```
% HW07
% Q. 3

% ------------ clear all ----------------
close all;
clear all;
clc;

% ------------ (a) ----------------------
t = 0: 0.01: 1;
func_x = @(t) sin(2*pi*t);
x_c = func_x(t);
x_c_0 = x_c; % save for later

f = figure(1);
plot(t, x_c, 'linewidth',2)
xlim([0,1])
ylim([-1.2, 1.2])
grid on
xlabel('Time (t) [secs.]')
ylabel('$\mathbf{x_c(t)}$', 'interpreter', 'latex')
saveas(f, 'hw07_3a.eps', 'epsc');

% ------------- (b) & (c) & (d) -----------------------
f = figure(2);

% ----- sample time 0.02 secs -------------
% --- (b) ---
tol = 1e-9;
t = 0: 0.02: 1;
x_c = func_x(t);

subplot(3,2,1)
stem(t, x_c, 'linewidth',2)
xlim([0,1])
ylim([-1.2, 1.2])
set(gca,'Xtick',0 : .2 : 1)
set(gca,'Ytick',-1.2 : .4 : 1.2)
grid on
xlabel('Time (t) [secs.]')
ylabel('$\mathbf{x_c(t)}$', 'interpreter', 'latex')
title("(a-i)", 'Units', 'normalized', 'Position', [0.5, -0.55, 0])

% --- (c) ---
t0 = 0: 0.01: 1;
T = (t(2)-t(1))/(t0(2)-t0(1));
N = length(x_c_0); % CT signal
N_c = length(x_c);
y_c = zeros(1,N);
for m = 1: N
    for n = 1: N_c
```

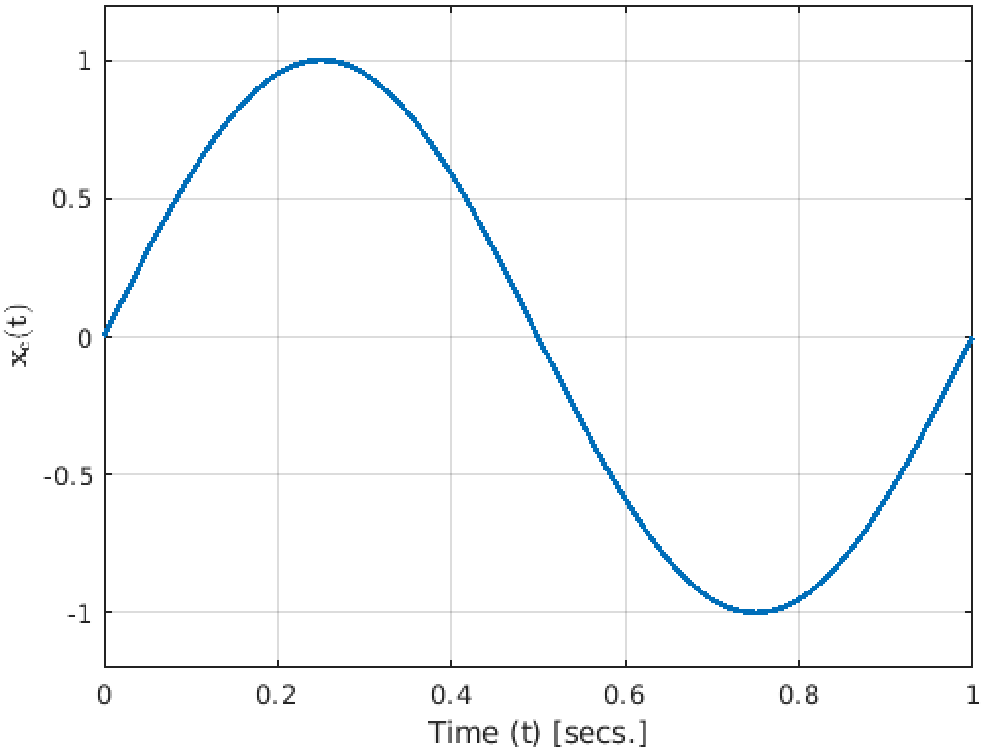
```
        y_c(m) = y_c(m) + x_c(n)*sin( pi*(m/T-n) +tol)/(pi*(m/T-n) +
 tol);
    end
end

subplot(3,2,2)
plot(t0, y_c, 'linewidth',2)
xlim([0,1])
ylim([-1.2, 1.2])
set(gca,'Xtick',0 : .2 : 1)
set(gca,'Ytick',-1.2 : .4 : 1.2)
grid on
xlabel('Time (t) [secs.]')
ylabel('$\mathbf{y_c(t)}$', 'interpreter', 'latex')
title("(a-ii)", 'Units', 'normalized', 'Position', [0.5, -0.55, 0])

% --- (d) ---
err = immse(x_c_0, y_c);
fprintf("With t=%g, Mean square error is %g\n", t(2)-t(1), err);


% ----- sample time 0.05 secs -------------
% --- (b) ---
t = 0: 0.05: 1;
x_c = func_x(t);
subplot(3,2,3)
stem(t, x_c, 'linewidth',2)
xlim([0,1])
ylim([-1.2, 1.2])
set(gca,'Xtick',0 : .2 : 1)
set(gca,'Ytick',-1.2 : .4 : 1.2)
grid on
xlabel('Time (t) [secs.]')
ylabel('$\mathbf{x_c(t)}$', 'interpreter', 'latex')
title("(b-i)", 'Units', 'normalized', 'Position', [0.5, -0.55, 0])

% --- (c) ---
t0 = 0: 0.01: 1;
T = (t(2)-t(1))/(t0(2)-t0(1));
N = length(x_c_0); % CT signal
N_c = length(x_c);
y_c = zeros(1,N);
for m = 1: N
    for n = 1: N_c
        y_c(m) = y_c(m) + x_c(n)*sin( pi*(m/T-n) +tol)/(pi*(m/T-n) +
 tol);
    end
end

subplot(3,2,4)
plot(t0, y_c, 'linewidth',2)
xlim([0,1])
ylim([-1.2, 1.2])
set(gca,'Xtick',0 : .2 : 1)
set(gca,'Ytick',-1.2 : .4 : 1.2)
grid on
xlabel('Time (t) [secs.]')
ylabel('$\mathbf{y_c(t)}$', 'interpreter', 'latex')
title("(b-ii)", 'Units', 'normalized', 'Position', [0.5, -0.55, 0])

% --- (d) ---
err = immse(x_c_0, y_c);
fprintf("With t=%g, Mean square error is %g\n", t(2)-t(1), err);

% ----- sample time 0.10 secs -------------
% --- (b) ---
t = 0: 0.10: 1;
x_c = func_x(t);
subplot(3,2,5)
stem(t, x_c, 'linewidth',2)
xlim([0,1])
ylim([-1.2, 1.2])
grid on
set(gca,'Xtick',0 : .2 : 1)
set(gca,'Ytick',-1.2 : .4 : 1.2)
xlabel('Time (t) [secs.]')
ylabel('$\mathbf{x_c(t)}$', 'interpreter', 'latex')
title("(c-i)", 'Units', 'normalized', 'Position', [0.5, -0.55, 0])

% --- (c) ---
t0 = 0: 0.01: 1;
T = (t(2)-t(1))/(t0(2)-t0(1));
N = length(x_c_0); % CT signal
N_c = length(x_c);
y_c = zeros(1,N);
for m = 1: N
    for n = 1: N_c
        y_c(m) = y_c(m) + x_c(n)*sin( pi*(m/T-n) +tol)/(pi*(m/T-n) +
 tol);
    end
end

subplot(3,2,6)
plot(t0, y_c, 'linewidth',2)
xlim([0,1])
ylim([-1.2, 1.2])
set(gca,'Xtick',0 : .2 : 1)
set(gca,'Ytick',-1.2 : .4 : 1.2)
grid on
xlabel('Time (t) [secs.]')
ylabel('$\mathbf{y_c(t)}$', 'interpreter', 'latex')
title("(c-ii)", 'Units', 'normalized', 'Position', [0.5, -0.55, 0])

% --- (d) ---
err = immse(x_c_0, y_c);
fprintf("With t=%g, Mean square error is %g\n", t(2)-t(1), err);

saveas(f, 'hw07_3bc.eps', 'epsc');
```

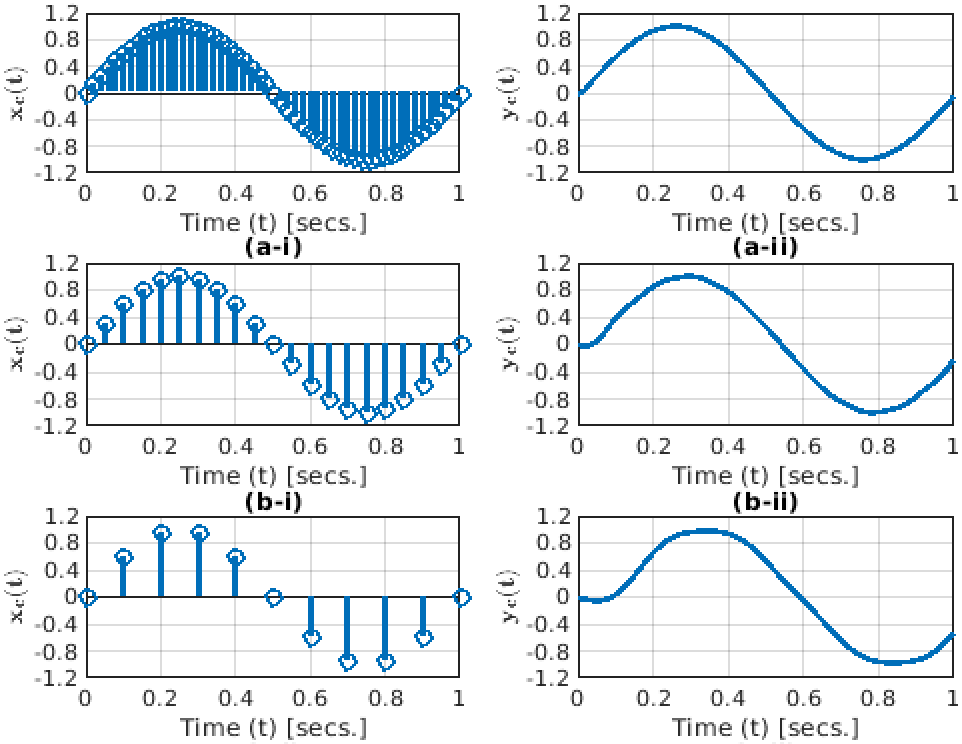
```
% ------------------------------ (e), (f) -------------------------
f=figure(3);

% ---- sampled at 0.02 secs -----
tol = 1e-9;
t = 0: 0.02: 1;
x_c = func_x(t);

% ---- quantized 2-bits -----
N_bits = 2;
L = power(2, N_bits);
min_level = -1;
max_level = 1;
x = round((x_c-min_level)/(max_level-min_level) * (L-1), 0);

subplot(3,1,1)
stem(t, x, 'linewidth',2)
xlim([0,1])
ylim([0, L-1])
grid on
set(gca,'Xtick',0 : 0.2 : 1)
set(gca,'Ytick',0 :1 : L)
xlabel('Time (t) [secs.]')
ylabel(sprintf("x [%d bits]", N_bits))
title("(a)", 'Units', 'normalized', 'Position', [0.5, -0.55, 0])

% --- DAC ---
x_q = x/(L-1) * (max_level-min_level) + min_level;
t0 = 0: 0.01: 1;
T = (t(2)-t(1))/(t0(2)-t0(1));
N = length(x_c_0); % CT signal
N_c = length(x_q);
y_c = zeros(1,N);
for m = 1: N
    for n = 1: N_c
        y_c(m) = y_c(m) + x_q(n)*sin( pi*(m/T-n) +tol)/(pi*(m/T-n) +
 tol);
    end
end

% --- MSE ---
err = immse(x_c_0, y_c);
fprintf("With t=%g, bits=%d, level=%d Mean square error is %g\n",
 t(2)-t(1), N_bits, L, err);



% ---- quantized 3-bits -----
N_bits = 3;
L = power(2, N_bits);
```

```
min_level = -1;
max_level = 1;
x = round((x_c-min_level)/(max_level-min_level) * (L-1), 0);

subplot(3,1,2)
stem(t, x, 'linewidth',2)
xlim([0,1])
ylim([0, L-1])
grid on
set(gca,'Xtick',0 : 0.2 : 1)
set(gca,'Ytick',0 :2 : L)
xlabel('Time (t) [secs.]')
ylabel(sprintf("x [%d bits]", N_bits))
title("(b)", 'Units', 'normalized', 'Position', [0.5, -0.55, 0])

% --- DAC ---
x_q = x/(L-1) * (max_level-min_level) + min_level;
t0 = 0: 0.01: 1;
T = (t(2)-t(1))/(t0(2)-t0(1));
N = length(x_c_0); % CT signal
N_c = length(x_q);
y_c = zeros(1,N);
for m = 1: N
    for n = 1: N_c
        y_c(m) = y_c(m) + x_q(n)*sin( pi*(m/T-n) +tol)/(pi*(m/T-n) +
 tol);
    end
end

% --- MSE ---
err = immse(x_c_0, y_c);
fprintf("With t=%g, bits=%d, level=%d Mean square error is %g\n",
 t(2)-t(1), N_bits, L, err);




% ---- quantized 4-bits -----
N_bits = 4;
L = power(2, N_bits);
min_level = -1;
max_level = 1;
x = round((x_c-min_level)/(max_level-min_level) * (L-1), 0);

subplot(3,1,3)
stem(t, x, 'linewidth',2)
xlim([0,1])
ylim([0, L-1])
grid on
set(gca,'Xtick',0 : 0.2 : 1)
set(gca,'Ytick',0 :4 : L)
xlabel('Time (t) [secs.]')
ylabel(sprintf("x [%d bits]", N_bits))
```

```
title("(c)", 'Units', 'normalized', 'Position', [0.5, -0.55, 0])

% --- DAC ---
x_q = x/(L-1) * (max_level-min_level) + min_level;
t0 = 0: 0.01: 1;
T = (t(2)-t(1))/(t0(2)-t0(1));
N = length(x_c_0); % CT signal
N_c = length(x_q);
y_c = zeros(1,N);
for m = 1: N
    for n = 1: N_c
        y_c(m) = y_c(m) + x_q(n)*sin( pi*(m/T-n) +tol)/(pi*(m/T-n) +
 tol);
    end
end

% --- MSE ---
err = immse(x_c_0, y_c);
fprintf("With t=%g, bits=%d, level=%d Mean square error is %g\n",
 t(2)-t(1), N_bits, L, err);




saveas(f, 'hw07_3e.eps', 'epsc');
```

```
With t=0.02, Mean square error is 0.00195946
With t=0.05, Mean square error is 0.0304444
With t=0.1, Mean square error is 0.144214
With t=0.02, bits=2, level=4 Mean square error is 0.033179
With t=0.02, bits=3, level=8 Mean square error is 0.00831795
With t=0.02, bits=4, level=16 Mean square error is 0.00343567
```

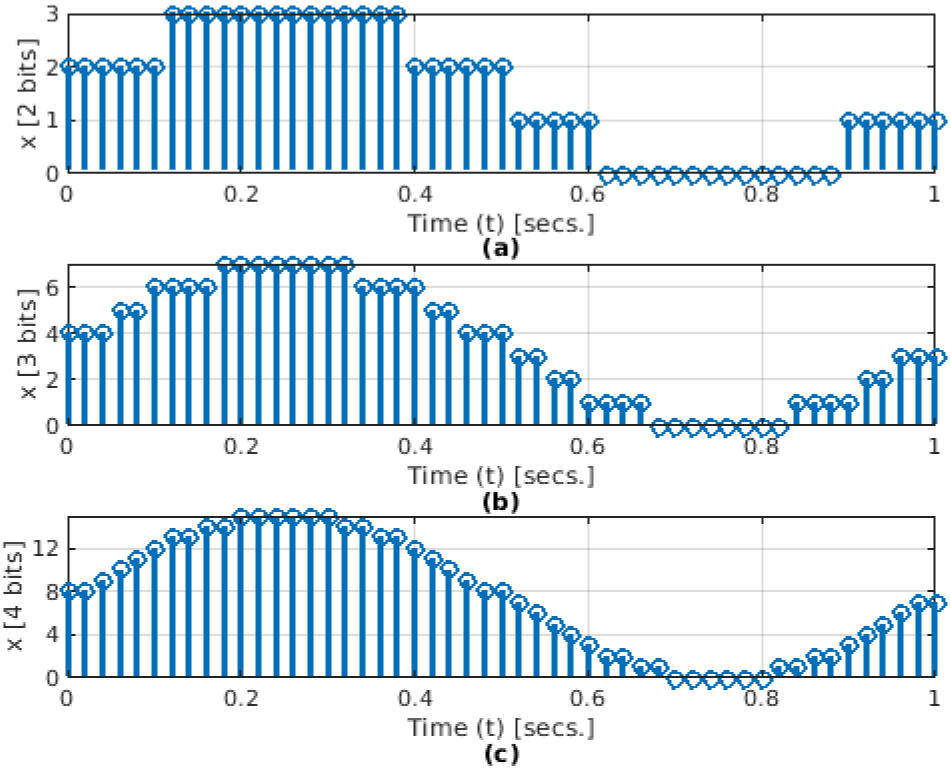




(a-i) **(a-ii)**

(b-i) **(b-ii)**

(c-i) **(c-ii)**



Published with MATLAB® R2018a