به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

اصول سیستم های مخابراتی

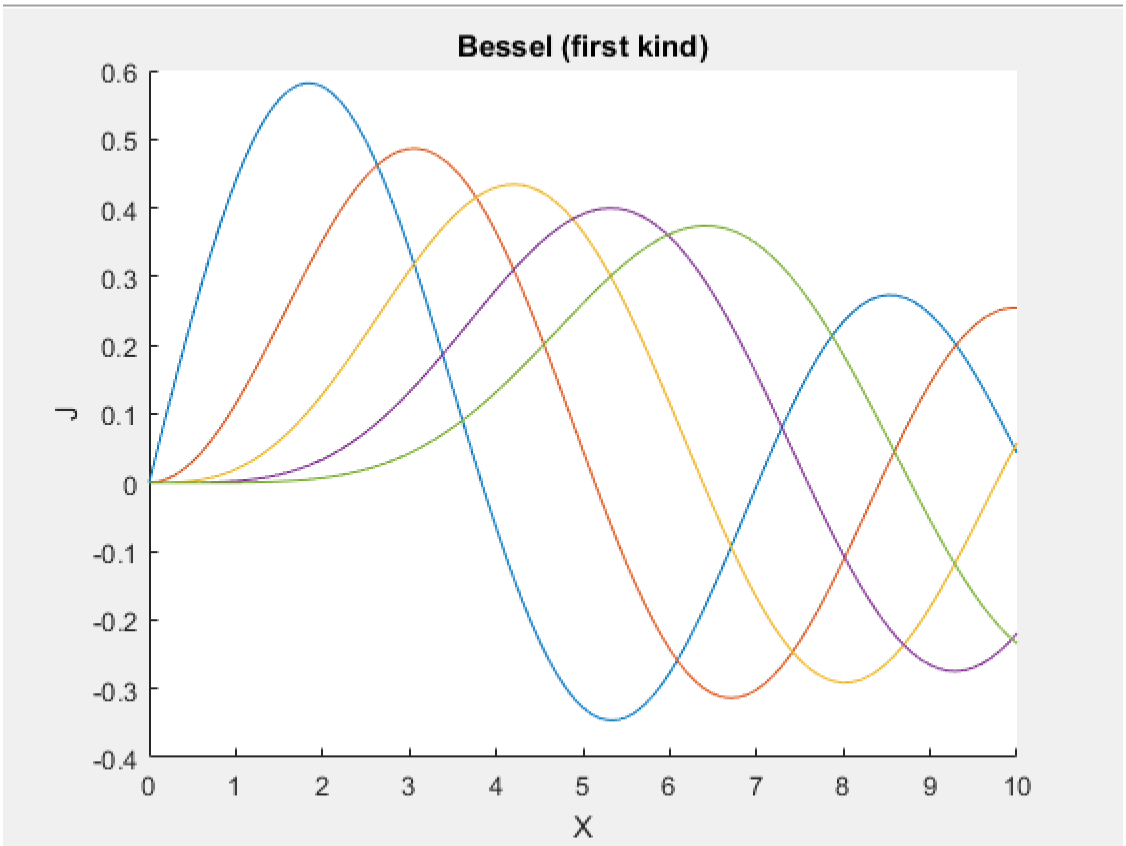
تمرین کامپیوتری اول

نام و نام خانوادگی	محمدرضا بختیاری
شماره دانشجویی	810197468
تاریخ ارسال گزارش	1399/9/29

فهرست گزارش سوالات

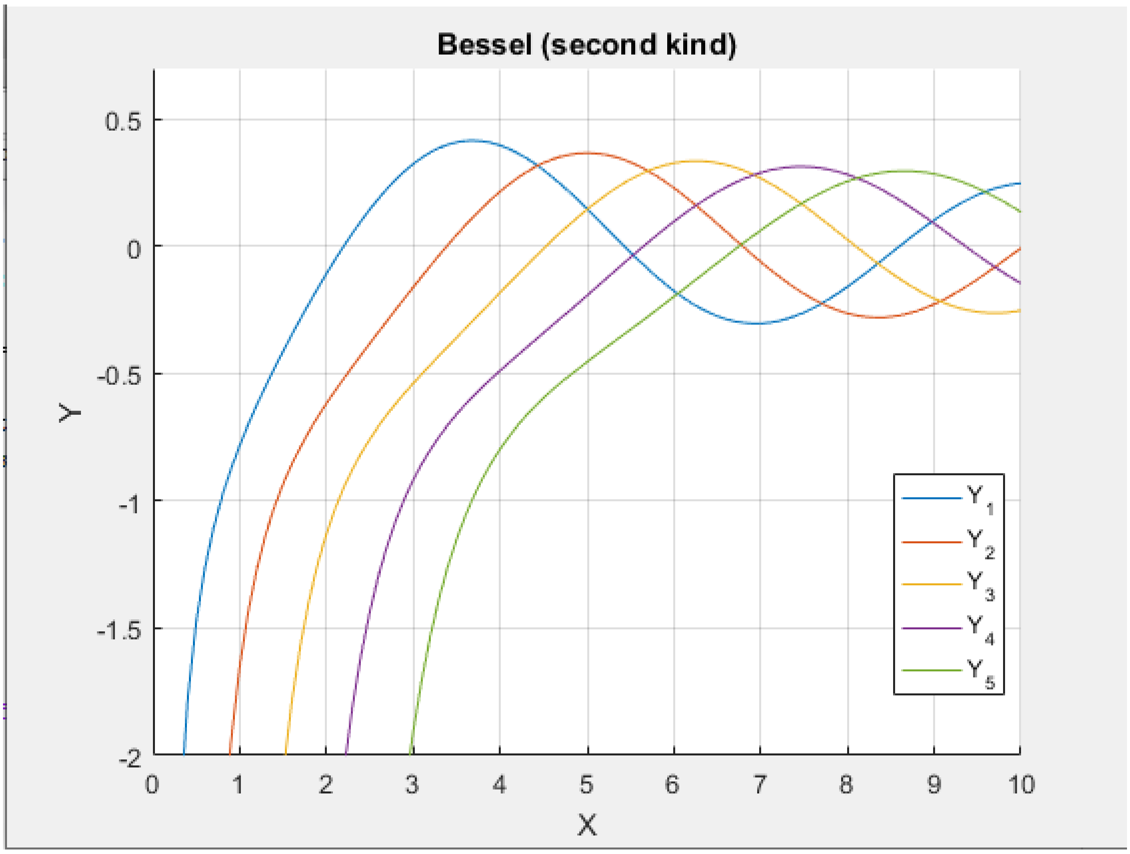
Bessel first kind :

```
%% Part_1
J = [] ;
Z = [0 : 0.1 : 10] ;
hold on
for nu = 1:5                        %calculate J and plot
    J = besselj(nu,Z) ;
    plot(Z,J) ;
end
title(' Bessel (first kind) ')
xlabel('X');
ylabel('J');
% grid on
% legend('J_1','J_2','J_3','J_4','J_5','Location','Best')
hold off
```



Bessel second kind :

```
%% Part_2
Y = [] ;
Z = [0 : 0.1 : 10] ;
hold on
for nu = 1:5                        %calculate Y and plot
    Y = bessely(nu,Z) ;
    plot(Z,Y) ;
    axis([0 10 -2 0.7])
end
title(' Bessel (second kind) ')
xlabel('X');
ylabel('Y');
grid on
legend('Y_1','Y_2','Y_3','Y_4','Y_5','Location','Best')
hold off
```



pmm function :

```
function y = p3_pmm(x , Ac , fc , kp , fs)
    t = 0 : 1/fs : (1/fs)*(length(x)-1) ;
    y = Ac*cos((2*pi*fc*t)+(x.*kp)) ;
end
```

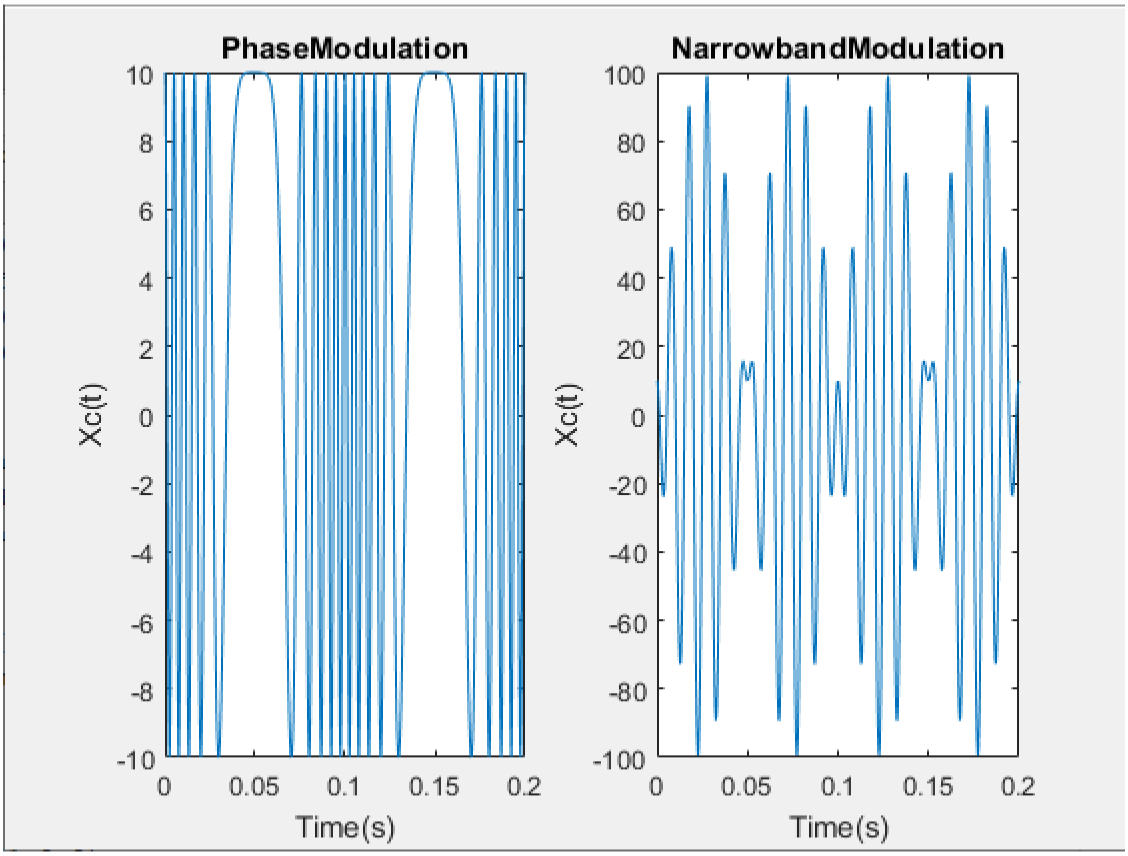
nb function :

```
function y = p4_nb(x , Ac , fc , kp , fs)
    t = 0 : 1/fs : (1/fs)*(length(x)-1) ;
    y = Ac*cos(2*pi*fc*t) - Ac*kp*x.*sin(2*pi*fc*t) ;
end
```

modulating message signal :

```
%% Part_5_6_7
fs = 10000 ;
Ts = 1/fs ;            % Sampling period
time = 0:Ts:0.2 ;
fc = 100 ;
Ac = 10 ;
kp = 10 ;
Xm = sin(20*pi*time) ;
PhaseModulation = p3_pmm(Xm , Ac , fc , kp , fs) ;
NarrowbandModulation = p4_nb(Xm , Ac , fc , kp , fs) ;

figure;
subplot(1,2,1);
plot(time,PhaseModulation) ;
title(' PhaseModulation ')
xlabel('Time(s)');
ylabel('Xc(t)');

subplot(1,2,2);
plot(time,NarrowbandModulation) ;
title(' NarrowbandModulation ')
xlabel('Time(s)');
ylabel('Xc(t)');
```

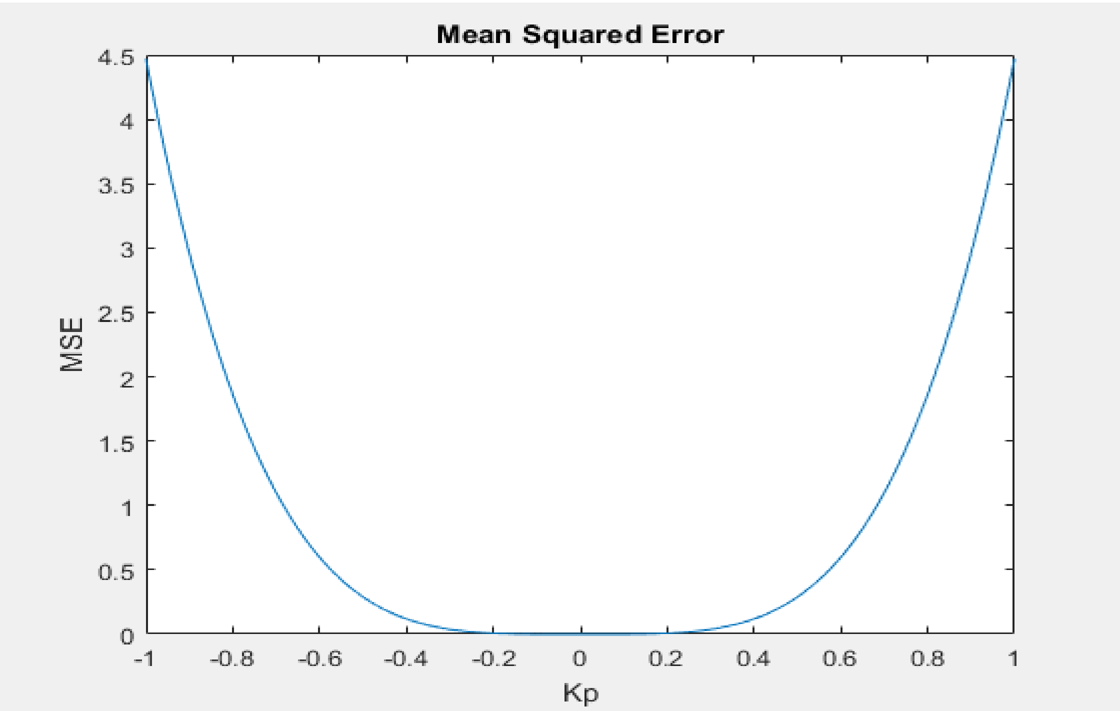
modulated signal :



As mentioned in the question, the condition for using the Narrow Band function is that the value of phi is small (phi << 1)
but in this case phi is 10 !
so we can no longer use this approximation , And this is the reason for the difference between the two plots .

Mean Squared Error :

```
%% Part_8_9
fs = 10000 ;
Ts = 1/fs ;             % Sampling period
time = 0:Ts:0.2 ;
fc = 100 ;
Ac = 10 ;
Xm = sin(20*pi*time) ;
Kp = -1:0.001:1 ;        % Kp --> [-1 , 1]
MSE = [] ;              % Mean Squared Error
MaxKp = 0 ;             % Maximum Kp to have 1 percent error
for kp = -1:0.001:1
    PhaseModulation = p3_pmm(Xm , Ac , fc , kp , fs) ;
    NarrowbandModulation = p4_nb(Xm , Ac , fc , kp , fs) ;
    MSE(end + 1) =  immse( PhaseModulation , NarrowbandModulation ) ;
    if immse( PhaseModulation , NarrowbandModulation ) < 0.1
        MaxKp = kp ;
    end
end

plot(Kp,MSE) ;
title(' Mean Squared Error ')
xlabel('Kp');
ylabel('MSE');
```



As we expected, with increasing the amount of Kp, the difference between the two modulations increases, and finally, the amount of error also increases.

```
MaxKp =  0.3820
```

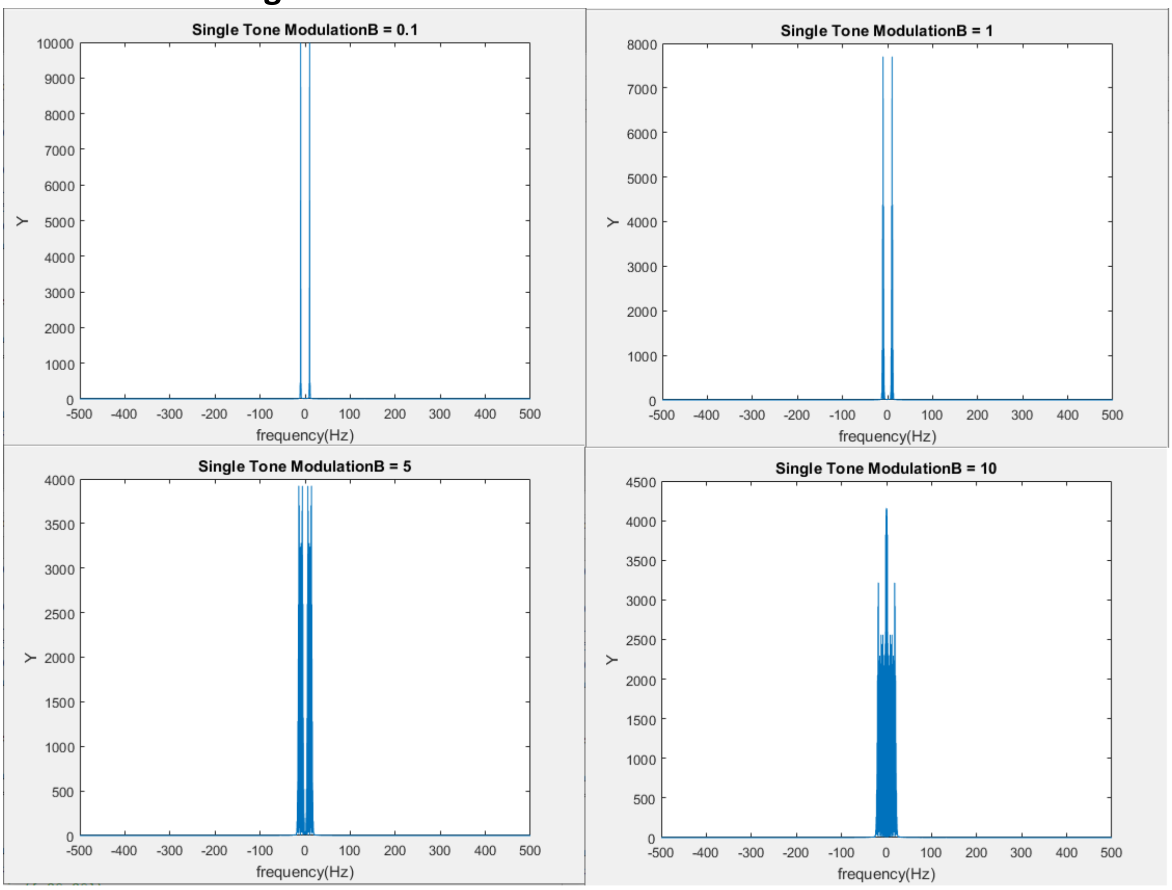
Single Tone Modulation

Single Tone Modulation :

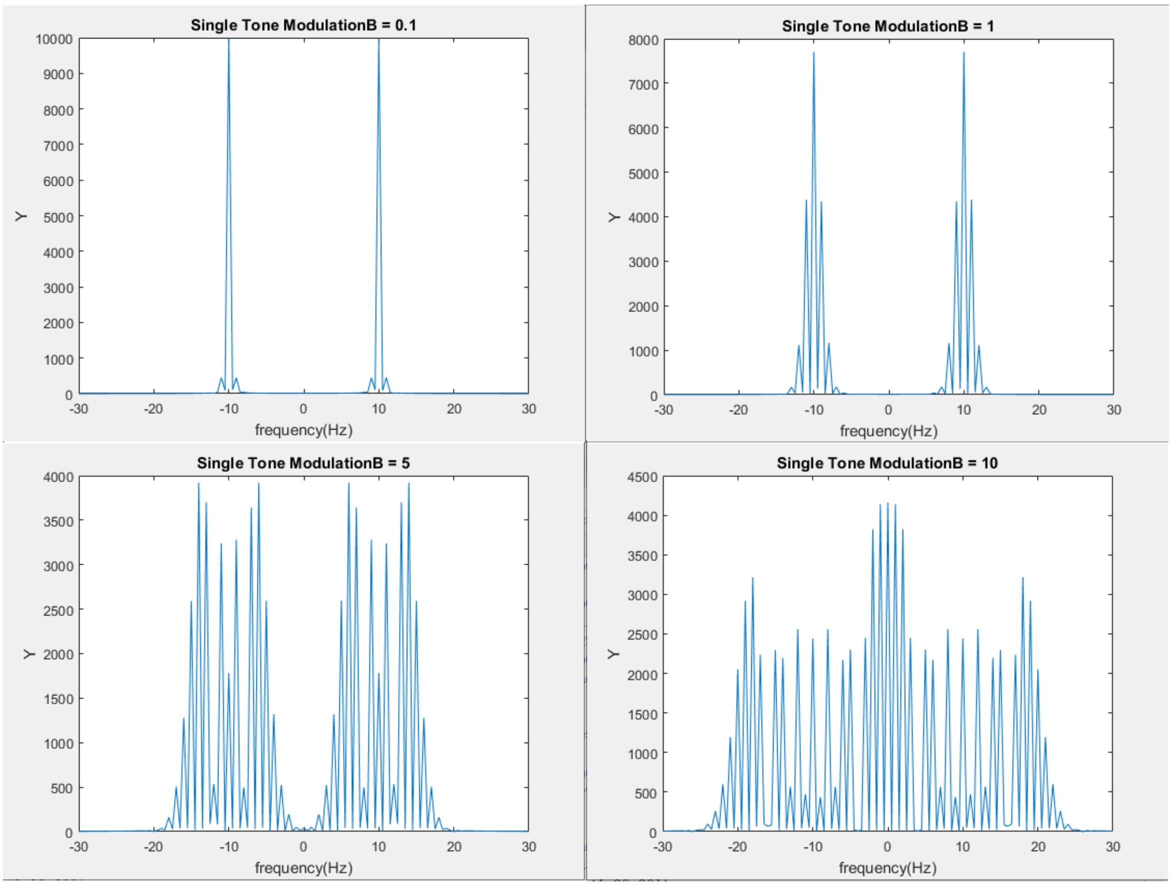
```
%% Part_10
fs = 10000 ;
Ac = 10 ;
fc = 100 ;
B = [0.1 ,1 ,5 ,10] ;
Ts = 1/fs ;             % Sampling period
time = 0:Ts:0.2 ;
Xm = sin(20*pi*time) ;
fm = 10 ;
Xc = [] ;
frequency = -500:0.5:500 ;

for Counter = 1:1:4
    Xc = Ac * cos((2*pi*fc*time)+(B(Counter)*sin(2*pi*fm*time))) ;
    Y = fft(Xc) ;
    Y = fftshift(Y) ;
    figure;
    plot(frequency,abs(Y)) ;
    xlim([-30 30])
    title(['Single Tone Modulation','B = ',num2str(B(Counter))])
    xlabel('frequency(Hz)');
    ylabel('Y');
end
```

Note : To make the diagrams more accurate, we use the “xlim” function to limit the horizontal axis.

Plots before using “xlim” :



Plots after using “xlim” :

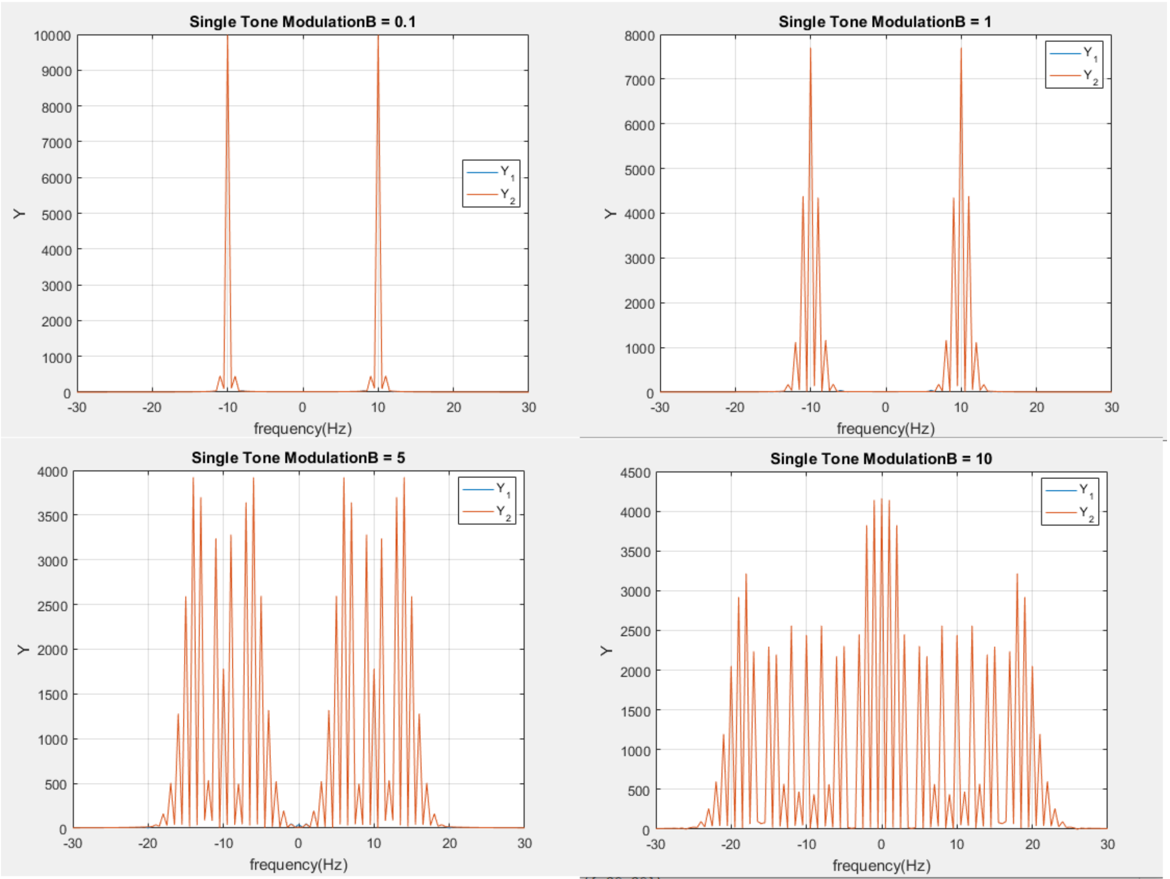


As mentioned in the question, with increasing beta, the number of sentences increases and we can not ignore some sentences, which is why the signal is wider.

modulating message signal using functional expansion :

```
%% Part_11_12
fs = 10000 ;
Ac = 10 ;
fc = 100 ;
B = [0.1 ,1 ,5 ,10] ;
Ts = 1/fs ;             % Sampling period
time = 0:Ts:0.2 ;
Xm = sin(20*pi*time) ;
fm = 10 ;
Xc = [] ;
Bc = [] ;
N = [1 , 3 , 9 , 23] ;
frequency = -500:0.5:500 ;

for Counter = 1:1:4
    Xc = Ac * cos((2*pi*fc*time)+(B(Counter)*sin(2*pi*fm*time))) ;
    Y = fft(Xc) ;
    Y = fftshift(Y) ;
    figure;
    plot(frequency,abs(Y)) ;
    hold on
    title(['Single Tone Modulation','B = ',num2str(B(Counter))])
    xlabel('frequency(Hz)');
    ylabel('Y');

    Xc = 0 ;
    for n = -N(Counter):1:N(Counter)
        Xc = Xc + ( besselj(n,B(Counter)) * cos(2*pi*(fc + n*fm)*time) ) ;
    end
    Xc = Xc * Ac ;
    Y = fft(Xc) ;
    Y = fftshift(Y) ;
    plot(frequency,abs(Y)) ;
    grid on
    legend('Y_1','Y_2','Location','Best')
    hold off
    Bc(Counter) = 2 * (B(Counter) + 1) * fm ;
end
```

Note : To make the diagrams more accurate, we use the "xlim" function to limit the horizontal axis.

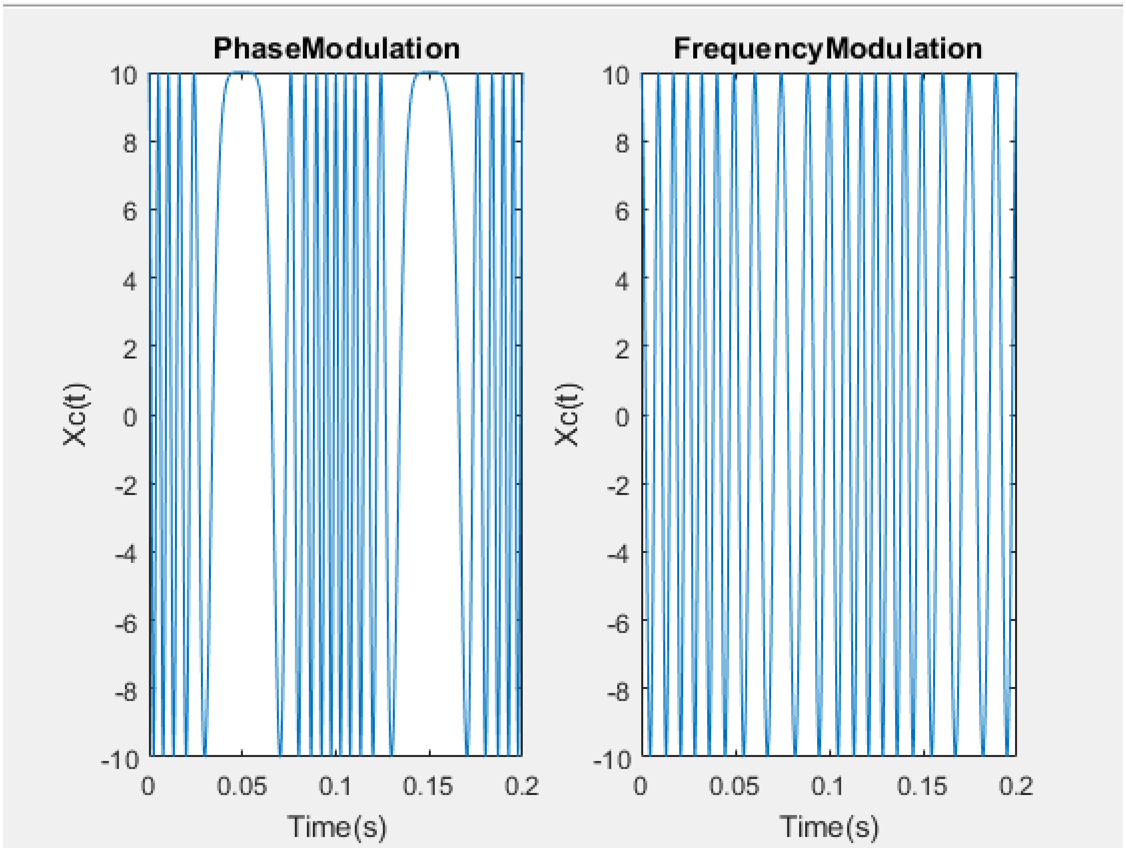


fmm function :

```
function y = p3_fmm(x , Ac , fc , kf , fs)
    N = 100 ;
    Integral = 0 ;
    for Counter = 1:1:N
        Integral = Integral + x(N) + x(N+1) ;
    end
    Integral = ((0.2-0)/(2*N)) * Integral ;
    Phi = 2*pi*kf*Integral ;
    y = p3_pmm(Phi , Ac , fc , 1 , fs) ;

end
```

pm modulation and fm modulation :

```
%% Part_14
fs = 10000 ;
Ts = 1/fs ;            % Sampling period
time = 0:Ts:0.2 ;
fc = 100 ;
Ac = 10 ;
kp = 10 ;
kf = 100 ;
Xm = sin(20*pi*time) ;
PhaseModulation = p3_pmm(Xm , Ac , fc , kp , fs) ;
FrequencyModulation = p3_fmm(Xm , Ac , fc , kf , fs) ;

figure;
subplot(1,2,1);
plot(time,PhaseModulation) ;
title(' PhaseModulation ')
xlabel('Time(s)');
ylabel('Xc(t)');

subplot(1,2,2);
plot(time,FrequencyModulation) ;
title(' FrequencyModulation ')
xlabel('Time(s)');
ylabel('Xc(t)');
```



Armstrong Indirect Method

مشترک و ویژگی: فرکانس موج حامل به صورت جدا از $x(t)$ تولید می‌شود و ربطی به بقیه مدار ندارد

① NBPM $\xrightarrow{\text{انتگرال‌گیر}}$ ② NBFM $\xrightarrow[\text{فرکانسی}]{\text{ضرب‌کننده‌ی}}$ ③ WBFM

$$f_i \longrightarrow \boxed{\times n} \longrightarrow f_o \qquad n f_i = f_o$$

① Narrowband PM: فرکانس موج حامل: $\omega_{c_1} \rightsquigarrow f_{c_1}$

$$x_c(t) = A_c \cos[\omega_{c_1} t + \phi(t)] = A_c \cos[\omega_{c_1} t + \Phi_\Delta x(t)]$$

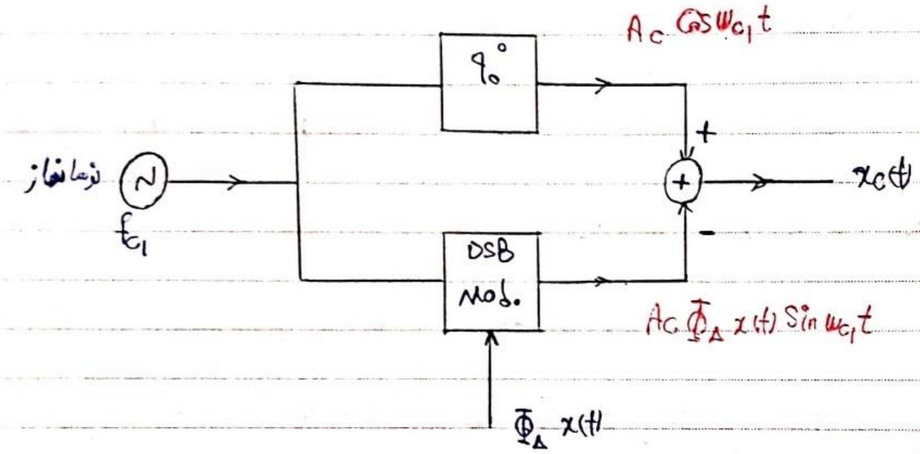
NB: $\Phi_\Delta x(t) \ll 1$

$$x_c(t) = A_c \underbrace{\cos(\Phi_\Delta x(t))}_{1} \cos\omega_{c_1} t - A_c \underbrace{\sin(\Phi_\Delta x(t))}_{\Phi_\Delta x(t)} \sin\omega_{c_1} t$$

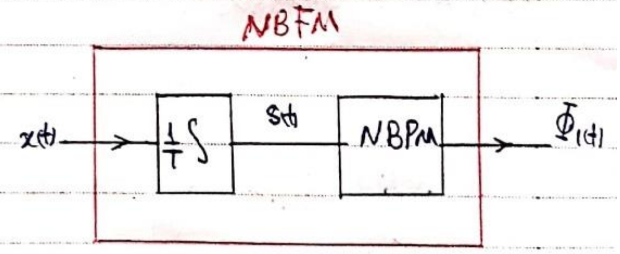
با مدولاتور DSB می‌توان این شکل موج را تولید کرد.

$$\boxed{x_c(t) = A_c \cos\omega_{c_1} t - A_c \Phi_\Delta x(t) \sin\omega_{c_1} t}$$

NBPM $\downarrow$ NBFM ②



2_1: NBPM $\longrightarrow$ NBFM

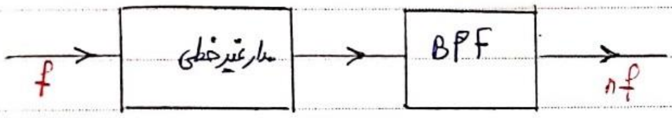


$$s(t)=\frac{1}{T}\int x(\lambda)\,d\lambda \qquad \Phi_1(t)=\Phi_\Delta\, s(t)=\frac{\Phi_\Delta}{T}\int x(\lambda)\,d\lambda$$

$$f_1(t)=f_{c_1}+\underbrace{\frac{\Phi_\Delta}{2\pi T}}_{f_\Delta}\,x(t)$$

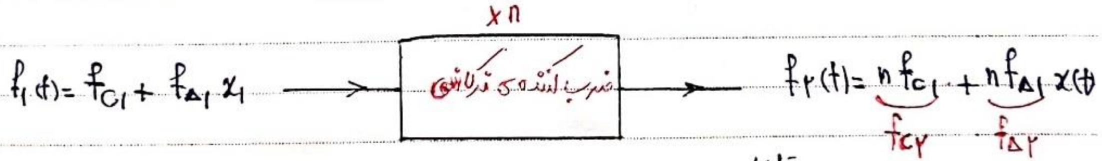
3ـ ضرب کننده‌ی فرکانس : با یک عنصر غیرخطی فرکانس را N برابر می‌کنیم.

غالباً ترکیبی از ۲ برابر کننده و ۳ برابر کننده‌ی فرکانسی است.



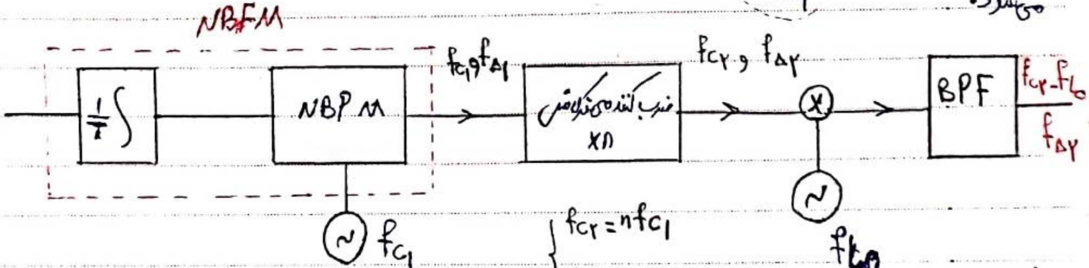
۱۰۰۰ برابر کننده‌ی فرکانس : ۱۰ تا دوبرابر کننده $\quad f\times 2^{10}$

مثال : برای اینکه f_{Δ_2} برابر مقدار خواسته شده یعنی f_Δ باشد، n باید خیلی بزرگ باشد :



NBPM Φ_Δ کوچک $\xrightarrow{\frac{1}{T}\int d\lambda}$ کوچک f_{Δ_1} $\xrightarrow{\times n}$ f_Δ

مقدار زیاد $n=\frac{f_\Delta}{f_{\Delta_1}}$ ⟵ f_{c_2} خیلی بزرگ می‌شود.



$$\begin{cases} f_{c_2}=nf_{c_1} \\ f_{\Delta_2}=nf_{\Delta_1}\end{cases}$$