

Nama : Rihan Hidayat  
Nim : 225150301111009  
kelas : Pemsin C

1. Sound dengan format wav yang masih ada noisenya yang saya gunakan pada projek ini terdapat pada link berikut :

<https://drive.google.com/drive/folders/1aP1NARtt91rX9O3R9AoBOdyo6zzvHlom?usp=sharing>

2. Codingan

```
[y, Fs] = wavread('C:\Pengguna\asus\Musik\recording.wav');

order = 100; // Order of the FIR filter
cutoff_freq = 1000; // Cutoff frequency of the filter in Hz

h = zeros(1, order+1);
for i = 1:order+1
if i == (order+1)/2
h(i) = 2 * cutoff_freq / Fs;
else
h(i) = sin(2 * %pi * cutoff_freq * (i - 1) / Fs) / ((i - 1 - (order + 1) / 2) * %pi);
end
end

h = h / sum(h);

y_filtered = conv(y, h);
```

```
wavwrite(real(y_filtered), Fs, 'C:\Pengguna\asus\Musik\baru.wav');
```

```
[y, Fs] = wavread('C:\Pengguna\asus\Musik\recording.wav')
```

```
t = (0:length(y)-1)/Fs;
```

```
subplot(221)
```

```
plot2d3(t, y)
```

```
xlabel('Waktu (detik)')
```

```
ylabel('Magnitudo')
```

```
title('Sinyal Asli')
```

```
playsnd(y, Fs)
```

```
Y = fft(y);
```

```
Y_abs = abs(Y);
```

```
subplot(222)
```

```
plot2d3(Y_abs)
```

```
xlabel('Frekuensi')
```

```
ylabel('Magnitudo')
```

```
title('Sinyal Asli setelah Transformasi Fourier')
```

```
[yl, Fsl] = wavread("C:\Pengguna\asus\Musik\baru.wav")
```

```
tl = (0:length(yl)-1)/Fsl;
```

```
subplot(223)
```

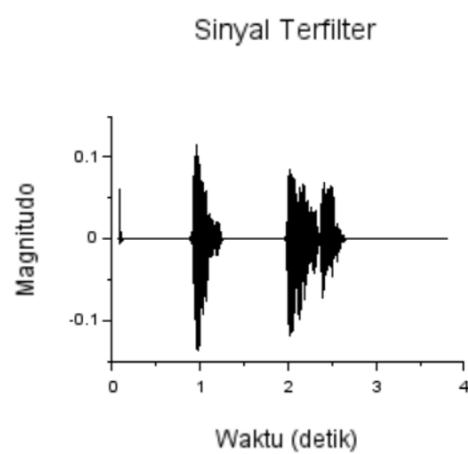
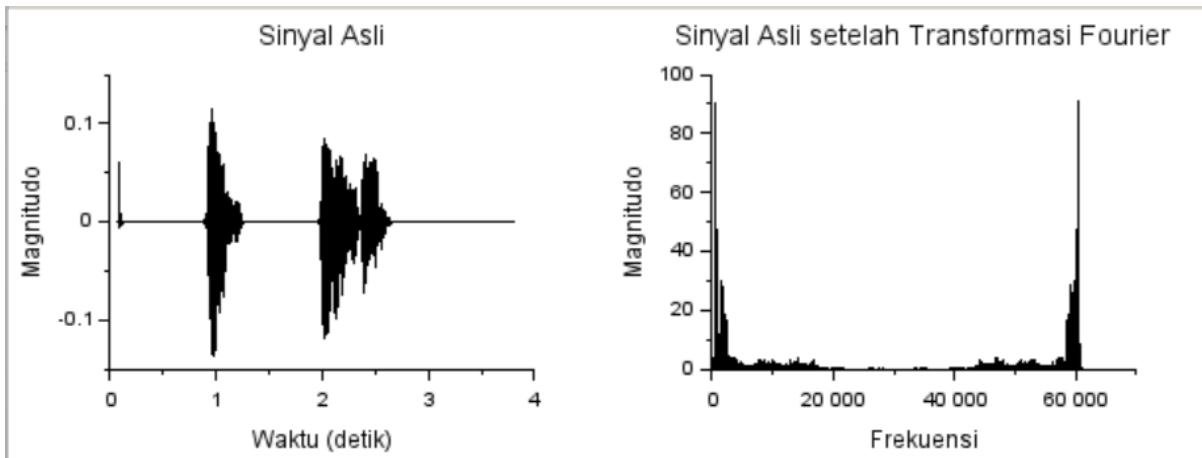
```
plot2d3(tl, yl)
```

```
xlabel('Waktu (detik)')
```

```
ylabel('Magnitudo')
```

```
title('Sinyal Terfilter')
```

hasil plotting



3. Output sound yang sudah terfilter dari noise:

<https://drive.google.com/drive/folders/1aP1NARTt91rX9O3R9AoBOdyo6zzvHlom?usp=sharing>

4. Penjelasan: Proses FFT (Fast Fourier Transform) mengubah sinyal audio dari domain waktu ke domain frekuensi, sehingga memungkinkan kita untuk menganalisis komponen frekuensi yang membentuk sinyal tersebut, termasuk yang berasal dari noise. Dengan menggunakan FFT, kita bisa mengidentifikasi komponen frekuensi dalam sinyal audio. Setelah FFT, kita dapat menerapkan filter untuk mengurangi atau menghilangkan noise dari sinyal audio. Filter ini bisa berupa FIR (Finite Impulse Response) atau IIR (Infinite Impulse Response), yang bekerja dengan menekan atau menghilangkan komponen frekuensi yang berhubungan dengan noise, sehingga menghasilkan sinyal audio yang lebih bersih. Jadi, secara singkat, proses ini melibatkan perubahan sinyal audio ke domain frekuensi dengan FFT, penerapan filter untuk mengurangi noise dalam domain frekuensi, dan kemudian mengembalikan sinyal yang telah difilter ke domain waktu jika diperlukan.