Hochschule Zittau/Görlitz
UNIVERSITY OF APPLIED SCIENCES

Faculty of
MECHANICAL ENGINEERING

Department of
TECHNICAL THERMODYNAMICS

Property Library for Gas Mixtures in the Energy-Technological Process Modelling

FluidVIEW
with LibIdGasMix
for LabVIEW™

Prof. Hans-Joachim Kretzschmar
Dr. Ines Stoecker
Matthias Kunick
K. Knobloch
B. Nowak

Property Library for Gas Mixtures in the Energy-Technological Process Modeling

LibIdGasMix

FluidVIEW for LabVIEW™

Contents

Faculty of Mechanical Engineering
Department of Technical Thermodynamics
Prof. Hans-Joachim Kretzschmar
Dr. Ines Stoecker
Phone: +49-3583-61-1846 or -1881
Fax: +49-3583-61-1846
E-mail: hj.kretzschmar@hs-zigr.de
Internet: www.thermodynamics-zittau.de

0. Package Contents

0.1 Zip files for 32-bit Windows®

In order to install FluidVIEW on a computer running a 32-bit version of Windows® the zip file **CD_FluidVIEW_LibIdGasMix.zip** is delivered. The directory structure of the archive is corresponding to the default directory of LabVIEW™. All contained files, their paths and the structure of the archive are shown in the screenshot of the WinRAR file archiver and compression tool illustrated in Figure 0.1.

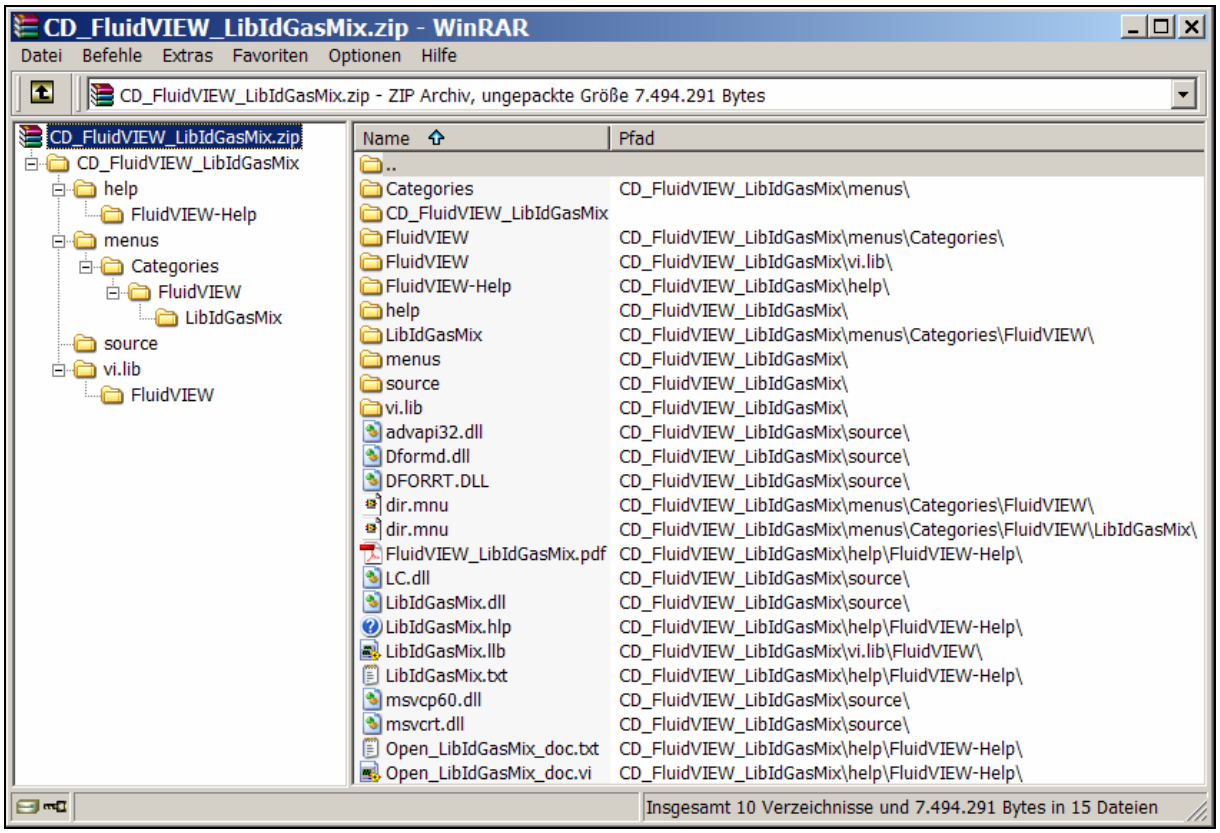


Figure 0.1 Screenshot of WinRAR showing the **CD_FluidVIEW_LibIdGasMix.zip** archive.

The effects of the fifteen files, which are stored in the different directories of the zip archive, are shown in the Tables 0.1, 0.2, 0.3 and 0.4.

Table 0.1 Effects of the files located in the archive directory **CD_FluidVIEW_LibIdGasMix\vi.lib \FluidVIEW\LibIdGasMix**

Filename	Effects
LibIdGasMix.llb	LabVIEW™ library file, containing every function of the LibIdGasMix property library in the form of subprograms (SubVIs)

Table 0.2 Effects of the files located in the archive directory **CD_FluidVIEW_LibIdGasMix\menus \Categories\FluidVIEW**

Filename	Effects
dir.mnu	The palette view of LabVIEW™ is based on the palette files (*.mnu). They include the palette data (e. g. the display name, the palette icon, the palette description, the help information, the synchronize information and the items)

Table 0.3 Effects of the files located in the archive directory **CD_FluidVIEW_LibIdGasMix\source**

Filename	Effects
LibIdGasMix.dll	Dynamic-link library containing the algorithms for the calculation of the property functions of 25 ideal gases and their ideal mixtures
advapi32.dll	Runtime library
Dformd.dll	Runtime library for the Fortran DLL
Dforrt.dll	Runtime library for the Fortran DLL
LC.dll	Auxiliary library
msvcp60.dll	Runtime library
msvcrt.dll	Runtime library

Table 0.4 Effects of the files located in the archive directory **CD_FluidVIEW_LibIdGasMix\help \FluidVIEW-help**

Filename	Effects
FluidVIEW_LibIdGasMix.pdf	User's guide of the property library LibIdGasMix for the LabVIEW™ Add-On FluidVIEW
LibIdGasMix.hlp	Help file with descriptions for each function
OpenLibIdGasMix_doc.vi	LabVIEW™ instrument to open the user's guide via the help menu
LibIdGasMix.txt	Text file to change the name of the menu item of the help file
OpenLibIdGasMix_doc.txt	Text file to change the name of the menu item of the file OpenLibIdGasMix_doc.vi

0.2 Zip files for 64-bit Windows®

In order to install FluidVIEW on a computer running a 64-bit version of Windows® the zip file **CD_FluidVIEW_LibIdGasMix_x64.zip** is delivered. The directory structure of the archive is corresponding to the default directory of LabVIEW™. All contained files, their paths and the structure of the archive are shown in the screenshot of the WinRAR file archiver and compression tool illustrated in Figure 0.2.

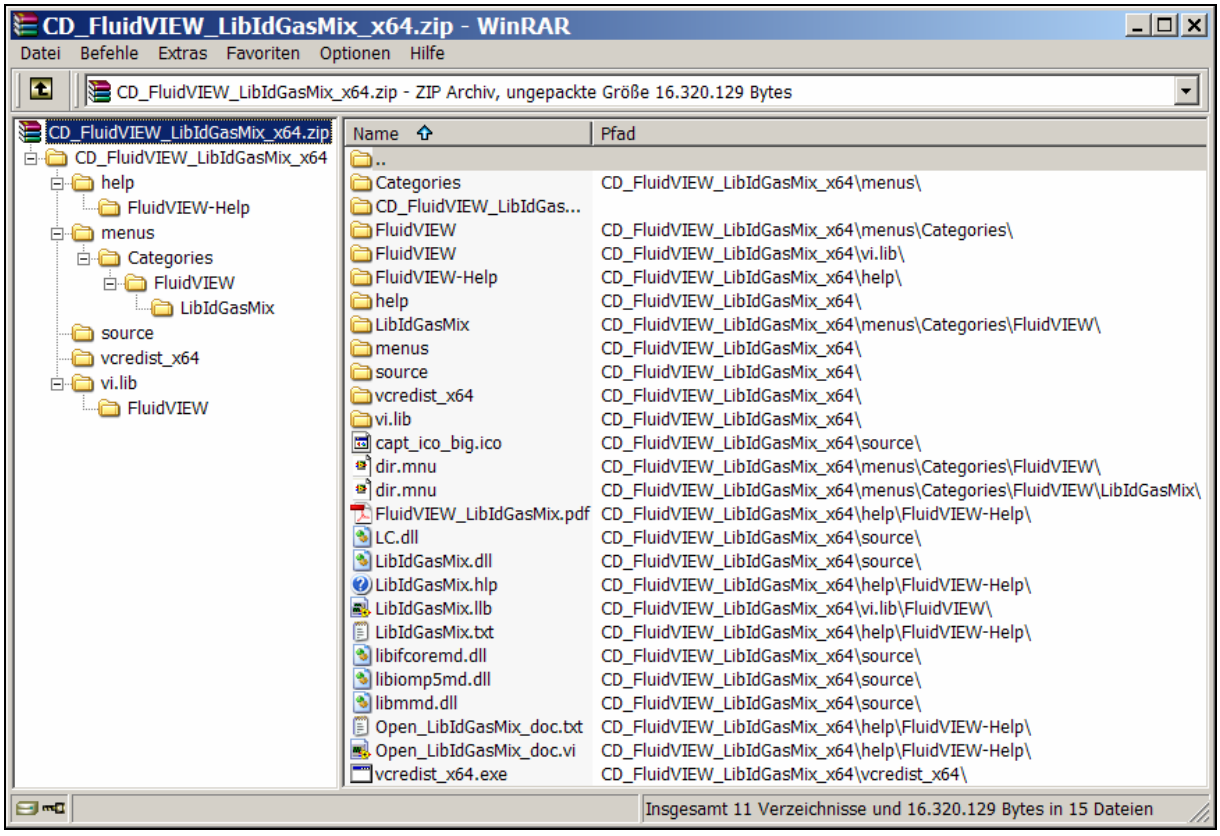


Figure 0.2 Screenshot of WinRAR showing the **CD_FluidVIEW_LibIdGasMix_x64.zip** archive.

The effects of the fifteen files, which are stored in the different directories of the zip archive, are shown in the Tables 0.5, 0.6, 0.7 and 0.8.

Table 0.5 Effects of the files located in the archive directory **CD_FluidVIEW_LibIdGasMix_x64\vi.lib\FluidVIEW\LibIdGasMix**

Filename	Effects
LibIdGasMix.llb	LabVIEW™ library file, containing every function of the LibIdGasMix property library in the form of subprograms (SubVIs)

Table 0.6 Effects of the files located in the archive directory **CD_FluidVIEW_LibIdGasMix_x64\menus\Categories\FluidVIEW**

Filename	Effects
dir.mnu	The palette view of LabVIEW™ is based on the palette files (*.mnu). They include the palette data (e. g. the display name, the palette icon, the palette description, the help information, the synchronize information and the items)

Table 0.7 Effects of the files located in the archive directory **CD_FluidVIEW_LibIdGasMix_x64 \source**

Filename	Effects
LibIdGasMix.dll	Dynamic-link library containing the algorithms for the calculation of the property functions of 25 ideal gases and their ideal mixtures
Capt_ico_big.ico	Icon file
Libmmd.dll	Runtime library
Libifcoremd.dll	Runtime library
LC.dll	Auxiliary library
Libiomp5md.dll	Runtime library

Table 0.8 Effects of the files located in the archive directory **CD_FluidVIEW_LibIdGasMix_x64\help \FluidVIEW-help**

Filename	Effects
FluidVIEW_LibIdGasMix.pdf	User's guide of the LibIdGasMix property library for the LabVIEW™ Add-On FluidVIEW
LibIdGasMix.hlp	Help file with descriptions for each function
OpenLibIdGasMix_doc.vi	LabVIEW™ instrument to open the user's guide via the help menu
LibIdGasMix.txt	Text file to change the name of the menu item of the help file
OpenLibIdGasMix_doc.txt	Text file to change the name of the menu item of the file OpenLibIdGasMix_doc.vi

Table 0.9 Effects of the files located in the archive directory **CD_FluidVIEW_LibIdGasMix_x64 \vcredist_x64**

Filename	Effects
vcredist_x64.exe	Executable file to install the Microsoft Visual C++ 2008 Redistributable Package (x64). Within runtime components of Visual C++ Libraries required to run 64-bit applications developed with Visual C++ on a computer that does not have Visual C++ 2010 installed.

1. Property Functions

1.1 Property Functions for Ideal Gas Mixtures (igmix-Functions)

Property function	Function Name	Call from Fortran	Property or function	Unit of the value calculated	Details
a = f(p,t,type,$\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)	a_pt_igmix	A_PT_IGMIX(P,T,TYPE,COMP(0:30)) C_ A_PT_IGMIX (A_P,T,TYPE,COMP(0:30))	Thermal diffusivity of the mixture	m^2/s	3/1
c_p = f(p,t,type,$\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)	cp_pt_igmix	CP_PT_IGMIX(P,T,TYPE,COMP(0:30)) C_ CP_PT_IGMIX (CP_P,T,TYPE,COMP(0:30))	Isobaric heat capacity of the mixture	kJ/(kg K)	3/2
c_v = f(p,t,type,$\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)	cv_pt_igmix	CV_PT_IGMIX(P,T,TYPE,COMP(0:30)) C_ CV_PT_IGMIX(CV_P,T,TYPE,COMP(0:30))	Isochoric heat capacity of the mixture	kJ/(kg K)	3/3
η = f(p,t,type,$\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)	eta_pt_igmix	ETA_T_IGMIX(P,T,TYPE,COMP(0:30)) C_ ETA_T_IGMIX(ETA,P,T,TYPE,COMP(0:30))	Dynamic viscosity of the mixture	Pa s = kg/(m s)	3/4
h = f(p,t,type,$\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)	h_pt_igmix	H_PT_IGMIX(P,T,TYPE,COMP(0:30)) C_ H_PT_IGMIX(P,T,TYPE,COMP(0:30))	Enthalpy of the mixture	kJ/kg	3/5
κ = f(p,t,type,$\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)	kappa_pt_igmix	KAPPA_PT_IGMIX(P,T,TYPE,COMP(0:30)) C_ KAPPA_PT_IGMIX(KAPPA,P,T,TYPE,COMP(0:30))	Isentropic exponent of the mixture		3/6
λ = f(p,t,type,$\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)	lambda_pt_igmix	LAMBDA_T_IGMIX(P,T,TYPE,COMP(0:30)) C_ LAMBDA_T_IGMIX(LAMBDA,P,T,TYPE,COMP(0:30))	Thermal conductivity of the mixture	W/(m K)	3/7
M = f(type,$\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)	M_igmix	M_IGMIX(TYPE,COMP(0:30)) C_ M_IGMIX(M,TYPE,COMP(0:30))	Molar mass of the mixture	kg/kmol	3/8
ν = f(p,t,type,$\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)	ny_pt_igmix	NY_PT_IGMIX(P,T,TYPE,COMP(0:30)) C_NY_PT_IGMIX(NY,P,T,TYPE,COMP(0:30))	Kinematic viscosity of the mixture	m^2/s	3/9

Property function	Function Name	Call from Fortran	Property or function	Unit of the value calculated	Details
p = f(p,t,type,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	p_ts_igmix	P_TS_IGMIX(T,S,TYPE,COMP(0:30)) C_ P_TS_IGMIX(P,T,S,TYPE,COMP(0:30))	Backward function: Mixture pressure from temperature and entropy	bar	3/10
p = f(t,v,type,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	p_tv_igmix	P_TV_IGMIX(T,V,TYPE,COMP(0:30)) C_ P_TV_IGMIX(P,T,V,TYPE,COMP(0:30))	Backward function: Mixture pressure from temperature and specific volume	bar	3/11
Pr = f(p,t,type,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	Pr_pt_igmix	PR_PT_IGMIX(P,T,TYPE,COMP(0:30)) C_ PR_PT_IGMIX(PR,P,T,TYPE,COMP(0:30))	Prandtl number of the mixture		3/12
ψ_i = f(igas,$\xi_1 ... \xi_{30}$)	psi_igas_xsi_igmix	PSI_IGAS_XSI_IGMIX(IGAS,XSI(0:30)) C_ PSI_IGAS_XSI_IGMIX(PSI,IGAS,XSI(0:30))	Mole fraction of the gas igas from the mass fractions of all components	kmol/kmol	3/13
R = f(type,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	R_igmix	R_IGMIX(TYPE,COMP(0:30)) C_ R_IGMIX(R,TYPE,COMP(0:30))	Specific gas constant of the mixture	kJ/(kg K)	3/14
ρ = f(p,t,type,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	rho_pt_igmix	RHO_PT_IGMIX(P,T,COMP(0:30)) C_ RHO_PT_IGMIX(RHO,P,T,COMP(0:30))	Density of the mixture	kg/m^3	3/15
s = f(p,t,type,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	s_pt_igmix	S_PT_IGMIX(P,T,TYPE,COMP(0:30)) C_ S_PT_IGMIX(S,P,T,TYPE,COMP(0:30))	Entropy of the mixture	kJ/(kg K)	3/16
t = f(p,h,type,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	t_ph_igmix	T_PH_IGMIX(P,H,TYPE,COMP(0:30)) C_ T_PH_IGMIX(T,P,H,TYPE,COMP(0:30))	Backward function: Temperature from mixture pressure and enthalpy	°C	3/17
t = f(p,s,type,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	t_ps_igmix	T_PS_IGMIX(P,S,TYPE,COMP(0:30)) C_ T_PS_IGMIX(T,P,S,TYPE,COMP(0:30))	Backward function: Temperature from mixture pressure and entropy	°C	3/18

Property function	Function Name	Call from Fortran	Property or function	Unit of the value calculated	Details
t = f(p,v,type,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	t_pv_igmix	T_PV_IGMIX(P,V, IGAS) C_ T_PV_IGMIX(T,P,V, IGAS)	Backward function: Temperature from mixture pressure and specific volume	°C	3/19
u = f(p,t,type,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	u_pt_igmix	U_PT_IGMIX(P,T, IGAS) C_ U_PT_IGMIX(U,P,T, IGAS)	Internal energy of the mixture	kJ/kg	3/20
v = f(p,t,ttype,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	v_pt_igmix	V_PT_IGMIX(P,T,TYPE,COMP(0:30)) C_ V_PT_IGMIX(V,P,T,TYPE,COMP(0:30))	Specific volume of the mixture	m^3/kg	3/21
w = f(p,t,type,$\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30}$)	w_pt_igmix	W_PT_IGMIX(P,T,TYPE,COMP(0:30)) C_ W_PT_IGMIX(W,P,T,TYPE,COMP(0:30))	Isentropic speed of sound of the mixture	m/s	3/22
ξ_i = f(igas,$\psi_1 ... \psi_{30}$)	xsi_igas_psi_igmix	XSI_IGAS_PSI_IGMIX(IGAS,PSI(0:30)) C_ XSI_IGAS_PSI_IGMIX(XSI,IGAS,PSI(0:30))	Mass fraction of the gas igas from the mole fractions of all components	kg/kg	3/23

Units

Symbol	Name	Unit
T	Temperature	°C
p	Total pressure	bar
ξ_1 ... ξ_{30}	Mass fractions of the mixture gases	kg/kg
ψ_1... ψ_{30}	Mole fractions/volume fractions of the mixture gases	kmol/kmol
type	Input parameter: type = 1 for the composition as mass fractions ξ_1, ... ξ_{30} type = 0 for the composition as mole fractions ψ_1, ... ψ_{30}	
comp(0:30) for type =1	Composition as mass fractions ξ_1, ... ξ_{30}	kg/kg
comp(0:30) for type =0	Composition as mole fractions ψ_1, ... ψ_{30}	kmol/kmol

Reference states:

Property	Gases (except steam)	Steam
Pressure	1.01325 bar	0.006112127 bar
Temperature	0 °C	0 °C
Enthalpy	0 kJ/kg	2500.9342 kJ/kg
Entropy	0 kJ/(kg K)	9.15591 kJ/(kg K)

Types of variables for the function call from the LibIdGasMix DLL:

All functions	Real*8
Variable p, T, v, h, s	Real*8
Variable comp(1..30)	Real*8
Variable type, i	Integer*4

1.2 Property Functions for Single Ideal Gases (igas-Functions)

Property function	Function Name	Call from Fortran	Property or function	Unit of the value calculated	Details
a = f(p,t,igas)	a_pt_igas	A_PT_IGAS(P,T,IGAS) C_ A_PT_IGAS(A,P,T, IGAS)	Thermal diffusivity of the gas igas	m^2/s	4/1
c_p = f(p,t,igas)	cp_pt_igas	CP_PT_IGAS(P,T, IGAS) C_ CP_PT_IGAS(CP,P,T, IGAS)	Isobaric heat capacity of the gas igas	kJ/(kg K)	4/2
c_v = f(p,t,igas)	cv_pt_igas	CV_PT_IGAS(P,T, IGAS) C_ CV_PT_IGAS(CV,P,T, IGAS)	Isochoric heat capacity of the gas igas	kJ/(kg K)	4/3
η = f(p,t,igas)	eta_pt_igas	ETA_T_IGAS(P,T, IGAS) C_ ETA_T_IGAS(ETA,P,T,IGAS)	Dynamic viscosity of the gas igas	Pa s = kg/(m s)	4/4
h = f(p,t,igas)	h_pt_igas	H_PT_IGAS(P,T, IGAS) C_ H_PT_IGAS(P,T, IGAS)	Enthalpy of the gas igas	kJ/kg	4/5
κ = f(p,t,igas)	kappa_pt_igas	KAPPA_PT_IGAS(P,T IGAS) C_ KAPPA_PT_IGAS(KAPPA,P,T IGAS)	Isentropic exponent of the gas igas		4/6
λ = f(p,t,igas)	lambda_pt_igas	LAMBDA_T_IGAS(P,T, IGAS) C_ LAMBDA_T_IGAS(LAMBDA,P,T, IGAS)	Thermal conductivity of the gas igas	W/(m K)	4/7
M = f(igas)	M_igas	M_IGAS(IGAS) C_ M_IGAS(M,IGAS)	Molar mass of the gas igas	kg/kmol	4/8
ν = f(p,t,igas)	ny_pt_igas	NY_PT_IGAS(P,T, IGAS) C_ NY_PT_IGAS(NY,P,T, IGAS)	Kinematic viscosity of the gas igas	m^2/s	4/9
p = f(t,s,igas)	p_ts_igas	P_TS_IGAS(T,S, IGAS) C_ P_TS_IGAS(P,T,S, IGAS)	Backward function: Pressure from temperature and entropy of the gas igas	bar	4/10

Property function	Function Name	Call from Fortran	Property or function	Unit of the value calculated	Details
p = f(t,v,igas)	p_tv_igas	P_TV_IGAS(T,V, IGAS) C_ P_TV_IGAS(P,T,V, IGAS)	Backward function: Pressure from temperature and specific volume of the gas igas	bar	4/11
Pr = f(p,t,igas)	Pr_pt_igas	PR_PT_IGAS(P,T, IGAS) C_ PR_PT_IGAS(PR,P,T, IGAS)	Prandtl number of the gas igas		4/12
R = f(igas)	R_igas	R_IGAS(IGAS) C_ R_IGAS(R,IGAS)	Specific gas constant of the gas igas	kJ/(kg K)	4/13
ρ = f(p,t,igas)	rho_pt_igas	RHO_PT_IGAS(P,T, IGAS) C_ RHO_PT_IGAS(RHO,P,T IGAS)	Density of the gas igas	kg/m^3	4/14
s = f(p,t,igas)	s_pt_igas	S_PT_IGAS(P,T, IGAS) C_ S_PT_IGAS(S,P,T, IGAS)	Entropy of the gas igas	kJ/(kg K)	4/15
t = f(p,h,igas)	t_ph_igas	T_PH_IGAS(P,H, IGAS) C_ T_PH_IGAS(T,P,H IGAS)	Backward function: Temperature from pressure and enthalpy of the gas igas	°C	4/16
t = f(p,s,igas)	t_ps_igas	T_PS_IGAS(P,S, IGAS) C_ T_PS_IGAS(T,P,S, IGAS)	Backward function: Temperature from pressure and entropy of the gas igas	°C	4/17
t = f(p,v,igas)	t_pv_igas	T_PV_IGAS(P,V, IGAS) C_ T_PV_IGAS(T,P,V, IGAS)	Backward function: Temperature from pressure and specific volume of the gas igas	°C	4/18
u = f(p,t,igas)	u_pt_igas	U_PT_IGAS(P,T, IGAS) C_ U_PT_IGAS(U,P,T, IGAS)	Specific internal energy of the gas igas	kJ/kg	4/19
v = f(p,t,igas)	v_pt_igas	V_PT_IGAS(P,T, IGAS) C_ V_PT_IGAS(V,P,T, IGAS)	Specific volume of the gas igas	m^3/kg	4/20
w = f(p,t,igas)	w_pt_igas	W_PT_IGAS_SI(P,T IGAS) C_ W_PT_IGAS(W,P,T, IGAS)	Isentropic speed of sound of the gas igas	m/s	4/21

Units:

Symbol	Name	Unit
t	Temperature	°C
p	Mixture pressure	bar
igas	Number of the gas	

Reference states:

Property	Gases (except steam)	Steam
Pressure	1.01325 bar	0.006112127 bar
Temperature	0 °C	0 °C
Enthalpy	0 kJ/kg	2500.9342 kJ/kg
Entropy	0 kJ/(kg K)	9.15591 kJ/(kg K)

Types of variables for the function call from the LibIdGasMix DLL:

All functions	Real*8
Variable p, t, v, h, s	Real*8
Variable igas	Integer*4

1.3 Range of Validity

Table 1 contains a list of gases which can be calculated in the LibIdGasMix property library either as a component of a gas mixture or as a single gas. The calculation of thermodynamic properties is carried out by the algorithms stated in Table 1. The algorithms for the transport properties are listed in Table 2.

The calculation programs are valid in a temperature range

from t = -73.15 °C to 3026.85 °C.

Exceptions are:

Fluorine from -73.15 °C to 976.85 °C.

The pressure range is limited to the region where the mixture gases or single gases can be considered as ideal gases and, thus, ranges

from above 0.01 bar to 10 (30) bar, maximum 50 bar.

For temperatures above 1000 °C and mole fractions of oxygen of more than 10 % ($\psi_{O_2} \geq 0.1$) the dissociation based on the VDI 4670 for the gases nitrogen, oxygen, carbon dioxide, steam, and sulfur dioxide are considered. The dissociation of other gases is not considered. For programming reasons, the calculation of the correction for the dissociation is already carried out from 500 °C.

Note:
A calculated value of -9999 indicates that the input values have been entered outside the range of validity and/or the sum of the values ξ_1, ... ξ_{30} *or* ψ_1, ... ψ_{30} *entered does not result in 1.*

Table 1. Gases and algorithms for the calculation of thermodynamic properties.

Gas no.	Gas / component		Algorithm, bibliographic reference
1	Ar	Argon	VDI 4670 [21]
2	Ne	Neon	VDI 4670 [21]
3	N_2	Nitrogen	VDI 4670 [21]
4	O_2	Oxygen	VDI 4670 [21]
5	CO	Carbon monoxide	VDI 4670 [21]
6	CO_2	Carbon dioxide	VDI 4670 [21]
7	H_2O	Steam	VDI 4670 [21]
8	SO_2	Sulfur dioxide	VDI 4670 [21]
9	AIR	Air (dry)	Mixture, VDI 4670 [1] [21]
10	AIR-N_2	Air nitrogen	Mixture, VDI 4670 [2] [21]
11	NO	Nitrogen oxide	NASA [40]
12	H_2S	Sulfur hydrogen	Span, Lemmon [41]
13	OH	Hydroxyl	NASA [40]
14	CH_3OH	Methanol	IUPAC [26]
15	CH_4	Methane	IUPAC [27]
16	C_2H_4	Ethylene	IUPAC [35]
17	C_2H_6	Ethane	Buecker [29]
18	C_3H_6	Propylene	Overhoff [42]
19	C_3H_8	Propane	Lemmon [43]
20	n-C_4H_{10}	n-Butane	Buecker [29]
21	Iso-C_4H_{10}	Iso-Butane	Buecker [29]
22	C_6H_6	Benzene	Polt [44]
23	H_2	Hydrogen	Leachman [45]
24	He	Helium	GERG [46]
25	NH_3	Ammonia	Tillner-Roth [38]/ NASA [40] [3]
26	free [4]		
27	free [4]		
28	free [4]		
29	free [4]		
30	F_2	Fluorine [5]	IUPAC [28]

1) Composition of dry air

Mole fractions	78.1109 % N_2 20.9548 % O_2 0.9343 % Ar
Mass fractions	75.5577 % N_2 23.1535 % O_2 1.2888 % Ar

2) Composition of air nitrogen

Mole fractions	98.8180 % N_2 1.1820 % Ar
Mass fractions	98.3229 % N_2 1.6771 % Ar

3) Thermodynamic properties of ammonia are calculated on the algorithms corresponding to *Tillner-Roth* [38] to a temperature of 1273.15 °C. Equations of NASA [40] are applied when calculating with

temperatures which are higher than 1273.15 °C. Data are smoothed between temperatures ranging from 1273.15 °C to 2273.15 °C.

4) The gas numbers 26 to 29 are currently not defined.

5) Due to its chemical properties, fluorine can not be calculated as a mixture gas but as a single gas.

Table 2. Gases and algorithms for the calculation of transport properties.

Gas no.	Mixture gas		Algorithm, bibliographic reference
1	Ar	Argon	Brandt [15]
2	Ne	Neon	Brandt [15]
3	N_2	Nitrogen	Brandt [15]
4	O_2	Oxygen	Brandt [15]
5	CO	Carbon monoxide	Brandt [15]
6	CO_2	Carbon dioxide	Brandt [15]
7	H_2O	Steam	Brandt [15]
8	SO_2	Sulfur dioxide	Brandt [15]
9	AIR	Air (dry)	Brandt [15]
10	AIR-N_2	Air nitrogen	Brandt [15]
11	NO	Nitrogen oxide	Brandt [15]
12	H_2S	Sulfur hydrogen	Brandt [15]
13	OH	Hydroxyl	- 6)
14	CH_3OH	Methanol	VB [33]
15	CH_4	Methane	Brandt [15]
16	C_2H_4	Ethylene	VB [33]
17	C_2H_6	Ethane	Brandt [15]
18	C_3H_6	Propylene	VB [33]
19	C_3H_8	Propane	Brandt [15]
20	C_4H_{10}	n-Butane	VB [33]
21	C_4H_{10}	Iso-Butane	VB [33]
22	C_6H_6	Benzene	VB [33]
23	H_2	Hydrogen	Brandt [15]
24	He	Helium	Brandt [15]
25	NH_3	Ammonia	Brandt [15]
26 to 29	free		
30	F_2	Fluorine	VB [33]

6) Regarding hydroxyl OH, there are no algorithms for the transport properties. The following details are valid for mixtures with the gas hydroxyl:

Mass fraction up to 10% OH	→	when calculating transport properties the fraction of OH is not considered
Mass fraction from 10% up to 100%	→	Error message -130666

Additional Information

For further information, please see Table 3 which provides data of critical points (c) and triple states (t) of the gas *i* which is determined as follows:

$t_{t,i} > t_{min}$

and/or

$t_{c,i} > t_{min}$.

This means the triple state and/or the critical point of the gas *i* is located in the LibIdGasMix range of validity. In the LibIdGasMix program, corresponding to Figure 1, for every gas i = igas is examined whether it is actually existent in the gaseous state at the given temperature *t* and its present partial pressure p_i.

If the given temperature *t* is lower than the triple temperature $t_{t,i}$ of the gas *i*, $p_i \le p_{sub,i}(t)$ has to be fulfilled with $p_{sub,i}(t)$ as the sublimation pressure of the gas number *i*; see Figure 1. If not, the result is - xx999, for which xx is the number of the gas corresponding to Table 1. This test is carried out for H_2O and CO_2.

If the given temperature *t* has a value between the triple temperature and the critical temperature the relation $p_i \le p_{s,i}(t)$ has to be valid for the partial pressure p_i, where $p_{s,i}(t)$ is the saturation pressure of the gas *i*; see Figure 1. If not, the result will again be - xx999. The gases to be tested are listed in Table 3. This table also contains values of the critical and tripel states.

The calculation is carried out in any case at temperatures above the critical temperature.

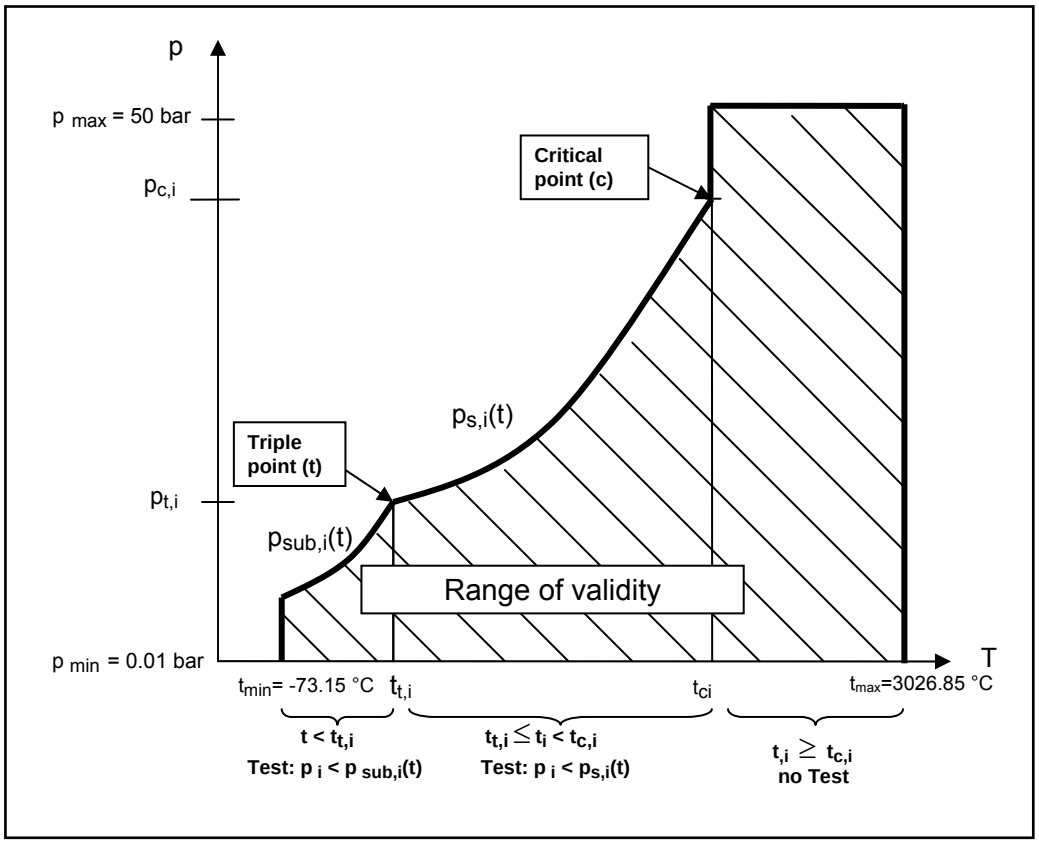


Figure 1. p-t diagram with the range of validity of a gas i = igas of the LibIdGasMix property library.

Table 3. Data of triple states (t) and critical points (c).

			Critical point			Triple state		
Gas No. i	**Mixture gas**		**$p_{c\,i}$ in bar[Source]**	**$t_{c\,i}$ in °C[Source]**	**$p_s(t)$**	**$p_{t\,i}$ in bar[Source]**	**$t_{t\,i}$ in °C [Source]**	**$p_{sub}(t)$[Source]**
1	Ar	Argon						
2	Ne	Neon						
3	N_2	Nitrogen						
4	O_2	Oxygen						
5	CO	Carbon monoxide						
6	CO_2	Carbon dioxide	73.773 [36]	30.9782 [36]	[36]	5.1795 [36]	- 56.558 [36]	[36]
7	H_2O	Steam	220.69 [39]	373.946 [39]	[39]	0.00611657 [39]	0.01 [39]	[39]
8	SO_2	Sulfur dioxide	78.8 [20]	157.45 [20]	[20]			
9	AIR	Air (dry)						
10	AIR-N_2	Air nitrogen						
11	NO	Nitrogen oxide						
12	H_2S	Sulfur hydrogen		99.95 [41]	[41]			
13	OH	Hydroxyl						
14	CH_3OH	Methanol	81.035 [26]	239.45 [26]	[26]			
15	CH_4	Methane						
16	C_2H_4	Ethylene	50.418 [35]	9.2 [35]	[35]			
17	C_2H_6	Ethane	48.722 [26]	32.172 [26]	[29]			
18	C_3H_6	Propylene	46.646 [42]	92.42 [42]	[42]			
19	C_3H_8	Propane	42.4766 [43]	96.675 [43]	[43]			
20	n-C_4H_{10}	n-Butane	37.96 [26]	151.975 [26]	[26]			
21	Iso-C_4H_{10}	Iso-Butane	36.29 [26]	134.66 [26]	[26]			
22	C_6H_6	Benzene	48.9794 [44]	289.01 [44]	[44]			
23	H_2	Hydrogen						
24	He	Helium						
25	NH_3	Ammonia	113.3926 [38],[40]	132.36 [38],[40]	[38],[40]			
26 to 29	free		-	-		-	-	
30	F_2	Fluorine[2)]						

2 Application of FluidVIEW in LabVIEW™

The FluidVIEW Add-on has been developed to calculate thermodynamic properties in LabVIEW™ (version 10.0 or higher) more conveniently. Within LabVIEW™, it enables the direct call of functions relating to ideal gas mixtures from the LibIdGasMix property library.

2.1 Installing FluidVIEW

If a FluidVIEW property library has not yet been installed, please complete the initial installation procedure described below.

If a FluidVIEW property library has already been installed, you only need to copy several files which belong to the LibIdGasMix library. In this case, follow the subsection "Adding the LibIdGasMix Library" on page 2/3.

In both cases folders and files from the zip archive

CD_FluidVIEW_LibIdGasMix.zip	(for 32-bit version of Windows®)
CD_FluidVIEW_LibIdGasMix_x64.zip	(for 64-bit version of Windows®)

have to be copied into the default directory of the LabVIEW™ development environment. In the following text these zipped directories for the 32-bit or 64-bit operating system will be symbolised with the term **<CD>**.

You can see the current default directory of LabVIEW™ in the paths page (options dialog box). To display this page please select *Tools* and click on *Options* to open the options dialog box and then select *Paths* from the category list.

By choosing *Default Directory* from the drop-down list the absolute pathname to the default directory, where LabVIEW™ automatically stores information, is displayed. In the following sections the pathname of the default directory will be symbolised by the term **<LV>**.

Additional Requirement When Using a 64-bit Operating System

If you want to use FluidVIEW on a 64-bit computer that does not have Visual C++ installed, please make sure the Microsoft Visual C++ 2010 x64 Redistributable Package is installed.

If it is not the case, please install it by double clicking the file

vcredist_x64.exe

which you find in the folder **\vcredist_x64** in the **64-bit** CD folder "CD_FluidVIEW_LibIdGasMix_x64."

In the following window you are required to accept the Microsoft® license terms to install the Microsoft Visual C++ 2010 runtime libraries by ticking the box next to "I have read and accept the license terms" (see Figure 2.1).

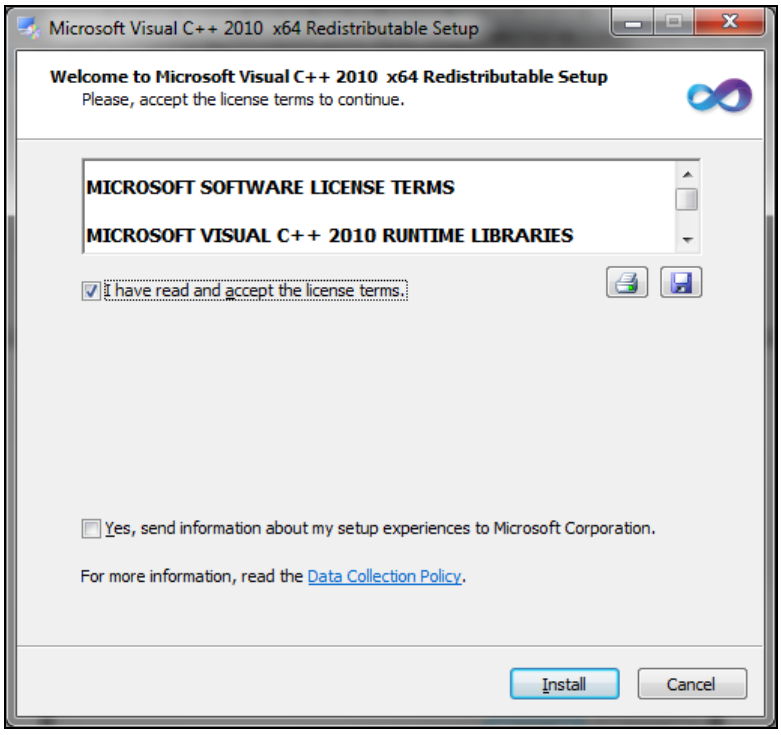


Figure 2.1 Accepting the license terms to install the Microsoft Visual C++ 2010 x64 Redistributable Package

Now click on "Install" to continue installation.

After the "Microsoft Visual C++ 2010 x64 Redistributable Pack" has been installed, you will see the sentence "Microsoft Visual C++ 2010 x64 Redistributable has been installed." Confirm this by clicking "Finish."

Now you can use the FluidVIEW Add-On on your 64-bit operating system. Please follow the instructions below to install FluidVIEW.

Initial Installation of FluidVIEW

The initial installation of FluidVIEW is carried out by copying three directories with its contents from the zip archive to the standard directory of LabVIEW™. The directories that have to be copied, their paths in the zip archive and their target paths are listed in Table 2.1.

The installation is complete after copying the files and restarting LabVIEW™.

Due to the fact, that the functions of the DLL are called with a variable pathname, the source files you will find in the directory **<CD>\source** can be stored in a random directory. The pathname has to be indicated in order to calculate the property functions.

All source files have to be stored in the same directory to make the property functions of the LibIdGasMix library work. These files are for the

- **32-bit system**: LibIdGasMix.dll, advapi32.dll, Dformd.dll, Dforrt.dll, LC.dll, msvcp60.dll, and msvcrt.dll

and for the

- **64-bit system**: LibIdGasMix.dll, capt_ico_big.ico, LC.dll, libifcoremd.dll, libiomp5md.dll, and libmmd.dll.

Table 2.1 Directories which have to be copied from the zip archive in the default directory of LabVIEW™ (<LV>) for the initial installation of FluidVIEW

Name of the directory	Pathname in the zip archive	Target path in the default directory of LabVIEW (<LV>)
FluidVIEW	<CD>\vi.lib	<LV>\vi.lib
FluidVIEW	<CD>\menus\Categories	<LV>\menus\Categories
FluidVIEW-Help	<CD>\help	<LV>\help

Adding the LibIdGasMix Library

In order to add the LibIdGasMix property library to an existing FluidVIEW installation, one folder with its contents and five files have to be copied from the zip archive to the standard directory of LabVIEW™. This directory, the files plus their pathnames in the zip archive and their target paths are listed in Table 2.2.

The installation is complete after copying the files and restarting LabVIEW™. Due to the fact, that the functions of the DLL are called with a variable pathname, the source files you will find in the directory **<CD>\source** can be stored in a random directory. The pathname has to be indicated in order to calculate the property functions. All source files have to be stored in the same directory to make the property functions of the LibIdGasMix library work. These files are for the

- **32-bit system**: LibIdGasMix.dll, advapi32.dll, Dformd.dll, Dforrt.dll, LC.dll, msvcp60.dll, and msvcrt.dll

and for the

- **64-bit system**: LibIdGasMix.dll, capt_ico_big.ico, LC.dll, libifcoremd.dll, libiomp5md.dll, and libmmd.dll

Table 2.2 Data which have to be copied from the zip archive in the default directory of LabVIEW™ (<LV>) for adding the LibIdGasMix property library to an existing installation of FluidVIEW

File name with file extension or name of the directory	Pathname in the zip archive	Target path in the default directory of LabVIEW (<LV>)
LibIdGasMix.llb	<CD>\vi.lib\FluidVIEW	<LV>\vi.lib\FluidVIEW
LibIdGasMix	<CD>\menus\Categories \FluidVIEW	<LV>\menus\Categories \FluidVIEW
LibIdGasMix.hlp	<CD>\\help\FluidVIEW-Help	<LV>\help\FluidVIEW-Help
LibIdGasMix.txt	<CD>\\help\FluidVIEW-Help	<LV>\help\FluidVIEW-Help
FluidVIEW_LibIdGasMix.pdf	<CD>\\help\FluidVIEW-Help	<LV>\help\FluidVIEW-Help
Open_LibIdGasMix_doc.vi	<CD>\\help\FluidVIEW-Help	<LV>\help\FluidVIEW-Help
Open_LibIdGasMix_doc.txt	<CD>\\help\FluidVIEW-Help	<LV>\help\FluidVIEW-Help

After you have restarted LabVIEW™ you will find the functions of the LibIdGasMix property library in the functions palette under the sub palette FluidVIEW. An example calculation of the specific enthalpy *h* and the specific entropy *s* is shown in section 2.4.

2.2 The FluidVIEW Help System

FluidVIEW provides detailed online help functions. If you are running Windows Vista or Windows 7, please note the paragraph
"Using the FluidVIEW Online-Help in Windows Vista or Windows 7."

General Information

The FluidVIEW Help System consists of the Microsoft WinHelp file **LibIdGasMix.hlp** and this user's guide as PDF document **FluidVIEW_LibIdGasMix.pdf**. Both files can be opened via the help menu. To do this please click *Help* in the menu bar. In the submenu *FluidVIEW-Help* you will find the commands *LibIdGasMix Help File* and *LibIdGasMix User's Guide* to open an appropriate file.

Context-Sensitive Help

If you have activated the context help function in LabVIEW™ (Ctrl-H) and move the cursor over a FluidVIEW object basic information is displayed in the context help window. The in- and output parameters plus a short information text are displayed for a property function. By clicking the **Detailed help** button in the **Context help** window the online help will be opened. The context help window of the function v_ptx_air.vi is shown in Figure 2.2.

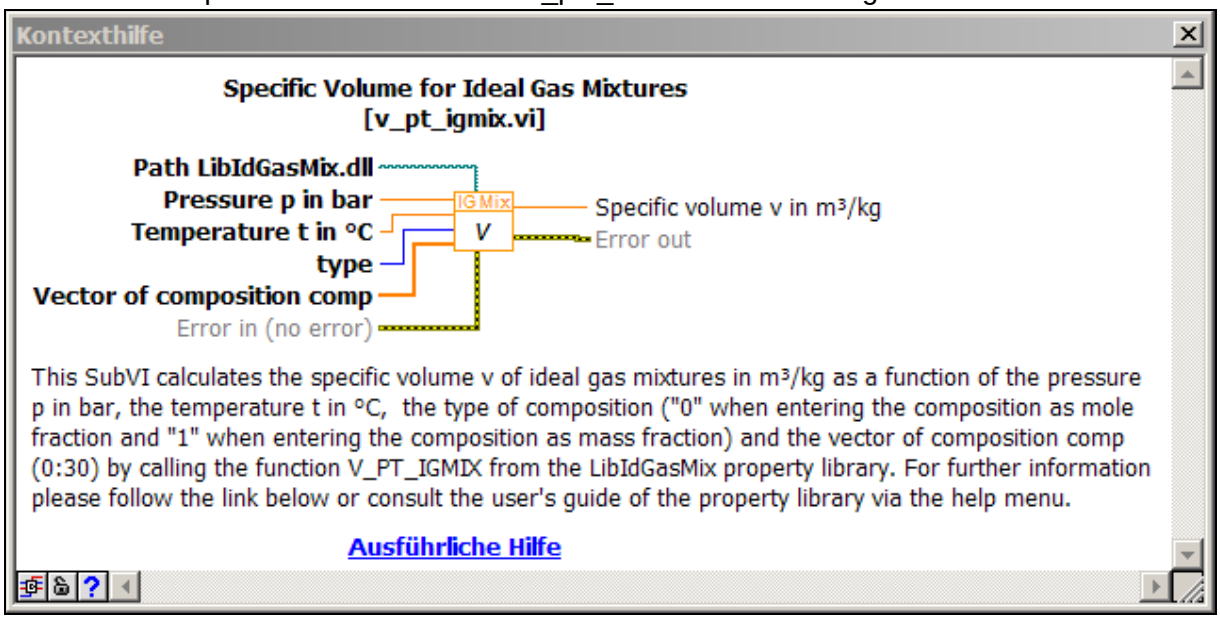


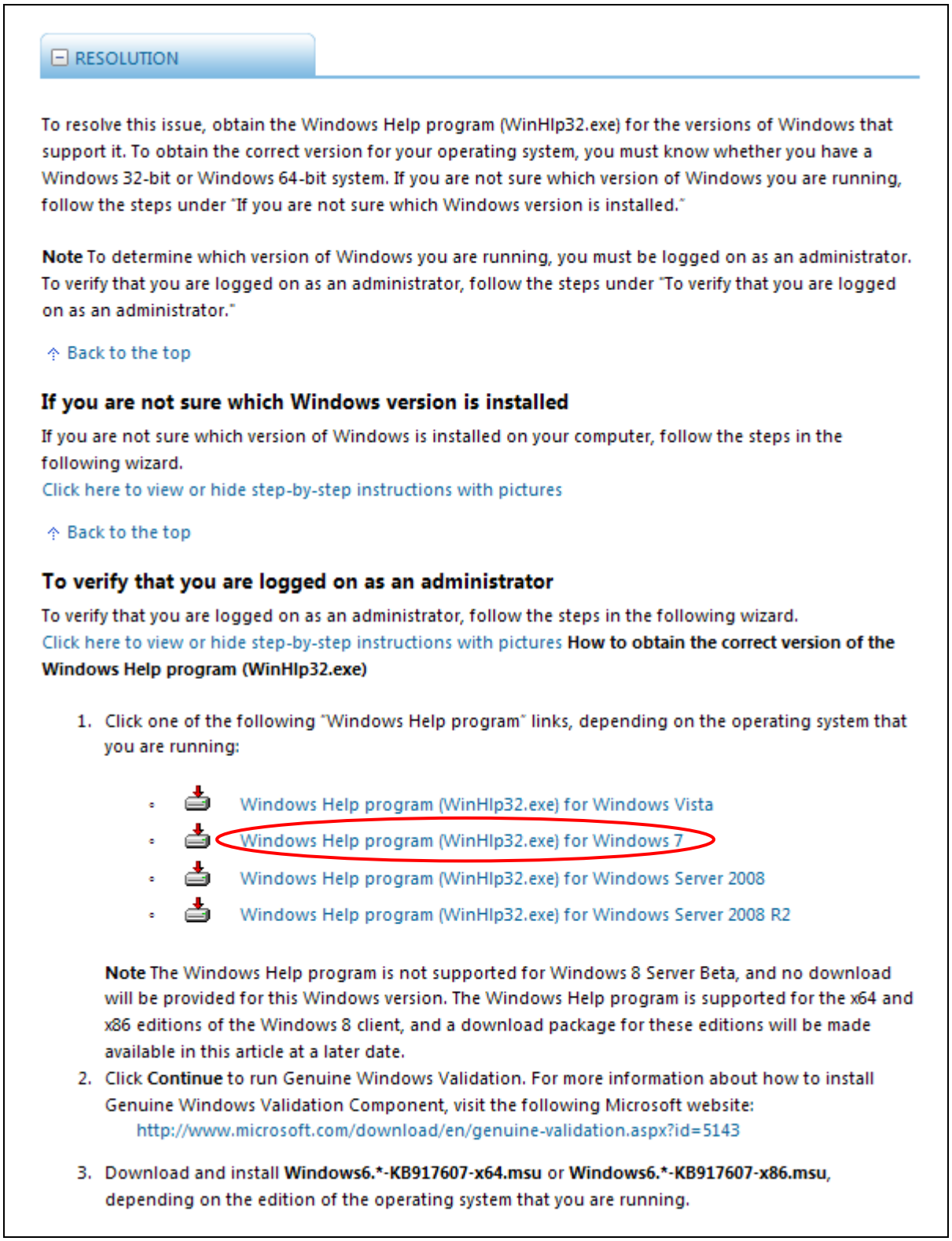
Figure 2.2 Context help window of the function v_ptx_air.vi

Using the FluidVIEW Online-Help in Windows Vista or Windows 7

If you are running Windows Vista or Windows 7 on your computer, you might not be able to open Help files. To view these files you have to install the Microsoft® Windows Help program which is provided by Microsoft®. Please carry out the following steps in order to download and install the Windows Help program. The description relates to Windows® 7.

The procedure is analogous for Windows® Vista.

Open Microsoft Internet Explorer® and go to http://support.microsoft.com/kb/917607. Scroll down until you see the headline "Resolution". Under the first Point you'll find the links to download the Windows Help program. Click on the link "Windows Help program (WinHlp32.exe) for Windows 7" (see Figure 2.3)



RESOLUTION

To resolve this issue, obtain the Windows Help program (WinHlp32.exe) for the versions of Windows that support it. To obtain the correct version for your operating system, you must know whether you have a Windows 32-bit or Windows 64-bit system. If you are not sure which version of Windows you are running, follow the steps under "If you are not sure which Windows version is installed."

Note To determine which version of Windows you are running, you must be logged on as an administrator. To verify that you are logged on as an administrator, follow the steps under "To verify that you are logged on as an administrator."

Back to the top

If you are not sure which Windows version is installed

If you are not sure which version of Windows is installed on your computer, follow the steps in the following wizard.
Click here to view or hide step-by-step instructions with pictures

Back to the top

To verify that you are logged on as an administrator

To verify that you are logged on as an administrator, follow the steps in the following wizard.
Click here to view or hide step-by-step instructions with pictures **How to obtain the correct version of the Windows Help program (WinHlp32.exe)**

1. Click one of the following "Windows Help program" links, depending on the operating system that you are running:
   - Windows Help program (WinHlp32.exe) for Windows Vista
   - Windows Help program (WinHlp32.exe) for Windows 7
   - Windows Help program (WinHlp32.exe) for Windows Server 2008
   - Windows Help program (WinHlp32.exe) for Windows Server 2008 R2

   Note The Windows Help program is not supported for Windows 8 Server Beta, and no download will be provided for this Windows version. The Windows Help program is supported for the x64 and x86 editions of the Windows 8 client, and a download package for these editions will be made available in this article at a later date.
2. Click **Continue** to run Genuine Windows Validation. For more information about how to install Genuine Windows Validation Component, visit the following Microsoft website:
   http://www.microsoft.com/download/en/genuine-validation.aspx?id=5143
3. Download and install **Windows6.*-KB917607-x64.msu** or **Windows6.*-KB917607-x86.msu**, depending on the edition of the operating system that you are running.

Figure 2.3 Selecting your Windows® Version

You will be forwarded to the Microsoft Download Center where you can download the Microsoft Windows Help program. First, a validation of your Windows License is required. To do this click on the "Continue" button (see Figure 2.4).

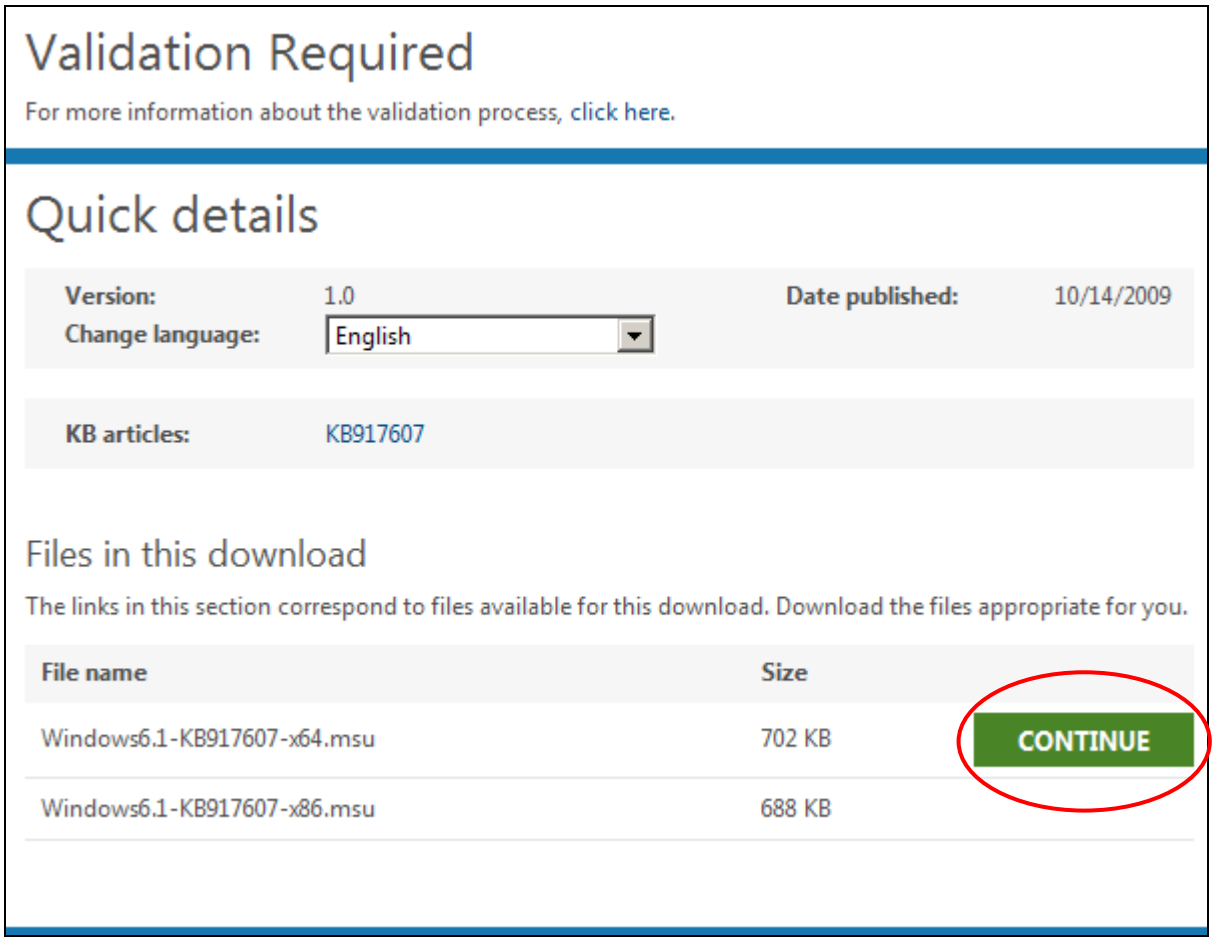


Figure 2.4 Microsoft® Download Center

Afterwards a web page with instructions on how to install the Genuine Windows Validation Component opens. At the top of your Windows Internet Explorer you will see a yellow information bar. It reads

> *"This website wants to install the following add-on: 'Windows Genuine Advantage' from 'Microsoft Corporation'. If you trust this website and the add-on and want to install it, click here."*

Right-click this bar and select "Install ActiveX Control" in the context menu. A dialog window appears in which you are asked if you want to install the software. Click the "Install" button to continue. After the validation has been carried out you will be able to download the appropriate version of Windows Help program (see Figure 2.5).

To download and install the correct file you need to know which Windows version (32-bit or 64-bit) you are running on your computer.

If you are running a 64-bit operating system, please download the file

Windows6.1-KB917607-x64.msu.

If you are running a 32-bit operating system, please download the file

Windows6.1-KB917607-x86.msu.

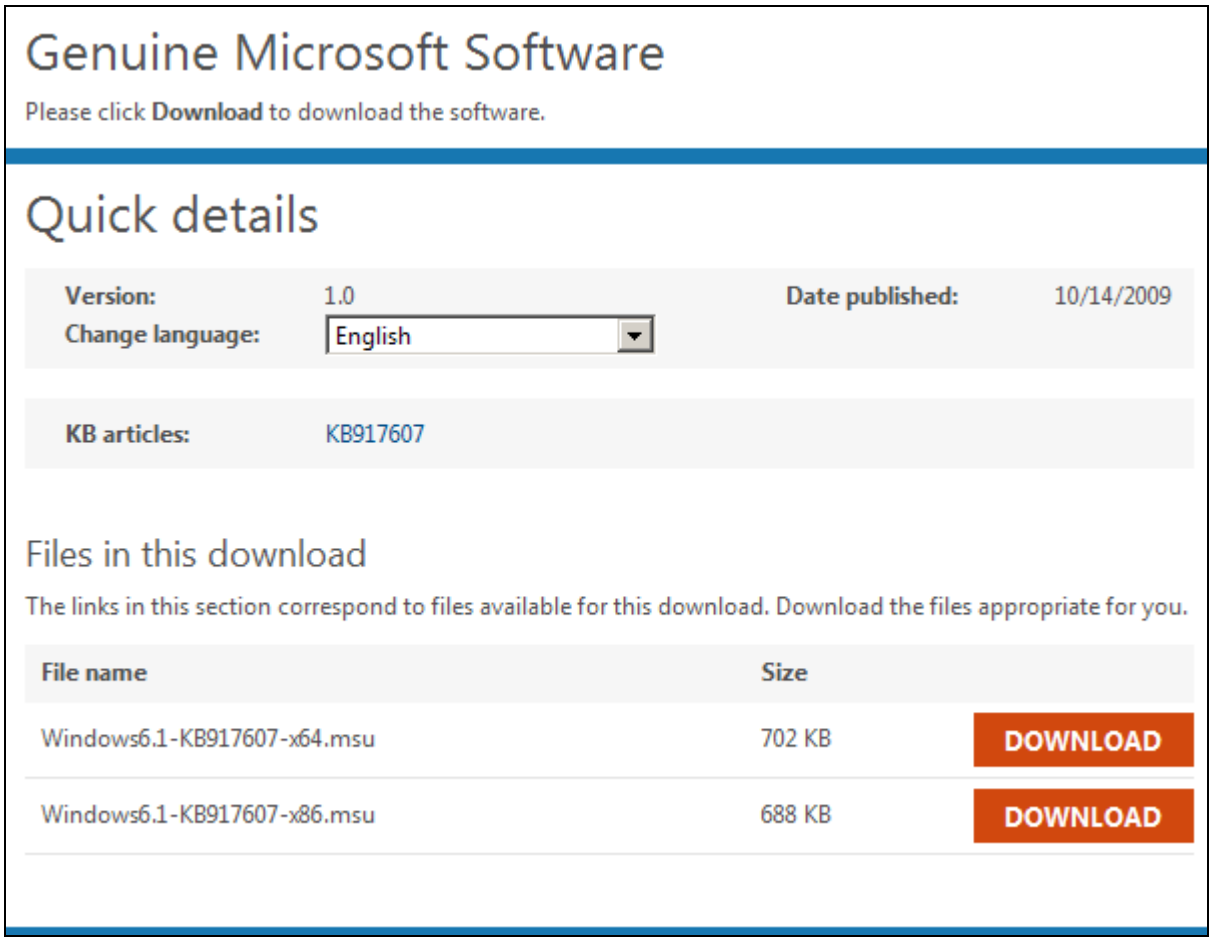


Figure 2.5 Downloading the Windows Help Program

In order to run the installation of the Windows Help program double-click the file you have just downloaded on your computer.

Installation starts with a window searching for updates on your computer.

After the program has finished searching you may be asked, if you want to install the "Update for Windows (KB917607)."

(If you have already installed this update, you will see the message "Update for Windows (KB917607) is already installed on this computer.")

The installation can be continued by clicking the "Yes" button.

In the next window you have to accept the Microsoft license terms before installing the update by clicking on "I Accept".

After the Windows Help program has been installed, the notification "Installation complete" will appear. Confirm this by clicking the "Close" button.

The installation of the Windows Help program has been completed and you will now be able to open the Help files.

2.3 Licensing the LibIdGasMix Property Library

The licensing procedure has to be carried out when calculating a LibIdGasMix function and a FluidVIEW prompt message appears. In this case, you will see the "License Information" window (see figure below).

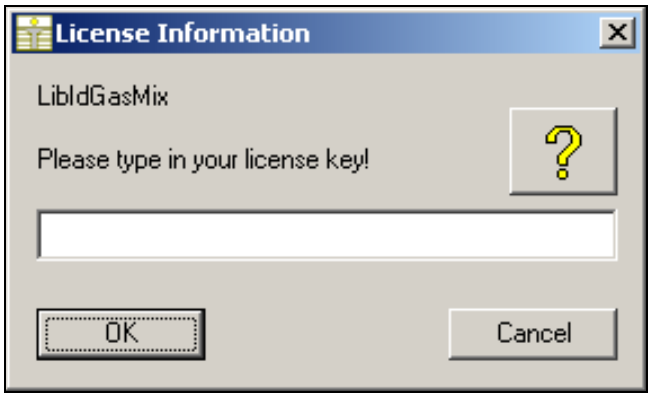


Figure 2.6 "License Information" window

Here you will have to type in the license key which you have obtained from the Zittau/Goerlitz University of Applied Sciences. You can 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:

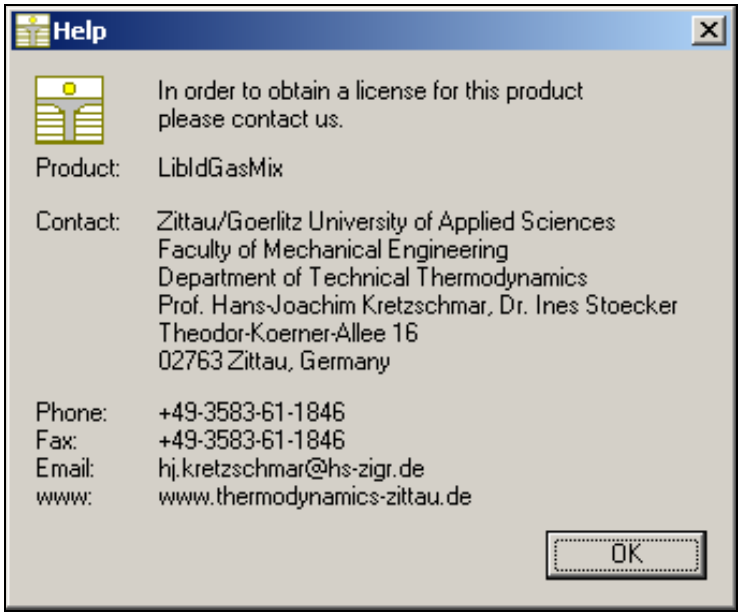


Figure 2.7 "Help" window

If you do not enter a valid license it is still possible to run your VI by clicking "Cancel". In this case, the LibIdGasMix property library will display the result "-1.11111E+7" for every calculation.

The "License Information" window will appear every time you reopen your Virtual Instrument (VI) or reload the path of the LibIdGasMix.dll. Should you not wish to license the LibIdGasMix property library, you have to uninstall FluidVIEW according to the description in section 2.6 of this User's Guide.

2.4 Example: Calculation of $h = f(p,t,\xi_1 \ldots \xi_{30})$ of the Gas Mixture

After the delivered files have been copied in the appropriate folders of the default directory LabVIEW™ (described in section 2.1), the LibIdGasMix property library is ready to use. The function nodes of the LibIdGasMix property library can be used by dragging them from the functions palette into the block diagram and connecting them with the wires representing the required input parameters.

Now we will calculate, step by step, the specific enthalpy h as a function of pressure of p = 1.45 bar and a temperature of t = 100 °C for a given mixture composed of the following mass fractions using FluidVIEW.

Mass fraction in %	Mixture gas
12	Neon
9	Steam
21	Air
39	Ethylene
14	n-Butane
5	Hydrogen

Corresponding to table 1 in chapter 1.1 the given mixture gases take the following numbers in the LibIdGasMix library. The given mass fractions have been added:

Gas no. i	Mixture gas	ξ_i in kg/kg
2	Neon	0.12
7	Steam	0.09
9	Air	0.21
16	Ethylene	0.39
20	n-Butane	0.14
23	Hydrogen	0.05

You have to carry out the following steps to calculate the specific enthalpy h:

- Start LabVIEW™ and wait for the *Getting Started* window to be displayed. Then select *Blank VI*. The *Blank VI* will be displayed in two windows, the front panel and the block diagram.
- Open the functions palette in the block diagram **via view / Functions Palette** (or by clicking the right mouse button anywhere in the free area of the block diagram) if not yet displayed.
- In addition to the default LabVIEW™ palettes the functions palette contains the sub palette *FluidVIEW* (see Figure 2.8) with the sub palette *LibIdGasMix* (see Figure 2.9).

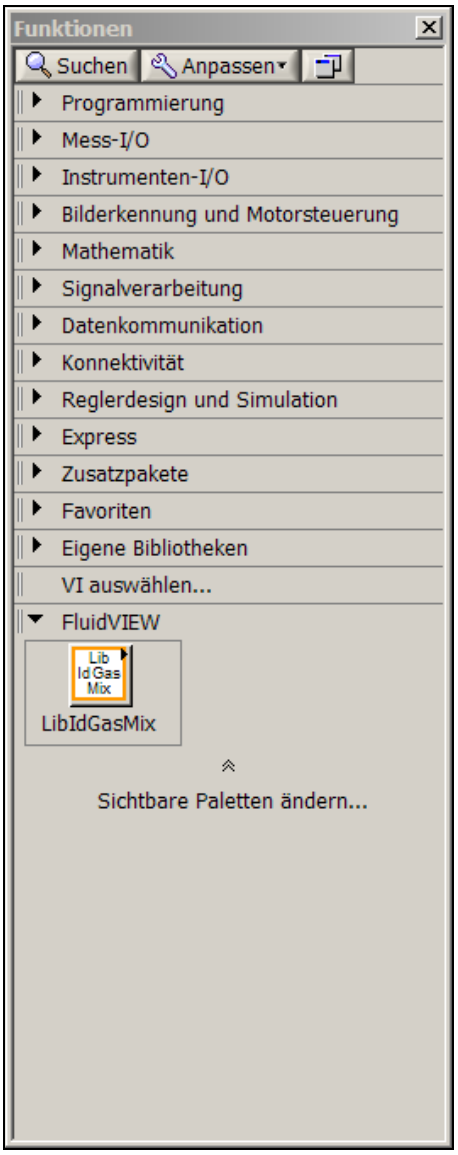


Figure 2.8
Functions palette with the sub palettes FluidVIEW and LibHuGas

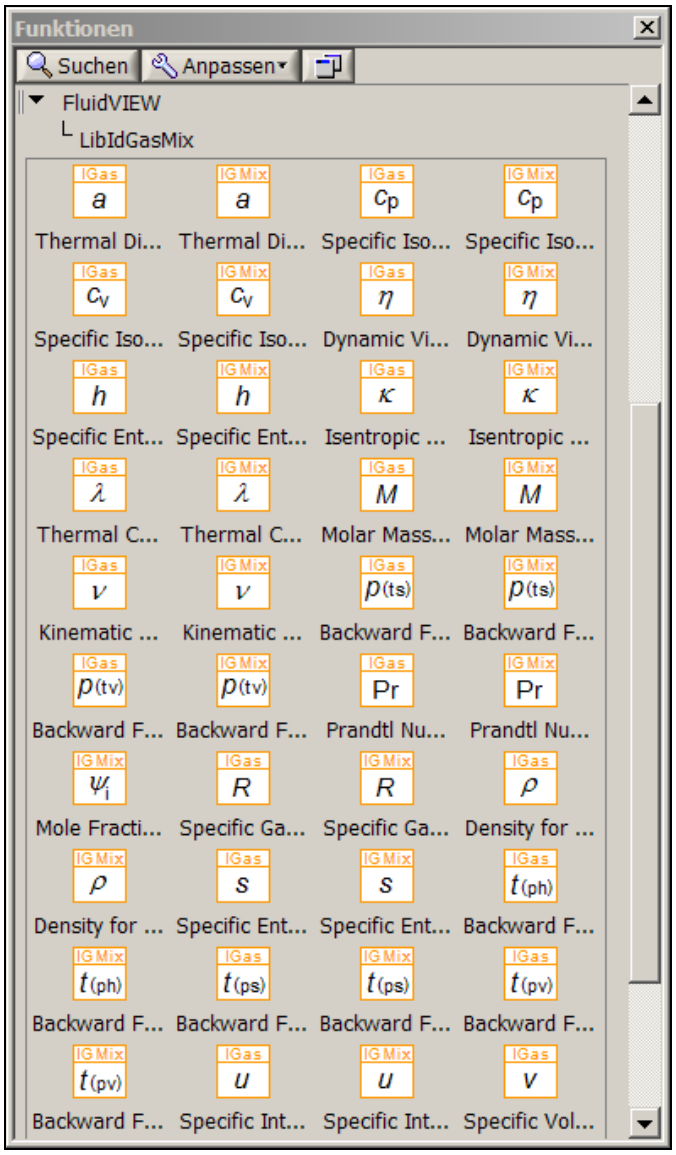


Figure 2.9
Functions palette with the property functions of the LibHuGas library

In order to calculate the specific enthalpy *h*, drag the function (SubVI) whose symbol shows the *h* from the functions palette into the block diagram.
While the short names of the SubVIs behind the symbols will be shown in the control tip, the full names and brief descriptions of the property functions are displayed in the *Context Help* window (see Figure 2.2). To use the context help press <Ctrl>+<H> on your keyboard.

- After placing the node of the SubVI **h_pt_igmix.vi** on your block diagram the required input parameters have to be defined. The input parameters which are set as required appear in bold type in the Context Help window. In this case these input parameters are **Path LibIdGasMix.dll** (LabVIEW™ data type: Path), **Pressure p in bar** (LabVIEW™ data type: Double precision, floating-point), **Temperature t in °C** (LabVIEW™ data type: Double precision, floating-point) **type** (LabVIEW™ data type: Long signed integer) and **Vector of composition comp** (1-D Array, LabVIEW™ data type: Double precision, floating-point).

- To define these variables wire their input terminals with input elements on the front panel. You can accomplish this in one step by choosing **Create / Control** in the context menu of all required input terminals. In order to wire the output terminal of the function node with an output element on the front panel, choose **Create / Indicator** in the context menu of the output terminal **Specific enthalpy h in kJ/kg** (LabVIEW™ data type: Double precision, floating-point). After cleaning up the block diagram by pressing <Ctrl>+<U> it has the appearance illustrated in Figure 2.10. The same input and output elements are available on the appropriate front panel (see Figure 2.11).

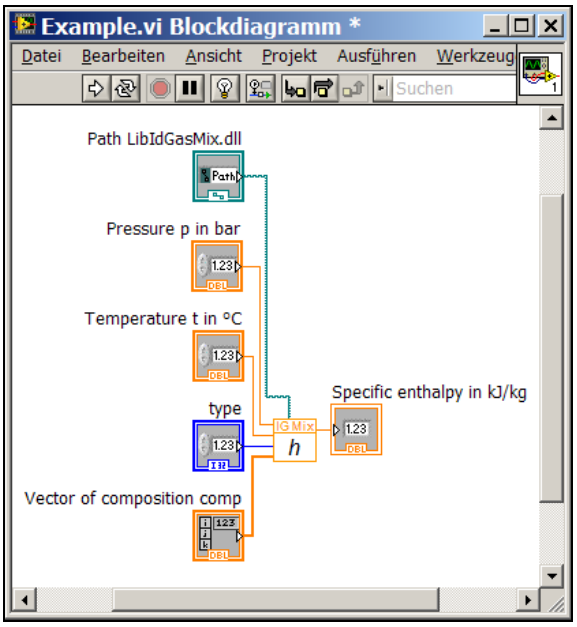


Figure 2.10
Block diagram of the example calculation

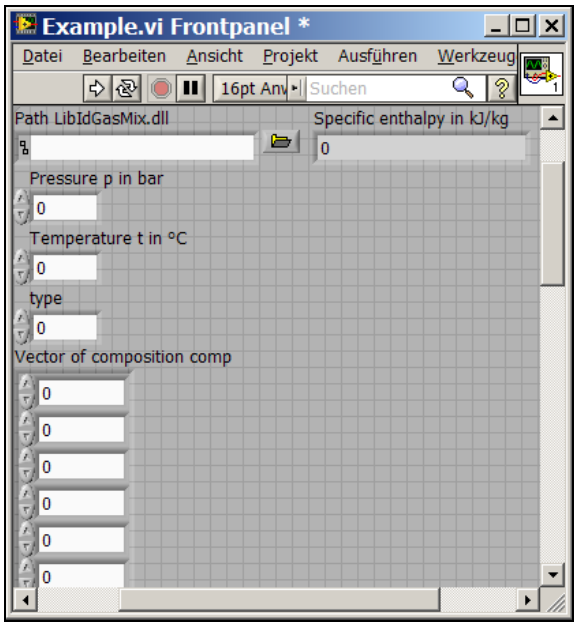


Figure 2.11
Front panel of the example calculation

- Enter the path of the LibIdGasMix.dll in the input element *Path LibIdGasMix.dll* on the front panel (as explained in section 2.1 the LibIdGasMix.dll and the other library files from the directory **<CD>\source** have to be stored in the same directory which is arbitrary). To do this you can use the *File Open Dialog* which appears by clicking the yellow folder symbol on the right of the input element.
- Enter a value in the input element *pressure p in bar* on the front panel
  (Range of validity: p = 0.01 mbar up to 50 bar)

  ⇒ e. g.: Enter the value 1.45
- Enter a value in the input element *temperature t in °C* on the front panel
  (Range of validity: t = - 73.15 °C … 3026.85 °C)

  ⇒ e. g.: Enter the value 100
- Enter a specification into the input element *type* on the front panel. This input determines whether the entry of the composition vector comp will be made in mass fractions or mole fractions, i.e., volume fractions.

  Type = 1 for the entry of mass fractions ξ_1 … ξ_{30}

  Type = 0 for the entry of mole fractions, i.e., volume fractions ψ_1 … ψ_{30}

  ⇒ e. g.: Enter the value 1

- Enter the given mass fractions ξ_1 ... ξ_{30} of the mixture gases into the input element *Vector of composition comp* on the frontpanel. In doing so, please consider, that the vector *comp* consists of the elements 0 to 30. Whereas the element 0 is a dummy.

ξ_1	for argon	Ar	⇒ e.g.: Enter 0	into element 01
ξ_2	for neon	Ne	⇒ e.g.: Enter 0.12	into element 02
ξ_3	for nitrogen	N_2	⇒ e.g.: Enter 0	into element 03
ξ_4	for oxygen	O_2	⇒ e.g.: Enter 0	into element 04
ξ_5	for carbon monoxide	CO	⇒ e.g.: Enter 0	into element 05
ξ_6	for carbon dioxide	CO_2	⇒ e.g.: Enter 0	into element 06
ξ_7	for steam	H_2O	⇒ e.g.: Enter 0.09	into element 07
ξ_8	for sulfur dioxide	SO_2	⇒ e.g.: Enter 0	into element 08
ξ_9	for air - dry		⇒ e.g.: Enter 0.21	into element 09
ξ_{10}	for atmospheric nitrogen		⇒ e.g.: Enter 0	into element 10
ξ_{11}	for nitrogen oxide	NO	⇒ e.g.: Enter 0	into element 11
ξ_{12}	for hydrogen sulfide	H_2S	⇒ e.g.: Enter 0	into element 12
ξ_{13}	for hydroxyl	OH	⇒ e.g.: Enter 0	into element 13
ξ_{14}	for methanol	CH_3OH	⇒ e.g.: Enter 0	into element 14
ξ_{15}	for methane	CH_4	⇒ e.g.: Enter 0	into element 15
ξ_{16}	for ethylene	C_2H_4	⇒ e.g.: Enter 0.39	into element 16
ξ_{17}	for ethane	C_2H_6	⇒ e.g.: Enter 0	into element 17
ξ_{18}	for propylene	C_3H_6	⇒ e.g.: Enter 0	into element 18
ξ_{19}	for propane	C_3H_8	⇒ e.g.: Enter 0	into element 19
ξ_{20}	for n-butane	C_4H_{10}	⇒ e.g.: Enter 0.14	into element 20
ξ_{21}	for iso-butane	C_4H_{10}	⇒ e.g.: Enter 0	into element 21
ξ_{22}	for benzene	C_6H_6	⇒ e.g.: Enter 0	into element 22
ξ_{23}	for hydrogen	H_2	⇒ e.g.: Enter 0.05	into element 23
ξ_{24}	for helium	He	⇒ e.g.: Enter 0	into element 24
ξ_{25}	for ammonia	NH_3	⇒ e.g.: Enter 0	into element 25
ξ_{26}	free		⇒ e.g.: Enter 0	into element 26
ξ_{27}	free		⇒ e.g.: Enter 0	into element 27
ξ_{28}	free		⇒ e.g.: Enter 0	into element 28
ξ_{29}	free		⇒ e.g.: Enter 0	into element 29
ξ_{30}	for fluorine	F_2	⇒ e.g.: Enter 0	into element 30

Note!

The gas numbers 26 to 29 are currently not defined. All entries above 0 (zero) made into these cells result in – xx0999, for which xx corresponds to the gas number. Please note that due to its chemical properties, fluorine can be calculated only as a mixture gas with ξ_{30} = 1 or ψ_{30} = 1. If it is entered as a mixture gas with ξ_{30} < 1 or ψ_{30} < 1 the value calculated will result in 30999.

- To run the calculation of the specific enthalpy click on the *Run* button or press <Ctrl>+<R>. The result for *h* in kJ/kg appears in the output element (see Figure 2.12).

  ⇒ The result for *h* in our sample calculation is h = 435.528553 kJ/kg.

  The calculation of $h = f(p,t, \xi_1 \ldots \xi_{30})$ has thus been completed.

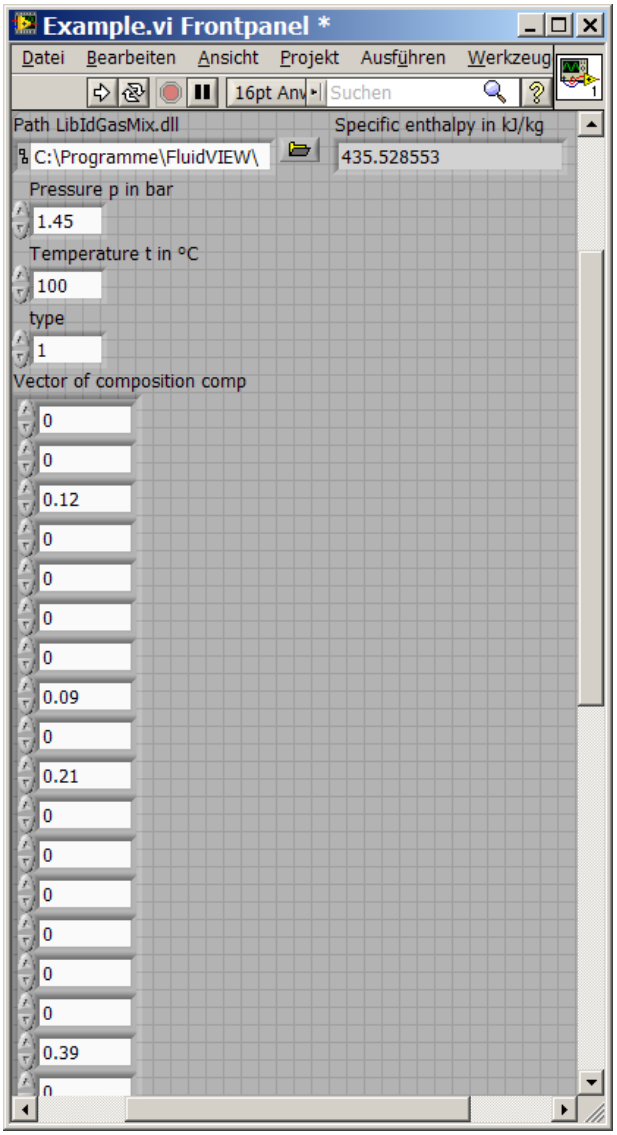


Figure 2.12 Result of the example calculation of *h*

Important!

LibIdGasMix checks internally whether the partial pressures of the mixture gases comprising the composition and the total pressure are lower than the saturation pressure at a given temperature. If this not be the case, the calculation will result in "-xx999", for which xx corresponds to the gas number, e.g., -7999 for steam.

2.5 Example: Calculation of the Mole Fraction ψ_i = f(igas, ξ_1 ... ξ_{30}) of the Gas i within the Gas Mixture

After calculating a property function depending on pressure *p*, temperature *t* and the mass fraction of the mixture components (described in part 2.4), now we will compute the mole fraction of ethylene (C_2H_4). In addition to the mass fractions of the mixture components only the gas number igas has to be specified. Carry out the following steps:

- Open the functions palette in the block diagram **via view / Functions Palette** (or by clicking the right mouse button anywhere in the free area of the block diagram) if not yet displayed.
- In addition to the default LabVIEW™ palettes the functions palette contains the sub palette *FluidVIEW* (see Figure 2.8) with the sub palette *LibIdGasMix* (see Figure 2.9). In order to calculate the mole fraction ψ, drag the *igmix* function (SubVI) whose symbol shows the ψ from the functions palette into the block diagram. While the short names of the SubVIs behind the symbols will be shown in the control tip (in this case “Mole Fraction of the Gas igas for Ideal Gas Mixtures”), the full names and brief descriptions of the property functions are displayed in the *Context Help* window (see Figure 2.2). To use the context help press <Ctrl>+<H> on your keyboard.
- After placing the node of the SubVI **psi_igas_xsi_igmix.vi** on your block diagram the required input parameters have to be defined. The input parameters which are set as required appear in bold type in the Context Help window. In this case these input parameters are **Path LibIdGasMix.dll** (LabVIEW™ data type: Path), the gas number **igas** (LabVIEW™ data type: Long signed integer) and **Vector of the mass fractions of all components xsi** (1-D Array, LabVIEW™ data type: Double precision, floating-point).
- To define these variables wire their input terminals with input elements on the front panel. You can accomplish this in one step by choosing **Create / Control** in the context menu of all required input terminals. In order to wire the output terminal of the function node with an output element on the front panel, choose **Create / Indicator** in the context menu of the output terminal **Mole fraction psi** (LabVIEW™ data type: Double precision, floating-point). After cleaning up the block diagram by pressing <Ctrl>+<U> it has the appearance illustrated in Figure 2.13. The same input and output elements are available on the appropriate front panel (see Figure 2.14).

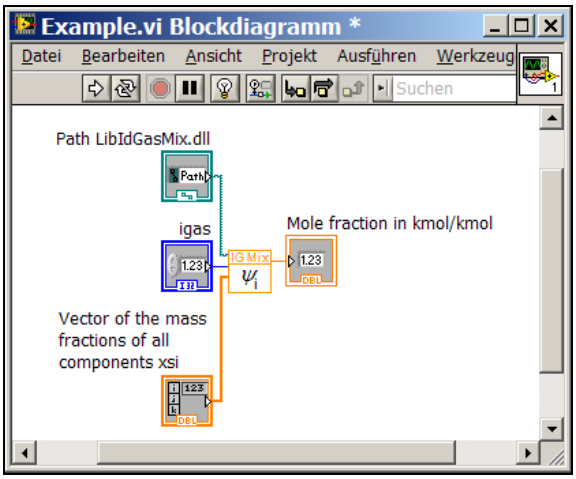


Figure 2.13
Block diagram of the example calculation

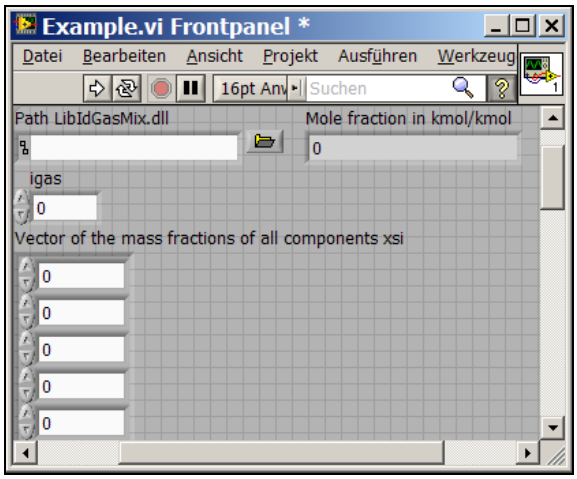


Figure 2.14
Front panel of the example calculation

- Enter the path of the LibIdGasMix.dll in the input element *Path LibIdGasMix.dll* on the front panel (as explained in section 2.1 the LibIdGasMix.dll and the other library files from the directory **<CD>\source** have to be stored in the same directory which is arbitrary). To do this you can use the *File Open Dialog* which appears by clicking the yellow folder symbol on the right of the input element.
- Enter the gas number for ethylene in the input element *igas* on the front panel

  ⇒ e. g.: Enter the gas number 16
- Enter the given mass fractions ξ_1 ... ξ_{30} of the mixture gases into the input element *Vector of composition comp* on the frontpanel. In doing so, please consider, that the vector *comp* consists of the elements 0 to 30. Whereas the element 0 is a dummy.

ξ_1	for argon	Ar	⇒ e.g.: Enter 0	into element 01
ξ_2	for neon	Ne	⇒ e.g.: Enter 0.12	into element 02
ξ_3	for nitrogen	N_2	⇒ e.g.: Enter 0	into element 03
ξ_4	for oxygen	O_2	⇒ e.g.: Enter 0	into element 04
ξ_5	for carbon monoxide	CO	⇒ e.g.: Enter 0	into element 05
ξ_6	for carbon dioxide	CO_2	⇒ e.g.: Enter 0	into element 06
ξ_7	for steam	H_2O	⇒ e.g.: Enter 0.09	into element 07
ξ_8	for sulfur dioxide	SO_2	⇒ e.g.: Enter 0	into element 08
ξ_9	for air - dry		⇒ e.g.: Enter 0.21	into element 09
ξ_{10}	for atmospheric nitrogen		⇒ e.g.: Enter 0	into element 10
ξ_{11}	for nitrogen oxide	NO	⇒ e.g.: Enter 0	into element 11
ξ_{12}	for hydrogen sulfide	H_2S	⇒ e.g.: Enter 0	into element 12
ξ_{13}	for hydroxyl	OH	⇒ e.g.: Enter 0	into element 13
ξ_{14}	for methanol	CH_3OH	⇒ e.g.: Enter 0	into element 14
ξ_{15}	for methane	CH_4	⇒ e.g.: Enter 0	into element 15
ξ_{16}	for ethylene	C_2H_4	⇒ e.g.: Enter 0.39	into element 16
ξ_{17}	for ethane	C_2H_6	⇒ e.g.: Enter 0	into element 17
ξ_{18}	for propylene	C_3H_6	⇒ e.g.: Enter 0	into element 18
ξ_{19}	for propane	C_3H_8	⇒ e.g.: Enter 0	into element 19
ξ_{20}	for n-butane	C_4H_{10}	⇒ e.g.: Enter 0.14	into element 20
ξ_{21}	for iso-butane	C_4H_{10}	⇒ e.g.: Enter 0	into element 21
ξ_{22}	for benzene	C_6H_6	⇒ e.g.: Enter 0	into element 22
ξ_{23}	for hydrogen	H_2	⇒ e.g.: Enter 0.05	into element 23
ξ_{24}	for helium	He	⇒ e.g.: Enter 0	into element 24
ξ_{25}	for ammonia	NH_3	⇒ e.g.: Enter 0	into element 25
ξ_{26}	free		⇒ e.g.: Enter 0	into element 26
ξ_{27}	free		⇒ e.g.: Enter 0	into element 27
ξ_{28}	free		⇒ e.g.: Enter 0	into element 28
ξ_{29}	free		⇒ e.g.: Enter 0	into element 29

ξ_{30} for fluorine F_2 ⇒ e.g.: Enter 0 into element 30

- To run the calculation of the the mole fraction click on the *Run* button or press <Ctrl>+<R>. The result for ψ in kmol/kmol appears in the output element (see Figure 2.12).

  ⇒ The result in our sample calculation here is: 0.2344039584 in kmol/kmol

  The calculation of ψ = f(igas, ξ_1 ... ξ_{30}) has thus been completed.

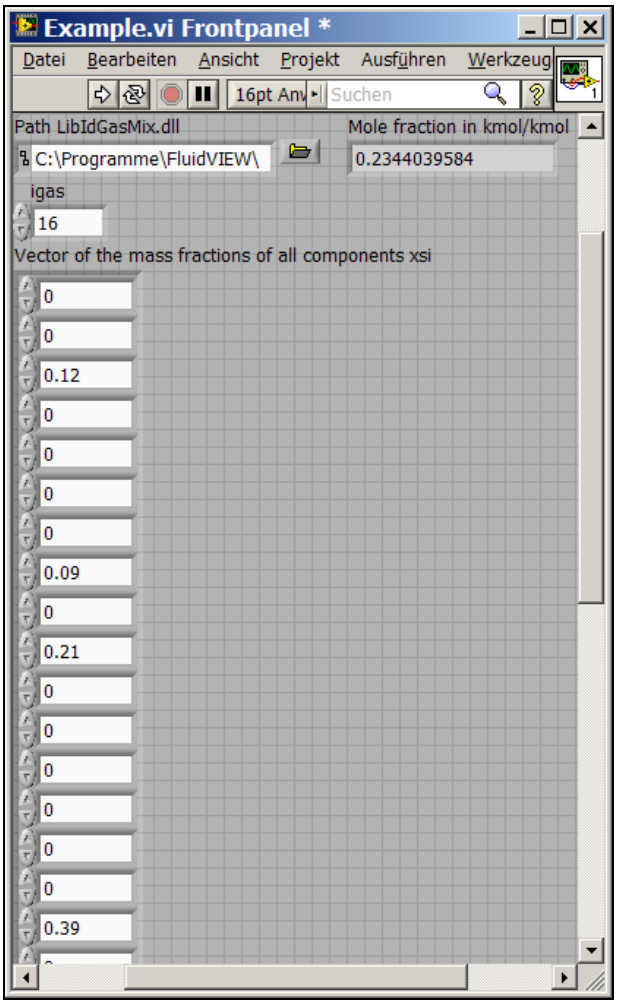


Figure 2.15 Result of the example calculation of ψ

Important!

LibIdGasMix checks internally whether the partial pressures of the mixture gases comprising the composition and the total pressure are lower than the saturation pressure at a given temperature. If this not be the case, the calculation will result in "-xx999", for which xx corresponds to the gas number, e.g., -7999 for steam.

2.6 Removing FluidVIEW

Should you wish to remove the LibIdGasMix library or the complete FluidVIEW Add-on you have to delete the files that have been copied in the default directory of the LabVIEW™ development environment **<LV>**.

Removing the FluidVIEW Add-on

To remove the FluidVIEW Add-on please delete the folders listed in Table 2.3 from the default directory of LabVIEW™.

Table 2.3 Directories that have to be deleted from the default directory of LabVIEW™ to remove the FluidVIEW Add-on

Name of the directory	Path in the default directory of LabVIEW™ (<LV>)
FluidVIEW	<LV>\vi.lib
FluidVIEW	<LV>\menus\Categories
FluidVIEW-Help	<LV>\help

Removing only the LibIdGasMix library

To remove only the LibIdGasMix library please delete the folders or files listed in Table 2.4 from the default directory of LabVIEW™.

Table 2.4 Data that have to be deleted from the default directory of LabVIEW™ (<LV>) to remove only the LibIdGasMix library.

File name with file extension or name of the directory	Path in the default directory of LabVIEW (<LV>)
LibIdGasMix.llb	<LV>\vi.lib\FluidVIEW
LibIdGasMix	<LV>\menus\Categories\FluidVIEW
LibIdGasMix.hlp	<LV>\help\FluidVIEW-Help
LibIdGasMix.txt	<LV>\help\FluidVIEW-Help
FluidVIEW_LibIdGasMix.pdf	<LV>\help\FluidVIEW-Help
Open_LibIdGasMix_doc.vi	<LV>\help\FluidVIEW-Help
Open_LibIdGasMix_doc.txt	<LV>\help\FluidVIEW-Help

The changes will take effect after restarting LabVIEW™.

3. Software Documentation for Gas Mixtures (igmix-Functions)

Thermal Diffusivity $a = f(p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name: **a_pt_igmix**

Subprogram with function value: **REAL*8 FUNCTION A_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:
REAL*8 P, T, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_A_PT_IGMIX(A,P,T,TYPE, COMP)**
For call from DLL:
REAL*8 A, P, T, COMP(0:30)
INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

a_pt_igmix – Thermal diffusivity *a* in m^2 / s

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C
Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

Thermal diffusivity $a = \frac{\lambda}{\rho \cdot c_p}$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Specific Isobaric Heat Capacity c_p = f($p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)

Function Name:	**cp_pt_igmix**
Subprogram with function value: For call from FORTRAN:	**REAL*8 FUNCTION CP_PT_IGMIX(P,T,TYPE, COMP)** REAL*8 P, T, COMP(0:30) INTEGER*4 TYPE
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_CP_PT_IGMIX(CP,P,T,TYPE, COMP)** REAL*8 CP, P, T, COMP(0:30) INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

cp_pt_igmix – Specific isobaric heat capacity c_p in kJ / (kg K)

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C

Exceptions:		
	Propylene	from -73.15 °C to 1500 °C
	Fluorine	from -73.15 °C to 976.8 °C

Details:

Model of the ideal mixture with in consideration of the dissociation effect above 500 °C and $\psi_{H_2O} \geq 0.1$

Results for wrong input values:

Error	Meaning
-9999	input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Specific Isochoric Heat Capacity c_V = f($p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)

Function Name: **cv_pt_igmix**

Subprogram with function value: **REAL*8 FUNCTION CV_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:
REAL*8 P, T, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_CV_PT_IGMIX(CV,P,T,TYPE, COMP)**
For call from DLL:
REAL*8 CV, P, T, COMP(0:30)
INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 ... \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 ... \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

cv_pt_igmix – Specific isochoric heat capacity c_V in kJ / (kg K)

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C

Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

$c_V = c_p - R$

Results for wrong input values:

Error	Meaning
-9999	input values beyond the range of validity and/or sum of the values entered $\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Dynamic Viscosity η = f($p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)

Function Name: **eta_pt_igmix**

Subprogram with function value: **REAL*8 FUNCTION ETA_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:
REAL*8 P, T, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_ETA_PT_IGMIX(ETA,P,T, TYPE, COMP)**
For call from DLL:
REAL*8 ETA, P, T, COMP(0:30)
INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

eta_pt_igmix – Dynamic viscosity η in Pa s

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C
Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

Calculation according to *Brandt* [15] and *VB* [33] – model of the ideal mixture

Results for wrong input values:

Error	Meaning
-9999	input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Specific Enthalpy $h = f(p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name: **h_pt_igmix**

Subprogram with function value: **REAL*8 FUNCTION H_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:
REAL*8 P, T, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_H_PT_IGMIX(H,P,T,TYPE, COMP)**
For call from DLL:
REAL*8 H, P, T, COMP(0:30)
INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

h_pt_igmix – Specific enthalpy *h* in kJ/kg

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C
Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

Model of the ideal mixture in consideration of the dissociation effect
above 500 °C and $\psi_{H_2O} \geq 0.1$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Isentropic Exponent κ = f($p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)

Function Name:	**kappa_pt_igmix**
Subprogram with function value:	**REAL*8 FUNCTION KAPPA_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:	REAL*8 P, T, COMP(0:30) INTEGER*4 TYPE
Subprogram with parameter:	**INTEGER*4 FUNCTION C_KAPPA_PT_IGMIX(KAPPA,P,T,TYPE, COMP)**
For call from DLL:	REAL*8 KAPPA, P, T, COMP(0:30) INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:

TYPE = 1 for the composition in mass fractions ξ

TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1

– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0

COMP(0) - dummy

COMP(1)...COMP(30) mass or mole fractions of the mixture gases (see table of Combustion Gases)

Result:

kappa_pt_igmix – Isentropic exponent κ

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C

Exceptions: Propylene from -73.15 °C to 1500 °C

Fluorine from -73.15 °C to 976.8 °C

Details:

Kappa $\kappa = \dfrac{c_p}{c_p - R}$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas no in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Thermal Conductivity λ = f($p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)

Function Name:	**lambda_pt_igmix**
Subprogram with function value:	**REAL*8 FUNCTION LAMBDA_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:	REAL*8 P, T, COMP(0:30) INTEGER*4 TYPE
Subprogram with parameter:	**INTEGER*4 FUNCTION C_LAMBDA_PT_IGMIX(LAMBDA,P,T,TYPE, COMP)**
For call from DLL:	REAL*8 LAMBDA, P, T, COMP(0:30) INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:

TYPE = 1 for the composition in mass fractions ξ

TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1

– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0

COMP(0) - dummy

COMP(1)...COMP(30) mass or mole fractions of the mixture gases (see table of Combustion Gases)

Result:

lambda_pt_igmix – Thermal conductivity λ in W/(m K)

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C

Exceptions: Propylene from -73.15 °C to 1500 °C

Fluorine from -73.15 °C to 976.8 °C

Details:

Calculation according to *Brandt* [15] and *VB* [33] – model of the ideal mixture

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Molar Mass M = f($\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)

Function Name:	**M_igmix**
Subprogram with function value: For call from FORTRAN:	**REAL*8 FUNCTION M_IGMIX(TYPE, COMP)** REAL*8 COMP(0:30) INTEGER*4 TYPE
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_M_IGMIX(M,TYPE, COMP)** REAL*8 M, COMP(0:30) INTEGER*4 TYPE

Input values:

TYPE - Type of composition:

TYPE = 1 for the composition in mass fractions ξ

TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1

– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0

COMP(0) - dummy

COMP(1)...COMP(30) mass or mole fractions of the mixture gases

(see table of Combustion Gases)

Result:

M_igmix - Molar mass *M* in kg/kmol

Details:

Calculation according to *Blanke*

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Kinematic Viscosity $\nu = f(p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name: **ny_pt_igmix**

Subprogram with function value: **REAL*8 FUNCTION NY_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN: REAL*8 P, T, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_NY_PT_IGMIX(NY,P,T,TYPE, COMP)**
For call from DLL: REAL*8 NY, P, T, COMP(0:30)
INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

ny_pt_igmix – Kinematic viscosity ν in m^2 / s

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C
Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

Kinematic viscosity

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Backward Function : $p = f(t, s, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name:	**p_ts_igmix**
Subprogram with function value: For call from FORTRAN:	**REAL*8 FUNCTION P_TS_IGMIX(T,S,TYPE, COMP)** REAL*8 P, T, COMP(0:30) INTEGER*4 TYPE
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_P_TS_IGMIX(P,T,S,TYPE, COMP)** REAL*8 P, T,S, COMP(0:30) INTEGER*4 TYPE

input values:

T - Temperature *t* in °C

S - Specific entropy *s* in kJ/(kg K)

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

p_ts_igmix - Total pressure *p* in bar

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from 200 K to 3026.85 °C

Exceptions:	Propylene	from -73.15 °C to 1500 °C
	Fluorine	from -73.15 °C to 976.8 °C

Details:

- Model of the ideal mixture in consideration of the dissociation effect above 500 °C
- Iteration of *p* from *s* = f(*p*, *t*, comp(0:30))

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Backward Function : $p = f(t, v, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name: **p_tv_igmix**

Subprogram with function value: **REAL*8 FUNCTION P_TV_IGMIX(T,V,TYPE, COMP)**
For call from FORTRAN:
REAL*8 T, V, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_P_TV_IGMIX(P,T,V,TYPE, COMP)**
For call from DLL:
REAL*8 P, P, V, COMP(0:30)
INTEGER*4 TYPE

Input values:

T - Temperature *t* in °C

V - Specific volume *v* in m^3/kg

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

p_tv_igmix - Total pressure *p* in bar

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from 200 K to 3026.85 °C
Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

$$p = \frac{R \cdot T}{v}$$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

PRANDTL - Number $Pr = f(p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name:	**Pr_pt_igmix**
Subprogram with function value:	**REAL*8 FUNCTION PR_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:	REAL*8 P, T, COMP(0:30) INTEGER*4 TYPE
Subprogram with parameter:	**INTEGER*4 FUNCTION C_PR_PT_IGMIX(PR,P,T,TYPE, COMP)**
For call from DLL:	REAL*8 PR, P, T, COMP(0:30) INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 ... \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 ... \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

Pr_pt_igmix – PRANDTL number Pr

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C

Exceptions:	Propylene	from -73.15 °C to 1500 °C
	Fluorine	from -73.15 °C to 976.8 °C

Details:

PRANDTL number

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Mole Fraction $\psi_i = f(i, \xi_1 \ldots \xi_{30})$

Function Name:	**psi_igas_xsi_igmix**
Subprogram with function value:	**REAL*8 FUNCTION PSI_IGAS_XSI_IGMIX(IGAS, COMP)**
For call from FORTRAN:	REAL*8 COMP(0:30) INTEGER*4 IGAS
Subprogram with parameter:	**INTEGER*4 FUNCTION C_PSI_PT_IGMIX(PSI,IGAS, COMP)**
For call from DLL:	REAL*8 PSI, COMP(0:30) INTEGER*4 IGAS

Input values:

IGAS - Gas number

COMP(0:30) - Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0

COMP(0) - dummy

COMP(1)...COMP(30) mass fractions of the mixture gases

(see table of Combustion Gases)

Result:

psi_igas_xsi_igmix – Mole fraction ψ_i in kmol/kmol

Range of validity:

gas number IGAS : from 1 to 30; gas numbers 26 to 29 are not occupied

Details:

Mole fraction : $\psi_i = \frac{R_i}{\sum(\xi_i \cdot R_i)} \cdot \xi_i$

Results for wrong input values:

Error	Meaning
-9999	sum of the values entered $\xi_1 \ldots \xi_{30} \neq 1$ gas numbers beyond the range of validity
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Specific Gas Constant R = f($\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)

Function Name:	**R_igmix**
Subprogram with function value:	**REAL*8 FUNCTION R_IGMIX(TYPE, COMP)**
For call from FORTRAN:	REAL*8 COMP(0:30) INTEGER*4 TYPE
Subprogram with parameter:	**INTEGER*4 FUNCTION C_R_IGMIX(R,TYPE, COMP)**
For call from DLL:	REAL*8 R, COMP(0:30) INTEGER*4 TYPE

Input values:

TYPE - Type of composition:

TYPE = 1 for the composition in mass fractions ξ

TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 ... \xi_{30}$ in kg / kg at TYPE = 1

– Composition in mole fractions $\psi_1 ... \psi_{30}$ in kmol / kmol at TYPE = 0

COMP(0) - dummy

COMP(1)...COMP(30) mass or mole fractions of the mixture gases

(see table of Combustion Gases)

Result:

R_igmix – Specific gas constant R in kJ/(kg K)

Details:

Specific gas constant : $R = \sum_i (\xi_i \cdot R_i)$ or $R = \frac{1}{\sum_i (\frac{\psi_i}{R_i})}$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30} \neq 1$
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Density ρ = f($p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)

Function Name:	**rho_pt_igmix**
Subprogram with function value:	**REAL*8 FUNCTION RHO_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:	REAL*8 P, T, COMP(0:30) INTEGER*4 TYPE
Subprogram with parameter:	**INTEGER*4 FUNCTION C_RHO_PT_IGMIX_SI(RHO,P,T,TYPE, COMP)**
For call from DLL:	REAL*8 RHO, P, T, COMP(0:30) INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:

TYPE = 1 for the composition in mass fractions ξ

TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1

– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0

COMP(0) - dummy

COMP(1)...COMP(30) mass or mole fractions of the mixture gases

(see table of Combustion Gases)

Result:

rho_pt_igmix – Density ρ in kg/m^3

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Calculation : $\rho = \dfrac{p}{R \cdot T}$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas numbers; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Specific Entropy $s = f(p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name: **s_pt_igmix**

Subprogram with function value: **REAL*8 FUNCTION S_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:
REAL*8 P, T, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_S_PT_IGMIX(S,P,T,TYPE, COMP)**
For call from DLL:
REAL*8 S, P, T, COMP(0:30)
INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

s_pt_igmix – Specific entropy *s* in kJ/(kg K)

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C
Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

Model of the ideal mixture in consideration of the dissociation effect
above 500 °C and $\psi_{H_2O} \geq 0.1$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Backward Function : Temperature $t = f(p, h, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name:	**t_ph_igmix**
Subprogram with function value: For call from FORTRAN:	**REAL*8 FUNCTION T_PH_IGMIX(P,H,TYPE, COMP)** REAL*8 P, T, COMP(0:30) INTEGER*4 TYPE
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_T_PH_IGMIX(T,P,H,TYPE, COMP)** REAL*8 T, P, H, COMP(0:30) INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

H - Specific enthalpy *h* in kJ/kg

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 ... \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 ... \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

t_ph_igmix – Temperature *t* in °C

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C

Exceptions:	Propylene	from -73.15 °C to 1500 °C
	Fluorine	from -73.15 °C to 976.8 °C

Details:

Iteration of *t* from *h* = f(*p*,*t*,comp(0:30))

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Backward Function : Temperature t = f($p, s, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30}$)

Function Name: **t_ps_igmix**

Subprogram with function value: **REAL*8 FUNCTION T_PS_IGMIX(P,S,TYPE, COMP)**
For call from FORTRAN:
REAL*8 P, T, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_T_PS_IGMIX(T,P,S,TYPE, COMP)**
For call from DLL:
REAL*8 T, P, S, COMP(0:30)
INTEGER*4 TYPE

Input values:

P - Total pressure p in bar

S - Specific entropy s in kJ/(kg K)

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 ... \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 ... \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

t_ps_igmix – Temperature t in °C

Range of validity:

Total pressure p : from 0.01 mbar to 50 bar

Temperature t : from -73.15 °C to 3026.85 °C
Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

Iteration of t from h = f(p,t,comp(0:30))

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Backward Function : Temperature $t = f(p, v, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name: **t_pv_igmix**

Subprogram with function value: **REAL*8 FUNCTION T_PV_IGMIX(P,V,TYPE, COMP)**
For call from FORTRAN:
REAL*8 P, T, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_T_PV_IGMIX(T,P,V,TYPE, COMP)**
For call from DLL:
REAL*8 T, P, V, COMP(0:30)
INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

V - Specific volume *v* in m^3/kg

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 ... \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 ... \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

t_pv_igmix – Temperature *t* in °C

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C
Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

Iteration of *t* from *h* = f(*p*,*t*,comp(0:30))

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of values entered $\xi_1 ... \xi_{30}$ or $\psi_1 ... \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Specific Internal Energy $u = f(p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name: **u_pt_igmix**

Subprogram with function value: **REAL*8 FUNCTION U_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:
REAL*8 P, T, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_U_PT_IGMIX(U,P,T,TYPE, COMP)**
For call from DLL:
REAL*8 U, P, T, COMP(0:30)
INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

u_pt_igmix – Specific internal energy *u* in kJ/kg

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C
Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

Specific internal energy *u* from *u* = *h*(*p*,*t*,comp(0:30)) - *R* * *T*

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Specific Volume $v = f(p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name: **v_pt_igmix**

Subprogram with function value: **REAL*8 FUNCTION V_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:
REAL*8 P, T, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_V_PT_IGMIX(V,P,T,TYPE, COMP)**
For call from DLL:
REAL*8 V, P, T, COMP(0:30)
INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

v_pt_igmix – Specific volume *v* in m^3/kg

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C
Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

Specific volume *v* from $v = \frac{R_m \cdot T}{p}$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Isentropic Speed of Sound $w = f(p, t, \xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30})$

Function Name: **w_pt_igmix**

Subprogram with function value: **REAL*8 FUNCTION W_PT_IGMIX(P,T,TYPE, COMP)**
For call from FORTRAN:
REAL*8 P, T, COMP(0:30)
INTEGER*4 TYPE

Subprogram with parameter: **INTEGER*4 FUNCTION C_W_PT_IGMIX(W,P,T,TYPE, COMP)**
For call from DLL:
REAL*8 W, P, T, COMP(0:30)
INTEGER*4 TYPE

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

TYPE - Type of composition:
TYPE = 1 for the composition in mass fractions ξ
TYPE = 0 for the composition in mole fractions ψ

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1
– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0
COMP(0) - dummy
COMP(1)...COMP(30) mass or mole fractions of the mixture gases
(see table of Combustion Gases)

Result:

w_pt_igmix – Isentropic speed of sound *w* in m/s

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C
Exceptions: Propylene from -73.15 °C to 1500 °C
Fluorine from -73.15 °C to 976.8 °C

Details:

Isentropic speed of sound *w* from : $w = \sqrt{\frac{R_m \cdot T \cdot c_p}{c_p - R_m}}$

$c_p = f(p, t, \text{comp}(0:30))$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity and/or sum of the values entered $\xi_1 \ldots \xi_{30}$ or $\psi_1 \ldots \psi_{30} \neq 1$
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

Mass Fraction $\xi_i = f(i, \psi_1 \ldots \psi_{30})$

Function Name:	**xsi_igas_psi_igmix**
Subprogram with function value:	**REAL*8 FUNCTION XSI_IGAS_PSI_IGMIX(IGAS, COMP)**
For call from FORTRAN:	REAL*8 COMP(0:30) INTEGER*4 IGAS
Subprogram with parameter:	**INTEGER*4 FUNCTION C_XSI_IGAS_PSI_IGMIX(XSI,IGAS, COMP)**
For call from DLL:	REAL*8 XSI, COMP(0:30) INTEGER*4 IGAS

Input values:

IGAS - Gas number

COMP(0:30) – Composition in mass fractions $\xi_1 \ldots \xi_{30}$ in kg / kg at TYPE = 1

– Composition in mole fractions $\psi_1 \ldots \psi_{30}$ in kmol / kmol at TYPE = 0

COMP(0) - dummy

COMP(1)...COMP(30) mass or mole fractions of the mixture gases (see table of Combustion Gases)

TYPE = 1 for the composition in mass fractions ξ

TYPE = 0 for the composition in mole fractions ψ

Result:

xsi_igas_psi_igmix – Mass fraction ξ_i in kg/kg

Range of validity:

Gas number IGAS: from 1 to 30; gas numbers 26 to 29 are not occupied

Results for wrong input values:

Error	Meaning
-9999	sum of the values entered $\psi_1 \ldots \psi_{30} \neq 1$ 1 gas number beyond the range of validity
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied
-30777	Fluorine (F2) as a mixture gas

4. Program Documentation for Single Gases (igas-Functions)

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

Function Name: **a_pt_igas**

Subprogram with function value: **REAL*8 FUNCTION A_PT_IGAS(P,T,IGAS)**
For call from FORTRAN:
REAL*8 P, T
INTEGER*4 IGAS

Subprogram with parameter: **INTEGER*4 FUNCTION C_A_PT_IGAS(A,P,T,IGAS)**
For call from DLL:
REAL*8 A, P, T
INTEGER*4 IGAS

Input values:

P - Total pressure *p* in bar

T - Temperature *t* in °C

IGAS - Gas number (see table of Combustion Gases)

Result:

a_pt_igas – Thermal diffusivity *a* in m^2/s

Range of validity:

Total pressure *p* : from 0.01 mbar to 50 bar

Temperature *t* : from -73.15 °C to 3026.85 °C

Exceptions: Propylene from -73.15 °C to 1500 °C

Fluorine from -73.15 °C to 976.8 °C

Details:

Thermal diffusivity $a = \frac{\lambda}{\rho \cdot c_p}$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; Gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

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

Function Name:	**cp_pt_igas**
Subprogram with function value: For call from FORTRAN:	**REAL*8 FUNCTION CP_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_CP_PT_IGAS(CP,P,T,IGAS)** REAL*8 CP, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure *p* in bar
T	- Temperature *t* in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

cp_pt_igas – Specific isobaric heat capacity c_p in kJ/(kg K)

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Model of the ideal mixture in consideration of the dissociation effect above 500 °C and $\psi_{H_2O} \geq 0.1$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

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

Function Name:	**cv_pt_igas**
Subprogram with function value: For call from FORTRAN:	**REAL*8 FUNCTION CV_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_CV_PT_IGAS(CV,P,T,IGAS)** REAL*8 CV, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure p in bar
T	- Temperature t in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

cv_pt_igas – Specific isochoric heat capacity c_v in kJ / (kg K)

Range of validity:

Total pressure p :	from 0.01 mbar to 50 bar		
Temperature t :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

$c_v = c_p - R$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Dynamic Viscosity η = f(p,t,igas)

Function Name:	**eta_pt_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION ETA_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_ETA_PT_IGAS(ETA,P,T,IGAS)** REAL*8 ETA, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure *p* in bar
T	- Temperature *t* in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

eta_pt_igas – Dynamic viscosity η in Pa s

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Calculation according to *Brandt*[15] and *VB* [33] – model of the ideal mixture.

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Specific Enthalpy *h* = f(*p*, *t*, igas)

Function Name:	**h_pt_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION H_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_H_PT_IGAS(H,P,T,IGAS)** REAL*8 H, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure *p* in bar
T	- Temperature *t* in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

h_pt_igas – Specific enthalpy *h* in kJ/kg

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Model of the ideal mixture in consideration of the dissociation effect

above 500 °C and $\psi_{H_2O} \geq 0.1$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Isentropic Exponent κ = f(p,t,igas)

Function Name:	**kappa_pt_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION KAPPA_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_KAPPA_PT_IGAS(KAPPA,P,T,IGAS)** REAL*8 KAPPA, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure p in bar
T	- Temperature t in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

kappa_pt_igas – Isentropic exponent κ

Range of validity:

Total pressure p :	from 0.01 mbar to 50 bar		
Temperature t :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Kappa $\kappa = \dfrac{c_p}{c_p - R}$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Thermal Conductivity λ = f(p,t,igas)

Function Name:	**lambda_pt_igas**
Subprogram with function value:	**REAL*8 FUNCTION LAMBDA_PT_IGAS(P,T,IGAS)**
For the call out of FORTRAN	REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter:	**INTEGER*4 FUNCTION C_LAMBDA_PT_IGAS (LAMBDA,P,T,IGAS)**
For call from DLL:	REAL*8 LAMBDA, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure *p* in bar
T	- Temperature *t* in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

lambda_pt_igas – Isentropic exponent λ in W/(m K)

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Calculation according to *Brandt*[15] and *VB* [33] – model of the ideal mixture.

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Molar Mass *M* = f(*p*, *t*, igas)

Function Name:	**M_igas**
Subprogram with function value: For call from FORTRAN:	**REAL*8 FUNCTION M_IGAS(IGAS)** INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_M_IGAS(M,IGAS)** REAL*8 M INTEGER*4 IGAS

Input values:

IGAS - Gas number (see table of Combustion Gases)

Result:

M_igas – Molar mass *M* in kg/kmol

Details:

Molar mass

Results for wrong input values:

Error	Meaning
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Kinematic Viscosity ν = f(p,t,igas)

Function Name:	**ny_pt_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION NY_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_NY_PT_IGAS(NY,P,T,IGAS)** REAL*8 NY, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure p in bar
T	- Temperature t in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

ny_pt_igas – Kinematic viscosity ν in m^2/s

Range of validity:

Total pressure p :	from 0.01 mbar to 50 bar		
Temperature t :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Kinematic viscosity

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

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

Function Name:	**p_ts_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION P_TS_IGAS_SI(T,S,IGAS)** REAL*8 T,S INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_P_TS_IGAS(P,T,S,IGAS)** REAL*8 P, T,S INTEGER*4 IGAS

Input values:

T	- Temperature *t* in °C
S	- Specific entropy *s* in kJ/(kg K)
IGAS	- Gas number (see table of Combustion Gases)

Result:

p_ts_igas - Total pressure *p* in bar

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

- Model of the ideal mixture in consideration of the dissociation effect above 500 °C
- Iteration of *p* from *s* = f(*p*,*t*)

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-7777	Result beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Backward Function: Pressure *p* = f(*t*, *v*, igas)

Function Name:	**p_tv_igas**
Subprogram with function value:	**REAL*8 FUNCTION P_TV_IGAS(T,V,IGAS)**
For call from FORTRAN:	REAL*8 T, V INTEGER*4 IGAS
Subprogram with parameter:	**INTEGER*4 FUNCTION C_P_TV_IGAS(P,T,V,IGAS)**
For call from DLL:	REAL*8 P, P, V INTEGER*4 IGAS

Input values:

T	- Temperature *t* in °C
V	- Specific volume *v* in m^3/kg
IGAS	- Gas number (see table of Combustion Gases)

Result:

p_ts_igas - Total pressure *p* in bar

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

$$p = \frac{R \cdot T}{v}$$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-7777	Result beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

PRANDTL Number *Pr* = f(*p*, *t*, igas)

Function Name:	**Pr_pt_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION PR_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_PR_PT_IGAS(PR,P,T,IGAS)** REAL*8 PR, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure *p* in bar
T	- Temperature *t* in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

Pr_pt_igas - PRANDTL number *Pr*

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

PRANDTL number

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Specific Gas Constant *R* = f(igas)

Function Name:	**R_igas**
Subprogram with function value: For call from FORTRAN	**REAL*8 FUNCTION R_IGAS(IGAS)** INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_R_IGAS(R,IGAS)** REAL*8 R INTEGER*4 IGAS

Input values:

IGAS - Gas number (see table of Combustion Gases)

Result:

R_igas - Specific gas constant *R* in kJ/(kg K)

Details:

Specific gas constant: $R = \sum_i (\xi_i \cdot R_i)$ or $R = \frac{1}{\sum_i (\frac{\psi_i}{R_i})}$

Results for wrong input values:

Error	Meaning
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Density ρ = f(p,t,igas)

Function Name:	**rho_pt_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION RHO_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_RHO_PT_IGAS(RHO,P,T,IGAS)** REAL*8 RHO, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure p in bar
T	- Temperature t in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

rho_pt_igas – Density ρ in kg/m^3

Range of validity:

Total pressure p :	from 0.01 mbar to 50 bar		
Temperature t :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Calculation : $\rho = \frac{p}{R \cdot T}$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Specific Entropy *s* = f(*p*, *t*, igas)

Function Name:	**s_pt_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION S_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_S_PT_IGAS(S,P,T,IGAS)** REAL*8 S, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure *p* in bar
T	- Temperature *t* in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

s_pt_igas – Specific entropy *s* in kJ/(kg K)

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Model of the ideal mixture in consideration of the dissociation effect

above 500 °C and $\psi_{H_2O} \geq 0.1$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

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

Function Name:	**t_ph_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION T_PH_IGAS(P,H,IGAS)** REAL*8 P,H INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_T_PH_IGAS(T,P,H,IGAS)** REAL*8 T, P, H INTEGER*4 IGAS

Input values:

P	- Total pressure *p* in bar
H	- Specific enthalpy *h* in kJ/kg
IGAS	- Gas number (see table of Combustion Gases)

Result:

t_ph_igas - Temperature *t* in °C

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

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

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-7777	Result beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

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

Function Name:	**t_ps_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION T_PS_IGAS(P,S,IGAS)** REAL*8 P,S INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_T_PS_IGAS(T,P,S,IGAS)** REAL*8 T, P, S INTEGER*4 IGAS

Input values:

P	- Total pressure *p* in bar
S	- Specific entropy *s* in kJ/(kg K)
IGAS	- Gas number (see table of Combustion Gases)

Result:

t_ps_igas - Temperature *t* in °C

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

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

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-7777	Result beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Backward Function: Temperature *t* = f(*p*, *v*, igas)

Function Name:	**t_pv_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION T_PV_IGAS(P,V,IGAS)** REAL*8 P,V INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_T_PV_IGAS(T,P,V,IGAS)** REAL*8 T, P, V INTEGER*4 IGAS

Input values:

P	- Total pressure *p* in bar
V	- Specific volume *v* in m^3/kg
IGAS	- Gas number (see table of Combustion Gases)

Result:

t_pv_igas - Temperature *t* in °C

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

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

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-7777	Result beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Specific Internal Energy *u* = f(*p*, *t*, igas)

Function Name:	**u_pt_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION U_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_U_PT_IGAS(U,P,T,IGAS)** REAL*8 U, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure *p* in bar
T	- Temperature *t* in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

u_pt_igas – Specific internal energy *u* in kJ/kg

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Specific internal energy *u* from: $u = h(p,t) - R * T$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Specific Volume v = f(p, t, igas)

Function Name:	**v_pt_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION V_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_V_PT_IGAS(V,P,T,IGAS)** REAL*8 V, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure p in bar
T	- Temperature t in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

v_pt_igas – Specific volume v in m^3/kg

Range of validity:

Total pressure p :	from 0.01 mbar to 50 bar		
Temperature t :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Specific volume v from : $v = \frac{R_m \cdot T}{p}$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

Isentropic Speed of Sound *w* = f(*p*, *t*, igas)

Function Name:	**w_pt_igas**
Subprogram with function value: For the call out of FORTRAN	**REAL*8 FUNCTION W_PT_IGAS(P,T,IGAS)** REAL*8 P,T INTEGER*4 IGAS
Subprogram with parameter: For call from DLL:	**INTEGER*4 FUNCTION C_W_PT_IGAS(V,P,T,IGAS)** REAL*f8 W, P,T INTEGER*4 IGAS

Input values:

P	- Total pressure *p* in bar
T	- Temperature *t* in °C
IGAS	- Gas number (see table of Combustion Gases)

Result:

w_pt_igas – Isentropic speed of sound *w* in m/s

Range of validity:

Total pressure *p* :	from 0.01 mbar to 50 bar		
Temperature *t* :	from -73.15 °C to 3026.85 °C		
	Exceptions:	Propylene	from -73.15 °C to 1500 °C
		Fluorine	from -73.15 °C to 976.8 °C

Details:

Isentropic speed of sound *w* from :

$$w = \sqrt{\frac{R_m \cdot T \cdot c_p}{c_p - R_m}}$$

$$c_p = f(p,t)$$

Results for wrong input values:

Error	Meaning
-9999	Input values beyond the range of validity
-xx999	xx…gas number; gas not in the gaseous state
-xx0999	xx…gas number; gas; gas numbers 26 to 29 not occupied

5. 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
  - IAPWS-IF97-S03rev
  - IAPWS-IF97-S04
  - IAPWS-IF97-S05
- IAPWS Revised Advisory Note No. 3 on Thermodynamic Derivatives (2008)

Humid Combustion Gas Mixtures

Library LibHuGas

Ideal mixture of the real fluids:

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

and of the ideal gases:

SO_2, CO, Ne (scientific equations of Bücker et al.)
Consideration of
Dissociation from VDI 4670 and Poynting effect

Library LibIDGAS

Ideal gas mixture calculated from the VDI-Guideline 4670

Humid Air

Library LibHuAir

Ideal mixture of the real fluids:

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

Consideration of

- Dissociation from VDI-Guideline 4670
- Poynting effect

Library LibIdAir

Ideal gas mixture calculated from VDI-Guideline 4670

Carbon Dioxide

Library LibCO2

Formulation of Span and Wagner (1994)

Hydrogen

Library LibH2

Formulation of Leachman et al. (2007)

Helium

Library LibHe

Formulation of McCarty and Arp (1990)

Methanol

Library LibCH3OH

Formulation of de Reuck and Craven (1993)

Ideal Gas Mixtures

Library LibIdGasMix

Ideal mixture of the ideal gases:

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

Consideration of

- Dissociation from VDI-Guideline 4670

Seawater

Library LibSeaWa

IAPWS Formulation (2008) and IAPWS-IF97

Refrigerant R134a

Library LibR134a

Formulation of Tillner-Roth and Baehr (1994)

Refrigerant NH_3

Library LibNH3

Formulation of Tillner-Roth (1995)

ORC Working Fluids

Library LibMM

Siloxane $C_6H_{18}OSi_2$ (MM)

Formulation of Colonna et al. (2006)

Library LibD4

Siloxane $C_8H_{24}O_4Si_4$ (D4)

Formulation of Colonna et al. (2006)

Library LibD5

Siloxane $C_{10}H_{30}O_5Si_5$ (D5)

Formulation of Colonna et al. (2006)

Library LibMD4M

Siloxane $C_{14}H_{42}O_5Si_6$ (MD4M)

Formulation of Colonna et al. (2006)

Mixtures for Absorption Processes

Library LibAmWa

Ammonia/Water Mixtures

IAPWS Guideline 2005 of Tillner-Roth and Friend (1998)

Helmholtz energy formulation for the mixing term

Library LibWaLi

Water/Lithium Bromide Mixtures

Formulation of Kim and Infante Ferreira (2004)

Gibbs energy equation for the mixing term

Refrigerants

Library LibPropan

Refrigerant Propane

Formulation of Lemmon et al. (2008)

Library LibButan_Iso

Refrigerant Iso-Butane

Formulation of Bücker et al. (2003)

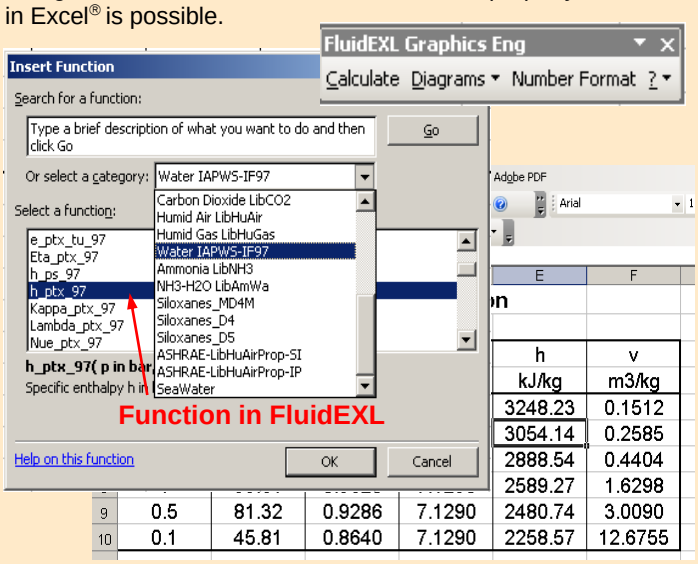
Library LibButan_n

Refrigerant n-Butane

Formulation of Bücker et al. (2003)

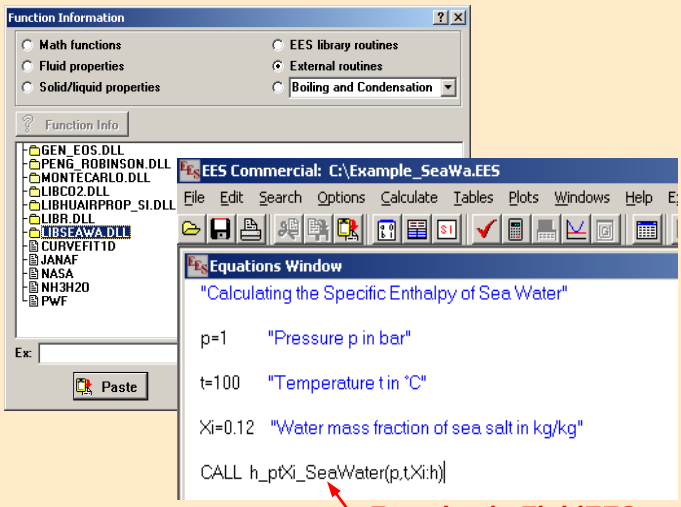
Add-In FluidEXL for Excel®

Using the Add-In FluidEXL a direct call of the property functions in Excel® is possible.



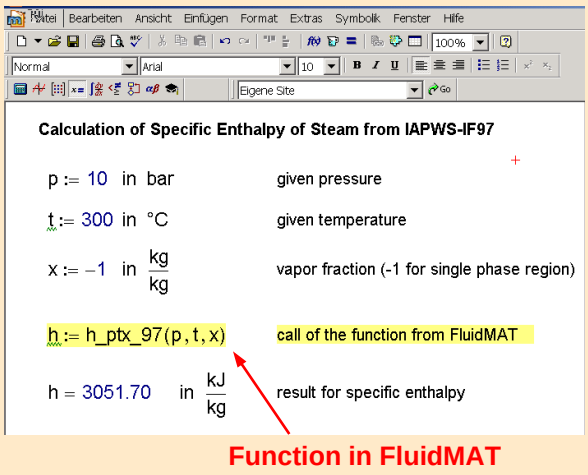
Add-In FluidEES for EES®

The Add-In FluidEES allows to call the functions of the property libraries within the Engineering Equation Solver EES®.



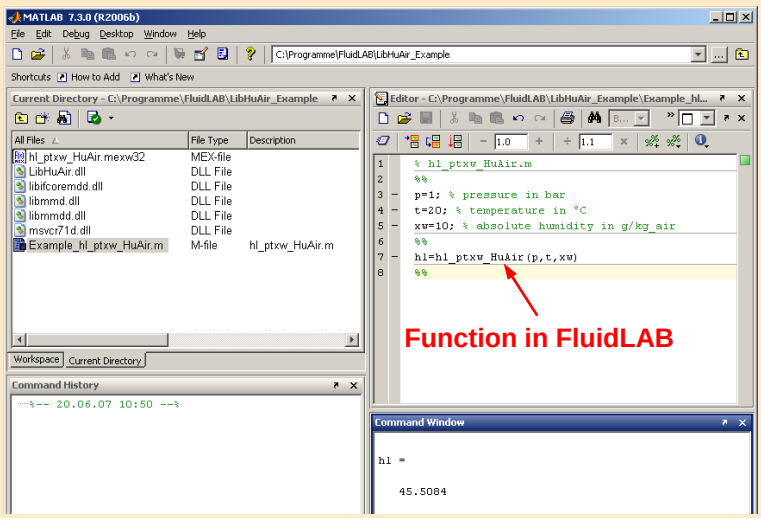
Add-In FluidMAT for Mathcad®

Using the Add-on FluidMAT, the functions of the property libraries can be used in Mathcad®.



Add-In FluidLAB for MATLAB®

The property functions can be called in MATLAB®.



The following thermodynamic and transport properties can be calculated[1]:

Thermodynamic Properties

- Saturation 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

Thermodynamic Derivatives

- Partial derivatives can be calculated

Transport Properties

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

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)

For information please contact:

Zittau/Goerlitz University of Applied Sciences
Department of Technical Thermodynamics
Prof. Hans-Joachim Kretzschmar
Dr. Ines Stoecker
Theodor-Koerner-Allee 16
02763 Zittau, Germany

E-mail: hj.kretzschmar@hs-zigr.de
Internet: www.thermodynamics-zittau.de
Phone: +49-3583-61-1846
Fax.: +49-3583-61-1846

[1] Not all of these property functions are available in all property libraries listed before.

6. References

[1] IAPWS Secretariat, Dooley, B., EPRI, Palo Alto CA (1997):
Release on the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam IAPWS-IF97.

[2] Wagner, W.; Cooper, J.R.; Dittmann, A.; Kijima, J.; Kretzschmar, H.-J.; Kruse, A.; Mareš, R.; Oguchi, K.; Sato, H.; Stöcker, I.; Šifner, O.; Takaishi, Y.; Tanishita, I.; Trübenbach, J.; Willkommen, Th. (2000):
The IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam.
ASME Journal of Engineering for Gas Turbines and Power 122, No. 1, pp. 150-182.

[3] Wagner, W.; Kruse, A. (1998):
Properties of Water and Steam.
Springer-Verlag, Berlin

[4] Kretzschmar, H.-J.; Stöcker, I.; Klinger, J.; Dittmann, A. (2000):
Calculation of Thermodynamic Derivatives for Water and Steam Using the New Industrial Formulation IAPWS-IF97.
in: *Steam, Water and Hydrothermal Systems: Physics and Chemistry Meeting the Needs of Industry, Proceedings of the 13th International Conference on the Properties of Water and Steam*. Eds. P.G. Hill et al., NRC Press, Ottawa

[5] Kretzschmar, H.-J. (1998):
Mollier h-s Diagram.
Springer-Verlag, Berlin

[6] IAPWS Secretariat, Dooley, B., EPRI, Palo Alto CA, (1997):
Revised Release on the IAPS Formulation 1985 for the Thermal Conductivity of Ordinary Water Substance.

[7] IAPWS Secretariat, Dooley, B., EPRI, Palo Alto CA, (1997):
Revised Release on the IAPS Formulation 1985 for the Viscosity of Ordinary Water Substance.

[8] IAPWS Secretariat, Dooley, B., EPRI, Palo Alto CA, (1994):
I*APWS Release on Surface Tension of Ordinary Water Substance 1994.*

[9] Kretzschmar, H.-J.; Stöcker, I.; Willkommen, Th.; Trübenbach, J.; Dittmann, A. (2000):
Supplementary Equations $v(p,T)$ for the Critical Region to the New Industrial Formulation IAPWS-IF97 for Water and Steam.
in: *Steam, Water and Hydrothermal Systems: Physics and Chemistry Meeting the Needs of Industry, Proceedings of the 13th International Conference on the Properties of Water and Steam*. Eds. P.G. Hill et al., NRC Press, Ottawa.

[10] Kretzschmar, H.-J.; Stöcker, I.; Knobloch, K.; Trübenbach, J.; Willkommen, Th.:
Dittmann, A.; Friend, D.G.:
Supplementary Backward Equations for Pressure as a Function of Enthalpy and Entropy p(h,s) to the Industrial Formulation IAPWS-IF97 for Water and Steam.
ASME Journal of Engineering for Gas Turbines and Power - in preparation

[11] IAPWS Secretariat, Dooley, B., EPRI, Palo Alto CA, (1995):
Release on the IAPWS Formulation 1995 for the Thermodynamic Properties of Ordinary Water Substance for General and Scientific Use.

[12] Grigull, U. (1989):
Properties of Water and Steam in SI Units.
Springer-Verlag, Berlin.

[13] Kretzschmar, H.-J. (1990):
Zur Aufbereitung und Darbietung thermophysikalischer Stoffdaten für die Energietechnik (Preparation of Thermophysical Properties for Power Engineering).
Habilitation. Dresden University of Technology, Faculty of Mechanical Engineering, Germany

[14] Baehr, H.D.; Diederichsen, Ch. (1988):
Berechnungsgleichungen für Enthalpie und Entropie der Komponenten von Luft und Verbrennungsgasen (Equations for the Enthalpy and Entropy of Components of Air and Combustion Gases).
BWK 40 (1988), no. 1/2, pp. 30-33.

[15] Brandt, F. (1985):
Wärmeübertragung in Dampferzeugern und Wärmetauschern (Heat Transfer in Steam Generators and Heat Exchangers).
FDBR-Fachbuchreihe, Bd. 2, Vulkan Verlag, Essen.

[16] Bücker, D.; Span, R.; Wagner, W. (2002):
Thermodynamic Property Models of Moist Air Flue Gases.
J. Phys. Chem. Ref. Data 29 – submitted.

[17] Reid, R. C.; Prausnitz, J. M.; Poling, B. E. (1987):
The Properties of Gases and Liquids.
4th Edition, McGraw-Hill Book Company, New York.

[18] IAPWS Secretariat, Dooley, B., EPRI, Palo Alto CA, (1975):
Release on Surface Tension of Ordinary Water Substance 1975.

[19] VDI Wärmeatlas, 1995, 7th Edition
VDI-Verlag, Düsseldorf.

[20] Blanke, W. (1989):
Thermophysikalische Stoffgrößen (Thermophysical Properties).
Springer-Verlag, Berlin.

[21] *VDI-Richtlinie 4670*
Thermodynamische Stoffwerte von feuchter Luft und Verbrennungsgasen (VDI 4670 Guideline – Thermodynamic Properties of Humid Air and Combustion Gases).
VDI Manual for Energy Technology, VDI Society for Energy Technology, Düsseldorf (2000).

[22] Lemmon, E. W.; Jacobsen, R. T.; Penoncello, S. G.; Friend, D. G. (2000):
Thermodynamic Properties of Air and Mixtures of Nitrogen, Argon and Oxygen from 60 to 2000 K at Pressures to 2000 MPa.
J. Phys. Chem. Ref. Data 29. Nr. 2, S. 331-385

[23] Lemmon, E. W.; Jacobsen, R. T. (2000):
Transport Properties of Air.
National Institute of Standards and Technology, Boulder CO. Private communication.

[24] IAPWS Secretariat, Dooley, B, EPRI, Alto CA (1992):
Revised Supplementary Release on Saturation Properties of Ordinary Water Substance.

[25] Hyland, R. W.; Wexler, A. (1983):
Formulations for the Thermodynamic Properties of Saturated Phases of H_2O from 173.15 K to 473.15 K.
Report No. 2793 (RP-216), National Bureau of Standards, Washington, D.C.

[26] de Reuck, K. M.; Craven, R. J. B. (1993):
Methanol. International Thermodynamic Tables of the Fluid State - 12.
Blackwell Scientific Publications, London.

[27] Wagner, W.; de Reuck, K. M. (1996):
Methane. International Thermodynamic Tables of the Fluid State - 13.
Blackwell Scientific Publications, London.

[28] de Reuck, K. M. (1990):
Fluorine. International Thermodynamic Tables of the Fluid State - 11.
Blackwell Scientific Publications, London.

[29] Buecker, D. (2003):
Neue Fundamentalgleichungen als Referenz für die thermodynamischen Zustandsgrößen von Ethan, n-Butan und Isobutan.
Fortschr.-Ber. VDI, Reihe 6, Nr. 499, VDI Verlag, Düsseldorf.

[30] Vargaftik, N. B.; Vinogradov, Y. K.; Yagarin, V. S. (1996):
Handbook of Physical Properties of Liquids and Gases.
Revised Edition. Begell House, New York.

[31] Span, R. (2000):
Multiparameter Equations of State. An Accurate Source of Thermodynamic Property Data.
Springer-Verlag, Berlin.

[32] Angus, S.; Amstrong, B.; de Reuck, K. M. (1978):
Final Report on IUPAC Propylene (Propene) Tables, Part 1: Text, International Thermodynamic Tables of the Fluid State – 7.
Imperial College London, IUPAC Thermodynamic Tables Project Centre.

[33] *Verfahrenstechnische Berechnungsmethoden, Teil 7: Stoffwerte (Process-Related Calulation Techniques, Part 7: Properties).*
Deutscher Verlag für Grundstoffindustrie, Leipzig (1981)

[34] *Brennstofftechnische Arbeitsmappe (Combustion-Related Working Paper).*
Bergakademie Freiberg (1988)

[35] Smukala, J.; Span, R.; Wagner, W. (2000):
New Equation for Ethylene Covering The Fluid Region for Temperatures From the Melting Line to 450 K at Pressures up to 300 MPa.
Journal of Physical and Chemical Reference Data, Vol. 29, No. 5, pp. 1053-1121.

[36] Span, R.; Wagner, W. (1996):
A New Equation of State for Carbon Dioxide Covering the Fluid Region from the Triple-Point Temperature to 1100K at Pressures up to 800 MPa.
Journal of Physical and Chemical Reference Data, Vol. 25, No.6, pp. 1509-1596.

[37] Younglove, B. A. (1982):
Thermophysical Properties of Fluids. I. Argon, Ethylene, Parahydrogen, Nitrogen, Nitrogen Trifluoride, and Oxygen.
Journal of Physical and Chemical Reference Data, Vol. 11, Supplement No.1.

[38] Baehr, H. D.; Tillner-Roth, R. (1995):
Thermodynamische Eigenschaften umweltverträglicher Kältemittel (Thermodynamic Properties of Environmentally Acceptable Refrigerants).
Springer-Verlag, Berlin

[39] Wagner, W.; Saul, A.; Pruss, A. (1994):
International Equations for the Pressure along the Melting And along the Sublimation Curve of Ordinary Water Substance.
Journal of Physical and Chemical Reference Data. Vol.23, No. 3, pp. 515-527.

[40] ASME Standards Technology, LLC:
Thermophysical Properties of Working Gases used in Turbine Applications.
STP-TS-012 (2008)

[41] Lemmon, E.W.; Span, R.:
Short Fundamental Equations of State for 20 Industrial Fluids.
J. Chem. Eng. Data, 51:785-850, 2006

[42] Overhoff, U.:
Development of a new equation of state for the fluid region of propene for temperatures from the melting line to 575 K with pressures to 1000 MPa as well as software for the computation of thermodynamic properties of fluids.
Ph.D. Dissertation, Ruhr University, Bochum, Germany, 2006

[43] Lemmon, E.W.; McLinden, M.O.; Wagner, W.:
Thermodynamic Properties of Propane. IV. A Reference Equation of State for Temperatures from the Melting Line to 650 K and Pressures up to 1000 MPa.
to be submitted to J. Phys. Chem. Ref. Data, 2009

[44] Polt, A.; Platzer, B.; Maurer, G.:
Parameter der thermischen Zustandsgleichung von Bender fuer 14 mehratomige reine Stoffe.
Chem. Tech. (Leipzig), 44(6):216-224, 1992

[45] Leachman, J.W.; Jacobsen, R.T.; Lemmon, E.W.:
Fundamental Equations of State for Parahydrogen, Normal Hydrogen, and Orthohydrogen.
to be published in the International Journal of Thermophysics, 2007

[46] Kunz, O.; Klimeck, R.; Wagner, W.; Jaeschke, M.:
The GERG-2004 Wide-Range Reference Equation of State for Natural Gases and Other Mixtures.
Fortschr.-Ber. VDI, VDI-Verlag, Düsseldorf, 2006

7. Satisfied Customers

Date: 10/2011

The following companies and institutions use the property libraries

- FluidEXL*Graphics* for Excel®
- FluidLAB for MATLAB®
- FluidMAT for Mathcad®
- FluidEES for Engineering Equation Solver® EES
- FluidDYM for Dymola® (Modelica)
- FluidVIEW for LabVIEW®:

2011

Lopez, Munguia, Spain	10/2011
University of KwaZulu-Natal, Westville, South Africa	10/2011
Voith, Heidenheim	09/2011
SpgBe Montreal, Canada	09/2011
SPG TECH, Montreuil Cedex, France	09/2011
Voith, Heidenheim-Mergelstetten	09/2011
MTU Aero Engines, Munich	08/2011
MIBRAG, Zeitz	08/2011
RWE, Essen	07/2011
Fels, Elingerode	07/2011
Weihenstephan University of Applied Sciences	07/2011, 09/2011, 10/2011
Forschungszentrum Juelich	07/2011
RWTH Aachen University	07/2011, 08/2011
INNEO Solutions, Ellwangen	06/2011
Caliqua, Basel, Switzerland	06/2011
Technical University of Freiberg	06/2011
Fichtner IT Consulting, Stuttgart	05/2011, 06/2011, 08/2011
Salzgitter Flachstahl, Salzgitter	05/2011
Helbling Beratung & Bauplanung, Zurich, Switzerland	05/2011
INEOS, Cologne	04/2011
Enseleit Consulting Engineers, Siebigerode	04/2011
Witt Consulting Engineers, Stade	03/2011

Helbling, Zurich, Switzerland	03/2011
MAN Diesel, Copenhagen, Denmark	03/2011
AGO, Kulmbach	03/2011
University of Duisburg	03/2011, 06/2011
CCP, Marburg	03/2011
BASF, Ludwigshafen	02/2011
ALSTOM Power, Baden, Switzerland	02/2011
Universität der Bundeswehr, Munich	02/2011
Calorifer, Elgg, Switzerland	01/2011
STRABAG, Vienna, Austria	01/2011
TUEV Sued, Munich	01/2011
ILK Dresden	01/2011
Technical University of Dresden	01/2011, 05/2011, 06/2011, 08/2011

2010

Umweltinstitut Neumarkt	12/2010
YIT Austria, Vienna, Austria	12/2010
MCI Innsbruck, Austria	12/2010
University of Stuttgart	12/2010
HS Cooler, Wittenburg	12/2010
Visteon, Novi Jicin, Czech Republic	12/2010
CompuWave, Brunntal	12/2010
Stadtwerke Leipzig	12/2010
MCI Innsbruck, Austria	12/2010
EVONIK Energy Services, Zwingenberg	12/2010
Caliqua, Basel, Switzerland	11/2010
Shanghai New Energy Resources Science & Technology, China	11/2010
Energieversorgung Halle	11/2010
Hochschule für Technik Stuttgart, University of Applied Sciences	11/2010
Steinmueller, Berlin	11/2010
Amberg-Weiden University of Applied Sciences	11/2010
AREVA NP, Erlangen	10/2010
MAN Diesel, Augsburg	10/2010
KRONES, Neutraubling	10/2010
Vaillant, Remscheid	10/2010

PC Ware, Leipzig	10/2010
Schubert Consulting Engineers, Weißenberg	10/2010
Fraunhofer Institut UMSICHT, Oberhausen	10/2010
Behringer Consulting Engineers, Tagmersheim	09/2010
Saacke, Bremen	09/2010
WEBASTO, Neubrandenburg	09/2010
Concordia University, Montreal, Canada	09/2010
Compañía Eléctrica de Sochagota, Bogota, Colombia	08/2010
Hannover University of Applied Sciences	08/2010
ERGION, Mannheim	07/2010
Fichtner IT Consulting, Stuttgart	07/2010
TF Design, Matieland, South Africa	07/2010
MCE, Berlin	07/2010, 12/2010
IPM, Zittau/Goerlitz University of Applied Sciences	06/2010
TUEV Sued, Dresden	06/2010
RWE IT, Essen	06/2010
Glen Dimplex, Kulmbach	05/2010, 07/2010 10/2010
Hot Rock, Karlsruhe	05/2010
Darmstadt University of Applied Sciences	05/2010
Voith, Heidenheim	04/2010
CombTec, Zittau	04/2010
University of Glasgow, Great Britain	04/2010
Universitaet der Bundeswehr, Munich	04/2010
Technical University of Hamburg-Harburg	04/2010
Vattenfall Europe, Berlin	04/2010
HUBER Consulting Engineers, Berching	04/2010
VER, Dresden	04/2010
CCP, Marburg	03/2010
Offenburg University of Applied Sciences	03/2010
Technical University of Berlin	03/2010
NIST Boulder CO, USA	03/2010
Technical University of Dresden	02/2010
Siemens Energy, Nuremberg	02/2010
Augsburg University of Applied Sciences	02/2010

ALSTOM Power, Baden, Switzerland	02/2010, 05/2010
MIT Massachusetts Institute of Technology Cambridge MA, USA	02/2010
Wieland Werke, Ulm	01/2010
Siemens Energy, Goerlitz	01/2010, 12/2010
Technical University of Freiberg	01/2010
ILK, Dresden	01/2010, 12/2010
Fischer-Uhrig Consulting Engineers, Berlin	01/2010

2009

ALSTOM Power, Baden, Schweiz	01/2009, 03/2009, 05/2009
Nordostschweizerische Kraftwerke AG, Doettingen, Switzerland	02/2009
RWE, Neurath	02/2009
Brandenburg University of Technology, Cottbus	02/2009
Hamburg University of Applied Sciences	02/2009
Kehrein, Moers	03/2009
EPP Software, Marburg	03/2009
Bernd Münstermann, Telgte	03/2009
Suedzucker, Zeitz	03/2009
CPP, Marburg	03/2009
Gelsenkirchen University of Applied Sciences	04/2009
Regensburg University of Applied Sciences	05/2009
Gatley & Associates, Atlanta, USA	05/2009
BOSCH, Stuttgart	06/2009, 07/2009
Dr. Nickolay, Consulting Engineers, Gommersheim	06/2009
Ferrostal Power, Saarlouis	06/2009
BHR Bilfinger, Essen	06/2009
Intraserv, Wiesbaden	06/2009
Lausitz University of Applied Sciences, Senftenberg	06/2009
Nuernberg University of Applied Sciences	06/2009
Technical University of Berlin	06/2009
Fraunhofer Institut UMSICHT, Oberhausen	07/2009
Bischoff, Aurich	07/2009
Fichtner IT Consulting, Stuttgart	07/2009
Techsoft, Linz, Austria	08/2009
DLR, Stuttgart	08/2009

Wienstrom, Vienna, Austria	08/2009
RWTH Aachen University	09/2009
Vattenfall, Hamburg	10/2009
AIC, Chemnitz	10/2009
Midiplan, Bietigheim-Bissingen	11/2009
Institute of Air Handling and Refrigeration ILK, Dresden	11/2009
FZD, Rossendorf	11/2009
Techgroup, Ratingen	11/2009
Robert Sack, Heidelberg	11/2009
EC, Heidelberg	11/2009
MCI, Innsbruck, Austria	12/2009
Saacke, Bremen	12/2009
ENERKO, Aldenhoven	12/2009

2008

Pink, Langenwang	01/2008
Fischer-Uhrig, Berlin	01/2008
University of Karlsruhe	01/2008
MAAG, Kuesnacht, Switzerland	02/2008
M&M Turbine Technology, Bielefeld	02/2008
Lentjes, Ratingen	03/2008
Siemens Power Generation, Goerlitz	04/2008
Evonik, Zwingenberg (general EBSILON program license)	04/2008
WEBASTO, Neubrandenburg	04/2008
CFC Solutions, Munich	04/2008
RWE IT, Essen	04/2008
Rerum Cognitio, Zwickau	04/2008, 05/2008
ARUP, Berlin	05/2008
Research Center, Karlsruhe	07/2008
AWECO, Neukirch	07/2008
Technical University of Dresden, Professorship of Building Services	07/2008
Technical University of Cottbus, Chair in Power Plant Engineering	07/2008, 10/2008
Ingersoll-Rand, Unicov, Czech Republic	08/2008
Technip Benelux BV, Zoetermeer, Netherlands	08/2008
Fennovoima Oy, Helsinki, Finland	08/2008

Fichtner Consulting & IT, Stuttgart	09/2008
PEU, Espenhain	09/2008
Poyry, Dresden	09/2008
WINGAS, Kassel	09/2008
TUEV Sued, Dresden	10/2008
Technical University of Dresden, Professorship of Thermic Energy Machines and Plants	10/2008, 11/2008
AWTEC, Zurich, Switzerland	11/2008
Siemens Power Generation, Erlangen	12/2008

2007

Audi, Ingolstadt	02/2007
ANO Abfallbehandlung Nord, Bremen	02/2007
TUEV NORD SysTec, Hamburg	02/2007
VER, Dresden	02/2007
Technical University of Dresden, Chair in Jet Propulsion Systems	02/2007
Redacom, Nidau, Switzerland	02/2007
Universität der Bundeswehr, Munich	02/2007
Maxxtec, Sinsheim	03/2007
University of Rostock, Chair in Technical Thermodynamics	03/2007
AGO, Kulmbach	03/2007
University of Stuttgart, Chair in Aviation Propulsions	03/2007
Siemens Power Generation, Duisburg	03/2007
ENTHAL Haustechnik, Rees	05/2007
AWECO, Neukirch	05/2007
ALSTOM, Rugby, Great Britain	06/2007
SAAS, Possendorf	06/2007
Grenzebach BSH, Bad Hersfeld	06/2007
Reichel Engineering, Haan	06/2007
Technical University of Cottbus, Chair in Power Plant Engineering	06/2007
Voith Paper Air Systems, Bayreuth	06/2007
Egger Holzwerkstoffe, Wismar	06/2007
Tissue Europe Technologie, Mannheim	06/2007
Dometic, Siegen	07/2007
RWTH Aachen University, Institute for Electrophysics	09/2007
National Energy Technology Laboratory, Pittsburg, USA	10/2007

Energieversorgung Halle	10/2007
AL-KO, Jettingen	10/2007
Grenzebach BSH, Bad Hersfeld	10/2007
Wiesbaden University of Applied Sciences, Department of Engineering Sciences	10/2007
Endress+Hauser Messtechnik, Hannover	11/2007
Munich University of Applied Sciences, Department of Mechanical Engineering	11/2007
Rerum Cognitio, Zwickau	12/2007
Siemens Power Generation, Erlangen	11/2007
University of Rostock, Chair in Technical Thermodynamics	11/2007, 12/2007

2006

STORA ENSO Sachsen, Eilenburg	01/2006
Technical University of Munich, Chair in Energy Systems	01/2006
NUTEC Engineering, Bisikon, Switzerland	01/2006, 04/2006
Conwel eco, Bochov, Czech Republic	01/2006
Offenburg University of Applied Sciences	01/2006
KOCH Transporttechnik, Wadgassen	01/2006
BEG Bremerhavener Entsorgungsgesellschaft	02/2006
Deggendorf University of Applied Sciences, Department of Mechanical Engineering and Mechatronics	02/2006
University of Stuttgart, Department of Thermal Fluid Flow Engines	02/2006
Technical University of Munich, Chair in Apparatus and Plant Engineering	02/2006
Energietechnik Leipzig (company license),	02/2006
Siemens Power Generation, Erlangen	02/2006, 03/2006
RWE Power, Essen	03/2006
WAETAS, Pobershau	04/2006
Siemens Power Generation, Goerlitz	04/2006
Technical University of Braunschweig, Department of Thermodynamics	04/2006
EnviCon & Plant Engineering, Nuremberg	04/2006
Brassel Engineering, Dresden	05/2006
University of Halle-Merseburg, Department of USET Merseburg incorporated society	05/2006
Technical University of Dresden, Professorship of Thermic Energy Machines and Plants	05/2006

Fichtner Consulting & IT Stuttgart (company licenses and distribution)	05/2006
Suedzucker, Ochsenfurt	06/2006
M&M Turbine Technology, Bielefeld	06/2006
Feistel Engineering, Volkach	07/2006
ThyssenKrupp Marine Systems, Kiel	07/2006
Caliqua, Basel, Switzerland (company license)	09/2006
Atlas-Stord, Rodovre, Denmark	09/2006
Konstanz University of Applied Sciences, Course of Studies Construction and Development	10/2006
Siemens Power Generation, Duisburg	10/2006
Hannover University of Applied Sciences, Department of Mechanical Engineering	10/2006
Siemens Power Generation, Berlin	11/2006
Zikesch Armaturentechnik, Essen	11/2006
Wismar University of Applied Sciences, Seafaring Department	11/2006
BASF, Schwarzheide	12/2006
Enertech Energie und Technik, Radebeul	12/2006

2005

TUEV Nord, Hannover	01/2005
J.H.K Plant Engineering and Service, Bremerhaven	01/2005
Electrowatt-EKONO, Zurich, Switzerland	01/2005
FCIT, Stuttgart	01/2005
Energietechnik Leipzig (company license)	02/2005, 04/2005, 07/2005
eta Energieberatung, Pfaffenhofen	02/2005
FZR Forschungszentrum, Rossendorf/Dresden	04/2005
University of Saarbruecken	04/2005
Technical University of Dresden Professorship of Thermic Energy Machines and Plants	04/2005
Grenzebach BSH, Bad Hersfeld	04/2005
TUEV Nord, Hamburg	04/2005
Technical University of Dresden, Waste Management	05/2005
Siemens Power Generation, Goerlitz	05/2005
Duesseldorf University of Applied Sciences, Department of Mechanical Engineering and Process Engineering	05/2005

Redacom, Nidau, Switzerland	06/2005
Dumas Verfahrenstechnik, Hofheim	06/2005
Alensys Engineering, Erkner	07/2005
Stadtwerke Leipzig	07/2005
SaarEnergie, Saarbruecken	07/2005
ALSTOM ITC, Rugby, Great Britain	08/2005
Technical University of Cottbus, Chair in Power Plant Engineering	08/2005
Vattenfall Europe, Berlin (group license)	08/2005
Technical University of Berlin	10/2005
Basel University of Applied Sciences, Department of Mechanical Engineering, Switzerland	10/2005
Midiplan, Bietigheim-Bissingen	11/2005
Technical University of Freiberg, Chair in Hydrogeology	11/2005
STORA ENSO Sachsen, Eilenburg	12/2005
Energieversorgung Halle (company license)	12/2005
KEMA IEV, Dresden	12/2005

2004

Vattenfall Europe (group license)	01/2004
TUEV Nord, Hamburg	01/2004
University of Stuttgart, Institute of Thermodynamics and Heat Engineering	02/2004
MAN B&W Diesel A/S, Copenhagen, Denmark	02/2004
Siemens AG Power Generation, Erlangen	02/2004
Ulm University of Applied Sciences	03/2004
Visteon, Kerpen	03/2004, 10/2004
Technical University of Dresden, Professorship of Thermic Energy Machines and Plants	04/2004
Rerum Cognitio, Zwickau	04/2004
University of Saarbruecken	04/2004
Grenzebach BSH, Bad Hersfeld	04/2004
SOFBID Zwingenberg (general EBSILON program license)	04/2004
EnBW Energy Solutions, Stuttgart	05/2004
HEW-Kraftwerk, Tiefstack	06/2004
h s energieanlagen, Freising	07/2004
FCIT, Stuttgart	08/2004
Physikalisch Technische Bundesanstalt (PTB), Braunschweig	08/2004
Mainova Frankfurt	08/2004

Rietschle Energieplaner, Winterthur, Switzerland	08/2004
MAN Turbo Machines, Oberhausen	09/2004
TUEV Sued, Dresden	10/2004
STEAG Kraftwerk, Herne	10/2004, 12/2004
University of Weimar	10/2004
energeticals (e-concept), Munich	11/2004
SorTech, Halle	11/2004
Enertech EUT, Radebeul (company license)	11/2004
Munich University of Applied Sciences	12/2004
STORA ENSO Sachsen, Eilenburg	12/2004
Technical University of Cottbus, Chair in Power Plant Engineering	12/2004
Freudenberg Service, Weinheim	12/2004

2003

Paper Factory, Utzenstorf, Switzerland	01/2003
MAB Plant Engineering, Vienna, Austria	01/2003
Wulff Energy Systems, Husum	01/2003
Technip Benelux BV, Zoetermeer, Netherlands	01/2003
ALSTOM Power, Baden, Switzerland	01/2003, 07/2003
VER, Dresden	02/2003
Rietschle Energieplaner, Winterthur, Switzerland	02/2003
DLR, Leupholdhausen	04/2003
Emden University of Applied Sciences, Department of Technology	05/2003
Petterssson+Ahrends, Ober-Moerlen	05/2003
SOFBID ,Zwingenberg (general EBSILON program license)	05/2003
Ingenieurbuero Ostendorf, Gummersbach	05/2003
TUEV Nord, Hamburg	06/2003
Muenstermann GmbH, Telgte-Westbevern	06/2003
University of Cali, Colombia	07/2003
Atlas-Stord, Rodovre, Denmark	08/2003
ENERKO, Aldenhoven	08/2003
STEAG RKB, Leuna	08/2003
eta Energieberatung, Pfaffenhofen	08/2003
exergie, Dresden	09/2003
AWTEC, Zurich, Switzerland	09/2003
Energie, Timelkam, Austria	09/2003

Electrowatt-EKONO, Zurich, Switzerland	09/2003
LG, Annaberg-Buchholz	10/2003
FZR Forschungszentrum, Rossendorf/Dresden	10/2003
EnviCon & Plant Engineering, Nuremberg	11/2003
Visteon, Kerpen	11/2003
VEO Vulkan Energiewirtschaft Oderbruecke, Eisenhuettenstadt	11/2003
Stadtwerke Hannover	11/2003
SaarEnergie, Saarbruecken	11/2003
Fraunhofer-Gesellschaft, Munich	12/2003
Erfurt University of Applied Sciences, Department of Supply Engineering	12/2003
SorTech, Freiburg	12/2003
Mainova, Frankfurt	12/2003
Energieversorgung Halle	12/2003

2002

Hamilton Medical AG, Rhaezuens, Switzerland	01/2002
Bochum University of Applied Sciences, Department of Thermo- and Fluid Dynamics	01/2002
SAAS, Possendorf/Dresden	02/2002
Siemens, Karlsruhe (general license for the WinIS information system)	02/2002
FZR Forschungszentrum, Rossendorf/Dresden	03/2002
CompAir, Simmern	03/2002
GKS Gemeinschaftskraftwerk, Schweinfurt	04/2002
ALSTOM Power Baden, Switzerland (group licenses)	05/2002
InfraServ, Gendorf	05/2002
SoftSolutions, Muehlhausen (company license)	05/2002
DREWAG, Dresden (company license)	05/2002
SOFBID, Zwingenberg (general EBSILON program license)	06/2002
Kleemann Engineering, Dresden	06/2002
Caliqua, Basel, Switzerland (company license)	07/2002
PCK Raffinerie, Schwedt (group license)	07/2002
Fischer-Uhrig Engineering, Berlin	08/2002
Fichtner Consulting & IT, Stuttgart (company licenses and distribution)	08/2002
Stadtwerke Duisburg	08/2002

Stadtwerke Hannover	09/2002
Siemens Power Generation, Goerlitz	10/2002
Energieversorgung Halle (company license)	10/2002
Bayer, Leverkusen	11/2002
Dillinger Huette, Dillingen	11/2002
G.U.N.T. Geraetebau, Barsbuettel (general license and training test benches)	12/2002
VEAG, Berlin (group license)	12/2002

2001

ALSTOM Power, Baden, Switzerland	01/2001, 06/2001, 12/2001
KW2 B. V., Amersfoot, Netherlands	01/2001, 11/2001
Eco Design, Saitamaken, Japan	01/2001
M&M Turbine Technology, Bielefeld	01/2001, 09/2001
MVV Energie, Mannheim	02/2001
Technical University of Dresden, Department of Power Machinery and Plants	02/2001
PREUSSAG NOELL, Wuerzburg	03/2001
Fichtner Consulting & IT Stuttgart (company licenses and distribution)	04/2001
Muenstermann GmbH, Telgte-Westbevern	05/2001
SaarEnergie, Saarbruecken	05/2001
Siemens, Karlsruhe (general license for the WinIS information system)	08/2001
Neusiedler AG, Ulmerfeld, Austria	09/2001
h s energieanlagen, Freising	09/2001
Electrowatt-EKONO, Zurich, Switzerland	09/2001
IPM Zittau/Goerlitz University of Applied Sciences (general license)	10/2001
eta Energieberatung, Pfaffenhofen	11/2001
ALSTOM Power Baden, Switzerland	12/2001
VEAG, Berlin (group license)	12/2001

2000

SOFBID, Zwingenberg (general EBSILON program license)	01/2000
AG KKK - PGW Turbo, Leipzig	01/2000
PREUSSAG NOELL, Wuerzburg	01/2000
M&M Turbine Technology, Bielefeld	01/2000

IBR Engineering Reis, Nittendorf-Undorf	02/2000
GK, Hannover	03/2000
KRUPP-UHDE, Dortmund (company license)	03/2000
UMAG W. UDE, Husum	03/2000
VEAG, Berlin (group license)	03/2000
Thinius Engineering, Erkrath	04/2000
SaarEnergie, Saarbruecken	05/2000, 08/2000
DVO Data Processing Service, Oberhausen	05/2000
RWTH Aachen University	06/2000
VAUP Process Automation, Landau	08/2000
Knuerr-Lommatec, Lommatzsch	09/2000
AVACON, Helmstedt	10/2000
Compania Electrica, Bogota, Colombia	10/2000
G.U.N.T. Geraetebau, Barsbuettel (general license for training test benches)	11/2000
Steinhaus Informationssysteme, Datteln (general license for process data software)	12/2000

1999

Bayernwerk, Munich	01/1999
DREWAG, Dresden (company license)	02/1999
KEMA IEV, Dresden	03/1999
Regensburg University of Applied Sciences	04/1999
Fichtner Consulting & IT, Stuttgart (company licenses and distribution)	07/1999
Technical University of Cottbus, Chair in Power Plant Engineering	07/1999
Technical University of Graz, Department of Thermal Engineering, Austria	11/1999
Ostendorf Engineering, Gummersbach	12/1999

1998

Technical University of Cottbus, Chair in Power Plant Engineering	05/1998
Fichtner Consulting & IT (CADIS information systems) Stuttgart (general KPRO program license)	05/1998
M&M Turbine Technology Bielefeld	06/1998
B+H Software Engineering Stuttgart	08/1998
Alfa Engineering, Switzerland	09/1998
VEAG Berlin (group license)	09/1998
NUTEC Engineering, Bisikon, Switzerland	10/1998

SCA Hygiene Products, Munich	10/1998
RWE Energie, Neurath	10/1998
Wilhelmshaven University of Applied Sciences	10/1998
BASF, Ludwigshafen (group license)	11/1998
Energieversorgung, Offenbach	11/1998

1997

Gerb, Dresden	06/1997
Siemens Power Generation, Goerlitz	07/1997