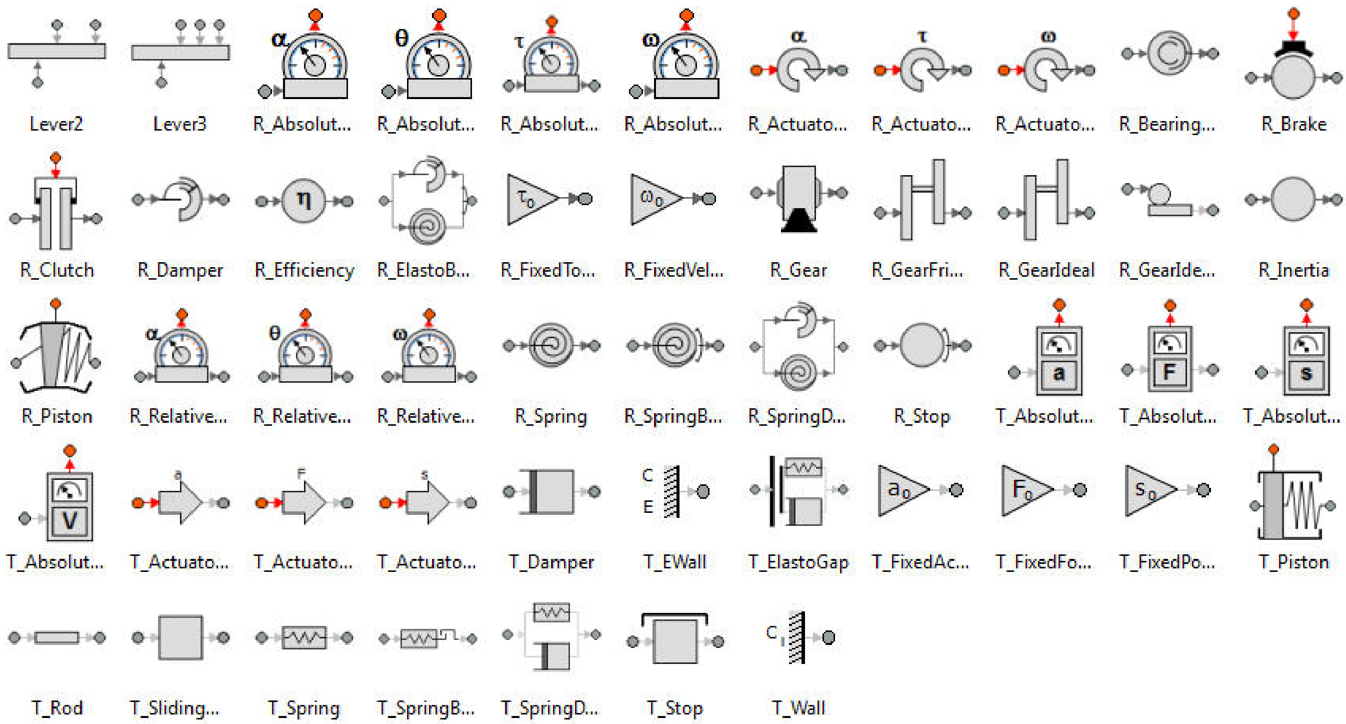
MECHANICAL Library 3.1.5

Reference Manual



EMPRESARIOS AGRUPADOS

Edition 2016

This page intentionally left blank.

Contents

This page intentionally left blank.

1. Overview of the library

1.1 Description

MECHANICAL is an EcosimPro library for the simulation of 1-dimensional mechanical systems. Most of the typical MECHANICAL units are included as components that can be used to build graphically complex systems:

- Translational components for systems with linear displacements driven by forces with frictional losses
- Rotational components for systems with angular displacements driven by momentum with frictional losses
- Kinetic converters (levers, gears, ideal gear rotational to translational): They transform an angular movement into other angular movement at different velocity or into a translational movement.

The most important elements can be grouped in the following groups:

- Masses and inertia (Sliding mass, End stop mass, Inertia and End stop Inertia): These elements implement the second Newton law.
- Force and torque generators (Coulomb friction, dampers and springs): They calculate force or torque from the position or the velocity in their ports.
- Actuators (force, position, torque, acceleration and angle generators): They provide force, torque, acceleration, position and angle depending on an external input signal (user defined law).
- Sensors (acceleration, velocity, position, angle, force and torque measurement): These components generate output CONTROL signals with the acceleration, velocity, position and force signals.
- Special rotational elements (levers, pistons, clutch, brake and efficiency).

A detailed examination of the formulation reveals that components in the MECHANICAL library calculate the forces, torques and positions in different ways: Indeed, mass or inertia components calculate explicitly (second order derivatives) the position at their ports; other components such springs and dumpers include linear or non-linear algebraic equations. Finally, some components include events in the calculation of the forces / displacement (Coulomb friction, Stops ...).

In this way, final models result in a complex system of differential-algebraic equations of index 3 (constraint equations have to be differentiated) with discrete events to be captured. It is a work of EcosimPro to order all the equations and to be able to symbolically differentiate equations, no matter the way the user includes the different components in a particular mechanical system.

Typical mechanical devices that could be frequently used in other more complex models can be built topologically from the basic components, and then considered as new components thanks to the EcosimPro object-oriented programming properties, i.e. Encapsulation, Inheritance and Aggregation.

1.2 Ports

This library used the following types of elementary standard ports that are defined in the library PORTS_LIB:

Port Type	Description	Colour
analog_signal	1-dimension array of analog signal	orange
mech_trans	1D translational flange	black
mech_rot	1D rotational flange	black

The variables defined in each port are the following:

- analog_signal port (for the treatment of the transducer signals)

Name	Description	Units
signal[n]	Array of analog signals	-

Left and right side of a mechanical component represent mechanical flanges:

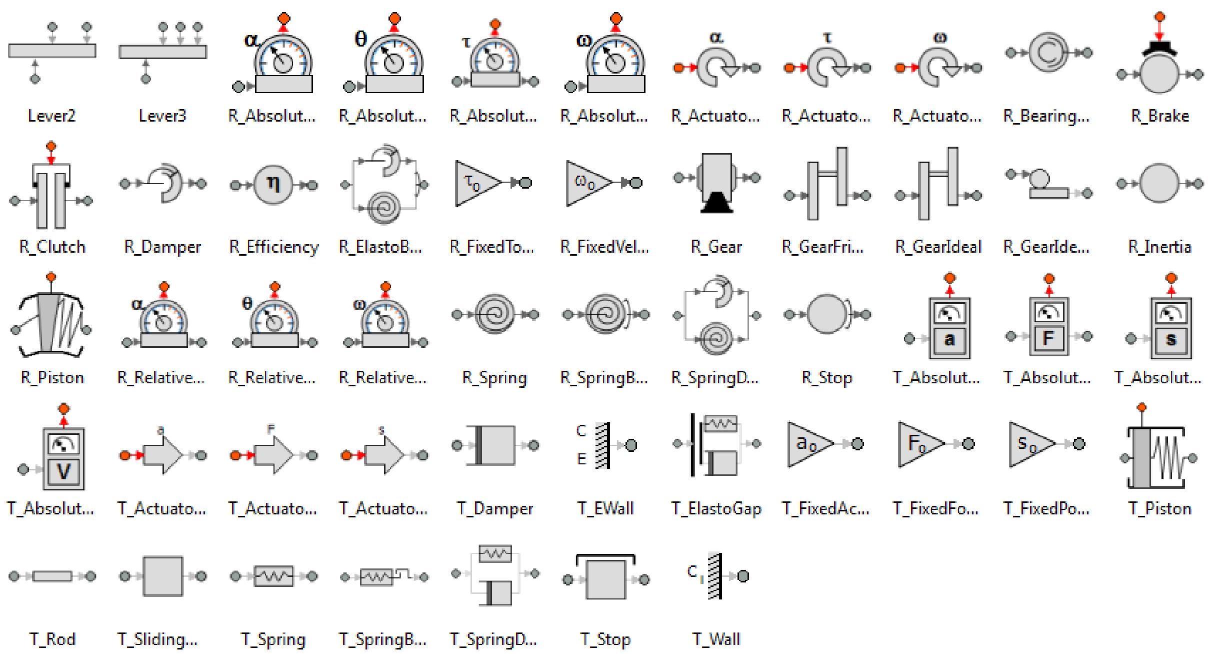
- mech_trans port

Name	Description	Units
T	Torque	N*m
S	Absolute position	m

- mech_rot port

Name	Description	Units
F	Force	N
n	Angular velocity	rpm
omega	Angular velocity	rad/s

1.3 Palette of symbols

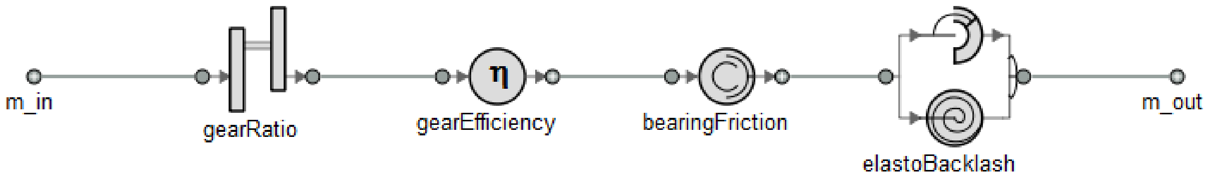


The components available in this library are depicted in the table below.

COMPONENTS	DESCRIPTION
Lever2	Lever Model (two points)
Lever3	Lever Model (three points)
R_AbsoluteSensor Acceleration	These components represent different type of sensors (rotational acceleration, angle, torque and angular velocity respectively). The output value is transmitted to a CONTROL signal, so a control block can be modelled
R_AbsoluteSensorAngle	
R_AbsoluteSensorTorque	
R_AbsoluteSensorVelocity	
R_ActuatorAcceleration	These components represent different type of actuators (rotational acceleration, torque and angular velocity respectively) They force a mechanical magnitude in one port, which is determined by an input signal.
R_ActuatorTorque	
R_ActuatorVelocity	
R_BearingFriction	This component represents a bearing. It includes Coulomb friction in bearings, i.e., a frictional torque acting between a flange and the housing
R_Brake	This component models a brake, where a controlled normal force presses the flange to the housing in order to increase friction.
R_Clutch	This component represents a clutch It can be engaged and disengaged. Coulomb friction is present between the two flanges which are pressed together via a normal force.
R_Damper	This component represents an ideal rotational damper.
R_Efficiency	This component represents the effect of mechanical losses in terms of efficiency.
R_ElastoBacklash	It represents a backlash element connected in series to a helicoidal spring and damper element which are connected in parallel.
R_FixedTorque	These components represents constant boundary conditions in torque and in angular velocity respectively
R_FixedVelocity	
R_Gear	This component includes the following components: an ideal Gear, a BearingFriction, a rotational Efficiency and a ElastoBacklash. The inertia of the gear wheels should be taken into account by connecting components of type R_Inertia to the inlet and/or the outlet ports
R_GearFriction	It models the gear ratio and the losses of a standard gear box in a reliable way including the stuck phases that may occur at zero speed
R_GearIdeal	This component represents a frictionless gearbox that transforms a rotational movement
R_GearIdealR2T	This component represents a frictionless gearbox that transforms a rotational movement into a linear movement
R_Inertia	This component calculates the movement (second derivative of the angular position) of a rotating mass with inertia
R_Piston	Dynamic model of an angular Piston submitted to the forces of pressure, spring, friction ... Discrete events are foreseen to stop the piston when the stops are reached
R_RelativeSensorAcceleration	These components represent different type of relative sensors (relative acceleration, relative angle and relative angular velocity respectively). The output value is transmitted to a CONTROL signal, so a control block can be modelled
R_RelativeSensorAngle	
R_RelativeSensorSpeed	
R_Spring	This component represents an ideal rotational spring
R_SpringBacklash	This component represents a spring with a dead zone where the spring does not act.
R_SpringDamper	This component represents a spring and a damper element connected in parallel
R_Stop	Rotating mass with inertia and stops Discrete events are foreseen to stop the mass when the stops are reached
T_AbsoluteSensorAcceleration	These components represent different type of translational sensors (acceleration, force, position and velocity respectively). The output value is transmitted to a CONTROL signal, so a control block can be modelled
T_AbsoluteSensorForce	
T_AbsoluteSensorPosition	
T_AbsoluteSensorVelocity	

COMPONENTS	DESCRIPTION
T_ActuatorAcceleration	These components represent different type of translational actuators (acceleration, force and position respectively) They force a mechanical magnitude in one port, which is determined by an input signal.
T_ActuatorForce	
T_ActuatorPosition	
T_Damper	This component represents an ideal damper
T_EWall	Wall with viscous coefficient .This element simulates a solid wall that allows shocks and bounces with it by means of calculating the reaction force of the ground
T_ElastoGap	Ideal spring damper combination with gap. This component represents a translational spring damper combination with gap
T_FixedAcceleration	These components represents constant boundary conditions in translational acceleration, force and position respectively
T_FixedForce	
T_FixedPosition	
T_Piston	Dynamic model of a Piston submitted to the forces of pressure, spring, friction ... Discrete events are foreseen to stop the piston when the stops are reached
T_Rod	This component represents a rod without inertia and two rigidly connected flanges
T_SlidingMass	This component calculates the movement (second derivative of the position) of a sliding mass in longitudinal movement
T_Spring	This component represents an ideal spring
T_SpringBacklash	This component represents a spring with a dead zone where the spring does not act
T_SpringDamper	Ideal linear damper and spring in parallel. This component represents a spring and a damper element connected in parallel
T_Stop	This component represents a sliding mass with hard stop and Stribeck friction. Discrete events are foreseen to stop the mass when the stops are reached
T_Wall	Wall with restitution coefficient. This component simulates a solid wall that allows shocks and bounces with it by means of a restitution coefficient.

Other components that a user may possibly require can be easily built by means of inheritance and aggregation of present components. In this respect, the "R_Gear" component is an example of a topological model behaving as a new component where several more simple units have been aggregated:



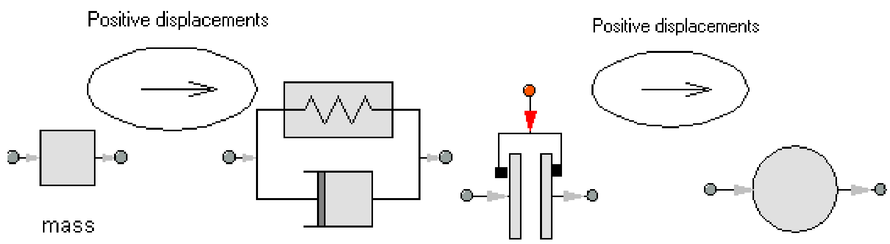
For a detailed description of the formulation and input data, please refer to the "htm" file present in the DOC directory of the Library

1.4 Sign convention

The direction of the arrows of the ports corresponds to the positive direction of the forces and displacements. The sign convention for translational or rotational ports is:

- Displacements are positive if they go from an IN port (marked with an inlet arrow, see figure below) to an OUT port (marked with an outlet arrow).
- Forces (or torques) are positive, either at inlet or outlet ports, if they will favor a positive movement of the mass.

In other words, If the force (torque) and the velocity in an inlet port are both positive, there is a power input in the component. Similarly, if the force (torque) and the velocity in an outlet port are both positive, there is a power output in the component



There is no restriction in the way the ports of a component are connected: Positive with negative, two positive ports together or two negative ports.

1.5 Guidelines for Building a Model

Components can be connected together in every meaningful combination (e.g. direct connection of two springs or two inertias).

However, difficulties may occur, if the elements which can lock the relative motion between two flanges are connected rigidly together. Currently, this type of problem can occur with the Coulomb friction elements BearingFriction, Clutch, Brake, LossyGear when the elements become stuck:

Sensors components of position extract the information from only one port, i.e., the value corresponding to that port, whereas force and torque sensors must be inserted between the ports where the force is required to be measured.

This page intentionally left blank.

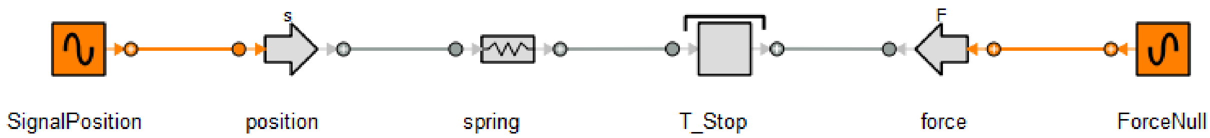
2. Building a Model

This library is designed for building MECHANICAL models by dragging and pasting components as if it was a real system.

Next sections describe how to build a simple model, and how to simulate it. The basic methodology to create and simulate this simple model is the same as that for more complex models.

2.1 Description of the example

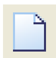
The model described in this chapter represents Translational Stop example.



The system consists of an external force and spring acting on a mass whose displacements is limited between two stops. Left side of the spring is fixed.

2.2 How to Build the Model

To create the model you should follow the following steps under the schematic view of EcosimPro. The active Workspace must include the MATH, PORTS_LIB, MECHANICAL _EXAMPLES and the CONTROL libraries loaded:

- Create a new schematic using the button  or the File -> New -> Schematics. Save it in the MECHANICAL _EXAMPLES library and choose a name (for example, my_model).
- From the icon explorer tab of the MECHANICAL library, select one by one the different components showed in the figure above and drag them to the schematic window
- Arrange the components in the schematic like in the figure above:
  - ◇ Use the Rotate buttons if necessary.
  - ◇ To change the size of a component, select the component, right button, select "component shape option" and change the size by dragging the symbol's corners.
  - ◇ To change the position of the component's name, press the SHIFT key at the same time as you move the mouse pointer over the labels, and drag it.
- Draw connectors between the components like in the figure above. A tooltip will appear whenever the mouse runs over a port, displaying the information of that port:

- Select the connection button on the right-hand toolbar , or press the SHIFT key at the same time as you move the mouse pointer over a port
- Left-click on the port to be connected
- Click the various points of the schematic drawing where the connector is required to run (if any)
- Left-click the target port, which must be of the same type as the origin port
- To connect different components with right angles, there are two options: 1) select right angle connections mode by pressing  button and connect components as explained before. 2) Create a connection straight line between two components, press SHIFT and left click over the line. A point will be created to divide the line in two different segments. Drag the point to the desired place and use  button to force right angles.
- To delete extra points of a connector line, just press SHIFT over the point and click on the point to be deleted

- Set the data of the components. Now you must customize the component data according to the table below:

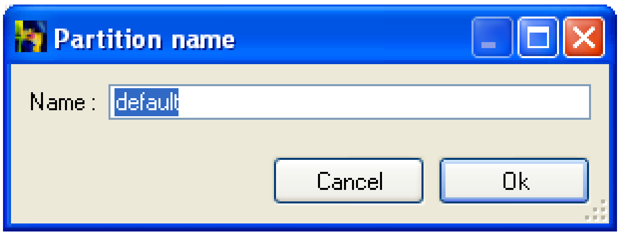
COMPONENT INSTANCES DATA:

DATUM	VALUE	DESCRIPTION	UNITS
ForceNull.Amp	0	Signal amplitude or height (-)	"-"
ForceNull.Offset	0	Offset of output signal (-)	"-"
ForceNull.Period	10	Period of sine, pulse, sawtooth, or square source (s)	"s"
ForceNull.Phase	0	Phase of sine source (rad)	"rad"
ForceNull.Tstart	0	Starting time of signal generation (s)	"s"
ForceNull.n_out	1	Dimension of outputs (-)	"-"
ForceNull.pulseWidth	0.001	Pulse width of pulse wave (s)	"s"
ForceNull.rampDuration	10	Duration of the ramp (s)	"s"
ForceNull.source	Constant	Waveform	""
ForceNull.timeTable		Table for table source (-)	"-"
SignalPosition.Amp	-1.75	Signal amplitude or height (-)	"-"
SignalPosition.Offset	0	Offset of output signal (-)	"-"
SignalPosition.Period	10	Period of sine, pulse, sawtooth, or square source (s)	"s"
SignalPosition.Phase	0	Phase of sine source (rad)	"rad"
SignalPosition.Tstart	0	Starting time of signal generation (s)	"s"
SignalPosition.n_out	1	Dimension of outputs (-)	"-"
SignalPosition.pulseWidth	0.001	Pulse width of pulse wave (s)	"s"
SignalPosition.rampDuration	10	Duration of the ramp (s)	"s"
SignalPosition.source	Constant	Waveform	""
SignalPosition.timeTable		Table for table source (-)	"-"
T_Stop.F_Coulomb	3	Constant friction: Coulomb force (N)	"N"
T_Stop.F_Stribeck	5	Stribeck effect (N)	"N"
T_Stop.F_prop	1	Velocity dependent friction (N/[m/s])	"N/[m/s]"
T_Stop.fexp	2	Exponential decay (1/[m/s])	"1/[m/s]"
T_Stop.m	1	Mass (kg)	"kg"
T_Stop.smax	0.9	Right stop for sliding mass (m)	"m"
T_Stop.smin	-0.9	Left stop for sliding mass (m)	"m"
spring.k	500	Spring constant (N/m)	"N/m"
spring.s_rel0	1	Distance between ports for unloaded spring (m)	"m"

- To change data of a component double-click on the corresponding symbol to open the Attributes Editor. For example, for the T_stop component, data values should be like this:

		DATA		
m	REAL	1	kg	Mass [kg]
smax	REAL	0.9	m	Right stop for sliding mass [m]
smin	REAL	-0.9	m	Left stop for sliding mass [m]
F_prop	REAL	1	N/[m/s]	Velocity dependent friction [N/[m/s]]
F_Coulomb	REAL	3	N	Constant friction: Coulomb force [N]
F_Stribeck	REAL	5	N	Stribeck effect [N]
fexp	REAL	2	1/[m/s]	Exponential decay [1/[m/s]]

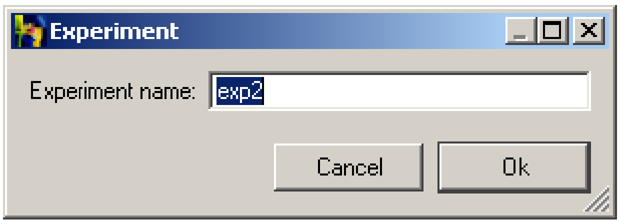
- Finally, generate the EcosimPro model using the button . The following window will be display. Click OK and you will have finalised the construction of your model and you can simulate it as described in the following section, "Simulating the Model":



2.3 Simulate the Model

Now it is ready for the simulation. Go to the Simulation View.

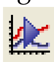
- Select "my_model.default" (if the name of the partition you have created is "default") from the MECHANICAL_-EXAMPLES library, right-click, select option "New experiment". The following window will be display. Type a name for the experiment:

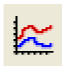


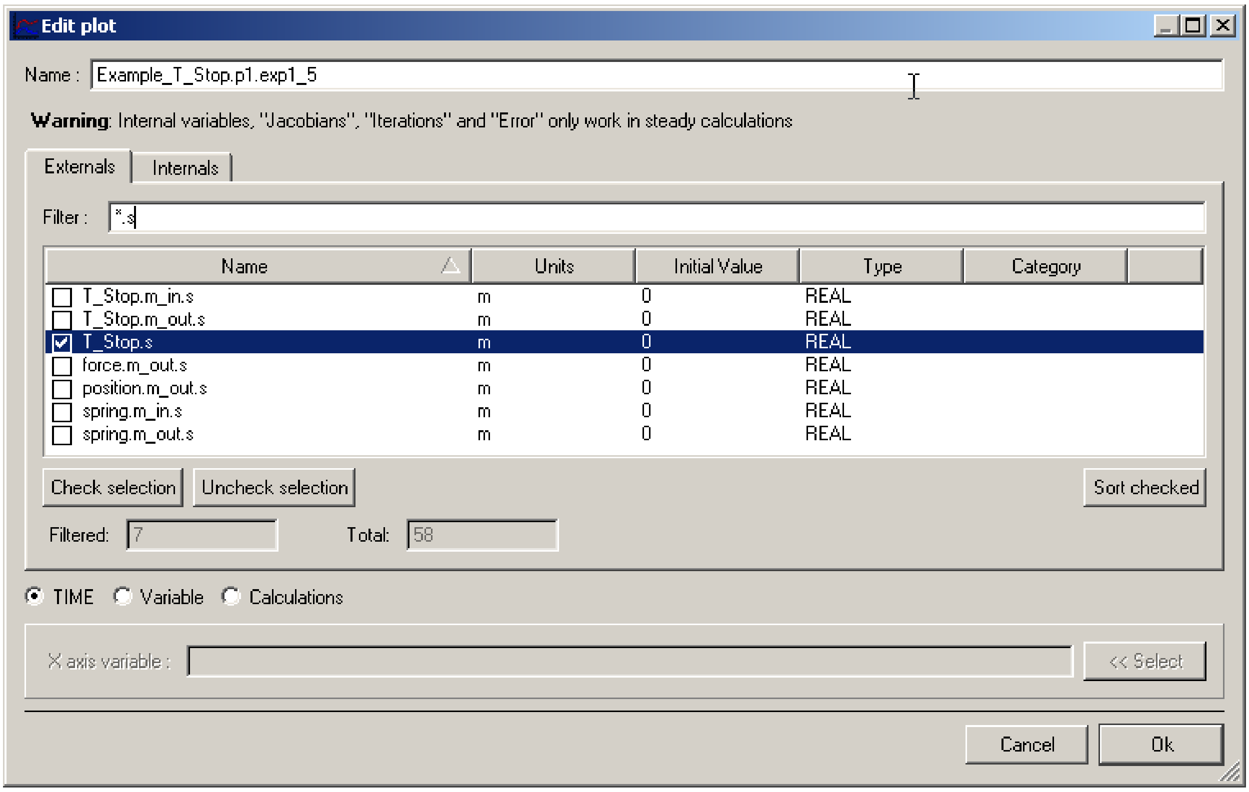
- A default experiment text should appear in the editing window. Change the TSTOP, CINT (Communications Interval) and other values as indicated here below:

```
EXPERIMENT exp1 ON Example_T_Stop.p1
DECLS
INIT    -- set initial values for variables
     -- Dynamic variables
   T_Stop.s = 0
   T_Stop.v = 5

BOUNDS    -- set expressions for boundary variables: v = f(t,...)

BODY
   T_Stop.smin = -0.4
   TIME = 0
   TSTOP = 0.5
   CINT = 0.01
   INTEG()
END EXPERIMENT
```

- ◇ Double dash (–) means a comment.
- ◇ "IMETHOD = DASSL_SPARSE" means that the DASSL sparse integration method will be employed (See the EcosimPro User Manual.
- ◇ "REPORT_MODE = IS_STEP" means that an output (in the plots and in the reports) will be produced at each internal time step (given by the integration method), that is normally shorter than the communication interval CINT.

- Save the experiment. The experiment name will appear in the Workspace area
- Now you're going to simulate the experiment using the Monitor. Right button over the experiment name in the Workspace area, and select "Simulate in Monitor". Clicking  button will produce same action
- The Experiments monitor comes up. Add the necessary plots to visualize the results.

- ◇ To add a plot click on "Tab_1" and then click "New Plot" . A window will appear with the "drawable / plotable" variables of the experiment.
- ◇ Apply a filter to help you selecting variables. Tick the boxes alongside selected variables:



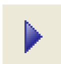
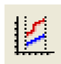
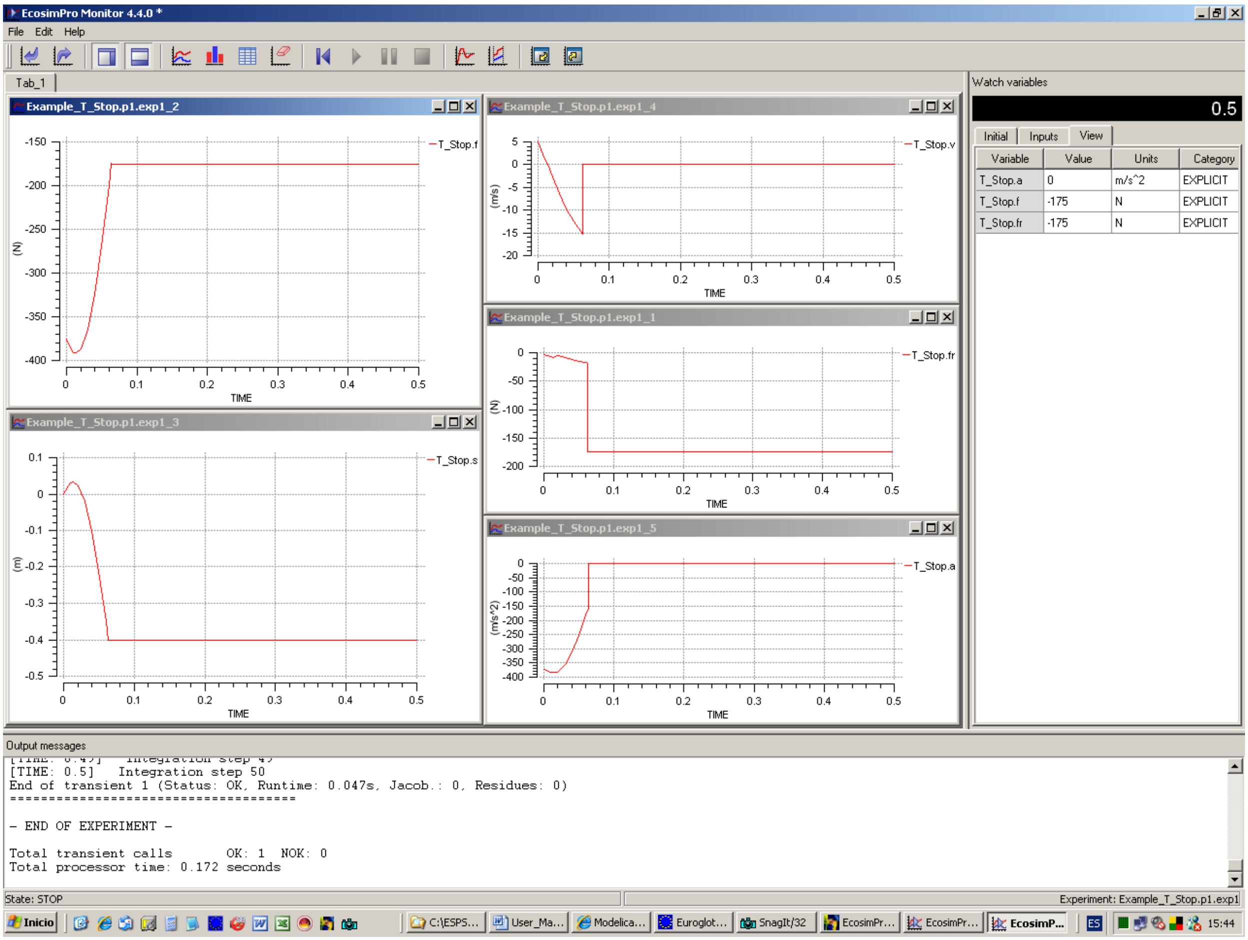
- Just as the user may wish to plot the evolution of a variable compared with another, he may also wish to track the numerical values of the variables. The main task of the Watch is to display the variable values of the experiment, enabling them to be modified wherever possible.
  - ◇ The first step is to add variables by doing right click in the Watch Area and select the option "Edit Watch".
  - ◇ A menu very similar to that of the plots will appear, enabling the user to add or delete variables.
- To simulate the experiment click "Simulate" . By clicking "New Integration"  on the toolbar, the simulation can be extended or started at new times.

Figure below shows some typical time histories of the main variables of the model:


EcosimPro Monitor 4.4.0 *
Example_T_Stop.p1.exp1_2
T_Stop.f
Example_T_Stop.p1.exp1_4
T_Stop.v
Example_T_Stop.p1.exp1_1
T_Stop.fr
Example_T_Stop.p1.exp1_3
T_Stop.s
Example_T_Stop.p1.exp1_5
T_Stop.a
TIME
Watch variables
0.5
Variable Value Units Category
T_Stop.a 0 m/s^2 EXPLICIT
T_Stop.f -175 N EXPLICIT
T_Stop.fr -175 N EXPLICIT
Output messages
[TIME: 0.5] Integration step 50
End of transient 1 (Status: OK, Runtime: 0.047s, Jacob.: 0, Residues: 0)
- END OF EXPERIMENT -
Total transient calls OK: 1 NOK: 0
Total processor time: 0.172 seconds
State: STOP
Experiment: Example_T_Stop.p1.exp1

3. Abstract Components of MECHANICAL library

3.1 Component R_AbsFriction

3.1.1 Description

Abstract component of rotational coulomb friction components. This component is used as a base component of Coulomb friction elements. It calculates the configuration mode of the Coulomb friction elements: Free, Forward, Backward or Stuck by means of the variable "imode".

The meaning of the values of the "imode" variable is shown in the following table:

Mode	Meaning
-2	The element rotates backward
-1	The element starts to rotate backward
0	The element is stuck
1	The element starts to rotate forward
2	The element rotates forward
3	The element is not active, it is free.

3.1.2 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
a_relfric	REAL			Relative angular acceleration between frictional surfaces	rad/s^2
free	BOOLEAN	FALSE		TRUE, if frictional element is not active	-
imode	INTEGER	-2		Operation mod	-
sa	REAL			Path parameter of friction characteristic	-
tau	REAL			Friction torque: positive, if directed in opposite direction of w_rel	N·m
tau0	REAL			Friction torque for w=0 and forward sliding	N·m
tau0_max	REAL			Maximum friction torque for w=0 and locked	N·m
w_relfric	REAL			Relative angular velocity between frictional surface	

3.1.3 Formulation

The component calls the function fun_R_AbsFriction to calculate the configuration mode of the Coulomb friction elements.

```
fun_R_AbsFriction(imode, a_relfric, sa, tau0) = 0
```

3.2 Component R_AbsoluteSensor

Inherited from CONTROL.Asensor

3.2.1 Description

Abstract class for definition of generic rotational absolute sensors. This abstract component is the base for rotational kinetic sensors (acceleration, velocity and angle sensors). In this type of components, the measure of the magnitude is carried out in one single port. This is why these components are called "Absolute".

This component has one mechanical rotational port and one signal port.

3.2.2 Parameters

NAME	TYPE	DEFAULT	DESCRIPTION
n_out	INTEGER	1	Dimension of outputs

3.2.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN	Rotational mechanical inlet port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT	Outlet signal

3.2.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
Gain[n_out]	REAL	1		Gain	-

3.2.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
var[n_out]	REAL			Measured variable	-

3.3 Component R_Actuator

3.3.1 Description

Abstract class for definition of generic rotational actuators. This abstract component is the base for rotational actuators that transform input signal into mechanical magnitudes such as velocity or momentum.

This component has one mechanical rotational port and one signal port.

3.3.2 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_out	PORTS_LIB.mech_rot		OUT	Outlet rotational mechanical port
s_in	PORTS_LIB.analog_signal		IN	Input signal port

3.4 Component R_Compliant

Inherited from R_Two_Flanges.

3.4.1 Description

Abstract class for definition of a compliant connection of two rotational ports. This abstract component is the base for components where the relative angular velocity between its two ports may change. The momentum in the ports are identical. It is used by components like sliding springs and dampers.

3.4.2 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN	Left / driven mech_rot
m_out	PORTS_LIB.mech_rot		OUT	Right / driven mech_rot

3.4.3 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
phi_rel_i	REAL	0		Initial angular distance between ports	rad

3.4.4 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
T	REAL			Rotational transmited torque	N·m
phi_rel	REAL			Angular distance between ports	rad

3.4.5 Formulation

The mathematical model associated to this component is given as follows:

- Relative angle: phi_rel = m_out.phi - m_in.phi
- Torque in inlet port: m_in.tau = tau
- Torque in outlet port: m_out.tau = tau

The outlet torque is equal to the inlet torque.

3.5 Component R_RelativeSensor

Inherited from CONTROL.Asensor

3.5.1 Description

Abstract class for definition of generic rotational relative sensors. This abstract component is the base for rotational sensors that generate its signal by comparing the values of a given magnitude in two ports. This is why these components are called "Relative". The sensors created from this abstract component must be placed in series.

This component has two mechanical rotational ports and one signal port.

3.5.2 Parameters

NAME	TYPE	DEFAULT	DESCRIPTION
n_out	INTEGER	1	Dimension of outputs

3.5.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN	Rotational mechanical inlet port
m_out	PORTS_LIB.mech_rot		OUT	Rotational mechanical outlet port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT	Outlet signal

3.5.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-

3.5.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
var[n_out]	REAL			Measured variable	-

3.6 Component R_Rigid

Inherited from R_Two_Flanges

3.6.1 Description

This abstract component is the base for rotational components where a constant angular velocity w is considered. This angular velocity is supplied to the ports. On the other hand, the momentum in the ports can be different. It is used by components like inertia.

3.6.2 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN	Rotational mechanical inlet port
m_out	PORTS_LIB.mech_rot		OUT	Rotational mechanical outlet port

3.6.3 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
w	REAL			Absolute angular velocity	rad/s

3.6.4 Formulation

The mathematical model associated to this component is shown below:

- Angular velocity at inlet port: m_in.omega = w
- Angular velocity at outlet port: m_out.omega = w

The two mechanical connections (flanges) have the same angular speed.

3.7 Component R_Two_Flanges

3.7.1 Description

Abstract class for definition of components with two rotational ports. This abstract component is the base for two ports rotational components: It provides two mechanical rotational ports without any internal calculation.

3.7.2 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN	Left / driven mech_rot
m_out	PORTS_LIB.mech_rot		OUT	Right / driven mech_rott

3.8 Component T_AbsoluteSensor

Inherited from CONTROL.Asensor

3.8.1 Description

Abstract class for definition of generic translational absolute sensors. This abstract component is the base for translational kinetic sensors (acceleration, velocity and position sensors). In this type of components, the measure of the magnitude is carried out in one single port. This is why these components are called "Absolute". This component has one mechanical translational port and one signal port.

3.8.2 Parameters

NAME	TYPE	DEFAULT	DESCRIPTION
n_out	INTEGER	1	Dimension of outputs

3.8.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN	Mechanical left port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT	Outlet signal

3.8.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-

3.8.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
var[n_out]	REAL			Measured variable	-

3.9 Component T_Actuator

3.9.1 Description

Abstract class for definition of generic translational actuators. This abstract component is the base for translational actuators that transform input signal into mechanical magnitudes such as position or force. This component has one mechanical translational port and one signal port.

3.9.2 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_out	PORTS_LIB.mech_trans		OUT	Output translational mechanical port
s_in	PORTS_LIB.analog_signal		IN	Input signal port

3.10 Component T_Compliant

Inherited from T_Two_Flanges.

3.10.1 Description

Abstract class for definition of a compliant connection of two translational ports. This abstract component is the base for translational components where the relative position between its two ports may change. The force in the ports are identical but opposite. It is used by components like springs and dampers.

3.10.2 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN	Left / driven mech_trans
m_out	PORTS_LIB.mech_trans		OUT	Right / driven mech_trans

3.10.3 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
F	REAL			Translational transmited force	N
s_rel	REAL			Distance between ports	rad

3.10.4 Formulation

The mathematical formulation associated to this component are the following:

- Relative displacement: s_rel = m_out.s - m_in.s
- Force at inlet port: m_in.F = -F
- Force at outlet port: m_out.F = F

3.11 Component T_RelativeSensor

Inherited from CONTROL.Asensor

3.11.1 Description

Abstract class for definition of generic translational relative sensors. This abstract component is the base for translational sensors that generate its signal by comparing the values of a given magnitude in two ports. This is why these components are called "Relative". The sensors created from this abstract component must be placed in series. This component has two mechanical translational ports and one signal port.

3.11.2 Parameters

NAME	TYPE	DEFAULT	DESCRIPTION
n_out	INTEGER	1	Dimension of outputs

3.11.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN	Mechanical left port
m_out	PORTS_LIB.mech_trans		OUT	Mechanical right port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT	Outlet signal

3.11.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias	REAL	0		Bias	-
gain	REAL	1		Gain	-

3.11.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
var[n_out]	REAL			Measured variable	-

3.12 Component T_Rigid

Inherited from T_Two_Flanges

3.12.1 Description

Abstract class for definition of a rigid connection of two translational ports. This abstract component is the base for translational components where a constant position s is considered. This position is supplied to the ports. On the other hand, the force in the ports can be different. It is used by components like masses.

3.12.2 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN	Left / driving mech_trans
m_out	PORTS_LIB.mech_trans		OUT	Right / driving mech_trans

3.12.3 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
s	REAL			Absolute position of component	m

3.12.4 Formulation

The mathematical formulation associated to this component is the following:

- Position at inlet port: m_in.s = s
- Position at outlet port: m_out.s = s

3.13 Component T_Two_Flanges

3.13.1 Description

Abstract class for definition of components with two translational ports. This abstract component is the base for two ports translational components: It provides two mechanical translational ports without any internal calculation.

3.13.2 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN	Left / driven mech_trans
m_out	PORTS_LIB.mech_trans		OUT	Right / driven mech_trans

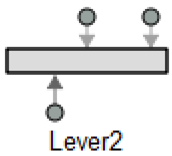
4. Component Types of MECHANICAL library

4.1 Component Lever2

4.1.1 Description

Lever with two points

4.1.2 Symbol



4.1.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
fulcrum	PORTS_LIB.mech_rot		IN		
load1	PORTS_LIB.mech_trans		IN		
load2	PORTS_LIB.mech_trans		IN		

4.1.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
I	REAL	1		Inertia of the lever	kg·m^2
L1	REAL	1		Distance from load input 1 to fulcrum	m
L2	REAL	0.5		Distance from load input 2 to fulcrum	m
phi0	REAL	0		Initial angular position (rad)	rad
phi_max	REAL	0.5		Lever upper limit angular position	rad
phi_min	REAL	-0.5		Lever lower limit angular position	rad

4.1.5 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
initial	BOOLEAN	TRUE			
phi	REAL			Lever angle	rad
s1	REAL			Displacement of load 1 application point	m
s2	REAL			Displacement of load 2 application point	m

4.1.6 Formulation

This component named is a topologic component. It does not have specific formulation, it consists of the following components:

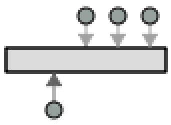
- R_GearIdealR2T
- R_GearIdealR2T
- R_Stop

4.2 Component Lever3

4.2.1 Description

Lever with three points

4.2.2 Symbol



4.2.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
fulcrum	PORTS_LIB.mech_rot		IN		
load1	PORTS_LIB.mech_trans		IN		
load2	PORTS_LIB.mech_trans		IN		
load3	PORTS_LIB.mech_trans		IN		

4.2.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
I	REAL	1		Inertia of the lever	kg·m^2
L1	REAL	1		Distance from load input 1 to fulcrum	m
L2	REAL	0.5		Distance from load input 2 to fulcrum	m
L3	REAL	0.5		Distance from load input 3 to fulcrum	m
phi0	REAL	0		Initial angular position (rad)	rad
phi_max	REAL	0.5		Lever upper limit angular position	rad
phi_min	REAL	-0.5		Lever lower limit angular position	rad

4.2.5 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
phi	REAL			Lever angle	rad
s1	REAL			Displacement of load 1 application point	m
s2	REAL			Displacement of load 2 application point	m
s3	REAL			Displacement of load 2 application point	m

4.2.6 Formulation

This component named is a topologic component. It does not have specific formulation, it consists of the following components:

- R_GearIdealR2T
- R_GearIdealR2T
- R_GearIdealR2T
- R_Stop

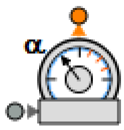
4.3 Component R_AbsoluteSensorAcceleration

Inherited from R_AbsoluteSensor.

4.3.1 Description

Absolute angular acceleration sensor

4.3.2 Symbol



R_AbsoluteSensorAcceleration

4.3.3 Construction Parameters

NAME	TYPE	DEFAULT	DESCRIPTION	UNITS
n_out	INTEGER	1	Dimension of outputs	-

4.3.4 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Rotational mechanical inlet port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT		Outlet signal ()

4.3.5 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias (-)	-
gain[n_out]	REAL	1		Gain (-)	-

4.3.6 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
ac	REAL			Absolute angular acceleration	rad/s^2
var[n_out]	REAL			Measured variable	-

4.3.7 Formulation

The equations associated to this component are shown below:

- Angular acceleration: ac = m_in.omega'
- Torque: m_in.T = 0.
- Output signal: var[1] = ac

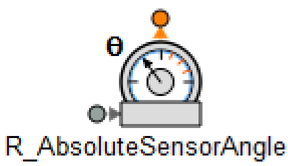
4.4 Component R_AbsoluteSensorAngle

Inherited from R_AbsoluteSensor.

4.4.1 Description

AbsoluteSensorAngle simulates an angle sensor. The measure of the position takes place in one single port. This component has one mechanical input port and one signal output port.

4.4.2 Symbol



4.4.3 Construction Parameters

NAME	TYPE	DEFAULT	DESCRIPTION	UNITS
n_out	INTEGER	1	Dimension of outputs	-

4.4.4 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Rotational mechanical inlet port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT		Outlet signal ()

4.4.5 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-
phi0	REAL	0		Initial angular position	rad

4.4.6 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
phi	REAL			Absolute angular position	rad
var[n_out]	REAL			Measured variable	-

4.4.7 Formulation

The equations associated to this component are shown below:

- Angle: phi' = m_in.omega
- Torque: m_in.T = 0
- Output signal: var =phi

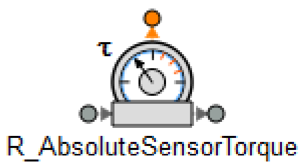
4.5 Component R_AbsoluteSensorTorque

Inherited from R_AbsoluteSensor.

4.5.1 Description

This component simulates a torque sensor. The measure of the torque takes place in two ports. This component has two mechanical ports and one signal port. Additionally, it must be placed in series in the branch where the torque has to be measured.

4.5.2 Symbol



4.5.3 Construction Parameters

NAME	TYPE	DEFAULT	DESCRIPTION	UNITS
n_out	INTEGER	1	Dimension of outputs	-

4.5.4 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Rotational mechanical inlet port
m_out	PORTS_LIB.mech_rot		OUT		Rotational mechanical outlet port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT		Outlet signal ()

4.5.5 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-

4.5.6 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
T	REAL			Measured torque	N·m
var[n_out]	REAL			Measured variable	-

4.5.7 Formulation

The equations associated to this component are shown below:

- Torque at inlet port: m_in.T = T
- Torque at outlet port: m_out.T = T
- Ports angular velocity: m_out.omega = m_in.omega
- Output signal: var[1] = T

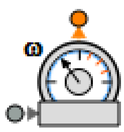
4.6 Component R_AbsoluteSensorVelocity

Inherited from R_AbsoluteSensor.

4.6.1 Description

AbsoluteSensorVelocity simulates an angular velocity sensor. The measure of the velocity takes place in one single port. This component has one mechanical input port and one signal output port.

4.6.2 Symbol



R_AbsoluteSensorVelocity

4.6.3 Construction Parameters

NAME	TYPE	DEFAULT	DESCRIPTION	UNITS
n_out	INTEGER	1	Dimension of outputs	-

4.6.4 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Rotational mechanical inlet port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT		Outlet signal ()

4.6.5 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-

4.6.6 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
var[n_out]	REAL			Measured variable	-
w	REAL			Absolute angular velocity	rad/s

4.6.7 Formulation

The equations associated to this component are shown below:

- Angular velocity: w = m_in.omega
- Torque: m_in.T = 0
- Output signal: var[1] = w

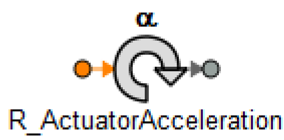
4.7 Component R_ActuatorAcceleration

Inherited from R_Actuator.

4.7.1 Description

ActuatorAcceleration forces a mechanical angular acceleration in one port, which is determined by an input signal. This component has one mechanical port and one signal port.

4.7.2 Symbol



4.7.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_rot		OUT		Rotational mechanical outlet port
s_in	PORTS_LIB.analog_signal	(n = n_out)	IN		Input signal ()

4.7.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
wi	REAL	0		Initial angular velocity	rad/s

4.7.5 Formulation

The mathematical equations governing this component are given as follows:

- Input signal: a = input.signal
- Angular velocity: a = m_out.omega'

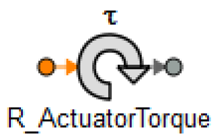
4.8 Component R_ActuatorTorque

Inherited from R_Actuator.

4.8.1 Description

T_ActuatorTorque generates a mechanical torque from an input signal. This component has one mechanical port and one signal port.

4.8.2 Symbol



4.8.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_rot		OUT		Rotational mechanical outlet port
s_in	PORTS_LIB.analog_signal	(n = n_out)	IN		Input signal ()

4.8.4 Formulation

The mathematical equation governing this component is given as follows:

- Port torque: m_out.T = s_in.signal[1]

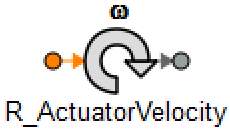
4.9 Component R_ActuatorVelocity

Inherited from R_Actuator.

4.9.1 Description

R_ActuatorVelocity forces a mechanical angular velocity in one port, which is determined by an input signal. This component has one mechanical port and one signal port.

4.9.2 Symbol



4.9.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_rot		OUT		Rotational mechanical outlet port
s_in	PORTS_LIB.analog_signal	(n = n_out)	IN		Input signal ()

4.9.4 Formulation

The mathematical equation governing this component is given as follows:

- Angular velocity: m_out.omega = s_in.signal[1]

4.10 Component R_BearingFriction

Inherited from R_Rigid and R_AbsFriction.

4.10.1 Description

This component represents a bearing that is a mechanical component used to reduce friction in a machine. It describes Coulomb friction in bearings, i.e., a frictional torque acting between a flange and the housing.

4.10.2 Symbol

4.10.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driving mech_rot

4.10.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
peak	REAL	1		Auxiliary variable to calculate the maximum friction torque	-
tau_pos	TABLE_1D			Positive sliding friction characteristic for positive speed	-

4.10.5 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
a	REAL			Absolute angular acceleration	rad/s^2
a_relfric	REAL			Relative angular acceleration between frictional surfaces	rad/s^2
free	BOOLEAN	FALSE		TRUE, if frictional element is not active	-
imode	INTEGER	-2		Operation mod	-
sa	REAL			Path parameter of friction characteristic	-
tau	REAL			Friction torque: positive, if directed in opposite direction of w_rel	N·m
tau0	REAL			Friction torque for w=0 and forward sliding	N·m
tau0_max	REAL			Maximum friction torque for w=0 and locked	N·m
w	REAL			Absolute angular velocity	rad/s
w_relfric	REAL			Relative angular velocity between frictional surface	

4.10.6 Formulation

The mathematical model associated to this component is shown below:

- Angular velocity at inlet port: m_in.omega = w
- Angular velocity at outlet port: m_out.omega = w

The two mechanical connections (flanges) have the same angular speed.

The positive sliding friction torque "tau" has to be defined by table "tau_pos" as function of the absolute angular velocity "w". This torque is calculated interpolating linearly in the table with the value of the angular velocity. It is assumed that the negative sliding friction force has the same characteristic with negative values. Friction is modelled in the following way:

When the absolute angular velocity "w" is not zero, the friction torque is a function of w and of a constant normal force. This dependency is defined via table tau_pos and can be determined by measurements, e.g. by driving the gear with constant velocity and measuring the needed motor torque (friction torque).

```
tau = ZONE(imode == 0)
                  sa
      ZONE(imode == 2 OR imode == 1)
                  linearInterp1D(tau_pos, w)
      ZONE(imode == -2 OR imode == -1)
                  -linearInterp1D(tau_pos, -w)
      OTHERS
                  0.0
```

When the absolute angular velocity becomes zero, the elements connected by the friction element become stuck, i.e., the absolute angle remains constant. In this phase the friction torque is calculated from a torque balance due to the requirement, that the absolute acceleration shall be zero. The elements begin to slide when the friction torque exceeds a threshold value, called the maximum static friction torque "tau0_max", computed via:

```
tau0_max = peak * tau0
```

where tau0 is the sliding friction torque when the angular velocity is zero.

This procedure is implemented by state events and leads to continuous/discrete systems.

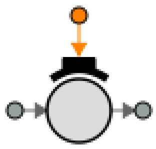
4.11 Component R_Brake

Inherited from R_Rigid and R_AbsFriction.

4.11.1 Description

This component models a brake, that is., a component where a frictional torque is acting between the housing and a flange and a controlled normal force presses the flange to the housing in order to increase friction.

4.11.2 Symbol



4.11.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
inPort	PORTS_LIB.analog_signal	(n = 1)	IN	Normalized force signal (= normal_force/fn_max; clutch is engaged if > 0)
m_in	PORTS_LIB.mech_rot		IN	Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT	Right / driving mech_rot

4.11.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
cgeo	REAL	1		Geometry constant containing friction distribution assumption	
fn_max	REAL	20		Maximum normal force	N
mue_pos	TABLE_1D			[w,mue] positive sliding friction coefficient: w_rel>=0	
peak	REAL	1.1		Maximum value of mue for w_rel=0	-
w_i	REAL	1		Initial relative angular velocity	rad/s

4.11.5 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
a	REAL			Absolute angular acceleration	rad/s^2
a_relfric	REAL			Relative angular acceleration between frictional surfaces	rad/s^2
fn	REAL			Normal force	N
free	BOOLEAN	FALSE		TRUE, if frictional element is not active	-
imode	INTEGER	-2		Operation mod	-
mue0	REAL			Friction coefficient for w=0 and forward sliding	-
sa	REAL			Path parameter of friction characteristic	-
tau	REAL			Friction torque: positive, if directed in opposite direction of w_rel	N·m
tau0	REAL			Friction torque for w=0 and forward sliding	N·m
tau0_max	REAL			Maximum friction torque for w=0 and locked	N·m
w	REAL			Absolute angular velocity	rad/s
w_relfric	REAL			Relative angular velocity between frictional surface	

4.11.6 Formulation

The mathematical model associated to this component is shown below:

- Angular velocity at inlet port: m_in.omega = w
- Angular velocity at outlet port: m_out.omega = w

The two mechanical connections (flanges) have the same angular speed.

The normal force "fn" has to be provided as input signal "InPort.signal[1]" in a normalized form:

```
fn = fn_max* InPort.signal[1]
```

where "fn_max" has to be provided as input data. Friction in the brake is modelled in the following way:

When the absolute angular velocity "w" is not zero, the friction torque "tau" is a function of the velocity dependent friction coefficient given by the table "mue_pos" , of the normal force "fn", and of a geometry constant "cgeo" which takes into account the geometry of the device and the assumptions on the friction distributions:

```
tau = cgeo*fn*linearInterp1D(mue_pos, w)
```

The value of the friction coefficient is computed by interpolating linearly in the table "mue_pos" with the value of the angular velocity:

```
mue = linearInterp1D(mue_pos, w)
```

Typical values of coefficients of friction:

- dry operation : mue = 0.2 .. 0.4
- operating in oil: mue = 0.05 .. 0.1

When plates are pressed together, where ri is the inner radius, ro is the outer radius and N is the number of friction interfaces, the geometry constant is calculated in the following way under the assumption of a uniform rate of wear at the interfaces:

```
cgeo = N*(r0 + ri)/2
```

When the absolute angular velocity becomes zero, the elements connected by the friction element become stuck, i.e., the absolute angle remains constant. In this phase the friction torque is calculated from a torque balance due to the requirement, that the absolute acceleration shall be zero. The elements begin to slide when the friction torque "tau" exceeds a threshold value "tau0_max", called the maximum static friction torque, computed via:

```
tau0_max = peak * cgeo * fn * linearInterp1D(mue_pos, 0.0)
```

This procedure is implemented by state events and leads to continuous/discrete systems of equations.

4.12 Component R_Clutch

Inherited from R_AbsFriction and R_Compliant.

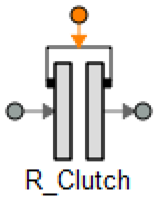
4.12.1 Description

This component rerpresents a clutch that is a mechanism for transmitting rotation, which can be engaged and disengaged. It is a component with mechanical ports (flanges) where friction is present between the two flanges and these flanges are pressed together via a normal force. The normal force "fn" has to be provided as input signal "inPort.signal[1]" in a normalized form:

```
fn = fn_max* inPort.signal[1]
```

where fn_max has to be provided as input data.

4.12.2 Symbol



4.12.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	DESCRIPTION
inPort	PORTS_LIB.analog_signal	(n = 1)	IN	Normalized force signal (= normal_force/fn_max; clutch is engaged if > 0)
m_in	PORTS_LIB.mech_rot		IN	Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT	Right / driving mech_rot

4.12.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
cgeo	REAL	1		Geometry constant containing friction distribution assumption	
fn_max	REAL	20		Maximum normal force	N
mue_pos	TABLE_1D			[w,mue] positive sliding friction coefficient: w_rel>=0	
peak	REAL	1.1		Maximum value of mue for w_rel=0	-
phi_rel_i	REAL	0		Initial angular distance between ports	rad
w_rel_i	REAL	1		Initial relative angular velocity	rad/s

4.12.5 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
T	REAL			Rotational transmited torque	N·m
a_rel	REAL			Relative angular acceleration	rad/s^2
a_relfric	REAL			Relative angular acceleration between frictional surfaces	rad/s^2
fn	REAL			Normal force	N
free	BOOLEAN	FALSE		TRUE, if frictional element is not active	-
imode	INTEGER	-2		Operation mod	-
mue0	REAL			Friction coefficient for w=0 and forward sliding	-
phi_rel	REAL			Angular distance between port	rad
sa	REAL			Path parameter of friction characteristic	-
tau	REAL			Friction torque: positive, if directed in opposite direction of w_rel	N·m
tau0	REAL			Friction torque for w=0 and forward sliding	N·m
tau0_max	REAL			Maximum friction torque for w=0 and locked	N·m
w_rel	REAL			Relative angular velocity	rad/s
w_relfric	REAL			Relative angular velocity between frictional surface	

4.12.6 Formulation

The mathematical model associated to this component is given as follows:

- Relative angle: phi_rel = m_out.phi - m_in.phi
- Torque in inlet port: m_in.tau = tau
- Torque in outlet port: m_out.tau = tau

The outlet torque is equal to the inlet torque.

The friction in the clutch is modelled in the following way:

When the relative angular velocity "w_rel" is not zero, the friction torque "tau" is a function of the velocity dependent friction coefficient given by the table "mue_pos" , of the normal force "fn", and of a geometry constant "cgeo" which takes into account the geometry of the device and the assumptions on the friction distributions:

```
tau = cgeo * linearInterp1D(mue_pos, w_rel) * fn
```

The value of the friction coefficient "mue" is computed by interpolating linearly in the table "mue_pos" with the value of the relative angular velocity:

```
mue = linearInterp1D(mue_pos, w_rel)
```

Typical values of coefficients of friction:

- dry operation : mue = 0.2 .. 0.4
- operating in oil: mue = 0.05 .. 0.1

When plates are pressed together, where ri is the inner radius, ro is the outer radius and N is the number of friction interfaces, the geometry constant is calculated in the following way under the assumption of a uniform rate of wear at the interfaces:

```
cgeo = N*(r0 + ri)/2
```

When the relative angular velocity becomes zero, the elements connected by the friction element become stuck, i.e., the relative angle remains constant. In this phase the friction torque is calculated from a torque balance due to the requirement, that the relative acceleration shall be zero. The elements begin to slide when the friction torque exceeds a threshold value "tau0_max", called the maximum static friction torque, computed via:

```
tau0_max = peak * cgeo * fn * linearInterp1D(mue_pos, 0.0)
```

This procedure is implemented by state events and leads to continuous/discrete systems of equations.

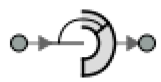
4.13 Component R_Damper

Inherited from R_Compliant.

4.13.1 Description

This component represents an ideal rotational damper. A damper is a general term for a mechanical device that damps vibrations or oscillations.

4.13.2 Symbol



4.13.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driving mech_rot

4.13.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
d	REAL	0		Damping constant	N*m*s/rad
phi_rel_i	REAL	0		Initial angular distance between ports	rad

4.13.5 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
T	REAL			Rotational transmited torque	N·m
phi_rel	REAL			Angular distance between port	rad
w_rel	REAL			Relative angular velocity between port	rad/s

4.13.6 Formulation

The mathematical model associated to this component is given as follows:

- Relative angle: phi_rel = m_out.phi - m_in.phi
- Torque in inlet port: m_in.tau = tau
- Torque in outlet port: m_out.tau = tau

The outlet torque is equal to the inlet torque.

This component calculates a torque T depending on the relative velocity of its ports "w_rel":

```
T = - d * w_rel
```

Where "d" is the damping constant and "w_rel" is the relative velocity of the ports.

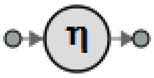
4.14 Component R_Efficiency

Inherited from R_Rigid.

4.14.1 Description

This component represents the effect of mechanical losses in terms of efficiency.

4.14.2 Symbol



R_Efficiency

4.14.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driving mech_rot

4.14.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
eta	REAL	0		Efficiency: [0,1]	-

4.14.5 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
P_a	REAL			Power	W
w	REAL			Absolute angular velocity	rad/s

4.14.6 Formulation

The mathematical model associated to this component is shown below:

- Angular velocity at inlet port: m_in.omega = w
- Angular velocity at outlet port: m_out.omega = w

The two mechanical connections (flanges) have the same angular speed.

The angular velocity in both ports is the same. If Pin and Pout are the input and output power respectively, the efficiency (h) is defined as:

$$\eta = \frac{P_{out}}{P_{in}}$$

It is supposed that Pin $\geq$ Pout. Since both ports have the same angular velocity, the efficiency can be expressed in terms of torque:

$$\left.\begin{array}{l} P_{out} = \omega \cdot \mathrm{T}_{out} \\ P_{in} = \omega \cdot \mathrm{T}_{in} \end{array}\right\} \Rightarrow \eta = \frac{T_{out}}{T_{in}} \Rightarrow T_{out} = \eta \cdot T_{in}$$

Tin and Tout can be defined by any of the two ports. The port that produces positive work is considered "in".

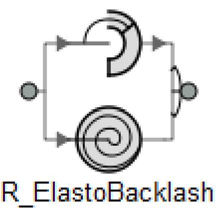
4.15 Component R_ElastoBacklash

Inherited from R_Compliant.

4.15.1 Description

This component represents a backlash element connected in series to a spring and damper element which are connected in parallel

4.15.2 Symbol



4.15.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driving mech_rot

4.15.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
c	REAL	0		Spring constant	N·m/rad
d	REAL	0		Damping constant	N·m*s/rad
ddz	REAL	0		Direct dead zone	rad
idz	REAL	0		Inverse dead zone	rad
phi_rel0	REAL	0		Angular distance between ports for unstretched spring	rad
phi_rel_i	REAL	0		Initial angular distance between ports	rad

4.15.5 Variables

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
T	REAL			Rotational transmited torque	N·m
phi_rel	REAL			Angular distance between ports	rad
w_rel	REAL			Relative angular velocity between ports	rad/s

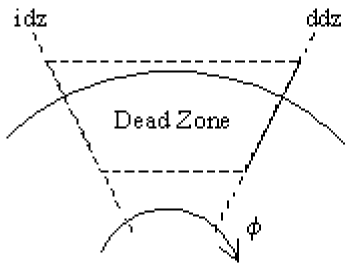
4.15.6 Formulation

The mathematical model associated to this component is given as follows:

- Relative angle: phi_rel = m_out.phi - m_in.phi
- Torque in inlet port: m_in.tau = tau
- Torque in outlet port: m_out.tau = tau

The outlet torque is equal to the inlet torque.

To define the dead zone, the user must enter a "direct dead zone" (ddz) and an "inverse dead zone" (idz) angles as component data.



When the relative rotation angle "phi_rel" is out of the dead zone, this component calculates a torque T as follows:

T = - c * (phi_rel - phi_rel0 - ddz) - d * w_rel for phi_rel >= ddz

T = - c * (phi_rel - phi_rel0 + idz) - d * w_rel for phi_rel <= idz

Where c is the spring constant, phi_rel0 is the relative position when the spring is unloaded and d is the damping constant.

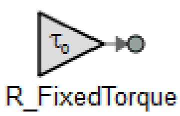
When the relative rotation angle "phi_rel" is within the dead zone, the torque is zero.

4.16 Component R_FixedTorque

4.16.1 Description

This component generates a constant torque output. In other words, this component represents a boundary condition in torque.

4.16.2 Symbol



4.16.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_rot		OUT		Rotational outlet port

4.16.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
T0	REAL	0		Fixed torque value	N·m

4.16.5 Formulation

The value of the torque at the outlet port is fixed by the value of the input data called T0:

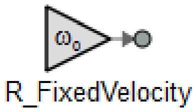
```
m_out.T = T0
```

4.17 Component R_FixedVelocity

4.17.1 Description

This component generates a constant angular velocity in the port. In other words, this component represents a boundary condition in angular velocity.

4.17.2 Symbol



4.17.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_rot		OUT		Rotational outlet port

4.17.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
w0	REAL	0		Fixed angular velocity	rad/s

4.17.5 Formulation

The value of the velocity at the outlet port is fixed by the value of the input data called w0:

```
m_out.omega = w0
```

4.18 Component R_Gear

Inherited from R_Two_Flanges.

4.18.1 Description

This component models the essential effects of a gearbox, in particular gear efficiency due to friction between the teeth, bearing friction, gear elasticity and damping, backlash. The inertia of the gear wheels is not modeled. Inertia can be taken into account by connecting components of type R_Inertia to the inlet and/or the outlet ports.

4.18.2 Symbol

4.18.3 Ports

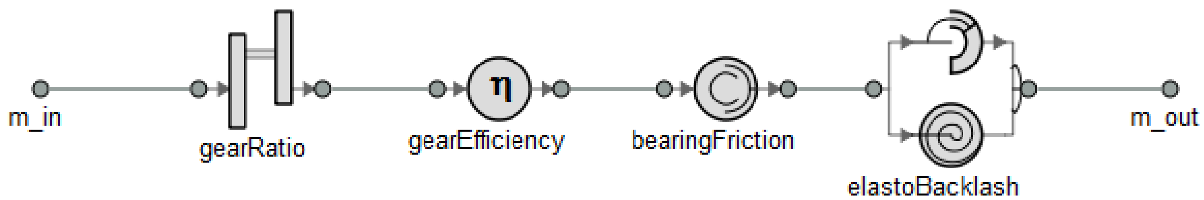
NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driving mech_rot

4.18.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
b	REAL	0		Total backlash	rad
c	REAL	0		Gear elasticity	N·m/rad
d	REAL	0		Gear damping	N·m·s/rad
eta	REAL	1		Gear efficiency	-
friction_pos	REAL			Positive sliding friction characteristic	
peak	REAL	1		Maximum friction torque at zero velocity	-
ratio	REAL	1		Transmision ratio	-

4.18.5 Formulation

This is a topological component and the following figure shows how the gear sub-system has been built:



4.19 Component R_GearFriction

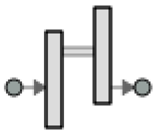
Inherited from R_Two_Flanges.

4.19.1 Description

This component represents a gear box that is a toothed wheel designed to transmit torque to another gear or toothed component.

It models the gear ratio and the losses of a standard gear box in a reliable way including the stuck phases that may occur at zero speed.

4.19.2 Symbol



4.19.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driving mech_rot

4.19.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
ideal	BOOLEAN	FALSE		TRUE, if losses are neglected	
tab_eta_mf1	TA-BLE_1D			Data of the mess efficiency in case of input shaft driving as a function of angular velocity	
tab_eta_mf2	TA-BLE_1D			Data of the mess efficiency in case of output shaft driving as a function of angular velocity	
tab_tau_bf1	TA-BLE_1D			Data of the absolute bearing friction torque in case of input shaft driving as a function of angular velocity	-
tab_tau_bf2	TA-BLE_1D			Data of the bsolute bearing friction torque in case of output shaft driving as a function of angular velocity	
trans_ratio	REAL	2		Transmission ratio	-

4.19.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
T	REAL			Torque	N·m
Tloss	REAL			Torque loss due to friction in the gear teeth and in the bearings	N·m
Tloss_max	REAL			Torque loss for positive speed	N·m
Tloss_min	REAL			Torque loss for negative speed	N·m
a	REAL			Angular acceleration	rad/s^2
eta_mf1	REAL			Mesh efficiency in case of input shaft driving	-
eta_mf2	REAL			Mesh efficiency in case of output shaft drivin	-
imode	INTEGER			Operation mode	
lossTable_ideal[4]	CONST REAL	{ 1,1,0,0}		Array for mesh efficiencies and bearing friction if losses are neglected	
q1	REAL			First quadrant	N·m
q2	REAL			Second quadrant	N·m
q3	REAL			Third quadrant	N·m
q4	REAL			Fourth quadrant	N·m
sa	REAL			Path parameter for acceleration and torque loss	
tau_bf1	REAL			Absolute bearing friction torque in case of input shaft driving	N·m
tau_bf2	REAL			Absolute bearing friction torque in case of output shaft driving	N·m
w	REAL			Angular velocity of inlet port	rad/s

4.19.6 Formulation

The gear boxes that can be handeled are fixed in the ground, have one input and one output shaft, and are essentially described by the equations:

```
m_in.omega = trans_ratio * m_out.omega
m_out.T = trans_ratio * (eta_mf * m_in.T - tau_bf) = 0
```

where:

trans_ratio = the constant gear ratio

eta_mf = the mesh efficiency due to the friction between the teeth of the gear wheels

tau_bf = the bearing friction torque

The loss terms "eta_mf" and "tau_bf" are functions of the absolute value of the input shaft speed "w" and of the energy flow direction. They are defined by the following input tables that have the following meaning:

- tab_eta_mf1 = Data of the mess efficiency in case of input shaft driving "eta_mf1" as a function of angular velocity
- tab_eta_mf2 = Data of the mess efficiency in case of output shaft driving "eta_mf2"as a function of angular velocity
- tab_tau_bf1 = Data of the absolute bearing friction torque in case of input shaft driving "|tau_bf1|" as a function of angular velocity
- tab_tau_bf2 = Data of the bsolute bearing friction torque in case of output shaft driving "|tau_bf2|" as a function of angular velocity

With these variables, the mesh efficiency "eta_mf" and the bearing friction "tau_bf" are formally defined as:

```
IF (T * w > 0 OR (T == 0 AND w > 0))
                        eta_mf = eta_mf1
                        tau_bf = tau_bf1
            ELSEIF (T * w < 0 OR (T == 0 AND w < 0))
                        eta_mf = 1/eta_mf2
                        tau_bf = tau_bf2
            ELSE
                        eta_mf and tau_bf are calculated such that a = 0. The gear is
                           stuck.
```

where:

a = angular acceleration

T = torque at inlet port

w = angular velocity

The losses are modeled taking into account that at null speed the movement may be locked due to the friction in the gear teeth and/or in the bearings.

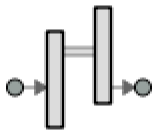
4.20 Component R_GearIdeal

Inherited from R_Two_Flanges.

4.20.1 Description

This component represents a frictionless gearbox that transforms a rotational movement.

4.20.2 Symbol



4.20.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driving mech_rot

4.20.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
ratio	REAL	1		Transmission ratio	-

4.20.5 Formulation

This component represents a frictionless gearbox that transforms a rotational movement at angular velocity "m_out.omega" into another at angular velocity "m_in.omega" as follows:

```
        m_in.omega = ratio * m_out.omega
```

where ratio is the transmission ratio.

The torque is calculated accordingly as:

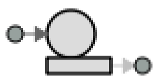
```
ratio * m_in.T = m_out.T
```

4.21 Component R_GearIdealR2T

4.21.1 Description

This component represents a frictionless gearbox that transforms a rotational movement into a linear movement.

4.21.2 Symbol



4.21.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
R_m_in	PORTS_LIB.mech_rot		IN		Inlet rotational port
T_m_out	PORTS_LIB.mech_trans		OUT		Outlet translational port

4.21.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
ratio	REAL	1		Transmission ratio	rad/m

4.21.5 Formulation

The angular velocity and the displacement are linked by means of the equation:

```
T_m_out.s' = R_m_in.omega / ratio
```

where ratio is the transmission ratio (in rad/m).

The relation between torque and force is given by:

```
T_m_out.F = ratio * R_m_in.T
```

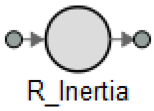
4.22 Component R_Inertia

Inherited from R_Rigid.

4.22.1 Description

This component represents a rotating mass with inertia.

4.22.2 Symbol



4.22.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driven mech_rot

4.22.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
I	REAL	1		Moment of inertia of body	kg·m^2
w0	REAL	0		Initial angular velocity	rad/s

4.22.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
a	REAL			Absolute angular acceleration	rad/s^2
w	REAL			Absolute angular velocity	rad/s

4.22.6 Formulation

The mathematical model associated to this component is shown below:

- Angular velocity at inlet port: m_in.omega = w
- Angular velocity at outlet port: m_out.omega = w

The two mechanical connections (flanges) have the same angular speed.

This component calculates the angular acceleration (a) and angular velocity (w) of a rotating body (inertia I) depending on the momentum in the ports as follows:

```
w' = a
I * a = m_in.T - m_out.T
```

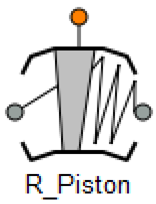
4.23 Component R_Piston

Inherited from T_Rigid.

4.23.1 Description

Dynamic model of an angular Piston submitted to the forces of pressure, spring, friction...

4.23.2 Symbol



4.23.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Left / driving mech_trans
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans
s_Mext	PORTS_LIB.analog_signal		IN		Piston external torque signal

4.23.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
Ip	REAL	0.01		Movil parts inertia	kg·m^2
K_spr	REAL	1000		Spring constan	N·m/rad
Mpl	REAL	10		Piston preload at themin	N·m
R	REAL	0.1		Effective radiu	m
cv	REAL	20		Viscous friction coeffien	N/m·s
fc1	REAL	0		Direct coulomb friction torque	N
fc2	REAL	0		Inverse coulomb friction torque	N
theini	REAL	0		Piston initial position	rad
themax	REAL	1		Piston position upper limit	rad
themin	REAL	0		Piston position lower limit	rad

4.23.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
M	REAL			Net M on piston	N·m
s	REAL			Absolute position of component	m
the	REAL			Angular piston position	rad
the0	REAL			Angular piston precharge	rad

4.23.6 Formulation

The mathematical formulation associated to this component is the following:

- Position at inlet port: m_in.s = s
- Position at outlet port: m_out.s = s

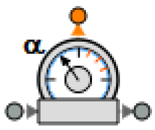
4.24 Component R_RelativeSensorAcceleration

Inherited from R_RelativeSensor.

4.24.1 Description

This component represents a relative acceleration sensor. It measures the relative angular acceleration a_rel between the two ports in an ideal way.

4.24.2 Symbol



R_RelativeSensorAcceleration

4.24.3 Construction Parameters

NAME	TYPE	DEFAULT	DESCRIPTION	UNITS
n_out	INTEGER	1	Dimension of outputs	-

4.24.4 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Rotational mechanical inlet port
m_out	PORTS_LIB.mech_rot		OUT		Rotational mechanical outlet port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT		Outlet signal ()

4.24.5 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-

4.24.6 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
a_rel	REAL			Relative angular acceleration	rad/s^2
var[n_out]	REAL			Measured variable	-
w_rel	REAL			Relative speed acceleration	rad/s

4.24.7 Formulation

The equations associated to this component are shown below:

- Relative velocity: w_rel = m_out.omega - m_in.omega
- Relative acceleration: a_rel = w_rel′
- Torque at the inlet port: m_in.T = 0
- Torque at the outlet port: m_out.T = 0
- Output signal: var = a_rel

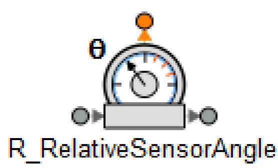
4.25 Component R_RelativeSensorAngle

Inherited from R_RelativeSensor.

4.25.1 Description

This component represents a relative angle sensor. It measures the relative angle phi_rel between the two ports in an ideal way.

4.25.2 Symbol



4.25.3 Construction Parameters

NAME	TYPE	DEFAULT	DESCRIPTION	UNITS
n_out	INTEGER	1	Dimension of outputs	-

4.25.4 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Rotational mechanical inlet port
m_out	PORTS_LIB.mech_rot		OUT		Rotational mechanical outlet port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT		Outlet signal ()

4.25.5 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-
phi_rel_0	REAL	0		Initial relative angle	rad

4.25.6 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
phi_rel	REAL			Relative angle	rad
var[n_out]	REAL			Measured variable	-

4.25.7 Formulation

The equations associated to this component are shown below:

- Relative angle: phi_rel' = m_out.omega - m_in.omega
- Torque at the inlet port: m_in.T = 0
- Torque at the outlet port: m_out.T = 0
- Output signal: var =phi_rel

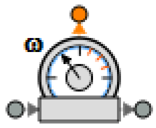
4.26 Component R_RelativeSensorSpeed

Inherited from R_RelativeSensor.

4.26.1 Description

This component represents a relative velocity sensor. It measures the relative angular velocity w_rel between the two ports in an ideal way.

4.26.2 Symbol



4.26.3 Construction Parameters

NAME	TYPE	DEFAULT	DESCRIPTION	UNITS
n_out	INTEGER	1	Dimension of outputs	-

4.26.4 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Rotational mechanical inlet port
m_out	PORTS_LIB.mech_rot		OUT		Rotational mechanical outlet port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT		Outlet signal ()

4.26.5 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-

4.26.6 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
var[n_out]	REAL			Measured variable	-
w_rel	REAL			Relative angular speed	rad/s

4.26.7 Formulation

The equations associated to this component are shown below:

- Torque at the inlet port: m_in.T = 0
- Torque at the outlet port: m_out.T = 0
- Relative velocity: w_rel = m_out.omega - m_in.omega
- Output signal: var =w_rel

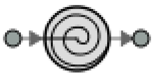
4.27 Component R_Spring

Inherited from R_Compliant.

4.27.1 Description

This component represents an ideal rotational spring.

4.27.2 Symbol



4.27.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driven mech_rot

4.27.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
c	REAL	0		Spring constant	N·m/rad
phi_rel0	REAL	0		Angular distance between ports for unstretched spring	rad
phi_rel_i	REAL	0		Initial angular distance between ports	rad

4.27.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
T	REAL			Rotational transmited torque	N·m
phi_rel	REAL			Angular distance between port	rad

4.27.6 Formulation

The mathematical model associated to this component is given as follows:

- Relative angle: phi_rel = m_out.phi - m_in.phi
- Torque in inlet port: m_in.tau = tau
- Torque in outlet port: m_out.tau = tau

The outlet torque is equal to the inlet torque.

This component calculates a torque T depending on the relative position of its ports phi_rel:

```
T = - c * (phi_rel - phi_rel0)
```

Where c is the spring constant and phi_rel0 is the relative position when the spring is unloaded.

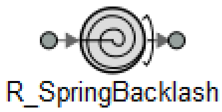
4.28 Component R_SpringBacklash

Inherited from R_Compliant.

4.28.1 Description

Helicoidal spring with backlash. This component represents a spring with a dead zone where the spring does not act.

4.28.2 Symbol



4.28.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driven mech_rot

4.28.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
c	REAL	0		Spring constant	N·m/rad
ddz	REAL	0		Direct dead zone	rad
idz	REAL	0		Inverse dead zone	rad
phi_rel0	REAL	0		Angular distance between ports for unstretched spring	rad
phi_rel_i	REAL	0		Initial angular distance between ports	rad

4.28.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
T	REAL			Rotational transmited torque	N·m
phi_rel	REAL			Angular distance between port	rad

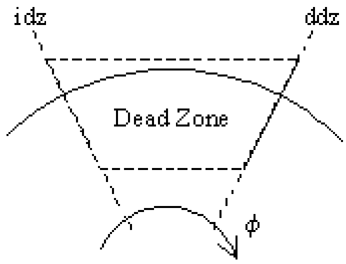
4.28.6 Formulation

The mathematical model associated to this component is given as follows:

- Relative angle: phi_rel = m_out.phi - m_in.phi
- Torque in inlet port: m_in.tau = tau
- Torque in outlet port: m_out.tau = tau

The outlet torque is equal to the inlet torque.

To define the dead zone, the user must enter a "direct dead zone" (ddz) and an "inverse dead zone" (idz) angles as component data.



When the relative rotation angle "phi_rel" is out of the dead zone, this component calculates a torque T as follows:

- T = - c * (phi_rel - phi_rel0 - ddz) for phi_rel >= ddz
- T = - c * (phi_rel - phi_rel0 + idz) for phi_rel <= idz

Where c is the spring constant and phi_rel0 is the relative position when the spring is unloaded.

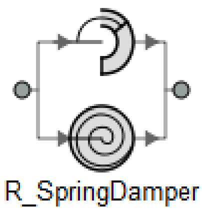
4.29 Component R_SpringDamper

Inherited from R_Compliant.

4.29.1 Description

This component represents a spring and a damper element connected in parallel.

4.29.2 Symbol



4.29.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driven mech_rot

4.29.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
c	REAL	0		Spring constant	N·m/rad
d	REAL	0		Damping constant	N·m·s/rad
phi_rel0	REAL	0		Angular distance between ports for unstretched spring	rad
phi_rel_i	REAL	0		Initial angular distance between ports	rad

4.29.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
T	REAL			Rotational transmited torque	N·m
phi_rel	REAL			Angular distance between port	rad
w_rel	REAL			Relative angular velocity between ports	rad/s

4.29.6 Formulation

The mathematical model associated to this component is given as follows:

- Relative angle: phi_rel = m_out.phi - m_in.phi
- Torque in inlet port: m_in.tau = tau
- Torque in outlet port: m_out.tau = tau

The outlet torque is equal to the inlet torque.

This component calculates the torque T as follows:

```
T = - c * (phi_rel - phi_rel0 ) - d * w_rel
```

Where c is the spring constant, phi_rel0 is the relative position when the spring is unloaded, d is the damping constant and w_rel is the relative angular velocity between the two ports.

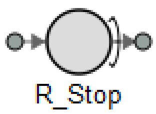
4.30 Component R_Stop

Inherited from R_Rigid.

4.30.1 Description

Rotating mass with inertia and stops

4.30.2 Symbol



4.30.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_rot		IN		Left / driving mech_rot
m_out	PORTS_LIB.mech_rot		OUT		Right / driven mech_rot

4.30.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
I	REAL	1		Moment of inertia of body	kg·m^2
phi0	REAL	0		Initial angular position	rad
phi_max	REAL	0.5		Upper limit angular position	rad
phi_min	REAL	-0.5		Lower limit angular position	rad
w0	REAL	0		Initial angular velocity	rad/s

4.30.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
M	REAL			Net Moment on leve	N·m
a	REAL			Absolute angular acceleration	rad/s^2
phi	REAL			Absolute angular position	rad
w	REAL			Absolute angular velocity	rad/s

4.30.6 Formulation

The mathematical model associated to this component is shown below:

- Angular velocity at inlet port: m_in.omega = w
- Angular velocity at outlet port: m_out.omega = w

The two mechanical connections (flanges) have the same angular speed.

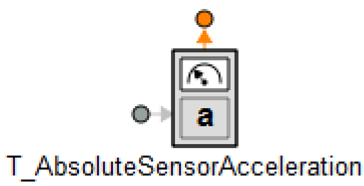
4.31 Component T_AbsoluteSensorAcceleration

Inherited from T_AbsoluteSensor.

4.31.1 Description

This component simulates an acceleration sensor. The measure of the acceleration takes place in one single port. This component has one mechanical input port and one signal output port.

4.31.2 Symbol



4.31.3 Construction Parameters

NAME	TYPE	DEFAULT	DESCRIPTION	UNITS
n_out	INTEGER	1	Dimension of outputs	-

4.31.4 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Mechanical left port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT		Outlet signal ()

4.31.5 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-

4.31.6 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
a	REAL			Absolute acceleration	m/s^2
var[n_out]	REAL			Measured variable	-

4.31.7 Formulation

The mathematical formulation associated to this component is the following:

- Acceleration: a = m_in.s''
- Force: m_in.F = 0
- Output signal: var[1] =a

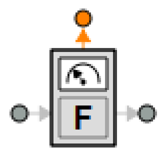
4.32 Component T_AbsoluteSensorForce

Inherited from T_AbsoluteSensor.

4.32.1 Description

This component simulates a force sensor. The measure of the force takes place in two ports. This component has two mechanical ports and one signal port. Additionally, it must be placed in series in the branch where the force has to be measured.

4.32.2 Symbol



T_AbsoluteSensorForce

4.32.3 Construction Parameters

NAME	TYPE	DEFAULT	DESCRIPTION	UNITS
n_out	INTEGER	1	Dimension of outputs	-

4.32.4 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Mechanical left port
m_out	PORTS_LIB.mech_trans		OUT		Mechanical right port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT		Outlet signal ()

4.32.5 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-

4.32.6 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
F	REAL			Measured force	N
var[n_out]	REAL			Measured variable	-

4.32.7 Formulation

The mathematical formulation associated to this component is the following:

- Position: m_out.s = m_in.s
- Force at the inlet port: m_in.F = F
- Force at the outlet port: m_out.F = F
- Output signal: var[1] = F

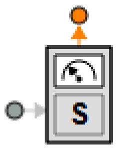
4.33 Component T_AbsoluteSensorPosition

Inherited from T_AbsoluteSensor.

4.33.1 Description

AbsoluteSensorPosition simulates a position sensor. The measure of the position takes place in one single port. This component has one mechanical input port and one signal output port.

4.33.2 Symbol



4.33.3 Construction Parameters

NAME	TYPE	DEFAULT	DESCRIPTION	UNITS
n_out	INTEGER	1	Dimension of outputs	-

4.33.4 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Mechanical left port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT		Outlet signal ()

4.33.5 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-

4.33.6 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
s	REAL			Absolute position	m
var[n_out]	REAL			Measured variable	-

4.33.7 Formulation

The mathematical formulation associated to this component is the following:

- Position: s = m_in.s
- Force: m_in.F = 0
- Output signal: var[1] = s

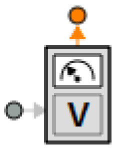
4.34 Component T_AbsoluteSensorVelocity

Inherited from T_AbsoluteSensor.

4.34.1 Description

AbsoluteSensorVelocity simulates a velocity sensor. The measure of the velocity takes place in one single port. This component has one mechanical input port and one signal output port..

4.34.2 Symbol



4.34.3 Construction Parameters

NAME	TYPE	DEFAULT	DESCRIPTION	UNITS
n_out	INTEGER	1	Dimension of outputs	-

4.34.4 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Mechanical left port
s_out	PORTS_LIB.analog_signal	(n = n_out)	OUT		Outlet signal ()

4.34.5 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
bias[n_out]	REAL	0		Bias	-
gain[n_out]	REAL	1		Gain	-

4.34.6 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
v	REAL			Absolute velocity	m/s
var[n_out]	REAL			Measured variable	-

4.34.7 Formulation

The mathematical formulation associated to this component is the following:

- Velocity: v = m_in.s'
- Force: m_in.F = 0
- Output signal: var[1] = v

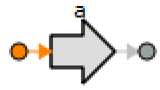
4.35 Component T_ActuatorAcceleration

Inherited from T_Actuator.

4.35.1 Description

This component forces a movement in a port from an input acceleration signal. This component has one mechanical port and one signal port.

4.35.2 Symbol



4.35.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_trans		OUT		Output translational mechanical port
s_in	PORTS_LIB.analog_signal	(n = n_out)	IN		Input signal port ()

4.35.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
s_0	REAL	0		Start position	m
v_0	REAL	0		Start velocity	m/s

4.35.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
a	REAL			Absolute acceleration	m/s^2
s	REAL			Absolute position	m
v	REAL			Absolute speed	m/s

4.35.6 Formulation

The mathematical model associated to this component is given as follows:

- Velocity: v′ = a
- Acceleration: a = s_in.signal[1]
- Position: s″ = a
- Port position: m_out.s = s

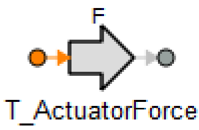
4.36 Component T_ActuatorForce

Inherited from T_Actuator.

4.36.1 Description

This component generates a mechanical force from an input signal. This component has one mechanical port and one signal port.

4.36.2 Symbol



4.36.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_trans		OUT		Output translational mechanical port
s_in	PORTS_LIB.analog_signal	(n = n_out)	IN		Input signal port ()

4.36.4 Formulation

The mathematical model associated to this component is given as follows:

- Outlet force: m_out.F = s_in.signal[1]

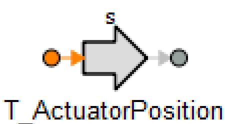
4.37 Component T_ActuatorPosition

Inherited from T_Actuator.

4.37.1 Description

ActuatorPosition forces a mechanical position in one port which is determined by an input signal. This component has one mechanical port and one signal port.

4.37.2 Symbol



4.37.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_trans		OUT		Output translational mechanical port
s_in	PORTS_LIB.analog_signal	(n = n_out)	IN		Input signal port ()

4.37.4 Formulation

The mathematical model associated to this component is given as follows:

- Position: m_out.s = s_in.signal[1]

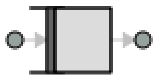
4.38 Component T_Damper

Inherited from T_Compliant.

4.38.1 Description

This component represents an ideal damper.

4.38.2 Symbol



4.38.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Left / driving mech_trans
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans

4.38.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
d	REAL	0		Damping constant	N·s/m

4.38.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
F	REAL			Translational transmited force	N
s_rel	REAL			Distance between ports	m
v_rel	REAL	0		Relative velocity between ports	m/s

4.38.6 Formulation

The mathematical formulation associated to this component are the following:

- Relative displacement: s_rel = m_out.s - m_in.s
- Force at inlet port: m_in.F = -F
- Force at outlet port: m_out.F = F

This component calculates a force depending on the relative velocity of its ports:

```
F = - d * v_rel
```

Where d is the damping constant and v_rel is the relative velocity of the ports, calculated as the derivate of the relative displacement.

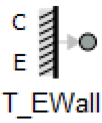
```
v_rel = s_rel'
```

4.39 Component T_EWall

4.39.1 Description

This element simulates a solid wall that allows shocks and bounces with it by means of calculating the reaction force of the ground.

4.39.2 Symbol



4.39.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans

4.39.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
C	REAL	1		Viscous coefficient	N·s/m
E	REAL	1.e308		Elastic coefficient	N/m
wp	REAL	0		Wall position	m
ws	INTEGER	-1		Wall solid side: -1=Left & 1=RighT	

4.39.5 Formulation

This element simulates a solid wall that allows shocks and bounces with it calculating a reaction force N. This reaction force consists of two terms: one is conservative (elastic) and the other is disipative (friction):

$$N = E \cdot (s - wp) + c \frac{v}{|v|}$$

Where s and v are respectively the position and the velocity of the hitting mass, wp is the wall position, E is the elastic coefficient of the wall and c is a dissipative coefficient.

The reaction force is calculated when the moving mass crosses the wall position to the wall solid side. When the position of the moving mass is in the wall-free area, N is set to 0. The status of the moving body is determined in the DISCRETE block.

The integer flag ws defines the solid side of the wall (in other words, where the mass movement is not possible). When ws = 0, the solid side is for s $\leq$ wp (movement forbidden in the left side). The movement is forbidden in the right side of wp when ws =1.

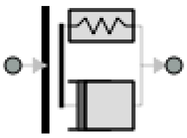
4.40 Component T_ElastoGap

Inherited from T_Compliant.

4.40.1 Description

This component represents a translational spring damper combination with gap.

4.40.2 Symbol



4.40.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Left / driving mech_trans
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans

4.40.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
d	REAL	0		Damping constant	N·s/m
k	REAL	0		Spring constant	N·m
s_rel0	REAL	0		Distance between ports for unloaded spring	m

4.40.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
F	REAL			Translational transmited force	N
contact	BOOLEAN				
s_rel	REAL			Distance between ports	m
v_rel	REAL	0		Relative velocity between ports	m/s

4.40.6 Formulation

The mathematical formulation associated to this component are the following:

- Relative displacement: s_rel = m_out.s - m_in.s
- Force at inlet port: m_in.F = -F
- Force at outlet port: m_out.F = F

This component calculates a force depending on the relative velocity and the relative position of its ports:

F = -k * (s_rel - s_rel0) - d * v_rel if s_rel < s_rel0

F = 0.0 if s_rel > s_rel0

where k is the spring constant, s_rel0 is the distance between ports for unloaded spring, d is the damping constant and v_rel is the relative velocity of the ports, calculated as the derivative of the relative displacement:

```
v_rel = s_rel'
```

4.41 Component T_FixedAcceleration

4.41.1 Description

This component generates a constant acceleration output. In other words, this component represents a boundary condition in acceleration, such as gravity.

4.41.2 Symbol

4.41.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_trans		OUT		

4.41.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
a0	REAL	0		Fixed acceleration value	m/s^2

4.41.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
s	REAL			Absolute position	m

4.41.6 Formulation

The value of the acceleration at the outlet port is fixed by the value of the input data called a0:

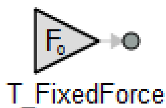
```
s''= a0
m_out.s = s
```

4.42 Component T_FixedForce

4.42.1 Description

This component generates a constant force output. In other words, this component represents a boundary condition in force.

4.42.2 Symbol



4.42.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_trans		OUT		

4.42.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
F0	REAL	0		Fixed force value	N

4.42.5 Formulation

The value of the force at the outlet port is fixed by the value of the input data called F0:

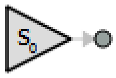
```
m_out.F = F0
```

4.43 Component T_FixedPosition

4.43.1 Description

This component generates a constant position output. In other words, this component represent a boundary condition in position, such as a ground fixation.

4.43.2 Symbol



4.43.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_trans		OUT		

4.43.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
s0	REAL	0		Fixed offset position	m

4.43.5 Formulation

The value of the position at the outlet port is fixed by the value of the input data called s0:

```
            m_out.s = s0
```

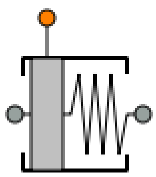
4.44 Component T_Piston

Inherited from T_Rigid.

4.44.1 Description

Dynamic model of a Piston submitted to the forces of pressure, spring, friction ...

4.44.2 Symbol



4.44.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Left / driving mech_trans
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans
S_Fext	PORTS_LIB.analog_signal		IN		Piston external force signal

4.44.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
Fpl	REAL	100		Piston preload at xmin	N
K_spr	REAL	100000		Spring constant	N·m
Mp	REAL	1		Piston mass	kg
cv	REAL	20		Viscous friction coeffient	N/m·s
fc1	REAL	1		Direct coulomb friction	N
fc2	REAL	1		Inverse coulomb friction	N
xini	REAL	0		Piston initial position	m
xmax	REAL	0.05		Piston position upper limit	m
xmin	REAL	0		Piston position lower limit	m

4.44.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
F	REAL			Translational transmited force	N
s	REAL			Absolute position of component	m
x	REAL			Relative velocity between ports	m
x0	REAL			Piston precharge	m

4.44.6 Formulation

The mathematical formulation associated to this component is the following:

- Position at inlet port: m_in.s = s
- Position at outlet port: m_out.s = s

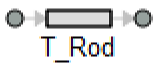
4.45 Component T_Rod

Inherited from T_Rigid.

4.45.1 Description

This component represents a rod without inertia and two rigidly connected flanges.

4.45.2 Symbol



4.45.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Left / driving mech_trans
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans

4.45.4 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
s	REAL			Absolute position of component	m

4.45.5 Formulation

The mathematical formulation associated to this component is the following:

- Position at inlet port: m_in.s = s
- Position at outlet port: m_out.s = s

The force balance applied to this component is shown next:

```
0 = m_in.F + m_out.F
```

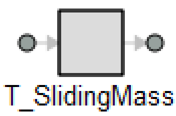
4.46 Component T_SlidingMass

Inherited from T_Rigid.

4.46.1 Description

This component represents a sliding mass in longitudinal movement.

4.46.2 Symbol



4.46.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Left / driving mech_trans
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans

4.46.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
M	REAL	1		Mass of body	kg
s0	REAL	0		Initial body position	m
v0	REAL	0		Initial body velocity	m/s

4.46.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
a	REAL			Absolute acceleration	m/s^2
s	REAL			Absolute position of component	m
v	REAL			Absolute velocity	m/s

4.46.6 Formulation

The mathematical formulation associated to this component is the following:

- Position at inlet port: m_in.s = s
- Position at outlet port: m_out.s = s

This component calculates the acceleration (a), velocity (v) and position (s) of a mass m depending on the forces in the ports as follows:

- Position: s' = v
- Velocity: v' = a
- Acceleration: m * a = m_in.F + m_out.F

4.47 Component T_Spring

Inherited from T_Compliant.

4.47.1 Description

This component represents an ideal spring..

4.47.2 Symbol

4.47.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Left / driving mech_trans
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans

4.47.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
k	REAL	0		Spring constant	N/m
s_rel0	REAL	0		Distance between ports for unloaded spring	m

4.47.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
F	REAL			Translational transmited force	N
s_rel	REAL			Distance between ports	m

4.47.6 Formulation

The mathematical formulation associated to this component are the following:

- Relative displacement: s_rel = m_out.s - m_in.s
- Force at inlet port: m_in.F = -F
- Force at outlet port: m_out.F = F

This component calculates a force depending on the relative position of its ports s_rel:

```
F = - k * (s_rel - s_rel0)
```

Where k is the spring constant and s_rel0 is the relative position when the spring is unloaded

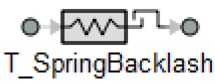
4.48 Component T_SpringBacklash

Inherited from T_Compliant.

4.48.1 Description

This component represents a spring with a dead zone where the spring does not act.

4.48.2 Symbol



4.48.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Left / driving mech_trans
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans

4.48.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
ddz	REAL	0		Direct dead zone	m
idz	REAL	0		Inverse dead zone	m
k	REAL			Spring constant	N/m
s_rel0	REAL	0		Distance between ports for unloaded spring	m

4.48.5 Variables

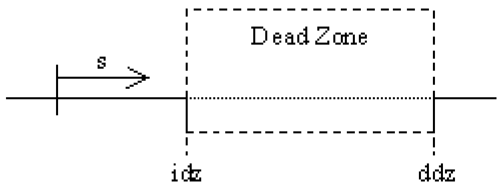
NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
F	REAL			Translational transmited force	N
s_rel	REAL			Distance between ports	m

4.48.6 Formulation

The mathematical formulation associated to this component are the following:

- Relative displacement: s_rel = m_out.s - m_in.s
- Force at inlet port: m_in.F = -F
- Force at outlet port: m_out.F = F

To define the dead zone of the spring, the user must enter a "direct dead zone" (ddz) and an "inverse dead zone" (idz) position as component data.



When the position s is out of the dead zone, this component calculates a force depending on the relative position of its ports s_rel:

F = k * (s_rel - s_rel0 - ddz) for s_rel >= ddz

F = k * (s_rel - s_rel0 - idz) for s_rel <= idz

Where k is the spring constant and s_rel0 is the relative position when the spring is unloaded.

When the position s is within the dead zone, the force is zero.

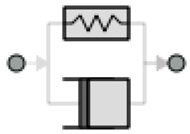
4.49 Component T_SpringDamper

Inherited from T_Compliant.

4.49.1 Description

This component represents a spring and a damper element connected in parallel.

4.49.2 Symbol



4.49.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Left / driving mech_trans
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans

4.49.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
d	REAL	0		Damping constant	N·s/m
k	REAL	0		Spring constant	N/m
s_rel0	REAL	0		Distance between ports for unloaded spring	m

4.49.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
F	REAL			Translational transmited force	N
s_rel	REAL			Distance between ports	m
v_rel	REAL			Relative velocity between ports	m/s

4.49.6 Formulation

The mathematical formulation associated to this component are the following:

- Relative displacement: s_rel = m_out.s - m_in.s
- Force at inlet port: m_in.F = -F
- Force at outlet port: m_out.F = F

This component calculates the force F as follows:

```
F =  - k * (s_rel - s_rel0) - d * v_rel
```

Where k is the spring constant, s_rel0 is the distance between ports for unloaded spring, d is the damping constant and v_rel is the relative velocity between the two ports.

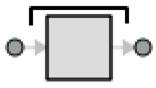
4.50 Component T_Stop

Inherited from T_Rigid.

4.50.1 Description

This component represents a sliding mass with hard stop and Stribeck friction.

4.50.2 Symbol



4.50.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_in	PORTS_LIB.mech_trans		IN		Left / driving mech_trans
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans

4.50.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
F_Coulomb	REAL	5		Constant friction: Coulomb force	N
F_Stribeck	REAL	10		Stribeck effect	N
F_prop	REAL	1		Velocity dependent friction	N/[m/s]
fexp	REAL	2		Exponential decay	1/[m/s]
m	REAL	1		Mass	kg
smax	REAL	25		Right stop for sliding mass	m
smin	REAL	-25		Left stop for sliding mass	m
sini	REAL	0		Initial position	m

4.50.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
a	REAL			Acceleration	m/s^2
f	REAL			External forces	N
fr	REAL			Friction force	N
s	REAL			Absolute position of component	m
v	REAL			Velocity	m/s
var_initial	BOOLEAN	TRUE		Initial state	

4.50.6 Formulation

The mathematical formulation associated to this component is the following:

- Position at inlet port: m_in.s = s
- Position at outlet port: m_out.s = s

This element describes the Stribeck friction characteristics of a sliding mass, i. e. the frictional force acting between the sliding mass and the support. Included is a hard stop for the position.

The surface is fixed and there is friction between sliding mass and surface. The frictional force fr is given for positive velocity v by:

```
fr = F_prop*v +  sign(v)*F_Stribeck*exp(-fexp*abs(v)) - rev_fri(v,F_Coulomb,F_Coulomb
  )
```

There are hard stops at smax and smin, i. e. if m_in.s >= smin and m_out.s <= xmax the sliding mass can move freely.

When the absolute velocity becomes zero, the sliding mass becomes stuck, i.e., the absolute position remains constant. In this phase the friction force is calculated from a force balance due to the requirement that the absolute acceleration shall be zero. The elements begin to slide when the friction force exceeds a threshold value.

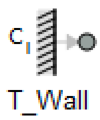
This requires the states position and velocity. If these states are eliminated during the index reduction the model will not work. To avoid this any inertias should be connected via springs to the T_Stop component, other sliding masses, dampers or hydraulic chambers must be avoided.

4.51 Component T_Wall

4.51.1 Description

This component simulates a solid wall that allows shocks and bounces with it by means of a restitution coefficient.

4.51.2 Symbol



4.51.3 Ports

NAME	TYPE	PARAMETERS	DIRECTION	CARDINALITY	DESCRIPTION
m_out	PORTS_LIB.mech_trans		OUT		Right / driven mech_trans

4.51.4 Data

NAME	TYPE	DEFAULT	RANGE	DESCRIPTION	UNITS
cl	REAL	1		Restitution coefficient: 0=inelastic <-> 1=elastic	-
wp	REAL	0		Wall position	m
ws	INTEGER	-1		Wall solid side: -1=Left & 1=Right	-

4.51.5 Variables

NAME	TYPE	INITIAL	RANGE	DESCRIPTION	UNITS
a	REAL			Absolute acceleration	m/s^2
s	REAL			Absolute position	m
v	REAL			Absolute velocity	m/s

4.51.6 Formulation

This element simulates a solid wall that allows shocks and bounces with it by means of a restitution coefficient (cl). When a mass with a velocity v hits the wall (that is, its position is equal to the wall position, wp), its velocity is set to - cl·v while keeping the same position. This calculation is performed in the DISCRETE block.

The integer flag ws defines the solid side of the wall (in other words, where the mass movement is not possible). When ws = 0, the solid side is for $s \leq wp$ (movement forbidden in the left side). The movement is forbidden in the right side of wp when ws =1.

This page intentionally left blank.

5. References

EcosimPro Dynamic Simulation Tool. User Manual. EA International. 1992-2015

This page intentionally left blank.

This page intentionally left blank.

EA Internacional
Magallanes, 3. 28015 Madrid. SPAIN.
E-mail: info@ecosimpro.com
Web: www.ecosimpro.com

EMPRESARIOS AGRUPADOS
Edition 2016