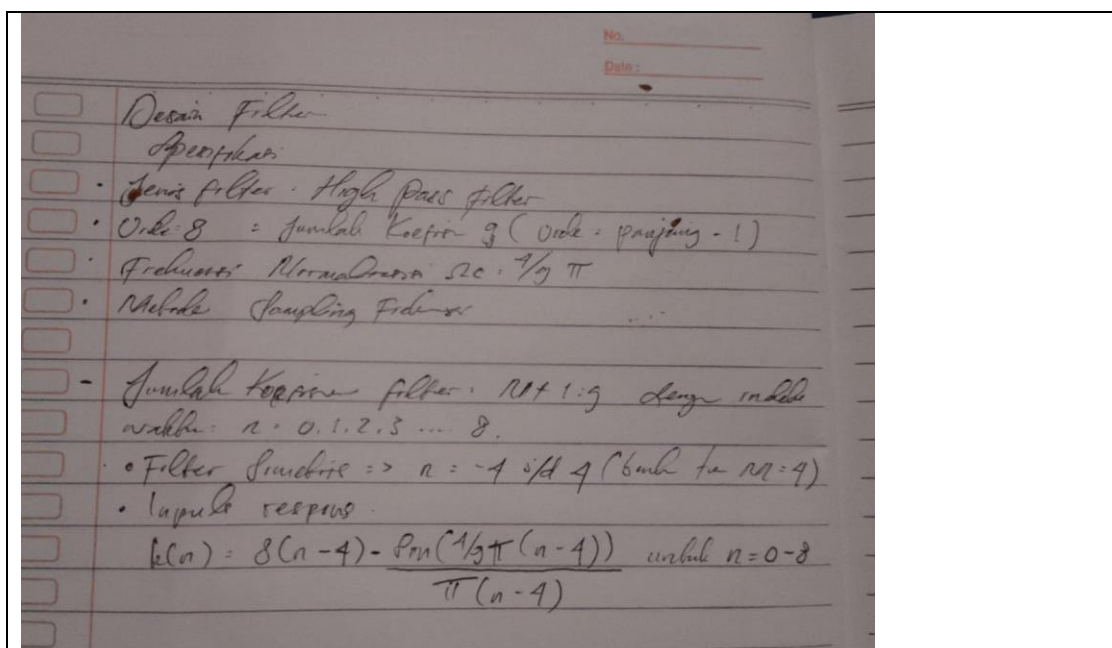


Nama : Ahmad Husin

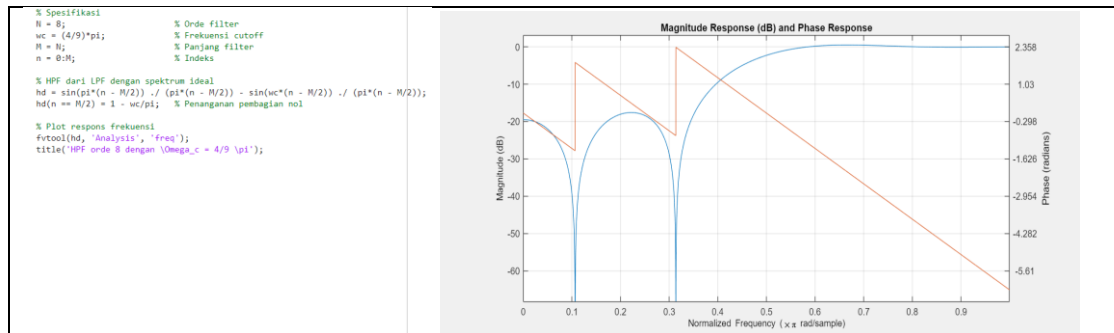
NPM : 2340304028

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 $f_c = 2 \text{ KHz}$, $f_{stop} = 3 \text{ KHz}$ dan $f_s = 10 \text{ KHz}$ 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 $f_s = 10 \text{ KHz}$) 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.

1. Jawab



2. Jawab



3. Jawab

Rancangan Low pass filter (LPF)

- Orde $g \rightarrow$ Koef : 10
- Frekuensi Cut off $f_c = 2$ kHz
- Frekuensi Sampling $f_s = 10$ kHz
- Metode Sample and hold
- Menghitung normalisasi f_c

$$\omega_c = 2\pi \cdot f_c / f_s = 2\pi \cdot \frac{2000}{10000} = 0.4\pi$$
- Input respons

$$h(n) = \frac{\sin(0.4\pi(n - 4.5))}{\pi(n - 4.5)}$$

4. Jawab

```

% Spesifikasi
N = 9; % Orde filter
fs = 10000; % Frekuensi sampling (Hz)
fc = 2000; % Frekuensi cutoff (Hz)
wc = 2*pi*fc/fs; % Frekuensi cutoff normalisasi

n = 0:N;
M = N/2; % Pusat simetri = 4.5

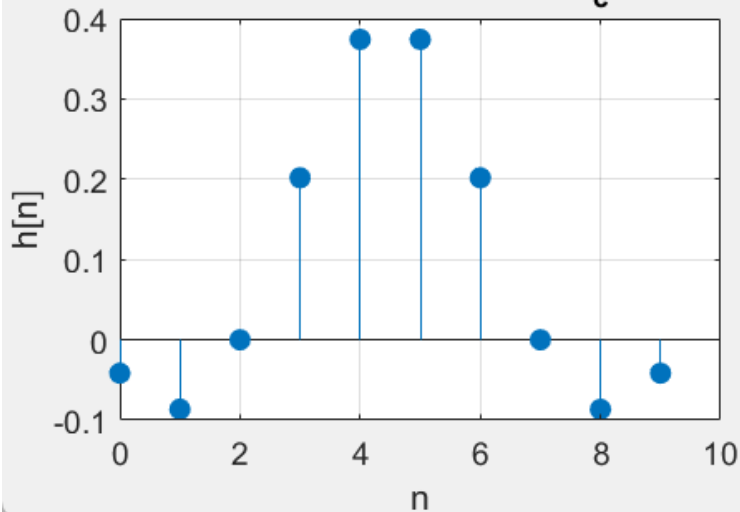
% Impuls respons LPF ideal (dengan metode sampling frekuensi)
hd = zeros(1, N+1);
for i = 1:length(n)
    k = n(i) - M;
    if abs(k) < 1e-10
        hd(i) = wc/pi; % nilai tengah
    else
        hd(i) = sin(wc * k)/(pi * k);
    end
end

% Plot impuls respons
stem(n, hd, 'filled');
xlabel('n'); ylabel('h[n]');
title('Impuls Respons LPF Orde 9, f_c = 2 kHz');
grid on;

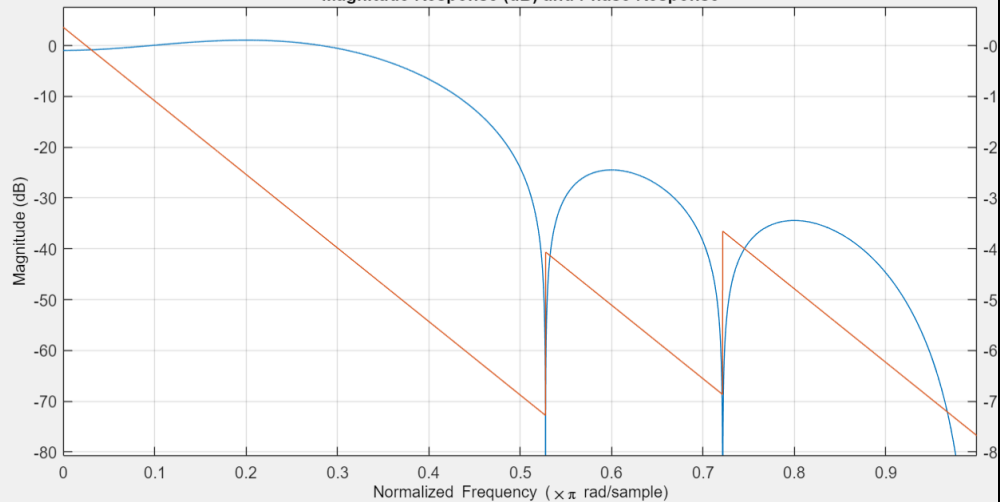
% Plot respons frekuensi
fvtool(hd, 'Analysis', 'freq');

```

Impuls Respons LPF Orde 9, $f_c = 2$ kHz



Magnitude Response (dB) and Phase Response



5. Jawab

```
% sampling
% frekuensi sampling
fs = 10000;
% waktu simulasi
t = 0:1/fs:0.01;

% sinyal input sinusoidal pada berbagai frekuensi (Hz)
f_list = [500, 1000, 2000, 3000, 4000];

% membuat filter LPF dari soal sebelumnya (orde 9, cutoff 2kHz)
N = 9;
M = N + 1;
omega = 2 * pi * (0:N) / M;
wc = 2 * pi * 2000 / fs;

% respons frekuensi ideal LPF
Hk = double(omega <= wc);

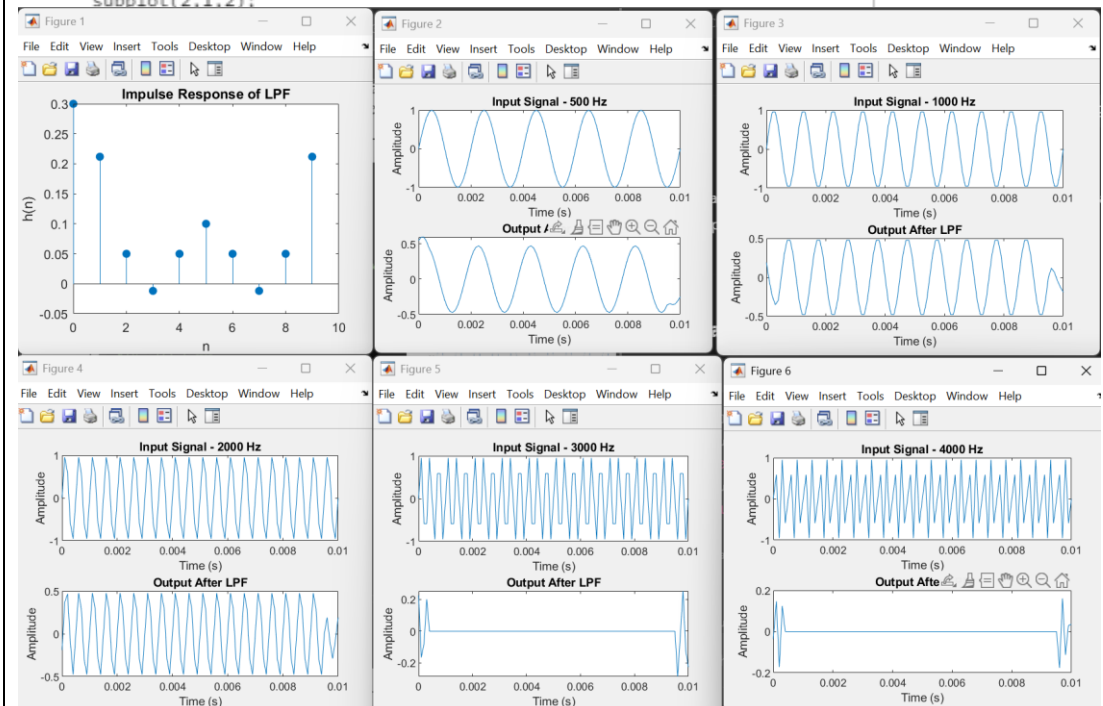
% hitung h(n) via IDFT
h = zeros(1, M);
for n = 0:N
    for k = 0:N
        h(n+1) = h(n+1) + Hk(k+1) * exp(1j * 2 * pi * k * n / M);
    end
    h(n+1) = real(h(n+1)) / M;
end

% menampilkan respon impuls filter
figure;
stem(0:N, h, 'filled');
title('Impulse Response of LPF');
xlabel('n'); ylabel('h(n)');

% menguji dengan masing-masing frekuensi input
for i = 1:length(f_list)
    f = f_list(i);
    x = sin(2*pi*f*t);
    y = conv(x, h, 'same');

    % plot input dan output
    figure;
    subplot(2,1,1);
    plot(t, x); title(['Input Signal - ', num2str(f), ' Hz']);
    xlabel('Time (s)'); ylabel('Amplitude');

    subplot(2,1,2);
```



6. Jawab

```

% sampling settings
% Frekuensi sampling (Hz)
fs = 10000;
% waktu simulasi 10 ms
t = 0:1/fs:0.01;

% frekuensi input untuk pengujian
f_list = [500, 1000, 2000, 3000, 4000];

% desain filter LPF menggunakan metode sampling frekuensi
N = 9;
M = N + 1;
omega = 2 * pi * (0:N) / M;
wc = 2 * pi * 2000 / fs;

% respons frekuensi ideal
Hk = double(omega <= wc);

% menghitung koefisien filter FIR h(n)
h = zeros(1, M);
for n = 0:N
    for k = 0:N
        h(n+1) = h(n+1) + Hk(k+1) * exp(1j * 2 * pi * k * n / M);
    end
    h(n+1) = real(h(n+1)) / M;
end

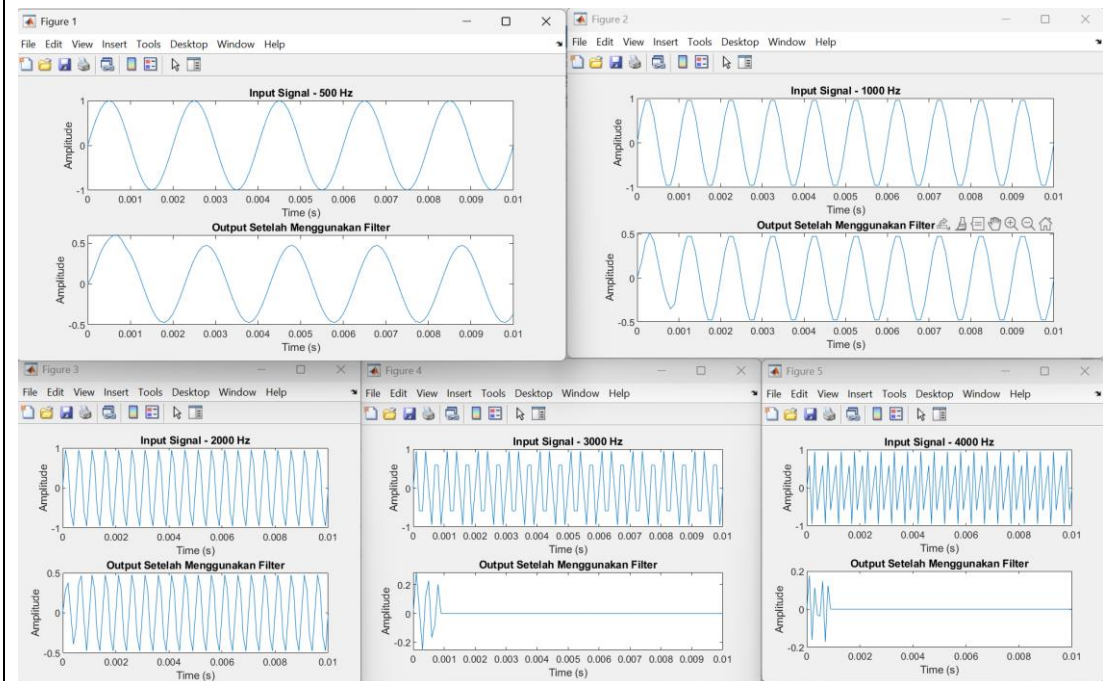
% menggunakan filter FIR tersebut dengan fungsi filter()
for i = 1:length(f_list)
    f = f_list(i);

    % sinyal input
    x = sin(2*pi*f*t);
    y = filter(h, 1, x);

    % plot input dan output
    figure;
    subplot(2,1,1);
    plot(t, x); title(['Input Signal - ', num2str(f), ' Hz']);
    xlabel('Time (s)'); ylabel('Amplitude');

    subplot(2,1,2);
    plot(t, y); title('Output Setelah Menggunakan Filter');
    xlabel('Time (s)'); ylabel('Amplitude');
end

```



7. Jawab

