Lastname: _______________

5Digit SS:___________________

LABORATORY
IIR Filters

LAB 6: INTRODUCTION TO IIR FILTERS
(10 points)

In this lab, you will be introduced to the design of Infinite Impulse Response (IIR) filters. Filters are one of the most important elements in DSP and are typically used to isolate a specific frequency band of a signal. IIR filters are of particular interest because, with just a few coefficients, relatively sharp transition bands can be realized.

In the **pre-lab,** you will use "pencil-and-paper" to compute the results you expect later in your design implementation. In the **design part,** you will design a first-order IIR filter and a third order system direct form filter.

Lab Objectives

After completing this lab you should be able to

- Design and simulate an first order IIR filter
- Determine magnitude, phase and pole zero diagram of IIR filters
- Design a 3. order elliptic low pass filters
- Compare IIR and FIR design parameter

Pre-lab (3 points)

1. For a first-order IIR filter with a transfer function $H(z)=b/(1+az^{-1})$, determine a and b such that the filter is a halfband filter (i.e., $|H(\omega=0)| = 1$ and $|H(\omega=\pi/2)| = 0.5$).

Feedback gain a = _________.

Feedforward gain b = _________.

Hint: The quadratic equation $x^2+px+q=0$ has the solution $x_{1,2}=p/2+/-\text{sqrt}((p/2)^2-q)$)

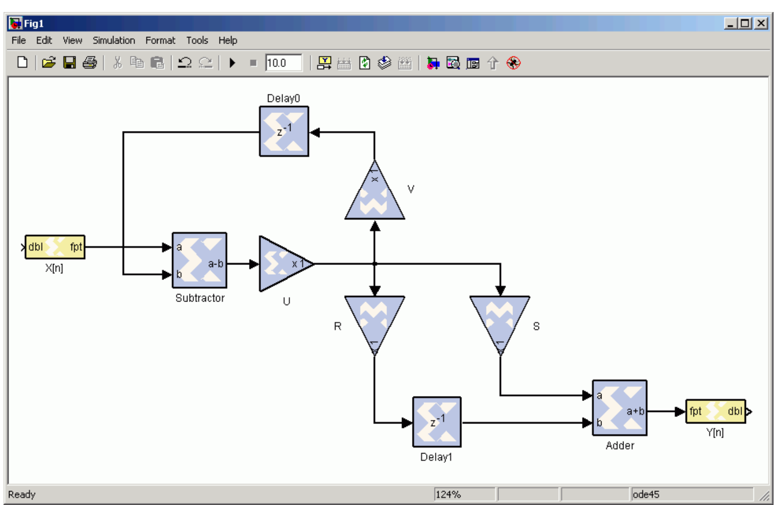


Figure 1.

2. Determine the transfer function for the system in Figure 1.

H(z) = Y(z)/X(z) =

3. Determine zero(s) and pole(s) of the system in terms of the coefficients U,V,R, and S.

Zero(s) at = _________ **Pole(s) at =** ________

4. Find the values of the coefficients (U, V, R, S) so that the transfer function H(z) from part 1 is realized.

U =

V =

R =

S =

5. Compare FIR and IIR filters regarding the following properties:

	FIR filter	**IIR filter**
Filter Length		
Filter Linearity		
Coefficient design method		
Pole/zero locations		
Coefficient sensitivity to quantization		

Lastname: _______________

5Digit SS:___________________

LABORATORY
IIR Filters

Simulink Design-lab

Follow the directions below to implement a first and third-order IIR filter.

A. Getting Started

If you are in B114 or the digital logic lab:

1. On the desktop, double-click on **Engineering folder**.
2. Double click on the **MatLab** icon  to start **MatLab**. From the to top icon list in the **MatLab** window click on the **Simulink** icon  to start **Simulink.**
3. You should not save anything on the local hard disk. You will have to use your own Zip, floppy disk, USB flash drive, or your "mapped" home directory to save the files. Create a New Folder named **DSPwFPGAs** on your mapped network drive.

B. Compiling an Existing Design

1. Download the "**iirorder1.mdl**" and "**showfft.m**" files from the class webpage and put them in your **DSPwFPGAs** folder.
2. Click on the "Current Directory" selection icon and select your **DSPwFPGAs** folder as the current directory.
3. The files in the **DSPwFPGAs** can now be easily accessed with the "open file" button  on the **MatLab** toolbar**.** Double click on the "**iirorder1.mdl**" file and after a moment you should see the incomplete design:

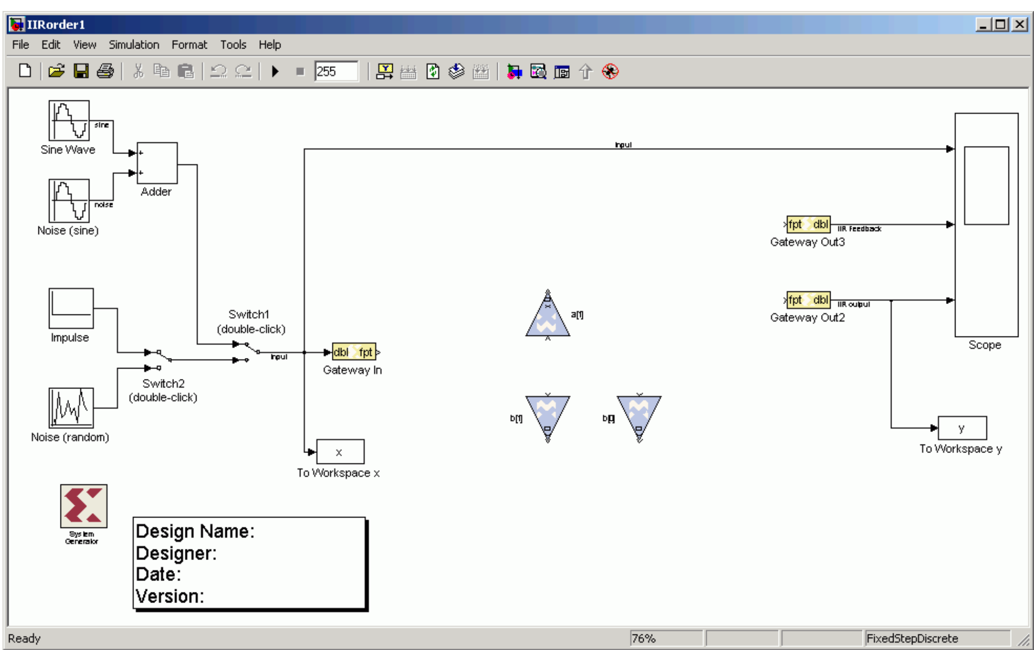
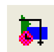


Figure 2: Incomplete First-Order IIR Filter.

4. Complete the design

   a. Change the design coefficients to those you computed in part 1 of the pre-lab. This can be done by double-clicking the CMult blocks and changing "Value of Constant." Also note that if your values are not accurate enough, you can increase the "Binary Point of Constant" by again double-clicking a specific block, checking "Show Constant Data Type Parameters," and then changing the number of bits used to define your constant.

   b. Complete your design by adding delay and adder/subtractor elements from the Xilinx Blockset library by following the instructions below:

      - Click on the **Library Browser** icon  and then double-click the **Xilinx Blockset** directory.
      - Under the **Math** subdirectory, select the **AddSub** block drag it into your **Simulink** workspace.
      - By default, **AddSub** is configured as an adder. To change it to a subtractor, simply double-click on the block in your workspace and change the **mode** to **subtraction** (see figure below).

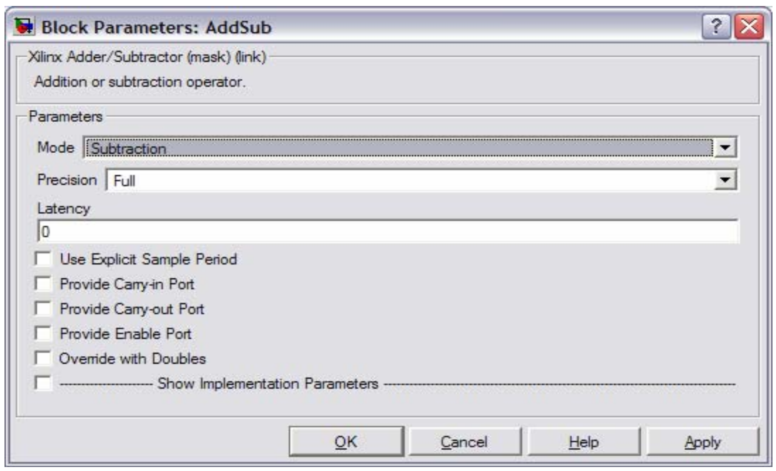


Figure 3

      - Add delay elements to your design by selecting the **Basic Elements** subdirectory, selecting the **Delay** block, and dragging it into your **Simulink** workspace.

   c. Other tricks:

      - **Rotate** a block in your workspace by selecting it and pressing **CTRL+R**
      - **Flip** a block by selecting it and pressing **CTRL+I**
      - To quickly **Connect** blocks together, first select the "output" block (with your mouse), then **hold CTRL** and select the "input" block with your mouse.

5. At the MatLab prompt, use the predefined functions **freqz()** and **zplane()** to plot the frequency spectrum and pole/zero plot, respectively. First, you must define the numerator and denominator polynomial coefficients in vector form.

   - Recall the transfer function you found in the pre-lab, of form
     $H[z] = b[z]/a[z] = (b[0]+b[1]z^{-1}+...) / (a[0]+a[1]z^{-1}+...)$

   - **Use your coefficient values** to define the coefficient vectors. At the MatLab prompt, type the following:

Lastname: _______________

5Digit SS:___________________

LABORATORY
IIR Filters

>> **b=[b[0], b[1], ...]; a=[a[0], a[1], ...];**

- Now, call the functions to generate the desired plots:

>> **freqz(b,a)**
>> **zplane(b,a)**

6. Complete the following diagrams:

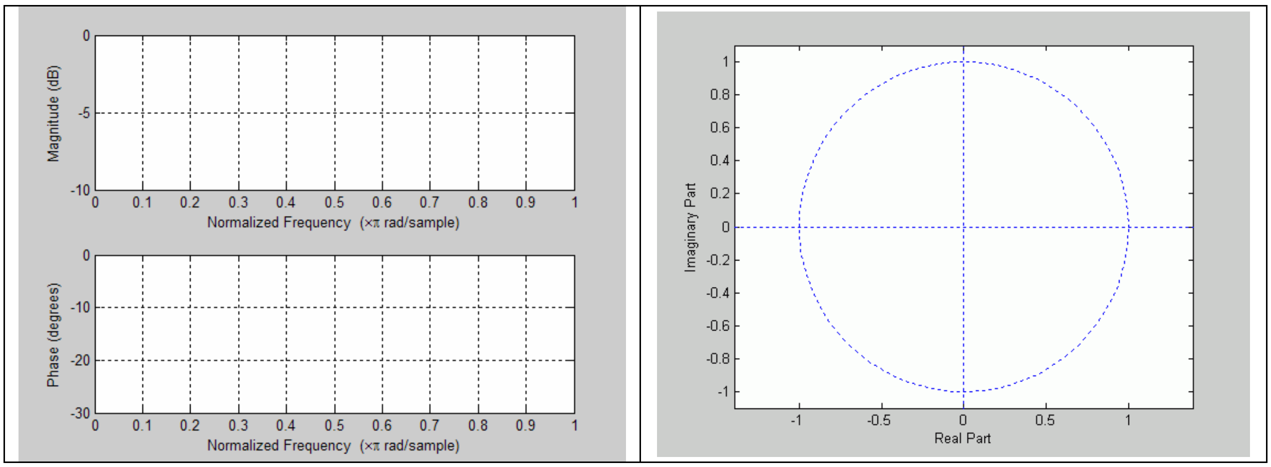


Figure 4: Fill-in Magnitude, Phase and Pole/Zero Plots for First-Order IIR Filter

7. Simulate the design with the two sine input signals. At the MatLab prompt use showfft(x) and showfft(y) to display the spectra. Determine the 2nd sine (i.e. noise) component amplitude before and after filtering:

Before Filtering, $H(\omega_2)$ = ______

After Filtering $H(\omega_2)$ = ______

Note: Make sure the "**showfft.m**" file is in your current directory!

8. To Compile the design, double-click on the **System Generator** block and ensure that the path "**./iir**" exists in the "Target Directory" field. Also make sure that under "Part," you have selected the Spartan3 xc3s200-5ft256. To compile, click on **generate**. Find the new folder called "**iir**" that has been created in your **DSPwFPGAs** folder. Now, double-click "**iirorder1_clk_wrapper.ise"** to open in ISE.

Right click on Generate Programming File near the bottom of the **processes** window to compile. Next, determine the “Total number 4-input LUTs” by clicking on View Design Summary. Also, find the maximum frequency by looking at the **Post Place and Route Static Timing Report,** under **Detailed Reports** (at the bottom of the design summary). The number of required **18x18 Multipliers** and **Black RAM** can be found in the **MAP Report**.

Total number 4-input LUTs = ___________

18x18 Multipliers = ___________

Block RAM = ___________

Max. Freq. = ___________

C. Designing a 3rd Order Direct-Form Filter

1. Download "**IIRorder3.mdl**" and "**setup_iir3.m**" from the class webpage and put them in your **DSPwFPGAs** folder.

2. Open "**setup_iir.m**" with a text editor. There, you will find the filter coefficients and calls to the predefined MatLab functions freqz() and zplane() for the spectrum and pole/zero plot, respectively. At the MatLab prompt, type "setup_iir3" and complete the following diagrams:

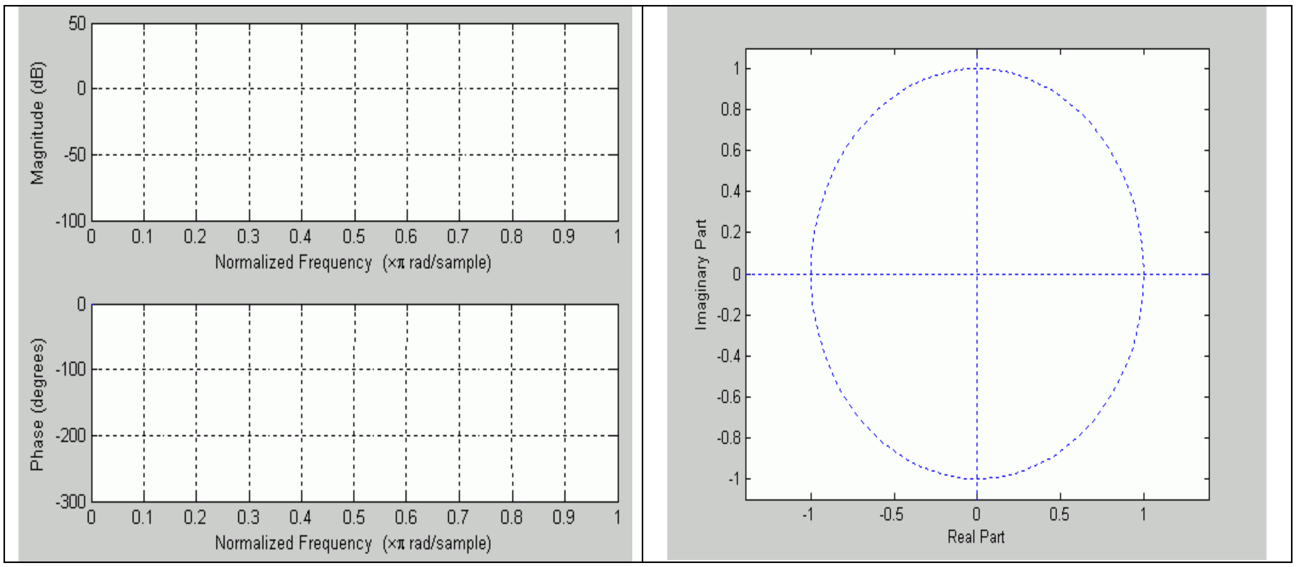


Figure 5: Fill-in Magnitude, Phase and Pole/Zero Plot for Third-Order IIR Filter

3. Complete the design: add the delay and adder elements and change the values of the gains according to those in the "**setup_iir3.m**" file and the instructions from part B.

Lastname: _______________

5Digit SS:___________________

LABORATORY
IIR Filters

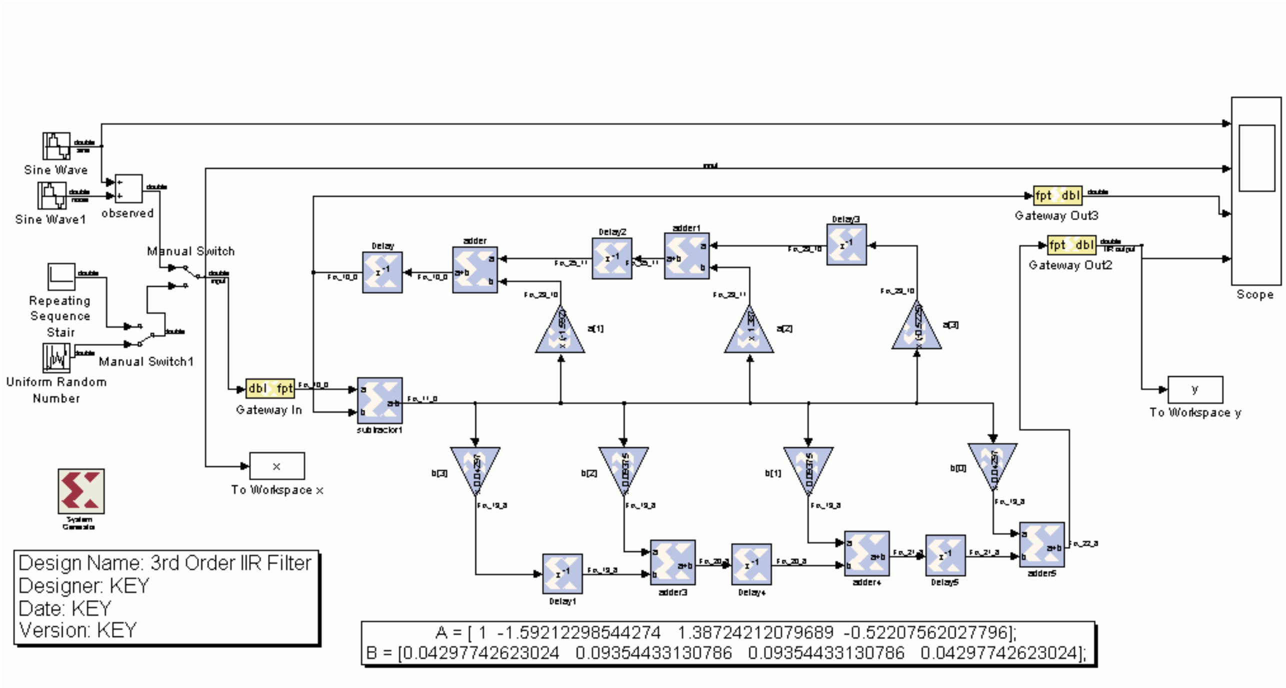


Figure 6: Third-Order IIR Filter

4. Set the manual switches such that the **impulse** signal is the IIR filter input. Simulate the design using the impulse input signal and **complete the spectra** using use showfft(x) and showfft(y):

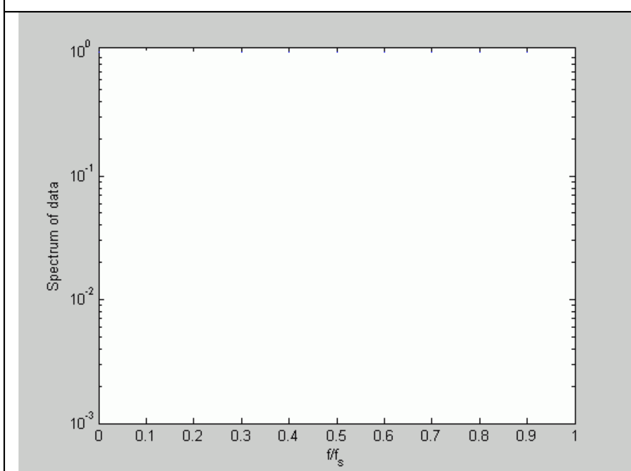
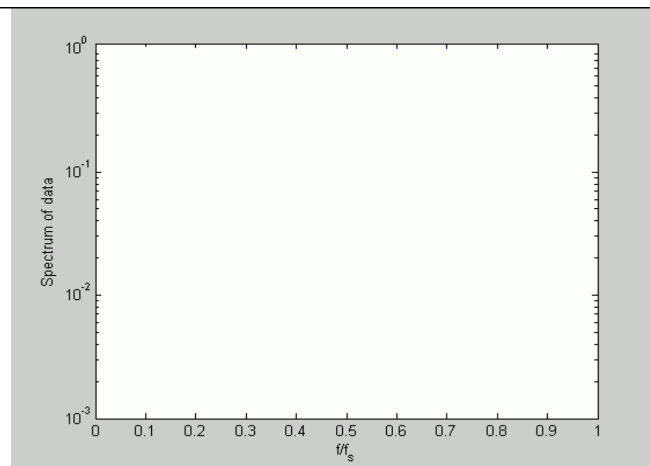
Input spectrum	Output spectrum
	

Figure 7: Fill-in Third-Order IIR Impulse Response

5. Set the manual switches such that the sine signals are the IIR filter input. Simulate the design using the sine input signals and **complete the spectra**:

Lastname: _______________

5Digit SS:___________________

LABORATORY
IIR Filters

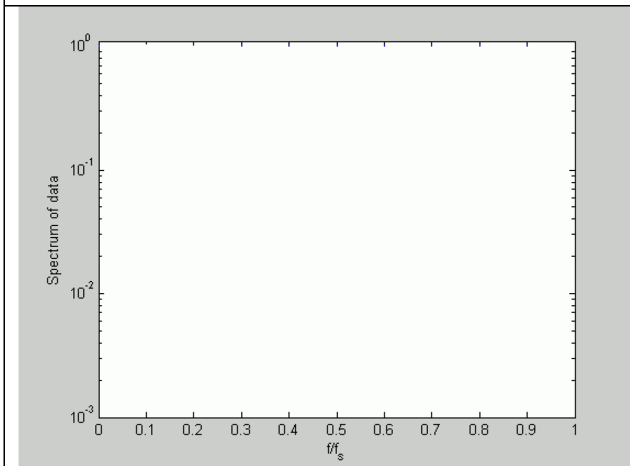
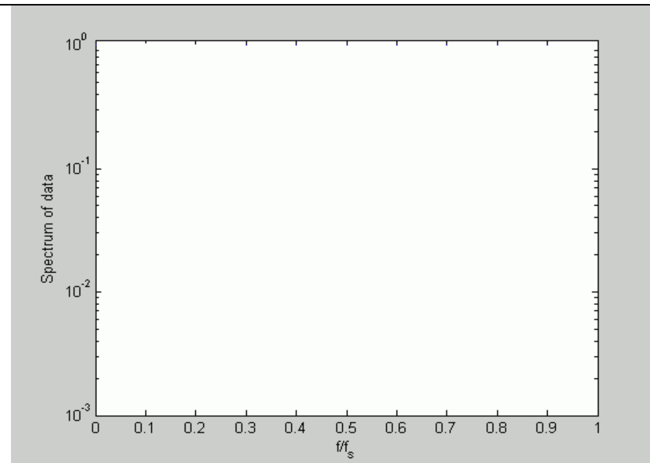
Input spectrum	Output spectrum
	

Figure 8: Fill-in Third-Order IIR Response to Sine Waves

6. Use the function showfft(x) and showfft(y) provided to "measure" the amplitude of the noise sine component before and after filtering. Compare the results to the first order system.

Before Filtering, H(ω_2) = ___________

After Filtering H(ω_2) = __________

7. To Compile the design, double-click on the **System Generator** block and ensure that the path "**./iir**" exists in the "Target Directory" field. Also make sure that under "Part," you have selected the Spartan3 xc3s200-5ft256. To compile, click on **generate**. Find the new folder called "**iir**" that has been created in your **DSPwFPGAs** folder. Now, double-click "**iirorder3_clk_wrapper.ise"** to open in ISE.

Right click on Generate Programming File near the bottom of the **processes** window to compile. Next, determine the "Total number 4-input LUTs" by clicking on View Design Summary. Also, find the maximum frequency by looking at the **Post Place and Route Static Timing Report,** under **Detailed Reports** (at the bottom of the design summary). The number of required **18x18 Multipliers** and **Black RAM** can be found in the **MAP Report**.

Total number 4-input LUTs = __________

18x18 Multipliers = ___________

Block RAM = ___________

Max. Freq. = ___________

Lastname: _______________

5Digit SS:___________________

LABORATORY
IIR Filters

F. Deliverables:

1. Solve the problems of the pre-lab. (3 points).

2. Print the .MDL files and the Simulink simulations (7 extra points).

Make sure your name and SS is on all pages you turn in!