ELEC2100 **Prelab #3**

Objective

- To be familiar with Fourier series for signals and frequency response for systems
- To be familiar with practical filters (lowpass and bandpass)

Explanation on "fft"

Given : $x(t) = 2\cos\left(200\pi t - \frac{\pi}{4}\right) = e^{-j\frac{\pi}{4}} e^{j200\pi t} + e^{j\frac{\pi}{4}} e^{-j200\pi t}$ → FS coefficient : a_k

Fundamental frequency = 200π rad/s = 100 Hz FS coefficients of $x(t)$: $a_1 = e^{-j\frac{\pi}{4}}$ and $a_{-1} = e^{j\frac{\pi}{4}}$

When k = 1 and k = –1, the actual frequency is 100 Hz and –100 Hz, respectively.

Remember that Matlab is only able to process discrete values. Matrix x shown below is a sampled version of $x(t)$ and the entire matrix x is regarded as one period of $x(t)$ in Matlab.

```
fs=1e3;                          % sampling frequency = 1e3 = 10^3 = 1000 Hz
t=0:1/fs:1-1/fs;                 % time index
x=2*cos(200*pi*t-pi/4);          % sampled x
ak=fft(x)/length(x);             % obtain the FS coefficients using fft

% Compare the following expression (help fft) with the analysis equation (DTFS)
```

```
 For length N input vector x, the DFT is a length N vector X,
 with elements
                      N
    X(k) =           sum  x(n)*exp(-j*2*pi*(k-1)*(n-1)/N), 1 <= k <= N.
                     n=1
```

(fft)

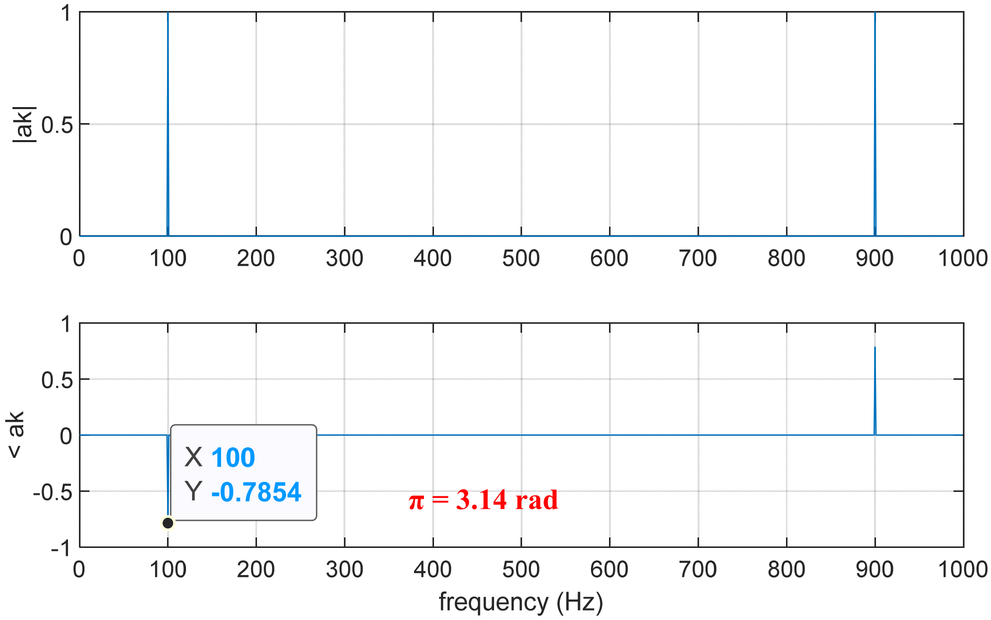
$$a_k = \frac{1}{N}\sum_{n=<N>} x[n]e^{-jk\omega_o n} \qquad \text{(DTFS)}$$

% The difference is $\frac{1}{N}$ where N is the total number of points contained in matrix x.
i.e. length(x)

% To obtain the FS coefficients fft(x) is divided by length(x).
i.e. fft(x)/length(x)

```
akp=angle(ak).*(abs(ak)>0.001);                   % phase of ak

% You may ignore the phase if the magnitude is too small, say less than 0.001.

f=[0:length(x)-1]*fs/length(x);                   % frequency index (0 to fs)
figure(1);
subplot(211); plot(f,abs(ak)); ylabel('|ak|'); grid;   % plot |ak|   (magnitude of FS)
subplot(212); plot(f,akp); ylabel('< ak'); grid;       % plot < ak  (phase of FS)
xlabel('frequency (Hz)');
```

ak is a complex sequence returned by Matlab. The frequency range is from 0 to the sample rate (fs) by default.

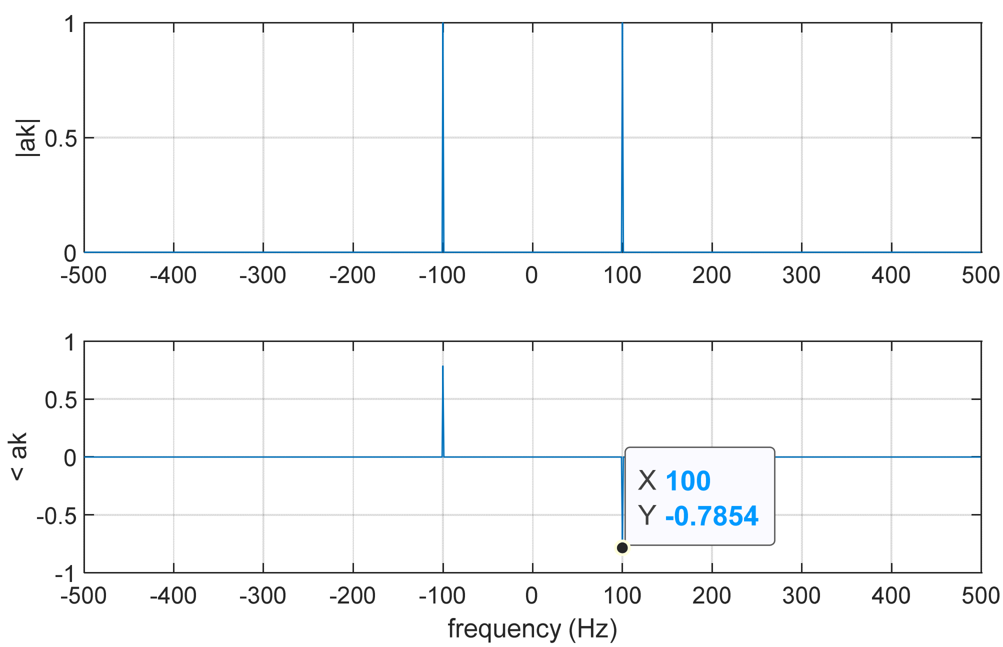


Self-check :

- Directly plot the phase of ak (i.e. plot(f, angle(ak)) without considering the magnitude.
- Explain the difference.

Use "fftshift" to rearrange the data so as to show the negative frequency axis.

```
f1=f-fs/2;                                                     % shift the frequency index (– fs/2 to fs/2)
figure(2);
subplot(211); plot(f1,fftshift(abs(ak))); ylabel('|ak|'); grid;   % plot |ak|  (magnitude of FS)
subplot(212); plot(f1,fftshift(akp));; ylabel('< ak'); grid;      % plot < ak  (phase of FS)
xlabel('frequency (Hz)');
```



Example for “fftshift”

e.g.

```
>> z=[1   4   7   9   10   15]

z =
    1     4     7     9    10    15

>> fftshift(z)

ans =
     9    10    15     1     4     7
```

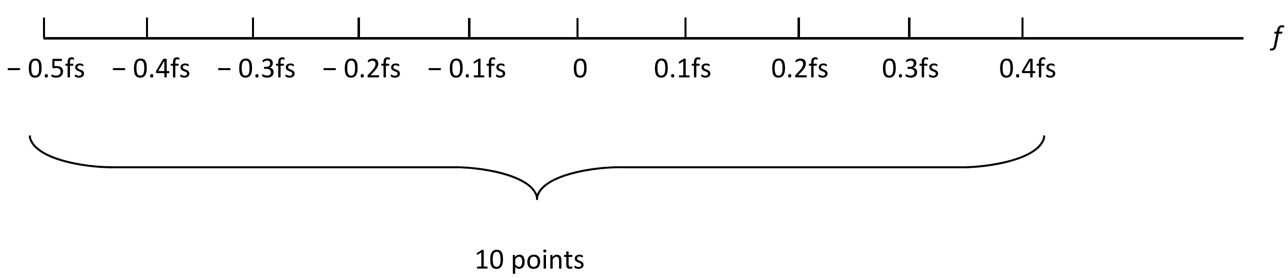
e.g.

```
>> z=[1   4   7   9   10   15   19]

z =
    1     4     7     9    10    15    19

>> fftshift(z)

ans =
    10    15    19     1     4     7     9
```

Frequency index

e.g. Number of points N = 10   step = fs/N   (i.e. frequency interval between two points)

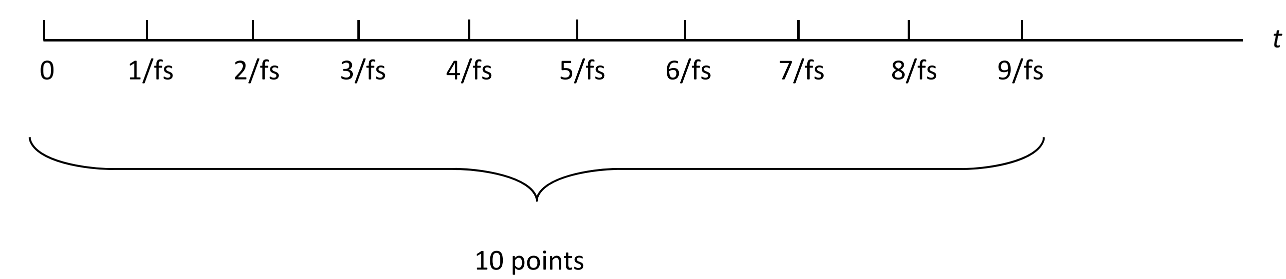
f = [− fs/2 : step : fs/2 – step]   or   [− N/2 : N/2 − 1] * fs/N



Time index

e.g. Number of points N = 10   step = 1/fs   (i.e. time interval between two points)

t = [0 : step : (N-1)/fs]   or   [0 : N − 1]/fs



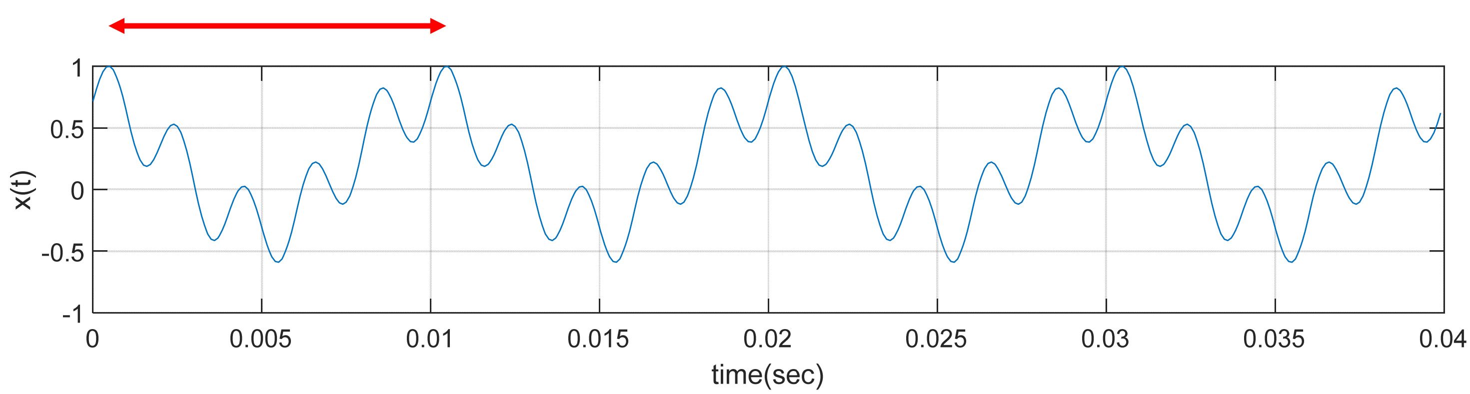
Ex.1 Fourier analysis for signals

Use “audioread” to read the sample file (**sample3.wav**).
Define a time index. Plot x versus time.
Use “fft” to obtain the FS coefficients for the periodic signal x.
Define a frequency index (in Hz).
Plot the magnitude of FS versus frequency (in Hz). Plot the phase of FS versus frequency (in Hz).

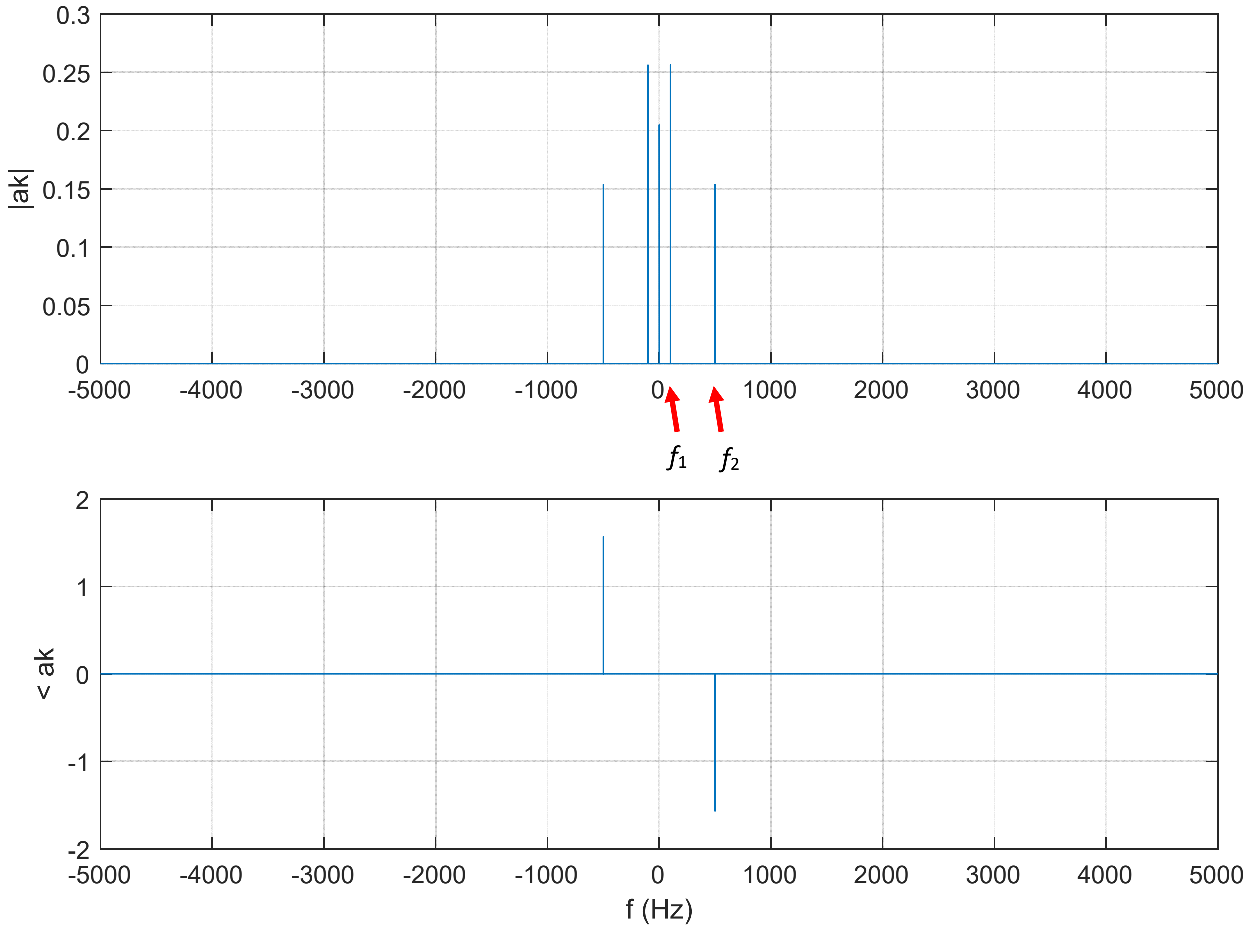
```
[x,fs]=audioread('sample3.wav');                    % sampled periodic signal
t=[0:length(x)-1]/fs;                               % time index

figure(3);
subplot(311); plot(t(1:400), x(1:400)); grid;       % Only show the first 400 points
ylabel('x(t)');

ak=fft(x)/length(x);                                % obtain ak for the sampled signal
f=[-length(x)/2:length(x)/2-1]*fs/length(x);        % frequency index
a=(abs(ak)>0.001).*angle(ak);                       % phase of ak

figure(4);
subplot(211); plot(f,abs(fftshift(ak))); ylabel('|ak|'); grid;   % plot magnitude of FS vs frequency index
subplot(212); plot(f,fftshift(a)); ylabel('< ak'); grid;         % plot phase of FS vs frequency index
xlabel('f (Hz)')
```

Self-check :
- Use Data Tips to check the frequency (in Hz) of each frequency component.
- What is the value of the DC term ?
- What is the fundamental period (in seconds) of x ?
- What is the fundamental frequency (in Hz) of x ?

Fundamental period : Observe the time duration from one peak to another peak in the time domain.



Fundamental frequency : Find the HCF (highest common factor) of f_1 and f_2 in the frequency domain.



Vertical line at $f = 0$ is the dc value (or average value).
Use Data Tips to read the frequency of each component.
Fundamental frequency (in Hz) = HCF (f_1 , f_2)
Remember that $\omega = k\omega_o$ or $f = kf_o$ where k must be integer.

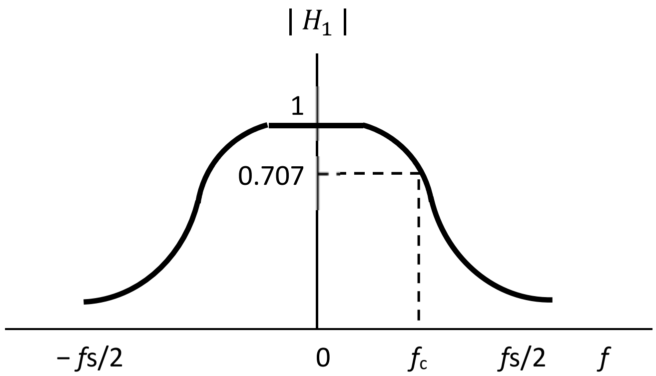
e.g. $x(t) = \cos(12\pi t) + \cos(20\pi t) = \cos(3 \times 4\pi t) + \cos(5 \times 4\pi t)$

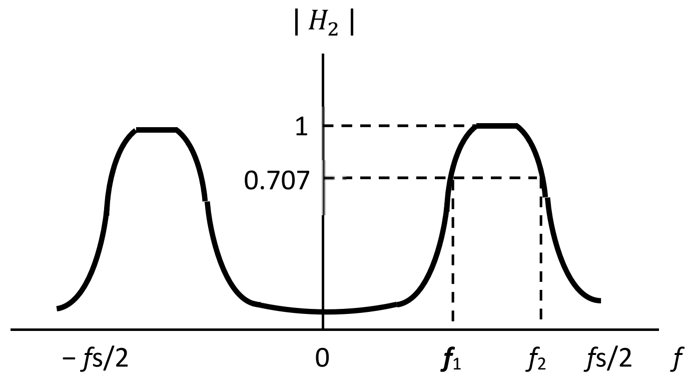
Fundamental frequency = 4π rad/s = 2 Hz

Self check :
- Use Data Tips to observe the magnitude and phase for each FS coefficient.
- Write down the mathematical expression of $x(t)$ as the sum of real sinusoids.
- Use your mathematical expression to plot $x(t)$ and compare with x shown in figure(3) to verify your answer.

Ex.2 Filters

Consider the following two frequency responses (i.e. lowpass filter and bandpass filter)





Passband : f_c

Cutoff frequency : f_c

Passband : $f_2 - f_1$
Center frequency : $(f_2 + f_1) / 2$
Cutoff frequencies : f_1 , f_2

```
[B1,A1] = butter(6, 0.04);              % filter 1 (lowpass)   N = 6   Wn = 0.04
[H1,fh] = freqz(B1,A1,1e3,fs);          % Use 1000 points (1e3) to represent H1

[B2,A2] = butter(6, [0.05  0.2]);       % filter 2 (bandpass)  N = 6   Wn = [0.05  0.2]
[H2,fh] = freqz(B2,A2,1e3,fs);          % Use 1000 points (1e3) to represent H2

figure(5)
subplot(211); plot(fh,abs(H1)); axis([0 fs/2 0 1.2]); grid; ylabel('|H1|');     % magnitude
subplot(212); plot(fh,angle(H1)); axis([0 fs/2 -pi pi]); grid; ylabel('< H1');  % phase
xlabel('f (Hz)');

figure(6)
subplot(211); plot(fh,abs(H2)); axis([0 fs/2 0 1.2]); grid; ylabel('|H2|');     % magnitude
subplot(212); plot(fh,angle(H2)); axis([0 fs/2 -pi pi]); grid; ylabel('< H2');  % phase
xlabel('f (Hz)');

% Vector H1 and H2 are the frequency responses returned by Matlab.
% The frequency range is from 0 to half of the sample rate (fs/2).
% The cutoff frequency Wn must be 0.0 < Wn < 1.0, with 1.0 corresponding to half the sample rate.
```

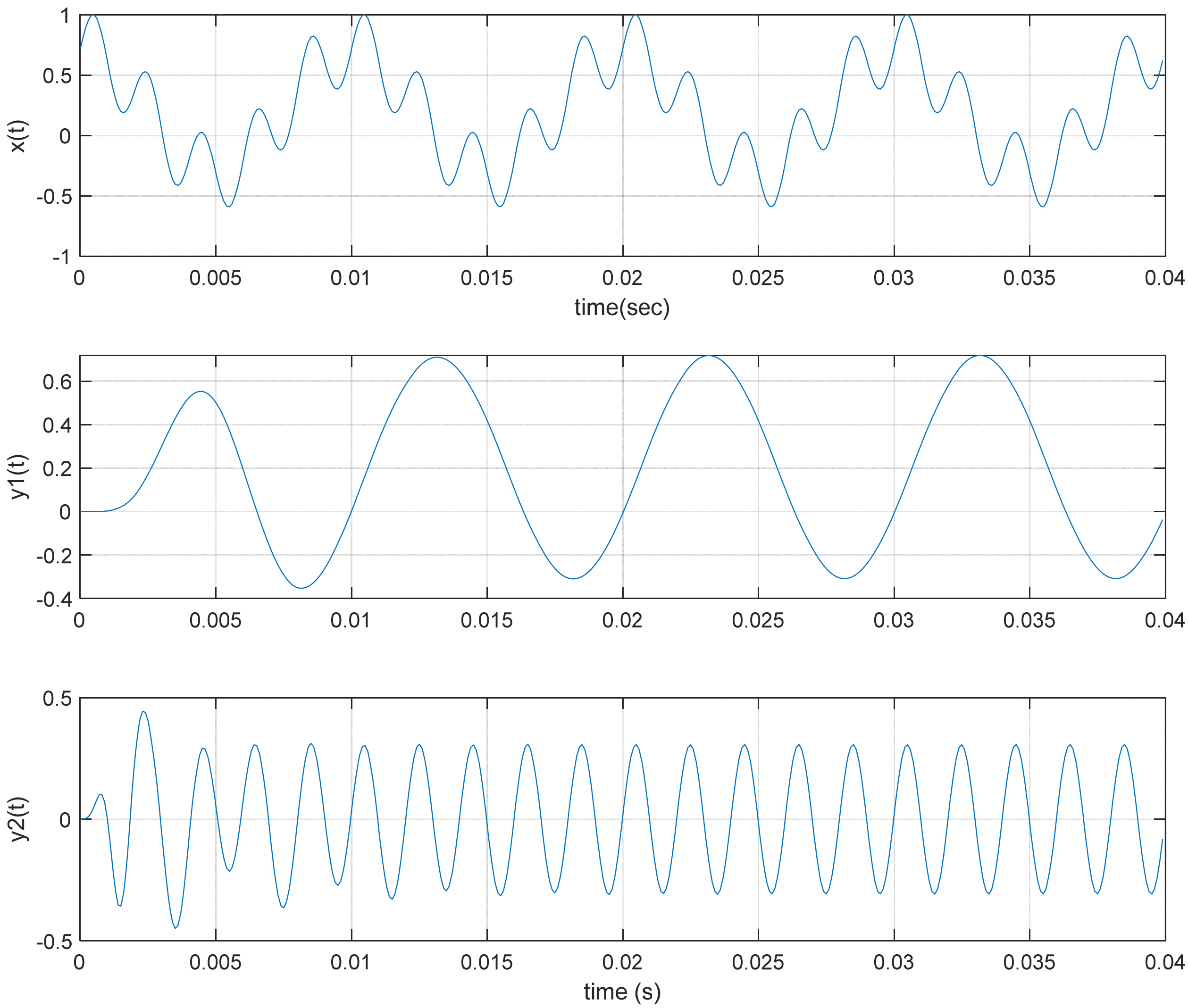
$$w_n = \frac{\text{actual cutoff frequency (in Hz)}}{\frac{fs}{2}} = \text{normalized cutoff frequency}$$

Self-check :
- What is the actual frequency (in Hz) if Wn is equal to 0.04 as shown in filter 1?
- What are the actual frequencies (in Hz) if Wn is equal to [0.05   0.2] as shown in filter 2 ?
- What is the difference on the magnitude response between ideal filter and practical filter ?

Ex.3 Filtering

Apply x to filter 1 and filter 2.
Plot y1 and y2 versus time (in sec).

```
y1=filter(B1,A1,x);                          % x is applied to filter 1 and y1 is the output
y2=filter(B2,A2,x);                          % x is applied to filter 2 and y2 is the output

figure(3)
subplot(312); plot(t(1:400),y1(1:400)); grid;          % plot y1
ylabel('y1(t)');
subplot(313); plot(t(1:400),y2(1:400)); grid;          % plot y2
ylabel('y2(t)'); xlabel('time (s)');
```



Self-check :
- Compare the input (x) and the outputs (y1 and y2).
- Use “sound” to hear the input (x) and the outputs (y1 and y2).
- What does a lowpass do ?
- What does a bandpass do ?