

MathModelica® System Designer™

Servo Mechanism

© 2006 MathCore Engineering AB

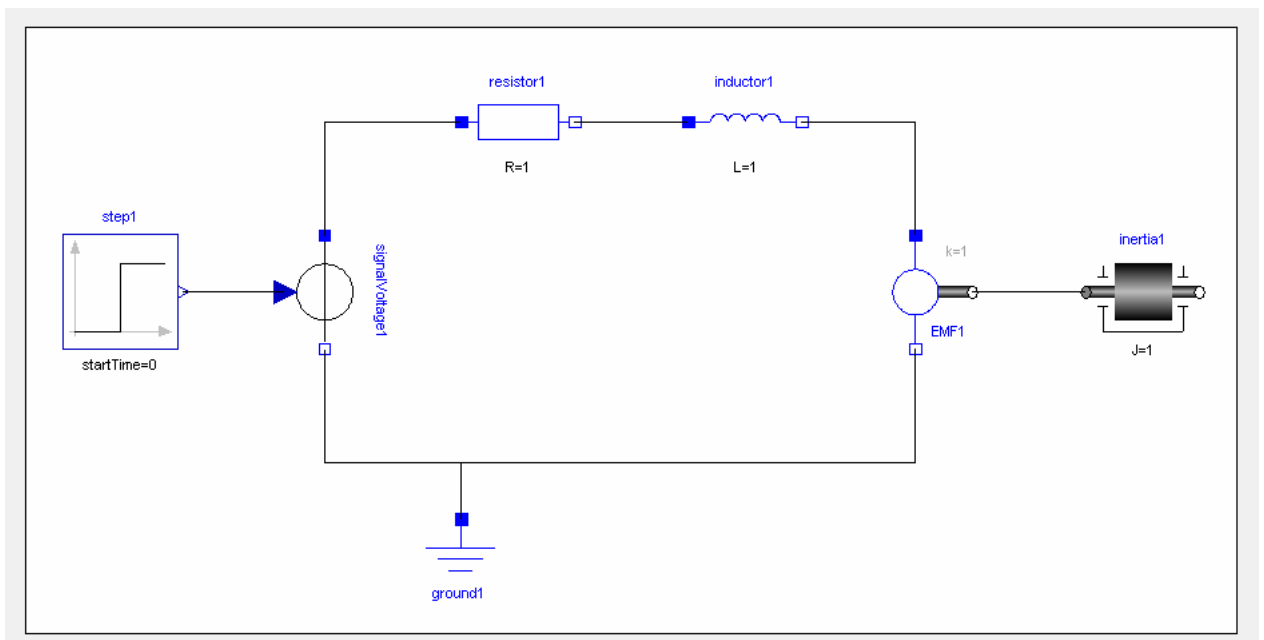
1 Introduction

This example shows how to develop a servo mechanism model step by step in MathModelica System Designer. It illustrates the multi engineering capabilities and shows how you can use Simulation Center to analyze models created in the Model Editor, synthesize controllers, and do comparison studies.

2 DC Motor

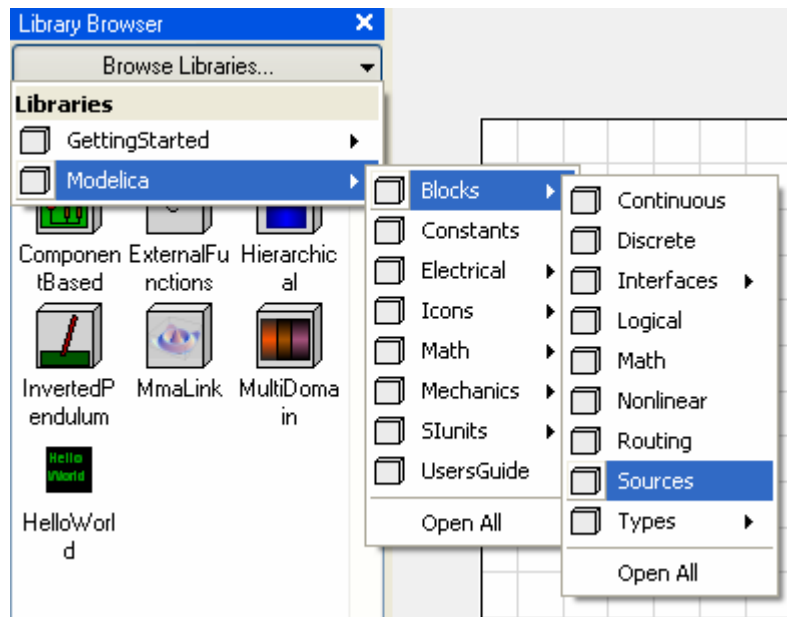
A simple dynamic model of a controlled DC motor consists of a variable voltage source, a resistor, an inductor, and an electro-motoric force element representing the coupling between electric energy and mechanical energy provided by the magnetic field in the DC motor. The motor axis is represented by a rotating mass or inertia.

All this components can be found in the Modelica Standard Library, included in MathModelica System Designer, and using drag-and-drop to compose the model as illustrated in the figure below.

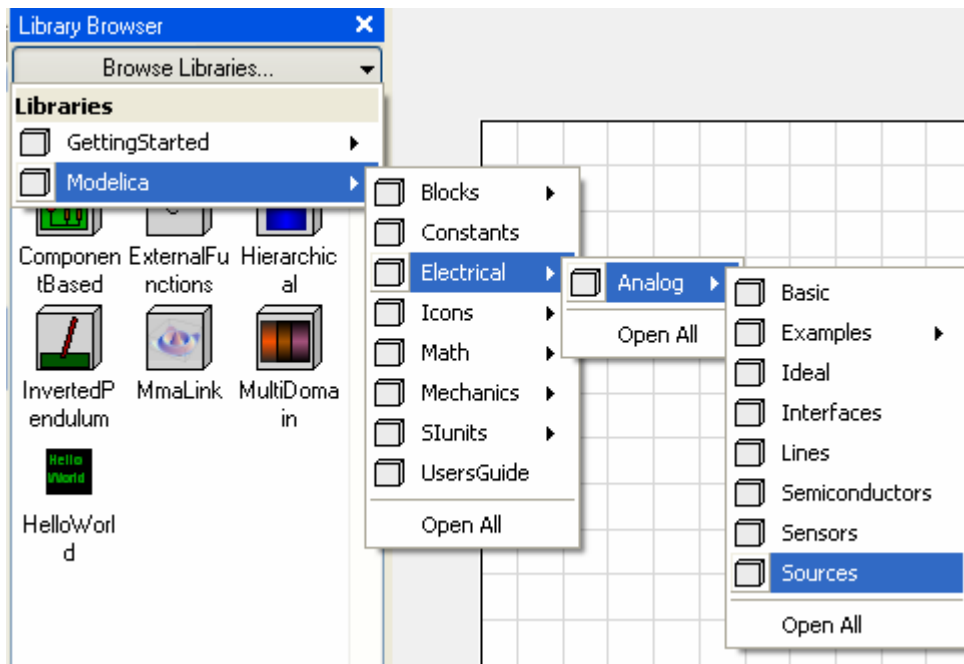


To build this model we need to find the appropriate libraries, drag and drop them into the diagram area, and then connect them using the connect tool. In this case you can find the components by opening the following libraries.

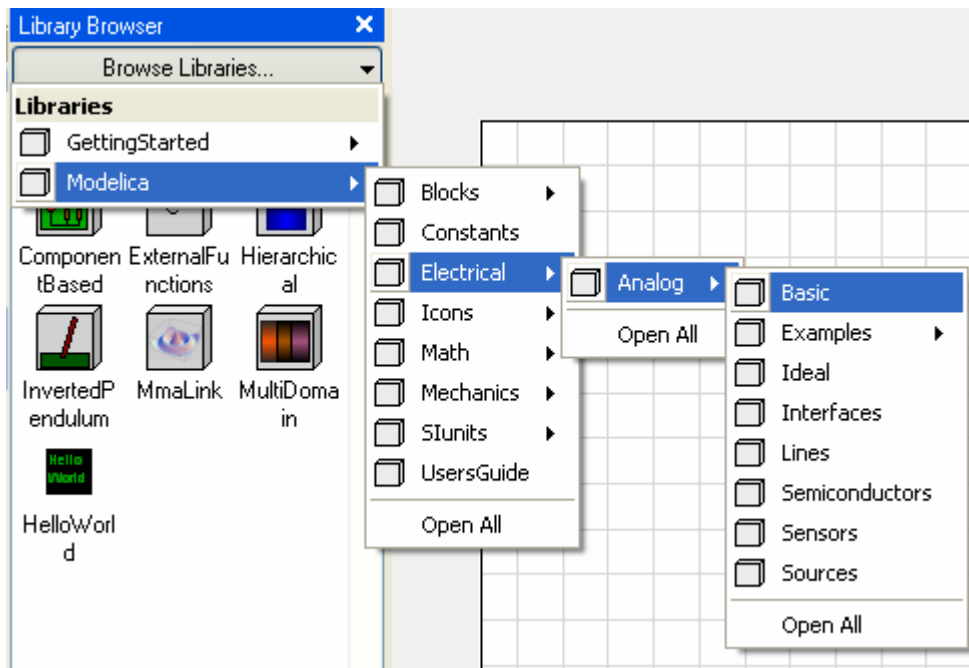
The step source can be found in the Modelica.Blocks.Sources library:



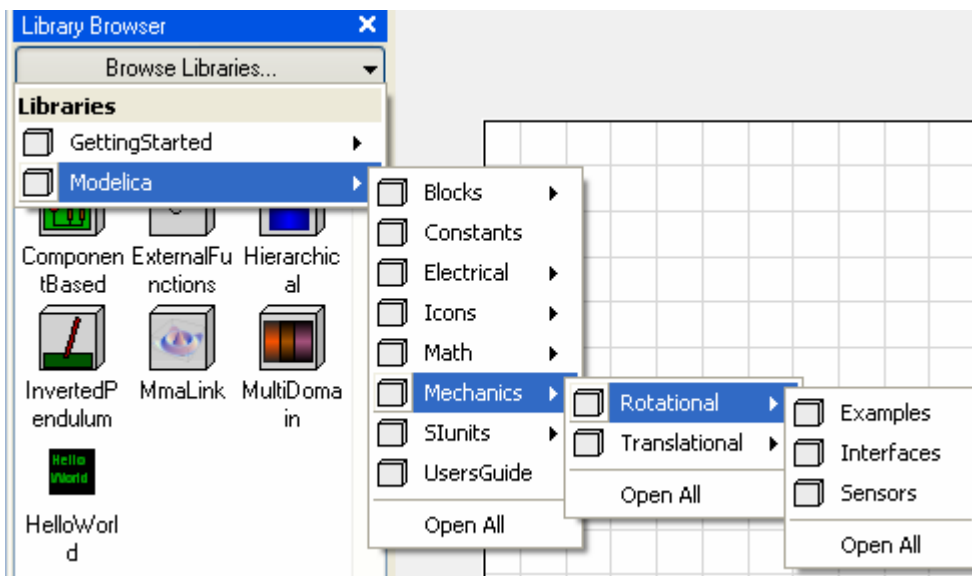
Likewise the signal voltage can be found in Modelica.Electrical.Analog.Sources:





The electrical components are found in Modelica.Electrical.Analog.Basic:



Finally the mechanical components are found in Modelica.Mechanics.Rotational:



The components are then connected using the connection tool: 

When creating this model a corresponding Modelica code is generated, the code is seen by clicking on the Modelica Text View icon: . In the model code each component is first declared, and the connection between components is made in connect statements in the equation section.

```
model DCMotor
```

```

Modelica.Blocks.Sources.Step step;
Modelica.Electrical.Analog.Sources.SignalVoltage signalVoltage1;
Modelica.Electrical.Analog.Basic.Resistor resistor1;
Modelica.Electrical.Analog.Basic.Inductor inductor1;
Modelica.Electrical.Analog.Basic.EMF EMF1;
Modelica.Mechanics.Rotational.Inertia inertial;
Modelica.Electrical.Analog.Basic.Ground ground1;

```


```
equation
```

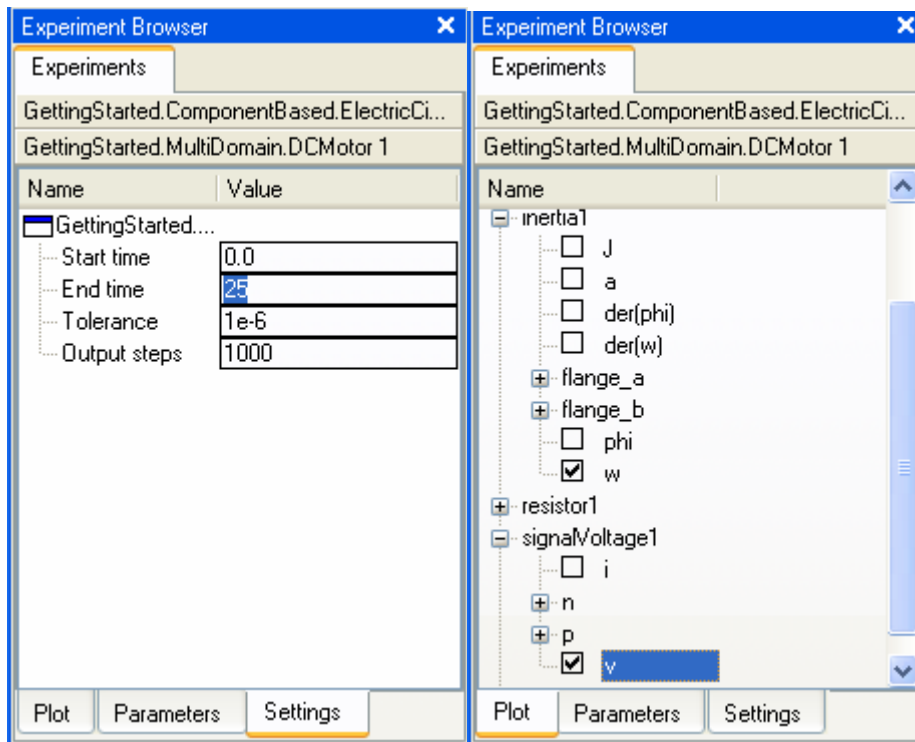
```

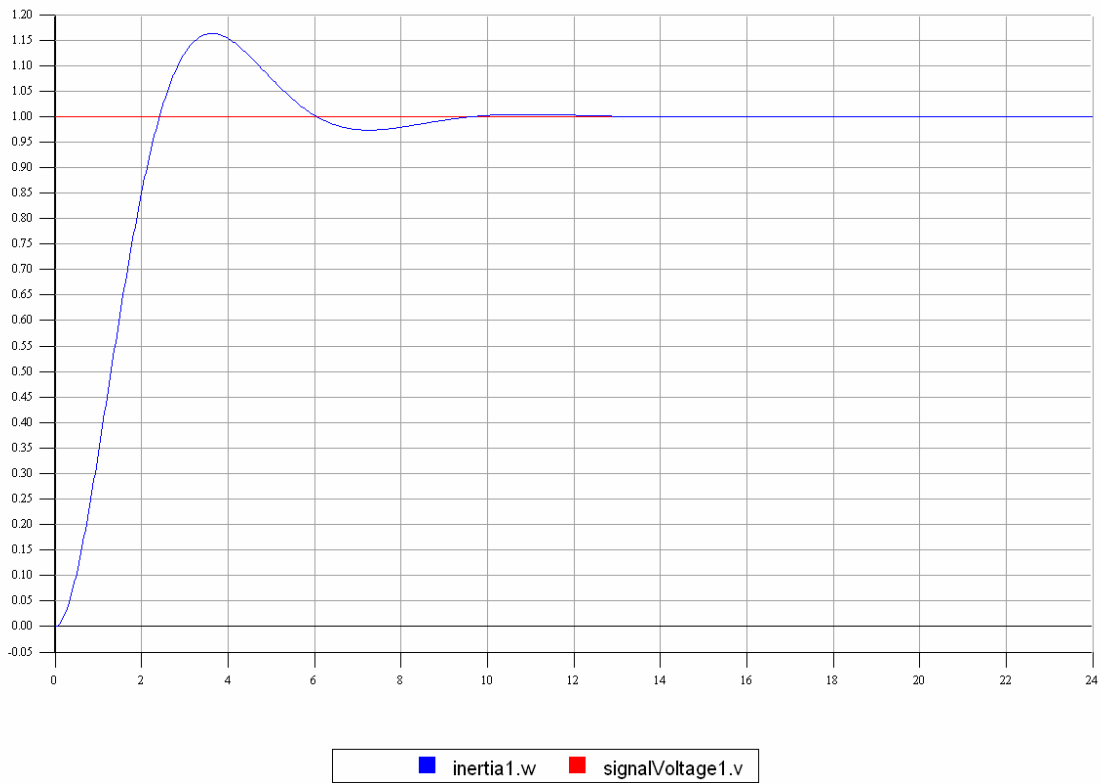
connect(EMF1.flange_b,inertial.flange_a);
connect(EMF1.n,signalVoltage1.n);
connect(signalVoltage1.n,ground1.p);
connect(inductor1.n,EMF1.p);
connect(resistor1.n,inductor1.p);
connect(signalVoltage1.p,resistor1.p);
connect(step1.y,signalVoltage1.v);

```

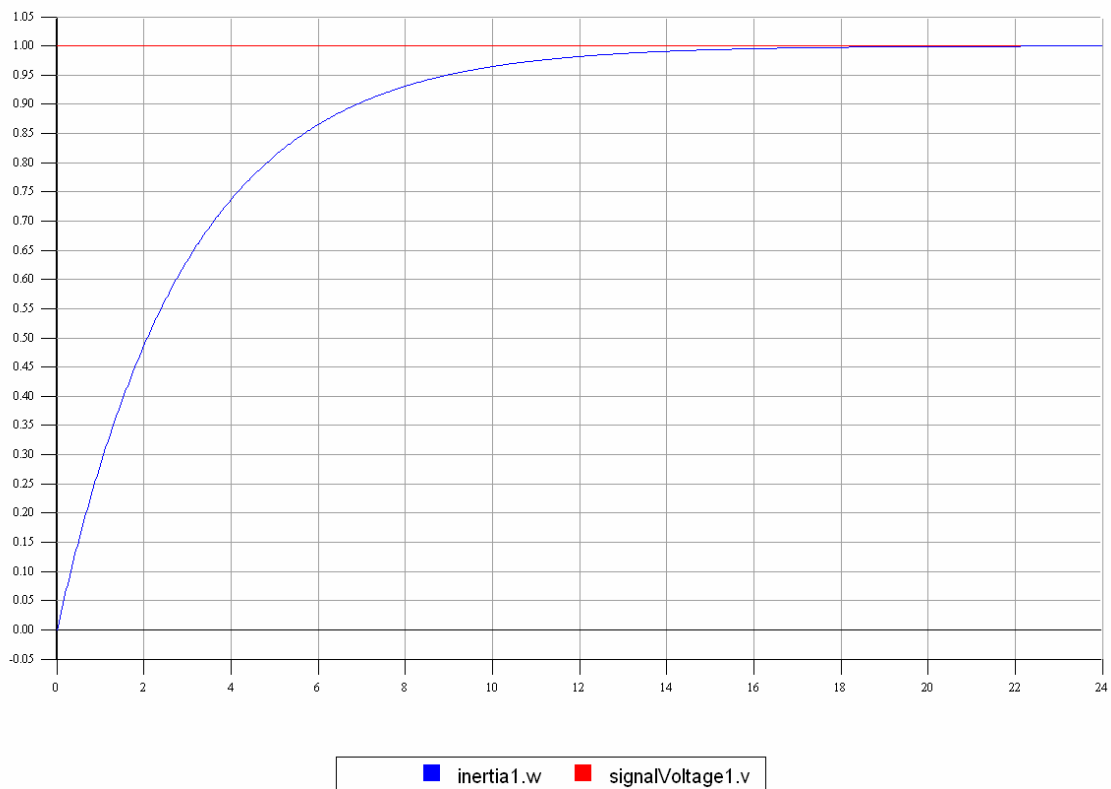
```
end DCMotor;
```

After creating the model it is possible to simulate it by starting the Simulation Center (Click ). In this case we begin by setting the simulation time to 25 second and then we study the step response for the rotational speed of the inertia.



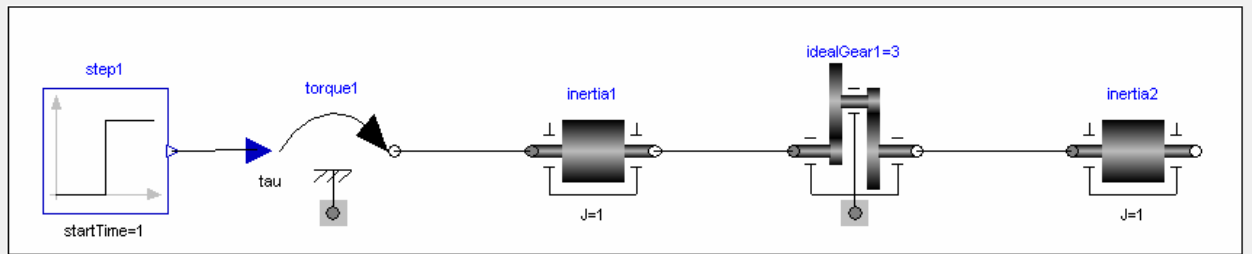


It is also easy to change parameter values to modify system behavior. For instances setting resistance to 10 Ohm, Inductance to 0.1 H, and moment of inertia to 0.3 kgm^2 yields a damped instead of oscillative step response.

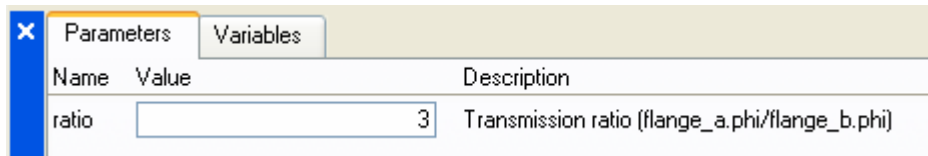


3 Gear and Axis

Next step is to develop a gear and axis model and study its step response by applying a step torque as illustrated below. The needed components can be found in Modelica.Blocks.Sources and Modelica.Mechanics.Rotational.

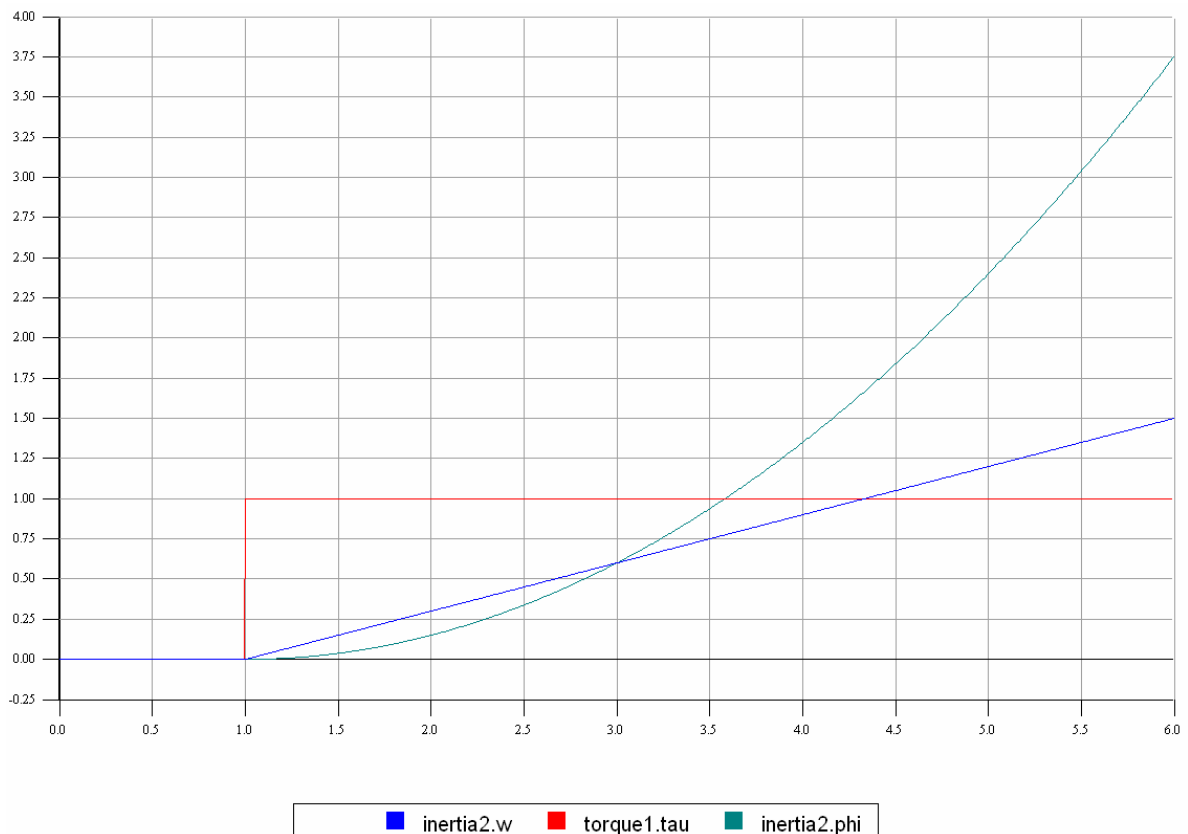


By clicking on the ideal gear we can edit the gear ratio in the parameters window below the diagram area:

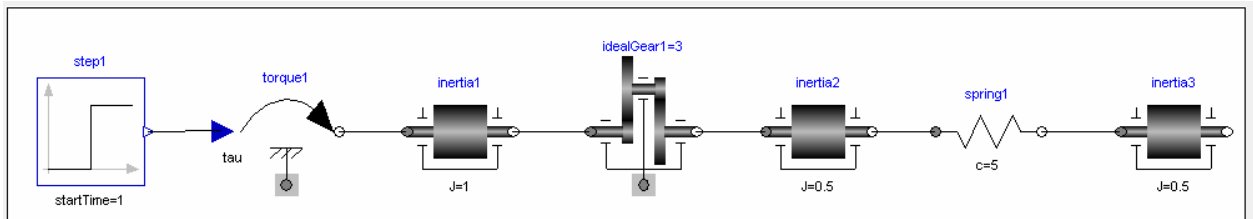


In this case a gear ratio of 1 to 3 ratio I chosen. This means that angles and angular velocity are amplified three times and torque is attenuated by a factor three from one side of the gear to the other.

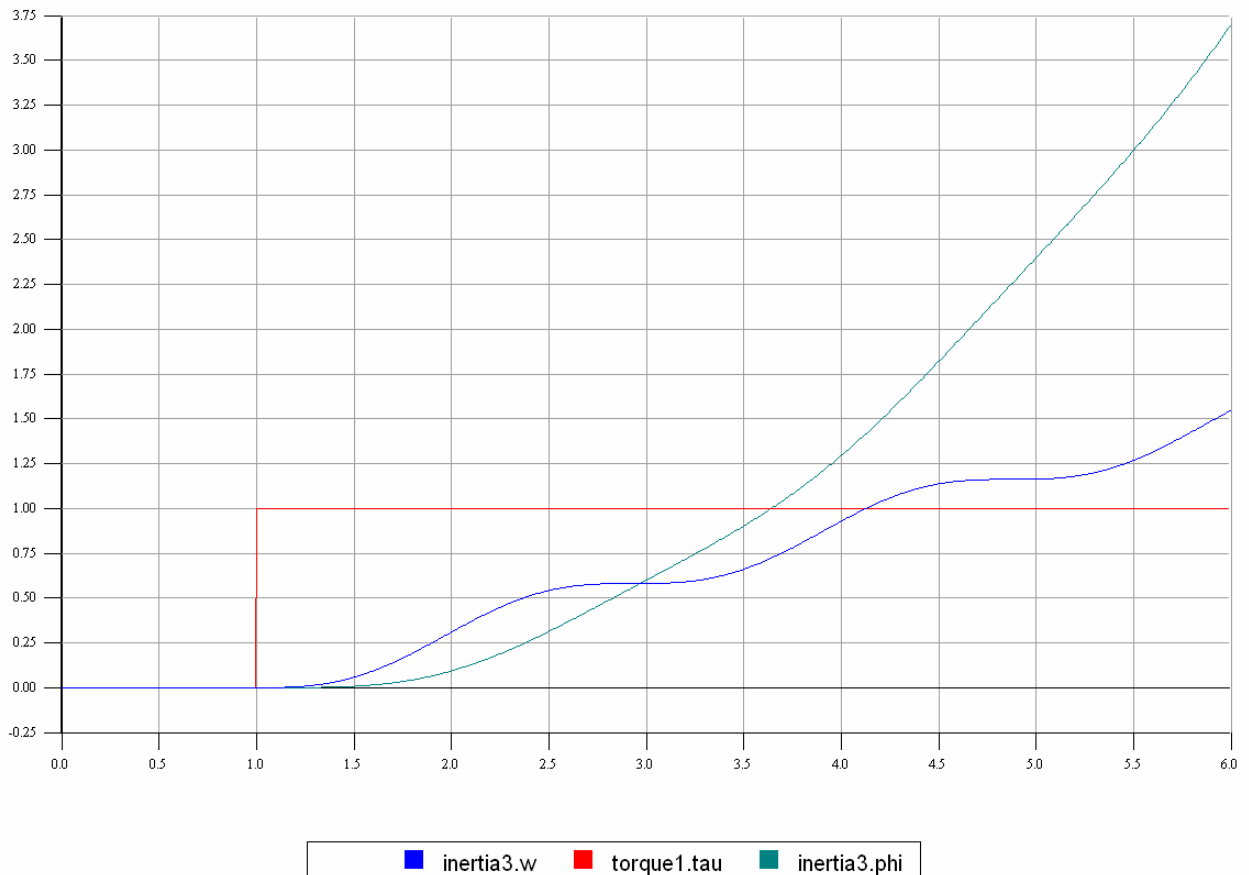
After simulate the system for 6 seconds we can observe that a constant torque results in a constant angular acceleration that is a ramp in angular velocity and a square curve for the angle of the axis.



By including an additional weakness the axis can be more accurately modeled. This can be done by substituting the above axis model with a model consisting of two rotating masses connected by a torsion spring according to the figure below. The torsion spring is also found in the Modelica.Mechanics.Rotational library.



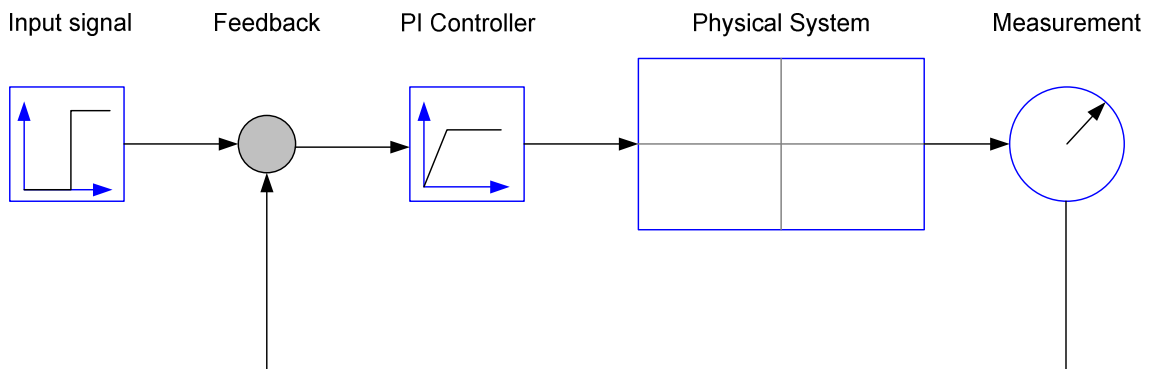
We simulate this subsystem for 6 seconds and study the result. A comparison with the stiff axis model shows that we have a similar behavior but with a deflection added. Note that inertia3 and not inertia2 as earlier is the last element of the axis, therefore we plot the rotational velocity and angle for inertia3 in order to do a fair comparison.



4 Control System

The structure of the control system is shown in the schematic picture below. It consists of an input signal, a sensor, a feedback loop, and a regulator. The physical system consists of the gear and axis

system described above. Since the physical system has negative static gain the PI gain also has to be negative.

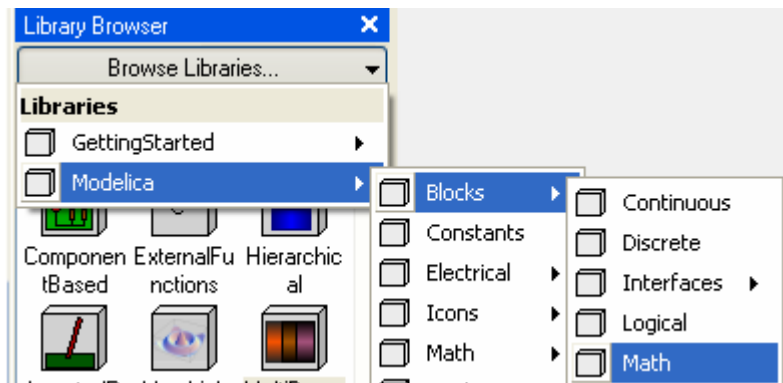


We now connect all three subsystems together according to this figure. The default choice of regulator parameters are $k=1$ and $T=1$, where the PI regulator transfer function is $G_{PI} = k T s + 1/(T s)$. For the new components we need to open the following libraries.

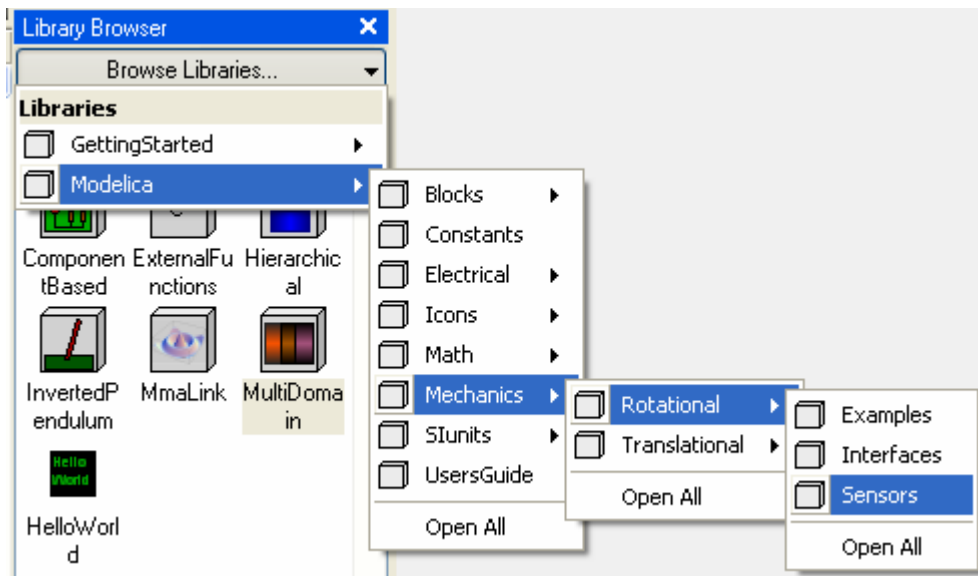
The PI controller is found in Modelica.Blocks.Continuous:



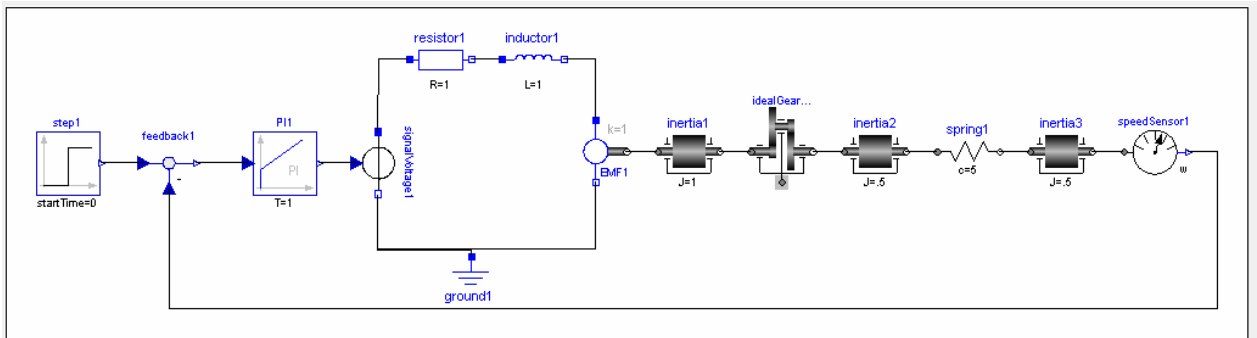
The feedback is found in Modelica.Blocks.Math:



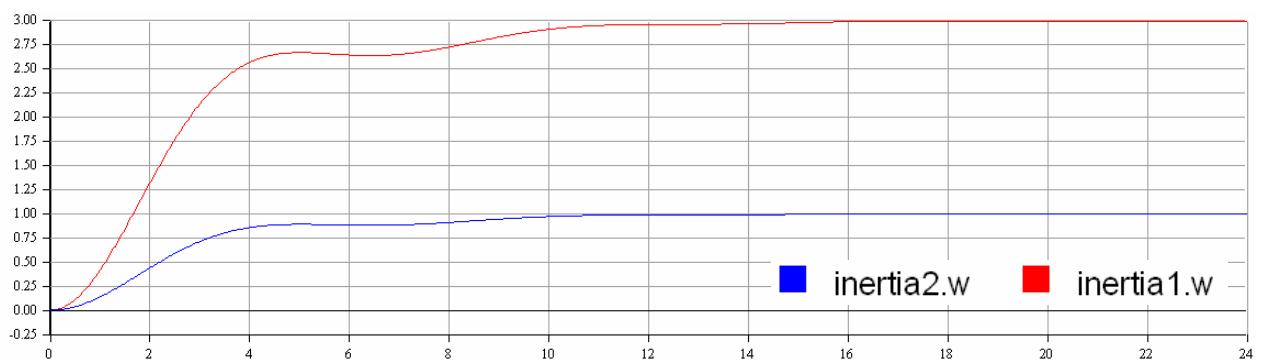
Finally the velocity sensor is found in Modelica.Mechanics.Rotational.Sensors:



Using the components from these libraries we can connect the complete model:

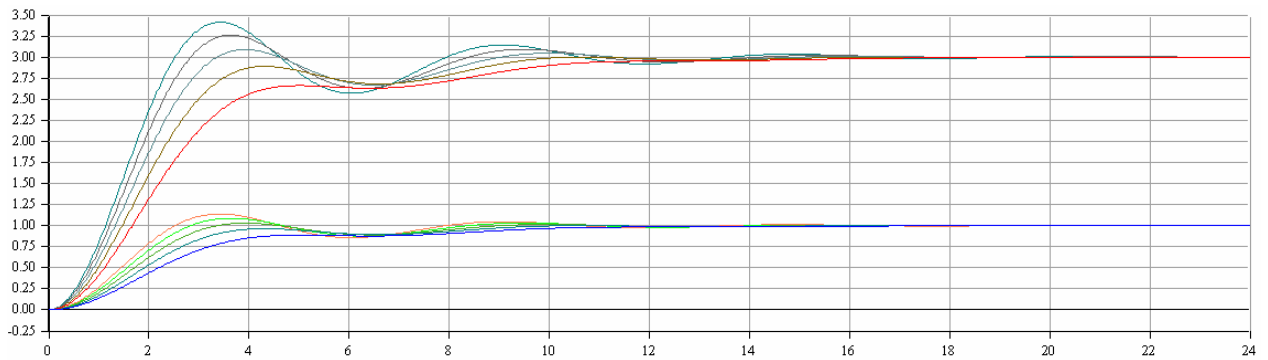


Note that the model above uses the weak axis, but will actually start by simulating using the stiff axis with a step response in the reference angular velocity signal. A plot of the response for the angular velocity of both the motor axis and the gear axis is shown in the figure below:



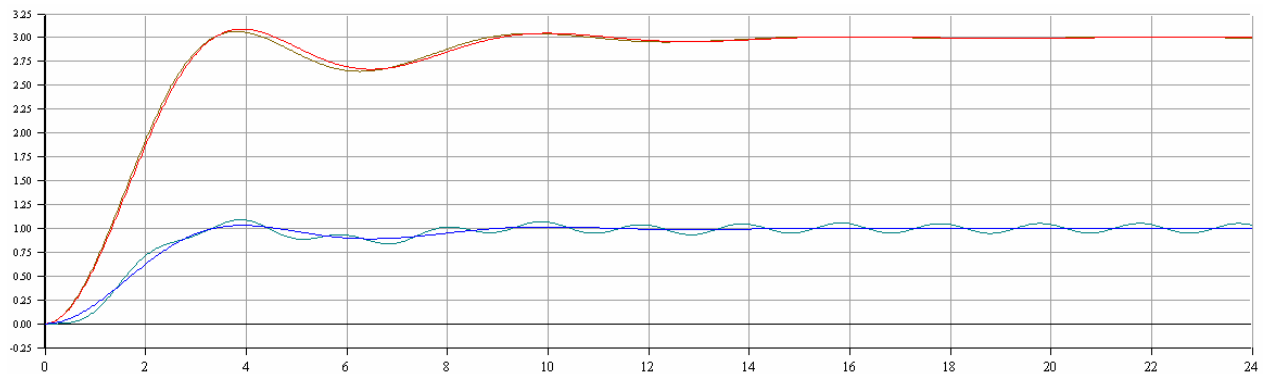
To this moment we have used default parameters for the controller, by varying the controller gain k we can the response. In this case we vary the gain from 1 to 2 in an interval of 0.25. This is done by

creating new experiments clicking on [New Experiment](#) in the Simulation Center, and then setting the appropriate parameter for each experiment. The results from the experiments can then be plotted in the same window.



By studying the angular velocity response for the motor and gear axis using different regulator gains we conclude that by choosing $k = 1.5$ we get a sufficiently fast response with few oscillations.

Finally we use these regulator settings for the model with the weak axis and compare this with the result for the stiff axis system.



As seen the controller design made using the stiff axis model performs well also when the more accurate weak axis is used in simulations.