* 1. **Window Method**

**We have seen that in the design of FIR filters, Gibbs oscillations are produced in the passband and stopband, which are not desirable features of the FIR filter. To solve this problem, window method (Fourier transform design with window method) is developed.**

**The Gibbs oscillations mainly result from the abrupt truncation of the infinite-length coefficient sequence of impulse response of the FIR filter. A window function is a symmetrical function which can gradually weight the designed FIR coefficients down to zeros at both ends of for the range. Applying the window sequence to the filter coefficients gives**

**where, designates the window function. Commonly used window functions in FIR filters are:**

|  |  |
| --- | --- |
| 1. Rectangular Window: | (7.15) |
| 2. Triangular (Bartlett) Window: | **(7.16)** |
| 3. Hanning Window: | **(7.16)** |
| 4. Hamming Window: | **(7.16)** |
| 5. Blackman Window: | **(7.16)** |

**Example 7.5**

1. **Calculate the filter coefficients for a 3-tap FIR lowpass filter with a cutoff frequency of 800 Hz and a sampling rate of 8000 Hz using the Hamming window function.**
2. **Determine the transfer function and difference equation of the designed FIR system.**
3. **Compute and plot the frequency response.**

**Solution**

**Part (a): We first determine the normalized cutoff frequency**

**radians**

**In this case, therefore, using table 7.1**

**Therefore,**

**(using symmetry)**

**The windowed impulse response is calculated as**

**Delaying by samples, we get, , and.**

**Part (b): Therefore, the transfer function in this case is**

**The difference equation is**

**Part (c): The frequency response of the filter is**

**It can be written as**

**Thus, the magnitude frequency response is**

**And the phase response is**

**A MATLAB program to plot the frequency response is given in the following.**

**%**

**% Example 7.5**

**%**

**fs = 8000;**

**B = [0.01497 0.2 0.01497];**

**A = [1];**

**[h w] = freqz(B, A, 1024);**

**f = w\*fs/(2\*pi);**

**phi = 180\*unwrap(angle(h))/pi;**

**figure**

**subplot(2,1,1), plot(f, 20\*log10(abs(h))); grid on;**

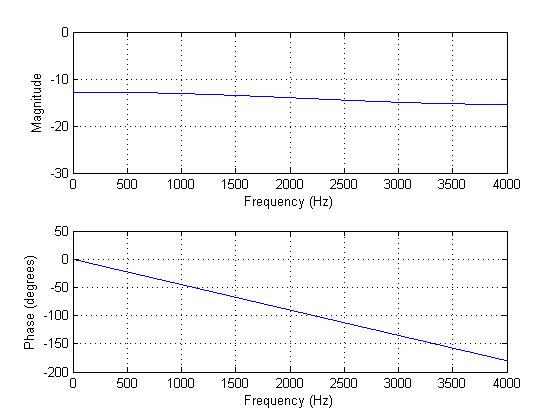
**axis([0 fs/2 -30 0]);**

**xlabel('Frequency (Hz)'); ylabel('Magnitude');**

**subplot(2,1,2), plot(f, phi); grid on;**

**axis([0 fs/2 -200 50]);**

**xlabel('Frequency (Hz)'); ylabel('Phase (degrees)');**

****

**Figure 7.4: Frequency response in Example 7.2**

**Example 7.6**

1. **Design a 5-tap FIR bandreject filter with a lower cutoff frequency of 2000 Hz, an upper cutoff frequency of 2400 Hz and a sampling rate of 8000 Hz using the Hamming window method.**
2. **Determine the transfer function.**

**Solution**

**Part (a): We first determine the normalized cutoff frequencies**

**radians**

**radians**

**In this case, therefore, using table 7.1**

**Therefore,**

**(using symmetry)**

**(using symmetry)**

**Applying the Hamming window function, we have**

**(using symmetry)**

**(using symmetry)**

**The windowed impulse response is calculated as**

**Delaying by samples, we get, ,, and.**

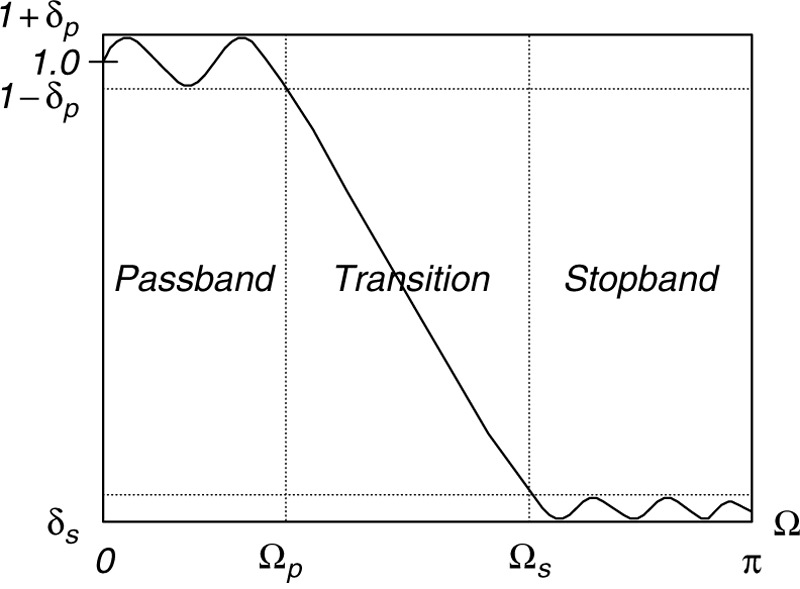
**Part (b): Therefore, the transfer function in this case is**

**FIR Filter Length Estimation using Window Functions**

**Given the required passband ripples specification and stopband attenuation, the appropriate window length can be estimated based on the performances of the window functions. For illustrative purpose, we use the lowpass filter frequency domain specification (the same can be extended to other types of filter specifications).**

**The normalized transition band frequency is defined as**

**Based on this, the FIR filter lengths for various window functions are given in Table 7.7 below. It can be noted that the cutoff frequency is determined by**



**The passband ripple is defined as**

**while the stopband attenuation is defined as**

**Table 7.7: Filter length estimation using window ()**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Window Type | Window  Function, | Window Length | Passband Ripple (dB) | Stopband Attenuation (dB) |
| Rectangular |  |  | **0.7416** | **21** |
| Hanning |  |  | **0.0546** | **44** |
| Hamming |  |  | **0.0194** | **53** |
| Blackman |  |  | **0.0017** | **74** |

**Example 7.10**

**Design a bandpass FIR filter with the following specifications:**

**Lower Stopband = 0 - 500 Hz**

**Passband = 1600 - 2300 Hz**

**Upper Stopband = 3500 - 4000 Hz**

**Stopband Attenuation = 50 dB**

**Passband ripple = 0.05dB**

**Sampling rate = 8000 Hz**

**Determine the FIR filter length and the cutoff frequency to be used in the design equation.**

**Solution**

**We first determine the normalized transition band**

**The filter lengths based on above transition bands (for Hamming window) are**

**The nearest higher odd is chosen as for the Hamming window**

**The lower and higher cutoff frequencies for the bandpass filter will be**

**The normalized lower and higher cutoff frequencies for the bandpass filter will be**

**In this case will be the number of taps for the bandpass filter.**

**A MATLAB program to plot the frequency response is given in the following.**

**%**

**% Example 7.10**

**%**

**Ntap = 25; % number of filter taps**

**Ftype = 3; % lowpass filter**

**WnL = 0.2625\*pi; % lower cutoff frequency**

**WnH = 0.725\*pi; % upper cutoff frequency**

**Wtype = 4; % Hamming window type**

**fs = 8000; % sampling rate**

**% use firwd to compute B coefficients**

**B = firwd(Ntap, Ftype, WnL, WnH, Wtype);**

**A = 1;**

**omega = 0:0.01:pi;**

**fHz = omega\*fs/(2\*pi);**

**% frequency response of the fir filter**

**[hz, w] = freqz(B, A, omega);**

**magnitude = 20\*log10(abs(hz));**

**phase = 180\*unwrap(angle(hz))/pi;**

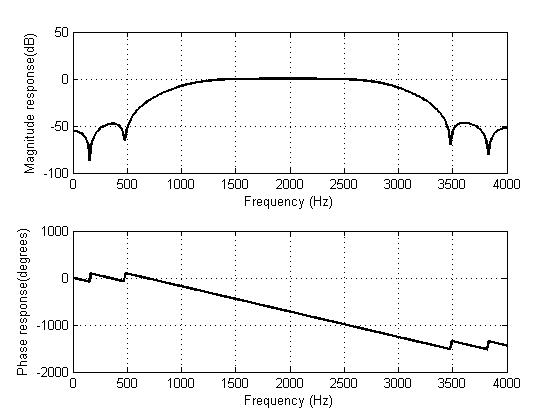
**figure,**

**subplot(2,1,1); plot(fHz, magnitude,'k','LineWidth',2); grid on;**

**xlabel('Frequency (Hz)'); ylabel('Magnitude response(dB)');**

**subplot(2,1,2); plot(fHz, phase,'k','LineWidth',2); grid on;**

**xlabel('Frequency (Hz)'); ylabel('Phase response(degrees)');**

****

**Figure 7.18: Frequency response in Example 7.2**