Property Library for Carbon Dioxide

FluidLAB
with LibCO2
for MATLAB®

Prof. Hans-Joachim Kretzschmar
Dr. Sebastian Herrmann
Dr. Matthias Kunick

Property Software for Carbon Dioxide

FluidLAB for MATLAB®

LibCO2

Contents

Professor Hans-Joachim Kretzschmar
Wallotstr. 3, 01307 Dresden, Germany
Phone: +49-351-27597860
Mobile: +49-172-7914607
Fax: +49-3222-4262250
Email: info@thermofluidprop.com
Internet: www.thermofluidprop.com

0 Package Contents

0.1 Zip file for 32-bit MATLAB®

The following zip file is delivered for your computer running a 32-bit version of MATLAB®.

"CD_FluidLAB_LibCO2.zip"

Including the following files:

FluidLAB_LibCO2_Setup.msi	- Installation program for the FluidLAB Add-On for use in MATLAB®
LibCO2.dll	- Dynamic Link Library for carbon dioxide for use in MATLAB®
FluidLAB_LibCO2_Docu_Eng.pdf	- User's Guide
Setup.exe	- Self-extracting and self-installing program for FluidLAB

0.2 Zip file for 64-bit MATLAB®

The following zip file is delivered for your computer running a 64-bit version of MATLAB®.

"CD_FluidLAB_LibCO2_x64.zip"

Including the following files and folders:

Files:

Setup.exe	- Self-extracting and self-installing program for FluidLAB
FluidLAB_LibCO2_64.msi	- Installation program for the FluidLAB Add-On for use in MATLAB®
LibCO2.dll	- Dynamic Link Library for carbon dioxide for use in MATLAB®
FluidLAB_LibCO2_Docu_Eng.pdf	- User's Guide

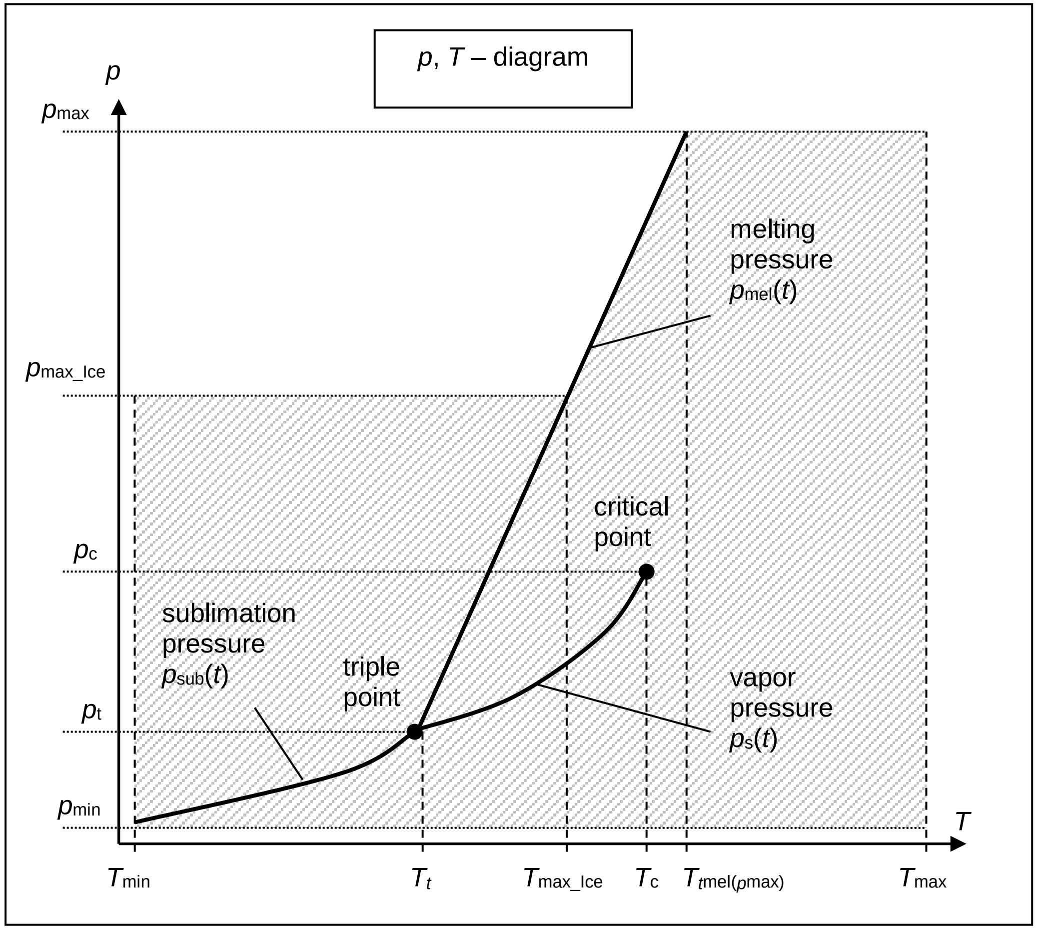
Folders:

vcredist_x64	- Folder containing the "Microsoft Visual C++ 2010 x64 Redistributable Pack"
WindowsInstaller3_1	- Folder containing the "Microsoft Windows Installer"

1. Property Functions

Units: t in °C
p in bar
x in kg /kg (Phase fraction, see the following explanations)

Range of Validity:



Reference State:

At p = 1.01325 bar and T = 298.15 K (25 °C) :
h = –0.938457860 kJ/kg and s = –0.00219606205 kJ/(kg K)

Factor	Abbreviation	Value and Unit
Minimum temperature	T_{min} (t_{min})	85 K (–188.15 °C)
Maximum temperature	T_{max} (t_{max})	1500 K (1226.85 °C)
Triple temperature	T_t (t_t)	216.592 K (–56.558 °C)
Temperature at the critical point	T_c (t_c)	304.1282 K (30.9782 °C)
Maximum temperature of solid region	T_{max_Ice} (t_{miax_Ice})	236.0309 K (–37.119 °C)
Maximum temperature of melting pressure curve	$T_{mel}(p_{max})$ ($t_{mel}(p_{max})$)	327.671 K (54.521 °C)
Minimum pressure	$p_{min} = p_{sub}(T_{min})$	$2.9081875815 \cdot 10^{-10}$ bar
Maximum pressure	p_{max}	8000 bar
Triple pressure	p_t	5.179618369088 bar
Pressure at the critical point	p_c	73.773 bar
Maximum pressure of solid region	p_{max_Ice}	1000 bar

General Property Functions

Functional Dependence	Function Name	Call from Fortran program	Call from the DLL LibCO2 as Parameter	Property or Function	Unit of the Result
a = f(p,t,x)	a_ptx_CO2	APTXCO2(P,T,X)	C_APTXCO2(A,P,T,X)	Thermal diffusivity	m^2/s
c_p = f(p,t,x)	cp_ptx_CO2	CPPTXCO2(P,T,X)	C_CPPTXCO2(CP,P,T,X)	Specific isobaric heat capacity	kJ/(kg K)
η = f(p,t,x)	eta_ptx_CO2	ETAPTCO2(P,T,X)	C_ETAPTXCO2(ETA,P,T,X)	Dynamic viscosity	Pa s
h = f(p,t,x)	h_ptx_CO2	HPTXCO2(P,T,X)	C_HPTXCO2(H,P,T,X)	Specific enthalpy	kJ/kg
κ = f(p,t,x)	kappa_ptx_CO2	KAPTXCO2(P,T,X)	C_KAPTXCO2(KAP,P,T,X)	Isentropic exponent	-
λ = f(p,t,x)	lambda_ptx_CO2	LAMPTCO2(P,T,X)	C_LAMPTXCO2(LAM,P,T,X)	Thermal conductivity	W/(m K)
ν = f(p,t,x)	ny_ptx_CO2	NYPTXCO2(P,T,X)	C_NYPTXCO2(NY,P,T,X)	Kinematic viscosity	m^2/s
p_{mel} = f(t)	pmel_t_CO2	PMELCO2(T)	C_PMELCO2(PMEL,T)	Melting pressure from temperature	bar
p_{sub} = f(t)	psub_t_CO2	PSUBCO2(T)	C_PSUBCO2(PSUB,T)	Sublimation pressure from temperature	bar
p_s = f(t)	ps_t_CO2	PSTCO2(T)	C_PSTCO2(PS,T)	Vapor pressure from temperature	bar
Pr = f(p,t,x)	Pr_ptx_CO2	PRPTXCO2(P,T,X)	C_PRPTXCO2(PR,P,T,X)	*Prandtl*-Number	-
ρ = f(p,t,x)	rho_ptx_CO2	ROPTXCO2(P,T,X)	C_ROPTXCO2(RHO,P,T,X)	Density	kg/ m^3
s = f(p,t,x)	s_ptx_CO2	SPTXCO2(P,T,X)	C_SPTXCO2(S,P,T,X)	Specific entropy	kJ/(kg K)
t = f(p,h)	t_ph_CO2	TPHCO2(P,H)	C_TPHCO2(T,P,H)	Backward function: Temperature from pressure and enthalpy	°C
t = f(p,s)	t_ps_CO2	TPSCO2(P,S)	C_TPSCO2(T,P,S)	Backward function: Temperature from pressure and entropy	°C

Functional Dependence	Function Name	Call from Fortran program	Call from the DLL LibCO2 as Parameter	Property or Function	Unit of the Result
$t_{mel} = f(p)$	tmel_p_CO2	TMELCO2(P)	C_TMELCO2(TMEL,P)	Melting temperature from pressure	°C
$t_{sub} = f(p)$	tsub_p_CO2	TSUBCO2(P)	C_TSUBCO2(TSUB,P)	Sublimation temperature from pressure	°C
$t_s = f(p)$	ts_p_CO2	TSPCO2(P)	C_TSPCO2(TS,P)	Saturation temperature from pressure	°C
$v = f(p,t,x)$	v_ptx_CO2	VPTXCO2(P,T,X)	C_VPTXCO2(V,P,T,X)	Specific volume	m^3/kg
$w = f(p,t,x)$	w_ptx_CO2	WPTXCO2(P,T,X)	C_WPTXCO2(W,P,T,X)	Isentropic speed of sound	m/s
$x = f(p,h)$	x_ph_CO2	XPHCO2(P,H)	C_XPHCO2(X,P,H)	Backward function: Phase fraction from pressure and enthalpy	kg/kg
$x = f(p,s)$	x_ps_CO2	XPSCO2(P,S)	C_XPSCO2(X,P,S)	Backward function: Phase fraction from pressure and entropy	kg/kg

Property Functions for Solid Carbon Dioxide (Dry Ice)

Functional Dependence	Function Name	Call from Fortran program	Call from the DLL LibCO2 as Parameter	Property or Function	Unit of the Result
$a = f(p,t)$	aICE_pt_CO2	APICETCO2(P,T)	C_APICEPTCO2 (A, P, T)	Thermal diffusivity	m^2/s
$c_p = f(p,t)$	cpICE_pt_CO2	CPICETCO2(P,T)	C_CPICEPTCO2 (CP, P, T)	Specific isobaric heat capacity	kJ/(kg K)
$h = f(p,t)$	hICE_pt_CO2	HICETCO2(P,T)	C_HICEPTCO2 (H, P, T)	Specific enthalpy	kJ/ kg
$\lambda = f(t)$	lambdaICE_t_CO2	LAMICETCO2(T)	C_LAMICETCO2 (LAM,T)	Thermal conductivity	W/(m K)
$\rho = f(p,t)$	rhoICE_pt_CO2	RHOICETCO2(P,T)	C_RHOICEPTCO2 (RHO, P, T)	Density	kg/ m^3
$s = f(p,t)$	sICE_pt_CO2	SICETCO2(P,T)	C_SICEPTCO2 (S, P, T)	Specific entropy	kJ/ (kg K)

Functional Dependence	Function Name	Call from Fortran program	Call from the DLL LibCO2 as Parameter	Property or Function	Unit of the Result
$v = f(p,t)$	vICE_pt_CO2	VICETCO2(P,T)	C_VICEPTCO2 (V, P, T)	Specific volume	m^3/kg
$t = f(p,h)$	tICE_ph_CO2	TICEHCO2(P,H)	C_TICEPHCO2 (T, P, H)	Backward function: Temperature from pressure and enthalpy	°C
$t = f(p,s)$	tICE_ps_CO2	TICESCO2(P,S)	C_TICEPSCO2 (T, P, S)	Backward function: Temperature from pressure and entropy	°C

Details on the Phase Fraction *x*

If the state point to be calculated is located in the single phase region (solid, liquid or superheated vapor), x= –1 must be entered as a pro-forma value. Here the backward functions will also result in x = –1.

The two phase regions wet vapor region, melting region and sublimation region (cp. the following lg *p,h*-diagram) are calculated automatically by the subprograms. Please consider the following facts:

1. Wet Vapor Region (0 ≤ x ≤ 1):

The phase fraction *x* equates to the vapor fraction *x* in the wet vapor region in (kg dry saturated vapor)/(kg wet vapor).

In this case it is adequate to enter either the given value for *t* and p = –1000, or the given value for *p* and t = –1000, plus the value for *x* between 0 and 1. If values for both *p* and *t* are entered when calculating wet vapor, the program will consider *p* and *t* to be appropriate to represent the vapour pressure curve. If it is not the case the calculation for the quantity of the chosen function to be calculated results in –1000.

When calculating wet vapor, *x* results in a value between 0 and 1 (x = 0 for boiling liquid, x = 1 for dry saturated vapor). In this case, the backward functions result in the appropriate value between 0 and 1 for *x*.

Wet vapor region: Temperature range from T_t = 216.592 K (t_t =–56.558 °C) to T_c = 304.1282 K (t_t = 30.9782 °C)
Pressure ranges from p_t = 5.179618369088 bar to p_c = 73.773 bar

2. Melting Region (10 ≤ x ≤ 11):

The phase fraction *x* equates to the liquid fraction *x* in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.

In this case it is adequate to enter either the given value for *t* and p = –1000, or the given value for *p* and t = –1000, plus the value for *x* between 10 and 11. If values for both *p* and *t* are entered, the program will consider *p* and *t* to be appropriate to represent the melting pressure curve. If it is not the case the calculation for the quantity of the chosen function to be calculated results in –1000.

When calculating a melt, *x* results in a value between 10 and 11 (x = 10 for melting solid, x = 11 for solidifying liquid). In this case, the backward functions result in the appropriate value between 10 and 11 for *x*.

Melting region: Temperature range from T_{max_Ice} = 236.0309 K (t_{max_Ice} =–37.119 °C) to T_t = 216.592 K (t_t =–56.558 °C)
Pressure range from p_t = 5.179618369088 bar to p_{max_Ice} = 1000 bar

Only the limiting curve on the right hand side, thus solidifying liquid (x = 11) is calculated for pressures *p* for which applies $p_{max\_Ice} < p \leq p_{max}$.

3. Sublimation Region (100 ≤ x ≤ 101):

The phase fraction *x* equates to the vapor fraction *x* in the sublimation region in (kg desublimating vapor)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating vapor.

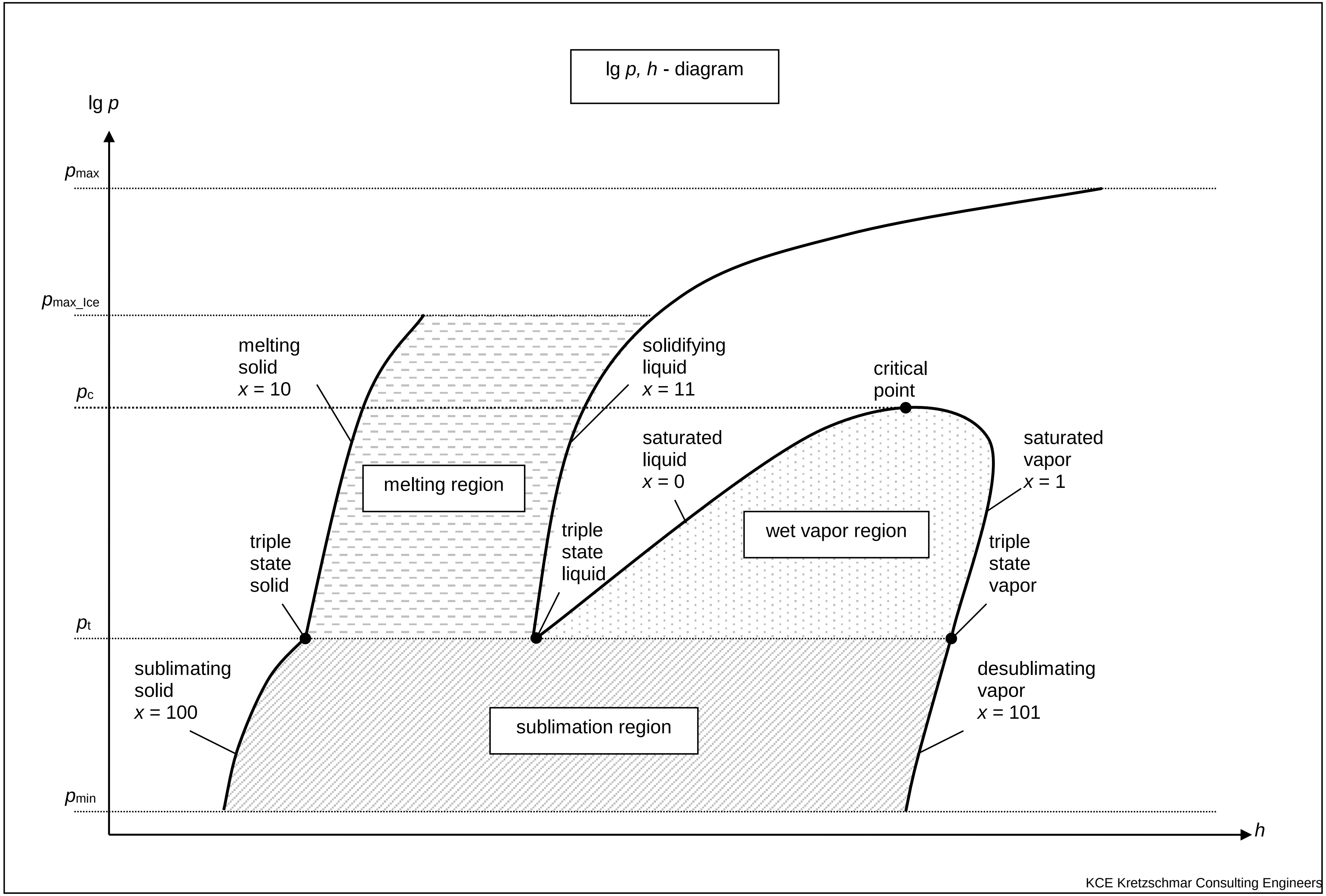
In this case it is adequate to enter either the given value for *t* and p = –1000, or the given value for *p* and t = –1000, plus the value for *x* between 100 and 101. If values for both *p* and *t* are entered, the program will consider *p* and *t* to be appropriate to represent the sublimation pressure curve. If it is not the case the calculation for the quantity of the chosen function to be calculated results in –1000.

When calculating sublimation powder, *x* takes a value between 100 and 101 (x = 100 for sublimating solid, x = 101 for desublimating vapor). In this case, the backward functions result in the appropriate value between 100 and 101 for *x*.

Sublimation region: Temperature range from T_{min} = 85 K (t_{min} =–188.15 °C) to T_t = 216.592 K (t_t =–56.558 °C)
Pressure range from p_{min} = $2.9081875815 \cdot 10^{-10}$ bar to p_t = 5.179618369088 bar

Note:

If the input values are located outside the range of validity, the calculated function will always result in –1000. Please find more exact details on every function and its corresponding range of validity in the enclosed software documentation in Chapter 3. The same information may also be accessed via the online help pages.


lg p, h - diagram
lg p
p_max
p_max_Ice
p_c
p_t
p_min
h
melting solid x = 10
solidifying liquid x = 11
critical point
saturated liquid x = 0
saturated vapor x = 1
melting region
wet vapor region
triple state solid
triple state liquid
triple state vapor
sublimating solid x = 100
desublimating vapor x = 101
sublimation region
KCE Kretzschmar Consulting Engineers

2 Application of FluidLAB in MATLAB®

The FluidLAB Add-In has been developed to calculate thermodynamic properties in MATLAB®. Within FluidLAB, it enables the direct call of functions relating to carbon dioxide from the LibCO2 property library.

2.1 Installing FluidLAB LibCO2

Installing FluidLAB including LibCO2 for 32-bit MATLAB®

This section describes the installation of FluidLAB LibCO2 for a 32-bit version of MATLAB®.
Before you begin, it is best to close any Windows® applications, since Windows® may need to be rebooted during the installation process.
After you have downloaded and extracted the zip-file "CD_FluidLAB_LibCO2.zip", you will see the folder

CD_FluidLAB_LibCO2

in your Windows Explorer®, Norton Commander® or another similar program you are using.

Open this folder by double-clicking on it.

In this folder you will see the following files:

FluidLAB_LibCO2_Docu_Eng.pdf
FluidLAB_LibCO2_Setup.msi
Setup.exe
LibCO2.dll.

In order to run the installation of FluidLAB including, the LibCO2 property library, double-click on the file

Setup.exe.

Installation may start with a window noting that all Windows® programs should be closed. When this is the case, the installation can be continued. Click the "Next >" button.

In the following dialog box, "Destination Location", the default path offered automatically for the installation of FluidLAB is

C:\Program Files\FluidLAB\LibCO2	(for English version of Windows)
C:\Programme\FluidLAB\LibCO2	(for German version of Windows)

By clicking the "Browse…" button, you can change the installation directory before installation (see Figure 2.1).

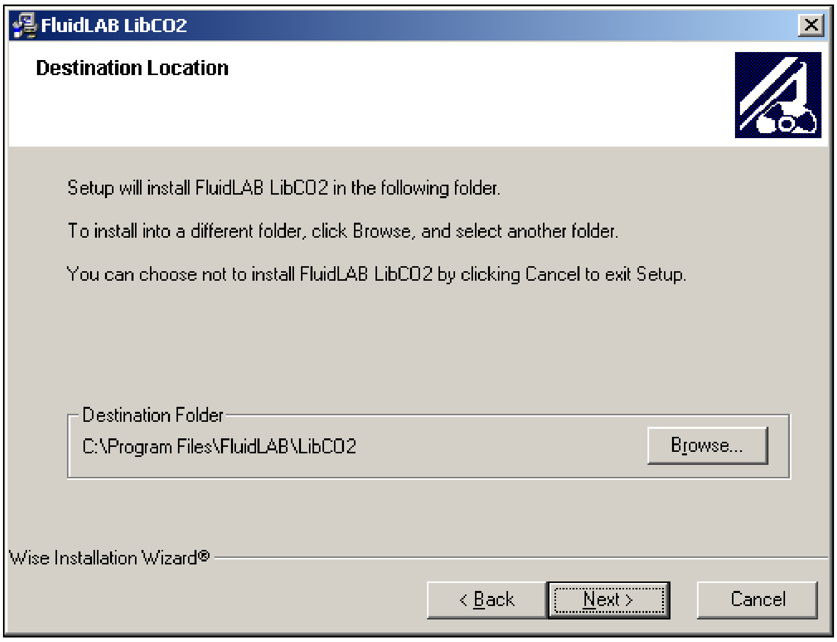


Figure 2.1: "Destination Location"

If you wish to change directories, click the "Browse…" button and select your desired directory. The instructions in this documentation refer to the stated default directory. Leave this window by clicking the "Next >" button.

The dialog window "Start Installation" pops up. Click the "Next >" button to continue installation. The FluidLAB files are now being copied into the created directory on your hard drive. Click the "Finish >" button in the following window to complete installation.

The installation program has copied the following files for LibCO2

advapi32.dll	LC.dll
Dformd.dll	msvcp60.dll
Dforrt.dll	msvcrt.dll
INSTALL.LOG	Unwise.exe
LibCO2.dll	Unwise.ini

into the directory

C:\Program Files\FluidLAB\LibCO2 (for English version of Windows)
C:\Programme\FluidLAB\LibCO2 (for German version of Windows).

Now, you have to overwrite the file "LibCO2.dll" in your FluidLAB directory with the file of the same name provided in your CD folder with FluidLAB LibCO2.

To do this, open the CD folder in "My Computer" and click on the file "LibCO2.dll" in order to highlight it.

Then click on the "Edit" menu in your Explorer and select "Copy".

Now, open your FluidLAB directory (the standard being

C:\Program Files\FluidLAB\LibCO2 (for English version of Windows)
C:\Programme\FluidLAB\LibCO2 (for German version of Windows))

and insert the file "LibCO2.dll" by clicking the "Edit" menu in your Explorer and then select "Paste". Answer the question whether you want to replace the file by clicking the "Yes" button. Now, you have overwritten the file "LibCO2.dll" successfully and the property functions are available in MATLAB.

Installing FluidLAB including LibCO2 for 64-bit MATLAB®

This section describes the installation of FluidLAB LibCO2 for a 64-bit version of MATLAB®. Before you begin, it is best to close any Windows® applications, since Windows® may need to be rebooted during the installation process.

After you have downloaded and extracted the zip-file "CD_FluidLAB_LibCO2_x64.zip", you will see the folder

CD_FluidLAB_LibCO2

in your Windows Explorer®, Norton Commander® or other similar program you are using.

Open this folder by double-clicking on it.

In this folder you will see the following files:

FluidLAB_LibCO2_64.msi
FluidLAB_LibCO2_Docu_Eng.pdf
LibCO2.dll
Setup.exe

and folders

/vcredist_x64
/WindowsInstaller3_1.

In order to run the installation of FluidLAB including, the LibCO2 property library, double-click on the file

Setup.exe.

Installation of FluidLAB LibCO2 starts with a window noting that the installer will guide you through the installation process. Click the "Next >" button to continue.

In the following dialog box, "Destination Location", the default path offered automatically for the installation of FluidLAB is

C:\Program Files\FluidLAB\LibCO2	(for English version of Windows)
C:\Programme\FluidLAB\LibCO2	(for German version of Windows)

By clicking the "Browse…" button, you can change the installation directory before installation (see Figure 2.2).

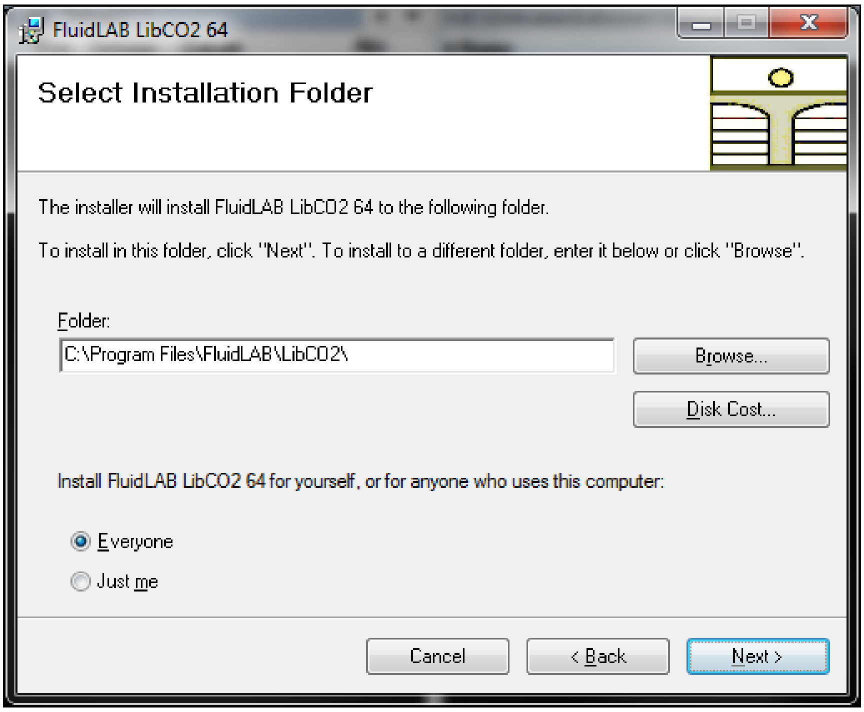


Figure 2.2: "Select Installation Folder"

Finally, click on "Next >" to continue installation; click "Next >" again in the "Confirm Installation" window which follows in order to start the installation of FluidLAB.

After FluidLAB has been installed, you will see the sentence "FluidLAB LibCO2 64 has been successfully installed." Confirm this by clicking the "Close" button.

The installation program has copied the following files for LibCO2

capt_ico_big.ico	libifcoremd.dll
LC.dll	libiomp5md.dll
LibCO2.dll	libmmd.dll

into the directory

C:\Program Files\FluidLAB\LibCO2	(for English version of Windows)
C:\Programme\FluidLAB\LibCO2	(for German version of Windows).

Now, you have to overwrite the file "LibCO2.dll" in your FluidLAB directory with the file of the same name provided in your CD folder with FluidLAB LibCO2.

To do this, open the CD folder in "My Computer" and click on the file "LibCO2.dll" in order to highlight it.

Then click on the "Edit" menu in your Explorer and select "Copy".

Now, open your FluidLAB directory (the standard being

C:\Program Files\FluidLAB\LibCO2	(for English version of Windows)
C:\Programme\FluidLAB\LibCO2	(for German version of Windows))

and insert the file "LibCO2.dll" by clicking the "Edit" menu in your Explorer and then select "Paste". Answer the question whether you want to replace the file by clicking the "Yes" button. Now, you have overwritten the file "LibCO2.dll" successfully and the property functions are available in MATLAB.

The installation programs for both the 32-bit and the 64-bit MATLAB® version have copied the following function files for LibCO2 into the directory

C:\Program Files\FluidLAB\LibCO2	(for English version of Windows)
C:\Programme\FluidLAB\LibCO2	(for German version of Windows):

- Dynamic Link Library "LibCO2.dll" and other necessary system DLL files.

- MATLAB®-Interface-Programme for calculable functions

a_ptx_CO2	rho_ptx_CO2
cp_ptx_CO2	s_ptx_CO2
eta_ptx_CO2	t_ph_CO2
h_ptx_CO2	t_ps_CO2
kappa_ptx_CO2	tmel_p_CO2
lambda_ptx_CO2	tsub_p_CO2
ny_ptx_CO2	ts_p_CO2
pmel_t_CO2	v_ptx_CO2
psub_t_CO2	w_ptx_CO2
ps_t_CO2	x_ph_CO2
Pr_ptx_CO2	x_ps_CO2
aICE_t_CO2	sICE_t_CO2
cpICE_t_CO2	vICE_t_CO2
hICE_t_CO2	tICE_h_CO2
lambdaICE_t_CO2	tICE_s_CO2
rhoICE_t_CO2	

Please note that there is a difference in the file extension of the function files.

The 32-bit installation program has copied function files with the file extension

.mexw32

and the 64-bit installation program has copied function files with the file extension

.mexw64

into your LibCO2 directory (the standard being

C:\Program Files\FluidLAB\LibCO2	(for English version of Windows)
C:\Programme\FluidLAB\LibCO2	(for German version of Windows)).

2.2 Licensing the LibCO2 Property Library

The licensing procedure must be carried out when the prompt message appears. In this case, you will see the "License Information" window for LibCO2 (see figure below).

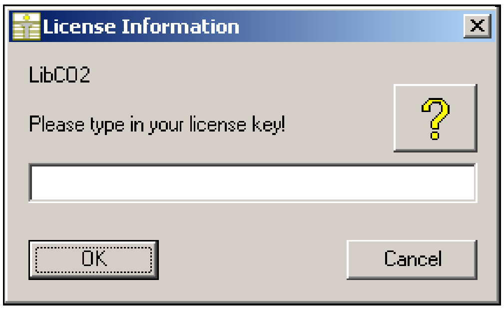


Figure 2.3: "License Information" window

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

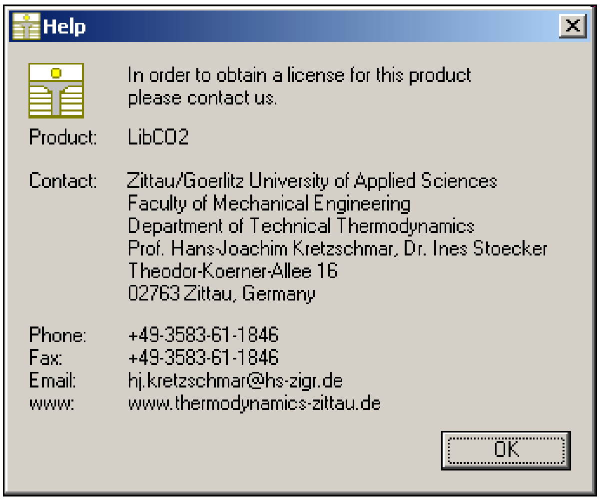


Figure 2.4: "Help" window

If you do not enter a valid license it is still possible to use MATLAB® by clicking "Cancel". In this case, the LibCO2 property library will display the result "–11111111" for every calculation.

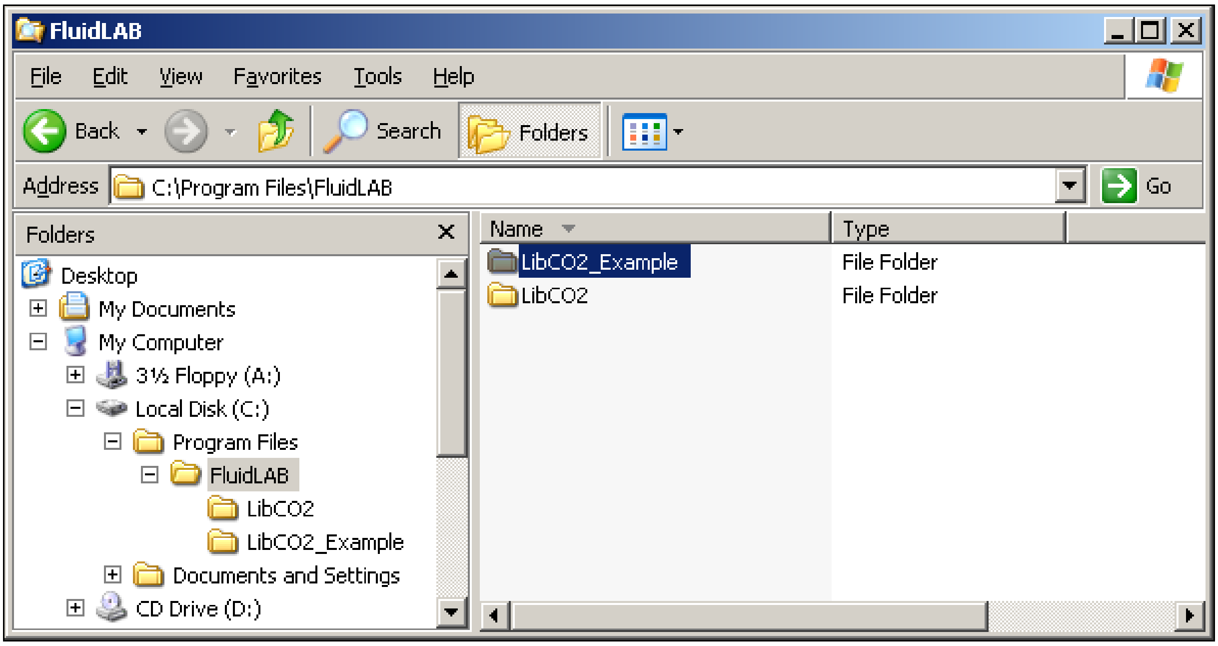
The "License Information" window will appear every time you use FluidLAB LibCO2 until you enter a license code to complete registration. If you decide not to use FluidLAB LibCO2, you can uninstall the program following the instructions given in section 2.5 of this user's guide.

2.3 Example: Calculation of the Specific Enthalpy *h* = f(*p*,*t*,*x*) for Carbon Dioxide in an M-File

Now we will calculate, step by step, the specific enthalpy *h* as a function of pressure *p*, temperature *t* and vapor fraction *x* for carbon dioxide using FluidLAB.

Please carry out the following instructions:

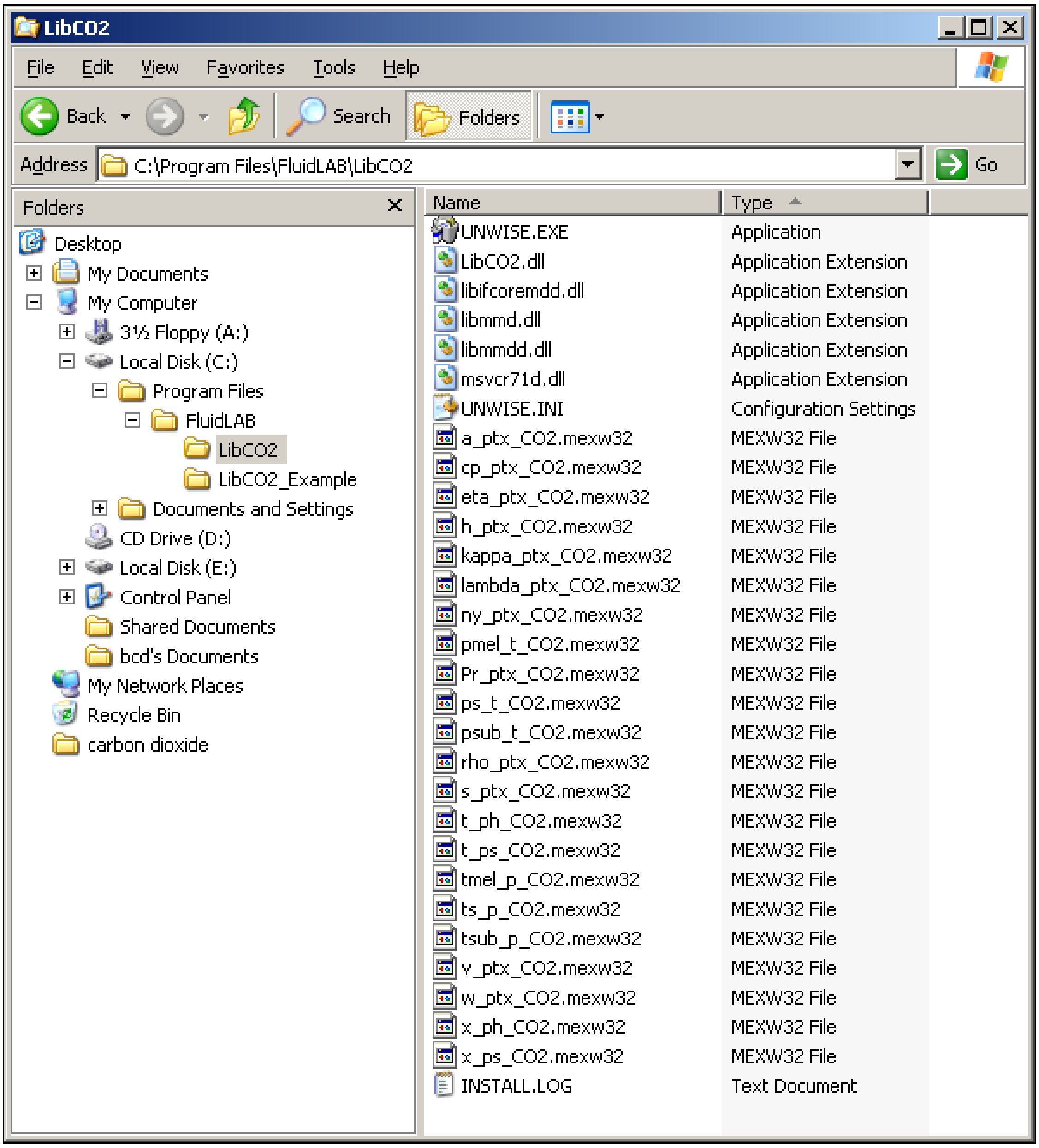
- Start Windows-Explorer®, Total Commander, My Computer or another file manager program.
  The following description refers to Windows-Explorer
- Your Windows-Explorer should be set to Details for a better view. Click the "Views" button and select "Details".
- Switch into the program directory of FluidLAB in which you will find the folder "\LibCO2"; in the standard case: "C:\Program Files\FluidLAB"
- Create the folder "\LibCO2_Example". Click "File", then click "New" in the pop-up menu and afterwards select "Folder". Name the new folder "\LibCO2_Example ".
- You will see the following window:



- Switch into the directory "\LibCO2" within "\FluidLAB", in the standard case:

C:\Program Files\FluidLAB\LibCO2	(for English version of Windows)
C:\Programme\FluidLAB\LibCO2	(for German version of Windows)).

- You will see the following window:



You will now have to copy the necessary files following into the directory

C:\Program Files\FluidLAB\LibCO2_Example (for English version of Windows)
C:\Programme\FluidLAB\LibCO2_Example (for German version of Windows)

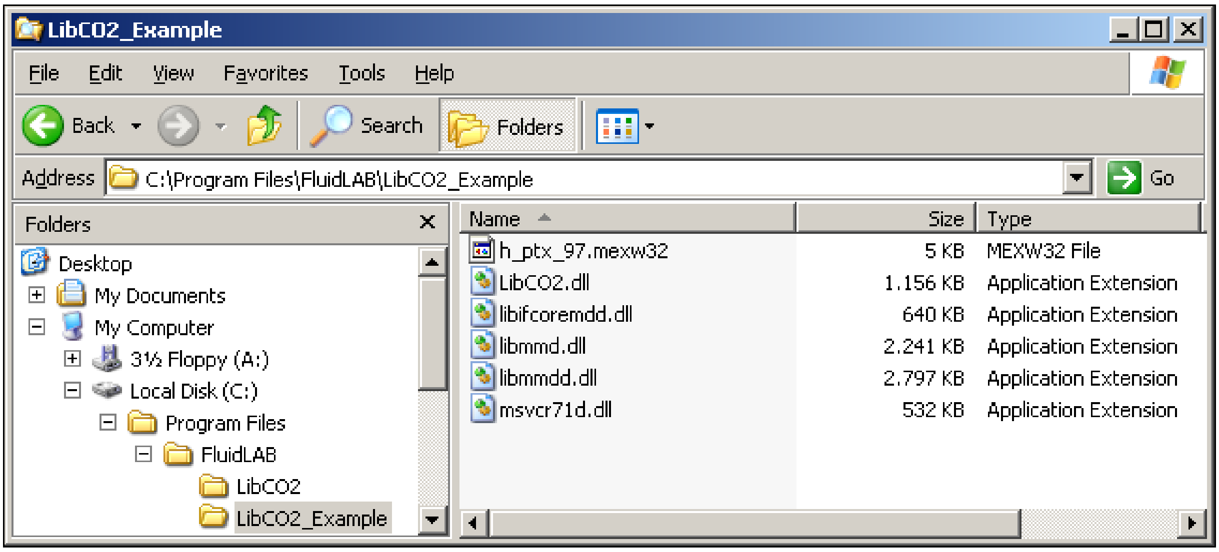
in order to calculate the function $h = f(p,t,x)$.

- The following files are necessary:
  - "h_ptx_CO2.mexw32"
  - "LibCO2.dll"
  - "libifcoremdd.dll"
  - "libmmd.dll"
  - "libmmdd.dll"
  - "msvcr71d.dll"

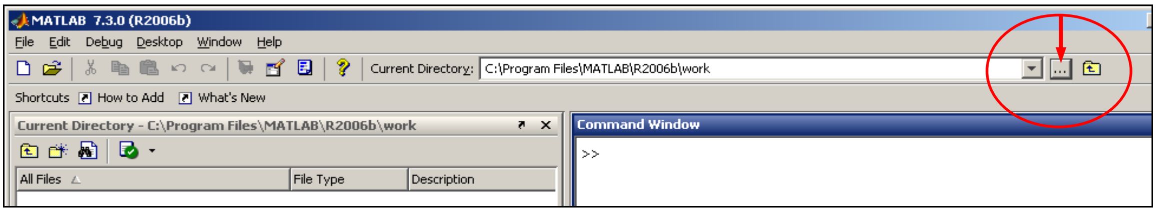
- Click the file "h_ptx_CO2.mexw32", then click "Edit" in the upper menu bar and select "Copy".
- Switch into the directory

  C:\Program Files\FluidLAB\LibCO2_Example (for English version of Windows)
  C:\Programme\FluidLAB\LibCO2_Example (for German version of Windows),

  click "Edit" and select "Paste".
- Repeat these steps in order to copy the other files listed above.
- You will see the following window:



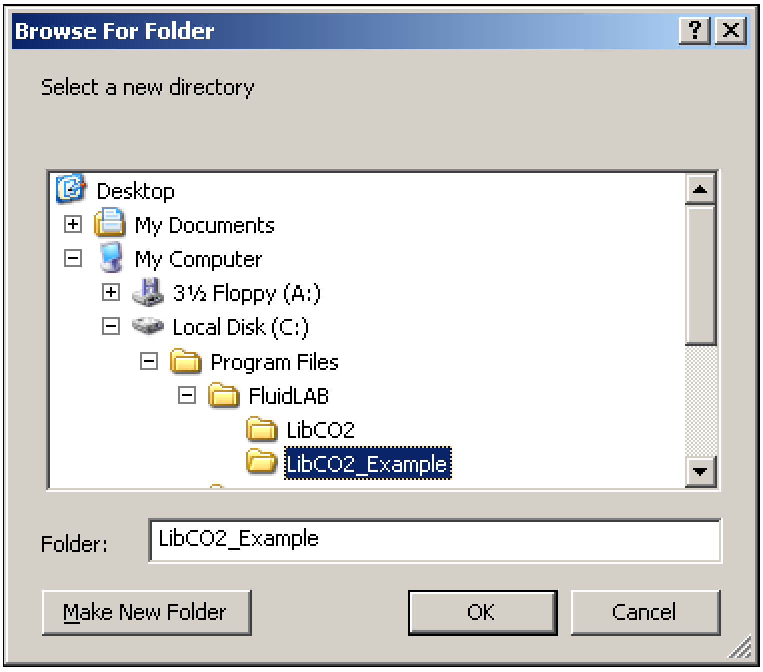
- Start MATLAB (if you did not start it before).
- Click the button marked in the following image in order to open the folder "\LibCO2_Example" in the window "Current Directory".



- Search and click the directory

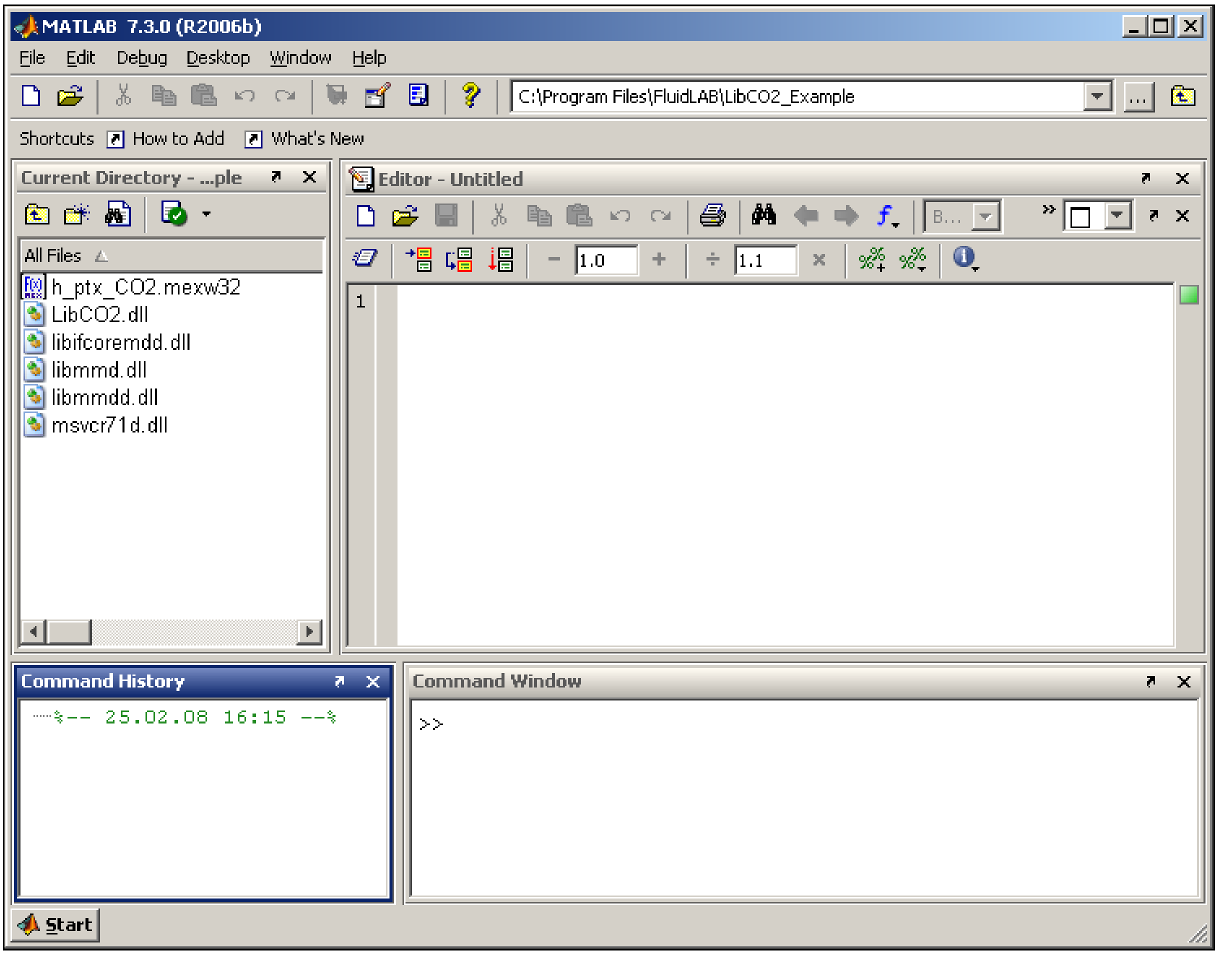
  C:\Program Files\FluidLAB\LibCO2_Example (for English version of Windows)
  C:\Programme\FluidLAB\LibCO2_Example (for German version of Windows)).

in the menu that appears (see figure below).



- Confirm your selection by clicking the "OK" button.
- First of all you need to create a M–File in MATLAB. Within MATLAB click "File", then select "New" and afterwards click "M-File".
- If the "Editor" window appears as a separate window, you can embed it into MATLAB by clicking the arrow (see the following image) in order to obtain a better view.

- In the following you will see the window "Editor - Untitled".



- Now write the following lines in the window "Editor - Untitled":

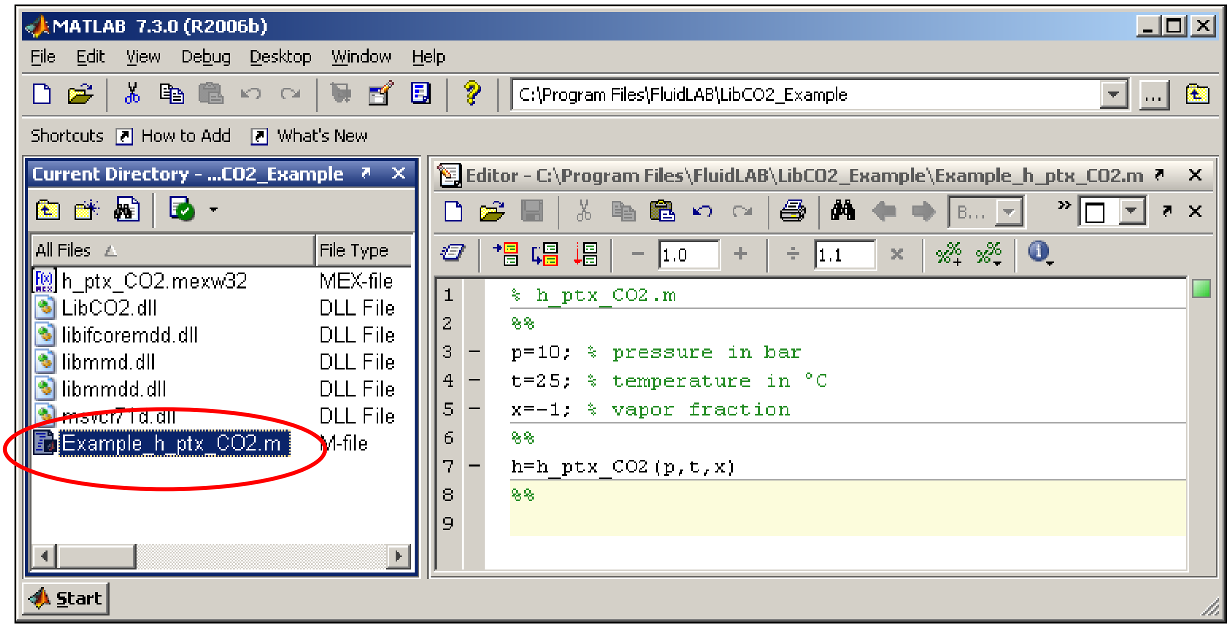
Text to be written:	Explanation:
`% h_ptx_CO2.m`	file name as comment
`%%`	paragraph separation
`p=10; % pressure in bar`	declaration of the variables
`t=25; % temperature in °C`	pressure, temperature and
`x=-1; % vapor fraction`	vapor fraction
`%%`	paragraph separation
`h=h_ptx_CO2(p,t,x)`	function call
`%%`	paragraph separation

- Remarks:
  - The program interprets the first line which starts with " % " to be a data description in "Current Directory"
  - Paragraph separations which are mandatory are being realised through " %%". By this, declaration of variables and calculation instructions are also being separated.
  - The words which are printed in green, start with " % " and stand behind the variables are comments. In fact they are not necessary but they are reasonable for your overview and comprehensibility.
  - You have to leave out the semicolons behind the numerical values if you wish to see the result for *h* and the input parameters as well.

The values of the function parameters in their corresponding units stand for:

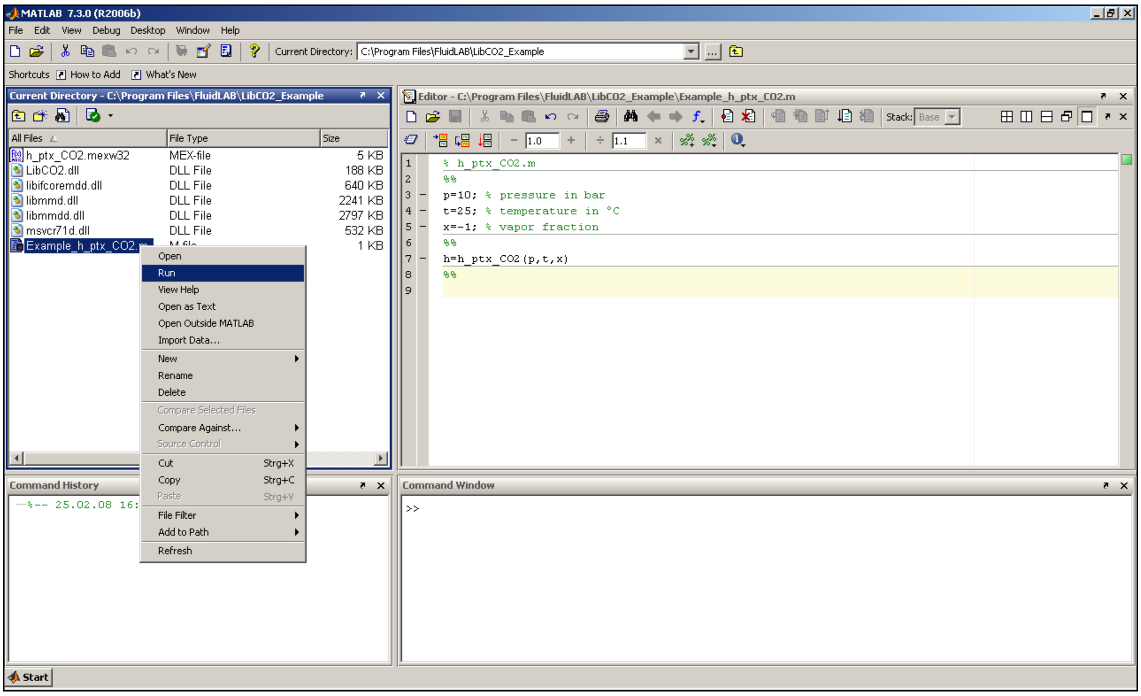
- First operand: Value for p = 10 bar
  (Range of validity: p = 0.001 bar ... 8000 bar)
- Second operand: Value for t = 25 °C
  (Range of validity: $t = t_{mel}$ ... 826.85 °C)
- Third operand: Value for x = –1

- Save the "M-File" by clicking the "File" button and then click "Save As...".
- The menu "Save file as:" pops up; therein the folder name "LibCO2_Example" must be displayed next to "Save in:"
- Next to "File name" you have to type in "Example_h_ptx_CO2.m" and afterwards click the "Save" button.
- You will see the following window:

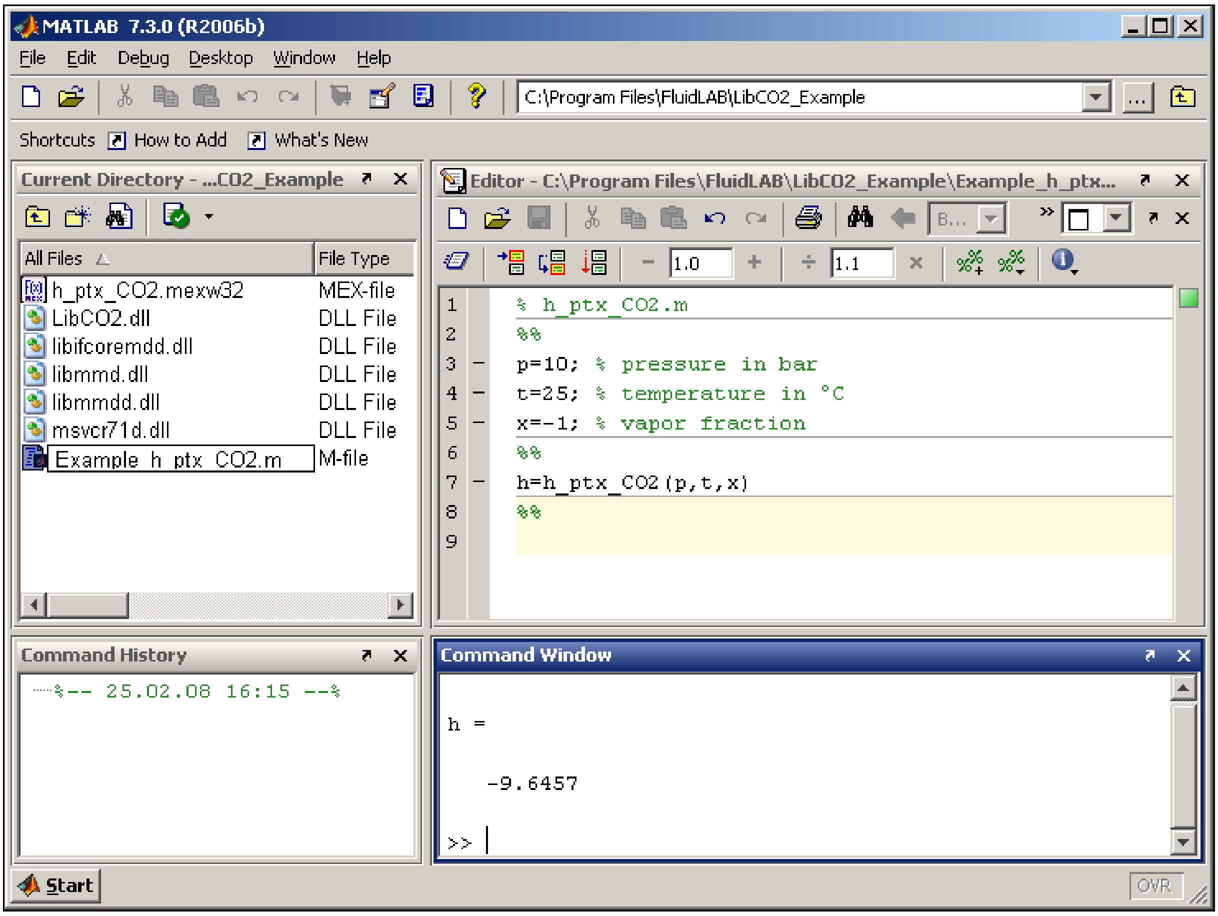


- Within the window "Current Directory" the file "Example_h_ptx_CO2.m" appears.

- Right-click this file and click "Run" in the menu which pops up (see figure below).



- You will see the following window:



In the "Command Window" you will see the result "h = -9.6457 ". The corresponding unit is kJ/kg (cp. table of the property functions in Chapter 1).

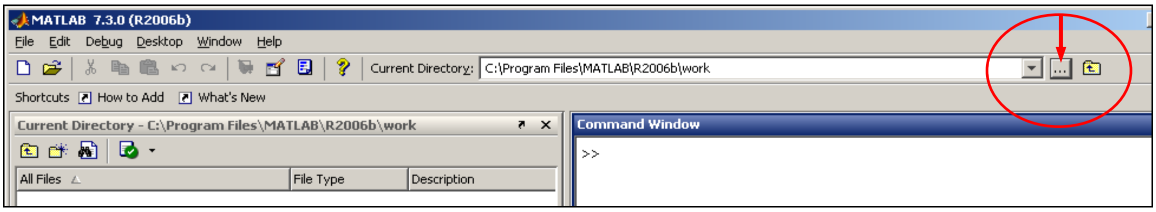
To be able to calculate other values, you have to copy the associated mexw32 or mexw64 files as well because MATLAB can only call functions if they are located in the window "Current Directory". The example calculated can be found in the directory

C:\Program Files\FluidLAB\LibCO2_Example (for English version of Windows)
C:\Programme\FluidLAB\LibCO2_Example (for German version of Windows)

and you may use it as a basis for further calculations with FluidLAB.

2.4 Example: Calculation of the Specific Enthalpy *h* = f(*p*,*t*,*x*) for Carbon Dioxide in Command Window

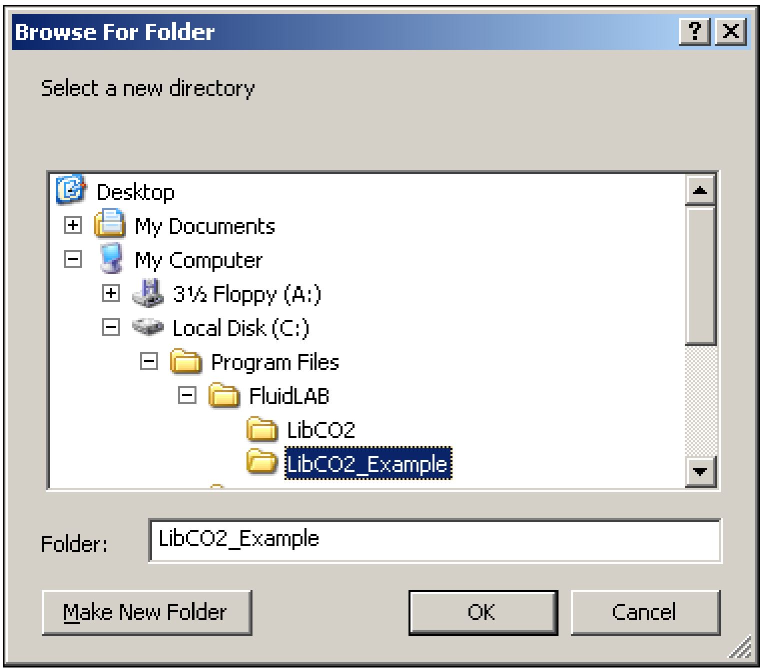
- Please follow the instructions from page 2/7 to 2/9.
- Start MATLAB®.
- Click the button marked in the following image in order to open the folder "\LibCO2_Example" in the window "Current Directory".



- Search and click the directory

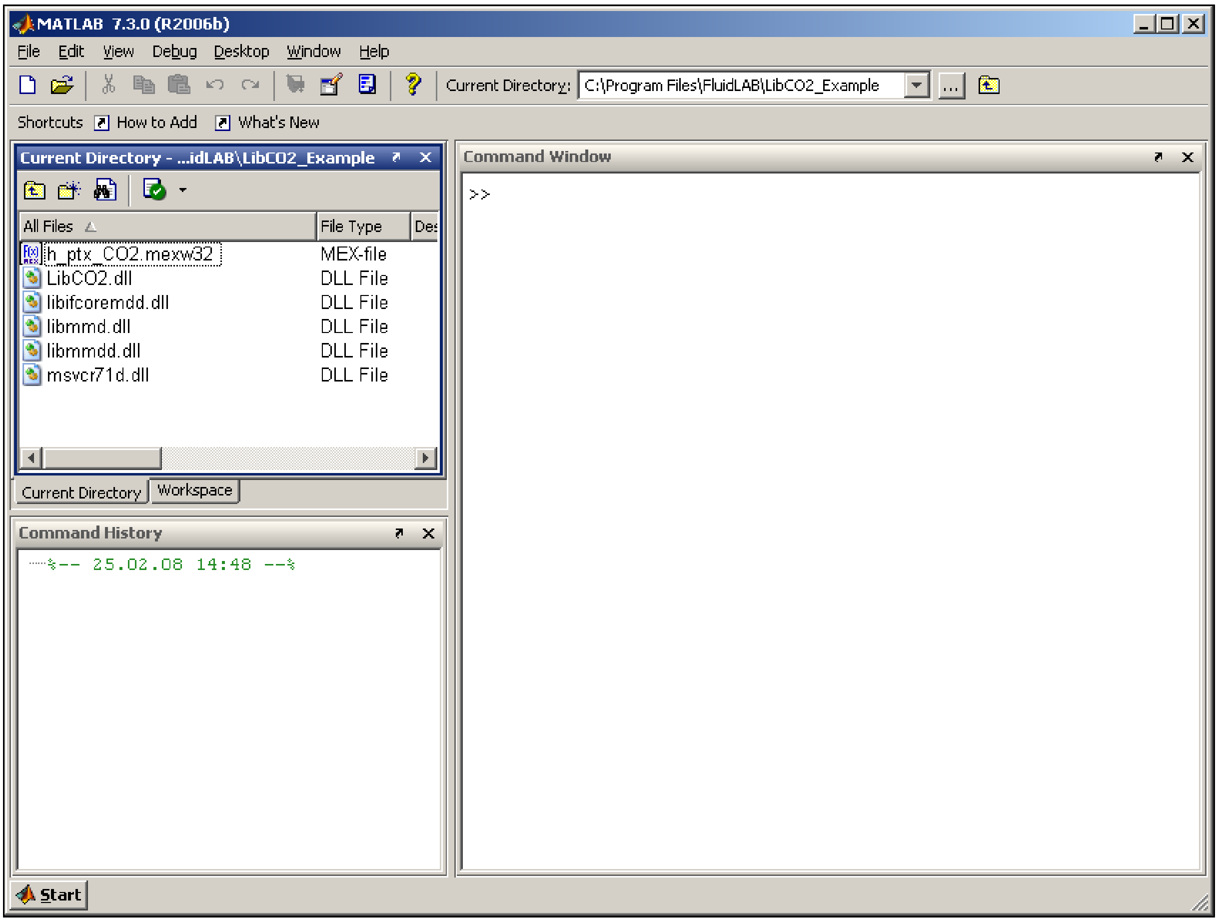
  C:\Program Files\FluidLAB\LibCO2_Example (for English version of Windows)
  C:\Programme\FluidLAB\LibCO2_Example (for German version of Windows)

  in the appearing menu (see the following image).



- Confirm your selection by clicking the "OK" button.

- You will see the following window:



Corresponding to the table of property functions in Chapter 1 you have to call the function "h_ptx_CO2" as follows for calculating h=f(p,t,x):

- Write "h=h_ptx_CO2(10,25,–1)" within the "Command Window".

  The values of the function parameters in their corresponding units stand for:

  - First operand: Value for p = 10 bar
    (Range of validity: $p = 2.9081875815 \cdot 10^{-10}$ bar ... 8000 bar)
  - Second operand: Value for t = 25 °C
    (Range of validity: $t = t_{mel}$ ... 826.85 °C)
  - Third operand: Value for x = –1

    Enter the value for x in kg saturated steam / kg wet steam

    The wet steam region is calculated automatically by the subprograms. For this purpose the following fixed details on the vapor fraction x are to be considered:

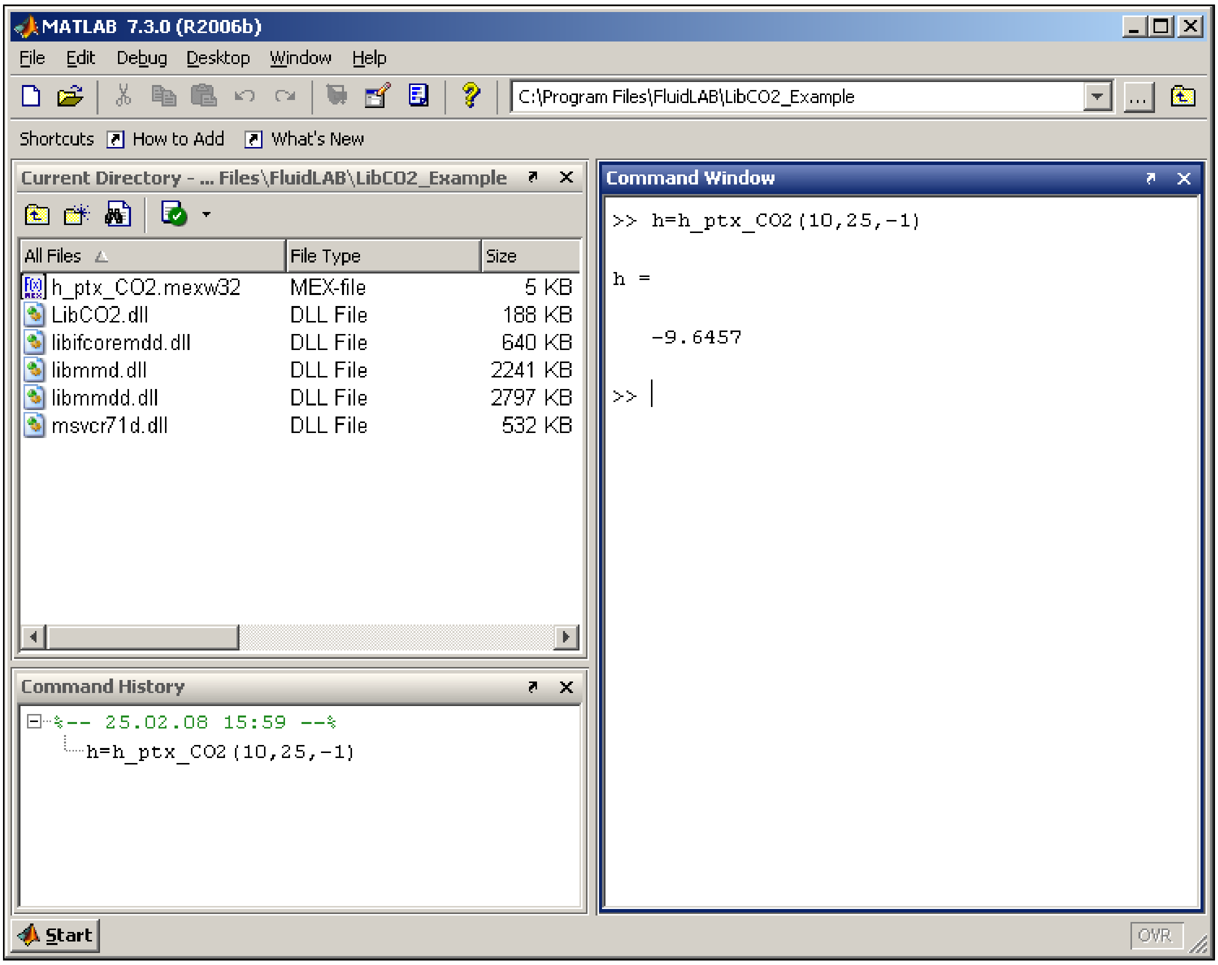
    If the state point to be calculated is located in the single phase region (liquid or superheated steam), i.e. pressure p and temperature t are given, x = –1 must be entered as a pro-forma value.

    If the state point to be calculated is located in the wet steam region, values between 0 and 1 have to be entered for x (the value 0 for saturated liquid, the value 1 for saturated steam).

When calculating wet steam, it is adequate to enter either the given value for *t* <u>and</u> p = –1000, or the given value for *p* <u>and</u> t = –1000, plus the value for *x* between 0 and 1.
If values for both *t* and *p* are entered when calculating wet steam, the program will consider *p* and *t* to be appropriate to represent the saturation-pressure curve. If this is not the case the calculation for the quantity of the chosen function to be calculated results in –1000.

(Saturation line of the CO2: t_t = -56.558 °C ... t_c = 30.9782 °C
p_t = 5.179618369088 bar ... p_c = 73.773 bar)

- Confirm your entry by pressing the "ENTER" button.
- You will see the following window:



In the "Command Window" you will see the result "h = -9.6457 ". The corresponding unit is kJ/kg (cp. table of the property functions in Chapter 1).

To be able to calculate other values, you have to copy the associated mexw32 or mexw64 files into the working directory as well because MATLAB can only call functions if they are located in the window "Current Directory".

2.5 Using FluidLAB with SIMULINK

To use the functions of FluidLAB with the simulation program SIMULINK you have to start SIMULINK in MATLAB® by clicking on Simulink in the upper menu bar shown in Figure 2.19.

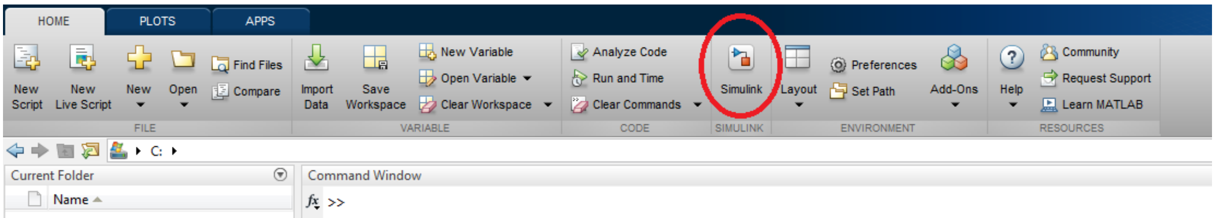


Figure 2.20: Starting Simulink

Then choose a blank model or a simulation in which you would like to use FluidLAB. Now you need to add a MATLAB function block that you can find in the library browser shown in Figure 2.20.

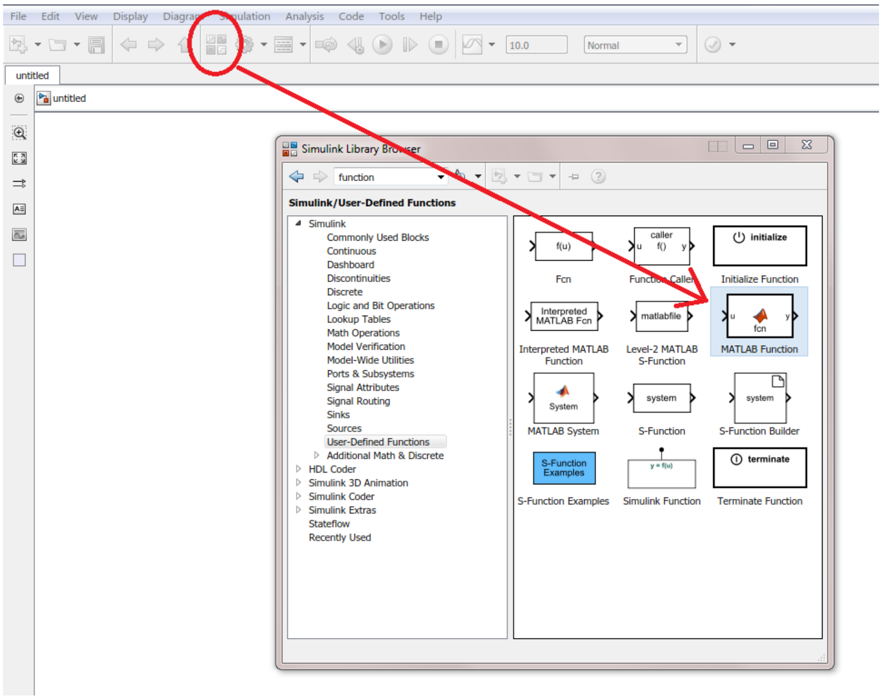


Figure 2.20: Simulink library browser and choosing a MATLAB Function

By dragging and dropping you can drag a Simulink block in your model. The function needs inputs and output that you can find in the Simulink library browser under sources and sinks. For this example constants were taken for the inputs and a display block were taken for outputting.

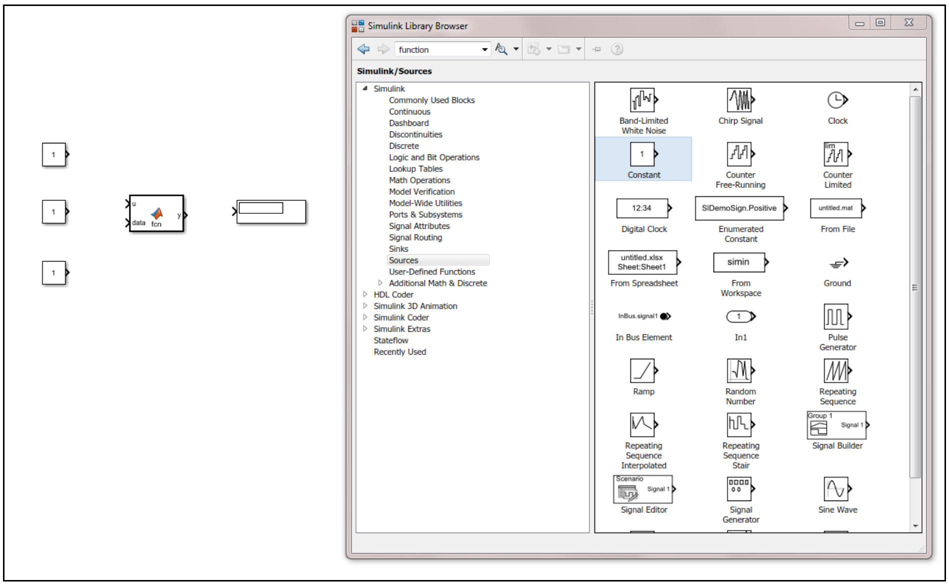


Figure 2.21: Inputs and outputs of the example

Now you have to link inputs and outputs to the MATLAB function block. By pressing and holding the left mouse button on the arrow of a block, you can draw a line and drag it to the MATLAB function block. With this method you can link all blocks together.

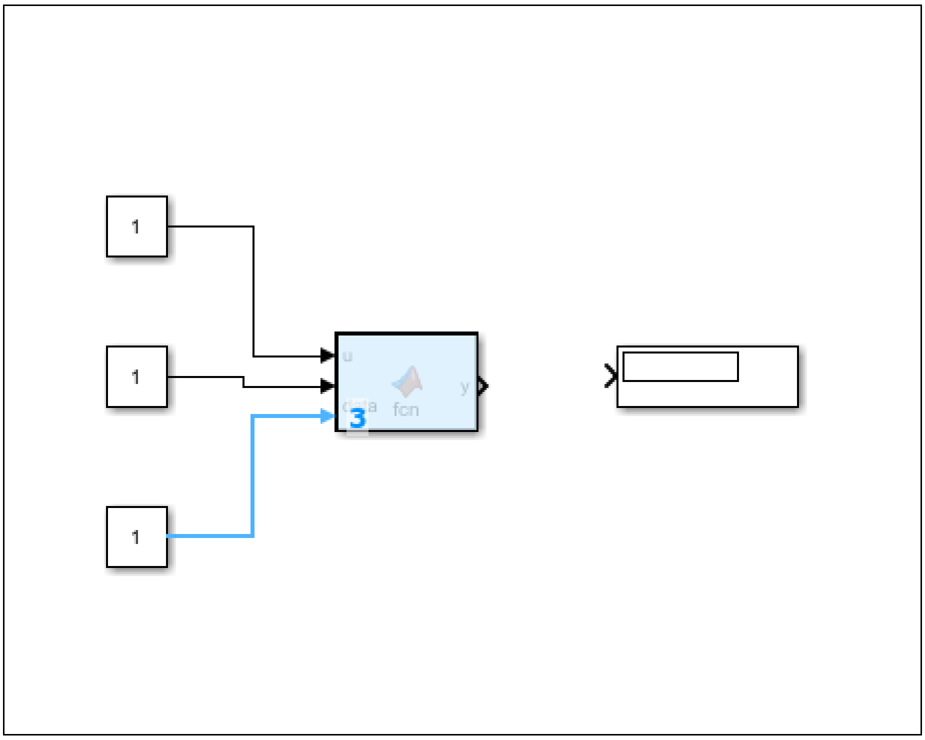


Figure 2.22: Linking blocks in Simulink

You can define the value of a constant block by double-click on them. If you want to calculate the example use the values you can find in section 2.3 and 2.4. With a double-click on the MATLAB function block you can define the function in MATLAB®. The following source code is for the example calculation and the table below describes the source code closer. You can adapt these few lines to call all other function of FluidLAB.

```
function h = fcn(p, t, x)
coder.extrinsic('addpath');
```

```
coder.extrinsic('h_ptx_CO2');
addpath('C:\Program Files\FluidLAB\LibCO2');
h = h_ptx_CO2(p,t,x);
```

Matlab source code	Explanation
`function h = fcn(p, t, x)`	function header, you can define the function name and the inputs like p, t and x of the example
`coder.extrinsic('addpath');`	necessary to add a path
`coder.extrinsic('h_ptx_CO2');`	Choose the function name of the FluidLAB function
`addpath('C:\Program Files\FluidLAB\LibCO2');`	Add the installation path of FluidLAB
`h = h_ptx_CO2(p,t,x);`	Linking the FluidLAB function to the MATLAB function block

You can copy and paste the sourcecode in MATLAB® or write it into the MATLAB® editor. The simulation will start by clicking the run button in Matlab or Simulink and you can see the example in the display block of the simulation which is shown in figure 2.23.

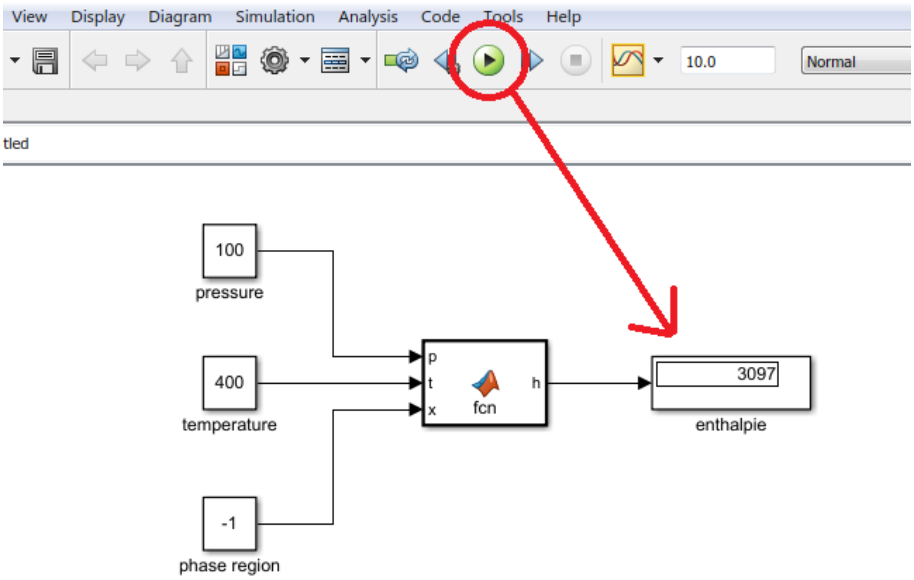


Figure 2.23: Starting the simulation and result of the calculation

Your result is may an other than shown in figure 2.23. If you want to calculate the example please use the values from section 2.2 and 2.3.

2.5 Removing FluidLAB LibCO2

To remove the LibCO2 property library from your hard disk drive in Windows®, click "Start" in the Windows® task bar, select "Settings" and click "Control Panel".

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

In the list box of the "Add or Remove Programs" window that appears select "FluidLAB LibCO2" by clicking on it and click the "Change/Remove" button.

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

Confirm the following menu "Perform Uninstall" by clicking the "Finish" button.

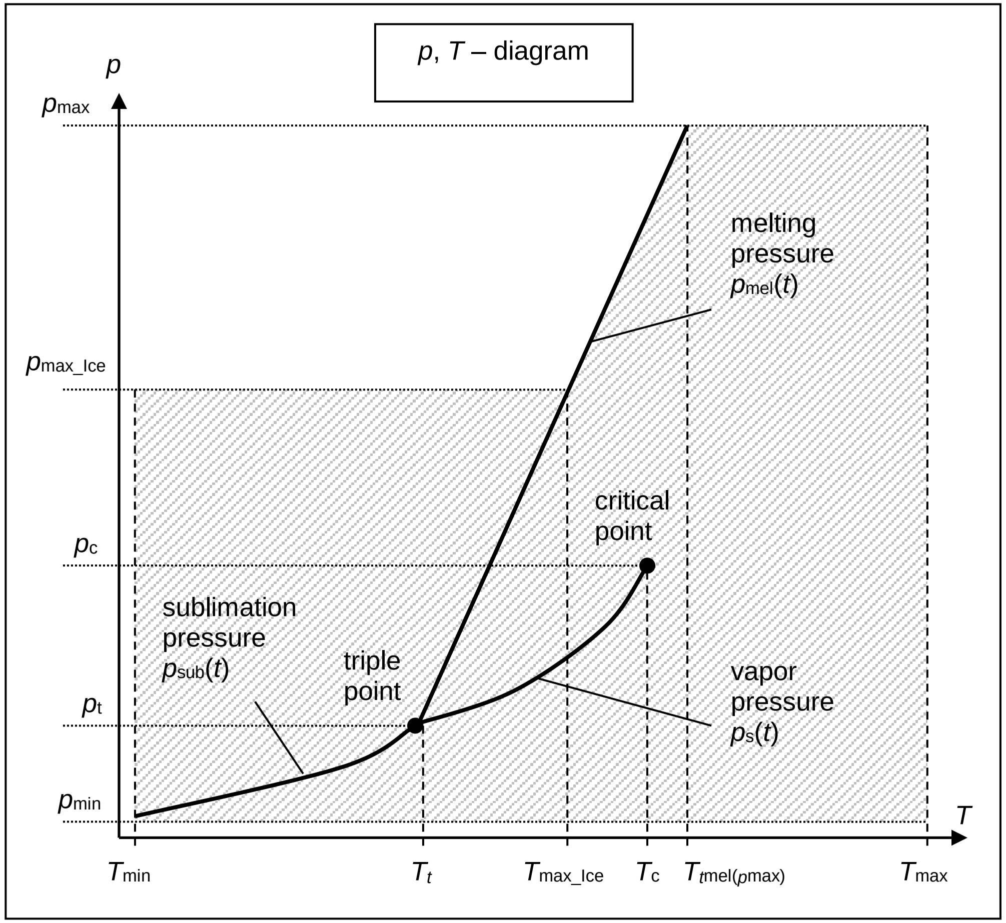
Finally, close the windows "Add or Remove Programs" and "Control Panel" windows.

Now, FluidLAB has been removed.

If there is no other library than LibCO2 installed then the directory "FluidLAB" will be removed as well.

3. Program Documentation

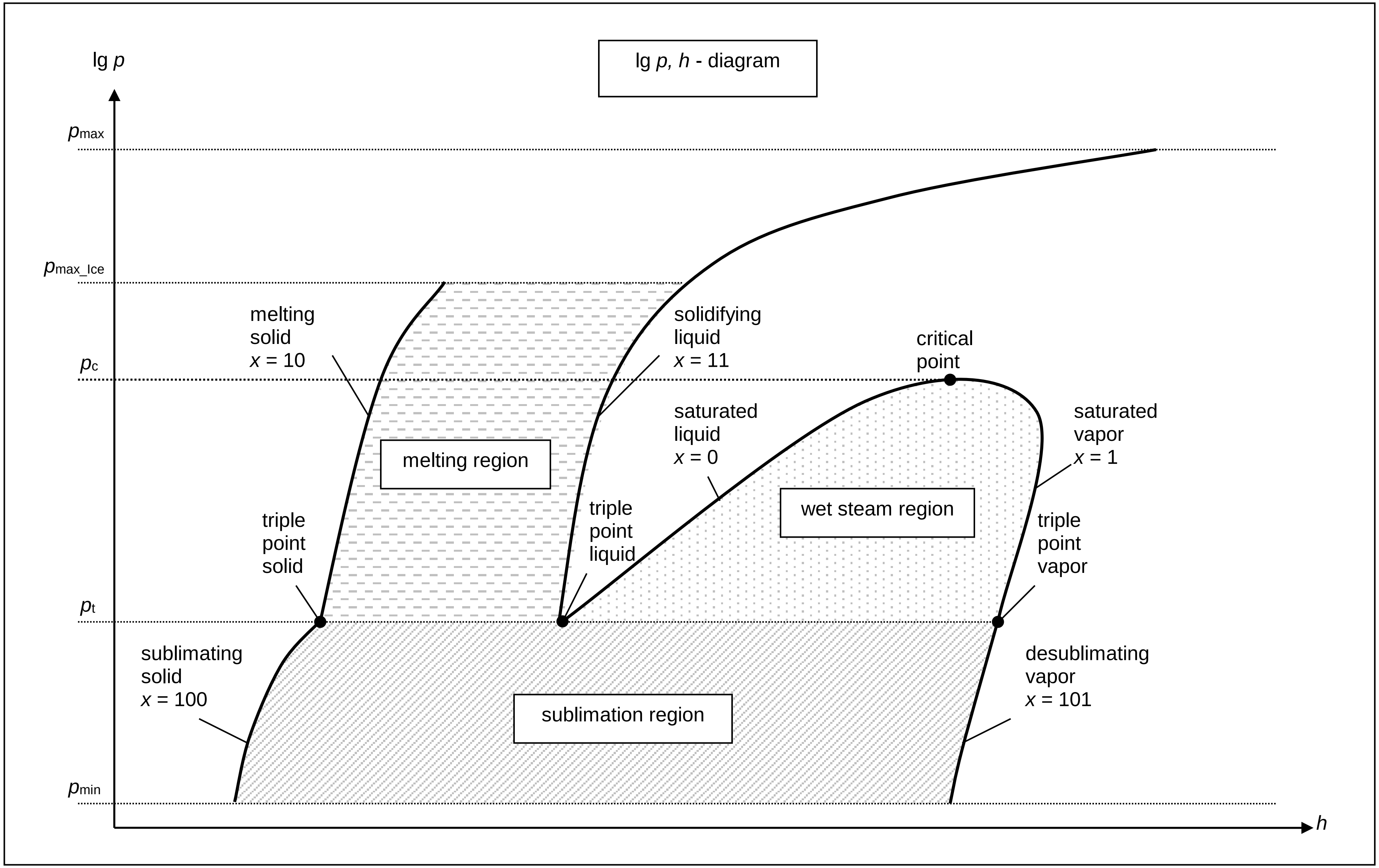
3.1 Ranges of Validity



Reference State:

At p = 1.01325 bar and T = 298.15 K (25 °C) :
h = –0.938457860 kJ/kg and s = –0.00219606205 kJ/(kg K)

Factor	**Abbreviation**	**Value and Unit**
Minimum temperature	T_{min} (t_{min})	85 K (–188.15 °C)
Maximum temperature	T_{max} (t_{max})	1500 K (1226.85 °C)
Triple temperature	T_t (t_t)	216.592 K (–56.558 °C)
Temperature at the critical point	T_c (t_c)	304.1282 K (30.9782 °C)
Maximum temperature of solid region	T_{max_Ice} (t_{miax_Ice})	236.0309 K (–37.119 °C)
Maximum temperature of melting pressure curve	$T_{mel}(p_{max})$ ($t_{mel}(p_{max})$)	327.671 K (54.521 °C)
Minimum pressure	$p_{min} = p_{sub}(T_{min})$	$2.9081875815 \cdot 10^{-10}$ bar
Maximum pressure	p_{max}	8000 bar
Triple pressure	p_t	5.179618369088 bar
Pressure at the critical point	p_c	73.773 bar
Maximum pressure of solid region	p_{max_Ice}	1000 bar


lg p, h - diagram
lg p
p max
p max_Ice
p c
p t
p min
h
melting solid x = 10
solidifying liquid x = 11
critical point
saturated liquid x = 0
saturated vapor x = 1
melting region
wet steam region
triple point solid
triple point liquid
triple point vapor
sublimating solid x = 100
desublimating vapor x = 101
sublimation region

3.2 General Property Functions

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

Function Name: **a_ptx_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION APTXCO2(P,T,X)
REAL*8 P,T,X

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_APTXCO2(A,P,T,X)
REAL*8 A,P,T,X

Input Values

P – Pressure *p* in bar

T - Temperature *t* in °C

X - *x* in kg / kg (Phase fraction, see the following explanations)

Result

APTXCO2, **A** or **a_ptx_CO2** – Thermal diffusivity $a = \frac{\lambda * v}{c_p}$ in m²/s

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Temperature range: from t_{min} to t_{max}
Pressure range: from p_{min} to p_{max} for liquid and steam
from p_{min} to p_{max_Ice} for solid

Details on the phase fraction *x*

If the state point to be calculated is located in the single phase region (solid, liquid or superheated steam), x= -1 must be entered as a pro-forma value.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg *p*,*h*-diagram chapter 3.1):

1. Wet Vapor Region $(0 \le x \le 1)$:

The phase fraction *x* equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
The calculation for *x* values between 0 and 1 is not possible.
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for *x*. In case of dry saturated steam (dew curve) x = 1 has to be entered.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 0 or 1 for *x* must be entered when calculating boiling liquid or dry saturated steam If *p* and *t* and *x* are entered, the program will consider *p* and *t* to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \le x \le 11)$:

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
The calculation for *x* values between 10 and 11 is not possible.
If melting solid (melting curve) is to be calculated, the value 10 has to be entered for *x*. In case of solidifying liquid (solidification curve) x = 11 has to be entered.
When calculating a melting solid or solidifying liquid it is adequate, Concerning pressure and temperature, to put in either the value given for *t* and p = -1000 or the value given for *p* and t = -1000 and the value for x (x = 10 or x = 11). If *p* and *t* and *x* are entered as given values, the program tests whether *p* and *t* fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}
Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}
Pressure range from p_t to p_{max}

Only the solidification curve (x = 11) is calculated for pressures *p* for which applies $p_{max\_Ice} < p \leq p_{max}$.

3. Sublimation Region $(100 \leq x \leq 101)$:

The phase fraction x equates to the vapour fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
The calculation for *x* values between 100 and 101 is not possible.
If sublimating solid is to be calculated, the value 100 has to be entered for *x*. In case of desublimating steam x = 101 has to be entered.
When calculating a sublimating solid or desublimating steam it is adequate, Concerning pressure and temperature, to put in either the value given for *t* and p = -1000 or the value given for *p* and t = -1000 and the value for x (x = 100 or x = 101). If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **APTXCO2 = –1000**, **A = –1000** or **a_ptx_CO2 = –1000** for input values:

Single phase region:
Solid, liquid, overheated steam (x = –1):
- at $p < p_{min}$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at Feststoff
- at $t < t_{min}$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at Feststoff

Two phase regions:

Wet steam region ($0 \leq x \leq 1$):
- at $0 < x < 1$, i.e. calculation in the wet steam region not possible!
- at $p = –1000$ and $t < t_t$ or $t > t_c$
- at $t = –1000$ and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \leq x \leq 11$):
- at $10 < x < 11$, i.e. calculation in the melting region not possible!
- at $p = –1000$ and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at $t = –1000$ and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ *or* $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \leq x \leq 101$):
- at $100 < x < 101$, i.e. calculation in the sublimation region not possible!
- at $p = –1000$ and $t < t_{min}$ or $t > t_t$
- at $t = –1000$ and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ *or* $t > t_t$

References: [2], [3]

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

Function Name: **cp_ptx_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION CPPTXCO2(P,T,X)
REAL*8 P,T,X

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_CPPTXCO2(CP,P,T,X)
REAL*8 CP,P,T,X

Input Values

P – Pressure p in bar

T - Temperature t in °C

X - x in kg / kg (Phase fraction, see the following explanations)

Result

CPPTXCO2, **CP** or **cp_ptx_CO2** - specific isobaric heat capacity c_p in kJ/(kg K)

Range of Validity (cp. p,t-diagram in chapter 3.1)

Temperature range: from t_{min} to t_{max}
Pressure range: from p_{min} to p_{max} for liquid and steam
from p_{min} to p_{max_Ice} for solid

Details on the phase fraction x

If the state point to be calculated is located in the single phase region (solid, liquid or superheated steam), x= -1 must be entered as a pro-forma value.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg p,h-diagram chapter 3.1):

1. Wet Vapor Region $(0 \le x \le 1)$:

The phase fraction x equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
The calculation for x values between 0 and 1 is not possible.
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for x. In case of dry saturated steam (dew curve) x = 1 has to be entered.
Concerning pressure and temperature either the given value for t and p = -1000 or the given value for p and t = -1000 and in both cases the value 0 or 1 for x must be entered when calculating boiling liquid or dry saturated steam If p and t and x are entered, the program will consider p and t to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \le x \le 11)$:

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
The calculation for x values between 10 and 11 is not possible.
If melting solid (melting curve) is to be calculated, the value 10 has to be entered for x. In case of solidifying liquid (solidification curve) x = 11 has to be entered.
When calculating a melting solid or solidifying liquid it is adequate, Concerning pressure and temperature, to put in either the value given for t and p = -1000 or the value given for p and t = -1000 and the value for x (x = 10 or x = 11). If p and t and x are entered as given values, the program tests whether p and t fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}
Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}

Pressure range from p_t to p_{max}

Only the solidification curve (x = 11) is calculated for pressures p for which applies $p_{max\_Ice} < p \leq p_{max}$.

3. Sublimation Region $(100 \leq x \leq 101)$:

The phase fraction x equates to the vapor fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
The calculation for x values between 100 and 101 is not possible.
If sublimating solid is to be calculated, the value 100 has to be entered for x. In case of desublimating steam x = 101 has to be entered.
When calculating a sublimating solid or desublimating steam it is adequate, Concerning pressure and temperature, to put in either the value given for t and p = -1000 or the value given for p and t = -1000 and the value for x (x = 100 or x = 101). If p and t and x are entered as given values the program tests whether p and t fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **CPPTXCO2 = –1000**, **CP = –1000** or **cp_ptx_CO2 = –1000** for input values:

Single phase region:

Solid, liquid, overheated steam (x = –1):
- at $p < p_{min}$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at $t < t_{min}$ or $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Two phase regions:

Wet steam region ($0 \leq x \leq 1$):
- at $0 < x < 1$, i.e. calculation in the wet steam region not possible!
- at p = –1000 and $t < t_t$ or $t > t_c$
- at t = –1000 and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \leq x \leq 11$):
- at $10 < x < 11$, i.e. calculation in the melting region not possible!
- at p = –1000 and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at t = –1000 and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \leq x \leq 101$):
- at $100 < x < 101$, i.e. calculation in the sublimation region not possible!
- at p = –1000 and $t < t_{min}$ or $t > t_t$
- at t = –1000 and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ or $t > t_t$

References: [2]

Dynamic Viscosity η = f(*p*,*t*,*x*)

Function Name: **eta_ptx_CO2**

Subprogram with value of the function: **REAL*8 FUNCTION ETAPTCO2(P,T,X)**
for call from Fortran REAL*8 P,T,X

Subprogram with parameter: **INTEGER*4 FUNCTION C_ETAPTCO2(ETA,P,T,X)**
for call from the DLL REAL*8 ETA,P,T,X

Input Values

P – Pressure *p* in bar

T - Temperature *t* in °C

X - *x* in kg / kg (Phase fraction, see the following explanations)

Result

ETAPTXCO2, **ETA** or **eta_ptx_CO2** – dynamic viscosity η in Pa s

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Temperature range: from t_t to t_{max}
Pressure range: from p_{min} to p_{max} for liquid and steam

Details on the phase fraction *x*

If the state point to be calculated is located in the single phase region (liquid or superheated steam), x= -1 must be entered as a pro-forma value. Calculating solid is not possible.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg *p*,*h*-diagram chapter 3.1):

1. Wet Vapor Region $(0 \leq x \leq 1)$:

The phase fraction *x* equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
The calculation for *x* values between 0 and 1 is not possible.
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for *x*. In case of dry saturated steam (dew curve) x = 1 has to be entered.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 0 or 1 for *x* must be entered when calculating boiling liquid or dry saturated steam. If *p* and *t* and *x* are entered, the program will consider *p* and *t* to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \leq x \leq 11)$:

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
The calculation for *x* values of x = 10 (melting curve) and between 10 and 11 is not possible.
If solidifying liquid (solidification curve) is to be calculated, the value 11 has to be entered for *x*.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 11 for *x* must be entered when calculating solidifying liquid.
If *p* and *t* and *x* are entered as given values, the program tests whether *p* and *t* fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}
Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}

Pressure range from p_t to p_{max}

3. Sublimation Region $(100 \leq x \leq 101)$:

The phase fraction x equates to the vapor fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
The calculation for *x* values of x = 100 (sublimation curve) and between 100 and 101 is not possible.
If desublimating steam (desublimation curve) is to be calculated, the value 101 has to be entered for *x*.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 101 for *x* must be entered when calculating desublimating steam. If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **ETAPTXCO2 = –1000**, **ETA = –1000** or **eta_ptx_CO2 = –1000** for input values:

Single phase region:

Liquid and overheated steam (x = –1):
- at $p < p_{min}$ or $p > p_{max}$
- t $t < t_{mt}$ or $t > t_{max}$

Two phase regions:

Wet steam region ($0 \leq x \leq 1$):
- at $0 < x < 1$, i.e. calculation in the wet steam region not possible!
- at $p = -1000$ and $t < t_t$ or $t > t_c$
- at $t = -1000$ and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \leq x \leq 11$):
- at $10 < x < 11$, i.e. calculation on the melting curve and in the melting region not possible!
- at $p = -1000$ and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at $t = -1000$ and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \leq x \leq 101$):
- at $100 \leq x < 101$, i.e. calculation on the sublimation curve and in the sublimation region not possible!
- at $p = -1000$ and $t < t_{min}$ or $t > t_t$
- at $t = -1000$ and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ or $t > t_t$

References: [2], [3]

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

Function Name: **h_ptx_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION HPTXCO2(P,T,X)
REAL*8 P,T,X

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_HPTXCO2(H,P,T,X)
REAL*8 H,P,T,X

Input Values

P – Pressure *p* in bar

T - Temperature *t* in °C

X - *x* in kg / kg (Phase fraction, see the following explanations)

Result

HPTXCO2, **H** or **h_ptx_CO2** - specific enthalpy *h* in kJ/kg

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Temperature range: from t_{min} to t_{max}
Pressure range: from p_{min} to p_{max} for liquid and steam
from p_{min} to p_{max_Ice} for solid

Details on the phase fraction *x*

If the state point to be calculated is located in the single phase region (solid, liquid or superheated steam), x= -1 must be entered as a pro-forma value.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg *p*,*h*-diagram chapter 3.1):

1. Wet Vapor Region $(0 \le x \le 1)$ **:**

The phase fraction *x* equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
If the state point to be calculated is located in the wet steam region, a value between 0 and 1 must be entered for x.
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for *x*. In case of dry saturated steam (dew curve) x = 1 has to be entered.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 0 or 1 for *x* must be entered when calculating boiling liquid or dry saturated steam. If *p* and *t* and *x* are entered, the program will consider *p* and *t* to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \le x \le 11)$ **:**

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
If the state point to be calculated is located in the melting region, a value between 10 and 11 must be entered for x.
If melting solid (melting curve) is to be calculated, the value 10 has to be entered for *x*. In case of solidifying liquid (solidification curve) x = 11 has to be entered.
When calculating a melting solid or solidifying liquid it is adequate, Concerning pressure and temperature, to put in either the value given for *t* and p = -1000 or the value given for *p* and t = -1000 and the value for x (x = 10 or x = 11). If *p* and *t* and *x* are entered as given values, the program tests whether *p* and *t* fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}
Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}
Pressure range from p_t to p_{max}

Only the solidification curve (x = 11) is calculated for pressures *p* for which applies $p_{max\_Ice} < p \le p_{max}$.

3. Sublimation Region $(100 \le x \le 101)$:

The phase fraction x equates to the vapor fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
If the state point to be calculated is located in the melting region, a value between 100 and 101 must be entered for x.
If sublimating solid is to be calculated, the value 100 has to be entered for *x*. In case of desublimating steam x = 101 has to be entered.
When calculating a sublimating solid or desublimating steam it is adequate, Concerning pressure and temperature, to put in either the value given for *t* and p = -1000 or the value given for *p* and t = -1000 and the value for x (x = 100 or x = 101). If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **HPTXCO2 = –1000**, **H = –1000** or **h_ptx_CO2 = –1000** for input values:

Single phase region:
Solid, liquid, overheated steam (x = –1):
- at $p < p_{min}$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at $t < t_{min}$ or $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Two phase regions:
Wet steam region ($0 \le x \le 1$):
- at $p = -1000$ and $t < t_t$ or $t > t_c$
- at $t = -1000$ and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \le x \le 11$):
- at $p = -1000$ and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at $t = -1000$ and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \le x \le 101$):
- at $p = -1000$ and $t < t_{min}$ or $t > t_t$
- at $t = -1000$ and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ or $t > t_t$

References: [2]

Isentropic Exponent κ = f(p,t,x)

Function Name: **kappa_ptx_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION KAPTXCO2(P,T,X)
REAL*8 P,T,X

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_KAPTXCO2(KAP,P,T,X)
REAL*8 KAP,P,T,X

Input Values

P – Pressure p in bar

T - Temperature t in °C

X - x in kg / kg (Phase fraction, see the following explanations)

Result

KAP_PTX_CO2, **KAP** or **kappa_ptx_CO2** – Isentropic exponent $\kappa = \frac{w^2}{p * v}$

Range of Validity (cp. p,t-diagram in chapter 3.1)

Temperature range: from t_t to t_{max}
Pressure range: from p_{min} to p_{max} for liquid and steam

Details on the phase fraction x

If the state point to be calculated is located in the single phase region (liquid or superheated steam), x= -1 must be entered as a pro-forma value. Calculating solid is not possible.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg p,h-diagram chapter 3.1):

1. Wet Vapor Region $(0 \leq x \leq 1)$:

The phase fraction x equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
The calculation for x values between 0 and 1 is not possible.
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for x. In case of dry saturated steam (dew curve) x = 1 has to be entered.
Concerning pressure and temperature either the given value for t and p = -1000 or the given value for p and t = -1000 and in both cases the value 0 or 1 for x must be entered when calculating boiling liquid or dry saturated steam. If p and t and x are entered, the program will consider p and t to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \leq x \leq 11)$:

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
The calculation for x values of x = 10 (melting curve) and between 10 and 11 is not possible.
If solidifying liquid (solidification curve) is to be calculated, the value 11 has to be entered for x.
Concerning pressure and temperature either the given value for t and p = -1000 or the given value for p and t = -1000 and in both cases the value 11 for x must be entered when calculating solidifying liquid.
If p and t and x are entered as given values, the program tests whether p and t fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}
Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}

Pressure range from p_t to p_{max}

3. Sublimation Region $(100 \le x \le 101)$:

The phase fraction x equates to the vapor fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
The calculation for *x* values of x = 100 (sublimation curve) and between 100 and 101 is not possible.
If desublimating steam (desublimation curve) is to be calculated, the value 101 has to be entered for *x*.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 101 for *x* must be entered when calculating desublimating steam. If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **KAP_PTX_CO2**, **KAP = –1000** or **kappa_ptx_CO2 = –1000** for input values:

Single phase region:

Liquid and overheated steam (x = –1):

- at $p < p_{min}$ or $p > p_{max}$
- at $t < t_t$ or $t > t_{max}$

Two phase regions:

Wet steam region ($0 \le x \le 1$):

- at $0 < x < 1$, i.e. calculation in the wet steam region not possible!
- at $p = –1000$ and $t < t_t$ or $t > t_c$
- at $t = –1000$ and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \le x \le 11$):

- at $10 < x < 11$, i.e. calculation on the melting curve and in the melting region not possible!
- at $p = –1000$ and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at $t = –1000$ and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \le x \le 101$):

- at $100 \le x < 101$, i.e. calculation on the sublimation curve and in the sublimation region not possible!
- at $p = –1000$ and $t < t_{min}$ or $t > t_t$
- at $t = –1000$ and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ or $t > t_t$

References: [2]

Thermal Conductivity λ = f(p,t,x)

Function Name:	**lambda_ptx_CO2**
Subprogram with value of the function: for call from Fortran	**REAL*8 FUNCTION LAMPTCO2(P,T,X)** REAL*8 P,T,X
Subprogram with parameter: for call from the DLL	**INTEGER*4 FUNCTION C_LAMPTXCO2(LAM,P,T,X)** REAL*8 LAM,P,T,X

Input Values

P – Pressure p in bar

T - Temperature t in °C

X - x in kg / kg (Phase fraction, see the following explanations)

Result

LAMPTXCO2, **LAM** or **lambda_ptx_CO2** – Thermal conductivity λ in W/m K

Range of Validity (cp. p,t-diagram in chapter 3.1)

Temperature range: from t_{min} to t_{max}
Pressure range: from p_{min} to p_{max} for liquid and steam
from p_{min} to p_{max_Ice} for solid

Details on the phase fraction *x*

If the state point to be calculated is located in the single phase region (solid, liquid or superheated steam), x= -1 must be entered as a pro-forma value.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg p,h-diagram chapter 3.1):

1. Wet Vapor Region $(0 \le x \le 1)$:

The phase fraction x equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
The calculation for x values between 0 and 1 is not possible.
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for x. In case of dry saturated steam (dew curve) x = 1 has to be entered.
Concerning pressure and temperature either the given value for t and p = -1000 or the given value for p and t = -1000 and in both cases the value 0 or 1 for x must be entered when calculating boiling liquid or dry saturated steam. If p and t and x are entered, the program will consider p and t to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \le x \le 11)$:

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
The calculation for x values between 10 and 11 is not possible.
If melting solid (melting curve) is to be calculated, the value 10 has to be entered for x. In case of solidifying liquid (solidification curve) x = 11 has to be entered.
When calculating a melting solid or solidifying liquid it is adequate, Concerning pressure and temperature, to put in either the value given for t and p = -1000 or the value given for p and t = -1000 and the value for x (x = 10 or x = 11). If p and t and x are entered as given values, the program tests whether p and t fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}
Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}
Pressure range from p_t to p_{max}

Only the solidification curve (x = 11) is calculated for pressures p for which applies $p_{max\_Ice} < p \leq p_{max}$.

3. Sublimation Region $(100 \leq x \leq 101)$:

The phase fraction x equates to the vapor fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
The calculation for x values between 100 and 101 is not possible.
If sublimating solid is to be calculated, the value 100 has to be entered for x. In case of desublimating steam x = 101 has to be entered.
When calculating a sublimating solid or desublimating steam it is adequate, Concerning pressure and temperature, to put in either the value given for t and p = -1000 or the value given for p and t = -1000 and the value for x (x = 100 or x = 101). If p and t and x are entered as given values the program tests whether p and t fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **LAMPTXCO2 = - 1000**, **LAM = -1000** or **lambda_ptx_CO2 = - 1000** for input values:

Single phase region:

Solid, liquid, overheated steam (x = –1):
- at $p < p_{min}$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at $t < t_{min}$ or $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Two phase regions:

Wet steam region ($0 \leq x \leq 1$):
- at $0 < x < 1$, i.e. calculation in the wet steam region not possible!
- at p = –1000 and $t < t_t$ or $t > t_c$
- at t = –1000 and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \leq x \leq 11$):
- at $10 < x < 11$, i.e. calculation in the melting region not possible!
- at p = –1000 and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at t = –1000 and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \leq x \leq 101$):
- at $100 < x < 101$, i.e. calculation in the sublimation region not possible!
- at p = –1000 and $t < t_{min}$ or $t > t_t$
- at t = –1000 and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ or $t > t_t$

References: [2], [3]

Kinematic Viscosity ν = f(p,t,x)

Function Name: **ny_ptx_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION NYPTXCO2(P,T,X)
REAL*8 P,T,X

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_NYPTXCO2(NY,P,T,X)
REAL*8 NY,P,T,X

Input Values

P – Pressure p in bar

T - Temperature t in °C

X - x in kg / kg (Phase fraction, see the following explanations)

Result

NYPTXCO2, **NY** or **ny_ptx_CO2** – Kinematic viscosity $\nu = \eta * v$ in m^2/s

Range of Validity (cp. p,t-diagram in chapter 3.1)

Temperature range: from t_t to t_{max}
Pressure range: from p_{min} to p_{max} for liquid and steam

Details on the phase fraction x

If the state point to be calculated is located in the single phase region (liquid or superheated steam), x= -1 must be entered as a pro-forma value. Calculating solid is not possible.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg p,h-diagram chapter 3.1):

1. Wet Vapor Region $(0 \le x \le 1)$:

The phase fraction x equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
The calculation for x values between 0 and 1 is not possible.
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for x. In case of dry saturated steam (dew curve) x = 1 has to be entered.
Concerning pressure and temperature either the given value for *t* <u>and</u> *p* = -1000 or the given value for p <u>and</u> t = -1000 and in both cases the value 0 or 1 for x must be entered when calculating boiling liquid or dry saturated steam. If p <u>and</u> t <u>and</u> x are entered, the program will consider p and t to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \le x \le 11)$:

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
The calculation for x values of x = 10 (melting curve) and between 10 and 11 is not possible.
If solidifying liquid (solidification curve) is to be calculated, the value 11 has to be entered for x.
Concerning pressure and temperature either the given value for t <u>and</u> p = -1000 or the given value for p <u>and</u> t = -1000 and in both cases the value 11 for x must be entered when calculating solidifying liquid**.**
If p <u>and</u> t <u>and</u> x are entered as given values, the program tests whether p and t fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}
Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}

Pressure range from p_t to p_{max}

3. Sublimation Region $(100 \le x \le 101)$:

The phase fraction x equates to the vapor fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
The calculation for *x* values of x = 100 (sublimation curve) and between 100 and 101 is not possible.
If desublimating steam (desublimation curve) is to be calculated, the value 101 has to be entered for *x*.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 101 for *x* must be entered when calculating desublimating steam. If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **NYPTXCO2 = –1000**, **NY = –1000** or **ny_ptx_CO2 = –1000** for input values:

Single phase region:

Liquid and overheated steam (x = –1):
- at $p < p_{min}$ or $p > p_{max}$
- at $t < t_t$ or $t > t_{max}$

Two phase regions:

Wet steam region ($0 \le x \le 1$):
- at $0 < x < 1$, i.e. calculation in the wet steam region not possible!
- at $p = –1000$ and $t < t_t$ or $t > t_c$
- at $t = –1000$ and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \le x \le 11$):
- at $10 < x < 11$, i.e. calculation on the melting curve and in the melting region not possible!
- at $p = –1000$ and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at $t = –1000$ and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \le x \le 101$):
- at $100 \le x < 101$, i.e. calculation on the sublimation curve and in the sublimation region not possible!
- at $p = –1000$ and $t < t_{min}$ or $t > t_t$
- at $t = –1000$ and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ or $t > t_t$

References: [2], [3]

Melting Pressure p_{mel} = f(t)

Function Name: **pmel_t_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION PMELCO2(T)
REAL*8 T

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_PMELCO2(PMEL,T)
REAL*8 PMEL,P,T,X

Input Values

T - Temperature t in °C

Result

PMELCO2, **PMEL** or **pmel_t_CO2** – Melting pressure p_{mel} in bar

Range of Validity (cp. p,t-diagram in chapter 3.1)

Temperature range: from t_t to $t_{mel}(p_{max})$

Results for wrong input values

Result **PMELCO2 = –1000**, **PMEL = –1000** or **pmel_t_CO2 = –1000** for input values:

- at $t < t_t$ or $t > t_{mel}(p_{max})$

References: [2]

Sublimation Pressure p_{sub} = f(t)

Function Name: **psub_t_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION PSUBCO2(T)
REAL*8 T

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_PSUBCO2(PSUB,T)
REAL*8 PSUB,P,T,X

Input Values

T - Temperature t in °C

Result

PSUBCO2, **PSUB** or **psub_t_CO2** – Sublimation pressure p_{sub} in bar

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Temperature range: from t_{min} to t_t

Results for wrong input values

Result **PSUBCO2 = –1000**, **PSUB = –1000** or **psub_t_CO2 = –1000** for input values:

- at $t < t_{min}$ or $t > t_c$

References: [2]

Prandtl-Number *Pr* = f(*p*,*t*,*x*)

Function Name: **Pr_ptx_CO2**

Subprogram with value of the function: for call from Fortran — **REAL*8 FUNCTION PRPTXCO2(P,T,X)**
REAL*8 P,T,X

Subprogram with parameter: for call from the DLL — **INTEGER*4 FUNCTION C_PRPTXCO2(PR,P,T,X)**
REAL*8 PR,P,T,X

Input Values

P – Pressure p in bar

T - Temperature t in °C

X - x in kg / kg (Phase fraction, see the following explanations)

Result

PRPTXCO2, **PR** or **Pr_ptx_CO2** – Prandtl-Number $Pr = \frac{\eta * c_p}{\lambda}$

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Temperature range: from t_t to t_{max}

Pressure range: from p_{min} to p_{max} for liquid and steam

Details on the phase fraction *x*

If the state point to be calculated is located in the single phase region (liquid or superheated steam), x= -1 must be entered as a pro-forma value. Calculating solid is not possible.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg *p*,*h*-diagram chapter 3.1):

1. Wet Vapor Region $(0 \le x \le 1)$:

The phase fraction x equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
The calculation for x values between 0 and 1 is not possible.
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for x. In case of dry saturated steam (dew curve) x = 1 has to be entered.
Concerning pressure and temperature either the given value for t and p = -1000 or the given value for p and t = -1000 and in both cases the value 0 or 1 for x must be entered when calculating boiling liquid or dry saturated steam. If p and t and x are entered, the program will consider p and t to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \le x \le 11)$:

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
The calculation for x values of x = 10 (melting curve) and between 10 and 11 is not possible.
If solidifying liquid (solidification curve) is to be calculated, the value 11 has to be entered for x.
Concerning pressure and temperature either the given value for t and p = -1000 or the given value for p and t = -1000 and in both cases the value 11 for x must be entered when calculating solidifying liquid.
If p and t and x are entered as given values, the program tests whether p and t fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}
Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}

Pressure range from p_t to p_{max}

3. Sublimation Region $(100 \le x \le 101)$:

The phase fraction x equates to the vapor fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
The calculation for *x* values of x = 100 (sublimation curve) and between 100 and 101 is not possible.
If desublimating steam (desublimation curve) is to be calculated, the value 101 has to be entered for *x*.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 101 for *x* must be entered when calculating desublimating steam. If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **PRPTXCO2 = –1000**, **PR = –1000** or **Pr_ptx_CO2 = –1000** for input values:

Single phase region:

Liquid and overheated steam (x = –1):
- at $p < p_{min}$ or $p > p_{max}$
- at $t < t_t$ or $t > t_{max}$

Two phase regions:

Wet steam region ($0 \le x \le 1$):
- at $0 < x < 1$, i.e. calculation in the wet steam region not possible!
- at $p = –1000$ and $t < t_t$ or $t > t_c$
- at $t = –1000$ and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \le x \le 11$):
- at $10 < x < 11$, i.e. calculation on the melting curve and in the melting region not possible!
- at $p = –1000$ and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at $t = –1000$ and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \le x \le 101$):
- at $100 \le x < 101$, i.e. calculation on the sublimation curve and in the sublimation region not possible!
- at $p = –1000$ and $t < t_{min}$ or $t > t_t$
- at $t = –1000$ and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ or $t > t_t$

References: [2], [3]

Vapor Pressure p_s = f(t)

Function Name: **ps_t_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION PSTCO2(T)
REAL*8 T

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_PSTCO2(PS,T)
REAL*8 PS,T

Input Values

T - Temperature t in °C

Result

PSTCO2, **PS** or **ps_t_CO2** – Vapor pressure p_s in bar

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Temperature range: from t_t to t_c

Results for wrong input values

Result **PSTCO2 = –1000**, **PS = –1000** or **ps_t_CO2 = –1000** for input values:

- at $t < t_t$ or $t > t_c$

References: [2]

Density ρ = f(p,t,x)

Function Name:	**rho_ptx_CO2**
Subprogram with value of the function: for call from Fortran	**REAL*8 FUNCTION ROPTXCO2(P,T,X)** REAL*8 P,T,X
Subprogram with parameter: for call from the DLL	**INTEGER*4 FUNCTION C_ROPTXCO2(RHO,P,T,X)** REAL*8 RHO,P,T,X

Input Values

P – Pressure *p* in bar

T - Temperature *t* in °C

X - *x* in kg / kg (Phase fraction, see the following explanations)

Result

RHO_PTX_CO2, **RHO** or **rho_ptx_CO2** - Density ρ in kg/m^3

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Temperature range: from t_{min} to t_{max}
Pressure range: from p_{min} to p_{max} for liquid and steam
from p_{min} to p_{max_Ice} for solid

Details on the phase fraction *x*

If the state point to be calculated is located in the single phase region (solid, liquid or superheated steam), x= -1 must be entered as a pro-forma value.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg *p*,*h*-diagram chapter 3.1):

1. Wet Vapor Region $(0 \leq x \leq 1)$:

The phase fraction *x* equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
If the state point to be calculated is located in the wet steam region, a value between 0 and 1 must be entered for x.
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for *x*. In case of dry saturated steam (dew curve) x = 1 has to be entered.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 0 or 1 for *x* must be entered when calculating boiling liquid or dry saturated steam. If *p* and *t* and *x* are entered, the program will consider *p* and *t* to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \leq x \leq 11)$:

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
If the state point to be calculated is located in the melting region, a value between 10 and 11 must be entered for x.
If melting solid (melting curve) is to be calculated, the value 10 has to be entered for *x*. In case of solidifying liquid (solidification curve) x = 11 has to be entered.
When calculating a melting solid or solidifying liquid it is adequate, Concerning pressure and temperature, to put in either the value given for *t* and p = -1000 or the value given for *p* and t = -1000 and the value for x (x = 10 or x = 11). If *p* and *t* and *x* are entered as given values, the program tests whether *p* and *t* fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}

Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}
Pressure range from p_t to p_{max}

Only the solidification curve (x = 11) is calculated for pressures p for which applies $p_{max\_Ice} < p \leq p_{max}$.

3. Sublimation Region $(100 \leq x \leq 101)$:

The phase fraction x equates to the vapor fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
If the state point to be calculated is located in the melting region, a value between 100 and 101 must be entered for x.
If sublimating solid is to be calculated, the value 100 has to be entered for *x*. In case of desublimating steam x = 101 has to be entered.
When calculating a sublimating solid or desublimating steam it is adequate, Concerning pressure and temperature, to put in either the value given for *t* and p = -1000 or the value given for *p* and t = -1000 and the value for x (x = 100 or x = 101). If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **RHOPTXCO2 = - 1000**, **RHO = -1000** or **rho_ptx_CO2 = - 1000** for input values:

Single phase region:

Solid, liquid, overheated steam (x = –1):
- at $p < p_{min}$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at $t < t_{min}$ or $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Two phase regions:

Wet steam region ($0 \leq x \leq 1$):
- at $p = –1000$ and $t < t_t$ or $t > t_c$
- at $t = –1000$ and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \leq x \leq 11$):
- at $p = –1000$ and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at $t = –1000$ and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \leq x \leq 101$):
- at $p = –1000$ and $t < t_{min}$ or $t > t_t$
- at $t = –1000$ and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ or $t > t_t$

References: [2]

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

Function Name: **s_ptx_CO2**

Subprogram with value of the function: for call from Fortran — **REAL*8 FUNCTION SPTXCO2(P,T,X)**
REAL*8 P,T,X

Subprogram with parameter: for call from the DLL — **INTEGER*4 FUNCTION C_SPTXCO2(S,P,T,X)**
REAL*8 S,P,T,X

Input Values

P – Pressure *p* in bar

T - Temperature *t* in °C

X - *x* in kg / kg (Phase fraction, see the following explanations)

Result

SPTXCO2, **S** or **s_ptx_CO2** - Specific entropy *s* in kJ/kg K

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Temperature range: from t_{min} to t_{max}
Pressure range: from p_{min} to p_{max} for liquid and steam
from p_{min} to p_{max_Ice} for solid

Details on the phase fraction *x*

If the state point to be calculated is located in the single phase region (solid, liquid or superheated steam), x= -1 must be entered as a pro-forma value.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg *p*,*h*-diagram chapter 3.1):

1. Wet Vapor Region $(0 \le x \le 1)$:

The phase fraction *x* equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
If the state point to be calculated is located in the wet steam region, a value between 0 and 1 must be entered for x.
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for *x*. In case of dry saturated steam (dew curve) x = 1 has to be entered.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 0 or 1 for *x* must be entered when calculating boiling liquid or dry saturated steam. If *p* and *t* and *x* are entered, the program will consider *p* and *t* to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \le x \le 11)$:

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
If the state point to be calculated is located in the melting region, a value between 10 and 11 must be entered for x.
If melting solid (melting curve) is to be calculated, the value 10 has to be entered for *x*. In case of solidifying liquid (solidification curve) x = 11 has to be entered.
When calculating a melting solid or solidifying liquid it is adequate, concerning pressure and temperature, to put in either the value given for *t* and p = -1000 or the value given for *p* and t = -1000 and the value for x (x = 10 or x = 11). If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}

Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}
Pressure range from p_t to p_{max}

Only the solidification curve (x = 11) is calculated for pressures p for which applies $p_{max\_Ice} < p \leq p_{max}$.

3. Sublimation Region $(100 \leq x \leq 101)$:

The phase fraction x equates to the vapor fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
If the state point to be calculated is located in the melting region, a value between 100 and 101 must be entered for x.
If sublimating solid is to be calculated, the value 100 has to be entered for *x*. In case of desublimating steam x = 101 has to be entered.
When calculating a sublimating solid or desublimating steam it is adequate, Concerning pressure and temperature, to put in either the value given for *t* and p = -1000 or the value given for *p* and t = -1000 and the value for x (x = 100 or x = 101). If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **SPTXCO2 = –1000**, **S = –1000** or **s_ptx_CO2 = –1000** for input values:
Single phase region:
Solid, liquid, overheated steam (x = –1):
- at $p < p_{min}$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at $t < t_{min}$ or $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Two phase regions:
Wet steam region ($0 \leq x \leq 1$):
- at p = –1000 and $t < t_t$ or $t > t_c$
- at t = –1000 and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \leq x \leq 11$):
- at p = –1000 and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at t = –1000 and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \leq x \leq 101$):
- at p = –1000 and $t < t_{min}$ or $t > t_t$
- at t = –1000 and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ or $t > t_t$

References: [2]

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

Function Name: **t_ph_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION TPHCO2(P,H)
REAL*8 P,H

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_TPHCO2(T,P,H)
REAL*8 T,P,H

Input Values

P – Pressure p in bar

H - Specific enthalpy h in kJ/kg

Result

TPHCO2, **T** or **t_ph_CO2** – Temperature t in °C

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Pressure range: from p_{min} to p_{max} for liquid and steam
from p_{min} to p_{max_Ice} for solid

Enthalpy range: from $h < h_{max} = h(p_{min}, t_{max})$ to $h > h_{min} = h(p_{min}, t_{min})$

Temperature range: from t_{min} to t_{max}(resulting from internal calculation of the subprograms)

Details on calculating the two phase regions

Using the given values for p and h, the program determines whether the point of state to be calculated is located in the single phase region (solid, liquid or steam) or in the two phase region. The two phase regions wet steam region, melting region and sublimation region (cp. lg *p,h*-diagram chapter 3.1) are calculated automatically by the subprograms. After that, the calculation is realized for the certain region.

Results for wrong input values

Result **T_PH_CO2**, **T = –1000** or **t_ph_CO2 = –1000** for input values:

Single phase region:

Solid, liquid, overheated steam:
- at $p < p_{min}$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at calculation result $t < t_{min}$ or $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Two phase regions:

Wet steam region:
- at $p < p_t$ or $p > p_c$
- at calculation result $t < t_t$ or $t > t_c$

Melting region:
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at calculation result $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Sublimation region:
- at $p < p_{min}$ or $p > p_t$
- at calculation result $t < t_{min}$ or $t > t_t$

References: [2]

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

Function Name: **t_ps_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION TPSCO2(P,S)
REAL*8 P,S

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_TPSCO2(T,P,S)
REAL*8 T,P,S

Input Values

P – Pressure *p* in bar

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

Result

TPSCO2, **T** or **t_ps_CO2** – Temperature *t* in °C

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Pressure range: from p_{min} to p_{max} for liquid and steam
from p_{min} to p_{max_Ice} for solid

Entropy range: from $s < s_{max} = s(p_{min}, t_{max})$ to $s > s_{min} = s(p_{min}, t_{min})$

Temperature range: from t_{min} to t_{max}(resulting from internal calculation of the subprograms)

Details on calculating the two phase regions

Using the given values for *p* and *h*, the program determines whether the point of state to be calculated is located in the single phase region (solid, liquid or steam) or in the two phase region. The two phase regions wet steam region, melting region and sublimation region (cp. lg *p,h*-diagram chapter 3.1) are calculated automatically by the subprograms. After that, the calculation is realized for the certain region.

Results for wrong input values

Result **T_PS_CO2**, **T = –1000** or **t_ps_CO2 = –1000** for input values:

Single phase region:

Solid, liquid, overheated steam:
- at $p < p_{min}$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at calculation result $t < t_{min}$ or $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Two phase regions:

Wet steam region:
- at $p < p_t$ or $p > p_c$
- at calculation result $t < t_t$ or $t > t_c$

Melting region:
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at calculation result $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Sublimation region:
- at $p < p_{min}$ or $p > p_t$
- at calculation result $t < t_{min}$ or $t > t_t$

References: [2]

Saturation Temperature $t_s = f(p)$

Function Name:	**ts_p_CO2**
Subprogram with value of the function: for call from Fortran	**REAL*8 FUNCTION TSPCO2(P)** REAL*8 P
Subprogram with parameter: for call from the DLL	**INTEGER*4 FUNCTION C_TSPCO2(TS,P)** REAL*8 TS,P

Input Values

P – Pressure p in bar

Result

TSPCO2, **TS** or **ts_p_CO2** – Saturation temperature t_s in °C

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Pressure range: from p_t to p_c

Results for wrong input values

Result **TSPCO2= –1000**, **TS = –1000** or **ts_p_CO2 = –1000** for input values:

- at $p < p_t$ or $p > p_c$

References: [2]

Melting Temperature t_{mel} = f(p)

Function Name: **tmel_p_CO2**

Subprogram with value of the function: for call from Fortran

REAL*8 FUNCTION TMELCO2(P)
REAL*8 P

Subprogram with parameter: for call from the DLL

INTEGER*4 FUNCTION C_TMELCO2(TMEL,P)
REAL*8 TMEL,P

Input Values

P – Pressure p in bar

Result

TMELCO2, **TMEL** or **tmel_p_CO2** – Melting temperature t_{mel} in °C

Range of Validity (cp. p,t-diagram in chapter 3.1)

Pressure range: from p_t to p_{max}

Results for wrong input values

Result **TMELCO2 = –1000**, **TMEL = –1000** or **tmel_p_CO2 = –1000** for input values:

- at $p < p_t$ or $p > p_{max}$

References: [2]

Sublimation Temperature t_{sub} = f(p)

Function Name: **tsub_p_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION TSUBCO2(P)
REAL*8 P

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_TSUBCO2(TSUB,P)
REAL*8 TSUB,P

Input Values

P – Pressure p in bar

Result

TSUBCO2, **TSUB** or **tsub_p_CO2** – Sublimation temperature t_{sub} in °C

Range of Validity (cp. p,t-diagram in chapter 3.1)

Pressure range: from p_{min} to p_t

Results for wrong input values

Result **TSUBCO2, = –1000**, **TSUB = –1000** or **tsub_p_CO2 = –1000** for input values:

- at $p < p_{min}$ or $p > p_t$

References: [2]

Specific Volume *v* = f(*p*,*t*,*x*)

Function Name: **v_ptx_CO2**

Subprogram with value of the function: for call from Fortran — **REAL*8 FUNCTION VPTXCO2(P,T,X)**
REAL*8 P,T,X

Subprogram with parameter: for call from the DLL — **INTEGER*4 FUNCTION C_VPTXCO2(V,P,T,X)**
REAL*8 V,P,T,X

Input Values

P – Pressure *p* in bar

T - Temperature *t* in °C

X - *x* in kg / kg (Phase fraction, see the following explanations)

Result

VPTXCO2, **V** or **v_ptx_CO2** – Specific volume *v* in m^3/kg

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Temperature range: from t_{min} to t_{max}
Pressure range: from p_{min} to p_{max} for liquid and steam
from p_{min} to p_{max_Ice} for solid

Details on the phase fraction *x*

If the state point to be calculated is located in the single phase region (solid, liquid or superheated steam), x= -1 must be entered as a pro-forma value.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg *p*,*h*-diagram chapter 3.1):

1. Wet Vapor Region $(0 \le x \le 1)$:

The phase fraction *x* equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for *x*. In case of dry saturated steam (dew curve) x = 1 has to be entered.
If the state point to be calculated is located in the wet steam region, a value between 0 and 1 must be entered for x.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 0 or 1 for *x* must be entered when calculating boiling liquid or dry saturated steam. If *p* and *t* and *x* are entered, the program will consider *p* and *t* to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \le x \le 11)$:

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
If melting solid (melting curve) is to be calculated, the value 10 has to be entered for *x*. In case of solidifying liquid (solidification curve) x = 11 has to be entered.
If the state point to be calculated is located in the melting region, a value between 10 and 11 must be entered for x.
When calculating a melting solid or solidifying liquid it is adequate, concerning pressure and temperature, to put in either the value given for *t* and p = -1000 or the value given for *p* and t = -1000 and the value for x (x = 10 or x = 11). If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}

Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}
Pressure range from p_t to p_{max}

Only the solidification curve (x = 11) is calculated for pressures p for which applies $p_{max\_Ice} < p \leq p_{max}$.

3. Sublimation Region $(100 \leq x \leq 101)$:

The phase fraction x equates to the vapor fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
If sublimating solid is to be calculated, the value 100 has to be entered for *x*. In case of desublimating steam x = 101 has to be entered.
If the state point to be calculated is located in the melting region, a value between 100 and 101 must be entered for x.
When calculating a sublimating solid or desublimating steam it is adequate, Concerning pressure and temperature, to put in either the value given for *t* and p = -1000 or the value given for *p* and t = -1000 and the value for x (x = 100 or x = 101). If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **VPTXCO2 = –1000**, **H = –1000** or **v_ptx_CO2 = –1000** for input values:

Single phase region:

Solid, liquid, overheated steam (x = –1):
- at $p < p_{min}$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at $t < t_{min}$ or $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Two phase regions:

Wet steam region ($0 \leq x \leq 1$):
- at $p = –1000$ and $t < t_t$ or $t > t_c$
- at $t = –1000$ and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \leq x \leq 11$):
- at $p = –1000$ and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at $t = –1000$ and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \leq x \leq 101$):
- at $p = –1000$ and $t < t_{min}$ or $t > t_t$
- at $t = –1000$ and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ or $t > t_t$

References: [2]

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

Function Name:	**w_ptx_CO2**
Subprogram with value of the function: for call from Fortran	**REAL*8 FUNCTION WPTXCO2(P,T,X)** REAL*8 P,T,X
Subprogram with parameter: for call from the DLL	**INTEGER*4 FUNCTION C_WPTXCO2(W,P,T,X)** REAL*8 W,P,T,X

Input Values

P – Pressure *p* in bar

T - Temperature *t* in °C

X - *x* in kg / kg (Phase fraction, see the following explanations)

Result

WPTXCO2, **W** or **w_ptx_CO2** – Speed of sound *w* in m/s

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Temperature range: from t_t to t_{max}
Pressure range: from p_{min} to p_{max} for liquid and steam

Details on the phase fraction *x*

If the state point to be calculated is located in the single phase region (liquid or superheated steam), x= -1 must be entered as a pro-forma value. Calculating solid is not possible.

The two phase regions wet steam region, melting region and sublimation region are calculated automatically by the subprograms. For this purpose the following facts have to be considered (cp. lg *p*,*h*-diagram chapter 3.1):

1. Wet Vapor Region $(0 \le x \le 1)$:

The phase fraction *x* equates to the vapor fraction x in the wet steam region in (kg dry saturated steam)/(kg wet steam).
The calculation for *x* values between 0 and 1 is not possible.
If boiling liquid (boiling curve) is to be calculated, the value 0 has to be entered for *x*. In case of dry saturated steam (dew curve) x = 1 has to be entered.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 0 or 1 for *x* must be entered when calculating boiling liquid or dry saturated steam. If *p* and *t* and *x* are entered, the program will consider *p* and *t* to represent the vapor pressure curve.

Boiling and dew curve: Temperature range from t_t to t_c
Pressure range from p_t to p_c

2. Melting Region $(10 \le x \le 11)$:

The phase fraction x equates to the liquid fraction x in the melting region in (kg solidifying liquid)/(kg melt), whereas melt is a mixture of melting solid and solidifying liquid.
The calculation for *x* values of x = 10 (melting curve) and between 10 and 11 is not possible.
If solidifying liquid (solidification curve) is to be calculated, the value 11 has to be entered for *x*.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 11 for *x* must be entered when calculating solidifying liquid**.**
If *p* and *t* and *x* are entered as given values, the program tests whether *p* and *t* fulfil the melting pressure curve.

Melting curve: Temperature range from t_t to t_{max_Ice}
Pressure range from p_t to p_{max_Ice}

Solidification curve: Temperature range from t_t to t_{max}

Pressure range from p_t to p_{max}

3. Sublimation Region $(100 \leq x \leq 101)$:

The phase fraction x equates to the vapor fraction x in the sublimation region in (kg desublimating steam)/(kg sublimation powder), whereas sublimation powder is a mixture of sublimating solid and desublimating steam.
The calculation for *x* values of x = 100 (sublimation curve) and between 100 and 101 is not possible.
If desublimating steam (desublimation curve) is to be calculated, the value 101 has to be entered for *x*.
Concerning pressure and temperature either the given value for *t* and *p* = -1000 or the given value for *p* and *t* = -1000 and in both cases the value 101 for *x* must be entered when calculating desublimating steam. If *p* and *t* and *x* are entered as given values the program tests whether *p* and *t* fulfil the sublimation-pressure-curve.

Sublimation and
desublimation curve Temperatur range from t_{min} to t_t
Pressure range from p_{min} to p_t

Results for wrong input values

Result **WPTXCO2 = –1000** or **w_ptx_CO2 = –1000** for input values:

Single phase region:

Liquid and overheated steam (x = –1):
- at $p < p_{min}$ or $p > p_{max}$
- at $t < t_t$ or $t > t_{max}$

Two phase regions:

Wet steam region ($0 \leq x \leq 1$):
- at $0 < x < 1$, i.e. calculation in the wet steam region not possible!
- at $p = –1000$ and $t < t_t$ or $t > t_c$
- at $t = –1000$ and $p < p_t$ or $p > p_c$
- at $p < p_t$ or $p > p_c$
- at $t < t_t$ or $t > t_c$

Melting region ($10 \leq x \leq 11$):
- at $10 < x < 11$, i.e. calculation on the melting curve and in the melting region not possible!
- at $p = –1000$ and $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$
- at $t = –1000$ and $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ at $x = 10$
- at $t < t_t$ or $t > t_{max}$ or $t > t_{max\_Ice}$ at $x = 10$

Sublimation region ($100 \leq x \leq 101$):
- at $100 \leq x < 101$, i.e. calculation on the sublimation curve and in the sublimation region not possible!
- at $p = –1000$ and $t < t_{min}$ or $t > t_t$
- at $t = –1000$ and $p < p_{min}$ or $p > p_t$
- at $p < p_{min}$ or $p > p_t$
- at $t < t_{min}$ or $t > t_t$

References: [2]

Backward Function: Vapor Fraction *x* = f(*p*,*h*)

Function Name: **x_ph_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION XPHCO2(P,H)
REAL*8 P,H

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_XPHCO2(T,P,H)
REAL*8 X,P,H

Input Values

P – Pressure *p* in bar

H - Specific enthalpy *h* in kJ/kg

Result

XPHCO2, **X** or **x_ph_CO2** – Vapor fraction *x* in (kg saturated steam/kg wet steam)

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Pressure range: from p_{min} to p_{max} for liquid and steam
from p_{min} to p_{max_Ice} for solid

Enthalpy range: from $h < h_{max} = h(p_{min}, t_{max})$ to $h > h_{min} = h(p_{min}, t_{min})$

Temperature range: from t_{min} to t_{max}(resulting from internal calculation of the subprograms)

Details on calculating the two phase regions

The two phase regions are calculated automatically by the subprograms. Using the given values for *p* and *h*, the program determines whether the point of state to be calculated is located in the single phase region (solid, liquid or steam) or in one of the two phase regions wet steam region, melting region and sublimation region (cp. lg *p,h*-diagram in chapter 3.1). When calculating a two phase mixture, *x* will be calculated. If the state point to be calculated is located in the single-phase region the result is set to x = -1.

Results for wrong input values

Result **X_PH_CO2**, **X = –1** or **x_ph_CO2 = –1** for input values:

If the state point to be calculated is located in the single phase region (cp. lg *p,h*-diagram in chapter 3.1).

Two phase regions:

Wet steam region ($0 \leq x \leq 1$):
- at $p < p_t$ or $p > p_c$
- at calculation result $t < t_t$ or $t > t_c$

Melting region ($10 \leq x \leq 11$):
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at calculation result $t < t_t$ *or* $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Sublimation region ($100 \leq x \leq 101$):
- at $p < p_{min}$ or $p > p_t$
- at calculation result $t < t_{min}$ or $t > t_t$

References: [2]

Backward Function: Vapor Fraction *x* = f(*p*,*s*)

Function Name: **x_ps_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION XPSCO2(P,S)
REAL*8 P,S

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_XPSCO2(X,P,S)
REAL*8 X,P,S

Input Values

P – Pressure p in bar

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

Result

XPSCO2, **X** or **x_ps_CO2** – Vapor fraction x in (kg saturated steam/kg wet steam)

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Pressure range: from p_{min} to p_{max} for liquid and steam
from p_{min} to p_{max_Ice} for solid

Entropy range: from $s < s_{max} = s(p_{min}, t_{max})$ to $s > s_{min} = s(p_{min}, t_{min})$

Temperature range: from t_{min} to t_{max}(resulting from internal calculation of the subprograms)

Details on calculating the two phase regions

The two phase regions are calculated automatically by the subprograms. Using the given values for p and s, the program determines whether the point of state to be calculated is located in the single phase region (solid, liquid or steam) or in one of the two phase regions wet steam region, melting region and sublimation region (cp. lg *p,h*-diagram in chapter 3.1). When calculating a two phase mixture, x will be calculated. If the state point to be calculated is located in the single-phase region the result is set to x = -1.

Results for wrong input values

Result **X_PS_CO2**, **X = -1** or **x_ps_CO2 = -1** for input values:

If the state point to be calculated is located in the single phase region (cp. lg *p,h*-diagram in chapter 3.1).

Two phase regions:

Wet steam region ($0 \le x \le 1$):
- at $p < p_t$ or $p > p_c$
- at calculation result $t < t_t$ or $t > t_c$

Melting region ($10 \le x \le 11$):
- at $p < p_t$ or $p > p_{max}$ or $p > p_{max\_Ice}$ when calculating solid
- at calculation result $t < t_t$ *or* $t > t_{max}$ or $t > t_{max\_Ice}$ when calculating solid

Sublimation region ($100 \le x \le 101$):
- at $p < p_{min}$ or $p > p_t$
- at calculation result $t < t_{min}$ or $t > t_t$

References: [2]

3.3 Property Functions for Solid Carbon Dioxide (Dry Ice)

#$K+Thermal Diffusivity *a* = f(*p*,*t*)

Function Name:	**aICE_pt_CO2**
Subprogram with value of the function: for call from Fortran	**REAL*8 FUNCTION APICEPTCO2(P,T)** REAL*8 P,T
Subprogram with parameter: for call from the DLL	**INTEGER*4 FUNCTION C_AICEPTCO2(A,P,T)** REAL*8 A,P,T

Input Values

P - Pressure p in bar

T - Temperature t in °C

Result

AICEPTCO2, **a** or **aICE_pt_CO2** – Thermal diffusivity $a = \frac{\lambda * v}{c_p}$ in m²/s

Range of Validity (cp. p,t-diagram in chapter 3.1)

Pressure range: from p_{min} to p_{max_Ice}

Temperature range: from t_{min} to t_{max_Ice}

Results for wrong input values

Result **AICEPTCO2 = –1000**, **A = –1000** or **aICE_pt_CO2 = –1000** for input values:

- at $t < t_{min}$ or $t > t_{max\_Ice}$
- at $p < p_{min}$ or $p > p_{max\_Ice}$

References: [4], [5]

# FUNC_300
$ a = f(t)
K a = f(t)
+ SUCH:300

#$K+ Specific Isobaric Heat Capacity c_p = f(p,t)

Function Name:	**cpICE_pt_CO2**
Subprogram with value of the function: for call from Fortran	**REAL*8 FUNCTION CPICEPTCO2(P,T)** REAL*8 P,T
Subprogram with parameter: for call from the DLL	**INTEGER*4 FUNCTION C_CPICEPTCO2(CP,P,T)** REAL*8 CP,P,T

Input Values

P - Pressure p in bar

T - Temperature t in °C

Result

CPICETCO2, **CP** or **cpICE_t_CO2** – specific isobaric heat capacity c_p in kJ/(kg K)

Range of Validity (cp. p,t-diagram in chapter 3.1)

Pressure range: from p_{min} to p_{max_Ice}

Temperature range: from t_{min} to t_{max_Ice}

Results for wrong input values

Result **CPICEPTCO2 = –1000**, **CP = –1000** or **cpICE_pt_CO2 = –1000** for input values:

- at $t < t_{min}$ or $t > t_{max\_Ice}$
- at $p < p_{min}$ or $p > p_{max\_Ice}$

References: [4]

# FUNC_310
$ cpICE = f(t)
K cpICE = f(t)
+ SUCH:310

#$K+ Specific Enthalpy *h* = f(*p*,*t*)

Function Name:	**hICE_pt_CO2**
Subprogram with value of the function: for call from Fortran	**REAL*8 FUNCTION HICEPTCO2(P,T)** REAL*8 P,T
Subprogram with parameter: for call from the DLL	**INTEGER*4 FUNCTION C_HICEPTCO2(H,P,T)** REAL*8 H,P,T

Input Values

P - Pressure *p* in bar

T - Temperature *t* in °C

Result

HICETCO2, **H** or **hICE_t_CO2** – specific enthalpy h in kJ / kg

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Pressure range: from p_{min} to p_{max_Ice}

Temperature range: from t_{min} to t_{max_Ice}

Results for wrong input values

Result **HICEPTCO2 = –1000**, **H = –1000** or **hICE_pt_CO2 = –1000** for input values:

- at $t < t_{min}$ or $t > t_{max\_Ice}$
- at $p < p_{min}$ or $p > p_{max\_Ice}$

References: [4]

[#] FUNC_320
[$] hICE = f(t)
[K] hICE = f(t)
[+] SUCH:320

#$K+ Thermal Conductivity $\lambda = \mathrm{f}(t)$

Function Name:	**lambdaICE_t_CO2**
Subprogram with value of the function: for call from Fortran	**REAL*8 FUNCTION LAMICETCO2(T)** REAL*8 T
Subprogram with parameter: for call from the DLL	**INTEGER*4 FUNCTION C_LAMICETCO2(LAM,T)** REAL*8 LAM,T

Input Values

T - Temperature t in °C

Result

LAMICETCO2, **LAM** or **lambdaICE_t_CO2** – Thermal conducivity λ in W/m K

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Temperature range: from t_{min} to t_{max_Ice}

Results for wrong input values

Result **LAMICETCO2 = –1000**, **LAM = –1000** or **lambdaICE_t_CO2 = –1000** for input values:

- at $t < t_{min}$ or $t > t_{max\_Ice}$

References: [5]

[#] FUNC_330
[$] lambdaICE = f(t)
[K] lambdaICE = f(t)
[+] SUCH:330

#$K+ Density ρ = f(p,t)

Function Name: **rhoICE_pt_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION RHOICEPTCO2(P,T)
REAL*8 P,T

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_RHOICEPTCO2(RHO,P,T)
REAL*8 RHO,P,T

Input Values

P - Pressure p in bar

T - Temperature t in °C

Result

RHOICEPTCO2, **RHO** or **rhoICE_pt_CO2** - Density ρ in kg/m³

Range of Validity (cp. p,t-diagram in chapter 3.1)

Pressure range: from p_{min} to p_{max_Ice}

Temperature range: from t_{min} to t_{max_Ice}

Results for wrong input values

Result **RHOICEPTCO2 = –1000**, **RHO = –1000** or **rhoICE_pt_CO2 = –1000** for input values:

- at $t < t_{min}$ or $t > t_{max\_Ice}$
- at $p < p_{min}$ or $p > p_{max\_Ice}$

References: [4]

\# FUNC_340
$ rhoICE = f(t)
K rhoICE = f(t)
\+ SUCH:340

#$K+Specific Entropy *s* = f(*p*,*t*)

Function Name: **sICE_pt_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION SICEPTCO2(P,T)
REAL*8 P,T

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_SICEPTCO2(S,P,T)
REAL*8 S,P,T

Input Values

P - Pressure *p* in bar

T - Temperature *t* in °C

Result

SICEPTCO2, **S** or **sICE_pt_CO2** – Specific Entropy s in kJ/(kg K)

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Pressure range: from p_{min} to p_{max_Ice}

Temperature range: from t_{min} to t_{max_Ice}

Results for wrong input values

Result **SICEPTCO2 = –1000**, **S = –1000** or **sICE_pt_CO2 = –1000** for input values:

- at $t < t_{min}$ or $t > t_{max\_Ice}$
- at $p < p_{min}$ or $p > p_{max\_Ice}$

References: [4]

# FUNC_350
$ sICE = f(t)
K sICE = f(t)
+ SUCH:350

#$K+ Specific Volume *v* = f(*p*,*t*)

Function Name: **vICE_pt_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION VICEPTCO2(P,T)
REAL*8 P,T

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_VICETCO2(V,P,T)
REAL*8 V,P,T

Input Values

P - Pressure p in bar

T - Temperature t in °C

Result

VICEPTCO2, **V** or **vICE_pt_CO2** – specific volume v in m^3/kg

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Pressure range: from p_{min} to p_{max_Ice}

Temperature range: from t_{min} to t_{max_Ice}

Results for wrong input values

Result **VICEPTCO2 = –1000**, **V = –1000** or **vICE_pt_CO2 = –1000** for input values:

- at $t < t_{min}$ or $t > t_{max\_Ice}$
- at $p < p_{min}$ or $p > p_{max\_Ice}$

References: [4]

\# FUNC_360
$ vICE = f(t)
K vICE = f(t)
\+ SUCH:360

#$K+ Backward function: Temperature t = f(p,h)

Function Name: **tICE_ph_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION TICEPHCO2(P,H)
REAL*8 P,H

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_TICEPHCO2(T,P,H)
REAL*8 T,P,H

Input Values

P - Pressure p in bar

H – Specific enthalpy h in kJ/kg

Result

TICEPHCO2, **T** or **tICE_ph_CO2** - Temperature in °C

Range of Validity (cp. p,t-diagram in chapter 3.1)

Enthalpy range: from $h > h_{min} = h(t_{min})$ to $h < h_{max\_Ice} = h(t_{max\_Ice})$

Pressure range: from p_{min} to p_{max_Ice}

Temperature range: for results from t_{min} to t_{max_Ice}

Results for wrong input values

Result **TICEPHCO2 = –1000**, **T = –1000** or **tICE_ph_CO2 = –1000** for input values:

- at $h < h_{min} = h(t_{min})$ or $h > h_{max\_Ice} = h(t_{max\_Ice})$ or
- at $p < p_{min}$ or $p > p_{max\_Ice}$

References: [4]

[#] FUNC_370
[$] tICE = f(h)
[K] tICE = f(h)
[+] SUCH:370

#$K+ Backward function: Temperature *t* = f(*p*,*s*)

Function Name: **tICE_ps_CO2**

Subprogram with value of the function: for call from Fortran
REAL*8 FUNCTION TICEPSCO2(P,S)
REAL*8 P,S

Subprogram with parameter: for call from the DLL
INTEGER*4 FUNCTION C_TICEPSCO2(T,P,S)
REAL*8 T,P,S

Input Value

P - Pressure p in bar

S – Specific Entropy in kJ/(kg K)

Result

TICEPSCO2, **T** or **tICE_ps_CO2** - Temperature in °C

Range of Validity (cp. *p*,*t*-diagram in chapter 3.1)

Entropy range: from $s > s_{min} = s(t_{min})$ to $s < s_{max\_Ice} = s(t_{max\_Ice})$
Pressure range: from p_{min} to p_{max_Ice}
Temperature range: for results from t_{min} to t_{max_Ice}

Results for wrong input values

Result **TICEPSCO2 = –1000**, **T = –1000** or **tICE_ps_CO2 = –1000** for input values:

- at $s < s_{min} = s(t_{min})$ or $s > s_{max\_Ice} = s(t_{max\_Ice})$
- at $p < p_{min}$ or $p > p_{max\_Ice}$

References: [4]

[#] FUNC_380
[$] tICE = f(s)
[K] tICE = f(s)
[+] SUCH:380

KCE-ThermoFluidProperties
www.thermofluidprop.com

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)

Library LibSBTL_IF97
Library LibSBTL_95

Extremely fast property calculations according to the IAPWS Guideline 2015 Spline-based Table Look-up Method (SBTL) applied to the Industrial Formulation IAPWS-IF97 and to the Scientific Formulation IAPWS-95 for Computational Fluid Dynamics and simulating non-stationary processes

Humid Combustion Gas Mixtures

Library LibHuGas

Model: Ideal mixture of the real fluids:

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

and of the ideal gases:

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

Consideration of:

- Dissociation from VDI 4670
- Poynting effect

Humid Air

Library LibHuAir

Model: Ideal mixture of the real fluids:

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

Consideration of:

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

Carbon Dioxide Including Dry Ice

Library LibCO2

Formulation of Span and Wagner (1996)

Seawater

Library LibSeaWa

IAPWS Industrial Formulation 2013

Ice

Library LibICE

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

Ideal Gas Mixtures

Library LibIdGasMix

Model: Ideal mixture of the ideal gases:

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

Consideration of:

- Dissociation from the VDI Guideline 4670

Library LibIDGAS

Model: Ideal gas mixture from VDI Guideline 4670

Consideration of:

- Dissociation from the VDI Guideline 4670

Humid Air

Library ASHRAE LibHuAirProp

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

Consideration of:

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

www.ashrae.org/bookstore

Dry Air Including Liquid Air

Library LibRealAir

Formulation of Lemmon et al. (2000)

Refrigerants

Ammonia

Library LibNH3

Formulation of Tillner-Roth et al. (1993)

R134a

Library LibR134a

Formulation of Tillner-Roth and Baehr (1994)

Iso-Butane

Library LibButane_Iso

Formulation of Bücker and Wagner (2006)

n-Butane

Library LibButane_n

Formulation of Bücker and Wagner (2006)

Mixtures for Absorption Processes

Ammonia/Water Mixtures

Library LibAmWa

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

Helmholtz energy equation for the mixing term

(also useable for calculating the Kalina Cycle)

Water/Lithium Bromide Mixtures

Library LibWaLi

Formulation of Kim and Infante Ferreira (2004)

Gibbs energy equation for the mixing term

Liquid Coolants

Liquid Secondary Refrigerants

Library LibSecRef

Liquid solutions of water with

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

Formulation of the International Institute of Refrigeration (IIR 2010)

Ethanol
Library LibC2H5OH

Formulation of Schroeder (2012)

Methanol
Library LibCH3OH

Formulation of de Reuck and Craven (1993)

Propane
Library LibPropane

Formulation of Lemmon et al. (2009)

Siloxanes as ORC Working Fluids

Octamethylcyclotetrasiloxane $C_8H_{24}O_4Si_4$ **Library LibD4**

Decamethylcyclopentasiloxane $C_{10}H_{30}O_5Si_5$ **Library LibD5**

Tetradecamethylhexasiloxane $C_{14}H_{42}O_5Si_6$ **Library LibMD4M**

Hexamethyldisiloxane $C_6H_{18}OSi_2$ **Library LibMM**

Formulation of Colonna et al. (2006)

Dodecamethylcyclohexasiloxane $C_{12}H_{36}O_6Si_6$ **Library LibD6**

Decamethyltetrasiloxane $C_{10}H_{30}O_3Si_4$ **Library LibMD2M**

Dodecamethylpentasiloxane $C_{12}H_{36}O_4Si_5$ **Library LibMD3M**

Octamethyltrisiloxane $C_8H_{24}O_2Si_3$ **Library LibMDM**

Formulation of Colonna et al. (2008)

Nitrogen and Oxygen
Libraries LibN2 and LibO2

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

Hydrogen
Library LibH2

Formulation of Leachman et al. (2009)

Helium
Library LibHe

Formulation of Arp et al. (1998)

Hydrocarbons

Decane $C_{10}H_{22}$ **Library LibC10H22**

Isopentane C_5H_{12} **Library LibC5H12_ISO**

Neopentane C_5H_{12} **Library LibC5H12_NEO**

Isohexane C_6H_{14} **Library LibC6H14**

Toluene C_7H_8 **Library LibC7H8**

Formulation of Lemmon and Span (2006)

Further Fluids

Carbon monoxide CO **Library LibCO**

Carbonyl sulfide COS **Library LibCOS**

Hydrogen sulfide H_2S **Library LibH2S**

Nitrous oxide N_2O **Library LibN2O**

Sulfur dioxide SO_2 **Library LibSO2**

Acetone C_3H_6O **Library LibC3H6O**

Formulation of Lemmon and Span (2006)

For more information please contact:

KCE-ThermoFluidProperties UG (limited liability) & Co. KG
Professor Hans-Joachim Kretzschmar

Wallotstr. 3
01307 Dresden, Germany

Internet: www.thermofluidprop.com
E-mail: info@thermofluidprop.com
Phone: +49-351-27597860
Mobile: +49-172-7914607
Fax: +49-3222-4262250

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

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

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

Backward Functions
- T, v, s (p,h)
- T, v, h (p,s)
- p, T, v (h,s)
- p, T (v,h)
- p, T (v,u)

Thermodynamic Derivatives
- Partial derivatives can be calculated.

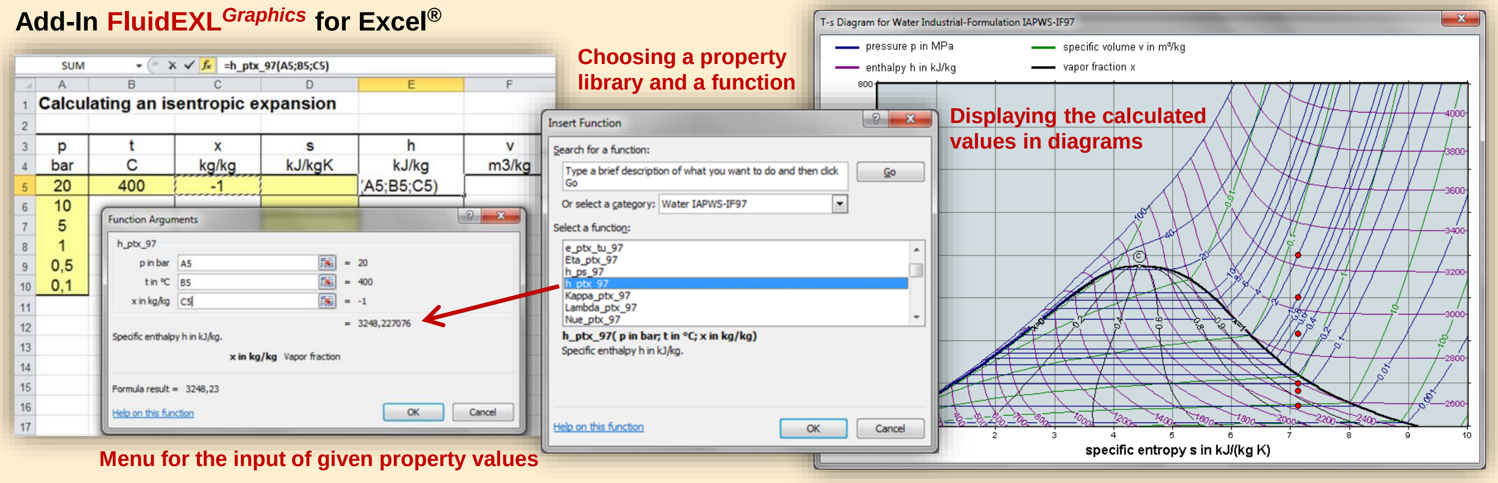
[a] Not all of these property functions are available in all property libraries.

www.thermofluidprop.com

KCE-ThermoFluidProperties
www.thermofluidprop.com

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

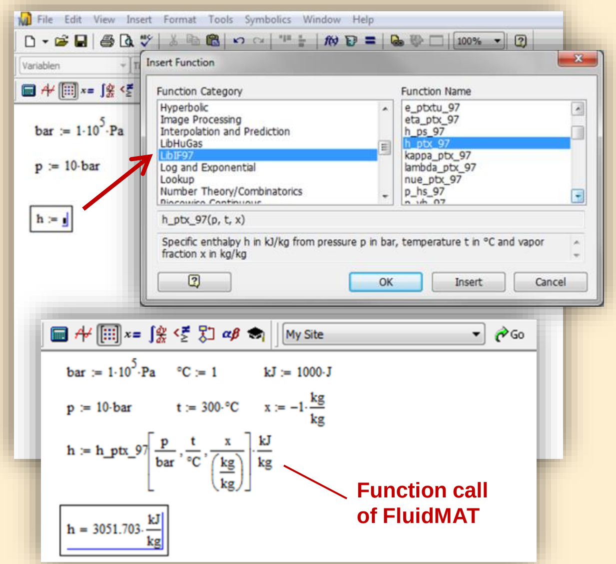
Add-In FluidEXL*Graphics* for Excel®



Menu for the input of given property values

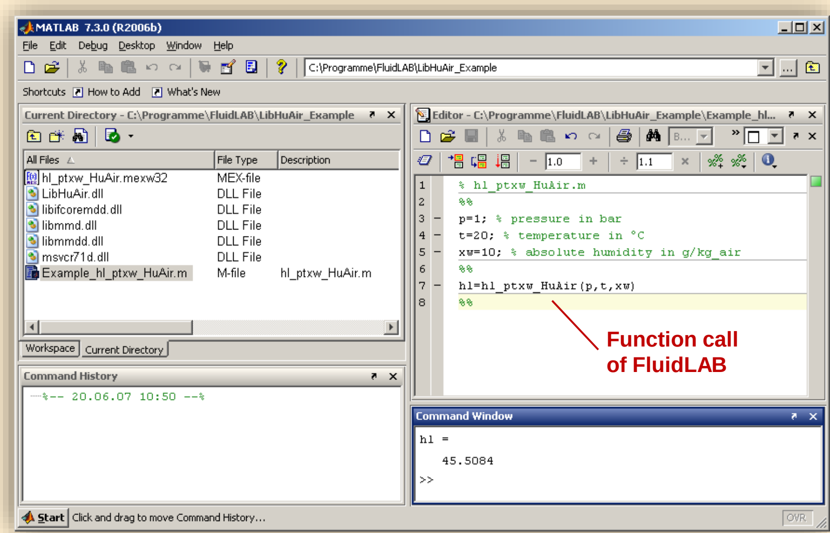
Add-In FluidMAT for Mathcad®

The property libraries can be used in Mathcad®.



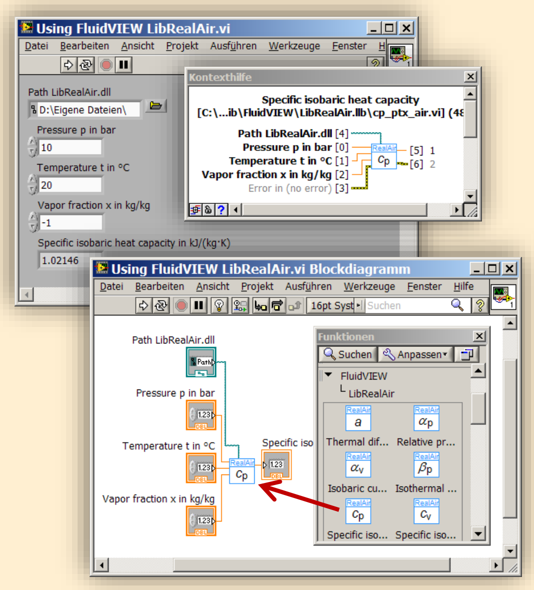
Add-In FluidLAB for MATLAB®

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



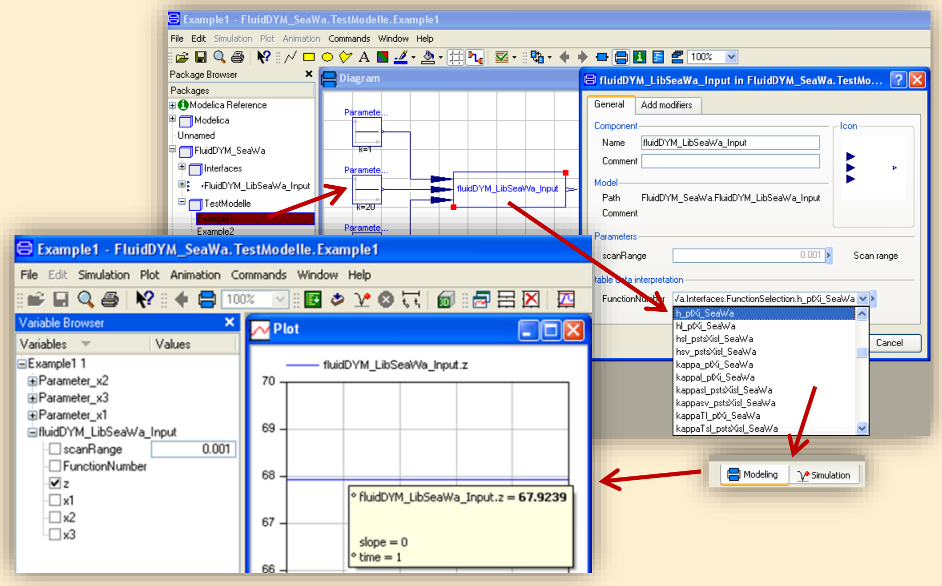
Add-On FluidVIEW for LabVIEW™

The property functions can be calculated in LabVIEW™.

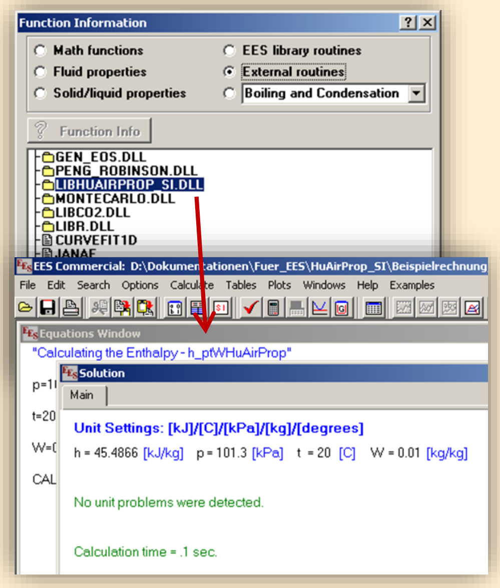


Add-In FluidDYM for DYMOLA® (Modelica) and SimulationX®

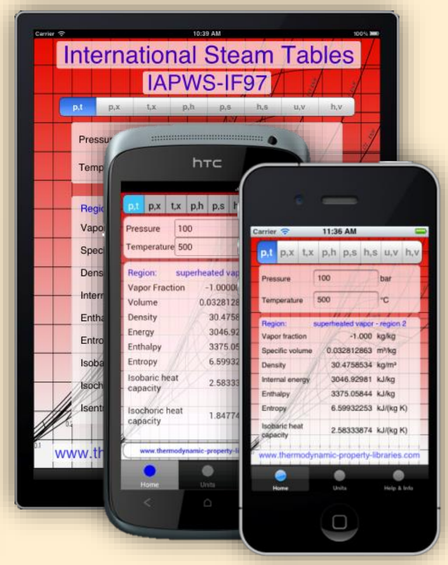
The property functions can be called in DYMOLA® and SimulationX®.



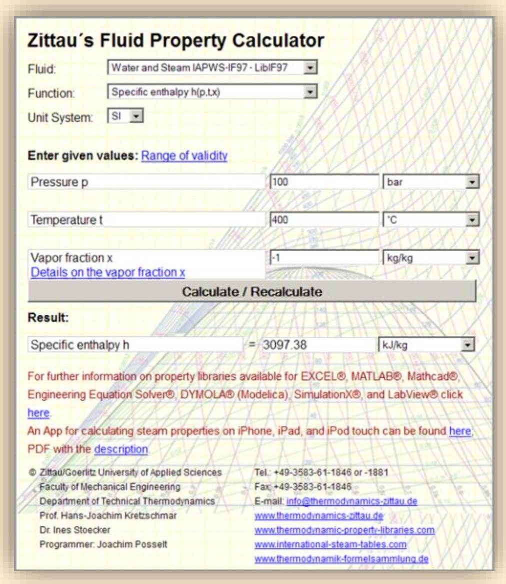
Add-In FluidEES for Engineering Equation Solver®



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



Online Property Calculator at www.thermofluidprop.com



Property Software for Pocket Calculators

FluidCasio


fx 9750 G II


CFX 9850
fx-GG20


CFX 9860 G
Graph 85


ALGEBRA
FX 2.0

FluidHP


HP 48


HP 49

FluidTI


TI Nspire CX CAS
TI Nspire CAS


TI 83
TI 84
TI 89


TI Voyage 200


TI 92

For more information please contact:

KCE-ThermoFluidProperties UG (limited liability) & Co. KG
Professor Hans-Joachim Kretzschmar

Wallotstr. 3
01307 Dresden, Germany

Internet: www.thermofluidprop.com
E-mail: info@thermofluidprop.com
Phone: +49-351-27597860
Mobile: +49-172-7914607
Fax: +49-3222-4262250

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

Thermodynamic Properties

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

Transport Properties

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

Backward Functions

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

Thermodynamic Derivatives

- Partial derivatives can be calculated.

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

www.thermofluidprop.com

5. References

[1] Kretzschmar, H.-J.:
Zur Aufbereitung und Darbietung thermophysikalischer Stoffdaten für die Energietechnik.
Habilitation, TU Dresden, Fakultät Maschinenwesen (1990)

[2] Span, R.; Wagner W.:
A New Equation of State for Carbon Dioxide Covering the Fluid Region from the Triple-Point Temperature to 1100 K at Pressures up to 800 MPa.
J. Phys. Chem. Ref. Data, 25, (1996) Nr. 6, S. 1506-1596

[3] Vesovic, V.; Wakeham, W. A.; Olchowy, G. A.; Sengers, J. V.; Watson, J. T. R.; Millat, J.:
The Transport Properties of Carbon Dioxide.
J. Phys. Chem. Ref. Data, 19, (1990) Nr. 3, S. 763-808

[4] Jäger, A.; Span, R.:
Equation of State for Solid Carbon Dioxide Based on the Gibbs Free Energy.
J. Chem. Eng. Data 57 (2012), 590-597

[5] Kuprianoff, J.:
Die feste Kohlensäure (Trockeneis) – Herstellung und Verwendung
Ferdinand Enke Verlag Stuttgart, 1953

6. Satisfied Customers

Date: 05/2018

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) and SimulationX®
- FluidVIEW for LabVIEW™.

2018

Universität Madrid, Madrid, Spanien	05/2018
HS Zittau/ Görlitz, Fakultät Wirtschaft, Zittau	05/2018
HS Niederrhein, Krefeld	05/2018
GRS, Köln	03/2018
RONAL AG, Härklingen, Schweiz	02/2018
Ingenieurbüro Leipert, Riegelsberg	02/2018
AIXPROCESS, Aachen	02/2018
KRONES, Neutraubling	02/2018
Doosan Lentjes, Ratingen	01/2018

2017

Compact Kältetechnik, Dresden	12/2017
Endress + Hauser Messtechnik GmbH +Co. KG, Hannover	12/2017
TH Mittelhessen, Gießen	11/2017
Haarslev Industries, Søndersø, Denmark	11/2017
Hochschule Zittau/Görlitz, Fachgebiet Energiesystemtechnik	11/2017
ATESTEO, Alsdorf	10/2017
Wijbenga, PC Geldermalsen, Netherlands	10/2017
Fels-Werke GmbH, Elbingerode	10/2017
KIT Karlsruhe, Institute für Neutronenphysik und Reaktortechnik	09/2017
Air-Consult, Jena	09/2017
Papierfabrik Koehler, Oberkirch	09/2017
ZWILAG, Würenlingen, Switzerland	09/2017
TLK-Thermo Universität Braunschweig, Braunschweig	08/2017
Fichtner IT Consulting AG, Stuttgart	07/2017
Hochschule Ansbach, Ansbach	06/2017
RONAL, Härkingen, Switzerland	06/2017
BORSIG Service, Berlin	06/2017

BOGE Kompressoren, Bielefeld	06/2017
STEAG Energy Services, Zwingenberg	06/2017
CES clean energy solutions, Wien, Austria	04/2017
Princeton University, Princeton, USA	04/2017
B2P Bio-to-Power, Wadersloh	04/2017
TU Dresden, Institute for Energy Engineering, Dresden	04/2017
SAINT-GOBAIN, Vaujours, France	03/2017
TU Bergakademie Freiberg, Chair of Thermodynamics, Freiberg	03/2017
SCHMIDT + PARTNER, Therwil, Switzerland	03/2017
KAESER Kompressoren, Gera	03/2017
F&R, Praha, Czech Republic	03/2017
ULT Umwelt-Lufttechnik, Löbau	02/2017
JS Energie & Beratung, Erding	02/2017
Kelvion Brazed PHE, Nobitz-Wilchwitz	02/2017
MTU Aero Engines, München	02/2017
Hochschule Zittau/Görlitz, IPM	01/2017
CombTec ProCE, Zittau	01/2017
SHELL Deutschland Oil, Wesseling	01/2017
MARTEC Education Center, Frederikshaven, Denmark	01/2017
SynErgy Thermal Management, Krefeld	01/2017

2016

BOGE Druckluftsysteme, Bielefeld	12/2016
BFT Planung, Aachen	11/2016
Midiplan, Bietigheim-Bissingen	11/2016
BBE Barnich IB	11/2016
Wenisch IB,	11/2016
INL, Idaho Falls	11/2016
TU Kältetechnik, Dresden	11/2016
Kopf SynGas, Sulz	11/2016
INTVEN, Bellevne (USA)	11/2016
DREWAG Dresden, Dresden	10/2016
AGO AG Energie+Anlagen, Kulmbach	10/2016
Universität Stuttgart, ITW, Stuttgart	09/2016
Pöyry Deutschland GmbH, Dresden	09/2016
Siemens AG, Erlangen	09/2016
BASF über Fichtner IT Consulting AG	09/2016
B+B Engineering GmbH, Magdeburg	09/2016
Wilhelm Büchner Hochschule, Pfungstadt	08/2016

Webasto Thermo & Comfort SE, Gliching	08/2016
TU Dresden, Dresden	08/2016
Endress+Hauser Messtechnik GmbH+Co. KG, Hannover	08/2016
D + B Kältetechnik, Althausen	07/2016
Fichtner IT Consulting AG, Stuttgart	07/2016
AB Electrolux, Krakow, Poland	07/2016
ENEXIO Germany GmbH, Herne	07/2016
VPC GmbH, Vetschau/Spreewald	07/2016
INWAT, Lodz, Poland	07/2016
E.ON SE, Düsseldorf	07/2016
Planungsbüro Waidhas GmbH, Chemnitz	07/2016
EEB Enerko, Aldershoven	07/2016
IHEBA Naturenergie GmbH & Co. KG, Pfaffenhofen	07/2016
SSP Kälteplaner AG, Wolfertschwenden	07/2016
EEB ENERKO Energiewirtschaftliche Beratung GmbH, Berlin	07/2016
BOGE Kompressoren Otto BOGE GmbH & Co KG, Bielefeld	06/2016
Universidad Carlos III de Madrid, Madrid, Spain	04/2016
INWAT, Lodzi, Poland	04/2016
Planungsbüro WAIDHAS GmbH, Chemnitz	04/2016
STEAG Energy Services GmbH, Laszlo Küppers, Zwingenberg	03/2016
WULFF & UMAG Energy Solutions GmbH, Husum	03/2016
FH Bielefeld, Bielefeld	03/2016
EWT Eckert Wassertechnik GmbH, Celle	03/2016
ILK Institut für Luft- und Kältetechnik GmbH, Dresden	02/2016, 06/2016 (2x)
IEV KEMA - DNV GV – Energie, Dresden	02/2016
Allborg University, Department of Energie, Aalborg, Denmark	02/2016
G.A.M. Heat GmbH, Gräfenhainichen	02/2016
Institut für Luft- und Kältetechnik, Dresden	02/2016, 05/2016, 06/2016
Bosch, Stuttgart	02/2016
INL Idaho National Laboratory, Idaho, USA	11/2016, 01/2016
Friedl ID, Wien, Austria	01/2016
Technical University of Dresden, Dresden	01/2016

2015

EES Enerko, Aachen	12/2015
Ruldolf IB, Strau, Austria	12/2015
Allborg University, Department of Energie, Aalborg, Denmark	12/2015
University of Lyubljana, Slovenia	12/2015
Steinbrecht IB, Berlin	11/2015
Universidad Carlos III de Madrid, Madrid, Spain	11/2015
STEAK, Essen	11/2015

Bosch, Lohmar	10/2015
Team Turbo Machines, Rouen, France	09/2015
BTC – Business Technology Consulting AG, Oldenburg	07/2015
KIT Karlsruhe Institute of Technology, Eggenstein-Leopoldshafen	07/2015
ILK, Dresden	07/2015
Schniewindt GmbH & Co. KG, Neuenwalde	08/2015

2014

PROJEKTPLAN, Dohna	04/2014
Technical University of Vienna, Austria	04/2014
MTU Aero Engines AG, Munich	04/2014
GKS, Schweinfurt	03/2014
Technical University of Nuremberg	03/2014
EP-E, Niederstetten	03/2014
Rückert NatUrgas GmbH, Lauf	03/2014
YESS-World, South Korea	03/2014
ZAB, Dessau	02/2014
KIT-TVT, Karlsruhe	02/2014
Stadtwerke Neuburg	02/2014
COMPAREX, Leipzig for RWE Essen	02/2014
Technical University of Prague, Czech Republic	02/2014
HS Augsburg	02/2014
Envi-con, Nuremberg	01/2014
DLR, Stuttgart	01/2014
Doosan Lentjes, Ratingen	01/2014
Technical University of Berlin	01/2014
Technical University of Munich	01/2014
Technical University of Braunschweig	01/2014
M&M Turbinentechnik, Bielefeld	01/2014

2013

TRANTER-GmbH, Artern	12/2013
SATAKE, Shanghai, China	12/2013
VOITH, Kunshan, China	12/2013
ULT, Löbau	12/2013
MAN, Copenhagen, Dänemark	11/2013
DREWAG, Dresden	11/2013
Haarslev Industries, Herlev, Dänemark	11/2013
STEAG, Herne	11/2013, 12/2013
Ingersoll-Rand, Oberhausen	11/2013
Wilhelm-Büchner HS, Darmstadt	10/2013

IAV, Chemnitz	10/2013
Technical University of Regensburg	10/2013
PD-Energy, Bitterfeld	09/2013
Thermofin, Heinsdorfergrund	09/2013
SHI, New Jersey, USA	09/2013
M&M Turbinentechnik, Bielefeld	08/2013
BEG-BHV, Bremerhaven	08/2013
TIG-Group, Husum	08/2013
COMPAREX, Leipzig	08/2013, 11/2013
for RWE Essen	12/2013
University of Budapest, Hungary	08/2013
Siemens, Frankenthal	08/2013, 10/2013
	11/2013
VGB, Essen	07/2013, 11/2013
Brunner Energieberatung, Zurich, Switzerland	07/2013
Technical University of Deggendorf	07/2013
University of Maryland, USA	07/2013, 08/2013
University of Princeton, USA	07/2013
NIST, Boulder, USA	06/2013
IGUS GmbH, Dresden	06/2013
BHR Bilfinger, Essen	06/2013
SÜDSALZ, Bad Friedrichshall	06/2013, 12/2013
Technician School of Berlin	05/2013
KIER, Gajeong-ro, Südkorea	05/2013
Schwing/Stetter GmbH, Memmingen	05/2013
Vattenfall, Berlin	05/2013
AUTARK, Kleinmachnow	05/2013
STEAG, Zwingenberg	05/2013
Hochtief, Düsseldorf	05/2013
University of Stuttgart	04/2013
Technical University -Bundeswehr, Munich	04/2013
Rerum Cognitio Forschungszentrum, Frankfurt	04/2013
Kältetechnik Dresen + Bremen, Alfhausen	04/2013
University Auckland, New Zealand	04/2013
MASDAR Institut, Abu Dhabi, United Arab Emirates	03/2013
Simpelkamp, Dresden	02/2013
VEO, Eisenhüttenstadt	02/2013
ENTEC, Auerbach	02/2013
Caterpillar, Kiel	02/2013
Technical University of Wismar	02/2013
Technical University of Dusseldorf	02/2013

ILK, Dresden	01/2013, 08/2013
Fichtner IT, Stuttgart	01/2013, 11/2013
Schnepf Ingeniuerbüro, Nagold	01/2013
Schütz Engineering, Wadgassen	01/2013
Endress & Hauser, Reinach, Switzerland	01/2013
Oschatz GmbH, Essen	01/2013
frischli Milchwerke, Rehburg-Loccum	01/2013

2012

Voith, Bayreuth	12/2012
Technical University of Munich	12/2012
Dillinger Huette	12/2012
University of Stuttgart	11/2012
Siemens, Muehlheim	11/2012
Sennheiser, Hannover	11/2012
Oschatz GmbH, Essen	10/2012
Fichtner IT, Stuttgart	10/2012, 11/2012
Helbling Technik AG, Zurich, Switzerland	10/2012
University of Duisburg	10/2012
Rerum Cognitio Forschungszentrum, Frankfurt	09/2012
Pöyry Deutschland GmbH, Dresden	08/2012
Extracciones, Guatemala	08/2012
RWE, Essen	08/2012
Weghaus Consulting Engineers, Wuerzburg	08/2012
GKS, Schweinfurt	07/2012
COMPAREX, Leipzig for RWE Essen	07/2012
GEA, Nobitz	07/2012
Meyer Werft, Papenburg	07/2012
STEAG, Herne	07/2012
GRS, Cologne	06/2012
Fichtner IT Consult, Chennai, India	06/2012
Siemens, Freiburg	06/2012
Nikon Research of America, Belmont, USA	06/2012
Niederrhein University of Applied Sciences, Krefeld	06/2012
STEAG, Zwingenberg	06/2012
Mainova, Frankfurt on Main via Fichtner IT Consult	05/2012
Endress & Hauser	05/2012
PEU, Espenheim	05/2012
Luzern University of Applied Sciences, Switzerland	05/2012

BASF, Ludwigshafen (general license) via Fichtner IT Consult	05/2012
SPX Balcke-Dürr, Ratingen	05/2012, 07/2012
Gruber-Schmidt, Wien, Austria	04/2012
Vattenfall, Berlin	04/2012
ALSTOM, Baden	04/2012
SKW, Piesteritz	04/2012
TERA Ingegneria, Trento, Italy	04/2012
Siemens, Erlangen	04/2012, 05/2012
LAWI Power, Dresden	04/2012
Stadtwerke Leipzig	04/2012
SEITZ, Wetzikon, Switzerland	03/2012, 07/2012
M & M, Bielefeld	03/2012
Sennheiser, Wedemark	03/2012
SPG, Montreuil Cedex, France	02/2012
German Destilation, Sprendlingen	02/2012
Lopez, Munguia, Spain	02/2012
Endress & Hauser, Hannover	02/2012
Palo Alto Research Center, USA	02/2012
WIPAK, Walsrode	02/2012
Freudenberg, Weinheim	01/2012
Fichtner, Stuttgart	01/2012
airinotec, Bayreuth	01/2012, 07/2012
University Auckland, New Zealand	01/2012
VPC, Vetschau	01/2012
Franken Guss, Kitzingen	01/2012

2011

XRG-Simulation, Hamburg	12/2011
Smurfit Kappa PPT, AX Roermond, Netherlands	12/2011
AWTEC, Zurich, Switzerland	12/2011
eins-energie, Bad Elster	12/2011
BeNow, Rodenbach	11/2011
Luzern University of Applied Sciences, Switzerland	11/2011
GMVA, Oberhausen	11/2011
CCI, Karlsruhe	10/2011
W.-Büchner University of Applied Sciences, Pfungstadt	10/2011
PLANAIR, La Sagne, Switzerland	10/2011
LAWI, Dresden	10/2011
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