

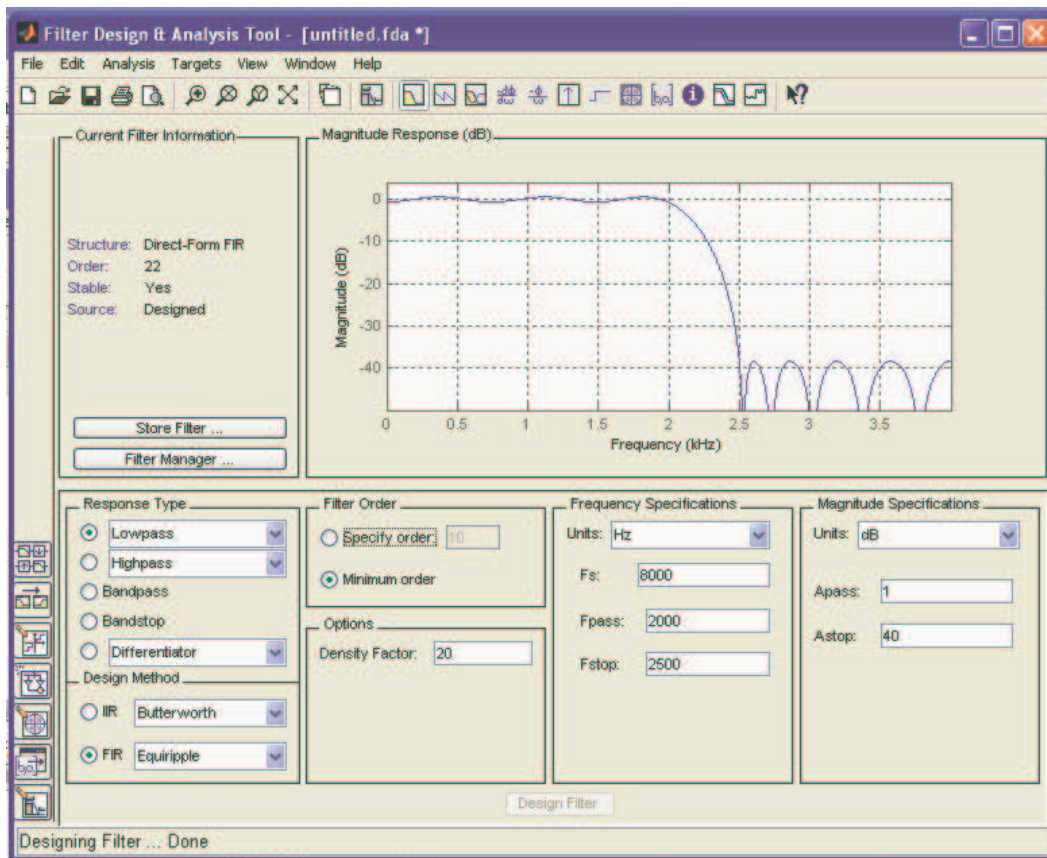
# EE/TE 4385 DSP-Based Design Project I


## Lab 3: FIR Filtering – Part 2

Lab Report Due: 9/20/06, 2PM,

### Lab 3c. FIR Filter Design and Filtering

- We will design FIR filter coefficients using `fdatool` in MATLAB.
- Start MATLAB from the link on your desktop.
- Type `fdatool` in the MATLAB command line to bring up the filter design window. Here, we will design a low pass filter coefficients. But, using linear filters, one can implement various operations such as differentiation or transformation.
- Modify the filter specification as shown in the figure below and click Design Filter button.



- If the order of the filter is  $N$ , the number of the coefficients will be  $N+1$ . Thus, the above filter is  $22^{\text{nd}}$  order filter and it has 23 coefficients. By click on the impulse response icon , we can see the stem graph of the filter coefficients over time axis. You can also try other choices: magnitude response, phase response, etc.
- We need to export these coefficients into a header file.

- But, first, download the Lab3c source files from Digital Filtering link on course web site.
- Unzip the download file into your myprojects directory. Open the lab3c project file in CC. Inspect the main file `filter.c`. Notice
 

```

#define N ?? //filter order
and
// FIR filter coefficients
float h[N+1] = {?,?,?,?}; // insert your FIR filter
coefficients here.

```
- Now go back to fdatool window. Check filter order from fdatool window and use it to define N in `filter.c`.
- Click on targets from toolbar menu on fdatool window. Select Code Composer Studio option to export the coefficients into C header file. We will use this option so that we can copy and paste them into our file in CC. Click on Generate button on the pop-up Export to Code Composer Studio IDE window. Save the header file (fdacoefs.h is the default choice) into your working directory or on your Desktop. Close this window. Open the header file using notepad. Copy the coefficients in the brackets into `float h[N+1] = {?,?,?,?};`
- FIR filtering computation is in `transmit_ISR()` interrupt handler function in `filter.c`. Inspect the computation and relate it to the actual equation.
- Connect the Run your loaded program in your DSP
- Apply a sinusoidal signal with 1V peak-to-peak amplitude to the input of DSP.
- Vary the input frequency at the proper intervals. Recall the sampling frequency is set to 8000 Hz for this experiment. Output should be connected to the Oscilloscope.
- Then, compile/load/run your program.
- Measure and record the output amplitude at every frequency and plot the measured magnitude response with respect to the frequency
- Compare the measured magnitude response to the magnitude response on fdatool to verify the characteristics of the running filter.
- Report your results in your lab report.

### Lab 3d. Implementing FIR Filter Computation Using Circular Input Buffer

- Copy your project files for lab3c into separate directory. Now, you need to modify the main FIR computation in `transmit_ISR()` interrupt handler to reduce the movement of input samples to save clock cycles. This type of approach is critical for DSP programming. To make a proper modification, consult the lecture slides which are distributed in the lab.
- Then, compile/load/run your program.
- Measure and record the output amplitude at every frequency and plot the measured magnitude response with respect to the frequency
- Compare the measured magnitude response to the magnitude response on fdatool to verify the characteristics of the running filter.
- Report your results in your lab report.