DESIGNING A DUAL-TONE MULTI-FREQUENCY TONE DETECTOR EE479 Project-1

İbrahim Kahraman
2015401108

INTRODUCTION

Dual tone multi-frequency signaling(DTMF) is a method in communication systems, where frequencies of button signals are used to separate them from each other. Today all people use phones and when they want to call someone they push on the digits in their phone. Each digits are composed by 2 sinusoidal signals with different frequencies. At the receiver side they should correctly decode in order to understand the message. In this project we deal with the detection of DTMF signals. In order to distinguish dual tones we design bandpass filters and after filtering the received signal we calculated energy of each filtered signals. Then, we compare energy of all filtered signals and detect which frequency component dominant. To receive sound signal we benefited from Arduino and to decode received signal we used Matlab. Each dual tone consist of two main frequency levels and it can be seen in the figure below:
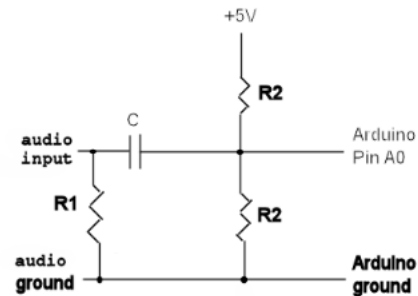
	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

Dual Tones

OBJECTIVES

Our aim is to read voice signals and detect which dual tone is used.

METHOD

R1 = 1K ohms
R2 = 100k
C = 0.1uF

Circuit

```
int sensorValue = 0;

void setup() {
  Serial.begin(115200);
}

void loop() {
  sensorValue = analogRead(0);
  sensorValue = sensorValue / 4;
  Serial.write(sensorValue);
}
```

By using this code, we read the sound signals and then sent them to port. Then we read the port on matlab. The main problem here is sampling rte for arduino is approximately 10000 sample per second but for matlab the sampling rate should be less.

Sampling Rate

To calculate the sampling rate for matlab, first we sent the the signal from online signal generator that we have already know its frequency. Then by taking fourier transorm, we detect its sampling rate. We can also use another method. For example i design a filter:

```
freqs=1000;
range=20;
sample_rate=8800:5:9200;
for i=1:length(sample_rate)
my_filter=designfilt('bandpassfir',...
'FilterOrder',150, ...
'CutoffFrequency1',freqs-f_range, ...
'CutoffFrequency2',freqs+f_range, ...
'SampleRate',sample_rate(i));
out=filter(my_filter,data);
temp(i)=sum(out.^2)
end

[x,y]=max(temp);
sampling_rate=8800+5*(y-1);
```

We sent the signal whose frequency is 1000Hz, then by using designfilt function, then for the manny sampling rate, i calculate the energy. The most energy is seen when sampling rate was 8930. So i select sampling rate 8930.

Processing the Signal on Matlab

I design bandpass FIR filter by using designfilt function. The cut-off frequncies are 15 below and 15 above from the center frequency. For each frequency value , i design bandpass filter so i design 8 filter. I select the order of filter 150.
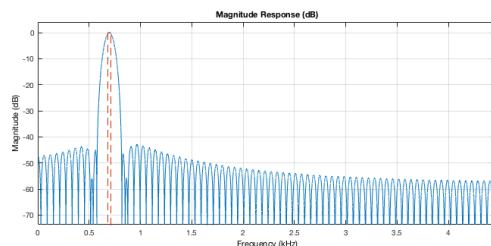
Bandpass

First i subtract the dc value from data and then filtered the data bu using 8 bandpass FIR filter. So for the frequency values 697, 770, 852, 941, i select the the frequency level by looking the energy of the output after filtered. I select the frequency that the output has maximum energy. On the other hand, if we do not put the button, there will be also some maximum component but we should not print it. In order to detect it, first i look the energy of input, then i look the energy of the output. If energy of filtered data is smaller then %10 of energy of the data, i classify this data as a out of range data. On the other hand the other task is, if we put a button very long time, the matlab should print only once. To control this issue, i print only when the button changes.

TESTING AND RESULT

'arduino_code.ino' is the name of my arduino code. The output should be connect to analog_0 part. 'ibrahim_kahraman.m' is my main matlab file, in this code i design filters and read the data from port that arduino sent. 'detection_button.m' is my detection function. We test the circuit and also the codes, we can succesfully detect which dual tone is used.