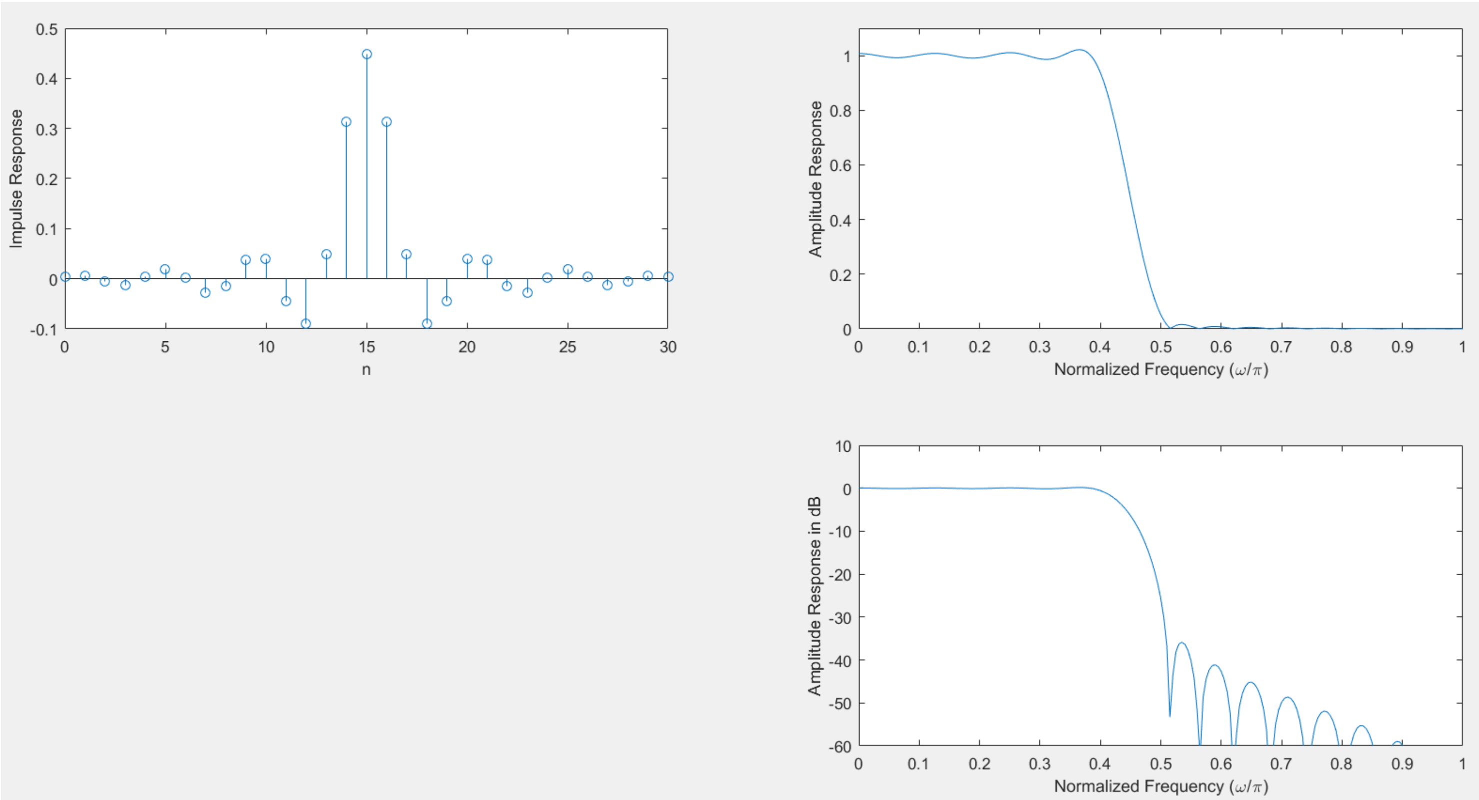
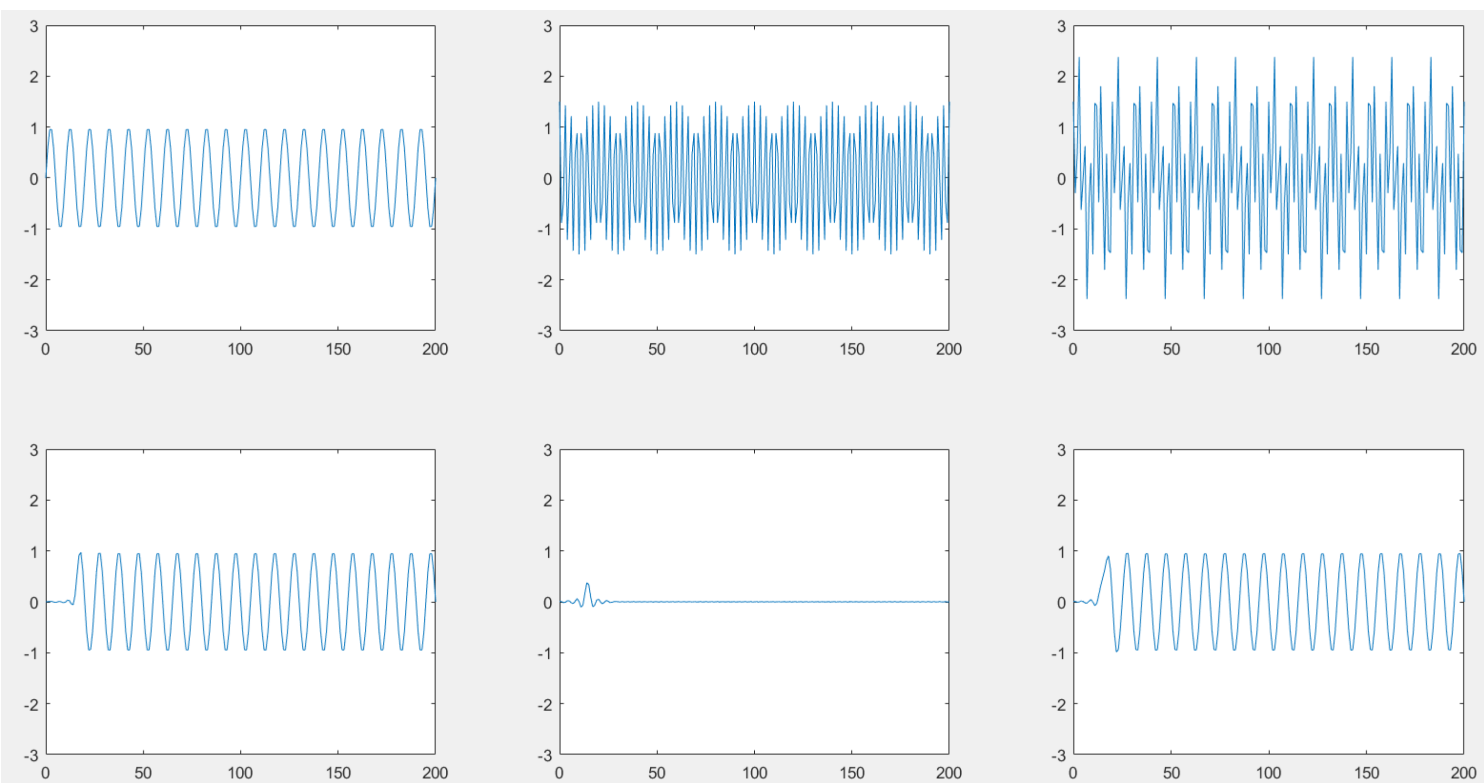
```
%                          DSP_Project_1
%
% Design of Case 1 Linear-Phase FIR Digital Filter(odd,symmetric)
%
clear all;   % clear workspace
clc;         % clear command window
N = 31;
wp = 0.4*pi;
ws = 0.5*pi;
% Wp = Ws = 1;
%
NH = (N-1)/2;
%
%
P = zeros(NH+1,1);
Qp = zeros(NH+1,NH+1);
Qs = zeros(NH+1,NH+1);
for i=0:NH % 0:1:NH
    if i==0
        P(i+1) = -2*wp;
    else
        P(i+1) = -2*sin(i*wp)/i;
    end
    for j=0:NH
       if i==0 && j==0
          Qp(i+1,j+1) = wp;
          Qs(i+1,j+1) = pi - ws;
       elseif i==j
          Qp(i+1,j+1) = 0.5*wp + 0.5*sin((i+j)*wp)/(i+j);
          Qs(i+1,j+1) = 0.5*(pi - ws) - 0.5*sin((i+j)*ws)/(i+j);
       else
          Qp(i+1,j+1) = 0.5*sin((i-j)*wp)/(i-j) +
0.5*sin((i+j)*wp)/(i+j);
          Qs(i+1,j+1) = -0.5*sin((i-j)*ws)/(i-j) -
0.5*sin((i+j)*ws)/(i+j);
       end
    end
end
Q = Qp + Qs;
A = -0.5*inv(Q)*P;
%
%
h = zeros(N,1);
h(NH+1) = A(1);
h(1:NH) = 0.5*A(NH+1:-1:2);
h(NH+2:N) = 0.5*A(2:NH+1);
%
```

```
% need to show several plots simultaneously
subplot(2,2,1);
stem(0:N-1,h);
xlabel('n');
ylabel('Impulse Response');
%
% Amplitude Response
subplot(2,2,2);
AR = abs(freqz(h,1,0:pi/200:pi));
plot(0:1/200:1,AR);
axis([0,1,0,1.1]);
xlabel('Normalized Frequency (\omega/\pi)');
ylabel('Amplitude Response');
%
% Amplitude Response in dB
subplot(2,2,4);
plot(0:1/200:1,20*log10(AR));
axis([0,1,-60,10]);
xlabel('Normalized Frequency (\omega/\pi)');
ylabel('Amplitude Response in dB');
pause;
close all; % close formal plots
%
% signal simulation
%
% low frequency input 1
NT = (0:200)';
xL = sin(0.2*pi*NT);
subplot(2,3,1);
plot(NT,xL);
axis([0,200,-3,3]);
%
% low frequency output 2
yL = filter(h,1,xL);
subplot(2,3,4);
plot(NT,yL); % plot shows input delay
axis([0,200,-3,3]);
%
% high frequency input 3
xH = 1.5*cos(0.7*pi*NT);
subplot(2,3,2);
plot(NT,xH);
axis([0,200,-3,3]);
%
% high frequency output 4
yH = filter(h,1,xH);
subplot(2,3,5);
```

```
plot(NT,yH); % plot shows transition time
axis([0,200,-3,3]);
%
% high and low combined input 5
subplot(2,3,3);
plot(NT,xL + xH);
axis([0,200,-3,3]);
%
% high and low combined output 6
y = filter(h,1,xL + xH);
subplot(2,3,6);
plot(NT,y); % plot 6 is pretty the same with 2
axis([0,200,-3,3]);
%
%
%
```

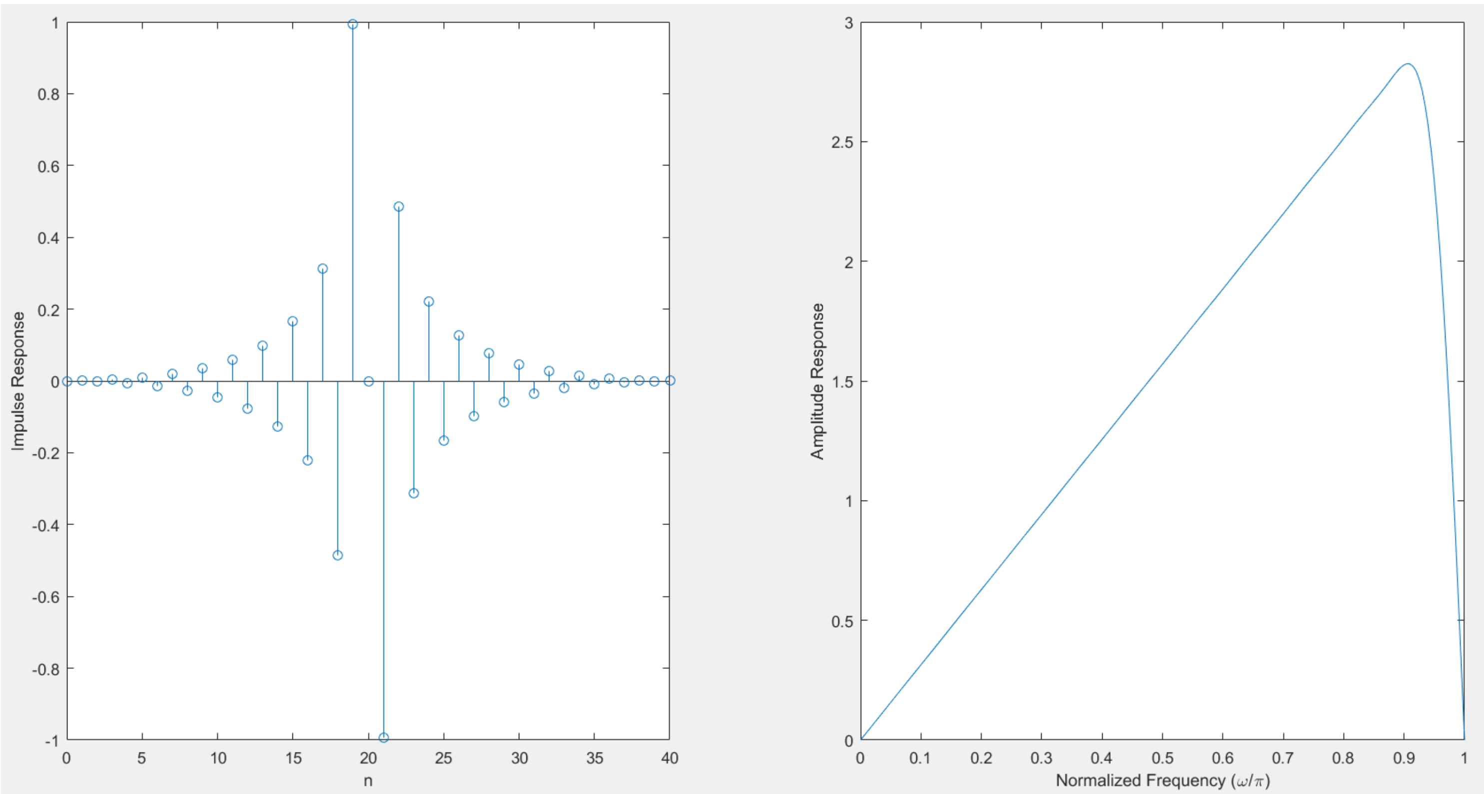
```
% Case 1 Linear-Phase FIR Digital Filter
```



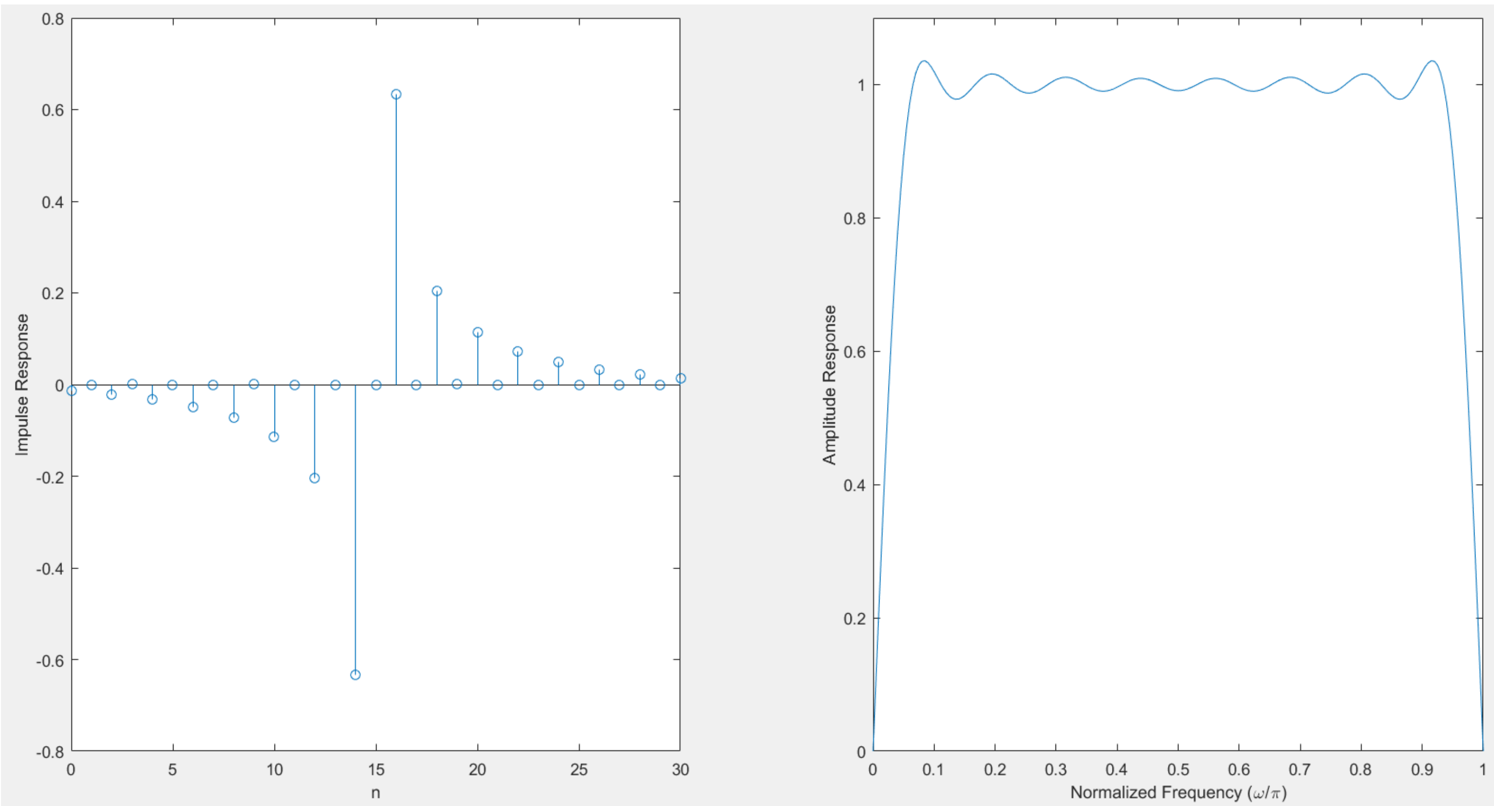
```
%
% Design of Case 3 Differentiator
%
clear all;   % clear workspace
clc;         % clear command window
N = 41;
wp = 0.9*pi; % wp = wp1
% no need wp2
%
%
NH = (N-1)/2;
%
%
P = zeros(NH,1);
Q = zeros(NH,NH);
for i=1:NH % 0:1:NH
    P(i) = 2*wp*cos(i*wp)/i - 2*sin(i*wp)/i^2;
    for j=1:NH
        if i==j
            Q(i,j) = 0.5*wp - 0.5*sin((i+j)*wp)/(i+j);
        else
            Q(i,j) = 0.5*sin((i-j)*wp)/(i-j) -
0.5*sin((i+j)*wp)/(i+j);
       end
    end
end
A = -0.5*inv(Q)*P;
%
%
h = zeros(N,1);
h(1:NH) = 0.5*A(NH:-1:1);
h(NH+2:N) = -0.5*A;
%
% subplot - need to show several plots simultaneously
% 從圖可知是 high-pass filter
subplot(1,2,1);
stem(0:N-1,h);
xlabel('n');
ylabel('Impulse Response');
%
% Amplitude Response
subplot(1,2,2);
AR = abs(freqz(h,1,0:pi/200:pi));
plot(0:1/200:1,AR);
axis([0,1,0,3]);
xlabel('Normalized Frequency (\omega/\pi)');
ylabel('Amplitude Response');
```

```
%
% Case 3 don't need dB plots
%

% Case 3 Differentiator
```



```
%
% Design of case 3 Hilbert Transformers
%
clear all;   % clear workspace
clc;         % clear command window
N = 31;
wp1 = 0.05*pi;
wp2 = 0.95*pi;
%
%
NH = (N-1)/2;
%
%
P = zeros(NH,1);
Q = zeros(NH,NH);
for i=1:NH % 0:1:NH
    P(i) = -2*(cos(i*wp2) - cos(i*wp1))/i;
    for j=1:NH
        if i==j
            Q(i,j) = 0.5*(wp2-wp1) - 0.5*sin((i+j)*wp2)/(i+j) +
0.5*sin((i+j)*wp1)/(i+j);
        else
            Q(i,j) = 0.5*sin((i-j)*wp2)/(i-j) - 0.5*sin((i-
j)*wp1)/(i-j) ....
                  - 0.5*sin((i+j)*wp2)/(i+j) +
0.5*sin((i+j)*wp1)/(i+j);
       end
    end
end
A = -0.5*inv(Q)*P;
%
%
h = zeros(N,1);
h(1:NH) = 0.5*A(NH:-1:1);
h(NH+2:N) = -0.5*A;
%
% subplot - need to show several plots simultaneously
subplot(1,2,1);
stem(0:N-1,h);
xlabel('n');
ylabel('Impulse Response');
%
% Amplitude Response
subplot(1,2,2);
AR = abs(freqz(h,1,0:pi/200:pi));
plot(0:1/200:1,AR);
axis([0,1,0,1.1]);
```

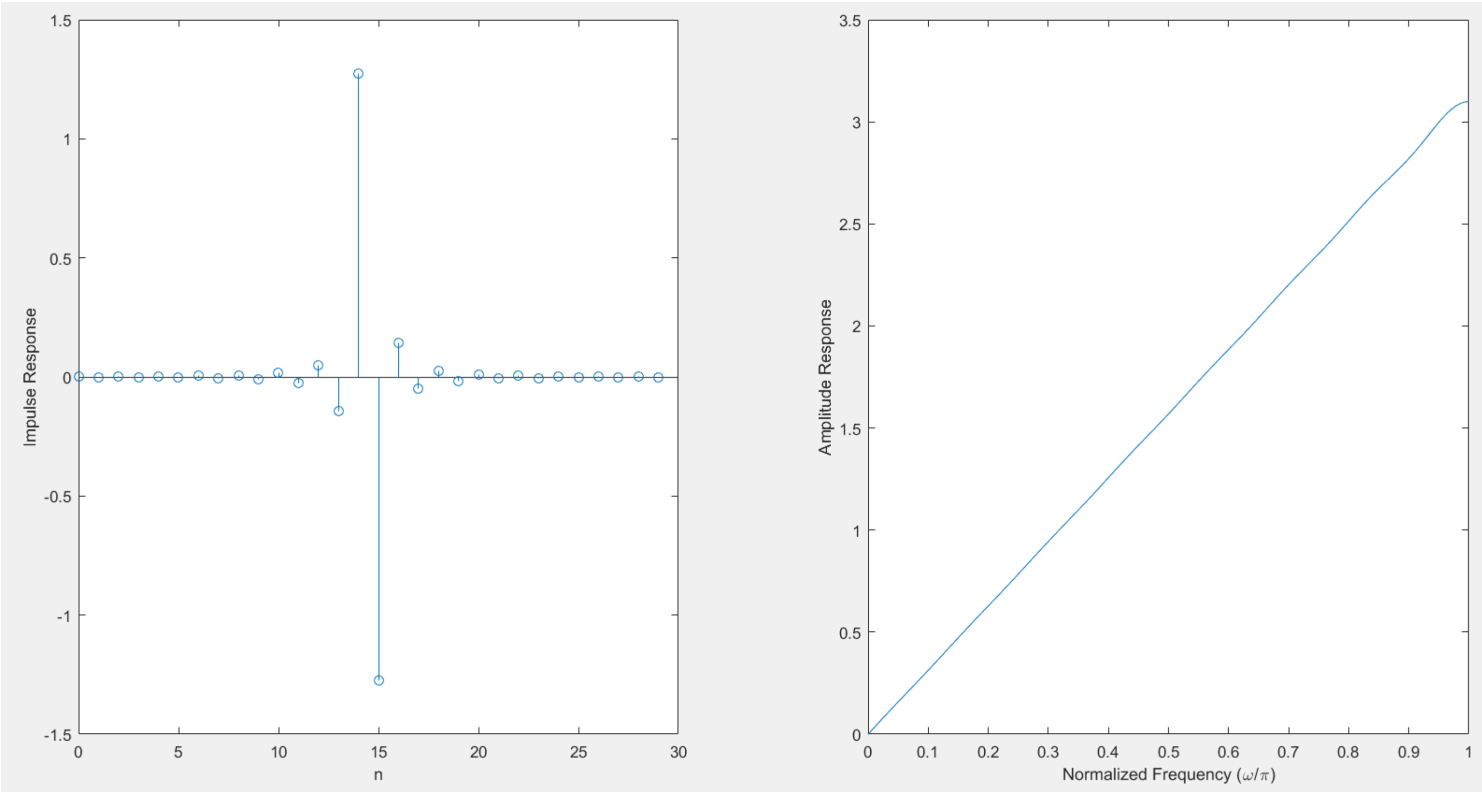
```
xlabel('Normalized Frequency (\omega/\pi)');
ylabel('Amplitude Response');
%
% case 3 don't need dB plots
%


% Case 3 Hilbert Transformers
```

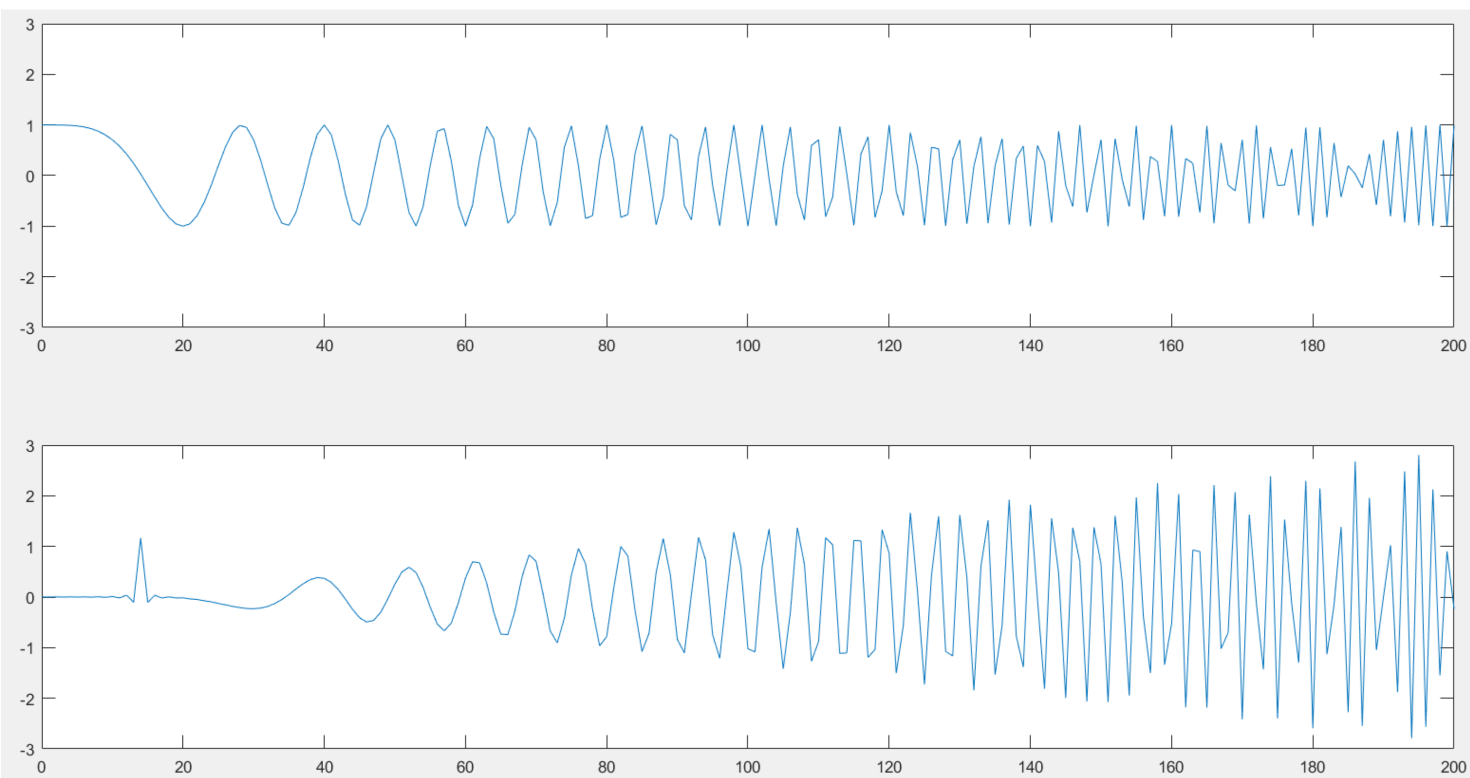


```
%
% Design of Case 4 Differentiator
%
clear all;   % clear workspace
clc;         % clear command window
N = 30;      % filter length
% wp1 = 0;  因為下限從0開始，所以不再需要wp1
% wp2 = pi; 因為上限可以到pi，所以不再需要wp2
%
NH = N/2;
%
%
P = zeros(NH,1); % declaration 宣告
for i=1:NH % 0:1:NH
   P(i) = -2*sin((i-0.5)*pi)/(i-0.5)^2;
end
Q = 0.5*pi*eye(NH);
A = -0.5*inv(Q)*P;
%
%
h = zeros(N,1); % 怎麼求得h ?
h(1:NH) = 0.5*A(NH:-1:1);
h(NH+1:N) = -0.5*A;
%
% subplot - need to show several plots simulteneously
subplot(1,2,1);
stem(0:N-1,h);
xlabel('n');
ylabel('Impulse Response');
%
% Amplitude Response
subplot(1,2,2);
AR = abs(freqz(h,1,0:pi/200:pi));
plot(0:1/200:1,AR);
axis([0,1,0,3.5]);
xlabel('Normalized Frequency (\omega/\pi)');
ylabel('Amplitude Response');
%
% signal simulation
%
NT = (0:200)';
x = zeros(1,201);
for in=0:200
  x(in+1) = cos(in*(in/400)*pi);
end
subplot(2,1,1);
plot(NT,x);
```

```
axis([0,200,-3,3]);
%
%
y = filter(h,1,x);
subplot(2,1,2);
plot(NT,y); % plot shows input delay
axis([0,200,-3,3]);
%
%
```

```
% Case 4 Differentiator
```



```
%
% Design of case 4 Hilbert Transformers
%
clear all;   % clear workspace
clc;         % clear command window
N = 30;
wp1 = 0.05*pi;
% wp2 = pi; 因為上限可到pi，所以不再需要wp2
%
%
NH = N/2;
%
%
P = zeros(NH,1);
Q = zeros(NH,NH);
for i=1:NH % 0:1:NH
    P(i) = 2*cos((i-0.5)*wp1/(i-0.5));
    for j=1:NH
        if i==j
            Q(i,j) = 0.5*(pi-wp1) + 0.5*sin((i+j-1)*wp1)/(i+j-1);
        else
            Q(i,j) = -0.5*sin((i-j)*wp1)/(i-j) + 0.5*sin((i+j-
1)*wp1)/(i+j-1);
       end
    end
end
A = -0.5*inv(Q)*P;
%
%
h = zeros(N,1);
h(1:NH) = 0.5*A(NH:-1:1);
h(NH+1:N) = -0.5*A;
%
% subplot - need to show several plots simulteneously
subplot(1,2,1);
stem(0:N-1,h);
xlabel('n');
ylabel('Impulse Response');
%
% Amplitude Response
subplot(1,2,2);
AR = abs(freqz(h,1,0:pi/200:pi));
plot(0:1/200:1,AR);
axis([0,1,0,1.1]);
xlabel('Normalized Frequency (\omega/\pi)');
ylabel('Amplitude Response');
%
```

```
%
% case 4 Hilbert Transformers
```