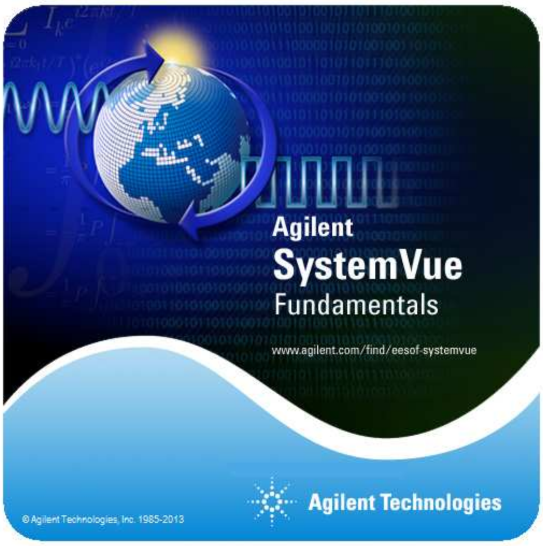
Student Course Workbook

Labs

SystemVue

SystemVue Fundamentals



COURSE: N3248A(1) – 1 Day

August 2013

Agilent EEsof EDA - Customer Education

Santa Rosa, California USA

Part Number E8900-_____

Agilent Technologies

Restricted Rights Legend

Agilent Technologies, USA

Agilent Document Part Number E8900-_______

Version 1.2 – August 2013

SystemVue 2013.08

Warranty

Acknowledgments

UNIX is a registered trademark of UNIX System Laboratories Inc. in the U. S. A. and other countries.

Some of the system and circuit simulators that are used in the Advanced Design System are based on the Spectre program and the Ptolemy software environment, and on work conducted at the Department of Electrical Engineering and Computer Sciences at the University of California - Berkeley.

SystemVue Fundamentals

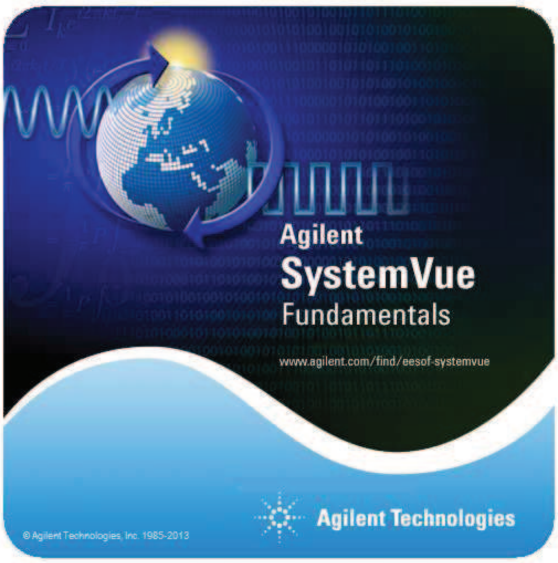
Contents: Slides and Lab Exercises

LAB EXERCISE 1

SystemVue Basics

This lab exercise introduces the basic use model of SystemVue, including using the built-in templates, running analyses and tuning variables. It also covers some details about plotting data and using markers and sliders.

There are no prerequisites for this lab exercise.



Agilent Technologies

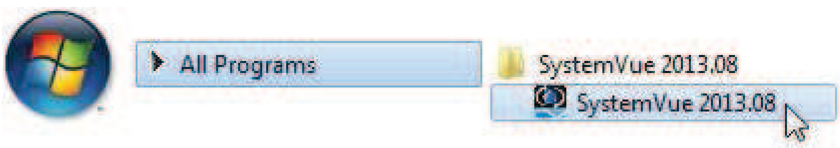
Table of Contents: Lab 1

Lab 1: SystemVue Basics

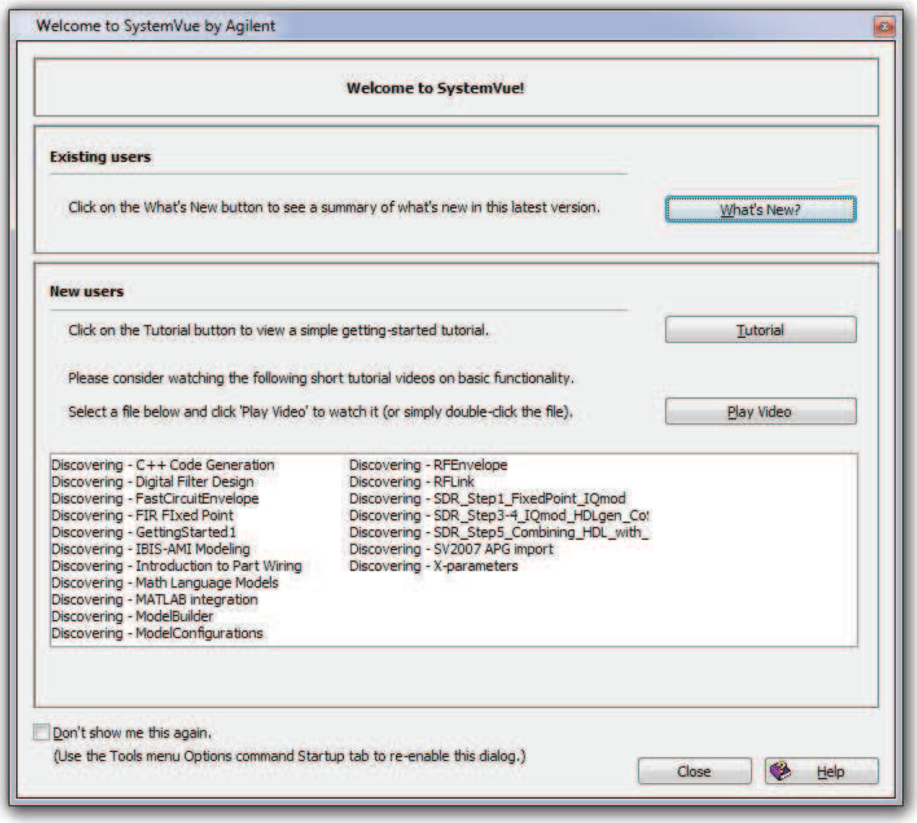
IMPORTANT: This lab assumes **SystemVue 2013.08** is installed on your computer or you have access to it. If you are in a classroom, ask the instructor if you have any questions. This lab uses version 2013.08 and should work for most versions afterward.

1. Start SystemVue on your computer

a. Click the shortcut icon for SystemVue as it appears on your desktop, or use the Start > Programs command to find and start SystemVue as shown here:

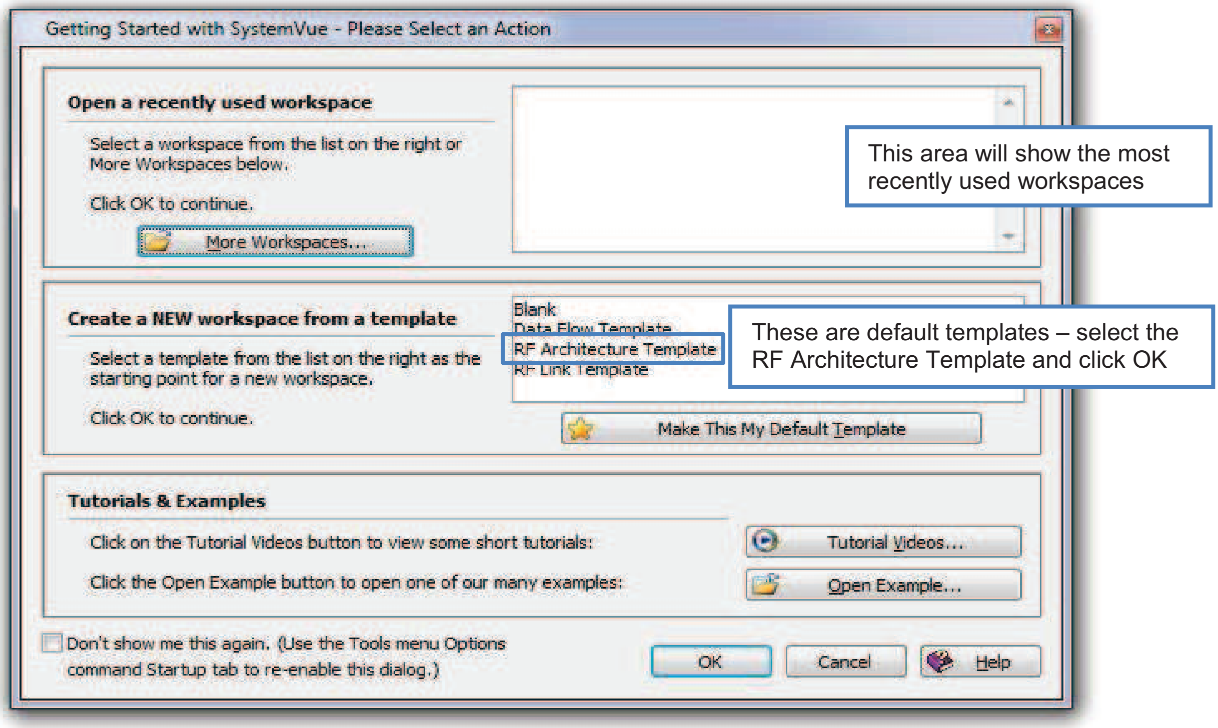


b. After initialization, you will see the Welcome dialog for SystemVue – this always appears unless you check the box at the bottom. For this lab, click the **Close** button at the bottom. After this course, you can view the videos and check for what's new.



2. Data Flow and RF Architecture Templates

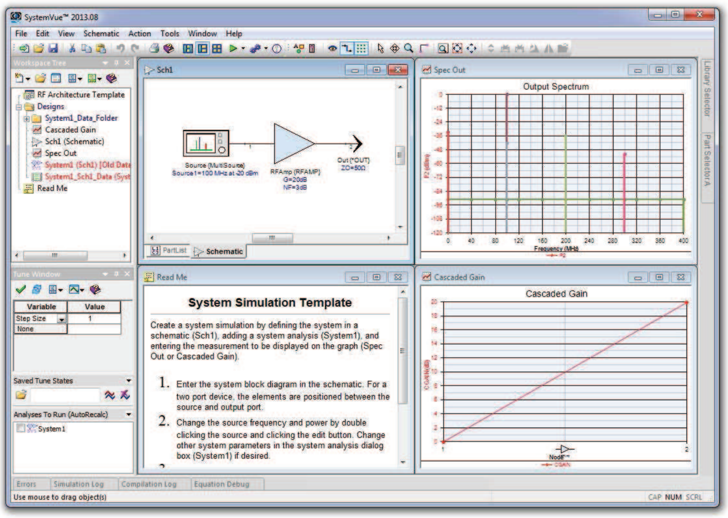
a. Now, you should see the Getting Started dialog (or Start Page) which gives you access to recent workspaces, templates, example designs and videos. **Select** the **RF Architecture Template** and click **OK**.



b. Now you should see the RF template. It has a simple amp and source with the analysis results plotted. It also has a note about how to use it. You can experiment with it for a moment but do not save it.

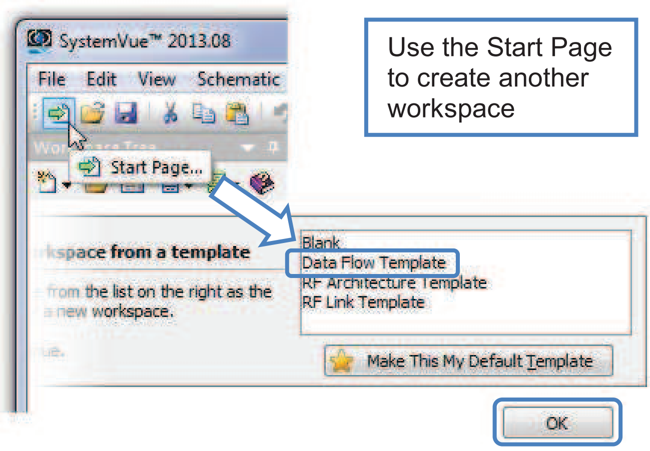
***NOTE**: You will learn how to create schematics, analyses, and graphs as this class proceeds.*

For example, schematic items are inserted using Hot Keys or the Part Selector and libraries as you will soon see.



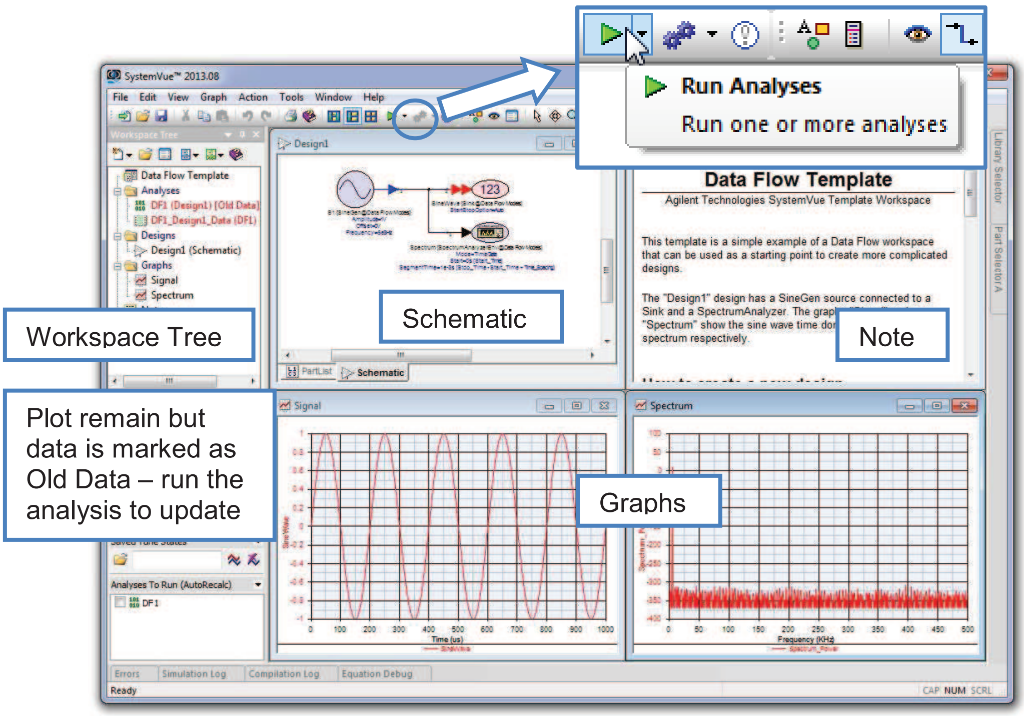
c. When you are ready to continue, **click the Start Page icon** at the top left of your SystemVue window. This will bring you back to the Getting Started dialog – also called the Start Page. Then **select** the **Data Flow Template** and click **OK** at the bottom as shown here.

The Start Page icon makes it easy to create new workspaces or access others.



d. Examine the Data Flow template. Notice the Workspace shows several icons for the schematic, data, analysis and note. Although the plots have data, you must run the analysis again to update the results. Click the **Run Analysis icon** shown here and watch what happens – the red description goes away.

When the analysis is complete the red letter (old data) disappears. Next, you will use these plots and the design.

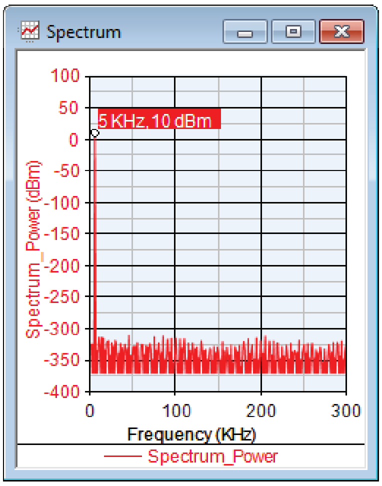


3. Graph Properties

a. Look at the Spectrum plot. Then **click** on the spectral **peak** and notice that a maker is added as shown here.

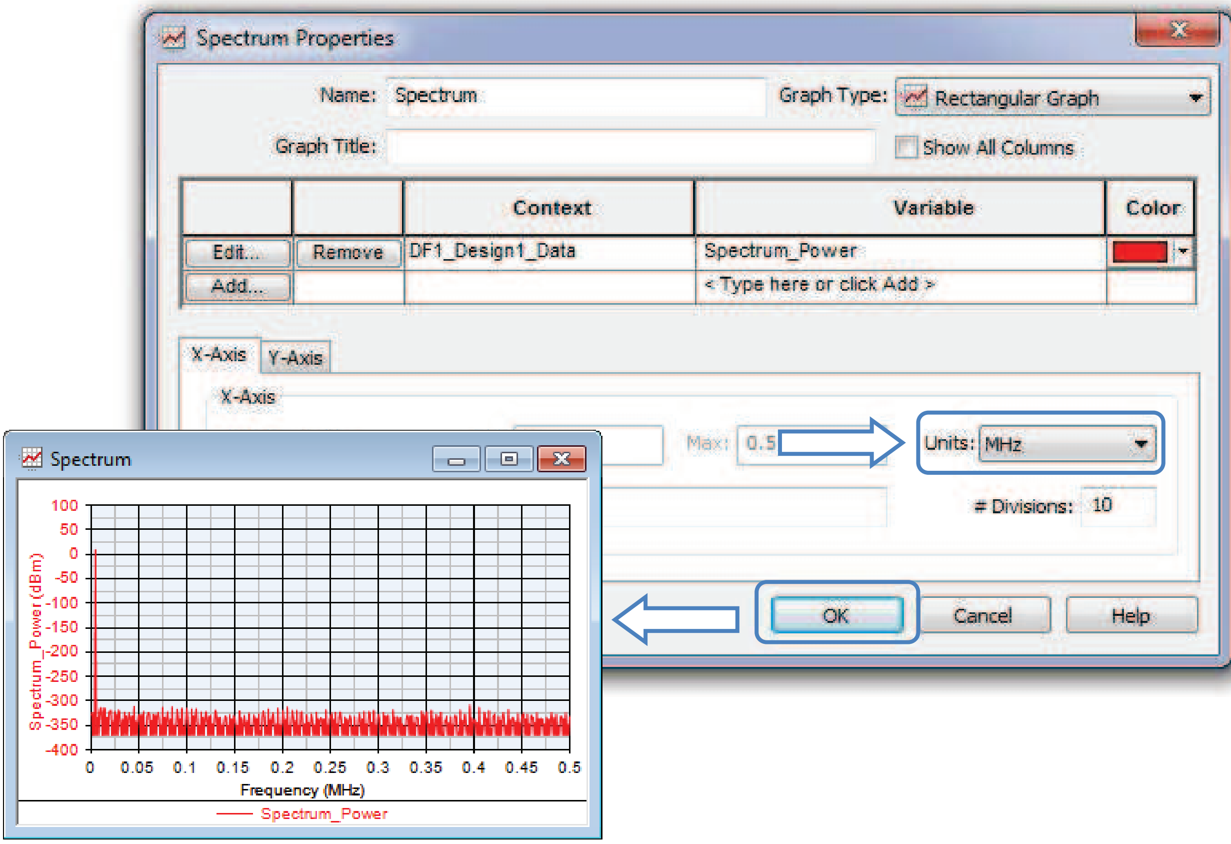
b. With the marker still highlighted, press the keyboard **Delete** key and the marker will be deleted. Then add the marker back again. Now you know how to add and remove markers.

The default X-Axis is in Hz but you will change this next.



c. Double-click in a blank area of the plot and the Spectrum properties dialog will appear.

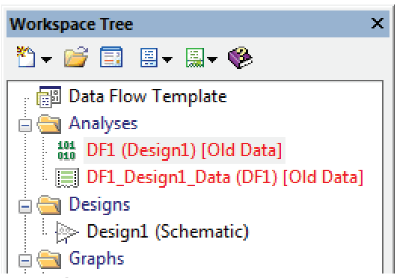
d. As shown here, type in a title for the plot: **Baseband Spectrum**. Also, change the X-Axis units to **MHz** as shown here and click **OK.** Now you can see the mathematical noise floor in MHz around -325 dBm.

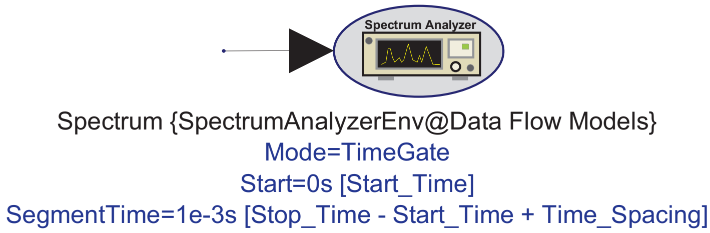


4. Editing Schematic Sink Properties

a. In the schematic, click the icon to maximize the window.

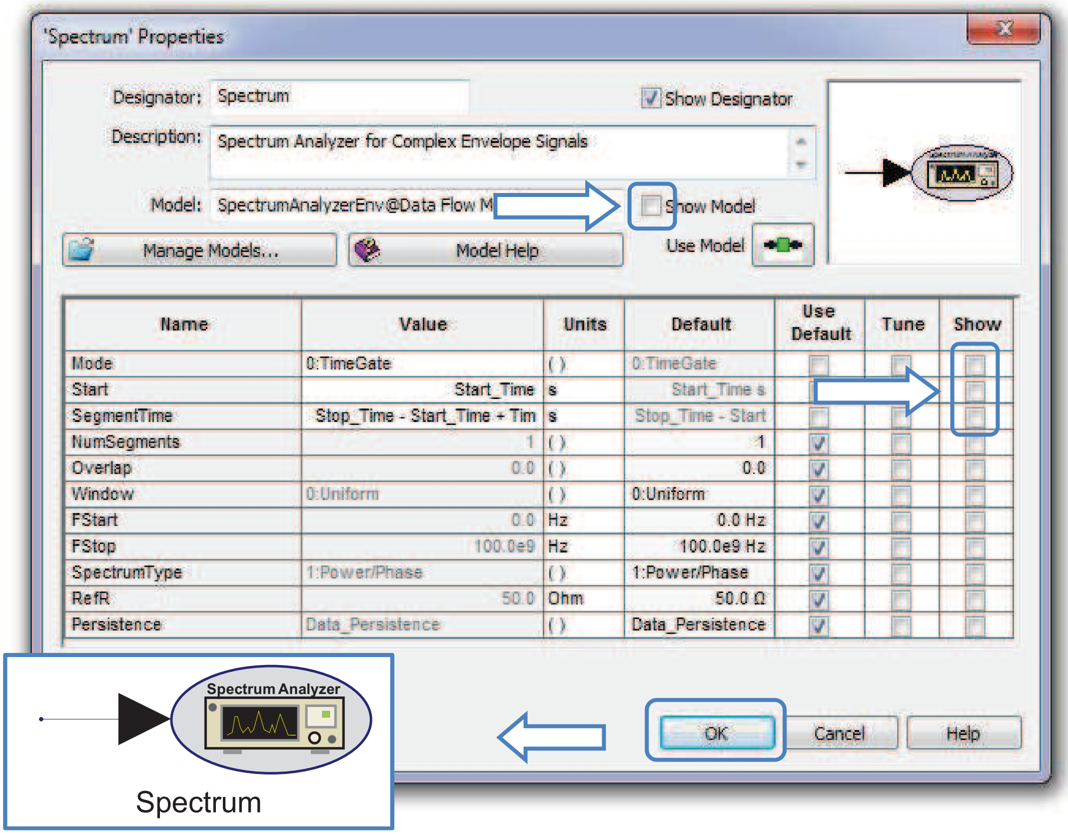
b. Double-click the Spectrum Analyzer sink. Notice the Workspace tree shows the red Old Data – this happens whenever the system recognizes that something is about to change, and the data might be out of date – this



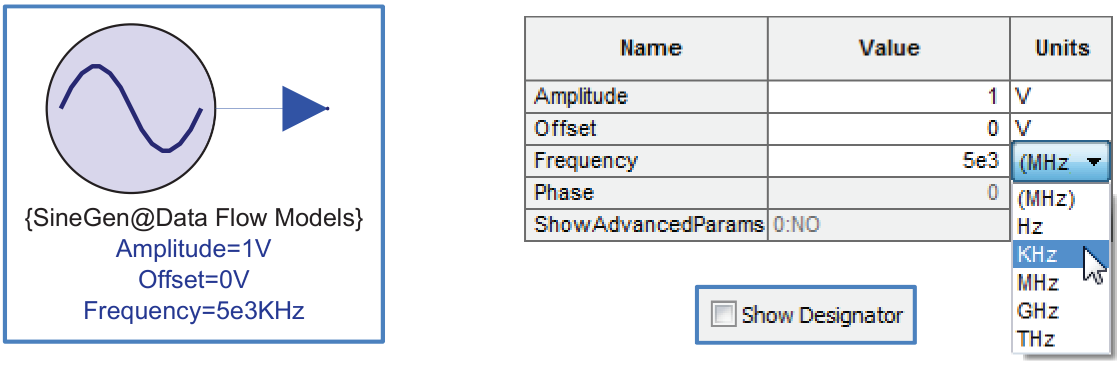


is OK.

c. In the Spectrum Properties dialog, uncheck the box to Show Model and also uncheck the boxes for the three time parameters as shown here. This will clean up the annotation on the Spectrum component in schematic.

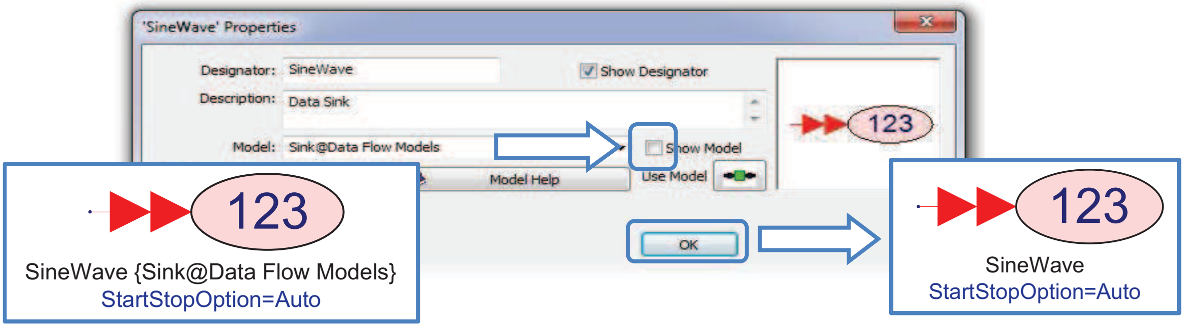


d. Edit (double-click) the **Sine** wave generator as shown here. Change the frequency from 5e3 Hz to **5 KHz** as shown here. This is a reminder that you always need to delete the exponent form (e) if you use change the Units, such as KHz, so that the values do not multiply. Do not click OK yet.

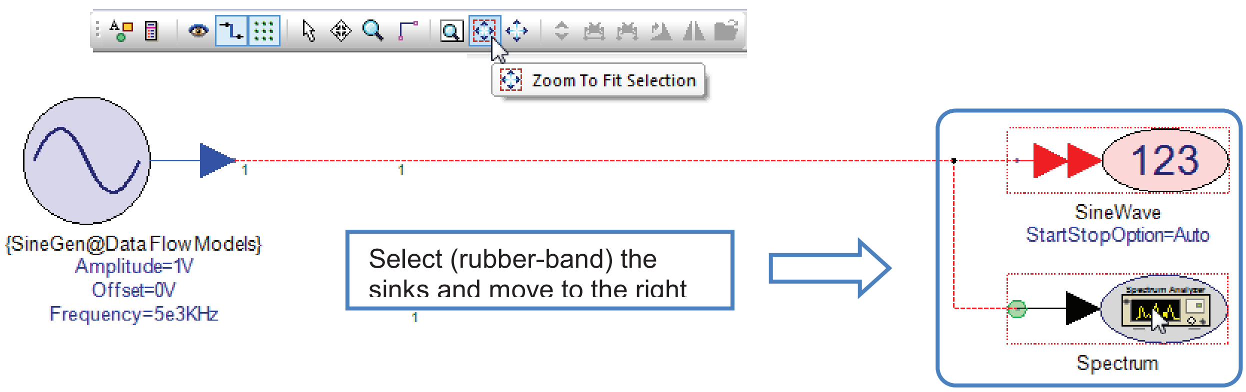


e. Uncheck the box to show the **S1** designator and then click **OK**.

f. Edit the **SineWave** sink and **uncheck** the box to Show Model and click **OK**. Removing unwanted text makes it easier to read.



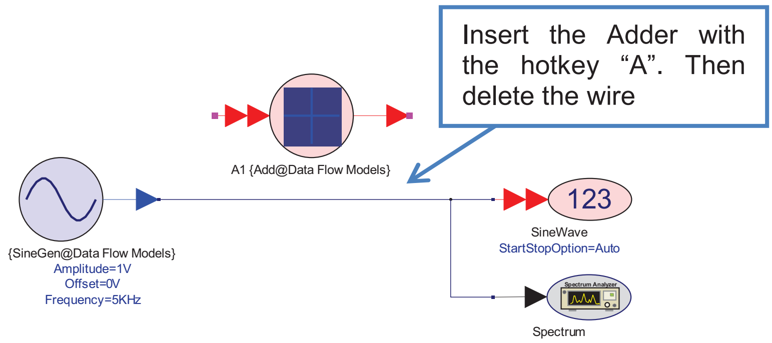
g. Click the **zoom** icon or use the keyboard shortcut ***z*** to Fit to and then select the sinks and move them to the right to make room for more components. Next, you will add noise.



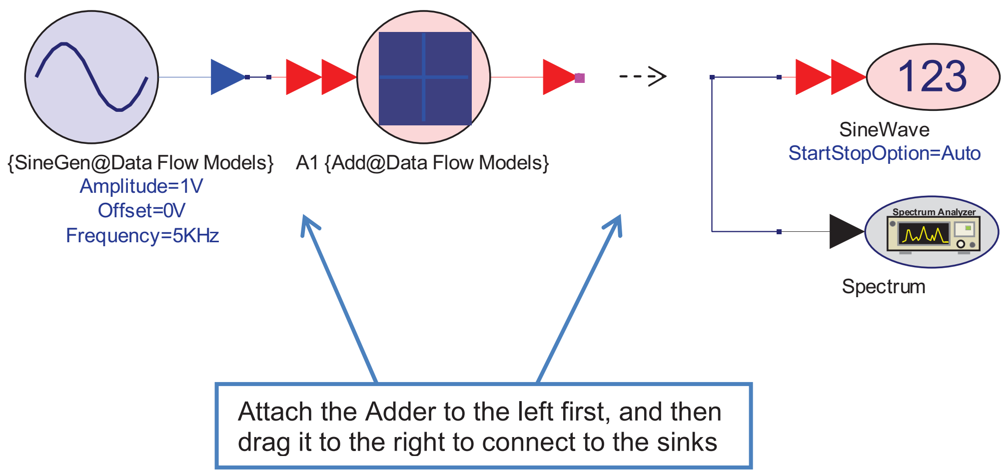
5. Inserting Components: Hot Key and Part Selector

a. Press the **A** key on your keyboard and then **click in a blank area of the schematic** and the Adder will be inserted as shown here. If not, click in the blank area first and then repeat. You will get used to using Hot Keys.

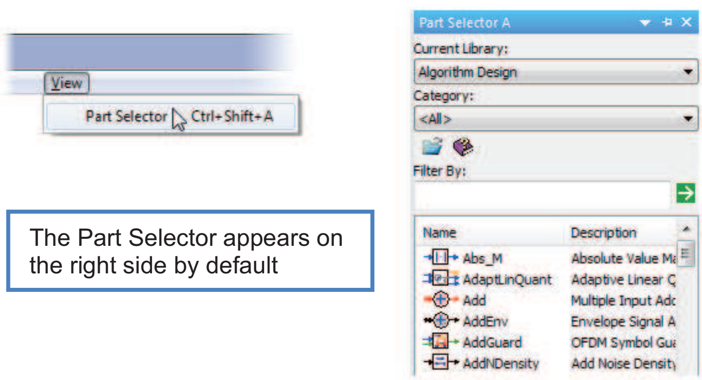
b. After you insert the adder, delete the wire between the source and sinks: select the wire and then use your keyboard delete key.



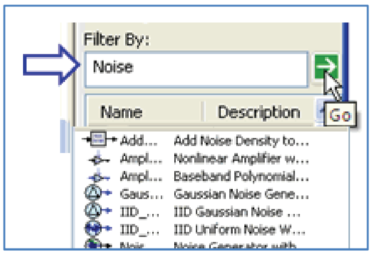
c. **Connect the Adder** as shown here – notice how the pins and wires change color when connected. The Adder is required for the next steps.



d. Select the command: **View > Part Selector** and it will appear in your window. Notice that the Algorithm Design library appears by default. Next, you will be inserting components.

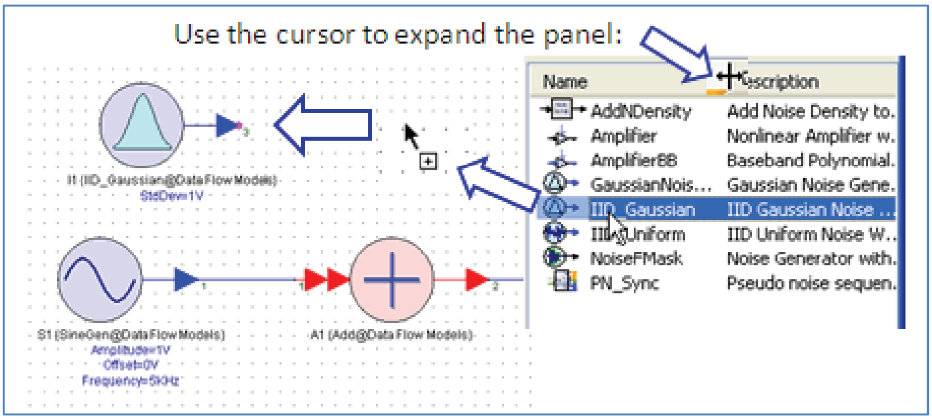


e. Type the word **Noise** in the Filter By field and click the green arrow **Go**. Immediately, all the noise components from this library appear.

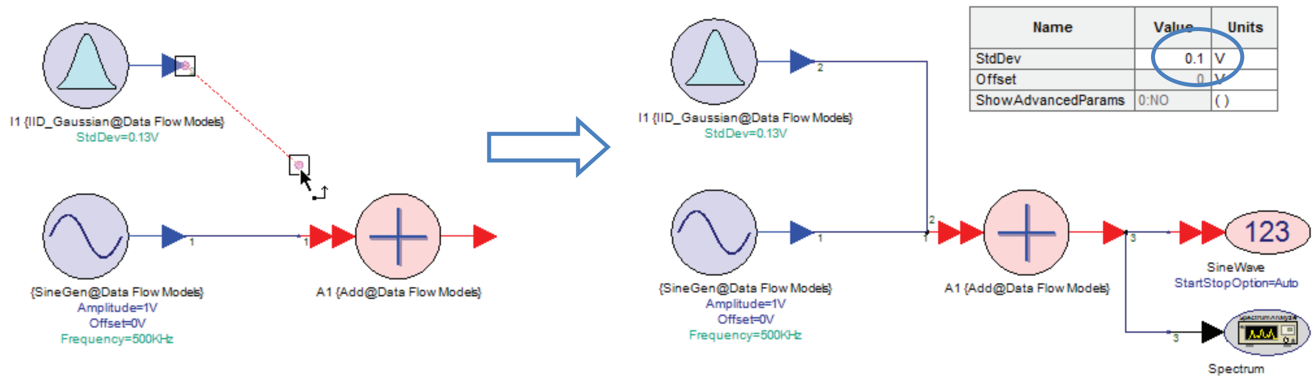


f. Place your cursor in between the Name and Description panel as shown here and move the panel so you can see the complete Name or Description.

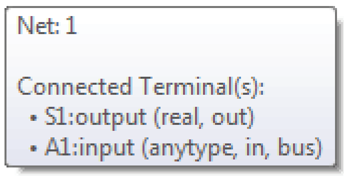
g. Select the **IDD Gaussian Noise** component and drag it onto the schematic. Notice the cursor shows the + sign as you drag it and then click to insert it.



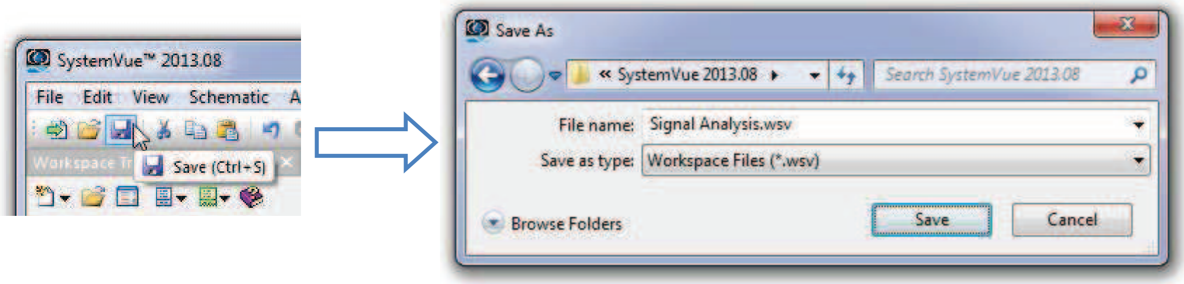
h. Click the **wire icon** (connection line) to connect the IDD noise directly to the input pin (not the wire) of the Adder. Next, edit the IDD noise component and change the StdDev to 0.1 V.



NOTE: The wires and pins have different net or ID names. You can move your cursor over the wire and the Net information will appear as shown here.

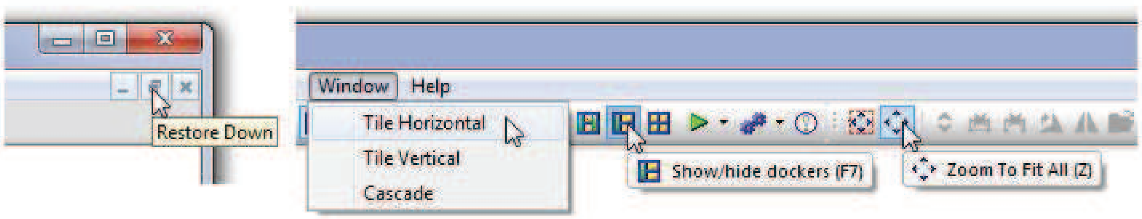


i. Click on the Save As icon and save the workspace as: Signal Analysis.



6. Windows, Dockers and Views

a. To restore the previous view, click the corner **Restore View** button.

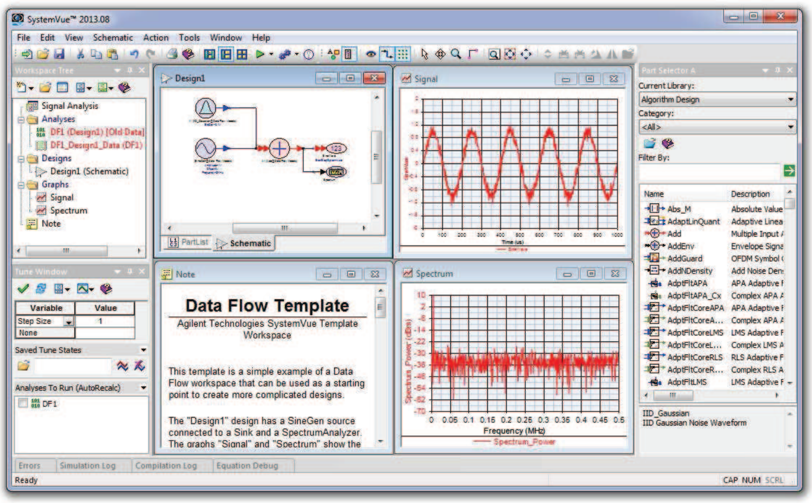


b. Use the commands **Window > Tile Horizontal** or Tile Vertical – you will use these often after maximizing different windows.

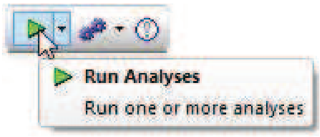
c. Use the **Show/hide dockers** icon and the Part Selector (docker) will appear or disappear.

d. For individual windows, use the Zoom to Fit All icon to see all the contents of the window (such as the schematic).

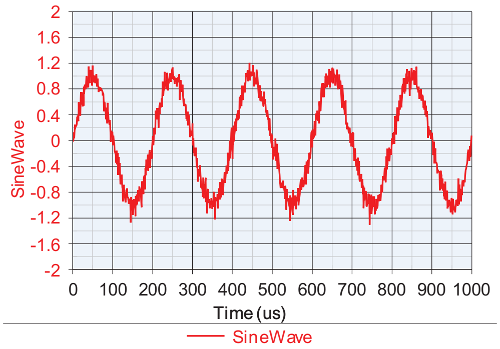
Next, you will run the analysis.

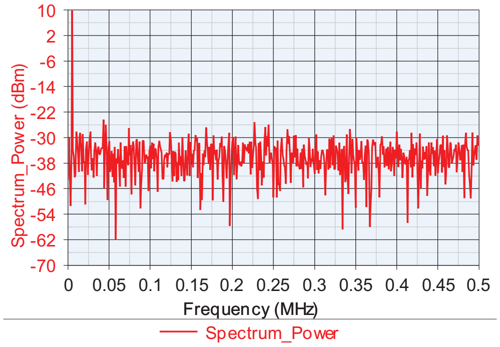


7. Analysis and Results



a. Click the **Run Analyses** icon and observe the results with random noise added to the signal. As expected, the noise appears on the traces.

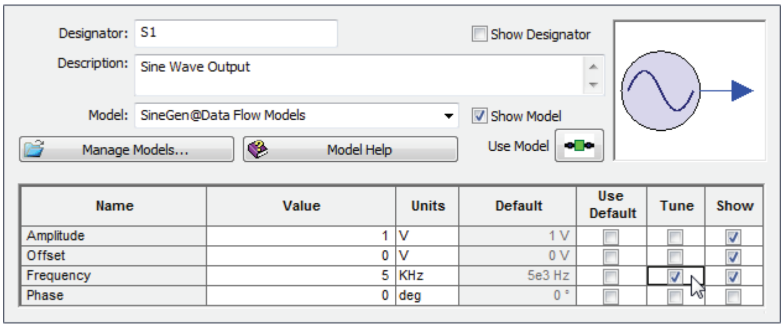


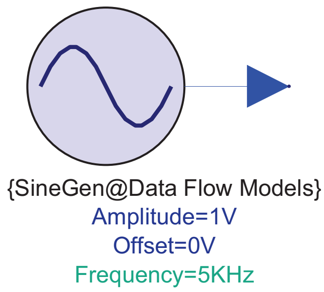


In the next steps you will learn how to tuning works in SystemVue.

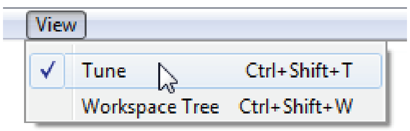
8. Tuning Source Frequency

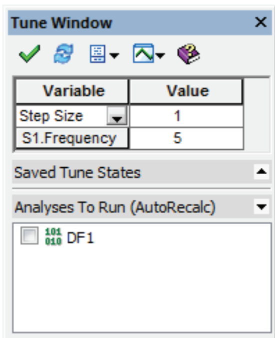
a. Edit (double-click) the signal generator (S1). In the Properties dialog, check the box for the **Frequency** parameter as shown here – click **OK**. Notice the schematic symbol now shows Frequency in green (tunable).



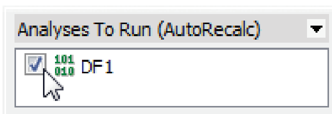


b. Click the command: **View > Tune** and the Tune Window will appear (bottom left) showing the S1 Frequency variable, the default step size and the analysis. The Analysis was already set up with the template.

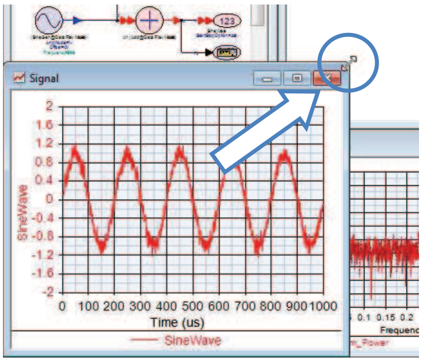




c. In the **Tune Window**, **check the box** to have the DF1 Analysis perform an **AutoRecalculation** as shown here. This will allows the analysis to run every time you change the tune value – as you will soon see.

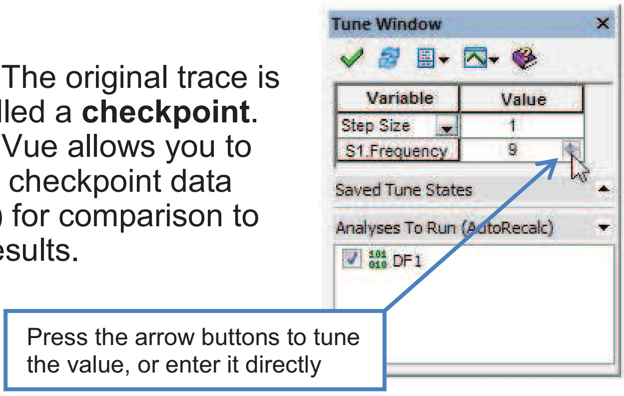


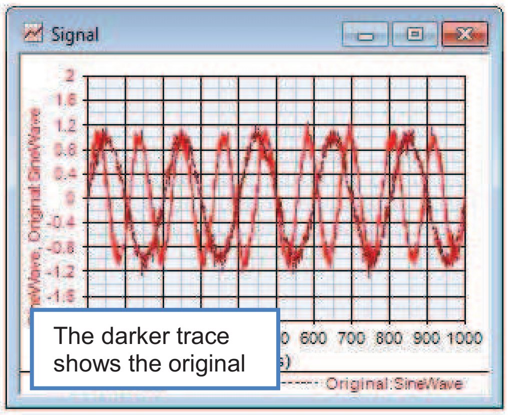
d. Be sure you can see the signal plot. Try dragging the corner of the plot to make it a little bigger – you can resize it later.



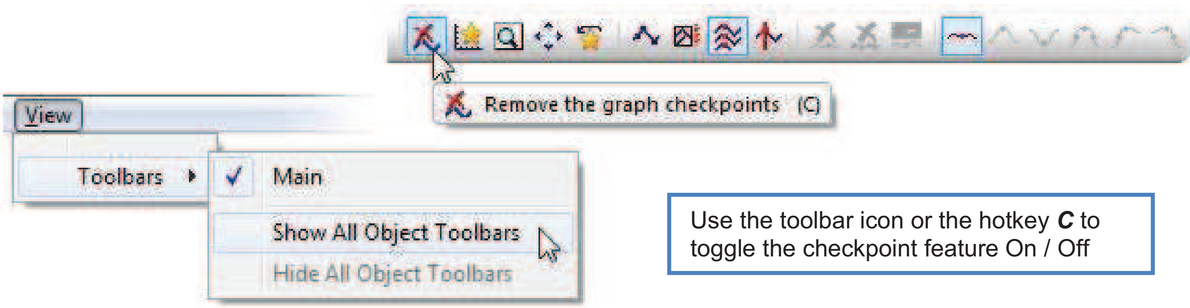
e. In the Tune Window, slowly press the up arrow button for the Value of S1 Frequency, increasing it to about 9 or more to see the increased frequency trace. You will also see two Y- axis labels: SineWave, Original SineWave. The light red trace is the tuned value and the darker red is the original.

NOTE: The original trace is also called a **checkpoint**. SystemVue allows you to save or checkpoint data (traces) for comparison to other results.



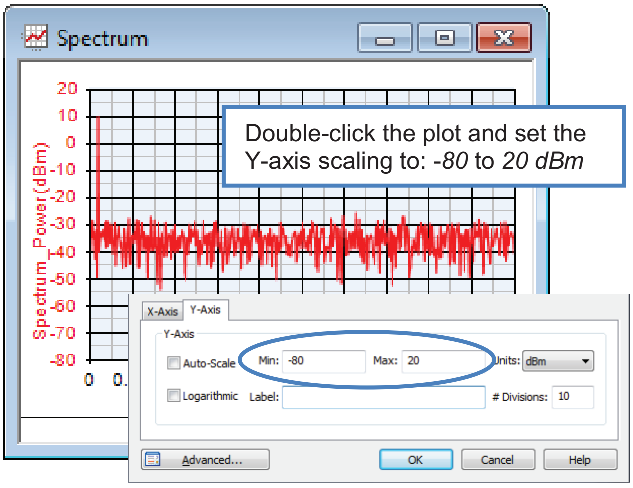


f. To remove the original 5 KHz trace, click **View > Toolbars** and make sure **Show All Object Toolbars** is turned on. Click the icon shown here: Remove the graph checkpoints. Notice the only new tuned trace appears. Now, select the plot (activate it) and press your keyboard C key this will make the current trace a checkpoint. Try tuning again and using the C hot-key or the icon to get used to using this checkpoint feature. Spend a few minutes on this and then move on.

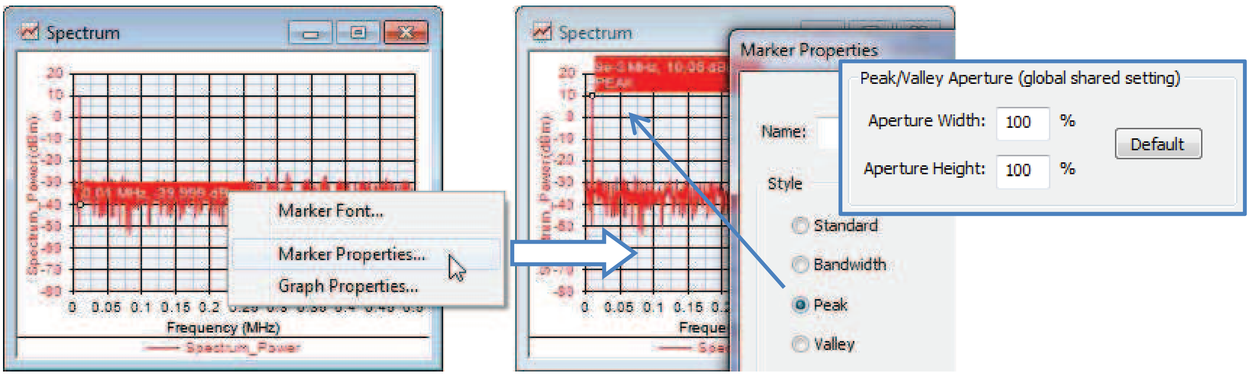


9. Plot and Marker Properties

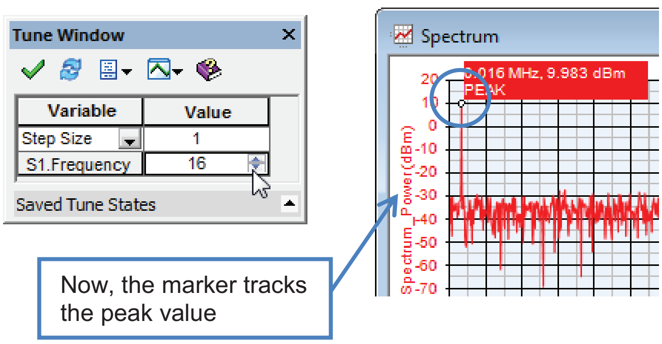
a. If needed, use **Window > Tile Horizontal** and then **edit** (double-click) the Baseband Spectrum plot. In the Y-axis tab, uncheck the box for Auto-Scale and set the scaling as shown: -80 to 20 dBm - click **OK**. This is because the peak is probably at the top of the scale.



b. Notice that your marker is probably no longer on the peak after tuning. To set the marker to stay on the peak value, **right-click the marker readout** and select **Marker Properties** as shown here. Select **Peak** and set the marker search range **Aperture Width** and **Height** to **100 %** and click **OK**.



c. Try changing the Frequency value (use the tuner button) again and notice that the marker now stars on the peak.

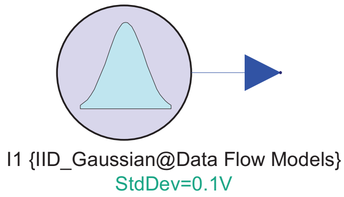


Next, you will use the slider to tune the noise standard deviation value.

10. Tuning with Sliders

a. Edit the Gaussian **Noise** component and make the **StdDev** variable **tunable** and visible as shown here – notice it appears in green on the

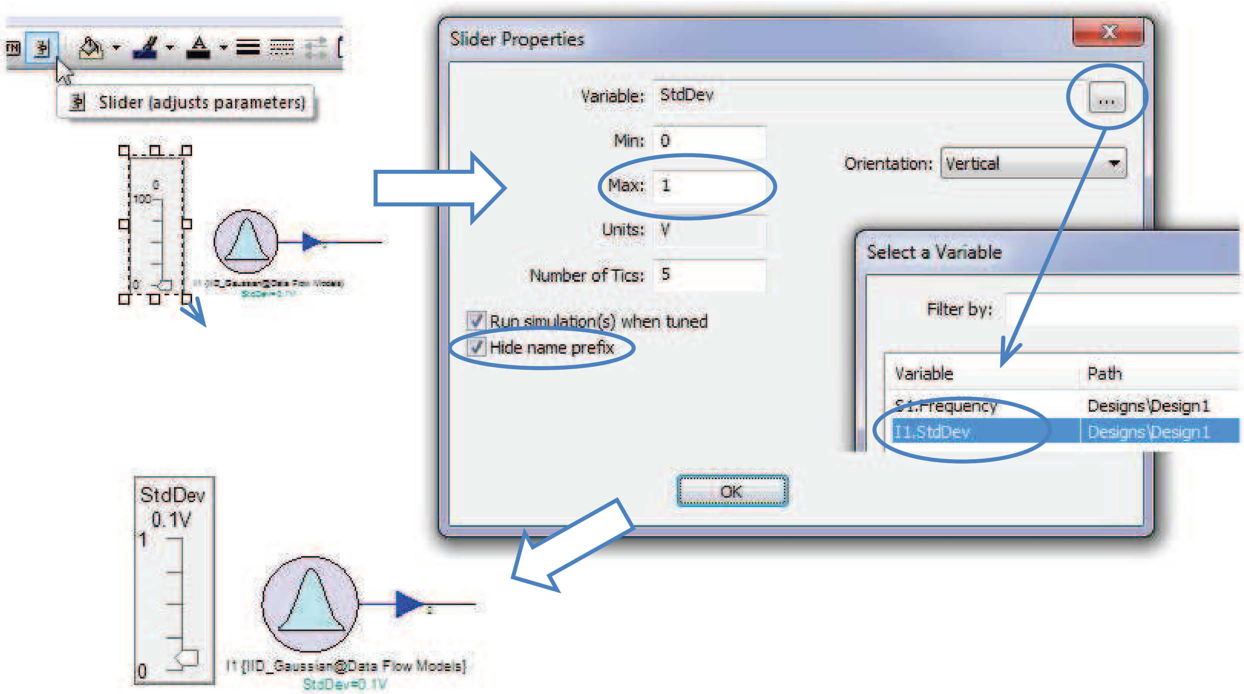
Name	Value	Units	Default	Use Default	Tune	Show
StdDev	0.1	V	1 V	☐	☑	☑
Offset	0	V	0 V	☑	☐	☐
ShowAdvancedParams	0:NO	( )	0:NO	☑	☐	☐



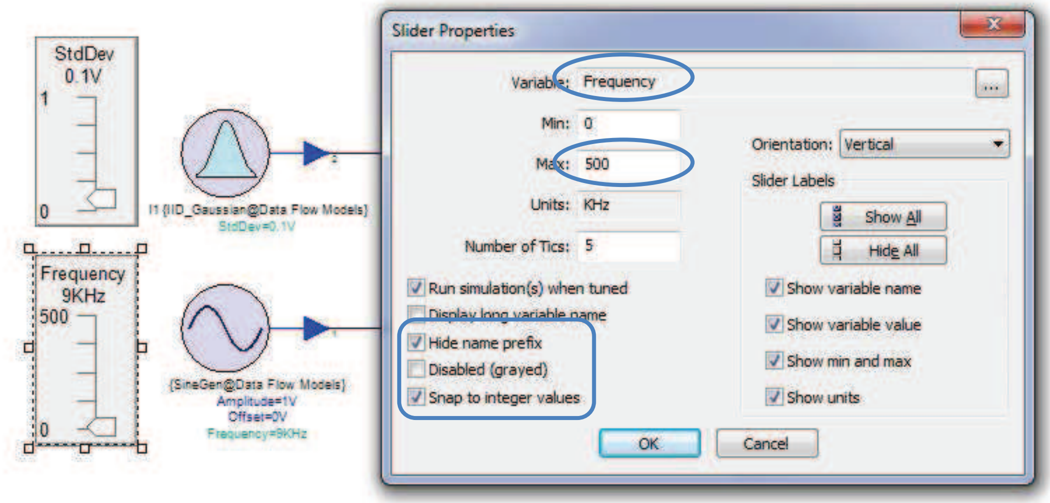
schematic.

b. Look in the upper left corner of the SystemVue window and click on the **Show/hide Annotation Toolbar**. This toolbar has the slider you will use next.

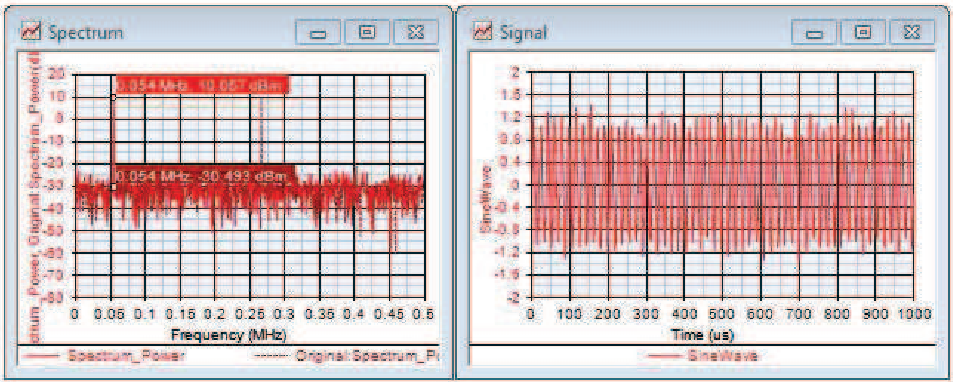
c. Click on the **Slider** (adjusts parameters) icon to activate it. Then click in the schematic (next to the Noise component) and use your left mouse button to drag the Slider to a reasonable size. Then **edit** (double-click) the slider and set the **Max** to **1**, check the box to **Hide name prefix** (this eliminates unwanted text), and select the **I1 StdDev** variable as shown here. Click **OK** and the Slider is ready to use.



d. Now that you know how to insert a tuning Slider, go ahead and **insert another Slider** for the **signal generator frequency**, as you did for the Nose StdDev in the last step. Select Frequency with a **max** of **500**, and check the boxes for **Hide name** and **Snap to integer**. Refer to the step above if needed. Your Frequency slider should look like the one shown here. Your schematic should now have both sliders ready to use.

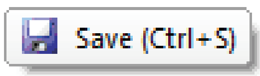


e. Move the slider controls and observe the effects on the plots. You can also try toggling the checkpoint feature on or off as desired.



NOTE: As you move the sliders, the spectral plot may jump around as it autoscales the data. Double-click on the plot and deselect the Y axis Auto-Scale to prevent this if desired.

f. When finished, **save** your workspace using Ctrl + S.
You will use this workspace for the next lab exercise.



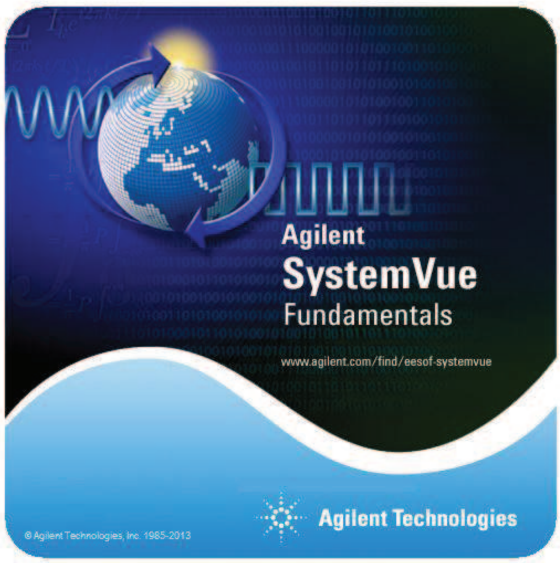
<u>END OF LAB EXERCISE</u>

LAB EXERCISE 2

Envelope Analysis and MathLang Basics

This lab exercise introduces SystemVue MathLang, FM analysis techniques, and data plotting. It also covers useful details for building designs, analyzing data, and avoiding errors from incorrect settings.

Prerequisite: Lab Exercise 1 (workspace is required).



Agilent Technologies

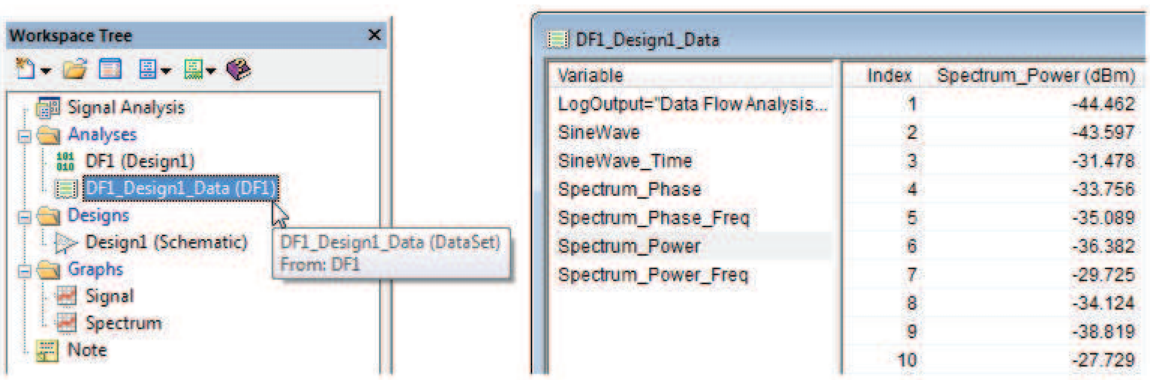
Table of Contents: Lab 2

Lab 2: Envelope Analysis and MathLang Basics

NOTE: This lab requires the previous workspace from Lab 1.

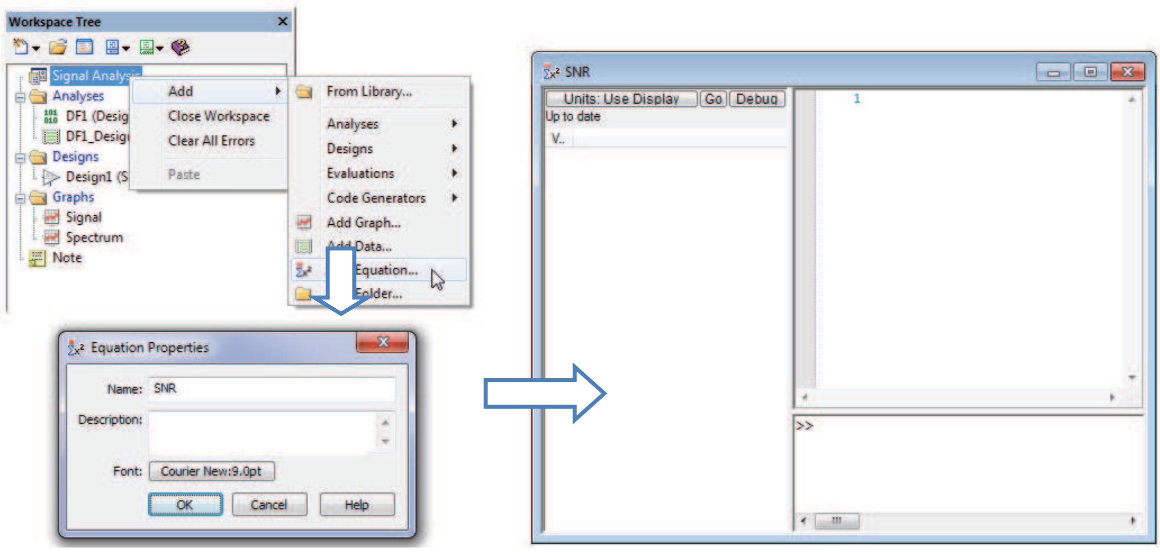
1. Examine the Signal Analysis Dataset

a. Datasets contain all the analysis data. In your Signal Analysis workspace tree, edit (double-click) the **dataset icon** as shown here. For example, click on the Spectrum Power variable and you see the index number (beginning at 1), the frequency and the power in Watts. The purpose of this is to show the dataset contents for Design1. Next, you will use it to write some equations.



2. Add an Equation to the Workspace

a. In the Workspace tree, right-click the workspace name, Signal Analysis, and click: **Add > Equation**. Type in the name **SNR** for Signal to Noise Ratio and click OK. Notice the new window that appears, ready for your equation.

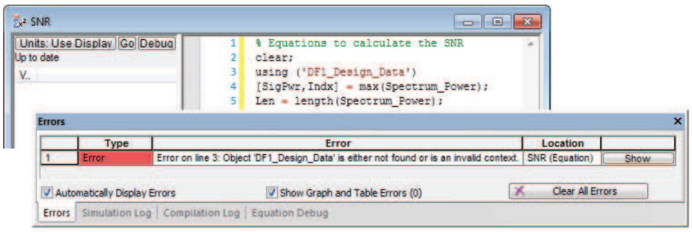


NOTE: *By adding the equation to the top level workspace folder, they are available everywhere. For eample, if you add the equation under the analysis folder, they will calculate correctly, but won't be available in the other folders*

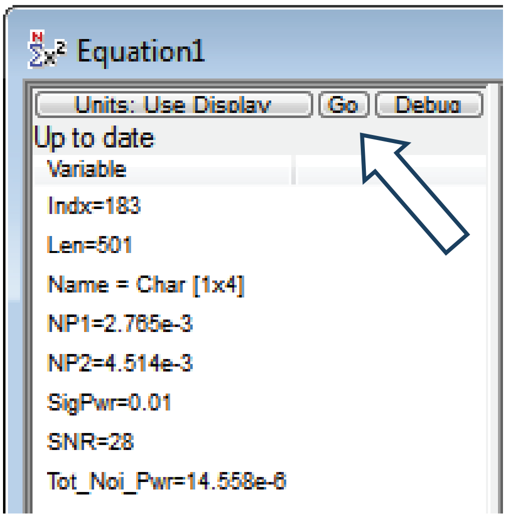
3. Use MathLang for SNR Equation

a. In the SNR equation window, beginning at line 1, type in the following Math Language syntax that is compatible with other math languages – type all 11 lines as shown. When finished, press the **Go** button on the left.

NOTE: *If there are any errors, you will see a message with the line number as shown here.*



Syntax and results for calculating SNR of the baseband spectrum:



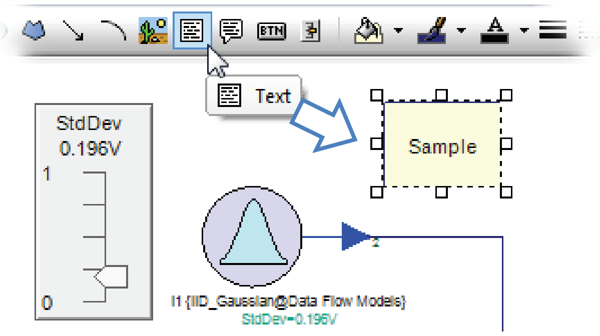

```
1  % Equations to calculate the SNR
2  clear;
3  using ('DF1_Design1_Data')
4  [SigPwr,Indx] = max(Spectrum_Power);
5  Len = length(Spectrum_Power);
6  NP1 = sum (Spectrum_Power(1:Indx-1));
7  NP2 = sum(Spectrum_Power(Indx+1:Len));
8  Tot_Noi_Pwr = (NP1+NP2) / (Len-1);
9  SNR = 10*log10(SigPwr/Tot_Noi_Pwr);
10 Name = 'SNR=';
11 SNR = round(SNR);
```

b. Examine the results. If you have typed the correct syntax, the left slide will show the results, especially the SNR which will be different, depending upon your tuned values.

NOTE on SYNTAX: *The clear command is not required, but does clear out any previously calculated values that appear in the left column. In general, the index value of the spectrum max value is obtained. Then the noise powers (NP1 and NP2) are calculated using the length (all the data points) and the index value. The total power is calculated and then the SNR is calculated using the signal power divided by total noise power. The result is rounded off in dB.*

c. Use the command: **Window > Tile Horizontal** so that you can see all the windows. Also, close the Note window because you are not using it.

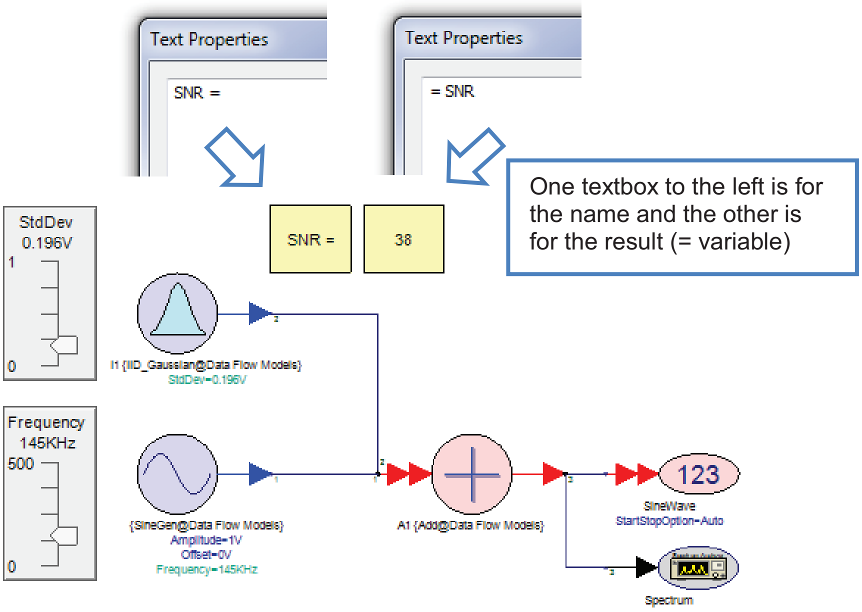
4. Use Text Boxes to View Results



a. From the **Annotation Toolbar**, click on the **Text** icon and then drag **two text boxes** onto the schematic – increase the schematic size if needed.

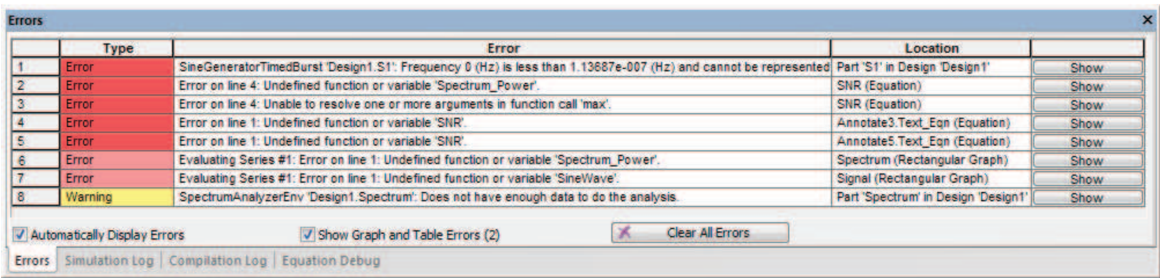
b. Edit the first text box and type in the name of the variable followed by the equal sign: **SNR =**

c. Edit the second text box, but type in the equal sign first, followed by the variable name: **= SNR** Now try tuning the noise to see the results as shown. This technique using text boxes can be use to easily display pass/fail results.



5. Errors and Sampling for Data Collection

a. Move the **Frequency slider** for the signal generator to **0 hertz** and see what happens. Notice the error window pops up showing all errors and warnings.



Errors

	Type	Error	Location	
1	Error	SineGeneratorTimedBurst 'Design1.S1': Frequency 0 (Hz) is less than 1.13687e-007 (Hz) and cannot be represented	Part 'S1' in Design 'Design1'	Show
2	Error	Error on line 4: Undefined function or variable 'Spectrum_Power'.	SNR (Equation)	Show
3	Error	Error on line 4: Unable to resolve one or more arguments in function call 'max'.	SNR (Equation)	Show
4	Error	Error on line 1: Undefined function or variable 'SNR'.	Annotate3.Text_Eqn (Equation)	Show
5	Error	Error on line 1: Undefined function or variable 'SNR'.	Annotate5.Text_Eqn (Equation)	Show
6	Error	Evaluating Series #1: Error on line 1: Undefined function or variable 'Spectrum_Power'.	Spectrum (Rectangular Graph)	Show
7	Error	Evaluating Series #1: Error on line 1: Undefined function or variable 'SineWave'.	Signal (Rectangular Graph)	Show
8	Warning	SpectrumAnalyzerEnv 'Design1.Spectrum': Does not have enough data to do the analysis.	Part 'Spectrum' in Design 'Design1'	Show

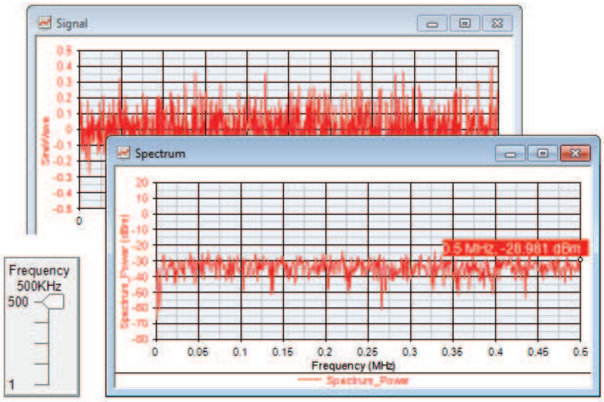
Automatically Display Errors Show Graph and Table Errors (2) Clear All Errors

Errors | Simulation Log | Compilation Log | Equation Debug

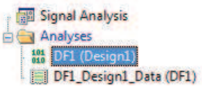
b. This occurs because 0 kHz is not a valid condition for the source. Often, one error might lead to other errors; simulation doesn't finish, data isn't collected and graphs cannot plot. Therefore, if you fix the first error, the others will be OK. To address this error, **edit the Frequency slider** and set the Min = **1** instead of 0. Then move the slider and it will be OK

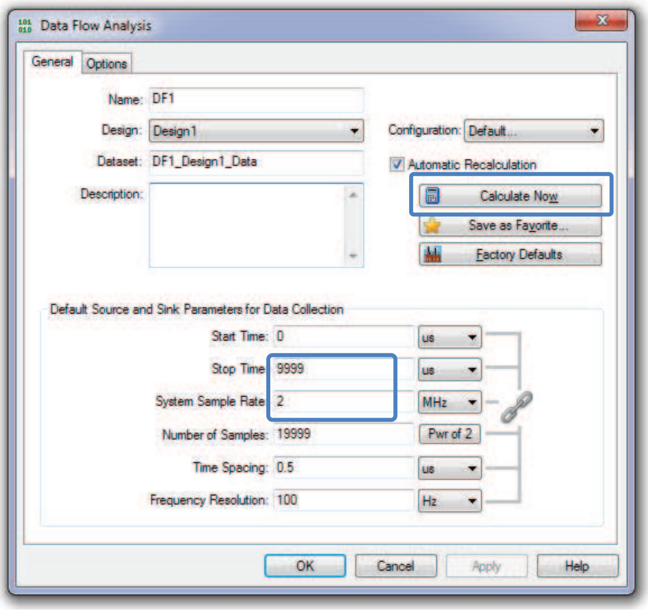
NOTE*: Try using the button to Clear All Errors – if the Error window still appears, close it as shown here.*

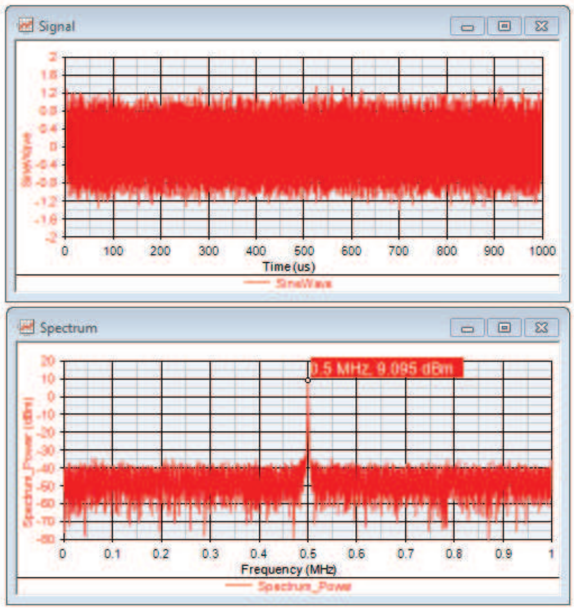
c. Move the Frequency slider to 500 kHz and observe the graphs. It looks like the signal is modulated and the spectral peak is in the noise. This is due to measurement aliasing and improper sampling (Nyquist theory: Fs/2).



d. To fix this problem, edit the **DF1** analysis control dialog box in the Workspace tree and increase the **Stop Time** to **9999** uSec and Sample Rate to 2. Click **Calculate Now** and you should see the correct results.







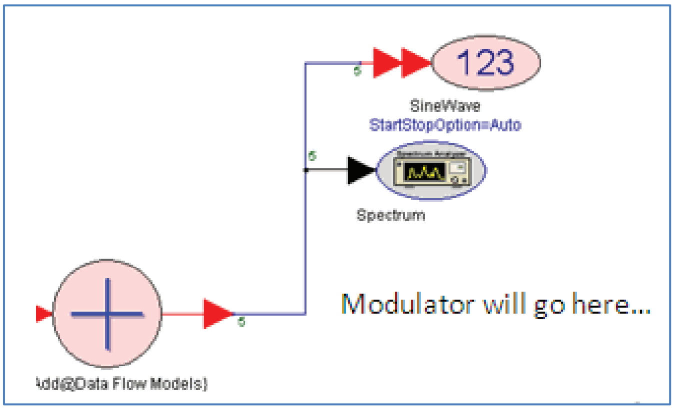
e. **Save** the workspace. The next steps will be to use a modulator and analyze the envelope, including more details about setups and plots in SystemVue.

6. Modulator Component in Schematic

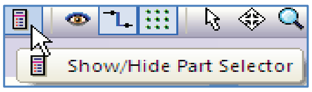
a. Save the current workspace with a new name (File > Save As): **Signal Analysis FM**.

b. Right click in an open area of the schematic and **uncheck** the box to **show the grid**. This will make it easier to see details in schematic.

c. Make the schematic window larger (drag the corner). Select (rubber-band) both **sinks** and **move them up and to the right** to make room for the modulator.

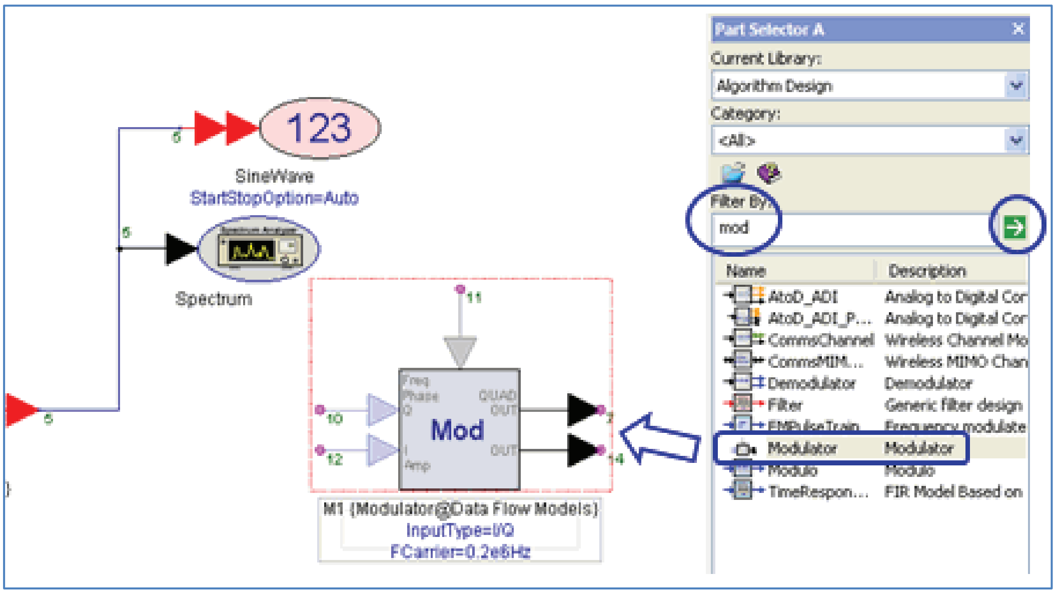


d. Go to the **Part Selector** – you can show or hide the Part Selector using the icon.

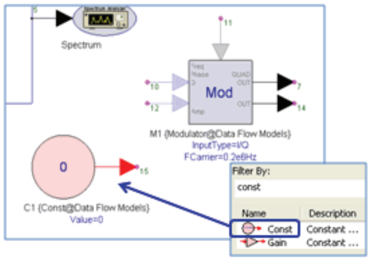


e. In the **Filter By** area, type in the name **mod** as shown here and press the green arrow button or Enter on your keyboard.

f. **Select the modulator and insert it** as shown here - to the right of the Adder.

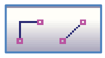


g. **Add a Constant** – Either use the Part Selector (Filter By) type in the name **const** and insert the constant, or simply use the schematic hotkey, ***C***.

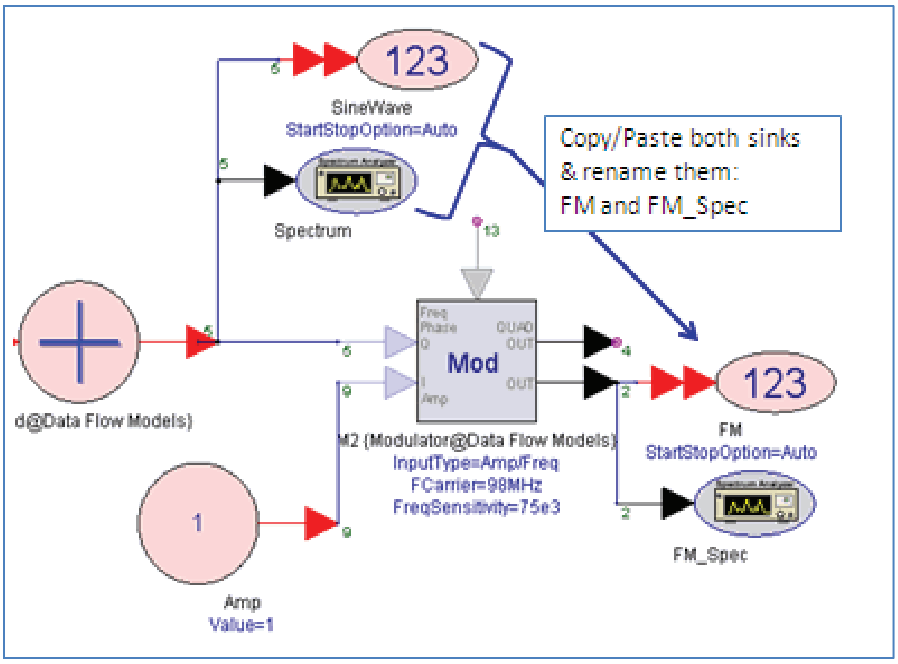


h. Copy/ Paste the 2 sinks and connect them to the modulator OUT. **Edit** them and name them **FM** and **FM_Spec** as shown.

i. **Connect** the adder, modulator and constant as shown – snap the pins together. Next, you will set the values.



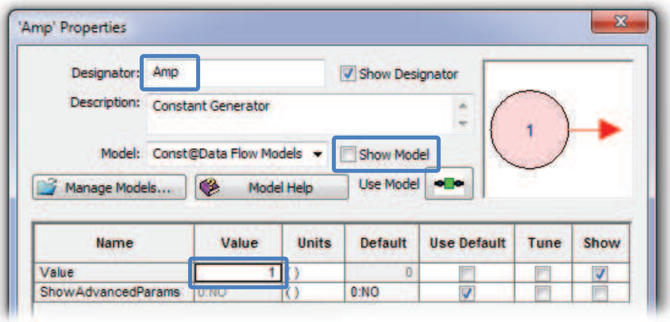
NOTE: *To move and disconnect selected components, use your left mouse button while holding down the keyboard **Alt** key.*



7. Modulation with Constant Amplitude

a. **Edit the Constant**:

- Change the Designator name to **Amp** (amplitude).
- **Uncheck** the box to Show Model.
- Set the Value to **1** and click OK.



b. **Edit the modulator:**

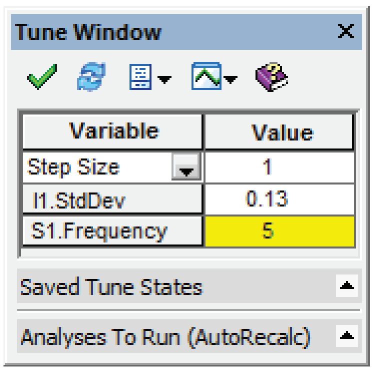
- Change the **InputType** to **Amp/Freq**
- Change the **FCarrier** to **98 MHz**
- Change the **FreqSensitivity** (Hz/V) to **75e3**

Also, the Show boxes should be checked for these.

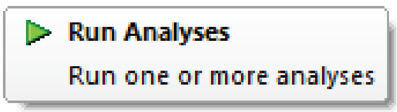
Name	Value	Units	Default	Use Default	Tune	Show
InputType	2:Amp/Freq	( )	0:I/Q	☐	☐	☑
FCarrier	98	MHz	0.2e6 Hz	☐	☐	☑
InitialPhase	0	deg	0 °	☑	☐	☐
AmpSensitivity	1	( )	1	☑	☐	☐
FreqSensitivity	75e3	( )	10000	☐	☐	☑

About the setup: The +/-1 V sine wave will modulate the frequency +/- 75 KHz from the nominal 98 MHz Center Freq, based upon the Frequency Sensitivity value. The FM waveform will have nominal amplitude of 1 V, based on setting the Constant value to 1.

c. Run the Analysis at 5 kHz by setting the signal generator frequency to **5 kHz** by entering the frequency directly in the Tune window – this is easier than using the slider.



d. Run the Analysis (green button).

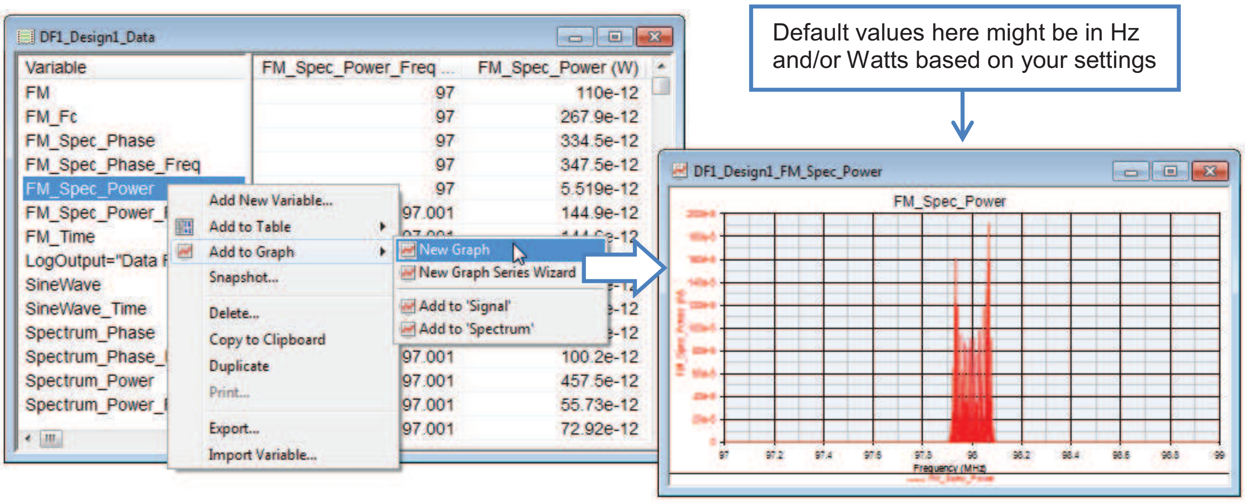


8. Graph the Power Spectrum

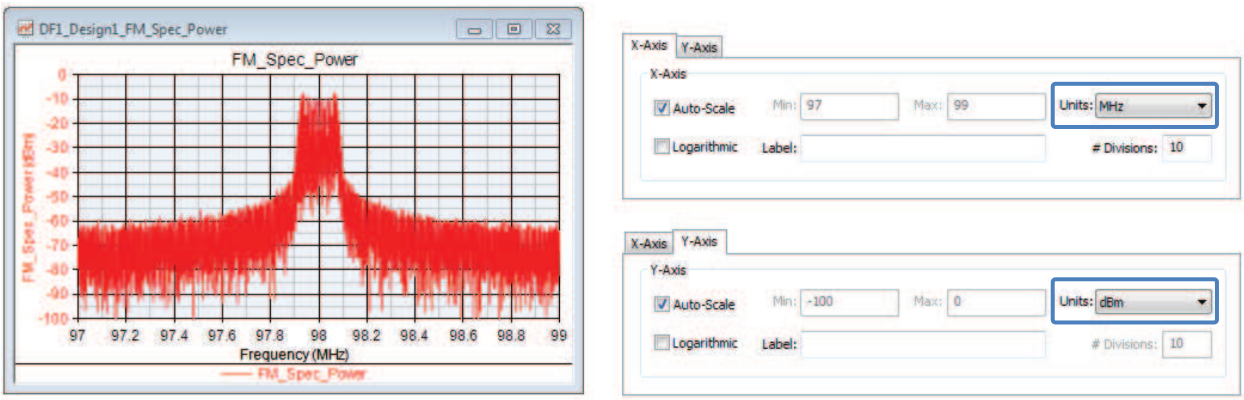
a. Edit (Double-click) the dataset to open it. DF1_Design1_Data (DF1)

b. Select the FM Spectral Power and right click to add a default New Graph, as shown here. Notice the results might not look as expected – this is because of the default graph axes settings, and the set units for a variable in the dataset.

(You will see why and learn more about this in section *9b)* on the next page)

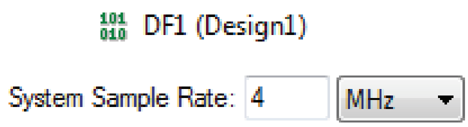


c. You can **Double-click** the graph background and change the units for the X-axis to **MHz** and the Y-axis units to **dBm**. The results should look correct now.



d. Notice the RF simulation Bandwidth is 2 MHz - this is from the System Sample rate set in the **DF1** Analysis Controller.

e. **Open** the Analysis controller, and **Change the rate to 4 MHz** as shown here, and click **OK**.

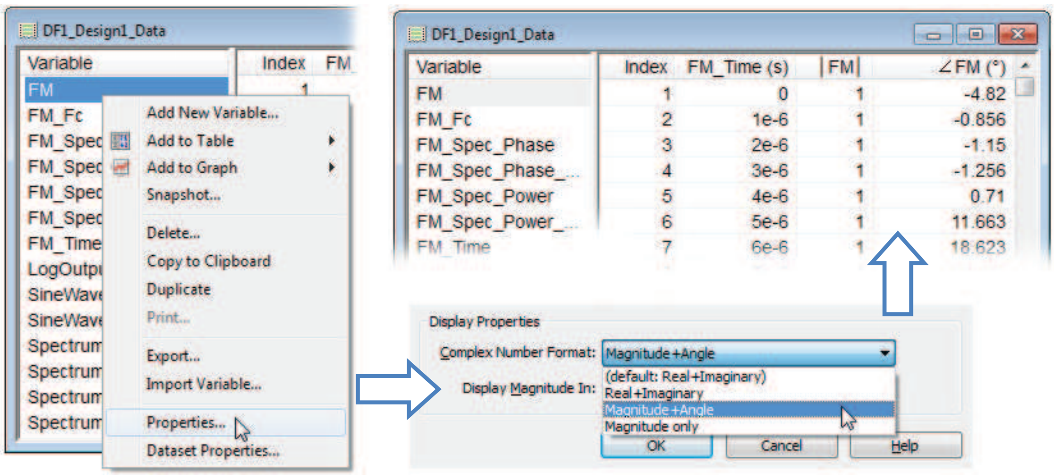


f. Notice the analysis runs again and the plot shows 4 MHz as the analysis BW. **Change it back to 1 MHz** – click OK. Now you know how to specify the sample rate for the resulting BW.

9. Data Equations and Plot Features

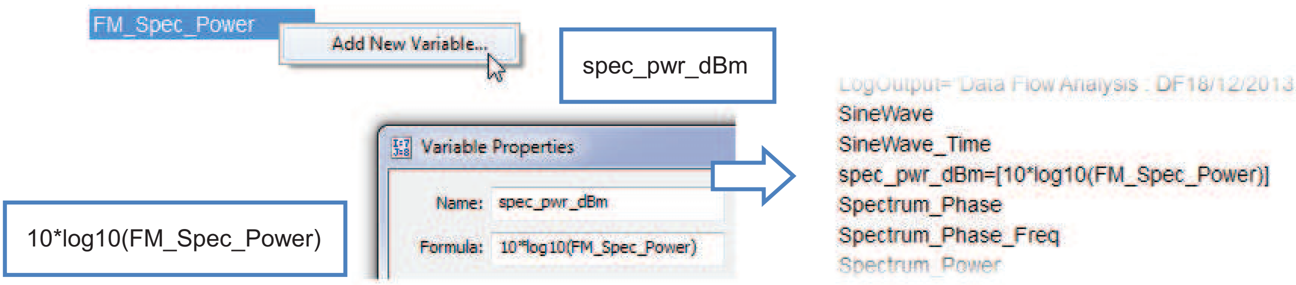
a. In the **dataset**, click on the **FM** variable and observe the values in the list.

b. **Right-click** on the **FM** variable and select **Properties** as shown here. Change the **Complex Number Format** to **Magnitude + Angle** and click OK. This shows how you can control the format and units of the data.



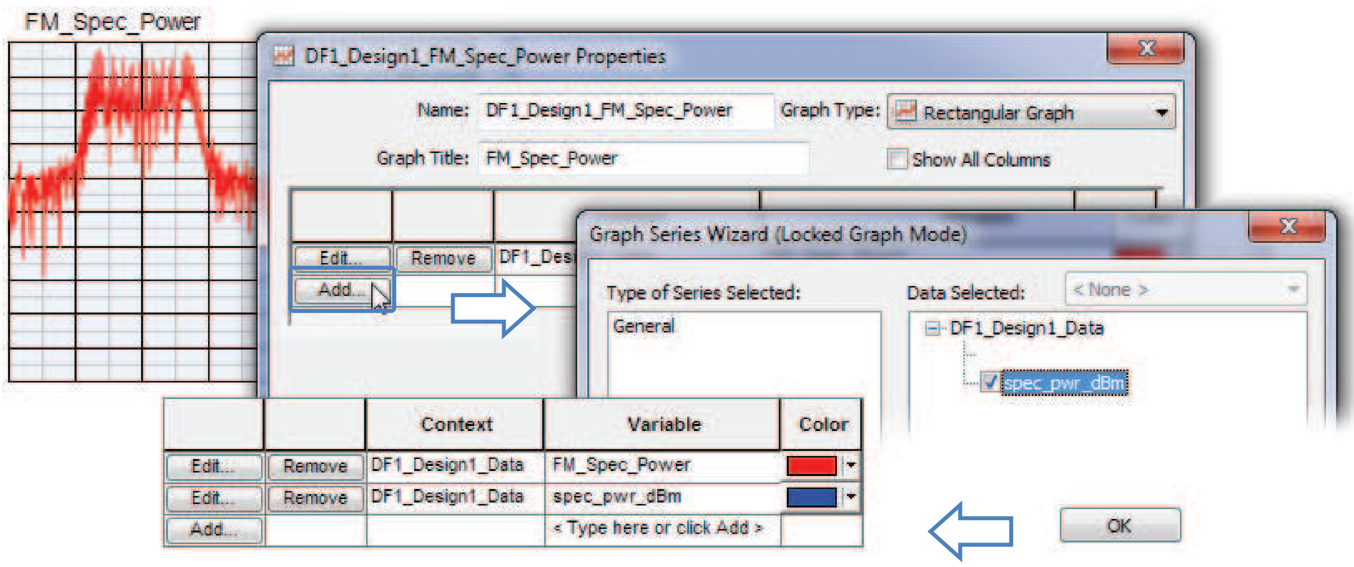
For example, changing the Unit of Measure of spectrum analyzer data from **W** to **dBm** will yield a *default graph* using the logarithmic dBm scale. Next, an equation.

c. In the dataset, **right-click** on the **FM_Spec_Power** variable and select **Add New Variable**. Type in the name and formula shown here and click OK.

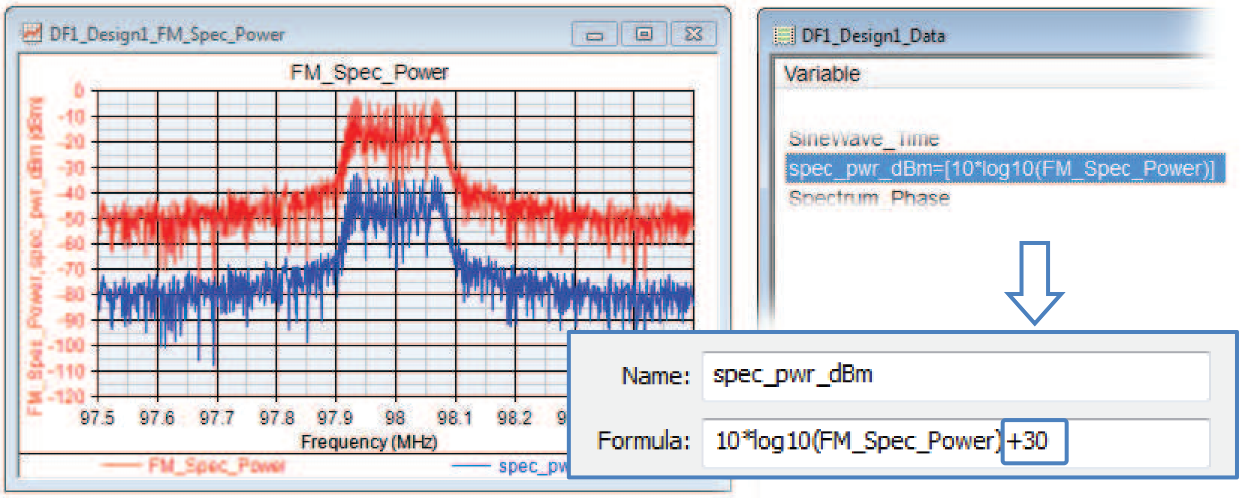


d. Verify that your dataset now contains the new variable equation as shown to the right.

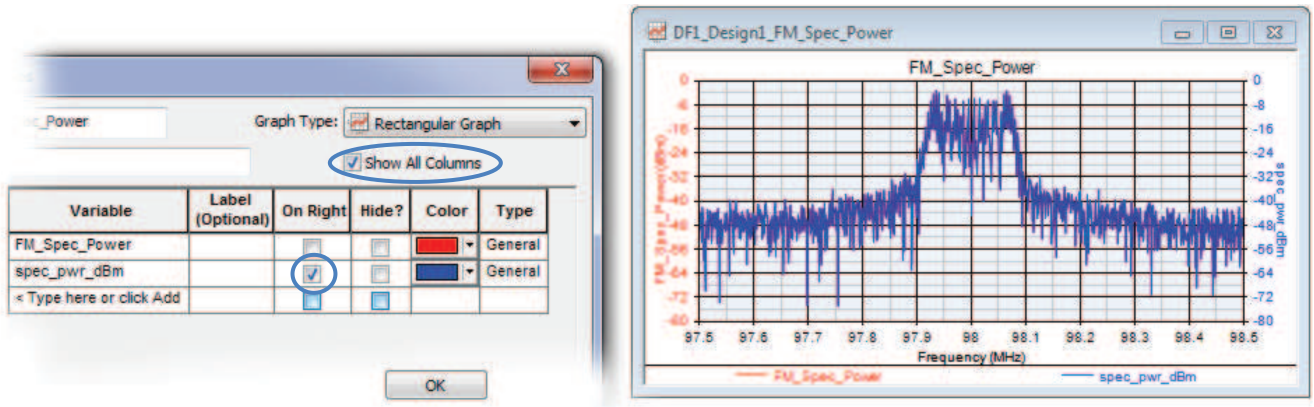
e. Add the new variable equation to your **FM_Spec_Power** plot. To do this, edit (double-click) in an open area of the plot. Click the **Add** button and then select your equation, **spec_pwr_dbm**, and click **OK** as shown here.



f. The plot will now show both traces. However, spec_pwr_dbm, is about 30 dBm below the other trace. This is because the equation is really dBW (watts). So, you need to **edit the equation** and add: **+30**. Edit the equation in the dataset and change it as shown here with +30 at the end and click OK.



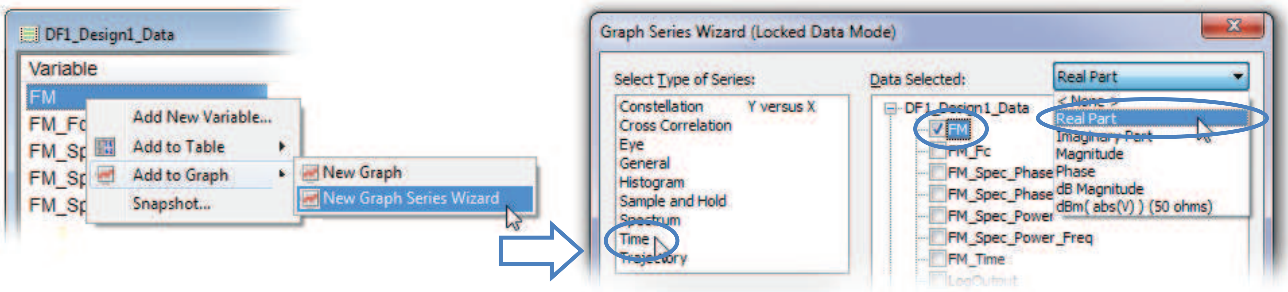
g. The plot will now be correct. Also, edit the plot and click the button to **Show all Columns** se the equation to be On Right and click OK. Now it is easy to identify the two traces with two Y-axes.

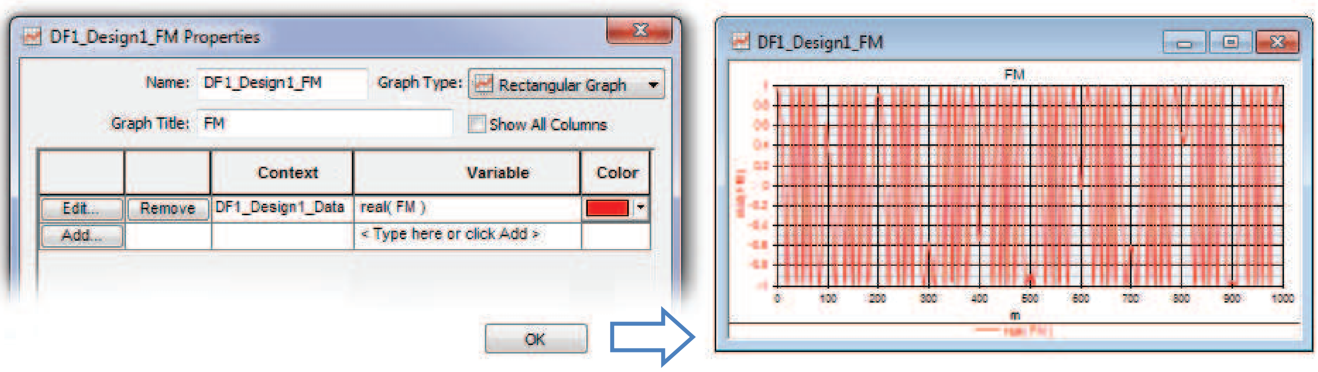


h. Save the workspace - this is always recommended when you achieve the desired results.

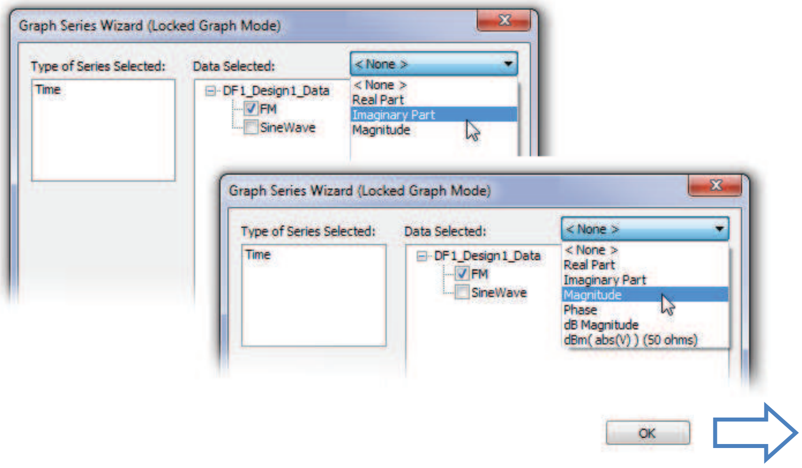
10. Use the Graph Wizard

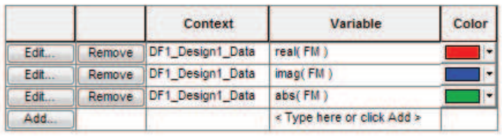
a. In the dataset, right-click on **FM** and select **Add to Graph > New Graph Series Wizard**. In the wizard dialog, select **Time** and **FM** and **Real Part** and click OK.

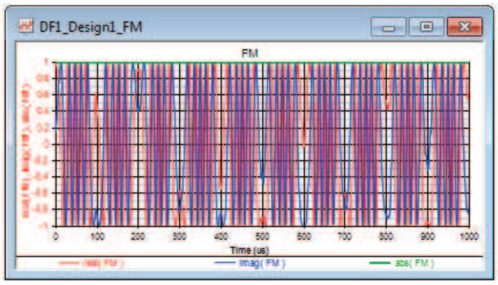




b. Next, you will add the Imaginary part and set the magnitude (abs) by pressing the Add button and selecting each one as shown here.

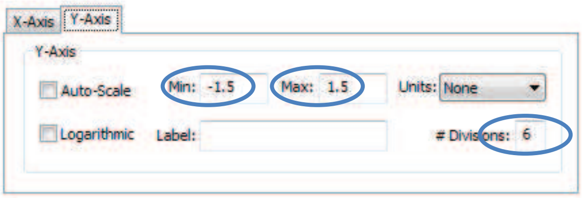


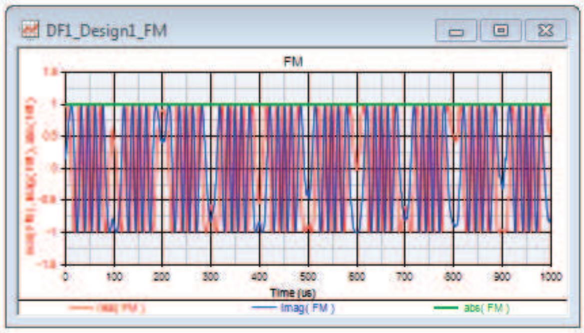




c. Note that the magnitude response is difficult to see. Edit (double-click) on the graph background and change the Y-axis scale **to -1.5** to **1.5** in **6** steps as shown here. Now you can see the flat line magnitude response which is the value of 1 at the top.

Now you know how to use the Graph Wizard to quickly plot data. Next, we will work with the FM radio band.

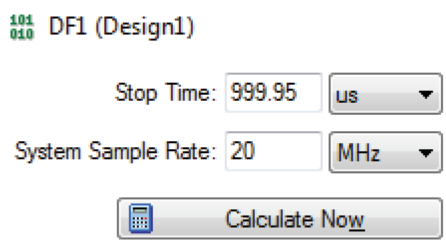




11. FM Analysis

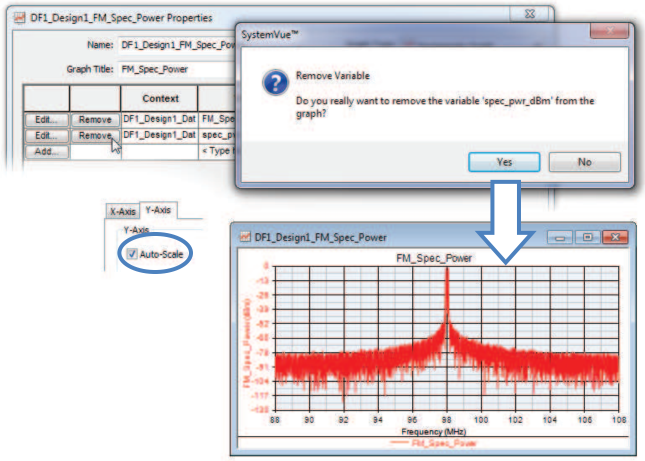
The US FM Radio band covers 88 to 108 MHz and has one hundred 200 KHz wide channels centered at every odd decimal place. (88.1, 88.3, 88.5, etc.). Our task is to have the simulation cover the whole band and utilize the tuner to switch between channels while observing the results.

a. In the Data Flow Analysis controller, change the **System Sample Rate** to **20 MHz**. Also change the **Stop Time** to **999.95 uSec.** Click **Calculate Now.**

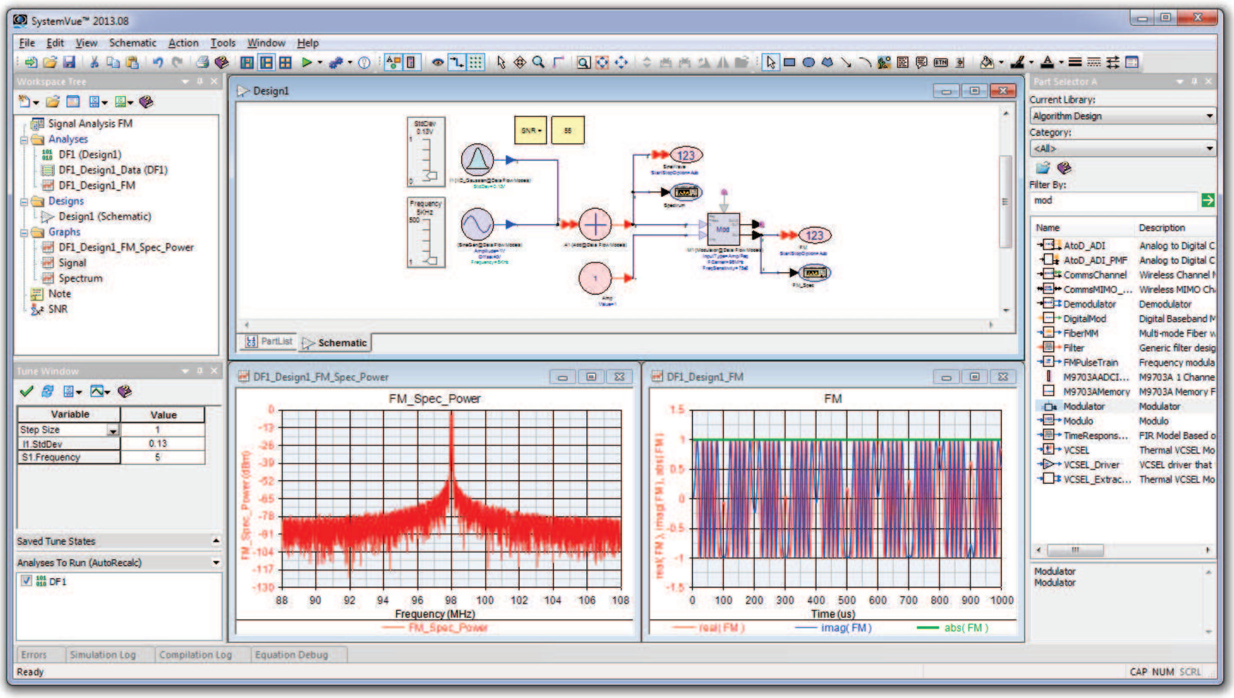


NOTE*: Using 999.95 us, and collecting 20000 samples, the Frequency Resolution and Time Spacing will be in round numbers (1000 Hz and 0.05).*

b. In the FM Spec_Power plot, remove the equation (spec_pwr_dbm) so that the results will be easier to see. Change the Y-Axis back to Auto-Scale.



c. In your SystemVue window, close all the plots and windows except the Schematic, and the FM Signal and FM Spectral Plots as shown here.

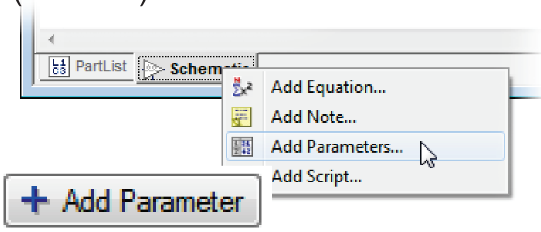


12. Creating a Parameterized Schematic

In order to make the tuning values meaningful, the schematic needs to be parameterized to select each FM channel (1 to 100).

a. At the bottom of the schematic window, right-click on the **Schematic tab** and select **Add Parameters**.

b. Click **Add Parameter** and then type in and select everything as shown here.

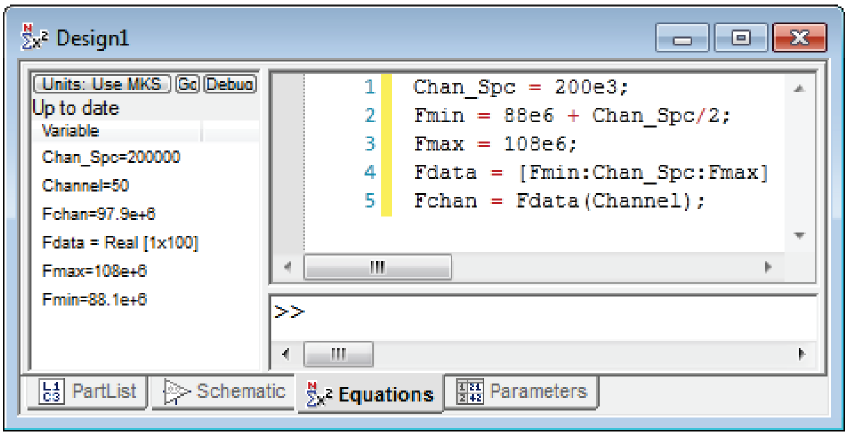


Name	Description	Default Value	Units	Tune	Show	Initially Use Default	Validation
Channel	Channel # (1 - 100)	50	( )	☑	☑	☐	Positive integer

c. Select the **Equations** tab that now also appear at the bottom of the Schematic. Notice the Channel value will appear in the left-hand column.

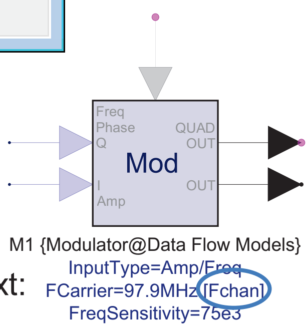
d. Add the following Equations 1 through 5. You do not need to type in the comments (%) – they are only here to explain the equations.

```
1  Chan_Spc = 200e3;               % Channel Spacing
2  Fmin = 88e6 + Chan_Spc/2;       % First Frequency (offset of 1/2 Ch. Spacing)
3  Fmax = 108e6;                   % Last Frequency
4  Fdata = [Fmin:Chan_Spc:Fmax]    % Create a vector of frequencies
5  Fchan = Fdata(Channel);         % Select the frequency for the channel selected
```



e. Click the **Schematic** tab, then edit the Modulator. Change the **FCarrier** value from 98 to **Fchan** (units = MHz) and click OK. Notice the FCarrier value should appear as 97.9 MHz on the Model text:

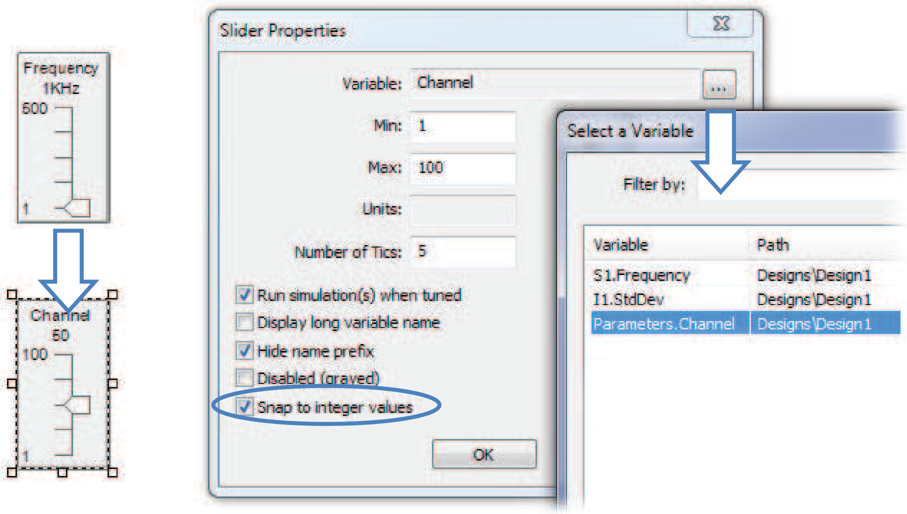
Name	Value	Units
InputType	2:Amp/Freq	( )
FCarrier	Fchan	MHz



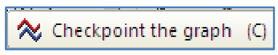
13. Tuning a Parameterized Schematic

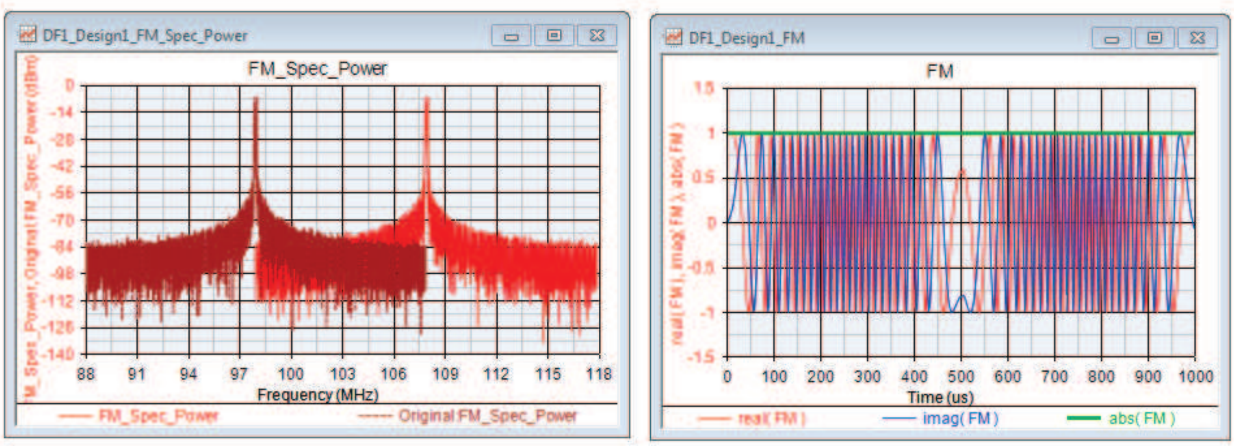
a. **Add another tuning slider** by copying an existing one: select it and then use your keyboard: **Ctrl C** and **Ctrl V**.

b. **Edit** the new slider and change the Variable = **Channel**, **Max = 100**, and set **Snap to integer values**.



c. Checkpoint the spectral plot and keep the Frequency at 5 kHz, and slowly change the Channel value using the slider. Notice the Time Domain and Spectral Plots.





QUESTION: *When you change the FCarrier value, why does the fc value change with it?*

ANSWER: *Without RF frequency dependence, as Fc changes, the I & Q values do not change. But Fc changes in the equation. Therefore, the time-domain FM signal plot will not change and the Spectral plot will shift to be centered around the new Fc value:*

$$x_a(t) = \left(x_i(t) + jx_q(t)\right) e^{j2\pi F_c}$$

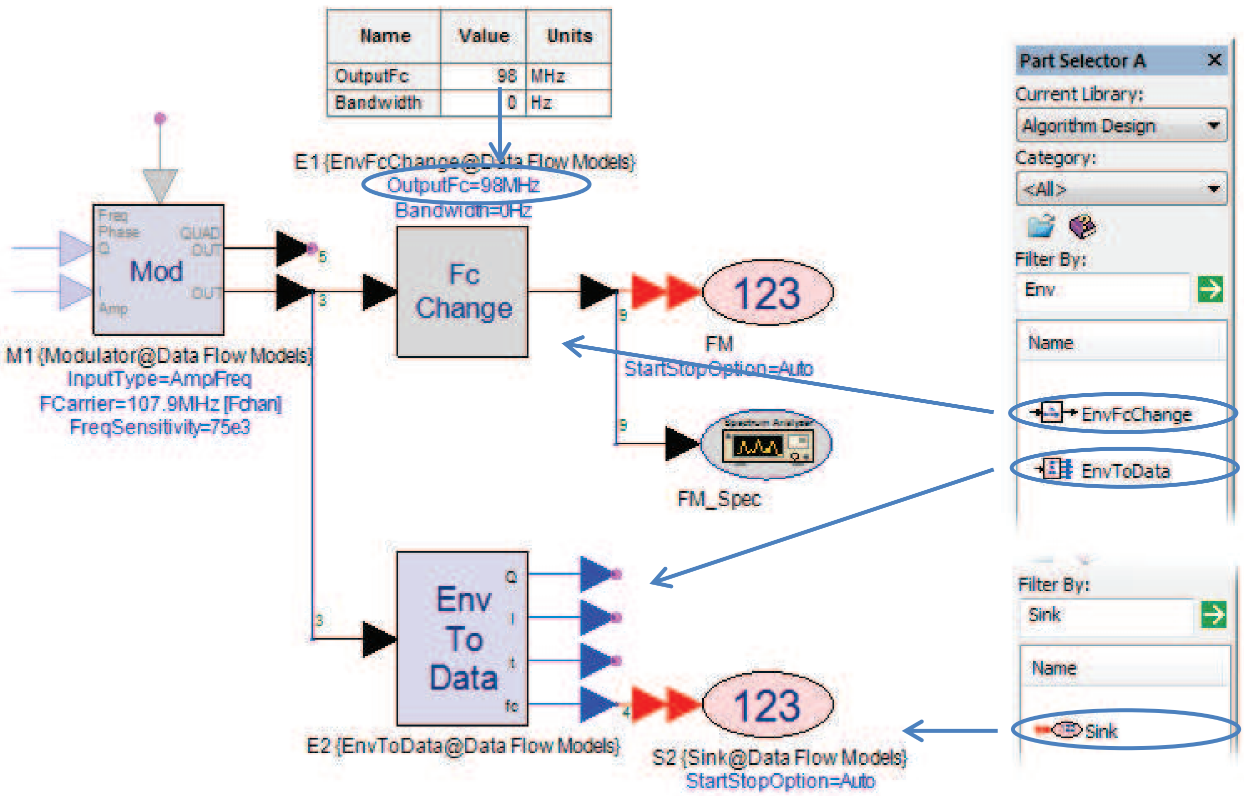
14. Evaluating the Envelope Parameters

Now you will use two new components for extracting data and debugging. First, the EnvToData component allows you to monitor any or all values of the envelope signal: *I(t), Q(t), time* or *the characterization frequency, Fc*. The EnvFcChange component allows you to convert the characteristic frequency – in this case it will fix the frequency at the specified value.

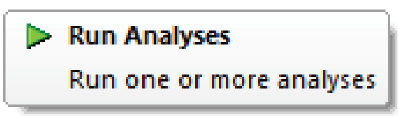
a. Move the **FM** and **FM_Spec** sinks to the right (use your Alt key to disconnect them). Then go to the **Part Selector**, in the Algorithm Library, and type **env** and press the green arrow button to use the *Filter By* function. Insert the **EnvToData** and the **EnvFcChange** components and connect them as shown.

b. Edit the **FcChange** component and set the **OutputFc** to **98** MHz. This will fix the characterization frequency. It won't change the signal, just how we see it.

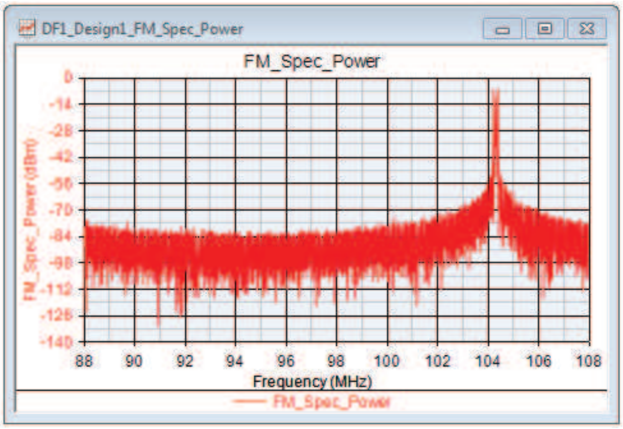
c. Also connect a new sink to the **EnvToData** component **fc** pin.



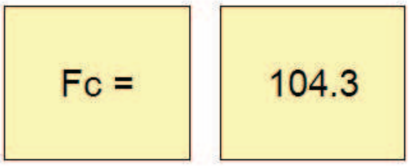
d. Run the Analysis.



e. Use the Channel tuner and notice that your spectral plot now shows the center frequency move across the FM band. If you did not use the FcChange component, the spectrum would be centered all the time. This way, you can better evaluate or compare channels.

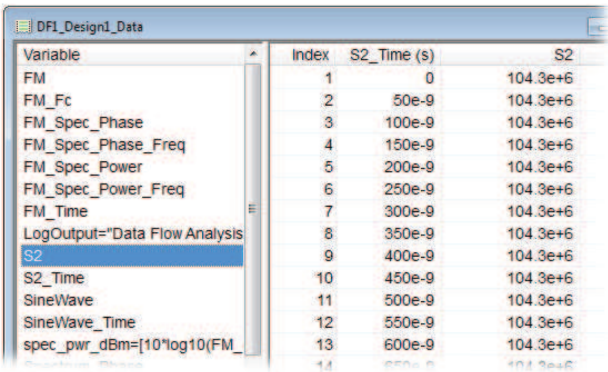


f. Add a text box, like the SNR, where Fc = (Fchan / 1e6).



NOTE: *If the Channel Frequency is moved close to the band edge some aliasing may occur. To avoid this, a simple solution is to increase the sample rate to make the bandwidth wider. This would have the added benefit of including the RF filter bandwidth and possibly including nearby interference signals, such as VHF Channel 6 which covers 82 to 88 MHz.*

g. Look in the dataset and you will see the S2 sink data for **Fc** is the same as your text box. This is just another way to get a numerical value using the sink with the **EnvToData** component.



Index	S2_Time (s)	S2
1	0	104.3e+6
2	50e-9	104.3e+6
3	100e-9	104.3e+6
4	150e-9	104.3e+6
5	200e-9	104.3e+6
6	250e-9	104.3e+6
7	300e-9	104.3e+6
8	350e-9	104.3e+6
9	400e-9	104.3e+6
10	450e-9	104.3e+6
11	500e-9	104.3e+6
12	550e-9	104.3e+6
13	600e-9	104.3e+6

h. Save the workspace.

<u>END OF LAB EXERCISE</u>

LAB EXERCISE 3

Bit Stream Analysis and Filters

This lab exercise introduces the bit source as the basis for digital designs, including eye diagrams and other data flow analysis tools. The basic design will represent the transmitter section of a system.

No perquisite is required for this lab.

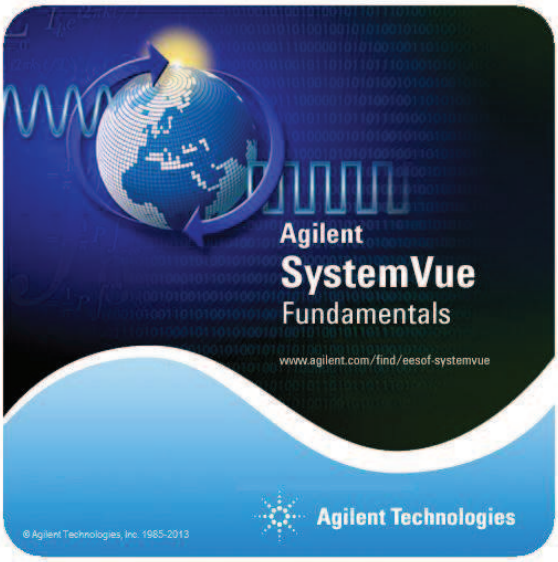


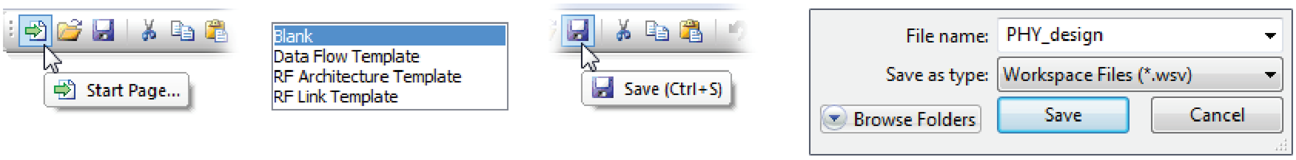
Table of Contents: Lab 3

Lab 3: Bit Stream Analysis and Filter Basics

1. Create a new blank workspace with folders

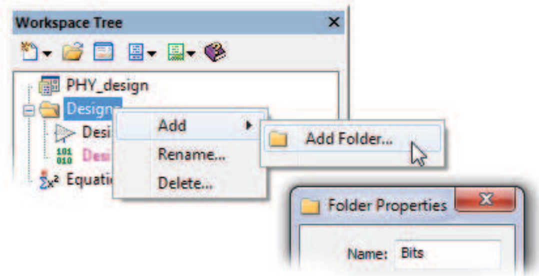
a. Create a new workspace using the Data Flow template.

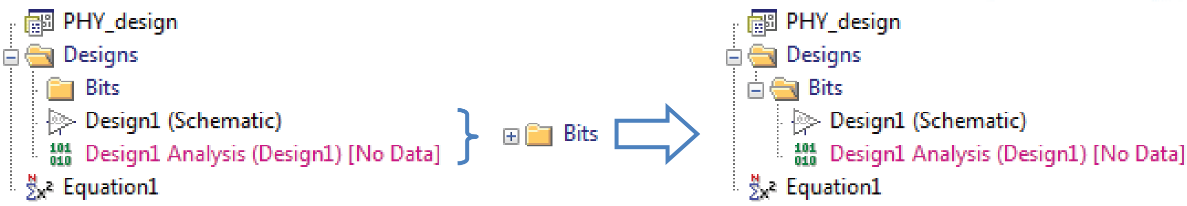
b. Save it with the name: **PHY_design**.



c. Place your cursor over the Designs folder and **right-click** as needed to add another folder and name it **Bits** as shown here.

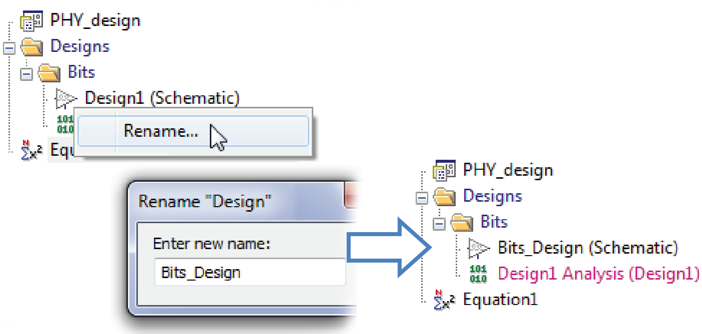
d. Drag the schematic and analysis icons into the new **Bits** folder as shown here.





e. Right-click the schemic icon and rename the schematic: **Bits_design.** Also, rename the Analysis as **DF_Bits** the same way.

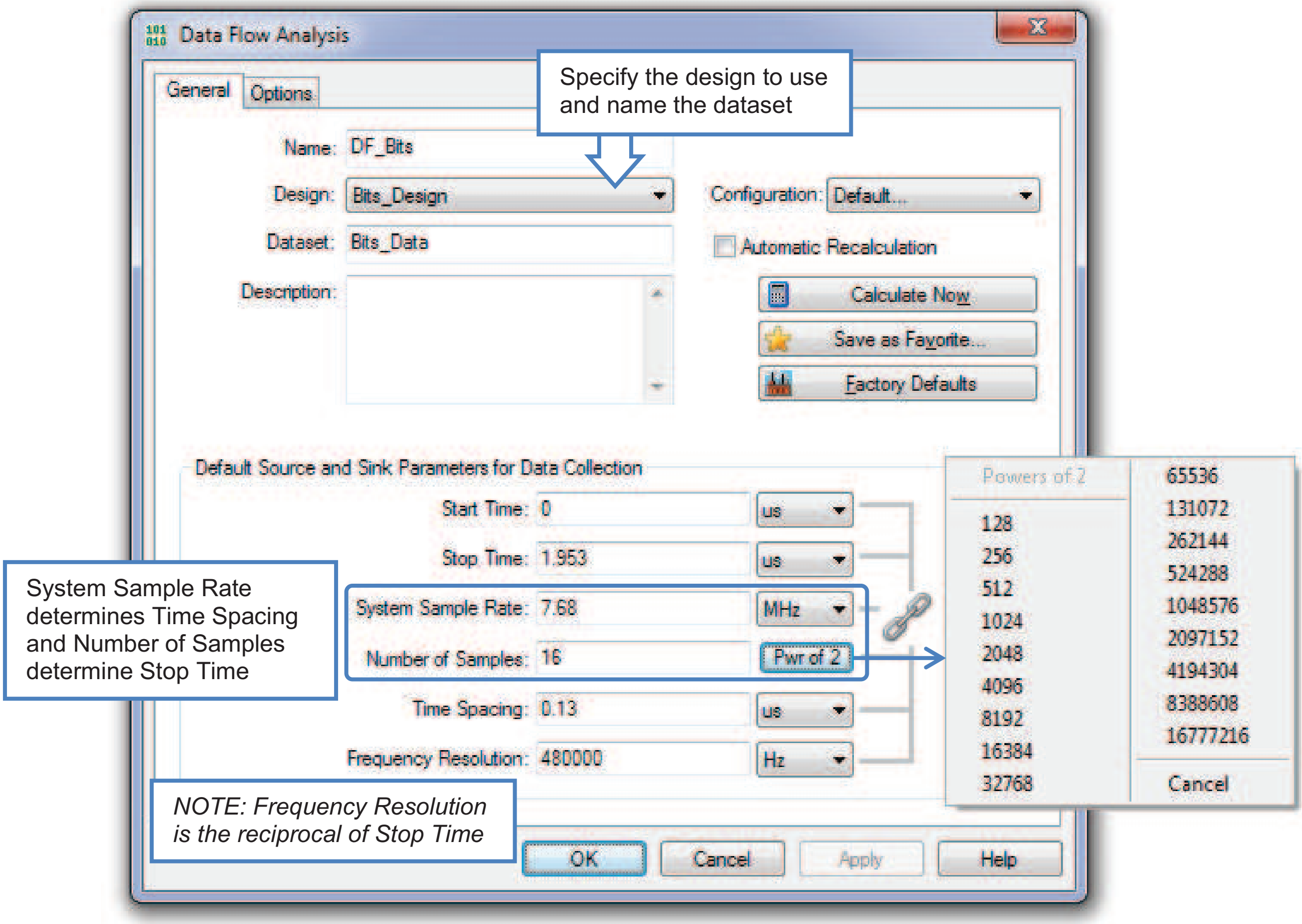
The result in an organized workspace with properly named components and folders.



2. Data Flow Analysis Setup

a. Edit (double-click) the DF Bits controller and setup the sampling for data collection as shown here: **System Sampling Rate: 7.68 MHz** and **16** Samples. Notice that you can always select a power of 2 from the pull down list. But for now, we will use 16.

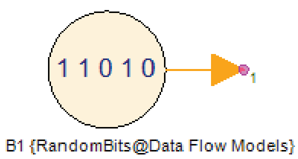
NOTE: *The System Sample rate used here is the rate of a standard clock oscillator: 7.68 MHz*

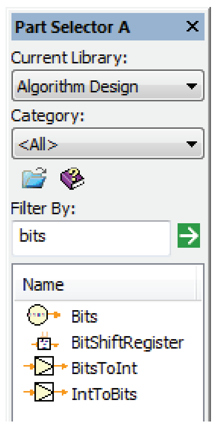


b. From the **Part Selector (View > Part Selector)**, type in **bits** and click the Filter green arrow – or, use the keyboard shortcut hotkey, ***B***. Then insert the **Random Bit Generator**.

c. Click **OK** when finished.

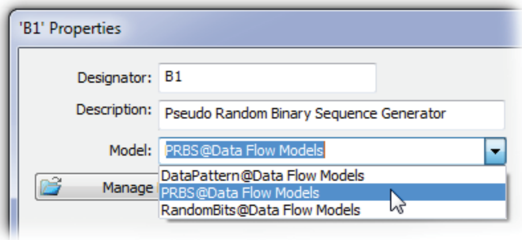
d. Save the workspace – this is always good practice.



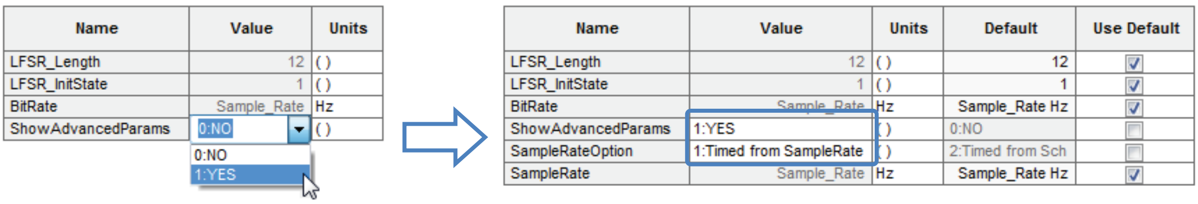


3. Bit Generator and Sink setup

a. Edit the bit generator and select the Model: **PRBS** as shown here. PRBS = Pseudo Random Bit Stream.

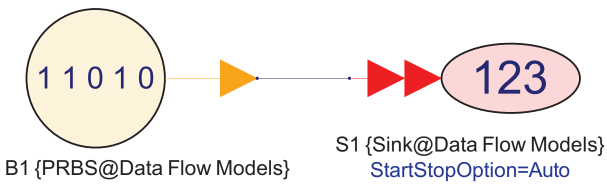


b. For the parameter settings, click on the the arrow button to Show Advanced Params and change it to **YES** as shown here. Then select the **Timed from SampleRate** option. Sample_Rate is a reserved system variable and this bit source will read the value that was set in the analysis (7.68 MHz).



c. Check the box to use the default (Sample Rate) as shown here if it is not checked. Click OK when finished.

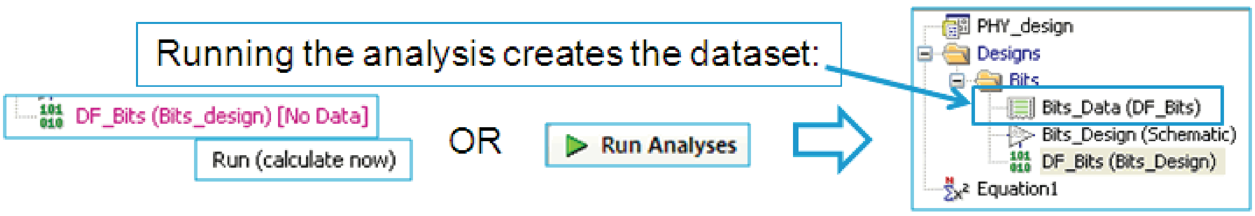
d. Connect a **Sink** to collect the data. To add the Sink quickly, use the keyboard hotkey ***s*** and click in the schematic. Then snap the pins together as shown here.



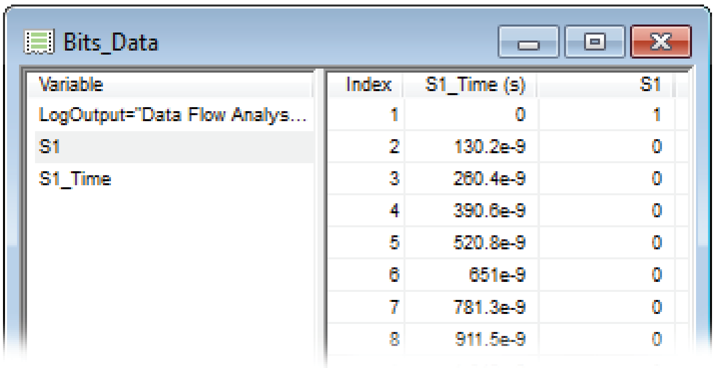
Notice the name of the sink is S1 – this is the designator name that you will see in the dataset after the analysis.

4. Analysis, Dataset graphs, and Trace control

a. In the workspace tree, right-click on the **DF_Bits** analysis and click **Run** (calculate now) – this is the same as the green arrow button you used before. Notice it immediately creates the dataset in the workspace tree.

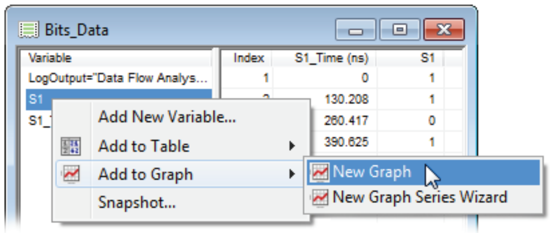


b. **Open the dataset** (Bits Data) by double clicking on it. Notice it has S1 (all the data samples) and S1_Time (time points only). Use your cursor to enlarge the window and move the columns so you can see the index values, the time points, and the 1s and 0s from the bit generator as shown here.



c. **Right-click** on the **S1** variable and then select: **Add to Graph > New Graph**.

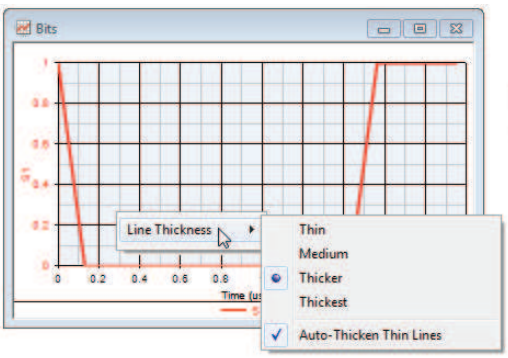
This is faster than adding a graph in the workspace tree and it allows you to specify the exact data you want – it is plotted immediately with default settings.

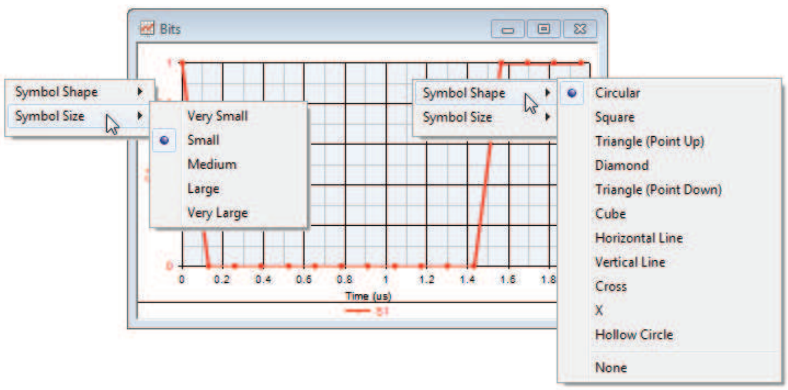


d. When the plot appears, it only has the narrow trace. To modify it, **right-click** on the trace and set the **Line Thickness** to **Thicker**.

Right-click again and set the **Symbol Size** to **Large**. Finally, use your keyboard **V** key to toggle the Vertex symbols on. The plot should be easy to read now with symbols at the 1s and 0s. You can also change symbol shape if desired.

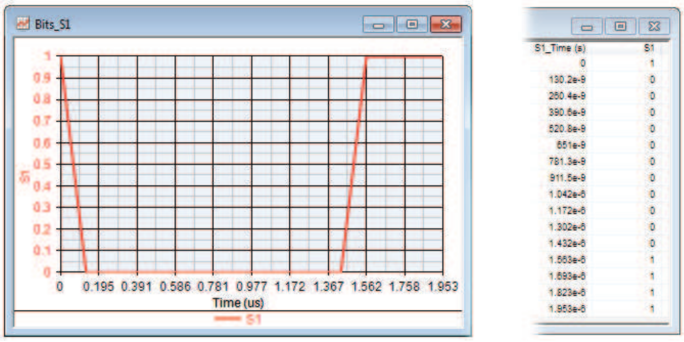
***NOTE**: If you accidently put a marker on the trace, delete it with the toolbar icon or use Shift +Del as shown here.*





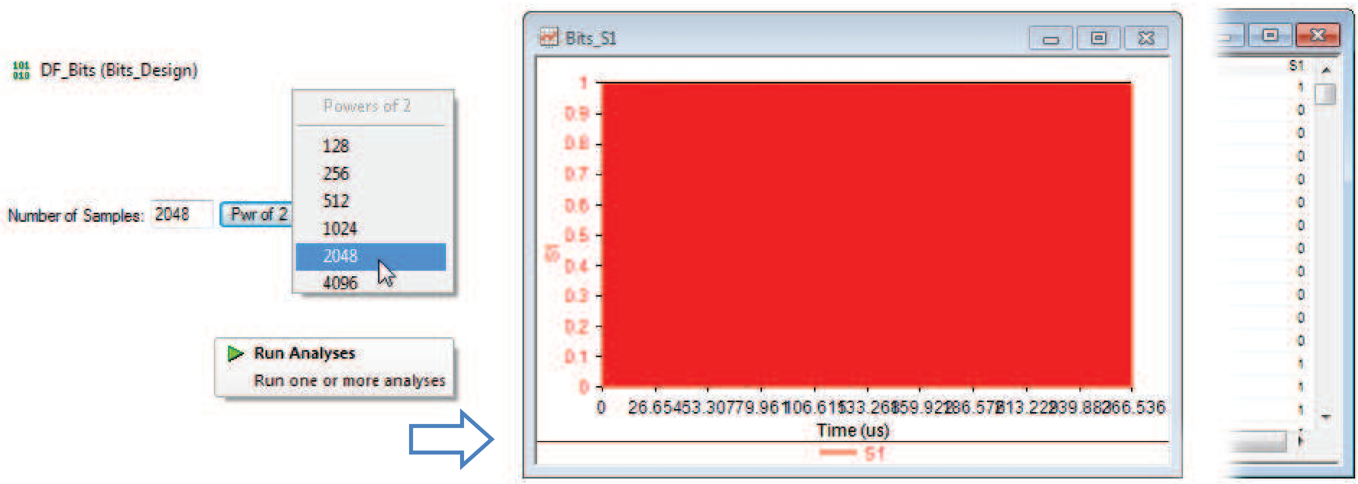
5. Sample Rate and PRBS source

a. Compare the dataset S1 (0s and 1s) to the plot - notice the bits match the trace symbols – this is due to setting the Sample Rate option to be timed to the analysis.

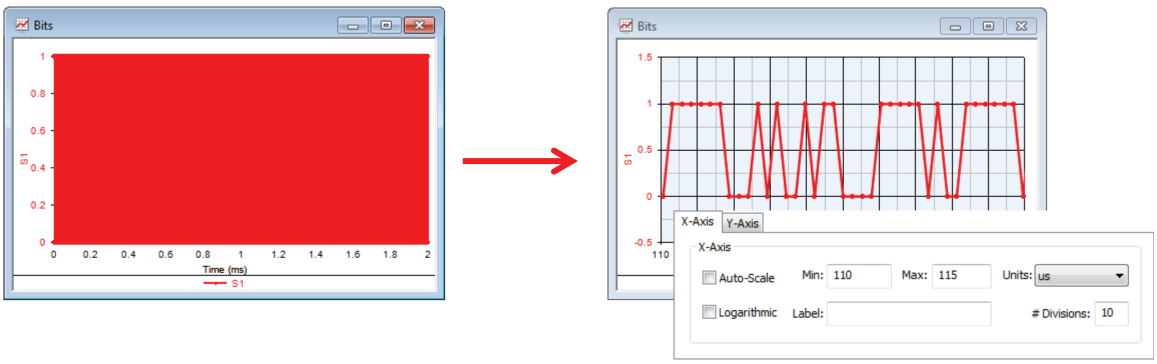


b. Run the analysis again and notice that there is no change to the data.

c. Go back to the Analysis controller (DF_Bits) and change the Number of Samples: use the ***Pwr of 2*** button to select **2048** and run the analysis again. Notice the bits do not change but the increased samples fill the plot.

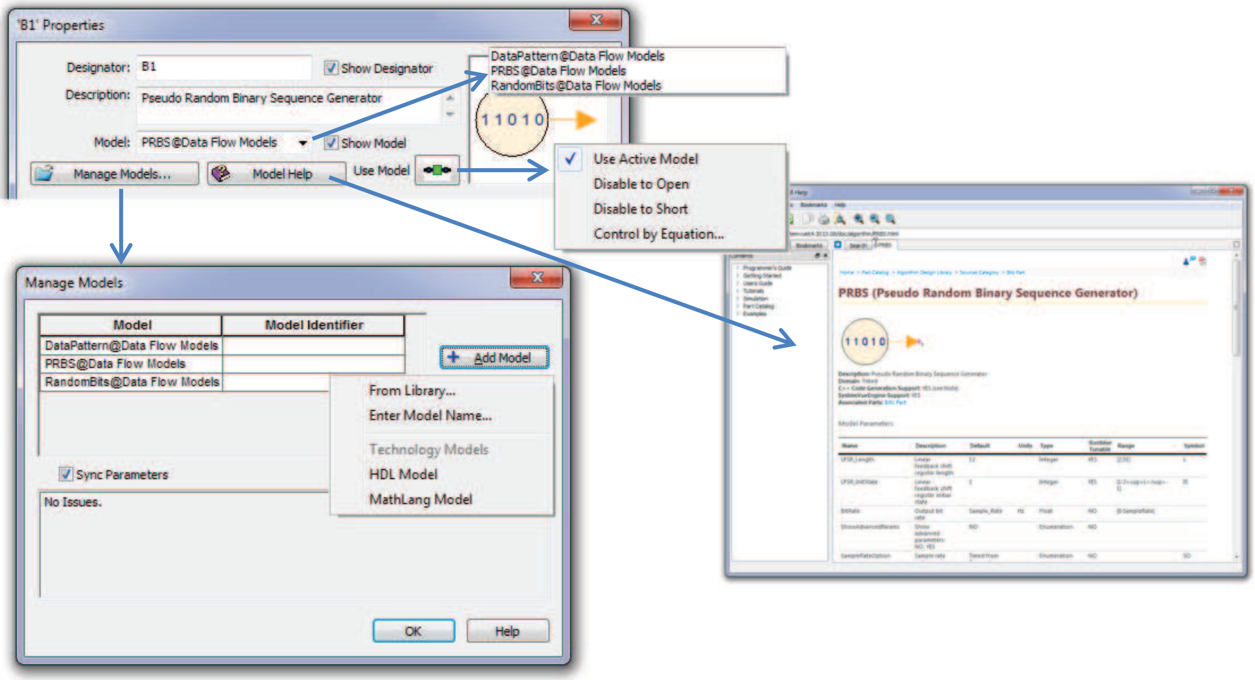


d. To scale the plot X axis, either A) use your mouse wheel with the cursor on the plot or, B) double-click the plot, un-check auto scale and set the Min and Max similar to the values shown here so you still see one trace symbol per bit.



6. Component Models, Help, and Disable

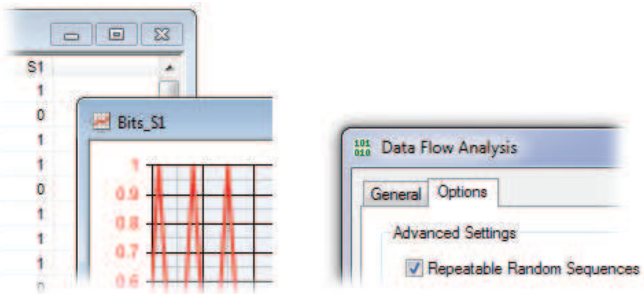
a. **Edit (double-click)** the **Bit source (PRBS)** and examine the settings as shown here. Try the **Manage Models** dialog and notice that you can add models if available. Try the **Help** button. Also, click the **Use Model** button and notice that you can disable the component (short or open) – try it but set it back to Use **Active Model**.



b. Finally, change the Model back to **RandomBits** and click OK.

c. Run the analysis again (2 or 3 times) and notice that the bits change when using this source.

d. Edit the analysis controller and, in the **Options** tab, check the box for **Repeatable Random Sequence** and run the analysis again (2 or 3 times) – now you see it repeats.



e. Go back to the **Analysis** controller and **uncheck** the Options for repeating the sequence and click **OK**.

f. Go back to the **Bit Source** and reset the model to **PRBS**.

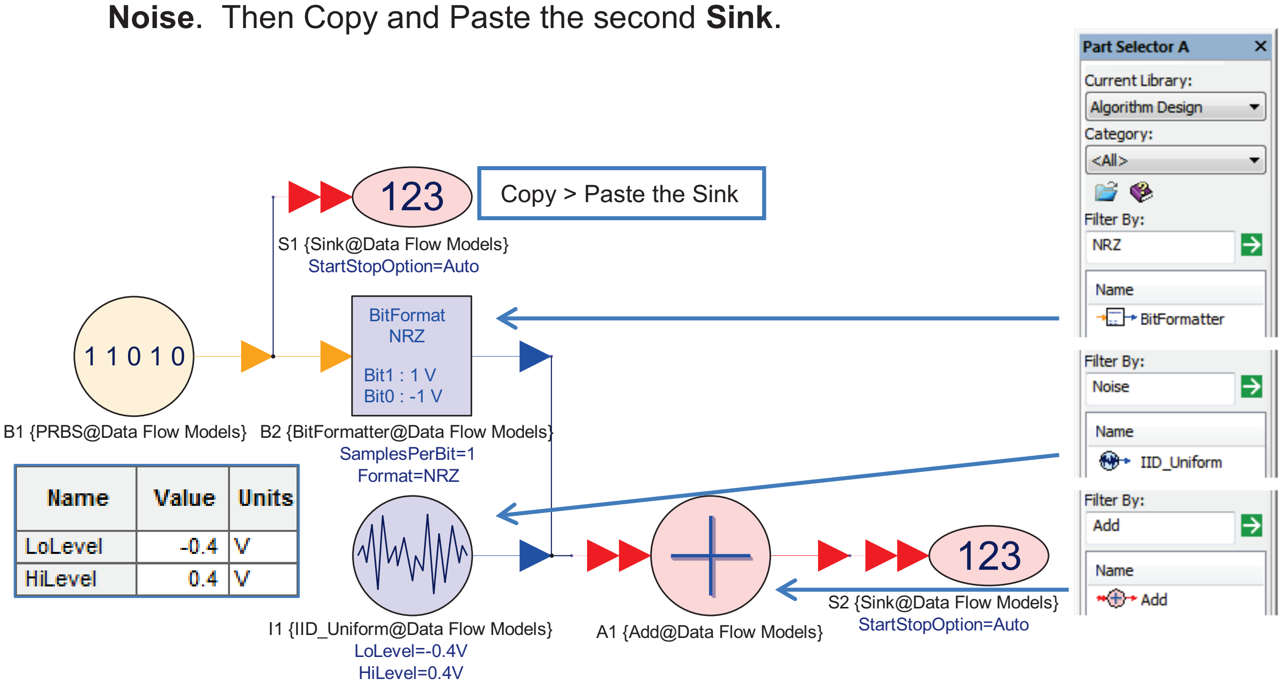
g. Save the workspace again.

***NOTE**: Setting the Repeatable Random Sequence is recommended for use with the RandomBits model for repetitive bit streams.*

7. Bit setup with NRZ and Noise for Eye Diagram

a. Enlarge the schematic window and modify the design as shown here. Go to the Part Selector and filter these three components and wire them as shown.

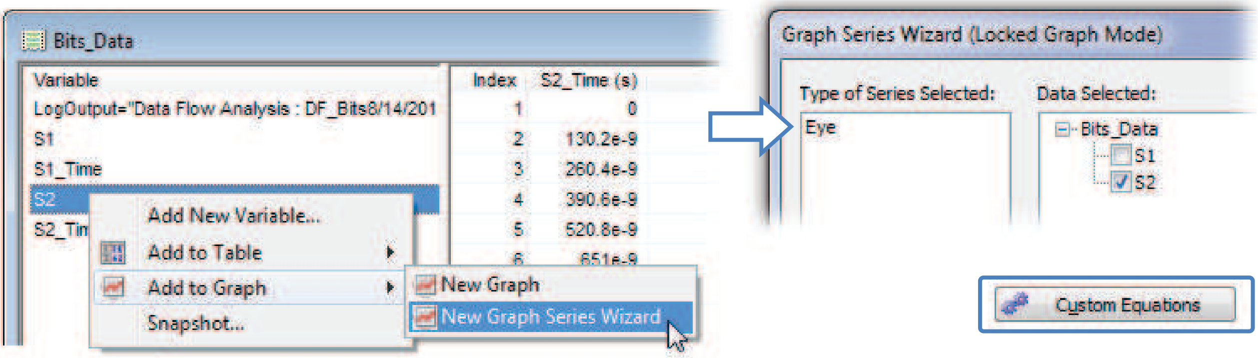
In the Part selector, type each of these in the Filter By field: **NRZ**, **Add**, and **Noise**. Then Copy and Paste the second **Sink**.



b. Edit the Noise IID and set the Level from -0.4V to 0.4V.

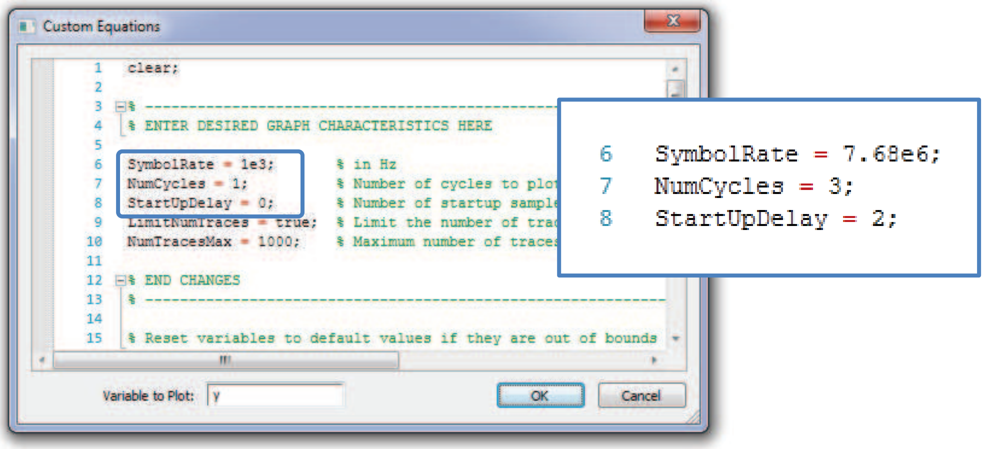
c. Save the workspace again and then **Run the Analysis**. 

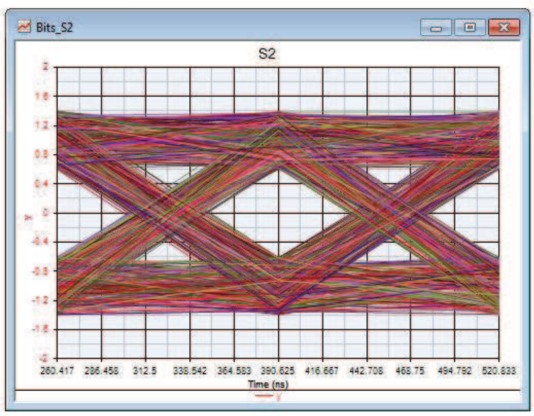
d. When the analysis completes, **edit** (double-click) the Bits_Data dataset and right-click the **S2** data: select **Add to Graph** and **Graph Series Wizard** as shown here. Then select Eye and click **Custom Equations**.



NOTE: *If you only use the New Graph selection, you will get another rectangular plot. For the Eye diagram, use the wizard.*

e. When the Custom Equations appear, change the **SymbolRate** to **7.68e6** (same as the analysis), set the **NumCycles** to **3** and **StartUpDelay** to **2**. Then click OK as needed (3 times) and the Eye Diagram will appear.





f. Edit the plot and **change the name** to **Bits_S2_EYE** as shown here.

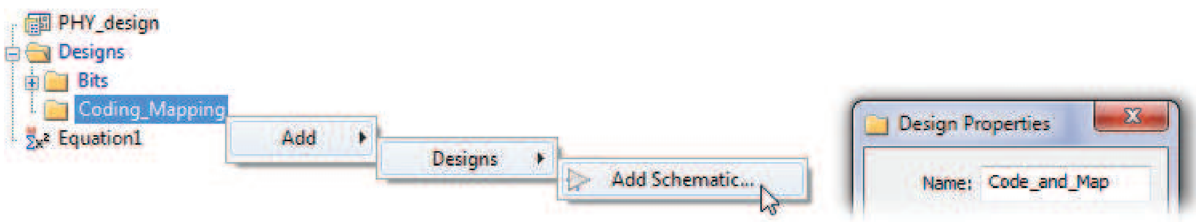
NOTE: *Setting the StartUpDelay to 2 or more is a good practice but not necessary for this design.*

g. Save the workspace again and close all the windows except the schematic.

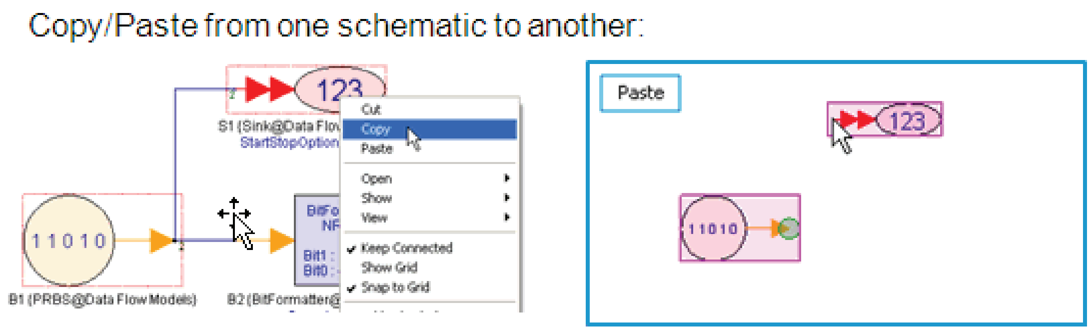
8. Coding and Mapping Components

a. Right-click on the Design folder in the workspace. Then add a new folder and name it: **Coding_Mapping**.

b. Right-click on the Coding_Mapping folder and add a new design. Name the new schematic: **Code_and_Map**. You cannot use the same name as the folder, as every item in the workspace tree requires a unique name.

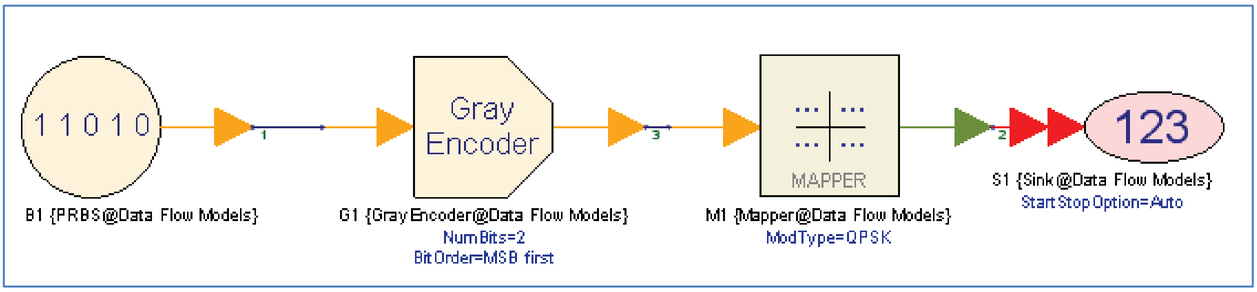


c. Go to the **Bits_Design** schematic. Select both the **B1 PRBS** and the **S1 Sink** and right-click the Copy command. Then go to the new schematic, **Code_Map** and paste the two components – this saves time.



d. Close the Bits_Design and collapse the Bits folder as shown here. 

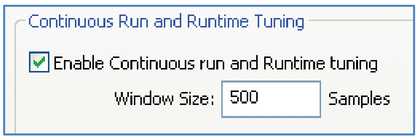
e. Go to the **Part Selector** and add a **Gray Encoder** and a **Mapper** – use the **Filter By** field and type in the names as you have done before. Edit the encoder and change the **NumBits = 2**. Also, set the Mapper to **QPSK**.



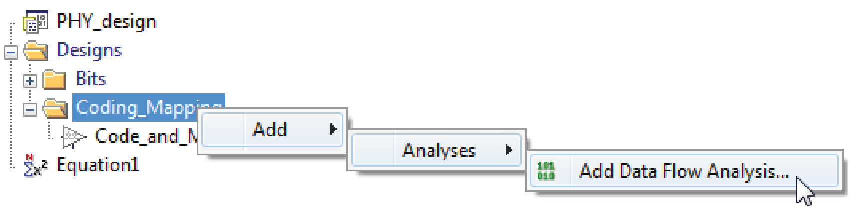
The Gray Encoder will ensure the synchronous change of state (one bit at a time) – this is especially useful when spikes or noise are present. Also, the number of bits is set to 2 for QPSK (2 bits per symbol).

9. Continuous Data Run and Plot

a. Edit the **Sink** and **Enable Continuous run and Runtime tuning**. Then click OK. This sets up the run time plot as you will see.

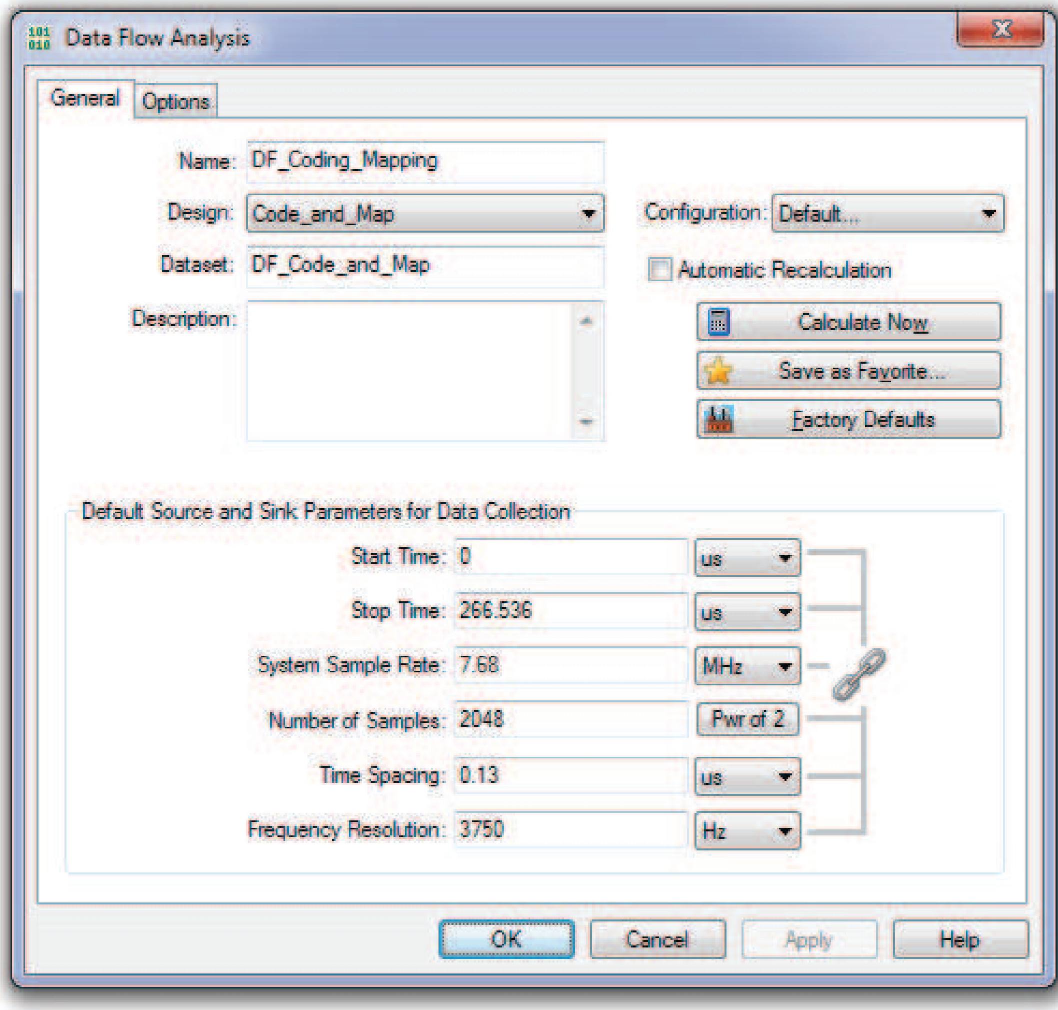


b. Right-click the **Coding_Mapping** folder again and add a **Data Flow Analysis**.



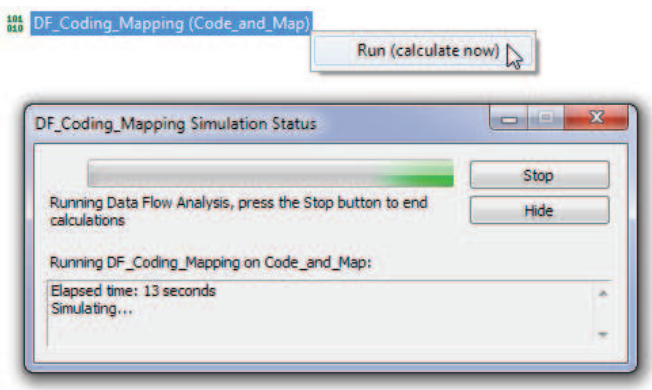
c. Name the analysis: **DF_Coding_Mapping**. Also, select the Design: **Code_ and_ Map)** and name the Dataset**: DF_Code_and_Map**. You should always check these items if you have multiple designs in the workspace.

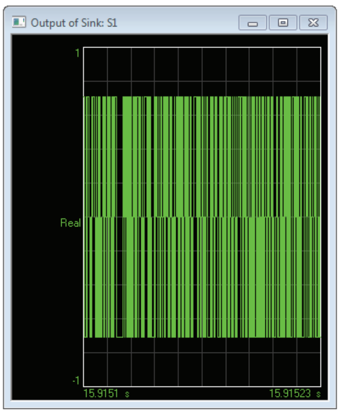
d. Specify the **System Sample Rate: 7.68 MHz** and **the Number of Samples: 2048** - use the ***Pwr of 2*** button. Click **OK**.



Now it is time to run the analysis and view the real time plot to verify that the QPSK data has the correct behavior.

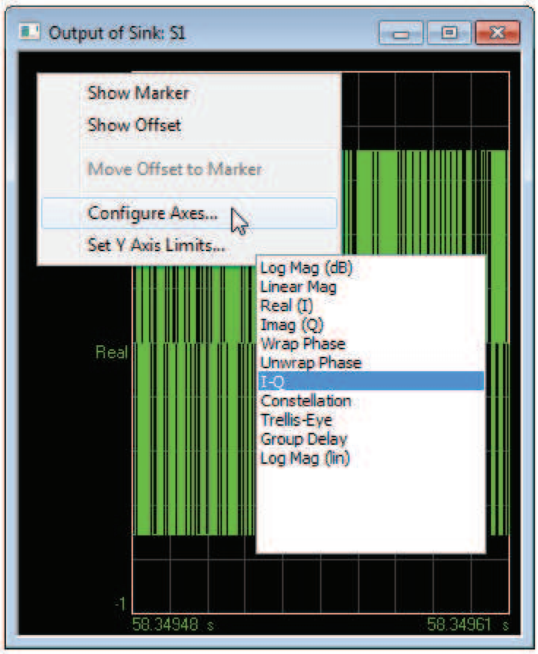
e. **Right-click** on the analysis in the Coding_and_Mapping folder. Then select **Run (calculate now).** This means you only run this specific analysis and not all the simulations in the workspace.

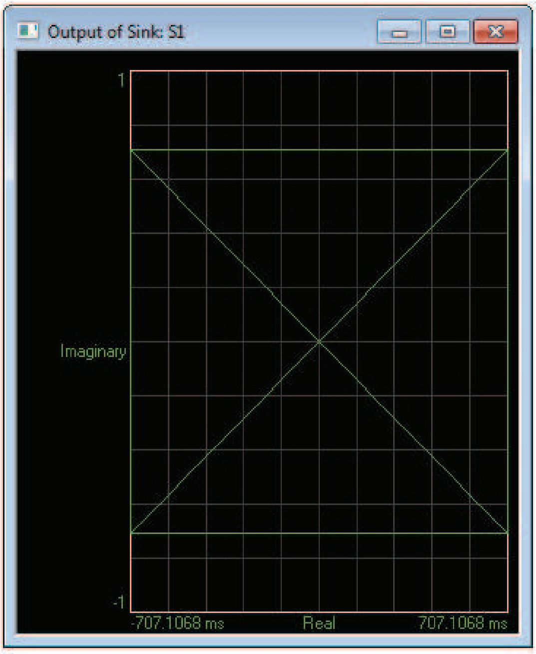




You should immediately see the continuous plot and the Status dialog that allows you to stop the analysis. The plot default might be the real valued signal – so to view the perfect QPSK constellation, you can change it by right-clicking on it and using **Configure Axes…** to change the format to **I-Q** or **Constellation**. You can also adjust the scaling by using the **Set Y Axis Limits**.

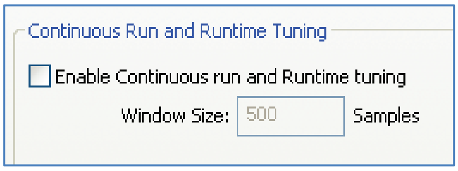
NOTE: *For your designs, this real time plot could show abnormalities over time – this would allow you to determine when they occurred so you could correct them.*





f. When you are finished, stop the analysis using the **Stop** button in the dialog. Then close the plot as shown here – these plots are not saved.

g. Reset the sink by disabling the continuous run – this means uncheck the box as shown here. The sink will then be in the default automatic data collection mode to the dataset.



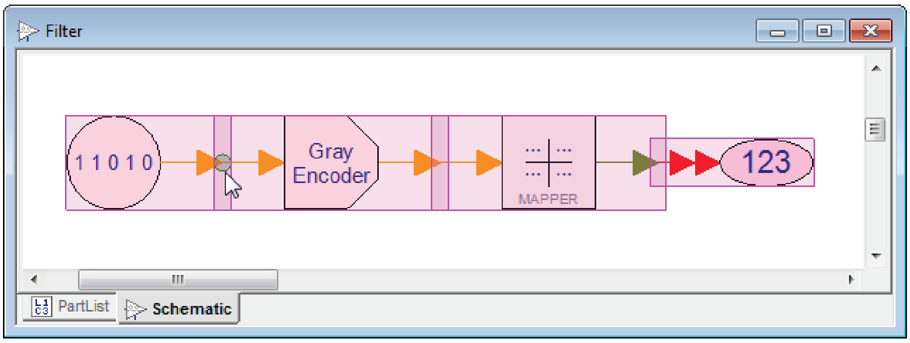
h. The next steps will be to use the filter designer.

i. Save the workspace.

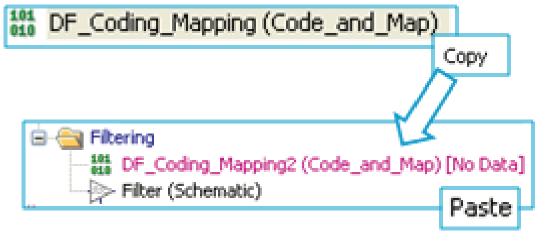
10. Envelope Analysis: Errors and Models

a. Add a new folder to Designs name it: **Filtering**. Then add a schematic and name it: **Filter**. First you will learn about spectrum component models and then you will build a filter.

b. Copy and paste the schematic components from the Code_and_Map schematic into the new Filter schematic.



c. Right-click on the **DF_Coding_Mapping** analysis and **Copy** it – then right-click on the **Filtering** folder and **Paste** it as shown here.

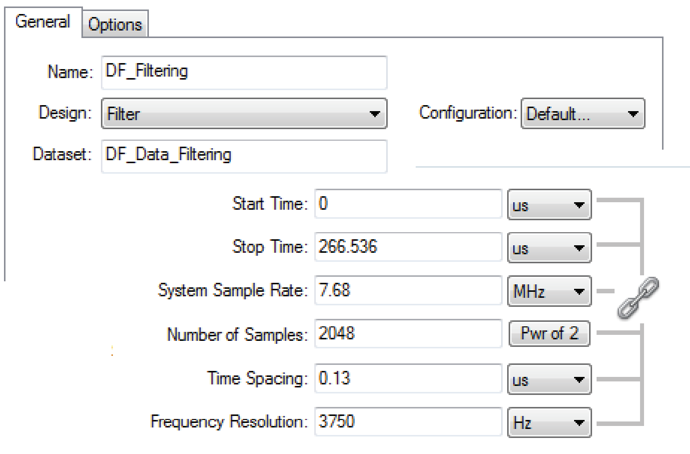


d. Edit the copied analysis in the **Filtering** folder and **change the names** as shown here:

Name: DF_Filtering

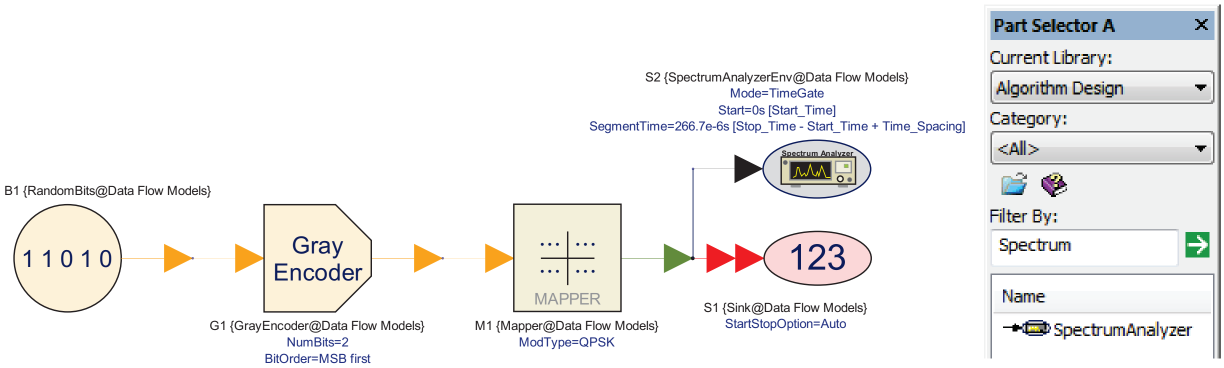
Design: Filter

Dataset: DF_Data_Filtering

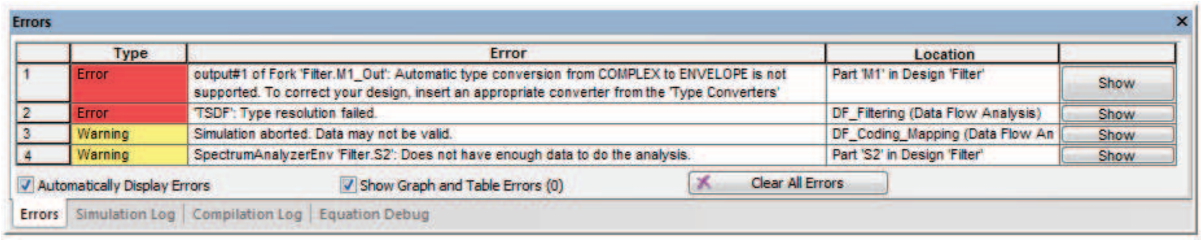


e. **Click OK**. This way, the analysis values are already specified from the previous analysis.

f. Insert a spectrum analyzer - type **Spectrum** in the Part Selector filter and connect a **Spectrum Analyzer** to the output as shown here. Notice the black arrow on the analyzer input – this means the input is an envelope signal.



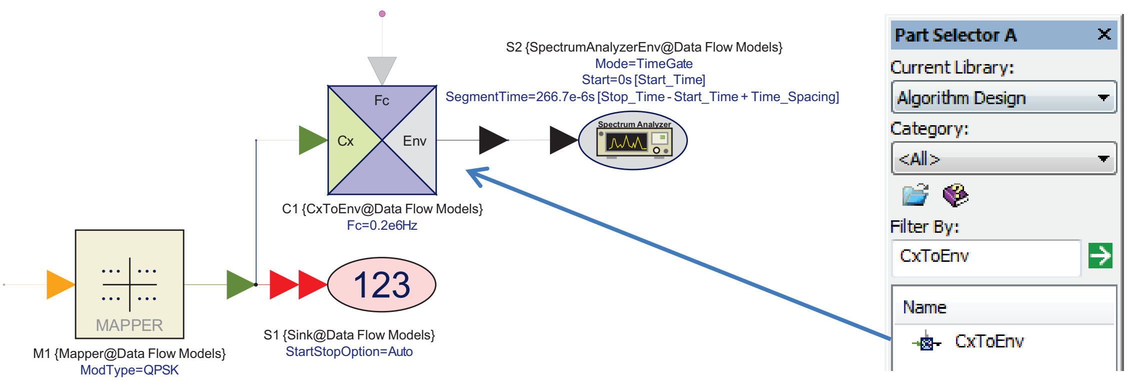
g. Right-click on the **DF_Filtering** analysis controller and click **Run** to calculate - you will see an error message immediately. Click on each of the Show buttons and notice that the first one refers to the controller and the other two refer to the spectrum analyzer.



	Type	Error	Location	
1	Error	output#1 of Fork 'Filter.M1_Out': Automatic type conversion from COMPLEX to ENVELOPE is not supported. To correct your design, insert an appropriate converter from the 'Type Converters'	Part 'M1' in Design 'Filter'	Show
2	Error	'TSDF': Type resolution failed.	DF_Filtering (Data Flow Analysis)	Show
3	Warning	Simulation aborted. Data may not be valid.	DF_Coding_Mapping (Data Flow An	Show
4	Warning	SpectrumAnalyzerEnv 'Filter.S2': Does not have enough data to do the analysis.	Part 'S2' in Design 'Filter'	Show

The reason for the error is that the Spectrum Analyzer is not receiving a carrier signal - the QPSK is only at baseband and not modulated onto a carrier. As the first error indicates, you can insert a **CxtoEnv** (complex to envelope) component. Also, it's a good idea to clear the errors before closing the error window.

h. Insert a **CxtoEnv** component as shown here and run the analysis again. There should be no error now because a carrier is included.

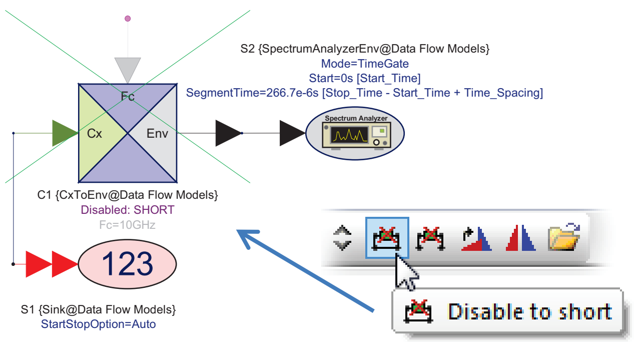


i. Add a graph of the S2 power as shown here – you see a very wide spectrum around the carrier which will be changed soon with a filter.

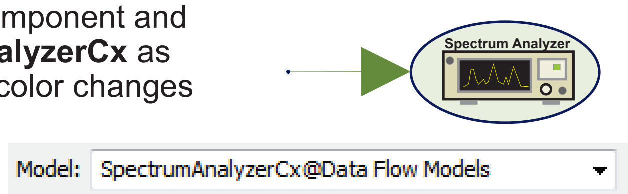
j. Edit the **CxtoEnv** component and edit the carrier frequency **Fc** to **10 GHz** (as shown above, to the right) and simulate again. Notice it is just as fast at the higher frequency.

NOTE: *In SystemVue we only sample the complex envelope of the modulate signal, and* *<u>not</u>* *the complete carrier. As a result, we achieve great reductions in analysis time. Also, all the RF characteristics of the models are retained and this is valuable for closely spaced interference issues, phase-noise effects, and BER testing.*

k. Select the **CxtoEnv** component and then click on the **Disable to short** icon – this effectively shorts out the component. Now, run the analysis again and you will see the errors.



l. Edit the **S2 Spectrum Analyzer** component and change the model to **SpectrumAnalyzerCx** as shown here. Notice that the arrow color changes from black to green to indicate a different input type of data.



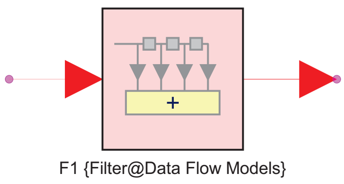
m. Run the analysis one more time and notice the errors do not appear and the data is the same. This simply shows another way to use the Spectrum Analyzer by changing the model to receive complex, baseband data.

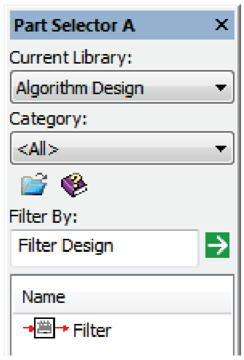
n. Delete the plot and the dataset – save the workspace again.

Now, it's time to use the Filter Designer.

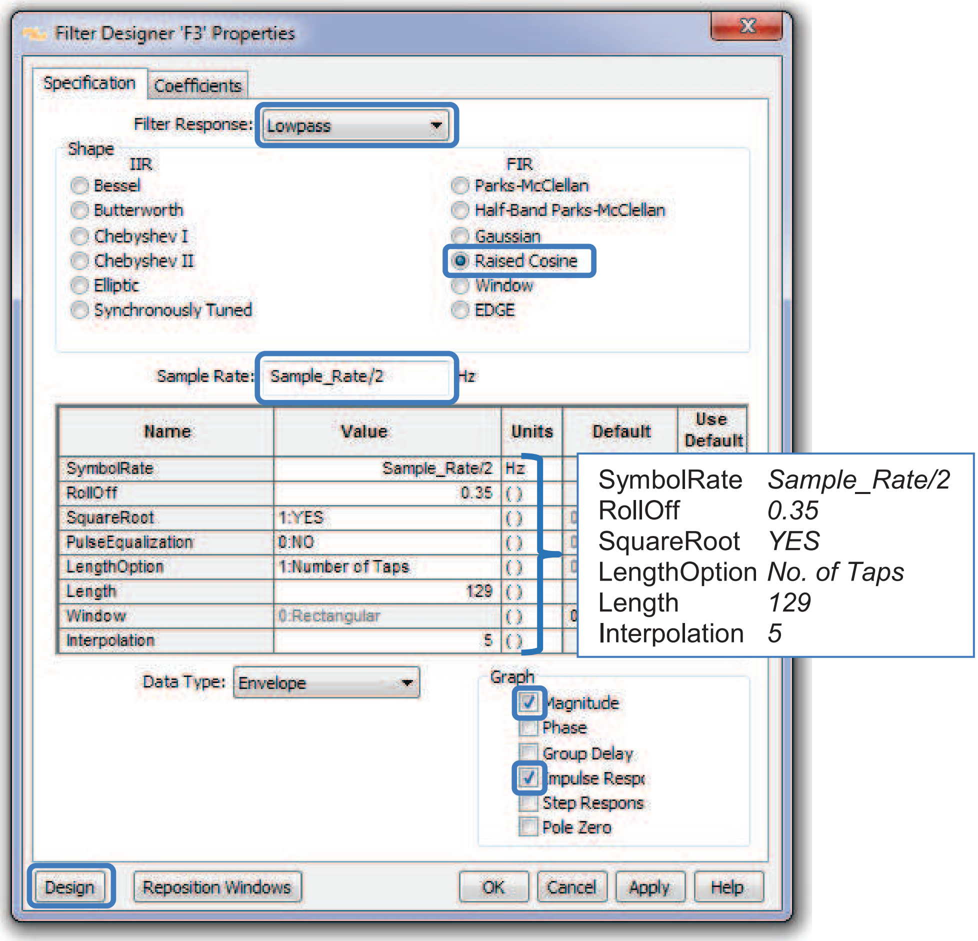
11. Using the Filter Designer

a. From the Part Selector, or hotkey **f,** to insert a **Filter Design** component into the Filter schematic as shown here – but do not connect it.

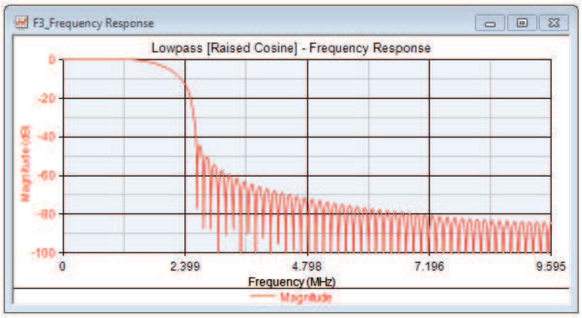


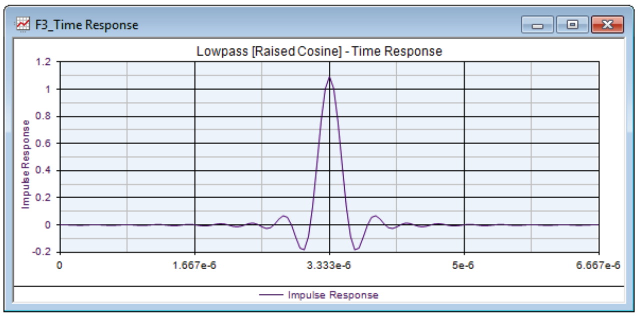


b. Edit (double click) the **Filter Designer** and specify a Lowpass Raised Cosine FIR filter with the parameters shown here. Once finished, click **Design** at the lower left and notice the filter response plots are updated.



c. Verify that your response looks like the plots shown here. If not, go back and check your filter parameters.

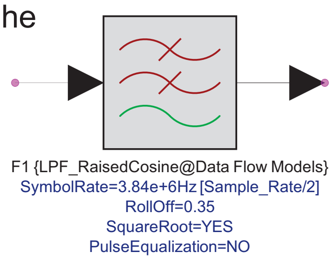




NOTE: *This filter will reduce the bandwidth but will still allow a reasonalbe value of inter-symbol interference. Of course, this is a trade-off and some of the filtering can be done in the receiver and the rest in the transmitter, as often specified by the system designer.*

Also note that the random bit source generates bits timed by the variable Sample_Rate (this is the global variable coming from the System Sample Rate *setting in the Data Flow Analysis). Because the Mapper set to QPSK produces one symbol per 2 bits in, the Symbol Rate of the filter is set to* Sample_Rate/2.

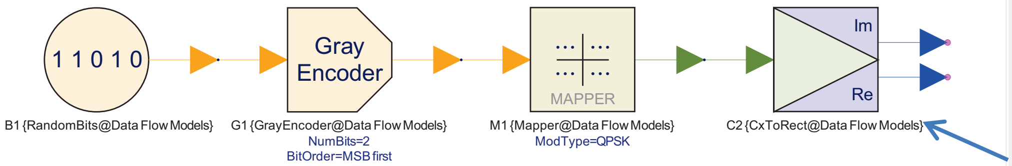
d. Accept these values and click **OK** in the filter designer. The two filter plots will automatically close. You will see the filter component change to the filter you designed.

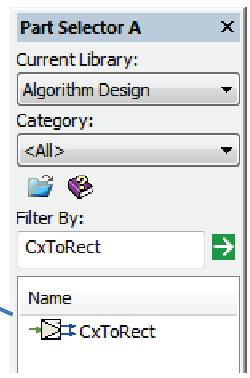


12. Real & Imaginary data with Filters

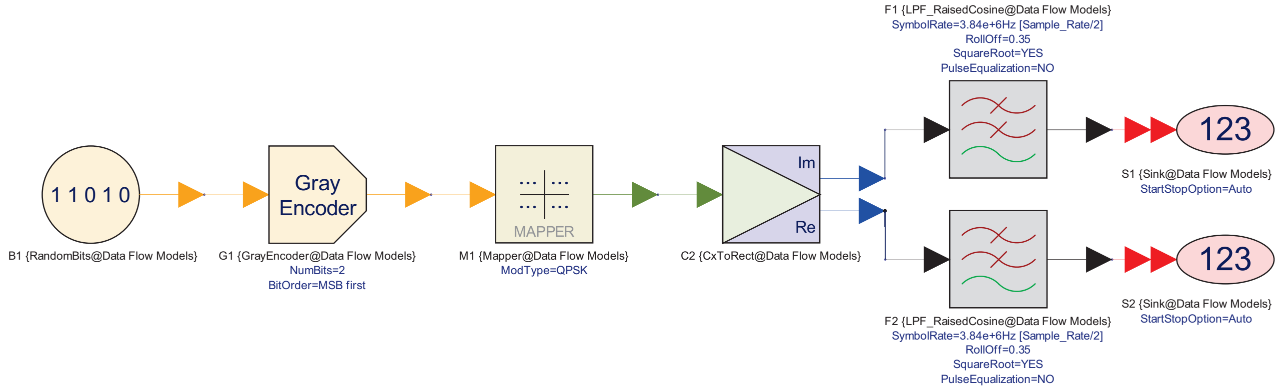
a. In the Filter schematic, **delete** the **CxtoEnv** component and the **Spectrum Analyzer** and their associated wires.

b. Delete the wire between the Mapper and the Sink and (from the Part Selector) insert a **CxtoRect** (complex to rectangular) component as shown here.





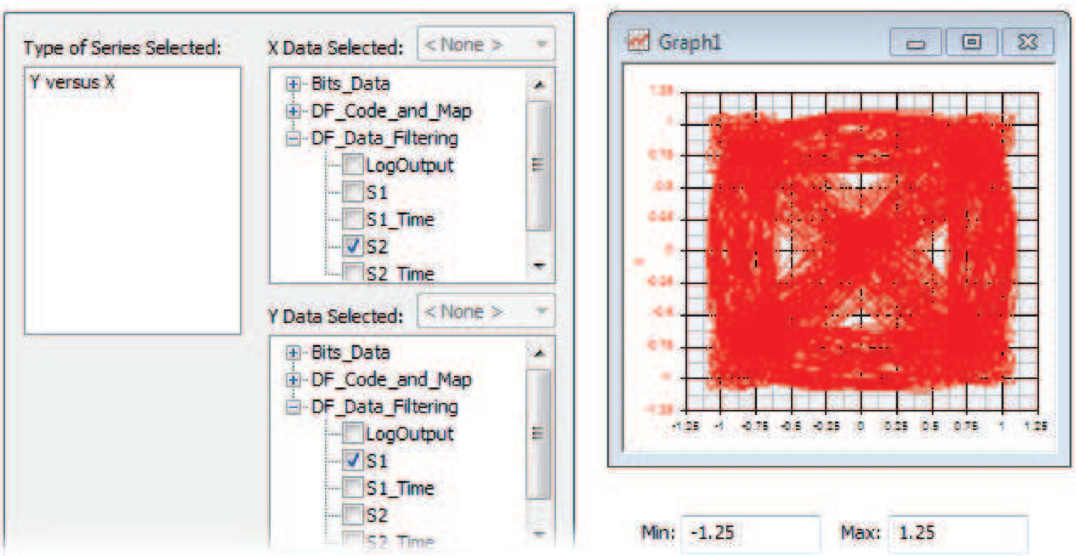
c. We do this, since we need to filter both I and Q separately. Copy the Filter you just created and also copy the Sink – then connect them as shown here. This represents a simple baseband section for a transmitter.



d. Right-click the **DF_Filtering analysis** controller and run the analysis again.

e. The dataset will now contain new data because of the two sinks and the different format for the data. Therefore, ignore any error messages and **delete** the existing graph.

f. Right-click on the **Filtering folder** and add a new graph: Select Type **Y versus X**. You need to select the correct dataset (**DF_Filtering**) and check the box S2 (**I**) for the X data and S1 (**Q**) for the Y data, as shown below (assuming the same Sink names). Also, turn off the autoscale for X and Y and set both axes to: **-1.25** to **1.25**. This will make the plot square if it is not. You should now see the X-Y or *Real* and *Imaginary* as shown here.



***NOTE**: You can right-click the trace and change the thickness or other attributes.*

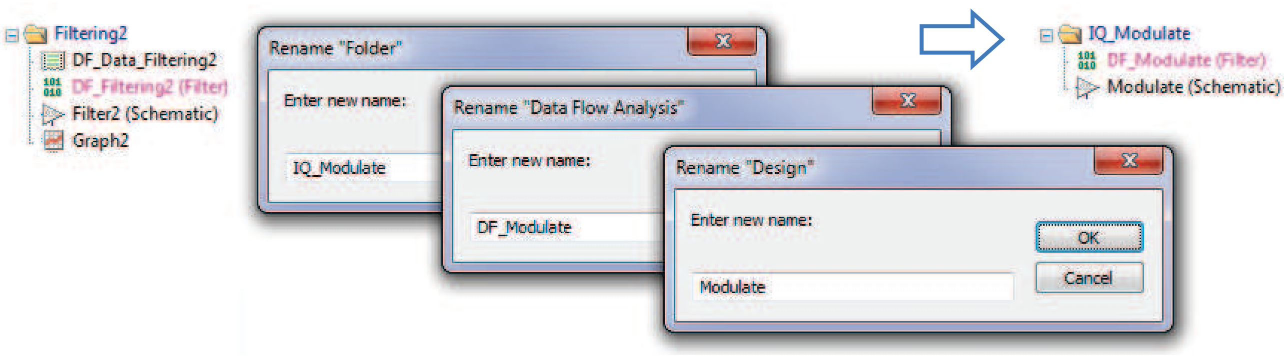
***Summary**: Now you should have a good level of skill using bit and sinewave signal sources with some basic components for viewing the spectrum and the constellaion, including using the filter designer. You should also have a reasonable working knowledge of the workspace: folders, plots, datasets, and analysis controllers.*

g. Save the workspace.

13. Modulating I & Q signals

a. Copy and paste the Filtering folder into the Designs main folder – it will become Filtering2 as shown below, to the left.

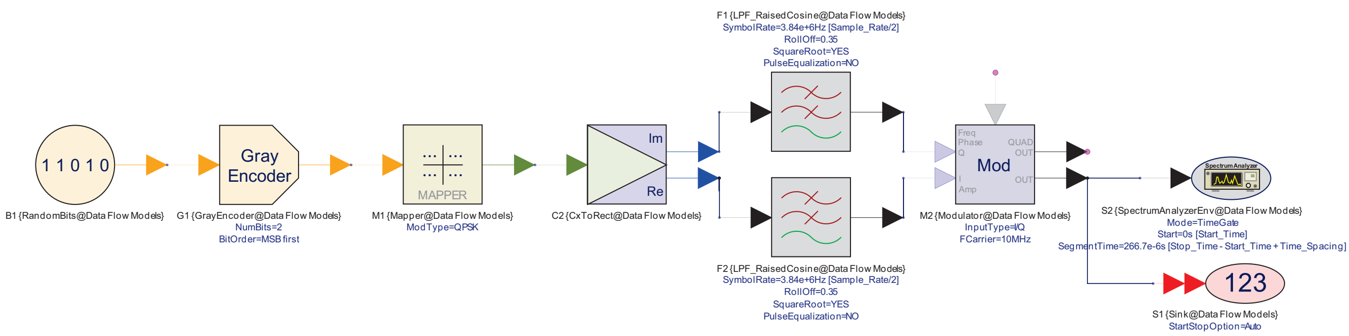
b. Next, rename the copied folder: **IQ_Modulate**, rename the Data Flow Analysis: **DF_Modulate** and rename the Design: **Modulate**. You can right-click each of these 3 items, and select Rename – it should be as shown below, right. (You can also **delete the old plots and datasets**)



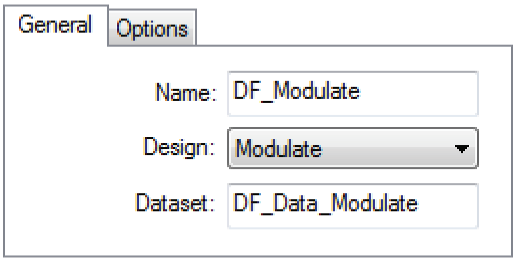
c. Modify the design: Delete both sinks and connect a **Modulator** from the Part Selector. Then, add a new sink and spectrum analyzer to the OUT pin of the modulator as shown here.

d. **Edit the Modulator** and set the carrier frequency Fc to 10e6 or 10 MHz as shown here.

Name	Value	Units
InputType	0:I/Q	( )
FCarrier	10	MHz

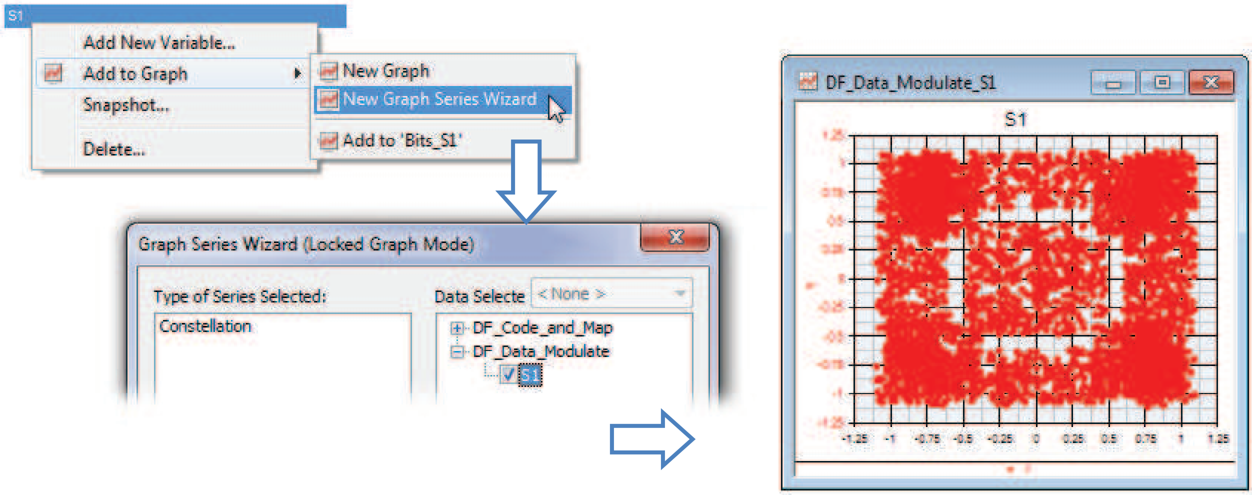


e. Edit the **DF_Modulate** analysis controller and set the Design to **Modulate**. Also, rename the dataset to **DF_Data_Modulate**. Whenever you copy an analysis, you need to set the names or it will use the copied names.



f. Run the analysis.

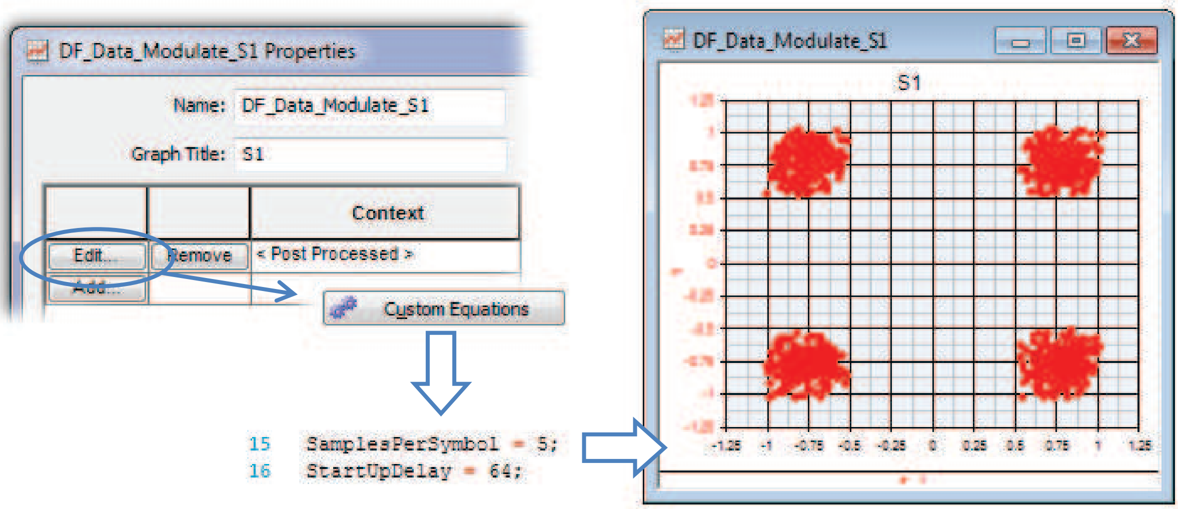
g. Open the dataset and add a new graph using the wizard – select a constellation of the S1 data from the sink, and click OK as needed.



Obviously, the data is not clean (it's fuzzy) – but there is a reason for this. The filter introduces delay and interpolation.

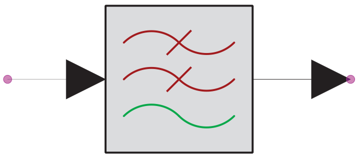
h. Double click the plot to bring up the S1 Graph Properties. Click the Edit button – then click the **Custom Equations** button (as shown below left).

Change the **SamplesPerSymbol** to **5** to match the interpolation and set the **StartUpDelay** to **64**. With 129 taps set in the filter, using one-half that value (-1 for the middle) should results in the correct delay. Click OK as needed and you will see the improved results as shown here.



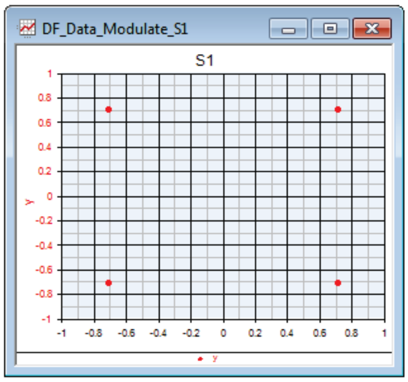
Still, the constellation could be even better. Why? There is one more adjustment for an almost perfect constellation – we have no receive filter yet to match the *root raised cosine* of the transmit filter.

i. Edit **both** filters in the schematic and change the **SquareRoot** parameter to **NO** as shown here. Click OK as needed.

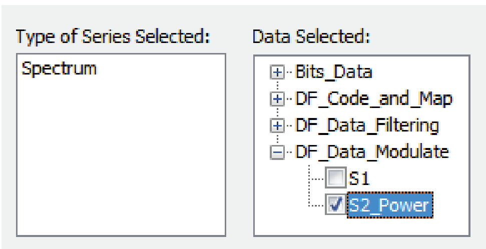


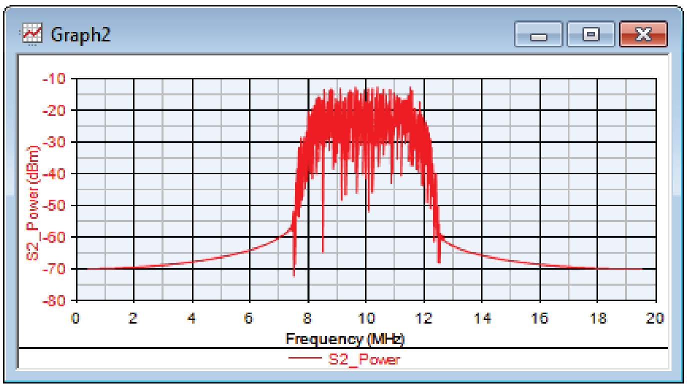
Name	Value	Units
Loss	0	( )
SymbolRate	Sample_Rate/2	Hz
RollOff	0.35	( )
SquareRoot	0:NO	( )
PulseEqualization	0:NO	( )
LengthOption	1:Number of Taps	( )
Length	129	( )
Window	0:Rectangular	( )

j. **Run the analysis** again. Look at the constellation now – you can remove the auto scaling and set both X and Y to 1- and 1 respectively. As you can see, the results are now perfect peak values of the QPSK signal in each quadrant.



k. Add a final plot of the spectrum to the folder using the wizard to plot the spectrum of S2_Power as shown here. Name the plot Spectrum - this validates the response in the frequency domain for the 10 MHz carrier.





NOTE*: You can set the X axis to MHz and the Y to dBm. Also, if you have named the Spectrum Analyzer component something other than S2 (default) it would be easier to identify (you did this in the earlier lab exercises.*

<u>END OF LAB EXERCISE</u>

LAB EXERCISE 4

BER, EVM and Sub-circuits

This lab exercise introduces cross correlation, EVM and BER measurements and shows how to create sub-circuit models.

Prerequisite: Lab Exercise 3 (workspace is required).

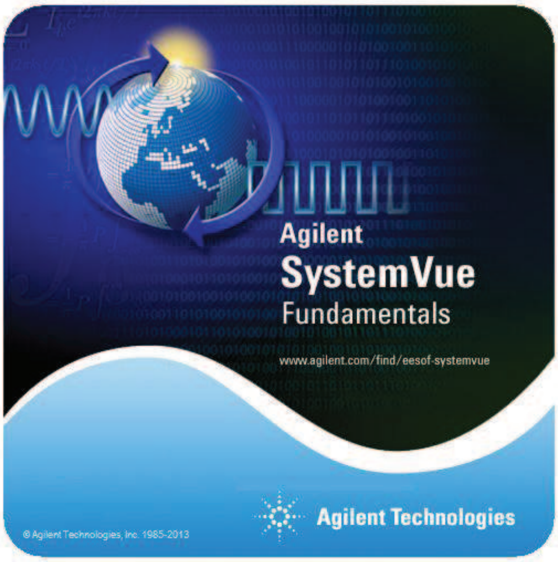


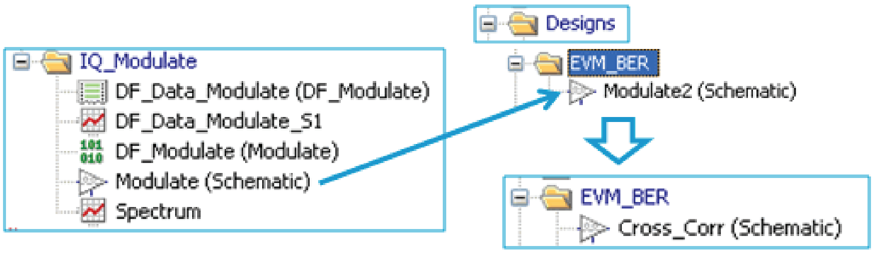
Table of Contents: Lab 4

Lab 4: EVM and BER Basics

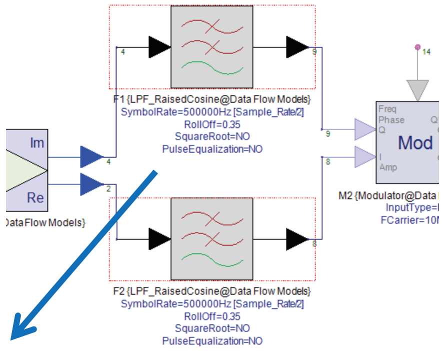
1. Build the receiver section

a. In the Designs folder of the open **PHY_Design** workspace, **add a new folder** and name it: **EVM_BER**.

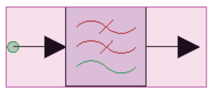
b. From the IQ_Modulate folder, **copy** the **Modulate schematic** into the new folder (EVM_BER) and rename it: **Cross_Corr** as shown here.

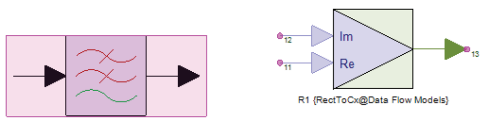


c. Open the schematic Cross_Corr and delete the SpectrumAnalyzer and the Sink. Then, **copy the two filters** – select the filters and then right-click to copy and paste.

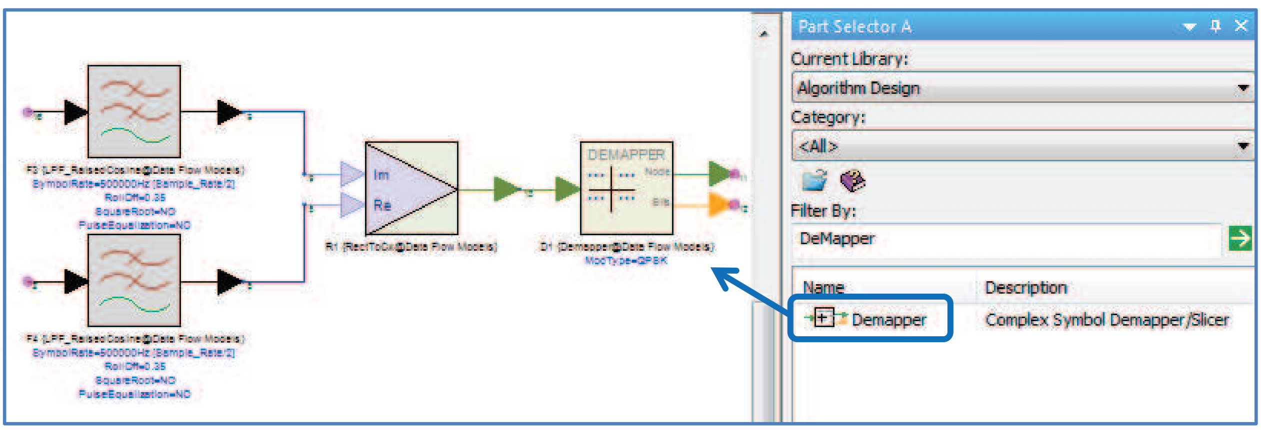


Next, place a **RectToCx** component from the Part Selector and place it to the right of the copied filters, as shown here.

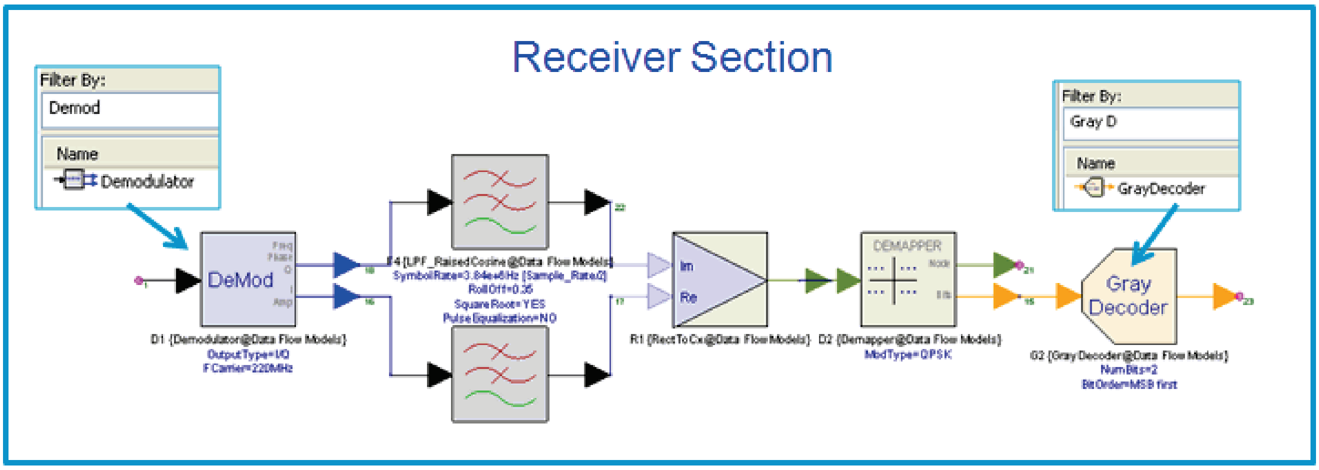




d. Connect the RectToCx to the filter outputs and, from the Part Selector, insert a **Demapper**, like this:

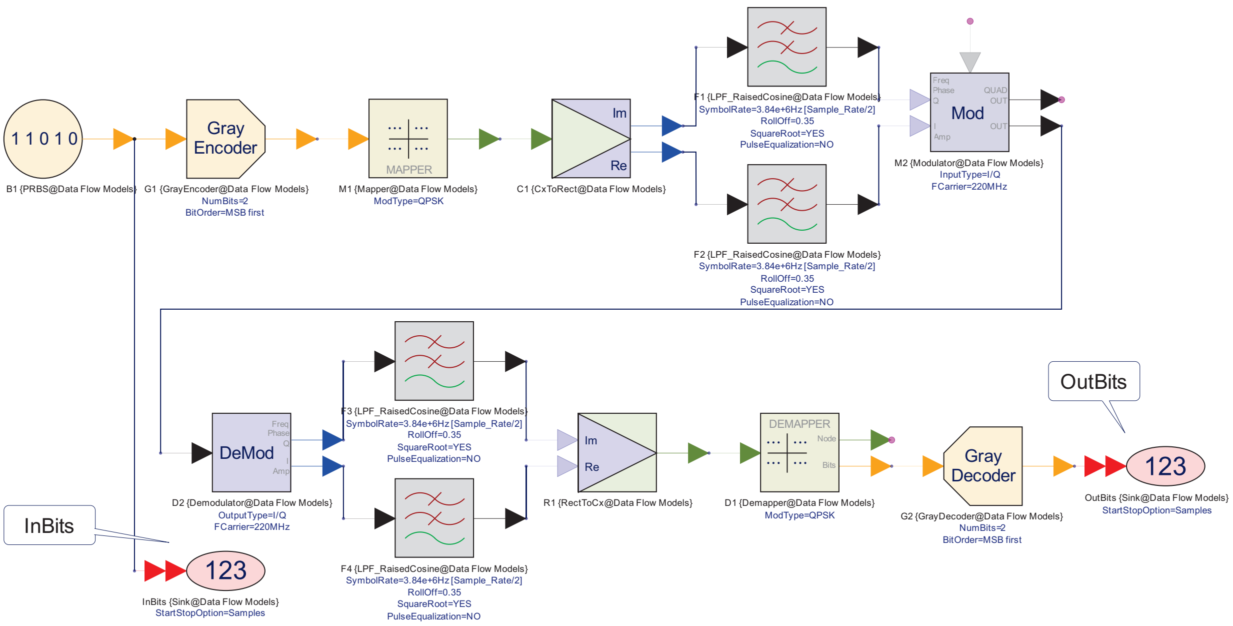


To complete the receiver section, insert a **Demodulator** and a **Gray Decoder** from the Part Selector, and connect them as shown here.



2. Connect Transmitter to Receiver

a. Connect the **OUT** pin of the transmitter **modulator** to the to the input of the Demodulator as shown here.



b. Insert two new **Sinks** (use keyboard key S). Edit them and name them **InBits** and **OutBits** as shown here.

The next steps will be to specify the component parameter settings,

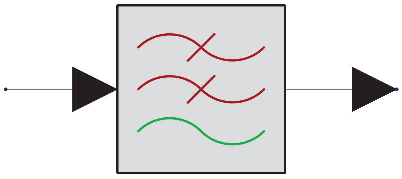
3. Component Parameter Settings

a. First, edit the two **transmitter** filters and reset the **SquareRoot** value to **YES**.

b. Then, edit the two **receiver** filters and also here set **SquareRoot** to **YES**. Next, set **Interpolation** to **1**, **Decimation** to **5** and **DecimationPhase** to **3** (this selects which of the 5 incoming samples will represent the decimated, or down-sampled, output). All the other settings should be the same as before – shown here. The decimation on the receiver is required (if we are not using a downsampler).

Transmit Filter

Name	Value
Loss	0
SymbolRate	Sample_Rate/2
RollOff	0.35
SquareRoot	1:YES
PulseEqualization	0:NO
LengthOption	1:Number of Taps
Length	129
Window	0:Rectangular
Interpolation	5
Decimation	1
DecimationPhase	0
InterpolationScaling	1:YES



Receive Filter

Name	Value
Loss	0
SymbolRate	Sample_Rate/2
RollOff	0.35
SquareRoot	1:YES
PulseEqualization	0:NO
LengthOption	1:Number of Taps
Length	129
Window	0:Rectangular
Interpolation	1
Decimation	5
DecimationPhase	3
InterpolationScaling	1:YES

c. Edit both the **Modulator** and **Demodulator** and set **FCarrier** to **220** MHz on each of them.

Name	Value	Units
FCarrier	220	MHz

d. The **Mapper** and **Demapper** should already be **QPSK**. The **Gray Encoder** and **decoder** should be have **NumBits** set to **2** and **BitOrder** set to **1:MSB first**.

e. Reset the **PRBS** bit souce SampleRateOption to **0:UnTimed**.

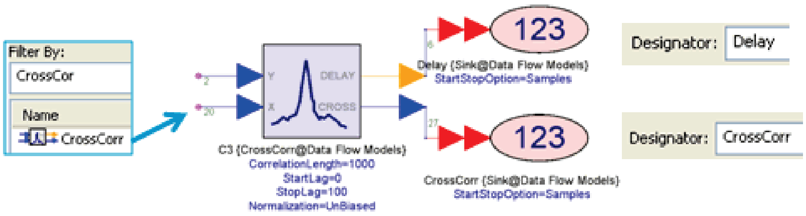
Name	Value	Units
SampleRateOption	0:UnTimed	( )

f. Check the all settings and connections to be sure they are correct.

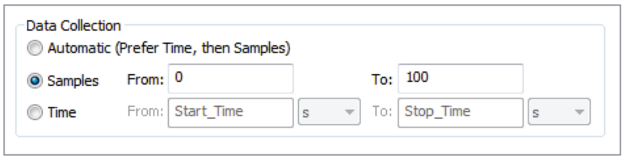
The final steps will be to insert the Cross Correlation component and set up the analysis, etc.

4. Cross Correlation Setup and Analysis

a. From the Part Selector, insert a **CrossCorr** measurement near the bottom of the receiver – also add two more **Sinks** as shown here.



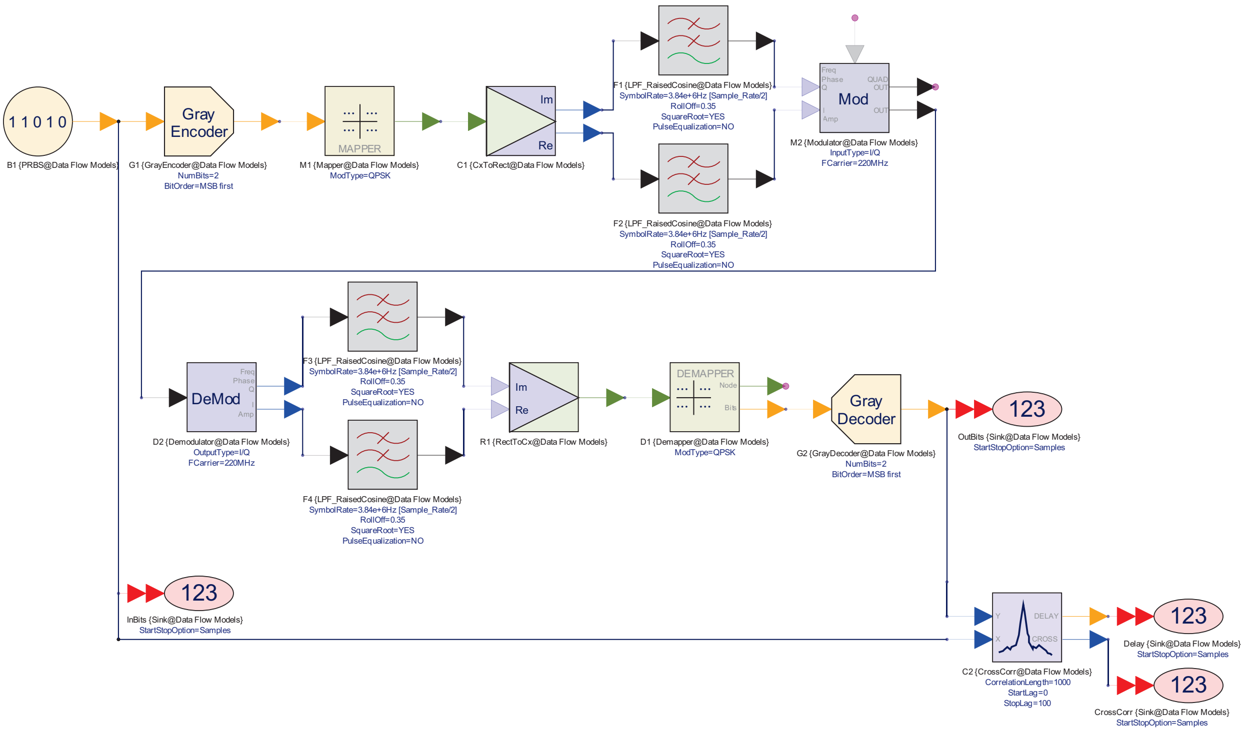
b. Name the Sinks **Delay** and **CrossCorr** and connect them to the respective CrossCor outputs.



c. **Set all four Sinks**: **Data Collection** to **Samples** from **0** to **100** as shown here. This will make it easy to plot the results instead of using the Num_Samples.

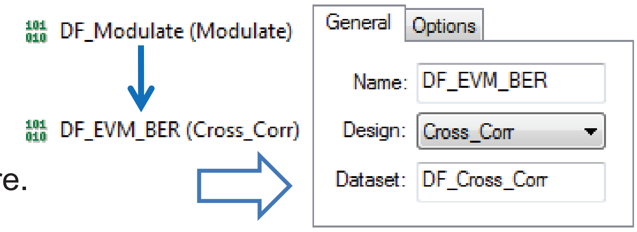
d. Wire the **CrossCorr X** pin to the **InBits** node and wire the **Y pin** to the **OutBits** as shown here. Now, set the **CorrelationLenght, StartLag, StopLag** and **Normalization** as shown here. This shows the completed schematic:

Name	Value	Units
CorrelationType	0:NonCircular	( )
CorrelationLength	1000	( )
StartLag	0	( )
StopLag	100	( )
Normalization	1:UnBiased	( )



e. **Copy** the **DF_Modulate** analysis from the IQ_Modulate folder into the **Cross_Corr folder**.

f. Edit the analysis and rename it: **DF_EVM_BER**. Then select the **Cross_Corr design** and name the dataset **DF_Cross_Corr,** as shown here.



g. Click OK – all the timing and sampling settings can remain as before.

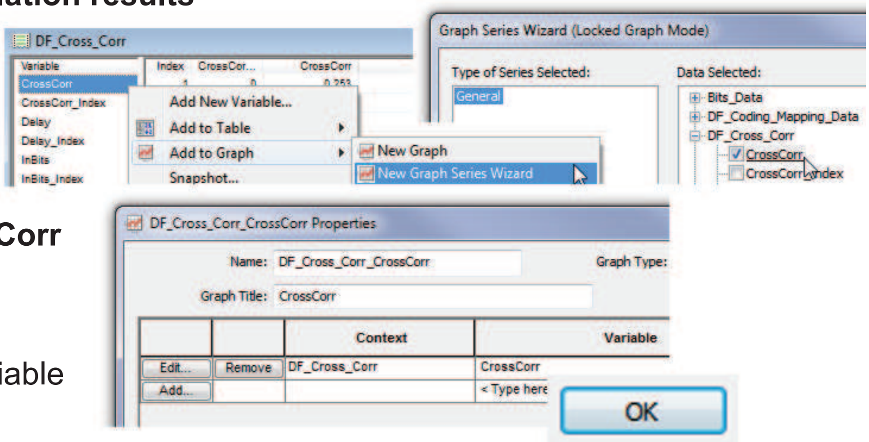
h. Right-click and run the DF_EVM_BER analysis. If you have any errors, go back and check your work – there should be no errors with the correct setup.

5. Delay and Cross Correlation results

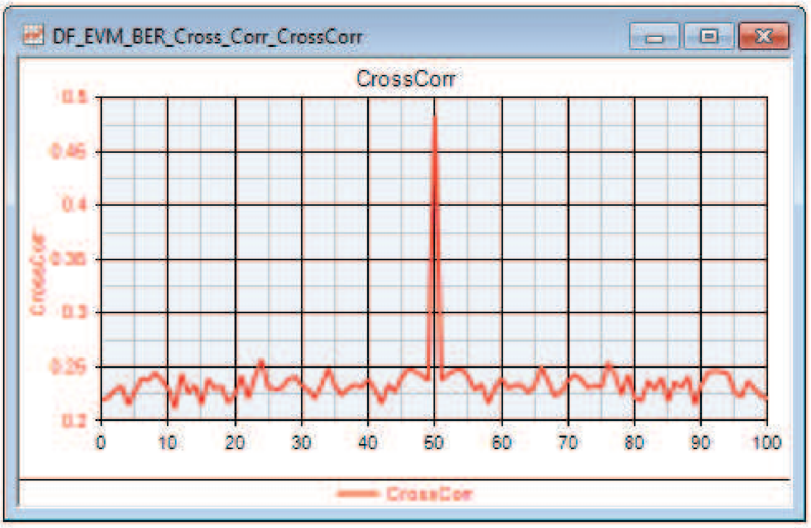
a. **Open the dataset** and then **right-click** on the **CrossCorr** variable.

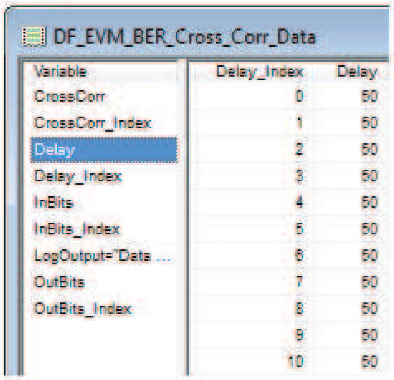
Then select Type Series **General** and data **CrossCorr** and click **OK** as needed to create the plot.

Also check the **Delay** variable in the dataset.

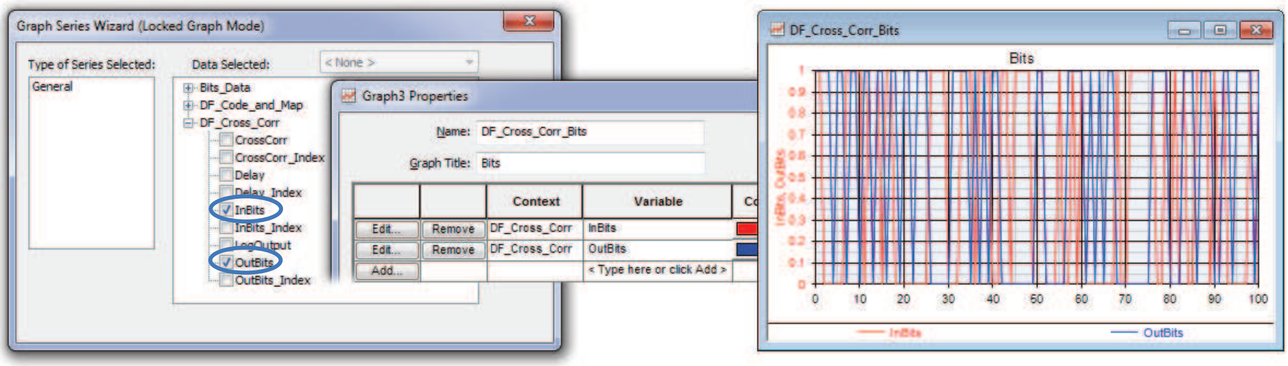


b. Your plot and the Delay data should both show 50 bits bits delayed between the input and the output. The Cross Correlation is about 0.5. If the BER were zero, then the Cross Correlation would be 1.

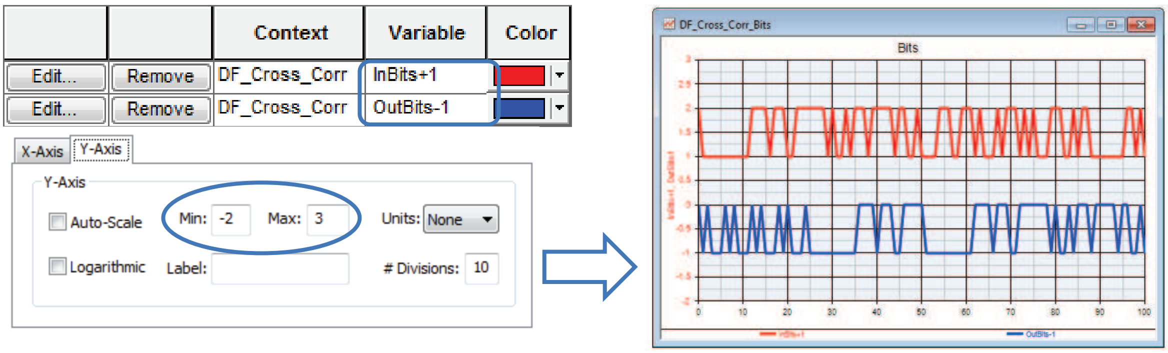




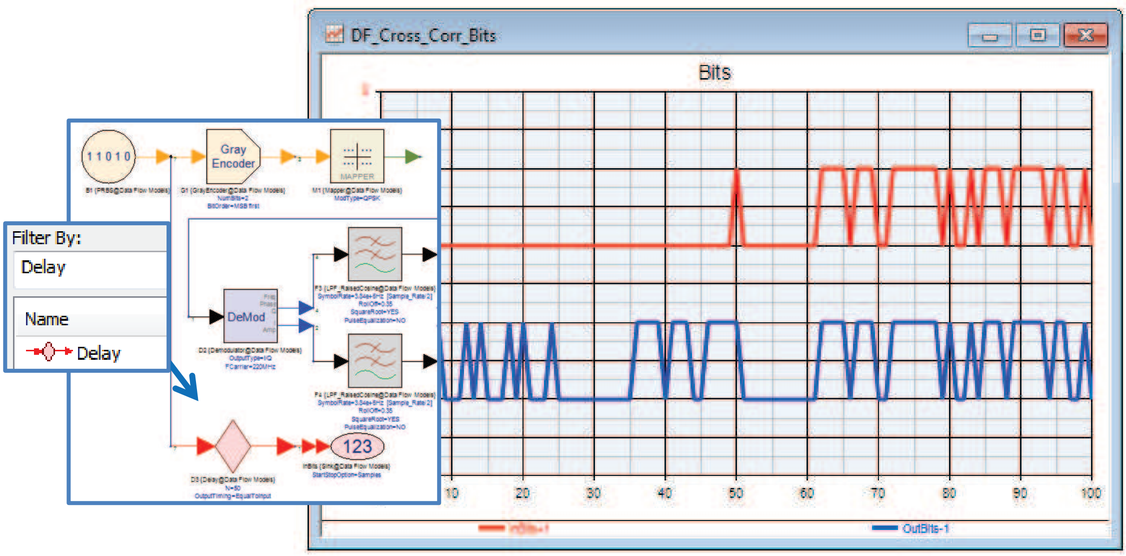
c. Add another plot from the dataset using the wizard – plot the **InBits** and **Outbits** as shown here. Notice that the bits overlap – you will fix this next.



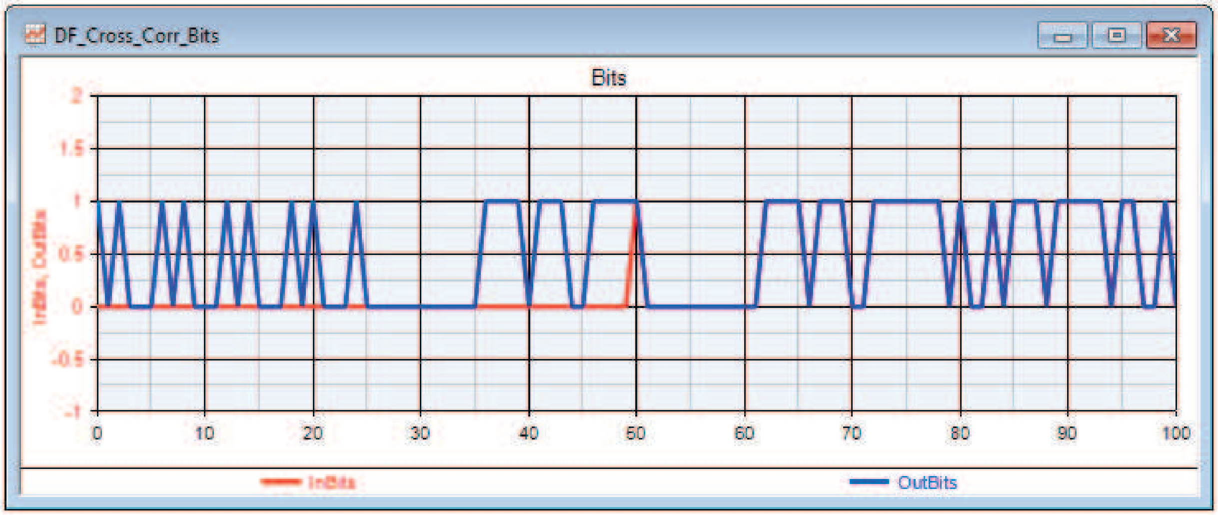
d. **Edit the plot** and name it: **DF_Cross_Corr_Bits** and title it as shown here. Also, offset the traces on the Y axis by by adding **+1 to InBits** and **-1 to Outbits** in the Variable field – also set the **Y axis Min** and **Max to -2** and **3** and click OK. Your plot should now show the bits data clearly.



e. Go back to the **schematic** and insert a **Delay** in front of the **InBits** sink. Set **N** to **50** as shown here. **Run the analysis again**. Now you can see the alighment with the added delay (50) determined from the Cross Correlation.



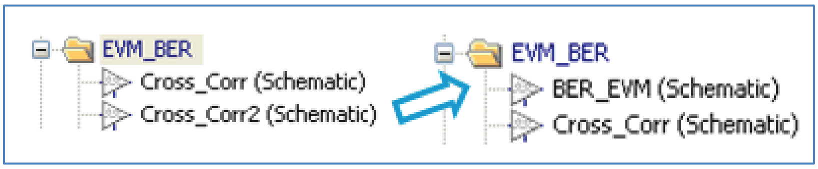
f. Go back to the plot one last time and **remove the offset of -1 and +1**. Also, rescale the **Y axis from -1 to 2** as shown here and **thicken the InBits trace**. Now, after the 50 sample delay, the traces are aligned on top of each other as you can easily see here.



g. Close all the open windows ans Save the workspace. The next steps will be to compute EVM and BER.

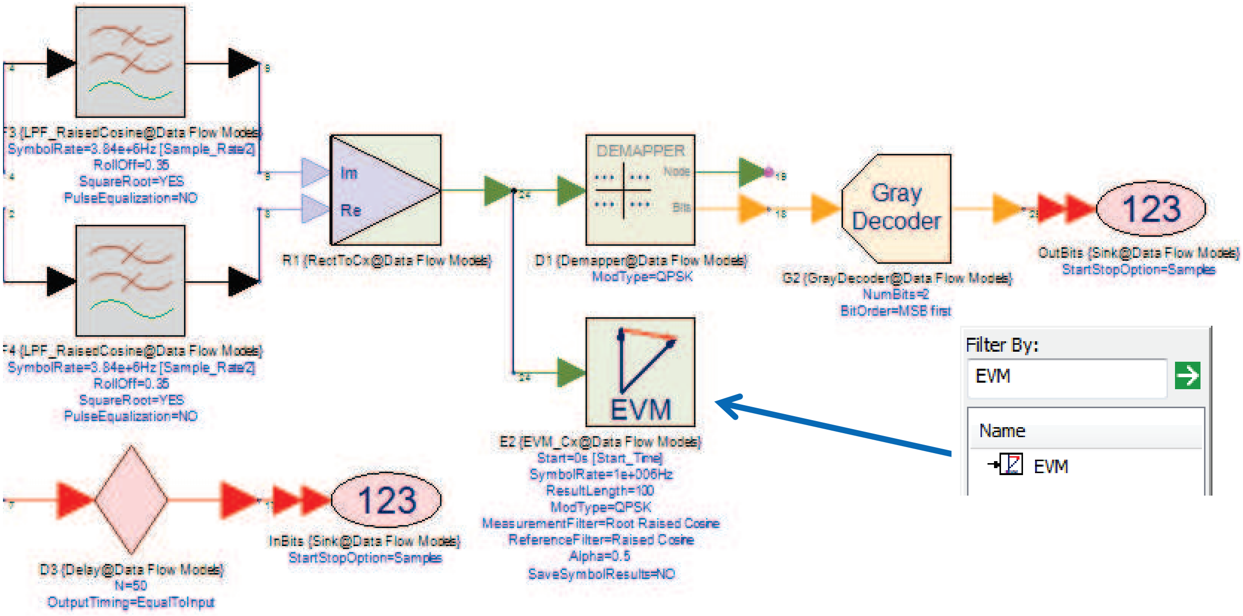
6. BER and EVM Schematic Setup

a. Copy the Cross_Corr schematic and rename it BER_EVM.

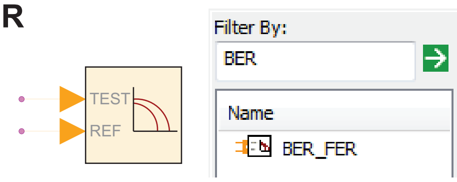


The next steps will be to reconfigure the schematic for BER and EVM. Afterward, you will set up the analysis.

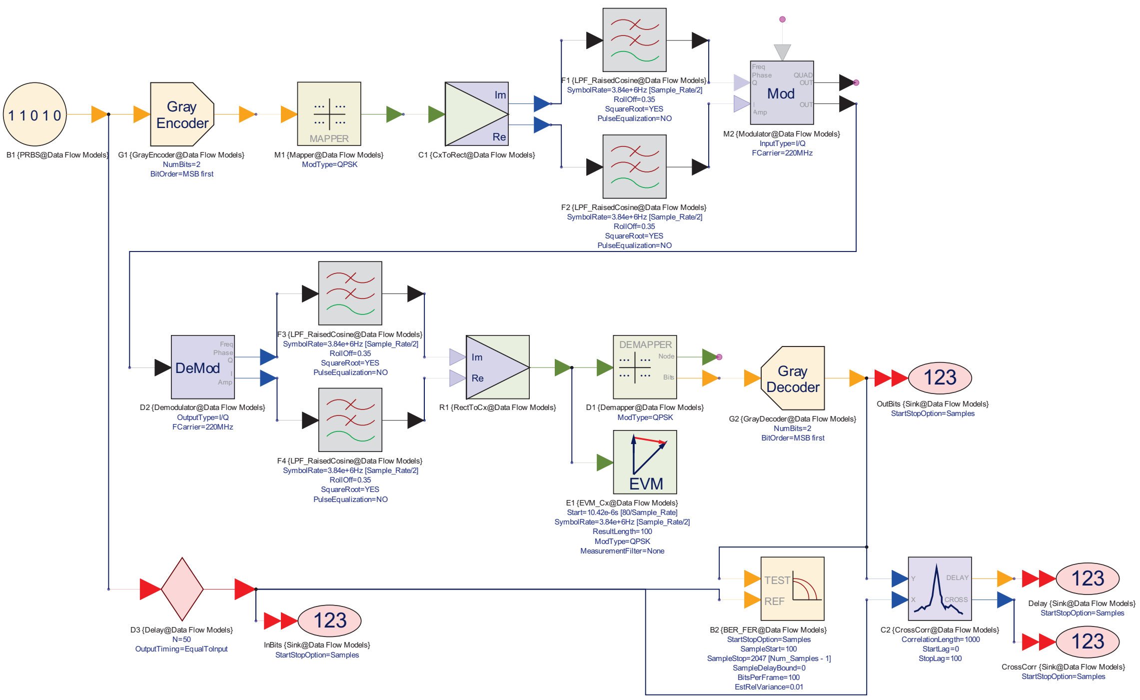
b. Open the **BER_EVM** schematic. Use the Part Selector to add an **EVM** sink, connected as shown. Make sure to change the Model parameter from EVM_Env to **EVM_Cx@Data Flow Models** as SystemVue supports complex baseband, or zero-IF (note the pin change to complex green).



c. Insert the final component for the analysis: **BER_FER** and wire it to the **Delay** and **BitsOut** as shown here. This is the finished schematic. Make sure the *Delay* is placed as shown below. Next, you will set the componets and analysis.

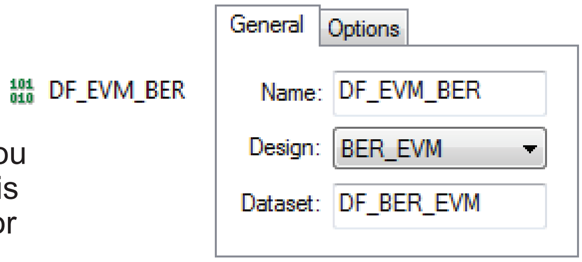


BER and EVM Schematic



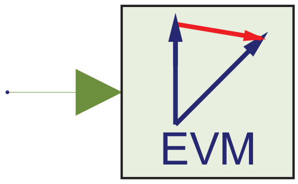
7. BER and EVM Analysis and Results

a. **Edit the DF_EVM_BER analysis** and select the **BER_EVM** design and name the dataset **DF_BER_EVM** as shown here. This means you use the same analysis but you specify a different design and dataset – this is efficient for many designs that use the same or similar analysis settings.



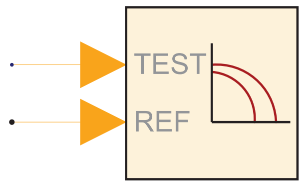
b. Edit the **EVM** component and set the values as shown here – the results from this component will be available in the dataset. The Start numerator 80 is taken from the number of samples obtained from the delay (50) in the previous steps and additional 30 to ensure the impulse response in the filters has died. And again, the SymbolRate is ½ the sample rate of the analysis, as we map 2 input bit samples to one complex QPSK symbol.

Name	Value	Units
Start	80/Sample_Rate	s
SymbolRate	Sample_Rate/2	Hz
ResultLength	100	( )
ModType	1:QPSK	( )
MeasurementFilter	0:None	( )
ReferenceFilter	3:Raised Cosine	( )
Alpha	0.5	( )
SaveSymbolResult	0:NO	( )

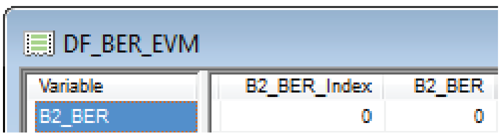


c. The last component to edit is the **BER_FER** – the results will be available in the dataset. Set the **StartStopOption** to **Samples**, set **SampleStart** to **100.** The other values are defaults, i.e. **Stop** should be **Num_Samples-1**.

Name	Value	Units
StartStopOption	1:Samples	( )
SampleStart	100	( )
SampleStop	Num_Samples - 1	( )
SampleDelayBound	0	( )
BitsPerFrame	100	( )
EstRelVariance	0.01	( )
StatusUpdatePeriod	1000	( )

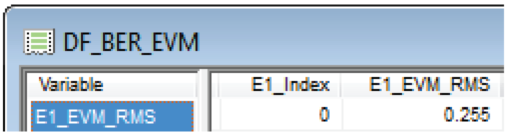


d. **Run the analysis**. When it is finished, **Open** the dataset **DF_BER_EVM** and verify the results as shown here. If you set up the component parameters and analysis correctly, you should see a zero BER (all bits recovered without errors).



***NOTE**: Your component name may differ from the B2 name used here.*

Also, in the dataset, the EVM [%] (E1_EVM_RMS in the dataset) is only a small residual value.

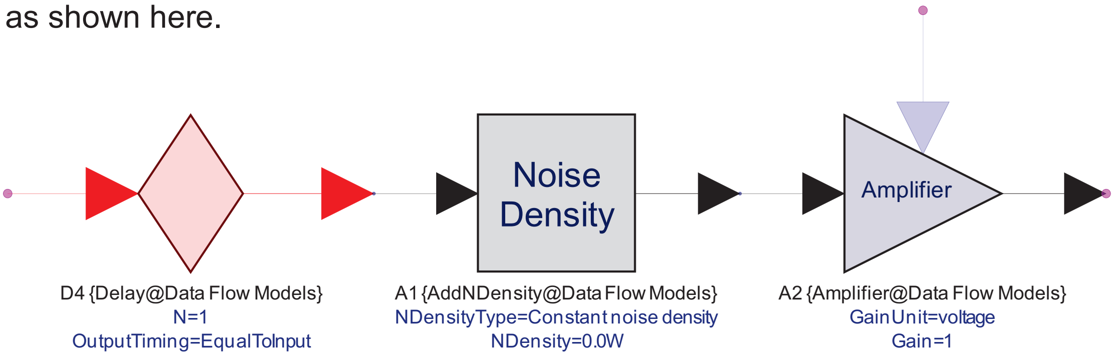


e. **Save** the workspace.

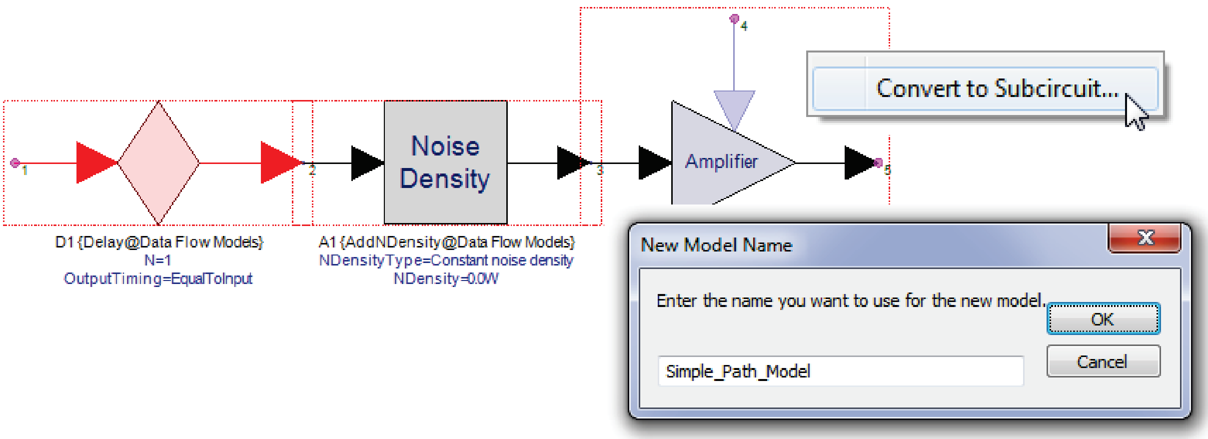
8. Create a Subcircuit (path/channel model)

SystemVue has some built-in path or channel models. However, this QPSK receiver is not capable of synchronizing to the input to correct for the phasing. Therefore, you will create a simple subcircuit to characterize the channel effects – this model will also have parameters that can be tuned or adjusted.

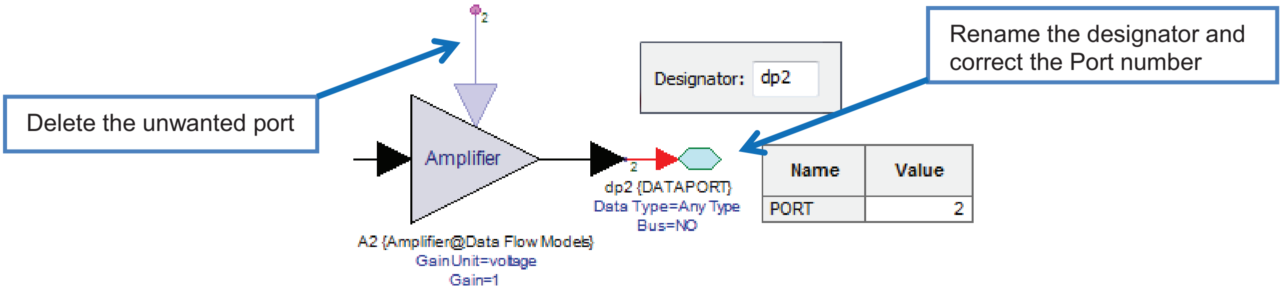
a. In an open area of the schematic, insert the following three components from the Part Selector and connect together: **Delay, AddNDensity**, and **Amplifier** as shown here.



b. Use your keyborad **Ctrl** key to select all three components together - then right-click and select **Convert to Subcircuit** as shown here. **Type in a name: Simple_Path_Model** and clickOK to create the model or subcircuit.



c. When the Simple_Path_Model window appears, **delete the unwanted data port** from the amplifier as shown here, as we don't need it for our model.



d. Also, edit the other amplifier ouput dataport and rename it dp2 and port 2 as shown – click OK. It is recommended that you specify names and designators appropriately – this makes it easier to troublehoot and work with your designs.

e. Notice the new model icon in the workspace tree. Simple_Path_Model (Model)

f. Back in the BER_EVM schematic, delete the 3 original components (Delay, Noise and Amplifier) – you no longer need them now that you have a model.

g. **Save** the Workspace.

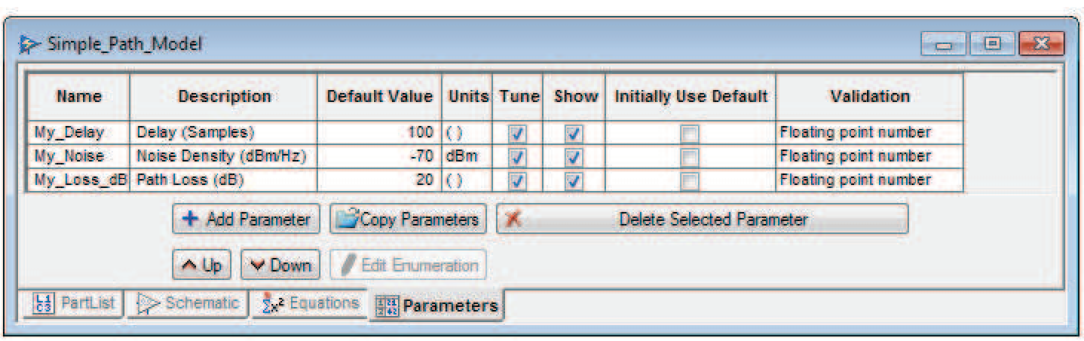
Next, you will set up the model parameter values with variables.

9. Subcircuit Parameters and Equations

This step shows how to setup parameters that can be passed into the model for use at the higher level design. In this case, you will set up parameters for the delay, noise, and loss through the simple path model.

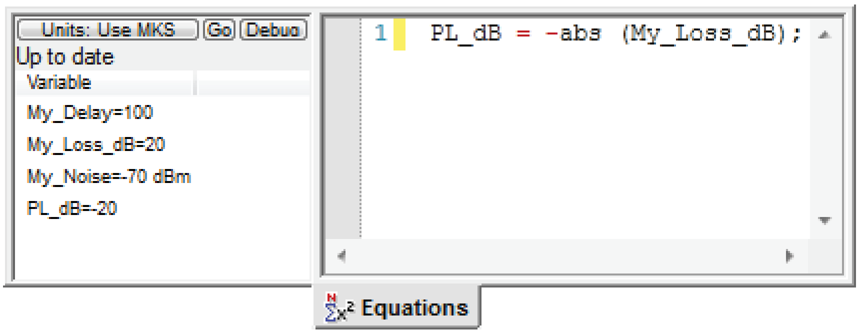
a. Open the new schematic **Simple_Path_Model**, go to the bottom of the model window and click the **Parameters** tab. This is where you specify parameters for the model – it is like delcaring variables or equations that you can pass.

b. Click **Add Parameter** and insert your cursor in the Name field and type in **My_Delay** as shown here. Then type in the Description: **Delay (Samples**) and set the value to **100**. Check the boxes for Tune and Show.



c. Use the **Add Parameter** button again as needed to specify the other two parameters exactly as shown here: **My_Noise** and **My_Loss_dB**. The Description is not requried but it is recommended to make it easy to identify the parameter and its units.

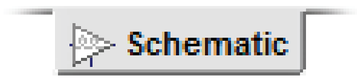
d. Check all three parameters and be sure they are typed in as shown. When they are complete, click the **Equations** tab. On the right side, type in the equation: **PL_dB = -abs(My_Loss_dB)** as shown here. Press **GO** and they should appear as shown here.



***NOTE**: If the values are not correct, go back to the Parameters and re-enter them or try the Go button here.*

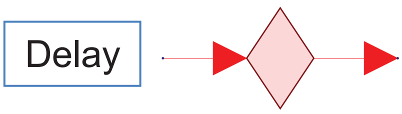
Now that you have declared these values, the next step is to assign them.

e. At the bottom of the model window, select the **Schematic** tab.



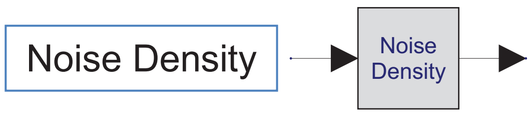
f. Edit the **Delay** and set the **N** value to the parameter **My_Delay** as shown here.

Also, check the boxes for Tune and Show and click OK.



Name	Value	Units
N	My_Delay	( )
OutputTiming	0:EqualToInput	( )

g. Edit the **Noise Density** component and set the **NDensity** to the variable: **My_Noise** with the **Units** in **dBm** as shown. Also, check the boxes for Tune and Show and click OK.

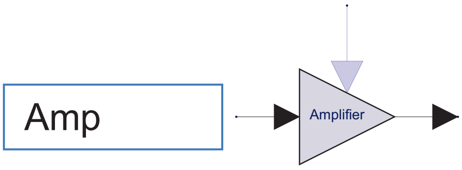


Name	Value	Units
NDensityType	0:Constant noise density	( )
NDensity	My_Noise	dBm
RefR	50	Ohm

h. Edit the **Amp** and set the GainUnit to **dB** as shown here. Also, type in the **Gain** which is path loss in dB passed from the paarameter equation you wrote:

PL_dB

Also, check the boxes for Tune and Show and click OK

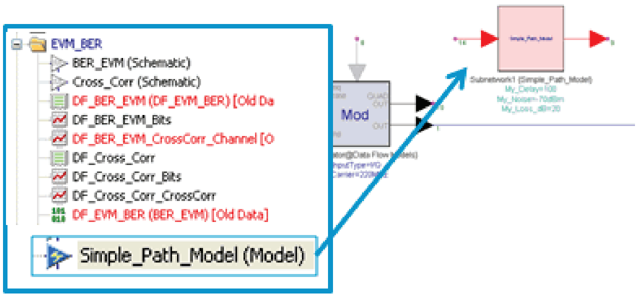


Name	Value	Units
GainUnit	1:dB	( )
Gain	PL_dB	( )
NoiseFigure	0	( )
GCType	0:none	( )
RefR	50	Ohm

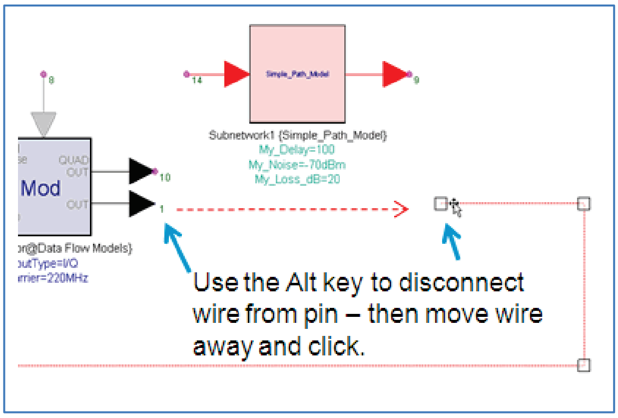
i. Check all you work, then **Save** the workspace and close the model window. The next step will be to inser thte model in the top level design: BER_EVM.

10. Inserting the Subcircuit model

a. With the BER_EVM schematic open, click on the model **icon**, Simple_Path_Model, and drag it into the schematic – insert upper right of the modulator as shown here.

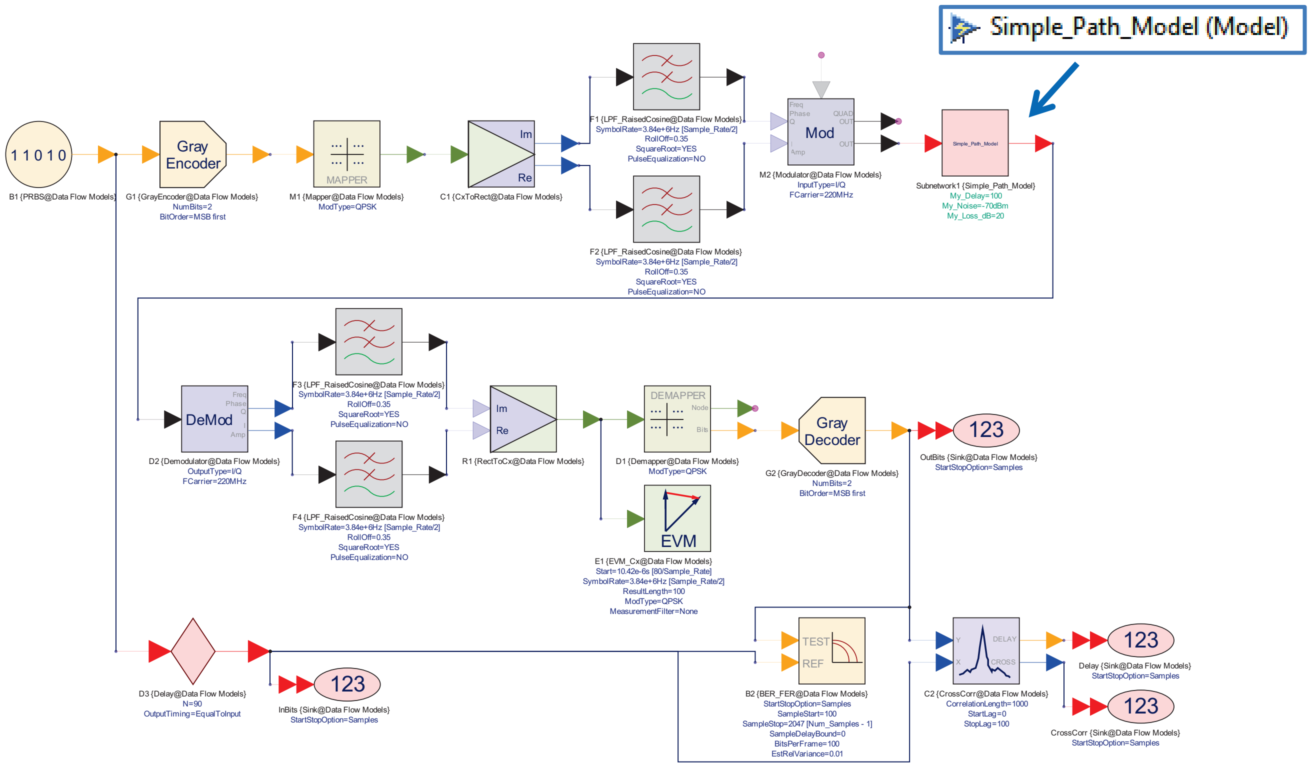


b. Use one hand to press the keyboard **Alt** key and the other to **drag the wire** away from the modulator OUT pin as shown here. Leave enough room for the model to be inserted.



NOTE: *If you have any trouble, use the Undo icon and try again*.

c. When ready, connect the model between the modulator OUT pin and the existing wire that connects the the demodulator as shown here.

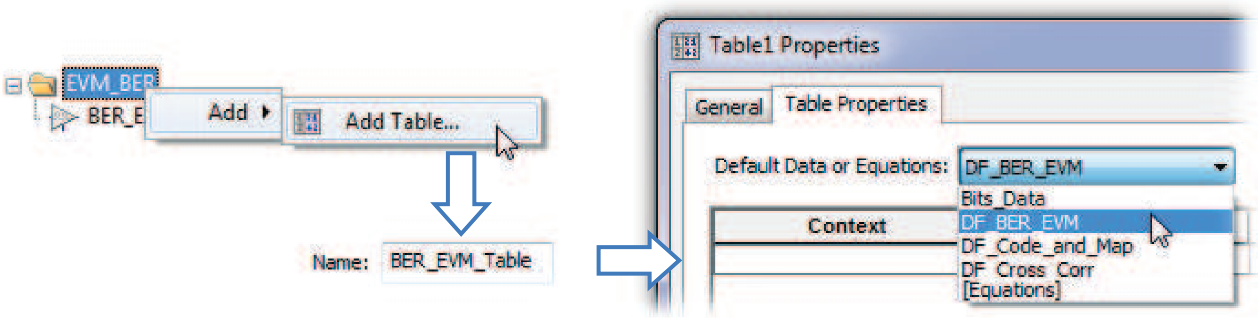


d. Save the workspace again.

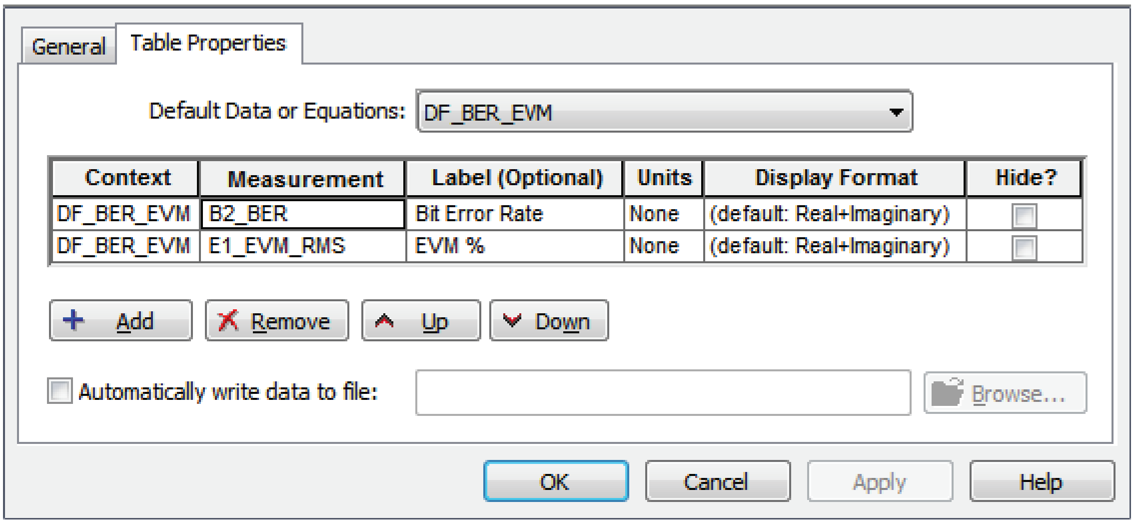
11. Adding a Table to View Results

a. **Run the analysis** – this will give the results with the path model and its initial settings. Next, you will set up a table and a slider to easilil the effects of the added path model to the design.

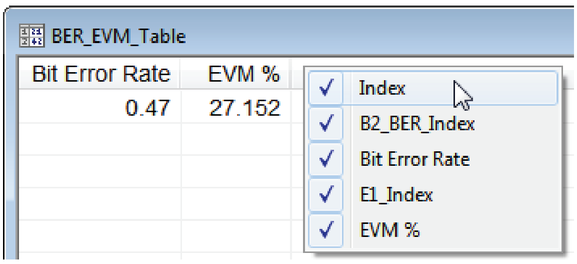
b. In the workspace tree, **add a table** to the EVM_BER folder as shown here and type in a **name** for the table: **BER_EVM_table**. In the **Table Properties** tab, select the dataset: **DF_BER_EVM** as shown here – click OK as needed.



c. Also in the **Table Properties**, add (type in) the two measurements and labels shown here: **B2_BER** labeled **Bit Error Rate** and **E1_EVM_RMS** labeled **EVM %**. Remember that B2 and E1 are only component designators, which in turn are appended by the resulting measurement – click OK.



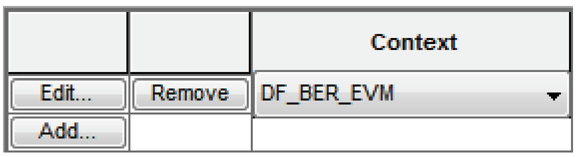
d. The new table should look like the one shown here with BER and the % EVM – easy to read (your EVM% may vary slightly). Try moving the panels, or right click to turn off index values, to hide them. You size the table also and SystemVue will keep it that way even after you close it.

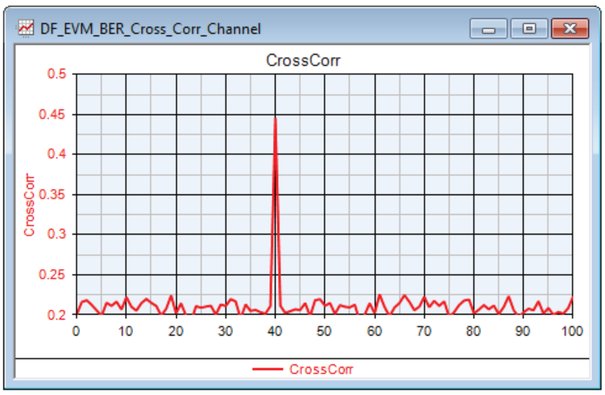


e. Save the workspace again.

12. Cross Correlation Results with the Path Model

a. In the Workspace Tree, make a copy of your **cross correlation** plot and rename it **DF_BER_EVM_CrossCorr_Channel**. Go to it's Properties dialog and change the **Context** from **DF_Cross_Corr** to **DF_BER_EVM**. Click OK.



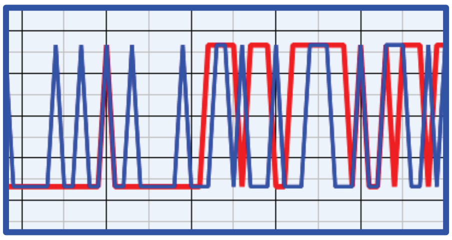


You should see the peak value at **40** samples now that the model delay has been added.

Now, here's a question: Can you figure out why the delay now is at additional 40 samples? *(Hint: Remember the Tx Filters upsample/interpolate. Ask your instructor if you're stuck.)*

Also, make a copy of your DF_Cross_Corr_Bits plot, name this DF_BER_EVM_Bits. Notice the Bits are no longer aligned

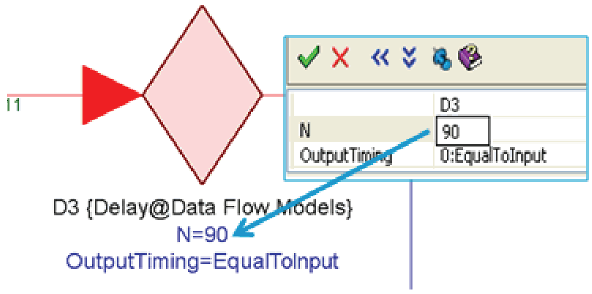
(***NOTE****: As above, change the Context in this Graph Properties dialog as well, to DF_EVM_BER, to ensure you're viewing the correct data*):

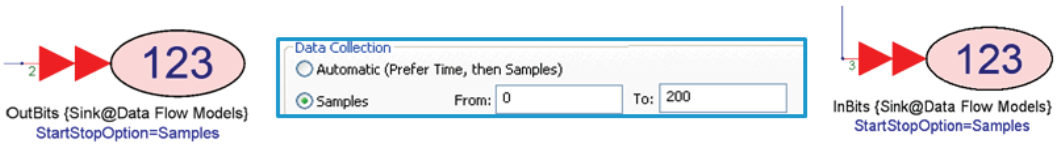


With these results, you can adjust the input delay from 50 to 90 (add the 40 sample time) to the input.

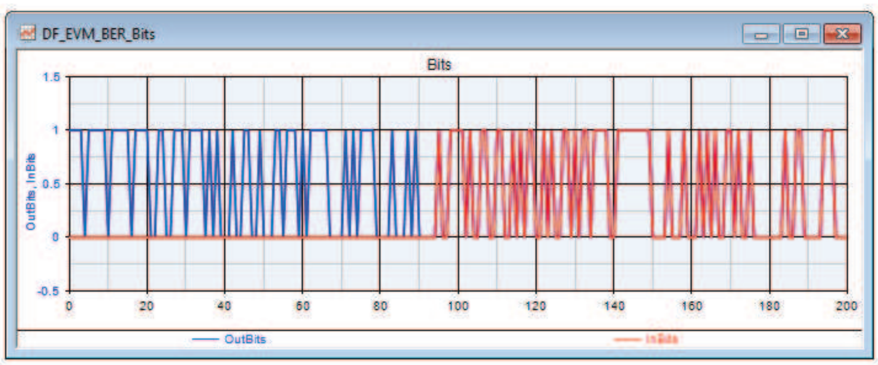
b. Set the **Delay** from the **bit source** from 50 to **90** as shown here.

c. Go to the **InBits** and **OutBits** sinks and increase the samples to **200** as shown here. With a delay of 90 samples, you need more data in the plots to see the bit alignment.





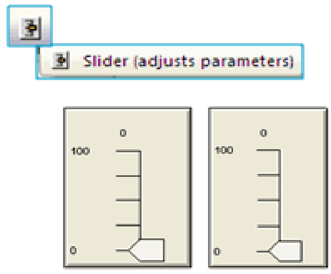
d. **Run the analysis** again and you should see correlation and aligned bits after 90 samples as shown here.



e. Edit your **DF_BER_EVM_Bits** plot and change the Y axis to 200, if needed.

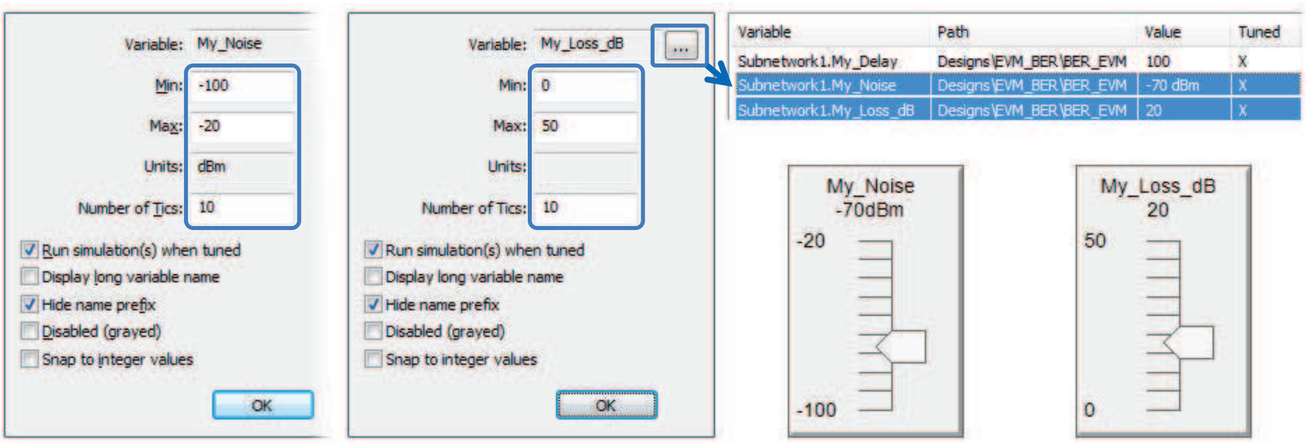
Next, you will see the effects of noise and loss.

13. Tuning Sliders and Constellation



a. From the Annotation Toolbar, use the icon to insert two sliders into your BER_EVM schematic as shown here.

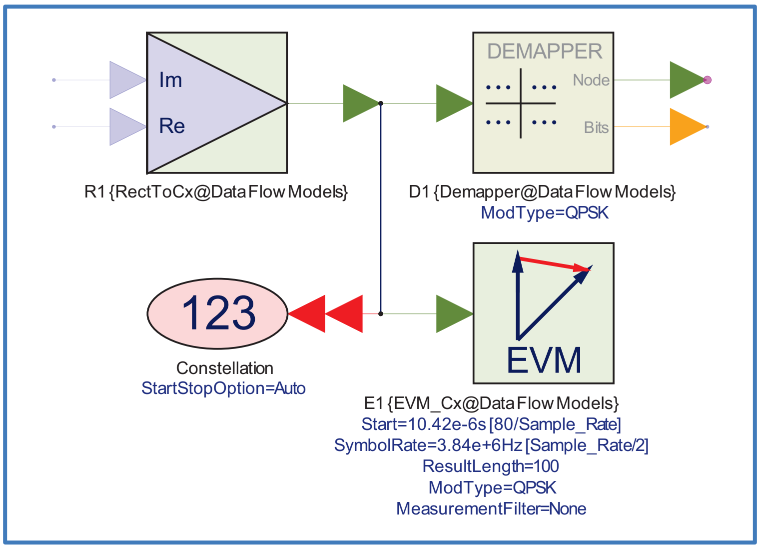
b. Edit the first slider and set it to the variable: (Subnetwork1.)**My_Noise**. Set the min: **-100** and Max **-20** using **10** Tics. Also, uncheck the box to Display long variable name and check Hide name prefix – click OK as needed.



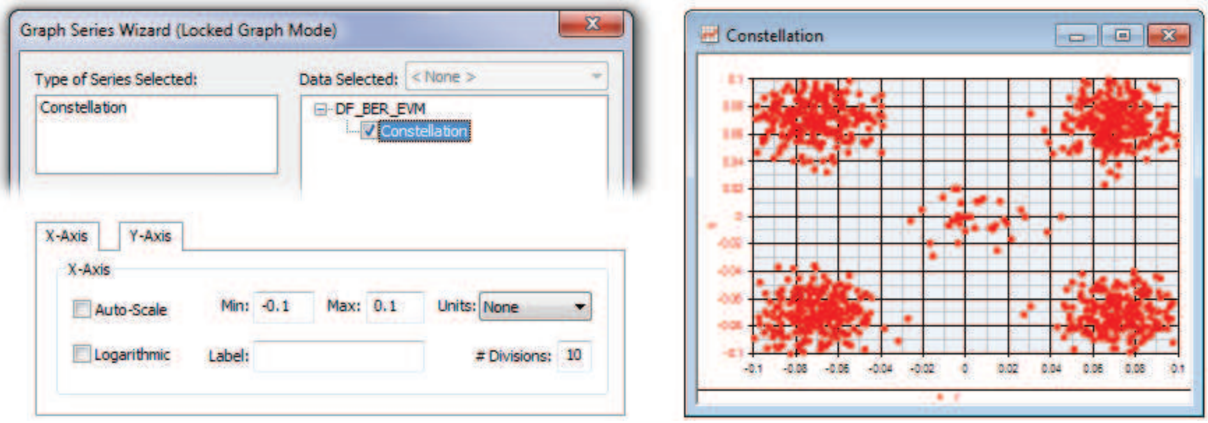
c. Edit the other slider for the **My_Loss** variable from **0** to **50** dB with all other settings the same as My_Noise. You should now have two sliders to adjust the path loss and noise.

d. Insert one more sink (keyboard **s**) connected to to the intput of the Demapper as shown here. Name the sink **Constellation** and leave the default settings (automatic). This will collect data for the coded constellation plot that you will set up next.

e. Set new values with the sliders and run the simulation.



Add one more **Graph** to the workspace tree in the EVM_BER folder – you should know how to do this by now. Select a **Constellation** type and the Constellation sink data. You can also set the **X and Y** axes form **-0.1 to 0.1**. You should now have a plot of the coded constalltion as shown here.



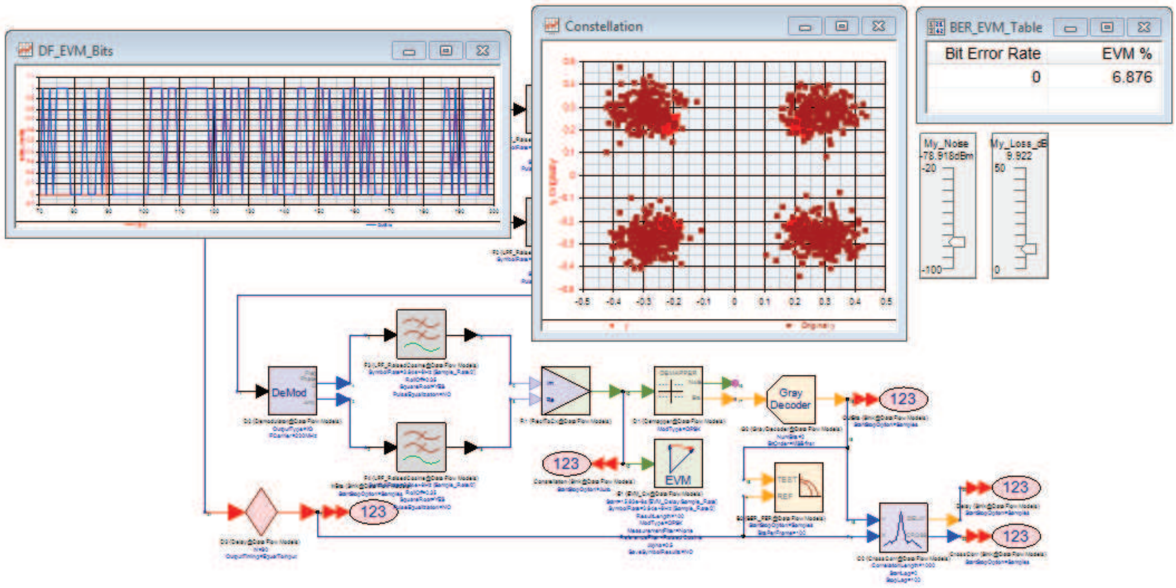
NOTE: The samples in the middle are the startup (delayed). You can remove them on the graph by editing the custom equations and entering the startup value. These might also ruin your EVM measurement – you can therefore add additional 40 samples of delay to the EVM sink's ***Start*** *parameter from 80/Sample_Rate to* ***120/SampleRate****. You can play around with this parameter, if you have time.*

f. *S*ave the workspace again.

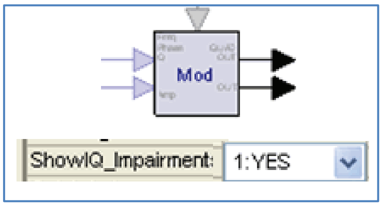
14. Tuning with BER, EVM and Constellation Results

a. Go ahead and **adjust the sliders** to see the effects of loss and noise on the BER, EVM%, Constellation and Bit alignment.

b. Experiment with the settings if you have time. Save the workspace and close all window when you are finished – unless you do the optional step (next).



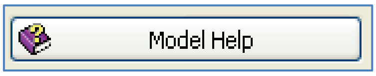
15. OPTIONAL: Modulator IQ Impairments



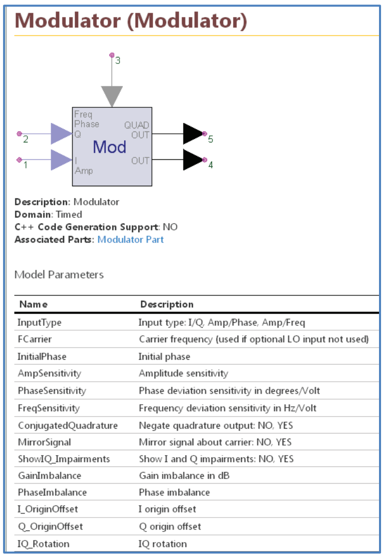
a. Reset your schemat and set Noise to -70 dBm and My_Loss to 20 dB.

b. Edit the modulator and select the option to **ShowIQ_Impairment: 1:YES**, as shown here.

c. Click the Model Help button. All the components have this button which you can use to read about the various parameters and how to set them. Take a moment to read about the impairment parameters as shown here. Close the Help when finished.



Name	Value	Units
InputType	0:I/Q	( )
FCarrier	220	(MHz)
InitialPhase	0	deg
AmpSensitivity	1	( )
ConjugatedQuadrature	0:NO	( )
MirrorSignal	0:NO	( )
ShowIQ_Impairments	1:YES	( )
GainImbalance	4	( )
PhaseImbalance	11	deg
I_OriginOffset	0.0	( )
Q_OriginOffset	0.0	( )
IQ_Rotation	8	deg


Modulator (Modulator)

Description: Modulator
Domain: Timed
C++ Code Generation Support: NO
Associated Parts: Modulator Part

Model Parameters

Name	Description
InputType	Input type: I/Q, Amp/Phase, Amp/Freq
FCarrier	Carrier frequency (used if optional LO input not used)
InitialPhase	Initial phase
AmpSensitivity	Amplitude sensitivity
PhaseSensitivity	Phase deviation sensitivity in degrees/Volt
FreqSensitivity	Frequency deviation sensitivity in Hz/Volt
ConjugatedQuadrature	Negate quadrature output: NO, YES
MirrorSignal	Mirror signal about carrier: NO, YES
ShowIQ_Impairments	Show I and Q impairments: NO, YES
GainImbalance	Gain imbalance in dB
PhaseImbalance	Phase imbalance
I_OriginOffset	I origin offset
Q_OriginOffset	Q origin offset
IQ_Rotation	IQ rotation

d. Try setting some of the impairments (similar to those shown here) – click **OK**. Then **run the analysis again** and look at the results in the dataset to see the effects. Of course, your values will vary depending upon how you set the parameters, but further degrading should occur.

e. When finished experimenting with modulator impairments, **edit the modulator** again and click **Parameter Options** and **Use Default:Check All**. Then uncheck the **FCarrier** box (Use Default) to return to 220 MHz with no impairments.

f. Now, your schematic should only have the channel and no other impariments.

g. Save the workspace and close all windows.

<u>**END OF LAB EXERCISE**</u>