NOISE CANCELLATION USING WINDOWINGTECHNIQUE

by Lekhya Tata, Karthik Govinda Raju

Motivation:

*Many methods have been implemented for noise cancellation in audio signals. Mostly in software like MATLAB. Filters that are implemented in Verilog are confined to only small class of noise removal and no study proves that each method is completely efficient.*

Tasks:

*Project aim is to filter audio signal with noise in MATLAB and get the results. By taking MATLAB results as reference filter is designed in Verilog for the particular audio file with the constrains that are used in MATLAB code. Compare both the results. To get faster results and reduce hardware mathematical computations in filter will also be improved.*

Significance:

*There is no formal study that shows which transforms method works better, that are been currently employed. Our proposal is to design a FIR filter that includes FFT, IFFT and windowing techniques for noise cancellation.*

Table of contents

[1. Introduction 1](#_Toc7555909)

[2. Noise 3](#_Toc7555910)

[2.1 Hum 3](#_Toc7555911)

[2.2 Hiss 4](#_Toc7555912)

[2.3 Rumble 4](#_Toc7555913)

[2.4 Crackle 5](#_Toc7555914)

[2.5 White noise 6](#_Toc7555915)

[3. Filters 7](#_Toc7555916)

[3.1 FIR filter 7](#_Toc7555917)

[3.2 IIR filter 8](#_Toc7555918)

[4. Windowing 10](#_Toc7555919)

[4.1 Hann window 12](#_Toc7555920)

[5. MATLAB 14](#_Toc7555921)

[5.1 Generating plots 14](#_Toc7555922)

[5.2 Initialization 15](#_Toc7555923)

[6. System verilog 23](#_Toc7555924)

[6.1 Test bench 24](#_Toc7555925)

[6.2 Windowing 26](#_Toc7555926)

[6.3 FFT 27](#_Toc7555927)

[6.1 FFT 29](#_Toc7555928)

[References 36](#_Toc7555929)

[Figure 1. Importance of ASIC projects 1](#_Toc7600676)

[Figure 2. Role of ASIC adoption in various industries 1](#_Toc7600677)

[Figure 3. Flow of the project 2](#_Toc7600678)

[Figure 4. Low frequency hum noise 3](#_Toc7600679)

[Figure 5. Hiss noise plot 4](#_Toc7600680)

[Figure 6. Rumble noise 5](#_Toc7600681)

[Figure 7. Gramophone crackling noise 6](#_Toc7600682)

[Figure 8. Plot showing white noise in an audio file 6](#_Toc7600683)

[Figure 9. A basic depiction of four major filter types 7](#_Toc7600684)

[Figure 10. Butterworth filter representation 8](#_Toc7600685)

[Figure 11. FIR filter formula 8](#_Toc7600686)

[Figure 12. Signal with non-integer number of periods 10](#_Toc7600687)

[Figure 13. Spectral leakage to the FFT of signal in the figure 11](#_Toc7600688)

[Figure 14. Result of windowing 11](#_Toc7600689)

[Figure 15 . Applying a window minimizes spectral leakage. 12](#_Toc7600690)

[Figure 16. Multiplying a original signal by hanning window 13](#_Toc7600691)

[Figure 17. Code for generating plots 14](#_Toc7600692)

[Figure 18. Code for reading input and setting FFT points 15](#_Toc7600693)

[Figure 19. For loop constraints 15](#_Toc7600694)

[Figure 20. Filter design 16](#_Toc7600695)

[Figure 21. For loop 16](#_Toc7600696)

[Figure 22. Hann window 16](#_Toc7600697)

[Figure 23. Applying filter 17](#_Toc7600698)

[Figure 24. Inverse fourier transform 17](#_Toc7600699)

[Figure 25. Selection of multiplication factor for hanning window 18](#_Toc7600700)

[Figure 26. Producing final filtered output 19](#_Toc7600701)

[Figure 27. Plot of Input signal in time domain 20](#_Toc7600702)

[Figure 28. Plot of input signal in frequency domain 21](#_Toc7600703)

[Figure 29. Plot of output signal in time domain 21](#_Toc7600704)

[Figure 30. Plot of output signal in frequency domain 22](#_Toc7600705)

[Figure 31. Design flow in system Verilog 23](#_Toc7600706)

[Figure 32. Fetching input 24](#_Toc7600707)

[Figure 33. fetching filter data 25](#_Toc7600708)

[Figure 34. Hann window formula 26](#_Toc7600709)

[Figure 35. FFT formula 27](#_Toc7600710)

[Figure 36. Saving 512 Inputs to Buffer 28](#_Toc7600711)

[Figure 37. Calculating address for input Buffer (100 -> 001) 29](#_Toc7600712)

[Figure 38. Writing Input Data to input Buffer 29](#_Toc7600713)

[Figure 39. Transferring Data to Working Buffer 30](#_Toc7600714)

[Figure 40. Writing data from working buffer to butterfly inputs 30](#_Toc7600715)

[Figure 41. Writing output of Butterflies to Working Buffer 31](#_Toc7600716)

[Figure 42. Writing output of Butterflies to Output Buffer (when last stage) 32](#_Toc7600717)

[Figure 43. Butterfly units running in Parallel 33](#_Toc7600718)

[Figure 44. States Providing INput to Butterflies 35](#_Toc7600719)

[Figure 45. Fetching output From Output Buffer 35](#_Toc7600720)

## Introduction

ASIC chips are leading in the industry of semiconductor technology because they are working at a quicker rate and as a result for better operation and task involving great technology, design is becoming more complex. It includes many challenges and trade-offs.

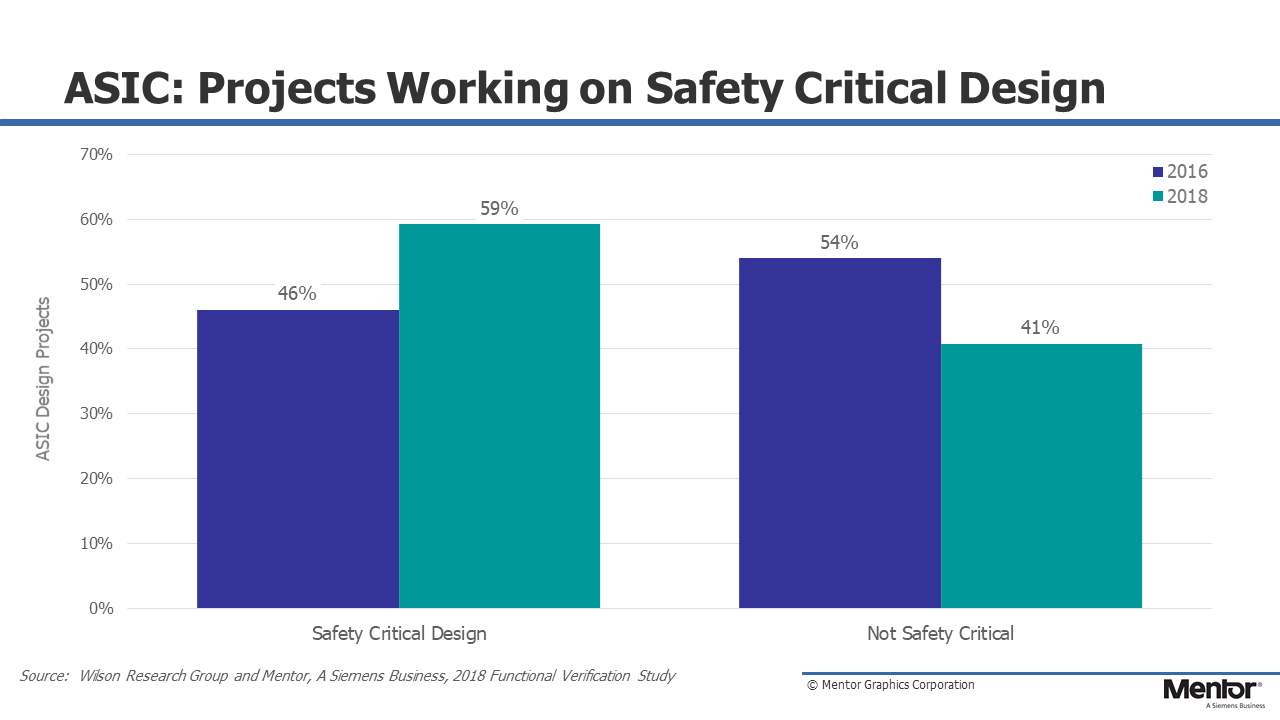


Figure . Importance of ASIC projects

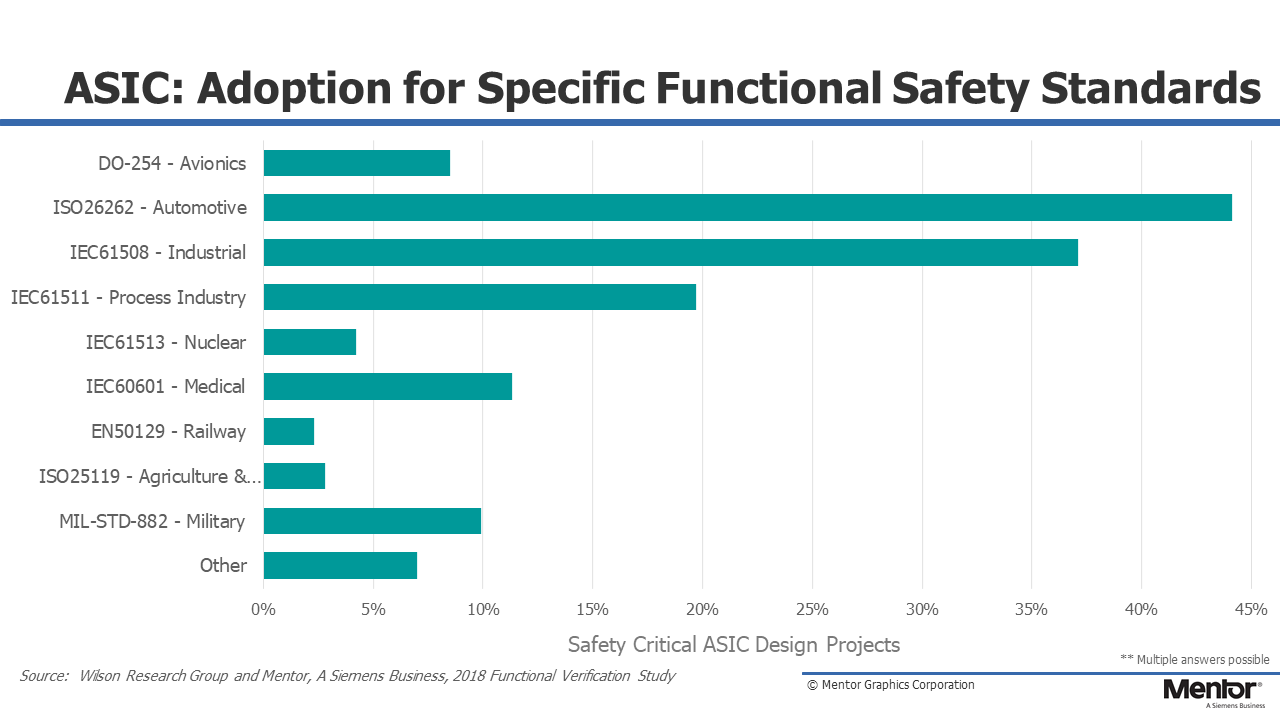


Figure . Role of ASIC adoption in various industries

From the above graph[1] it is clear that ASIC is leading in various industries. Medical is the fourth leading industry. This project involves noise removal in the ultrasound of fetal heartbeat.

Aim of the project is to design a filter that performs better and reduce maximum noise in a faster way. Generally, noise can be eliminated in time domain and frequency domain. In this project design is made for frequency domain as speech and noise signals may be better separated in that space, which enables better filter estimation and noise reduction performance. Noise can be categorized in many ways. Additive noise (white noise) is selected for filtering.

When the audio signal with noise is been fed to MATLAB and cut off frequency of noise signal is known from magnitude spectrum plot. This cutoff frequency is used to calculate digital cut off using sampling frequency and analog cut off frequency

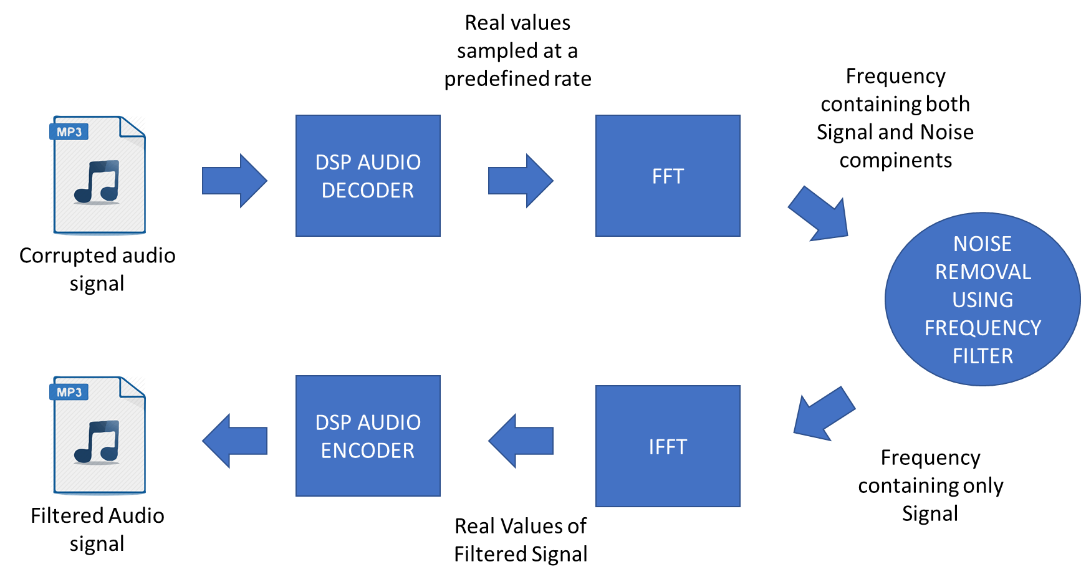


Figure . Flow of the project

## Noise

Noise can be seen in all shapes, sizes and sounds. This is project concentrates on noise in the form of sound. Sound wave can be broken down to two fundamental characteristics, frequency and amplitude Any unwanted sound that is added to the original audio file is considered as a sound noise. During audio recording, noise gets included in Analog tapes or digital recordings. Among many files section of “hiss” noise found throughout the recording. Addition of noise into the recording also depends on the environment of recording and equipment. Below is the list of four major kinds of audio noises.

### 2.1 Hum

The sound which is at low frequency than the actual sound and whose frequency ranges between 40 – 80hz [2] is considered as hum. For example, whirring of low pitched motor. Hum usually occurs by electrical interferences or the ground wiring of recording equipment is not done properly.



Figure . Low frequency hum noise

### 2.2 Hiss

This can be seen in devices that includes electronic components. Due to rise in temperature (compared to room temperature) path of electrons in the device is deviated as a results output is disturbed. This disturbance in the distorted output is considered as “hiss”. Hiss noise depends on the quality of recording equipment. This noise can also be a result of environmental factors like A/C, fans etc.,

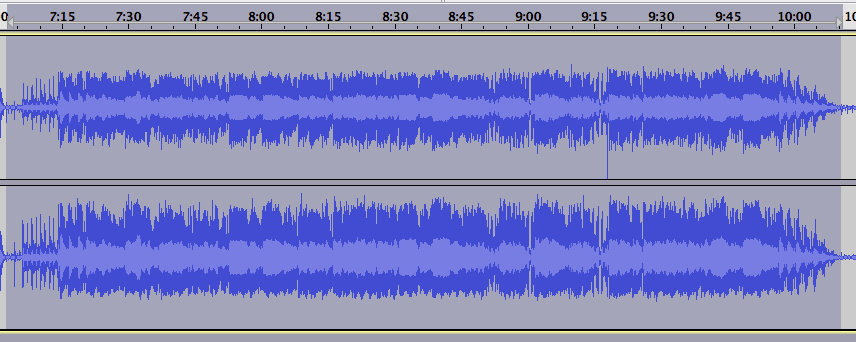


Figure . Hiss noise plot

### 2.3 Rumble

This is very common audio noise. Like hum noise this is also at low frequency. This can be seen only between specific points in a given bandwidth. It is most observable noise in mechanical devices. For example, ball bearings at a joint, gear rotation in a gear box, mechanical parts moving in the audio recording equipment creates this noise. There are filters that are inbuilt in a device to eliminate this kind of noise.



Figure . Rumble noise

### 2.4 Crackle

These are kind of discontinuous and non-musical. This kind of noise is similar to that sound heard during burning of wood. This is caused by explosion of air pockets inside the object. These are of two types fine crackles which are high pitched and last for less duration of time, coarse crackles which are of low pitched and last for longer duration.

Noise can be categorized by giving color names. This project involves an audio file that contains “white noise”. White noise generally comes to hiss noise category. It is present everywhere in the entire audio file. To put entire concept in a nutshell, it is a layer of sound that is considered as noise floor while processing it.



Figure . Gramophone crackling noise

### 2.5 White noise

White noise can be stated as the noise that contains many frequencies. sometimes white noise is used to mask the other unwanted frequencies.

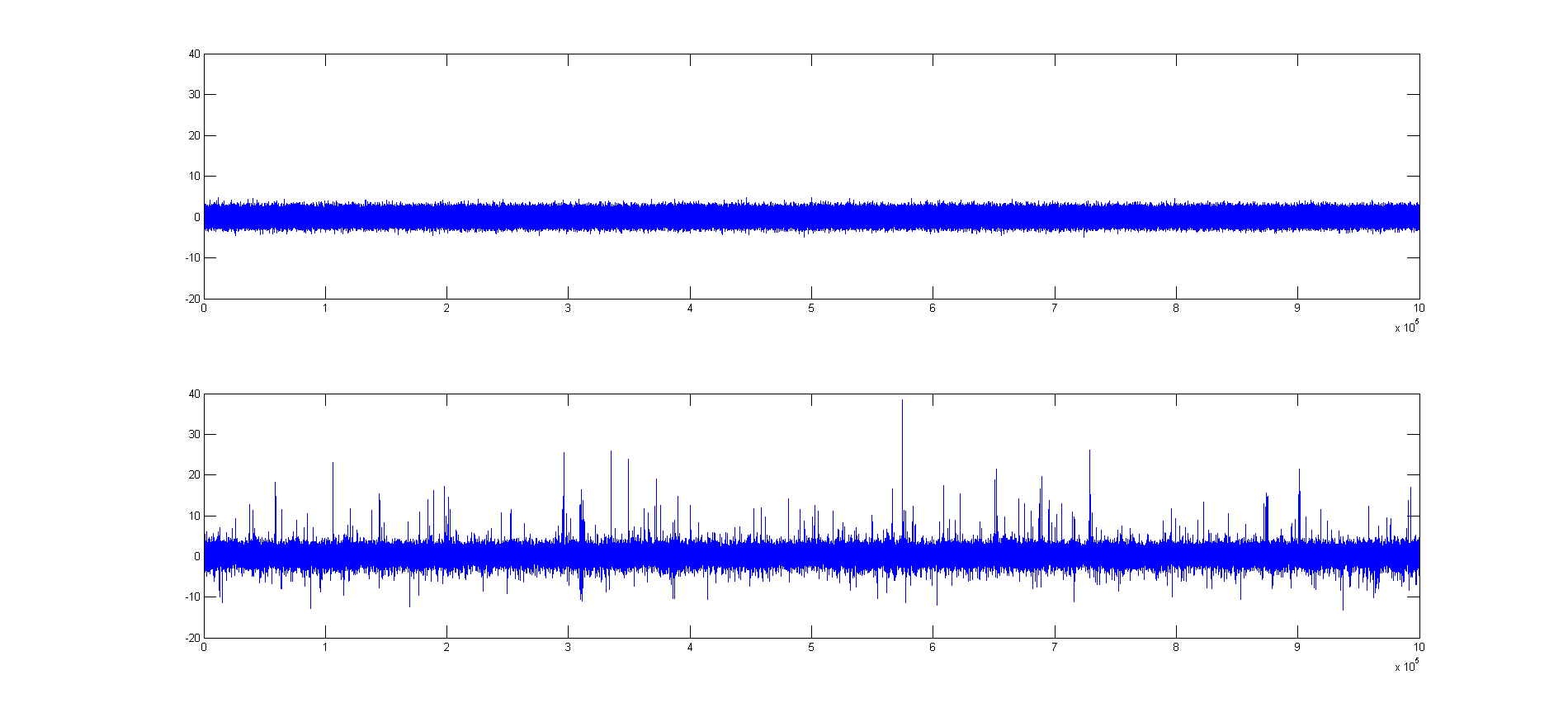


Figure . Plot showing white noise in an audio file

## 3. Filters

The main function of a filter is to pass desired frequencies and attenuate the unwanted frequencies. The four primary types of filters that can be listed down as

* Low-pass filter
* High-pass filter
* Band-pass filter
* Notch filter (band-stop filter)

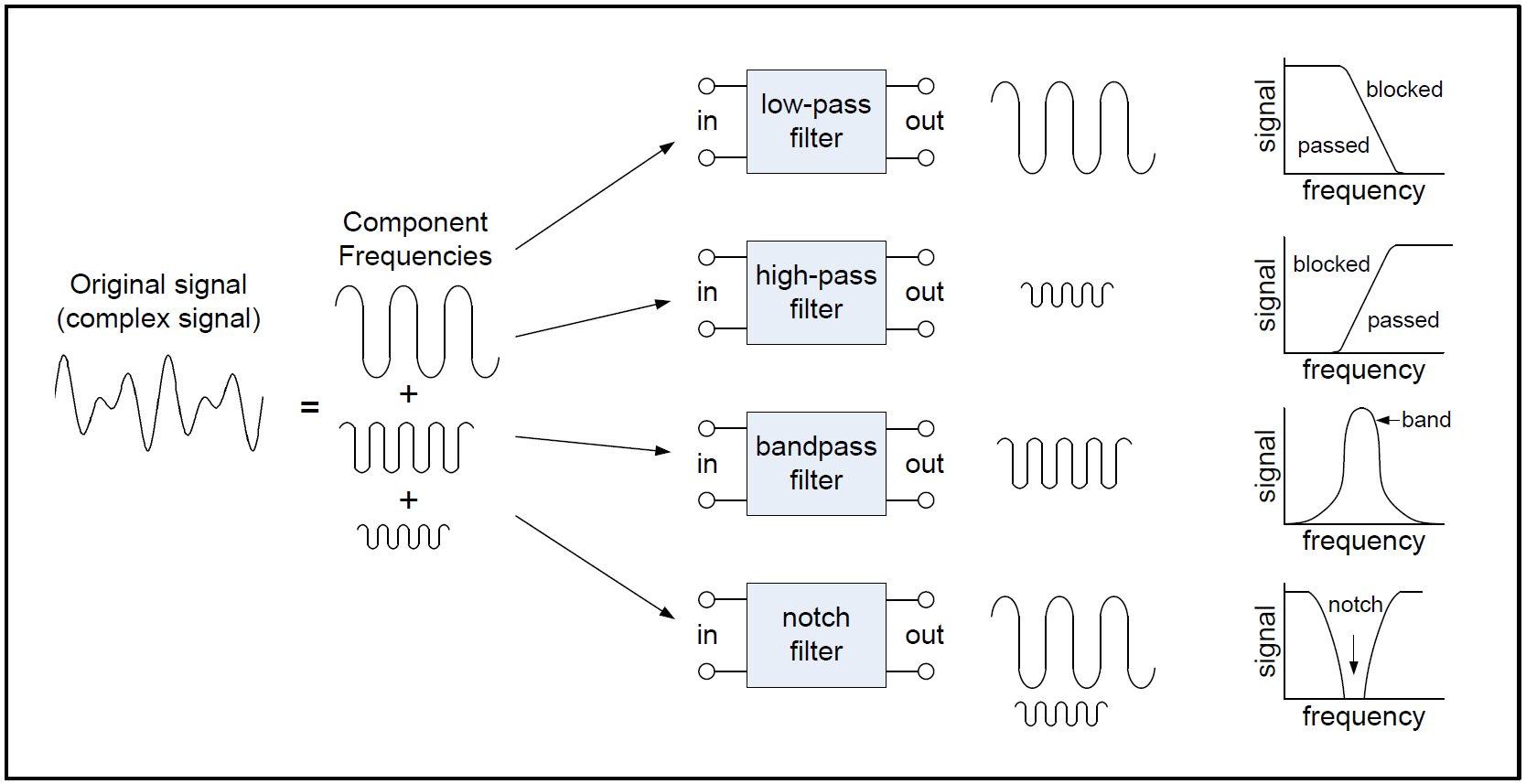


Figure . A basic depiction of four major filter types

There are two major digital filters. FIR filters and IIR filters.

### 3.1 FIR filter

The Fourier filter is a type of filtering function that is based on manipulation of specific frequency components of a signal. It works by taking the Fourier transform of the signal, then attenuating or amplifying specific frequencies, and finally inverse transforming the result. The main advantage of FIR filter is that they can easily be designed to be “linear phase”. FIR filter works better with 2N sample points. Stability of FIR filter is very good which can also be framed as finite output can be obtained for every finite input.

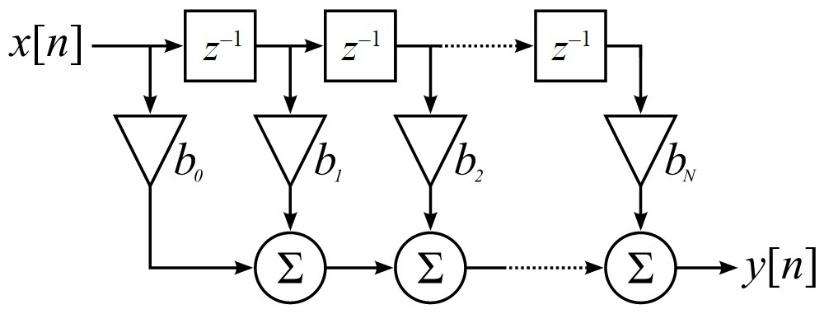


Figure . Butterworth filter representation

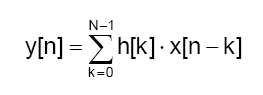


Figure . FIR filter formula

### 3.2 IIR filter

Infinite impulse response filter. This filter is infinite because of presence of feedback in the filter. Desired filtering characteristics can be achieved using IIR filter consuming low memory and calculations when compared to an FIR filter. The main disadvantage is that implementation of filtering using fixed point arithmetic becomes slower and harder. They don’t offer the computational advantages of FIR filters for multi rate applications.

FIR filters are more preferable over IIR filters as the response is a linear phase and non-recursive. IIR filter performance is good when dealing with analog filter responses. This project uses FIR filter for the particular input wave file selected.

## 4. Windowing

In most of the signal measurements, the signal doesn’t contain integer number of periods. Due to this finiteness is not perfect and result in loss of some part of waveform and characteristics of whole signal changes. This finiteness introduces sharp transition changes in the signal that is measured.

When the waveform’s period is non-integer, discontinuities are seen at the endpoints. These unwanted discontinuous components become high frequency harmonics in FFT which are not actually present in the original signal. The frequency of these unwanted components is higher than the Nyquist frequency value. Because of thse distortion the FFT spectrum is not the actual spectrum of actual signal. It appears as if there is a leakage of energy from one frequency component to other frequency component.

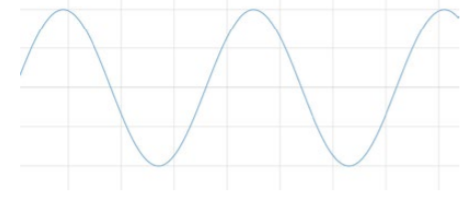


Figure . Signal with non-integer number of periods

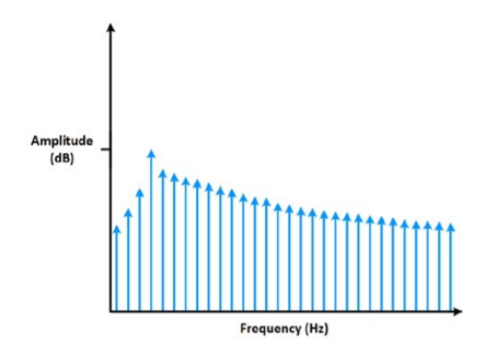


Figure . Spectral leakage to the FFT of signal in the figure

In a signal that contains non-integer periods, this distortion in FFT can be minimized by using a technique called “windowing”. Windowing basically reduce the amplitude of these discontinuities at starting and ending and gradually rolls of to zero of given boundary for a particular finite length.

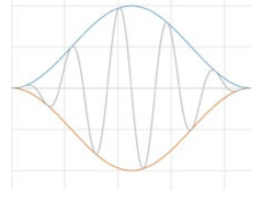


Figure . Result of windowing

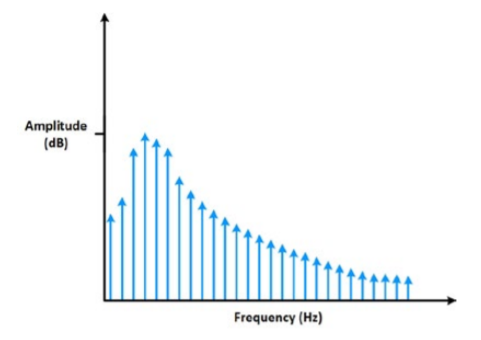


Figure . Applying a window minimizes spectral leakage.

Mostly used windows are listed below.

### 4.1 Hann window

This window is used for a signal whose characteristics are unknown. Instead of doing trade off between amplitude and frequency accuracy a good compromise is provided between the both by using this windowing technique.

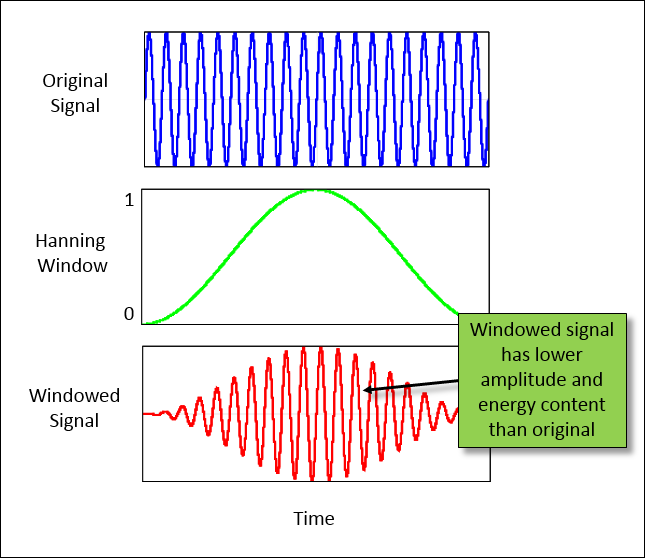


Figure . Multiplying a original signal by hanning window reduces the amplitude and

# energy in the signal[3]

## 5. MATLAB

This software is excessively used for computing. Integration of computation, visualization and programming is possible in this platform.

Entire filter design is first made in Matlab and checked for output. Based on the results obtained cutoff frequency is determined and it is used in filter designing in Verilog.

Flow of design in the MATLAB goes as listed below.

### 5.1 Generating plots

Code for generating all required plots is made in a separate folder and it is called in the file that contains actual code.

Code



Figure . Code for generating plots

To maintain good clarity in the actual code, function “generategraph” is called into actual code. This gives plots in both time domain and frequency domain. Input signal is taken in the variable “signal” and it is plotted in time domain. After processing in the final stage frequency plot of filtered signal is obtained.

### 5.2 Initialization

In the actual code file, “audioread” function is used to fetch the input signal which is ultrasound heartbeat of a fetus. Sampling frequency is given as 22050 hz. BS (512) is the block size for which the filtering is performed. This is nothing but 512 point FFT which is further designed in Verilog. The length of input file is 13782600.

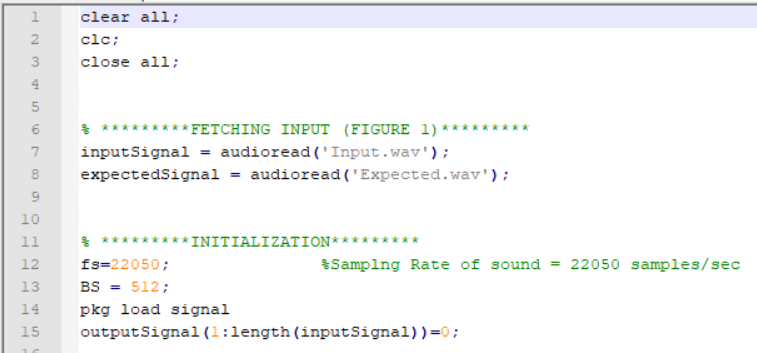


Figure . Code for reading input and setting FFT points

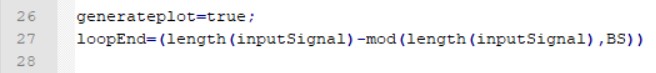


Figure . For loop constraints

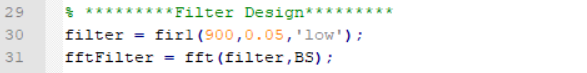


Figure . Filter design

FIR filter of 900 order and 0.05 roll off which is a low pass filter is used. So the entire signal length is divided into chunks whose length is of 512.

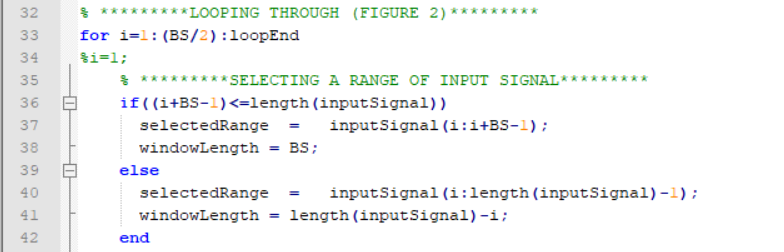


Figure . For loop

A for loop is made which includes which gives the portion of input signal which starts from 1 and end at 512. The next block starts from 512.

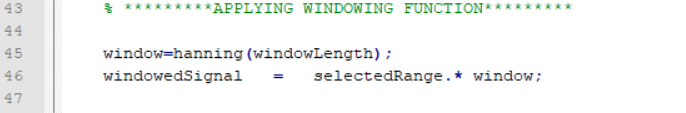


Figure . Hann window

Hanning window is applied to window length which is of block size (intial size of window length is block size). Windowed signal is magnitude of vector product of selected range of input signal and the window. To perform this operation array and order size of both the elements should match.

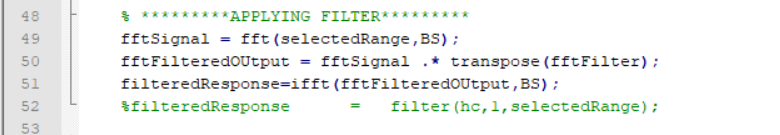


Figure . Applying filter

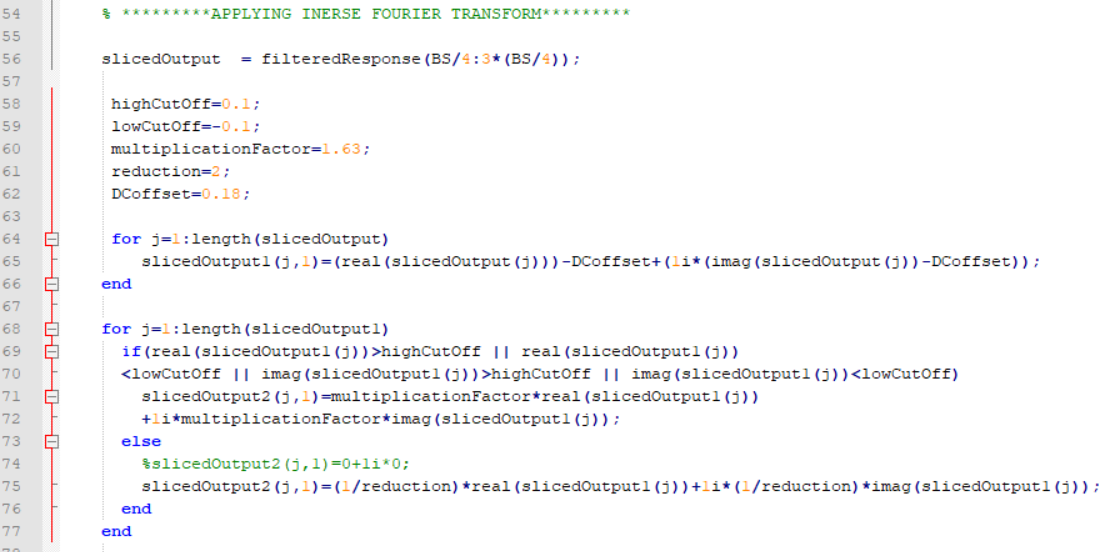


Figure . Inverse fourier transform

FFT is performed for the windowed signal and. Because of windowing DC offset is introduced and amplitude of original signal is reduced. So multiplication factor is selected as 1.63 [3] to bring back the amplitude of filtered signal to the original.

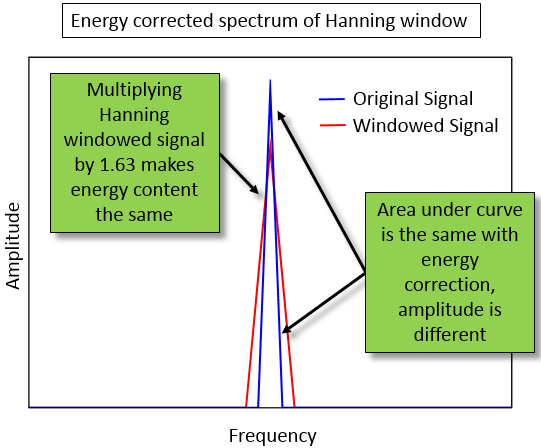


Figure . Selection of multiplication factor for hanning window

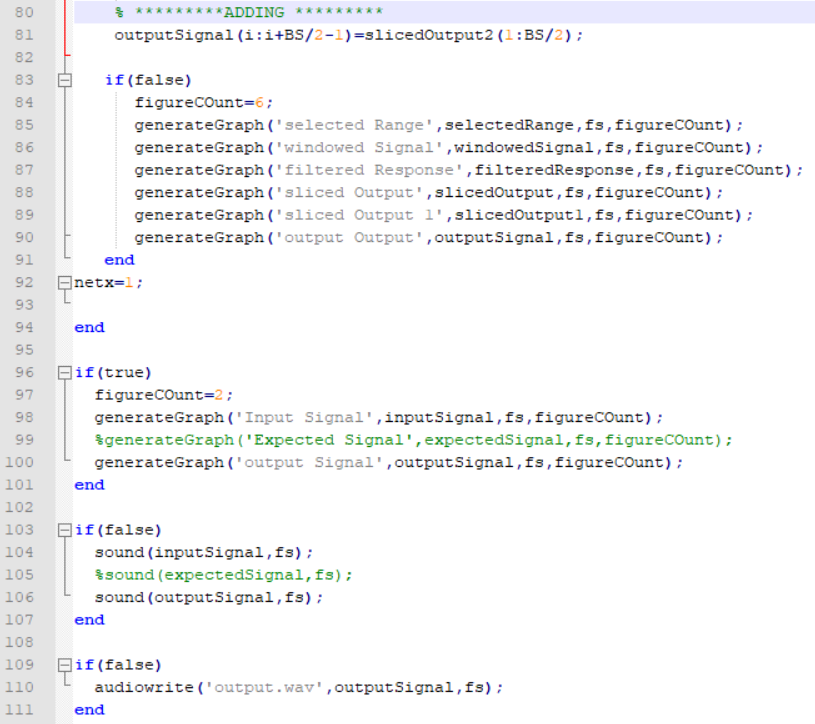


Figure . Producing final filtered output

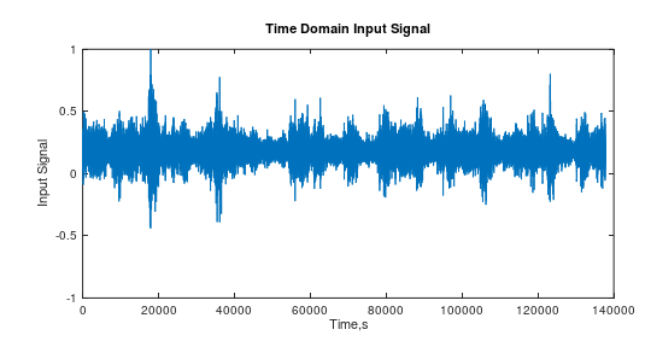


Figure . Plot of Input signal in time domain

Input signal plot in time domain. This contains lot of noise whose frequency can’t be determined from time domain plot. The spikes indicates heartbeat. Main aim is to remove all other frequency components which are all considered as noise components.

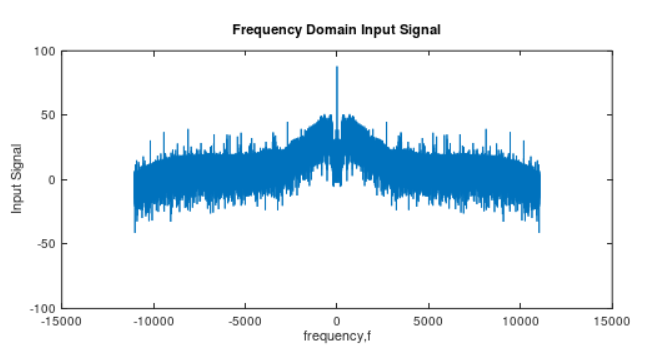


Figure . Plot of input signal in frequency domain

In figure 27 the frequency component at zero frequency determines the doppler heartbeat frequency. The maximum noise floor ranges from -30db to 50db. To eliminate those frequencies low pass FIR filter of order 900 and roll off of 0.05 is selected.

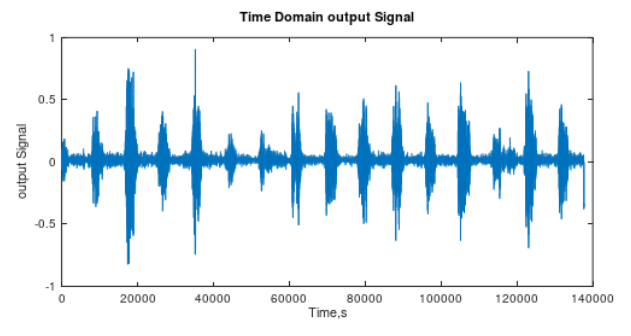


Figure . Plot of output signal in time domain

Before plotting the plot in figure DC offset of 0.18db is observed and amplitude was reduced to nearly 2. So appropriate code is made to overcome that error and the desired output is obtained.

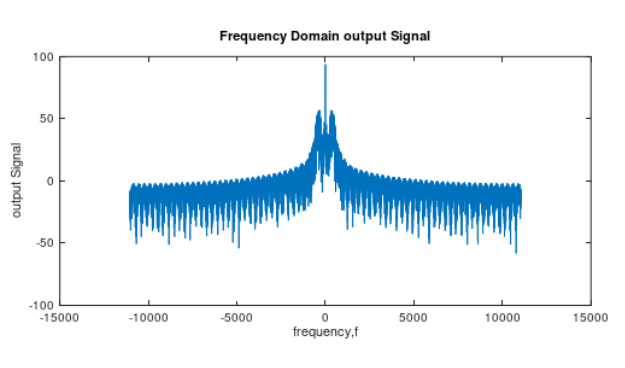


Figure . Plot of output signal in frequency domain

The noise floor in the above figure is pushed down to 0db. The output of this is heard and the noise is eliminated.

## 6. System verilog

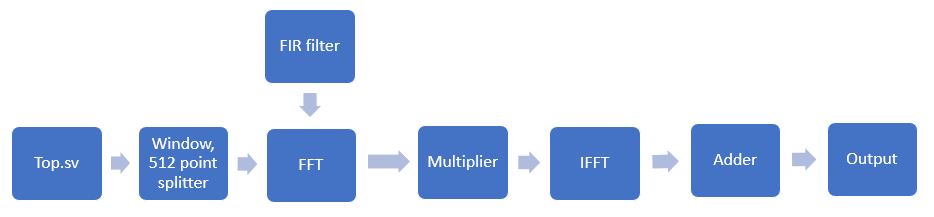
****

Figure . Design flow in system Verilog

After getting filtered results of input signal from Matlab for ultrasound of heartbeat of fetus, the input.wav file is converted into text file, hann window is converted to text file and fir filter co ordinates is converted into text file. All these files are used to perform filtering of input and resultant output values matches with the output from MATLAB.

### 6.1 Test bench



Figure . Fetching input

Input “fetal doppler” is in .wav format. to convert this into digital form which is a vector representation, MATLAB is used. This is of size 1307800. Audio file in .wav format is easy to convert to digital form. So any audio file which is any other format is first converted into .wav file for representing it in digital format.

After converting input into text file that contains real and imaginary values, at every posedge of the clock real values are read in a variable calle “realdata” and imaginary values are read in a variable “imagindata”.



Figure . fetching filter data

Like how the real and imaginary parts of input data is read, in the same manner low pass FIR filter vector representation text file is loaded into design using similar code in system Verilog. Here FIR filter is of order 900, roll 0ff 0.05 and it’s a low pass filter. These numbers are selected by trial and error method. The white noise floor is way below the frequency level of required heartbeat. So low pass Filter is selected for filtering.

### 6.2 Windowing

Windowing technique is used to eliminate the spectral leakage in the signals. Usually the signals which are not periodic contains frequency harmonics in FFT which are not actually present in the signal. These are of high frequency components. the signal frequencies seem like as if the energy been transferred from one harmonic to other harmonic. To eliminate this spectral leakage windowing is applied to the signal which rolls down the signal at starting and at ending to zero, this is been performed because spectral leakage is usually observed at the starting and ending of the recorded non periodic signal. Hann window is used. The hann window’s vector representation is obtained using the MATLAB.

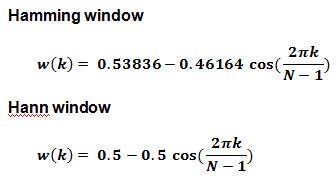


Figure . Hann window formula

### 6.3 FFT

Fast Fourier transform concept is used to convert time domain input signal into frequency domain. If the input to FFT are two to the race N, then it works better. Input signal is a continuous signal in time vs amplitude. In frequency domain signal is plotted between normalized frequency and magnitude response (dB) of signal. The harmonic that is at higher magnitude is determined as heartbeat of fetus.

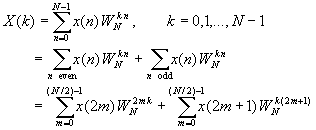


Figure . FFT formula

The formula in above figure is used to compute FFT of filter co-ordinates and hann window co-ordinates. The weight components that are used in computation are called twiddle factors or butterflies. These twiddles contain both real and imaginary values. Real values and imaginary values are of floating-point format and they are saved in arrays in system Verilog.

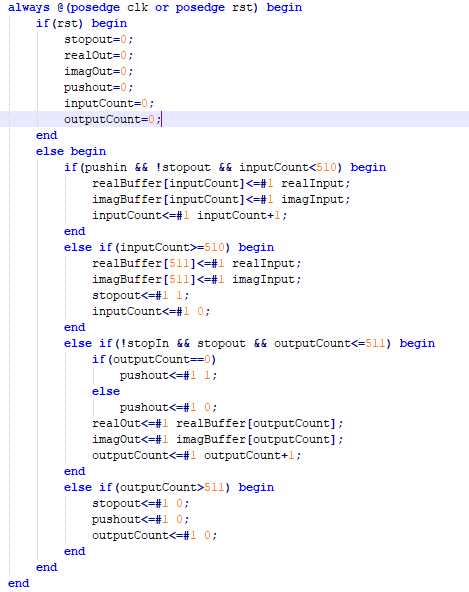


Figure . Saving 512 Inputs to Buffer

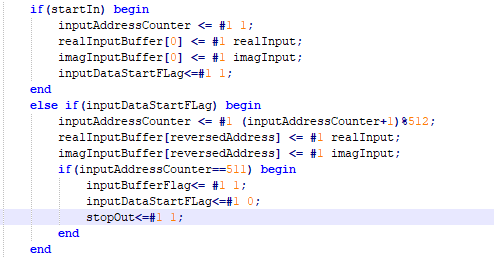
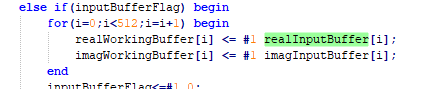


Figure . Writing Input Data to input Buffer



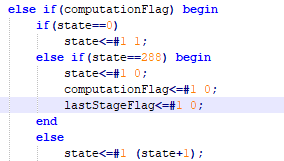


Figure . Transferring Data to Working Buffer

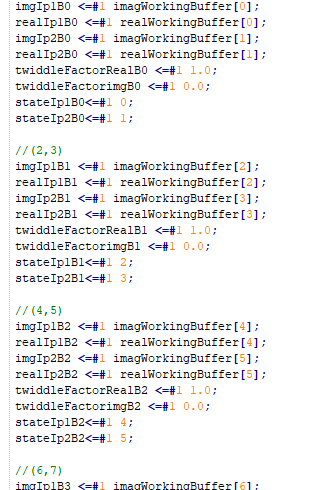


Figure . Writing data from working buffer to butterfly inputs

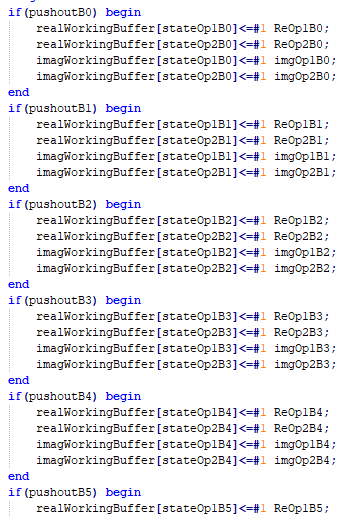


Figure . Writing output of Butterflies to Working Buffer

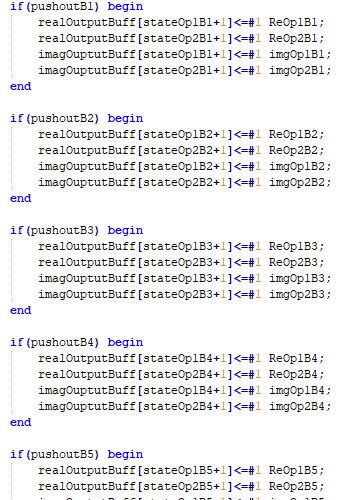
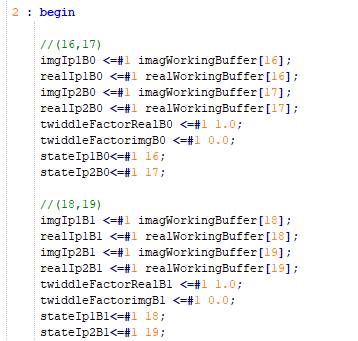


Figure . Writing output of Butterflies to Output Buffer (when last stage)



Figure . Butterfly units running in Parallel





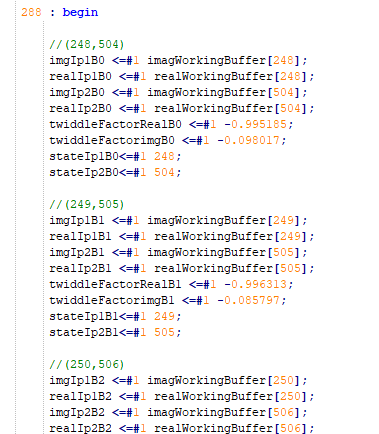


Figure . States Providing INput to Butterflies



Figure . Fetching output From Output Buffer

### 6.1 FFT

## References

[1]. [https://blogs.mentor.com/verificationhorizons/blog/2019/01/22/part-7-the-2018-wilson- research-group-functional-verification-study/](https://blogs.mentor.com/verificationhorizons/blog/2019/01/22/part-7-the-2018-wilson-%20%20research-group-functional-verification-study/)

[2]. <https://www.audioshapers.com/blog/audio-noise.html>

[3]. <https://community.plm.automation.siemens.com/t5/Testing-Knowledge-Base/Window-Correction-Factors/ta-p/431775>

[1] <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6701859>.

[2] <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=625606>

[3] <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6096796>

[4] <https://ieeexplore.ieee.org/document/5109763>

[5] Y. Ephraim and D. Malah "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator" Acoustics Speech and Signal Processing IEEE Transactions on vol. 32 no. 6 pp. 1109-1121 1984.

[6] R. McAulay and M. Malpass "Speech enhancement using a soft-decision noise suppression filter" Acoustics Speech and Signal Processing IEEE Transactions on vol. 28 no. 2 pp. 137-145 1980. (Pubitemid 10487373)

[7] P. J. Wolfe and S. J. Godsill "Efficient alternatives to the Ephraim and Malah suppression rule for audio signal enhancement" EURASIP Journal on Advances in Signal Processing vol. 2003 no. 10 p. 910167 2003.

[8] R. Martin "Speech enhancement based on minimum meansquare error estimation and supergaussian priors" Speech and Audio Processing IEEE Transactions on vol. 13 no. 5 pp. 845-856 2005. (Pubitemid 41558900)

[9] S. Doclo and M. Moonen "On the output SNR of the speechdistortion weighted multichannel Wiener filter" Signal Processing Letters IEEE vol. 12 no. 12 pp. 809-811 Dec. 2005. (Pubitemid 41800439)

[10] T. Den Bogaert J. Wouters S. Doclo and M. Moonen "Binaural cue preservation for hearing aids using an interaural transfer function multichannel Wiener filter" in ICASSP 2007 Proceedings vol. 4 April pp. IV-565-IV-568.

[11] B. Cornelis M. Moonen and J. Wouters "Performance analysis of multichannel Wiener filter-based noise reduction in hearing aids under second order statistics estimation errors" Audio Speech and Language Processing IEEE Transactions on vol. 19 no. 5 pp. 1368-1381 2011.

[12] <https://terpconnect.umd.edu/~toh/spectrum/FourierFilter.html>

[13] iiste.org