PHOTOPLETHYSMOGRAPHY SIGNAL BASED HEART-RATE MONITORING SYSTEM USING ARDUINO MICRO-CONTROLLER

PROJECT REPORT

Submitted by

Promit Roy,
Department of Electronics and Communication Engineering
National Institute of Technology Durgapur

Narra Hemanth,
Department of Electronics and Communication Engineering
Jawaharlal Nehru Technological University Hyderabad

Towards the partial fulfilment for the award of the degree of
Bachelor of Technology in
Electronics and Communication Engineering
Under the supervision of

Dr. Priya Ranjan Muduli,
Department of Electronics and Communication Engineering,
Indian Institute of Technology (B. H. U.) Varanasi

ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our project guide, **Dr. Priya Ranjan Muduli**, who provided us the opportunity to work on a technical project on **Embedded Systems** of a **Photoplethysmography Signal based Heart-rate Monitoring System using Arduino micro-controller**, which encouraged us to research extensively on the topics and we have also learned many new tools and technologies related to our work.

Secondly, we would also like to thank our faculties and friends who helped us a lot in finalising this project within the limited time frame.

Date: 21st July'2021

Yours obediently,

Promit Roy

Narra Hemanth

CONTENTS IN DETAIL

ABSTRACT

The objective of this project is the analysis of PPG signal to monitor heart-rate using Arduino micro-controller Board.

In principle, PPG illuminates the skin and subcutaneous tissue with light of a specific wavelength from a LED to measure the organ volume. This light is absorbed, passed through, or reflected. A photodiode produces current corresponding to the measured light intensity in the form of an analog signal. This information can be used to measure the heart-rate of a person.

A green LED is used in the reflective sensor to extract the PPG signal. Due to the wavelength of the green light, which is around 530 nm, it is known to penetrate the tissue lesser than lights of higher wavelengths. Hence, more unabsorbed or reflected light comes out of the tissue with green light than with other colours.

The Signal Conditioning Circuit is simulated using SPICE and the PCB is designed using Altium PCB Designer software, the purpose is to make the photodiode current signal, compatible for the Signal Processing Circuit, the Arduino Uno Micro-controller Board, after filtering out noise signals of unwanted frequencies. PPG sensor SFH7072 is used in the schematic of the Signal Conditioning Circuit which consists of green LED and a broadband photodiode.

The current signal generated from the photodiode is passed through a TIA (Trans-impedance Amplifier). TIA OPA2380 is used for the purpose to convert the current signal, which is in the order of few hundreds of nA, to an amplified voltage signal, in the order of mV. The voltage signal obtained is then passed through a Second Order Band-pass filter for the objective of filtering out noise of unwanted frequencies. The output signal from Second Order BPF needs to be scaled as per the requirement of the analog input pins of Arduino Uno micro-controller Board. Two operational amplifiers are used for the purpose; after the first stage amplification, the negative part of the is shifted by a voltage adder which also further does second stage amplification to adjust the input voltage signal in the range of 0 - 5V. This analog output signal of the Signal Conditioning Circuit is then given as input to the Arduino Uno board for further processing. The Arduino Uno Board is consist of ATmega328p microprocessor which have 10-bit ADC and 16 MHz clock that can be used to process the filtered and amplified PPG signal to measure the heart-rate.

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

Symbol	Explanation
ADC	Analog-to-Digital Converter
AWGN	Added white gaussian noise
AVR	Advanced Virtual RISC
HRM	Heart Rate Monitoring
IR	Infrared
LED	Light Emitting Diode
LPF	Low-Pass Filter
PCB	Printed Circuit Board
SO-BPF	Second Order Band-Pass Filter
TIA	Trans-impedance Amplifier

1. INTRODUCTION

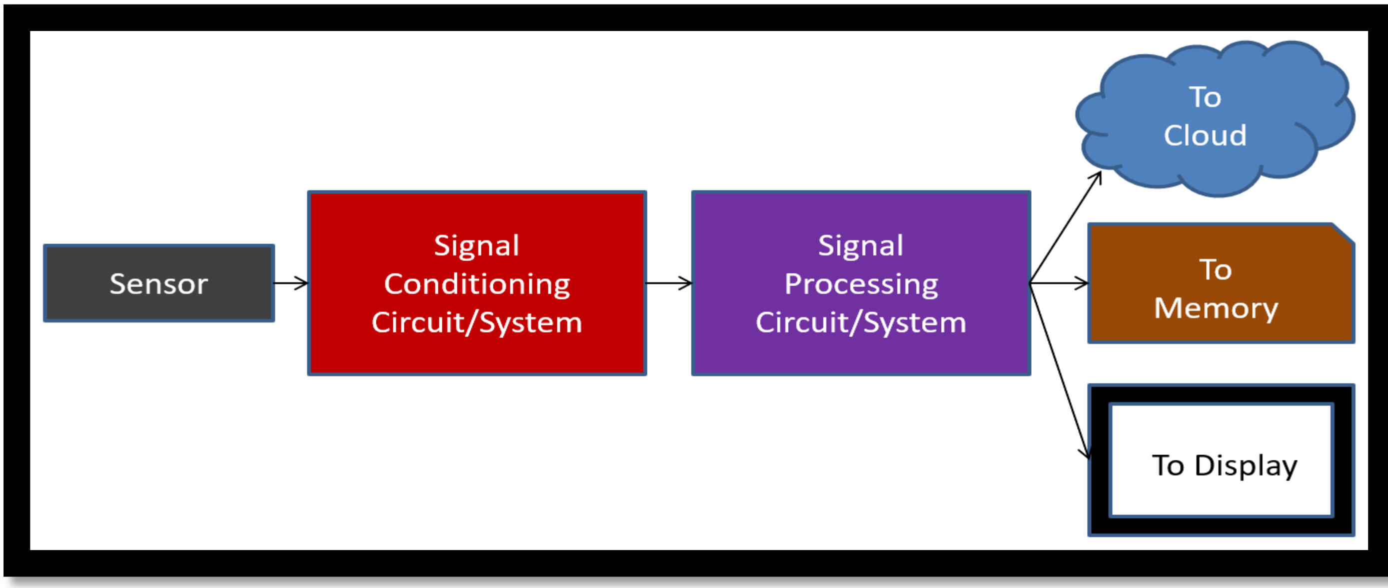
In this contemporary regime, we need everything obtainable in one click. Due to this present pandemic, demand for medical equipments has increased rapidly. With increase in demand, the need for urbane and competent devices has increased. The scope for innovation and research has reached to the next level. Heart rate monitoring has become an integral part of our life. There are many ways to monitor heart rate but the easiest way is to analyze the PPG signals. PPG is an optical measurement of the volume of an organ. In principle, PPG illuminates the skin and subcutaneous tissue with light of a specific wavelength from a light-emitting diode (LED) to measure organ volume. This light is absorbed, passed through, or reflected back. A photodiode sensor measures the light that is either transmitted or reflected, depending on where it is placed relative to the LED. The light is then converted to an electrical signal. The information can be used to determine the heart rate of a person. PPG also provides more information about blood flow and blood pressure. This measurement can be conducted at various locations on the body to examine the blood flow to different regions. When measured closest to the aorta of the heart (for example, the left arm), some additional information can be gained regarding the cardiac output and heart valve function. One advantage that PPG has is the number of skin contacts which are required to measure it. Because PPG can be determined from reflected or transmitted light, only a single point of contact is necessary to measure it. This feature allows for easy and continuous time measurements, which is the most attractive advantage to wearable electronics such as fitness trackers.

1.1 Problem Statement and Research Objectives

1.1.1 Problem Statement

Analyzing the PPG signals to monitor heart-rate.

The Signal Conditioning Circuit should be designed such that the PPG signal should be compatible for Arduino Uno. The Analog Output from Signal Conditioning Circuit is given as input to Arduino Uno Board for processing the signal. The Arduino Uno is our Signal Processing Circuit. After the processing, the output can be sent to cloud for remote acccess, or to monitor for display, or can be stored in a memory.



1.1.2 Research Objectives

Designing an effective and innovative Signal Conditioning Circuit by resolving problems of previous works.

1. Analog Front-End for PPG sensor.
2. Single LED Photoplethysmography based Non-invasive Glucose Monitoring Prototype System.
3. An Arduino Compactible Signals and Electronics teaching system.
4. Estimation of Breathing Rate and Heart Rate from PPG.
5. Testing the Circuit for real life PPG signals.

1.2 Organization of this report

This report is classified into four parts:

1) Introduction
2) Methodology and Experimentation
3) Results and Discussion

The Introduction part has the basic information about the methods used and idea of the project. It explains the flow of the project. It also deals with the review of Research aspects involved in the project.

The next part has the PSPICE aspects of the projects. It consists of the Step-by-Step building of the Schematic and Simulation. It contains the Simulation outputs of each and every step of creation of the final circuit.

The final part has the results and discussion of the outputs.

2. METHODOLOGY AND EXPERIMENTATION

3.1 Components Selection

The selections of components play a major role in shaping the outcome of the project. The components used in the simulation are available at: https://www.digikey.in/. It is taken care that all the passive elements are available in market. The major components involved are:

Table 1.1 Bill of Materials

Comment	Footprint	LibRef
22uF	INDC2112X145N	LMK212BBJ226MG-T
0.39uF	CAPC2012X140N	0805YC394JAT2A
0.68uF	CAPC1608X90N	C1608X7R1C684K080AC
10pF	CAPC1608X90N	06035A100JAT2A
27nF	CAPC5750X330N	06033C104KAT4A
TSW-101-24-G-T	SAMTEC_TSW-101-24-G-T	TSW-101-24-G-T
100	RESC0603X26N	RC0201FR-07100RL
220K	RESC1005X40N	CRGCQ0402F220K
10K	RESC0603X26N	RMCF0201FT10K0
100K	RESC1005X40N	RR0510P-104-D
5.1K	RES_ERA3AEB512V	ERA3AEB512V
240	RES_ERA2AED241X	ERA2AED241X
43K	RES_ERJ1GNF4302C	ERJ1GNF4302C
3.6K	RESC1005X35N	RMCF0402FT3K60
160	RESC0603X26N	RC0201FR-07160RL
82K	RESC1005X40N	CRGCQ0402F82K
7.5K	RESC1508X50N	RNCP0603FTD7K50
510	RESC0603X26N	RC0201FR-07510RL
400K	RESC2012X50N	CRCW0805400KJNTA
100K	RESC0603X26N	RC0201JR-07100KL
300K	RES_ERA3AEB304V	ERA3AEB304V
56K	RESC1005X40N	CRGCQ0402F56K
270K	RESC2012X65N	CRG0805F270K
22K	RES_ERA2AEB223X	ERA2AEB223X
750	RES_ERA3AEB751V	ERA3AEB751V
1000K	RESC0603X26N	RC0201FR-071ML
SFH_7072	XDCR_SFH_7072	SFH_7072
TIA 1(OPA2380)	SOP65P490X110-8N	OPA2380AIDGKT
OP AMP-BPF	SOT95P280X145-5N	LMH6609MFX/NOPB
OP AMP-Amp Stage-2	SOT95P280X145-5N	LMH6609MFX/NOPB
LMH6609MFX/NOPB	SOT95P280X145-5N	LMH6609MFX/NOPB

3.2 PSPICE Simulation of Individual Components

The Individual analysis of important components involved in the project. The main components are:

1) LED from SFH7072 Sensor.
2) Photodiode from SFH7072 Sensor
3) Transimpedance Amplifier (OPA2380)
4) Second Order Bandpass Filter

1) LED from SFH7072 Sensor

The LED used is Green LED. The LED is the Transmitter of the light. The light from the LED hits the subcutaneous tissues. In the Figure 1.1, the Schematic is shown. The Figure shows the supplied voltage TX_SUP and the passive elements used for maintaining the voltage and current flow through the diode. With these arrangements the LED is producing a current of value around 20 mA as shown in Figure 1.2. This wavelength of the light produced is 526nm.

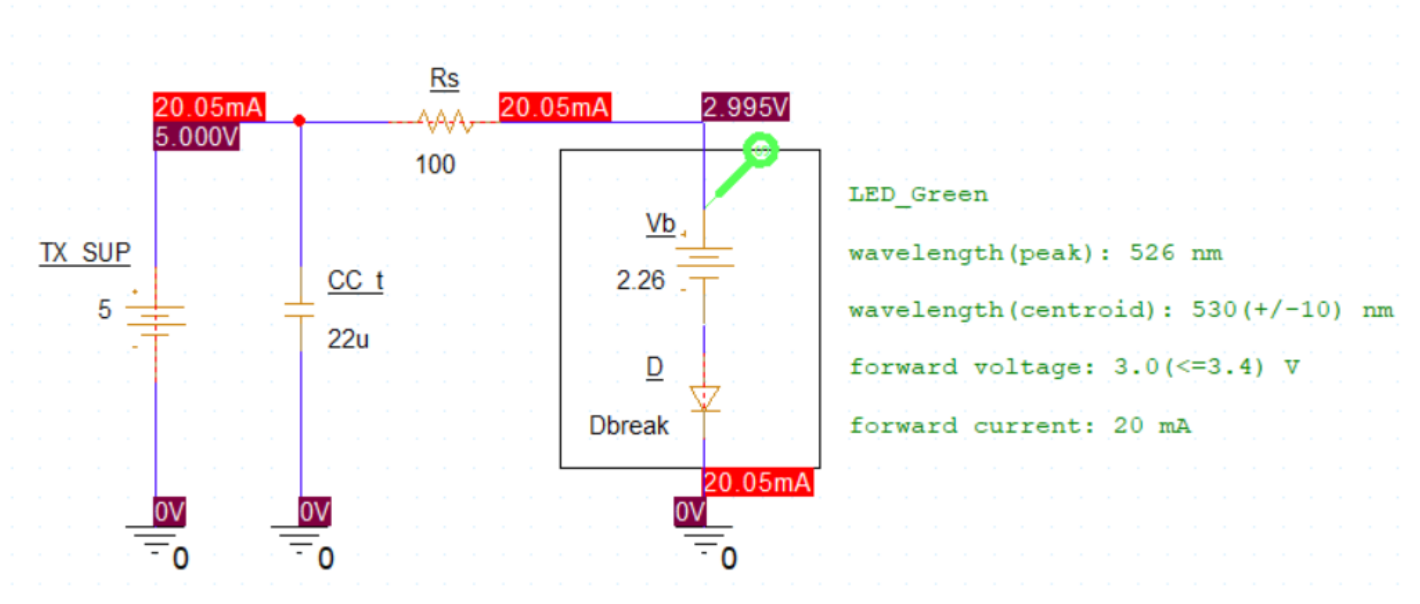


Figure 1.1

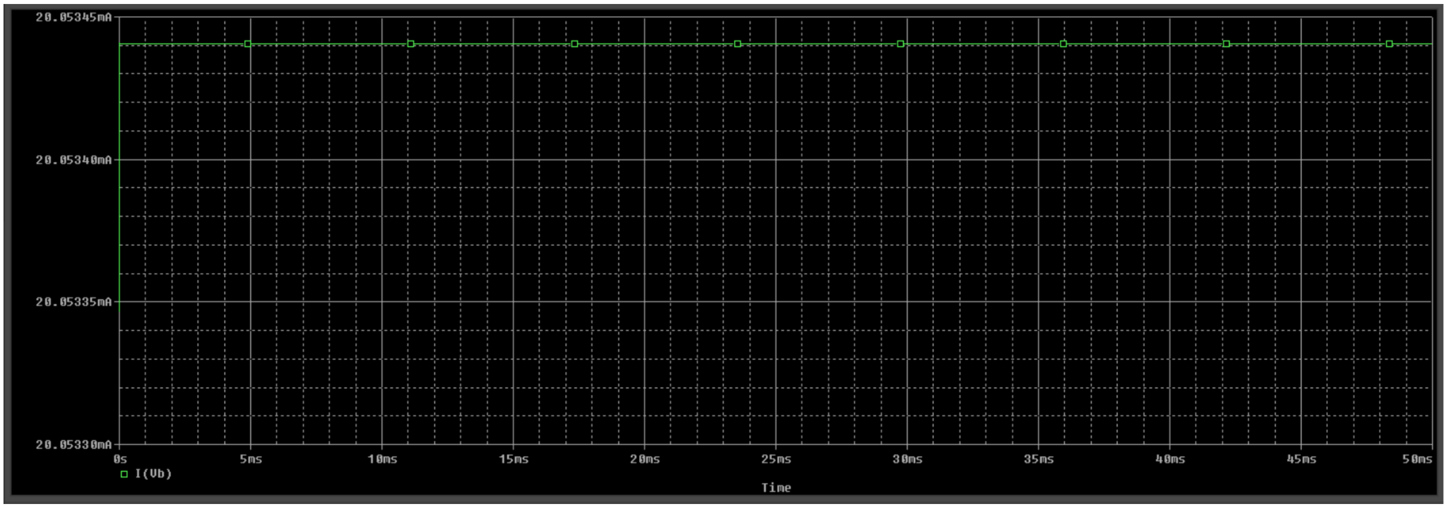


Figure 1.2

2) <u>Photodiode from SFH7072 Sensor</u>

Generally, a photodiode is a diode which generates current when exposed to light. In this case, we are going to use the current for our voltage signal generation. Figure 1.3 shows the schematic of the photodiode. The light transmitted by the LED, after the reflection on the body surface is received by this photodiode. The photodiode acts as the receiver of the signal. For the simulation purpose, we have use the PWL PSPICE modelling source to show the output signal. The synthetic current signal generated is shown in Figure 1.4.

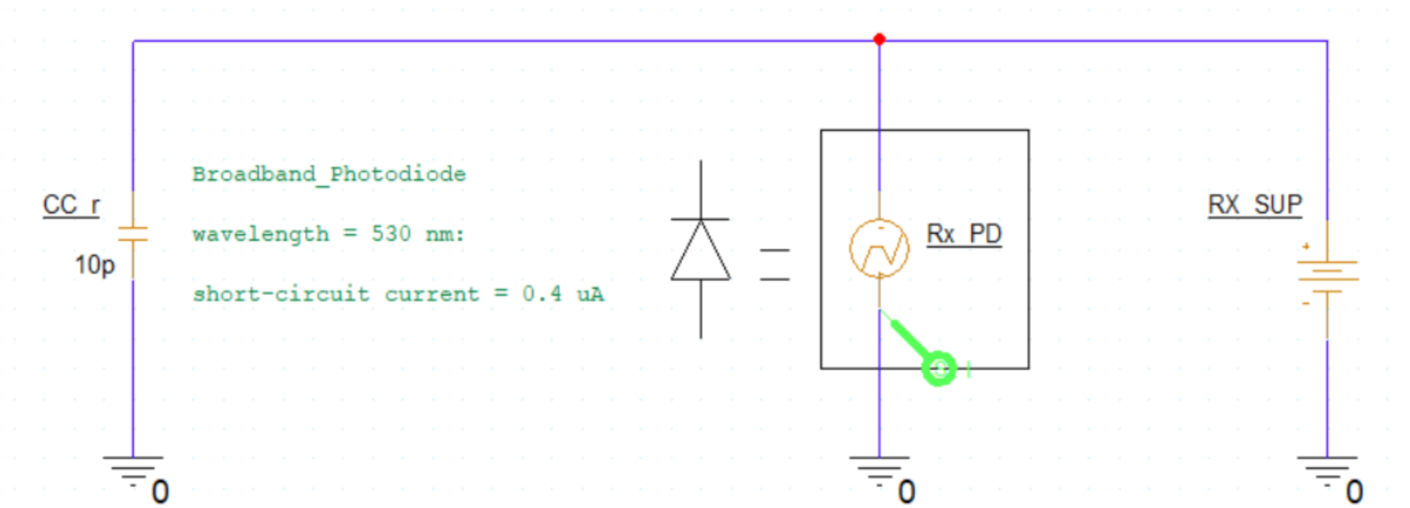


Figure 1.3

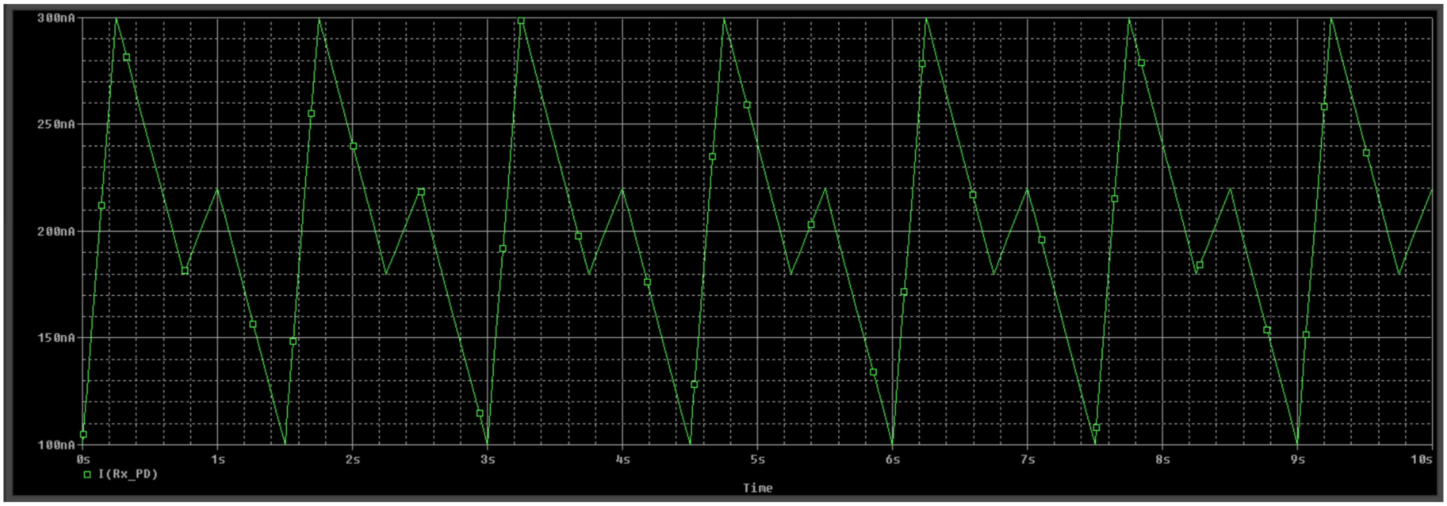


Figure 1.4

3) Trans-impedance Amplifier

The Trans-impedance Amplifier is a device which can convert the current signal to voltage signal. OPA2380 is used in this project. The schematic is shown in Figure 1.5. The capacitor Cp is used to eliminate the high frequency signals, i.e. it will also work as a LPF. The resistor Rp is used to control the output. The output voltage signal is proportional to input current signal.

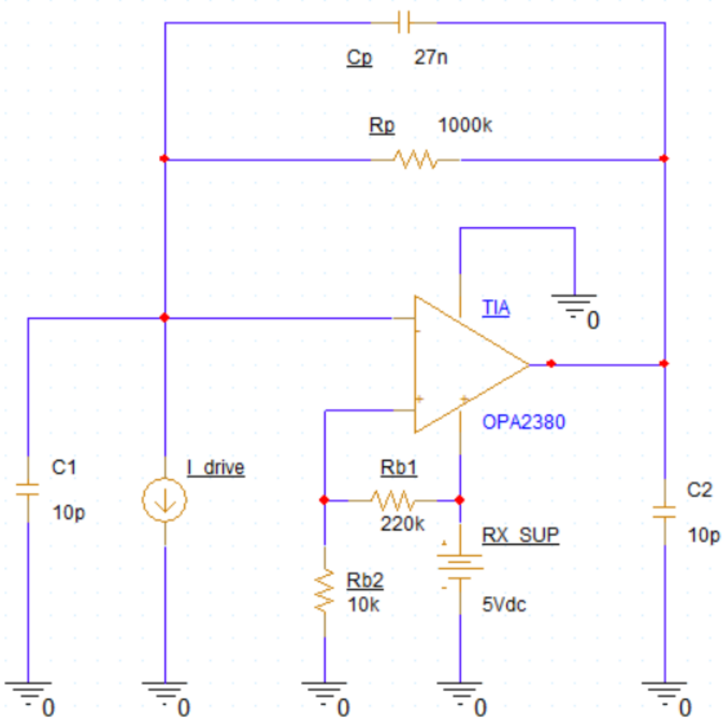


Figure 1.5

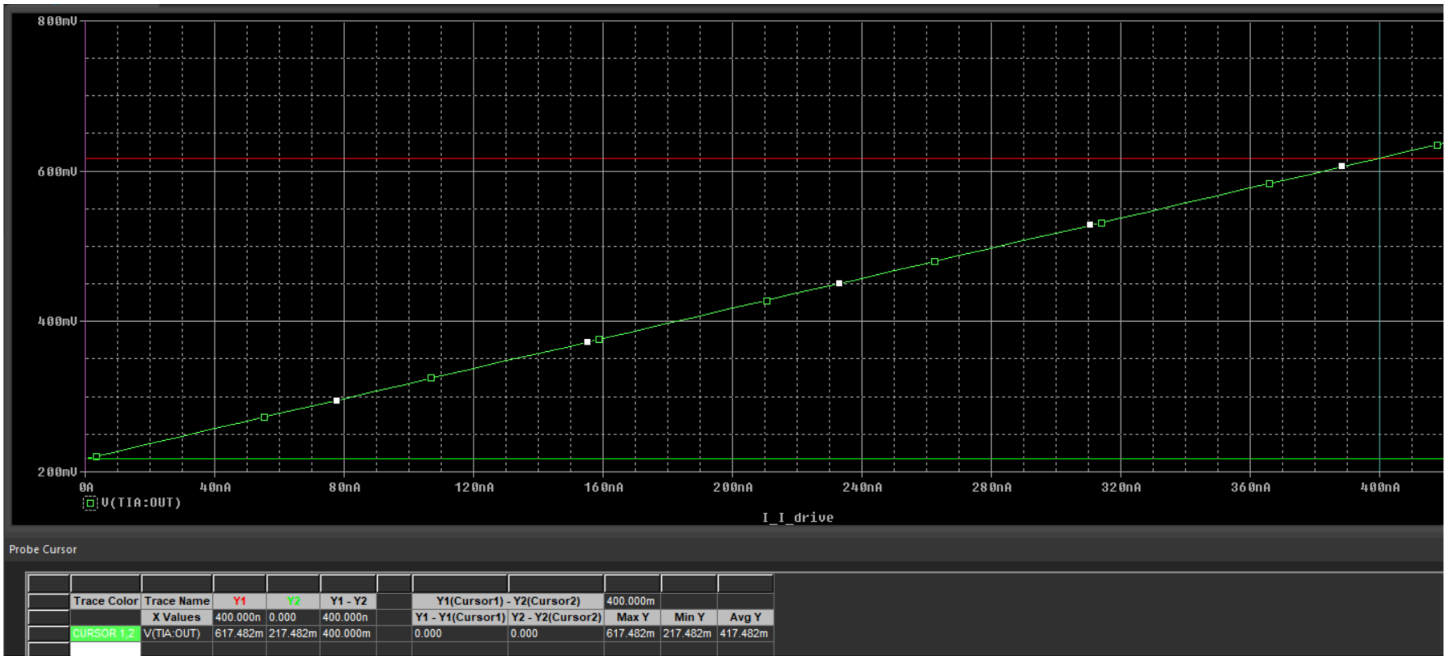


Figure 1.6

In Figure 1.6, we simulated the DC Sweep simulation of the TIA and we can conclude from the figure that the voltage corresponding to minimum value of current, i.e. at 0 nA is 217.482 mV and the voltage corresponding to maximum value of current, i.e. at 400 nA is 617.482 mV.

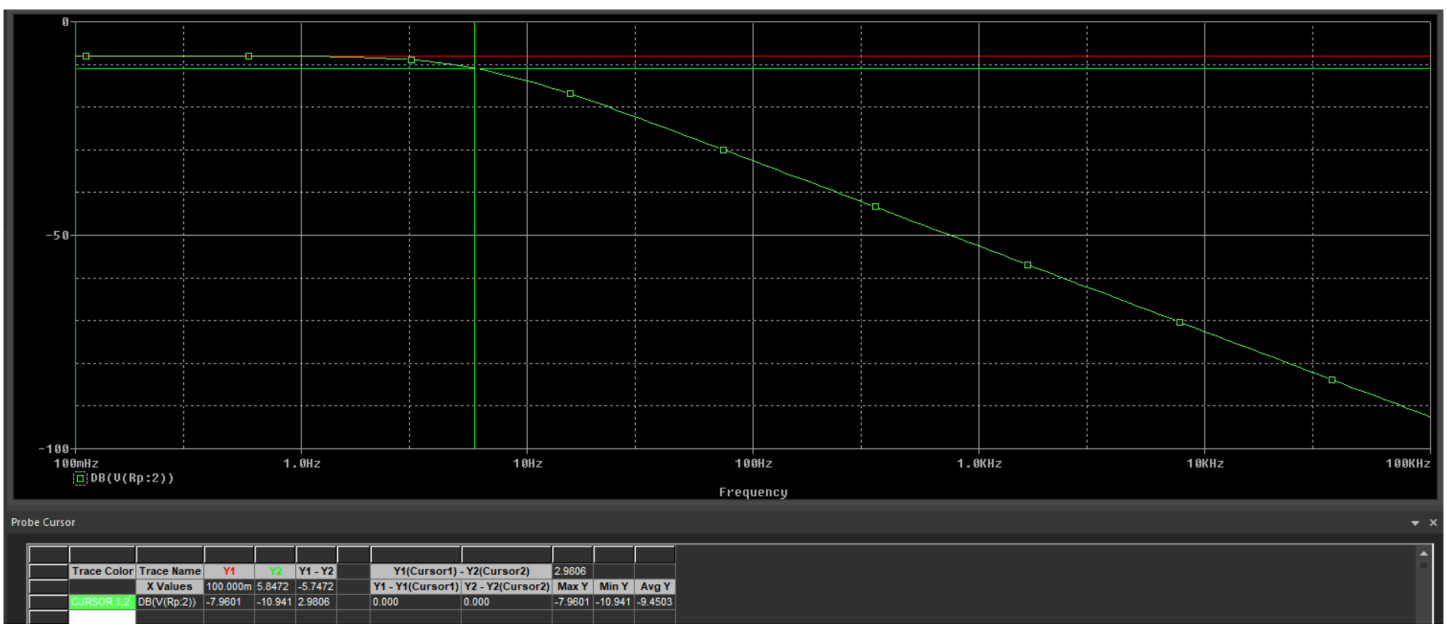


Figure 1.7

In Figure 1.7, we simulated the Frequency Response of the TIA used in our design and we can observe that the -3 dB frequency, i.e. the cut-off frequency of the LPF is about 5.8472 Hz.

4) <u>Second Order Band-pass Filter</u>

The first order filters can be easily converted into second order filters simply by using an additional RC network within the input or feedback path. Then we can define second order filters as simply being two 1st-order filters cascaded together with amplification circuit. These filters are used in a communication system for choosing the signals with a particular bandwidth. The schematic is shown in Figure 1.10. This filter is tested under various frequencies as shown in Figure 1.11, 1.12 & 1.13. The output from the TIA is filtered using this filter.

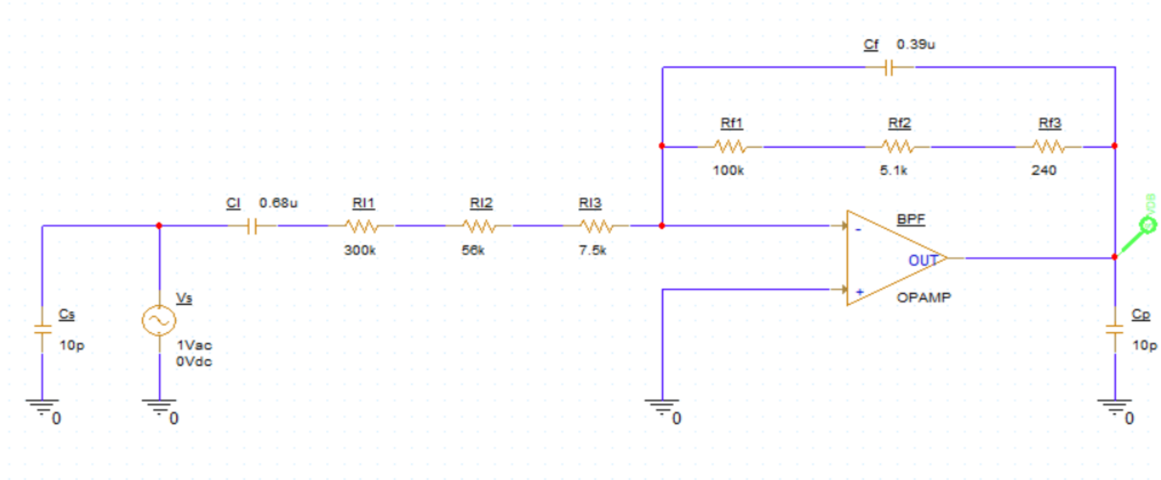


Figure 1.10

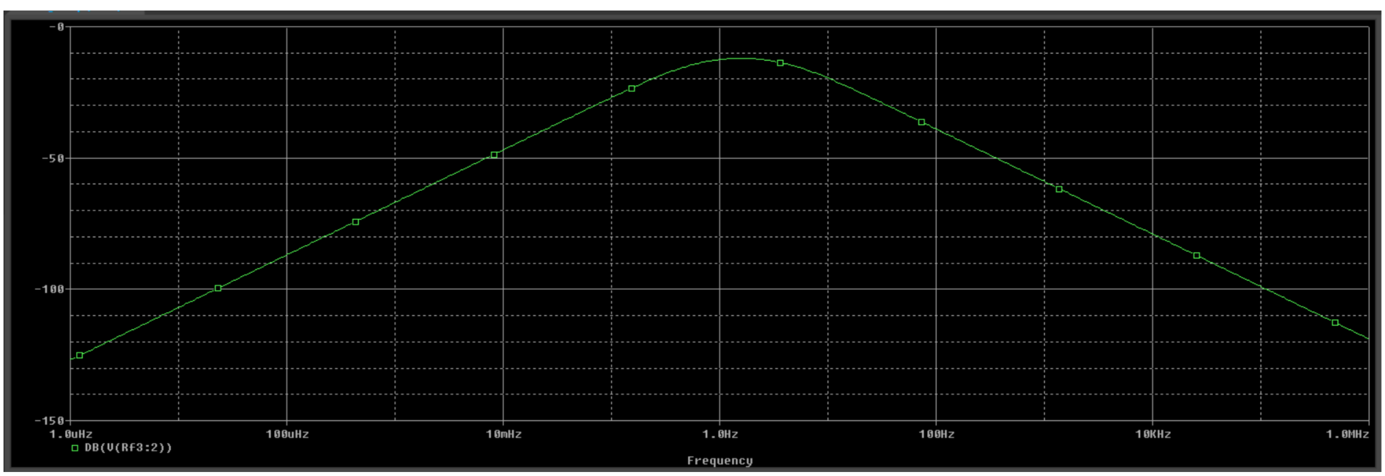


Figure 1.11

According to the previous research works cited below in the references, range of the frequencies of PPG signals are in the frequency range from 0.5Hz to 5Hz. In the Figure 1.11, we can observe the Frequency response of Second order Band-pass Filter.

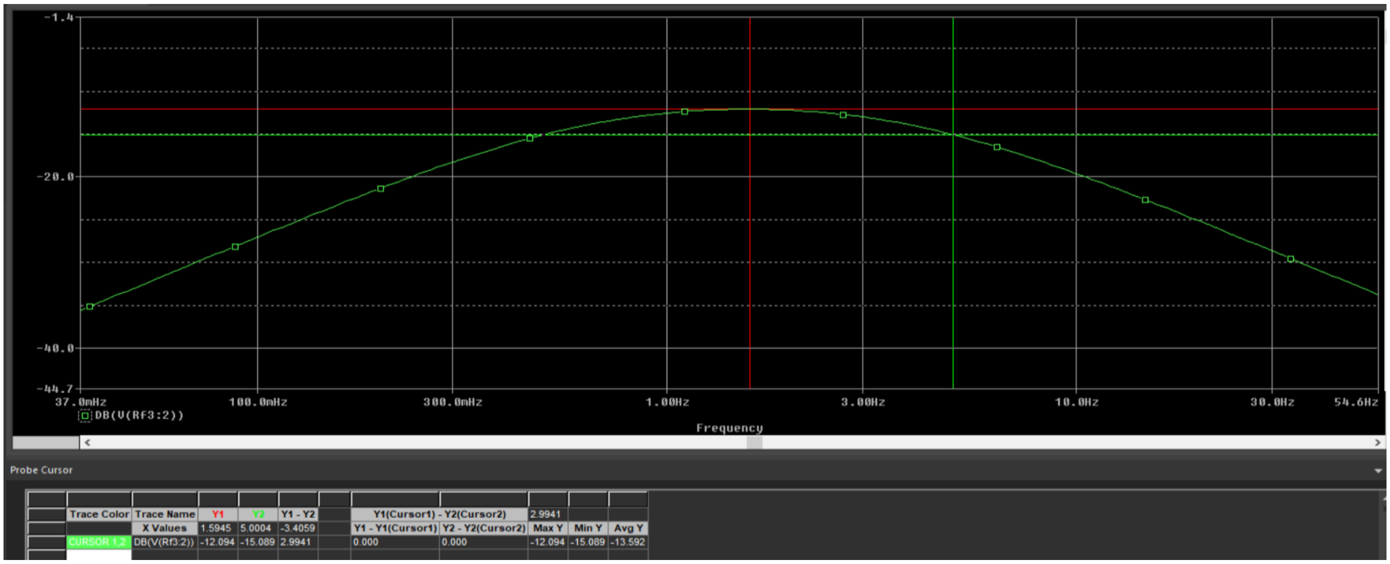


Figure 1.12

In the Figure 1.12, we can observe that the resonant frequency is approximate around -12.004 dB and the -3 db frequency, i.e. the high cut-off frequency is around 5.0004 Hz.

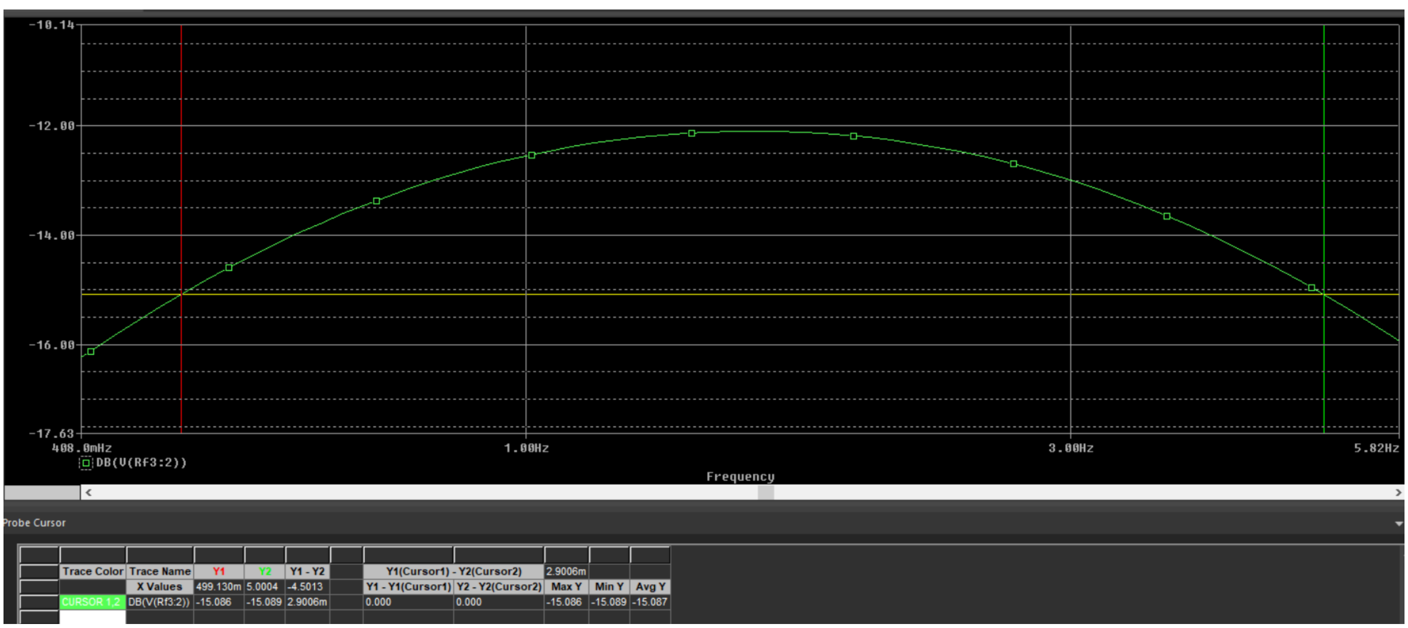


Figure 1.13

In the Figure 1.13, we can observe that the -3 db frequency, i.e. the high cut-off frequency is around 5.0004 Hz and the low cut-off frequency is around 499.130 mHz of this BPF.

Error Calculation:

Error Margin = (Measured Bandwidth – Exact Bandwidth) / (Exact Bandwidth)

Error Margin = ((5.004-0.499130) – (5-0.5)) / (5-0.5)

Error Margin = 0.0018222

So, the error margin is low proving the good performance of the model.

3.3 Integration and Step-by-Step Simulation

The complete simulation is done step-by-step to reduce the occurrence of any error. The steps involved in integration are:

Step 1: Photodiode and the TIA.

Step 2: TIA and the Second Oder BPF.

Step 3: Photodiode, TIA and the Second Order BPF.

Step 4: TIA, Second Order BPF and Amplifier.

1) Photodiode and TIA

The output of the photodiode and TIA need to be analysed. The current signal generated in the photodiode, when passed though a TIA converts it into corresponding voltage signal. The combined schematic is shown in Figure 1.8. The waveform obtained after performing the simulation of the previously generated synthetic signal (Figure 1.4) is shown in Figure 1.9.

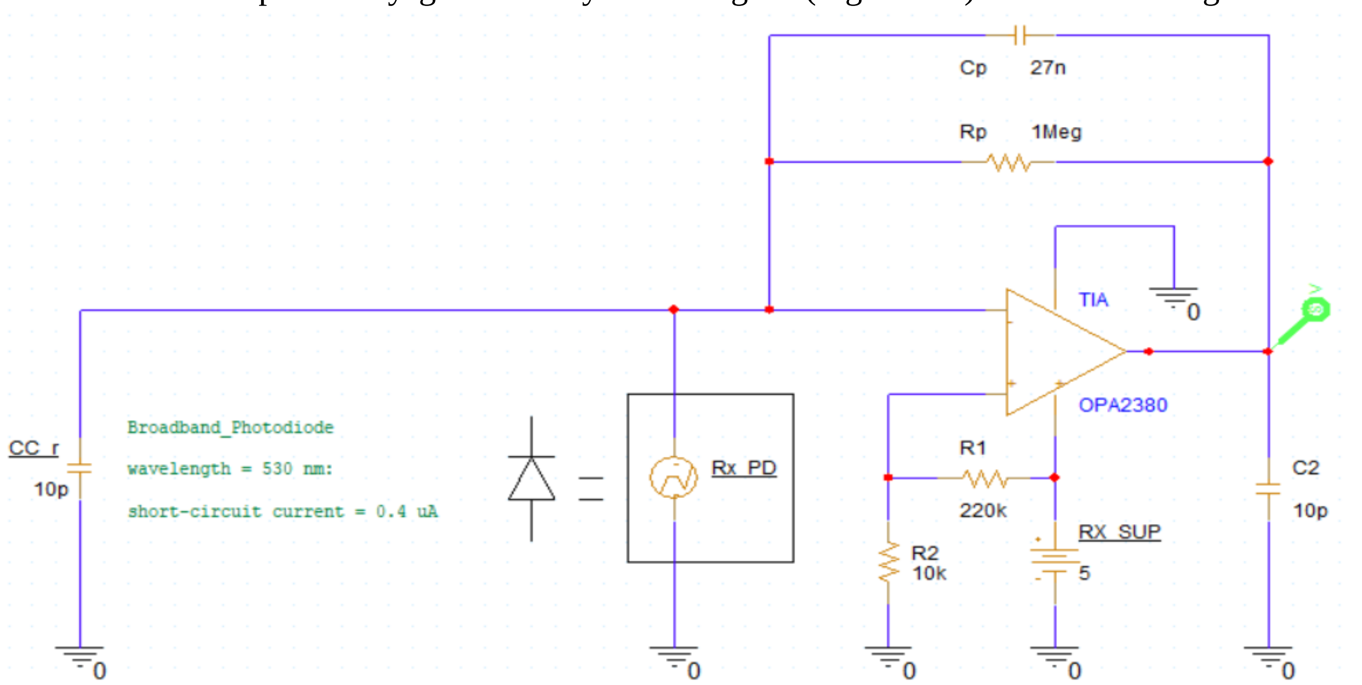


Figure 1.8

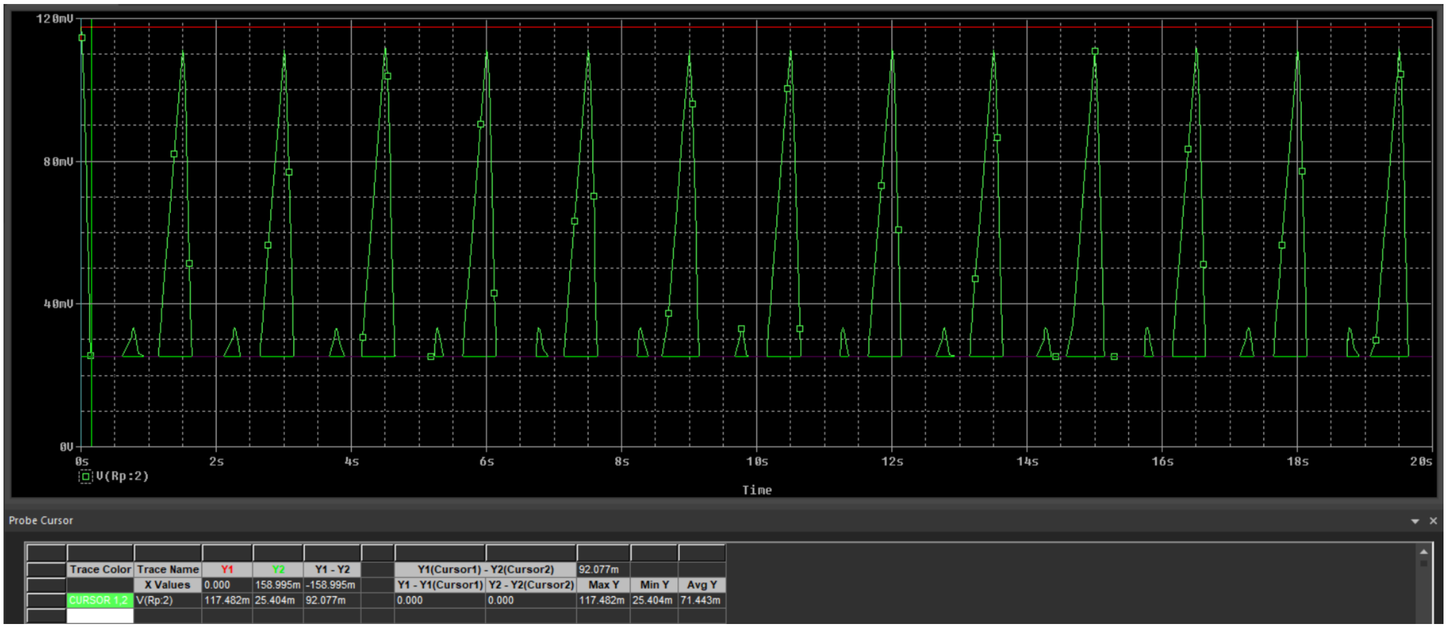


Figure 1.9

2) TIA+SO-BPF

The TIA and SO-BPF is shown in Figure 1.16. The simulation is done with sinusoidal current signals from photodiode of peak values 0 and 400 nA of frequencies ranging from 0.5 Hz to 5 Hz at an interval of 1.5 Hz, i.e. four signals of frequencies: 0.5 Hz, 2 Hz, 3.5 Hz, and 5 Hz, as shown in Figure 1.17 to check the performance of this model. The voltage signal is measured across the BPF for the various frequencies. The simulation output for this sub-circuit is shown in Figure 1.18. In the next step, our objective is to find the maximum and minimum value of voltage outputs as shown in Figures 1.19 & 1.20. The maximum and minimum voltages are noted, and with the help of the noted values, we can further amplify the signal in the required range for processing.

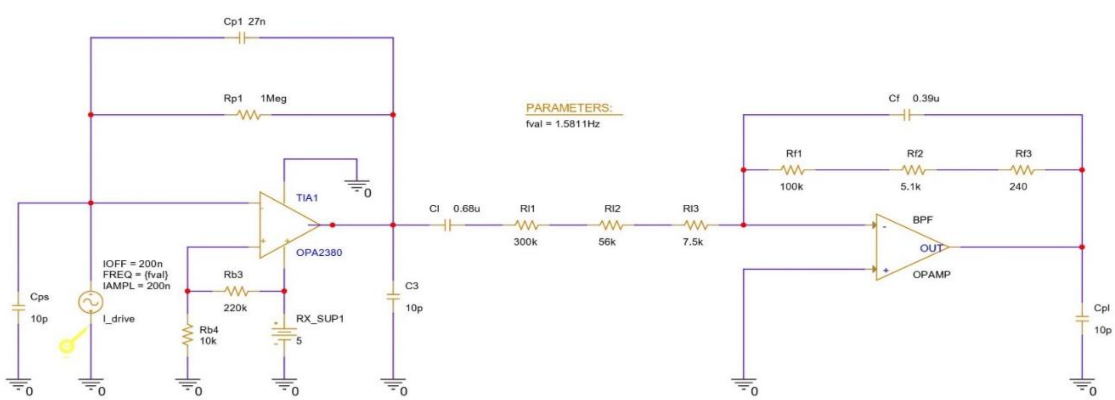


Figure 1.16

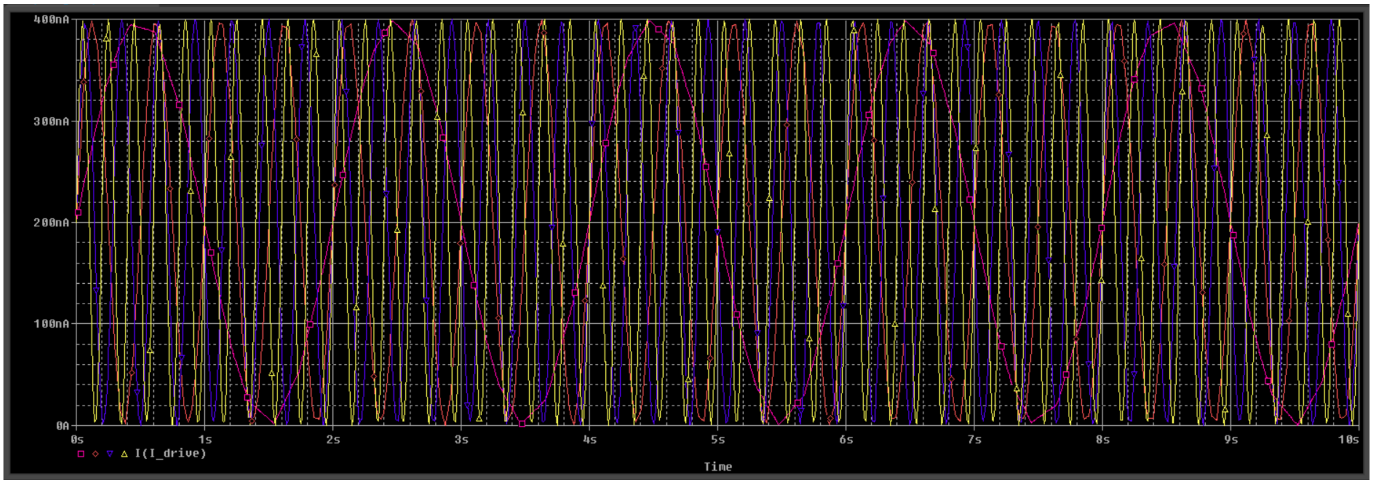


Figure 1.17

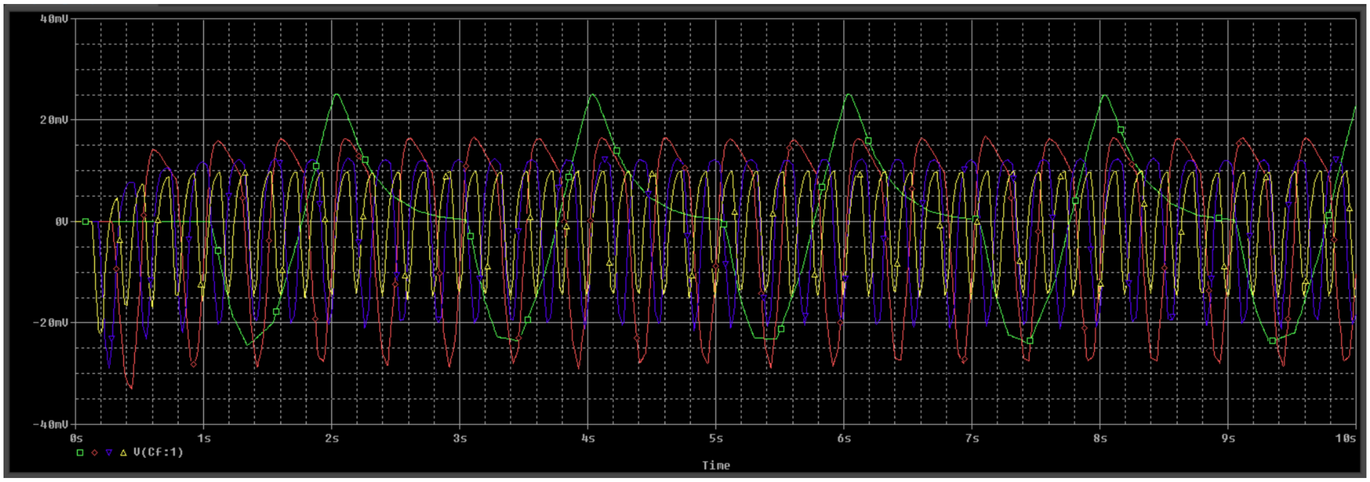


Figure 1.18

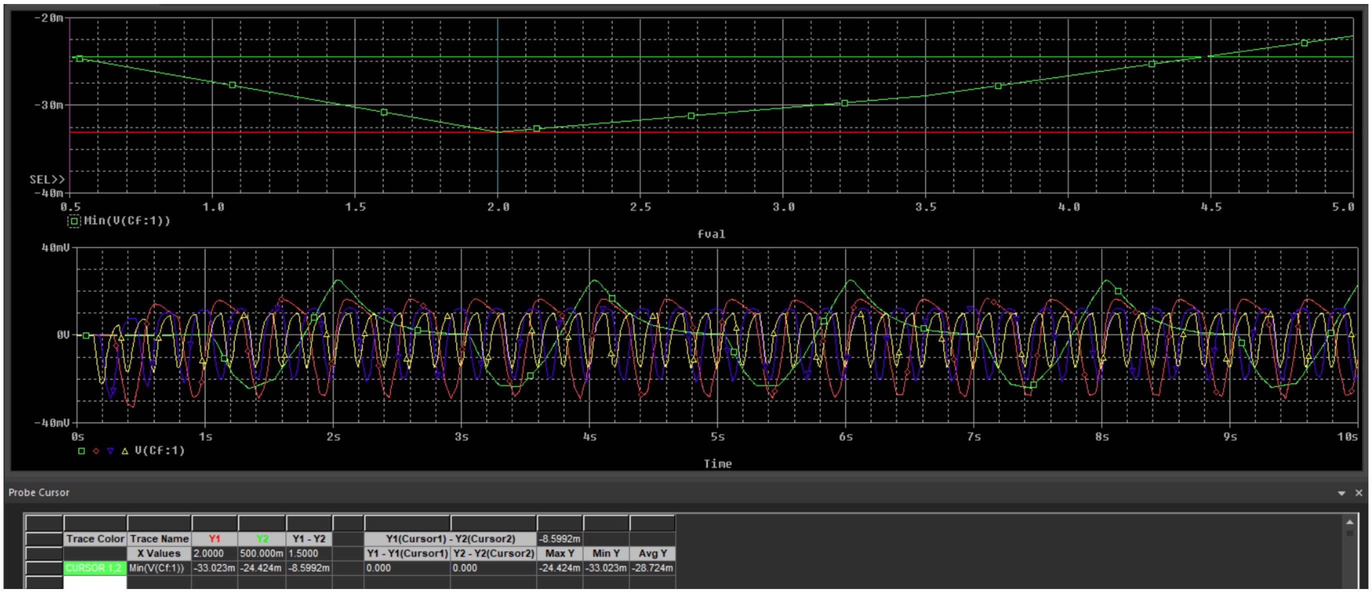


Figure 1.19

The minimum value of all the test signals of different frequencies in Figure 1.19 is -33.023mV.

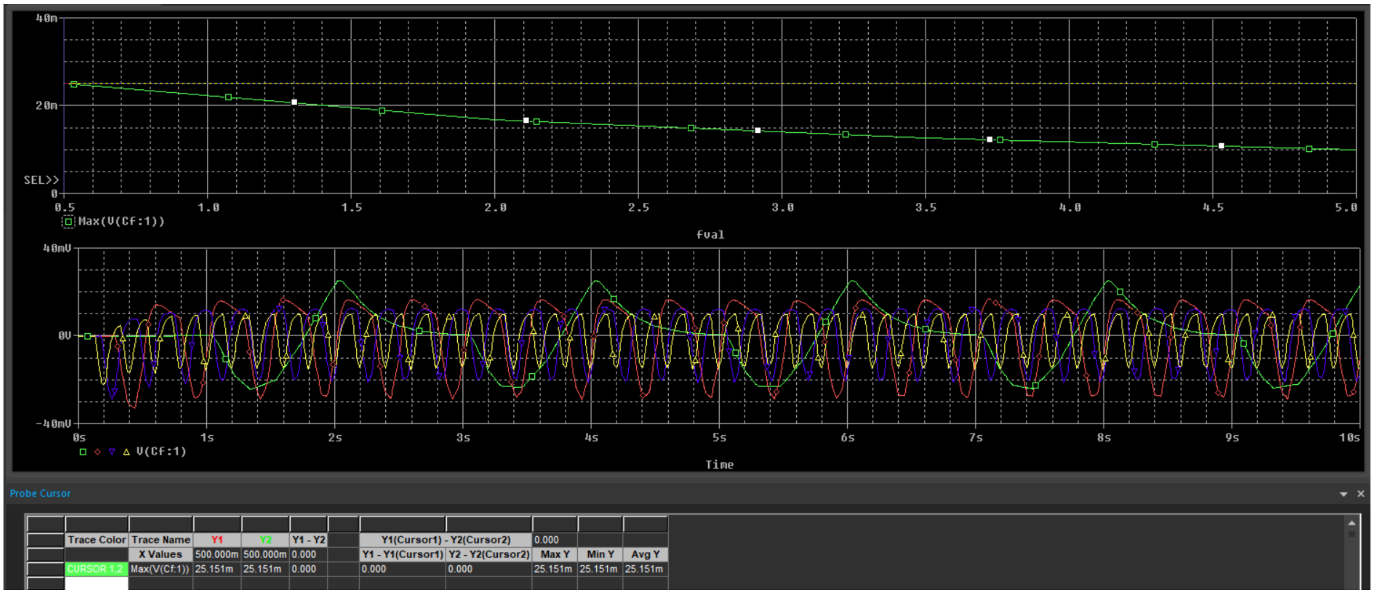


Figure 1.20

The maximum value of all the test signals of different frequencies in Figure 1.20 is 25.151mV.

3) Photodiode, TIA and SO-BPF

The next step is integrating the previous models. The photodiode, TIA and SO-BPF are integrated to check the outputs. The schematic is shown in Figure 1.14. Now, the signal has converted into a PPG signal. But, the signal is very weak. It should be amplified to the range 0 - 5 V. There is also a need for removal of negative part of signal. The waveform obtained after performing the simulation of the previously generated synthetic signal (Figure 1.4) is shown in Figure 1.15.

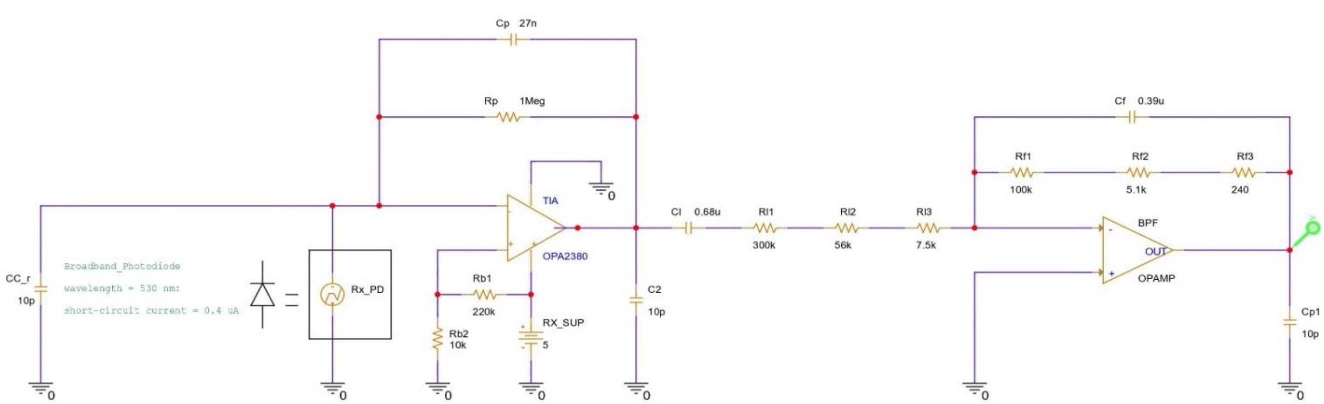


Figure 1.14

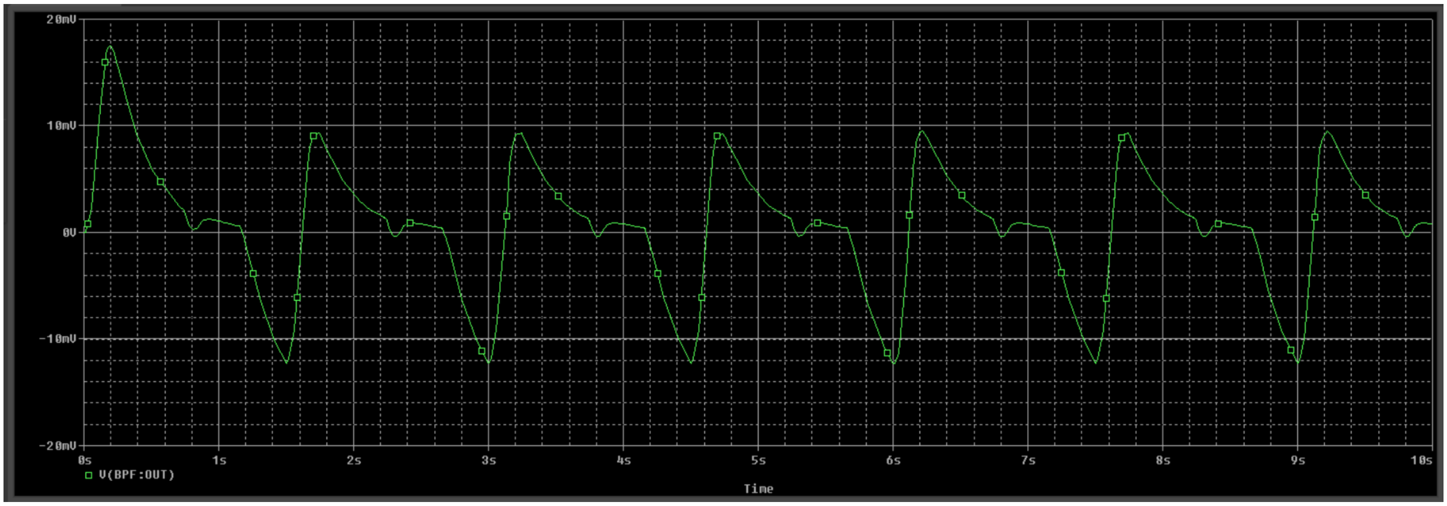


Figure 1.15

4) TIA, SO-BPF and Stage-I_Amplifier

The PPG signals generated in previous model need to be amplified. The resulting signal from the BPF is amplified using two operational amplifiers to make the signal compatible for the analog input pins of Arduino micro-controller. The Stage-I is a non-inverting operational amplifier and Stage-II is a voltage adder amplifier. The Integrated schematic is shown in Figure 1.21. The simulation output is shown in Figure 1.22. The output has been amplified but we can observe negative part of the signal. It should be removed. The maximum and minimum values of the signals according to Figure 1.23 and Figure 1.24 respectively are noted for next step.

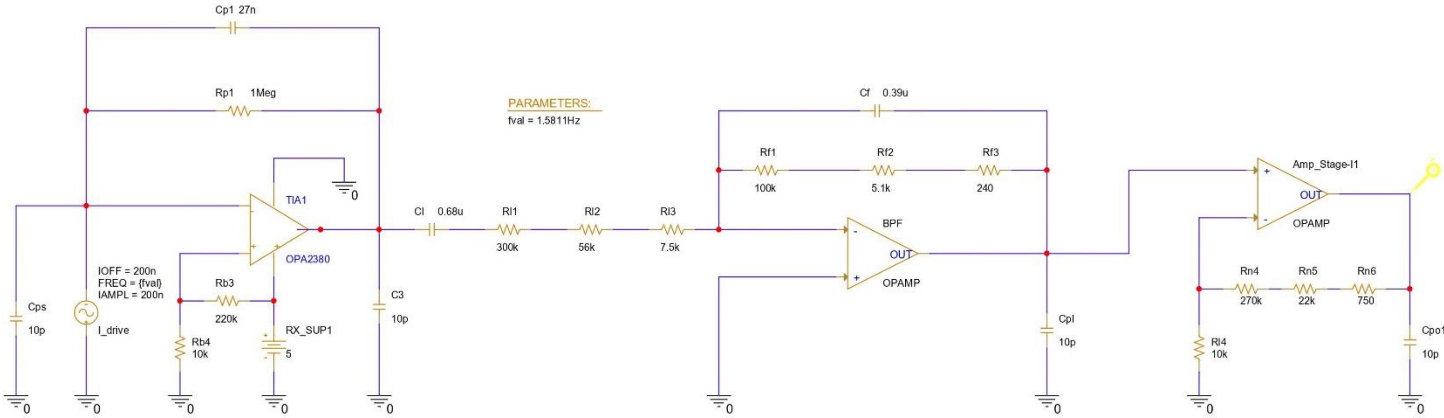


Figure 1.21

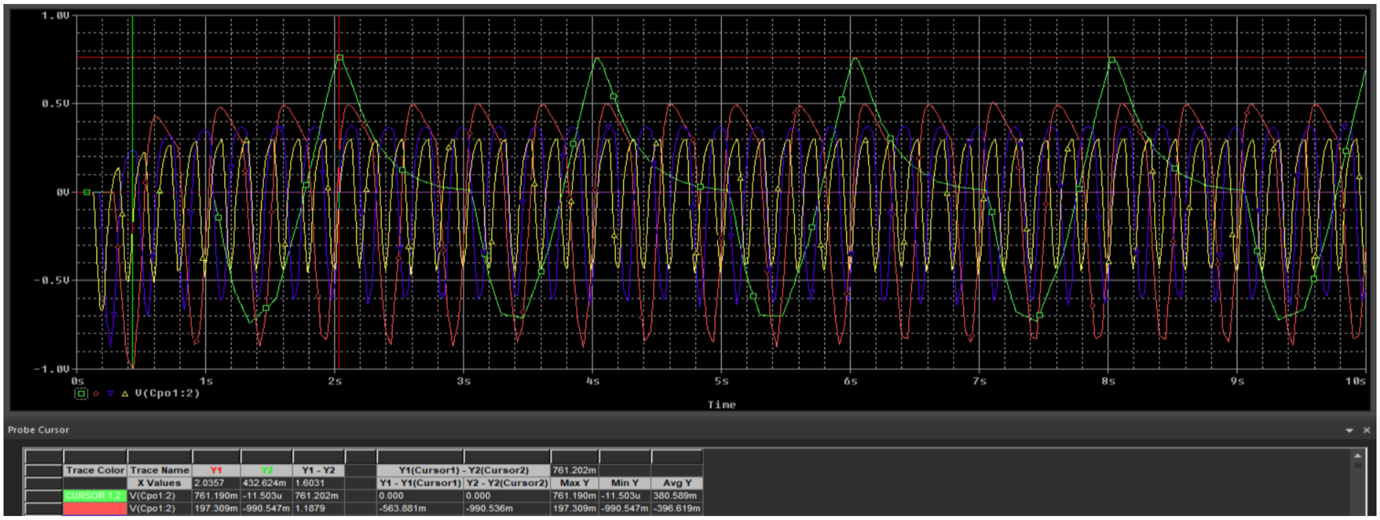


Figure 1.22

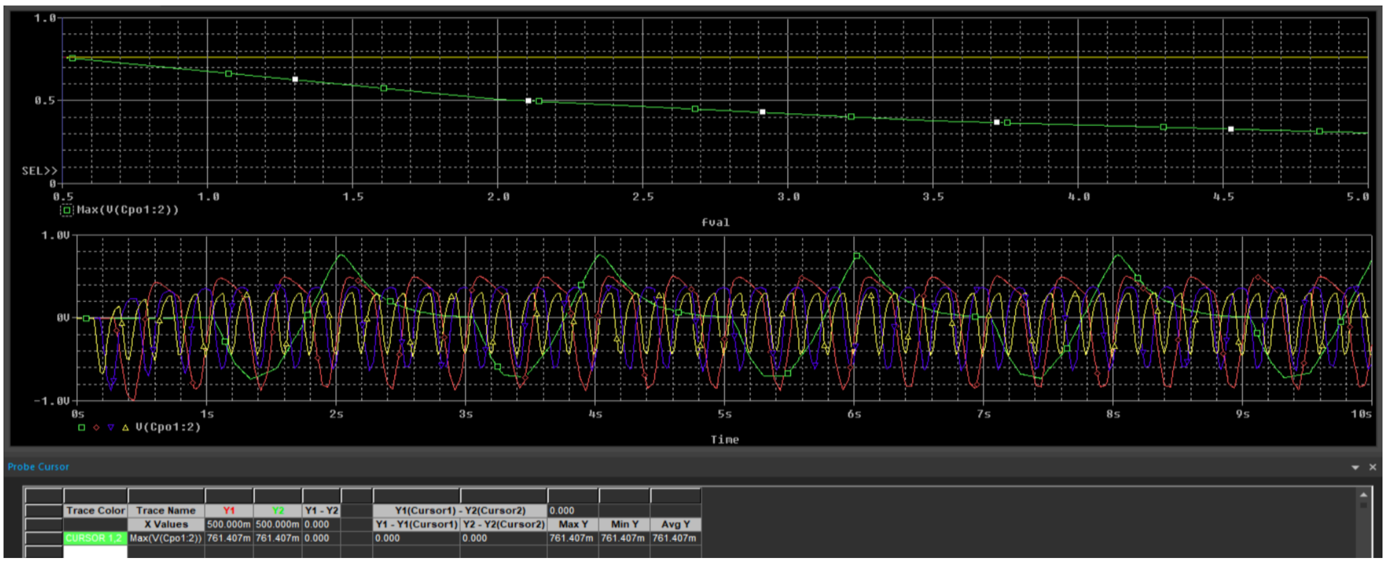


Figure 1.23

The maximum value of the voltage signal generated after the first-stage amplification is about 781.407 mV as displayed in the simulation result in Figure 1.23.

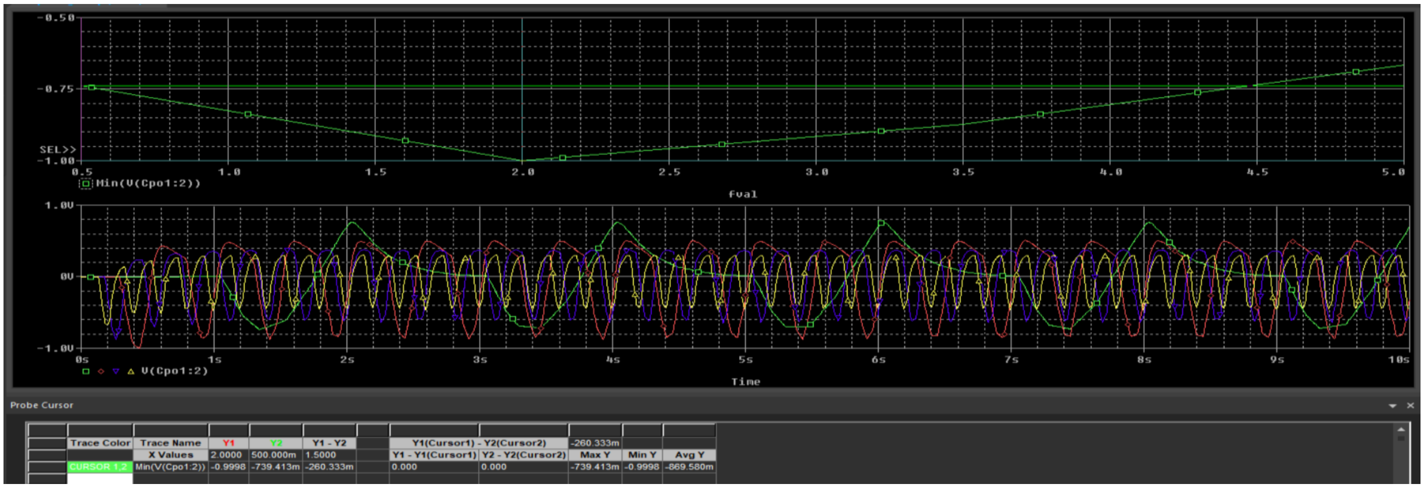


Figure 1.24

The minimum value of the voltage signal after the first-stage amplification is about -0.9998 V as displayed in the simulation result in Figure 1.24.

3.4 Scaling the output signal for the Compatibility of Arduino

The scaling of the signal is very important step. The Arduino Uno micro-controller is compatible in the range of 0 to 5V. So, our signal must fall within that range for the proper results and also there is a possibility that some quantisation level during analog to digital conversion may remain completely unused for any kind of signal.
ATmega328p micro-processor has a 10-bit ADC.
So, the total number of quantisation levels = 2^{10} = 1024.
The following are the corresponding voltage values at the corresponding quantization level: -
0^{th} level: (5/1024)*0 = 0 V.
1^{st} level: (5/1024)*1 = 0.0048828125 V.
…
…
…
1023^{rd} level = (5/1024)*1023 = 4.995117188 V.
Hence, the minimum value of the input signal to the Arduino micro-controller among all the PPG signals of different frequencies should lie in the range of (0 – 0.0048828125 V) and the maximum value of the input signal to the Arduino micro-controller among all the PPG signals of different frequencies should lie in the range of (4.995117188 V – 5 V).

1) TIA, BPF, Stage-I Amplifier and shifted voltage signal

In this step, we are going to integrate the previous steps and shift the voltage signal using an external bias voltage supply. The voltage is shifted to remove the negative part of the signal. The schematic is shown in the Figure 1.25. The simulation output in Figure 1.26 shows the removal of the negative part. The maximum and minimum values of the signal are noted for analysis, which is examined in Figure 1.27 and 1.28 respectively.

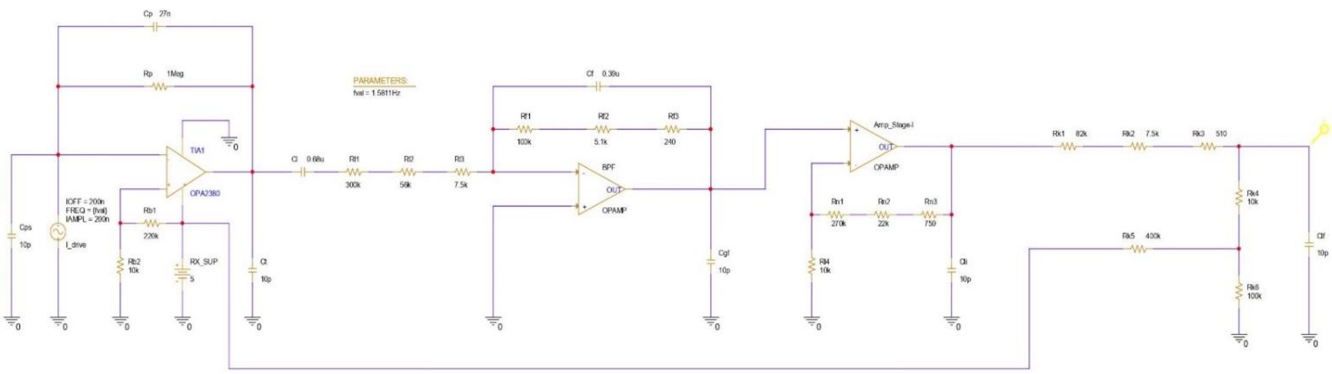


Figure 1.25

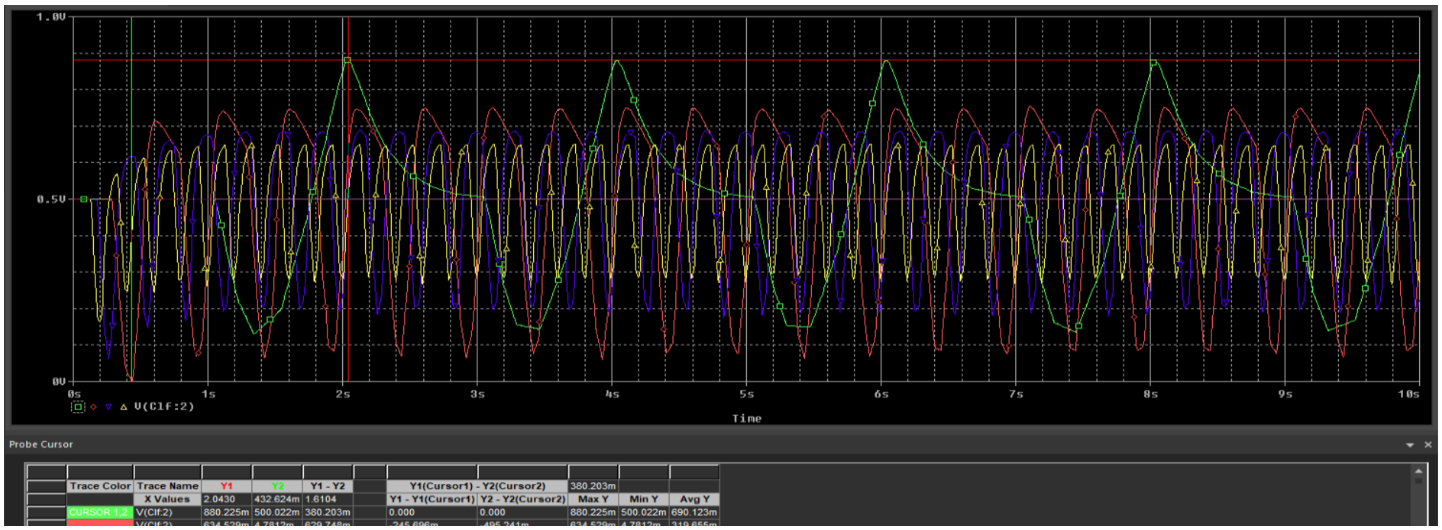


Figure 1.26

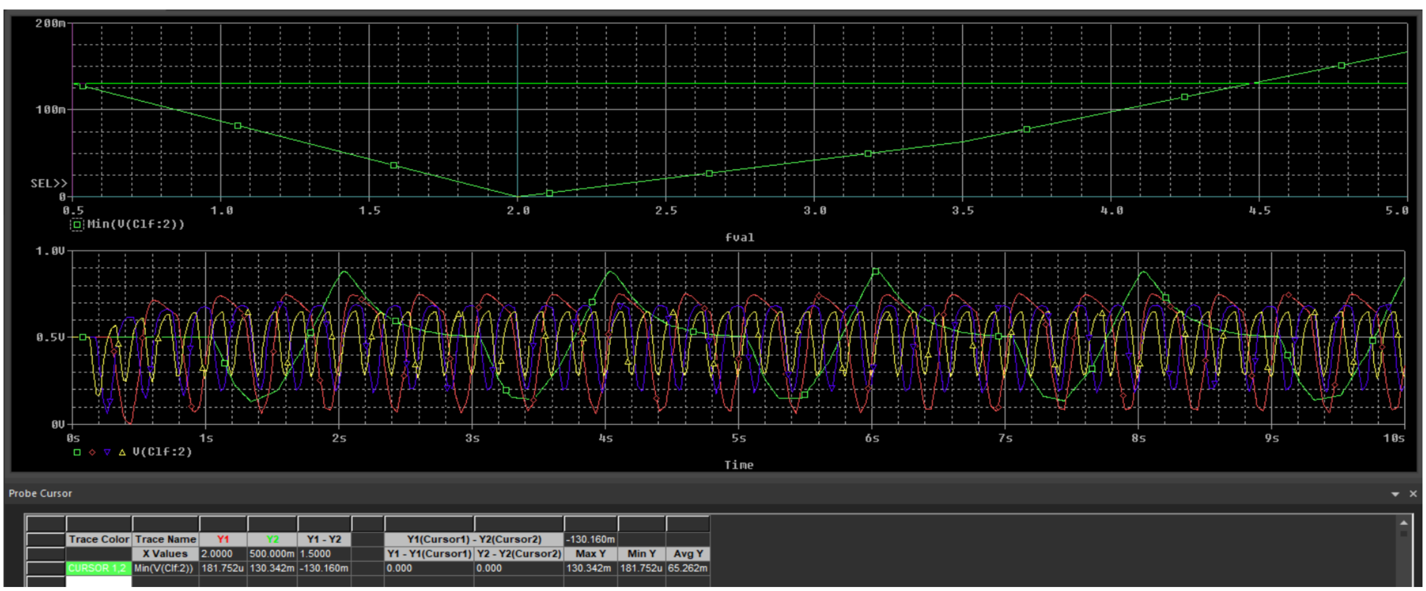


Figure 1.27

The minimum value of the voltage signal after biasing according to Figure 1.27 is 181.752 uV.

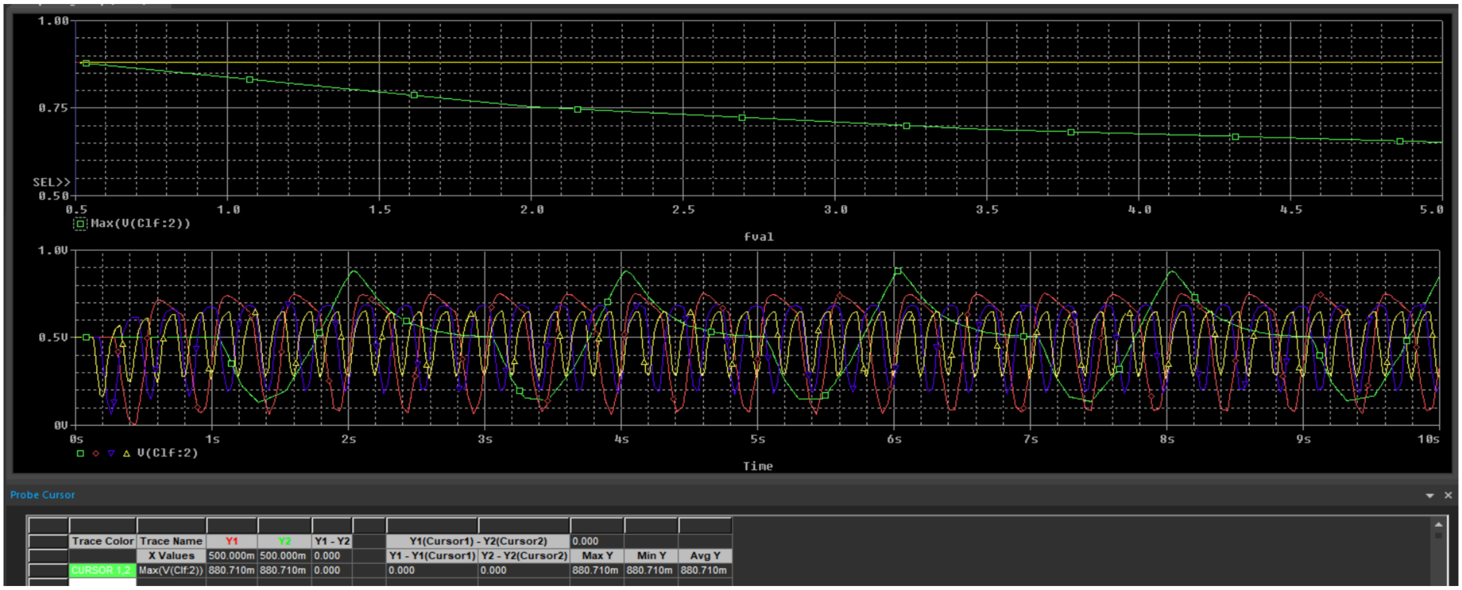


Figure 1.28

The maximum value of the voltage signal after biasing according to Figure 1.28 is 880.710 mV.

2) <u>TIA, BPF, Stage-I Amplifier, voltage shifter and Stage-II Amplifier</u>

The last step in the analog-front-end circuit designing is integration of all the steps. In this step, another operational amplifier is added. The schematic is shown in the Figure 1.29. The output of simulation result of this sub-circuit is shown in Figure 1.30. The maximum value and minimum value of the signal generated is analysed in Figure 1.31 and Figure 1.32 respectively.

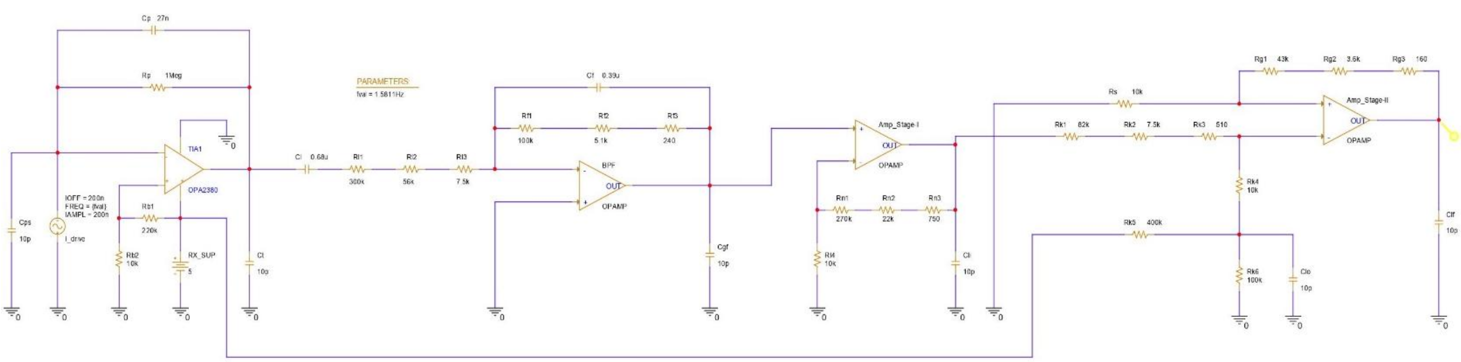


Figure 1.29

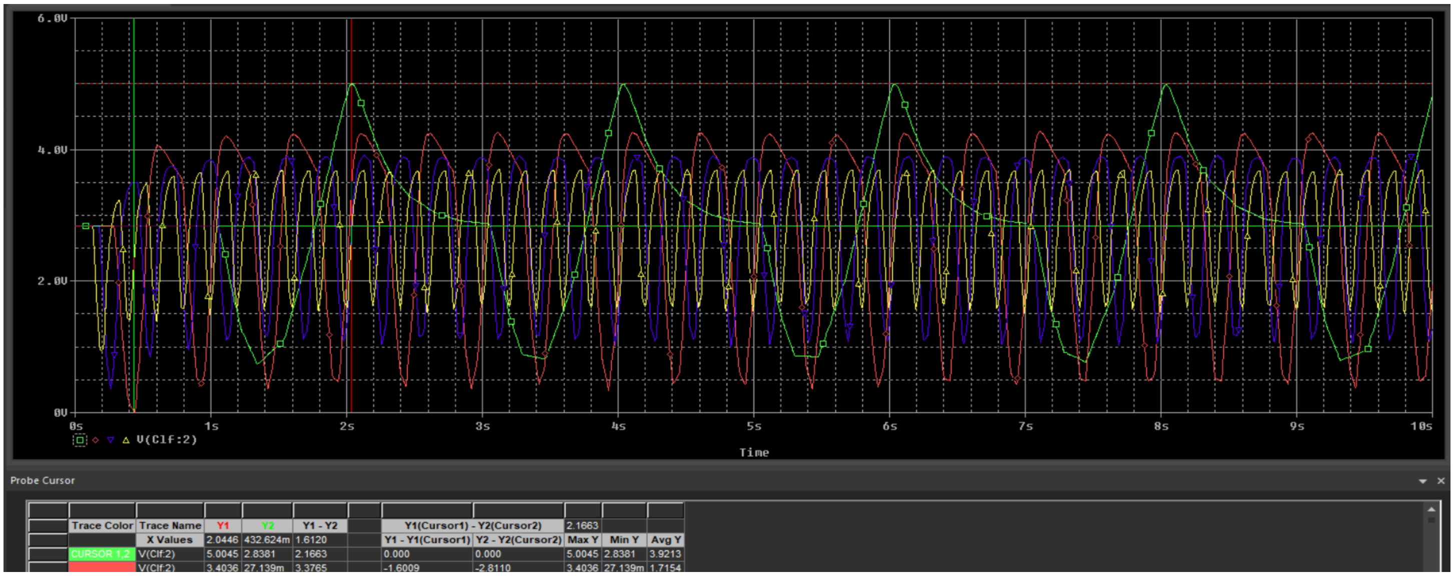


Figure 1.30

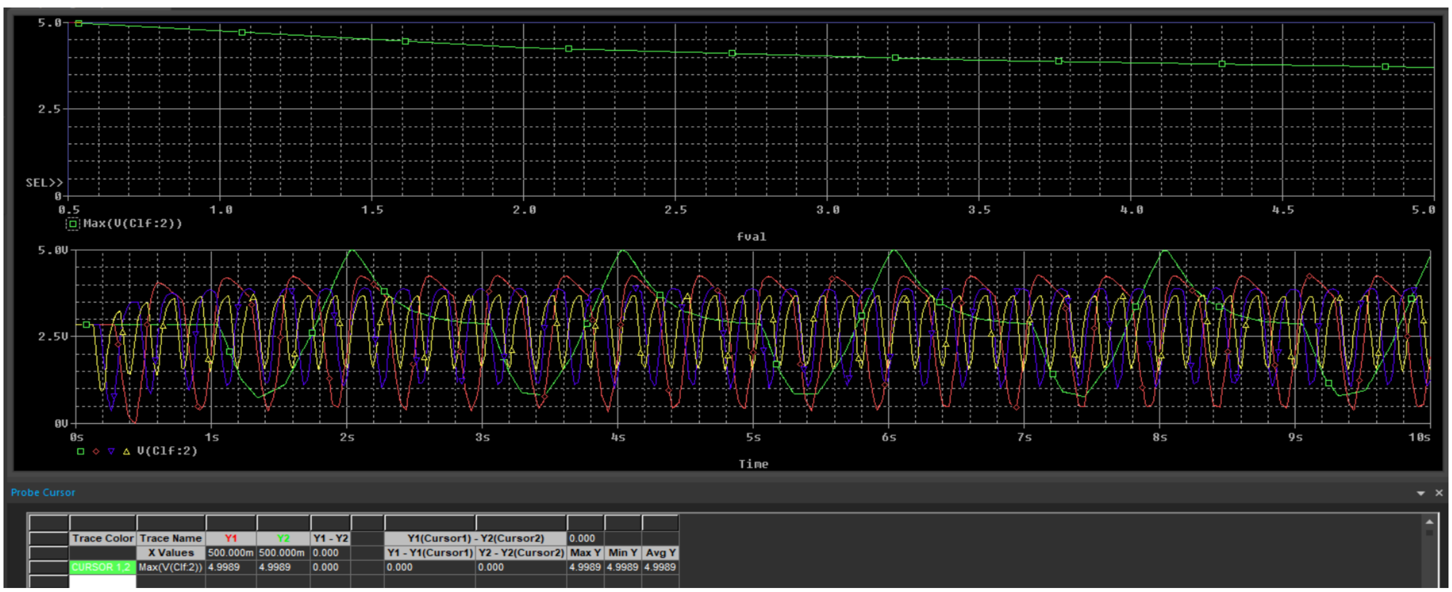


Figure 1.31

The maximum value of VOUT according to the Figure 1.32 is 4.9999 V ~ 5 V.

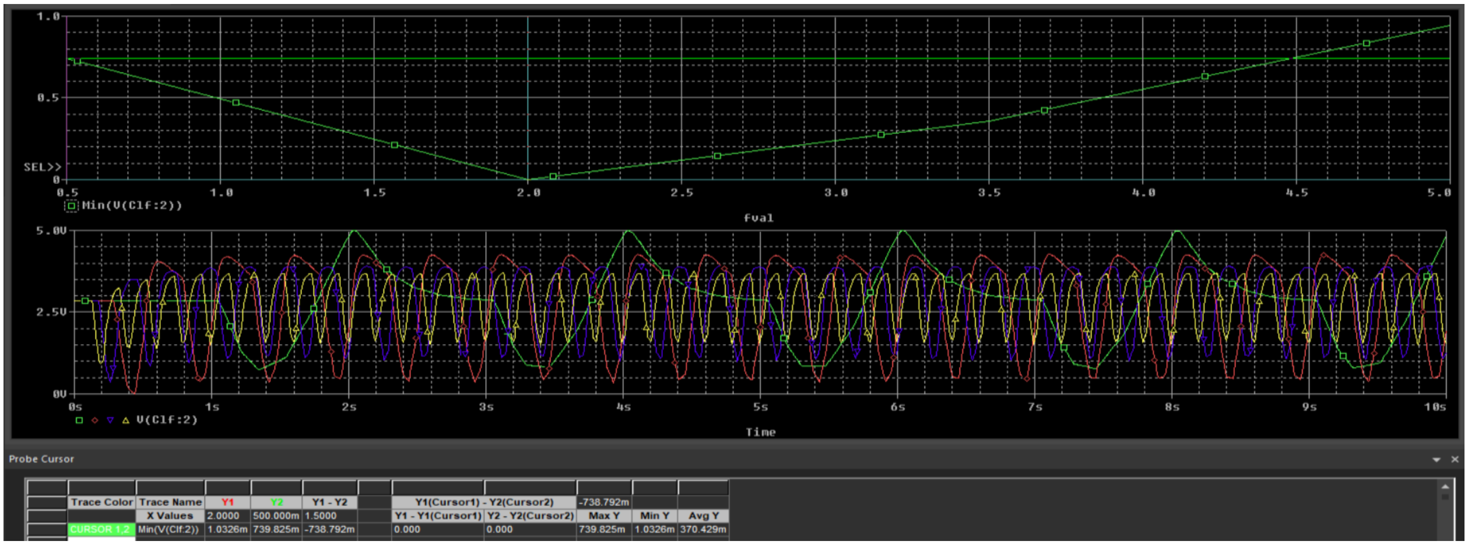


Figure 1.32

The minimum value of VOUT according to the Figure 1.32 is 1.0326 mV.

3.5 <u>Integrated PSPICE Simulation of complete schematic</u>

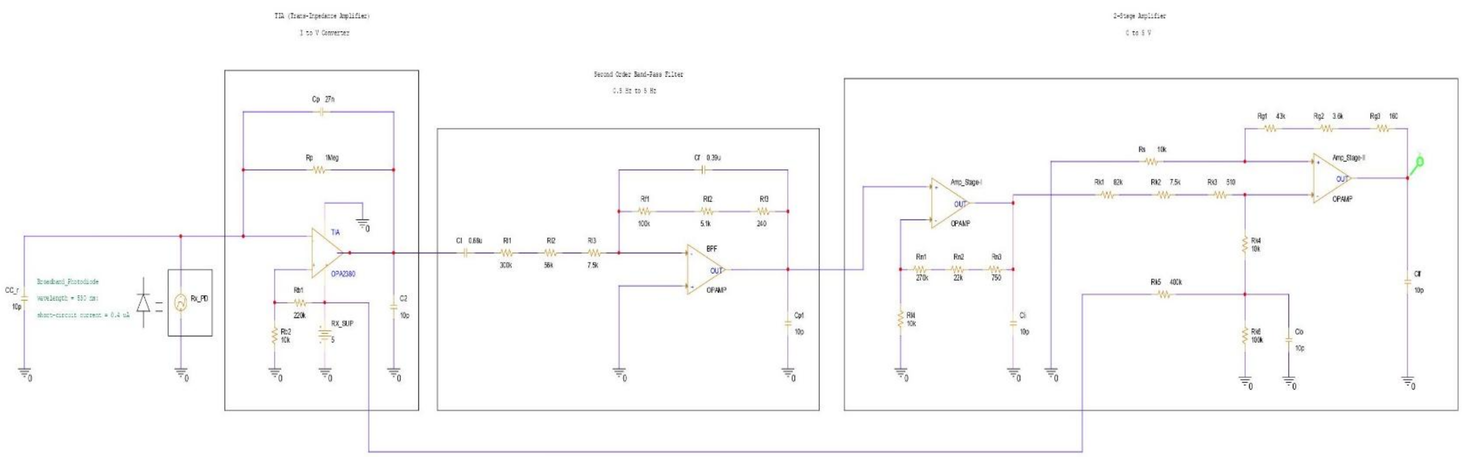


Figure 1.33

In Figure 1.33, the overall schematic design of the analog-front-end circuit for the PPG signal is shown. The weak PPG signal from photodiode current in the order of few hundreds of nA is first amplified in the order of 10^6 to generate voltage signal in the order of mV. The voltage signal is then filtered using a second order band-pass filter to cancel out noises of unwanted frequencies. The voltage signal is finally fed to non-inverting operational amplifiers in two stages so that the final output voltage signal lies within a range compatible for the Arduino micro-controller to process the signal.

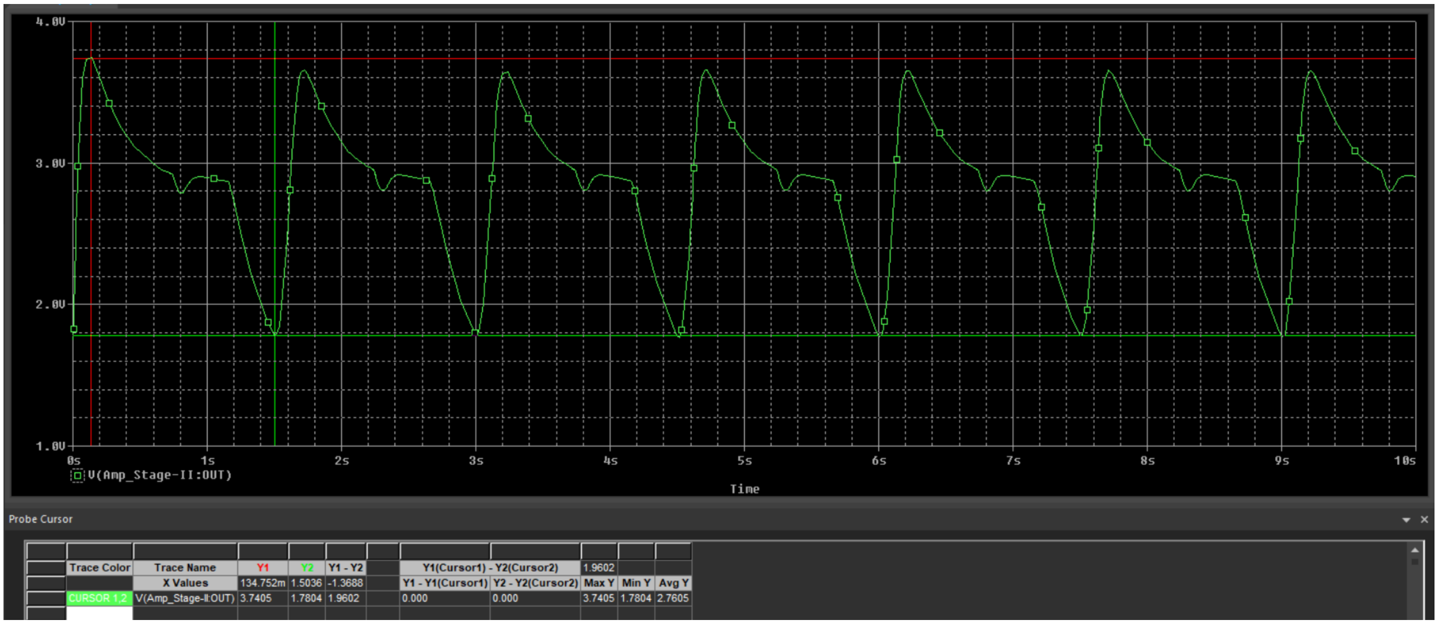


Figure 1.34

As we can see from the output waveform of the synthetic signal generated previously (in Figure 1.4) is obtained above in the Figure 1.34. The output voltage signal of the Signal Conditioning Circuit is compatible for processing and can be given as analog input signal to any of the analog input pin of Arduino micro-controller Board.

3.6 Real-World PPG Signals testing using MATLAB

In this section, we are going to deal with the Real-World PPG Signals. We are going to give the PPG-Signal as the input with AWGN to find the output.

The Steps involved in the analysis:
Step 1: Real-World PPG signals are collected in CSV format.
Step 2: The CSV files are imported into MATLAB workspace.
Step 3: Using MATLAB library, we plot the waveform using the data (Figure 1.35, 1.56, & 1.77) and added AWGN at different SNR values: 5db (Figure 1.38, 1.59, & 1.80), 10db (Figure 1.41, 1.62, & 1.83), 15db (Figure 1.44, 1.65, & 1.86), 20db (Figure 1.47, 1.68, & 1.89), 25db (Figure 1.50, 1.71, & 1.92), 30db (Figure 1.53, 1.74, & 1.95).
Step 4: That sampled values of the resulting waveforms are exported back to CSV format in tabular form.
Step 5: Using the PWL modelling source in PSPICE, we generated the waveforms in PSPICE (Figure 1.36, 1.39, 1.42, 1.45, 1.48, 1.51, 1.54, 1.57, 1.60, 1.63, 1.66, 1.69, 1.72, 1.75, 1.78, 1.81, 1.84, 1.87, 1.90, 1.93, & 1.96) to analyse the performance of band-pass filter designed in our schematic (Figure 1.37, 1.40, 1.43, 1.46, 1.49, 1.52, 1.55, 1.58, 1.61, 1.64, 1.67, 1.70, 1.73, 1.76, 1.79, 1.82, 1.85, 1.88, 1.91, 1.94, & 1.97) .

The MATLAB Code used for adding the AWGN to the PPG signals and for exporting the sampled values back to CSV format:

```
clc;

% Original Signal figure (1);
plot (time, val);
xlabel('Time');
legend ('Original Signal');
PPG = [time val];
% AWGN at SNR = 5 db n_5db = awgn (val, 5, 'measured');

figure (2);
plot (time, val, time, n_5db);
xlabel('Time');
legend ('Original Signal','Noisy Signal');
PPG_5 = [time n_5db];
csvwrite ('PPG_5.csv', PPG_5);
% AWGN at SNR = 10 db n_10db = awgn (val, 10, 'measured');

figure (3);
plot (time, val, time, n_10db);
xlabel('Time');
legend ('Original Signal','Noisy Signal');
PPG_10 = [time n_10db];
csvwrite ('PPG_10.csv', PPG_10);
% AWGN at SNR = 15 db n_15db = awgn (val, 15, 'measured');

figure (4);
plot (time, val, time, n_15db);
xlabel('Time');
legend ('Original Signal','Noisy Signal');
PPG_15 = [time n_15db];
csvwrite ('PPG_15.csv', PPG_15);
% AWGN at SNR = 20 db n_20db = awgn (val, 20, 'measured');

figure (5);
 plot (time, val, time, n_20db);
xlabel('Time');
legend ('Original Signal','Noisy Signal');
PPG_20 = [time n_20db]; csvwrite ('PPG_20.csv', PPG_20);
% AWGN at SNR = 25 db n_25db = awgn (val, 25, 'measured');

Figure (6);
plot (time, val, time, n_25db);
```

```
xlabel('Time');
legend ('Original Signal','Noisy Signal');
PPG_25 = [time n_25db];
csvwrite ('PPG_25.csv', PPG_25);
% AWGN at SNR = 30 db n_30db = awgn (val, 30, 'measured');

figure (7);
plot (time, val, time, n_30db);
xlabel('Time');
legend ('Original Signal','Noisy Signal');
PPG_30 = [time n_30db];
csvwrite ('PPG_30.csv', PPG_30);
```

WAVEFORMS OF REAL WORLD PPG SIGNALS WITH ADDED AWGN AT DIFFERENT SNR VALUES:

Sample Signal 1:

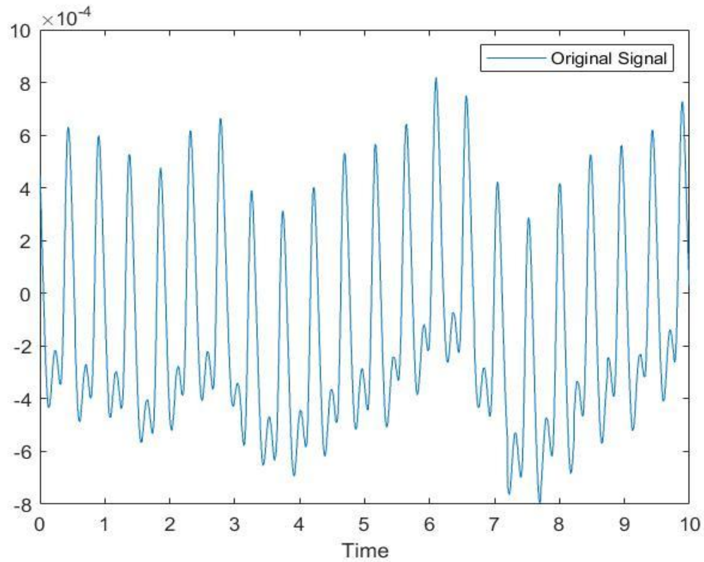


Figure 1.35

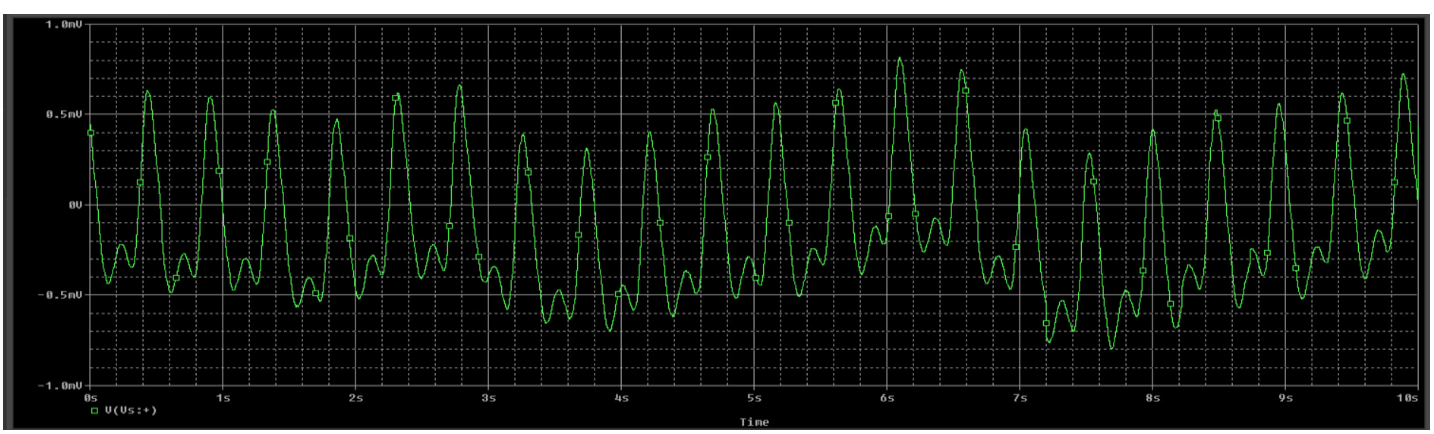


Figure 1.36

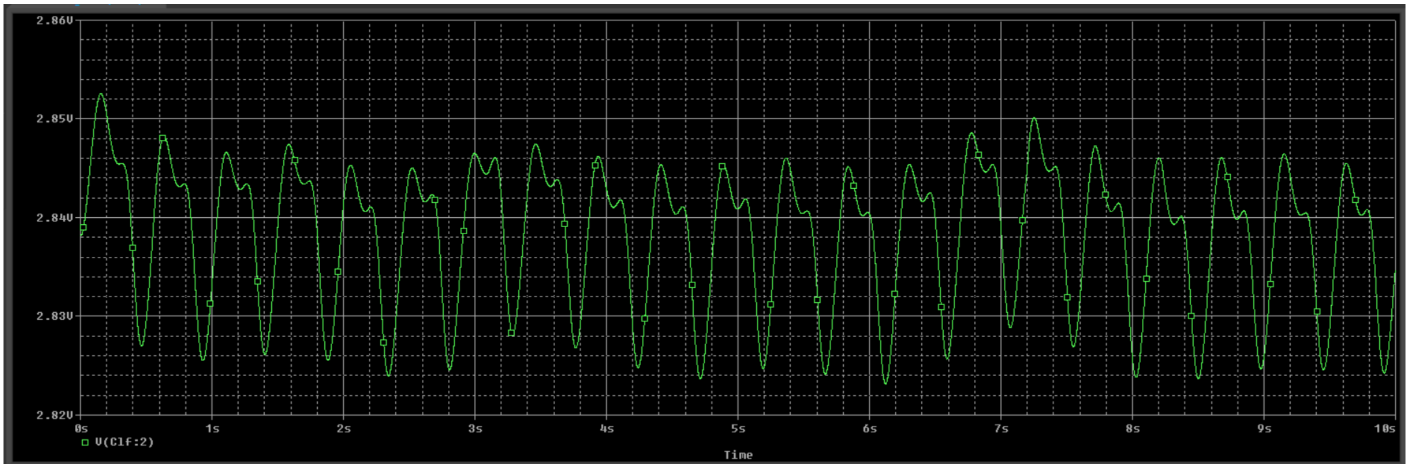


Figure 1.37

Sample Signal 1 at SNR = 5dB:

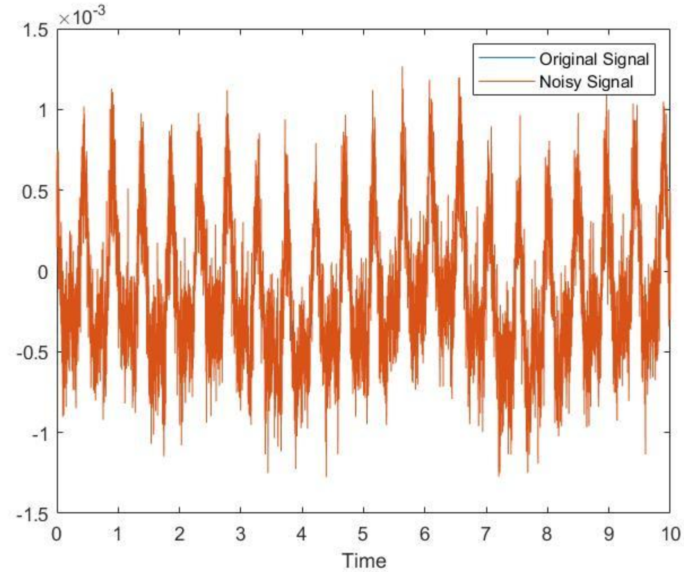


Figure 1.38

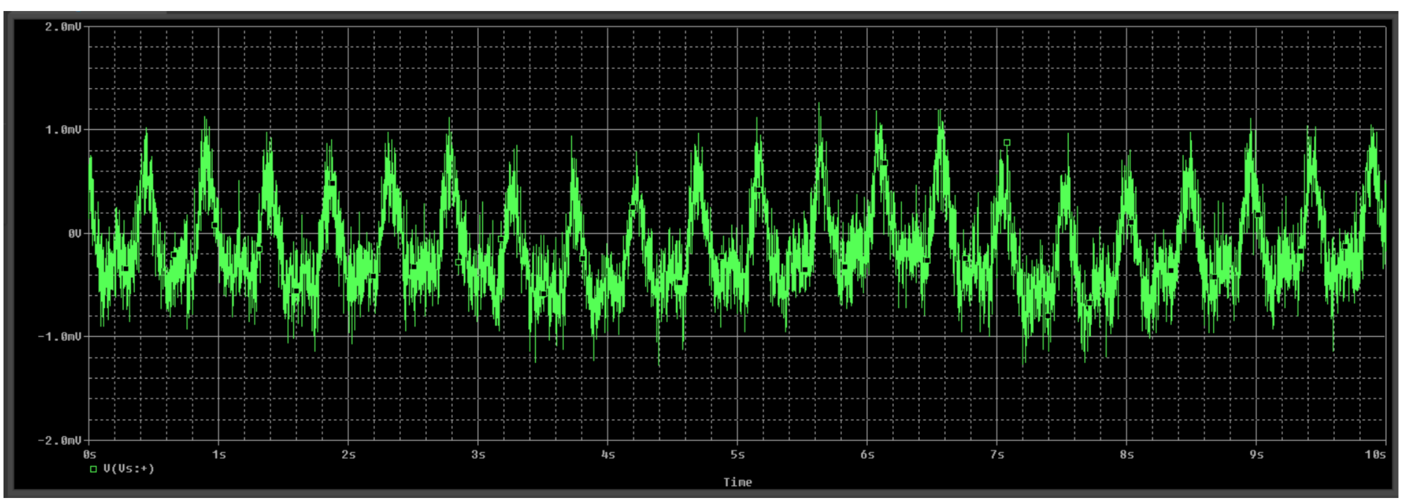


Figure 1.39

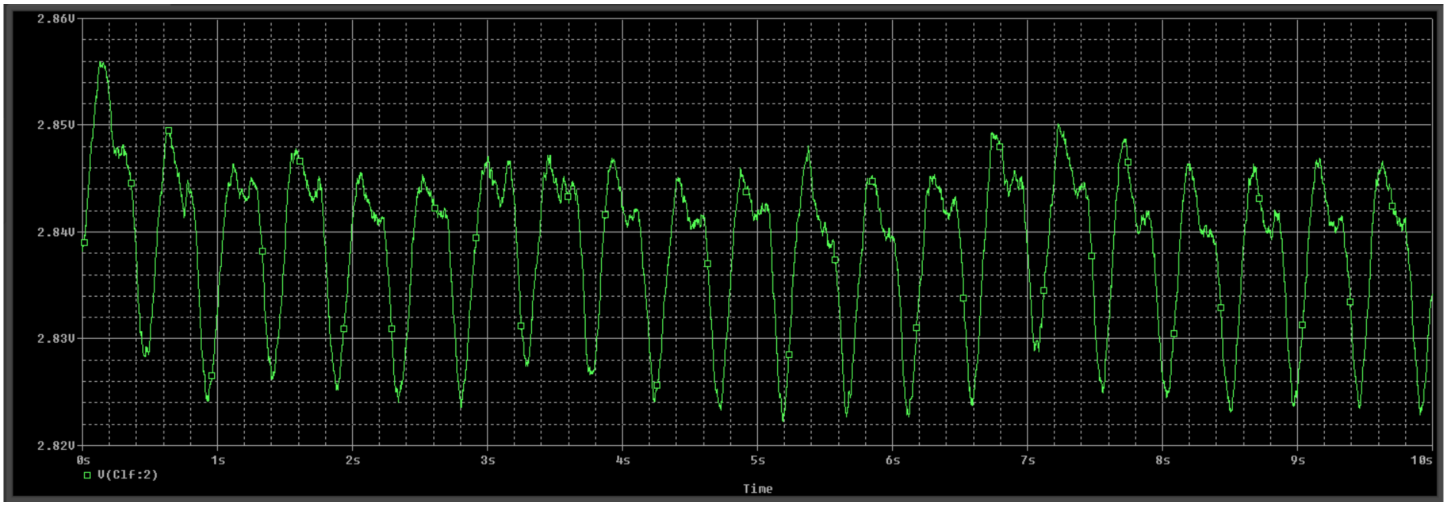


Figure 1.40

<u>Sample signal 1 at SNR = 10dB</u>:

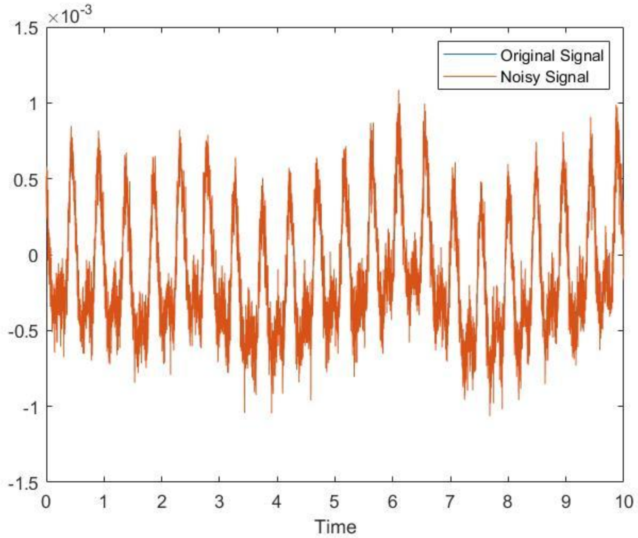


Figure 1.41

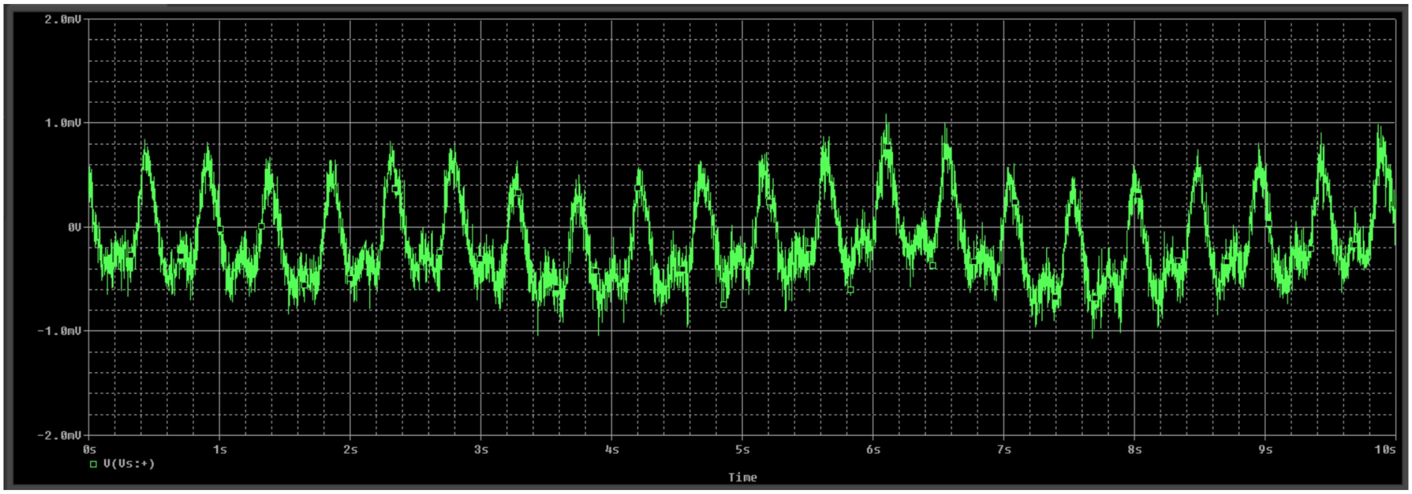


Figure 1.42

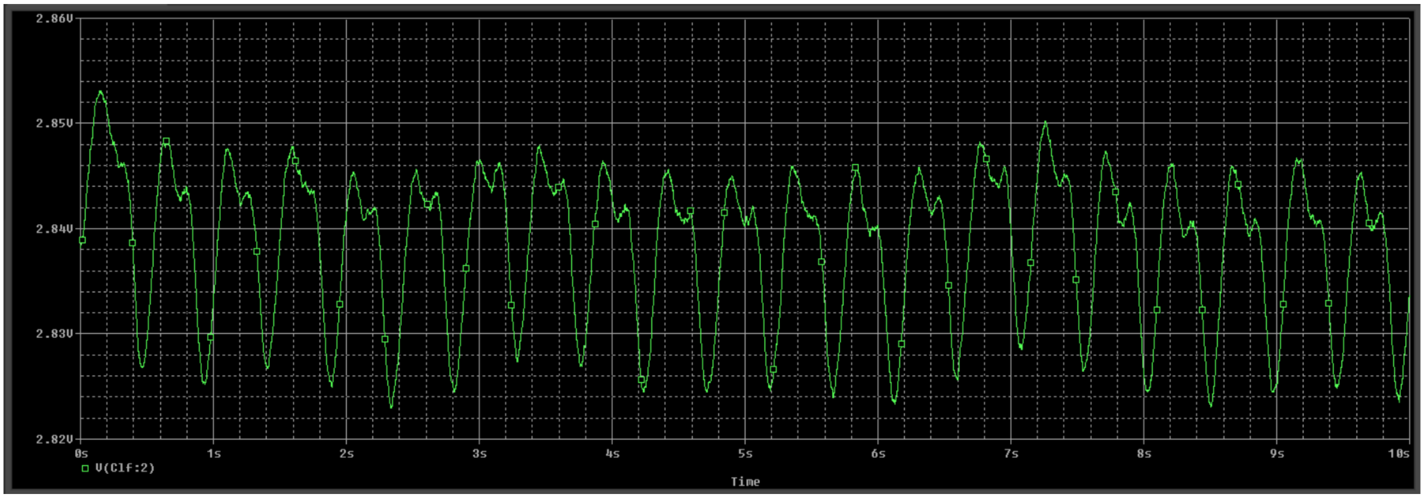


Figure 1.43

Sample signal 1 at SNR = 15dB:

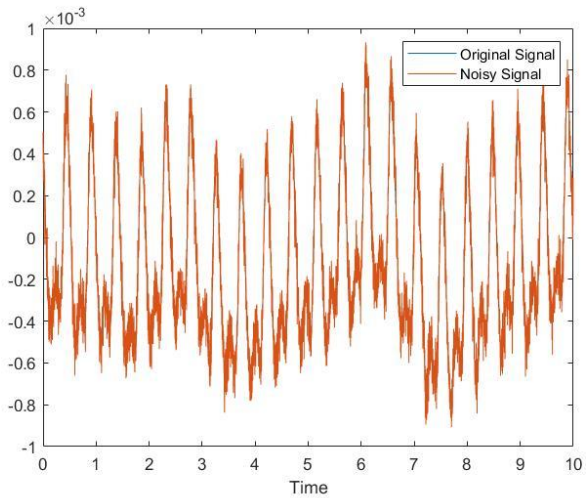


Figure 1.44

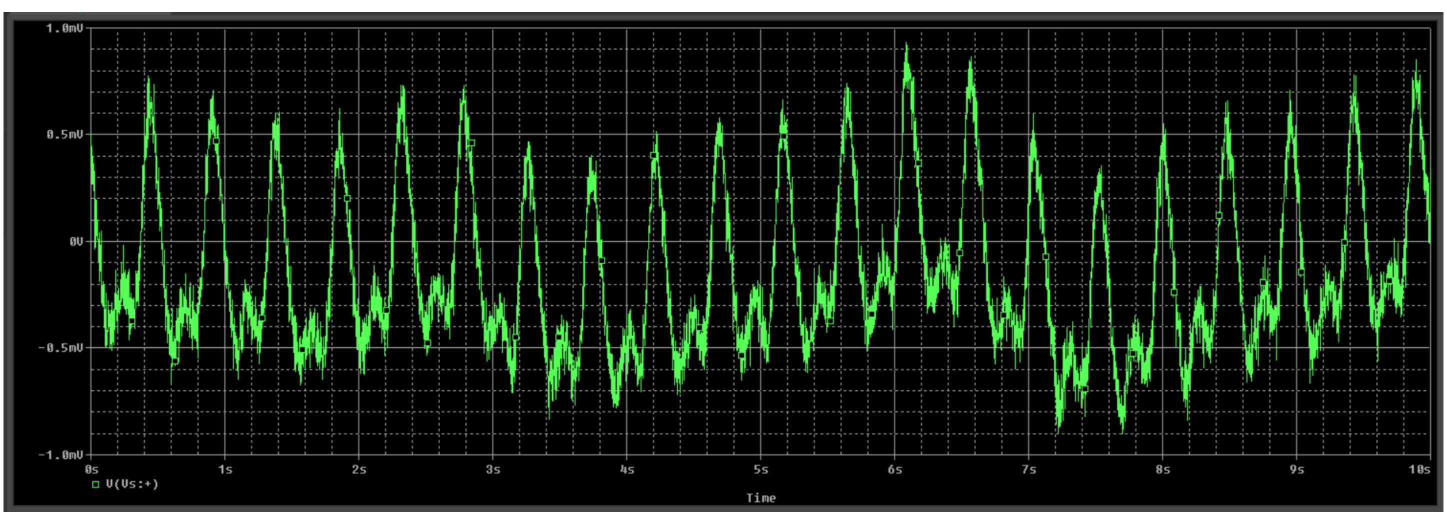


Figure 1.45

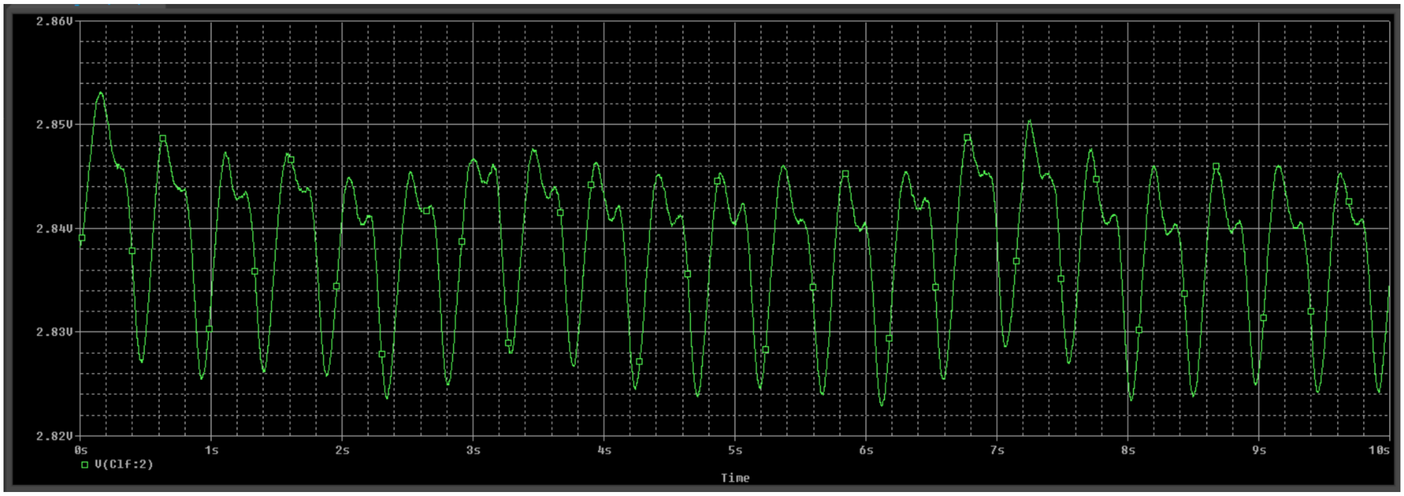


Figure 1.46

<u>Sample signal 1 at SNR = 20dB</u>:

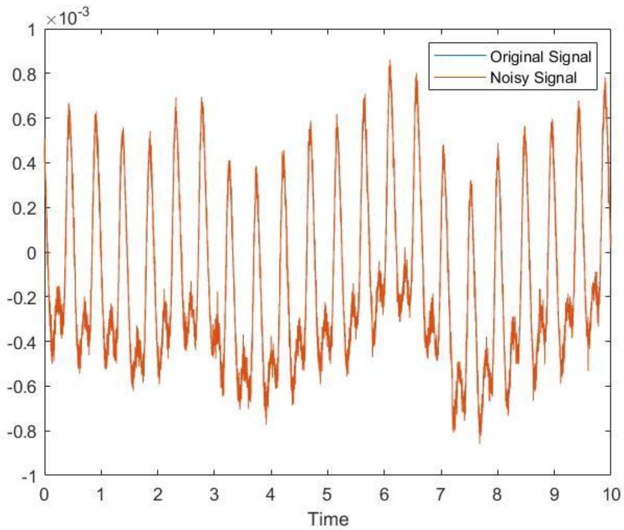


Figure 1.47

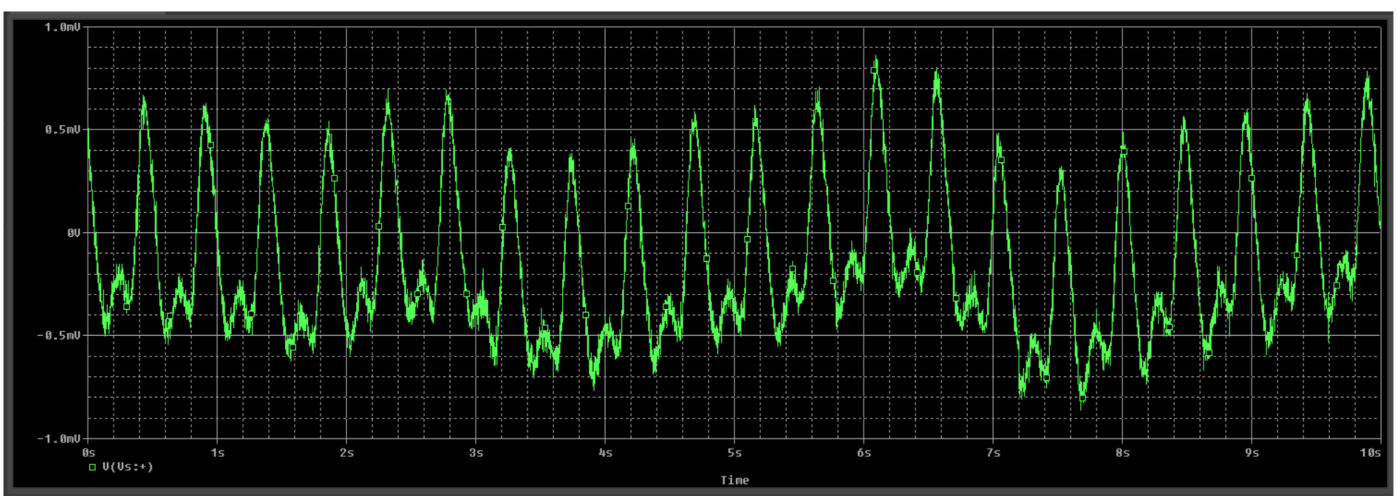


Figure 1.48

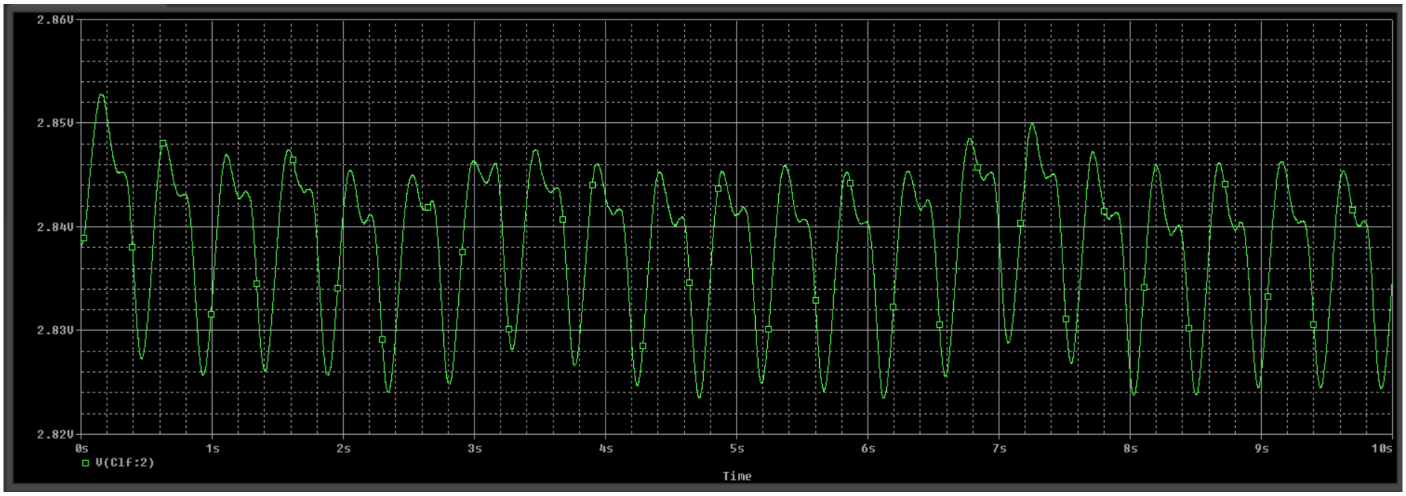


Figure 1.49

Sample signal 1 at SNR = 25dB:

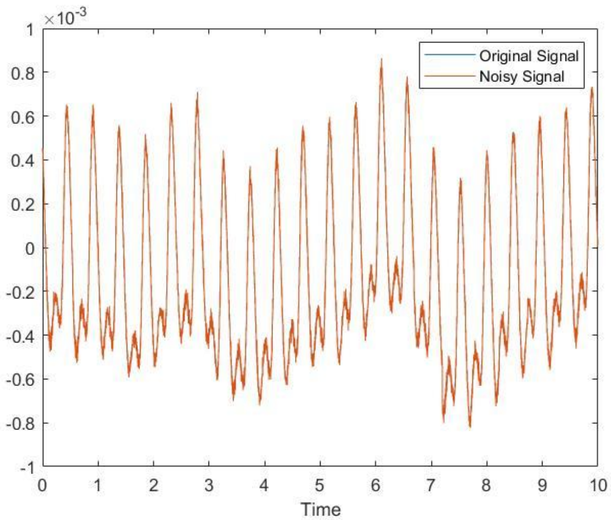


Figure 1.50

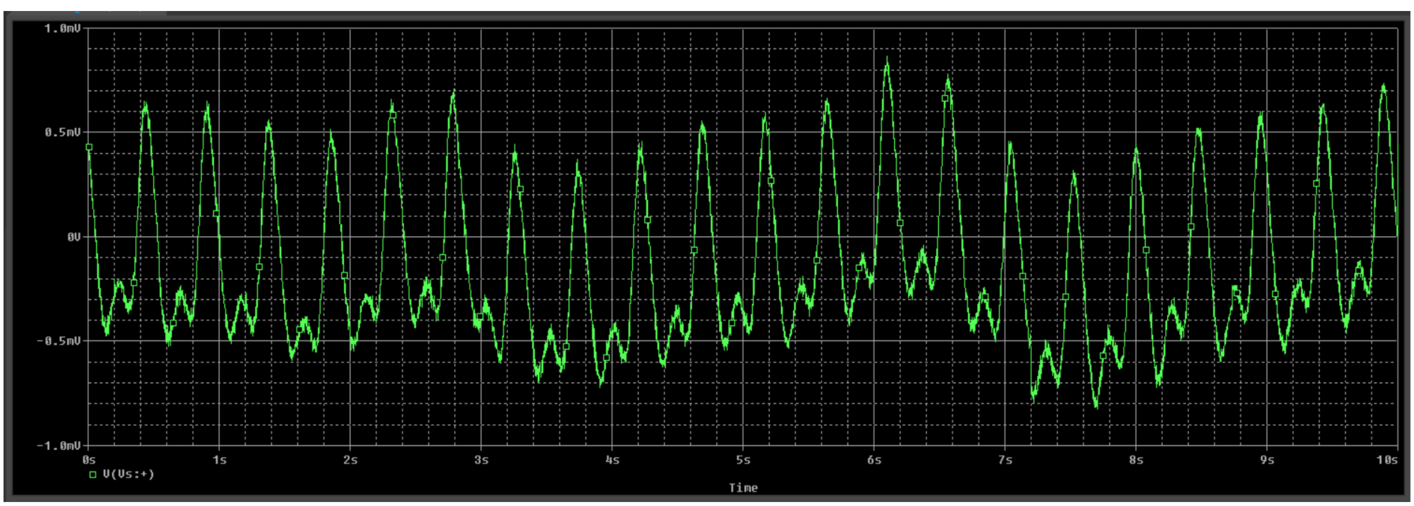


Figure 1.51

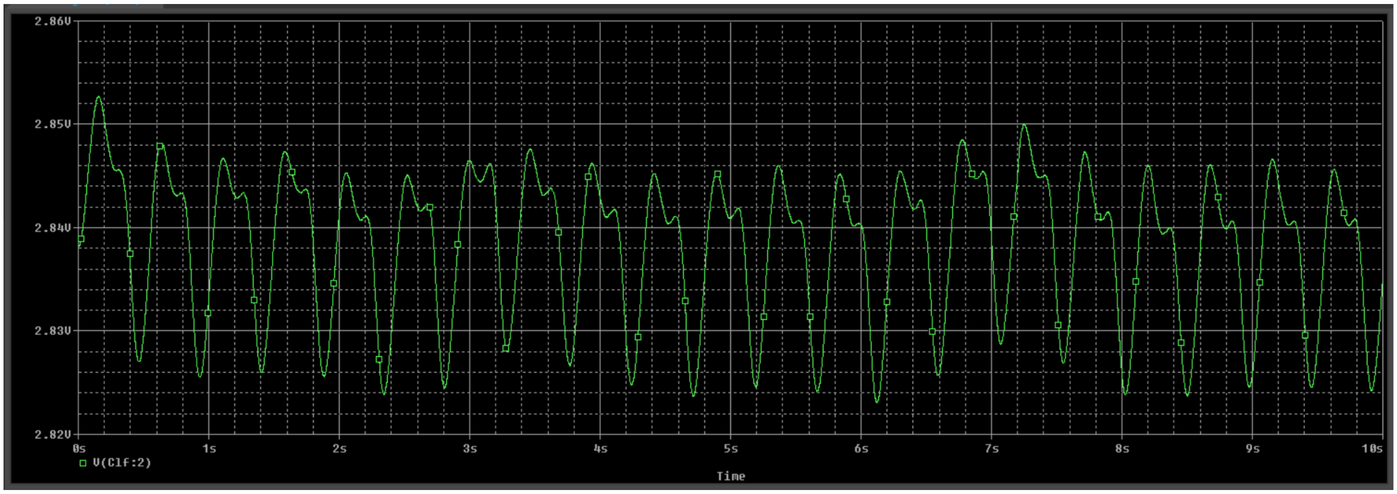


Figure 1.52

<u>Sample signal 1 at SNR = 30dB</u>:

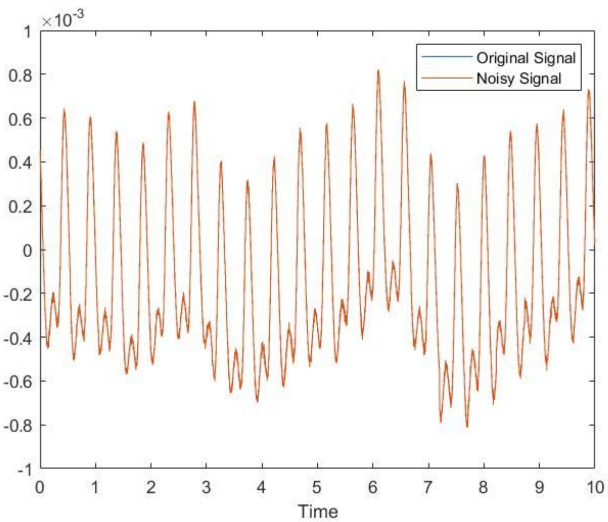


Figure 1.53

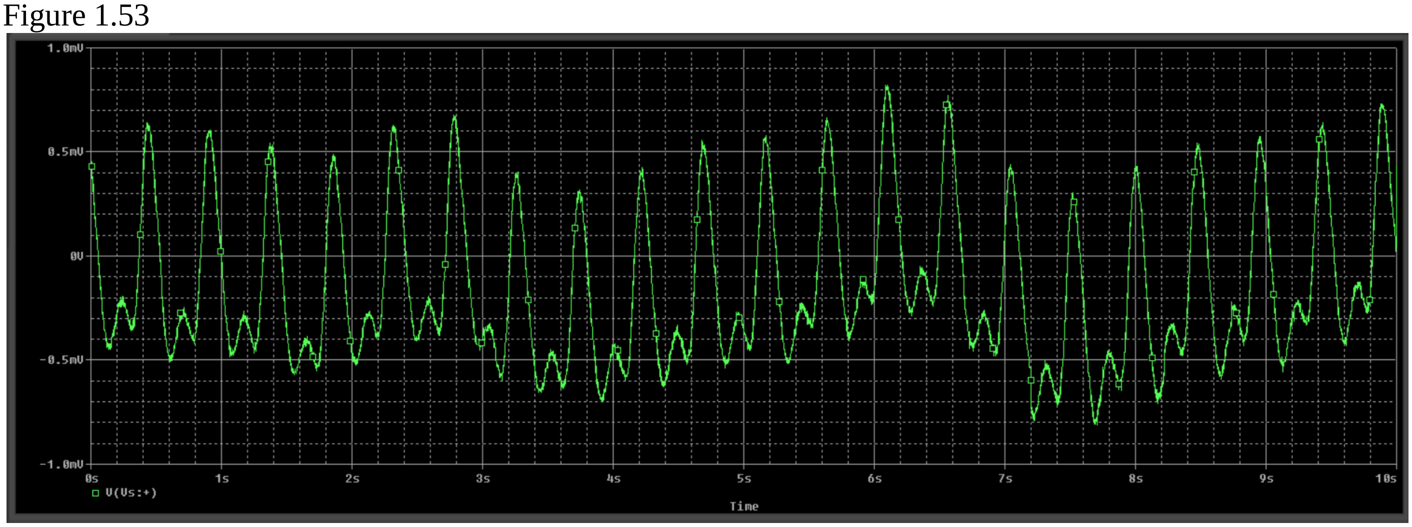


Figure 1.54

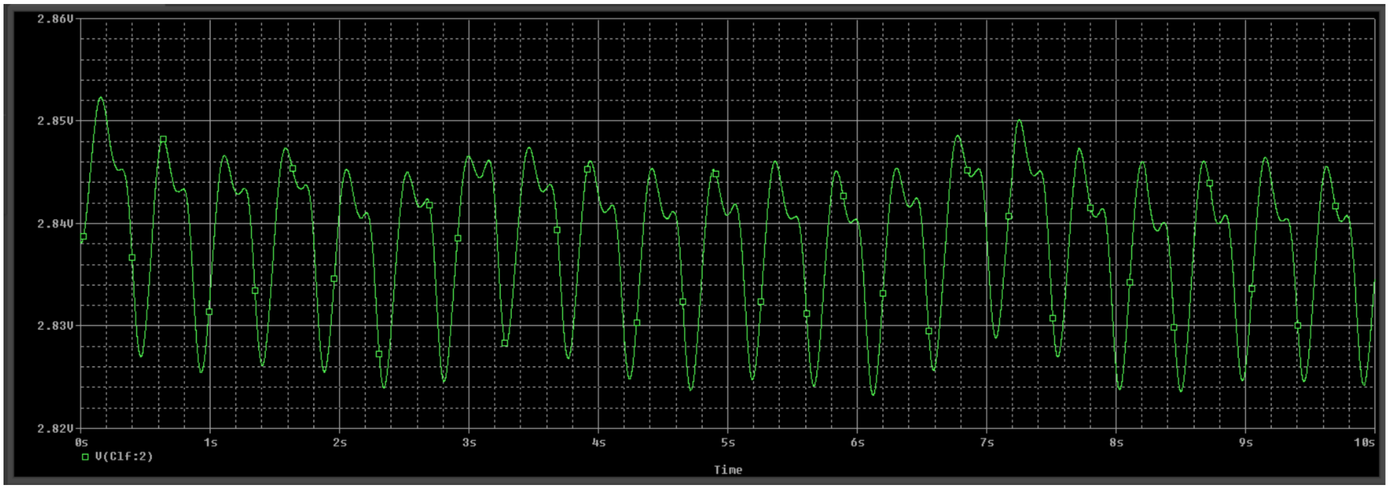


Figure 1.55

<u>Sample signal 2</u>:

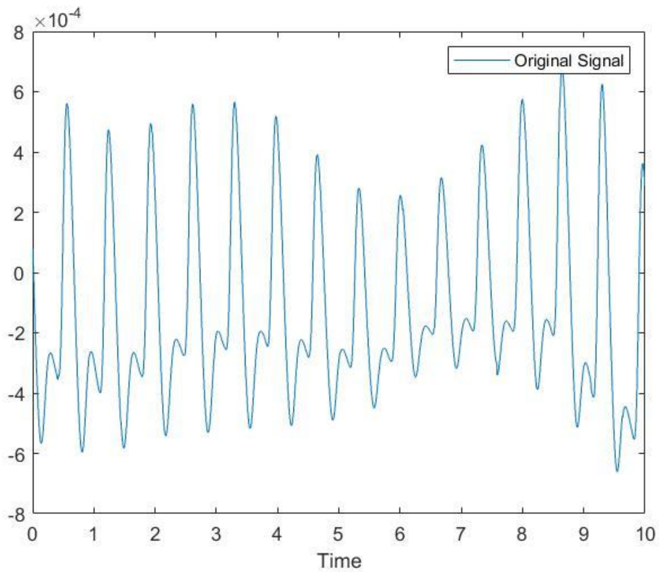


Figure 1.56

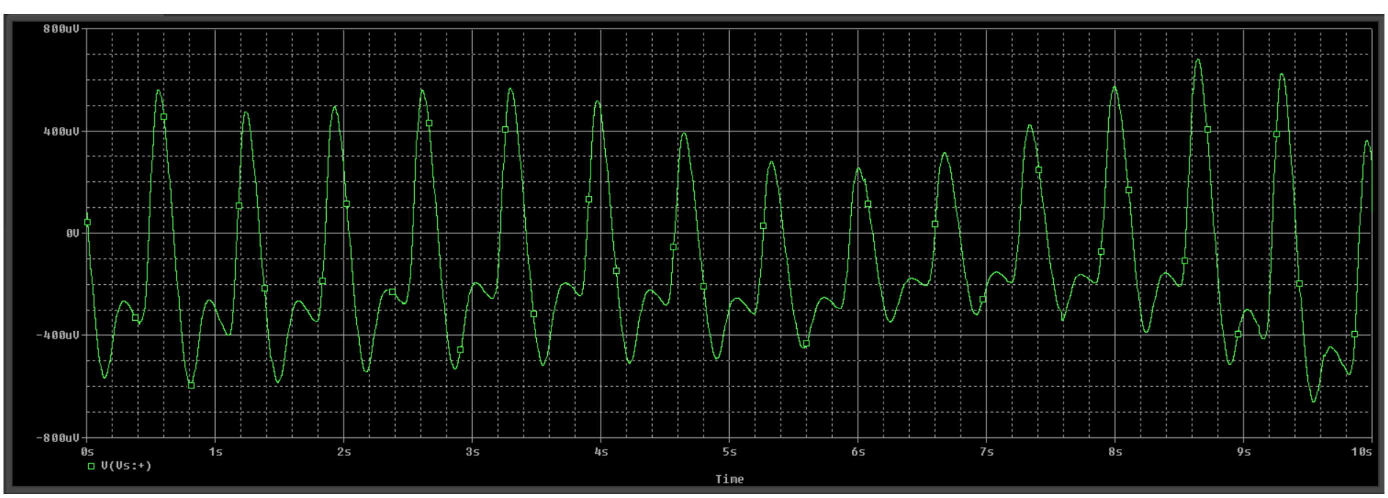


Figure 1.57

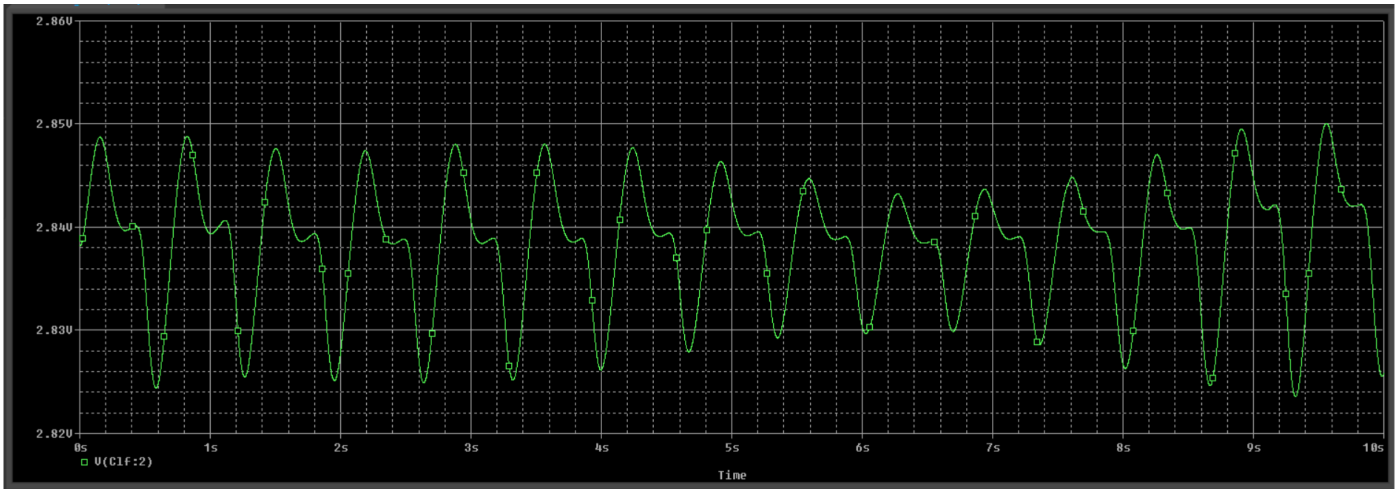


Figure 1.58

<u>Sample signal 2 at SNR = 5dB</u>:

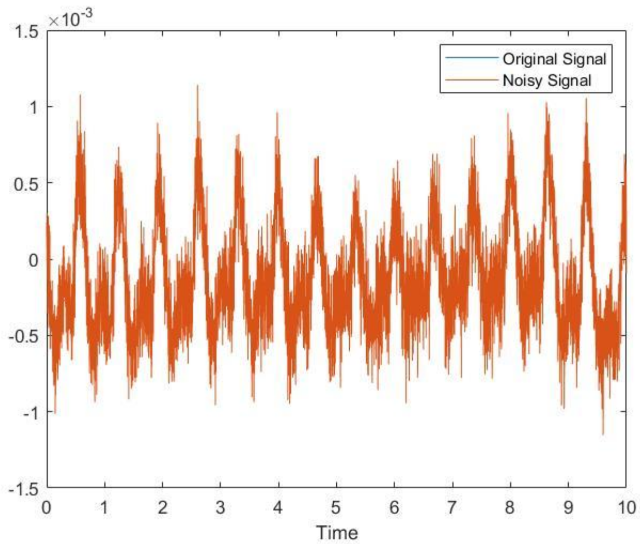


Figure 1.59

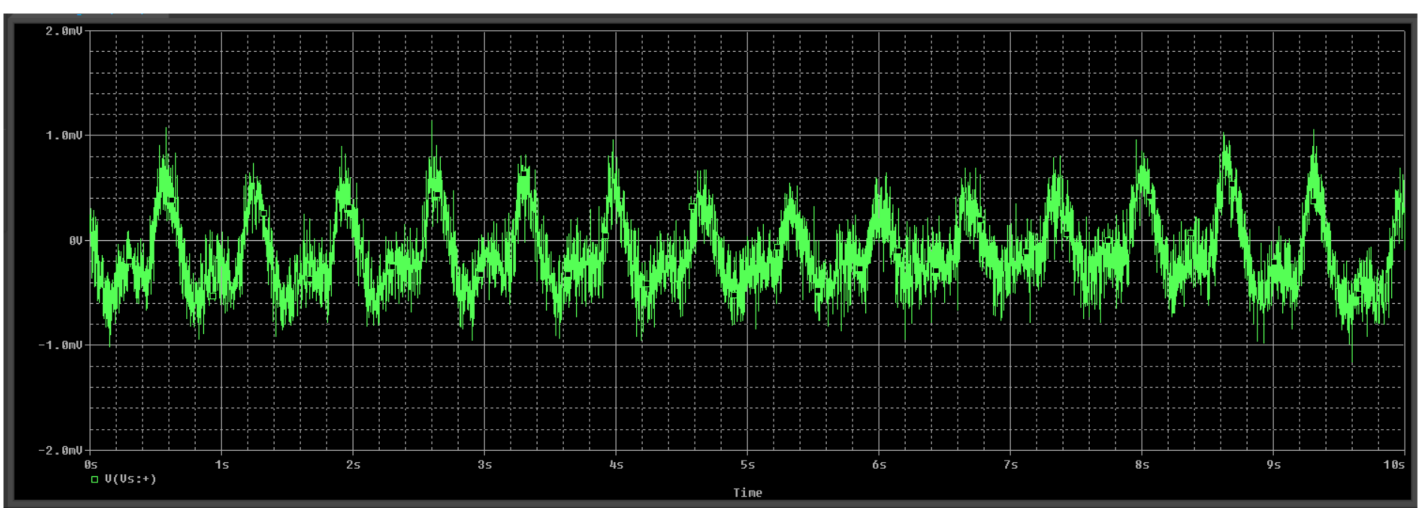


Figure 1.60

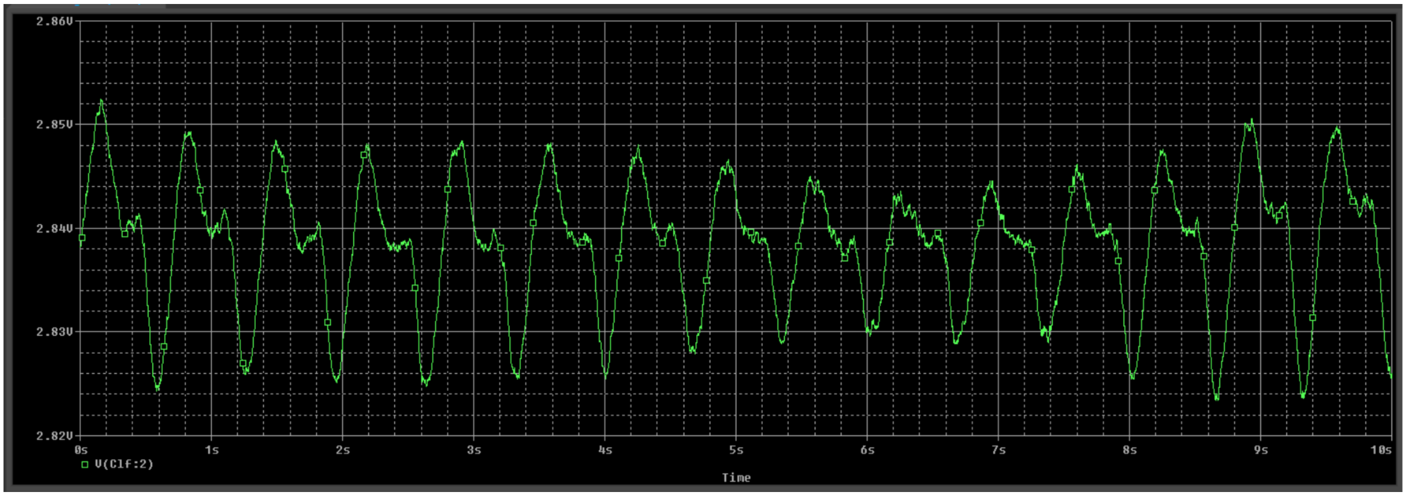


Figure 1.61

Sample signal 2 at SNR = 10dB:

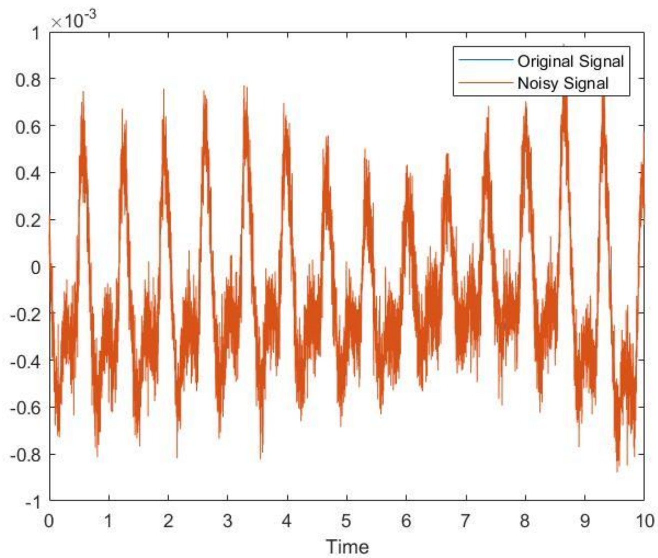


Figure 1.62

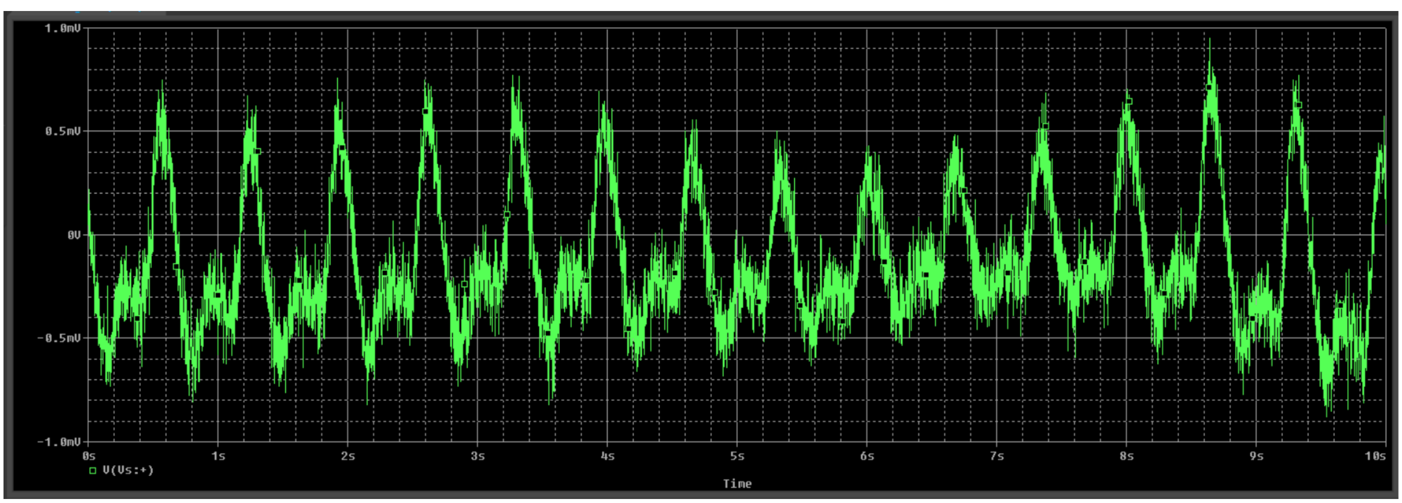


Figure 1.63

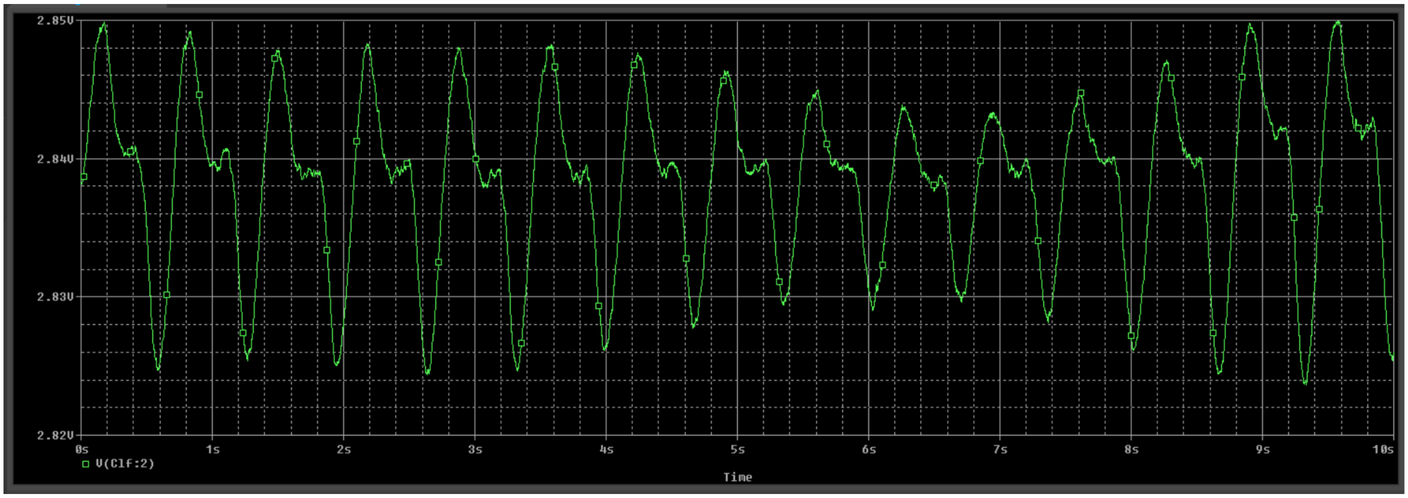


Figure 1.64

<u>Sample signal 2 at SNR = 15dB</u>:

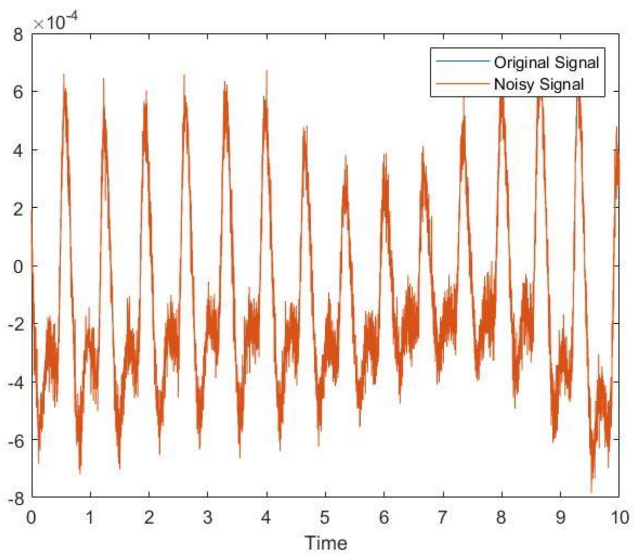


Figure 1.65

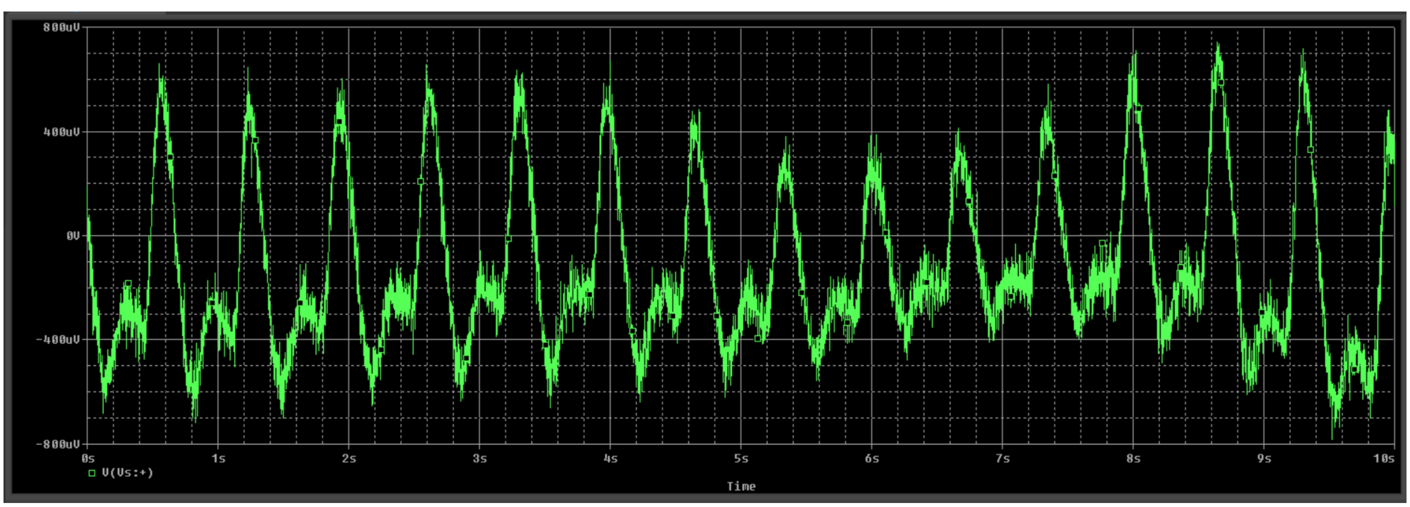


Figure 1.66

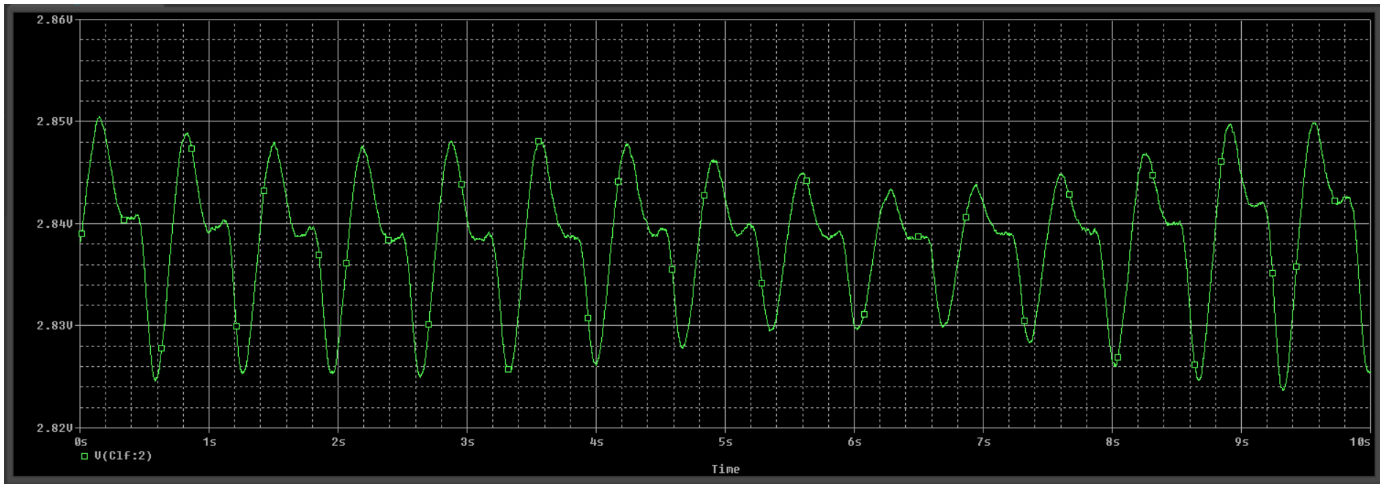


Figure 1.67

<u>Sample signal 2 at SNR = 20dB</u>:

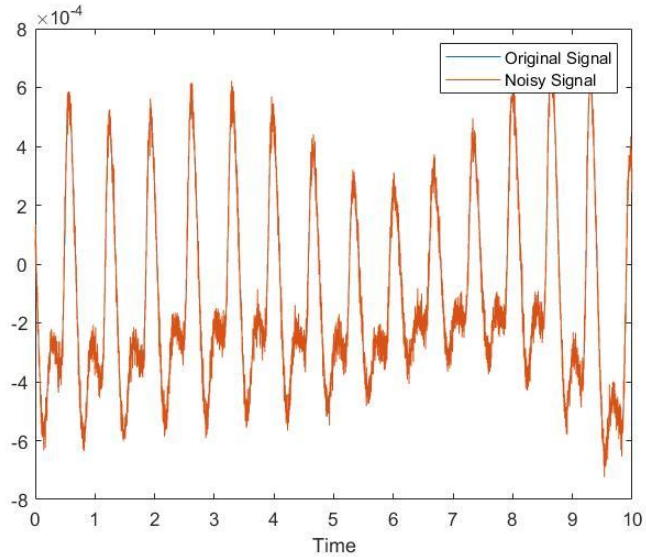


Figure 1.68

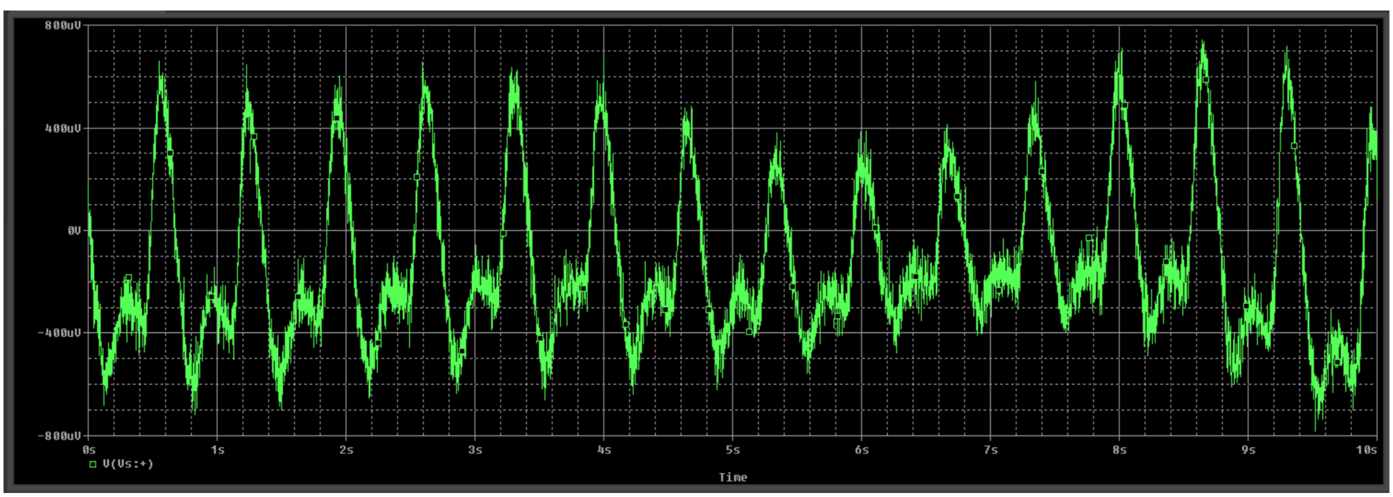


Figure 1.69

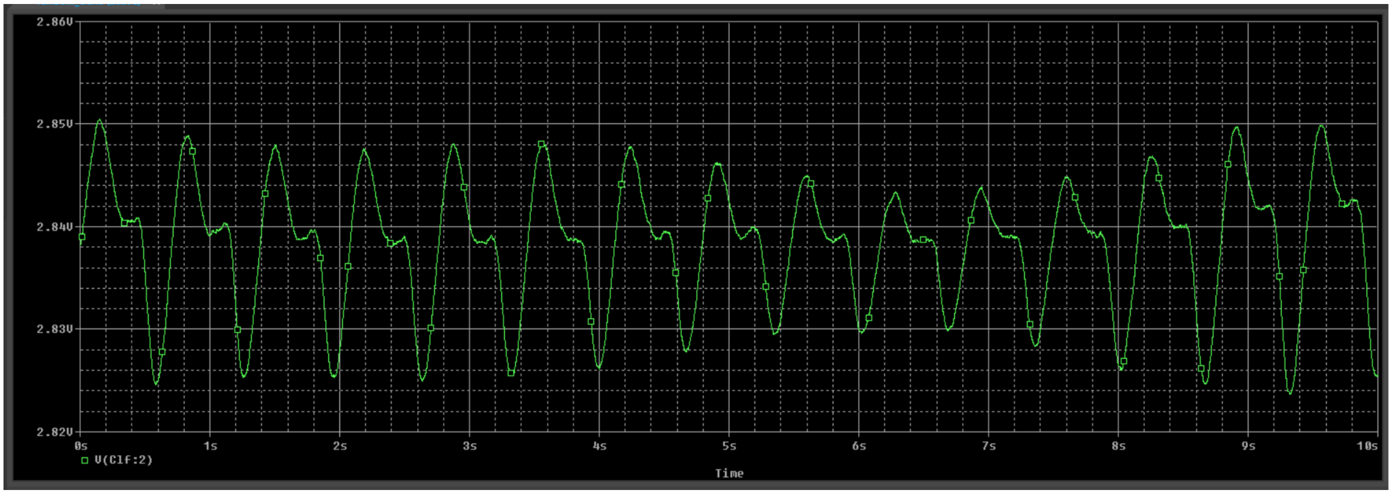


Figure 1.70

Sample signal 2 at SNR = 25dB:

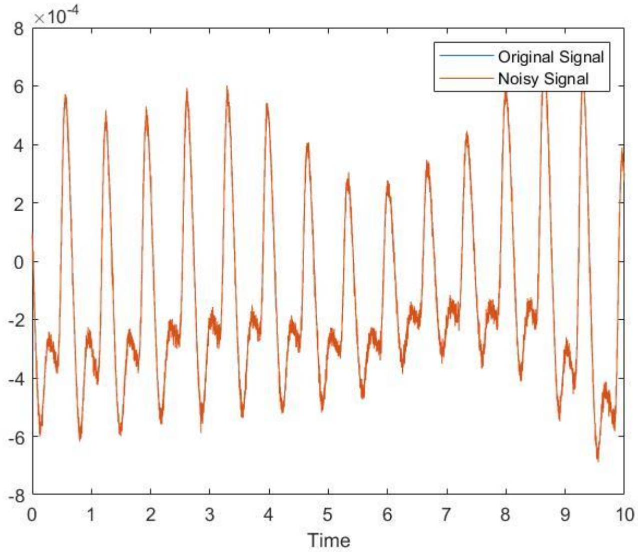


Figure 1.71

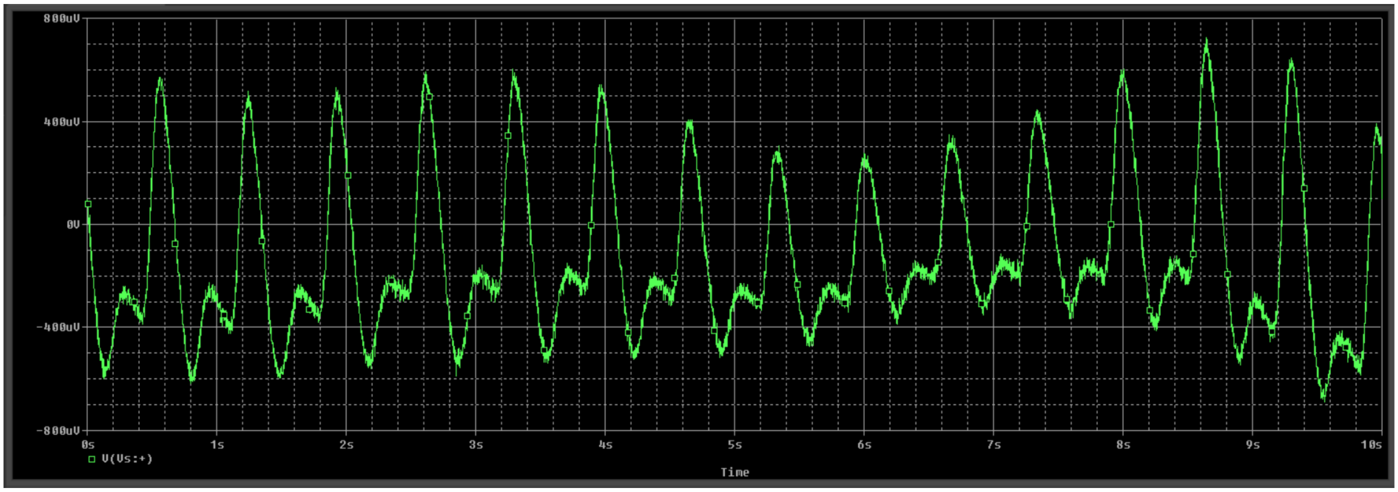


Figure 1.72

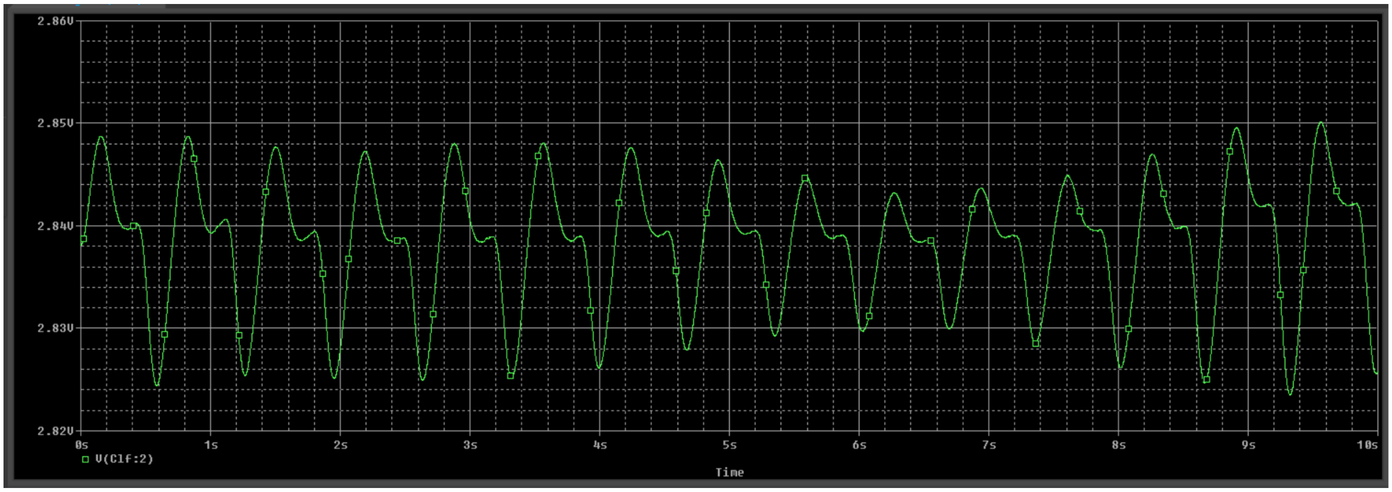


Figure 1.73

<u>Sample signal 2 at SNR = 30dB</u>:

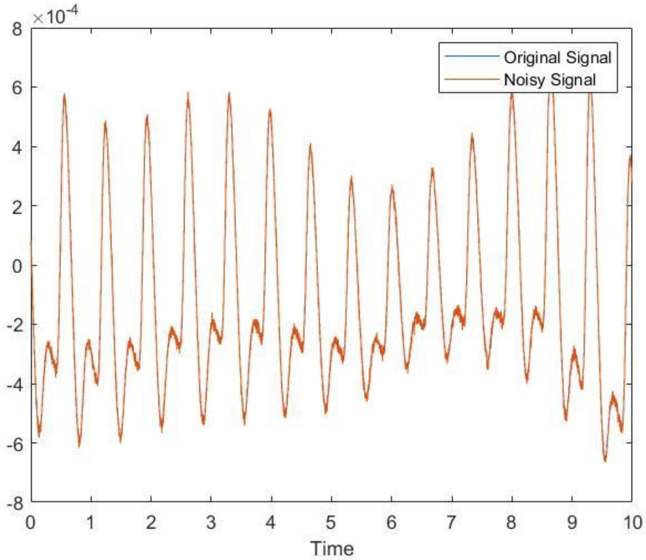


Figure 1.74

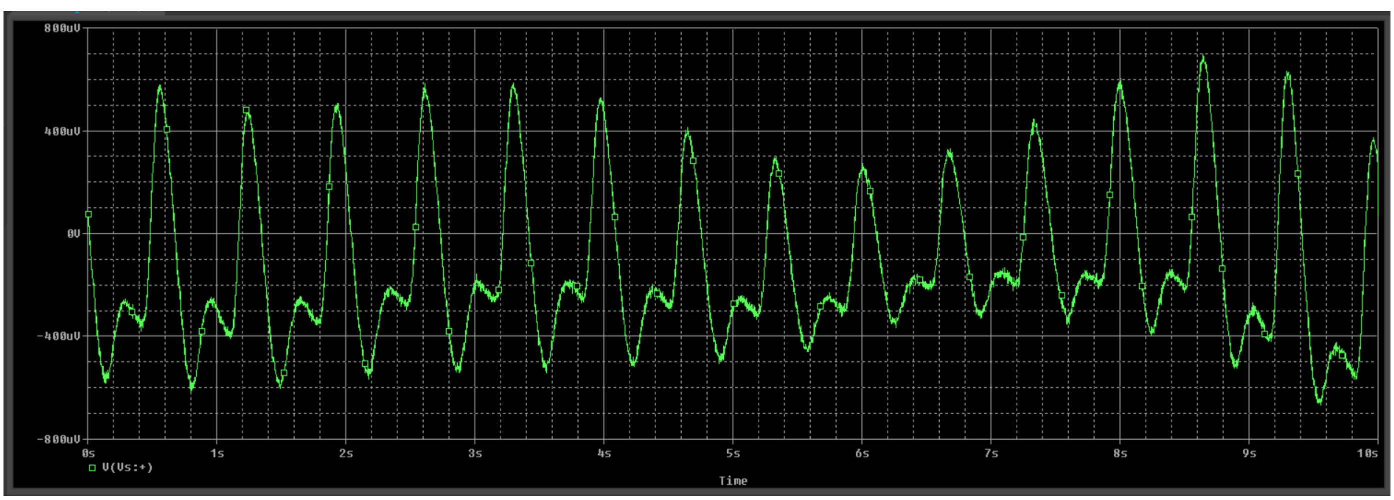


Figure 1.75

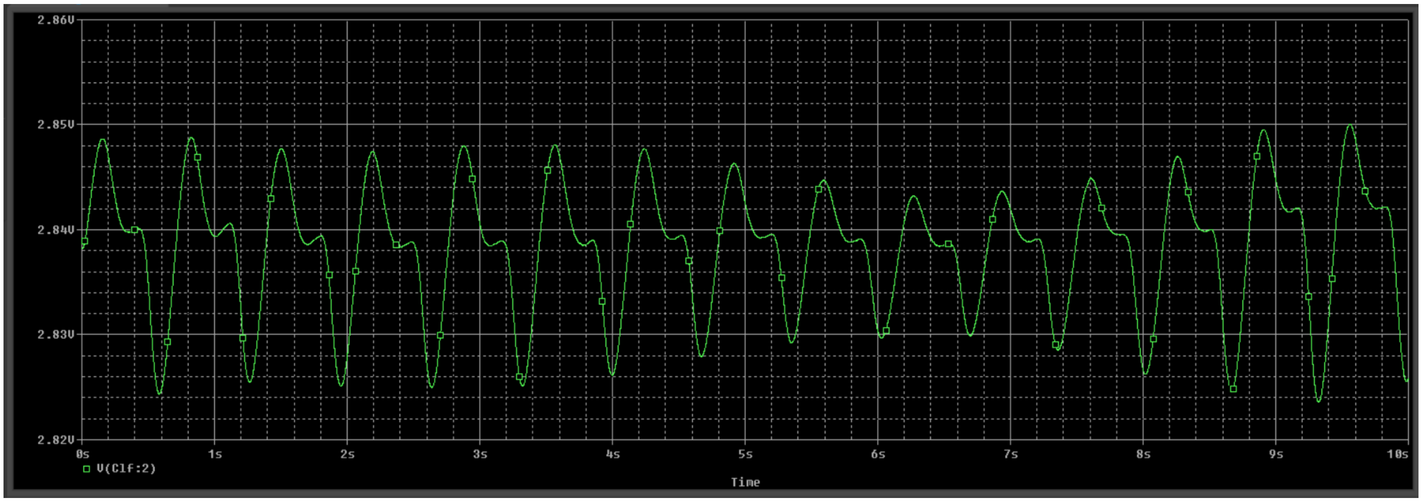


Figure 1.76

<u>Sample signal 3</u>:

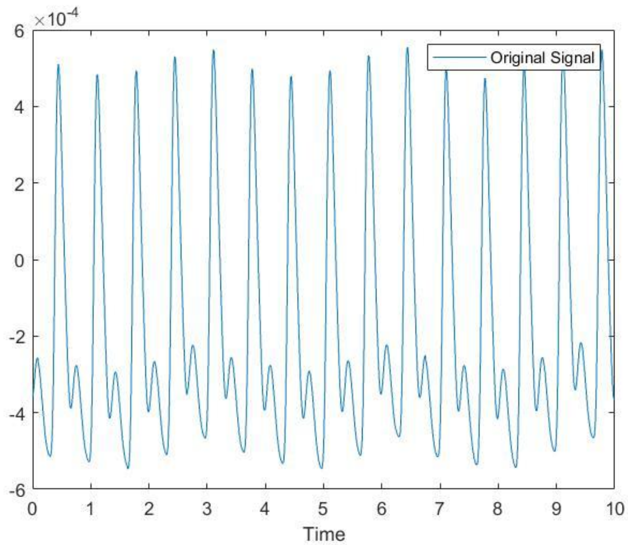


Figure 1.77

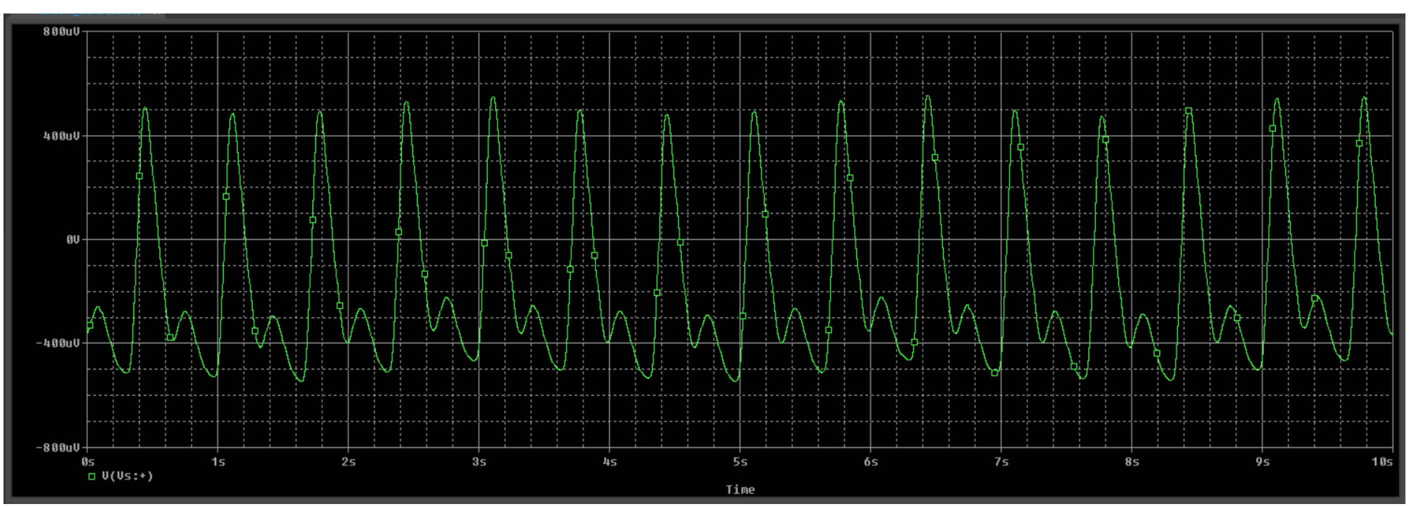


Figure 1.78

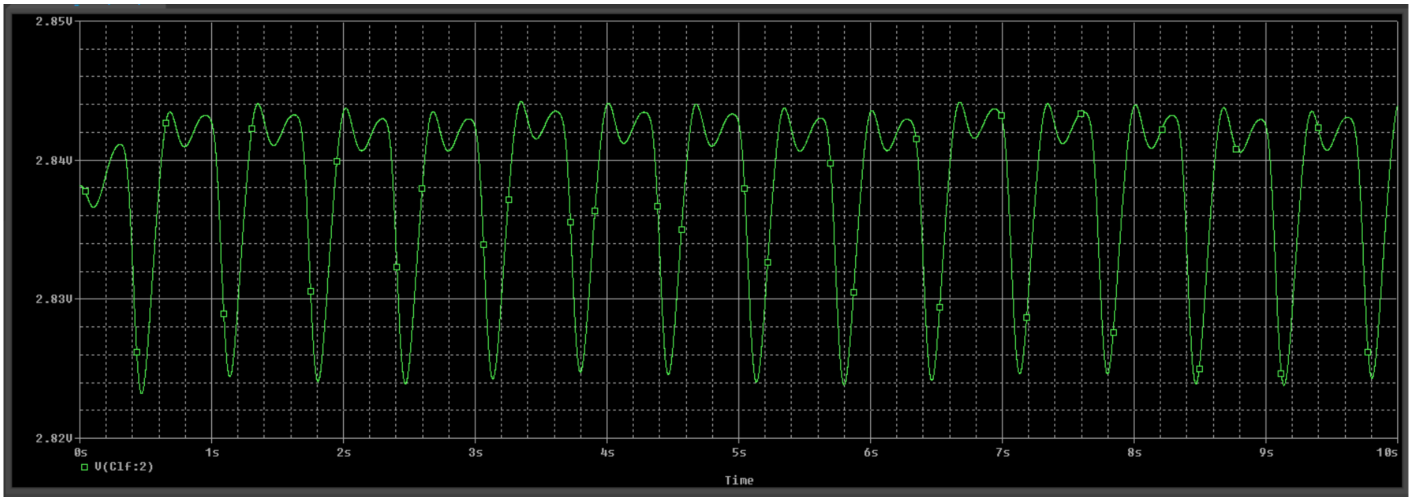


Figure 1.79

<u>Sample signal 3 at SNR = 5dB</u>:

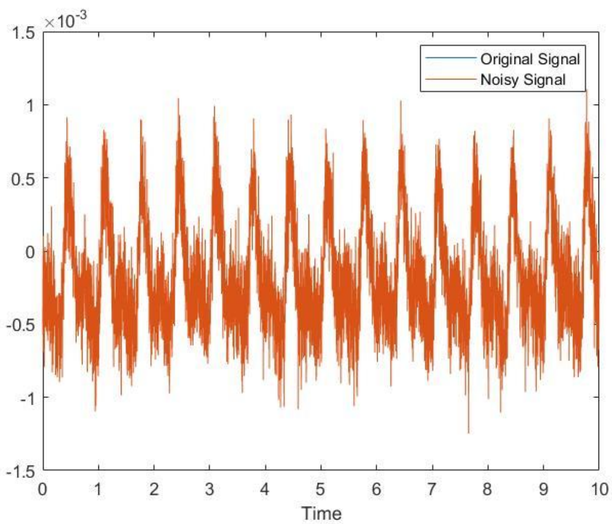


Figure 1.80

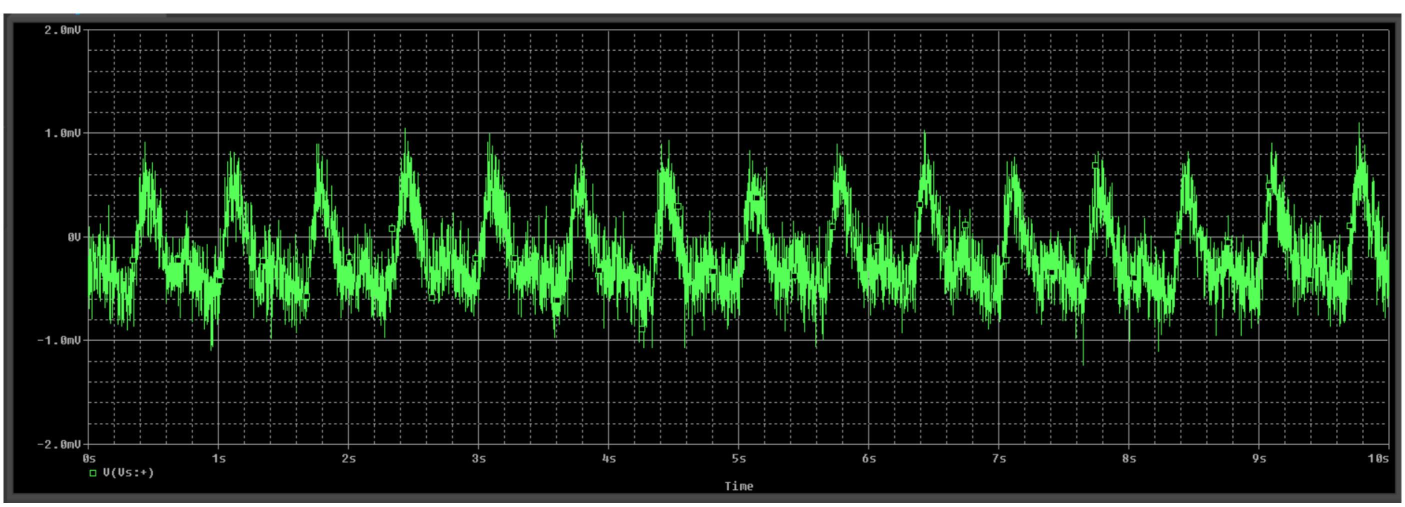


Figure 1.81

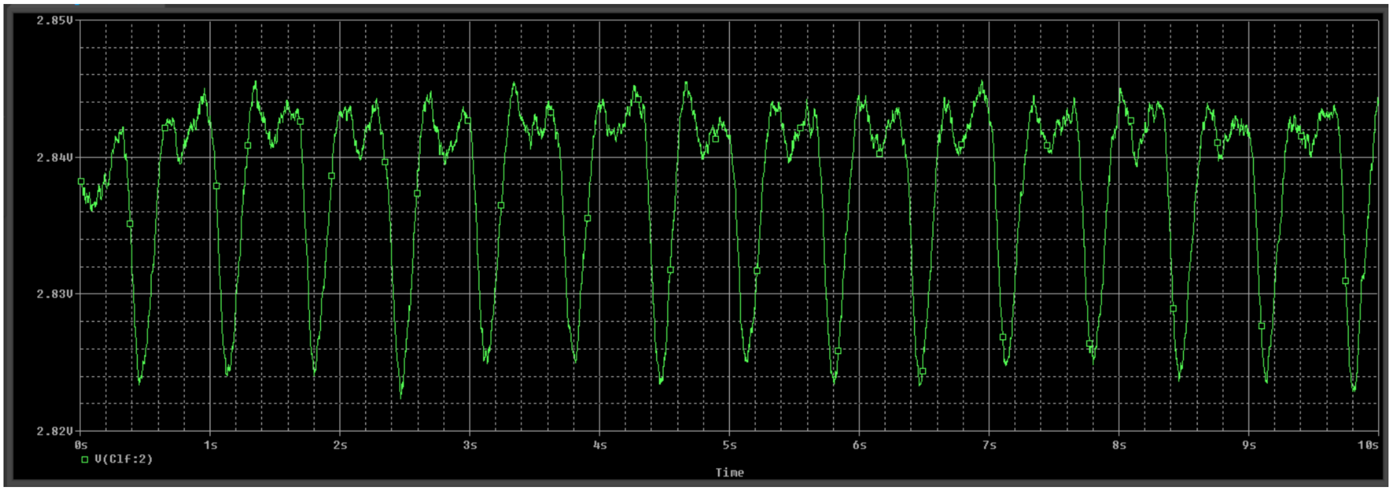


Figure 1.82

Sample signal 3 at SNR = 10dB:

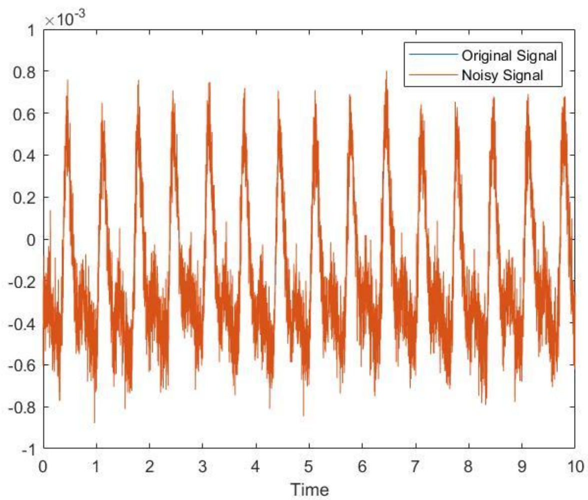


Figure 1.83

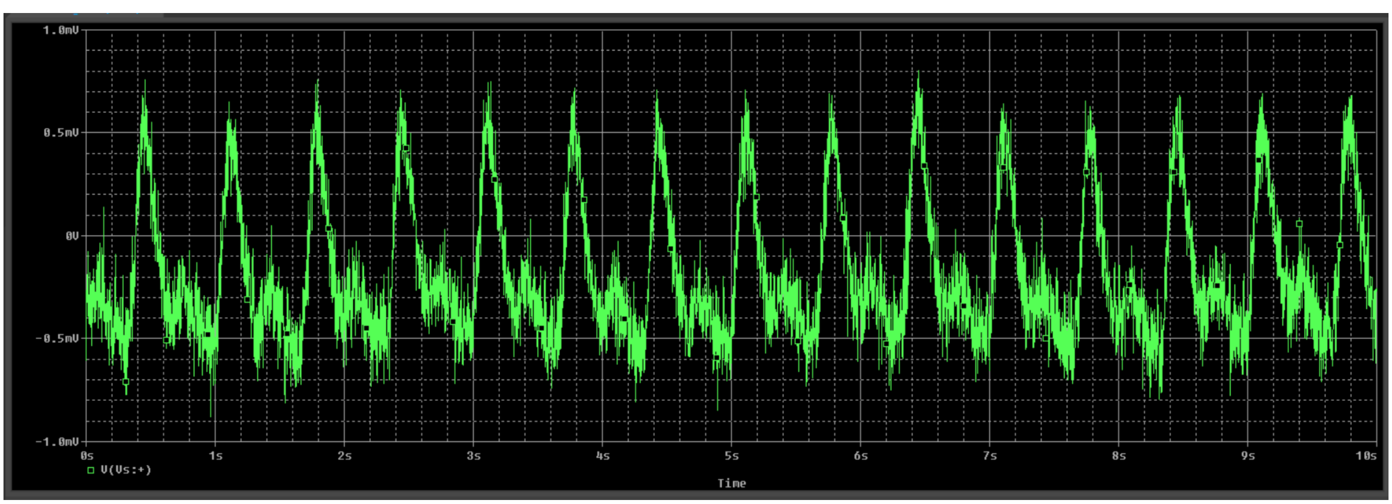


Figure 1.84

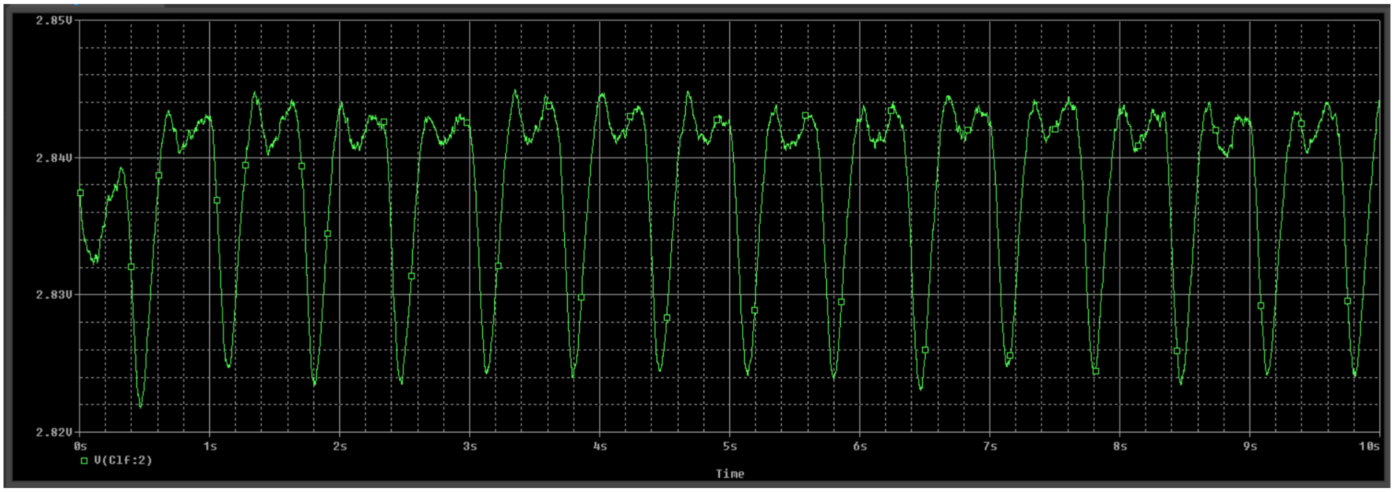


Figure 1.85

<u>Sample signal 3 at SNR = 15dB</u>:

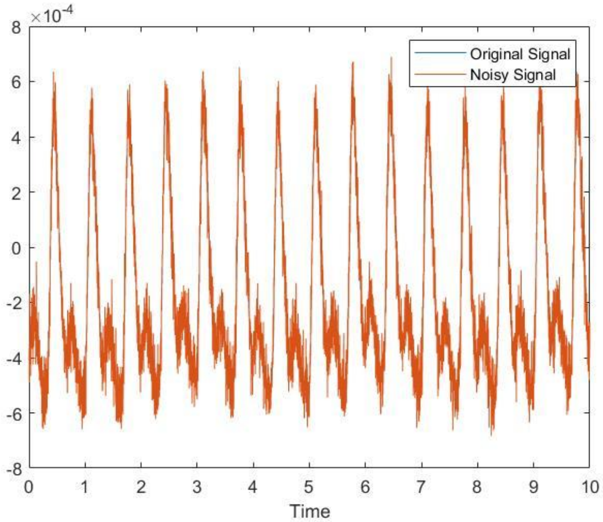


Figure 1.86

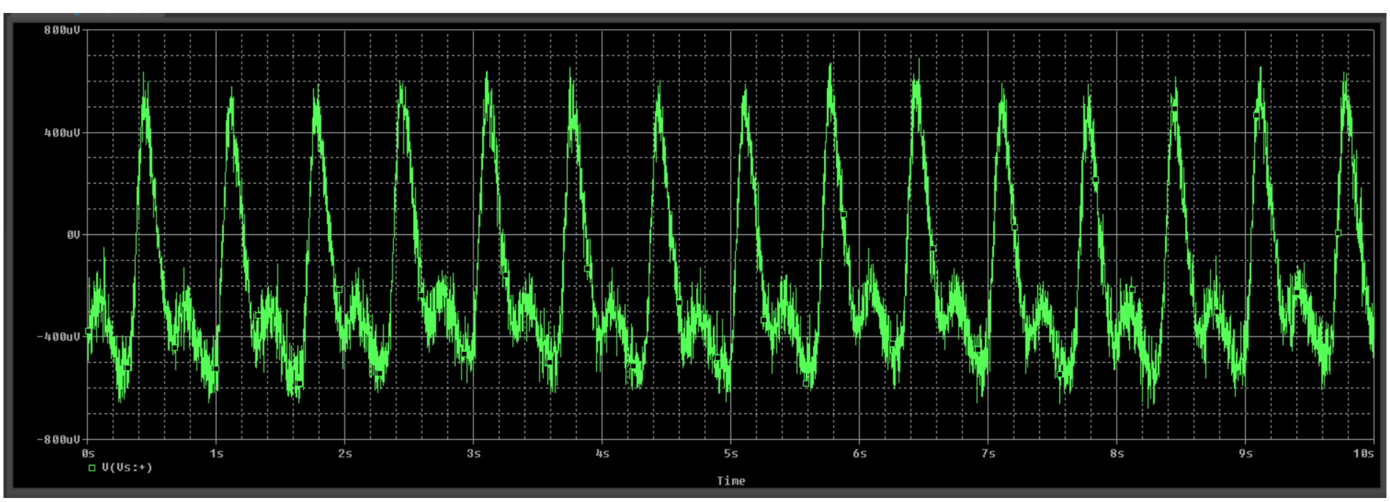


Figure 1.87

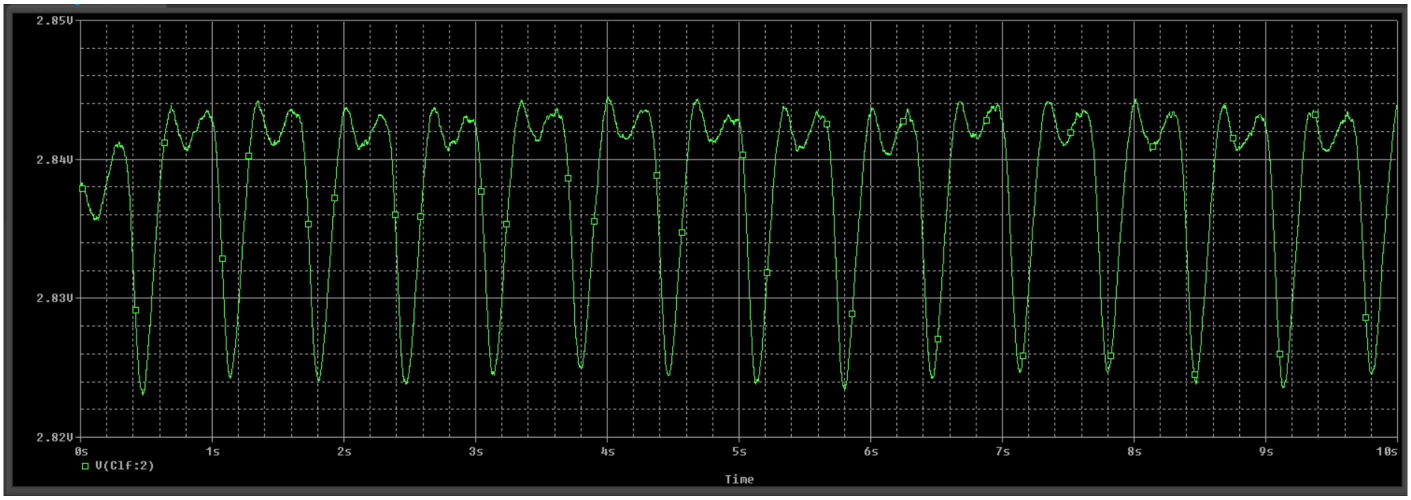


Figure 1.88

<u>Sample signal 3 at SNR = 20dB</u>:

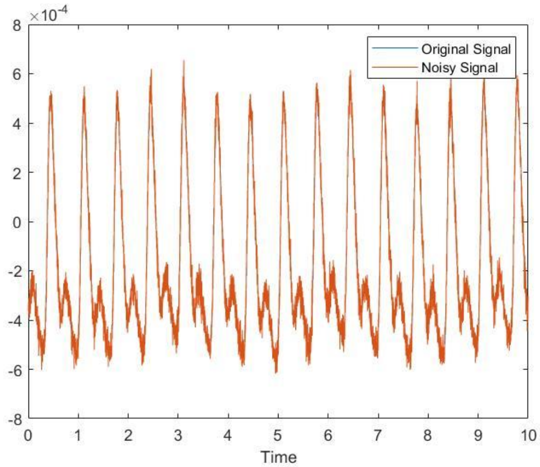


Figure 1.89

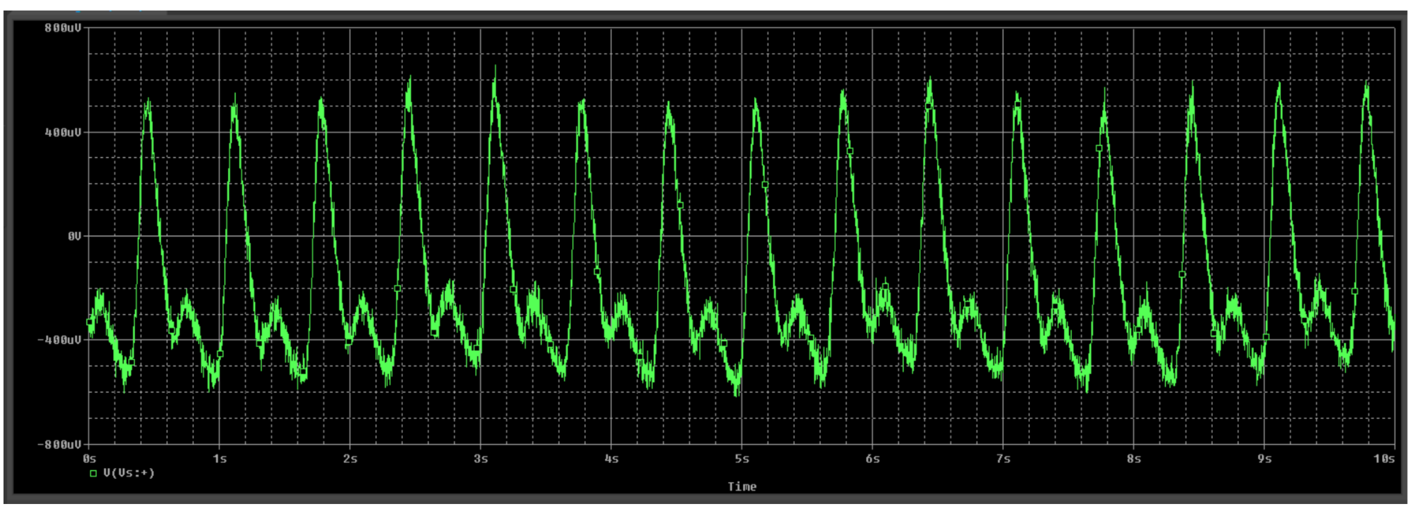


Figure 1.90

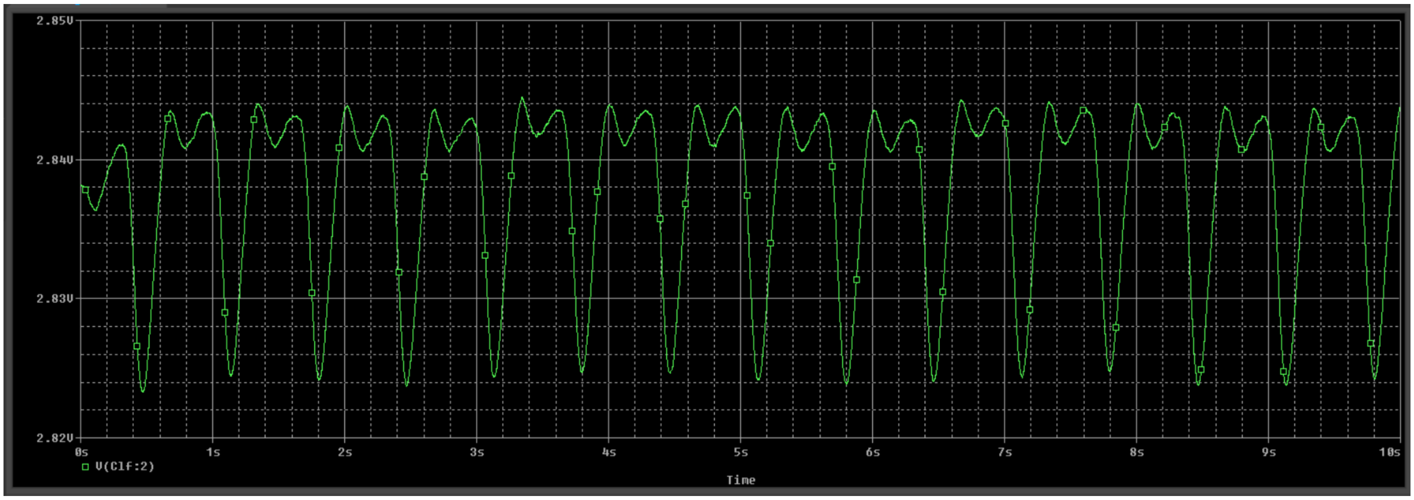


Figure 1.91

<u>Sample signal 3 at SNR = 25dB</u>:

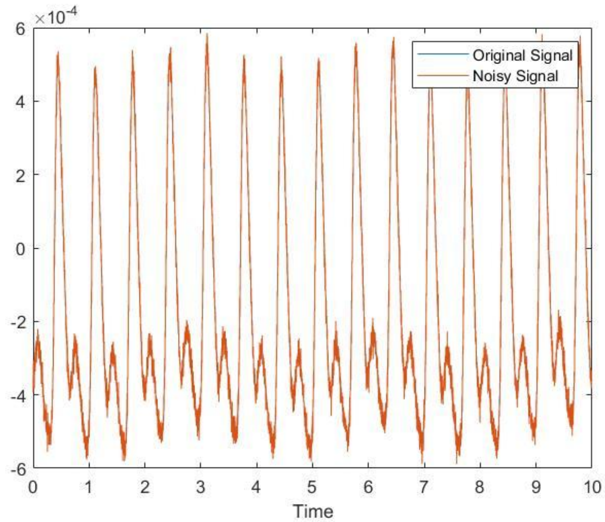


Figure 1.92

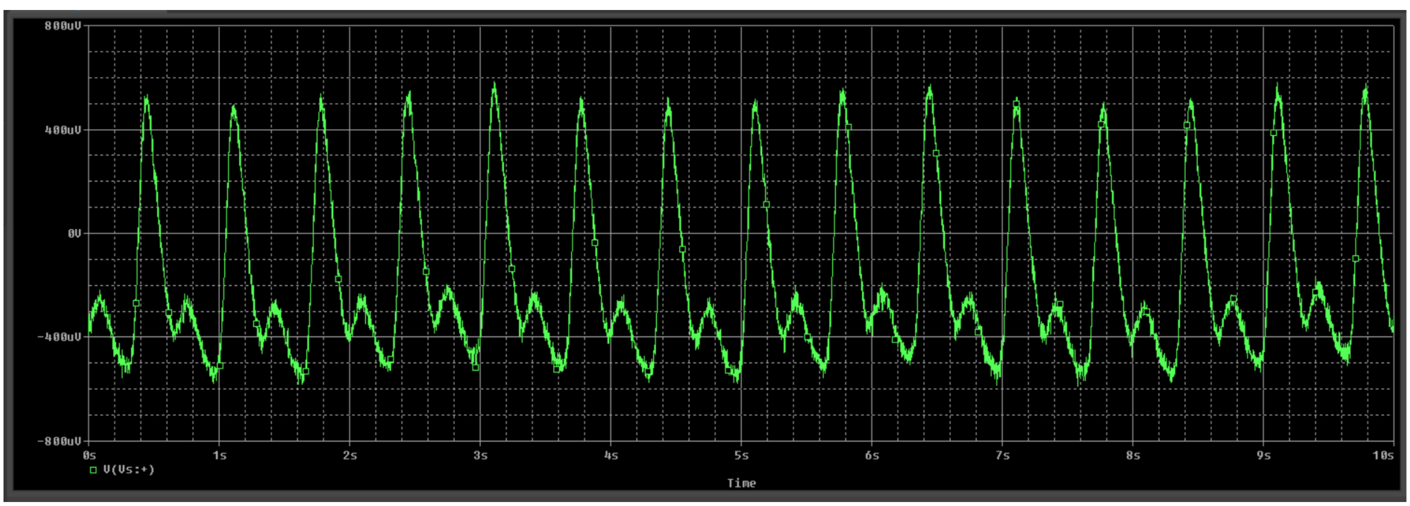


Figure 1.93

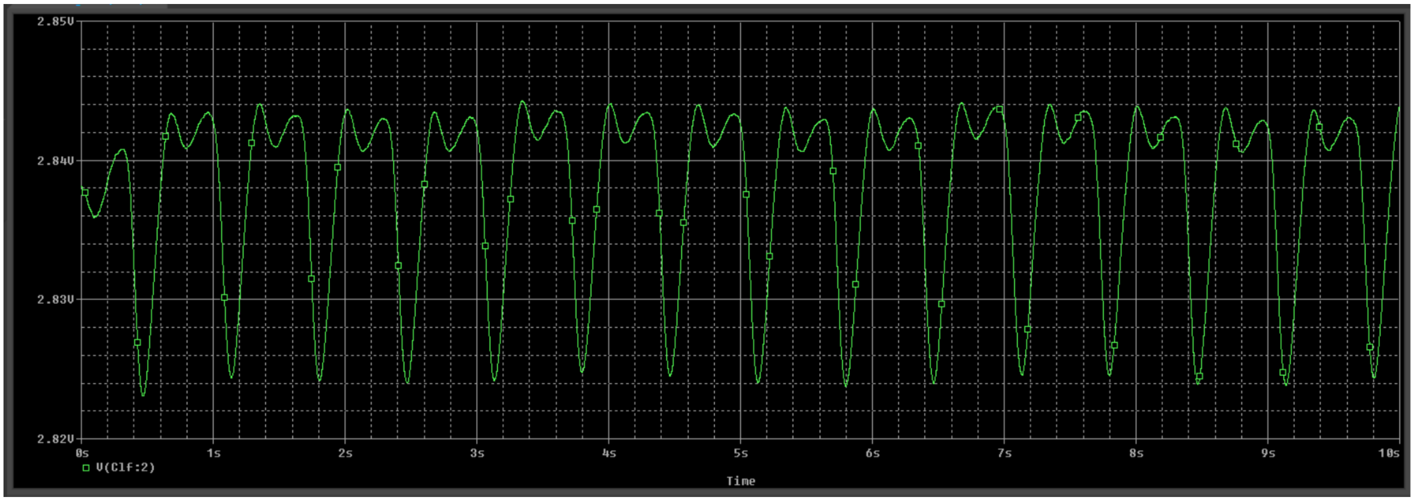


Figure 1.94

<u>Sample signal 3 at SNR = 30dB</u>:

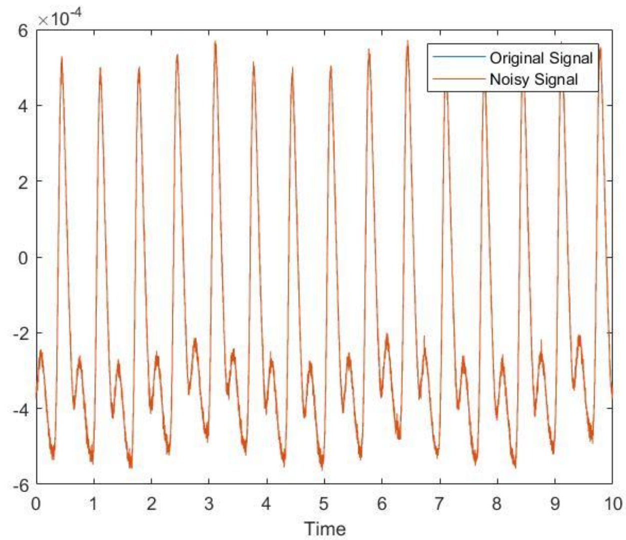


Figure 1.95

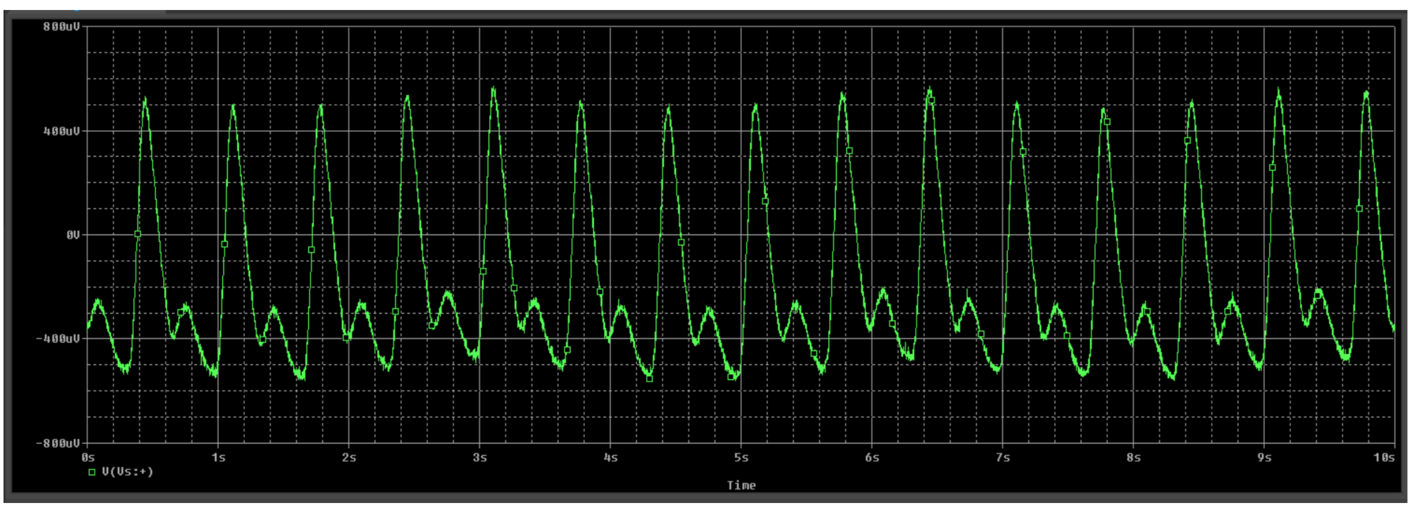


Figure 1.96

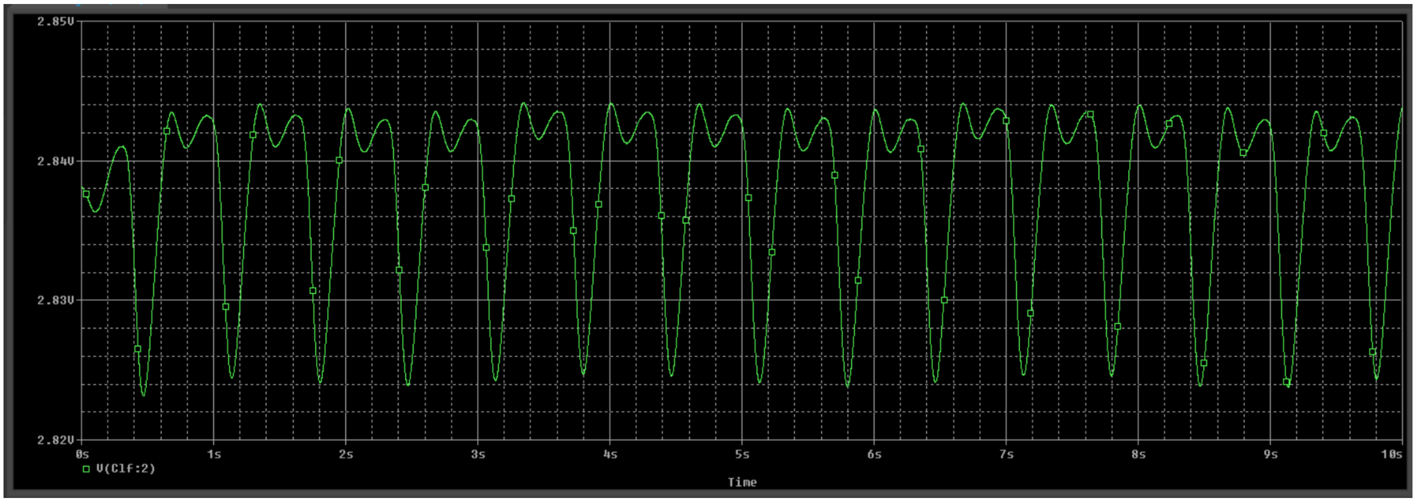


Figure 1.97

3.7 <u>PCB Layout and Connection with Arduino Uno</u>

The PCB of the project was designed with the help of Altium Designer. It is a two layered PCB. The two layers are the Top Layer and the Bottom Layer. The Dimensions of Board are 25.527 mm and 18.288 mm. The traces through 5V VCC and GND are made larger than other traces because more current passes through them. This adjustment gives the advantage of not having a GND plane.

The Steps involved in designing the PCB are:

<u>Step 1</u>: Prepare the Schematic from the simulation outputs. We have selected the appropriate components as explained in the previous sections. After the arrangement, proceed for the Layout.

<u>Step 2</u>: Export the Schematic to the PCB document. Place the components on the PCB board. Arrange them on both sides to decrease the PCB size.

<u>Step 3</u>: Draw the traces. Connect the Top and Bottom traces through vias. The traces connecting the GND and VCC should be thicker.

<u>Step 4</u>: Place the Designators and 3D models to make up the Final PCB.

The layout is shown in the Figure 1.98.

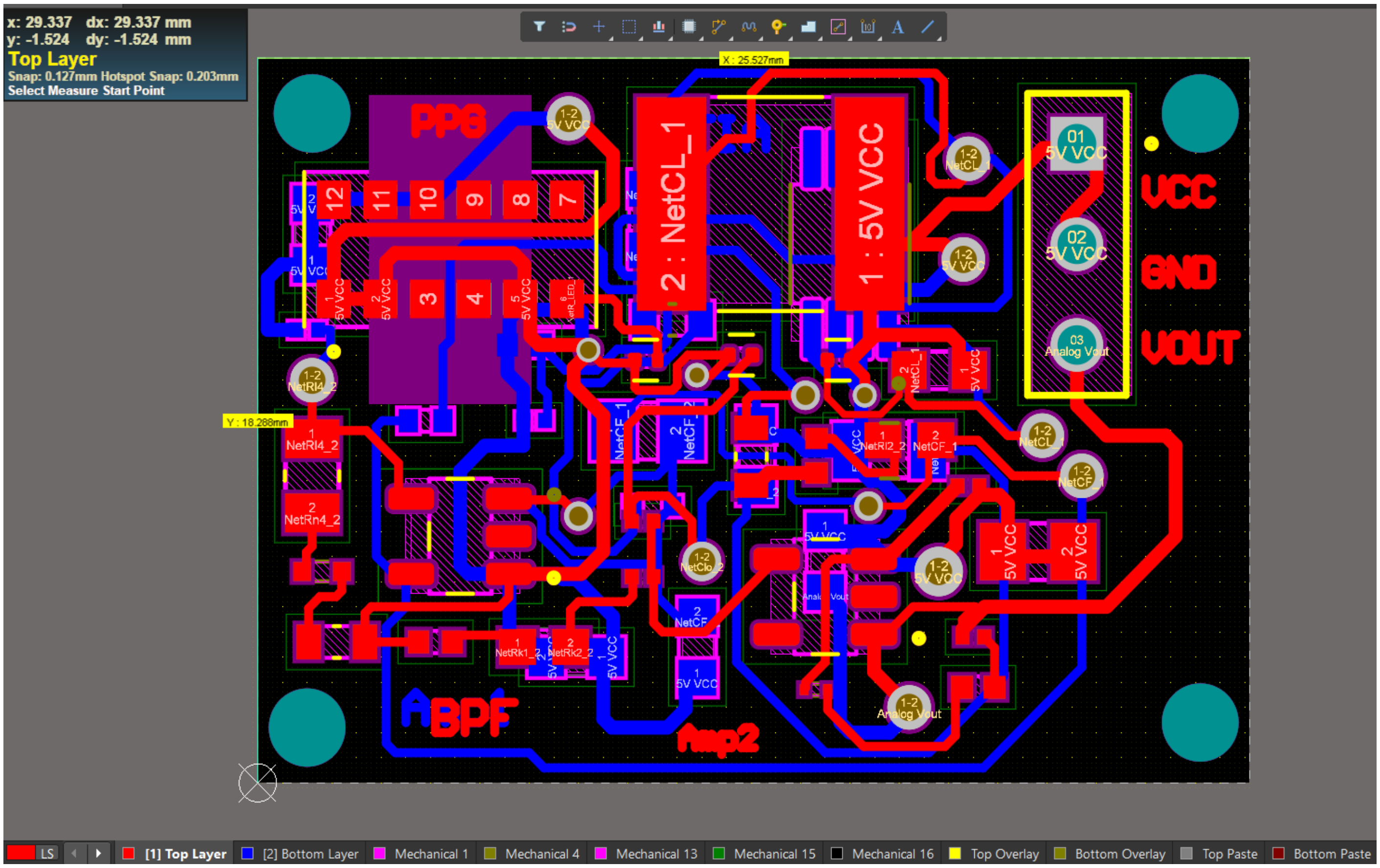


Figure 1.98

BLOCK DIAGRAM

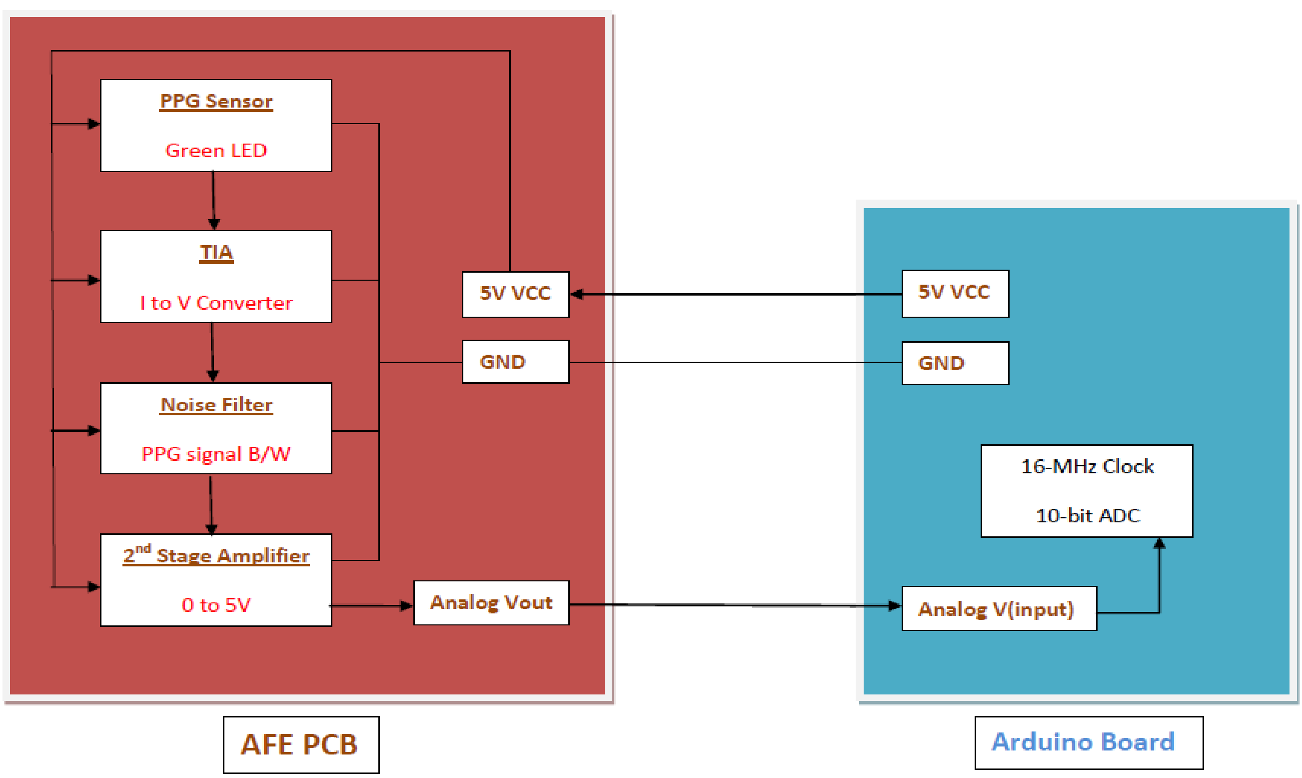


Figure 1.99

3.8 The Analysis of PPG signal with Arduino Uno

The following code can be used to plot the PPG signals in the Serial Plotter of Arduino IDE:

```
int Vpin = A0;

void setup () {
  // put your setup code here, to run once:
  Serial.begin(9600);
}

void loop () {
  // put your main code here, to run repeatedly:
  int PPGval = analogRead (Vpin);     //reads values from Vpin.
  float PPGval_f = PPGval*(5.0/1023.0);  //maps the extreme ends analog values from (0 -
1023) to (0 - 5V)  for our understanding.
  Serial.print(PPFval_f);   //print out the value we read in Serial Plotter.
}
```

The following code can be used for calculating the heart-rate by measuring the frequency of PPG signal using FFT algorithm:

```
#include "arduinoFFT.h"


#define SAMPLES 128             //Must be a power of 2
#define SAMPLING_FREQUENCY 1000 //Hz, must be less than 10000 due to ADC


arduinoFFT FFT = arduinoFFT();


unsigned int sampling_period_us;
unsigned long microseconds;


double vReal[SAMPLES];
double vImag[SAMPLES];


float BPM;


void setup() {
```

```
  Serial.begin(115200);

  sampling_period_us = round(1000000*(1.0/SAMPLING_FREQUENCY));
}

void loop() {

  /*SAMPLING*/
  for(int i=0; i<SAMPLES; i++)
  {
    microseconds = micros();    //Overflows after around 70 minutes!

    vReal[i] = analogRead(0);
    vImag[i] = 0;

    while(micros() < (microseconds + sampling_period_us)){
    }
  }

  /*FFT*/
  FFT.Windowing(vReal, SAMPLES, FFT_WIN_TYP_HAMMING, FFT_FORWARD);
  FFT.Compute(vReal, vImag, SAMPLES, FFT_FORWARD);
  FFT.ComplexToMagnitude(vReal, vImag, SAMPLES);
  double peak = FFT.MajorPeak(vReal, SAMPLES, SAMPLING_FREQUENCY);     //Find
out what frequency is the most dominant.

  /*PRINT RESULTS*/
  BPM = peak*60.0*1000;     //Calculate the heart-rate in terms of beats-per-minute (BPM).
  Serial.println(BPM);     //Print out the heart-rate in terms of beats-per-minute (BPM).

  //for(int i=0; i<(SAMPLES/2); i++)
  //{
    /*View all these three lines in serial terminal to see which frequencies has which
amplitudes*/
```

```
    //Serial.print((i * 1.0 * SAMPLING_FREQUENCY) / SAMPLES, 1);
    //Serial.print(" ");
    //Serial.println(vReal[i], 1);    //View only this line in serial plotter to visualize the bins
  //}

  delay(1000);  //Repeat the process every second OR:
  //while(1);       //Run code once
}
```

3. RESULT AND DISCUSSION

4.1 Results from PSPICE Simulation

The simulation outputs display the detailed performance of the analog-front-end circuit for the PPG sensor. The simulations are done at each level step-by-step as shown in detail in the above sections. The final output from the Signal Conditioning Circuit is a PPG signal within range of 0 – 5 V, which is compatible for input analog pin to Arduino Uno micro-controller Board without the loss of any quantisation level for PPG signals of any frequency.

4.2 Results from MATLAB Simulation

The purpose of MATLAB simulation is shown in Section 3.6, to analyse the performance of noise cancelling ability the band-pass filter used in our design to real world PPG signals by adding AWGN at different SNR values. The output waveforms generated after passing the real-world PPG signal through the band-pass filter and amplifier shown in Section 3.6. The output signals were almost accurate as for the original signal even for AWGN with low SNR value.

4.3 PCB Design Analysis

From the PCB Layout analysis, we can understand the detailed structure and functionality of the PCB. The Figures 1.100 & 1.101 shows the Top view and Bottom view of the PCB respectively. The main focus while designing was to keep it simple and compact. To make it visually appealing, we have chosen the best suited 3D models available.

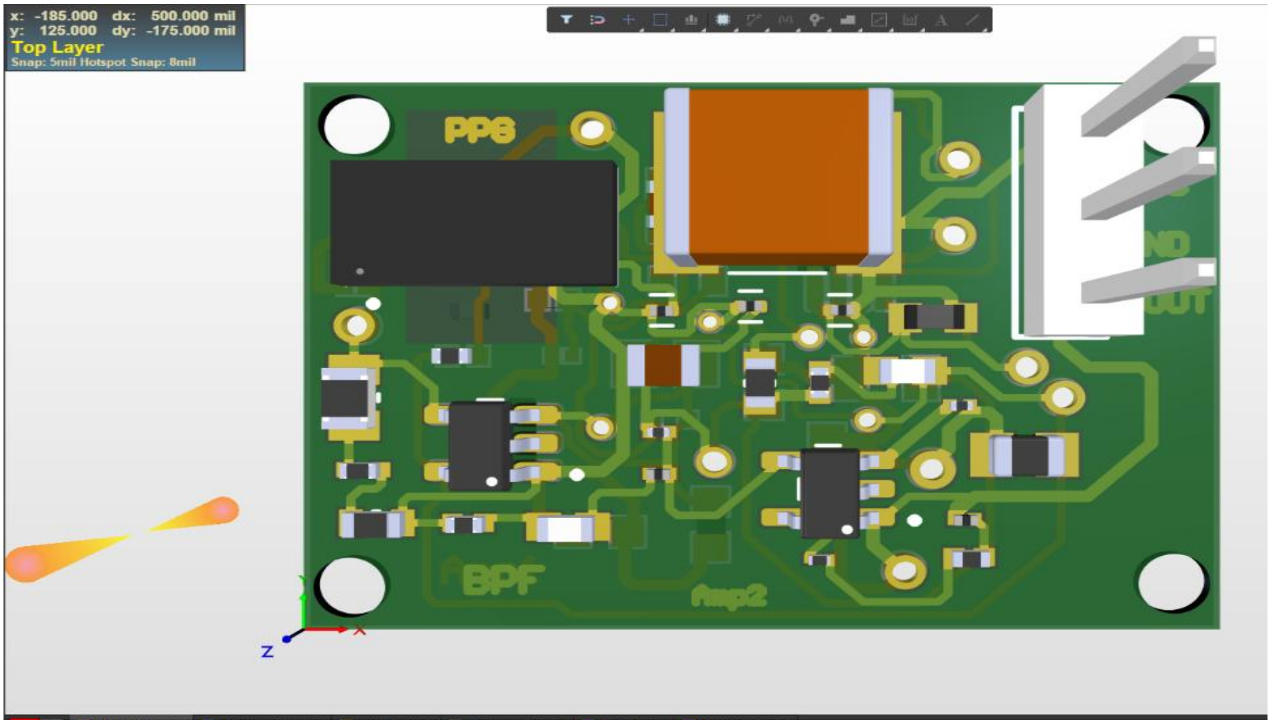


Figure 1.100

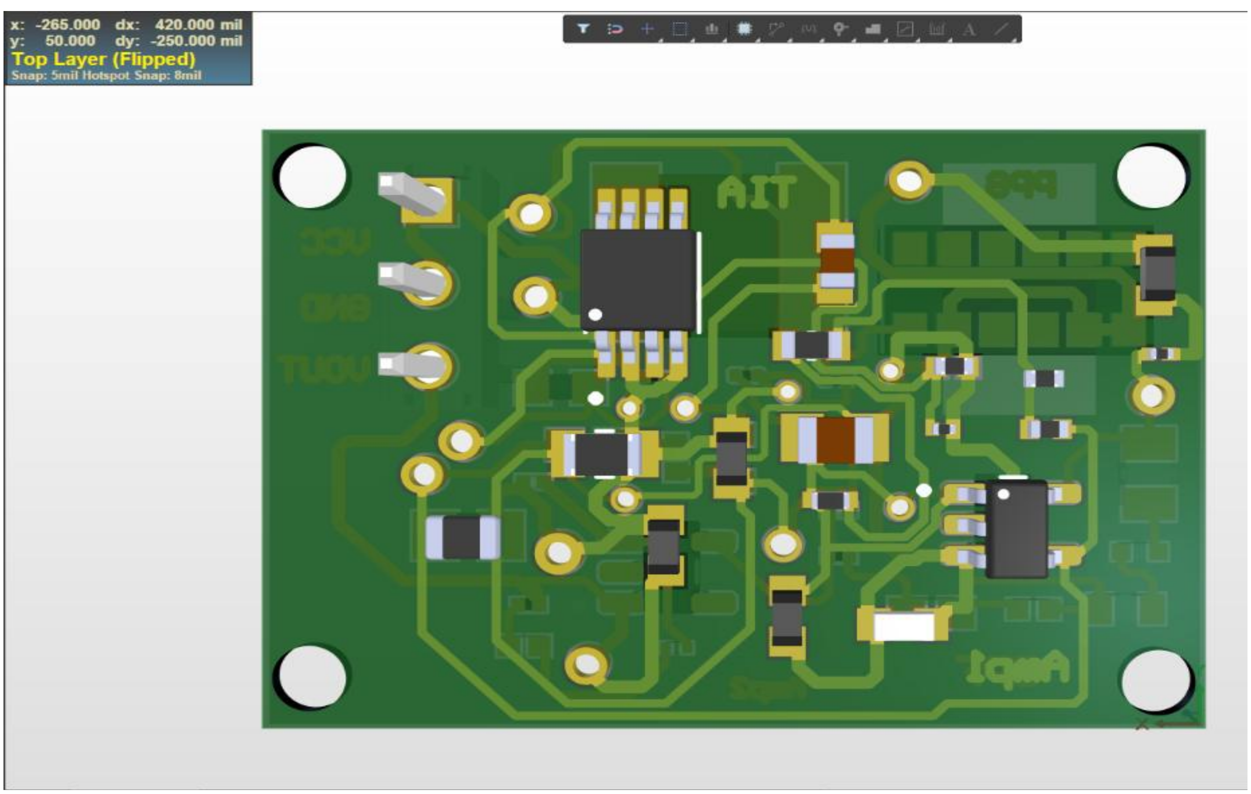


Figure 1.101

4.4 Comparison with previous models

This project has been developed with immense research based on the previous models. The main inspiration has been the one mentioned in references below.

The Signal Conditioning Circuit developed can operate with a 5 V supply consuming less power. It is more efficient in terms of durability and dimensions, the dimensions of the PCB are 25.57 mm × 18 mm.

The Arduino Uno micro-controller Board is used as the Signal Processing Circuit. It is powered by an Atmega328P micro-processor which is an Advanced Virtual RISC (AVR) microcontroller operating at 16MHz, includes 32KB of program memory, 1KB of EEPROM, 2KB of RAM. These features consist of advanced RISC architecture, good performance, low power consumption, real timer counter having separate oscillator, 6 PWM pins, programmable Serial UART, programming lock for software security, throughput up to 20 MIPS etc. The Arduino Uno is inexpensive, open source in hardware, don't need to external programmer (burner), Arduino IDE can operate on any operating system. These features have enhanced our project.

4. CONCLUSION

The PPG signal analysis is a modern way to monitor the heart-rate, oxygen saturation level, etc. Many innovative models involving unique and innovative Signal Conditioning Circuit have been developed and proposed. We hope this could be among them. The project achieves the target of improving the PPG sensor analysis through the Analog Front End circuit. This project develops a unique Signal Conditioning circuit which is compatible for the analog input pins of the Arduino Uno micro-controller Board.

We have used Arduino Uno as the Signal Processing Circuit. This helps in reducing the complexity in designing the analog-front-end circuit. The Arduino Uno micro-controller Board has high frequency 16 MHz clock which process the input analog voltage signal at a very fast rate. The ATmega328p micro-processor in Arduino Board have a 10 bit ADC to quantise the analog input signal into $2^{10} = 1024$ quantisation levels and hence improving the processing efficiency of the PPG signal. The focus was also to reduce the size of the PCB. We have taken care of the components used.

There is always a question about the authenticity of the project in real world applications. To counteract this issue, we have taken the real-world PPG signals and added the AWGN at some particular measured SNR. The output waveforms of the signals through the Second Order Band-pass filter and amplifiers depicted the worthiness of the design of the Signal Conditioning Circuit. The outputs were accurate even when the noise was interrupting. The advantage of using a micro-controller board like Arduino Uno is that we can project the processed data to various platforms for further analysis. The project has a good record of all the simulation outputs. Each and every step in the project is presented in a lucid manner. This helps new researchers to progress quickly without any doubt. The final conclusion is that we have developed a decent model to analyse the PPG signals.

5. FUTURE SCOPE

The project is based on Arduino Uno micro-controller Board and the Analog-Front-End device for PPG signals. Arduino Uno can be combined with a Bluetooth Module for example Blue giga WT11. The Bluetooth module can be used to control the project over the internet to access the processed data over a remote location. The Blue giga WT11 module on the Arduino BT provides Bluetooth communication with computers, phones, and other Bluetooth devices. The WT11 communicates with the ATmega328P via serial pins (shared with the RX and TX pins on the Arduino Uno board).

Red and infrared (IR) lights are used for the application of pulse oximetry using PPG signals to estimate the true haemoglobin oxygen saturation of arterial blood. Oxyhaemoglobin (HbO2) absorbs visible and infrared light differently than deoxyhaemoglobin (Hb) and appears bright red as opposed to the darker brown colour of Hb. This purpose can also be fulfilled using the SFH7072 sensor used in our design. The SFH7072 sensor contains a Red and an IR LED which can be used for the calculation of oxygen saturation level with a different Signal Conditioning Circuits that can be accommodated in the design of the analog-front-end circuit. In this way, we can measure both the oxygen saturation level and the heart-rate using a single device.

6. REFERENCES

1. SFH7072 Datasheet by *OSRAM*.

2. OPA2380 Datasheet by *Texas Instruments*.

3. LMH6609 Datasheet by *Texas Instruments*.

4. Bagha, S., & Shaw, L. (2011). A real time analysis of PPG signal for measurement of SpO2 and pulse rate. *International journal of computer applications*, *36*(11), 45-50.

5. "Design Guide: TIDA-010029 Wearable, 16-phase multi-sensor SpO2 and heart rate monitor (HRM) reference design with Bluetooth® 5" by Texas Instruments.