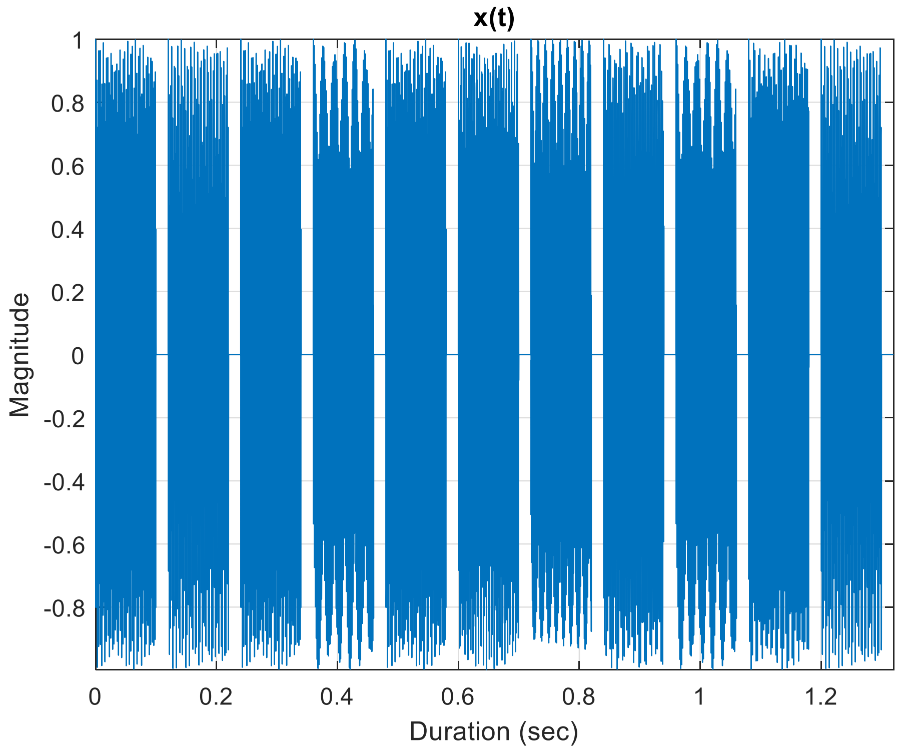
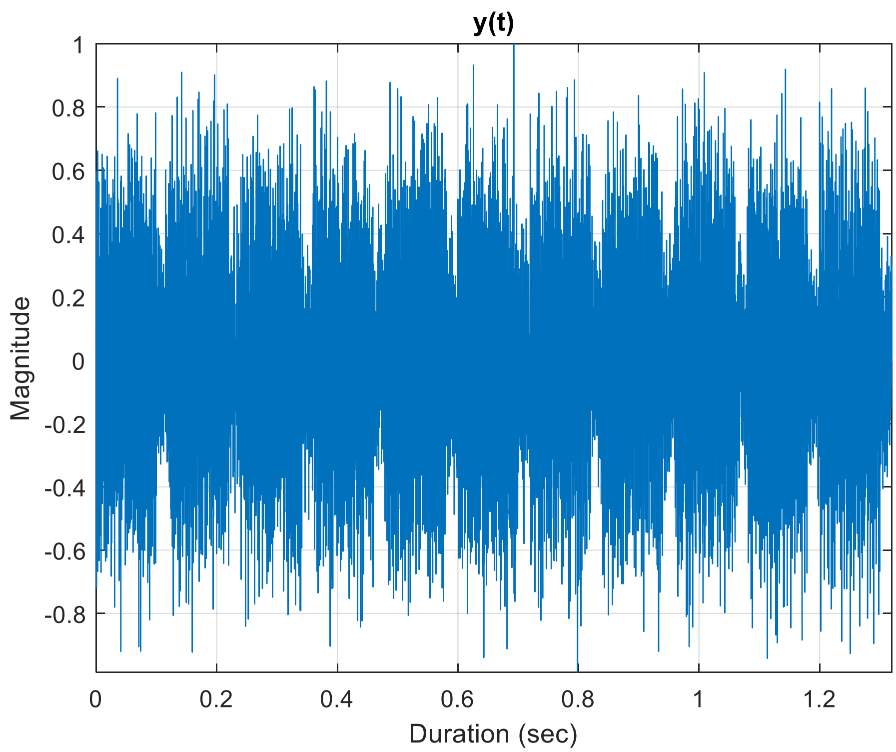
ECE 451s: Digital Signal Processing Basics Project

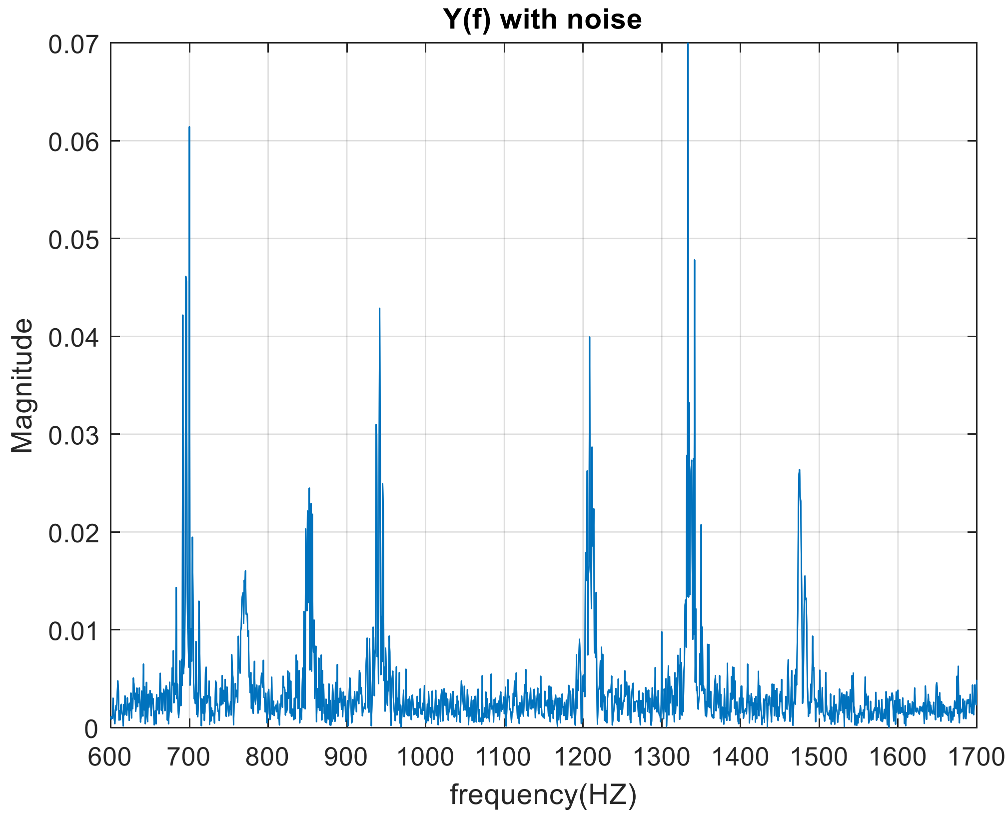
Submitted to:
Dr. Michael Naiem Abdelmassih Ibrahim
Eng. Omar Mohamed Eid Ahmed

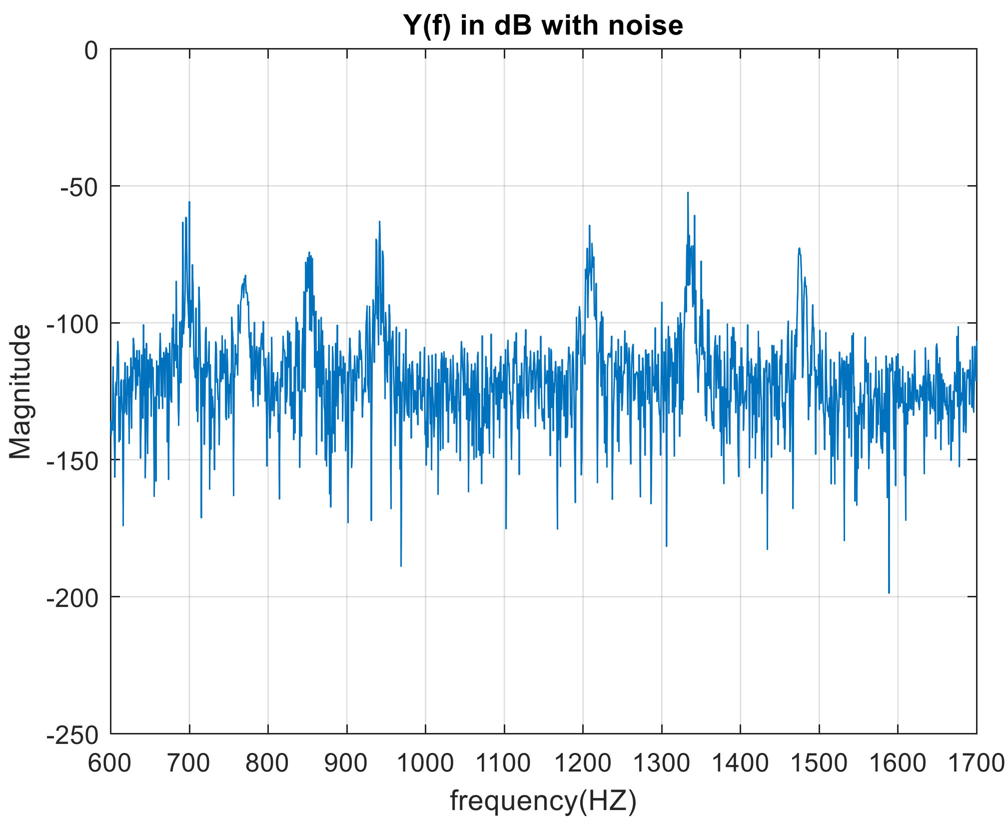
Sama Ahmed Mohammed Kamal	1805508
Sohila Ashraf Said Noureldin	1902226
Mariam Taher Eldin Mohamed	18Q4141
Shohrt Helmy Hussien Ahmed	1803632

- **Signal plots:**



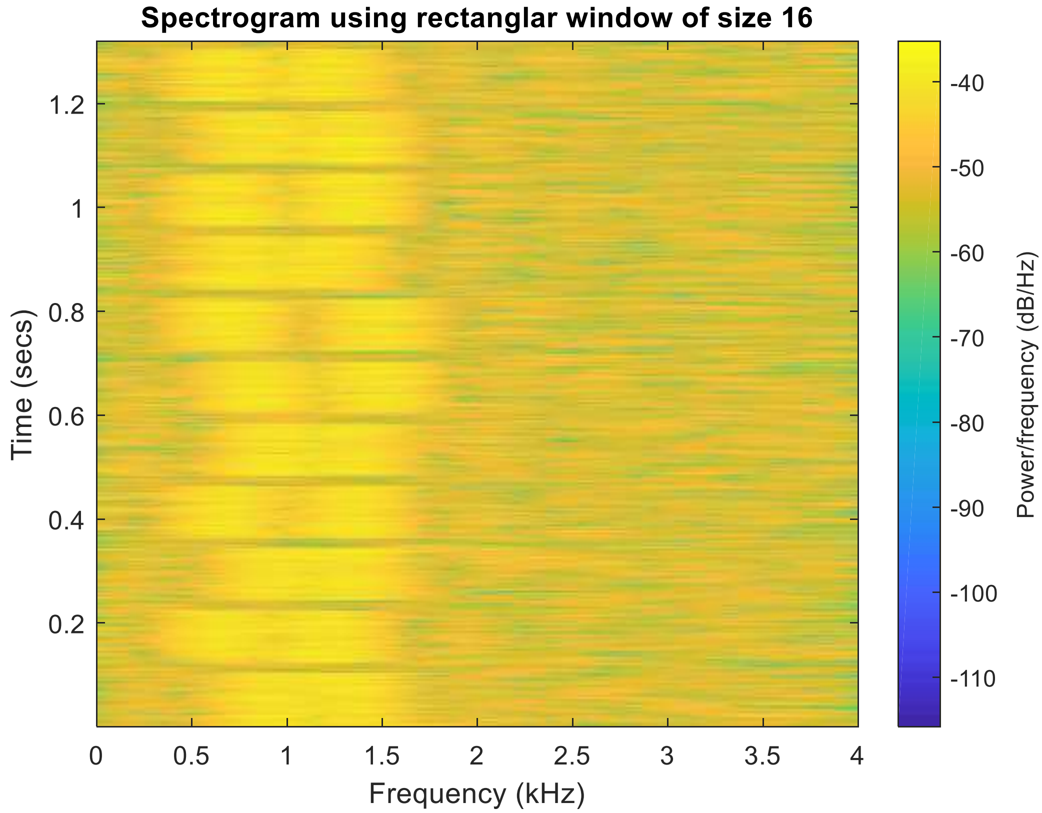


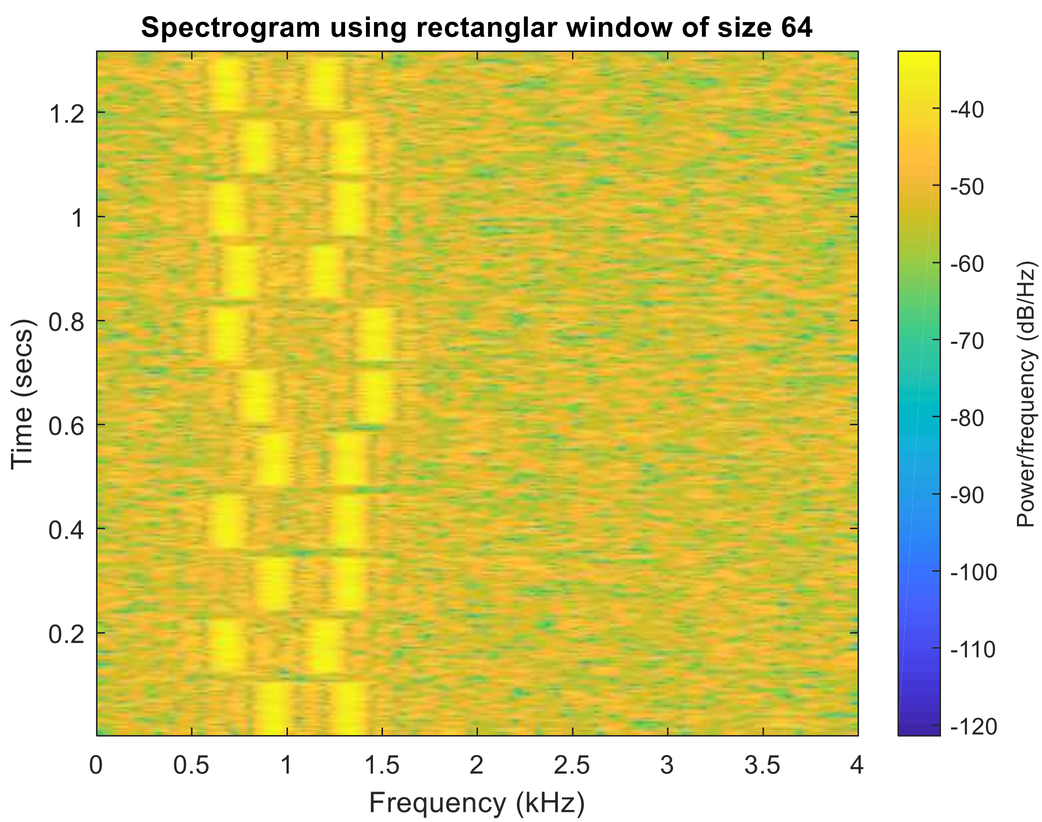

Y(f) with noise
Magnitude
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
600
700
800
900
1000
1100
1200
1300
1400
1500
1600
1700
frequency(HZ)

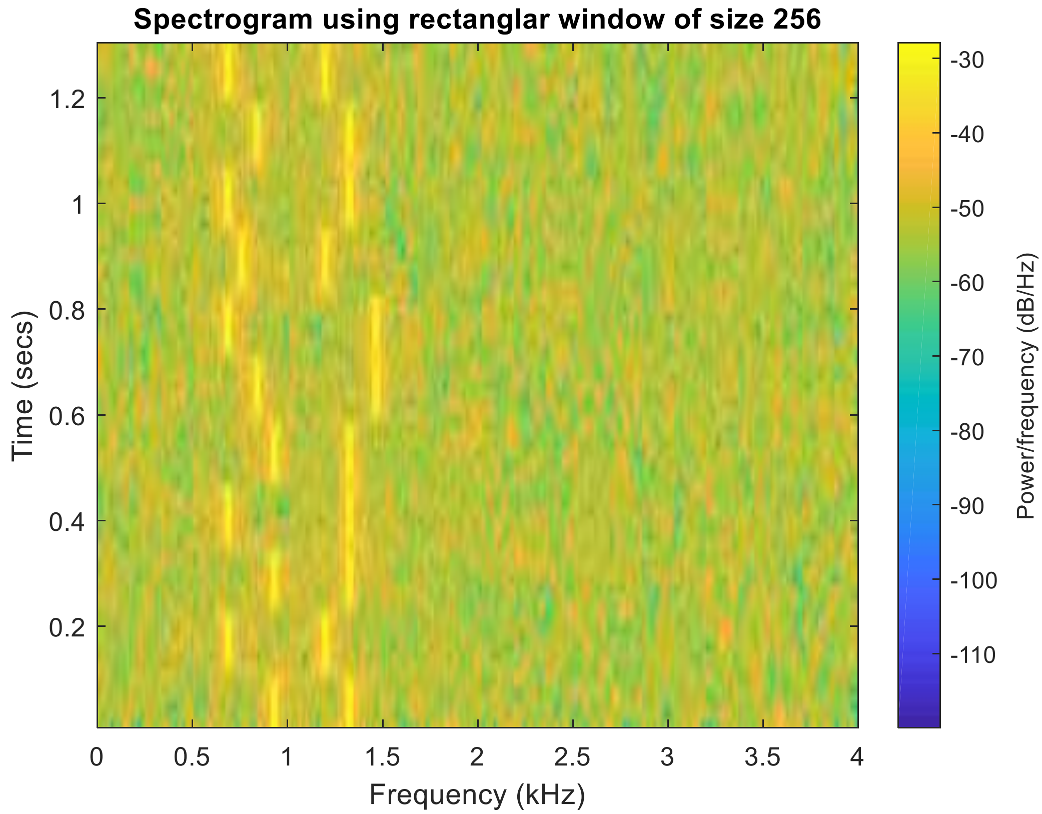

Y(f) in dB with noise
Magnitude
0
-50
-100
-150
-200
-250
600
700
800
900
1000
1100
1200
1300
1400
1500
1600
1700
frequency(HZ)

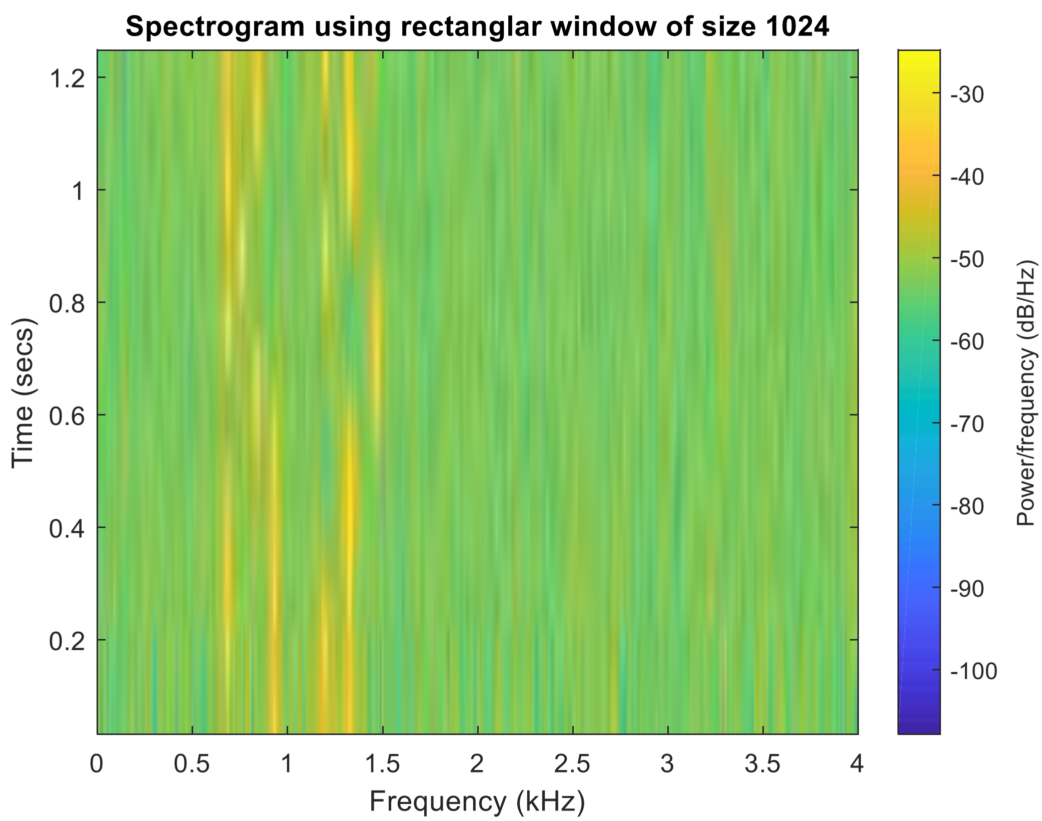
- ## Spectrogram Figures

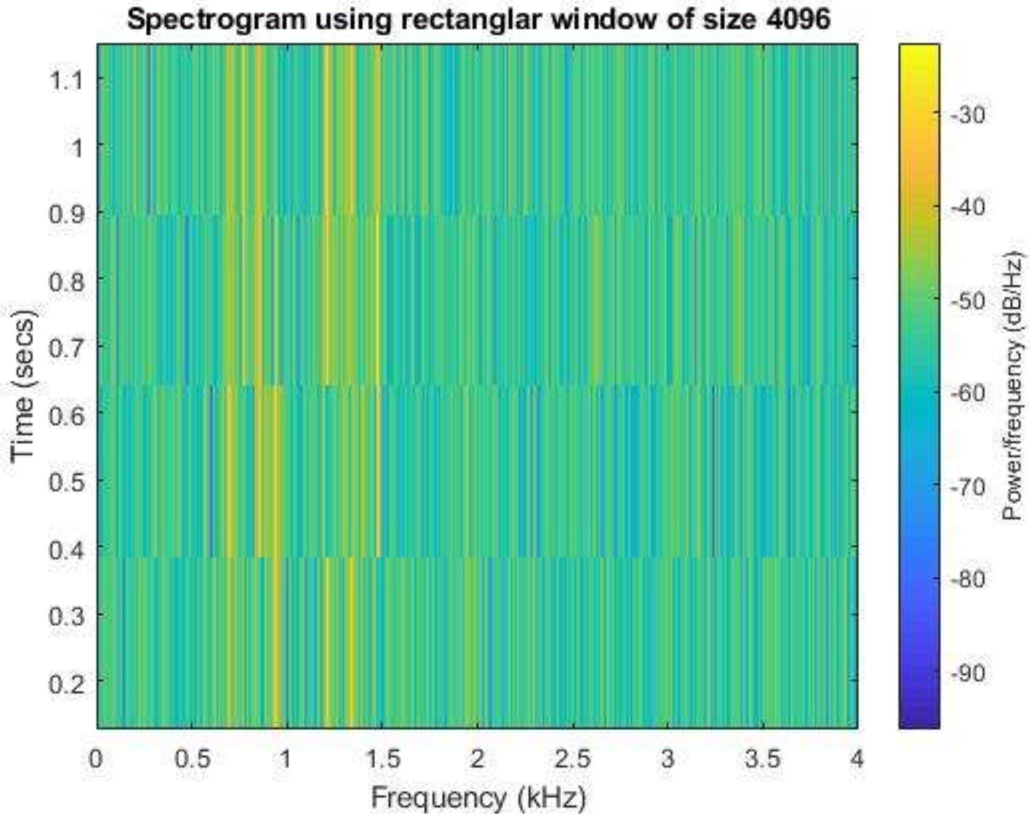
1. Using Rectangular window



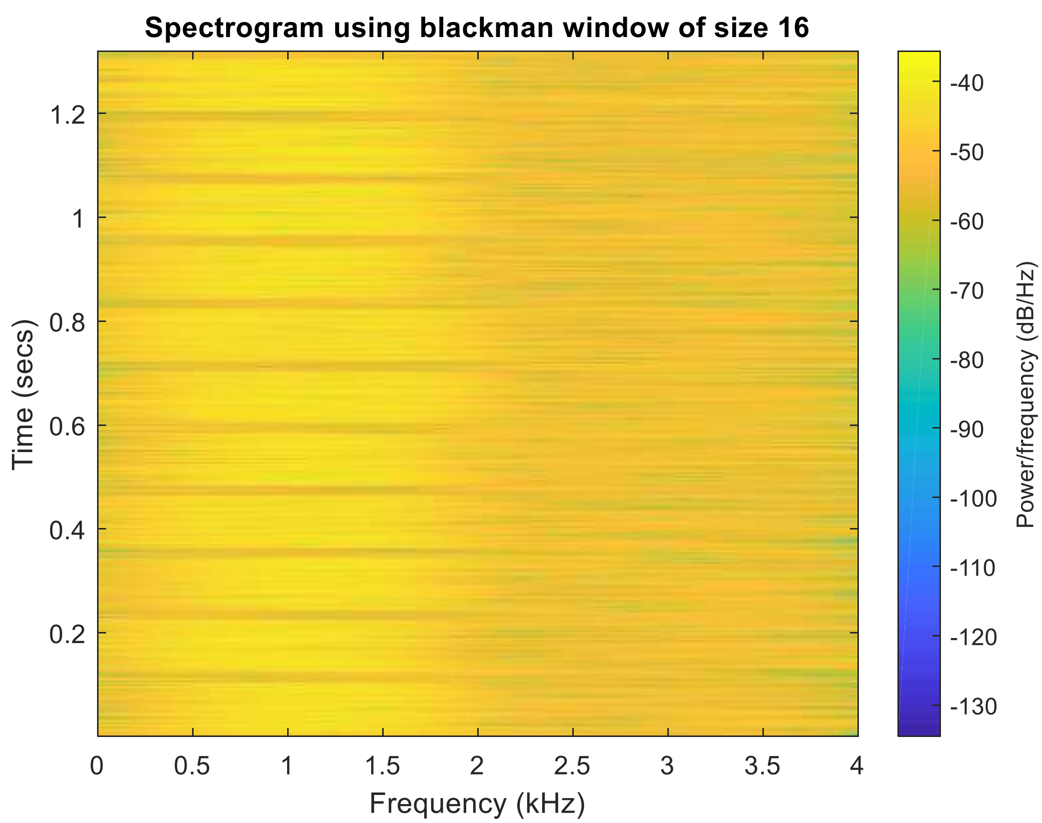


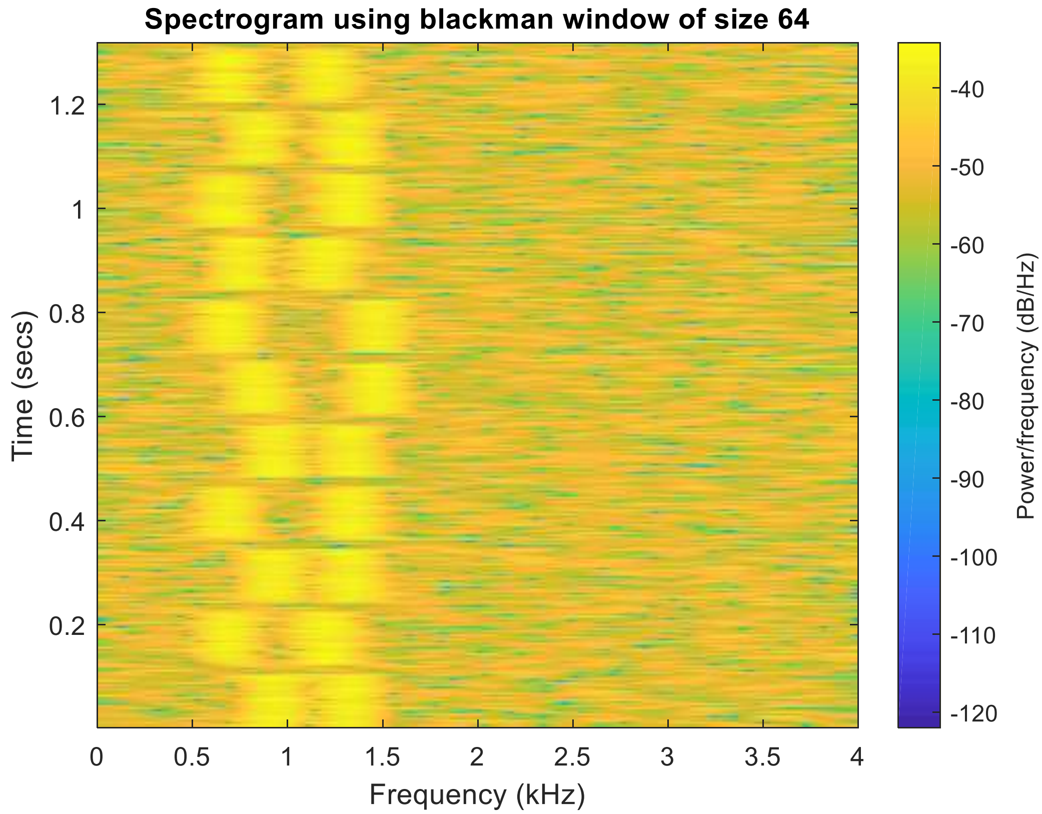

Spectrogram using rectanglar window of size 256
Time (secs)
1.2
1
0.8
0.6
0.4
0.2
0
0.5
1
1.5
2
2.5
3
3.5
4
Frequency (kHz)
-30
-40
-50
-60
-70
-80
-90
-100
-110
Power/frequency (dB/Hz)


Spectrogram using rectanglar window of size 1024
Time (secs)
1.2
1
0.8
0.6
0.4
0.2
0
0.5
1
1.5
2
2.5
3
3.5
4
Frequency (kHz)
-30
-40
-50
-60
-70
-80
-90
-100
Power/frequency (dB/Hz)

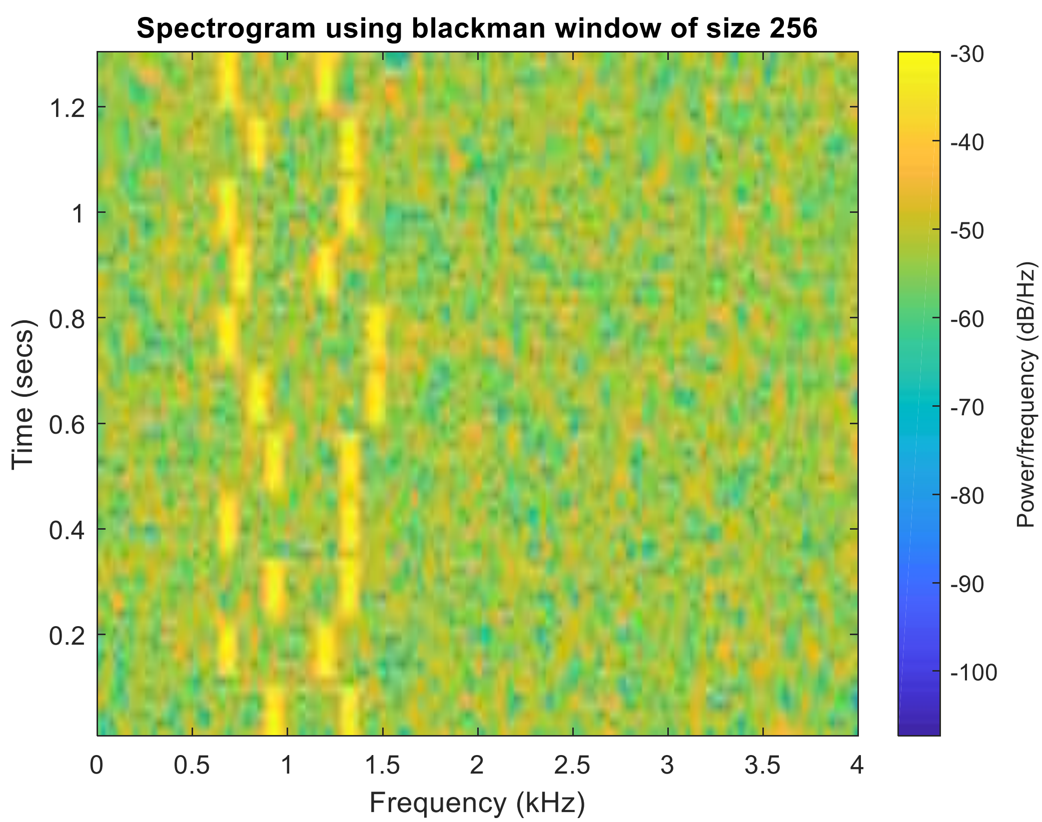


2. Using Blackman window

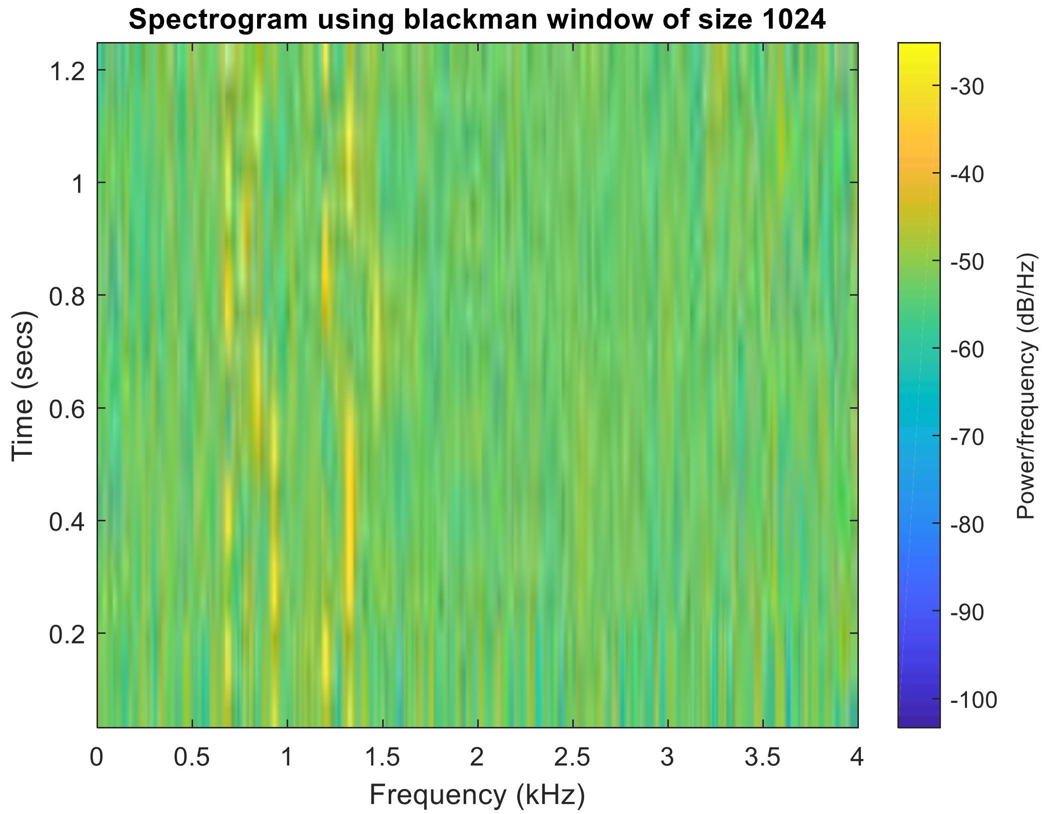


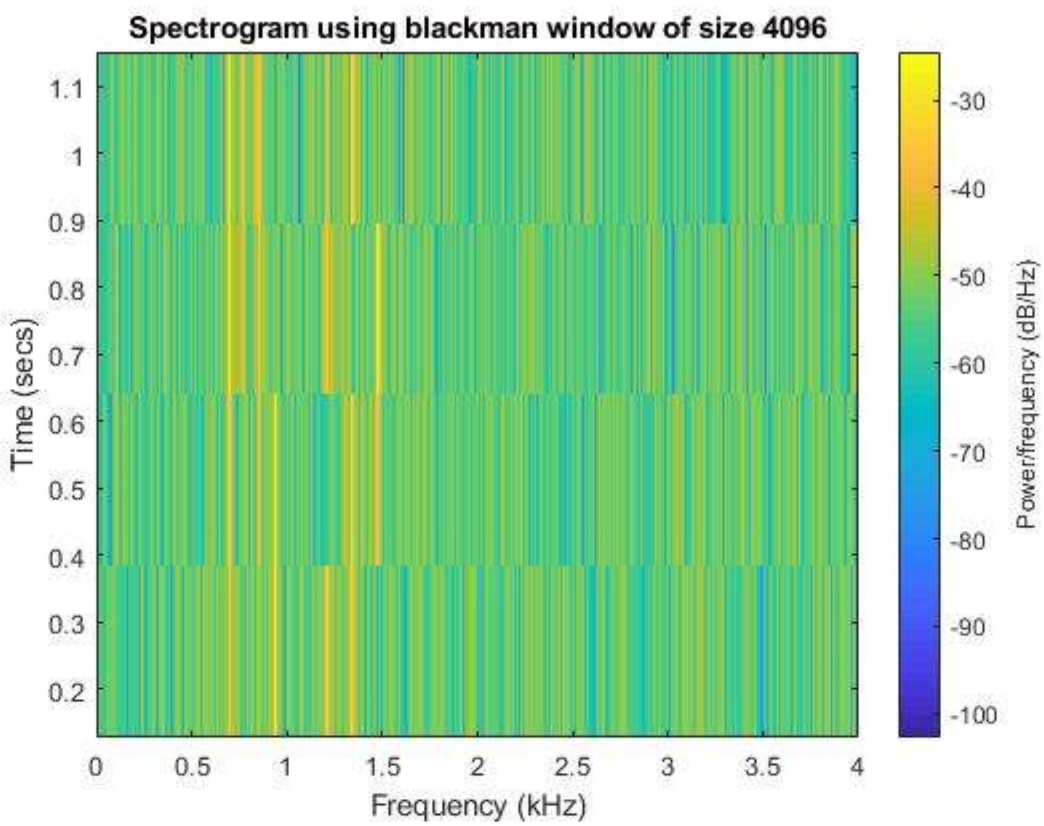


Spectrogram using blackman window of size 64
Time (secs)
1.2
1
0.8
0.6
0.4
0.2
0
0.5
1
1.5
2
2.5
3
3.5
4
Frequency (kHz)
-40
-50
-60
-70
-80
-90
-100
-110
-120
Power/frequency (dB/Hz)



Spectrogram using blackman window of size 256
Time (secs)
1.2
1
0.8
0.6
0.4
0.2
0
0.5
1
1.5
2
2.5
3
3.5
4
Frequency (kHz)
-30
-40
-50
-60
-70
-80
-90
-100
Power/frequency (dB/Hz)


Spectrogram using blackman window of size 1024
Time (secs)
1.2
1
0.8
0.6
0.4
0.2
0
0.5
1
1.5
2
2.5
3
3.5
4
Frequency (kHz)
Power/frequency (dB/Hz)
-30
-40
-50
-60
-70
-80
-90
-100


Spectrogram using blackman window of size 4096
Time (secs)
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
0.5
1
1.5
2
2.5
3
3.5
4
Frequency (kHz)
Power/frequency (dB/Hz)
-30
-40
-50
-60
-70
-80
-90
-100

3. **Which window size provides the worst time-domain resolution?**

   Worst time-domain resolution is provided by size 4096
   As window size increases the time domain resolution gets worse because different pulses interfere with each other.

4. **Which window size provides the worst frequency-domain resolution?**

   Worst frequency-domain resolution is provided by size 16
   As window size decreases, its Sinc frequency domain gets wider (getting away from the ideal case which is an impulse) so the frequency domain resolution gets worse.
   Also note that: any window size greater than 800 (number of samples of each tone, or 960 taking the guard interval into account) would give a very bad frequency domain resolution because we have more than one tone at the same time window.

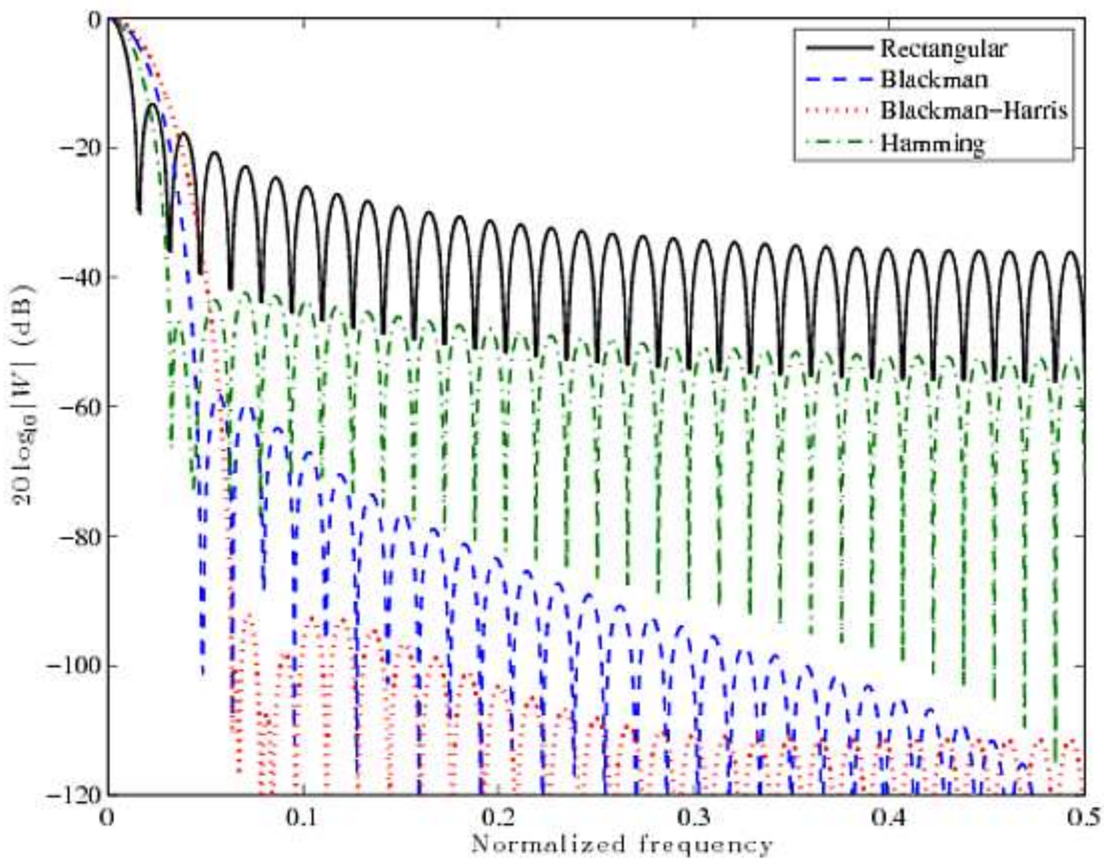
5. **Which window size provides a "kind-of" optimal trade-off between time and frequency resolutions?**

   The optimum tradeoff window size is 256, it’s an intermediate size which is not very low destroying the frequency domain resolution and at the same time not very high destroying the time domain resolution.

6. **which window type (rectangular or blackman) provides a better frequency resolution?**

   The rectangular window is better.
   Explanation: the frequency domain of the rectangular window has a narrower main lobe width than that of the blackman window, which makes it more selective for our case (the tones are quite close in frequency).

Window Function	$0 \leq n \leq N_w - 1$	First Null Beam Width	Sidelobe Peak Relative to Mainlobe
Rectangular $w_R(n)$	1	$2/N_w$	−13.5 dB
Blackman $w_B(n)$	$0.42 - 0.5\cos\left(\frac{2\pi n}{Nw}\right) + 0.08\cos\left(\frac{4\pi n}{Nw}\right)$	$6/N_w$	−57 dB



7. Overview on the Goertzel function

Goertzel algorithm analyses the spectrum only at the specific set of frequencies thus is more suitable for systems like DTMF in order to save memory as only eight DTMF frequencies need to be calculated for this application, and the Goertzel algorithm can calculate selected frequencies, this saves computation time.
The Goertzel algorithm is used for each DTMF frequency to determine the frequency at which the incoming signal has maximum energy. Since maximum energy corresponds to DTMF frequency, this procedure enables us to detect the DTMF frequency thus obtaining the original phone number.

The simple structure of the Goertzel algorithm makes it more favorable and efficient than FFT in computing small number of selected frequency components. It’s essentially based on DFT but with nearly double the efficiency which is why it’s suitable for checking the presence of certain frequencies in a signal.

For a stable and causal system with an input sequence, $x[n]$, and output sequence, $y[n]$, we can implement Geortzel function using 2 simple cascaded stages:

1. IIR digital filter:

$$s[n] = x[n] + 2\cos(\omega_0)\, s[n-1] - s[n-2]$$

Where $s[n]$ is an intermediate sequence.

2. FIR digital filter:

$$y[n] = s[n] - e^{-j\omega_0}\, s[n-1]$$

What's special about these stages is that the first filtering stage applied directly to the input sequence uses only real numbers. Regarding the second stage, we are only concerned with the final value of y which means that the filter will be evaluated just once which, in return, boosts the efficiency of computation.

We can apply Z-transform to evaluate the transfer function of Geortzel algorithm and we get:

$$\frac{Y(z)}{X(z)} = \frac{1}{1 - e^{j\omega_0} z^{-1}}$$

And then transform it back to the time-domain:

$$y[n] = e^{j\omega_0 n} \sum_{k=0}^{n} x[k] e^{-j\omega_0 k}$$