Modeling Power Electronics in Dymola 5

Nick Benavides
University of Illinois
Urbana-Champaign

Outline

- Dymola/Modelica overview
- Interface and basic operation
- Text based model example
- Component creation
- Half wave rectifier example
- Controlled 2-output flyback converter

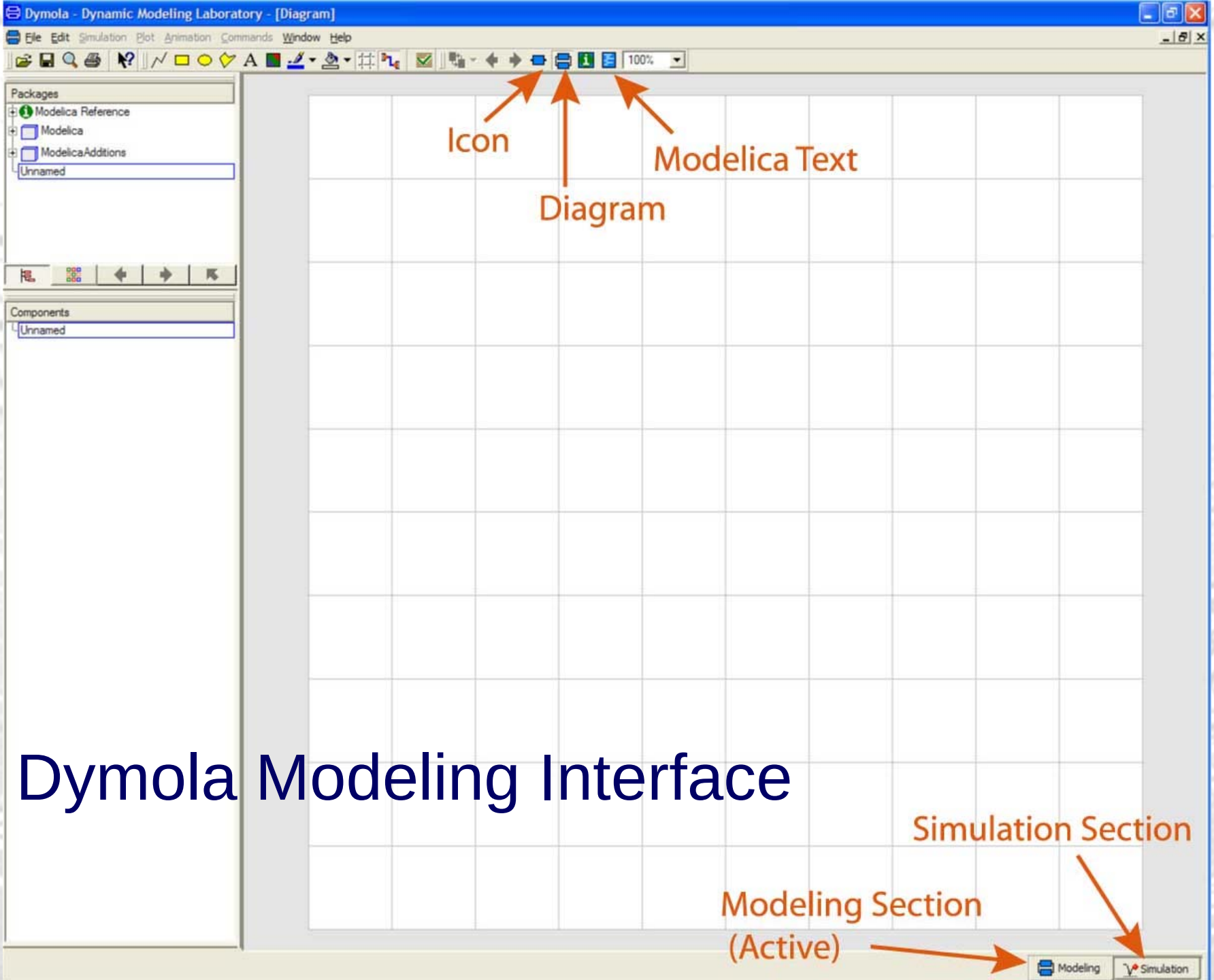
ECE ILLINOIS

Modelica/Dymola

- Modelica is the underlying modeling language used in the Dymola Software package
- The Modelica language allows differential equation models to be built without state form, from which state equations are derived automatically
- Dymola allows Modelica models to carry a graphical form, and also provides integration algorithms for use with the models
- http://www.modelica.org/
- http://www.dynasim.de/

ECE ILLINOIS

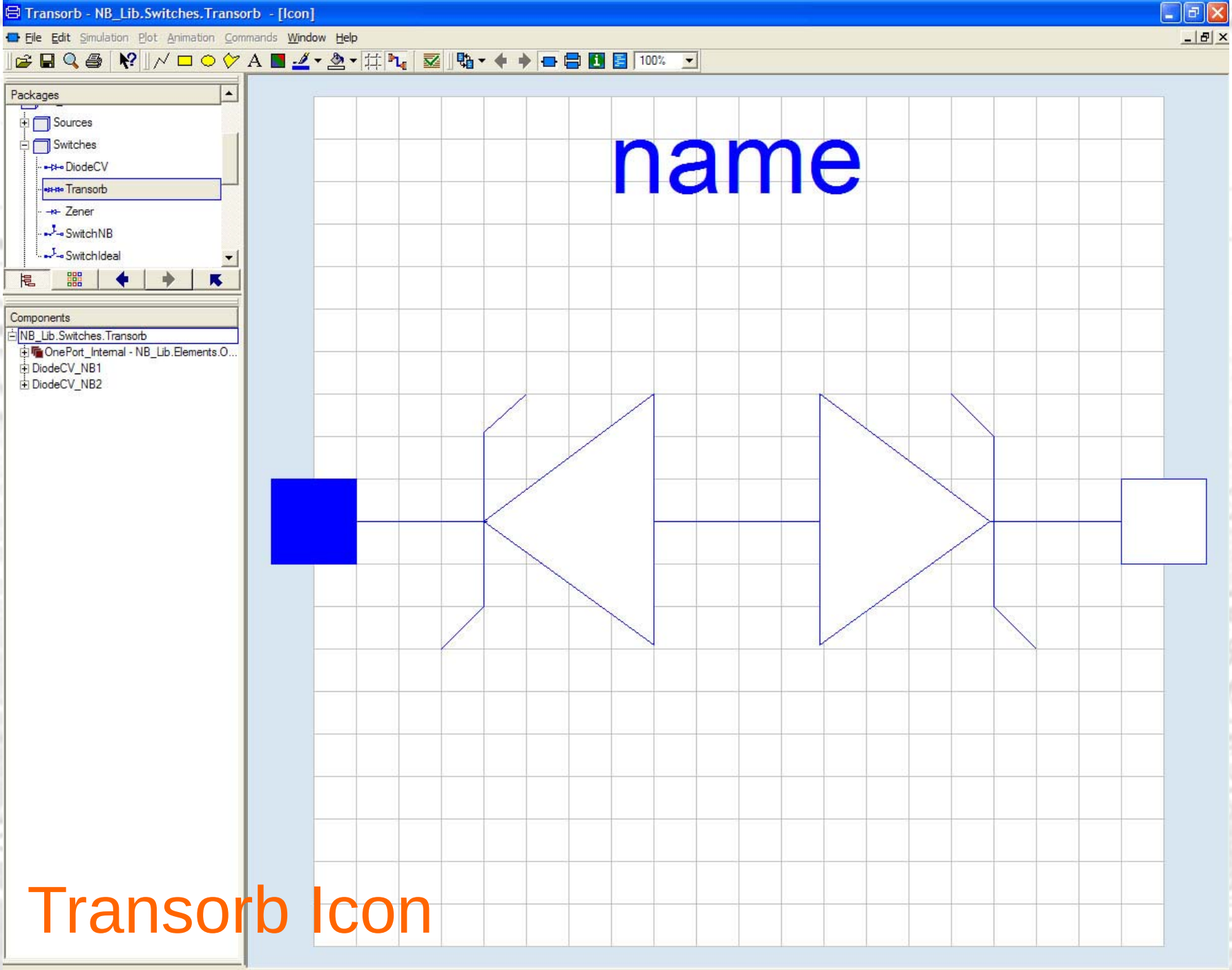
Dymola Modeling Interface



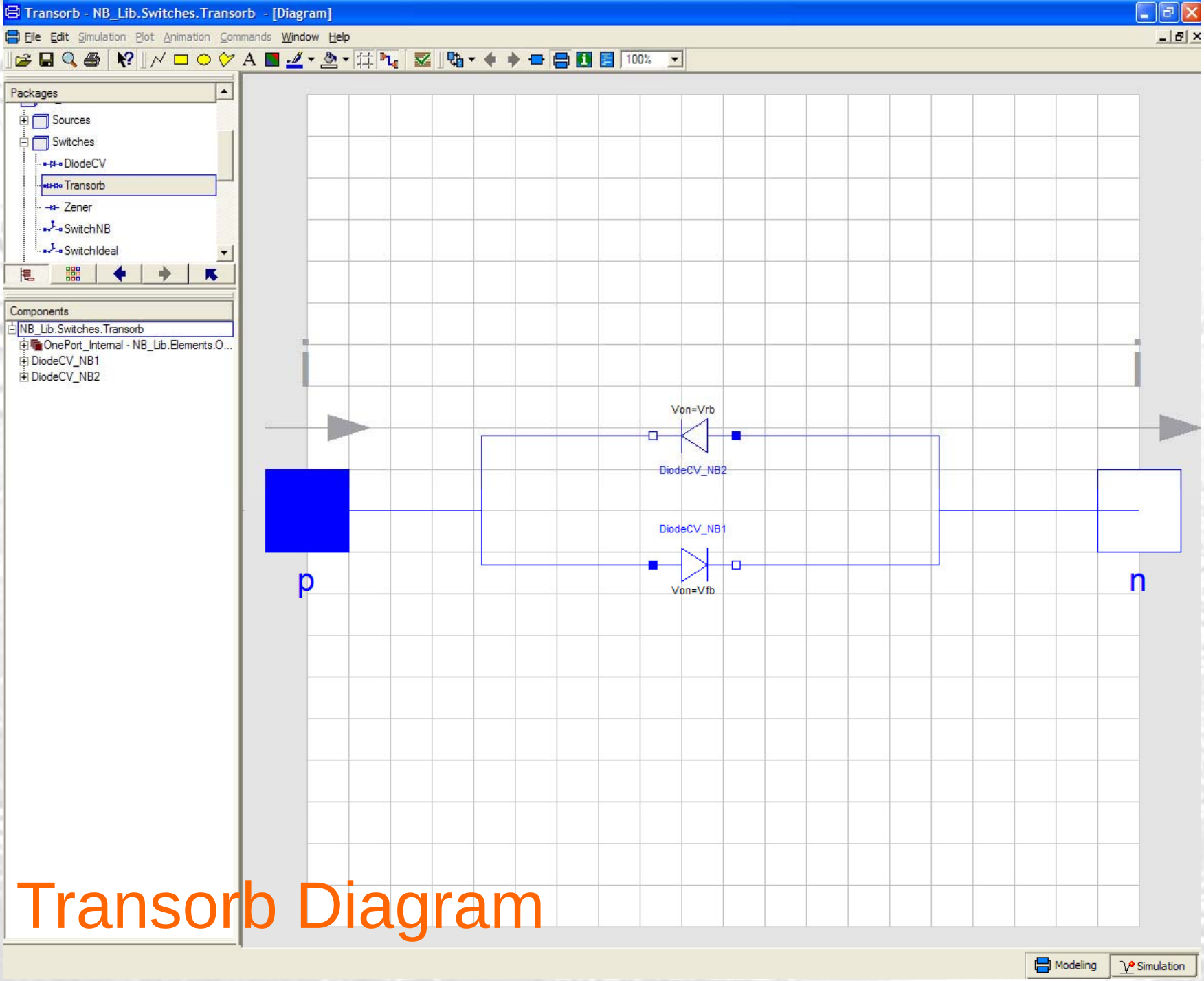
Modeling Window

- Library Tree
  - Selects active Model
  - Components can be dragged into active Model's Diagram
- Icon
  - Graphic used when placing in other models
- Diagram
  - Graphical Internal Model
- Modelica Text
  - Contains all information about Icon, Diagram,
  - and Documentation when text is expanded

ECE ILLINOIS



Transorb Icon



Transorb Diagram

Transorb - NB_Lib.Switches.Transorb - [Modelica Text]

File Edit Simulation Plot Animation Commands Window Help

Packages

- Sources
- Switches
  - DiodeCV
  - Transorb
  - Zener
  - SwitchNB
  - SwitchIdeal

Components

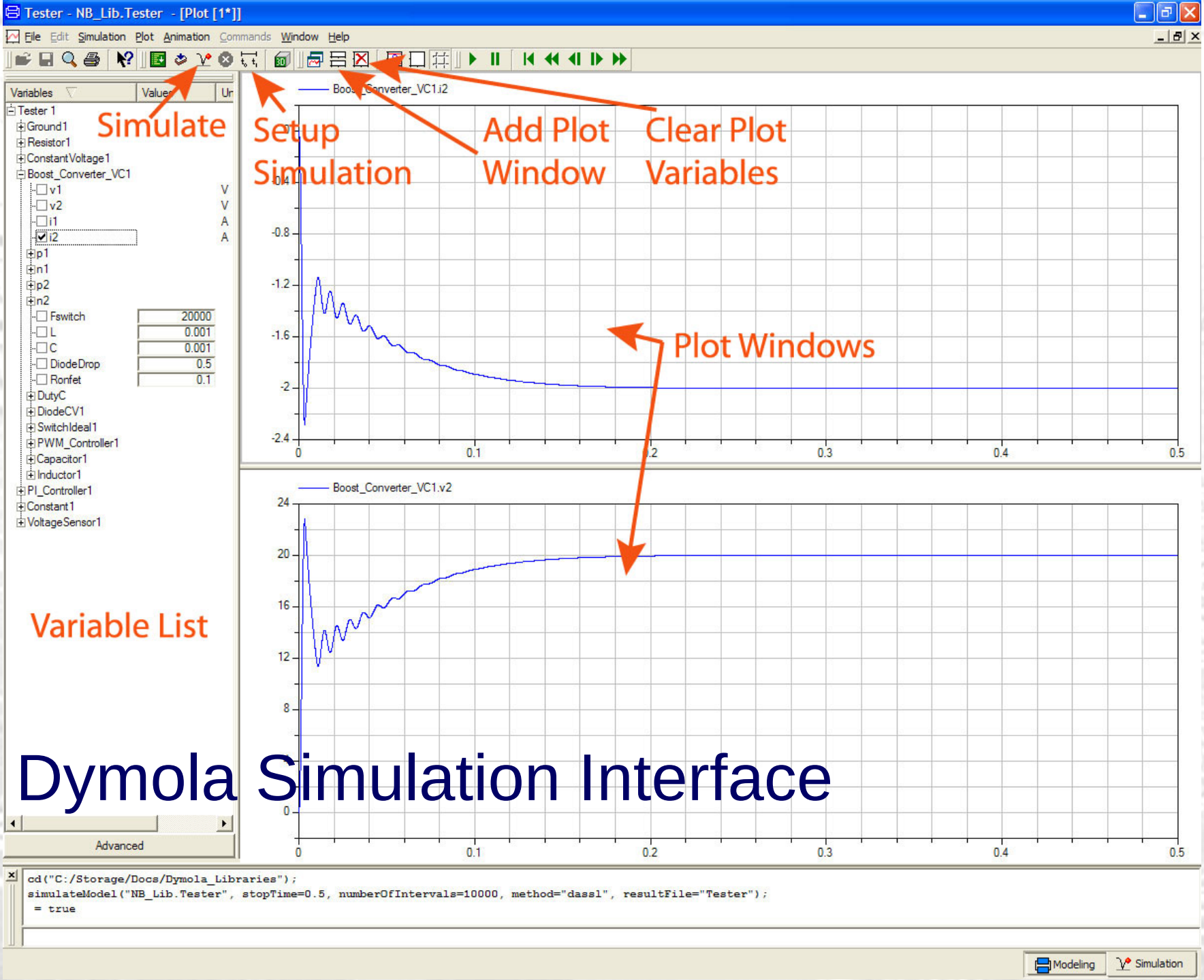
- NB_Lib.Switches.Transorb
  - OnePort_Internal - NB_Lib.Elements.O...
  - DiodeCV_NB1
  - DiodeCV_NB2

```
model Transorb
  extends Elements.OnePort_Internal;
  parameter Real Vrb=50;
  parameter Real Vfb=50;
  annotation (
    Coordsys(
      extent=[-100, -100; 100, 100],
      grid=[1, 1],
      component=[20, 20]),
    Icon(
      Polygon(points=[59, 0; 19, 30; 19, -29; 59, 0], style(
          color=3,
          gradient=0,
          fillColor=7)),
      Text(extent=[-100, 100; 100, 70], string="%name"),
      Polygon(points=[-60, 0; -20, 30; -20, -29; -60, 0], style(
          color=3,
          gradient=0,
          fillColor=7)),
      Line(points=[-95, 0; -94, 0; -60, 0; -59, 0]),
      Line(points=[-20, 0; 19, 0]),
      Line(points=[59, 0; 90, 0]),
      Line(points=[50, 30; 60, 20; 60, -20; 70, -30]),
      Line(points=[-70, -30; -60, -20; -60, 21; -50, 30])),
    Diagram,
    Window(
      x=0.26,
      y=0.11,
      width=0.6,
      height=0.6));

  Switches.DiodeCV DiodeCV_NB1(Von=Vfb) annotation (extent=[-17, -23; 3, -3]);
  Switches.DiodeCV DiodeCV_NB2(
    Von=Vrb,
    Goff=1e-5,
    Ron=1e-5) annotation (extent=[-17, 8; 3, 28], rotation=180);
equation
  connect(DiodeCV_NB2.n, p)
    annotation (points=[-17, 18; -58, 18; -58, 0; -100, 0], style(color=3));
  connect(DiodeCV_NB1.p, p)
    annotation (points=[-17, -13; -58, -13; -58, 0; -100, 0], style(color=3));
  connect(DiodeCV_NB2.p, n)
    annotation (points=[3, 18; 52, 18; 52, 0; 100, 0], style(color=3));
  connect(DiodeCV_NB1.n, n)
    annotation (points=[3, -13; 52, -13; 52, 0; 100, 0], style(color=3));
end Transorb;
```

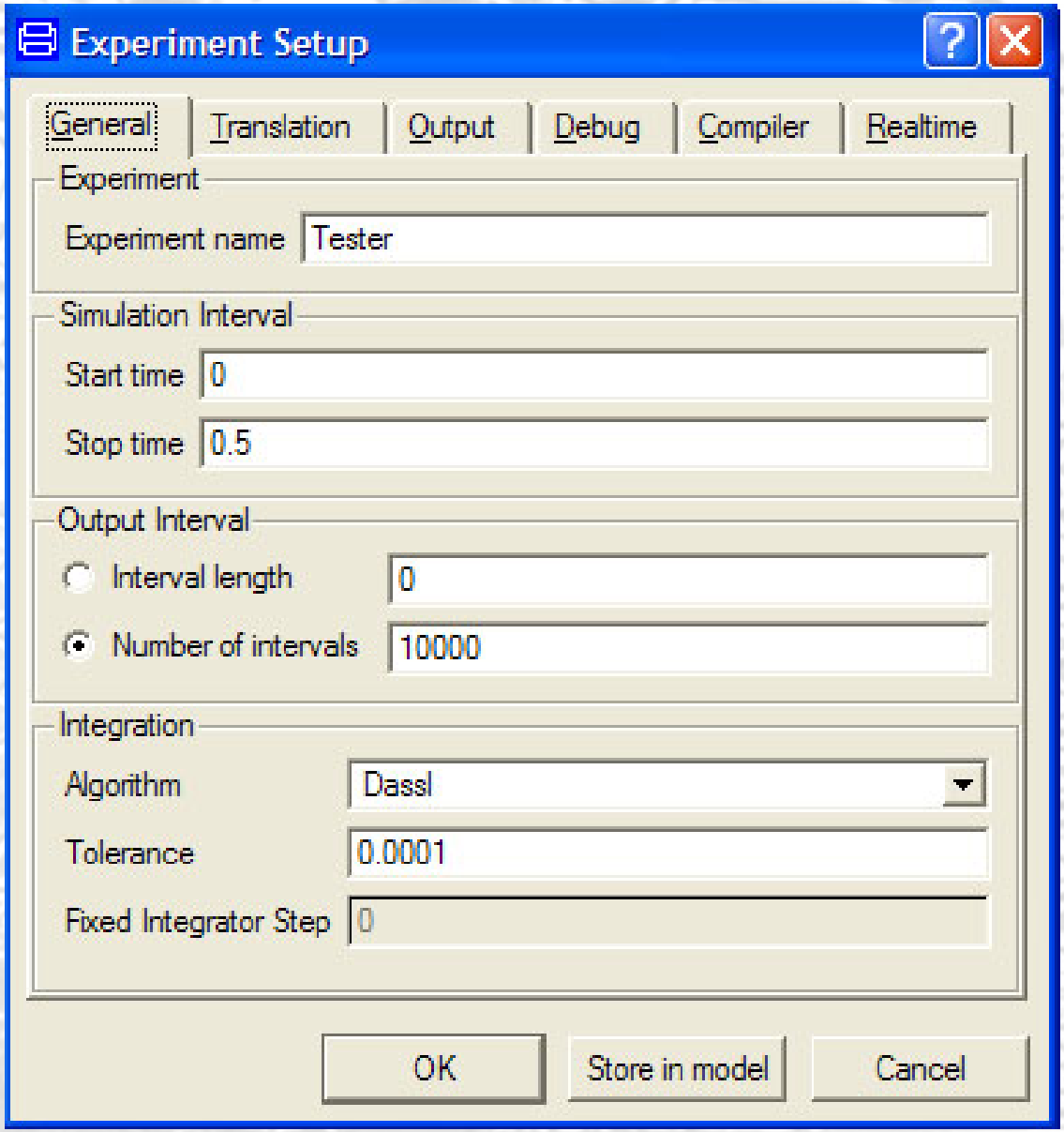
Line: 1 Modeling Simulation

Transorb Text

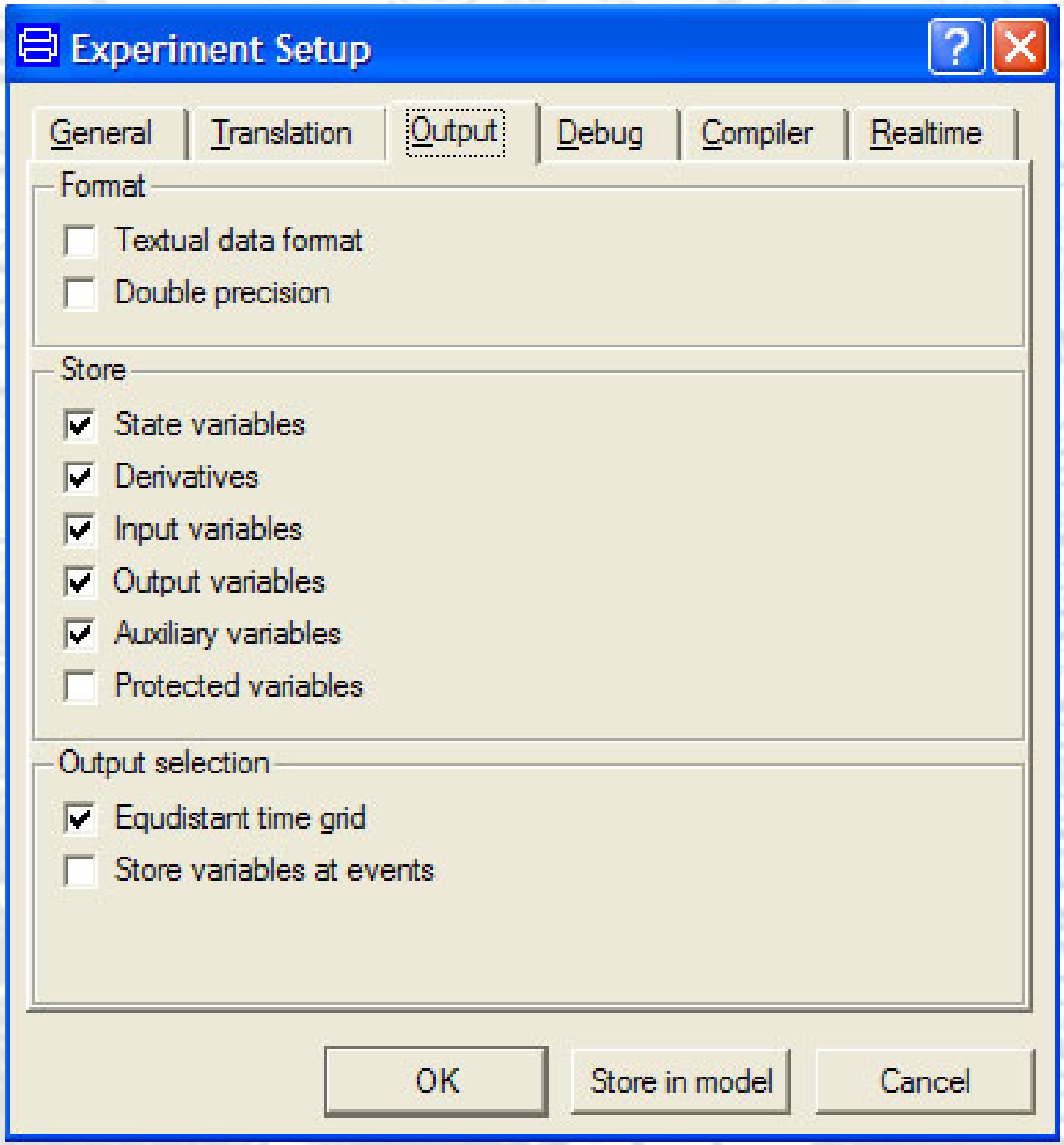


Dymola Simulation Interface

Experiment Setup (General)



Experiment Setup (Output)



ECE ILLINOIS

Modelica.Electrical.Analog.Interface.Pin

- Annotation (Graphical Attributes) Removed

```
connector Pin
  Real v;
  flow Real i ;
end Pin;
```

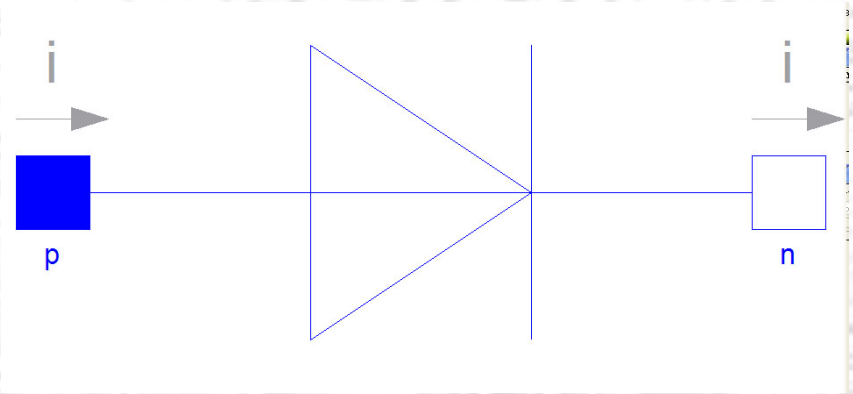
M.Electrical.Analog.Interface.OnePort

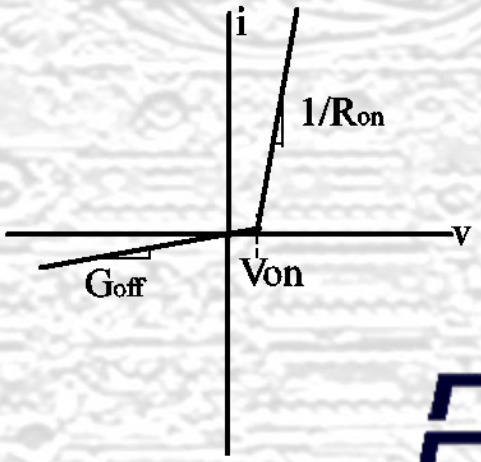
- Annotation (Graphical Attributes) Removed

```
partial model OnePort
Real v ;
Real I;
Pin p;
Pin n;
equation
  v = p.v - n.v;
  0 = p.i + n.i;
  i = p.i;
end OnePort;
```

NbLib.Switches.DiodeCV

```
model DiodeCV
  extends
Modelica.Electrical.Analog.
    Interfaces.OnePort;
  parameter Real Goff=1e-5;
  parameter Real Ron=1e-5;
  parameter Real Von=0.5;
  parameter Real Ion=Von*Goff ;
  Boolean off(start=true);
protected
  Real s "Auxiliary variable";
equation
  off = s < 0;
  v - Von = s*(if off then 1
else Ron);
  i - Ion = s*(if off then Goff
else 1);
end DiodeCV;
```





Starting a Package

- Return to Modeling interface
- Select File->New ->Package
- Enter name “DymolaTutorial” and leave all other entries blank.
- Check Save Package as single file and click OK
- Now double click the new package in the Library Tree and click save to save the new file

Starting a Model

- Right-click on package DymolaTutorial and select Edit->New Class in Package->new Model.
- Enter Name “BouncingBall” and be sure that the model is inserted in package “DymolaTutorial”
- To begin editing the BouncingBall, double click on the model in the Library Tree
- Enter Text Modeling mode

Bouncing Ball Simulation (1)

```
model BouncingBall
  Real x(unit="m");
  Real y(unit="m");
  Real dx(unit="m/s", start=3);
  Real dy(unit="m/s", start=10);
  parameter Real g(unit="m/s^2") =
9.8;
  parameter Real c=0.9 "Coefficienct
of elasticity";
```

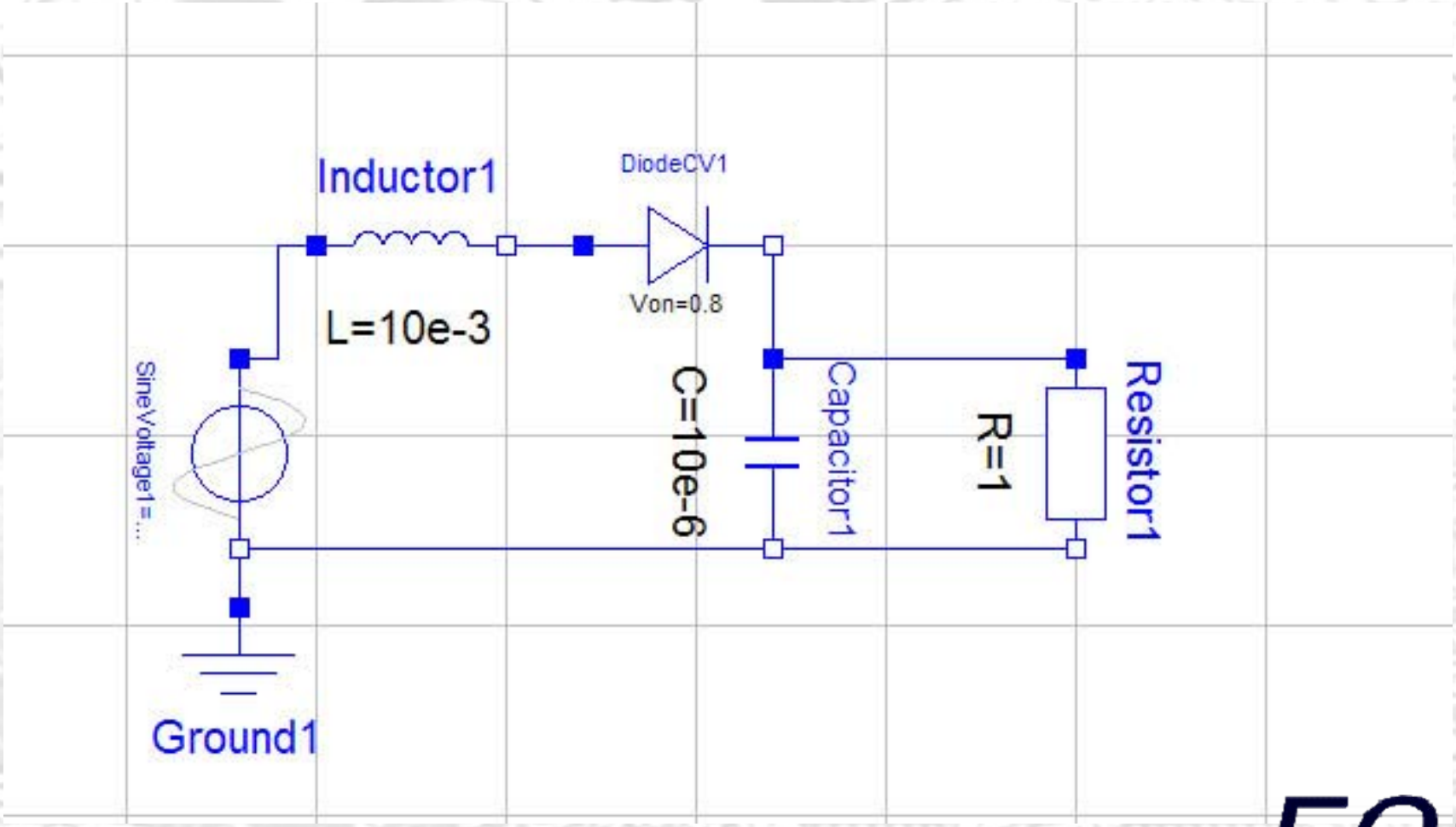
Bouncing Ball Simulation (2)

```
equation
  der(x) = dx;
  der(dx) = 0;
  der(y) = dy;
  der(dy) = -g;
  when (y) < 0 then
     reinit(dy, -c*dy);
  end when;
 end BouncingBall;
```

Bouncing Ball (3)

- Integration Options:
  - Start Time: 0
  - Stop Time: 20
  - Intervals: 5000
- Click store in model.
- Now try Plotting some variables such as y by clicking on them in the Variable list
- Also try right clicking on x and selecting independent variable as x, then click on y to plot y(x)

Half Wave Rectifier Circuit (1)



Half Wave Rectifier Circuit (2)

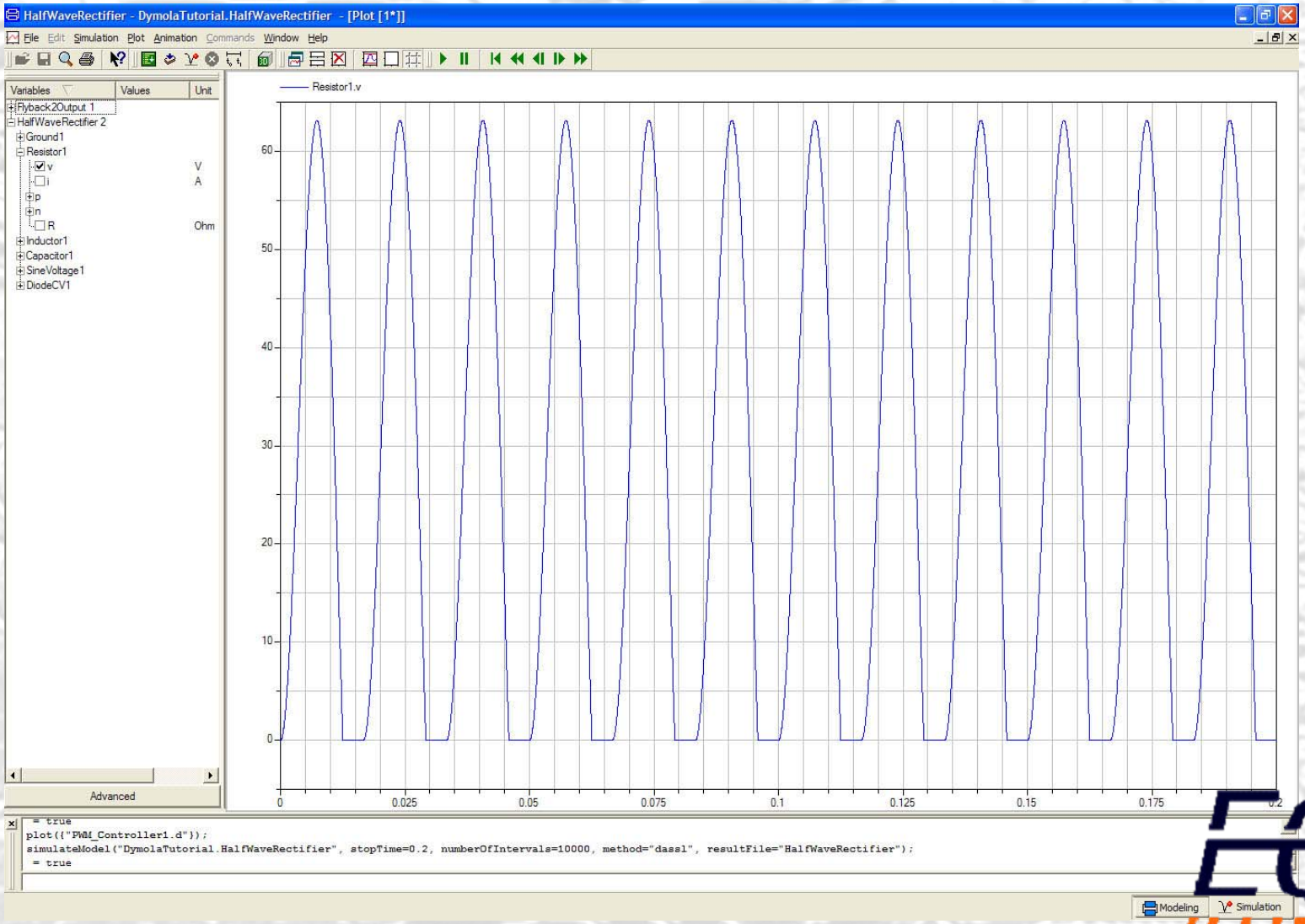
- Components Needed:
  - Modelica.Electrical.Analog.Basic.Ground
  - Modelica.Electrical.Analog.Sources.SineVoltage
    - V=170 Freq=60Hz
  - Modelica.Electrical.Analog.Basic.Inductor
    - L=10e-3 H
  - Modelica.Electrical.Analog.Basic.Capacitor
    - C=10e-6 F
  - Modelica.Electrical.Analog.Basic.Resistor
    - R=1 ohm
  - NB_Lib.Switches.Diode_CV
    - Ron=1e-5, Goff=1e-5, Von=0.8

Half Wave Rectifier Circuit (3)

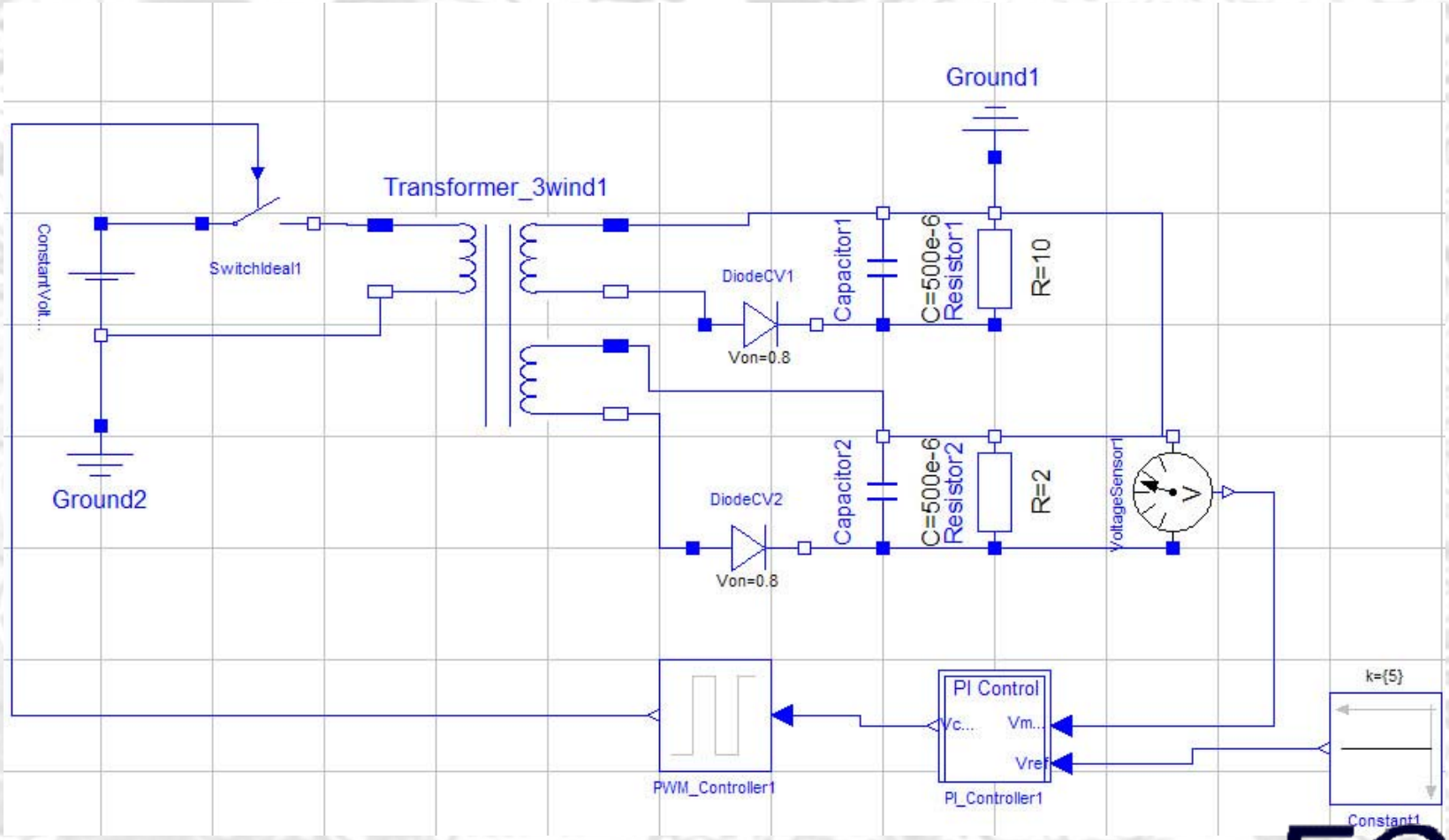
- Integration Options:
  - Start Time: 0
  - Stop Time: 0.2
  - Intervals: 5000
  - Algorithm: Dassl
- Click store in model.

ECE ILLINOIS

Half Wave Rectifier Circuit (4)



Controlled 2 Output FB (1)



Controlled 2 Output FB (2)

- Components Needed:
  - Modelica.Electrical.Analog.Basic.Ground
  - Modelica.Electrical.Analog.Sources.ConstantVoltage
    - V=48
  - Modelica.Electrical.Analog.Basic.Capacitor
    - C1=500e-6 F, C2=500e-6 F
  - Modelica.Electrical.Analog.Basic.Resistor
    - R=10 ohm, R2=2 ohm
  - Modelica.Electrical.Analog.Sensors.VoltageSensor
  - Modelica.Blocks.Sources.Constant
    - K=5

Controlled 2 Output FB (3)

- Components Needed (cont):
  - NB_Lib.Passive.Transformer_3wind
    - L11=180e-6, L12=112e-6, L13=41e-6, L22=73e-6,
    - L23=25.5e-6, L33=9.77e-6
  - NB_Lib.Switches.SwitchIdeal
    - Ron=1e-5, Goff=1e-5, Vth = 0.0001
  - NB_Lib.Switches.Diode_CV
    - Ron=1e-5, Goff=1e-5, Von=0.8
  - NB_Lib.Controllers.PWM_Controller
    - F=75e3 Hz, Von=1
  - NB_Lib.Controllers.PI_Controller
    - kp=0.1, ki=50, LowLim=0.05, HighLim=0.95,
    - non-inv (1)

ECE ILLINOIS

Controlled 2 Output FB (4)