



به نام خدا



دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

پردازش سیگنال‌های زمان‌گسسته

گزارش پروژه ۲

فاطمه صالحی	نام و نام خانوادگی
۸۱۰۱۹۸۴۲۳	شماره‌ی دانشجویی

بخش اول: تبدیل فوریه زمان کوتاه :

۱- مقایسه سیگنال کانال های ۱ و ۲:

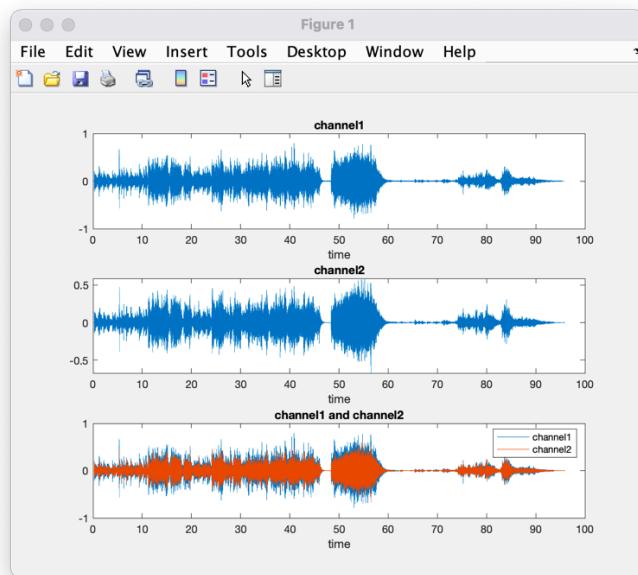


Figure 1.1 – comparing the signals of both channels

همانطور که در Figure 1.1 میبینیم، سیگنال ها تقریباً یکسان هستند و صرفاً دامنه آنهاست که متفاوت است که احتمالاً میتواند هنگام write شدن سیگنال ها در متلب این اختلاف بوجود آمده باشد، و یا بر روی یکی از کانال ها به صورت خاص نویزی افتاده باشد.

۲- مقایسه تبدیل فوریه سیگنال کانال های ۱ و ۲:

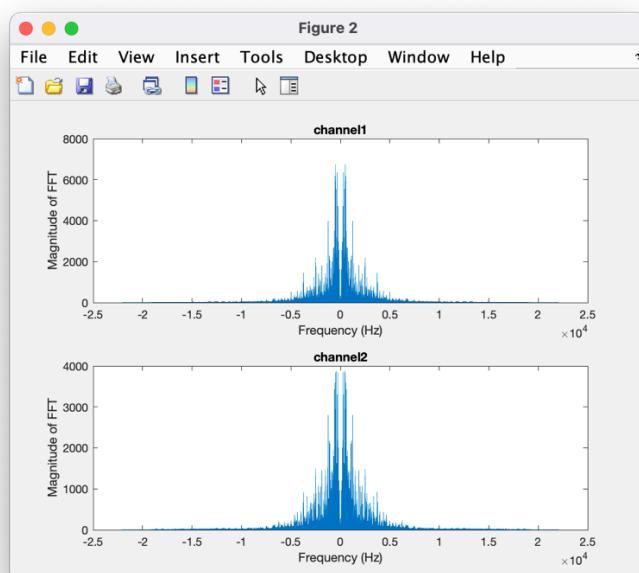


Figure 1.2 – comparing the Fourier transport of signals of both channels

همانطور که توقع داشتیم پهانی باند این ۲ سیگنال تقریباً یکسان بوده و تقریباً برابر 20 kHz میباشد، و تفاوتی که مشاهده میشود در دامنه این ۲ تبدیل فوریه است که با بخش قبل همخوانی دارد، چون هر چه دامنه تبدیل فوریه بیشتر باشد، دامنه سیگنال در بعد زمان نیز بیشتر میشود.

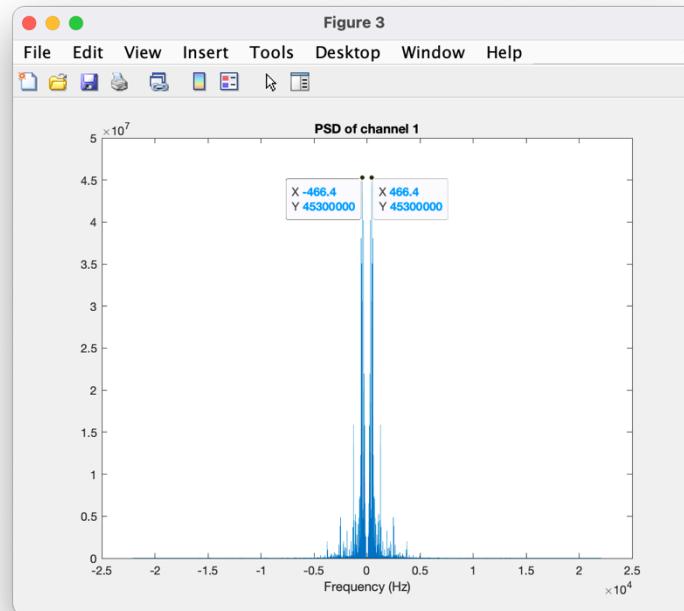


Figure 1.3 – PSD of channel 1

با توجه به Figure 1.3 ، فرکانس اصلی این سیگنال 466.4Hz میباشد، و فرکانس های قوی دیگر نیز در همسایگی این فرکانس میباشند

۳- تابع **spectrogram** :

برای این قسمت از تابع مذکور با آرگومان های زیر استفاده شده است.

```
[s,f,t] = spectrogram(x,window,noverlap,f,fs)
```

که در کد از f طرف نظر شده و در واقع یه وکتور خالی به عنوان ورودی داده شده، با این کار بررسی فرکانس ها موجود در لحظه راحت تر بود fs نیز فرکانس نمونه برداری میباشد.

با کاهش سایز $window$ ، باعث افزایش $resolution$ در هر لحظه میشویم، در واقع انگار از هر تبدیل فوریه میگریم و آن را پلات میکنیم، هرچه پنجره کوچک تر باشد ، نمودار رسم شده فرکانس را در لحظه بیشتر نشان میدهد و دقت افزایش میابد.

همانطور که در Figure 1.4 دیده میشود، با افزایش $window size$ دقت در هر لحظه کاهش یافته است و با افزایش $noverlap$ که تعداد خانه های همپوشانی هر دو $window$ همسایه میباشد تاثیر به سزایی در کیفیت نداشته است.

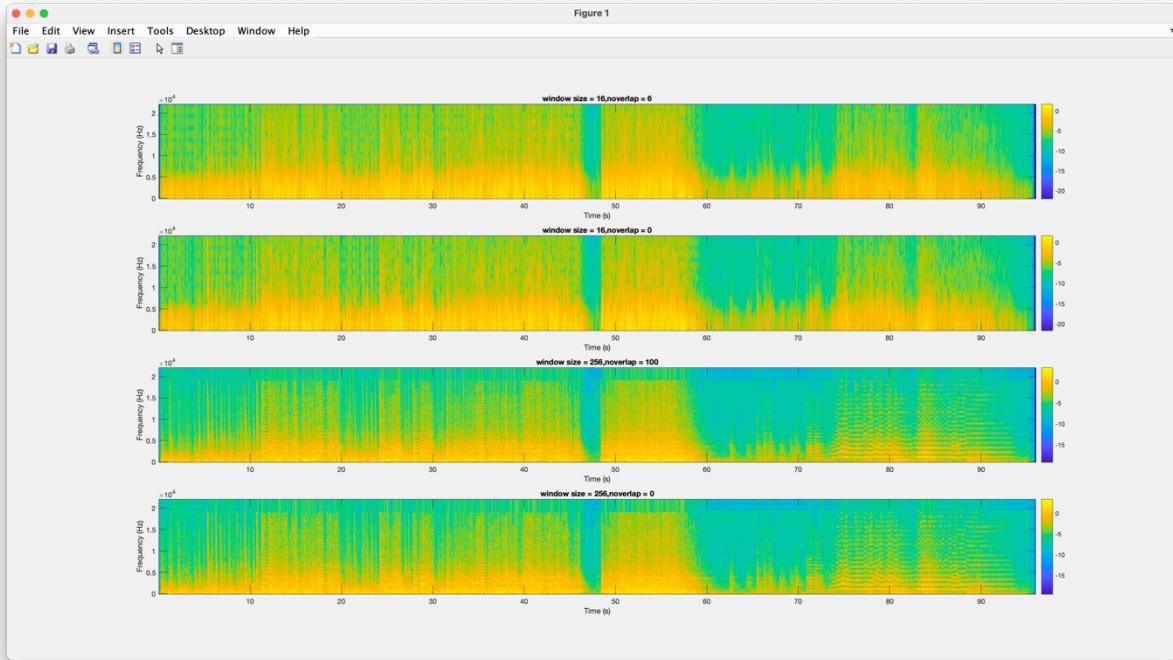


Figure 1.4 – comparing the operation of the spectrogram function with different window_size and nooverlap values.

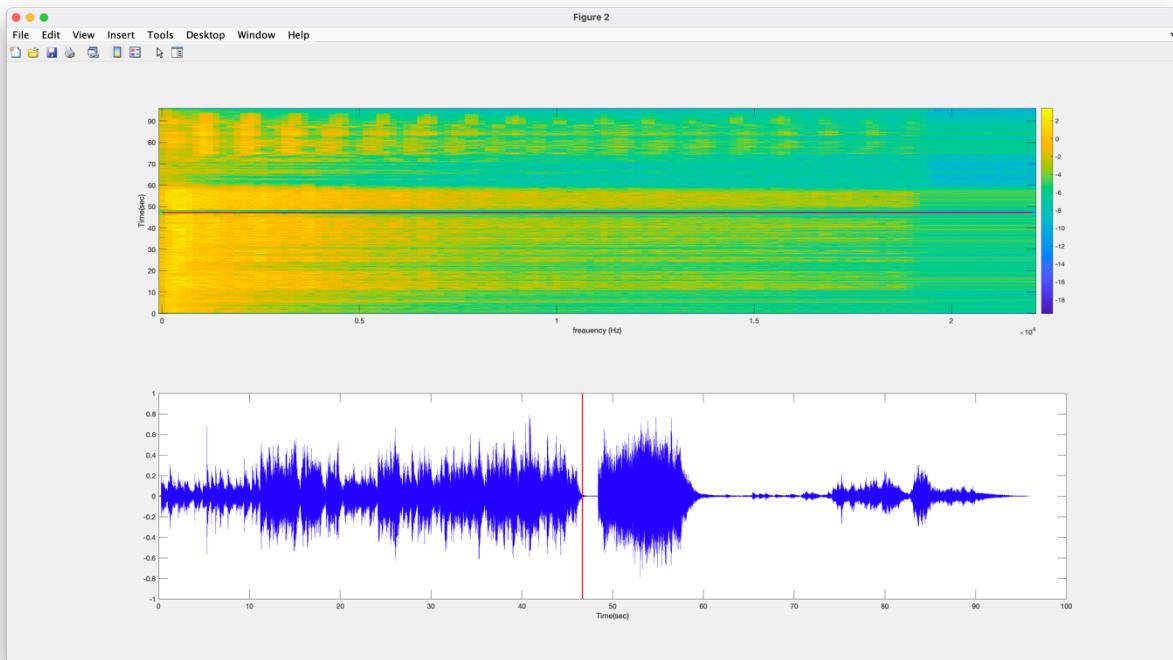


Figure 1.5 – signal of the channel 1 and its spectrum.

در فایل Part1.m ، قسمتی به اسم signal of the channel 1 and its spectrum وجود دارد که سیگنال فایل صوتی play شده در هر لحظه، دامنه سیگنال و فرکانس موجود در هر لحظه با خط قرمز نشان داده میشود؛ در Figure 1.5 لحظه ای که صدا قطع شده است را نشان میدهد که در طیف آن نیز مشاهده میشود که فرکانس صفر میباشد.

۴- علت وجود ۲ کanal: علت این امر میتواند بخاطر این باشد که این فایل mp3. قرار است توسط یک دستگاه با ۲ خروجی، مانند هندزفری، پخش شود و یا اینکه اطلاعات این فایل به ۲ قسمت تقسیم شده و یک قسمت از کanal اول و قسمت دیگر توسط کanal دوم پخش شود.

۵- سوال مربوط به تبدیل موجک:

طبق قوانین فیزیک میدانیم که نمیتوانیم بدانیم که چه فرکانس های در هر لحظه در یک سیگنال بوجود می آیند، ولی میتوانیم بدانیم که محدوده ای از فرکانس ها در یک بازه زمانی وجود دارند، همچنین نامساوی زیر را داریم:

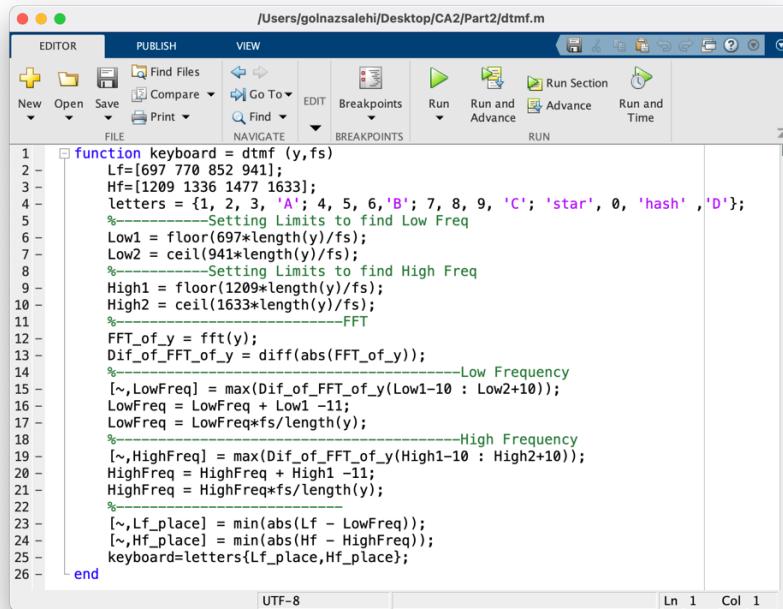
$$\Delta t \Delta f \geq \frac{1}{4\pi}$$

هرچه فرکانس موجک بیشتر باشد، در یک بازه زمانی مشخص اطلاعات بیشتری از موجک را نسبت به موجک فرکانس های پایین تر داریم، پس در با اطمینان بیشتری میتوانیم بگوییم که آن فرکانس در بازه زمانی وجود دارد یا نه، همچنین چون تغییرات موجک با فرکانس بالا زیاد است، نمیتوان در کل نظر داد که اندازه این فرکانس در کل سیگنال چقدر است، پس برای موجک با فرکانس بالا رزولوشن زمانی بالا و رزولوشن فرکانسی پایینی داریم.

به طریق مشابه برای موجک با فرکانس پایین، رزولوشن زمانی پایین و رزولوشن فرکانسی بالایی داریم.

فایل مربوط با این بخش Part1.m میباشد .

بخش دوم: تشخیص صوت های DTMF



```
function keyboard = dtmf (y,fs)
1 - Lf=[697 770 852 941];
2 - Hf=[1209 1336 1477 1633];
3 - letters = {1, 2, 3, 'A'; 4, 5, 6, 'B'; 7, 8, 9, 'C'; 'star', 0, 'hash' , 'D'};
4 - %-----Setting Limits to find Low Freq
5 - Low1 = floor(697*klength(y)/fs);
6 - Low2 = ceil(941*klength(y)/fs);
7 - %-----Setting Limits to find High Freq
8 - High1 = floor(1209*klength(y)/fs);
9 - High2 = ceil(1633*klength(y)/fs);
10 - %-----FFT
11 - FFT_of_y = fft(y);
12 - Dif_of_FFT_of_y = diff(abs(FFT_of_y));
13 - %-----Low Frequency
14 - [~,LowFreq] = max(Dif_of_FFT_of_y(Low1-10 : Low2+10));
15 - LowFreq = LowFreq + Low1-11;
16 - LowFreq = LowFreq*fs/length(y);
17 - %-----High Frequency
18 - [~,HighFreq] = max(Dif_of_FFT_of_y(High1-10 : High2+10));
19 - HighFreq = HighFreq + High1-11;
20 - HighFreq = HighFreq*fs/length(y);
21 - %
22 - [~,Lf_place] = min(abs(Lf - LowFreq));
23 - [~,Hf_place] = min(abs(Hf - HighFreq));
24 - keyboard=letters{Lf_place,Hf_place};
25 -
26 - end
```

Figure 2.1 – dtmf function

شماره تلفن بدست آمده ۸۸۰۸۴۱۸۰ میباشد.

فایل مربوط با این بخش Part2 میباشد.

بخش سوم: فشرده سازی تصویر با استفاده از تبدیل فوریه دو بعدی

در راستای y تناوب داریم بنابراین خط $y=0$ مقدار بیشتری نسبت به بقیه خطوط دارد (سفید است) ولی در راستای x این تناوب را نداریم، پس خط $x=0$ سیاه است. در واقع چون در این شکل نسبت به محور x تقارن داریم، در تبدیل فوریه نیز این تقارن حفظ میشود.

در راستای x و هم در راستای y در بعضی از سطرها تناوب داریم، و این فرکانس تناوب صفر میباشد پس خطوط $x=0$ و $y=0$ باید مقادیر بیشتری از بقیه خطوط داشته باشند. محور x و y تقارن داریم، در تبدیل فوریه نیز این تقارن حفظ میشود.

در Shape3، نسبت به دو شکل قبل تغییرات بیشتری دیده میشود بنابراین فرکانس های بیشتر و بزرگ تری برای تولید آن لازم است، همچنین این شکل نسبت به قطرهای اصلی و فرعی متقاض میباشد، بنابراین در تبدیل فوریه نیز چنین میباشد.

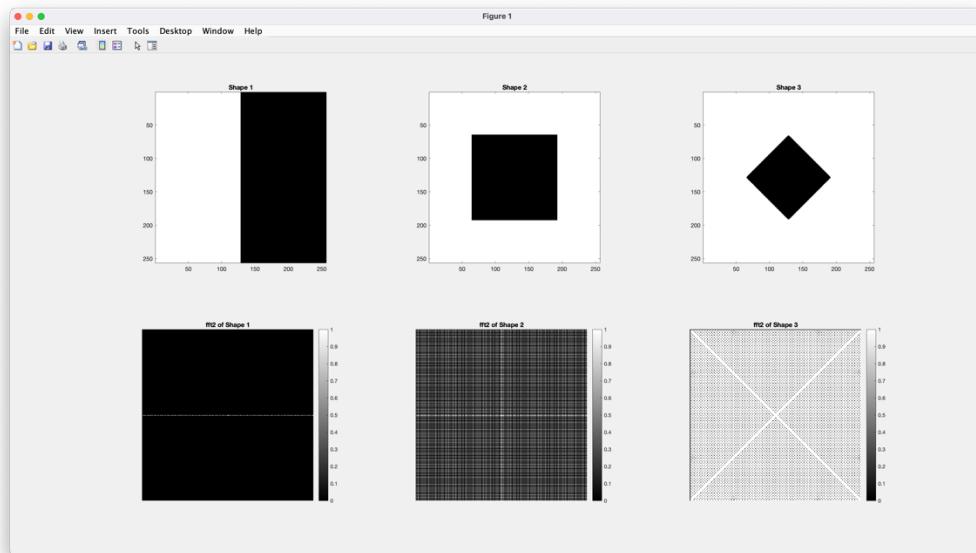


Figure 3.1 – comparing the Fourier transform of the Shape 1, 2, 3

کاهش حجم تصویر :

برای کاهش حجم، ابتدا از تبدیل فوریه میگیریم، سپس آن را به شکل یک وکتور در می آوریم و هر چند در صد اول آن را در نظر میگیریم و توسط آنها یک **threshold** تعریف میکنیم و مقدار تمام فرکانس هایی که از آن کوچک تر باشد را صفر میکنیم.

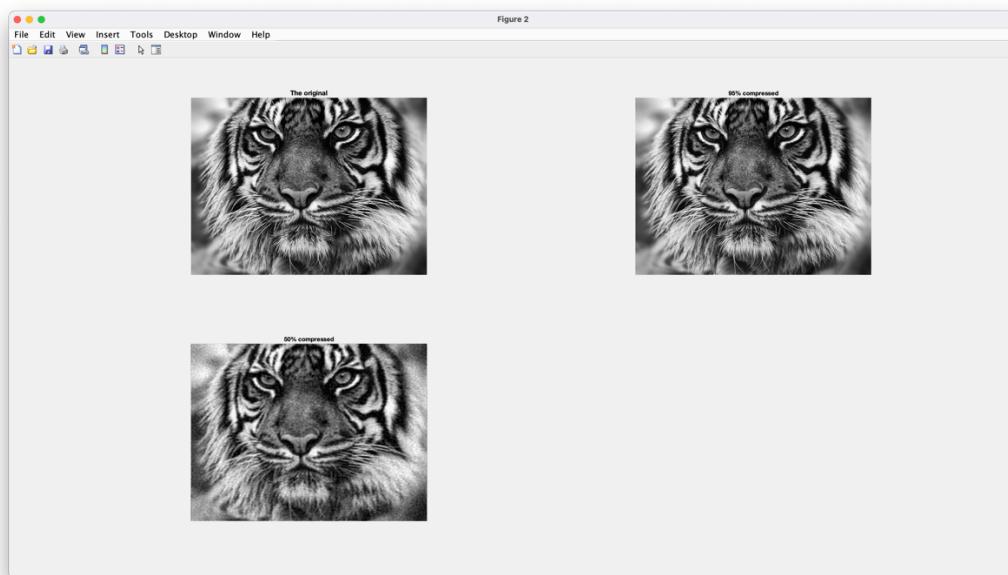


Figure 3.2 – comparing 95%, 50% compressed pictures

- چرا از **fft** نمیتوان برای کاهش حجم تصاویر رنگی استفاده کرد؟

تصاویر رنگی شامل ۳ قرمز، سبز و آبی بُعد میباشند و **fft** فقط میتواند در یکی از آن بُعد تبدیل فوریه را بدست آورد، اگر سیاست گفته شده را برای هر یک از بعد ها پیاده کنیم، احتمال اینکه تعادل رنگ ها بهم بخورد وجود دارد، بنابراین نمیتوان از این روش برای کاهش حجم تصاویر رنگی استفاده کرد.

فایل مربوط با این بخش **Part3.m** میباشد.

بخش چهارم: تحلیل فرکانسی سیگنال های مغزی :

۱- رسم سه سیگنال در حوزه زمان و فرکانس:

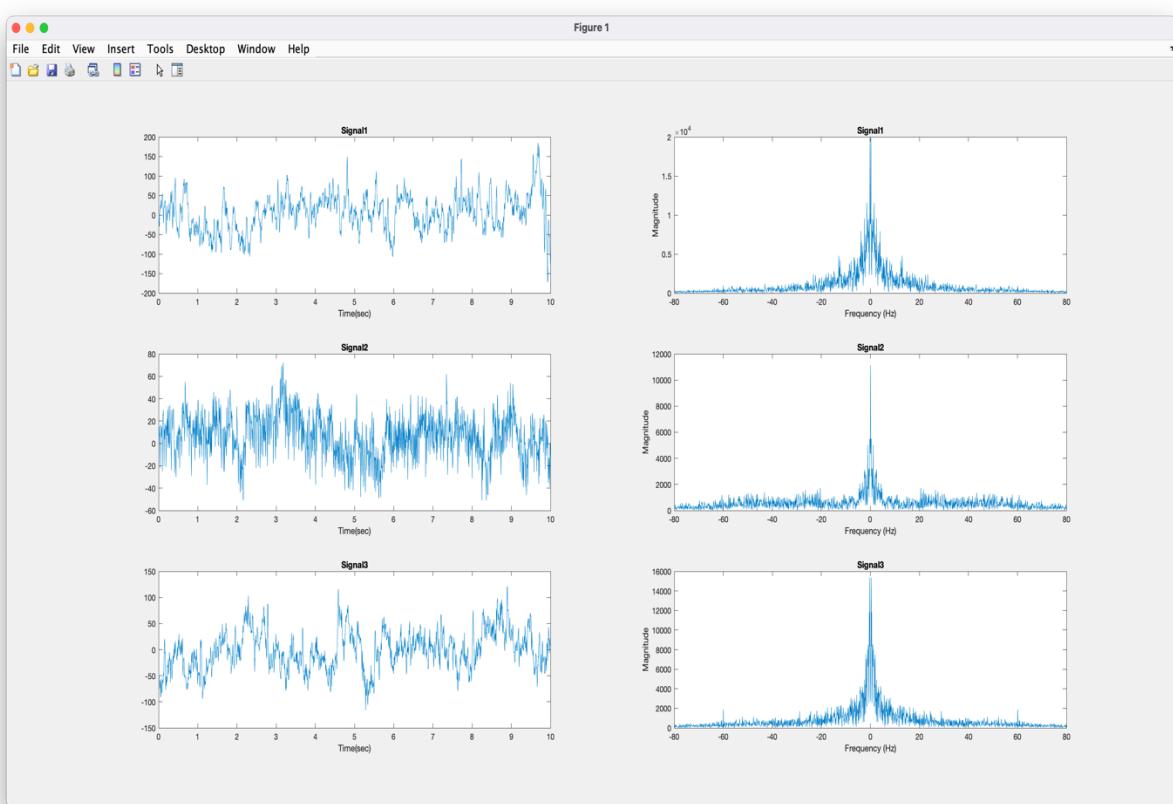


Figure 4.1 – time and frequency domain of the selected signals

۲- رابطه ای بین دامنه‌ی سیگنال مغزی و فرکانسی

با توجه به Figure 4.1 ، حدس زده می‌شود که هرچه فرکانس یک سیگنال مغزی بالاتر باشد، دامنه آن کمتر خواهد بود.

۳- پیش‌بینی امواج آلفا، بتا، دلتا، تتا و گاما برای سیگنال اول و تحلیل آنها:

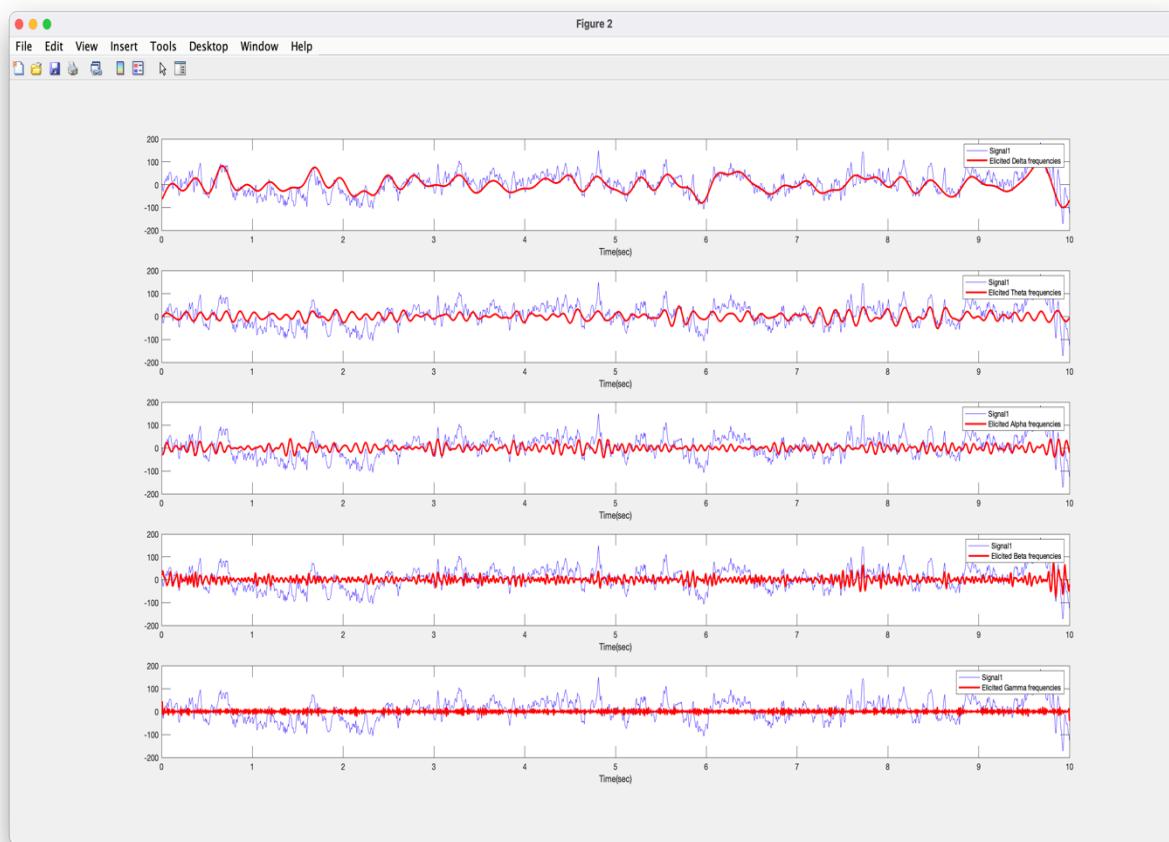
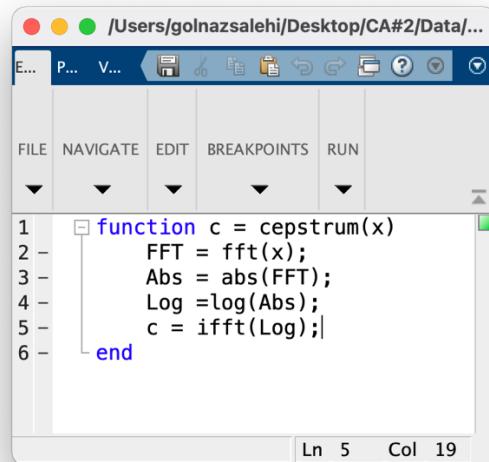


Figure 4.2 – Exploring the first signal

طبق Figure 4.2 ، بنظر می‌آید که سیگنال اول ترکیبی از سیگنال دلتا و تتا می‌باشد، و تطابق چندانی با فرکانس‌های گاما، آلفا و بتای خود ندارد پس میتوان نتیجه گرفت که حالت کیس مورد بررسی چیزی بین Deep Meditation و sleep بوده است. این بین سه سیگنال انتخاب شده دامنه سیگنال اول از بقیه بیشتر می‌باشد، و بدست آوریم که این سیگنال ترکیبی از دلتا و تتا می‌باشد، پس بنظر می‌آید که حدسی که در قسمت دوم زدا شد صحیح می‌باشد.

فایل مربوط با این بخش Part4.m می‌باشد.

بخش پنجم: آشنایی با کپسکتروم



The screenshot shows a MATLAB code editor window. The title bar indicates the file path: /Users/golnazsalehi/Desktop/CA#2/Data/. The menu bar includes FILE, NAVIGATE, EDIT, BREAKPOINTS, and RUN. The code area contains the following MATLAB function:

```
function c = cepstrum(x)
    FFT = fft(x);
    Abs = abs(FFT);
    Log = log(Abs);
    c = ifft(Log);|
end
```

The cursor is positioned at the end of the fifth line. The status bar at the bottom right shows "Ln 5 Col 19".

Figure 5.1 – ceptrum function

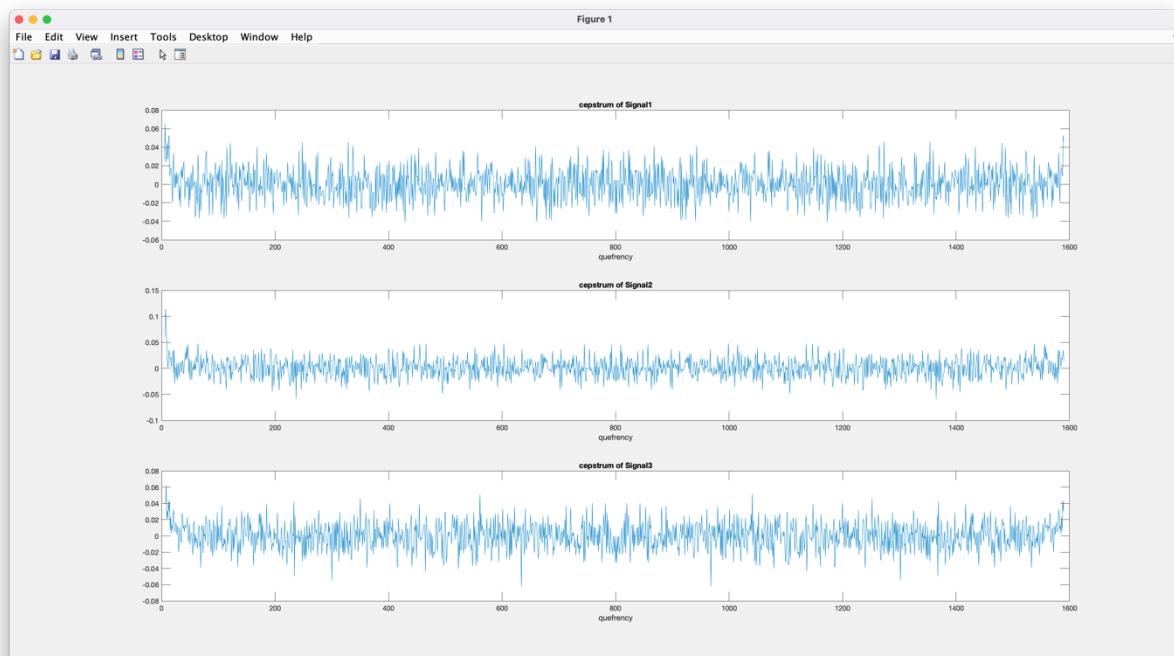


Figure 5.2 – ceptrum of three selected signals

فایل مربوط با این بخش **Part5.m** میباشد .