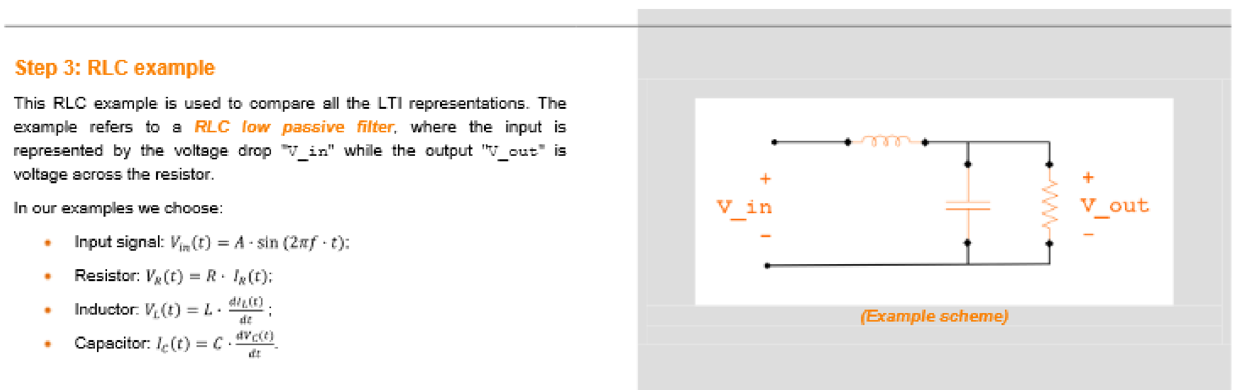
Here are some examples of RLC circuits analyzed using the following methods as implemented in SciLab:

- Differential Equation(s),
- Process Flow Diagram(s),
- State Space,
- Transfer Function,
- Zeros-Poles, and
- Modelica.

These examples will illustrate SciLab constructions for each method for (a) both a parallel and a series configuration each driven by (b) AC and DC input voltage.

Configuration I. A parallel circuit: resistor in parallel with the capacitor. Here is openeering's layout.


Step 3: RLC example

This RLC example is used to compare all the LTI representations. The example refers to a *RLC low passive filter*, where the input is represented by the voltage drop "V_in" while the output "V_out" is voltage across the resistor.

In our examples we choose:

- Input signal: $V_{in}(t) = A \cdot \sin(2\pi f \cdot t)$;
- Resistor: $V_R(t) = R \cdot I_R(t)$;
- Inductor: $V_L(t) = L \cdot \frac{dI_L(t)}{dt}$;
- Capacitor: $I_C(t) = C \cdot \frac{dV_C(t)}{dt}$.

(Example scheme)

Source.
http://www.openeering.com/sites/default/files/Control%20System%20Toolbox%20in%20Scilab.pdf

Given sinusoidal input voltage, this circuit generates a low pass filter. The examples will use

AC: The openeering example: $V_{in} = A * \sin(2\pi fq * t)$ with parameter values as specified in

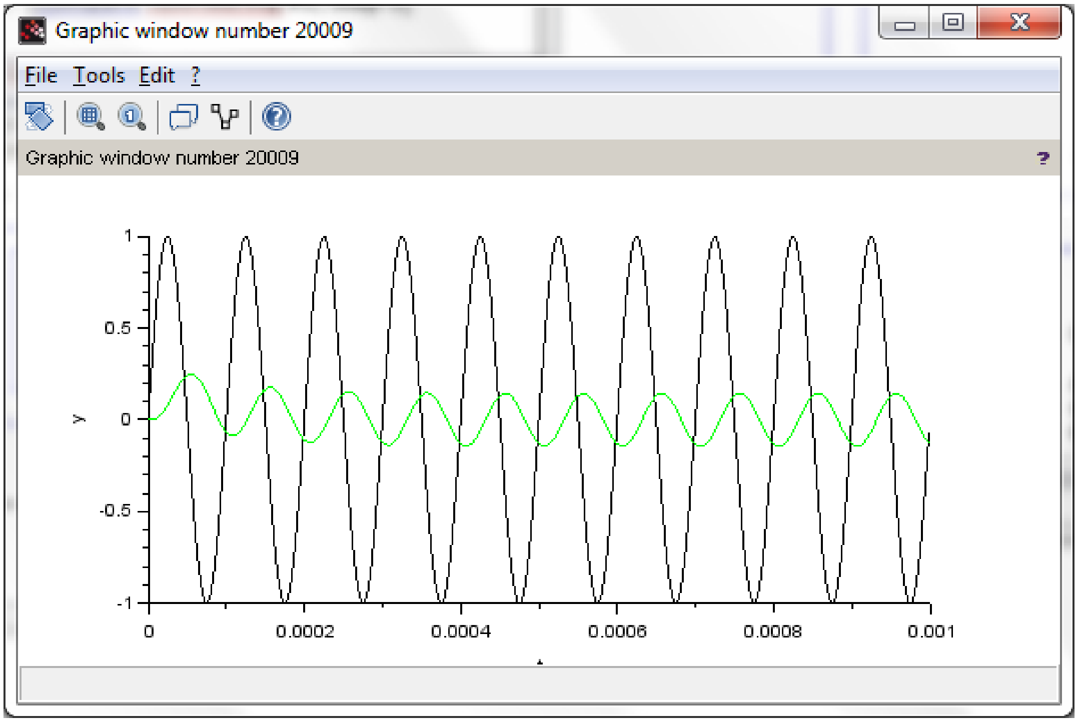
A = 1
fq = 1e+4
R = 10
L = 1e-3
C = 1e-6

DC: The same configuration as for AC but with $V_{in} = A$ and parameter values

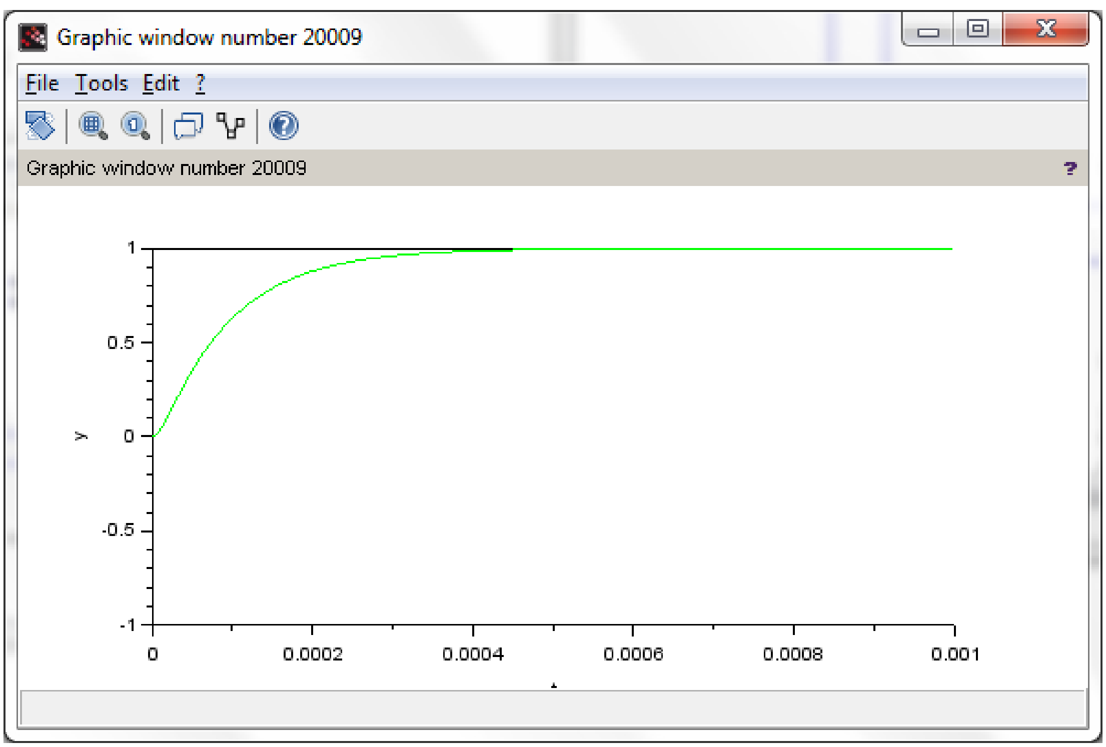
A = 1
R = 10
L = 1e-3
C = 1e-6

Results. All methods in this section (parallel configuration) generated the following graphs.

AC Graph.



DC Graph.



Example 1. Differential Equation [openeering]

Step 4: Analytical solution of the RLC example

The relation between the input and the output of the system is:

$$
\begin{aligned}
V_{in}(t) &= V_L(t) + V_C(t) \\
&= L \cdot \frac{dI_L(t)}{dt} + V_C(t) \\
&= L \cdot \frac{d}{dt}\left(I_C(t) + I_R(t)\right) + V_C(t) \\
&= L \cdot \frac{dI_C(t)}{dt} + L \cdot \frac{d}{dt}\left(\frac{V_R(t)}{R}\right) + V_C(t) \\
&= L \cdot \frac{d}{dt}\left(C \cdot \frac{dV_C(t)}{dt}\right) + \frac{L}{R}\frac{dV_C(t)}{dt} + V_C(t) \\
&= V_C(t) + \frac{L}{R}\frac{dV_C(t)}{dt} + LC \cdot \frac{d^2V_C(t)}{dt^2} \\
&= V_{out}(t) + \frac{L}{R}\frac{dV_{out}(t)}{dt} + LC \cdot \frac{d^2V_{out}(t)}{dt^2}
\end{aligned}
$$

Example 1_parAC_ODE. SciLab Code (parAC_ODE.sce).

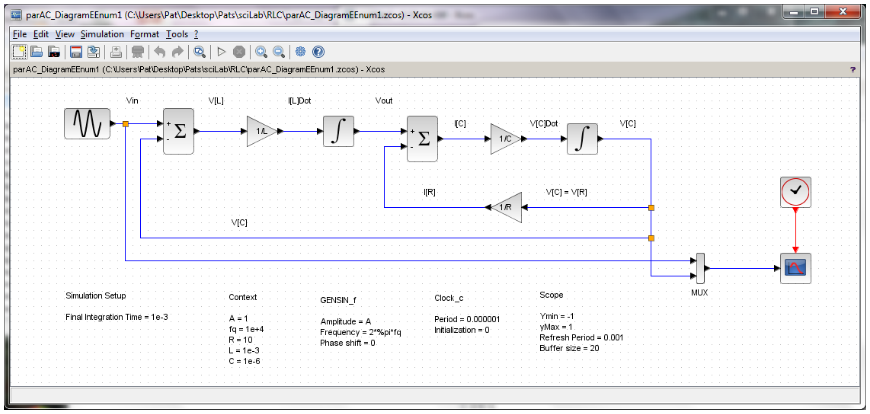
```
// ref:  www.openeering.com, "Introduction to Control Systems in SciLab
//
// Problem data
A = 1.0;              // Amplitude
fq = 1e+4;            // Frequency
R = 10;               // Resistor [Ohm]
L = 1e-3;             // Inductor [H]
C = 1e-6;             // Capacitor [F]

numPts = 1001;
lastT = .001;
dcVolts = 1;

// Problem function
function zDot=RLCsystem(t, y)
   z1 = y(1); z2 = y(2);
   // Compute input
   Vin = A*sin(2*%pi*fq*t);
   zDot(1) = z2; zDot(2) = (Vin - z1 - L*z2/R) / (L*C);
endfunction

// Simulation time [1 ms]
t = linspace(0,lastT,numPts);

// Initial conditions and solving the ODE
y0 = [0;0]; t0 = t(1);
y = ode(y0,t0,t,RLCsystem);

// Plotting results
Vin = A*sin(2*%pi*fq*t)';
scf(1); clf(1); plot(t',[Vin,y(1,:)']); legend(["Vin";"Vout"]);
```

Example 1_parDC_ODE. SciLab code (parDC_ODE.sce).

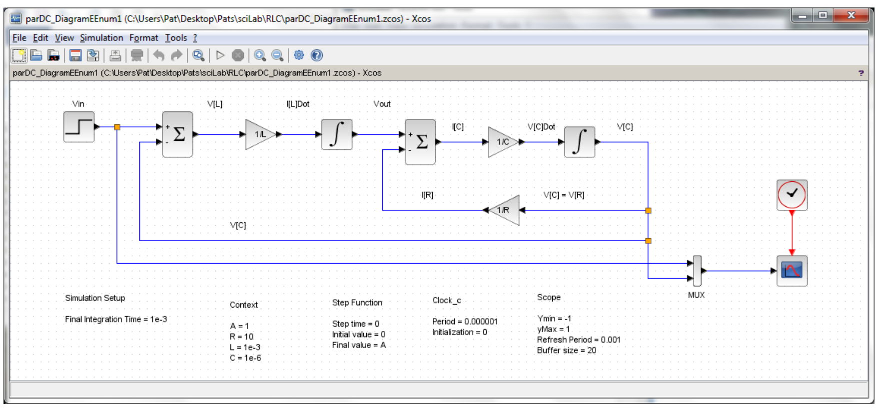
```
// ref:  www.openeering.com, "Introduction to Control Systems in SciLab
//
// Code returns a plot of Vin and Vout = Vresistor = Vcapacitor
//
// Problem data    DC input voltage = A
//
A = 1.0;              // Amplitude
R = 10;               // Resistor [Ohm]
L = 1e-3;             // Inductor [H]
C = 1e-6;             // Capacitor [F]

numPts = 1001;
lastT = .001;

// Problem function
function zDot=RLCsystem(t, y)
   z1 = y(1); z2 = y(2);
   // Compute input
   Vin = A;
   zDot(1) = z2; zDot(2) = (Vin - z1 - L*z2/R) / (L*C);
endfunction

// Simulation time [1 ms]
t = linspace(0,lastT,numPts);

// Initial conditions and solving the ODE
y0 = [0;0]; t0 = t(1);
y = ode(y0,t0,t,RLCsystem);

// Plotting results
Vin = A*ones(t)';
scf(1); clf(1); plot(t',[Vin,y(1,:)']); legend(["Vin";"Vout"]);
```

Example 2. Process Flow Diagrams.

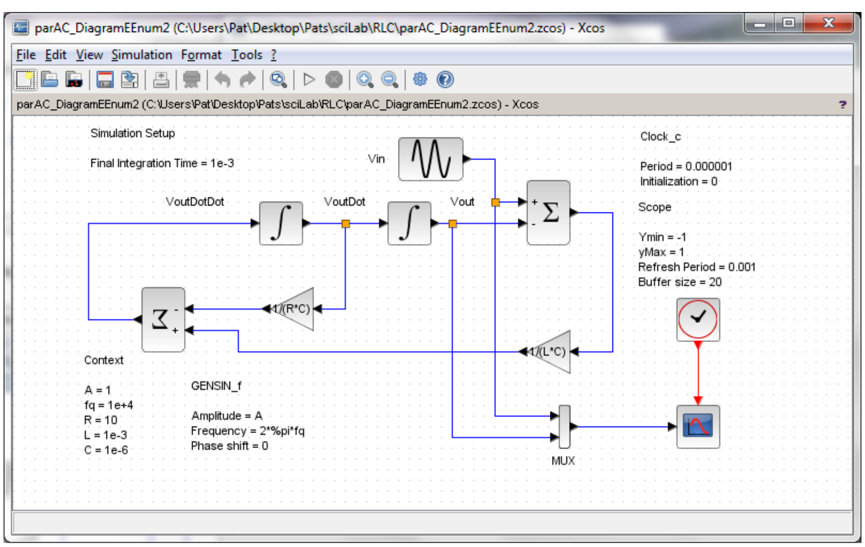
Example 2.1_parAC_Flow. [compare openeering #1, step 5]



Example 2.1_parDC_Flow.



Example 2.2_parAC_Flow2. [compare openeering #2, step 6]



Example 3. State Space [openeering]

Step 8: State space representation of the RLC circuit

In order to write the space state representation of the RLC circuit we perform the following steps:

- Choose the modeling variable: Here we use $x = (I_L(t), V_C(t))$, $u = (V_{in}(t))$ and $y = (V_{out}(t))$;
- Write the state update equation in the form $\dot{x}(t) = Ax(t) + Bu(t)$;

  For the current in the inductor we have:

  $$\dot{I}_L(t) = \frac{1}{L} V_L(t) = \frac{1}{L}(V_{in}(t) - V_C(t)) = -\frac{1}{L} V_C(t) + \frac{1}{L} V_{in}(t)$$

  For the voltage across the capacitor we have:

  $$\begin{aligned} \dot{V}_C(t) &= \frac{1}{C} I_C(t) = \frac{1}{C}\left(I_L(t) - I_R(t)\right) = \frac{1}{C} I_L(t) - \frac{1}{C}\frac{V_R(t)}{R} \\ &= \frac{1}{C} I_L(t) - \frac{1}{RC} V_C(t) \end{aligned}$$

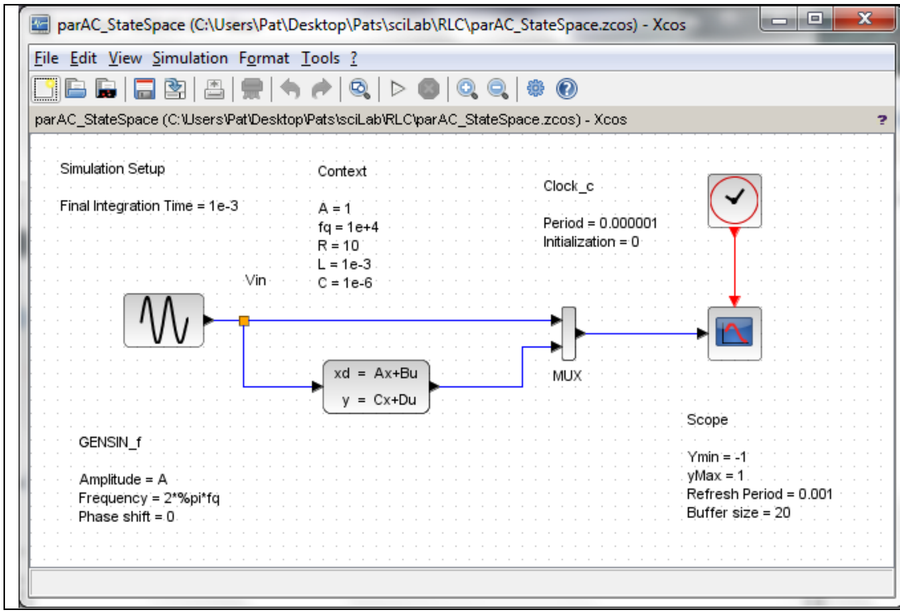
  $$\begin{bmatrix} \dot{I}_L(t) \\ \dot{V}_C(t) \end{bmatrix} = \begin{bmatrix} 0 & -\frac{1}{L} \\ \frac{1}{C} & -\frac{1}{RC} \end{bmatrix} \cdot \begin{bmatrix} I_L(t) \\ V_C(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} \cdot [V_{in}(t)]$$

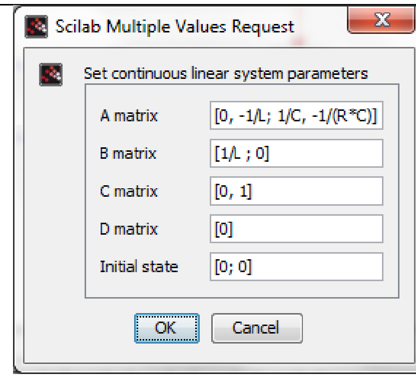
- Write the observer equation in the form $y(t) = Cx(t) + Du(t)$.

  The output voltage V_{out} is equal to the voltage of the capacitor $v_C(t)$. Hence the equation can be written as

  $$[V_{out}(t)] = [0 \quad 1] \cdot \begin{bmatrix} I_L(t) \\ V_C(t) \end{bmatrix} + [0] \cdot [V_{in}(t)]$$

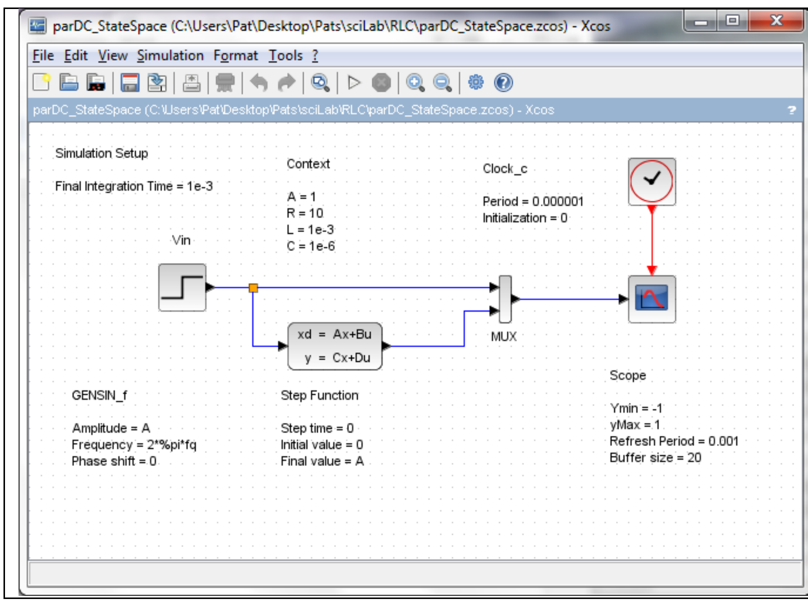
Example 3_parAC_State.

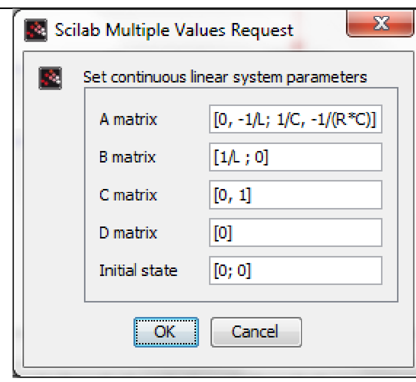




SciLab Continuous Linear State-Space (CLSS) block.

Example 3_parDC_State.





SciLab Continuous Linear State-Space (CLSS) block.

Example 4. Transfer Function [openeering]

Step 9: Transfer function representation

In a LTI ***SISO system***, a transfer function is a mathematical relation between the input and the output in the Laplace domain considering its initial conditions and equilibrium point to be zero.

For example, starting from the differential equation of the RLC example,

$$V_{in}(\mathrm{t}) = V_{out}(\mathrm{t}) + \frac{L}{R}\frac{dV_{out}(t)}{dt} + LC \cdot \frac{d^2V_{out}(t)}{dt^2}$$

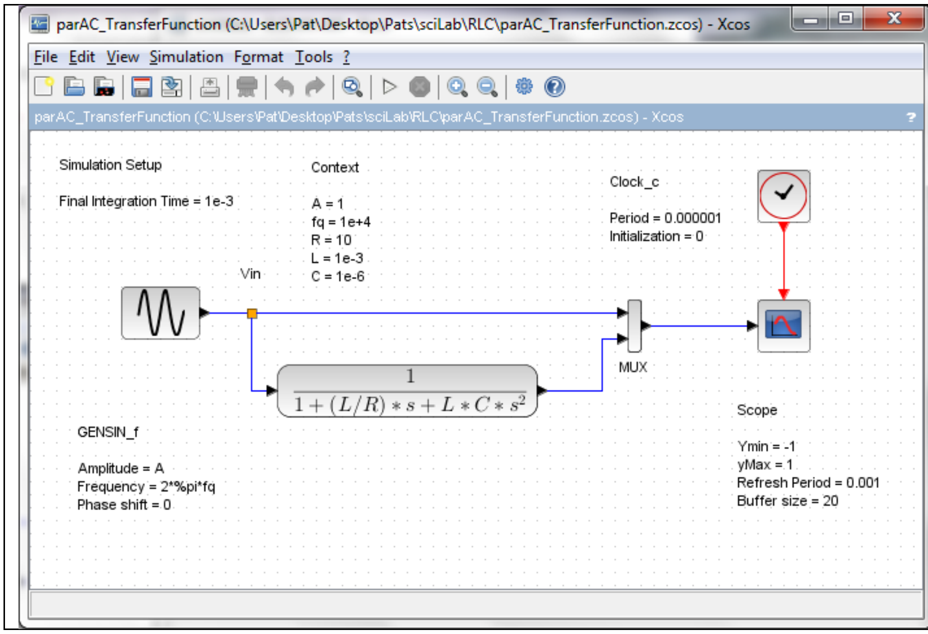
the transfer function is obtained as follows:

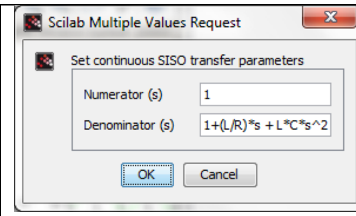
$$\begin{aligned} V_{in}(s) &= = V_{out}(s) + \frac{L}{R}s \cdot V_{out}(s) + LC \cdot s^2 \cdot V_{out}(s) \\ &= \left(1 + \frac{L}{R}s + LC \cdot s^2\right) \cdot V_{out}(s) \end{aligned}$$

that is:

$$\boldsymbol{\frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{\left(1 + \frac{L}{R}s + LC \cdot s^2\right)}}$$

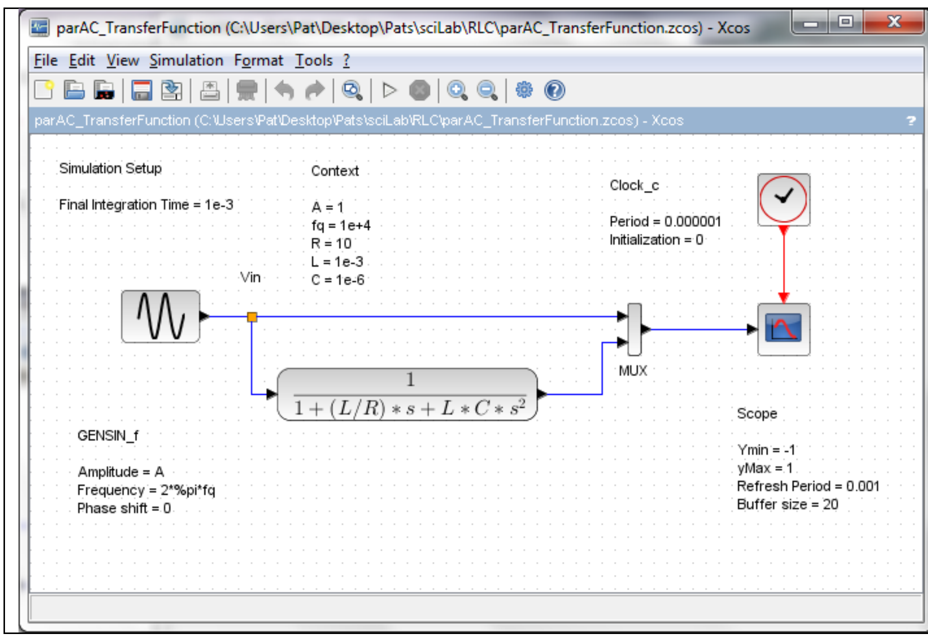
Example 4_parAC_Transfer.

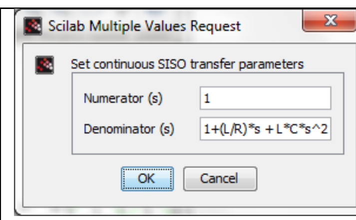




SciLab CLR block

Example 4_parDC_Transfer.





SciLab CLR block

Example 5. Zeros-Poles [openeering]

Step 11: Zero-pole representation and example

Another possible representation is obtained by the use of the partial fraction decomposition reducing the transfer function

$$H(s) = \frac{num(s)}{den(s)}$$

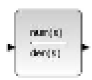
into a function of the form:

$$H(s) = \mathrm{k}\frac{(s-z_1)\cdot(s-z_2)\cdots(s-z_{nz})}{(s-p_1)\cdot(s-p_2)\cdots(s-p_{np})}$$

where k is the gain constant and z_i and p_j are, respectively, the zeros of the numerator and poles of the denominator of the transfer function.

This representation has the advantage to explicit the zeros and poles of the transfer function and so the performance of the dynamic system.

If we want to specify the transfer function in term of this representation in Xcos, we can do that using the block



and specifying the numerator and denominator.

In our case, we have

$$\frac{\mathrm{num}(s)}{\mathrm{den}(s)} = \frac{1}{\left(1+\frac{L}{R}s+LC\cdot s^2\right)} = \frac{1/(LC)}{(s-p_1)\cdot(s-p_2)}$$

with

$$p_{1,2} = \frac{-\frac{L}{R}\pm\sqrt{\left(\frac{L}{R}\right)^2-4\cdot LC\cdot 1}}{2\cdot LC} = \frac{-L\pm\sqrt{L^2-4R^2LC}}{2RLC}$$

Poles require the denominator to be written as a monic polynomial so first rewrite the Transfer Function:

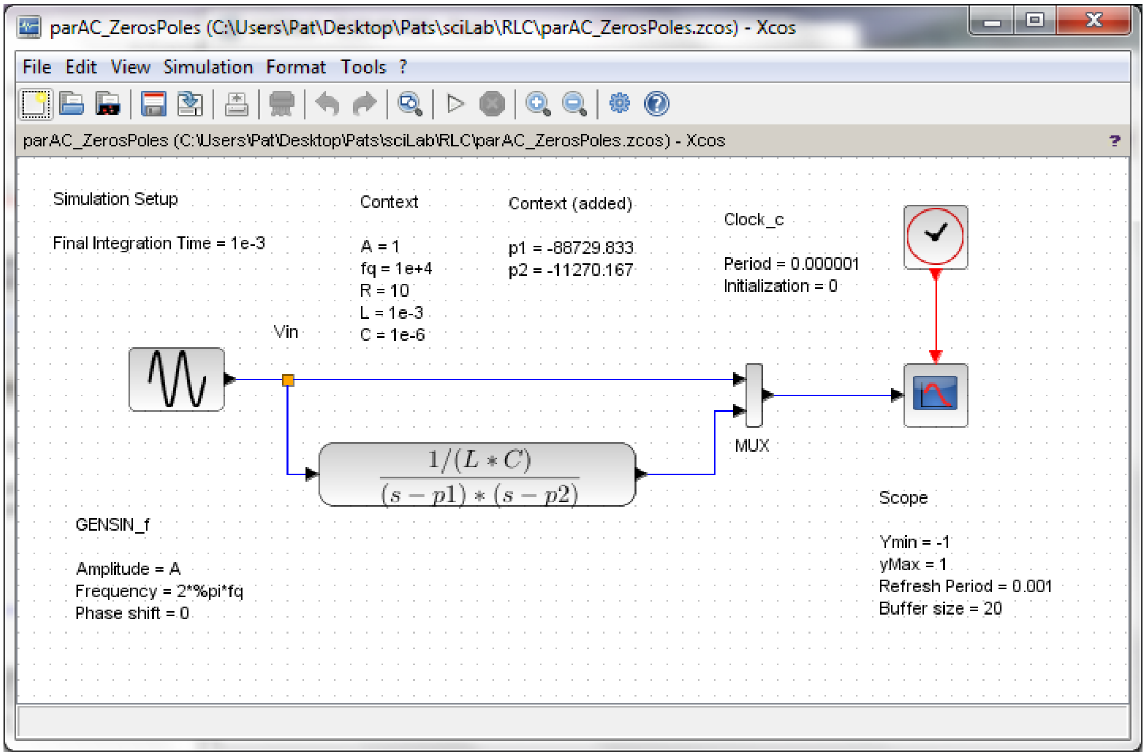
$$H(s) = \frac{1}{1+\left(\frac{L}{R}\right)s+LCs^2}$$

$$H(s) = \frac{1/LC}{1/LC+\left(\frac{1}{RC}\right)s+s^2}$$

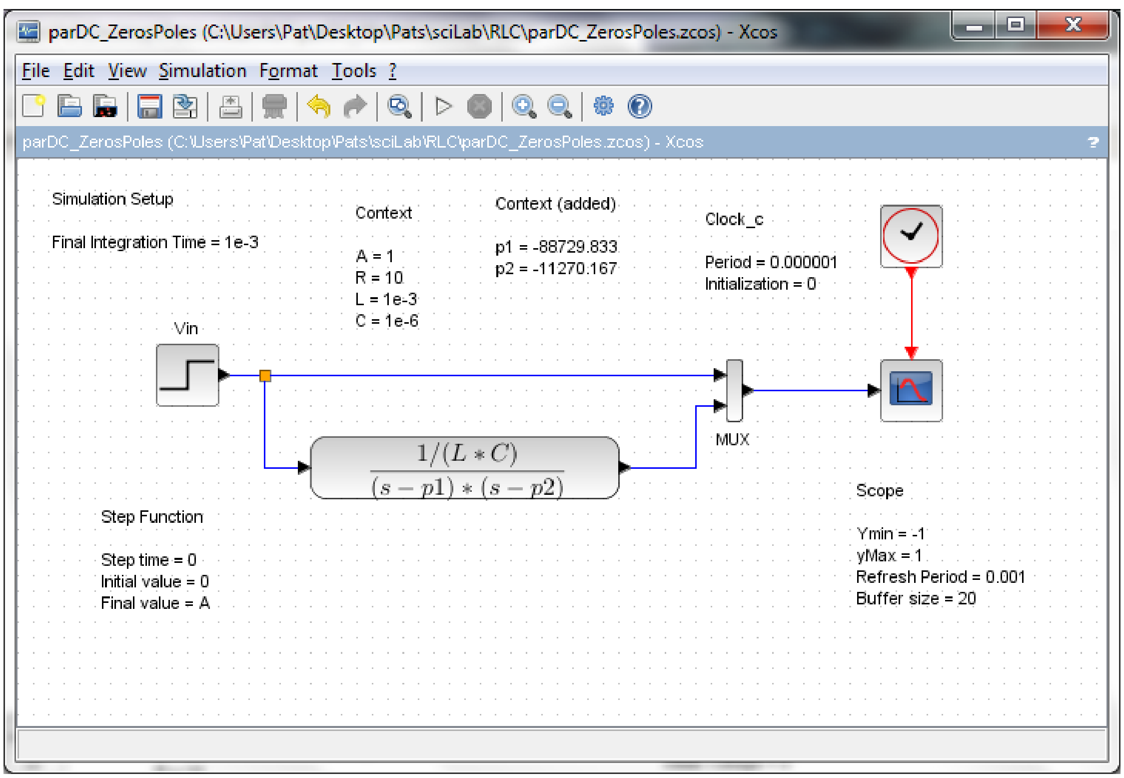
Then get the roots of the (numerator and) denominator (from SciLab):

```
-->den = [1 1/(R*C) 1/(L*C)]
 den  =     1.    100000.    1.000D+09
 -->poles = roots(den)
 poles  =   - 88729.833    - 11270.167
-->p1 = poles(1)
 p1  =   - 88729.833
 -->p2 = poles(2)
 p2  =   - 11270.167
```

Example 5_parAC_Zeros.



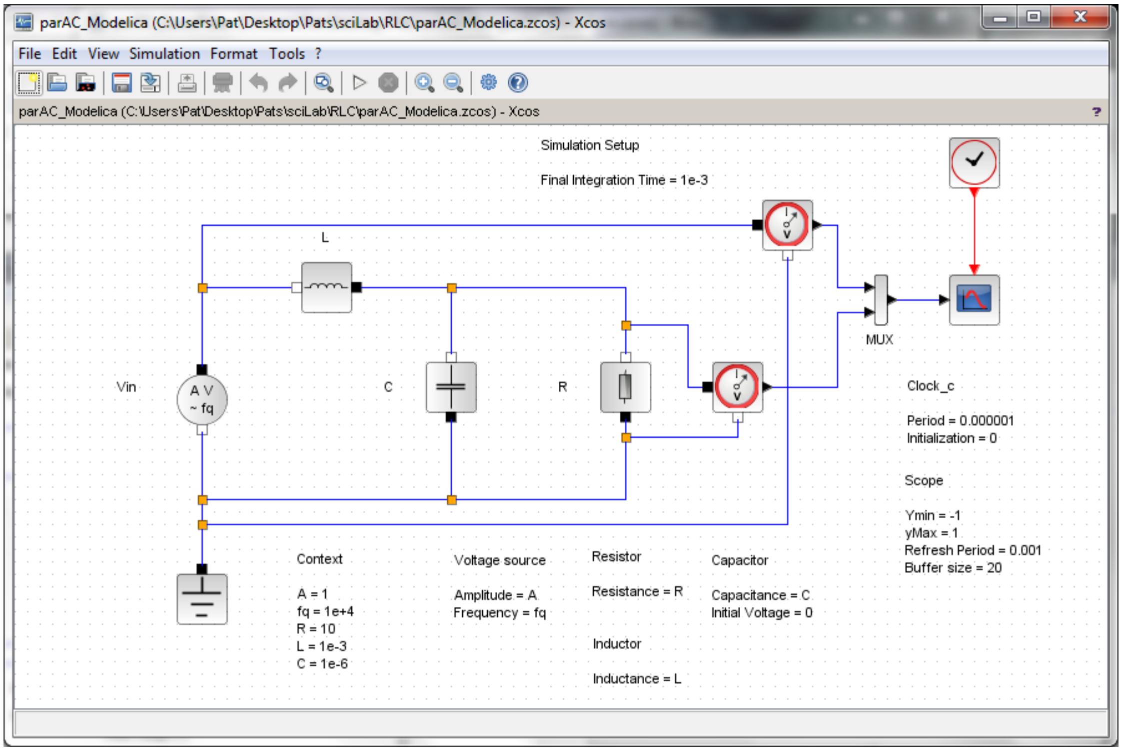
Example 5_parDC_Zeros.



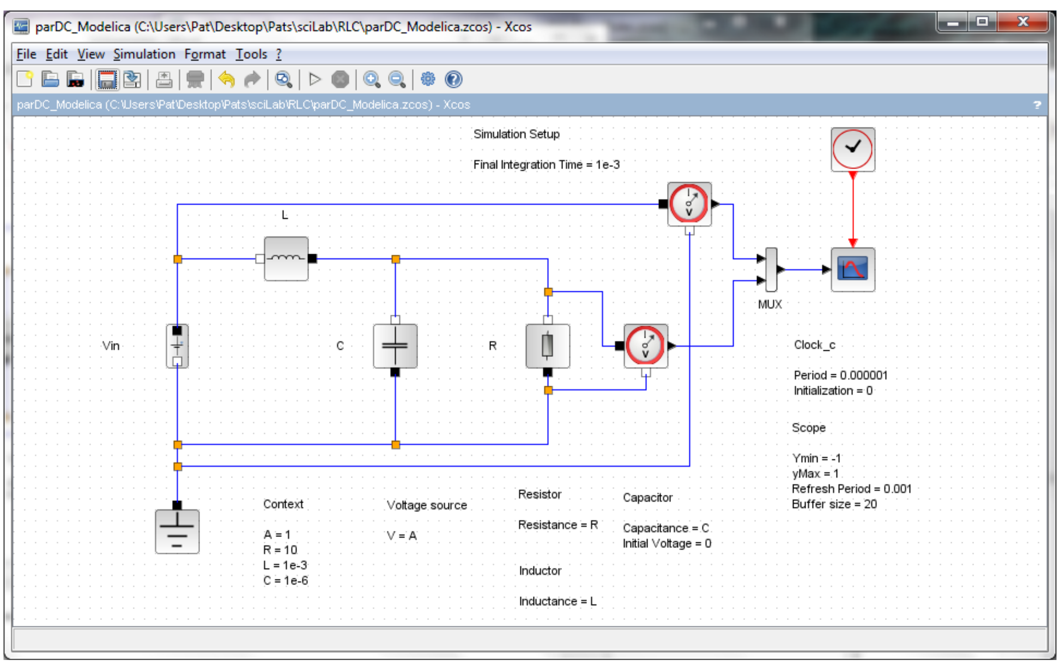
Example 6. Modelica

Example 6_parAC_Modelica.

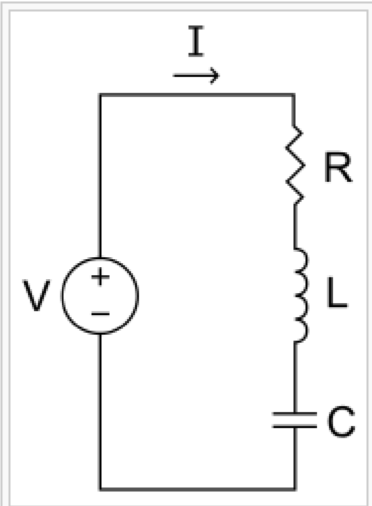
Note. The Modelica AC voltage source block expects frequency in cycles/second instead of rad/sec.



Example 6_parDC_Modelica.



Configuration II. The series circuit.

<table>
<tr><td>
</td>
<td>Figure 1: RLC series circuit

V – the voltage source powering the circuit

I – the current admitted through the circuit

R – the effective resistance of the combined load, source, and components

L – the inductance of the inductor component

C – the capacitance of the capacitor component</td></tr>
</table>

Source: https://en.wikipedia.org/wiki/RLC_circuit

As for Configuration I, we will vary the voltage between AC and DC but, this time, report the current, I(t). The examples will use

AC: $V_{in} = A * \sin(2\pi fq * t)$ with parameter values as specified in

A = 1
fq = 1
R = 1
L = 1
C = 1

DC: The same configuration as for AC but with $V_{in} = A$ and parameter values

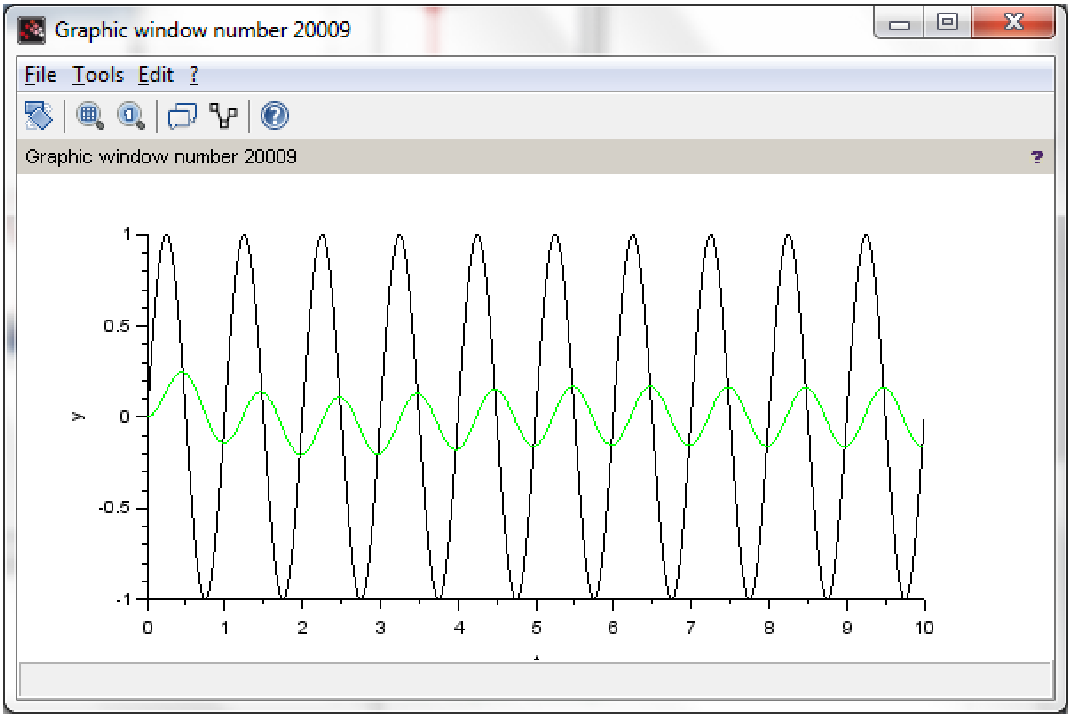
A = 1
R = 1
L = 1
C = 1

For initial conditions, take (see ocw.nthu.edu.tw/ocw/upload/12/240/08handout.pdf, pp. 16, 17, but reverse the direction of the current flow)

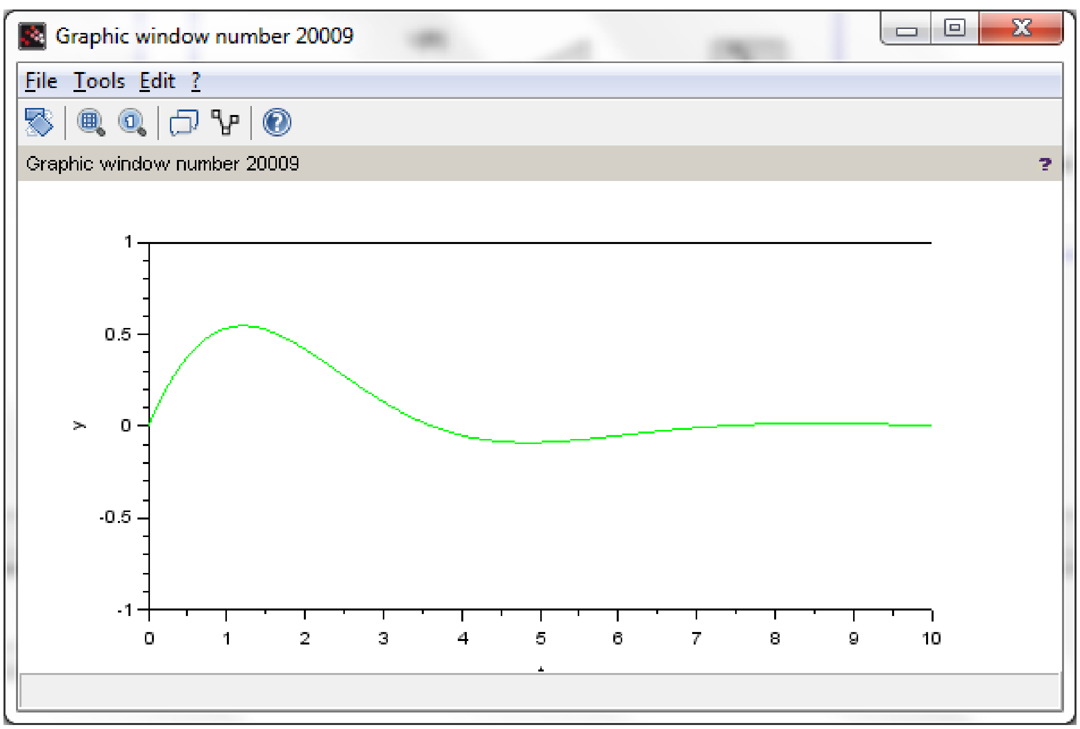
$$I(0+) = 0;\ \dot{I}(0+) = \frac{I(0+)R}{L} + \frac{V(0+)}{L} = \frac{V(0+)}{L};\ and\ V_C(0+) = V_C(0-)$$

All methods in this section (series configuration) generated the following graphs.

AC.



DC. (Compare to Wikipedia/RLC_Circuit Transient response figure.)



Analytical solution of the RLC series circuit.

From KVL,

$$\frac{d^2V_C}{dt^2} + \frac{R}{L}\frac{dV_C}{dt} + \frac{1}{LC}V_C = \frac{1}{LC}V_{in}$$

Differentiate throughout, then replace $\dot{V}_C\ by\ I$ to get

$$\ddot{I} + \frac{R}{L}\dot{I} + \frac{1}{LC}I = \frac{1}{L}\dot{V_{in}}$$

For convenience, set

$\omega_0 = \frac{1}{\sqrt{LC}}$ and $\alpha = \frac{R}{2L}$.

The resulting equation is

$$\ddot{I} + 2 * \alpha * \dot{I} + \omega_0^2 I = \frac{1}{L}\dot{V_{in}}$$

Example 1_serAC_ODE. SciLab code (serAC_ODE.sce)

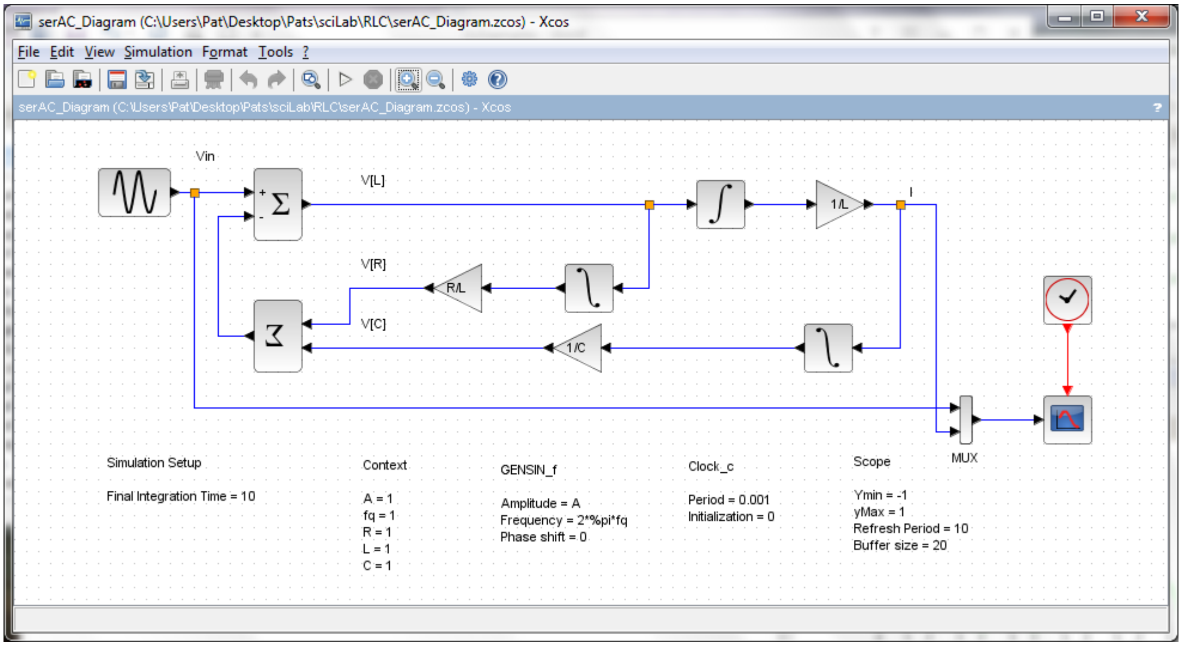
```
// cf:  www.openeering.com, "Introduction to Control Systems in SciLab,"
// wikipedia, "RLC_Circuit," and
// ocw.nthu.edu.tw/ocw/upload/12/240/08handout.pdf
//
// Code returns a plot of Vin and (resulting) current, I
//
// Problem data    AC input voltage = A*sin(2*%pi*fq*t);
//
A = 1;              // Amplitude
fq = 1;             // Frequency
R = 1;              // Resistor [Ohm]
L = 1;              // Inductor [H]
C = 1;              // Capacitor [F]

// so I(0+) = 0; IDot(0+) = V(0+)/L; see nthu reference; pp 16, 17
I0Plus = 0;
V0Plus = 0; // because sin(0) = 0
IDot0Plus = V0Plus/L;

// Following wikipedia,
alpha = R/(2*L);
omega0 = 1/sqrt(L*C);

// computational parameters
numPts = 1001;
lastT = 10;

// Problem function
function zDot=RLCsystem(t, y)
   z1 = y(1); z2 = y(2);
   // Write the derivative of the input voltage
   VinDot = 2*%pi*fq*A*cos(2*%pi*fq*t);
   zDot(1) = z2; zDot(2) = VinDot/L - omega0^2*z1 - 2*alpha*z2;
endfunction

// Simulation time [1 ms]
t = linspace(0,lastT,numPts);

// Initial conditions and solving the ODE
y0 = [I0Plus;IDot0Plus]; t0 = t(1);
y = ode(y0,t0,t,RLCsystem);

// Plotting results
Vin = A*sin(2*%pi*fq*t)';
scf(1); clf(1); plot(t',[Vin,y(1,:)']); legend(["Vin";"I"]);
```

Example 1_serDC_ODE. SciLab code (serDC_ODE.sce)

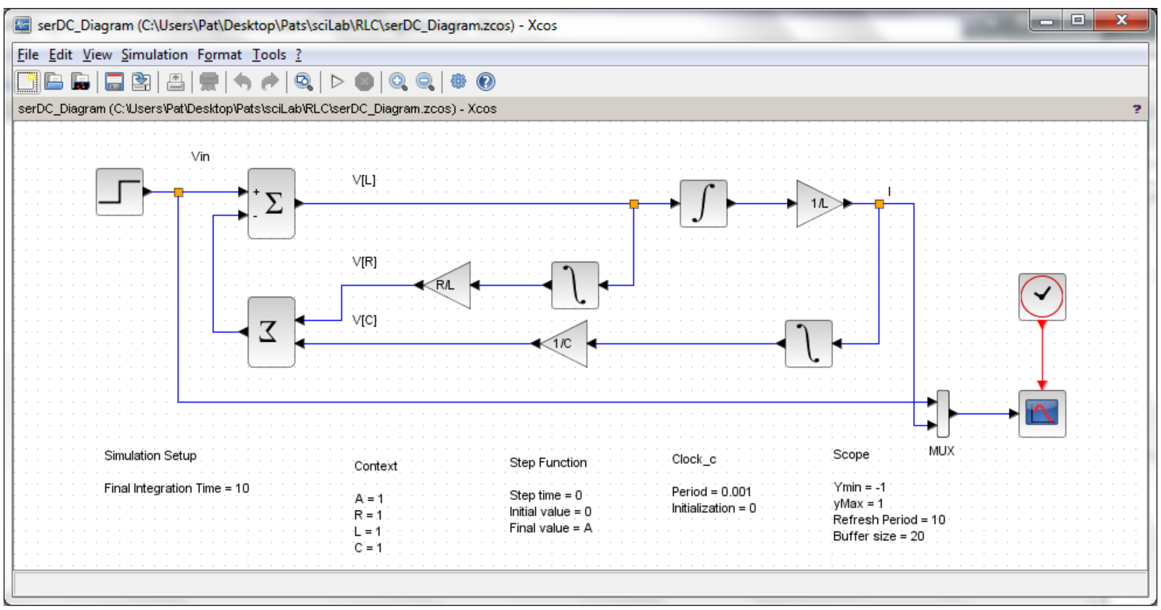
```
// cf:  www.openeering.com, "Introduction to Control Systems in SciLab,"
// wikipedia, "RLC_Circuit," and
// ocw.nthu.edu.tw/ocw/upload/12/240/08handout.pdf
//
// Code returns a plot of Vin and (resulting) current, I
//
// Problem data    DC input voltage = A
//
A = 1;              // Amplitude
R = 1;              // Resistor [Ohm]
L = 1;              // Inductor [H]
C = 1;              // Capacitor [F]

// so I(0+) = 0; IDot(0+) = V(0+)/L; see nthu reference, pp 16, 17
I0Plus = 0;
V0Plus = A;
IDot0Plus = V0Plus/L;

// Following wikipedia,
alpha = R/(2*L);
omega0 = 1/sqrt(L*C);

// computational parameters
numPts = 1001;
lastT = 16;

// Problem function
function zDot=RLCsystem(t, y)
  z1 = y(1); z2 = y(2);
  // Compute the derivative of the input voltage
  VinDot = 0;
  zDot(1) = z2; zDot(2) = VinDot/L - omega0^2*z1 - 2*alpha*z2;
endfunction

// Simulation time [1 ms]
t = linspace(0,lastT,numPts);

// Initial conditions and solving the ODE
y0 = [I0Plus;IDot0Plus]; t0 = t(1);
y = ode(y0,t0,t,RLCsystem);

// Plotting results
Vin = A*ones(t)';
scf(1); clf(1); plot(t',[Vin,y(1,:)']); legend(["Vin";"I"]);
```

Example 2. Process Flow Diagrams.

Example 2_serAC_Flow.



Example 2_serDC_Flow.



Example 3. State Space

Here, take the state variable to be $\vec{x} = \begin{bmatrix} V_C \\ I \end{bmatrix}$. Then

$$V_C = \frac{1}{C} \int I$$

$$=> \dot{V}_C = \frac{1}{C}\, I$$

$$V_{in} = V_R + V_L + V_C$$

$$=> V_{in} = RI + L\dot{I} + V_C$$

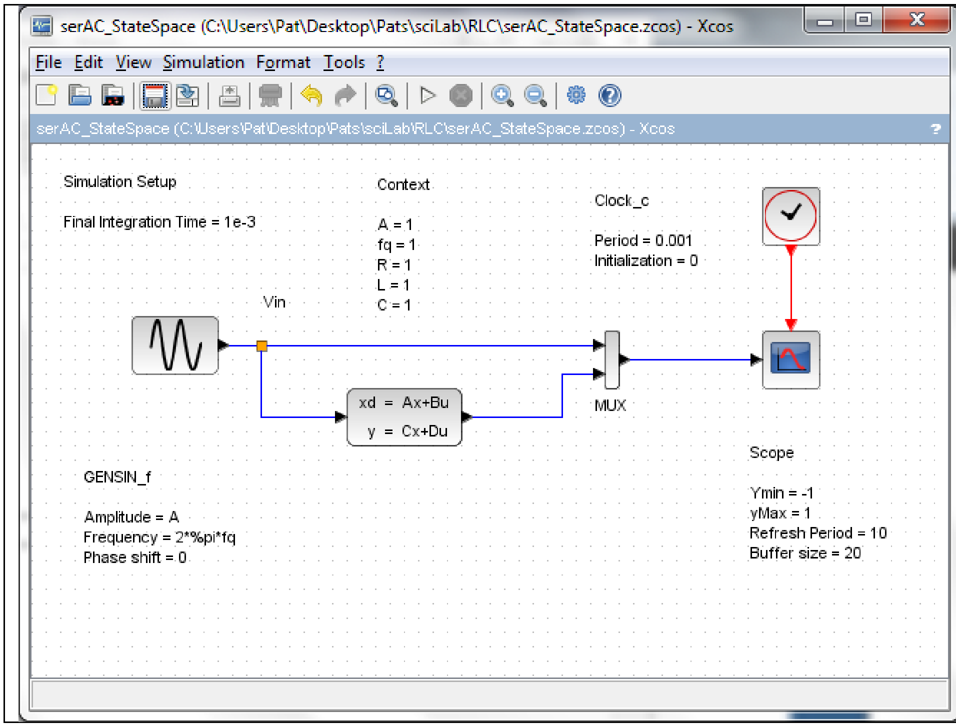
$$=> \dot{I} = \frac{-R}{L}\, I + \frac{-1}{L}\, V_C + \frac{1}{L}\, V_{in}$$

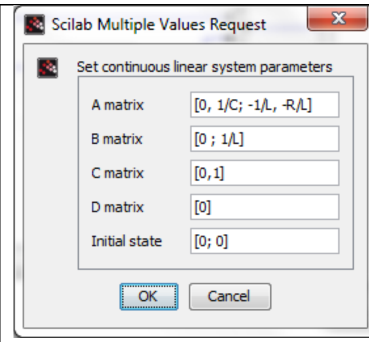
So $$\begin{bmatrix} V_C \\ I \end{bmatrix}^{dot} = \begin{bmatrix} 0 & {}^{1}\!/_{C} \\ -{}^{1}\!/_{L} & -{}^{R}\!/_{L} \end{bmatrix} \begin{bmatrix} V_C \\ I \end{bmatrix} + \begin{bmatrix} 0 \\ {}^{1}\!/_{L} \end{bmatrix} [V_{in}]$$

The initial condition is $\vec{x}(0+) = \begin{bmatrix} V_C(0+) \\ I(0+) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ because V_C and I are continuous (see o8handout.pdf, p. 5), and the system is initially quiescent.

Report $y = [0\ \ 1] \begin{bmatrix} V_C \\ I \end{bmatrix}$.

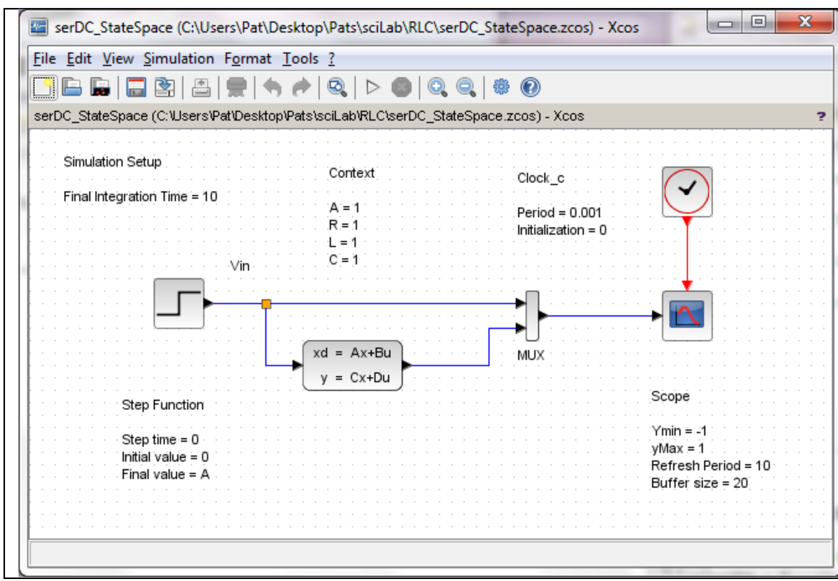
Example 3_serAC_State.

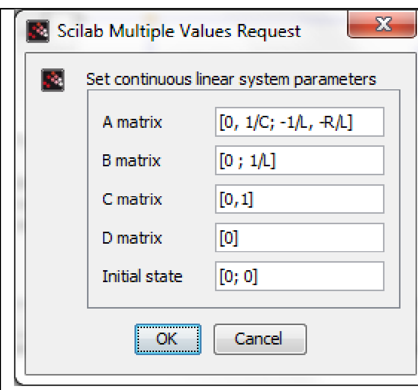




SciLab Continuous Linear State-Space (CLSS) block.

Example 3_serDC_State.





SciLab Continuous Linear State-Space (CLSS) block.

Example 4. Transfer Function.

The Differential Equation is

$$\ddot{I} + (\frac{R}{L})\dot{I} + (\frac{1}{LC})I = \frac{1}{L}\dot{V_{in}}$$

So the transfer function is

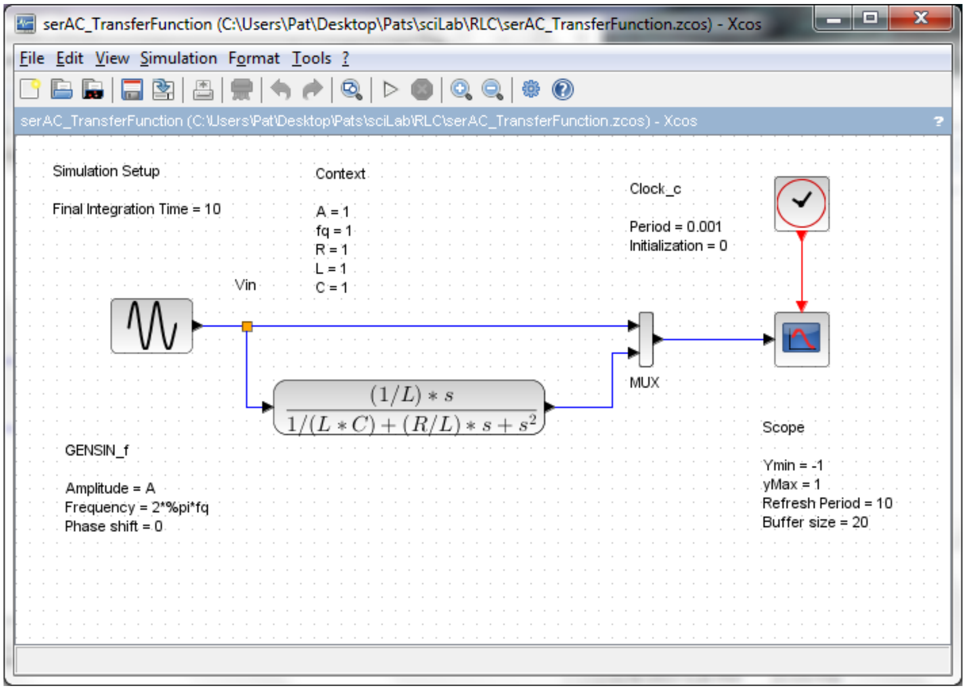
$$H(s) = \frac{I}{V_{in}}(s) = \frac{(1/L)s}{s^2+(R/L)s+(1/LC)}$$

SciLab (apparently) knows how to handle the initial conditions

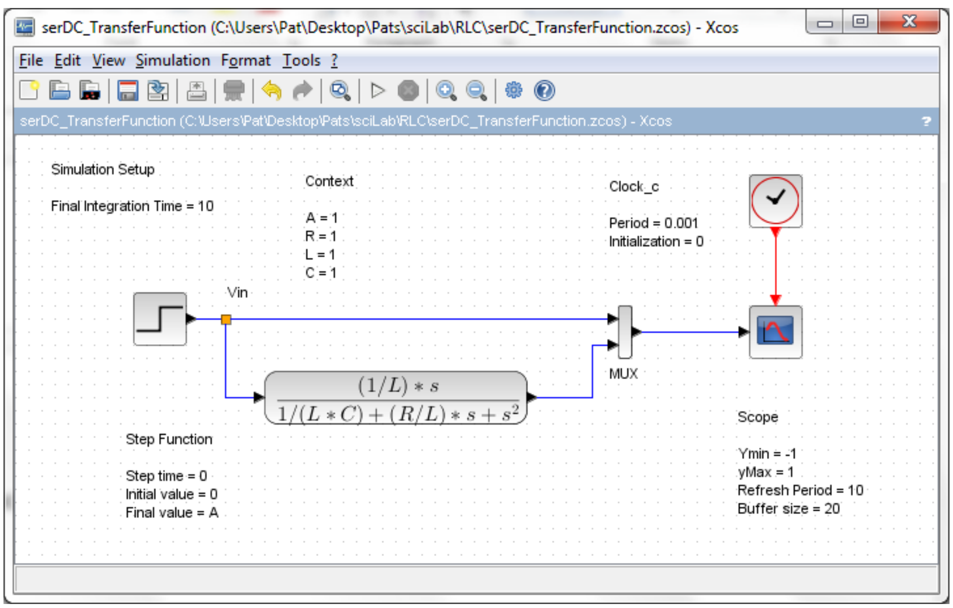
$$I(0+) = 0;\ \dot{I}(0+) = (1/L)V_{in}(0+)$$

(see 08handout.pdf, pp. 16, 17).

Example 4_serAC_Transfer.



Example 4_serDC_Transfer.



Example 5. Zeros-Poles.

Again, write the denominator as a monic polynomial, then factor it.

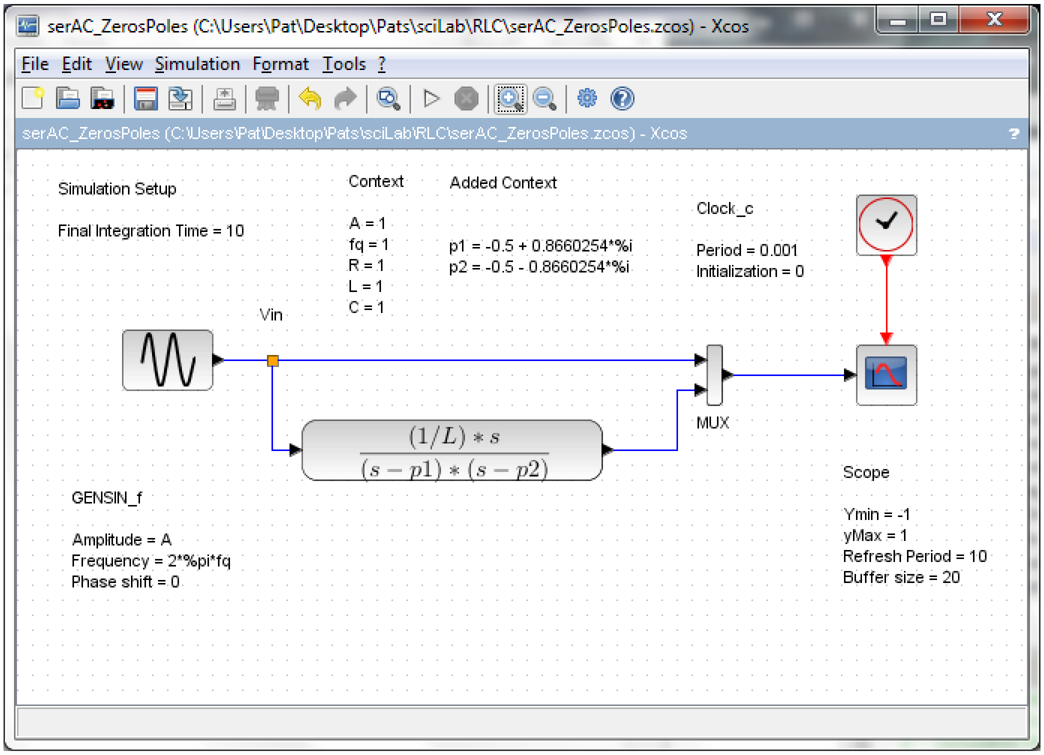
With R = L = C = 1, the denominator is

$$s^2 + s + 1$$

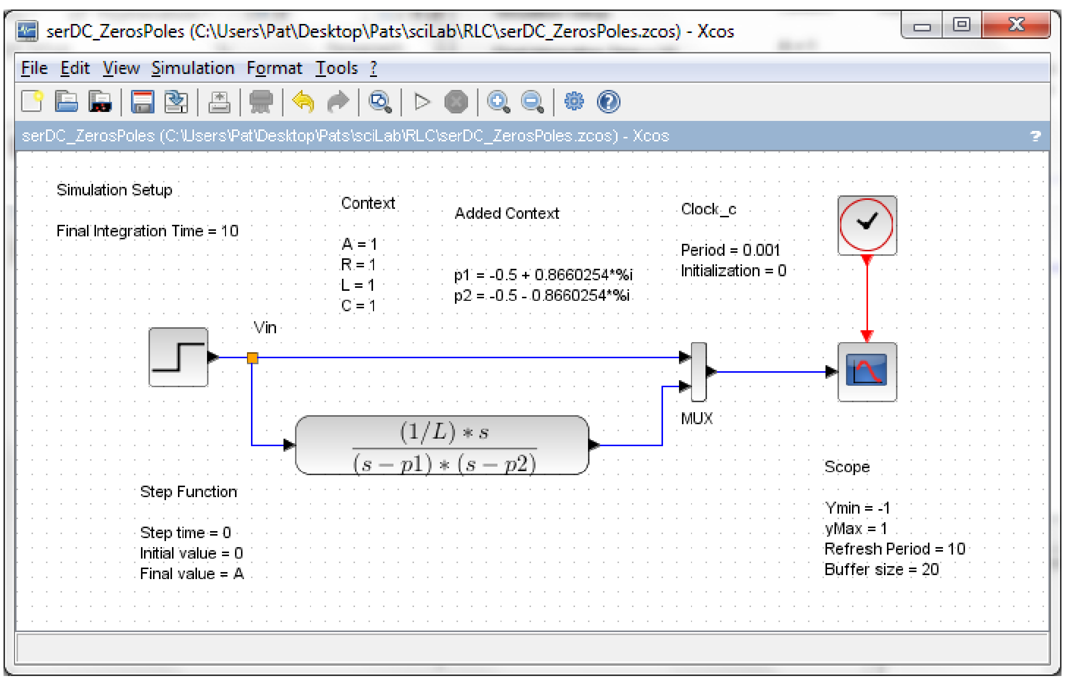
Which SciLab claims factors as $(s - p_1)(s - p_2)$ where

```
-->den = [1 1 1]
 den  =    1.   1.   1.
-->poles = roots(den)
 poles  =
              - 0.5 + 0.8660254i
             - 0.5 - 0.8660254i
-->p1 = poles(1)
p1  =  - 0.5 + 0.8660254i
-->p2 = poles(2)
 p2  =  - 0.5 - 0.8660254i
```

Example 5_serAC_Zeros.



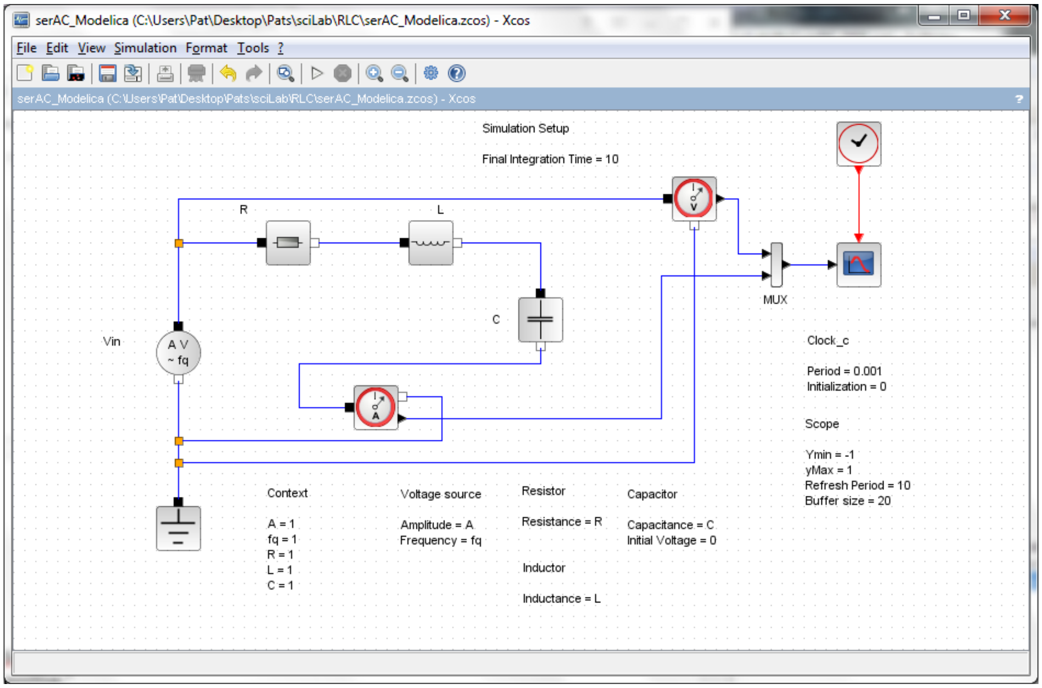
Example 5_serDC_Zeros.



Example 6. Modelica.

Example 6_serAC_Modelica.

Note. Using the CurrentSensor. From the Help Description, “Conventionally, current flowing into the black port is considered positive.”



Example 6_serDC_Modelica.