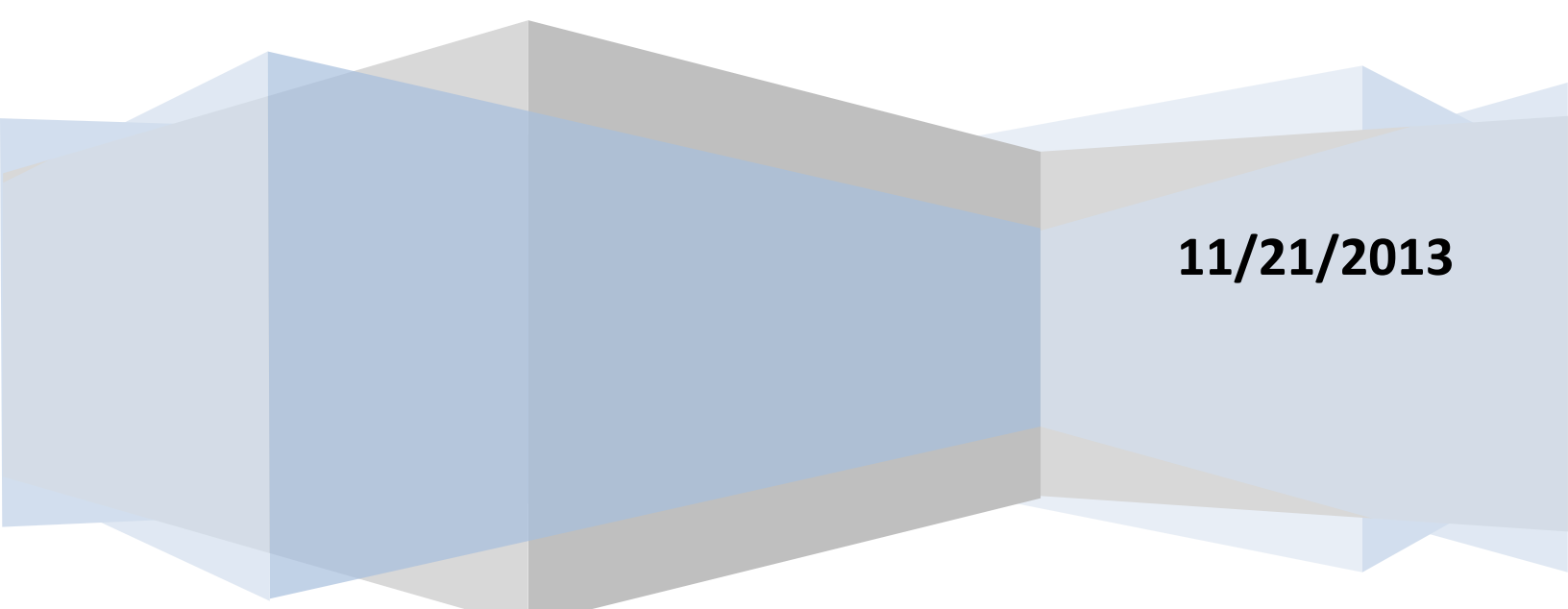# Digital filters

Assignment 05

Arjun Darbha

Instructor: Dr.Jason Blough

TA: Craig Reynolds

11/21/2013

Abstract

The purpose of this experiment is to implement different digital filters on the signals provided and decompose the signal into its components. Two basic types of digital filters, FIR and IIR filters can be designed using different algorithms. A digital filter is designed for a particular purpose based on different criteria like stop-band attenuation, transition band length, order of the filter, computational efficiency and so on. In this experiment, spectral analysis is performed on each time-history provided and the signal characteristics of each component signal is hypothesized based on the pattern of frequencies in the spectrum. In case of harmonics, at least five harmonics have been captured to generate component signal in time domain. While designing the filters, it has been observed that FIR filters are stable when compared to IIR filters especially at higher orders. However, computational efficiency of FIR filters is much lower when compared IIR filters in most cases which could be noticed in terms of higher computational time while applying FIR filters. It is concluded that this is due to the higher number of coefficients involved in case of FIR filters. It has also been observed that IIR filters should be applied with caution by carefully analyzing the signal in time-domain after the applying the filter.

1 Introduction

A digital filter is an algorithm or a mathematical function which converts digital input to digital output with desired signal characteristics. Different types of filtering that can be performed on the input data include

- Low-pass filtering
- High-pass filtering
- Band-pass filtering
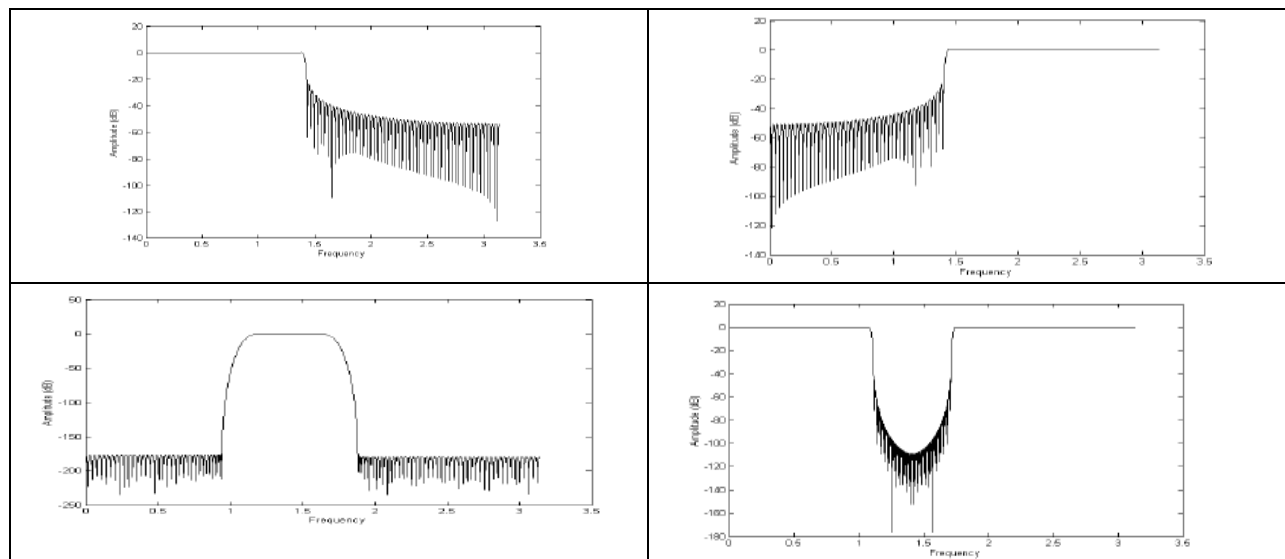- Band-stop filtering
- Data smoothing
- Averaging

Figure 1 (Clockwise) Low-pass, high-pass, band stop and band pass digital filters

As this filtering technique does not require hardware components, digital filter design is less complicated when compared to analog filter design. The parameters of the filter can be tweaked and a suitable filter, for a particular application, can be developed iteratively. Digital filters are used for various applications, some of them include,

- Digital anti-alias filtering in data acquisition systems
- Decimation
- Noise reduction
- Fixed and adaptive interpolation
- Order tracking

This is not an exhaustive list of applications as digital filters are used absolutely everywhere where signal processing is involved. When compared to analog filters, digital filters have certain advantages as discussed below

- Accuracy – Digital filters can be developed either in Laplace domain or by performing an inverse fft on the desired frequency spectrum. As digital filters are mathematical equations, characteristics of a digital filter can be fully understood
- Repeatability – As there are no hardware components involved, there is no problem of aging and the results acquired by a particular digital filter are highly repeatable
- Implementation – Digital filters can be hard-coded into the chip or can be coded for application after the data is acquired by DAQ
- Flexibility – It is possible to design almost any shape using digital filters. Most signal processing toolboxes have a frequency versus gain plot which can be manipulated to get a filter of desired shape
- Cost effective – Digital filter is just a piece of software of code. As this does not require any hardware components for implementation it is the most cost-effective
- No channel to channel variation – All that matters is to check if the filter behaves properly and as desired. If same filtering application is performed on all channels, same piece of code can be used without introducing any variation in the filtering process

Different types of digital filters available are

- Finite impulse response – FIR
- Infinite impulse response – IIR
- Adaptive IIR and Adaptive FIR
- Kalman filtering
- Long FFT filtering
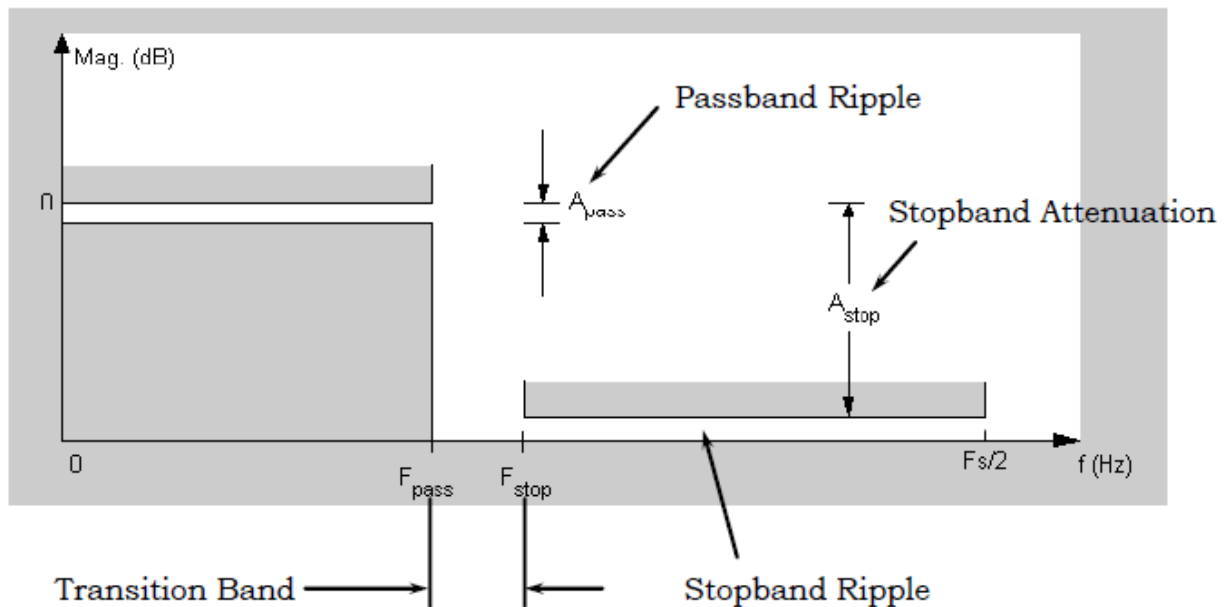
Important parameters of a filter design

Figure 2 Filter design parameters

The filter design schematic in figure 1 describes a low-pass filter

Passband – These are frequencies through which data passes without any change in the frequency. Ideally, this band should not introduce phase delay. However, all the filters introduce some phase delay, be it linear or non-linear

Passband ripple – This is the variance in gain in pass-band

Stopband – These are the frequencies in which the gain is attenuated. The algorithms are not programmed for agreeable phase characteristics in stop-band.

Stopband ripple – This is the variation of gain in a stop-band

Transition band – It can be seen in figure 1 that the transition from start band to stop band occurs over a small band of frequencies. This band is called the transition band. Transition band changes when different filters are used. For example, butterworth filter has a greater transition band when compared to elliptical filters

Roll-off rate – It is defined as the attenuation rate of the gain in the transition band. Roll off rate is one of the most important characteristics that help in choosing a filter

Stopband Attenuation – Stopband attenuation is defined as the attenuation in the gain of the filter in the stopband. It is advisable to have the stopband attenuated to the noise floor. This parameter is also filter dependent and varies drastically when different filters are used

FIR filter

An FIR filter is mathematically defined as

$$y(n) = \sum_{k=0}^{N} a_k x(n-k) \tag{1}$$

where, y(n) is the filtered data, x(n) is the actual data and a_k denotes the coefficient that each data point is multiplied with. In equation 1, k denotes the number of data points used to calculate the filtered value at a particular data point. Higher the value of k, higher the number of coefficients and higher is the order of the filter. These coefficients are also called taps.

From equation 1, it can be observed that an FIR filter relies only on the data to calculate filtered output. It does not rely on any filtered output values. This makes the filtered output stable. It can also be deduced from equation 1 that an FIR filter has a finite startup transient. Startup transient is dependent on the number points after which the filter coefficients get multiplied by actual data points. If the order is higher, the number of coefficients is higher and hence, the startup transient is higher.

Other advantages of using an FIR filter include a filtered output with a linear phase and a fairly easy implementation of the algorithm when compared to other filtering techniques. One of the major disadvantages of using an FIR filter is that it is computationally demanding. This is because desired characteristics from a filter are achieved at higher orders when compared to IIR filters.

FIR filter methods can be applied in three ways

- Window method – In this method, typical windows like box-car, hanning, kaiser etc are used to obtain the desired shape in frequency domain. Window shape can be changed by adjusting the order of the filter
- Least squares – In this method, the algorithm chooses a closest fit to the desired shape by using a least square curve fit. The output is a filter with minimum number of coefficients for a particular shape
- Equiripple – As the name suggests, this filter has equal ripple in the stop band. The number of coefficients is a parameter that has to be provided as a user input

IIR filter

Mathematically an IIR filter is defined by equation 2

$$y(n) = \sum_{k=0}^{N} a_k x(n-k) + \sum_{k=1}^{M} b_k y(n-k) \tag{2}$$

When compared to equation 1 for the FIR filters, this filter has an extra 'b' coefficient which depends on the filtered values. This has both advantages and disadvantages. As filtered data points are used while calculating the output, the number of coefficients involved in an IIR filter equation is less than that of FIR filter coefficients. Hence it is computationally efficient when compared to FIR filters. However, use of filtered data points makes the output unstable in some cases. Other disadvantages of using an IIR filter include non-linear phase, infinite start-up transient and sensitivity to round-off error. IIR filters are usually used to generate typical shapes of a transfer function and it is difficult to design unorthodox shapes.

IIR filters are based out of analog filters and possess similar names. Some of the common IIR filters are,

- Butterworth – This filter has a very flat passband but a gradual roll off rate
- Chebyschev Type I – This filter has a sharp roll-off rate and a flat pass band. However, the filtered output has a high non-linear phase
- Chebyschev Type II – This filter has a sharp roll-off rate and some non-linear phase in the filtered output.
- Elliptic – This filter has the shortest transition width, but has equiripple- in both stop and pass bands. Also the filtered signal has significant non-linear phase

From the discussion above, it is clear that filter design process is a trade-off between ripple, transition and stop band attenuation. Different applications require different filters. Based on the characteristics described above, filter has to be selected for a particular application in such a way that the desired spectrum is obtained.
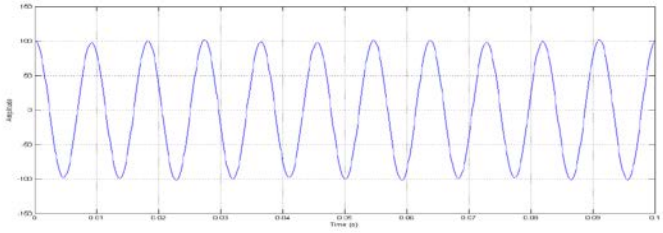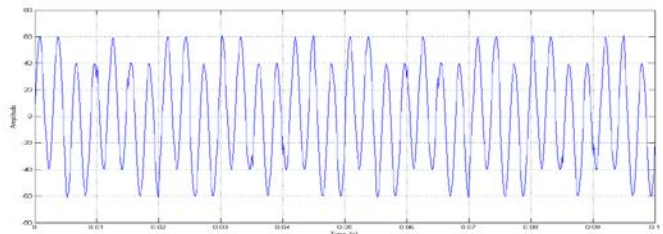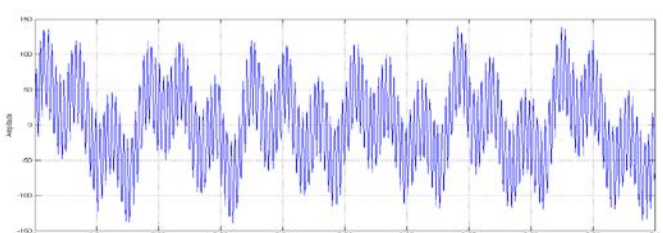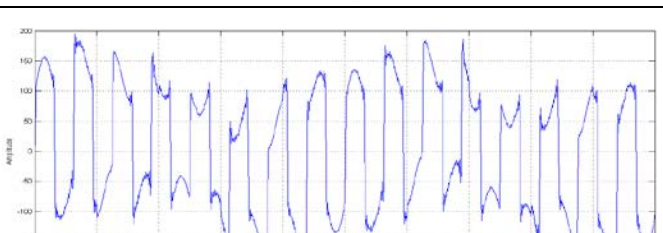
The objectives of this experiment are:

- To import the signals provided into matlab's "sptool" for analysis and filtering
- Generate FFT of the signal and analyse the spectrum. Also analyze frequencies at which there is significant energy and hypothesize different signals present in the time history
- Design digital filters in sptool and apply the designed filters to the time history in such a way that spectrum contains frequencies only from hypothesized signals. Repeat this step until all the components in the signal are obtained
- Use different techniques to design filters and come up with the most optimal design for the filter such that the required frequencies can be easily differentiated from the spectrum
- In case of each time history, export the decomposed signals and filtered spectrums back to workspace and generate zoomed-in plots for each signal

3 Data and measurements

Zoomed-in time history of signals is presented in table-1. Non-zoomed in graphs have not been displayed because no important conclusions can be made by looking at the entire time history.

Table 1 Table denoting the time histories of corresponding signals

Signal	Time history (Zoomed in)
Signal-1	
Signal-2	
Signal-3	
Signal-4	

3 Experimental proceudure

Four matlab signals have been provided from four channels with a single time vector. Mean of time difference between consecutive time observations is calculated. This value is Δt and is evaluated as 1e-04. From this sampling rate can be calculated as

$$F_s = \frac{1}{\Delta t} = 10000 Hz \qquad (3)$$

Sampling rate is a constant value for all the four signals.

The signals are imported into 'sptool', the signal processing toolbox provided by matlab. Following steps have been followed to analyze each signal separately in the tool box

- Import the signal into the toolbox with a sampling rate of 10000Hz
- Create a spectrum using FFT option in the spectrum viewer. In order to create the spectrum a block size of 4096 has been used. A hanning window is applied before generating the data and block averaging is performed for obtaining the spectrum of a signal. The frequency resolution can then be calculated using

$$\Delta f = \frac{1}{N\Delta t} = \frac{1}{4096 * 1e - 04} = 2.44 Hz$$

  This means that frequencies can be differentiated in a spectrum with a step size of 2.44Hz

- Frequencies are then estimated using vertical markers provided in the spectrum viewer. If the frequencies follow a pattern, for example, if there are frequencies at odd harmonics of the fundamental, then it is concluded that they are from a square wave of frequency equal to fundamental frequency. If the frequencies do not follow a pattern, it is concluded that these energies are from signals at corresponding frequencies
- If a square wave is detected in the spectrum, effort is made to apply band stop filters and eliminate frequencies between harmonics. Based on the requirements, IIR or FIR filters have been used. However, if an IIR is used, the filtered signal is carefully analyzed in the time domain because of instability issues caused by IIR filters. It has also been attempted to capture atleast five harmonics in case of a square wave. The filtered outputs are obtained in time-domain
- In order to obtain independent frequencies which do not follow a pattern, band pass filters are used, for one frequency at a time. These frequencies are then transformed into time domain
- To consolidate, all frequencies are overlaid on top of each other by exporting the filtered time histories to matlab

4 Data Analysis and interpretation

Signal 1

FFT performed on this signal reveals that it contains energy only at one frequency at 110Hz (See figure 3). The time history is also plotted in the figure 4. It can be concluded from this figure that amplitude of the signal is 100 units.
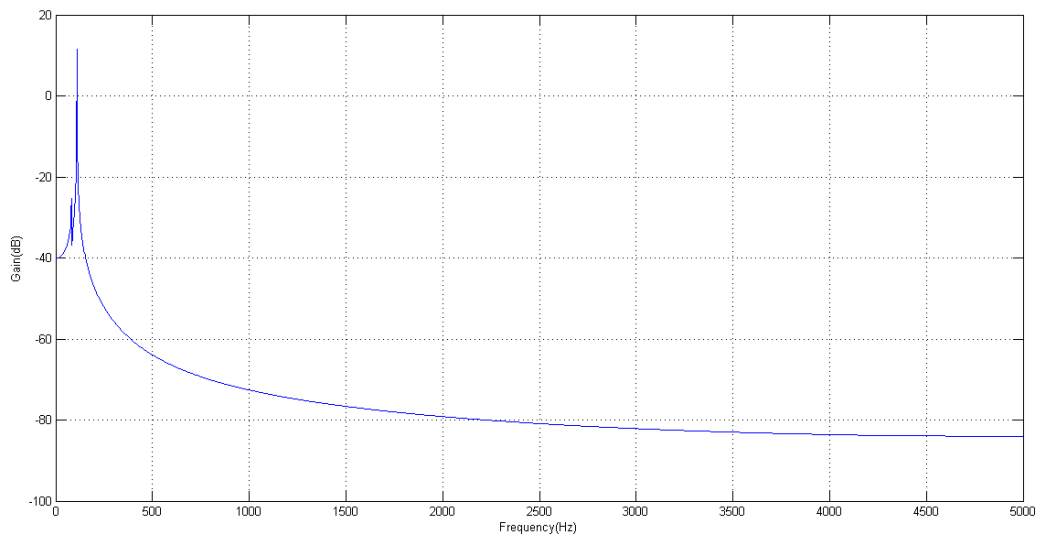
Figure 3 FFT performed on signal 1 (Sin wave 110 Hz)

Signal 2

Spectral analysis of signal 2 is performed by performing an FFT on the time history and the spectrum is presented in figure 4.
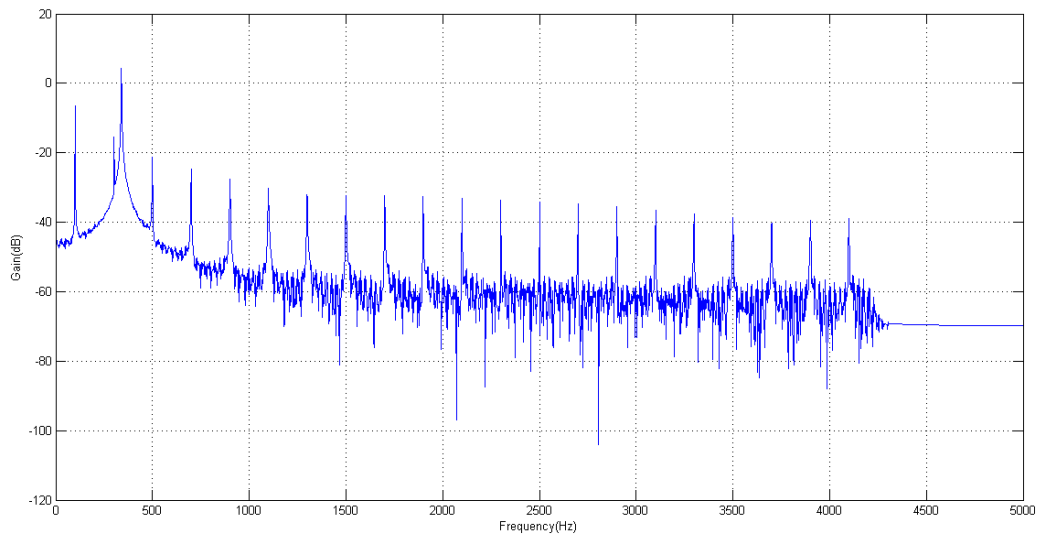

Figure 4 FFT of the time history before filtering the frequencies

Following observations can be made from figure 4

- Spectrum has odd harmonics frequencies at 100Hz, 300Hz, 500Hz so on. These frequencies form a square wave with a frequency 100Hz

- Spectrum has significant energy at 340 Hz. As this signal is not a part of the pattern, it can be concluded that this spike is generated by a standalone signal at 340Hz

To filter the signal, schematic described by figure 4 has been followed. Signal 1 in the schematic is the actual signal which is imported to sptool and digitally filtered.

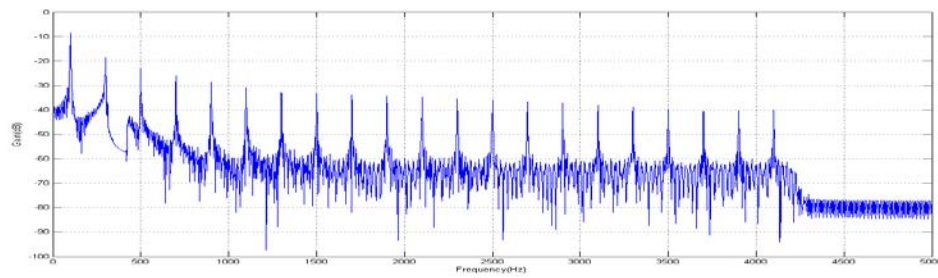- Actual signal is filtered using an FIR band stop filter to obtain square wave's spectrum

Figure 5 100Hz square wave spectrum

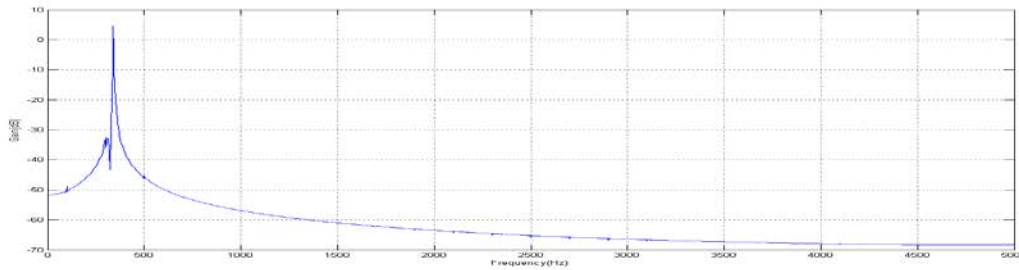- FIR band pass filter is used to obtain the energy at stand-alone frequency of 340Hz

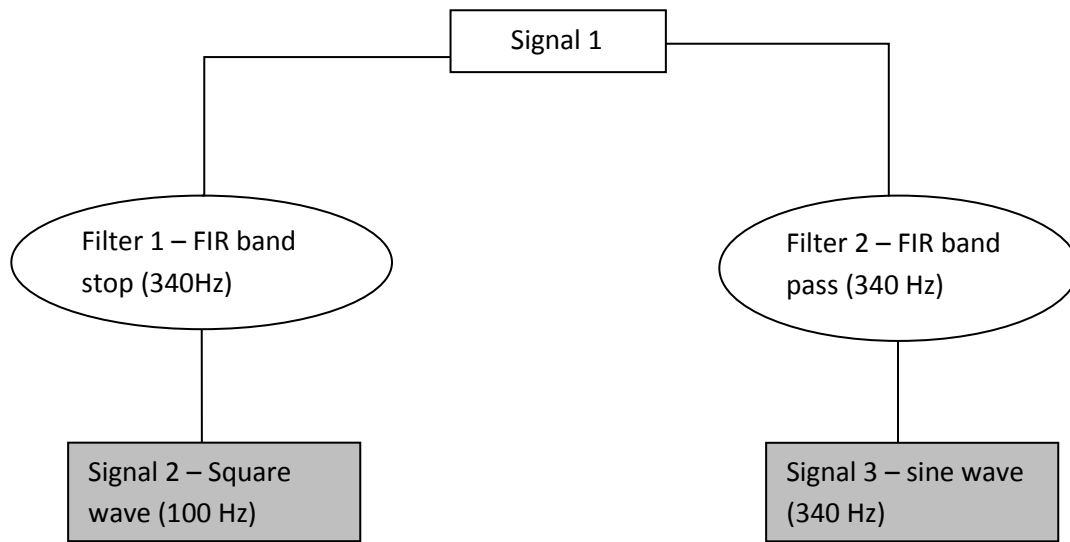**Figure 6 340Hz square wave spectrum**

Figure 7 Schematic of signal decomposition process using digital filters (signal 1)

Table 2 represents the characteristics of the two digital filters used to filter the signal out. FIR filters are used for both band-stop and band-pass. Kaiser window method has been used to design these filters and the filter order is obtained automatically by using the "minimum order" feature of the tool. It has also been noticed that a higher order filter provides more stop band attenuation when compared to a lower order signal; however such attenuation is not required as any attenuation below the noise floor of 40dB below zero is not worthy to achieve.

Table 2 Analysis of digital filters used for decomposition of signal 2

Filter #	Response Type	Design Method	Options used	Filter order	Frequency specifications	Stopband attenuation
1	Bandstop	FIR	Kaiser window	1814	Fpass1=300,Fstop1=320 Fstop2=400,Fpass2=450	60dB
2	Bandpass	FIR	Kaiser window	5019	Fstop1=320,Fpass1=330 Fpass2=350,Fstop2=360	60-80dB

The overlay of time history of the two signals obtained by filtering the data digitally is presented in figure 8. This figure is plotted by zooming into the time-history much farther than the start point so that startup transient could be avoided.
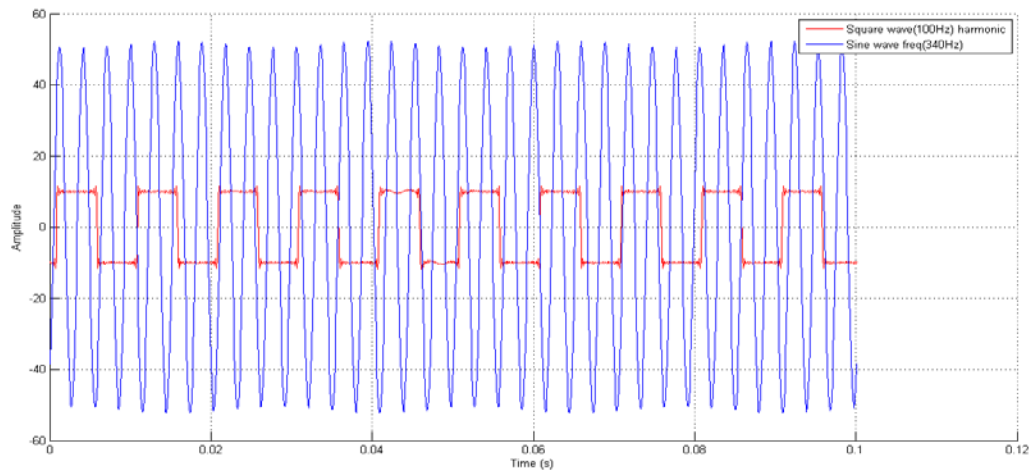
Figure 8 Signal overlay of signals used in signal 2

To better understand start-up transient, a zoomed in view of the first few points of the time-history has been presented in figure 9
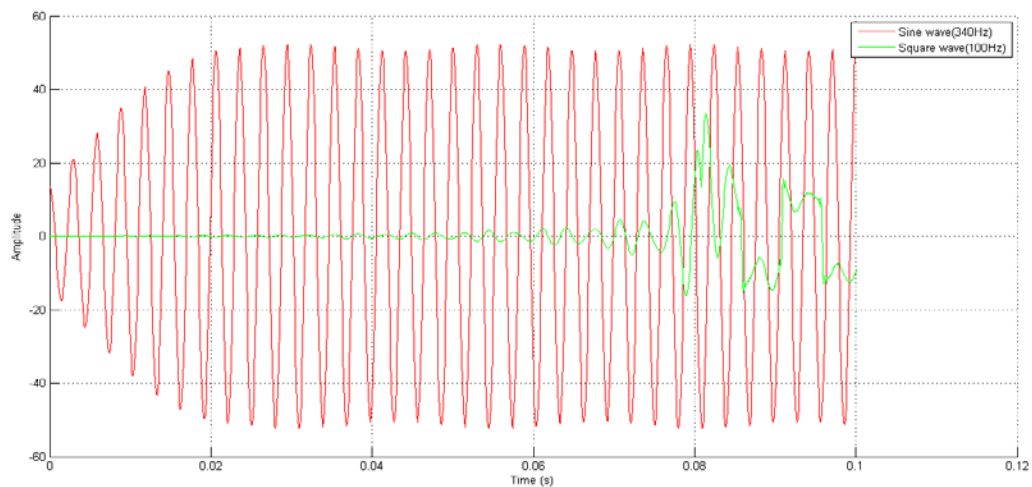

Figure 9 Start-up transient (Signal 2)

 It can be clearly seen that first few points in the time-history are distorted and the signal gradually evolves into the actual signal. Because an FIR signal is used, start-up transient is finite. This distortion is encountered in while filtering all the signals during this experiment. To avoid start-up transient, initial parts of the signal have been avoided while plotting zoomed-in views of signal overlay.

To illustrate the stability issues of an IIR filter, 500 order butterworth bandstop filter (Fc = 340Hz) is generated and the filtered time-history is presented in figure 10
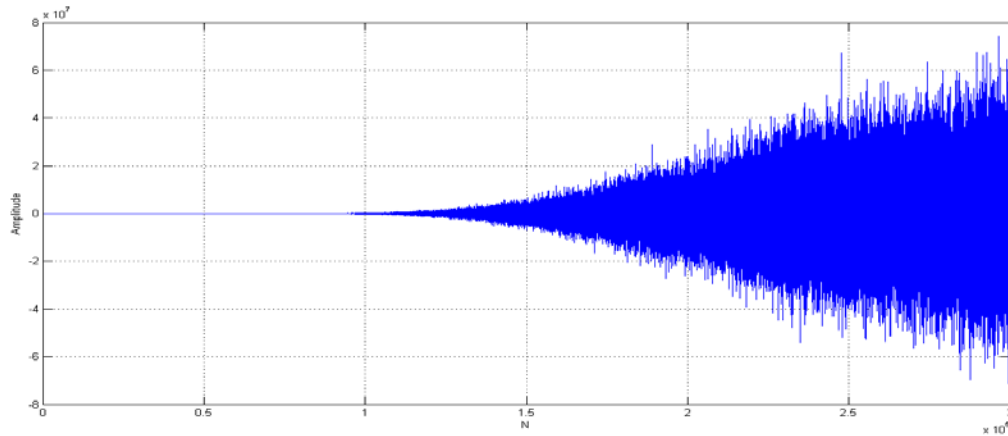
Figure 10 Filtered time history of unstable IIR filter

It is evident from this figure that IIR filters can result in unstable time histories when not applied with carefully

Signal – 3

Spectral analysis (figure 11) has been performed by importing the signal into matlab's 'sptool'.
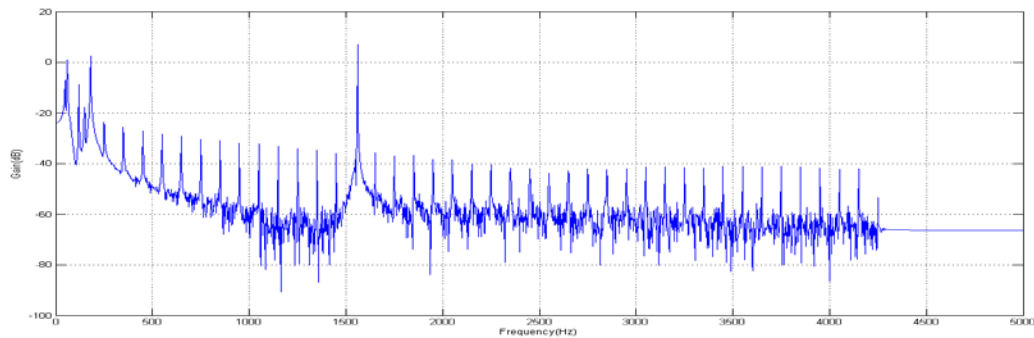

Figure 11 FFT of signal 3 before digitally processing the data

Following observations can be made from spectrum in figure 11

- A pattern of harmonics can be easily noticed. On further investigation, it has been revealed that these form odd harmonics of 50Hz frequency. Therefore these frequencies form a square of frequency 50Hz
- A huge spike after 1500Hz on zooming revealed a frequency at 1560Hz. This frequency evidently carries maximum amount of energy in the signal
- The other spikes in the frequency band include energies at 60Hz, 120Hz and 180Hz. These frequencies do not form a pattern and therefore it is concluded that they contribute individually to the spectrum, in the form of sine-waves

From these observations it can be concluded that the signal consists of five components – One square wave of 50Hz frequency and four sine-waves of frequencies 1560Hz, 60Hz, 120Hz and 180Hz. The process followed to extract each signal from the time history is presented in the form of a schematic in figure 14. The schematic is summarized below

- Signal 1 is the time history of the actual signal. An IIR low pass filter is applied such that all the frequencies above 1500 Hz are eliminated and the only. Later, two band-stop filters are applied successively in order to eliminate 60 – 120 Hz band and later the frequency at 180 Hz. This ensures that the spectrum has odd harmonics of 50Hz. The time history of this signal is obtained as signal 4 in the schematic
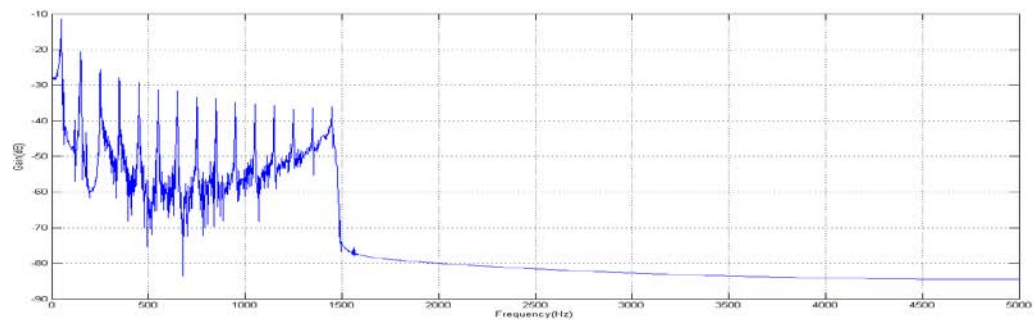
Figure 12 50 Hz square wave spectrum

- For obtaining each of the signals at 1560Hz, 60Hz, 120Hz and 180Hz; band pass filters have been used. Signals 5,6,7 and 8 in the schematic respectively represent these frequencies

Figure 13 Filtered Sine wave spectrums- (Top to bottom) 1560Hz, 60Hz, 120Hz, 180Hz

Figure 14 Schematic of signal decomposition process using digital filters (Signal 3)

Characteristics of the digital filters used are discussed in detail in table 2. Filters described in the schematic are discussed in the table, in the respective order.

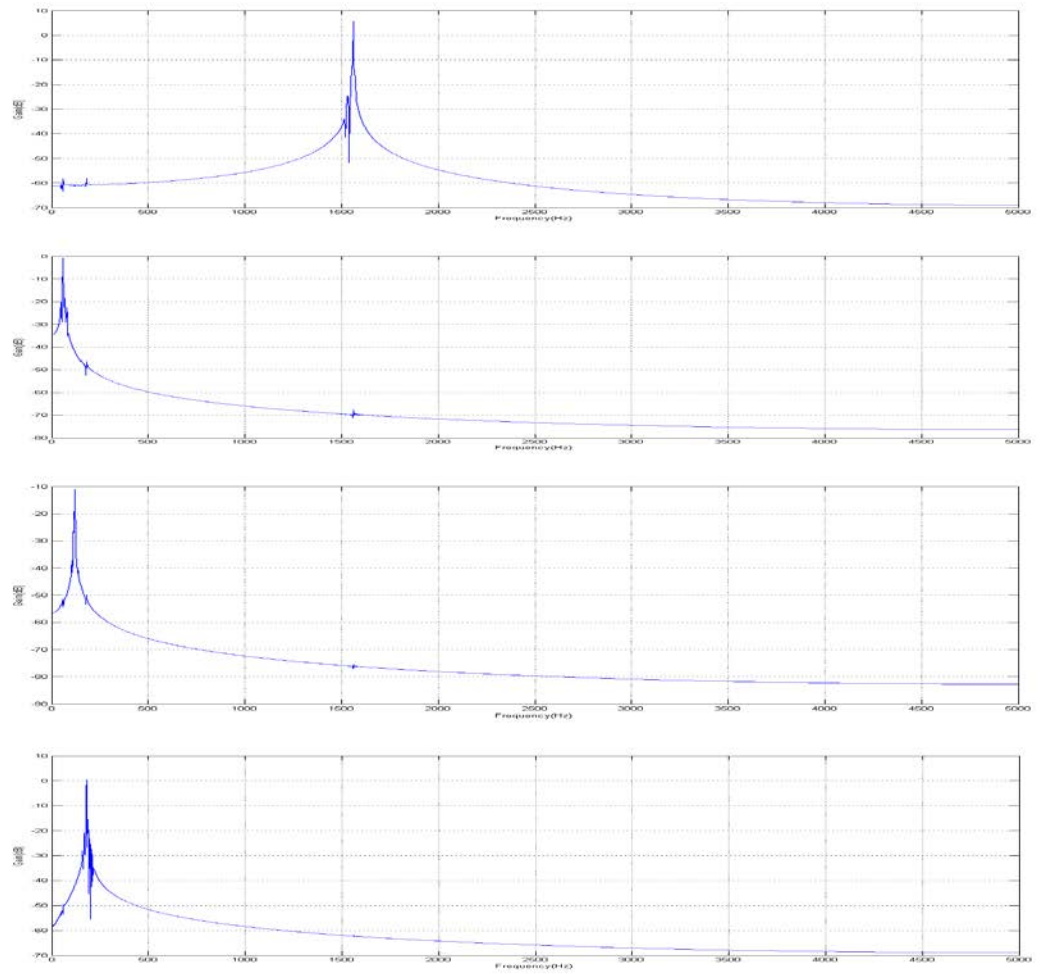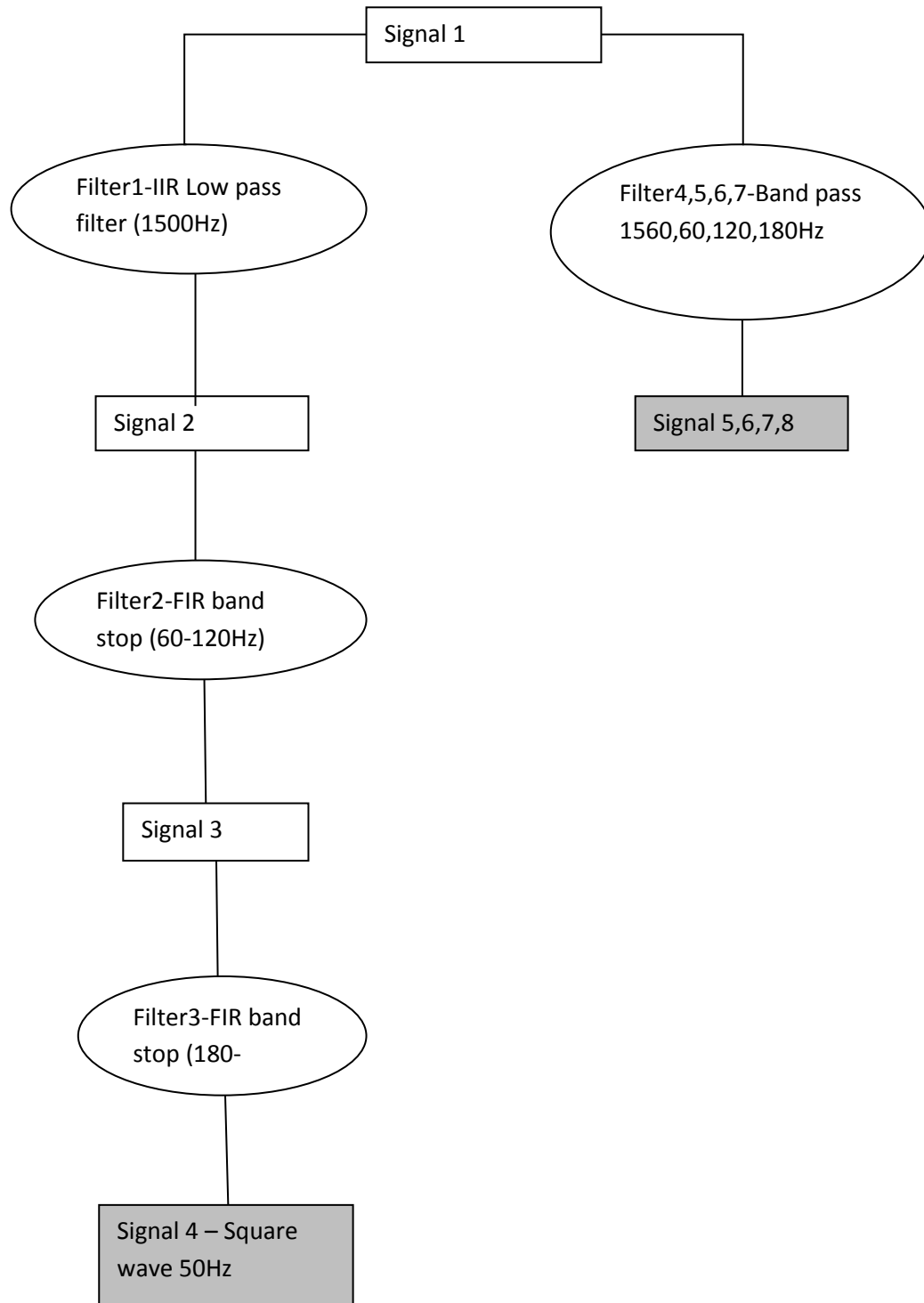**Table 2 Analysis of digital filters used for decomposition of signal 3**

Filter #	Response Type	Design Method	Options used	Filter order	Frequency specifications	Stopband attenuation
1	Lowpass	IIR	Butterworth	258	Fpass = 1450, Fstop=1500	80dB
2	Bandstop	FIR	Least squares	1000	Fpass1=50,Fstop1=55 Fstop2=130,Fpass2=140	60dB
3	Bandstop	FIR	Least squares	1000	Fpass1=150,Fstop1=160 Fstop2=240,Fpass2=250	60dB
4	Bandpass	FIR	Least squares	1000	Fstop1=1510,Fpass1=1530 Fpass2=1560,Fstop2=1580	60dB
5	Bandpass	FIR	Least squares	2000	Fstop1=50,Fpass1=53, Fpass2=70,Fstop2=90	50dB
6	Bandpass	FIR	Least squares	2000	Fstop1=150,Fpass1=160, Fpass2=200,Fstop2=220	60dB
7	Bandpass	FIR	Least squares	2000	Fstop1=100,Fpass1=110, Fpass2=130,Fstop2=140	60dB

Time history of each signal obtained by decomposition of actual signal is presented in figure 15. From the figure, it is evident that maximum energy is carried by signal at 1560Hz which is also revealed by the spectrum in figure 15.
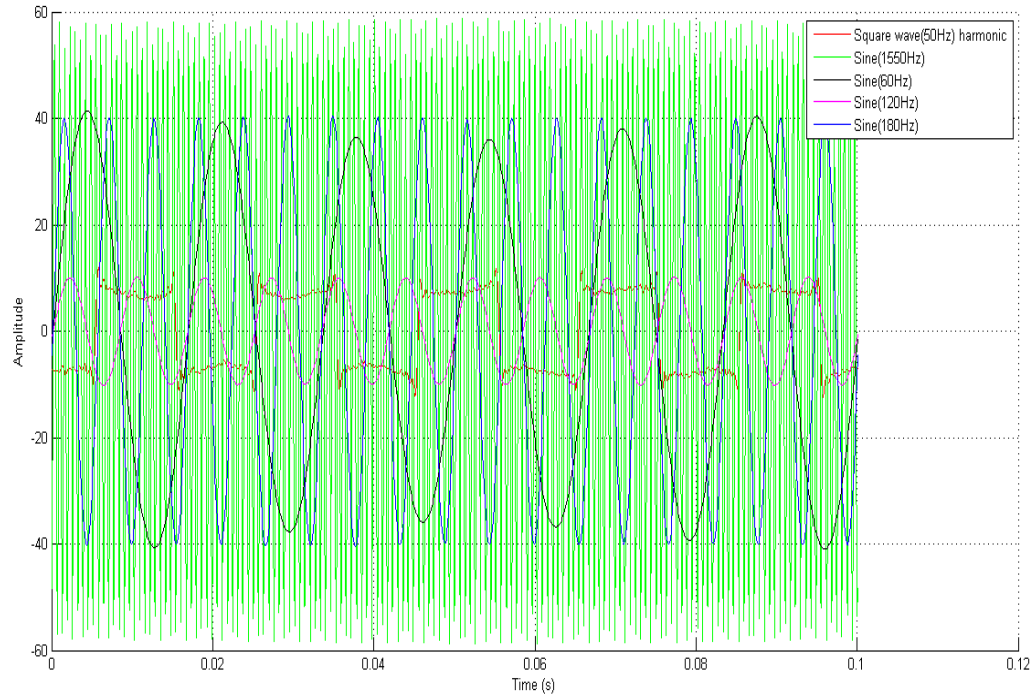

Figure 15 Signal overlay of signals used in signal 3

Signal 4

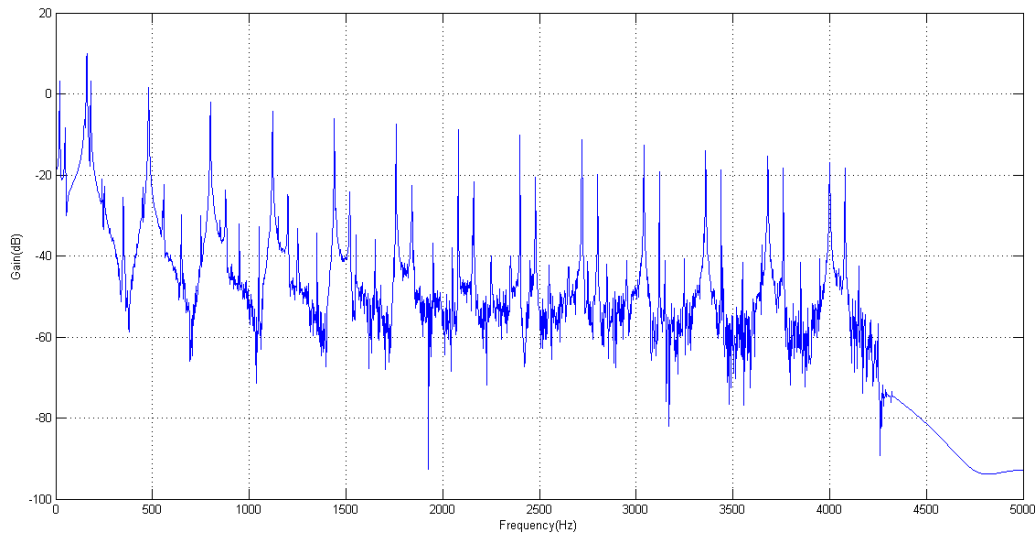Spectral analysis of signal 4 are presented in figure 16

Figure 16 FFT of signal 4 before digitally processing the data

Following observations can be made from the spectrum presented in figure 16

- Profound spikes which represent maximum energy follow an odd harmonic pattern at 160Hz, 480Hz, 800Hz and so on. Therefore they represent a square of frequency 160Hz
- By zooming in closer, an odd harmonic pattern can also be observed at 50Hz,150Hz,250Hz and so on. This pattern forms a square wave of frequency 50Hz
- Other patterns of spikes that are significant are the frequencies at 160Hz, 240Hz, 560Hz etc. These frequencies denote non-integral multiples at 1.5, 3.5, 5.5 times the fundamental frequency. These frequencies are also called the inharmonic frequencies
- There is significant energy in the frequencies at 20Hz and 180Hz. These frequencies do not follow any pattern

The process followed for signal decomposition is presented as a schematic in figures 21 and 22. The schematic is summarized below

- To capture the square wave at 160 Hz, first a low-pass filter(filter1) is used to remove all frequencies above 1500Hz and also 5 harmonics are captured (160*9=1440Hz). Later high pass filter(filter2) is used to remove frequencies until 160Hz. Bandstop filters(filters 3 and 4) are used successively to remove frequencies between the harmonics. Resulting spectrum contains energies only at odd harmonics describing a square wave
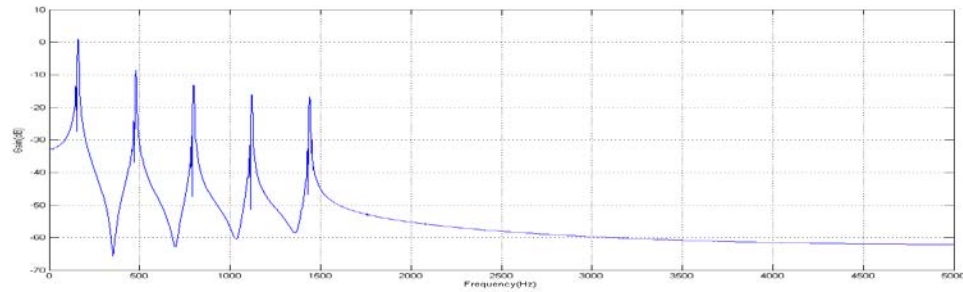
Figure 17 Filtered Square wave 160Hz spectrum

- To capture square wave at 50Hz, low pass filter is used to eliminate all frequencies above 450Hz. Bandstop filters are then applied to eliminate frequencies between the harmonics
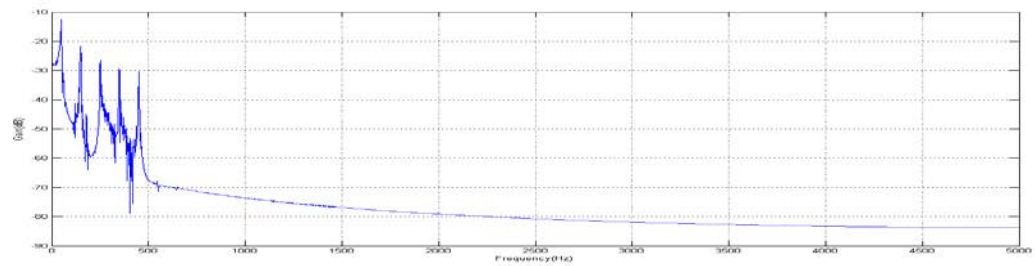
Figure 18 Filtered Square wave 50Hz spectrum

- Inharmonic pattern is captured by applying a low pass filter to remove frequencies above 1200Hz. Then band stop filter is applied to eliminate frequencies between "inharmonics"
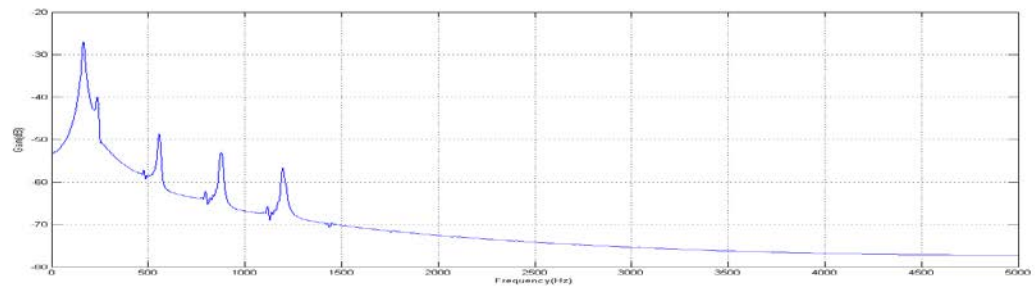
Figure 19 Filtered Inharmonic frequency (160 Hz) spectrum

- Two sine-waves are captured by using a band-pass filter at 20Hz and 180Hz
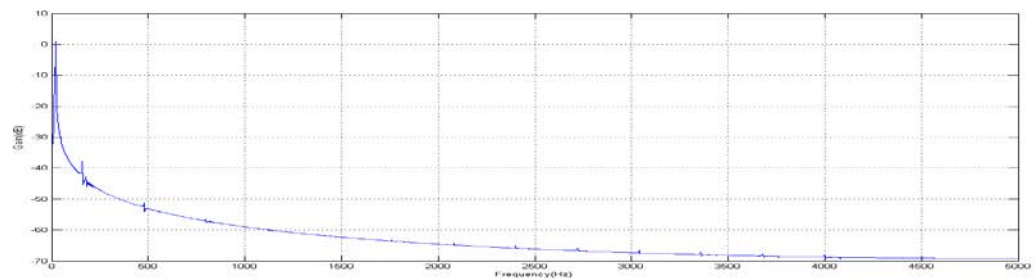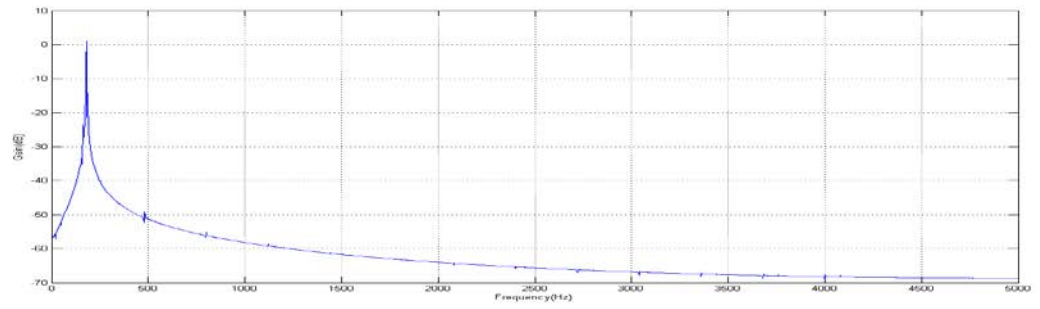
Figure 20 (Top to bottom) Filtered sine wave frequencies at 20Hz and 180Hz

Characteristics of the digital filters used are discussed in detail in table 4. Filters described in the schematic are discussed in the table, in the respective order.

Signal 1

Filter -1 Low pass filter (Fc = 1500Hz)

Signal 2

Filter -2 High pass filter (Fc = 140Hz)

Signal 3

Filter -3 Band stop filter (Fc = 170-460Hz)

Signal 4

Filter 4- Band stop filter (Fc = 490-780Hz)

Signal 5

Filter -5 Band stop filter (Fc = 810-1100Hz)

Signal 6

Filter -6 Band stop filter (Fc = 1130-1420Hz)

Signal 7 Square wave (160Hz)

Filter -7 Low pass filter (Fc = 460 Hz)

Signal 8

**Filter -8 Band stop filter (Fc = 355-445Hz)**

Signal 9

Filter -9 Band stop filter (Fc = 255-345Hz)

Signal 10

Filter -10 Band stop filter (Fc = 155-245Hz)

Signal 11

Filter -11 Band stop filter (Fc = 55-145Hz)

Signal 12

Filter -12 High pass filter (Fc = 40)Hz

Signal 13 Square wave (50Hz)

Arjun Darbha

Figure 21 Schematic of signal decomposition process using digital filters (Signal 4)

Signal 1

Filter -13 Low pass filter (Fc = 1200Hz)

Filter -18 Band pass filter (Fpass = 15-25Hz)

Filter -19 Band pass filter (Fpass = 175-185Hz)

Signal 14

Signal 19 Sinewave 20 Hz

Signal 20 Sinewave 180 Hz

Filter -14 Band stop filter (Fc = 890-1190Hz)

Signal 15

Filter -15 Band stop filter (Fc = 570-870Hz)

Signal 16

Filter -16 Band stop filter (Fc = 250-550Hz)

Signal 17

Filter -17 Band stop filter (Fc = 170-230Hz)

Signal 18 Inharmonic (160Hz)

Arjun Darbha

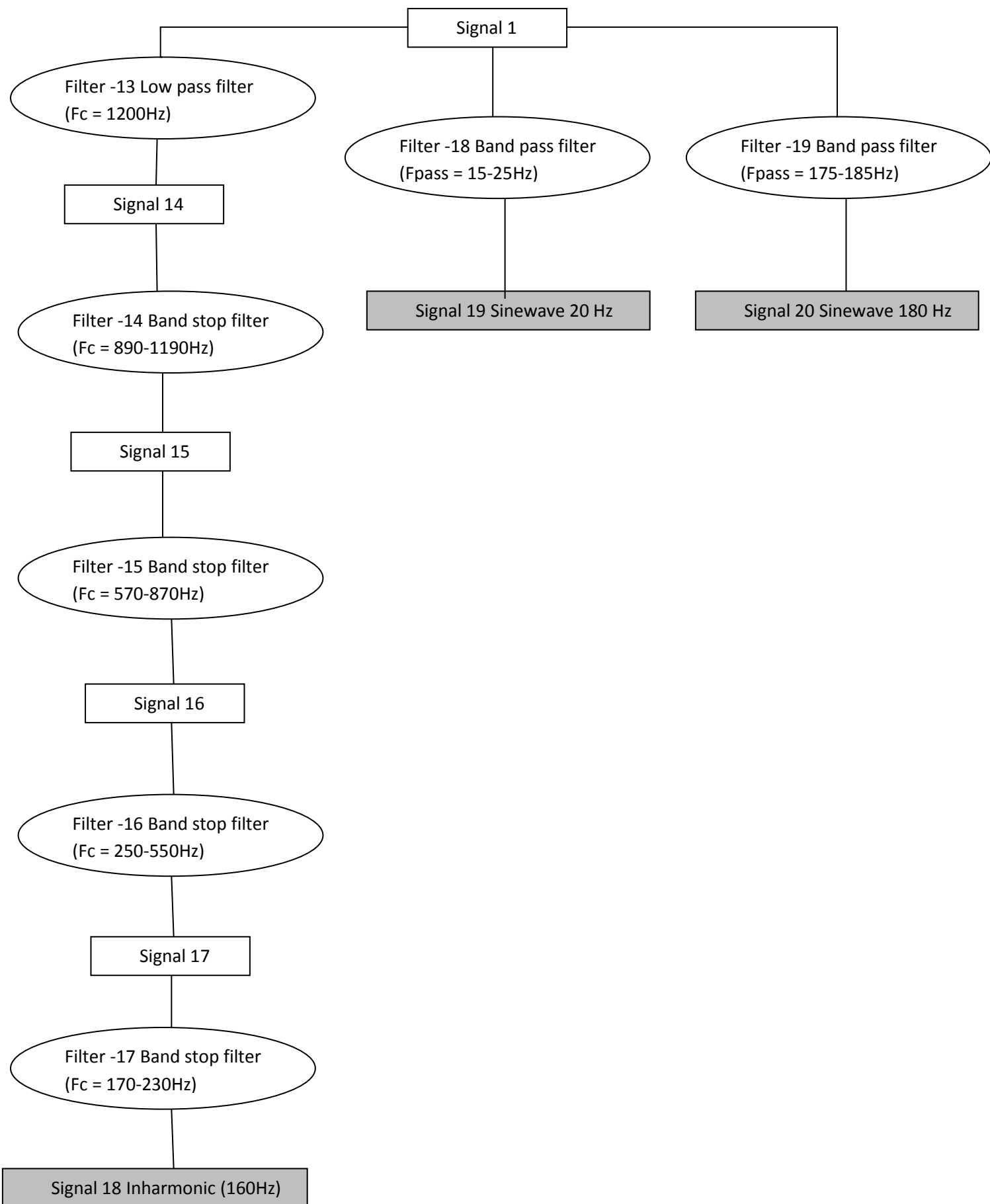**Figure 22 Schematic of signal decomposition process using digital filters (Signal 4) Contd.**

Table 3 Analysis of digital filters used for decomposition of signal 4

| Filter #	Response Type	Design Method	Options used	Filter order	Frequency specifications (Hz)	Stopband attenuation
1	Lowpass	IIR	Butterworth	258	Fpass = 1500, Fstop=1550	80dB
2	Highpass	FIR	Least squares	1000	Fstop=140,Fpass=155	50dB
3	Bandstop	FIR	Least squares	1000	Fpass1=160,Fstop1=170 Fstop2=460,Fpass2=470	40dB
4	Bandstop	FIR	Least squares	1000	Fpass1=480,Fstop1=490 Fstop2=780,Fpass2=790	40dB
5	Bandstop	FIR	Least squares	2000	Fpass1=800,Fstop1=810 Fstop2=1100,Fpass2=1110	40dB
6	Bandstop	FIR	Least squares	2000	Fpass1=1120,Fstop1=1130 Fstop2=1420,Fpass2=1430	40dB
7	Lowpass	FIR	Least squares	2000	Fpass = 460Hz, Fstop=470Hz	60dB
8	Bandstop	FIR	Least squares	2000	Fpass1=350,Fstop1=355 Fstop2=445,Fpass2=450	60dB
9	Bandstop	FIR	Least squares	2000	Fpass1=250,Fstop1=255 Fstop2=345,Fpass2=350	60dB
10	Bandstop	FIR	Least squares	2000	Fpass1=150,Fstop1=155 Fstop2=245,Fpass2=250	60dB
11	Bandstop	FIR	Least squares	2000	Fpass1=50,Fstop1=55 Fstop2=145,Fpass2=150	60dB
12	Highpass	FIR	Least squares	1500	Fstop = 40, Fpass=50	60dB
13	Lowpass	FIR	Least squares	1500	Fpass=1210,Fstop=1220	70dB
14	Bandstop	FIR	Least squares	1500	Fpass1=880,Fstop1=890 Fstop2=1190,Fpass2=1200	55dB
15	Bandstop	FIR	Least squares	1500	Fpass1=560,Fstop1=570 Fstop2=870,Fpass2=880	55dB
16	Bandstop	FIR	Least squares	1500	Fpass1=240,Fstop1=250 Fstop2=550,Fpass2=560	60dB
17	Bandstop	FIR	Least squares	2000	Fpass1=160,Fstop1=170 Fstop2=230,Fpass2=240	60dB
18	Bandpass	FIR	Least squares	2000	Fstop1=10,Fpass1=15, Fpass2=25,Fstop2=30	70dB
19	Bandpass	FIR	Least squares	2000	Fstop1=170,Fpass1=175, Fpass2=185,Fstop2=190	70dB
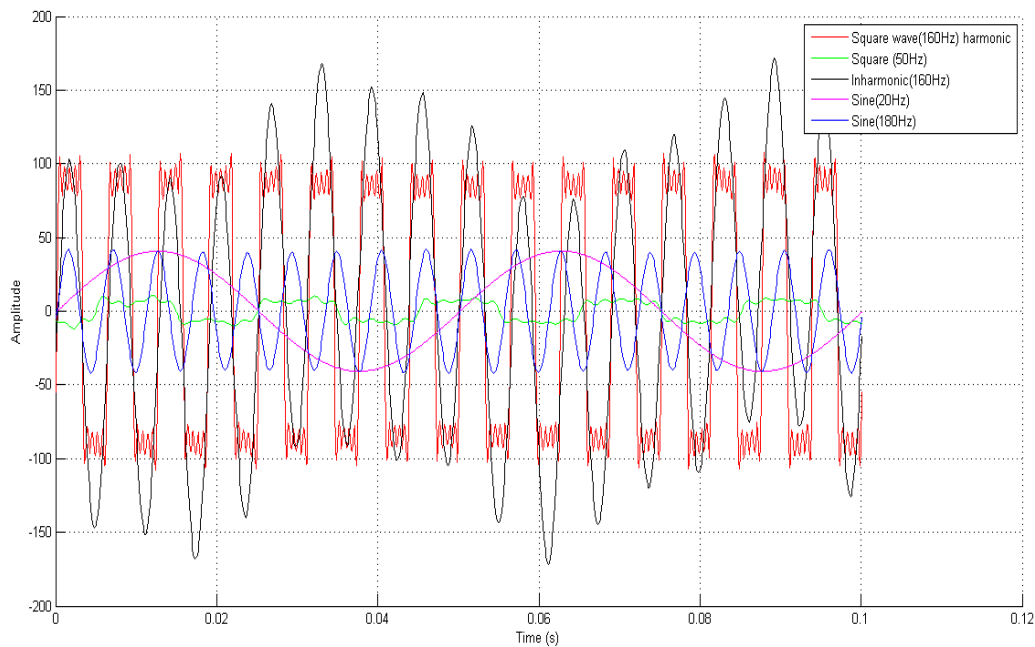
Figure 23 Signal overlay of signals used in signal 4

Decomposition of signal 4 into its components and overlay of these components generates the figure 23. It is evident from this figure that significant energy is carried by the non-harmonic frequency and also by the square-wave of frequency 160Hz. This is justified by the spectrum in figure 16.

5 Conclusions

It can be concluded that digital filters can be used effectively to decompose the signal into its components. To obtain shorter transition bands, FIR filters are most effective when compared to IIR filters. Hence almost all the filters used in this work are FIR digital filters. Different techniques were also used to generate filters. However, technique of least squares has been favored in this work because of its accuracy. However, this technique requires a large number of coefficients and the computational efficiency has been compromised to achieve the desired results. If computational efficiency is desired, it is suggested to use IIR filters only if stability issues are not encountered.

Code used for generating plots of signals imported into workspace

```matlab
close all

%Actual unfiltered signal%
signal = sig1.data(1:1000);
figure
plot((1:1000)*(1/Fs),signal);
xlabel('Time (s)')
ylabel('Amplitude')
grid

%Filtered signals
signal_f_1 = sig7.data(5000:6000);
signal_f_2 = sig10.data(5000:6000);
signal_f_3 = sig18.data(5000:6000);
signal_f_4 = sig19.data(5000:6000);
signal_f_5 = sig20.data(5000:6000);

figure
hold on
plot((1:1001)*(1/Fs),signal_f_1,'r')
plot((1:1001)*(1/Fs),signal_f_2,'g')
plot((1:1001)*(1/Fs),signal_f_3,'k')
plot((1:1001)*(1/Fs),signal_f_4,'m')
plot((1:1001)*(1/Fs),signal_f_5)
hold off
legend('Square wave(160Hz) harmonic','Square
(50Hz)','Inharmonic(160Hz)','Sine(20Hz)','Sine(180Hz)');
xlabel('Time (s)')
ylabel('Amplitude')
grid

%Spectra
plot_spect(spect1)
plot_spect(spect2)
plot_spect(spect3)
```

Plot_spect function

```matlab
function [ a ] = plot_spect( spect )
start_f = find(spect.f==0);
stop_f = find(spect.f==5000);
figure
plot(spect.f(start_f:stop_f),10*log10(spect.P(start_f:stop_f)))
xlabel('Frequency(Hz)')
ylabel('Gain(dB)')
grid
a=1;
end
```

END OF DOCUMENT