User's Guide

for

LibHuAirProp

Library for the Calculation of Psychrometric, Thermodynamic, and Transport Properties for *Real* Humid Air, Steam, Water, and Ice

I-P & SI Units

Version 10.0

FluidEES for EES

Add-On for the comfortable use of LibHuAirProp in Engineering Equation Solver

Based on ASHRAE Research Projects
RP-1485 and RP-1767

Prepared by
THERMO
FLUID
PROPERTIES
www.thermofluidprop.com

Sebastian Herrmann
Hans-Joachim Kretzschmar
Matthias Kunick
Martin Suender

DISCLAIMER

ASHRAE has compiled this publication with care, but ASHRAE has not investigated, and ASHRAE expressly disclaims any duty to investigate, any product, service, process, procedure, design, or the like that may be described herein. The appearance of any technical data or editorial material in this publication does not constitute endorsement, warranty, or guaranty by ASHRAE of any product, service, process, procedure, design, or the like. ASHRAE does not warrant that the information in the publication is free of errors, and ASHRAE does not necessarily agree with any statement or opinion in this publication. The entire risk of the use of any information in this publication is assumed by the user.

The program developer, ASHRAE, and the distributors of this product do not warrant that the information in this software is free of errors. This information is provided “as is” without warranty of any kind, either expressed or implied. The entire risk as to the quality and performance of the program and data is with you. In no event will ASHRAE or the program developer be liable to you for any damages, including without limitation any lost profits, lost savings, or other incidental or consequential damages arising out of the use of or inability to use this information. ASHRAE reserves the right to discontinue support for this product at any time in the future, including without limitation discontinuing support of servers facilitating access to this product. In the event of discontinuation of support, ASHRAE will make a reasonable effort to notify purchasers before support is discontinued.

LICENSING AGREEMENT

Using this product indicates your acceptance of the terms and conditions of this agreement. The title and all copyrights and ownership rights of the product are retained by ASHRAE. You assume responsibility for the selection of the product to achieve your results and for the installation, use, and results obtained from the product.

This product may not be used on a LAN or WAN. You may use the product on a single machine. You may also copy the product into any machine-readable form for backup purposes in support of your use of the product on a single machine. You may not copy or transfer the product except as expressly provided for in this license. Specifically, you may not copy or transfer the product onto a machine other than your own unless the person to whom you are copying or transferring the product also has a license to use it. Doing so will result in the termination of your license, and ASHRAE will consider options to recover damages from unauthorized use of its intellectual property. Distribution to third parties in print or in electronic form is expressly prohibited unless authorized in writing by ASHRAE.

The program and data contained in the product are for your personal use only. “Personal use” includes showing the information at a meeting or group setting and allowing other individuals to view the content. “Personal use” does not include making copies, in whole or in part, of the content for the purposes of distribution, reusing the information contained in the product in your own presentation, or posting any of the files on a server for access by others. You shall not merge, adapt, translate, modify, rent, lease, sell, sublicense, assign, or otherwise transfer any of the content. To obtain permission to copy and paste this publication’s content for other than only personal use, go to www.ashrae.org/permissions.

ISBN 978-1-964173-43-6 (32-bit)

ISBN 978-1-964173-44-3 (64-bit)

LibHuAirProp Product Information

Do you need property values for moist air in I-P or SI units in your daily work?

► Use the property library LibHuAirProp ◄

Do you need these properties in Excel®, MATLAB®, Mathcad®, Mathcad Prime®, Engineering Equation Solver®, LabVIEW™, DYMOLA®, or SimulationX®?

► Use the add-ins FluidEXL, FluidLAB, FluidMAT, FluidPRIME, FluidEES, FluidVIEW, or FluidDYM ◄

What properties can be calculated using this software?

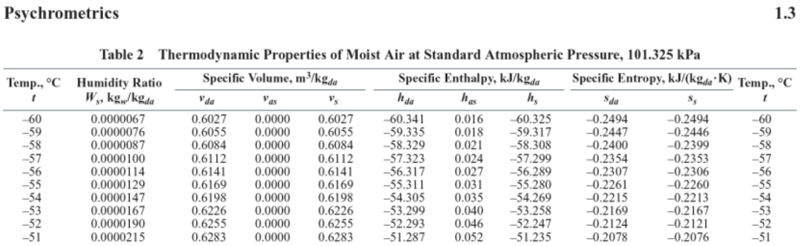
► thermodynamic properties psychrometric functions ◄
► transport properties backward functions ◄

What range of state is covered by this property library?

► unsaturated and saturated moist air ◄
► supersaturated moist air (liquid fog and ice fog) ◄
► temperatures from −143.15°C (−225.67°F) to 350°C (662°F) ◄
► pressures from 0.01 kPa (0.00145 psi) to 10,000 kPa (1450.4 psi) ◄

What are the references of LibHuAirProp?

Tables for moist air properties in the 2009, 2013, 2017, 2021, and 2025 *ASHRAE Handbook of Fundamentals* were calculated using LibHuAirProp

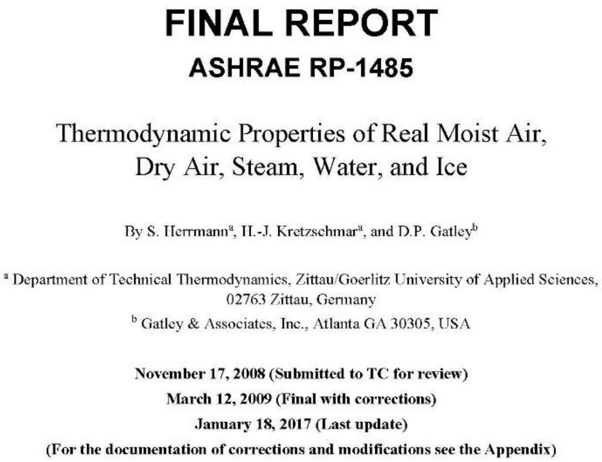


Psychrometrics 1.3

Table 2 Thermodynamic Properties of Moist Air at Standard Atmospheric Pressure, 101.325 kPa

Temp., °C	Humidity Ratio	Specific Volume, m^3/kg_{da}			Specific Enthalpy, kJ/kg_{da}			Specific Entropy, $kJ/(kg_{da} \cdot K)$		Temp., °C
t	W_s, kg_w/kg_{da}	v_{da}	v_{as}	v_s	h_{da}	h_{as}	h_s	s_{da}	s_s	t
–60	0.0000067	0.6027	0.0000	0.6027	–60.341	0.016	–60.325	–0.2494	–0.2494	–60
–59	0.0000076	0.6055	0.0000	0.6055	–59.335	0.018	–59.317	–0.2447	–0.2446	–59
–58	0.0000087	0.6084	0.0000	0.6084	–58.329	0.021	–58.308	–0.2400	–0.2399	–58
–57	0.0000100	0.6112	0.0000	0.6112	–57.323	0.024	–57.299	–0.2354	–0.2353	–57
–56	0.0000114	0.6141	0.0000	0.6141	–56.317	0.027	–56.289	–0.2307	–0.2306	–56
–55	0.0000129	0.6169	0.0000	0.6169	–55.311	0.031	–55.280	–0.2261	–0.2260	–55
–54	0.0000147	0.6198	0.0000	0.6198	–54.305	0.035	–54.269	–0.2215	–0.2213	–54
–53	0.0000167	0.6226	0.0000	0.6226	–53.299	0.040	–53.258	–0.2169	–0.2167	–53
–52	0.0000190	0.6255	0.0000	0.6255	–52.293	0.046	–52.247	–0.2124	–0.2121	–52
–51	0.0000215	0.6283	0.0000	0.6283	–51.287	0.052	–51.235	–0.2078	–0.2076	–51

Thermodynamic and psychrometric property algorithms from ASHRAE Research Project 1485



FINAL REPORT

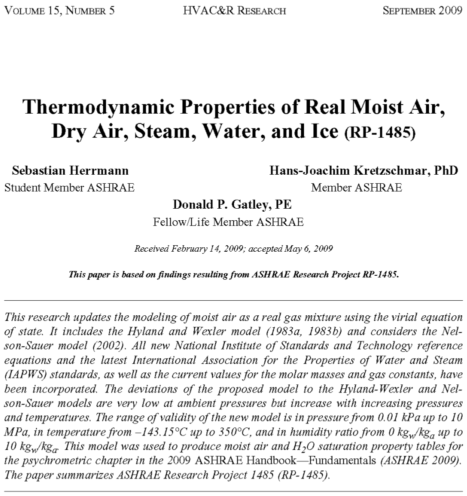
ASHRAE RP-1485

Thermodynamic Properties of Real Moist Air,
Dry Air, Steam, Water, and Ice

By S. Herrmann[a], H.-J. Kretzschmar[a], and D.P. Gatley[b]

[a] Department of Technical Thermodynamics, Zittau/Goerlitz University of Applied Sciences, 02763 Zittau, Germany
[b] Gatley & Associates, Inc., Atlanta GA 30305, USA

November 17, 2008 (Submitted to TC for review)
March 12, 2009 (Final with corrections)
January 18, 2017 (Last update)
(For the documentation of corrections and modifications see the Appendix)



VOLUME 15, NUMBER 5 HVAC&R RESEARCH SEPTEMBER 2009

Thermodynamic Properties of Real Moist Air, Dry Air, Steam, Water, and Ice (RP-1485)

Sebastian Herrmann
Student Member ASHRAE

Hans-Joachim Kretzschmar, PhD
Member ASHRAE

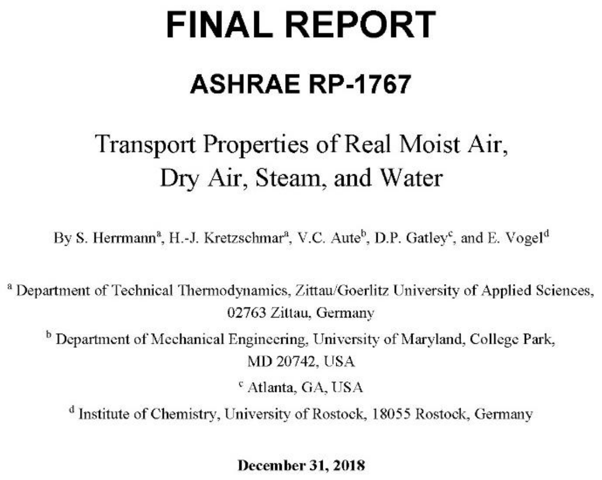
Donald P. Gatley, PE
Fellow/Life Member ASHRAE

Received February 14, 2009; accepted May 6, 2009

This paper is based on findings resulting from ASHRAE Research Project RP-1485.

This research updates the modeling of moist air as a real gas mixture using the virial equation of state. It includes the Hyland and Wexler model (1983a, 1983b) and considers the Nelson-Sauer model (2002). All new National Institute of Standards and Technology reference equations and the latest International Association for the Properties of Water and Steam (IAPWS) standards, as well as the current values for the molar masses and gas constants, have been incorporated. The deviations of the proposed model to the Hyland-Wexler and Nelson-Sauer models are very low at ambient pressures but increase with increasing pressures and temperatures. The range of validity of the new model is in pressure from 0.01 kPa up to 10 MPa, in temperature from –143.15°C up to 350°C, and in humidity ratio from 0 kg_w/kg_a *up to 10* kg_w/kg_a*. This model was used to produce moist air and* H_2O *saturation property tables for the psychrometric chapter in the 2009* ASHRAE Handbook—Fundamentals *(ASHRAE 2009). The paper summarizes ASHRAE Research Project 1485 (RP-1485).*

Transport property algorithms of moist air from ASHRAE Research Project 1767



FINAL REPORT

ASHRAE RP-1767

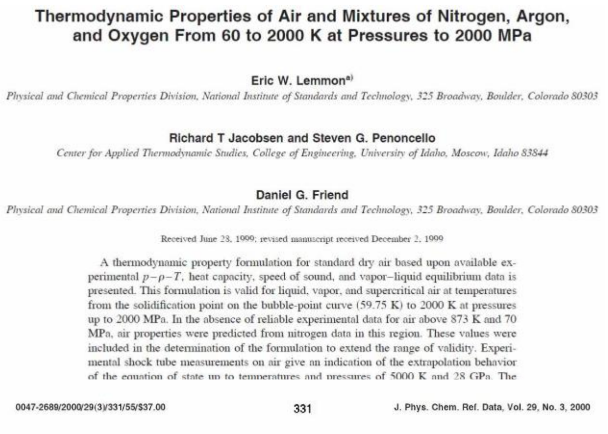
Transport Properties of Real Moist Air,
Dry Air, Steam, and Water

By S. Herrmann[a], H.-J. Kretzschmar[a], V.C. Aute[b], D.P. Gatley[c], and E. Vogel[d]

[a] Department of Technical Thermodynamics, Zittau/Goerlitz University of Applied Sciences,
02763 Zittau, Germany
[b] Department of Mechanical Engineering, University of Maryland, College Park,
MD 20742, USA
[c] Atlanta, GA, USA
[d] Institute of Chemistry, University of Rostock, 18055 Rostock, Germany

December 31, 2018

Properties of dry air from the NIST Reference Equation of *Lemmon et al.* and properties of steam, water, and ice from the Industrial Formulation IAPWS-IF97, the Scientific Formulation IAPWS-95, and other current IAPWS formulations



Thermodynamic Properties of Air and Mixtures of Nitrogen, Argon, and Oxygen From 60 to 2000 K at Pressures to 2000 MPa

Eric W. Lemmon[a)]
Physical and Chemical Properties Division, National Institute of Standards and Technology, 325 Broadway, Boulder, Colorado 80303

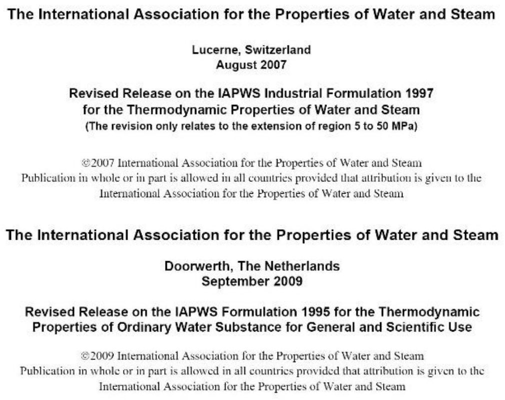
Richard T Jacobsen and Steven G. Penoncello
Center for Applied Thermodynamic Studies, College of Engineering, University of Idaho, Moscow, Idaho 83844

Daniel G. Friend
Physical and Chemical Properties Division, National Institute of Standards and Technology, 325 Broadway, Boulder, Colorado 80303

Received June 28, 1999; revised manuscript received December 2, 1999

A thermodynamic property formulation for standard dry air based upon available experimental $p-\rho-T$, heat capacity, speed of sound, and vapor–liquid equilibrium data is presented. This formulation is valid for liquid, vapor, and supercritical air at temperatures from the solidification point on the bubble-point curve (59.75 K) to 2000 K at pressures up to 2000 MPa. In the absence of reliable experimental data for air above 873 K and 70 MPa, air properties were predicted from nitrogen data in this region. These values were included in the determination of the formulation to extend the range of validity. Experimental shock tube measurements on air give an indication of the extrapolation behavior of the equation of state up to temperatures and pressures of 5000 K and 28 GPa. The

0047-2689/2000/29(3)/331/55/$37.00 331 J. Phys. Chem. Ref. Data, Vol. 29, No. 3, 2000



The International Association for the Properties of Water and Steam

Lucerne, Switzerland
August 2007

Revised Release on the IAPWS Industrial Formulation 1997
for the Thermodynamic Properties of Water and Steam
(The revision only relates to the extension of region 5 to 50 MPa)

©2007 International Association for the Properties of Water and Steam
Publication in whole or in part is allowed in all countries provided that attribution is given to the
International Association for the Properties of Water and Steam

The International Association for the Properties of Water and Steam

Doorwerth, The Netherlands
September 2009

Revised Release on the IAPWS Formulation 1995 for the Thermodynamic
Properties of Ordinary Water Substance for General and Scientific Use

©2009 International Association for the Properties of Water and Steam
Publication in whole or in part is allowed in all countries provided that attribution is given to the
International Association for the Properties of Water and Steam

Who are the authors of LibHuAirProp?

Dr. Sebastian Herrmann

Dr. Hans-Joachim Kretzschmar
Professor for Technical Thermodynamics

Dr. Matthias Kunick
Professor for Energy Systems Technology

Martin Suender

Kretzschmar Consulting Engineers, Amberg, Germany

Zittau/Goerlitz University of Applied Sciences, Zittau, Germany

Property Library for *Real* Humid Air, Steam, Water, and Ice

ASHRAE LibHuAirProp

Contents

© KCE-ThermoFluidProperties UG
Haager Weg 6
92224 Amberg, Germany
Phone: +49-172-7914607
Fax: +49-3222-1095810
E-mail: info@thermofluidprop.com
Internet: www.thermofluidprop.com

Responsible person:

Dr. Sebastian Herrmann
Phone: +49-172-5619222
E-mail: Herrms@web.de

0 Package Contents

0.1 Add-On for 32-bit version of Engineering Equation Solver®

The following ZIP file is delivered for your computer running a 32-bit version of Engineering Equation Solver®.

ZIP file "CD_FluidEES_ASHRAE_LibHuAirProp.zip" for Engineering Equation Solver®

The ZIP file contains the following files:

FluidEES_LibHuAirProp_Setup.exe	Installation program for the FluidEES Add-On for use in Engineering Equation Solver®
FluidEES_ASHRAE_LibHuAirProp_Users_Guide.pdf	User's Guide

0.2 Add-On for 64-bit version of Engineering Equation Solver®

The following ZIP file is delivered for your computer running a 64-bit version of Engineering Equation Solver®.

ZIP file "CD_FluidEES_ASHRAE_LibHuAirProp_x64.zip" for Engineering Equation Solver®

The ZIP file contains the following files:

FluidEES_ASHRAE_LibHuAirProp_Users_Guide.pdf	User's Guide
FluidEES_ASHRAE_LibHuAirProp_64_Setup.msi	Self-extracting and self-installing program
setup.exe	Installation program for the FluidEES Add-On for use in Engineering Equation Solver®
vcredist_x64	Folder containing the "Microsoft Visual C++ 2010 x64 Redistributable Pack"
WindowsInstaller3_1	Folder containing the "Microsoft Windows Installer"

Part I-P Units

1 Property Library ASHRAE-LibHuAirProp-IP

1.1 Function Overview

1.1.1 Function Overview for Real Moist Air

Functional Dependence	Function Name	Property or Function	Unit of the Result	Page
$a = \mathrm{f}(p,t,W)$	a_ptW_HAP_IP	Thermal diffusivity	ft²/s	3/2
$\alpha_\mathrm{p} = \mathrm{f}(p,t,W)$	alphap_ptW_HAP_IP	Relative pressure coefficient	1/°R	3/3
$\beta_\mathrm{p} = \mathrm{f}(p,t,W)$	betap_ptW_HAP_IP	Isothermal stress coefficient	lb/ft³	3/4
$c = \mathrm{f}(p,t,W)$	c_ptW_HAP_IP	Speed of sound	ft/s	3/5
$c_p = \mathrm{f}(p,t,W)$	cp_ptW_HAP_IP	Specific isobaric heat capacity	Btu/(lb·°R)	3/6
$c_v = \mathrm{f}(p,t,W)$	cv_ptW_HAP_IP	Specific isochoric heat capacity	Btu/(lb·°R)	3/7
$f = \mathrm{f}(p,t)$	f_pt_HAP_IP	Enhancement factor (decimal ratio)	-	3/8
$h = \mathrm{f}(p,t,W)$	h_ptW_HAP_IP	Air-specific enthalpy	$\mathrm{Btu/lb_a}$	3/9
$\eta = \mathrm{f}(p,t,W)$	Eta_ptW_HAP_IP	Dynamic viscosity	lb·s/ft²	3/10
$\kappa = \mathrm{f}(p,t,W)$	Kappa_ptW_HAP_IP	Isentropic exponent	-	3/11
$\lambda = \mathrm{f}(p,t,W)$	Lambda_ptW_HAP_IP	Thermal conductivity	Btu/(h·ft·°R)	3/12
$\nu = \mathrm{f}(p,t,W)$	Ny_ptW_HAP_IP	Kinematic viscosity	ft²/s	3/13
$p = \mathrm{f}(t,s,W)$	p_tsW_HAP_IP	Pressure of humid air	psi	3/14
$p = \mathrm{f}(z_\mathrm{ele})$	p_zele_HAP_IP	Pressure of humid air from elevation	psi	3/15
$p_\mathrm{Air} = \mathrm{f}(p,t,W)$	pAIR_ptW_HAP_IP	Partial pressure of dry air in moist air	psi	3/16
$p_\mathrm{H2O} = \mathrm{f}(p,t,W)$	pH2O_ptW_HAP_IP	Partial pressure of water vapor in moist air	psi	3/17
$p_\mathrm{H2Os} = \mathrm{f}(p,t)$	pH2Os_pt_HAP_IP	Partial saturation pressure of water vapour in moist air	psi	3/18

Functional Dependence	**Function Name**	**Property or Function**	**Unit of the Result**	**Page**
$\varphi = \mathrm{f}(p,t,W)$	phi_ptW_HAP_IP	Relative humidity (decimal ratio)	-	3/19
$Pr = \mathrm{f}(p,t,W)$	Pr_ptW_HAP_IP	PRANDTL number	-	3/20
$\psi_{\mathrm{Air}} = \mathrm{f}(W)$	PsiAir_W_HAP_IP	Mole fraction of dry air in moist air	$\mathrm{mol_a/mol}$	3/21
$\psi_{\mathrm{H2O}} = \mathrm{f}(W)$	PsiH2O_W_HAP_IP	Mole fraction of water vapor in moist air	$\mathrm{mol_w/mol}$	3/22
$\rho = \mathrm{f}(p,t,W)$	Rho_ptW_HAP_IP	Density	$\mathrm{lb/ft^3}$	3/23
$s = \mathrm{f}(p,t,W)$	s_ptW_HAP_IP	Air-specific entropy	$\mathrm{Btu/(lb_a \cdot °R)}$	3/24
$t = \mathrm{f}(p,h,\varphi)$	t_phphi_HAP_IP	Backward function: temperature from total pressure, air-specific enthalpy and relative humidity	°F	3/25
$t = \mathrm{f}(p,h,W)$	t_phW_HAP_IP	Backward function: temperature from total pressure, enthalpy and humidity ratio	°F	3/26
$t = \mathrm{f}(p,s,W)$	t_psW_HAP_IP	Backward function: temperature from total pressure, entropy and humidity ratio	°F	3/27
$t = \mathrm{f}(p,t_{\mathrm{wb}},W)$	t_ptwbW_HAP_IP	Backward function: temperature from total pressure, wet-bulb temperature and humidity ratio	°F	3/28
$t_{\mathrm{d}} = \mathrm{f}(p,W)$	td_pW_HAP_IP	Dew-point/frost-point temperature	°F	3/29
$t_{\mathrm{s}} = \mathrm{f}(p,p_{\mathrm{H2O}})$	ts_ppH2O_HAP_IP	Backward function: saturation temperature of water from total pressure and partial pressure of water vapor	°F	3/30
$t_{\mathrm{wb}} = \mathrm{f}(p,t,W)$	twb_ptW_HAP_IP	Wet-bulb/ice-bulb temperature	°F	3/31
$u = \mathrm{f}(p,t,W)$	u_ptW_HAP_IP	Air-specific internal energy	$\mathrm{Btu/lb_a}$	3/32
$v = \mathrm{f}(p,t,W)$	v_ptW_HAP_IP	Air-specific volume	$\mathrm{ft^3/lb_a}$	3/33
$W = \mathrm{f}(p,t,p_{\mathrm{H2O}})$	W_ptpH2O_HAP_IP	Humidity ratio from total pressure, temperature, and partial pressure of water vapor	$\mathrm{lb_w/lb_a}$	3/34
$W = \mathrm{f}(p,t,\varphi)$	W_ptphi_HAP_IP	Humidity ratio from total pressure, temperature, and relative humidity	$\mathrm{lb_w/lb_a}$	3/35
$W = \mathrm{f}(p,t_{\mathrm{d}})$	W_ptd_HAP_IP	Humidity ratio from total pressure and dew-point temperature	$\mathrm{lb_w/lb_a}$	3/36

Functional Dependence	**Function Name**	**Property or Function**	**Unit of the Result**	**Page**
$W = \mathrm{f}(p,t,t_{\mathrm{wb}})$	W_pttwb_HAP_IP	Humidity ratio from total pressure, (dry bulb) temperature, and wet-bulb temperature	$\mathrm{lb_w/lb_a}$	3/37
$W_{\mathrm{s}} = \mathrm{f}(p,t)$	Ws_pt_HAP_IP	Saturation humidity ratio	$\mathrm{lb_w/lb_a}$	3/38
$\xi_{\mathrm{Air}} = \mathrm{f}(W)$	XiAir_W_HAP_IP	Mass fraction of dry air in moist air	$\mathrm{lb_a/lb}$	3/39
$\xi_{\mathrm{H2O}} = \mathrm{f}(W)$	XiH2O_W_HAP_IP	Mass fraction of water vapor in moist air	$\mathrm{lb_w/lb}$	3/40
$Z = \mathrm{f}(p,t,W)$	Z_ptW_HAP_IP	Compression factor (decimal ratio)	-	3/41

Range of Validity of Thermodynamic Properties

Property	Range of Validity					
Pressure:	0.00145	≤	p	≤	1450.4	psi
Temperature:	–225.67	≤	t	≤	662	°F
Humidity ratio:	0	≤	W	≤	10	lb_w/lb_a
Relative humidity:	0	≤	φ	≤	1	(decimal ratio)
Dew-point temperature:	–225.67	≤	t_d	≤	662	°F
Wet-bulb temperature:	–225.67	≤	t_{wb}	≤	662	°F

Units

Symbol	Quantity	Unit
p	Pressure	psi
t	Temperature	°F
W	Humidity ratio	lb_w/lb_a (lb water / lb dry air)
φ	Relative humidity	(decimal ratio)
t_d	Dew point temperature	°F
t_{wb}	Wet bulb temperature	°F

Range of Validity of Transport Properties

Property	Range of Validity					
Pressure:	0.00145	≤	p	≤	1450.4	psi
Temperature:	–99.67	≤	t	≤	662	°F
Humidity ratio:	0	≤	W	≤	10	lb_w/lb_a
Relative humidity:	0	≤	φ	≤	1	(decimal ratio)

Molar Masses

Component	Molar Mass	Reference
Dry Air	63.859 lb/kmol	[17]
Water	39.7168998 lb/kmol	[5], [6]

Reference States

Property	Dry Air	Steam, Water, and Ice
Pressure	14.6959 psi	p_s(32.018°F) = 0.088714 psi
Temperature	32°F	32.018°F
Enthalpy	0 Btu/lb	0.00026301926 Btu/lb
Entropy	0 Btu/(lb·°R)	0 Btu/(lb·°R)

1.1.2 Function Overview for Steam and Water for Temperatures $t \geq 32$°F

Functional Dependence	Function Name	Property or Function	Unit of the Result	Page
$h_{liq} = f(p,t)$	hliq_pt_97_IP	Specific enthalpy of liquid water	Btu/lb	3/43
$h_{liq,s} = f(t)$	hliqs_t_97_IP	Specific enthalpy of saturated liquid water	Btu/lb	3/44
$h_{vap,s} = f(t)$	hvaps_t_97_IP	Specific enthalpy of saturated water vapor	Btu/lb	3/45
$p_s = f(t)$	ps_t_97_IP	Saturation pressure of water	psi	3/46
$s_{liq} = f(p,t)$	sliq_pt_97_IP	Specific entropy of liquid water	Btu/(lb·°R)	3/47
$s_{liq,s} = f(t)$	sliqs_t_97_IP	Specific entropy of saturated liquid water	Btu/(lb·°R)	3/48
$s_{vap,s} = f(t)$	svaps_t_97_IP	Specific entropy of saturated water vapor	Btu/(lb·°R)	3/49
$t_s = f(p)$	ts_p_97_IP	Saturation temperature of water	°F	3/50
$v_{liq} = f(p,t)$	vliq_pt_97_IP	Specific volume of liquid water	ft³/lb	3/51
$v_{liq,s} = f(t)$	vliqs_t_97_IP	Specific volume of saturated liquid water	ft³/lb	3/52
$v_{vap,s} = f(t)$	vvaps_t_97_IP	Specific volume of saturated water vapor	ft³/lb	3/53

Range of Validity

Property	Range of Validity					
Pressure:	0.00145	≤	*p*	≤	1450.4	psi
Temperature:	32	≤	*t*	≤	662	°F

Units

Symbol	Quantity	Unit
p	Pressure	psi
t	Temperature	°F

Reference State

Property	Water Vapor and Liquid Water
Pressure	p_s(32.018°F) = 0.088714 psi
Temperature	32.018°F
Enthalpy	0.00026301926 Btu/lb
Entropy	0 Btu/(lb·°R)

1.1.3 Function Overview for Steam and Ice for Temperatures $t \leq 32°\mathrm{F}$

Functional Dependence	**Function Name**	**Property**	**Unit of the Result**	**Page**
$h_{\mathrm{ice,sub}} = \mathrm{f}(t)$	hicesub_t_06_IP	Specific enthalpy of saturated ice	Btu/lb	3/55
$h_{\mathrm{vap,sub}} = \mathrm{f}(t)$	hvapsub_t_95_IP	Specific enthalpy of saturated water vapor	Btu/lb	3/56
$p_{\mathrm{mel}} = \mathrm{f}(t)$	pmel_t_08_IP	Melting pressure of ice	psi	3/57
$p_{\mathrm{sub}} = \mathrm{f}(t)$	psub_t_08_IP	Sublimation pressure of ice	psi	3/58
$s_{\mathrm{ice,sub}} = \mathrm{f}(t)$	sicesub_t_06_IP	Specific entropy of saturated ice	Btu/(lb·°R)	3/59
$s_{\mathrm{vap,sub}} = \mathrm{f}(t)$	svapsub_t_95_IP	Specific entropy of saturated water vapor	Btu/(lb·°R)	3/60
$t_{\mathrm{mel}} = \mathrm{f}(p)$	tmel_p_08_IP	Melting temperature of ice	°F	3/61
$t_{\mathrm{sub}} = \mathrm{f}(p)$	tsub_p_08_IP	Sublimation temperature of ice	°F	3/62
$v_{\mathrm{ice,sub}} = \mathrm{f}(t)$	vicesub_t_06_IP	Specific volume of saturated ice	$\mathrm{ft^3/lb}$	3/63
$v_{\mathrm{vap,sub}} = \mathrm{f}(t)$	vvapsub_t_95_IP	Specific volume of saturated water vapor	$\mathrm{ft^3/lb}$	3/64

Range of Validity

Property	Range of Validity					
Pressure:	p_{sub}(–225.67°F) = 1.7407E-12	≤	*p*	≤	1450.4	psi
Temperature:	–225.67	≤	*t*	≤	32	°F

Units

Symbol	Quantity	Unit
p	Pressure	psi
t	Temperature	°F

Reference State

Property	Water Vapor and Ice
Pressure	p_s(32.018°F) = 0.088714 psi
Temperature	32.018°F
Enthalpy	0.00026301926 Btu/lb
Entropy	0 Btu/(lb·°R)

1.2 Conversion of SI and I-P Units

Property		Conversion: SI Units → I-P Units	Conversion: I-P Units → SI Units	Units SI	Units I-P
Thermal diffusivity	a	$\frac{a_{\mathrm{IP}}}{\frac{\mathrm{ft^2}}{\mathrm{s}}} = \frac{a_{\mathrm{SI}}}{\frac{\mathrm{m^2}}{\mathrm{s}}} \times 10.76391042$	$\frac{a_{\mathrm{SI}}}{\frac{\mathrm{m^2}}{\mathrm{s}}} = \frac{a_{\mathrm{IP}}}{\frac{\mathrm{ft^2}}{\mathrm{s}}} \times 0.0929304$	m²/s	ft²/s
Relative pressure coefficient	α_p	$\frac{\alpha_{p,\mathrm{IP}}}{\frac{1}{^\circ\mathrm{R}}} = \frac{\alpha_{p,\mathrm{SI}}}{\frac{1}{\mathrm{K}}} \times \frac{9}{5}$	$\frac{\alpha_{p,\mathrm{SI}}}{\frac{1}{\mathrm{K}}} = \frac{\alpha_{p,\mathrm{IP}}}{\frac{1}{^\circ\mathrm{R}}} \times \frac{5}{9}$	1/K	1/°R
Isothermal stress coefficient	β_p	$\frac{\beta_{p,\mathrm{IP}}}{\frac{\mathrm{lb}}{\mathrm{ft^3}}} = \frac{\beta_{p,\mathrm{SI}}}{\frac{\mathrm{kg}}{\mathrm{m^3}}} \times 0.062428$	$\frac{\beta_{p,\mathrm{SI}}}{\frac{\mathrm{kg}}{\mathrm{m^3}}} = \frac{\beta_{p,\mathrm{IP}}}{\frac{\mathrm{lb}}{\mathrm{ft^3}}} \times 16.018463$	kg/m³	lb/ft³
Speed of sound	c	$\frac{c_{\mathrm{IP}}}{\frac{\mathrm{ft}}{\mathrm{s}}} = \frac{c_{\mathrm{SI}}}{\frac{\mathrm{m}}{\mathrm{s}}} \times 3.2808399$	$\frac{c_{\mathrm{SI}}}{\frac{\mathrm{m}}{\mathrm{s}}} = \frac{c_{\mathrm{IP}}}{\frac{\mathrm{ft}}{\mathrm{s}}} \times 0.3048$	m/s	ft/s
Specific isobaric heat capacity	c_p	$\frac{c_{p,\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}\ ^\circ\mathrm{R}}} = \frac{c_{p,\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg\ K}}} \times 0.2388459$	$\frac{c_{p,\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg\ K}}} = \frac{c_{p,\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}\ ^\circ\mathrm{R}}} \times 4.1868$	kJ/(kg·K)	Btu/(lb·°R)
Specific isochoric heat capacity	c_v	$\frac{c_{v,\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}\ ^\circ\mathrm{R}}} = \frac{c_{v,\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg\ K}}} \times 0.2388459$	$\frac{c_{v,\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg\ K}}} = \frac{c_{v,\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}\ ^\circ\mathrm{R}}} \times 4.1868$	kJ/(kg·K)	Btu/(lb·°R)
Dynamic viscosity	η	$\frac{\eta_{\mathrm{IP}}}{\frac{\mathrm{lb\ s}}{\mathrm{ft^2}}} = \frac{\eta_{\mathrm{SI}}}{\frac{\mathrm{Pa}}{\mathrm{s}}} \times 0.02088543$	$\frac{\eta_{\mathrm{SI}}}{\frac{\mathrm{Pa}}{\mathrm{s}}} = \frac{\eta_{\mathrm{IP}}}{\frac{\mathrm{lb\ s}}{\mathrm{ft^2}}} \times 47.880259$	Pa·s	lb·s/ft²
Enhancement factor	f	$f_{\mathrm{IP}} = f_{\mathrm{SI}}$	$f_{\mathrm{SI}} = f_{\mathrm{IP}}$	-	-

Property		**Conversion: SI Units → I-P Units**	**Conversion: I-P Units → SI Units**	**Units SI**	**Units I-P**
Air-specific enthalpy (moist air)	h	$\frac{h_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb_a}}} = \frac{h_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg_a}}} \times 0.4299226 + 7.68565365666$	$\frac{h_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg_a}}} = \left(\frac{h_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb_a}}} - 7.68565365666 \right) \times 2.326$	$\mathrm{kJ/kg_a}$	$\mathrm{Btu/lb_a}$
Specific enthalpy (water, water vapor, ice)	h_w	$\frac{h_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}}} = \frac{h_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg}}} \times 0.4299226$	$\frac{h_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg}}} = \frac{h_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}}} \times 2.326$	kJ/kg	Btu/lb
Isentropic exponent	κ	$\kappa_{\mathrm{IP}} = \kappa_{\mathrm{SI}}$	$\kappa_{\mathrm{SI}} = \kappa_{\mathrm{IP}}$	-	-
Thermal conductivity	λ	$\frac{\lambda_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{h\ ft\ °R}}} = \frac{\lambda_{\mathrm{SI}}}{\frac{\mathrm{W}}{\mathrm{m\ K}}} \times 0.57778932$	$\frac{\lambda_{\mathrm{SI}}}{\frac{\mathrm{W}}{\mathrm{m\ K}}} = \frac{\lambda_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{h\ ft\ °R}}} \times 1.73073467$	W/(m·K)	Btu/(h·ft·°R)
Kinematic viscosity	ν	$\frac{\nu_{\mathrm{IP}}}{\frac{\mathrm{ft^2}}{\mathrm{s}}} = \frac{\nu_{\mathrm{SI}}}{\frac{\mathrm{m^2}}{\mathrm{s}}} \times 10.763910417$	$\frac{\nu_{\mathrm{SI}}}{\frac{\mathrm{m^2}}{\mathrm{s}}} = \frac{\nu_{\mathrm{IP}}}{\frac{\mathrm{ft^2}}{\mathrm{s}}} \times 0.092903040$	$\mathrm{m^2/s}$	$\mathrm{ft^2/s}$
Pressure	p	$\frac{p_{\mathrm{IP}}}{\mathrm{psi}} = \frac{p_{\mathrm{SI}}}{\mathrm{kPa}} \times 0.14503774$	$\frac{p_{\mathrm{SI}}}{\mathrm{kPa}} = \frac{p_{\mathrm{IP}}}{\mathrm{psi}} \times 6.894757$	kPa	psi
Relative humidity	φ	$\varphi_{\mathrm{IP}} = \varphi_{\mathrm{SI}}$	$\varphi_{\mathrm{SI}} = \varphi_{\mathrm{IP}}$	-	-
Prandtl number	Pr	$Pr_{\mathrm{IP}} = Pr_{\mathrm{IP}}$	$Pr_{\mathrm{SI}} = Pr_{\mathrm{IP}}$	-	-
Mole fraction	ψ	$\psi_{\mathrm{IP}} = \psi_{\mathrm{SI}}$	$\psi_{\mathrm{SI}} = \psi_{\mathrm{IP}}$	mol/mol	mol/mol
Density	ρ	$\frac{\rho_{\mathrm{IP}}}{\frac{\mathrm{lb}}{\mathrm{ft^3}}} = \frac{\rho_{\mathrm{SI}}}{\frac{\mathrm{kg}}{\mathrm{m^3}}} \times 0.062428$	$\frac{\rho_{\mathrm{SI}}}{\frac{\mathrm{kg}}{\mathrm{m^3}}} = \frac{\rho_{\mathrm{IP}}}{\frac{\mathrm{lb}}{\mathrm{ft^3}}} \times 16.018463$	$\mathrm{kg/m^3}$	$\mathrm{lb/ft^3}$
Air-specific entropy (moist air)	s	$\frac{s_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb_a\ °R}}} = \frac{s_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg_a\ K}}} \times 0.2388459 + 0.01616365106$	$\frac{s_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg_a\ K}}} = \left(\frac{s_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb_a\ °R}}} - 0.01616365106 \right) \times 4.1868$	$\mathrm{kJ/(kg_a{\cdot}K)}$	$\mathrm{Btu/(lb_a{\cdot}°R)}$

Property		Conversion: SI Units → I-P Units	Conversion: I-P Units→ SI Units	Units SI	Units I-P
Specific entropy (water, water vapor, ice)	s_w	$\frac{s_{IP}}{\frac{\mathrm{Btu}}{\mathrm{lb_a\ ^\circ R}}} = \frac{s_{SI}}{\frac{\mathrm{kJ}}{\mathrm{kg_a\ K}}} \times 0.23884589$	$\frac{s_{SI}}{\frac{\mathrm{kJ}}{\mathrm{kg_a\ K}}} = \frac{s_{IP}}{\frac{\mathrm{Btu}}{\mathrm{lb_a\ ^\circ R}}} \times 4.1868$	$\mathrm{kJ/(kg_a{\cdot}K)}$	$\mathrm{Btu/(lb_a{\cdot}^\circ R)}$
Temperature	t	$\frac{t_{IP}}{^\circ\mathrm{F}} = \frac{t_{SI}}{^\circ\mathrm{C}} \times \frac{9}{5} + 32$	$\frac{t_{SI}}{^\circ\mathrm{C}} = \left(\frac{t_{IP}}{^\circ\mathrm{F}} - 32\right) \times \frac{5}{9}$	°C	°F
Air-specific internal energy (moist air)	u	$(u = h - pv)$ $\frac{u_{IP}}{\frac{\mathrm{Btu}}{\mathrm{lb_a}}} = \frac{h_{SI}}{\frac{\mathrm{kJ}}{\mathrm{kg_a}}} \times 0.4299226 + 7.68565365666$ $- \frac{p_{SI}}{\mathrm{kPa}} \times 0.145037738 \cdot \frac{v_{SI}}{\frac{\mathrm{m^3}}{\mathrm{kg_a}}} \times 16.018453$	$(u = h - pv)$ $\frac{u_{SI}}{\frac{\mathrm{kJ}}{\mathrm{kg_a}}} = \left(\frac{h_{IP}}{\frac{\mathrm{Btu}}{\mathrm{lb_a}}} - 7.68565365666\right) \times 2.236$ $- \frac{p_{IP}}{\mathrm{psi}} \times 6.894757293 \cdot \frac{v_{SIP}}{\frac{\mathrm{ft^3}}{\mathrm{lb_a}}} \times 0.062428$	$\mathrm{kJ/kg_a}$	$\mathrm{Btu/lb_a}$
Air-specific volume (moist air)	v	$\frac{v_{IP}}{\frac{\mathrm{ft^3}}{\mathrm{lb_a}}} = \frac{v_{SI}}{\frac{\mathrm{m^3}}{\mathrm{kg_a}}} \times 16.018453$	$\frac{v_{SI}}{\frac{\mathrm{m^3}}{\mathrm{kg_a}}} = \frac{v_{IP}}{\frac{\mathrm{ft^3}}{\mathrm{lb_a}}} \times 0.062428$	$\mathrm{m^3/kg_a}$	$\mathrm{ft^3/lb_a}$
Specific volume (water, water vapor, ice)	v_w	$\frac{v_{IP}}{\frac{\mathrm{ft^3}}{\mathrm{lb}}} = \frac{v_{SI}}{\frac{\mathrm{m^3}}{\mathrm{kg}}} \times 16.018453$	$\frac{v_{SI}}{\frac{\mathrm{m^3}}{\mathrm{kg}}} = \frac{v_{IP}}{\frac{\mathrm{ft^3}}{\mathrm{lb}}} \times 0.062428$	$\mathrm{m^3/kg}$	$\mathrm{ft^3/lb}$
Humidity ratio	W	$W_{IP} = W_{SI}$	$W_{SI} = W_{IP}$	$\mathrm{kg_w/kg_a}$	$\mathrm{lb_w/lb_a}$
Mass fraction	ξ	$\xi_{IP} = \xi_{SI}$	$\xi_{SI} = \xi_{IP}$	$\mathrm{kg_w/kg}$	$\mathrm{lb_w/lb}$
Compression factor	Z	$Z_{IP} = Z_{SI}$	$Z_{SI} = Z_{IP}$	-	-

1.3 Calculation Algorithms

1.3.1 Algorithms for Real Moist Air

The properties of moist air are calculated from the modified Hyland-Wexler model given in Herrmann, Kretzschmar, and Gatley (HKG) [1], [2]. The modifications incorporate:

- the value for the universal molar gas constant from the CODATA standard by Mohr and Taylor [22]
- the value for the molar mass of dry air from Gatley et al. [17] and that of water from IAPWS-95 [5], [6]
- the calculation of the ideal-gas parts of the heat capacity, enthalpy, and entropy for dry air from the fundamental equation of Lemmon et al. [14]
- the calculation of the ideal-gas parts of the heat capacity, enthalpy, and entropy for water vapor from IAPWS-IF97 [7], [8], [9] for $t \geq 32°\mathrm{F}$ and from IAPWS-95 [5], [6] for $t \leq 32°\mathrm{F}$
- the calculation of the vapor-pressure enhancement factor from the equation given by the models of Hyland and Wexler [21]
- the calculation of the second and third molar virial coefficients B_{aa} and C_{aaa} for dry air from the fundamental equation of Lemmon et al. [14] according to Feistel et al. [24]
- the calculation of the second and third molar virial coefficients B_{ww} and C_{www} for water and steam from IAPWS-95 [5], [6] according to Feistel et al. [24]
- the calculation of the air-water second molar cross-virial coefficient B_{aw} from Harvey and Huang [15]
- the calculation of the air-water third molar cross-virial coefficients C_{aaw} and C_{aww} from Nelson and Sauer [12], [13]
- the calculation of the saturation pressure of water from IAPWS-IF97 [7], [8], [9] for $t \geq 32°\mathrm{F}$ and of the sublimation pressure of water from IAPWS-08 [11] for $t \leq 32°\mathrm{F}$
- the calculation of the isothermal compressibility of saturated liquid water from IAPWS-IF97 [7], [8], [9] for $t \geq 32°\mathrm{F}$ and that of ice from IAPWS-06 [10] for $t \leq 32°\mathrm{F}$ in the determination of the vapor-pressure enhancement factor
- the calculation of Henry's constant from the IAPWS Guideline 2004 [16] in the determination of the enhancement factor. The mole fractions for the three main components of dry air were taken from Lemmon et al. [14]. Argon was not considered in the calculation of Henry’s constant in the former research projects, but it is now the third component of dry air.

The transport properties of moist air are calculated from the model given in Herrmann et al. [3], [4].

1.3.2 Algorithms for Steam and Water for Temperatures *t* ≥ 32°F

The *p*-*T* diagram in Fig. 1 shows the formulations used for water and water vapor. The temperature range above 32°F is covered by IAPWS-IF97 [7], [8], [9]:

- The saturation line is calculated from the IAPWS-IF97 saturation pressure equation $p_\mathrm{s}^{97}(t)$ and saturation temperature equation $t_\mathrm{s}^{97}(p)$.
- The properties in the liquid region including saturated-liquid line are calculated from the fundamental equation of the IAPWS-IF97 region 1.
- The properties in the vapor region including saturated-vapor line are calculated from the fundamental equation of the IAPWS-IF97 region 2.

1.3.3 Algorithms for Steam and Ice for Temperatures *t* ≤ 32°F

- The sublimation curve is covered by the IAPWS-08 sublimation pressure equation $p_\mathrm{subl}^{08}(t)$ [11] (see Fig. 1).
- The properties of ice including saturated ice are determined by the fundamental equation of the IAPWS-06 [10].
- The properties of vapor including saturated vapor are calculated from the fundamental equation of IAPWS-95 [5], [6].

1.3.4 Overview of the Applied Formulations for Steam, Water, and Ice

The following *p*-*T* diagram shows the used IAPWS Formulations and the ranges where they are applied.

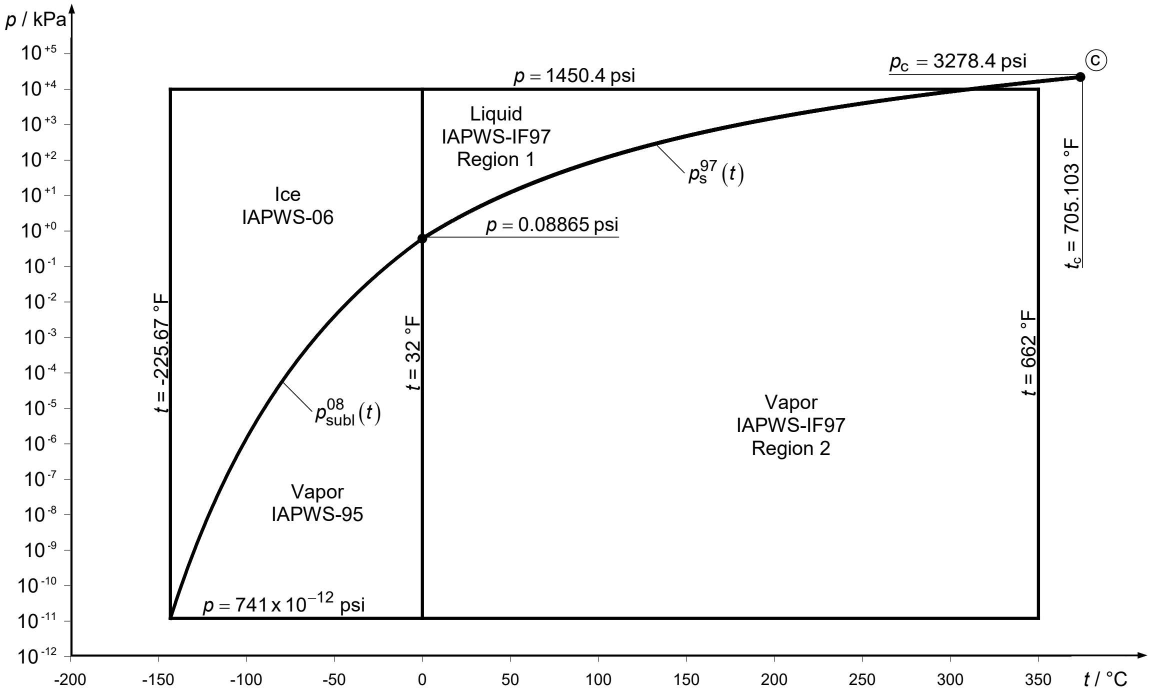


Figure 1: *p*-*T* diagram with used IAPWS formulations for steam, water, and ice.

2 Add-On FluidEES for Engineering Equation Solver® for ASHRAE-LibHuAirProp-IP

2.1 Installing FluidEES

The FluidEES Add-On has been developed to conveniently calculate thermodynamic properties in the Engineering Equation Solver® (EES). It enables, within EES, the direct call of functions relating to real moist air, steam, water and ice from the ASHRAE-LibHuAirProp property library.

2.1.1 Installing FluidEES including LibHuAirProp

In this section, the installation of FluidEES LibHuAirProp_IP and of LibHuAirProp_SI is described. Before you begin, it is best to close any Windows® applications you may have open, since Windows® may need to be rebooted during the installation process.

The installation routine for **32-bit** and **64-bit** versions of **EES** is similar. The following instructions are valid for both versions.

After you have downloaded and extracted the zip-file

for 64-bit version of EES:

"CD_FluidEES_ASHRAE_LibHuAirProp_x64.zip"

for 32-bit version of EES:

"CD_FluidEES_ASHRAE_LibHuAirProp.zip"

you will see the folder

for 64-bit version of EES:

\CD_FluidEES_ASHRAE_LibHuAirProp_x64

for 32-bit version of EES:

\CD_FluidEES_ASHRAE_LibHuAirProp

in your Windows Explorer, Norton Commander or any other similar program you may be using.

Now, open this folder by double-clicking on it.

Within this folder you will see the following files:

for 64-bit version of EES:

FluidEES_ASHRAE_LibHuAirProp_Users_Guide.pdf
FluidEES_ASHRAE_LibHuAirProp_64_Setup.msi
setup.exe

and the folders

vcredist_x64
WindowsInstaller3_1

for 32-bit version of EES:

FluidEES_ASHRAE_LibHuAirProp_Users_Guide.pdf
FluidEES_LibHuAirProp_Setup.msi

I-P – 2/2

In order to run the installation of FluidEES including the ASHRAE-LibHuAirProp-IP and ASHRAE-LibHuAirProp-SI property library, double-click on the file

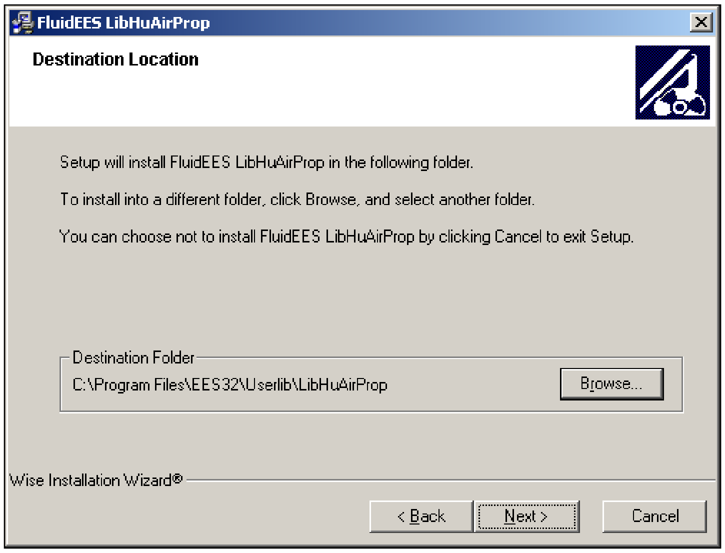
setup.exe (for 64-bit version of EES) or on
FluidEES_LibHuAirProp_Setup.exe (for 32-bit version of EES).

Installation may start with a window noting that all Windows® programs should be closed.

When this is the case, the installation can be continued. Click the "Next >" button.

In the following dialog box, "Destination Location" (see figure below), the default path where Engineering Equation Solver has been installed will be shown (the standard location is:

for 64-bit version of EES:
C:\Program Files\EES64\Userlib64\LibHuAirProp).
for 32-bit version of EES:
C:\Program Files\EES32\Userlib\LibHuAirProp).



FluidEES LibHuAirProp

Destination Location

Setup will install FluidEES LibHuAirProp in the following folder.

To install into a different folder, click Browse, and select another folder.

You can choose not to install FluidEES LibHuAirProp by clicking Cancel to exit Setup.

Destination Folder
C:\Program Files\EES32\Userlib\LibHuAirProp

Figure 2.1.1: "Destination Location" for the 32-bit version

Click on "Next >" in the window "Destination Location."
Click on the "Next >" button in the "Start Installation" window.
The FluidEES files are now being copied into the "\LibHuAirProp" folder on your hard drive.
Click the "Finish >" button in the next window to complete installation.
The installation program has copied the following files for the 64-bit installation into the directory "C:\Program Files\EES64\Userlib64\LibHuAirProp":

LC.dll	- Dynamic link library for use in Windows® programs
LibHuAirProp_IP.chm	- Help file of the LibHuAirProp_IP property library
LibHuAirProp_IP.ctx	- Interface including property functions of LibHuAirProp_IP for EES
LibHuAirProp_IP.dll64	- Dynamic link library with property functions of LibHuAirProp_IP
LibHuAirProp_SI.chm	- Help file of the LibHuAirProp_SI property library
LibHuAirProp_SI.ctx	- Interface including property functions of LibHuAirProp_SI for EES
LibHuAirProp_SI.dll64	- Dynamic link library with property functions of LibHuAirProp_SI

The installation program has copied the following files for the 32-bit installation into the directory "C:\Program Files\EES32\Userlib\LibHuAirProp":

advapi32.dll	- Dynamic link library for use in Windows® programs
Dformd.dll	- Dynamic link library for use in Windows® programs
Dforrt.dll	- Dynamic link library for use in Windows® programs
INSTALL.LOG	- Log file
LCKCE.dll	- Dynamic link library for use in Windows® programs
LibHuAirProp_IP.chm	- Help file of the LibHuAirProp_IP property library
LibHuAirProp_IP.ctx	- Interface including property functions of LibHuAirProp_IP for EES
LibHuAirProp_IP.dll	- Dynamic link library with property functions of LibHuAirProp_IP
LibHuAirProp_SI.chm	- Help file of the LibHuAirProp_SI property library
LibHuAirProp_SI.ctx	- Interface including property functions of LibHuAirProp_SI for EES
LibHuAirProp_SI.dll	- Dynamic link library with property functions of LibHuAirProp_SI
msvcp60.dll	- Dynamic link library for use in Windows® programs
msvcrt.dll	- Dynamic link library for use in Windows® programs
UNWISE.EXE	- File to remove the LibHuAirProp library
UNWISE.INI	- File to remove the LibHuAirProp library

You can now select the ASHRAE-LibHuAirProp property functions from within Engineering Equation Solver®.

2.1.2 The FluidEES Help System

As mentioned earlier, FluidEES also provides detailed online help functions.
Information on individual property functions may be accessed via the following steps:

- Click "Options" in the EES menu bar and select "Function Info".
- The "Function Information" window will appear. Select "External routines" and double-click on the entry "LIBHUAIRPROP_IP.DLL".
- A list with calculable functions of the "LibHuAirProp_IP" library appears.
- Find and select the desired function, e.g. "h_ptW_HAP_IP" and click the ? Function Info button above.

If the "LibHuAirProp_IP.chm" function help cannot be found, confirm the question whether you want to look for it yourself with "Yes." Select the "LibHuAirProp_IP.chm" file in the installation menu of FluidEES in the window which is opened, the standard being

for 64-bit version of EES:

C:\Program Files\EES64\Userlib64\LibHuAirProp

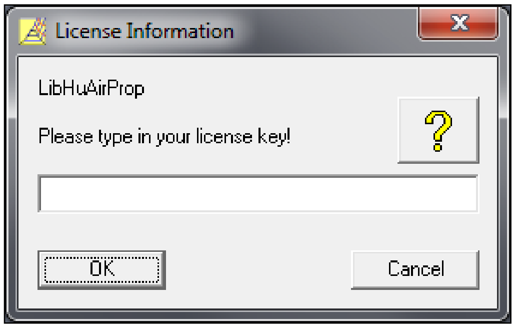
for 32-bit version of EES:

C:\Program Files\EES32\Userlib\LibHuAirProp

and click "Yes" in order to complete the search.

2.2 Licensing the LibHuAirProp Property Library

The licensing procedure must be carried out when Engineering Equation Solver® starts up and a FluidEES prompt message appears. In this case, you will see the "License Information" window for LibHuAirProp (see figure below).

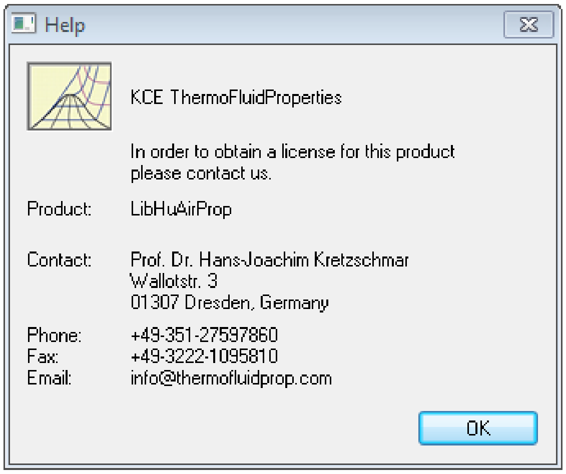


LibHuAirProp

Please type in your license key!

Figure 2.2.1: "License Information" window

Here you are asked to type in the license key which you have obtained from ASHRAE. If you do not have this, or have any questions, you will find contact information on the "Content" page of this User's Guide or by clicking the yellow question mark in the "License Information" window. Then the following window will appear:



KCE ThermoFluidProperties

In order to obtain a license for this product
please contact us.

Product: LibHuAirProp

Contact: Prof. Dr. Hans-Joachim Kretzschmar
Wallotstr. 3
01307 Dresden, Germany

Phone: +49-351-27597860
Fax: +49-3222-1095810
Email: info@thermofluidprop.com

Figure 2.2.2: "Help" window

If you do not enter a valid license it is still possible to start EES by clicking "Cancel". In this case, the LibHuAirProp property library will display the result "–11111111" for every calculation you ask it to make.

The "License Information" window will appear every time you use FluidEES LibHuAirProp until you enter a license code to complete registration. If you decide not to use FluidEES LibHuAirProp, you can uninstall the program following the instructions given in section 2.4 of this User's Guide.

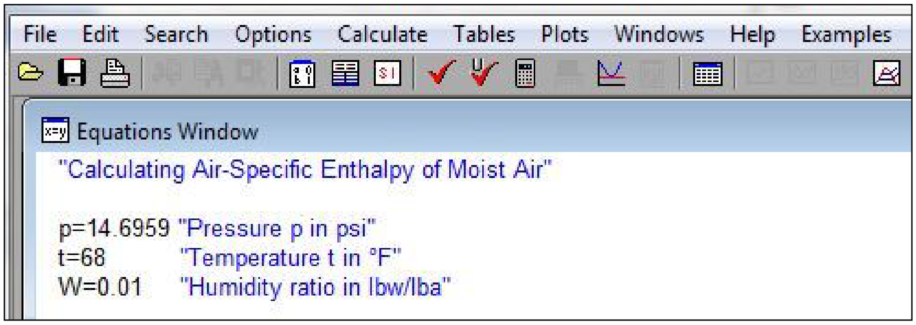
With this procedure both the LibHuAirProp-SI and LibHuAirProp-IP property libraries have been licensed.

2.3 Example: Calculation of *h* = f(*p*,*t*,*W*)

Now we will calculate, step by step, the air-specific enthalpy *h* of humid air as a function of total pressure *p*, temperature *t* and humidity ratio *W* for humid air, using FluidEES with LibHuAirProp-IP in the Engineering Equation Solver®.

How to perform a calculation with FluidEES:

- Start Engineering Equation Solver® (EES).
- The LibHuAirProp-IP library, if installed, is loaded by the program automatically.
- We recommend preparing an EES sheet, as shown in Figure 2.3.1.
  Note: the units of *p*, *t*, and *W* must correspond to those in Chapter 1.



```
"Calculating Air-Specific Enthalpy of Moist Air"

p=14.6959 "Pressure p in psi"
t=68      "Temperature t in °F"
W=0.01    "Humidity ratio in lbw/lba"
```

Figure 2.3.1: Preparing an EES® sheet for the calculation

- The function parameters values stand for:
  - First operand: Total pressure *p* = 14.6959 psi
    (Range of validity: p = 0.00145 … 1450.4 psi)
  - Second operand: Temperature *t* = 68°F
    (Range of validity: t = –226.67 ... 662°F)
  - Third operand: Humidity Ratio *W* = 0.01 lb_w/lb_a (*lb water* per *lb dry air*)
    (Range of validity: W = 0 … 10 lb_w/lb_a)
- Confirm your entry by pressing the "ENTER" key.

Note:
EES adapts to the language that is set in the "Regional and Language Options," which can be found in the "Control Panel." If you run Engineering Equation Solver® on an English version of Windows®, the standard decimal separator will be a dot (as shown in Fig. 2.1.4 and in the following sample calculation). If your computer is set to German, for example, the expected decimal separator will be a comma. In this case enter a comma in the values above instead of a dot. You can find additional information on this issue by clicking on "Help" in the EES menu bar and then select "Help Index". Click on "Search" in the window which appears, type "decimal separator" and press the "ENTER" key.

- For calculating h = f(p,t,W), call up the function "h_ptW_HAP_IP" of the property library LibHuAirProp_IP as follows:
- Click on "Options" in the EES menu bar and select "Function Info".
- The "Function Information" window will appear. Select "External routines" and you will see the screen shown here in Figure 2.3.2.

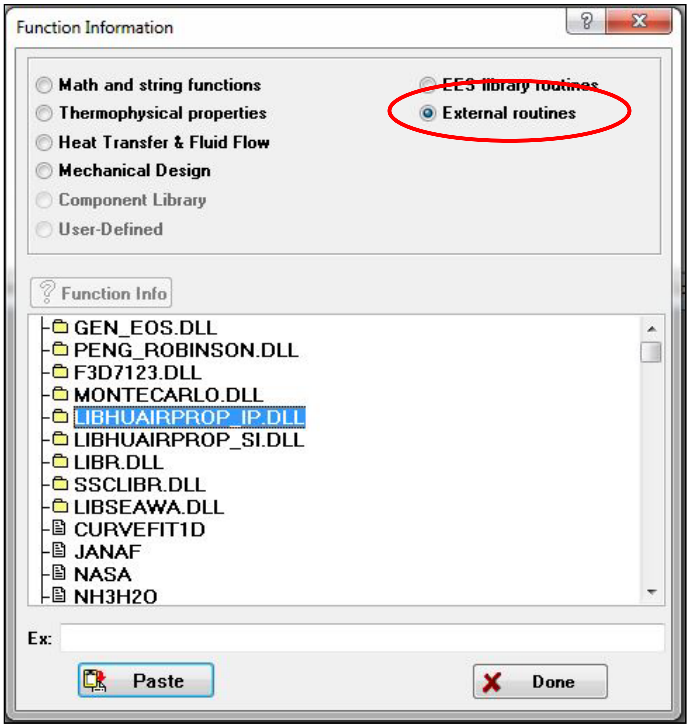
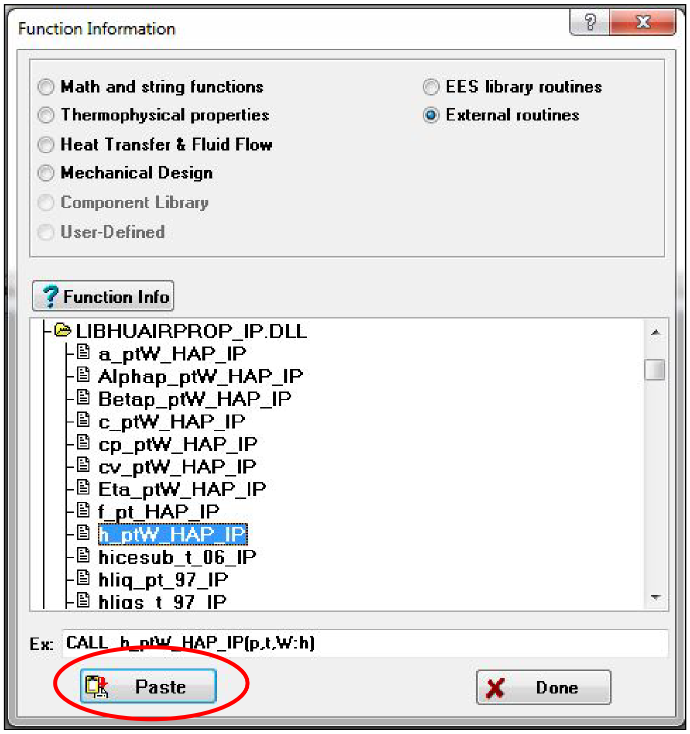


Figure 2.3.2: "Function Information" window offering different libraries (routines)

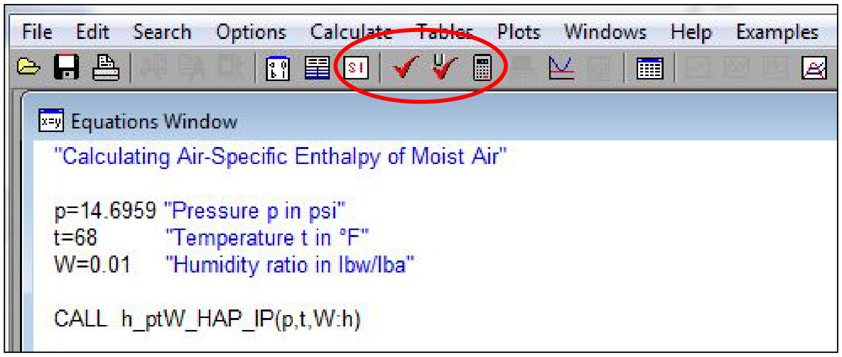
- Double-click on the entry "LIBHUAIRPROP_IP.DLL".
- A list with calculable functions of the "LibHuAirProp_IP" library appears.
- Find and select the desired function, here "h_ptW_HAP_IP" (see Figure 2.3.3), and click the "Paste" button below.



Ex: CALL h_ptW_HAP_IP(p,t,W:h)

Figure 2.3.3: Selecting the "h_ptW_HAP_IP" function

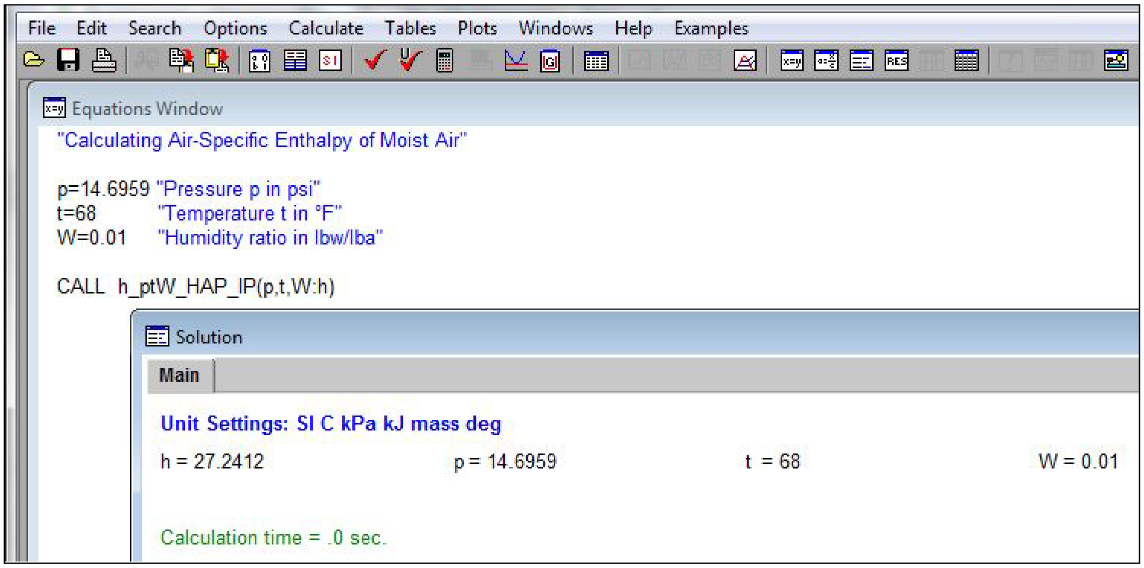
- The selected function will be copied and now appears in the "Equations Window" (see Fig. 2.3.4).



```
"Calculating Air-Specific Enthalpy of Moist Air"

p=14.6959 "Pressure p in psi"
t=68       "Temperature t in °F"
W=0.01     "Humidity ratio in lbw/lba"

CALL  h_ptW_HAP_IP(p,t,W:h)
```

Figure 2.3.4: "Equations Window" with the call of the property function

- Now, you can check the syntax of the instructions in the "Equations Window" by clicking the ✓ symbol in the upper menu bar of EES. The program tests whether or not the syntax is correct (e.g. dots as decimal separators versus commas). Confirm the "Information" window which appears by clicking the "OK" button.
- Then click the 🖩 symbol in the upper menu bar of EES to start the calculation.
- Soon you will see the "Calculations Completed" window. Leave this window by clicking the "Continue" button.
- The result for the air-specific enthalpy h appears in the "Solution" window (see Figure 2.3.5).



```
"Calculating Air-Specific Enthalpy of Moist Air"

p=14.6959 "Pressure p in psi"
t=68       "Temperature t in °F"
W=0.01     "Humidity ratio in lbw/lba"

CALL  h_ptW_HAP_IP(p,t,W:h)
```

```
Unit Settings: SI C kPa kJ mass deg
h = 27.2412            p = 14.6959            t = 68            W = 0.01

Calculation time = .0 sec.
```

Figure 2.3.5: "Solution" window showing the result

The calculation of *h* = f(*p*,*t*,*W*) has thus been carried out.

⇒ The result in our sample calculation here is: "h = 27.2412". The corresponding unit is Btu/lba (see table of the property functions in Chapter 1).

For further property functions calculable in FluidEES see the function table in Chapter 1.

2.4 Removing FluidEES including LibHuAirProp

In order to remove the property library ASHRAE-LibHuAirProp from your hard drive in Windows®, click "Start" in the lower task bar, then "Settings" and "Control Panel."

Afterwards double-click on "Add or Remove Programs."

In the list box of the "Add or Remove Programs" menu which appears, select "FluidEES LibHuAirProp" by clicking on it and click the "Change/Remove" button.

In the following dialog box select "Automatic" and then click the "Next >" button.

Then confirm the menu "Perform Uninstall" by clicking the "Finish" button.

Finally, close the "Add or Remove Programs" and "Control Panel" windows.

"FluidEES LibHuAirProp" has now been removed.

3 Property Functions of ASHRAE-LibHuAirProp-IP

3.1 Functions for Real Moist Air

Thermal Diffusivity *a* = f(*p*,*t*,*W*)

Function Name:

a_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION A_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi

t - Temperature t in °F

W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

a_ptW_HAP_IP - Thermal diffusivity of humid air in $\mathrm{ft^2/s}$

Range of Validity:

Temperature t:	from -99.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le W_s$

Comments:

- Thermal diffusivity $a = \dfrac{\lambda}{\rho \cdot c_p}$
- Calculation for supersaturated humid air ($W > W_s$) is not possible

Result for Wrong Input Values:

a_ptW_HAP_IP = -1000

References:

$\lambda(p,t,W)$	Herrmann et al. [3], [4]
$\rho(p,t,W)$	Herrmann et al. [1], [2]
$c_p(p,t,W)$	Herrmann et al. [1], [2]

Relative Pressure Coefficient α_p = f(p,t,W)

Function Name:

alphap_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION ALPHAP_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\text{lb}_\text{w}/\text{lb}_\text{a}$

Result:

alphap_ptW_HAP_IP - Relative pressure coefficient of humid air in 1/°R

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le W_\text{s}$

Comments:

- Relative pressure coefficient $\alpha_p = \frac{1}{p}\left(\frac{\partial p}{\partial T}\right)_\text{v}$
- Calculation for supersaturated humid air ($W > W_\text{s}$) is not possible

Result for Wrong Input Values:

alphap_ptW_HAP_IP = -1000

References:

$\alpha_p(p,t,W)$ Herrmann et al. [1], [2]

Isothermal Stress Coefficient β_p = f(p,t,W)

Function Name:

betap_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION BETAP_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\text{lb}_\text{w}/\text{lb}_\text{a}$

Result:

betap_ptW_HAP_IP - Isothermal stress coefficient of humid air in lb/ft^3

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le W_\text{s}$

Comments:

- Isothermal stress coefficient $\beta_p = -\frac{1}{p}\left(\frac{\partial p}{\partial v}\right)_\text{T}$
- Calculation for supersaturated humid air ($W > W_\text{s}$) is not possible

Result for Wrong Input Values:

betap_ptW_HAP_IP = -1000

References:

$\beta_p(p,t,W)$ Herrmann et al. [1], [2]

Speed of Sound *c* = f(*p*,*t*,*W*)

Function Name:

c_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION C_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

c_ptW_HAP_IP - Speed of sound of humid air in ft/s

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- Speed of sound $c = v\sqrt{-\left(\frac{\partial p}{\partial v}\right)_s}$
- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

c_ptW_HAP_IP = -1000

References:

$c(p,t,W)$ Herrmann et al. [1], [2]

Isobaric Heat Capacity c_p = f(p,t,W)

Function Name:

cp_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION   CP_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi

t - Temperature t in °F

W - Humidity ratio W in $\text{lb}_\text{w}/\text{lb}_\text{a}$

Result:

cp_ptW_HAP_IP - Isobaric heat capacity of humid air in Btu/(lb °R)

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le W_\text{s}$

Comments:

- Isobaric heat capacity $c_p = \left(\frac{\partial h}{\partial T}\right)_\text{p}$

- Calculation for supersaturated humid air ($W > W_\text{s}$) is not possible

Result for Wrong Input Values:

cp_ptW_HAP_IP = -1000

References:

$c_p(p,t,W)$ Herrmann et al. [1], [2]

Isochoric Heat Capacity c_v = f(p,t,W)

Function Name:

cv_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION CV_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\text{lb}_\text{w}/\text{lb}_\text{a}$

Result:

cv_ptW_HAP_IP - Isochoric heat capacity of humid air in Btu/(lb °R)

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le W_\text{s}$

Comments:

- Isochoric heat capacity $c_v = \left(\frac{\partial u}{\partial T}\right)_\text{v}$
- Calculation for supersaturated humid air ($W > W_\text{s}$) is not possible

Result for Wrong Input Values:

cv_ptW_HAP_IP = -1000

References:

$c_v(p,t,W)$ Herrmann et al. [3], [4]

Enhancement Factor *f* = f(*p*,*t*)

Function Name:

f_pt_HAP_IP

Fortran Program:

REAL*8 FUNCTION F_PT_HUAIRPROP(P,T), REAL*8 P,T

Input Values:

p	-	Total pressure p in psi
t	-	Temperature t in °F

Result:

f_pt_HAP_IP - Enhancement factor of water (decimal ratio)

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi

Comments:

- Enhancement factor $f = \frac{p_{H2O,s}}{p_s(t)}$

with $p_s(t)$ for $t \geq 32$°F - Steam pressure of water

for $t < 32$°F - Sublimation pressure of water

- Describes the enhancement of the saturation pressure of water in the air atmosphere under elevated pressure
- Derived iteratively from the isothermal compressibility of liquid water, from Henry's constant [15], [16] and from the virial coefficients of air, water, and the air-water mixture

Result for Wrong Input Values:

f_pt_HAP_IP = -1000

References:

$f(p,t)$	Herrmann et al. [1], [2]

Air-Specific Enthalpy *h* = f(*p*,*t*,*W*)

Function Name:

h_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION H_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p	-	Total pressure p in psi
t	-	Temperature t in °F
W	-	Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

h_ptW_HAP_IP - Air-specific enthalpy in $\mathrm{Btu/lb_a}$

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

h_ptW_HAP_IP = -1000

References:

$h(p,t,W)$	Herrmann et al. [1], [2]
$h_w(p,t)$	IAPWS-IF97 [7], [8] and IAPWS-06 [11]
$h_a(t)$	Lemmon et al. [14]

Dynamic Viscosity η = f(p,t,W)

Function Name:

Eta_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION ETA_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

Eta_ptW_HAP_IP - Dynamic viscosity of humid air in (lb s/ft²)

Range of Validity:

Temperature t:	from -99.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- A new very accurate algorithm is implemented between 32°F and 662°F
- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

Eta_ptW_HAP_IP = -1000

References:

$\eta(p,t,W)$	Herrmann et al. [3], [4]
$\eta_\mathrm{w}(p,t)$	IAPWS-IF97 [7], [8] and IAPWS-06 [19]
$\eta_\mathrm{a}(t)$	Lemmon et al. [18]

Isentropic Exponent $\kappa = f(p,t,W)$

Function Name:

Kappa_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION KAPPA_PTW_HUAIRPROP(P,T, W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

Kappa_ptW_HAP_IP - Isentropic exponent

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- Isentropic exponent $\kappa = -\frac{v}{p}\left(\frac{\partial p}{\partial v}\right)_s$
- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets homogeneously mixed) is applied for $t \ge 32$°F. For temperatures below (ice fog) the value of the saturated state is applied.

Result for Wrong Input Values:

Kappa_ptW_HAP_IP = -1000

References:

$v(p,t,W)$ Herrmann et al. [1], [2]

Thermal Conductivity $\lambda = f(p, t, W)$

Function Name:

Lambda_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION LAMBDA_PTW_HUAIRPROP(P,T, W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

Lambda_ptW_HAP_IP - Thermal conductivity in Btu/(h ft °R)

Range of Validity:

Temperature t:	from -99.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- A new very accurate algorithm is implemented between 32°F and 662°F
- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

Lambda_ptW_HAP_IP = -1000

References:

$\lambda(p,t,W)$	Herrmann et al. [3], [4]
$\lambda_w(p,t)$	IAPWS-IF97 [7], [8] and IAPWS-08 [20]
$\lambda_a(t)$	Lemmon et al. [18]

Kinematic Viscosity ν = f(p,t,W)

Function Name:

Ny_ptW_HAP_IP

Fortran Program:

```
REAL*8 FUNCTION   NY_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W
```

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in lb_w/lb_a

Result:

Ny_ptW_HAP_IP - Kinematic viscosity in ft^2/s

Range of Validity:

Temperature t:	from -99.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le W_s$

Comments:

- Kinematic Viscosity $\nu = \frac{\eta}{\rho}$

Result for Wrong Input Values:

Ny_ptW_HAP_IP = -1000

References:

$\eta(p,t,W)$ Herrmann et al. [3], [4]
$\rho(p,t,W)$ Herrmann et al. [1], [2]

Backward Function: Pressure *p* = f(*t*,*s*,*W*)

Function Name:

p_tsW_HAP_IP

Fortran Program:

REAL*8 FUNCTION P_TSW_HUAIRPROP(T,S,W), REAL*8 T,S,W

Input Values:

t - Temperature *t* in °F

s - Air-specific entropy *s* in Btu/($\mathrm{lb_a}$ °R)

W - Humidity ratio *W* in $\mathrm{lb_w/lb_a}$

Result:

p_tsW_HAP_IP - Total pressure of humid air in psi

Range of Validity:

Temperature *t*:	from -225.67°F to 662°F
Air-specific entropy *s*:	from -6.32 Btu/($\mathrm{lb_a}$ °R) to 9.32877 Btu/($\mathrm{lb_a}$ °R)
Humidity ratio *W*:	$0 \le W \le 10$ $\mathrm{lb_w/lb_a}$

Comments:

- Iteration of total pressure *p* from *s* = f(*p*,*t*,*W*)

Result for Wrong Input Values:

p_tsW_HAP_IP = -1000

References:

$s(p,t,W)$ Herrmann et al. [1], [2]

Pressure p = f(z_{ele})

Function Name:

p_zele_HAP_IP

Fortran Program:

REAL*8 FUNCTION   P_ZELE_HUAIRPROP(ZELE), REAL*8 ZELE

Input Values:

z_{ele} - Elevation z_{ele} in ft

Result:

p_zele_HAP_IP - Pressure of humid air in psi

Range of Validity:

Elevation z_{ele}	from -16,404 ft to 36,089 ft

Comments:

- Pressure of humid air from elevation

- $p(z_{ele}) = 14.696\,\text{psi} \cdot \left(1 - 6.8754 \cdot 10^{-6} \cdot \frac{z_{ele}}{\text{ft}}\right)^{5.256}$

Result for Wrong Input Values:

p_zele_HAP_IP = -1000

References:

$p(z_{ele})$	ASHRAE [23]

Partial Pressure of Air p_{Air} = f(p,t,W)

Function Name:

pAir_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION PAIR_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi

t - Temperature t in °F

W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

pAir_ptW_HAP_IP - Partial pressure of (dry) air in humid air in psi

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- Partial pressure of (dry) air in humid air $p_{\mathrm{Air}} = 1 - p_{\mathrm{H2O}}$
- Partial pressure of water vapor at saturation is calculated in case of supersaturated humid air ($W > W_{\mathrm{s}}(p,t)$)
- The temperature value is used to calculate the saturation state

Result for Wrong Input Values:

pAir_ptW_HAP_IP = -1000

References:

$p_{\mathrm{H2O}}(p,W)$ Herrmann et al. [1], [2]

Partial Pressure of Water Vapor p_{H2O} = f(p, t, W)

Function Name:

pH2O_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION   PH2O_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

pH2O_ptW_HAP_IP - Partial pressure of water vapor in humid air in psi

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- Partial pressure of water vapor in humid air $p_{H2O} = \dfrac{W \cdot p}{\left(\dfrac{R_a}{R_w} + W\right)}$
- Partial pressure of water vapor at saturation is calculated in case of supersaturated humid air ($W > W_s(p,t)$)
- The temperature value is used to calculate the saturation state

Result for Wrong Input Values:

pH2O_ptW_HAP_IP = -1000

References:

$p_{H2O}(p,W)$ Herrmann et al. [1], [2]

Partial Sat. Pressure of Water Vapor in Humid Air $p_{H2O,s}$ = f(p,t)

Function Name:

pH2Os_pt_HAP_IP

Fortran Program:

```
REAL*8 FUNCTION   PH2OS_PT_HUAIRPROP(P,T), REAL*8 P,T
```

Input Values:

p - Total pressure p in psi

t - Temperature t in °F

Result:

pH2Os_pt_HAP_IP - Partial saturation pressure of water vapor in humid air in psi

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi

Comments:

- Partial pressure of water vapor at saturation $p_{H2O,s} = f \cdot p_s(t)$
  with $p_s(t)$ for $t \geq 32$°F - Steam pressure of water

  for $t < 32$°F - Sublimation pressure of water

Result for Wrong Input Values:

pH2Os_pt_HAP_IP = -1000

References:

$f(p,t)$		Herrmann et al. [1], [2]
$p_s(t)$	for $t \geq 32$°F	IAPWS-IF97 [7], [8]
	for $t < 32$°F	IAPWS-08 [11]

Relative Humidity φ = f(p,t,W)

Function Name:

phi_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION PHI_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in lb_w/lb_a

Result:

phi_ptW_HAP_IP - Relative humidity (decimal ratio)

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\ \text{lb}_w/\text{lb}_a$

Comments:

- Relative humidity $\varphi = \frac{p_{\text{H2O}}}{p_{\text{H2O,s}}}$
- This equation is valid for $p_{\text{H2O}} \le p_{\text{H2O,s}}$ and for $0 \le \varphi \le 1$

Result for Wrong Input Values:

phi_ptW_HAP_IP = -1000

References:

$\varphi(p,t,W)$ Herrmann et al. [1], [2]

Prandtl Number *Pr* = f(*p*,*t*,*W*)

Function Name:

Pr_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION PR_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

Pr_ptW_HAP_IP - Prandtl number

Range of Validity:

Temperature t:	from -99.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10 \mathrm{lb_w/lb_a}$

Comments:

- Prandtl number $Pr = \frac{\eta \cdot c_p}{\lambda}$
- Calculation for supersaturated humid air ($W > W_s$) is not possible

Result for Wrong Input Values:

Pr_ptW_HAP_IP = -1000

References:

$\eta(p,t,W)$	Herrmann et al. [3], [4]
$c_p(p,t,W)$	Herrmann et al. [3], [4]
$\lambda(p,t,W)$	Lemmon et al. [20]

Mole Fraction of Air ψ_{Air} = f(W)

Function Name:

PsiAir_W_HAP_IP

Fortran Program:

REAL*8 FUNCTION PSIAIR_W_HUAIRPROP(W), REAL*8 W

Input Values:

W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

PsiAir_W_HAP_IP - Mole fraction of (dry) air in humid air in $\mathrm{mol_a/mol}$

Range of Validity:

Humidity ratio W: $0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- Mole fraction of air $\psi_{\mathrm{Air}} = 1 - \psi_{\mathrm{H2O}} = 1 - \left(\dfrac{W}{\dfrac{R_{\mathrm{a}}}{R_{\mathrm{H2O}}} + W} \right)$

Result for Wrong Input Values:

PsiAir_W_HAP_IP = -1000

References:

$\psi_{\mathrm{Air}}(W)$ Herrmann et al. [1], [2]

Mole Fraction of Water ψ_{H2O} = f(W)

Function Name:

PsiH2O_W_HAP_IP

Fortran Program:

REAL*8 FUNCTION PSIH2O_W_HUAIRPROP(W), REAL*8 W

Input Values:

W - Humidity ratio W in lb_w/lb_a

Result:

PsiH2O_W_HAP_IP - Mole fraction of water in humid air in mol_w/mol

Range of Validity:

Humidity ratio W: $0 \le W \le 10\, lb_w/lb_a$

Comments:

- Mole fraction of water $\psi_{H2O} = \dfrac{W}{\dfrac{R_a}{R_{H2O}} + W}$

Result for Wrong Input Values:

PsiH2O_W_HAP_IP = -1000

References:

$\psi_{H2O}(W)$ Herrmann et al. [1], [2]

Density ρ = f(p,t,W)

Function Name:

Rho_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION RHO_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in lb_w/lb_a

Result:

Rho_ptW_HAP_IP - Density of humid air in lb/ft^3

Range of Validity:

Temperature *t*:	from -225.67°F to 662°F
Total pressure *p*:	from 0.00145 psi to 1450.4 psi
Humidity ratio *W*:	$0 \le W \le 10\,\text{lb}_w/\text{lb}_a$

Comments:

- Density of humid air obtained from air-specific volume: $\rho = \dfrac{1+W}{v}$

Result for Wrong Input Values:

Rho_ptW_HAP_IP = -1000

References:

$\rho(p,t,W)$ Herrmann et al. [1], [2]

Air-Specific Entropy $s = \mathrm{f}(p,t,W)$

Function Name:

s_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION S_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi

t - Temperature t in °F

W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

s_ptW_HAP_IP - Air-specific entropy in $\mathrm{Btu/(lb_a \cdot {}^\circ R)}$

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

s_ptW_HAP_IP = -1000

References:

$s(p,t,W)$ Herrmann et al. [1], [2]

Backward Function: Temperature *t* = f(*p*,*h*,*φ*)

Function Name:

t_phphi_HAP_IP

Fortran Program:

REAL*8 FUNCTION   T_PHPHI_HUAIRPROP(P,H,PHI), REAL*8 P,H,PHI

Input Values:

p - Total pressure p in psi
h - Air-specific enthalpy h in Btu/lb$_a$
φ - Relative humidity φ (decimal ratio)

Result:

t_phphi_HAP_IP - Temperature from pressure, enthalpy, and relative humidity in °F

Range of Validity:

Total pressure p:	from 0.00145 psi to 1450.4 psi
Air-specific enthalpy h:	from -2469.22 Btu/lb$_a$ to 12772.088 Btu/lb$_a$
Relative humidity φ:	$0 \le \varphi \le 1$

Comments:

- Iteration of temperature t from $h = f(p,t,W)$ using $W = f(p,t,\varphi)$

Result for Wrong Input Values:

t_phphi_HAP_IP = -1000

References:

$h(p,t,W)$ Herrmann et al. [1], [2]

Backward Function: Temperature *t* = f(*p*,*h*,*W*)

Function Name:

t_phW_HAP_IP

Fortran Program:

REAL*8 FUNCTION   T_PHW_HUAIRPROP(P,H,W), REAL*8 P,H,W

Input Values:

p - Total pressure *p* in psi
h - Air-specific enthalpy *h* in Btu/lb_a
W - Humidity ratio *W* in lb_w/lb_a

Result:

t_phW_HAP_IP - Temperature from pressure, enthalpy, and humidity ratio in °F

Range of Validity:

Total pressure *p*:	from 0.00145 psi to 1450.4 psi
Air-specific enthalpy *h*:	from -2469.22 Btu/lb_a to 12772.088 Btu/lb_a
Humidity ratio *W*:	$0 \le W \le 10\,\text{lb}_w/\text{lb}_a$

Comments:

- Iteration of temperature *t* from *h* = f(*p*,*t*,*W*)

Result for Wrong Input Values:

t_phW_HAP_IP = -1000

References:

$h(p,t,W)$ Herrmann et al. [1], [2]

Backward Function: Temperature *t* = f(*p*,*s*,*W*)

Function Name:

t_psW_HAP_IP

Fortran Program:

REAL*8 FUNCTION T_PSW_HUAIRPROP(P,S,W), REAL*8 P,S,W

Input Values:

p - Total pressure *p* in psi

s - Air-specific entropy in Btu/($\mathrm{lb_a} \cdot$ °R)

W - Humidity ratio *W* in $\mathrm{lb_w/lb_a}$

Result:

t_psW_HAP_IP - Temperature from pressure, entropy, and humidity ratio in °F

Range of Validity:

Total pressure *p*:	from 0.00145 psi to 1450.4 psi
Air-specific entropy *s*:	from -6.32 Btu/($\mathrm{lb_a}$ °R) to 9.32877 Btu/($\mathrm{lb_a}$ °R)
Humidity ratio *W*:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- Iteration of temperature *t* from *s* = f(*p*,*t*,*W*)

Result for Wrong Input Values:

t_psW_HAP_IP = -1000

References:

$s(p,t,W)$ Herrmann et al. [1], [2]

Backward Function: Temperature $t = \mathrm{f}(p, t_{\mathrm{wb}}, W)$

Function Name:

t_ptwbW_HAP_IP

Fortran Program:

REAL*8 FUNCTION T_PTWBW_HUAIRPROP(P,TWB,W), REAL*8 P,TWB,W

Input Values:

p - Total pressure p in psi
t_{wb} - Wet-bulb temperature in °F
W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

t_ptwbW_HAP_IP - Temperature from pressure, wet bulb temperature and humidity ratio in °F

Range of Validity:

Total pressure p:	from 0.00145 psi to 1450.4 psi
Wet bulb temperature t_{wb}:	from -225.67°F to 662°F
Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- Iterationof temperature t from $t_{\mathrm{wb}} = \mathrm{f}(p, t, W)$

Result for Wrong Input Values:

t_ptwbW_HAP_IP = -1000

References:

$t_{\mathrm{wb}}(p, t, W)$ Herrmann et al. [1], [2]

Dew-Point/Frost-Point Temperature t_d = f(p,W)

Function Name:

td_pW_HAP_IP

Fortran Program:

REAL*8 FUNCTION TD_PW_HUAIRPROP(P,W), REAL*8 P,W

Input Values:

p - Total pressure p in psi

W - Humidity ratio W in lb_w/lb_a

Result:

td_pW_HAP_IP - Dew-point/frost-point temperature in °F

Range of Validity:

Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\,\text{lb}_w/\text{lb}_a$

Comments:

Dew-point temperature $t_d = t_s(p_{H2O})$ for $t \ge 32°\text{F}$ (saturation temperature of water in humid air)

$t_d = t_{sub}(p_{H2O})$ for $t \le 32°\text{F}$ (sublimation temperature of water in humid air)

Result for Wrong Input Values:

td_pW_HAP_IP = -1000

References:

$t_s(p_{H2O})$	for $t_d \ge 32°\text{F}$	IAPWS-IF97 [7], [8]
$t_{sub}(p_{H2O})$	for $t_d \le 32°\text{F}$	IAPWS-08 [11]
p_{H2O}		Herrmann et. al. [1], [2]

Saturation Temperature t_s = f(p,p_{H2O})

Function Name:

ts_ppH2O_HAP_IP

Fortran Program:

REAL*8 FUNCTION   TS_PPH2O_HUAIRPROP(P,PH2O), REAL*8 P,PH2O

Input Values:

p - Total pressure p in psi

p_{H2O} - Partial saturation pressure of water p_{H2O} in psi

Result:

ts_ppH2O_HAP_IP - Saturation temperature of water in humid air in °F

Range of Validity:

Total pressure p:	from 0.00145 psi to 1450.4 psi
Partial pressure p_{H2O} :	from 0.00145 psi to 1450.4 psi

Comments:

- Iteration of saturation temperature t_s from $p_{H2O,s} = f(p,t)$

Result for Wrong Input Values:

ts_ppH2O_HAP_IP = -1000

References:

$p_{H2O,s}$ Herrmann et. al. [1], [2]

Wet-Bulb/Ice-Bulb Temperature t_{wb} = f(p,t,W)

Function Name:

twb_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION TWB_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

twb_ptW_HAP_IP - Wet-bulb/ice-bulb temperature in °F

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- Iteration of wet-bulb temperature t_{wb} from $h^{\text{unsaturated}}(p,t,W) = h^{\text{fog}}(p,t_{wb},W)$
- Calculation for supersaturated humid air ($W > W_s$) is not possible

Result for Wrong Input Values:

twb_ptW_HAP_IP = -1000

References:

$t_{wb}(p,t,W)$ Herrmann et al. [1], [2]

Air-Specific Internal Energy *u* = f(*p*,*t*,*W*)

Function Name:

u_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION U_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

u_ptW_HAP_IP - Air-specific internal energy in $\mathrm{Btu/lb_a}$

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- Internal energy $u = h - pv$

Result for Wrong Input Values:

u_ptW_HAP_IP = -1000

References:

$u(p,t,W)$ Herrmann et al. [1], [2]

Air-Specific Volume *v* = f(*p*,*t*,*W*)

Function Name:

v_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION   V_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p	-	Total pressure p in psi
t	-	Temperature t in °F
W	-	Humidity ratio W in lb_w/lb_a

Result:

v_ptW_HAP_IP - Air-specific volume in ft^3/lb_a

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Humidity ratio W:	$0 \le W \le 10\ lb_w/lb_a$

Comments:

- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

v_ptW_HAP_IP = -1000

References:

$v(p,t,W)$ Herrmann et al. [1], [2]

Humidity Ratio from Partial Pressure of Water Vapor $W = f(p,t,p_{H2O})$

Function Name:

W_ptpH2O_HAP_IP

Fortran Program:

```
REAL*8 FUNCTION   W_PTPH2O_HUAIRPROP(P,T,PH2O), REAL*8 P,T,PH2O
```

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
p_{H2O} - Partial pressure of water p_{H2O} in psi

Result:

W_ptpH2O_HAP_IP - Humidity ratio from pressure, temperature and partial pressure of water vapor in lb_w/lb_a

Range of Validity:

Total pressure p:	from 0.00145 psi to 1450.4 psi
Temperature t :	from -225.67°F to 662°F
Partial pressure p_{H2O} :	from 0.00145 psi to 1450.4 psi

Comments:

- Iteration of humidity ratio W from $p_{H2O} = f(p,t,W)$
- Result for supersaturated humid air is W_s

Result for Wrong Input Values:

W_ptpH2O_HAP_IP = -1000

References:

$p_{H2O}(p,t,W)$ Herrmann et al. [1], [2]

Humidity Ratio from Relative Humidity $W = f(p, t, \varphi)$

Function Name:

W_ptphi_HAP_IP

Fortran Program:

```
REAL*8 FUNCTION   W_PTPHI_HUAIRPROP(P,T,PHI), REAL*8 P,T,PHI
```

Input Values:

p - Total pressure p in psi

t - Temperature t in °F

φ - Relative humidity (decimal ratio)

Result:

W_ptphi_HAP_IP - Humidity ratio from pressure, temperature and relative humidity in lb_w/lb_a

Range of Validity:

Temperature t:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi
Relative humidity φ:	$0 \le \varphi \le 1$

Comments:

- Iteration of humidity ratio W from $\varphi = f(p, t, W)$

Result for Wrong Input Values:

W_ptphi_HAP_IP = -1000

References:

$\varphi(p, t, W)$ Herrmann et al. [1], [2]

Humidity Ratio from Dew-Point Temperature $W = f(p,t_d)$

Function Name:

W_ptd_HAP_IP

Fortran Program:

```
REAL*8 FUNCTION   W_PTD_HUAIRPROP(P,TD), REAL*8 P,TD
```

Input Values:

p - Total pressure p in psi
t_d - Dew-point temperature t_d in °F

Result:

W_ptd_HAP_IP - Humidity ratio from pressure and dew-point temperature in lb_w/lb_a

Range of Validity:

Dew point temperature t_d:	from -225.67°F to 662°F
Total pressure p:	from 0.00145 psi to 1450.4 psi

Comments:

- Iteration of humidity ratio W from $t_d = f(p,W)$

Result for Wrong Input Values:

W_ptd_HAP_IP = -1000

References:

$t_d(p,W)$ Herrmann et al. [1], [2]

Humidity Ratio from Wet-Bulb Temperature $W = f(p,t,t_{wb})$

Function Name:

W_pttwb_HAP_IP

Fortran Program:

REAL*8 FUNCTION W_PTTWB_HUAIRPROP(P,T,TWB), REAL*8 P,T,TWB

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
t_{wb} - Wet-bulb temperature in °F

Result:

W_pttwb_HAP_IP - Humidity ratio from pressure, temperature and wet-bulb temperature in lb_w/lb_a

Range of Validity:

Total pressure p:	from 0.00145 psi to 1450.4 psi
Temperature t:	from -225.67°F to 662°F
Wet-bulb temperature t_{wb}:	from -225.67°F to 662°F

Comments:

- Iteration of humidity ratio W from $t_{wb} = f(p,t,W)$
- Calculation for supersaturated humid air ($W > W_s$) is not possible

Result for Wrong Input Values:

W_pttwb_HAP_IP = -1000

References:

$t_{wb}(p,t,W)$ Herrmann et al. [1], [2]

Saturation Humidity Ratio W_s = f(p,t)

Function Name:

Ws_pt_HAP_IP

Fortran Program:

REAL*8 FUNCTION WS_PT_HUAIRPROP(P,T), REAL*8 P,T

Input Values:

p	-	Total pressure p in psi
t	-	Temperature t in °F

Result:

Ws_pt_HAP_IP - Saturation humidity ratio in lb_w/lb_a

Range of Validity:

Total pressure p:	from 0.00145 psi to 1450.4 psi
Temperature t:	from -225.67°F to 662°F

Comments:

- Calculation of saturation humidity ratio W_s from $W_s = \frac{M_{H2O}}{M_a} \frac{p_{H2O,s}}{\left(p - p_{H2O,s}\right)}$

Result for Wrong Input Values:

Ws_pt_HAP_IP = -1000

References:

$p_{H2O,s}$ Herrmann et al. [1], [2]

Mass Fraction of Air ξ_{Air} = f(W)

Function Name:

XiAir_W_HAP_IP

Fortran Program:

REAL*8 FUNCTION XIAIR_W_HUAIRPROP(W), REAL*8 W

Input Values:

W - Humidity ratio W in lb_w/lb_a

Result:

XiAir_W_HAP_IP - Mass fraction of (dry) air in humid air in lb_a/lb

Range of Validity:

Humidity ratio W: $0 \leq W \leq 10\,lb_w/lb_a$

Comments:

- Mass fraction of (dry) air $\xi_{Air} = 1 - \xi_{H2O} = 1 - \frac{W}{1+W}$

Result for Wrong Input Values:

XiAir_W_HAP_IP = -1000

References:

$\xi_{Air}(W)$ Herrmann et al. [1], [2]

Mass Fraction of Water Vapor in Humid Air ξ_{H2O} = f(W)

Function Name:

XiH2O_W_HAP_IP

Fortran Program:

REAL*8 FUNCTION XIH2O_W_HUAIRPROP(W), REAL*8 W

Input Values:

W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

XiH2O_W_HAP_IP - Mass fraction of water vapor in humid air in $\mathrm{lb_w/lb}$

Range of Validity:

Humidity ratio W:	$0 \le W \le 10\,\mathrm{lb_w/lb_a}$

Comments:

- Mass fraction of water $\xi_{\mathrm{H2O}} = \dfrac{W}{1+W}$

Result for Wrong Input Values:

XiH2O_W_HAP_IP = -1000

References:

$\xi_{\mathrm{H2O}}(W)$ Herrmann et al. [1], [2]

Compression Factor *Z* = f(*p*,*t*,*W*)

Function Name:

Z_ptW_HAP_IP

Fortran Program:

REAL*8 FUNCTION Z_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in psi
t - Temperature t in °F
W - Humidity ratio W in $\mathrm{lb_w/lb_a}$

Result:

Z_ptW_HAP_IP - Compression factor (decimal ratio)

Range of Validity:

Total pressure *p*:	from 0.00145 psi to 1450.4 psi
Temperature *t*:	from -225.67°F to 662°F
Humidity ratio *W*:	$0 \le W \le W_s$

Comments:

- Compression factor $Z = 1 + \dfrac{B_m}{\bar{v}} + \dfrac{C_m}{\bar{v}^2}$

  with $\bar{v} = \dfrac{M}{\rho} = \dfrac{M v}{1+W}$

  and M is the molar mass of humid air

- Calculation for supersaturated humid air ($W > W_s$) is not possible

Result for Wrong Input Values:

Z_ptW_HAP_IP = -1000

References:

$B_m(t,W), C_m(t,W)$	Herrmann et al. [1], [2]
$\rho(p,t,W), v(p,t,W)$	Herrmann et al. [1], [2]

3.2 Functions for Steam and Water for Temperatures $t \geq 32°F$

Specific Enthalpy of Liquid Water $h_{\text{liq}} = \mathrm{f}(p,t)$

Function Name:

hliq_pt_97_IP

Fortran Program:

```
REAL*8 FUNCTION   HLIQ_PT_97(P,T), REAL*8 P,T
```

Input Values:

p - Pressure *p* in psi
t - Temperature *t* in °F

Result:

hliq_pt_97_IP - Specific enthalpy of liquid water in Btu/lb

Range of Validity:

Pressure *p*: from $p_{\text{s}}(32°\text{F}) = 0.08865$ psi to 1450.4 psi

Temperature *t*: from 32°F to 662°F

Comments:

- Specific enthalpy of liquid water $h_{\text{liq}} = h^{97}(p,t)$ (Region 1)

Result for Wrong Input Values:

hliq_pt_97_IP = -1000

References:

$h^{97}(p,t)$ IAPWS-IF97 [7], [8]

Specific Enthalpy of Saturated Liquid Water $h_{\text{liq,s}} = f(t)$

Function Name:

hliqs_t_97_IP

Fortran Program:

REAL*8 FUNCTION HLIQS_T_97(T), REAL*8 T

Input Values:

t - Temperature *t* in °F

Result:

hliqs_t_97_IP - Specific enthalpy of saturated liquid water in Btu/lb

Range of Validity:

Temperature *t*: from 32°F to 662°F

Comments:

- Specific enthalpy of liquid water $h_{\text{liq,s}} = h^{97}(p_{\text{s}}, t)$ (Region 1)
  with $p_{\text{s}} = p_{\text{s}}^{97}(t)$

Result for Wrong Input Values:

hliqs_t_97_IP = -1000

References:

$h^{97}(p,t)$, $p_{\text{s}}^{97}(t)$ IAPWS-IF97 [7], [8]

Specific Enthalpy of Saturated Water Vapor $h_{\mathbf{vap,s}}$ = f(t)

Function Name:

hvaps_t_97_IP

Fortran Program:

REAL*8 FUNCTION HVAPS_T_97(T), REAL*8 T

Input Values:

t - Temperature *t* in °F

Result:

hvaps_t_97_IP - Specific enthalpy of saturated water vapor in Btu/lb

Range of Validity:

Temperature *t*: from 32°F to 662°F

Comments:

- Specific enthalpy of saturated water vapor $h_{\mathrm{vap,s}} = h^{97}(p_{\mathrm{s}}, t)$ (Region 2)
  with $p_{\mathrm{s}} = p_{\mathrm{s}}^{97}(t)$

Result for Wrong Input Values:

hvaps_t_97_IP = -1000

References:

$h^{97}(p,t)$, $p_{\mathrm{s}}^{97}(t)$ IAPWS-IF97 [7], [8]

Saturation Pressure of Water p_s = f(t)

Function Name:

ps_t_97_IP

Fortran Program:

```
REAL*8 FUNCTION   PS_T_97(T), REAL*8 T
```

Input Values:

t - Temperature t in °F

Result:

ps_t_97_IP - Saturation pressure of water in psi

Range of Validity:

Temperature t: from 32°F to 662°F

Comments:

- Saturation pressure of water $p_s = p_s^{97}(t)$ (Region 4)

Result for Wrong Input Values:

ps_t_97_IP -1000

References:

$p_s^{97}(t)$	IAPWS-IF97 [7], [8]

Specific Entropy of Liquid Water $s_{\text{liq}} = \mathrm{f}(p, t)$

Function Name:

sliq_pt_97_IP

Fortran Program:

REAL*8 FUNCTION   SLIQ_PT_97(P,T), REAL*8 P,T

Input Values:

p - Pressure *p* in psi
t - Temperature *t* in °F

Result:

sliq_pt_97_IP - Specific entropy of liquid water in Btu/(lb °R)

Range of Validity:

Pressure *p*:	from $p_{\text{s}}(32°\text{F}) = 0.08865$ psi to 1450.4 psi
Temperature *t*:	from 32°F to 662°F

Comments:

- Specific entropy of liquid water $s_{\text{liq}} = s^{97}(p,t)$ (Region 1)

Result for Wrong Input Values:

sliq_pt_97_IP = -1000

References:

$s^{97}(p,t)$ IAPWS-IF97 [7], [8]

Specific Entropy of Saturated Liquid Water $s_{\mathrm{liq,s}} = \mathrm{f}(t)$

Function Name:

sliqs_t_97_IP

Fortran Program:

REAL*8 FUNCTION   SLIQS_T_97(T), REAL*8 T

Input Values:

t - Temperature t in °F

Result:

sliqs_t_97_IP - Specific entropy of saturated liquid water in Btu/(lb °R)

Range of Validity:

Temperature t: from 32°F to 662°F

Comments:

- Specific entropy of liquid water $s_{\mathrm{liq,s}} = s^{97}(p_{\mathrm{s}}, t)$ (Region 1)
  with $p_{\mathrm{s}} = p_{\mathrm{s}}^{97}(t)$

Result for Wrong Input Values:

sliqs_t_97_IP = -1000

References:

$s^{97}(p,t)$, $p_{\mathrm{s}}^{97}(t)$ IAPWS-IF97 [7], [8]

Specific Entropy of Saturated Water Vapor $s_{\mathrm{vap,s}}$ = f(t)

Function Name:

svaps_t_97_IP

Fortran Program:

```
REAL*8 FUNCTION   SVAPS_T_97(T), REAL*8 T
```

Input Values:

t - Temperature *t* in °F

Result:

svaps_t_97_IP - Specific entropy of saturated water vapor in Btu/(lb °R)

Range of Validity:

Temperature *t*: from 32°F to 662°F

Comments:

- Specific entropy of saturated water vapor $s_{\mathrm{vap,s}} = s^{97}(p_{\mathrm{s}}, t)$ (Region 2)
  with $p_{\mathrm{s}} = p_{\mathrm{s}}^{97}(t)$

Result for Wrong Input Values:

svaps_t_97_IP = -1000

References:

$s^{97}(p,t)$, $p_{\mathrm{s}}^{97}(t)$ IAPWS-IF97 [7], [8]

Saturation Temperature of Water t_s = f(p)

Function Name:

ts_p_97_IP

Fortran Program:

```
REAL*8 FUNCTION   TS_P_97(P), REAL*8 P
```

Input Values:

p - Pressure *p* in psi

Result:

ts_p_97_IP - Saturation temperature of water in °F

Range of Validity:

Pressure *p*: from 0.08865 psi to 1450.4 psi

Comments:

- Saturation temperature of water $t_s = t_s^{97}(p)$ (Region 4)

Result for Wrong Input Values:

ts_p_97_IP = -1000

References:

$t_s^{97}(p)$ IAPWS-IF97 [7], [8]

Specific Volume of Liquid Water $v_{\text{liq}} = f(p,t)$

Function Name:

vliq_pt_97_IP

Fortran Program:

```
REAL*8 FUNCTION   VLIQ_PT_97(P,T), REAL*8 P,T
```

Input Values:

p - Pressure p in psi
t - Temperature t in °F

Result:

vliq_pt_97_IP - Specific volume of liquid water in ft^3/lb

Range of Validity:

Pressure p:	from $p_s(32°\text{F}) = 0.08865$ psi to 1450.4 psi
Temperature t:	from 32°F to 662°F

Comments:

- Specific volume of liquid water $v_{\text{liq}} = v^{97}(p,t)$ (Region 1)

Result for Wrong Input Values:

vliq_pt_97_IP = -1000

References:

$v^{97}(p,t)$ IAPWS-IF97 [7], [8]

Specific Volume of Saturated Liquid Water $v_{\mathrm{liq,s}} = \mathrm{f}(t)$

Function Name:

vliqs_t_97_IP

Fortran Program:

REAL*8 FUNCTION VLIQS_T_97(T), REAL*8 T

Input Values:

t - Temperature *t* in °F

Result:

vliqs_t_97_IP - Specific volume of saturated liquid water in $\mathrm{ft}^3/\mathrm{lb}$

Range of Validity:

Temperature *t*: from 32°F to 662°F

Comments:

- Specific volume of liquid water $v_{\mathrm{liq,s}} = v^{97}(p_\mathrm{s}, t)$ (Region 1)
  with $p_\mathrm{s} = p_\mathrm{s}^{97}(t)$

Result for Wrong Input Values:

vliqs_t_97_IP = -1000

References:

$v^{97}(p,t)$, $p_\mathrm{s}^{97}(t)$ IAPWS-IF97 [7], [8]

Specific Volume of Saturated Water Vapor $v_{\text{vap,s}} = \text{f}(t)$

Function Name:

vvaps_t_97_IP

Fortran Program:

REAL*8 FUNCTION VVAPS_T_97(T), REAL*8 T

Input Values:

t - Temperature *t* in °F

Result:

vvaps_t_97_IP - Specific volume of saturated water vapor in ft^3/lb

Range of Validity:

Temperature *t*: from 32°F to 662°F

Comments:

- Specific volume of saturated water vapor $v_{\text{vap,s}} = v^{97}(p_{\text{s}}, t)$ (Region 2) with $p_{\text{s}} = p_{\text{s}}^{97}(t)$

Result for Wrong Input Values:

vvaps_t_97_IP = -1000

References:

$v^{97}(p,t)$, $p_{\text{s}}^{97}(t)$ IAPWS-IF97 [7], [8]

3.3 Functions for Steam and Ice for Temperatures *t* ≤ 32°F

Specific Enthalpy of Saturated Ice $h_{\text{ice,sub}} = \text{f}(t)$

Function Name:

hicesub_t_06_IP

Fortran Program:

REAL*8 FUNCTION HICESUB_T_06(T), REAL*8 T

Input Values:

t - Temperature *t* in °F

Result:

hicesub_t_06_IP - Specific enthalpy of saturated ice in Btu/lb

Range of Validity:

Temperature *t*: from -225.67°F to 32°F

Comments:

- Specific enthalpy of saturated ice $h_{\text{ice,sub}} = h^{06}(p_{\text{sub}}, t)$
  with $p_{\text{sub}} = p_{\text{sub}}^{08}(t)$

Result for Wrong Input Values:

hicesub_t_06_IP = -1000

References:

$h^{06}(p,t)$	IAPWS-06 [10]
$p_{\text{sub}}^{08}(t)$	IAPWS-08 [11]

Specific Enthalpy of Saturated Water Vapor $h_{\mathrm{vap,sub}} = \mathrm{f}(t)$

Function Name:

hvapsub_t_95_IP

Fortran Program:

REAL*8 FUNCTION HVAPSUB_T_95(T), REAL*8 T

Input Values:

t - Temperature *t* in °F

Result:

hvapsub_t_95_IP - Specific enthalpy of saturated water vapor in Btu/lb

Range of Validity:

Temperature *t*: from -225.67°F to 32°F

Comments:

- Specific enthalpy of saturated water vapor $h_{\mathrm{vap,sub}} = h^{95}(p_{\mathrm{sub}}, t)$

  with $p_{\mathrm{sub}} = p_{\mathrm{sub}}^{08}(t)$

Result for Wrong Input Values:

hvapsub_t_95_IP = -1000

References:

$h^{95}(p,t)$	IAPWS-95 [5], [6]
$p_{\mathrm{sub}}^{08}(t)$	IAPWS-08 [11]

Melting Pressure of Ice $p_{\mathrm{mel}} = \mathrm{f}(t)$

Function Name:

pmel_t_08_IP

Fortran Program:

REAL*8 FUNCTION   PMEL_T_08 (T), REAL*8 T

Input Values:

t - Temperature *t* in °F

Result:

pmel_t_08_IP - Melting pressure of ice in psi

Range of Validity:

Temperature *t*: from -7.573°F to 32°F

Result for Wrong Input Values:

pmel_t_08_IP = -1000

References:

$p_{\mathrm{mel}}^{08}(t)$ IAPWS-08 [11]

Sublimation Pressure of Ice $p_{\mathrm{sub}} = \mathrm{f}(t)$

Function Name:

psub_t_08_IP

Fortran Program:

REAL*8 FUNCTION   PSUB_T_08 (T), REAL*8 T

Input Values:

t - Temperature *t* in °F

Result:

psub_t_08_IP - Sublimation pressure of ice in psi

Range of Validity:

Temperature *t*: from -225.67°F to 32°F

Result for Wrong Input Values:

psub_t_08_IP = -1000

References:

$p_{\mathrm{sub}}^{08}(t)$	IAPWS-08 [11]

Specific Entropy of Saturated Ice $s_{\text{ice,sub}}$ = f(t)

Function Name:

sicesub_t_06_IP

Fortran Program:

REAL*8 FUNCTION SICESUB_T_06(T), REAL*8 T

Input Values:

t - Temperature *t* in °F

Result:

sicesub_t_06_IP - Specific entropy of saturated ice in Btu/(lb °R)

Range of Validity:

Temperature *t*: from -225.67°F to 32°F

Comments:

- Specific entropy of saturated ice $s_{\text{ice,sub}} = s^{06}(p_{\text{sub}}, t)$ with $p_{\text{sub}} = p_{\text{sub}}^{08}(t)$

Result for Wrong Input Values:

sicesub_t_06_IP = -1000

References:

$s^{06}(p,t)$ IAPWS-06 [10]

$p_{\text{sub}}^{08}(t)$ IAPWS-08 [11]

Specific Entropy of Saturated Water Vapor $s_{\mathrm{vap,sub}}$ = f(t)

Function Name:

svapsub_t_95_IP

Fortran Program:

REAL*8 FUNCTION SVAPSUB_T_95(T), REAL*8 T

Input Values:

t - Temperature t in °F

Result:

svapsub_t_95_IP - Specific entropy of saturated water vapor in Btu/(lb °R)

Range of Validity:

Temperature t: from -225.67°F to 32°F

Comments:

- Specific entropy of saturated water vapor $s_{\mathrm{vap,sub}} = s^{95}(p_{\mathrm{sub}}, t)$
  with $p_{\mathrm{sub}} = p_{\mathrm{sub}}^{08}(t)$

Result for Wrong Input Values:

svapsub_t_95_IP = -1000

References:

$s^{95}(p,t)$	IAPWS-95 [7], [8]
$p_{\mathrm{sub}}^{08}(t)$	IAPWS-08 [11]

Melting Temperature of Ice $t_{\mathrm{mel}} = \mathrm{f}(p)$

Function Name:

tmel_p_08_IP

Fortran Program:

```
REAL*8 FUNCTION   TMEL_P_08(P), REAL*8 P
```

Input Values:

p - Pressure p in psi

Result:

tmel_p_08_IP - Melting temperature of ice in °F

Range of Validity:

Pressure p:	from $p_\mathrm{s}\,(32^\circ\mathrm{F}) = 0.08865$ psi to 1450.4 psi

Result for Wrong Input Values:

tmel_p_08_IP = -1000

References:

$t_{\mathrm{mel}}^{08}(p)$	IAPWS-08 [11]

Sublimation Temperature of Ice t_{sub} = f(p)

Function Name:

tsub_p_08_IP

Fortran Program:

REAL*8 FUNCTION   TSUB_P_08(P), REAL*8 P

Input Values:

p   -   Pressure *p* in psi

Result:

tsub_p_08_IP - Sublimation temperature of ice in °F

Range of Validity:

Pressure *p*:   from $p_{subl}(-225.67°F) = 1.7407 \times 10^{-12}$ psi to $p_{subl}(32°F) = 0.08865$ psi

Result for Wrong Input Values:

tsub_p_08_IP = -1000

References:

$t_{sub}^{08}(p)$ IAPWS-08 [11]

Specific Volume of Saturated Ice $v_{\text{ice,sub}} = f(t)$

Function Name:

vicesub_t_06_IP

Fortran Program:

REAL*8 FUNCTION VICESUB_T_06(T), REAL*8 T

Input Values:

t - Temperature *t* in °F

Result:

vicesub_t_06_IP - Specific volume of saturated ice in ft^3/lb

Range of Validity:

Temperature *t*: from -225.67°F to 32°F

Comments:

- Specific volume of saturated ice $v_{\text{ice,sub}} = v^{06}(p_{\text{sub}}, t)$

  with $p_{\text{sub}} = p_{\text{sub}}^{08}(t)$

Result for Wrong Input Values:

vicesub_t_06_IP = -1000

References:

$v^{06}(p,t)$	IAPWS-06 [10]
$p_{\text{sub}}^{08}(t)$	IAPWS-08 [11]

Specific Volume of Saturated Water Vapor $v_{\mathrm{vap,sub}} = \mathrm{f}(t)$

Function Name:

vvapsub_t_95_IP

Fortran Program:

```
REAL*8 FUNCTION   VVAPSUB_T_95(T), REAL*8 T
```

Input Values:

t - Temperature *t* in °F

Result:

vvapsub_t_95_IP - Specific volume of saturated water vapor in ft^3/lb

Range of Validity:

Temperature *t*: from -225.67°F to 32°F

Comments:

- Specific volume of saturated water vapor $v_{\mathrm{vap,sub}} = v^{95}(p_{\mathrm{sub}}, t)$

  with $p_{\mathrm{sub}} = p_{\mathrm{sub}}^{08}(t)$

Result for Wrong Input Values:

vvapsub_t_95_IP = -1000

References:

$v^{95}(p,t)$	IAPWS-95 [7], [8]
$p_{\mathrm{sub}}^{08}(t)$	IAPWS-08 [11]

KCE ThermoFluidProperties

KCE-ThermoFluidProperties
www.thermofluidprop.com

Hochschule Zittau/Görlitz
UNIVERSITY OF APPLIED SCIENCES

4. Property Libraries for Calculating Heat Cycles, Boilers, Turbines and Refrigerators

Water and Steam

Library LibIF97

- Industrial Formulation IAPWS-IF97 (Revision 2007)
- Supplementary Standards IAPWS-IF97-S01, -S03rev, -S04, and -S05
- IAPWS Revised Advisory Note No. 3 on Thermodynamic Derivatives (2008)

Library LibIF97_META

- Industrial Formulation IAPWS-IF97 (Revision 2007) for metastable steam

Humid Combustion Gas Mixtures

Library LibHuGas

Model: Ideal mixture of the real fluids:

CO_2 - Span, Wagner H_2O - IAPWS-95
O_2 - Schmidt, Wagner N_2 - Span et al.
Ar - Tegeler et al.

and of the ideal gases:

SO_2, CO, Ne
(Scientific Formulation of Bücker et al.)

Consideration of:

- Dissociation from VDI 4670
- Poynting effect

Humid Air

Library LibHuAir

Model: Ideal mixture of the real fluids:

- Dry air from Lemmon et al.
- Steam, water and ice from IAPWS-IF97 and IAPWS-06

Consideration of:

- Condensation and freezing of steam
- Dissociation from VDI 4670
- Poynting effect from ASHRAE RP-1485

Extremely Fast Property Calculations

Spline-Based Table Look-up Method (SBTL)

Library LibSBTL_IF97
Library LibSBTL_95
Library LibSBTL_HuAir

For steam, water, humid air, carbon dioxide and other fluids and mixtures according IAPWS Guideline 2015 for Computational Fluid Dynamics (CFD), real-time and non-stationary simulations

Carbon Dioxide Including Dry Ice

Library LibCO2

Formulation of Span and Wagner (1996)

Seawater

Library LibSeaWa

IAPWS Industrial Formulation 2013

Ice

Library LibICE

Ice from IAPWS-06, Melting and sublimation pressures from IAPWS-08, Water from IAPWS-IF97, Steam from IAPWS-95 and -IF97

Ideal Gas Mixtures

Library LibIdGasMix

Model: Ideal mixture of the ideal gases:

Ar
Ne
N_2
O_2
CO
CO_2
Air
NO
H_2O
SO_2
H_2
H_2S
OH
He
F_2
NH_3
Methane
Ethane
Ethylene
Propylene
Propane
Iso-Butane
n-Butane
Benzene
Methanol

Consideration of:

- Dissociation from the VDI Guideline 4670

Library LibIDGAS

Model: Ideal gas mixture from VDI Guideline 4670

Consideration of:

- Dissociation from the VDI Guideline 4670

Humid Air

Library ASHRAE LibHuAirProp

Model: Virial equation from ASHRAE Report RP-1485 for real mixture of the real fluids:
- Dry air
- Steam

Consideration of:

- Enhancement of the partial saturation pressure of water vapor at elevated total pressures

www.ashrae.org/bookstore

Dry Air Including Liquid Air

Library LibRealAir

Formulation of Lemmon et al. (2000)

Refrigerants

Ammonia

Library LibNH3

Formulation of Tillner-Roth et al. (1993)

R134a

Library LibR134a

Formulation of Tillner-Roth and Baehr (1994)

Iso-Butane

Library LibButane_Iso

Formulation of Bücker and Wagner (2006)

n-Butane

Library LibButane_n

Formulation of Bücker and Wagner (2006)

Mixtures for Absorption Processes

Ammonia/Water Mixtures

Library LibAmWa

IAPWS Guideline 2001 of Tillner-Roth and Friend (1998)

Helmholtz energy equation for the mixing term

(also useable for calculating the Kalina Cycle)

Water/Lithium Bromide Mixtures

Library LibWaLi

Formulation of Kim and Infante Ferreira (2004)

Gibbs energy equation for the mixing term

Liquid Coolants

Liquid Secondary Refrigerants

Library LibSecRef

Liquid solutions of water with

$C_2H_6O_2$	Ethylene glycol
$C_3H_8O_2$	Propylene glycol
C_2H_5OH	Ethanol
CH_3OH	Methanol
$C_3H_8O_3$	Glycerol
K_2CO_3	Potassium carbonate
$CaCl_2$	Calcium chloride
$MgCl_2$	Magnesium chloride
NaCl	Sodium chloride
$C_2H_3KO_2$	Potassium acetate
$CHKO_2$	Potassium formate
LiCl	Lithium chloride
NH_3	Ammonia

Formulation of the International Institute of Refrigeration (IIR 2010)

Ethanol

Library LibC2H5OH

Formulation of
Schroeder et al. (2014)

Methanol

Library LibCH3OH

Formulation of
de Reuck and Craven (1993)

Propane

Library LibPropane

Formulation of
Lemmon et al. (2009)

Siloxanes as ORC Working Fluids

Octamethylcyclotetrasiloxane	$C_8H_{24}O_4Si_4$	**Library LibD4**
Decamethylcyclopentasiloxane	$C_{10}H_{30}O_5Si_5$	**Library LibD5**
Tetradecamethylhexasiloxane	$C_{14}H_{42}O_5Si_6$	**Library LibMD4M**
Hexamethyldisiloxane	$C_6H_{18}OSi_2$	**Library LibMM**

Formulation of Colonna et al. (2006)

Dodecamethylcyclohexasiloxane	$C_{12}H_{36}O_6Si_6$	**Library LibD6**
Decamethyltetrasiloxane	$C_{10}H_{30}O_3Si_4$	**Library LibMD2M**
Dodecamethylpentasiloxane	$C_{12}H_{36}O_4Si_5$	**Library LibMD3M**
Octamethyltrisiloxane	$C_8H_{24}O_2Si_3$	**Library LibMDM**

Formulation of Colonna et al. (2008)

Nitrogen and Oxygen

Libraries
LibN2 and LibO2

Formulations of Span et al. (2000)
and Schmidt and Wagner (1985)

Hydrogen

Library LibH2

Formulation of
Leachman et al. (2009)

Helium

Library LibHe

Formulation of
Arp et al. (1998)

Hydrocarbons

Decane	$C_{10}H_{22}$	**Library LibC10H22**
Isopentane	C_5H_{12}	**Library LibC5H12_Iso**
Neopentane	C_5H_{12}	**Library LibC5H12_Neo**
Isohexane	C_6H_{14}	**Library LibC6H14**
Toluene	C_7H_8	**Library LibC7H8**

Formulation of Lemmon and Span (2006)

Further Fluids

Carbon monoxide	CO	**Library LibCO**
Carbonyl sulfide	COS	**Library LibCOS**
Hydrogen sulfide	H_2S	**Library LibH2S**
Nitrous oxide	N_2O	**Library LibN2O**
Sulfur dioxide	SO_2	**Library LibSO2**
Acetone	C_3H_6O	**Library LibC3H6O**

Formulation of Lemmon and Span (2006)

KCE
ThermoFluidProperties

For more information please contact:

KCE-ThermoFluidProperties UG
Prof. Dr. Hans-Joachim Kretzschmar
Haager Weg 6
92224 Amberg, Germany

Internet: www.thermofluidprop.com
Email: info@thermofluidprop.com
Phone: +49-9621-1762047
Mobile: +49-172-7914607
Fax: +49-3222-1095810

The following thermodynamic and transport properties can be calculated[a]:

Thermodynamic Properties

- Vapor pressure p_s
- Saturation temperature T_s
- Density ρ
- Specific volume v
- Enthalpy h
- Internal energy u
- Entropy s
- Exergy e
- Isobaric heat capacity c_p
- Isochoric heat capacity c_v
- Isentropic exponent κ
- Speed of sound w
- Surface tension σ

Transport Properties

- Dynamic viscosity η
- Kinematic viscosity ν
- Thermal conductivity λ
- Prandtl number Pr
- Thermal diffusivity a

Backward Functions

- $T, v, s\ (p,h)$
- $T, v, h\ (p,s)$
- $p, T, v\ (h,s)$
- $p, T\ (v,h)$
- $p, T\ (v,u)$

Thermodynamic Derivatives

- Partial derivatives used in process modeling can be calculated.

---

[a] Not all of these property functions are available in all property libraries.

www.thermofluidprop.com

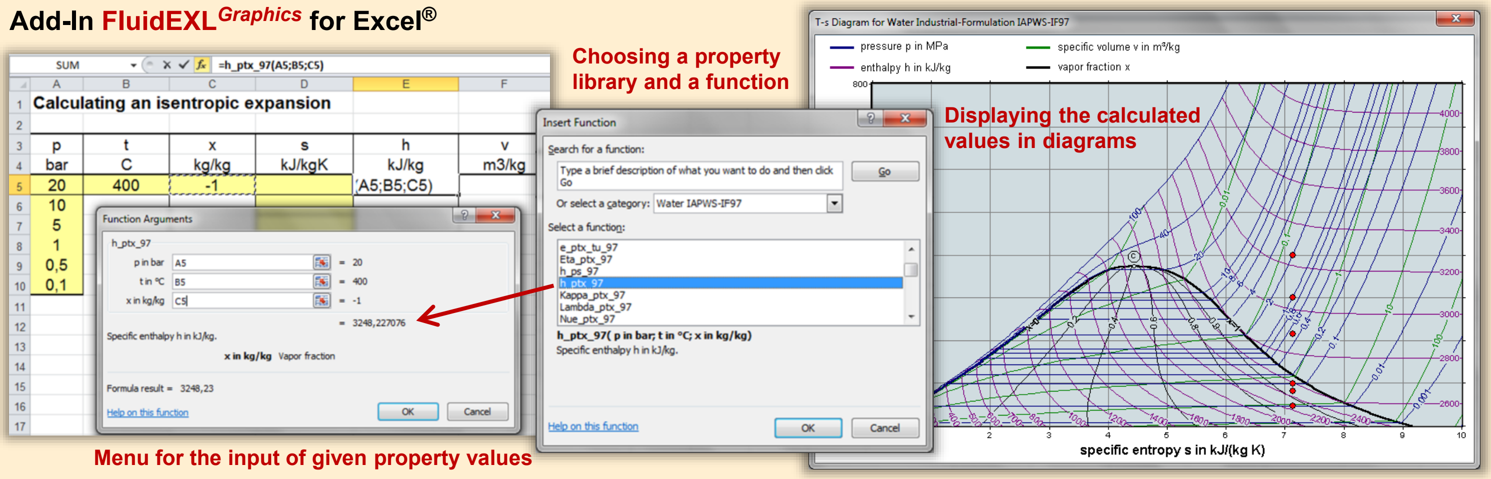
KCE ThermoFluidProperties

KCE-ThermoFluidProperties
www.thermofluidprop.com

Hochschule Zittau/Görlitz
UNIVERSITY OF APPLIED SCIENCES

Property Software for Calculating Heat Cycles, Boilers, Turbines and Refrigerators

Add-In FluidEXL*Graphics* for Excel®



=h_ptx_97(A5;B5;C5)

	A	B	C	D	E	F
1	Calculating an isentropic expansion					
2						
3	p	t	x	s	h	v
4	bar	C	kg/kg	kJ/kgK	kJ/kg	m3/kg
5	20	400	-1		(A5;B5;C5)	
6	10					
7	5					
8	1					
9	0,5					
10	0,1					

Function Arguments

h_ptx_97
p in bar: A5 = 20
t in °C: B5 = 400
x in kg/kg: C5 = -1
= 3248,227076
Specific enthalpy h in kJ/kg.
x in kg/kg Vapor fraction
Formula result = 3248,23
Help on this function

Insert Function

Search for a function: Type a brief description of what you want to do and then click Go
Or select a category: Water IAPWS-IF97
Select a function:
- e_ptx_tu_97
- Eta_ptx_97
- h_ps_97
- h_ptx_97
- Kappa_ptx_97
- Lambda_ptx_97
- Nue_ptx_97

h_ptx_97(p in bar; t in °C; x in kg/kg)
Specific enthalpy h in kJ/kg.
Help on this function

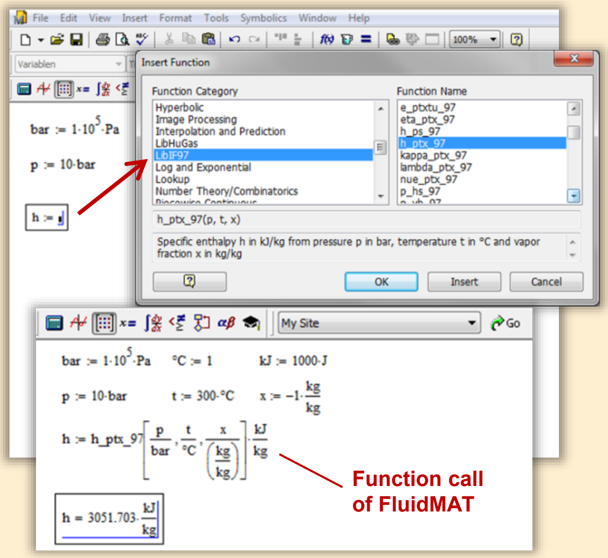
Choosing a property library and a function

Displaying the calculated values in diagrams

Menu for the input of given property values

Add-On FluidMAT for Mathcad®
Add-On FluidPRIME for Mathcad Prime®

The property libraries can be used in Mathcad® and Mathcad Prime®.



$bar := 1 \cdot 10^5 \cdot Pa$

$p := 10 \cdot bar$

$h :=$

Insert Function

Function Category:
- Hyperbolic
- Image Processing
- Interpolation and Prediction
- LibHuGas
- LibIF97
- Log and Exponential
- Lookup
- Number Theory/Combinatorics

Function Name:
- e_ptxtu_97
- eta_ptx_97
- h_ps_97
- h_ptx_97
- kappa_ptx_97
- lambda_ptx_97
- nue_ptx_97
- p_hs_97

h_ptx_97(p, t, x)

Specific enthalpy h in kJ/kg from pressure p in bar, temperature t in °C and vapor fraction x in kg/kg

$bar := 1 \cdot 10^5 \cdot Pa \quad °C := 1 \quad kJ := 1000 \cdot J$

$p := 10 \cdot bar \quad t := 300 \cdot °C \quad x := -1 \cdot \frac{kg}{kg}$

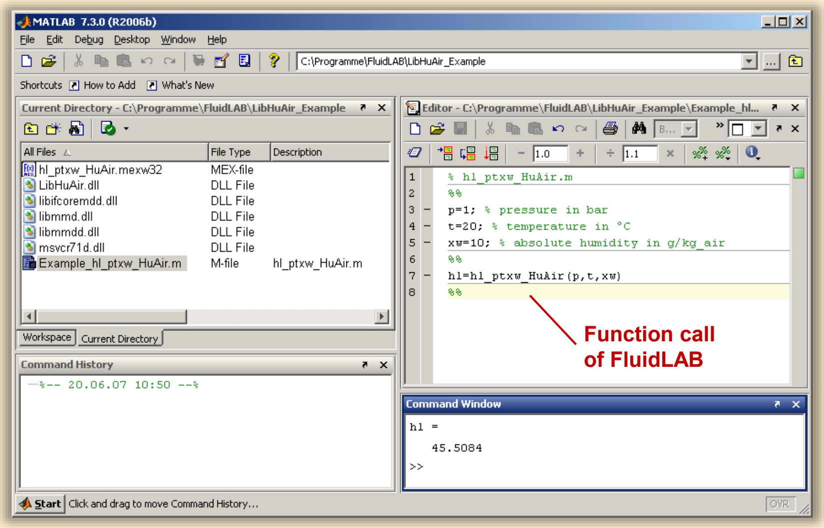
$h := h\_ptx\_97\left[\frac{p}{bar}, \frac{t}{°C}, \frac{x}{\left(\frac{kg}{kg}\right)}\right] \cdot \frac{kJ}{kg}$

$h = 3051.703 \cdot \frac{kJ}{kg}$

Function call of FluidMAT

Add-On FluidLAB for MATLAB® and SIMULINK®

Using the Add-In FluidLAB the property functions can be called in MATLAB® and SIMULINK®.



All Files	File Type	Description
hl_ptxw_HuAir.mexw32	MEX-file	
LibHuAir.dll	DLL File	
libifcoremdd.dll	DLL File	
libmmd.dll	DLL File	
libmmdd.dll	DLL File	
msvcr71d.dll	DLL File	
Example_hl_ptxw_HuAir.m	M-file	hl_ptxw_HuAir.m

```
%-- 20.06.07 10:50 --%
```

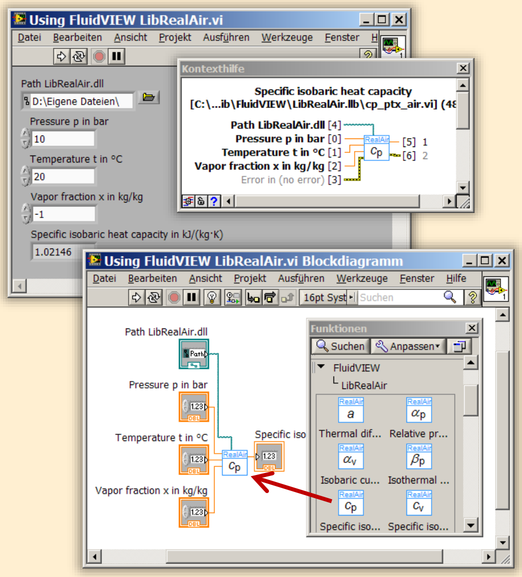
```
% hl_ptxw_HuAir.m
%%
p=1; % pressure in bar
t=20; % temperature in °C
xw=10; % absolute humidity in g/kg_air
%%
hl=hl_ptxw_HuAir(p,t,xw)
%%
```

```
hl =

   45.5084

>>
```

Function call of FluidLAB

Add-On FluidVIEW for LabVIEW™

The property functions can be calculated in LabVIEW™.

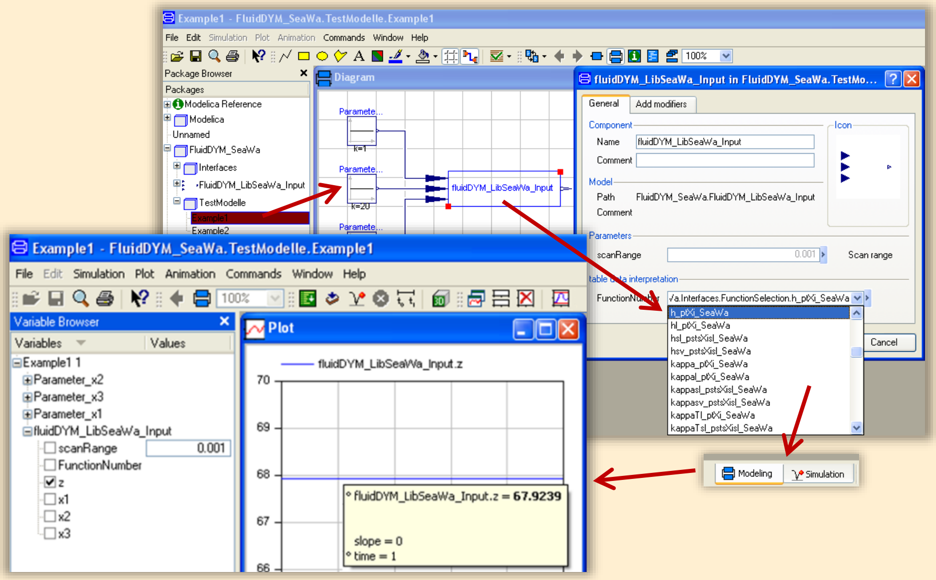


Path LibRealAir.dll: D:\Eigene Dateien\
Pressure p in bar: 10
Temperature t in °C: 20
Vapor fraction x in kg/kg: -1
Specific isobaric heat capacity in kJ/(kg·K): 1.02146

Specific isobaric heat capacity
[C:\...ib\FluidVIEW\LibRealAir.llb\cp_ptx_air.vi] (48
Path LibRealAir.dll [4]
Pressure p in bar [0]
Temperature t in °C [1]
Vapor fraction x in kg/kg [2]
Error in (no error) [3]
[5] 1
[6] 2

Add-On FluidDYM for DYMOLA® (Modelica) and SimulationX®

The property functions can be called in DYMOLA® and SimulationX®.



fluidDYM_LibSeaWa_Input in FluidDYM_SeaWa.TestMo...

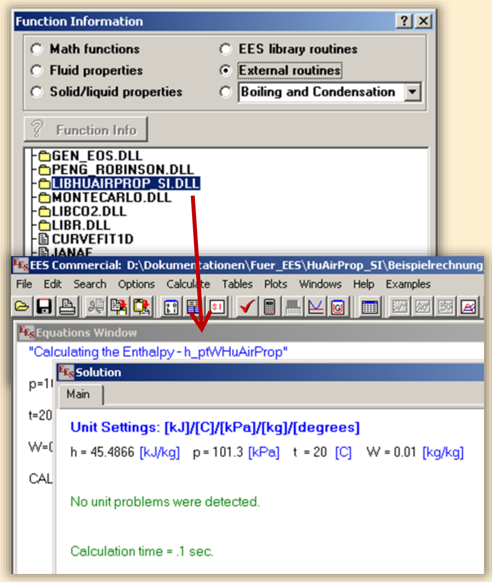
General | Add modifiers

Component
Name: fluidDYM_LibSeaWa_Input
Comment:
Model
Path: FluidDYM_SeaWa.FluidDYM_LibSeaWa_Input
Comment:
Parameters
scanRange: 0.001 Scan range
table data interpretation
FunctionNumber: √a.Interfaces.FunctionSelection.h_pXi_SeaWa
- h_pXi_SeaWa
- hl_pXi_SeaWa
- hsl_pstsXisl_SeaWa
- hsv_pstsXisl_SeaWa
- kappa_pXi_SeaWa
- kappal_pXi_SeaWa
- kappasl_pstsXisl_SeaWa
- kappasv_pstsXisl_SeaWa
- kappaTl_pXi_SeaWa
- kappaTsl_pstsXisl_SeaWa

Variables	Values
Example1 1	
Parameter_x2	
Parameter_x3	
Parameter_x1	
fluidDYM_LibSeaWa_Input	
☐ scanRange	0.001
☐ FunctionNumber	
☑ z	
☐ x1	
☐ x2	
☐ x3	

fluidDYM_LibSeaWa_Input.z
70
69
68
67
66
fluidDYM_LibSeaWa_Input.z = **67.9239**
slope = 0
time = 1

Add-On **FluidEES** for Engineering Equation Solver®



EES Commercial: D:\Dokumentationen\Fuer_EES\HuAirProp_SI\Beispielrechnung

"Calculating the Enthalpy - h_ptWHuAirProp"

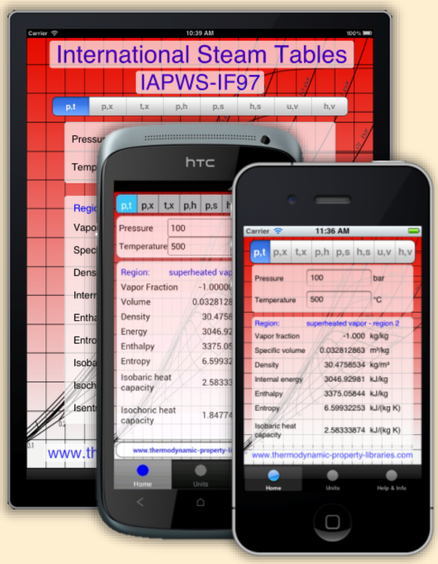
Unit Settings: [kJ]/[C]/[kPa]/[kg]/[degrees]

h = 45.4866 [kJ/kg]   p = 101.3 [kPa]   t = 20 [C]   W = 0.01 [kg/kg]

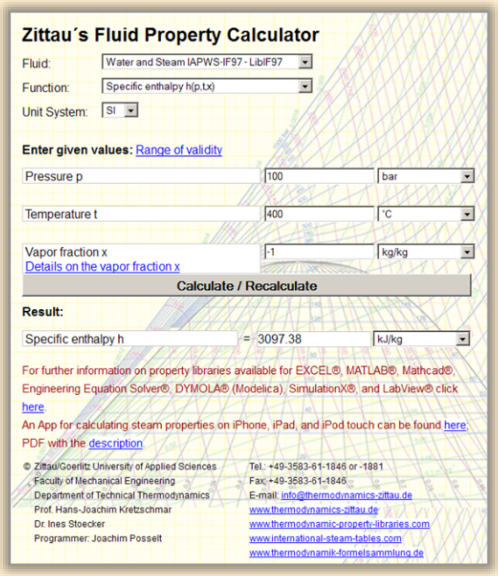
No unit problems were detected.

Calculation time = .1 sec.

App International Steam Tables for iPhone, iPad, iPod touch, Android Smartphones and Tablets



Online Property Calculator at www.thermofluidprop.com



Zittau´s Fluid Property Calculator

Fluid: Water and Steam IAPWS-IF97 - LibIF97

Function: Specific enthalpy h(p,t,x)

Unit System: SI

Enter given values: Range of validity

Pressure p: 100 bar

Temperature t: 400 °C

Vapor fraction x: -1 kg/kg

Details on the vapor fraction x

Result:

Specific enthalpy h = 3097.38 kJ/kg

For further information on property libraries available for EXCEL®, MATLAB®, Mathcad®, Engineering Equation Solver®, DYMOLA® (Modelica), SimulationX®, and LabView® click here.

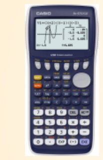
An App for calculating steam properties on iPhone, iPad, and iPod touch can be found here; PDF with the description.

© Zittau/Goerlitz University of Applied Sciences
Faculty of Mechanical Engineering
Department of Technical Thermodynamics
Prof. Hans-Joachim Kretzschmar
Dr. Ines Stoecker
Programmer: Joachim Posselt

Tel.: +49-3583-61-1846 or -1881
Fax: +49-3583-61-1846
E-mail: info@thermodynamics-zittau.de
www.thermodynamics-zittau.de
www.thermodynamic-property-libraries.com
www.international-steam-tables.com
www.thermodynamik-formelsammlung.de

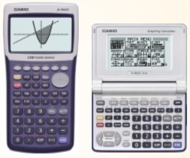
Property Software for Pocket Calculators

FluidCasio

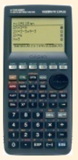


fx 9750 G II

CFX 9850
fx-GG20



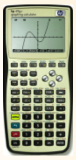
CFX 9860 G
Graph 85



ALGEBRA
FX 2.0

FluidHP

HP 48



HP 49

FluidTI

TI Nspire CX CAS
TI Nspire CAS

TI 83
TI 84
TI 89

TI Voyage 200

TI 92

For more information please contact:

KCE-ThermoFluidProperties UG
Prof. Dr. Hans-Joachim Kretzschmar
Haager Weg 6
92224 Amberg, Germany

Internet: www.thermofluidprop.com
Email: info@thermofluidprop.com
Phone: +49-9621-1762047
Mobile: +49-172-7914607
Fax: +49-3222-1095810

The following thermodynamic and transport properties[a] can be calculated in Excel®, MATLAB®, Mathcad®, Engineering Equation Solver® (EES), DYMOLA® (Modelica), SimulationX® and LabVIEW™:

Thermodynamic Properties

- Vapor pressure p_s
- Saturation temperature T_s
- Density ρ
- Specific volume v
- Enthalpy h
- Internal energy u
- Entropy s
- Exergy e
- Isobaric heat capacity c_p
- Isochoric heat capacity c_v
- Isentropic exponent κ
- Speed of sound w
- Surface tension σ

Transport Properties

- Dynamic viscosity η
- Kinematic viscosity ν
- Thermal conductivity λ
- Prandtl number Pr
- Thermal diffusivity a

Backward Functions

- $T, v, s\ (p,h)$
- $T, v, h\ (p,s)$
- $p, T, v\ (h,s)$
- $p, T\ (v,h)$
- $p, T\ (v,u)$

Thermodynamic Derivatives

- Partial derivatives used in process modeling can be calculated.

---

[a] Not all of these property functions are available in all property libraries.

www.thermofluidprop.com

5 References

[1] Herrmann, S.; Kretzschmar, H.-J.; Gatley, D.P.: Thermodynamic Properties of Real Moist Air, Dry Air, Steam, Water, and Ice. HVAC&R Research 5, 961-986 (2009).

[2] Herrmann, S.; Kretzschmar, H.-J.; Gatley, D.P.: Thermodynamic Properties of Real Moist Air, Dry Air, Steam, Water, and Ice. ASHRAE RP-1485, American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc., Atlanta, GA (2009).

[3] Herrmann, S.; Kretzschmar, H.-J.; Aute, V.C.; Gatley, D.P.; Vogel, E.: Transport Properties of Real Moist Air, Dry Air, Steam, and Water. ASHRAE RP-1767, American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc., Atlanta, GA (2018).

[4] Herrmann, S.; Kretzschmar, H.-J.; Aute, V.C.; Gatley, D.P.; Vogel, E.: Transport Properties of Real Moist Air, Dry Air, Steam, and Water. Science and Technology for the Built Environment 27, 393-401 (2021).

[5] IAPWS. R6-95, Revised Release on the IAPWS Formulation 1995 for the Thermodynamic Properties of Ordinary Water Substance for General and Scientific Use. (2009), available from www.iapws.org.

[6] Wagner, W.; Pruß, A.: The IAPWS Formulation 1995 for the Thermodynamic Properties of Ordinary Water Substance for General and Scientific Use. J. Phys. Chem. Ref. Data 31, 387-535 (2002).

[7] IAPWS. R7-97, Revised Release on the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam IAPWS-IF97. (2007), available from www.iapws.org.

[8] Wagner, W.; Kretzschmar, H.-J.: International Steam Tables. Springer, Berlin (2008).

[9] Parry, W.T.; Bellows, J.C.; Gallagher, J.S.; Harvey, A.H.: ASME International Steam Tables for Industrial Use. ASME Press, New York (2000).

[10] IAPWS. R10-06, Revised Release on the Equation of State 2006 for H_2O Ice Ih. (2009); available from www.iapws.org.

[11] IAPWS. R14-08, Revised Release on the Pressure along the Melting and Sublimation Curves of Ordinary Water Substance. (2008); available from www.iapws.org.

[12] Nelson, H.F.; Sauer, H.J.: Formulation of High-Temperature Properties for Moist Air. HVAC&R Research 8, 311-334 (2002).

[13] Gatley, D.P.: Understanding Psychrometrics, 2nd ed., ASHRAE, Atlanta (2005).

[14] Lemmon, E.W.; Jacobsen, R.T.; Penoncello, S.G.; Friend, D.G.: Thermodynamic Properties of Air and Mixture of Nitrogen, Argon, and Oxygen from 60 to 2000 K at Pressures to 2000 MPa. J. Phys. Chem. Ref. Data 29, 331-385 (2000).

[15] Harvey, A.H.; Huang, P.H.: First-Principles Calculation of the Air-Water Second Virial Coefficient. Int. J. Thermophys. 28, 556-565 (2007).

[16] IAPWS. G7-04, Guideline on the Henry's Constant and Vapor-Liquid Distribution Constant for Gases in H_2O and D_2O at High Temperatures. (2004), available from www.iapws.org.

[17] Gatley, D.P.; Herrmann, S.; Kretzschmar, H.-J.: A Twenty-First Century Molar Mass for Dry Air. HVAC&R Research 14, 655-662 (2008).

[18] Lemmon, E.W.; Jacobsen, R.T.: Viscosity and Thermal Conductivity Equations for Nitrogen, Oxygen, Argon, and Air. Int. J. Thermophys. 25, 21-69 (2004).

[19] IAPWS. R12-08, Release on the IAPWS Formulation 2008 for the Viscosity of Ordinary Water Substance. (2008), available from www.iapws.org.

[20] IAPWS. R15-11, Revised Release on the IAPS Formulation 1985 for the Thermal Conductivity of Ordinary Water Substance. (2008), available from www.iapws.org.

[21] Hyland, R.W.; Wexler, A.: Formulations for the Thermodynamic Properties of Dry Air from 173.15 K to 473.15 K, and of Saturated Moist Air from 173.15 K to 372.15 K, at Pressures to 5 MPa. ASHRAE Trans. 89, 520-535 (1983).

[22] Mohr, P.J.; Taylor, P.N.: CODATA Recommended Values of the Fundamental Physical Constants: 2002. Rev. Mod. Phys. 77, 1-107 (2005).

[23] ASHRAE. 2009 Handbook of Fundamentals. Chapter 1 - Psychrometrics. American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc., Atlanta, GA (2009).

[24] Feistel, R.; Lovell-Smith, J.W.; Hellmuth, O.: Virial Approximation of the TEOS-10 Equation for the Fugacity of Water in Humid Air. Int. J. Thermophys. 36, 44-68 (2015).

6 Satisfied Customers

Period from 2018 to 2022

The following companies and institutions use the property libraries:

- FluidEXL^{Graphics} for Excel® incl. VBA
- FluidLAB for MATLAB® and Simulink
- FluidMAT for Mathcad®
- FluidPRIME for Mathcad Prime®
- FluidEES for Engineering Equation Solver® EES
- FluidDYM for Dymola® (Modelica) and SimulationX®
- FluidVIEW for LabVIEW™
- FluidPYT for Python
- FluidJAVA for Java
- DLLs for Windows Applications
- Shared Objects for Linux
- Shared Objects for macOS.

2022

ASTG, Graz, Austria	12/2022
Wandschneider + Gutjahr, Hamburg	
RWE Supply & Trading, Essen	11/2022
Stadtwerke Rosenheim	
CEA, Saclay, France	10/2022
RWE Supply & Trading, Essen	
SEEC Saudi Energy Efficiency Center, Riyadh, Saudi Arabia	
MAN, Copenhagen, Denmark	
Hermeler & Partner Consulting Engineers, Sassenberg	09/2022
Envi Con, Nürnberg	
Drill Cool Systems, Bakersfield CA, USA	
RWE Supply & Trading, Essen	
Maerz Ofenbau, Zürich, Switzerland	
Saale Energie, Schkopau	
ERGO, Dresden	
Mainova, Frankfurt/Main	
Bundeswehr, Koblenz	08/2022
RWE Supply & Trading, Essen	
Grenzebach Corporation, Newnan GE, USA	
AGRANA, Gmuend, Austria	07/2022
MIBRAG, Zeitz	
Hochschule Niederrhein, Krefeld	
ULT, Löbau	06/2022
LEAG, Cottbus	

VPC Group, Vetschau	
Wärme, Hamburg	
ILK, Dresden	
Stricker IB, Küssnacht a. Rigi, Switzerland	
LEAG, Cottbus	05/2022
RWE Supply & Trading, Essen	
IGT Tomalla, Kreuztal	
B+T Engineering, Dübendorf, Switzerland	
Stricker IB, Küssnacht a. Rigi, Switzerland	
Vogelsang & Benning, Bochum	04/2022
Frischli, Rehburg-Loccum	
BPS Consulting, Sprenge	03/2022
HS Hannover, Maschinenbau & BioVT	
M+M Turbinentechnik, Bad Salzuflen	
Uni. Strathclyde, Glasgow, UK	02/2022
Delta Energy Group, Jiaozhou City, Qingdao, China	
Wetzel IB, Guben	
Wijbenga, PC Geldermalsen, The Netherlands	
Voith Paper, Heidenheim	
HS Zittau/Görlitz, Maschinenwesen	01/2022
Thermische Abfallbehandlung, Lauta	
Webb Institute, Glen Cove NY, USA	
TU Berlin, Umweltverfahrenstechnik	
SachsenEnergie, Dresden	
Doosan, Chang-won-si, Gyeongsangnam-do, South Korea	
KW3, LH Veenendaal, The Netherlands	
Université du Luxembourg, Esch-sur-Alzette	
Enseleit IB, Mansfeld	
Caliqua/Equans, Zürich, Switzerland	
Rudnick & Enners, Alpenrod	

2021

Wenisch IB, Vetschau	12/2021
PPCHEM, Hinwil, Switzerland	
KW3, The Netherlands	
BASF Ludwigshafen	
Air-Consult, Jena	
Sjerp & Jongeneel, RB Zoetermeer, The Netherlands	11/2021
Maerz Ofenbau, Zürich, Switzerland	
RWE Supply & Trading, Essen	
Hahn IB, Dresden	10/2021
Therm, South Africa	
RWE Supply & Trading, Essen	
TH Nürnberg, Verfahrenstechnik	09/2021
RWE Supply & Trading, Essen	
Enseleit IB, Mansfeld	

SachsenEnergie, Dresden	
BSH Hausgeräte, Berlin	
Norsk Energi, Oslo, Norway	08/2021
AKM Industrieanlagen, Haltern	
Drill Cool Systems, Bakersfield CA, USA	
Siemens Energy Global, Erlangen	07/2021
Wulff & Umag, Husum	
Planungsbüro Waidhas, Chemnitz	
Burkhardt Energie Technik, Mühlhausen	
Lücke IB, Paderborn	06/2021
TU Dresden, Energieverfahrenstechnik	
Wärme, Hamburg	
AL-KO Therm, Kötz	
PCK Raffinerie, Schwedt	
Vogelsang & Benning, Bochum	05/2021
MTU, München	
VPC Group, Vetschau	
AVG, Köln	04/2021
TH Ulm, Institut für Fahrzeugtechnik	
Marty IB, Oberwil, Switzerland	
HypTec, Lebring, Austria	
Lopez IB, Getxo, Bizkaia, Spain	03/2021
GM Remediation Systems, Leoben, Austria	
Jager Kältetechnik, Osnabrück	
T&M Automation, GR Leidschendam, The Netherlands	
RWE Supply & Trading, Essen	
Stadtwerke Leipzig	
Beuth Hochschule für Technik, Berlin	
Beleth IB, Woeth	02/2021
ZTL, Thal, Austria	
ETABO Bochum	
RWE Supply & Trading, Essen	
Onyx Germany, Berlin	
TU Dresden, Kältetechnik	
GOHL-KTK, Durmersheim	
Therm Development, South Africa	
thermofin, Heinsdorfergrund	
RWE Supply & Trading, Essen	01/2021
STEAG, Essen	
ETA Energieberatung, Pfaffenhofen	
Enex Power, Kirchseeon	

2020

Drill Cool, Bakersfield CA, USA	12/2020
Manders, The Netherlands	
RWE Supply & Tranding, Essen	

NEOWAT Lodz, Poland	
University of Duisburg-Essen, Duisburg	11/2020
Stellenbosch University, South Africa	
University De France-COMTe, France	
RWE, Essen	
STEAG, Herne	
Isenmann Ingenierbüro	
University of Stuttgart, ITLR, Stuttgart	
Norsk Energi, Oslo, Norway	
TGM Kanis, Nürnberg	
Stadtwerke Neuburg	10/2020
Smurfit Kappa, Roermond, The Netherlands	
RWE, Essen	
Hochschule Zittau/Görlitz, Wirtschaftsingenieurwesen	
Stadtwerke, Neuburg	
ILK, Dresden	
ATESTEO, Alsdorf	
Hochschule Zittau/Görlitz, Maschinenwesen	
TH Nürnberg, Verfahrenstechnik	
Drill Cool, Bakersfield CA,USA	09/2020
RWE, Essen	
2Meyers Ingenieurbüro, Nürnberg	
FELUWA, Mürlenbach	
Stadtwerke Neuburg	
Caverion, Wien, Austria	
GMVA Niederrhein, Oberhausen	
INWAT Lodz, Poland	
Troche Ingenieurbüro, Hayingen	08/2020
CEA Saclay, France	
VPC, Vetschau	07/2020
FSK System-Kälte-Klima, Dortmund	
Exergie Etudes, Sarl, Switzerland	
AWG Wuppertal	
STEAG Energy Services, Zwingenberg	
Hochschule Braunschweig	06/2020
DBI, Leipzig	
GOHL-KTK, Dumersheim	
TU Dresden, Energieverfahrenstechnik	
BASF SE, ESI/EE, Ludwigshafen	
Wärme Hamburg	
Ruchti Ingenieurbüro, Uster, Switzerland	
IWB, Basel, Switzerland	
Midiplan, Bietingen-Bissingen	05/2020
Knieschke, Ingenieurbüro	
RWE, Essen	
Leser, Hamburg	

AGRANA, Gmünd, Austria	
EWT Wassertechnik, Celle	
Hochschule Darmstadt	04/2020
MTU München CCP	
HAW Hamburg	03/2020
Hanon, Novi Jicin, Czech Republic	
TU Dresden, Kältetechnik	
MAN, Copenhagen, Denmark	
EnerTech, Radebeul	02/2020
LEAG, Cottbus	
B+B Enginering Magdeburg	
Hochschule Offenburg	
WIB, Dennheritz	01/2020
Universität Duisburg-Essen, Strömungsmaschinen	
Kältetechnik Dresen-Bremen	
TH Ingolstadt	
Vattenfall AB, Jokkmokk, Sweden	
Fraunhofer UMSICHT	

2019

PEU Leipzig, Rötha	12/2019
MB-Holding, Vestenbergsgreuth	
RWE, Essen	
Georg-Büchner-Hochschule, Darmstadt	11/2019
EEB ENERKO, Aldenhoven	
Robert Benoufa Energietechnik, Wiesloch	
Kehrein & Kubanek Klimatechnik, Moers	10/2019
Hanon Systems Autopal Services, Hluk, Czech Republic	
CEA Saclay, Gif Sur Yvette cedex, France	
Saudi Energy Efficiency Center SEEC, Riyadh, Saudi Arabia	
VPC, Vetschau	09/2019
jGanser PM + Engineering, Forchheim	
Endress+Hauser Flowtec AG, Reinach, Switzerland	
Ruchti IB, Uster, Switzerland	
ZWILAG Zwischenlager Würenlingen, Switzerland	08/2019
Hochschule Zittau/Görlitz, Faculty Maschinenwesen	
Stadtwerke Neubrandenburg	
Physikalisch Technische Bundesanstalt PTB, Braunschweig	
GMVA Oberhausen	07/2019
Endress+Hauser Flowtec AG, Reinach, Switzerland	
WARNICA, Waterloo, Canada	
MIBRAG, Zeitz	06/2019
Pöyry, Zürich, Switzerland	
RWTH Aachen, Institut für Strahlantriebe und Turbomaschinen	
Midiplan, Bietigheim-Bissingen	
GKS Schweinfurt	

I-P – 6/6

HS Zittau/Görlitz, Wirtschaftswissenschaften und Wirtschaftsingenieurwesen	
ILK Dresden	
HZDR Helmholtz Zentrum Dresden-Rossendorf	
TH Köln, Technische Gebäudeausrüstung	05/2019
IB Knittel, Braunschweig	
Norsk Energi, Oslo, Norway	
STEAG, Essen	
Stora Enso, Eilenburg	
IB Lücke, Paderborn	
Haarslev, Sonderso, Denmark	
MAN Augsburg	
Wieland Werke, Ulm	04/2019
Fels-Werke, Elbingerode	
Univ. Luxembourg, Luxembourg	
BTU Cottbus, Power Engineering	03/2009
Eins-Energie Sachsen, Schwarzenberg	
TU Dresden, Kälte- und Kryotechnik	
ITER, St. Paul Lez Durance Cedex, France	
Fraunhofer UMSICHT, Oberhausen	
Comparex Leipzig for Spedition Thiele HEMMERSBACH	
Rückert NaturGas, Lauf/Pegnitz	
BASF, Basel, Switzerland	02/2019
Stadtwerke Leipzig	
Maerz Ofenbau Zürich, Switzerland	
Hanon Systems Germany, Kerpen	
Thermofin, Heinsdorfergrund	01/2019
BSH Berlin	

2018

Jaguar Energy, Guatemala	12/2018
WEBASTO, Gilching	
Smurfit Kappa, Oosterhout, Netherlands	
Univ. BW München	
RAIV, Liberec for VALEO, Prague, Czech Republic	11/2018
VPC Group Vetschau	
SEITZ, Wetzikon, Switzerland	
MVV, Mannheim	10/2018
IB Troche	
KANIS Turbinen, Nürnberg	
TH Ingolstadt, Institut für neue Energiesysteme	
IB Kristl & Seibt, Graz, Austria	09/2018
INEOS, Köln	
IB Lücke, Paderborn	
Südzucker, Ochsenfurt	08/2018
K&K Turbinenservice, Bielefeld	07/2018
OTH Regensburg, Elektrotechnik	

Comparex Leipzig for LEAG, Berlin	06/2018
Münstermann, Telgte	05/2018
TH Nürnberg, Verfahrenstechnik	
Universität Madrid, Madrid, Spanien	
HS Zittau/Görlitz, Wirtschaftsingenieurwesen	
HS Niederrhein, Krefeld	
Wilhelm-Büchner HS, Pfungstadt	03/2018
GRS, Köln	
WIB, Dennheritz	
RONAL AG, Härklingen, Schweiz	02/2018
Ingenieurbüro Leipert, Riegelsberg	
AIXPROCESS, Aachen	
KRONES, Neutraubling	
Doosan Lentjes, Ratingen	01/2018

Part SI Units

1 Property Library ASHRAE-LibHuAirProp-SI

1.1 Function Overview

1.1.1 Function Overview for Real Moist Air

Functional Dependence	Function Name	Property or Function	Unit of the Result	Page
$a = \mathrm{f}(p,t,W)$	a_ptW_HAP_SI	Thermal diffusivity	m²/s	3/2
$\alpha_p = \mathrm{f}(p,t,W)$	alphap_ptW_HAP_SI	Relative pressure coefficient	1/K	3/3
$\beta_p = \mathrm{f}(p,t,W)$	betap_ptW_HAP_SI	Isothermal stress coefficient	kg/m³	3/4
$c = \mathrm{f}(p,t,W)$	c_ptW_HAP_SI	Speed of sound	m/s	3/5
$c_p = \mathrm{f}(p,t,W)$	cp_ptW_HAP_SI	Specific isobaric heat capacity	kJ/(kg·K)	3/6
$c_v = \mathrm{f}(p,t,W)$	cv_ptW_HAP_SI	Specific isochoric heat capacity	kJ/(kg·K)	3/7
$f = \mathrm{f}(p,t)$	f_pt_HAP_SI	Enhancement factor (decimal ratio)	-	3/8
$h = \mathrm{f}(p,t,W)$	h_ptW_HAP_SI	Air-specific enthalpy	$\mathrm{kJ/kg_a}$	3/9
$\eta = \mathrm{f}(p,t,W)$	Eta_ptW_HAP_SI	Dynamic viscosity	Pa·s	3/10
$\kappa = \mathrm{f}(p,t,W)$	Kappa_ptW_HAP_SI	Isentropic exponent	-	3/11
$\lambda = \mathrm{f}(p,t,W)$	Lambda_ptW_HAP_SI	Thermal conductivity	W/(m·K)	3/12
$\nu = \mathrm{f}(p,t,W)$	Ny_ptW_HAP_SI	Kinematic viscosity	m²/s	3/13
$p = \mathrm{f}(t,s,W)$	p_tsW_HAP_SI	Pressure of humid air	kPa	3/14
$p = \mathrm{f}(z_{\mathrm{ele}})$	p_zele_HAP_SI	Pressure of humid air from elevation	kPa	3/15
$p_{\mathrm{Air}} = \mathrm{f}(p,t,W)$	pAIR_ptW_HAP_SI	Partial pressure of dry air in moist air	kPa	3/16
$p_{\mathrm{H2O}} = \mathrm{f}(p,t,W)$	pH2O_ptW_HAP_SI	Partial pressure of water vapor in moist air	kPa	3/17
$p_{\mathrm{H2Os}} = \mathrm{f}(p,t)$	pH2Os_pt_HAP_SI	Partial saturation pressure of water vapor	kPa	3/18

Functional Dependence	**Function Name**	**Property or Function**	**Unit of the Result**	**Page**
$\varphi = \mathrm{f}(p,t,W)$	phi_ptW_HAP_SI	Relative humidity (decimal ratio)	-	3/19
$Pr = \mathrm{f}(p,t,W)$	Pr_ptW_HAP_SI	PRANDTL number	-	3/20
$\psi_{Air} = \mathrm{f}(W)$	PsiAir_W_HAP_SI	Mole fraction of dry air in moist air	$\mathrm{mol_a/mol}$	3/21
$\psi_{\mathrm{H2O}} = \mathrm{f}(W)$	PsiH2O_W_HAP_SI	Mole fraction of water vapor in moist air	$\mathrm{mol_w/mol}$	3/22
$\rho = \mathrm{f}(p,t,W)$	Rho_ptW_HAP_SI	Density	$\mathrm{kg/m^3}$	3/23
$s = \mathrm{f}(p,t,W)$	s_ptW_HAP_SI	Air-specific entropy	$\mathrm{kJ/(kg_a \cdot K)}$	3/24
$t = \mathrm{f}(p,h,\varphi)$	t_phphi_HAP_SI	Backward function: temperature from total pressure, air-specific enthalpy and relative humidity	°C	3/25
$t = \mathrm{f}(p,h,W)$	t_phW_HAP_SI	Backward function: temperature from total pressure, air-specific enthalpy and humidity ratio	°C	3/26
$t = \mathrm{f}(p,s,W)$	t_psW_HAP_SI	Backward function: temperature from total pressure, air-specific entropy and humidity ratio	°C	3/27
$t = \mathrm{f}(p,t_{\mathrm{wb}},W)$	t_ptwbW_HAP_SI	Backward function: temperature from total pressure, wet-bulb temperature and humidity ratio	°C	3/28
$t_{\mathrm{d}} = \mathrm{f}(p,W)$	td_pW_HAP_SI	Dew-point/frost-point temperature	°C	3/29
$t_{\mathrm{s}} = \mathrm{f}(p,p_{\mathrm{H2O}})$	ts_ppH2O_HAP_SI	Backward function: saturation temperature of water from total pressure and partial pressure of water vapor	°C	3/30
$t_{\mathrm{wb}} = \mathrm{f}(p,t,W)$	twb_ptW_HAP_SI	Wet-bulb/ice-bulb temperature	°C	3/31
$u = \mathrm{f}(p,t,W)$	u_ptW_HAP_SI	Air-specific internal energy	$\mathrm{kJ/kg_a}$	3/32
$v = \mathrm{f}(p,t,W)$	v_ptW_HAP_SI	Air-specific volume	$\mathrm{m^3/kg_a}$	3/33
$W = \mathrm{f}(p,t,p_{\mathrm{H2O}})$	W_ptpH2O_HAP_SI	Humidity ratio from total pressure, temperature, and partial pressure of water vapor	$\mathrm{kg_w/kg_a}$	3/34
$W = \mathrm{f}(p,t,\varphi)$	W_ptphi_HAP_SI	Humidity ratio from total pressure, temperature, and relative humidity	$\mathrm{kg_w/kg_a}$	3/35
$W = \mathrm{f}(p,t_{\mathrm{d}})$	W_ptd_HAP_SI	Humidity ratio from total pressure and dew-point temperature	$\mathrm{kg_w/kg_a}$	3/36

Functional Dependence	**Function Name**	**Property or Function**	**Unit of the Result**	**Page**
$W = \mathrm{f}(p,t,t_{\mathrm{wb}})$	W_pttwb_HAP_SI	Humidity ratio from total pressure, (dry bulb) temperature, and wet-bulb temperature	$\mathrm{kg_w/kg_a}$	3/37
$W_{\mathrm{s}} = \mathrm{f}(p,t)$	Ws_pt_HAP_SI	Saturation humidity ratio	$\mathrm{kg_w/kg_a}$	3/38
$\xi_{\mathrm{Air}} = \mathrm{f}(W)$	XiAir_W_HAP_SI	Mass fraction of dry air in moist air	$\mathrm{kg_a/kg}$	3/39
$\xi_{\mathrm{H2O}} = \mathrm{f}(W)$	XiH2O_W_HAP_SI	Mass fraction of water vapor in moist air	$\mathrm{kg_w/kg}$	3/40
$Z = \mathrm{f}(p,t,W)$	Z_ptW_HAP_SI	Compression factor (decimal ratio)	-	3/41

Range of Validity of Thermodynamic Properties

Property	Range of Validity
Pressure:	$0.01 \leq p \leq 10\,000$ kPa
Temperature:	$-143.15 \leq t \leq 350$ °C
Humidity ratio:	$0 \leq W \leq 10$ $\mathrm{kg_w/kg_a}$
Relative humidity:	$0 \leq \varphi \leq 1$ (decimal ratio)
Dew-point temperature:	$-143.15 \leq t_d \leq 350$ °C
Wet-bulb temperature:	$-143.15 \leq t_{wb} \leq 350$ °C

Range of Validity of Transport Properties

Property	Range of Validity
Pressure:	$0.01 \leq p \leq 10\,000$ kPa
Temperature:	$-73.15 \leq t \leq 350$ °C
Humidity ratio:	$0 \leq W \leq 10$ $\mathrm{kg_w/kg_a}$
Relative humidity:	$0 \leq \varphi \leq 1$ (decimal ratio)

Molar Masses

Component	Molar Mass	Reference
Dry Air	28.966 kg/kmol	[17]
Water	18.015268 kg/kmol	[5], [6]

Reference States

Property	Dry Air	Steam, Water, and Ice
Pressure	101.325 kPa	$p_s(0.01°C) = 0.611657$ kPa
Temperature	0°C	0.01°C
Enthalpy	0 kJ/kg	0.000611782 kJ/kg
Entropy	0 kJ/(kg K)	0 kJ/(kg K)

Units

Symbol	Quantity	Unit
p	Pressure	kPa
t	Temperature	°C
W	Humidity ratio	$\mathrm{kg_w/kg_a}$ (kg water / kg dry air)
φ	Relative humidity	(decimal ratio)
t_d	Dew point temperature	°C
t_{wb}	Wet bulb temperature	°C

1.1.2 Function Overview for Steam and Water for Temperatures $t \geq 0°C$

Functional Dependence	Function Name	Property	Unit of the Result	Page
$h_{liq} = f(p,t)$	hliq_pt_97_SI	Specific enthalpy of liquid water	kJ/kg	3/43
$h_{liq,s} = f(t)$	hliqs_t_97_SI	Specific enthalpy of saturated liquid water	kJ/kg	3/44
$h_{vap,s} = f(t)$	hvaps_t_97_SI	Specific enthalpy of saturated water vapor	kJ/kg	3/45
$p_s = f(t)$	ps_t_97_SI	Saturation pressure of water	kPa	3/46
$s_{liq} = f(p,t)$	sliq_pt_97_SI	Specific entropy of liquid water	kJ/(kg·K)	3/47
$s_{liq,s} = f(t)$	sliqs_t_97_SI	Specific entropy of saturated liquid water	kJ/(kg·K)	3/48
$s_{vap,s} = f(t)$	svaps_t_97_SI	Specific entropy of saturated water vapor	kJ/(kg·K)	3/49
$t_s = f(p)$	ts_p_97_SI	Saturation temperature of water	°C	3/50
$v_{liq} = f(p,t)$	vliq_pt_97_SI	Specific volume of liquid water	m^3/kg	3/51
$v_{liq,s} = f(t)$	vliqs_t_97_SI	Specific volume of saturated liquid water	m^3/kg	3/52
$v_{vap,s} = f(t)$	vvaps_t_97_SI	Specific volume of saturated water vapor	m^3/kg	3/53

Range of Validity

Property	Range of Validity
Pressure:	$0.01 \leq p \leq 10\,000$ kPa
Temperature:	$0 \leq t \leq 350$ °C

Units

Symbol	Quantity	Unit
p	Pressure	kPa
t	Temperature	°C

Reference State

Property	Water Vapor and Liquid Water
Pressure	p_s(0.01°C) = 0.611657 kPa
Temperature	0.01°C
Enthalpy	0.000611782 kJ/kg
Entropy	0 kJ/(kg K)

1.1.3 Function Overview for Steam and Ice for Temperatures $t \leq 0°C$

Functional Dependence	**Function Name**	**Property**	**Unit of the Result**	**Page**
$h_{ice,sub} = f(t)$	hicesub_t_06_SI	Specific enthalpy of saturated ice	kJ/kg	3/55
$h_{vap,sub} = f(t)$	hvapsub_t_95_SI	Specific enthalpy of saturated water vapor	kJ/kg	3/56
$p_{mel} = f(t)$	pmel_t_08_SI	Melting pressure of ice	kPa	3/57
$p_{sub} = f(t)$	psub_t_08_SI	Sublimation pressure of ice	kPa	3/58
$s_{ice,sub} = f(t)$	sicesub_t_06_SI	Specific entropy of saturated ice	kJ/(kg·K)	3/59
$s_{vap,sub} = f(t)$	svapsub_t_95_SI	Specific entropy of saturated water vapor	kJ/(kg·K)	3/60
$t_{mel} = f(p)$	tmel_p_08_SI	Melting temperature of ice	°C	3/61
$t_{sub} = f(p)$	tsub_p_08_SI	Sublimation temperature of ice	°C	3/62
$v_{ice,sub} = f(t)$	vicesub_t_06_SI	Specific volume of saturated ice	m^3/kg	3/63
$v_{vap,sub} = f(t)$	vvapsub_t_95_SI	Specific volume of saturated water vapor	m^3/kg	3/64

Range of Validity

Property	**Range of Validity**
Pressure:	$p_{\text{sub}}(-143.15°\text{C}) = 1.2002 \times 10^{-11} \leq p \leq 10\,000$ kPa
Temperature:	$-143.15 \leq t \leq 0$ °C

Units

Symbol	**Quantity**	**Unit**
p	Pressure	kPa
t	Temperature	°C

Reference State

Property	**Water Vapor and Ice**
Pressure	$p_s(0.01°\text{C}) = 0.611657$ kPa
Temperature	0.01°C
Enthalpy	0.000611782 kJ/kg
Entropy	0 kJ/(kg K)

1.2 Conversion of SI and I-P Units

Property		Conversion: SI Units → I-P Units	Conversion: I-P Units → SI Units	Units SI	Units I-P
Thermal diffusivity	a	$\frac{a_{\mathrm{IP}}}{\frac{\mathrm{ft}^2}{\mathrm{s}}} = \frac{a_{\mathrm{SI}}}{\frac{\mathrm{m}^2}{\mathrm{s}}} \times 10.76391042$	$\frac{a_{\mathrm{SI}}}{\frac{\mathrm{m}^2}{\mathrm{s}}} = \frac{a_{\mathrm{IP}}}{\frac{\mathrm{ft}^2}{\mathrm{s}}} \times 0.0929304$	m²/s	ft²/s
Relative pressure coefficient	α_p	$\frac{\alpha_{p,\mathrm{IP}}}{\frac{1}{°\mathrm{R}}} = \frac{\alpha_{p,\mathrm{SI}}}{\frac{1}{\mathrm{K}}} \times \frac{9}{5}$	$\frac{\alpha_{p,\mathrm{SI}}}{\frac{1}{\mathrm{K}}} = \frac{\alpha_{p,\mathrm{IP}}}{\frac{1}{°\mathrm{R}}} \times \frac{5}{9}$	1/K	1/°R
Isothermal stress coefficient	β_p	$\frac{\beta_{p,\mathrm{IP}}}{\frac{\mathrm{lb}}{\mathrm{ft}^3}} = \frac{\beta_{p,\mathrm{SI}}}{\frac{\mathrm{kg}}{\mathrm{m}^3}} \times 0.062428$	$\frac{\beta_{p,\mathrm{SI}}}{\frac{\mathrm{kg}}{\mathrm{m}^3}} = \frac{\beta_{p,\mathrm{IP}}}{\frac{\mathrm{lb}}{\mathrm{ft}^3}} \times 16.018463$	kg/m³	lb/ft³
Speed of sound	c	$\frac{c_{\mathrm{IP}}}{\frac{\mathrm{ft}}{\mathrm{s}}} = \frac{c_{\mathrm{SI}}}{\frac{\mathrm{m}}{\mathrm{s}}} \times 3.2808399$	$\frac{c_{\mathrm{SI}}}{\frac{\mathrm{m}}{\mathrm{s}}} = \frac{c_{\mathrm{IP}}}{\frac{\mathrm{ft}}{\mathrm{s}}} \times 0.3048$	m/s	ft/s
Specific isobaric heat capacity	c_p	$\frac{c_{p,\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}\ °\mathrm{R}}} = \frac{c_{p,\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg\ K}}} \times 0.2388459$	$\frac{c_{p,\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg\ K}}} = \frac{c_{p,\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}\ °\mathrm{R}}} \times 4.1868$	kJ/(kg·K)	Btu/(lb·°R)
Specific isochoric heat capacity	c_v	$\frac{c_{v,\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}\ °\mathrm{R}}} = \frac{c_{v,\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg\ K}}} \times 0.2388459$	$\frac{c_{v,\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg\ K}}} = \frac{c_{v,\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}\ °\mathrm{R}}} \times 4.1868$	kJ/(kg·K)	Btu/(lb·°R)
Dynamic viscosity	η	$\frac{\eta_{\mathrm{IP}}}{\frac{\mathrm{lb\ s}}{\mathrm{ft}^2}} = \frac{\eta_{\mathrm{SI}}}{\frac{\mathrm{Pa}}{\mathrm{s}}} \times 0.02088543$	$\frac{\eta_{\mathrm{SI}}}{\frac{\mathrm{Pa}}{\mathrm{s}}} = \frac{\eta_{\mathrm{IP}}}{\frac{\mathrm{lb\ s}}{\mathrm{ft}^2}} \times 47.880259$	Pa·s	lb·s/ft²
Enhancement factor	f	$f_{\mathrm{IP}} = f_{\mathrm{SI}}$	$f_{\mathrm{SI}} = f_{\mathrm{IP}}$	-	-

Property		Conversion: SI Units → I-P Units	Conversion: I-P Units → SI Units	Units SI	Units I-P
Air-specific enthalpy (moist air)	h	$\frac{h_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb_a}}} = \frac{h_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg_a}}} \times 0.4299226 + 7.68565365666$	$\frac{h_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg_a}}} = \left(\frac{h_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb_a}}} - 7.68565365666 \right) \times 2.326$	kJ/kg$_a$	Btu/lb$_a$
Specific enthalpy (water, water vapor, ice)	h_w	$\frac{h_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}}} = \frac{h_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg}}} \times 0.4299226$	$\frac{h_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg}}} = \frac{h_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb}}} \times 2.326$	kJ/kg	Btu/lb
Isentropic exponent	κ	$\kappa_{\mathrm{IP}} = \kappa_{\mathrm{SI}}$	$\kappa_{\mathrm{SI}} = \kappa_{\mathrm{IP}}$	-	-
Thermal conductivity	λ	$\frac{\lambda_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{h\ ft\ ^\circ R}}} = \frac{\lambda_{\mathrm{SI}}}{\frac{\mathrm{W}}{\mathrm{m\ K}}} \times 0.57778932$	$\frac{\lambda_{\mathrm{SI}}}{\frac{\mathrm{W}}{\mathrm{m\ K}}} = \frac{\lambda_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{h\ ft\ ^\circ R}}} \times 1.73073467$	W/(m·K)	Btu/(h·ft·°R)
Kinematic viscosity	ν	$\frac{\nu_{\mathrm{IP}}}{\frac{\mathrm{ft^2}}{\mathrm{s}}} = \frac{\nu_{\mathrm{SI}}}{\frac{\mathrm{m^2}}{\mathrm{s}}} \times 10.763910417$	$\frac{\nu_{\mathrm{SI}}}{\frac{\mathrm{m^2}}{\mathrm{s}}} = \frac{\nu_{\mathrm{IP}}}{\frac{\mathrm{ft^2}}{\mathrm{s}}} \times 0.092903040$	m²/s	ft²/s
Pressure	p	$\frac{p_{\mathrm{IP}}}{\mathrm{psi}} = \frac{p_{\mathrm{SI}}}{\mathrm{kPa}} \times 0.14503774$	$\frac{p_{\mathrm{SI}}}{\mathrm{kPa}} = \frac{p_{\mathrm{IP}}}{\mathrm{psi}} \times 6.894757$	kPa	psi
Relative humidity	φ	$\varphi_{\mathrm{IP}} = \varphi_{\mathrm{SI}}$	$\varphi_{\mathrm{SI}} = \varphi_{\mathrm{IP}}$	-	-
Prandtl number	Pr	$Pr_{\mathrm{IP}} = Pr_{\mathrm{IP}}$	$Pr_{\mathrm{SI}} = Pr_{\mathrm{IP}}$	-	-
Mole fraction	ψ	$\psi_{\mathrm{IP}} = \psi_{\mathrm{SI}}$	$\psi_{\mathrm{SI}} = \psi_{\mathrm{IP}}$	mol/mol	mol/mol
Density	ρ	$\frac{\rho_{\mathrm{IP}}}{\frac{\mathrm{lb}}{\mathrm{ft^3}}} = \frac{\rho_{\mathrm{SI}}}{\frac{\mathrm{kg}}{\mathrm{m^3}}} \times 0.062428$	$\frac{\rho_{\mathrm{SI}}}{\frac{\mathrm{kg}}{\mathrm{m^3}}} = \frac{\rho_{\mathrm{IP}}}{\frac{\mathrm{lb}}{\mathrm{ft^3}}} \times 16.018463$	kg/m³	lb/ft³
Air-specific entropy (moist air)	s	$\frac{s_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb_a\ ^\circ R}}} = \frac{s_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg_a\ K}}} \times 0.2388459 + 0.01616365106$	$\frac{s_{\mathrm{SI}}}{\frac{\mathrm{kJ}}{\mathrm{kg_a\ K}}} = \left(\frac{s_{\mathrm{IP}}}{\frac{\mathrm{Btu}}{\mathrm{lb_a\ ^\circ R}}} - 0.01616365106 \right) \times 4.1868$	kJ/(kg$_a$·K)	Btu/(lb$_a$·°R)

Property		Conversion: SI Units → I-P Units	Conversion: I-P Units → SI Units	Units SI	Units I-P
Specific entropy (water, water vapor, ice)	s_w	$\frac{s_{IP}}{\frac{\mathrm{Btu}}{\mathrm{lb_a}\ {}^\circ\mathrm{R}}} = \frac{s_{SI}}{\frac{\mathrm{kJ}}{\mathrm{kg_a}\ \mathrm{K}}} \times 0.23884589$	$\frac{s_{SI}}{\frac{\mathrm{kJ}}{\mathrm{kg_a}\ \mathrm{K}}} = \frac{s_{IP}}{\frac{\mathrm{Btu}}{\mathrm{lb_a}\ {}^\circ\mathrm{R}}} \times 4.1868$	kJ/(kg$_a$·K)	Btu/(lb$_a$·°R)
Temperature	t	$\frac{t_{IP}}{{}^\circ\mathrm{F}} = \frac{t_{SI}}{{}^\circ\mathrm{C}} \times \frac{9}{5} + 32$	$\frac{t_{SI}}{{}^\circ\mathrm{C}} = \left(\frac{t_{IP}}{{}^\circ\mathrm{F}} - 32\right) \times \frac{5}{9}$	°C	°F
Air-specific internal energy (moist air)	u	$(u = h - pv)$ $\frac{u_{IP}}{\frac{\mathrm{Btu}}{\mathrm{lb_a}}} = \frac{h_{SI}}{\frac{\mathrm{kJ}}{\mathrm{kg_a}}} \times 0.4299226 + 7.68565365666$ $- \frac{p_{SI}}{\mathrm{kPa}} \times 0.145037738 \cdot \frac{v_{SI}}{\frac{\mathrm{m^3}}{\mathrm{kg_a}}} \times 16.018453$	$(u = h - pv)$ $\frac{u_{SI}}{\frac{\mathrm{kJ}}{\mathrm{kg_a}}} = \left(\frac{h_{IP}}{\frac{\mathrm{Btu}}{\mathrm{lb_a}}} - 7.68565365666\right) \times 2.236$ $- \frac{p_{IP}}{\mathrm{psi}} \times 6.894757293 \cdot \frac{v_{SIP}}{\frac{\mathrm{ft^3}}{\mathrm{lb_a}}} \times 0.062428$	kJ/kg$_a$	Btu/lb$_a$
Air-specific volume (moist air)	v	$\frac{v_{IP}}{\frac{\mathrm{ft^3}}{\mathrm{lb_a}}} = \frac{v_{SI}}{\frac{\mathrm{m^3}}{\mathrm{kg_a}}} \times 16.018453$	$\frac{v_{SI}}{\frac{\mathrm{m^3}}{\mathrm{kg_a}}} = \frac{v_{IP}}{\frac{\mathrm{ft^3}}{\mathrm{lb_a}}} \times 0.062428$	m³/kg$_a$	ft³/lb$_a$
Specific volume (water, water vapor, ice)	v_w	$\frac{v_{IP}}{\frac{\mathrm{ft^3}}{\mathrm{lb}}} = \frac{v_{SI}}{\frac{\mathrm{m^3}}{\mathrm{kg}}} \times 16.018453$	$\frac{v_{SI}}{\frac{\mathrm{m^3}}{\mathrm{kg}}} = \frac{v_{IP}}{\frac{\mathrm{ft^3}}{\mathrm{lb}}} \times 0.062428$	m³/kg	ft³/lb
Humidity ratio	W	$W_{IP} = W_{SI}$	$W_{SI} = W_{IP}$	kg$_w$/kg$_a$	lb$_w$/lb$_a$
Mass fraction	ξ	$\xi_{IP} = \xi_{SI}$	$\xi_{SI} = \xi_{IP}$	kg$_w$/kg	lb$_w$/lb
Compression factor	Z	$Z_{IP} = Z_{SI}$	$Z_{SI} = Z_{IP}$	-	-

1.3 Calculation Algorithms

1.3.1 Algorithms for Real Moist Air

The properties of moist air are calculated from the modified Hyland-Wexler model given in Herrmann, Kretzschmar, and Gatley (HKG) [1], [2]. The modifications incorporate:

- the value for the universal molar gas constant from the CODATA standard by Mohr and Taylor [22]
- the value for the molar mass of dry air from Gatley et al. [17] and that of water from IAPWS-95 [5], [6]
- the calculation of the ideal-gas parts of the heat capacity, enthalpy, and entropy for dry air from the fundamental equation of Lemmon et al. [14]
- the calculation of the ideal-gas parts of the heat capacity, enthalpy, and entropy for water vapor from IAPWS-IF97 [7], [8], [9] for $t \geq 0°\text{C}$ and from IAPWS-95 [5], [6] for $t \leq 0°\text{C}$
- the calculation of the vapor-pressure enhancement factor from the equation given by the models of Hyland and Wexler [21]
- the calculation of the second and third molar virial coefficients B_{aa} and C_{aaa} for dry air from the fundamental equation of Lemmon et al. [14] according to Feistel et al. [24]
- the calculation of the second and third molar virial coefficients B_{ww} and C_{www} for water and steam from IAPWS-95 [5], [6] according to Feistel et al. [24]
- the calculation of the air-water second molar cross-virial coefficient B_{aw} from Harvey and Huang [15]
- the calculation of the air-water third molar cross-virial coefficients C_{aaw} and C_{aww} from Nelson and Sauer [12], [13]
- the calculation of the saturation pressure of water from IAPWS-IF97 [7], [8], [9] for $t \geq 0°\text{C}$ and of the sublimation pressure of water from IAPWS-08 [11] for $t \leq 0°\text{C}$
- the calculation of the isothermal compressibility of saturated liquid water from IAPWS-IF97 [7], [8], [9] for $t \geq 0°\text{C}$ and that of ice from IAPWS-06 [10] for $t \leq 0°\text{C}$ in the determination of the vapor-pressure enhancement factor
- the calculation of Henry's constant from the IAPWS Guideline 2004 [16] in the determination of the enhancement factor. The mole fractions for the three main components of dry air were taken from Lemmon et al. [14]. Argon was not considered in the calculation of Henry's constant in the former research projects, but it is now the third component of dry air.

The transport properties of moist air are calculated from the model given in Herrmann et al. [3], [4].

1.3.2 Algorithms for Steam and Water for Temperatures $t \geq 0°C$

The *p*-*T* diagram in Fig. 1 shows the formulations used for water and water vapor. The temperature range above 0°C is covered by IAPWS-IF97 [7], [8], [9]:

- The saturation line is calculated from the IAPWS-IF97 saturation pressure equation $p_s^{97}(t)$ and saturation temperature equation $t_s^{97}(p)$.
- The properties in the liquid region including saturated-liquid line are calculated from the fundamental equation of the IAPWS-IF97 region 1.
- The properties in the vapor region including saturated-vapor line are calculated from the fundamental equation of the IAPWS-IF97 region 2.

1.3.3 Algorithms for Steam and Ice for Temperatures $t \leq 0°C$

- The sublimation curve is covered by the IAPWS-08 sublimation pressure equation $p_{\text{subl}}^{08}(t)$ [11] (see Fig. 1).
- The properties of ice including saturated ice are determined by the fundamental equation of the IAPWS-06 [10].
- The properties of vapor including saturated vapor are calculated from the fundamental equation of IAPWS-95 [5], [6].

1.3.4 Overview of the Applied Formulations for Steam, Water, and Ice

The following *p*-*T* diagram shows the used IAPWS Formulations and the ranges where they are applied.

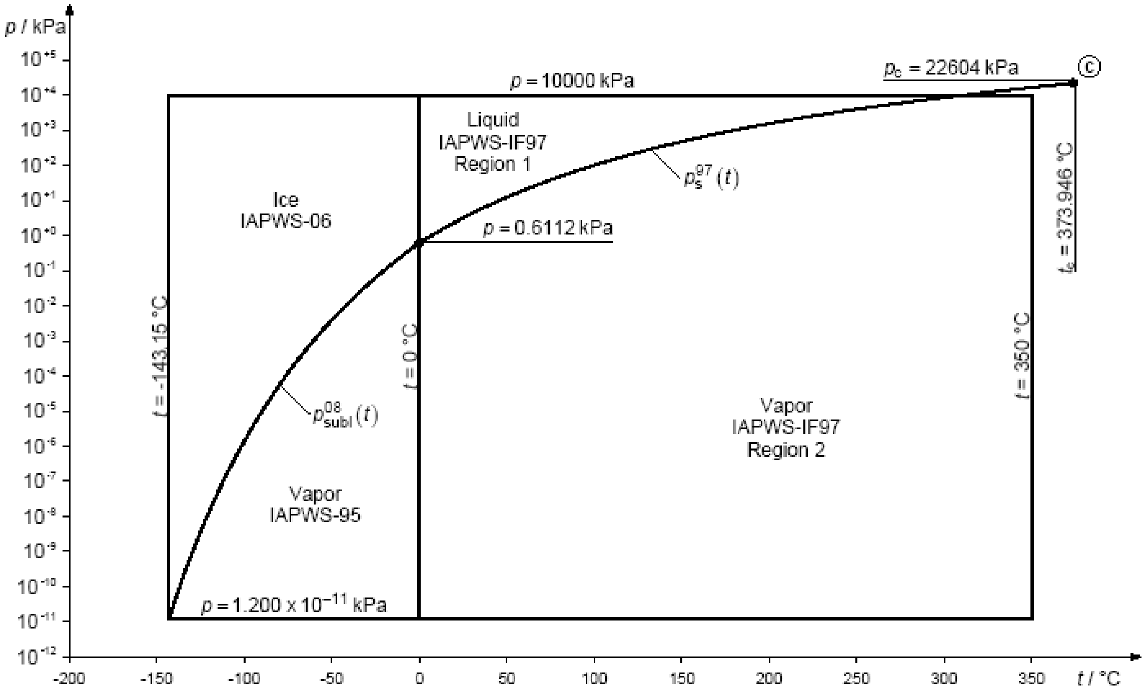


Figure 1: *p*-*T* diagram with used IAPWS formulations for steam, water, and ice.

2 Add-On FluidEES for Engineering Equation Solver® for ASHRAE-LibHuAirProp-SI

2.1 Installing FluidEES

The FluidEES Add-On has been developed to conveniently calculate thermodynamic properties in the Engineering Equation Solver® (EES). It enables, within EES, the direct call of functions relating to real moist air, steam, water and ice from the ASHRAE-LibHuAirProp property library.

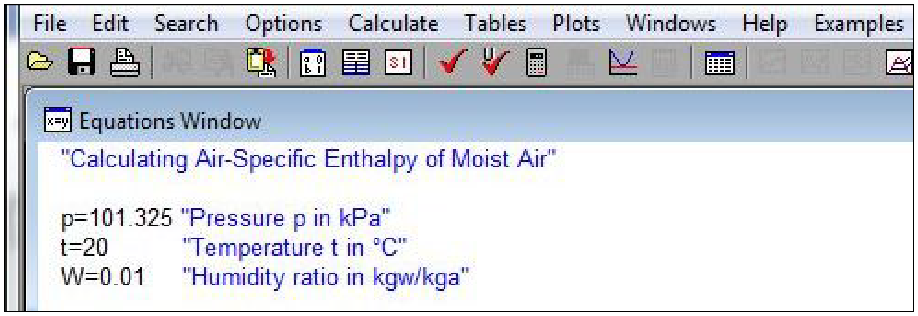
The installation of FluidEES and ASHRAE-LibHuAirProp_SI is described in Section 2.1.1 in "Part I-P Units" of this User's Guide.

2.2 Example: Calculation of *h* = f(*p*,*t*,*W*)

Now we will calculate, step by step, the air-specific enthalpy *h* of humid air as a function of total pressure *p*, temperature *t* and humidity ratio *W* for humid air, using FluidEES with LibHuAirProp_SI in the Engineering Equation Solver®.

How to perform a calculation with FluidEES:

- Start Engineering Equation Solver® (EES).
- The LibHuAirProp_SI library, if installed, is loaded by the program automatically.
- We recommend preparing an EES sheet, as shown in Figure 2.2.1.
  Note: the units of *p*, *t*, and *W* must correspond to those in Chapter 1.



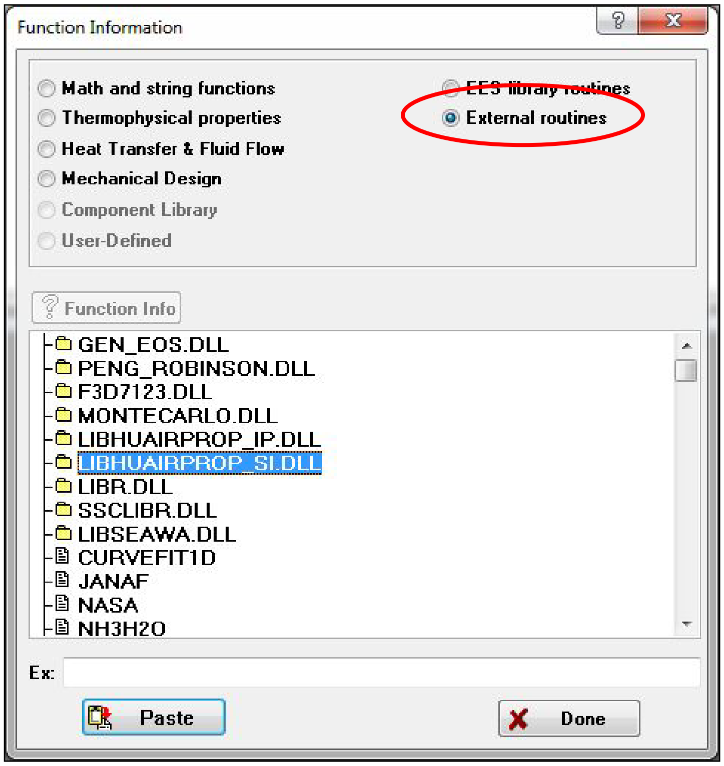
```
"Calculating Air-Specific Enthalpy of Moist Air"

p=101.325 "Pressure p in kPa"
t=20      "Temperature t in °C"
W=0.01    "Humidity ratio in kgw/kga"
```

Figure 2.2.1: Preparing an EES sheet for the calculation

- The function parameter values stand for:
  - First operand: Total pressure *p* = 101.325 kPa
    (Range of validity: *p* = 0.01 … 10 000 kPa)
  - Second operand: Temperature *t* = 20 °C
    (Range of validity: *t* = –143.15 ... 350°C)
  - Third operand: Humidity Ratio *W* = 0.01 kg_w/kg_a (kg water per kg dry air)
    (Range of validity: W = 0 … 10 kg_w/kg_a)
- Confirm your entry by pressing the "ENTER" key.

Note:
EES adapts to the language that is set in the "Regional and Language Options," which can be found in the "Control Panel." If you run Engineering Equation Solver® on an English version of Windows®, the standard decimal separator will be a dot (as shown in Fig. 2.2.1 and in the following sample calculation). If your computer is set to German, for example, the expected decimal separator will be a comma. In this case enter a comma in the values above instead of a dot. You can find additional information on this issue by clicking on "Help" in the EES menu bar and then select "Help Index". Click on "Search" in the window which appears, type "decimal separator" and press the "ENTER" key.

- For calculating *h* = f(*p*,*t*,*W*), call up the function "h_ptW_HAP_SI" of the property library LibHuAirProp_SI as follows:
- Click on "Options" in the EES menu bar and select "Function Info."
- The "Function Information" window will appear. Select "External routines" and you will see the screen shown here in Figure 2.2.2.

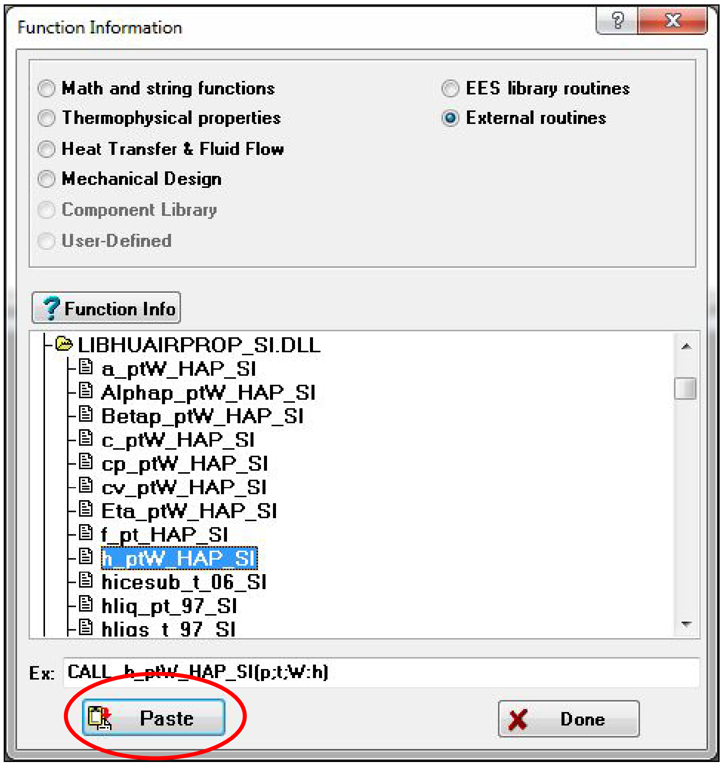


○ Math and string functions
○ Thermophysical properties
○ Heat Transfer & Fluid Flow
○ Mechanical Design
○ Component Library
○ User-Defined
○ EES library routines
● External routines

GEN_EOS.DLL
PENG_ROBINSON.DLL
F3D7123.DLL
MONTECARLO.DLL
LIBHUAIRPROP_IP.DLL
LIBHUAIRPROP_SI.DLL
LIBR.DLL
SSCLIBR.DLL
LIBSEAWA.DLL
CURVEFIT1D
JANAF
NASA
NH3H2O

Figure 2.2.2: "Function Information" window offering different libraries (routines)

- Double-click on the entry "LIBHUAIRPROP_SI.DLL".
- A list with calculable functions of the "LibHuAirProp_SI" library appears.
- Find and select the desired function, here "h_ptW_HAP_SI" function (see Figure 2.2.3) and click the "Paste" button below.



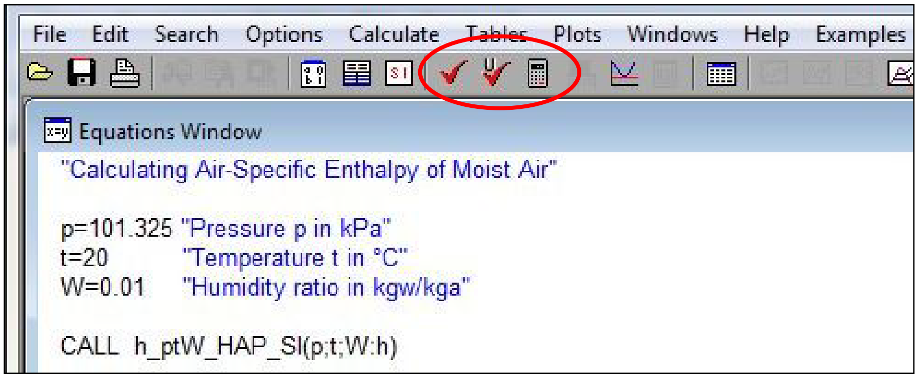
○ Math and string functions
○ Thermophysical properties
○ Heat Transfer & Fluid Flow
○ Mechanical Design
○ Component Library
○ User-Defined
○ EES library routines
● External routines

LIBHUAIRPROP_SI.DLL
- a_ptW_HAP_SI
- Alphap_ptW_HAP_SI
- Betap_ptW_HAP_SI
- c_ptW_HAP_SI
- cp_ptW_HAP_SI
- cv_ptW_HAP_SI
- Eta_ptW_HAP_SI
- f_pt_HAP_SI
- h_ptW_HAP_SI
- hicesub_t_06_SI
- hliq_pt_97_SI
- hliqs_t_97_SI

Ex: CALL h_ptW_HAP_SI(p;t;W:h)

Figure 2.2.3: Selecting the "h_ptW_HAP_SI" function

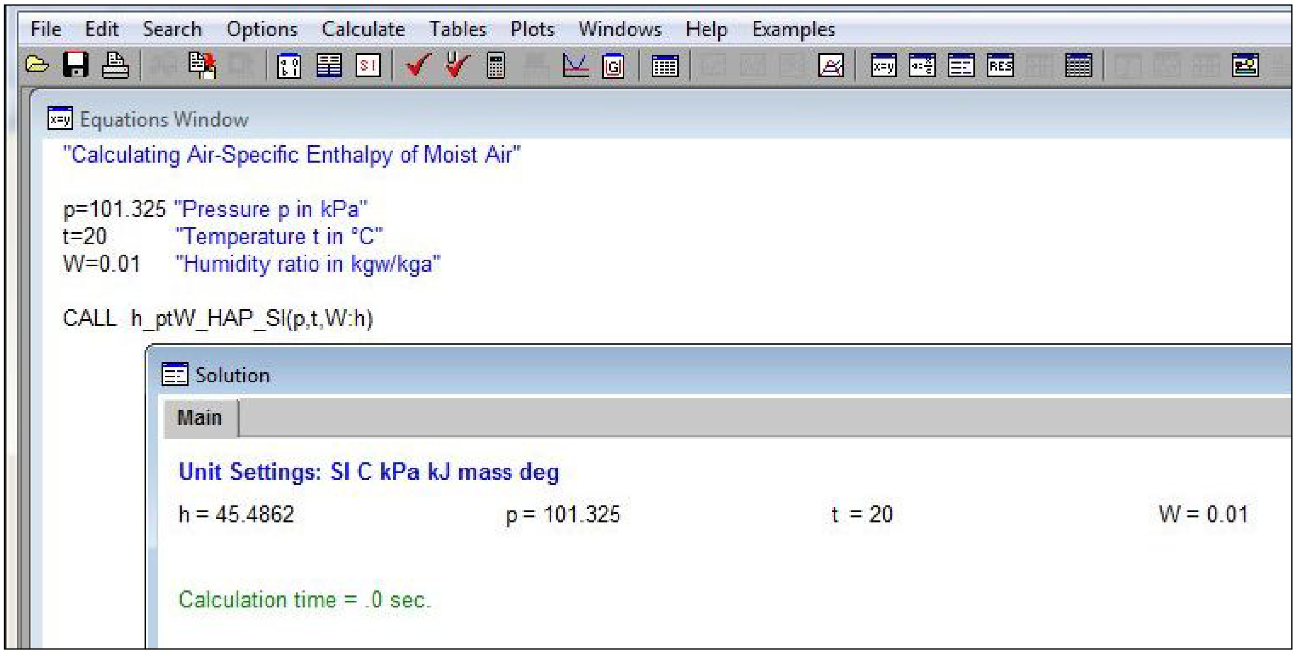
- The selected function will be copied and now appears in the "Equations Window" (see Fig. 2.2.4).



```
"Calculating Air-Specific Enthalpy of Moist Air"

p=101.325 "Pressure p in kPa"
t=20      "Temperature t in °C"
W=0.01    "Humidity ratio in kgw/kga"

CALL  h_ptW_HAP_SI(p;t;W:h)
```

Figure 2.2.4: "Equations Window" with the call of the property function

- Now, you can check the syntax of the instructions in the "Equations Window" by clicking the ✓ symbol in the upper menu bar of EES. The program tests whether or not the syntax is correct (e.g. dots as decimal separators versus commas). Confirm the "Information" window which appears by clicking the "OK" button.
- Then click the 🖩 symbol in the upper menu bar of EES to start the calculation.
- Soon you will see the "Calculations Completed" window. Leave this window by clicking the "Continue" button.
- The result for the air-specific enthalpy h appears in the "Solution" window.



```
"Calculating Air-Specific Enthalpy of Moist Air"

p=101.325 "Pressure p in kPa"
t=20      "Temperature t in °C"
W=0.01    "Humidity ratio in kgw/kga"

CALL  h_ptW_HAP_SI(p,t,W:h)
```

```
Unit Settings: SI C kPa kJ mass deg
h = 45.4862          p = 101.325          t = 20          W = 0.01

Calculation time = .0 sec.
```

Figure 2.2.5: "Solution" window showing the result

The calculation of *h* = f(*p*,*t*,*W*) has thus been carried out.

⇒ The result in our sample calculation here is: "h = 45.4862". The corresponding unit is kJ/kg (see table of the property functions in Chapter 1).

For further property functions calculable in FluidEES see the function table in Chapter 1.

2.3 Removing FluidEES including LibHuAirProp

The de-installation of FluidEES and ASHRAE-LibHuAirProp_SI is described in Section 2.4 in "Part I-P Units" of this User's Guide.

3 Property Functions of ASHRAE-LibHuAirProp-SI

3.1 Functions for Real Moist Air

Thermal Diffusivity $a = f(p,t,W)$

Function Name:

a_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION A_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in kg_w/kg_a

Result:

a_ptW_HAP_SI - Thermal diffusivity of humid air in m^2/s

Range of Validity:

Temperature t:	from -73.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le W_s$

Comments:

- Thermal diffusivity $a = \dfrac{\lambda}{\rho \cdot c_p}$
- Calculation for supersaturated humid air ($W > W_s$) is not possible

Result for Wrong Input Values:

a_ptW_HAP_SI = -1000

References:

$\lambda(p,t,W)$	Herrmann et al. [3], [4]
$\rho(p,t,W)$	Herrmann et al. [1], [2]
$c_p(p,t,W)$	Herrmann et al. [1], [2]

Relative Pressure Coefficient α_p = f(p,t,W)

Function Name:

alphap_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION ALPHAP_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

alphap_ptW_HAP_SI - Relative pressure coefficient of humid air in 1/K

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le W_\mathrm{s}$

Comments:

- Relative pressure coefficient $\alpha_p = \frac{1}{p}\left(\frac{\partial p}{\partial T}\right)_\mathrm{v}$
- Calculation for supersaturated humid air ($W > W_\mathrm{s}$) is not possible

Result for Wrong Input Values:

alphap_ptW_HAP_SI = -1000

References:

$\rho(p,t,W)$ Herrmann et al. [1], [2]

Isothermal Stress Coefficient β_p = f(p,t,W)

Function Name:

betap_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION BETAP_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

betap_ptW_HAP_SI - Isothermal stress coefficient of humid air in $\mathrm{kg/m^3}$

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le W_\mathrm{s}$

Comments:

- Isothermal stress coefficient $\beta_p = -\dfrac{1}{p}\left(\dfrac{\partial p}{\partial v}\right)_\mathrm{T}$
- Calculation for supersaturated humid air ($W > W_\mathrm{s}$) is not possible

Result for Wrong Input Values:

betap_ptW_HAP_SI = -1000

References:

$v(p,t,W)$ Herrmann et al. [1], [2]

Speed of Sound *c* = f(*p*,*t*,*W*)

Function Name:

c_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION   C_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa

t - Temperature t in °C

W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

c_ptW_HAP_SI - Speed of sound of humid air in m/s

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Speed of sound $c = v\sqrt{-\left(\frac{\partial p}{\partial v}\right)_s}$
- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

c_ptW_HAP_SI = -1000

References:

$v(p,t,W)$ Herrmann et al. [1], [2]

Specific Isobaric Heat Capacity c_p = f(p,t,W)

Function Name:

cp_ptW_HAP_SI

Fortran Program:

```
REAL*8 FUNCTION   CP_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W
```

Input Values:

p - Total pressure p in kPa

t - Temperature t in °C

W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

cp_ptW_HAP_SI - Specific isobaric heat capacity of humid air in kJ/(kg K)

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le W_s$

Comments:

- Specific isobaric heat capacity $c_p = \left(\frac{\partial h}{\partial T}\right)_p$
- Calculation for supersaturated humid air ($W > W_s$) is not possible

Result for Wrong Input Values:

cp_ptW_HAP_SI = -1000

References:

$h(p,t,W)$ Herrmann et al. [1], [2]

Specific Isochoric Heat Capacity c_v = f(p,t,W)

Function Name:

cv_ptW_HAP_SI

Fortran Program:

```
REAL*8 FUNCTION   CV_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W
```

Input Values:

p - Total pressure *p* in kPa
t - Temperature *t* in °C
W - Humidity ratio *W* in kg_w/kg_a

Result:

cv_ptW_HAP_SI - Specific isochoric heat capacity of humid air in kJ/(kg K)

Range of Validity:

Temperature *t*:	from -143.15°C to 350°C
Total pressure *p*:	from 0.01 kPa to 10 000 kPa
Humidity ratio *W*:	$0 \le W \le W_s$

Comments:

- Specific isochoric heat capacity $c_v = \left(\frac{\partial u}{\partial T}\right)_v$

- Calculation for supersaturated humid air ($W > W_s$) is not possible

Result for Wrong Input Values:

cv_ptW_HAP_SI = -1000

References:

$c_v(p,t,W)$ Herrmann et al. [3], [4]

Enhancement Factor *f* = f(*p*,*t*)

Function Name:

f_pt_HAP_SI

Fortran Program:

REAL*8 FUNCTION   F_PT_HUAIRPROP(P,T), REAL*8 P,T

Input Values:

p	-	Total pressure p in kPa
t	-	Temperature t in °C

Result:

f_pt_HAP_SI - Enhancement factor of water (decimal ratio)

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa

Comments:

- Enhancement factor $f = \dfrac{p_{\mathrm{H2O,s}}}{p_{\mathrm{s}}(t)}$

  with $p_{\mathrm{s}}(t)$ for $t \geq 0.01$°C - Steam pressure of water

  for $t < 0.01$°C - Sublimation pressure of water

- Describes the enhancement of the saturation pressure of water in the air atmosphere under elevated pressure
- Derived iteratively from the isothermal compressibility of liquid water, from Henry's constant [15], [16] and from the virial coefficients of air, water, and the air-water mixture

Result for Wrong Input Values:

f_pt_HAP_SI = -1000

References:

$f(p,t)$	Herrmann et al. [1], [2]

Air-Specific Enthalpy *h* = f(*p*,*t*,*W*)

Function Name:

h_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION H_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa

t - Temperature t in °C

W - Humidity ratio W in kg_w/kg_a

Result:

h_ptW_HAP_SI - Air-specific enthalpy in kJ/kg_a

Range of Validity:

Temperature t:	from -143.5°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,kg_w/kg_a$

Comments:

- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

h_ptW_HAP_SI = -1000

References:

$h(p,t,W)$	Herrmann et al. [1], [2]
$h_w(p,t)$	IAPWS-IF97 [7], [8] and IAPWS-06 [11]
$h_a(t)$	Lemmon et al. [14]

Dynamic Viscosity η = f(p,t,W)

Function Name:

Eta_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION ETA_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

Eta_ptW_HAP_SI - Dynamic viscosity of humid air in Pa s

Range of Validity:

Temperature t:	from -73.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- A new very accurate algorithm is implemented between 0°C and 350°C
- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

Eta_ptW_HAP_SI = -1000

References:

$\eta(p,t,W)$	Herrmann et al. [3], [4]
$\eta_a(t)$	Lemmon et al. [18]
$\eta_w(p,t)$	IAPWS-IF97 [7], [8] and IAPWS-08 [19]

Isentropic Exponent $\kappa = \mathrm{f}(p, t, W)$

Function Name:

Kappa_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION KAPPA_PTW_HUAIRPROP(P,T, W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

Kappa_ptW_HAP_SI - Isentropic exponent

Range of Validity:

Temperature t:	from -143.5°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Isentropic exponent $\kappa = -\frac{v}{p}\left(\frac{\partial p}{\partial v}\right)_s$
- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets homogeneously mixed) is applied for $t \geq 0.01$°C. For temperatures below (ice fog) the value of the saturated state is applied.

Result for Wrong Input Values:

Kappa_ptW_HAP_SI = -1000

References:

$v(p,t,W)$ Herrmann et al. [1], [2]

Thermal Conductivity $\lambda = \mathrm{f}(p,t,W)$

Function Name:

Lambda_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION LAMBDA_PTW_HUAIRPROP(P,T, W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

Lambda_ptW_HAP_SI - Thermal conductivity in W/(m K)

Range of Validity:

Temperature t:	from -73.5°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- A new very accurate algorithm is implemented between 0°C and 350°C
- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

Lambda_ptW_HAP_SI = -1000

References:

$\lambda(p,t,W)$	Herrmann et al. [3], [4]
$\lambda_\mathrm{a}(t)$	Lemmon et al. [18]
$\lambda_\mathrm{w}(p,t)$	IAPWS-IF97 [7], [8] and IAPWS-08 [20]

Kinematic Viscosity ν = f(p,t,W)

Function Name:

Ny_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION NY_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\text{kg}_\text{w}/\text{kg}_\text{a}$

Result:

Ny_ptW_HAP_SI - Kinematic viscosity in m^2/s

Range of Validity:

Temperature t:	from -73.5°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le W_\text{s}$

Comments:

- Kinematic Viscosity $\nu = \dfrac{\eta}{\rho}$

Result for Wrong Input Values:

Ny_ptW_HAP_SI = -1000

References:

$\eta(p,t,W)$	Herrmann et al. [3], [4]
$\rho(p,t,W)$	Herrmann et al. [1], [2]

Backward Function: Total Pressure p = f(t,s,W)

Function Name:

p_tsW_HAP_SI

Fortran Program:

REAL*8 FUNCTION   P_TSW_HUAIRPROP(T,S,W), REAL*8 T,S,W

Input Values:

t - Temperature t in °C

s - Air-specific entropy s in kJ/(kg$_a$ K)

W - Humidity ratio W in kg$_w$/kg$_a$

Result:

p_tsW_HAP_SI - Total pressure in kPa

Range of Validity:

Temperature t:	from -143.5°C to 350°C
Air-specific entropy s:	from -26.53 kJ/(kg$_a$ K) to 38.990 kJ/(kg$_a$ K)
Humidity ratio W:	$0 \le W \le 10$ kg$_w$/kg$_a$

Comments:

- Iteration of total pressure p from s = f(p,t,W)

Result for Wrong Input Values:

p_tsW_HAP_SI = -1000

References:

$s(p,t,W)$ Herrmann et al. [1], [2]

Pressure $p = \mathrm{f}(z_{\mathrm{ele}})$

Function Name:

p_zele_HAP_SI

Fortran Program:

REAL*8 FUNCTION   P_ZELE_HUAIRPROP(ZELE), REAL*8 ZELE

Input Values:

z_{ele} - Elevation z_{ele} in m

Result:

p_zele_HAP_SI - Pressure of humid air in kPa

Range of Validity:

Elevation z_{ele}	from -5,000 m to 11,000 m

Comments:

- Pressure of humid air from elevation
- $p(z_{\mathrm{ele}}) = 101.325\,\mathrm{kPa} \cdot \left(1 - 2.25577 \cdot 10^{-5} \cdot \frac{z_{\mathrm{ele}}}{\mathrm{m}}\right)^{5.256}$

Result for Wrong Input Values:

p_zele_HAP_SI = -1000

References:

$p(z_{\mathrm{ele}})$	ASHRAE [23]

Partial Pressure of Dry Air p_{Air} = f(p,t,W)

Function Name:

pAir_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION PAIR_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

pAir_ptW_HAP_SI - Partial pressure of (dry) air in humid air in kPa

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Partial pressure of (dry) air in humid air $p_{Air} = 1 - p_{H2O}$
- Partial pressure of water vapor at saturation is calculated in case of supersaturated humid air ($W > W_s(p,t)$)
- The temperature value is used to calculate the saturation state

Result for Wrong Input Values:

pAir_ptW_HAP_SI = -1000

References:

$p_{H2O}(p,W)$ Herrmann et al. [1], [2]

Partial Pressure of Water Vapor p_{H2O} = f(p, t, W)

Function Name:

pH2O_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION PH2O_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

pH2O_ptW_HAP_SI - Partial pressure of water vapor in humid air in kPa

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Partial pressure of water vapor in humid air $p_{H2O} = \dfrac{W \cdot p}{\left(\dfrac{R_a}{R_w} + W\right)}$
- Partial pressure of water vapor at saturation is calculated in case of supersaturated humid air ($W > W_s(p,t)$)
- The temperature value is used to calculate the saturation state

Result for Wrong Input Values:

pH2O_ptW_HAP_SI = -1000

References:

$p_{H2O}(p,W)$ Herrmann et al. [1], [2]

Partial Saturation Pressure of Water Vapor $p_{H2O,s}$ = f(p, t)

Function Name:

pH2Os_pt_HAP_SI

Fortran Program:

```
REAL*8 FUNCTION   PH2OS_PT_HUAIRPROP(P,T), REAL*8 P,T
```

Input Values:

p	-	Total pressure *p* in kPa
t	-	Temperature *t* in °C

Result:

pH2Os_pt_HAP_SI - Partial saturation pressure of water vapor in humid air in kPa

Range of Validity:

Temperature *t*:	from -143.15°C to 350°C
Total pressure *p*:	from 0.01 kPa to 10 000 kPa

Comments:

- Partial pressure of steam at saturation $p_{H2O,s} = f \cdot p_s(t)$

with $p_s(t)$ for $t \geq 0.01$°C - Steam pressure of water

for $t < 0.01$°C - Sublimation pressure of water

Result for Wrong Input Values:

pH2Os_pt_HAP_SI = -1000

References:

$f(p,t)$		Herrmann et al. [1], [2]
$p_s(t)$	for $t \geq 0.01$°C	IAPWS-IF97 [7], [8]
	for $t < 0.01$°C	IAPWS-08 [11]

Relative Humidity φ = f(p,t,W)

Function Name:

phi_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION PHI_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa

t - Temperature t in °C

W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

phi_ptW_HAP_SI - Relative humidity (decimal ratio)

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Relative humidity $\varphi = \dfrac{p_{\mathrm{H2O}}}{p_{\mathrm{H2O,s}}}$
- This equation is valid for $p_{\mathrm{H2O}} \le p_{\mathrm{H2O,s}}$ and for $0 \le \varphi \le 1$

Result for Wrong Input Values:

phi_ptW_HAP_SI = -1000

References:

$\varphi(p,t,W)$ Herrmann et al. [1], [2]

Prandtl Number $Pr = f(p,t,W)$

Function Name:

Pr_ptW_HAP_SI

Fortran Program:

```
REAL*8 FUNCTION   PR_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W
```

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

Pr_ptW_HAP_SI - Prandtl number

Range of Validity:

Temperature t:	from -73.5°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Prandtl number $Pr = \dfrac{\eta \cdot c_p}{\lambda}$
- Calculation for supersaturated humid air ($W > W_s$) is not possible

Result for Wrong Input Values:

Pr_ptW_HAP_SI = -1000

References:

$\eta(p,t,W)$	Herrmann et al. [3], [4]
$c_p(p,t,W)$	Herrmann et al. [3], [4]
$\lambda(p,t,W)$	Lemmon et al. [20]

Mole Fraction of Dry Air ψ_{Air} = f(W)

Function Name:

PsiAir_W_HAP_SI

Fortran Program:

REAL*8 FUNCTION PSIAIR_W_HUAIRPROP(W), REAL*8 W

Input Values:

W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

PsiAir_W_HAP_SI - Mole fraction of (dry) air in humid air in $\mathrm{mol_a/mol}$

Range of Validity:

Humidity ratio W: $0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Mole fraction of air $\psi_{\mathrm{Air}} = 1 - \psi_{\mathrm{H2O}} = 1 - \left(\dfrac{W}{\dfrac{R_{\mathrm{a}}}{R_{\mathrm{H2O}}} + W} \right)$

Result for Wrong Input Values:

PsiAir_W_HAP_SI = -1000

References:

$\psi_{\mathrm{Air}}(W)$ Herrmann et al. [1], [2]

Mole Fraction of Water $\psi_{\mathbf{H2O}}$ = f($\boldsymbol{W}$)

Function Name:

PsiH2O_W_HAP_SI

Fortran Program:

REAL*8 FUNCTION PSIH2O_W_HUAIRPROP(W), REAL*8 W

Input Values:

W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

PsiH2O_W_HAP_SI - Mole fraction of water in humid air in $\mathrm{mol_w/mol}$

Range of Validity:

Humidity ratio W: $0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Mole fraction of water $\psi_{\mathrm{H2O}} = \dfrac{W}{\dfrac{R_\mathrm{a}}{\mathrm{R_{H2O}}} + W}$

Result for Wrong Input Values:

PsiH2O_W_HAP_SI = -1000

References:

$\psi_{\mathrm{H2O}}(W)$ Herrmann et al. [1], [2]

Density ρ = f(p,t,W)

Function Name:

Rho_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION RHO_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

Rho_ptW_HAP_SI - Density of humid air in $\mathrm{kg/m^3}$

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Density of humid air obtained from air-specific volume: $\rho = \frac{1+W}{v}$

Result for Wrong Input Values:

Rho_ptW_HAP_SI = -1000

References:

$\rho(p,t,W)$ Herrmann et al. [1], [2]

Air-Specific Entropy s = f(p,t,W)

Function Name:

s_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION S_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

s_ptW_HAP_SI - Air-specific entropy in $\mathrm{kJ/(kg_a\,K)}$

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \leq W \leq 10\,\mathrm{kg_w/kg_a}$

Comments:

- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

s_ptW_HAP_SI = -1000

References:

$s(p,t,W)$ Herrmann et al. [1], [2]

Backward Function: Temperature *t* = f(*p*,*h*,*φ*)

Function Name:

t_phphi_HAP_SI

Fortran Program:

REAL*8 FUNCTION T_PHPHI_HUAIRPROP(P,H,PHI), REAL*8 P,H,PHI

Input Values:

p	-	Total pressure p in kPa
h	-	Air-specific enthalpy h in kJ/kg_a
φ	-	Relative humidity φ (decimal ratio)

Result:

t_phphi_HAP_SI - Temperature from pressure, enthalpy, and relative humidity in °C

Range of Validity:

Total pressure p:	from 0.01 kPa to 10 000 kPa
Air-specific enthalpy h:	from -5745 kJ/kg_a to 29690 kJ/kg_a
Relative humidity φ:	$0 \le \varphi \le 1$

Comments:

- Iteration of temperature t from $h = f(p,t,W)$ using $W = f(p,t,\varphi)$

Result for Wrong Input Values:

t_phphi_HAP_SI = -1000

References:

$h(p,t,W)$ Herrmann et al. [1], [2]

Backward Function: Temperature $t = \mathrm{f}(p,h,W)$

Function Name:

t_phW_HAP_SI

Fortran Program:

REAL*8 FUNCTION   T_PHW_HUAIRPROP(P,H,W), REAL*8 P,H,W

Input Values:

p - Total pressure *p* in kPa
h - Air-specific enthalpy *h* in $\mathrm{kJ/kg_a}$
W - Humidity ratio *W* in $\mathrm{kg_w/kg_a}$

Result:

t_phW_HAP_SI - Temperature from pressure, enthalpy, and humidity ratio in °C

Range of Validity:

Total pressure *p*:	from 0.01 kPa to 10 000 kPa
Air-specific enthalpy *h*:	from -5745 $\mathrm{kJ/kg_a}$ to 29690 $\mathrm{kJ/kg_a}$
Humidity ratio *W*:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Iteration of temperature *t* from $h = \mathrm{f}(p,t,W)$

Result for Wrong Input Values:

t_phW_HAP_SI = -1000

References:

$h(p,t,W)$ Herrmann et al. [1], [2]

Backward Function: Temperature *t* = f(*p*,*s*,*W*)

Function Name:

t_psW_HAP_SI

Fortran Program:

REAL*8 FUNCTION   T_PSW_HUAIRPROP(P,S,W), REAL*8 P,S,W

Input Values:

p	-	Total pressure *p* in kPa
s	-	Air-specific entropy *s* in kJ/(kg_a K)
W	-	Humidity ratio *W* in kg_w/kg_a

Result:

t_psW_HAP_SI - Temperature from pressure, entropy, and humidity ratio in °C

Range of Validity:

Total pressure *p*:	from 0.01 kPa to 10 000 kPa
Air-specific entropy *s*:	from -26.53 kJ/(kg_a K) to 38.990 kJ/(kg_a K)
Humidity ratio *W*:	$0 \le W \le 10\,kg_w/kg_a$

Comments:

- Iteration of temperature *t* from *s* = f(*p*,*t*,*W*)

Result for Wrong Input Values:

t_psW_HAP_SI = -1000

References:

$s(p,t,W)$ Herrmann et al. [1], [2]

Backward Function: Temperature t = f(p,t_{wb},W)

Function Name:

t_ptwbW_HAP_SI

Fortran Program:

```
REAL*8 FUNCTION   T_PTWBW_HUAIRPROP(P,TWB,W), REAL*8 P,TWB,W
```

Input Values:

p - Total pressure p in kPa

t_{wb} - Wet-bulb temperature in °C

W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

t_ptwbW_HAP_SI - Temperature from pressure, wet bulb temperature and humidity ratio in °C

Range of Validity:

Total pressure p:	from 0.01 kPa to 10 000 kPa
Wet bulb temperature t_{wb}:	from -143.15°C to 350°C
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Iterationof temperature t from $t_{\mathrm{wb}} = \mathrm{f}(p,t,W)$

Result for Wrong Input Values:

t_ptwbW_HAP_SI = -1000

References:

$t_{\mathrm{wb}}(p,t,W)$ Herrmann et al. [1], [2]

Dew-Point/Frost-Point Temperature t_d = f(p,W)

Function Name:

td_pW_HAP_SI

Fortran Program:

REAL*8 FUNCTION TD_PW_HUAIRPROP(P,W), REAL*8 P,W

Input Values:

p - Total pressure p in kPa

W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

td_pW_HAP_SI - Dew-point/frost-point temperature in °C

Range of Validity:

Total pressure p: from 0.01 kPa to 10 000 kPa

Humidity ratio W: $0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

Dew-point temperature $t_\mathrm{d} = t_\mathrm{s}(p_\mathrm{H2O})$ for $t \ge 0.01°\mathrm{C}$ (saturation temperature of water in humid air)

$t_\mathrm{d} = t_\mathrm{sub}(p_\mathrm{H2O})$ for $t \le 0.01°\mathrm{C}$ (sublimation temperature of water in humid air)

Result for Wrong Input Values:

td_pW_HAP_SI = -1000

References:

$t_\mathrm{s}(p_\mathrm{H2O})$	for $t_\mathrm{d} \ge 0.01°\mathrm{C}$	IAPWS-IF97 [7], [8]
$t_\mathrm{sub}(p_\mathrm{H2O})$	for $t_\mathrm{d} \le 0.01°\mathrm{C}$	IAPWS-08 [11]
p_H2O		Herrmann et. al. [1], [2]

Saturation Temperature t_s = f(p, p_{H2O})

Function Name:

ts_ppH2O_HAP_SI

Fortran Program:

REAL*8 FUNCTION TS_PPH2O_HUAIRPROP(P,PH2O), REAL*8 P,PH2O

Input Values:

p - Total pressure p in kPa
p_{H2O} - Partial pressure of water vapor p_{H2O} in kPa

Result:

ts_ppH2O_HAP_SI - Saturation temperature in °C

Range of Validity:

Total pressure p:	from 0.01 kPa to 10 000 kPa
Partial Pressure p_{H2O} :	from 0.01 kPa to 10 000 kPa

Comments:

- Iteration of saturation temperature t_s from $p_{H2O,s} = f(p,t)$

Result for Wrong Input Values:

ts_ppH2O_HAP_SI = -1000

References:

$p_{H2O,s}$ Herrmann et. al. [1], [2]

Wet-Bulb/Ice-Bulb Temperature t_{wb} = f(p,t,W)

Function Name:

twb_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION TWB_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

twb_ptW_HAP_SI - Wet-bulb/ice-bulb temperature in °C

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Iteration of wet-bulb/ice-bulb temperature t_{wb} from $h^{\mathrm{unsaturated}}(p,t,W) = h^{\mathrm{fog}}(p,t_{\mathrm{wb}},W)$
- Calculation for supersaturated humid air ($W > W_{\mathrm{s}}$) is not possible

Result for Wrong Input Values:

twb_ptW_HAP_SI = -1000

References:

$t_{\mathrm{wb}}(p,t,W)$ Herrmann et al. [1], [2]

Air-Specific Internal Energy *u* = f(*p*,*t*,*W*)

Function Name:

u_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION U_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

u_ptW_HAP_SI - Air-specific internal energy in $\mathrm{kJ/kg_a}$

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Humidity ratio W:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Internal energy $u = h - pv$

Result for Wrong Input Values:

u_ptW_HAP_SI = -1000

References:

$u(p,t,W)$ Herrmann et al. [1], [2]

Air-Specific Volume $v = f(p,t,W)$

Function Name:

v_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION V_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

v_ptW_HAP_SI - Air-specific volume in $\mathrm{m^3/kg_a}$

Range of Validity:

Temperature *t*:	from -143.15°C to 350°C
Total pressure *p*:	from 0.01 kPa to 10 000 kPa
Humidity ratio *W*:	$0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- When calculating supersaturated air an ideal mixing model (saturated humid air and water droplets (or ice crystals) homogeneously mixed) is applied

Result for Wrong Input Values:

v_ptW_HAP_SI = -1000

References:

$v(p,t,W)$ Herrmann et al. [1], [2]

Humidity Ratio from Partial Pressure of Steam $W = f(p,t,p_{H2O})$

Function Name:

W_ptpH2O_HAP_SI

Fortran Program:

REAL*8 FUNCTION W_PTPH2O_HUAIRPROP(P,T,PH2O), REAL*8 P,T,PH2O

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
p_{H2O} - Partial pressure of water p_{H2O} in kPa

Result:

W_ptpH2O_HAP_SI - Humidity ratio from temperature and partial pressure of water vapor in kg_w/kg_a

Range of Validity:

Total pressure p:	from 0.01 kPa to 10 000 kPa
Temperature t :	from -143.15°C to 350°C
Partial pressure p_{H2O} :	from 0.01 kPa to 10 000 kPa

Comments:

- Iteration of humidity ratio W from $p_{H2O} = f(p,t,W)$
- Result for supersaturated humid air is W_s

Result for Wrong Input Values:

W_ptpH2O_HAP_SI = -1000

References:

$p_{H2O}(p,t,W)$ Herrmann et al. [1], [2]

Humidity Ratio from Relative Humidity $W = f(p, t, \varphi)$

Function Name:

W_ptphi_HAP_SI

Fortran Program:

REAL*8 FUNCTION W_PTPHI_HUAIRPROP(P,T,PHI), REAL*8 P,T,PHI

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
φ - Relative humidity (decimal ratio)

Result:

W_ptphi_HAP_SI - Humidity ratio from temperature and relative humidity in kg_w/kg_a

Range of Validity:

Temperature t:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa
Relative humidity φ:	$0 \le \varphi \le 1$

Comments:

- Iteration of humidity ratio W from $\varphi = f(p, t, W)$

Result for Wrong Input Values:

W_ptphi_HAP_SI = -1000

References:

$\varphi(p, t, W)$ Herrmann et al. [1], [2]

Humidity Ratio from Dew-Point Temperature $W = f(p,t_d)$

Function Name:

W_ptd_HAP_SI

Fortran Program:

REAL*8 FUNCTION W_PTD_HUAIRPROP(P,TD), REAL*8 P,TD

Input Values:

p - Total pressure p in kPa
t_d - Dew-point temperature t_d in °C

Result:

W_ptd_HAP_SI - Humidity ratio from temperature and dew-point temperature in kg_w/kg_a

Range of Validity:

Dew point temperature t_d:	from -143.15°C to 350°C
Total pressure p:	from 0.01 kPa to 10 000 kPa

Comments:

- Iteration of humidity ratio W from $t_d = f(p,W)$

Result for Wrong Input Values:

W_ptd_HAP_SI = -1000

References:

$t_d(p,W)$ Herrmann et al. [1], [2]

Humidity Ratio from Wet-Bulb Temperature $W = f(p,t,t_{wb})$

Function Name:

W_pttwb_HAP_SI

Fortran Program:

REAL*8 FUNCTION W_PTTWB_HUAIRPROP(P,T,TWB), REAL*8 P,T,TWB

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
t_{wb} - Wet-bulb temperature in °C

Result:

W_pttwb_HAP_SI - Humidity ratio from temperature and wet-bulb temperature in kg_w/kg_a

Range of Validity:

Total pressure p:	from 0.01 kPa to 10 000 kPa
Temperature t:	from -143.15°C to 350°C
Wet-bulb temperature t_{wb}:	from -143.15°C to 350°C

Comments:

- Iteration of humidity ratio W from $t_{wb} = f(p,t,W)$
- Calculation for supersaturated humid air ($W > W_s$) is not possible

Result for Wrong Input Values:

W_pttwb_HAP_SI = -1000

References:

$t_{wb}(p,t,W)$ Herrmann et al. [1], [2]

Saturation Humidity Ratio W_s = f(p,t)

Function Name:

Ws_pt_HAP_SI

Fortran Program:

REAL*8 FUNCTION WS_PT_HUAIRPROP(P,T), REAL*8 P,T

Input Values:

p - Total pressure *p* in kPa
t - Temperature *t* in °C

Result:

Ws_pt_HAP_SI - Saturation humidity ratio (mass fraction) in kg_w/kg_a

Range of Validity:

Total pressure *p*:	from 0.01 kPa to 10 000 kPa
Temperature *t*:	from -143.15°C to 350°C

Comments:

- Calculation of saturation humidity ratio W_s from $W_s = \frac{M_{H2O}}{M_a}\frac{p_{H2O,s}}{\left(p - p_{H2O,s}\right)}$

Result for Wrong Input Values:

Ws_pt_HAP_SI = -1000

References:

$p_{H2O,s}$ Herrmann et al. [1], [2]

Mass Fraction of Dry Air ξ_{Air} = f(W)

Function Name:

XiAir_W_HAP_SI

Fortran Program:

REAL*8 FUNCTION XIAIR_W_HUAIRPROP(W), REAL*8 W

Input Values:

W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

XiAir_W_HAP_SI - Mass fraction of (dry) air in humid air in $\mathrm{kg_a/kg}$

Range of Validity:

Humidity ratio W: $0 \le W \le 10\,\mathrm{kg_w/kg_a}$

Comments:

- Mass fraction of (dry) air $\xi_{\mathrm{Air}} = 1 - \xi_{\mathrm{H2O}} = 1 - \dfrac{W}{1+W}$

Result for Wrong Input Values:

XiAir_W_HAP_SI = -1000

References:

$\xi_{\mathrm{Air}}(W)$ Herrmann et al. [1], [2]

Mass Fraction of Water Vapor ξ_{H2O} = f(W)

Function Name:

XiH2O_W_HAP_SI

Fortran Program:

REAL*8 FUNCTION XIH2O_W_HUAIRPROP(W), REAL*8 W

Input Values:

W - Humidity ratio W in kg_w/kg_a

Result:

XiH2O_W_HAP_SI - Mass fraction of water vapor in humid air in kg_w/kg

Range of Validity:

Humidity ratio W: $0 \le W \le 10\, kg_w/kg_a$

Comments:

- Mass fraction of water vapor $\xi_{H2O} = \frac{W}{1+W}$

Result for Wrong Input Values:

XiH2O_W_HAP_SI = -1000

References:

$\xi_{H2O}(W)$ Herrmann et al. [1], [2]

Compression Factor Z = f(p,t,W)

Function Name:

Z_ptW_HAP_SI

Fortran Program:

REAL*8 FUNCTION Z_PTW_HUAIRPROP(P,T,W), REAL*8 P,T,W

Input Values:

p - Total pressure p in kPa
t - Temperature t in °C
W - Humidity ratio W in $\mathrm{kg_w/kg_a}$

Result:

Z_ptW_HAP_SI - Compression factor (decimal ratio)

Range of Validity:

Total pressure p:	from 0.01 kPa to 10 000 kPa
Temperature t:	from -143.15°C to 350°C
Humidity ratio W:	$0 \le W \le W_\mathrm{s}$

Comments:

- Compression factor $Z = 1 + \dfrac{B_\mathrm{m}}{\bar{v}} + \dfrac{C_\mathrm{m}}{\bar{v}^2}$

  with $\bar{v} = \dfrac{M}{\rho} = \dfrac{M\,v}{1+W}$

  and M is the molar mass of humid air

- Calculation for supersaturated humid air ($W > W_\mathrm{s}$) is not possible

Result for Wrong Input Values:

Z_ptW_HAP_SI = -1000

References:

$B_\mathrm{m}(t,W), C_\mathrm{m}(t,W)$	Herrmann et al. [1], [2]
$\rho(p,t,W), v(p,t,W)$	Herrmann et al. [1], [2]

3.2 Functions for Steam and Water for Temperatures *t* ≥ 0°C

Specific Enthalpy of Liquid Water $h_{\mathrm{liq}} = \mathrm{f}(p, t)$

Function Name:

hliq_pt_97_SI

Fortran Program:

REAL*8 FUNCTION HLIQ_PT_97(P,T), REAL*8 P,T

Input Values:

p	-	Pressure p in kPa
t	-	Temperature t in °C

Result:

hliq_pt_97_SI - Specific enthalpy of liquid water in kJ/kg

Range of Validity:

Pressure p:	from $p_{\mathrm{s}}(0°\mathrm{C}) = 0.6112$ kPa to 10 000 kPa
Temperature t:	from 0°C to 350°C

Comments:

- Specific enthalpy of liquid water $h_{\mathrm{liq}} = h^{97}(p,t)$ (Region 1)

Result for Wrong Input Values:

hliq_pt_97_SI = -1000

References:

$h^{97}(p,t)$	IAPWS-IF97 [7], [8]

Specific Enthalpy of Saturated Liquid Water $h_{\mathrm{liq,s}} = \mathrm{f}(t)$

Function Name:

hliqs_t_97_SI

Fortran Program:

REAL*8 FUNCTION   HLIQS_T_97(T), REAL*8 T

Input Values:

t   -   Temperature *t* in °C

Result:

hliqs_t_97_SI - Specific enthalpy of saturated liquid water in kJ/kg

Range of Validity:

Temperature *t*:   from 0°C to 350°C

Comments:

- Specific enthalpy of liquid water $h_{\mathrm{liq,s}} = h^{97}(p_{\mathrm{s}}, t)$ (Region 1)
  with $p_{\mathrm{s}} = p_{\mathrm{s}}^{97}(t)$

Result for Wrong Input Values:

hliqs_t_97_SI = -1000

References:

$h^{97}(p,t)$, $p_{\mathrm{s}}^{97}(t)$ IAPWS-IF97 [7], [8]

Specific Enthalpy of Saturated Water Vapor $h_{\mathrm{vap,s}} = \mathrm{f}(t)$

Function Name:

hvaps_t_97_SI

Fortran Program:

REAL*8 FUNCTION HVAPS_T_97(T), REAL*8 T

Input Values:

t - Temperature *t* in °C

Result:

hvaps_t_97_SI - Specific enthalpy of saturated water vapor in kJ/kg

Range of Validity:

Temperature *t*: from 0°C to 350°C

Comments:

- Specific enthalpy of saturated water vapor $h_{\mathrm{vap,s}} = h^{97}(p_{\mathrm{s}}, t)$ (Region 2)
  with $p_{\mathrm{s}} = p_{\mathrm{s}}^{97}(t)$

Result for Wrong Input Values:

hvaps_t_97_SI = -1000

References:

$h^{97}(p,t)$, $p_{\mathrm{s}}^{97}(t)$ IAPWS-IF97 [7], [8]

Saturation Pressure of Water p_s = f(t)

Function Name:

ps_t_97_SI

Fortran Program:

REAL*8 FUNCTION PS_T_97(T), REAL*8 T

Input Values:

t - Temperature *t* in °C

Result:

ps_t_97_SI - Saturation pressure of water in kPa

Range of Validity:

Temperature *t*: from 0°C to 350°C

Comments:

- Saturation pressure of water $p_s = p_s^{97}(t)$ (Region 4)

Result for Wrong Input Values:

ps_t_97_SI -1000

References:

$p_s^{97}(t)$ IAPWS-IF97 [7], [8]

Specific Entropy of Liquid Water $s_{\mathrm{liq}} = \mathrm{f}(p, t)$

Function Name:

sliq_pt_97_SI

Fortran Program:

```
REAL*8 FUNCTION   SLIQ_PT_97(P,T), REAL*8 P,T
```

Input Values:

p	-	Pressure p in kPa
t	-	Temperature t in °C

Result:

sliq_pt_97_SI - Specific entropy of liquid water in kJ/(kg K)

Range of Validity:

Pressure p:	from $p_{\mathrm{s}}(0°\mathrm{C}) = 0.6112$ kPa to 10 000 kPa
Temperature t:	from 0°C to 350°C

Comments:

- Specific entropy of liquid water $s_{\mathrm{liq}} = s^{97}(p, t)$ (Region 1)

Result for Wrong Input Values:

sliq_pt_97_SI = -1000

References:

$s^{97}(p, t)$	IAPWS-IF97 [7], [8]

Specific Entropy of Saturated Liquid Water $s_{\mathrm{liq,s}}$ = f(t)

Function Name:

sliqs_t_97_SI

Fortran Program:

REAL*8 FUNCTION   SLIQS_T_97(T), REAL*8 T

Input Values:

t - Temperature *t* in °C

Result:

sliqs_t_97_SI - Specific entropy of saturated liquid water in kJ/(kg K)

Range of Validity:

Temperature *t*:	from 0°C to 350°C

Comments:

- Specific entropy of liquid water $s_{\mathrm{liq,s}} = s^{97}(p_{\mathrm{s}}, t)$ (Region 1) with $p_{\mathrm{s}} = p_{\mathrm{s}}^{97}(t)$

Result for Wrong Input Values:

sliqs_t_97_SI = -1000

References:

$s^{97}(p,t)$, $p_{\mathrm{s}}^{97}(t)$	IAPWS-IF97 [7], [8]

Specific Entropy of Saturated Water Vapor $s_{\mathrm{vap,s}} = \mathrm{f}(t)$

Function Name:

svaps_t_97_SI

Fortran Program:

REAL*8 FUNCTION SVAPS_T_97(T), REAL*8 T

Input Values:

t - Temperature *t* in °C

Result:

svaps_t_97_SI - Specific entropy of saturated water vapor in kJ/(kg K)

Range of Validity:

Temperature *t*: from 0°C to 350°C

Comments:

- Specific entropy of saturated water vapor $s_{\mathrm{vap,s}} = s^{97}(p_{\mathrm{s}}, t)$ (Region 2)
  with $p_{\mathrm{s}} = p_{\mathrm{s}}^{97}(t)$

Result for Wrong Input Values:

svaps_t_97_SI = -1000

References:

$s^{97}(p,t)$, $p_{\mathrm{s}}^{97}(t)$ IAPWS-IF97 [7], [8]

Saturation Temperature of Water $t_s = f(p)$

Function Name:

ts_p_97_SI

Fortran Program:

REAL*8 FUNCTION   TS_P_97(P), REAL*8 P

Input Values:

p -   Pressure p in kPa

Result:

ts_p_97_SI - Saturation temperature of water in °C

Range of Validity:

Pressure p:	from 0.6112 kPa to 10 000 kPa

Comments:

- Saturation temperature of water $t_s = t_s^{97}(p)$ (Region 4)

Result for Wrong Input Values:

ts_p_97_SI = -1000

References:

$t_s^{97}(p)$	IAPWS-IF97 [7], [8]

Specific Volume of Liquid Water $v_{\text{liq}} = f(p,t)$

Function Name:

vliq_pt_97_SI

Fortran Program:

```
REAL*8 FUNCTION   VLIQ_PT_97(P,T), REAL*8 P,T
```

Input Values:

p - Pressure p in kPa
t - Temperature t in °C

Result:

vliq_pt_97_SI - Specific volume of liquid water in m^3/kg

Range of Validity:

Pressure p:	from $p_s(0°C) = 0.6112$ kPa to 10 000 kPa
Temperature t:	from 0°C to 350°C

Comments:

- Specific volume of liquid water $v_{\text{liq}} = v^{97}(p,t)$ (Region 1)

Result for Wrong Input Values:

vliq_pt_97_SI = -1000

References:

$v^{97}(p,t)$	IAPWS-IF97 [7], [8]

Specific Volume of Saturated Liquid Water $v_{\mathrm{liq,s}} = \mathrm{f}(t)$

Function Name:

vliqs_t_97_SI

Fortran Program:

REAL*8 FUNCTION VLIQS_T_97(T), REAL*8 T

Input Values:

t - Temperature *t* in °C

Result:

vliqs_t_97_SI - Specific volume of saturated liquid water in m^3/kg

Range of Validity:

Temperature *t*: from 0°C to 350°C

Comments:

- Specific volume of liquid water $v_{\mathrm{liq,s}} = v^{97}(p_{\mathrm{s}}, t)$ (Region 1)
  with $p_{\mathrm{s}} = p_{\mathrm{s}}^{97}(t)$

Result for Wrong Input Values:

vliqs_t_97_SI = -1000

References:

$v^{97}(p,t)$, $p_{\mathrm{s}}^{97}(t)$ IAPWS-IF97 [7], [8]

Specific Volume of Saturated Water Vapor $v_{\mathrm{vap,s}}$ = f(t)

Function Name:

vvaps_t_97_SI

Fortran Program:

REAL*8 FUNCTION VVAPS_T_97(T), REAL*8 T

Input Values:

t - Temperature t in °C

Result:

vvaps_t_97_SI - Specific volume of saturated water vapor in m^3/kg

Range of Validity:

Temperature t:	from 0°C to 350°C

Comments:

- Specific volume of saturated water vapor $v_{\mathrm{vap,s}} = v^{97}(p_{\mathrm{s}}, t)$ (Region 2) with $p_{\mathrm{s}} = p_{\mathrm{s}}^{97}(t)$

Result for Wrong Input Values:

vvaps_t_97_SI = -1000

References:

$v^{97}(p,t)$, $p_{\mathrm{s}}^{97}(t)$	IAPWS-IF97 [7], [8]

3.3 Functions for Steam and Ice for Temperatures $t \le 0°\mathrm{C}$

Specific Enthalpy of Saturated Ice $h_{\text{ice,sub}} = \text{f}(t)$

Function Name:

hicesub_t_06_SI

Fortran Program:

REAL*8 FUNCTION   HICESUB_T_06(T), REAL*8 T

Input Values:

t - Temperature *t* in °C

Result:

hicesub_t_06_SI - Specific enthalpy of saturated ice in kJ/kg

Range of Validity:

Temperature *t*: from -143.15°C to 0°C

Comments:

- Specific enthalpy of saturated ice $h_{\text{ice,sub}} = h^{06}(p_{\text{sub}}, t)$
  with $p_{\text{sub}} = p_{\text{sub}}^{08}(t)$

Result for Wrong Input Values:

hicesub_t_06_SI = -1000

References:

$h^{06}(p,t)$	IAPWS-06 [10]
$p_{\text{sub}}^{08}(t)$	IAPWS-08 [11]

Specific Enthalpy of Saturated Water Vapor $h_{\mathrm{vap,sub}} = \mathrm{f}(t)$

Function Name:

hvapsub_t_95_SI

Fortran Program:

REAL*8 FUNCTION HVAPSUB_T_95(T), REAL*8 T

Input Values:

t - Temperature *t* in °C

Result:

hvapsub_t_95_SI - Specific enthalpy of saturated water vapor in kJ/kg

Range of Validity:

Temperature *t*: from -143.15°C to 0°C

Comments:

- Specific enthalpy of saturated water vapor $h_{\mathrm{vap,sub}} = h^{95}(p_{\mathrm{sub}}, t)$
  with $p_{\mathrm{sub}} = p_{\mathrm{sub}}^{08}(t)$

Result for Wrong Input Values:

hvapsub_t_95_SI = -1000

References:

$h^{95}(p,t)$	IAPWS-95 [5], [6]
$p_{\mathrm{sub}}^{08}(t)$	IAPWS-08 [11]

Melting Pressure $p_{\mathrm{mel}} = \mathrm{f}(t)$

Function Name:

pmel_t_08_SI

Fortran Program:

REAL*8 FUNCTION   PMEL_T_08 (T), REAL*8 T

Input Values:

t - Temperature t in °C

Result:

pmel_t_08_SI - Melting pressure of ice in kPa

Range of Validity:

Temperature t:	from -21.985°C to 0°C

Result for Wrong Input Values:

pmel_t_08_SI = -1000

References:

$p_{\mathrm{mel}}^{08}(t)$	IAPWS-08 [11]

Sublimation Pressure $p_{\text{sub}} = f(t)$

Function Name:

psub_t_08_SI

Fortran Program:

```
REAL*8 FUNCTION   PSUB_T_08 (T), REAL*8 T
```

Input Values:

t - Temperature *t* in °C

Result:

psub_t_08_SI - Sublimation pressure of ice in kPa

Range of Validity:

Temperature *t*: from -143.15°C to 0°C

Result for Wrong Input Values:

psub_t_08_SI = -1000

References:

$p_{\text{sub}}^{08}(t)$ IAPWS-08 [11]

Specific Entropy of Saturated Ice $s_{\mathrm{ice,sub}}$ = f(t)

Function Name:

sicesub_t_06_SI

Fortran Program:

REAL*8 FUNCTION SICESUB_T_06(T), REAL*8 T

Input Values:

t - Temperature t in °C

Result:

sicesub_t_06_SI - Specific entropy of saturated ice in kJ/(kg K)

Range of Validity:

Temperature t: from -143.15°C to 0°C

Comments:

- Specific entropy of saturated ice $s_{\mathrm{ice,sub}} = s^{06}(p_{\mathrm{sub}}, t)$
  with $p_{\mathrm{sub}} = p_{\mathrm{sub}}^{08}(t)$

Result for Wrong Input Values:

sicesub_t_06_SI = -1000

References:

$s^{06}(p,t)$	IAPWS-06 [10]
$p_{\mathrm{sub}}^{08}(t)$	IAPWS-08 [11]

Specific Entropy of Saturated Water Vapor $s_{\mathrm{vap,sub}} = \mathrm{f}(t)$

Function Name:

svapsub_t_95_SI

Fortran Program:

```
REAL*8 FUNCTION   SVAPSUB_T_95(T), REAL*8 T
```

Input Values:

t - Temperature *t* in °C

Result:

svapsub_t_95_SI - Specific entropy of saturated water vapor in kJ/(kg K)

Range of Validity:

Temperature *t*: from -143.15°C to 0°C

Comments:

- Specific entropy of saturated water vapor $s_{\mathrm{vap,sub}} = s^{95}(p_{\mathrm{sub}}, t)$
  with $p_{\mathrm{sub}} = p_{\mathrm{sub}}^{08}(t)$

Result for Wrong Input Values:

svapsub_t_95_SI = -1000

References:

$s^{95}(p,t)$	IAPWS-95 [7], [8]
$p_{\mathrm{sub}}^{08}(t)$	IAPWS-08 [11]

Melting Temperature t_{mel} = f(p)

Function Name:

tmel_p_08_SI

Fortran Program:

REAL*8 FUNCTION   TMEL_P_08(P), REAL*8 P

Input Values:

p - Pressure *p* in kPa

Result:

tmel_p_08_SI - Melting temperature of ice in °C

Range of Validity:

Pressure *p*:	from $p_s(0°C) = 0.6112$ kPa to 10 000 kPa

Result for Wrong Input Values:

tmel_p_08_SI = -1000

References:

$t_{mel}^{08}(p)$	IAPWS-08 [11]

Sublimation Temperature t_{sub} = f(p)

Function Name:

tsub_p_08_SI

Fortran Program:

REAL*8 FUNCTION TSUB_P_08(P), REAL*8 P

Input Values:

p - Pressure p in kPa

Result:

tsub_p_08_SI - Sublimation temperature of ice in °C

Range of Validity:

Pressure p: from $p_{subl}(-143.15°C) = 1.2002 \times 10^{-11}$ kPa to $p_{subl}(0°C) = 0.6112$ kPa

Result for Wrong Input Values:

tsub_p_08_SI = -1000

References:

$t_{sub}^{08}(p)$ IAPWS-08 [11]

Specific Volume of Saturated Ice $v_{\mathrm{ice,sub}} = \mathrm{f}(t)$

Function Name:

vicesub_t_06_SI

Fortran Program:

REAL*8 FUNCTION VICESUB_T_06(T), REAL*8 T

Input Values:

t - Temperature *t* in °C

Result:

vicesub_t_06_SI - Specific volume of saturated ice in m^3/kg

Range of Validity:

Temperature *t*: from -143.15°C to 0°C

Comments:

- Specific volume of saturated ice $v_{\mathrm{ice,sub}} = v^{06}(p_{\mathrm{sub}}, t)$
  with $p_{\mathrm{sub}} = p_{\mathrm{sub}}^{08}(t)$

Result for Wrong Input Values:

vicesub_t_06_SI = -1000

References:

$v^{06}(p,t)$	IAPWS-06 [10]
$p_{\mathrm{sub}}^{08}(t)$	IAPWS-08 [11]

Specific Volume of Saturated Water Vapor $v_{\mathrm{vap,sub}} = \mathrm{f}(t)$

Function Name:

vvapsub_t_95_SI

Fortran Program:

`REAL*8 FUNCTION   VVAPSUB_T_95(T), REAL*8 T`

Input Values:

t - Temperature *t* in °C

Result:

vvapsub_t_95_SI - Specific volume of saturated water vapor in m^3/kg

Range of Validity:

Temperature *t*: from -143.15°C to 0°C

Comments:

- Specific volume of saturated water vapor $v_{\mathrm{vap,sub}} = v^{95}(p_{\mathrm{sub}}, t)$ with $p_{\mathrm{sub}} = p_{\mathrm{sub}}^{08}(t)$

Result for Wrong Input Values:

vvapsub_t_95_SI = -1000

References:

$v^{95}(p,t)$	IAPWS-95 [7], [8]
$p_{\mathrm{sub}}^{08}(t)$	IAPWS-08 [11]

4. Property Libraries for Calculating Heat Cycles, Boilers, Turbines and Refrigerators

Water and Steam

Library LibIF97

- Industrial Formulation IAPWS-IF97 (Revision 2007)
- Supplementary Standards IAPWS-IF97-S01, -S03rev, -S04, and -S05
- IAPWS Revised Advisory Note No. 3 on Thermodynamic Derivatives (2008)

Library LibIF97_META

- Industrial Formulation IAPWS-IF97 (Revision 2007) for metastable steam

Humid Combustion Gas Mixtures

Library LibHuGas

Model: Ideal mixture of the real fluids:

CO_2 - Span, Wagner H_2O - IAPWS-95
O_2 - Schmidt, Wagner N_2 - Span et al.
Ar - Tegeler et al.

and of the ideal gases:

SO_2, CO, Ne
(Scientific Formulation of Bücker et al.)

Consideration of:

- Dissociation from VDI 4670
- Poynting effect

Humid Air

Library LibHuAir

Model: Ideal mixture of the real fluids:

- Dry air from Lemmon et al.
- Steam, water and ice from IAPWS-IF97 and IAPWS-06

Consideration of:

- Condensation and freezing of steam
- Dissociation from VDI 4670
- Poynting effect from ASHRAE RP-1485

Extremely Fast Property Calculations

Spline-Based Table Look-up Method (SBTL)

Library LibSBTL_IF97
Library LibSBTL_95
Library LibSBTL_HuAir

For steam, water, humid air, carbon dioxide and other fluids and mixtures according IAPWS Guideline 2015 for Computational Fluid Dynamics (CFD), real-time and non-stationary simulations

Carbon Dioxide Including Dry Ice

Library LibCO2

Formulation of Span and Wagner (1996)

Seawater

Library LibSeaWa

IAPWS Industrial Formulation 2013

Ice

Library LibICE

Ice from IAPWS-06, Melting and sublimation pressures from IAPWS-08, Water from IAPWS-IF97, Steam from IAPWS-95 and -IF97

Ideal Gas Mixtures

Library LibIdGasMix

Model: Ideal mixture of the ideal gases:

Ar
Ne
N_2
O_2
CO
CO_2
Air
NO
H_2O
SO_2
H_2
H_2S
OH
He
F_2
NH_3
Methane
Ethane
Ethylene
Propylene
Propane
Iso-Butane
n-Butane
Benzene
Methanol

Consideration of:

- Dissociation from the VDI Guideline 4670

Library LibIDGAS

Model: Ideal gas mixture from VDI Guideline 4670

Consideration of:

- Dissociation from the VDI Guideline 4670

Humid Air

Library ASHRAE LibHuAirProp

Model: Virial equation from ASHRAE Report RP-1485 for real mixture of the real fluids:
- Dry air
- Steam

Consideration of:

- Enhancement of the partial saturation pressure of water vapor at elevated total pressures

www.ashrae.org/bookstore

Dry Air Including Liquid Air

Library LibRealAir

Formulation of Lemmon et al. (2000)

Refrigerants

Ammonia

Library LibNH3

Formulation of Tillner-Roth et al. (1993)

R134a

Library LibR134a

Formulation of Tillner-Roth and Baehr (1994)

Iso-Butane

Library LibButane_Iso

Formulation of Bücker and Wagner (2006)

n-Butane

Library LibButane_n

Formulation of Bücker and Wagner (2006)

Mixtures for Absorption Processes

Ammonia/Water Mixtures

Library LibAmWa

IAPWS Guideline 2001 of Tillner-Roth and Friend (1998)

Helmholtz energy equation for the mixing term

(also useable for calculating the Kalina Cycle)

Water/Lithium Bromide Mixtures

Library LibWaLi

Formulation of Kim and Infante Ferreira (2004)

Gibbs energy equation for the mixing term

Liquid Coolants

Liquid Secondary Refrigerants

Library LibSecRef

Liquid solutions of water with

$C_2H_6O_2$	Ethylene glycol
$C_3H_8O_2$	Propylene glycol
C_2H_5OH	Ethanol
CH_3OH	Methanol
$C_3H_8O_3$	Glycerol
K_2CO_3	Potassium carbonate
$CaCl_2$	Calcium chloride
$MgCl_2$	Magnesium chloride
NaCl	Sodium chloride
$C_2H_3KO_2$	Potassium acetate
$CHKO_2$	Potassium formate
LiCl	Lithium chloride
NH_3	Ammonia

Formulation of the International Institute of Refrigeration (IIR 2010)

Ethanol

Library LibC2H5OH

Formulation of
Schroeder et al. (2014)

Methanol

Library LibCH3OH

Formulation of
de Reuck and Craven (1993)

Propane

Library LibPropane

Formulation of
Lemmon et al. (2009)

Siloxanes as ORC Working Fluids

Octamethylcyclotetrasiloxane **$C_8H_{24}O_4Si_4$** **Library LibD4**

Decamethylcyclopentasiloxane **$C_{10}H_{30}O_5Si_5$** **Library LibD5**

Tetradecamethylhexasiloxane **$C_{14}H_{42}O_5Si_6$** **Library LibMD4M**

Hexamethyldisiloxane **$C_6H_{18}OSi_2$** **Library LibMM**

Formulation of Colonna et al. (2006)

Dodecamethylcyclohexasiloxane **$C_{12}H_{36}O_6Si_6$** **Library LibD6**

Decamethyltetrasiloxane **$C_{10}H_{30}O_3Si_4$** **Library LibMD2M**

Dodecamethylpentasiloxane **$C_{12}H_{36}O_4Si_5$** **Library LibMD3M**

Octamethyltrisiloxane **$C_8H_{24}O_2Si_3$** **Library LibMDM**

Formulation of Colonna et al. (2008)

Nitrogen and Oxygen

Libraries
LibN2 and LibO2

Formulations of Span et al. (2000)
and Schmidt and Wagner (1985)

Hydrogen

Library LibH2

Formulation of
Leachman et al. (2009)

Helium

Library LibHe

Formulation of
Arp et al. (1998)

Hydrocarbons

Decane **$C_{10}H_{22}$** **Library LibC10H22**

Isopentane **C_5H_{12}** **Library LibC5H12_Iso**

Neopentane **C_5H_{12}** **Library LibC5H12_Neo**

Isohexane **C_6H_{14}** **Library LibC6H14**

Toluene **C_7H_8** **Library LibC7H8**

Formulation of Lemmon and Span (2006)

Further Fluids

Carbon monoxide **CO** **Library LibCO**

Carbonyl sulfide **COS** **Library LibCOS**

Hydrogen sulfide **H_2S** **Library LibH2S**

Nitrous oxide **N_2O** **Library LibN2O**

Sulfur dioxide **SO_2** **Library LibSO2**

Acetone **C_3H_6O** **Library LibC3H6O**

Formulation of Lemmon and Span (2006)

KCE
ThermoFluidProperties

For more information please contact:

KCE-ThermoFluidProperties UG
Prof. Dr. Hans-Joachim Kretzschmar
Haager Weg 6
92224 Amberg, Germany

Internet: www.thermofluidprop.com
Email: info@thermofluidprop.com
Phone: +49-9621-1762047
Mobile: +49-172-7914607
Fax: +49-3222-1095810

The following thermodynamic and transport properties can be calculated[a]:

Thermodynamic Properties

- Vapor pressure p_s
- Saturation temperature T_s
- Density ρ
- Specific volume v
- Enthalpy h
- Internal energy u
- Entropy s
- Exergy e
- Isobaric heat capacity c_p
- Isochoric heat capacity c_v
- Isentropic exponent κ
- Speed of sound w
- Surface tension σ

Transport Properties

- Dynamic viscosity η
- Kinematic viscosity ν
- Thermal conductivity λ
- Prandtl number Pr
- Thermal diffusivity a

Backward Functions

- T, v, s (p,h)
- T, v, h (p,s)
- p, T, v (h,s)
- p, T (v,h)
- p, T (v,u)

Thermodynamic Derivatives

- Partial derivatives used in process modeling can be calculated.

---

[a] Not all of these property functions are available in all property libraries.

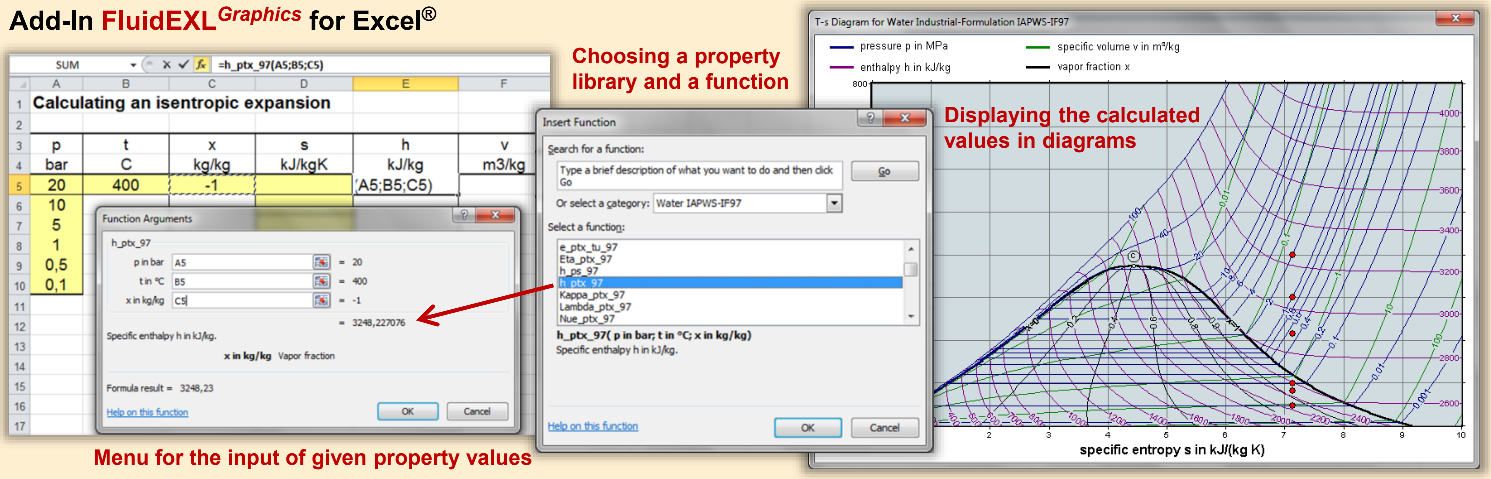
www.thermofluidprop.com

KCE-ThermoFluidProperties
www.thermofluidprop.com

Hochschule Zittau/Görlitz
UNIVERSITY OF APPLIED SCIENCES

Property Software for Calculating Heat Cycles, Boilers, Turbines and Refrigerators

Add-In FluidEXL^{Graphics} for Excel®



SUM =h_ptx_97(A5;B5;C5)

Calculating an isentropic expansion

p	t	x	s	h	v
bar	C	kg/kg	kJ/kgK	kJ/kg	m3/kg
20	400	-1		(A5;B5;C5)	
10					
5					
1					
0,5					
0,1					

Function Arguments

h_ptx_97
p in bar A5 = 20
t in °C B5 = 400
x in kg/kg C5 = -1
= 3248,227076
Specific enthalpy h in kJ/kg.
x in kg/kg Vapor fraction
Formula result = 3248,23

Insert Function

Search for a function: Type a brief description of what you want to do and then click Go
Or select a category: Water IAPWS-IF97
h_ptx_97(p in bar; t in °C; x in kg/kg)
Specific enthalpy h in kJ/kg.

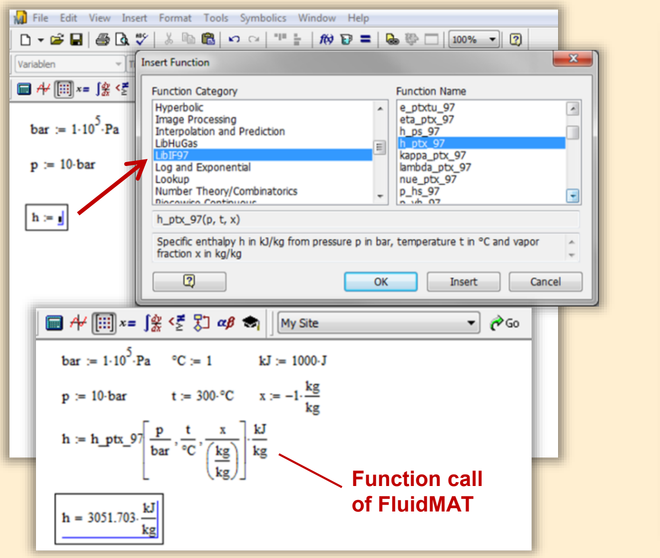
Choosing a property library and a function

Displaying the calculated values in diagrams

Menu for the input of given property values

Add-On FluidMAT for Mathcad®
Add-On FluidPRIME for Mathcad Prime®

The property libraries can be used in Mathcad® and Mathcad Prime®.



Insert Function

Function Category: LibIF97
Function Name: h_ptx_97

h_ptx_97(p, t, x)

Specific enthalpy h in kJ/kg from pressure p in bar, temperature t in °C and vapor fraction x in kg/kg

$bar := 1 \cdot 10^5 \cdot Pa$ $\quad$ $°C := 1$ $\quad$ $kJ := 1000 \cdot J$

$p := 10 \cdot bar$ $\quad$ $t := 300 \cdot °C$ $\quad$ $x := -1 \cdot \frac{kg}{kg}$

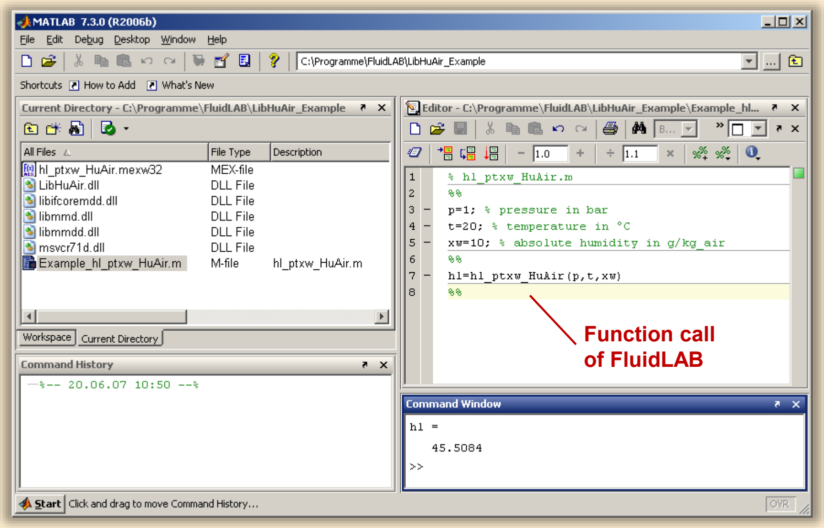
$$h := h\_ptx\_97\left(\frac{p}{bar}, \frac{t}{°C}, \frac{x}{\left(\frac{kg}{kg}\right)}\right) \cdot \frac{kJ}{kg}$$

$$h = 3051.703 \cdot \frac{kJ}{kg}$$

Function call of FluidMAT

Add-On FluidLAB for MATLAB® and SIMULINK®

Using the Add-In FluidLAB the property functions can be called in MATLAB® and SIMULINK®.



MATLAB 7.3.0 (R2006b)

C:\Programme\FluidLAB\LibHuAir_Example

All Files	File Type	Description
hl_ptxw_HuAir.mexw32	MEX-file	
LibHuAir.dll	DLL File	
libifcoremdd.dll	DLL File	
libmmd.dll	DLL File	
libmmdd.dll	DLL File	
msvcr71d.dll	DLL File	
Example_hl_ptxw_HuAir.m	M-file	hl_ptxw_HuAir.m

Command History

```
%-- 20.06.07 10:50 --%
```

Editor - C:\Programme\FluidLAB\LibHuAir_Example\Example_hl...

```
% hl_ptxw_HuAir.m
%%
p=1; % pressure in bar
t=20; % temperature in °C
xw=10; % absolute humidity in g/kg_air
%%
hl=hl_ptxw_HuAir(p,t,xw)
%%
```

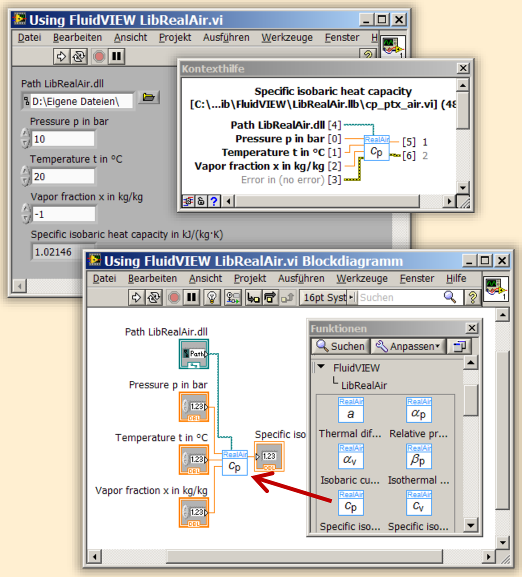
Command Window

```
hl =

   45.5084

>>
```

Function call of FluidLAB

Add-On FluidVIEW for LabVIEW™

The property functions can be calculated in LabVIEW™.



Using FluidVIEW LibRealAir.vi

Path LibRealAir.dll D:\Eigene Dateien\
Pressure p in bar 10
Temperature t in °C 20
Vapor fraction x in kg/kg -1
Specific isobaric heat capacity in kJ/(kg·K) 1.02146

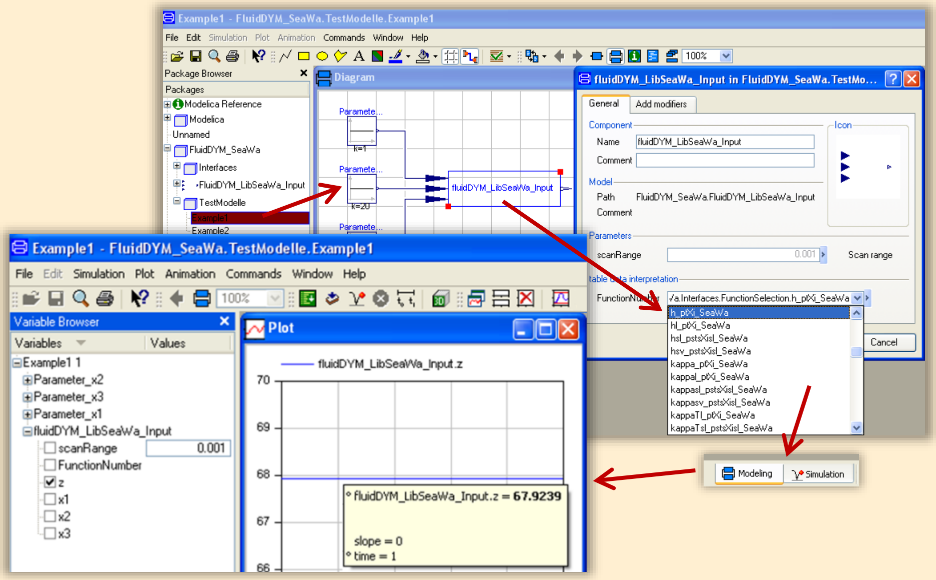
Kontexthilfe

Specific isobaric heat capacity
[C:\...ib\FluidVIEW\LibRealAir.llb\cp_ptx_air.vi] (48
Path LibRealAir.dll [4]
Pressure p in bar [0]
Temperature t in °C [1]
Vapor fraction x in kg/kg [2]
Error in (no error) [3]

Using FluidVIEW LibRealAir.vi Blockdiagramm

Add-On FluidDYM for DYMOLA® (Modelica) and SimulationX®

The property functions can be called in DYMOLA® and SimulationX®.



Example1 - FluidDYM_SeaWa.TestModelle.Example1

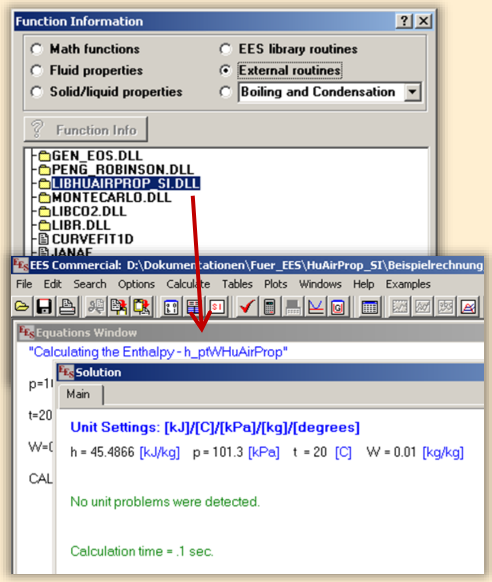
fluidDYM_LibSeaWa_Input in FluidDYM_SeaWa.TestMo...

Component
Name fluidDYM_LibSeaWa_Input
Model
Path FluidDYM_SeaWa.FluidDYM_LibSeaWa_Input
Parameters
scanRange 0.001 Scan range
FunctionNumber Modelica.Interfaces.FunctionSelection.h_pXi_SeaWa

Variables	Values
Example1 1	
Parameter_x2	
Parameter_x3	
Parameter_x1	
fluidDYM_LibSeaWa_Input	
scanRange	0.001
FunctionNumber	
z	
x1	
x2	
x3	

fluidDYM_LibSeaWa_Input.z = 67.9239
slope = 0
time = 1

Add-On FluidEES for Engineering Equation Solver®



"Calculating the Enthalpy - h_ptWHuAirProp"

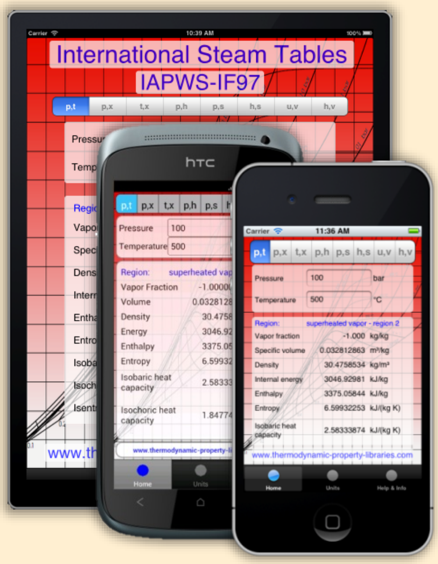
Unit Settings: [kJ]/[C]/[kPa]/[kg]/[degrees]

h = 45.4866 [kJ/kg] p = 101.3 [kPa] t = 20 [C] W = 0.01 [kg/kg]

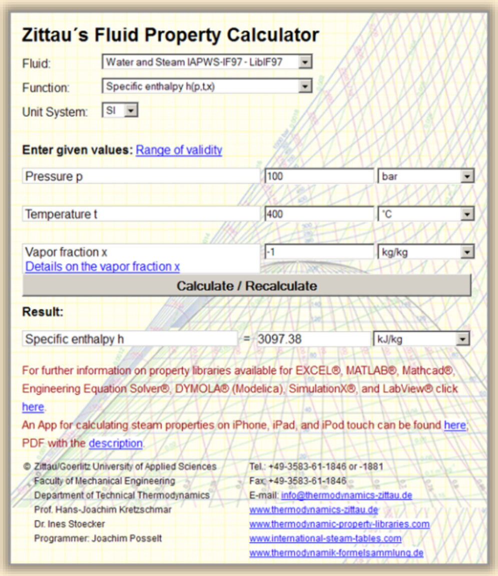
No unit problems were detected.

Calculation time = .1 sec.

App International Steam Tables for iPhone, iPad, iPod touch, Android Smartphones and Tablets



Online Property Calculator at www.thermofluidprop.com



Zittau´s Fluid Property Calculator

Fluid: Water and Steam IAPWS-IF97 - LibIF97
Function: Specific enthalpy h(p,t,x)
Unit System: SI

Enter given values: Range of validity

Pressure p 100 bar
Temperature t 400 °C
Vapor fraction x -1 kg/kg
Details on the vapor fraction x

Result:

Specific enthalpy h = 3097.38 kJ/kg

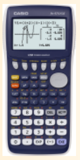
For further information on property libraries available for EXCEL®, MATLAB®, Mathcad®, Engineering Equation Solver®, DYMOLA® (Modelica), SimulationX®, and LabView® click here.
An App for calculating steam properties on iPhone, iPad, and iPod touch can be found here; PDF with the description.

© Zittau/Goerlitz University of Applied Sciences
Faculty of Mechanical Engineering
Department of Technical Thermodynamics
Prof. Hans-Joachim Kretzschmar
Dr. Ines Stoecker
Programmer: Joachim Posselt

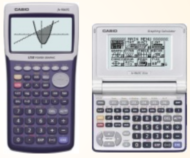
Tel.: +49-3583-61-1846 or -1881
Fax: +49-3583-61-1846
E-mail: info@thermodynamics-zittau.de
www.thermodynamics-zittau.de
www.thermodynamic-property-libraries.com
www.international-steam-tables.com
www.thermodynamik-formelsammlung.de

Property Software for Pocket Calculators

FluidCasio


fx 9750 G II


CFX 9850
fx-GG20


CFX 9860 G
Graph 85


ALGEBRA
FX 2.0

FluidHP


HP 48


HP 49

FluidTI


TI Nspire CX CAS
TI Nspire CAS


TI 83
TI 84
TI 89


TI Voyage 200


TI 92

For more information please contact:

KCE
ThermoFluidProperties

KCE-ThermoFluidProperties UG
Prof. Dr. Hans-Joachim Kretzschmar
Haager Weg 6
92224 Amberg, Germany

Internet: www.thermofluidprop.com
Email: info@thermofluidprop.com
Phone: +49-9621-1762047
Mobile: +49-172-7914607
Fax: +49-3222-1095810

The following thermodynamic and transport properties[a] can be calculated in Excel®, MATLAB®, Mathcad®, Engineering Equation Solver® (EES), DYMOLA® (Modelica), SimulationX® and LabVIEW™:

Thermodynamic Properties

- Vapor pressure p_s
- Saturation temperature T_s
- Density ρ
- Specific volume v
- Enthalpy h
- Internal energy u
- Entropy s
- Exergy e
- Isobaric heat capacity c_p
- Isochoric heat capacity c_v
- Isentropic exponent κ
- Speed of sound w
- Surface tension σ

Transport Properties

- Dynamic viscosity η
- Kinematic viscosity ν
- Thermal conductivity λ
- Prandtl number Pr
- Thermal diffusivity a

Backward Functions

- T, v, s (p,h)
- T, v, h (p,s)
- p, T, v (h,s)
- p, T (v,h)
- p, T (v,u)

Thermodynamic Derivatives

- Partial derivatives used in process modeling can be calculated.

---

[a] Not all of these property functions are available in all property libraries.

www.thermofluidprop.com

5 References

[1] Herrmann, S.; Kretzschmar, H.-J.; Gatley, D.P.: Thermodynamic Properties of Real Moist Air, Dry Air, Steam, Water, and Ice. HVAC&R Research 5, 961-986 (2009).

[2] Herrmann, S.; Kretzschmar, H.-J.; Gatley, D.P.: Thermodynamic Properties of Real Moist Air, Dry Air, Steam, Water, and Ice. ASHRAE RP-1485, American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc., Atlanta, GA (2009).

[3] Herrmann, S.; Kretzschmar, H.-J.; Aute, V.C.; Gatley, D.P.; Vogel, E.: Transport Properties of Real Moist Air, Dry Air, Steam, and Water. ASHRAE RP-1767, American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc., Atlanta, GA (2018).

[4] Herrmann, S.; Kretzschmar, H.-J.; Aute, V.C.; Gatley, D.P.; Vogel, E.: Transport Properties of Real Moist Air, Dry Air, Steam, and Water. Science and Technology for the Built Environment 27, 393-401 (2021).

[5] IAPWS. R6-95, Revised Release on the IAPWS Formulation 1995 for the Thermodynamic Properties of Ordinary Water Substance for General and Scientific Use. (2009), available from www.iapws.org.

[6] Wagner, W.; Pruß, A.: The IAPWS Formulation 1995 for the Thermodynamic Properties of Ordinary Water Substance for General and Scientific Use. J. Phys. Chem. Ref. Data 31, 387-535 (2002).

[7] IAPWS. R7-97, Revised Release on the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam IAPWS-IF97. (2007), available from www.iapws.org.

[8] Wagner, W.; Kretzschmar, H.-J.: International Steam Tables. Springer, Berlin (2008).

[9] Parry, W.T.; Bellows, J.C.; Gallagher, J.S.; Harvey, A.H.: ASME International Steam Tables for Industrial Use. ASME Press, New York (2000).

[10] IAPWS. R10-06, Revised Release on the Equation of State 2006 for H_2O Ice Ih. (2009); available from www.iapws.org.

[11] IAPWS. R14-08, Revised Release on the Pressure along the Melting and Sublimation Curves of Ordinary Water Substance. (2008); available from www.iapws.org.

[12] Nelson, H.F.; Sauer, H.J.: Formulation of High-Temperature Properties for Moist Air. HVAC&R Research 8, 311-334 (2002).

[13] Gatley, D.P.: Understanding Psychrometrics, 2nd ed., ASHRAE, Atlanta (2005).

[14] Lemmon, E.W.; Jacobsen, R.T.; Penoncello, S.G.; Friend, D.G.: Thermodynamic Properties of Air and Mixture of Nitrogen, Argon, and Oxygen from 60 to 2000 K at Pressures to 2000 MPa. J. Phys. Chem. Ref. Data 29, 331-385 (2000).

[15] Harvey, A.H.; Huang, P.H.: First-Principles Calculation of the Air-Water Second Virial Coefficient. Int. J. Thermophys. 28, 556-565 (2007).

[16] IAPWS. G7-04, Guideline on the Henry's Constant and Vapor-Liquid Distribution Constant for Gases in H_2O and D_2O at High Temperatures. (2004), available from www.iapws.org.

[17] Gatley, D.P.; Herrmann, S.; Kretzschmar, H.-J.: A Twenty-First Century Molar Mass for Dry Air. HVAC&R Research 14, 655-662 (2008).

[18] Lemmon, E.W.; Jacobsen, R.T.: Viscosity and Thermal Conductivity Equations for Nitrogen, Oxygen, Argon, and Air. Int. J. Thermophys. 25, 21-69 (2004).

[19] IAPWS. R12-08, Release on the IAPWS Formulation 2008 for the Viscosity of Ordinary Water Substance. (2008), available from www.iapws.org.

[20] IAPWS. R15-11, Revised Release on the IAPS Formulation 1985 for the Thermal Conductivity of Ordinary Water Substance. (2008), available from www.iapws.org.

[21] Hyland, R.W.; Wexler, A.: Formulations for the Thermodynamic Properties of Dry Air from 173.15 K to 473.15 K, and of Saturated Moist Air from 173.15 K to 372.15 K, at Pressures to 5 MPa. ASHRAE Trans. 89, 520-535 (1983).

[22] Mohr, P.J.; Taylor, P.N.: CODATA Recommended Values of the Fundamental Physical Constants: 2002. Rev. Mod. Phys. 77, 1-107 (2005).

[23] ASHRAE. 2009 Handbook of Fundamentals. Chapter 1 - Psychrometrics. American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc., Atlanta, GA (2009).

[24] Feistel, R.; Lovell-Smith, J.W.; Hellmuth, O.: Virial Approximation of the TEOS-10 Equation for the Fugacity of Water in Humid Air. Int. J. Thermophys. 36, 44-68 (2015).

6 Satisfied Customers

Period from 2018 to 2022

The following companies and institutions use the property libraries:

- FluidEXL$^{\textit{Graphics}}$ for Excel® incl. VBA
- FluidLAB for MATLAB® and Simulink
- FluidMAT for Mathcad®
- FluidPRIME for Mathcad Prime®
- FluidEES for Engineering Equation Solver® EES
- FluidDYM for Dymola® (Modelica) and SimulationX®
- FluidVIEW for LabVIEW™
- FluidPYT for Python
- FluidJAVA for Java
- DLLs for Windows Applications
- Shared Objects for Linux
- Shared Objects for macOS.

2022

ASTG, Graz, Austria	12/2022
Wandschneider + Gutjahr, Hamburg	
RWE Supply & Trading, Essen	11/2022
Stadtwerke Rosenheim	
CEA, Saclay, France	10/2022
RWE Supply & Trading, Essen	
SEEC Saudi Energy Efficiency Center, Riyadh, Saudi Arabia	
MAN, Copenhagen, Denmark	
Hermeler & Partner Consulting Engineers, Sassenberg	09/2022
Envi Con, Nürnberg	
Drill Cool Systems, Bakersfield CA, USA	
RWE Supply & Trading, Essen	
Maerz Ofenbau, Zürich, Switzerland	
Saale Energie, Schkopau	
ERGO, Dresden	
Mainova, Frankfurt/Main	
Bundeswehr, Koblenz	08/2022
RWE Supply & Trading, Essen	
Grenzebach Corporation, Newnan GE, USA	
AGRANA, Gmuend, Austria	07/2022
MIBRAG, Zeitz	
Hochschule Niederrhein, Krefeld	
ULT, Löbau	06/2022
LEAG, Cottbus	

VPC Group, Vetschau	
Wärme, Hamburg	
ILK, Dresden	
Stricker IB, Küssnacht a. Rigi, Switzerland	
LEAG, Cottbus	05/2022
RWE Supply & Trading, Essen	
IGT Tomalla, Kreuztal	
B+T Engineering, Dübendorf, Switzerland	
Stricker IB, Küssnacht a. Rigi, Switzerland	
Vogelsang & Benning, Bochum	04/2022
Frischli, Rehburg-Loccum	
BPS Consulting, Sprenge	03/2022
HS Hannover, Maschinenbau & BioVT	
M+M Turbinentechnik, Bad Salzuflen	
Uni. Strathclyde, Glasgow, UK	02/2022
Delta Energy Group, Jiaozhou City, Qingdao, China	
Wetzel IB, Guben	
Wijbenga, PC Geldermalsen, The Netherlands	
Voith Paper, Heidenheim	
HS Zittau/Görlitz, Maschinenwesen	01/2022
Thermische Abfallbehandlung, Lauta	
Webb Institute, Glen Cove NY, USA	
TU Berlin, Umweltverfahrenstechnik	
SachsenEnergie, Dresden	
Doosan, Chang-won-si, Gyeongsangnam-do, South Korea	
KW3, LH Veenendaal, The Netherlands	
Université du Luxembourg, Esch-sur-Alzette	
Enseleit IB, Mansfeld	
Caliqua/Equans, Zürich, Switzerland	
Rudnick & Enners, Alpenrod	

2021

Wenisch IB, Vetschau	12/2021
PPCHEM, Hinwil, Switzerland	
KW3, The Netherlands	
BASF Ludwigshafen	
Air-Consult, Jena	
Sjerp & Jongeneel, RB Zoetermeer, The Netherlands	11/2021
Maerz Ofenbau, Zürich, Switzerland	
RWE Supply & Trading, Essen	
Hahn IB, Dresden	10/2021
Therm, South Africa	
RWE Supply & Trading, Essen	
TH Nürnberg, Verfahrenstechnik	09/2021
RWE Supply & Trading, Essen	
Enseleit IB, Mansfeld	

SachsenEnergie, Dresden	
BSH Hausgeräte, Berlin	
Norsk Energi, Oslo, Norway	08/2021
AKM Industrieanlagen, Haltern	
Drill Cool Systems, Bakersfield CA, USA	
Siemens Energy Global, Erlangen	07/2021
Wulff & Umag, Husum	
Planungsbüro Waidhas, Chemnitz	
Burkhardt Energie Technik, Mühlhausen	
Lücke IB, Paderborn	06/2021
TU Dresden, Energieverfahrenstechnik	
Wärme, Hamburg	
AL-KO Therm, Kötz	
PCK Raffinerie, Schwedt	
Vogelsang & Benning, Bochum	05/2021
MTU, München	
VPC Group, Vetschau	
AVG, Köln	04/2021
TH Ulm, Institut für Fahrzeugtechnik	
Marty IB, Oberwil, Switzerland	
HypTec, Lebring, Austria	
Lopez IB, Getxo, Bizkaia, Spain	03/2021
GM Remediation Systems, Leoben, Austria	
Jager Kältetechnik, Osnabrück	
T&M Automation, GR Leidschendam, The Netherlands	
RWE Supply & Trading, Essen	
Stadtwerke Leipzig	
Beuth Hochschule für Technik, Berlin	
Beleth IB, Woeth	02/2021
ZTL, Thal, Austria	
ETABO Bochum	
RWE Supply & Trading, Essen	
Onyx Germany, Berlin	
TU Dresden, Kältetechnik	
GOHL-KTK, Durmersheim	
Therm Development, South Africa	
thermofin, Heinsdorfergrund	
RWE Supply & Trading, Essen	01/2021
STEAG, Essen	
ETA Energieberatung, Pfaffenhofen	
Enex Power, Kirchseeon	

2020

Drill Cool, Bakersfield CA, USA	12/2020
Manders, The Netherlands	
RWE Supply & Tranding, Essen	

NEOWAT Lodz, Poland	
University of Duisburg-Essen, Duisburg	11/2020
Stellenbosch University, South Africa	
University De France-COMTe, France	
RWE, Essen	
STEAG, Herne	
Isenmann Ingenierbüro	
University of Stuttgart, ITLR, Stuttgart	
Norsk Energi, Oslo, Norway	
TGM Kanis, Nürnberg	
Stadtwerke Neuburg	10/2020
Smurfit Kappa, Roermond, The Netherlands	
RWE, Essen	
Hochschule Zittau/Görlitz, Wirtschaftsingenieurwesen	
Stadtwerke, Neuburg	
ILK, Dresden	
ATESTEO, Alsdorf	
Hochschule Zittau/Görlitz, Maschinenwesen	
TH Nürnberg, Verfahrenstechnik	
Drill Cool, Bakersfield CA,USA	09/2020
RWE, Essen	
2Meyers Ingenieurbüro, Nürnberg	
FELUWA, Mürlenbach	
Stadtwerke Neuburg	
Caverion, Wien, Austria	
GMVA Niederrhein, Oberhausen	
INWAT Lodz, Poland	
Troche Ingenieurbüro, Hayingen	08/2020
CEA Saclay, France	
VPC, Vetschau	07/2020
FSK System-Kälte-Klima, Dortmund	
Exergie Etudes, Sarl, Switzerland	
AWG Wuppertal	
STEAG Energy Services, Zwingenberg	
Hochschule Braunschweig	06/2020
DBI, Leipzig	
GOHL-KTK, Dumersheim	
TU Dresden, Energieverfahrenstechnik	
BASF SE, ESI/EE, Ludwigshafen	
Wärme Hamburg	
Ruchti Ingenieurbüro, Uster, Switzerland	
IWB, Basel, Switzerland	
Midiplan, Bietingen-Bissingen	05/2020
Knieschke, Ingenieurbüro	
RWE, Essen	
Leser, Hamburg	

AGRANA, Gmünd, Austria	
EWT Wassertechnik, Celle	
Hochschule Darmstadt	04/2020
MTU München CCP	
HAW Hamburg	03/2020
Hanon, Novi Jicin, Czech Republic	
TU Dresden, Kältetechnik	
MAN, Copenhagen, Denmark	
EnerTech, Radebeul	02/2020
LEAG, Cottbus	
B+B Enginering Magdeburg	
Hochschule Offenburg	
WIB, Dennheritz	01/2020
Universität Duisburg-Essen, Strömungsmaschinen	
Kältetechnik Dresen-Bremen	
TH Ingolstadt	
Vattenfall AB, Jokkmokk, Sweden	
Fraunhofer UMSICHT	

2019

PEU Leipzig, Rötha	12/2019
MB-Holding, Vestenbergsgreuth	
RWE, Essen	
Georg-Büchner-Hochschule, Darmstadt	11/2019
EEB ENERKO, Aldenhoven	
Robert Benoufa Energietechnik, Wiesloch	
Kehrein & Kubanek Klimatechnik, Moers	10/2019
Hanon Systems Autopal Services, Hluk, Czech Republic	
CEA Saclay, Gif Sur Yvette cedex, France	
Saudi Energy Efficiency Center SEEC, Riyadh, Saudi Arabia	
VPC, Vetschau	09/2019
jGanser PM + Engineering, Forchheim	
Endress+Hauser Flowtec AG, Reinach, Switzerland	
Ruchti IB, Uster, Switzerland	
ZWILAG Zwischenlager Würenlingen, Switzerland	08/2019
Hochschule Zittau/Görlitz, Faculty Maschinenwesen	
Stadtwerke Neubrandenburg	
Physikalisch Technische Bundesanstalt PTB, Braunschweig	
GMVA Oberhausen	07/2019
Endress+Hauser Flowtec AG, Reinach, Switzerland	
WARNICA, Waterloo, Canada	
MIBRAG, Zeitz	06/2019
Pöyry, Zürich, Switzerland	
RWTH Aachen, Institut für Strahlantriebe und Turbomaschinen	
Midiplan, Bietigheim-Bissingen	
GKS Schweinfurt	

HS Zittau/Görlitz, Wirtschaftswissenschaften und Wirtschaftsingenieurwesen	
ILK Dresden	
HZDR Helmholtz Zentrum Dresden-Rossendorf	
TH Köln, Technische Gebäudeausrüstung	05/2019
IB Knittel, Braunschweig	
Norsk Energi, Oslo, Norway	
STEAG, Essen	
Stora Enso, Eilenburg	
IB Lücke, Paderborn	
Haarslev, Sonderso, Denmark	
MAN Augsburg	
Wieland Werke, Ulm	04/2019
Fels-Werke, Elbingerode	
Univ. Luxembourg, Luxembourg	
BTU Cottbus, Power Engineering	03/2009
Eins-Energie Sachsen, Schwarzenberg	
TU Dresden, Kälte- und Kryotechnik	
ITER, St. Paul Lez Durance Cedex, France	
Fraunhofer UMSICHT, Oberhausen	
Comparex Leipzig for Spedition Thiele HEMMERSBACH	
Rückert NaturGas, Lauf/Pegnitz	
BASF, Basel, Switzerland	02/2019
Stadtwerke Leipzig	
Maerz Ofenbau Zürich, Switzerland	
Hanon Systems Germany, Kerpen	
Thermofin, Heinsdorfergrund	01/2019
BSH Berlin	

2018

Jaguar Energy, Guatemala	12/2018
WEBASTO, Gilching	
Smurfit Kappa, Oosterhout, Netherlands	
Univ. BW München	
RAIV, Liberec for VALEO, Prague, Czech Republic	11/2018
VPC Group Vetschau	
SEITZ, Wetzikon, Switzerland	
MVV, Mannheim	10/2018
IB Troche	
KANIS Turbinen, Nürnberg	
TH Ingolstadt, Institut für neue Energiesysteme	
IB Kristl & Seibt, Graz, Austria	09/2018
INEOS, Köln	
IB Lücke, Paderborn	
Südzucker, Ochsenfurt	08/2018
K&K Turbinenservice, Bielefeld	07/2018
OTH Regensburg, Elektrotechnik	

Comparex Leipzig for LEAG, Berlin	06/2018
Münstermann, Telgte	05/2018
TH Nürnberg, Verfahrenstechnik	
Universität Madrid, Madrid, Spanien	
HS Zittau/Görlitz, Wirtschaftsingenieurwesen	
HS Niederrhein, Krefeld	
Wilhelm-Büchner HS, Pfungstadt	03/2018
GRS, Köln	
WIB, Dennheritz	
RONAL AG, Härklingen, Schweiz	02/2018
Ingenieurbüro Leipert, Riegelsberg	
AIXPROCESS, Aachen	
KRONES, Neutraubling	
Doosan Lentjes, Ratingen	01/2018