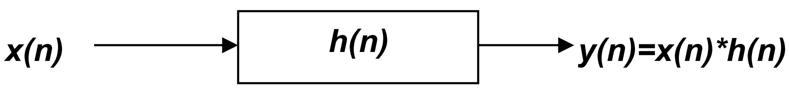
12

Desain Filter FIR dengan Metode Sampling Frekuensi

Pada Bab 12 sampai Bab 14, kita akan mempelajari beberapa metode untuk mendesain filter FIR. Filter FIR pada umumnya dinyatakan dalam bentuk *impulse response* seperti pada Gambar 12.1, sehingga hasil akhir dari proses desain suatu filter FIR adalah nilai dari *h(n)*.



Gambar 12.1
Filter FIR dalam bentuk impulse respons

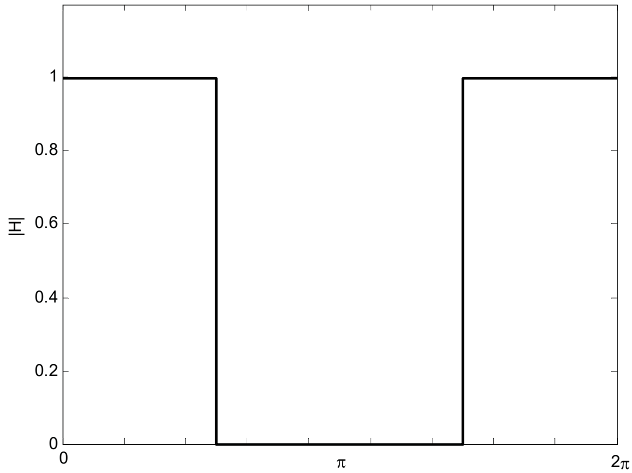
Filter FIR dapat didesain dengan banyak metode tetapi pada buku ini hanya akan dibahas tiga metode yaitu metode Sampling Frekuensi (Bab 12), metode Window (Bab 13), dan metode Optimal (Bab 14). Metode yang diajarkan dalam bab-bab ini bertujuan untuk memberikan pengetahuan dasar tentang cara menghitung *impulse response*, *h(n)*, dari filter FIR secara manual. Dalam aplikasi praktisnya *h(n)* tidak akan selalu dihitung secara manual. Desain filter FIR akan dilakukan dengan menggunakan program komputer karena nilai *h(n)* yang harus dihitung akan sangat banyak. Ada banyak program komputer yang dapat digunakan untuk menghitung *h(n)* dari FIR. Pada modul-modul ini kita akan mempelajari juga cara menggunakan Matlab untuk mendesain filter FIR.

Setelah mempelajari materi pada bab ini maka mahasiswa akan dapat mendesain filter FIR dengan metode sampling frekuensi, baik secara perhitungan manual maupun dengan menggunakan perangkat lunak pendukung.

12.1 Metode Sampling Frekuensi

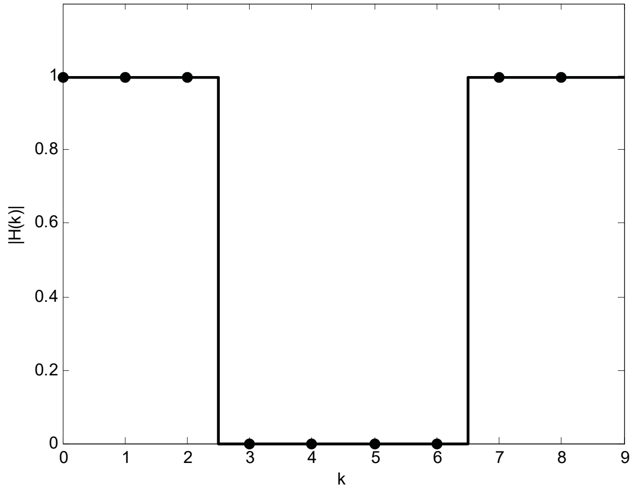
Metode ini adalah metode yang sederhana tetapi menghasilkan respon frekuensi yang tidak terlalu sempurna karena kita tidak dapat dengan mudah menentukan *ripple* baik pada *passband* maupun *stopband*. Metode ini menggunakan langkah-langkah yang akan dijelaskan dengan menggunakan contoh berikut ini dimana kita akan merancang LPF filter dengan orde, $N = 8$, dan $\Omega_c = 5/9\ \pi$. Langkah-langkah tersebut adalah sebagai berikut:

1. Gambarlah respon frekuensi ideal dari filter yang akan dibuat dari 0 sampai 2π. Gambar tersebut dimulai dari |H| = 1 sampai titik $5/9\ \pi$ kemudian turun menjadi |H|=0 sampai π. Gambar sesudah $\Omega = \pi$. adalah pencerminan dari respon frekuensi dari $\Omega = 0$ sampai $\Omega = \pi$.



Gambar 12.2
Respons frekuensi ideal dari LPF dengan $\Omega_c = 5/9\ \pi$.

2. Tandailah sebanyak *N+1* titik sample (dengan interval frekuensi yang sama) pada gambar respons frekuensi tersebut dimana *N* adalah orde dari filter yang akan dibuat. Karena kita akan merancang filter dengan orde, N=8, maka kita menandai 9 titik seperti pada Gambar 12.3 di bawah ini. Catatlah nilai dari sample-sample tersebut. Titik-titik sample ini disebut *H(k)* untuk *k = 0 ... N-1*.



Gambar 12.3
Titik sample pada respons frekuensi LPF dengan $\Omega_c = 5/9\ \pi$.

Nilai *H(k)* adalah sebagai berikut:

$H(0) = 1$
$H(1) = 1$
$H(2) = 1$
$H(3) = 0$
$H(4) = 0$
$H(5) = 0$
$H(6) = 0$
$H(7) = 1$
$H(8) = 1$

3. Hitunglah *h(n)* dengan menggunakan

$$\boxed{h(n) = \frac{1}{2M+1}\left\{H(0) + 2\sum_{k=1}^{M} H(k)\cos\left(\frac{2\pi k(n-M)}{2M+1}\right)\right\}}$$

dimana $M = ½\ N$, jadi untuk contoh ini, $M = ½\ (8) = 4$

maka

$$h(n) = \frac{1}{9}\left\{1 + 2\sum_{k=1}^{4} H(k)\ \cos\left(\frac{2\pi k(n-4)}{9}\right)\right\}$$

sehingga

$$h(0) = \frac{1}{9}\left\{1 + 2\left(H(1)\ \cos\left(\frac{2\pi 1(0-4)}{9}\right) + H(2)\ \cos\left(\frac{2\pi 2(0-4)}{9}\right)\right)\right\}$$

$$h(0) = \frac{1}{9}\left\{1 + 2\left(\cos\left(\frac{-8\pi}{9}\right) + \cos\left(\frac{-16\pi}{9}\right)\right)\right\} = 0.0725$$

$$h(1) = \frac{1}{9}\left\{1 + 2\left(\cos\left(\frac{-6\pi}{9}\right) + \cos\left(\frac{-12\pi}{9}\right)\right)\right\} = -0.1111$$

$$h(2) = \frac{1}{9}\left\{1 + 2\left(\cos\left(\frac{-4\pi}{9}\right) + \cos\left(\frac{-8\pi}{9}\right)\right)\right\} = -0.0591$$

$$h(3) = \frac{1}{9}\left\{1 + 2\left(\cos\left(\frac{-2\pi}{9}\right) + \cos\left(\frac{-4\pi}{9}\right)\right)\right\} = 0.3199$$

$$h(4) = \frac{1}{9}\{1 + 2(\cos(0) + \cos(0))\} = 0.5556$$

$$h(5) = h(3) = 0.3199$$

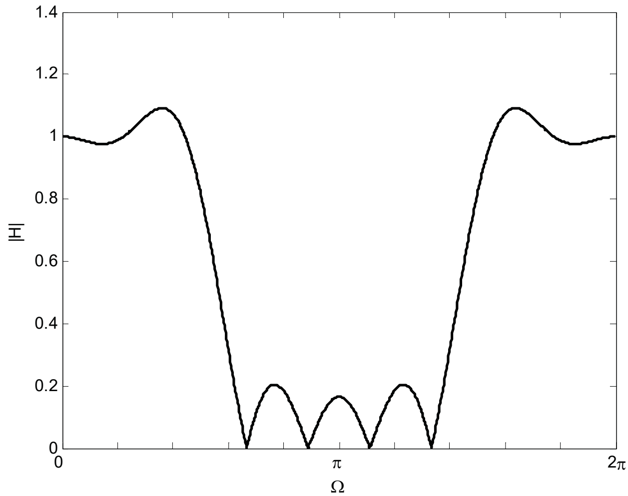
$$h(6) = h(2) = -0.0591$$

$$h(7) = h(1) = -0.1111$$

$$h(8) = h(0) = 0.0725$$

Dengan melakukan 3 langkah di atas maka kita telah selesai mendesain filter FIR orde 8 dengan $\Omega_c = 5/9\ \pi$.

Untuk memastikan respon frekuensi dari filter yang baru didesain, maka *h(n)* tersebut ditransformasi ke domain frekuensi dengan FFT dengan bantuan zero padding (lihat Bab 11) dan hasilnya ditunjukkan pada Gambar 12.4.



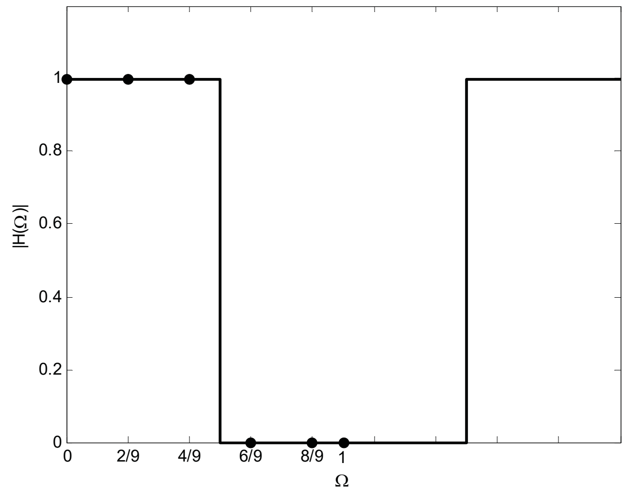
Gambar 12.4
Respons frekuensi dari LPF dengan $\Omega_c = 5/9\ \pi$.

Gambar 12.4 menunjukkan bahwa filter hasil desain ini memiliki respon frekuensi yang tidak persis sama dengan respon frekuensi yang diharapkan pada Gambar 12.2. Terdapat *ripple* baik pada *passband* maupun pada *stopband*. Inilah kelemahan dari metode desain dengan sampling frekuensi. Kita tidak dapat dengan mudah mengatur bersarnya *ripple* dan lebarnya *transition band*. Metode desain ini memeliki ***kelebihan*** yaitu memungkinkan kita merancang filter dengan bentuk respon frekuensi yang sangat bervariasi.
Jika filter FIR yang akan kita rancang memiliki orde yang bernilai gasal maka digunakan rumus:

$$h(n) = \frac{1}{2M+1}\left\{H(0) + 2\sum_{k=1}^{M-\frac{1}{2}} H(k)\cos\left(\frac{2\pi k(n-M)}{2M+1}\right)\right\}$$

12.2 Metode Sampling Frekuensi pada Matlab

Matlab tidak memiliki fungsi untuk mendesain filter FIR dengan metode sampling frekuensi seperti pada langkah-langkah di atas. Fungsi pada Matlab yang paling mirip dengan metode di atas adalah fungsi 'fir2'. Jika kita menggunakan fungsi ini maka kita harus meletakkan titik sample pada frekuensi Ω = 0 dan Ω = 1 (ingat, untuk Matlab Ω = 1 ini setara dengan *fs/2*). Fungsi 'fir2' tidak mengharuskan titik-titik sample diletakkan pada interval frekuensi yang sama sehingga Gambar 12.3 harus digambarkan seperti pada Gambar 12.5.



Gambar 12.5
Titik sample pada respons frekuensi LPF dengan Ω_c = 5/9 π sesuai format Matlab.

Perintah Matlab adalah:

```
>> F=[0  2/9 4/9  6/9  8/9   1];
>> A=[1   1   1    0    0     0];
>> N=8;
>> B=fir2(N,F,A,boxcar(N+1))
```

F adalah frekuensi dari titik-titik sample, *A* adalah nilai dari titik-titik sample tersebut, dan *N* adalah orde dari filter yang dirancang. Output dari perintah tersebut adalah *B* sebagai berikut:

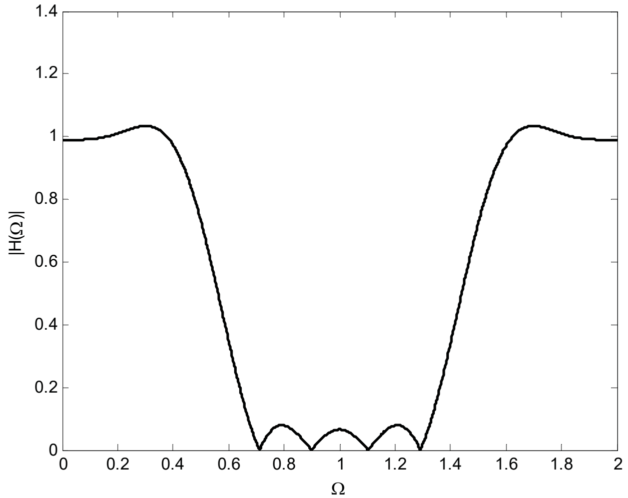
```
B =
  Columns 1 through 6
    0.0363   -0.0762   -0.0503    0.3072    0.5557    0.3072

  Columns 7 through 9
   -0.0503   -0.0762    0.0363
```

maka

$h(0)$ =	0.0363
$h(1)$ =	-0.0762
$h(2)$ =	-0.0503
$h(3)$ =	0.3072
$h(4)$ =	0.5557
$h(5)$ =	0.3072
$h(6)$ =	-0.0503
$h(7)$ =	-0.0762
$h(8)$ =	0.0363

Nilai ini berbeda dengan hasil perhitungan manual di atas karena Matlab menuntut kita untuk harus menyelipkan satu titik pada $\Omega = 1$ yang setara dengan $f_s/2$.

Untuk memastikan respon frekuensi dari filter yang baru didesain, maka $h(n)$ tersebut ditransformasi ke domain frekuensi dengan FFT seperti pada Gambar 12.6.



Gambar 12.6
Respons frekuensi LPF dengan $\Omega_c = 5/9\ \pi$ yang didesain dengan Matlab.

SOAL LATIHAN

1. Rancanglah suatu filter HPF orde *8* dengan $\Omega_c = 4/9\ \pi$ dengan metode sampling frekuensi.

2. Ulangi soal nomor 2 di atas dengan menggunakan Matlab.

3. Rancanglah suatu filter LPF orde *9* dengan $f_c = 2\ KHz$, $f_{stop} = 3\ KHz$ dan $f_s = 10\ kHz$ dengan metode sampling frekuensi.

4. Ulangi soal nomor 3 di atas dengan menggunakan Matlab.

5. Implementasikan filter tersebut dengan memodifikasi file `soal11_5.m`. Berikan beberapa sinyal sinusoidal dengan frekuensi *500 Hz, 1 kHz, 2 kHz, 4 kHz* dan *4 kHz* (gunakan $f_s = 10\ kHz$) sebagai inputnya. Amati apakah filter tersebut berfungsi sebagaimana mestinya.

6. Matlab memiliki fungsi 'filter' seperti terlihat pada file `soal11_6.m`. Ulangi soal nomor 5 di atas dengan menggunakan fungsi filter. Amati apakah filter tersebut berfungsi sebagaimana mestinya.

7. Implementasikan filter di atas dengan menggunakan Simulink untuk sinyal input seperti pada soal nomor 5.