Elektronica Project -2

Lakshitha Totakura

Explanation of Steps Performed

1. Firstly installed the DSP Toolbox from Add Ons.

2. Then typed `filterDesigner` in the Command Prompt. Then selected Low Pass Filter, FIR (Equiripple), Minimum order, set the Fpass = 3000, Fstop = 5000, density Factor = 20, Fs = 48,000.

3. Design Filter → File → Generate MATLAB Code. Generated the `.m` file of `Elektronica_Filter_Fina` Saved the file in a folder.

4. Wrote a code for the FFT of the original signal and the frequencies present, to plot the unfiltered signal and the filtered signal in the time and frequency domain.

Filter design choices and justification

1. Used FIR (Finite Impulse Response) because it settles to zero in a finite time. So it is simpler and more efficient to implement a filter upon. It is also a more stable option.

2. An equiripple filter (used) minimizes the maximum error (like the Chebyshev inequality does) between the desired and actual frequency response.

3. Used minimum order of the filter, it is lesser in efficiency compared to higher order filters, but simpler to analyze.

4. The low pass filter allows frequencies below 3000Hz (Passband) and highly attenuates those above 5000Hz.

5. The sampling period of 48000, is taken as it is a standard rate.

MATLAB code used for signal generation, filtering, and FFT analysis.

Code for Filter

```
function Hd = Elektronica_Filter_Final
% Equiripple Lowpass filter designed using the FIRPM function.

% All frequency values are in Hz.
Fs = 48000;  % Sampling Frequency
```

```
Fpass = 3000;             % Passband Frequency
Fstop = 5000;             % Stopband frequency
Dpass = 0.057501127785;  % Passband Ripple
Dstop = 0.0001;          % Stopband Attenuation
dens  = 20;              % Density Factor

% Calculate the order from the parameters using FIRPMORD.
[N, Fo, Ao, W] = firpmord([Fpass, Fstop]/(Fs/2), [1 0], [Dpass, Dstop]);

% Calculate the coefficients using the FIRPM function.
b  = firpm(N, Fo, Ao, W, {dens});
Hd = dfilt.dffir(b);
```

Code for filtering and FFT Analysis

```
% Final_Filter_Analysis.m

% Sampling settings
Fs = 48000;
t = 0:1/Fs:1-1/Fs;

% Frequencies to include (both inside and outside the passband)
frequencies = [500, 1000, 2000, 2500, 3000, 3500, 4000, 4500, 5000, 6000];

% Generate the mixed signal
signal = sum(sin(2*pi*frequencies'.*t), 1);  % 10 sine waves added

% Plot original signal in frequency domain
N = length(signal);
f_axis = Fs*(0:N/2-1)/N;
Y = abs(fft(signal));

figure;
plot(f_axis, Y(1:N/2));
title('Original Signal - Frequency Domain');
xlabel('Frequency (Hz)');
ylabel('Magnitude');

Hd = Elektronica_Filter_Final;
b = Hd.Numerator;
a = 1;  % FIR filter


filtered_signal = filter(b, a, signal);

% Plot filtered signal in frequency domain
```

```
Yf = abs(fft(filtered_signal));
figure;
plot(f_axis, Yf(1:N/2));
title('Filtered Signal - Frequency Domain');
xlabel('Frequency (Hz)');
ylabel('Magnitude');

% Time domain plots
figure;
subplot(2,1,1);
plot(t, signal);
title('Original Signal - Time Domain');
xlabel('Time (s)'); ylabel('Amplitude');

subplot(2,1,2);
plot(t, filtered_signal);
title('Filtered Signal - Time Domain');
xlabel('Time (s)'); ylabel('Amplitude');
```

Plots of input and output signals in both time and frequency domains.

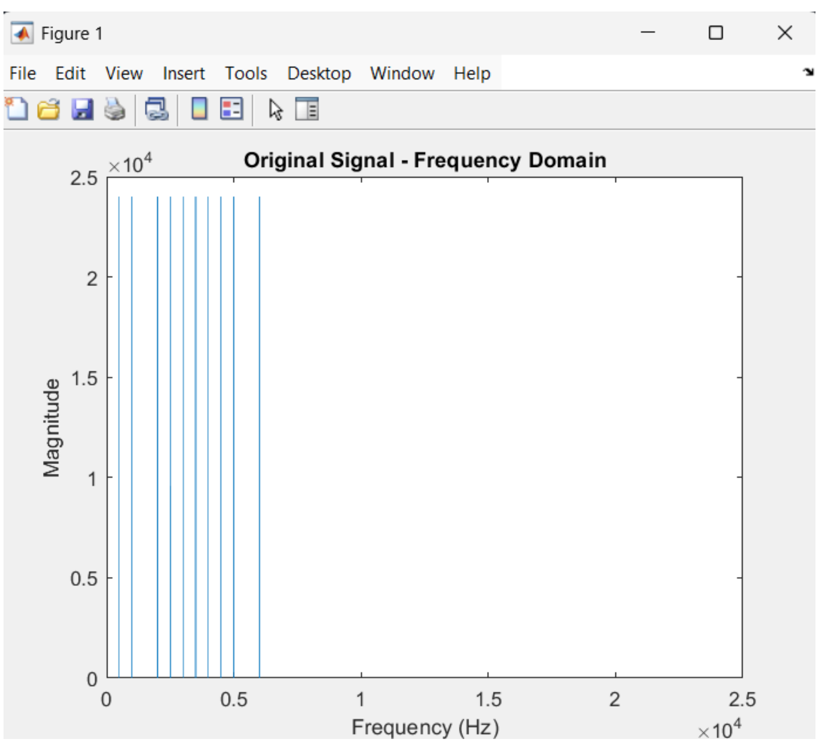

Figure 1: Original Signal - Frequency Domain

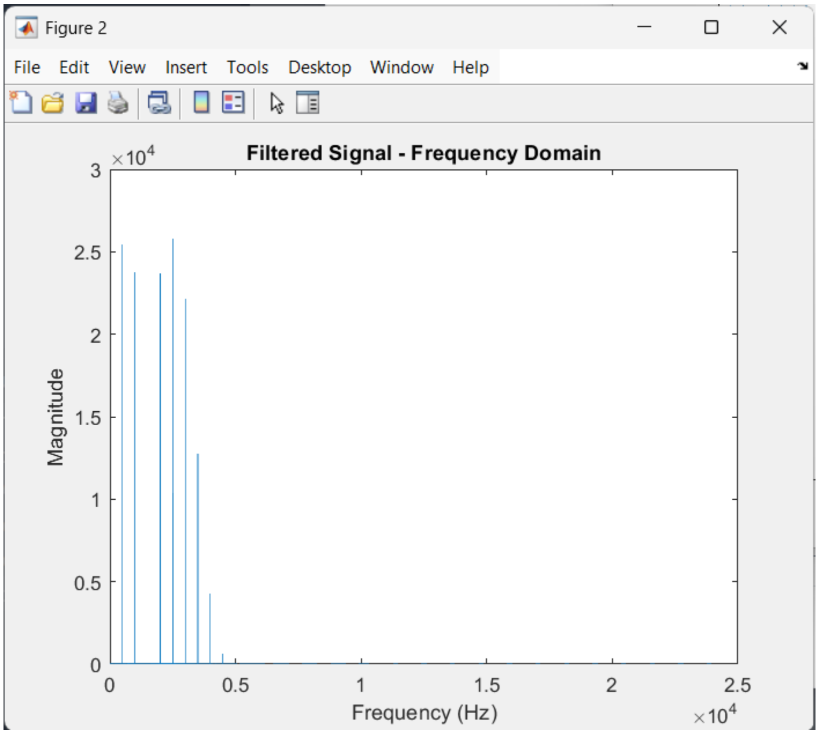

Figure 2: Filtered Signal - Frequency Domain

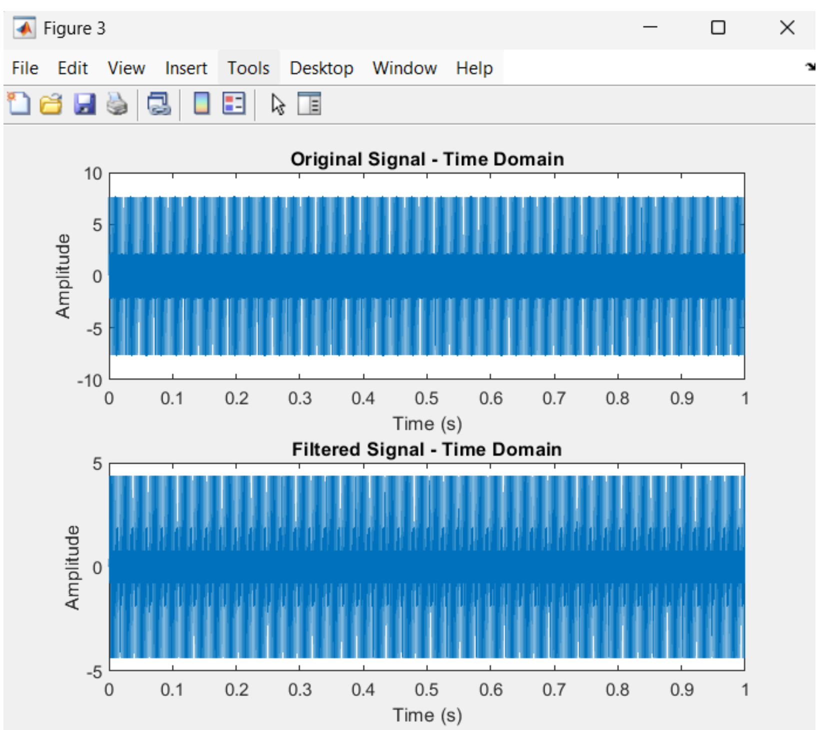

Figure 3: Filtered and Original Signal - Time Domain

Observations and Conclusions

- Frequencies = [500, 1000, 2000, 2500, 3000, 3500, 4000, 4500, 5000, 6000]
- In the original signal plot, the 10 given frequencies are clearly visible as spikes.
- In the filtered signal, Frequencies below 3000Hz are almost unaffected, while those between 3000 and 5000 Hz are partially allowed, their magnitude slowly reducing. The frequencies above 5000Hz are almost all attenuated.
- The Fig 3, Original Signals' Time Domain signal is relatively more crowded and noisy compared to the filtered signal, where more of the higher frequency signals are filtered out.
- In Fig 1, the original signal composed of the 10 frequencies is shown, whereas in Fig 2, the filtered signal is shown. Here the filtering is more visible.
- Only the low frequency components of the 10 sine waves remain.