LAB 6: Design Guide, Transient, Momentum Simulation and the DAC

Overview - This exercise teaches the fundamentals of creating filters in ADS and using the Transient simulator. The Design Guide is used to build a lumped component filter and Momentum is used to test a microstrip filter.

OBJECTIVES

• Build a 200 MHz IF low pass lumped filter using a DesignGuide

• Build a 1900 MHz RF bandpass filter in microstrip.

• Perform a Transient analysis on the microstrip filter.

• Simulate the microstrip filter in Momentum.

• OPTIONAL – DAC (data access component) exercise.

TABLE OF CONTENTS

PROCEDURE

1. Change projects and start the Design Guide.

NOTE: If you do not have access to the DesignGuide, build the filter with lumped components and save it as filter_lpf and use it for the final analysis lab.

These steps will show how a design guide can quickly and accurately produce a filter, similar to E-syn, but with more options and greater control.

a. Go to the ADS Main Window and click **File > Open Project**.

b. Answer *Yes to All* if you are prompted to save all your current work. Then open your earlier project: **system_prj**.

c. Create a new schematic named: **filter_lpf.**

d. <u>Be sure it is the only schematic opened on your screen</u>. Now, three Now, three steps are needed to start the process:

♦ Click the commands: **DesignGuide > Filter > Filter Control Control Window**.
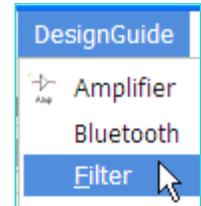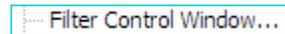
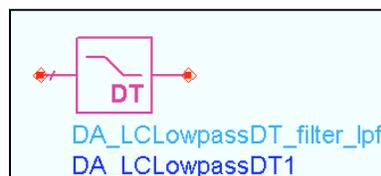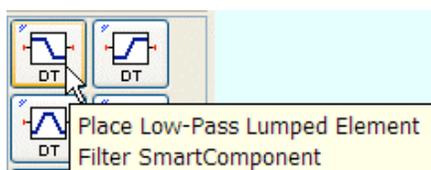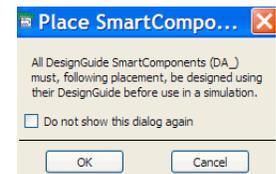♦ The new window, *Filter DesignGuide*, *DesignGuide*, will not be active until you insert a schematic *smart component* from the palette (next step).

♦ In the Filter DesignGuide control window, click on the **Component Palette – All** icon (shown here). Immediately, the palette will appear on your schematic window. You are now ready to insert smart component. You could also insert the component first and then open the Filter Design window – either method works.

2. Insert the LPF Smart Component and design the filter.

a. In the Filter DG palette, select the **Low-Pass Lumped filter filter** shown here (DT = dual terminated). If a message appears, (shown here) click **OK** to dismiss it. Then insert the the component on the schematic. Use the Esc key to end the the command.

b. Position the Filter DesignGuide window next to the schematic so it is active and click on the **Filter Assistant tab** – it should activate as shown here. If not, select the filter in the SmartComponent list. You are now ready to set the specifications.
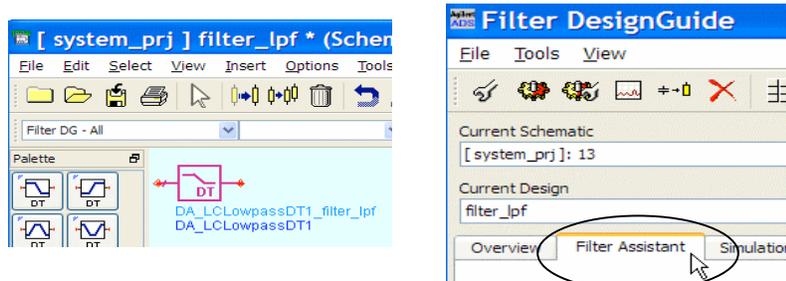
c. In the Filter DesignGuide window (shown here), enter the following three filter specifications:

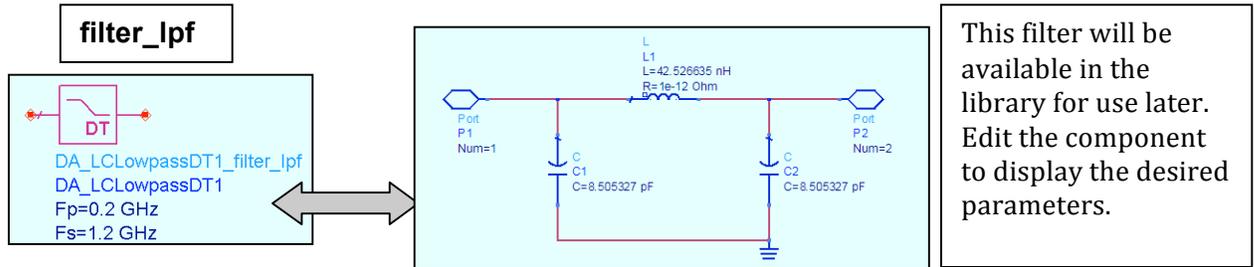> (ripple) Ap (dB) = **0.1**, (passband) Fp = **0.2** GHz, (stopband) Fs = **1.2** GHz

d. After setting the filter response, click **Redraw** to see the response plot change. This is a Butterworth response. It will provide a low pass lumped filter for the system IF output.
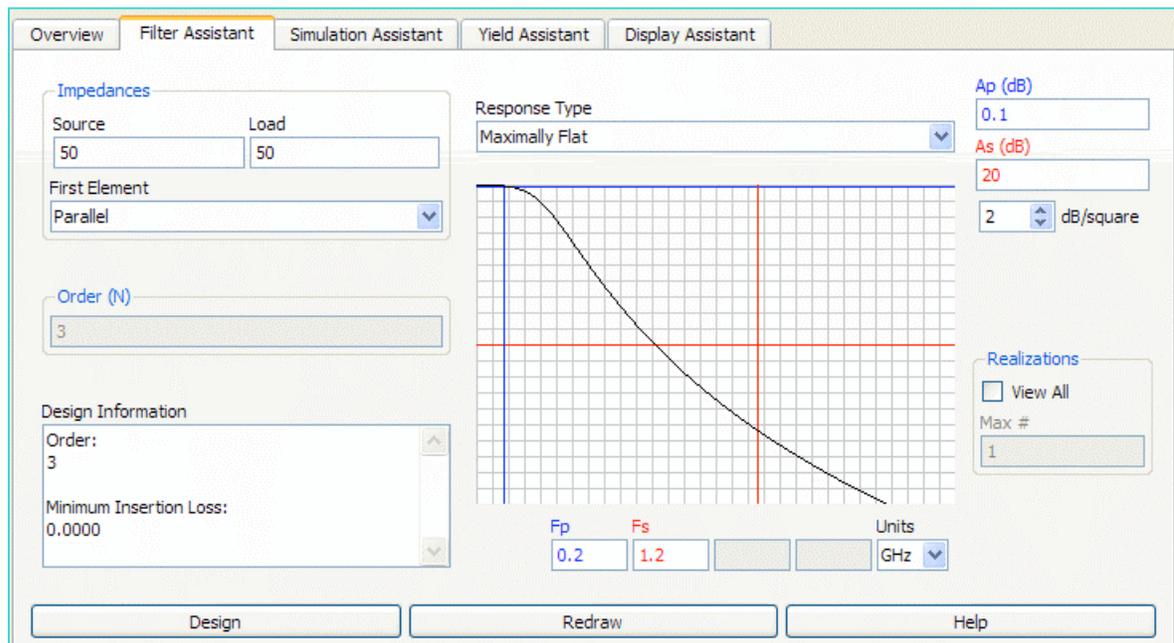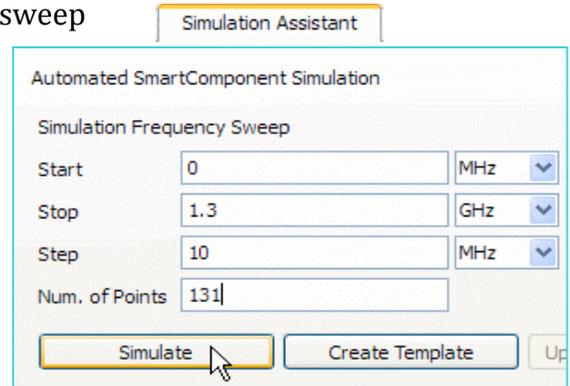

e. To design the filter, click the **Design** button.

f. In your schematic, the Smart Component now has your filter specifications and the filter circuit is created. To verify this, **push** into **push** into the DT component. Afterward, **pop** out as shown here: here:

filter_lpf

DT

DA_LCLowpassDT1_filter_lpf
DA_LCLowpassDT1
Fp=0.2 GHz
Fs=1.2 GHz

L
L1
L=42.526635 nH
R=1e-12 Ohm

Port
P1
Num=1

Port
P2
Num=2

C
C1
C=8.505327 pF

C
C2
C=8.505327 pF

This filter will be available in the library for use later. Edit the component to display the desired parameters.

g. Go to the **Simulation Assistant** and set the sweep sweep from **0 MHz to 1.3 GHz** in **10 MHz 10 MHz** steps. Then click the **Simulate Simulate** button and watch what happens: happens: a template runs the S-parameter parameter simulation and displays the the results.

Simulation Assistant

Automated SmartComponent Simulation

Simulation Frequency Sweep

Start	0	MHz
Stop	1.3	GHz
Step	10	MHz
Num. of Points	131	

Simulate   Create Template   Up

h. Examine the data display and move the the marker **M1 to 100 MHz**. Notice the the spec line at –20dB (A to B) and the the performance table that compares the compares the input parameters to the results: the –20 dB stopband is at 800 MHz which will be OK for our system.
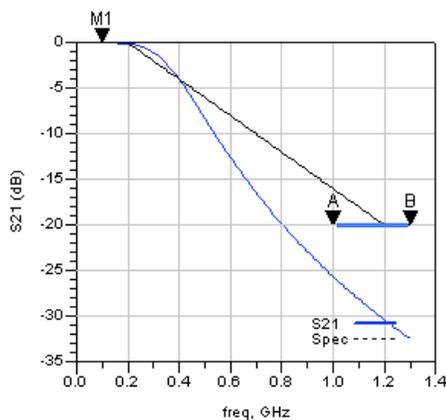
	Fp	Fs	Ap	As
Input Parameters	2.000E8	1.200E9	0.100	20.000

	PB Edge	SB Edge	Gain Dev (dB)	Delay Dev (ns)
Performance	200.0 MHz	800.0 MHz	0.100	0.167

	F	S11 (dB)	S21 (dB)	Delay (ns)
Marker M1	100. MHz	-34.39	-1.58E-3	0.88
Marker M2	1.30 GHz	-2.47E-3	-32.45	0.07

PB Edge: Actual Passband Corner
SB Edge: Actual Stopband Edge
Dev: Deviation in Passband
1/2: Input/Output Ports
Spec: Frequency Specification

F: Frequency
Fp: Passband Edge
Fs: Stopband Edge
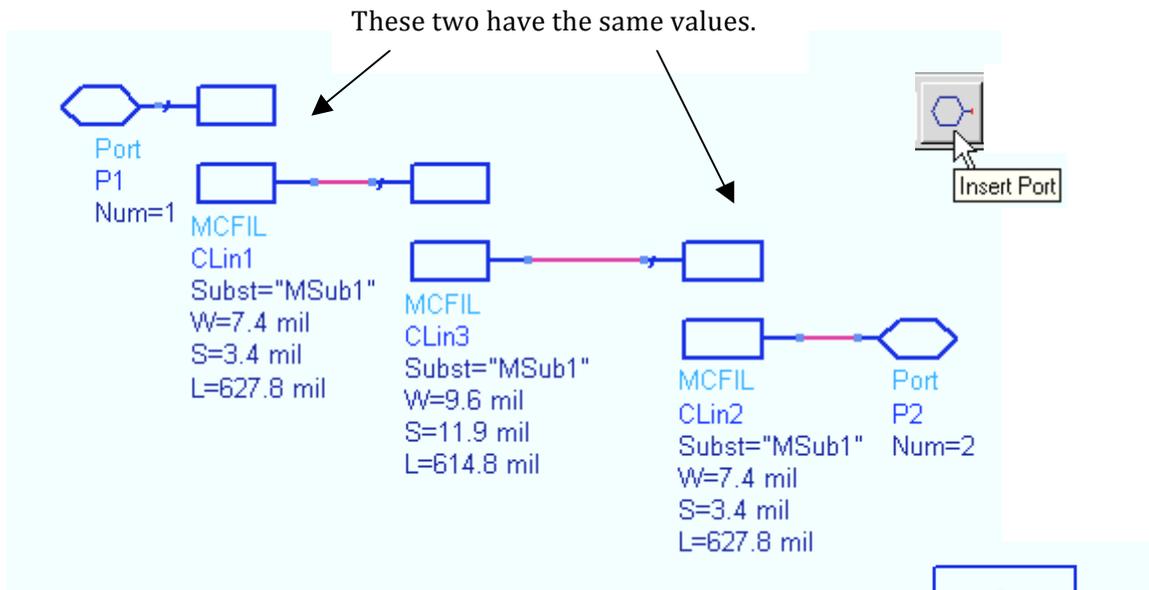Ap: Atten at PB Edge or Ripple
As: Atten at SB Edge
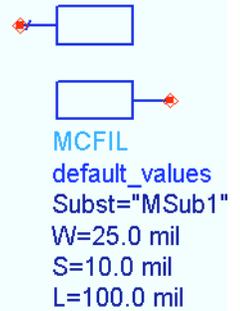
i. Scroll down and examine all the data.

j. <u>Save and close</u> all windows using the ADS main window command: **File > Close All**. You will use this output signal filter later on in the course. Next, you will build the input filter.
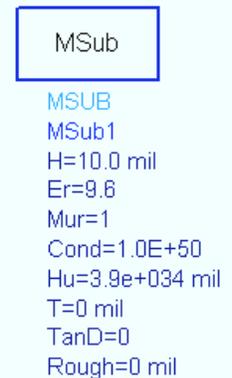
6-5

3. Microstrip 1900 MHz bandpass filter.

In these next steps, you will build and simulate a coupled line filter with the ADS circuit simulator. Afterward, you will transfer it to ADS layout and simulate using Momentum. This step is only intended to briefly show how Momentum works in a simple microstrip application.

a. Create a new schematic in the current project (system_prj) named: named: **filter_1900.**

b. From the **TLines-Microstrip palette**, insert an **MCFIL** as shown. Because the two end components are symmetrical (CLin1 and CLin2), save time by typing the W,S, and L parameters (shown here) for the first one and then copy it. Then insert the middle component (Clin3) and type in its values.

c. Insert port connectors on each end as shown also.

These two have the same values.

```
Port
P1
Num=1  MCFIL
       CLin1
       Subst="MSub1"
       W=7.4 mil
       S=3.4 mil
       L=627.8 mil

                MCFIL
                CLin3
                Subst="MSub1"
                W=9.6 mil
                S=11.9 mil
                L=614.8 mil

                        MCFIL       Port
                        CLin2       P2
                        Subst="MSub1"   Num=2
                        W=7.4 mil
                        S=3.4 mil
                        L=627.8 mil
```

Insert Port

d. From the Microstrip palette, insert the substrate definition **MSUB** as shown - no other settings are required – the default substrate will be used.

```
MSub

MSUB
MSub1
H=10.0 mil
Er=9.6
Mur=1
Cond=1.0E+50
Hu=3.9e+034 mil
T=0 mil
TanD=0
Rough=0 mil
```

Mcfil

MCFIL
default_values
Subst="MSub1"
W=25.0 mil
S=10.0 mil
L=100.0 mil

NOTE on the filter S-21 response – The S-parameter simulation (S-21) of this filter is shown here as a reference - there is no need to simulate it with the S-parameter simulator at this time.
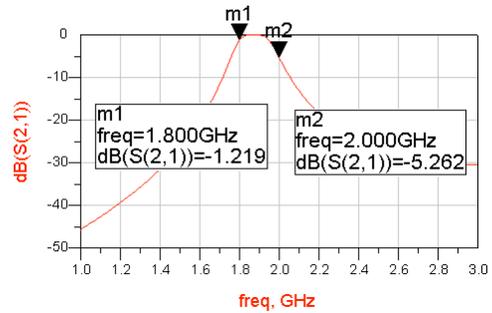
e. In the schematic, click: **File > Design Parameters**. When the dialog appears, select the ADS built-in symbol for a bandpass filter: **SYM_BPF** and click **OK**.


Saved design: filter_1900 for use later on in the system.
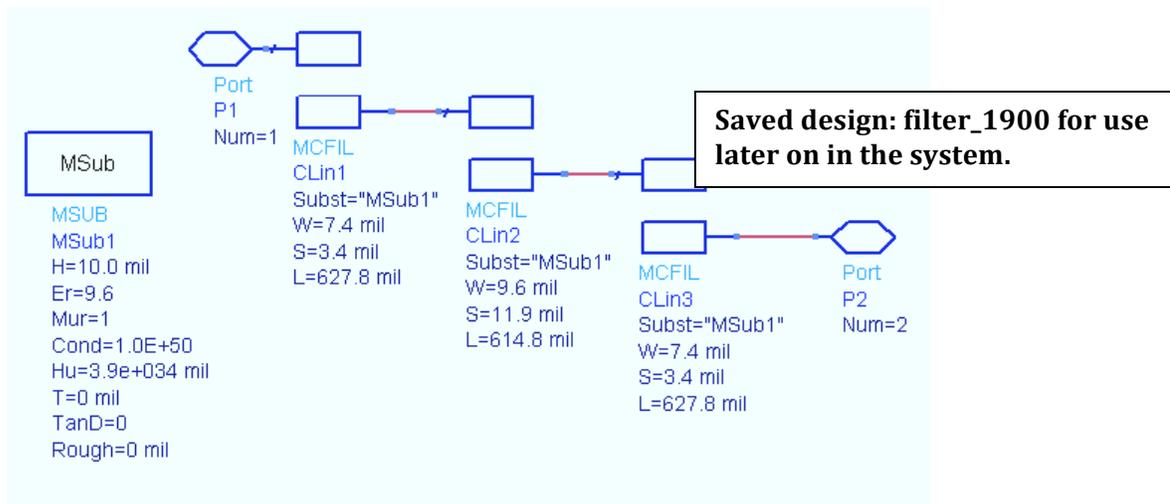
f. **Save** the design again so that all the modifications are also saved. This is the filter that will be used for the following steps and for the final system design.

4. Transient Analysis on the microstrip filter.

a. Save the schematic (filter_1900) with a new name: **filter_trans**.

b. Modify the schematic to look like the one shown below by **deleting** the port connectors.

c. From the Sources-Time Domain palette palette insert a **VtSine** at the input (be input (be sure to ground it as shown). It shown). It is recommended to use Time use Time Domain sources for Transient Transient analysis.
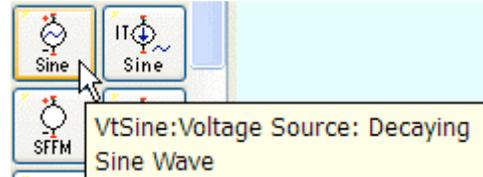

d. Insert a **50 ohm** resistor at the input and on the output (with ground).

e. Label pin/nodes **Vin** and **Vout** as shown. The circuit should look like the one shown here.
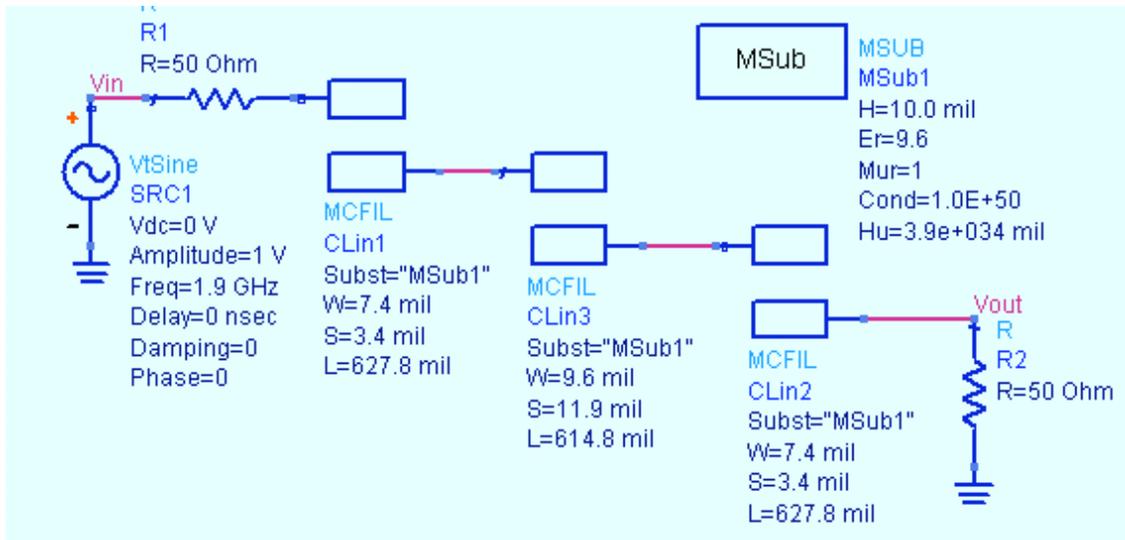
f. From the **Simulation-Transient** palette, insert a Transient simulation controller and set the Stop Time and MaxTimeStep as shown. The simulation will begin at time zero (default) and stop after fifteen periods of the input signal (8 ns). In addition, the time step will sample at twice (Nyquist rule) the rate of the highest spectral component desired, here the 15th harmonic.

6-9
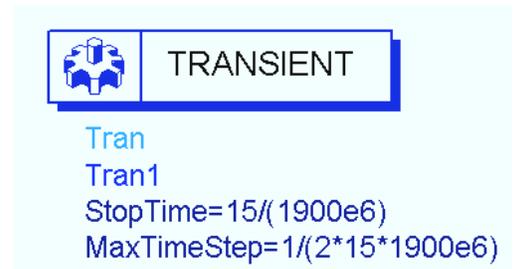
g. **Simulate** and after the simulation, insert a rectangular plot of **Vin** and **Vout** and edit each trace (Trace Options) to **place symbol at data** – this will show you the time points taken in the simulation.

Trace Options Tab – Linear Tab
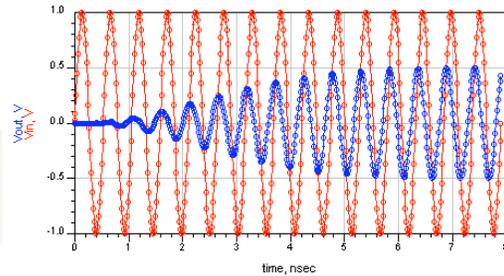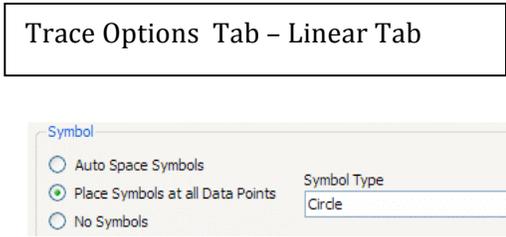
h. **Zoom** into the plot <u>after 5 ns</u> and put and put the markers on the peaks of peaks of Vin and Vout as shown here. here.

i. **Write an equation** as shown to calculate the delay through the filter: filter: **marker_difference**. This calculates the X axis difference between markers using the **indep** **indep** function (independent variable = time).
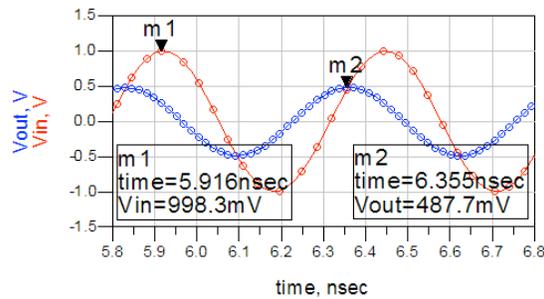
Eqn marker_difference = indep (m2) - indep (m1)

j. **Insert a list** of **marker_difference** and use Plot Options to remove the independent data as shown. The value is the delay through the filter after start up (settling). Depending upon where you zoom in and where you place the markers, your value may differ slightly. The delay through the filter is about 44 pico seconds as shown here.

☐ Display Indep. Data

marker_difference
4.386E-10

k. Save the schematic but keep it opened. Save and close the data display.

l. The next steps will be to simulate this filter in Momentum.

5. **Momentum simulation from layout.**

   a. Open **filter_1900** and save it with a new name: **filter_mom**.

   b. Transfer the filter_mom schematic to layout by clicking clicking the schematic window command**: Layout > Generate/Update Layout**.
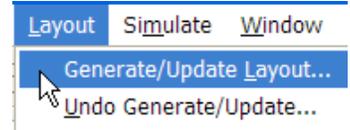
   c. When the next dialog appear, be sure the Starting Component is **P1** (port connector 1 from schematic) so that the left-to-right layout will be generated correctly. Click **OK** and you should see another dialog indicating that all components have been created in the layout window.
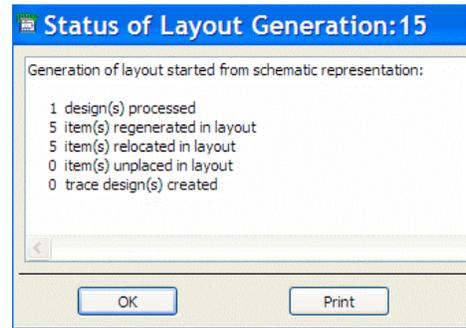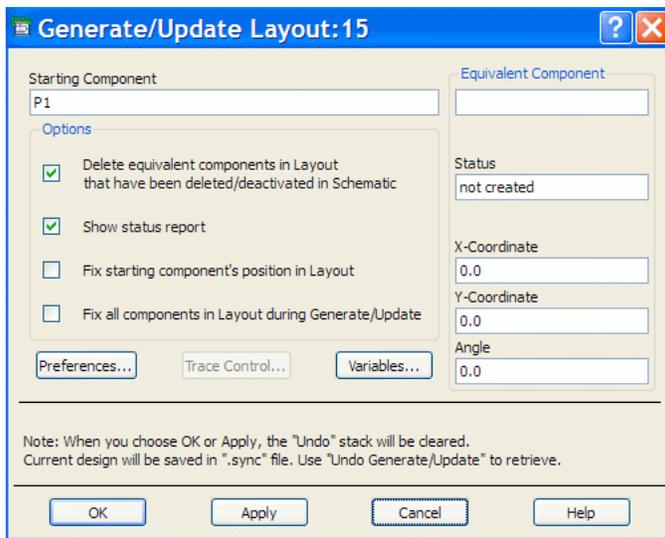
   d. The layout window and the Layers control will open. In the the layout, verify that you have three long coupled lines with two with two ports. Zoom in on a port (shown here), and verify that it verify that it is connected to the edge of the metal. For many many Momentum solutions, it is not necessary for the port to to connect to the middle of the line – it only needs to be on the
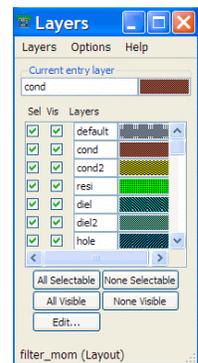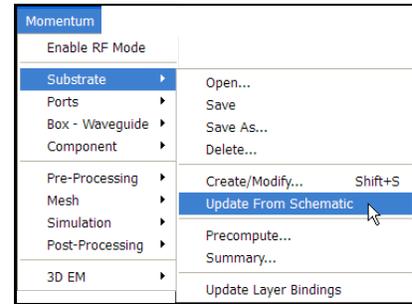
Momentum ports can be inserted in layout if the drawing is created in layout.

Zoomed-in port connector.

6-11
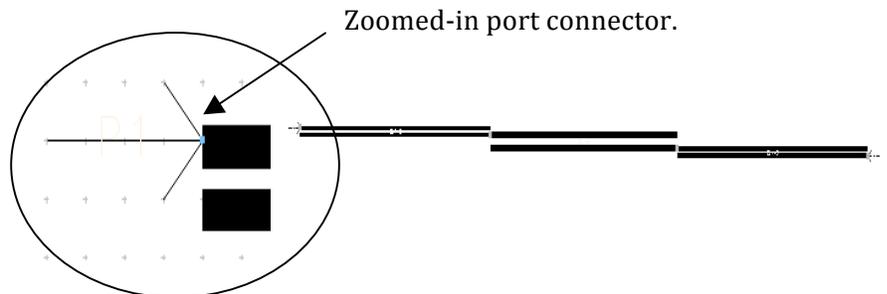
the edge.

e. The next step is to define the substrate in Momentum. To do this, use the Layout commands shown here to transfer the schematic schematic MSUB definition: **Momentum > Substrate > Update From Schematic**.

f. Verify that the 10mil substrate definition is now in Momentum. Click: **Momentum > Substrate > Create/Modify** and you should see the MSUB values in the dialog.
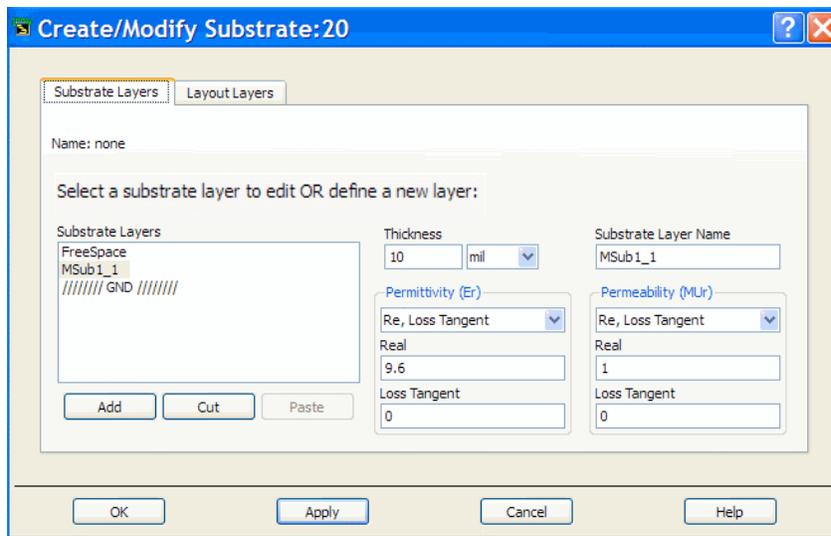
g. You can also look at the **Layout Layers** tab to see how the drawing layers in layout are mapped to the substrate: *cond* is a metal strip layer. But DO NOT change anything. Click **OK** to dismiss the dialog.

6-13
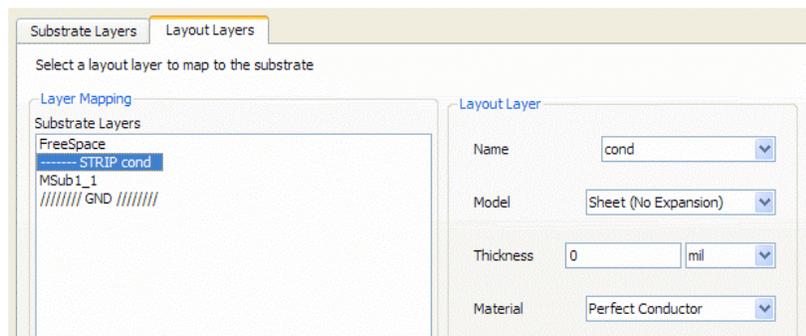
h. Before simulating, look at the Momentum mesh features. Click**: Momentum > Mesh > Setup**. When the When the dialog appears, notice there are many mesh mesh features. But for now, be sure the Edge Mesh box Mesh box is unchecked as shown and click **OK**.

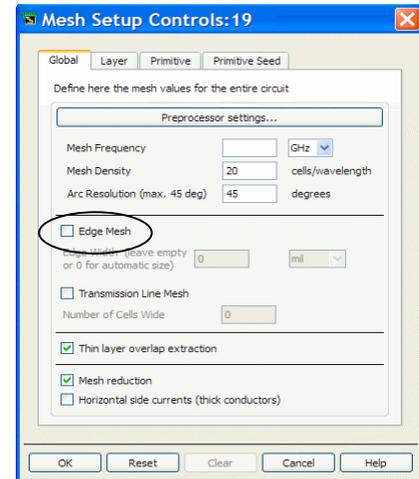**NOTE on the Momentum process** - the order of steps is: compute the substrate definition (Green's functions), create the mesh or grid pattern (cells), and then simulate.

i. To simulate, click**: Momentum > Simulation > S-parameters**. When the dialog appears, type in the sweep as shown: **1 GHz** to **3 GHz** using **25 points** as the limit. Click **Update** and it will appear in the Frequency Plan. Next, click the **Simulate** button and click **Yes** if the Question dialog appears. Watch the status window - the Adaptive sweep type is the default and its curve fitting-like algorithm will run the EM (electromagnetic) simulation by solving selected points.

Momentum autoplot feature

© Copyright Agilent Technologies
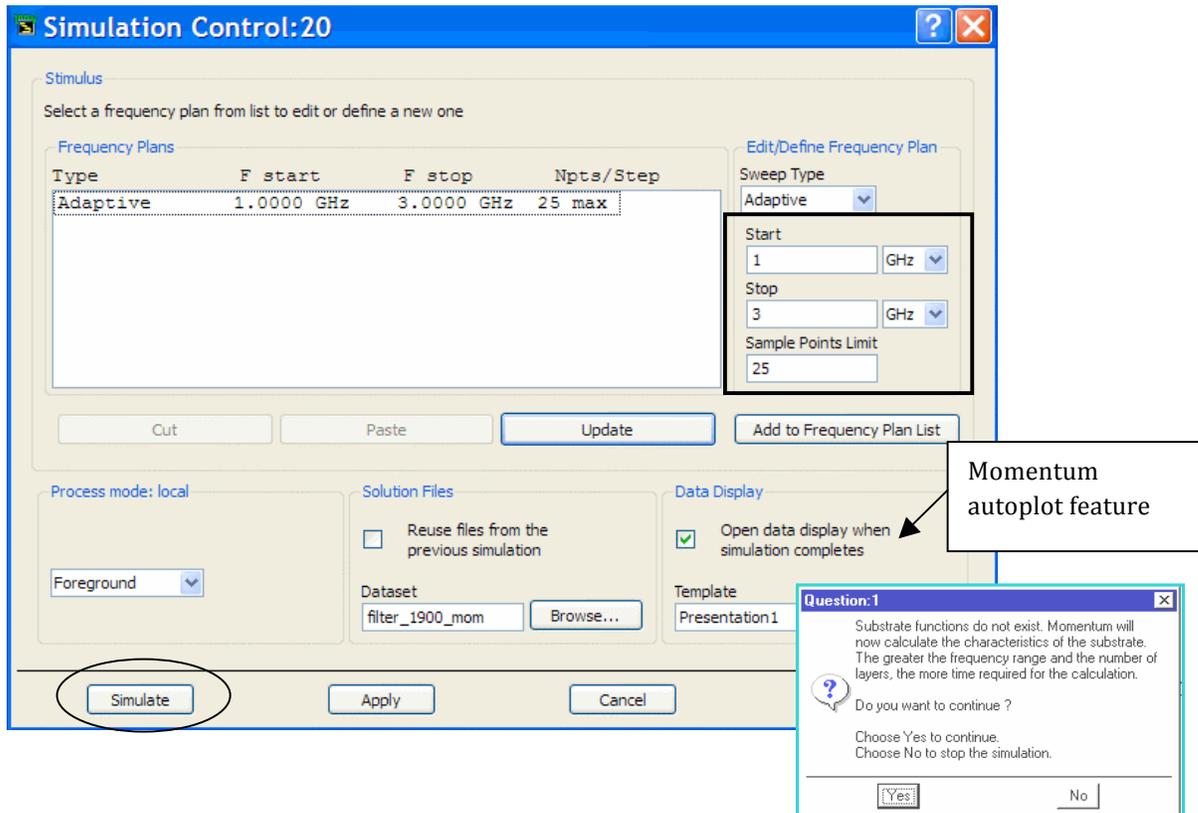
j. When the ADS Data Display opens, you will see a message
    message about the autoplot: check the box as shown here
    here and click **OK** to dismiss it. Notice that the Momentum
    Momentum autoplot template looks slightly different than
    than ADS.

k. **Zoom** in to the S21 plot. As shown here, the expected response is similar
    to the response that you would get from the circuit simulator with any
    small difference due to the solution methods. Also, **zoom** in and look at
    the mesh pattern.
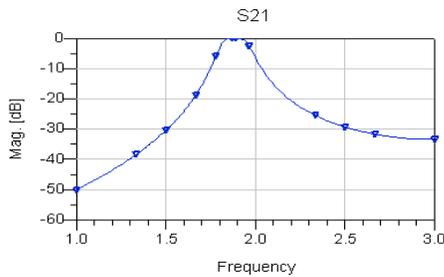
Zoom in to
see the
mesh cells.
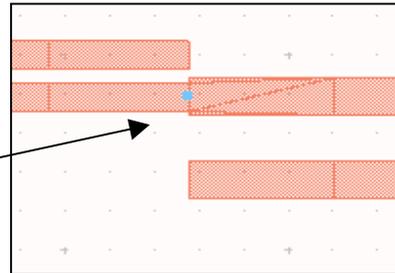
l. Draw a simple crude rectangle (icon shown here) along the length of the
    filter. It should be about 2000 mils long and 60 or 70 mils wide. When
    you draw the rectangle, notice that the cursor begins drawing at x-y 0,0 as
    shown here. Also, you can measure an object, using the cursor, by
    clicking in layout and watching the values change here. The point of this
    step is to show how Momentum can analyze the coupling between the
    filter and the rectangle (metal).

Insert Rectangle

Spacing between the filter and the
rectangle is about 10 mils on this end.

W: 60 -
70 mils

L : about 2000 mils or slightly longer then the filter

0.0000, 0.0000   mil A/RF SimSchem
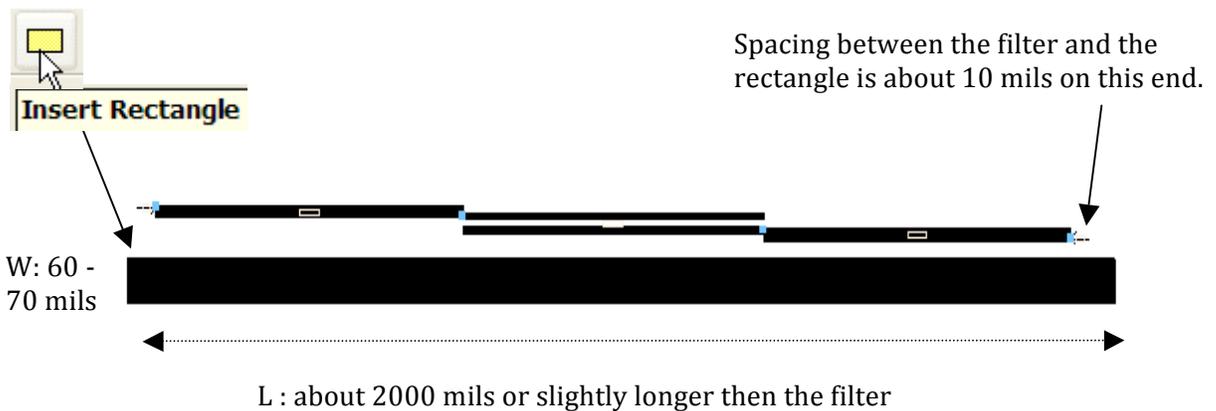
Click, then move the cursor and look at
the window bottom or, to measure an
object, use: **Insert > Measure** and click
on two points.

6-15

n. With the rectangle drawn next to the layout, perform another Momentum simulation. Click: **Momentum > Simulation > S-parameters** and when the dialog appears, <u>change the dataset name as shown here</u>, **filter_mom_strip** and click **Apply** and **Simulate**.

o. This simulation will take a little longer because there are now more unknowns (cells) to calculate and therefore more computation time is required. When the data display appears, you should see that there is a resonance somewhere near the bandpass or its edges, depending upon your rectangle. This is the type of simulation that can only be accomplished with Momentum. View the S-21 results and then **close** the layout and data display windows. It is not necessary to save this design.

EXAMPLE: Narrow resonance at 2.05 GHz due to coupling effects of rectangle.
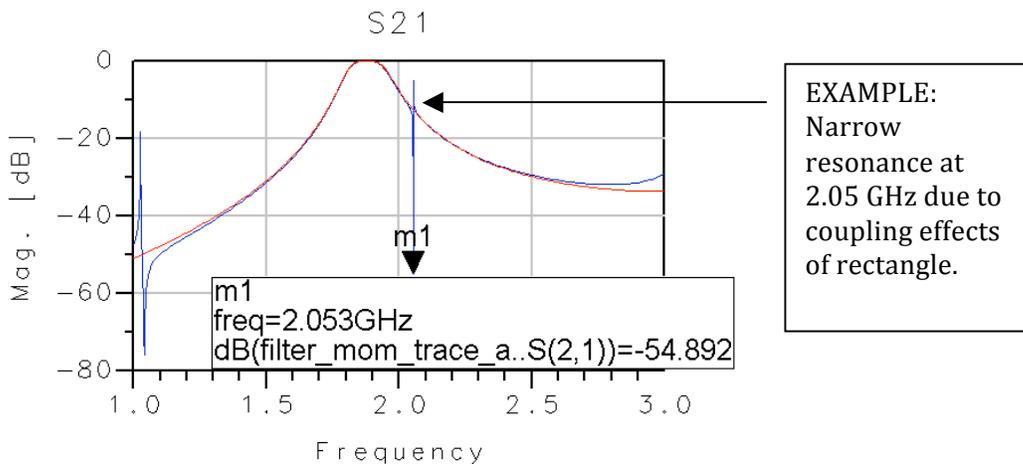
p. You now have two filters that can be used later for the RF system. They are: 1) **filter_lpf** - this is the 200 MHz low pass created first, and 2) **filter_1900** - you simulated this microstrip coupled line filter using the Transient simulator and Momentum. At this time, **close** and **save** all designs and windows.

6. OPTIONAL - Impedance response Data Access Component.

A DAC component is a file-based component. It may contain various formats of data from measurements, listed data, or any other valid file type that ADS can read. In this step, you will create a simple file with complex impedance values that vary over a frequency range. Keep in mind that the DAC can be used to create such models as: frequency sensitive impedances, varactors, step functions, bit sequences for sources, time domain sequences, and many other uses where a file is more efficient than typing in long equations or lists on your schematic. In general, the DAC works like this: it is a component that points to a file in the data directory. In this example you create a file that will be used for the impedance parameter of an ADS component - the simulator will simply read the file.

   a. Open a new schematic with the name: **Z_DAC**.

   b. Refer to the schematic shown here. Insert a termination with ground, an equation base linear Z1P_Eqn (Z1port) from the Eqn Based Linear palette. Then insert an S-parameter controller, and a DAC from the Data items palette in their default states (no setting yet).

   c. Set the S-parameter simulation as if it was the LPF: 10 MHz to 200 MHz in 10 MHz steps as shown and save the schematic again.
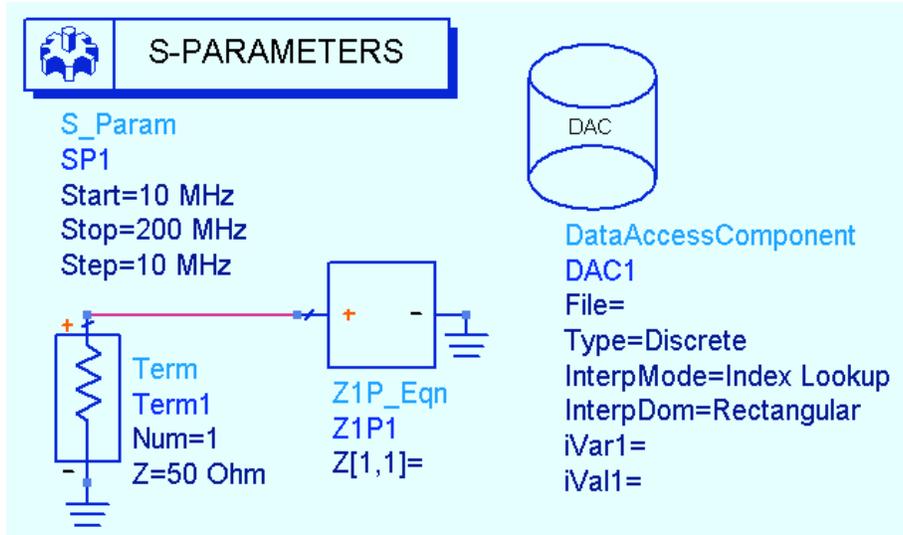
   d. Open the Windows **Notepad** program in **Start >
       > Programs > Accessories**. Or use the ADS Main
       Main Window (Tools >Text Editor) if Notepad is the
       Notepad is the default text editor (this can be
       changed).
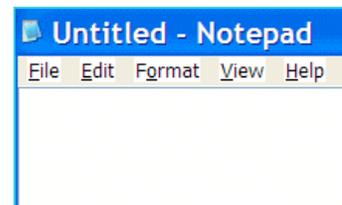
IMPORTANT NOTE on DAC text files - You should **not** use a program that has formatting like MSWORD. However, Notepad is OK.

e. Write an **mdf** (multi-dimensional data file) file shown here and save it in the DATA directory as: **z_dac.mdf**.
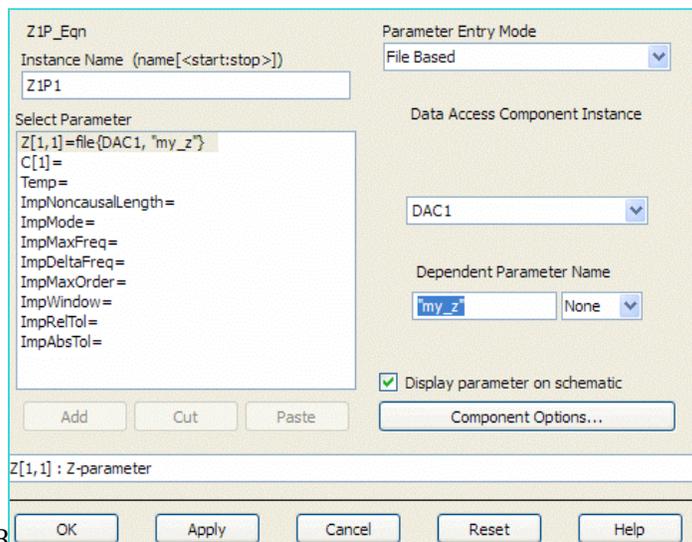
NOTE on file type .mdf - If necessary, use windows windows file explorer to change the name if it is is saved as a .txt file. Also, the syntax in the first first column is 4 frequency points, the second and and third columns contain the real and imaginary imaginary impedance values at each frequency. frequency.

```
z_dac.mdf - Notepad
File  Edit  Format  View  Help
BEGIN my_DATA
% my_freq(real) my_z(complex)
  10e6 10   20
  10e7 10   40
  12e7 200 60
  20e7 400 200
END
```

f. On schematic, edit the S-parameter controller. In **Parameters tab**, set to compute **Z parameters** not S. In the **Display** tab, check Sweep Var, Start, Stop and set them as shown to sweep **freq** from 10 to 200 MHz in 10 MHz steps. You will get interpolated data for all the steps.

```
S_Param
SP1
SweepVar="freq"
Start=10 MHz
Stop=200 MHz
Step=10 MHz
CalcZ=yes
```

g. On the schematic, edit (double click) the **Z1P_Eqn** and its edit dialog will appear as shown here.

h. As shown here, select **File Based** as the Parameter Entry Mode, and **DAC1**, and set the Dependent Parameter Name to **my_z** and click OK.

```
Z1P_Eqn                              Parameter Entry Mode
Instance Name  (name[<start:stop>])   File Based
Z1P1
                                      Data Access Component Instance
Select Parameter
Z[1,1]=file{DAC1, "my_z"}
C[1]=
Temp=
ImpNoncausalLength=                   DAC1
ImpMode=
ImpMaxFreq=
ImpDeltaFreq=                         Dependent Parameter Name
ImpMaxOrder=
ImpWindow=                            "my_z"        None
ImpRelTol=
ImpAbsTol=

                                      ✔ Display parameter on schematic
    Add        Cut        Paste           Component Options...

Z[1,1] : Z-parameter

    OK        Apply      Cancel       Reset        Help
```

i. **Edit the DAC** as shown: Type in the file name, select the Type, etc. Also, *iVar1*

DataAccessComponent
DAC1
File="z_dac.mdf"
Type=Generalized Multi-dimensional Data
InterpMode=Linear
InterpDom=Rectangular
ExtrapMode=Interpolation Mode
iVar1="my_freq"
iVal1=freq

Generic MDIF
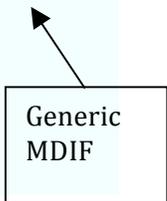
(my_freq) is the independent variable name and *iVal1* (freq) is the independent variable value. As "freq" is swept, "my_freq" will be indexed in the file and the DAC will return complex values of "my_z" interpolated over the range.

j. Simulate and plot two traces, real and imag, of Z(1,1) as shown. As you can see, the Zport can be used wherever a frequency sensitive component is required. For multiple components, simply create different files and access them as required.

m1
freq=100.0MHz
imag(Z(1,1))=40.000
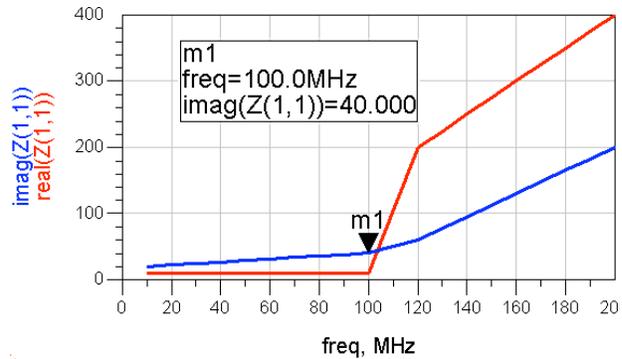
EXTRA EXERCISES:

1. Calculate the bandwidth of the filter in Transient simulation using the rule-of-thumb: BW= 0.35 / rise time. From time zero until about 6 ns should be the rise time. However, use the markers and the equations to do the calculation.

2. Try using the Design Guide for another circuit type or some other design.

3. Create a DAC for a frequency sensitive inductor and simulate it in a simple CLC filter. The DAC will contain the inductor values at each frequency.