

سیگنال و سیستم ها

پروژه 5

سهیل حاجیان منش

بخش اول) تمرین 1-1:

کد مربوط به رسم نمودار سیگنال و رسم نمودار اندازه سیگنال در حوزه سری فوریه را در زیر آورده ام.

```
% PART A
t_start=-1;
t_end=1;
fs=50;
ts=1/fs;
t=t_start:ts:t_end-ts;
x=cos(10*pi*t);
subplot(2,1,1);
plot(t,x);
xlabel("Time(s)");
ylabel("cos(10pi*t)");
xlim([-0.6 0.6]);
title("countinous signal");

% PART B
N=length(t);
fourier_fs=-fs/2:fs/N:fs/2-fs/N;
y=fftshift(fft(x));
subplot(2,1,2);
plot(fourier_fs,abs(y)/max(abs(y)));
xlabel("Frequency(Hz)");
ylabel("amplitude");
title("signal amplitude in fourier");
```

محاسبات تیوری مربوط به محاسبه سری را مطابق زیر انجام می دهیم:

Sunday 27 Jun. 2010

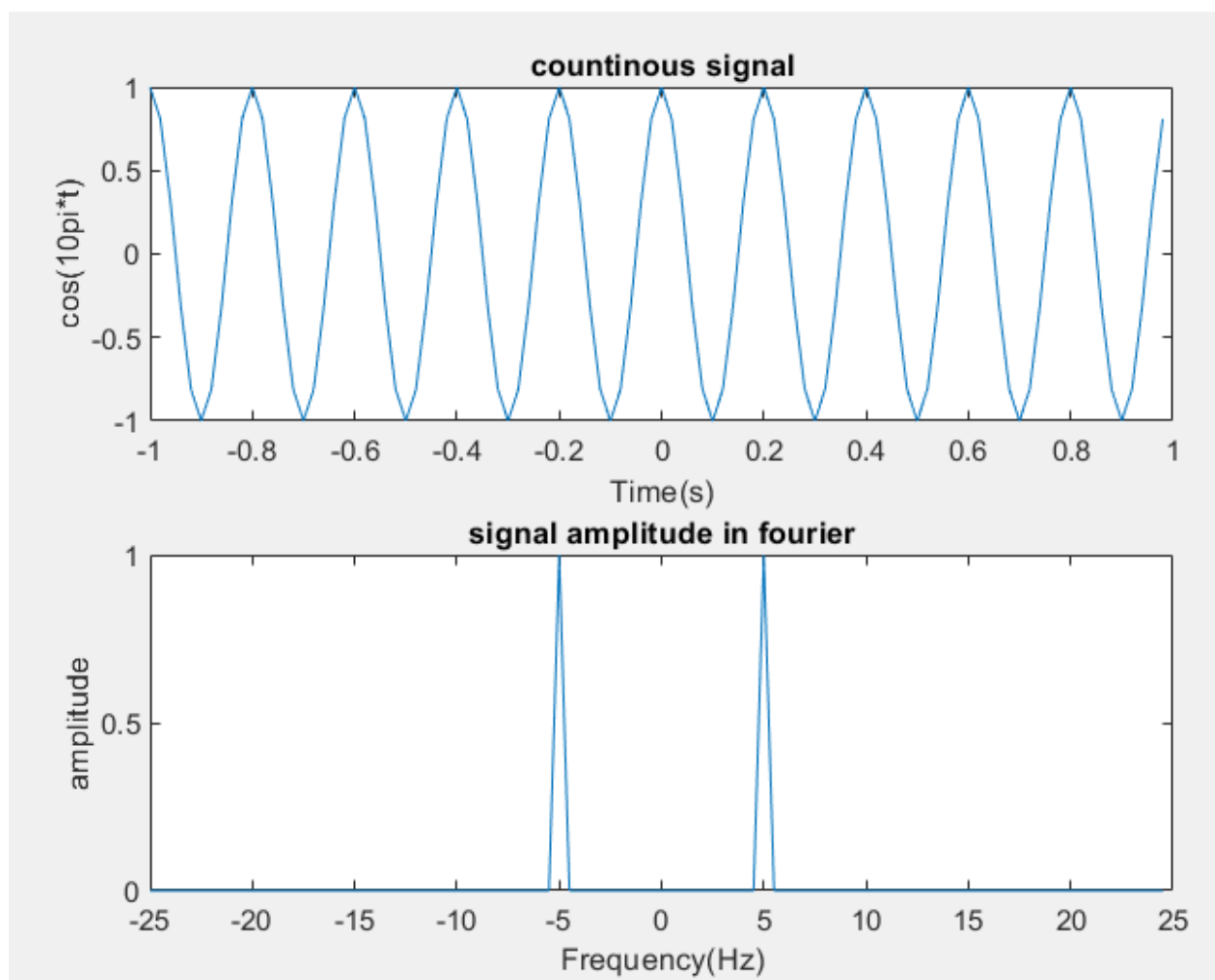
$$\cos(10\pi t) = \frac{1}{2} e^{j10\pi t} + \frac{1}{2} e^{-j10\pi t} \quad f = \frac{\omega}{2\pi}$$

$$F_1 = \frac{10\pi}{2\pi} = 5, \quad F_2 = \frac{-10\pi}{2\pi} = -5$$

$$\mathcal{F} \rightarrow \pi(\delta(\omega - 10\pi) + \delta(\omega + 10\pi))$$

طبق محاسبات اندازه ضرایب سری فوریه این سیگنال باید در دو فرکانش 5 و -5 پیک داشته باشد که نتایج بدست آمده با محاسبات ما مطابقت دارد.

شکل نمودار این دو سیگنال در کنار هم بصورت زیر است:



تمرین 2-1:

در زیر کدهای مربوط به رسم نمودار سیگنال، نمودار اندازه سیگنال در حوزه فوریه و نمودار فاز سیگنال در حوزه فوریه را بترتیب نشان داده ام:

```
% PART A
t_start=0;
t_end=1;
fs=100;
ts=1/fs;
t=t_start:ts:t_end-ts;
x=cos(30*pi*t+pi/4);
subplot(3,1,1);
plot(t,x);
xlabel("Time(s)");
ylabel("cos(30pi*t+pi/4)");
xlim([0 1]);
title("countinuous signal");
% PART B
N=length(t);
fourier_fs=-fs/2:fs/N:fs/2-fs/N;
y=fftshift(fft(x));
subplot(3,1,2);
plot(fourier_fs,abs(y)/max(abs(y)));
xlabel("Frequency(Hz)");
ylabel("amplitude");
title("signal amplitude in fourier");
% PART C
tol=1e-6;
y(abs(y)<tol)=0;
theta=angle(y);
subplot(3,1,3);
plot(fourier_fs,theta/pi);
xlabel("Frequency(Hz)");
ylabel("Phase/\pi");
title("signal angle in fourier");
```

محاسبات تیوری مربوط به اندازه و فاز سیگنال در حوزه فوریه را مطابق زیر انجام می دهیم.

$$\cos(30\pi t + \frac{\pi}{4}) \xrightarrow{\mathcal{F}} \frac{1}{2} e^{j(30\pi t + \frac{\pi}{4})} + \frac{1}{2} e^{-j(30\pi t + \frac{\pi}{4})} \quad 12$$

$$\omega_1 = 30\pi \rightarrow f_1 = 15 \quad \omega_2 = -30\pi \rightarrow f_2 = -15 \quad 13$$

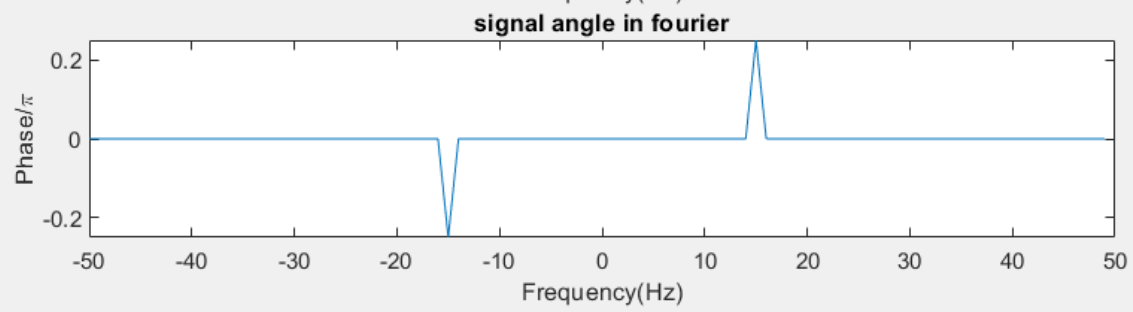
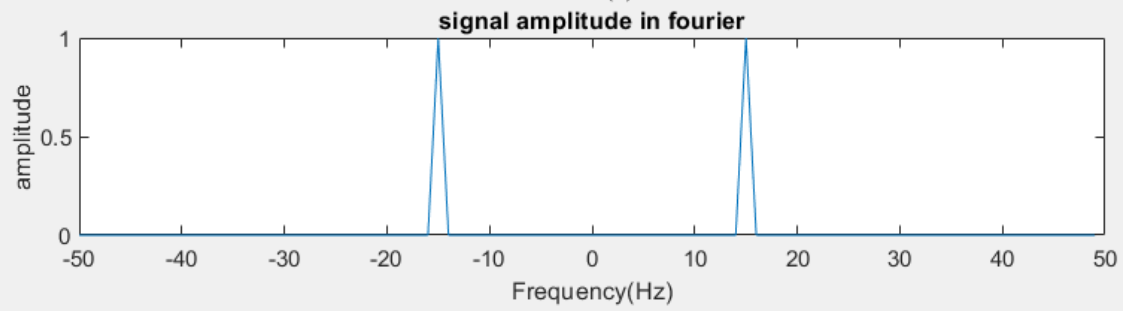
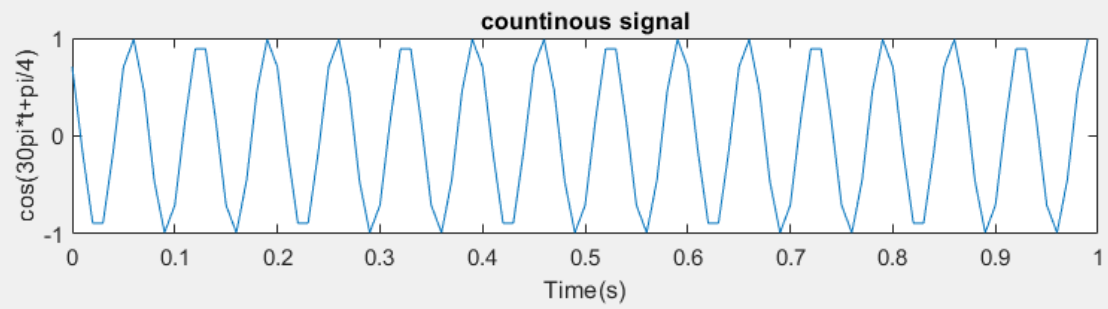
$$\angle x_1 = \frac{\pi}{4}, -\frac{\pi}{4} \rightarrow \text{Phase} \quad 14$$

$$\xrightarrow{\mathcal{F}} \sqrt{\frac{\pi}{2}} e^{-j\frac{\pi}{4}} \delta(\omega - 30\pi) + \sqrt{\frac{\pi}{2}} e^{j\frac{\pi}{4}} \delta(\omega + 30\pi) \quad 15$$

طبق محاسبات اندازه ضرایب تبدیل فوریه این سیگنال باید در دو فرکانس 15- و 15 پیک داشته باشد که با نمودار بدست آمده برای اندازه ضرایب مطابقت دارد.

همچنین طبق محاسبات تیوری فاز ضرایب این سیگنال باید در دو نقطه 15 و 15- برابر $\pi/4$ و $-\pi/4$ باشد که با نمودار بدست آورده مطابقت دارد. (تقسیم بر π شده است)

عکس سه نمودار بدست آمده برای قسمت الف، ب و ج را در زیر قرار داده ام.



بخش دوم) تمرین 1-2:

به مانند کد تمرین قبل مپ ست را می سازیم.

```
mapSet=cell(2,32);  
mapSet = {'a' , 'b' , 'c' , 'd' , 'e' , 'f' , 'g' , 'h' , 'i' , 'j' , 'k' ,  
for i=1:32  
    mapSet{2,i}=dec2bin(i-1,5);  
end
```

تمرین 2-2:

با کمک توضیحات گفته شده سیگنال کدگذاری شده را تشکیل می دهیم. سیگنال سینوسی بوده و بسته به تعداد قسمت های پیام که باید کدگذاری شود باید ضریب متفاوت تولید کرد. میتوان فرمول زیر را برای سیگنال کدگذاری شده تشکیل داد:

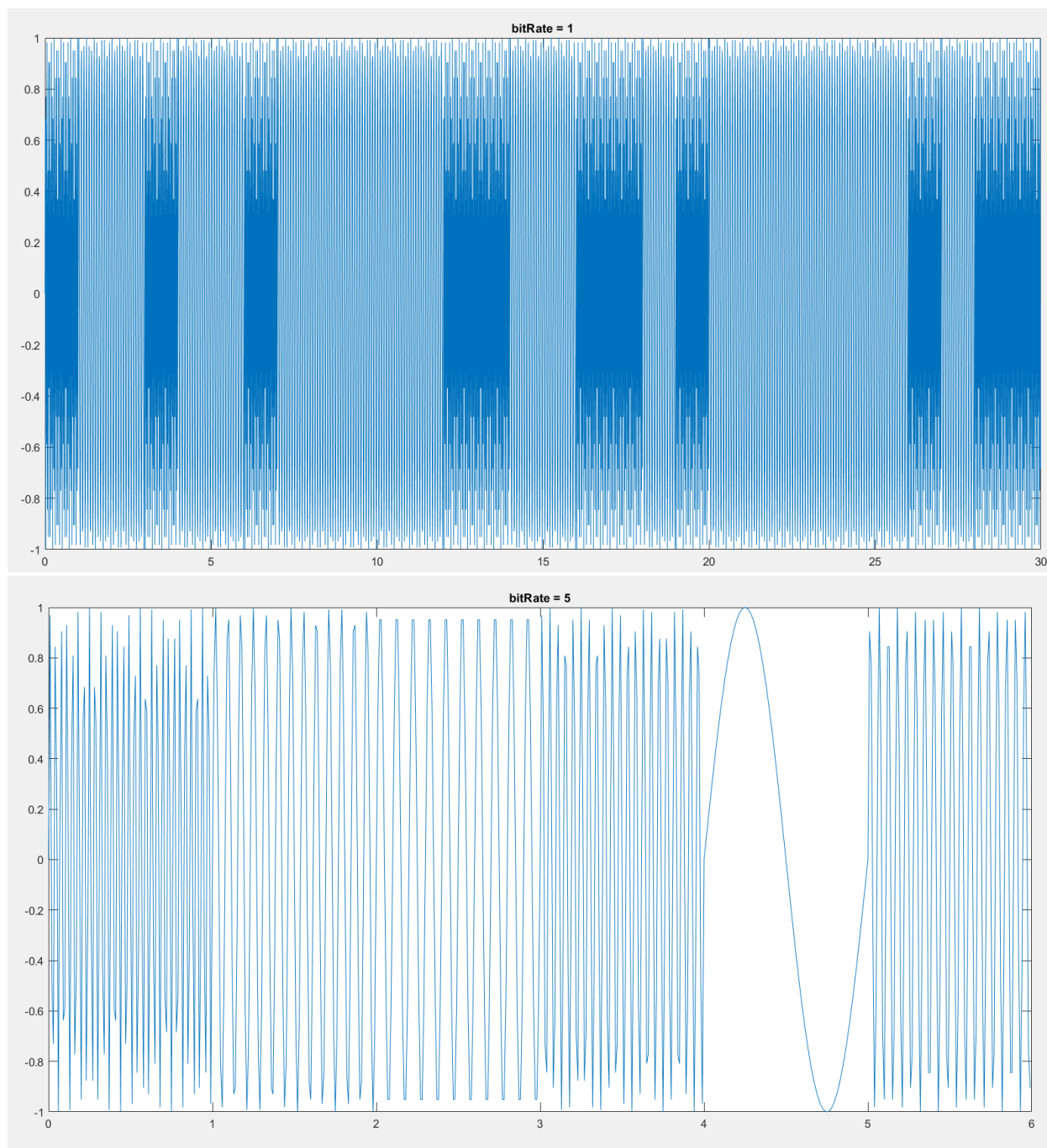
$$\sum_{i=0}^{parts_len} \sin(2\pi * freq_i * t)$$
$$freq_i = \frac{fs}{2^{bitRate+2}} + u(bin2dec(part_i) * \left(\frac{fs}{2^{bitRate+1}}\right) + 1$$

$$parts_len = \frac{msg_length}{bitRate}$$

کد مربوط به این بخش در فایل coding_freq قرار دارد.

تمرین 3-2:

سیگنال های خروجی پیام "signal" و $\text{bitrate}=1\text{ bit/sec}$ و $\text{bitrate}=5\text{ bit/sec}$ بترتیب:



تمرین 4-2:

با استفاده از کد تمرین دوم سیگنالی را generate می کنیم. به ازای هر ثانیه از سیگنال را به حوزه فوریه میبریم و فرکانس مربوط به ماکزیمم اندازه های سیگنال در حوزه فوریه را پیدا میکنیم. برای هر عدد یک threshold در نظر میگیریم و بسته به اینکه فرکانس استخراج شده از فوریه در کدام رنج قرار دارد عدد باینری مربوط به آن را پیدا می کنیم. این کار را برای هر ثانیه انجام داده و به یک عدد باینری می رسیم که می توانیم آن را با استفاده از mapset کلمه پنهان شده در سیگنال را استخراج کنیم.

کد مربوط به استخراج عدد باینری از سیگنال کدگذاری شده:

```
for i=1:fs:size(signal, 2)
    partSignal = signal(i:i+fs-1);
    [maxValue, maxIdx] = max(abs(fftshift(fft(partSignal))));
    maxIdx = fs/2 - maxIdx + 1;
    for j=1:(2^speed)
        if(maxIdx < floor(j*fs/(2^(speed+1))))+1
            binarizedMessage = [binarizedMessage dec2bin(j-1, speed)];
            break;
        end
    end
end
```

تمرین 5-2:

مقدار خروجی به ازای سرعت ارسال اطلاعات 1 و 5 هر دو به درستی تشخیص داده شد. کد مربوط به این بخش به همراه عکس خروجی را در زیر آورده ام:


```

noiseRate= 0.01;
msg='signal';
coded_signal_1=coding_freq(msg,1,mapSet);
coded_signal_5=coding_freq(msg,5,mapSet);

noise_signal_1=coded_signal_1+noiseRate*randn(size(coded_signal_1));
noise_signal_5=coded_signal_5+noiseRate*randn(size(coded_signal_5));

msg_decoded_1 = decoding_freq(coded_signal_1, 1, mapSet);
msg_decoded_5 = decoding_freq(coded_signal_5, 5, mapSet);

fprintf("noise rate = %.2f\n",noiseRate);
fprintf('decoded msg bitRate=1: %s \n', msg_decoded_1);
fprintf('decoded msg bitRate=5: %s \n', msg_decoded_5);

```

```

noise rate = 0.01
decoded msg bitRate=1: signal
decoded msg bitRate=5: signal

```

تمرین 6-2:

برای این قسمت به ازای هر مقدار از **noise rate** صد بار عملیات دیکودینگ را انجام می‌دهیم و در نهایت میزان موفقیت را بصورت درصدی برای هر سرعت حساب می‌کنیم.

کد این قسمت به همراه خروجی نمونه را در زیر قرار داده‌ام.

```

function [accuracy_1,accuracy_5] = calcAccuracy(mapSet, noiseRate,originalMsg)
    mistakesCount1=0;
    mistakesCount5=0;
    for i=1:100
        coded_signal_5 = coding_freq(originalMsg,5,mapSet);
        noise_signal_5 = noiseRate*randn(size(coded_signal_5));
        coded_signal_5 = coded_signal_5 + noise_signal_5;
        coded_signal_1 = coding_freq(originalMsg,1,mapSet);
        noise_signal_1 = noiseRate*randn(size(coded_signal_1));
        coded_signal_1 = coded_signal_1 + noise_signal_1;
        msg_decoded_5 = decoding_freq(coded_signal_5, 5,mapSet);
        msg_decoded_1 = decoding_freq(coded_signal_1,1,mapSet);
        if(~strcmp(msg_decoded_5, originalMsg))
            mistakesCount5 = mistakesCount5 + 1;
        end
        if(~strcmp(msg_decoded_1, originalMsg))
            mistakesCount1 = mistakesCount1 + 1;
        end
    end
    accuracy_1=(100-mistakesCount1)/100;
    accuracy_5=(100-mistakesCount5)/100;
end

```

```

noise rate = 1.25
accuracy bitRate=1: 0.94
accuracy bitRate=5: 0.96

```

به طور عجیبی نتایج با آنچه در مقدمه خواندیم هم خوانی نداشت یعنی به ازای bit rate بیشتر نسبت به نویز مقاومت بیشتر شده است و bit rate های کمتر مقاومت کمتری دارند.

تمرین 7-2:

برای bit rate=1 به ازای قدرت نویز 1.24 . برای bit rate=5 به ازای قدرت نویز 1.35 بطور تقریبی.

```
noise rate = 1.35
accuracy bitRate=1: 0.80
accuracy bitRate=5: 0.90
>>

noise rate = 1.24
accuracy bitRate=1: 0.86
accuracy bitRate=5: 0.97
```

تمرین 8-2:

هر فرکانس های انتخابی، کدگذاری انجام شده نسبت به نویز مقاوم ی ت بیشتر ی تر خواهد بود. پس هر چه پهنای باند بیشتر ی ی اطلاعات را ارسال کنیم و در ع یر مصرف کنیم یم حال نسبت به نویز مقاوم باشیم

تمرین 9-2:

خیر مقاوم تر نمی شود. اگر پهنای باند را افزایش دهیم ولی از بخش کوچکی از آن استفاده کنیم باقی فرکانس ها را بال استفاده گذاشتیم ولی اگر مثال فرکانس نمونه برداری را به جای 100 تا 1000 تا بکنیم تا فرکانس 500 هم می توانیم سیگنال هایمان را پخش کنیم ولی با فرکانس 100 ما 50 فرکانس بیشتر استفاده نمی کردیم که اگر فرکانس را بالاتر هم ببریم و باز از 50 تای اول استفاده بکنیم و بین کل بازه فرکانس هایمان را پخش نکنیم مقاومتمان نسبت به نویز تغییری نمی کند.

