

CENG 384 - Signals and Systems for Computer Engineers
Spring 2023
Homework 1

Geçit, Emre
e2521581@ceng.metu.edu.tr

Yancı, Baran
e2449015@ceng.metu.edu.tr

March 30, 2023

1. (a)

$$z = x + yj \implies \bar{z} = x - yj$$

$$2z + 5 = j - \bar{z}$$

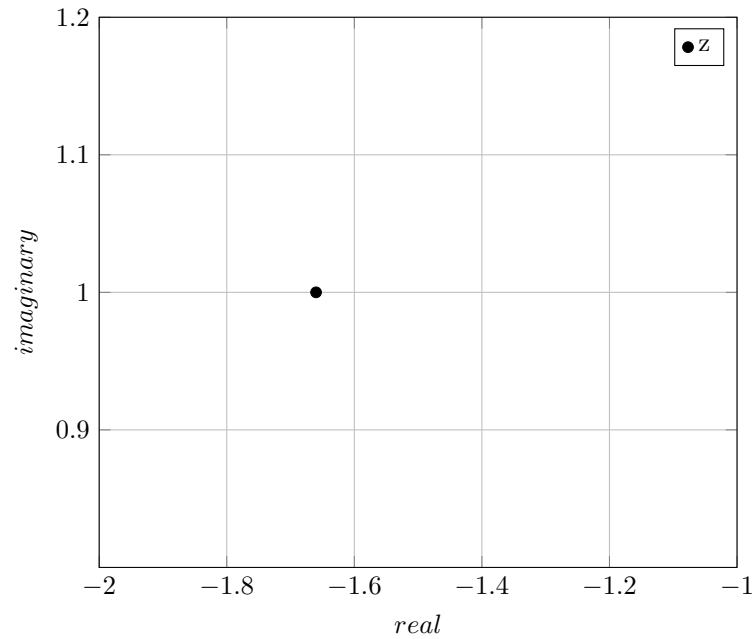
$$2(x + yj) + 5 = j - (x - yj)$$

$$2x + 5 + 2yj = (1 + y)j - x$$

$$y = 1, x = \frac{-5}{3}$$

$$z = \frac{-5}{3} + j$$

$$|z|^2 = \frac{25}{9} + 1 = \frac{34}{9}$$



(b)

$$z = re^{j\theta} \implies z^5 = r^5 e^{j5\theta}$$

$$32j = 32e^{j\pi/2}$$

$$32e^{j\pi/2} = r^5 e^{j5\theta} \implies r = 2, \theta = \pi/10$$

$$z = 2e^{j\pi/10}$$

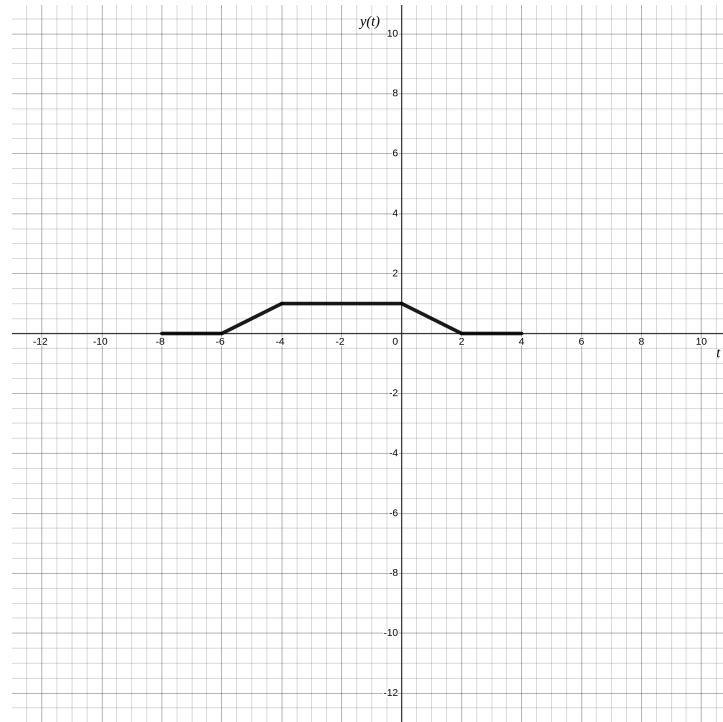
(c)

$$\begin{aligned}z &= \frac{(1+j)(\frac{1}{2} + \frac{\sqrt{3}}{2})j}{j-1} \\&= \frac{(j+1)(1+j)(\frac{1}{2} + \frac{\sqrt{3}}{2})j}{(j+1)(j-1)} \\&= \frac{(j+1)^2(\frac{1}{2} + \frac{\sqrt{3}}{2})}{-2} \\&= \frac{(j^2 + 2j + 1)(\frac{1}{2} + \frac{\sqrt{3}}{2})}{-2} \\&= \frac{(-1 + 2j + 1)(\frac{1}{2} + \frac{\sqrt{3}}{2})}{-2} \\&= \frac{2j(\frac{1}{2} + \frac{\sqrt{3}}{2})}{-2} \\&= -j(\frac{1}{2} + \frac{\sqrt{3}}{2}) \\z &= r \cos \theta + r \sin \theta j \\j(-\frac{1}{2} - \frac{\sqrt{3}}{2}) &= r \cos \theta + r \sin \theta j \\r \cos \theta &= 0 \\r \sin \theta &= -\frac{1}{2} - \frac{\sqrt{3}}{2} \\\cos \theta &= 0 \\\sin \theta &= -1 \\r &= \frac{1}{2} + \frac{\sqrt{3}}{2} \\\theta &= -\pi/2\end{aligned}$$

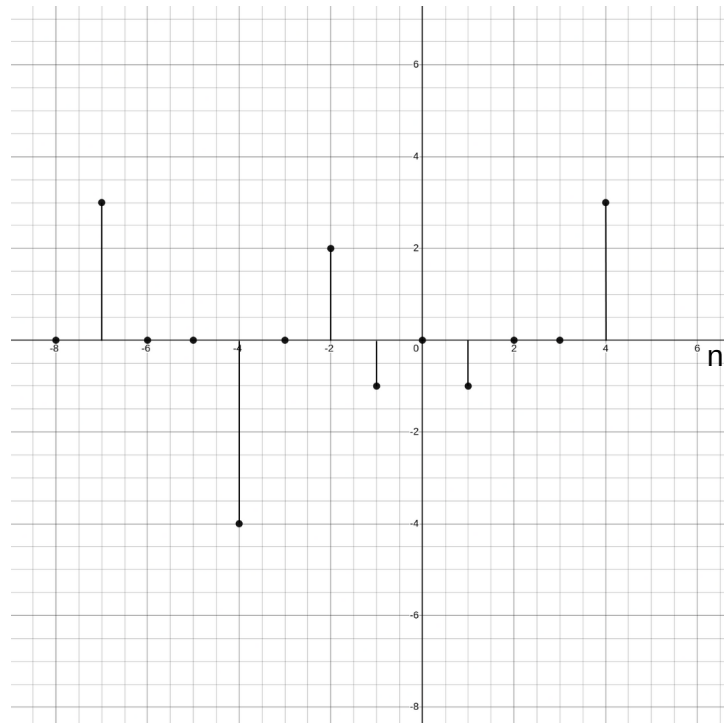
(d)

$$\begin{aligned}z &= je^{-j\pi/2} \\&= e^{j\pi/2}e^{-j\pi/2} \\&= e^0 = 1\end{aligned}$$

2. The graph of the function is given below.



3. (a) The graph of the function $x[-n] + x[2n - 1]$ is given below.



(b)

$$x[n] = -\delta[n - 1] + 2\delta[n - 2] - 4\delta[n - 4] + 3\delta[n - 7]$$

$$x[-n] = -\delta[-n - 1] + 2\delta[-n - 2] - 4\delta[-n - 4] + 3\delta[-n - 7]$$

$$x[2n - 1] = -\delta[2n - 2] + 2\delta[2n - 3] - 4\delta[2n - 5] + 3\delta[2n - 8]$$

$$x[-n] + x[2n - 1] = -\delta[-n - 1] + 2\delta[-n - 2] - 4\delta[-n - 4] + 3\delta[-n - 7] - \delta[2n - 2] + 2\delta[2n - 3] - 4\delta[2n - 5] + 3\delta[2n - 8]$$

4. (a)

$$\begin{aligned}
x(t) &= x(t + t_0) \\
5 \sin(3t - \frac{\pi}{4}) &= 5 \sin(3(t + t_0) - \frac{\pi}{4}) \\
\sin(3t - \frac{\pi}{4}) &= \sin(3t + 3t_0 - \frac{\pi}{4}) \\
\sin(3t - \frac{\pi}{4}) &= \sin(3t) \cos(3t_0 - \frac{\pi}{4}) + \cos(3t) \sin(3t_0 - \frac{\pi}{4}) \\
\sin(3t) \cos(-\frac{\pi}{4}) + \cos(3t) \sin(-\frac{\pi}{4}) &= \sin(3t) \cos(3t_0 - \frac{\pi}{4}) + \cos(3t) \sin(3t_0 - \frac{\pi}{4}) \\
\sin(3t) \frac{\sqrt{2}}{2} - \cos(3t) \frac{\sqrt{2}}{2} &= \sin(3t) \cos(3t_0 - \frac{\pi}{4}) + \cos(3t) \sin(3t_0 - \frac{\pi}{4}) \\
\sin(3t) \frac{\sqrt{2}}{2} - \sin(3t) \cos(3t_0 - \frac{\pi}{4}) &= \cos(3t) \frac{\sqrt{2}}{2} + \cos(3t) \sin(3t_0 - \frac{\pi}{4}) \\
\sin(3t) \left(\frac{\sqrt{2}}{2} - \cos(3t_0 - \frac{\pi}{4}) \right) &= \cos(3t) \left(\frac{\sqrt{2}}{2} + \sin(3t_0 - \frac{\pi}{4}) \right) \\
\frac{\sqrt{2}}{2} - \cos(3t_0 - \frac{\pi}{4}) &= 0 \\
\frac{\sqrt{2}}{2} + \sin(3t_0 - \frac{\pi}{4}) &= 0 \\
t_0 &= \frac{2\pi}{3}
\end{aligned}$$

(b)

$$\begin{aligned}
x[n] &= x[n + t_0] \\
\cos[\frac{13\pi}{10}n] + \sin[\frac{7\pi}{10}n] &= \cos[\frac{13\pi}{10}(n + t_0)] + \sin[\frac{7\pi}{10}(n + t_0)] \\
\sin[\frac{\pi}{2} - \frac{13\pi}{10}n] + \sin[\frac{7\pi}{10}n] &= \sin[\frac{\pi}{2} - \frac{13\pi}{10}(n + t_0)] + \sin[\frac{7\pi}{10}(n + t_0)] \\
\sin[\frac{5\pi}{10} - \frac{13\pi}{10}n] + \sin[\frac{7\pi}{10}n] &= \sin[\frac{5\pi}{10} - \frac{13\pi}{10}(n + t_0)] + \sin[\frac{7\pi}{10}(n + t_0)] \\
\sin[\frac{\pi}{10}(13n - 5)] + \sin[\frac{7\pi}{10}n] &= \sin[\frac{\pi}{10}(13n + 13t_0 - 5)] + \sin[\frac{7\pi}{10}(n + t_0)] \\
2 \sin(\frac{\frac{\pi}{10}(13n - 5) + \frac{7\pi}{10}n}{2}) \cos(\frac{\frac{\pi}{10}(13n - 5) - \frac{7\pi}{10}n}{2}) &= 2 \sin(\frac{\frac{\pi}{10}(13n + 13t_0 - 5) + \frac{7\pi}{10}(n + t_0)}{2}) \cos(\frac{\frac{\pi}{10}(13n + 13t_0 - 5) - \frac{7\pi}{10}(n + t_0)}{2}) \\
\sin(\frac{\pi}{20}(20n - 5)) \cos(\frac{\pi}{20}(6n - 5)) &= \sin(\frac{\pi}{20}(20n + 20t_0 - 5)) \cos(\frac{\pi}{20}(6n + 6t_0 - 5)) \\
\sin(n\pi - \frac{\pi}{4}) \cos(\frac{3n\pi}{10} - \frac{\pi}{4}) &= \sin(n\pi + t_0\pi - \frac{\pi}{4}) \cos(\frac{3n\pi + 3t_0\pi}{10} - \frac{\pi}{4})
\end{aligned}$$

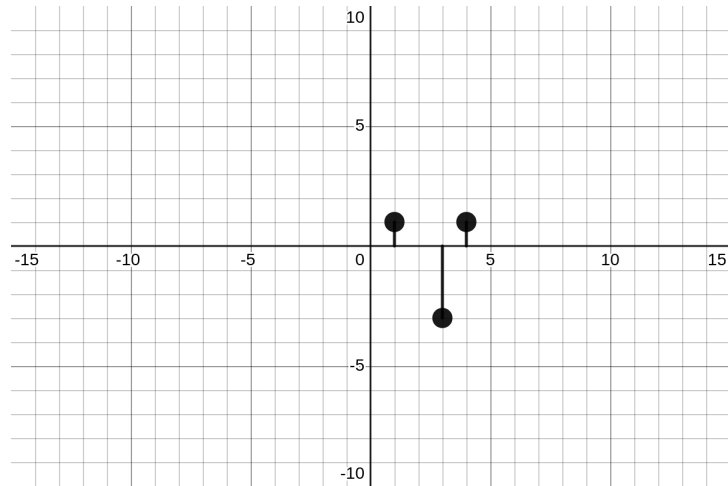
The smallest integer t_0 that satisfies the equation above is $t_0 = 20$.

(c) The signal is not periodic as there is no integer that satisfies the equation below.

$$\frac{1}{2} \cos(7n - 5) = \frac{1}{2} \cos(7(n + t_0) - 5)$$

5. (a) $x(t) = u[t - 1] - 3u[t - 3] + u[t - 4]$

(b) $\frac{dx(t)}{dt} = \delta(t-1) - 3\delta(t-3) + \delta(t-4)$ The graph of $\frac{dx(t)}{dt}$ is given below.



6. (a)

(b)

7. (a)

```
def decompose(signal_name):
    """Read the CSV file with the signal name, decompose the signal into even and odd components,
    and save the results as PNG files."""

    with open(signal_name + ".csv", "r", encoding="ascii") as file:
        data = [float(item) for item in file.read().split(",")]
        start = int(data[0])
        signal = data[1:]
        end = start + len(signal) - 1

    pyplot.title("Original Signal")
    pyplot.plot(range(start, end + 1), signal)
    pyplot.savefig(IMGES_PATH + signal_name + "_original.png")
    pyplot.clf()

    if abs(start) > end:
        signal = signal + [0] * (abs(start) - end)
        end = -start
    else:
        signal = [0] * (end - abs(start)) + signal
        start = -end

    even = [(x + y) / 2 for x, y in zip(signal, signal[::-1])]
    odd = [(x - y) / 2 for x, y in zip(signal, signal[::-1])]

    pyplot.title("Even Component")
    pyplot.plot(range(start, end + 1), even)
    pyplot.savefig(IMGES_PATH + signal_name + "_even.png")
    pyplot.clf()

    pyplot.title("Odd Component")
    pyplot.plot(range(start, end + 1), odd)
    pyplot.savefig(IMGES_PATH + signal_name + "_odd.png")
    pyplot.clf()
```

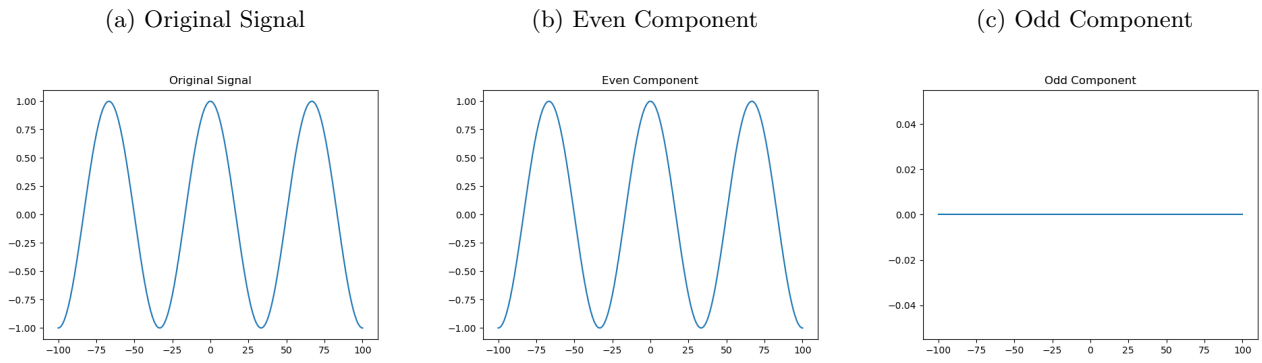


Figure 1: Sinusoidal Signal Decomposition

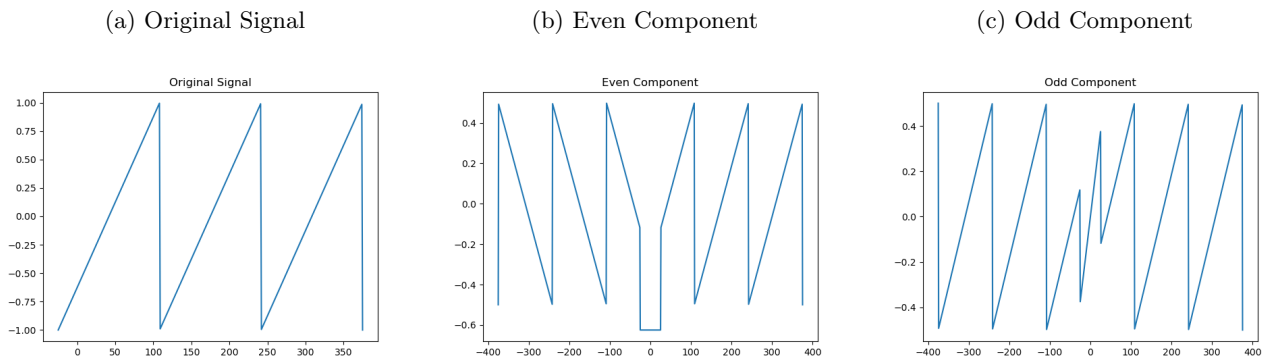


Figure 2: Shifted Sawtooth Signal Decomposition

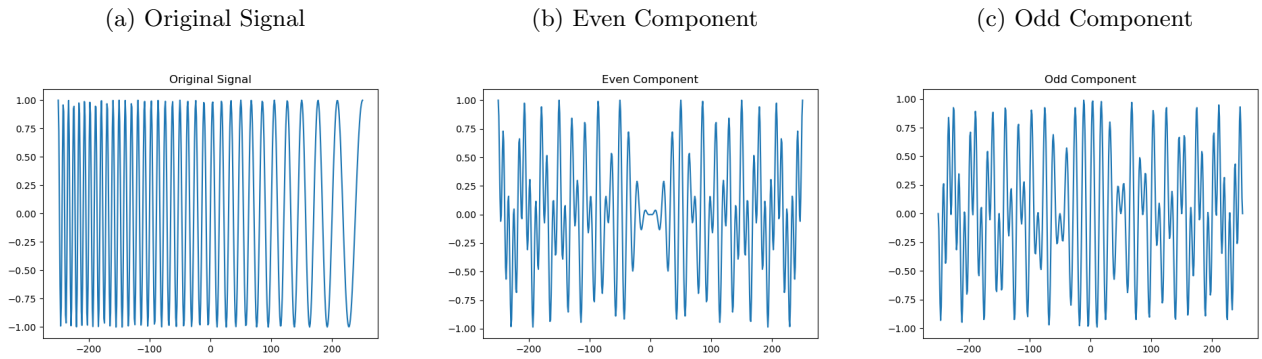


Figure 3: Chirp Signal Decomposition

(b)

```
class Signal:
    def __init__(self, signal, start, a, b):
        self.signal = signal
        self.start = start
        self.a = a
        self.b = b

    def __getitem__(self, index):
        return self.signal[self.a * index + self.b - self.start]

def shift_n_scale(signal_name):
    """
    Read the CSV file with the signal name, shift and scale the signal,
```

and save the results as PNG files.

This functions reads a signal $x[n]$, and produces $x[a*n + b]$ for a and b

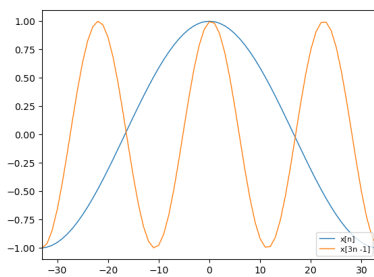
```
"""
with open(signal_name + ".csv", "r", encoding="ascii") as file:
    data = [float(item) for item in file.read().split(",")]
start = int(data[0])
a = int(data[1])
b = int(data[2])
signal = Signal(data[3:], start, a, b)
end = start + len(signal.signal) - 1

new_start = (start - b) // a
new_end = (end - b) // a
pyplot.xlim(new_start, new_end)

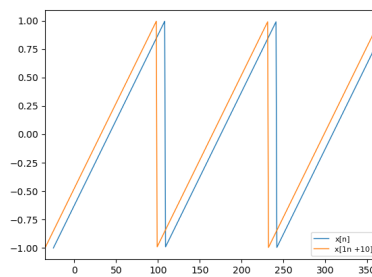
pyplot.plot(range(start, end + 1), signal.signal, linewidth=1)

if new_start > new_end:
    domain = range(new_start, new_end, -1)
else:
    domain = range(new_start, new_end + 1)
pyplot.plot(
    domain,
    [signal[i] for i in domain],
    linewidth=1,
)
pyplot.legend(
    ["x[n]", "x[" + str(a) + "n " + ("+" if b >= 0 else "") + str(b) + "]"],
    loc="lower right",
    fontsize=8,
)
pyplot.savefig(IMAGES_PATH + signal_name)
pyplot.clf()
```

(a) Sinusoidal Signal



(b) Shifted Sawtooth Signal



(c) Chirp Signal

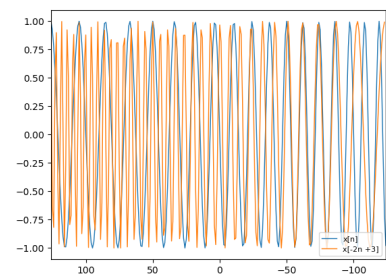


Figure 4: Shift and Scale