



# Teknik Pengolahan Sinyal

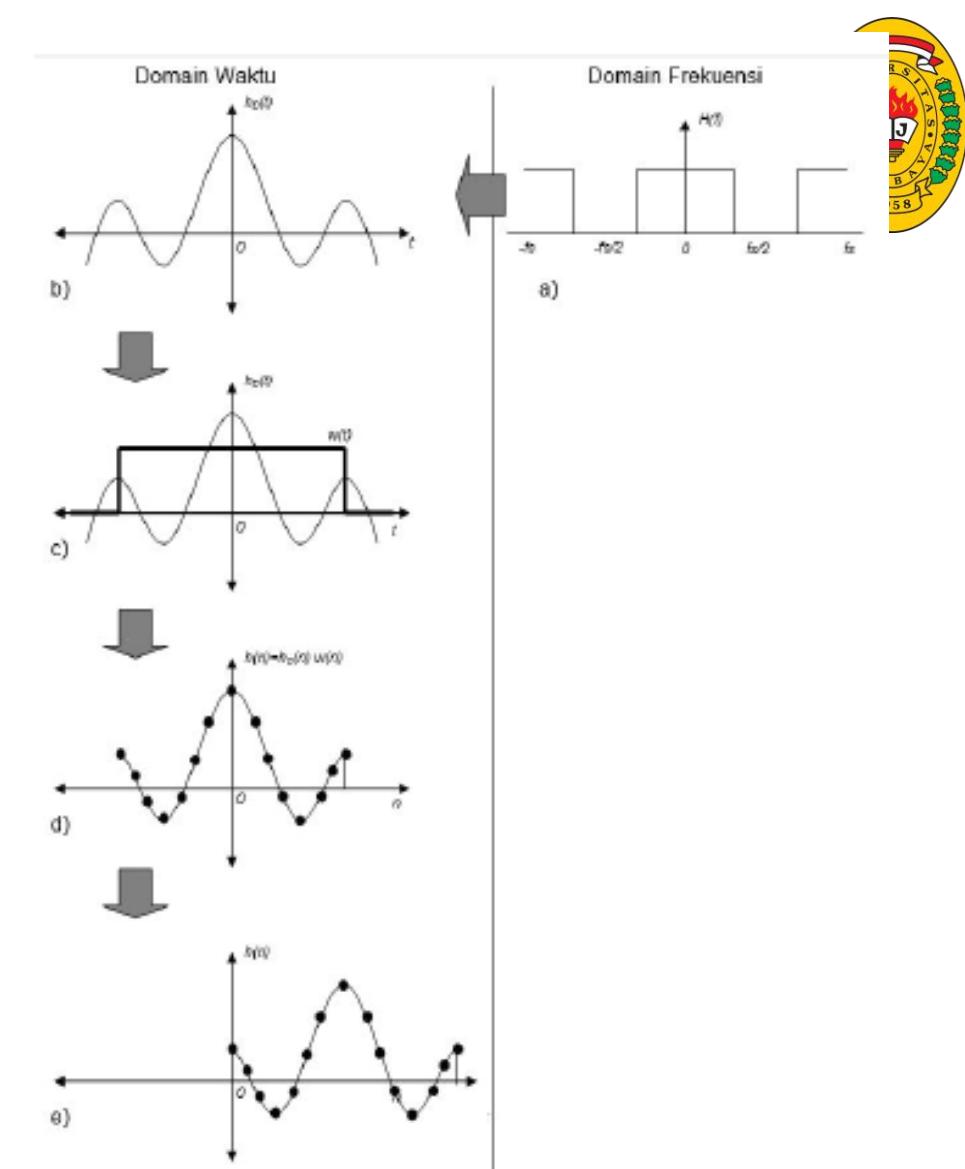
**Desain Filter FIR dengan Metode  
Window dan Optimal**

**Aqil Aqthobirrobbany, S.T., M.Eng.**

# Proses Desain dengan Metode Window

Prinsip dasar dari metode desain ini digambarkan pada Gambar 13.2. Proses desain dimulai dari menggambarkan dan menulis persamaan response frekuensi,  $H(f)$ , dari filter yang akan dibuat dari  $-f_s$  sampai  $f_s$ . Pada gambar tersebut contoh yang akan dibuat adalah filter LPF (Gambar 13.2a).

Persamaan ini kemudian ditransformasi dengan **Inverse Fourier Transform** untuk memperoleh **impulse response** ideal,  $h_D(t)$ . Impulse response ini memiliki panjang yang tak berhingga (Gambar 13.2b). Karena kita akan mendesain filter **FIR** maka kita harus mempunyai **impulse response** dengan panjang yang terbatas. Untuk membatasi panjang  $h_D(t)$ , maka **impulse response** tersebut dikali dengan sebuah **window**,  $w(t)$ . Dalam Gambar 13.2c, digunakan **window rectangular**.

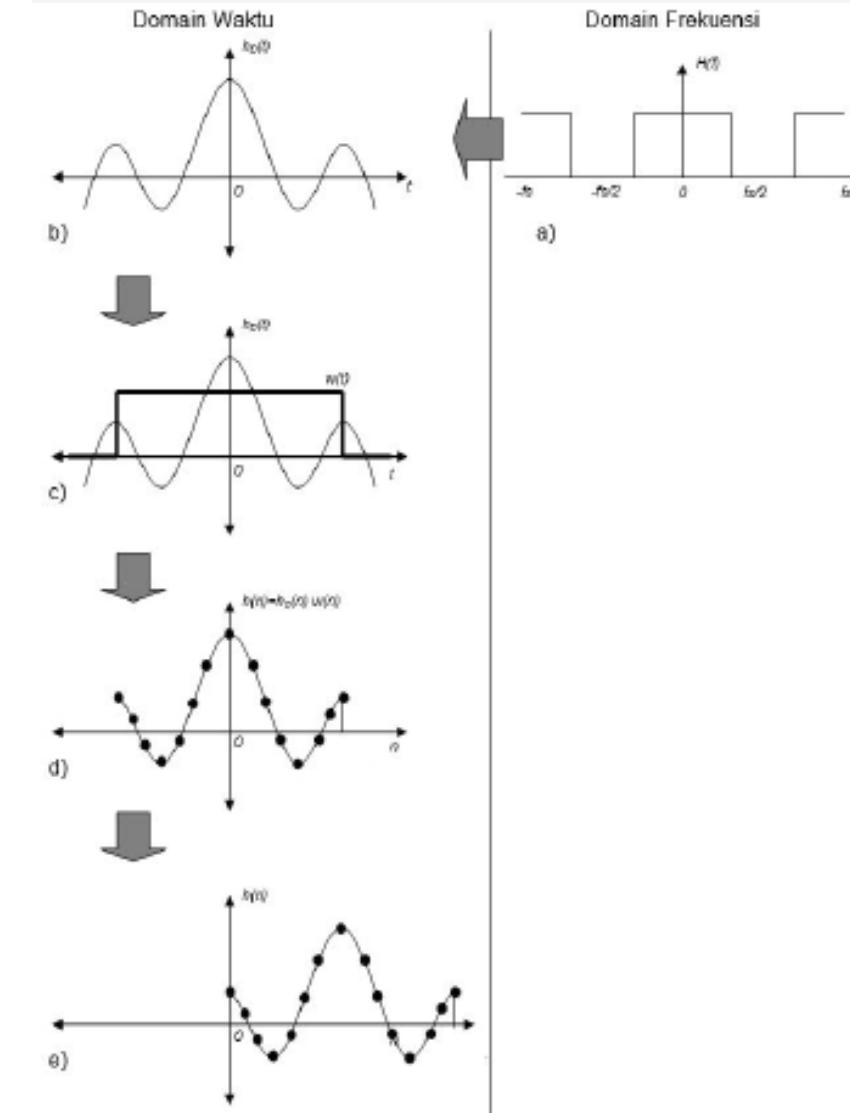


Gambar 13.2  
Proses desain FIR dengan metode window

# Proses Desain dengan Metode Window

Impulse response yang sudah dibatasi panjangnya ini kemudian disample dengan jumlah titik sesuai dengan orde dari filter yang akan dibuat,  $h(n)$ , seperti pada Gambar 13.2d. Langkah terakhir adalah menggeser impulse response tersebut agar kita tidak memiliki  $h(n)$  untuk  $n$  negatif (Gambar 13.2e). Jika ada  $n$  yang bernilai negatif maka filter tersebut tidak bersifat causal (bisa menghasilkan output sebelum ada input).

Setelah kita menggeser  $h(n)$  maka proses desain telah selesai dan kita telah memiliki filter FIR dengan respon frekuensi sesuai yang kita harapkan.



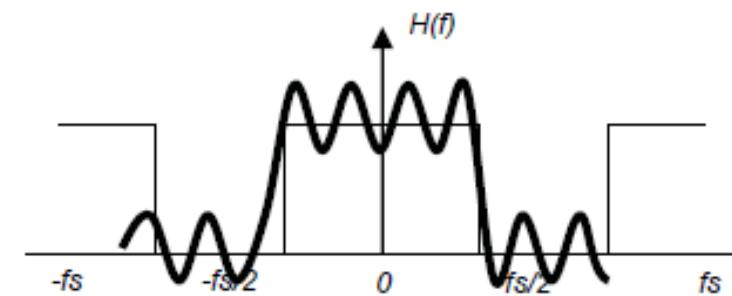
Gambar 13.2  
Proses desain FIR dengan metode window



# Proses Desain dengan Metode Window

Setelah kita menggeser  $h(n)$  maka proses desain telah selesai dan kita telah memiliki filter FIR dengan respon frekuensi sesuai yang kita harapkan.

Dampak dari penggunaan **window** terlihat pada saat kita mengamati kembali respon frekuensi dari filter yang sudah dibuat. Ternyata, respon frekuensi tidak sesuai dengan desain awal seperti ditunjukkan pada Gambar 13.3. Terdapat **ripple** dan **transition band** yang lebar.



Gambar 13.3  
Response frekuensi dari filter dengan rectangular window

# Proses Desain dengan Metode Window

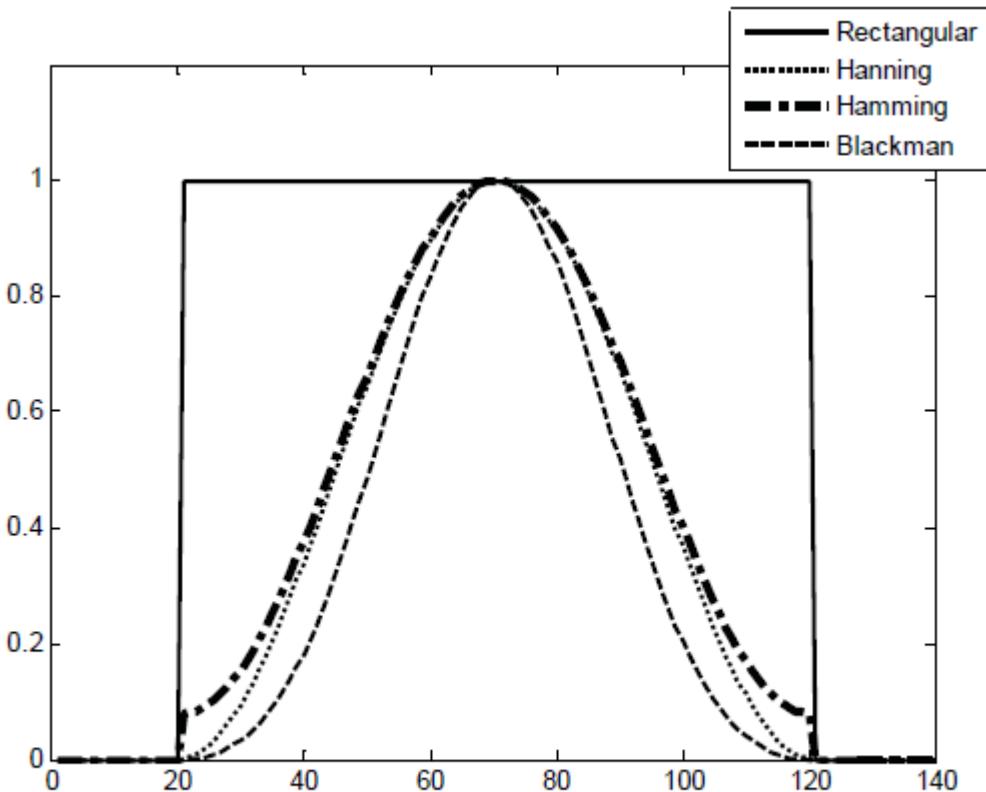


Kita dapat mengurangi dampak buruk ini dengan memilih bentuk **window** yang sesuai. Terdapat banyak pilihan **window**, beberapa diantaranya ditunjukkan pada Tabel 13.1 dan Gambar 13.4.

**Table 13.1**  
Berbagai window dan parameternya

Nama Window	Transition Band (Normalized Hz)	Passband ripple (dB)	Stopband attenuation (dB)	Persamaan $w(n)$ $ n  \leq (N/2)$
Rectangular	$0.9/(N+1)$	0.7416	21	1
Hanning	$3.1/(N+1)$	0.0546	44	$0.5 + 0.5 \cos\left(\frac{2\pi n}{N+1}\right)$
Hamming	$3.3/(N+1)$	0.0194	53	$0.54 + 0.46 \cos\left(\frac{2\pi n}{N+1}\right)$
Blackman	$5.5/(N+1)$	0.0017	74	$0.42 + 0.5 \cos\left(\frac{2\pi n}{N}\right) + 0.08 \cos\left(\frac{4\pi n}{N}\right)$

$N$  adalah orde dari filter



**Gambar 13.4**  
Bentuk dari window pada Tabel 13.1

# Proses Desain dengan Metode Window



Secara sepintas, proses ini terlihat sangat rumit khususnya proses dari Gambar 13.2a ke Gambar 13.2b dimana kita harus melakukan Inverse Fourier Transform (IFT) secara analitik. Untuk memudahkan proses ini maka Tabel 13.2 menuliskan  $h_D(n)$  dari beberapa filter yang umum digunakan. Tabel ini menolong kita karena kita tidak perlu melakukan proses IFT.

**Table 13.2**  
Impulse response ideal dari berbagai filter

Tipe Filter	Impulse response ideal, $h_D(n)$	
	$h_D(0)$	$h_D(n), n \neq 0$
LPF	$2f_c$	$2f_c \frac{\sin(n\omega_c)}{n\omega_c}$
HPF	$1-2f_c$	$-2f_c \frac{\sin(n\omega_c)}{n\omega_c}$
BPF	$2(f_2 - f_1)$	$2f_2 \frac{\sin(n\omega_2)}{n\omega_2} - 2f_1 \frac{\sin(n\omega_1)}{n\omega_1}$
BSF	$1-2(f_2 - f_1)$	$2f_1 \frac{\sin(n\omega_1)}{n\omega_1} - 2f_2 \frac{\sin(n\omega_2)}{n\omega_2}$

$f_c$  adalah frekuensi di tengah transition band, demikian juga dengan  $f_1$  dan  $f_2$

# Proses Desain dengan Metode Window



Langkah-langkah desain akan ditunjukkan dengan contoh berikut ini. Kita akan mendesain suatu filter:

Tipe filter	: HPF
Frekuensi sampling ( $f_s$ )	: 8 kHz
Frekuensi cut-off ( $f_c$ )	: 1.5 kHz
Transition band	: 500 Hz
Stopband attenuation	: > 50 dB
Passband ripple	: < 0.1 dB



# Proses Desain dengan Metode Window

## Langkah 1:

Kita akan melihat Tabel 13.2 untuk menemukan persamaan  $h_D(n)$  yang sesuai. Karena kita mendesain HPF maka  $h_D(n)$  yang sesuai adalah

$$h_D(0) = 1 - 2f_c$$

$$h_D(n) = -2f_c \frac{\sin(n\omega_c)}{n\omega_c} \quad \text{untuk } n \neq 0$$

Nilai  $f_c$  yang digunakan adalah  $f_c$  ternormalisasi dan terkoreksi yaitu di tengah-tengah transition band. Karena transition band sebesar 500 Hz, maka transition band ini berada dari frekuensi 1 kHz sampai 1.5 kHz. Nilai  $f_c'$  tepat berada di tengah-tengah transition band ini yaitu pada 1.25 kHz. Nilai  $f_c$  ternormalisasi dihitung dengan:

$$f_c = \frac{f'_c}{f_s} = \frac{1.5 \text{ kHz} - 0.5(500 \text{ Hz})}{8 \text{ kHz}} = \frac{1.25 \text{ kHz}}{8 \text{ kHz}} = 0.1563$$

Table 13.2  
Impulse response ideal dari berbagai filter

Tipe Filter	Impulse response ideal, $h_D(n)$	
	$h_D(0)$	$h_D(n), n \neq 0$
LPF	$2f_c$	$2f_c \frac{\sin(n\omega_c)}{n\omega_c}$
HPF	$1-2f_c$	$-2f_c \frac{\sin(n\omega_c)}{n\omega_c}$
BPF	$2(f_1 - f_2)$	$2f_2 \frac{\sin(n\omega_2)}{n\omega_2} - 2f_1 \frac{\sin(n\omega_1)}{n\omega_1}$
BSF	$1-2(f_1 - f_2)$	$2f_1 \frac{\sin(n\omega_1)}{n\omega_1} - 2f_2 \frac{\sin(n\omega_2)}{n\omega_2}$

$f_c$  adalah frekuensi di tengah transition band, demikian juga dengan  $f_1$  dan  $f_2$



# Proses Desain dengan Metode Window

## Langkah 2:

Kita akan melihat Tabel 13.1 untuk menemukan persamaan  $w(n)$  yang sesuai. Karena kita membutuhkan passband ripple  $< 0.1$  dB dan stopband attenuation  $> 50$  dB, maka kita hanya mungkin menggunakan *Hamming window* atau *Blackman window*. Sebaiknya kita memilih *Hamming window* karena memiliki transition band yang lebih sempit.

$$w(n) = 0.54 + 0.46 \cos\left(\frac{2\pi n}{N+1}\right)$$

Table 13.1  
Berbagai window dan parameternya

Nama Window	Transition Band (Normalized Hz)	Passband ripple (dB)	Stopband attenuation (dB)	Persamaan $w(n)$ $ n  \leq (N/2)$
Rectangular	$0.9/(N+1)$	0.7416	21	1
Hanning	$3.1/(N+1)$	0.0546	44	$0.5 + 0.5 \cos\left(\frac{2\pi n}{N+1}\right)$
Hamming	$3.3/(N+1)$	0.0194	53	$0.54 + 0.46 \cos\left(\frac{2\pi n}{N+1}\right)$
Blackman	$5.5/(N+1)$	0.0017	74	$0.42 + 0.5 \cos\left(\frac{2\pi n}{N}\right) + 0.08 \cos\left(\frac{4\pi n}{N}\right)$

$N$  adalah orde dari filter



# Proses Desain dengan Metode Window

## Langkah 3:

Menentukan panjang dari filter. Karena transition band harus 500 Hz maka normalized transition band

$$\Delta f = \frac{\text{transition band}}{f_s} = \frac{500}{8 \text{ kHz}} = 0.0625$$

Normalized transition band dari *Hamming window* adalah  $3.3/(N+1)$  maka

$$0.0625 = \frac{3.3}{N+1}$$

Sehingga  $N+1 = 52.8$  (atau dibulatkan menjadi 53), sehingga  $N = 52$ . Artinya untuk memenuhi parameter dari filter ini, kita harus mendesain filter dengan orde 52.



# Proses Desain dengan Metode Window

## Langkah 4:

Menghitung  $h(n) = h_D(n) w(n)$ . Karena kita akan mendesain filter dengan orde 52 maka kita akan menghitung 53 buah nilai  $h(n)$  untuk  $-26 \leq n \leq 26$ .

Kita cukup menghitung  $h(n)$  untuk  $0 \leq n \leq 26$ . Nilai  $h(n)$  untuk  $n$  negatif adalah pencerminan dari nilai  $h(n)$  untuk  $n$  positif yaitu

$$h(-1) = h(1)$$

$$h(-2) = h(2)$$

...

$$h(-26) = h(26)$$

$$h(0) = (1 - 2f_c) \left( 0.54 + 0.46 \cos\left(\frac{2\pi 0}{53}\right) \right) = (1 - 2(0.1563))(0.54 + 0.46) = 0.6874$$

$$h(1) = \left( -2f_c \frac{\sin(n\omega_c)}{n\omega_c} \right) \left( 0.54 + 0.46 \cos\left(\frac{2\pi 1}{53}\right) \right) = \left( -2(0.1563) \frac{\sin(1 \cdot 2\pi \cdot 0.1563)}{1 \cdot (2\pi \cdot 0.1563)} \right) \left( 0.54 + 0.46 \cos\left(\frac{2\pi}{53}\right) \right) = -0.2639$$

$$h(2) = \left( -2f_c \frac{\sin(n\omega_c)}{n\omega_c} \right) \left( 0.54 + 0.46 \cos\left(\frac{2\pi 2}{53}\right) \right) = \left( -2(0.1563) \frac{\sin(2 \cdot 2\pi \cdot 0.1563)}{2 \cdot (2\pi \cdot 0.1563)} \right) \left( 0.54 + 0.46 \cos\left(\frac{2\pi 2}{53}\right) \right) = -0.1451$$

Nilai nilai  $h(n)$  adalah:

\*\*\*

$$h(26) = \left( -2f_c \frac{\sin(n\omega_c)}{n\omega_c} \right) \left( 0.54 + 0.46 \cos\left(\frac{2\pi 26}{53}\right) \right) = \left( -2(0.1563) \frac{\sin(26 \cdot 2\pi \cdot 0.1563)}{26 \cdot (2\pi \cdot 0.1563)} \right) \left( 0.54 + 0.46 \cos\left(\frac{2\pi 26}{53}\right) \right) = -0.0004$$



# Proses Desain dengan Metode Window

Nilai-nilai ini akan sangat sulit dihitung secara manual. File Fig13\_5.m pada modul ini dapat digunakan untuk menghitung semua nilai  $h(n)$ . Hasil perhitungannya ditunjukkan pada Tabel 13.3.

**Table 13.3**  
**Impulse response dari HPF**

	0.6874	= $h(0)$
$h(-1) =$	-0.2639	= $h(1)$
$h(-2) =$	-0.1451	= $h(2)$
$h(-3) =$	-0.0200	= $h(3)$
$h(-4) =$	0.0535	= $h(4)$
$h(-5) =$	0.0575	= $h(5)$
$h(-6) =$	0.0180	= $h(6)$
$h(-7) =$	-0.0216	= $h(7)$
$h(-8) =$	-0.0322	= $h(8)$
$h(-9) =$	-0.0149	= $h(9)$
$h(-10) =$	0.0087	= $h(10)$
$h(-11) =$	0.0188	= $h(11)$
$h(-12) =$	0.0114	= $h(12)$
$h(-13) =$	-0.0027	= $h(13)$
$h(-14) =$	-0.0105	= $h(14)$
$h(-15) =$	-0.0078	= $h(15)$
$h(-16) =$	0.0000	= $h(16)$
$h(-17) =$	0.0053	= $h(17)$
$h(-18) =$	0.0048	= $h(18)$
$h(-19) =$	0.0008	= $h(19)$
$h(-20) =$	-0.0024	= $h(20)$
$h(-21) =$	-0.0026	= $h(21)$
$h(-22) =$	-0.0008	= $h(22)$
$h(-23) =$	0.0009	= $h(23)$
$h(-24) =$	0.0013	= $h(24)$
$h(-25) =$	0.0006	= $h(25)$
$h(-26) =$	-0.0004	= $h(26)$

# Proses Desain dengan Metode Window



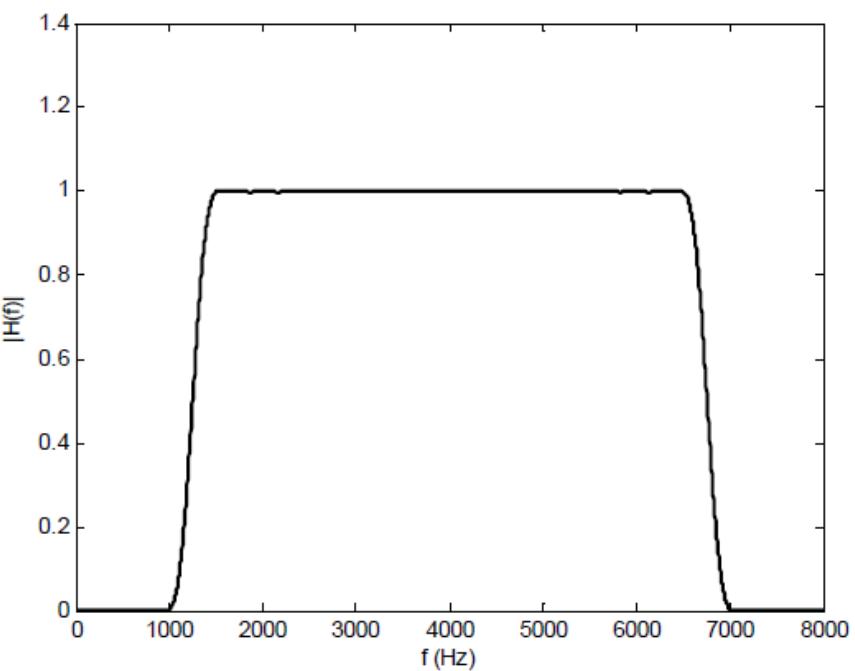
## Langkah 5:

Menggeser impulse response agar semua  $n$  bernilai positif sehingga filter bersifat causal. Proses ini dilakukan dengan mengganti semua  $n$  dengan  $n+26$  seperti pada Tabel 13.4.

**Table 13.4**  
Impulse response dari HPF yang bersifat causal

	0.6874	= $h(26)$
$h(25) =$	-0.2639	= $h(27)$
$h(24) =$	-0.1451	= $h(28)$
$h(23) =$	-0.0200	= $h(29)$
$h(22) =$	0.0535	= $h(30)$
$h(21) =$	0.0575	= $h(31)$
$h(20) =$	0.0180	= $h(32)$
$h(19) =$	-0.0216	= $h(33)$
$h(18) =$	-0.0322	= $h(34)$
$h(17) =$	-0.0149	= $h(35)$
$h(16) =$	0.0087	= $h(36)$
$h(15) =$	0.0188	= $h(37)$
$h(14) =$	0.0114	= $h(38)$
$h(13) =$	-0.0027	= $h(39)$
$h(12) =$	-0.0105	= $h(40)$
$h(11) =$	-0.0078	= $h(41)$
$h(10) =$	0.0000	= $h(42)$
$h(9) =$	0.0053	= $h(43)$
$h(8) =$	0.0048	= $h(44)$
$h(7) =$	0.0008	= $h(45)$
$h(6) =$	-0.0024	= $h(46)$
$h(5) =$	-0.0026	= $h(47)$
$h(4) =$	-0.0008	= $h(48)$
$h(3) =$	0.0009	= $h(49)$
$h(2) =$	0.0013	= $h(50)$
$h(1) =$	0.0006	= $h(51)$
$h(0) =$	-0.0004	= $h(52)$

Untuk memastikan respon frekuensi dari filter yang baru didesain, maka  $h(n)$  tersebut ditransformasi ke domain frekuensi dengan FFT dan hasilnya ditunjukkan pada Gambar 13.5. Gambar tersebut menunjukkan bahwa filter yang dirancang memiliki respon frekuensi sesuai spesifikasi desain.



**Gambar 13.5**  
Respons frekuensi dari HPF yang didesain



# Metode Window pada Matlab

Untuk mendesain FIR dengan metode window pada Matlab kita gunakan fungsi 'fir1'. Untuk filter yang didesain di atas kita menggunakan perintah:

```
>> h=fir1(52,1250/4000,'high',hamming(53))
```

Angka 52 adalah orde dari filter yang dirancang,  $1250$  adalah frekuensi cut-off terkoreksi,  $f_c$ ,  $4000$  adalah setengah  $f_s$ , dan window hamming dengan panjang 53.



# Desain Filter FIR dengan Metode Optimal

# Proses Desain dengan Metode Optimal

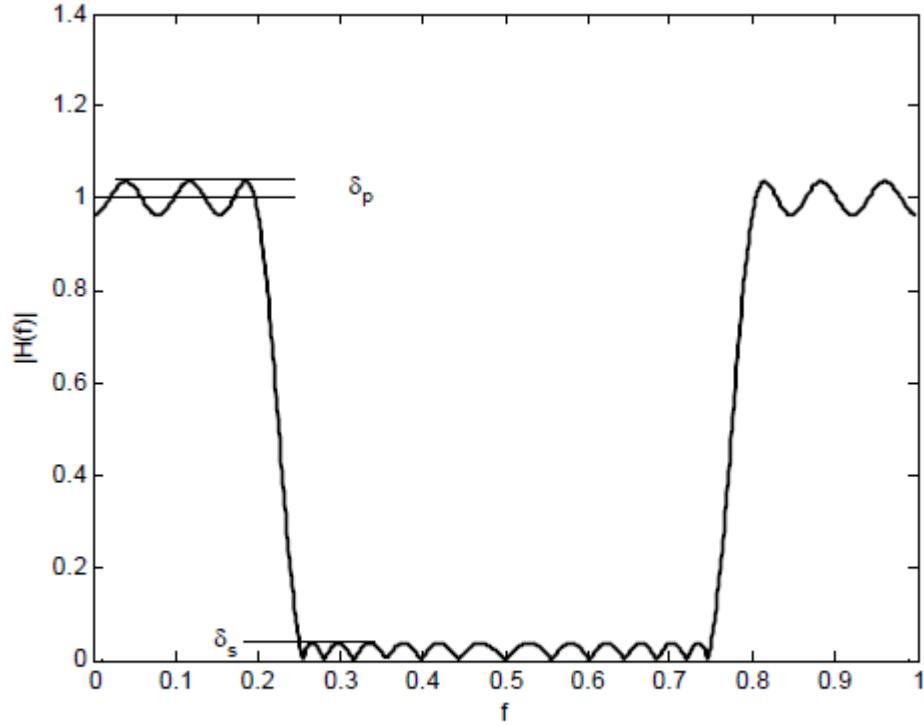


Setelah mempelajari materi pada bab ini mahasiswa akan dapat mendesain filter FIR dengan metode optimal, baik secara perhitungan manual maupun dengan menggunakan perangkat lunak pendukung.

Metode ini menghasilkan filter yang sangat baik sehingga menjadi pilihan pertama dalam setiap desain FIR. Metode ini menggunakan proses komputasi yang sangat panjang sehingga lebih praktis untuk diterapkan dengan bantuan komputer.

Kedua metode terdahulu selalu menghasilkan filter dengan *ripple* baik di *passband* maupun di *stopband*. Kita juga melihat bahwa *ripple* akan semakin besar di daerah sekitar frekuensi cut-off. Metode optimal ini berusaha untuk membuat filter FIR dengan *ripple* yang memiliki amplitudo konstan (*equiripple*) seperti ditunjukkan pada Gambar 14.2. Dengan demikian maka kita akan mendapat filter dengan respon frekuensi yang paling optimal.

# Proses Desain dengan Metode Optimal



Gambar 14.2  
Filter FIR dengan equiripple

Metode ini berusaha untuk meminimalisasi error,  $E(\omega)$ , yaitu selisih antara respon frekuensi ideal,  $HD(\omega)$ , dengan respon frekuensi hasil desain,  $H(\omega)$  setelah dikalikan dengan fungsi weighting,  $W(\omega)$ .

$$E(\omega) = W(\omega)[HD(\omega) - H(\omega)]$$

Langkah-langkah desain akan ditunjukkan dengan contoh berikut ini. Kita akan mendesain suatu filter

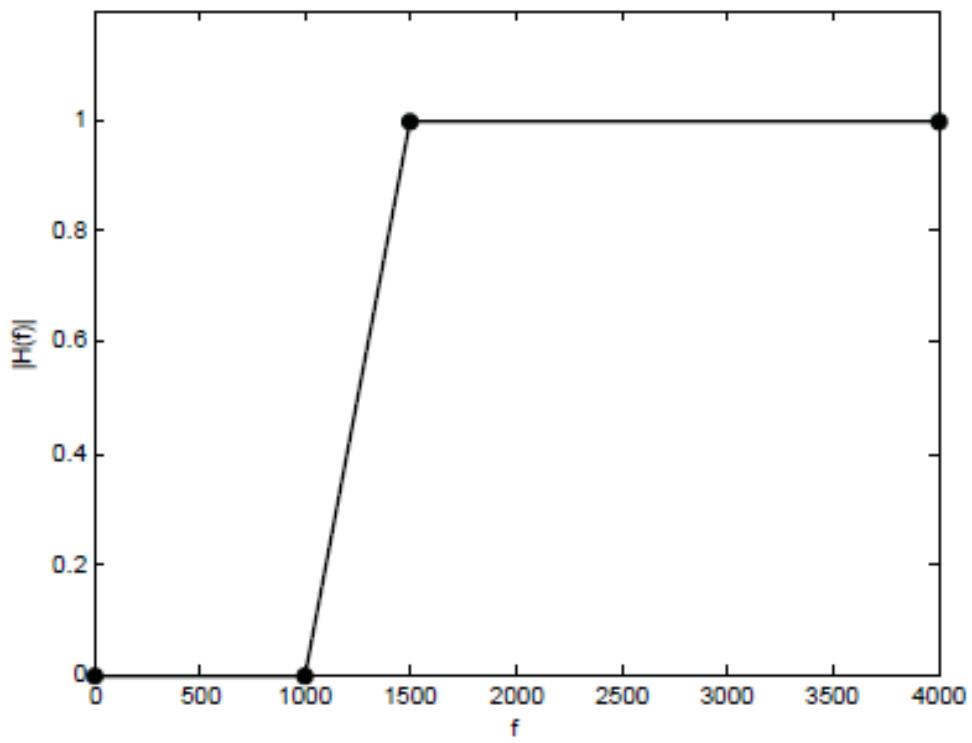
Tipe filter	: HPF
Frekuensi sampling ( $f_s$ )	: 8 kHz
Frekuensi cut-off ( $f_c$ )	: 1.5 kHz
Transition band	: 500 Hz
Stopband attenuation	: > 40 dB
Passband ripple	: < 0.1 dB
Orde	: 40

# Proses Desain dengan Metode Optimal



## Langkah 1:

Untuk memudahkan proses, kita akan menggambar respon frekuensi ideal yang diharapkan sampai dengan  $f_s/2$ . Kita menandai titik-titik sudut dari gambar tersebut.



Gambar 14.3

Respon frekuensi ideal dari filter yang akan didesain

## Langkah 2:

Catat frekuensi dan amplitudo dari titik-titik sudut tersebut. Kita gunakan frekuensi yang sudah dinormalisasi ke  $f_s/2$ .

$$F = [0 \quad 1000/4000 \quad 1500/4000 \quad 4000/4000]$$

dan

$$A = [0 \quad 0 \quad 1 \quad 1]$$

## Langkah 3:

Mencari perbandingan  $\delta_p$  dan  $\delta_s$ . Pada filter ini

$$\delta_p = 0.1 \text{ dB} = 10^{\left(\frac{0.1}{20}\right)} = 1.0116$$

$$\delta_s = -40 \text{ dB} = 10^{\left(\frac{-40}{20}\right)} = 0.01$$

Jadi perbandingan

$$\delta_p : \delta_s = 101 : 1$$

maka kita membuat

$$W = [1 \quad 101]$$

Karena filter yang dibuat adalah HPF maka nilai W dimulai dari rasio *stopband* (1) diikuti rasio *passband* (101).

# Proses Desain dengan Metode Optimal



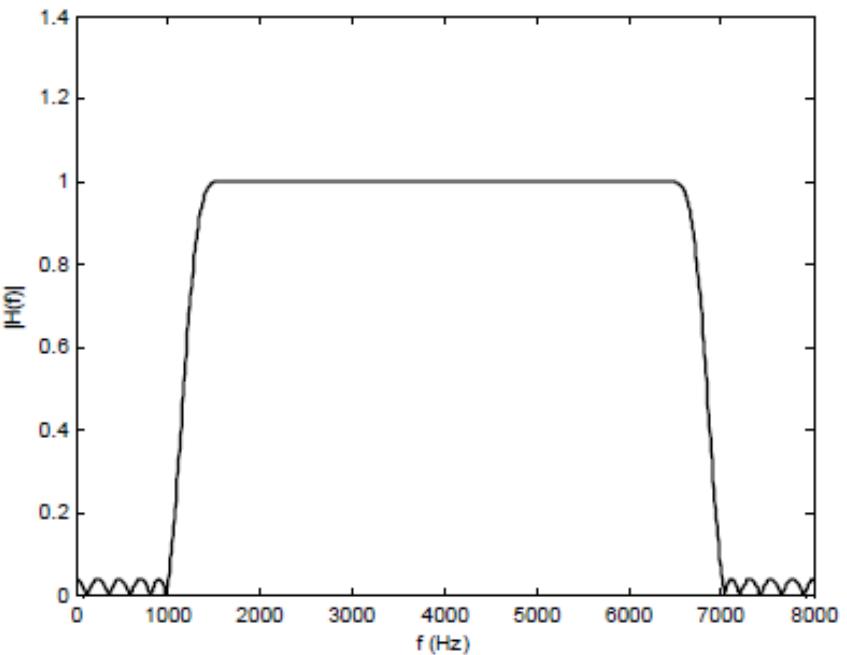
## Langkah 4:

Menggunakan perintah Matlab untuk menghitung  $h(n)$ .

```
>> h=firpm(40,F,A,W);
```

Detail proses desain dapat dilihat di file Fig13\_4.m.

Untuk memastikan respon frekuensi dari filter yang baru didesain, maka  $h(n)$  tersebut ditransformasi ke domain frekuensi dengan FFT dan hasilnya ditunjukkan pada Gambar 14.4. Gambar tersebut menunjukkan bahwa filter yang dirancang memiliki respon frekuensi sesuai spesifikasi desain.



**Gambar 14.4**  
**Respon frekuensi dari filter hasil desain.**