Report on Thermal

Generated by MTT using :
(mtt -u -q -q Thermal rep pdf)

Tue Aug 19 15:42:55 BST 2003

Contents

List of Figures

Part I

GasTurbines

Chapter 1

SimpleGasTurbine

1.1 SimpleGasTurbine_abg.tex *(-o -ss)*

MTT command:

```
mtt -o -ss SimpleGasTurbine abg tex
```

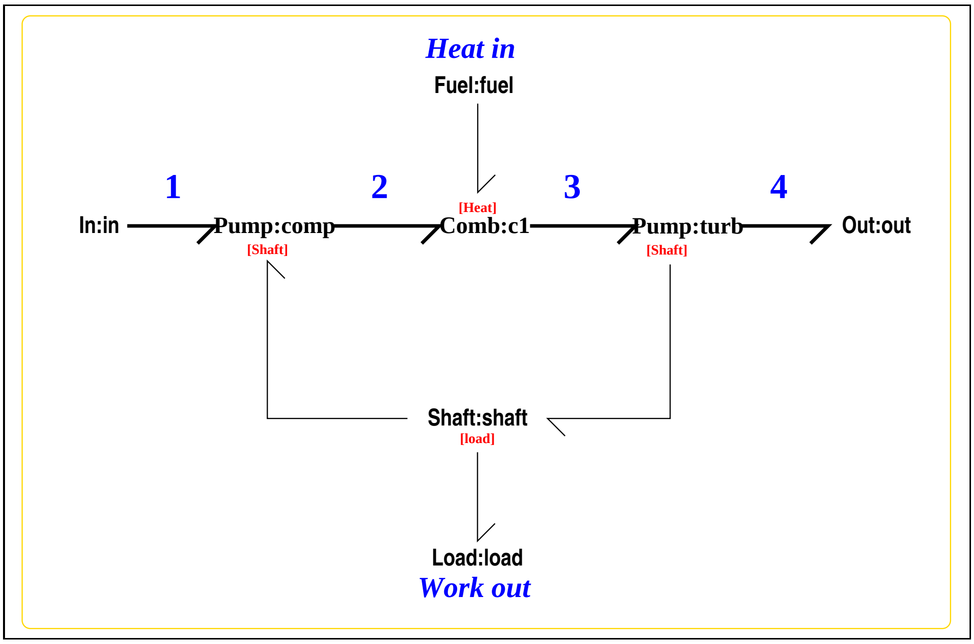


Figure 1.1: System **SimpleGasTurbine**: acausal bond graph

The acausal bond graph of system **SimpleGasTurbine** is displayed in Figure 1.1 (on page 9) and its label file is listed in Section 1.1.1 (on page 11). The subsystems are listed in Section 1.1.2 (on page 15).

SimpleGasTurbine can be regarded as an single-spool gas turbine (producing shaft power) with an ideal-gas working fluid. It corresponds to the simple Joule Cycle as described in Chapter 12 of Rogers and Mayhew and in Chapter 2 of Cohen, Rogers and Saravanamutto. However, unlike those examples, the system is written with dynamics in mind.

The system is described using an energy Bond Graph- this ensures that the first law is observed. In particular transformers are used to explicitly convert between energy covariables. Although this is a simple model, I believe that it provides the basis for building complex thermodynamic systems involving gas power cycles.

There are five main components:

1. p1 – a **Pump** component representing the compressor stage. This converts shaft work to energy flow in the working fluid.
2. c1 – a **Comb** component representing the combustion chamber. This converts the heat obtained by burning fuel to energy flow in the working fluid.
3. t1 – a **Turb** component representing the turbine component. This converts the energy flow in the working fluid to shaft work
4. j_s – an **I** component representing the combined inertia of the shaft and compressor and turbine rotors.
5. a **Load** component to absorb the shaft power.

The components **In** and **Out** provide the inlet and outlet conditions.

Both **Pump** and **Turb** are implemented with the *polytropic* constitutive relationship with index n. When $n = \gamma = \frac{c_p}{c_v}$ this corresponds to isentropic compression and expansion and thus the **SimpleGasTurbine** achieves its cycle efficiency. However, other values of n can be used to account for isentropic efficiency of less than unity.

To obtain a very simple dynamic model (and to avoid the need for an accurate combustion chamber model) there are no dynamics associated with the combustion chamber, but rahter it is assumed that the corresponding temperature is imposed on the component (that is T_3 is the system input) the corresponding heat flow is then an output.

Both heat input and work output are measured using the **PS** (power sensor) component, that for work output is embedded in the **Load** component. These can be monitored to give the efficiency of the **SimpleGasTurbine**.

A symbolic steady-state for the model was computed – see Section **??**. In particular, the load resistance was chosen to absorb all the generated work at the steady state and the shaft inertia was chosen to give a unit time constant for the linearised system. The mass flow and shaft speeds were taken as unity.

For the purposed of simulation, the numerical values given in Examples 12.1 of Chapter 12 of Rogers and Mayhew, except that the isentropic efficiencies are 100% ($n = \gamma$) – see Section **??**.

Simulations were performed starting at the steady state and increasing the combustion chamber temperature by 10% at $t = 1$ and reducing by 10% at $t = 5$. Graphs of the various outputs are plotted:

- Figure **??** (on page **??**) – the temperatures at the output of the
  - compressor,
  - combustion chamber and
  - turbine
- Figure **??** (on page **??**) – the heat input and work output
- Figure **??** (on page **??**) – the shaft speed and
- Figure **??** (on page **??**) – the pressure at the output of the
  - compressor,
  - combustion chamber and
  - turbine

This model can be modified extended in various ways to yield related dynamic systems. For example:

- an air cooler is obtained by changing the direction of heat and work flows
- additional **Turb** and **Comb** components add reheat to the cycle
- an isentropic nozzle can be added and the work output removed to give a jet engine.

1.1.1 Summary information

System SimpleGasTurbine::single-spool gas turbine producing shaft power SimpleGasTurbine can be regarded as an single-spool gas turbine (producing shaft power) with an ideal-gas working fluid. It corresponds to the simple Joule Cycle as described in Chapter 12 of Rogers and Mayhew and in Chapter 2 of Cohen, Rogers and Saravanamutto. However, unlike those examples, the system is written with dynamics in mind.

Interface information:

Component Comb is in library **CompressibleFlow/Comb**

Component Pump is in library **CompressibleFlow/Pump**

Variable declarations:

c_p

gamma_0

mdot

mom_0

omega_0

p_2

p_3

p_4

q_0

r_p

t_2

t_3

t_4

w_0

Units declarations:

This component has no UNITs declarations

The label file: SimpleGasTurbine_lbl.txt

```
#SUMMARY SimpleGasTurbine: single-spool gas turbine producing shaft powe
#DESCRIPTION SimpleGasTurbine can be regarded as an single-spool gas
#DESCRIPTION turbine (producing shaft power) with an ideal-gas working f
#DESCRIPTION corresponds to the simple Joule Cycle as described in Chapt
#DESCRIPTION Rogers and Mayhew and in Chapter 2 of Cohen, Rogers and
#DESCRIPTION Saravanamutto. However, unlike those examples, the system i
#DESCRIPTION written with dynamics in mind.

## Explicitly copy appropriate components
#ALIAS Pump CompressibleFlow/Pump
#ALIAS Comb CompressibleFlow/Comb

#PAR t_2
#PAR t_3
#PAR t_4
#PAR p_2
#PAR p_3
#PAR p_4
#PAR mdot
#PAR gamma_0
#PAR q_0
#PAR w_0
#PAR omega_0
#PAR r_p
#PAR c_p
#PAR mom_0

#NOTPAR  density
#NOTPAR ideal_gas
#NOTPAR q_0

## Label file for system SimpleGasTurbine (SimpleGasTurbine_lbl.txt)

# ###############################################################
# ## Version control history
# ###############################################################
# ## $Id: SimpleGasTurbine_lbl.txt,v 1.6 2003/06/11 16:10:26 gawthrop Ex
# ## $Log: SimpleGasTurbine_lbl.txt,v $
# ## Revision 1.6  2003/06/11 16:10:26  gawthrop
```

```
# ## Updated examples for latest MTT.
# ##
# ## Revision 1.5  2000/12/28 18:08:28  peterg
# ## To RCS
# ##
# ## Revision 1.4  1998/07/30 15:27:42  peterg
# ## Use #VAR inplace of dummy component.
# ##
# ## Revision 1.3  1998/07/03 14:54:45  peterg
# ## k_p --> k
# ## k_t --> k
# ##
# ## Revision 1.2  1998/07/03 14:53:38  peterg
# ## Renames tank to comb to be consistent.
# ##
# ## Revision 1.1  1998/05/18 15:46:02  peterg
# ## Initial revision
# ##
# ###############################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank

# Component type Fuel
fuel

# Component type In
in

# Component type Pump
comp none c_v;density,ideal_gas,r;alpha;effort,k

# Component type Comb
c1 none m_c;v_c;r

# Component type Pump
turb none c_v;density,ideal_gas,r;alpha;effort,k

# Component type Out
```

```
out

# Component type Shaft
        shaft none j_s

# Component type Load
load none r_l
```

1.1.2 Subsystems

- Comb: Combustion chamber model (1)
  - hPipe: Pipe for compressible fluid with heat transfer and heat storage. (1)
- Fuel (1)
  - Df Simple flow detector (1)
  - Se Simple effort source (1)
- In: Inflow conditions (1) No subsystems.
- Load (1)
  - Df Simple flow detector (1)
- Out: Outflow conditions (1) No subsystems.
- Pump Ideal pump component for compressible flow (2)
  - Poly - computes polytropic expansion temperature. (1)
  - wPipe Isentropic pipe with work transfer. (1)
- Shaft (3)
  - Df Simple flow detector (1)

1.1.3 Comb

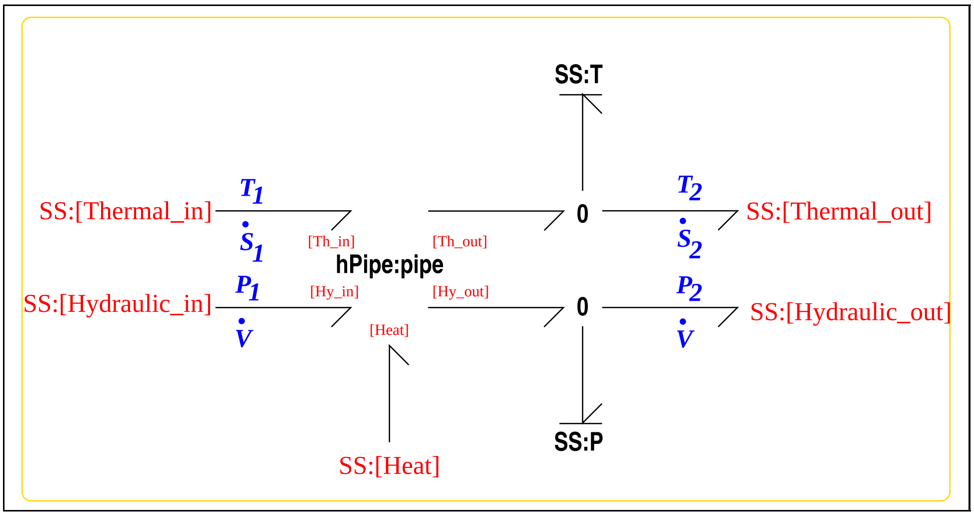


Figure 1.2: System **Comb**: acausal bond graph

The acausal bond graph of system **Comb** is displayed in Figure 1.2 (on page 16) and its label file is listed in Section 1.1.3 (on page 16). The subsystems are listed in Section 1.1.3 (on page 19).

This thermal tank model has been developed to represent an ideal (and non-dynamic) combustion chamber for a gas turbine. The major simplification is that the mass contained in the tank is assumed constant – this is consistent with using an ideal compressor and an ideal turbine with identical mass flows. Energy conservatioin is ensured by using true bonds and **TF** components.

Summary information

System Comb::Combustion chamber model This thermal tank model has been developed to represent an ideal (and non-dynamic) combustion chamber for a gas turbine. The major simplification is that the mass contained in the tank is assumed constant – this is consistent with using an ideal compressor and an ideal turbine with identical mass flows. Energy conservation is ensured by using true bonds and components.

Interface information:

Component hPipe is in library **CompressibleFlow/hPipe**

Parameter $1 represents actual parameter **m_c**

Parameter $2 represents actual parameter **v_c**

Parameter $3 represents actual parameter **r**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Comb_lbl.txt

```
%SUMMARY Comb: Combustion chamber model

%DESCRIPTION   This thermal tank model has been developed to represent a
%DESCRIPTION   (and non-dynamic) combustion chamber for a gas turbine. T
%DESCRIPTION   simplification is that the mass contained in the tank is
%DESCRIPTION   constant -- this is consistent with using an ideal compre
%DESCRIPTION   ideal turbine with identical mass flows. Energy conservat
%DESCRIPTION   ensured by using true bonds and components.

%ALIAS in Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out
```

```
%ALIAS $1 m_c
%ALIAS $2 v_c
%ALIAS $3 r

%ALIAS hPipe CompressibleFlow/hPipe

%% Label file for system Comb (Comb_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Comb_lbl.txt,v 1.7 1998/07/17 16:45:00 peterg Exp $
% %% $Log: Comb_lbl.txt,v $
% %% Revision 1.7  1998/07/17 16:45:00  peterg
% %% Alias for hPipe
% %%
% %% Revision 1.6  1998/07/04 08:24:25  peterg
% %% New-style SS
% %%
% %% Revision 1.5  1998/07/03 14:55:33  peterg
% %% Aliased parameters.
% %% Removed _c from parameters,
% %%
% %% Revision 1.4  1998/07/02 19:46:34  peterg
% %% New aliases
% %%
% %% Revision 1.3  1998/07/02 10:54:42  peterg
% %% Lower case in out
% %%
% %% Revision 1.2  1998/07/02 10:49:32  peterg
% %% Added port aliases
% %%
% %% Revision 1.1  1998/05/19 09:11:29  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
```

```
% blank



% Component type SS
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Heat] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
P SS external,0
T SS external,0

% Component type hPipe
pipe none m_c;v_c;r
```

Subsystems

- hPipe: Pipe for compressible fluid with heat transfer and heat storage. (1) No subsystems.

1.1.4 Density

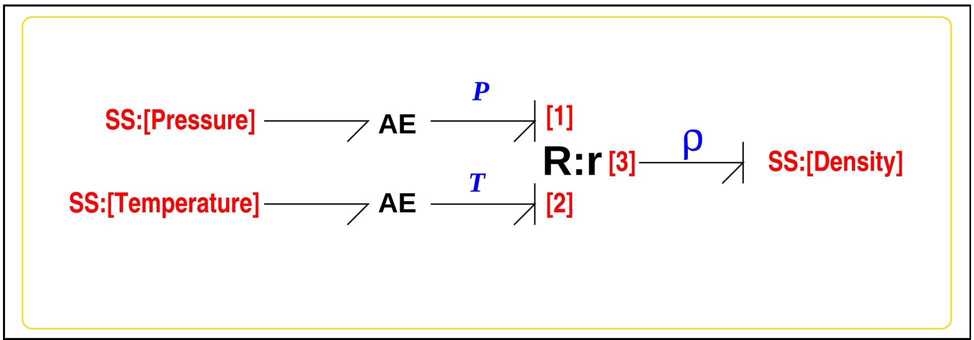


Figure 1.3: System **Density**: acausal bond graph

The acausal bond graph of system **Density** is displayed in Figure 1.3 (on page 19) and its label file is listed in Section 1.1.4 (on page 20).

This three port component computes the density ρ of substance in terms of the temperature T and the pressure P. At the moment, there are four forms dependent on the four forms of the component parameter:

1. *density,incompressible,rho*
2. *specific_volume,incompressible,rho*
3. *density,ideal_gas,R*
4. *specific_volume,ideal_gas,R*

$$y = \begin{cases} \rho & \text{if the parameter is } \textit{density,incompressible,rho} \\ \frac{1}{\rho} & \text{if the parameter is } \textit{specific\_volume,incompressible,rho} \\ \frac{P}{RT} & \text{if the parameter is } \textit{density,ideal\_gas,R} \\ \frac{RT}{P} & \text{if the parameter is } \textit{specific\_volume,ideal\_gas,R} \end{cases} \tag{1.1}$$

where ρ is the density of the incompressible fluid and R the universal gas constant of the ideal gas.

Summary information

System Density:- Computes P and T. Parameter: density,ideal_gas,gas_constant OR : specific_volume,ideal_gas,gas_constant OR : density,incompressible,gas_constant OR : specific_volume,incompressible,gas_constant Port [P]: Pressure Port [T]: Temperature Port [rho]: Density

Interface information:

Parameter $1 represents actual parameter **density,ideal_gas,r**

Port P represents actual port **Pressure**

Port T represents actual port **Temperature**

Port out represents actual port **Density**

Port rho represents actual port **Density**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Density_lbl.txt

```
%SUMMARY Density - Computes  P and T.

%DESCRIPTION Parameter: density,ideal_gas,gas_constant
%DESCRIPTION OR       : specific_volume,ideal_gas,gas_constant
%DESCRIPTION OR       : density,incompressible,gas_constant
%DESCRIPTION OR       : specific_volume,incompressible,gas_constant
%DESCRIPTION Port [P]: Pressure
%DESCRIPTION Port [T]: Temperature
%DESCRIPTION Port [rho]: Density

%ALIAS P Pressure
%ALIAS T Temperature
%ALIAS rho|out Density

%ALIAS $1 density,ideal_gas,r



%% Label file for system Density (Density_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Density_lbl.txt,v 1.1 1998/07/04 09:10:26 peterg Exp $
% %% $Log: Density_lbl.txt,v $
% %% Revision 1.1  1998/07/04 09:10:26  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank
```

```
% Component type R
r Density density,ideal_gas,r

% Component type SS
[Pressure] SS external,external
[Temperature] SS external,external
[Density] SS external,external
```

Subsystems

No subsystems.

1.1.5 Df

Figure 1.4: System **Df**: acausal bond graph

The acausal bond graph of system **Df** is displayed in Figure 1.4 (on page 22) and its label file is listed in Section 1.1.5 (on page 22). The subsystems are listed in Section 1.1.5 (on page 24).

Summary information

System Df:Simple flow detector Simple flow detector constructed from SS with fixed causality

Interface information:

Parameter $1 represents actual parameter **external**

Port in represents actual port **port**

Port out represents actual port **port**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Df_lbl.txt

```
%% Label file for system Df (Df_lbl.txt)
%SUMMARY Df Simple flow detector
%DESCRIPTION Simple flow detector constructed from SS with fixed causali


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Df_lbl.txt,v 1.4 2002/11/07 04:28:29 gawthrop Exp $
% %% $Log: Df_lbl.txt,v $
% %% Revision 1.4  2002/11/07 04:28:29  gawthrop
% %% Now has argument - either internal or external
% %%
% %% Revision 1.3  1999/09/07 03:31:47  peterg
% %% Fixed alias bug
% %%
% %% Revision 1.2  1999/09/07 03:20:34  peterg
% %% Aliased to out as well as in
% %%
% %% Revision 1.1  1999/03/03 22:05:16  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% Port aliases
%ALIAS in|out port

% Argument aliases
%ALIAS $1 external
%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type SS
[port] SS external,external
y SS 0,external
```

Subsystems

No subsystems.

1.1.6 Fuel

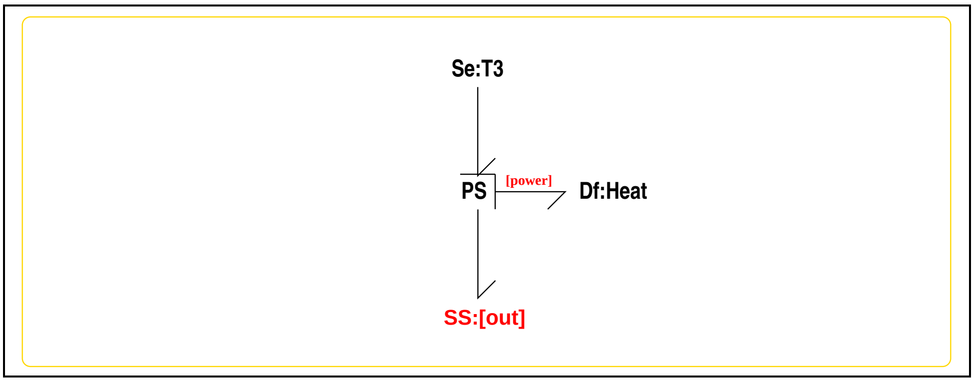


Figure 1.5: System **Fuel**: acausal bond graph

The acausal bond graph of system **Fuel** is displayed in Figure 1.5 (on page 24) and its label file is listed in Section 1.1.6 (on page 25). The subsystems are listed in Section 1.1.6 (on page 26).

Summary information

System Fuel:

Interface information:

Parameter $1 represents actual parameter **Heat**

Parameter $2 represents actual parameter **T3**

Port out represents actual port **out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Fuel_lbl.txt

```
%% Label file for system Fuel (Fuel_lbl.txt)
%SUMMARY Fuel
%DESCRIPTION

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Fuel_lbl.txt,v 1.1 2000/12/28 18:08:28 peterg Exp $
% %% $Log: Fuel_lbl.txt,v $
% %% Revision 1.1  2000/12/28 18:08:28  peterg
% %% To RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS out out

% Argument aliases
%ALIAS $1 Heat
```

```
%ALIAS $2 T3

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type Df
  Heat SS external

% Component type SS
[out] SS external,external

% Component type Se
T3 SS external
```

Subsystems

- Df Simple flow detector (1) No subsystems.
- Se Simple effort source (1) No subsystems.

1.1.7 In

The acausal bond graph of system **In** is displayed in Figure 6.3 (on page 133) and its label file is listed in Section 6.1.4 (on page 133). The subsystems are listed in Section 6.1.4 (on page 134).

Summary information

System In::Inflow conditions ¡Detailed description here¿

Interface information:

Port in represents actual port **Th_out,Hy_out**

Port out represents actual port **Th_out,Hy_out**

Variable declarations:

This component has no PAR declarations

Figure 1.6: System **In**: acausal bond graph

Units declarations:

This component has no UNITs declarations

The label file: In_lbl.txt

```
%SUMMARY In: Inflow conditions
%DESCRIPTION <Detailed description here>
%ALIAS in|out Th_out,Hy_out
%% Label file for system In (In_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: In_lbl.txt,v 1.2 2000/12/28 18:08:28 peterg Exp $
% %% $Log: In_lbl.txt,v $
% %% Revision 1.2  2000/12/28 18:08:28  peterg
% %% To RCS
% %%
% %% Revision 1.1  1998/07/04 09:41:53  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank



% Component type SS
[Hy_out] SS external,external
[Th_out] SS external,external
Hy_in SS p_1,internal
Th_in SS t_1,internal
```

Subsystems

No subsystems.

1.1.8 Load

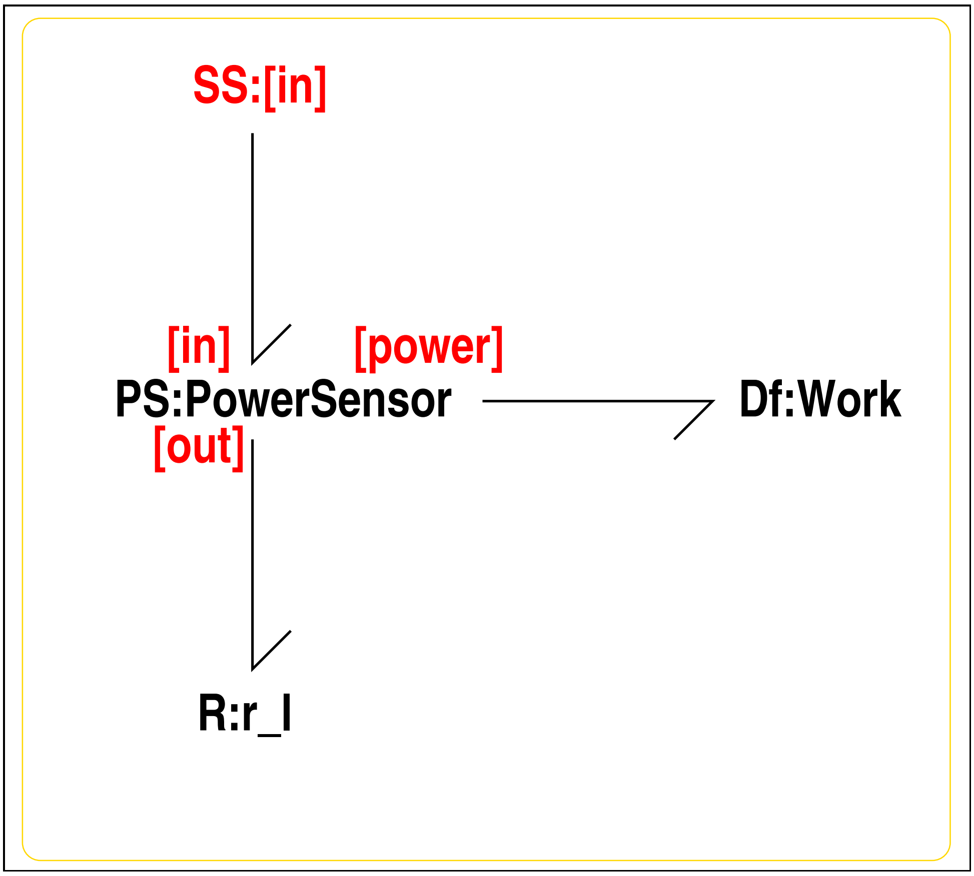


Figure 1.7: System **Load**: acausal bond graph

The acausal bond graph of system **Load** is displayed in Figure 1.7 (on page 29) and its label file is listed in Section 1.1.8 (on page 29). The subsystems are listed in Section 1.1.8 (on page 31).

Summary information

System Load:

Interface information:

Parameter $1 represents actual parameter **r_l**

Port in represents actual port **in**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Load_lbl.txt

```
%% Label file for system Load (Load_lbl.txt)
%SUMMARY Load
%DESCRIPTION

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Load_lbl.txt,v 1.1 2000/12/28 18:08:28 peterg Exp $
% %% $Log: Load_lbl.txt,v $
% %% Revision 1.1  2000/12/28 18:08:28  peterg
% %% To RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in in

% Argument aliases
%ALIAS $1 r_l

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----


PowerSensor

% Component type Df
```

```
Work SS external

% Component type R
r_l lin flow,r_l

% Component type SS
[in] SS external,external
```

Subsystems

- Df Simple flow detector (1) No subsystems.

1.1.9 Out

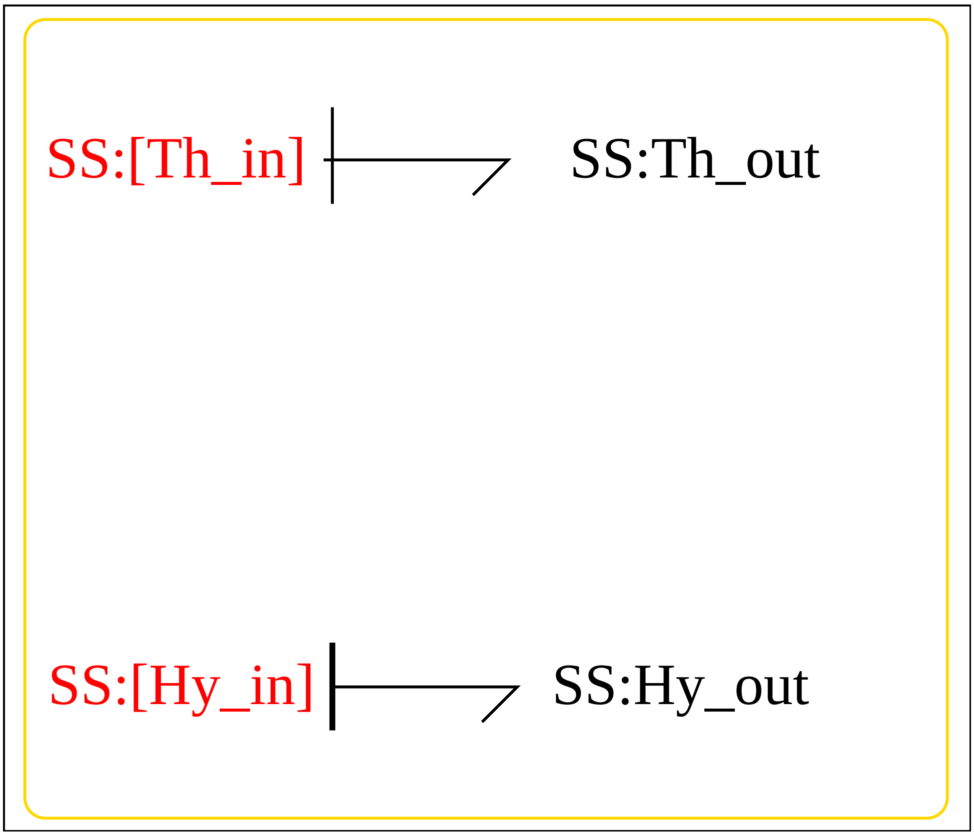


Figure 1.8: System **Out**: acausal bond graph

The acausal bond graph of system **Out** is displayed in Figure 6.4 (on page

135) and its label file is listed in Section 6.1.5 (on page 135). The subsystems are listed in Section 6.1.5 (on page 136).

Summary information

System Out::Outflow conditions ¡Detailed description here¿

Interface information:

Port in represents actual port **Th_in,Hy_in**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Out_lbl.txt

```
%SUMMARY Out: Outflow conditions
%DESCRIPTION <Detailed description here>
%ALIAS in  Th_in,Hy_in
%% Label file for system Out (Out_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Out_lbl.txt,v 1.1 1998/07/04 09:40:48 peterg Exp $
% %% $Log: Out_lbl.txt,v $
% %% Revision 1.1  1998/07/04 09:40:48  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank
```

```
% Component type SS
[Hy_in] SS external,external
[Th_in] SS external,external
Hy_out SS p_1,internal
Th_out SS t_1,internal
```

Subsystems

No subsystems.

1.1.10 Poly

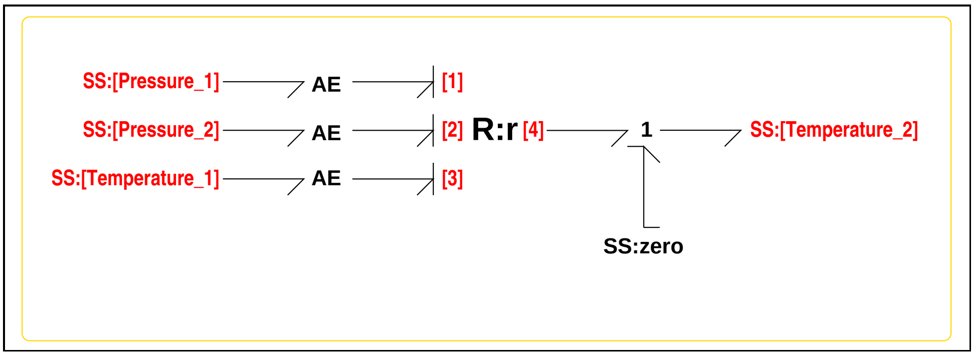


Figure 1.9: System **Poly**: acausal bond graph

The acausal bond graph of system **Poly** is displayed in Figure 1.9 (on page 33) and its label file is listed in Section 1.1.10 (on page 34).

This four-port component computes the temperature following a polytropic expansion using:

$$T_2 = T_1 \left(\frac{P_2}{P_1}\right)^{\alpha} \tag{1.2}$$

where $\alpha = \frac{n-1}{n}$ and n is the coefficient of polytropic expansion. This component imposes zero flow at all its ports and therefore does not affect energy balance.

The output is *bicausal* as it imposes both T_2 and a zero flow. This is implemented using the bicausal **SS** component labeled “zero”.

Summary information

System Poly:- computes polytropic expansion temperature. Parameter: alpha = (n-1)/n (n = polytropic index) This four-port component computes the temperature following a polytropic expansion using where alpha = (n-1)/n and n is the coefficient of polytropic expansion. This component imposes zero flow at all its ports and therefore does not affect energy balance. The output is bicausal as it imposes both T_2 and a zero flow. This is implemented using the bicausal SS component labeled "zero".

Interface information:

Parameter $1 represents actual parameter **alpha**

Port P1 represents actual port **Pressure_1**

Port P2 represents actual port **Pressure_2**

Port T1 represents actual port **Temperature_1**

Port T2 represents actual port **Temperature_2**

Port out represents actual port **Temperature_2**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Poly_lbl.txt

```
%SUMMARY Poly - computes polytropic expansion temperature.
%DESCRIPTION Parameter: alpha = (n-1)/n (n = polytropic index)

%DESCRIPTION This four-port component computes the temperature fol
%DESCRIPTION a polytropic expansion using where alpha = (n-1)/n an
%DESCRIPTION the coefficient of polytropic expansion.  This compon
%DESCRIPTION imposes zero flow at all its ports and therefore does
%DESCRIPTION affect energy balance.  The output is bicausal as it 
%DESCRIPTION imposes both T_2 and a zero flow.  This is implemente
```

```
%DESCRIPTION the bicausal SS component labeled ‘‘zero’’.

%ALIAS P1 Pressure_1
%ALIAS P2 Pressure_2
%ALIAS T1 Temperature_1
%ALIAS T2|out Temperature_2

%ALIAS $1 alpha

%% Label file for system Poly (Poly_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Poly_lbl.txt,v 1.2 1998/07/04 09:31:26 peterg Exp $
% %% $Log: Poly_lbl.txt,v $
% %% Revision 1.2  1998/07/04 09:31:26  peterg
% %% New-style + documentation
% %%
% %% Revision 1.1  1998/03/27 10:48:50  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank



% Component type R
r Poly alpha

% Component type SS
zero SS 0,0
[Pressure_1] SS external,external
[Temperature_1] SS external,external
[Pressure_2] SS external,external
[Temperature_2] SS external,external
```

Subsystems

No subsystems.

1.1.11 Pump

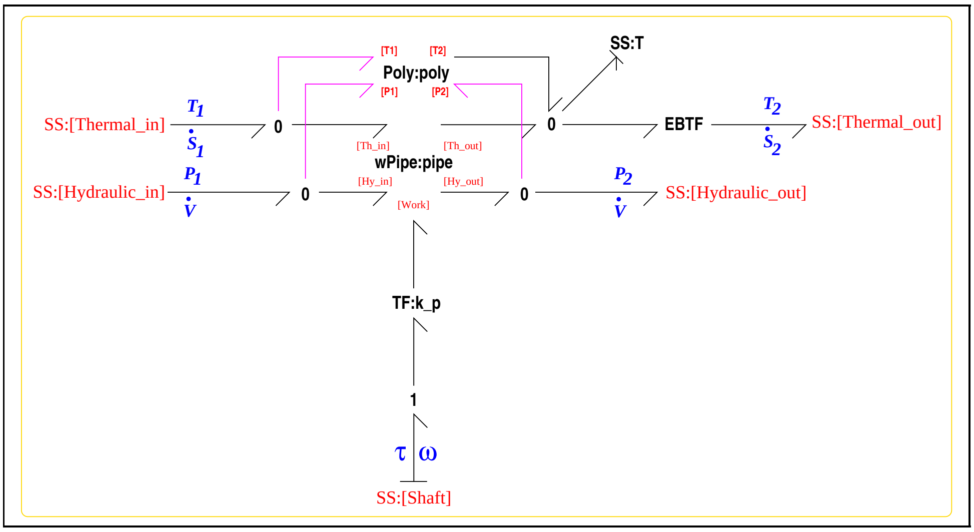


Figure 1.10: System **Pump**: acausal bond graph

The acausal bond graph of system **Pump** is displayed in Figure 5.5 (on page 110) and its label file is listed in Section 5.1.6 (on page 110). The subsystems are listed in Section 5.1.6 (on page 112).

Pump represents an ideal pumping component for compressible or incompressible flow though a pipe, it may also be used as a turbine.

The pump is ideal in the sense that the mass flow rate $\dot{m}$ depends only on the shaft speed ω:

$$\dot{m} = k_p \omega \tag{1.3}$$

It is implemented using three components:

- the ideal isentropic **wPipe** component which gives the correct energy flows
- the polytropic expansion **Poly** component which imposes the correct temperature at the output of the pump. This component imposes zero flow at all its ports and therefore does not affect energy balance. It has a bicausal output imposing both the temperature measured by the **SS** component “T” and a zero flow.

- the *effort-bicausal transformer* **EBTF** component. This component is an energy-conserving **TF** component with non-standard causality. The modulus is determined by the two imposed efforts (T and T_2), and this modulus determines the flows in the usual way. In particular, it makes sure that the internal energy flowing from the pump to the following components (imposing T_2) is correct.

Summary information

System Pump:Ideal pump component for compressible flow Pump represents an ideal pumping component for compressible or incompressible flow though a pipe, it may also be used as a turbine. The pump is ideal in the sense that the mass flow rate depends only on the shaft speed. Parameter 1: c_v - specific heat of fluid Parameter 2: Parameter passed to Density component Parameter 3: alpha = (n-1)/n, n coefficient of polytropic expansion. Parameter 4: k_p pump constant: mass flow = k_p*shaft speed

Interface information:

Component Poly is in library **CompressibleFlow/Poly**

Component wPipe is in library **CompressibleFlow/wPipe**

Parameter $1 represents actual parameter **c_v**

Parameter $2 represents actual parameter **density,ideal_gas,r**

Parameter $3 represents actual parameter **alpha**

Parameter $4 represents actual parameter **flow,k_p**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port Work represents actual port **Shaft**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Pump_lbl.txt

```
%SUMMARY Pump Ideal pump component for compressible flow

%DESCRIPTION Pump represents an ideal pumping component for compre
%DESCRIPTION or incompressible flow though a pipe, it may also be
%DESCRIPTION The pump is ideal in the sense that the mass flow rat
%DESCRIPTION depends only on the shaft speed.

%DESCRIPTION Parameter 1: c_v - specific heat of fluid
%DESCRIPTION Parameter 2: Parameter passed to Density component
%DESCRIPTION Parameter 3: alpha = (n-1)/n, n coefficient of polytr
%DESCRIPTION expansion.
%DESCRIPTION Parameter 4: k_p pump constant: mass flow = k_p*shaft

%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out

%ALIAS Work Shaft

%ALIAS  $1 c_v
%ALIAS  $2 density,ideal_gas,r
%ALIAS  $3 alpha
%ALIAS  $4 flow,k_p

%ALIAS wPipe CompressibleFlow/wPipe
%ALIAS Poly CompressibleFlow/Poly
```

```
%% Label file for system Pump (Pump_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Pump_lbl.txt,v 1.8 1998/07/17 16:46:26 peterg Exp $
% %% $Log: Pump_lbl.txt,v $
% %% Revision 1.8  1998/07/17 16:46:26  peterg
% %% Added aliases for Poly and wPipe
% %%
% %% Revision 1.7  1998/07/04 08:39:58  peterg
% %% New-style SS
% %%
% %% Revision 1.6  1998/07/03 15:02:25  peterg
% %% Work alias added
% %%
% %% Revision 1.5  1998/07/03 14:43:24  peterg
% %% Added parameter aliases
% %%
% %% Revision 1.4  1998/07/02 19:46:19  peterg
% %% New aliases
% %%
% %% Revision 1.3  1998/07/02 10:55:54  peterg
% %% Lower case in out
% %%
% %% Revision 1.2  1998/07/02 10:52:33  peterg
% %% Added port aliases
% %%
% %% Revision 1.1  1998/04/07 15:23:30  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank



% Component type wPipe
```

```
pipe none c_v;density,ideal_gas,r

% Component type Poly
poly Poly alpha

% Component type SS
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Shaft] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
T SS external,0

% Component type TF
k_p lin flow,k_p
```

Subsystems

- Poly - computes polytropic expansion temperature. (1) No subsystems.
- wPipe Isentropic pipe with work transfer. (1)
  - Density - Computes P and T. (2)

1.1.12 Se

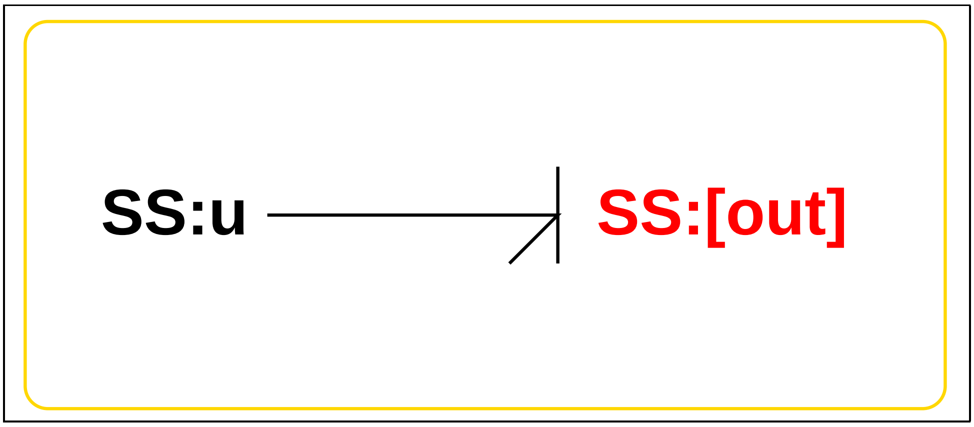


Figure 1.11: System **Se**: acausal bond graph

The acausal bond graph of system **Se** is displayed in Figure 8.6 (on page 199) and its label file is listed in Section 8.1.7 (on page 200). The subsystems are listed in Section 8.1.7 (on page 201).

Summary information

System Se:Simple effort source Simple effort source constructed from SS with fixed causality

Interface information:

Parameter $1 represents actual parameter **e_s**

Port in represents actual port **out**

Port out represents actual port **out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Se_lbl.txt

```
%% Label file for system Se (Se_lbl.txt)
%SUMMARY Se Simple effort source
%DESCRIPTION Simple effort source constructed from SS with fixed causali


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Se_lbl.txt,v 1.3 1999/08/05 07:31:39 peterg Exp $
% %% $Log: Se_lbl.txt,v $
% %% Revision 1.3  1999/08/05 07:31:39  peterg
% %% Added in alias
% %%
% %% Revision 1.2  1999/03/12 04:04:27  peterg
% %% Single argument - the effort value e_s
```

```
% %%
% %% Revision 1.1  1999/03/03 21:55:46  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS out|in out

% Argument aliases
%ALIAS $1 e_s

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type SS
[out] SS external,external
u SS e_s,internal
```

Subsystems

No subsystems.

1.1.13 Shaft

The acausal bond graph of system **Shaft** is displayed in Figure 1.12 (on page 43) and its label file is listed in Section 1.1.13 (on page 42). The subsystems are listed in Section 1.1.13 (on page 45).

Summary information

System Shaft:

Interface information:

Parameter $1 represents actual parameter **j_s**

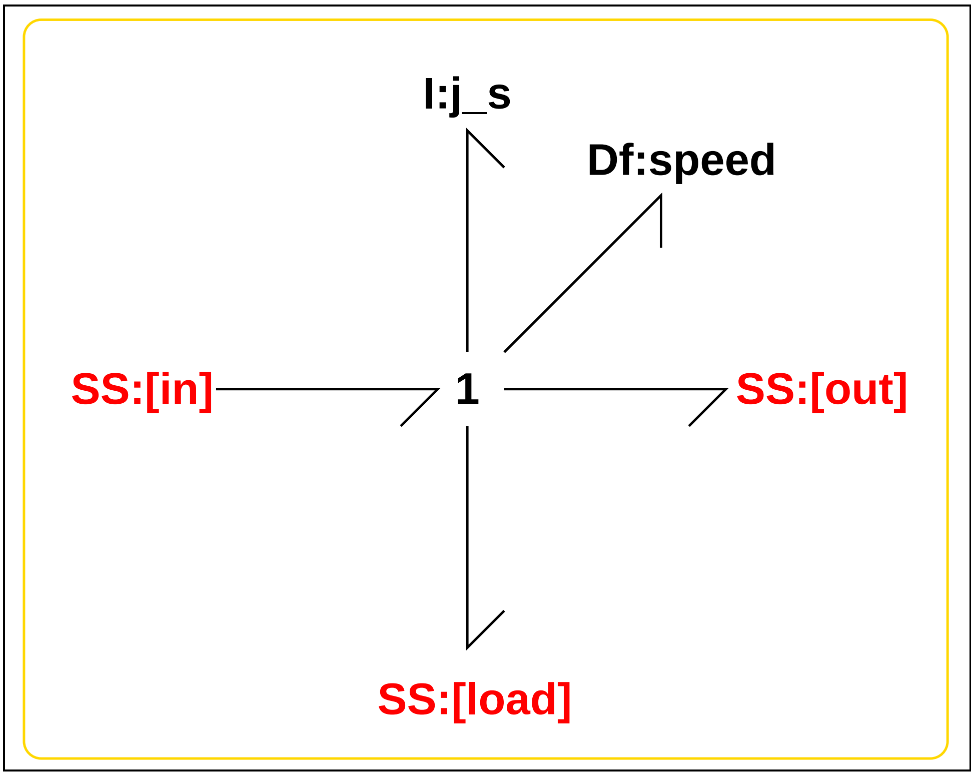


Figure 1.12: System **Shaft**: acausal bond graph

Port in represents actual port **in**

Port load represents actual port **load**

Port out represents actual port **out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Shaft_lbl.txt

```
%% Label file for system Shaft (Shaft_lbl.txt)
%SUMMARY Shaft
%DESCRIPTION

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Shaft_lbl.txt,v 1.1 2000/12/28 18:08:28 peterg Exp $
% %% $Log: Shaft_lbl.txt,v $
% %% Revision 1.1  2000/12/28 18:08:28  peterg
% %% To RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in in
%ALIAS load load
%ALIAS out out

% Argument aliases
%ALIAS $1 j_s

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
```

```
%       blank

% ---- Component labels ----
speed SS external

% Component type I
j_s lin flow,j_s

% Component type SS
[in] SS external,external
[load] SS external,external
[out] SS external,external
```

Subsystems

- Df Simple flow detector (1) No subsystems.

1.1.14 hPipe

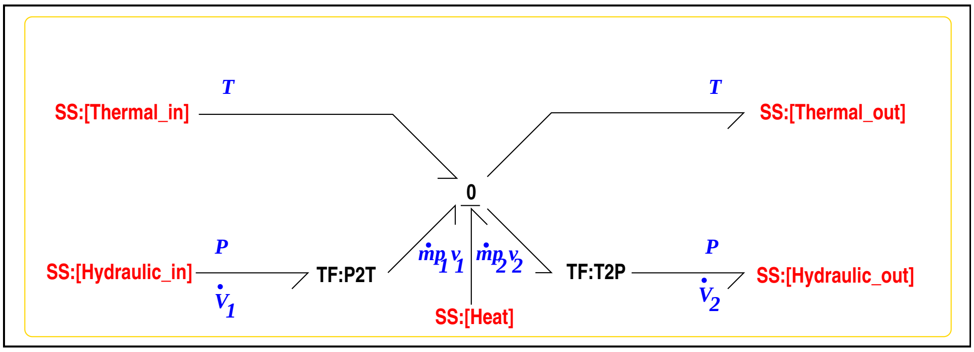


Figure 1.13: System **hPipe**: acausal bond graph

The acausal bond graph of system **hPipe** is displayed in Figure 1.13 (on page 45) and its label file is listed in Section 1.1.14 (on page 46). The subsystems are listed in Section 1.1.14 (on page 48).

hPipe represents an ideal (energy conserving) pipe carrying a fluid with heat transfer. To ensure energy conservation, power bonds are used and connected by (energy conserving) **TF** components. It is assumed that the working fluid is an ideal gas (gas constant r) and that a mass m_t is stored within pipe with a volume v_t.

The central **0** junction carries temperature (T) and the two hydraulic ports are connected to this by appropriate transformers. The modulus of the **TF** component labeled “P2T” is such P and T are related by the ideal gas law

$$P = \frac{Rm_t}{v_t} T \tag{1.4}$$

Summary information

System hPipe::Pipe for compressible fluid with heat transfer and heat storage. hPipe represents an ideal (energy conserving) pipe carrying a fluid with heat transfer. To ensure energy conservation, power bonds are used and connected by (energy conserving) TF components. It is assumed that the working fluid is an ideal gas (gas constant r) and that a mass m_t is stored within pipe with a volume v_t. Parameter 1: m_t (mass in pipe) Parameter 2: v_t (volume of pipe) Parameter 3: r (gas constant) Typical lable entry pipe none m_p;v_p;r

Interface information:

Parameter $1 represents actual parameter **m**

Parameter $2 represents actual parameter **v**

Parameter $3 represents actual parameter **r**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: hPipe_lbl.txt

```
%SUMMARY hPipe: Pipe for compressible fluid with heat transfer and heat

%DESCRIPTION hPipe represents an ideal (energy conserving) pipe carrying
%DESCRIPTION fluid with heat transfer. To ensure energy conservation, po
%DESCRIPTION connected by (energy conserving) TF components.
%DESCRIPTION It is assumed that the working fluid is an ideal gas (gas c
%DESCRIPTION m_t is stored within pipe with a volume v_t.

%DESCRIPTION Parameter 1: m_t (mass in pipe)
%DESCRIPTION Parameter 2: v_t (volume of pipe)
%DESCRIPTION Parameter 3: r (gas constant)
%DESCRIPTION Typical lable entry
%DESCRIPTION % Component type hPipe
%DESCRIPTION  pipe none m_p;v_p;r

%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out

%ALIAS $1 m
%ALIAS $2 v
%ALIAS $3 r



%% Label file for system hPipe (hPipe_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: hPipe_lbl.txt,v 1.2 1998/07/04 08:45:05 peterg Exp $
% %% $Log: hPipe_lbl.txt,v $
% %% Revision 1.2  1998/07/04 08:45:05  peterg
% %% New-style SS labels
```

```
% %%
% %% Revision 1.1  1998/07/03 17:38:32  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank



% Component type SS
[Heat] SS external,external
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external

% Component type TF
P2T lin flow,r*m/v
T2P lin effort,r*m/v
```

Subsystems

No subsystems.

1.1.15 wPipe

The acausal bond graph of system **wPipe** is displayed in Figure 1.14 (on page 49) and its label file is listed in Section 1.1.15 (on page 49). The subsystems are listed in Section 1.1.15 (on page 52).

wPipe represents an ideal (energy conserving) pipe carrying a fluid with work transfer. To ensure energy conservation, power bonds are used and connected by (energy conserving) **TF** components.

The central **1** junction carries mass flow ($\dot{m}$) and the four ports are connected to this by appropriate transformers. In the case of the hydraulic ports, these transformers are *modulated* by the corresponding fluid density. The bonds impinging on this **1** junction carry the corresponding effort variables; in particular, the thermal bonds carry specific internal energy u and the hydraulic bonds carry Pv where

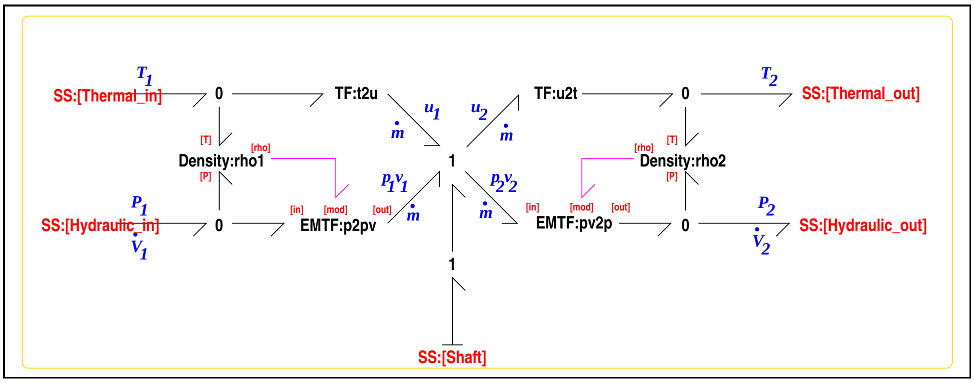


Figure 1.14: System **wPipe**: acausal bond graph

P is the pressure and *v* the specific volume.

The ports "Work_in" and "Work_out" are convenient for attaching (for example) the shadt work of a pump, turbine or compressor.

The ports are

Hy_in Pressure/volume-flow inflow

Hy_in Pressure/volume-flow outflow

Th_in Temperature/Entropy-flow in flow

Th_out Temperature/Entropy-flow out flow

Shaft Torque/angular velocity input.

Summary information

System wPipe:Isentropic pipe with work transfer. wPipe represents an ideal (energy conserving) pipe carrying a fluid with work transfer. To ensure energy conservation, power bonds are used and connected by (energy conserving) TF components. Parameter 1: c_v - specific heat of fluid Parameter 2: Parameter passed to Density component Ports: [Hy_in] Pressure/volume-flow inflow [Hy_in] Pressure/volume-flow outflow [Th_in] Temperature/Entropy-flow in flow [Th_out] Temperature/Entropy-flow out flow [Shaft] Torque/angular velocity input.

Interface information:

Component Density is in library **CompressibleFlow/Density**

Parameter $1 represents actual parameter **c_v**

Parameter $2 represents actual parameter **density,ideal_gas,r**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port Work represents actual port **Shaft**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: wPipe_lbl.txt

```
%SUMMARY wPipe Isentropic pipe with work transfer.

%DESCRIPTION wPipe represents an ideal (energy conserving) pipe ca
%DESCRIPTION fluid with work transfer. To ensure energy conservati
%DESCRIPTION connected by (energy conserving) TF components.

%DESCRIPTION Parameter 1: c_v - specific heat of fluid
%DESCRIPTION Parameter 2: Parameter passed to Density component
%DESCRIPTION Ports:

%DESCRIPTION [Hy_in] Pressure/volume-flow inflow
%DESCRIPTION [Hy_in] Pressure/volume-flow outflow
%DESCRIPTION [Th_in] Temperature/Entropy-flow in flow
%DESCRIPTION [Th_out] Temperature/Entropy-flow out flow
%DESCRIPTION [Shaft] Torque/angular velocity input.

%ALIAS in  Thermal_in,Hydraulic_in
```

```
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out

%ALIAS Work Shaft

%ALIAS  $1 c_v
%ALIAS  $2 density,ideal_gas,r

%ALIAS Density CompressibleFlow/Density

%% Label file for system wPipe (wPipe_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: wPipe_lbl.txt,v 1.3 1998/07/17 16:53:43 peterg Exp $
% %% $Log: wPipe_lbl.txt,v $
% %% Revision 1.3  1998/07/17 16:53:43  peterg
% %% Added density alias
% %%
% %% Revision 1.2  1998/07/04 08:33:30  peterg
% %% New-style SS
% %%
% %% Revision 1.1  1998/07/03 17:38:20  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank



% Component type Density
```

```
rho1 none density,ideal_gas,r
rho2 none density,ideal_gas,r

% Component type EMTF
p2pv lin flow
pv2p lin effort

% Component type SS
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Shaft] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external

% Component type TF
t2u lin effort,c_v
u2t lin flow,c_v
```

Subsystems

- Density - Computes P and T. (2) No subsystems.

1.2 SimpleGasTurbine_struc.tex *(-o -ss)*

MTT command:

```
mtt -o -ss SimpleGasTurbine struc tex
```

List of inputs for system SimpleGasTurbine			
	Component	System	Repetition
1	u	SimpleGasTurbine__fuel__T3__u	1

List of outputs for system SimpleGasTurbine

	Component	System	Repetition
1	y	SimpleGasTurbine__fuel__Heat__y	1
2	T	SimpleGasTurbine__comp__T	1
3	P	SimpleGasTurbine__c1__P	1
4	T	SimpleGasTurbine__c1__T	1
5	T	SimpleGasTurbine__turb__T	1

List of outputs for system SimpleGasTurbine (continued)			
	Component	System	Repetition
6	y	SimpleGasTurbine__shaft__speed__y	1
7	y	SimpleGasTurbine__load__Work__y	1

List of states for system SimpleGasTurbine			
	Component	System	Repetition
1	j_s	SimpleGasTurbine__shaft__j_s	1

1.3 SimpleGasTurbine_sympar.tex *(-o -ss)*

MTT command:

```
mtt -o -ss SimpleGasTurbine sympar tex
```

1.4 SimpleGasTurbine_ode.tex *(-o -ss)*

MTT command:

```
mtt -o -ss SimpleGasTurbine ode tex
```

$$\dot{x}_1 = \frac{\left(-\left(\frac{(m_c u_1 r)}{(p_1 v_c)}\right)^{\alpha} c_p j_s t_1 - \left(\frac{(p_1 v_c)}{(m_c u_1 r)}\right)^{\alpha} c_p j_s u_1 + c_p j_s u_1 + c_p j_s t_1 - k x_1 r_l\right)}{(j_s k)} \tag{1.5}$$

Parameter	System
alpha	SimpleGasTurbine
c_p	SimpleGasTurbine
c_v	SimpleGasTurbine
gamma_0	SimpleGasTurbine
j_s	SimpleGasTurbine
k	SimpleGasTurbine
m_c	SimpleGasTurbine
mdot	SimpleGasTurbine
mom_0	SimpleGasTurbine
omega_0	SimpleGasTurbine
p_1	SimpleGasTurbine__out,
p_2	SimpleGasTurbine
p_3	SimpleGasTurbine
p_4	SimpleGasTurbine
r	SimpleGasTurbine
r_l	SimpleGasTurbine
r_p	SimpleGasTurbine
t_1	SimpleGasTurbine__out,
t_2	SimpleGasTurbine
t_3	SimpleGasTurbine
t_4	SimpleGasTurbine
v_c	SimpleGasTurbine
w_0	SimpleGasTurbine

Table 1.1: Parameters

$$
\begin{aligned}
y_1 &= \frac{\left(c_p x_1 \left(-\left(\frac{(m_c u_1 r)}{(p_1 v_c)}\right)^{\alpha} t_1 + u_1\right)\right)}{(j_s k)} \\
y_2 &= \left(\frac{(m_c u_1 r)}{(p_1 v_c)}\right)^{\alpha} t_1 \\
y_3 &= \frac{(m_c u_1 r)}{v_c} \\
y_4 &= u_1 \\
y_5 &= \left(\frac{(p_1 v_c)}{(m_c u_1 r)}\right)^{\alpha} u_1 \\
y_6 &= \frac{x_1}{j_s} \\
y_7 &= \frac{(x_1^2 r_l)}{j_s^2}
\end{aligned}
\tag{1.6}
$$

1.5 SimpleGasTurbine_sspar.r *(-o -ss)*

MTT command:

```
mtt -o -ss SimpleGasTurbine sspar r
```

1.6 SimpleGasTurbine_ss.tex *(-o -ss)*

MTT command:

```
mtt -o -ss SimpleGasTurbine ss tex
```

$$
x = \left(c_p k \left(-\left(\frac{p_1}{p_3}\right)^{\alpha} t_3 - \left(\frac{p_3}{p_1}\right)^{\alpha} t_1 + t_1 + t_3\right)\right) \tag{1.7}
$$

$$
u = (t_3) \tag{1.8}
$$

$$y = \begin{pmatrix} \frac{\left(c_p\left(-\left(\frac{p_3}{p_1}\right)^{\alpha} t_1 + t_3\right)\right)}{k^2} \\ \left(\frac{p_3}{p_1}\right)^{\alpha} t_1 \\ p_3 \\ t_3 \\ \left(\frac{p_1}{p_3}\right)^{\alpha} t_3 \\ \frac{1}{k} \\ c_p\left(-\left(\frac{p_1}{p_3}\right)^{\alpha} t_3 - \left(\frac{p_3}{p_1}\right)^{\alpha} t_1 + t_1 + t_3\right) \end{pmatrix} \tag{1.9}$$

$$\dot{x} = \left(\frac{\left(c_p\left(\left(\frac{p_1}{p_3}\right)^{\alpha} k^2 t_3 - \left(\frac{p_1}{p_3}\right)^{\alpha} t_3 + \left(\frac{p_3}{p_1}\right)^{\alpha} k^2 t_1 - \left(\frac{p_3}{p_1}\right)^{\alpha} t_1 - k^2 t_1 - k^2 t_3 + t_1 + t_3\right)\right)}{k} \right) \tag{1.10}$$

1.7 SimpleGasTurbine_sm.tex *(-o -ss)*

MTT command:

```
mtt -o -ss SimpleGasTurbine sm tex
```

$$A = \left(-1\right) \tag{1.11}$$

$$B = \left(\frac{\left(c_p\left(\left(\frac{p_1}{p_3}\right)^{\alpha} \alpha t_3 - \left(\frac{p_1}{p_3}\right)^{\alpha} t_3 - \left(\frac{p_3}{p_1}\right)^{\alpha} \alpha t_1 + t_3\right)\right)}{(k t_3)} \right) \tag{1.12}$$

$$C = \begin{pmatrix} \frac{\left(\left(\frac{p_3}{p_1}\right)^{\alpha} t_1 - t_3\right)}{\left(k^3\left(\left(\frac{p_1}{p_3}\right)^{\alpha} t_3 + \left(\frac{p_3}{p_1}\right)^{\alpha} t_1 - t_1 - t_3\right)\right)} \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{(-1)}{\left(c_p k^2\left(\left(\frac{p_1}{p_3}\right)^{\alpha} t_3 + \left(\frac{p_3}{p_1}\right)^{\alpha} t_1 - t_1 - t_3\right)\right)} \\ \frac{2}{k} \end{pmatrix} \tag{1.13}$$

$$D = \left(\frac{\left(c_p\left(-\left(\frac{p_3}{p_1}\right)^{\alpha} \alpha t_1 + t_3\right)\right)}{\left(k^2 t_3\right)} \right) \tag{1.14}$$

1.8 SimpleGasTurbine_numpar.tex *(-o -ss)*

MTT command:

```
mtt -o -ss SimpleGasTurbine numpar tex
```

```
# Numerical parameter file (SimpleGasTurbine_numpar.txt)
# Generated by MTT at Tue Mar 31 12:15:00 BST 1998

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: SimpleGasTurbine_numpar.txt,v 1.1 2000/12/28 18:08:28 pe
# %% $Log: SimpleGasTurbine_numpar.txt,v $
# %% Revision 1.1  2000/12/28 18:08:28  peterg
# %% To RCS
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Parameters
c_p =  1005.0;
c_v =  718.0;
gamma_0 =  c_p/c_v;
alpha = (gamma_0-1)/gamma_0;
k =  1.0;
p_1 =  1e5; # 1 bar
p_4 =  p_1;
r =  c_p-c_v;
t_1 =  288.0; # In
v_c =  1.0;

%Set the CC pressure and temperature
t_3 = 1000.0;
r_p = 6.0;
p_3 = r_p*p_1;

%Find stored mass to give combustion chamber pressure p_3 (at
% temperature t_3
m_c = (p_3*v_c)/(t_3*r);

%Equate pressures
p_4 = p_1;
```

```
p_2 = p_3;

%Compute ss temperatures (isentropic)
t_2 = t_1*(p_2/p_1)^alpha;
t_4 = t_3*(p_4/p_3)^alpha;

%Find the steady-state work output
w_0 = c_p*(t_3-t_4) - c_p*(t_2-t_1);

%Unit mass flow
mdot = 1;

%Corresponding shaft speed
omega_0 = mdot/k;

%Compute the corresponding load resistance (to absorb that work)
r_l = w_0/(omega_0)^2;

%Compute shaft inertia to give unit time constant (j_s*r_l)
j_s = r_l;

%Find angular momentum to give shaft speed omega_0
mom_0 =  omega_0*j_s;
```

1.9 SimpleGasTurbine_input.tex *(-o -ss)*

MTT command:

```
mtt -o -ss SimpleGasTurbine input tex
```

```
# Numerical parameter file (SimpleGasTurbine_input.txt)
# Generated by MTT at Tue Mar 31 12:38:39 BST 1998

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: SimpleGasTurbine_input.txt,v 1.2 2003/06/11 16:10:13 gawthrop
# %% $Log: SimpleGasTurbine_input.txt,v $
# %% Revision 1.2  2003/06/11 16:10:13  gawthrop
# %% Updated examples for latest MTT.
# %%
```

```
# %% Revision 1.1  2000/12/28 18:08:28  peterg
# %% To RCS
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

t_3 = 1000;

# Set the inputs
## Removed by MTT on Mon Nov 27 15:18:54 GMT 2000: u(1) =
t_3 + 0.1*t_3*(t>1) -  0.2*t_3*(t>5); # SimpleGasTurbine (T3)
## Removed by MTT on Wed Jun 11 15:42:11 BST 2003: simplegasturbin
= t_3 + 0.1*t_3*(t>1) -  0.2*t_3*(t>5);

simplegasturbine__fuel__t3__u = t_3 + 0.1*t_3*(t>1) -  0.2*t_3*(t>
```

1.10 SimpleGasTurbine_state.tex *(-o -ss)*

MTT command:

```
mtt -o -ss SimpleGasTurbine state tex
```

```
# State initialisation file (SimpleGasTurbine_state.txt)
# Generated by MTT at Tue Mar 31 12:37:17 BST 1998

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: SimpleGasTurbine_state.txt,v 1.2 2003/06/11 16:11:15 gaw
# %% $Log: SimpleGasTurbine_state.txt,v $
# %% Revision 1.2  2003/06/11 16:11:15  gawthrop
# %% Updated examples for latest MTT.
# %%
# %% Revision 1.1  2000/12/28 18:08:28  peterg
# %% To RCS
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


# Set the states
## Removed by MTT on Mon Nov 27 15:20:21 GMT 2000: x(1) =
j_s/k ; # SimpleGasTurbine (j_s)
```

```
## Removed by MTT on Wed Jun 11 15:41:26 BST 2003: simplegasturbine_shaf
= j_s/k ;
simplegasturbine__shaft__j_s = j_s/k;
```

1.11 SimpleGasTurbine_simpar.tex *(-o -ss)*

MTT command:

```
mtt -o -ss SimpleGasTurbine simpar tex
```

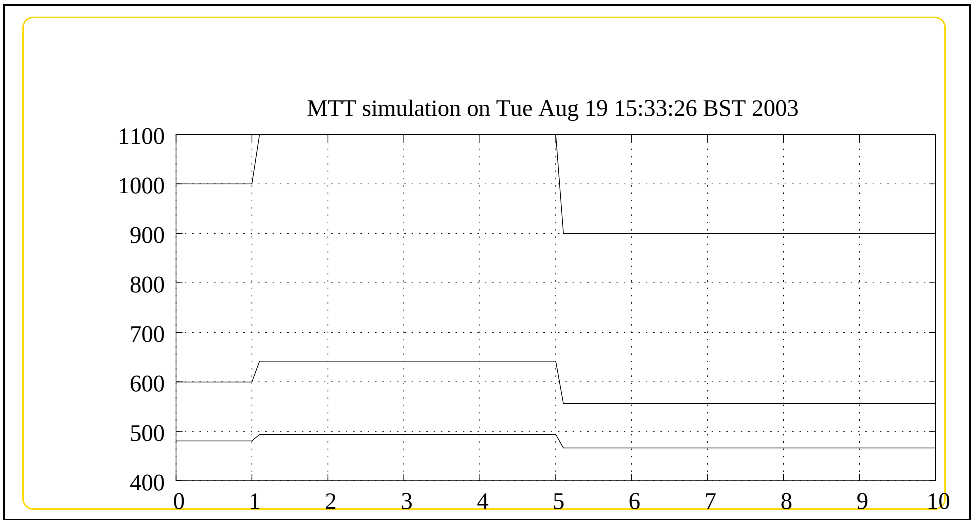
```
# -*-octave-*- Put Emacs into octave-mode
# Simulation parameters for system SimpleGasTurbine (SimpleGasTurbine_si
# Generated by MTT on Tue Aug 19 15:33:20 BST 2003.
###############################################################
## Version control history
###############################################################
## $Id: rcs_header.sh,v 1.1 2000/12/28 11:58:07 peterg Exp $
## $Log: rcs_header.sh,v $
## Revision 1.1  2000/12/28 11:58:07  peterg
## Put under RCS
##
###############################################################


FIRST        = 0.0;       # First time in simulation output
DT          = 0.1;       # Print interval
LAST        = 10.0;      # Last time in simulation
STEPFACTOR  = 1;         # Integration steps per print interval
WMIN        = -1;        # Minimum frequency = 10^WMIN
WMAX        = 2;         # Maximum frequency = 10^WMAX
WSTEPS      = 100;       # Number of frequency steps
INPUT       = 1;         # Index of the input
```

1.12 SimpleGasTurbine_odeso.ps *(-o -ss - SimpleGasTurbine__comp__T,SimpleGasTurbine__c1__T,SimpleGa*

MTT command:

```
mtt -o -ss SimpleGasTurbine odeso ps 'SimpleGasTurbine__comp__T,SimpleGa
```

This representation is given as Figure 1.15 (on page 62).

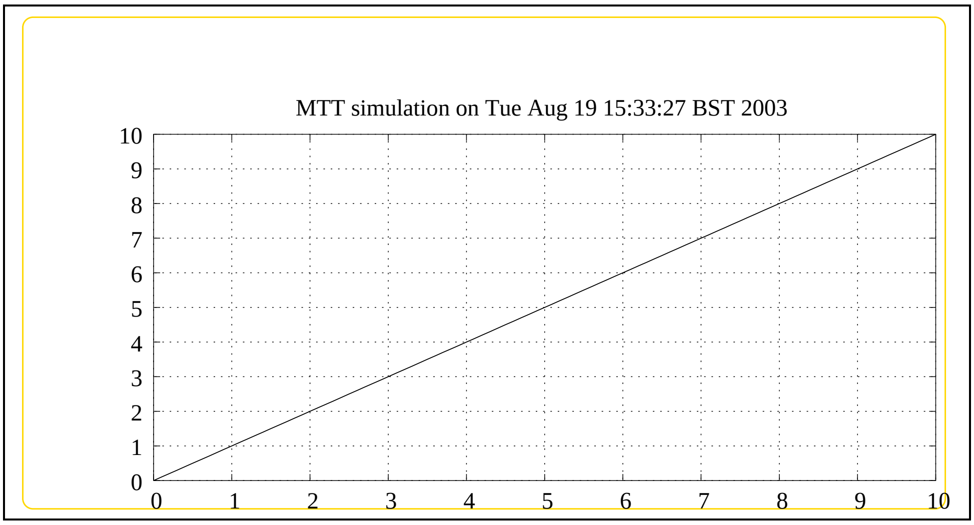


1.13 SimpleGasTurbine_odeso.ps *(-o -ss -SimpleGasTurbine_fuel_1_Heat_1_y,SimpleGasTurbine_load*

MTT command:

```
mtt -o -ss SimpleGasTurbine odeso ps 'SimpleGasTurbine_fuel_1_Heat
```

This representation is given as Figure 1.16 (on page 62).

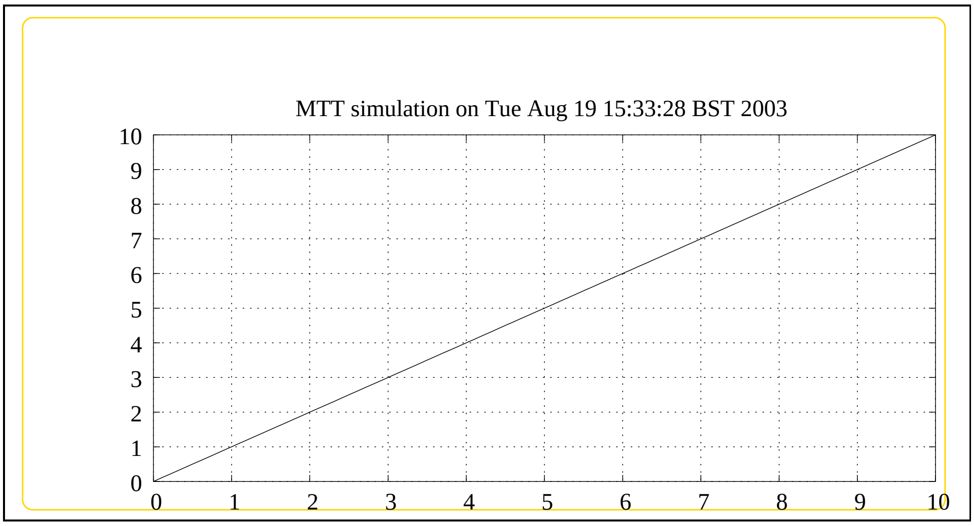


1.14 SimpleGasTurbine_odeso.ps *(-o -ss -SimpleGasTurbine_shaft_1_speed_1_y)*

MTT command:

```
mtt -o -ss SimpleGasTurbine odeso ps 'SimpleGasTurbine_shaft_1_speed_1_y
```

This representation is given as Figure 1.17 (on page 63).

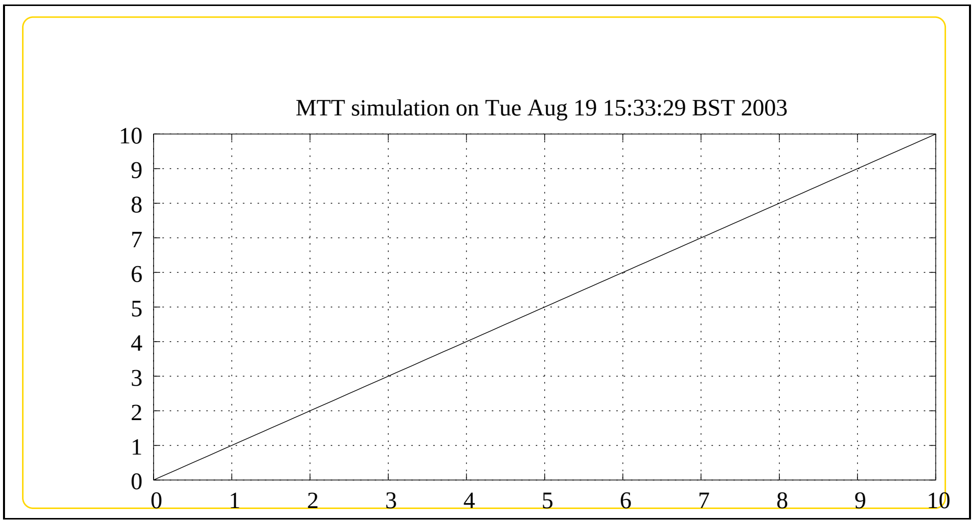


1.15 SimpleGasTurbine_odeso.ps *(-o -ss -SimpleGasTurbine_c1_1_P)*

MTT command:

```
mtt -o -ss SimpleGasTurbine odeso ps 'SimpleGasTurbine_c1_1_P'
```

This representation is given as Figure 1.18 (on page 64).


MTT simulation on Tue Aug 19 15:33:29 BST 2003
0
1
2
3
4
5
6
7
8
9
10
0
1
2
3
4
5
6
7
8
9
10

Part II

Incompressible-Components

Chapter 2

TestPipe

2.1 TestPipe_abg.tex

MTT command:

```
mtt TestPipe abg tex
```

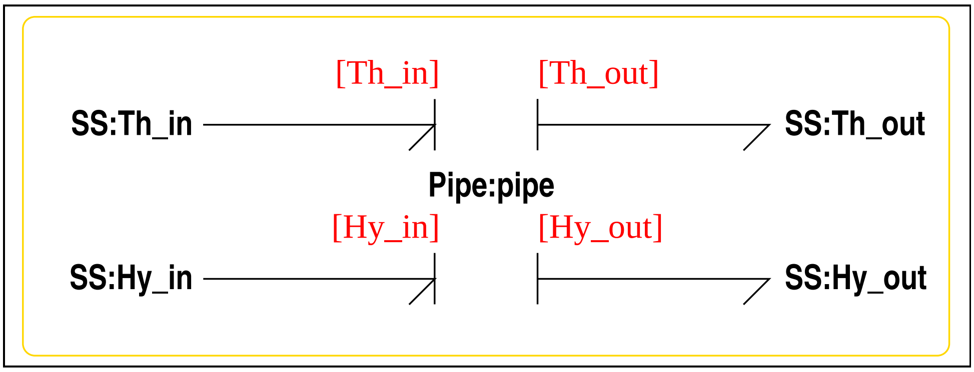


Figure 2.1: System **TestPipe**: acausal bond graph

The acausal bond graph of system **TestPipe** is displayed in Figure 2.1 (on page 67) and its label file is listed in Section 2.1.1 (on page 67). The subsystems are listed in Section 2.1.2 (on page 69).

2.1.1 Summary information

System TestPipe: ¡Detailed description here¿

Interface information:

Component Pipe is in library **IncompressibleFlow/Pipe**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: TestPipe_lbl.txt

```
%SUMMARY TestPipe
%DESCRIPTION <Detailed description here>
%% Label file for system TestPipe (TestPipe_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: TestPipe_lbl.txt,v 1.2 2000/12/28 18:09:33 peterg Exp $
% %% $Log: TestPipe_lbl.txt,v $
% %% Revision 1.2  2000/12/28 18:09:33  peterg
% %% To RCS
% %%
% %% Revision 1.1  1998/11/20 08:02:53  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

%ALIAS Pipe IncompressibleFlow/Pipe

% Component type Pipe
pipe lin rho;c_p;lin,r

% Component type SS
```

```
Hy_in SS external,external
Hy_out SS external,external
Th_in SS external,external
Th_out SS external,external
```

2.1.2 Subsystems

- Pipe: Pipe containing hot incompressible liquid (1) No subsystems.

2.1.3 Pipe

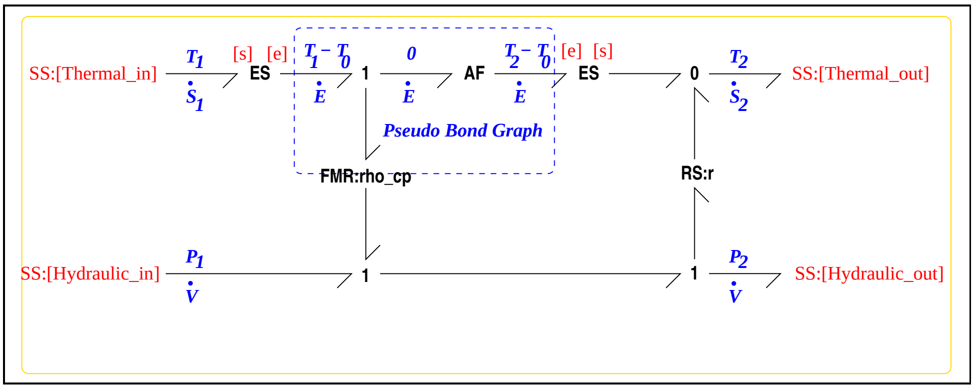


Figure 2.2: System **Pipe**: acausal bond graph

The acausal bond graph of system **Pipe** is displayed in Figure 6.5 (on page 137) and its label file is listed in Section 6.1.6 (on page 137). The subsystems are listed in Section 6.1.6 (on page 139).

The **Pipe** component represents one way flow of incompressible fluid though a pipe. Externally, it has true energy bonds: $P/\dot{V}$ (Pressure/volume-flow) representing hydraulic energy and $T/\dot{S}$(Temperature/Entropy-flow) representing convected thermal energy.

Internally, however, the thermal part is represented by a pseudo bond graph which computes the flow of internal energy $\dot{E}$ from the upstream temperature T_1 and the volumetric flow rate $\dot{V}$ as:

$$\dot{E} = \rho c_p T_1 \dot{V} \tag{2.1}$$

The AF component makes the FMR component use T_1 rather than $T_1 - T_2$. The two **ES** components provide the conversion from true to psuedo thermal bonds and vice versa.

The pipe has an resistance to flow represented by the **RS** component labeled ‘r’ which can be linear or nonlinear. The hydraulic energy loss reappears on the thermal bond of this (energy-conserving) **RS** component.

Summary information

System Pipe::Pipe containing hot incompressible liquid

Interface information:

Parameter $1 represents actual parameter **rho**

Parameter $2 represents actual parameter **c_p**

Parameter $3 represents actual parameter **flow,r**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Pipe_lbl.txt

```
%% Label file for system Pipe (Pipe_lbl.txt)
%SUMMARY Pipe: Pipe containing hot incompressible liquid
%DESCRIPTION

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
```

```
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Pipe_lbl.txt,v 1.2 1998/11/20 11:35:38 peterg Exp $
% %% $Log: Pipe_lbl.txt,v $
% %% Revision 1.2  1998/11/20 11:35:38  peterg
% %% Removed redundant port label
% %%
% %% Revision 1.1  1998/11/20 11:34:17  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out


%ALIAS $1 rho
%ALIAS $2 c_p
%ALIAS $3 flow,r


% Argument aliases

%% Each line should be of one of the following forms:
%      a comment (ie starting with %)
%      component-name cr_name arg1,arg2,..argn
%      blank

% ---- Component labels ----

% Component type FMR
rho_cp lin effort,$1*$2

% Component type RS
```

```
r $1 $3

% Component type SS
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
```

Subsystems

No subsystems.

2.2 TestPipe_cbg.ps

MTT command:

```
mtt TestPipe cbg ps
```

This representation is given as Figure 2.3 (on page 72).

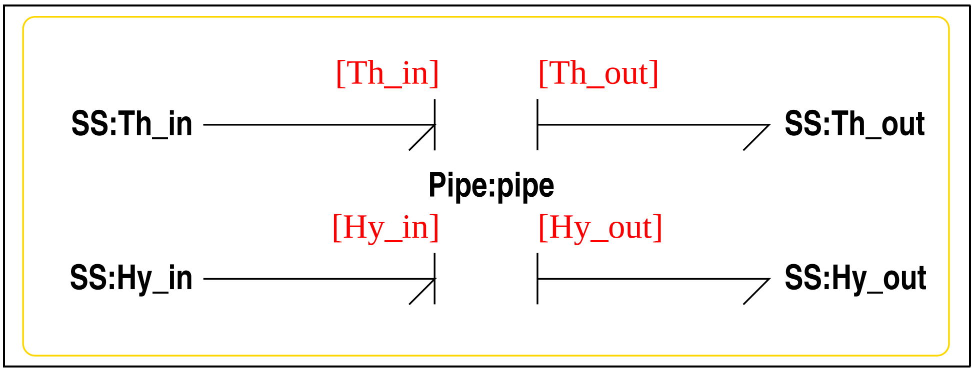


Figure 2.3: System **TestPipe**, representation cbg (-noargs)

2.3 TestPipe_struc.tex

MTT command:

```
mtt TestPipe struc tex
```

List of inputs for system TestPipe			
	Component	System	Repetition
1	Hy_in	TestPipe__Hy_in	1
2	Hy_out	TestPipe__Hy_out	1
3	Th_in	TestPipe__Th_in	1
4	Th_out	TestPipe__Th_out	1

List of outputs for system TestPipe			
	Component	System	Repetition
1	Hy_in	TestPipe__Hy_in	1
2	Hy_out	TestPipe__Hy_out	1
3	Th_in	TestPipe__Th_in	1
4	Th_out	TestPipe__Th_out	1

2.4 TestPipe_dae.tex

MTT command:

```
mtt TestPipe dae tex
```

$$
\begin{aligned}
y_1 &= \frac{(u_1 - u_2)}{r} \\
y_2 &= \frac{(u_1 - u_2)}{r} \\
y_3 &= \frac{(c_p \rho (u_1 - u_2))}{r} \\
y_4 &= \frac{\left(c_p u_1 u_3 \rho - c_p u_2 u_3 \rho + u_1^2 - 2 u_1 u_2 + u_2^2\right)}{(u_4 r)}
\end{aligned}
\tag{2.2}
$$

Chapter 3

TestPump

3.1 TestPump_abg.tex

MTT command:

```
mtt TestPump abg tex
```

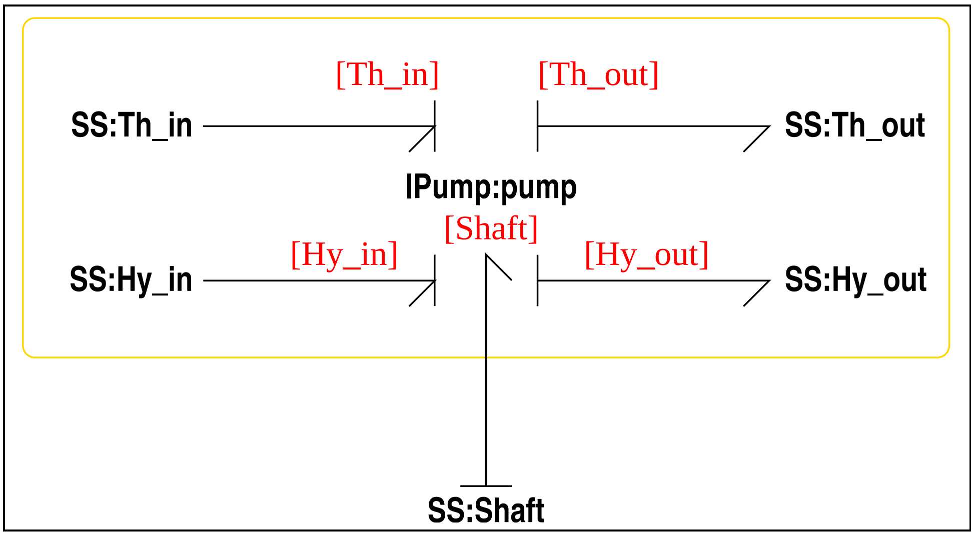


Figure 3.1: System **TestPump**: acausal bond graph

The acausal bond graph of system **TestPump** is displayed in Figure 3.1 (on page 75) and its label file is listed in Section 3.1.1 (on page 76). The subsystems are listed in Section 3.1.2 (on page 77).

3.1.1 Summary information

System TestPump::test of incompressible-flow pump component

Interface information:

Component lPump is in library **IncompressibleFlow/lPump**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: TestPump_lbl.txt

```
%SUMMARY TestPump: test of incompressible-flow pump component
%DESCRIPTION
%% Label file for system TestPump (TestPump_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: TestPump_lbl.txt,v 1.6 2000/12/28 18:10:00 peterg Exp $
% %% $Log: TestPump_lbl.txt,v $
% %% Revision 1.6  2000/12/28 18:10:00  peterg
% %% To RCS
% %%
% %% Revision 1.5  1998/11/20 13:00:27  peterg
% %% Replaces Pump by lPump in ALIAS
% %%
% %% Revision 1.4  1998/11/20 08:31:24  peterg
% %% Fixed alias error
% %%
% %% Revision 1.3  1998/11/20 08:28:41  peterg
% %% Tidied
% %%
% %% Revision 1.2  1998/11/20 08:09:57  peterg
% %% Added alias for Pump
% %%
```

```
% %% Revision 1.1  1998/11/20 08:06:28  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

%ALIAS lPump IncompressibleFlow/lPump


% Component type lPump
pump lin;lin rho;c_p;flow,r_p;k_p;flow,r_l

% Component type SS
Hy_in SS external,external
Hy_out SS external,external
Shaft SS external,external
Th_in SS external,external
Th_out SS external,external
```

3.1.2 Subsystems

- lPump: a hydraulic pump with leakage - incompressible flow (1)
  - Pipe: Pipe containing hot incompressible liquid (1)
  - Pump: a hydraulic pump - incompressible flow (1)

3.1.3 Pipe

The acausal bond graph of system **Pipe** is displayed in Figure 6.5 (on page 137) and its label file is listed in Section 6.1.6 (on page 137). The subsystems are listed in Section 6.1.6 (on page 139).

The **Pipe** component represents one way flow of incompressible fluid though a pipe. Externally, it has true energy bonds: $P/\dot{V}$ (Pressure/volume-flow) representing hydraulic energy and $T/\dot{S}$(Temperature/Entropy-flow) representing convected thermal energy.

Internally, however, the thermal part is represented by a pseudo bond graph which computes the flow of internal energy $\dot{E}$ from the upstream temperature T_1

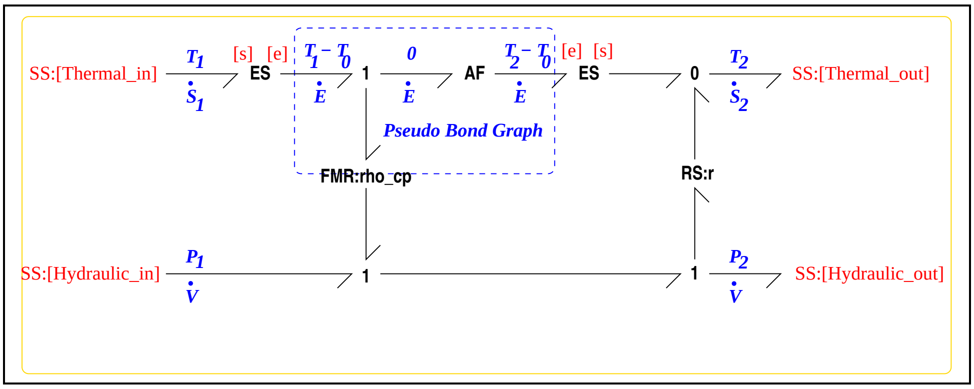


Figure 3.2: System **Pipe**: acausal bond graph

and the volumetric flow rate $\dot{V}$ as:

$$\dot{E} = \rho c_p T_1 \dot{V} \tag{3.1}$$

The *AF* component makes the *FMR* component use T_1 rather than $T_1 - T_2$.
The two **ES** components provide the conversion from true to psuedo thermal bonds and vice versa.
The pipe has an resistance to flow represented by the **RS** component labeled 'r' which can be linear or nonlinear. The hydraulic energy loss reappears on the thermal bond of this (energy-conserving) **RS** component.

Summary information

System Pipe::Pipe containing hot incompressible liquid

Interface information:

Parameter $1 represents actual parameter **rho**

Parameter $2 represents actual parameter **c_p**

Parameter $3 represents actual parameter **flow,r**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Pipe_lbl.txt

```
%% Label file for system Pipe (Pipe_lbl.txt)
%SUMMARY Pipe: Pipe containing hot incompressible liquid
%DESCRIPTION

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Pipe_lbl.txt,v 1.2 1998/11/20 11:35:38 peterg Exp $
% %% $Log: Pipe_lbl.txt,v $
% %% Revision 1.2  1998/11/20 11:35:38  peterg
% %% Removed redundant port label
% %%
% %% Revision 1.1  1998/11/20 11:34:17  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out
```

```
%ALIAS $1 rho
%ALIAS $2 c_p
%ALIAS $3 flow,r


% Argument aliases

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type FMR
rho_cp lin effort,$1*$2

% Component type RS
r $1 $3

% Component type SS
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
```

Subsystems

No subsystems.

3.1.4 Pump

The acausal bond graph of system **Pump** is displayed in Figure 5.5 (on page 110) and its label file is listed in Section 5.1.6 (on page 110). The subsystems are listed in Section 5.1.6 (on page 112).

Pump represents an ideal pump for incompressible fluid driving fluid though a **Pipe** component. The pipe component provides the correct thermal flow; if its resistance is set to zero, the pump is an ideal component.

The flow must be one way (in to out) for correct thermal properties.

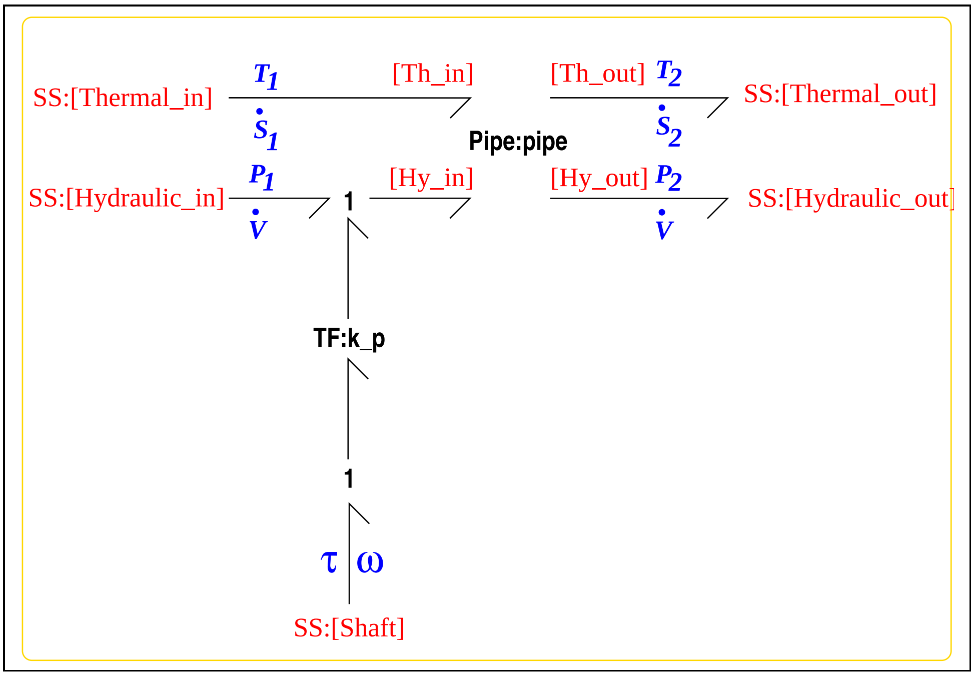


Figure 3.3: System **Pump**: acausal bond graph

The ports are

Summary information

System Pump::a hydraulic pump - incompressible flow Typical lable: pump lin rho;c_p;flow,r;k_p

Interface information:

Shaft
Component Pipe is in library **IncompressibleFlow/Pipe**

Parameter $1 represents actual parameter **lin**

Parameter $1 represents actual parameter **rho**

Parameter $2 represents actual parameter **c_p**

Parameter $3 represents actual parameter **flow,r**

Parameter $4 represents actual parameter **flow,k_p**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port Work represents actual port **Shaft**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Pump_lbl.txt

```
%SUMMARY Pump: a hydraulic pump - incompressible flow
%DESCRIPTION Typical lable: pump  lin  rho;c_p;flow,r;k_p

%% Label file for system Pump (Pump_lbl.txt)


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Pump_lbl.txt,v 1.2 1998/11/20 13:13:04 peterg Exp $
% %% $Log: Pump_lbl.txt,v $
% %% Revision 1.2  1998/11/20 13:13:04  peterg
% %% Lots of aliases!
% %%
% %% Revision 1.1  1998/11/20 10:07:14  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%ALIAS Pipe IncompressibleFlow/Pipe
%ALIAS in  Thermal_in,Hydraulic_in
```

```
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out

%ALIAS Work Shaft

%ALIAS  $1 rho
%ALIAS  $2 c_p
%ALIAS  $3 flow,r
%ALIAS  $4 flow,k_p

%ALIAS $1 lin

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank


% Component type Pipe
pipe lin rho;c_p;flow,r

% Component type TF
k_p lin flow,k_p

% Component type SS
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
[Shaft] SS external,external
```

Subsystems

- Pipe: Pipe containing hot incompressible liquid (1) No subsystems.

3.1.5 lPump

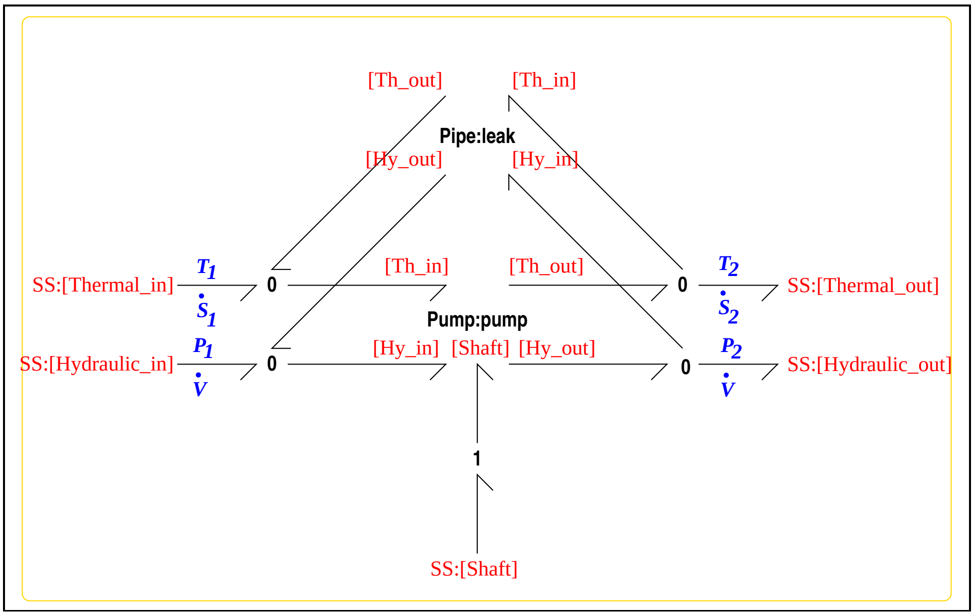


Figure 3.4: System **lPump**: acausal bond graph

The acausal bond graph of system **lPump** is displayed in Figure 5.7 (on page 116) and its label file is listed in Section 5.1.8 (on page 116). The subsystems are listed in Section 5.1.8 (on page 118).
lPump corrsponds to the **Pump** component but with a backflow leakage around the ump driven by the pressure drop across the pump. This leakage is implemented using the **Pipe** component to give the correct thermal behaviour.

Summary information

System lPump::a hydraulic pump with leakage - incompressible flow

Interface information:

Component Pipe is in library **IncompressibleFlow/Pipe**

Component Pump is in library **IncompressibleFlow/Pump**

Parameter $1 represents actual parameter **rho**

Parameter $2 represents actual parameter **c_p**

Parameter $3 represents actual parameter **flow,r**

Parameter $4 represents actual parameter **flow,k_p**

Parameter $5 represents actual parameter **flow,r_l**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: lPump_lbl.txt

```
%% Label file for system lPump (lPump_lbl.txt)
%SUMMARY lPump: a hydraulic pump with leakage - incompressible flow

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: lPump_lbl.txt,v 1.1 1998/11/20 13:13:24 peterg Exp $
% %% $Log: lPump_lbl.txt,v $
% %% Revision 1.1  1998/11/20 13:13:24  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Component aliases
%ALIAS Pipe IncompressibleFlow/Pipe
%ALIAS Pump IncompressibleFlow/Pump
```

```
% Port aliases
%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out

% Argument aliases
%ALIAS  $1 rho
%ALIAS  $2 c_p
%ALIAS  $3 flow,r
%ALIAS  $4 flow,k_p
%ALIAS  $5 flow,r_l


%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type Pipe
leak lin $1;$2;$5

% Component type Pump
pump lin $1;$2;$3;$4

% Component type SS
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
[Shaft] SS external,external
```

Subsystems

- Pipe: Pipe containing hot incompressible liquid (1) No subsystems.
- Pump: a hydraulic pump - incompressible flow (1)
  - Pipe: Pipe containing hot incompressible liquid (1)

3.2 TestPump_cbg.ps

MTT command:

```
mtt TestPump cbg ps
```

This representation is given as Figure 3.5 (on page 87).

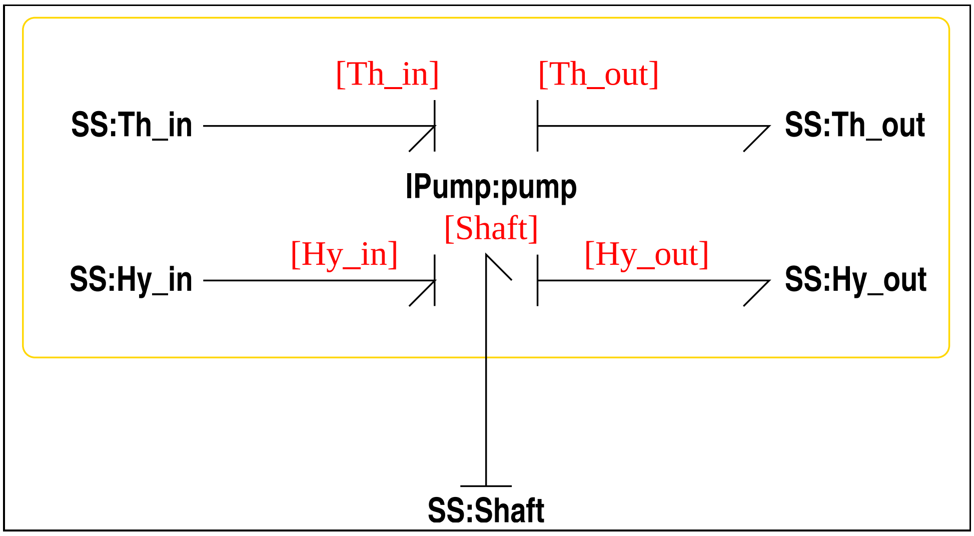


Figure 3.5: System **TestPump**, representation cbg (-noargs)

3.3 TestPump_struc.tex

MTT command:

```
mtt TestPump struc tex
```

List of inputs for system TestPump			
	Component	System	Repetition
1	Hy_in	TestPump__Hy_in	1
2	Hy_out	TestPump__Hy_out	1
3	Shaft	TestPump__Shaft	1
4	Th_in	TestPump__Th_in	1
5	Th_out	TestPump__Th_out	1

List of outputs for system TestPump			
	Component	System	Repetition
1	Hy_in	TestPump__Hy_in	1
2	Hy_out	TestPump__Hy_out	1
3	Shaft	TestPump__Shaft	1
4	Th_in	TestPump__Th_in	1
5	Th_out	TestPump__Th_out	1

3.4 TestPump_ode.tex

MTT command:

```
mtt TestPump ode tex
```

$$
\begin{aligned}
y_1 &= \frac{(k_p u_3 r_l + u_1 - u_2)}{r_l} \\
y_2 &= \frac{(k_p u_3 r_l + u_1 - u_2)}{r_l} \\
y_3 &= \frac{(k_p u_3 r_p - u_1 + u_2)}{k_p} \\
y_4 &= \frac{\left(c_p k_p u_3 u_4 r_l \rho + c_p u_1 u_5 \rho - c_p u_2 u_5 \rho - u_1^2 + 2 u_1 u_2 - u_2^2\right)}{(u_4 r_l)} \\
y_5 &= \frac{\left(c_p k_p u_3 u_4 r_l \rho + c_p u_1 u_5 \rho - c_p u_2 u_5 \rho + k_p^2 u_3^2 r_l r_p\right)}{(u_5 r_l)}
\end{aligned}
\tag{3.2}
$$

Chapter 4

TestTank

4.1 TestTank_abg.tex

MTT command:

```
mtt TestTank abg tex
```

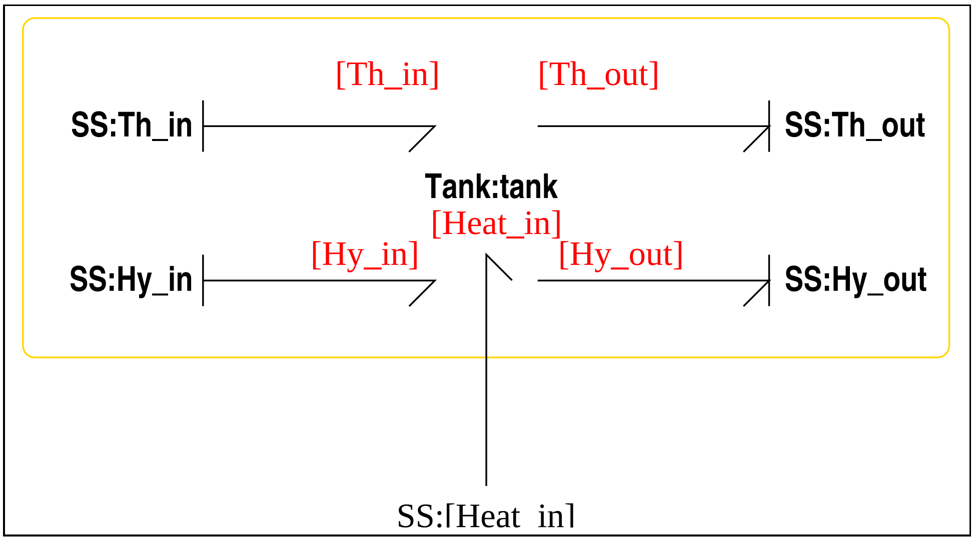


Figure 4.1: System **TestTank**: acausal bond graph

The acausal bond graph of system **TestTank** is displayed in Figure 4.1 (on page 89) and its label file is listed in Section 4.1.1 (on page 90). The subsystems are listed in Section 4.1.2 (on page 91).

4.1.1 Summary information

System TestTank::Equations for incompressible-flow Tank component

Interface information:

Component Tank is in library **IncompressibleFlow/Tank**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: TestTank_lbl.txt

```
%SUMMARY TestTank: Equations for incompressible-flow Tank componer
%DESCRIPTION
%% Label file for system TestTank (TestTank_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: TestTank_lbl.txt,v 1.4 2000/12/28 18:10:28 peterg Exp $
% %% $Log: TestTank_lbl.txt,v $
% %% Revision 1.4  2000/12/28 18:10:28  peterg
% %% To RCS
% %%
% %% Revision 1.3  1998/11/20 08:36:30  peterg
% %% Corrected alias
% %%
% %% Revision 1.2  1998/11/20 08:09:20  peterg
% %% Added alias for Tank
% %%
% %% Revision 1.1  1998/11/20 08:07:20  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
```

```
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

%ALIAS Tank IncompressibleFlow/Tank

% Component type SS
Heat_in SS external,external
Hy_in SS external,external
Hy_out SS external,external
Th_in SS external,external
Th_out SS external,external

% Component type Tank
tank none rho;c_p;c
```

4.1.2 Subsystems

- Tank: Tank of hot incompressible liquid (1) No subsystems.

4.1.3 Tank

The acausal bond graph of system **Tank** is displayed in Figure 5.6 (on page 113) and its label file is listed in Section 5.1.7 (on page 113). The subsystems are listed in Section 5.1.7 (on page 115).

Summary information

System Tank::Tank of hot incompressible liquid c is the pressure constant: P=(rho*V)/c Typical lable: tank Tank rho;c_p;c

Interface information:

Parameter $1 represents actual parameter **rho**

Parameter $2 represents actual parameter **c_p**

Parameter $3 represents actual parameter **c**

Port Heat represents actual port **Heat_in**

Port Hy_in represents actual port **Hydraulic_in**

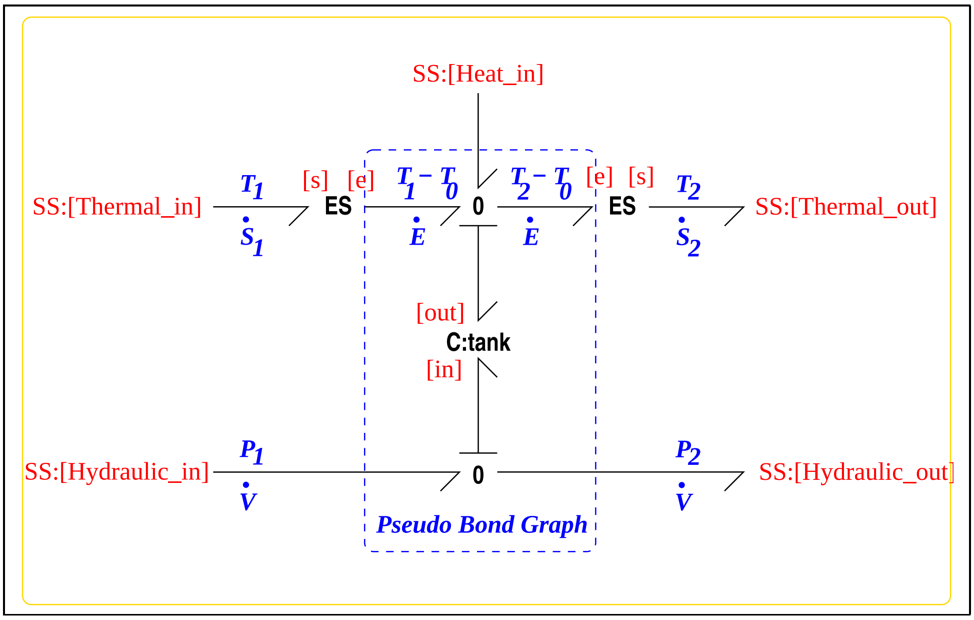


Figure 4.2: System **Tank**: acausal bond graph

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Tank_lbl.txt

```
%% Label file for system Tank (Tank_lbl.txt)
%SUMMARY Tank: Tank of hot incompressible liquid
%DESCRIPTION c is the pressure constant: P=(rho*V)/c
```

```
%DESCRIPTION Typical lable: tank  Tank  rho;c_p;c

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Tank_lbl.txt,v 1.3 1998/11/20 13:20:27 peterg Exp $
% %% $Log: Tank_lbl.txt,v $
% %% Revision 1.3  1998/11/20 13:20:27  peterg
% %% Aliased ports
% %%
% %% Revision 1.2  1998/11/20 09:46:34  peterg
% %% Modernised lbl syntax
% %%
% %% Revision 1.1  1998/11/20 08:57:19  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out

%ALIAS Heat Heat_in

% Argument aliases
%ALIAS $1 rho
%ALIAS $2 c_p
%ALIAS $3 c

%% Each line should be of one of the following forms:
%      a comment (ie starting with %)
%      component-name cr_name arg1,arg2,..argn
%      blank
```

```
% ---- Component labels ----

% Component type C
tank Tank         rho,c_p,c

% Component type SS
[Heat_in] SS external,external
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
```

Subsystems

No subsystems.

4.2 TestTank_cbg.ps

MTT command:

```
mtt TestTank cbg ps
```

This representation is given as Figure 4.3 (on page 94).

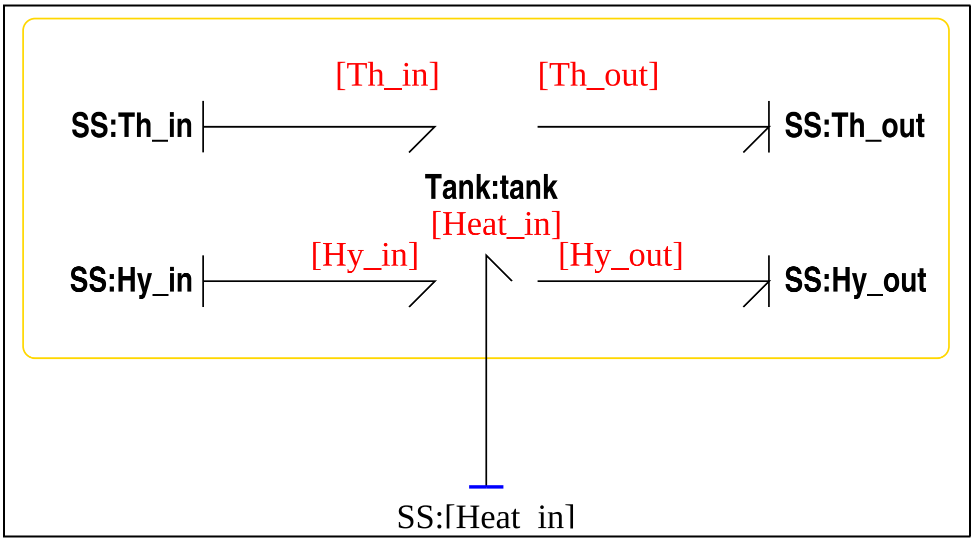


Figure 4.3: System **TestTank**, representation cbg (-noargs)

4.3 TestTank_struc.tex

MTT command:

```
mtt TestTank struc tex
```

List of inputs for system TestTank			
	Component	System	Repetition
1	Heat_in	TestTank__Heat_in	1
2	Hy_in	TestTank__Hy_in	1
3	Hy_out	TestTank__Hy_out	1
4	Th_in	TestTank__Th_in	1
5	Th_out	TestTank__Th_out	1

List of outputs for system TestTank			
	Component	System	Repetition
1	Heat_in	TestTank__Heat_in	1
2	Hy_in	TestTank__Hy_in	1
3	Hy_out	TestTank__Hy_out	1
4	Th_in	TestTank__Th_in	1
5	Th_out	TestTank__Th_out	1

List of states for system TestTank			
	Component	System	Repetition
1	tank	TestTank__tank__tank	1
2	tank	TestTank__tank__tank_2	1

4.4 TestTank_ode.tex

MTT command:

```
mtt TestTank ode tex
```

$$\begin{aligned} \dot{x}_1 &= u_2 - u_3 \\ \dot{x}_2 &= \frac{(c_p u_1 x_1 \rho + u_4 x_2 - u_5 x_2)}{(c_p x_1 \rho)} \end{aligned} \tag{4.1}$$

$$
\begin{aligned}
y_1 &= \frac{x_2}{(c_p x_1 \rho)} \\
y_2 &= \frac{(x_1 \rho)}{c} \\
y_3 &= \frac{(x_1 \rho)}{c} \\
y_4 &= \frac{x_2}{(c_p x_1 \rho)} \\
y_5 &= \frac{x_2}{(c_p x_1 \rho)}
\end{aligned} \tag{4.2}
$$

Part III

Incompressible-Systems

Chapter 5

LiquidTurbine

5.1 LiquidTurbine_abg.tex *(-o -ss)*

MTT command:

```
mtt -o -ss LiquidTurbine abg tex
```

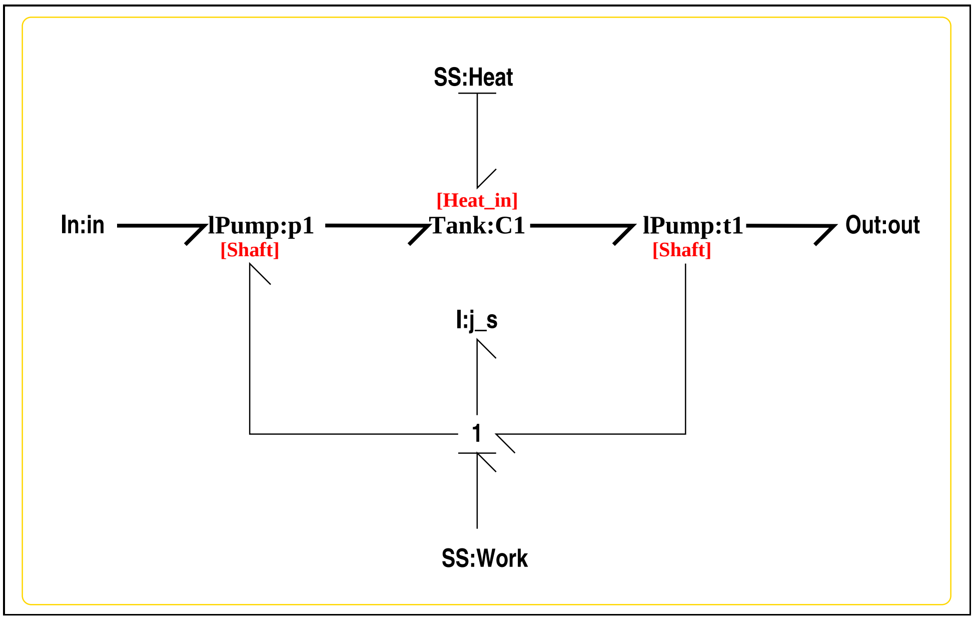


Figure 5.1: System **LiquidTurbine**: acausal bond graph

The acausal bond graph of system **LiquidTurbine** is displayed in Figure 5.1 (on page 99) and its label file is listed in Section 5.1.1 (on page 100). The subsystems are listed in Section 5.1.2 (on page 102).

LiquidTurbine can be regarded as a single-spool gas turbine with an incompressible working fluid. Of course, such a device cannot convert heat to work; however, it provides a useful first step towards modelling a gas turbine. There are three main components:

1. p1 – a leaky pump **lPump** component. This is analogous to the gas turbine compressor.

2. c1 – a tank **Tank** component. This is analogous to the gas turbine combustion chamber.

3. t1 – a leaky turbine **lTurb** component. This is analogous to the gas turbine turbine.

The components **In** and **Out** provide the inlet and outlet conditions.

5.1.1 Summary information

System LiquidTurbine: ¡Detailed description here¿

Interface information:

Component Tank is in library **IncompressibleFlow/Tank**

Component lPump is in library **IncompressibleFlow/lPump**

Variable declarations:

p_0

q_0

t_0

Units declarations:

This component has no UNITs declarations

The label file: LiquidTurbine_lbl.txt

```
%SUMMARY LiquidTurbine
%DESCRIPTION <Detailed description here>
%% Label file for system LiquidTurbine (LiquidTurbine_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: LiquidTurbine_lbl.txt,v 1.1 2000/12/28 18:11:16 peterg Exp $
% %% $Log: LiquidTurbine_lbl.txt,v $
% %% Revision 1.1  2000/12/28 18:11:16  peterg
% %% To RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%ALIAS lPump IncompressibleFlow/lPump
%ALIAS Tank IncompressibleFlow/Tank

%VAR p_0
%VAR q_0
%VAR t_0


%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

% Component type I
j_s lin flow,j_s

% Component type Pump
p1 lin;lin rho;c_p;flow,r_p;k_p;flow,r_pl

% Component type SS
Heat SS external,external
Work SS external,external

% Component type Turb
t1 lin;lin rho;c_p;flow,r_t;k_t;flow,r_tl
```

```
% Component type Tank
C1 none rho;c_p;c_t

% Component type In
in

% Component type Out
out
```

5.1.2 Subsystems

- In: Inflow conditions (1) No subsystems.
- Out: Outflow conditions (1) No subsystems.
- Tank: Tank of hot incompressible liquid (1) No subsystems.
- lPump: a hydraulic pump with leakage - incompressible flow (2)
  - Pipe: Pipe containing hot incompressible liquid (1)
  - Pump: a hydraulic pump - incompressible flow (1)

5.1.3 In

The acausal bond graph of system **In** is displayed in Figure 6.3 (on page 133) and its label file is listed in Section 6.1.4 (on page 133). The subsystems are listed in Section 6.1.4 (on page 134).

Summary information

System In::Inflow conditions ¡Detailed description here¿

Interface information:

Port Hy_out represents actual port **Hydraulic_out**

Port Th_out represents actual port **Thermal_out**

Port out represents actual port **Thermal_out,Hydraulic_out**

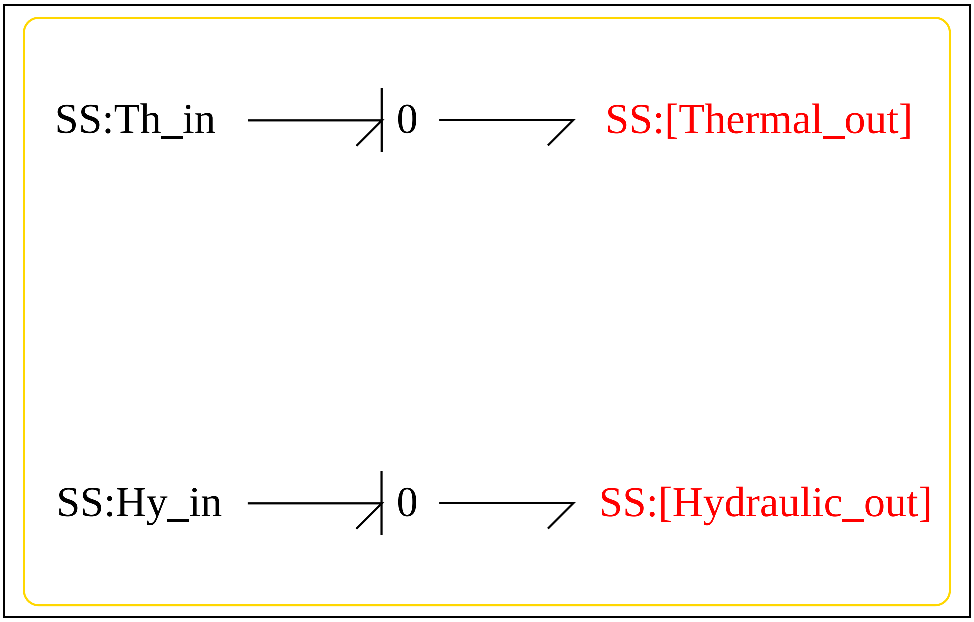


Figure 5.2: System **In**: acausal bond graph

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: In_lbl.txt

```
%SUMMARY In: Inflow conditions
%DESCRIPTION <Detailed description here>
%% Label file for system In (In_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: In_lbl.txt,v 1.1 2000/12/28 18:11:16 peterg Exp $
% %% $Log: In_lbl.txt,v $
% %% Revision 1.1  2000/12/28 18:11:16  peterg
% %% To RCS
% %%
```

```
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

%ALIAS out Thermal_out,Hydraulic_out
%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out



% Component type SS
Hy_in SS p_0,external
Th_in SS t_0,external
[Hydraulic_out] SS external,external
[Thermal_out] SS external,external
```

Subsystems

No subsystems.

5.1.4 Out

The acausal bond graph of system **Out** is displayed in Figure 6.4 (on page 135) and its label file is listed in Section 6.1.5 (on page 135). The subsystems are listed in Section 6.1.5 (on page 136).

Summary information

System Out::Outflow conditions ¡Detailed description here¿

Interface information:

Port Hy_in represents actual port **Hydraulic_in**

Port Th_in represents actual port **Thermal_in**

Port in represents actual port **Thermal_in,Hydraulic_in**

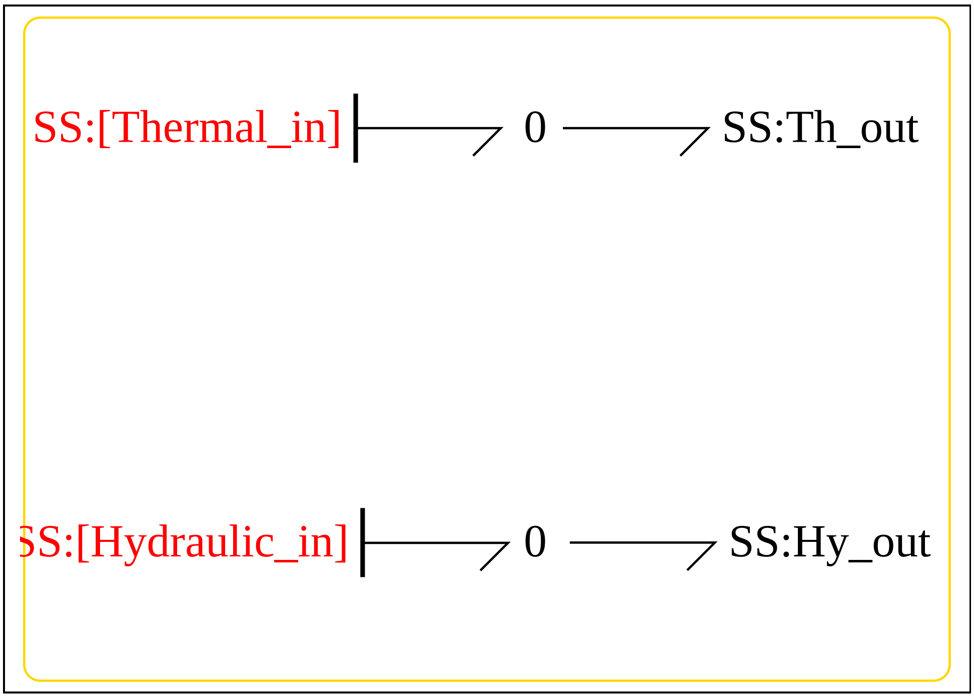


Figure 5.3: System **Out**: acausal bond graph

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Out_lbl.txt

```
%SUMMARY Out: Outflow conditions
%DESCRIPTION <Detailed description here>
%% Label file for system Out (Out_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Out_lbl.txt,v 1.1 2000/12/28 18:11:16 peterg Exp $
% %% $Log: Out_lbl.txt,v $
% %% Revision 1.1  2000/12/28 18:11:16  peterg
```

```
% %% To RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in


% Component type SS
Hy_out SS p_0,external
Th_out SS t_0,external
[Hydraulic_in] SS external,external
[Thermal_in] SS external,external
```

Subsystems

No subsystems.

5.1.5 Pipe

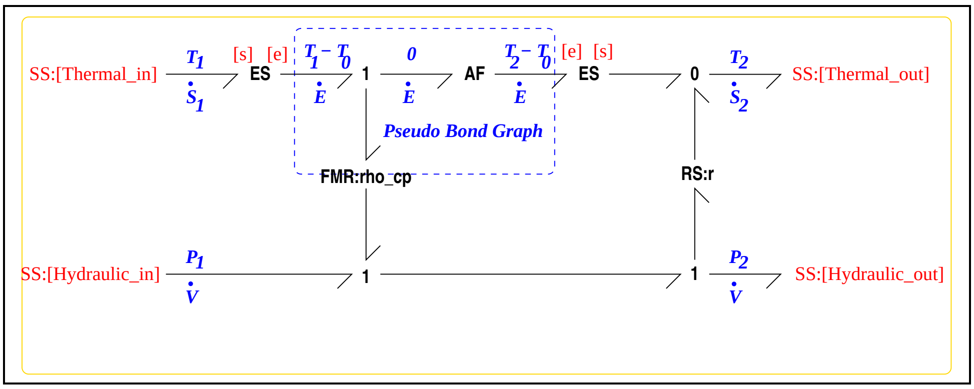


Figure 5.4: System **Pipe**: acausal bond graph

The acausal bond graph of system **Pipe** is displayed in Figure 6.5 (on page 137) and its label file is listed in Section 6.1.6 (on page 137). The subsystems are listed in Section 6.1.6 (on page 139).

The **Pipe** component represents one way flow of incompressible fluid though a pipe. Externally, it has true energy bonds: $P/\dot{V}$ (Pressure/volume-flow) representing hydraulic energy and $T/\dot{S}$(Temperature/Entropy-flow) representing convected thermal energy.

Internally, however, the thermal part is represented by a pseudo bond graph which computes the flow of internal energy $\dot{E}$ from the upstream temperature T_1 and the volumetric flow rate $\dot{V}$ as:

$$\dot{E} = \rho c_p T_1 \dot{V} \tag{5.1}$$

The *AF* component makes the *FMR* component use T_1 rather than $T_1 - T_2$. The two **ES** components provide the conversion from true to psuedo thermal bonds and vice versa.

The pipe has an resistance to flow represented by the **RS** component labeled 'r' which can be linear or nonlinear. The hydraulic energy loss reappears on the thermal bond of this (energy-conserving) **RS** component.

Summary information

System Pipe::Pipe containing hot incompressible liquid

Interface information:

Parameter $1 represents actual parameter **rho**

Parameter $2 represents actual parameter **c_p**

Parameter $3 represents actual parameter **flow,r**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Pipe_lbl.txt

```
%% Label file for system Pipe (Pipe_lbl.txt)
%SUMMARY Pipe: Pipe containing hot incompressible liquid
%DESCRIPTION

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Pipe_lbl.txt,v 1.2 1998/11/20 11:35:38 peterg Exp $
% %% $Log: Pipe_lbl.txt,v $
% %% Revision 1.2  1998/11/20 11:35:38  peterg
% %% Removed redundant port label
% %%
% %% Revision 1.1  1998/11/20 11:34:17  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out


%ALIAS $1 rho
%ALIAS $2 c_p
%ALIAS $3 flow,r
```

```
% Argument aliases

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type FMR
rho_cp lin effort,$1*$2

% Component type RS
r $1 $3

% Component type SS
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
```

Subsystems

No subsystems.

5.1.6 Pump

The acausal bond graph of system **Pump** is displayed in Figure 5.5 (on page 110) and its label file is listed in Section 5.1.6 (on page 110). The subsystems are listed in Section 5.1.6 (on page 112).

Pump represents an ideal pump for incompressible fluid driving fluid though a **Pipe** component. The pipe component provides the correct thermal flow; if its resistance is set to zero, the pump is an ideal component.

The flow must be one way (in to out) for correct thermal properties.

The ports are

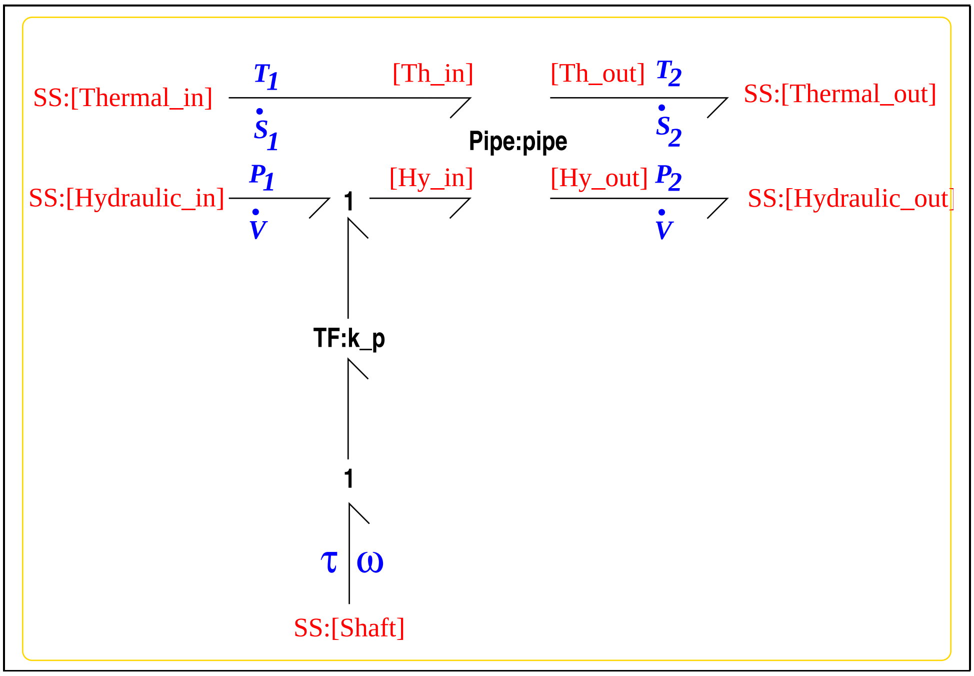


Figure 5.5: System **Pump**: acausal bond graph

Summary information

System Pump::a hydraulic pump - incompressible flow Typical lable: pump lin rho;c_p;flow,r;k_p

Interface information:

Component Pipe is in library **IncompressibleFlow/Pipe**

Parameter $1 represents actual parameter **lin**

Parameter $1 represents actual parameter **rho**

Parameter $2 represents actual parameter **c_p**

Parameter $3 represents actual parameter **flow,r**

Parameter $4 represents actual parameter **flow,k_p**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port Work represents actual port **Shaft**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Pump_lbl.txt

```
%SUMMARY Pump: a hydraulic pump - incompressible flow
%DESCRIPTION Typical lable: pump  lin  rho;c_p;flow,r;k_p

%% Label file for system Pump (Pump_lbl.txt)


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Pump_lbl.txt,v 1.2 1998/11/20 13:13:04 peterg Exp $
% %% $Log: Pump_lbl.txt,v $
% %% Revision 1.2  1998/11/20 13:13:04  peterg
% %% Lots of aliases!
% %%
% %% Revision 1.1  1998/11/20 10:07:14  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%ALIAS Pipe IncompressibleFlow/Pipe
%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out
```

```
%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out

%ALIAS Work Shaft

%ALIAS  $1 rho
%ALIAS  $2 c_p
%ALIAS  $3 flow,r
%ALIAS  $4 flow,k_p

%ALIAS $1 lin

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank


% Component type Pipe
pipe lin rho;c_p;flow,r

% Component type TF
k_p lin flow,k_p

% Component type SS
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
[Shaft] SS external,external
```

Subsystems

- Pipe: Pipe containing hot incompressible liquid (1) No subsystems.

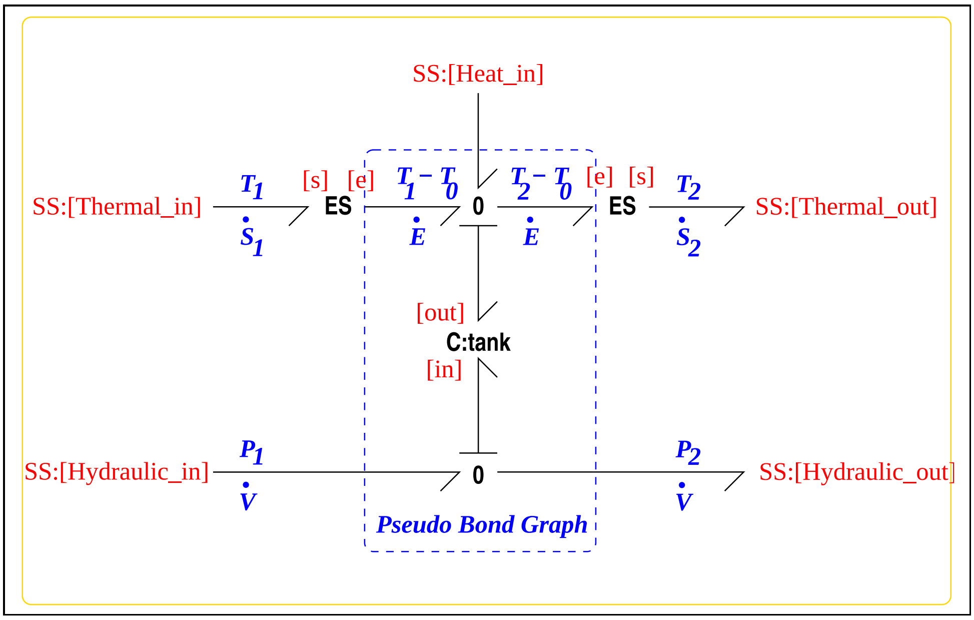


Figure 5.6: System **Tank**: acausal bond graph

5.1.7 Tank

The acausal bond graph of system **Tank** is displayed in Figure 5.6 (on page 113) and its label file is listed in Section 5.1.7 (on page 113). The subsystems are listed in Section 5.1.7 (on page 115).

Summary information

System Tank::Tank of hot incompressible liquid c is the pressure constant: P=(rho*V)/c Typical lable: tank Tank rho;c_p;c

Interface information:

Parameter $1 represents actual parameter **rho**

Parameter $2 represents actual parameter **c_p**

Parameter $3 represents actual parameter **c**

Port Heat represents actual port **Heat_in**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Tank_lbl.txt

```
%% Label file for system Tank (Tank_lbl.txt)
%SUMMARY Tank: Tank of hot incompressible liquid
%DESCRIPTION c is the pressure constant: P=(rho*V)/c
%DESCRIPTION Typical lable: tank  Tank  rho;c_p;c

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Tank_lbl.txt,v 1.3 1998/11/20 13:20:27 peterg Exp $
% %% $Log: Tank_lbl.txt,v $
% %% Revision 1.3  1998/11/20 13:20:27  peterg
% %% Aliased ports
% %%
% %% Revision 1.2  1998/11/20 09:46:34  peterg
% %% Modernised lbl syntax
% %%
% %% Revision 1.1  1998/11/20 08:57:19  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in  Thermal_in,Hydraulic_in
```

```
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out

%ALIAS Heat Heat_in

% Argument aliases
%ALIAS $1 rho
%ALIAS $2 c_p
%ALIAS $3 c

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type C
tank Tank        rho,c_p,c

% Component type SS
[Heat_in] SS external,external
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
```

Subsystems

No subsystems.

5.1.8 lPump

The acausal bond graph of system **lPump** is displayed in Figure 5.7 (on page 116) and its label file is listed in Section 5.1.8 (on page 116). The subsystems are listed in Section 5.1.8 (on page 118).

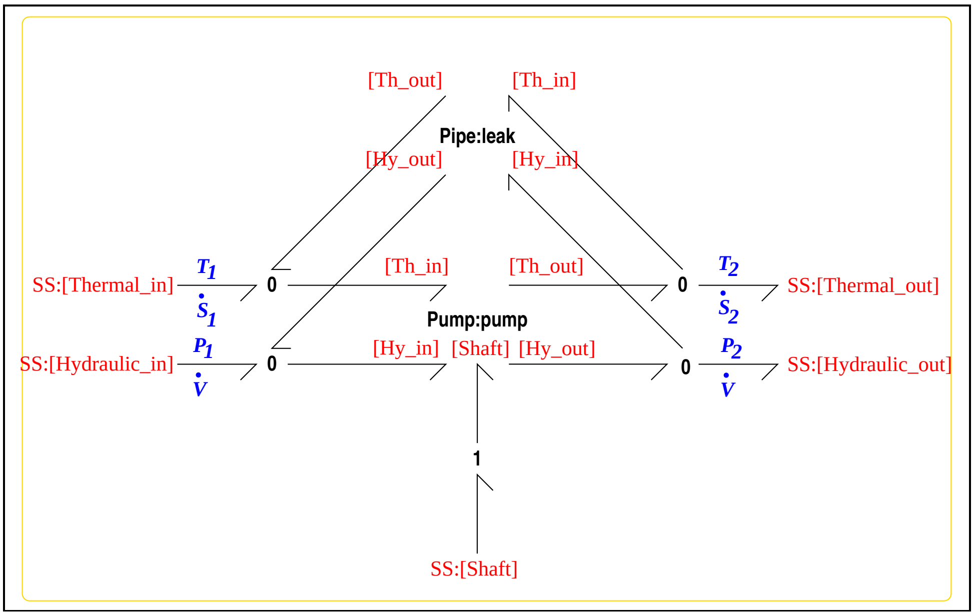


Figure 5.7: System **lPump**: acausal bond graph

lPump corrsponds to the **Pump** component but with a backflow leakage around the ump driven by the pressure drop across the pump. This leakage is implemented using the **Pipe** component to give the correct thermal behaviour.

Summary information

System lPump::a hydraulic pump with leakage - incompressible flow

Interface information:

Component Pipe is in library **IncompressibleFlow/Pipe**

Component Pump is in library **IncompressibleFlow/Pump**

Parameter $1 represents actual parameter **rho**

Parameter $2 represents actual parameter **c_p**

Parameter $3 represents actual parameter **flow,r**

Parameter $4 represents actual parameter **flow,k_p**

Parameter $5 represents actual parameter **flow,r_l**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: lPump_lbl.txt

```
%% Label file for system lPump (lPump_lbl.txt)
%SUMMARY lPump: a hydraulic pump with leakage - incompressible flow

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: lPump_lbl.txt,v 1.1 1998/11/20 13:13:24 peterg Exp $
% %% $Log: lPump_lbl.txt,v $
% %% Revision 1.1  1998/11/20 13:13:24  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Component aliases
%ALIAS Pipe IncompressibleFlow/Pipe
%ALIAS Pump IncompressibleFlow/Pump

% Port aliases
%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out
```

```
%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out

% Argument aliases
%ALIAS  $1 rho
%ALIAS  $2 c_p
%ALIAS  $3 flow,r
%ALIAS  $4 flow,k_p
%ALIAS  $5 flow,r_l


%% Each line should be of one of the following forms:
%      a comment (ie starting with %)
%      component-name cr_name arg1,arg2,..argn
%      blank

% ---- Component labels ----

% Component type Pipe
leak lin $1;$2;$5

% Component type Pump
pump lin $1;$2;$3;$4

% Component type SS
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
[Shaft] SS external,external
```

Subsystems

- Pipe: Pipe containing hot incompressible liquid (1) No subsystems.
- Pump: a hydraulic pump - incompressible flow (1)
  - Pipe: Pipe containing hot incompressible liquid (1)

5.2 LiquidTurbine_struc.tex *(-o -ss)*

MTT command:

```
mtt -o -ss LiquidTurbine struc tex
```

List of inputs for system LiquidTurbine

	Component	System	Repetition
1	Heat	LiquidTurbine__Heat	1
2	Work	LiquidTurbine__Work	1

List of outputs for system LiquidTurbine

	Component	System	Repetition
1	Heat	LiquidTurbine__Heat	1
2	Work	LiquidTurbine__Work	1
3	Hy_in	LiquidTurbine__in__Hy_in	1
4	Th_in	LiquidTurbine__in__Th_in	1
5	Hy_out	LiquidTurbine__out__Hy_out	1
6	Th_out	LiquidTurbine__out__Th_out	1

List of states for system LiquidTurbine

	Component	System	Repetition
1	j_s	LiquidTurbine__j_s	1
2	tank	LiquidTurbine__C1__tank	1
3	tank	LiquidTurbine__C1__tank_2	1

5.3 LiquidTurbine_sympar.tex *(-o -ss)*

MTT command:

```
mtt -o -ss LiquidTurbine sympar tex
```

Parameter	System
c_p	LiquidTurbine
c_t	LiquidTurbine
j_s	LiquidTurbine
k_p	LiquidTurbine
k_t	LiquidTurbine
p_0	LiquidTurbine__in,LiquidTurbine__out,
q_0	LiquidTurbine
r_p	LiquidTurbine
r_pl	LiquidTurbine
r_t	LiquidTurbine
r_tl	LiquidTurbine
rho	LiquidTurbine
t_0	LiquidTurbine__in,LiquidTurbine__out,

Table 5.1: Parameters

5.4 LiquidTurbine_ss.tex *(-o -ss)*

MTT command:

```
mtt -o -ss LiquidTurbine ss tex
```

$$x = \begin{pmatrix} 1 \\ \frac{(c_t p_0)}{\rho} \\ c_p c_t p_0 t_0 \end{pmatrix} \tag{5.2}$$

$$u = \begin{pmatrix} q_0 \\ 0 \end{pmatrix} \tag{5.3}$$

$$y = \begin{pmatrix} t_0 \\ \frac{1}{j_s} \\ \frac{k_p}{j_s} \\ \frac{(c_p k_p \rho)}{j_s} \\ \frac{k_t}{j_s} \\ \frac{(k_t (c_p j_s \rho t_0 + k_t r_t))}{(j_s^2 t_0)} \end{pmatrix} \tag{5.4}$$

$$\dot{x} = \begin{pmatrix} \frac{\left(-\left(r_p+r_t\right)\right)}{j_s} \\ \frac{\left(k_p-k_t\right)}{j_s} \\ \frac{\left(c_p j_s k_p \rho t_0 - c_p j_s k_t \rho t_0 + j_s^2 q_0 + k_p^2 r_p\right)}{j_s^2} \end{pmatrix} \tag{5.5}$$

5.5 LiquidTurbine_ode.tex *(-o -ss)*

MTT command:

```
mtt -o -ss LiquidTurbine ode tex
```

$$\begin{aligned}
\dot{x}_1 &= \frac{(c_t j_s k_p k_t u_2 - c_t j_s k_p p_0 + c_t j_s k_t p_0 - c_t k_p k_t x_1 r_p - c_t k_p k_t x_1 r_t + j_s k_p x_2 \rho - j_s k_t x_2 \rho)}{(c_t j_s k_p k_t)} \\
\dot{x}_2 &= \frac{\left(c_t j_s p_0 r_{pl} + c_t j_s p_0 r_{tl} + c_t k_p x_1 r_{pl} r_{tl} - c_t k_t x_1 r_{pl} r_{tl} - j_s x_2 r_{pl} \rho - j_s x_2 r_{tl} \rho\right)}{\left(c_t j_s r_{pl} r_{tl}\right)} \\
\dot{x}_3 &= \frac{\left(c_p c_t^2 j_s^2 x_2 p_0 r_{pl} \rho t_0 + c_p c_t^2 j_s k_p x_1 x_2 r_{pl} r_{tl} \rho t_0 - c_p c_t j_s^2 x_2^2 r_{pl} \rho^2 t_0 + c_t^2 j_s^2 u_1 x_2 r_{pl} r_{tl} + c_t^2 j_s^2 x_2 p_0^2 r_{pl} + c_t^2 j_s^2 x_3 p_0 \ldots\right.}{\left(c_t^2 j_s^2 x_2 r_{pl} r_{tl}\right)}
\end{aligned} \tag{5.6}$$

$$\begin{aligned}
y_1 &= \frac{x_3}{(c_p x_2 \rho)} \\
y_2 &= \frac{x_1}{j_s} \\
y_3 &= \frac{\left(c_t j_s p_0 + c_t k_p x_1 r_{pl} - j_s x_2 \rho\right)}{\left(c_t j_s r_{pl}\right)} \\
y_4 &= \frac{\left(c_p c_t^2 k_p x_1 x_2 r_{pl} \rho t_0 - c_t^2 j_s x_2 p_0^2 + c_t^2 j_s x_3 p_0 + 2 c_t j_s x_2^2 p_0 \rho - c_t j_s x_2 x_3 \rho - j_s x_2^3 \rho^2\right)}{\left(c_t^2 j_s x_2 r_{pl} t_0\right)} \\
y_5 &= \frac{(-c_t j_s p_0 + c_t k_t x_1 r_{tl} + j_s x_2 \rho)}{(c_t j_s r_{tl})} \\
y_6 &= \frac{\left(-c_p c_t j_s^2 x_2 p_0 \rho t_0 + c_p j_s^2 x_2^2 \rho^2 t_0 + c_t j_s k_t x_1 x_3 r_{tl} + c_t k_t^2 x_1^2 x_2 r_t r_{tl}\right)}{(c_t j_s^2 x_2 r_{tl} t_0)}
\end{aligned} \tag{5.7}$$

5.6 LiquidTurbine_numpar.txt *(-o -ss)*

MTT command:

```
mtt -o -ss LiquidTurbine numpar txt
```

```
# Numerical parameter file (LiquidTurbine_numpar.txt)
# Generated by MTT at Mon Mar  9 09:16:28 GMT 1998

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: LiquidTurbine_numpar.txt,v 1.2 2003/08/19 13:06:02 gawth
# %% $Log: LiquidTurbine_numpar.txt,v $
# %% Revision 1.2  2003/08/19 13:06:02  gawthrop
# %% Updated for new MTT
# %%
# %% Revision 1.1  2000/12/28 18:11:16  peterg
# %% To RCS
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Parameters
c_p =  1.0; # LiquidTurbine
c_t =  1.0; # LiquidTurbine
j_s =  1.0; # LiquidTurbine
k_p =  1.0; # LiquidTurbine
k_t =  1.0; # LiquidTurbine
p_0 =  1e5; # In,Out
q_0 =  1e5; # Heat in
r_p =  1.0; # LiquidTurbine
r_pl =  100.0; # LiquidTurbine
r_t =  1.0; # LiquidTurbine
r_tl =  100.0; # LiquidTurbine
rho =  1.0; # LiquidTurbine
t_0 =  300.0; # In,Out
```

5.7 LiquidTurbine_input.txt *(-o -ss)*

MTT command:

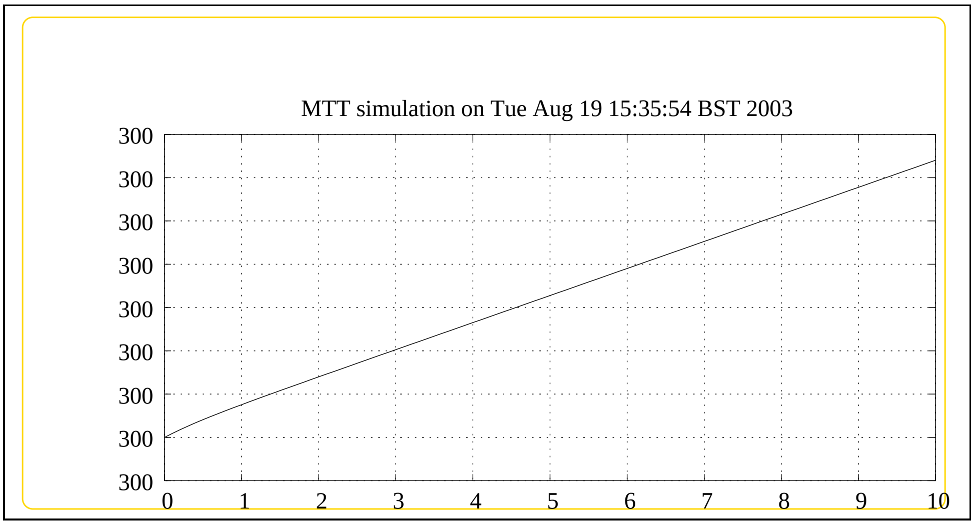
```
mtt -o -ss LiquidTurbine input txt

## -*-octave-*- Put Emacs into octave-mode ##

##
## System LiquidTurbine, representation input, language txt;
## File LiquidTurbine_input.txt;
## Generated by MTT on Sun Aug 17 14:16:29 BST 2003;



LiquidTurbine__Heat = 1.0; # Default
LiquidTurbine__Work = 1.0; # Default
```

5.8 LiquidTurbine_odeso.ps *(-o -ss -LiquidTurbine__Heat)*

MTT command:

```
mtt -o -ss LiquidTurbine odeso ps 'LiquidTurbine__Heat'
```

This representation is given as Figure 5.8 (on page 123).

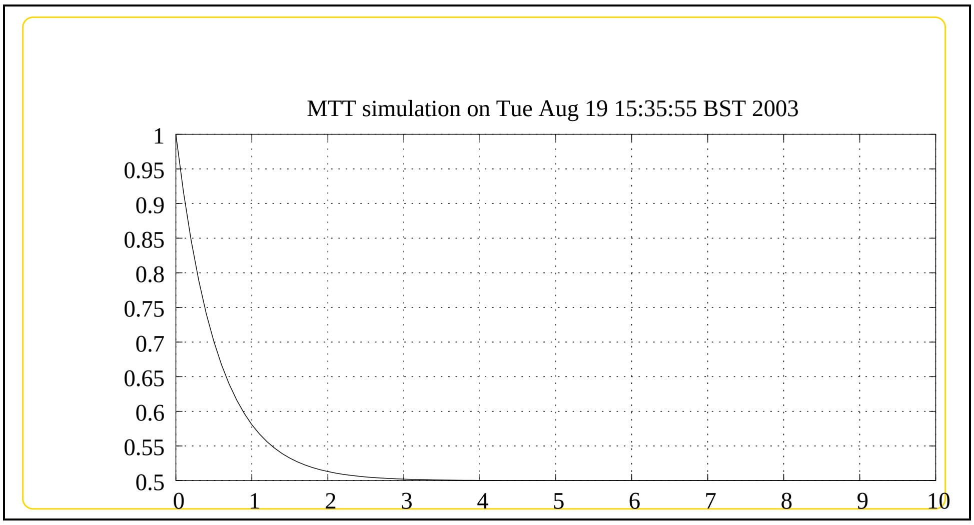


5.9 LiquidTurbine_odeso.ps *(-o -ss -LiquidTurbine__Work)*

MTT command:

```
mtt -o -ss LiquidTurbine odeso ps 'LiquidTurbine__Work'
```

This representation is given as Figure 5.9 (on page 124).



5.10 LiquidTurbine_odeso.ps *(-o -ss -LiquidTurbine__in__Hy_in)*

MTT command:

```
mtt -o -ss LiquidTurbine odeso ps 'LiquidTurbine__in__Hy_in'
```

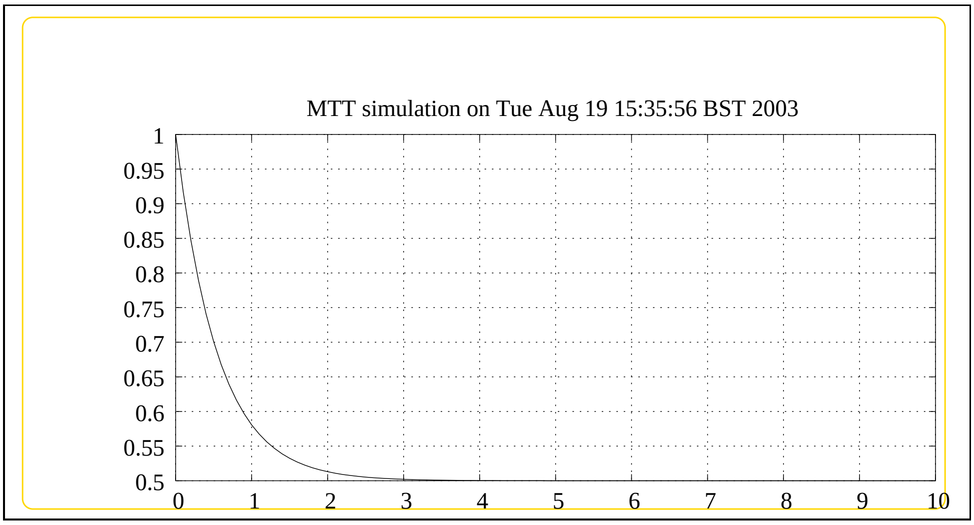
This representation is given as Figure 5.10 (on page 125).

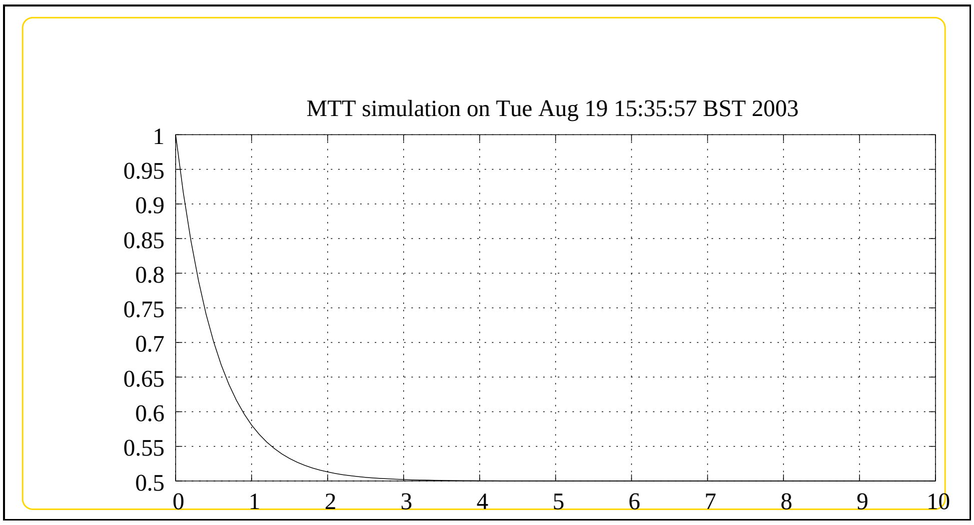
5.11 LiquidTurbine_odeso.ps *(-o -ss -LiquidTurbine__out__Hy_out)*

MTT command:

```
mtt -o -ss LiquidTurbine odeso ps 'LiquidTurbine__out__Hy_out'
```

This representation is given as Figure 5.11 (on page 125).


MTT simulation on Tue Aug 19 15:35:56 BST 2003
1
0.95
0.9
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
0
1
2
3
4
5
6
7
8
9
10

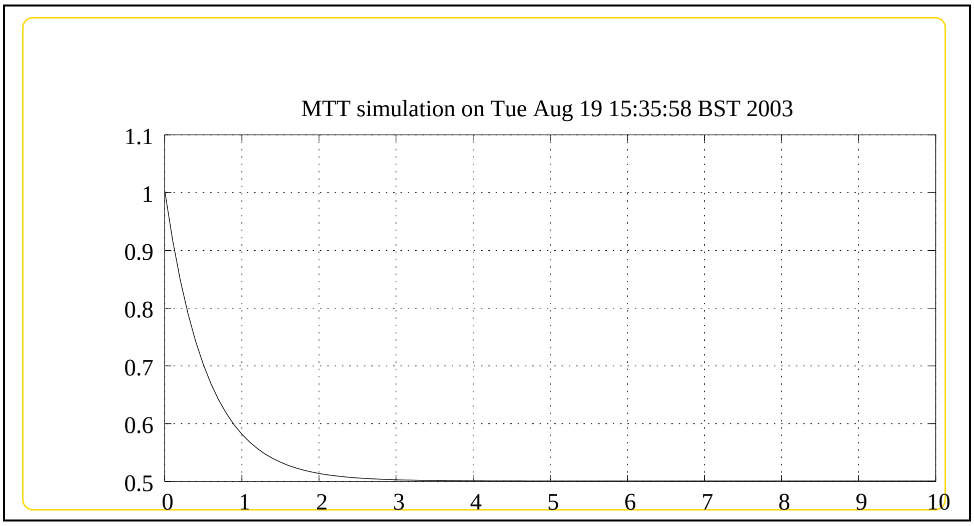

MTT simulation on Tue Aug 19 15:35:57 BST 2003
1
0.95
0.9
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
0
1
2
3
4
5
6
7
8
9
10

5.12 LiquidTurbine_odeso.ps *(-o -ss -LiquidTurbine__out__Th_out)*

MTT command:

```
mtt -o -ss LiquidTurbine odeso ps 'LiquidTurbine__out__Th_out'
```

This representation is given as Figure 5.12 (on page 126).



Chapter 6

ShowerHeater

6.1 ShowerHeater_abg.tex

MTT command:

```
mtt ShowerHeater abg tex
```

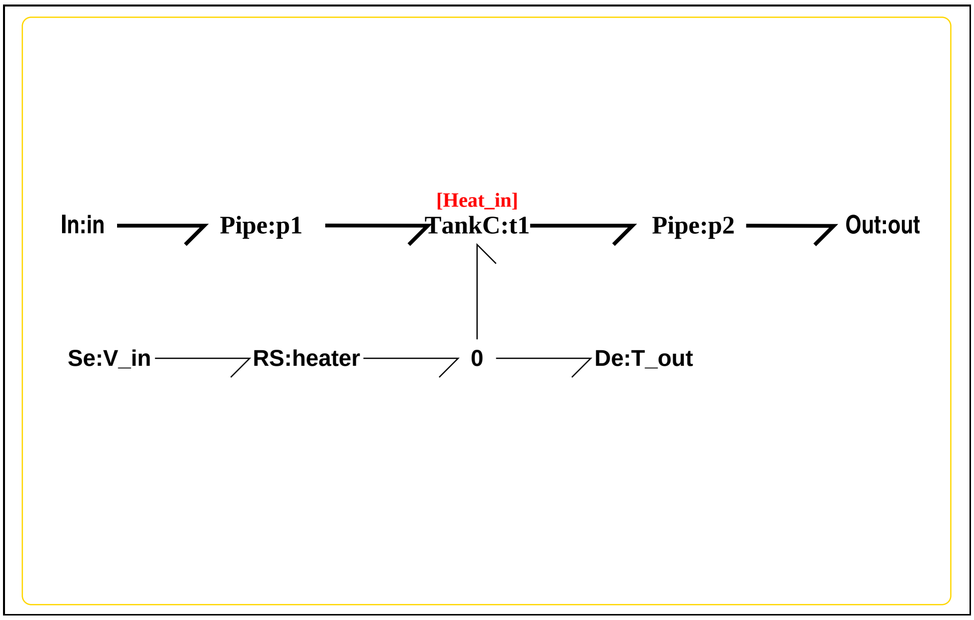


Figure 6.1: System **ShowerHeater**: acausal bond graph

The acausal bond graph of system **ShowerHeater** is displayed in Figure 6.1 (on page 127) and its label file is listed in Section 6.1.1 (on page 128). The subsystems are listed in Section 6.1.2 (on page 130).

ShowerHeater is a very elementary model of an electric heater suitable for a shower. It illustates the use of bond graph components which are internally pseudo, but externally true bond graphs (temperature/entropy flow).
There are three main components:

1. p1 and p2 – a **Pipe** component (see Section 6.1.6 (on page 136)). It is assumed that the pipes have zero flow resistance and thus do not generate heat via flow resistance.
2. t1 – a tank **Tank** component.
3. Heater – a resistive heater modelled by the thermodynamic **R** component **RS**.

Other components could be added to represent thermal conduction and thermal capacities.
The components **In** and **Out** provide the inlet and outlet conditions.
The three inputs are

u_1 The flow rate

u_2 The inlet temperature

u_3 The voltage across the heating element.

The single output is

y_1 The outflow temperature

and the state is

x_1 The heat contained in the tank.

6.1.1 Summary information

System ShowerHeater: ¡Detailed description here¿

Interface information:

Component Pipe is in library **IncompressibleFlow/Pipe**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: ShowerHeater_lbl.txt

```
#SUMMARY ShowerHeater
#DESCRIPTION <Detailed description here>
## Label file for system ShowerHeater (ShowerHeater_lbl.txt)

# ###############################################################
# ## Version control history
# ###############################################################
# ## $Id: ShowerHeater_lbl.txt,v 1.2 2003/08/06 18:52:37 gawthrop Exp $
# ## $Log: ShowerHeater_lbl.txt,v $
# ## Revision 1.2  2003/08/06 18:52:37  gawthrop
# ## Updated for latest MTT version.
# ##
# ## Revision 1.1  2000/12/28 18:11:47  peterg
# ## To RCS
# ##
# ###############################################################

#NOTPAR t_0

#ALIAS Pipe IncompressibleFlow/Pipe

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank


# Component type Pipe
p1 lin         rho;c_p;flow,0
p2 lin         rho;c_p;flow,0


# Component type TankC
t1 TankC rho;c_p;v
```

```
# Component type In
in

# Component type Out
out

# Component type RS
        heater         lin flow,r_h

# Component type Se
        V_in             SS                external

# Component type De
        T_out            SS                external
```

6.1.2 Subsystems

- De Simple effort detector (1) No subsystems.
- In: Inflow conditions (1) No subsystems.
- Out: Outflow conditions (1) No subsystems.
- Pipe: Pipe containing hot incompressible liquid (2) No subsystems.
- Se Simple effort source (1) No subsystems.
- TankC: TankC of hot incompressible liquid - fixed volume (1) No subsystems.

6.1.3 De

The acausal bond graph of system **De** is displayed in Figure 6.2 (on page 131) and its label file is listed in Section 6.1.3 (on page 130). The subsystems are listed in Section 6.1.3 (on page 132).

Summary information

System De:Simple effort detector Simple effort detector constructed from SS with fixed causality

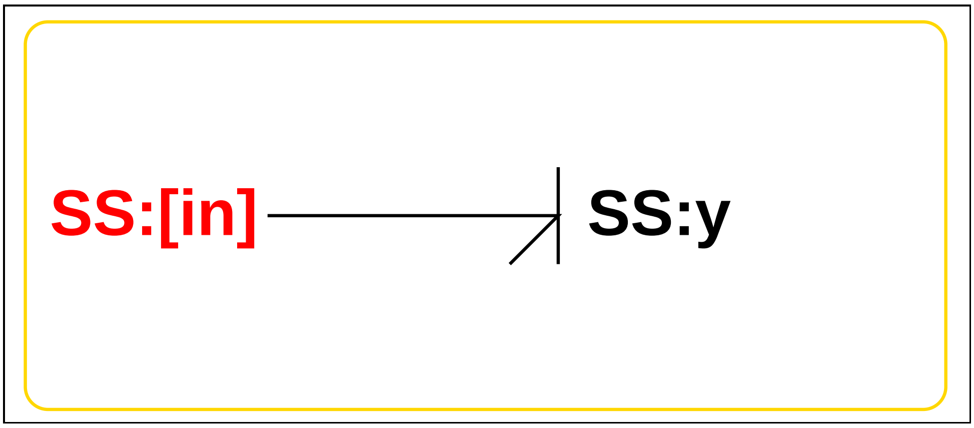


Figure 6.2: System **De**: acausal bond graph

Interface information:

Parameter $1 represents actual parameter **external**

Port in represents actual port **in**

Port out represents actual port **in**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: De_lbl.txt

```
%% Label file for system De (De_lbl.txt)
%SUMMARY De Simple effort detector
%DESCRIPTION Simple effort detector constructed from SS with fixed causa

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: De_lbl.txt,v 1.4 2002/11/07 04:28:23 gawthrop Exp $
% %% $Log: De_lbl.txt,v $
% %% Revision 1.4  2002/11/07 04:28:23  gawthrop
```

```
% %% Now has argument - either internal or external
% %%
% %% Revision 1.3  1999/09/07 03:32:21  peterg
% %% Fixed alias bug
% %%
% %% Revision 1.2  1999/09/07 03:21:02  peterg
% %% Aliased to out as well as in
% %%
% %% Revision 1.1  1999/03/03 22:02:04  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in|out in

% Argument aliases
%ALIAS $1 external

%% Each line should be of one of the following forms:
%      a comment (ie starting with %)
%      component-name cr_name arg1,arg2,..argn
%      blank

% ---- Component labels ----

% Component type SS
[in] SS external,external
y SS external,0
```

Subsystems

No subsystems.

6.1.4 In

The acausal bond graph of system **In** is displayed in Figure 6.3 (on page 133) and its label file is listed in Section 6.1.4 (on page 133). The subsystems are listed in Section 6.1.4 (on page 134).

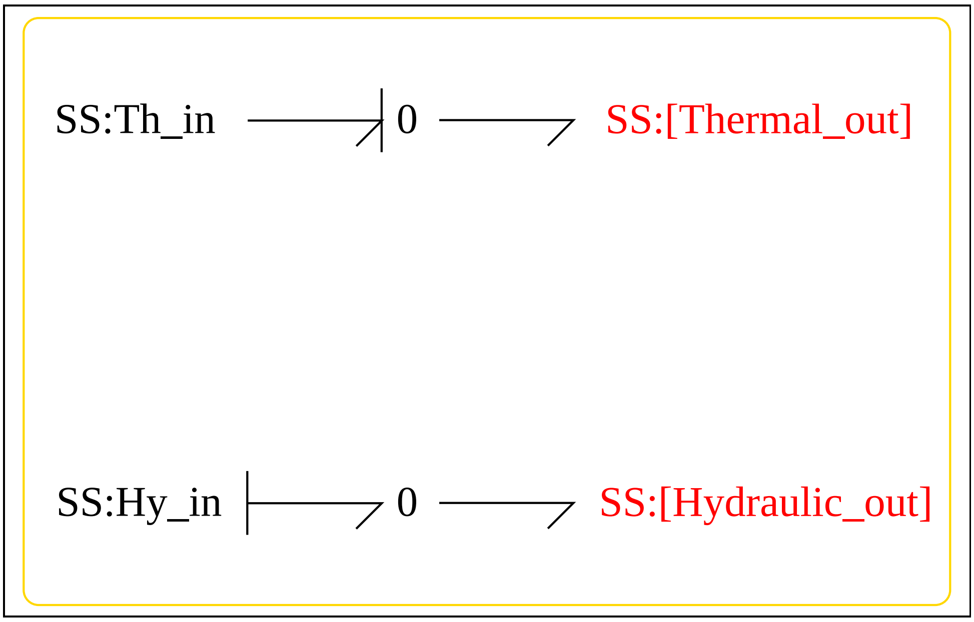


Figure 6.3: System **In**: acausal bond graph

Summary information

System In::Inflow conditions ¡Detailed description here¿

Interface information:

Port Hy_out represents actual port **Hydraulic_out**

Port Th_out represents actual port **Thermal_out**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: In_lbl.txt

```
%SUMMARY In: Inflow conditions
%DESCRIPTION <Detailed description here>
%% Label file for system In (In_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: In_lbl.txt,v 1.1 2000/12/28 18:11:47 peterg Exp $
% %% $Log: In_lbl.txt,v $
% %% Revision 1.1  2000/12/28 18:11:47  peterg
% %% To RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

%ALIAS out Thermal_out,Hydraulic_out
%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out



% Component type SS
Hy_in SS internal,external
Th_in SS external,internal
[Hydraulic_out] SS external,external
[Thermal_out] SS external,external
```

Subsystems

No subsystems.

6.1.5 Out

The acausal bond graph of system **Out** is displayed in Figure 6.4 (on page 135) and its label file is listed in Section 6.1.5 (on page 135). The subsystems are

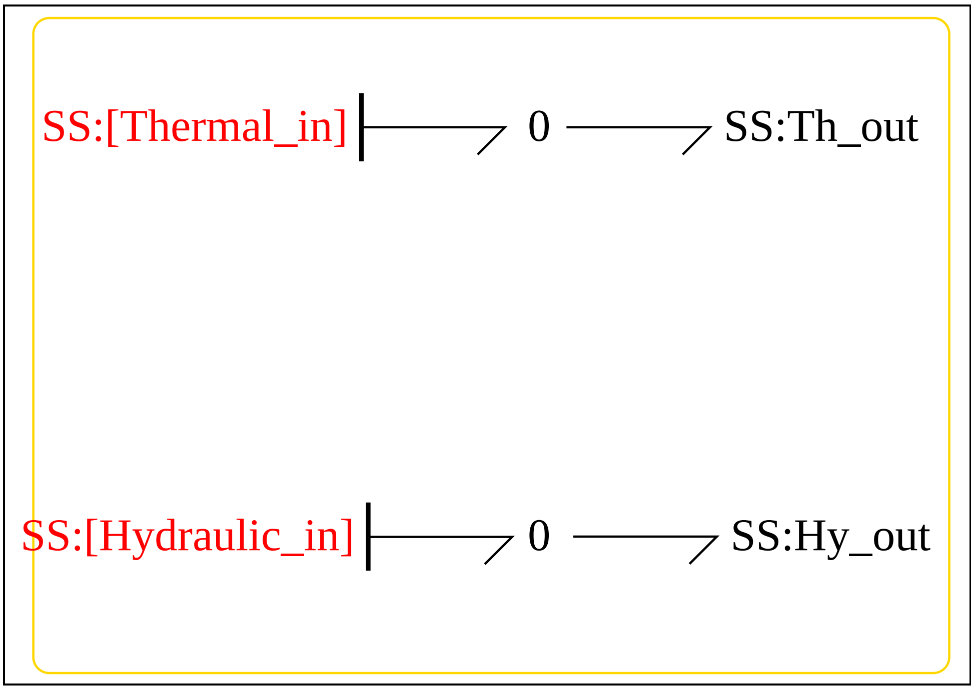


Figure 6.4: System **Out**: acausal bond graph

listed in Section 6.1.5 (on page 136).

Summary information

System Out::Outflow conditions ¡Detailed description here¿

Interface information:

Port Hy_in represents actual port **Hydraulic_in**

Port Th_in represents actual port **Thermal_in**

Port in represents actual port **Thermal_in,Hydraulic_in**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Out_lbl.txt

```
%SUMMARY Out: Outflow conditions
%DESCRIPTION <Detailed description here>
%% Label file for system Out (Out_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Out_lbl.txt,v 1.1 2000/12/28 18:11:47 peterg Exp $
% %% $Log: Out_lbl.txt,v $
% %% Revision 1.1  2000/12/28 18:11:47  peterg
% %% To RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in


% Component type SS
Hy_out SS        p_0,internal
Th_out SS        t_0,internal
[Hydraulic_in] SS external,external
[Thermal_in] SS external,external
```

Subsystems

No subsystems.

6.1.6 Pipe

The acausal bond graph of system **Pipe** is displayed in Figure 6.5 (on page 137) and its label file is listed in Section 6.1.6 (on page 137). The subsystems are listed in Section 6.1.6 (on page 139).

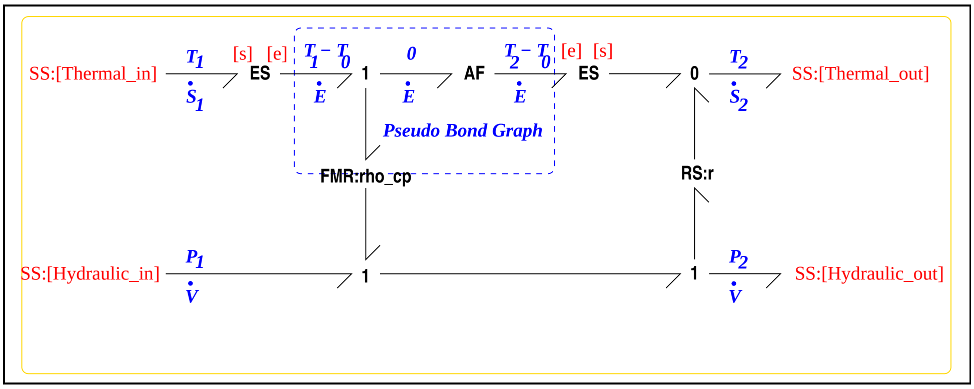


Figure 6.5: System **Pipe**: acausal bond graph

The **Pipe** component represents one way flow of incompressible fluid though a pipe. Externally, it has true energy bonds: $P/\dot{V}$ (Pressure/volume-flow) representing hydraulic energy and $T/\dot{S}$(Temperature/Entropy-flow) representing convected thermal energy.
Internally, however, the thermal part is represented by a pseudo bond graph which computes the flow of internal energy $\dot{E}$ from the upstream temperature T_1 and the volumetric flow rate $\dot{V}$ as:

$$\dot{E} = \rho c_p T_1 \dot{V} \tag{6.1}$$

The *AF* component makes the *FMR* component use T_1 rather than $T_1 - T_2$. The two **ES** components provide the conversion from true to psuedo thermal bonds and vice versa.
The pipe has an resistance to flow represented by the **RS** component labeled ‘r’ which can be linear or nonlinear. The hydraulic energy loss reappears on the thermal bond of this (energy-conserving) **RS** component.

Summary information

System Pipe::Pipe containing hot incompressible liquid

Interface information:

Parameter $1 represents actual parameter **rho**

Parameter $2 represents actual parameter **c_p**

Parameter $3 represents actual parameter **flow,r**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Pipe_lbl.txt

```
%% Label file for system Pipe (Pipe_lbl.txt)
%SUMMARY Pipe: Pipe containing hot incompressible liquid
%DESCRIPTION

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Pipe_lbl.txt,v 1.2 1998/11/20 11:35:38 peterg Exp $
% %% $Log: Pipe_lbl.txt,v $
% %% Revision 1.2  1998/11/20 11:35:38  peterg
% %% Removed redundant port label
% %%
% %% Revision 1.1  1998/11/20 11:34:17  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out
```

```
%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out


%ALIAS $1 rho
%ALIAS $2 c_p
%ALIAS $3 flow,r


% Argument aliases

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type FMR
rho_cp lin effort,$1*$2

% Component type RS
r $1 $3

% Component type SS
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
```

Subsystems

No subsystems.

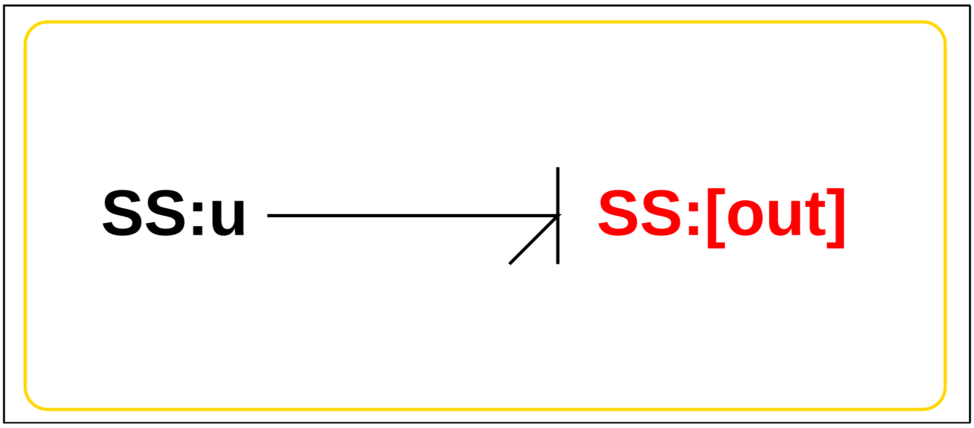


Figure 6.6: System **Se**: acausal bond graph

6.1.7 Se

The acausal bond graph of system **Se** is displayed in Figure 8.6 (on page 199) and its label file is listed in Section 8.1.7 (on page 200). The subsystems are listed in Section 8.1.7 (on page 201).

Summary information

System Se:Simple effort source Simple effort source constructed from SS with fixed causality

Interface information:

Parameter $1 represents actual parameter **e_s**

Port in represents actual port **out**

Port out represents actual port **out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Se_lbl.txt

```
%% Label file for system Se (Se_lbl.txt)
%SUMMARY Se Simple effort source
%DESCRIPTION Simple effort source constructed from SS with fixed causali


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Se_lbl.txt,v 1.3 1999/08/05 07:31:39 peterg Exp $
% %% $Log: Se_lbl.txt,v $
% %% Revision 1.3  1999/08/05 07:31:39  peterg
% %% Added in alias
% %%
% %% Revision 1.2  1999/03/12 04:04:27  peterg
% %% Single argument - the effort value e_s
% %%
% %% Revision 1.1  1999/03/03 21:55:46  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS out|in out

% Argument aliases
%ALIAS $1 e_s

%% Each line should be of one of the following forms:
%      a comment (ie starting with %)
%      component-name cr_name arg1,arg2,..argn
%      blank

% ---- Component labels ----

% Component type SS
[out] SS external,external
u SS e_s,internal
```

Subsystems

No subsystems.

6.1.8 TankC

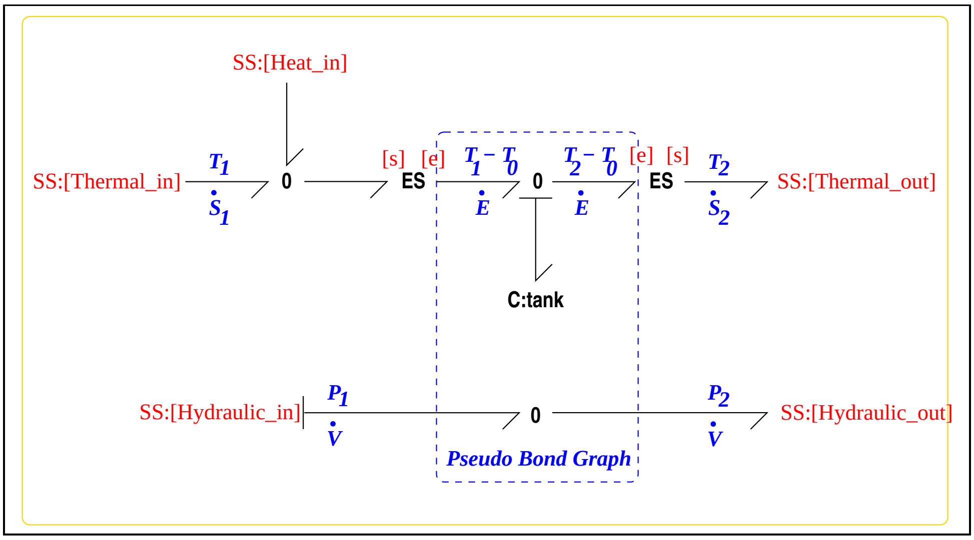


Figure 6.7: System **TankC**: acausal bond graph

The acausal bond graph of system **TankC** is displayed in Figure 6.7 (on page 142) and its label file is listed in Section 6.1.8 (on page 142). The subsystems are listed in Section 6.1.8 (on page 145).

Summary information

System TankC::TankC of hot incompressible liquid - fixed volume rho - density; c_p - specific heat; v - volume Typical lable: tank TankC rho;c_p;v

Interface information:

Parameter $1 represents actual parameter **rho**

Parameter $2 represents actual parameter **c_p**

Parameter $3 represents actual parameter **v**

Port Heat represents actual port **Heat_in**

Port Hy_in represents actual port **Hydraulic_in**

Port Hy_out represents actual port **Hydraulic_out**

Port Th_in represents actual port **Thermal_in**

Port Th_out represents actual port **Thermal_out**

Port in represents actual port **Thermal_in,Hydraulic_in**

Port out represents actual port **Thermal_out,Hydraulic_out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: TankC_lbl.txt

```
%% Label file for system TankC (TankC_lbl.txt)
%SUMMARY TankC: TankC of hot incompressible liquid - fixed volume
%DESCRIPTION rho - density; c_p - specific heat; v - volume
%DESCRIPTION Typical lable: tank  TankC  rho;c_p;v

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: TankC_lbl.txt,v 1.1 2000/12/28 18:11:47 peterg Exp $
% %% $Log: TankC_lbl.txt,v $
% %% Revision 1.1  2000/12/28 18:11:47  peterg
% %% To RCS
% %%
% %% Revision 1.3  1998/11/20 13:20:27  peterg
% %% Aliased ports
% %%
% %% Revision 1.2  1998/11/20 09:46:34  peterg
% %% Modernised lbl syntax
% %%
% %% Revision 1.1  1998/11/20 08:57:19  peterg
% %% Initial revision
```

```
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in  Thermal_in,Hydraulic_in
%ALIAS out Thermal_out,Hydraulic_out

%ALIAS Th_in  Thermal_in
%ALIAS Hy_in  Hydraulic_in

%ALIAS Th_out  Thermal_out
%ALIAS Hy_out  Hydraulic_out

%ALIAS Heat Heat_in

% Argument aliases
%ALIAS $1 rho
%ALIAS $2 c_p
%ALIAS $3 v

%% Each line should be of one of the following forms:
%      a comment (ie starting with %)
%      component-name cr_name arg1,arg2,..argn
%      blank

% ---- Component labels ----

% Component type C
tank TankC        rho,c_p,v

% Component type SS
[Heat_in] SS external,external
[Hydraulic_in] SS external,external
[Hydraulic_out] SS external,external
[Thermal_in] SS external,external
[Thermal_out] SS external,external
```

Subsystems

No subsystems.

6.2 ShowerHeater_struc.tex

MTT command:

```
mtt ShowerHeater struc tex
```

List of inputs for system ShowerHeater			
	Component	System	Repetition
1	Hy_in	ShowerHeater__in__Hy_in	1
2	Th_in	ShowerHeater__in__Th_in	1
3	u	ShowerHeater__V_in__u	1

List of outputs for system ShowerHeater			
	Component	System	Repetition
1	y	ShowerHeater__T_out__y	1

List of states for system ShowerHeater			
	Component	System	Repetition
1	tank	ShowerHeater__t1__tank	1

6.3 ShowerHeater_sympar.tex

MTT command:

```
mtt ShowerHeater sympar tex
```

Parameter	System
c_p	ShowerHeater
p_0	ShowerHeater__out
r_h	ShowerHeater
rho	ShowerHeater
t_0	ShowerHeater__out
v	ShowerHeater

Table 6.1: Parameters

6.4 ShowerHeater_ode.tex

MTT command:

```
mtt ShowerHeater ode tex
```

$$\dot{x}_1 = \frac{\left(c_p u_1 u_2 r_h \rho v - u_1 x_1 r_h + u_3^2 v\right)}{(r_h v)} \tag{6.2}$$

$$y_1 = \frac{x_1}{(c_p \rho v)} \tag{6.3}$$

6.5 ShowerHeater_numpar.txt

MTT command:

```
mtt ShowerHeater numpar txt
```

```
# -*-octave-*- Put Emacs into octave-mode
# Numerical parameter file (ShowerHeater_numpar.txt)
# Generated by MTT at Tue Dec 14 09:52:42 EST 1999

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: ShowerHeater_numpar.txt,v 1.2 2003/08/06 18:52:44 gawthr
# %% $Log: ShowerHeater_numpar.txt,v $
# %% Revision 1.2  2003/08/06 18:52:44  gawthrop
# %% Updated for latest MTT version.
# %%
```

```
# %% Revision 1.1  2000/12/28 18:11:47  peterg
# %% To RCS
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Parameters
c_p =  4.184*1e3; # Specific heat
p_0 =  0.0; # Ambient pressure
r_h =  10.0; # Heater resistance
rho =  1e3; # Density
v =  1e-3; # Volume (1 lit)
t_0 = 1.0; # Added by MTT on Wed Aug 06 19:05:01 BST 2003
```

6.6 ShowerHeater_input.txt

MTT command:

```
mtt ShowerHeater input txt
```

```
# -*-octave-*- Put Emacs into octave-mode
# Input specification (ShowerHeater_input.txt)
# Generated by MTT at Tue Dec 14 10:03:59 EST 1999
###############################################################
## Version control history
###############################################################
## $Id: ShowerHeater_input.txt,v 1.2 2003/08/06 18:51:56 gawthrop Exp $
## $Log: ShowerHeater_input.txt,v $
## Revision 1.2  2003/08/06 18:51:56  gawthrop
## Updated for latest MTT version.
##
## Revision 1.1  2000/12/28 18:11:47  peterg
## To RCS
##
###############################################################

# Set the inputs
showerheater__in__hy_in = 1e-4*(t>30); # Inflow
showerheater__in__th_in =  280; # In temperature
showerheater__v_in__u = 240*((t<60)&&(t>1)); # Input voltage
```

6.7 ShowerHeater_odeso.ps

MTT command:

```
mtt ShowerHeater odeso ps
```

This representation is given as Figure 6.8 (on page 148).

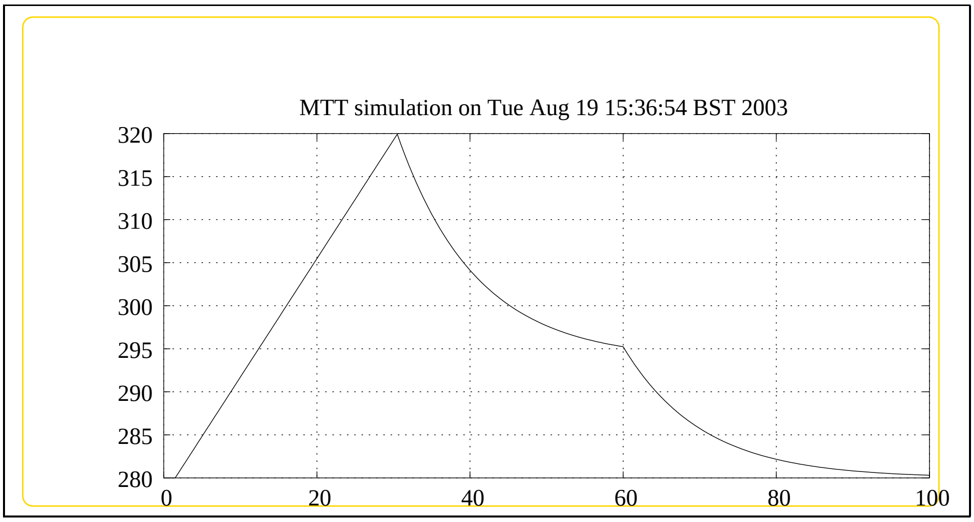


Figure 6.8: System **ShowerHeater**, representation odeso (-noargs)

Part IV

ThermalConduction

Chapter 7

HeatedRod

7.1 HeatedRod_abg.tex *(-o -ss)*

MTT command:

```
mtt -o -ss HeatedRod abg tex
```

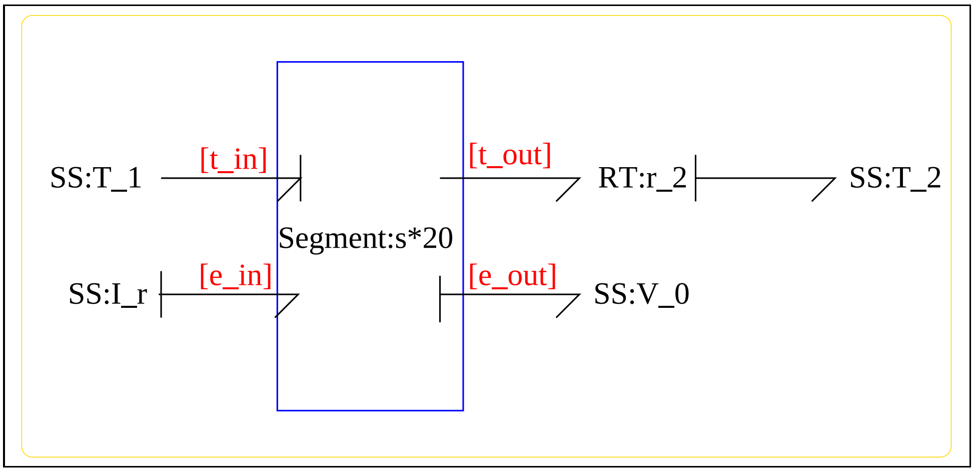


Figure 7.1: System **HeatedRod**: acausal bond graph

System **HeatedRod** is a model of a well-insulated rod of copper with an electric current passing through it which warms it up. The two ends of the rod are fixed at ambient temperature; this is where all the heat loss occurs.
This example introduces the idea of the **FP**, **RT** and **CT** components in the context of thermal conduction.

Parameter	Symbol	Value
Length	L_r	1m
Diameter	D_r	1mm
Resistivity	ρ	$1.68 \times 10^{-9}\Omega$m
Thermal conductivity	σ	390 Wm^{-1}
Thermal capacity	κ	380 Jm^{-3}

Table 7.1: Heated rod parameters

The model is similar to that described in chapter 8 of Cel91. However, instead of representing the thermal resistance by **RS** components and reinserting the entropy flow, the **RT** component uses two **FP** components to convert from true to pseudo bonds and back again. Similary, the thermal capacity is modelled by the **CT** component.

This distributed system (which strictly speaking has a partial differential equation model) is approximated by an ordinary diffferential equation model by modelling the system by a number of discrete segments of length Δx. Each segment model consists of two conceptual parts.

- An ideal lump of copper with no thermal resistance but with the normal attributes of electrical resistance (modelled by the **RS** component and thermal capacity (modelled by the **CF** component).

- A thin lump wtih thermal resistance but no thermal capacity or electrical resistance (modeled by the **RT** component).

At this level of the hierarchy, all bonds are true energy bonds and thus energy conservation is assured. Note that the **RS** component correctely transforms electrical to thermal energy.

The system was simulated with a total of nine lumps whilst passing a current of 1A though the rod for a total of 10s. The initial temperature and the end temperatures were all set at 300K.

7.1.1 Summary information

System HeatedRod::Thermal/Electrical model of Electric rod Introduces the idea of the ES component which transforms a relative-temperature/enthalpy pseudo bond (at the [e]port) into an absolute-temperature/enntropy energy bond (at the [s] port) and vice versa.

Interface information:

This component has no ALIAS declarations

Variable declarations:

area

delta_x

density

electrical_resistivity

mass

rod_length

rod_radius

segments

thermal_capacity

thermal_resistivity

volume

Units declarations:

This component has no UNITs declarations

The label file: HeatedRod_lbl.txt

```
%SUMMARY HeatedRod: Thermal/Electrical model of Electric rod
%DESCRIPTION Introduces the idea of the ES component which
%DESCRIPTION transforms a relative-temperature/enthalpy pseudo bond
%DESCRIPTION (at the [e]port) into an absolute-temperature/enntropy
%DESCRIPTION energy bond (at the [s] port) and vice versa.

%VAR rod_length
%VAR rod_radius
%VAR electrical_resistivity
%VAR thermal_resistivity
%VAR thermal_capacity
```

```
%VAR segments
%VAR area
%VAR delta_x
%VAR volume
%VAR density
%VAR mass

%% Label file for system HeatedRod (HeatedRod_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: HeatedRod_lbl.txt,v 1.1 2000/12/28 18:12:41 peterg Exp $
% %% $Log: HeatedRod_lbl.txt,v $
% %% Revision 1.1  2000/12/28 18:12:41  peterg
% %% To RCS
% %%
% %% Revision 1.1  1997/09/11 16:16:29  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank



% Component type RT
r_2 lin flow,r_2

% Component type SS
I_r SS internal,external
T_1 SS t_0,internal
T_2 SS t_0,internal
V_0 SS internal,internal

% Component type Segment
s
```

7.1.2 Subsystems

- RT: Two port thermal resistance with T/Sdot bonds (1) No subsystems.
- Segment: Segment of HeatedRod (1)
  - CT: One-port thermal C component with T/Sdot bond (1)
  - RT: Two port thermal resistance with T/Sdot bonds (1)

7.1.3 CT

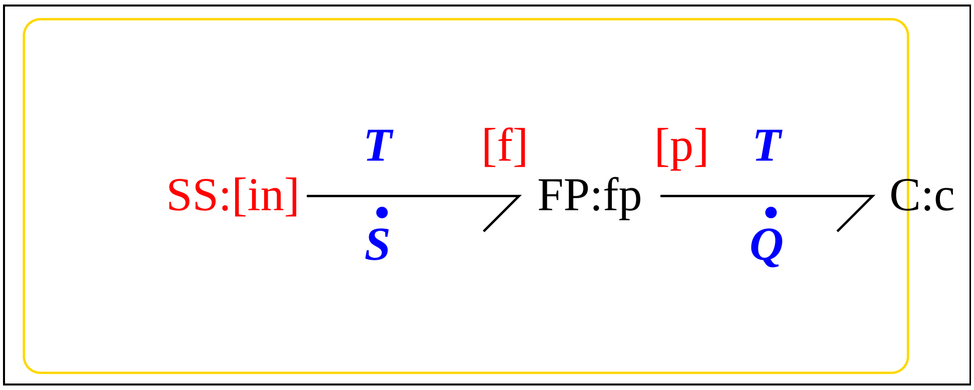


Figure 7.2: System **CT**: acausal bond graph

Component **CT** is a two port thermal resistor with true power bonds. Internally, it has a pseudo Bond Graph representation, and the corresponding thermal resistance just acts as an ordinary one-port **C** component.

Summary information

System CT::One-port thermal C component with T/Sdot bond CR and parameters as for a one-port C component Internally pseudo Example label file entry: c lin effort,c

Interface information:

Parameter $1 represents actual parameter **effort,c_t**

Parameter $a1 represents actual parameter **lin**

Port Thermal represents actual port **in**

Port out represents actual port **in**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: CT_lbl.txt

```
%SUMMARY CT: One-port thermal C component with T/Sdot bond
%DESCRIPTION CR and parameters as for a one-port C component
%DESCRIPTION Internally pseudo
%DESCRIPTION Example label file entry:
%DESCRIPTION % Component type CT
%DESCRIPTION  c lin effort,c

%Port aliases
%ALIAS Thermal|out in

%CR aliases
%ALIAS          $1 effort,c_t
%ALIAS          $a1 lin

%% Label file for system CT (CT_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: CT_lbl.txt,v 1.8 2001/07/05 08:42:43 gawthrop Exp $
% %% $Log: CT_lbl.txt,v $
% %% Revision 1.8  2001/07/05 08:42:43  gawthrop
% %% Updated to allow auto-generation of sensitivity version
% %%
% %% Revision 1.7  2001/07/03 22:59:10  gawthrop
% %% Fixed problems with argument passing for CRs
% %%
% %% Revision 1.6  2001/06/13 17:10:26  gawthrop
% %% Alias for the cr (ie ALIAS $1 lin)
% %%
% %% Revision 1.5  2001/06/11 15:09:18  gawthrop
% %% Removed spurious parameter
```

```
% %%
% %% Revision 1.4  1998/07/22 11:28:15  peterg
% %% Out as port alias
% %%
% %% Revision 1.3  1998/07/22 11:27:41  peterg
% %% Changed port name
% %%
% %% Revision 1.2  1998/06/29 10:12:58  peterg
% %% Converted to FP component
% %% Removed FP label
% %%
% %% Revision 1.1  1997/09/04  09:49:19  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank



% Component type C
c lin effort,c_t

% Component type FP
        fp

% Component type SS
[in] SS external,external
```

Subsystems

No subsystems.

7.1.4 RT

Component **RT** is a two port thermal resistor with true power bonds. Internally, it has a pseudo Bond Graph representation, and the corresponding thermal resistance just acts as an ordinary one-port **R** component.

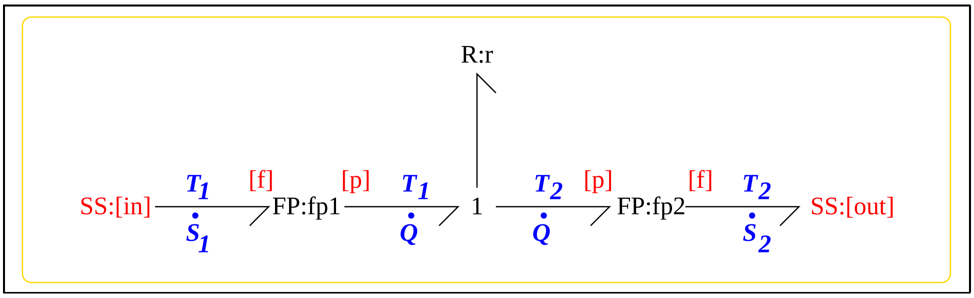


Figure 7.3: System **RT**: acausal bond graph

Summary information

System RT::Two port thermal resistance with T/Sdot bonds Port [in]: T/Sdot power in Port [out]: T/Sdot power out CR and parameters as for a one-port R component Internally pseudo bond graph Example label file entry: r lin flow,r

Interface information:

Parameter $1 represents actual parameter **flow,r**

Parameter $a1 represents actual parameter **lin**

Port ThermalIn represents actual port **in**

Port ThermalOut represents actual port **out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: RT_lbl.txt

```
%SUMMARY RT: Two port thermal resistance with T/Sdot bonds
%DESCRIPTION Port [in]: T/Sdot power in
%DESCRIPTION Port [out]: T/Sdot power out
%DESCRIPTION CR and parameters as for a one-port R component
```

```
%DESCRIPTION Internally pseudo bond graph
%DESCRIPTION Example label file entry:
%DESCRIPTION % Component type RT
%DESCRIPTION r lin flow,r

%ALIAS ThermalIn in
%ALIAS ThermalOut out

%ALIAS $1 flow,r
%ALIAS $a1 lin


%% Label file for system RT (RT_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: RT_lbl.txt,v 1.8 2001/07/05 08:42:41 gawthrop Exp $
% %% $Log: RT_lbl.txt,v $
% %% Revision 1.8  2001/07/05 08:42:41  gawthrop
% %% Updated to allow auto-generation of sensitivity version
% %%
% %% Revision 1.7  2001/07/03 22:59:10  gawthrop
% %% Fixed problems with argument passing for CRs
% %%
% %% Revision 1.6  2001/06/13 17:10:26  gawthrop
% %% Alias for the cr (ie ALIAS $1 lin)
% %%
% %% Revision 1.5  2001/06/11 19:51:08  gawthrop
% %% Zapped spurious $1 alias
% %%
% %% Revision 1.4  1998/07/22 11:31:42  peterg
% %% New port names
% %%
% %% Revision 1.3  1998/07/21 16:26:05  peterg
% %% Now has aliased parameters.
% %%
% %% Revision 1.2  1998/06/29 10:08:14  peterg
% %% Converted to FP component
% %% Removed lables from FP
% %%
```

```
% %% Revision 1.1  1997/09/04  09:48:47  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank


% Component type R
r lin flow,r

% Component type FP
        fp1
        fp2

% Component type SS
[in] SS external,external
[out] SS external,external
```

Subsystems

No subsystems.

7.1.5 Segment

The acausal bond graph of system **Segment** is displayed in Figure **??** and its label file is listed in Section 7.1.5. The subsystems are listed in Section 7.1.5.

Summary information

System Segment::Segment of HeatedRod Part of the HeatedRod example.

Interface information:

This component has no ALIAS declarations

Variable declarations:

This component has no PAR declarations

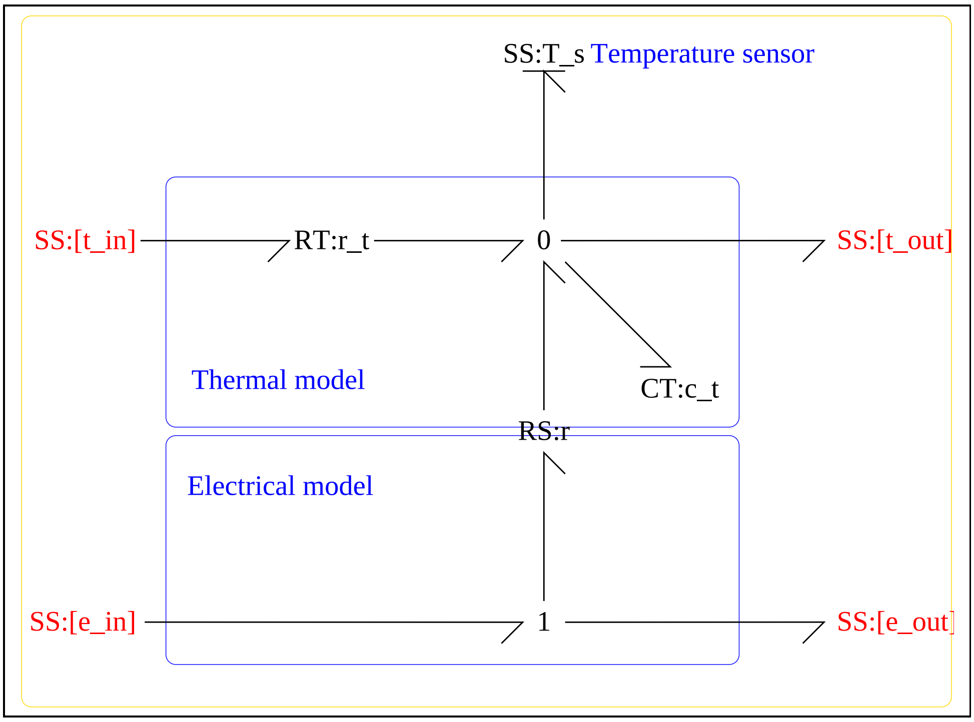


Figure 7.4: System **Segment**: acausal bond graph

Units declarations:

This component has no UNITs declarations

The label file: Segment_lbl.txt

```
%SUMMARY Segment: Segment of HeatedRod
%DESCRIPTION Part of the HeatedRod example.
%% Label file for system Segment (Segment_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Segment_lbl.txt,v 1.1 2000/12/28 18:12:41 peterg Exp $
% %% $Log: Segment_lbl.txt,v $
% %% Revision 1.1  2000/12/28 18:12:41  peterg
% %% To RCS
% %%
% %% Revision 1.2  1998/08/10 12:29:48  peterg
% %% Added missing ports.
% %%
% %% Revision 1.1  1997/09/11 16:17:14  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank



% Component type CT
c_t lin effort,c_t

% Component type RS
r lin flow,r

% Component type RT
r_t lin flow,r_t
```

```
% Component type SS
T_s SS external,0
[t_in] SS external,external
[t_out] SS external,external
[e_in] SS external,external
[e_out] SS external,external
```

Subsystems

- CT: One-port thermal C component with T/Sdot bond (1) No subsystems.
- RT: Two port thermal resistance with T/Sdot bonds (1) No subsystems.

7.2 HeatedRod_struc.tex *(-o -ss)*

MTT command:

```
mtt -o -ss HeatedRod struc tex
```

List of inputs for system HeatedRod			
	Component	System	Repetition
1	I_r	HeatedRod__I_r	1

List of outputs for system HeatedRod			
	Component	System	Repetition
1	T_s	HeatedRod__s__T_s	1
2	T_s	HeatedRod__s__2__T_s	2
3	T_s	HeatedRod__s__3__T_s	3
4	T_s	HeatedRod__s__4__T_s	4
5	T_s	HeatedRod__s__5__T_s	5
6	T_s	HeatedRod__s__6__T_s	6
7	T_s	HeatedRod__s__7__T_s	7
8	T_s	HeatedRod__s__8__T_s	8
9	T_s	HeatedRod__s__9__T_s	9
10	T_s	HeatedRod__s__10__T_s	10
11	T_s	HeatedRod__s__11__T_s	11
12	T_s	HeatedRod__s__12__T_s	12
13	T_s	HeatedRod__s__13__T_s	13
14	T_s	HeatedRod__s__14__T_s	14
15	T_s	HeatedRod__s__15__T_s	15
16	T_s	HeatedRod__s__16__T_s	16

List of outputs for system HeatedRod (continued)			
	Component	System	Repetition
17	T_s	HeatedRod__s__17__T_s	17
18	T_s	HeatedRod__s__18__T_s	18
19	T_s	HeatedRod__s__19__T_s	19
20	T_s	HeatedRod__s__20__T_s	20

List of states for system HeatedRod			
	Component	System	Repetition
1	c	HeatedRod__s__c_t__c	1
2	c	HeatedRod__s__2__c_t__c	1
3	c	HeatedRod__s__3__c_t__c	1
4	c	HeatedRod__s__4__c_t__c	1
5	c	HeatedRod__s__5__c_t__c	1
6	c	HeatedRod__s__6__c_t__c	1
7	c	HeatedRod__s__7__c_t__c	1
8	c	HeatedRod__s__8__c_t__c	1
9	c	HeatedRod__s__9__c_t__c	1
10	c	HeatedRod__s__10__c_t__c	1
11	c	HeatedRod__s__11__c_t__c	1
12	c	HeatedRod__s__12__c_t__c	1
13	c	HeatedRod__s__13__c_t__c	1
14	c	HeatedRod__s__14__c_t__c	1
15	c	HeatedRod__s__15__c_t__c	1
16	c	HeatedRod__s__16__c_t__c	1
17	c	HeatedRod__s__17__c_t__c	1
18	c	HeatedRod__s__18__c_t__c	1
19	c	HeatedRod__s__19__c_t__c	1
20	c	HeatedRod__s__20__c_t__c	1

7.3 HeatedRod_ode.tex *(-o -ss)*

MTT command:

```
mtt -o -ss HeatedRod ode tex
```

$$
\begin{aligned}
\dot{x}_1 &= \frac{\left(c_t u_1^2 r r_t + c_t t_0 - 2x_1 + x_2\right)}{(c_t r_t)}\\
\dot{x}_2 &= \frac{\left(c_t u_1^2 r r_t + x_1 - 2x_2 + x_3\right)}{(c_t r_t)}\\
\dot{x}_3 &= \frac{\left(c_t u_1^2 r r_t + x_2 - 2x_3 + x_4\right)}{(c_t r_t)}\\
\dot{x}_4 &= \frac{\left(c_t u_1^2 r r_t + x_3 - 2x_4 + x_5\right)}{(c_t r_t)}\\
\dot{x}_5 &= \frac{\left(c_t u_1^2 r r_t + x_4 - 2x_5 + x_6\right)}{(c_t r_t)}\\
\dot{x}_6 &= \frac{\left(c_t u_1^2 r r_t + x_5 - 2x_6 + x_7\right)}{(c_t r_t)}\\
\dot{x}_7 &= \frac{\left(c_t u_1^2 r r_t + x_6 - 2x_7 + x_8\right)}{(c_t r_t)}\\
\dot{x}_8 &= \frac{\left(c_t u_1^2 r r_t + x_7 - 2x_8 + x_9\right)}{(c_t r_t)}\\
\dot{x}_9 &= \frac{\left(c_t u_1^2 r r_t + x_{10} + x_8 - 2x_9\right)}{(c_t r_t)}\\
\dot{x}_{10} &= \frac{\left(c_t u_1^2 r r_t - 2x_{10} + x_{11} + x_9\right)}{(c_t r_t)}\\
\dot{x}_{11} &= \frac{\left(c_t u_1^2 r r_t + x_{10} - 2x_{11} + x_{12}\right)}{(c_t r_t)}\\
\dot{x}_{12} &= \frac{\left(c_t u_1^2 r r_t + x_{11} - 2x_{12} + x_{13}\right)}{(c_t r_t)}\\
\dot{x}_{13} &= \frac{\left(c_t u_1^2 r r_t + x_{12} - 2x_{13} + x_{14}\right)}{(c_t r_t)}\\
\dot{x}_{14} &= \frac{\left(c_t u_1^2 r r_t + x_{13} - 2x_{14} + x_{15}\right)}{(c_t r_t)}\\
\dot{x}_{15} &= \frac{\left(c_t u_1^2 r r_t + x_{14} - 2x_{15} + x_{16}\right)}{(c_t r_t)}\\
\dot{x}_{16} &= \frac{\left(c_t u_1^2 r r_t + x_{15} - 2x_{16} + x_{17}\right)}{(c_t r_t)}\\
\dot{x}_{17} &= \frac{\left(c_t u_1^2 r r_t + x_{16} - 2x_{17} + x_{18}\right)}{(c_t r_t)}\\
\dot{x}_{18} &= \frac{\left(c_t u_1^2 r r_t + x_{17} - 2x_{18} + x_{19}\right)}{(c_t r_t)}\\
\dot{x}_{19} &= \frac{\left(c_t u_1^2 r r_t + x_{18} - 2x_{19} + x_{20}\right)}{(c_t r_t)}\\
\dot{x}_{20} &= \frac{\left(c_t u_1^2 r r_2 r_t + c_t r_t t_0 + x_{19} r_2 - x_{20} r_2 - x_{20} r_t\right)}{(c_t r_2 r_t)}
\end{aligned}
\tag{7.1}
$$

$$
\begin{aligned}
y_1 &= \frac{x_1}{c_t}\\
y_2 &= \frac{x_2}{c_t}\\
y_3 &= \frac{x_3}{c_t}\\
y_4 &= \frac{x_4}{c_t}\\
y_5 &= \frac{x_5}{c_t}\\
y_6 &= \frac{x_6}{c_t}\\
y_7 &= \frac{x_7}{c_t}\\
y_8 &= \frac{x_8}{c_t}\\
y_9 &= \frac{x_9}{c_t}\\
y_{10} &= \frac{x_{10}}{c_t}\\
y_{11} &= \frac{x_{11}}{c_t}\\
y_{12} &= \frac{x_{12}}{c_t}\\
y_{13} &= \frac{x_{13}}{c_t}\\
y_{14} &= \frac{x_{14}}{c_t}\\
y_{15} &= \frac{x_{15}}{c_t}\\
y_{16} &= \frac{x_{16}}{c_t}\\
y_{17} &= \frac{x_{17}}{c_t}\\
y_{18} &= \frac{x_{18}}{c_t}\\
y_{19} &= \frac{x_{19}}{c_t}\\
y_{20} &= \frac{x_{20}}{c_t}
\end{aligned}
\tag{7.2}
$$

7.4 HeatedRod_ss.tex *(-o -ss)*

MTT command:

```
mtt -o -ss HeatedRod ss tex
```

$$x = \begin{pmatrix} c_t t_0 \\ c_t t_0 \\ c_t t_0 \\ c_t t_0 \\ c_t t_0 \\ c_t t_0 \\ c_t t_0 \\ c_t t_0 \\ c_t t_0 \\ x_{10} \\ x_{11} \\ x_{12} \\ x_{13} \\ x_{14} \\ x_{15} \\ x_{16} \\ x_{17} \\ x_{18} \\ x_{19} \\ x_{20} \end{pmatrix} \tag{7.3}$$

$$u = \begin{pmatrix} 1 \end{pmatrix} \tag{7.4}$$

$$y = \begin{pmatrix} t_0 \\ t_0 \\ t_0 \\ t_0 \\ t_0 \\ t_0 \\ t_0 \\ t_0 \\ t_0 \\ \frac{x_{10}}{c_t} \\ \frac{x_{11}}{c_t} \\ \frac{x_{12}}{c_t} \\ \frac{x_{13}}{c_t} \\ \frac{x_{14}}{c_t} \\ \frac{x_{15}}{c_t} \\ \frac{x_{16}}{c_t} \\ \frac{x_{17}}{c_t} \\ \frac{x_{18}}{c_t} \\ \frac{x_{19}}{c_t} \\ \frac{x_{20}}{c_t} \end{pmatrix} \quad (7.5)$$

$$\dot{x} = \begin{pmatrix} r \\ r \\ r \\ r \\ r \\ r \\ r \\ r \\ \frac{(c_t r r_t - c_t t_0 + x_{10})}{(c_t r_t)} \\ \frac{(c_t r r_t + c_t t_0 - 2x_{10} + x_{11})}{(c_t r_t)} \\ \frac{(c_t r r_t + x_{10} - 2x_{11} + x_{12})}{(c_t r_t)} \\ \frac{(c_t r r_t + x_{11} - 2x_{12} + x_{13})}{(c_t r_t)} \\ \frac{(c_t r r_t + x_{12} - 2x_{13} + x_{14})}{(c_t r_t)} \\ \frac{(c_t r r_t + x_{13} - 2x_{14} + x_{15})}{(c_t r_t)} \\ \frac{(c_t r r_t + x_{14} - 2x_{15} + x_{16})}{(c_t r_t)} \\ \frac{(c_t r r_t + x_{15} - 2x_{16} + x_{17})}{(c_t r_t)} \\ \frac{(c_t r r_t + x_{16} - 2x_{17} + x_{18})}{(c_t r_t)} \\ \frac{(c_t r r_t + x_{17} - 2x_{18} + x_{19})}{(c_t r_t)} \\ \frac{(c_t r r_t + x_{18} - 2x_{19} + x_{20})}{(c_t r_t)} \\ \frac{(c_t r r_2 r_t + c_t r_t t_0 + x_{19} r_2 - x_{20} r_2 - x_{20} r_t)}{(c_t r_2 r_t)} \end{pmatrix} \tag{7.6}$$

7.5 HeatedRod_sm.tex *(-o -ss)*

MTT command:

```
mtt -o -ss HeatedRod sm tex
```

$$A_{11} = \frac{(-2)}{(c_t r_t)} \tag{7.7}$$

$$A_{12} = \frac{1}{(c_t r_t)} \tag{7.8}$$

$$A_{21} = \frac{1}{(c_t r_t)} \tag{7.9}$$

$$A_{22} = \frac{(-2)}{(c_t r_t)} \tag{7.10}$$

$$A_{23} = \frac{1}{(c_t r_t)} \tag{7.11}$$

$$A_{32} = \frac{1}{(c_t r_t)} \tag{7.12}$$

$$A_{33} = \frac{(-2)}{(c_t r_t)} \tag{7.13}$$

$$A_{34} = \frac{1}{(c_t r_t)} \tag{7.14}$$

$$A_{43} = \frac{1}{(c_t r_t)} \tag{7.15}$$

$$A_{44} = \frac{(-2)}{(c_t r_t)} \tag{7.16}$$

$$A_{45} = \frac{1}{(c_t r_t)} \tag{7.17}$$

$$A_{54} = \frac{1}{(c_t r_t)} \tag{7.18}$$

$$A_{55} = \frac{(-2)}{(c_t r_t)} \tag{7.19}$$

$$A_{56} = \frac{1}{(c_t r_t)} \tag{7.20}$$

$$A_{65} = \frac{1}{(c_t r_t)} \tag{7.21}$$

$$A_{66} = \frac{(-2)}{(c_t r_t)} \tag{7.22}$$

$$A_{67} = \frac{1}{(c_t r_t)} \tag{7.23}$$

$$A_{76} = \frac{1}{(c_t r_t)} \tag{7.24}$$

$$A_{77} = \frac{(-2)}{(c_t r_t)} \tag{7.25}$$

$$A_{78} = \frac{1}{(c_t r_t)} \tag{7.26}$$

$$A_{87} = \frac{1}{(c_t r_t)} \tag{7.27}$$

$$A_{88} = \frac{(-2)}{(c_t r_t)} \tag{7.28}$$

$$A_{89} = \frac{1}{(c_t r_t)} \tag{7.29}$$

$$A_{98} = \frac{1}{(c_t r_t)} \tag{7.30}$$

$$A_{99} = \frac{(-2)}{(c_t r_t)} \tag{7.31}$$

$$A_{910} = \frac{1}{(c_t r_t)} \tag{7.32}$$

$$A_{109} = \frac{1}{(c_t r_t)} \tag{7.33}$$

$$A_{1010} = \frac{(-2)}{(c_t r_t)} \tag{7.34}$$

$$A_{1011} = \frac{1}{(c_t r_t)} \tag{7.35}$$

$$A_{1110} = \frac{1}{(c_t r_t)} \tag{7.36}$$

$$A_{1111} = \frac{(-2)}{(c_t r_t)} \tag{7.37}$$

$$A_{1112} = \frac{1}{(c_t r_t)} \tag{7.38}$$

$$A_{1211} = \frac{1}{(c_t r_t)} \tag{7.39}$$

$$A_{1212} = \frac{(-2)}{(c_t r_t)} \tag{7.40}$$

$$A_{1213} = \frac{1}{(c_t r_t)} \tag{7.41}$$

$$A_{1312} = \frac{1}{(c_t r_t)} \tag{7.42}$$

$$A_{1313} = \frac{(-2)}{(c_t r_t)} \tag{7.43}$$

$$A_{1314} = \frac{1}{(c_t r_t)} \tag{7.44}$$

$$A_{1413} = \frac{1}{(c_t r_t)} \tag{7.45}$$

$$A_{1414} = \frac{(-2)}{(c_t r_t)} \tag{7.46}$$

$$A_{1415} = \frac{1}{(c_t r_t)} \tag{7.47}$$

$$A_{1514} = \frac{1}{(c_t r_t)} \tag{7.48}$$

$$A_{1515} = \frac{(-2)}{(c_t r_t)} \tag{7.49}$$

$$A_{1516} = \frac{1}{(c_t r_t)} \tag{7.50}$$

$$A_{1615} = \frac{1}{(c_t r_t)} \tag{7.51}$$

$$A_{1616} = \frac{(-2)}{(c_t r_t)} \tag{7.52}$$

$$A_{1617} = \frac{1}{(c_t r_t)} \tag{7.53}$$

$$A_{1716} = \frac{1}{(c_t r_t)} \tag{7.54}$$

$$A_{1717} = \frac{(-2)}{(c_t r_t)} \tag{7.55}$$

$$A_{1718} = \frac{1}{(c_t r_t)} \tag{7.56}$$

$$A_{1817} = \frac{1}{(c_t r_t)} \tag{7.57}$$

$$A_{1818} = \frac{(-2)}{(c_t r_t)} \tag{7.58}$$

$$A_{1819} = \frac{1}{(c_t r_t)} \tag{7.59}$$

$$A_{1918} = \frac{1}{(c_t r_t)} \tag{7.60}$$

$$A_{1919} = \frac{(-2)}{(c_t r_t)} \tag{7.61}$$

$$A_{1920} = \frac{1}{(c_t r_t)} \tag{7.62}$$

$$A_{2019} = \frac{1}{(c_t r_t)} \tag{7.63}$$

$$A_{2020} = \frac{(-(r_2 + r_t))}{(c_t r_2 r_t)} \tag{7.64}$$

$$B = \begin{pmatrix} 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \\ 2r \end{pmatrix} \tag{7.65}$$

$$C_{11} = \frac{1}{c_t} \tag{7.66}$$

$$C_{22} = \frac{1}{c_t} \tag{7.67}$$

$$C_{33} = \frac{1}{c_t} \tag{7.68}$$

$$C_{44} = \frac{1}{c_t} \tag{7.69}$$

$$C_{55} = \frac{1}{c_t} \tag{7.70}$$

$$C_{66} = \frac{1}{c_t} \tag{7.71}$$

$$C_{77} = \frac{1}{c_t} \tag{7.72}$$

$$C_{88} = \frac{1}{c_t} \tag{7.73}$$

$$C_{99} = \frac{1}{c_t} \tag{7.74}$$

$$C_{1010} = \frac{1}{c_t} \tag{7.75}$$

$$C_{1111} = \frac{1}{c_t} \tag{7.76}$$

$$C_{1212} = \frac{1}{c_t} \tag{7.77}$$

$$C_{1313} = \frac{1}{c_t} \tag{7.78}$$

$$C_{1414} = \frac{1}{c_t} \tag{7.79}$$

$$C_{1515} = \frac{1}{c_t} \tag{7.80}$$

$$C_{1616} = \frac{1}{c_t} \quad (7.81)$$

$$C_{1717} = \frac{1}{c_t} \quad (7.82)$$

$$C_{1818} = \frac{1}{c_t} \quad (7.83)$$

$$C_{1919} = \frac{1}{c_t} \quad (7.84)$$

$$C_{2020} = \frac{1}{c_t} \quad (7.85)$$

$$D = \begin{pmatrix} 0 \end{pmatrix} \quad (7.86)$$

7.6 HeatedRod_lmfr.ps *(-o -ss)*

MTT command:

```
mtt -o -ss HeatedRod lmfr ps
```

This representation is given as Figure 7.5 (on page 179).

7.7 HeatedRod_numpar.txt *(-o -ss)*

MTT command:

```
mtt -o -ss HeatedRod numpar txt
```

```
# Numerical parameter file (HeatedRod_numpar.txt)
# Generated by MTT at Thu Sep  4 16:11:04 BST 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

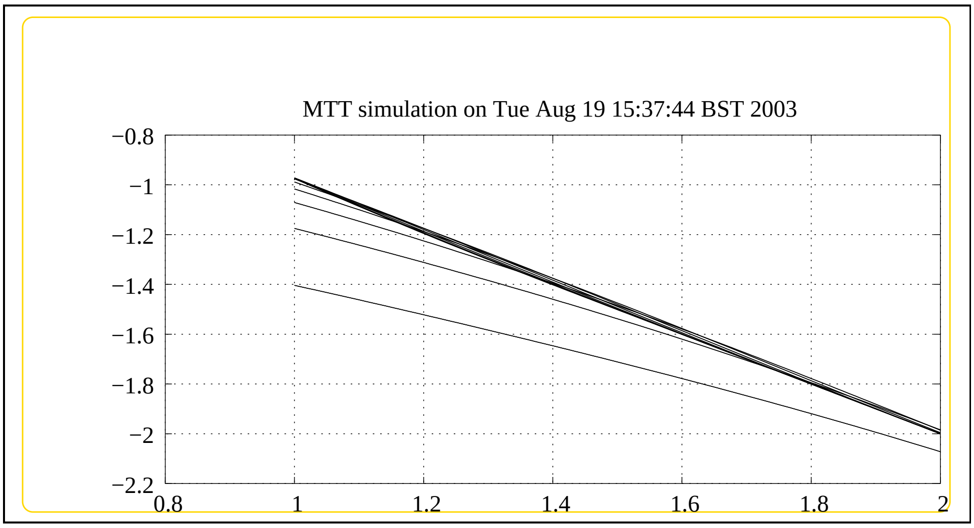


Figure 7.5: System **HeatedRod**, representation lmfr (-o-ss)

```
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: HeatedRod_numpar.txt,v 1.1 2000/12/28 18:12:41 peterg Exp $
# %% $Log: HeatedRod_numpar.txt,v $
# %% Revision 1.1  2000/12/28 18:12:41  peterg
# %% To RCS
# %%
# %% Revision 1.1  1998/08/15 13:40:33  peterg
# %% Initial revision
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


# Constants for copper
density = 8.96;
rod_length = 1.0;
rod_radius = 1e-3;
electrical_resistivity = 16.8*0.000000001;
thermal_resistivity = 1/390.0;
thermal_capacity = 380.0;


segments = 20;
```

```
area = pi*rod_radius*rod_radius;
delta_x = rod_length/segments;
volume = area*delta_x;
mass = volume*density;

# Parameters
c_t =  thermal_capacity*mass;
r =  electrical_resistivity*delta_x/area;
r_t =  thermal_resistivity*delta_x/area;
r_2 =  r_t;
t_0 =  300; # Ambient
```

7.8 HeatedRod_input.txt *(-o -ss)*

MTT command:

```
mtt -o -ss HeatedRod input txt
```

```
# Numerical parameter file (HeatedRod_input.txt)
# Generated by MTT at Thu Sep  4 16:11:06 BST 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: HeatedRod_input.txt,v 1.2 2003/08/04 07:39:32 gawthrop E
# %% $Log: HeatedRod_input.txt,v $
# %% Revision 1.2  2003/08/04 07:39:32  gawthrop
# %% Updated for current MTT
# %%
# %% Revision 1.1  2000/12/28 18:12:41  peterg
# %% To RCS
# %%
# %% Revision 1.1  1998/08/15 13:40:20  peterg
# %% Initial revision
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
# Set the inputs
## Removed by MTT on Thu Jun 12 14:33:52 BST 2003: u() =
10.0*(t<5.0); # I_r
heatedrod__i_r = 10.0*(t<5.0); # I_r
```

7.9 HeatedRod_odeso.ps *(-o -ss)*

MTT command:

```
mtt -o -ss HeatedRod odeso ps
```

This representation is given as Figure 7.6 (on page 181).

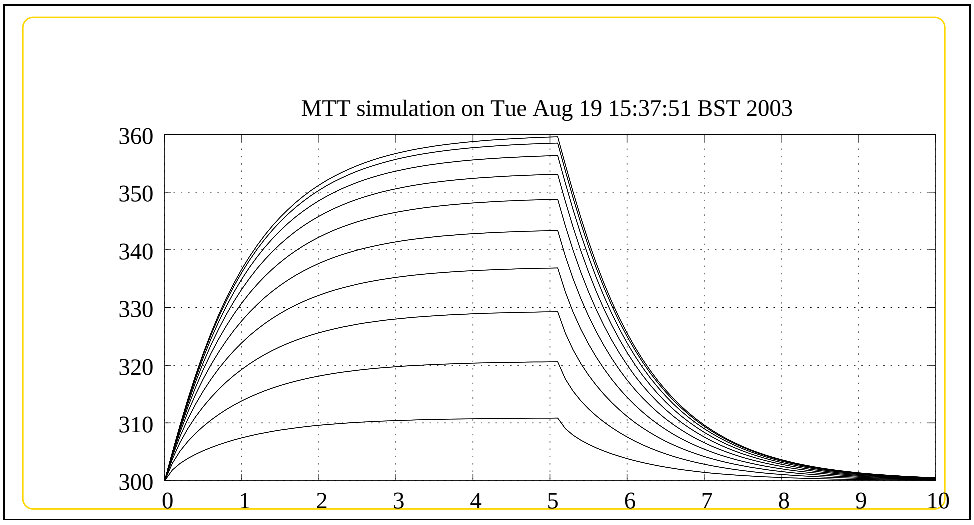


Figure 7.6: System **HeatedRod**, representation odeso (-o-ss)

Part V

ThermodynamicCycles

Chapter 8

CarnotCycle

8.1 CarnotCycle_abg.tex *(-ss -o)*

MTT command:

```
mtt -ss -o CarnotCycle abg tex
```

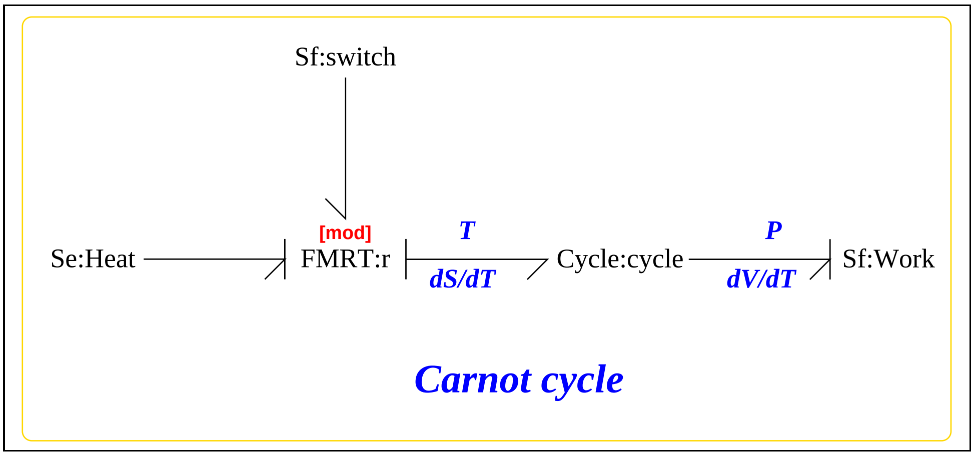


Figure 8.1: System **CarnotCycle**: acausal bond graph

The acausal bond graph of system **CarnotCycle** is displayed in Figure 8.1 (on page 185) and its label file is listed in Section 8.1.1 (on page 186). The subsystems are listed in Section 8.1.2 (on page 189).

The Carnot cycle is a simple closed thermodynamic cycle with four parts:

1. Isentropic compression

2. Heat injection at constant temperature
3. Isentropic expansion
4. Heat extraction at constant temperature

The subsystem **Cycle** (Section 13.1.4 (on page 295)) is a two-port component describing an ideal gas. It has two energy ports which, with integral causality correspond to

1. Entropy flow in; temperature out
2. Volume rate of change in; pressure out

In contast to the Otto cycle (see Table 13.1 (on page 296) where each table entry gives the causality on the heat and work ports respectively). The ideal Carnot cycle has derivative causality on the **[Heat]** port for two parts of the cycle.
To avoid this causlity change, the Carnot cycle is approximated by applying the heat from a temperature source via a thermal resistance **RT** component. During the *heat injection* and *heat extraction* parts of the cycle, the resistance parameter $r \approx 0$, but during the *isentropic compression* and *isentropic expansion* parts of the cycle, the resistance parameter $r \approx$ inf.
The simulation parameters appear in Section **??** (on page **??**). The results are plotted against time as follows:

- Volume (Figure **??** (on page **??**))
- Pressure (Figure **??** (on page **??**))
- Entropy (Figure **??** (on page **??**))
- Temperature (Figure **??** (on page **??**))

These values are replotted as the standard PV and TS diagrams in Figures **??** (on page **??**) and **??** (on page **??**) respectively.
The PV diagram shows the long and thin form typical of the Carnot cycle – this implies a poor work ratio. The TS diagram is not informative; it is not the expected rectangle because both T and S change in a stepwise manner.

8.1.1 Summary information

System CarnotCycle::a simple closed thermodynamic cycle The Carnot cycle is a simple closed thermodynamic cycle with four parts: o Isentropic compression o Heat injection at constant temperature o Isentropic expansion o Heat extraction at constant temperature

Interface information:

This component has no ALIAS declarations

Variable declarations:

P_0

T_0

TopTemp

V_0

alpha

Units declarations:

This component has no UNITs declarations

The label file: CarnotCycle_lbl.txt

```
#SUMMARY CarnotCycle: a simple closed thermodynamic cycle
#DESCRIPTION The Carnot cycle is a simple closed thermodynamic cycle
#DESCRIPTION with four parts:
#DESCRIPTION o Isentropic compression
#DESCRIPTION o Heat injection at constant temperature
#DESCRIPTION o Isentropic expansion
#DESCRIPTION o Heat extraction at constant temperature

#PAR P_0
#PAR T_0
#PAR V_0
#PAR alpha
#PAR TopTemp


#NOTPAR ideal_gas

## Label file for system CarnotCycle (CarnotCycle_lbl.txt)

# ###############################################################
# ## Version control history
```

```
# ###############################################################
# ## $Id: CarnotCycle_lbl.txt,v 1.4 2003/08/14 20:17:25 gawthrop E
# ## $Log: CarnotCycle_lbl.txt,v $
# ## Revision 1.4  2003/08/14 20:17:25  gawthrop
# ## Tidy up minor bugs
# ##
# ## Revision 1.3  2003/08/13 17:03:00  gawthrop
# ## Updated for new MTT
# ## Use FMR in place of R
# ## Fixed bug in sspar
# ##
# ## Revision 1.2  1998/08/10 16:40:07  peterg
# ## Added VARs and parametrs
# ##
# ## Revision 1.1  1998/07/21 15:18:18  peterg
# ## Initial revision
# ##
# ###############################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank



# Component type Cycle
cycle none ideal_gas;c_v;gamma_g;m_g

# Component type RT
r lin flow,1

# Component type Se
Heat SS external

# Component type Sf
Work    SS external
switch  SS external
```

8.1.2 Subsystems

- (1)
  - CU (1)
  - INTF: flow integrator (2)
- (1) No subsystems.
- Se Simple effort source (1) No subsystems.
- Sf Simple flow source (2) No subsystems.

8.1.3 CU

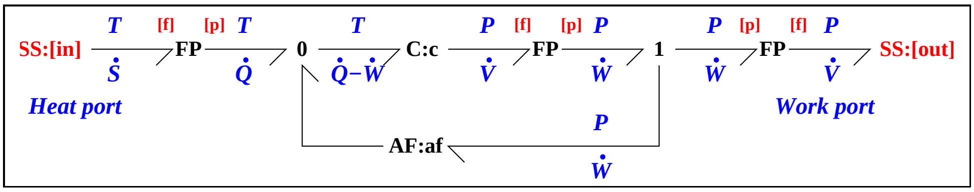


Figure 8.2: System **CU**: acausal bond graph

The acausal bond graph of system **CU** is displayed in Figure 13.2 (on page 294) and its label file is listed in Section 13.1.3 (on page 294). The subsystems are listed in Section 13.1.3 (on page 295).

Summary information

System CU: ¡Detailed description here¿

Interface information:

This component has no ALIAS declarations

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: CU_lbl.txt

```
%SUMMARY CU
%DESCRIPTION <Detailed description here>
%% Label file for system CU (CU_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: CU_lbl.txt,v 1.1 2000/12/28 10:34:56 peterg Exp $
% %% $Log: CU_lbl.txt,v $
% %% Revision 1.1  2000/12/28 10:34:56  peterg
% %% Put under RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

% Component type AF (gain of -1)
af lin -1

% Component type C
c CU $1

% Component type SS
[in] SS external,external
[out] SS external,external
```

Subsystems

No subsystems.

8.1.4 Cycle

The acausal bond graph of system **Cycle** is displayed in Figure 13.3 (on page 295) and its label file is listed in Section 13.1.4 (on page 296). The subsystems are listed in Section 13.1.4 (on page 299).

The system has two heat engine ports:

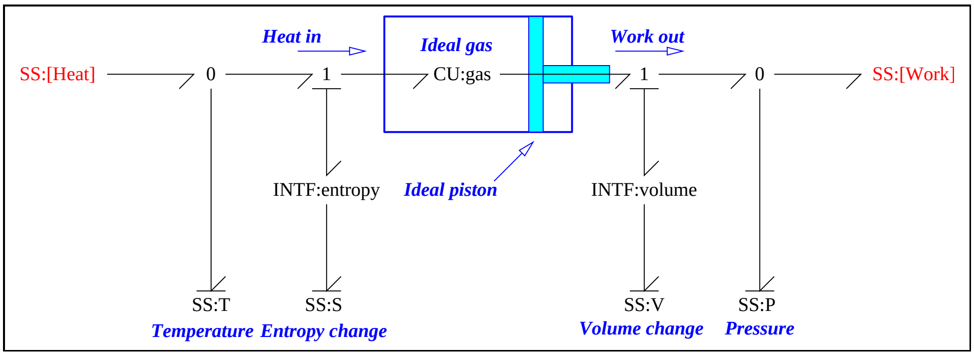


Figure 8.3: System **Cycle**: acausal bond graph

1. **[Heat]** and
2. **[Work]**

By convention, energy flows in to the **[Heat]** port and out of the **[Work]** port. Both ports are true energy ports.

The subsystem **CU** (Section 13.1.3 (on page 293)) is a two-port component describing an ideal gas. It has two energy ports which, with integral causality correspond to

1. Entropy flow in; temperature out
2. Volume rate of change in; pressure out

Four **SS** components are set up as sensors to measure the important quantities pertaining to the cycle:

1. **S** to measure the integrated entropy flow
2. **T** to measure the (absolute) temperature
3. **V** to measure the integrated volume change
4. **P** to measure the pressure

A number of cycles can be built depending on the causality of the two ports **[Heat]** and **[Work]** of **Cycle** Some possible cycles listed in Tables 13.1 (on page 296) where each table entry gives the causality on the heat and work ports respectively.

Cycle	Compression	Heating	Expansion	Cooling
Otto	II	II	II	II
Carnot	II	DI	II	DI
Diesel	II	ID	II	II
Joule	II	ID	II	ID

Table 8.1: Cycles and their causality

Summary information

System Cycle::Closed cycle with ideal gas Uses the CU two-port thermal capacitor.

Interface information:

Parameter $1 represents actual parameter **ideal_gas**

Parameter $2 represents actual parameter **c_v**

Parameter $3 represents actual parameter **gamma_g**

Parameter $4 represents actual parameter **m_g**

Port in represents actual port **Heat**

Port out represents actual port **Work**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Cycle_lbl.txt

```
#SUMMARY Cycle: Closed cycle with ideal gas
#DESCRIPTION Uses the CU two-port thermal capacitor.

#ALIAS in Heat
#ALIAS out Work
```

```
#ALIAS $1 ideal_gas
#ALIAS $2 c_v
#ALIAS $3 gamma_g
#ALIAS $4 m_g

## Label file for system Cycle (Cycle_lbl.txt)

# ###############################################################
# ## Version control history
# ###############################################################
# ## $Id: Cycle_lbl.txt,v 1.3 2000/12/27 16:38:28 peterg Exp $
# ## $Log: Cycle_lbl.txt,v $
# ## Revision 1.3  2000/12/27 16:38:28  peterg
# ## *** empty log message ***
# ##
# ## Revision 1.2  1998/07/21 14:21:04  peterg
# ## New style file
# ##
# ## Revision 1.1  1997/12/08 20:24:43  peterg
# ## Initial revision
# ##
# ## Revision 1.1  1997/12/07 20:38:05  peterg
# ## Initial revision
# ##
# ###############################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank




# Component type CU
gas CU ideal_gas,c_v,gamma_g,m_g

# Component type INTF
entropy
volume
```

```
# Component type SS
S SS external,0
T SS external,0
V SS external,0
P SS external,0
[Heat] SS external,external
[Work] SS external,external
```

Subsystems

- CU (1) No subsystems.
- INTF: flow integrator (2) No subsystems.

8.1.5 FMRT

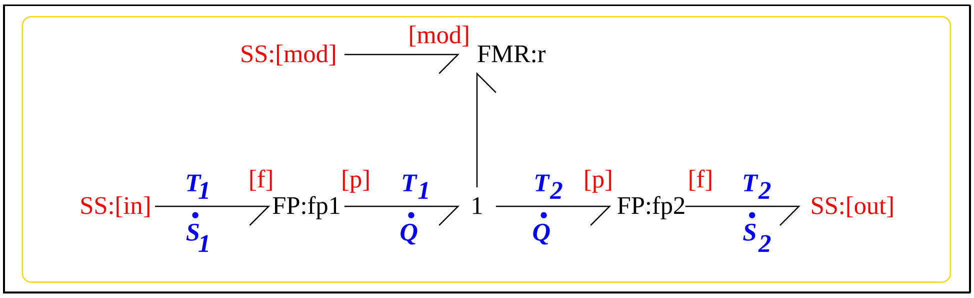


Figure 8.4: System **FMRT**: acausal bond graph

Component **FMRT** is a two port thermal resistor with true power bonds. Internally, it has a pseudo Bond Graph representation, and the corresponding thermal resistance just acts as an ordinary one-port **R** component.
The resistance r is modulated by a flow variable f on the [mod] port so that the *conductance* σ is:

$$\sigma = \frac{f}{r} \tag{8.1}$$

Summary information

System FMRT::Two port thermal resistance with T/Sdot bonds with flow modulation Port [in]: T/Sdot power in Port [out]: T/Sdot power out CR and parameters as for a one-port R component Internally pseudo bond graph Example label file entry: Component type RT r lin flow,r

Interface information:

Parameter $1 represents actual parameter **flow,r**

Parameter $1 represents actual parameter **lin**

Port Modulation represents actual port **mod**

Port ThermalIn represents actual port **in**

Port ThermalOut represents actual port **out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: FMRT_lbl.txt

```
#SUMMARY FMRT: Two port thermal resistance with T/Sdot bonds with flow m
#DESCRIPTION Port [in]: T/Sdot power in
#DESCRIPTION Port [out]: T/Sdot power out
#DESCRIPTION CR and parameters as for a one-port R component
#DESCRIPTION Internally pseudo bond graph
#DESCRIPTION Example label file entry:
#DESCRIPTION # Component type RT
#DESCRIPTION r lin flow,r

#ALIAS ThermalIn in
#ALIAS ThermalOut out
#ALIAS Modulation mod

#ALIAS $1 flow,r
```

```
#ALIAS $1 lin


## Label file for system FMRT (FMRT_lbl.txt)

# ###############################################################
# ## Version control history
# ###############################################################
# ## $Id: FMRT_lbl.txt,v 1.1 2003/08/13 16:12:55 gawthrop Exp $
# ## $Log: FMRT_lbl.txt,v $
# ## Revision 1.1  2003/08/13 16:12:55  gawthrop
# ## New modulated thermal resistance for use in Thermodynamic Cyc
# ## Eg CarnotCycle
# ##
# ## Revision 1.8  2001/07/05 08:42:41  gawthrop
# ## Updated to allow auto-generation of sensitivity version
# ##
# ## Revision 1.7  2001/07/03 22:59:10  gawthrop
# ## Fixed problems with argument passing for CRs
# ##
# ## Revision 1.6  2001/06/13 17:10:26  gawthrop
# ## Alias for the cr (ie ALIAS $1 lin)
# ##
# ## Revision 1.5  2001/06/11 19:51:08  gawthrop
# ## Zapped spurious $1 alias
# ##
# ## Revision 1.4  1998/07/22 11:31:42  peterg
# ## New port names
# ##
# ## Revision 1.3  1998/07/21 16:26:05  peterg
# ## Now has aliased parameters.
# ##
# ## Revision 1.2  1998/06/29 10:08:14  peterg
# ## Converted to FP component
# ## Removed lables from FP
# ##
# ## Revision 1.1  1997/09/04  09:48:47  peterg
# ## Initial revision
# ##
# ###############################################################
```

```
## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank


# Component type FMR
r lin flow,r

# Component type FP
        fp1
        fp2

# Component type SS
[in] SS external,external
[out] SS external,external
[mod] SS external,external
```

Subsystems

No subsystems.

8.1.6 INTF

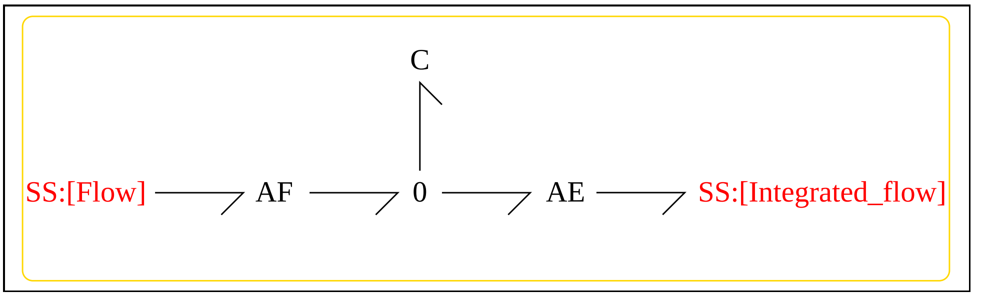


Figure 8.5: System **INTF**: acausal bond graph

The acausal bond graph of system **INTF** is displayed in Figure 13.4 (on page 299) and its label file is listed in Section 13.1.5 (on page 299). The subsystems are listed in Section 13.1.5 (on page 301).
INTF is a two-port component where the effort on port [out] is the integral of the flow on port [in].

Summary information

System INTF::flow integrator Port [in]: Flow to be integrated Port [out]: Effort = integral of flow on port [in]

Interface information:

Port in represents actual port **Flow**

Port out represents actual port **Integrated_flow**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: INTF_lbl.txt

```
%% Label file for system INTF (INTF_lbl.txt)
%SUMMARY INTF: flow integrator
%DESCRIPTION Port [in]: Flow to be integrated
%DESCRIPTION Port [out]: Effort = integral of flow on port [in]


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: INTF_lbl.txt,v 1.3 1998/07/16 07:35:10 peterg Exp $
% %% $Log: INTF_lbl.txt,v $
% %% Revision 1.3  1998/07/16 07:35:10  peterg
% %% Aliased version
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in Flow
%ALIAS out Integrated_flow
```

```
% Argument aliases

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type SS
[Flow] SS external,external
[Integrated_flow] SS external,external
```

Subsystems

No subsystems.

8.1.7 Se

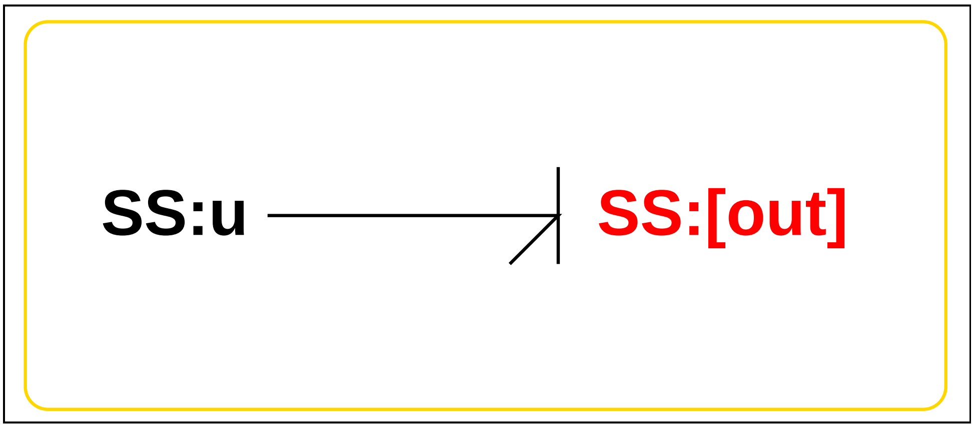


Figure 8.6: System **Se**: acausal bond graph

The acausal bond graph of system **Se** is displayed in Figure 8.6 (on page 199) and its label file is listed in Section 8.1.7 (on page 200). The subsystems are listed in Section 8.1.7 (on page 201).

Summary information

System Se:Simple effort source Simple effort source constructed from SS with fixed causality

Interface information:

Parameter $1 represents actual parameter **e_s**

Port in represents actual port **out**

Port out represents actual port **out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Se_lbl.txt

```
%% Label file for system Se (Se_lbl.txt)
%SUMMARY Se Simple effort source
%DESCRIPTION Simple effort source constructed from SS with fixed c


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Se_lbl.txt,v 1.3 1999/08/05 07:31:39 peterg Exp $
% %% $Log: Se_lbl.txt,v $
% %% Revision 1.3  1999/08/05 07:31:39  peterg
% %% Added in alias
% %%
% %% Revision 1.2  1999/03/12 04:04:27  peterg
% %% Single argument - the effort value e_s
% %%
% %% Revision 1.1  1999/03/03 21:55:46  peterg
% %% Initial revision
% %%
```

```
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS out|in out

% Argument aliases
%ALIAS $1 e_s

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type SS
[out] SS external,external
u SS e_s,internal
```

Subsystems

No subsystems.

8.1.8 Sf

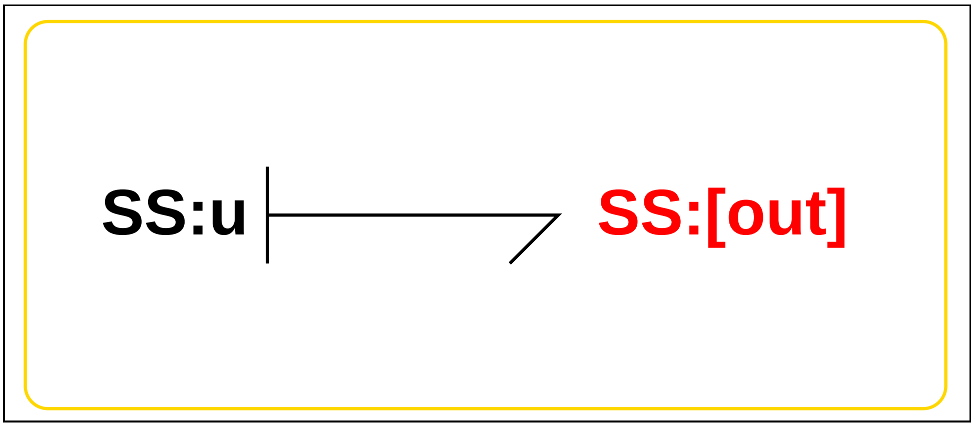


Figure 8.7: System **Sf**: acausal bond graph

The acausal bond graph of system **Sf** is displayed in Figure 9.5 (on page 224) and its label file is listed in Section 9.1.6 (on page 224). The subsystems are listed in Section 9.1.6 (on page 226).

Summary information

System Sf:Simple flow source Simple flow source constructed from SS with fixed causality

Interface information:

Parameter $1 represents actual parameter **f_s**

Port in represents actual port **out**

Port out represents actual port **out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Sf_lbl.txt

```
%% Label file for system Sf (Sf_lbl.txt)
%SUMMARY Sf Simple flow source
%DESCRIPTION  Simple flow source constructed from SS with fixed ca

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Sf_lbl.txt,v 1.3 1999/08/05 07:32:07 peterg Exp $
% %% $Log: Sf_lbl.txt,v $
% %% Revision 1.3  1999/08/05 07:32:07  peterg
% %% Added in alias
% %%
% %% Revision 1.2  1999/03/12 04:03:09  peterg
% %% Single argument - the value of the flow
% %%
```

```
% %% Revision 1.1  1999/03/03 21:50:15  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS out|in out

% Argument aliases
%ALIAS    $1 f_s
%% Each line should be of one of the following forms:
%      a comment (ie starting with %)
%      component-name cr_name arg1,arg2,..argn
%      blank

% ---- Component labels ----

% Component type SS
[out] SS external,external
u SS internal,f_s
```

Subsystems

No subsystems.

8.2 CarnotCycle_struc.tex *(-ss -o)*

MTT command:

```
mtt -ss -o CarnotCycle struc tex
```

List of inputs for system CarnotCycle			
	Component	System	Repetition
1	u	CarnotCycle__Heat__u	1
2	u	CarnotCycle__Work__u	1
3	u	CarnotCycle__switch__u	1

List of outputs for system CarnotCycle			
	Component	System	Repetition
1	S	CarnotCycle__cycle__S	1
2	T	CarnotCycle__cycle__T	1
3	V	CarnotCycle__cycle__V	1
4	P	CarnotCycle__cycle__P	1

List of states for system CarnotCycle			
	Component	System	Repetition
1	c	CarnotCycle__cycle__gas__c	1
2	c	CarnotCycle__cycle__gas__c_2	1
3	mttC	CarnotCycle__cycle__entropy__mttC	1
4	mttC	CarnotCycle__cycle__volume__mttC	1

8.3 CarnotCycle_ode.tex *(-ss -o)*

MTT command:

```
mtt -ss -o CarnotCycle ode tex
```

$$\begin{aligned}
\dot{x}_1 &= \frac{(-c_v \gamma_g m_g u_2 x_1 + c_v m_g u_1 u_3 x_2 + c_v m_g u_2 x_1 - u_3 x_1 x_2)}{(c_v m_g x_2)} \\
\dot{x}_2 &= u_2 \\
\dot{x}_3 &= \frac{(u_3 (c_v m_g u_1 - x_1))}{x_1} \\
\dot{x}_4 &= u_2
\end{aligned} \tag{8.2}$$

$$\begin{aligned}
y_1 &= x_3 \\
y_2 &= \frac{x_1}{(c_v m_g)} \\
y_3 &= x_4 \\
y_4 &= \frac{(x_1 (\gamma_g - 1))}{x_2}
\end{aligned} \tag{8.3}$$

8.4 CarnotCycle_ss.tex *(-ss -o)*

MTT command:

```
mtt -ss -o CarnotCycle ss tex
```

$$x = \begin{pmatrix} \frac{(p_0 v_0)}{(\gamma_g - 1)} \\ v_0 \\ \frac{(p_0 v_0)}{(t_0(\gamma_g - 1))} \\ v_0 \end{pmatrix} \tag{8.4}$$

$$u = \begin{pmatrix} t_0 \\ 0 \\ 0 \end{pmatrix} \tag{8.5}$$

$$y = \begin{pmatrix} \frac{(p_0 v_0)}{(t_0(\gamma_g - 1))} \\ t_0 \\ v_0 \\ p_0 \end{pmatrix} \tag{8.6}$$

$$\dot{x} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{8.7}$$

8.5 CarnotCycle_numpar.txt *(-ss -o)*

MTT command:

```
mtt -ss -o CarnotCycle numpar txt
```

```
# Numerical parameter file (CarnotCycle_numpar.txt)
# Generated by MTT at Mon Dec  8 20:02:31 GMT 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
# %% $Id: CarnotCycle_numpar.txt,v 1.4 2003/08/14 20:17:31 gawthro
# %% $Log: CarnotCycle_numpar.txt,v $
# %% Revision 1.4  2003/08/14 20:17:31  gawthrop
# %% Tidy up minor bugs
# %%
# %% Revision 1.3  2003/08/13 17:03:10  gawthrop
# %% Updated for new MTT
# %% Use FMR in place of R
# %% Fixed bug in sspar
# %%
# %% Revision 1.2  2000/12/28 18:14:40  peterg
# %% To RCS
# %%
# %% Revision 1.1  1998/03/04 11:49:01  peterg
# %% Initial revision
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Initial states -- needed to choose an approppriate mass
P_0 = 1e5;
V_0 = 1;
T_0 = 300;

# Parameters
c_v =  718.0; # Parameter c_v for CU
gamma_g = 1.4; # Parameter gamma for CU
m_g =  P_0*V_0/(T_0*(gamma_g-1)*c_v); # Parameter m for CU

## Use in input.txt
alpha = 1.0; # Added by MTT on Thu Aug 14 11:42:15 BST 2003
toptemp = 1.0; # Added by MTT on Thu Aug 14 11:42:15 BST 2003
```

8.6 CarnotCycle_input.txt *(-ss -o)*

MTT command:

```
mtt -ss -o CarnotCycle input txt
```

```
# Input file (CarnotCycle_input.txt)
# Generated by MTT at Mon Dec  8 20:05:30 GMT 1997
```

```
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: CarnotCycle_input.txt,v 1.2 2003/08/13 17:02:49 gawthrop Exp $
# %% $Log: CarnotCycle_input.txt,v $
# %% Revision 1.2  2003/08/13 17:02:49  gawthrop
# %% Updated for new MTT
# %% Use FMR in place of R
# %% Fixed bug in sspar
# %%
# %% Revision 1.1  2000/12/28 18:14:40  peterg
# %% To RCS
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


alpha = 0.553;

# Set the inputs
if ((t>=0.0)&&(t<1.0)) #Cooling
  carnotcycle__heat__u= 300.0; # Source temperature
  carnotcycle__switch__u    = 1e10; # Large conductance - isothermal
  carnotcycle__work__u = -alpha; # Volume rate-of-change
endif;

if ((t>=1.0)&&(t<2.0)) #Compression
  carnotcycle__heat__u = 300.0; # Source temperature
  carnotcycle__switch__u    = 0; # Small conductance -- isentropic
  carnotcycle__work__u = -(0.8-alpha); # Volume rate-of-change
endif;

if ((t>=2.0)&&(t<3.0)) #Heating
  TopTemp = x(1)/(m_g*c_v);
  carnotcycle__heat__u = TopTemp; # Source temperature
  carnotcycle__switch__u    = 1e10; # Large conductance - isothermal
  carnotcycle__work__u = (0.8-alpha); # Volume rate-of-change
endif;

if (t>=3.0) #Cooling
  carnotcycle__heat__u = 300.0; # Source temperature
```

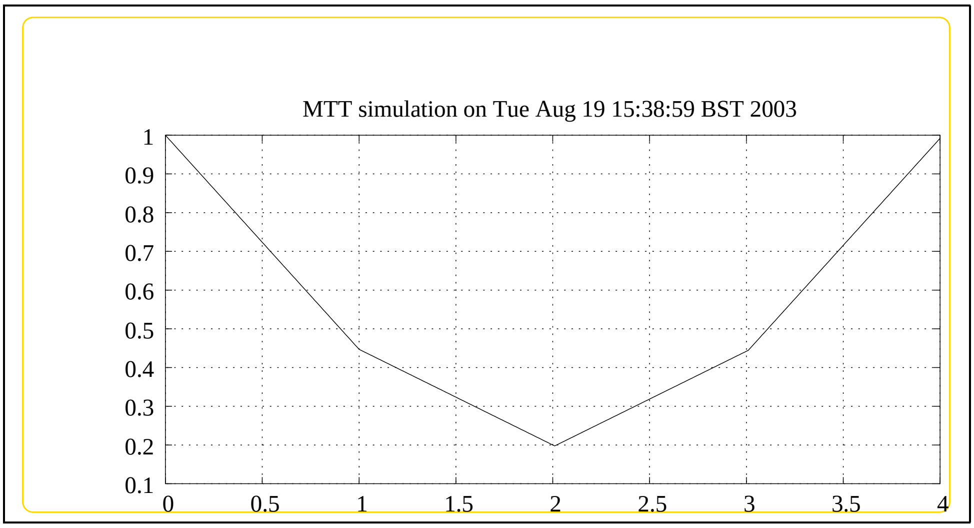
```
  carnotcycle__switch__u    = 0; # Small conductance -- isentropic
  carnotcycle__work__u = alpha; # Volume rate-of-change
endif;
```

8.7 CarnotCycle_odeso.ps *(-ss -o -CarnotCycle__cycle__V)*

MTT command:

```
mtt -ss -o CarnotCycle odeso ps 'CarnotCycle__cycle__V'
```

This representation is given as Figure 8.8 (on page 208).

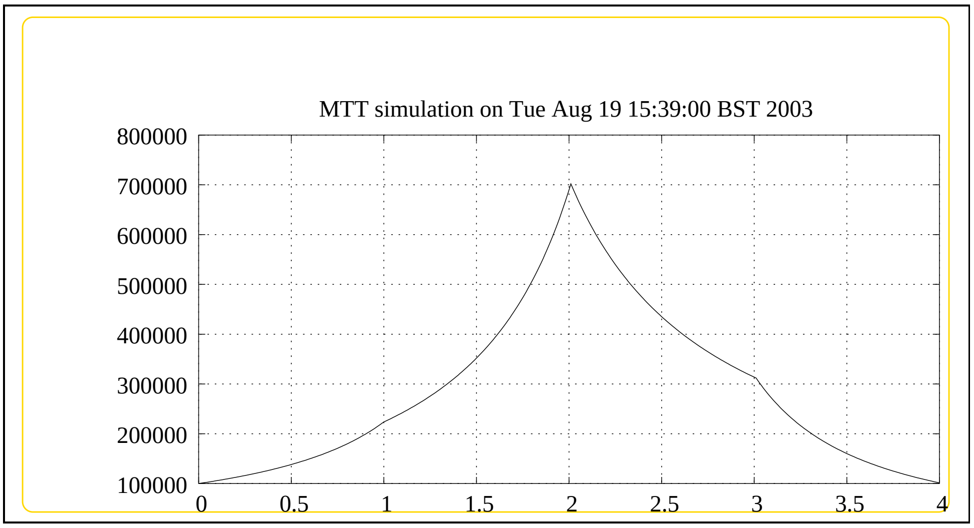


8.8 CarnotCycle_odeso.ps *(-ss -o -CarnotCycle__cycle__P)*

MTT command:

```
mtt -ss -o CarnotCycle odeso ps 'CarnotCycle__cycle__P'
```

This representation is given as Figure 8.9 (on page 209).

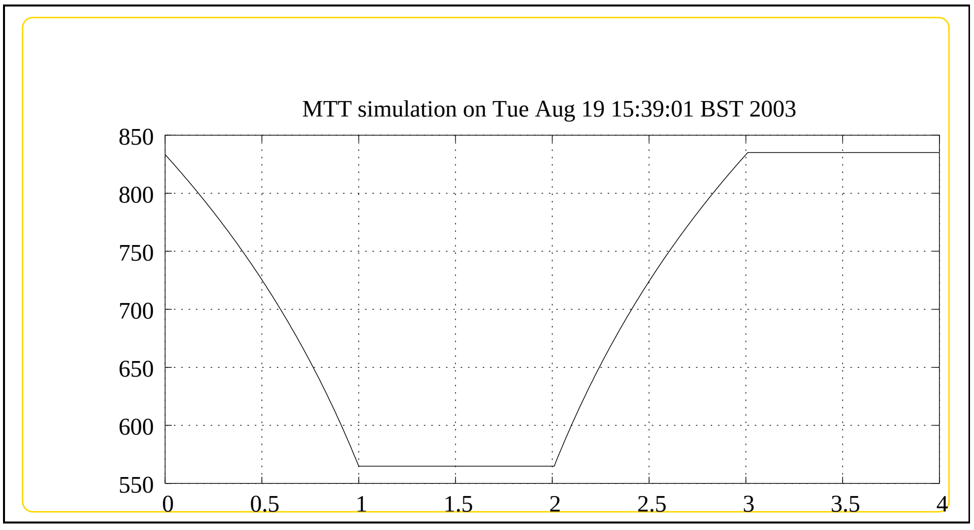


8.9 CarnotCycle_odeso.ps *(-ss -o -CarnotCycle__cycle__S)*

MTT command:

```
mtt -ss -o CarnotCycle odeso ps 'CarnotCycle__cycle__S'
```

This representation is given as Figure 8.10 (on page 209).

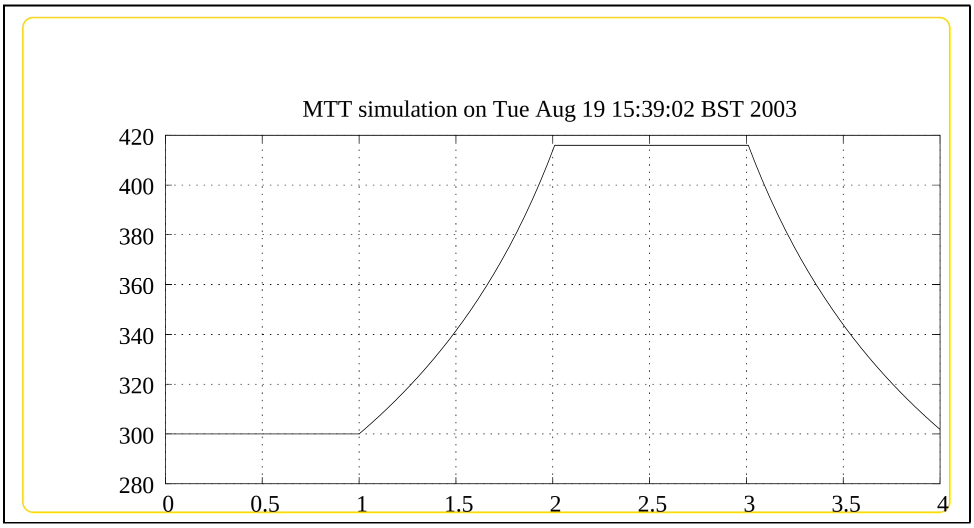


8.10 CarnotCycle_odeso.ps *(-ss -o -CarnotCycle__cycle__T)*

MTT command:

```
mtt -ss -o CarnotCycle odeso ps 'CarnotCycle__cycle__T'
```

This representation is given as Figure 8.11 (on page 210).



8.11 CarnotCycle_odeso.ps *(-ss -o -CarnotCycle__cycle__V:CarnotCycle__cycle__P)*

MTT command:

```
mtt -ss -o CarnotCycle odeso ps 'CarnotCycle__cycle__V:CarnotCycle
```

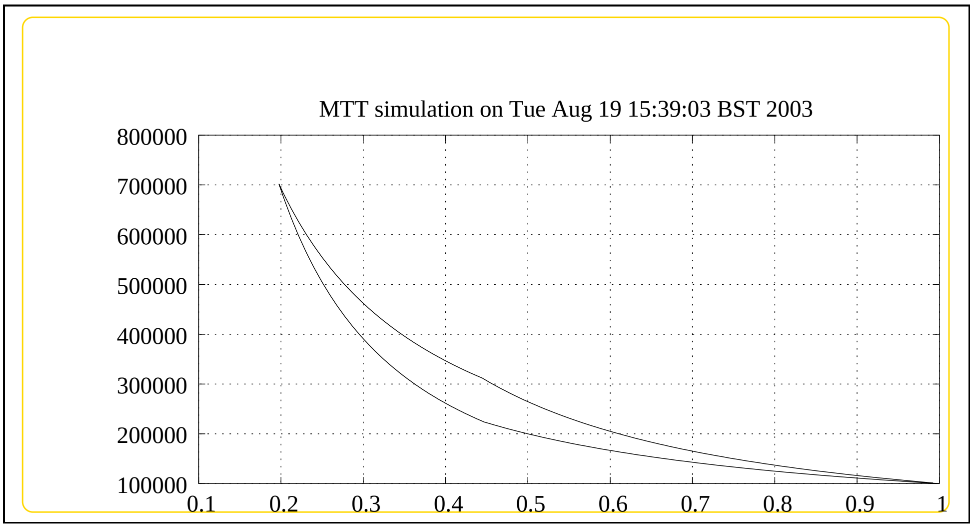
This representation is given as Figure 8.12 (on page 211).

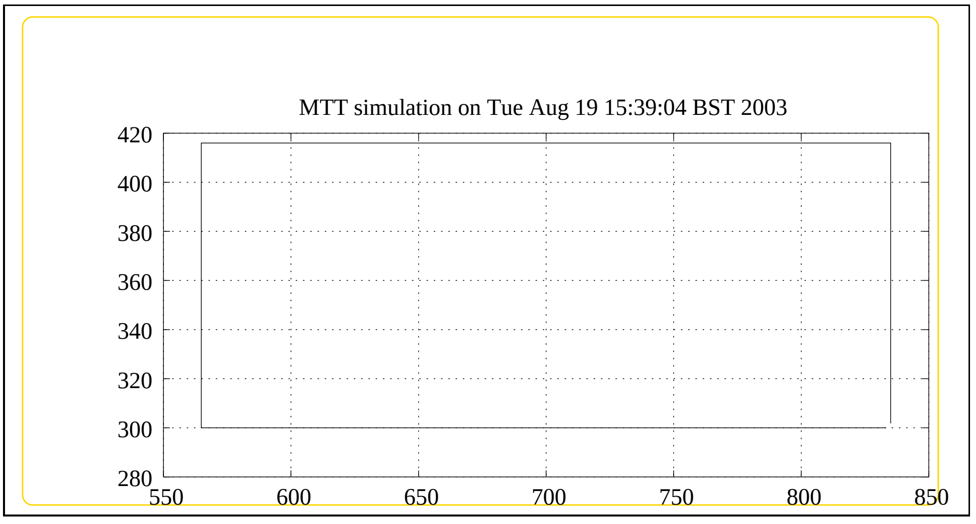
8.12 CarnotCycle_odeso.ps *(-ss -o -CarnotCycle__cycle__S:CarnotCycle__cycle__T)*

MTT command:

```
mtt -ss -o CarnotCycle odeso ps 'CarnotCycle__cycle__S:CarnotCycle
```

This representation is given as Figure 8.13 (on page 211).


MTT simulation on Tue Aug 19 15:39:03 BST 2003
800000
700000
600000
500000
400000
300000
200000
100000
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


MTT simulation on Tue Aug 19 15:39:04 BST 2003
420
400
380
360
340
320
300
280
550
600
650
700
750
800
850

Chapter 9

Otto Cycle

9.1 OttoCycle_abg.tex *(-o -ss)*

MTT command:

```
mtt -o -ss OttoCycle abg tex
```

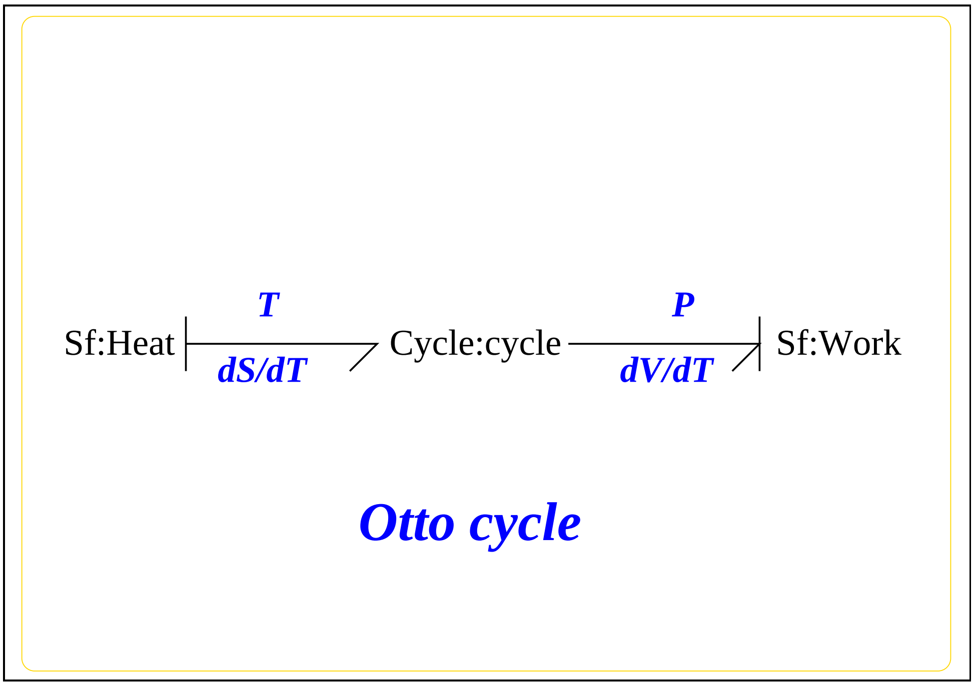


Figure 9.1: System **OttoCycle**: acausal bond graph

The acausal bond graph of system **OttoCycle** is displayed in Figure 9.1 (on page 213) and its label file is listed in Section 9.1.1 (on page 214). The subsystems are listed in Section 9.1.2 (on page 216).

The Otto cycle is a simple closed thermodynamic cycle with four parts:

1. Isentropic compression
2. Heating at constant volume
3. Isentropic expansion
4. Cooling at constant volume

The subsystem **Cycle** (Section 13.1.4 (on page 295)) is a two-port component describing an ideal gas. It has two energy ports which, with integral causality correspond to

1. Entropy flow in; temperature out
2. Volume rate of change in; pressure out

In Bond Graph terms, each of the four parts of the Otto cycle correspond to integral causality as in each case a *flow* is constrained. This is in contrast to other cycles listed in Table 13.1 (on page 296) where each table entry gives the causality on the heat and work ports respectively. This is possibly why the Otto cycle is conceptually and practically simple.

The simulation parameters appear in Section **??** (on page **??**). The results are plotted against time as follows:

- Volume (Figure **??** (on page **??**))
- Pressure (Figure **??** (on page **??**))
- Entropy (Figure **??** (on page **??**))
- Temperature (Figure **??** (on page **??**))

These values are replotted as the standard PV and TS diagrams in Figures **??** (on page **??**) and **??** (on page **??**) respectively.

9.1.1 Summary information

System OttoCycle::a simple closed thermodynamic cycle The Otto cycle is a simple closed thermodynamic cycle with four parts: o Isentropic compression o Heating at constant volume o Isentropic expansion o Cooling at constant volume

Interface information:

This component has no ALIAS declarations

Variable declarations:

P_0

T_0

V_0

Units declarations:

This component has no UNITs declarations

The label file: OttoCycle_lbl.txt

```
#SUMMARY OttoCycle: a simple closed thermodynamic cycle
#DESCRIPTION The Otto cycle is a simple closed thermodynamic cycle
#DESCRIPTION with four parts:
#DESCRIPTION o Isentropic compression
#DESCRIPTION o Heating at constant volume
#DESCRIPTION o Isentropic expansion
#DESCRIPTION o Cooling at constant volume

#PAR P_0
#PAR T_0
#PAR V_0
#NOTPAR ideal_gas


## Label file for system OttoCycle (OttoCycle_lbl.txt)

# ###############################################################
# ## Version control history
# ###############################################################
# ## $Id: OttoCycle_lbl.txt,v 1.7 2003/08/14 20:17:50 gawthrop Exp $
# ## $Log: OttoCycle_lbl.txt,v $
# ## Revision 1.7  2003/08/14 20:17:50  gawthrop
# ## Tidy up minor bugs
# ##
```

```
# ## Revision 1.4  2000/12/28 18:42:17  peterg
# ## New input definition.
# ## NB Needs to handle mutiports properly and state.txt modified
# ##
# ## Revision 1.3  1998/08/10 16:05:52  peterg
# ## Added VARs and parameters
# ##
# ## Revision 1.2  1998/07/21 15:15:27  peterg
# ## Documentation + new format
# ##
# ## Revision 1.1  1998/07/21 15:12:21  peterg
# ## Initial revision
# ##
# ##################################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank



# Component type Cycle
cycle none ideal_gas;c_v;gamma_g;m_g

# Component type Sf
Heat SS external
Work SS external
```

9.1.2 Subsystems

- (1)
  - CU (1)
  - INTF: flow integrator (2)
- Sf Simple flow source (2) No subsystems.

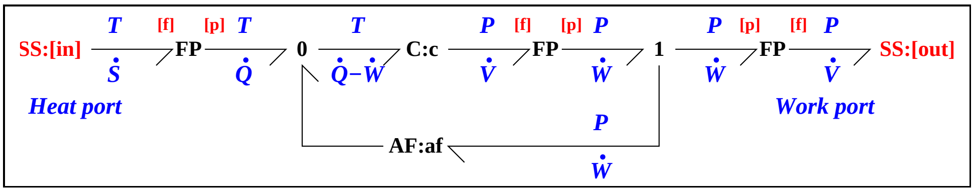


Figure 9.2: System **CU**: acausal bond graph

9.1.3 CU

The acausal bond graph of system **CU** is displayed in Figure 13.2 (on page 294) and its label file is listed in Section 13.1.3 (on page 294). The subsystems are listed in Section 13.1.3 (on page 295).

Summary information

System CU: ¡Detailed description here¿

Interface information:

This component has no ALIAS declarations

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: CU_lbl.txt

```
%SUMMARY CU
%DESCRIPTION <Detailed description here>
%% Label file for system CU (CU_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: CU_lbl.txt,v 1.1 2000/12/28 10:34:56 peterg Exp $
% %% $Log: CU_lbl.txt,v $
```

```
% %% Revision 1.1  2000/12/28 10:34:56  peterg
% %% Put under RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

% Component type AF (gain of -1)
af lin -1

% Component type C
c CU $1

% Component type SS
[in] SS external,external
[out] SS external,external
```

Subsystems

No subsystems.

9.1.4 Cycle

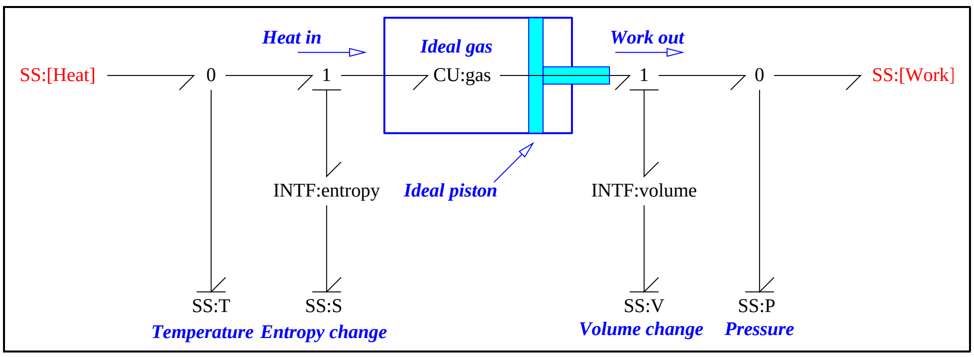


Figure 9.3: System **Cycle**: acausal bond graph

The acausal bond graph of system **Cycle** is displayed in Figure 13.3 (on page 295) and its label file is listed in Section 13.1.4 (on page 296). The subsystems

are listed in Section 13.1.4 (on page 299).
The system has two heat engine ports:

1. **[Heat]** and
2. **[Work]**

By convention, energy flows in to the **[Heat]** port and out of the **[Work]** port. Both ports are true energy ports.
The subsystem **CU** (Section 13.1.3 (on page 293)) is a two-port component describing an ideal gas. It has two energy ports which, with integral causality correspond to

1. Entropy flow in; temperature out
2. Volume rate of change in; pressure out

Four **SS** components are set up as sensors to measure the important quantities pertaining to the cycle:

1. **S** to measure the integrated entropy flow
2. **T** to measure the (absolute) temperature
3. **V** to measure the integrated volume change
4. **P** to measure the pressure

Cycle	Compression	Heating	Expansion	Cooling
Otto	II	II	II	II
Carnot	II	DI	II	DI
Diesel	II	ID	II	II
Joule	II	ID	II	ID

Table 9.1: Cycles and their causality

A number of cycles can be built depending on the causality of the two ports **[Heat]** and **[Work]** of **Cycle** Some possible cycles listed in Tables 13.1 (on page 296) where each table entry gives the causality on the heat and work ports respectively.

Summary information

System Cycle::Closed cycle with ideal gas Uses the CU two-port thermal capacitor.

Interface information:

Parameter $1 represents actual parameter **ideal_gas**

Parameter $2 represents actual parameter **c_v**

Parameter $3 represents actual parameter **gamma_g**

Parameter $4 represents actual parameter **m_g**

Port in represents actual port **Heat**

Port out represents actual port **Work**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Cycle_lbl.txt

```
#SUMMARY Cycle: Closed cycle with ideal gas
#DESCRIPTION Uses the CU two-port thermal capacitor.

#ALIAS in Heat
#ALIAS out Work

#ALIAS $1 ideal_gas
#ALIAS $2 c_v
#ALIAS $3 gamma_g
#ALIAS $4 m_g

## Label file for system Cycle (Cycle_lbl.txt)

# ###############################################################
# ## Version control history
# ###############################################################
# ## $Id: Cycle_lbl.txt,v 1.3 2000/12/27 16:38:28 peterg Exp $
# ## $Log: Cycle_lbl.txt,v $
# ## Revision 1.3  2000/12/27 16:38:28  peterg
```

```
# ## *** empty log message ***
# ##
# ## Revision 1.2  1998/07/21 14:21:04  peterg
# ## New style file
# ##
# ## Revision 1.1  1997/12/08 20:24:43  peterg
# ## Initial revision
# ##
# ## Revision 1.1  1997/12/07 20:38:05  peterg
# ## Initial revision
# ##
# ###################################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank




# Component type CU
gas CU ideal_gas,c_v,gamma_g,m_g

# Component type INTF
entropy
volume

# Component type SS
S SS external,0
T SS external,0
V SS external,0
P SS external,0
[Heat] SS external,external
[Work] SS external,external
```

Subsystems

- CU (1) No subsystems.
- INTF: flow integrator (2) No subsystems.

9.1.5 INTF

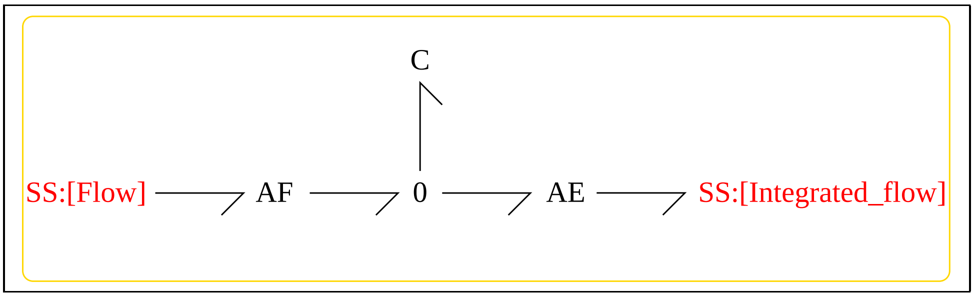


Figure 9.4: System **INTF**: acausal bond graph

The acausal bond graph of system **INTF** is displayed in Figure 13.4 (on page 299) and its label file is listed in Section 13.1.5 (on page 299). The subsystems are listed in Section 13.1.5 (on page 301).
INTF is a two-port component where the effort on port [out] is the integral of the flow on port [in].

Summary information

System INTF::flow integrator Port [in]: Flow to be integrated Port [out]: Effort = integral of flow on port [in]

Interface information:

Port in represents actual port **Flow**

Port out represents actual port **Integrated_flow**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: INTF_lbl.txt

```
%% Label file for system INTF (INTF_lbl.txt)
%SUMMARY INTF: flow integrator
%DESCRIPTION Port [in]: Flow to be integrated
%DESCRIPTION Port [out]: Effort = integral of flow on port [in]


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: INTF_lbl.txt,v 1.3 1998/07/16 07:35:10 peterg Exp $
% %% $Log: INTF_lbl.txt,v $
% %% Revision 1.3  1998/07/16 07:35:10  peterg
% %% Aliased version
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in Flow
%ALIAS out Integrated_flow

% Argument aliases

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type SS
[Flow] SS external,external
[Integrated_flow] SS external,external
```

Subsystems

No subsystems.

9.1.6 Sf

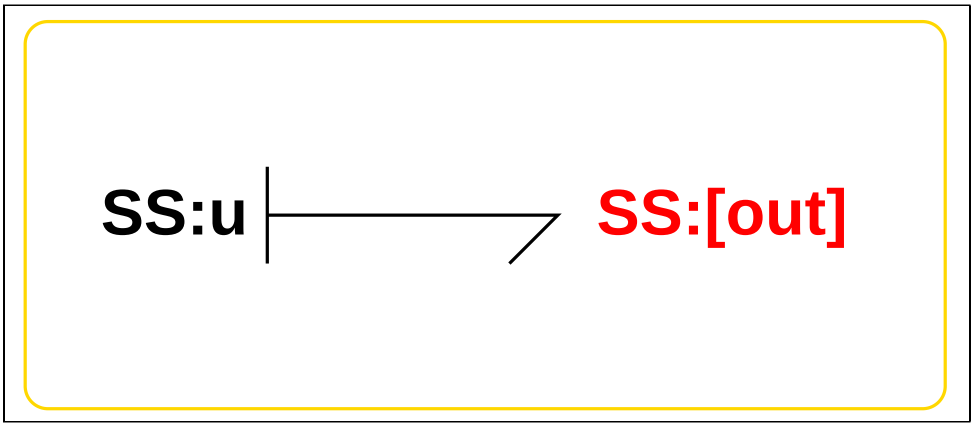


Figure 9.5: System **Sf**: acausal bond graph

The acausal bond graph of system **Sf** is displayed in Figure 9.5 (on page 224) and its label file is listed in Section 9.1.6 (on page 224). The subsystems are listed in Section 9.1.6 (on page 226).

Summary information

System Sf:Simple flow source Simple flow source constructed from SS with fixed causality

Interface information:

Parameter $1 represents actual parameter **f_s**

Port in represents actual port **out**

Port out represents actual port **out**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Sf_lbl.txt

```
%% Label file for system Sf (Sf_lbl.txt)
%SUMMARY Sf Simple flow source
%DESCRIPTION  Simple flow source constructed from SS with fixed causalit

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: Sf_lbl.txt,v 1.3 1999/08/05 07:32:07 peterg Exp $
% %% $Log: Sf_lbl.txt,v $
% %% Revision 1.3  1999/08/05 07:32:07  peterg
% %% Added in alias
% %%
% %% Revision 1.2  1999/03/12 04:03:09  peterg
% %% Single argument - the value of the flow
% %%
% %% Revision 1.1  1999/03/03 21:50:15  peterg
% %% Initial revision
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS out|in out

% Argument aliases
%ALIAS    $1 f_s
%% Each line should be of one of the following forms:
%      a comment (ie starting with %)
%      component-name cr_name arg1,arg2,..argn
%      blank

% ---- Component labels ----

% Component type SS
[out] SS external,external
```

```
u SS internal,f_s
```

Subsystems

No subsystems.

9.2 OttoCycle_struc.tex *(-o -ss)*

MTT command:

```
mtt -o -ss OttoCycle struc tex
```

List of inputs for system OttoCycle			
	Component	System	Repetition
1	u	OttoCycle__Heat__u	1
2	u	OttoCycle__Work__u	1

List of outputs for system OttoCycle			
	Component	System	Repetition
1	S	OttoCycle__cycle__S	1
2	T	OttoCycle__cycle__T	1
3	V	OttoCycle__cycle__V	1
4	P	OttoCycle__cycle__P	1

List of states for system OttoCycle			
	Component	System	Repetition
1	c	OttoCycle__cycle__gas__c	1
2	c	OttoCycle__cycle__gas__c_2	1
3	mttC	OttoCycle__cycle__entropy__mttC	1
4	mttC	OttoCycle__cycle__volume__mttC	1

9.3 OttoCycle_ode.tex *(-o -ss)*

MTT command:

```
mtt -o -ss OttoCycle ode tex
```

$$\begin{aligned}
\dot{x}_1 &= \frac{(x_1(-c_v\gamma_g m_g u_2 + c_v m_g u_2 + u_1 x_2))}{(c_v m_g x_2)} \\
\dot{x}_2 &= u_2 \\
\dot{x}_3 &= u_1 \\
\dot{x}_4 &= u_2
\end{aligned} \tag{9.1}$$

$$\begin{aligned}
y_1 &= x_3 \\
y_2 &= \frac{x_1}{(c_v m_g)} \\
y_3 &= x_4 \\
y_4 &= \frac{(x_1(\gamma_g - 1))}{x_2}
\end{aligned} \tag{9.2}$$

9.4 OttoCycle_ss.tex *(-o -ss)*

MTT command:

```
mtt -o -ss OttoCycle ss tex
```

$$x = \begin{pmatrix} \frac{(p_0 v_0)}{(\gamma_g - 1)} \\ v_0 \\ \frac{(p_0 v_0)}{(t_0(\gamma_g - 1))} \\ v_0 \end{pmatrix} \tag{9.3}$$

$$u = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{9.4}$$

$$y = \begin{pmatrix} \frac{(p_0 v_0)}{(t_0(\gamma_g - 1))} \\ t_0 \\ v_0 \\ p_0 \end{pmatrix} \tag{9.5}$$

$$\dot{x} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{9.6}$$

9.5 OttoCycle_numpar.txt *(-o -ss)*

MTT command:

```
mtt -o -ss OttoCycle numpar txt

# Numerical parameter file (OttoCycle_numpar.txt)
# Generated by MTT at Thu Dec  4 11:44:46 GMT 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: OttoCycle_numpar.txt,v 1.2 2000/12/28 18:15:52 peterg Ex
# %% $Log: OttoCycle_numpar.txt,v $
# %% Revision 1.2  2000/12/28 18:15:52  peterg
# %% To RCS
# %%
# %% Revision 1.1  1998/03/04 11:45:49  peterg
# %% Initial revision
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Initial states -- needed to choose an appropriate mass
P_0 = 1e5;
V_0 = 1;
T_0 = 300;

# Parameters
c_v =  718.0; # Parameter c_v for CU
gamma_g = 1.4; # Parameter gamma for CU
m_g =  P_0*V_0/(T_0*(gamma_g-1)*c_v);# Parameter m for CU
```

9.6 OttoCycle_input.txt *(-o -ss)*

MTT command:

```
mtt -o -ss OttoCycle input txt

# Numerical parameter file (OttoCycle_input.txt)
```

```
# Generated by MTT at Thu Dec  4 11:17:09 GMT 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: OttoCycle_input.txt,v 1.6 2003/08/14 20:17:45 gawthrop Exp $
# %% $Log: OttoCycle_input.txt,v $
# %% Revision 1.6  2003/08/14 20:17:45  gawthrop
# %% Tidy up minor bugs
# %%
# %% Revision 1.3  2000/12/28 18:42:16  peterg
# %% New input definition.
# %% NB Needs to handle mutiports properly and state.txt modified accor
# %%
# %% Revision 1.2  2000/12/28 18:15:52  peterg
# %% To RCS
# %%
# %% Revision 1.1  1998/08/10 14:42:13  peterg
# %% Initial revision
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Set the inputs

if ((t>=0.0)&&(t<1.0)) #Compression
  OttoCycle__Heat__u = 0.0; # Entropy flow
  OttoCycle__Work__u = -0.8; # Volume rate-of-change
endif;

if ((t>=1.0)&&(t<2.0)) #Heating
  OttoCycle__Heat__u = 1000; # Entropy flow
  OttoCycle__Work__u = 0.0; # Volume rate-of-change
endif;

if ((t>=2.0)&&(t<3.0)) #Expansion
  OttoCycle__Heat__u = 0.0; # Entropy flow
  OttoCycle__Work__u = 0.8; # Volume rate-of-change
endif;

if (t>=3.0) #Cooling
  OttoCycle__Heat__u = -1000; # Entropy flow
```

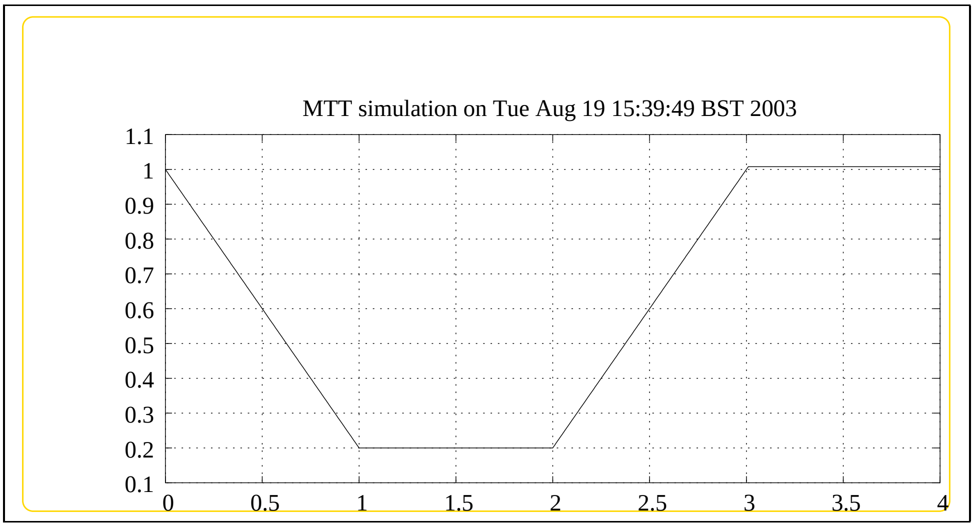
```
  OttoCycle__Work__u = 0.0; # Volume rate-of-change
endif;
```

9.7 OttoCycle_odeso.ps *(-o -ss -OttoCycle__cycle__V)*

MTT command:

```
mtt -o -ss OttoCycle odeso ps 'OttoCycle__cycle__V'
```

This representation is given as Figure 9.6 (on page 230).

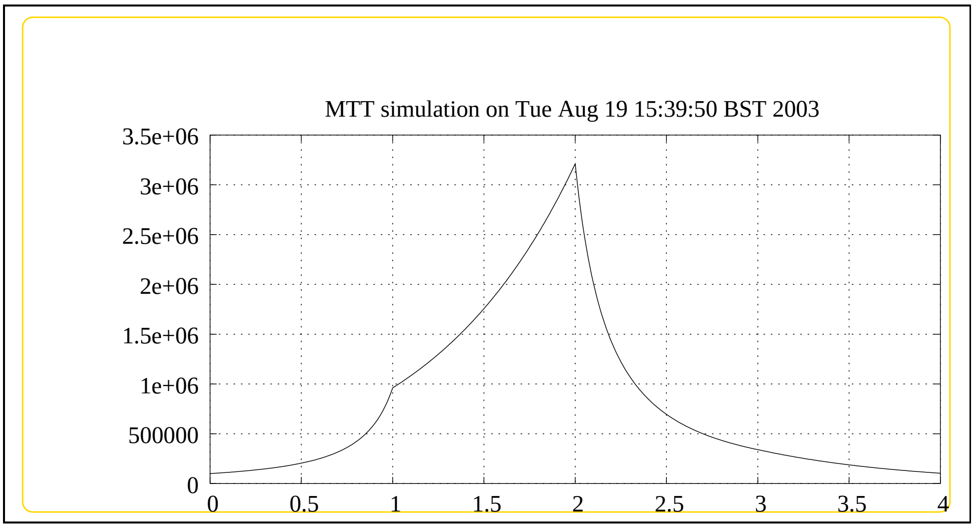


9.8 OttoCycle_odeso.ps *(-o -ss -OttoCycle__cycle__P)*

MTT command:

```
mtt -o -ss OttoCycle odeso ps 'OttoCycle__cycle__P'
```

This representation is given as Figure 9.7 (on page 231).

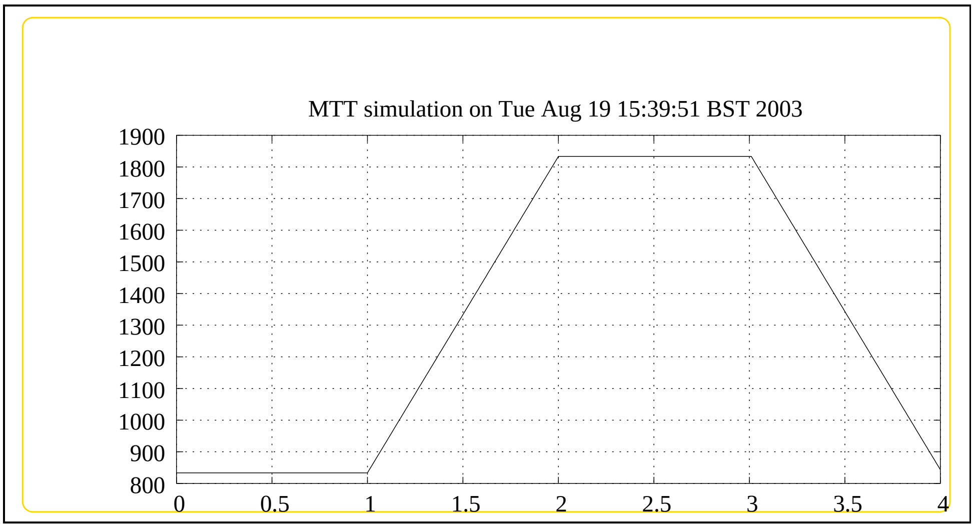


9.9 OttoCycle_odeso.ps *(-o -ss -OttoCycle__cycle__S)*

MTT command:

```
mtt -o -ss OttoCycle odeso ps 'OttoCycle__cycle__S'
```

This representation is given as Figure 9.8 (on page 231).

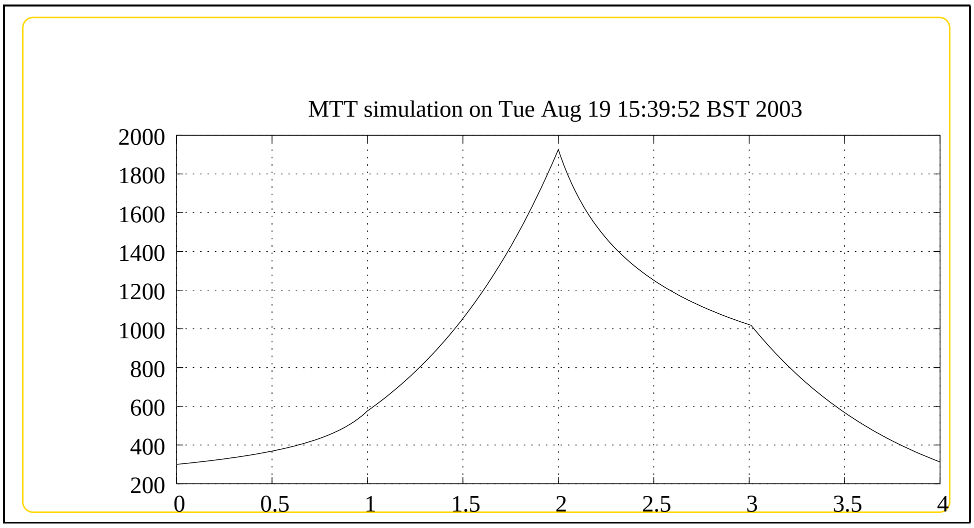


9.10 OttoCycle_odeso.ps *(-o -ss -OttoCycle__cycle__T)*

MTT command:

```
mtt -o -ss OttoCycle odeso ps 'OttoCycle__cycle__T'
```

This representation is given as Figure 9.9 (on page 232).



9.11 OttoCycle_odeso.ps *(-o -ss -OttoCycle__cycle__V:OttoCycle__cycle__P)*

MTT command:

```
mtt -o -ss OttoCycle odeso ps 'OttoCycle__cycle__V:OttoCycle__cycl
```

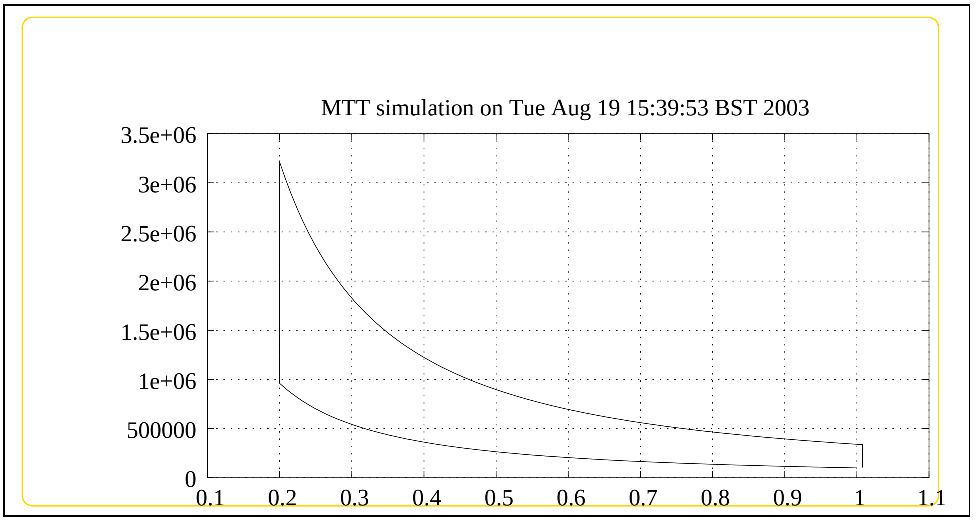
This representation is given as Figure 9.10 (on page 233).

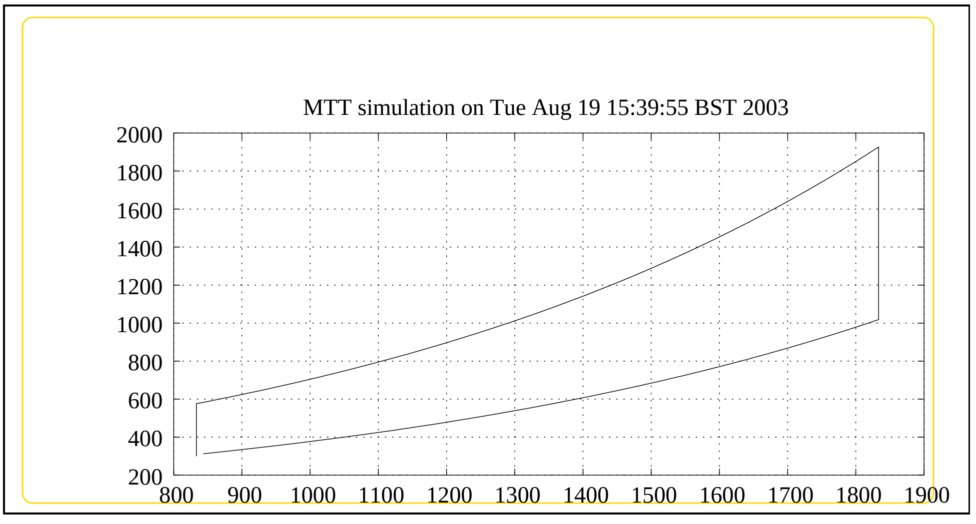
9.12 OttoCycle_odeso.ps *(-o -ss -OttoCycle__cycle__S:OttoCycle__cycle__T)*

MTT command:

```
mtt -o -ss OttoCycle odeso ps 'OttoCycle__cycle__S:OttoCycle__cycl
```

This representation is given as Figure 9.11 (on page 233).


MTT simulation on Tue Aug 19 15:39:53 BST 2003
3.5e+06
3e+06
2.5e+06
2e+06
1.5e+06
1e+06
500000
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.1


MTT simulation on Tue Aug 19 15:39:55 BST 2003
2000
1800
1600
1400
1200
1000
800
600
400
200
800
900
1000
1100
1200
1300
1400
1500
1600
1700
1800
1900

Part VI

ThermodynamicProcesses

Chapter 10

Isentropic

10.1 Isentropic_abg.tex *(-o -ss)*

MTT command:

```
mtt -o -ss Isentropic abg tex
```

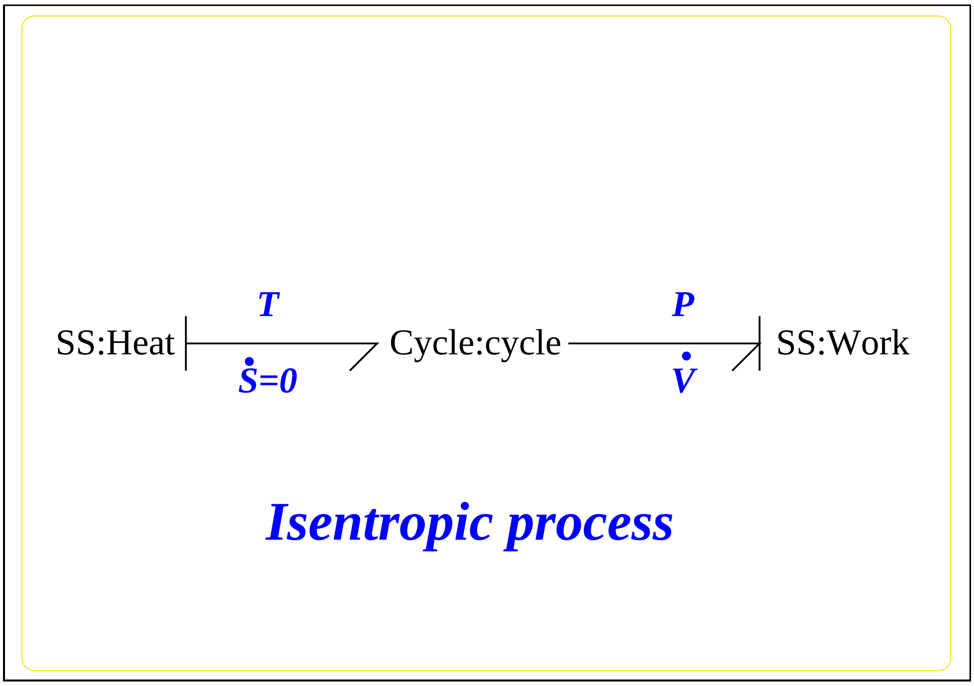


Figure 10.1: System **Isentropic**: acausal bond graph

10.1.1 Summary information

System Isentropic::Isentropic thermodynamic process - ideal gas A dynamic simulation of an isentropic process using the Cycle component and the two-prt CU component.

Interface information:

This component has no ALIAS declarations

Variable declarations:

P_0

T_0

V_0

Units declarations:

This component has no UNITs declarations

The label file: Isentropic_lbl.txt

```
#SUMMARY Isentropic: Isentropic thermodynamic process - ideal gas
#DESCRIPTION A dynamic simulation of an isentropic process using
#DESCRIPTION the Cycle component and the two-prt CU component.

#PAR P_0
#PAR T_0
#PAR V_0

#NOTPAR ideal_gas



## Label file for system Isentropic (Isentropic_lbl.txt)

# ###############################################################
# ## Version control history
# ###############################################################
# ## $Id: Isentropic_lbl.txt,v 1.3 2003/08/06 18:54:09 gawthrop Ex
```

```
# ## $Log: Isentropic_lbl.txt,v $
# ## Revision 1.3  2003/08/06 18:54:09  gawthrop
# ## Updated for latest MTT version.
# ##
# ## Revision 1.2  2000/12/28 18:16:47  peterg
# ## To RCS
# ##
# ## Revision 1.1  1998/07/21 14:27:44  peterg
# ## Initial revision
# ##
# ###############################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank



# Component type Cycle
cycle none ideal_gas;c_v;gamma_g;m_g

# Component type SS
Heat SS internal,0
Work SS internal,external
```

10.1.2 Subsystems

- (1)
  - CU (1)
  - INTF: flow integrator (2)

10.1.3 CU

The acausal bond graph of system **CU** is displayed in Figure 13.2 (on page 294) and its label file is listed in Section 13.1.3 (on page 294). The subsystems are listed in Section 13.1.3 (on page 295).

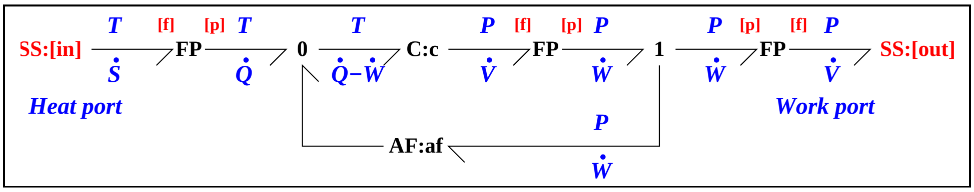


Figure 10.2: System **CU**: acausal bond graph

Summary information

System CU: ¡Detailed description here¿

Interface information:

This component has no ALIAS declarations

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: CU_lbl.txt

```
%SUMMARY CU
%DESCRIPTION <Detailed description here>
%% Label file for system CU (CU_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: CU_lbl.txt,v 1.1 2000/12/28 10:34:56 peterg Exp $
% %% $Log: CU_lbl.txt,v $
% %% Revision 1.1  2000/12/28 10:34:56  peterg
% %% Put under RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
```

```
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

% Component type AF (gain of -1)
af lin -1

% Component type C
c CU $1

% Component type SS
[in] SS external,external
[out] SS external,external
```

Subsystems

No subsystems.

10.1.4 Cycle

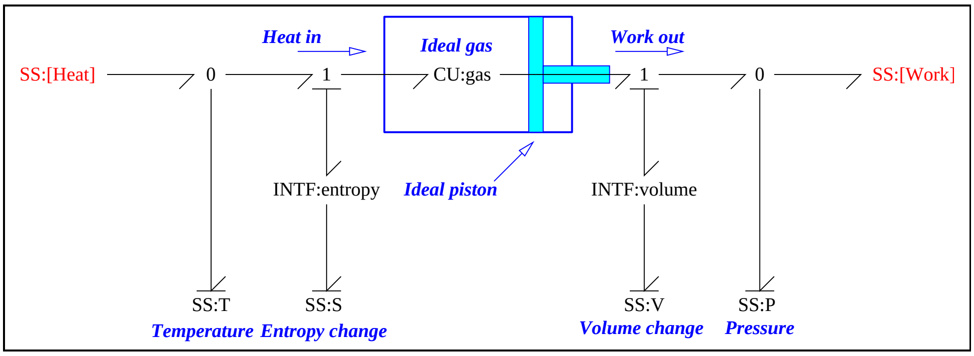


Figure 10.3: System **Cycle**: acausal bond graph

The acausal bond graph of system **Cycle** is displayed in Figure 13.3 (on page 295) and its label file is listed in Section 13.1.4 (on page 296). The subsystems are listed in Section 13.1.4 (on page 299).

The system has two heat engine ports:

1. **[Heat]** and
2. **[Work]**

By convention, energy flows in to the **[Heat]** port and out of the **[Work]** port. Both ports are true energy ports.

The subsystem **CU** (Section 13.1.3 (on page 293)) is a two-port component describing an ideal gas. It has two energy ports which, with integral causality correspond to

1. Entropy flow in; temperature out

2. Volume rate of change in; pressure out

Four **SS** components are set up as sensors to measure the important quantities pertaining to the cycle:

1. **S** to measure the integrated entropy flow

2. **T** to measure the (absolute) temperature

3. **V** to measure the integrated volume change

4. **P** to measure the pressure

Cycle	Compression	Heating	Expansion	Cooling
Otto	II	II	II	II
Carnot	II	DI	II	DI
Diesel	II	ID	II	II
Joule	II	ID	II	ID

Table 10.1: Cycles and their causality

A number of cycles can be built depending on the causality of the two ports **[Heat]** and **[Work]** of **Cycle** Some possible cycles listed in Tables 13.1 (on page 296) where each table entry gives the causality on the heat and work ports respectively.

Summary information

System Cycle::Closed cycle with ideal gas Uses the CU two-port thermal capacitor.

Interface information:

Parameter $1 represents actual parameter **ideal_gas**

Parameter $2 represents actual parameter **c_v**

Parameter $3 represents actual parameter **gamma_g**

Parameter $4 represents actual parameter **m_g**

Port in represents actual port **Heat**

Port out represents actual port **Work**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Cycle_lbl.txt

```
#SUMMARY Cycle: Closed cycle with ideal gas
#DESCRIPTION Uses the CU two-port thermal capacitor.

#ALIAS in Heat
#ALIAS out Work

#ALIAS $1 ideal_gas
#ALIAS $2 c_v
#ALIAS $3 gamma_g
#ALIAS $4 m_g

## Label file for system Cycle (Cycle_lbl.txt)

# ###############################################################
# ## Version control history
# ###############################################################
# ## $Id: Cycle_lbl.txt,v 1.3 2000/12/27 16:38:28 peterg Exp $
# ## $Log: Cycle_lbl.txt,v $
# ## Revision 1.3  2000/12/27 16:38:28  peterg
```

```
# ## *** empty log message ***
# ##
# ## Revision 1.2  1998/07/21 14:21:04  peterg
# ## New style file
# ##
# ## Revision 1.1  1997/12/08 20:24:43  peterg
# ## Initial revision
# ##
# ## Revision 1.1  1997/12/07 20:38:05  peterg
# ## Initial revision
# ##
# ###############################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank




# Component type CU
gas CU ideal_gas,c_v,gamma_g,m_g

# Component type INTF
entropy
volume

# Component type SS
S SS external,0
T SS external,0
V SS external,0
P SS external,0
[Heat] SS external,external
[Work] SS external,external
```

Subsystems

- CU (1) No subsystems.
- INTF: flow integrator (2) No subsystems.

10.1.5 INTF

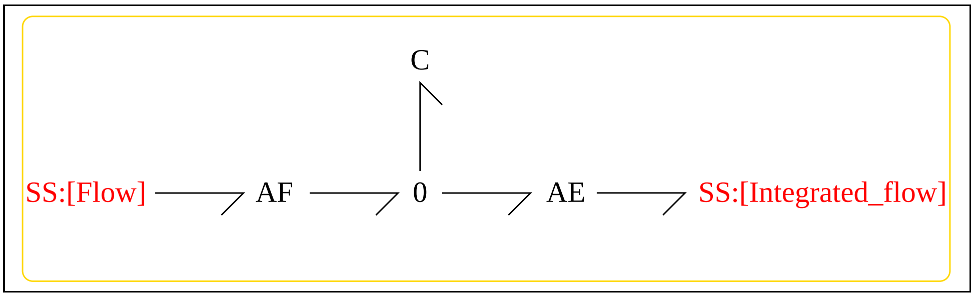


Figure 10.4: System **INTF**: acausal bond graph

The acausal bond graph of system **INTF** is displayed in Figure 13.4 (on page 299) and its label file is listed in Section 13.1.5 (on page 299). The subsystems are listed in Section 13.1.5 (on page 301).

INTF is a two-port component where the effort on port [out] is the integral of the flow on port [in].

Summary information

System INTF::flow integrator Port [in]: Flow to be integrated Port [out]: Effort = integral of flow on port [in]

Interface information:

Port in represents actual port **Flow**

Port out represents actual port **Integrated_flow**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: INTF_lbl.txt

```
%% Label file for system INTF (INTF_lbl.txt)
%SUMMARY INTF: flow integrator
%DESCRIPTION Port [in]: Flow to be integrated
%DESCRIPTION Port [out]: Effort = integral of flow on port [in]


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: INTF_lbl.txt,v 1.3 1998/07/16 07:35:10 peterg Exp $
% %% $Log: INTF_lbl.txt,v $
% %% Revision 1.3  1998/07/16 07:35:10  peterg
% %% Aliased version
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in Flow
%ALIAS out Integrated_flow

% Argument aliases

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type SS
[Flow] SS external,external
[Integrated_flow] SS external,external
```

Subsystems

No subsystems.

10.2 Isentropic_struc.tex *(-o -ss)*

MTT command:

```
mtt -o -ss Isentropic struc tex
```

List of inputs for system Isentropic			
	Component	System	Repetition
1	Work	Isentropic__Work	1

List of outputs for system Isentropic			
	Component	System	Repetition
1	S	Isentropic__cycle__S	1
2	T	Isentropic__cycle__T	1
3	V	Isentropic__cycle__V	1
4	P	Isentropic__cycle__P	1

List of states for system Isentropic			
	Component	System	Repetition
1	c	Isentropic__cycle__gas__c	1
2	c	Isentropic__cycle__gas__c_2	1
3	mttC	Isentropic__cycle__entropy__mttC	1
4	mttC	Isentropic__cycle__volume__mttC	1

10.3 Isentropic_ode.tex *(-o -ss)*

MTT command:

```
mtt -o -ss Isentropic ode tex
```

$$\begin{aligned}
\dot{x}_1 &= \frac{(u_1 x_1 (-\gamma_g + 1))}{x_2} \\
\dot{x}_2 &= u_1 \\
\dot{x}_3 &= 0 \\
\dot{x}_4 &= u_1
\end{aligned} \tag{10.1}$$

$$y_1 = x_3$$
$$y_2 = \frac{x_1}{(c_v m_g)}$$
$$y_3 = x_4 \tag{10.2}$$
$$y_4 = \frac{(x_1(\gamma_g - 1))}{x_2}$$

10.4 Isentropic_ss.tex *(-o -ss)*

MTT command:

```
mtt -o -ss Isentropic ss tex
```

$$x = \begin{pmatrix} \frac{100000}{(\gamma_g-1)} \\ 1 \\ \frac{1000}{(3(\gamma_g-1))} \\ 1 \end{pmatrix} \tag{10.3}$$

$$u = \begin{pmatrix} 0 \end{pmatrix} \tag{10.4}$$

$$y = \begin{pmatrix} \frac{1000}{(3(\gamma_g-1))} \\ \frac{100000}{(c_v m_g(\gamma_g-1))} \\ 1 \\ 100000 \end{pmatrix} \tag{10.5}$$

$$\dot{x} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{10.6}$$

10.5 Isentropic_numpar.txt *(-o -ss)*

MTT command:

```
mtt -o -ss Isentropic numpar txt
```

```
# Numerical parameter file (Isentropic_numpar.txt)
# Generated by MTT at Thu Dec  4 11:44:46 GMT 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: Isentropic_numpar.txt,v 1.1 2000/12/28 18:16:47 peterg Exp $
# %% $Log: Isentropic_numpar.txt,v $
# %% Revision 1.1  2000/12/28 18:16:47  peterg
# %% To RCS
# %%
# %% Revision 1.1  1998/03/04 11:45:49  peterg
# %% Initial revision
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Initial states -- needed to choose an appropppriate mass
P_0 = 1e5;
V_0 = 1;
T_0 = 300;

# Parameters
c_v =  718.0; # Parameter c_v for CU
gamma_g = 1.4; # Parameter gamma for CU
m_g =  P_0*V_0/(T_0*(gamma_g-1)*c_v); # Parameter m for CU
```

10.6 Isentropic_input.txt *(-o -ss)*

MTT command:

```
mtt -o -ss Isentropic input txt
```

```
# Numerical parameter file (Isentropic_input.txt)
```

```
# Generated by MTT at Thu Dec  4 11:17:09 GMT 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: Isentropic_input.txt,v 1.2 2003/08/06 18:54:03 gawthrop
# %% $Log: Isentropic_input.txt,v $
# %% Revision 1.2  2003/08/06 18:54:03  gawthrop
# %% Updated for latest MTT version.
# %%
# %% Revision 1.1  2000/12/28 18:16:47  peterg
# %% To RCS
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Set the inputs
## Removed by MTT on Wed Aug  6 10:14:17 BST 2003:   u(1) = -0.8;
isentropic__work = -0.8;     # Volume rate-of-change
```

10.7 Isentropic_odeso.ps *(-o -ss -Isentropic__cycle__V)*

MTT command:

```
mtt -o -ss Isentropic odeso ps 'Isentropic__cycle__V'
```

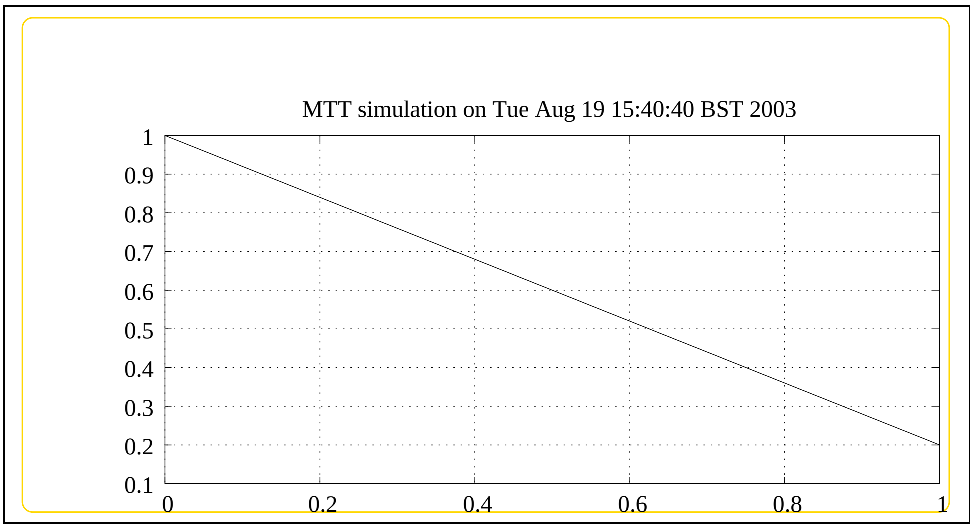
This representation is given as Figure 10.5 (on page 251).

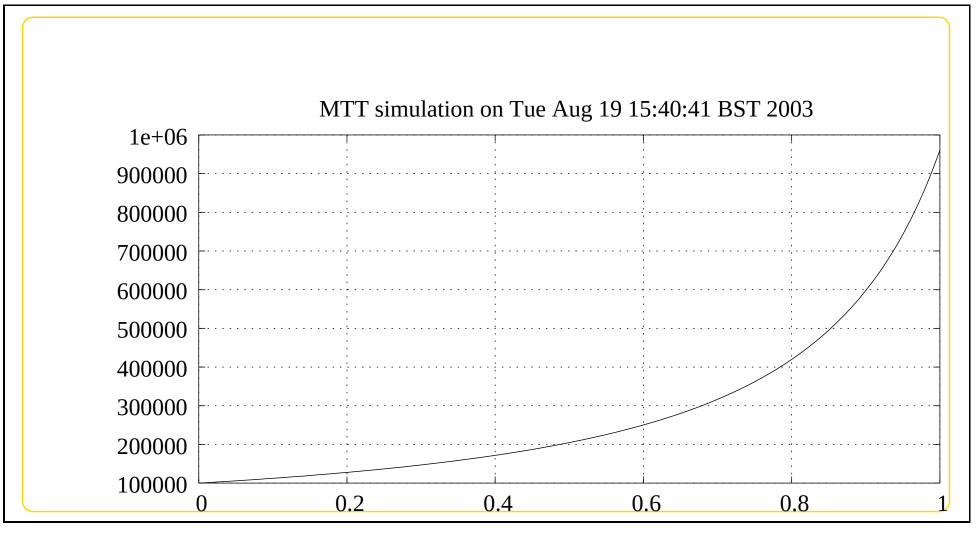
10.8 Isentropic_odeso.ps *(-o -ss -Isentropic__cycle__P)*

MTT command:

```
mtt -o -ss Isentropic odeso ps 'Isentropic__cycle__P'
```

This representation is given as Figure 10.6 (on page 251).


MTT simulation on Tue Aug 19 15:40:40 BST 2003
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.2
0.4
0.6
0.8
1

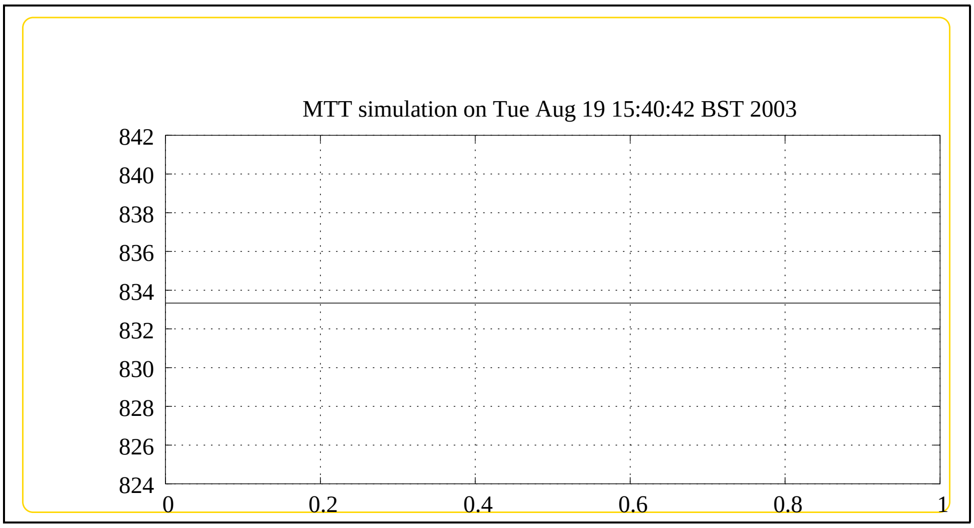

MTT simulation on Tue Aug 19 15:40:41 BST 2003
1e+06
900000
800000
700000
600000
500000
400000
300000
200000
100000
0
0.2
0.4
0.6
0.8
1

10.9 Isentropic_odeso.ps *(-o -ss -Isentropic__cycle__S)*

MTT command:

```
mtt -o -ss Isentropic odeso ps 'Isentropic__cycle__S'
```

This representation is given as Figure 10.7 (on page 252).



10.10 Isentropic_odeso.ps *(-o -ss -Isentropic__cycle__T)*

MTT command:

```
mtt -o -ss Isentropic odeso ps 'Isentropic__cycle__T'
```

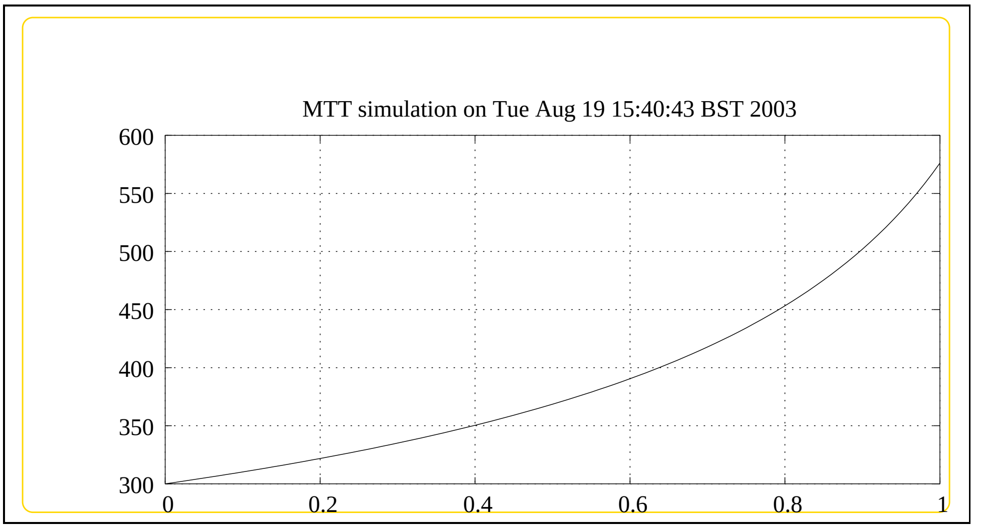
This representation is given as Figure 10.8 (on page 253).

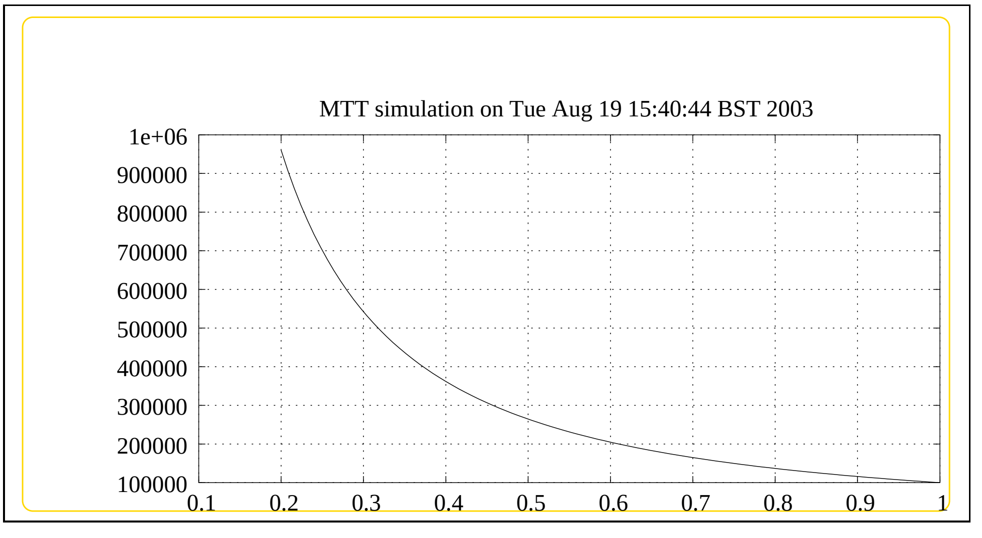
10.11 Isentropic_odeso.ps *(-o -ss -Isentropic__cycle__V:Isentropic__cycle__P)*

MTT command:

```
mtt -o -ss Isentropic odeso ps 'Isentropic__cycle__V:Isentropic__c
```

This representation is given as Figure 10.9 (on page 253).


MTT simulation on Tue Aug 19 15:40:43 BST 2003
600
550
500
450
400
350
300
0
0.2
0.4
0.6
0.8
1

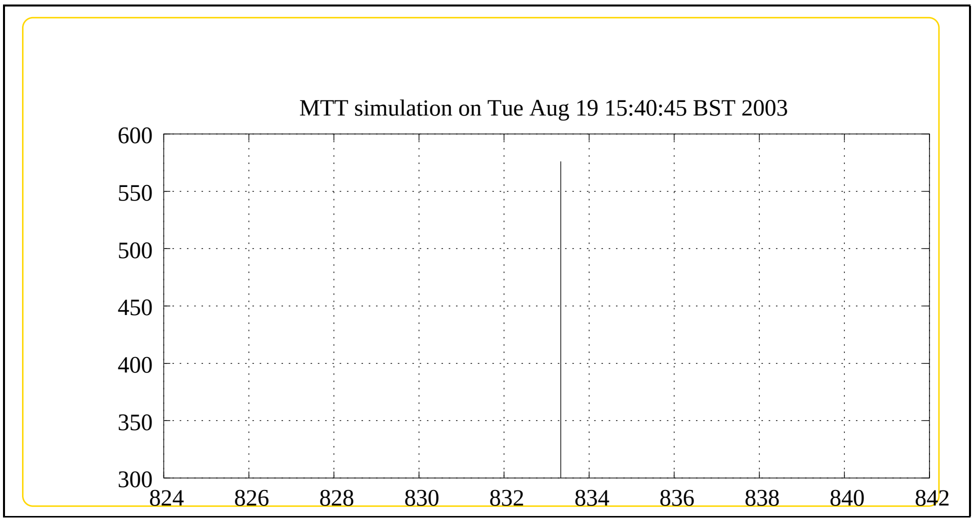

MTT simulation on Tue Aug 19 15:40:44 BST 2003
1e+06
900000
800000
700000
600000
500000
400000
300000
200000
100000
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

10.12 Isentropic_odeso.ps *(-o -ss -Isentropic__cycle__S:Isentropic__cycle__T)*

MTT command:

```
mtt -o -ss Isentropic odeso ps 'Isentropic__cycle__S:Isentropic__c
```

This representation is given as Figure 10.10 (on page 254).



Chapter 11

Isobaric

11.1 Isobaric_abg.tex *(-ss)*

MTT command:

```
mtt -ss Isobaric abg tex
```

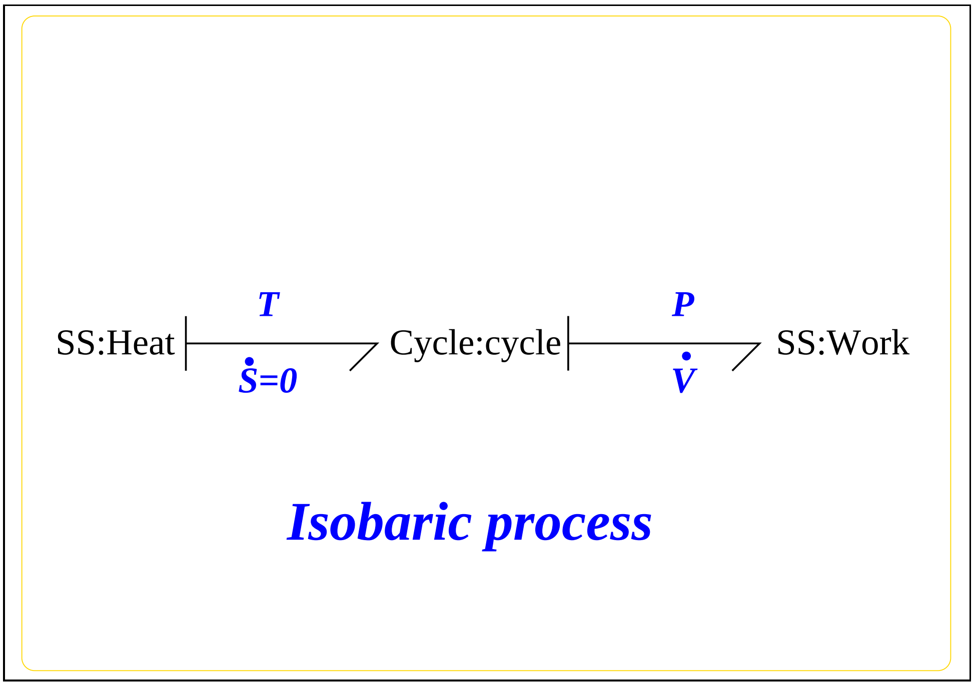


Figure 11.1: System **Isobaric**: acausal bond graph

11.1.1 Summary information

System Isobaric::Isobaric thermodynamic process - ideal gas A dynamic simulation of an isobaric (constant pressure) process using the Cycle component and the two-port CU component.

Interface information:

This component has no ALIAS declarations

Variable declarations:

P_0

T_0

V_0

Units declarations:

This component has no UNITs declarations

The label file: Isobaric_lbl.txt

```
#SUMMARY Isobaric: Isobaric thermodynamic process - ideal gas
#DESCRIPTION A dynamic simulation of an isobaric (constant pressur
#DESCRIPTION  process using the Cycle component and the two-port C

#PAR P_0
#PAR T_0
#PAR V_0

#NOTPAR ideal_gas

## Label file for system Isobaric (Isobaric_lbl.txt)

# ###############################################################
# ## Version control history
# ###############################################################
# ## $Id: Isobaric_lbl.txt,v 1.3 2003/08/06 18:54:38 gawthrop Exp
# ## $Log: Isobaric_lbl.txt,v $
# ## Revision 1.3  2003/08/06 18:54:38  gawthrop
```

```
# ## Updated for latest MTT version.
# ##
# ## Revision 1.2  2000/12/28 18:17:13  peterg
# ## To RCS
# ##
# ## Revision 1.1  1998/07/21 14:32:49  peterg
# ## Initial revision
# ##
# ###################################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank



# Component type Cycle
cycle none ideal_gas;c_v;gamma_g;m_g

# Component type SS
Heat SS internal,external
Work SS P_0,internal
```

11.1.2 Subsystems

- (1)
  - CU (1)
  - INTF: flow integrator (2)

11.1.3 CU

The acausal bond graph of system **CU** is displayed in Figure 13.2 (on page 294) and its label file is listed in Section 13.1.3 (on page 294). The subsystems are listed in Section 13.1.3 (on page 295).

Summary information

System CU: ¡Detailed description here¿

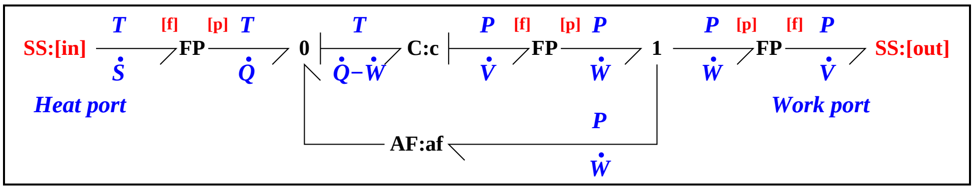


Figure 11.2: System **CU**: acausal bond graph

Interface information:

This component has no ALIAS declarations

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: CU_lbl.txt

```
%SUMMARY CU
%DESCRIPTION <Detailed description here>
%% Label file for system CU (CU_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: CU_lbl.txt,v 1.1 2000/12/28 10:34:56 peterg Exp $
% %% $Log: CU_lbl.txt,v $
% %% Revision 1.1  2000/12/28 10:34:56  peterg
% %% Put under RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank
```

```
% Component type AF (gain of -1)
af lin -1

% Component type C
c CU $1

% Component type SS
[in] SS external,external
[out] SS external,external
```

Subsystems

No subsystems.

11.1.4 Cycle

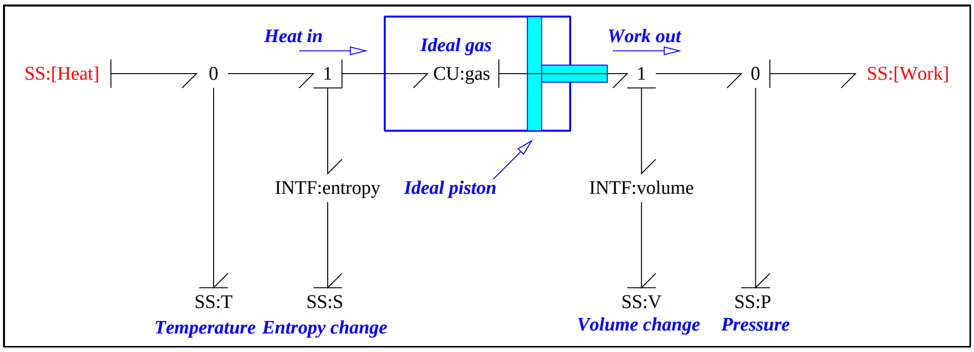


Figure 11.3: System **Cycle**: acausal bond graph

The acausal bond graph of system **Cycle** is displayed in Figure 13.3 (on page 295) and its label file is listed in Section 13.1.4 (on page 296). The subsystems are listed in Section 13.1.4 (on page 299).

Summary information

System Cycle::Closed cycle with ideal gas Uses the CU two-port thermal capacitor.

Interface information:

Parameter $1 represents actual parameter **ideal_gas**

Parameter $2 represents actual parameter **c_v**

Parameter $3 represents actual parameter **gamma_g**

Parameter $4 represents actual parameter **m_g**

Port in represents actual port **Heat**

Port out represents actual port **Work**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Cycle_lbl.txt

```
#SUMMARY Cycle: Closed cycle with ideal gas
#DESCRIPTION Uses the CU two-port thermal capacitor.

#ALIAS in Heat
#ALIAS out Work

#ALIAS $1 ideal_gas
#ALIAS $2 c_v
#ALIAS $3 gamma_g
#ALIAS $4 m_g

## Label file for system Cycle (Cycle_lbl.txt)

# ###################################################################
# ## Version control history
# ###################################################################
# ## $Id: Cycle_lbl.txt,v 1.1 2000/12/28 18:17:13 peterg Exp $
# ## $Log: Cycle_lbl.txt,v $
# ## Revision 1.1  2000/12/28 18:17:13  peterg
# ## To RCS
# ##
# ## Revision 1.2  1998/07/21 14:21:04  peterg
# ## New style file
```

```
# ##
# ## Revision 1.1  1997/12/08 20:24:43  peterg
# ## Initial revision
# ##
# ## Revision 1.1  1997/12/07 20:38:05  peterg
# ## Initial revision
# ##
# ###############################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank




# Component type CU
gas CU ideal_gas,c_v,gamma_g,m_g

# Component type INTF
entropy
volume

# Component type SS
S SS external,0
T SS external,0
V SS external,0
P SS external,0
[Heat] SS external,external
[Work] SS external,external
```

Subsystems

- CU (1) No subsystems.
- INTF: flow integrator (2) No subsystems.

11.1.5 INTF

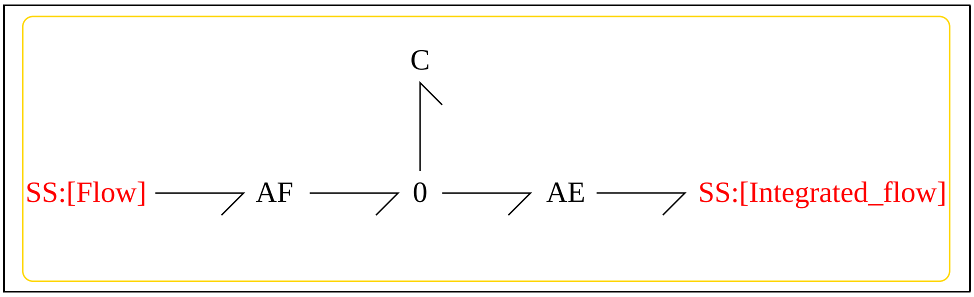


Figure 11.4: System **INTF**: acausal bond graph

The acausal bond graph of system **INTF** is displayed in Figure 13.4 (on page 299) and its label file is listed in Section 13.1.5 (on page 299). The subsystems are listed in Section 13.1.5 (on page 301).
INTF is a two-port component where the effort on port [out] is the integral of the flow on port [in].

Summary information

System INTF::flow integrator Port [in]: Flow to be integrated Port [out]: Effort = integral of flow on port [in]

Interface information:

Port in represents actual port **Flow**

Port out represents actual port **Integrated_flow**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: INTF_lbl.txt

```
%% Label file for system INTF (INTF_lbl.txt)
%SUMMARY INTF: flow integrator
%DESCRIPTION Port [in]: Flow to be integrated
%DESCRIPTION Port [out]: Effort = integral of flow on port [in]


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: INTF_lbl.txt,v 1.3 1998/07/16 07:35:10 peterg Exp $
% %% $Log: INTF_lbl.txt,v $
% %% Revision 1.3  1998/07/16 07:35:10  peterg
% %% Aliased version
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in Flow
%ALIAS out Integrated_flow

% Argument aliases

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type SS
[Flow] SS external,external
[Integrated_flow] SS external,external
```

Subsystems

No subsystems.

11.2 Isobaric_struc.tex *(-ss)*

MTT command:

```
mtt -ss Isobaric struc tex
```

List of inputs for system Isobaric			
	Component	System	Repetition
1	Heat	Isobaric__Heat	1

List of nonstates for system Isobaric			
	Component	System	Repetition
1	c	Isobaric__cycle__gas__c	1

List of outputs for system Isobaric			
	Component	System	Repetition
1	S	Isobaric__cycle__S	1
2	T	Isobaric__cycle__T	1
3	V	Isobaric__cycle__V	1
4	P	Isobaric__cycle__P	1

List of states for system Isobaric			
	Component	System	Repetition
1	c	Isobaric__cycle__gas__c	1
2	mttC	Isobaric__cycle__entropy__mttC	1
3	mttC	Isobaric__cycle__volume__mttC	1

11.3 Isobaric_ode.tex *(-ss)*

MTT command:

```
mtt -ss Isobaric ode tex
```

$$\begin{aligned} \dot{x}_1 &= \frac{(u_1 x_1)}{(c_v \gamma_g^2 m_g)} \\ \dot{x}_2 &= u_1 \\ \dot{x}_3 &= \frac{(u_1 x_1 (\gamma_g - 1))}{(c_v \gamma_g m_g p_0)} \end{aligned} \tag{11.1}$$

$$
\begin{aligned}
y_1 &= x_2 \\
y_2 &= \frac{x_1}{(c_v m_g)} \\
y_3 &= x_3 \\
y_4 &= p_0
\end{aligned} \tag{11.2}
$$

11.4 Isobaric_ss.tex *(-ss)*

MTT command:

```
mtt -ss Isobaric ss tex
```

$$
x = \begin{pmatrix} \frac{100000}{(\gamma_g - 1)} \\ \frac{1000}{(3(\gamma_g - 1))} \\ 1 \end{pmatrix} \tag{11.3}
$$

$$
u = \begin{pmatrix} 0 \end{pmatrix} \tag{11.4}
$$

$$
y = \begin{pmatrix} \frac{1000}{(3(\gamma_g - 1))} \\ \frac{100000}{(c_v m_g (\gamma_g - 1))} \\ 1 \\ 100000 \end{pmatrix} \tag{11.5}
$$

$$
\dot{x} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{11.6}
$$

11.5 Isobaric_numpar.txt *(-ss)*

MTT command:

```
mtt -ss Isobaric numpar txt
```

```
# Numerical parameter file (Isobaric_numpar.txt)
# Generated by MTT at Thu Dec  4 11:44:46 GMT 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: Isobaric_numpar.txt,v 1.1 2000/12/28 18:17:13 peterg Exp
# %% $Log: Isobaric_numpar.txt,v $
# %% Revision 1.1  2000/12/28 18:17:13  peterg
# %% To RCS
# %%
# %% Revision 1.1  1998/03/04 11:45:49  peterg
# %% Initial revision
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Initial states -- needed to choose an approppriate mass
P_0 = 1e5;
V_0 = 1;
T_0 = 300;

# Parameters
c_v =  718.0; # Parameter c_v for CU
gamma_g = 1.4; # Parameter gamma_g for CU
m_g =  P_0*V_0/(T_0*(gamma_g-1)*c_v); # Parameter m for CU
```

11.6 Isobaric_input.txt *(-ss)*

MTT command:

```
mtt -ss Isobaric input txt
```

```
# Numerical parameter file (Isobaric_input.txt)
# Generated by MTT at Thu Dec  4 11:17:09 GMT 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: Isobaric_input.txt,v 1.2 2003/08/06 18:54:32 gawthrop Ex
```

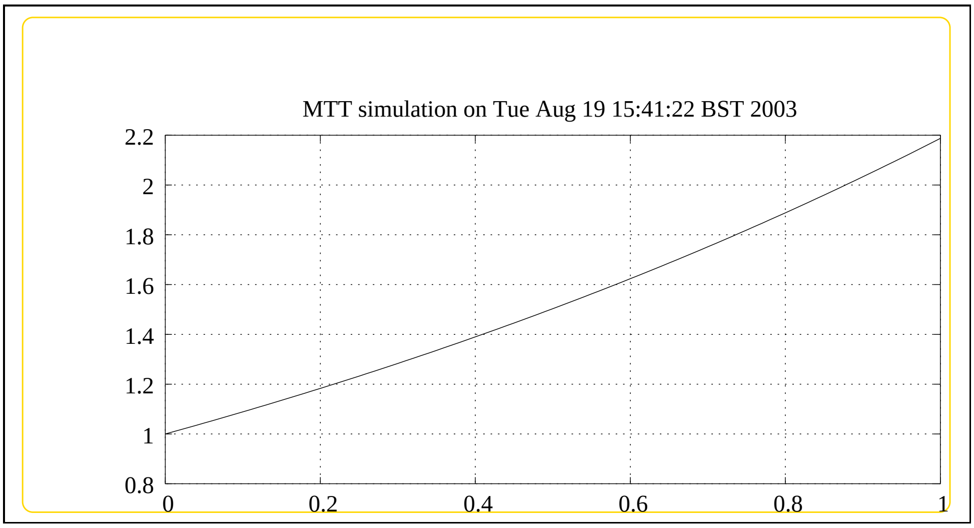
```
# %% $Log: Isobaric_input.txt,v $
# %% Revision 1.2  2003/08/06 18:54:32  gawthrop
# %% Updated for latest MTT version.
# %%
# %% Revision 1.1  2000/12/28 18:17:13  peterg
# %% To RCS
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Set the inputs
## Removed by MTT on Wed Aug  6 11:41:46 BST 2003:   u(1) = 1000;     #E
isobaric__heat = 1000;     #Entropy flow
```

11.7 Isobaric_odeso.ps *(-ss -Isobaric__cycle__V)*

MTT command:

```
mtt -ss Isobaric odeso ps 'Isobaric__cycle__V'
```

This representation is given as Figure 11.5 (on page 267).

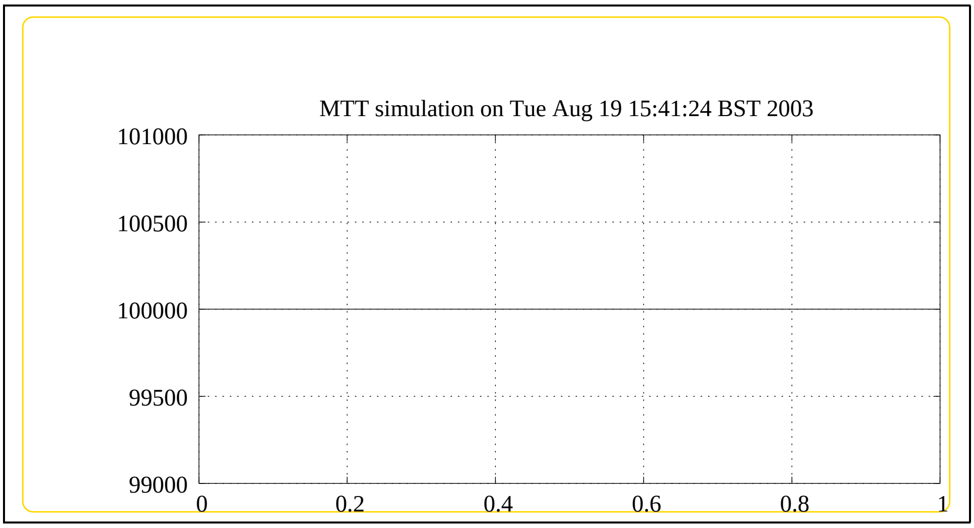


11.8 Isobaric_odeso.ps *(-ss -Isobaric__cycle__P)*

MTT command:

```
mtt -ss Isobaric odeso ps 'Isobaric__cycle__P'
```

This representation is given as Figure 11.6 (on page 268).



11.9 Isobaric_odeso.ps *(-ss -Isobaric__cycle__S)*

MTT command:

```
mtt -ss Isobaric odeso ps 'Isobaric__cycle__S'
```

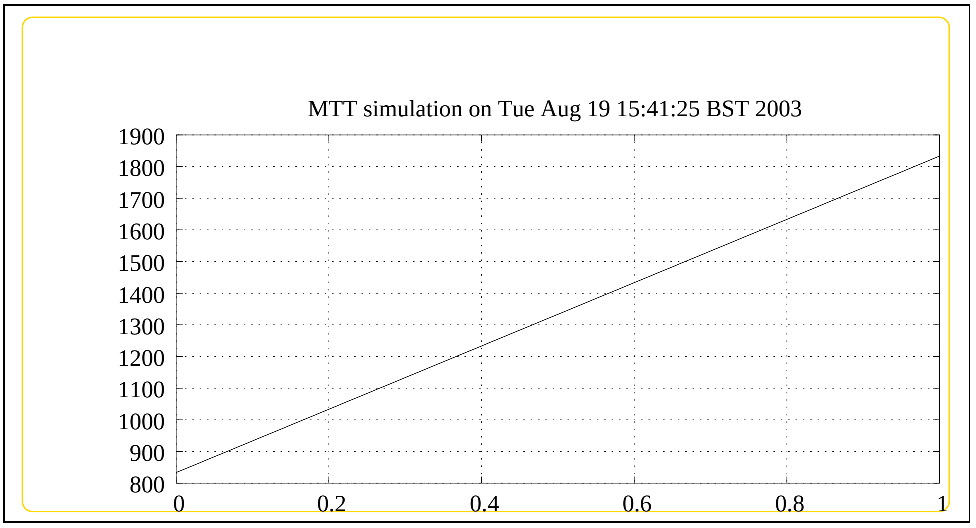
This representation is given as Figure 11.7 (on page 269).

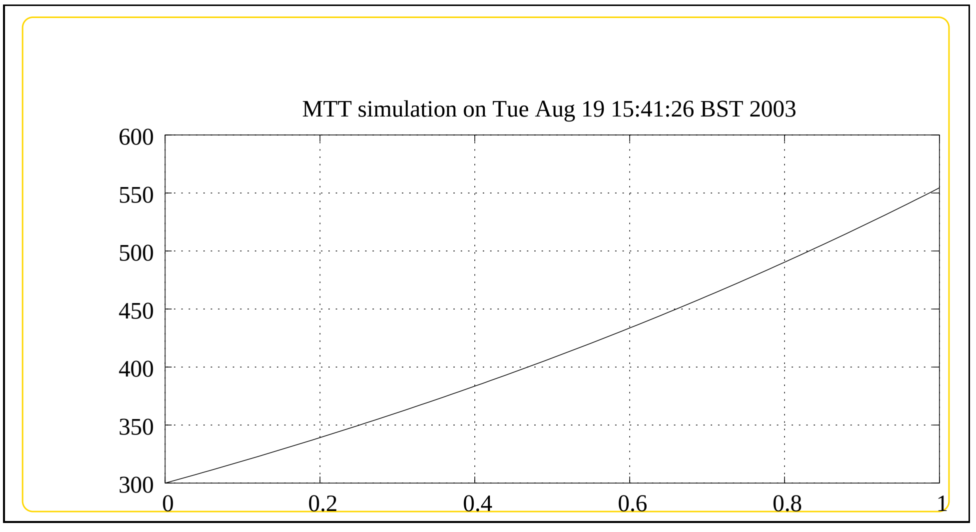
11.10 Isobaric_odeso.ps *(-ss -Isobaric__cycle__T)*

MTT command:

```
mtt -ss Isobaric odeso ps 'Isobaric__cycle__T'
```

This representation is given as Figure 11.8 (on page 269).


MTT simulation on Tue Aug 19 15:41:25 BST 2003
1900
1800
1700
1600
1500
1400
1300
1200
1100
1000
900
800
0
0.2
0.4
0.6
0.8
1

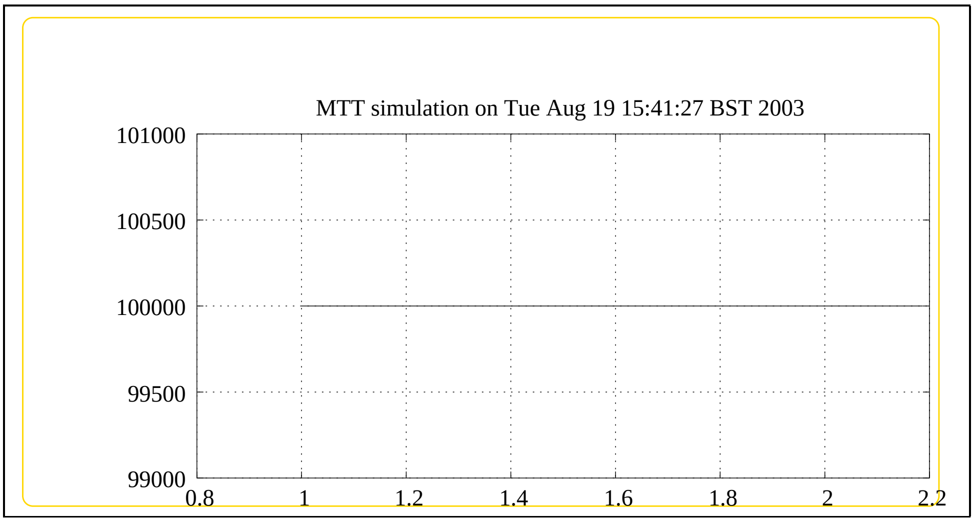

MTT simulation on Tue Aug 19 15:41:26 BST 2003
600
550
500
450
400
350
300
0
0.2
0.4
0.6
0.8
1

11.11 Isobaric_odeso.ps (*-ss -Isobaric__cycle__V:Isobaric__cycle__P)*

MTT command:

```
mtt -ss Isobaric odeso ps 'Isobaric__cycle__V:Isobaric__cycle__P'
```

This representation is given as Figure 11.9 (on page 270).

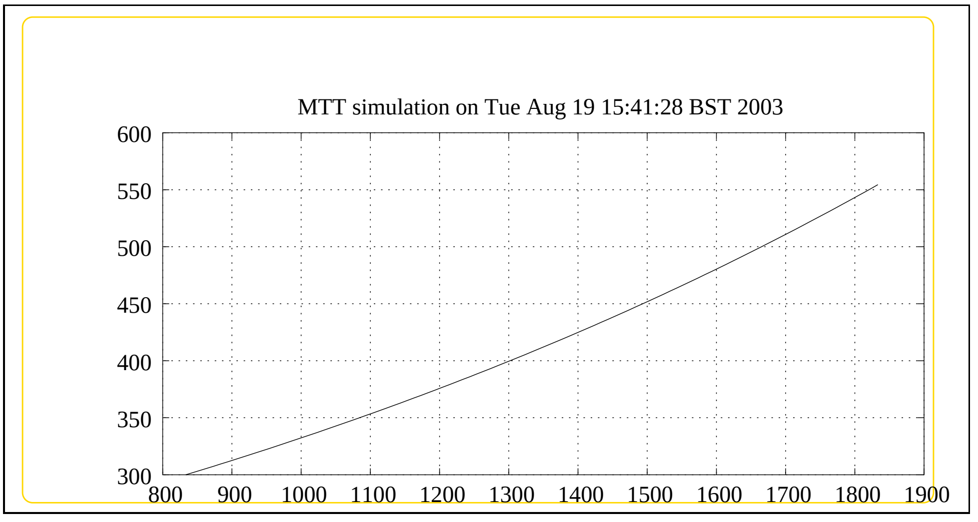


11.12 Isobaric_odeso.ps (*-ss -Isobaric__cycle__S:Isobaric__cycle__T)*

MTT command:

```
mtt -ss Isobaric odeso ps 'Isobaric__cycle__S:Isobaric__cycle__T'
```

This representation is given as Figure 11.10 (on page 271).


MTT simulation on Tue Aug 19 15:41:28 BST 2003
600
550
500
450
400
350
300
800
900
1000
1100
1200
1300
1400
1500
1600
1700
1800
1900

Chapter 12

Isothermal

12.1 Isothermal_abg.tex *(-ss)*

MTT command:

```
mtt -ss Isothermal abg tex
```

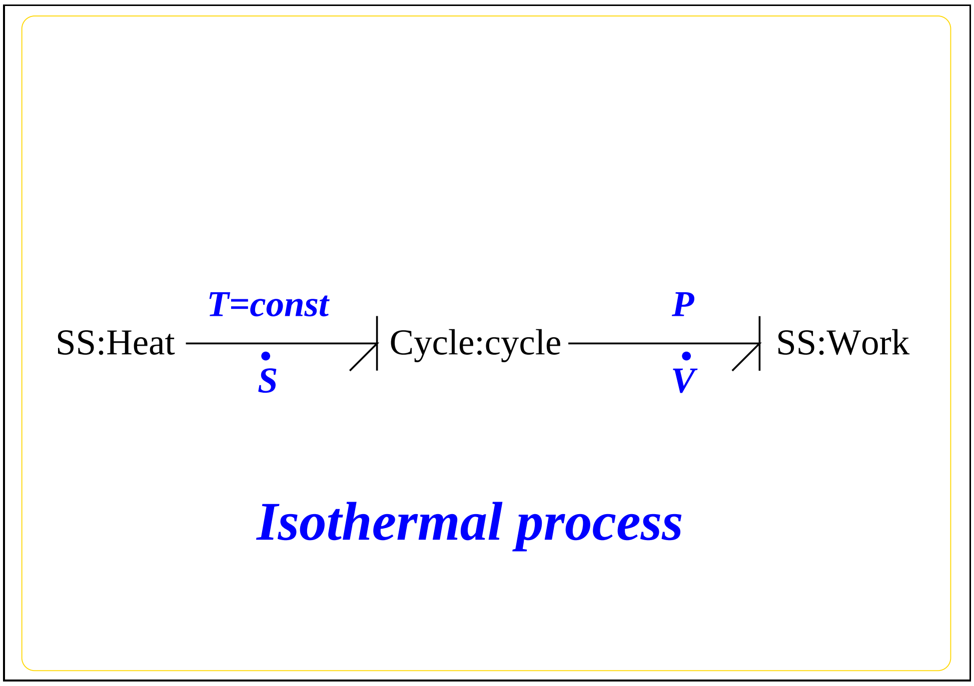


Figure 12.1: System **Isothermal**: acausal bond graph

12.1.1 Summary information

System Isothermal::Isothermal thermodynamic process - ideal gas A dynamic simulation of an isothermal process using the Cycle component and the two-port CU component.

Interface information:

This component has no ALIAS declarations

Variable declarations:

P_0

T_0

V_0

Units declarations:

This component has no UNITs declarations

The label file: Isothermal_lbl.txt

```
#SUMMARY Isothermal: Isothermal thermodynamic process - ideal gas
#DESCRIPTION A dynamic simulation of an isothermal process using
#DESCRIPTION the Cycle component and the two-port CU component.

#PAR P_0
#PAR T_0
#PAR V_0

#NOTPAR ideal_gas

## Label file for system Isothermal (Isothermal_lbl.txt)

# #################################################################
# ## Version control history
# #################################################################
# ## $Id: Isothermal_lbl.txt,v 1.3 2003/08/06 18:54:56 gawthrop Ex
# ## $Log: Isothermal_lbl.txt,v $
# ## Revision 1.3  2003/08/06 18:54:56  gawthrop
```

```
# ## Updated for latest MTT version.
# ##
# ## Revision 1.2  2000/12/28 18:17:37  peterg
# ## To RCS
# ##
# ## Revision 1.1  1998/07/21 14:30:29  peterg
# ## Initial revision
# ##
# ###############################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank



# Component type Cycle
cycle none ideal_gas;c_v;gamma_g;m_g

# Component type SS
Heat SS T_0,internal
Work SS internal,external
```

12.1.2 Subsystems

- (1)
  - CU (1)
  - INTF: flow integrator (2)

12.1.3 CU

The acausal bond graph of system **CU** is displayed in Figure 13.2 (on page 294) and its label file is listed in Section 13.1.3 (on page 294). The subsystems are listed in Section 13.1.3 (on page 295).

Summary information

System CU: ¡Detailed description here¿

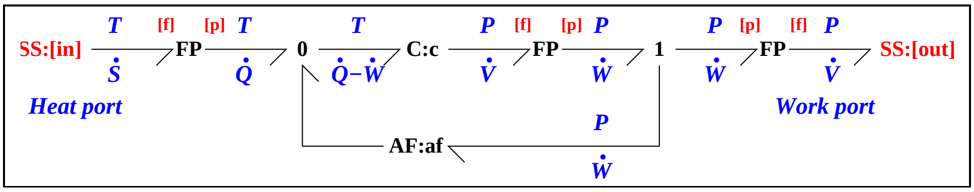


Figure 12.2: System **CU**: acausal bond graph

Interface information:

This component has no ALIAS declarations

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: CU_lbl.txt

```
%SUMMARY CU
%DESCRIPTION <Detailed description here>
%% Label file for system CU (CU_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: CU_lbl.txt,v 1.1 2000/12/28 10:34:56 peterg Exp $
% %% $Log: CU_lbl.txt,v $
% %% Revision 1.1  2000/12/28 10:34:56  peterg
% %% Put under RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank
```

```
% Component type AF (gain of -1)
af lin -1

% Component type C
c CU $1

% Component type SS
[in] SS external,external
[out] SS external,external
```

Subsystems

No subsystems.

12.1.4 Cycle

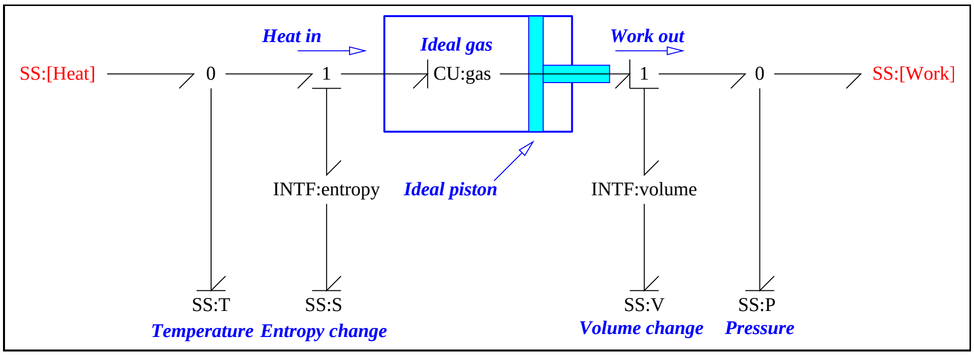


Figure 12.3: System **Cycle**: acausal bond graph

The acausal bond graph of system **Cycle** is displayed in Figure 13.3 (on page 295) and its label file is listed in Section 13.1.4 (on page 296). The subsystems are listed in Section 13.1.4 (on page 299).

The system has two heat engine ports:

1. **[Heat]** and

2. **[Work]**

By convention, energy flows in to the **[Heat]** port and out of the **[Work]** port. Both ports are true energy ports.

The subsystem **CU** (Section 13.1.3 (on page 293)) is a two-port component describing an ideal gas. It has two energy ports which, with integral causality correspond to

1. Entropy flow in; temperature out
2. Volume rate of change in; pressure out

Four **SS** components are set up as sensors to measure the important quantities pertaining to the cycle:

1. **S** to measure the integrated entropy flow
2. **T** to measure the (absolute) temperature
3. **V** to measure the integrated volume change
4. **P** to measure the pressure

Cycle	Compression	Heating	Expansion	Cooling
Otto	II	II	II	II
Carnot	II	DI	II	DI
Diesel	II	ID	II	II
Joule	II	ID	II	ID

Table 12.1: Cycles and their causality

A number of cycles can be built depending on the causality of the two ports **[Heat]** and **[Work]** of **Cycle** Some possible cycles listed in Tables 13.1 (on page 296) where each table entry gives the causality on the heat and work ports respectively.

Summary information

System Cycle::Closed cycle with ideal gas Uses the CU two-port thermal capacitor.

Interface information:

Parameter $1 represents actual parameter **ideal_gas**

Parameter $2 represents actual parameter **c_v**

Parameter $3 represents actual parameter **gamma_g**

Parameter $4 represents actual parameter **m_g**

Port in represents actual port **Heat**

Port out represents actual port **Work**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Cycle_lbl.txt

```
#SUMMARY Cycle: Closed cycle with ideal gas
#DESCRIPTION Uses the CU two-port thermal capacitor.

#ALIAS in Heat
#ALIAS out Work

#ALIAS $1 ideal_gas
#ALIAS $2 c_v
#ALIAS $3 gamma_g
#ALIAS $4 m_g

## Label file for system Cycle (Cycle_lbl.txt)

# ###############################################################
# ## Version control history
# ###############################################################
# ## $Id: Cycle_lbl.txt,v 1.1 2000/12/28 18:17:37 peterg Exp $
# ## $Log: Cycle_lbl.txt,v $
# ## Revision 1.1  2000/12/28 18:17:37  peterg
# ## To RCS
# ##
# ## Revision 1.2  1998/07/21 14:21:04  peterg
# ## New style file
# ##
# ## Revision 1.1  1997/12/08 20:24:43  peterg
# ## Initial revision
# ##
# ## Revision 1.1  1997/12/07 20:38:05  peterg
# ## Initial revision
# ##
# ###############################################################
```

```
## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank




# Component type CU
gas CU ideal_gas,c_v,gamma_g,m_g

# Component type INTF
entropy
volume

# Component type SS
S SS external,0
T SS external,0
V SS external,0
P SS external,0
[Heat] SS external,external
[Work] SS external,external
```

Subsystems

- CU (1) No subsystems.
- INTF: flow integrator (2) No subsystems.

12.1.5 INTF

The acausal bond graph of system **INTF** is displayed in Figure 13.4 (on page 299) and its label file is listed in Section 13.1.5 (on page 299). The subsystems are listed in Section 13.1.5 (on page 301).

INTF is a two-port component where the effort on port [out] is the integral of the flow on port [in].

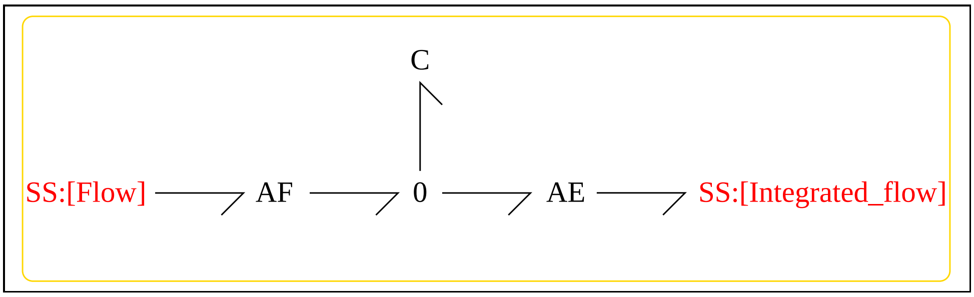


Figure 12.4: System **INTF**: acausal bond graph

Summary information

System INTF::flow integrator Port [in]: Flow to be integrated Port [out]: Effort = integral of flow on port [in]

Interface information:

Port in represents actual port **Flow**

Port out represents actual port **Integrated_flow**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: INTF_lbl.txt

```
%% Label file for system INTF (INTF_lbl.txt)
%SUMMARY INTF: flow integrator
%DESCRIPTION Port [in]: Flow to be integrated
%DESCRIPTION Port [out]: Effort = integral of flow on port [in]


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: INTF_lbl.txt,v 1.3 1998/07/16 07:35:10 peterg Exp $
```

```
% %% $Log: INTF_lbl.txt,v $
% %% Revision 1.3  1998/07/16 07:35:10  peterg
% %% Aliased version
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in Flow
%ALIAS out Integrated_flow

% Argument aliases

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type SS
[Flow] SS external,external
[Integrated_flow] SS external,external
```

Subsystems

No subsystems.

12.2 Isothermal_struc.tex *(-ss)*

MTT command:

```
mtt -ss Isothermal struc tex
```

List of inputs for system Isothermal			
	Component	System	Repetition
1	Work	Isothermal__Work	1

List of nonstates for system Isothermal			
	Component	System	Repetition
1	c	Isothermal__cycle__gas__c	1

List of outputs for system Isothermal			
	Component	System	Repetition
1	S	Isothermal__cycle__S	1
2	T	Isothermal__cycle__T	1
3	V	Isothermal__cycle__V	1

List of outputs for system Isothermal (continued)			
	Component	System	Repetition
4	P	Isothermal__cycle__P	1

List of states for system Isothermal			
	Component	System	Repetition
1	c	Isothermal__cycle__gas__c	1
2	mttC	Isothermal__cycle__entropy__mttC	1
3	mttC	Isothermal__cycle__volume__mttC	1

12.3 Isothermal_ode.tex *(-ss)*

MTT command:

```
mtt -ss Isothermal ode tex
```

$$\begin{aligned} \dot{x}_1 &= u_1 \\ \dot{x}_2 &= \frac{(c_v m_g u_1 (\gamma_g - 1))}{x_1} \\ \dot{x}_3 &= u_1 \end{aligned} \tag{12.1}$$

$$\begin{aligned} y_1 &= x_2 \\ y_2 &= t_0 \\ y_3 &= x_3 \\ y_4 &= \frac{(c_v m_g t_0 (\gamma_g - 1))}{x_1} \end{aligned} \tag{12.2}$$

12.4 Isothermal_ss.tex *(-ss)*

MTT command:

```
mtt -ss Isothermal ss tex
```

$$x = \begin{pmatrix} 1 \\ \frac{1000}{(3(\gamma_g - 1))} \\ 1 \end{pmatrix} \tag{12.3}$$

$$u = \begin{pmatrix} 0 \end{pmatrix} \tag{12.4}$$

$$y = \begin{pmatrix} \frac{1000}{(3(\gamma_g - 1))} \\ 300 \\ 1 \\ 300 c_v m_g (\gamma_g - 1) \end{pmatrix} \tag{12.5}$$

$$\dot{x} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{12.6}$$

12.5 Isothermal_numpar.txt *(-ss)*

MTT command:

```
mtt -ss Isothermal numpar txt
```

```
# Numerical parameter file (Isothermal_numpar.txt)
# Generated by MTT at Thu Dec  4 11:44:46 GMT 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: Isothermal_numpar.txt,v 1.1 2000/12/28 18:17:37 peterg Exp $
# %% $Log: Isothermal_numpar.txt,v $
# %% Revision 1.1  2000/12/28 18:17:37  peterg
# %% To RCS
# %%
# %% Revision 1.1  1998/03/04 11:45:49  peterg
# %% Initial revision
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
# Initial states -- needed to choose an approppriate mass
P_0 = 1e5;
V_0 = 1;
T_0 = 300;

# Parameters
c_v =  718.0; # Parameter c_v for CU
gamma_g = 1.4; # Parameter gamma_g for CU
m_g =  P_0*V_0/(T_0*(gamma_g-1)*c_v); # Parameter m for CU
```

12.6 Isothermal_input.txt *(-ss)*

MTT command:

```
mtt -ss Isothermal input txt
```

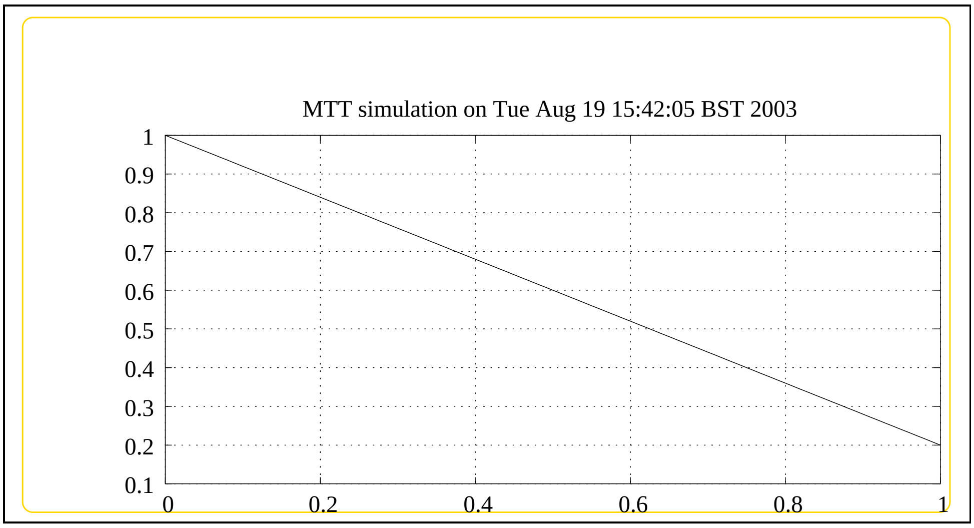
```
# Numerical parameter file (Isothermal_input.txt)
# Generated by MTT at Thu Dec  4 11:17:09 GMT 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: Isothermal_input.txt,v 1.2 2003/08/06 18:54:50 gawthrop
# %% $Log: Isothermal_input.txt,v $
# %% Revision 1.2  2003/08/06 18:54:50  gawthrop
# %% Updated for latest MTT version.
# %%
# %% Revision 1.1  2000/12/28 18:17:37  peterg
# %% To RCS
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Set the inputs
## Removed by MTT on Wed Aug  6 10:47:34 BST 2003:   u(1) = -0.8;
isothermal__work = -0.8;     # Volume rate-of-change
```

12.7 Isothermal_odeso.ps *(-ss -Isothermal__cycle__V)*

MTT command:

```
mtt -ss Isothermal odeso ps 'Isothermal__cycle__V'
```

This representation is given as Figure 12.5 (on page 287).



12.8 Isothermal_odeso.ps *(-ss -Isothermal__cycle__P)*

MTT command:

```
mtt -ss Isothermal odeso ps 'Isothermal__cycle__P'
```

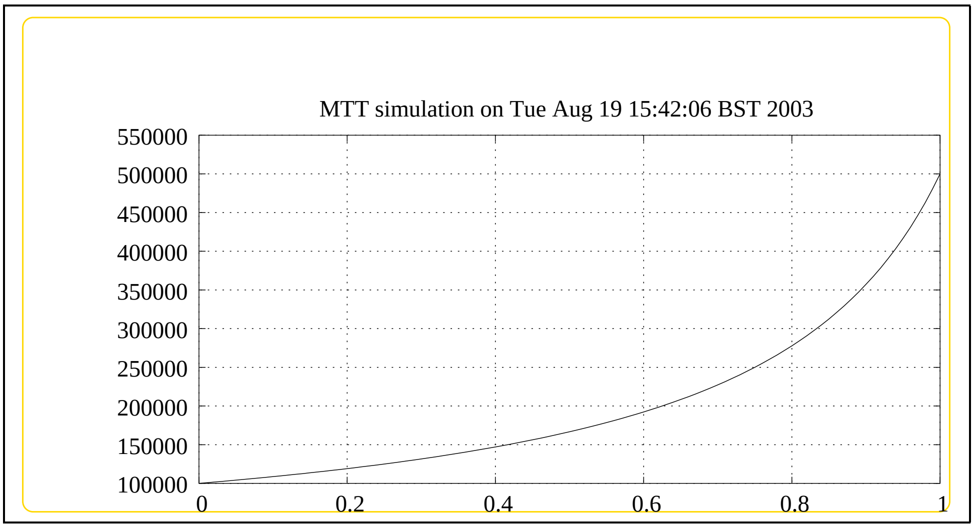
This representation is given as Figure 12.6 (on page 288).

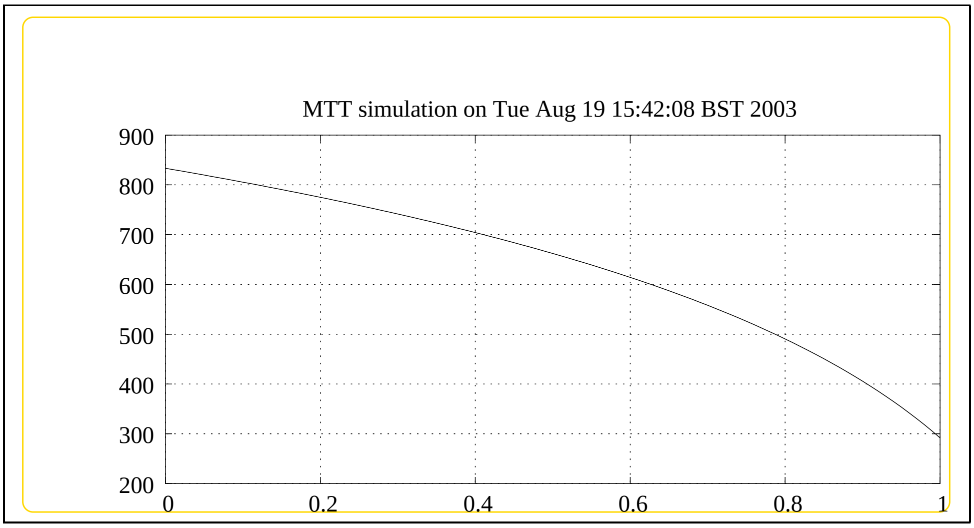
12.9 Isothermal_odeso.ps *(-ss -Isothermal__cycle__S)*

MTT command:

```
mtt -ss Isothermal odeso ps 'Isothermal__cycle__S'
```

This representation is given as Figure 12.7 (on page 288).


MTT simulation on Tue Aug 19 15:42:06 BST 2003
550000
500000
450000
400000
350000
300000
250000
200000
150000
100000
0
0.2
0.4
0.6
0.8
1

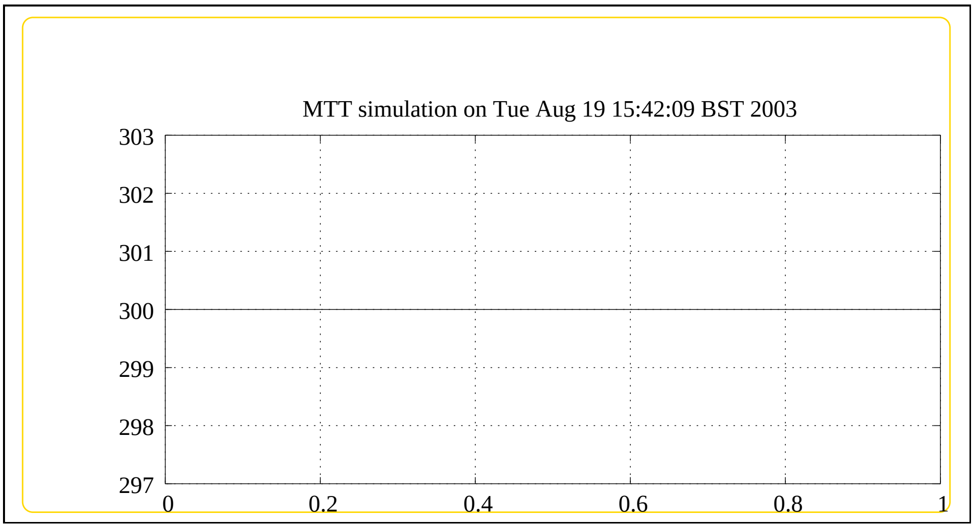

MTT simulation on Tue Aug 19 15:42:08 BST 2003
900
800
700
600
500
400
300
200
0
0.2
0.4
0.6
0.8
1

12.10 Isothermal_odeso.ps *(-ss -Isothermal__cycle__T)*

MTT command:

```
mtt -ss Isothermal odeso ps 'Isothermal__cycle__T'
```

This representation is given as Figure 12.8 (on page 289).



12.11 Isothermal_odeso.ps *(-ss -Isothermal__cycle__V:Isothermal__cycle__P)*

MTT command:

```
mtt -ss Isothermal odeso ps 'Isothermal__cycle__V:Isothermal__cycle__P'
```

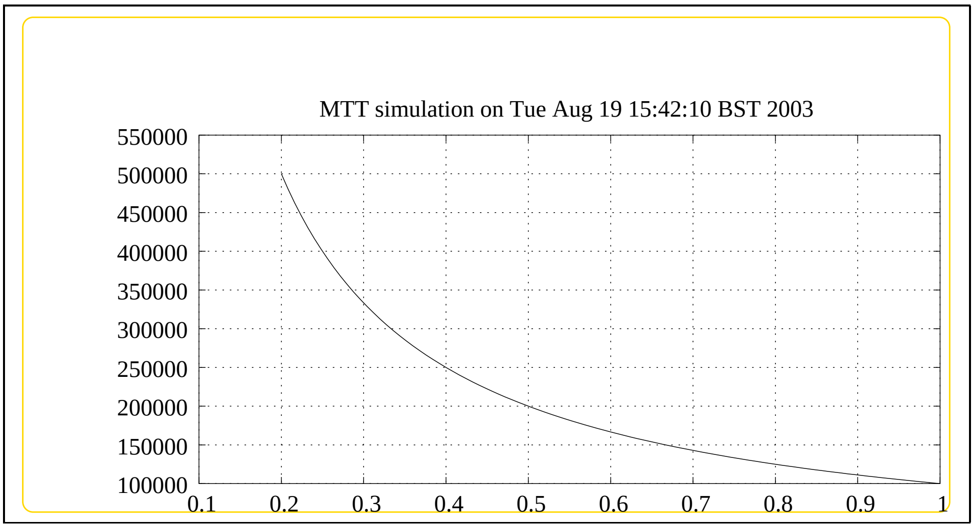
This representation is given as Figure 12.9 (on page 290).

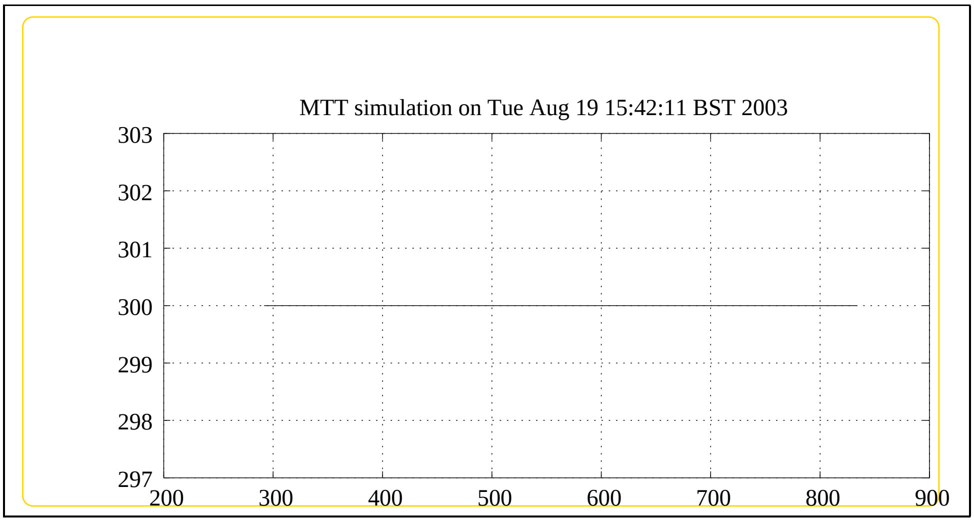
12.12 Isothermal_odeso.ps *(-ss -Isothermal__cycle__S:Isothermal__cycle__T)*

MTT command:

```
mtt -ss Isothermal odeso ps 'Isothermal__cycle__S:Isothermal__cycle__T'
```

This representation is given as Figure 12.10 (on page 290).


MTT simulation on Tue Aug 19 15:42:10 BST 2003
550000
500000
450000
400000
350000
300000
250000
200000
150000
100000
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


MTT simulation on Tue Aug 19 15:42:11 BST 2003
303
302
301
300
299
298
297
200
300
400
500
600
700
800
900

Chapter 13

Isovolumetric

13.1 Isovolumetric_abg.tex *(-o -ss)*

MTT command:

```
mtt -o -ss Isovolumetric abg tex
```

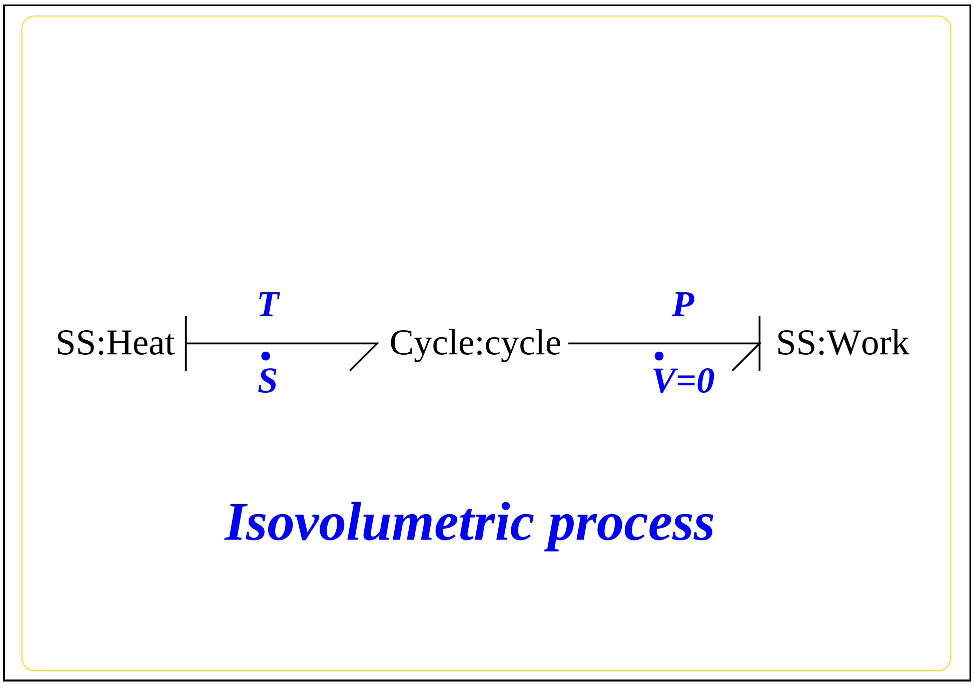


Figure 13.1: System **Isovolumetric**: acausal bond graph

13.1.1 Summary information

System Isovolumetric::Isovolumetric thermodynamic process - ideal gas
A dynamic simulation of an Isovolumetric (constant volume) process using the Cycle component and the two-port CU component.
the Cycle component and the two-port CU component.

Interface information:

This component has no ALIAS declarations

Variable declarations:

P_0

T_0

V_0

Units declarations:

This component has no UNITs declarations

The label file: Isovolumetric_lbl.txt

```
#SUMMARY Isovolumetric: Isovolumetric thermodynamic process - idea
#DESCRIPTION A dynamic simulation of an Isovolumetric (constant vo
#DESCRIPTION  process using the Cycle component and the two-port C
#DESCRIPTION
#DESCRIPTION the Cycle component and the two-port CU component.

#PAR P_0
#PAR T_0
#PAR V_0

#NOTPAR ideal_gas

## Label file for system Isovolumetric (Isovolumetric_lbl.txt)

# ###################################################################
# ## Version control history
# ###################################################################
```

```
# ## $Id: Isovolumetric_lbl.txt,v 1.3 2003/08/06 18:55:14 gawthrop Exp $
# ## $Log: Isovolumetric_lbl.txt,v $
# ## Revision 1.3  2003/08/06 18:55:14  gawthrop
# ## Updated for latest MTT version.
# ##
# ## Revision 1.2  2000/12/28 18:17:57  peterg
# ## To RCS
# ##
# ## Revision 1.1  1998/07/21 14:37:03  peterg
# ## Initial revision
# ##
# ###############################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank



# Component type Cycle
cycle none ideal_gas;c_v;gamma_g;m_g

# Component type SS
Heat SS internal,external
Work SS internal,0
```

13.1.2 Subsystems

- (1)
  - CU (1)
  - INTF: flow integrator (2)

13.1.3 CU

The acausal bond graph of system **CU** is displayed in Figure 13.2 (on page 294) and its label file is listed in Section 13.1.3 (on page 294). The subsystems are listed in Section 13.1.3 (on page 295).

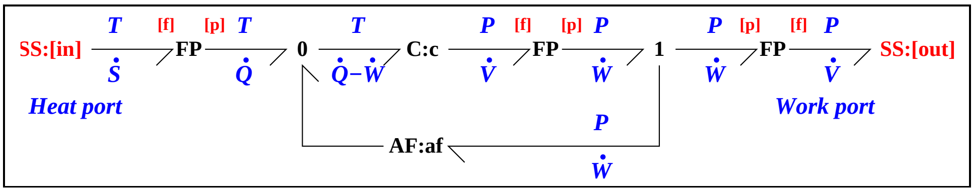


Figure 13.2: System **CU**: acausal bond graph

Summary information

System CU: ¡Detailed description here¿

Interface information:

This component has no ALIAS declarations

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: CU_lbl.txt

```
%SUMMARY CU
%DESCRIPTION <Detailed description here>
%% Label file for system CU (CU_lbl.txt)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: CU_lbl.txt,v 1.1 2000/12/28 10:34:56 peterg Exp $
% %% $Log: CU_lbl.txt,v $
% %% Revision 1.1  2000/12/28 10:34:56  peterg
% %% Put under RCS
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Each line should be of one of the following forms:
```

```
% a comment (ie starting with %)
% Component-name CR_name arg1,arg2,..argn
% blank

% Component type AF (gain of -1)
af lin -1

% Component type C
c CU $1

% Component type SS
[in] SS external,external
[out] SS external,external
```

Subsystems

No subsystems.

13.1.4 Cycle

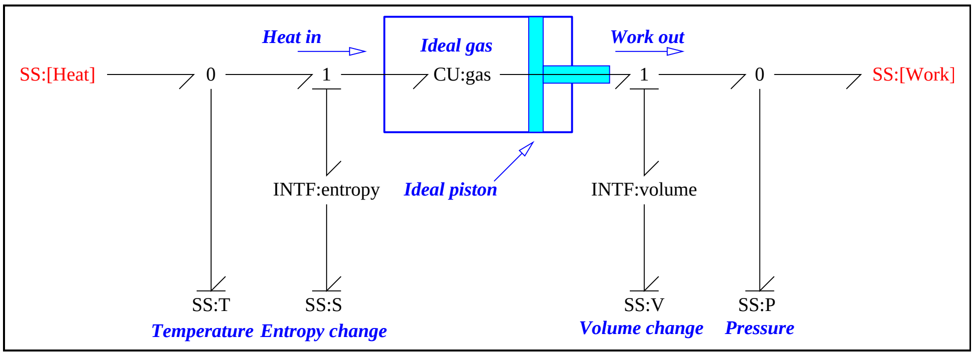


Figure 13.3: System **Cycle**: acausal bond graph

The acausal bond graph of system **Cycle** is displayed in Figure 13.3 (on page 295) and its label file is listed in Section 13.1.4 (on page 296). The subsystems are listed in Section 13.1.4 (on page 299).

The system has two heat engine ports:

1. **[Heat]** and
2. **[Work]**

By convention, energy flows in to the **[Heat]** port and out of the **[Work]** port. Both ports are true energy ports.
The subsystem **CU** (Section 13.1.3 (on page 293)) is a two-port component describing an ideal gas. It has two energy ports which, with integral causality correspond to

1. Entropy flow in; temperature out

2. Volume rate of change in; pressure out

Four **SS** components are set up as sensors to measure the important quantities pertaining to the cycle:

1. **S** to measure the integrated entropy flow

2. **T** to measure the (absolute) temperature

3. **V** to measure the integrated volume change

4. **P** to measure the pressure

Cycle	Compression	Heating	Expansion	Cooling
Otto	II	II	II	II
Carnot	II	DI	II	DI
Diesel	II	ID	II	II
Joule	II	ID	II	ID

Table 13.1: Cycles and their causality

A number of cycles can be built depending on the causality of the two ports **[Heat]** and **[Work]** of **Cycle** Some possible cycles listed in Tables 13.1 (on page 296) where each table entry gives the causality on the heat and work ports respectively.

Summary information

System Cycle::Closed cycle with ideal gas Uses the CU two-port thermal capacitor.

Interface information:

Parameter $1 represents actual parameter **ideal_gas**

Parameter $2 represents actual parameter **c_v**

Parameter $3 represents actual parameter **gamma_g**

Parameter $4 represents actual parameter **m_g**

Port in represents actual port **Heat**

Port out represents actual port **Work**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: Cycle_lbl.txt

```
#SUMMARY Cycle: Closed cycle with ideal gas
#DESCRIPTION Uses the CU two-port thermal capacitor.

#ALIAS in Heat
#ALIAS out Work

#ALIAS $1 ideal_gas
#ALIAS $2 c_v
#ALIAS $3 gamma_g
#ALIAS $4 m_g

## Label file for system Cycle (Cycle_lbl.txt)

# ###############################################################
# ## Version control history
# ###############################################################
# ## $Id: Cycle_lbl.txt,v 1.3 2000/12/27 16:38:28 peterg Exp $
# ## $Log: Cycle_lbl.txt,v $
# ## Revision 1.3  2000/12/27 16:38:28  peterg
```

```
# ## *** empty log message ***
# ##
# ## Revision 1.2  1998/07/21 14:21:04  peterg
# ## New style file
# ##
# ## Revision 1.1  1997/12/08 20:24:43  peterg
# ## Initial revision
# ##
# ## Revision 1.1  1997/12/07 20:38:05  peterg
# ## Initial revision
# ##
# ###############################################################

## Each line should be of one of the following forms:
# a comment (ie starting with #)
# Component-name CR_name arg1,arg2,..argn
# blank




# Component type CU
gas CU ideal_gas,c_v,gamma_g,m_g

# Component type INTF
entropy
volume

# Component type SS
S SS external,0
T SS external,0
V SS external,0
P SS external,0
[Heat] SS external,external
[Work] SS external,external
```

Subsystems

- CU (1) No subsystems.
- INTF: flow integrator (2) No subsystems.

13.1.5 INTF

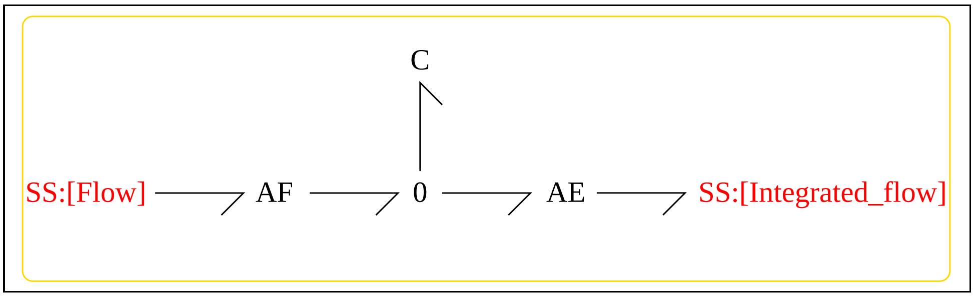


Figure 13.4: System **INTF**: acausal bond graph

The acausal bond graph of system **INTF** is displayed in Figure 13.4 (on page 299) and its label file is listed in Section 13.1.5 (on page 299). The subsystems are listed in Section 13.1.5 (on page 301).

INTF is a two-port component where the effort on port [out] is the integral of the flow on port [in].

Summary information

System INTF::flow integrator Port [in]: Flow to be integrated Port [out]: Effort = integral of flow on port [in]

Interface information:

Port in represents actual port **Flow**

Port out represents actual port **Integrated_flow**

Variable declarations:

This component has no PAR declarations

Units declarations:

This component has no UNITs declarations

The label file: INTF_lbl.txt

```
%% Label file for system INTF (INTF_lbl.txt)
%SUMMARY INTF: flow integrator
%DESCRIPTION Port [in]: Flow to be integrated
%DESCRIPTION Port [out]: Effort = integral of flow on port [in]


% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% Version control history
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% $Id: INTF_lbl.txt,v 1.3 1998/07/16 07:35:10 peterg Exp $
% %% $Log: INTF_lbl.txt,v $
% %% Revision 1.3  1998/07/16 07:35:10  peterg
% %% Aliased version
% %%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Port aliases
%ALIAS in Flow
%ALIAS out Integrated_flow

% Argument aliases

%% Each line should be of one of the following forms:
%       a comment (ie starting with %)
%       component-name cr_name arg1,arg2,..argn
%       blank

% ---- Component labels ----

% Component type SS
[Flow] SS external,external
[Integrated_flow] SS external,external
```

Subsystems

No subsystems.

13.2 Isovolumetric_struc.tex *(-o -ss)*

MTT command:

```
mtt -o -ss Isovolumetric struc tex
```

List of inputs for system Isovolumetric			
	Component	System	Repetition
1	Heat	Isovolumetric__Heat	1

List of outputs for system Isovolumetric			
	Component	System	Repetition
1	S	Isovolumetric__cycle__S	1
2	T	Isovolumetric__cycle__T	1
3	V	Isovolumetric__cycle__V	1
4	P	Isovolumetric__cycle__P	1

List of states for system Isovolumetric			
	Component	System	Repetition
1	c	Isovolumetric__cycle__gas__c	1
2	c	Isovolumetric__cycle__gas__c_2	1
3	mttC	Isovolumetric__cycle__entropy__mttC	1
4	mttC	Isovolumetric__cycle__volume__mttC	1

13.3 Isovolumetric_ode.tex *(-o -ss)*

MTT command:

```
mtt -o -ss Isovolumetric ode tex
```

$$\begin{aligned}
\dot{x}_1 &= \frac{(u_1 x_1)}{(c_v m_g)} \\
\dot{x}_2 &= 0 \\
\dot{x}_3 &= u_1 \\
\dot{x}_4 &= 0
\end{aligned} \tag{13.1}$$

$$
\begin{aligned}
y_1 &= x_3 \\
y_2 &= \frac{x_1}{(c_v m_g)} \\
y_3 &= x_4 \\
y_4 &= \frac{(x_1(\gamma_g - 1))}{x_2}
\end{aligned} \tag{13.2}
$$

13.4 Isovolumetric_ss.tex *(-o -ss)*

MTT command:

```
mtt -o -ss Isovolumetric ss tex
```

$$
x = \begin{pmatrix} \frac{100000}{(\gamma_g - 1)} \\ 1 \\ \frac{1000}{(3(\gamma_g - 1))} \\ 1 \end{pmatrix} \tag{13.3}
$$

$$
u = \begin{pmatrix} 0 \end{pmatrix} \tag{13.4}
$$

$$
y = \begin{pmatrix} \frac{1000}{(3(\gamma_g - 1))} \\ \frac{100000}{(c_v m_g(\gamma_g - 1))} \\ 1 \\ 100000 \end{pmatrix} \tag{13.5}
$$

$$
\dot{x} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{13.6}
$$

13.5 Isovolumetric_numpar.txt *(-o -ss)*

MTT command:

```
mtt -o -ss Isovolumetric numpar txt

# Numerical parameter file (Isovolumetric_numpar.txt)
# Generated by MTT at Thu Dec  4 11:44:46 GMT 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: Isovolumetric_numpar.txt,v 1.1 2000/12/28 18:17:57 peterg Exp
# %% $Log: Isovolumetric_numpar.txt,v $
# %% Revision 1.1  2000/12/28 18:17:57  peterg
# %% To RCS
# %%
# %% Revision 1.1  1998/03/04 11:45:49  peterg
# %% Initial revision
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Initial states -- needed to choose an approppriate mass
P_0 = 1e5;
V_0 = 1;
T_0 = 300;

# Parameters
c_v =  718.0; # Parameter c_v for CU
gamma_g = 1.4; # Parameter gamma_g for CU
m_g =  P_0*V_0/(T_0*(gamma_g-1)*c_v); # Parameter m for CU
```

13.6 Isovolumetric_input.txt *(-o -ss)*

MTT command:

```
mtt -o -ss Isovolumetric input txt

# Numerical parameter file (Isovolumetric_input.txt)
```

```
# Generated by MTT at Thu Dec  4 11:17:09 GMT 1997

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% Version control history
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
# %% $Id: Isovolumetric_input.txt,v 1.2 2003/08/06 18:55:08 gawthr
# %% $Log: Isovolumetric_input.txt,v $
# %% Revision 1.2  2003/08/06 18:55:08  gawthrop
# %% Updated for latest MTT version.
# %%
# %% Revision 1.1  2000/12/28 18:17:57  peterg
# %% To RCS
# %%
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

# Set the inputs
## Removed by MTT on Wed Aug  6 11:04:07 BST 2003:   u(1) = 1000;
isovolumetric__heat  = 1000;     #Entropy flow
```

13.7 Isovolumetric_odeso.ps *(-o -ss -Isovolumetric__cycle__V)*

MTT command:

```
mtt -o -ss Isovolumetric odeso ps 'Isovolumetric__cycle__V'
```

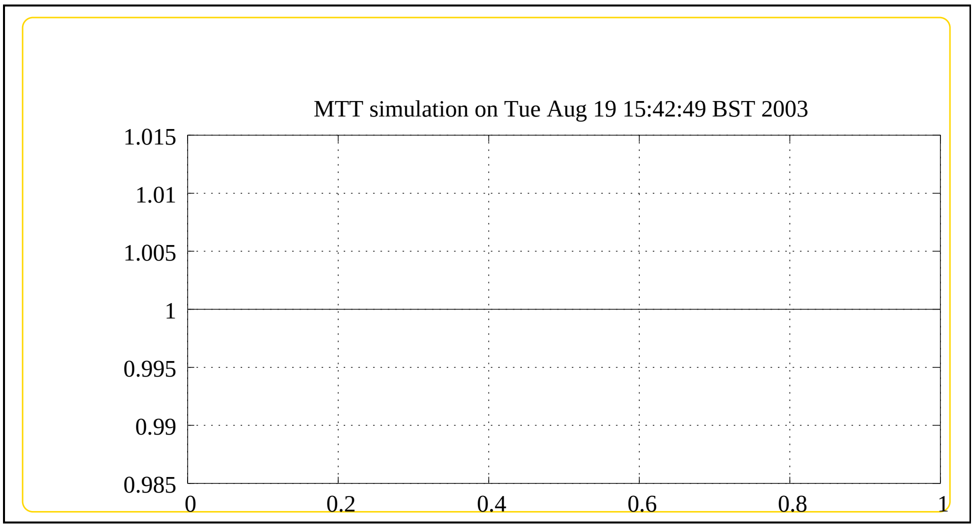
This representation is given as Figure 13.5 (on page 305).

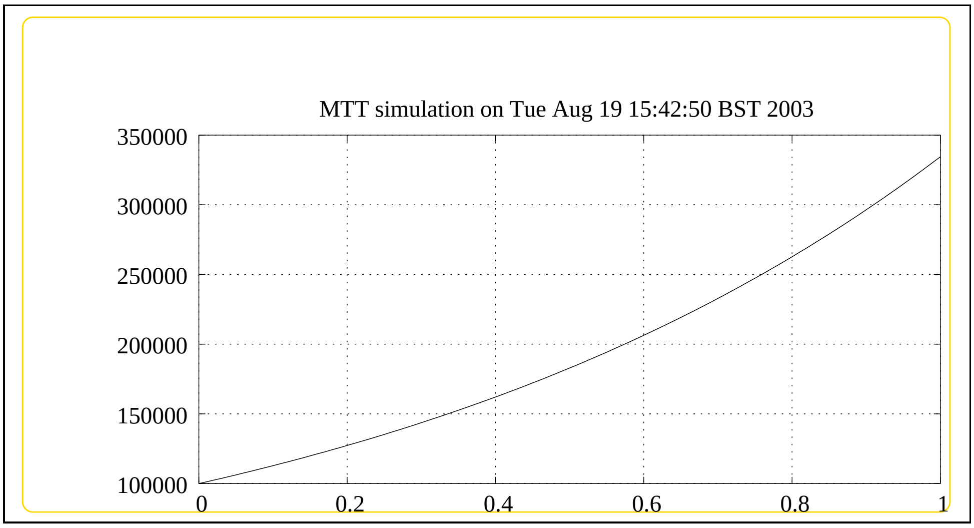
13.8 Isovolumetric_odeso.ps *(-o -ss -Isovolumetric__cycle__P)*

MTT command:

```
mtt -o -ss Isovolumetric odeso ps 'Isovolumetric__cycle__P'
```

This representation is given as Figure 13.6 (on page 305).


MTT simulation on Tue Aug 19 15:42:49 BST 2003
1.015
1.01
1.005
1
0.995
0.99
0.985
0
0.2
0.4
0.6
0.8
1

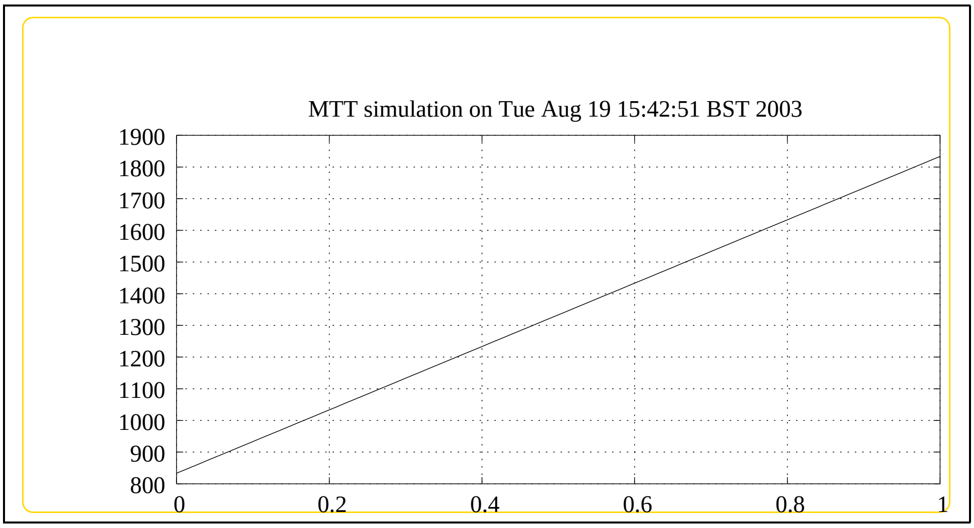

MTT simulation on Tue Aug 19 15:42:50 BST 2003
350000
300000
250000
200000
150000
100000
0
0.2
0.4
0.6
0.8
1

13.9 Isovolumetric_odeso.ps *(-o -ss -Isovolumetric__cycle__S)*

MTT command:

```
mtt -o -ss Isovolumetric odeso ps 'Isovolumetric__cycle__S'
```

This representation is given as Figure 13.7 (on page 306).



13.10 Isovolumetric_odeso.ps *(-o -ss -Isovolumetric__cycle__T)*

MTT command:

```
mtt -o -ss Isovolumetric odeso ps 'Isovolumetric__cycle__T'
```

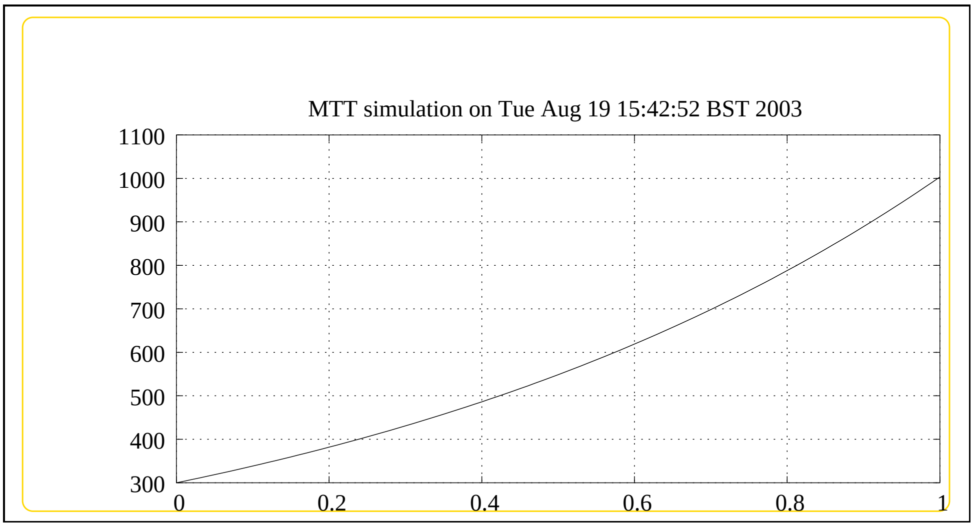
This representation is given as Figure 13.8 (on page 307).

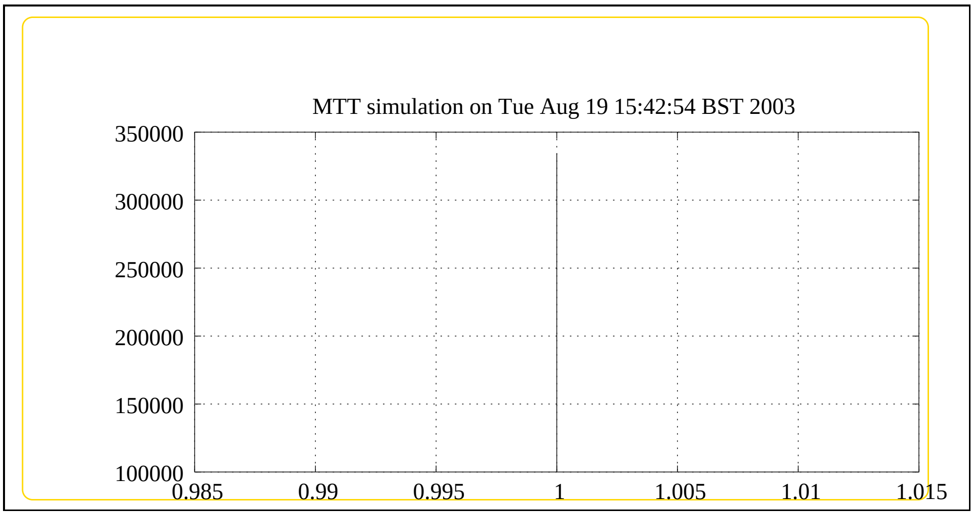
13.11 Isovolumetric_odeso.ps *(-o -ss -Isovolumetric__cycle__V:Isovolumetric__cycle__P)*

MTT command:

```
mtt -o -ss Isovolumetric odeso ps 'Isovolumetric__cycle__V:Isovolu
```

This representation is given as Figure 13.9 (on page 307).


MTT simulation on Tue Aug 19 15:42:52 BST 2003
1100
1000
900
800
700
600
500
400
300
0
0.2
0.4
0.6
0.8
1

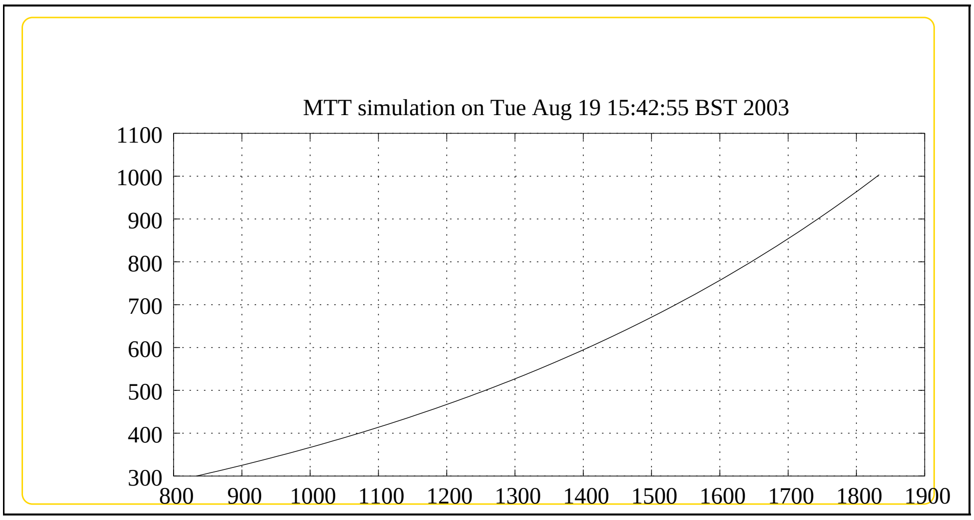

MTT simulation on Tue Aug 19 15:42:54 BST 2003
350000
300000
250000
200000
150000
100000
0.985
0.99
0.995
1
1.005
1.01
1.015

13.12 Isovolumetric_odeso.ps *(-o -ss -Isovolumetric__cycle__S:Isovolumetric__cycle__T)*

MTT command:

```
mtt -o -ss Isovolumetric odeso ps 'Isovolumetric__cycle__S:Isovolu
```

This representation is given as Figure 13.10 (on page 308).



Index