Simulink Report: Vol_4D_

Christian Müller

2008-03-06

Model - Vol_4D_

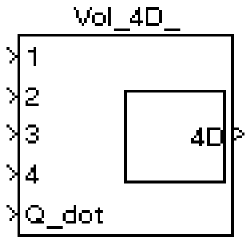


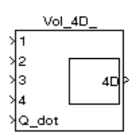
Tabelle 1.1. Vol_4D_ Simulation Parameters

Solver ode14x	*ZeroCross* on	*StartTime* 0.0*StopTime* 10.0
RelTol 1e-3	*AbsTol* auto	*Refine* 1
InitialStep auto	*FixedStep* auto	*MaxStep* auto

Tabelle 1.2. Vol_4D_ Summary Information

NumModelInputs	N/A	*NumModelOutputs*	N/A
NumVirtualSubsystems	N/A	*NumNonvirtSubsystems*	N/A
NumNonVirtBlocksInModel	N/A	*NumBlockTypeCounts*	N/A
NumBlockSignals	N/A	*NumBlockParams*	N/A
NumZCEvents	N/A	*NumNonsampledZCs*	N/A

Systems

Name	**Parent**	**Snapshot**	**Blocks**	**Signals**
Vol_4D_	<root>		Vol_4D_	Vol_4D_<1>

Blocks

Name	**Parent**	**V**	**P init**	**T init**	**X H2O gas init**	**X CO2 init**	**X H2O liq init**
Vol_4D_	Vol_4D_ [1]	NaN	NaN	NaN	NaN	NaN	NaN

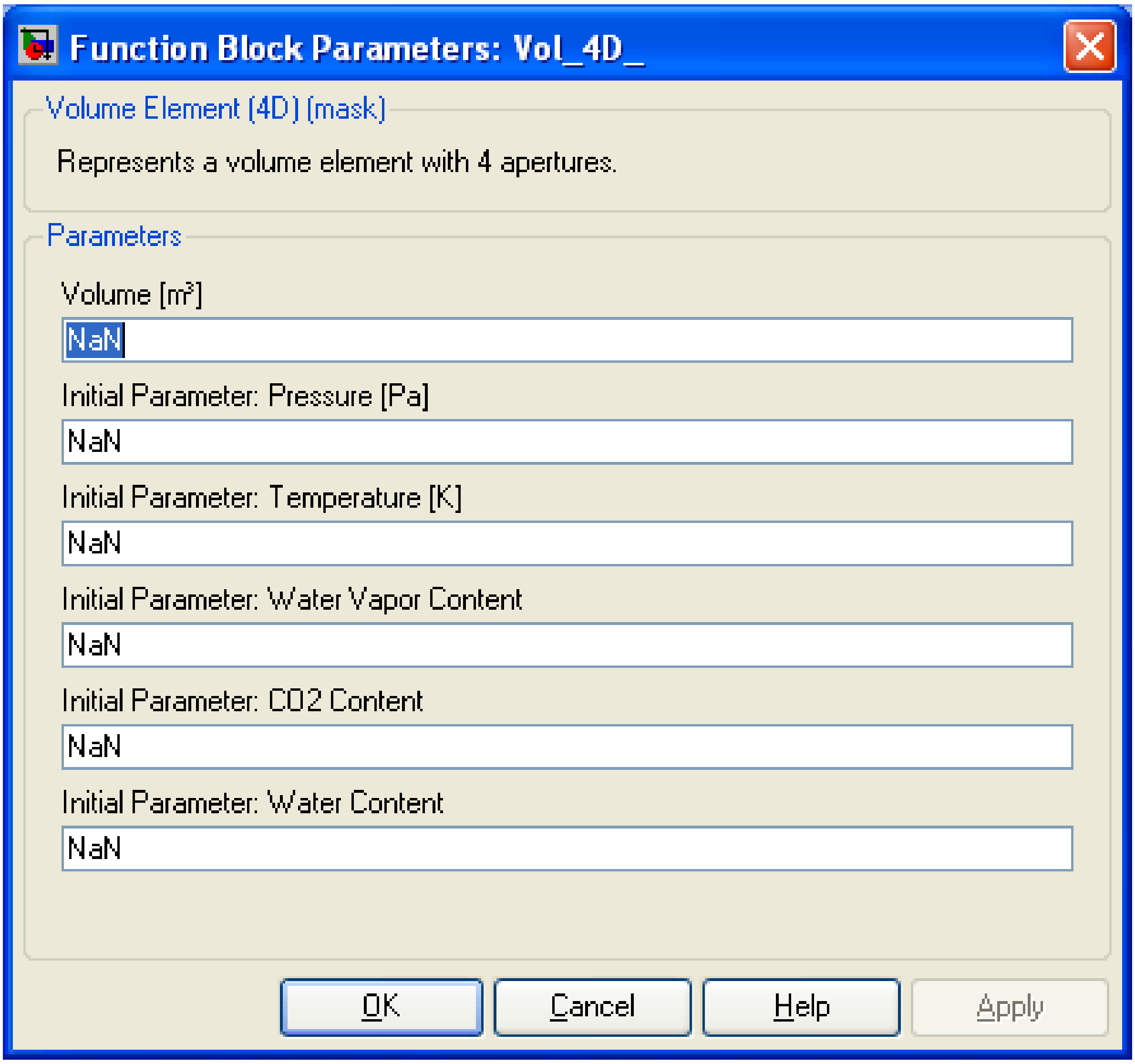
Tabelle 1.3. Block Type Count

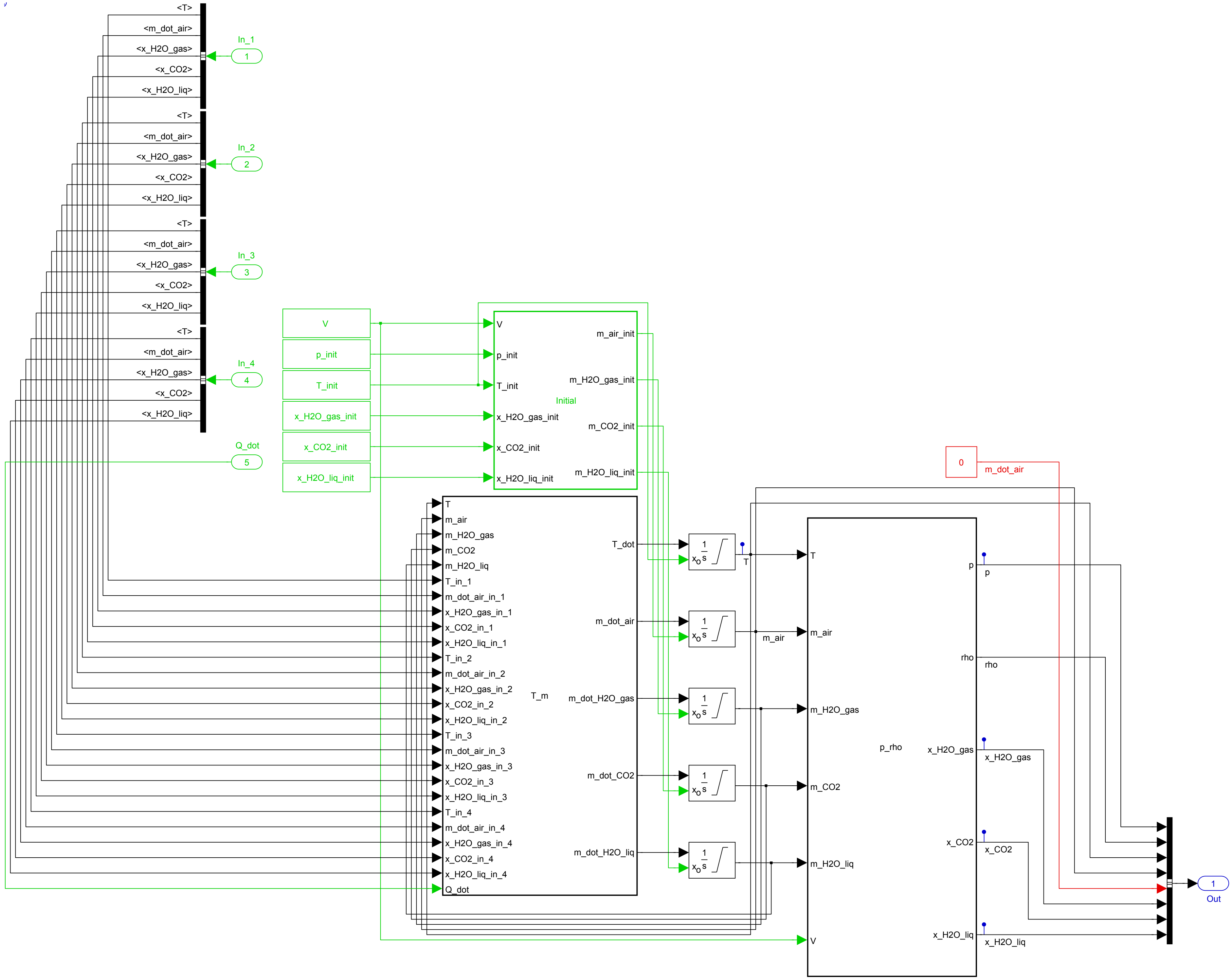
BlockType	Count	Block Names
Inport	43	In_1, In_2, In_3, In_4, Q_dot, V, p_init, T_init, x_H2O_gas_init, x_CO2_init, x_H2O_liq_init, T, m_air, m_H2O_gas, m_CO2, m_H2O_liq, T_in_1, m_dot_air_in_1, x_H2O_gas_in_1, x_CO2_in_1, x_H2O_liq_in_1, T_in_2, m_dot_air_in_2, x_H2O_gas_in_2, x_CO2_in_2, x_H2O_liq_in_2, T_in_3, m_dot_air_in_3, x_H2O_gas_in_3, x_CO2_in_3, x_H2O_liq_in_3, T_in_4, m_dot_air_in_4, x_H2O_gas_in_4, x_CO2_in_4, x_H2O_liq_in_4, Q_dot, T, m_air, m_H2O_gas, m_CO2, m_H2O_liq, V
Outport	15	m_air_init, m_H2O_gas_init, m_CO2_init, m_H2O_liq_init, T_dot, m_dot_air, m_dot_H2O_gas, m_dot_CO2, m_dot_H2O_liq, p, rho, x_H2O_gas, x_CO2, x_H2O_liq, Out
Constant	7	T_init, V, m_dot_air, p_init, x_CO2_init, x_H2O_gas_init, x_H2O_liq_init
Integrator	5	Integrator1, Integrator2, Integrator3, Integrator4, Integrator5
BusSelector	4	Bus Selector1, Bus Selector2, Bus Selector3, Bus Selector5
Terminator	3	Terminator , Terminator , Terminator
Stateflow (m)	3	Embedded MATLAB Function, Embedded_MATLAB_Function_1, Embedded_MATLAB_Function_2
S-Function	3	SFunction , SFunction , SFunction
Demux	3	Demux , Demux , Demux
Volume Element (4D) (m)	1	Vol_4D_
BusCreator	1	Bus Creator1

Data and Functions

Tabelle 1.4. Model Functions

Function Name	Parent Blocks	Calling string
NaN	Vol_4D_ Vol_4D_ Vol_4D_ Vol_4D_ Vol_4D_ Vol_4D_	`NaN` `NaN` `NaN` `NaN` `NaN` `NaN`


Function Block Parameters: Vol_4D_
Volume Element (4D) (mask)
Represents a volume element with 4 apertures.
Parameters
Volume [m³]
NaN
Initial Parameter: Pressure [Pa]
NaN
Initial Parameter: Temperature [K]
NaN
Initial Parameter: Water Vapor Content
NaN
Initial Parameter: CO2 Content
NaN
Initial Parameter: Water Content
NaN
OK
Cancel
Help
Apply



<T>
<m_dot_air>
<x_H2O_gas>
<x_CO2>
<x_H2O_liq>
In_1
1
In_2
2
In_3
3
In_4
4
Q_dot
5
V
p_init
T_init
x_H2O_gas_init
x_CO2_init
x_H2O_liq_init
Initial
m_air_init
m_H2O_gas_init
m_CO2_init
m_H2O_liq_init
T
m_air
m_H2O_gas
m_CO2
m_H2O_liq
T_in_1
m_dot_air_in_1
x_H2O_gas_in_1
x_CO2_in_1
x_H2O_liq_in_1
T_in_2
m_dot_air_in_2
x_H2O_gas_in_2
x_CO2_in_2
x_H2O_liq_in_2
T_in_3
m_dot_air_in_3
x_H2O_gas_in_3
x_CO2_in_3
x_H2O_liq_in_3
T_in_4
m_dot_air_in_4
x_H2O_gas_in_4
x_CO2_in_4
x_H2O_liq_in_4
Q_dot
T_m
T_dot
m_dot_air
m_dot_H2O_gas
m_dot_CO2
m_dot_H2O_liq
1/s
x o
0
m_dot_air
p_rho
p
rho
x_H2O_gas
x_CO2
x_H2O_liq
Out

```
function [m_air_init,m_H2O_gas_init,m_CO2_init,m_H2O_liq_init] = Initial(V,p_init,↙
T_init,x_H2O_gas_init,x_CO2_init,x_H2O_liq_init)

% ************************************************************************
% * Definition of a volume with 4 apertures
% *
% * Number of Inputs:        5
% *
% * Parameter : Volume:      V
% *
% *
% * Relevant input variables of Vol_4D:
% *
% * Temperature:             T
% * Mass flow dry air:       m_dot_air
% * Content water vapor:     x_H2O_gas
% * Content CO2:             x_CO2
% * Content water:           x_H2O_liq
% * External Heat Flow:      Q_dot
% *
% *
% * Relevant output variables of Vol_4D:
% *
% * Pressure:                p
% * Density:                 rho
% * Temperature:             T
% * Mass dry air:            m_air
% * Content water vapor:     x_H2O_gas
% * Content CO2:             x_CO2
% * Content water:           x_H2O_liq
% *
% ***********************************************************************
% * Embedded MATLAB Function Initial:
% *
% * Calculations:
% * 1. Definition specific gas constants.
% * 2. Calculation of the saturation density by a given temperature.
% * 3. Calculation of the saturation mass.
% * 4. Initialisation.
% *
% *
% *
% * Assumptions:
% * 1. The gas mixture inside the volume consists of water vapor (H2O_gas),
% *    CO2 and water (H2O_liq, state: fog).
% * 2. The saturation density is a function of the temperature. The used
% *    function is described in literature.
% *
% *
% * Last modification : 15.03.2008
% * Author : Christian Müller(HAW)
% *
% ***********************************************************************

% * 1. Definition specific gas constants
R_air             = 287.058;
```

```
R_H2O_gas          = 461.523;
R_CO2              = 188.924;
% ***********************************************************************

% * 2. Calculation of the saturation density by a given temperature
rho_H2O_gas_sat    = 4.44259*exp(15.05703*(T_init-273.15)/(208.07254+(T_init-273.15)))↙
/1000;
% ***********************************************************************

% * 3. Calculation of the saturation mass
m_H2O_gas_sat      = V*rho_H2O_gas_sat;
% ***********************************************************************

% * 4. Initialisation
m_air_init         = p_init*V/(T_init*↙
(R_air+x_H2O_gas_init*R_H2O_gas+x_CO2_init*R_CO2));
m_H2O_gas_init     = m_air_init*x_H2O_gas_init;

if m_H2O_gas_init > m_H2O_gas_sat
   m_H2O_gas_init = m_H2O_gas_sat;
   m_air_init     = ((p_init*V/T_init)-(m_H2O_gas_init*R_H2O_gas))/↙
(R_air+x_CO2_init*R_CO2);
end

m_CO2_init         = m_air_init*x_CO2_init;
m_H2O_liq_init     = m_air_init*x_H2O_liq_init;
% ***********************************************************************
```

```
function [T_dot,m_dot_air,m_dot_H2O_gas,m_dot_CO2,m_dot_H2O_liq] = T_m(T,m_air,↙
m_H2O_gas,m_CO2,m_H2O_liq,T_in_1,m_dot_air_in_1,x_H2O_gas_in_1,x_CO2_in_1,↙
x_H2O_liq_in_1,T_in_2,m_dot_air_in_2,x_H2O_gas_in_2,x_CO2_in_2,x_H2O_liq_in_2,T_in_3,↙
m_dot_air_in_3,x_H2O_gas_in_3,x_CO2_in_3,x_H2O_liq_in_3,T_in_4,m_dot_air_in_4,↙
x_H2O_gas_in_4,x_CO2_in_4,x_H2O_liq_in_4,Q_dot)

% **************************************************************************
% * Definition of a volume with 4 apertures
% *
% * Number of Inputs:         5
% *
% * Parameter : Volume:       V
% *
% *
% * Relevant input variables of Vol_4D:
% *
% * Temperature:              T
% * Mass flow dry air:        m_dot_air
% * Content water vapor:      x_H2O_gas
% * Content CO2:              x_CO2
% * Content water:            x_H2O_liq
% * External Heat Flow:       Q_dot
% *
% *
% * Relevant output variables of Vol_4D:
% *
% * Pressure:                 p
% * Density:                  rho
% * Temperature:              T
% * Mass dry air:             m_air
% * Content water vapor:      x_H2O_gas
% * Content CO2:              x_CO2
% * Content water:            x_H2O_liq
% *
% *************************************************************************
% * Embedded MATLAB Function T_m:
% *
% * Calculations:
% * 1. Definition specific gas constants.
% * 2. Modification of the mass inside the system.
% * 3. Redefinition of the input variables.
% * 4. Modification of the temperature inside the system.
% *
% *
% * Assumptions:
% * 1. The gas mixture inside the volume consists of water vapor (H2O_gas),
% *    CO2 and water (H2O_liq, state: fog).
% * 2. The increase or decrease of the mass inside the volume is defined
% *    by the incoming (sign: plus) or outgoing (sign: minus) mass flow.
% * 3. In the case of outgoing mass flow, the characteristic variables of
% *    the mass flow are defined by the state variables of the volume.
% * 4. The Enthalpy equation of the gas inside the volume is used. the
% *    enthalpy of the liquid water inside the volume is neglected.
% *    Assuming that the liquid water is always in thermal equilibrium with
% *    the gas.
% *
```

```
% *
% * Last modification : 15.03.2008
% * Author : Christian Müller(HAW)
% *
% ***********************************************************************

% * 1. Definition specific gas constants
R_air             = 287.058;
R_H2O_gas         = 461.523;
R_CO2             = 188.924;
c_p_air           = 1005;
c_p_H2O_gas       = 1870;
c_p_CO2           = 830;
c_v_air           = c_p_air-R_air;
c_v_H2O_gas       = c_p_H2O_gas-R_H2O_gas;
c_v_CO2           = c_p_CO2-R_CO2;
% ***********************************************************************

% * 2. Redefinition of the input variables
if m_dot_air_in_1 < 0
   T_in_1         = T;
   m_dot          = m_dot_air_in_1*(1+x_H2O_gas_in_1+x_CO2_in_1);
   m_dot_air_in_1 = m_dot*(1/(1+(m_H2O_gas/m_air)+(m_CO2/m_air)));
   x_H2O_gas_in_1 = (m_H2O_gas/m_air);
   x_CO2_in_1     = (m_CO2/m_air);
   x_H2O_liq_in_1 = (m_H2O_liq/m_air);
end

if m_dot_air_in_2 < 0
   T_in_2         = T;
   m_dot          = m_dot_air_in_2*(1+x_H2O_gas_in_2+x_CO2_in_2);
   m_dot_air_in_2 = m_dot*(1/(1+(m_H2O_gas/m_air)+(m_CO2/m_air)));
   x_H2O_gas_in_2 = (m_H2O_gas/m_air);
   x_CO2_in_2     = (m_CO2/m_air);
   x_H2O_liq_in_2 = (m_H2O_liq/m_air);
end

if m_dot_air_in_3 < 0
   T_in_3         = T;
   m_dot          = m_dot_air_in_3*(1+x_H2O_gas_in_3+x_CO2_in_3);
   m_dot_air_in_3 = m_dot*(1/(1+(m_H2O_gas/m_air)+(m_CO2/m_air)));
   x_H2O_gas_in_3 = (m_H2O_gas/m_air);
   x_CO2_in_3     = (m_CO2/m_air);
   x_H2O_liq_in_3 = (m_H2O_liq/m_air);
end

if m_dot_air_in_4 < 0
   T_in_4         = T;
   m_dot          = m_dot_air_in_4*(1+x_H2O_gas_in_4+x_CO2_in_4);
   m_dot_air_in_4 = m_dot*(1/(1+(m_H2O_gas/m_air)+(m_CO2/m_air)));
   x_H2O_gas_in_4 = (m_H2O_gas/m_air);
   x_CO2_in_4     = (m_CO2/m_air);
   x_H2O_liq_in_4 = (m_H2O_liq/m_air);
end
% ***********************************************************************
```

```
% * 3. Modification of the mass inside the system
m_dot_air        = m_dot_air_in_1+m_dot_air_in_2+m_dot_air_in_3+m_dot_air_in_4;
m_dot_H2O_gas    =↙
m_dot_air_in_1*x_H2O_gas_in_1+m_dot_air_in_2*x_H2O_gas_in_2+m_dot_air_in_3*x_H2O_gas_in↙
_3+m_dot_air_in_4*x_H2O_gas_in_4;
m_dot_CO2        =↙
m_dot_air_in_1*x_CO2_in_1+m_dot_air_in_2*x_CO2_in_2+m_dot_air_in_3*x_CO2_in_3+m_dot_air↙
_in_4*x_CO2_in_4;
m_dot_H2O_liq    =↙
m_dot_air_in_1*x_H2O_liq_in_1+m_dot_air_in_2*x_H2O_liq_in_2+m_dot_air_in_3*x_H2O_liq_in↙
_3+m_dot_air_in_4*x_H2O_liq_in_4;
% *************************************************************************

% * 4. Modification of the temperature inside the system
T_dot            = (Q_dot+m_dot_air_in_1*(c_p_air*T_in_1-c_v_air*T)+m_dot_air_in_2*↙
(c_p_air*T_in_2-c_v_air*T)+m_dot_air_in_3*(c_p_air*T_in_3-c_v_air*T)+m_dot_air_in_4*↙
(c_p_air*T_in_4-c_v_air*T)+m_dot_air_in_1*x_H2O_gas_in_1*(c_p_H2O_gas*T_in_1-↙
c_v_H2O_gas*T)+m_dot_air_in_2*x_H2O_gas_in_2*(c_p_H2O_gas*T_in_2-c_v_H2O_gas*T)↙
+m_dot_air_in_3*x_H2O_gas_in_3*(c_p_H2O_gas*T_in_3-c_v_H2O_gas*T)↙
+m_dot_air_in_4*x_H2O_gas_in_4*(c_p_H2O_gas*T_in_4-c_v_H2O_gas*T)↙
+m_dot_air_in_1*x_CO2_in_1*(c_p_CO2*T_in_1-c_v_CO2*T)+m_dot_air_in_2*x_CO2_in_2*↙
(c_p_CO2*T_in_2-c_v_CO2*T)+m_dot_air_in_3*x_CO2_in_3*(c_p_CO2*T_in_3-c_v_CO2*T)↙
+m_dot_air_in_4*x_CO2_in_4*(c_p_CO2*T_in_4-c_v_CO2*T))/↙
(c_v_air*m_air+c_v_H2O_gas*m_H2O_gas+c_v_CO2*m_CO2);
% *************************************************************************
```

```
function [p,rho,x_H2O_gas,x_CO2,x_H2O_liq] = p_rho(T,m_air,m_H2O_gas,m_CO2,m_H2O_liq,V)

% ***************************************************************************
% * Definition of a volume with 4 apertures
% *
% * Number of Inputs:        5
% *
% * Parameter : Volume:      V
% *
% *
% * Relevant input variables of Vol_4D:
% *
% * Temperature:             T
% * Mass flow dry air:       m_dot_air
% * Content water vapor:     x_H2O_gas
% * Content CO2:             x_CO2
% * Content water:           x_H2O_liq
% * External Heat Flow:      Q_dot
% *
% *
% * Relevant output variables of Vol_4D:
% *
% * Pressure:                p
% * Density:                 rho
% * Temperature:             T
% * Mass dry air:            m_air
% * Content water vapor:     x_H2O_gas
% * Content CO2:             x_CO2
% * Content water:           x_H2O_liq
% *
% **************************************************************************
% * Embedded MATLAB Function p_rho:
% *
% * Calculations:
% * 1. Definition specific gas constants.
% * 2. Calculation of the density.
% * 3. Calculation of the pressure.
% * 4. Calculation of the water vapor content, CO2 content and
% *    water content.
% *
% *
% * Assumptions:
% * 1. The gas mixture inside the volume consists of water vapor (H2O_gas),
% *    CO2 and water (H2O_liq, state: fog).
% * 4. The differnt contents are defined in respect of the mass of the
% *    dry air.
% *
% *
% * Last modification : 15.03.2008
% * Author : Christian Müller(HAW)
% *
% **************************************************************************

% * 1. Definition specific gas constants
R_air     = 287.058;
R_H2O_gas = 461.523;
```

```
R_CO2     = 188.924;
% *************************************************************************

% * 2. Calculation of the density
rho       = (m_air+m_H2O_gas+m_CO2)/V;
% *************************************************************************

% * 3. Calculation of the pressure
p         = T*(m_air*R_air+m_H2O_gas*R_H2O_gas+m_CO2*R_CO2)/V;
% *************************************************************************

% * 4. Calculation of the water vapor content, CO2 content and
% *    water content
x_H2O_gas = m_H2O_gas/m_air;
x_CO2     = m_CO2/m_air;
x_H2O_liq = m_H2O_liq/m_air;
% *************************************************************************
```