(1)
00125 20* C SEE IF TIME TO BUFFER VARIABLES TO PLOT.
00126 21* IF(T .GE. TIMEC .AND. JTSTEP(ISTEP,4)
00126 22* I .EQ. JRUN) GO TO 71
00130 23* IF(T .GE. TIMEC .AND. JTSTEP(ISTEP,4)
00130 24* I .EQ. IPLL) GO TO 71
00132 25* IF(PLOTTM .GT. T) GO TO 80
00134 26* 71 IF(IPLTPT .GT. 101) IPLTPT = 101
00136 27* TLOTBF(IPLTPT) = T
00137 28* DO 81 I = 1,K
00142 29* J = NO(I)
00143 30* PLOTBF(IPLTPT,I) = A(J)
00144 31* 81 CONTINUE
00146 32* IPLTPT = IPLTPT + 1
00147 33* PLOTTM = PLOTTM + PLOTPT
00150 34* IF(IEXECI .NE. IPLL) GO TO 80
00152 35* CALL PLOT33
00152 36* C SEE IF THIS ROUTINE CALLED JUST AFTER INPUTTING.
00153 37* 80 IF(ITME .EQ. 0) GO TO 100
00155 38* ISTEP = 1
00156 39* IVC = 1
00157 40* IF(IEXECI .EQ. IPLL) GO TO 200
00161 41* CALL NEWPAG
00162 42* WRITE(6,361) NDEXP,(HEAD(J),J=1,15)
00171 43* WRITE(6,211) (ALPHA(J),J=1,K)
00177 44* IPAG = 2
00200 45* GO TO 200
00200 46* C
00200 47* C HERE AFTER A LOOP THRU MODEL.
00201 48* 100 IF(IEXECI .EQ. IPLL) GO TO 35
00201 49* C SEE IF TIME TO PRINT.
00203 50* IF(T .LT. TIMEP) RETURN
00203 51* C
00203 52* C OUTPUT VARIABLES.
00203 53* C
00203 54* C COMPUTE TIME.
00205 55* CALL TIMPRIT,IDI,IHI,IMI,ISI)
00205 56* C BUILD OUTPUT IN COL(S).
00206 57* 11 DO 15 I = 1,K.
00211 58* J = NO(I)
00212 59* COL(I) = A(J)
00213 60* 15 CONTINUE
00213 61* C

```



```

00215 62* ASSIGN 17 TO LBACKK
00216 63* GO TO 310
00217 64* 17 WRITE(6,18) ID1,IH1,IM1,IS1,(COL(J),J=1,K)
00231 65* 18 FORMAT(' ',4I3,5F10.4,F9.3)
00231 66* C
00231 67* C
00231 68* C COMPUTE NEXT TIME TO PRINT.
00232 69* 30 I = (T / TDELTA(ITSY)) + JTSTEP(ISTEP,5)
00233 70* TIMEP = L * TDELTA(ITSY)
00233 71* C
00233 72* C SEE IF END OF TIME STEP.
00234 73* 35 IF(T .LT. TIMEC) RETURN
00234 74* C
00234 75* C
00234 76* C HERE WHEN FINISHED A TIME STEP.
00236 77* IF(JTSTEP(ISTEP,4) .EQ. JMORE) GO TO 50
00236 78* C
00236 79* C HERE WHEN FINISHED RUN STEP.
00240 80* KSTOPP = 1
00241 81* IF(IEXECI .EQ. IPLL) GO TO 360
00243 82* CALL PAGE3
00243 83* C GO TO ROUTINE TO PLOT.
00244 84* 360 CALL PLOT33
00244 85* C
00244 86* C
00244 87* C HERE WHEN NEED TO GO BACK TO TTYIN TO GET NEW TIME STEPS.
00245 88* IEXECN = 1
00246 89* RETURN
00246 90* C
00246 91* C
00246 92* C HERE WHEN NEXT TIME STEP IS IN JTSTEP(50,4), AND
00246 93* CHGS.FOR SAME IN VCHGS(200,2). EXEC PARAM.IS MORE.
00247 94* 50 ISTEP = ISTEP + 1
00247 95* C
00247 96* C COMPUTE TIME.
00250 97* 200 CALL TIMPRT(T, ID1, IH1, IM1, IS1)
00250 98* C SEE IF ANY CHGS. ASSOCIATED WITH THIS TIME STEP.
00251 99* 201 IF(JTSTEP(ISTEP,3) .EQ. 0) GO TO 210
00253 100* IF(A(2) .GT. .5) A(2) = .5
00255 101* I1 = VCHGS(IVC,1)
00256 102* IF(IEXECI .EQ. IPLL) GO TO 408
00260 103* ASSIGN 205 TO LBACKK
00261 104* GO TO 310
00262 105* 205 WRITE(6,206) ID1,IH1,IM1,IS1,TITLE(11),A(11),VCHGS(IVC,2)
00273 106* 206 FORMAT(' AT ',I3,' DAY',' ',I2,' HR',' ',I2,
00273 107* ' ' MIN',' ',I2,' SEC',
00273 108* ' ' . . . ' ',A4,' FROM',F10.4,' TO',F10.4)
00274 109* 408 A(11) = VCHGS(IVC,2)
00275 110* IVC = IVC + 1
00276 111* JTSTEP(ISTEP,3) = JTSTEP(ISTEP,3) - 1
00277 112* GO TO 201.
00277 113* C
00277 114* C
00277 115* C
00277 116* C PUT MAX.TIME FOR TIME STEP IN TIMEC.
00300 117* 210 ITSY = JTSTEP(ISTEP,2)
00301 118* I = (T/TDELTA(ITSY)) + JTSTEP(ISTEP,4)

```

```

00302 119*      TIMEC = I * IDLTA(IISY)
00303 120*      IF(IEXECI .EQ. IPLL) GO TO 35
00303 121*      C
00303 122*      C OUTPUT IF THIS 1ST TIME TTYOUT CALLED.
00305 123*      IF(IIME .EQ. 1) GO TO 11
00305 124*      C
00305 125*      C PUT NEXT TIME TO PRINT IN TIMEP.
00307 126*      GO TO 30
00307 127*      C
00307 128*      C ROUTINE TO ASK FOR HARD COPY EVERY 25 LINES,ERASE PAGE,OUT HEADING,
00310 129*      310 IPAG = IPAG + 1
00311 130*      IF(IPAG .LT. 25) GO TO 332
00313 131*      CALL PAGE3
00314 132*      WRITE(6,361) NOEXP,(HEAD(J),J=1,15)
00323 133*      361 FORMAT(' ',14,2X,15A4)
00324 134*      WRITE(6,211) (ALPHA(J),J=1,K)
00332 135*      211 FORMAT('0DAY HR MN SE',6(6X,A4))
00333 136*      IPAG = 2
00334 137*      332 GO TO 1BACKK(17,205,360)
00335 138*      END

```

END OF COMPILATION: NO DIAGNOSTICS.

C-2