```
%%Digital Signal Processing Laboratory Assignment 3
%%Minsung Cho
%2020/10/29

% # Sampling
%1.a Determine m_a(t) and show that s_a(t):
%mat(k) = 1/(pi*t)*sin(wc*t);

%S_a(t) = (1-cos(wc*t))/(pi*wc*t^2)
%s_a(t) = 1/(2*pi)*(integral from -wc to 0 of %(1/wc*w+1)*e^(jwt)dw +
 integral from 0 to wc of (-1/wc*w+1)*e^(jwt)dw)
% = (e^(-300j*pi*t)-e^(300j*pi*t))(-1+300j*pi*t)/(600*pi^2*t^2)
% = (1-cos(wc*t))/(pi*wc*t.^2)

% 1.b Using Matlab, sample m_a(t) and s_a(t) for -1<=t<=1 at the
 following
% sampling rates.
wc = 300*pi;
ot = wc/(2*pi);
%sat = (1-cos(wc*t))/(pi*wc*t.^2);
k = 1;
time = -3*pi/wc:pi/10000:3*pi/wc;
for t = time
    if t == 0
        mat(k) = wc/pi;
    else
        mat(k) = 1/(pi*t)*sin(wc*t);
    end
    k = k+1;
end

figure(1)
plot(time,mat,'linewidth',2)
title({'The Inverse CTFT of \it Ma(j\Omega)\rm\bf','The Analog Signal,
\itM_a(t)'})
grid on
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

clearvars

%1.b.i) f_s = Nyquist rate = 2*f
wc = 300*pi;
ot = wc/(2*pi);
fsn = 300;
T1 = 1/fsn;
k = 1;
nTime = -30:30;
for n = nTime
    if n == 0
        mat1(k) = wc/pi;
    else
```

```
        mat1(k) = 1/(pi*n*T1)*sin(n*wc*T1);
    end
    k = k + 1;
end

figure(2)
subplot(3,1,1)
stem(nTime,mat1,'-r','filled','linewidth',2)
title({'The Finite Length Sequence, \it M_a[n] \rm \bf'; 'At \itf\rm
\bf = Nyquist'},'fontname', 'Comic Sans MS', 'fontsize',14)
grid on
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

[H1,w1] = freqz(mat1,1,512);
mag1 = abs(H1);
rad2deg = 180/pi;
angleH1 = wrapTo180(unwrap(angle(H1))*rad2deg);

subplot(3,1,2)
plot(w1,mag1,'-r','linewidth',2)
grid on
title({'The DFT, \it M_a[\Omega] \rm \bf'; 'At \itf\rm\bf =
 Nyquist'},'fontname', 'Comic Sans MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

subplot(3,1,3)
plot(w1,angleH1,'-r','linewidth',2)
grid on
title({'The Angle of, \it M_a[\Omega] \rm \bf'; 'At \itf\rm\bf
 =Nyquist'},'fontname', 'Comic Sans MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)
set(gcf, 'Position', [100, 100, 800, 500])

clearvars

%1.b.ii)f_s = 900Hz
wc = 300*pi;
ot = wc/(2*pi);
fs2 = 900;
T2 = 1/fs2;
k = 1;
nTime = -30:30;
for n = nTime
    if n == 0
        mat2(k) = wc/pi;
    else
        mat2(k) = 1/(pi*n*T2)*sin(n*wc*T2);
    end
    k = k + 1;
end
```

```
figure(3)
subplot(3,1,1)
stem(nTime,mat2,'-r','filled','linewidth',2)
grid on
title({'The Finite Length Sequence, \it M_a[n] \rm \bf'; 'At \itf\rm
\bf = 900Hz'},'fontname', 'Comic Sans MS', 'fontsize',14)
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

[H2,w2] = freqz(mat2,1,512);
mag2 = abs(H2);
rad2deg = 180/pi;
angleH2 = wrapTo180(unwrap(angle(H2))*rad2deg);

subplot(3,1,2)
plot(w2,mag2,'-r','linewidth',2)
grid on
title({'The DFT, \it M_a[\Omega] \rm \bf'; 'At \itf\rm\bf = 900Hz'})
xlabel('\omega')
ylabel('Amplitude')

subplot(3,1,3)
plot(w2,angleH2,'-r','linewidth',2)
grid on
title({'The Angle of, \it M_a[\Omega] \rm \bf'; 'At \itf\rm\bf
 =900Hz'})
xlabel('\omega')
ylabel('Amplitude')
set(gcf, 'Position', [100, 100, 800, 500])

clearvars

%1.b.iii)f_s = 200Hz
wc = 300*pi;
ot = wc/(2*pi);
fs3 = 200;
T3 = 1/fs3;
k = 1;
nTime = -30:30;
for n = nTime
    if n == 0
        mat3(k) = wc/pi;
    else
        mat3(k) = 1/(pi*n*T3)*sin(n*wc*T3);
    end
    k = k + 1;
end

figure(4)
subplot(3,1,1)
stem(nTime,mat3,'-r','filled','linewidth',2)
grid on
title({'The Finite Length Sequence, \it M_a[n] \rm \bf'; 'At \itf\rm
\bf = 200Hz'},'fontname', 'Comic Sans MS', 'fontsize',14)
```

```
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

[H3,w3] = freqz(mat3,1,512);
mag3 = abs(H3);
rad2deg = 180/pi;
angleH3 = wrapTo180(unwrap(angle(H3))*rad2deg);

subplot(3,1,2)
plot(w3,mag3,'-r','linewidth',2)
grid on
title({'The DFT, \it M_a[\Omega] \rm \bf'; 'At \itf\rm\bf = 200Hz'})
xlabel('\omega')
ylabel('Amplitude')

subplot(3,1,3)
plot(w3,angleH3,'-r','linewidth',2)
grid on
title({'The Angle of, \it M_a[\Omega] \rm \bf'; 'At \itf\rm\bf
 =200Hz'})
xlabel('\omega')
ylabel('Amplitude')
set(gcf, 'Position', [100, 100, 800, 500])

clearvars

%Sampling s(a)
% = (1-cos(wc*t))/(pi*wc*t.^2)
wc = 300*pi;
ot = wc/(2*pi);
k = 1;
time = -3*pi/wc:pi/10000:3*pi/wc;
for t = time
    if t == 0
        sat(k) = wc/(2*pi);
    else
        sat(k) = (1-cos(wc*t))/(pi*wc*t.^2);
    end
    k = k+1;
end

figure(5)
plot(time,sat,'linewidth',2)
title({'The Inverse CTFT of \it Sa(j\Omega)\rm\bf','The Analog Signal,
\itS_a(t)'})
grid on
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

clearvars

%1.b.i) f_s = Nyquist rate = 2*f
wc = 300*pi;
ot = wc/(2*pi);
```

```
fsn = 300;
T1 = 1/fsn;
k = 1;
nTime = -30:30;
for n = nTime
    if n == 0
        sat1(k) = wc/(2*pi);
    else
        sat1(k) = (1-cos(wc*n*T1))/(pi*wc*(n*T1).^2);
    end
    k = k + 1;
end

figure(6)
subplot(3,1,1)
stem(nTime,sat1,'-r','filled','linewidth',2)
title({'The Finite Length Sequence, \it S_a[n] \rm \bf'; 'At \itf\rm
\bf = Nyquist'},'fontname', 'Comic Sans MS', 'fontsize',14)
grid on
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

[H1,w1] = freqz(sat1,1,512);
sag1 = abs(H1);
rad2deg = 180/pi;
angleH1 = wrapTo180(unwrap(angle(H1))*rad2deg);

subplot(3,1,2)
plot(w1,sag1,'-r','linewidth',2)
grid on
title({'The DFT, \it S_a[\Omega] \rm \bf'; 'At \itf\rm\bf =
 Nyquist'},'fontname', 'Comic Sans MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

subplot(3,1,3)
plot(w1,angleH1,'-r','linewidth',2)
grid on
title({'The Angle of, \it S_a[\Omega] \rm \bf'; 'At \itf\rm\bf
 =Nyquist'},'fontname', 'Comic Sans MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)
set(gcf, 'Position', [100, 100, 800, 500])

clearvars

%1.b.ii)f_s = 900Hz
wc = 300*pi;
ot = wc/(2*pi);
fs2 = 900;
T2 = 1/fs2;
k = 1;
nTime = -30:30;
for n = nTime
```

```
    if n == 0
        sat2(k) = wc/(2*pi);
    else
        sat2(k) = (1-cos(wc*n*T2))/(pi*wc*(n*T2).^2);
    end
    k = k + 1;
end

figure(7)
subplot(3,1,1)
stem(nTime,sat2,'-r','filled','linewidth',2)
grid on
title({'The Finite Length Sequence, \it S_a[n] \rm \bf'; 'At \itf\rm
\bf = 900Hz'},'fontname', 'Comic Sans MS', 'fontsize',14)
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

[H2,w2] = freqz(sat2,1,512);
sag2 = abs(H2);
rad2deg = 180/pi;
angleH2 = wrapTo180(unwrap(angle(H2))*rad2deg);

subplot(3,1,2)
plot(w2,sag2,'-r','linewidth',2)
grid on
title({'The DFT, \it S_a[\Omega] \rm \bf'; 'At \itf\rm\bf = 900Hz'})
xlabel('\omega')
ylabel('Amplitude')

subplot(3,1,3)
plot(w2,angleH2,'-r','linewidth',2)
grid on
title({'The Angle of, \it S_a[\Omega] \rm \bf'; 'At \itf\rm\bf
 =900Hz'})
xlabel('\omega')
ylabel('Amplitude')
set(gcf, 'Position', [100, 100, 800, 500])

clearvars

%1.b.iii)f_s = 200Hz
wc = 300*pi;
ot = wc/(2*pi);
fs3 = 200;
T3 = 1/fs3;
k = 1;
nTime = -30:30;
for n = nTime
    if n == 0
        sat3(k) = wc/(2*pi);
    else
        sat3(k) = (1-cos(wc*n*T3))/(pi*wc*(n*T3).^2);
    end
    k = k + 1;
```

```
end

figure(8)
subplot(3,1,1)
stem(nTime,sat3,'-r','filled','linewidth',2)
grid on
title({'The Finite Length Sequence, \it S_a[n] \rm \bf'; 'At \itf\rm
\bf = 200Hz'},'fontname', 'Comic Sans MS', 'fontsize',14)
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

[H3,w3] = freqz(sat3,1,512);
sag3 = abs(H3);
rad2deg = 180/pi;
angleH3 = wrapTo180(unwrap(angle(H3))*rad2deg);

subplot(3,1,2)
plot(w3,sag3,'-r','linewidth',2)
grid on
title({'The DFT, \it S_a[\Omega] \rm \bf'; 'At \itf\rm\bf = 200Hz'})
xlabel('\omega')
ylabel('Amplitude')

subplot(3,1,3)
plot(w3,angleH3,'-r','linewidth',2)
grid on
title({'The Angle of, \it S_a[\Omega] \rm \bf'; 'At \itf\rm\bf
 =200Hz'})
xlabel('\omega')
ylabel('Amplitude')
set(gcf, 'Position', [100, 100, 800, 500])

%Sampling rate conversion
%2.1.a f_s = 900Hz->300Hz

clearvars

wc = 300*pi;
ot = wc/(2*pi);
fs2 = 900;
T2 = 1/fs2;
k = 1;
nTime = -30:30;
for n = nTime
    if n == 0
        mat2(k) = wc/pi;
    else
        mat2(k) = 1/(pi*n*T2)*sin(n*wc*T2);
    end
    k = k + 1;
end

y = decimate(mat2, 3);
goodTime = 0:size(y,2)-1;
```

```
figure(9)
subplot(2,1,1)
stem(nTime, mat2,'-r','filled','linewidth',2)
grid on
title({'The Finite Length Sequence, \it M_a[n] \rm \bf'; 'At \itf\rm
\bf = 900Hz'},'fontname', 'Comic Sans MS', 'fontsize',14)
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

subplot(2,1,2)
stem(goodTime,y,'-r','filled','linewidth',2)
grid on
title({'The Finite Length Sequence, \it M_a[n] \rm \bf'; 'At \itf\rm
\bf = decimated 300Hz'},'fontname', 'Comic Sans MS', 'fontsize',14)
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

%2.1.b f_s = 300Hz -> 900Hz

clearvars

wc = 300*pi;
ot = wc/(2*pi);
fsn = 300;
T1 = 1/fsn;
k = 1;
nTime = -30:30;
for n = nTime
    if n == 0
        mat1(k) = wc/pi;
    else
        mat1(k) = 1/(pi*n*T1)*sin(n*wc*T1);
    end
    k = k + 1;
end

y = interp(mat1, 3);
badTime = 0:size(y,2)-1;

figure(10)
subplot(2,1,1)
stem(nTime,mat1,'-r','filled','linewidth',2)
title({'The Finite Length Sequence, \it M_a[n] \rm \bf'; 'At \itf\rm
\bf = Nyquist'},'fontname', 'Comic Sans MS', 'fontsize',14)
grid on
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

subplot(2,1,2)
stem(badTime,y,'-r','filled','linewidth',2)
title({'The Finite Length Sequence, \it M_a[n] \rm \bf'; 'At \itf\rm
\bf = 900Hz'},'fontname', 'Comic Sans MS', 'fontsize',14)
grid on
```

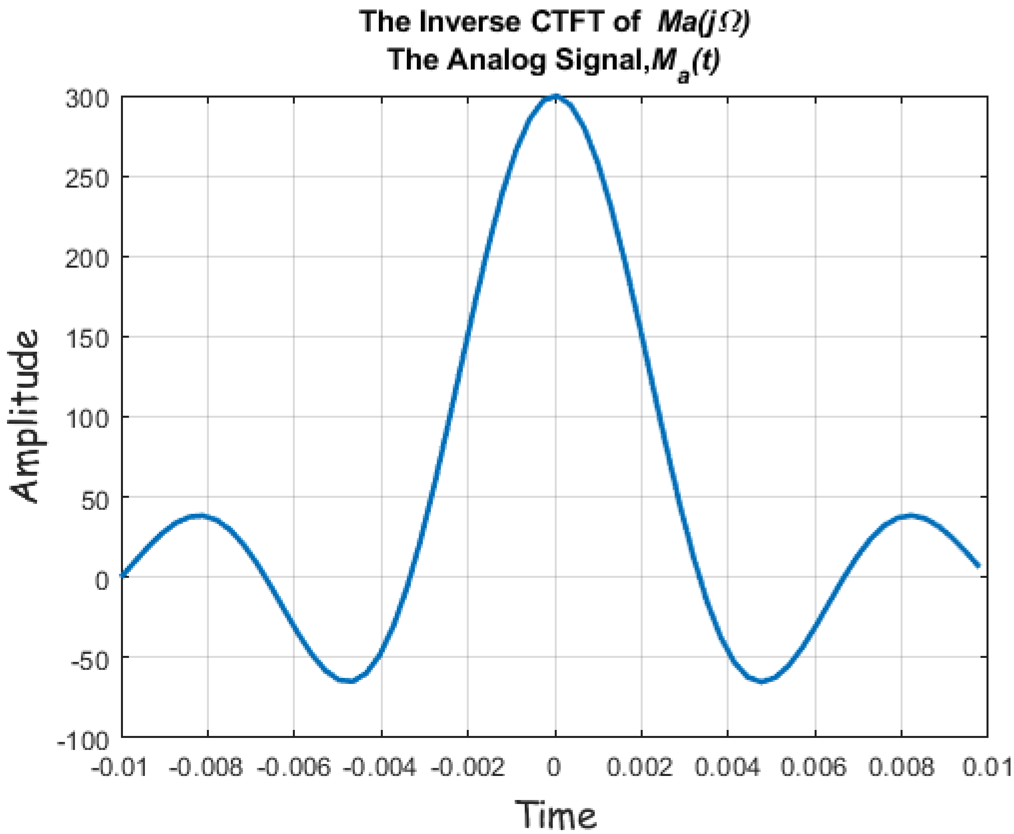
```
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

%2.c. Design and implement an interpolator that will bandlimit m(n) to
 the
%frequencies |w|<pi/3 and bandlimits s(n) to the frequencies pi/3<|w|
<pi.

clearvars

wc = 300*pi;
ot = wc/(2*pi);
fs2 = 900;
T2 = 1/fs2;
k = 1;
nTime = -30:30;
for n = nTime
    if n == 0
        mat2(k) = wc/pi;
    else
        mat2(k) = 1/(pi*n*T2)*sin(n*wc*T2);
    end
    k = k + 1;
end

for n = nTime
    if n == 0
        sat2(k) = wc/(2*pi);
    else
        sat2(k) = (1-cos(wc*n*T2))/(pi*wc*(n*T2).^2);
    end
    k = k + 1;
end

[m_filt, nm_filt] = FIRdesign(0, pi/3, 10);
[s_filt, ns_filt] = FIRdesign(pi/3, pi, 10);
[y, ny] = convolve(mat2, nTime, m_filt, nm_filt);
[g, ng] = convolve(sat2, nTime, s_filt, ns_filt);
t_chan = y + g;

figure(11)
subplot(3,3,1)
stem(ny, t_chan)
title({'The Transmission channel of m(n) and s(n) combined,  N =
 10'},'fontname', 'Comic Sans MS', 'fontsize',14)
grid on
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

[H1,w1] = freqz(t_chan,1,512);
mag1 = abs(H1);
rad2deg = 180/pi;
angleH1 = wrapTo180(unwrap(angle(H1))*rad2deg);
```

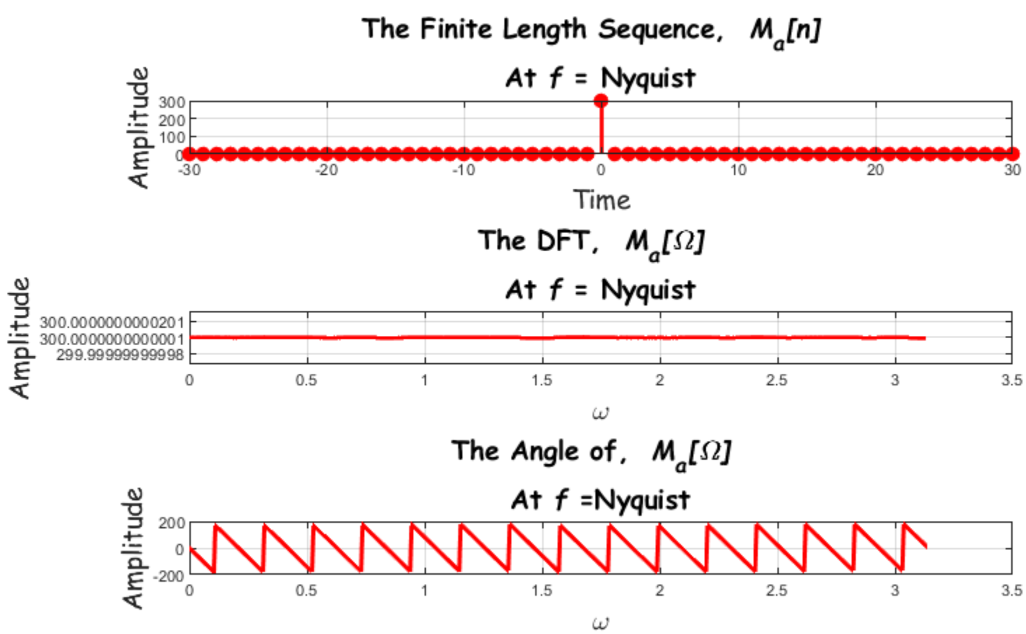
```
subplot(3,3,2)
plot(w1,mag1,'-r','linewidth',2)
grid on
title({'The DFT of Transmission Channel'},'fontname', 'Comic Sans
 MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

subplot(3,3,3)
plot(w1,angleH1,'-r','linewidth',2)
grid on
title({'The Angle of Transmission Channel'},'fontname', 'Comic Sans
 MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)
set(gcf, 'Position', [100, 100, 800, 500])

subplot(3,3,4)
stem(ny, y)
title({'m(n) separated'},'fontname', 'Comic Sans MS', 'fontsize',14)
grid on
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

[H2,w2] = freqz(y,1,512);
mag2 = abs(H2);
rad2deg = 180/pi;
angleH2 = wrapTo180(unwrap(angle(H2))*rad2deg);

subplot(3,3,5)
plot(w2,mag2,'-r','linewidth',2)
grid on
title({'The DFT of separated m(n)'},'fontname', 'Comic Sans
 MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

subplot(3,3,6)
plot(w2,angleH2,'-r','linewidth',2)
grid on
title({'The Angle of separated m(n)'},'fontname', 'Comic Sans
 MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)
set(gcf, 'Position', [100, 100, 800, 500])

subplot(3,3,7)
stem(ng, g)
title({'s(n) separated'},'fontname', 'Comic Sans MS', 'fontsize',14)
grid on
xlabel('Time','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

[H3,w3] = freqz(g,1,512);
```

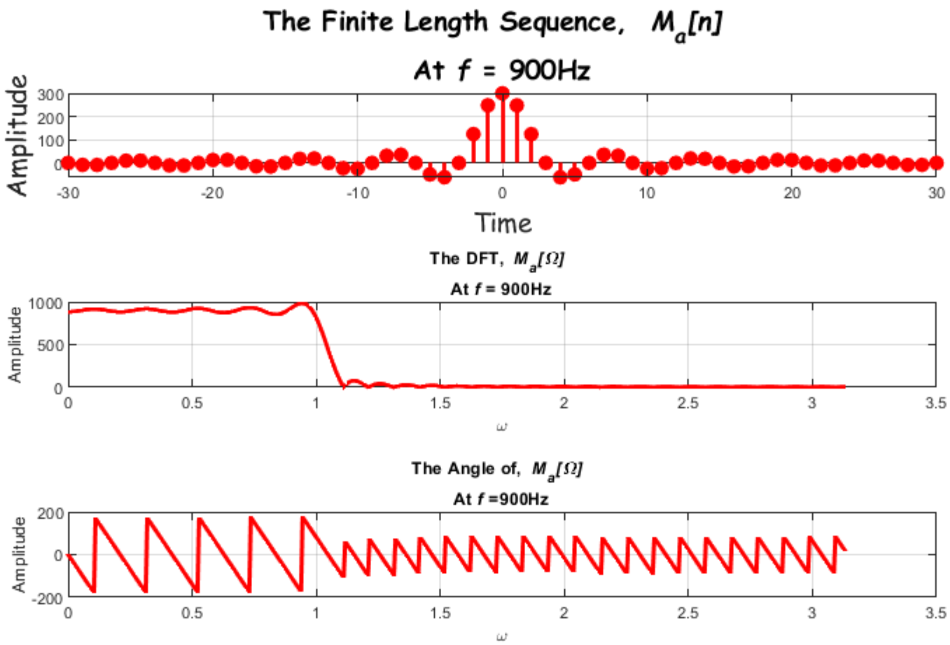
```
mag3 = abs(H3);
rad2deg = 180/pi;
angleH3 = wrapTo180(unwrap(angle(H3))*rad2deg);

subplot(3,3,8)
plot(w3,mag3,'-r','linewidth',2)
grid on
title({'The DFT of separated s(n)'},'fontname', 'Comic Sans
 MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

subplot(3,3,9)
plot(w3,angleH3,'-r','linewidth',2)
grid on
title({'The Angle of separated s(n)'; 'At \itf\rm\bf
 =Nyquist'},'fontname', 'Comic Sans MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)
set(gcf, 'Position', [100, 100, 800, 500])

%Changing the Sampling rate by a Noninteger Factor

clearvars

wm = 40*pi;
ot = wm/(2*pi);
k = 1;
time = 0:0.0001:0.25;
for t = time
        xat(k) = cos(wm*t);
    k = k+1;
end

fx = 300/ot;
T1 = 1/fx;
k = 1;
nTime = -30:30;
for n = nTime
    xat1(k) = cos(wm*n*T1);
    k = k + 1;
end

figure(12)
subplot(2,1,1)
plot(time, xat)
grid on
title({'x_a(t)'},'fontname', 'Comic Sans MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)

subplot(2,1,2)
stem(nTime, xat1)
grid on
```

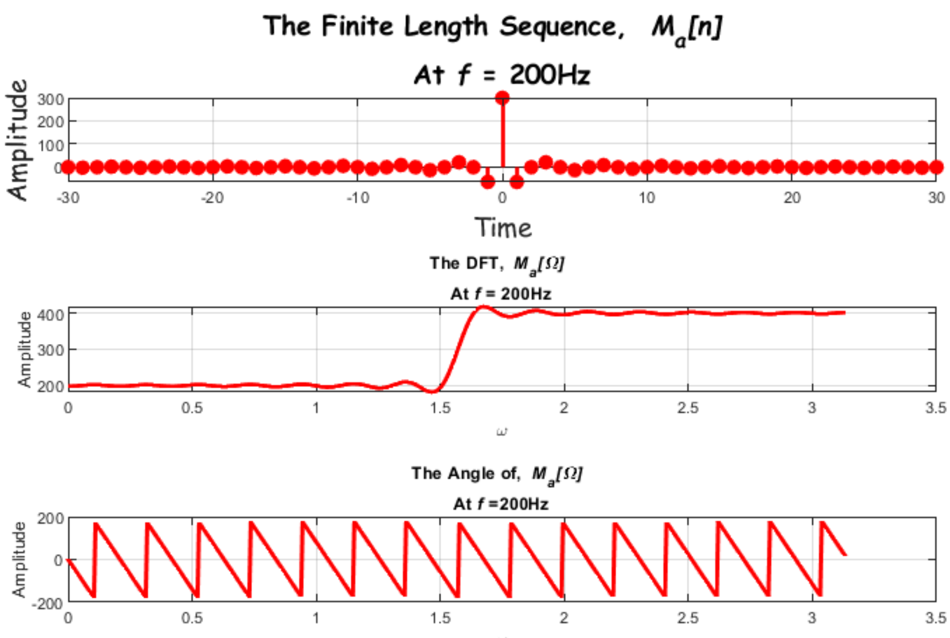
```
title({'Systematically adjusted to 300Hz sampling'},'fontname', 'Comic
 Sans MS', 'fontsize',14)
xlabel('\omega','fontname', 'Comic Sans MS', 'fontsize',14)
ylabel('Amplitude','fontname', 'Comic Sans MS', 'fontsize',14)


%finite impulse response frequency selective filter function
function [y, ny] = FIRdesign(wl, wu, N)
    %make one-sided h
    for n = 1:2*N
        h(n) = (sin(wu*n)-sin(wl*n))/(pi*n);
    end



    %the h is finite with the double the length of the window
    n = [1:2*N];

    %the process of making h hermitian even
    [h_r, n_r] = reverse(h, n);
    [h_comp, h_r_comp, n_comp] = compsig(h, n, h_r, n_r);
    h_comp = h_comp + h_r_comp;

    %the process to make the box to sweep
    for n = 1:N
        box(n) = 1;
    end
    n_box = -(N-1)/2:(N-1)/2;

    %colvolution of the h and the window
    [y, ny] = convolve(h_comp, n_comp, box, n_box);
end

%impulse convolution funciton
function y = convolution(h)
    prompt = 'Enter your impulse response: ';
    x = 3;
    xt = 1:x;
    x = [zeros(1, length(xt)-1)];
    y = [x h];
end

%convolution function
function [y, ny] = convolve(h, nh, x, nx)
    [x_r, nx_r] = reverse (x, nx);
    k = length(nh);
    x_r = [x_r zeros(1,k-1)];
    y = zeros(1, length(x_r));
    n = 0;
    while(k>=0)
            y = y + x_r.*h(find(h,1,'first')+n);
            k = k-1;
            x_r = circshift(x_r, 1);
            x_r(1) = 0;
```

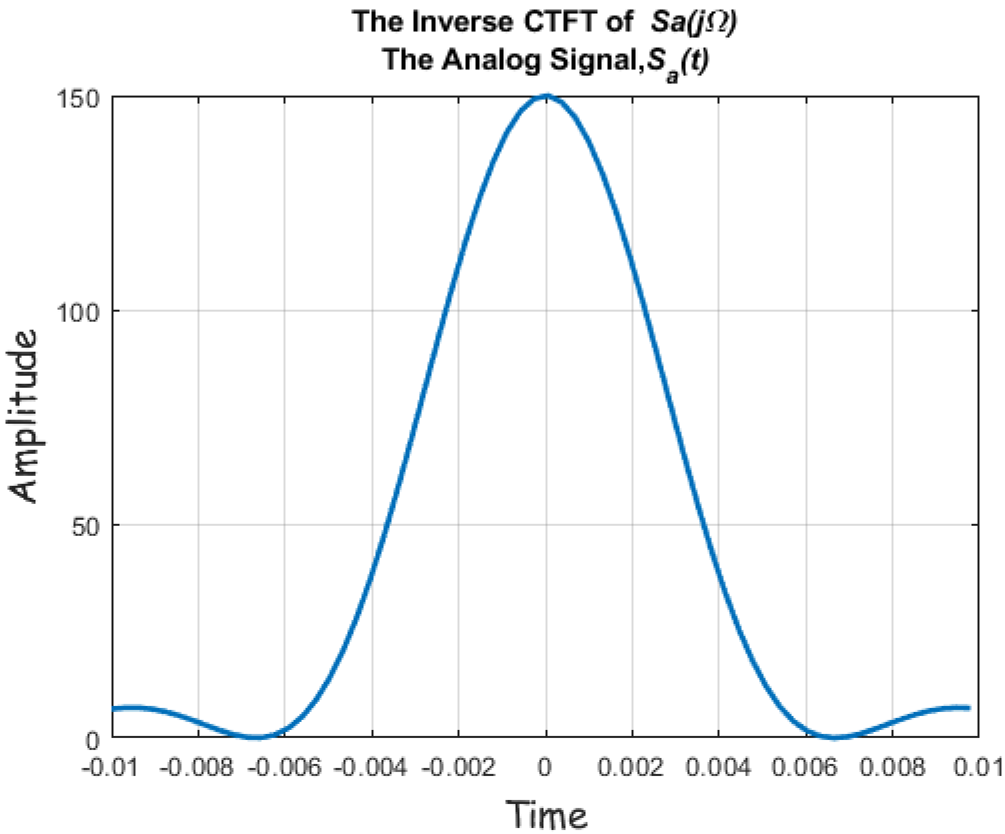
```
            if(find(h, 1, 'first')+n == find(h, 1, 'last'));
                k = -1;
            end
            n = n + 1;
    end
    ny = min(nh)-length(x)+1:max(nh);
end

%shift function
function [s, n] = shift (x, N, nx)
s = x;
n = nx + N;
end

%reverse function
function [r, k] = reverse (x, nx)
r = fliplr(x);
k = -fliplr(nx);
end

% compsig function
function [s1, s2, n] = compsig(x1, n1, x2, n2)
    nmin = min([min(n1), min(n2)]);     %determines the minimum time
 index.
    nmax = max([max(n1), max(n2)]);     %determines the maximum time
 index.
    n = nmin:nmax;                      %time indices are set.

    nsize = size(n, 2);                 %size to be
    [s1, s2] = deal(zeros(1, nsize));   %s1 and s2 are initialized
 with size n.

    x1size = size(x1,2);                %determines the size of x1.
    x2size = size(x2,2);                %determines the size of x2.
    x1first = find(n == n1(1));         %finds the
    x2first = find(n == n2(1));

    switch (n1(1) < n1(2))              %as long as the index is
 incremental
        case true                       %the right index for x1 is
 transferred.
            s1(x1first : x1first + x1size - 1) = x1;
        otherwise                       %this is when the order is
 flipped.
            s1(x1first - x1size + 1 : x1first) = fliplr(x1);
    end

    switch (n2(1) < n2(2))
        case true
            s2(x2first : x2first + x2size - 1) = x2;
        otherwise
            s2(x2first - x2size + 1 : x2first) = fliplr(x2);
    end
end
```

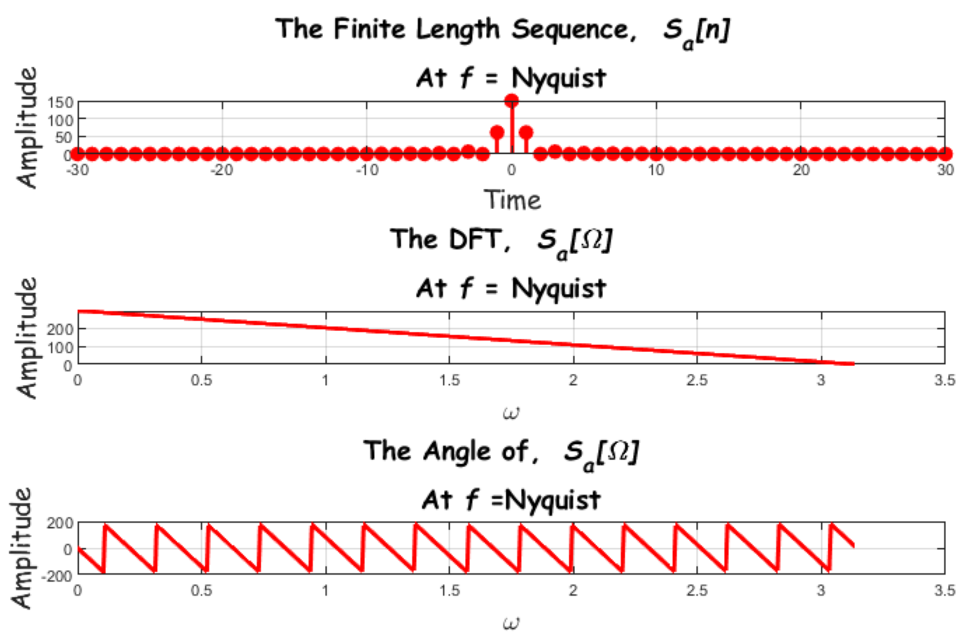

The Inverse CTFT of Ma(jΩ)
The Analog Signal, $M_a(t)$
300
250
200
150
100
50
0
-50
-100
-0.01 -0.008 -0.006 -0.004 -0.002 0 0.002 0.004 0.006 0.008 0.01
Amplitude
Time

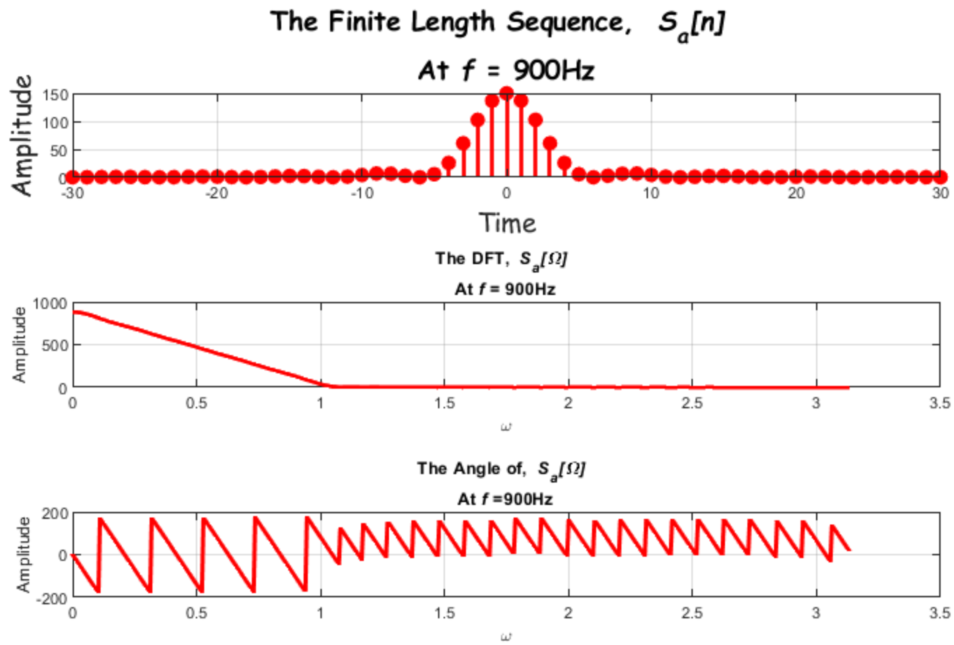

The Finite Length Sequence, $M_a[n]$
At f = Nyquist
Amplitude
300
200
100
0
-30 -20 -10 0 10 20 30
Time
The DFT, $M_a[\Omega]$
At f = Nyquist
Amplitude
300.0000000000201
300.0000000000001
299.99999999998
0 0.5 1 1.5 2 2.5 3 3.5
ω
The Angle of, $M_a[\Omega]$
At f =Nyquist
Amplitude
200
0
-200
0 0.5 1 1.5 2 2.5 3 3.5
ω

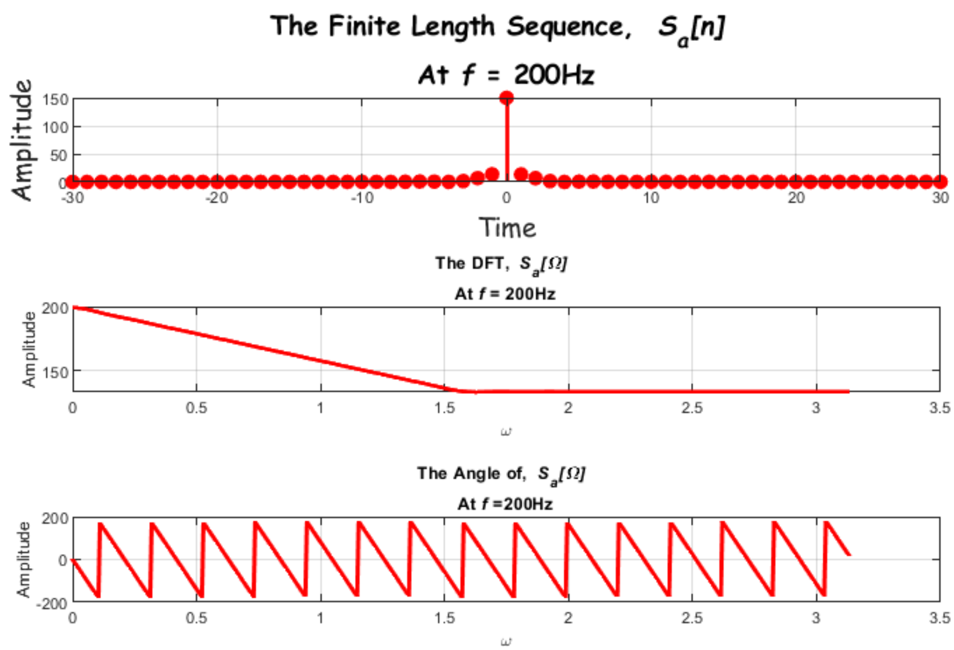

The Finite Length Sequence, $M_a[n]$
At f = 900Hz
Amplitude
Time
The DFT, $M_a[\Omega]$
At f = 900Hz
Amplitude
ω
The Angle of, $M_a[\Omega]$
At f =900Hz
Amplitude
ω

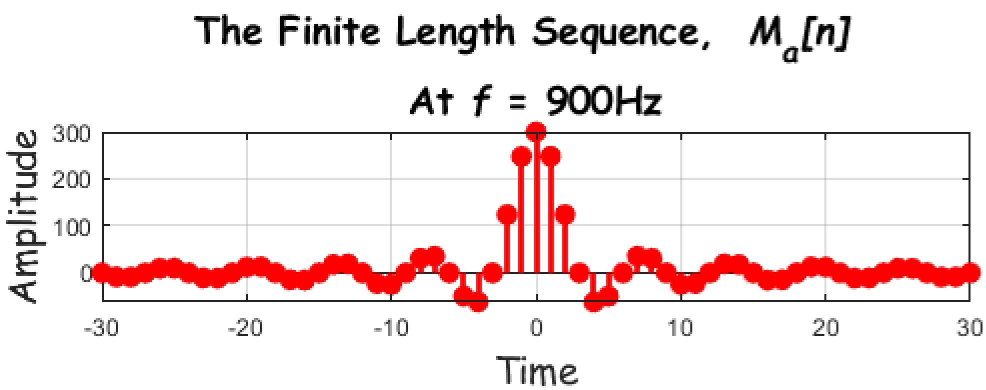

The Finite Length Sequence, $M_a[n]$
At f = 200Hz
Amplitude
Time
The DFT, $M_a[\Omega]$
At f = 200Hz
Amplitude
ω
The Angle of, $M_a[\Omega]$
At f =200Hz
Amplitude
ω

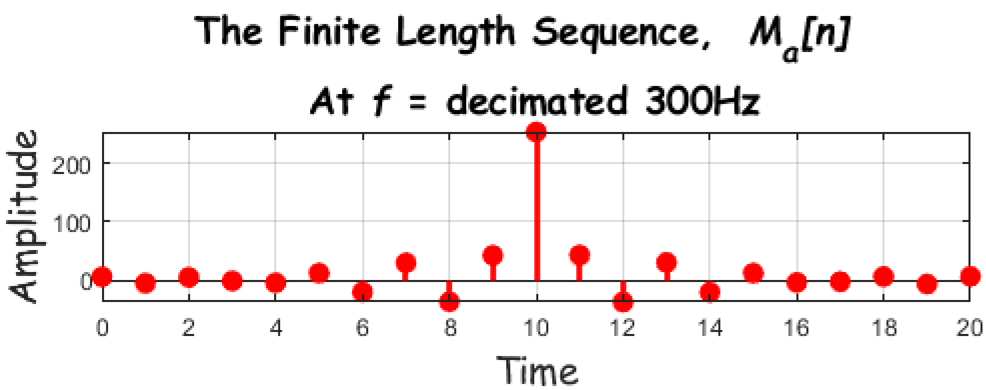

The Inverse CTFT of Sa(jΩ)
The Analog Signal, $S_a(t)$
150
100
50
0
Amplitude
-0.01 -0.008 -0.006 -0.004 -0.002 0 0.002 0.004 0.006 0.008 0.01
Time

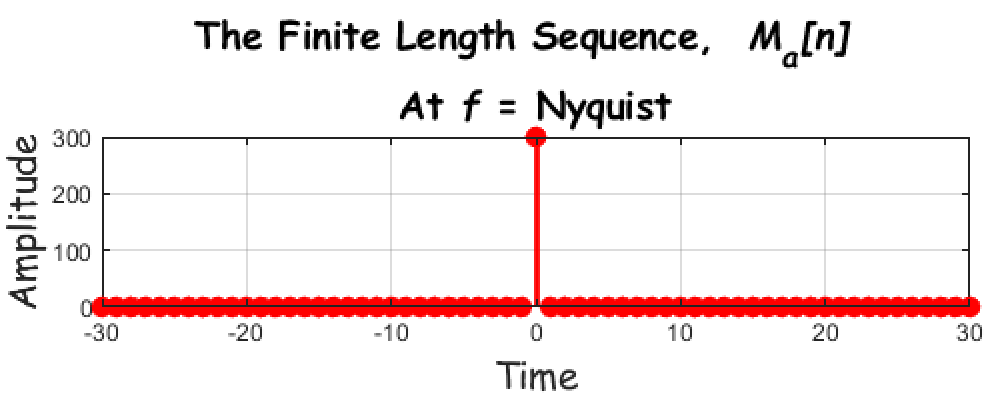

The Finite Length Sequence, $S_a[n]$
At f = Nyquist
Amplitude
150
100
50
0
-30 -20 -10 0 10 20 30
Time
The DFT, $S_a[\Omega]$
At f = Nyquist
Amplitude
200
100
0
0 0.5 1 1.5 2 2.5 3 3.5
ω
The Angle of, $S_a[\Omega]$
At f =Nyquist
Amplitude
200
0
-200
0 0.5 1 1.5 2 2.5 3 3.5
ω

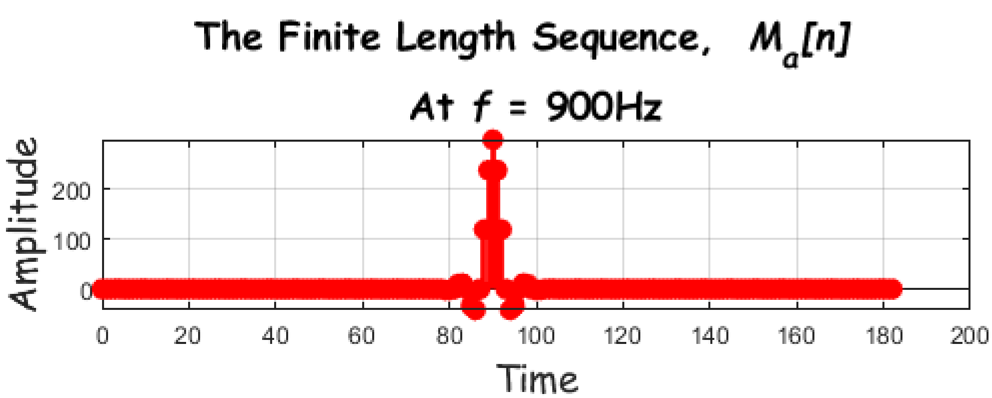

The Finite Length Sequence, $S_a[n]$
At f = 900Hz
Amplitude
Time
The DFT, $S_a[\Omega]$
At f = 900Hz
Amplitude
ω
The Angle of, $S_a[\Omega]$
At f =900Hz
Amplitude
ω

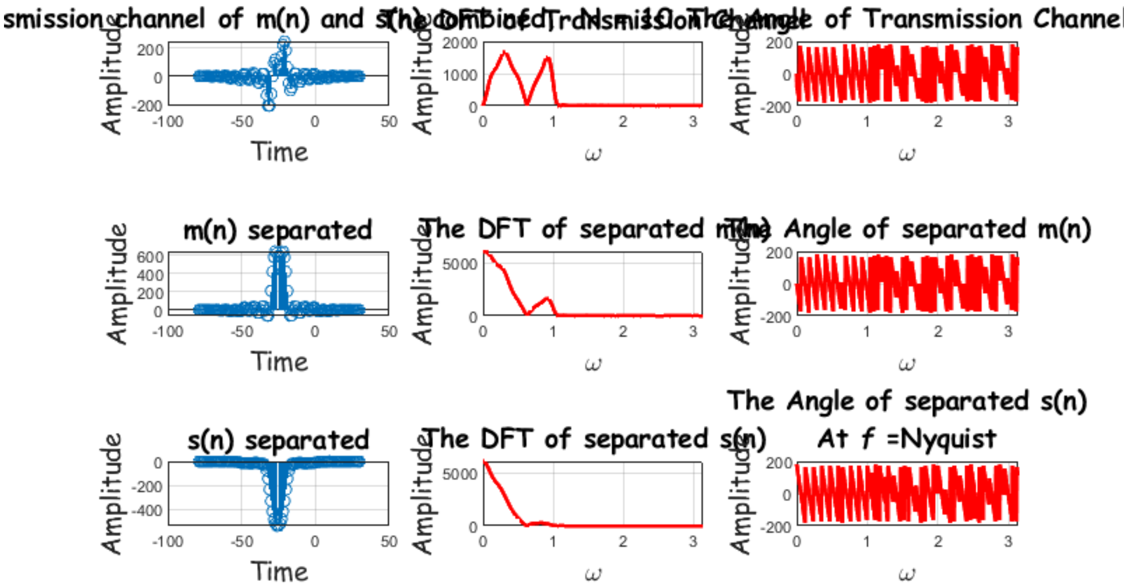

The Finite Length Sequence, $S_a[n]$
At f = 200Hz
Amplitude
Time
The DFT, $S_a[\Omega]$
At f = 200Hz
Amplitude
ω
The Angle of, $S_a[\Omega]$
At f =200Hz
Amplitude
ω

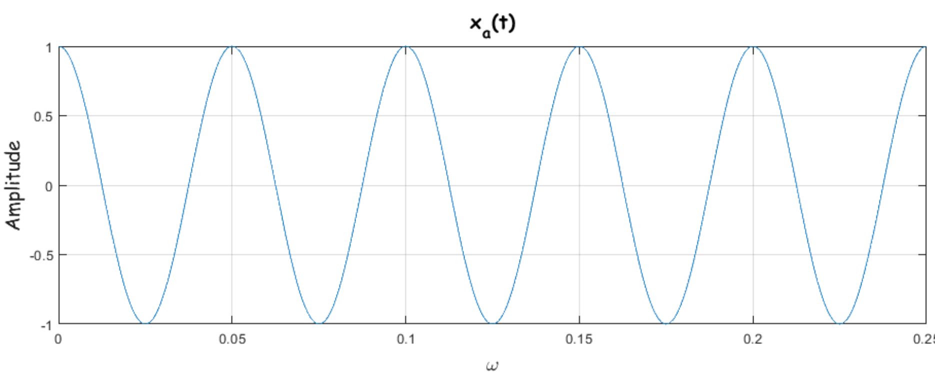

The Finite Length Sequence, $M_a[n]$
At f = 900Hz
Amplitude
300
200
100
0
-30
-20
-10
0
10
20
30
Time

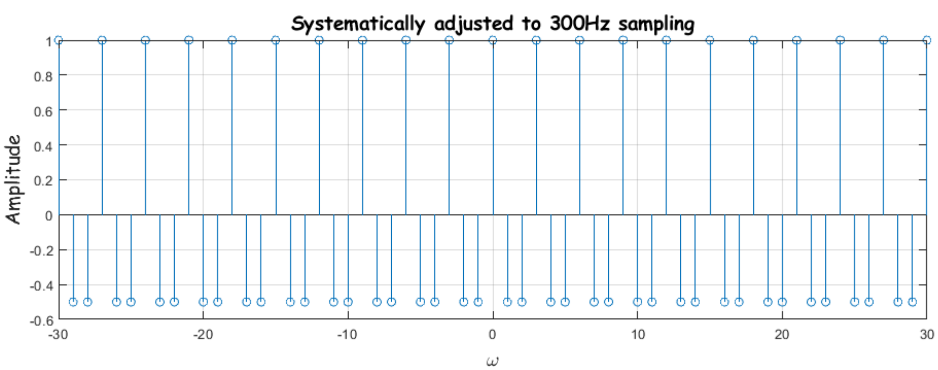

The Finite Length Sequence, $M_a[n]$
At f = decimated 300Hz
Amplitude
200
100
0
0
2
4
6
8
10
12
14
16
18
20
Time


The Finite Length Sequence, $M_a[n]$
At f = Nyquist
Amplitude
300
200
100
0
-30
-20
-10
0
10
20
30
Time


The Finite Length Sequence, $M_a[n]$
At f = 900Hz
Amplitude
200
100
0
0
20
40
60
80
100
120
140
160
180
200
Time







Published with MATLAB® R2020a