EE 113DA Digital Signal Processing Design

Lab 4: Digital Filtering

Fall 2018



Brian Tehrani

UID: 604715464

Milad Nourian

UID:004854226

Attiano Purpura-Pontoniere

UID:504847318

**Objective**

The objective of this lab is to compare filters from the LCDK to the digital filter in Matlab. In step 1, we use the function generator and excel to create bode plots of the LDCK frequency response. In Step 2 we create a digital filter in Matlab and compare it with a filter created in Code Composer studio. The results of the experiments will be compared and discussed to further elaborate filters within the LCDK.

**Step 1A: Measuring the Frequency Response of the LCDK**

In step one of the lab we are tasked with observing the frequency response of the LCDK without any software capable of altering the frequency response. The sampling frequency is set to 48 Ksps, the function generator is set to output a sinusoid with amplitude 500 mV peak to peak, and the waveforms are output to the oscilloscope. The LCDK is then inserted sine wave values of the following frequencies: 20, 50, 100, 200, 500, 1K, 2K, 5K, 10K, 14K, and 20 KHz. The ratio of the output to input amplitude is then taken and converted to a dB scale. The bode plot is shown in figure 2.

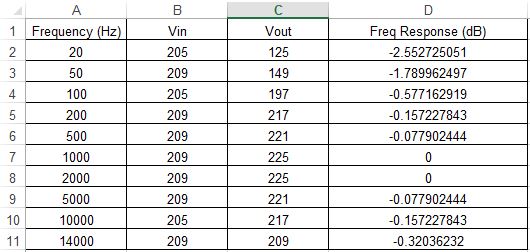


Figure 1: Data for part 1a

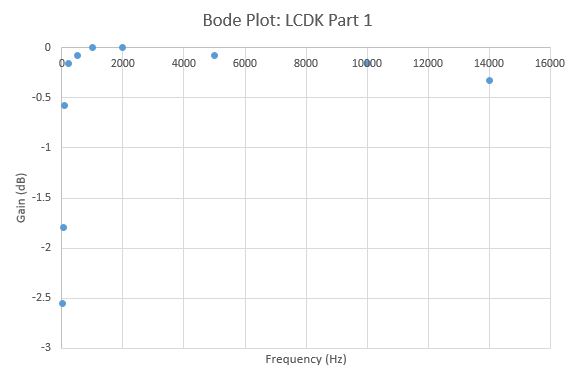


Figure 2: Bode plot for part 1a

**Step 1B: Single Pole LPF With Different Cutoffs**

In Step 1B we are tasked with transforming the LCDK into a low-pass filter using the same sampling frequency, 48 Ksps. A bilinear transformation is used to design two similar single pole analog RC LPFs with respective following cut-off frequencies: 1 KHz and 20 KHz. Figure 5 shows the measurements taken in part B, figures 3 show the response for the 1 kHz cutoff, and figure 4 shows the response form the 20 kHz cutoff.

Step 1b.1:

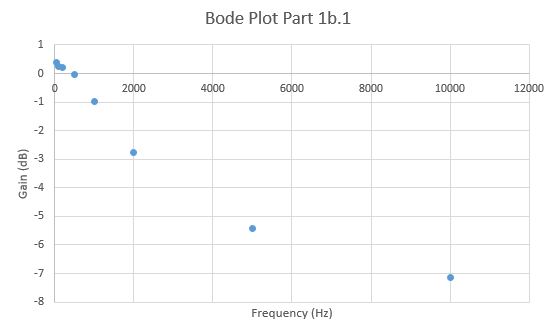


Figure 3: Bode Plot for 1 kHz cutoff

Step 1b.2:

:

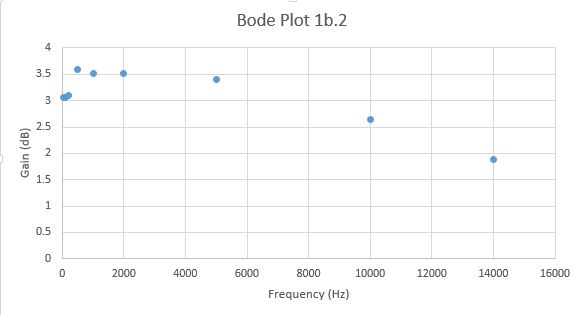
****

Figure 4: Bode Plot for 20 kHz cutoff

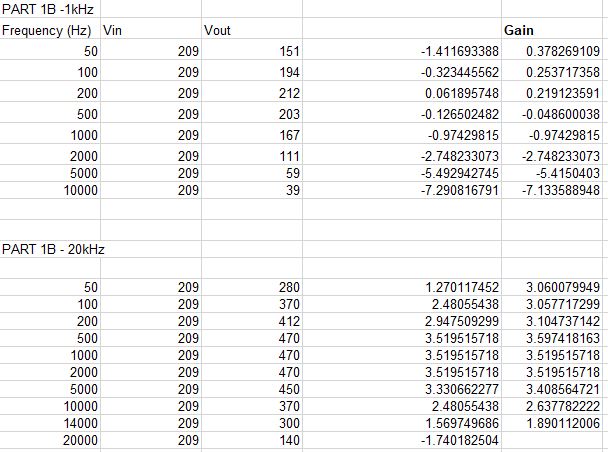
****

Figure 5: Data Values for Part 1b plots

Discussion Step 1:

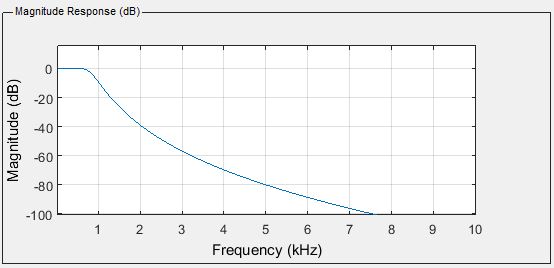
In step 1a/b of the lab we turned the LCDK into a low-pass filter with cutoff frequencies at 1kHz and 20kHz respectively. We accomplished this using the bilinear transform that relates the analog single pole transfer function in the s (real/imaginary) domain to the corresponding discrete transfer function in the z domain. We calculated the dB gain of the 1kHz filter for frequncies of 100, 200, 500, 1000, 2000, 5000, and 10000hz. We calculated the dB gain of the 20kHz filter for frequencies of 200, 500, 1000, 2000, 5000, 10000, 14000, and 2000. The plotted frequency response of the 1kHz and 20kHz filter produced with the LCDK are shown in Figure 3 and 4 respectively. As can be seen in both figures, the higher frequencies have lower gain, corresponding to the expected attenuation a low-pass filter applies to higher frequencies. The bode plot for the 20kHz has an unexpected outcome in that the first few points are also attenuated because of the DC-blocking capacitor built into the LCDK. The theory behind what we saw is essentially that of the frequency response of a lowpass filter to increasing frequency input, as expected.

**Step 2: IIR LPF**

The first part of Step 2 involves the use of a Matlab GUI called *fdatool* to create digital filters. With this tool, we are tasked to create a filter with the design parameters as follows: Fpass of 500 Hz, Fstop of 5 KHz, Apass = 1 dB, and Astop = 80 dB. Use a sample rate of 48 Ksps. The response plots of the magnitude are shown in figure 6. The top plot in figure 6 is the screenshot of the zoom with parameters: frequencies from 0 to 10 KHz and Magnitudes from -100 to 0 dB. The lower plot in figure 6 has parameters: frequency from 0 to 3 KHz and magnitudes from -20 to 0 dB. The -3 dB point in this lower plot correlates with a frequency of approximately 852 Hz. The number of second order sections are found to be 3 and shown in figure 6. The data in figure 7 show the corresponding G and SOS values for the created filter in Matlab. This filter is shown to be a low-pass.

The next part of Step 2 involves the use of Code Composer Studio and a prewritten code

C code iirsos.c. The G and SOS values from figure 6 are placed in this code and measurements of the gain are recorded by the LCDK using an oscilloscope at frequencies: 20, 50, 100, 200, 500, 1K, 2K, 5K, 10 KHz, 20 KHz. A bode plot of the response is shown in figure 8.



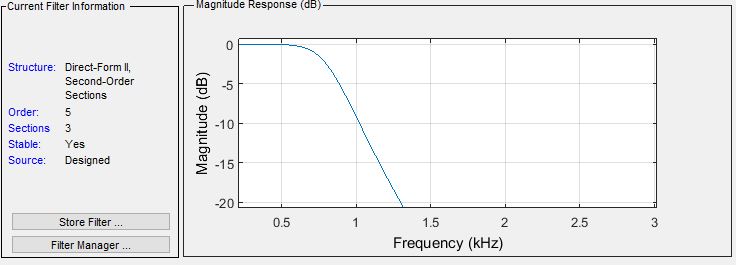


Figure 6: Matlab Magnitude and Frequency Response

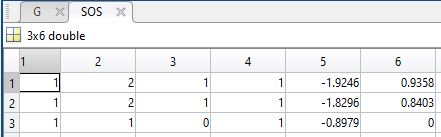
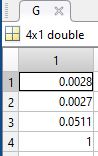


Figure 7: G and SOS values form Matlab

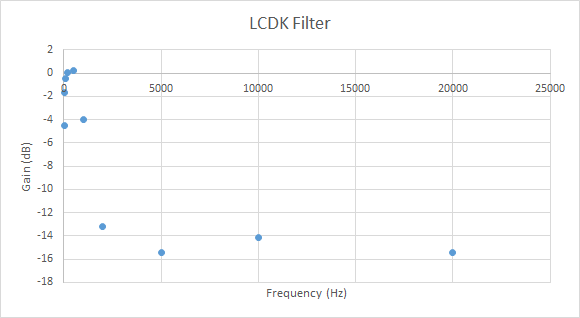


Figure 8: LCDK Bode plot for iirsos.c code with G and SOS values

Discussion Step 2:

In step 2 of the lab, we have used the MATLAB GUI tool ***fdatool*** to design a low pass filter. Using the given specifications of the filter including the Fpass, Fstop, Apass and Astop to get the proper filter, we obtain a transfer function for the LPF. We then look at the response of the filter by zooming in and looking at the amplitude values versus frequency values that are passed – for frequencies 0 to 3kHz and for amplitude from -20dB to 0 dB (figure 6). Essentially, we used an IIR low-pass filter to pass only the low frequency components of the LCDK. In the second part of Step 2, we use the obtained values of G and SOS (which are essentially the numerator and denominator of the filter transfer function) to find the frequency response of the LCDK after using the low pass filter (Figure 8). Comparing results from Step 1 for frequency response of the LCDK (Figure 2), and Figure 8 after the signal generated from the LCDK is passed through a low pass filter, the result of application of low pass filter (figure 8) is that the higher frequency components of the LCDK frequency response are cut and attenuated sharply (everything after the cut-off frequency of the filter we used) whereas the lower frequency components stay at almost the same gain (gain = 0dB).