

Pre-lab Questions

- Q1.** In dB, what should be the maximum gain in the stopband?
- Q2.** Answer the following questions about your filter that Matlab designed for you.
- What is the order of the filter Matlab designed?
 - If you were to make it a Butterworth IIR filter, what would be the order?
- Q3.** Zooming in as needed, is the passband requirement still met? The stopband? How close are they?
- Q4.** Zooming in as needed, is the passband requirement now met? The stopband? How close are they?
- Q5.** Attach the code you used to generate the coefficients in the format shown above. You do not need to attach the actual coefficients to your pre-lab report but you will need them for the in-lab part.
- Q6.** What is the largest (in terms of absolute value) integer value you have as a coefficient? The smallest? What is the largest Q value you could have used to represent these coefficients (assuming you continued to use 16-bits)?

```
module lab4(  
    input  [17:0] SW,  
    input  [3:0] KEY,  
    output [7:0] HEX0,  
    output [7:0] HEX1,  
    output [7:0] HEX2,  
    output [7:0] HEX3  
);  
  
reg [3:0] a;  
reg [3:0] b;  
reg [3:0] c;  
reg [3:0] d;  
  
always@(negedge KEY[0])  
begin  
    a<=SW[3:0];  
    b<=a;  
    c<=b;  
    d<=c;  
end  
  
bh one    (a,HEX0);  
bh two    (b,HEX1);  
bh three  (c,HEX2);
```

```
    bh four    (d,HEX3) ;  
  
endmodule
```

- Q7.** Consider the above code.
- Describe how you would expect the code to behave. Be detailed.
 - How would it be different if the “<=” in the always block was changed to a “=”?
- Q8.** Consider these two ways of implementing the same functionality.
- What are the advantages of the second scheme (with the for loop and array of busses) over the first?
 - When would you really want to use the “for” loop?
 - What might be an issue/worry with using the for loop solution?
- Q9.** For the PmodDA2:
- For each male pin, describe what each pin does. Provide more than just a signal name: describe what role the pin plays.
 - As part “a” of this question, but for the female pins.
- Q10.** Find the datasheet for the ADC. What is the fastest sclk could be according to the specification? (Hint: you should search for the datasheet for the ADC chip instead of the reference manual for the Pmod provided above.)
- Q11.** Do these two Pmod devices use SPI?
- Q12.** Write a Verilog module which implements a 3-tap direct form FIR filter. It should use 16-bit Q15 for its input and output. Let the coefficient $b[0]=0.25$, $b[1]=-0.25$, and $b[2]=0.125$. Name the input “x”, the output “y” and the clock “clk”. Do not use a for loop or an array of busses. This problem will be graded fairly generously given that you aren’t expected to test your solution. Just think about how it should be written and you don’t have to worry about rounding. You may do this problem with your lab partner.
- Q13.** Design a gain of 0.5 level shifting amplifier. The circuit diagram and analysis for such can be found in on the website. With one exception it should be possible to use only 10 kOhm resistors. The exception can make use of two 10 kOhm resistors perhaps connected in series or maybe in parallel... You may do this problem with your lab partner.

In-lab & Post-lab Questions

- Q1.** After playing around with the inputs, answer the following questions:
- What is the phase shift of a 1 KHz input? Give your answer as an (fairly approximate) value between 180 and -180 degrees.
 - How much total delay do you see in the system? How did you figure it out and why are you certain you aren't seeing the phase "wrapping around"?
 - Observe the frequency's impact on amplitude. At what frequency is the output amplitude about half that of the input?
 - Measure the gain and phase at 20 KHz
- Q2.** With the signal generator output set to 500Hz, *carefully* increment the output voltage by 0.1 V until you see clipping. At what output voltage (absolute) do you start to see clipping?
- Q3.** What modules are instantiated in top.v? What modules are instantiated in the whole design?
- Q4.** Our ADC inverts the input so in order to avoid a 180 degree phase shift between the input and output; we have to negate the input values before feeding to the output. Given that, what else do you think the lines below are doing?

```
always@(negedge AUD_DACLRCK)
    DataOut<= (DataIn == 16'h8000) ? 16'h7fff:-DataIn;
```

Change the code above to the code below and look at a 50Hz sine wave with $V_{pp} = 1$ V.

```
always@(negedge AUD_DACLRCK)
    DataOut<={-DataIn[15:13],13'd0};
```

- Q5.** What happens? Explain what that code is doing and why the output changes as it does.
- Q6.** Look over the DAC code
- Draw a figure that indicates what you would expect the sclk and sync signals to look like. Specifically indicate the frequency you would expect to have of each frequency. .
 - Explain how you figured out the answer to (a).
 - What frequency does the DAC output data at? How does that compare to the audio inputs on the FPGA board?
- Q7.** Look over the ADC code and the ADC datasheet.
- How fast is the sclk being sent to the ADC by the supplied code?
 - How many samples per second does the supplied code convert?
- Q8.** What is the approximate delay from the ADC to the DAC output?
- Q9.** Change the input signal to 20 KHz, measure gain and phase and compare to your results in Q1(d).

- Q10.** Say, after programming the chip, you were to set SW[15:0]=0x4000 and leave it there. What output would you expect to get after each press of KEY[3]?
- Q11.** Say, after programming the chip, you were to set SW[15:0]=0x4000, press and release KEY[3] and then set SW[15:0] to 0. What output would you expect to get after each press of KEY[3]?
- Q12.** Look at the flow summary under Compilation Report (it should come up automatically when you finish your build).
- Discuss the utilization of resources including the use of embedded multipliers. What is the largest order filter you think we could probably build (assume 90% utilization is as good as we can realistically do).
- Q13.** What type of filter (lowpass, bandpass, highpass) do we have? What are the cutoffs?
- Q14.** What is the worst case delay for this design? Where does it start and where does it end?
- Q15.** Look at the code and explain why “sum” is computed using a blocking assignment while “delay” is computed using a non-blocking assignment.
- Q16.** For your answer to G3, what did the utilization of resources look like? Did it scale in a reasonable way given the order of the filter? How about the worst-case path?
- Q17.** Answer the following questions.
- What is the worst-case path for your design? How does it compare to the direct form? Does that seem reasonable? Try to explain the results as best you can.
 - What about the utilization of resources? Any change with respect to the direct form? Does this seem reasonable? Try to explain the results as best you can.
- Q18.** What is the new cut-off frequency?
- Q19.** The signal processing we can do with the audio inputs and outputs on the DE2-70 and the C5515 USB stick are fairly limited due to filters built into those devices. Why do you suppose those filters are there?
- Q20.** Consider the worst-case delay you found for the largest filter you built.
- What speed (in Hz) could it in theory run at? Assume you need to add 5ns (total) to deal with clock skew, latch delay and setup time.
 - Give a rough idea how that compares to the C5515 doing a similar sized filter.