NOTICE
This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

A UNIVERSITY OF UTAH
DIRECT CONTACT GEOTHERMAL POWER
PROJECT REPORT

A Computer Program
for Determining the Thermodynamic
Properties of Water

MASTER

Prepared for the
Energy Research and Development Administration
Under

Grant No. AER 75-01039

by

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

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

JUN 11 1976

15598

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

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.

NOMENCLATURE

A_{ij}	constants required for properties
C_i	constants required for ψ_0
F_i	constants required for saturation pressure
H	enthalpy per lbm
i	index of summation
j	index of summation
P	pressure
P_c	critical pressure
P_s	saturation pressure
Q	part of ψ , function of density and temperature
R	universal gas constant
S	entropy per lbm
t	temperature in °C
T	temperature in °K
t_c	critical temperature in °C
U	internal energy per lbm
V	specific volume per lbm
τ	dimensionless temperature $1000^\circ\text{K}/T^\circ\text{K}$
ρ	density
τ_{ai}	constant dependent on j
ρ_{aj}	constant dependent on j
ψ	Helmholtz free energy
ψ_0	Helmholtz free energy which is only temperature dependent

ABSTRACT

This program was written to be used as a subroutine. The program determines the thermodynamic properties of water given any of the following pairs of knowns to define a thermodynamic state: pressure and entropy, pressure and enthalpy, pressure and quality, temperature and pressure, or temperature and quality. These five pairs of knowns allow the user to evaluate any thermodynamic cycle using water, as a working fluid. The basic equations came from Keenan, Keyes, Hill and Moore, Steam Tables, John Wiley and Sons, 1969. This report includes a complete derivation of equations, program listing, program symbol description, a complete set of flow charts and a sample steam turbine calculation.

INTRODUCTION

This subroutine computes the thermodynamic properties of water. The subroutine was written to allow for a wide range of applications such that it may be used to determine the properties of water in a conventional steam power plant or, as the authors developed it, for use in evaluating geothermal power cycles. The routine can be treated as a "black box" for the user with a limited background in programming or the program may be modified to the user's needs. This manual describes the general use of subroutine WATER and appendices contain a detailed derivation and explanation of WATER.

DEVELOPMENT OF PROPERTIES

The basic equation used in deriving the thermodynamic properties of water is the Helmholtz free energy equation since all properties can be determined from it¹.

$$\psi = \psi_0(T) + RT (\ln \rho + \rho Q(\rho, T)) \quad (1)$$

where

$$\psi_0(T) = \sum_{i=1}^6 C_i / \tau^{i-1} + C_7 \ln T + \frac{C_8 \ln T}{\tau} \quad (2)$$

and

$$Q = (\tau - \tau_c) \sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} \left[\sum_{i=1}^8 A_{ij} (\rho - \rho_{aj})^{i-1} + e^{-E\rho} \sum_{i=9}^{10} A_{ij} \rho^{i-9} \right] \quad (3)$$

As stated previously all of the thermodynamic properties can be developed from (1) and they are as follows:

$$P(\rho, T) = \rho RT(1 + \rho Q(\rho, T) + \rho^2 \left(\frac{\partial Q(\rho, T)}{\partial \rho} \right)_T) \quad (4)$$

$$U(\rho, T) = 1000R\rho \left(\frac{\partial Q(\rho, T)}{\partial T} \right) + \frac{d(\psi_0 T)}{dT} \quad (5)$$

$$H(\rho, T) = RT \left(\rho T \left(\frac{\partial Q(\rho, T)}{\partial T} \right)_\rho + 1 + Q(\rho, T) + \rho^2 \left(\frac{\partial Q(\rho, T)}{\partial \rho} \right)_T \right) + \frac{d(\psi_0 T)}{dT} \quad (6)$$

$$S(\rho, T) = -R \left(\ln \rho + \rho Q(\rho, T) - \rho T \left(\frac{\partial Q(\rho, T)}{\partial T} \right)_\rho \right) - \frac{d\psi_0}{dT} \quad (7)$$

For a complete derivation of all equations see Appendix I. Recall that a thermodynamic state requires that two properties be known and for saturation states pressure and temperature are not independent. For evaluating an ideal power cycle the following processes are employed, constant entropy (turbine and pump), constant enthalpy (expansion valve), constant pressure (boiler or condenser). In order to determine properties for these processes, properties must be determined with one of the following pairs known: temperature and pressure, temperature and quality, pressure and quantities, a search technique must be used to find the density. Considering one of these at a time, first for temperature and pressure as known quantities, a search technique must be used to find the density in order that the remaining properties can be determined. A search is required because the pressure is a function of temperature and density and the equation cannot be solved explicitly for density. For all searches in WATER the Newton-Raphson iteration technique was used. This technique was selected because of its generally rapid convergence. Second, the saturation states can be solved in a similar manner if the saturation pressure were a function of temperature, such a function is;¹

$$P_s = P_c \exp\left(\tau 10^{-5} (t_c - t) \sum_{i=1}^8 F_i (0.65 - 0.01t)^{i-1}\right) \quad (8)$$

This equation allows the saturation states to be solved just as the temperature and pressure case, by searching for both the saturated liquid and saturated vapor roots to Equation (4). After finding the saturated liquid and vapor properties the effect of a given quality can be determined. Third, for cases of pressure and entropy or enthalpy a modified version of the Newton-Raphson method was used.² In these cases a search for two unknowns must be carried out, temperature and density, (see Appendix I for a complete derivation).

PROGRAM DEVELOPMENT

Subroutine WATER was written in FORTRAN V for the UNIVAC 1108 series computer but should easily be adapted to any similar FORTRAN compiler. Mixed mode arithmetic (real and integers) is required for subroutine PROPER, however, this can be easily changed for compilers which do not allow mixed mode calculations. 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 user's needs, the appendices contain a complete documentation to allow the user to modify WATER. Subroutines were used throughout to allow for easy modification and as a means of determining any possible problems. The names of the subroutines used and a brief description appear in Table 1. All quantities required by these subroutines are global variables (common to all subroutines) placed in blank common. The general flow chart for WATER and detailed flow charts for all subroutines are contained in Appendix II. The program listing and symbol table are in Appendix III.

Table 1. Listing of Subroutines Used in WATER and a Description of Their Functions

<u>SUBROUTINE NAME</u>	<u>DESCRIPTION</u>
WATER	Main routine-determines process to be evaluated and calls proper routines
PSATUR	Computes saturation pressure given temperature
TSATUR	Determines saturation temperature given pressure by using Newton-Raphson method
DFIND	Determines density given pressure and temperature by using Newton-Raphson method
PROPER	Calculates all properties and their derivatives for use in search routines
LIQDEN	Sets up search for saturated liquid properties
VAPDEN	Sets of search for saturated vapor properties
PHFIND	Performs search for temperature and density given pressure and enthalpy for superheated vapor or compressed liquid by using a modified Newton-Raphson method
PSFIND	Performs search for temperature and density given pressure and entropy for superheated vapor or compressed liquid by using a modified Newton - Raphson method
PRINT	Prints out results only if NPRINT=1

USE OF WATER

In order to use WATER the user must supply to the subroutine the following information (in the order that they appear in the parameter list):

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

PX-pressure and quality

PH-pressure and enthalpy

PS-pressure and entropy

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 WATER

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. The line number from which the call on WATER originated.
(This is used to aid the user in locating an error should WATER terminate due to improper data)
5. State number. This is an arbitrary number assigned by the user to a location in his thermodynamic cycle (used only if print out is desired, to inform the user which state in the thermodynamic cycle properties are being determined).
6. Print out selecton--if =1 then print out, $\neq 1$ then no print out.

WATER will return the following information (in the order they appear in the parameter list:

1. Quality-applies to saturation states (if this is greater than 1 then it indicates superheat and if it is less than 0 it indicates a compressed liquid state)
2. Entropy (BTU/lbm-°R)
3. Enthalpy (BTU/lbm)
4. Specific volume (ft^3/lbm)

```
10 CALL WATER ('TP',TEMP,PRESS,10,1,1,X,S,H,V,U,T,P,SL,HL,VL,UL,  
SV,HV,VV,UV, NERROR)
```

Appendix IV contains a sample run to illustrate the use of WATER. A sample steam turbine is evaluated using WATER to show the use of WATER in cycle evaluations.

For a complete analysis of the accuracy of this generalized equation for thermodynamic properties see Reference 1. The values found from this program may vary slightly with those tabulated in Reference 1 due to the fact that on convergence tests a tolerance of ± 0.5 percent was used.

APPENDIX I

DERIVATION OF ALL EQUATIONS REQUIRED FOR WATER

A. Derivation of Properties

9

From Reference (1)

$$\psi = \psi_0(T) + RT(\ln \rho + \rho Q(\rho, \tau)) \quad (1)$$

where

$$\psi_0(T) = \sum_{i=1}^6 \frac{C_i}{\tau^{i-1}} + C_7 \ln T + \frac{C_8 \ln T}{\tau} \quad (2)$$

and

$$Q = (\tau - \tau_c) \sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} \left[\sum_{i=1}^8 A_{ij} (\rho - \rho_{aj})^{i-1} + e^{-E\rho} \sum_{i=9}^{10} A_{ij} \rho^{i-9} \right] \quad (3)$$

From Reference (3)

$$S = - \left(\frac{\partial \psi}{\partial T} \right)_\rho \quad (AI.1)$$

$$P = \rho^2 \left(\frac{\partial \psi}{\partial \rho} \right)_T \quad (AI.2)$$

$$U = \left(\frac{\partial (\psi T)}{\partial \tau} \right)_\rho \quad (AI.3)$$

To determine S substitute Equation (2) into Equation (AI.1)

$$S = - \frac{\partial}{\partial T} (\psi_0(T) + RT(\ln \rho + \rho Q(\rho, \tau)))$$

differentiating

$$S = - \left(\frac{d\psi_0(T)}{dT} \right) + (\ln \rho + \rho Q(\rho, \tau) + RT\rho \left(\frac{\partial Q(\rho, \tau)}{\partial T} \right)_\rho)$$

differentiating τ

$$d\tau = \frac{-\tau}{T} dT$$

substituting in

$$\frac{\partial Q(\rho, \tau)}{\partial \tau} = \frac{\partial Q(\rho, \tau)}{\partial \tau}; \quad \frac{d\tau}{dT} = \frac{-\tau}{T} \frac{\partial Q(\rho, \tau)}{\partial \tau}$$

$$S = -R(\ln \rho + \rho Q(\rho, \tau) - \rho \tau \left(\frac{\partial Q(\rho, \tau)}{\partial \tau} \right)_\rho) - \frac{d\psi_0(T)}{dT} \quad (AI.4)$$

To determine P, substitute (2) into (AI.2)

$$P = \rho^2 \frac{\partial}{\partial \rho} (\psi_0(T) + RT(\ln \rho + \rho Q(\rho, \tau)))_T$$

differentiating

$$P = \rho^2 RT(1/\rho + Q(\rho, \tau) + \rho(\frac{\partial Q(\rho, \tau)}{\partial \rho})_\tau)$$

simplifying

$$P = \rho RT(1 + \rho Q(\rho, \tau) + \rho^2 (\frac{\partial Q(\rho, \tau)}{\partial \rho})_\tau) \quad (\text{AI.5})$$

To determine U, multiplying (2) by τ

$$\psi\tau = \tau\psi_0(T) + RT\tau(\ln \rho + \rho Q(\rho, \tau))$$

substituting $\tau = 1000/T$

$$\psi\tau = \tau\psi_0(T) + 1000R(\ln \rho + Q(\rho, \tau))$$

substituting this into (2) and differentiating

$$U = \frac{d(\psi_0(T)\tau)}{d\tau} + 1000R (\frac{\partial Q(\rho, \tau)}{\partial \tau})_\rho \quad (\text{AI.6})$$

To determine H, use definition of $H = u + Pv$ or $H = u + P/\rho$

substituting in U and P

$$H = RT (\rho\tau(\frac{\partial Q(\rho, \tau)}{\partial \tau})_\rho + 1 + \rho Q(\rho, \tau) + \rho^2 (\frac{\partial Q(\rho, \tau)}{\partial \rho})_\tau) + \frac{d(\psi_0(T)\tau)}{d\tau} \quad (\text{AI.7})$$

To determine P, U and S requires the following derivatives $(\frac{\partial Q}{\partial \rho})_T$, $(\frac{\partial Q}{\partial \rho})_\rho$,

$$\frac{d(\psi_0(T)\tau)}{d\tau}, \quad \frac{d\psi_0(T)}{dT}$$

I. $\left(\frac{\partial Q}{\partial \rho}\right)_\tau$

differentiating (3)

$$\left(\frac{\partial Q}{\partial \rho}\right)_\tau = (\tau - \tau_c) \sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} \left(\sum_{i=1}^8 (A_{ij}(i-1)(\rho - \rho_{aj})^{i-2} \right. \\ \left. - E e^{-E\rho} \sum_{i=9}^{10} A_{ij} \rho^{i-9} + e^{-E\rho} \sum_{i=9}^{10} A_{ij} (i-9) \rho^{i-10} \right)$$

evaluating $\sum_{i=9}^{10} A_{ij} (i-9) \rho^{i-10} = A_{10,j}$

collecting terms

$$\left(\frac{\partial Q}{\partial \rho}\right)_\tau = (\tau - \tau_c) \sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} \left[\sum_{i=1}^8 A_{ij} (i-1) (\rho - \rho_{aj})^{i-2} \right. \\ \left. + e^{-E\rho} \left(-E \sum_{i=9}^{10} A_{ij} \rho^{i-9} + A_{10,j} \right) \right] \quad (A1.8)$$

II. $\left(\frac{\partial Q}{\partial \tau}\right)_\rho$

Let $B_j = \sum_{i=1}^8 A_{ij} (\rho - \rho_{aj})^{i-1} + \sum_{i=9}^{10} e^{-E\rho} A_{ij} \rho^{i-9}$

therefore $Q = (\tau - \tau_c) \sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} B_j$

differentiating

$$\left(\frac{\partial Q}{\partial \tau}\right)_\rho = \sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} B_j + (\tau - \tau_c) \sum_{j=1}^7 (j-2) (\tau - \tau_{aj})^{j-3} B_j$$

collecting terms and substituting in B_j

$$\left(\frac{\partial Q}{\partial \tau}\right)_\rho = \sum_{j=1}^7 [(\tau - \tau_c)(j-2)(\tau - \tau_{aj})^{j-3} + (\tau - \tau_{aj})^{j-2}] \left(\sum_{i=1}^8 A_{ij} (\rho - \rho_{aj})^{i-1} + \sum_{i=9}^{10} e^{-E_{\rho} A_{ij}} \rho^{i-9}\right) \quad (\text{AI.9})$$

III. $\frac{d(\psi_o(T)\tau)}{d\tau}$

multiplying (2) by τ

$$\psi_o(T)\tau = \sum_{i=1}^6 \frac{C_i}{\tau^{i-2}} + C_1 \tau \ln T + C_8 \ln T$$

noting that $\ln T = \ln \frac{1000}{\tau} = \ln 1000 - \ln \tau$ differentiating and collecting terms

$$\frac{d(\psi_o(T)\tau)}{d\tau} = \sum_{i=1}^6 \frac{C_i(2-i)}{\tau^{i-1}} + C_1(\ln T - 1) - \frac{C_8}{\tau} \quad (\text{AI.10})$$

IV. $\frac{d\psi_o(T)}{dT}$

substituting $\tau = \frac{1000}{T}$ into (1), and differentiating and simplifying

$$\frac{d\psi_o(T)}{dT} = \sum_{i=1}^6 \frac{C_i(i-1)\tau^{2-1}}{1000} + \frac{C_7}{T} + \frac{C_8}{1000} (\ln T + 1) \quad (\text{AI.11})$$

Derivation of density search given temperature and pressure (DFIND)
basic iteration Newton-Raphson equation

$$\rho_{i+1} = \rho_i - \left[\frac{P(\rho, T) - P_{\text{known}}}{\left(\frac{\partial P}{\partial \rho}\right)_T} \right]_i$$

evaluating $(\frac{\partial P}{\partial \rho})_T$ by

differentiating AI.5 and simplifying

$$(\frac{\partial P}{\partial \rho})_T = RT (1 + 2\rho Q + 4\rho^2 (\frac{\partial Q}{\partial \rho})_T + \rho^3 (\frac{\partial^2 Q}{\partial \rho^2})_T) \quad (A.12)$$

$$(\frac{\partial P}{\partial \rho})_T \text{ requires } (\frac{\partial^2 Q}{\partial \rho^2})_T$$

$$V. (\frac{\partial^2 Q}{\partial \rho^2})_T$$

differentiating AI.8

$$(\frac{\partial^2 Q}{\partial \rho^2}) = (\tau - \tau_c) \sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} [\sum_{i=1}^8 A_{ij} (i-1)(i-2)(\rho - \rho_{aj})^{i-3}$$

$$-E e^{-E\rho} [-E \sum_{i=9}^{10} A_{ij} \rho^{i-9} + A_{10,j}] + e^{-E\rho} [-E A_{10,j}]]$$

collecting terms

$$(\frac{\partial^2 Q}{\partial \rho^2})_T = (\tau - \tau_c) \sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} [\sum_{i=1}^8 A_{ij} (i-1)(i-2)(\rho - \rho_{aj})^{i-3} + e^{-E\rho}$$

$$[E^2 \sum_{i=9}^{10} A_{ij} \rho^{i-9} - 2E A_{10,j}]] \quad (AI.13)$$

Derivation of saturated temperature search given pressure (TSATUR).

basic Newton-Raphson iteration equation

$$T_{i+1} = T_i - \left[\frac{P_s(T_i)}{(\frac{\partial P_s}{\partial T})_i} \right]$$

$$\frac{\partial P_s}{\partial T} = P_c \exp[\tau 10^{-5} (t_c - t)] \frac{\sum_{i=1}^8 F_i [0.65 - 0.01 t]^{i-1}}{d \left(\frac{1000}{T} 10^{-5} (t_c - t) \sum_{i=1}^8 F_i (0.65 - 0.01 t)^{i-1} \right)} dT$$

simplifying

$$\frac{dP_s}{dT} = \frac{P_s}{100} \cdot \frac{d \left(\left(\frac{t_c}{T} - \frac{t}{T} \right) \sum_{i=1}^8 F_i (0.65 - 0.01 t)^{i-1} \right)}{dT}$$

differentiating and simplifying

$$\begin{aligned} \frac{dP_s}{dT} = \frac{P_s}{100} & \left(\left(\frac{t_c - t}{T^2} - \frac{1}{T} \right) \sum_{i=1}^8 F_i (0.65 - 0.01 t)^{i-1} \right. \\ & \left. + \left(-\frac{t_c}{T} = -\frac{t}{T} \right) \sum_{i=1}^8 F_i (0.01) (i-1) (0.65 - 0.01 t)^{i-2} \right) \quad (\text{AI.14}) \end{aligned}$$

Derivation of temperature and density search given pressure and entropy or enthalpy (PSFIND or PHFIND)

basic iteration equations

$$\begin{aligned} \rho_{i+1} &= \rho_i - \left[\frac{(P - P_{\text{known}}) \left(\frac{\partial I}{\partial T} \right)_\rho - (I - I_{\text{known}}) \left(\frac{\partial P}{\partial T} \right)_\rho}{J(P, I)} \right]_i \\ T_{i+1} &= T_i - \left[\frac{(I - I_{\text{known}}) \left(\frac{\partial P}{\partial \rho} \right)_T - (P - P_{\text{known}}) \left(\frac{\partial I}{\partial \rho} \right)_T}{J(P, I)} \right]_i \end{aligned}$$

where I is either H or S

$$\text{and } J(P, I) = \left(\frac{\partial P}{\partial \rho} \right)_T \left(\frac{\partial I}{\partial T} \right)_\rho - \left(\frac{\partial I}{\partial \rho} \right)_T \left(\frac{\partial P}{\partial T} \right)_\rho$$

$$A. \left(\frac{\partial P}{\partial T} \right)$$

differentiating AI.5

$$\left(\frac{\partial P}{\partial T} \right) = \rho R (1 + \rho Q + \rho^2 \left(\frac{\partial Q}{\partial \rho} \right)_T + T \rho \left(\left(\frac{\partial Q}{\partial T} \right)_\rho + \left(\frac{\partial^2 Q}{\partial \rho \partial T} \right) \right))$$

substituting in $(\frac{\partial Q}{\partial T})_{\rho} = (\frac{\partial Q}{\partial \tau})_{\rho} (-\frac{\tau}{T})$

$$(\frac{\partial P}{\partial T})_{\rho} = \rho R(1 + \rho Q + \rho^2 (\frac{\partial Q}{\partial \rho})_T + T\rho (-\frac{\tau}{T} (\frac{\partial Q}{\partial \tau})_{\rho} + \rho (\frac{\partial^2 Q}{\partial \rho \partial T}))) \quad (AI.15)$$

B. $(\frac{\partial P}{\partial \rho})_T$ - was evaluated for density search (DFIND)

C. $(\frac{\partial S}{\partial T})_{\rho}$

differentiating equation AI.4

$$(\frac{\partial S}{\partial T}) = \frac{dT}{dT} \frac{\partial S}{\partial \tau}$$

$$= -\frac{\tau}{T} (-R(\rho (\frac{\partial Q}{\partial \tau})_{\rho} - \rho (\frac{\partial Q}{\partial \tau})_{\rho} - \rho \tau (\frac{\partial^2 Q}{\partial \tau^2})_{\rho}) - \frac{\partial}{\partial \tau} \frac{d\psi}{dT})$$

simplifying

$$= -\frac{\tau^2}{T} \rho R \frac{\partial^2 Q}{\partial \tau^2} + \frac{\tau}{T} \frac{dT}{d\tau} \frac{d^2 \psi}{dT^2}$$

substituting in $\frac{dT}{d\tau} = -\frac{T}{\tau}$

$$\frac{\partial S}{\partial T} = -\frac{\tau^2}{T} R \rho \frac{\partial^2 Q}{\partial \tau^2} - \frac{d^2 \psi}{dT^2} \quad (AI.16)$$

D. $(\frac{\partial S}{\partial \rho})_T$

differentiating equation AI.4

$$(\frac{\partial S}{\partial \rho})_T = -R(1/\rho + \rho (\frac{\partial Q}{\partial \rho})_T + Q - \tau (\frac{\partial Q}{\partial \tau})_{\rho} - \rho \tau \frac{\partial^2 Q}{\partial \tau \partial \rho})$$

noting that

$$\frac{\partial^2 Q}{\partial \tau \partial \rho} = \frac{\partial}{\partial \rho} (\frac{\partial Q}{\partial \tau}) = \frac{\partial}{\partial \rho} (\frac{dT}{d\tau} (\frac{\partial Q}{\partial T}))$$

where

$$\frac{dT}{d\tau} = -\frac{T}{\tau}$$

$$(\frac{\partial S}{\partial \rho})_T = -R(1/\rho + Q + \rho (\frac{\partial Q}{\partial \rho})_T - \tau (\frac{\partial Q}{\partial \tau})_{\rho} + T\rho \frac{\partial^2 Q}{\partial \rho \partial T}) \quad (AI.17)$$

E. $(\frac{\partial H}{\partial T})_{\rho}$

$$\frac{\partial H}{\partial T} = \frac{d\tau}{dT} \left(\frac{\partial H}{\partial \tau} \right)$$

differentiating AI.7 with respect to τ

$$\begin{aligned} \frac{\partial H}{\partial \tau} = & -\frac{R(1000)}{\tau^2} (\rho\tau \left(\frac{\partial Q}{\partial \tau} \right)_{\rho} + 1 + \rho Q + \rho^2 \left(-\frac{\partial Q}{\partial \rho} \right)_{\tau}) \\ & + \frac{R(1000)}{\tau} \left(2\rho \frac{\partial Q}{\partial \tau} + \rho\tau \frac{\partial^2 Q}{\partial \tau^2} + \rho^2 \frac{\partial^2 Q}{\partial \rho \partial \tau} \right) + \frac{d^2(\psi_0(T)\tau)}{d\tau^2} \end{aligned}$$

collecting terms and substituting

$$\frac{\partial^2 Q}{\partial \rho \partial \tau} = -\frac{T}{\tau} \frac{\partial^2 Q}{\partial T \partial \rho}$$

$$\frac{d\tau}{dT} = -\frac{\tau}{T}$$

$$\left(\frac{\partial H}{\partial T} \right)_{\rho} = -R \left(1 + \rho Q + \rho^2 \left(\frac{\partial Q}{\partial \rho} \right)_{\tau} - \tau \rho \left(\frac{\partial Q}{\partial \tau} \right)_{\rho} - \rho \tau^2 \left(\frac{\partial^2 Q}{\partial \tau^2} \right)_{\rho} \right.$$

$$\left. + T \rho^2 \frac{\partial^2 Q}{\partial \rho \partial T} \right) - \frac{\tau}{T} \frac{d^2(\psi_0(T)\tau)}{d\tau^2} \quad (\text{AI.18})$$

F. $(\frac{\partial H}{\partial \rho})_T$

differentiating AI.7

$$\begin{aligned} \left(\frac{\partial H}{\partial \rho} \right)_T = & RT \left(\tau \left(\frac{\partial Q}{\partial \tau} \right)_{\rho} + \rho \tau \left(\frac{\partial^2 Q}{\partial \tau \partial \rho} \right) + Q + 3\rho \left(\frac{\partial Q}{\partial \rho} \right)_T \right. \\ & \left. + \rho^2 \left(\frac{\partial^2 Q}{\partial \rho^2} \right)_T \right) \end{aligned}$$

substituting in $\frac{\partial^2 Q}{\partial \tau \partial \rho} = -\frac{T}{\tau} \frac{\partial^2 Q}{\partial T \partial \rho}$

$$\left(\frac{\partial H}{\partial \rho} \right)_T = RT \left(\tau \left(\frac{\partial Q}{\partial \tau} \right)_{\rho} - T \rho \left(\frac{\partial^2 Q}{\partial T \partial \rho} \right) + Q + 3\rho \left(\frac{\partial Q}{\partial \rho} \right)_T + \rho^2 \left(\frac{\partial^2 Q}{\partial \rho^2} \right)_T \right) \quad (\text{AI.19})$$

All of the following derivatives of Q are required for the preceding derivatives.

VI. $\left(\frac{\partial^2 Q}{\partial \tau^2}\right)_\rho$

differentiating AI.9

let

$$M_\rho = \sum_{i=1}^8 A_{ij} (\rho - \rho_{aj})^{i-1} + e^{-E\rho} \sum_{i=9}^{10} A_{ij} \rho^{i-9}$$

$$\begin{aligned} \left(\frac{\partial^2 Q}{\partial \tau^2}\right)_\rho &= \sum_{j=1}^7 [(j-2)(\tau - \tau_{aj})^{j-3} + (\tau - \tau_c)(j-2)(j-3)(\tau - \tau_{aj})^{j-4} \\ &\quad + (j-2)(\tau - \tau_{aj})^{j-3}] [M_\rho] \end{aligned}$$

collecting terms

$$\begin{aligned} \left(\frac{\partial^2 Q}{\partial \tau^2}\right)_\rho &= \sum_{j=1}^7 [2(j-2)(\tau - \tau_{aj})^{j-3} + (\tau - \tau_c)(j-2)(j-3)(\tau - \tau_{aj})^{j-4}] \\ &\quad \left[\sum_{i=1}^8 A_{ij} (\rho - \rho_{aj})^{i-1} + e^{-E\rho} \sum_{i=9}^{10} A_{ij} \rho^{i-9} \right] \quad (\text{AI.20}) \end{aligned}$$

VII. $\frac{\partial^2 Q}{\partial \rho \partial \tau}$

differentiating AI.8 with respect to τ

$$\frac{\partial^2 Q}{\partial \rho \partial \tau} = -\frac{T}{T} \frac{\partial^2 Q}{\partial \rho \partial \tau}$$

$$\text{let } N_\rho = \sum_{i=1}^8 A_{ij} (i-1)(i-2)(\rho - \rho_{aj})^{i-2} + e^{-E} \left(-E \sum_{i=9}^{10} A_{ij} \rho^{i-9} + A_{10,j} \right)$$

$$\frac{\partial^2 Q}{\partial \rho \partial \tau} = \sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} N_{\rho} + (\tau - \tau_c) \sum_{j=1}^7 (j-2)(\tau - \tau_{aj})^{j-2} N_{\rho}$$

collecting terms

$$\frac{\partial^2 Q}{\partial \rho \partial \tau} = \sum_{j=1}^7 [(\tau - \tau_{aj})^{j-2} + (\tau - \tau_c)(j-2)(\tau - \tau_{aj})^{j-2}]$$

$$\left[\sum_{i=1}^8 A_{ij} (i-1)(i-2)(\rho - \rho_{aj})^{i-2} + e^{-E\rho} (-E \sum_{i=9}^{10} A_{ij} \rho^{i-9} + A_{10,j}) \right] \quad (\text{AI.21})$$

VIII. $\frac{d^2 \psi_{\alpha}(T)}{dT^2}$

differentiating AI.11

$$\frac{d^2 \psi_{\alpha}(T)}{dT^2} = \sum_{i=1}^6 \frac{(i-1)(i-2)}{10^6} C_i \tau^{3-i} - \frac{C_7}{T^2} + \frac{C_8}{1000T} \quad (\text{AI.22})$$

IX. $\frac{d^2(\psi_{\alpha}(T)\tau)}{d\tau^2}$

differentiating AI.10

$$\frac{d^2(\psi_{\alpha}(T)\tau)}{d\tau^2} = \sum_{i=1}^6 (2-i)(1-i) \frac{C_i}{\tau^i} - \frac{C_7}{\tau} + \frac{C_8}{\tau^2} \quad (\text{AI.23})$$

APPENDIX II

General Flow Chart of Routine

WATER and Detailed Flow Charts of all Subroutines

GENERALIZED FLOWCHART OF SUBROUTINE WATER

DETAILED FLOWCHART OF SUBROUTINE WATER

DETAILED FLOWCHART OF SUBROUTINE DFIND

DETAILED FLOWCHART OF SUBROUTINE LIQDEN

DETAILED FLOWCHART OF SUBROUTINE VAPDEN

DETAILED FLOWCHART OF SUBROUTINE PHFIND

DETAILED FLOWCHART OF SUBROUTINE PRINT

DETAILED FLOWCHART OF SUBROUTINE PROPER

DETAILED FLOWCHART OF SUBROUTINE PSATUR

DETAILED FLOWCHART OF SUBROUTINE PSFIND

DETAILED FLOWCHART OF SUBROUTINE TSATUR

APPENDIX III

Complete Program Listing and Symbol

Table Defining All Variables

Used in WATER and Its Subroutines


```

1      C      #####
2      C      #
3      C      #
4      C      #
5      C      #
6      C      #
7      C      #
8      C      #
9      C      #
10     C      #
11     C      #
12     C      #
13     C      #
14     C      #
15     C      #
16     C      #
17     C      #####
18     C      SUBROUTINE WATER(CYCLE,FGIVEN,SGIVEN,CALLIN,NSTAT1,NPRINA,QUAL1,
19     C      1  ENTR1,ENTH1,VOL1,UINT1,TEMPF1,PRESS1,LSATS1,LSATH1,LSATV1,
20     C      2  LSATU1,VSATS1,VSATH1,VSATV1,VSATU1,NERROR)
21     C      DATA TTOLER,BTOLER,CONVR/1.005,0.995,0.4299229/
22     C      DATA TCRIT,PCRT,TAUCRT,GAS1,GAS/374.136,3203.6,1.544912,66.9366,
23     C      1  0.46151/
24     C      DATA ITERA1/200/
25     C      COMMON VSATS, VSATH, VSATU, VSATV, LSATS, LSATH, LSATU, LSATV,
26     C      1  TEMPK, TCRIT, PCRT, TAUCRT, TGUSS, GAS, GAS1, CONVR,
27     C      2  DFDOT, DGUSS, DENSIT, DPDRHO, TEMPF, PRESS, UINT, ENTH, VOL,
28     C      3  ENTR, DHDT, DHDRHO, DPDT, SCAL, HCAL, DSDRHO, SJOULE, SKNOWN,
29     C      4  ITERAT, PSAT, DENTOL, TTOLER, BTOLER, PCAL, QUAL, PKNOWN, DSDT
30     C      COMMON/PRINTA/NPRINT,NSTATE
31     C      PARAMETER M2=6,M1=5
32     C      IMPLICIT REAL(L)
33     C      INTEGER CALLIN
34     C      NSTATE=NSTAT1
35     C      NPRINT=NPRINA

```

```

36      TEMPF=TEMPF1
37      ENTR=ENTR1
38      ENTH=ENTH1
39      PRESS=PRESS1
40      LSATH=0.0
41      LSATU=0.0
42      LSATV=0.0
43      LSATS=0.0
44      VSATH=0.0
45      VSATU=0.0
46      VSATV=0.0
47      VSATS=0.0
48      NERROR=0.0
49      WRITE(M2,9000)
50      IF (CYCLE.EQ.'TP') GO TO 1930
51      IF (CYCLE.EQ.'TX'.OR.CYCLE.EQ.'PX') GO TO 1720
52      IF (CYCLE.EQ.'PS') GO TO 1100
53      IF(CYCLE.EQ.'PH') GO TO 1330
54      WRITE (M2,9010) CYCLE, CALLIN
55      NERROR=1
56      RETURN

```

C-----PRESSURE-ENTROPY-----

```

57      1100 SKNOWN=SGIVEN*1.8
58      PKNOWN=FGIVEN
59      PRESS=FGIVEN
60      ENTR=SGIVEN
61      WRITE(M2,9050)
62      IF(PRESS.LT.PCRIT) GO TO 1200
63      TEMPR=500.0
64      IF(ENTR.GT.1.0580) DENSIT=1.0E-05
65      IF(ENTR.GT.1.0580) GO TO 1550
66      DENSIT=0.90
67      GO TO 1650
68      1200 CALL TSATUR
69      CALL VAPDER
70      IF(ENTR/SCAL.GT.TTOLER) GO TO 1550
71

```

```

72      CALL LIQDEN
73      IF(ENTR/SCAL.LT.BTOLER) GO TO 1630
74      QUAL=(ENTR-LSATS)/(VSATS-LSATS)
75      ENTH=(1.0-QUAL)*LSATH + QUAL*VSATH
76      UINT=(1.0-QUAL)*LSATU + QUAL*VSATU
77      VOL =(1.0-QUAL)*LSATV + QUAL*VSATV
78      CALL PRINT
79      GO TO 2200
80  C-----PRESSURE-ENTHALPY-----
81      1330 PKNOWN=FGIVEN
82          HKNOWN=SGIVEN
83          PRESS =FGIVEN
84          ENTH=SGIVEN
85          WRITE(M2,9060)
86          IF(PRESS.LT.PCRIT) GO TO 1400
87          TEMPR=500.0
88          IF(ENTH.GT.902.5) DENSIT=1.0E-05
89          IF(ENTH.GT.902.5) GO TO 1550
90          DENSIT=0.96
91          GO TO 1630
92      1400 CALL TSATUR
93          CALL VAPDEN
94          IF(HKNOWN/HCAL.GT.TTOLER) GO TO 1550
95          CALL LIQDEN
96          IF(HKNOWN/HCAL.LT.BTOLER) GO TO 1630
97          QUAL=(HKNOWN-LSATH)/(VSATH-LSATH)
98          ENTR=(1.0-QUAL)*LSATS + QUAL*VSATS
99          UINT=(1.0-QUAL)*LSATU + QUAL*VSATU
100         VOL =(1.0-QUAL)*LSATV + QUAL*VSATV
101         CALL PRINT
102         GO TO 2200
103      1550 DENTOL=1.0E-10
104          IF (CYCLE.EQ.'PH') CALL PHFIND
105          IF (CYCLE.EQ.'PS') CALL PSFIND
106          ENTH=HCAL
107          ENTR=SCAL

```

```

108      IF(PRESS.GT.PCRIT) QUAL=-3.0
109      IF(PRESS.LE.PCRIT) QUAL=+2.0
110      CALL PRINT
111      GO TO 2200
112      1630 DENTOL=1.0E-05
113      IF(CYCLE.EQ.'PH') GO TO 1680
114      TEMPK=180.0*(5.4+9.630628*ENTR-2.640989*SQRT(ENTR)-0.10592*ENTR**2
115      1)
116      DENSIT=10.0/(62.4283*(0.12      -.5276609*ENTR+.2878426*SQRT(ENTR)
117      1      +.4208526*ENTR**2))
118      CALL PSFIND
119      GO TO 1700
120      1680 TEMPK=1.8*(512.0+1.370531*ENTH+4.737845*SQRT(ENTH)+4.540232E-04*
121      1      ENTH**2)
122      DENSIT=1.0E+04/(62.4283*(149.2-.185626*ENTH+2.671239*SQRT(ENTH)
123      1      +.3508786E-03*ENTH**2))
124      CALL PHFIND
125      1700 ENTH=HCAL
126      ENTR=SCAL
127      IF(PRESS.GT.PCRIT) QUAL=-3.0
128      IF(PRESS.LE.PCRIT) QUAL=-2.0
129      CALL PRINT
130      GO TO 2200
131      1720 QUAL=SGIVEN
132      IF (QUAL.GE.0.0) GO TO 1780
133      WRITE (M2,9020)
134      QUAL=0.0
135      1780 IF (QUAL.LE.1.0) GO TO 1810
136      WRITE (M2,9030)
137      QUAL=1.0
138      C-----PRESSURE-QUALITY-----
139      1810 IF (CYCLE.EQ.'TX') GO TO 1870
140      PRESS =FGIVEN
141      PRKNOWN=FGIVEN
142      WRITE(M2,9070)
143      IF(PRESS.LE.PCRIT*0.99) GO TO 1890

```

```

144      NERROR=1
145      WRITE(M2,9100)
146      RETURN
147      1890 CALL TSATUR
148      GO TO 2120
149      C-----TEMPERATURE-QUALITY -----
150      1870 TEMPF=FGIVEN
151      WRITE(M2,9080)
152      IF(TEMPF.LE.705.0) GO TO 1900
153      NERROR=1
154      WRITE(M2,9110)
155      RETURN
156      1900 TEMPK=(TEMPF+459.67)/1.8
157      CALL PSATUR
158      PRESS =PSAT
159      GO TO 2120
160      C-----PRESSURE-TEMPERATURE-----
161      1930 TEMPF=FGIVEN
162      PRESS =SGIVEN
163      WRITE(M2,9090)
164      TEMPK=(TEMPF+459.67)/1.8
165      IF(TEMPF.LT.705.0) GO TO 1960
166      IF(PRESS.GT.PCRIT) GO TO 2080
167      GO TO 2035
168      1960 CALL PSATUR
169      IF(PRESS /PSAT.GT.TTOLER.OR.PRESS /PSAT.LT.BTOLER) GO TO 2030
170      WRITE(M2,9040) CALLIN,TEMPF,PRESS
171      NERROR=1
172      RETURN
173      2030 IF(PRESS /PSAT.GT.TTOLER) GO TO 2080
174      2035 GVAL=2.0
175      DGUESS=1.0E-05
176      DENTOL=1.0E-10
177      GO TO 2090
178      2080 DGUESS=0.99
179      DENTOL=1.0E-07

```

```

180      IF (TEMPF.LT.705.) QUAL=-2.0
181      IF (TEMPF.GE.705.) QUAL=-3.0
182      2090 CALL DFIND
183      ENTH=HICAL
184      ENTR=SCAL
185      CALL PRINT
186      GO TO 2200
187      2120 IF (ABS(QUAL).LT.0.000001) GO TO 2150
188      CALL VAPDEN
189      IF (ABS(QUAL-1.0).LT.0.000001) GO TO 2160
190      2150 CALL LIQDEN
191      2160 ENTR=(1.0-QUAL)*LSATS + QUAL*VSATS
192      UINT=(1.0-QUAL)*LSATU + QUAL*VSATU
193      ENTH=(1.0-QUAL)*LSATH + QUAL*VSATH
194      VOL =(1.0-QUAL)*LSATV + QUAL*VSATV
195      CALL PRINT
196      2200 TEMPF1=TEMPF
197      VOL1=VOL
198      ENTR1=ENTR
199      ENTH1=ENTH
200      UINT1=UINT
201      PRESS1=PRESS
202      QUAL1=QUAL
203      LSATS1=LSATS
204      LSATH1=LSATH
205      LSATU1=LSATU
206      LSATV1=LSATV
207      VSATS1=VSATS
208      VSATH1=VSATH
209      VSATU1=VSATU
210      VSATV1=VSATV
211      9000 FORMAT(1H1,/,/,13X,'*****      THERMODYNAMIC PROPERTIES OF WATER
212      1  *****',/,/,/)
213      9010 FORMAT(1H ,4X,'***ERROR*** CYCLE = ',A3,' CYCLE MUST EQUAL TP, TX,
214      1 PX, PS, PH, LAST CALL ON WATER CAME FROM LINE',I6,/,/,10X,'WATER T
215      ERMINATED')

```

```

216 9020 FORMAT(1H,4X,'***ERROR*** A QUALITY OF < 0 IS NOT DEFINED. ASSUME
217      10 TO BE ZERO')
218 9030 FORMAT(1H,4X,'***ERROR*** A QUALITY OF > 1 IS NOT DEFINED. ASSUME
219      10 TO BE ONE')
220 9040 FORMAT(1H,4X,'***ERROR*** TEMPERATURE AND PRESSURE ARE NOT INDEPE
221      INENT UNDER THE SATURATION DOME',//,4X,'WATER TERMINATED, LAST CAL
222      2L CAME FROM LINE ',16,/,10X,'TEMPERATURE = ',F15.5,10X,'PRESSURE =
223      3 ',F15.5)
224 9050 FORMAT(1H,16X,'PRESSURE AND ENTROPY')
225 9060 FORMAT(1H,16X,'PRESSURE AND ENTHALPY')
226 9070 FORMAT(1H,16X,'PRESSURE AND QUALITY')
227 9080 FORMAT(1H,16X,'TEMPERATURE AND QUALITY')
228 9090 FORMAT(1H,16X,'TEMPERATURE AND PRESSURE')
229 9100 FORMAT(1H,4X,'***ERROR*** A SATURATION STATE DOES NOT EXIST FOR P
230      1PRESSURES ABOVE THE CRITICAL PRESSURE(3203.6PSIA), WATER TERMINATED
231      2 ')
232 9110 FORMAT(1H,4X,'***ERROR*** A SATURATION STATE DOES NOT EXIST FOR T
233      1TEMPERATURES ABOVE THE CRITICAL TEMPERATURE(705 DEG.F), WATER TERMI
234      2NATED')
235      RETURN
236      END

```



```
36      WRITE(M2,9000) ITERAT
37      9000  FORMAT(' ***TEMPERATURE-DENSITY SEARCH, GIVEN PRESSURE & ENTHALPY,
38             1FAILED TO CONVERGE IN ',I5,' ITERATIONS, RESULTS MAY NOT BE VALID'
39             2 )
40      RETURN
41      END
```



```

35 9040 FORMAT(1H,10(/),16X,'PROPERTIES OF WATER AT STATE ',I4,5(/))
36 9050 FORMAT(1H,20X,'TEMPERATURE = ',F7.2,' DEG.F',/,
37 1 21X,'PRESSURE = ',F10.5,' PSIA')
38 9060 FORMAT(1H,20X,'SUPERHEATED VAPOR',////////)
39 9070 FORMAT(1H,20X,'COMPRESSED LIQUID',////////)
40 9080 FORMAT(1H,20X,'SATURATED LIQUID-VAPOR EQUILIBRIUM',/,21X,
41 1 'QUALITY = ',F6.4)
42 9090 FORMAT(1H,20X,'INTERNAL ENERGY (BTU/LBM)',5X,F9.3,8X,F9.3)
43 9100 FORMAT(1H,20X,'ENTHALPY',8X,'(BTU/LBM)',5X,F9.3,8X,F9.3)
44 9120 FORMAT(1H,20X,'SPECIFIC VOLUME (CU.FT/LBM )',5X,F9.4,8X,F9.4,////
45 1 //)
46 9110 FORMAT(1H,20X,'ENTROPY',9X,'(BTU/LBM-R)',5X,F9.6,8X,F9.6)
47 9130 FORMAT(1H,/,51X,'SATURATED',8X,'SATURATED',/,51X,'VAPOR',12X,
48 1 'LIQUID',/,21X,60('H'))
49 9140 FORMAT(1H,////////)
50 RETURN
51 END
52

```

```

1 C $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
2 C $
3 C $ P R O P E R $
4 C $ $
5 C $ COMPUTES PROPERTIES AND THEIR DERIVATIVES. $
6 C $
7 C $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
8 SUBROUTINE PROPER
9 COMMON VSAIS, VSATH, VSATU, VSATV, LSATS, LSATH, LSATU, LSATV,
10 1 TEMPK, TCRIT, PCRIT, TAUCT, TGUSS, GAS, GAS1, CONVR,
11 2 DPSUT, DGUSS, DENSIT, DPDRHO, TEMPF, PRESS, UINT, ENTH, VOL,
12 3 ENTR, UHDT, UHDRHO, DPDT, SCAL, HCAL, DSDRHO, SJOULE, SKNOWN,
13 4 ITERAT, PSAT, DENTOL, TTOLER, BTOLER, PCAL, QUAL, PKNOWN, DSDT
14 IMPLICIT REAL(L)
15 DIMENSION A(10,7), C(8)
16 DATA ((A(I,J), J=1,7), I=1,10)/ 29.492937, -5.198586, 6.8335354,
17 1 -0.1564104, -0.3972405, -3.9661401, -0.69048554, -132.13917,
18 2 7.7779182, -26.149751, -0.72546108, 26.409282, 15.453061,
19 3 2.7407416, 274.64632, -33.301902, 65.326396, -9.2734289,
20 4 -47.740374, -29.142470, -5.1028070, -360.93828, -16.254622,
21 5 -26.181978, 4.312584, 56.32313, 29.568796, 3.9636085, 342.1843
22 6 1, -177.51074, 0.0, 0.0, 0.0, 0.0, 0.0, -244.50042, 127.48742,
23 7 0.0, 0.0, 0.0, 0.0, 0.0, 155.18535, 137.46153, 0.0, 0.0, 0.0,
24 8 0.0, 0.0, 5.9728487, 155.97836, 0.0, 0.0, 0.0, 0.0, 0.0,
25 9 -410.30848, 337.31180, -137.46618, 6.7874983, 136.87317,
26 A 79.84797, 13.041253, -416.0586, -209.88866, -733.96848,
27 B 10.401717, 645.8188, 399.1757, 71.531353 /
28 DATA (C(I), I=1,8) /1857.065, 3229.12, -419.465, 36.6649, -20.5516
29 1 4.85233, 46.0, -1011.249/
30 VOL=1.0/(62.4283*DENSIT)
31 TAU=1000.0/TEMPK
32 Z3SUM=0.0
33 Z4SUM=0.0
34 Z5SUM=0.0
35 DZGUTA=0.0

```

```

36      W2SUM=0.0
37      Z9SUM=0.0
38      DO 2890 J=1,7
39      IF(J.EQ.1) GO TO 2680
40      DCON=1.0
41      TAUCON=2.5
42      GO TO 2700
43 2680 DCON=0.634
44      TAUCON=TAUCRT
45 2700 CONTINUE
46      ZSUM=0.0
47      Z2SUM=0.0
48      WSUM=0.0
49      DO 2740 I=1,8
50      ZSUM=ZSUM+A(I,J)*(DENSIT-DCON)**(I-1)
51      Z2SUM=Z2SUM+A(I,J)*(I-1)*(DENSIT-DCON)**(I-2)
52      WSUM=WSUM+A(I,J)*(I-1)*(I-2)*(DENSIT-DCON)**(I-3)
53 2740 CONTINUE
54      Z1SUM=0.0
55      DO 2780 I=9,10
56      Z1SUM=Z1SUM+A(I,J)*DENSIT**(I-9)
57 2780 CONTINUE
58      Z1SUM=Z1SUM*EXP(-4.8*DENSIT)
59      W1SUM=Z1SUM*23.04
60      Z3SUM=Z3SUM+(TAU-TAUCON)**(J-2)*(ZSUM+Z1SUM)
61      Z4SUM=Z4SUM+(TAU-TAUCON)**(J-2)*(Z2SUM-4.8*Z1SUM+EXP(-4.8*DENSIT)
62      1 *A(10,J))
63      Z5SUM=Z5SUM+((TAU-TAUCON)**(J-2)+(TAU-TAUCRT)*(J-2)*(TAU-TAUCON)
64      1 *(J-3))*(ZSUM+Z1SUM)
65      Z9SUM=Z9SUM+(J-2)*(TAU-TAUCON)**(J-3)*(Z2SUM-4.8*Z1SUM+
66      1 EXP(-4.8*DENSIT)*A(10,J))*(TAU-TAUCRT)
67      W2SUM=W2SUM+(TAU-TAUCON)**(J-2)*(WSUM+W1SUM-9.6*EXP(-4.8*DENSIT)
68      1 *A(10,J))
69      DZQDTA=DZQDTA+((TAU-TAUCRT)*(J-2)*(J-3)*(TAU-TAUCON)**(J-4)+
70      1 2.0*(J-2)*(TAU-TAUCON)**(J-3))*(ZSUM+Z1SUM)
71 2890 CONTINUE

```

```

72      Q=(TAU-TAUCRT)*Z3SUM
73      DQDRHO=(TAU-TAUCRT)*Z4SUM
74      DQDTAU=Z5SUM
75      Z6SUM=0.0
76      Z7SUM=0.0
77      Z8SUM=0.0
78      X7SUM=0.0
79      DO 2990  I=1,6
80      X7SUM=X7SUM+(2-I)*(1-I)*C(1)/TAU**I
81      Z6SUM=Z6SUM+(2-I)*C(1)/TAU**(I-1)
82      Z7SUM=C(1)*(1-I)*1AU**(2-I)/1000.0 +Z7SUM
83      Z8SUM=Z8SUM+C(1)*(I-1)*(1-2)*TAU**(3-I)
84 2990 CONTINUE
85      D2CCT2=Z8SUM*1.0E-06-C(7)/TEMPK**2+C(8)/(1000.0*TEMPK)
86      D2QDD2=(TAU-TAUCRT)*W2SUM
87      D2CTDT=X7SUM-C(7)/TAU+C(8)/TAU**2
88      DCTDT=Z6SUM+C(7)*(LOG(TEMPK)-1.0)-C(8)/TAU
89      DCDT=Z7SUM+C(7)/TEMPK+C(8)/1000.0*(LOG(TEMPK)+1.0)
90      PCAL=DENSIT*GAS1*TEMPK*(1.0+Q*DENSIT+DENSIT**2*DQDRHO)
91      UINT=CONVR*(1000.0*GAS*DENSIT*DQDTAU+DCTDT)
92      SCAL=CONVR*(-GAS*(LOG(DENSIT)+DENSIT*Q-DENSIT*TAU*DQDTAU)
93 1      -DCDT)/1.8
94      SCOLE=SCAL*1.8
95      HCAL=CONVR*(GAS*TEMPK*(DENSIT*TAU*DQDTAU+1.0+DENSIT*Q
96 1      +DENSIT**2*DQDRHO)+DCTDT)
97      DQDDDT=(Z9SUM+Z4SUM)*(-TAU/TEMPK)
98      DPDRHO=GAS1*TEMPK*(1.0+2.0*Q*DENSIT+4.0*DENSIT**2*DQDRHO
99 1      +DENSIT**3*D2QDD2)
100     DDDT=CONVR*(GAS*(1.0+DENSIT*Q+DENSIT**2*DQDRHO-TAU*DENSIT*DQDTAU
101 1      -DENSIT*TAU**2*D2QDDT+TEMPK*DENSIT**2*DQDDDT)-TAU/TEMPK*D2CTDT)
102     DDDRHO=GAS*CONVR*TEMPK*(TAU*DQDTAU-TEMPK*DENSIT*DQDDDT+Q
103 1      +3.0*DENSIT*DPDRHO+DENSIT**2*D2QDD2)
104     DDDT=DENSIT*GAS1*(1.0+DENSIT*Q+DENSIT**2*DQDRHO+TEMPK*DENSIT*
105 1      (-TAU/TEMPK*DQDTAU+DENSIT*DQDDDT))
106     DSDRHO=-CONVR*GAS*(1.0/DENSIT+Q+DENSIT*DPDRHO-TAU*DQDTAU
107 1      +DENSIT*TEMPK*DQDDDT)

```



```
108      USDT=-CONVR*(CAS*TAU**2/TEMPK*DENSIT*D2GDTA+D2CUT2)/1.8
109      RETURN
110      END
```



```
36      WRITE(M2,9000) ITERAT
37      9000 FORMAT(' ***TEMPERATURE-DENSITY SEARCH GIVEN PRESSURE AND ENTROPY
38      1FAILED TO CONVERGE IN ',I5,' ITERATIONS, RESULTS MAY NOT BE VALID'
39      2 )
40      RETURN
41      END
```


<u>Variable Name</u>	<u>Description</u>
A(I,J)	Constants required for Q
BTOLER	Lower tolerance on convergence and saturation tests
C(I)	Constants required for Ψ_0
CALLIN	Line number from which call on Water originated
CONVR	Conversion factor-converts joules/g to BTU/lbm = 0.4299229 ⁴
CYCLE	Type of calculation to be made, CYCLE has the following possible pairs of knowns: <ol style="list-style-type: none"> 1) Temperature and Pressure = TP 2) Temperature and Quality = TX 3) Pressure and Quality = PX 4) Pressure and Enthalpy = PH 5) Pressure and Entropy = PS
DCDT	$\frac{d\Psi_0}{dT}$
DCON	$\rho_{ij} \begin{cases} 0.634 & \text{for } j=1 \\ 1.0 & \text{for } j>1 \end{cases}$
DCTDT	$\frac{d\Psi_{or}}{dT}$
DENSIT	Density
DENTOL	Convergence tolerance on density in density search(DFIND)
DGUESS	Initial guess for density in DFIND when Temperature and Pressure are known
DHDRHO	$\left(\frac{\partial h}{\partial p}\right)_T$
DHDT	$\left(\frac{\partial h}{\partial T}\right)_p$
DLAST	Density from last iteration to be used in next iteration in density search, given pressure and entropy or enthalpy
DPDRHO	$\left(\frac{\partial P}{\partial p}\right)_T$
DPDT	$\left(\frac{\partial P}{\partial T}\right)_p$
DPSDT	$\frac{dP_{sat}}{dT}$

DQDDDT	$\frac{\partial^2 Q}{\partial \rho \partial T}$
DQDRHO	$\left(\frac{\partial Q}{\partial \rho}\right)_T$
DQDTAU	$\left(\frac{\partial Q}{\partial \tau}\right)_\rho$
DSDRHO	$\left(\frac{\partial S}{\partial \rho}\right)_T$
DSDT	$\left(\frac{\partial S}{\partial T}\right)_\rho$
D2CDT2	$\frac{d^2 \psi_o}{dT^2}$
D2CTDT	$\frac{d^2 (\psi_o \tau)}{d\tau^2}$
D2QDD2	$\left(\frac{\partial^2 Q}{\partial \rho^2}\right)_T$
D2QDTA	$\left(\frac{\partial^2 Q}{\partial \tau^2}\right)$
ENTH	Final enthalpy (BTU/lbm)
ENTHL	Final enthalpy, value returned from WATER (BTU/lbm)
ENTR	Final entropy (BTU/lbm-°R)
ENTR1	Final entropy, value returned from WATER (BTU/lbm°R)
F(I)	Constants required for saturated pressure calculation used in PSATUR
FGIVEN	First known passed to WATER, i.e. Temperature for values of CYCLE of TP, TX and pressure for values of CYCLE of PX, PS, and PH
GAS	Universal gas constant=0.46141 joule/g°K
GAS1	Universal gas constant=66.9366 psia-cc/g°K
HCAL	Enthalpy calculated in PROPER
I	Index
ITERAT	Maximum number of iterations allowed in any one search routine
J	Index

JACOB

Jacobian

59

LASTD2

Density from last iteration, in density search (DFIND)
given temperature and pressure

LH

Enthalpy of saturated liquid, value returned from WATER
(BTU/lbm)

LS

Entropy of saturated liquid, value returned from WATER
(BTU/lbm[°]R)

LSATH

Enthalpy of saturated liquid (BTU/lbm)

LSATS

Entropy of saturated liquid (BTU/lbm-[°]R)

LSATU

Internal energy of saturated liquid (BTU/lbm-[°]R)

LSATV

Specific volume of saturated liquid (ft³/lbm)

LU

Internal energy of saturated liquid, value returned from
WATER (BTU/lbm)

LV

Specific Volume of saturated liquid, value returned from
WATER (ft³/lbm)

NERROR

Error flag, set if data is incorrect

NPRINA

NPRINA=1 print out results
NPRINA $\neq$ 1 no print out

NPRINT

NPRINT=1 print out results
NPRINT $\neq$ 1 no print out

NSTATE

State number to which properties are desired

NSTAT1

State number to which properties are desired

PCAL

Pressure calculated in PROPER(psia)

PCRIT

Critical pressure = 3203.6(psia)

PKNOWN

Known pressure(psia)

PRESS

Pressure (psia)

PRESS1

Pressure, value returned from WATER (psia)

PSAT

Saturated pressure calculated in PSATUR (psia)

Q

Q- part of Ψ

QUAL

Quality

QUAL1

Quality, value returned from WATER

SCAL	Entropy calculated in PROPER (BTU/lbm°R)
SGIVEN	Second known passed to WATER, i.e. quality for CYCLE=TX or PX; pressure for CYCLE=TP; enthalpy for CYCLE=PH; entropy for CYCLE=PS
SJOULE	Entropy (BTU/lbm°K)
SUM1	Intermediate result for saturated pressure calculation $\sum_{i=1}^8 F_i (0.65-0.01t)^{i-1}$
SUM2	Intermediate result for saturated pressure calculation $\sum_{i=1}^8 F_i (0.65-0.01t)^{i-2} (i-1)/100$
TAU	$1000/T^{\circ}K(\tau)$
TAUCON	$\tau_{aj} \begin{cases} \tau_c & \text{for } j=1 \\ 2.5 & \text{for } j>1 \end{cases}$
TAUCRT	Critical $\tau=\tau_c=1.544912$
TCRIT	Critical temperature (t_c)=347.136°C
TEMPC	Temperature (°C)
TEMPF	Temperature (°F)
TEMPF1	Temperature (°F), value returned from WATER
TEMPK	Temperature (°K)
TGUESS	Initial guess for saturation temperature search (TSATUR) given pressure (°K)
TLAST	Temperature after last iteration to be used for next iteration in(PHFIND or PSFIND) search given pressure and enthalpy or entropy(°)
TSAT	Saturation temperature (°C)
TTOLER	Upper tolerance on convergence and saturation condition tests
UINT	Internal energy (BTU/lbm)
UINT1	Final internal energy, value returned from WATER (BTU/lbm)
VH	Saturated vapor enthalpy, value returned from WATER (BTU/lbm)
VOL	Specific volume (ft ³ /lbm)

VOL1	Specific volume, value returned from WATER (ft ³ /lbm)	61
VS	Saturated vapor entropy (BTU/lbm°R), value returned from WATER	
VSATH	Saturated vapor enthalpy (BTU/lbm)	
VSATS	Saturated vapor entropy (BTU/lbm°R)	
VSATU	Saturated vapor internal energy (BTU/lbm)	
VSATV	Saturated vapor specific volume (ft ³ /lbm)	
VU	Saturated vapor internal energy, value returned from WATER (BTU/lbm)	
VV	Saturated vapor specific volume, value returned from WATER (ft ³ /lbm)	

All of the following variables are intermediate results

WSUM	$\sum_{i=1}^8 A_{ij} (i-1) (i-2) (\rho - \rho_{aj})^{i-3}$
W1SUM	$E^2 e^{-E\rho} \sum_{i=9}^{10} A_{ij} \rho^{i-9}$
W2SUM	$\sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} (WSUM + W1SUM - 2Ee^{-E\rho} A_{10,j})$
X7SUM	$\sum_{i=1}^6 (2-i)(1-i) C_i / \tau^i$
ZSUM	$\sum_{i=1}^8 A_{ij} (\rho - \rho_{aj})^{i-1}$
Z1SUM	$e^{-E\rho} \sum_{i=9}^{10} A_{ij} \rho^{i-9}$
Z2SUM	$\sum_{i=1}^8 A_{ij} (i-1) (\rho - \rho_{aj})^{i-2}$
Z3SUM	$\sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} \left[\sum_{i=1}^8 A_{ij} (\rho - \rho_{aj})^{i-1} + e^{-E\rho} \sum_{i=9}^{10} A_{ij} \rho^{i-9} \right]$
Z4SUM	$\sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} \left(\sum_{i=1}^8 A_{ij} (i-1) (\rho - \rho_{aj})^{i-2} - Ee^{-E\rho} \sum_{i=9}^{10} A_{ij} \rho^{i-9} + e^{-E\rho} A_{10,j} \right)$

$$Z5SUM \quad \sum_{j=1}^7 (\tau - \tau_{aj})^{j-2} + (\tau - \tau_c)(j-2)(\tau - \tau_{aj})^{j-3} \left(\sum_{i=1}^8 A_{ij} (\rho - \rho_{aj})^{i-1} + e^{-E\rho} \sum_{i=9}^{10} A_{ij} \rho^{i-9} \right)$$

$$Z6SUM \quad \sum_{i=1}^6 (2-i) C_i / \tau$$

$$Z7SUM \quad \sum_{i=1}^6 (1-i) C_i \tau^{2-i} / 1000$$

$$Z8SUM \quad \sum_{i=1}^6 C_i (i-1)(i-2) \tau^{3-i} / 10^6$$

$$Z9SUM \quad (\tau - \tau_c) \sum_{j=1}^7 (j-2)(\tau - \tau_{aj})^{j-3} (Z2SUM - 4.8ZSUM + e^{-4.8} A_{10,j})$$

APPENDIX IV

Sample run of WATER to illustrate

Its Use in Evaluating a Steam Turbine

The Program Listing and Output are Contained

```

C      TEST OF WATER
C      SAMPLE RUN OF SUBROUTINE WATER TO ILLUSTRATE ITS USE IN EVALUATING
C      A STEAM TURBINE (ISENTROPIC EXPANSION)
C
      IMPLICIT REAL(L)
      READ(5,9000) PINLET,TINLET,PEXIT
10    CALL WATER('TP',TINLET,PINLET,10,1,1,X,S,H,V,U,T,P,LS,LH,LV,LU,VS,
      1      VH,VV,VU,NEERROR)
      IF(NEERROR.EQ.1) GO TO 60
      SINLET=S
      HINLET=H
      VINLET=V
      UINLET=U
20    CALL WATER('PS',PEXIT,SINLET, 20,2,1,X,S,H,V,U,T,P,LS,LH,LV,LU,VS,
      1      VH,VV,VU,NEERROR)
      IF(NEERROR.EQ.1) GO TO 60
      HEXIT=H
      SEXIT=S
      VEXIT=V
      UEXIT=U
      TEXIT=T
50    DELTAH=HINLET-HEXIT
      WRITE(6,9100) TINLET,PINLET,TEXIT,PEXIT,DELTAH
      STOP
60    WRITE(6,9200)
9000  FORMAT(3F10.2)
9100  FORMAT(1H1,10X,'SAMPLE STEAM TURBINE CALCULATION USING SUBROUTINE
      1WATER',//,15X,'INLET TEMPERATURE = ',F15.5,/,15X,'INLET PRESSURE =
      2 ', F15.5,/,15X,'EXIT TEMPERATURE = ',F15.5,/,15X,'EXIT PRESSURE
      3= ',F15.5,/,15X,'TOTAL WORK OUTPUT(BTU/LBM) = ',F15.5)
9200  FORMAT(2H ,///,10X,'SAMPLE TURBINE RUN FAILED TO CONVERGE')
      END

```

***** THERMODYNAMIC PROPERTIES OF WATER *****

TEMPERATURE AND PRESSURE.

PROPERTIES OF WATER AT STATE 1

TEMPERATURE = 1000.00 DEG.F
PRESSURE = 1400.00000 PSIA
SUPERHEATED VAPOR

INTERNAL ENERGY (BTU/LBM)	1342.836
ENTHALPY (BTU/LBM)	1493.500
ENTROPY (BTU/LBM-R)	1.609442
SPECIFIC VOLUME (CU.FT/LBM)	.5817

***** THERMODYNAMIC PROPERTIES OF WATER *****

PRESSURE AND ENTROPY

PROPERTIES OF WATER AT STATE 2

TEMPERATURE = 75.12 DEG.F
 PRESSURE = .43000 PSIA
 SATURATED LIQUID-VAPOR EQUILIBRIUM
 QUALITY = .7759

	SATURATED VAPOR	SATURATED LIQUID
INTERNAL ENERGY (BTU/LBM)	1035.405	43.212
ENTHALPY (BTU/LBM)	1094.278	43.213
ENTROPY (BTU/LBM-R)	2.049976	.084247
SPECIFIC VOLUME (CU.FT/LBM)	740.7535	.0161

INTERNAL ENERGY (BTU/LBM)	813.048
ENTHALPY (BTU/LBM)	858.727
ENTROPY (BTU/LBM-R)	1.609442
SPECIFIC VOLUME (CU.FT/LBM)	574.7490

SAMPLE STEAM TURBINE CALCULATION USING SUBROUTINE WATER

INLET TEMPERATURE = 1000.00000

INLET PRESSURE = 1400.00000

EXIT TEMPERATURE = 75.12062

EXIT PRESSURE = .45000

TOTAL WORK OUTPUT (BTU/LBM) = 634.77307

REFERENCES

1. Keenan, Keyes, Moose and Hill, Steam Tables, John Wiley and Sons, New York, 1996.
2. Conte and deBoor, Elementary Numerical Analysis, An Alogrithmic Approach, 2nd Edition, McGraw-Hill, 1972.
3. G. N. Hapsopoulos and J. H. Keenan, Principles of General Thermodynamics, John Wiley and Sons, New York, 1965, pp. 256.
4. Mechtly, E. A., The International System of Units, Physical and Constants and Conversion Factors, 2nd Revision, NASA, Washington, D.C. 1973.