

Tuneup #7 solution

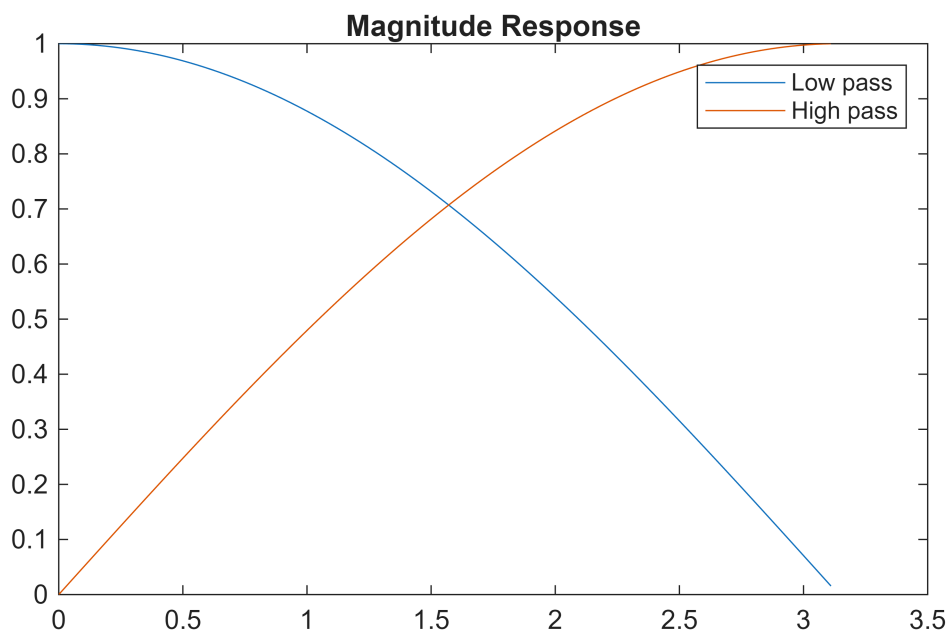
Filter bank using two-sample average and first order difference

```
% filter coefficients
LA = [0.5, 0.5]; % Lowpass Analysis
HA = [0.5, -0.5]; % Highpass Analysis
LS = [1, 1]; % Lowpass Synthesis
HS = [-1, 1]; % Highpass Synthesis
```

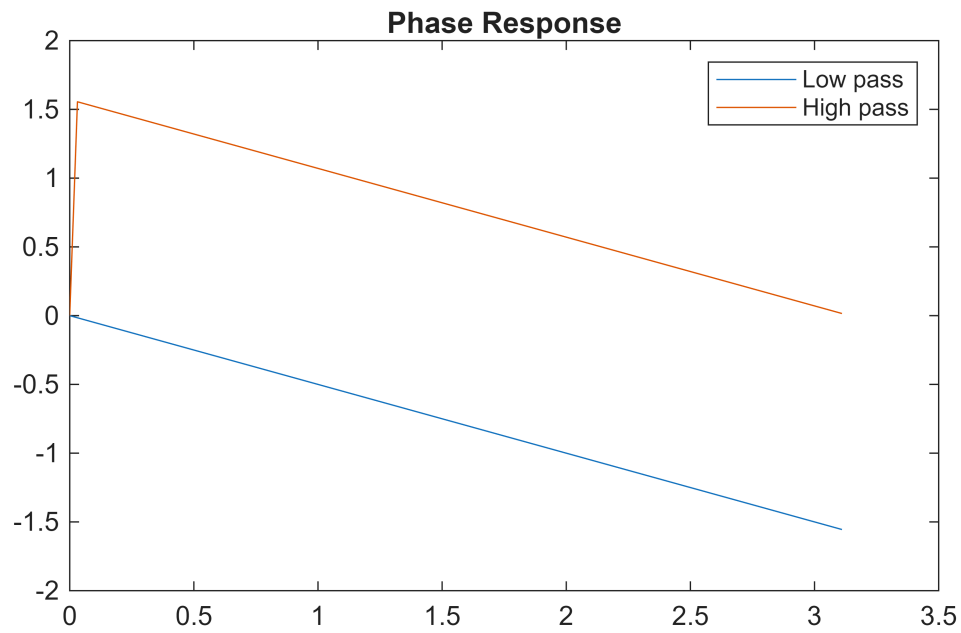
Part (a)

Use freqz to plot the frequency response of the averaging filter and the difference filter

```
Nplot = 101;
[HLA, w] = freqz(LA,1,Nplot);
[HHA, ~] = freqz(HA,1,Nplot);
figure; plot(w,abs(HLA));
hold on; plot(w,abs(HHA))
legend('Low pass', 'High pass')
title('Magnitude Response')
```

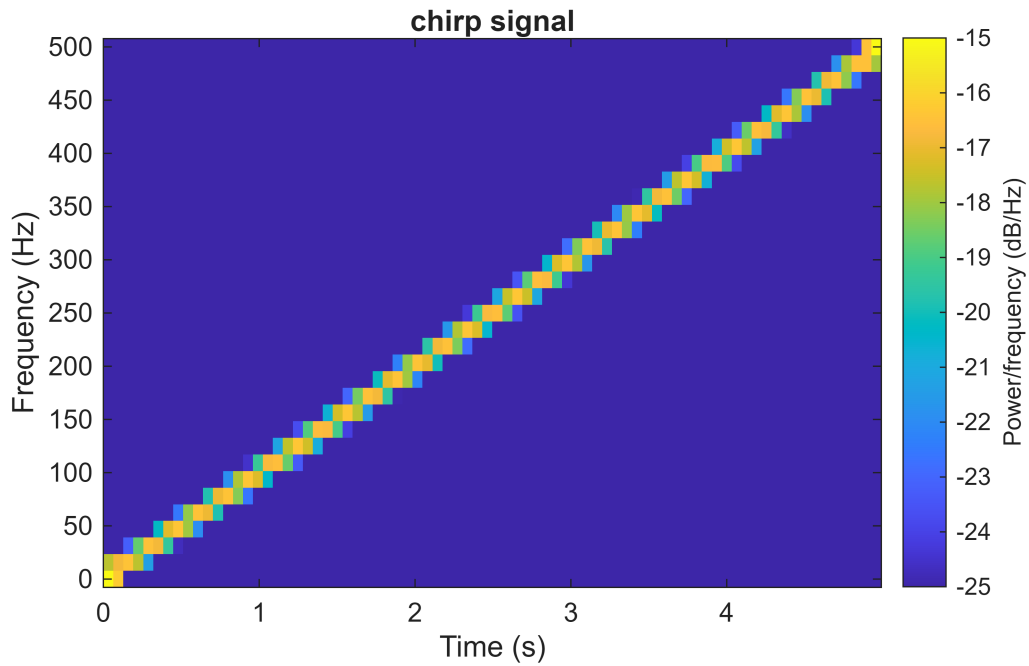


```
figure; plot(w,angle(HLA));
hold on; plot(w,angle(HHA))
legend('Low pass', 'High pass')
title('Phase Response')
```



Part (b)

```
% Input: chirp signal
fs = 1000;
t = linspace(0,5,5*fs);
f0 = 0;
fstep = 100;
x = cos(2*pi*f0*t + pi*fstep*t.*t);
figure; spectrogram(x,64,0,64,fs,'yaxis'); caxis([-25,-15]);
title('chirp signal');
```



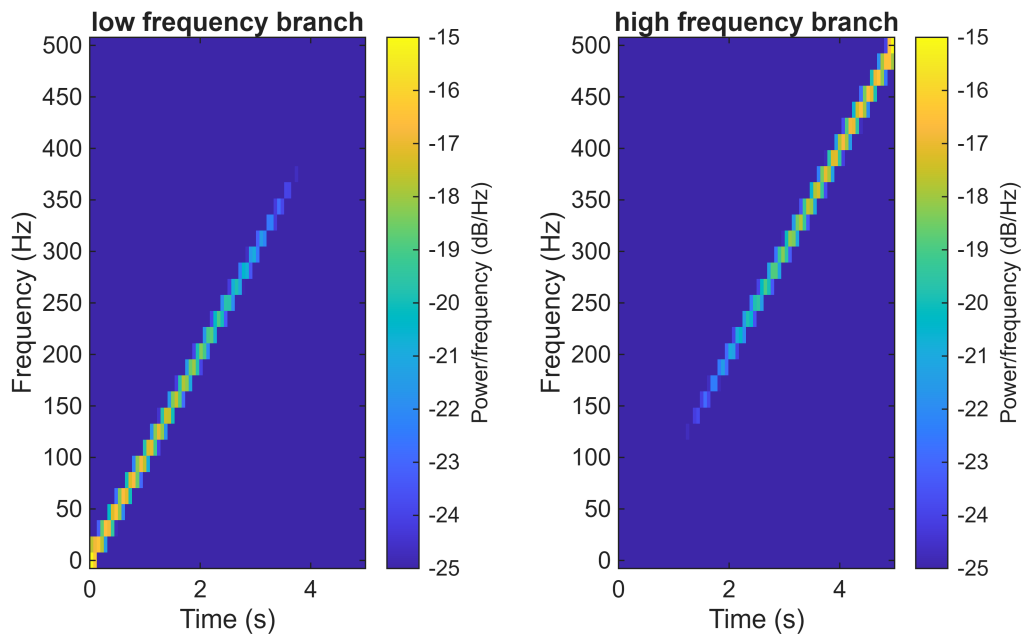
The first filter removes the high frequency component of the signal. The second filter removes the low frequency component.

```
% Low-pass analysis filter
```

```
xL0 = conv(x,LA);  
figure; tiledlayout(1,2); nexttile;  
spectrogram(xL0,64,0,64,fs,'yaxis'); caxis([-25,-15]);  
title('low frequency branch');
```

```
% High-pass analysis filter
```

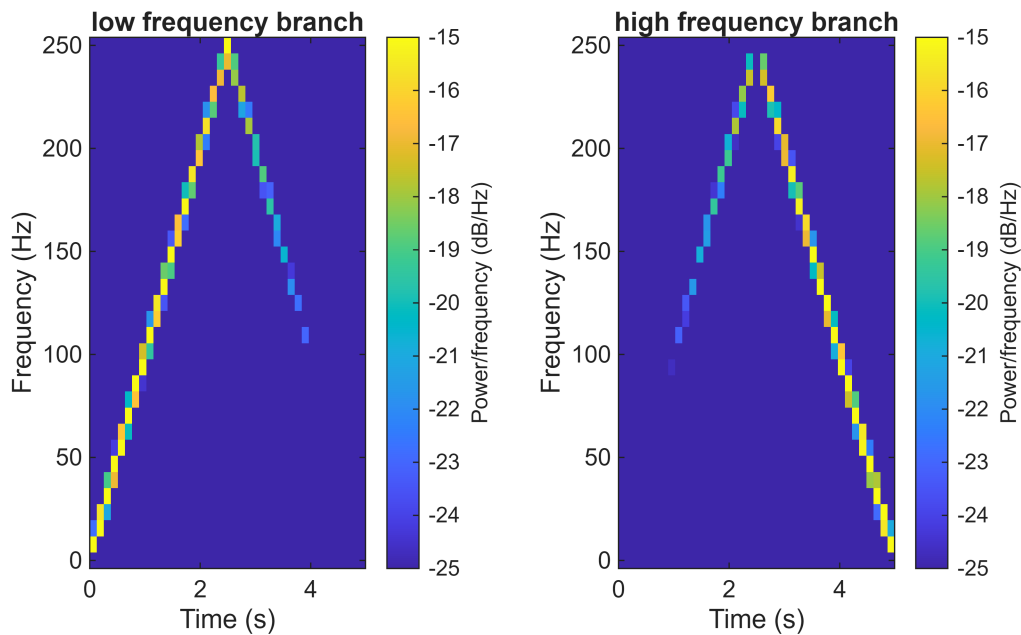
```
xH0 = conv(x,HA);  
nexttile; spectrogram(xH0,64,0,64,fs,'yaxis'); caxis([-25,-15]);  
title('high frequency branch');
```



Downsampling warps the spectrum, causing components that were originally above $\pi/2$ to alias.

```
% Downsample
xL1 = downsample(xL0,2);
figure; tiledlayout(1,2); nexttile;
spectrogram(xL1,64,0,64,fs/2,'yaxis'); caxis([-25,-15]);
title('low frequency branch');

xH1 = downsample(xH0,2);
nexttile; spectrogram(xH1,64,0,64,fs/2,'yaxis'); caxis([-25,-15]);
title('high frequency branch');
```

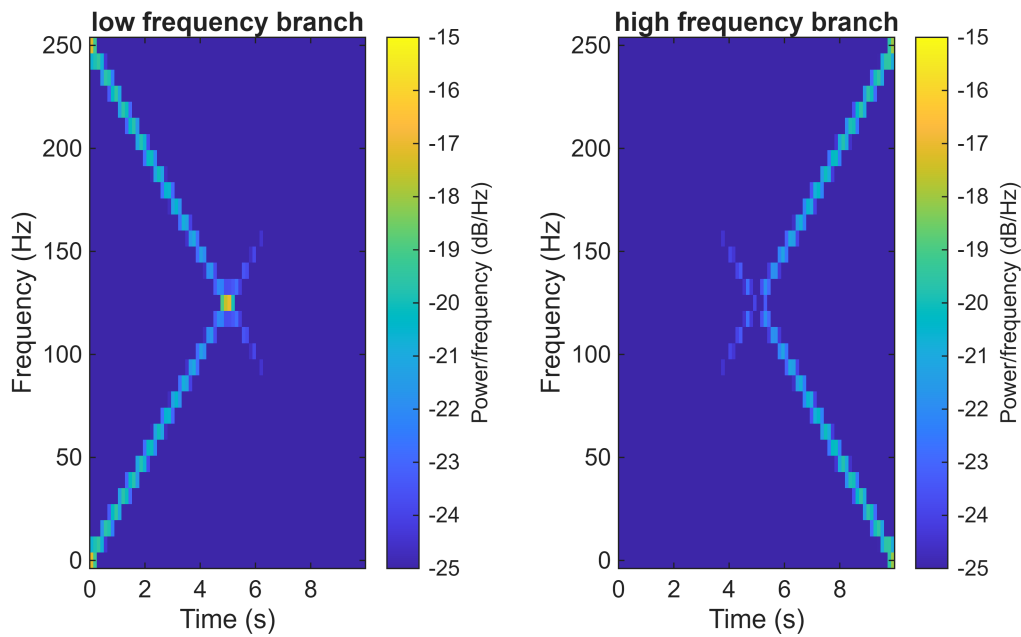


Upsampling warps the frequency spectrum and creates a copy of it shifted by π . The spectrogram appears mirrored because the spectrum is conjugate symmetric, so the magnitude of the component that appears between $\pi/2$ and π is the flipped version of the magnitude component between 0 and $\pi/2$.

The color corresponding to the signal on the spectrogram becomes a lighter shade of blue instead of yellow, indicating the intensity has decreased.

```
% Upsample
xL2 = upsample(xL1,2);
figure; tiledlayout(1,2); nexttile;
spectrogram(xL2,64,0,64,fs/2,'yaxis'); caxis([-25,-15]);
title('low frequency branch');

xH2 = upsample(xH1,2);
nexttile; spectrogram(xH2,64,0,64,fs/2,'yaxis'); caxis([-25,-15]);
title('high frequency branch');
```



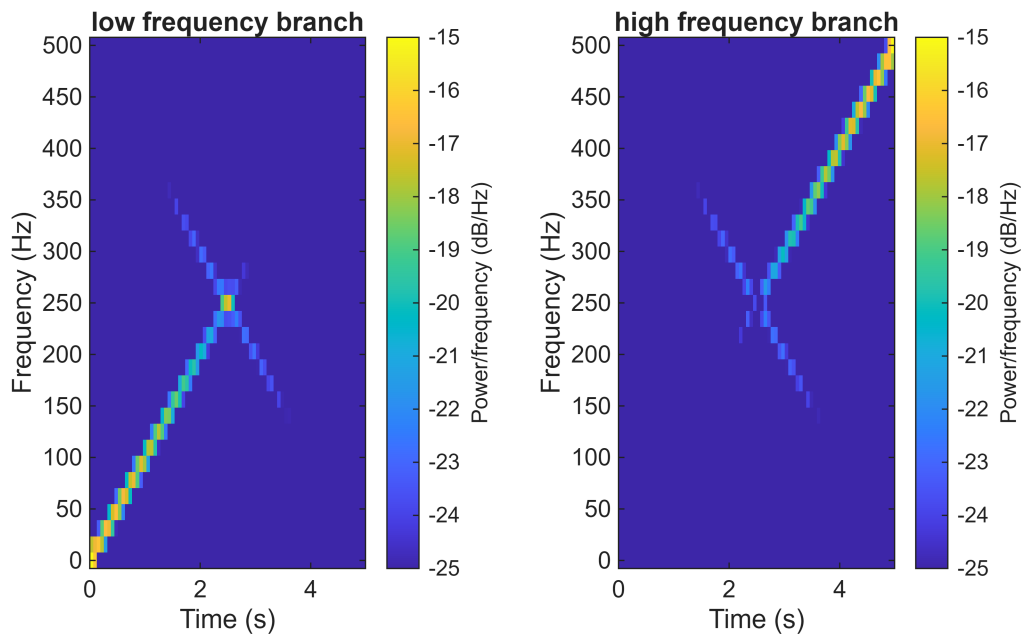
The synthesis filters attenuate the magnitude of the aliased components and apply a gain to the overall intensity to match the signal prior to downsampling and upsampling.

```
% Low-pass synthesis filter
```

```
xL3 = conv(xL2,LS);
figure; tiledlayout(1,2); nexttile;
spectrogram(xL3,64,0,64,fs,'yaxis'); caxis([-25,-15]);
title('low frequency branch');
```

```
% High-pass synthesis filter
```

```
xH3 = conv(xH2,HS);
nexttile; spectrogram(xH3,64,0,64,fs,'yaxis'); caxis([-25,-15]);
title('high frequency branch');
```



Part (c)

The magnitude response is shown corresponding to a branch of a filter bank where the analysis transform is

Lowpass -> downsample -> Highpass -> downsample

While the lowpass filter had a passband from approximately 0 to $\pi/2$, this frequency response shows a passband from approximately $\pi/2$ to $\pi/4$.

```
Nplot = 101;
[HLA, w] = freqz(LA,1,Nplot);
[HHA, w] = freqz(HA,1,Nplot);
[HHS, w] = freqz(HS,1,Nplot);
HLA2 = downsample([HLA; flipud(conj(HLA))],2);
HLS2 = downsample([HLS; flipud(conj(HLS))],2);
HHA2 = downsample([HHA; flipud(conj(HHA))],2);
HHS2 = downsample([HHS; flipud(conj(HHS))],2);

figure;
low_high = 0.25*HLA.*HHA2.*HHS2.*HLS;
hold on; plot(w,abs(low_high), linewidth=3)

xlim([0,pi]);
set(gca,'XTick',[0,pi/4,pi/2,pi])
set(gca,'XTickLabels',['0', '\pi/4', '\pi/2', '\pi'])

xlabel('Frequency \omega [radians per sample]')
ylabel('Magnitude response | H(e^{j\omega}) |')
```

