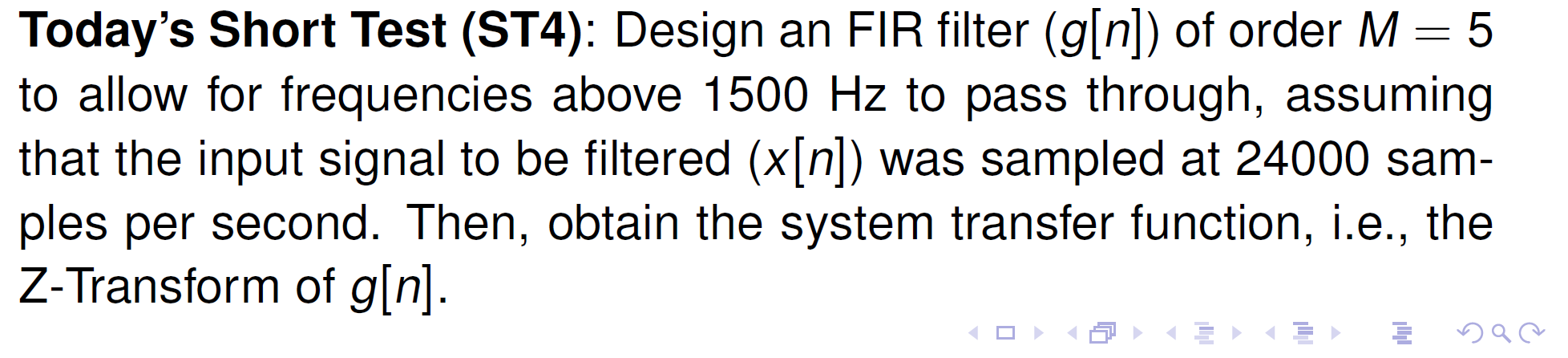
# Relatório - ST4: Projeto de Filtro FIR Passa-Altas

Disciplina: Digital Speech Processing

Aluno: Andrei Inoue Hirata

## Enunciado:



Projetar um filtro FIR g[n] de ordem M = 5 que permita a passagem de frequências acima de 1500 Hz, assumindo que o sinal de entrada x[n] foi amostrado a 24000 amostras por segundo. Em seguida, obter a função de transferência do sistema, ou seja, a Transformada Z de g[n].

## Código Python Utilizado:

# ST4 - Projeto de Filtro FIR Passa-Altas usando o método do professor (espelhamento e modulação)

import numpy as np

import matplotlib.pyplot as plt

from scipy.signal import freqz

from sympy import symbols, simplify

fs = 24000 # Frequência de amostragem

fc = 1500 # Frequência de corte

M = 5 # Ordem do filtro

n = np.arange(0, M + 1)

center = M / 2

# Frequência angular complementada: Wc = pi - pi \* (fc / (fs / 2)) = 7pi/8

wc = np.pi \* (1 - (fc / (fs / 2)))

# Passo 1: gerar h[n] com sinc deslocada

h = np.where(

n == center,

wc / np.pi,

np.sin(wc \* (n - center)) / (np.pi \* (n - center))

)

h\_reversed = h[::-1]

modulated = np.array([val \* (-1) \*\* i for i, val in enumerate(h\_reversed)])

print("Coeficientes do filtro passa-altas g[n]:")

print(np.round(modulated, 6))

g = modulated

z = symbols('z')

Gz = sum([g[i] \* z\*\*(-i) for i in range(len(g))])

Gz\_simplified = simplify(Gz)

print("\nTransformada Z G(z):")

print(Gz\_simplified)

ticks = [0, 1500, 6000, 12000]

labels = ["0", "1500 Hz", "6000 Hz (Nyquist)", "12000 Hz"]

w, h\_response = freqz(g, worN=8000)

frequencies = w \* fs / (2 \* np.pi)

plt.figure(figsize=(10, 4))

plt.plot(frequencies, 20 \* np.log10(np.abs(h\_response)), label="|G(f)| em dB", color='orange')

plt.axvline(x=1500, color='red', linestyle='--', label='Frequência de corte (1500 Hz)')

plt.title("Resposta em Frequência do Filtro FIR Passa-Altas")

plt.xlabel("Frequência (Hz)")

plt.ylabel("Magnitude (dB)")

plt.grid(True)

plt.legend()

plt.tight\_layout()

plt.show()

## Resultado:

g[n] = [0.070737, 0.176443, 0.624387, -0.624387, -0.176443, -0.070737]