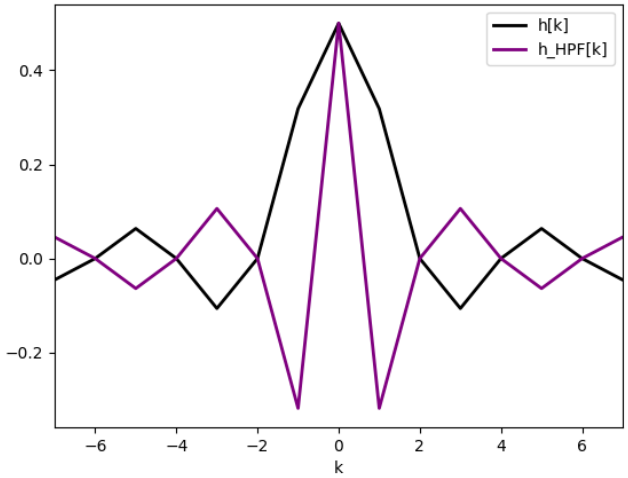
デジタル信号処理　第 14 回宿題

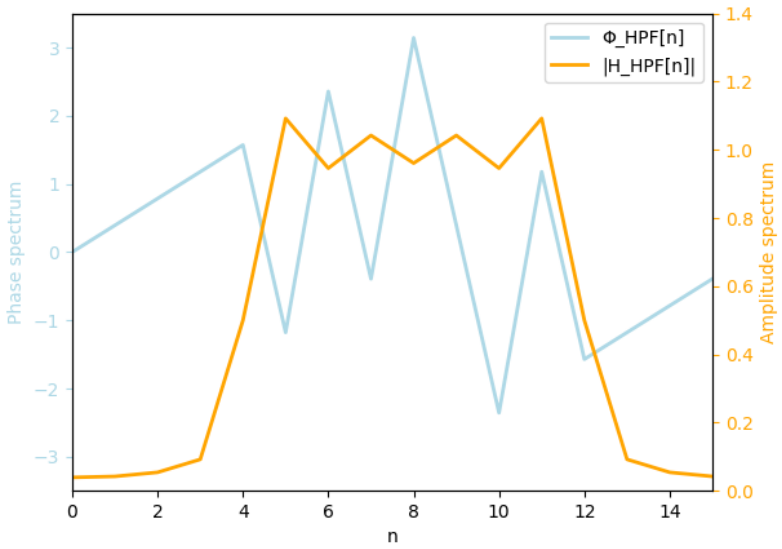
氏名：園山佳典
学籍番号：26002201991

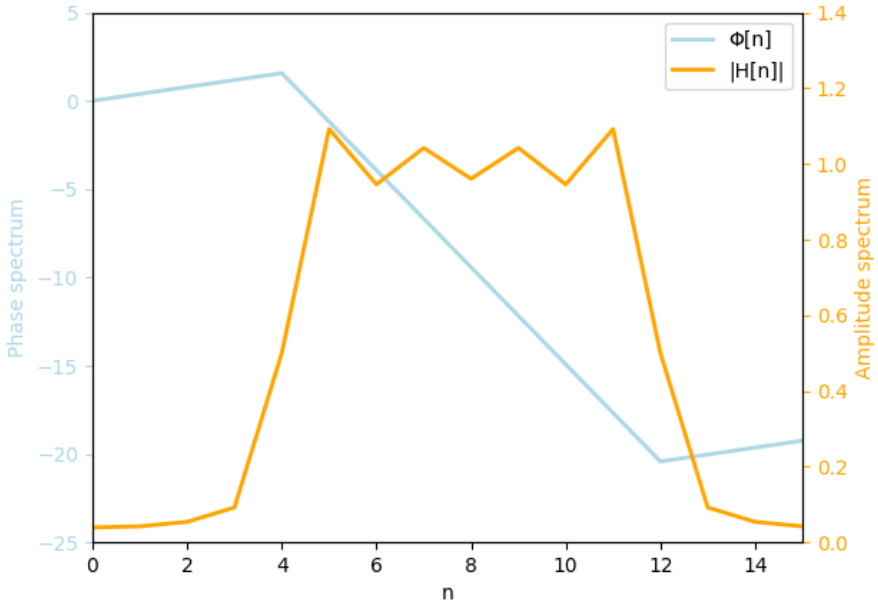
1 結果

(5-1)

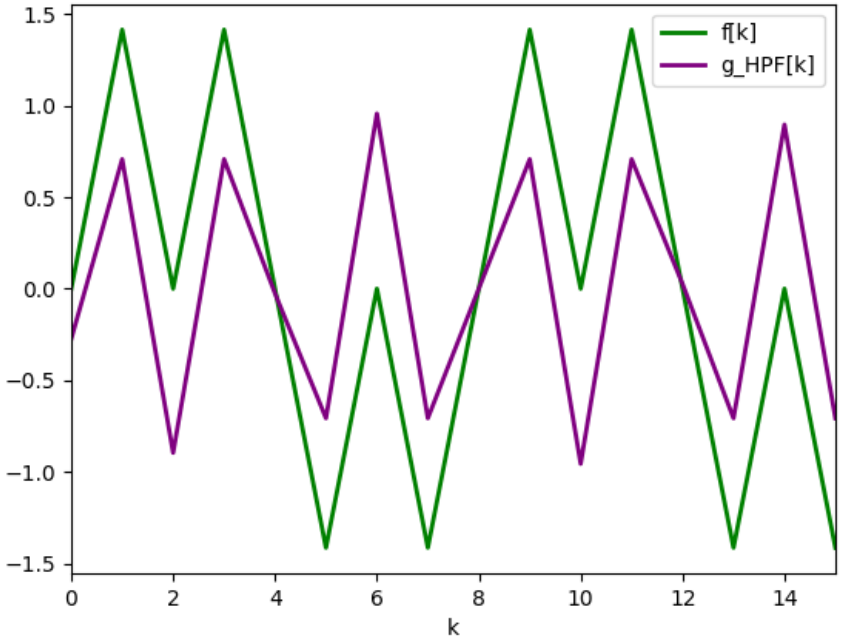


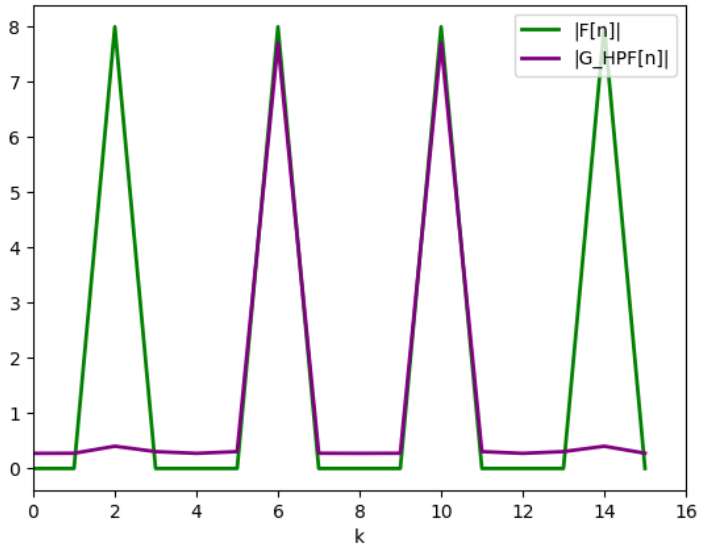
(5-2)


Φ_HPF[n]
|H_HPF[n]|
Phase spectrum
Amplitude spectrum
n


Φ[n]
|H[n]|
Phase spectrum
Amplitude spectrum
n

(5-3)


f[k]
g_HPF[k]
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
2
4
6
8
10
12
14
k


|F[n]|
|G_HPF[n]|
8
7
6
5
4
3
2
1
0
0
2
4
6
8
10
12
14
16
k

2.ソースコード

```
import numpy as np
import matplotlib.pyplot as plt

# パラメータ
T = 1
wc = 0.5 * np.pi / T
w1 = 0.25 * np.pi / T
w2 = 0.75 * np.pi / T
L = 8
N = 16

# k=0,1,2,...,L-1 のインデックスを作る
k1 = np.arange(start=-7, stop=10, step=1)
k2 = np.arange(start=0, stop=N, step=1)
# f[k] = sin(w1kT)の信号を作成
f = np.sin(w1 * k2 * T) + np.sin(w2 * k2 * T)
hL = (wc / np.pi) * np.sinc(wc * k1 * T / np.pi)
hH = hL * np.cos(k1 * np.pi)


g = [0] * (2 * N)
g1 = [0] * (N)

for i in range(2 * N - 2):
    g0 = 0
    for j in range(2 * N - 2):
        if (i - j >= 0) and (j < N) and (i - j < N):
            g0 += f[j] * hH[i - j]
    g[i] = g0
    if i >= 7 and i < 23:
        print(g0)
        g1[i - 7] = g0


dft_hH = np.fft.fft(hH, n=N)
dft_hH_amp = np.abs(dft_hH)
```

```
e = dft_hH / dft_hH_amp
theta1 = np.arctan2(e.imag, e.real)
unwrapped = np.unwrap(theta1)

F = np.fft.fft(f, n=N)
H = np.fft.fft(hL, n=N)
G = np.fft.fft(g1, n=N)
F_amp = np.abs(F)
G_amp = np.abs(G)


# 5-1
plt.figure()
plt.plot(k1, hL, color="black", linewidth=2, label="h[k]")
plt.plot(k1, hH, color="purple", linewidth=2, label="h_HPF[k]")
plt.xlim((-7, L - 1))
plt.xlabel("k")
# 凡例
plt.legend(loc="upper right")
plt.show()

# 5-2-1
fig, ax1 = plt.subplots()
# ax1 に f[k]を描画
ax1.plot(k2, theta1, color="lightblue", linewidth=2, label="Φ_HPF[n]")
ax1.set_xlabel("n")
ax1.set_ylabel("Phase spectrum", color="lightblue")
ax1.tick_params("y", colors="lightblue")
ax1.set_ylim(-3.5, 3.5)
ax1.set_xlim(0, N - 1)

# ax2 に θ を描画
ax2 = ax1.twinx()
ax2.plot(k2, dft_hH_amp, color="orange", linewidth=2, label="|H_HPF[n]|")
ax2.set_ylabel("Amplitude spectrum", color="orange")
ax2.tick_params("y", colors="orange")
ax2.set_ylim(0, 1.4)
```

```
# 凡例
lines = ax1.get_lines() + ax2.get_lines()
labels = [line.get_label() for line in lines]
ax1.legend(lines, labels, loc="upper right")
plt.show()

# 5-2-2
fig, ax1 = plt.subplots()
# ax1にf[k]を描画
ax1.plot(k2, unwrapped, color="lightblue", linewidth=2, label="Φ[n]")
ax1.set_xlabel("n")
ax1.set_ylabel("Phase spectrum", color="lightblue")
ax1.tick_params("y", colors="lightblue")
ax1.set_ylim(-25, 5)
ax1.set_xlim(0, N - 1)

# ax2にθを描画
ax2 = ax1.twinx()
ax2.plot(k2, dft_hH_amp, color="orange", linewidth=2, label="|H[n]|")
ax2.set_ylabel("Amplitude spectrum", color="orange")
ax2.tick_params("y", colors="orange")
ax2.set_ylim(0, 1.4)

# 凡例
lines = ax1.get_lines() + ax2.get_lines()
labels = [line.get_label() for line in lines]
ax1.legend(lines, labels, loc="upper right")

plt.show()

# 5-3-1
# h[k],h_HPF[k]の波形
plt.figure()
plt.plot(k2, f, color="g", linewidth=2, label="f[k]")
plt.plot(k2, g1, color="purple", linewidth=2, label="g_HPF[k]")
```

```
plt.xlim((0, N - 1))
plt.xlabel("k")
# 凡例
plt.legend(loc="upper right")
plt.show()

# 5-3-2
plt.figure()
plt.plot(k2, F_amp, color="g", linewidth=2, label="|F[n]|")
plt.plot(k2, G_amp, color="purple", linewidth=2, label="|G_HPF[n]|")
plt.xlim((0, N))
plt.xlabel("k")
# 凡例
plt.legend(loc="upper right")
plt.show()
```