

MEH-310 DIGITAL COMMUNICATIONS LABORATORY

MATLAB EXERCISE#3: FSK MODULATION (Deadline: June 8, 2020)

Zeynep SAKLI 160207013

Cansın DÜŞEN 160207051

In this exercise, we create a frequency shift keying (FSK) transmission system including a transmitter (TX), a receiver (RX) and a simple additive white Gaussian noise (AWGN) channel model. Demodulation in the RX side will be performed using both coherent and non-coherent demodulation methods.

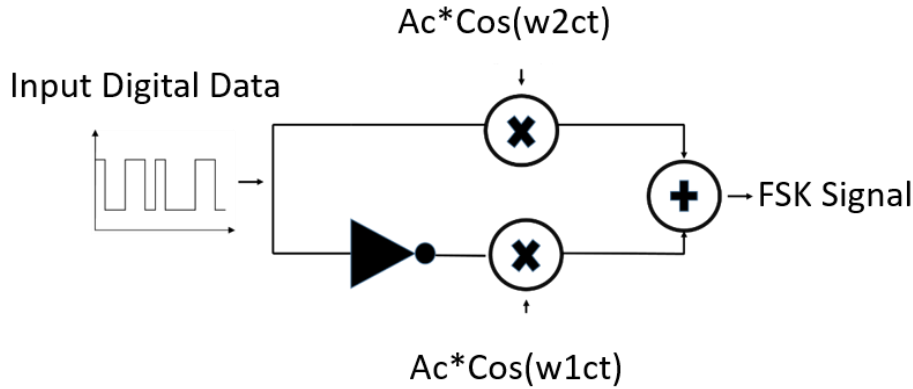


Figure 1. FSK modulation block diagram

PART A. MODULATION (40 pts)

Q1) Run the attached code for various SG variables (at line 33), such as 1, 2 ...5. Plot input bit signal and BFSK modulated signal. (Check below for the axis labels&sizes.) Non-orthogonal case is also studied (only one f_1 and f_2 frequencies).

For each SG value, phase jumps and bandwidth of the modulated signal (BW) should be studied and explained. Moreover, BW calculated from Matlab figure should be tabulated versus SG values. You should also calculate its BW theoretically and add this calculated value to table.

→ Dikgen olmayan şekilde ayarlamak için rho hesabında da gösterdiğimiz gibi 1 biti taşıyıcısı ve 0 biti taşıyıcısını T_b süresinde integralini alıp $1/T_b$ ile çarptık ve sonuç sıfır çıkmadı bu yüzden dikgen olmayan şekilde ayarladık. Eğer sonuç sıfır gelseydi dikgen olacaktı.

→ rho değeri sıfırdan küçük gelince mesafe artacak ve hata olasılığı düşecek bunu biliyoruz.

→ Faz atlamaları hakkında konuşmak gerekirse çift sıfır geçişleri kullanıldığında 0'dan 1'e ya da 1'den 0'a gibi farklı ardışık semboller için iletilen işarette bir devamsızlık meydana gelmemektedir. Yani faz atlaması gözlenmemekte. Tek sıfır geçişleri kullanıldığında ise ardışık olarak farklı semboller iletildiğinde 180 derece faz farkı oluştuğunu gözlemliyoruz. Bu yüzden çift sıfır geçişleri tercihimizdir.

→ Biz 1.43 sıfır geçişini de teoride çok fazla faz atlamasına sebep olduğunu öğrendiğimiz için deneyelim dedik ve bu durumu da ekstra gözlemledik denildiği gibi 1.43 SG değerinde -0.8 dB SNR sağlasa da daha fazla faz atlamasına sebep oldu.

→ Sıfırlarda frekans daha düşük olduğu için daha açık l'lerde ise daha sık sinyal görüyoruz frekans yüksek olduğu için. Aradaki fark ise 500 Hz.

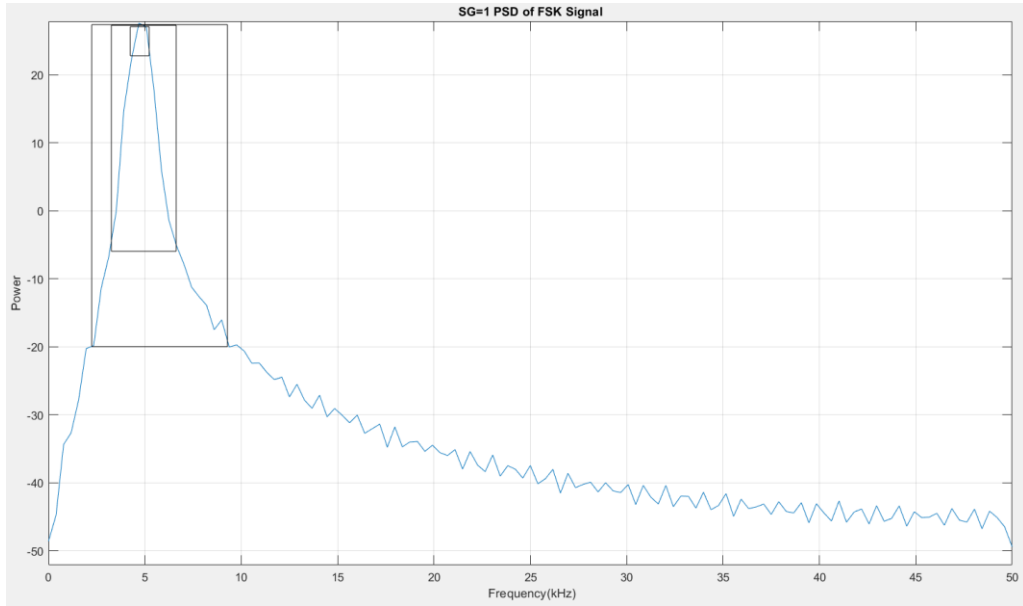
→ BW'ler için konuşmak gerekirse birinci sıfır geçişinde $f_{c1}=5k$ Hz $f_{c0}=4.5k$ Hz olduğu ve $T_b=0.001$ için;

Teorik olarak bant genişliği $(5000-4500)+2/T_b=2.5kHz$ çıkmaktadır. Pratikte de işaretin frekans spektrumuna bakıldığında ise bw'yi 30 dB'den -20 dB'e düştüğü yer olarak aldık ve 2.43 Hz olduğunu gördük. Aynı işlemleri diğer sıfır geçişleri için de yapınca aşağıdaki tabloda değerleri görebilirsiniz.

→ PSD grafiklerinden SG 1 ve SG 2 için frekans farkı 500 olduğu için iki kola ayrılmadı. Diğerlerinde fark geçtiği için iki ayrı tepe gördük.

→ Teorik BW = $(f_{c1}-f_{c0})+2/T_b = (f_{c1}-f_{c0})+2*R_b$ olduğunu tekrar yazalım.

Sıfır Geçişleri:	Teorik Hesap BW	Pratik BW
1 $f_{c1}=5k$ Hz $f_{c0}=4500$ Hz	2500 Hz	2.43 kHz
2 $f_{c1}=5k$ Hz $f_{c0}=4000$ Hz	3000 Hz	3.10 kHz
3 $f_{c1}=5k$ Hz $f_{c0}=3500$ Hz	3500 Hz	3.32 kHz
4 $f_{c1}=5k$ Hz $f_{c0}=3000$ Hz	4000 Hz	3.89 kHz
5 $f_{c1}=5k$ Hz $f_{c0}=2500$ Hz	4500 Hz	4.28 kHz



→ Öncelikle pwelch komutuyla çizdirdiğimiz PSD grafiğinde fourier transformunun karesini çiziriyoruz SG=1 için PSD grafiğinde sinc işareti olmaması oldukça normaldir çünkü iki işareti beraber görüyoruz ve pencereleme yapıldığı için sinc oluşmaz.

→ Yukarıdaki şekilde gösterdiğimiz gibi BW ölçmek için 30dbden -20 db'ye inip ölçülebilir 30 db'den -5 db'ye inip ölçülebilir diye derste Sultan hoca söyledi biz de farklı SG değerlerimiz için 30 db'den -5 db'ye olan yerden bw ölçtük ve sonuçlarımızı buna göre yazdık.

→ Bununla birlikte pwelch spektrumu tek bir işaret olarak gözüktü çünkü SG=1 için aradaki fark 500 Hz.

→ Diğer SG değerleri için PSD grafiklerini dosya boyutu büyümesin diye bu açıklamayı yaparak koymama kararı aldık.

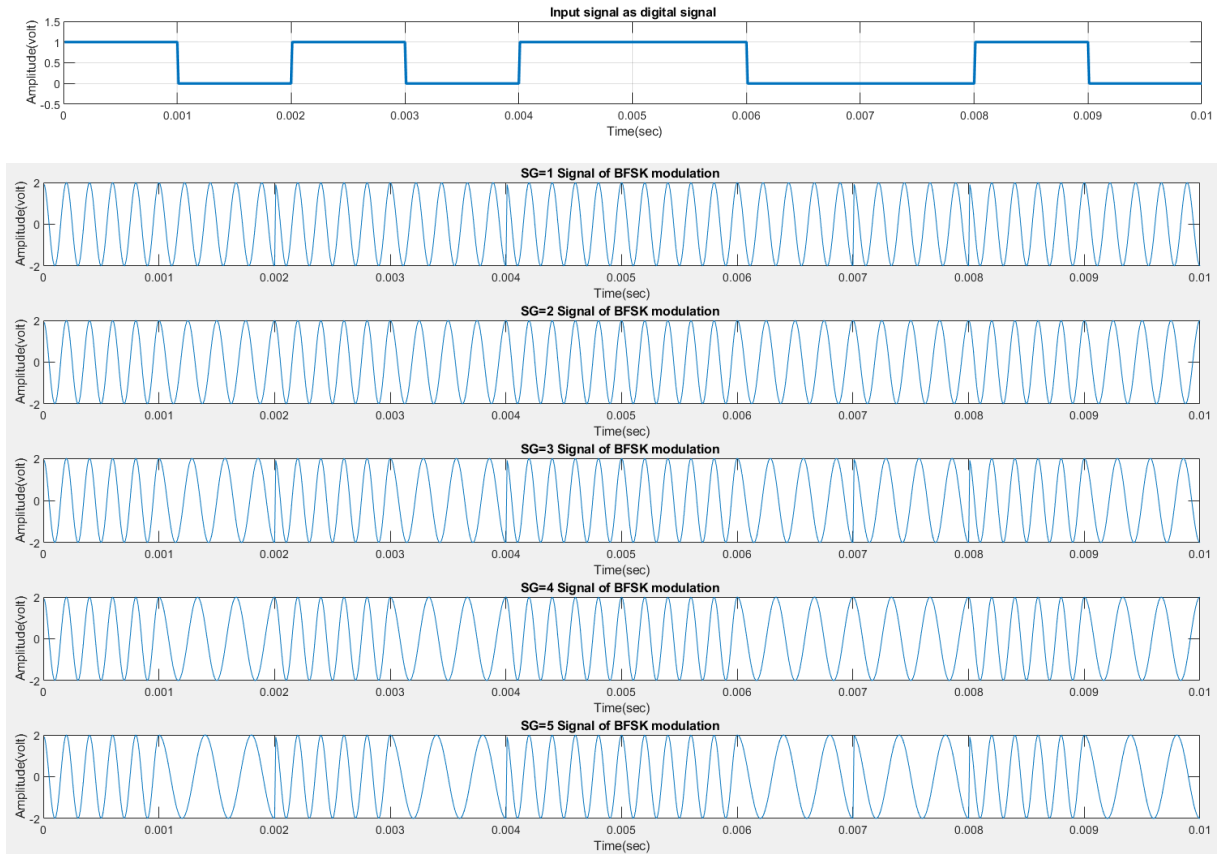


Figure 2. Example figure format (Your Matlab figures are placed here)

Q2) What's the output frequency that corresponds with logic-0s in the digital data?

Duraklatma frekansı. Sayısal verilerdeki mantık-0'lara karşılık gelen sinyalin frekansı (boşluk frekansı olarak adlandırılır) genellikle modülatörün nominal taşıyıcı frekansından daha düşüktür. Modülatör, taşıyıcı frekansında bir sinyal vermez, bu nedenle burada referansın "nominal" taşıyıcı frekansı olduğu anlamına gelir.

Yüksek frekanslarda 1 biti düşük frekanslarda 0 biti olur genelde.

Bizim kodumuzda $fc0=fc1-SG*(1/(2*Tb))$ % carrier frequency for bit 0
Kod satırıyla 0 biti için olan frekansımız SG değerine göre değişmekte.

Q3) What's the output frequency that corresponds with logic-1s in the digital data?

Bizim kodumuz için 5000 Hz yani 5kHz.
 $fc1=mc1*(1/Tb)$ % carrier frequency for bit 1

Sinüs dalgasının süresi logic “0” iletildiğinde duraklamadaki sinüs dalgasının süresinden daha kısa olduğu için, logic “1” e karşılık gelen frekans daha yüksektir.

Q4) For different SG values, check the orthogonality of these carriers by using trapz command.

```
% check the signal orthogonality
zeynep=c1.*c0;      %% s1(t) ve s(0)t
integral=trapz (Tb,zeynep) %Tb suresinde integral
rho=(1/Tb) * (integral) % rho boyle hesaplanıyor
```

→Bu kod üzerinden farklı sıfır geçişlerine göre $f_{c1}=5000$ Hz $f_{c0}=4285$ Hz frekanslarında rho değerleri sırasıyla ;

SG 1 için rho= -3.8266

SG 2 için rho= - 3.8424

SG 3 için rho= -3.8563

SG 4 için rho= -3.8684

SG 5 için rho= -3.8787

→Biz hatalı olduğunu düşünüyoruz. Ama bir türlü çıkartamadık daha mantıklı bir şey. Kitapta gösterdiği gibi T_b süresi boyunca $s_1(t)$ ve $s_0(t)$ çarpımının integralini alıp onu da $1/T_b$ ile çarptık.

→Sonucun negatif veya pozitif çıkmış olması dikgen olmadığını gösterirken sonuç sıfır çıkarsa dikgendir. Negatif olursa d mesafesi artar ve hata olasılığı iyileşir pozitif olması durumunda ise tam tersi gerçekleşir ve daha birbirine sık sinyaller oluşur ayırt etmekte zorluk çıkar.

→Unutulmamalıdır ki frekanslarla oynayarak d mesafesini arttırmak isteriz çünkü genlikle oynayarak bu işlemi yaparsak daha fazla güç tüketimi yapmış oluruz. Bu yüzden de frekanslarla oynadık.

PART B. COHERENT DEMODULATION (15 pts)

At line 94, demodulation process start.

Q5) Plot input signal and demodulated FSK signal for various SG values. (Check below for the axis labels&sizes.)

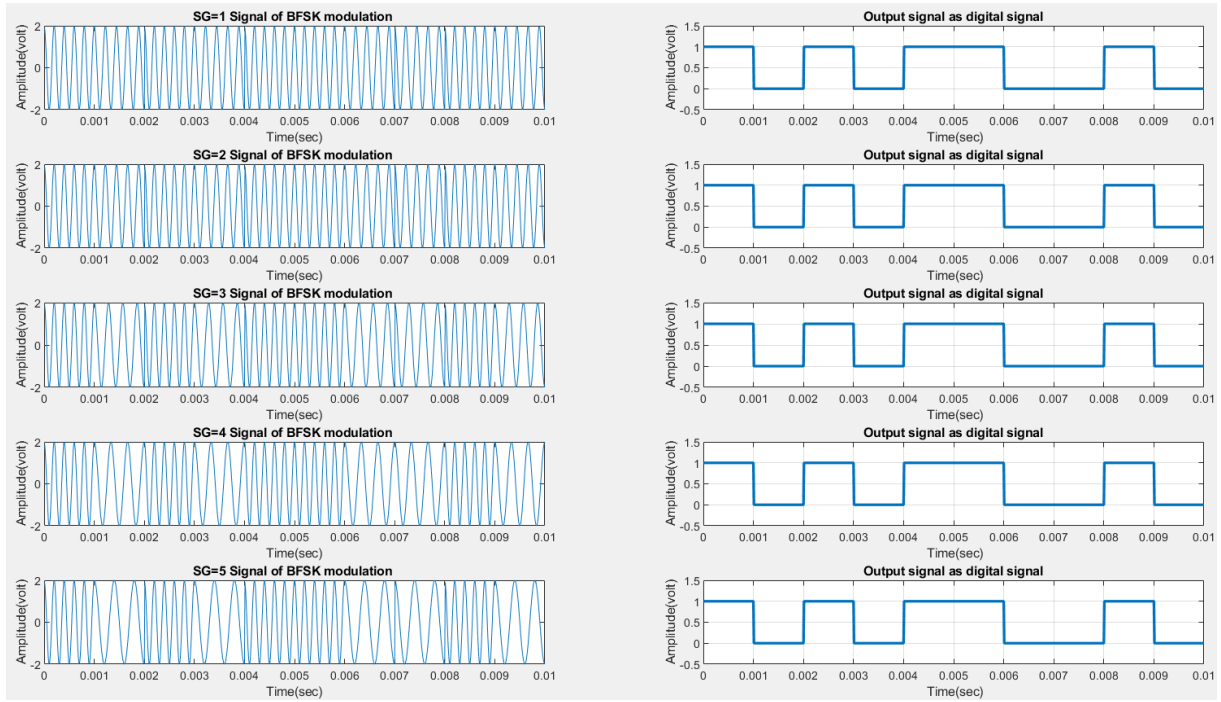
→Görüldüğü gibi sıfır geçişleri artınca işaretin elde ediminde fark olmuyor çünkü işaretin eşevreli demodülasyonla elde ediminde önce BPF'den geçirilir sonra ilinti alıcısına tabii tutulur ve karar devresinde ayırımı yapılır.

→Yani sıfır geçişleri de tam sayılar olduğu için yalnızca modülasyon kısmında tek ve çift durumlarında 180 derece faz sıkıntısı oluşturur demodulasyonda asıl önemli olan SG değildir.

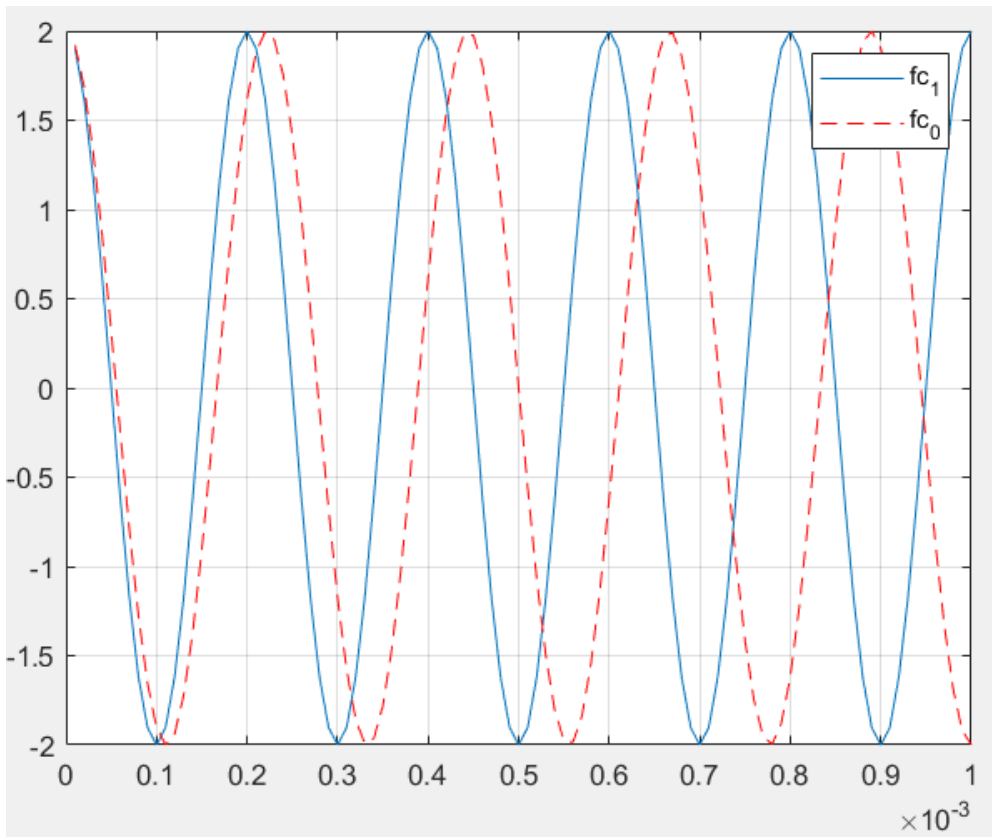
→Çünkü Sıfır geçişi değişiminde bant genişliği değişir o da çok az düzeylerde olur. Bu yüzden çıkışlarda aynı sonucu elde ederiz.

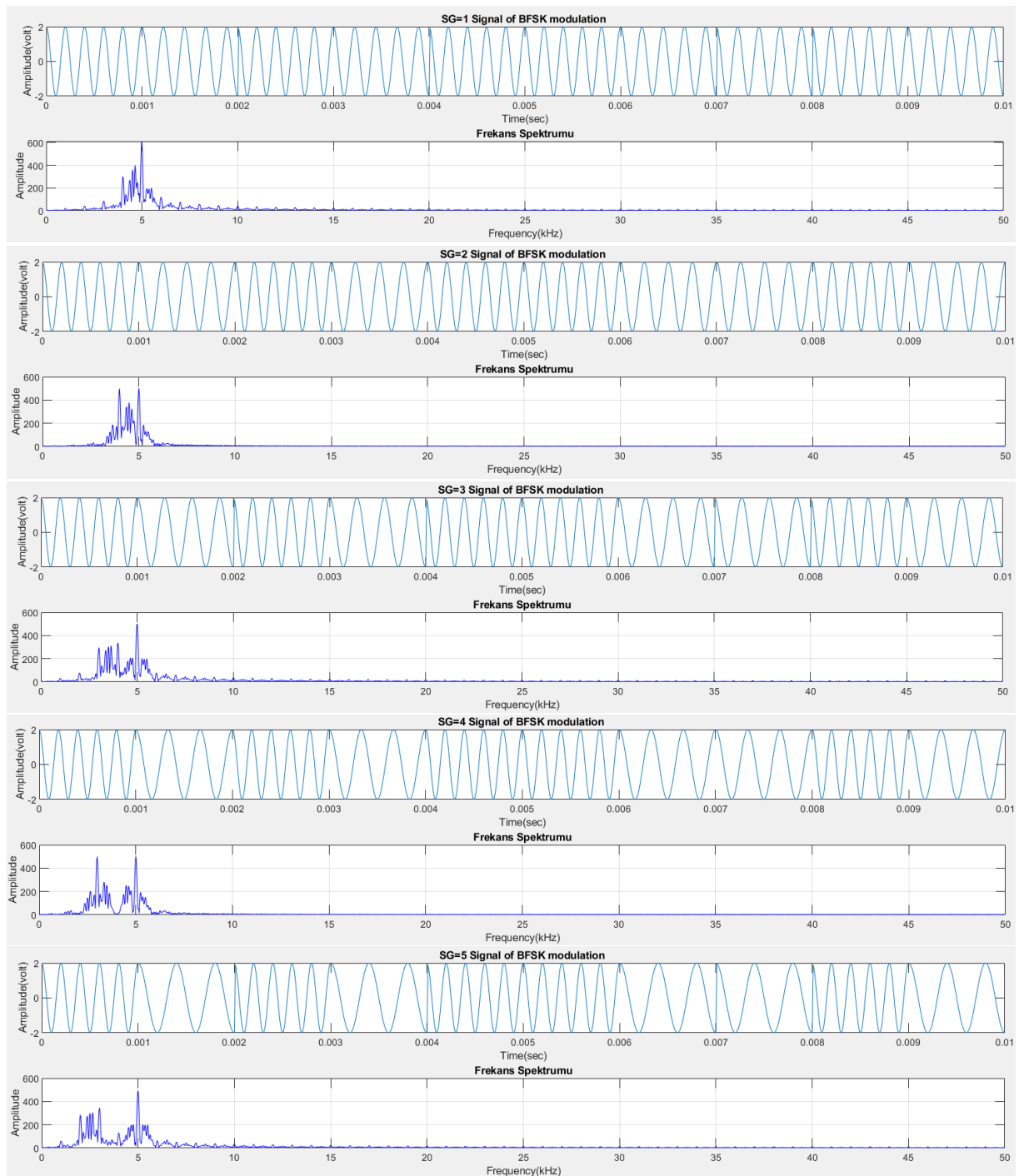
→SG=1 durumunda neredeyse ayırt edilemeyeceğini gözlemledik sebebi ise frekans aralığı diğerlerine göre en küçük.

→Frekans spektrumlarını incelediğimizde ise eğer farklı genlikte geldiyse bunun sebebi işaretteki 1 ve 0 iletim oranının eşit olmamasıdır diye yorumlayabiliriz.



→ Bununla birlikte taşıyıcı frekanslarına baktığımızda , frekansları daha yakın seçersek bant o kadar yaklaşacak ve eşevresiz demodulasyonda ayırt etmek çok daha zorlaşacaktır. Özellikle SG=1 için.

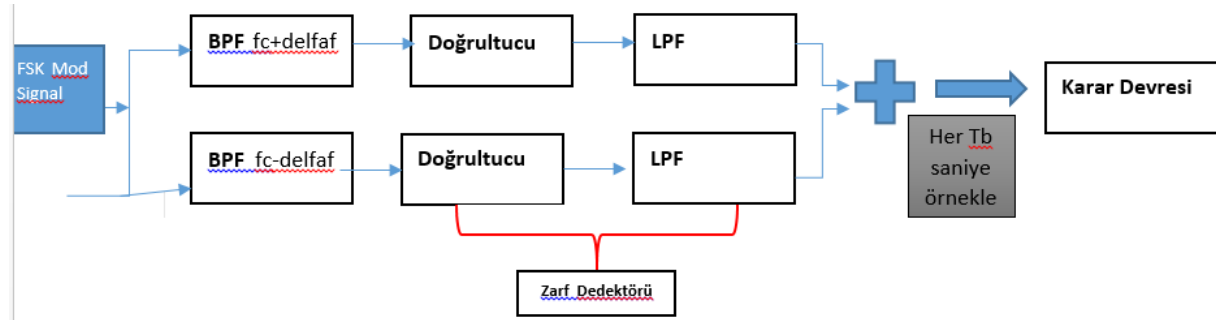




PART C. NON-COHERENT DEMODULATION (45 pts)

The code does not include the non-coherent demodulator. For this reason, you should add this section and answer below questions.

Q6) How could you demodulate FSK signal using envelope detector. Draw a block diagram and explain.



→BASK eşevreli olmayan demodülasyonda olduğu gibi burda da zarf dedektörü kullanılabilir bunun için $f_c + \Delta f$ ve $f_c - \Delta f$ frekanslarındaki işaretlerin ayrıştırılması gerekmektedir. Diyagramda gösterdiğimiz gibi iki paralel koldan birinde $f_c + \Delta f$ frekansı civarını geçiren bir BPF ve diğerinde $f_c - \Delta f$ civarını geçiren BPF kullanılır. Bundan sonra ise ayrıştırılan işaretler zarf dedektörüne gider ve işaretin genliği bulunur.

→Yani Zarf dedektörü genlik farkı olup olmadığına bakıyor diyebiliriz.

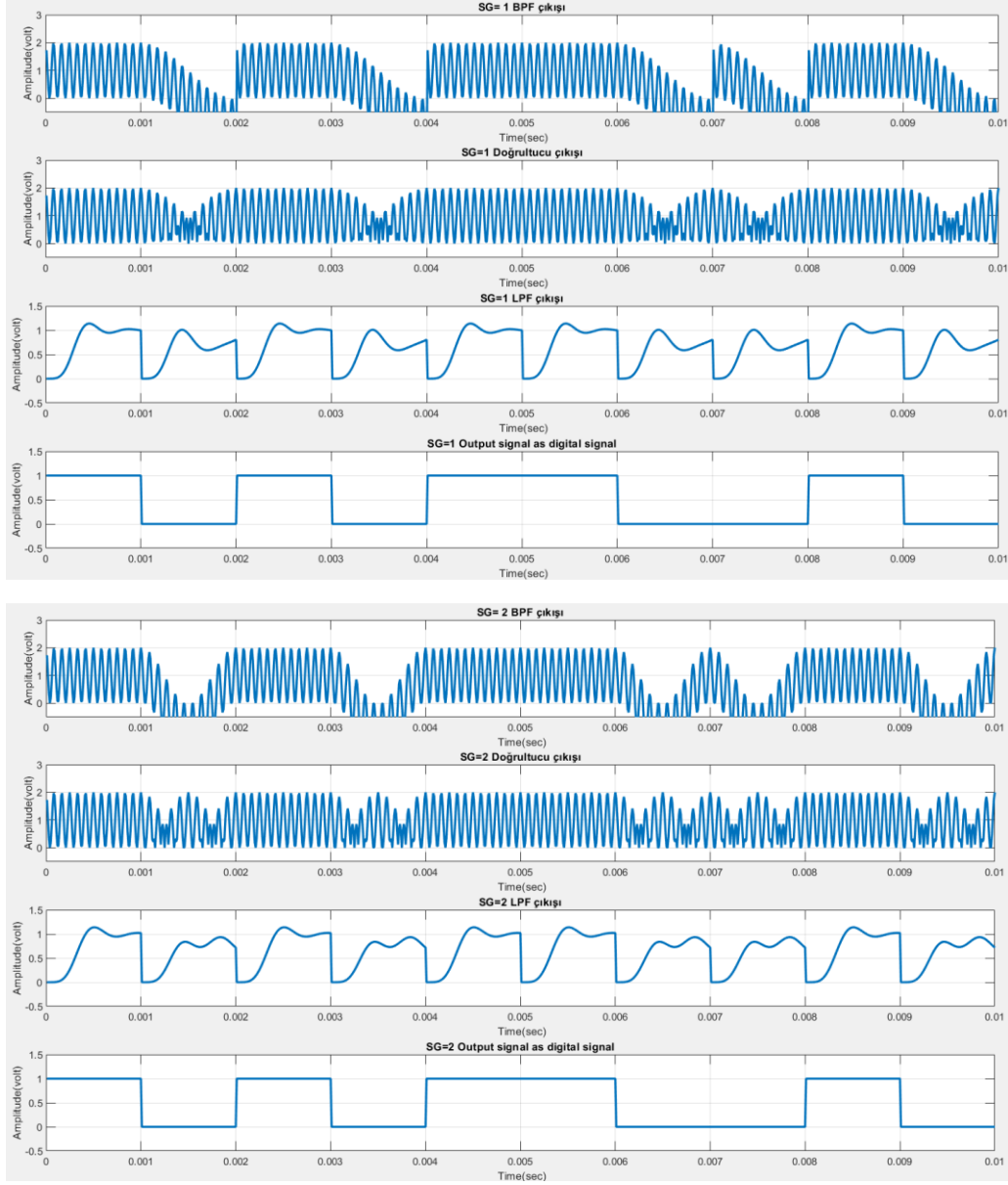
→Zarf dedektörü bildiğimiz gibi bir doğrultucu ve alçak geçiren filterden oluşmaktadır. Seçeceğimiz kesim frekansı burada işaretin geri elde edilmesinde önemli rol oynayacaktır.

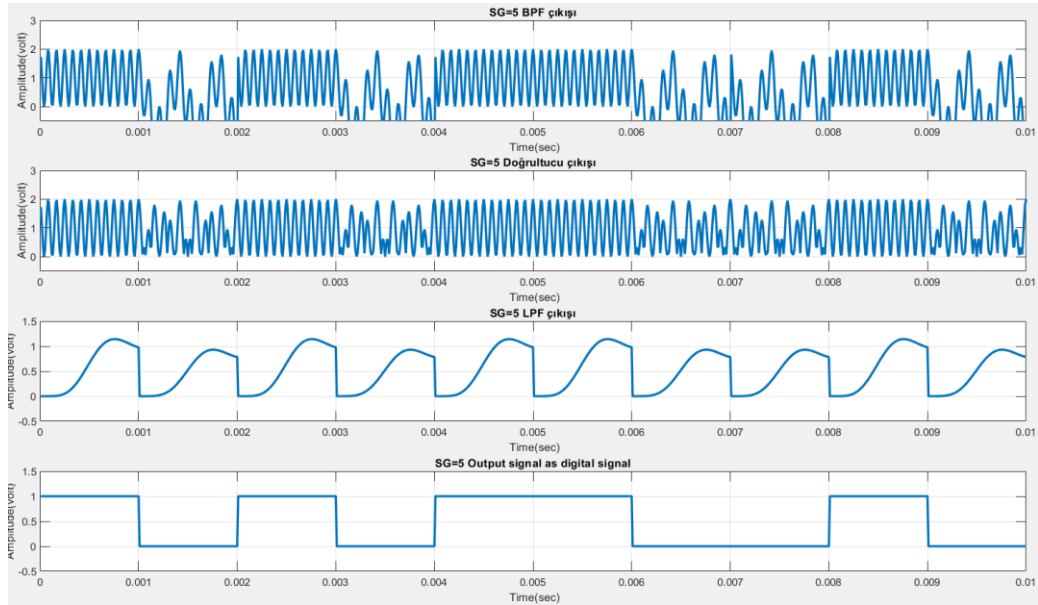
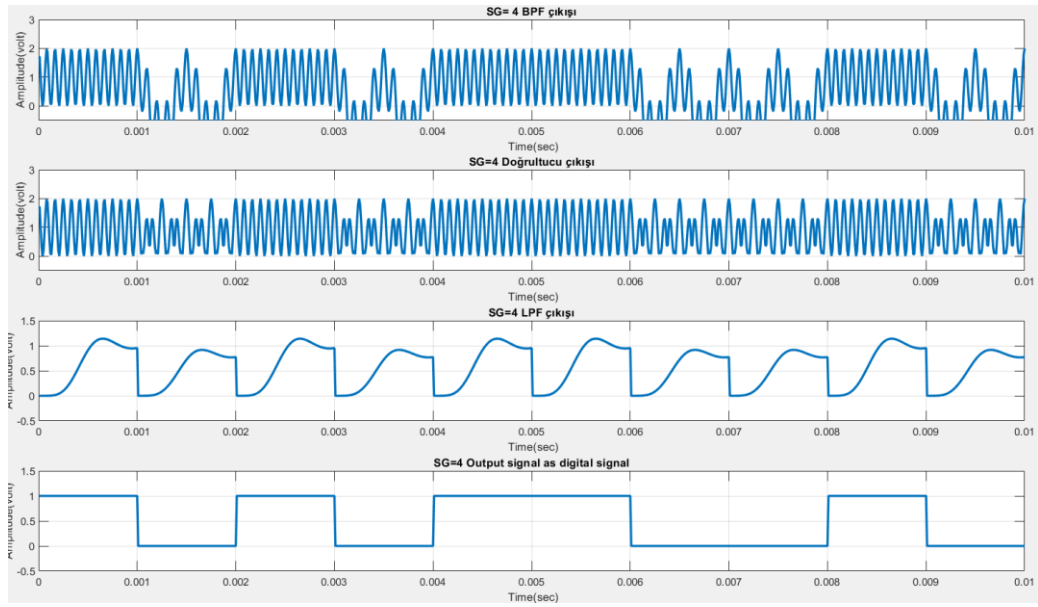
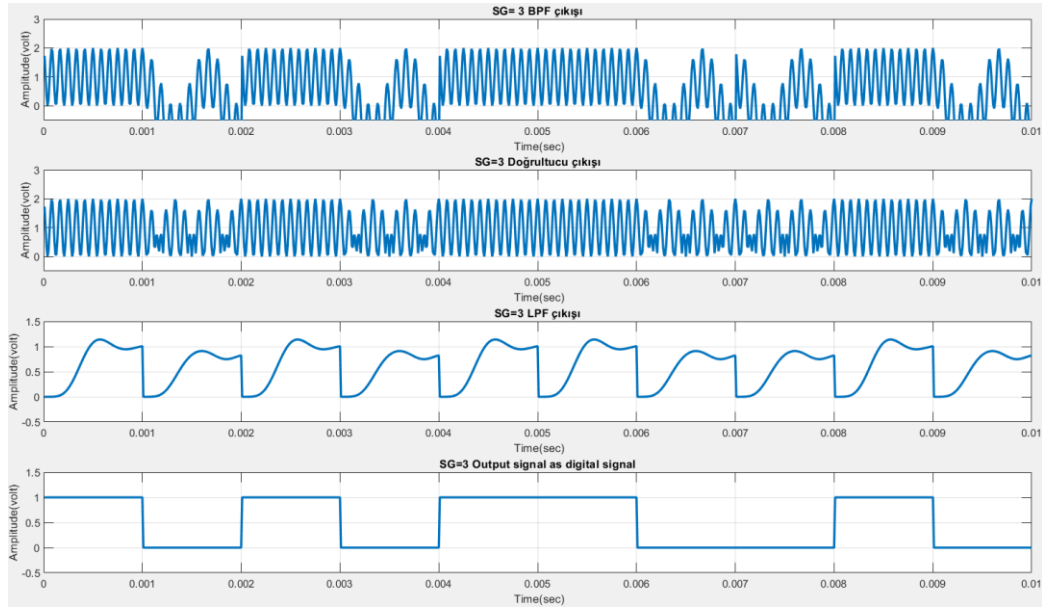
→OOK'da paralel iki koldan birinde taşıyıcı genliği değeri elde edilmekte, diğer kolda ise bir işaret oluşmadığından sıfır genliği olur ve karar devresini buna göre kodlamak oldukça kolay olur. Eğer iki taşıyıcının da genliği aynı ise yani var yok haberleşmesi değil ise genliklerin toplamının yarısından büyük mü küçük mü sorusunun cevabına göre karar devresinde 1 mi 0 mı ayrımı yapılabilir.

→İşaretler için aynı genlikte taşıyıcılar kullanıldığında sonuçta $s_{d1} = -s_{d0}$ olduğundan en uygun karar eşiği doğrudan sıfır seviyesi olarak bulunmakatdır.

NOT: $s_{d0} = R_{01} - R_{00}$ ve $s_{d1} = R_{11} - R_{10}$

Q7) Implement proposed method in MATLAB. Plot output of each block, and explain its purpose of use.





→Eşevreli olmayan BFSK için BASK eşevreli olmayan demodülasyon yöntemini iki ayrı kol için uygulayınca yani iki farklı frekans için uygulanınca doğru sonuç elde ediliyor. Block diyagramda görüldüğü gibi iki ayrı paralel koldan birinde $f_c + \Delta f$ ve diğerinde $f_c - \Delta f$ frekansları civarını geçiren BPF'ler kullanılır ki ayrıştırılabilirsin.

→Daha sonra negatif değerlerdeki sinyal için doğrultucudan ve sonra işareti belli bir f_c 'den geçirmek gerekir LPF aşamasında, bu ikisi zarf dedektörünü oluşturur. Zarf dedektörü sayesinde genlik farkı olup olmadığına bakıyoruz bunu yaparken LPF çıkışındaki işaretin üstündeki zarflardan yararlanıyoruz ve en son bu bilgiyi comparator yani karar devremizde 1 mi 0 mı sorusunun cevabını buluyoruz. Karar devresi zarf sezicide bulunan genliğine göre hangi sayısal sembolün alınmış olduğuna karar verir.

→Görüldüğü gibi SG arttıkça bant genişliği artar çok küçük oranlarla da olsa ve LPF çıkışı o yüzden düzleşir çünkü frekanslar da değişir.

Q8) Comment on type of filters used in demodulation.

→İlk etapta BPF kullanılır burada alt ve üst yan bant olarak düşünürsek bu civardaki frekansların iki ayrı koldan geçimi sağlanır. $f_c + \Delta f$ ve $f_c - \Delta f$ frekansları civarını geçiririz bu da kullanılan frekans aralığını ayırıştırıp bu sayede frekansta karışmayı önler, gürültü etkisini de azaltır.

→Zarf dedektöründeki LPF'de ise seçeceğimiz kesim frekansına göre işaretin geri elde edilmesinde önemli rol oynayacaktır. İşaretin yüksek ve düşük frekansına göre gerekli kısımları geçirir. Düşük frekanslar için 0, yüksek frekanslar için 1 demektir. LPF çıkışında ise zarf dedektörü sayesinde işaretin genlik farkına göre karar devresinde bu bilgiyi kullanabiliyoruz.

Q9) What is your cut off frequency (f_c) for the filter/filters you used?

→BPF için üst yan bant için yani analogtaki gibi düşünürsek diye öyle dedik. 1 için $f_c + \Delta f$ kadarlık yani bizim kodumuzda 5.5k Hz ve 0 için ise $f_c - \Delta f$ yani 4.5kHz BPF için uygun oldu.

→LPF için ise iki ayrı paralel kol için iki ayrı f_c belirledik. Diyagrama bakarsak üst kol için f_{c1} alt kol için ise f_{c0} frekansları kesim frekanslarımız oldu.

→Hocam açıkçası rho hesabında kafamız çok karıştı rho hiç 0 gelmiyor dikgen yapmaya çalışsak bile o yüzden biz de frekansı değiştirdik ve bunu yaparken rhoyu bizde çıkan hesaplara göre 0'dan küçük seçip mesafenin artmasını sağladık bunun üzerinden işlemlerimizi yaptık. Teorik bilgilerimizle çelişen durum çok fazla olmayınca bu şekilde koyalım dedik. Hatamız varsa kusura bakmayın.

→Sultan hocaya mail attığımızda ise neden sonucun böyle geldiğini teslim tarihinden sonra açıklayacağını söyledi. Biz de merak ediyoruz.

Only add rho calculation (at line 47) and non-coherent detection code:

```
% check the signal orthogonality
zeynep=c1.*c0;      %% s1(t) ve s(0)t
integral=trapz(Tb,zeynep) %Tb suresinde integral
rho=(1/Tb)*(integral)% rho boyle hesaplanıyor
```

ESEVRESİZ DEMODULASYON

```
x=x_mod; %%BFSK işareti
% ***** Channel model h and w *****
h=1; % Fading
w=0; % Noise
% ***** Received signal y *****
y=h.*x+w;

bandpass_mod=[]; %BPF yazdırmak için bilgileri tutturacağız
y_dem=[]; %işaretleri ayrı ayrı tutacağız
t=Tb/nb:Tb/nb:Tb;
c_dem1=cos(2*pi*fc1*t); % carrier signal for information 1
c_dem2=cos(2*pi*fc0*t); % carrier signal for information 0
dogrultucu_mod=[]; %dogrultucu bilgisini yazdırmak için
rend_mod=[]; %LPF filte çıkışı için
for n=t2L:t2L:length(y)

    if (x_inp(10)==1)
        y_dem1=c_dem1.*y((n-(t2L-1)):n);

        dogrultucu1=abs(y_dem1); %dogrultucu için kullandık
        [B,A] = butter(6,(fc1+500)/100000,'low'); %%LPF için kullandık
        rend=filter(B,A,dogrultucu1); %%LPF
    else
        y_dem1=c_dem2.*y((n-(t2L-1)):n);
        dogrultucu1=abs(y_dem1);
        [B,A] = butter(6,(fc0-500)/100000,'low');
        rend=filter(B,A,dogrultucu1);
    end

    if(rend<=1); % % logic level = (Ac)/2
        a=0;
    else
        a=1;
    end
    y_dem=[y_dem a];
    rend_mod=[rend_mod rend];
    dogrultucu_mod=[dogrultucu_mod dogrultucu1];
    bandpass_mod=[bandpass_mod y_dem1];
end
x_out=y_dem; % output signal;

% ***** Represent output signal as digital signal *****
xx_bit=[];
```

```

for n=1:length(x_out);
    if x_out(n)==1;
        xx_bitt=ones(1,nb);
    else x_out(n)==0;
        xx_bitt=zeros(1,nb);
    end
    xx_bit=[xx_bit xx_bitt];
end
t4=Tb/nb:Tb/nb:nb*length(x_out)*(Tb/nb);
subplot(4,1,4)
plot(t4,xx_bit,'LineWidth',2);grid on;
axis([ 0 Tb*length(x_out) -0.5 1.5]);
ylabel('Amplitude(volt)');
xlabel(' Time(sec)');
title('Output signal as digital signal');

subplot(4,1,3)
plot(t4,rend_mod,'LineWidth',2);grid on;
axis([ 0 Tb*length(x_out) -0.5 1.5]);
ylabel('Amplitude(volt)');
xlabel(' Time(sec)');
title('LPF çıkışı');
subplot(4,1,2)
plot(t4,dogrultucu_mod,'LineWidth',2);grid on;
axis([ 0 Tb*length(x_out) -0.5 3]);
ylabel('Amplitude(volt)');
xlabel(' Time(sec)');
title('Doğrultucu çıkışı');

subplot(4,1,1)
plot(t4,bandpass_mod,'LineWidth',2);grid on;
axis([ 0 Tb*length(x_out) -0.5 3]);
ylabel('Amplitude(volt)');
xlabel(' Time(sec)');
title('BPF çıkışı');

```

Zeynep SAKLI 160207013

Cansın DÜŞEN 160207051

NOT: Hocam Part C eşevresiz demodülasyon kısmını İrem İpekli 160207010 ve Aleyna Durmuş 160207003 arkadaşlarımızla birleşip yapabildik ve kopya olarak gözüksün istemeyip belirtmek istedik. Saygılarımızla.