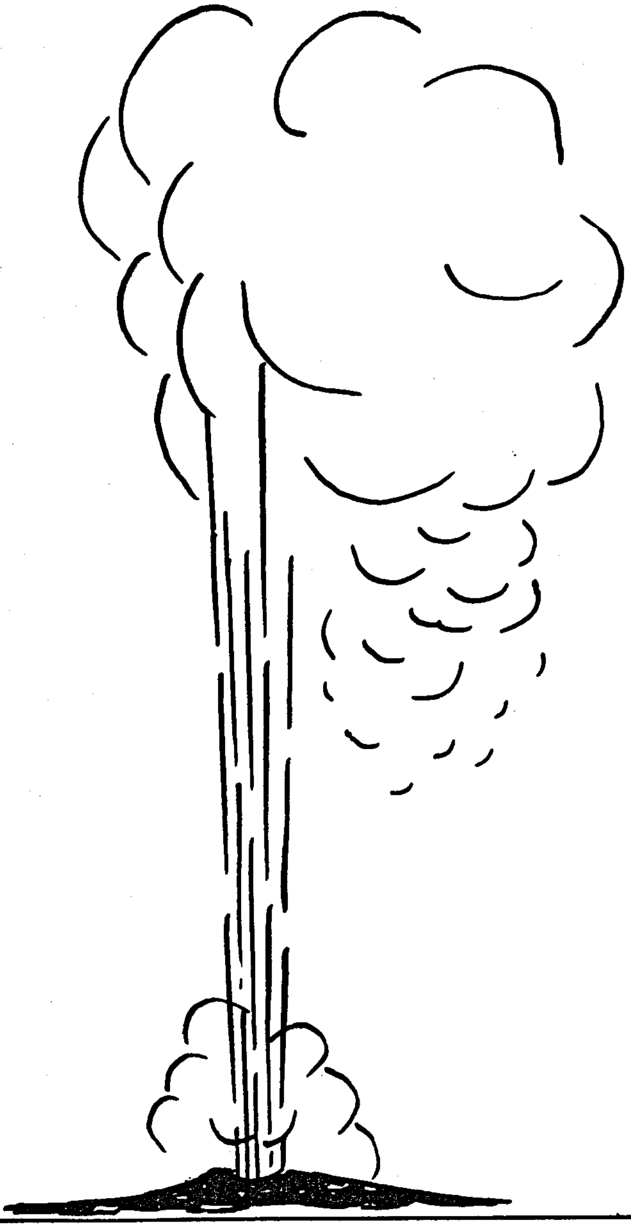
UTEC-ME-76-212



A COMPUTER PROGRAM FOR DETERMINING THE THERMODYNAMIC PROPERTIES OF FREON REFRIGERANTS

By
David H. Riemer
Harold R. Jacobs
Robert F. Boehm

MASTER

December 1977
(TIC Issuance Date)

Work Performed Under Contract No. EY-76-S-07-1523

University of Utah
Department of Mechanical Engineering
Salt Lake City, Utah

U. S. DEPARTMENT OF ENERGY
Geothermal Energy

DISCLAIMER

DISCLAIMER

NOTICE

Available from the National Technical Information Service, U. S. Department of Commerce, Springfield, Virginia 22161.

Price: Paper Copy $6.50
Microfiche $3.00

UTEC-ME-76-212
Distribution Category UC-66d

A UNIVERSITY OF UTAH
DIRECT CONTACT GEOTHERMAL POWER
PROJECT REPORT

A Computer Program for Determining the Thermodynamic Properties of Freon Refrigerants

David H. Riemer
Harold R. Jacobs
Robert F. Boehm

Prepared for the
Energy Research and Development Administration
Under

Grant No. AER 75-01039

by

ACKNOWLEDGEMENT

This work was prepared as a part of ERDA Grant No. AER-75-01039 which was originally funded as a National Science Foundation RANN Grant entitled, " Feasibility Study of the Application of Direct Contact Heat Exchangers to Power Cycles Utilizing Geothermal Brines". The work was conducted in the Mechanical and Industrial Engineering Department at the University of Utah under the direction of Drs. Harold R. Jacobs and Robert F. Boehm.

Appreciation for supplemental funding and encouragement is given to Dr. J. Kunze of Aerojet Nuclear Company, Idaho Falls, Idaho, for ERDA support under contract AT(10-1)1523.

The authors would also like to express their appreciation to Dr. Harold Jackson of E. I. DuPont De Nemours and Company for his personal assistance.

NOMENCLATURE

a	constant in heat capacity of vapor equation
b	
c	
d	
f	
A_2	constant in Downing-Martin equation
A_3	
A_4	
A_5	
A_6	
B_2	
B_3	
B_4	
B_5	
B_6	
C_1	
C_2	
C_3	
C_4	
C_5	
C_6	
h	enthalpy (Btu/lbm)
K	constant in Downing-Martin equation
P	pressure (psia)
R	universal gas constant
S	enthalpy (Btu/lbm°R)
T	temperature (°R)
V	specific volume (ft^3/lbm)
α	constant in Downing-Martin equation
ϕ	conversion factor
β	constant in Downing-Martin equation

ABSTRACT

This program was written to be used as a subroutine. The program determines the thermodynamics of Freon refrigerants. The following refrigerants can be analyzed F-11, F-12, F-13, F-14, F-21, F-22, F-23, F-113, and F-114. The subroutine can evaluate a thermodynamic state for these refrigerants given any of the following pairs of state quantities: pressure and quality, pressure and entropy, pressure and enthalpy, temperature and quality, temperature and specific volume and temperature and pressure. These six pairs of knowns allow the user to analyze any thermodynamic cycle utilizing a refrigerant as the working fluid. The Downing form of the Martin equation of state was used. This report contains a brief description, flow chart and listing of all subroutines required.

INTRODUCTION

This subroutine computes the thermodynamic properties of Freon refrigerants. The subroutine was developed to allow for a wide range of applications such that it may be used to determine the properties of Freon refrigerants in a refrigeration cycle or, as the authors developed it, for use in evaluating binary fluid cycles. This routine can be treated as a "black box" by the user with a limited background in programming or the program may be modified to meet the users requirements. This manual was written to assist the user in which ever manner he intends to use this subroutine.

DEVELOPMENT OF PROPERTIES

The basic equation used is the Downing form of the Martin equation of state which have the form:

$$P = \frac{RT}{V-\beta} + \frac{A_2 + B_2T + C_2e^{(-KT/Tc)}}{(V-\beta)^2} + \frac{A_3+B_3T+C_3e^{(-KT/Tc)}}{(V-\beta)^3}$$

$$+ \frac{A_4+B_4T+C_4e^{(-KT/Tc)}}{(V-\beta)^4} + \frac{A_5+B_5T+C_5e^{(-KT/Tc)}}{(V-\beta)^5} + \frac{A_6+B_6T+C_6e^{(-KT/Tc)}}{e^{\alpha V}(1+C_1e^{\alpha V})}$$

$$h = aT+bT^2/2+CT^3/3+dT^4/4-f/T+\emptyset PV+\emptyset\left(\frac{A_2}{(V-\beta)} + \frac{A_3}{2(V-\beta)^2} + \frac{A_4}{3(V-\beta)^3}\right.$$

$$\left. + \frac{A_5}{4(V-\beta)^4} + \frac{A_6}{\alpha}\left(\frac{1}{e^{\alpha V}} - C_1 \ln\left(\frac{C_1e^{\alpha V}+1}{C_1e^{\alpha V}}\right)\right)\right)$$

$$+ \emptyset e^{-KT/Tc}\ (1+KT/Tc)\left(\frac{C_2}{V-\beta} + \frac{C_3}{2(V-\beta)^3} + \frac{C_4}{3(V-\beta)^3} + \frac{C_5}{4(V-\beta)^4}\right.$$

$$\left. + \frac{C_6}{\alpha e^{\alpha V}} - \frac{C_6C_1}{\alpha} \ln\left(\frac{C_1e^{\alpha V}+1}{C_1e^{\alpha V}}\right)\right) + \Delta h(\text{latent at } -40°\text{F})-h(\text{saturated}$$

$$\text{vapor at } -40°\text{F})$$

$$\begin{aligned}S = {} & a\ \ln T+BT+CT^2/2+dT^3/3-f/2T^2+\phi R\ln(V-\beta)+\phi\Bigg(\frac{B_2}{V-\beta}+\frac{B_3}{2(V-\beta)^2}\\ & +\frac{B_4}{3(V-\beta)^3}+\frac{B_5}{4(V-\beta)^4}+\frac{B_6}{\alpha}\Bigg(\frac{1}{e^{\alpha v}}-C_1\ln\Bigg(\frac{C_1e^{\alpha V}+1}{C_1e^{\alpha V}}\Bigg)\Bigg)\Bigg)\\ & +\frac{\phi e^{-KT/TC}}{Tc}\Bigg(\frac{C_2}{V-\beta}+\frac{C_3}{2(V-\beta)^2}+\frac{C_4}{3(V-\beta)^3}+\frac{C_5}{4(V-\beta)^4}+\frac{C_6}{\alpha e^{\alpha V}}\\ & -\frac{C_6C_1}{\alpha}\ln\Bigg(\frac{C_1e^{\alpha V}+1}{C_1e^{\alpha V}}\Bigg)\Bigg)+\Delta h(\text{latent at }-40°\text{F})/-40-S(\text{saturated}\\ & \text{vapor at }-40°\text{F})\end{aligned}$$

The above equations only predict properties in the vapor region. In order to determine saturated liquid properties the Clapeyron equation is used to compute saturated liquid enthalpy and entropy. Saturated liquid specific volume is computed as a function of temperature. Compressed liquid states are approximated by saturated liquid states at the same temperature.

For evaluating an ideal cycle the following processes are employed, constant entropy(turbine, pump and compressor), constant enthalpy (expansion valve), constant pressure(heat exchanger). In order to determine properties for these processes, properties must be determined with one of the following pairs known (Recall that a thermodynamic state requires that two state quantities be known and for saturation states temperature and pressure are dependent): temperature and pressure, temperature and quality, pressure and quality, pressure and enthalpy and pressure and entropy. Since pressure is a function of specific volume and temperature, when pressure and temperature are known a search techinque must be used to find the specific volume. Once the specific is found the remaining properties may be calculated. The saturation pressure is known as a function of temperature, thus allowing saturation conditions of known pressure or temperature to be solved as the case of pressure and temperature already described. Once the saturation properties have been determined the effect of a given quality can be determined. The cases of pressure and entropy or enthalpy as knowns require a two variable search since all three of these quantities and functions of specific volume and temperature. This search is only used in the superheated vapor region or the compressed liquid approximated region because for saturation states only a quality need be computed. For all searches in FREON the Newton Raphson iteration technique was used. This method was selected for its generally rapid convergence.

PROGRAM DEVELOPMENT

Subroutine FREON was written in FORTRAN V for the UNIVAC 1108 series computer but should be easily adapted to any similar FORTRAN computer. The subroutine allows the user to use the program at two levels, first as a "black box" (the main text describes the use and general development of the program), second the program may be modified to best suit the users needs. (The remaining text contains a more complete description of each subroutine.) Subroutines were used throughout to allow for easy modification and as a means of tracing any possible problems. This report contains variable description tables, flow charts, equations and brief descriptions of all subroutines.

USE OF FREON

In order to use FREON the user must supply to the subroutine the following information:

1. The process for which properties are to be determined (CYCLE). The allowable values of CYCLE are the following pairs of knowns:

   TP - temperature and pressure

   TX - temperature and quality

TV - temperature and specific volume

PH - pressure and enthalpy

PS - pressure and entropy

PX - pressure and quality

These are the only allowable values of CYCLE and the order is important, i.e., PT or SP, etc., are illegal and will generate an error termination of FREON.

2. The value of the first known specified by the value of CYCLE (FGIVEN).
3. The value of the second known specified by the value of CYCLE (SGIVEN).
4. Freon refrigerant type (FTYPE). FTYPE is formed by placing an F in front of the refrigerant number. The following values are acceptable: F11, F12, F13, F14, F21, F22, F23, F113 and F114.
5. Print selector - if NPRT = 0 then do not print out results, otherwise print out results

FREON will return the following information:

1. Quality - applies to saturation states (if greater than 1 then superheated vapor and if less than 1 then compressed liquid.)
2. enthalpy (Btu/lbm)
3. entropy (Btu/lbm°R)
4. specific volume (ft^3/lbm)
5. temperature (°F)
6. pressure (psia)
7. saturated liquid enthalpy (Btu/lbm)

8. saturated liquid entropy (Btu/lbm°R)
9. saturated liquid specific volume (ft^3/lbm)
10. saturated vapor enthalpy (Btu/lbm)
11. saturated vapor entropy (Btu/lbm)
12. saturated vapor specific volume (ft^3/lbm)

A typical call on FREON would look like:

```
CALL FREON ('TP', TEMP, PRESS, 'F113', H, S, V, T, P, HV, SU, VV,
HL, SL, VL, QUAL, NPRT)
```

This call on FREON evaluates properties for FREON-113 given temperature (TEMP) and pressure (PRESS). Subroutine FREON was written so as to be independent of the calling program except for the passage of the above parameters, therefore no external variables need be placed in common.

Q Array

VAPOR PRESSURE COEFFICIENTS

Q(1,1) A	Q(2,1) B	Q(3,1) C	Q(4,1) D
Q(1,2) E	Q(2,2) F	Q(3,2) G	Q(4,2) H

EQUATION OF STATE COEFFICIENTS

Q(1,3) L	Q(2,3) R	Q(3,3) β	Q(4,3) Tc(°F)

LIQUID DENSITY

Q(1,4) A_L	Q(2,4) A_2	Q(3,4) B_2	Q(4,4) C_2
Q(1,5) B_L	Q(2,5) A_3	Q(3,5) B_3	Q(4,5) C_3
Q(1,6) C_L	Q(2,6) A_4	Q(3,6) B_4	Q(4,6) C_4
Q(1,7) D_L	Q(2,7) A_5	Q(3,7) B_5	Q(4,7) C_5
Q(1,8) E_L	Q(2,8) A_6	Q(3,8) B_6	Q(4,8) C_6

HEAT EQUATION COEFFICIENTS

Q(1,9) a	Q(2,9) b	Q(3,9) c	Q(4,9) d
Q(1,10) f	Q(2,10) X	Q(3,10) Y	Q(4,10) S_{CRIT}
Q(1,11) ABST	Q(2,11) H_{CRIT}	Q(3,11) NTYPE	Q(4,11) V_{CRIT}
Q(1,12) C_1	Q(2,12) k	Q(3,12) α	Q(4,12) P_{CRIT}

Figure 1.

FREON DESCRIPTION

FREON is the main routine for determining the thermodynamic properties of the Freon refrigerants. The routine selects the correct set of coefficients for the fluid to be evaluated. These coefficients are stored in the array Q which has the form as shown in Figure 1 FREON then must determine if the two state quantities given are legitimate values, i.e. for temperature and quality the temperature must be less than or equal to the critical temperature and the quality must lie between zero and one. Once the quantities are found to be legitimate FREON must determine the region to which the point corresponds, i.e. compressed liquid, saturated liquid-vapor equilibrium or superheated vapor. Given the region FREON then calls the proper subroutines to compute the remaining thermodynamic properties.

VARIABLE SYMBOL TABLE

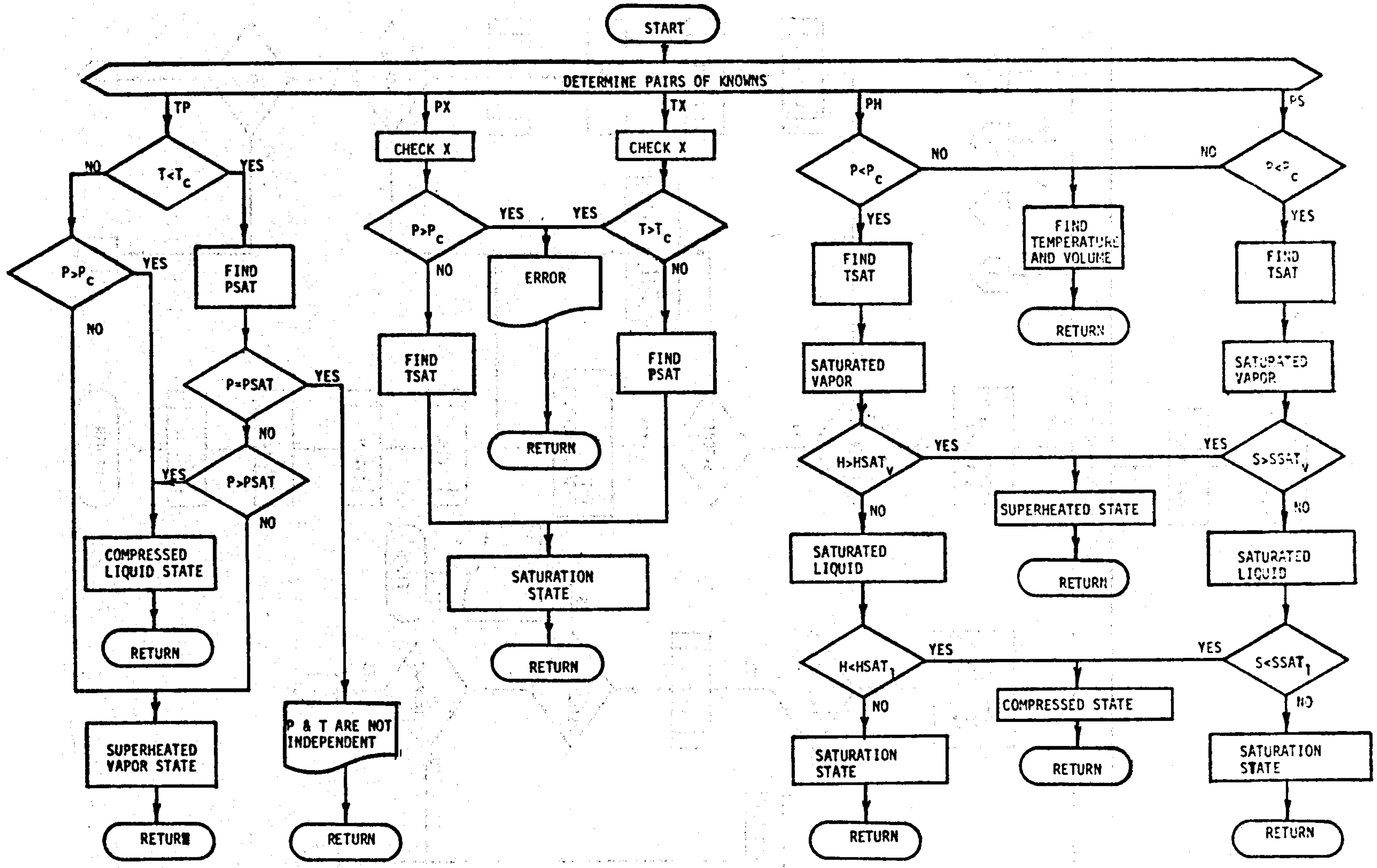
FREON

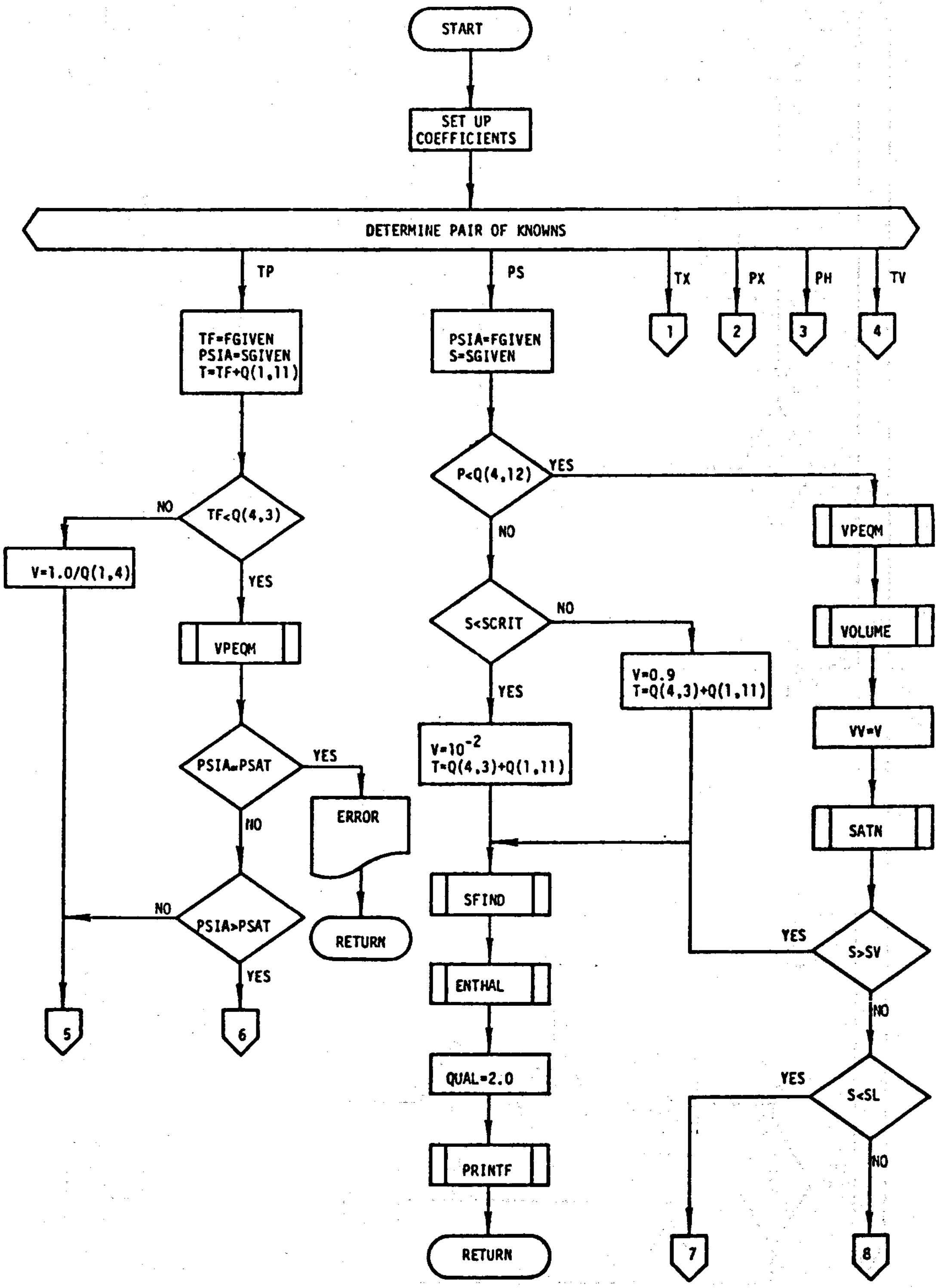
VARIABLE	DESCRIPTION
BTOLER	Lower tolerance on convergence and saturation tests
CYCLE	Type of calculation to be made, CYCLE has the following possible pairs of knowns: 1. temperature and pressure = TP 2. temperature and quality = TX 3. temperature and specific volume = TV 4. pressure and quality = PX 5. pressure and enthalpy = PH 6. pressure and entropy = PS

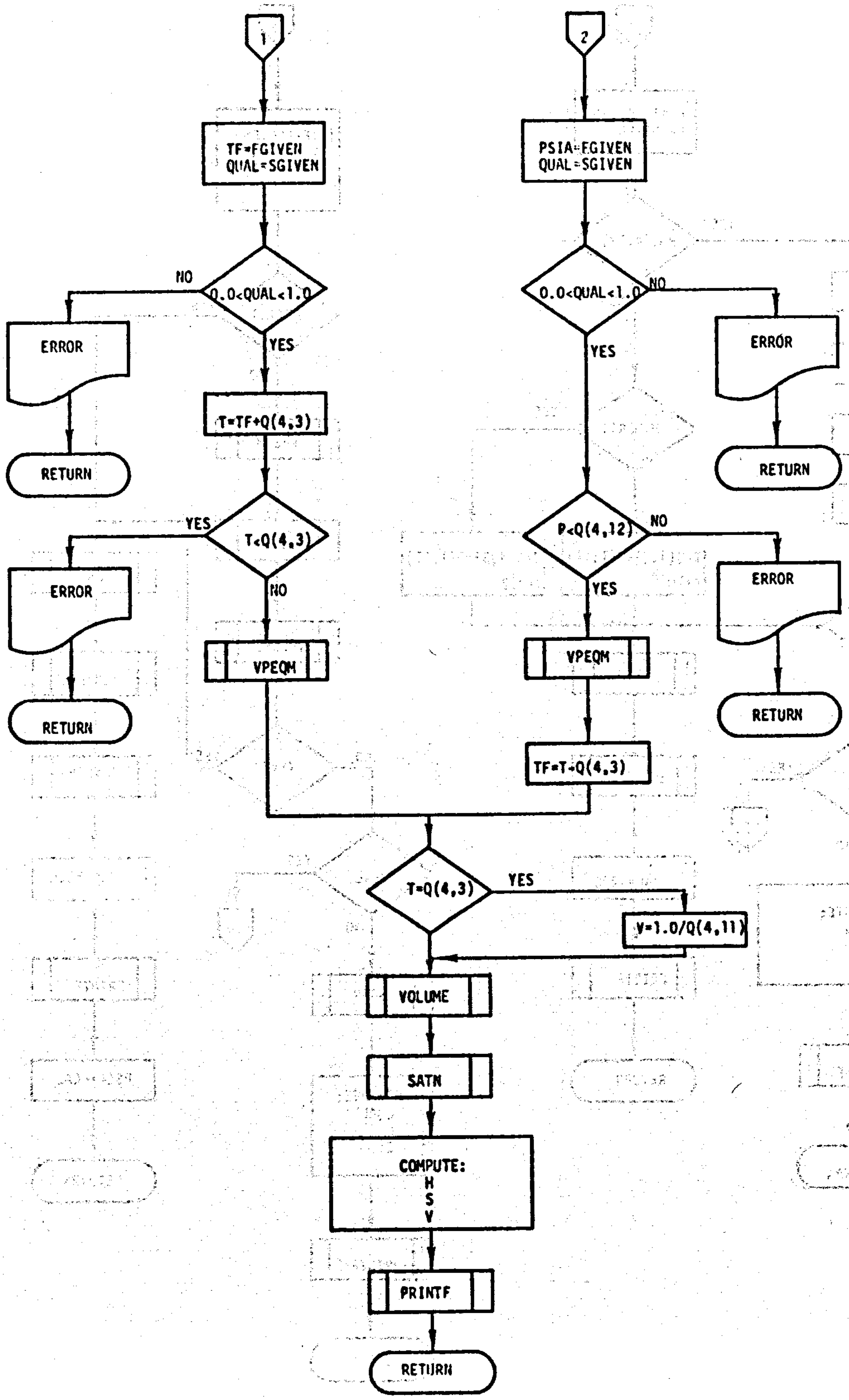
DELTH	h_{fg} - calculated from Clapeyron Equation
DERROR	tolerance used in checks for zero
DHDT	$\frac{\partial h(V,T)}{\partial T}$
DHDV	$\frac{\partial h(V,T)}{\partial V}$
DPSDT	$\frac{dP_{SAT}(T)}{dT}$
DPSDT2	$\frac{d^2P_{SAT}(T)}{dT^2}$
DSDT	$\frac{\partial S(V,T)}{\partial T}$
DSDV	$\frac{\partial S(V,T)}{\partial T}$
DVLDT	$\frac{dV_L(T)}{dT}$
F11	2-dimensional array containing the coefficients for F11
F12	2-dimensional array containing the coefficients for F12
F13	2-dimensional array containing the coefficients for F13
F14	2-dimensional array containing the coefficients for F14
F21	2-dimensional array containing the coefficients for F21
F22	2-dimensional array containing the coefficients for F22
F23	2-dimensional array containing the coefficients for F23
F113	2-dimensional array containing the coefficients for F113

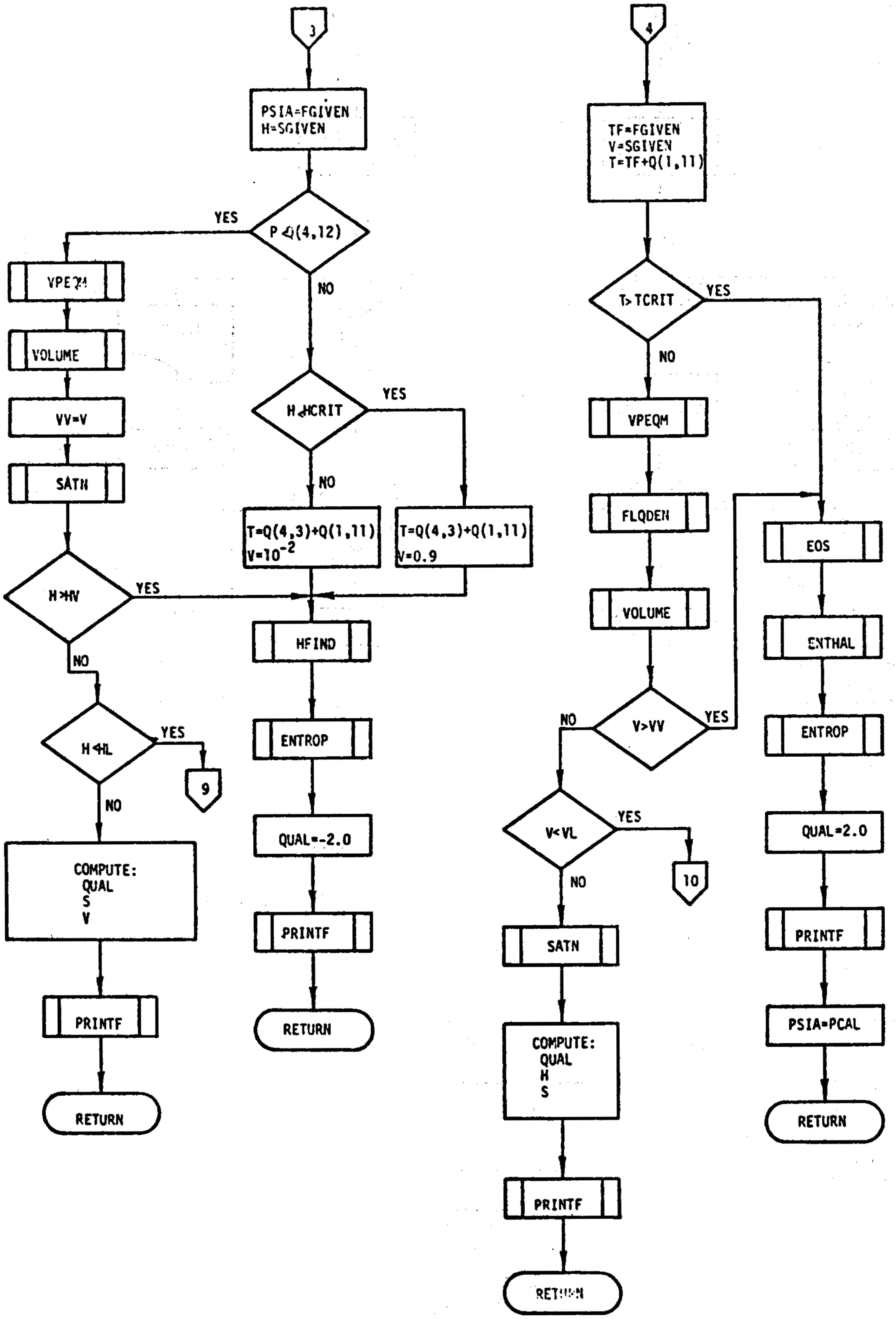
F114	2-dimensional array containing the coefficients for F114
FGIVEN	First known passed to FREON, i.e. temperature for values of CYCLE of TP, TX, TV and pressure for values of CYCLE of PS, PH AND PX
FTYPE	Freon type
H	enthalpy (Btu/lbm)
HCRIT	critical enthalpy (Btu/lbm)
HL	saturated liquid enthalpy (Btu/lbm)
HV	saturated vapor enthalpy (Btu/lbm)
I	index
J	index
LIMIT	maximum number of iterations allowed in any one search
M2	unit number for printer
NPRT	print selector NPRT = 1 print and results NPRT ≠ 1 do not print results
NTYPE	saturated liquid specific volume equation selector
PCAL	calculated pressure (psia)
PCRIT	critical pressure (psia)
PSAT	saturation pressure (psia)
PSIA	Pressure (psia)
Q	2-dimentional array containing the coefficients for the fluid to be evaluated
QUAL	quality

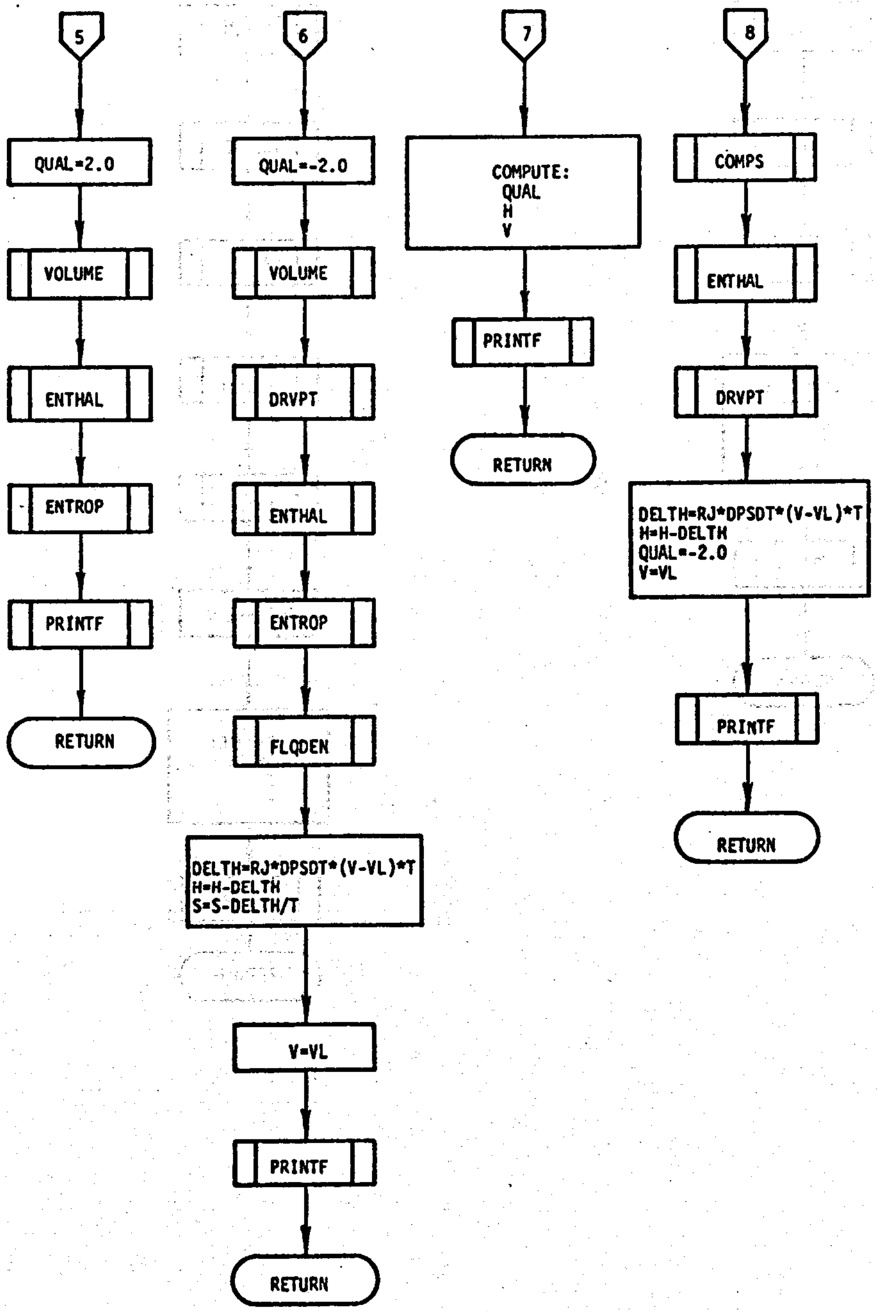
RJ	conversion factor = 0.185053
S	entropy (Btu/lbm°R)
SCRIT	critical entropy (Btu/lbm°R)
SGIVEN	second known passed to FREON
SL	saturated liquid entropy (Btu/lbm°R)
SV	saturated vapor entropy (Btu/lbm°R)
T	temperature (°R)
TF	temperature (°F)
TCRIT	critical temperature (°F)
TTOLER	upper tolerance on convergence and saturation tests
V	specific volume (ft^3/lbm)
VL	saturated liquid specific volume (ft^3/lbm)
VV	saturated vapor specific volume (ft^3/lbm)

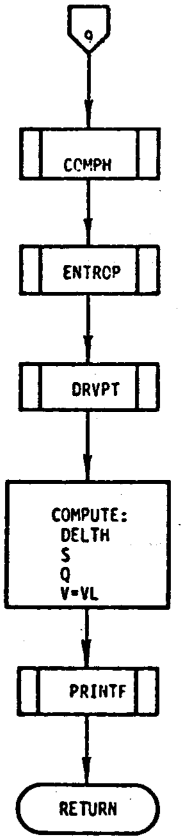

START
DETERMINE PAIRS OF KNOWNS
TP
PX
TX
PH
PS
T<T$_c$
NO
YES
P>P$_c$
YES
NO
FIND PSAT
P=PSAT
YES
NO
P>PSAT
YES
NO
COMPRESSED LIQUID STATE
RETURN
SUPERHEATED VAPOR STATE
RETURN
P & T ARE NOT INDEPENDENT
RETURN
CHECK X
P>P$_c$
YES
NO
FIND TSAT
CHECK X
T>T$_c$
YES
NO
FIND PSAT
ERROR
RETURN
SATURATION STATE
RETURN
P<P$_c$
NO
YES
FIND TSAT
SATURATED VAPOR
H>HSAT$_v$
YES
NO
SATURATED LIQUID
H<HSAT$_l$
YES
NO
SATURATION STATE
RETURN
FIND TEMPERATURE AND VOLUME
RETURN
SUPERHEATED STATE
RETURN
COMPRESSED STATE
RETURN
P<P$_c$
NO
YES
FIND TSAT
SATURATED VAPOR
S>SSAT$_v$
YES
NO
SATURATED LIQUID
S<SSAT$_l$
YES
NO
SATURATION STATE
RETURN

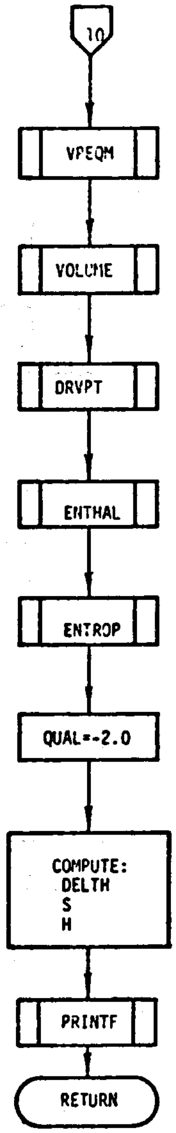

START
SET UP COEFFICIENTS
DETERMINE PAIR OF KNOWNS
TP
PS
TX
PX
PH
TV
1
2
3
4
TF=FGIVEN
PSIA=SGIVEN
T=TF+Q(1,11)
PSIA=FGIVEN
S=SGIVEN
P<Q(4,12)
YES
NO
VPEQM
VOLUME
VV=V
SATN
TF<Q(4,3)
NO
YES
V=1.0/Q(1,4)
VPEQM
S<SCRIT
NO
YES
V=0.9
T=Q(4,3)+Q(1,11)
V=10^-2
T=Q(4,3)+Q(1,11)
PSIA=PSAT
YES
NO
ERROR
RETURN
PSIA>PSAT
NO
YES
5
6
SFIND
ENTHAL
QUAL=2.0
PRINTF
RETURN
S>SV
YES
NO
S<SL
YES
NO
7
8


1
2
TF=FGIVEN
QUAL=SGIVEN
PSIA=FGIVEN
QUAL=SGIVEN
NO
0.0<QUAL<1.0
0.0<QUAL<1.0
NO
ERROR
YES
YES
ERROR
T=TF+Q(4,3)
RETURN
RETURN
YES
T<Q(4,3)
P<Q(4,12)
NO
ERROR
NO
YES
ERROR
VPEQM
VPEQM
RETURN
RETURN
TF=T-Q(4,3)
T=Q(4,3)
YES
V=1.0/Q(4,11)
VOLUME
SATN
COMPUTE:
H
S
V
PRINTF
RETURN


3
PSIA=FGIVEN
H=SGIVEN
P<Q(4,12)
YES
NO
VPEQM
VOLUME
VV=V
SATN
H>HV
YES
NO
H<HL
YES
9
NO
COMPUTE:
QUAL
S
V
PRINTF
RETURN
H<HCRIT
YES
NO
T=Q(4,3)+Q(1,11)
V=10^-2
T=Q(4,3)+Q(1,11)
V=0.9
HFIND
ENTROP
QUAL=-2.0
PRINTF
RETURN
4
TF=FGIVEN
V=SGIVEN
T=TF+Q(1,11)
T>TCRIT
YES
NO
VPEQM
FLQDEN
VOLUME
V>VV
NO
YES
V<VL
YES
10
NO
SATN
COMPUTE:
QUAL
H
S
PRINTF
RETURN
EOS
ENTHAL
ENTROP
QUAL=2.0
PRINTF
PSIA=PCAL
RETURN


5
QUAL=2.0
VOLUME
ENTHAL
ENTROP
PRINTF
RETURN
6
QUAL=-2.0
VOLUME
DRVPT
ENTHAL
ENTROP
FLQDEN
DELTH=RJ*DPSDT*(V-VL)*T
H=H-DELTH
S=S-DELTH/T
V=VL
PRINTF
RETURN
7
COMPUTE:
QUAL
H
V
PRINTF
RETURN
8
COMPS
ENTHAL
DRVPT
DELTH=RJ*DPSDT*(V-VL)*T
H=H-DELTH
QUAL=-2.0
V=VL
PRINTF
RETURN


9
COMPH
ENTROP
DRVPT
COMPUTE:
DELTH
S
Q
V=VL
PRINTF
RETURN


10
VPEQM
VOLUME
DRVPT
ENTHAL
ENTROP
QUAL=-2.0
COMPUTE:
DELTH
S
H
PRINTF
RETURN

```
C     ***********************************************************************
C     *                                                                     *
C     *                            F R E O N                                *
C     *                                                                     *
C     *    THIS SUBROUTINE CALCULATES THE THERMODYNAMIC PROPERTIES OF       *
C     *FREONS BY USING THE MARTIN-HOU EQUATION OF STATE. THE TYPES OF       *
C     *STATES WHICH CAN BE EVALUATED ARE, GIVEN THE FOLLOWING PAIRS OF      *
C     *KNOWNS:                                                              *
C     *                 1) TEMPERATURE-PRESSURE                             *
C     *                 2) TEMPERATURE-QUALITY                              *
C     *                 3) PRESSURE-ENTROPY                                 *
C     *                 4) PRESSURE-ENTHALPY                                *
C     *                 5) PRESSURE-QUALITY                                 *
C     *                 6) TEMPERATURE-SPECIFIC VOLUME                      *
C     *    COMPRESSED LIQUID STATES ARE APPROXIMATED BY SATURATED           *
C     *LIQUID STATES. THIS PROGRAM WAS WRITTEN BY DAVID H. RIEMER           *
C     *                                                                     *
C     ***********************************************************************
      SUBROUTINE FREON(CYCLE,FGIVEN,SGIVEN,FTYPE,H,S,V,T,PSIA,
     1    HV,SV,VV,HL,SL,VL,QUAL,NPRT)
      PARAMETER M2=6
      DIMENSION F113(4,12),F114(4,12),F11(4,12),F12(4,12),F13(4,12),
     1F14(4,12),F21(4,12),F22(4,12),F23(4,12)
      COMMON/FREON1/Q(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
      DATA LIMIT,DERROR,TTOLER,BTOLER/50,1.0E-10,1.005,0.995/
      DATA ((F113(I,J),I=1,4),J=1,12)/33.0655,-4330.98,-9.2635,
     1    0.0020539,0.0,0.0,0.0,0.0,0.0,0.05728,0.0,417.4,122.872,
     2    -4.035,0.002618,0.0,-0.0128,-0.0214,5.0E-05,0.0,-6.36E-05,0.0,
     3    0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.07963,1.159E-04,0.0,
     4    0.0,0.0,25.198,-0.40552,0.1699,459.6,109.49,3.0,0.02781,0.0,
     5    0.0,0.0,498.9/
      DATA((F114(I,J),I=1,4),J=1,12)/27.071306,-5113.7021,-6.3086761,
     1    6.913003E-04,0.76142111002,768.35,0.0,0.0,0.0,0.062780807,
     2    5.914907E-03,294.35,36.32,-2.3856704,1.0801207E-03,-6.5643648,
```

```
     3     0.021776,0.034055087,-5.3336494E-06,0.16366057,0.63649,
     4     -3.857481E-04,0.0,0.0,6.7186,1.6017659E-06,6.2632341E-10,
     5     -1.0165314E-05,1.97E-06,0.0,0.0,0.0,0.0175,3.49E-04,-1.67E-07,
     6     0.0,0.0,25.33968211,-0.1151371756,0.15842,459.6,95.4,2.0,
     7     0.027535,0.0,3.0,0.0,473.187/
      DATA((F11(I,J),I=1,4),J=1,12)/42.14702865,-4344.343807,
     1-12.84596733,0.004038372507,0.0313605356,862.07,0.0,0.0,0.0,
     20.078117,0.00190,388.47,34.57,-3.126759,1.318523E-03,-35.769990,
     357.03811,-0.025341,4.8751212E-05,1.220367,43.63220,1.687277E-03,
     4-1.805062E-06,0.0,-42.82356,-2.358930E-05,2.448303E-08,
     5-1.478379E-04,36.70603,1.057504E08,-9.472103E04,0.0,0.023815,
     62.798823E-04,-2.123734E-07,5.999018E-11,-336.807030,50.5418,
     7-0.0918395,0.17219,459.6,112.080,1.0,0.028927,0.0,4.50,580.00,
     8639.50/
      DATA((F12(I,J),I=1,4),J=1,12)/39.88381727,-3436.632228,
     1-12.47152228,0.004730442442,0.0,0.0,0.0,0.0,0.0,0.088734,
     20.00065093886,233.70,34.84,-3.409727134,0.00159434848,-56.7627671,
     30.02696,0.06023944634,-1.879618431E-05,1.311399084,0.834921,
     4-0.000548737007,0.0,0.0,6.02683,0.0,3.468834E-09,-2.54390678E-05,
     5-0.655549E-05,0.0,0.0,0.0,0.0080945,0.000332662,-2.413896E-07,
     66.72363E-11,0.0,-39.55655122,-0.0165379361,0.1359,459.6,78.86,2.0,
     70.02870,0.0,5.475,0.0,596.9/
      DATA((F13(I,J),I=1,4),J=1,12)/25.96797498,-2709.538217,
     1-7.172343913,0.00254515398,0.2803010913,546.00,0.0,0.0,0.0,
     20.102728,0.0048,84.00,38.06996128,-3.083417,2.341695E-03,
     3-18.212643,0.01566,0.058854,-5.671268E-05,0.571958,1.110,
     4-1.026061E-03,1.338679E-06,0.0,6.665,5.290649E-06,-7.395111E-9,
     5-3.874233E-05,3.245E-05,7.378601E07,-7.435565E04,0.0,0.01602,
     62.823E-04,-1.159E-07,0.0,0.0,20.911,-0.05676,0.08898,459.6,
     745.271,2.0,0.0277239,0.0,4.00,625.00,561.30/
      DATA((F14(I,J),I=1,4),J=1,12)/20.71545389,-2467.505285,
     1-4.69017025,0.00064798076,0.770707795,424.0,0.0,0.0,0.0,0.1219336,
     20.0015,-50.1,39.06,-2.162959,2.135114E-03,-18.941131,69.56848907,
     34.404057E-03,1.282818E-05,0.539776,4.58661139,1.921072E-04,
     4-3.918263E-07,0.0,36.17166615,-4.481049E-06,9.062318E-09,
```

```
     5-4.856678E-05,-8.05898583,5.838823E07,-9.263923E04,0.0,
     63.00559262E-02,2.37043352E-04,-2.85660077E-08,-2.95338806E-11,0.0,
     766.102162,0.36172528,0.5284,459.6,100.636,1.0,0.02560163,0.0,4.00,
     8661.199997,543.16/
      DATA((F21(I,J),I=1,4),J=1,12)/42.7908,-4261.34,-13.0295,0.0039851,
     10.0,0.0,0.0,0.0,0.0,0.10427,0.0,812.9,116.37962,-7.316,0.00464210,
     20.0,-0.03106808,-0.20382376,0.0003593,0.0,-0.0000501,0.0,0.0,0.0,
     30.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0427,0.000140,0.0,0.0,0.0,0.0,
     40.0,0.1906,459.6,120.45,3.0,0.030675,0.0,0.0,0.0,750.0/
      DATA((F22(I,J),I=1,4),J=1,12)/29.35754453,-3845.193152,
     1-7.86103122,0.002190939044,0.445746703,666.1,0.0,0.0,0.0,0.124098,
     20.002,204.81,32.76,-4.353547,0.002407252,-44.066868,54.6344093,
     3-0.017464,7.62789E-05,1.483763,36.74892,0.002310142,-3.605723E-06,
     40.0,-22.2925657,-3.724044E-05,5.355465E-08,-1.845051E-04,
     520.47328662,1.363387E08,-1.672612E05,0.0,0.02812836,2.255408E-04,
     6-6.509607E-08,0.0,257.341,62.4009,-0.0453335,0.16016,459.69,
     791.329,1.0,0.030525,0.0,4.2,548.2,721.906/
      DATA((F23(I,J),I=1,4),J=1,12)/328.90853,-7952.76913,-144.5142304,
     10.2421150182,0.0,0.0,0.0,-2.12806524E-04,9.43495542E-08,0.153270,
     20.00125,78.73,32.7758,-4.679499,3.472778E-03,-159.775232,63.37784,
     3-0.012475,7.733388E-05,5.941212,-25.30533,2.068042E-03,
     4-3.684238E-06,0.0,144.16182,-3.868546E-05,6.455643E-08,
     5-7.394214E-04,-106.13280,7.502357E07,-1.114202E05,0.0,0.07628087,
     6-7.561805E-06,3.9065696E-07,-2.454905E-10,0.0,0.0,0.0,0.1198,
     7459.6,60.77,1.0,0.030510,0.0,5.50,520.00,701.42/
      RJ=0.185053
      H=0.0
      S=0.0
      V=0.0
      T=0.0
      PSIA=0.0
      HV=0.0
      SV=0.0
      VV=0.0
      HL=0.0
```

```
      SL=0.0
      VL=0.0
      QUAL=0.0
      IF(FTYPE.EQ.'F113') GO TO 10
      IF(FTYPE.EQ.'F114') GO TO 20
      IF(FTYPE.EQ.'F11')GO TO 30
      IF(FTYPE.EQ.'F12')GO TO 40
      IF(FTYPE.EQ.'F13')GO TO 50
      IF(FTYPE.EQ.'F14')GO TO 60
      IF(FTYPE.EQ.'F21')GO TO 70
      IF(FTYPE.EQ.'F22')GO TO 80
      IF(FTYPE.EQ.'F23')GO TO 90
      WRITE(M2,9060) FTYPE
      RETURN
   10 DO 15 I=1,4
      DO 15 J=1,12
      Q(I,J)=F113(I,J)
   15 CONTINUE
      GO TO 350
   20 DO 25 I=1,4
      DO 25 J=1,12
      Q(I,J)=F114(I,J)
   25 CONTINUE
      GO TO 350
   30 DO 35 I=1,4
      DO 35 J=1,12
      Q(I,J)=F11(I,J)
   35 CONTINUE
      GO TO 350
   40 DO 45 I=1,4
      DO 45 J=1,12
      Q(I,J)=F12(I,J)
   45 CONTINUE
      GO TO 350
   50 DO 55 I=1,4
```

```
      DO 55 J=1,12
      Q(I,J)=F13(I,J)
   55 CONTINUE
      GO TO 350
   60 DO 65 I=1,4
      DO 65 J=1,12
      Q(I,J)=F14(I,J)
   65 CONTINUE
      GO TO 350
   70 DO 75 I=1,4
      DO 75 J=1,12
      Q(I,J)=F21(I,J)
   75 CONTINUE
      GO TO 350
   80 DO 85 I=1,4
      DO 85 J=1,12
      Q(I,J)=F22(I,J)
   85 CONTINUE
      GO TO 350
   90 DO 95 I=1,4
      DO 95 J=1,12
      Q(I,J)=F23(I,J)
   95 CONTINUE
  350 NTYPE=Q(3,11)
      HCRIT=Q(2,11)
      SCRIT=Q(4,10)
      PCRIT=Q(4,12)
      TCRIT=Q(4,3)
      IF(CYCLE.EQ.'TP') GO TO 400
      IF(CYCLE.EQ.'TX') GO TO 600
      IF(CYCLE.EQ.'PX') GO TO 700
      IF(CYCLE.EQ.'PS') GO TO 900
      IF(CYCLE.EQ.'PH') GO TO 1000
      IF(CYCLE.EQ.'TV') GO TO 1100
      WRITE(M2,9000)
```

```
      RETURN
  400 TF=FGIVEN
      PSIA=SGIVEN
      T=TF+Q(1,11)
      IF(TF.LT.Q(4,3)) GO TO 430
      V=1.0/Q(1,4)
      GO TO 480
  430 CALL VPEQM(PSAT,T,1)
      IF(PSIA/PSAT.GT.TTOLER.OR.PSIA/PSAT.LT.BTOLER) GO TO 470
      WRITE(M2,9010)
      RETURN
  470 IF(PSIA/PSAT.GT.TTOLER) GO TO 500
  480 QUAL=2.0
      CALL VOLUME(T,PSIA,V)
      CALL ENTHAL(PSIA,T,V,H,DHDT,DHDV)
      CALL ENTROP(T,V,S,DSDT,DSDV)
      CALL PRINTF(T,PSIA,QUAL,V,H,S,0.0,0.0,0.0,0.0,0.0,0.0,FTYPE,NPRT)
      RETURN
  500 CALL VOLUME(T,PSAT,V)
      CALL DRVPT(DPSDT,PSAT,T,DPSDT2)
      CALL ENTHAL(PSAT,T,V,H,DHDT,DHDV)
      CALL ENTROP(T,V,S,DSDT,DSDV)
      CALL FLQDEN(VL,T,DVLDT)
      DELTH=RJ*DPSDT*(V-VL)*T
      H=H-DELTH
      S=S-DELTH/T
      QUAL=-2.0
      CALL PRINTF(T,PSIA,QUAL,VL,H,S,0.0,0.0,0.0,0.0,0.0,0.0,FTYPE,NPRT)
      V=VL
      RETURN
  600 TF=FGIVEN
      QUAL=SGIVEN
      IF(QUAL.LE.1.0.AND.QUAL.GE.0.0) GO TO 610
      WRITE(M2,9070) QUAL
      RETURN
```

```
  910 CALL VPEQN(PSIA,T,2)
      CALL VOLUME(T,PSIA,V)
      VV=V
      CALL SATN(PSIA,T,VV,HV,SV,HL,SL,VL)
      IF(S/SV.GT.TTOLER) GO TO 940
      IF(S/SL.LT.BTOLER) GO TO 960
      QUAL=(S-SL)/(SV-SL)
      H=(1.0-QUAL)*HL+QUAL*HV
      V=(1.0-QUAL)*VL+QUAL*VV
      CALL PRINTF( T,PSIA,QUAL,V,H,S,VV,HV,SV,VL,HL,SL,FTYPE,NPRT)
      RETURN
  940 CALL SFIND(T,PSIA,V,S)
      CALL ENTHAL(PSIA,T,V,H,DHDT,DHDV)
      QUAL=2.0
      CALL PRINTF(T,PSIA,QUAL,V,H,S,0.0,0.0,0.0,0.0,0.0,0.0,FTYPE,NPRT)
      RETURN
  960 P=PSIA
      CALL COMPS(P,T,V,S,VL)
      CALL ENTHAL(P,T,V,H,DHDT,DHDV)
      CALL DRVPT(DPSDT,P,T,DPSDT2)
      DELTH=RJ*DPSDT*(V-VL)*T
      H=H-DELTH
      QUAL=-2.0
      CALL PRINTF(T,PSIA,QUAL,VL,H,S,0.0,0.0,0.0,0.0,0.0,0.0,FTYPE,NPRT)
      V=VL
      RETURN
 1000 PSIA=FGIVEN
      H=SGIVEN
      IF(PSIA.LT.Q(4,12)) GO TO 1020
      IF(H.GT.HCRIT) GO TO 1010
      WRITE(M2,9050)
      RETURN
 1010 T=Q(1,11)+Q(4,3)+10.0
      V=1.0/Q(4,11)
      GO TO 1040
```

```
  610 T=TF+Q(1,11)
      IF(TF.LE.Q(4,3)) GO TO 630
      WRITE(M2,9020)TCRIT
      RETURN
  630 CALL VPEQM(PSIA,T,1)
      GO TO 800
  700 PSIA=FGIVEN
      QUAL=SGIVEN
      IF(QUAL.LE.1.0.AND.QUAL.GE.0.0) GO TO 710
      WRITE(M2,9070) QUAL
      RETURN
  710 IF(PSIA.LE.Q(4,12)) GO TO 730
      WRITE( M2,9030) PCRIT
      RETURN
  730 CALL VPEQM(PSIA,T,2)
      TF=T-Q(1,11)
  800 IF(TF-Q(4,3)) 830,810,810
  810 V=1.0/Q(4,11)
      GO TO 840
  830 CALL VOLUME(T,PSIA,VV)
  840 CALL SATN(PSIA,T,VV,HV,SV,HL,SL,VL)
      H=(1.0-QUAL)*HL+QUAL*HV
      S=(1.0-QUAL)*SL+QUAL*SV
      V=(1.0-QUAL)*VL+QUAL*VV
      CALL PRINTF( T,PSIA,QUAL,V,H,S,VV,HV,SV,VL,HL,SL,FTYPE,NPRT)
      RETURN
  900 PSIA=FGIVEN
      S=SGIVEN
      IF(PSIA.LT.Q(4,12)) GO TO 910
      IF(S.LE.SCRIT) GO TO 905
      T=Q(4,3)+Q(1,11)
      V=1.0/Q(4,11)
      GO TO 940
  905 WRITE(M2,9040)
      RETURN
```

```
 1020 CALL VPEGM(PSIA,T,2)
      CALL VOLUME(T,PSIA,V)
      VV=V
      CALL SATN(PSIA,T,VV,HV,SV,HL,SL,VL)
      IF(H/HV.GT.TTOLER) GO TO 1040
      IF(H/HL.LT.BTOLER) GO TO 1060
      QUAL=(H-HL)/(HV-HL)
      S=(1.0-QUAL)*SL+QUAL*SV
      V=(1.0-QUAL)*VL+QUAL*VV
      CALL PRINTF( T,PSIA,QUAL,V,H,S,VV,HV,SV,VL,HL,SL,FTYPE,NPRT)
      RETURN
 1040 CALL HFIND(T,PSIA,V,H)
      CALL ENTROP(T,V,S,DSDT,DSDV)
      QUAL=2.0
      CALL PRINTF(T,PSIA,QUAL,V,H,S,0.0,0.0,0.0,0.0,0.0,0.0,FTYPE,NPRT)
      RETURN
 1060 P=PSIA
      CALL COMPH(P,T,V,H,VL)
      CALL ENTROP(T,V,S,DSDT,DSDV)
      CALL DRVPT(DPSDT,P,T,DPSDT2)
      DELTH=RJ*DPSDT*(V-VL)
      S=S-DELTH
      QUAL=-2.0
      CALL PRINTF(T,PSIA,QUAL,VL,H,S,0.0,0.0,0.0,0.0,0.0,0.0,FTYPE,NPRT)
      V=VL
      RETURN
 1100 TF=FGIVEN
      T=TF+G(1,11)
      V=SGIVEN
      IF(T.GE.TCRIT) GO TO 1140
      CALL VPEGM(PSAT,T,1)
      CALL FLQDEN(VL,T,DVLDT)
      CALL VOLUME(T,PSAT,VV)
      IF(V/VL.LT.BTOLER) GO TO 1160
      IF(V/VV.GT.TTOLER) GO TO 1140
```

```
      CALL SATN(PSAT,T,VV,HV,SV,HL,SL,VL)
      QUAL=(V-VL)/(VV-VL)
      H=(1.0-QUAL)*HL+QUAL*HV
      S=(1.0-QUAL)*SL+QUAL*SV
      CALL PRINTF(T,PSAT,QUAL,V,H,S,VV,HV,SV,VL,HL,SL,FTYPE,NPRT)
      PSIA=PSAT
      RETURN
 1140 CALL EOS(PCAL,T,V,DPDT,DPDV,DPDVDT,D2PDT2)
      CALL ENTHAL(PCAL,T,V,H,DHDT,DHDV)
      CALL ENTROP(T,V,S,DSDT,DSDV)
      QUAL=2.0
      CALL PRINTF(T,PCAL,QUAL,V,H,S,0.0,0.0,0.0,0.0,0.0,0.0,FTYPE,NPRT)
      PSIA=PCAL
      RETURN
 1160 CALL VPEQN(PSIA,T,1)
      CALL VOLUME(T,PSIA,VV)
      CALL DRVPT(DPSDT,PSIA,T,DPSDT2)
      CALL ENTHAL(PSIA,T,VV,H,DHDT,DHDV)
      CALL ENTROP(T,VV,S,DSDT,DSDV)
      QUAL=-2.0
      DELTH=RJ*DPSDT*(VV-V)
      S=S-DELTH
      H=H-DELTH*T
      CALL PRINTF(T,PSIA,QUAL,V,H,S,0.0,0.0,0.0,0.0,0.0,0.0,FTYPE,NPRT)
      RETURN
 9000 FORMAT(1H ,4X,'***ERROR*** CYCLE = ',A3,' CYCLE MUST EQUAL TP, TX,
     1 PX, PS, PH',//,5X,'FREON TERMINATED')
 9010 FORMAT(1H ,4X,'***ERROR*** TEMPERATURE AND PRESSURE ARE NOT INDEPE
     1NDENT UNDER THE SATURATION DOME',//,5X,'FREON TERMINATED')
 9020 FORMAT(1H ,4X,'***ERROR*** A SATURATION STATE DOES NOT EXIST FOR T
     1EMPERATURES ABOVE THE CRITICAL TEMPERATURE',F10.3,'DEG.F',//,5X,
     2    'FREON TERMINATED')
 9030 FORMAT(1H ,4X,'***ERROR*** A SATURATION STATE DOES NOT EXIST FOR P
     1RESSURES ABOVE THE CRITICAL PRESSURE',F10.3,'PSIA',//,5X,'FREON TE
     2RMINATED')
```

```
      9040 FORMAT(1H ,4X,'THE MARTIN-HOU EQUATION OF STATE IS NOT VALID IN TH
          1IS REGION,2.E. P>PC AND S;SC',//,5X,'FREON TERMINATED')
      9050 FORMAT(1H ,4X,'THE MARTIN HOU EQUATION OF STATE IS NOT VALID IN TH
          1IS REGION,2.E. P>PC AND H;HC',//,5X,'FREON TERMINATED')
      9060 FORMAT(1H ,4X,'***ERROR*** THE FLUID ',A4,' CAN NOT BE EVALUATED U
          1SING THE SUBROUTINE FREON',//,5X,'FREON TERMINATED')
      9070 FORMAT(1H ,4X,'***ERROR*** A QUALITY OF ',F6.3,'IS NOT ALLOWED',
          1//, 5X,'FREON TERMINATED')
           END
```

COMPH DESCRIPTION

COMPH searches for the temperature and specific volume in the compressed liquid region given pressure and enthalpy. The compressed liquid region is approximated by saturated liquid at the same temperature. A two variable Newton-Raphson iteration technique is used in the search.

EQUATIONS

$$F_p = P(V,T)-P_{SAT}(T)$$

$$F_h = h(V,T)-\text{CONVR}\frac{dP_{SAT}}{dT}(V-V_L)*T-h_{KNOWN}$$

$$\frac{\partial F_p}{\partial T} = \frac{\partial P(V,T)}{\partial T} - \frac{dP_{SAT}}{dT}$$

$$\frac{\partial F_h}{\partial T} = \frac{\partial h(V,T)}{\partial T} - \text{CONVR}\left(\frac{d^2P_{SAT}}{dT^2}(V-V_L)*T+\frac{dP_{SAT}}{dT}(V-V_L) -\left(\frac{dP_{SAT}}{dT}\right)\left(\frac{dV_L}{dT}\right)*T\right)$$

$$\frac{\partial F_h}{\partial V} = \frac{\partial h(V,T)}{\partial V} - \text{CONVR}*\frac{dP_{SAT}}{dT}*T$$

$$\text{Jacobian} = \left(\frac{\partial P(V,T)}{\partial V}\right)\left(\frac{\partial F_h}{\partial T}\right) - \left(\frac{\partial F_p}{\partial T}\right)\left(\frac{\partial F_h}{\partial T}\right)$$

$$V_{i+1} = V_i - \left[\frac{F_p*\frac{\partial F_h}{\partial T} - F_h*\frac{\partial F_h}{\partial T}}{\text{Jacobian}}\right]_i$$

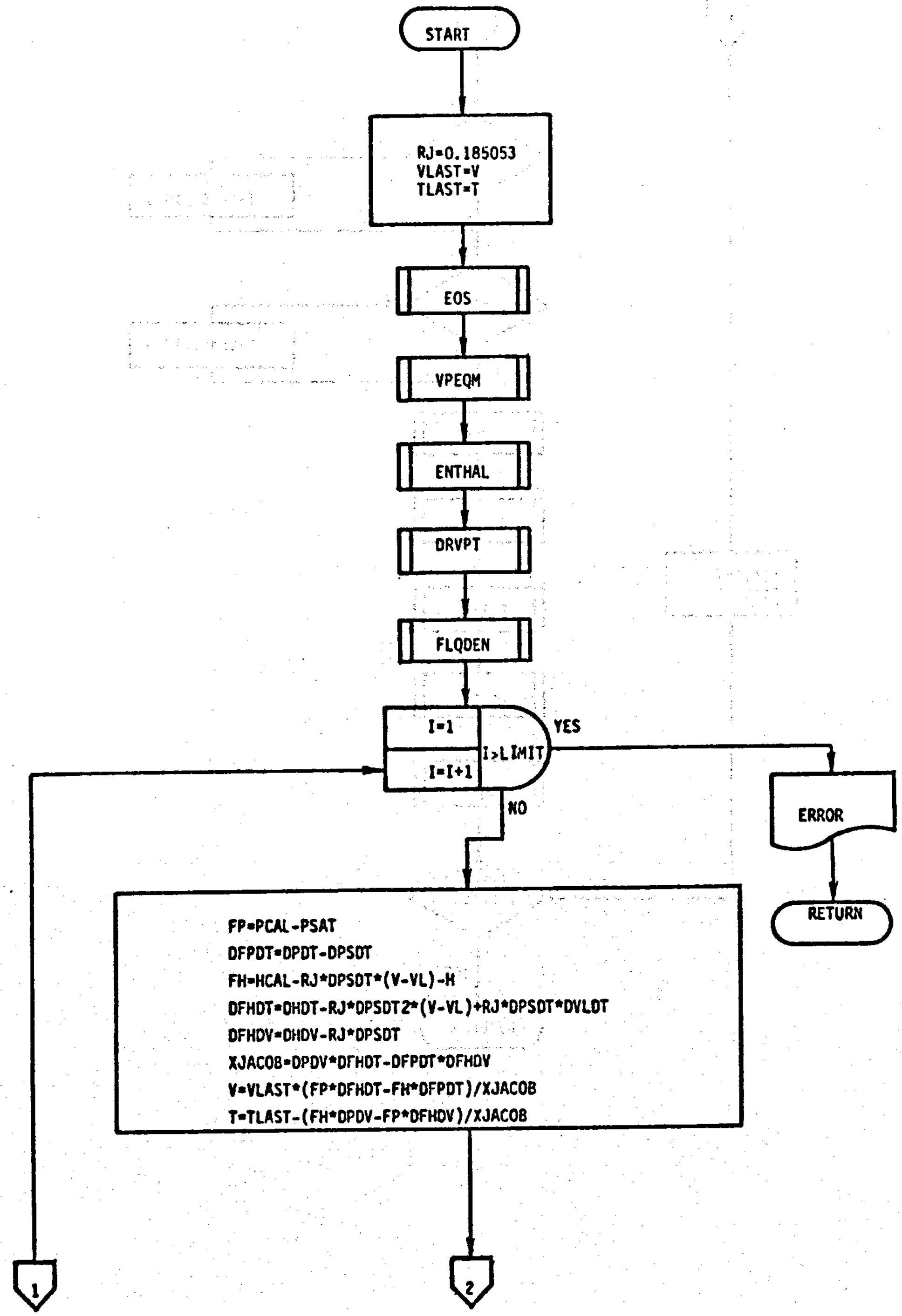
$$T_{i+1} = T_i - \left[\frac{F_h*\frac{\partial P}{\partial V} - F_p*\frac{\partial F_h}{\partial V}}{\text{Jacobian}}\right]_i$$

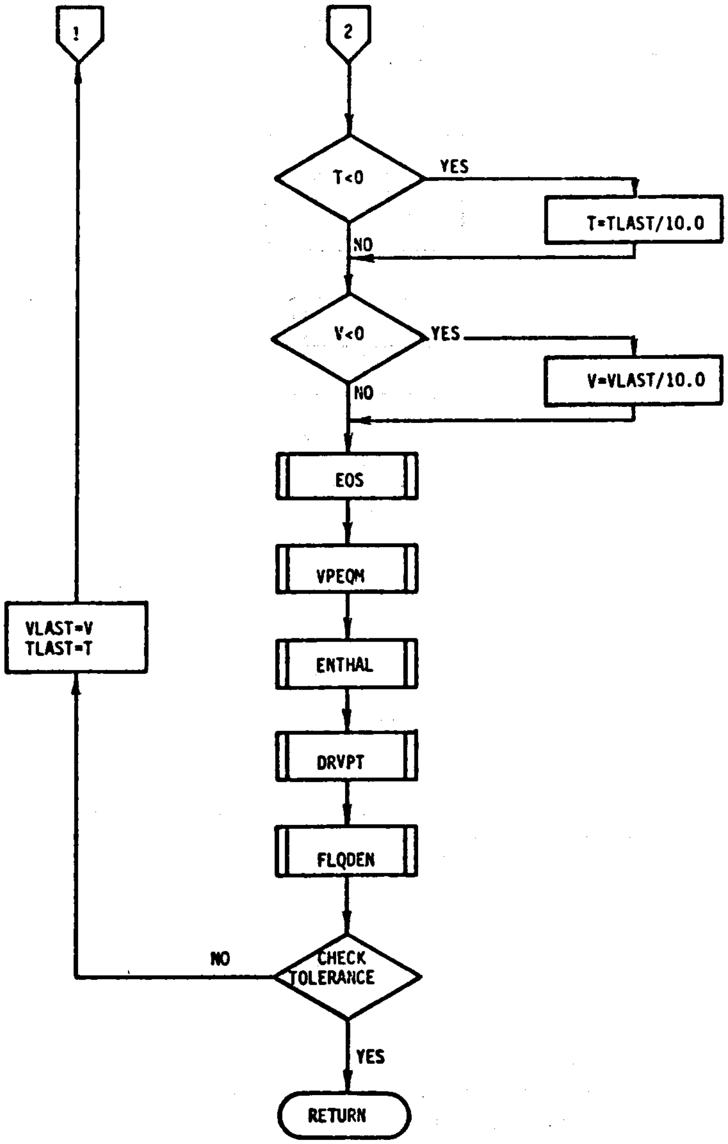
VARIABLE SYMBOL TABLE

BTOLER	lower tolerance on convergence and saturation tests
DFHDT	$\frac{\partial h(V,T)}{\partial T} - RJ*\frac{d^2P_{SAT}}{dT^2}*(V-V_L)*T - \frac{dP_{SAT}}{dT}*\frac{dV_L}{dT}*T+\frac{dP_{SAT}}{dT}(V-V_L)$
DFHDV	$\frac{\partial h(V,T)}{\partial V} - RJ*\frac{dP_{SAT}}{dT}*T$
DFPDT	$\frac{\partial P(V,T)}{\partial T} - \frac{dP_{SAT}}{dT}$

DHDT	$\frac{\partial h(V,T)}{\partial T}$
DHDV	$\frac{\partial h(V,T)}{\partial V}$
DP2DT2	$\frac{\partial^2 P(V,T)}{\partial T^2}$
DPDT	$\frac{\partial P(V,T)}{\partial T}$
DPDV	$\frac{\partial P(V,T)}{\partial V}$
DPDVDT	$\frac{\partial^2 P(V,T)}{\partial V \partial T}$
DPSDT	$\frac{dP_{SAT}}{dT}$
DPSDT2	$\frac{d^2 P_{SAT}}{dT^2}$
DVLDT	$\frac{dV_L}{dT}$
FH	h calculated - $RJ*\frac{dP_{SAT}}{dT}*(V-V_L)*T-h_{KNOWN}$
FP	P calculated - P_{KNOWN}
H	known enthalpy (Btu/lbm)
HCAL	calculated vapor enthalpy (Btu/lbm)
I	index number
LIMIT	maximum of iterations allowed in any one search
M2	unit number for printer
P	known pressure (psia)
PCAL	calculated pressure (psia)
PSAT	saturated pressure (psia)
RJ	conversion factor = 0.185053
T	temperature (°R)
TLAST	last temperature calculated, to be used in next iteration

TTOLER	upper tolerance in convergence and saturation tests
V	specific volume (ft^3/lbm)
VL	saturated liquid specific volume (ft^3/lbm)
VLAST	last specific volume calculated, to be used in next iteration
XJACOB	Jacobian of pressure and enthalpy


START
RJ=0.185053
VLAST=V
TLAST=T
EOS
VPEQM
ENTHAL
DRVPT
FLQDEN
I=1
I=I+1
I>LIMIT
YES
NO
ERROR
RETURN
FP=PCAL-PSAT
DFPDT=DPDT-DPSDT
FH=HCAL-RJ*DPSDT*(V-VL)-H
DFHDT=DHDT-RJ*DPSDT2*(V-VL)+RJ*DPSDT*DVLDT
DFHDV=DHDV-RJ*DPSDT
XJACOB=DPDV*DFHDT-DFPDT*DFHDV
V=VLAST*(FP*DFHDT-FH*DFPDT)/XJACOB
T=TLAST-(FH*DPDV-FP*DFHDV)/XJACOB
1
2


1
2
T<0
YES
T=TLAST/10.0
NO
V<0
YES
V=VLAST/10.0
NO
EOS
VPEQM
ENTHAL
DRVPT
FLQDEN
CHECK TOLERANCE
NO
YES
RETURN
VLAST=V
TLAST=T

```
C     *******************************************************************
C     *                                                                 *
C     *                           C O M P H                             *
C     *                                                                 *
C     *   THIS SUBROUTINE FINDS THE TEMPERATURE AND DENSITY GIVEN       *
C     *PRESSURE AND ENTHALPY FOR COMPRESSED LIQUID STATES.              *
C     *                                                                 *
C     *******************************************************************
      SUBROUTINE COMPH(P,T,V,H,VL)
      PARAMETER M2=6
      COMMON/FREON1/G(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
      RJ=0.185053
      VLAST=V
      TLAST=T
      CALL EOS(PCAL,T,V,DPDT,DPDV,DPDVDT,DP2DT2)
      CALL VPEQN(PSAT,T,1)
      CALL ENTHAL(PCAL,T,V,HCAL,DHDT,DHDV)
      CALL DRVPT(DPSDT,PSAT,T,DPSDT2)
      CALL FLQDEN(VL,T,DVLDT)
      DO 1000 I=1,LIMIT
      FP=PCAL-PSAT
      DFPDT=DPDT-DPSDT
      FH=HCAL-RJ*DPSDT*(V-VL)*T-H
      DFHDT=DHDT-RJ*(DPSDT2*(V-VL)*T-DPSDT*DVLDT*T+DPSDT*(V-VL))
      DFHDV=DHDV-RJ*DPSDT*T
      XJACOB=DPDV*DFHDT-DFPDT*DFHDV
      V=VLAST-(FP*DFHDT-FH*DFPDT)/XJACOB
      T=TLAST-(FH*DPDV-FP*DFHDV)/XJACOB
      IF(T.LE.0.0) T=TLAST/10.0
      IF(V.LE.0.0) V=VLAST/10.0
      CALL EOS(PCAL,T,V,DPDT,DPDV,DPDVDT,DP2DT2)
      CALL VPEQN(PSAT,T,1)
      CALL ENTHAL(PCAL,T,V,HCAL,DHDT,DHDV)
      CALL DRVPT(DPSDT,PSAT,T,DPSDT2)
```

```
      CALL FLUDEN(VL,T,DVLDT)
      HL=HCAL-RJ*DPSDT*(V-VL)*T
      IF(PCAL/PSAT.LT.TTOLER.AND.PCAL/PSAT.GT.BTOLER.AND.HL/H.LT.
     1    TTOLER.AND.HL/H.GT.BTOLER) GO TO 1050
      IF(ABS(T-TLAST).LT.1.0E-04.AND.ABS(V-VLAST).LT.1.0E-07) GO TO 1050
      VLAST=V
      TLAST=T
 1000 CONTINUE
      WRITE(M2,9000) LIMIT
 1050 P=PCAL
      RETURN
 9000 FORMAT(' ',10X,'***COMPH FAILED TO CONVERGE IN ',I5,' ITERATIONS')
      END
```

COMPS DESCRIPTION

COMPS searches for the temperature and specific volume in the compressed liquid region given pressure and entropy. The compressed liquid region is approximated by saturated liquid at the same temperature. A two variable Newton-Raphson iteration technique is used in the search.

EQUATIONS

$$F_p = P(V,T) - P_{SAT}(T)$$

$$F_S = S(V,T) - \text{CONVR}\,\frac{dP_{SAT}}{dT}\,(V-V_L)*T-S_{KNOWN}$$

$$\frac{\partial F_p}{\partial T} = \frac{\partial S(V,T)}{\partial T} - \frac{dP_{SAT}}{dT}$$

$$\frac{\partial F_S}{\partial T} = \frac{\partial S(V,T)}{\partial T} - \text{CONVR}\,\frac{d^2P_{SAT}}{dT^2}\,(V-V_L) + \text{CONVR}\,\left(\frac{dP_{SAT}}{dT}\right)\left(\frac{dV_L}{dT}\right)$$

$$\frac{\partial F_S}{\partial V} = \frac{\partial S(V,T)}{\partial V} - \text{CONVR}\,\frac{dP_{SAT}}{dT}$$

$$\text{Jacobian} = \left(\frac{\partial P(V,T)}{\partial V}\right)\left(\frac{\partial F_S}{\partial T}\right) - \left(\frac{\partial P(V,T)}{\partial T}\right)\left(\frac{\partial F_S}{\partial V}\right)$$

$$V_{i+1} = V_i - \left[\frac{F_p\left(\frac{\partial F_T}{\partial T}\right) - F_S\left(\frac{\partial F_p}{\partial T}\right)}{\text{Jacobian}}\right]_i$$

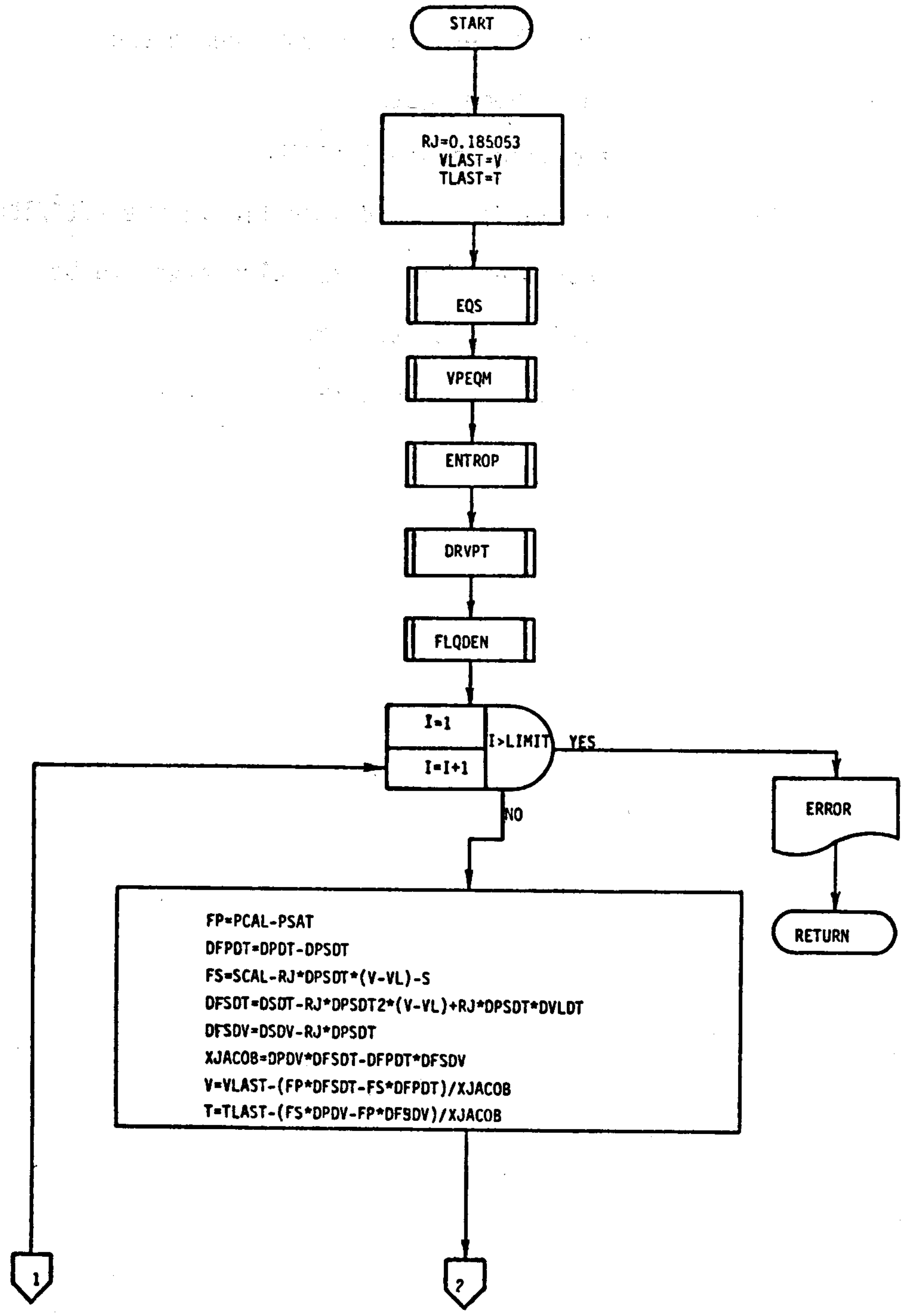
$$T_{i+1} = T_i - \left[\frac{F_S\left(\frac{\partial F_p}{\partial V}\right) - F_p\left(\frac{\partial F_S}{\partial V}\right)}{\text{Jacobian}}\right]_i$$

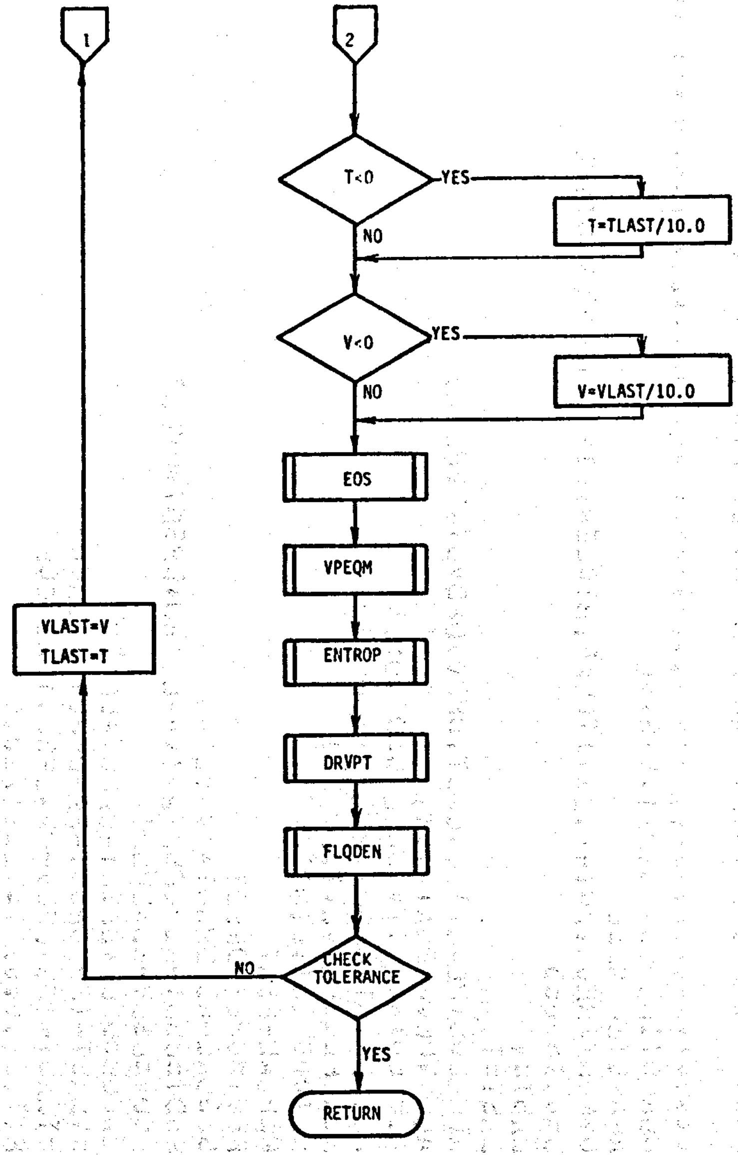
VARIABLE SYMBOL TABLE

BTOLER	lower tolerance on convergence and saturation tests
DFPDT	$\frac{d(P(V,T) - P_{SAT}(T))}{dT}$
DFSDT	$\frac{\partial S(V,T)}{\partial T} - RJ*\frac{dP_{SAT}(T)}{dT}*(V-V_L)$ $+RJ*\frac{dP_{SAT}(T)}{dT}*\frac{dV_L}{dT}$

DPSDV	$\frac{\partial S(V,T)}{\partial V} - RJ* \frac{dP_{SAT}(T)}{dT}$
DP2DT2	$\frac{\partial^2 P(V,T)}{\partial T^2}$
DPDT	$\frac{\partial P(V,T)}{\partial T}$
DPDV	$\frac{\partial P(V,T)}{\partial V}$
DPDVDT	$\frac{\partial^2 P(V,T)}{\partial V \partial T}$
DPSDT	$\frac{dP_{SAT}(T)}{dT}$
DPSDT2	$\frac{d^2 P_{SAT}(T)}{dT^2}$
DSDT	$\frac{\partial S(V,T)}{\partial T}$
DSDV	$\frac{\partial S(V,T)}{\partial V}$
DVLDT	$\frac{dV_L(T)}{dT}$
FP	P calculated - P_{GIVEN}
FS	S calculated - S_{GIVEN}
I	index
LIMIT	maximum number of iterations allowed in any one search
M2	unit number for printer
P	known pressure (psia)
PCAL	pressure calculated (psia)
PSAT	saturation pressure (psia)
RJ	conversion factor = 0.185053
S	known entropy (Btu/lbm°R)
SCAL	calculated entropy (Btu/lbm°R)

T	temperature (°R)
TLAST	last temperature calculated, to be used in next iteration
TTOLER	upper tolerance on convergence and saturation tests
V	specific volume (ft^3/lbm)
VL	saturation liquid specific volume (ft^3/lbm)
VLAST	last specific volume calculated, to be used in next iteration
XJACOB	Jacobian of pressure and entropy


START
RJ=0.185053
VLAST=V
TLAST=T
EQS
VPEQM
ENTROP
DRVPT
FLQDEN
I=1
I=I+1
I>LIMIT
YES
NO
ERROR
RETURN
FP=PCAL-PSAT
DFPDT=DPDT-DPSDT
FS=SCAL-RJ*DPSDT*(V-VL)-S
DFSDT=DSDT-RJ*DPSDT2*(V-VL)+RJ*DPSDT*DVLDT
DFSDV=DSDV-RJ*DPSDT
XJACOB=DPDV*DFSDT-DFPDT*DFSDV
V=VLAST-(FP*DFSDT-FS*DFPDT)/XJACOB
T=TLAST-(FS*DPDV-FP*DFSDV)/XJACOB
1
2


1
2
T<0
YES
NO
T=TLAST/10.0
V<0
YES
NO
V=VLAST/10.0
EOS
VPEQM
ENTROP
DRVPT
FLQDEN
VLAST=V
TLAST=T
CHECK TOLERANCE
NO
YES
RETURN

```
C     ******************************************************************
C     *                                                                *
C     *                            C O M P S                           *
C     *                                                                *
C     *   THIS SUBROUTINE FINDS THE TEMPERATURE AND DENSITY GIVEN      *
C     *PRESSURE AND ENTROPY FOR COMPRESSED LIQUID STATES.              *
C     *                                                                *
C     ******************************************************************
      SUBROUTINE COMPS(P,T,V,S,VL)
      PARAMETER M2=6
      COMMON/FREON1/Q(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
      RJ=0.185053
      VLAST=V
      TLAST=T
      CALL EOS(PCAL,T,V,DPDT,DPDV,DPDVDT,D2PDT2)
      CALL VPEQN(PSAT,T,1)
      CALL ENTROP(T,V,SCAL,DSDT,DSDV)
      CALL DRVPT(DPSDT,PSAT,T,DPSDT2)
      CALL FLQDEN(VL,T,DVLDT)
      DO 1000 I=1,LIMIT
      FP=PCAL-PSAT
      DFPDT=DPDT-DPSDT
      FS=SCAL-RJ*DPSDT*(V-VL)-S
      DFSDT=DSDT-RJ*DPSDT2*(V-VL)+RJ*DPSDT*DVLDT
      DFSDV=DSDV-RJ*DPSDT
      XJACOB=DPDV*DFSDT-DFPDT*DFSDV
      V=VLAST-(FP*DFSDT-FS*DFPDT)/XJACOB
      T=TLAST-(FS*DPDV-FP*DFSDV)/XJACOB
      IF(T.LE.0.0) T=TLAST/10.0
      IF(V.LE.0.0) V=VLAST/10.0
      CALL EOS(PCAL,T,V,DPDT,DPDV,DPDVDT,D2PDT2)
      CALL VPEQN(PSAT,T,1)
      CALL ENTROP(T,V,SCAL,DSDT,DSDV)
      CALL DRVPT(DPSDT,PSAT,T,DPSDT2)
```

```
      CALL FLQDER(VL,T,DV,DT)
      SL=SCAL-RO*(V-VL)*DPSDT
      IF(PCAL/PSAT.LT.TTOLER.AND.PCAL/PSAT.GT.BTOLER.AND.SL/S.LT.
     1    TTOLER.AND.SL/S.GT.BTOLER) GO TO 1050
      IF(ABS(T-TLAST).LT.1.0E-04.AND.ABS(V-VLAST).LT.1.0E-07) GO TO 1050
      VLAST=V
      TLAST=T
 1000 CONTINUE
      WRITE(M2,9000) LIMIT
 1050 P=PCAL
      RETURN
 9000 FORMAT(' ',10X,'***COMPS FAILED TO CONVERGE IN ',I5,' ITERATIONS')
      END
```

DRVPT DESCRIPTION

DRVPT calculates the first and second derivative of the saturation pressure with respect to temperature.

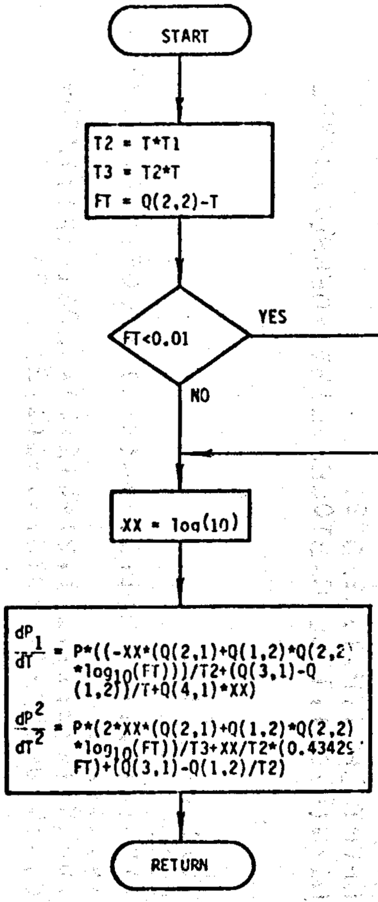
EQUATIONS

$$\frac{dP_{SAT}}{dT} = P_{SAT}\ (^{-}(-\frac{B+E*F\ LOG_{10}(F-T))\ln(10)}{T^2} + \frac{C-E}{T} + D\ln(10))$$

$$\frac{d^2P_{SAT}}{dT^2} = P_{SAT}\ (\frac{2((B+EF\ LOG_{10}(F-T))\ln(10)}{T^3} + \frac{1}{T^2(F-T)} + \frac{C-E}{T^2})$$

VARIABLE SYMBOL TABLE

DPDT	$\frac{dP_{SAT}}{dT}$
D2PDT2	$\frac{d^2P_{SAT}}{dT^2}$
FT	F-T
P	saturation pressure (psia)
Q	2-dimensional array containing the coefficients for the fluid to be evaluated
T	temperature (°R)
T2	T^2
T3	T^3
XX	$\ln(10)$


START
T2 = T*T1
T3 = T2*T
FT = Q(2,2)-T
FT<0.01
YES
NO
FT = 1.0
XX = log(10)
dP1/dT = P*((-XX*(Q(2,1)+Q(1,2)*Q(2,2)
*log10(FT)))/T2+(Q(3,1)-Q
(1,2))/T+Q(4,1)*XX)
dP2/dT2 = P*(2*XX*(Q(2,1)+Q(1,2)*Q(2,2)
log10(FT))/T3+XX/T2(0.4342S
FT)+(Q(3,1)-Q(1,2)/T2)
RETURN

```
C     ******************************************************************
C     *                                                                *
C     *                           D R V P T                            *
C     *                                                                *
C     *   THIS SUBROUTINE CALCULATES THE DERIVATIVE OF THE SATURATION  *
C     *PRESSURE WITH RESPECT TO TEMPERATURE                            *
C     *                                                                *
C     ******************************************************************
      SUBROUTINE DRVPT(DPDT,P,T,D2PDT2)
      COMMON/FREON1/Q(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
      T2 = T*T
      T3=T2*T
      FT=Q(2,2)-T
      IF (FT.LT.0.01) FT = 1.0
      XX = ALOG(10.0)
      DPDT = P*((-XX*(Q(2,1) + Q(1,2) *Q(2,2) *ALOG10(FT)))/T2+(Q(3,1)-Q
     1(1,2)) /T+Q(4,1)*XX)
      D2PDT2=P*(2.0*XX*(Q(2,1)+Q(1,2)*Q(2,2)*ALOG10(FT))/T3+
     1    XX/T2*(0.43429448/FT)+(Q(3,1)-Q(1,2))/T2)
      RETURN
      END
```

ENTHAL DESCRIPTION

ENTHAL calculates the enthalpy and its derivatives with respect to temperature and specific volume for vapor states using the Downing-Martin equation of state.

EQUATIONS

$$h = A*T+b*T^2/2+C*T^3/3+D*T^4/4-F/T+CONVR*P*V+CONVR\ (\frac{A_2}{(V-\beta)}$$

$$+\frac{A_3}{2(V-\beta)^2}+\frac{A_4}{3(V-\beta)^3}+\frac{A_5}{4(V-\beta)^4}+\frac{A_6}{\alpha}\ (\frac{1}{\exp(\alpha*V)}-C_1\ln(\frac{C_1\exp(\alpha*V)+1}{C_1\exp(\alpha*V)})))$$

$$+\ CONVR\ \exp(-K*T/TC)(1+K*T/TC)(\frac{C_2}{V-\beta}+\frac{C_3}{2(V-\beta)^2}+\frac{C_4}{3(V-\beta)^3}$$

$$+\frac{C_5}{4(V-\beta)^4}+\frac{C_6}{\alpha\exp(\alpha*V)}-\frac{C_6*C_1}{\alpha}\ln(\frac{C_1\exp(\alpha*V)+1}{C_1\exp(\alpha*V)}))$$

$$+\ \Delta h(\text{latent at }-40°\text{F})-h(\text{saturated vapor at }-40°\text{F})$$

$$\frac{\partial h(V,T)}{\partial T}=a+b*T+C*T^2+d*T^3+f/T^2+CONVR*V*\ \frac{\partial P(V,T)}{\partial T}$$

$$-\ \frac{CONVR*K^2*T}{T_C{}^2}\exp(-K*T/TC)(\frac{C_2}{V-\beta}+\frac{C_3}{2(V-\beta)^2}+\frac{C_4}{3(V-\beta)^3}+\frac{C_5}{4(V-\beta)^4}$$

$$+\frac{C_6}{\alpha\exp(\alpha*V)}-\frac{C_6*C_1}{\alpha}\ln(\frac{C_1\exp(\alpha*V)+1}{C_1\exp(\alpha*V)}))$$

$$\frac{\partial h(V,T)}{\partial V}=CONVR*P+CONVR*V*\frac{\partial P(V,T)}{\partial V}+CONVR(\frac{-A_2}{(V-\beta)^2}-\frac{A_3}{(V-\beta)^3}-\frac{A_4}{(V-\beta)^4}$$

$$-\frac{A_5}{(V-\beta)^5}+A_6\ (\frac{-1}{\exp(\alpha*V)}+\frac{C_1}{C_1\exp(\alpha*V)+1}))+\ CONVR*\exp(-K*T/TC)$$

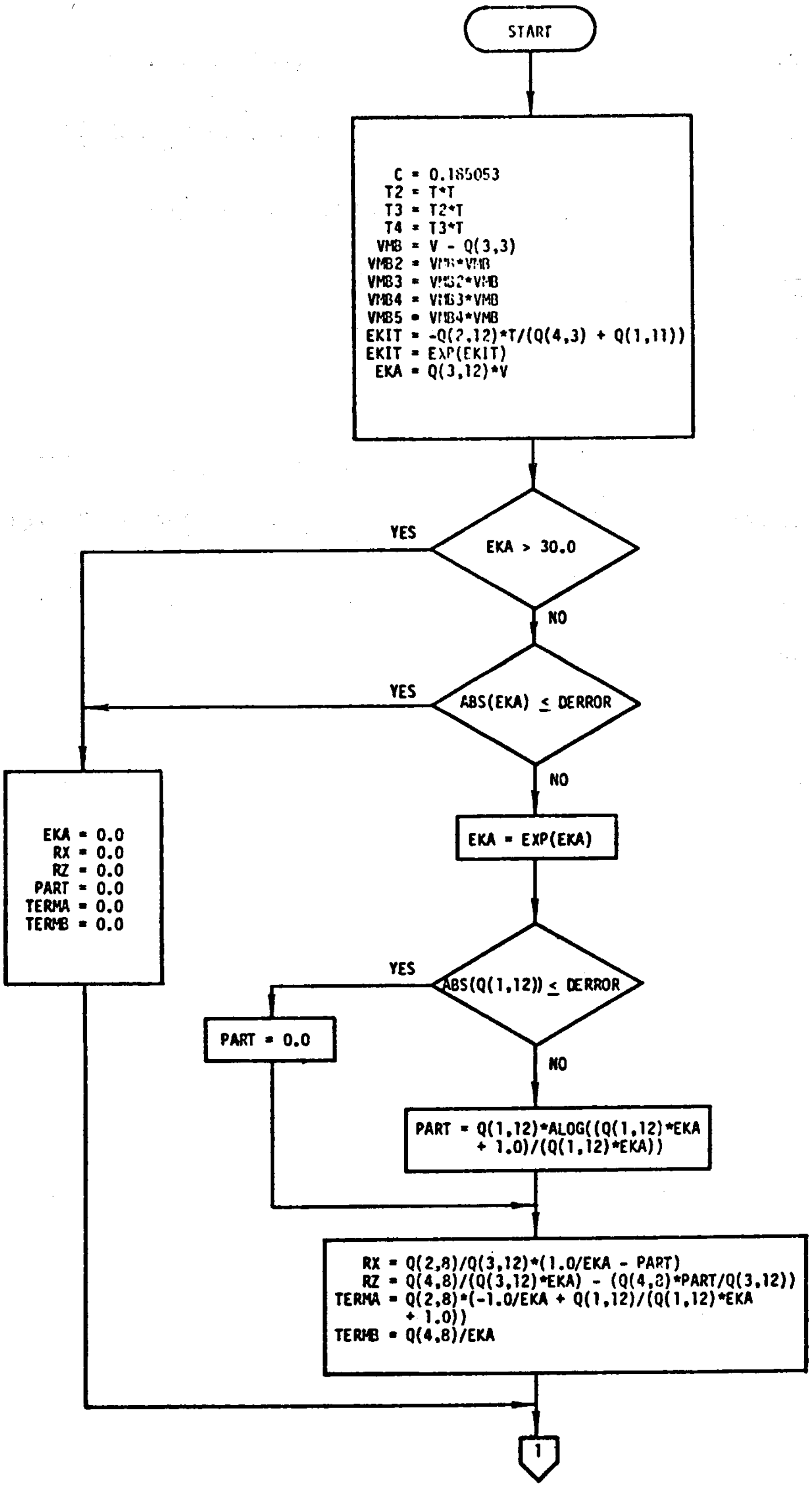
$$(-\frac{C_2}{(V-\beta)^2}-\frac{C_3}{(V-\beta)^3}-\frac{C_4}{(V-\beta)^4}-\frac{C_5}{(V-\beta)^5}-\frac{C_6}{\exp(\alpha*V)}+\frac{C_6*C_L}{C_1\exp(\alpha*V)+1})$$

VARIABLE SYMBOL TABLE

C	conversion factor = 0.185053
DERROR	tolerance used in checks for zero
DHDT	$\frac{\partial h(V,T)}{\partial T}$

DHDV	$\frac{\partial h(V,T)}{\partial V}$
DPDT	$\frac{\partial P(V,T)}{\partial T}$
DPDV	$\frac{\partial P(V,T)}{\partial V}$
D2PDT2	$\frac{\partial^2 P(V,T)}{\partial T^2}$
DPDVDT	$\frac{\partial^2 P(V,T)}{\partial V \partial T}$
EKA	$\exp(\alpha * V)$
EKIT	$\exp(-K*T/TC)$
H	enthalpy (Btu/lbm)
H1	$a*T+b*T^2/2+C*T^3/3*d*T^4/4-f/T+CONVR*P*V$
H2	$CONVR(\frac{A_2}{V-\beta}+\frac{A_3}{2(V-\beta)^2}+\frac{A_4}{3(V-\beta)^3}+\frac{A_5}{4(V-\beta)^4}+RX)$
H3	$CONVR(\frac{C_2}{V-\beta}+\frac{C_3}{2(V-\beta)^2}+\frac{C_4}{3(V-\beta)^3}$
	$+\frac{C_5}{4(V-\beta)^4}+RZ)(1+K*T/TC)(\exp(-K*T/TC))$
P	pressure (psia)
PART	$C_1 \ln(\frac{C_1 \exp(\alpha * V)+1}{C_1 \exp(\alpha * V)})$
PART1	$a+b*T+C*T^2+d*T^3+f/T^2+CONVR*\frac{\partial P(V,T)}{\partial T}*V$
PART2	$-(\frac{C_2}{V-\beta}+\frac{C_3}{2(V-\beta)^2}+\frac{C_4}{3(V-\beta)^3}+\frac{C_5}{4(V-\beta)^4}$
	$+RZ)(\frac{CONVR*K^2*\exp(-K*T/TC)}{T_C^2})$
Q	2-dimensional array which contains the coefficients for the fluid to be evaluated

RX	$A_6(1/\exp(\alpha*V)-PART)/\alpha$
RZ	$(C_6/\exp(\alpha*V)-C_6*PART)/\alpha$
T	temperature (°R)
T2	T^2
T3	T^3
T4	T^4
TC	critical temperature (°R)
TERMA	$A_6(-1/\exp(\alpha*V)+C_1)/(C_1\exp(\alpha*V)+1)$
V	specific volume (ft^3/lbm)
VMB	$V-\beta$
VMB2	$(V-\beta)^2$
VMB3	$(V-\beta)^3$
VMB4	$(V-\beta)^4$
VMB5	$(V-\beta)^5$



```
   C = 0.185053
  T2 = T*T
  T3 = T2*T
  T4 = T3*T
 VMB = V - Q(3,3)
VMB2 = VMB*VMB
VMB3 = VMB2*VMB
VMB4 = VMB3*VMB
VMB5 = VMB4*VMB
EKIT = -Q(2,12)*T/(Q(4,3) + Q(1,11))
EKIT = EXP(EKIT)
 EKA = Q(3,12)*V
```

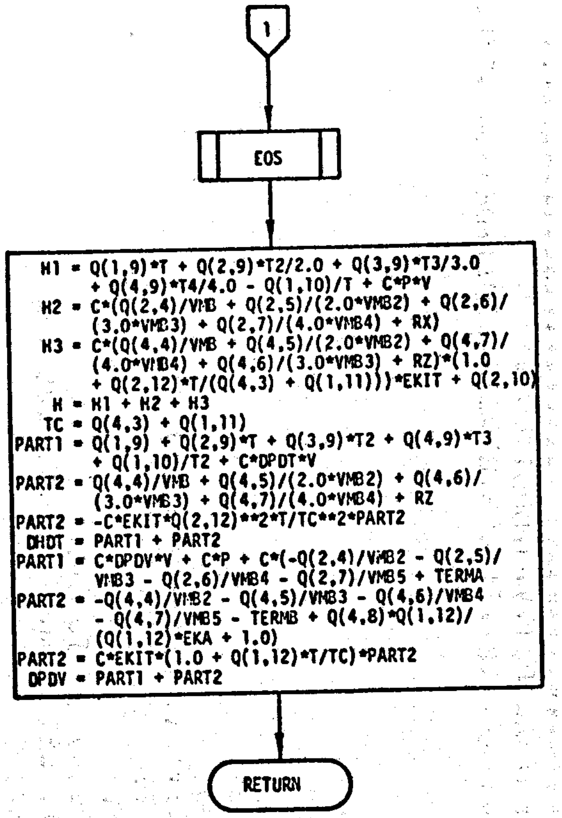
```
  EKA = 0.0
   RX = 0.0
   RZ = 0.0
 PART = 0.0
TERMA = 0.0
TERMB = 0.0
```

```
EKA = EXP(EKA)
```

```
PART = 0.0
```

```
PART = Q(1,12)*ALOG((Q(1,12)*EKA
       + 1.0)/(Q(1,12)*EKA))
```

```
   RX = Q(2,8)/Q(3,12)*(1.0/EKA - PART)
   RZ = Q(4,8)/(Q(3,12)*EKA) - (Q(4,8)*PART/Q(3,12))
TERMA = Q(2,8)*(-1.0/EKA + Q(1,12)/(Q(1,12)*EKA
        + 1.0))
TERMB = Q(4,8)/EKA
```



```
   H1 = Q(1,9)*T + Q(2,9)*T2/2.0 + Q(3,9)*T3/3.0
        + Q(4,9)*T4/4.0 - Q(1,10)/T + C*P*V
   H2 = C*(Q(2,4)/VMB + Q(2,5)/(2.0*VMB2) + Q(2,6)/
        (3.0*VMB3) + Q(2,7)/(4.0*VMB4) + RX)
   H3 = C*(Q(4,4)/VMB + Q(4,5)/(2.0*VMB2) + Q(4,7)/
        (4.0*VMB4) + Q(4,6)/(3.0*VMB3) + RZ)*(1.0
        + Q(2,12)*T/(Q(4,3) + Q(1,11)))*EKIT + Q(2,10)
    H = H1 + H2 + H3
   TC = Q(4,3) + Q(1,11)
PART1 = Q(1,9) + Q(2,9)*T + Q(3,9)*T2 + Q(4,9)*T3
        + Q(1,10)/T2 + C*DPDT*V
PART2 = Q(4,4)/VMB + Q(4,5)/(2.0*VMB2) + Q(4,6)/
        (3.0*VMB3) + Q(4,7)/(4.0*VMB4) + RZ
PART2 = -C*EKIT*Q(2,12)**2*T/TC**2*PART2
 DHDT = PART1 + PART2
PART1 = C*DPDV*V + C*P + C*(-Q(2,4)/VMB2 - Q(2,5)/
        VMB3 - Q(2,6)/VMB4 - Q(2,7)/VMB5 + TERMA
PART2 = -Q(4,4)/VMB2 - Q(4,5)/VMB3 - Q(4,6)/VMB4
        - Q(4,7)/VMB5 - TERMB + Q(4,8)*Q(1,12)/
        (Q(1,12)*EKA + 1.0)
PART2 = C*EKIT*(1.0 + Q(1,12)*T/TC)*PART2
 DPDV = PART1 + PART2
```

```
C     *********************************************************************
C     *                                                                   *
C     *                           E N T H A L                             *
C     *                                                                   *
C     *   THIS SUBROUTINE COMPUTES THE ENTHALPY OF THE VAPOR.             *
C     *                                                                   *
C     *********************************************************************
      SUBROUTINE ENTHAL(P,T,V,H,DHDT,DHDV)
      COMMON/FREON1/Q(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
      C=0. 185053
      T2=T*T
      T3=T2*T
      T4=T3*T
      VMB=V-Q(3,3)
      VMB2=VMB*VMB
      VMB3=VMB2*VMB
      VMB4=VMB3*VMB
      VMB5=VMB4*VMB
      EKIT=-Q(2,12)*T/(Q(4,3)+Q(1,11))
      EKIT=EXP(EKIT)
      EKA=Q(3,12)*V
      IF (EKA.GT.30.0) GO TO 105
      IF (ABS(EKA). LE. DERROR) GO TO 105
      EKA = EXP(EKA)
      IF (ABS(Q(1,12)).LE.DERROR) GO TO 100
      PART = Q(1,12)*ALOG((Q(1,12)*EKA+1.0)/(Q(1,12)*EKA))
      GO TO 110
  100 PART = 0.0
      GO TO 110
  105 EKA = 0.0
      RX = 0.0
      RZ = 0.0
      PART = 0.0
      TERMA=0.0
```

```
      TERMB=0.0
      GO TO 115
  110 RX = Q(2,8)/Q(3,12)*(1.0/EKA-PART)
      RZ=Q(4,8)/(Q(3,12)*EKA)-(Q(4,8)*PART/Q(3,12))
      TERMA=Q(2,8)*(-1.0/EKA+Q(1,12)/(Q(1,12)*EKA+1.0))
      TERMB=Q(4,8)/EKA
  115 CALL EOS(P,T,V,DPDT,DPDV,DPDVDT,D2PDT2)
      H1=Q(1,9)*T+Q(2,9)*T2/2.0+Q(3,9)*T3/3.0+Q(4,9)*T4/4.0-
     1 Q(1,10)/T+C*P*V
      H2=C*(Q(2,4)/VMB+Q(2,5)/(2.0*VMB2)+Q(2,6)/(3.0*VMB3)+Q(2,7
     1)/(4.0*VMB4)+RX)
      H3=C*(Q(4,4)/VMB+Q(4,5)/(2.0*VMB2)+Q(4,7)/(4.0*VMB4)+Q(4,6
     1 )/(3.0*VMB3)+RZ)*(1.0+Q(2,12)*T/(Q(4,3)+Q(1,11)))*EKIT+Q(
     22,10)
      H = H1 + H2 + H3
      TC=Q(4,3)+Q(1,11)
      PART1=Q(1,9)+Q(2,9)*T+Q(3,9)*T2+Q(4,9)*T3+Q(1,10)/T2+C*DPDT*V
      PART2=Q(4,4)/VMB+Q(4,5)/(2.0*VMB2)+Q(4,6)/(3.0*VMB3)+Q(4,7)/(4.0*
     1     VMB4)+RZ
      PART2=-C*EKIT*Q(2,12)**2*T/TC**2*PART2
      DHDT=PART1+PART2
      PART1=C*DPDV*V+C*P+C*(-Q(2,4)/VMB2-Q(2,5)/VMB3-Q(2,6)/VMB4-Q(2,7)/
     1     VMB5+TERMA)
      PART2=-Q(4,4)/VMB2-Q(4,5)/VMB3-Q(4,6)/VMB4-Q(4,7)/VMB5-TERMB
     1     +Q(4,8)*Q(1,12)/(Q(1,12)*EKA+1.0)
      PART2=C*EKIT*(1.0+Q(1,12)*T/TC)*PART2
      DHDV=PART1+PART2
      RETURN
      END
```

ENTROP DESCRIPTION

ENTROP calculates the entropy and its derivatives with respect to temperature and specific volume for the vapor region using the Downing-Martin equation of state.

EQUATIONS

$$S = a\ln T+B*T+C*T^2/2+d*T^3/3-f/2*T^2+CONVR*R\ln(V-\beta)-CONVR\Big(\frac{B_2}{V-\beta}$$

$$+\frac{B_3}{2(V-\beta)^2}+\frac{B_4}{3(V-\beta)^3}+\frac{B_5}{4(V-\beta)^4}+\frac{B_6}{\alpha}\,(1/\exp(\alpha*V)-C_1\ln(\frac{C_1\exp(\alpha*V)+1}{C_1\exp(\alpha*V)}))\Big)$$

$$\frac{\partial S(V,T)}{\partial V}=\frac{CONVR*R}{V-\beta}-CONVR\Big(\frac{-B_2}{(V-\beta)^2}-\frac{B_3}{(V-\beta)^3}-\frac{B_4}{(V-\beta)^4}-\frac{B_5}{(V-\beta)^5}$$

$$+B_6\ (-1/\exp(\alpha*V)+\frac{C_1}{C_1\exp(\alpha*V)+1})\Big)$$

$$\frac{\partial S(V,T)}{\partial T}=a/T+b+C*T+d*T^2+f/T^3-\frac{CONVR*K^2}{T_C^{\,2}}\exp(-K*T/TC)\Big(\frac{C_2}{(V-\beta)}$$

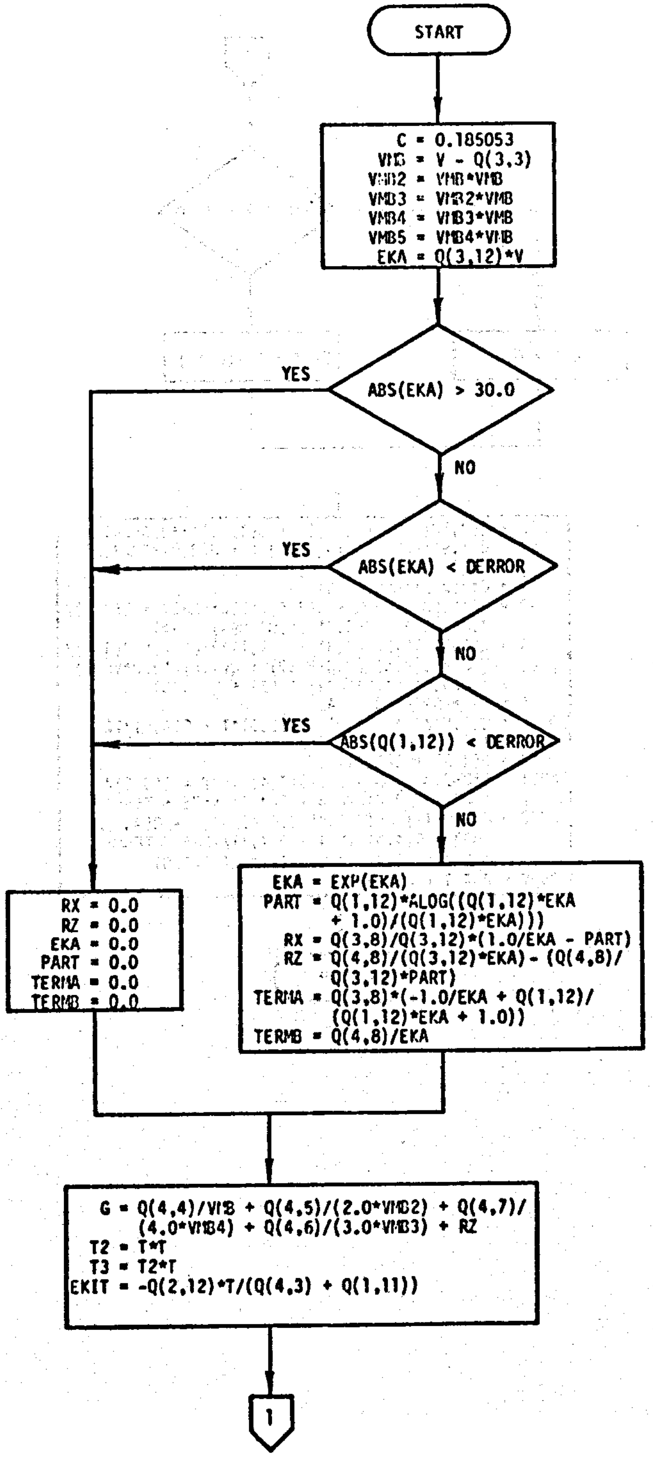
$$+\frac{C_3}{2(V-\beta)^2}+\frac{C_4}{3(V-\beta)^3}+\frac{C_5}{4(V-\beta)^4}+\frac{C_6}{\alpha\exp(\alpha*V)}-\frac{C_6*C_1}{\alpha}\ln(\frac{C_1\exp(\alpha*V)+1}{C_1\exp(\alpha*V)})\Big)$$

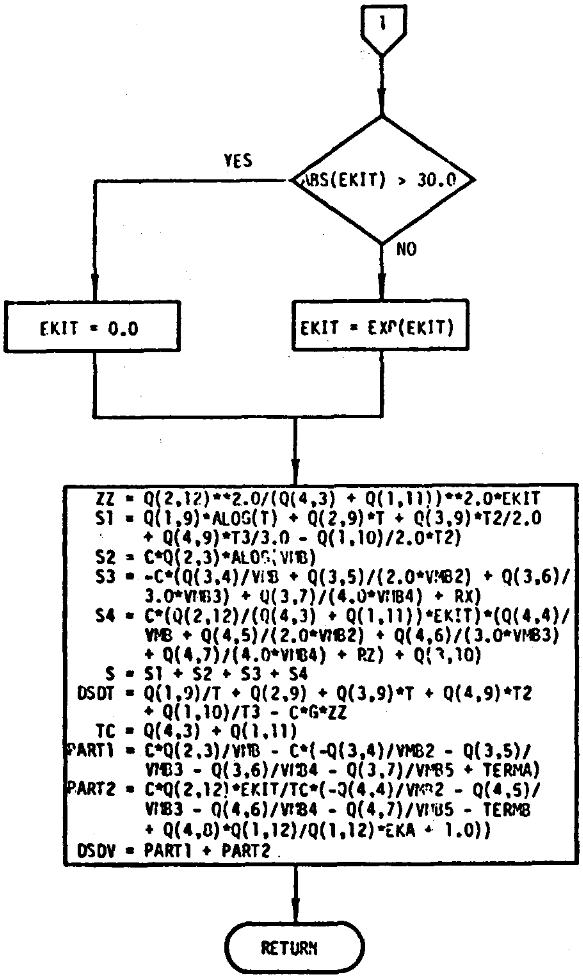
VARIABLE SYMBOL TABLE

C	conversion factor = 0.185053
DERROR	tolerance used in checks for zero
DSDT	$\frac{\partial S(V,T)}{\partial T}$
DSDV	$\frac{\partial S(V,T)}{\partial V}$
EKA	$\exp(\alpha*V)$
EKIT	$\exp(-K*T/TC)$
G	$\frac{C_2}{V-\beta}+\frac{C_3}{2(V-\beta)^2}+\frac{C_4}{3(V-\beta)^3}+\frac{C_5}{4(V-\beta)^4}+RZ$
PART	$C_1\ln(\frac{C_1\exp(\alpha*V)+1}{C_1\exp(\alpha*V)})$

PART1	$\frac{CONVR*R}{V-\beta} - CONVR\ (\frac{-B_2}{(V-\beta)^2} - \frac{B_3}{(V-\beta)^3}$ $- \frac{B_4}{(V-\beta)^4} - \frac{B_5}{(V-\beta)^5} + TERMA)$
PART2	$CONVR*K\exp(-K*T/TC)(\frac{-C_2}{(V-\beta)^2} - \frac{C_3}{(V-\beta)^3} - \frac{C_4}{(V-\beta)^4}$ $- \frac{C_5}{(V-\beta)^5} - TERMB + \frac{C_6*C_1}{C_1\exp(\alpha*V)+1})$
Q	2-dimensional array containing the coefficients for the fluid to be evaluated
RX	$\frac{B_6}{\alpha}\ (\frac{1}{\exp(\alpha*V)} - PART)$
RZ	$\frac{C_6}{\alpha\exp(\alpha*V)} - \frac{C_6*PART}{\alpha}$
S	entropy (Btu/lbm°R)
S1	$a\ln T+b*T+C*T^2/2+d*T^3/3-f/2*T^2$
S2	$CONVR*R\ln(V-\beta)$
S3	$-CONVR(\frac{B_2}{(V-\beta)} + \frac{B_3}{2(V-\beta)^2} + \frac{B_4}{3(V-\beta)^3}$ $+ \frac{B_5}{4(V-\beta)^4} + RX)$
S4	$CONVR(\frac{K}{TC}\exp(-K*T/TC)(\frac{C_2}{V-\beta} + \frac{C_3}{2(V-\beta)^2}$ $+ \frac{C_4}{3(V-\beta)^3} + \frac{C_5}{4(V-\beta)^4} + RZ) + \frac{\Delta S(\text{latent at-40°F})}{-40}$ -S(saturation -40°F)
T	temperature (°R)
T2	T^2
T3	T^3
TC	critical temperature (°R)
TERMA	$B_6(\frac{-1}{\exp(\alpha*V)} + \frac{C_1}{C_1\exp(\alpha*V)+1})$

TERMB	$C_6/\exp(\alpha*V)$
V	specific volume (ft^3/lbm)
VMB	$(V-\beta)$
VMB2	$(V-\beta)^2$
VMB3	$(V-\beta)^3$
VMB4	$(V-\beta)^4$
VMB5	$(V-\beta)^5$
ZZ	$\frac{K^2}{T_C^2}\exp(-K*T/TC)$


START
C = 0.185053
VMB = V - Q(3,3)
VMB2 = VMB*VMB
VMB3 = VMB2*VMB
VMB4 = VMB3*VMB
VMB5 = VMB4*VMB
EKA = Q(3,12)*V
YES
ABS(EKA) > 30.0
NO
YES
ABS(EKA) < DERROR
NO
YES
ABS(Q(1,12)) < DERROR
NO
RX = 0.0
RZ = 0.0
EKA = 0.0
PART = 0.0
TERMA = 0.0
TERMB = 0.0
EKA = EXP(EKA)
PART = Q(1,12)*ALOG((Q(1,12)*EKA + 1.0)/(Q(1,12)*EKA)))
RX = Q(3,8)/Q(3,12)*(1.0/EKA - PART)
RZ = Q(4,8)/(Q(3,12)*EKA) - (Q(4,8)/Q(3,12)*PART)
TERMA = Q(3,8)*(-1.0/EKA + Q(1,12)/(Q(1,12)*EKA + 1.0))
TERMB = Q(4,8)/EKA
G = Q(4,4)/VMB + Q(4,5)/(2.0*VMB2) + Q(4,7)/(4.0*VMB4) + Q(4,6)/(3.0*VMB3) + RZ
T2 = T*T
T3 = T2*T
EKIT = -Q(2,12)*T/(Q(4,3) + Q(1,11))
1



```
   ZZ = Q(2,12)**2.0/(Q(4,3) + Q(1,11))**2.0*EKIT
   S1 = Q(1,9)*ALOG(T) + Q(2,9)*T + Q(3,9)*T2/2.0
        + Q(4,9)*T3/3.0 - Q(1,10)/2.0*T2)
   S2 = C*Q(2,3)*ALOG(VMB)
   S3 = -C*(Q(3,4)/VMB + Q(3,5)/(2.0*VMB2) + Q(3,6)/
        3.0*VMB3) + Q(3,7)/(4.0*VMB4) + RX)
   S4 = C*(Q(2,12)/(Q(4,3) + Q(1,11))*EKIT)*(Q(4,4)/
        VMB + Q(4,5)/(2.0*VMB2) + Q(4,6)/(3.0*VMB3)
        + Q(4,7)/(4.0*VMB4) + RZ) + Q(3,10)
    S = S1 + S2 + S3 + S4
 DSDT = Q(1,9)/T + Q(2,9) + Q(3,9)*T + Q(4,9)*T2
        + Q(1,10)/T3 - C*G*ZZ
   TC = Q(4,3) + Q(1,11)
PART1 = C*Q(2,3)/VMB - C*(-Q(3,4)/VMB2 - Q(3,5)/
        VMB3 - Q(3,6)/VMB4 - Q(3,7)/VMB5 + TERMA)
PART2 = C*Q(2,12)*EKIT/TC*(-Q(4,4)/VMB2 - Q(4,5)/
        VMB3 - Q(4,6)/VMB4 - Q(4,7)/VMB5 - TERMB
        + Q(4,8)*Q(1,12)/Q(1,12)*EKA + 1.0))
 DSDV = PART1 + PART2
```

```
C     ***********************************************************************
C     *                                                                     *
C     *                             E N T R O P                             *
C     *                                                                     *
C     *   THIS SUBROUTINE COMPUTES THE ENTROPY OF THE VAPOR.                *
C     *                                                                     *
C     ***********************************************************************
      SUBROUTINE ENTROP(T,V,S,DSDT,DSDV)
      COMMON/FREON1/Q(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
      C=0.165053
      VMB=V-Q(3,3)
      VMB2=VMB*VMB
      VMB3=VMB2*VMB
      VMB4=VMB3*VMB
      VMB5=VMB4*VMB
      EKA=Q(3,12) *V
      IF(ABS(EKA).GT.30.0) GO TO 100
      IF(ABS(EKA).LT.DERROR) GO TO 100
      IF(ABS(Q(1,12)).LE.DERROR) GO TO 100
      EKA = EXP(EKA)
      PART = Q(1,12)*ALOG((Q(1,12)*EKA+1.0)/(Q(1,12)*EKA))
      GO TO 105
  100 RX = 0.0
      RZ = 0.0
      EKA = 0.0
      PART = 0.0
      TERMA=0.0
      TERMB=0.0
      GO TO 110
  105 RX = Q(3,8)/Q(3,12)*(1.0/EKA-PART)
      RZ = Q(4,8)/(Q(3,12)*EKA)-(Q(4,8)/Q(3,12)*PART)
      TERMA=Q(3,8)*(-1.0/EKA+Q(1,12)/(Q(1,12)*EKA+1.0))
      TERMB=Q(4,8)/EKA
  110 G=Q(4,4)/VMB+Q(4,5)/(2.0*VMB2)+Q(4,7)/(4.0*VMB4)+Q(4,6)/(
```

```
     1     3.0*VMB3)+RZ
      T2 = T*T
      T3 = T2*T
      EKIT=-Q(2,12)*T/(Q(4,3)+Q(1,11))
      IF (ABS(EKIT).GT.30.0) GO TO 115
      EKIT = EXP(EKIT)
      GO TO 120
  115 EKIT = 0.0
  120 ZZ = Q(2,12)**2.00/(Q(4,3)+Q(1,11))**2.0*EKIT
      S1=Q(1,9)*ALOG(T)+Q(2,9)*T+Q(3,9)*T2/2.0+Q(4,9)*T3/3.0-Q(1
     1 ,10)/(2.0*T2)
      S2=C*Q(2,3)*ALOG(VMB)
      S3=-C*(Q(3,4)/VMB+Q(3,5)/(2.0*VMB2)+Q(3,6)/(3.0*VMB3)+Q(3,
     1 7)/(4.0*VMB4)+RX)
      S4=C*(Q(2,12)/(Q(4,3)+Q(1,11))*EKIT)*(Q(4,4)/VMB+Q(4,5)/(2.0*V
     1 MB2)+Q(4,6)/(3.0*VMB3)+Q(4,7)/(4.0*VMB4)+RZ)+Q(3,10)
      S=S1+S2+S3+S4
      DSDT=Q(1,9)/T+Q(2,9)+Q(3,9)*T+Q(4,9)*T2+Q(1,10)/T3-C*G*ZZ
      TC=Q(4,3)+Q(1,11)
      PART1=C*Q(2,3)/VMB-C*(-Q(3,4)/VMB2-Q(3,5)/VMB3-Q(3,6)/VMB4-Q(3,7)/
     1     VMB5+TERMA)
      PART2=C*Q(2,12)*EKIT/TC*(-Q(4,4)/VMB2-Q(4,5)/VMB3-Q(4,6)/VMB4
     1     -Q(4,7)/VMB5-TERMB+Q(4,8)*Q(1,12)/(Q(1,12)*EKA+1.0))
      DSDV=PART1+PART2
  130 RETURN
      END
```

EOS DESCRIPTION

EOS calculates the pressure and its derivatives with respect to temperature and specific volume in the vapor region given temperature and specific volume by using the Downing-Martin equation of state.

EQUATIONS

$$P = \frac{R*T}{V-\beta} + \frac{A_2+B_2*T+C_2\exp(-K*T/TC)}{(V-\beta)^2} + \frac{A_3+B_3*T+C_3\exp(-K*T/TC)}{(V-\beta)^3}$$

$$+ \frac{A_4+B_4*T+C_4\exp(-K*T/TC)}{(V-\beta)^4} + \frac{A_5+B_5*T+C_5\exp(-K*T/TC)}{(V-\beta)^5} + \frac{A_6+B_6*T+C_6\exp(-K*T/TC)}{\exp(\alpha*V)(1+C_1\exp(\alpha*V))}$$

$$\frac{\partial P(V,T)}{\partial V} = \frac{-R*T}{(V-\beta)^2} - \frac{2(A_2+B_2*T+C_2\exp(-K*T/TC))}{(V-\beta)^3} - \frac{3(A_3+B_3*T+C_3\exp(-K*T/TC))}{(V-\beta)^4}$$

$$- \frac{4(A_4+B_4*T+C_4\exp(-K*T/TC))}{(V-\beta)^5} - \frac{5(A_5+B_5*T+C_5\exp(-K*T/TC))}{(V-\beta)^6}$$

$$- (A_6+B_6*T+C_6\exp(-K*T/TC)) \left(\frac{\exp(\alpha*V)+2* *C_1\exp(2*\alpha*V)}{(\exp(\alpha*V)+C_1\exp(2*\alpha*V))^2}\right)$$

$$\frac{\partial P(V,T)}{\partial T} = \frac{R}{V-\beta} + \frac{B_2 - \dfrac{KC_2\exp(-K*T/TC)}{T_C}}{(V-\beta)^2} + \frac{B_3 - \dfrac{KC_3\exp(-K*T/TC)}{T_C}}{(V-\beta)^3}$$

$$+ \frac{B_4 - \dfrac{KC_4\exp(-K*T/TC)}{T_C}}{(V-\beta)^4} + \frac{B_5 - \dfrac{KC_5\exp(-K*T/TC)}{T_C}}{(V-\beta)^5} + \frac{B_6 - \dfrac{KC_6\exp(-K*T/TC)}{T_C}}{\exp(\alpha*V)(C_1\exp(\alpha*V)+1)}$$

$$\frac{\partial^2 P(V,T)}{\partial T^2} = \frac{K^2\exp(-K*T/TC)}{{T_C}^2} \left(\frac{C_2}{(V-\beta)^2} + \frac{C_3}{(V-\beta)^3} + \frac{C_4}{(V-\beta)^4} + \frac{C_5}{(V-\beta)^5}\right.$$

$$\left. + \frac{C_6}{\exp(\alpha*V)(C_1\exp(\alpha*V)+1)}\right)$$

$$\frac{\partial^2 P(V,T)}{\partial V \partial T} = \frac{-R}{(V-\beta)^2} - \frac{2(B_2 - \frac{KC_2 \exp(-K*T/TC)}{T_C})}{(V-\beta)^3} - \frac{3(B_3 - \frac{KC_3 \exp(-K*T/TC)}{T_C})}{(V-\beta)^4}$$

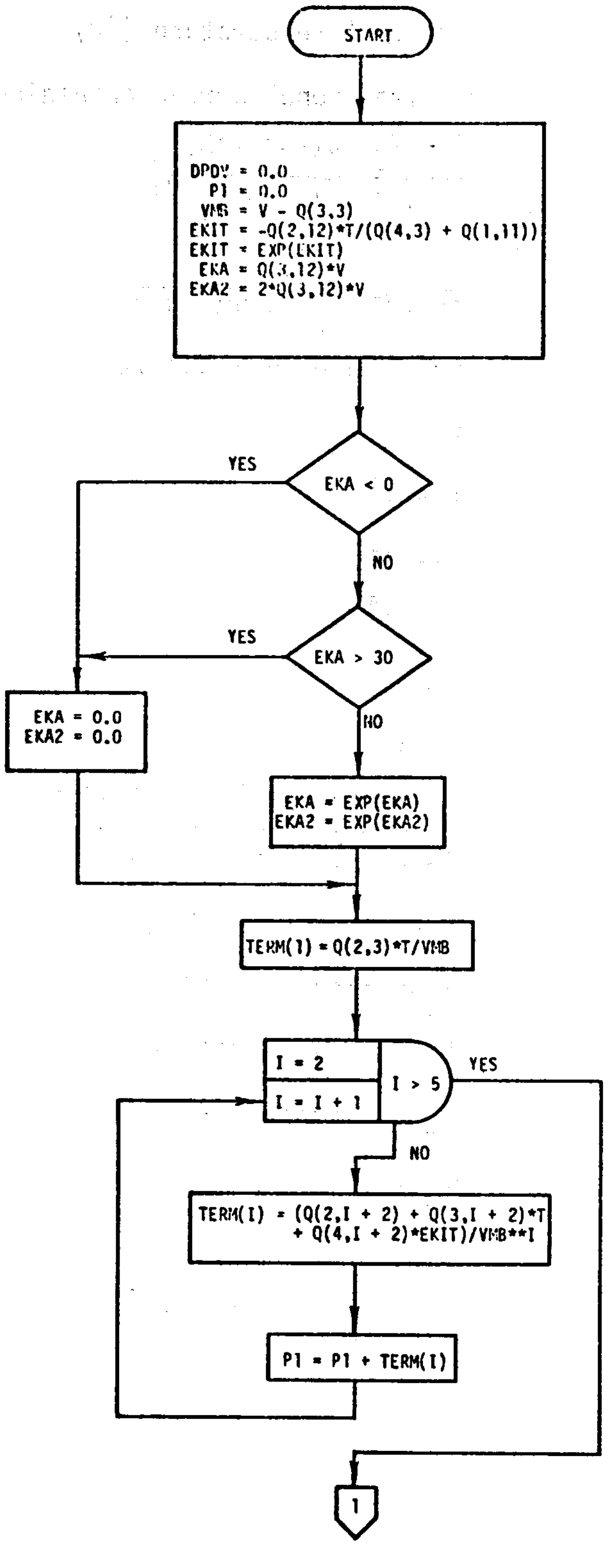
$$- \frac{4(B_4 - \frac{KC_4 \exp(-K*T/TC)}{T_C})}{(V-\beta)^5} - \frac{5(B_5 - \frac{KC_5 \exp(-K*T/TC)}{T_C})}{(V-\beta)^6} - (B_6 - \frac{KC_6 \exp(-K*T/TC)}{T_C})$$

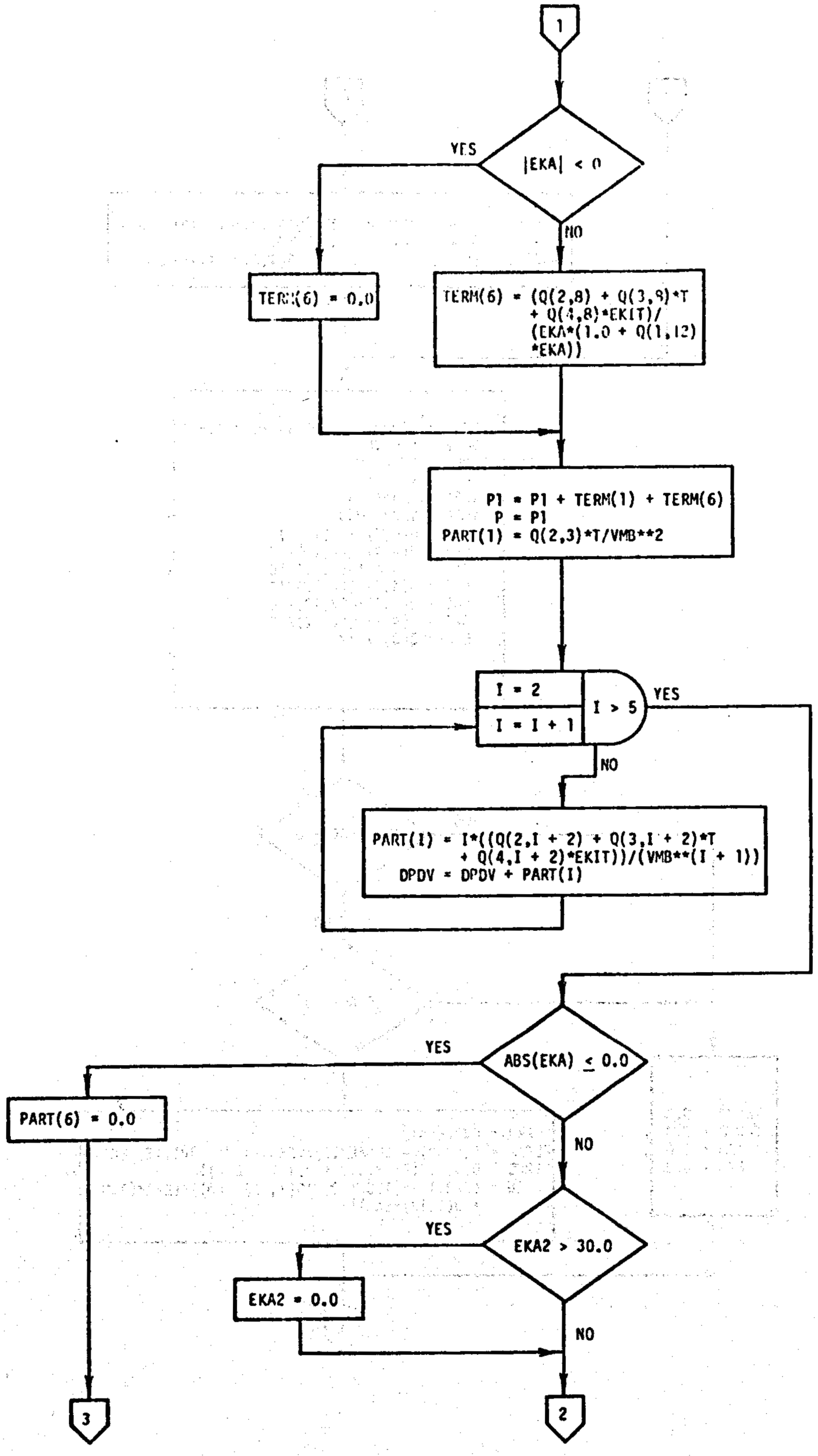
$$(\frac{\alpha(1+2*C_1 \exp(\alpha*V))}{\exp(\alpha*V)(1+C_1 \exp(\alpha*V))})$$

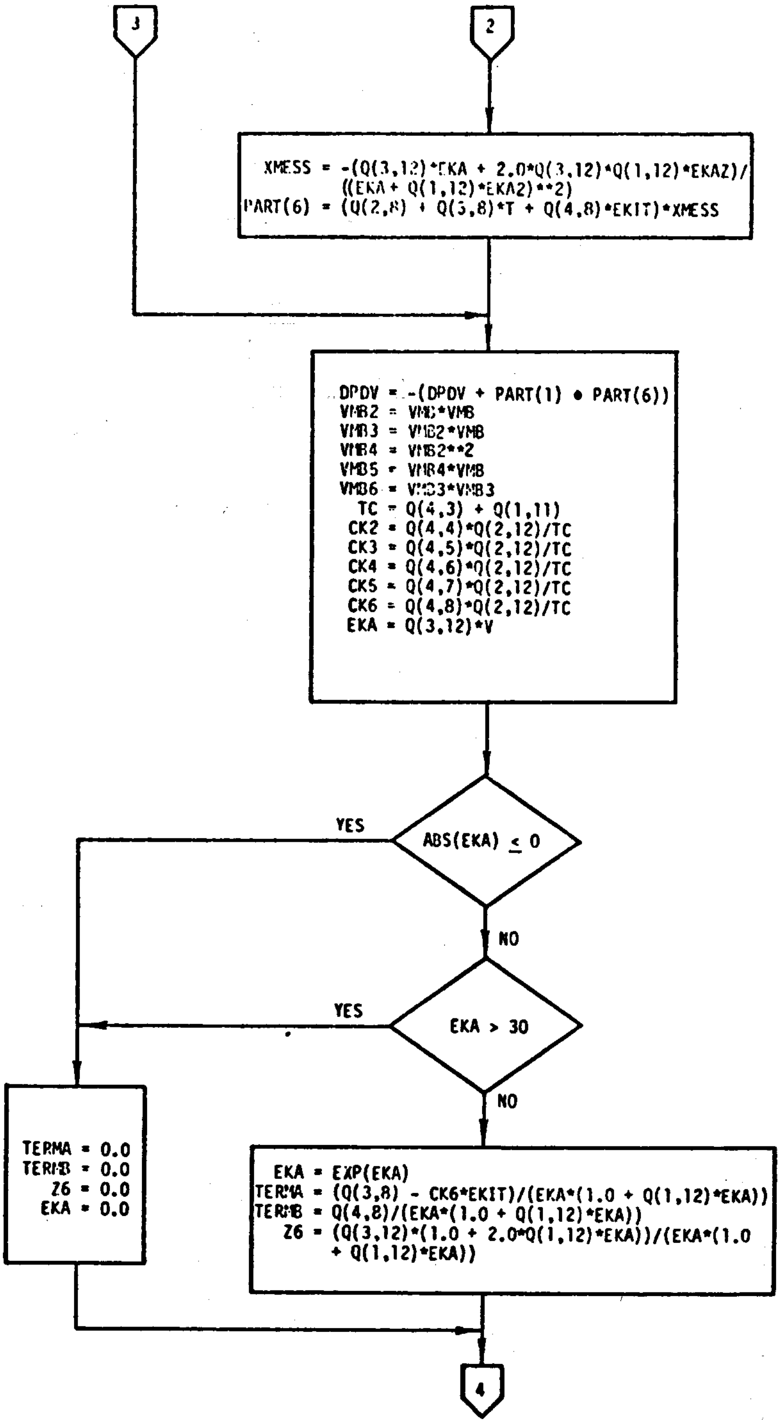
VARIABLE SYMBOL TABLE

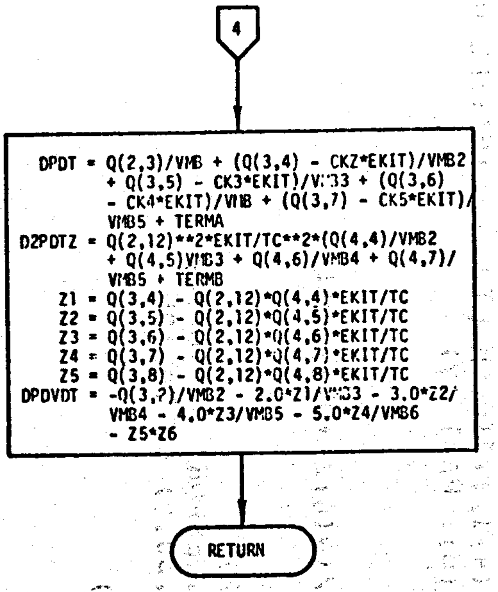
CK2	C_2*K/TC
CK3	C_3*K/TC
CK4	C_4*K/TC
CK5	C_5*K/TC
CK6	C_6*K/TC
DERROR	tolerance used in checks for zero
DPDT	$\frac{\partial P(V,T)}{\partial T}$
DPDV	$\frac{\partial P(V,T)}{\partial V}$
DPDVDT	$\frac{\partial^2 P(V,T)}{\partial V \partial T}$
D2PDT2	$\frac{\partial^2 P(V,T)}{\partial T^2}$
EKA	exp(α*V)
EKA2	exp(2*α*V)
EKIT	exp(-K*T/TC)
I	index
P	pressure (psia)
PART	1-dimensional array containing intermediate values
P1	pressure (psia)
Q	2-dimensional array containing the coefficients for the fluid to be evaluated

T	temperature (°R)
TC	critical temperature (°R)
TERM	1-dimensional array containing intermediate values
TERMA	$\frac{B_6\text{-}CK_6\exp(-K*T/TC)}{\exp(\alpha*V)(1+\ \exp(\alpha*V)}$
TERMB	$\frac{C_6}{\exp(\alpha*V)(1+\ \exp(\alpha*V))}$
V	specific volume (ft^3/lbm)
VMB	$V-\beta$
VMB2	$(V-\beta)^2$
VMB3	$(V-\beta)^3$
VMB4	$(V-\beta)^4$
VMB5	$(V-\beta)^5$
VMB6	$(V-\beta)^6$
XMESS	$-\frac{(\alpha\exp(\alpha*V)+Z*\ *C_1\exp(2*\alpha*V))}{(\exp(\alpha*V)+C_1\exp(2*\alpha*V))^2}$
Z1	$B_2-C_2*K\exp(-K*T/TC)/TC$
Z2	$B_3-C_3*K\exp(-K*T/TC)/TC$
Z3	$B_4-C_4*K\exp(-K*T/TC)/TC$
Z4	$B_5-C_5*K\exp(-K*T/TC)/TC$
Z5	$B_6-C_6*K\exp(-K*T/TC)/TC$
Z6	$\frac{(1+2*C_1\exp(\alpha*V))}{\exp(\alpha*V)(1+\ \exp(\alpha*V))}$


START
DPDV = 0.0
P1 = 0.0
VMB = V - Q(3,3)
EKIT = -Q(2,12)*T/(Q(4,3) + Q(1,11))
EKIT = EXP(EKIT)
EKA = Q(3,12)*V
EKA2 = 2*Q(3,12)*V
EKA < 0
YES
NO
EKA > 30
YES
NO
EKA = 0.0
EKA2 = 0.0
EKA = EXP(EKA)
EKA2 = EXP(EKA2)
TERM(1) = Q(2,3)*T/VMB
I = 2
I = I + 1
I > 5
YES
NO
TERM(I) = (Q(2,I + 2) + Q(3,I + 2)*T + Q(4,I + 2)*EKIT)/VMB**I
P1 = P1 + TERM(I)
1


1
|EKA| < 0
YES
NO
TERM(6) = 0.0
TERM(6) = (Q(2,8) + Q(3,8)*T + Q(4,8)*EKIT)/(EKA*(1.0 + Q(1,12)*EKA))
P1 = P1 + TERM(1) + TERM(6)
P = P1
PART(1) = Q(2,3)*T/VMB**2
I = 2
I = I + 1
I > 5
YES
NO
PART(I) = I*((Q(2,I + 2) + Q(3,I + 2)*T + Q(4,I + 2)*EKIT))/(VMB**(I + 1))
DPDV = DPDV + PART(I)
ABS(EKA) ≤ 0.0
YES
NO
PART(6) = 0.0
EKA2 > 30.0
YES
NO
EKA2 = 0.0
3
2


3
2
XMESS = -(Q(3,12)*EKA + 2.0*Q(3,12)*Q(1,12)*EKA2)/
((EKA + Q(1,12)*EKA2)**2)
PART(6) = (Q(2,8) + Q(3,8)*T + Q(4,8)*EKIT)*XMESS
DPDV = -(DPDV + PART(1) • PART(6))
VMB2 = VMB*VMB
VMB3 = VMB2*VMB
VMB4 = VMB2**2
VMB5 = VMB4*VMB
VMB6 = VMB3*VMB3
TC = Q(4,3) + Q(1,11)
CK2 = Q(4,4)*Q(2,12)/TC
CK3 = Q(4,5)*Q(2,12)/TC
CK4 = Q(4,6)*Q(2,12)/TC
CK5 = Q(4,7)*Q(2,12)/TC
CK6 = Q(4,8)*Q(2,12)/TC
EKA = Q(3,12)*V
YES
ABS(EKA) ≤ 0
NO
YES
EKA > 30
NO
TERMA = 0.0
TERMB = 0.0
Z6 = 0.0
EKA = 0.0
EKA = EXP(EKA)
TERMA = (Q(3,8) - CK6*EKIT)/(EKA*(1.0 + Q(1,12)*EKA))
TERMB = Q(4,8)/(EKA*(1.0 + Q(1,12)*EKA))
Z6 = (Q(3,12)*(1.0 + 2.0*Q(1,12)*EKA))/(EKA*(1.0
+ Q(1,12)*EKA))
4



```
   DPDT = Q(2,3)/VMB + (Q(3,4) - CKZ*EKIT)/VMB2
          + Q(3,5) - CK3*EKIT)/VMB3 + (Q(3,6)
          - CK4*EKIT)/VMB + (Q(3,7) - CK5*EKIT)/
          VMB5 + TERMA
 D2PDTZ = Q(2,12)**2*EKIT/TC**2*(Q(4,4)/VMB2
          + Q(4,5)VMB3 + Q(4,6)/VMB4 + Q(4,7)/
          VMB5 + TERMB
     Z1 = Q(3,4) - Q(2,12)*Q(4,4)*EKIT/TC
     Z2 = Q(3,5) - Q(2,12)*Q(4,5)*EKIT/TC
     Z3 = Q(3,6) - Q(2,12)*Q(4,6)*EKIT/TC
     Z4 = Q(3,7) - Q(2,12)*Q(4,7)*EKIT/TC
     Z5 = Q(3,8) - Q(2,12)*Q(4,8)*EKIT/TC
 DPDVDT = -Q(3,3)/VMB2 - 2.0*Z1/VMB3 - 3.0*Z2/
          VMB4 - 4.0*Z3/VMB5 - 5.0*Z4/VMB6
          - Z5*Z6
```

```
C      ************************************************************************
C      *                                                                      *
C      *                              E O S                                   *
C      *                                                                      *
C      *   THIS SUBROUTINE CALCULATES THE PRESSURE FOR VAPOR STATES AND *
C      *ITS DERIVATIVES GIVEN TEMPERATURE AND DENSITY                        *
C      *                                                                      *
C      ************************************************************************
       SUBROUTINE EOS(P,T,V,DPDT,DPDV,DPDVDT,D2PDT2)
       COMMON/FREON1/Q(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
       DIMENSION TERM (6), PART (7)
       DPDV = 0.0
       P1=0.0
       VMB=V-Q(3,3)
       EKIT=-Q(2,12)*T/(Q(4,3)+Q(1,11))
       EKIT=EXP(EKIT)
       EKA=Q(3,12)*V
       EKA2=2.00*Q(3,12)*V
       IF(ABS (EKA).LE.DERROR) GO TO 105
       IF (ABS(EKA).GT.50.0) GO TO 105
       EKA = EXP(EKA)
       EKA2 = EXP(EKA2)
       GO TO 115
   105 EKA = 0.0
       EKA2 = 0.0
   115 TERM(1) = Q(2,3)*T/VMB
       DO 125 I = 2,5
       TERM(I) = (Q(2,I+2)+Q(3,I+2)*T+Q(4,I+2)*EKIT)/VMB**I
   125 P1 = P1 +TERM(I)
       IF(ABS(EKA).LE.DERROR) TERM (6) = 0.0
       IF(ABS(EKA).LE.DERROR) GO TO 120
       TERM (6) = (Q(2,8) + Q(3,8)*T+Q(4,8)*EKIT)/(EKA*(1.0+Q(1,12)*E
      1KA))
   120 P1 = P1 + TERM(1) + TERM(6)
```

```
      P=P1
  140 PART (1) = Q(2,3)*T/VMB**2
      DO 150 I = 2,5
      PART(I)  = FLOAT (I)*(Q(2,I+2)+Q(3,I+2)*T+Q(4,I+2)*EKIT)/VMB**(I+1
     1)
  150 DPDV = DPDV + PART (I)
      IF (ABS (EKA).LE.DERROR) PART (6) = 0.0
      IF (ABS (EKA).LE.DERROR) GO TO 145
      IF (EKA2.GT.30.0) EKA2 = 0.0
      XMESS = (Q(3,12)*EKA+2.0*Q(3,12)*Q(1,12)*EKA2)/((EKA+Q(1,12)*E
     1KA2)**2)
      XMESS = -XMESS
      PART (6) = (Q(2,8)+Q(3,8)*T+Q(4,8)* EKIT)*XMESS
  145 DPDV =-(DPDV+PART(1) + PART (6))
      VMB2=VMB*VMB
      VMB3=VMB2*VMB
      VMB4=VMB2**2
      VMB5=VMB4*VMB
      VMB6=VMB3*VMB3
      TC=Q(4,3)+Q(1,11)
      CK2=Q(4,4)*Q(2,12)/TC
      CK3=Q(4,5)*Q(2,12)/TC
      CK4=Q(4,6)*Q(2,12)/TC
      CK5=Q(4,7)*Q(2,12)/TC
      CK6=Q(4,8)*Q(2,12)/TC
      EKA=Q(3,12)*V
      IF(ABS(EKA).LE.DERROR) GO TO 100
      IF(EKA.GT.30.0) GO TO 100
      EKA=EXP(EKA)
      TERMA=(Q(3,8)-CK6*EKIT)/(EKA*(1.0+Q(1,12)*EKA))
      TERMB=Q(4,8)/(EKA*(1.0+Q(1,12)*EKA))
      Z6=Q(3,12)*(1.0+2.0*Q(1,12)*EKA)/(EKA*(1.0+Q(1,12)*EKA))
      GO TO 180
  100 TERMA=0.0
      TERMB=0.0
```

```
          Z6=0.0
          EKA=0.0
      160 DPDT=Q(2,3)/VMB+(Q(3,4)-CK2*EKIT)/VMB2+(Q(3,5)-CK3*EKIT)/VMB3+
         1    (Q(3,6)-CK4*EKIT)/VMB4+(Q(3,7)-CK5*EKIT)/VMB5+TERMA
          D2PDT2=Q(2,12)**2*EKIT/TC**2*(Q(4,4)/VMB2+Q(4,5)/VMB3+Q(4,6)/VMB4+
         1    Q(4,7)/VMB5+TERMB)
          Z1=Q(3,4)-Q(2,12)*Q(4,4)*EKIT/TC
          Z2=Q(3,5)-Q(2,12)*Q(4,5)*EKIT/TC
          Z4=Q(3,7)-Q(2,12)*Q(4,7)*EKIT/TC
          Z3=Q(3,6)-Q(2,12)*Q(4,6)*EKIT/TC
          Z5=Q(3,8)-Q(2,12)*Q(4,8)*EKIT/TC
          DPDVDT=-Q(3,2)/VMB2-2.0*Z1/VMB3-3.0*Z2/VMB4-4.0*Z3/VMB5-5.0*Z4/
         1    VMB6-Z5*Z6
      175 RETURN
          END
```

FLQDEN DESCRIPTION

FLQDEN calculates the saturated liquid specific volume and its derivative with respect to temperature given temperature. One of three equations is selected depending on the refrigerant to be evaluated.

EQUATIONS

NTYPE = 1

$$1/V_L = A_L+B_L(1-T/TC)^{1/3}+C_L(1-T/TC)^{2/3}+D_L(1-T/TC)+E_L(1-T/TC)^{4/3}$$

$$\frac{dV_L}{dT} = V_L^2 \left(\frac{B_L}{3*TC} (1-T/TC)^{-2/3}+ \frac{2*C_L}{3*TC} (1-T/TC)^{-1/3} + \frac{4*E_L}{3*TC} (1-T/TC)^{1/3}\right)$$

NTYPE = 2

$$1/V_L = A_L+B_L(TC-T)+C_L(TC-T)^{1/2}+D_L(TC-T)^{1/3}+E_L(TC-T)^2$$

$$\frac{dV_L}{dT} = V_L^2 \; (B_L+C_L(TC-T)^{-1/2}/2+D_L(TC-T)^{-2/3}/3+2*E_L(TC-T))$$

NTYPE = 3

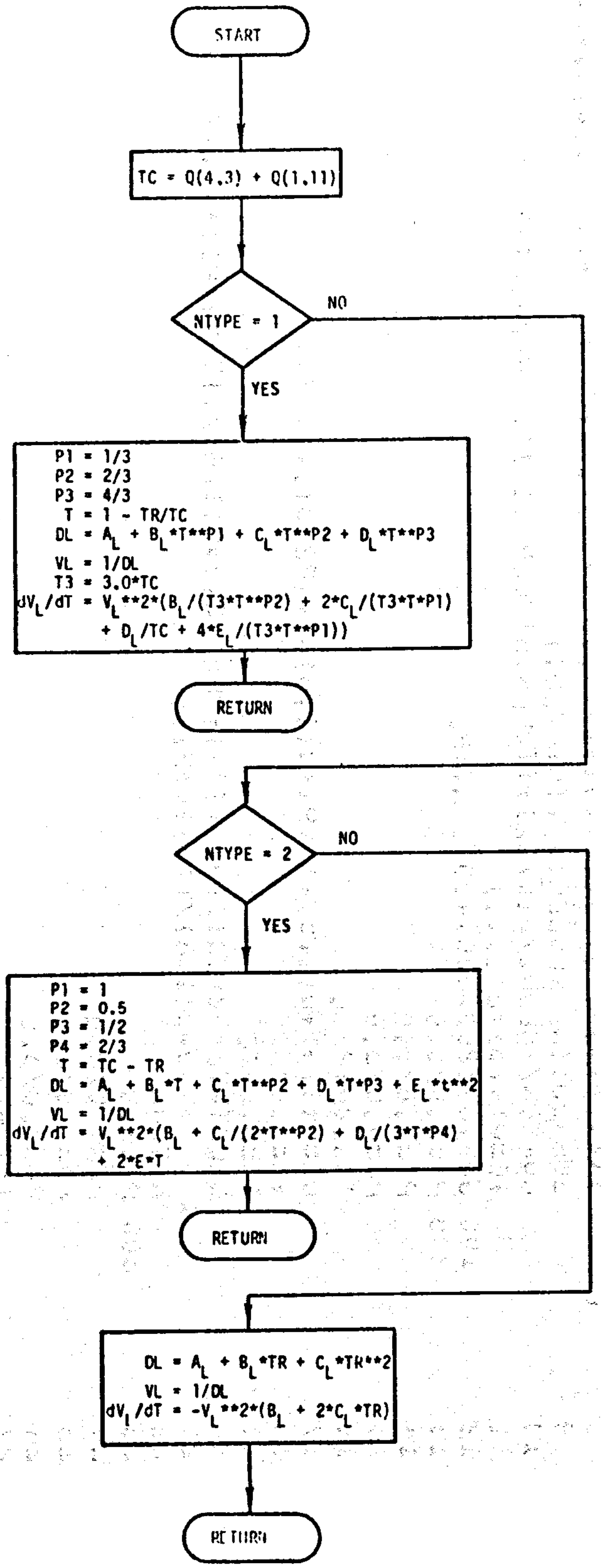
$$1/V_L = A_L+B_L*T+C_L*T^2$$

$$\frac{dV_L}{dT} = -V_L^2(B_L+2*C_L*T)$$

VARIABLE SYMBOL TABLE

DL	saturated liquid density (lbm/ft^3)
DVLDT	$\frac{dV_L(T)}{dT}$
NTYPE	saturated liquid specific volume equation selector
P1	temporary value
P2	temporary value
P3	temporary value
P4	temporary value
Q	2-dimensional array containing the coefficients for the fluid to be evaluated
T	temperature function

TC	critical temperature (°R)
TR	temperature (°R)
T3	3*TC (°R)
VL	saturated liquid specific volume (ft^{3}lbm)


START
TC = Q(4,3) + Q(1,11)
NTYPE = 1
NO
YES
P1 = 1/3
P2 = 2/3
P3 = 4/3
T = 1 - TR/TC
DL = A_L + B_L*T**P1 + C_L*T**P2 + D_L*T**P3
VL = 1/DL
T3 = 3.0*TC
dV_L/dT = V_L**2*(B_L/(T3*T**P2) + 2*C_L/(T3*T*P1)
+ D_L/TC + 4*E_L/(T3*T**P1))
RETURN
NTYPE = 2
NO
YES
P1 = 1
P2 = 0.5
P3 = 1/2
P4 = 2/3
T = TC - TR
DL = A_L + B_L*T + C_L*T**P2 + D_L*T*P3 + E_L*t**2
VL = 1/DL
dV_L/dT = V_L**2*(B_L + C_L/(2*T**P2) + D_L/(3*T*P4)
+ 2*E*T
RETURN
DL = A_L + B_L*TR + C_L*TR**2
VL = 1/DL
dV_l/dT = -V_L**2*(B_L + 2*C_L*TR)
RETURN

```
C       *********************************************************************
C       *                                                                   *
C       *                          F L Q D E N                              *
C       *                                                                   *
C       *   THIS SUBROUTINE COMPUTES THE SATURATED LIQUID SPECIFIC VOLUME*
C       *GIVEN TEMPERATURE USING ONE OF THREE EQUATIONS.                    *
C       *                                                                   *
C       *********************************************************************
        SUBROUTINE FLQDEN(VL,TR,DVLDT)
        COMMON/FREON1/Q(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
        TC=Q(4,3)+Q(1,11)
    100 GO TO (200,300,400), NTYPE
    200 P1=1.0/3.0
        P2=2.0/3.0
        P3=4.0/3.0
        T=1.0-TR/TC
        DL=Q(1,4)+Q(1,5)*T**P1+Q(1,6)*T**P2+Q(1,7)*T**P3
        VL=1.0/DL
        T3=3.0*TC
        DVLDT=VL**2*(Q(1,5)/(T3*T**P2)+2.0*Q(1,6)/(T3*T**P1)+Q(1,7)/TC
       1    +4.0*Q(1,8)/(T3*T**P1))
        RETURN
    300 P1=1.0
        P2=1.0/2.0
        P3=1.0/3.0
        P4=2.0/3.0
        T=TC-TR
        DL=Q(1,4)+Q(1,5)*T**P1+Q(1,6)*T**P2+Q(1,7)*T**P3+Q(1,8)*T**2
        VL=1.0/DL
        DVLDT=VL**2*(Q(1,5)+Q(1,6)/(2.0*T**P2)+Q(1,7)/(3.0*T**P4)+2.0*
       1    Q(1,8)*T)
        RETURN
    400 DL=Q(1,4)+Q(1,5)*TR+Q(1,6)*TR**2
        VL=1.0/DL
```

```
      DVLDT=VL**2*(-Q(1,5)-Q(1,6)*2.0*TR)
      RETURN
      END
```

HFIND DESCRIPTION

HFIND searches for the specific volume and temperature in the vapor region given pressure and enthalpy. A two variable Newton-Raphson iteration technique is used for the search.

EQUATIONS

$$\text{Jacobian} = \left(\frac{\partial P(V,T)}{\partial V}\right)\left(\frac{\partial h(V,T)}{\partial T}\right)-\left(\frac{\partial P(V,T)}{\partial T}\right)\left(\frac{\partial h(V,T)}{\partial V}\right)$$

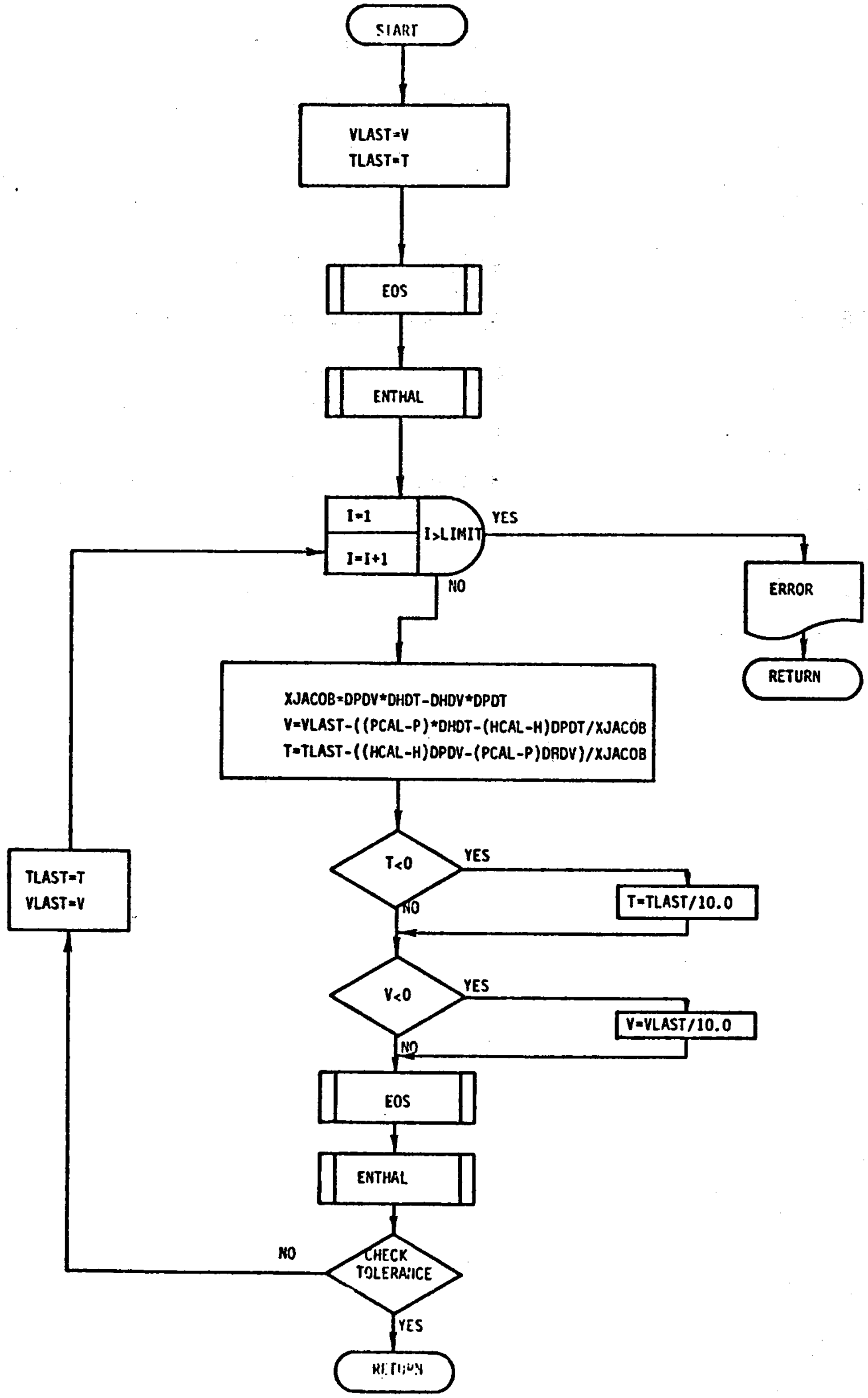
$$V_{i+1} = V_i - \left[\frac{(P(V,T)-P_{KNOWN})\frac{\partial h(V,T)}{\partial T} - (h(V,T)-h_{KNOWN})\frac{\partial P(V,T)}{\partial T}}{\text{Jacobian}}\right]_i$$

$$T_{i+1} = T_i - \left[\frac{(h(V,T)-h_{KNOWN})\frac{\partial P(V,T)}{\partial V} - (P(V,T)-P_{KNOWN})\frac{\partial h(V,T)}{\partial V}}{\text{Jacobian}}\right]_i$$

VARIABLE SYMBOL TABLE

BTOLER	lower tolerance on convergence and saturation tests
DHDT	$\frac{\partial h(V,T)}{\partial T}$
DHDV	$\frac{\partial h(V,T)}{\partial V}$
DPDT	$\frac{\partial P(V,T)}{\partial T}$
DPDV	$\frac{\partial P(V,T)}{\partial V}$
DPDVDT	$\frac{\partial^2 P(V,T)}{\partial V \partial T}$
D2PDT2	$\frac{\partial^2 P(V,T)}{\partial T^2}$
H	known enthalpy (Btu/lbm)
HCAL	calculated enthalpy (Btu/lbm)
I	index
LIMIT	maximum number of iterations allowed in any one search

M2	unit number for printer
P	known pressure (psia)
PCAL	calculated pressure (psia)
T	temperature (°R)
TLAST	last temperature calculated to be used in next iteration
TTOLER	upper tolerance on convergence and saturation tests
V	specific volume (ft^3/lbm)
VLAST	last specific volume calculated, to be used in next iteration
XJACOB	Jacobian of pressure and enthalpy


START
VLAST=V
TLAST=T
EOS
ENTHAL
I=1
I=I+1
I>LIMIT
YES
NO
ERROR
RETURN
XJACOB=DPDV*DHDT-DHDV*DPDT
V=VLAST-((PCAL-P)*DHDT-(HCAL-H)DPDT/XJACOB
T=TLAST-((HCAL-H)DPDV-(PCAL-P)DRDV)/XJACOB
TLAST=T
VLAST=V
T<0
YES
NO
T=TLAST/10.0
V<0
YES
NO
V=VLAST/10.0
EOS
ENTHAL
NO
CHECK TOLERANCE
YES
RETURN

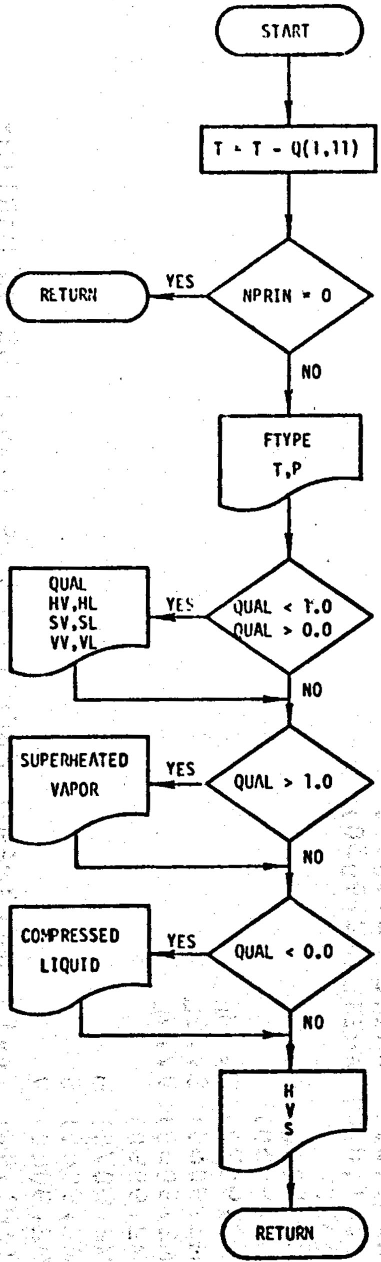
```
C     ******************************************************************
C     *                                                                *
C     *                          H F I N D                             *
C     *                                                                *
C     *   THIS SUBROUTINE FINDS TEMPERATURE AND DENSITY GIVEN PRESSURE *
C     *AND ENTHALPY FOR SUPERHEATED VAPOR STATES.                      *
C     *                                                                *
C     ******************************************************************
      SUBROUTINE HFIND(T,P,V,H)
      PARAMETER M2=6
      COMMON/FREON1/G(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
      VLAST=V
      TLAST=T
      CALL EOS(PCAL,T,V,DPDT,DPDV,DPDVDT,DP2DT2)
      CALL ENTHAL(PCAL,T,V,HCAL,DHDT,DHDV)
      DO 1000 I=1,LIMIT
      XJACOB=DPDV*DHDT-DPDT*DHDV
      V=VLAST-((PCAL-P)*DHDT-(HCAL-H)*DPDT)/XJACOB
      T=TLAST-((HCAL-H)*DPDV-(PCAL-P)*DHDV)/XJACOB
      IF(T.LE.0.0) T=TLAST/10.0
      IF(V.LE.0.0) V=VLAST/10.0
      CALL EOS(PCAL,T,V,DPDT,DPDV,DPDVDT,DP2DT2)
      CALL ENTHAL(PCAL,T,V,HCAL,DHDT,DHDV)
      IF(PCAL/P.LT.TTOLER.AND.PCAL/P.GT.BTOLER.AND.HCAL/H.LT.
     1    TTOLER.AND.HCAL/H.GT.BTOLER) GO TO 1050
      IF(ABS(T-TLAST).LT.1.0E-04.AND.ABS(V-VLAST).LT.1.0E-07) GO TO 1050
      VLAST=V
      TLAST=T
 1000 CONTINUE
      WRITE(M2,9000) LIMIT
 1050 RETURN
 9000 FORMAT(' ',10X,'HFIND FAILED TO CONVERGE IN ',I5,' ITERATIONS')
      END
```

PRINTF DESCRIPTION

PRINTF prints the final results of FREON if NPRIN does not equal zero.

VARIABLE SYMBOL TABLE

FTYPE	fluid type to which properties have been determined
H	enthalpy (Btu/lbm)
HL	saturated liquid enthalpy (Btu/lbm)
HV	saturated vapor enthalpy (Btu/lbm)
NPRIN	print selector
M2	unit number for printer
P	pressure (psia)
Q	2-dimensional array containing the coefficients for the fluid to be evaluated
S	entropy (Btu/lbm°R)
SL	saturated liquid entropy (BTU/lbm °R)
SV	saturated vapor entropy (BTU/lbm °R)
T	temperature (°R)
V	specific volume (ft^3/lbm)
VL	saturated liquid specific volume (ft^3/lbm)
VV	saturated vapor specific volume (ft^3/lbm)


START
T = T - Q(1,11)
NPRIN = 0
YES
RETURN
NO
FTYPE
T,P
QUAL < 1.0
QUAL > 0.0
YES
QUAL
HV,HL
SV,SL
VV,VL
NO
QUAL > 1.0
YES
SUPERHEATED
VAPOR
NO
QUAL < 0.0
YES
COMPRESSED
LIQUID
NO
H
V
S
RETURN

```
C     *******************************************************************
C     *                                                                 *
C     *                          P R I N T F                            *
C     *                                                                 *
C     *   THIS ROUTINE PRINTS THE RESULTS OF ROUTINE FREON.             *
C     *                                                                 *
C     *******************************************************************
      SUBROUTINE PRINTF(T,P,QUAL,V,H,S,VV,HV,SV,VL,HL,SL,FTYPE,NPRIN)
      PARAMETER M2=6
      COMMON/FREON1/G(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
      T=T-G(1,11)
      IF(NPRIN.EQ.0) RETURN
      WRITE(M2,9000) FTYPE
      WRITE(M2,9010) T,P
      IF(QUAL.LE.1.0.AND.QUAL.GE.0.0) GO TO 3280
      IF(QUAL.GT.1.0) WRITE(M2,9020)
      IF(QUAL.LT.0.0) WRITE(M2,9030)
      GO TO 3370
 3280 WRITE(M2,9040) QUAL
      WRITE(M2,9050)
      WRITE(M2,9070) HV,HL
      WRITE(M2,9080) SV,SL
      WRITE(M2,9060) VV,VL
 3370 WRITE(M2,9090)
      WRITE(M2,9070) H
      WRITE(M2,9080) S
      WRITE(M2,9060) V
 9000 FORMAT(1H1,4X,'***THERMODYNAMIC PROPERTIES OF ',A4,' USING MARTIN-
     1HOU EQUATION OF STATE***')
 9010 FORMAT(1H ,4X,'TEMPERATURE = ',F7.1,'DEG.F',/,5X,'PRESSURE    = ',
     1    F10.5,'PSIA')
 9020 FORMAT(1H ,4X,'SUPERHEATED VAPOR',//////)
 9030 FORMAT(1H ,4X,'COMPRESSED LIQUID-APPROXIMATED BY SATURATED LIQUID'
     1    ,//////)
```

```
9040 FORMAT(1H ,4X,'SATURATED LIQUID-VAPOR EQUILIBRUIM',/,5X,'QUALITY =
    1 ',F6.4)
9050 FORMAT(1H ,///,35X,'SATURATED',8X,'SATURATED',/,35X,'VAPOR',12X,
    1     'LIQUID',/,5X,63('#'))
9060 FORMAT(1H ,4X,'SPECIFIC VOLUME (CU.FT/LBM)',5X,F9.4,8X,F9.4,/////
    1     /)
9070 FORMAT(1H ,4X,'ENTHALPY',8X,'(BTU/LBM)',5X,F9.3,8X,F9.3)
9080 FORMAT(1H ,4X,'ENTROPY',9X,'(BTU/LBM-R)',5X,F9.6,8X,F9.6)
9090 FORMAT(1H ,//////)
     RETURN
     END
```

SATN DESCRIPTION

SATN determines the saturated vapor and liquid properties given temperature and the saturated vapor and liquid specific volumes. Saturated liquid properties are determined from the Clapeyron equation.

EQUATIONS

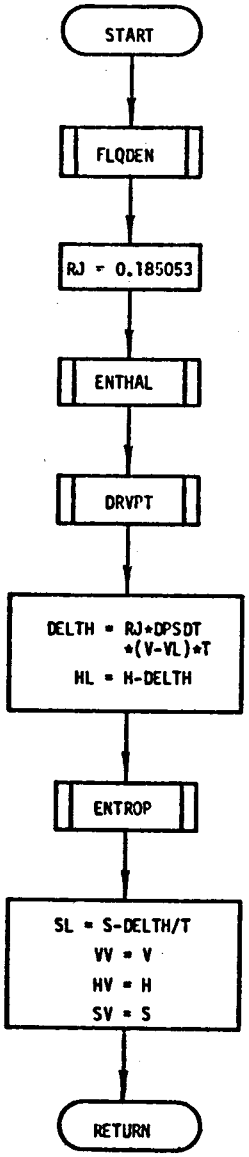
$$h_L = h_V - \text{CONVR}\left(\frac{dP_{SAT}}{dT}\right)(V_V - V_L)*T$$

$$S_L = S_V - \text{CONVR}\left(\frac{dP_{SAT}}{dT}\right)(V_V - V_L)$$

VARIABLE SYMBOL TABLE

DELTH	(h_L-h_g)calculated from (Clapeyron equation)
DHDT	$\frac{\partial h(V,T)}{\partial T}$
DHDV	$\frac{\partial h(V,T)}{\partial V}$
DPSDT	$\frac{dP_{SAT}}{dT}$
DPSDT2	$\frac{d^2P_{SAT}}{dT^2}$
DSDT	$\frac{\partial S(V,T)}{\partial T}$
DSDV	$\frac{\partial S(V,T)}{\partial V}$
DVLDT	$\frac{dV_L}{dT}$
H	enthalpy (Btu/lbm)
HL	saturated liquid enthalpy (Btu/lbm)
HV	saturated vapor enthalpy (Btu/lbm)
PSIA	pressure (psia)
RJ	conversion factor = 0.185053
S	entropy (Btu/lbm°R)

SL	saturated liquid entropy (Btu/lbm°R)
SV	saturated vapor entropy (Btu/lbm°R)
T	temperature (°R)
V	specific volume (ft^3/lbm)
VL	saturated liquid specific volume (ft^3/lbm)
VV	saturated vapor specific volume (ft^3/lbm)


START
FLQDEN
RJ = 0.185053
ENTHAL
DRVPT
DELTH = RJ*DPSDT *(V-VL)*T
HL = H-DELTH
ENTROP
SL = S-DELTH/T
VV = V
HV = H
SV = S
RETURN

```
C      ***********************************************************************
C      *                                                                     *
C      *                              S A T N                                *
C      *                                                                     *
C      *   THIS SUBROUTINE COMPUTES THE SATURATED VAPOR-LIQUID               *
C      *PROPERTIES GIVEN TEMPERATURE AND SPECIFIC VOLUME.                    *
C      *                                                                     *
C      ***********************************************************************
       SUBROUTINE SATN(PSIA,T,V,HV,SV,HL,SL,VL)
       COMMON/FREON1/Q(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
       CALL FLQDEN(VL,T,DVLDT)
       RJ=0.185053
       CALL ENTHAL(PSIA,T,V,H,DHDT,DHDV)
       CALL DRVPT(DPSDT,PSIA,T,DPSDT2)
       DELTH=RJ*DPSDT*(V-VL)*T
       HL=H-DELTH
       CALL ENTROP(T,V,S,DSDT,DSDV)
       SL=S-DELTH/T
       VV=V
       HV=H
       SV=S
       RETURN
       END
```

SFIND DESCRIPTION

SFIND searches for the specific volume and temperature in the vapor region given pressure and entropy. A two variable Newton-Raphson iteration technique is used for the search.

EQUATIONS

$$\text{Jacobian} = \left(\frac{\partial P(V,T)}{\partial V}\right)\left(\frac{\partial S(V,T)}{\partial T}\right) - \left(\frac{\partial P(V,T)}{\partial T}\right)\left(\frac{\partial S(V,T)}{\partial V}\right)$$

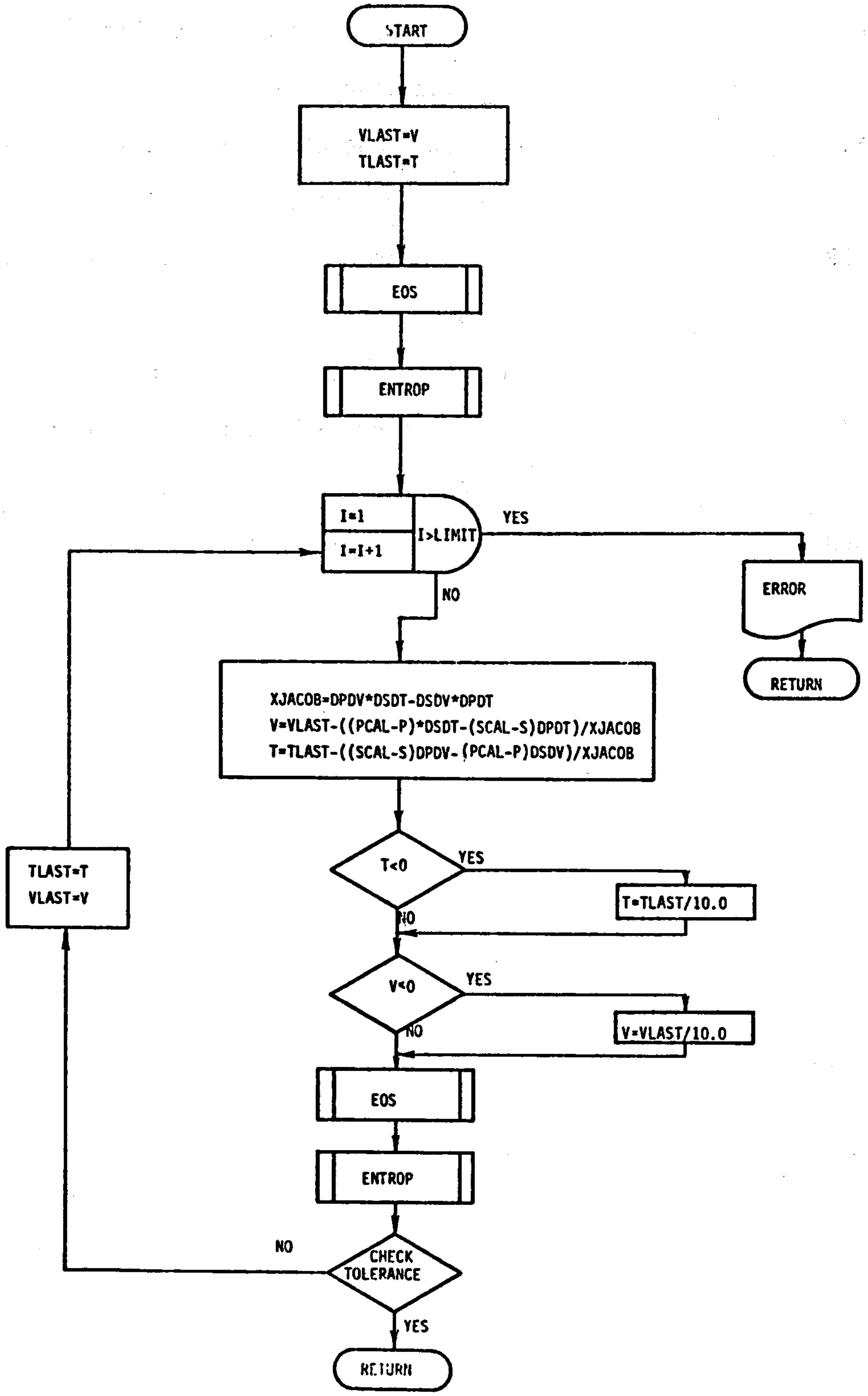
$$V_{i+1} = V_i - \left[\frac{(P(V,T)-P_{KNOWN})\frac{\partial S(V,T)}{\partial T} - (S(V,T)-S_{KNOWN})\frac{\partial P(V,T)}{\partial T}}{\text{Jacobian}}\right]_i$$

$$T_{i+1} = T_i - \left[\frac{(S(V,T)-S_{KNWON})\frac{\partial P(V,T)}{\partial V} - (S(V,T)-S_{KNOWN})\frac{\partial P(V,T)}{\partial T}}{\text{Jacobian}}\right]_i$$

VARIABLE SYMBOL TABLE

BTOLER	lower tolerance on convergence and saturation tests
DPDT	$\frac{\partial P(V,T)}{\partial T}$
DPDV	$\frac{\partial P(V,T)}{\partial V}$
DPDVDT	$\frac{\partial^2 P(V,T)}{\partial V \partial T}$
D2PDT2	$\frac{\partial^2 P(V,T)}{\partial T^2}$
DSDT	$\frac{\partial S(V,T)}{\partial T}$
DSDV	$\frac{\partial S(V,T)}{\partial V}$
I	index
LIMIT	maximum number of iterations allowed in any one search
M2	unit number for printer
P	known pressure (psia)

PCAL	calculated pressure (psia)
S	known entropy (Btu/lbm°R)
SCAL	calculated entropy (Btu/lbm°R)
T	temperature (°R)
TLAST	last temperature calculated, to be used in next iteration
TTOLER	upper tolerance on convergence and saturation tests
V	specific volume (ft^3/lbm)
VLAST	last specific volume calculated, to be used in next iteration
XJACOB	Jacobian of pressure and entropy


START
VLAST=V
TLAST=T
EOS
ENTROP
I=1
I=I+1
I>LIMIT
YES
NO
ERROR
RETURN
XJACOB=DPDV*DSDT-DSDV*DPDT
V=VLAST-((PCAL-P)*DSDT-(SCAL-S)DPDT)/XJACOB
T=TLAST-((SCAL-S)DPDV-(PCAL-P)DSDV)/XJACOB
T<0
YES
T=TLAST/10.0
NO
V<0
YES
V=VLAST/10.0
NO
EOS
ENTROP
TLAST=T
VLAST=V
NO
CHECK TOLERANCE
YES
RETURN

```
C     ******************************************************************
C     *                                                                *
C     *                           S F I N D                            *
C     *                                                                *
C     *   THIS SUBROUTINE FINDS TEMPERATURE AND DENSITY GIVEN PRESSURE *
C     *AND ENTROPY FOR SUPERHEATED VAPOR STATES.                       *
C     *                                                                *
C     ******************************************************************
      SUBROUTINE SFIND(T,P,V,S)
      PARAMETER M2=6
      COMMON/FREON1/G(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
      VLAST=V
      TLAST=T
      CALL EOS(PCAL,T,V,DPDT,DPDV,DPDVDT,D2PDT2)
      CALL ENTROP(T,V,SCAL,DSDT,DSDV)
      DO 1000 I=1,LIMIT
      XJACOB=DPDV*DSDT-DPDT*DSDV
      V=VLAST-((PCAL-P)*DSDT-(SCAL-S)*DPDT)/XJACOB
      T=TLAST-((SCAL-S)*DPDV-(PCAL-P)*DSDV)/XJACOB
      IF(T.LE.0.0) T=TLAST/10.0
      IF(V.LE.0.0) V=VLAST/10.0
      CALL ENTROP(T,V,SCAL,DSDT,DSDV)
      CALL EOS(PCAL,T,V,DPDT,DPDV,DPDVDT,D2PDT2)
      IF(PCAL/P.LT.TTOLER.AND.PCAL/P.GT.BTOLER.AND.SCAL/S.LT.
     1    TTOLER.AND.SCAL/S.GT.BTOLER) GO TO 1050
      IF(ABS(T-TLAST).LT.1.0E-04.AND.ABS(V-VLAST).LT.1.0E-07) GO TO 1050
      VLAST=V
      TLAST=T
 1000 CONTINUE
      WRITE(M2,9000) LIMIT
 1050 RETURN
 9000 FORMAT(' ',10X,'***SFIND FAILED TO CONVERGE IN ',I5,' ITERATIONS')
      END
```

VOLUME DESCRIPTION

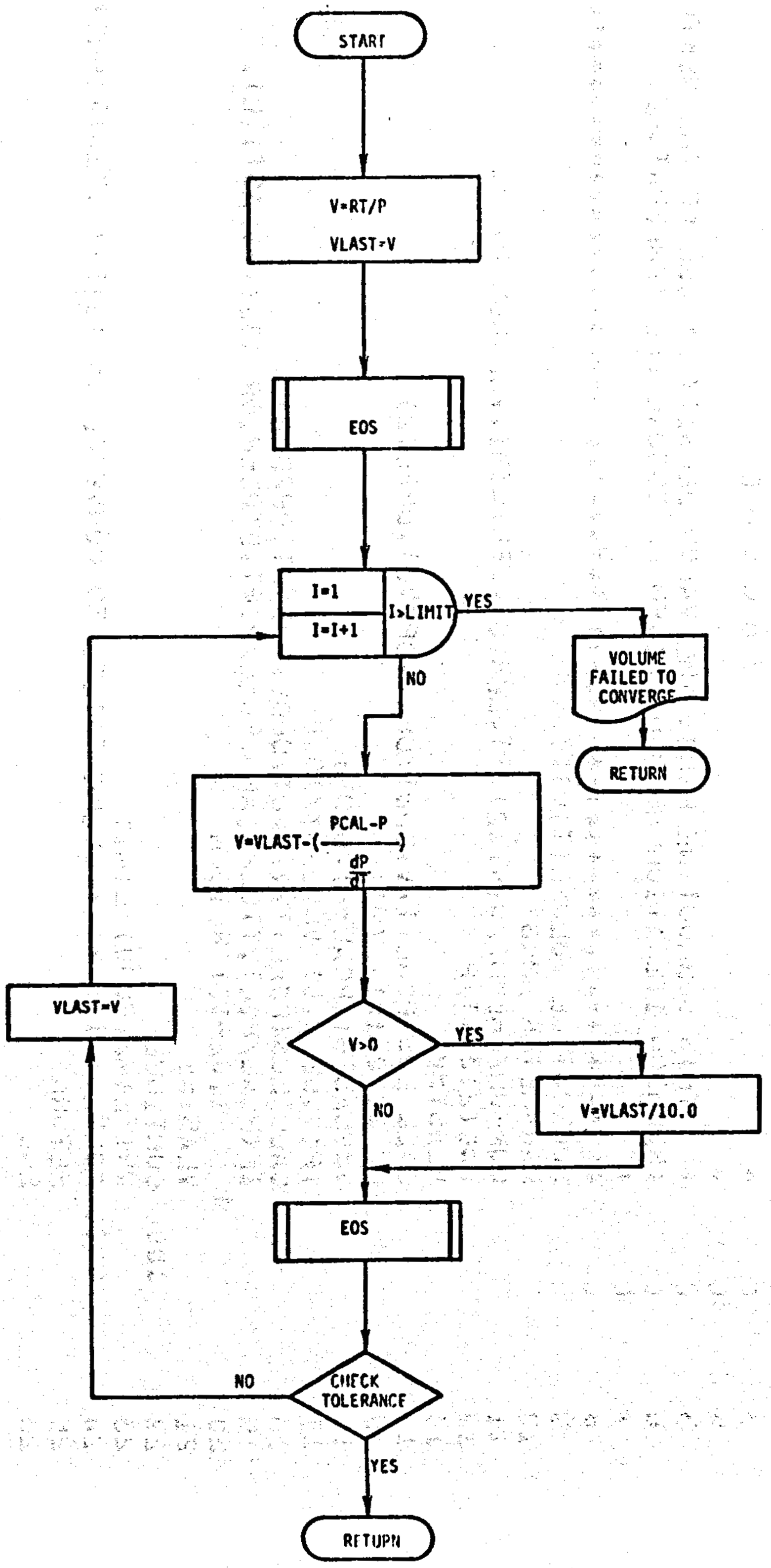
VOLUME searches for the specific volume in the vapor region given temperature and pressure by using a single variable Newton-Raphson iteration technique.

EQUATIONS

$$V_{i+1} = V_i - \left[\frac{P(V,T)-P_{KNOWN}}{\frac{\partial P(V,T)}{\partial V}}\right]_i$$

VARIABLE SYMBOL TABLE

BTOLER	lower tolerance on convergence and saturation tests
DPDT	$\frac{\partial P(V,T)}{\partial T}$
DPDV	$\frac{\partial P(V,T)}{\partial V}$
DPDVDT	$\frac{\partial^2 P(V,T)}{\partial V \partial T}$
DP2DT2	$\frac{\partial^2 P(V,T)}{\partial T^2}$
I	index
LIMIT	maximum number of iterations allowed in any one search
M2	unit number for printer
P	known pressure (psia)
PCAL	calculated pressure (psia)
Q	2-dimensional array containing the coefficients for the fluid to be evaluated
T	temperature (°R)
TTOLER	upper tolerance on convergence and saturation tests
V	specific volume (ft^3/lbm)
VLAST	last specific volume calculated, V_i to be used in next iteration (ft^3/lbm)


START
V=RT/P
VLAST=V
EOS
I=1
I=I+1
I>LIMIT
YES
VOLUME FAILED TO CONVERGE
RETURN
NO
V=VLAST-(PCAL-P / dP/dT)
VLAST=V
V>0
YES
V=VLAST/10.0
NO
EOS
CHECK TOLERANCE
NO
YES
RETURN

```
C     ******************************************************************
C     *                                                                *
C     *                          V O L U M E                           *
C     *                                                                *
C     *   THIS SUBROUTINE FINDS THE VAPOR DENSITY BY USE OF THE NEWTON-*
C     *RAPHSON ITERATION METHOD, GIVEN PPRESSURE AND TEMPERATURE.      *
C     *                                                                *
C     ******************************************************************
      SUBROUTINE VOLUME(T,P,V)
      PARAMETER M2=6
      COMMON/FREON1/G(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
      V=G(2,3)*T/P
      VLAST=V
      CALL EOS(PCAL,T,V,DPDT,DPDV,DPDVDT,DP2DT2)
      DO 1000 I=1,LIMIT
      V=VLAST-((PCAL-P)/DPDV)
      IF(V.LE.0.0) V=VLAST/10.0
      CALL EOS(PCAL,T,V,DPDT,DPDV,DPDVDT,DP2DT2)
      IF(PCAL/P.LT.TTOLER.AND.PCAL/P.GT.BTOLER.OR.ABS(V-VLAST).LT.
     1    1.0E-10) RETURN
      VLAST=V
1000  CONTINUE
      WRITE(M2,9000) LIMIT
9000  FORMAT( 11X,'***VOLUME FAILED TO CONVERGE IN ',I5,' ITERATIONS')
      RETURN
      END
```

VPEQM DESCRIPTION

VPEQM calculates the saturation pressure given temperature or searches for the saturation temperature given pressure. The Newton-Raphson iteration technique is used for the saturation temperature search.

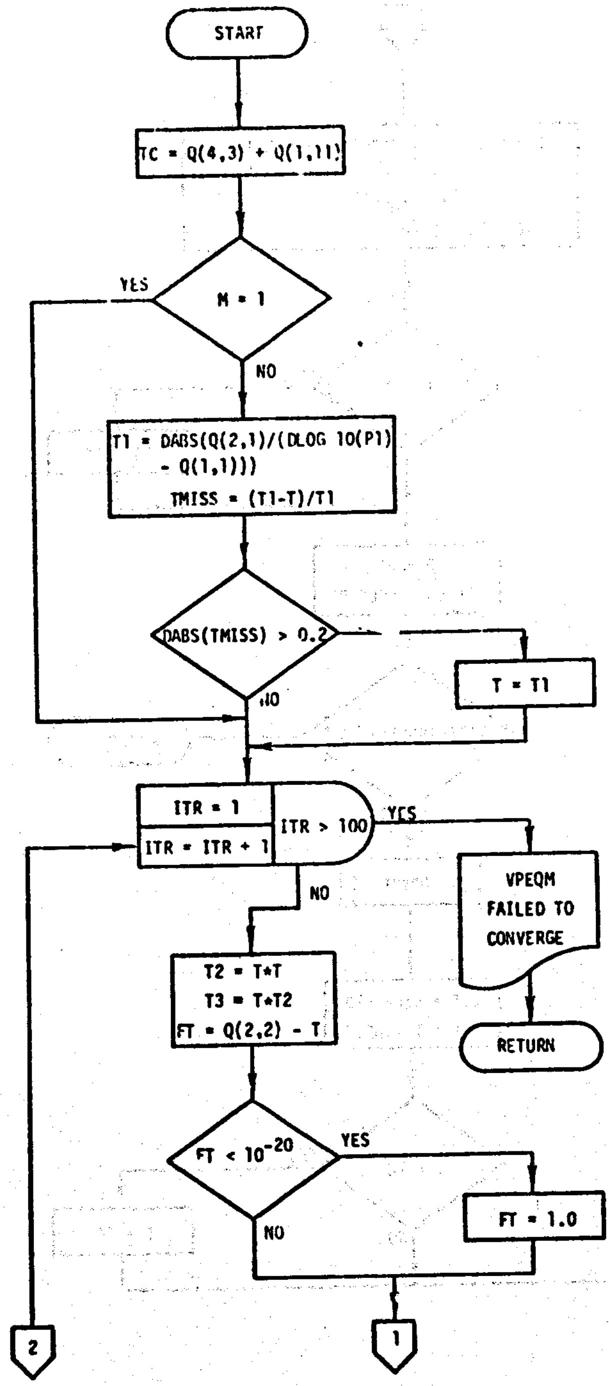
EQUATIONS

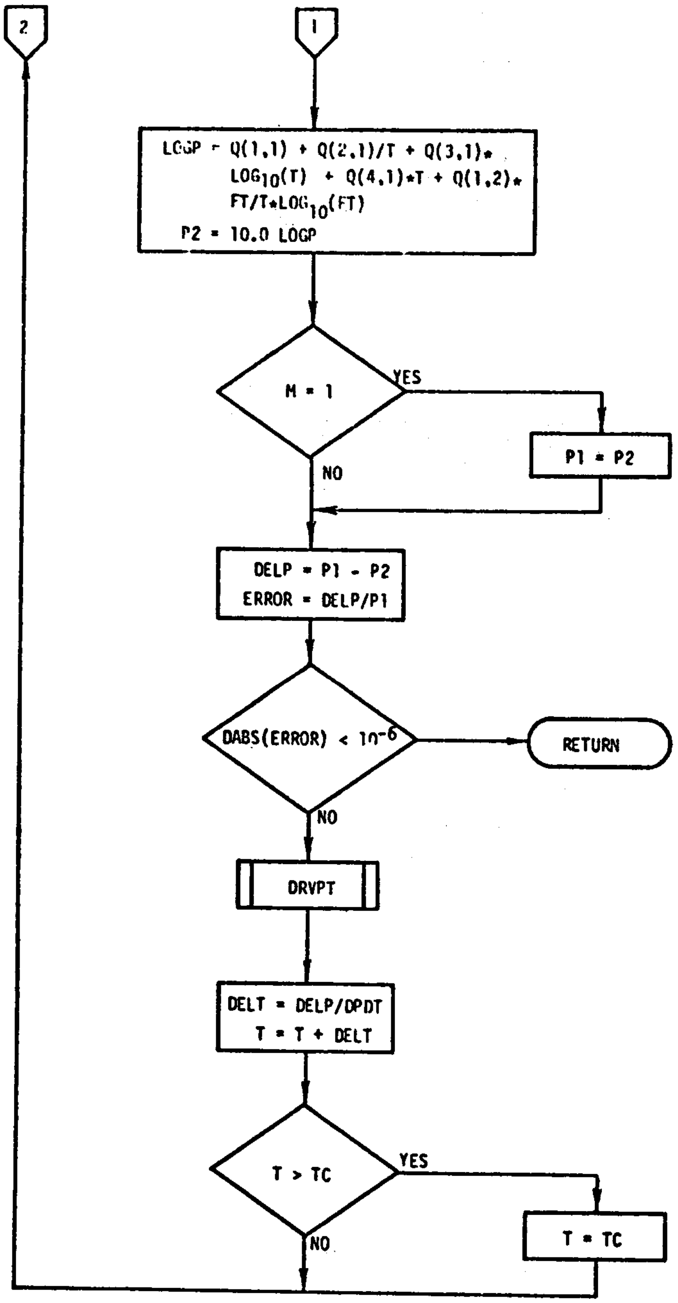
$$\log_{10}P_{SAT} = A+B/T+C*\log_{10}T+D*T+E\left(\frac{F-T}{T}\right)\log_{10}(F-T)$$

VARIABLE SYMBOL TABLE

DELP	P_{KNOWN}
DELT	$\dfrac{P_{KNOWN} - P_{CALCULATED}}{\frac{dP_{SAT}}{dT}}$ - change in temperature for next temperature iteration
DPDT	$\dfrac{dP_{SAT}}{dT}$
D2PDT2	$\dfrac{d^2P_{SAT}}{dT^2}$
ERROR	$\dfrac{P_{KNOWN} - P_{CALCULATED}}{P_{KNOWN}}$
FT	F-T
ITR	index
LIMIT	maximum number of iterations allowed in any one search
LOGP	$\log_{10}P_{SAT}$
M	flag indicating which of temperature or pressure is known $m = 1$ temperature is known else pressure is known
P1	known pressure (psia)
P2	calculated pressure (psia)
Q	2-dimensional array containing the coefficients for the fluid to be evaluated

T	temperature (°R)
TC	critical temperature (°R)
T1	initial guess for temperature search (°R)
T2	T^2
T3	T^3
TMISS	difference of initial guess and the value of temperature passed to VPEQM


START
TC = Q(4,3) + Q(1,11)
M = 1
YES
NO
T1 = DABS(Q(2,1)/(DLOG 10(P1) - Q(1,1)))
TMISS = (T1-T)/T1
DABS(TMISS) > 0.2
NO
T = T1
ITR = 1
ITR = ITR + 1
ITR > 100
YES
NO
VPEQM FAILED TO CONVERGE
RETURN
T2 = T*T
T3 = T*T2
FT = Q(2,2) - T
FT < 10^-20
YES
NO
FT = 1.0
2
1


2
1
LOGP = Q(1,1) + Q(2,1)/T + Q(3,1)*
LOG10(T) + Q(4,1)*T + Q(1,2)*
FT/T*LOG10(FT)
P2 = 10.0 LOGP
M = 1
YES
P1 = P2
NO
DELP = P1 - P2
ERROR = DELP/P1
DABS(ERROR) < 10-6
RETURN
NO
DRVPT
DELT = DELP/DPDT
T = T + DELT
T > TC
YES
T = TC
NO

```
C     ******************************************************************
C     *                                                                *
C     *                           V P E Q M                            *
C     *   THIS SUBROUTINE CALCULATES THE SATURATION PRESSURE OR        *
C     *TEMPERATURE GIVEN THE OTHER                                     *
C     *                                                                *
C     ******************************************************************
      SUBROUTINE VPEQM(P1,T,M)
      PARAMETER M2=6
      COMMON/FREON1/Q(4,12),TTOLER,BTOLER,LIMIT,DERROR,NTYPE
      REAL LOGP
      TC=Q(4,3)+Q(1,11)
      IF (M.EQ.1) GO TO 100
      T1 = ABS(Q(2,1)/(ALOG10(P1)-Q(1,1)))
      TMISS = (T1-T)/ T1
      IF (ABS(TMISS).GT..2D + 00) T = T1
  100 DO 105 ITR=1,LIMIT
      T2 = T*T
      T3 = T*T2
      FT = Q(2,2)-T
      IF (FT.LT..1E-20) FT = 1.0
      LOGP=Q(1,1)+Q(2,1)/T+Q(3,1)*ALOG10(T)+Q(4,1)*T+Q(1,2)*FT/T*ALOG10(
     1 FT)
      P2 = 10.0 **LOGP
      IF(M.EQ.1 ) P1 = P2
      DELP = P1-P2
      ERROR = DELP/P1
      IF(ABS(ERROR).LT..1D-06) GO TO 110
      CALL DRVPT(DPDT,P2,T,D2PDT2)
      DELT = DELP/DPDT
      T = T + DELT
      IF(T.GT.TC) T=TC
  105 CONTINUE
      WRITE (M2, 115) LIMIT
```

```
  110 RETURN
  115 FORMAT(1H ,4X,'***VPEQM FAILED TO CONVERGE IN',I6,'ITERATIONS')
      END
```

References

1. Mechtly, E. A., The International System of Units, Physical Constants and Conversion Factors, 2nd Revision, NASA, Washington, D.C. 1973
2. Downing, R. C., and Knight, B. W., Computer Program for Calculating Properties of "FREON" Refrigerants, RT-52, Du Pont Chemical Corporation, 1971
3. General Equations, X-88F, Du Pont Chemical Corporation

*U.S. GOVERNMENT PRINTING OFFICE: 1977-740-306/262. Region 4.