SIGNALS AND SYSTEM PROJECT

GROUP MEMBERS

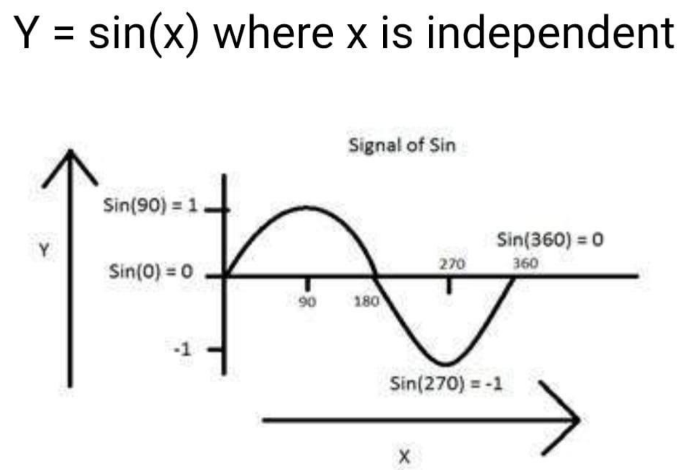
1. AMARANA PREMDEEP B210844EE
2. ASAM KASIREDDY B210904EE
3. ADDEPALLY JAYADEV B210899EE

Introduction to Signals

Any physical quantity that carries some information can be called a signal.The physical quantities like temperature, pressure, humidity, etc. are continuously monitored in a process. Usually, the information carried by asignal is a function of some independent variable, for example, time. The actual value of the signal at any instant of time is called its amplitude.
These signals are normally plotted as amplitude vs. time graph. This graphis termed as the waveform of the signal. The signal can be a function of one or more independent variables. Let us now define a signal.

Definition of a signal

A signal can be defined as any physical quantity that varies with one or more independent variables.



As previously mentioned, a signal represents some underlying physical variable of interest. As an abstraction, we consider a signal as simply a real-valued(or sometimes complex-valued) function defined on some domain. For example , we might represent a signal as just a real-valued function of timex(·) where x(t) represents the value of the signal at time t.

SYSTEM

The term system is used in this abstract and technical sense to refer to such mappings that take a signal as input and produce another signal asoutput.

In this figure, we think of a system as some part of the physical world that interacts with its environment and is designed to carry out some task. The box labeled “**information processing**” receives input signals from various sensors and produces output signals for various actuators. Thus, we think of the system as transforming input signals into output signals.

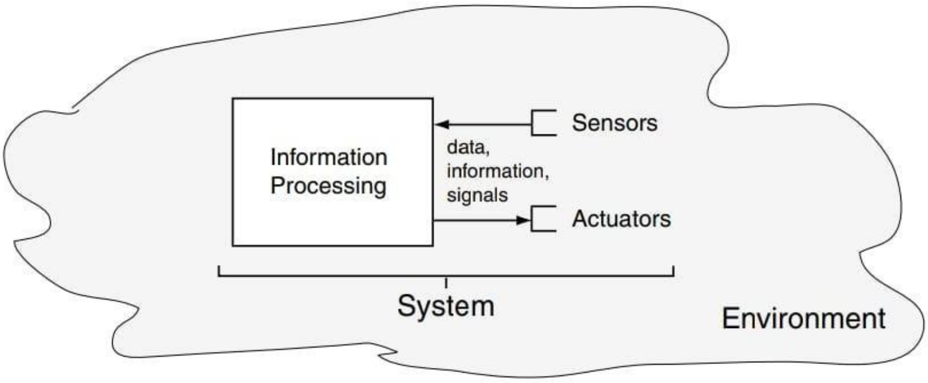


Figure 3.1: A general system immersed in its environment.

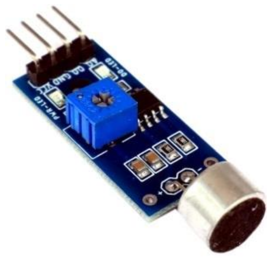
PROJECT

Fourier transformation of a real time audio signal and plotting amplitude vs frequency in MATLAB.

Here we used MATLAB and C programming to achieve this fourier transformation of real time sound signal.

HARDWARE USED

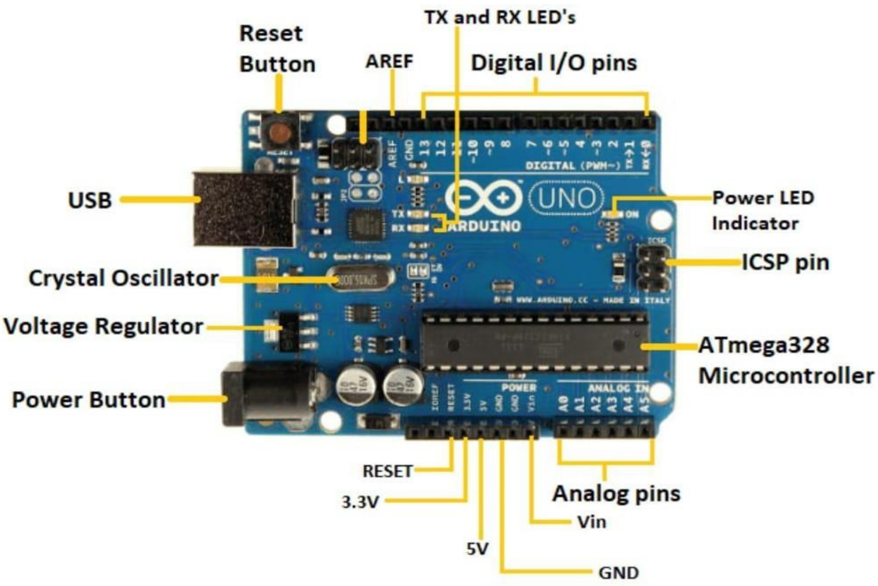
1.MICROPHONE



This sound sensor employs a microphone to provide input to the buffer, peak detector and an amplifier. This sensor notices a sound and processes an output voltage signal to a microcontroller. After that, it executes required processing.

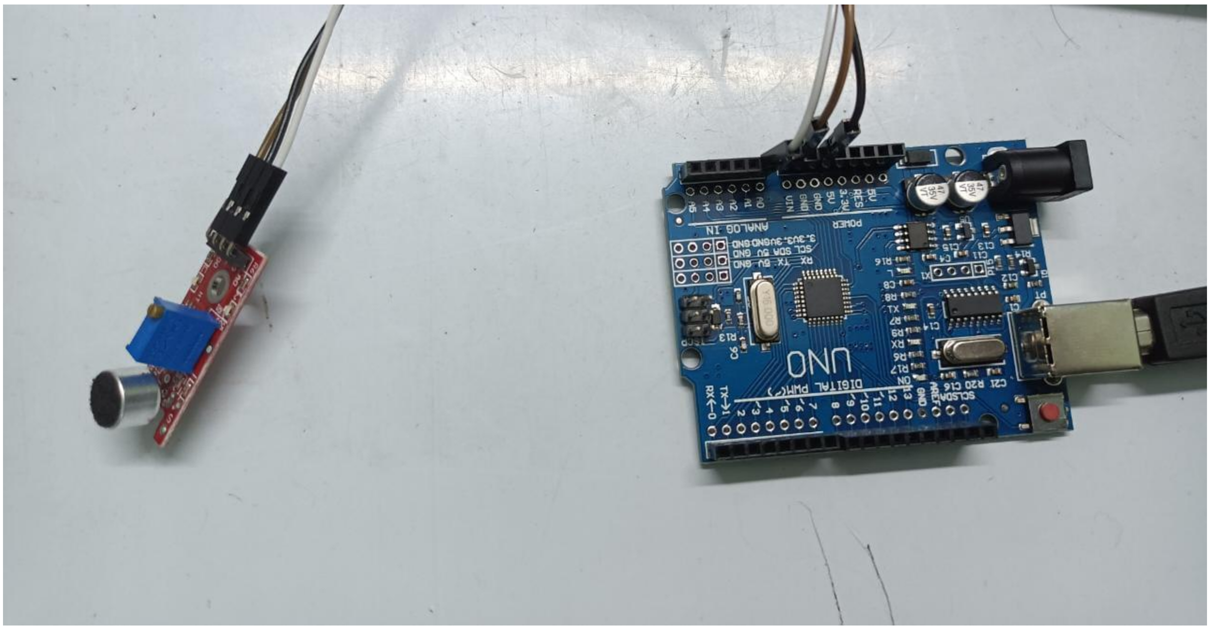
This sensor is capable of determining noise levels within decibel(db) or decibels at 3KHz/ 6KHz frequencies approximately wherever the humanear is sensitive.

2. ARDUINO UNO

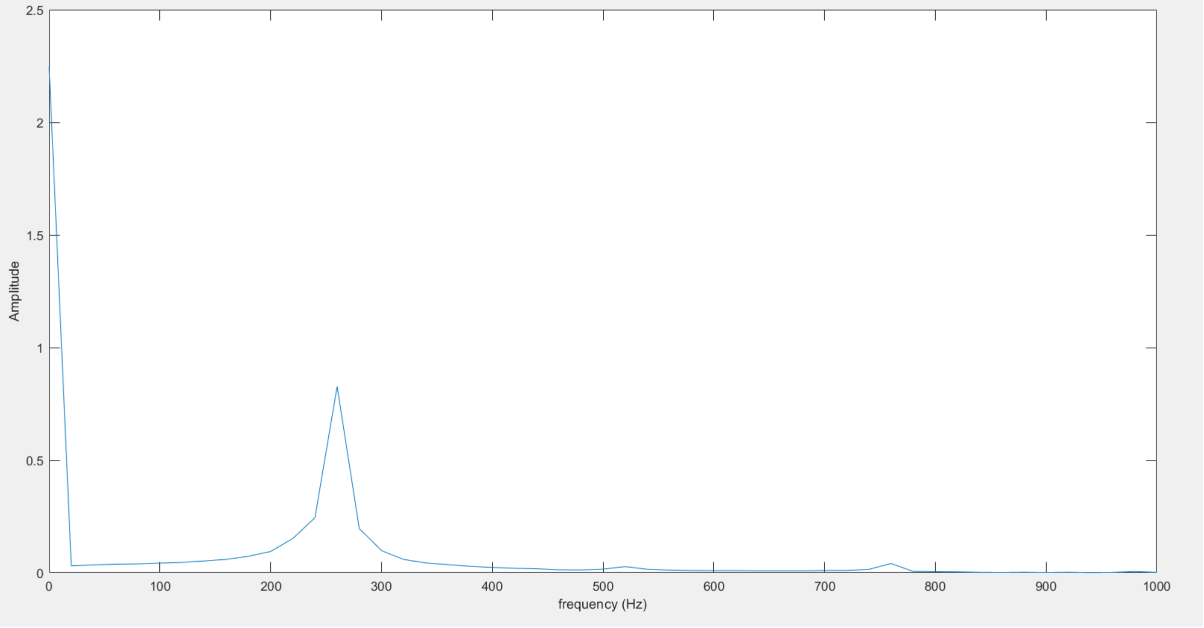


3.JUMPER WIRES

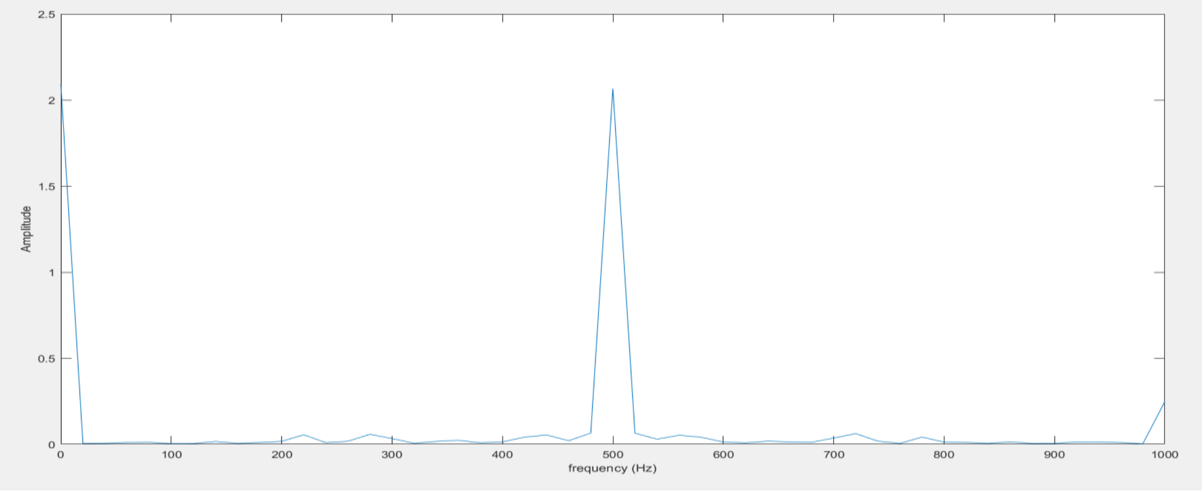
SNAPSHOT OF ACTUAL SETUP :



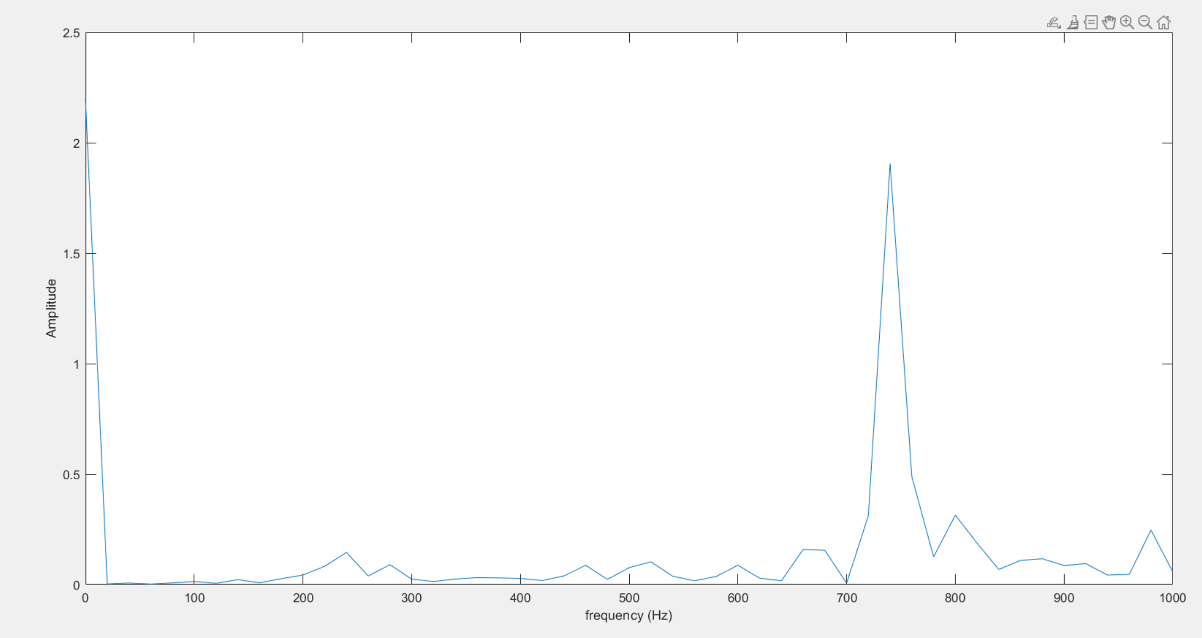
AMPLITUDE VS FREQUENCY OUTPUT OF 250HZ



2. AMPLITUDE VS FREQUENCY OUTPUT OF 500HZ



3. AMPLITUDE VS FREQUENCY OUTPUT OF 750 HZ



CODE:

IN ARDUINO IDE

```
void setup() {

  Serial.begin(120000);
}



String audiosig;

void loop() {

  audiosig = String(analogRead(A0));
if(audiosig.length()<4) Serial.print(0);

  if(audiosig.length()<3) Serial.print(0);
  if(audiosig.length()<2) Serial.print(0);
  Serial.println(audiosig);


}
```

IN MATLAB

```
A0=serialport("COM6",120000); %connection to arduino in serial port
    with 120000 baudrate

asignal=zeros(1,10000000);          %storing audio signal
i=1;            %looping index
freq=2000;          %sampling frequency
l=100;              %signal length
f=freq*(0:(l/2))/l;     %frequency interval


flush(A0);       % clearing arduino and taking input signal from it
while i<10000000         %loop until data variable is filled
    input = readline(A0);     %reading each line from arduino
    asignal(i) = str2double(input)*0.0049;  %converting string to voltage

     %Fast fourier transform
    if i>l && mod(i,100)==0       %for every 100 readings plotting fourier
    transform
        b=fft(asignal(i-l+1:i));       %performing fourier transform
        Y2 = abs(b/l);               %taking absolute values
        Y1 = Y2(1:l/2+1);           %fourier transform is symmetrical about
    the peak
        Y1(2:end-1) = 2*Y1(2:end-1);


      %plotting fourier transform with frequency on x and amplitude on   % yaxis
        plot(f,Y1)
        xlabel("Frequency(Hz)")
        ylabel("Amplitude")
    end
    i=i+1;       %increment loop variable
end
```

DESIGNING:
CODE IN ARDUINO IDE

This program starts with setup() function which is executed once at the beginning of the program only and we initialize the serial communication data transfer rate bits per second.

Then the loop function is executed repeatedly until program is stopped.

It reads the analog input from pin A0 using the analogRead() and converts into a string STRING() and then print the string to the serial port using the Serial.println().

Printed value is always 4 character long and if not it add zeroes in front which is checked by if function.

CODE IN MATLAB:

This code is used to read the data from an Arduino connected to the computer port and analyse the data using fast fourier transform and plot the amplitude vs frequency plot.

Firstly new serial port is created which is used to communicate with Arduino connected with the computer port and we define the baud rate.

Now we created a array of data where data is stored and i(counter in while loop)=1 and f(sampling frequency)=2000 and l(samples for fft calculation)=100 then we define time vector and frequency vector and flush the Arduino and run while loop and read the single line of text data and stores it in in and condition is checked of I and function is written to store the previous data to current data.

Now from two sided spectrum to single sided spectrum is converted and double it as it has same for negative and positive cycles and plot amplitude vs frequency plot.

RESULTS:

The amplitude vs frequency plot of any audio signal can be obtained using this setup.

CONCLUSION:

In conclusion, this project successfully demonstrated the creation of an amplitude vs frequency plot of an audio signal using Arduino board. The resulting plot showed the variation of amplitude with frequency and allowed the calculation of the dominant frequency of the audio signal. This project has important applications in various fields such as audio processing, music, and speech analysis. The Arduino board provides an affordable and accessible platform for processing audio signals and generating amplitude vs frequency plots.

REFERENCES

1) www.cambridge.org/core/books/abs/signals-and-systems/introduction-to-signals/3F537A4403031EE972B949BDC96EF3D7

2) www.princeton.edu/~cuff/ele201/kulkarni_text/systems.pdf

3) ocw.mit.edu/courses/res-6-007-signals-and-systems-spring-2011/

4) https://www.tutorialspoint.com/dip/signals_and_system_in