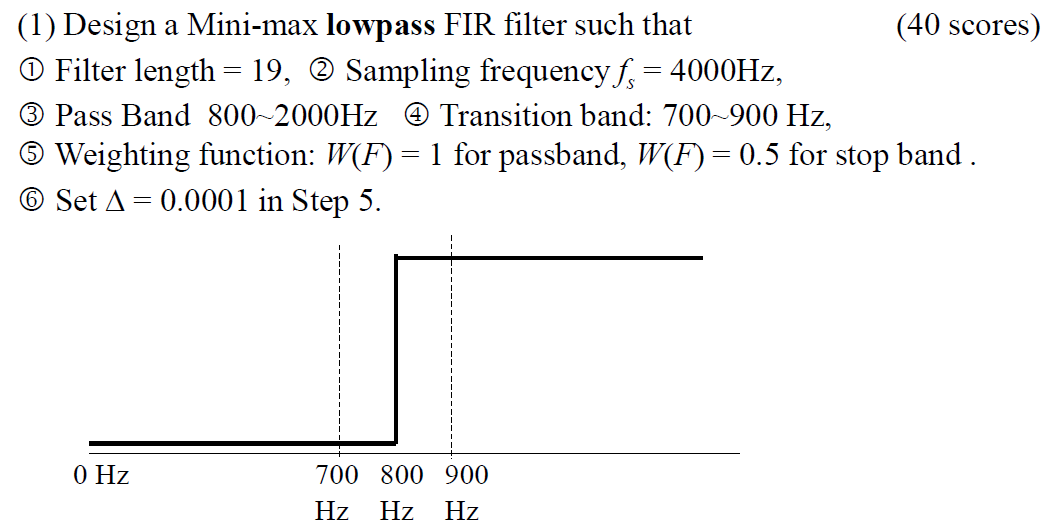
**MinMax Filter Implementations**

**- Frequency Response, Impulse Response, Maximal Errors**



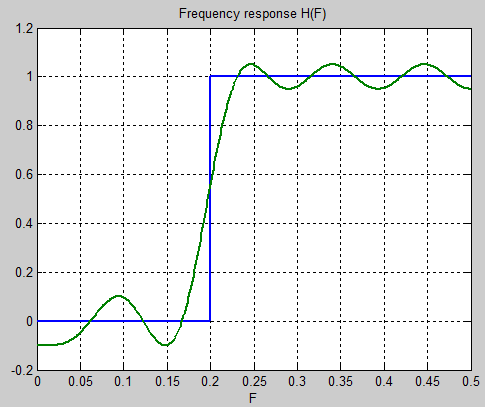
**(sol)**

**(b) the frequency response**

1. Given that **Sampling freq.= *4000Hz*** , so I map the **freq. domain** to normalized

***F*-domain** , which has 2000 intervals, i.e.***F = 0: 1/fs : 0.5***

1. The design process converges after ***6* iterations** , and the final Maximal error is ***0.050168***
2. The **intermediate** ***R(F)***, **impulse response *h[n]*** and **frequency response *H(f)*** of designed FIR filter are showed below.

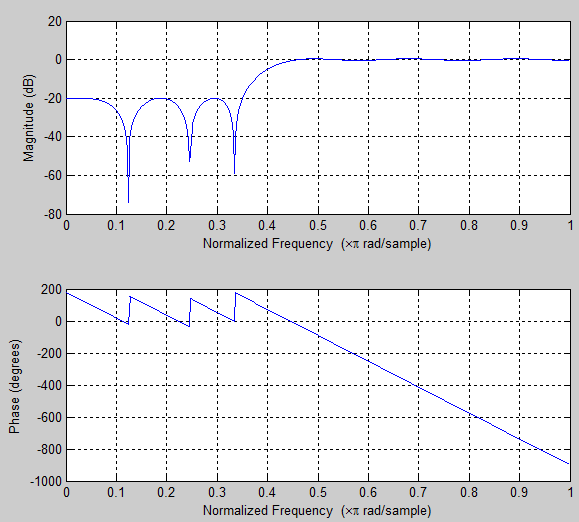


**Figure *1*. Mini-Max FIR HPF** & **Ideal HPF**

**Figure *1*.** shows the frequency response of our FIR filter.

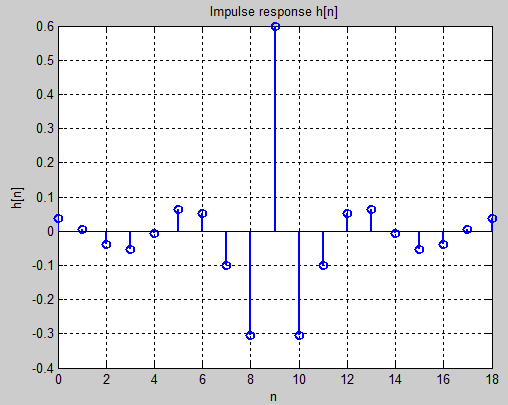
Since I’ve assigned different weighting function to pass-band and stop-band , we could see that the amplitude ripples are different in these two bands.

For the Weight value of pass-band is larger , we could find that the amplitude ripple of the pass-band is smaller , and vice versa in the case of stop-band.



***Figure 2.*** **The Amplitude & Frequency Response of h[n]**

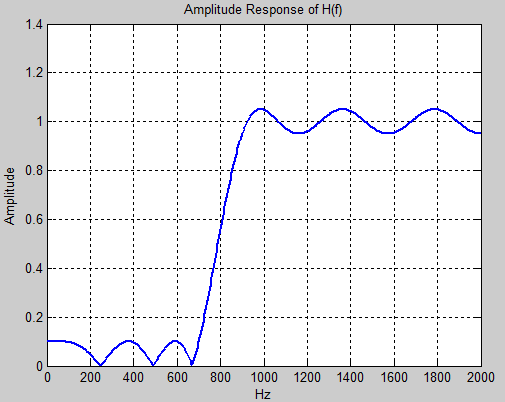
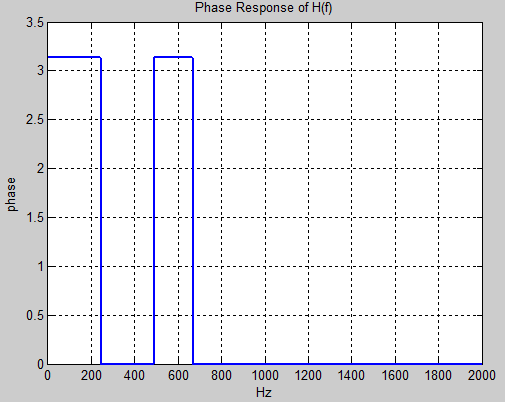
**(c) the impulse response *h*[*n*]**



**Figure *3*. Impulse Response of Mini-Max FIR HPF**

**Figure *3*** is the Impulse response of the Mini-max FIR filter.

Given that h[n] has 19 points as we expected and the whole response is even-symmetric to the middle point n = 9.

**Figure *4.*Amplitude Response & Phase Response of Mini-Max FIR HPF**

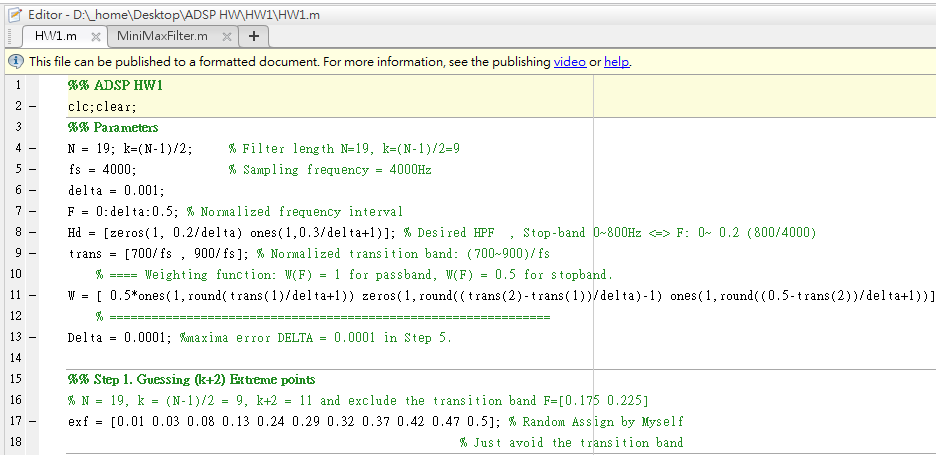
**(d) the maximal error for each iteration.**

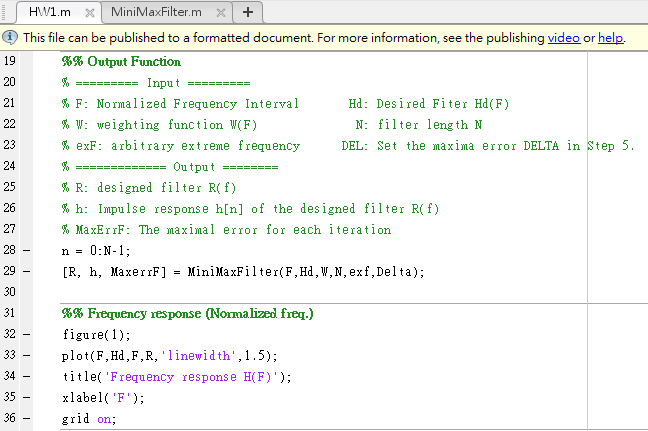
|  |  |
| --- | --- |
| Iteration | Maximal error |
| **1** | **0.191856** |
| **2** | **0.123296** |
| **3** | **0.069735** |
| **4** | **0.051108** |
| **5** | **0.050177** |
| **6** | **0.050168** |

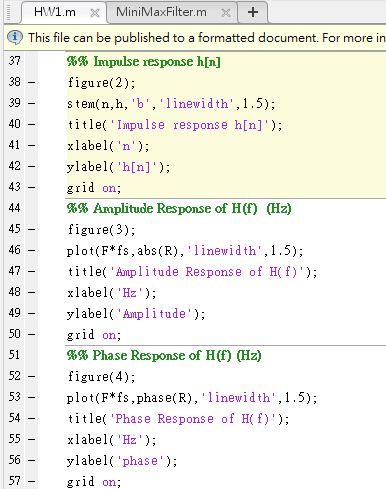
**Table 1. The Max. Error of each Iteration**

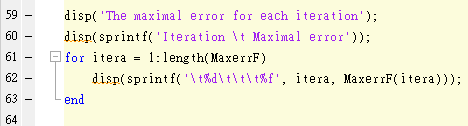
The final max. error is the value of the ***6th*** iteration , for the reason that the difference of max. error between ***5th*** and ***6th*** is ***|0.050168-0.050177| < delta = 0.0001*** ,

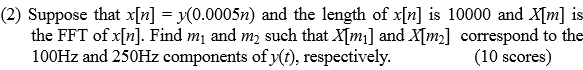
so we conclude that the program stops at the ***6th*** iteration.

1. **the Matlab program**

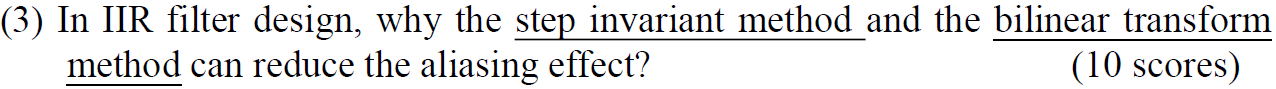








***(ANS)***

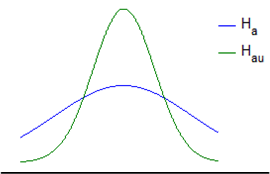


***(ANS)***

1. ***Step Invariant method***

Do sampling on the impulse response of step function, just like:



So that, the high-freq. term will be compressed that we could avoid the aliasing effect

1. ***Bilinear transform method***





*fs* = 1/*t* (sampling frequency)

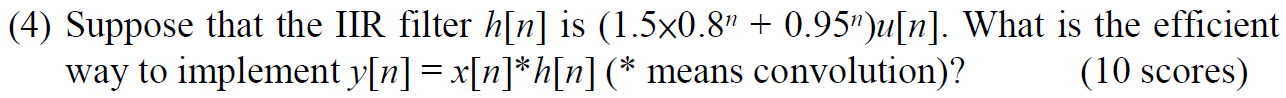
*fold*  (, ) *fnew*  (*fs*/2, *fs*/2)

By this method , we transform the original frequency-domain to a new one.(and it’s also a better one)

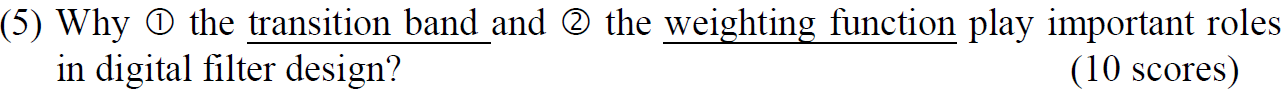


*fnew/fs*

By the figure above, we could perfectly confirm that the normalized frequency term will never larger than ***0.5*** , so that we could conclude the aliasing effect won’t happen.



***(ANS)***

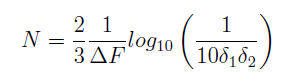


***(ANS)***

 ：

From ***1979 F.Mintzer and L.Bede Paper***,

The width of the transition band ΔF and the filter length N have the relation equation that: is the passband ripple (error)

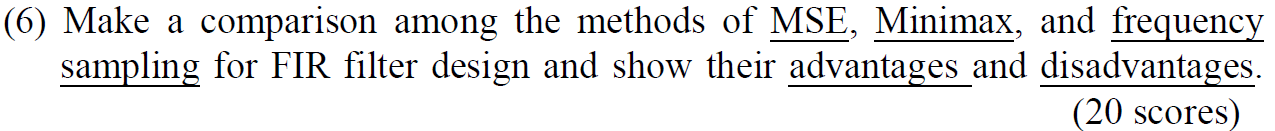
  is the stopband ripple (error)

When ***N***(filter length) is fixed ,once we increase the transition band ***ΔF*** , we could increase the accuracy of filter in passband and stopband.



The Weighting function could control the accuracy between passband and stopband, which means that we could make the accuracy between passband and stopband different.

For example, if we setting that the weight is larger in stopband, smaller in passband, then we could get a smaller ripple in the stopband, which means a higher accuracy, so that we view stopband accuracy is more important , and vice versa.



***(ANS)***

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Feature** | **Advantage(s)** | **Drawbacks** |
| **MSE** | Minimizing the mean square error | 1.Flexible designing  2. could also be optimized by weighting function & transition band | Large error near the transition band |
| **Mini-Max** | Minimizing the max. error | Min. max. error | 1. Transition band is necessary 2. Design duration is slow because of recursion 3. Impulse response h[n] must be odd or even |
| **Frequency**  **Sampling** | The discrete-time Fourier transform of the desired filter | Simplest for designing | 1. Hard to optimize 2. *Gibb’s Phenomenon* in time-domain |