|  |
| --- |
| 1. (25%)  Consider the linear FM signal *xc*(*t*)= sin(*Bt*2/**) , 0 ≤ *t* ≤ *τ* with *B* = 10 Hz and *τ* = 10 s. It is applied to the talk-through system shown below with sampling rate of *Fs* = *B* Hz to obtain sampled signal *x*[*n*] and reconstructed signal *yr*(*t*). Simulate this operation in MATLAB and graph *xc*(*t*), *x*[*n*], and *yr*(*t*)in one figure using sub-plots. |
| close all; clear all;    %% parameters.  % analog  analog\_fps = 1500;  analog\_window\_time = 10; %sec  t = 0:1/analog\_fps: analog\_window\_time-1/analog\_fps;    % digital  digital\_fps = 5;  n = downsample(t,analog\_fps/digital\_fps);    % ADC: Quantizer  X\_m = 1; % Range  B = 10;% Bit number.    %% Signal generation  freq\_hz = 10; % Hz.  x\_c = sin(pi\*freq\_hz\*t.^2/10);    %% ADC  % Sampling  x\_s = downsample(x\_c,analog\_fps/digital\_fps);    % Quantizing (abs of input value should not over 1)  % x\_d = Quantizing(x\_s,B,X\_m); % A For complete ADC, a quantizing should  % be added here.  x\_n = x\_s; % For basic case, we skip the quantizing here.    %% DAC  % up sample / DAC  x\_up = upsample(x\_n,analog\_fps/digital\_fps);    % LPF (Reconstruction Filters)  h = intfilt(analog\_fps/digital\_fps,4,0.9);  %% Important  % please not the parameter 0.9, ideally should be 1 for Nyquist rate.  % 0.9 here is ratio of Nyquist.  % Given known limit band signal, shourter ratio can enhance SNR by oversampling.  % (i,e, here I filterout the freq larger than 2.5(Nyquist rate) \* 0.9 = 2.25Hz)    y\_r = filter(h,1,x\_up);  y\_r(1:floor(mean(grpdelay(h)))) = [];  y\_r = [y\_r zeros(1,floor(mean(grpdelay(h))))];      %% Display    figure;  plot(t,x\_c);  hold on;  plot(n,x\_n);  plot(t,y\_r);  title('analog signal (1500Hz) v.s. digital signal (5Hz) v.s. Reconstructed signal (1500Hz)');  legend('x\_c','x[n]','y\_r'); |
| 2. (25%)  A multiband ideal bandstop filter is given by    **(a)** Determine the impulse response of the filter.  **(b)** Graph the impulse response for *n*d = 0 for −200 ≤ *n* ≤ 200.  **(c)** From the above truncated impulse response, compute and plot the magnitude response of the filter using MATLAB and compare it with the ideal filter response. |
| (a)  (b)  clear all; close all;  %%  n = -200:200;  n\_d = 0;  h = sin(pi/8\*n)./(pi\*n) + 2/3\*(-1).^n.\*sin(3/8\*pi\*n)./(pi\*n) + 1/3\*(-1).^n.\*sin(7/8\*pi\*n)./(pi\*n);  figure;  plot(h)  (c)  %%  clc;clear all;  n=-200:1:200;  h1=(1/(2\*pi))\*((exp(j\*pi\*n/8))-1)./(j\*n);  h2=(1/(2\*pi))\*(2/3)\*((exp(j\*5\*pi\*n/8))-(exp(j\*3\*pi\*n/8)))./(j\*n);  h3=(1/(2\*pi))\*(1/3)\*((exp(j\*pi\*n))-(exp(j\*7\*pi\*n/8)))./(j\*n);  h=h1+h2+h3;  h(1,201)=1/6  w=0:(pi/1000):pi;  [H,w]=freqz(h,1,w);  figure;  plot(w/pi,abs(H),'r')  Ideal filter should not have any ripple. |
| 3. (25%)  An analog signal *xa*(*t*) = 5sin(200*t*) + 2cos(300*t*) is to be processed by a DSP system in which the sampling frequency is 1000 samples/sec.  (a) Design a minimum-order IIR digital filter that will pass the 150-Hz component with attenuation of less than 1 dB and suppress the 100-Hz component to at least 40 dB. The filter should have a monotone passband and an equiripple stopband. Determine the system function in rational function form and plot the log-magnitude response.  (b) Repeat (a) with optimal equiripple FIR filter.  (c) Generate 300 samples of *xa*(*t*) and show the processing results of (a) and (b). |
| (a)  wp = 150/500\*pi; rp = 1; ws = 100/500\*pi; as = 40;  ripple = 10^(-rp/20); attn = 10^(-as/20); fp =wp/(pi\*2);fs =ws/(pi\*2);  [N, wn] = cheb2ord(2\*fp, 2\*fs, rp, as); [b, a] = cheby2(N, as, wn, 'high');  [c,b1,a1] = sdir2cas(b,a);  [H,w] = freqz(b,a,1000,'whole');  H = (H(1:1:501))'; w = (w(1:1:501))';  mag = abs(H);  db = 20\*log10((mag+eps)/max(mag)); plot(w/pi,db,'LineWidth',2);grid;title('Magnitude in dB') xlabel('frequency in pi units'); axis([0 1 -60 0]); ylabel('decibels') set(gca,'XTickMode','manual','XTick',[0 ws/pi wp/pi 1]) set(gca,'YTickMode','manual','YTick',[-60 -as -rp 0])    (b)  clc ;clear all;  wp = 150/500\*pi; rp = 1; ws = 100/500\*pi; as = 40;  d1 = (10^(rp/20)-1)/(10^(rp/20)+1);  d2 = (1+d1)\*(10^(-as/20));  df = (wp-ws)/(pi\*2); weights = [1,d2/d1];  M = ceil((-20\*log10(sqrt(d1\*d2))-13)/(14.6\*df)+1); M = 2\*floor(M/2)+1;  f = [0,ws/pi,wp/pi,1]; m = [0,0,1,1];  h = remez(M-1,f,m,weights); [H,w] = freqz(h,1,1000,'whole');  H = (H(1:1:501))'; w = (w(1:1:501))';  mag = abs(H);  db = 20\*log10((mag+eps)/max(mag)); M = M+2;  h = remez(M-1,f,m,weights); [H,w] = freqz(h,1,1000,'whole');  H = (H(1:1:501))'; w = (w(1:1:501))';  mag = abs(H);  db = 20\*log10((mag+eps)/max(mag)); plot(w/pi,db,'LineWidth',2);grid;title('Magnitude Response in dB'); xlabel('frequency in pi units'); ylabel('DECIBELS') |
| 4. (25%)  A digital signal *x*[*n*] contains a sinusoid of frequency 0.5 and a Gaussian noise *w*[*n*] of zero mean and unit variance; i.e., *x*[*n*] = 2cos(0.5*n*) + *w*[*n*]. We want to filter out the noise component using a **50th-order causal linear-phase FIR** filter.   1. Using Parks-McClellan algorithm, design a **narrow bandpass filter** with passband width of no more than 0.02 and stopband attenuation of at least 30 dB. Note that no other parameters are given, and you have to choose the remaining parameters for the remez function to satisfy the requirements. Provide a plot of the log-magnitude response in dB of the designed filter. 2. Generate 200 samples of *x*[*n*] and process through the above filter to obtain the output *y*[*n*]. Provide subplots of *x*[*n*] and *y*[*n*] for 100 <= *n* <= 200 on one plot and comment on your results |
| (a)    (b)    由(a)知，實際的Filter並不是ideal BandPass，沒辦法濾掉sampling frequency  附近的訊號，因此產生兩個具有不同頻率弦波相加的效應。  (a)  clear; close all;  %% Specifications  N = 50; % Order of the filter  w0 = 0.5\*pi; % Center frequency  Bandwidth = 0.02\*pi; % Bandwidth  %  % Deltaw = Transition bandwidth (iteration variable)  %  wp1 = w0-Bandwidth/2; wp2 = w0+Bandwidth/2;  % (a) Design  Deltaw = 0.02\*pi; % Initial guess  ws1=wp1-Deltaw; ws2=wp2+Deltaw;  F=[0, ws1, wp1, wp2, ws2, pi]/pi;  m=[0,0,1,1,0,0];  h=remez(50,F,m);  %%%%%%%%%%%%%%%%%%%  [H,w] = freqz(h,1,1000,'whole');  H = (H(1:1:501))'; w = (w(1:1:501))';  mag = abs(H);  db = 20\*log10((mag+eps)/max(mag));  pha = angle(H);  % pha = unwrap(angle(H));  grd = grpdelay(h,1,w);  %%%%%%%%%%%%%%%%%%  %[db,mag,pha,grd,w]=freqz\_m(h,1);  delta\_w = pi/500;  Asd = floor(-max(db([1:floor(ws1/delta\_w)]))), % Actual Attn    % Next iteration  Deltaw = Deltaw+0.01\*pi;  ws1=wp1-Deltaw; ws2=wp2+Deltaw;  F=[0, ws1, wp1, wp2, ws2, pi]/pi;  h=remez(50,F,m);  %%%%%%%%%%%%%%%%%%%  [H,w] = freqz(h,1,1000,'whole');  H = (H(1:1:501))'; w = (w(1:1:501))';  mag = abs(H);  db = 20\*log10((mag+eps)/max(mag));  pha = angle(H);  % pha = unwrap(angle(H));  grd = grpdelay(h,1,w);  %%%%%%%%%%%%%%%%%%  %[db,mag,pha,grd,w]=freqz\_m(h,1);  delta\_w = pi/500;  Asd = floor(-max(db([1:floor(ws1/delta\_w)]))); % Actual Attn    % Next iterationDeltaw = Deltaw+0.01\*pi;  ws1=wp1-Deltaw; ws2=wp2+Deltaw;  F=[0, ws1, wp1, wp2, ws2, pi]/pi;  h=remez(50,F,m);  %%%%%%%%%%%%%%%%%%%  [H,w] = freqz(h,1,1000,'whole');  H = (H(1:1:501))'; w = (w(1:1:501))';  mag = abs(H);  db = 20\*log10((mag+eps)/max(mag));  pha = angle(H);  % pha = unwrap(angle(H));  grd = grpdelay(h,1,w);  %%%%%%%%%%%%%%%%%%  %[db,mag,pha,grd,w]=freqz\_m(h,1);  delta\_w = pi/500;  Asd = floor(-max(db([1:floor(ws1/delta\_w)]))); % Actual Attn  % Next iteration  Deltaw = Deltaw+0.01\*pi;  ws1=wp1-Deltaw; ws2=wp2+Deltaw;  F=[0, ws1, wp1, wp2, ws2, pi]/pi;  h=remez(50,F,m);  %%%%%%%%%%%%%%%%%%%  [H,w] = freqz(h,1,1000,'whole');  H = (H(1:1:501))'; w = (w(1:1:501))';  mag = abs(H);  db = 20\*log10((mag+eps)/max(mag));  pha = angle(H);  % pha = unwrap(angle(H));  grd = grpdelay(h,1,w);  %%%%%%%%%%%%%%%%%%  %[db,mag,pha,grd,w]=freqz\_m(h,1);  delta\_w = pi/500;  Asd = floor(-max(db([1:floor(ws1/delta\_w)]))); % Actual Attn    plot(w/pi,db,'g','linewidth',1.5); axis([0,1,-50,0]);  title('Log-Magnitude Response','fontsize',10);  xlabel('\omega/\pi','fontsize',10); ylabel('DECIBELS','fontsize',10)  set(gca,'XTick',[0;ws1/pi;ws2/pi;1],'YTick',[-30;0]);  set(gca,'YTickLabel',[30; 0 ]);grid  (b)  clc;clear all;  %  % Deltaw = Transition bandwidth (iteration variable)  %  N = 50; % Order of the filter  w0 = 0.5\*pi; % Center frequency  Bandwidth = 0.02\*pi; % Bandwidth  wp1 = w0-Bandwidth/2; wp2 = w0+Bandwidth/2;  % (a) Design  Deltaw = 0.02\*pi; % Initial guess  ws1=wp1-Deltaw; ws2=wp2+Deltaw;  F=[0, ws1, wp1, wp2, ws2, pi]/pi;  m=[0,0,1,1,0,0];  h=remez(50,F,m);  %%%%%%%%%%%%%%%%%%%  [H,w] = freqz(h,1,1000,'whole');  H = (H(1:1:501))'; w = (w(1:1:501))';  mag = abs(H);  db = 20\*log10((mag+eps)/max(mag));  pha = angle(H);  % pha = unwrap(angle(H));  grd = grpdelay(h,1,w);  %%%%%%%%%%%%%%%%%%  %[db,mag,pha,grd,w]=freqz\_m(h,1);  delta\_w = pi/500;  Asd = floor(-max(db([1:floor(ws1/delta\_w)]))), % Actual Attn    % Next iteration  Deltaw = Deltaw+0.01\*pi;  ws1=wp1-Deltaw; ws2=wp2+Deltaw;  F=[0, ws1, wp1, wp2, ws2, pi]/pi;  h=remez(50,F,m);  %%%%%%%%%%%%%%%%%%%  [H,w] = freqz(h,1,1000,'whole');  H = (H(1:1:501))'; w = (w(1:1:501))';  mag = abs(H);  db = 20\*log10((mag+eps)/max(mag));  pha = angle(H);  % pha = unwrap(angle(H));  grd = grpdelay(h,1,w);  %%%%%%%%%%%%%%%%%%  %[db,mag,pha,grd,w]=freqz\_m(h,1);  delta\_w = pi/500;  Asd = floor(-max(db([1:floor(ws1/delta\_w)]))); % Actual Attn    % Next iterationDeltaw = Deltaw+0.01\*pi;  ws1=wp1-Deltaw; ws2=wp2+Deltaw;  F=[0, ws1, wp1, wp2, ws2, pi]/pi;  h=remez(50,F,m);  %%%%%%%%%%%%%%%%%%%  [H,w] = freqz(h,1,1000,'whole');  H = (H(1:1:501))'; w = (w(1:1:501))';  mag = abs(H);  db = 20\*log10((mag+eps)/max(mag));  pha = angle(H);  % pha = unwrap(angle(H));  grd = grpdelay(h,1,w);  %%%%%%%%%%%%%%%%%%  %[db,mag,pha,grd,w]=freqz\_m(h,1);  delta\_w = pi/500;  Asd = floor(-max(db([1:floor(ws1/delta\_w)]))); % Actual Attn  % Next iteration  Deltaw = Deltaw+0.01\*pi;  ws1=wp1-Deltaw; ws2=wp2+Deltaw;  F=[0, ws1, wp1, wp2, ws2, pi]/pi;  h=remez(50,F,m);  %%%%%%%%%%%%%%%%%%%  [H,w] = freqz(h,1,1000,'whole');  H = (H(1:1:501))'; w = (w(1:1:501))';  mag = abs(H);  db = 20\*log10((mag+eps)/max(mag));  pha = angle(H);  % pha = unwrap(angle(H));  grd = grpdelay(h,1,w);  %%%%%%%%%%%%%%%%%%  %[db,mag,pha,grd,w]=freqz\_m(h,1);  delta\_w = pi/500;  Asd = floor(-max(db([1:floor(ws1/delta\_w)]))); % Actual Attn    n = [0:1:200]; x = 2\*cos(pi\*n/2)+randn(1,201); y = filter(h,1,x);    Hf\_1 = figure('Units','inches','position',[1,1,6,4],'paperunits','inches','paperposition',[0,0,6,4]);  set(Hf\_1,'NumberTitle','off','Name','P7.34b');  subplot(2,1,1);  Hs\_1 = stem(n(101:201),x(101:201),'g','filled');  title('Input sequence x(n)','fontsize',10);  set(Hs\_1,'markersize',3);  ylabel('Amplitude','fontsize',10);  subplot(2,1,2);  Hs\_2 = stem(n(101:201),y(101:201),'m','filled');  title('Output sequence y(n)','fontsize',10); set(Hs\_2,'markersize',3);  xlabel('n','fontsize',10); ylabel('Amplitude','fontsize',10); |