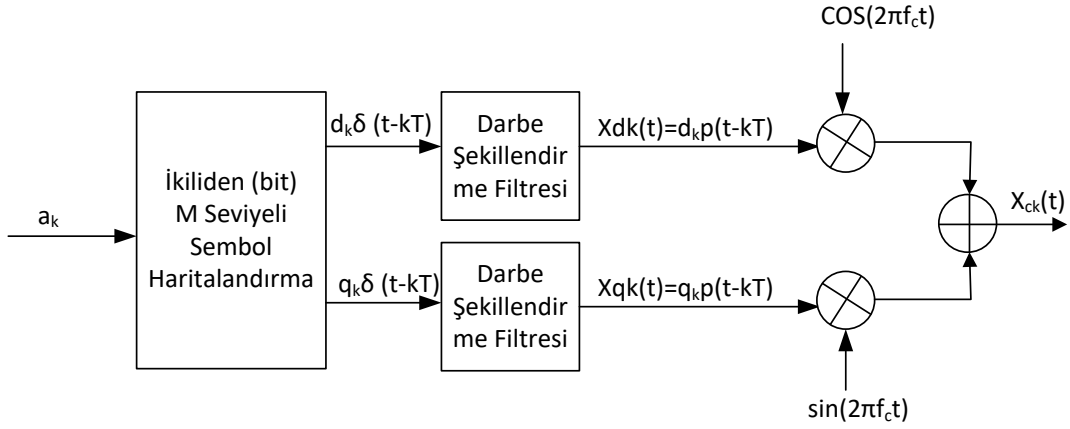


Sayısal Modülasyon Deneyi

(BPSK, QPSK(4-QAM) Modülasyonları)

Darbe Şekillendirme, Senkronizasyon ve ISI

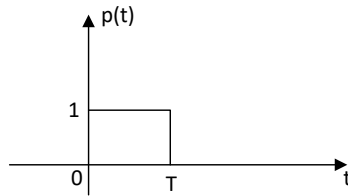
Sinyaller gerçek hayatta izin verilen bir band içinde yer alacak şekilde iletilmek zorundadır. Sinyalin kendi izin verilen bandı dışında ne kadar güce sahip olabileceği yasalarla belirlenmiştir ki bu güç diğer bandları rahatsız etmeyecek kadar küçük olmalıdır. Aşağıda bir sayısal modülatörün bandpass eş değer modeli gösterilmektedir.



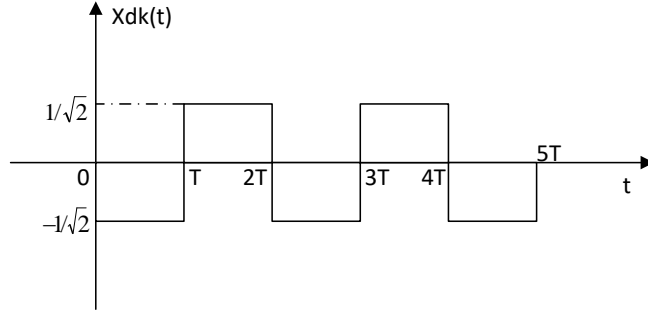
M seviyeli modülatörün gelen bitlere göre oluşturulan d_k ve q_k değerleri birim vuruş işaretleri ile çarpılıyor olarak varsayılın. Modülatör olarak QPSK kullanılıyor ise gelen bitlere karşılık d_k ve q_k değerleri aşağıdaki gibi oluşturulabilir.

a_k	d_k	q_k
00	$-1/\sqrt{2}$	$-1/\sqrt{2}$
10	$1/\sqrt{2}$	$-1/\sqrt{2}$
11	$1/\sqrt{2}$	$1/\sqrt{2}$
01	$-1/\sqrt{2}$	$1/\sqrt{2}$

Yukarıda k indisi sembol numarasını göstermektedir. Darbe şekillendirme filtresi ise birim vuruş tepkisi $p(t)$ olan bir filtre olsun. Farz edelim ki $p(t)$ aşağıdaki gibi olsun.



Yukarıdaki tabloya dikkat edilecek olursa d_k en anlamlı bite göre değişmektedir. Örnek olarak en anlamlı bitlerin sırası ile 01010 olduğu varsayılın. Bu durumda $x_{dk}(t)$ sinyali aşağıdaki gibi olacaktır.



Gelen bitlere göre değişim gösteren $x_{dk}(t)$ sinyalinden tam bit süresi T 'nin ortasında örnekler alınırsa elde edilen genlik değerlerine bağlı olarak bitlerin değerleri tekrar elde edilebilir. Bir senkronizasyon hatası sebebi ile örnekleme noktasının tam orta noktanın biraz sağında yada solunda olması durumunda da iletilen bitler doğru olarak algılanabilecektir. Yukarıdaki şekle göre zamanda semboller arası girişim (ISI) yoktur ve senkronizasyon hatasına duyarlılık oldukça azdır.

Aşağıdaki matlab kodunda 10 adet rastgele bit oluşturulup bunlardan QPSK modülasyonuna göre $X_{dk}(t)$ ve $X_{qk}(t)$ sinyalleri oluşturulmaktadır.

Program 1.

```
clear;
close all;
N=20;% Bir semboldeki örnek sayısı
p(1:N)=1;%kare darbe şeklinde birim vuruş tepkisi olan darbe şekillendirme
filtresi
bit=10;%iletilmek istenen bit sayısı
sembol=bit/2;%QPSK için sembol sayısı
veri=randsrc(2,sembol,[0 1]);% ilk satır d kanalı ikinci satır q kanalı
dk=veri(1,:);
qk=veri(2,:);
dk(dk==0)=-1/sqrt(2);%0 bitleri -1/sqrt(2) değeri ile temsil ediliyor
dk(dk==1)=1/sqrt(2);%1 bitleri 1/sqrt(2) değeri ile temsil ediliyor
qk(qk==0)=-1/sqrt(2);%0 bitleri -1/sqrt(2) değeri ile temsil ediliyor
qk(qk==1)=1/sqrt(2);%1 bitleri 1/sqrt(2) değeri ile temsil ediliyor
dkt(1:sembol*N)=0;%N semboldeki örnek sayısı kadar içinde sıfır olan dizi
qkt(1:sembol*N)=0;%N semboldeki örnek sayısı kadar içinde sıfır olan dizi
hold on;
for k=1:sembol
    dkt((k-1)*N+1)=dk(k);% d kanalındaki sembollere karşılık oluşturulan birim
    vuruşlar
    qkt((k-1)*N+1)=qk(k);%q kanalındaki sembollere karşılık oluşturulan birim
    vuruşlar
    subplot(4,1,1);
    plot(0:length(dkt)-1,dkt);
    xlabel('örnek numarası');
    ylabel('birim vuruş değeri');
    title('dk\delta(t-kT)');
    grid on;
    subplot(4,1,2);
    plot(0:length(qkt)-1,qkt);
    xlabel('örnek numarası');
    ylabel('birim vuruş değeri');
    title('qk\delta(t-kT)');
    grid on;
end;
dkt2=conv(dkt,p);
qkt2=conv(qkt,p);
subplot(4,1,3);
```

```

plot(0:length(dkt2)-1,dkt2);
    xlabel('örnek numarası');
    ylabel('sembol');
    title('Xdk(t)');
    grid on;
    subplot(4,1,4);
plot(0:length(qkt2)-1,qkt2);
    xlabel('örnek numarası');
    ylabel('sembol');
    title('Xqk(t)');
    grid on;
    hold off;

```

Öğrenci Görevi 1

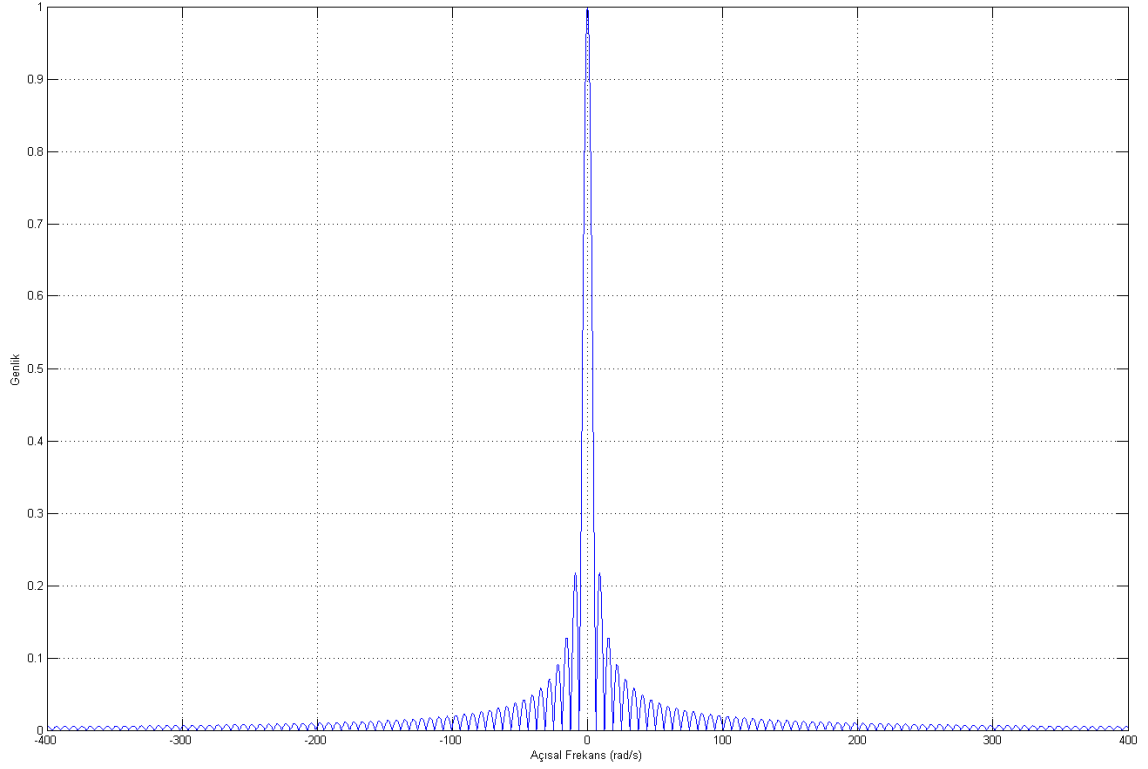
Kullanılan modülasyon BPSK olsaydı sayısal 1 bilgisine karşılık $d_k = +1$ sayısal 0 bilgisine karşılık $d_k = -1$ olacaktı. BPSK modülasyonunda q_k kanalı da olmayacaktı. Buna göre yukarıda verilen programdan faydalanarak kendiniz iletilmek istenen bit sayısını 10 olduğunu ve sembol başına 20 örnek kullanıldığını varsayarak program yazınız.

Lab Sorumlusu Onayı

Öte yandan darbe şekillendirme filtresinin frekans tepkisi incelendiğinde aşağıdaki genlik spektrumu elde edilir.

$$|P(\omega)| = \frac{T \left| \sin\left(\frac{\omega T}{2}\right) \right|}{\left| \frac{\omega T}{2} \right|}$$

$T = 1$ alınırsa elde edilen genlik spektrumu aşağıdaki gibi olacaktır.



Öğrenci Görevi 2

Program 1’i çalıştırıp elde edilen $x_{dk}(t)$ ’nin genlik spektrumunu çizdiriniz.

İpucu:

```
Xd=fft(dkt2);
```

```
>> plot(fftshift(abs(Xd)/length(Xd)))
```

$x_{dk}(t)$ işareti taşıyıcı $5\cos(2\pi 100t)$ ile çarpılmış olsaydı spektrum nasıl olurdu açıklayınız.

Lab Sorumlusu Onayı

Program 1’de $x_{dk}(t)$ ve $x_{qk}(t)$ taşıyıcılar ile çarpılmamıştır. Yeni bir program için new script deyip Program 1’i kopyala yapıştır ile bu boş dosyaya yapıştırınız ve bu dosyayı deney3_p2 ismi ile kaydediniz. Buradaki kodun baş tarafını aşağıdaki biçimde değiştiriniz.

```
T=1/8e2;%örnekleme periyodu
fs=1/T;
bit=16;%iletilmek istenen bit sayısı
sembol=bit/2;%QPSK için sembol sayısı
N=fs/10;% Bir semboldeki örnek sayısı
t=0:1:(N*sembol)+N-2;%zaman indeksleri
t=t*T;
p(1:N)=1;%kare darbe şeklinde birim vuruş tepkisi olan darbe şekillendirme filtresi
veri=randsrc(2,sembol,[0 1]);% ilk satır d kanalı ikinci satır q kanalı
```

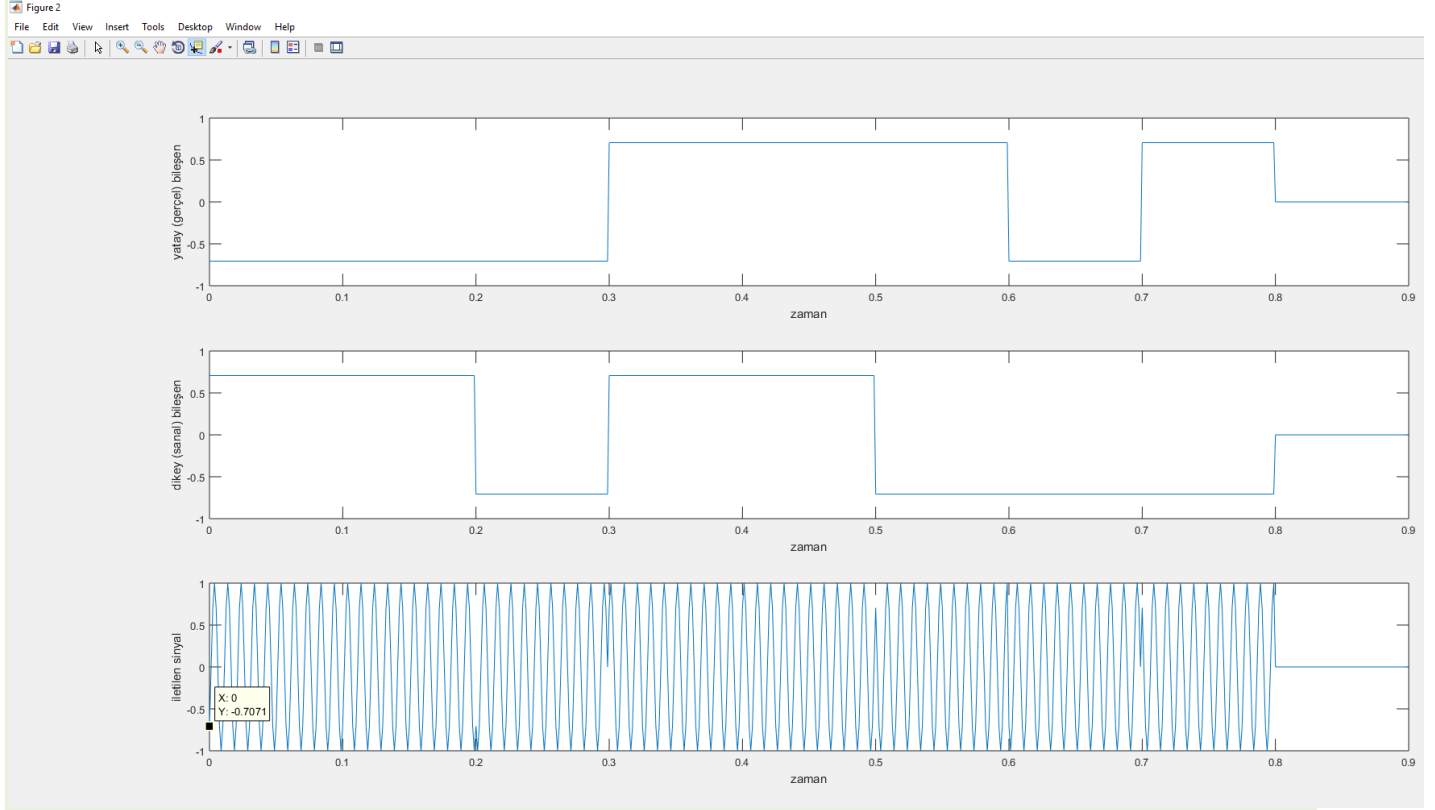
Kodun en altına aşağıdaki satırları ekleyiniz.

```
xck=cos(2*pi*1e2*t).*dkt2+sin(2*pi*1e2*t).*qkt2;%modülasyonlu işaret
figure(2);
subplot(3,1,1)
plot(t,dkt2)
xlabel('zaman');
ylabel('yatay (gerçek) bileşen');
grid on;
subplot(3,1,2)
plot(t,qkt2)
xlabel('zaman');
ylabel('dikey (sanal) bileşen');
grid on;
subplot(3,1,3)
plot(t,xck)
xlabel('zaman');
ylabel('iletilen sinyal');
grid on;
eyediagram(xck,N);
```

Öğrenci Görevi 3

Programı çalıştırınız. Figure 2’den faydalanarak yatay ve dikey bileşenlerin değerine göre iletilmiş olan $\cos(2\pi f_c t + \phi_k)$ sinyalindeki ϕ_k açılarını belirleyiniz.

İpucu:



Yukarıdaki şekilde $t = 0$ 'da iletilen sinyalin $t = 0$ 'da değeri -0.707 olarak görülmektedir. $t = 0$ 'da yatay bileşen -0.707 dikey bileşen 0.707 değerini almaktadır. Yani iletilen sembol $-0.707 + 0.707j$ 'dir. Buradan ϕ_k açısı $3\pi/4$ olarak bulunmaktadır. Sonuç olarak taşıyıcı sinyal $\cos(2\pi f_c t + \frac{3\pi}{4})$ olmakta ve $t = 0$ 'da -0.707 değerini almaktadır.

Sembol 0 Aç1 (ϕ_k)= $3\pi/4$

Sembol 1 Aç1 (ϕ_k)=.....

Sembol 2 Aç1 (ϕ_k)=.....

Sembol 3 Aç1 (ϕ_k)=.....

Lab Sorumlusu Onayı

Öğrenci Görevi 4

İletilen sinyalin Fourier dönüşümünü alıp frekansa göre genlik spektrumunu çizdiriniz.

İpucu:

```
>>figure(5);  
  
>>xckf=fft(xck(1:sembol*N));  
  
>> k=-sembol*N/2:1:sembol*N/2-1;  
  
>> f=k*fs/(sembol*N);  
  
>> plot(f,fftshift(abs(xckf))/(sembol*N))
```

Lab Sorumlusu Onayı

Bilindiği gibi zamanda katlama işlemi frekansta çarpma işlemine eşdeğerdir. Dolayısı ile darbe şekillendirme filtrelerinin girişindeki birim vuruş işaretlerinin Fourier dönüşümleri ile filtrelerin frekans tepkeleri çarpılırsa sonuçta yine yukarıdaki gibi bir spektrum elde edilir. Dikkat edilirse ana lob'un yanında birçok yan lob bulunmakta ve bunların bazılarının genlikleri ana lobun genliğinin 1/5'inden büyük olmaktadır.

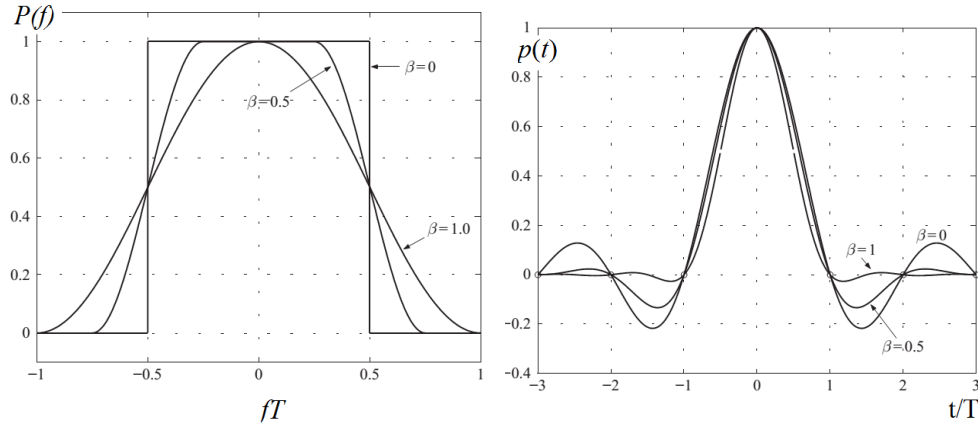
Darbe şekillendirme filtresinin frekans tepkisi sinc'ten daha az yan band'lara kaçak yapan bir işaret seçilebilir. Popüler olarak kullanılan darbe şekillendirme filtresi raised-cosine filtredir. Raised cosine filtrenin birim vuruş tepkisi aşağıdaki gibidir.

$$p_{RC}(t) = \frac{\sin\left(\frac{\pi t}{T}\right)}{\left(\frac{\pi t}{T}\right)} \frac{\cos\left(\frac{\pi \beta t}{T}\right)}{1 - \frac{4\beta^2 t^2}{T^2}} = \text{sinc}(\pi t/T) \frac{\cos\left(\frac{\pi \beta t}{T}\right)}{1 - \frac{4\beta^2 t^2}{T^2}}$$

Raised cosine filtrenin frekans tepkisi ise aşağıdaki gibidir.

$$P_{RC}(f) = \begin{cases} T, & |f| \leq \frac{1-\beta}{2T} \\ T \cos^2 \left[\frac{\pi T}{2\beta} \left(|f| - \frac{1-\beta}{2T} \right) \right], & \frac{1-\beta}{2T} \leq |f| \leq \frac{1+\beta}{2T} \\ 0, & |f| \geq \frac{1+\beta}{2T} \end{cases}$$

Raised-cosine filtrede β parametresi yan bandlardaki kaçak miktarını kontrol eder. $p_{RC}(t)$ ifadesine dikkat edilecek olursa $\beta = 0$ için $p_{RC}(t)$ sinc olmaktadır. Bu durumda yan bantlara hiç kaçak olmayacak ancak zamanda sinc elde edileceğinden sembol süresi sonunda da sinc işareti devam edeceğinden semboller arası girişim olacaktır. Aşağıda raised-cosine filtrenin birim vuruş tepkisi ile frekans tepkisi gösterilmektedir.



Yukarıdaki grafikler incelendiğinde $p_{RC}(t)$ yada grafikteki ismiyle $p(t)$ 'nin nedensel olmayan bir işaret olduğu görülür. $p_{RC}(t)$ mT (T sembol periyodu) kadar geciktirilirse nedensel bir işaret elde edilir. Ayrıca teorik olarak $p_{RC}(t)$ sonsuz uzunluktadır ancak pratikte bu mümkün değildir. Bu sebeple $p_{RC}(t)$ 'yi $2mT$ uzunluğunda olacak şekilde kesebiliriz. Buradaki m tamsayı değeri kolaylık ve doğruluk arasında bir denge parametresidir. Örnekleme periyodu T_s ve bir semboldeki örnek sayısı k olarak alınırsa $T = kT_s$ olur. Ayrıca gecikmeden ötürü t yerine $t - mT$ yazılır ve $t = nT_s$ ile ayrık zamana geçilirse $p_{RC}(t)$ aşağıdaki gibi elde edilir.

$$p_{RC}(t - mT)|_{t=nT_s} = \frac{\sin\left(\frac{\pi(nT_s - mkT_s)}{kT_s}\right) \cos\left(\frac{\pi\beta(nT_s - mkT_s)}{kT_s}\right)}{\frac{\pi(nT_s - mkT_s)}{kT_s} \left(1 - \frac{4\beta^2(nT_s - mkT_s)^2}{k^2T_s^2}\right)} = \frac{\sin\left(\pi\left(\frac{n}{k} - m\right)\right) \cos\left(\pi\beta\left(\frac{n}{k} - m\right)\right)}{\pi\left(\frac{n}{k} - m\right) \left(1 - 4\beta^2\left(\frac{n}{k} - m\right)^2\right)}$$

Sonuç olarak raised cosine filtreye ilişkin birim vuruş tepkesi ayrık zamanlı ve m sembol kadar geciktirilmiş olarak aşağıdaki gibi olacaktır.

$$p_{RC}(n) = \frac{\sin\left(\pi\left(\frac{n}{k} - m\right)\right) \cos\left(\pi\beta\left(\frac{n}{k} - m\right)\right)}{\pi\left(\frac{n}{k} - m\right) \left(1 - 4\beta^2\left(\frac{n}{k} - m\right)^2\right)}$$

Yeni bir program için new script deyip deney3_p2'deki programı kopyala yapıştır ile bu boş dosyaya yapıştırdınız ve bu dosyayı deney3_p3 ismi ile kaydediniz. Kodun en altına aşağıdaki satırları yapıştırdınız.

```
beta = 0.95;
m=5;
k=N;
n = 0:2*m*k;
z = (n/k)-m+eps;
t1 = cos(beta*pi*z);
t2 = sin(pi*z)./(pi*z);
t3 = 1-4*beta*beta*z.*z;
prc = t2.*t1./(t3);%raised cosine filtresinin birim vuruş tepkesi
```



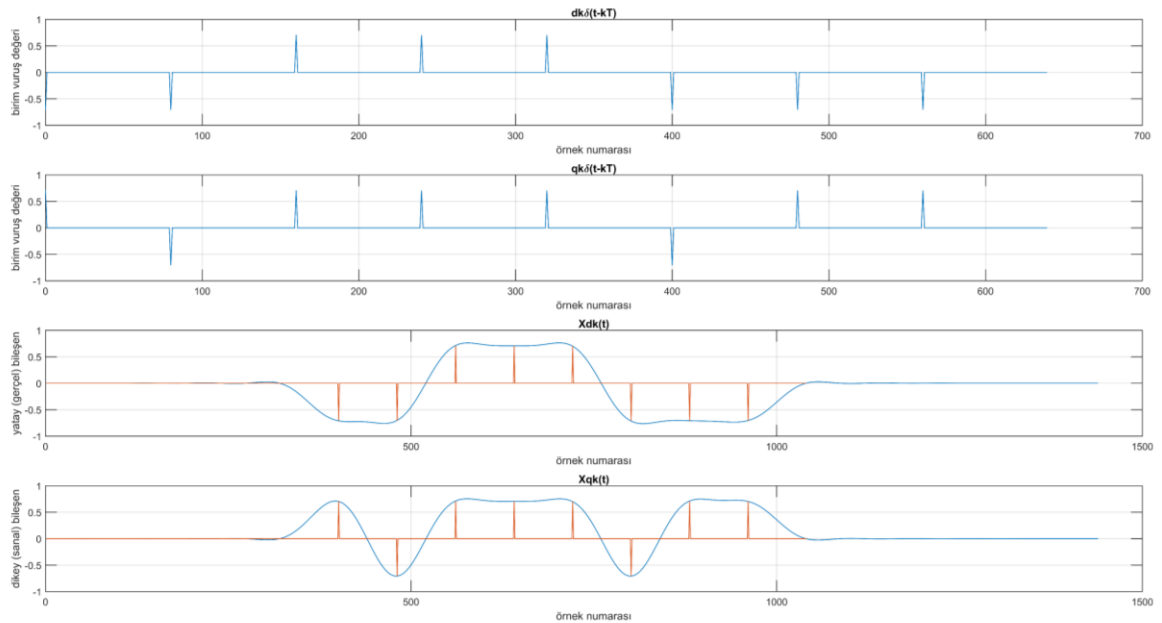
```

dktr=conv(dkt,prc);
qktr=conv(qkt,prc);
figure(4);
subplot(4,1,1);
plot(0:length(dkt)-1,dkt);
xlabel('örnek numarası');
ylabel('birim vuruş değeri');
title('dk\delta(t-kT)');
grid on;
subplot(4,1,2);
plot(0:length(qkt)-1,qkt);
xlabel('örnek numarası');
ylabel('birim vuruş değeri');
title('qk\delta(t-kT)');
grid on;

subplot(4,1,3);
plot(0:length(dktr)-1,dktr);
xlabel('örnek numarası');
ylabel('yatay (gerçek) bileşen');
title('Xdk(t)');
hold on;
dl(1:k*m)=0;
dl=[dl dkt];
plot(dl);
hold off;
grid on;
subplot(4,1,4);
plot(0:length(qktr)-1,qktr);
xlabel('örnek numarası');
ylabel('dikey (sanal) bileşen');
title('Xqk(t)');
hold on;
ql(1:k*m)=0;
ql=[ql qkt];
plot(ql);
grid on;
hold off;

```

Programı koşturunuz. Aşağıdaki gibi grafikler elde etmelisiniz.



En alttaki iki grafik $x_{dk}(t)$ ve $x_{qk}(t)$ sinyallerini göstermektedir. Görüldüğü gibi örnekleme zamanlamasına karşı duyarlılık artmıştır. Kırmızı çizgiler doğru örnekleme noktalarını göstermektedir.

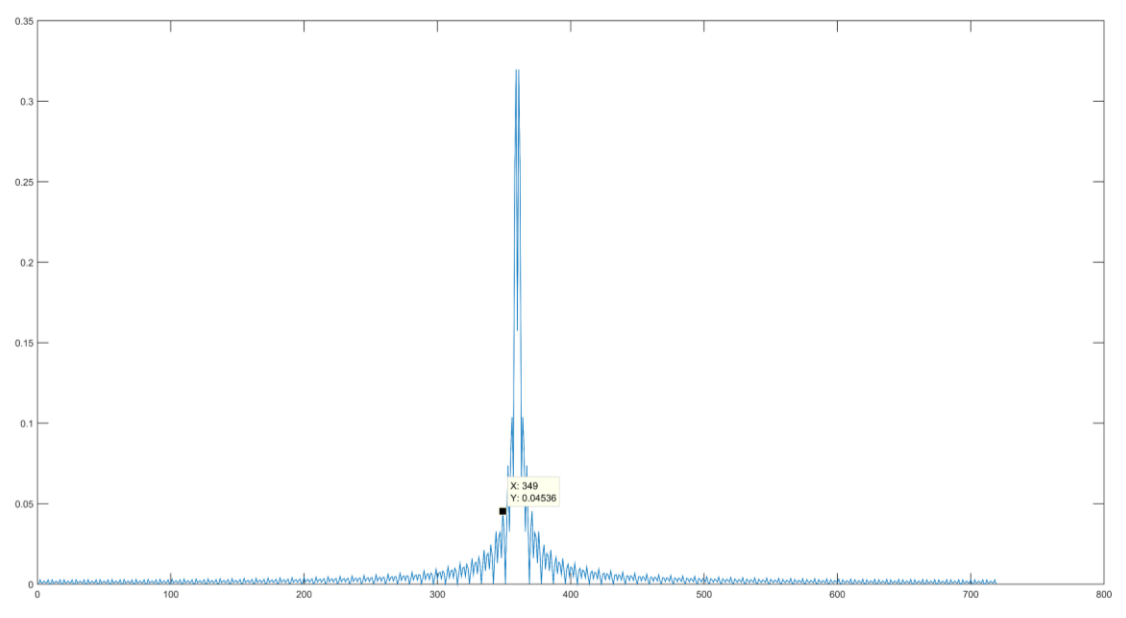
Öğrenci Görevi 4

$x_{dk}(t)$ sinyalinin Fourier dönüşümünü hem darbe şekillendirme filtresi olarak birim vuruş tepkisi kare darbe olan filtre hem de raised cosine darbesi olan filtre kullanılması durumunda çizdiriniz.

İpucu: Kare darbe kullanılması durumu.

```
Xd=fft(dkt2);
```

```
plot(fftshift(abs(Xd)/length(Xd)))
```

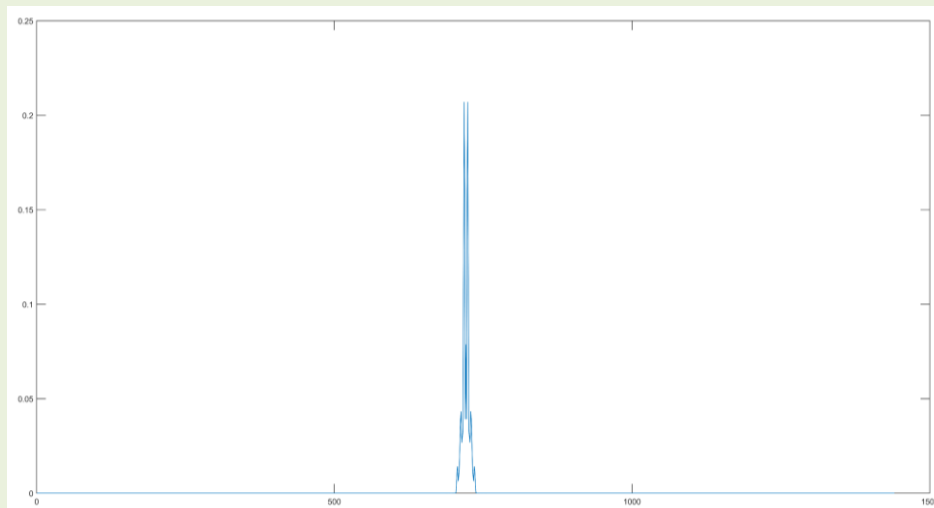


Yukarıdaki gibi bir şekil elde etmelisiniz.

Raised cosine darbesi kullanılması durumu.

```
Xd2=fft(dktr);
```

```
>> plot(fftshift(abs(Xd2)/length(Xd2)))
```



Yukarıdaki gibi bir şekil elde etmelisiniz.

Programdaki beta parametresini ve raised cosine darbesinin uzunluğunu ($2 \cdot m \cdot k$ 'lı kısımda) değiştirerek zamanlamaya karşı duyarlılığın ve spektrumun nasıl değiştiğini raised cosine darbe kullanılması durumu için inceleyiniz.

Lab Sorumlusu Onayı