ASTEM97

Based on the
IAPWS IF97

Water and Steam Properties
for Industrial Use

Implementation by

Edward D. Throm (C) 2002 , 2005

User's Guide

Version 2.0

Table of Contents

List of Figures

List of Tables

Pseudo Code

Introduction

ASTEM97, **<u>A</u> <u>S</u>**team **<u>T</u>**able **<u>E</u>**valuation **<u>M</u>**odel based on the IAPWS IF**<u>97</u>** release for Water and Steam Properties for Industrial Use (IAPWSa), is a FORTRAN-based dynamic link library (DLL) for the calculation of the thermodynamic properties of water and steam, and for selected additional transport properties, as listed in Table 1.

Table 1 - ASTEM97 properties

ID	1-phase	2-phase	Parameter	Sym	SI Units	÷ Conversion =	English Unit
1	✓	✓	Pressure	p	Pa	6.8947570e+03	psia
2	✓	✓	Temperature	T	K	not applicable	°F
3	✓	✓	Specific Volume	v	m^3/kg	6.2427974e-02	ft^3/lbm
4	✓	✓	Specific Internal Energy	u	kJ/kg	2.3260000e+00	BTU/lbm
5	✓	✓	Specific Enthalpy	h	kJ/kg	2.3260000e+00	BTU/lbm
6	✓	✓	Specific Entropy	s	kJ/kg-K	4.1868000e+00	BTU/lbm-°F
7	✓	✓	Quality	X	---	1.0000000e+00	---
8	✓		Specific Heat at p=const	C_p	kJ/kg-k	4.1868000e+00	BTU/lbm-°F
9	✓		Specific Heat at v=const	C_v	kJ/kg-k	4.1868000e+00	BTU/lbm-°F
10	✓		Sonic Velocity	w	m/sec	3.0480000e-01	ft/sec
11	✓		dv/dp at T=const	$(\partial v/\partial p)\|_T$	m^3/kg-Pa	9.0544125e-06	ft^3/lbm-psi
12	✓		dv/dT at p=const	$(\partial v/\partial T)\|_p$	m^3/kg-k	1.1237035e-01	ft^3/lbm-°F
13	✓		dp/dv at T=const	$(\partial p/\partial v)\|_T$	Pa-kg/m^3	1.1044339e+05	psi-lbm/ft^3
14	✓		dp/dT at v=const	$(\partial p/\partial T)\|_v$	Pa/K	1.2410563e+04	psi/°F
15	✓		Coef of Thermal Expansion	β	1/K	1.8000000e+00	1/°F
16	✓		Isothermal Compressibility	α	1/Pa	1.4503774e-04	1/psi
17	✓		Isentropic Exponent	κ	---	1.0000000e+00	---
18	✓		Dynamic Viscosity	η	micro kg/m-sec	1.4881640e+06	lbm/ft-sec
19	✓		Surface Tension	σ	milli N/m	1.4593900e+04	lbf/ft
20	✓		Thermal Conductivity (Ind)	k_{IND}	W/m-K	1.7307350e+00	BTU/hr-ft-°F
21	✓		Thermal Conductivity (Gsi)	k_{GSI}	W/m-K	1.7307350e+00	BTU/hr-ft-°F
22	✓		Refractive Index (λ=1)	n	---	1.0000000e+00	---
23	✓		Static Dielectric Constant	ϵ	---	1.0000000e+00	---
24	✓	✓	Gibbs Free Energy	g_{FE}	kJ/kg	2.3260000e+00	BTU/lbm
25	✓	✓	Helmholtz Free Energy	f_{FE}	kJ/kg	2.3260000e+00	BTU/lbm
26	✓		Joule-Thomson Coef	μ_{JT}	K/Pa	8.0576524e-05	°F/psi
27	✓		Isothermal Joule-Thom Coef	μ_T	kJ/kg-Pa	3.3735779e-04	BTU/lbm-psi
28	✓		Kinematic Viscosity	υ	micro m^2/sec	9.2903040e-02	ft^2/sec
29	✓	✓	Compressibility Factor	Z	---	1.0000000e+00	---
30	✓		Prandtl Number	Pr	---	1.0000000e+00	---
31	✓	✓	Density	ρ	kg/m^3	1.6018460e+01	lbm/ft^3

IAPWS IF-97 (IAPWSa) covers the range:

273.15 K to 1,073.15 K from 611.213 Pa to 100 MPa
and
273.15 K to 2,273.15 K from 611.213 Pa to 10 MPa

NOTE: For pressures less than 611.213 Pa, the IF97 Region 2 or Region 5 low-level functions may be used to obtain the properties, for the appropriate temperature range.

Five IF97 regions are used in the formulation, as shown in Figure 1. **ASTEM97** adds additional regional identifiers to simplify programming and improve computational speed.

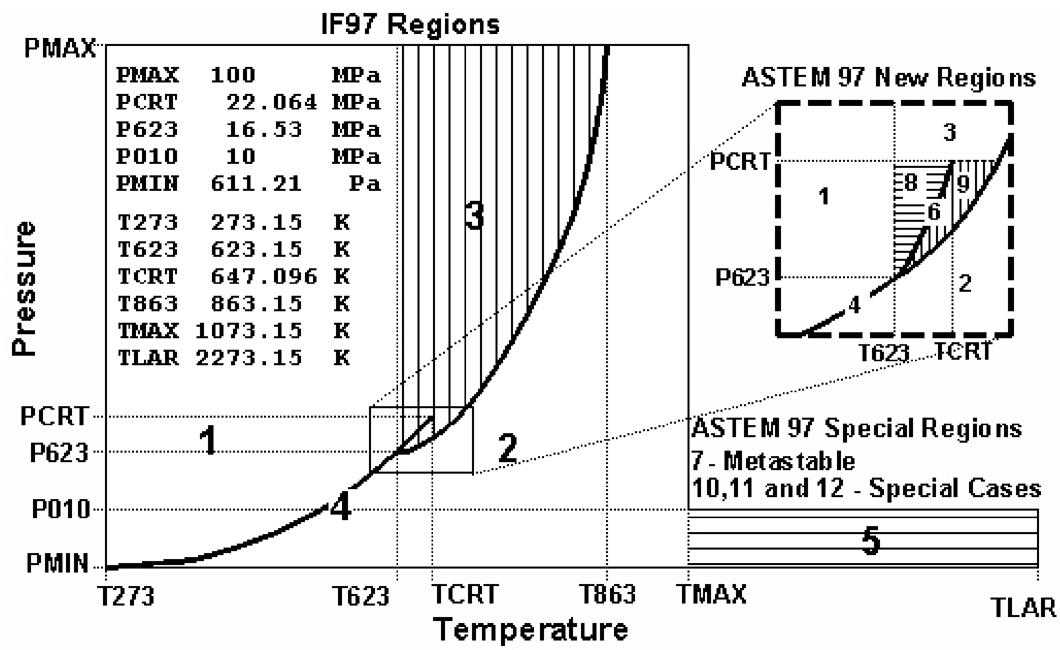

Figure 1 - IF97 and ASTEM97 Region Identifiers

Region 4 defines the saturation line. In **ASTEM97**, an additional region, Region 6, is used to describe the condition when the saturation line is in Region 3 (from 623.15 K to 647.096 K, 16.529 MPa to 22.064 MPa). A separate correlation is used to define the boundary between Region 3 and Region 2.

The thermodynamic properties and units used in routine calls are:

p	pressure (Pa)	**h**	specific enthalpy (kJ/kg)
T	temperature (K)	**s**	specific entropy (kJ/kg-K)
v	specific volume (m^3/kg)	**ρ**	density (kg/m^3)
u	specific internal energy (kJ/kg)	**X**	quality (---)

Subscript f denotes a saturated fluid and subscript g denotes a saturated vapor (or gas).
ASTEM97 Structure

ASTEM97 contains 307 FORTRAN routines to provide a wide range of interfaces to the thermodynamic and transport properties. High-level routines are available that return all the properties with a single call, mid-level routines for specific properties and low-level routines for specific properties in specific IF97 regions. Support routines are also available for unit conversion, to retrieve specific properties following calls to the high-level routines and to identify the IF97 region based on user provided data points. Some routines are used internally and not generally called by the user.

All the routines are based on SI input units: pressure in Pa, temperature in K, density in kg/m^3, specific volume in m^3/kg, enthalpy and internal energy in kJ/kg and entropy in kJ/kg-K. A summary description of the 307 routines, including error handling, is provided in Appendix A.

Some of the routines are redundant but are provided in the complete package to document the development of the package, and to provided insights for programmer's using **ASTEM97** to develop their own works. This also allows a user to develop a smaller (memory) package at the expense of computation speed.

Selection of Properties

For a simple substance, the phase rule requires that specification of any two properties, for example x and y, determines all other properties. Using z as the dependent property, the relationship between x, y and z is z(x,y). Other thermodynamic properties are derived from the total differential of z :

$$dz = \left(\frac{\partial z}{\partial x}\right)_y \cdot dx + \left(\frac{\partial z}{\partial y}\right)_x \cdot dy$$

The total differentials for z, x and y for variables *p* and *T* are:

$$dz = \left(\frac{\partial z}{\partial p}\right)_T \cdot dp + \left(\frac{\partial z}{\partial T}\right)_p \cdot dT$$

$$dx = \left(\frac{\partial x}{\partial p}\right)_T \cdot dp + \left(\frac{\partial x}{\partial T}\right)_p \cdot dT$$

$$dy = \left(\frac{\partial y}{\partial p}\right)_T \cdot dp + \left(\frac{\partial y}{\partial T}\right)_p \cdot dT$$

The expression for dy is equal to zero where the function z is held constant, therefore

$$dp = -\frac{\left(\frac{\partial y}{\partial T}\right)_p}{\left(\frac{\partial y}{\partial p}\right)_T} \cdot dT$$

and

$$\left(\frac{\partial z}{\partial x}\right)_y \equiv \frac{dz}{dx} = \frac{\left(\frac{\partial z}{\partial p}\right)_T \cdot dp + \left(\frac{\partial z}{\partial T}\right)_p \cdot dT}{\left(\frac{\partial x}{\partial p}\right)_T \cdot dp + \left(\frac{\partial x}{\partial T}\right)_p \cdot dT}$$

Substituting for dp yields the following result:

$$\left(\frac{\partial z}{\partial x}\right)_y = \frac{\left(\frac{\partial z}{\partial p}\right)_T \cdot \left(\frac{\partial y}{\partial T}\right)_p - \left(\frac{\partial z}{\partial T}\right)_p \cdot \left(\frac{\partial y}{\partial p}\right)_T}{\left(\frac{\partial x}{\partial p}\right)_T \cdot \left(\frac{\partial y}{\partial T}\right)_p - \left(\frac{\partial x}{\partial T}\right)_p \cdot \left(\frac{\partial y}{\partial p}\right)_T}$$

In a paper by H.-J. Kretzschmar, H.-J., et al, (1999b), the minimum set of thermodynamic properties required to obtain the thermodynamic derivatives for the Gibb's based pressure, temperature (p,T) equation or Helmholtz based temperature, specific volume (T,v) equation were presented based on the above relationship. Application of the Maxwell relations and Tables 3, 12 and 31 if IAPWSa with the above equations can be used to develop these results.

The results are provided in Tables 2 and 3. In addition to the basic thermodynamic propertied (p, T, s, v, c_p and c_v), four partial derivatives are needed:

$$\left(\frac{\partial v}{\partial p}\right)_T, \left(\frac{\partial v}{\partial T}\right)_p, \left(\frac{\partial p}{\partial T}\right)_v, \text{ and } \left(\frac{\partial p}{\partial v}\right)_T$$

ASTEM97 contains functions to compute these four partial derivatives affording the user the capability to obtain any of the partial derivatives, in Table 2 or 3, within the IF97 pressure, temperature domain.

Table 2 - Derivative relationships for pressure, temperature based equation (p,T)

$$\left(\frac{\partial z}{\partial x}\right)_y (p,T) \Rightarrow \left(\frac{\partial z}{\partial x}\right)_y = \frac{\left(\frac{\partial z}{\partial p}\right)_T \cdot \left(\frac{\partial y}{\partial T}\right)_p - \left(\frac{\partial z}{\partial T}\right)_p \cdot \left(\frac{\partial y}{\partial p}\right)_T}{\left(\frac{\partial x}{\partial p}\right)_T \cdot \left(\frac{\partial y}{\partial T}\right)_p - \left(\frac{\partial x}{\partial T}\right)_p \cdot \left(\frac{\partial y}{\partial p}\right)_T}$$

Derivatives:

x, y ,z	$\left(\frac{\partial z}{\partial T}\right)_p$	$\left(\frac{\partial z}{\partial p}\right)_T$
p	0	1
T	1	0
v	$\left(\frac{\partial v}{\partial T}\right)_p$	$\left(\frac{\partial v}{\partial p}\right)_T$
u	$c_p - p \cdot \left(\frac{\partial v}{\partial T}\right)_p$	$-T \cdot \left(\frac{\partial v}{\partial T}\right)_p - p \cdot \left(\frac{\partial v}{\partial p}\right)_T$
h	c_p	$v - T \cdot \left(\frac{\partial v}{\partial T}\right)_p$
s	$\frac{c_p}{T}$	$-\left(\frac{\partial v}{\partial T}\right)_p$
g_{FE}*	$-s$	v
f_{FE}*	$-p \cdot \left(\frac{\partial v}{\partial T}\right)_p - s$	$-p \cdot \left(\frac{\partial v}{\partial p}\right)_T$

* - g_{FE} - Gibb's free energy, f_{FE} - Helmholtz free energy

Minimum set of properties: $p,\ T,\ v,\ s,\ c_p,\ \left(\frac{\partial v}{\partial p}\right)_T$ and $\left(\frac{\partial v}{\partial T}\right)_p$

Table 3 - Derivative relationships for temperature, specific volume based equation (T,v)

$$\left(\frac{\partial z}{\partial x}\right)_y (T,v) \Rightarrow \quad \left(\frac{\partial z}{\partial x}\right)_y = \frac{\left(\frac{\partial z}{\partial v}\right)_T \cdot \left(\frac{\partial y}{\partial T}\right)_v - \left(\frac{\partial z}{\partial T}\right)_v \cdot \left(\frac{\partial y}{\partial v}\right)_T}{\left(\frac{\partial x}{\partial v}\right)_T \cdot \left(\frac{\partial y}{\partial T}\right)_v - \left(\frac{\partial x}{\partial T}\right)_v \cdot \left(\frac{\partial y}{\partial v}\right)_T}$$

Derivatives:

x, y, z	$\left(\frac{\partial z}{\partial T}\right)_v$	$\left(\frac{\partial z}{\partial v}\right)_T$
p	$\left(\frac{\partial p}{\partial T}\right)_v$	$\left(\frac{\partial p}{\partial v}\right)_T$
T	1	0
v	0	1
u	c_v	$T \cdot \left(\frac{\partial p}{\partial T}\right)_v - p$
h	$c_v + v \cdot \left(\frac{\partial p}{\partial T}\right)_v$	$T \cdot \left(\frac{\partial p}{\partial T}\right)_v + v \cdot \left(\frac{\partial p}{\partial v}\right)_T$
s	$\frac{c_v}{T}$	$\left(\frac{\partial p}{\partial T}\right)_v$
g_{FE}*	$v \cdot \left(\frac{\partial p}{\partial T}\right)_v - s$	$v \cdot \left(\frac{\partial p}{\partial v}\right)_T$
f_{FE}*	$-s$	$-p$

* - g_{FE} - Gibb's free energy, f_{FE} - Helmholtz free energy

Minimum set of properties: $v,\ T,\ p,\ s,\ c_v,\ \left(\frac{\partial p}{\partial T}\right)_v$ and $\left(\frac{\partial p}{\partial v}\right)_T$

High-Level Subroutines

High-level routines compute all the relevant properties based on the fluid phase, see Table 1. These routines do not require that the user identify the IF97 region, this is handled internally. Following calls to these high-level routines, calls to Function PROP97 are used to return each valid property. Function CSEU97 can then be used to convert the value from SI to English units, or from English to SI units, if necessary. The user should check the value of the internal flag IFLAG97(1) after a call to a high-level routine using function IERR97, for example ICHECK = IERR97(1). In general, a returned value that is either zero (0) or a positive (+) value — usually the IF97 region number, see Table 4, or one of the **ASTEM97** added region identifies — indicated a successful call. A negative (-) value indicates an error in the user supplied input parameters and the returned values are set to -1.0. See Appendix A for a description of the IFLAG97(1) returns values. Functions IERR97, PROP97 and CSEU97 are described under "User Interface Routines."

XPROP97(p,T)

Subroutine XPROP97 uses the low-level routines for its calculations. XPROP97 takes pressure (p) and temperature (T) as input. p or T may be input to specify quality (X) ($0 \leq X \leq 1$). The other input parameter must be in the saturation range. For X = 0 (saturated fluid properties) and for X = 1 (saturated vapor properties), XPROP97 computes all 31 properties (Table 1). Calls to Function PROP97 are used to retrieve each valid property. Function CSEU97 can then be used to convert the value from SI to English units if necessary. See code example 1a and 1b.

IXPROP97(p,T)

Function IXPROP97 can be used in place of subroutine XPROP97 when a function call is preferred.

PTPROP97(p,T), PXPROP97(p,X), and TXPROP97(T,X)

These subroutines use mid-level routines for their calculations. PTPROP97 takes pressure (p) and temperature (T) as input. PXPROP97 and TXPROP97 take quality (X) ($0 \leq X \leq 1$) as input. For X = 0 (saturated fluid properties) are for X = 1 (saturated vapor properties), PXPROP97 and TXPROP97 compute all 31 properties (Table 1). Calls to Function PROP97 are used to retrieve each valid property. Function CSEU97 can then be used to convert the value from SI to English units if necessary.

IPTCAL97(p,T), IPXCAL97(p,T), and ITXCAL97(T,X)

Functions IPTCAL97, IPXCAL97 and ITXCAL97 can be used in place of subroutines PTPROP97, PXPROP97 and TXPROP97, respectively, when a function call is preferred.

Pseudo code Example 1a - XPROP97 in SI Input

```
P = 1000    { PA }
T = 350     {  K }
CALL XPROP97(P,T)
SI_DENSITY      = PROPS97(31)
ENGLISH_DENSITY = CSEU97(31,SI_DENSITY)

{ RESULT
      SI_DENSITY      = 6.192E-03 KG/M^3
      ENGLISH_DENSITY = 3.865E-05 LBM/FT^3
}
```

Pseudo code Example 1b - XPROP97 in English Input

```
    P_E = 1000.                  { INPUT IN PSIA }
    P_S = CSEU97(-1,P_E)         { CONVERT TO PA }
    T_E = 300.                   { INPUT IN F    }
    T_S = CSEU97(-2,T_E)         { CONVERT TO K  }
{ MAKE CALL WITH SI VALUES }
    CALL XPROP97(P_S,T_S)
{ RETRIEVE SPECIFIC VOLUME }
    V_S = PROP97(3)
    V_E = CSEU97(3,V_S)          { CONVERT TO ENGLISH UNITS }
{ RETRIEVE THERMAL CONDUCTIVITY, GENERAL/SCIENTIFIC VALUE    }
    TC_S = PROP97(21)
    TC_E = CSEU97(21,TC_S)       { CONVERT TO ENGLISH UNITS }
{ DISPLAY RESULTS

                   P_S,T_S,V_S,TC_S     P_E,T_E,V_E,TC_E

  Pressure           0.689476E+07 Pa      0.100000E+04 psia
  Temperature        0.422039E+03 K       0.300000E+03 deg F
  Specific Volume    0.108503E-02 m^3/kg  0.173805E-01 ft^3/lbm
  Thermal Cond (GSI) 0.686534E+00 W/m-K   0.396672E+00 BTU/hr-ft-F
}
```

Mid-Level Regional Routines

Mid-level regional routines (and low-level property routines) require the user know the IF97 region before the call. Routines IREG97, IPRS97, ITEM97 and ITEM97A, are available to determine the IF97 region (See "Low-Level Regional Routines"). The user calls the appropriate routine based on the return value from one of these functions. The mid-level regional routines (and low-level property routines) do not, in general, internally check the input. See code example 2.

GIBB1(p,T)

Subroutine GIBB1 computes the dimensionless Gibbs free energy and its derivatives (Eq. 7, IAPWSa) for IF97 Region 1. Low-level routines PSAT97(T) or TSAT97(p) are available for use in obtaining the saturated fluid properties. See code example 4 (Data Point 4). Subsequent calls to P97CAL1 and P97CAL2 can be used to obtain the thermodynamic properties in SI units. A call to XTRAS97 can then be used to obtain the remaining thermodynamic and transport properties. Following calls to these mid-level routines, calls to Function PROP97 are used to return each valid property. Function CSEU97 can then be used to convert the value from SI to English units if necessary.

GIBB2(p,T) and GIBB2I(p,T)

Subroutine GIBB2 computes the dimensionless Gibbs free energy and its derivatives (Eqs. 15, 16 and 17, IAPWSa) for IF97 Region 2. Low-level routines PSAT97(T) or TSAT97(p) are available for use in obtaining the saturated vapor properties. See code example 3. Subroutine GIBB2I computes the dimensionless Gibbs free energy and its derivatives for the ideal-gas part only (Eq. 16, IAPWSa). Subsequent calls to P97CALA and P97CALB can be used to obtain the thermodynamic properties in SI units, or to P97CALG1 and P97CALG2 for the ideal-gas part only. A call to XTRAS97 can then be used to obtain the remaining thermodynamic and transport properties. Following calls to these mid-level routines, calls to Function PROP97 are used to return each valid property. Function CSEU97 can then be used to convert the value from SI to English units if necessary.

GIBBM(p,T) and GIBBMI(p,T)

Subroutine GIBBM computes the dimensionless Gibbs free energy and its derivatives (Eqs. 18, 16 (with modified coefficients) and 19, IAPWSa) for the supplemental, IF97 metastable-vapor Region 2. Subroutine GIBBMI computes the dimensionless Gibbs free energy and its derivatives for the ideal-gas part only (Eq. 16 (with modified coefficients), IAPWSa). Subsequent calls to P97CALA and P97CALB can be used to obtain the thermodynamic properties in SI units, or to P97CALG1 and P97CALG2 for the ideal-gas part only. A call to XTRAS97 can then be used to obtain the remaining thermodynamic and transport properties. Following calls to these mid-level routines, calls to Function PROP97 are used to return each valid property. Function CSEU97 can then be used to convert the value from SI to English units if necessary. See the section on the "Metastable Region" for additional information.

GIBB5(p,T) and GIBB5I(p,T)

Subroutine GIBB5 computes the dimensionless Gibbs free energy and its derivatives (Eqs. 32, 33 and 34, IAPWSa) for IF97 Region 5. Subroutine GIBB5I computes the dimensionless Gibbs free energy and its derivatives for the ideal-gas part only (Eq. 33, IAPWSa). Subsequent calls to P97CALA and P97CALB can be used to obtain the thermodynamic properties in SI units, or to P97CALG1 and P97CALG2 for the ideal-gas part only. A call to XTRAS97 can then be used to obtain the remaining thermodynamic and transport properties. Following calls to these mid-level routines, calls to Function PROP97 are used to return each valid property. Function CSEU97 can then be used to convert the value from SI to English units if necessary.

IGIBB97(IREG,p,T)

Function IGIBB97 can be used in place of the GIBB1, GIIB2, GIBB2I, GIBB5, GIBB5I, GIBBM or GIBBMI subroutines when a function call is preferred. The selection of IREG determines which IAPWS 97 region equation to call:

IREG	IF97 Region	Routine
1	Region 1	GIBB1
2	Region 2	GIBB2
-2	Region 2 Ideal Gas Only	GIBB2I
5	Region 5	GIBB5
-5	Region 5 Ideal Gas Only	GIBB5M
7	Region 2 Metastable	GIBBM
-7	Region 2 Metastable Ideal Gas	GIBBMI

HELM3(T,ρ)

Subroutine HELM3 computes the dimensionless Helmholtz free energy and its derivatives (Eq. 28, IAPWSa) for IF97 Region 3. Note that the Region 3 formulation is based on temperature and density, unlike the temperature and pressure formulation for the other regions. Subsequent calls to H97CALA and H97CALB can be used to obtain the thermodynamic properties in SI units. A call to XTRAS97 can then be used to obtain the remaining thermodynamic and transport properties. Following calls to these mid-level routines, calls to Function PROP97 are used to return each valid property. Function CSEU97 can then be used to convert the value from SI to English units if necessary.

There are additional routines available to support this formulation. Given pressure and temperature, Subroutine ROOT3, or function RHO3PT97, returns the density for use in the call to HELM3, or to any Region 3 mid- or low-level temperature, density function. For saturation properties, Subroutine ROOT3MAX, or function RHO3LG97, returns the saturation fluid and saturation vapor densities given the temperature. These densities are used in calls to HELM3, or to any Region 3 mid- or low-level temperature, density function, to obtain saturation properties. ***NOTE:*** Do not use TSAT97 to obtain the saturation temperature for this purpose when working in Region 3. See code example 3.

IHELM97(IREG,T,ρ)

Function IHELM97 can be in place of the subroutine HELM3 when a function call is preferred. IREG must be set to 3.

P97CALA, P97CAL1, P97CALG1 and H97CALA

These subroutines are called after calls to the appropriate Gibbs free energy or Helmholtz free energy routines, respectively. The dimensionless free energy and derivatives, based on pressure and temperature or temperature and density, are converted to properties and stored in an internal array: pressure (p), temperature (T), specific volume (v), specific internal energy (u), specific enthalpy (h), specific entropy (s), quality (X) and density (ρ). The Gibbs free energy (g_{FE})and the Helmholtz free energy (f_{FE}) are computed from the appropriate equation and also stored in an internal array. The internal array is accessed by Function PROP97.

Gibbs free energy = h - Ts   Helmholtz free energy = u - Ts

IGCALA97(IREG) and IHCALA97(IREG)

Function IGCALA97 can be used in place of the P97CALA, P97CAL1 or P97CALG1 subroutines, with IREG as defined for IGIBB97, when a function call is preferred. IHCALA97 can be used in place of subroutine H97CALA, IREG must be 3, when a function call is preferred.

P97CALB, P97CAL2, P97CALG2 and H97CALB

These subroutines are called after calls to the appropriate Gibbs free energy or Helmholtz free energy routines, respectively, and after the call to the appropriate xxxCALA or xxxCAL1 routine. The dimensionless free energy and derivatives, based on pressure and temperature or temperature and density, are converted to properties and stored in an internal array: specific isobaric heat capacity (C_p), specific isochoric heat capacity (C_v), speed of sound (or sonic velocity) (w), and the partial derivatives $(\partial p/\partial T)|_v$, $(\partial p/\partial v)|_T$, $(\partial v/\partial p)|_T$, $(\partial v/\partial T)|_p$. The coefficient of thermal expansion (β), isentropic exponent (κ), isothermal compressibility (α), isothermal Joule-Thomson coefficient (μ_T) and the Joule-Thomson coefficient (μ_{JT}) are also calculated. The internal array is accessed by Function PROP97. ***NOTE***: These properties are not defined for 2-phase conditions, quality in the range $0.0 < X < 1.0$, but are valid at X = 0.0 or X = 1.0.

IGCALB97(IREG) and IHCALB97(IREG)

Function IGCALB97 can be used in place of the P97CALB, P97CAL2 or P97CALG2 subroutines, with IREG as defined for IGIBB97, when a function call is preferred. IHCALB97 can be used in place of subroutine H97CALB, IREG must be 3, when a function call is preferred.

XTRAS97(ITYPE)

This subroutine is called after calls to the appropriate Gibbs free energy or Helmholtz free energy routines and after the calls to P97CALA/B, P97CAL1/2, P97CALG1/2 or H97CALA/B, respectively. The following additional thermodynamic and transport properties are computed and stored in an internal array: dynamic viscosity (η), surface tension (σ), thermal conductivity for Industrial use (k_{IND}), thermal conductivity for General and Scientific use (k_{GSI}), refractive index (n) which as coded for XTRAS97 is based on a wavelength λ=1.0 μm, for this purpose — to obtain the actual refractive index, the user must call the proper function with a specified wavelength — , static dielectric constant (ϵ), kinematic viscosity (υ), compressibility factor (Z) and Prandtl number (Pr). The internal array is accessed by Function PROP97. ***NOTE***: These properties are not defined for 2-phase conditions, quality in the range 0.0 < X < 1.0, but are valid at X = 0 or 1.

When ITYPE is set to 1 (one) pressure and temperature correlations are used to evaluate the transport properties — values are for subcooled or superheated conditions. When ITYPE is set to 2 (two), temperature and density correlations are used to evaluate the transport properties — values are for saturated fluid or saturated vapor conditions. The user must have previously called the appropriate Gibbs or Helmholtz routine with an equivalent quality of 0 (saturated fluid) or 1 (saturated vapor), to properly set the density (specific volume).

IXTRA97(ITYPE)

Function IXTRA97 can be in place of the subroutine XTRAS97 when a function call is preferred. ITYPE is defined under XTRAS97.

ROOT3(p,T,ρ)

Subroutine ROOT3 computes the density (ρ) given the pressure and temperature in IF97 Region 3. This allows the user to maintain consistency with a (p,T) model when in IF97 Region 3. The temperature and density are then used in HELM3 to obtain properties, or in any other temperature and density based routine. ***NOTE***: ROOT3 should not be used on the saturation line. ROOT3MAX is provided for calculations on the saturation line. [**Ver 2**] If the (p,T) input is outside IF97 Region 3, and return value is set to -1.0 and the error code (IERR97(1)) is set to -1.

RHO3PT97(p,T)

Function RHO3PT97(p,T) returns the density (ρ) given the pressure and temperature in IF97 Region 3, as an interface to ROOT3. It is recommended that this function be used in place of subroutine ROOT3(p,T,ρ). If the (p,T) input is outside IF97 Region 3, and return value is set to -1.0 and the error code (IERR97(1)) is set to -1.

ROOT3MAX(T,ρ_f,ρ_g)

Subroutine ROOT3MAX returns the saturation fluid density (ρ_f) and the saturation vapor density (ρ_g) at the input temperature. The temperature and saturation density are then used in HELM3 to obtain saturation properties, or in any other temperature and density based routine to obtain saturation properties. Appendix B provides additional details on ROOT3MAX options and obtaining saturated properties in IF97 Region 3. **[Ver 2**] This mid-level regional routine does an internal check on the input. The convergence criteria have been updated to fix an, albeit unlikely, internal math error of a divide by zero. In addition if T is less than 623.15 K the error flag is set to -3 and ρ_f and ρ_g are set to -1.0, or if T is greater than 647.096 K th error flag is set to -4 and ρ_f and ρ_g are set to -1.0.

RHO3LG97(T,IREG)

Function RHO3LG97(T,IREG) returns the saturation liquid density (ρ_f) at T if IREG = 0, and the saturation vapor density (ρ_g) at T if IREG = 1, as an interface to ROOT3MAX. It is recommended that this function be used in place of the subroutine ROOT3MAX(T,ρ_f,ρ_g).

Pseudo code Example 2 - GIBB1, GIBB2, ROOT3, HELM3

```
{ DATA POINT 1 - SUBCOOLED }
  P =   1.0E6 { PA }
  T = 403.15  {  K , 130 C }
  IREGION = IREG97(P,T)
{ RESULT POINT 1 IREG = 1 }
{ THEREFORE USE GIBB1     }
  CALL GIBB1(P,T)
  CALL P97CALA
  CALL P97CALB
  CALL XTRAS97(1)
  SI_ENTROPY  = PROP97(6)
  SI_ENTHALPY = PROP97(5)
  SI_KINVISV  = PROP97(28)
{ RESULTS
  SI_ENTROPY  =   1.63391
  SI_ENTHALPY = 546.882
  SI_KINVISC  =   0.2292   }
{ DATA POINT 2 - SUPERHEATED }
  P =  10.0E6 { PA }
  T = 673.15  {  K , 400 C }
  IREGION = IREG97(P,T)
{ RESULT POINT 2 IREG = 2 }
{ THEREFORE USE GIBB2     }
  CALL GIBB2(P,T)
  CALL P97CAL1
  CALL P97CAL2
  CALL XTRAS97(1)
  SI_ENTROPY  = PROP97(6)
  SI_ENTHALPY = PROP97(5)
  SI_KINVISC  = PROP97(28)

{ RESULTS
  SI_ENTROPY  =    6.21393
  SI_ENTHALPY = 3097.375
  SI_KINVISC  =    0.6474 }
```

```
{ DATA POINT 3 - REGION 3 }
  P =  24.0E6 { PA }
  T = 633.15  {  K , 360 C }
  IREGION = IREG97(P,T)
{ RESULT POINT 3 IREG = 3  }
{ THEREFORE USE HELM3 }
{ GET RHO AT (P,T) }
  RHO = RHO3PT97(P,T) [VER 2]
  RHO = 614.639
  CALL HELM3(T,RHO)
  CALL H97CALA
  CALL H97CALB
  CALL XTRAS97(1)
  SI_ENTROPY  = PROP97(6)
  SI_ENTHALPY = PROP97(5)
  SI_KINVISC  = PROP97(28)
{ RESULTS
  SI_ENTROPY  =    3.749795
  SI_ENTHALPY = 1675.5674
  SI_KINVISC  =    0.11636    }
{ DATAPOINT 4 - SATURATED FLUID }
  P = 10.0E6        { PA }
  T = TSAT97(P)     { GET T-SAT }
{ FOR FLUID WITH P ≤  PCRT,
  IF97 REGION 1, THEREFORE USE GIBB 1
  FUNCTION                           }
  CALL GIBB1(P,T)
  CALL P97CALA
  CALL P97CALB
{ SINCE A SATURATION REQUEST IS BEING
  MADE, CALL XTRAS97 WITH ITYPE =2  }
  CALL XTRAS97(2)
{ FOR A SATURATED VAPOR, USE GIBB2,
  P97CAL1,P97CAL2 AT P,T             }
{ WHEN P > PCRT, GET T-SAT THEN GET
  RHOF = RHO3LG97(T-SAT,0) [VER 2]
  RHOG = RHO3LG97(T-SAT,1) [VER 2]
  HELM3, H97CALA, H97CALB AND XTRAS97
  WITH ITYPE = 2                     }
```

Pseudo code Example 3 - TSAT97, ROOT3MAX

```
{SATURATED FLUID }

{ DATA POINT 1 }

  P =   15.0E6      { PA }
  TSAT = TSAT97(P)  {  K }
{ RESULT
  TSAT = 615.308  }
}
  ISAT = 1
  IF(P.GT.16.5291E6) ISAT = 2
{ RESULT POINT 1
  ISAT = 1
}
{ USE GIBB1 FOR FLUID }
  CALL GIBB1(P,T)
  CALL P97CALA
  CALL P97CALB
  CALL XTRAS97(1)
  SI_ENTHAPLY_F = PROP97(5)
{ RESULT
  SI_ENTHALPY_F = 1610.152
}
   ....
{ USE GIBB2 FOR VAPOR }
  CALL GIBB2(P,T)
  CALL P97CAL1
  CALL P97CAL2
  CALL XTRAS97(1)
  SI_ENTHAPLY_G = PROP97(5)
{ RESULT
  SI_ENTHAPLY_G = 2610.865
}
   ....
```

```
{ SATURATED FLUID }

{ DATA POINT 2 }

  P =  19.0E6       { PA }
  TSAT = TSAT97(P)  {  K }
{ RESULT
  TSAT = 634.621
}
  ISAT = 1
  IF(P.GT.16.5291E6) ISAT = 2
{ RESULT POINT 2
  ISAT = 2
}
{ GET RHOF AND RHOG }
  RHOF = RHO3LG97(TSAT,0)   [VER 2]
  RHOG = RHO3LG97(TSAT,1)   [VER 2]
{ RESULT
  RHOF = 519.34975,
  RHOG = 149.84931
}
{ USE RHOF FOR FLUID }
  CALL HELM3(T,RHOF)
  CALL H97CALA
  CALL H97CALB
  CALL XTRAS97(1)
  SI_CP_F = PROP97(8)
 { RESULT
   SI_CP_F = 16.2434
 }
   ....
{ USE RHOG FOR VAPOR }
  CALL HELM3(T,RHOG)
  CALL H97CALA
  CALL H97CALB
  CALL XTRAS97(1)
  SI_CP_G = PROP97(8)
{ RESULT
  SI_CP_G = 30.60394
}
   ....
```

Mid-Level Property-Specific Routines

The mid-level routines determine the appropriate IF97 region. The low-level and density based routines should only be called after determining the appropriate region (from low-level routines IREG97, IPRS97, ITEM97, or ITEM97A). See code example 4. The user should check the value of the internal flag IFLAG97(1) after a call to a mid-level property-specific routine using function IERR97, for example ICHECK = IERR97(1). In general, a returned value that is either zero (0) or positive (+) value — usually the IF97 region number, see Table 4, or one of the **ASTEM97** added region identifies — indicated a successful call. A negative (-) value indicates an error in the user supplied input parameters and the returned value is set to -1.0. See Appendix A for a description of the IFLAG97(1) returns values.

Subcooled or superheated properties are calculated with ***var***PT97 (pressure, temperature) routines. Saturated or 2-phase properties are calculated with pressure-based ***var***PX97(pressure, quality) routines, or temperature-based ***var***TX97(temperature, quality) routines. **ASTEM97** handles the IF97 region determination. ***NOTE***: For some properties, a 2-phase state point is not defined and the quality must be 0 for saturated fluid or 1 for saturated vapor.

In IF97 Region 3, there are ***var***TR3 (temperature, density) routines available for obtaining subcooled or superheated properties. For saturated or 2-phase properties it is recommended that mid-level routines ***var***TX97 or ***var***PX97 be used. To use ***var***TR3, the correct density must first be obtained from ROOT3 or RHO3PT97 for subcooled or supercritical conditions; or from ROOT3MAX or RHO3LG97 for saturated conditions.

Specific Volume (v) Functions

Mid-Level	Low-Level (Called by Mid-Level)	Low-Level Ideal-Gas (User called)	Density Based (User called)
VPT97(p,T)	VPT1(p,T)		
	VPT2(p,T)	VPT2I(p,T)	
	VPT3(p,T)		VTR3(T,ρ)
	VPT5(p,T)	VPT5I(p,T)	
	VPTM(p,T)	VPTMI(p,T)	
VPX97(p,X)	Note: 0.0 ≤ X ≤ 1.0		
VTX97(T,X)			

Specific Internal Energy (u) Functions

Mid-Level	Low-Level (Called by Mid-Level)	Low-Level Ideal-Gas (User called)	Density Based (User called)
UPT97(p,T)	UPT1(p,T)		
	UPT2(p,T)	UPT2I(p,T)	
	UPT3(p,T)		UTR3(T,ρ)
	UPT5(p,T)	UPT5I(p,T)	
	UPTM(p,T)	UPTMI(p,T)	
UPX97(p,X)	Note: 0.0 ≤ X ≤ 1.0		
UTX97(T,X)			

Specific Enthalpy (h) Functions

Mid-Level	Low-Level (Called by Mid-Level)	Low-Level Ideal-Gas (User called)	Density Based (User called)
HPT97(p,T)	HPT1(p,T)		
	HPT2(p,T)	HPT2I(p,T)	
	HPT3(p,T)		HTR3(T,ρ)
	HPT5(p,T)	HPT5I(p,T)	
	HPTM(p,T)	HPTMI(p,T)	
HPX97(p,X)	Note: 0.0 ≤ X ≤ 1.0		
HTX97(T,X)			

Specific Entropy (s) Functions

Mid-Level	Low-Level (Called by Mid-Level)	Low-Level Ideal-Gas (User called)	Density Based (User called)
SPT97(p,T)	SPT1(p,T)		
	SPT2(p,T)	SPT2I(p,T)	
	SPT3(p,T)		STR3(T,ρ)
	SPT5(p,T)	SPT5I(p,T)	
	SPTM(p,T)	SPTMI(p,T)	
SPX97(p,X)	Note: 0.0 ≤ X ≤ 1.0		
STX97(T,X)			

Speed of Sound (Sonic Velocity) Functions

Mid-Level	Low-Level (Called by Mid-Level)	Low-Level Ideal-Gas (User called)	Density Based (User called)
SVPT97(p,T)	SVPT1(p,T)		
	SVPT2(p,T)	SVPT2I(p,T)	
	SVPT3(p,T)		SVTR3(T,ρ)
	SVPT5(p,T)	SVPT5I(p,T)	
	SVPTM(p,T)	SVPTMI(p,T)	
SVPX97(p,X)	Note: X=0 or X=1		
SVTX97(T,X)			

Specific Isobaric Heat Capacity (C_p) Functions - (Specific heat at constant pressure)

Mid-Level	Low-Level (Called by Mid-Level)	Low-Level Ideal-Gas (User called)	Density Based (User called)
CPPT97(p,T)	CPPT1(p,T)		
	CPPT2(p,T)	CPPT2I(p,T)	
	CPPT3(p,T)		CPTR3(T,ρ)
	CPPT5(p,T)	CPPT5I(p,T)	
	CPPTM(p,T)	CPPTMI(p,T)	
CPPX97(p,X)	NOTE: X=0 or X=1		
CPTX97(T,X)			

NOTE: The equation for C_p exhibits some numerical inconsistencies near the critical point, yielding a negative value at (p_{crt} , T_{crt}). To address this issue, a user flag (number 6), set using ISET97 [for example, IFIXCP = ISET97(6,1)] is available to adjust the input pressure in steps of $1.0x10^{-7}$ Pa until a positive value is obtained. See Appendix C for additional details.

Specific Isochoric Heat Capacity (C_v) Functions - (Specific heat at constant volume)

Mid-Level	Low-Level (Called by Mid-Level)	Low-Level Ideal-Gas (User called)	Density Based (User called)
CVPT97(p,T)	CVPT1(p,T)		
	CVPT2(p,T)	CVPT2I(p,T)	
	CVPT3(p,T)		CVTR3(T,ρ)
	CVPT5(p,T)	CVPT5I(p,T)	
	CVPTM(p,T)	CVPTMI(p,T)	
CVPX97(p,X)	NOTE: X=0 or X=1		
CVTX97(T,X)			

$(\partial p/\partial T)|_v$ Functions - Partial derivative of p with respect to T at constant v

Mid-Level	Low-Level (Called by Mid-Level)	Low-Level Ideal-Gas (User called)	Density Based (User called)
DPDTV97(p,T)	DPDTV1(p,T)		
	DPDTV2(p,T)	DPDTV2I(p,T)	
	DPDTV3(p,T)		DPDTV3R(T,ρ)
	DPDTV5(p,T)	DPDTV5I(p,T)	
	DPDTVM(p,T)	DPDTVMI(p,T)	
DPDTVPX(p,X)	NOTE: X=0 or X=1		
DPDTVTX(T,X)			

$(\partial p/\partial v)|_T$ Functions - Partial derivative of p with respect to v at constant T

Mid-Level	Low-Level (Called by Mid-Level)	Low-Level Ideal-Gas (User called)	Density Based (User called)
DPDVT97(p,T)	DPDVT1(p,T)		
	DPDVT2(p,T)	DPDVT2I(p,T)	
	DPDVT3(p,T)		DPDVT3R(T,ρ)
	DPDVT5(p,T)	DPDVT5I(p,T)	
	DPDVTM(p,T)	DPDVTMI(p,T)	
DPDVTPX(p,X)	NOTE: X=0 or X=1		
DPDVTTX(T,X)			

$(\partial v/\partial p)|_T$ Functions - Partial derivative of v with respect to p at constant T

Mid-Level	Low-Level (Called by Mid-Level)	Low-Level Ideal-Gas (User called)	Density Based (User called)
DVDPT97(p,T)	DVDPT1(p,T)		
	DVDPT2(p,T)	DVDPT2I(p,T)	
	DVDPT3(p,T)		DVDPT3R(T,ρ)
	DVDPT5(p,T)	DVDPT5I(p,T)	
	DVDPTM(p,T)	DVDPTMI(p,T)	
DVDPTPX(p,X)	NOTE: X=0 or X=1		
DVDPTTX(T,X)			

$(\partial v/\partial T)|_p$ Functions - Partial derivative of v with respect to T at constant p

Mid-Level	Low-Level (Called by Mid-Level)	Low-Level Ideal-Gas (User called)	Density Based (User called)
DVDTP97(p,T)	DVDTP1(p,T)		
	DVDTP2(p,T)	DVDTP2I(p,T)	
	DVDTP3(p,T)		DVDTP3R(T,ρ)
	DVDTP5(p,T)	DVDTP5I(p,T)	
	DVDTPM(p,T)	DVDTPMI(p,T)	
DVDTPPX(p,X)	NOTE: X=0 or X=1		
DVDTPTX(T,X)			

Pseudo Code Example 4 - Mid- and Low-Level Routines

Mid-Level

```
CASE 1
    P =  15.E6    {PA}
    T = 450.      {K }


    H = HPT97(P,T)
    {RESULT H = 756.78 }

CASE 2
    P =  25.E6    {PA}
    T = 650.      {K }


    H = HPT97(P,T)
    {RESULT H = 1876.36 }
```

Low-Level

```
    P =  15.E6    {PA}
    T = 450.     {K }
    IREG = IREG97(P,T)
    { RESULT IREG = 1 }
    H = HPT1(P,T)
    {RESULT H = 756.78 }

    P =  25.E6    {PA}
    T = 650.     {K }
    IREG = IREG97(P,T)
    { RESULT IREG = 3 }
    H = HPT3(P,T)
    {RESULT H = 1876.36 }
```

Low-Level routines are most useful when performing iterative calculations in a fixed IF97 region by reducing overhead associated with finding the region in a Mid-Level routine (CPU TEST - your results may vary)

Mid-Level:

```
    P  =  15.E6         {PA}
    T1 = 300.           {K }
    T2 = 500.           {K }
    DT = (T2-T1)/100000.
    T0 = T1 - DT



    DO 10 I = 1 ,100001
    T0 = T0 + DT
    H  = HPT97(P,T0)
 10 CONTINUE

    ELAPSED CPU TIME
      2.2 SEC
```

Low-Level:

```
    P  =  15.E6    {PA}
    T1 = 300.      {K }
    T2 = 500.      {K }
    DT = (T2-T1)/100000.
    T0 = T1 - DT
    IR1 = IREG97(P,T1)
    IR2 = IREG97(P,T2)
    { IR1 = IR2 = 1  }
    DO 10 I = 1 ,100001
    T0 = T0 + DT
    H = HPT1(P,T0)
 10 CONTINUE

    ELAPSED CPU TIME
    0.35 SEC
    FACTOR OF ~6 FASTER
```

Pressure Related Functions

These functions return pressure (or quality) based on the user provided temperature and second variable (v, u, h or s). A temperature and density function, PTR3, is available for determining the IF97 Region 3 pressure given temperature and density. ***NOTE***: PTR3 should not be used to determine saturation pressure, use PSAT97. For saturated or 2-phase conditions, the **ASTEM97** default return value is quality (X) unless the user selects to have the saturation pressure returned by setting an internal flag (ISET97(11,1)). Alternatively, the saturation pressure can be obtained from a call to PSAT97 at T.

Functions PHS97, PVH97 and PVS97 are described under "Backward Equations."

<table>
<tr><td>Mid-Level</td><td>Low-Level
(Called by Mid-Level)</td><td>Density Based
(User called)</td></tr>
<tr><td>PTH97(T,h)</td><td rowspan="4">PVAR97(T,VAR,IVAR,IREG)
VAR   v   u   h   s
IVAR   1   2   3   4</td><td>None</td></tr>
<tr><td>PTS97(T,s)</td><td rowspan="3"></td></tr>
<tr><td>PTU97(T,u)</td></tr>
<tr><td>PTV97(T,v)</td></tr>
<tr><td>PHS97(h,s)</td><td colspan="2"></td></tr>
<tr><td>PVH97(v,h)</td><td colspan="2" rowspan="2">PVHS97(VIN,VAR,IVAR)
VAR   h   s
IVAR   1   2</td></tr>
<tr><td>PVS97(v,s)</td></tr>
</table>

The specification of temperature with enthalpy, internal energy or entropy does not always result in a unique fluid state point. The fluid could be 2-phase or subcooled, and even represent more than one subcooled pressure state when in IF97 Region 1. Following a call to one of these functions, the user should check the return flag value (IRTN = IERR97(1)). A negative value indicates an invalid set of input parameters. See Appendix A. To accommodate multiply pressure returns based on temperature, user function PTMANS97 provides the interface to these values. A user flag, IFLAG97(11), is available to return the 2-phase result as the saturation pressure instead of the default return value as quality (using function ISET97). See Appendix D for additional details.

The significance of the IRTN return value is shown below:

IRTN	Return Value	PTMANS97(1)	PTSMANS97(2)
1,2,3,5 or 8, 9	Pressure	-1 (no valid result)	-1 (no valid result)
4 or 6	Quality (FLAG 11=0) Pressure (FLAG 11=1)	-1 (no valid result)	-1 (no valid result)
10	Pressure	Pressure	-1 (no valid result)
11	Pressure	-1 (no valid result)	Quality (FLAG 11=0) Pressure (FLAG 11=1)
12	Pressure	Pressure	Quality (FLAG 11=0) Pressure (FLAG 11=1)

Note: When multiple pressure values are found (IRTN = 11 or 12), the function returns the lower value and PTMAN97 is used to retrieve the other values.

Figure 2 is a plot of the entropy as a function of pressure at a constant temperature, 1 °C. An example call to the pressure related function PTS97 for an entropy value of 0.0155 kJ/kg-K is provide in code example 5.

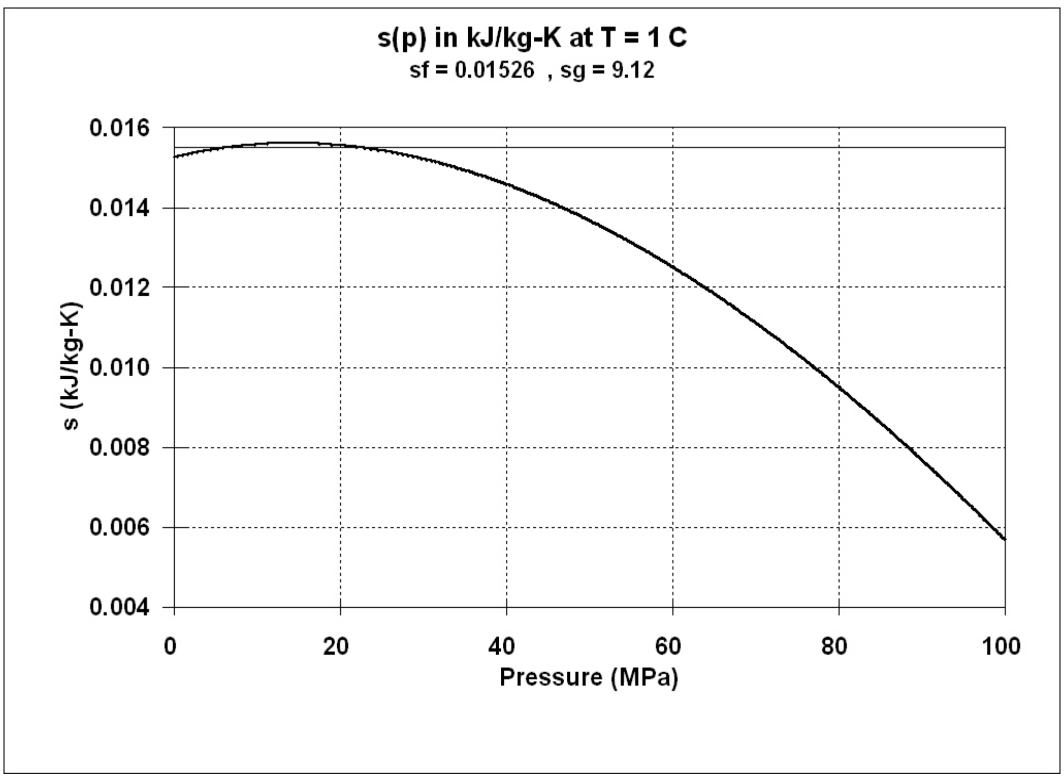

Figure 2 - s(p) in kJ/kg-K at 1°C

Pseudo Code Example 5 - PTS97 T=1°C, S=0.0155 kJ/kg-K

```
       TVALUE = T273+1.0           { CONVERT TO K }
       SVALUE = 0.0155
       P1  = PTS97(TVALUE,SVALUE)  { RETURN FIRST PRESSURE RESULT }
       INO = IERR97(1)             { RESULT, INO = 12 }
       P2  = PTMANS97(1)           { GET 2-ND PRESSURE RESULT }
      { P3 RETURNED AS QUALITY, ASTEM97 DEFAULT }
       P3  = PTMANS97(2)           { GET 3-RD PRESSURE RESULT - QUALITY }
      { COMPUTE PROPERTIES BASED ON PRESSURES AND QUALITY }
      { FOR RETURN VALUE, P1 IN KNOWN TO BE IN REGION 1 FROM INO VALUE }
       V1 = VPT1(P1,TVALUE)
       U1 = UPT1(P1,TVALUE)
       S1 = SPT1(P1,TVALUE)
       H1 = HPT1(P1,TVALUE)
      { USE MID-LEVEL JUST TO BE SAFE, COULD ALSO USE VPT1,UPT1, ETC. }
       V2 = VPT97(P2,TVALUE)
       U2 = UPT97(P2,TVALUE)
       S2 = SPT97(P2,TVALUE)
       H2 = HPT97(P2,TVALUE)
      { FOR THIRD QUALITY VALUE, P3 USE T,X FUNCTIONS }
       V3 = VTX97(TVALUE,P3)
       U3 = UTX97(TVALUE,P3)
       S3 = STX97(TVALUE,P3)
       H3 = HTX97(TVALUE,P3)
       PSAT = PSAT97(TVALUE)      { GET SATURATION PRESSURE AT TVALUE }
      { DISPLAY RESULTS
        NO.      P         T        V          U        H         S
                PA         K      M^3/KG     KJ/KG    KJ/KG    KJ/KG-K
         1    6163933.5  274.1  0.0009971   4.2513   10.3971   0.015500
         2   22690008.8  274.1  0.0009891   4.3656   26.8082   0.015500
         3{X} 2.633E-05 274.1  0.0060674   4.2385    4.2424   0.015500
         3{PSAT}  657.1                                                }
```

Temperature Related Functions

These functions return temperature (or quality) based on the user provided pressure and second variable (v, u, h or s). For saturated or 2-phase conditions, the return value is quality (X) unless the user selects to have the temperature returned by setting an internal flag (ISET97(11,1)). The saturation temperature can be obtained from a call to TSAT97 at p.

Functions THS97, TVH97 and TVS97 are described under "Backward Equations."

Mid-Level	Low-Level (Called by Mid-Level)	Density Based (User called)
TPH97(p,h)	TVAR97(T,VAR,IVAR,IREG)	none
TPS97(p,s)	VAR v u h s	
TPU97(p,u)	IVAR 1 2 3 4	
TPV97(p,v)		
THS97(h,s)		
TVH97(v,h)	None but uses PVHS97(VIN,VAR,IVAR) VAR h s	
TVS97(v,s)	IVAR 1 2	

Following a call to one of these functions, the user should check the return flag value (IRTN = IERR97(1)). A negative value indicates an invalid set of input parameters. See Appendix A.

Two-phase Functions

For these 2-phase functions, the quality, X, must be in the range 0.0 to 1.0. The temperature must be in the range 273.15 K (T273) to 647.096 K (TCRT) and the pressure in the range 611.21 Pa (PMIN) to 22.064 MPa (PCRT).

Following a call to one of these functions, the user should check the return flag value (IRTN = IERR97(1)). A negative value indicates an invalid set of input parameters. See Appendix A.

<table>
<tr><th>Variable returned</th><th>Temperature based</th><th>Pressure based</th><th>Low-level routine called to return saturated fluid and saturated vapor values</th></tr>
<tr><td>Enthalpy - h</td><td>HTX97(T,X)</td><td>HPX97(p,X)</td><td rowspan="4">TWOFAZ(T,IVAR,V_F,V_G)
variable IVAR, at T
IVR 1 2 3 4
v u h s
Note: 0.0 ≤ X ≤ 1.0</td></tr>
<tr><td>Entropy - s</td><td>STX97(T,X)</td><td>SPX97(p,X)</td></tr>
<tr><td>Internal energy - u</td><td>UTX97(T,X)</td><td>UPX97(p,X)</td></tr>
<tr><td>Specific volume - v</td><td>VTX97(T,X)</td><td>VPX97(p,X)</td></tr>
<tr><td>Speed of sound - w</td><td>SVTX97(T,X)</td><td>SVPX97(p,X)</td><td rowspan="7">Note: X = 0 or 1</td></tr>
<tr><td>Isobaric heat capacity - C_p</td><td>CPTX97(T,X)</td><td>CPPX97(p,X)</td></tr>
<tr><td>Isochoric heat capacity - C_v</td><td>CVTX97(T,X)</td><td>CVPX97(p,X)</td></tr>
<tr><td>$(\partial p/\partial T)|_v$</td><td>DPDTVPX(T,X)</td><td>DPDTVPX(p,X)</td></tr>
<tr><td>$(\partial p/\partial v)|_T$</td><td>DPDVTTX(T,X)</td><td>DPDVTPX(p,X)</td></tr>
<tr><td>$(\partial v/\partial p)|_T$</td><td>DVDPTTX(T,X)</td><td>DVDPTPX(p,X)</td></tr>
<tr><td>$(\partial v/\partial T)|_p$</td><td>DVDTPTX(T,X)</td><td>DVDTPPX(p,X)</td></tr>
</table>

Low-Level Regional Routines

Proper use of the low-level routines (see Table 5) requires the proper identification of the IF97 region. Proper use of the IAPWS backward equations developed for IF97 Regions 1,2, 3 and 4' (region 4' is saturated with s > 5.21 kJ/kg-K) also requires proper identification of the IF97 region (See "Backward Equations"). A negative (-) value indicates an error in the user supplied input parameters and the returned values are set to -1.0. See Appendix A for a description of the IFLAG97(1) returns values.

IREG97(p,T)

IREG97 locates the IF97 region based on the input pressure and temperature. IREG97 returns the IF97 region value 1 through 5 as shown in Figure 1, or the **ASTEM97** additional region value. **ASTEM97** adds three more regions to improve computational speed. Region 6 is the saturation line from 623.15 K to 647.096 K. Region 8 is a sub-region of IF97 Region 3, between 623.15 K and 647.096 K to the left of the saturation line. Region 9 is a sub-region of IF97 Region 3, between 623.15 K and 647.096 K to the right of the saturation line.

Table 4 - Return values from IREG97, IPRS97, ITEM97, ITEM97A and IBAK97

IRETURN	Single Phase	2-phase Saturated Fluid	2-phase Saturated Vapor	Ideal Gas
1	Gibb 1 functions	Gibb 1 functions		
2	Gibb 2 functions		Gibb 2 functions	Gibb 2I functions
3,8 or 9	Helm 3 functions			
4[1]		Gibb 1 functions	Gibb 2 functions	
6[2]		Helm 3 (T,ρ_f) functions	Helm 3 (T,ρg) functions	
5	Gibb 5 functions			Gibb 5I functions
Special Cases				
10[3]	ITEM97,ITEM97A			
11[3]	ITEM97			
12[3]	ITEM97			

1 - Based on given p or T obtain second parameter (T_{sat} or p_{sat}) from TSAT97 or PSAT97 and call appropriate region 1 or region 2 functions.

2 - If p_{sat} is known, get T_{sat} from TSAT97. Call ROOT3MAX(T_{sat} , ρ_f , ρ_g), then call appropriate region 3 temperature, density (T,ρ) functions with ρ_f or ρ_g.

3 - Special cases when in region 1 multiple pressure state points exist for given combinations of T with u, h and s, or in combination with a subcooled state point there is also a 2-phase state point. See "Pressure Related Functions."

IPRS97(p,VAR,IVAR)

IPRS97 is similar to IREG97 but allows the user to specify the pressure and, instead of temperature, either

specific volume (v)	with IVAR = 1
internal energy (u)	with IVAR = 2
enthalpy (h)	with IVAR = 3
entropy (s)	with IVAR = 4

[**Ver 2**] IPRS97 has been updated to improve the identification of the saturation line (to return quality of X=0 or X=1) from the **ASTEM97** pressure related functions PTV97, PTU97, PTH97 and PTS97. If the quality falls within $\pm 1.0 \times 10^{-12}$ of 0 or 1, the input point is identified as being on the saturation line.

ITEM97(T,VAR,IVAR)

ITEM97 is similar to IPRS97 but allows the user to specify the temperature and, instead of pressure, either

specific volume (v)	with IVAR = 1
internal energy (u)	with IVAR = 2
enthalpy (h)	with IVAR = 3
entropy (s)	with IVAR = 4

ITEM97 may return 3 special values for special cases in IF97 Region 1, see "Pressure Related Functions."

10 - two pressures in region 1 at T and VAR
11 - one pressure in region 1 at T and VAR, and a 2-phase state point
12 - two pressures in region 1 and a 2-phase state point

ITEM97A(T,VAR,IVAR)

ITEM97A is a special sub-set to ITEM97 and only checks for state points in IF97 Region 1 and does not look for a possible 2-phase state point. Valid returns are 1 or 10.

IBAK97(p,VAR,IVAR)

IBAK97 is coded to specifically identify the IF97 region for use with the IAPWS backward equations for T(p,h) and T(p,s) in IF97 regions 1 and 2 (IAPWSa) — see TPHBK1, TPHBK2 and TPH97B and see TPSBK1, TPSBK2 and TPS97B. It is also used with IAPWS backward equations for p(h,s) — see PHSBK1, PHSBK2 and PHS97B and see THSBK1, THSBK2 and THS97B — after the call to verify that the call was proper as described in IAPWSg.

IBAK97 determines the IF97 region based on pressure and a second properties, VAR, identified by IVAR: IVAR = 1 for VAR = h and 2 for s. IBAK97 returns 1 or 2 for IF97 Regions 1 or 2.

Low-Level Thermodynamic Properties

PSAT97(T) and TSAT97(p)

Functions PSAT97 and TSAT97 compute the saturation line, IF97 Region 4 — and **ASTEM97** Region 6 — (Eqs. 29, 29a and 29b, IAPWSa). PSAT97 is the saturation pressure as a function of temperature (Eq. 30, IAPWSa) and TSAT97 is the saturation temperature as a function of pressure (Eq. 31, IAPWSa). See Appendix B for additional information. The user should check the value of the internal flag IFLAG97(1) after a call to a one of these routines using function IERR97, for example ICHECK = IERR97(1). A negative (-) value indicates an error in the user supplied input parameters and the returned values are set to -1.0. See Appendix A for a description of the IFLAG97(1) returns values.

P2397(T) and T2397(p)

Functions P2397 and T2397 compute the (p,T) line between IF97 Region 2 and 3. P2397 is the pressure as a function of temperature (Eq. 5, IAPWSa) and T2397 is the temperature as a function of pressure (Eq. 6, IAPWSa). The user should check the value of the internal flag IFLAG97(1) after a call to a one of these routines using function IERR97, for example ICHECK = IERR97(1). A negative (-) value indicates an error in the user supplied input parameters and the returned values are set to -1.0. See Appendix A for a description of the IFLAG97(1) returns values.

Basic Thermodynamic Properties and Derivatives

Low-level routines are available for the basic parameters for each IF97 region and for the ideal gas portion in Regions 2 and 5 and in the supplemental region:

specific volume	v
specific internal energy	u
specific enthalpy	h
specific entropy	s
speed of sound (sonic velocity)	w
specific heat at constant pressure	C_p
specific heat at constant volume	C_v
dv/dp at T=const	$(\partial v/\partial p)\|_T$
dv/dT at p=const	$(\partial v/\partial T)\|_p$
dp/dv at T=const	$(\partial p/\partial v)\|_T$
dp/dT at v=const	$(\partial p/\partial T)\|_v$

In IF97 Region 3, both (p,T) and (T,ρ) formulations are available.

A summary of the 99 low-level routine for the calculation of the thermodynamic properties is provided in Table 5.

Table 5 - Summary of low-level thermodynamic property functions

	IF97 Region								
	Gibb 1 (p,T)	Gibb 2 (p,T)	Gibb 2 Ideal gas (p,T)	Gibb 2 Sup (p,T)	Gibb 2 Sup Ideal gas (p,T)	Helm 3 (p,T)	Helm 3 (T,ρ)	Gibb 5 (p,T)	Gibb 5 Ideal gas (p,T)
v	VPT1	VPT2	VPT2I	VPTM	VPTMI	VPT3	VTR3	VPT5	VPT5I
u	UPT1	UPT2	UPT2I	UPTM	UPTMI	UPT3	UTR3	UPT5	UPT5I
h	HPT1	HPT2	HPT2I	HPTM	HPTMI	HPT3	HTR3	HPT5	HPT5I
s	SPT1	SPT2	SPT2I	SPTM	SPTMI	SPT3	STR3	SPT5	SPT5I
w	SVPT1	SVPT2	SVPT2I	SVPTM	SVPTMI	SVPT3	SVTR3	SVPT5	SVPT5I
C_p	CPPT1	CPPT2	CPPT2I	CPPTM	CPPTMI	CPPT3	CPTR3	CPPT5	CPPT5I
C_v	CVPT1	CVPT3	CVPT2I	CVPTM	CVPTMI	CVPT3	CVTR3	CVPT5	CVPT5I
$\left.\frac{\partial p}{\partial T}\right\|_v$	DPDTV1	DPDTV2	DPDTV2I	DPDTVM	DPDTVMI	DPDTV3	DPDTV3R	DPDTV5	DPDTV5I
$\left.\frac{\partial p}{\partial v}\right\|_T$	DPDVT1	DPDVT2	DPDVT2I	DPDVTM	DPDVTMI	DPDVT3	DPDVT3R	DPDVT5	DPDVT5I
$\left.\frac{\partial v}{\partial p}\right\|_T$	DVDPT1	DVDPT2	DVDPT2I	DVDPTM	DVDPTMI	DVDPT3	DVDPT3R	DVDPT5	DVDPT5I
$\left.\frac{\partial v}{\partial T}\right\|_p$	DVDTP1	DVDTP2	DVDTP2I	DVDTPM	DVDTPMI	DVDTP3	DVDTP3R	DVDTP5	DVDTP5I

NOTE : w - speed of sound (sonic velocity)

Metastable Region

In the development of IF97, a Gibbs energy equation was developed for the metastable region (see GIBBM). This equation is thought to provide the best representation available for the metastable region. The implementation of the metastable region in **ASTEM97** is described here. See code example 6. ***NOTE***: In this implementation, the minimum metastable temperature is not checked against any physical limitations, for example at p = 611.657 Pa, T_M is calculated to be 245.844 K.

TPMETA(p)

TPMETA returns the temperature alone the saturation region for a point representing 5% moisture (or a quality of 95%) based on the 2-phase enthalpy. For p less than or equal to 16.53 MPa, the fluid enthalpy (h_f) is calculated with HPT1 and the vapor enthalpy (h_g) with HPT2. For p greater than 16.53 MPa, the enthalpies are calculated from HTR3 using ρ_f and ρ_g from a call to ROOT3MAX.

If the calling pressure is less than 611.21 Pa the return value is set to -1.0, with the error flag set to -1. The error flag is retrieved using function IERR97 with a call index of 1, for example, IERR = IERR97(1). If the pressure is greater than 22.064 MPa the return value is set to -1.0, with the error flag set to -2. If the pressure at the metastable temperature is greater than 22.064 MPa, the return value is set to -1.0 with the error flag set to -8. The internally computed temperature can be retrieved using function PTMANS97 with the call index set to 3, for example, TCALC = PTMANS97(3). The user can then evaluate the information and determine a course of action.

THMETA(p,h)

THMETA returns the temperature using the same methodology as TPMETA but for the specific user supplied enthalpy.

If the calling pressure is less than 611.21 Pa the return value is set to -1.0, with the error flag set to -1. The error flag is retrieved using function IERR97 with a call index of 1, for example, IERR = IERR97(1). If the pressure is greater than 22.064 MPa the return value is set to -1.0, with the error flag set to -2. If the user value for h is less than the 5% moisture limit the return value is set to -1.0, with the error flag set to -3. If h is greater than the saturation vapor enthalpy the return value is set to -1.0, with the error flag set to -4. If the pressure at the metastable temperature is greater than 22.064 MPa, the return value is set to -1.0 with the error flag set to -8. The computed temperature can be retrieved using function PTMANS97 with the call index set to 3, for example, TCALC = PTMANS97(3). The user can then evaluate the information and determine a course of action.

PTMETA97(p,T) and PHMETA97(p,h)

Once the metastable temperature (T_M) has been determined, from TPMETA or THMETA, a call to PTMETA97 will calculate the thermodynamic and transport properties of the metastable state point using the low-level ***var***PTM or ***var***TR3, as appropriate, routines. If the user supplied T is less than T_M or greater than T_{sat} an error is issued. For T_M less than or equal to 623.15 K, the supplemental metastable region is used to calculate the properties. For T_M greater than 623.15 K, the properties are calculated from Region 3.

PHMETA97 uses THMETA to calculate T_M. If T_M is valid, PTMETA97 is then called to calculate the properties. If the calling pressure is less than 611.21 Pa the return value is set to -1.0, with the error flag set to -1. The error flag is retrieved using function IERR97 with a call index of 1, for example, IERR = IERR97(1). If the pressure is greater than 22.064 MPa the return value is set to -1.0, with the error flag set to -2. In PHMETA97, if the user value for h is less than the 5% moisture limit the return value is set to -1.0, with the error flag set to -3 and if h is greater than the saturation vapor enthalpy the return value is set to -1.0, with the error flag set to -4.

In PTMETA97, if the user value for T is less than T_M the return value is set to -1.0, with the error flag set to -3 and if T is greater than the T_{sat} the return value is set to -1.0, with the error flag set to -4.

If the pressure at the metastable temperature is greater than 22.064 MPa, the return value is set to -1.0 with the error flag set to -8. The computed temperature can be retrieved using function PTMANS97 with the call index set to 3, for example, TCALC = PTMANS97(3). The user can then evaluate the information and determine a course of action.

The high-level GIBBM (and GIBBMI) function is also available for use once T_M has been determined.

IPTMET97(p,T) and IPHMET97(p,h)

Functions IPTMET97(p,T) and IPHMET97(p,h) can be used in place subroutines PTMETA97(p,T) and PHMETA97(p,h), respectively, when a function call is preferred.

```
Pseudo Code Example 6 - Metastable Properties
```

```
     P     = 5.E6     { PA }
     TMETA = TPMETA(P)     { RESULT TMETA = 520.838687560342 K }
     TSAT  = TSAT97(P)     { RESULT TSAT  = 537.092871186330 K }
     { SELECT A TEMPERATURE BETWEEN TMETA AND TSAT }
     TM = 525.         { K }
     METASTABLE CALCULATIONS            PRESSURE,TEMPERATURE CALCULATIONS
     CALL PTMETA97(P,TM)                { AT P,TM }
     H_META = PROP97(5)                   H_PT = HPT97(P,TM)
     S_META = PROP97(6)                   S_PT = SPT97(P,TM)
     V_META = PROP97(3)                   V_PT = VPT97(P,TM)
{ RESULTS:  H  2735.44212045429                H  1094.65600115682
            S     5.86294906237556             S     2.80805343785363
            V     3.692907137044172E-02        V     1.254400376029277E-03  }
```

Backward Equations

Selected backward equations were developed as part of the IAPWS IF97 release. The developed equations are computational fast and within a prescribed “accuracy” defined in IAPWSa. Backward equations for T(p,h) and T(p,s) in IF97 Regions 1 and 2 were developed since these combinations appear frequently in problem solutions.

Backward equations for p(h,s) and T(h,s) in IF97 Regions 1 and 2 have also been developed (IAPWSg), and a backward equation for v(p,T) in IF97 Region 3 has also been proposed (1999a).

[**Ver 2**] Functions for IF97 region 3 and IF97 Region 4' (the saturation region with s > 5.21 kJ/kg-K) have also been developed (IAPWSh and IAPWSi) for T(p,h), T(p,s), p(h,s) and T(h,s), as well as additional functions for v(p,h), v(p,s) and v(h,s) in IF97 Region 3, and for T_{sat}(h,s) and x(h,s) in IF97 Region 4' (IAPWSi).

IAPWS - Backward Region Equations in (h,s)

Use of the IF97 backward equations is based on the identification of the IF97 backward region. To facilitate the identification of the backward region, the IAPWS has developed functions to determine the IF97 backward region based on (h,s). For the equations based on (h,s), functions for the saturation line and the region 1-3 boundary were developed. A special function was developed by the IAPWS to assist in locating the boundary between IF97 Region 2 and Region 3.

To complete the identification of the backward region based on (h,s), three additional functions were developed for **ASTEM97**. (1) A function to obtain h given s for the 100 MPa pressure limit line and the 1073.15 K temperature limit line, and (2) for the 611.213 Pa pressure limit line. (3) A special function to obtain s given h was also developed for the 273.15 K limit line because a simple function in s was not practical. These functions were not developed by the IAPWS.

IAPWS - HSSAT97B(s), HS1397B(s) and THS2397B(h,s)

HSSAT97B(s) is used to obtain the saturation h (both liquid and vapor) line and includes the functions given as equations 3, 4, 5 and 6 of IAPWSi. HS1397B(s) is used to obtain h on the region boundary between IF97 Region 1 and Region 3, the function given as equation 7 of IAPWSi. THS2397B(h,s) is used to obtain the temperature on the boundary between IF97 Region 2 and Region 3, the function given as equation 8 in IAPWSi. Error codes are listed in Appendix A.

ASTEM97 - HSMAX97(s), HSPMIN97(s) and SHTMIN97(h)

HSMAX97(s) and HSPMIN97(s) are piecewise fits to obtain h given s. HSMAX97(s) is used to obtain h at s on the 100 MPa and the 1073.15 K limit lines. HSPMIN97(s) is used to obtain h at s on the 611.213 Pa limit line. SHTMIN97(h) is used to obtain s at h on the 273.15 K limit line. The functions were tested for 100,000,000 points with the following error and uncertainties:

Function		Min △diff	Max △diff	RMS Error
SHTMAX97(H)	△s	-9.986D-10	5.414D-10	3.607D-10
HSMAX97(S) at 100 MPa	△h	-4.552D-06	2.464D-06	4.446D-07
HSMAX97(S) at 1073.15 K	△h	-9.105D-06	1.474D-06	4.243D-06
HSPMIN97(S)	△h	-6.130D-06	2.507D-06	6.848D-07

The minimum or maximum differences are accounted for in determining if the (h,s) point is in the IF97 backward region. These functions are only intended for use in determining the IF97 backward region, and for quick graphics, but not for obtaining property values. Error codes are listed in Appendix A. The HSMAX97 treatment of the enthalpy discontinuity near 623.15 K (Region 1 to Region 3 interface) and near 863.15 K (Region 3 to Region 2 interface) is discussed in Appendix E.

IRGHS97B(h,s)

IRGHS97B(h,s) is used to determine the backward region at h and s, based on the six function described above, returning the IF97 backward region number 1, 2, 3 or 4 (if in IF97 backward Region 4' - s > 5.21 kJ/kg-K). If the (h,s) point is outside the IF97 backward region an error, in the form of a negative return value, is generated. See Appendix A. In the case where (h,s) is in the saturation region but s is less than 5.21 kJ/kg-K, the error, returned, value is -5. It is recommended that IRGHS97 be used to determine the IF97 backward region when using any of the IAPWS low-level (h,s)-based backward equations.

TSHS97B(h,s)

TSHS97B(h,s) is used to obtain the saturation temperature in the saturation region, IF97 Region 4', with s > 5.21 kJ/kg-K, as defined by equation 9 in IAPWSi. IRGHS97 is used to validate the (h,s) input. While no specific function is provided for the saturation pressure in Region 4' as a function of (h,s), the saturation pressure can be obtained from PSAT97(T=TSHS97B(h,s)), as discussed in Section 5.4 in IAPWSi, as is done in function PHS97BK.

XHSBK4(h,s)

XHSBK4(h,s) is used to obtain the quality in the saturation region, IF97 Region 4', with s > 5.21 kJ/kg-K, using the method described in Section 5.5 in IAPWSi. IRGHS97 is used to validate the (h,s) input.

For the equations based on p and h or s, IAPWS functions for the saturation pressure given h and the saturation pressure given s were developed by the IAPWS to assist in identifying IF97 Region 3.

PSHBK3(h) and PSSBK3(s)

PSHBK3(h) is used to obtain the saturation pressure value at h, as defined by equation 10 in IAPWSh. PSSBK3(s) is used to obtain the saturation pressure value at s, as defined by equation 11 in IAPWSh. Error codes are listed in Appendix A.

IRPHS97(p,VAR,IVAR)

IRPHS97(p,VAR,IVAR) is similar to IPRS97(p,VAR,IVAR), with VAR= h, IVAR=1 or with VAR=s, IVAR=2, but uses the two saturation functions described above to determine the IF97 backward region. Error codes are listed in Appendix A. It is recommended that IRPHS97 be used to determine the IF97 backward region when using any of the IAPWS low-level pressure-based backward equations.

IAPWS - Backward Equations for T(p,h) in IF97 Regions 1, 2 and 3

These functions were developed to improve the calculation speed of T(p,h) within a prescribed level of “accuracy” (IAPWSa and IAPWSh). Alternatively, **ASTEM97** mid-level function TPH97 (without an ending “B” or “BK”) can be used without compromising “accuracy” but may take longer to compute the value. TPH97 also handles IF97 Region 5, and the saturation region (Region 4).

TPHBK1(p,h), TPHBK2(p,h), and TPHBK3(p,h)

In IF97 Region 1, low-level function TPHBK1(p,h) is used to compute the temperature given p and h (Eq. 11, IAPWSa). In IF97 Region 2, low-level function TPHBK2(p,h) is used to compute the temperature given p and h (Eqs. 22, 23 and 24, IAPWSa). IF97 Region 2 is broken down into three separate sub-regions with the function internally handling the determination of which sub-region contains the (p,h) point. TPHBK1 and TPHBK2 do not check the (p,h) input - see TPH97B. If the computed temperature is less than 273.15 K, a -1.0 value is returned and the error code is set to -10. If the computed temperature is greater than 1073.15 K, a -1.0 value is returned and the error code is set to -20. The computed value can be retrieved from PTMANS97(3). See Appendix A.

In IF97 region 3, low-level function TPHBK3(p,h) is used to compute the temperature given p and h (Eqs. 2 and 3, IAPWSh), broken down into two separate sub-regions with the function internally handling the determination of which sub-region contains the (p,h) point. TPHBK3 uses IRPHS97 to validate the (p,h) input.

TPH97B(p,h)

TPHBK1 and TPHBK2 do not check the validity of p and h in the call. The user should call IRPHS97(p,h,1) with **ASTEM97 Ver 2** (or IPRS97(p,h,3) Ver 1) and check the return flag (IERR97(1)) and call the correct function based on the flag value (1 or 2). An additional **ASTEM97** mid-level function, TPH97B(p,h) is available for IF97 Regions 1 and 2 only and will check the p and h input and call the appropriate backward equation. If the inputs are out of range, a -1.0 value is returned, see Appendix A. TPH97B is maintained for legacy use (using IPRS97 for region checking) , and does not use the **Ver 2** routine IRPHS97.

TPH97BK(p,h)

High-level function TPH97BK(p,h) covers IF97 Regions 1, 2 and 3 and uses function IRPHS97(p,VAR,IVAR) to determine the IF97 backward region and calls the appropriate function to return the temperature. If the inputs are out of range, a -1.0 value is returned, see Appendix A. TPH97BK(p,h) is the recommended function for use, and TPH97K(p,h) is provide for use as legacy coding until the user's model is updated to use TPH97BK(p,h).

[**Ver 2**] The IAPWS T(p,h) backward equations are data fits. To accommodate cases where the resulting temperature returned from these functions is outside the IF97 range (273.15 to 1073.15 K), but still within an acceptable level of "accuracy," **ASTEM97** flag IFLAG97(14) can be set to 1 (for example IRNG = ISET97(14,1)) to force these function to return the minimum or maximum value enabling the user problem to continue to run without violating the IF97 (p,T) range.

IAPWS - Backward Equations for T(p,s) in IF97 Regions 1, 2, and 3

These functions were developed to improve the calculation speed of T(p,s) within a prescribed level of "accuracy" (IAPWSa and IAPWSh). Alternatively, **ASTEM97** mid-level function TPS97 (without an ending "B" or "BK") can be used without compromising "accuracy" but may take longer to compute the value. TPS97 also handles IF97 Region 5, and the saturation region (Region 4).

TPSBK1(p,s), TPSBK2(p,s), and TPSBK3(p,s)

In IF97 Region 1, low-level function TPSBK1(p,s) computes the temperature given p and s (Eq. 13, IAPWSa). In IF97 Region 2, low-level function TPSBK2(p,s) computes the temperature given p and s (Eqs. 25, 26 and 27, IAPWSa). IF97 Region 2 is broken down into three separate sub-regions with the function internally handling the determination of which sub-region contains the (p,s) point. TPSBK1 and TPSBK2 do not check the (p,s) input - see TPS97B. If the computed temperature is less than 273.15 K, a -1.0 value is returned and the error code is set to -10. If the computed temperature is greater than 1073.15 K, a -1.0 value is returned and the error code is set to -20. The computed value can be retrieved from PTMANS97(3). See Appendix A.

In IF97 region 3, low-level function TPSBK3(p,s) is used to compute the temperature given p and s (Eqs. 6 and 7, IAPWSh), broken down into two separate sub-regions with the function internally handling the determination of which sub-region contains the (p,s) point. TPSBK3 uses IRPHS97 to validate the (p,h) input.

Two of the validation data points for TPSBK3 are actually outside IF97 backward Region 3, they are located in Region 1. To accommodate verification of the IF97 Region 3 function, flag IFLAG97(13) is set to 1 (for example, IEXT = ISET97(13,1). With IFLAG97(13) set to 1, TPSBK3 does not check the input with IRPHS97. Use of this flag is intended for validation only. (See code example 8a.)

TPS97B(p,s)

TPSBK1 and TPSBK2 do not check the validity of p and s in the call. The user should call IRPHS97(p,s,2) with **ASTEM97** Ver 2 (or IPRS97(p,s,4) Ver 1) and check the return flag (IERR97(1)) and call the correct function based on the flag value (1 or 2). An additional **ASTEM97** mid-level function, TPS97B(p,s) is available for IF97 Regions 1 and 2 only and will check the pressure and entropy input and call the appropriate backward equation. If the inputs are out of range, a -1.0 value is returned, see Appendix A. TPS97B is maintained for legacy use (using IPRS97 for region checking), and does not use the **Ver 2** routine IRPHS97.

TPS97BK(p,s)

High-level function TPS97BK(p,s) covers IF97 Regions 1, 2 and 3 and uses function IRPHS97(p,VAR,IVAR) to determine the IF97 backward region and calls the appropriate function to return the temperature. If the inputs are out of range, a -1.0 value is returned, see Appendix A. TPS97BK(p,s) is the recommended function for use, and TPS97K(p,s) is provide for use as legacy coding until the user's model is updated to use TPS97BK(p,s).

[**Ver 2**] The IAPWS T(p,s) backward equations are data fits. To accommodate cases where the resulting temperature returned from these functions is outside the IF97 range (273.15 to 1073.15 K), but still within an acceptable level of "accuracy," **ASTEM97** flag IFLAG97(14) can be set to 1 (for example IRNG = ISET97(14,1)) to force these function to return the minimum or maximum value enabling the user problem to continue to run without violating the IF97 (p,T) range.

IAPWS - Backward Equations for p(h,s) in IF97 Regions 1, 2, 3 and 4'

These functions were developed to improve the calculation speed of p(h,s) within a prescribed level of "accuracy" (IAPWSa, IAPWSh and IAPWSi). Alternatively, **ASTEM97** mid-level function PHS97 (without an ending "B" or "BK") can be used without compromising "accuracy" but may take longer to compute the value. PHS97 handles all IF97 regions, including Region 5 and the full saturation region.

PHSBK1(h,s), PHSBK2(h,s) and PHSBK3(h,s)

In IF97 Region 1, low-level function PHSBK1(h,s) computes the pressure given h and s (Eq. 1, IAPWSg). In IF97 Region 2, low-level function PHSBK2(h,s) computes the pressure given h and s (Eqs. 3, 4 and 5, IAPWSg). IF97 Region 2 is broken down into three separate sub-regions with the function internally handling the determination of which sub-region contains the (h,s) point. If the computed pressure is less than 611.213 Pa, a -1.0 value is returned and the error code is set to -10. If the computed pressure is greater than 100 MPa, a -1.0 value is returned and the error code is set to -20. The computed value can be retrieved from PTMANS97(1). See Appendix A.

In IF97 Region 3, low-level function PHSBK3(h,s) computer the pressure given h and s (Eqs. 1 and 2, IAPWSi). IF97 Region 3 is broken down into two separate sub-regions with the function internally handling the determination of which sub-region contains the (h,s) point. PHSBK3 does not check the (h,s) point, therefore IRGHS97 should be first be called to verify the input. If the computed pressure is greater than 100 MPa, a -1.0 value is returned and the error code is set to -20. The computed value can be retrieved from PTMANS97(1). See Appendix A.

PHS97B(h,s,IREG)

PHSBK1 and PHSBK2 do not check the validity of h and s in the call. An additional **ASTEM97** mid-level function, PHS97B(h,s,IREG), is available for IF97 Regions 1 and 2 only and will check the h and s input after the call to the appropriate backward equation based on the user provided IREG (1 or 2) to determine if the resulting pressure is in the appropriate IF97 region. If an error is detected, the user can retrieve the calculated pressure result and the corresponding IF97 region identifiers. See Appendix A description for PHS97B.

PHS97BK(h,s)

High-level function PHS97BK(h,s) covers the entire IF97 backward range, Regions 1, 2, 3 and 4' (s > 5.21 kJ/kg-K) and used IRGHS97(h,s) to determine the IF97 backward region. No specific pressure function for IF97 backward Region 4' is provided however PHS97BK will return the saturation pressure based on TSHS97B(h,s). If the input is out of range, an error is set and the return values is set to -1.0, see Appendix A.

PHS97BK(h,s) is the recommended function for use and PHS97K(h,s,IREG) is provide for use as legacy coding until the user's model is updated to use PHS97BK(h,s).

[**Ver 2**] The IAPWS p(h,s) backward equations are data fits. To accommodate cases where the resulting pressure returned from these functions is outside the IF97 range (611.213 Pa to 100 MPa), but still within an acceptable level of "accuracy," **ASTEM97** flag IFLAG97(14) can be set to 1 (for example IRNG = ISET97(14,1)) to force these function to return the minimum or maximum value enabling the user problem to continue to run without violating the IF97 (p,T) range.

IAPWS - Backward Equations for T(h,s) in IF97 Regions 1, 2, 3 and 4'

These functions were developed to improve the calculation speed of T(h,s) within a prescribed level of "accuracy" (IAPWSa, IAPWSh and IAPWSi). Alternatively, **ASTEM97** mid-level function PHS97 (without an ending "B" or "BK") can be used without compromising "accuracy" but may take longer to compute the value. PHS97 handles all IF97 regions, including Region 5 and the full saturation region.

THSBK1(h,s), THSBK2(h,s), and THSBK3(h,s)

In IF97 Region 1, low-level function THSBK1(h,s) computes the temperature given h and s (Section 7, IAPWSg). In IF97 Region 2, low-level function THSBK2(h,s) computes the temperature given h and s (Section 7, IAPWSg). IF97 Region 2 is broken down into three separate sub-regions with the function internally handling the determination of which sub-region contains the (h,s) point.

In IF97 Region 3, low-level function THSBK3(h,s) computes the temperature given h and s, as described in Section 3.4 of IAPWSi. THSBK3 does not check the (h,s) point, therefore IRGHS97 should be first be called to verify the input.

THS97B(h,s,IREG)

THSBK1 and THSBK2 do not check the validity of the h and s in the call. An additional mid-level function, THS97B(h,s,IREG) is available and will check the h and s input after the call to the appropriate backward equation based on the user provided IREG (1 or 2) to determine if the resulting temperature is in the appropriate IF97 region. If an error is detected, the user can retrieve the calculated pressure result and the corresponding IF97 region identifiers. See Appendix A description for THS97B.

THS97BK(h,s)

High-level function THS97BK(h,s) covers the entire IF97 backward range, Regions 1, 2, 3 and 4' (s > 5.21 kJ/kg-K) and used IRGHS97(h,s) to determine the IF97 backward region. In IF97 backward Region 4', the saturation temperature is obtained from TSHS97B(h,s). If the input is out of range, an error is set and the return values is set to -1.0, see Appendix A.

THS97BK(h,s) is the recommended function for use and THS97K(h,s,IREG) is provide for use as legacy coding until the user's model is updated to use THS97BK(h,s).

[**Ver 2**] The IAPWS T(h,s) backward equations are data fits. To accommodate cases where the resulting temperature returned from these functions is outside the IF97 range (273.15 to 1073.15 K), but still within an acceptable level of "accuracy," **ASTEM97** flag IFLAG97(14) can be set to 1 (for example IRNG = ISET97(14,1)) to force these function to return the minimum or maximum value enabling the user problem to continue to run without violating the IF97 (p,T) range.

IAPWS - Backward Equations for v(p,h), v(p,s), v(h,s) and v(p,T) in IF97 Region 3

These functions were developed to improve the calculation speed of v(p,h) and v(p,s) within a prescribed level of "accuracy" (IAPWSh). **ASTEM97** does not have similar functions, however, the specific volume can be computed as v(p,h) = VPT97(p, TPH97(p,h)), or v(p,s) = VPT97(p , TPS97(p,s)).

VPHBK3(p,h) and VPSBK3(p,s)

In IF97 Region 3, function VPHBK3(p,h) is used to compute the specific volume given pressure and enthalpy (Eqs. 4 and 5, IAPWSh). In IF97 Region 3, function VPSBK3(p,s) is used to compute the specific volume given pressure and entropy (Eqs. 8 and 9, IAPWSh). IF97

Region 3 is broken down into two separate sub-regions with the functions internally handling the determination of which sub-region contains the (p,h) or (p,s) point. If the point is not in IF97 Region 3, the return value is set to -1.0. (See Appendix A for additional information on errors.) VPHBK3 and VPSBK3 use IRPHS97 to validate the (p,h or s) input.

Two of the validation data points for VPSBK3 are actually outside IF97 backward Region 3, they are located in Region 1. To accommodate verification of the IF97 Region 3 function, flag IFLAG97(13) is set to 1 (for example, IEXT = ISET97(13,1). With IFLAG97(13) set to 1, VPSBK3 does not check the input with IRGHS97. Use of this flag is intended for validation only. (See code example 8a.)

VHSBK3(h,s)

In IF97 Region 3, function VHSBK3(h,s) is used to compute the specific volume given h and s (as described in Section 3.5 of IAPWSi). VHSBK3 uses IRGHS97 to validate the (h,s) input. If the point is not in IF97 Region 3, the return value is set to -1.0. See Appendix A for additional information on errors.

VR3PT97B(p,T)

In IF97 Region 3, function VR3PT97B(p,T) is used to compute the specific volume given p and T as described in Tables 2 and 10 of IAPWSj. VR3PT97B used functions V3APT97B through V3ZPT97B and TR3PT97B to determine the sub-region containing the specific volume for the input values of p and T. IERR97(1) is set to the sub-region value, 1 to 26 for regions A through Z respectively. If the input temperature is equal to the saturation temperature at p, then IERR97(1) is set to 30 and VR3PT97B return the saturation liquid specific volume and PTMANS97(4) contains the saturation vapor specific volume. (***Note***: Using p obtained from PSAT97(T) may not be interpreted as a saturation request, use TSAT97 (PSAT97(T)) for a saturation request.) See pseudo code example 7. (See Appendix E for more information.)

If p is less than p_{sat} at 623.15 K, VR3PT97B returns -1.0 with IERR97(1) set to -1. If p is greater than 100 MPa, VR3PT97B returns -1.0 with IERR97(1) set to -2. If T is less than or equal to 623.15 K, VR3PT97B returns -1.0 with IERR97(1) set to -3. If T is greater than T2397(p), VR3PT97B returns -1.0 with IERR97(1) set to -4.

Pseudo Code Example 7 - VR3PT97B

```
         P = 22.E6                    { INPUT PRESSURE IN PA }
         T = 646.1                    { INPUT TEMPERATURE IN K}
         V = VR3PT97B( P , T )
      IERP = IERR97(1)                { GET ERROR FLAG }
 {RESULT  V = 2.296350553E-03   IERP = 21 VALIDATION TABLE 13 REGION U }
        TS = TSAT97(P)          RESULT {TS=646.856565224764 }
         V = VR3PT97B( P , TS)
      IERP = IERR97(1)
 {RESULT  V = 2.705718040E-03    ERP = 30
 PMANS97(4) = 3.570484262E-03               }
```

IAPWS Backward Equations Summary

	High-level	Region	Mid-level	Region	Low-level	Region
T(p,h)	TPH97BK	1,2, 3	TPH97B	1, 2	TPHBK1 TPHBK2 TPHBK3	1 2 3
T(p,s)	TPS97BK	1,2, 3	TPS97B	1, 2	TPSBK1 TPSBK2 TPSBK3	1 2 3
p(h,s)	PHS97BK	1,2,3, 4'	PHS97B	1, 2	PHSBK1 PHSBK2 PHSBK3	1 2 3
T(h,s)	THS97BK	1,2,3, 4'	THS97B	1, 2	THSBK1 THSBK2 THSBK3 TSHS97B	1 2 3 4'
x(h,s)	XHS97B					4'
v(p,h)	VPHBK3					3
v(p,s)	VPSBK3					3
v(h,s)	VHSBK3					3
v(p,T)	VR3PT97B					3

It is recommended that the high-level functions be used for general modeling. The mid-level functions are maintained for legacy coding until the user's model is updated to use the new high-level functions. The low-level functions should only be used following appropriate calls to either IRGHS97 or IRPHS97, if necessary, to determine the correct IF97 backward region.

ASTEM97 flag IFLAG97(13) is available for the validation of functions TPSBK3 and VPSBK3, and is intended for this use only. (See code example 8a.)

ASTEM97 flag IFLAG97(14) is available for use to force T(p,h), T(p,s), p(h,s) and T(h,s) to return the minimum or maximum temperature or pressure as defined by the IF97 (p,T) range. This would allow the user to continue with calculations using the returned p or T value. (See code example 8b.)

Pseudo Code Example 8a - TPSKB3 and VPSBK3(Data Validation)

```
     P = 20.E6                      { INPUT PRESSURE IN PA }
     S = 3.7                        { INPUT ENTROPY IN KJ/KG-K}
     T = TPSBK3( P , S )            { NOTE: IFLAG97(13) = 0 BY DEFAULT }
     IERP = IERR97(1)               { GET ERROR FLAG }
     V = VPSBK3( P , S )
     IERV = IERR97(1)
     IREG = IRPHS97( P , S , 2 )  { [VER 2] }
RESULT
     IREG = 1                       {P , S IN IF 97 REGION 1 }
     T = -1.0    IERP = -5          { POINT OUTSIDE REGION 3 DOMAIN }
     V = -1.0    IERV = -5
     IEXT = ISET97(13,1)            { SET FLAG FOR VALIADTION }
     T = TPSBK3( P , S )
     IERP = IERR97(1)
     V = VPSBK3( P , S )
     IERV = IERR97(1)
RESULT
     T = 6.208841563E+02 IERP = 0 { POINT IS IN REGION 1 }
     V = 1.639890984E-03 IERP = 0
     IEXT = ISET97(13,0)            { RETURN FLAGS TO DEFAULT STATE }
USE RECOMMENDED EQUATION FOR IAPWS T(P,S) AND ASTEM97 EQUATIONS FOR V(P,S)
     T = TPS97BK( P , S )           { [VER 2] }
     IERP = IERR97(1)
     V = VPT97( P , TPS97( P , S )
     IERV = IERR97(1)
     T = 6.208665832E+02 IERP = 1 { POINT IS IN REGION 1 }
     V = 1.639853174E-03 IERV = 1
```

Pseudo Code Example 8b - IAPWS Backward Equations Flag 14

```
 IFLAG = ISET97(14,0)             {Make sure flag at default value }
 P =              980.098000000000 Pa
 T =              273.150000000000 K
 IREG97(P,T) =           1
 H =            -4.121200391105893E-002 kJ/kg
 S =            -1.545244759342555E-004 kJ/kg-K
 IRGHS97(H,S)            1
                        IERR  RETURN VALUE
[VER 2] EQUATION CALLS
 PHS97BK(H,S)            1   977.154413894787 Pa
 THS97BK(H,S)          -10  -1.00000000000000
        PTMANS97(3) =        273.128476906780 K
 TPH97BK(P,H)          -10  -1.00000000000000
        PTMANS97(3) =        273.128476194528 K
 TPS97BK(P,S)          -10  -1.00000000000000
        PTMANS97(3) =        273.146350568734 K
LOW-LEVEL CALLS
 PHSBK1(H,S)             0   977.154413894787 Pa
 THSBK1(H,S)           -10  -1.00000000000000
        PTMANS97(3) =        273.128476906780 K
 TPHBK1(P,H)           -10  -1.00000000000000
        PTMANS97(3) =        273.128476194528 K
 TPSBK1(P,S)           -10  -1.00000000000000
        PTMANS97(3) =        273.146350568734 K
OLD LOW-LEVEL CALLS
 PHS97B(H,S,1)           1   977.154413894787 Pa
 THS97B(H,S,1)          -6  -1.00000000000000
        PTMANS97(3) =        273.128476906780 K
 TPH97B(P,H)            -3  -1.00000000000000
        PTMANS97(3) =        273.128476194528 K
 TPS97B(P,S)            -3  -1.00000000000000
        PTMANS97(3) =        273.146350568734 K


 IFLAG = ISET97(14,1)             {Set flag at return min/max values }
                        IERR  RETURN VALUE
[VER 2] EQUATION CALLS
 PHS97BK(H,S)            1   977.154413894787 Pa
 THS97BK(H,S)            1   273.150000000000 K
 TPH97BK(P,H)            1   273.150000000000 K
 TPS97BK(P,S)            1   273.150000000000 K
LOW-LEVEL CALLS
 PHSBK1(H,S)             0   977.154413894787 Pa
 THSBK1(H,S)             0   273.150000000000 K
 TPHBK1(P,H)             0   273.150000000000 K
 TPSBK1(P,S)             0   273.150000000000 K
OLD LOW-LEVEL CALLS
 PHS97B(H,S,1)           1   977.154413894787 Pa
 THS97B(H,S,1)           1   273.150000000000 K
 TPH97B(P,H)             1   273.150000000000 K
 TPS97B(H,S)             1   273.150000000000 K

 IFLAG = ISET97(14,0)             {Reset flag to default value }
```

ASTEM97 - Backward Equations for p(h,s) and T(h,s)

ASTEM97 includes routines to estimate the pressure or temperature as a functions of (h,s) based on the IF97 equations. Near the critical point, particular if two-phase, the estimates are very sensitive to the inputs. These functions handle all of the IF97 Regions. ***NOTE:*** The **ASTEM97** default calculation methods must be in place before using these routines, specifically for the use of ROOT3MAX. Two functions are provided to allow the user to chosen the primary return value, pressure or temperature. See Appendix E for additional information on the backward equations.

PHS97(h,s) and THS97(h,s)

PHS97 and THS97 check the input values. If the input entropy or enthalpy is out side the range established in IF97, the return value is set to -1.0 and the error flag is set as described in Appendix A. Once the pressure is determined, the temperature is stored and can be retrieved by a call to PTMANS97 with a call index of 3. For saturated or 2-phase conditions, the **ASTEM97** default return value is quality (X) unless the user selects to have the saturation pressure returned by setting an internal flag (ISET97(11,1)). Alternatively, the saturation pressure can be obtained from a call to PSAT97 at T. The resulting (p,T) can then be used to determine the other thermodynamic or transports properties. THS97 simply calls PHS97 and returns the temperature. Once the temperature is determined, the pressure is stored and can be retrieved by a call to PTMANS97 with a call index of 1. PHS97 keeps a copy of the input (h,s) set, so a call to PHS97 followed by a call to THS97 with the same (h,s) input will not result in a new computation but only a transfer of the previously computed (p,T) set — without the need to use PTMANS97. See code example 9.

To ensure a consistent set of parameters based on an (h,s) pair, the return values from PHS97(h,s) and THS97(h,s) are set based on the value of the internal flag, IFLAG97(11) as follows:

With the internal flag set to its default value of 0 (ISET97(11,0):

PHS97(h,s) return the quality if saturated or 2-phase and THS97(h,s) return the temperature (K)

Note: PTMAN97(3) also contains the temperature value after call to PHS97(h,s)

Note: PTMAN97(1) also contains the quality value after call to THS97(h,s)

With the internal flag set to a value of 1 (ISET97(11,1):

PHS97(h,s) return the pressure (Pa) and THS97(h,s) return the quality if saturated or 2-phase

Note: PTMAN97(3) also contains the quality value after call to PHS97(h,s)

Note: PTMAN97(1) also contains the pressure value after call to THS97(h,s)

Pseudo code Example 9 - p(h,s) and T(h,s) routines

```
Use IAPWS Region 1 validation point 1 from Table 3 (IAPWSg)
IAPWS Equations                           ASTEM97 Equations
    H_IN   = 0.001    {KJ/KG }               H_IN   = 0.001    {KJ/KG }
    S_IN   = 0.0      {KJ/KG-K }             S_IN   = 0.0      {KJ/KG-K }
    P_CALC = PHS97B(H_IN,S_IN,1)             P_CALC = PHS97(H_IN,S_IN)
    { RETURN EFLAG = 1                       { RETURN  EFLAG = 0
            P_CALC = 980.0980612 }                    P_CALC = 983.0663482
                                                  PTMANS97(3) = 273.1600024 }
    T_CALC = THS97B(H_IN,S_IN,1)              T = THS97(H_IN,S_IN)
    { RETURN EFLAG = -6                       { RETURN EFLAG = 0
            T_CALC = -1.0                             T_CALC = 273.1600024
       PTMANS97(3) = 273.1385417 }                PTMANS97(1) = 983.0663482 }
 NOTE: THS97B RETURNS A TEMPERATURE VALUE LESS THAN THE IF97 LOWER LIMIT OF
       273.15 K, BUT THE VALUE IS WITHIN THE ACCURACY RANGE OF ± 25 MK.
[Ver 2]NOTE: IF IFLAG97(14) IS SET TO 1 THS97B RETURNS T-MIN OF 273.15 K
       USE OF THE CALCULATED VALUE WITHIN ASTEM97 WOULD RESULT IN RANGE
       ERRORS. TO ADDRESS THIS POTENTIAL ISSUE ASTEM97 ROUTINES TPS97 OR TPH97
       CAN BE USED TO GET TEMPERATURE BASED ON THE VALID PRESSURE VALUE
     T_BASED_ON_S = TPS97(P_CALC,S_IN)     T_BASED_ON_H = TPH97(P_CALC,H_IN)
     { RETURN
       T_BASED_ON_S = 273.1600024          T_BASED_ON_H = 273.1600030 }
```

If the state-point is found to be near the saturation line, a quality of 0 (zero) or 1 (one) $\pm 1.0\text{x}10^{-5}$, then the return quality value is set to 0 or 1.

ASTEM97 - Backward Equations for p(v,h) and T(v,h)

ASTEM97 includes routines to estimate the pressure or temperature as a functions of (v,h) based on the IF97 equations. Near the critical point, particular if two-phase, the estimates are very sensitive to the inputs. These functions all of the IF97 Regions. ***NOTE:*** The **ASTEM97** default calculation methods must be in place before using these routines, specifically for the use of ROOT3MAX. Two functions are provided to allow the user to chosen the primary return value, pressure or temperature. See Appendix E for additional information on the backward equations.

PVH97(v,h) and TVH97(v,h)

PVH97 and TVH97 check the input values. If the input specific volume or enthalpy is out side the range established in IF97, the return value is set to -1.0 and the error flag is set as described in Appendix A. Once the pressure is determined, the temperature is stored and can be retrieved by a call to PTMANS97 with a call index of 3. For saturated or 2-phase conditions, the **ASTEM97** default return value is quality (X) unless the user selects to have the saturation pressure returned by setting an internal flag (ISET97(11,1)). Alternatively, the saturation pressure can be obtained from a call to PSAT97 at T. The resulting (p,T) can then be used to determine the other thermodynamic or transports properties. TVH97 simply calls PVH97 and returns the temperature. Once the temperature is determined, the pressure is stored and can be retrieved by a call to PTMANS97 with a call index of 1. PVH97 keeps a copy of the input (v,h) set, so a call to PVH97 followed by a call to TVH97 with the same (v,h) input will not result in a new computation but only a transfer of the previously computed (p,T) set — without the need to use PTMANS97.

If the state-point is found to be near the saturation line, a quality of 0 (zero) or 1 (one) $\pm 1.0 \times 10^{-5}$, then the return quality value is set to 0 or 1.

[**Ver 2**] The convergence criteria used in PVH97 and TVH97 have been relaxed to address a few cases of non-convergence.

To ensure a consistent set of parameters based on an (v,h) pair, the return values from PVH97(v,h) and TVH97(v,h) are set based on the value of the internal flag, IFLAG97(11) as follows:

With the internal flag set to its default value of 0 (ISET97(11,0):

PVH97(v,h) return the quality if saturated or 2-phase and TVH97(v,h) return the temperature (K)

Note: PTMAN97(3) also contains the temperature value after call to PVH97(v,h)

Note: PTMAN97(1) also contains the quality value after call to TVH97(v,h)

With the internal flag set to a value of 1 (ISET97(11,1):

PVH97(v,h) return the pressure (Pa) and TVH97(v,h) return the quality if saturated or 2-phase

Note: PTMAN97(3) also contains the quality value after call to PVH97(v,h)

Note: PTMAN97(1) also contains the pressure value after call to TVH97(v,h)

ASTEM97 - Backward Equations for p(v,s) and T(v,s)

ASTEM97 includes routines to estimate the pressure or temperature as a functions of (v,s) based on the IF97 equations. Near the critical point, particular if two-phase, the estimates are very sensitive to the inputs. These functions handle all of the IF97 Regions. ***NOTE:*** The **ASTEM97** default calculation methods must be in place before using these routines, specifically for the use of ROOT3MAX. Two functions are provided to allow the user to chosen the primary return value, pressure or temperature. See Appendix E for additional information on the backward equations.

PVS97(v,s) and TVS97(v,s)

PVS97 checks the input values. If the input specific volume or entropy is out side the range established in IF97, the return value is set to -1.0 and the error flag is set as described in Appendix A. Once the pressure is determined, the temperature is stored and can be retrieved by a call to PTMANS97 with a call index of 3. For saturated or 2-phase conditions, the **ASTEM97** default return value is quality (X) unless the user selects to have the saturation pressure returned by setting an internal flag (ISET97(11,1)). Alternatively, the saturation pressure can be obtained from a call to PSAT97 at T. The resulting (p,T) can then be used to determine the other thermodynamic or transports properties. TVS97 simply calls PVS97 and returns the temperature. Once the temperature is determined, the pressure is stored and can be retrieved by a call to PTMANS97 with a call index of 1. PVS97 keeps a copy of the input (v,s) set, so a call to PVS97 followed by a call to TVS97 with the same (v,s) input will not result in a new computation but only a transfer of the previously computed (p,T) set — without the need to use PTMANS97.

If the state-point is found to be near the saturation line, a quality of 0 (zero) or 1 (one) $\pm 1.0\text{x}10^{-5}$, then the return quality value is set to 0 or 1.

[**Ver 2**] The convergence criteria used in PVS97 and TVS97 have been relaxed to address a few cases of non-convergence.

To ensure a consistent set of parameters based on an (v,s) pair, the return values from PVS97(v,s) and TVS97(v,s) are set based on the value of the internal flag, IFLAG97(11) as follows:

With the internal flag set to its default value of 0 (ISET97(11,0):

PVS97(v,s) return the quality if saturated or 2-phase and TVS97(v,s) return the temperature (K)

Note: PTMAN97(3) also contains the temperature value after call to PVS97(v,s)

Note: PTMAN97(1) also contains the quality value after call to TVS97(v,s)

With the internal flag set to a value of 1 (ISET97(11,1):

PVS97(v,s) return the pressure (Pa) and TVS97(v,s) return the quality if saturated or 2-phase

Note: PTMAN97(3) also contains the quality value after call to PVS97(v,s)

Note: PTMAN97(1) also contains the pressure value after call to TVS97(v,s)

Equations for v(p,T) in IF97 Region 3

VPTREG3(p,T) and IVPT97(p,T)

H.-J. Kretzschmar, et al., have proposed a backward equation for obtaining a first-cut of the specific volume in IF97 Region 3 based on pressure and temperature (1999a). The proposed v(p,T) equation is comprised of 7 sub-regions. Mid-level Function VPTREG3(p,T) is available to return the specific volume at (p,T). VPTREG3 uses low-level routine IVPT97(p,T) to determine which sub-region contains the (p,T) point and calls the appropriate sub-region low-level routine (VPT3A97, VPT3B97, VPT3C97, VPT3D97, VPT3E97, VPT3F97, or VPT3G97. VPTREG3 is used in **ASTEM97** in conjunction with ROOT3 and ROOT3MAX.

User Interface Routines

IERR97(Iflag)

Error handling is treated through the use of an error flag. Error checks are performed in most routines, see Appendix A. When an error is located in a property-related routine, the routine generally returns a negative one (-1.0) and sets the error flag. The user can retrieve the value of the error flag with Function IERR97(Iflag) with Iflag set to 1. Appendix A lists the error flags for each routine.

In **Version 2.0**, an iteration counter has been added to the routines identified in the table below. IFLAG97(15) will contain the routine identifier if the number of iterations exceeds a pre-set, large value. No notice is provided as it is likely the solution is converging, just not to the pre-set convergence criterion. After relaxing the convergence criterion for PVH97, TVH97, PVS97 and TVS97, no situation has yet been identified where the iteration criterion has been reached. The purpose of adding the counter is to preclude an infinite loop. The value of IFLAG97(15) can be retrieved by a call to IERR97(15) at the end of a problem to verify convergence, with a return value of 0.

ASTEM97 Routine	IFLAG97(15) Value
OVHS97	-101
PHS97	-102
PVAR97	-103
PVHS97	-104
ROOT3	-105
ROO3L	-106
ROOTMAX	-107
TVAR97	-108
XREG1MM	-109
XREG1MP	-110
XREG1PM	-111
XREG1PP	-112

PROP97(Index)

PROP97 returns the value of the property specified by Index, see Table 1. PROP97 is used after call to high-level routines (XPROP97, PTPROP97, PXPROP97 or TXPROP97), or after a series of calls to mid-level Gibb or Helmholtz routines. For example, to get the coefficient of thermal expansion, Value = PROP97(15).

CSEU97(Ifromto,Value)

CSEU97 converts the Value from SI to English if Ifromto is positive (+) or from English to SI if Ifromto is negative (-). Ifromto is the property Index, see Table 1. For example, to convert pressure from Pa to psia, PSIA = CSEU97(1,PA), or to convert specific volume from ft^3/lbm to m^3/kg, SVSI = CSEU97(-3,SVENG).

ASTEM97 tracks 15 internal flags, and the current value can be retrieved by IERR97. Flags 1 and 12 are used to trace errors. Flag 1 is the error code and Flag 12 is the identifier of the last called routine (not necessarily the last user called routine). Flags 5, 6, 9, 10 , 11, 13 and 14 are available to the user to modify (see ISET97) the default calculations in **ASTEM97**.

```
IFLAG97   1 INTERNAL                REGION FLAG, IF NEGATIVE REGION ERROR
          2 INTERNAL                USED TO SKIP 2ND ORDER CALCS WHEN 2-PHASE
          3 INTERNAL PHS97B         ENTHALPY REGION ERROR FLAG
          4 INTERNAL PHS97B         ENTROPY REGION ERROR FLAG
          5 USER                    COMPUTE METASTABLE VALUES
          6 USER     CPPT3          COMPUTE POSITIVE CP (NEAR PCRT,TCRT)
          7 INTERNAL                USED TO MODIFY FUNCTION RETURNS
          8 INTERNAL XREG1xx        CHECK RANGE VALIDITY ONLY
          9 USER     ROOT3MAX       APPLY RHOV CORRECTION
         10 USER     ROOT3MAX       APPLY ASME-LIKE RHOL/RHOV CALCS
         11 USER     PVAR97,TVAR97  IF=0 RETURN X, ELSE RETURN P OR T
                     PHS97 ,THS97
                     PVH97 ,TVH97
                     PVS97 ,TVS97
         12 INTERNAL                ROUTINE IDENTIFIER (SEE ZZERR97)
[Ver 2]  13 USER     REGION 3 BACKWARD EQS SET TO 1 FOR VALIDATION ONLY
[Ver 2]  14 USER     RETURN P/T MIN/MAX FROM PHSBK1/2/3 OR TP H/S BK1/2
         15 INTERNAL USED FOR DEBUG, TRACK ITERATION FAILURE (UNLIKELY)
```

ISET97(Iflag,Ivalue)

ISET97 is used to set or reset a flag. Only flags 5, 6, 9, 10, 11, 13 and 14 should be modified. Ivalue must be 1 to set a flag or 0 to reset the flag. ISET97 returns a 0 (zero) if successful, and -1 if no change was made.

The effect of setting user flag 11 from it's default value of 0 to 1 on the return values from the **ASTEM97** backward equations, and where relevant the PTMANS97 return values, is shown below (Note PTU97 and TPU97 are not shown):

```
Results for 2-phase condition
Input:                     Pressure (Pa) 10000000.0000000
                          Quality (---)        0.500000000000000
Calculated:  Saturation Temperature (K)      584.149487998528
               Specific volume (m^3/kg)        9.743097546489474E-003
             Specific entropy (kJ/kg-K)        4.48809027947611
              Specific enthalpy (kJ/kg)     2066.67003350349
IFLAG97(11) =  0
Call      p or x        PTMANS97(3) T/x    Call         T or x       PTMANS97(1) p/x
PVH97 0.500000000006197  584.149487999218  TVH97 584.149487999218  0.500000000006197
PVS97 0.500000004186326  584.149488463769  TVS97 584.149488463769  0.500000004186326
PHS97 0.499999999980763  584.149488066179  THS97 584.149488066179  0.499999999980763
PTV97 0.500000000000000                    TPV97 0.499999999999989
PTS97 0.500000000000000                    TPS97 0.499999999999996
PTH97 0.500000000000000                    TPH97 0.499999999999997
IFLAG97(11) =  1
Call      p or x        PTMANS97(3) T/x    Call         T or x       PTMANS97(1) p/x
PVH97 10000000.0000939  0.500000000006197  TVH97 0.500000000006197   10000000.0000939
PVS97 10000000.0632901  0.500000004186326  TVS97 0.500000004186326   10000000.0632901
PHS97 10000000.0092031  0.499999999980763  THS97 0.499999999980763   10000000.0092031
PTV97 10000000.0000001                     TPV97  584.149487998528
PTS97 10000000.0000001                     TPS97  584.149487998528
PTH97 10000000.0000001                     TPH97  584.149487998528

Results for 1-phase condition
Input:       Specific entropy (kJ/kg-K)       9.50000000000000
              Specific enthalpy (kJ/kg)    4100.00000000000
Calculated:                Pressure (Pa)  102481.098436935
             Saturation Temperature (K)    1047.36105708049
               Specific volume (m^3/kg)       4.71581776181857
IFLAG97(11) =  0
Call      p or x        PTMANS97(3) T/x    Call         T or x       PTMANS97(1) p/x
PVH97 102481.098434079   1047.36105705501   TVH97 1047.36105705501  102481.098434079
PVS97 102481.098427203   1047.36105698367   TVS97 1047.36105698367  102481.098427203
PHS97 102481.098436935   1047.36105708049   THS97 1047.36105708049  102481.098436935
PTV97 102481.098436932                      TPV97 1047.36105708049
PTS97 102481.098459719                      TPS97 1047.36105708049
PTH97 102481.121248107                      TPH97 1047.36105706013
IFLAG97(11) =  1
Call      p or x        PTMANS97(3) T/x    Call         T or x       PTMANS97(1) p/x
PVH97 102481.098434079   1047.36105705501   TVH97 1047.36105705501   102481.098434079
PVS97 102481.098427203   1047.36105698367   TVS97 1047.36105698367   102481.098427203
PHS97 102481.098436935   1047.36105708049   THS97 1047.36105708049   102481.098436935
PTV97 102481.098436932                      TPV97 1047.36105708049
PTS97 102481.098459719                      TPS97 1047.36105708049
PTH97 102481.121248107                      TPH97 1047.36105706013
```

PTMANS97(Index)

PTMANS97 is used to retrieve information from various routines when more than one return value may be possible or to retrieve a calculated value that is outside the IF97 range, based on the value of a flag. See for example, "Pressure Related Functions," "Metastable Region," and "ASTEM97 - Backward Equations for p(h,s) and T(h,s)," "ASTEM97 - Backward Equations for p(v,h) and T(v,h)," and "ASTEM97 - Backward Equations for p(v,s) and T(v,s)."

```
AFTER A CALL TO PTH97, PTS97, OR PTU97
INDEX = 1 RETURNS 2-ND REGION 1 PRESSURE
      = 2 RETURNS TWO-PHASE QUALITY/PRESSURE
      = 3 RETURNS THE PRESSURE AT DVAR/DP|T = ZERO (SEE APPENDIX D)

AFTER A CALL TO METASTABLE OR PRESSURE-BASED BACKWARD EQUATION
INDEX = 3 RETURNS TEMPERATURE VALUE
AFTER A CALL TO METASTABLE OR TEMPERATURE-BASED BACKWARD EQUATION
INDEX = 1 RETURNS PRESSURE VALUE
```

PTMANS97 must be called immediately following the function call that returns a value by PTMANS97, as summarized in the table below. The return values are initially set to -1.0, consistent with an invalid call to a function. After a value is retrieved, the internal value is reset to -1.0. By resetting the internal value, a call to PTMANS97 later in a program will return a -1.0 if the internal value is no longer valid, for example the function call corrected the out-of-range parameters and the PTMANS97 value is no longer valid.

Table 6 - Summary of PTMANS97 Return Values

ASTEM97 Function(s)	Check error flag	Error ?	ptmans97 (ivalue)	Returned parameter
TPMETA	ierr97(1) < 0	Y	3	T_{meta}
THMETA	ierr97(1) < 0	Y	3	T_{meta}
				single [or 2-phase] (based on IFLAG 11 = 0 / 1)
PHS97	ierr97(1) >= 0	N	3	T [x / T]
THS97	ierr97(1) >= 0	N	1	p [x / p]
PVH97	ierr97(1) >= 0	N	3	T [x / T]
PVS97				
TVH97	ierr97(1) >= 0	N	1	p [x / p]
TVS97				
VR3PT97B	ierr97(1) = 30	N	4	v-vapor
PTV97	ierr97(1) = 10	N	1	p
PTU97			2	not valid
PTH97	ierr97(1) = 11	N	1	not valid
PTS97			2	x / p
	ierr97(1) = 12	N	1	p
			2	x / p
ITEM97	ierr97(1) = 10	-	3	p at inflection
	or		4	var at inflection
	ierr97(1) = 12		5	var a p-low
			6	var at p-high

ZZERR97(Index,Name)

ZZERR97 returns the Name of the routine identified by Index. Name is a FORTRAN CHARACTER*8 string. Index comes from internal flag 12 which is retrieved from IERR97(12). For example, a call to `ZZERR97(17,WHOAMI) returns WHOAMI = "CVPT97 "`.

ZTITL97(Index,Name1,Sunit,Eunit)

ZTITL97 returns the parameter name (Name1), the SI unit (Sunit) and the English unit (Eunit) identified by Index, see Table 1. Name1 is a FORTRAN CHARACTER*26 string, right justified. Sunit is a FORTRAN CHARACTER*14 string, left justified. Eunit is a FORTRAN CHARACTER*14 string, left justified. For example, a call to `ZTITL97(3,PARM,SIU,SEU)`

```
returns PARM = "           Specific Volume",
        SIU  = "m^3/kg        "
    and SEU  = "ft^3/lbm      ".
```

ZZZZZZ97(Name)

ZZZZZZ97 returns Name, an array of five strings, of FORTRAN CHARACTER*40.

```
      CHARACTER*40  LABIAM(5)
      LABIAM(1) = ' ASTEM97 IAPWS INDUSTRIAL FORMULATION '
      LABIAM(2) = '      FOR WATER AND STEAM IF-97       '
      LABIAM(3) = ' Edward D. Throm (C) 2005 Version 2.0 '
      LABIAM(4) = '     E-mail :  mister-ed@cox.net      '
      LABIAM(5) = '   http://members.cox.net/mister_ed   '
```

ZUNIT97(Index)

ZUNIT97 provides and interface to the **ASTEM97** constant. This allows the user the flexibility to use the identical stored values within a program. Table 6 lists the constants and their value.

Table 7 - ASTEM97 Constants

Index	Name	Value	Parameter
1	T273	273.15	Minimum temperature (K)
2	T623	623.15	Region 3 transition temperature (K)
3	T863	863.15	Maximum temperature, 2-3 boundary (K)
4	TMAX	1073.15	Maximum temperature (K)
5	TLAR	2273.15	Maximum temperature, Region 5 (K)
6	PMIN	611.212677444345	Minimum pressure (Pa)
7	P010	10.D6	Maximum pressure, Region 5 (Pa)
8	PMAX	100.D6	Maximum pressure (Pa)
9	P623	16529164.2526045	Region 3 transition pressure (Pa)
10	RGAS	0.461526	Specific gas constant (kJ/kg-K)
11	GASC	8.31451	Molar gas constant (J/mol-K)
12	OMOL	18.015257	Molar mass of ordinary water (g/mol)
13	TCRT	647.096	Critical temperature (K)
14	PXCRT	22.064	Critical pressure (MPa)
15	PCRT	22.064D6	Critical pressure (Pa)
16	RCRT	322.0	Critical density (kg/m^3)
17	TTRP	273.16	Triple point temperature (K)
18	PTRP	611.657	Triple point pressure (Pa)
19	TBOL	373.1234	Boiling point (K)
20	PRS2E	6.894757 D+03	Pressure conversion
21	HUS2E	2.326 D+00	Enthalpy conversion
22	CHS2E	4.1868 D+00	Entropy conversion
23	VDS2E	1.601846 D+01	Specific volume conversion
24	TCS2E	1.730735 D+00	Thermal conductivity conversion
25	SVS2E	3.048 D-01	Speed of sound conversion
26	DVS2E	1.488164 D+06	Dynamic viscosity conversion
27	STS2E	1.459390 D+04	Surface tension conversion

Note:
English units

10	RGAS	0.1102336	Specific gas constant (BTU/lbm-R)
11	GASC	1.9859	Molar gas constant (BTU/lbmol-R)

Retrieve Dimensionless Gibbs or Helmholtz Values

DERV97(Ivalue)

Should there be a need to retrieve the dimensionless Gibbs or Helmholtz values and their partial derivative values (see Equations 7, 15, 18, 28 and 32 in IAPWSa), function DERV97 can be called after a call to the appropriate mid-level routine (GIBB1, GIBB2, GIBB2I, GIBBM, GIBBMI, GIBB5, GIBB5I or HELM3). For Ivalue positive, the base or ideal gas part is returned. For Ivalue negative, the residual gas part is returned. Ivalue must be in the range from +1 to +10 or from -1 to -6. See Table 7 for a cross-reference of the **ASTEM97** parameter to the IAPWS IF97 parameter.

See "Pseudo code Example 10" under "Notes on Joule-Thomson Coefficients" for an example on how to use DERV97 in combination the Gibbs or Helmholtz mid-level routines.

Table 8 - Gibbs and Helmholtz dimensionless values

Base (Region 1 and 3) Ideal Gas Part (Region 2, 5 and 2 supplemental)	Residual Gas Part (Region 2, 5 and 2 supplemental)	Gibbs equation[(1)] (Region 1, 2, 5 and 2 supplemental)	Helmholtz equation (Region 3)
G0(1)	GR(1)	$\gamma^P(\pi,\tau)$	$\phi(\delta,\tau)$
G0(2)	GR(2)	$\gamma^P_\pi = \left[\frac{\partial\gamma^P}{\partial\pi}\right]_\tau$	$\phi_\delta = \left[\frac{\partial\phi}{\partial\delta}\right]_\tau$
G0(3)	GR(3)	$\gamma^P_\tau = \left[\frac{\partial\gamma^P}{\partial\tau}\right]_\pi$	$\phi_\tau = \left[\frac{\partial\phi}{\partial\tau}\right]_\delta$
G0(4)	GR(4)	$\gamma^P_{\pi\pi} = \left[\frac{\partial^2\gamma^P}{\partial\pi^2}\right]_\tau$	$\phi_{\delta\delta} = \left[\frac{\partial^2\phi}{\partial\delta^2}\right]_\tau$
G0(5)	GR(5)	$\gamma^P_{\tau\tau} = \left[\frac{\partial^2\gamma^P}{\partial\tau^2}\right]_\pi$	$\phi_{\tau\tau} = \left[\frac{\partial^2\phi}{\partial\tau^2}\right]_\delta$
G0(6)	GR(6)	$\gamma^P_{\pi\tau} = \left[\frac{\partial^2\gamma^P}{\partial\pi\partial\tau}\right]$	$\phi_{\delta\tau} = \left[\frac{\partial^2\phi}{\partial\delta\partial\tau}\right]$
G0(7)	---	p_{in}	$\delta = \rho_{in}/\rho^*$
G0(8)	---	T_{in}	ρ_{in}
G0(9)	---	$\pi = p_{in}/p^*$	T_{in}
G0(10)	---	$\tau=T^*/T_{in}$	$\tau = T^*/T_{in}$

(1) ***NOTE***: In IAPWSa γ^P, P is "0" for the ideal gas part and P is "r" for the residual gas part.

Practical User Interface

While **ASTEM97** contains 307 routines, a practical user interface at the single property level, enthalpy for example, only requires 22 functions. A set of two functions for each of the basic 7 properties and 4 partial derivatives, identified in Table 5. For example, for enthalpy, the user needs HPT97(p,T) and either HPX97(p,X) or HTX97(T,x) for the saturation line, without the need for the partial derivatives, this set is reduced to 14 functions.

The **ASTEM97** (recommended) and IAPWS backward equations add an additional set of up to six functions for a practical user interface: p(h,s), T(h,s), p(v,h),T(v,h) and p(v,s),T(v,s).

Unit conversion and error checking requires CSEU97 and IERR97. Under special circumstances, overriding default calculations requires ISET97.

ASTEM97 also provides equations for additional properties, as discussed in the following sections.

Additional Property Functions

Six additional properties are available though **ASTEM97** supplied functions. Two functions are provided for the thermal conductivity, one for industrial use and one for general or scientific use. Functions for the dynamic viscosity, surface tension, static dielectric constant and reflective index are also provided.

Thermal Conductivity (k_{IND}) for Industrial Use

The thermal conductivity (k_{IND}) is calculated in units of W/m-K. IAPWS endorses the validity of the equation for the thermal conductivity of ordinary water for industrial use (IAPWSb) in the following range of pressures p and temperatures T:

$p \leq 100$ MPa for 0 °C $\leq$ T $\leq$ 500 °C
$p \leq 70$ MPa for 500 °C $<$ T $\leq$ 650 °C
$P \leq 40$ MPa for 650 °C $<$ T $\leq$ 800 °C

Two functions are provided in **ASTEM97**:

FUNCTION **TC85RHO**(ρ , T): TC85RHO takes density and temperature as input, consistent with the IAPWS function development. [**Ver 2**] The density and temperature are checked. If exceeded, the return value is -1.0 and the error flag is set as described in Appendix A.

FUNCTION **TC85PRS**(p , T): TC85PRS takes pressure and temperature as input, allowing use of a consistent p,T model. ***NOTE***: for saturated values, the density at the saturation point should be calculated and TC85RHO used to obtain the thermal conductivity. The temperature and pressure are checked. If exceeded, the return value is -1.0 and the error flag is set as described in Appendix A.

Thermal Conductivity (k_{GSI}) for General and Scientific Use

The thermal conductivity (k_{GSI}) is calculated in units of W/m-K IAPWS endorses the validity of the equation for the thermal conductivity of ordinary water for general and scientific use (IAPWSb) in the following range of pressures p and temperatures T

$p \leq 400$ MPa for 0 °C $\leq$ T $\leq$ l25 °C
$p \leq 200$ MPa for 125 °C $<$ T $\leq$ 250 °C
$p \leq$ l50 MPa for 250 °C $<$ T $\leq$ 400 °C
$p \leq 100$ MPa for 400 °C $<$ T $\leq$ 800 °C

Two functions are provided in **ASTEM97**:

FUNCTION **TC97RHO**(ρ , T): TC97RHO takes density and temperature as input, consistent with the IAPWS function development.

FUNCTION **TC97PRS**(p , T): TC97PRS takes pressure and temperature as input, allowing use of a consistent p,T model. ***NOTE Special Consideration***: for saturated values, the user supplies a value of 0 (zero) for p (pressure) to obtain the saturated fluid thermal conductivity, and a value of 1 (one) for the saturated vapor thermal conductivity.

In **ASTEM97** the IF97 region range is checked. If the point is not in region 1, 2 or 3, the return value is -1.0 and the error flag is set to -IREG (as determined by a function ITEM97). ρ is not checked as the density should be obtained from the IF97 equations. To avoid numerical problems near the critical point, an adjustment to the correlation presented in IAPWSb is made, see Appendix C.

[**Ver 2**] The calculation for k_{GSI} includes the viscosity adjustment presented in IAPWSc2 (August 2003 Release) for the dynamic viscosity for general and scientific use, in the range $0.996 \leq T^* \leq 1.01$ and $0.71 \leq \rho^* \leq 1.36$. (In the September 1997 release, IAPWSc1, these ranges were: $0.9970 \leq T^* \leq 1.0082$ and $0.775 \leq \rho^* \leq 1.290$.) In **ASTEM97**, the dynamic viscosity is obtained for the industrial use formulation.

Figures 3 through 6 provide examples of the effects of the dynamic viscosity adjustment on the calculated value for k_{GSI}.

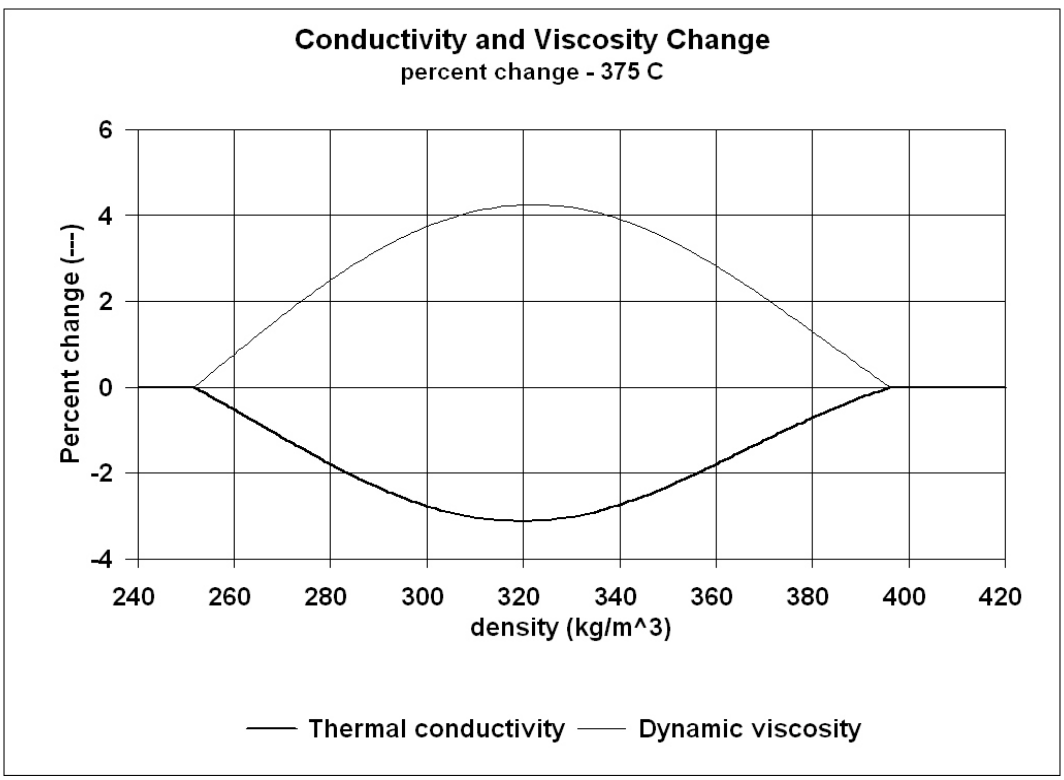

Figure 3 - Effects of dynamic viscosity adjustment

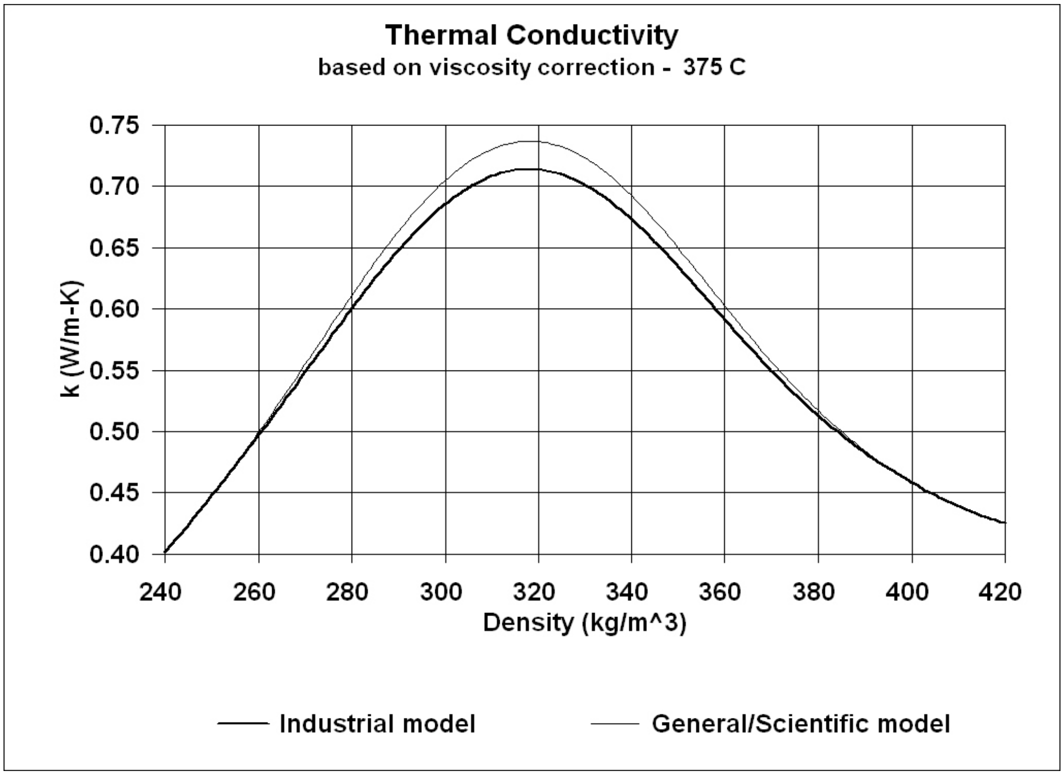

Figure 4 - Effect of dynamic viscosity adjustment on k_{GSI}

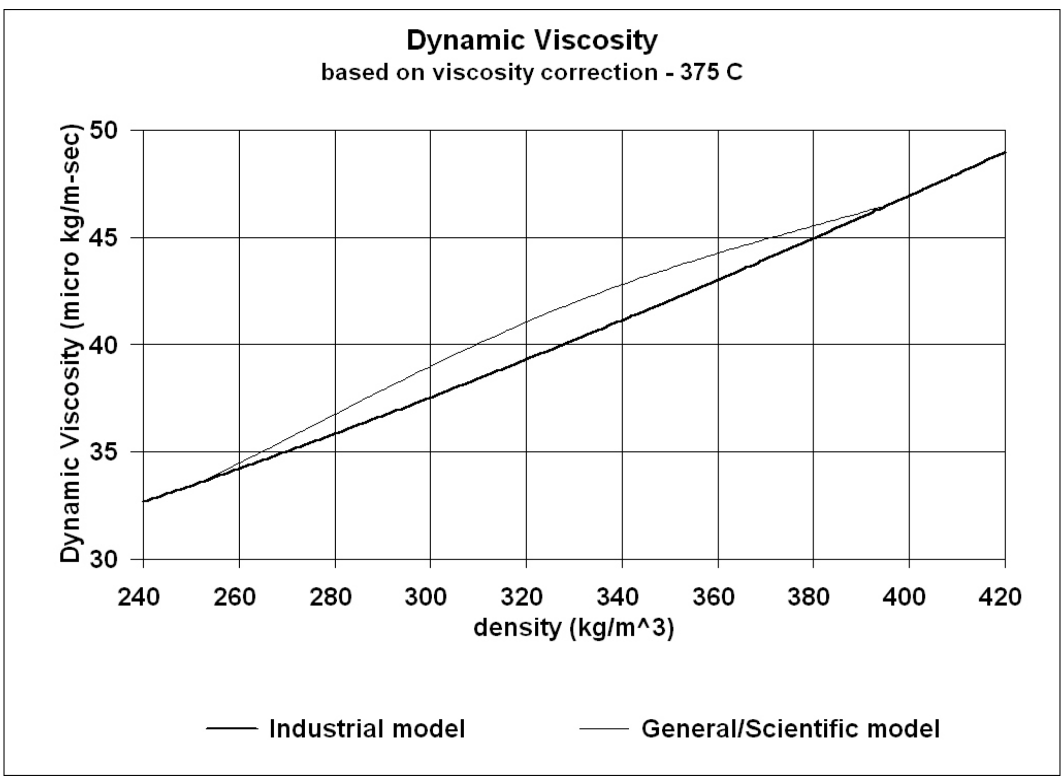

Figure 5 - Effect of dynamic viscosity adjustment on η

Dynamic Viscosity (η)

The dynamic viscosity (η) is calculated in units of μ- kg/m-sec (micro-kg/m-sec). IAPWS endorses the validity of equation for the dynamic viscosity of ordinary water (IAPWSc) in the following range of pressures p and temperatures T:

$p \leq 500$ MPa for $0\ ˚C \leq T \leq 150\ ˚C$
$p \leq 350$ MPa for $150\ ˚C < T \leq 600\ ˚C$
$p \leq 300$ MPa for $600\ ˚C < T \leq 900\ ˚C$

Two functions are provided in **ASTEM97**:

FUNCTION **DYNVRHO**(ρ , T): DYNVRHO takes density and temperature as input, consistent with the IAPWS function development. [**Ver 2**] The density and temperature are checked. If exceeded, the return value is -1.0 and the error flag is set as described in Appendix A

FUNCTION **DYNVPRS**(p , T): DYNVPRS takes pressure and temperature as input, allowing use of a consistent (p,T) model. ***NOTE***: for saturated values, the density at the saturation point should be calculated and DYNVRHO used to obtain the dynamic viscosity. The temperature and pressure are checked. If exceeded, the return value is -1.0 and the error flag is set as described in Appendix A.

Surface Tension (σ)

The surface tension (σ) is calculated in units of m-N/m (milli-N/m). The IAPWS formulation (IAPWSf) is valid for the temperature range of 273.16 K (the triple-point) to 647.096 K (the critical point).

FUNCTION **SURTEN**(T): SURTEN takes temperature as input, consistent with the IAPWS function development.

In **ASTEM97** if the temperature is outside the valid range the return value is set to -1.0 with the error flag set to -7.

Static Dielectric Constant (ϵ)

The static dielectric constant (ϵ) is a dimensionless number. The IAPWS formulation (IAPWSe) is valid for the temperature range of 238 K to 873 K to a maximum pressure of 1,000 MPa.

Two functions are provided in **ASTEM97**:

FUNCTION **STDIRHO**(ρ , T): STDIRHO takes density and temperature as input, consistent with the IAPWS function development. [**Ver 2**] The density and temperature are checked. If exceeded, the return value is -1.0 and the error flag is set as described in Appendix A

FUNCTION **STDIPRS**(p , T): STDIPRS takes pressure and temperature as input, allowing use of a consistent (p,T) model. ***NOTE***: for saturated values, the density at the saturation point should be calculated and STDIRHO used to obtain the static dielectric constant. The temperature and pressure are checked. If exceeded, the return value is -1.0 and the error flag is set as described in Appendix A.

Refractive Index (n)

The refractive index (n) is a dimensionless number. IAPWS endorses the formulation of the refractive index of ordinary water (IAPWSd) in the following range:

Temperature	-12 ˚C	≤ T	≤ 500 ˚C
Density	0 kg/m^3	≤ ρ	≤ 1060 kg/m^3
Wavelength	0.2 μm	≤ λ	≤ 1.1 μm

Extrapolation of the formulation to longer wavelengths has been tested by IAPWS. The formulation is in good agreement with results in liquid water at wavelengths up to 1.9 μm.

Two functions are provided in **ASTEM97**:

FUNCTION **RINDRHO**(ρ , T , λ): RINDRHO takes density and temperature as input, consistent with the IAPWS function development. RIN is the wavelength. [**Ver 2**] The density and temperature are checked. If exceeded, the return value is -1.0 and the error flag is set as described in Appendix A

FUNCTION **RINDPRS**(p , T , λ): RINDPRS takes pressure and temperature as input, allowing use of a consistent (p,T) model. ***NOTE***: for saturated values, the density at the saturation point should be calculated and RINDRHO used to obtain the refractive index. λ is the wavelength. The temperature and pressure are checked. If exceeded, the return value is -1.0 and the error flag is set as described in Appendix A.

In **ASTEM97** if λ is less then 0.2 μm then λ is set to 0.2 μm, and if λ is greater than 1.9 μm then λ is set to 1.9 μm. In the **ASTEM97** built-in high- and mid-level routines, λ is set to 1.0 μm as a default.

Additional Property Relationships

ASTEM97 provides the user with additional flexibility to obtain thermodynamic properties from the basic set of properties. Ten additional properties are readily obtained from the basic set. No specific functions are provided, however the high- and mid-level routines contain internal calculations for these properties.

Kinematic Viscosity (υ)

The kinematic viscosity (υ) is calculated in units of μ-m^2/sec (micro-m^2/sec). The kinematic viscosity is defined as the dynamic viscosity divided by the density:

$$\upsilon = \frac{\eta}{\rho},\ \text{or}\ \upsilon = v \cdot \eta$$

No specific function is provided in **ASTEM97** for this calculation. The computed specific volume (v) and the calculated dynamic viscosity (η) obtained at the user (p,T) state point are used directly in the high-level **ASTEM97** routines.

Coefficient of Thermal Expansion (β)

The coefficient of thermal expansion (β) is calculated in units of K^{-1} (1/K). It is calculated as:

$$\beta = \frac{1}{v} \cdot \left(\frac{\partial v}{\partial T}\right)_p$$

No specific function is provided in **ASTEM97** for this calculation. The computed specific volume (v) and the calculated partial derivative $(\partial v/\partial T)|_p$ obtained at the user (p,T) state point are used directly in the high-level **ASTEM97** routines.

Isothermal Compressibility (α)

The isothermal compressibility (α) is calculated in units of Pa^{-1} (1/Pa). It is calculated as:

$$\alpha = -\frac{1}{v} \cdot \left(\frac{\partial v}{\partial p}\right)_T$$

No specific function is provided in **ASTEM97** for this calculation. The computed specific volume (v) and the partial derivative $(\partial v/\partial p)|_T$ obtained at the user (p,T) state point are used directly in the high-level **ASTEM97** routines.

Compressibility Factor (Z)

The compressibility factor is (Z) is calculated in dimensionless units. It is calculated from the relationship pv ∝ nRT, as:

$$Z = \frac{p \cdot v}{T} \cdot \left(\frac{M_w \cdot 1.0x10^{-3}}{R_M} \right)$$

where M_w is the molar mass of water, 18.015257 g/mol, and R_M is the molar gas constant, 8.31451 J/mol-K. The $1.0x10^{-3}$ is needed for units conversion. M_w and R_M are available to the user from **ASTEM97** function ZUNIT97, as

R_M = ZUNIT97(11)
M_w = ZUNIT97(12)

No specific function is provided in **ASTEM97** for this calculation. The computed specific volume (v) obtained at the user (p,T) state point is used directly in the high-level **ASTEM97** routines.

Isentropic Exponent (κ)

The isentropic exponent (κ) is calculated in dimensionless units. It is defined as:

$$\kappa = -\frac{v}{p} \cdot \left(\frac{\partial p}{\partial v} \right)_s$$

In **ASTEM97**, the speed of sound (w), or sonic velocity, is used to obtain κ. w is defined as:

$$w = v \cdot \left[-\left(\frac{\partial p}{\partial v} \right)_s \right]^{\frac{1}{2}}$$

therefore:

$$\kappa = \frac{w^2}{p \cdot v}$$

No specific function is provided in **ASTEM97** for this calculation. The computed specific volume (v) and speed of sound (w), or sonic velocity, obtained at the user (p,T) state point are used directly in the high-level **ASTEM97** routines.

Prandtl Number (Pr)

The Prandtl number (Pr) is calculated in dimensionless units. It is calculated as the ratio of the product of the dynamic viscosity times the specific heat at constant pressure to the thermal conductivity:

$$\Pr = \frac{1.0x10^{-3} \cdot \eta \cdot C_p}{k_{IND}}$$

where the factor of $1.0x10^{-3}$ is needed for units conversion.

No specific function is provided in **ASTEM97** for this calculation. The computed dynamic viscosity (η), specific heat (C_p) and the thermal conductivity from the industrial formulation (k_{IND}) obtained at the user (p,T) state point are used directly in the high-level **ASTEM97** routines. k_{GSI}, from the general and scientific formulation, could also be used at the user's discretion. ***NOTE***: At the critical point, Pr will be equal to η, as C_p (with it's numerical correction) will be equal to k_{GSI} (with it's numerical correction).

Gibbs Free Energy (g_{FE})

The Gibbs free energy is g_{FE} = h - Ts. No specific function is provided in **ASTEM97** for this calculation. The computed enthalpy (h) and entropy (s) obtained at the user (p,T) state point are used directly in the high-level **ASTEM97** routines.

Helmholtz Free Energy (h_{FE})

The Helmholtz free energy is h_{FE} = u - Ts. No specific function is provided in **ASTEM97** for this calculation. The computed internal energy (u) and entropy (s) obtained at the user (p,T) state point are used directly in the high-level **ASTEM97** routines.

Isothermal Joule-Thomson Coefficient (μ_T)

The isothermal Joule-Thomson coefficient (μ_T) is defined as:

$$\left(\frac{\partial h}{\partial p}\right)_T = v - T\left(\frac{\partial v}{\partial T}\right)_p$$

No specific function is provided in **ASTEM97** for this calculation. The computed specific volume (v) and the partial derivative $(\partial v/\partial T)|_p$ obtained at the user (p,T) state point are used directly in the high-level **ASTEM97** routines.

Joule-Thomson Coefficient (μ_{JT})

The Joule-Thomson coefficient (μ_{JT}) is defined as:

$$\left(\frac{\partial T}{\partial p}\right)_h = -\frac{1}{C_p}\cdot\left(\frac{\partial h}{\partial p}\right)_T$$

No specific function is provided in **ASTEM97** for this calculation. The computed specific heat (C_p) and the isothermal Joule-Thomson coefficient (μ_T) obtained at the user (p,T) state point are used directly in the high-level **ASTEM97** routines.

Notes on the Joule-Thomson Coefficients

The method used to compute the numerical value of the Joule-Thomson coefficients and subsequently report the value in English units needs to consider the units of each thermodynamic property in the correlation.

Consider first SI units:

(1) Isothermal Joule-Thomson coefficient

$$\left(\frac{\partial h}{\partial p}\right)_T = v - T\left(\frac{\partial v}{\partial T}\right)_p \quad \text{in} \quad \frac{J}{kg - Pa}$$

Consider SI units:

$$h = \frac{J}{kg}\ ,\quad J = \frac{kg - m^2}{s^2}\ ,\quad Pa = \frac{kg}{m - s^2}$$

$$v = \frac{m^3}{kg}\ ,\quad T = K\ ,\quad \left(\frac{\partial v}{\partial T}\right)_p = \frac{m^3}{kg - K}$$

Therefore:

$$\left(\frac{J}{kg}\cdot\frac{1}{Pa}\right) = \left[\frac{\dfrac{kg - m^2}{s^2}}{kg}\right]\cdot\left[\frac{1}{\dfrac{kg}{m - s^2}}\right] = \frac{m^3}{kg}$$

In **ASTEM97**, the isothermal Joule-Thompson coefficient is computer directly from the SI properties and not from the dimensionless values, although this could have been done. The isothermal Joule-Thomson coefficient is presented in kJ/(kg-Pa) since C_p is presented in kJ/kg-K.

(2) Joule-Thomson Coefficient

$$\left(\frac{\partial T}{\partial p}\right)_h = -\frac{1}{C_p}\cdot\left(\frac{\partial h}{\partial p}\right)_T \quad \text{in } \frac{K}{Pa}$$

Now consider English units:

(1) Isothermal Joule-Thompson coefficient

$$\left(\frac{\partial h}{\partial p}\right)_T = v - T\left(\frac{\partial v}{\partial T}\right)_p \quad \text{in } \frac{BTU}{lbm\text{-}psi}$$

Consider English units:

$$h=\frac{BTU}{lbm}\ ,\ \frac{BTU}{lbm}=778\frac{ft\text{-}lbf}{lbm}\ ,\ psi=\frac{lbf}{in^2}$$

$$v=\frac{ft^3}{lbm}\ ,\ T={}^\circ R\ ,\ \left(\frac{\partial v}{\partial T}\right)_p=\frac{ft^3}{lbm-{}^\circ R}$$

Therefore,

$$\left(\frac{BTU}{lbm}\cdot\frac{in^2}{lbf}\right)=778\frac{ft-lbf}{lbm}\cdot\left[\frac{in^2}{lbf}\right]\cdot\left[\frac{1\cdot ft^2}{144\cdot in^2}\right]=\frac{ft^3}{lbm}$$

The isothermal Joule-Thomson coefficient calculation using English property values needs to be multiplied by 144/778 to account for the lbm-lbf conversion. In **ASTEM97**, the English value is obtained by direct unit conversion from the SI computed value.

(2) Joule-Thomson Coefficient

$$\left(\frac{\partial T}{\partial p}\right)_h = -\frac{1}{C_p}\cdot\left(\frac{\partial h}{\partial p}\right)_T \quad \text{in } \frac{{}^\circ F}{psi}$$

Example and Consistency Check

State point: 10^7 Pa and 400 °C (673.15 K) - IF97 Region 2

Results (partial listing) from **ASTEM97**

```
                   Pressure    1.000000000E+07 Pa         1.450377439E+03 psia
                Temperature    6.731500000E+02 K          7.520000000E+02 deg F
                                                          (1.211670000E+03 deg R)
            Specific Volume    2.643930874E-02 m^3/kg     4.235170095E-01 ft^3/lbm
    Specific Heat At P=const   3.095812323E+00 kJ/kg-K    7.394220702E-01 BTU/lbm-F
             dv/dt at p=const  7.167298442E-05 m^3/kg-K   6.378282411E-04 ft^3/lbm-F

 Isothermal Joule-Thom Coef   -2.180736072E-05 kJ/kg-Pa  -6.464163928E-02 BTU/lbm-psi
          Joule-Thomson Coef   7.044148172E-06 K/Pa       8.742184185E-02 F/psi
```

(1) Using the dimensionless formulation:

Pseudo code Example 10 - Use of DERV97 for dimensionless values

---

```
Obtain dimensionless partial derivatives from function DERV97
CALL GIBB2( 1.0E7 , 673.15 )
PIE_P   = DERV97( 9) Result  10.0000000000000
TAU_T   = DERV97(10) Result   0.802198618435713
G_PT_R  = DERV97(-6) Result  -8.750103207748675E-02
G_TT_0  = DERV97( 5) Result  -6.94751653312760
G_TT_R  = DERV97(-5) Result  -3.47600897024461
```

$\pi = 10.0$, $\tau = 0.8022$, $\gamma_{\pi\tau}^{r} = -8.75\text{x}10^{-2}$, $\gamma_{\tau\tau}^{o} = -6.9475$, $\gamma_{\tau\tau}^{r} = -3.476$

R_{gas}(SI) = 0.461526 kJ/(kg-K) , R_{gas}(English) = 0.1102336 BTU/(lbm-°R)

$$\left(\frac{\partial h}{\partial T}\right)_p = \frac{R_{gas} \cdot T}{p} \cdot \pi \cdot \tau \cdot \gamma_{\pi\tau}^{r}$$

$$\left(\frac{\partial T}{\partial p}\right)_h = \frac{T}{p} \cdot \frac{\pi \cdot \gamma_{\pi\tau}^{r}}{\tau \cdot \left(\gamma_{\tau\tau}^{o} + \gamma_{\tau\tau}^{r}\right)}$$

Substitute values:

$$\left(\frac{\partial h}{\partial T}\right)_p = -2.1807\text{x}10^{-5} \frac{\text{kJ}}{\text{kg} - \text{Pa}} \quad ; \quad -6.464\text{x}10^{-2} \frac{\text{BTU}}{\text{lbm} - \text{psi}}$$

$$\left(\frac{\partial T}{\partial p}\right)_h = 7.044\text{x}10^{-6} \frac{\text{K}}{\text{Pa}} \; ; \; 8.742\text{x}10^{-2} \frac{°\text{F}}{\text{psi}}$$

(2) Using property values:

$$\left(\frac{\partial h}{\partial p}\right)_T = v - T\cdot\left(\frac{\partial v}{\partial T}\right)_p$$

$$\left(\frac{\partial T}{\partial p}\right)_h = \frac{1}{C_p}\cdot\left(\frac{\partial h}{\partial p}\right)_T$$

Substitute values:

$$\left(\frac{\partial h}{\partial p}\right)_T = -2.1807^{-5}\,\frac{kJ}{kg-Pa}\ ;\ \frac{144}{778}\cdot(-.34932) = -6.464^{-2}\,\frac{BTU}{lbm-psi}$$

$$\left(\frac{\partial T}{\partial p}\right)_h = 7.044x10^{-6}\,\frac{K}{Pa}\ ;\ 8.742x10^{-2}\,\frac{{}^\circ F}{psi}$$

(3) Direct conversion:

1 °F = K/1.8 , 1 psi = 6.894757x10^3 Pa , 1 BTU/lbm = 2.326 kJ/kg

$$\left(\frac{\partial h}{\partial p}\right)_T \text{ in } \frac{BTU}{lbm-psi} = \frac{6.894757x10^3}{2.326}\,\frac{\frac{Pa}{psi}}{\left(\frac{kJ}{kg}\right)\div\left(\frac{BTU}{lbm}\right)} \text{ in } \frac{kJ}{kg-Pa}$$

$$\left(\frac{\partial T}{\partial p}\right)_h \text{ in } \frac{{}^\circ F}{psi} = \left(6.894757x10^3\,\frac{Pa}{psi}\right)\cdot\left(1.8\,\frac{{}^\circ F}{K}\right) \text{ in } \frac{K}{Pa}$$

$$\left(\frac{\partial h}{\partial p}\right)_{T,English} = 2{,}964.21\cdot\left(\frac{\partial h}{\partial p}\right)_{T,SI}$$

$$\left(\frac{\partial T}{\partial p}\right)_{h,English} = 12{,}410.56\cdot\left(\frac{\partial T}{\partial p}\right)_{h,SI}$$

Table 9 - Summary of Additional Properties

Parameter	**ASTEM97** Correlation	
Kinematic viscosity (υ)	$\upsilon = \frac{\eta}{\rho}$, or $\upsilon = v \cdot \eta$	where η is the dynamic viscosity
Coefficient of thermal expansion (β)	$\beta = \frac{1}{v} \cdot \left(\frac{\partial v}{\partial T}\right)_p$	
Isothermal compressibility (α)	$\alpha = -\frac{1}{v} \cdot \left(\frac{\partial v}{\partial p}\right)_T$	
Compressibility factor is (Z)[1]	$Z = \frac{p \cdot v}{T} \cdot \left(\frac{M_w \cdot 1.0x10^{-3}}{R_M}\right)$	where M_w = molar mass R_M = gas constant
Isentropic exponent (κ)	$\kappa = \frac{w^2}{p \cdot v}$	where w = sonic velocity
Prandtl number (Pr)	$Pr = \frac{1.0x10^{-3} \cdot \eta \cdot C_p}{k_{IND}}$	
Gibbs free energy (g_{FE})	h - Ts	
Helmholtz free energy (h_{FE})	u - Ts	
Isothermal Joule-Thomson coefficient (μ_T)	$\left(\frac{\partial h}{\partial p}\right)_T = v - T\left(\frac{\partial v}{\partial T}\right)_p$	
Joule-Thomson coefficient (μ_{JT})	$\left(\frac{\partial T}{\partial p}\right)_h = -\frac{1}{C_p} \cdot \left(\frac{\partial h}{\partial p}\right)_T$	

Note 1: R_M = ZUNIT97(11), M_w = ZUNIT97(12)

An example of using mid-level and low-level routines to calculate additional properties is provided below (for the isothermal compressibility):

```
Pseudo code Example 11 - Calculating Additional Properties
```

```
Low-level Example {Region 1}

 P = 10.E6   { PA }
 T = 350     {  K }
{CALCULATE ISOTHERMAL COMPRESSIBILITY}
 V  = VPT1(P,T)   { 0.001022 M^3/KG}
 DV = DVDPT1(P,T) {-4.54040E-13 M^3/KG-PA}
 C  = -DV/V       {4.44092E-10 1/PA}
```

```
Mid-Level Example {'Unknown' region}

 P = 26.5E6 { PA }
 T = 850.   {  K }

 V  = VPT97(P,T)   {0.012613    }
 DV = DVDPT97(P,T) {-5.62144E-10}
 C  = -DV/V        {4.45683E-08 }
 {ACTUAL REGION IS 2}
```

Notes on Units

Fundamental Units

Quantity	Name of Unit	Abbreviation
length	meter	m
mass	kilogram	kg
time	second	sec

Derived Units

Quantity	Name of Unit	Abbreviation	SI Units
force	newton	N	kg-m/sec^2
pressure	pascal	Pa	N/m^2 = kg/m-sec^2
energy	joule	J	kg-m^2/sec^2
power	watt	W	J/sec = kg-m^2/sec^3

Validation results from **ASTEM97** are provided in Appendix F. Data is provided for comparison to the verification values reported in the various references for the basic thermodynamic properties, the backward equations and the various additional properties.

References

1999a Kretzschmar, H.-J.; Stöcker, I.; Knobloch, K.; Willkommen, Th.; Trübenbach, J.; Dittmann A.: Supplementary Backward Equations p(h,s) and Equations v(p,T) for the Critical Region to the New Industrial Formulation IAPWS-IF97 for Water and Steam, 13th International Conference on the Properties of Water and Steam, Toronto 1999.

1999b Kretzschmar, H.-J.; Stöcker, I.; Klinger, J.; Dittmann A.: Calculation of Thermodynamic Derivatives for Water and Steam Using the New Industrial Formulation IAPWS-IF97, 13th International Conference on the Properties of Water and Steam, Toronto 1999.

1998a Kretzschmar,H.-J.; Zschunke,T.; Klinger,J.; Dittmann,A.: An Alternative Method for the Numerical Calculation of the Maxwell Criterion in Vapour Pressure Computations, 11th International Conference on the Properties of Water and Steam, Prag 1989 .

IAPWS Releases and Guidelines

IAPWSa Release on the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam (September 1997)

IAPWSb Release on the IAPS Formulation 1985 for the Thermal Conductivity of Ordinary Water Substance (September 1998)

IAPWSc1 Release on the IAPS Formulation 1985 for the Viscosity of Ordinary Water Substance (September 1997)

IAPWSc2 Revised Release on the IAPS Formulation 1985 for the Viscosity of Ordinary Water Substance (August 2003)

IAPWSd Release on the Refractive Index of Ordinary Water Substance as a Function of Wavelength, Temperature and Pressure (September 1997)

IAPWSe Release on the Static Dielectric Constant of Ordinary Water Substance for Temperatures from 238 K to 873 K and Pressures up to 1000 MPa

IAPWSf IAPWS Release on Surface Tension of Ordinary Water Substance (September 1994)

IAPWSg Supplementary Release on Backward Equations for Pressure as a Function of Enthalpy and Entropy p(h,s) to the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam (September 2001)

IAPWSh September 2004, Revised Supplementary Release on Backward Equations for the Functions *T*(*p*,*h*), *v*(*p*,*h*) and *T*(*p*,*s*), *v*(*p*,*s*) for Region 3 of the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam

IAPWSi September 2004, Supplemental Release on Backward Equations *p(h,s)* for Region 3, Equations as a function of *h* and *s* for the Region Boundaries, and an Equation $T_{sat}(h,s)$ for Region 4 of the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam.

IAPWSj July 2005, Supplementary Release on Backward Equations for Specific Volume as a Function of Pressure and Temperature *v(p,T)* for Region 3 of the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam

Wagner Wagner, W., Kruse, A., Universität Bochum,"Properties of Water and Steam (Zustandsgrößen von Wasser und Wasserdampf) The Industrial Standard IAPWS-IF97 for the Thermodynamic Properties and Supplemetary Equations for Other Properties,"(Der Industrie-Standard IAPWS-IF97 für die thermodynamischen Zustandsgrößen und ergänzende Gleichungen für andere Eigenschaften) 1998, ISBN 3-540-64339-7.

ASME ASME Steam Properties for Industrial Use Based on IAPWS-IF97, Professional Version, ASME Press,1999, ISBN 0791819566.