# ELE5508, Fall 2013

A Guide for Developing an MNA Formulation Program

## 1 Introduction

To generate a netlist, you would first number the nodes in the circuit starting from zero (which is always assigned to GND) and then write a line for each component in the circuit indicating which nodes it is connected to and its component value. For example, the netlist of the circuit in Figure 1 is:

```
% Circuit1.m:
% Description of Circuit 1 (Figure 1)
%
% ELEC5508
% Author:
% Date:
%=========================================================================
global G C; %define global variables
global b;

G = zeros(4,4); % Define G, 4 node circuit (do not include additional variables)
C = zeros(4,4); % Define C, 4 node circuit (do not include additional variables)
b = zeros(4,1); % Define b, 4 node circuit (do not include additional variables)

% Netlist:
cur(0,1,1e-3); % add current source between nodes 0 and 1 (1 mA)
res(1,0,1200); % add 1200 ohm resistor between nodes 1 and 0
cap(1,0,10e-9);
ind(1,2,2e-9);
res(1,2,1010);
cap(2,0,20e-9);
ind(2,3,3e-9);
res(2,3,1140);
cap(3,0,20e-9);
ind(3,4,3e-9);
cap(3,4,10e-9);
res(3,4,1450);
cap(4,0,10e-9);
res(4,0,1540);
%End
```

Note that the first few lines of the netlist are for variable initialization and need not change from circuit to circuit (except for the number of the nodes).

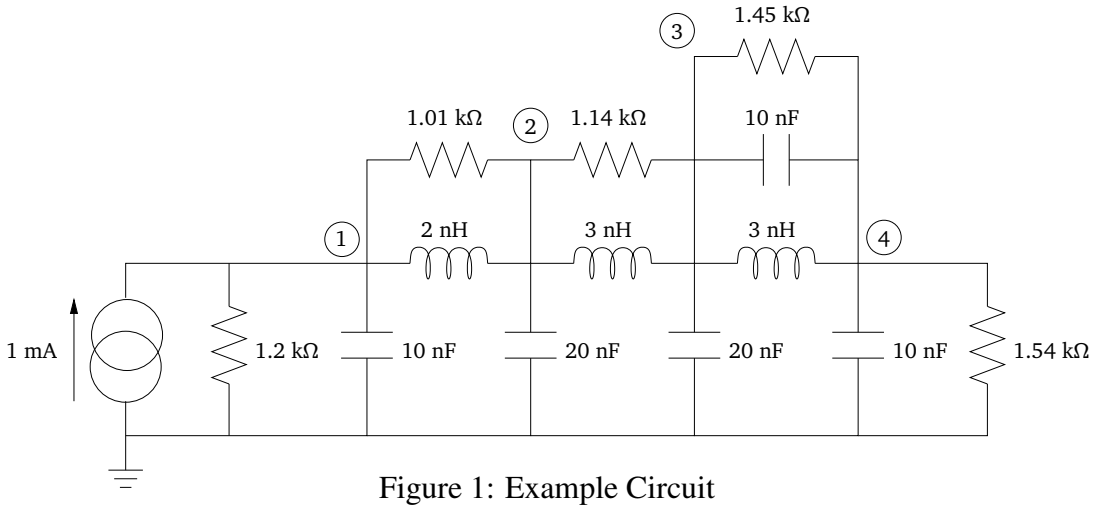


Figure 1: Example Circuit

# 2 Stamps

The netlist given in section 1 is simply a Matlab m-file. This file makes function calls to "res", "ind", "cur", etc. These functions add the appropriate stamp of the particular component to the MNA equations. These functions do not yet exist. Your task is to develop them in Matlab. Following are the functions that you need to write (some examples are given at the end of this document):

```
cur(n1,n2,val) % adds the stamp of a current source
% (current flowing from n1 to n2).
vol(n1,n2,val) % adds the stamp of a voltage source (n1 positive)
res(n1,n2,val) % adds the stamp of a resistor between node n1 and n2.
ind(n1,n2,val) % adds the inductor stamp.
cap(n1,n2,val) % adds the capacitor stamp.
vccs(nd1,nd2,ni1,ni2,val)
% Stamp of a voltage controlled current source.
% The dependent nodes are nd1 and nd2 (positive current from nd1 to nd2)
% The independent nodes are ni1 and ni2 (positive voltage at ni1).
% Ind1 to Ind2 = val*(Vni1 - Vni2)
vcvs(nd1,nd2,ni1,ni2,val)
% Stamp of a voltage controlled voltage source.
% The dependent nodes are nd1 and nd2 (positive voltage at nd1)
% The independent nodes are ni1 and ni2 (positive voltage at ni1).
% Vnd1 - Vnd2 = val*(Vni1 - Vni2)
```

# 3 Testing your program

Next, we will test the code for your stamps by considering a short example.

## 3.1 Passive Second-Order Low-Pass Filter

**1 .** Write the netlist for the circuit in Figure 2.

**2 .** Plot $|V_{out}|$ vs. frequency for 1000 equally-spaced points between DC and 10 kHz.
*Hint:* The Bodé (magnitude) plot (dBV vs. Hz) can be obtained using the following Matlab command: `semilogx(freq, 20*log10(Vout));`.

**3 .** Compare your results with Figure 3.

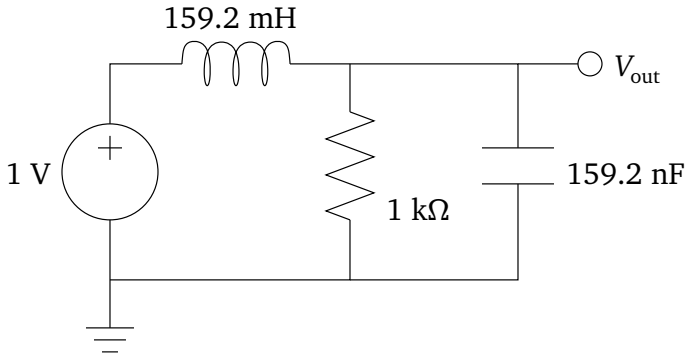


Figure 2: Passive Low-Pass Filter

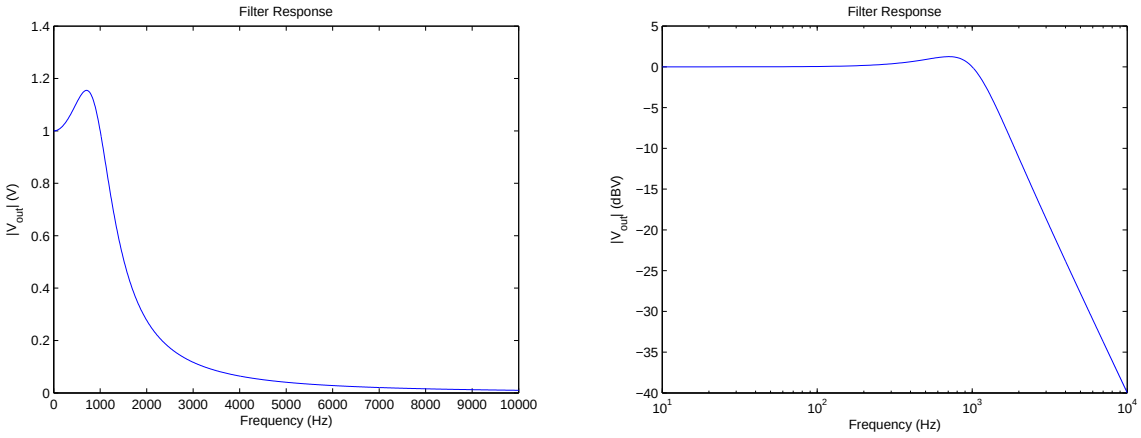


Figure 3: Passive Low-Pass Filter Response

# 4 Example Matlab files

The following is some sample code for the voltage source stamp and the capacitor stamp. You can use them as starting points for making the other stamps.

## 4.1 Voltage source

```
function vol(n1,n2,val)
% vol.m:
% Add stamp for voltage source to the global circuit representation
%
% ELEC5508
% Author:
% Date:
%=========================================================================
% define global variables
global G;
global b;
global C;

d = size(G,1); % current size of the MNA
xr = d+1; % new row
b(xr) = 0; % add new row
% Matlab automatically increases the size of a matrix
% if you use an index that is bigger than the current size.
G(xr,xr) = 0; % add new row/column
C(xr,xr) = 0; % add new row/column

if (n1 ~= 0)
    G(n1,xr) = 1;
    G(xr,n1) = 1;
end

if (n2 ~= 0)
    G(n2,xr) = -1;
    G(xr,n2) = -1;
end
b(xr) = val;
%End
```

## 4.2 Capacitor stamp

```
function cap(n1,n2,val)
% cap.m:
% Add stamp for capacitor to the global circuit representation
%
% ELEC5508
% Author:
% Date:
%=========================================================================
% define global variables
global C;

if (n1 ~= 0)
    C(n1,n1) = C(n1,n1) + val;
end

if (n2 ~= 0)
    C(n2,n2) = C(n2,n2) + val;
end

if (n1 ~= 0) & (n2 ~= 0)
    C(n1,n2) = C(n1,n2) - val;
    C(n2,n1) = C(n2,n1) - val;
end
%END
```

## 4.3 Suncircuit stamps

To add the stamps for the OpAmp model, shown in Figure-4 (as an example for the sub-circuit) the parameters of the OpAmp model can be defined in the main module as:

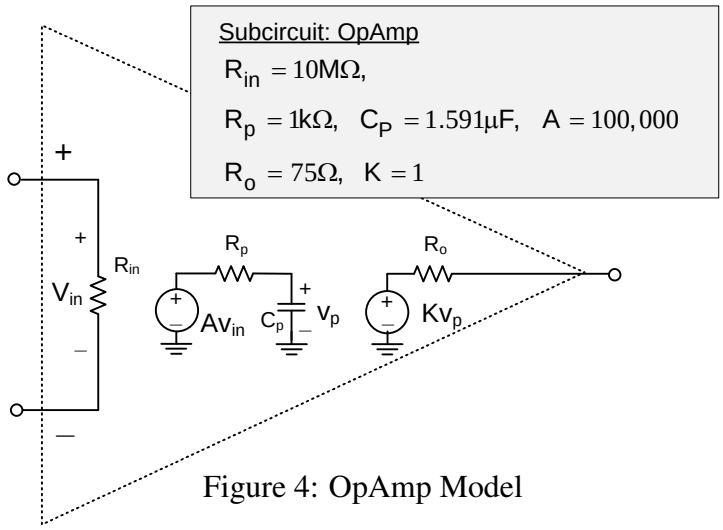


Figure 4: OpAmp Model

In the main module:

```
xpar.Rin = 10e6;      %Ohms
xpar.Rp  = 1e3;       %ohms
xpar.Cp  = 1.591e-6; %Farads
xpar.Ro  = 75;        %ohms
xpar.A   = 100000;
xpar.K   = 1;
```

For this purpose, the structure type variable that provides the means to store hierarchical data together in a single entity is used to pass the parameters to the function. Then the following `xopamp` function is invoked.

```
function xopamp(ni1,ni2 ,nd1, xpar)
% xopamp.m:
% Add stamps for the OpAmp subcircuit to the global circuit representation
% The subcircuit is known as the "basic OpAmp Model".
%
% ni1: is the node connected to the noninverting (+) input
% ni2: is the node connected to The inverting(-)input
% nd2: is the node connected to the output of opamp
%
% This function is dependent on the following functions and they should
% exist in the same directory.
% res(....), cap(....), and vcvs(....)
%
% ELEC5508
% Author:
% Date:
%=========================================================================
global G C b;

res(ni1, ni2, xpar.Rin)
```

```
%===> Expand C,G, and b for the FIRST INTERNAL NODE in the subckt:
d = size(G,1); % current size of the MNA
xr1 = d+1; % new row
b(xr1) = 0; % add new row
% Matlab automatically increases the size of a matrix
% if you use an index that is bigger than the current size.

G(xr1,xr1) = 0; % add new row/column
C(xr1,xr1) = 0;
vcvs(xr1, 0, ni1, ni2, xpar.A);

%===> Expand C,G, and b for the SECOND INTERNAL NODE in the subckt:
d = size(G,1); % current size of the MNA
xr2 = d+1; % new row
b(xr2) = 0; % add new row
G(xr2,xr2) = 0; % add new row/column
C(xr2,xr2) = 0;

res(xr1, xr2, xpar.Rp);
cap(xr2, 0, xpar.Cp);

%===> Expand C,G, and b for the THIRD INTERNAL NODE in the subckt:
d = size(G,1); % current size of the MNA
xr3 = d+1; % new row
b(xr3) = 0; % add new row
G(xr3,xr3) = 0; % add new row/column
C(xr3,xr3) = 0;

vcvs(xr3, 0, xr2, 0, xpar.K);
res(xr3, nd1, xpar.Ro);

end
```