Stanford University
EE 102A: Signals and Systems I
Winter 2023-24, Professor Joseph M. Kahn

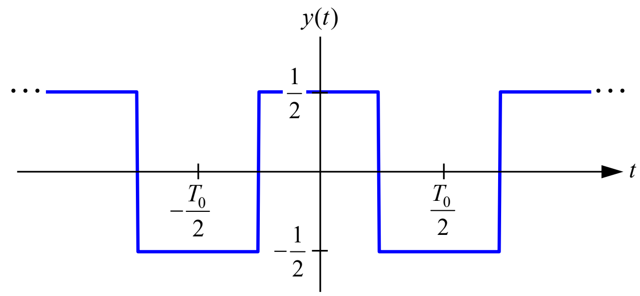
Homework 4, due Friday, February 9

All references are to the *EE 102A Course Reader.*

Using Properties to Compute CT FS Coefficients

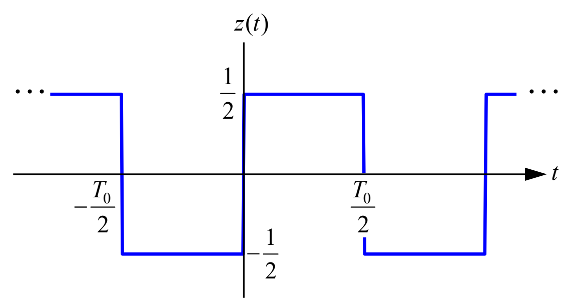
1. *CT even and odd square waves.*

   a. An even square wave $y(t) \leftrightarrow b_k$ is shown. It has a 50% duty cycle.



   Use the CT FS coefficients for a rectangular pulse train, with proper choice of pulse width $2T_1$, along with the linearity property of the CT FS, to find the CT FS coefficients b_k of $y(t)$. Sketch the real or imaginary parts, whichever are nonzero. Comment on how any symmetry in $y(t)$ is reflected in the symmetry of b_k.

   b. An odd square wave $z(t) \leftrightarrow c_k$ is shown. It has a 50% duty cycle.



   Using the CT FS coefficients found in part (a) and the time shift property of the CT FS, find the CT FS coefficients c_k of $z(t)$. Sketch the real or imaginary parts of the c_k, whichever is nonzero. Comment on how any symmetry in $z(t)$ is reflected in symmetry of c_k.

   c. The even and odd square waves are obviously orthogonal over one period:

$$\int_{T_0} y(t) z^*(t) dt = 0.$$

   Parseval's Identity (see pages 93-94) suggests that the CT FS coefficients must be orthogonal in frequency:

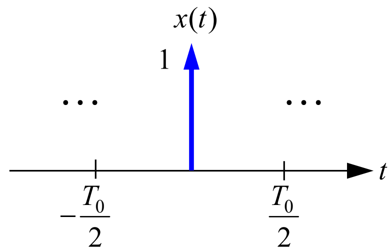
$$T_0 \sum_{k=-\infty}^{\infty} b_k c_k^* = 0.$$

Using the symmetries identified in parts (a) and (b), show this is true.

2. *CT impulse train*. The periodic CT impulse train plays an important role in the analysis of sampling and reconstruction. An impulse train of period T_0 can be represented as

$$x(t) = \sum_{l=-\infty}^{\infty} \delta\left(t - lT_0\right)$$

and is shown here in one period



a. Calculate its FS coefficients a_k.

b. The FS synthesis of $x(t)$ is

$$\hat{x}(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\omega_0 t}.$$

Since $x(t)$ is real, $a_{-k} = a_k^*$, so $a_k e^{jk\omega_0 t} + a_{-k} e^{-jk\omega_0 t}$ add up to give a purely real contribution for each $k \neq 0$. Sketch the first few terms in the synthesis, which are

$$\begin{gathered} a_0 \\ a_1 e^{j\omega_0 t} + a_{-1} e^{-j\omega_0 t} \\ a_2 e^{j2\omega_0 t} + a_{-2} e^{-j2\omega_0 t} \\ a_3 e^{j3\omega_0 t} + a_{-3} e^{-j3\omega_0 t} \end{gathered}$$

for $-T_0/2 \leq t \leq T_0/2$. These are a constant and three cosines. At most values of *t*, the cosines have different amplitudes, some positive and some negative, and tend to cancel out. At what value of *t* do they all add up constructively?

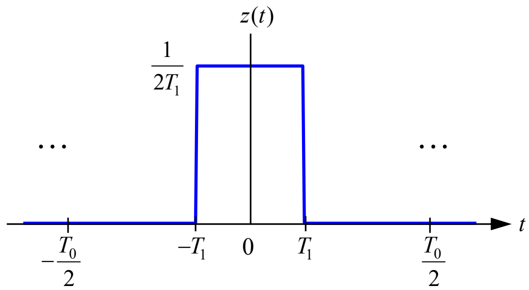
c. In lecture we calculated the FS coefficients b_k of a periodic rectangular pulse train $y(t)$:

$$y(t) = \sum_{l=-\infty}^{\infty} \Pi\left(\frac{t - lT_0}{2T_1}\right) \xleftrightarrow{FS} b_k = \frac{\omega_0 T_1}{\pi} \operatorname{sinc}\left(\frac{k\omega_0 T_1}{\pi}\right).$$

Here we scale it by $1/2T_1$ to obtain $z(t)$, which has FS coefficients c_k:

$$z(t) = \frac{1}{2T_1}\sum_{l=-\infty}^{\infty}\Pi\left(\frac{t-lT_0}{2T_1}\right) \overset{FS}{\longleftrightarrow} c_k.$$

$z(t)$ is sketched in one period



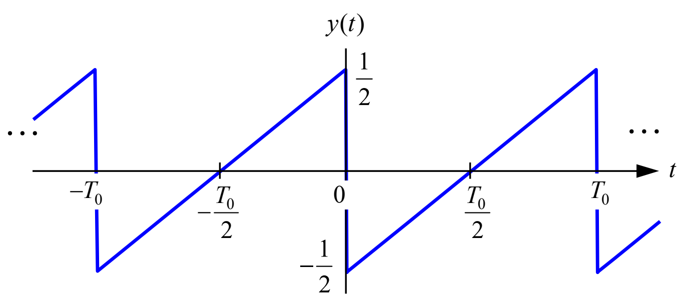
Since an impulse can be defined as the limiting case of a narrow, high rectangular pulse of unit area, we note that

$$\lim_{T_1 \to 0} z(t) = x(t).$$

Argue that

$$\lim_{T_1 \to 0} c_k = a_k.$$

3. *CT sawtooth wave*. Consider a periodic sawtooth signal $y(t) \overset{FS}{\longleftrightarrow} b_k$



It can be expressed as

$$y(t) = \frac{t}{T_0} - \frac{1}{2} \qquad 0 \le t \le T_0$$

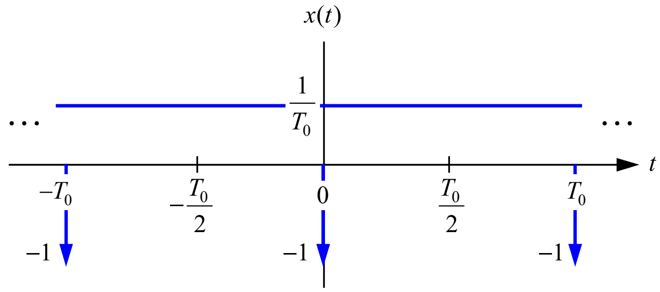
$$y(t+T_0) = y(t) \qquad \forall t$$

and its FS coefficients can be calculated by evaluating the analysis equation:

$$b_k = \frac{1}{T_0}\int_0^{T_0}\left(\frac{t}{t_0} - \frac{1}{2}\right)e^{-jk\omega_0 t}\,dt.$$

You need not evaluate this integral.

a. Here we consider an alternate approach. Consider a related periodic signal $x(t) \overset{FS}{\longleftrightarrow} a_k$:



$x(t)$ can be expressed as the sum of a constant and a negative impulse train:

$$x(t) = \frac{1}{T_0} - \sum_{l=-\infty}^{\infty} \delta\left(t - lT_0\right).$$

Using linearity and the known FS coefficients of a constant and an impulse train, calculate the FS coefficients of $x(t)$ given by a_k.

b. Observe that the two waveforms are related by

$$y(t) = \int_{-\infty}^{t} x(t')dt'.$$

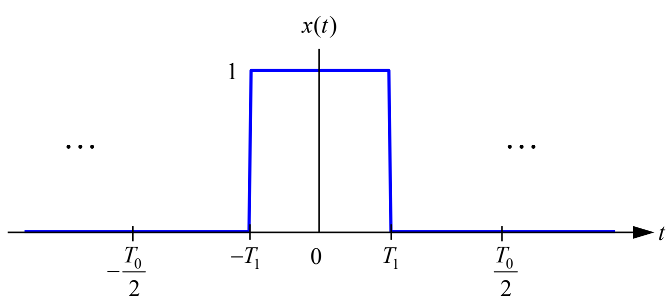
Refer to the running integration property in the Appendix, Table 1. Is the condition on the FS coefficients of $x(t)$, $a_0 = 0$, satisfied? If so, use the property to compute the FS coefficients of $y(t)$, given by b_k. Sketch the real or imaginary part of b_k, whichever is nonzero. Identify any symmetry in $y(t)$ and the corresponding symmetry in b_k.

4. *CT triangular pulse train.* Given two periodic signals $x(t)$ and $y(t)$ with a common period T_0, the periodic convolution is defined as

$$z(t) = \int_{T_0} x(t')y(t-t')dt'. \tag{1}$$

This looks like a regular convolution, but the integration over t' is performed over any interval of length T_0, instead of over $-\infty < t' < \infty$. Note that the periodic convolution $z(t)$ is periodic in t with period T_0, since the only place t appears on the right-hand side of (1) is in $y(t-t')$, which is periodic in t.

a. Suppose $x(t) = y(t)$ is the periodic rectangular pulse train shown



Assume $T_1 \le T_0 / 4$. Sketch the periodic convolution $z(t)$.

b. $z(t)$ has CT FS coefficients c_k, which we want to compute. Note that

$$z(t)=\begin{cases}2T_1\left(1-\dfrac{|t|}{2T_1}\right) & |t|<2T_1\\[2ex] 0 & 2T_1\le|t|\le\dfrac{T_0}{2}\end{cases}$$

$$z(t+T_0)=z(t)\qquad\forall t$$

We could compute the c_k using the analysis equation:

$$\begin{aligned}c_k&=\frac{1}{T_0}\int_{T_0}z(t)e^{-jk\omega_0 t}dt\\&=\frac{2T_1}{T_0}\int_{-2T_1}^{2T_1}\left(1-\frac{|t|}{2T_1}\right)e^{-jk\omega_0 t}dt.\end{aligned}$$

You do not need to evaluate this integral. Instead, use the periodic convolution property of the CT FS to determine c_k.

CT or DT LTI System Analysis

5. *Eigenfunctions of CT or DT LTI Systems.* As discussed in lecture, the signals e^{st}, s complex, $-\infty<t<\infty$, are eigenfunctions of all CT LTI systems for which the integral defining $H(s)$ exists. Likewise, the signals z^n, z complex, $-\infty<n<\infty$, are eigenfunctions of all DT LTI systems for which the sum defining $H(z)$ exists.

   We can also show that the e^{st} or z^n are the *only* signals that can be eigenfunctions of all CT or DT LTI systems, respectively. Prove this for the CT case. *Hint*: Identify a simple LTI system for which the e^{st} are the only eigenfunctions. It is a system we studied in lecture on several occasions.

6. *Time shift and differentiator*. Consider these LTI systems with input $x(t)$ and output $y(t)$. For each system, by setting $x(t)=e^{j\omega t}$ and $y(t)=H(j\omega)e^{j\omega t}$, find an expression for the frequency response $H(j\omega)$, and sketch the magnitude $|H(j\omega)|$ and phase $\angle H(j\omega)$.

   a. *Time shift*

   $$y(t)=x(t-t_0).$$

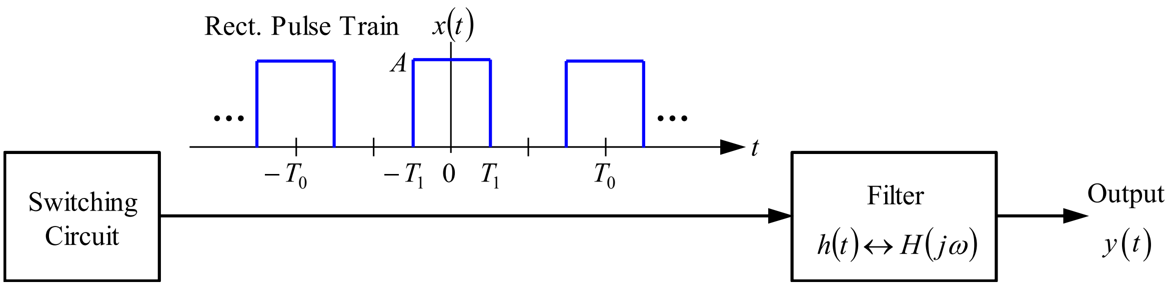
   b. *Differentiator*

   $$y(t)=\frac{dx}{dt}.$$

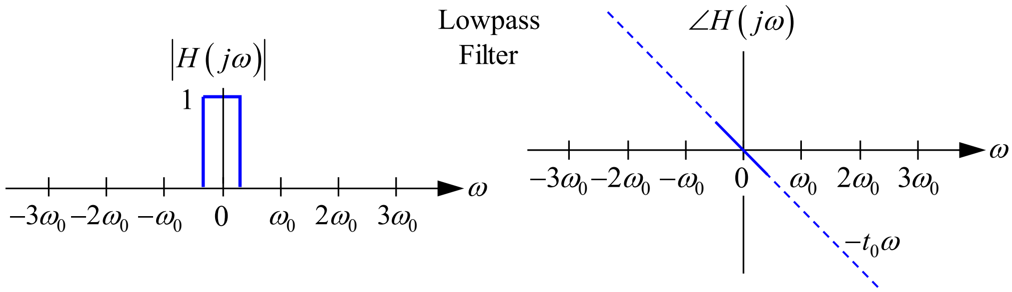
Applications of CT FS and LTI System Analysis

7. *Switching d.c. power supply or oscillator*. Switching circuits are an efficient way to generate a time-varying voltage from a fixed d.c. supply voltage. They are especially useful in high-power applications. Here, a switching circuit generates a rectangular pulse train $x(t)$ with fixed amplitude A, fixed fundamental frequency $\omega_0=2\pi/T_0$, and variable pulse width $2T_1$. It is convenient to define a duty

cycle $\eta = 2T_1 / T_0$. Depending on the design of the filter, the output $y(t)$ can be a d.c. voltage or sinusoid with variable amplitude.

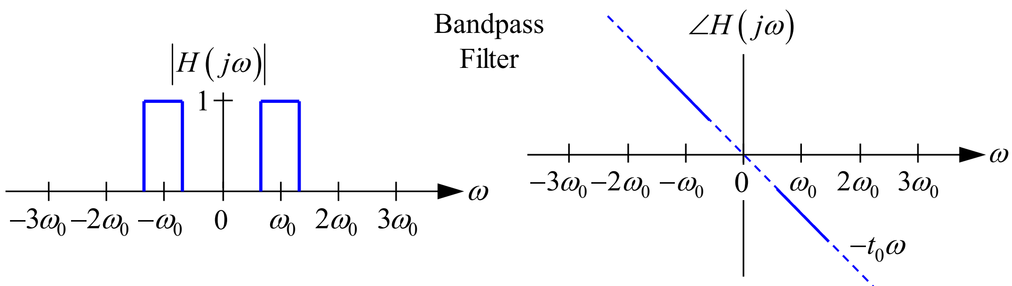


a. *Variable d.c. supply*. The filter has a lowpass response with the magnitude and phase shown.



Find an expression for $y(t)$ as a function of A, the duty cycle $\eta = 2T_1 / T_0$, and the filter group delay t_0. What value of duty cycle η maximizes $y(t)$, and what is the maximum value? *Hint*: the lowpass filter, described by the frequency response $H(j\omega)$, selects the CT FS component of $x(t)$ at just one value of *k*, corresponding to just one frequency $k\omega_0$. As a result, the CT FS expansion of $y(t)$ comprises just one term.

b. *Variable a.c. supply*. The filter has a bandpass response with the magnitude and phase shown.



Find an expression for $y(t)$ as a function of A, the duty cycle $\eta = 2T_1 / T_0$, and the filter group delay t_0. What value of η duty cycle maximizes the peak-to-peak value of $y(t)$, and what is the maximum peak-to-peak value? How does that maximum peak-to-peak value compare to A? *Hint*: the bandpass filter selects the CT FS components of $x(t)$ at two values of *k*, corresponding to two frequencies $k\omega_0$. As a result, the CT FS expansion of $y(t)$ comprises two terms. The bandpass filter imparts phase shifts to these two terms that correspond to a time shift in $y(t)$.

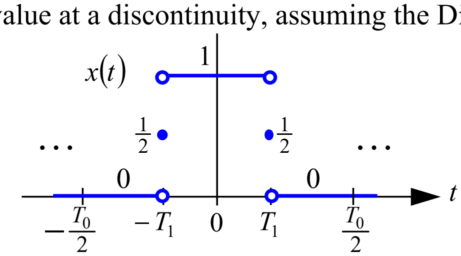
Laboratory 4

In lecture, we have been discussing the use of Fourier series to represent periodic continuous-time signals. Here, we will see Fourier series in action. In order to maximize the learning-to-work ratio, we provide all the scripts and some of the functions required, and ask you to implement four key functions called by the scripts. However, we strongly recommend that you go through the provided MATLAB codes line-by-line and make sure you understand them fully. They are posted in **`Homework_4_MATLAB_Files.zip`** in the Homework Assignments folder on the Canvas website.

Task 1: Reconstruction Error in Fourier Series

In this task, you will write a function to compute the Fourier series coefficients a_k for a rectangular pulse train. Then you will reconstruct the rectangular pulse train using a Fourier series and examine differences between the ideal rectangular pulse train and the Fourier series reconstruction.

We provide a script and a function you may use. The function, **`x_ideal_rectpulsetrain.m`**, creates an ideal pulse train to which you will be comparing your Fourier series reconstruction. This ideal pulse train has values of 1 inside each pulse, 0 outside each pulse, and ½ exactly at the boundaries, since a Fourier series converges to the average value at a discontinuity, assuming the Dirichlet conditions are satisfied.



You are asked to write a function **`a_rectpulsetrain.m`**, which is called from the main script and returns the Fourier series coefficients a_k for the rectangular pulse train of period $T_0 = 2\pi/\omega_0$ and pulse width $2T_1$. This function will accept three parameters: **`k`**, the coefficient index; **`omega0`**, the fundamental frequency; and **`T1`**, the pulse half-width. The script calls it as **`a_rectpulsetrain(k,omega0,T1)`**.

The FS reconstruction including imaginary exponentials up to $\pm K\omega_0$ is

$$\hat{x}_K(t) = \sum_{k=-K}^{K} a_k e^{jk\omega_0 t} .$$

In Task 1, we will study differences between the ideal rectangular pulse train $x(t)$ and its reconstruction $\hat{x}_K(t)$. We quantify the differences by an integrated absolute-square error between $x(t)$ and $\hat{x}(t)$, given by

$$\varepsilon_K = \int_{T_0} \left| x(t) - \hat{x}_K(t) \right|^2 dt , \tag{1}$$

where the integral runs over any period, such as $-T_0/2 \le t < T_0/2$. Ideally, $x(t)$ and $\hat{x}_K(t)$ are continuous-time signals. As in Laboratory 3, to represent them in MATLAB, we must discretize time with interval Δt, represented by the MATLAB variable **`deltat`**, and describe the signals by their discrete values $x(n\Delta t)$ and $\hat{x}_K(n\Delta t)$, represented by the MATLAB vectors **`x`** and **`xhatK`**. In Laboratory 3, we approximated a convolution integral by a Riemann sum. In Task 1, we will approximate the error integral (1) by a sum

$$\varepsilon_K = \lim_{\Delta t \to 0} \sum_n \left| x(n\Delta t) - \hat{x}_K(n\Delta t) \right|^2 \Delta t , \tag{2}$$

where the values of n correspond to one period of these signals. In MATLAB, we compute the summation (2) by `epsK = sum(abs(x - xhatK).^2)*deltat`, choosing `deltat` to be suitably small.

In Task 1, in reconstructing a signal using imaginary exponentials with frequencies up to $\pm K\omega_0$, we will choose `deltat = T0/K/128`. With this choice, the time increment Δt corresponds to 1/128 of a cycle of a sinusoid at frequency $K\omega_0$. This is necessary to model the reconstruction error accurately for small K.

After writing the function `a_rectpulsetrain.m`, run the script for `K = 4, 16, 64, 256`. Turn in the function and all four plots. Comment briefly on any trends you notice. *Do not worry if the legends* $x(t)$ *and* $\hat{x}_K(t)$ *do not print properly. You will not lose any credit.*

```
%% Task 1: reconstruction error

clear all
lw = 1.5;                  % line width for plots

% Signal and FS parameters
T0 = 1;                    % period
omega0 = 2*pi/T0;          % fundamental frequency
T1 = T0/4;                 % pulse half-width, T1 < T0/2 to avoid overlap
K = 256;                   % FS reconstruction sums from -K to +K
% Try K = 4, 16, 64, 256
deltat = T0/K/128;         % discretization interval chosen to give 128
                           % samples per cycle of highest frequency
components
t1 = -T0/2; t2 = T0/2;
t = t1:deltat:t2;          % time vector for all signals

% x(t), ideal rectangular pulse train to be synthesized
x = x_ideal_rectpulsetrain(t,T0,T1);

% xhatK(t), Fourier series synthesis using terms from -K to K
xhatK = zeros(size(t));
for k = -K:K;
    ak = a_rectpulsetrain(k,omega0,T1);
    xhatK = xhatK + ak*exp(j*k*omega0*t);
end

% epsK, integrated squared error between x(t) and xhatK(t)
epsK = sum(abs(x - xhatK).^2)*deltat;

% display results
figure(1)
plot(t,real(xhatK),'b-',t,x,'r--')
l=get(gca,'children'); set(l,'linewidth',lw)
xlabel('Time \itt'); ylabel('Ideal Signal and FS Synthesis');
leg = legend({'$x(t)$';'$\hat{x}_{K}(t)$'});
set(leg,'Interpreter','latex');
title(['2\itK\rm + 1 Terms, \itK\rm = ' num2str(K,4) ...
```

```
    ', Integrated Absolute Square Error \epsilon_{\itK} = '
num2str(epsK,2)]);

function x = x_ideal_rectpulsetrain(t,T0,T1)
% Returns ideal rectangular pulse train, with value of 1 inside each
% pulse, 0 outside the pulse, and exactly 1/2 at the boundaries.
x = double(abs(mod(t-T0/2,T0)-T0/2)<T1) + ...
    1/2*double(abs(mod(t-T0/2,T0)-T0/2)==T1);
end
```

Task 2: Filtering Rectangular Pulse Train in Frequency Domain

Tasks 2 and 3 will use the function **a_rectpulsetrain.m** you wrote for Task 1, so please complete Task 1 first.

In Homework 3 and Laboratory 3, we filtered a rectangular pulse in the *time domain*, by convolution of the signal with the impulse response of an LTI system, such as a lowpass filter. Here, in Task 2, we learn how to filter a rectangular pulse train in the *frequency domain*, by multiplication of the Fourier series coefficients by the frequency response of an LTI system. We will study three different LTI systems: a time shift, a differentiator, and a first-order lowpass filter.

Instead of filtering a single rectangular pulse, we filter a periodic pulse train input signal $x(t)$ because we can represent it in the frequency domain by its Fourier series coefficients a_k. Given any LTI system, we determine its frequency response $H(j\omega)$. The output signal $y(t)$ is periodic, so we represent it by its Fourier series coefficients b_k. To find these coefficients, we multiply each of the input signal Fourier series coefficients a_k by the frequency response at frequency $\omega = k\omega_0$, to obtain $b_k = a_k H(jk\omega_0)$. Finally, we synthesize the output signal using

$$\hat{y}_K(t) = \sum_{k=-K}^{K} b_k e^{jk\omega_0 t} = \sum_{k=-K}^{K} a_k H(jk\omega_0) e^{jk\omega_0 t} . \tag{3}$$

We provide the main script for Task 2 along with a function that returns the frequency response of a time shift filter, **Hshift.m**. You are asked to write functions **Hdiff.m** and **Hfolpf.m**, which return the frequency responses of the differentiator and the first-order low pass filter, respectively. We have derived the frequency responses of all three systems in lecture and include them below.

As stated already, you will need to use the function **a_rectpulsetrain.m** that you wrote in Task 1.

Time Shift

$$H_{\text{shift}}(j\omega) = e^{-j\omega t_0}.$$

This frequency response is implemented in the function **Hshift.m**. It accepts two input parameters, the frequency **omega** and the time shift **t0**. It is called as **Hshift(k*omega0,t0)** in the main script.

Differentiator

$$H_{\text{diff}}(j\omega) = j\omega$$

Write a function **Hdiff.m** to return this frequency response. This function accepts a single input parameter **omega**. It is called as **Hdiff(k*omega0)** in the main script.

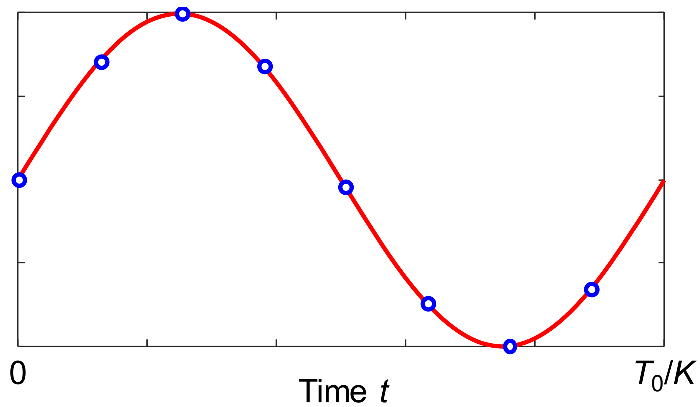
First-order Lowpass Filter

$$H_{\text{flopf}}(j\omega) = \frac{1}{1 + j\omega\tau}$$

Write the function **`Hfolpf.m`** to return this frequency response. This function will accept two parameters **`omega`** and **`tau`**. It is called as **`Hfolpf(k*omega0,tau)`** in the main script.

In Tasks 2 and 3, we are not studying reconstruction error. We will always use $K = 256$. So we will simply write the LTI system output (3) as $y(t)$.

In Tasks 2 and 3, we only need to achieve a reasonable visual rendering of the input and output signals. We will choose a time discretization interval **`deltat = T0/K/8`**. With this choice, the time increment Δt corresponds to 1/8 of a cycle of a sinusoid at frequency $K\omega_0$, the highest frequency used in our Fourier series synthesis. One cycle of this sinusoid and the eight discrete samples are shown in this figure.



After writing the functions **`Hdiff.m`** and **`Hfolpf.m`**, run the main script. Turn in the two functions and the plot obtained.

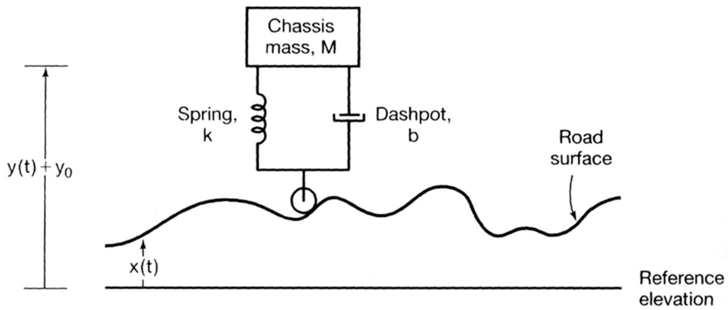
```
function H = Hshift(omega,t0)
% Time shift
H = exp(-sqrt(-1)*omega*t0);
end


%% Task 2: filtering of pulse train by time shift, differentiator or
%% first-order lowpass filter

clear all
lw = 1.5;                   % line width for plots

% Signal and FS parameters
T0 = 1;                     % period
omega0 = 2*pi/T0;           % fundamental frequency
T1 = T0/8;                  % pulse half-width, require T1 < T0/2 to avoid
overlap
K = 256;                    % FS reconstruction sums from -K to +K
deltat = T0/K/8;            % discretization interval chosen to give 8
                            % samples per cycle of highest freq. comps.
```

```
t1 = -T0/2; t2 = T0/2;
t = t1:deltat:t2;       % time vector for all signals

% Time shift parameters
t0 = 0.2*T0;

% First-order lowpass filter parameters
tau = 0.1*T0;            % time constant

% y(t), Fourier series synthesis of output using terms from -K to K
x = zeros(size(t));
yshift = zeros(size(t));
ydiff = zeros(size(t));
yfo = zeros(size(t));
for k = -K:K;
    ak = a_rectpulsetrain(k,omega0,T1);
    x = x + ak*exp(j*k*omega0*t);
    yshift = yshift + ak*Hshift(k*omega0,t0)*exp(j*k*omega0*t);
    ydiff = ydiff + ak*Hdiff(k*omega0)*exp(j*k*omega0*t);
    yfo = yfo + ak*Hfolpf(k*omega0,tau)*exp(j*k*omega0*t);
end

% display results
figure(2)
subplot(221)
plot(t,real(x)); grid
l=get(gca,'children'); set(l,'linewidth',lw)
xlabel('Time \itt'); ylabel('Input Signal \itx\rm(\itt\rm)');

subplot(222)
plot(t,real(yshift)); grid
l=get(gca,'children'); set(l,'linewidth',lw)
xlabel('Time \itt'); ylabel('Output Signal \ity\rm(\itt\rm)');
title(['Time Shift, \itt\rm_{0}/\itT\rm_{0} = ' num2str(t0/T0,3)]);

subplot(223)
plot(t,real(ydiff)); grid
l=get(gca,'children'); set(l,'linewidth',lw)
xlabel('Time \itt'); ylabel('Output Signal \ity\rm(\itt\rm)');
title('Differentiator');

subplot(224)
plot(t,real(yfo)); grid
l=get(gca,'children'); set(l,'linewidth',lw)
xlabel('Time \itt'); ylabel('Output Signal \ity\rm(\itt\rm)');
title(['First-Order Lowpass Filter, \tau/\itT\rm_{0} = '
num2str(tau/T0,3)]);
```

Task 3: Automobile Suspension

In this task, we will use Fourier series to model the response of an automobile driving at constant speed over a periodic series of speed bumps, represented by a rectangular pulse train. We will compare the responses of three different suspension types, roughly representative of sports, standard and luxury cars.

The key to this task is deriving the frequency response describing an automobile suspension (see OWN Section 6.7.1 for more details). As shown in this diagram, the input $x(t)$ represents the road elevation, while the output $y(t)$ represents the vehicle elevation.



The vehicle elevation $y(t)$ and road elevation $x(t)$ are related by a second-order constant-coefficient linear differential equation

$$m\frac{d^2y(t)}{dt}+b\frac{dy(t)}{dt}+ky(t)=kx(t)+b\frac{dx(t)}{dt}, \tag{4}$$

where m is the mass of the chassis and k and b are the spring and shock absorber constants, respectively. Knowing that the frequency response of this system exists, we can derive the frequency response using the method presented in lecture. Let the input and output be $x(t) = e^{j\omega t}$ and $y(t) = H(j\omega)e^{j\omega t}$. We substitute these into (4):

$$H(j\omega)[m(j\omega)^2e^{j\omega t}+b(j\omega)e^{j\omega t}+ke^{j\omega t}]=ke^{j\omega t}+b(j\omega)e^{j\omega t}.$$

Canceling the factors of $e^{j\omega t}$, and solving for $H(j\omega)$ yields

$$H(j\omega)=\frac{k+b(j\omega)}{M(j\omega)^2+b(j\omega)+k}. \tag{5a}$$

For convenience, we make the parameter substitutions $\omega_n=\sqrt{k/M}$ and $2\zeta\omega\_n = b/M$, obtaining a canonical form of the equation:

$$H(j\omega)=\frac{\omega_n^2+2\zeta\omega_n(j\omega)}{(j\omega)^2+2\zeta\omega_n(j\omega)+\omega_n^2}. \tag{5b}$$

The natural frequency ω_n is the resonant frequency at which the system would oscillate in the absence of damping. Choosing ω_n large (stiff suspension) makes the car more agile, while choosing it small (soft suspension) makes the car more sluggish but gives a smoother ride. The damping constant ζ is a measure of how strongly the oscillations are damped by the shock absorber. If $\zeta < 1$ (underdamped), the car is agile, but it will oscillate up and down after going over a bump. If $\zeta > 1$ (overdamped), the response of the car becomes sluggish. A good compromise is to choose $\zeta = 1$ (critically damped), the value we consider in this task. Many second-order systems, such as dynamic feedback control systems, are chosen to have a critically damped response.

In this task, you are given a main script. You are asked to write a function **`Hautosusp.m`**, which accepts input parameters **`omega`**, **`omegan`** and **`zeta`**, and returns the frequency response of the automobile suspension (5b). The main script call it as **`Hautosusp(k*omega0,omegan,zeta)`**.

You are asked to run the main script and turn in plots of $x(t)$ and $y(t)$ for the following values of the natural frequency. Comment briefly on the differences between the plots and provide an intuitive interpretation of these differences.

```
omegan = 10*omega0;       % stiff (sports car)
omegan = 1*omega0;        % moderate (standard car)
omegan = 0.1*omega0;      % soft (luxury car)

%% Task 3: automobile rolling over periodic speed bumps

clear all
lw = 1.5;                 % line width for plots

T0 = 1;                   % period
omega0 = 2*pi/T0;         % fundamental frequency
T1 = T0/32;               % pulse half-width, T1 < T0/2 to avoid overlap
K = 256;                  % FS reconstruction sums from -K to +K
deltat = T0/K/8;          % discretization interval chosen to give 8
                          % samples per cycle of highest freq. comps.
t1 = -3*T0/2; t2 = 3*T0/2;
t = t1:deltat:t2;         % time vector for all signals

% Automobile suspension system
omegan = 1*omega0;        % natural frequency
zeta = 1;                 % damping constant

% y(t), Fourier series synthesis of output using terms from -K to K
x = zeros(size(t));
yautosusp = zeros(size(t));
for k = -K:K;
    ak = a_rectpulsetrain(k,omega0,T1);
    x = x + ak*exp(j*k*omega0*t);
    yautosusp = yautosusp +
ak*Hautosusp(k*omega0,omegan,zeta)*exp(j*k*omega0*t);
end

% display results
figure(3)
subplot(211)
plot(t,real(x)); grid
l=get(gca,'children'); set(l,'linewidth',lw)
xlabel('Time \itt'); ylabel('Road Elevation \itx\rm(\itt\rm)');

subplot(212)
plot(t,real(yautosusp)); grid
l=get(gca,'children'); set(l,'linewidth',lw)
xlabel('Time \itt'); ylabel('Automobile Elevation \ity\rm(\itt\rm)');
title(['Auto Suspension, \omega_{\itn}/\omega_{0} = '
num2str(omegan/omega0,3),...
   ', \zeta = ' num2str(zeta,3)]);
```