

# Exercises Using OpenModelica

Version 2015-10-02

## Open Source Modelica Consortium

PELAB – Programming Environment Laboratory  
SE-581 83 Linköping, Sweden

### 1 Short Introduction to the Graphical Modeling Editor

Install OpenModelica. Start the OpenModelica Connection Editor OMEdit. Do the RLCircuit graphical exercise. See instructions in slides.

### 2 Simple Textual Modelica Modeling Exercises

**Start** OMNotebook: Start->Programs->OpenModelica->OMNotebook.

**Look** in the directory to where you copied the course material.

**Open** in OMNotebook the file: **Exercises-ModelicaTutorial.onb**

There will be additional instructions in the notebook.

#### 2.1 HelloWorld

Simulate and plot the following example with one differential equation and one initial condition. Do a slight change in the model, re-simulate and re-plot.

```
model HelloWorld "A simple equation"  
  Real x(start=1);  
equation  
  der(x) = -x;  
end HelloWorld;
```

#### 2.2 Try DrModelica with VanDerPol

Locate the VanDerPol model in DrModelica (link from Section 2.1), run it, change it slightly, and re-run it.

#### 2.3 DAEExample

Locate the DAEExample in DrModelica (Section 2.1: Differential Algebraic Equation System). Simulate and plot.

#### 2.4 A Simple Equation System

Develop a Modelica model that solves the following equation system with initial conditions:

$$\begin{aligned} \dot{x} &= 2 + x + y - 3 + x \\ \dot{y} &= 5 + y - 7 + x + y \\ x(0) &= 2 \\ y(0) &= 3 \end{aligned}$$

#### 2.5 Functions and Algorithm Sections (if you have time)

a) Write a function, `sum`, which calculates the sum of Real numbers, for a vector of arbitrary size.

b) Write a function, `average`, which calculates the average of Real numbers, in a vector of arbitrary size. The function `average` should make use of a function call to `sum`.

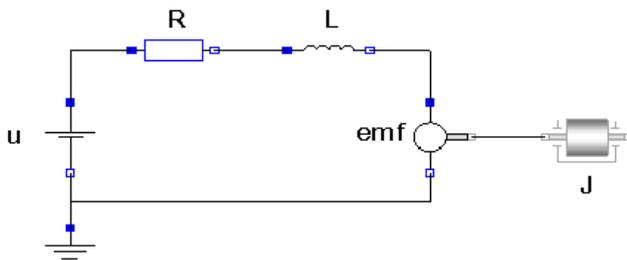
## 2.6 Hybrid Models - BouncingBall

Locate the `BouncingBall` model in one of the hybrid modeling sections of `DrModelica` (e.g. Section 2.9), run it, change it slightly, and re-run it.

# 3 Graphical Design using the Graphical Connection Editor

## 3.1 Simple DC-Motor

Make a simple DC-motor using the Modelica standard library that has the following structure:



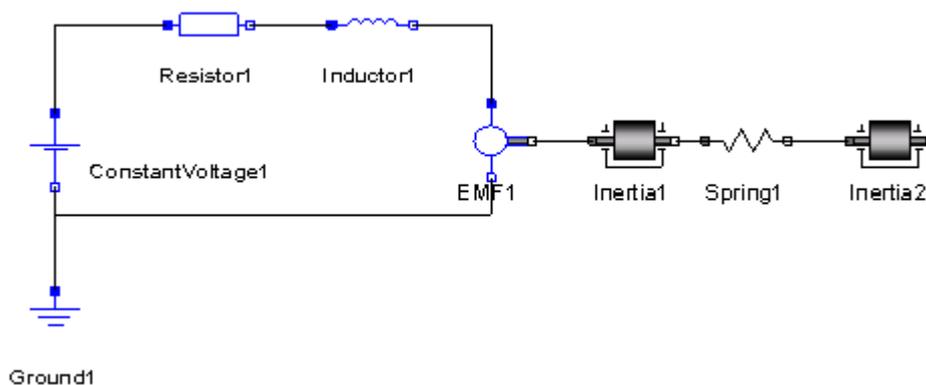
You can simulate and plot the model directly from the graphical editor. Simulate for 15s and plot the variables for the outgoing rotational speed on the inertia axis and the voltage on the voltage source (denoted `u` in the figure) in the same plot.

Option: You can also save the model, load it and simulate it using `OMShell` or `OMNotebook`. You can also go to the graphical editor text view and copy/paste the model into a cell in `OMNotebook`.

Hint: if you use the plot command in `OMNotebook` and you have difficulty finding the names of the variables to plot, you can flatten the model by calling `instantiateModel`, which exposes all variable names.

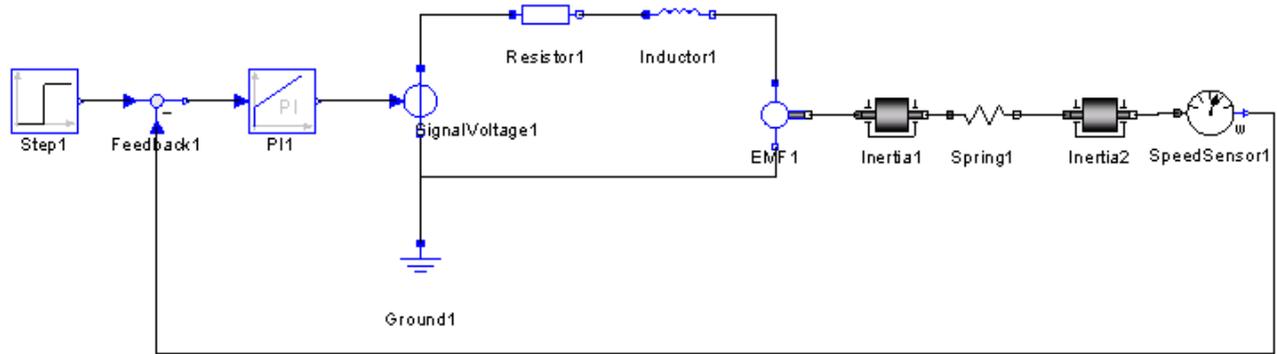
## 3.2 DC-Motor with Spring and Inertia

Add a torsional spring to the outgoing shaft and another inertia element. Simulate again and see the results. Adjust some parameters (right-click on corresponding model component icon) to make a rather stiff spring.



## 3.3 DC-Motor with Controller (Extra)

Add a PI controller to the system and try to control the rotational speed of the outgoing shaft. Verify the result using a step signal for input. Tune the PI controller by changing its parameters. Right-click on the PI Controller Icon to change parameters.



### 3.4 Exercises in Control Theory and Modelica with DrControl (Extra)

Look in the handout slides for further instructions and in `c:\OpenModelica1.9.3\share\omnotebook\drcontrol\`