

Digital Signal Processing - Class Activity Submission 2

NIM: 122140008

Analisis Sinyal FFT

```
In [42]: import numpy as np
import pandas as pd
from scipy.fft import fft
import matplotlib.pyplot as plt
from io import StringIO

# Membaca file CSV
file_name = 'signal_122140008.csv'
df = pd.read_csv(file_name)

# Mencocokkan data dengan format yang diinginkan
time = df['time']
signal = df['amplitude']

# FFT
fft_result = np.fft.fft(signal)
N = len(signal)
T = time.iloc[1] - time.iloc[0] # Sampling interval
fs = 1 / (time.iloc[1] - time.iloc[0]) # Sampling frequency
freq = np.arange(len(signal)//2) * fs / len(signal)
fft_values = np.abs(fft_result[:N//2]/N) # Normalisasi

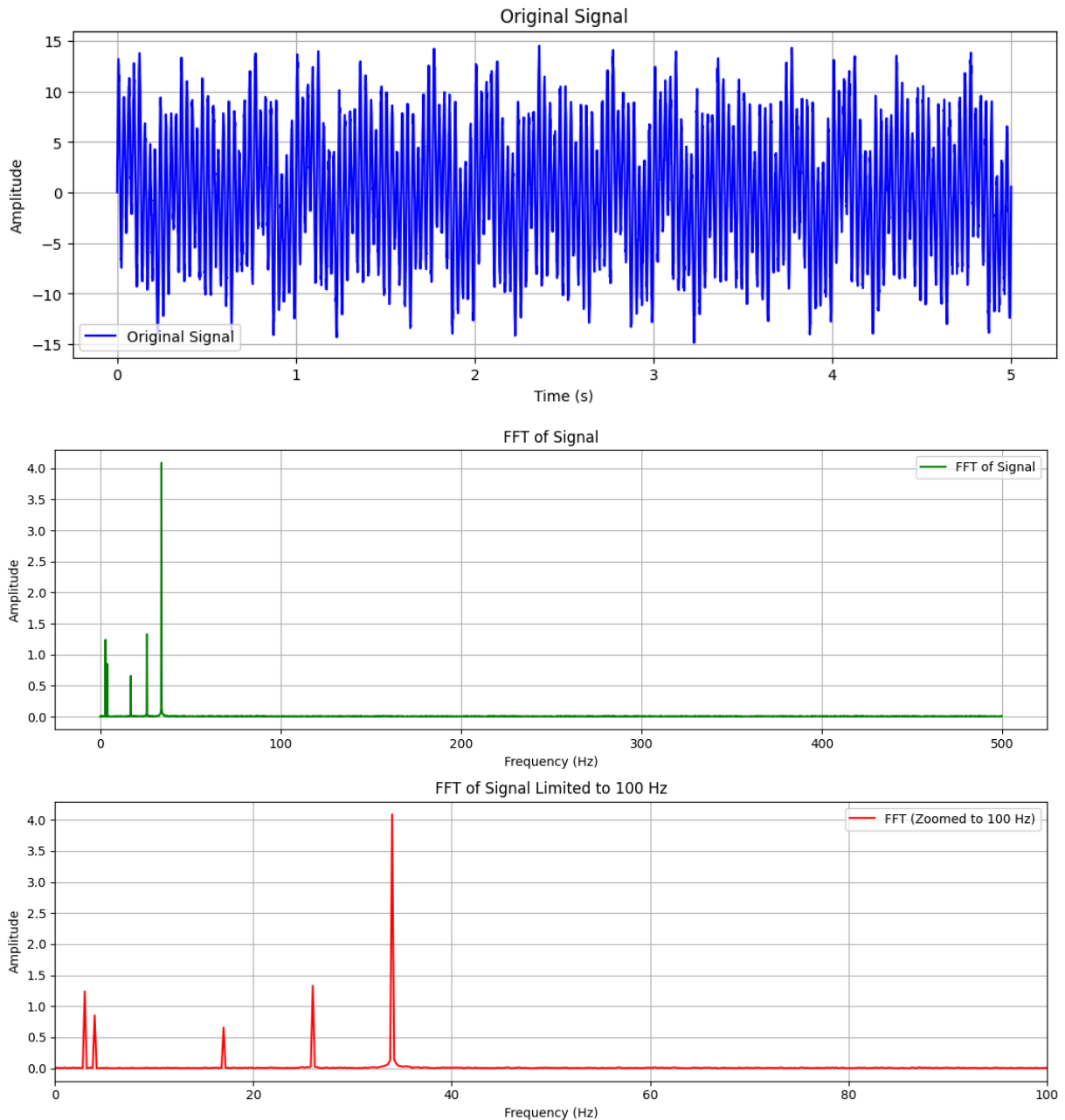
# Plot sinyal asli
plt.figure(figsize=(12, 4))
plt.plot(time, signal, label='Original Signal', color='blue')
plt.title('Original Signal')
plt.grid()
plt.xlabel('Time (s)')
plt.ylabel('Amplitude')
plt.legend()
plt.show()

# Plot FFT dan FFT (Zoomed) sebagai subplot
plt.figure(figsize=(12, 8))

# Plot FFT
plt.subplot(2, 1, 1)
plt.plot(freq, fft_values, label='FFT of Signal', color='green')
plt.title('FFT of Signal')
plt.grid()
plt.xlabel('Frequency (Hz)')
plt.ylabel('Amplitude')
plt.legend()
```

```
# Plot FFT (Zoomed)
plt.subplot(2, 1, 2)
plt.plot(freq, fft_values, label='FFT (Zoomed to 100 Hz)', color='red')
plt.xlim(0, 100) # Zoom in on the first 100 Hz
plt.title('FFT of Signal Limited to 100 Hz')
plt.grid()
plt.xlabel('Frequency (Hz)')
plt.ylabel('Amplitude')
plt.legend()

plt.tight_layout()
plt.show()
```



Analysis FFT

Dari hasil Visualisasi FFT sinyal di atas dapat dilihat bahwa beberapa puncak terlihat dominan pada spektrum frekuensi rendah sekitar di bawah 40Hz. Pada frekuensi ~35Hz terlihat merupakan puncak terbesar. Artinya frekuensi utama yang mendominasi sinyal tersebut terletak dalam rentang ~35Hz.

```
In [44]: from scipy.signal import butter, filtfilt

# Menentukan rentang frekuensi untuk band-pass filter
low_cutoff = 4 # Frekuensi rendah (4 Hz) hasil dari analisis FFT sebelumnya
high_cutoff = 40 # Frekuensi tinggi (40 Hz) hasil dari analisis FFT sebelumnya

# Membuat band-pass filter
b, a = butter(N=4, Wn=[low_cutoff, high_cutoff], fs=fs, btype='band')

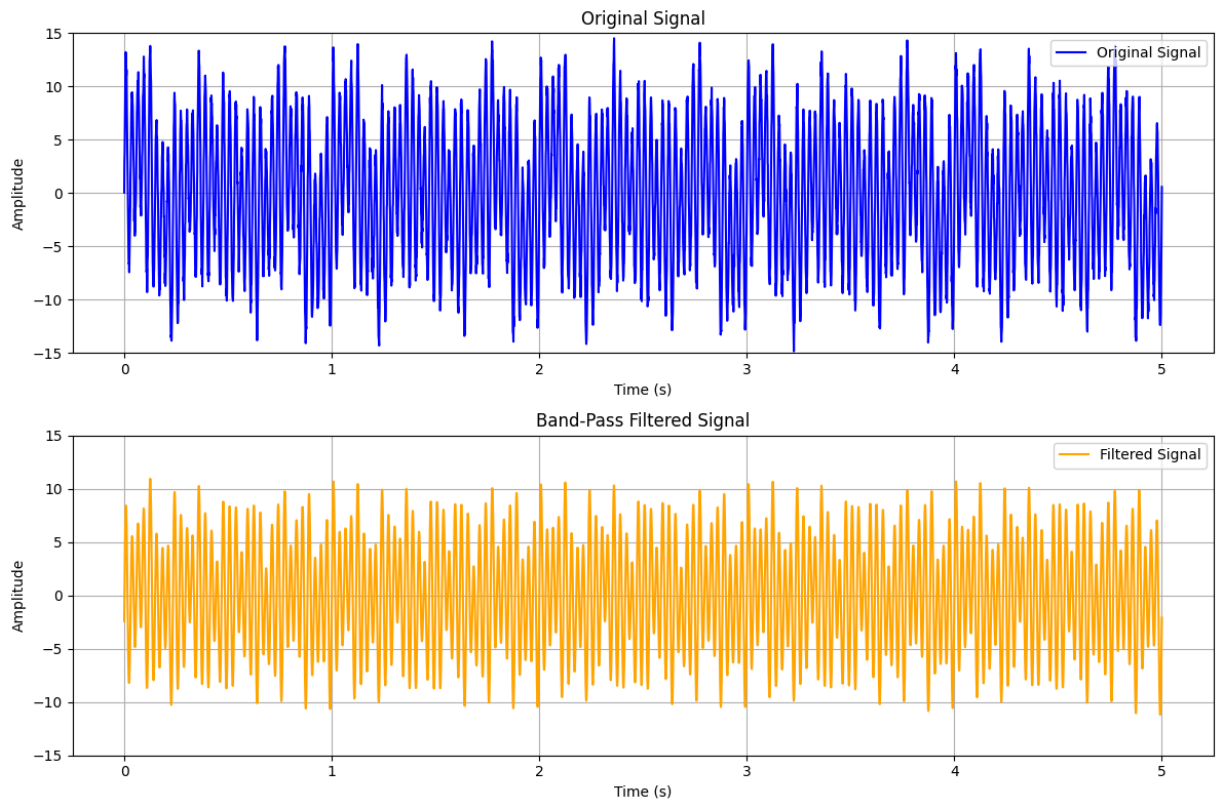
# Menerapkan filter pada sinyal asli
filtered_signal = filtfilt(b, a, signal)

# Membandingkan sinyal asli dan sinyal setelah band-pass filter
plt.figure(figsize=(12, 8))

# Plot sinyal asli
plt.subplot(2, 1, 1)
plt.plot(time, signal, label='Original Signal', color='blue')
plt.ylim(-15, 15) # Set y-limits for better visibility
plt.title('Original Signal')
plt.grid()
plt.xlabel('Time (s)')
plt.ylabel('Amplitude')
plt.legend()

# Plot sinyal hasil filter
plt.subplot(2, 1, 2)
plt.plot(time, filtered_signal, label='Filtered Signal', color='orange')
plt.ylim(-15, 15) # Set y-limits for better visibility
plt.title('Band-Pass Filtered Signal')
plt.grid()
plt.xlabel('Time (s)')
plt.ylabel('Amplitude')
plt.legend()

plt.tight_layout()
plt.show()
```



Analisis Band-Pass Signal

Dari hasil visualisasi di atas dapat disimpulkan bahwa terdapat perbedaan yang cukup signifikan antara sinyal sebelum dan sesudah dilakukan filter band-pass. Perbedaan dapat dilihat dengan perbandingan sebagai berikut.

Sinyal Asli:

- Tampak kompleks dan tidak teratur.
- Variasi Amplitudo yang sangat besar.
- Ada lonjakan mendadak di beberapa waktu.

Sinyal hasil Band-Pass Filtering:

- Bentuk gelombangnya jauh lebih halus dan teratur.
- Amplitudo lebih terkonsentrasi di kisaran ± 8 , dan tidak mengalami lonjakan ekstrem seperti sinyal asli.
- Tampak Amplitudo naik & turun secara berkala.

Lampiran

[Referensi penggunaan LLM](#)