```
% wavelet denoising

 % load subband coefficients
 D = 6;                                % number of levels
 thresh = 0.1;                         % coefficient threshold
 str=load('db1.mat'); varnames=fieldnames(str); db=getfield(str,varnames{1});
 h = db(:)/sum(db)*sqrt(2);            % normalize
 g = flipud(h.*((-1).^[1:length(h)]).');
 lpfa = flipud(h);                     % analysis lpf
 hpfa = flipud(g);                     % analysis hpf
 lpfs = h;                             % synthesis lpf
 hpfs = g;                             % synthesis hpf

 % create noisy step waveform
 %Nx = 512; Nb = 64; x = zeros(Nx-Nb+1,1); x(1:Nb:end) = rand(Nx/Nb,1);
 %x = conv(x,ones(Nb,1));
 %y = x + 0.03*randn(size(x));
 load steps.mat;  Nx = length(x);      % try D=6, thresh = 0.1, db1.mat

 % create noisy chirp waveform
 %Nx = 512; x = sin([0:Nx-1].^2/5000).'; %x = [x;flipud(x)];
 %y = x + 0.1*randn(size(x));
 %load chirp.mat;  Nx = length(x);     % try D=4, thresh = 0.3, db10.mat

 % wavelet filterbank analysis via circular convolution
 a_old = x;
 d = [];
 for i=1:D,
   tmp = real(ifft(fft(a_old).*fft(lpfa,length(a_old))));
   a = wshift('1',tmp(2:2:end),length(h)/2-1);          % scalefxn coefs
   tmp = real(ifft(fft(a_old).*fft(hpfa,length(a_old))));
   d = [wshift('1',tmp(2:2:end),length(h)/2-1);d];      % wavelet coefs
   a_old = a;
 end;
 w_clean = [a;d];                                       % total coefs

 % wavelet filterbank analysis via circular convolution
 a_old = y;
 d = [];
 for i=1:D,
   tmp = real(ifft(fft(a_old).*fft(lpfa,length(a_old))));
   a = wshift('1',tmp(2:2:end),length(h)/2-1);          % scalefxn coefs
   tmp = real(ifft(fft(a_old).*fft(hpfa,length(a_old))));
   d = [wshift('1',tmp(2:2:end),length(h)/2-1);d];      % wavelet coefs
   a_old = a;
 end;
 w = [a;d];                                             % total coefs

 % wavelet domain thresholding
 indx_small = find(abs(w)<thresh);
 ww = w; ww(indx_small) = zeros(length(indx_small),1);

 % wavelet filterbank synthesis via circular convolution
 a_new = ww(1:Nx/2^D);
 for i=D:-1:1,
   a_up = zeros(2*length(a_new),1); a_up(1:2:end) = a_new;
   d_up = zeros(2*length(a_new),1); d_up(1:2:end) = ww(Nx/2^i+1:Nx/2^(i-1));
   a_new = real(ifft(fft(a_up).*fft(lpfs,length(a_up)))) +...
           real(ifft(fft(d_up).*fft(hpfs,length(d_up))));
 end;
 x_hat = a_new;

 figure(1);
 subplot(321);
   plot(w); title('noisy transform coefs');
```

```
 subplot(323);
   plot(ww); title('thresholded transform coefs');
 subplot(325);
   plot(w_clean); title('noiseless transform coefs');
   %axis([1,Nx,-0.5,0.5]);
 subplot(322);
   plot(y); title('noisy signal');
 subplot(324);
   plot(x_hat); title('denoised signal');
 subplot(326);
   plot(x_hat-x); title('reconstruction error');
```