

Exercise 4, Mon 5.12.

Problem 1.

Simulate and Analyze a signal.

Create a VI that will produce a signal with sine wave and high frequency noise (Hint for high frequency noise generation can be found from the Example finder: Analyzing and Processing Signals >> Filtering And Conditioning >> Extract The Sine Wave). From the front panel of your VI you should be able to set the frequency (in Hz), amplitude, and phase (in deg) of the sine wave. You should also be able to set the sampling rate and the simulated number of points. Visualize the generated waveform using a Waveform Graph.

Calculate the Auto Power Spectrum of your generated signal and visualize that on a separate graph. Filter the generated signal so that you would get rid of the high frequency noise. Plot also this filtered signal. Calculate the Auto Power Spectrum of the filtered signal and display that on graph.

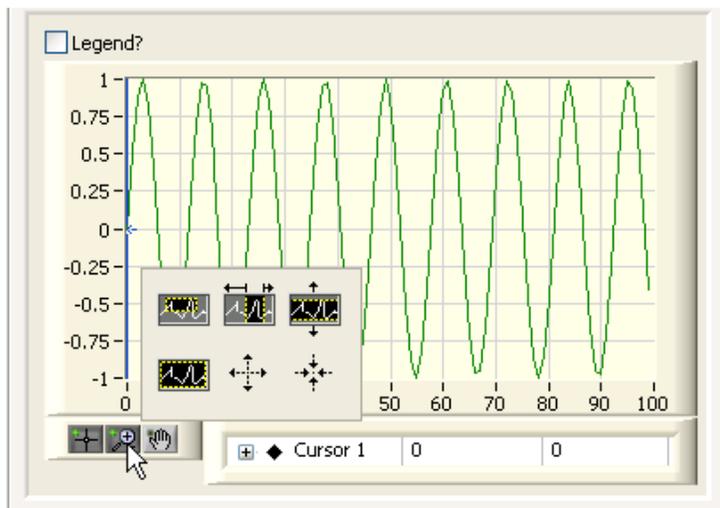
Make there a possibility to save the filtered sine wave.

Return your VI

Problem 2

Next we will practice the use of Mathscript node:

1. Make the program to read in data from a file. Use the data generated and saved in previous problem.
2. Place down a MathScript Node (**Programming»Structures»MathScript Node**).
3. Right-click on the left border of the MathScript Node and select **Add Input**. Name this input "In" by typing while the input node is highlighted black.
4. Wire the output from the file read to the "In" node on the MathScript Node.
5. In order to use the data in the Interactive MathScript Window it is necessary to declare the input variable as a global variable. Inside the MathScript Node type "global In;".
6. Run the VI.
7. Open the Interactive MathScript Window (**Tools»MathScript Window...**).
8. In the MathScript Window, the Command Window can be used to enter in the command that you wish to compute. In the Command Window, type "global In" and press "Enter". This will allow you to see the data passed to the variable "In" on the MathScript Node.
9. Notice that all declared variables in the script along with their dimensions and type are listed on the "Variables" tab. To display the graphed data, click once on the variable **In** and change the drop down menu from "Numeric" to "Graph". (Matlab might be using 'dot' as decimal point so to get the data in correctly to the Mathscript Window you might need to change system default decimal point to 'dot' and restart labview (Language and Regional Settings). After changing the decimal point you might need to save the data into a file again so that the load function recognizes it correctly.)
10. Use the graph palette to zoom in on your data.



11. Right-click on "Cursor 1" and choose **Bring to Center**. What does this do?
 12. Drag the cursor around. The cursor will not move if the zoom option is selected.
 13. Right-click on the graph and choose **Undock Window**. What does this do? Close this new window when you are finished.
 14. Multiply the data by a decreasing exponential function. Follow these steps:
 - a. Make an array of data that constitutes a ramp function going from 0.01 to 5 by typing "Array = [0.01:(5-0.01)/(length(In)-1):5];" in the Command Window and pressing Enter. What type of variable is "Array"?
 - b. Make an array containing a decreasing exponential. Type "Exp = 5*exp(-Array);" and press Enter.
 - c. Now multiply the Exp and In arrays element by element by typing "Out = In.*Exp;" and pressing Enter.
 - d. Look at the graph of the variable "Out".
 15. Go to the History tab and use Ctrl-click to choose the 4 commands you just entered. Copy those commands using <Ctrl-C>.
 16. On the Script tab, paste the commands into the Script Editor using <Ctrl-V>.
 17. Save your script by clicking "Save" at the bottom of the window. Save it as "myscript.m"
 18. Close the MathScript Window.
 19. Return to the block diagram. Load the script you just made by right-clicking on the MathScript Node border and selecting **Import...** Navigate to myscript.m, select it, and click "OK".
 20. Right-click on the right border of the MathScript Node and select **Add Output>>Out**
 21. Right-click on the variable "Out" and select **Choose Data Type>>All Data Types>1D-Array>DBL 1D**. Output data types must be set manually on the MathScript Node.
 22. Create Waveform graph from Front Panel and wire "Out" to the Waveform Graph.
 23. Return to the front panel and run the VI.
 24. Save and close your VI.
- Return your VI.