Moustafa Elsayed 201600848

Phase #2 Report

Table of contents:

* Calculation Mode (GUI – Results - Discussion)
* FIR filter desing using least square error (GUI – Results - Discussion)

Calculation Mode:

Theory:

There is no much theory in this tab as it is only some calculation of the past phase.

The power is summation of the fft values squared

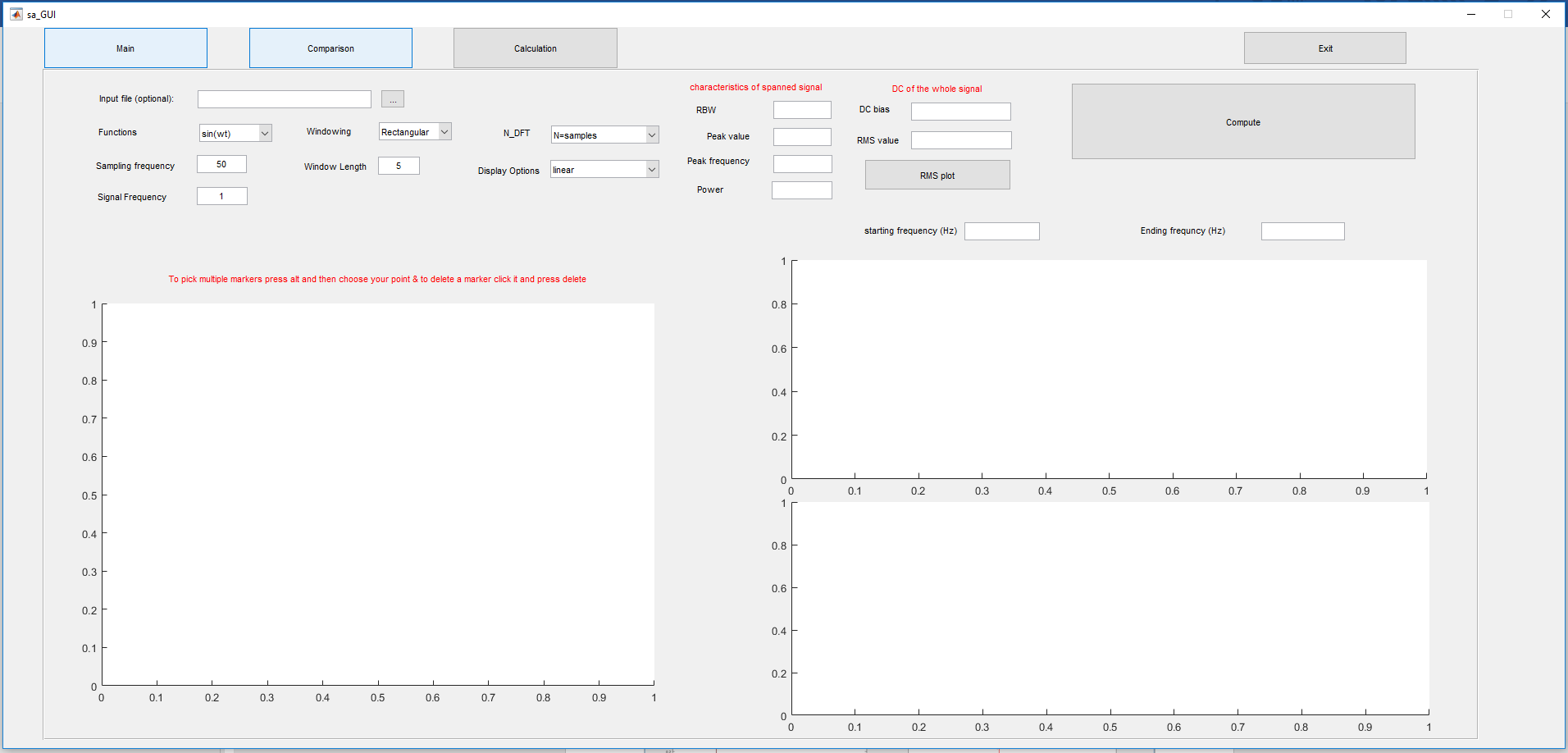
The DC bias is concluded from the first sample in the frequency domain if =0 then there is no DC bias if it has a value then it is the where N is the number of fft points

The GUI:

In Cacluation Mode we are trying to extract features of the signal we are displaying . We can see same parameters of the chosen function same as the last phase in main tab and also there are some new fields

If the usere clicked the calculation tab it will show this window

* The left axes is used for time domain signal and for RMS plot
* The upper right axes is used for spanned frequency
* The lower right axes is used for whole frequencey range from 0:fs



Those feature are :

1. Resolution band width: here I used the equation ( where fs is the sampling frequncy to get the resolution of my DFT
2. Peak Value : This gives me the magnitude of the peak value of the spanned frequency using the built\_in funciton max
3. Peak frequency: This computes the frequency value where it has the highest magnitude
4. Power of the spanned frequency: This is the
5. DC bias : This can be easly computed using the inverse DFT of the first value of the DFT
6. RMS (root mean square) value: this is the formula of RMS value 
7. Display options:

In this mode (unlike other modes) the user can only view only one option from this set of options (Linear – Logarithmic – Phase – Real \_ Imaginary)

1. And RMS Plot :

I used it a frame composed of 20 samples of the signal then applied fft to this frame and moved to the next 20 samples till the end of the signal and averaged all fft’s from all frames code will be attached to this report

There are also other features that are already like the same as the previous phase like determining window length ,N\_DFT , Different functions,.. etc. (You can find the capapilites of those in Phase #1 report)

Error Handling

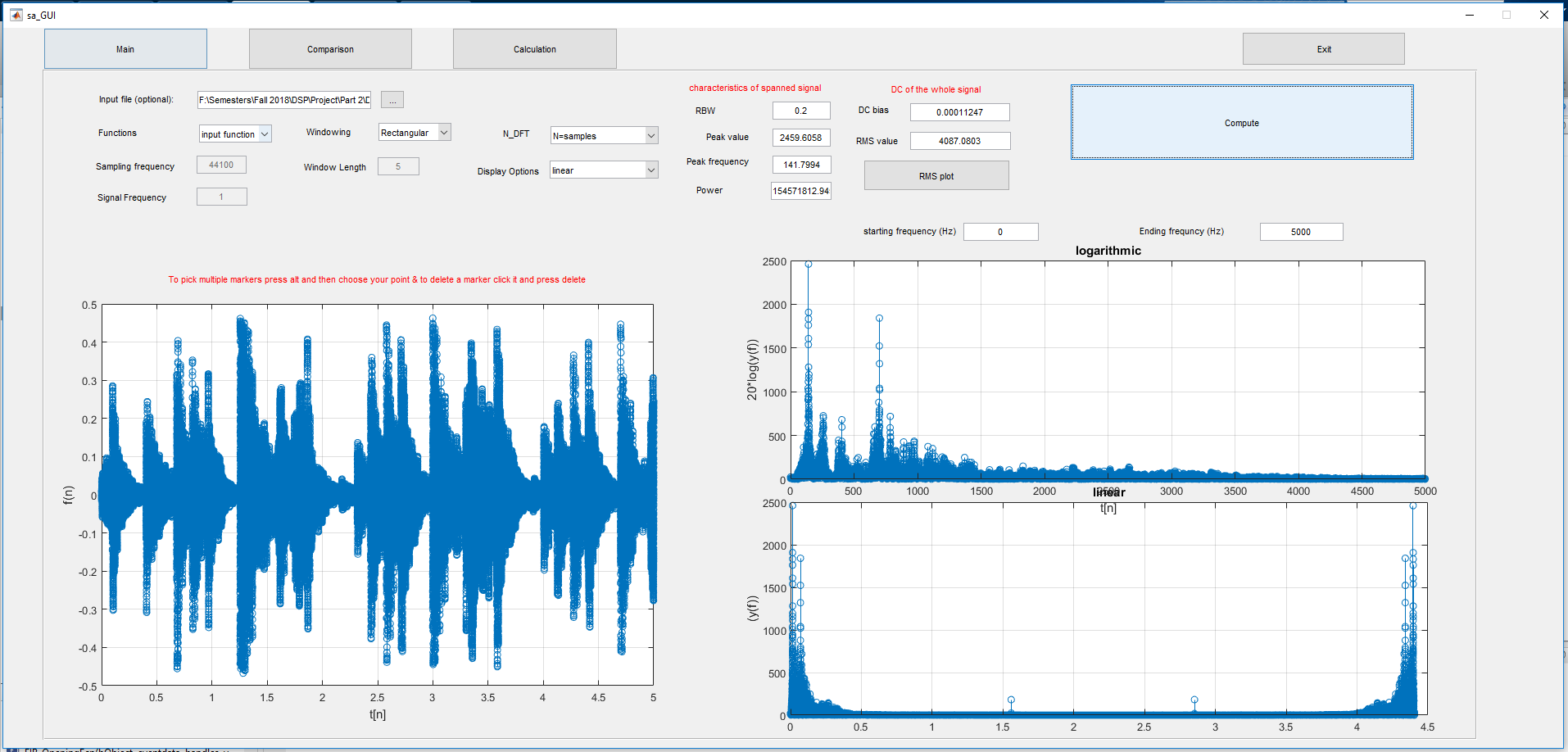
1. The user can not input a wrong span ( neither empty or letters will give an error and I also handled more cases of the users misinputing)
2. The same error handling used in Phase #1

Example 1 & discussion (calculation mode):

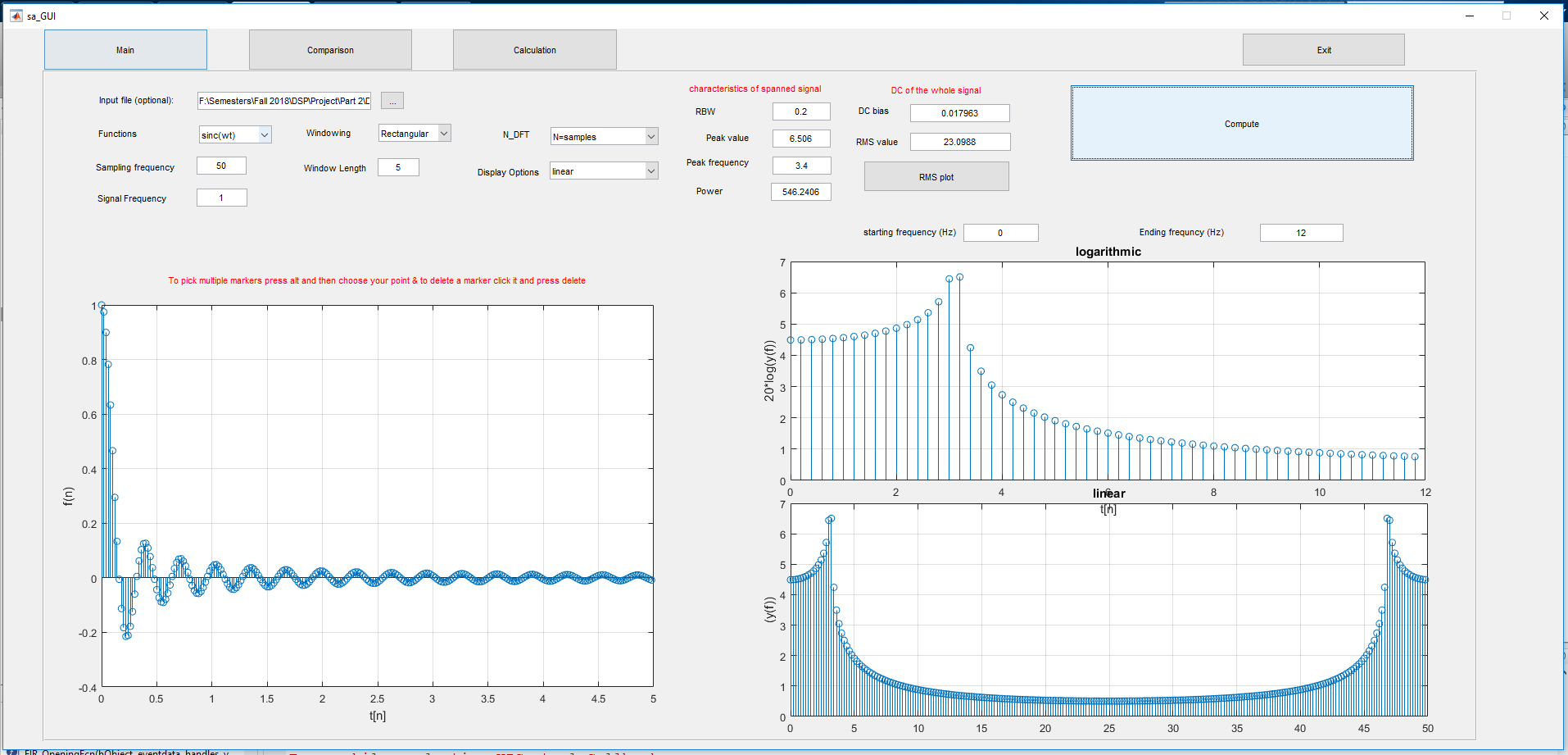
This is the input function of the user and both spanned frequence and the whole frequency span.

We can see that:

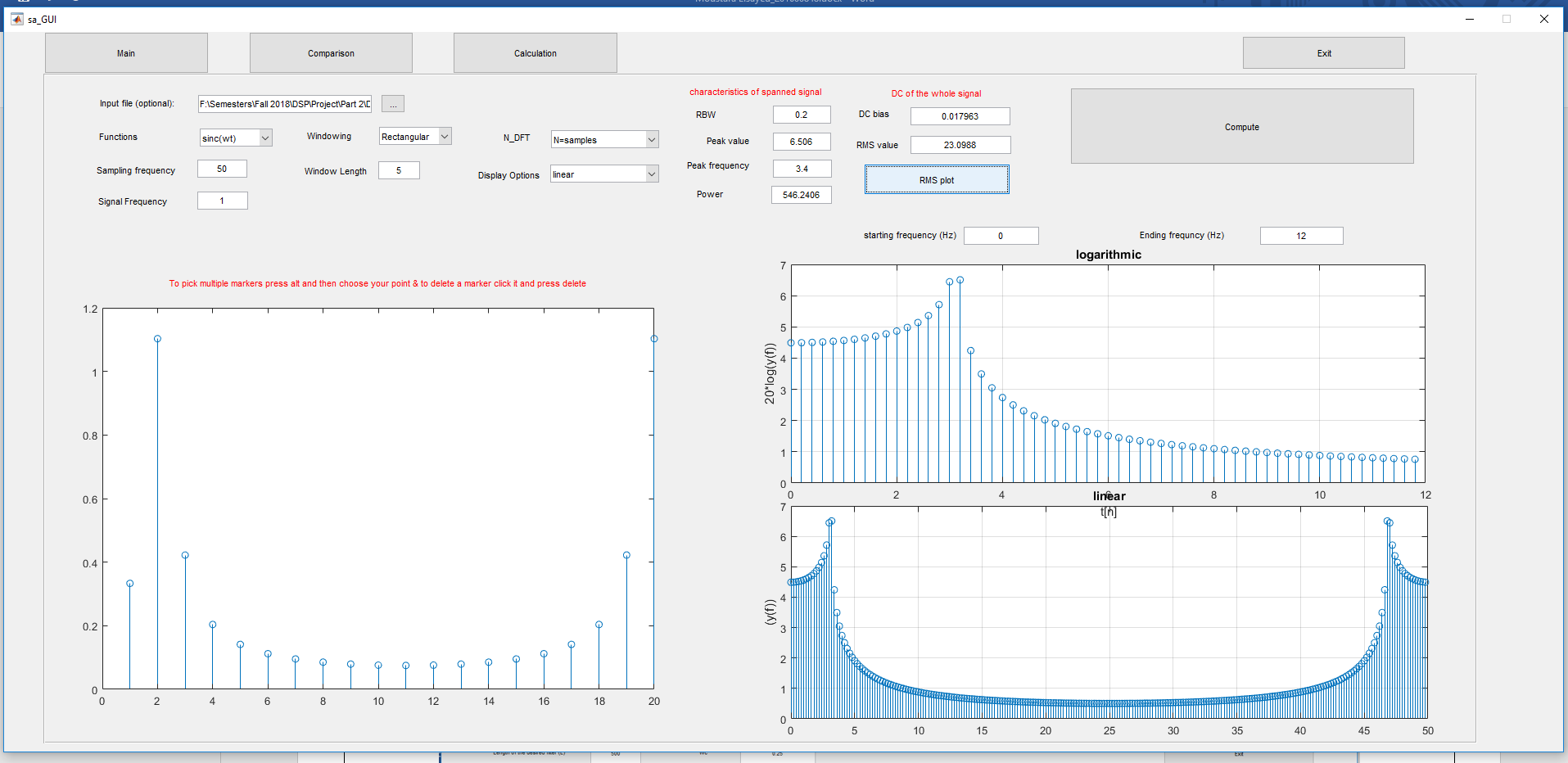
* Peak value is almost 2500 as spected from the graph. Peak frequency is 141 also as expected
* Also we observe that the DC bias is almost 0 as it repeats signal is not shifted away from 0 in time domain.



Here is another input



This is also the sinc but the left plot is the RMS Plot



FIR Theory:

The theory is that we start from the end not the start and we assume even symmetry of response so we can eliminate sin components and keep the cosine components of the DFT also we make matrix manipulation to solve (N-1)/2 equations (independent tabs that cannot be derived from other tabs) .

There is also other method called weighted least squares where we assign certain weight to each point. And this comes in handy in application that really cares about only one thing like (only passband in case I want the passband to be perfect) or only the stopband if I really care about attenuating anything else the passband.

FIR design GUI:

User input:

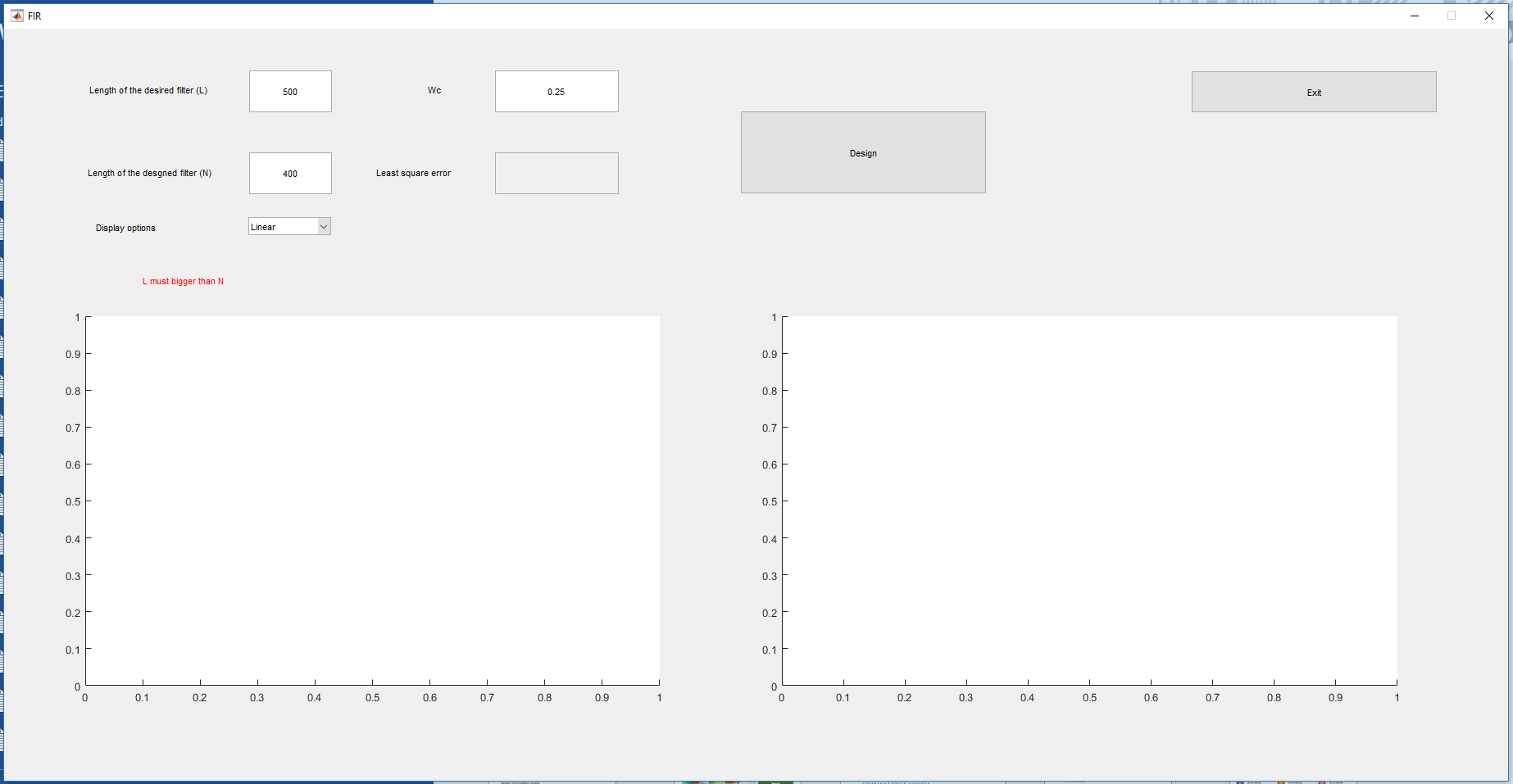
* L is the desired filter length and N is the designed filter length.
* Normally L is bigger than N so we use the least square algorithm to compute an accurate filter with the desired features
* Wc is the cutoff frequency. (the user gives me an input and I multiply that input by )

Features:

* I design the filter using the least square error method and show its linear frequency response compared to the desired frequency response in the left axes and on the right axes I view the logarithmic frequency response.
* I also compute the least square error and view it on the GUI

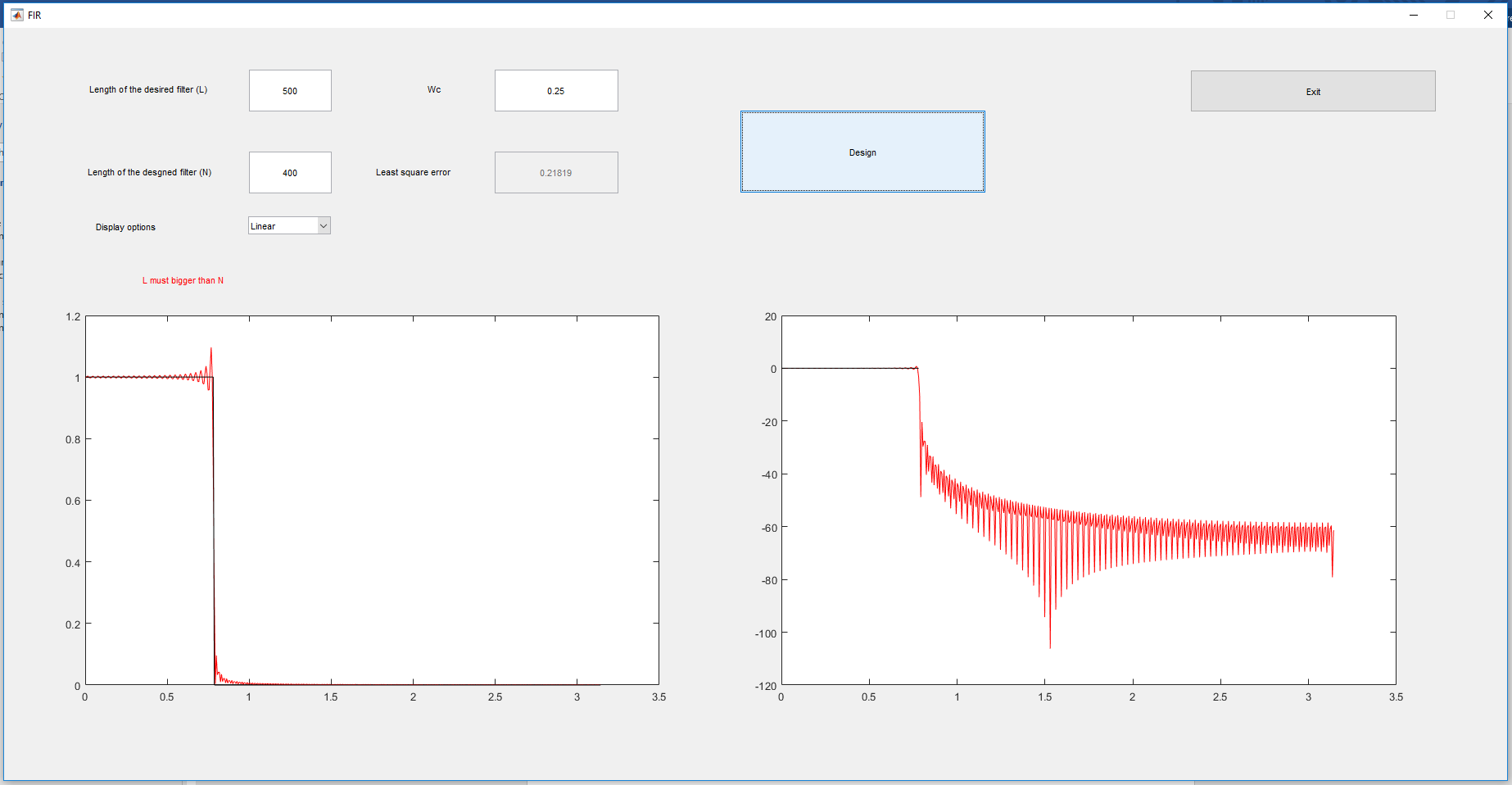
Error handling:

* The user cannot make N>L
* The editing text of the least square error is disabled



Example:

Here I made L = 500 and N = 400 and then we had a filter with order N-1 and it is pretty accurate as expected.



Note: I added the code of the weighted least square and made the weights function but for some unknown reason the code gives the same result as the firls filter so I commented this part so you can check it

References:

* Professor Lecture notes to implement FIR design
* Dr Lyon’s book (understanding digital signal processing)