Digital Signal Processing

G V V Sharma*

Contents

***Abstract*—This manual provides a simple introduction to digital signal processing.**

1 Software Installation

Run the following commands

```
sudo apt−get update
sudo apt−get install libffi−dev libsndfile1 python3
    −scipy python3−numpy python3−matplotlib
sudo pip install cffi pysoundfile
```

2 Digital Filter

2.1 Download the sound file from

```
wget https://raw.githubusercontent.com/
    gadepall/
EE1310/master/filter/codes/Sound_Noise.wav
```

2.2 You will find a spectrogram at https://academo.org/demos/spectrum-analyzer. Upload the sound file that you downloaded in Problem 2.1 in the spectrogram and play. Observe the spectrogram. What do you find?
Solution: There are a lot of yellow lines between 440 Hz to 5.1 KHz. These represent the synthesizer key tones. Also, the key strokes are audible along with background noise.

2.3 Write the python code for removal of out of band noise and execute the code.
Solution:

```
import soundfile as sf
from scipy import signal

#read .wav file
input_signal,fs = sf.read('./Sound_Noise.wav
    ')

#sampling frequency of Input signal
sampl_freq=fs

#order of the filter
order=4

#cutoff frquency 4kHz
cutoff_freq=4000.0

#digital frequency
Wn=2*cutoff_freq/sampl_freq

# b and a are numerator and denominator
    polynomials respectively
b, a = signal.butter(order,Wn, 'low')

#filter the input signal with butterworth filter
output_signal = signal.filtfilt(b, a,
    input_signal)

#write the output signal into .wav file
sf.write('Sound_With_ReducedNoise.wav',
    output_signal, fs)
```

2.4 The output of the python script in Problem 2.3 is the audio file Sound_With_ReducedNoise.wav. Play

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in. All content in the manuscript is released under GNU GPL. Free to use for anything.

the file in the spectrogram in Problem 2.2. What do you observe?
Solution: The key strokes as well as background noise is subdued in the audio. Also, the signal is blank for frequencies above 5.1 kHz.

3 Difference Equation

3.1 Let

$$x(n) = \left\{ \underset{\uparrow}{1}, 2, 3, 4, 2, 1 \right\} \tag{3.1}$$

Sketch $x(n)$.

3.2 Let

$$y(n) + \frac{1}{2}y(n-1) = x(n) + x(n-2), \quad y(n) = 0, n < 0 \tag{3.2}$$

Sketch $y(n)$.
Solution: The following code yields Fig. 3.2.

```
wget https://github.com/gadepall/EE1310/raw/
    master/filter/codes/xnyn.py
```

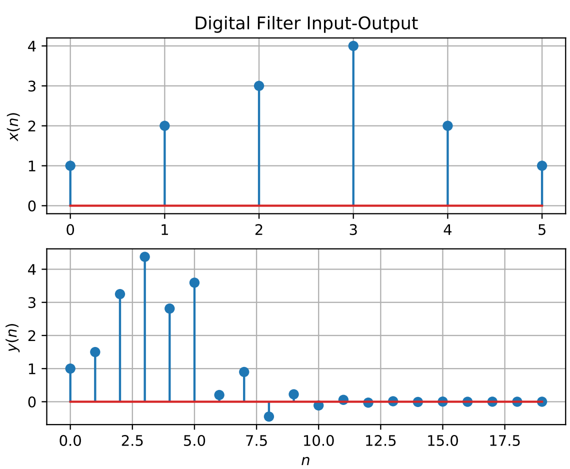


Fig. 3.2

3.3 Repeat the above exercise using a C code.
Solution:

```
wget https://github.com/TataSaiManoj/EE3900
    /blob/main/Assignment1/codes/toeplitz.c
```

4 Z - transform

4.1 The Z-transform of $x(n)$ is defined as

$$X(z) = \mathcal{Z}\{x(n)\} = \sum_{n=-\infty}^{\infty} x(n)z^{-n} \tag{4.1}$$

Show that

$$\mathcal{Z}\{x(n-1)\} = z^{-1}X(z) \tag{4.2}$$

and find

$$\mathcal{Z}\{x(n-k)\} \tag{4.3}$$

Solution: From (4.1),

$$\mathcal{Z}\{x(n-k)\} = \sum_{n=-\infty}^{\infty} x(n-k)z^{-n} \tag{4.4}$$

$$= \sum_{n=-\infty}^{\infty} x(n)z^{-n-k} = z^{-k} \sum_{n=-\infty}^{\infty} x(n)z^{-n} \tag{4.5}$$

$$\mathcal{Z}\{x(n-k)\} = z^{-k}X(z) \tag{4.6}$$

When $k = 1$, we get the equation

$$\mathcal{Z}\{x(n-1)\} = z^{-1}X(z) \tag{4.7}$$

4.2 Obtain $X(z)$ for $x(n)$ defined in problem 3.1.
Solution: From (4.1)

$$X(z) = \sum_{n=0}^{5} x(n)z^{-n} \tag{4.8}$$

$$= x(0) + x(1)z^{-1} + \cdots + x(5)z^{-5} \tag{4.9}$$

$$= 1 + 2z^{-1} + 3z^{-2} + 4z^{-3} + 2z^{-4} + z^{-5} \tag{4.10}$$

4.3 Find

$$H(z) = \frac{Y(z)}{X(z)} \tag{4.11}$$

from (3.2) assuming that the Z-transform is a linear operation.
Solution: Applying (4.6) in (3.2),

$$Y(z) + \frac{1}{2}z^{-1}Y(z) = X(z) + z^{-2}X(z) \tag{4.12}$$

$$\implies \frac{Y(z)}{X(z)} = \frac{1 + z^{-2}}{1 + \frac{1}{2}z^{-1}} \tag{4.13}$$

4.4 Find the Z transform of

$$\delta(n) = \begin{cases} 1 & n = 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.14}$$

and show that the Z-transform of

$$u(n) = \begin{cases} 1 & n \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.15}$$

is

$$U(z) = \frac{1}{1 - z^{-1}}, \quad |z| > 1 \tag{4.16}$$

Solution: It is easy to show that

$$\delta(n) \overset{\mathcal{Z}}{\rightleftharpoons} 1 \tag{4.17}$$

and from (4.15),

$$U(z) = \sum_{n=0}^{\infty} z^{-n} \tag{4.18}$$

$$= \frac{1}{1-z^{-1}}, \quad |z| > 1 \tag{4.19}$$

using the formula for the sum of an infinite geometric progression.

4.5 Show that

$$a^n u(n) \overset{\mathcal{Z}}{\rightleftharpoons} \frac{1}{1-az^{-1}} \quad |z| > |a| \tag{4.20}$$

Solution:

$$x(n) = a^n u(n) \tag{4.21}$$

$$\mathcal{Z}\{x(n)\} = \sum_{n=-\infty}^{\infty} a^n u(n) z^{-n} \tag{4.22}$$

$$\mathcal{Z}\{x(n)\} = \sum_{n=0}^{\infty} a^n z^{-n} \tag{4.23}$$

$$\mathcal{Z}\{x(n)\} = \sum_{n=0}^{\infty} (\frac{z}{a})^{-n} \tag{4.24}$$

$$\mathcal{Z}\{x(n)\} = \frac{1}{1-az^{-1}}, |z| > 1 \tag{4.25}$$

4.6 Let

$$H\left(e^{\jmath\omega}\right) = H\left(z = e^{\jmath\omega}\right). \tag{4.26}$$

Plot $|H\left(e^{\jmath\omega}\right)|$. Is it periodic? If so, find the period. $H(e^{\jmath\omega})$ is known as the *Discrete Time Fourier Transform* (DTFT) of $x(n)$.
Solution: The following code plots Fig. 4.6.

$$|H\left(e^{\jmath\omega}\right)| = \left|\frac{1+e^{-2\jmath\omega}}{1+\frac{1}{2}e^{-\jmath\omega}}\right| \tag{4.27}$$

$$= \sqrt{\frac{(1+\cos 2\omega)^2 + (\sin 2\omega)^2}{\left(1+\frac{1}{2}\cos\omega\right)^2 + \left(\frac{1}{2}\sin\omega\right)^2}} \tag{4.28}$$

$$= \sqrt{\frac{2\left(1+\cos 2\omega\right)}{\frac{5}{4}+\cos\omega}} \tag{4.29}$$

$$= \sqrt{\frac{2\left(2\cos^2\omega\right)}{\frac{5}{4}+\cos\omega}} \tag{4.30}$$

$$= \frac{4|\cos\omega|}{\sqrt{5+4\cos\omega}} \tag{4.31}$$

Thus,

$$H\left(e^{\jmath(\omega+2\pi)}\right) = \frac{4|\cos(\omega+2\pi)|}{\sqrt{5+4\cos(\omega+2\pi)}} \tag{4.32}$$

$$= \frac{4|\cos\omega|}{\sqrt{5+4\cos\omega}} \tag{4.33}$$

$$= H\left(e^{\jmath\omega}\right) \tag{4.34}$$

So, the fundamental period is 2π.

```
wget https://github.com/TataSaiManoj/EE3900
/blob/main/Assignment1/codes/dtft.py
```

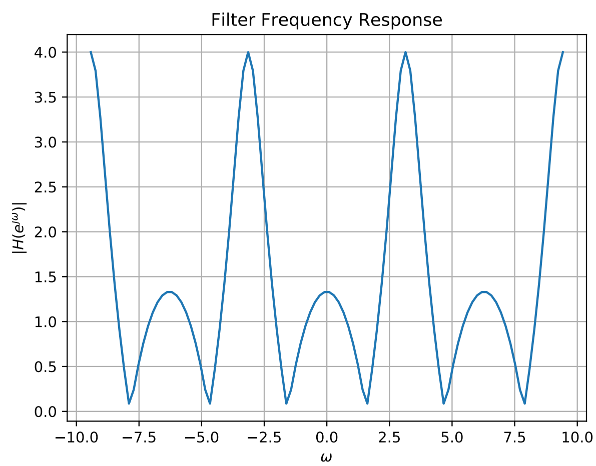


Fig. 4.6: $|H\left(e^{\jmath\omega}\right)|$

4.7 Express $h(n)$ in terms of $H\left(e^{\jmath\omega}\right)$.
Solution: We have,

$$H(e^{\jmath\omega}) = \sum_{k=-\infty}^{\infty} h(k)e^{-\jmath\omega k} \tag{4.35}$$

$$\int_{-\pi}^{\pi} e^{\jmath\omega(n-k)} d\omega = \begin{cases} 2\pi & n = k \\ 0 & \text{otherwise} \end{cases} \tag{4.36}$$

5 Impulse Response

5.1 Using long division, find

$$h(n), \quad n < 5 \tag{5.1}$$

for H(z) in (4.13).

Solution: Using the substitution $x := z^{-1}$, we perform long division.

$$
\begin{array}{r}
2x-4 \\
\frac{1}{2}x+1 \overline{\big) \; x^2 \qquad +1} \\
-x^2-2x \qquad \\
\hline
-2x+1 \\
2x+4 \\
\hline
5
\end{array}
$$

Thus,

$$H(z) = -4 + 2z^{-1} + \frac{5}{1+\frac{1}{2}z^{-1}} \tag{5.2}$$

$$= -4 + 2z^{-1} + 5\sum_{n=0}^{\infty}\left(-\frac{1}{2}\right)^n z^{-n} \tag{5.3}$$

$$= 1 - \frac{1}{2}z^{-1} + 5\sum_{n=2}^{\infty}\left(-\frac{1}{2}\right)^n z^{-n} \tag{5.4}$$

$$= \sum_{n=0}^{\infty}\left(-\frac{1}{2}\right)^n z^{-n} + 4\sum_{n=2}^{\infty}\left(-\frac{1}{2}\right)^n z^{-n} \tag{5.5}$$

$$= \sum_{n=-\infty}^{\infty} u(n)\left(-\frac{1}{2}\right)^n z^{-n} + \sum_{n=-\infty}^{\infty} u(n-2)\left(-\frac{1}{2}\right)^{n-2} z^{-n} \tag{5.6}$$

Therefore, from (4.1),

$$h(n) = \left(-\frac{1}{2}\right)^n u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2) \tag{5.7}$$

5.2 Find an expression for $h(n)$ using $H(z)$, given that

$$h(n) \overset{\mathcal{Z}}{\rightleftharpoons} H(z) \tag{5.8}$$

and there is a one to one relationship between $h(n)$ and $H(z)$. $h(n)$ is known as the *impulse response* of the system defined by (3.2).
Solution: From (4.13),

$$H(z) = \frac{1}{1+\frac{1}{2}z^{-1}} + \frac{z^{-2}}{1+\frac{1}{2}z^{-1}} \tag{5.9}$$

$$\implies h(n) = \left(-\frac{1}{2}\right)^n u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2) \tag{5.10}$$

using (4.20) and (4.6).

5.3 Sketch $h(n)$. Is it bounded? Justify theoretically.
Solution: The following code plots Fig. 5.3.

```
wget https://raw.githubusercontent.com/
    gadepall/EE1310/master/filter/codes/hn.py
```

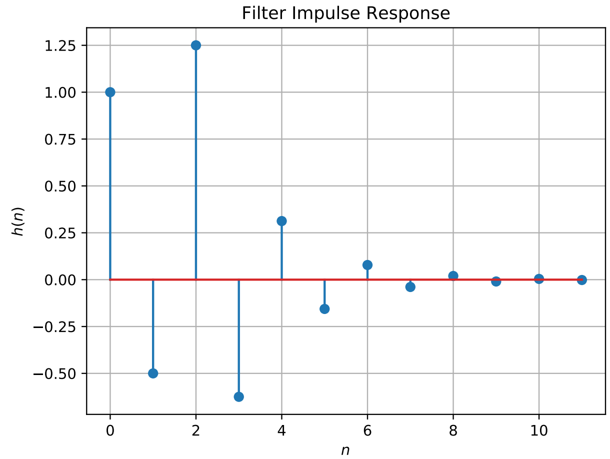


Fig. 5.3: $h(n)$ as the inverse of $H(z)$

From the Fig. 5.3 we can see that $h(n)$ is bounded.

5.4 Convergent? Justify using the ratio test.
Solution: For large n, we see that

$$h(n) = \left(-\frac{1}{2}\right)^n + \left(-\frac{1}{2}\right)^{n-2} \tag{5.11}$$

$$= \left(-\frac{1}{2}\right)^n (4+1) = 5\left(-\frac{1}{2}\right)^n \tag{5.12}$$

$$\implies \left|\frac{h(n+1)}{h(n)}\right| = \frac{1}{2} \tag{5.13}$$

and therefore, $\lim_{n\to\infty}\left|\frac{h(n+1)}{h(n)}\right| = \frac{1}{2} < 1$. Hence, we see that $h(n)$ converges.

5.5 The system with $h(n)$ is defined to be stable if

$$\sum_{n=-\infty}^{\infty} h(n) < \infty \tag{5.14}$$

Is the system defined by (3.2) stable for the impulse response in (5.8)?
Solution: Note that

$$\sum_{n=-\infty}^{\infty} h(n) = \sum_{n=-\infty}^{\infty}\left(-\frac{1}{2}\right)^n u(n) + \tag{5.15}$$

$$\left(-\frac{1}{2}\right)^{n-2} u(n-2) \tag{5.16}$$

$$= 2\left(\frac{1}{1+\frac{1}{2}}\right) = \frac{4}{3} \tag{5.17}$$

Thus, the given system is stable.

5.6 Verify the above result using a python code.

5.7 Compute and sketch $h(n)$ using

$$h(n) + \frac{1}{2}h(n-1) = \delta(n) + \delta(n-2), \quad (5.18)$$

This is the definition of $h(n)$.
Solution: The following code plots Fig. 5.7. Note that this is the same as Fig. 5.3.

```
wget https://raw.githubusercontent.com/
    gadepall/EE1310/master/filter/codes/hndef
    .py
```

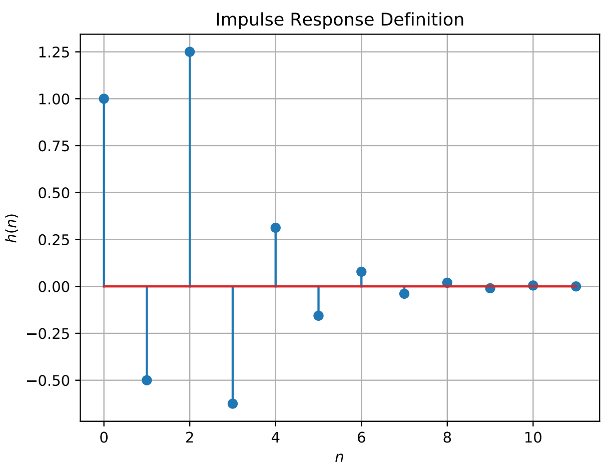


Fig. 5.7: $h(n)$ from the definition

5.8 Compute

$$y(n) = x(n) * h(n) = \sum_{n=-\infty}^{\infty} x(k)h(n-k) \quad (5.19)$$

Comment. The operation in (5.19) is known as *convolution*.
Solution: The following code plots Fig. 5.8. Note that this is the same as $y(n)$ in Fig. 3.2.

```
wget https://raw.githubusercontent.com/
    gadepall/EE1310/master/filter/codes/
    ynconv.py
```

5.9 Express the above convolution using a Toeplitz matrix.
Solution: We use Toeplitz matrices for convolution

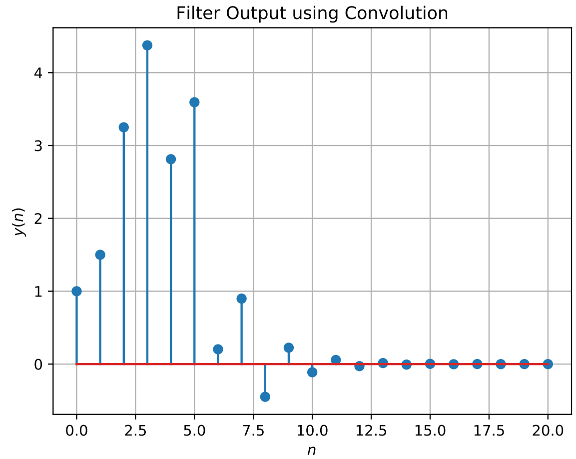


Fig. 5.8: $y(n)$ from the definition of convolution

$$\mathbf{y} = \mathbf{x} \circledast \mathbf{h} \quad (5.20)$$

$$\mathbf{y} = \begin{pmatrix} h_1 & 0 & . & . & . & 0 \\ h_2 & h_1 & . & . & . & 0 \\ h_3 & h_2 & h_1 & . & . & 0 \\ . & . & . & . & . & . \\ 0 & . & . & h_3 & h_2 & h_1 \\ 0 & . & . & . & h_2 & h_1 \\ 0 & . & . & . & 0 & h_1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad (5.21)$$

5.10 Show that

$$y(n) = \sum_{n=-\infty}^{\infty} x(n-k)h(k) \quad (5.22)$$

Solution: From (5.19), we substitute $k := n-k$ to get

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)\, h(n-k) \quad (5.23)$$

$$= \sum_{n-k=-\infty}^{\infty} x(n-k)\, h(k) \quad (5.24)$$

$$= \sum_{k=-\infty}^{\infty} x(n-k)\, h(k) \quad (5.25)$$

6 DFT

6.1 Compute

$$X(k) \triangleq \sum_{n=0}^{N-1} x(n) e^{-\jmath 2\pi kn/N}, \quad k = 0, 1, \ldots, N-1 \quad (6.1)$$

and $H(k)$ using $h(n)$.
Solution: Run the following codes to compute $X(k)$ and $H(k)$.

```
wget https://raw.githubusercontent.com/
    TataSaiManoj/EE3900/blob/main/
    Assignment1/codes/6_1_X(k).py
```

```
wget https://raw.githubusercontent.com/
    TataSaiManoj/EE3900/blob/main/
    Assignment1/codes/6_1_H(k).py
```

6.2 Compute

$$Y(k) = X(k)H(k) \tag{6.2}$$

Solution: Run the following code to compute $Y(k)$.

```
wget https://raw.githubusercontent.com/
    TataSaiManoj/EE3900/blob/main/
    Assignment1/codes/6_2.py
```

6.3 Compute

$$y(n) = \frac{1}{N}\sum_{k=0}^{N-1} Y(k)\cdot e^{j2\pi kn/N}, \quad n = 0, 1, \ldots, N-1 \tag{6.3}$$

Solution: The following code plots Fig. 5.8. Note that this is the same as $y(n)$ in Fig. 3.2.

```
wget https://raw.githubusercontent.com/
    TataSaiManoj/EE3900/blob/main/
    Assignment1/codes/6_3_yndft.py
```

6.4 Repeat the previous exercise by computing $X(k), H(k)$ and $y(n)$ through FFT and IFFT.
Solution:

```
wget https://raw.githubusercontent.com/
    TataSaiManoj/EE3900/blob/main/
    Assignment1/codes/6_4.py
```

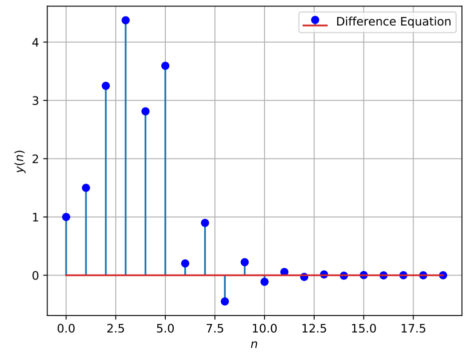


Fig. 6.4: From the Difference Equation

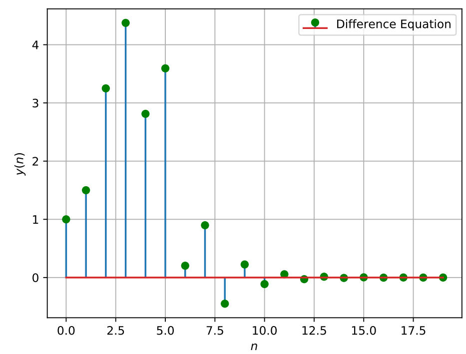


Fig. 6.4: From the IDFT

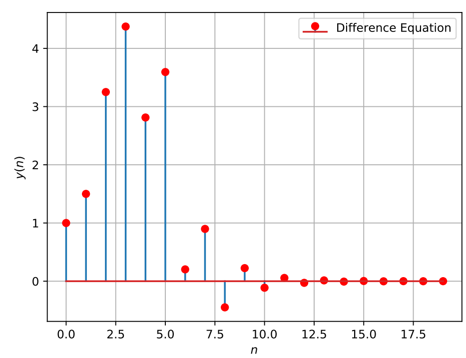


Fig. 6.4: From the IFFT

7 FFT

7.1 The DFT of $x(n)$ is given by

$$X(k) \triangleq \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}, \quad k = 0, 1, \ldots, N-1 \tag{7.1}$$

7.2 Let

$$W_N = e^{-j2\pi/N} \tag{7.2}$$

Then the N-point *DFT matrix* is defined as

$$\vec{F}_N = \left[W_N^{mn}\right], \quad 0 \le m, n \le N-1 \tag{7.3}$$

where W_N^{mn} are the elements of $\vec{F}_N$.

7.3 Let

$$\vec{I}_4 = \begin{pmatrix} \vec{e}_4^1 & \vec{e}_4^2 & \vec{e}_4^3 & \vec{e}_4^4 \end{pmatrix} \tag{7.4}$$

be the 4×4 identity matrix. Then the 4 point *DFT permutation matrix* is defined as

$$\vec{P}_4 = \begin{pmatrix} \vec{e}_4^1 & \vec{e}_4^3 & \vec{e}_4^2 & \vec{e}_4^4 \end{pmatrix} \tag{7.5}$$

7.4 The 4 point *DFT diagonal matrix* is defined as

$$\vec{D}_4 = diag\begin{pmatrix} W_8^0 & W_8^1 & W_8^2 & W_8^3 \end{pmatrix} \tag{7.6}$$

7.5 Show that

$$W_N^2 = W_{N/2} \tag{7.7}$$

Solution:

$$W_N^2 = \left(e^{(-j \cdot 2\pi N)}\right)^2 \tag{7.8}$$

$$= e^{-2\pi jN \cdot 2} \tag{7.9}$$

$$= e^{-j \cdot \left(\frac{2\pi}{N/2}\right)} \tag{7.10}$$

$$= W_{N/2} \tag{7.11}$$

7.6 Show that

$$\vec{F}_4 = \begin{bmatrix} \vec{I}_2 & \vec{D}_2 \\ \vec{I}_2 & -\vec{D}_2 \end{bmatrix} \begin{bmatrix} \vec{F}_2 & 0 \\ 0 & \vec{F}_2 \end{bmatrix} \vec{P}_4 \tag{7.12}$$

Solution:

$$\begin{bmatrix} \vec{I}_2 & \vec{D}_2 \\ \vec{I}_2 & -\vec{D}_2 \end{bmatrix} \begin{bmatrix} \vec{F}_2 & 0 \\ 0 & \vec{F}_2 \end{bmatrix} \tag{7.13}$$

$$= \begin{bmatrix} \vec{F}_2 & \vec{D}_2\vec{F}_2 \\ \vec{F}_2 & -\vec{D}_2\vec{F}_2 \end{bmatrix} \tag{7.14}$$

$$= \begin{bmatrix} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} & \begin{pmatrix} 1 & 0 \\ 0 & -j \end{pmatrix}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \\ \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} & -\begin{pmatrix} 1 & 0 \\ 0 & -j \end{pmatrix}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \end{bmatrix} \tag{7.15}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & -j & j \\ 1 & 1 & -1 & -1 \\ 1 & -1 & j & -j \end{bmatrix} \tag{7.16}$$

$$\vec{P}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7.17}$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & -j & j \\ 1 & 1 & -1 & -1 \\ 1 & -1 & j & -j \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7.18}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -\jmath & -1 & \jmath \\ 1 & -1 & 1 & -1 \\ 1 & \jmath & -1 & -\jmath \end{bmatrix} \tag{7.19}$$

$$= \begin{bmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^4 & W_4^6 \\ W_4^0 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix} \tag{7.20}$$

$$= \vec{F}_4 \tag{7.21}$$

7.7 Show that

$$\vec{F}_N = \begin{bmatrix} \vec{I}_{N/2} & \vec{D}_{N/2} \\ \vec{I}_{N/2} & -\vec{D}_{N/2} \end{bmatrix} \begin{bmatrix} \vec{F}_{N/2} & 0 \\ 0 & \vec{F}_{N/2} \end{bmatrix} \vec{P}_N \tag{7.22}$$

Solution:

$$\begin{bmatrix} \vec{I}_{N/2} & \vec{D}_{N/2} \\ \vec{I}_{N/2} & -\vec{D}_{N/2} \end{bmatrix} \begin{bmatrix} \vec{F}_{N/2} & 0 \\ 0 & \vec{F}_{N/2} \end{bmatrix} \tag{7.23}$$

$$= \begin{bmatrix} \vec{F}_{N/2} & \vec{D}_{N/2}\vec{F}_{N/2} \\ \vec{F}_{N/2} & -\vec{D}_{N/2}\vec{F}_{N/2} \end{bmatrix} \tag{7.24}$$

$$\vec{D}_{N/2} = \begin{bmatrix} W_N & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & W_N^{N/2+1} \end{bmatrix} \tag{7.25}$$

$$\vec{F}_{N/2} = \begin{bmatrix} W_N^0 & \cdots & W_N^0 \\ \vdots & \ddots & \vdots \\ W_{N/2}^{N/2-1} W_N^0 & \cdots & W_{N/2}^{(N/2-1)^2} \end{bmatrix} \tag{7.26}$$

Thus

$$\left(\vec{D}_{N/2}\vec{F}_{N/2}\right)_{ij} = W_N^i W_{N/2}^{ij} \tag{7.27}$$

$$= W_N^i W_N^{2\,ij} \tag{7.28}$$

$$= W_N^{i(2j+1)} \tag{7.29}$$

where $i, j = 0, \dots, N/2-1$

$\vec{D}_{N/2}\vec{F}_{N/2}$ forms the first $N/2$ rows of the odd columns of $\vec{F}_N$

$$W_N^{(i+N/2)(2j+1)} = e^{\left(-j\frac{2\pi}{N}(2j+1)(i+\frac{N}{2})\right)} \tag{7.30}$$

$$= W_N^{i(2j+1)} \tag{7.31}$$

The remaining $N/2$ rows of $\vec{F}_N$ are the negative of the first $N/2$ rows of $\vec{F}_N$.
Now, for the even-indexed columns, we have

$\vec{F}_{N/2}$ forms the first $N/2$ rows of the even columns of $\vec{F}_N$. The remaining $N/2$ rows of $\vec{F}_N$ are the same as the first $N/2$ rows of $\vec{F}_N$. Therefore

$$\begin{bmatrix} \vec{F}_{N/2} & \vec{D}_{N/2}\vec{F}_{N/2} \\ \vec{F}_{N/2} & -\vec{D}_{N/2}\vec{F}_{N/2} \end{bmatrix} = \vec{F}_N\vec{P}_N \tag{7.32}$$

where

$$\vec{P}_N = \begin{pmatrix} \vec{e}_N^1 & \vec{e}_N^3 & \cdots & \vec{e}_N^{N-1} & \vec{e}_N^2 & \vec{e}_N^4 & \cdots & \vec{e}_N^N \end{pmatrix} \tag{7.33}$$

Hence

$$\begin{bmatrix} \vec{F}_{N/2} & \vec{D}_{N/2}\vec{F}_{N/2} \\ \vec{F}_{N/2} & -\vec{D}_{N/2}\vec{F}_{N/2} \end{bmatrix}\vec{P}_N = \vec{F}_N\vec{P}_N^2 = \vec{F}_N \tag{7.34}$$

$$\vec{F}_N = \begin{bmatrix} \vec{I}_{N/2} & \vec{D}_{N/2} \\ \vec{I}_{N/2} & -\vec{D}_{N/2} \end{bmatrix}\begin{bmatrix} \vec{F}_{N/2} & 0 \\ 0 & \vec{F}_{N/2} \end{bmatrix}\vec{P}_N \tag{7.35}$$

where N is a power of 2.

7.8 Find

$$\vec{P}_4\vec{x} \tag{7.36}$$

Solution:
Let $\vec{x} = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \end{pmatrix}$

$$\vec{P}_4\vec{x} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{7.37}$$

$$= \begin{bmatrix} x_1 \\ x_3 \\ x_2 \\ x_4 \end{bmatrix} \tag{7.38}$$

7.9 Show that

$$\vec{X} = \vec{F}_N\vec{x} \tag{7.39}$$

where $\vec{x}, \vec{X}$ are the vector representations of $x(n), X(k)$ respectively.

Solution:

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N} \tag{7.40}$$

$$\vec{X} = \begin{bmatrix} \sum_{n=0}^{N-1} x(n)e^{-j2\pi n(0)/N} \\ \vdots \\ \sum_{n=0}^{N-1} x(n)e^{-j2\pi n(N-1)/N} \end{bmatrix} \tag{7.41}$$

$$= \begin{bmatrix} x(0) + \cdots + x(N-1) \\ \vdots \\ x(0) + \cdots + x(N-1)e^{-j2\pi(N-1)^2/N} \end{bmatrix} \tag{7.42}$$

$$\vec{X} = x(0)\begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} + \cdots + x(N-1)\begin{bmatrix} 1 \\ \vdots \\ e^{-j2\pi(N-1)^2/N} \end{bmatrix} \tag{7.43}$$

$$= \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & e^{-j2\pi(N-1)^2/N} \end{bmatrix}\begin{bmatrix} x(0) \\ \vdots \\ x(N-1) \end{bmatrix} \tag{7.44}$$

$$= \vec{F}_N\vec{x} \tag{7.45}$$

7.10 Derive the following Step-by-step visualisation of 8-point FFTs into 4-point FFTs and so on

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} X_1(0) \\ X_1(1) \\ X_1(2) \\ X_1(3) \end{bmatrix} + \begin{bmatrix} W_8^0 & 0 & 0 & 0 \\ 0 & W_8^1 & 0 & 0 \\ 0 & 0 & W_8^2 & 0 \\ 0 & 0 & 0 & W_8^3 \end{bmatrix}\begin{bmatrix} X_2(0) \\ X_2(1) \\ X_2(2) \\ X_2(3) \end{bmatrix} \tag{7.46}$$

$$\begin{bmatrix} X(4) \\ X(5) \\ X(6) \\ X(7) \end{bmatrix} = \begin{bmatrix} X_1(0) \\ X_1(1) \\ X_1(2) \\ X_1(3) \end{bmatrix} - \begin{bmatrix} W_8^0 & 0 & 0 & 0 \\ 0 & W_8^1 & 0 & 0 \\ 0 & 0 & W_8^2 & 0 \\ 0 & 0 & 0 & W_8^3 \end{bmatrix}\begin{bmatrix} X_2(0) \\ X_2(1) \\ X_2(2) \\ X_2(3) \end{bmatrix} \tag{7.47}$$

4-point FFTs into 2-point FFTs

$$\begin{bmatrix} X_1(0) \\ X_1(1) \end{bmatrix} = \begin{bmatrix} X_3(0) \\ X_3(1) \end{bmatrix} + \begin{bmatrix} W_4^0 & 0 \\ 0 & W_4^1 \end{bmatrix}\begin{bmatrix} X_4(0) \\ X_4(1) \end{bmatrix} \tag{7.48}$$

$$\begin{bmatrix} X_1(2) \\ X_1(3) \end{bmatrix} = \begin{bmatrix} X_3(0) \\ X_3(1) \end{bmatrix} - \begin{bmatrix} W_4^0 & 0 \\ 0 & W_4^1 \end{bmatrix}\begin{bmatrix} X_4(0) \\ X_4(1) \end{bmatrix} \tag{7.49}$$

$$\begin{bmatrix} X_2(0) \\ X_2(1) \end{bmatrix} = \begin{bmatrix} X_5(0) \\ X_5(1) \end{bmatrix} + \begin{bmatrix} W_4^0 & 0 \\ 0 & W_4^1 \end{bmatrix}\begin{bmatrix} X_6(0) \\ X_6(1) \end{bmatrix} \tag{7.50}$$

$$\begin{bmatrix} X_2(2) \\ X_2(3) \end{bmatrix} = \begin{bmatrix} X_5(0) \\ X_5(1) \end{bmatrix} - \begin{bmatrix} W_4^0 & 0 \\ 0 & W_4^1 \end{bmatrix}\begin{bmatrix} X_6(0) \\ X_6(1) \end{bmatrix} \tag{7.51}$$

$$P_8 \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix} = \begin{bmatrix} x(0) \\ x(2) \\ x(4) \\ x(6) \\ x(1) \\ x(3) \\ x(5) \\ x(7) \end{bmatrix} \tag{7.52}$$

$$P_4 \begin{bmatrix} x(0) \\ x(2) \\ x(4) \\ x(6) \end{bmatrix} = \begin{bmatrix} x(0) \\ x(4) \\ x(2) \\ x(6) \end{bmatrix} \tag{7.53}$$

$$P_4 \begin{bmatrix} x(1) \\ x(3) \\ x(5) \\ x(7) \end{bmatrix} = \begin{bmatrix} x(1) \\ x(5) \\ x(3) \\ x(7) \end{bmatrix} \tag{7.54}$$

Therefore,

$$\begin{bmatrix} X_3(0) \\ X_3(1) \end{bmatrix} = F_2 \begin{bmatrix} x(0) \\ x(4) \end{bmatrix} \tag{7.55}$$

$$\begin{bmatrix} X_4(0) \\ X_4(1) \end{bmatrix} = F_2 \begin{bmatrix} x(2) \\ x(6) \end{bmatrix} \tag{7.56}$$

$$\begin{bmatrix} X_5(0) \\ X_5(1) \end{bmatrix} = F_2 \begin{bmatrix} x(1) \\ x(5) \end{bmatrix} \tag{7.57}$$

$$\begin{bmatrix} X_6(0) \\ X_6(1) \end{bmatrix} = F_2 \begin{bmatrix} x(3) \\ x(7) \end{bmatrix} \tag{7.58}$$

Solution:
We write out the values of performing an 8-point FFT on $\vec{x}$ as follows.

$$X(k) = \sum_{n=0}^{7} x(n) e^{-\frac{j2kn\pi}{8}} \tag{7.59}$$

$$= \sum_{n=0}^{3} \left(x(2n) e^{-\frac{j2kn\pi}{4}} + e^{-\frac{j2k\pi}{8}} x(2n+1) e^{-\frac{j2kn\pi}{4}} \right) \tag{7.60}$$

$$= X_1(k) + e^{-\frac{j2k\pi}{4}} X_2(k) \tag{7.61}$$

where $\vec{X}_1$ is the 4-point FFT of the even-numbered terms and $\vec{X}_2$ is the 4-point FFT of the odd numbered terms. Noticing that for $k \geq 4$,

$$X_1(k) = X_1(k-4) \tag{7.62}$$

$$e^{-\frac{j2k\pi}{8}} = -e^{-\frac{j2(k-4)\pi}{8}} \tag{7.63}$$

We can now write out $X(k)$ in matrix form. We also need to solve the two 4-point FFT terms so formed.

$$X_1(k) = \sum_{n=0}^{3} x_1(n) e^{-\frac{j2kn\pi}{8}} \tag{7.64}$$

$$= \sum_{n=0}^{1} \left(x_1(2n) e^{-\frac{j2kn\pi}{4}} \right) \tag{7.65}$$

$$+ \sum_{n=0}^{1} \left(e^{-\frac{j2k\pi}{8}} x_2(2n+1) e^{-\frac{j2kn\pi}{4}} \right) \tag{7.66}$$

$$= X_3(k) + e^{-\frac{j2k\pi}{4}} X_4(k) \tag{7.67}$$

using $x_1(n) = x(2n)$ and $x_2(n) = x(2n+1)$. Thus we can write the 2-point FFTs

$$\begin{bmatrix} X_3(0) \\ X_3(1) \end{bmatrix} = F_2 \begin{bmatrix} x(0) \\ x(4) \end{bmatrix} \tag{7.68}$$

$$\begin{bmatrix} X_4(0) \\ X_4(1) \end{bmatrix} = F_2 \begin{bmatrix} x(2) \\ x(6) \end{bmatrix} \tag{7.69}$$

Using a similar idea for the terms X_2,

$$\begin{bmatrix} X_5(0) \\ X_5(1) \end{bmatrix} = F_2 \begin{bmatrix} x(1) \\ x(5) \end{bmatrix} \tag{7.70}$$

$$\begin{bmatrix} X_6(0) \\ X_6(1) \end{bmatrix} = F_2 \begin{bmatrix} x(3) \\ x(7) \end{bmatrix} \tag{7.71}$$

$$\vec{P}_8 \vec{x} = \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \end{pmatrix} \tag{7.72}$$

$$\vec{P}_4 \vec{x}_1 = \begin{pmatrix} \vec{x}_3 \\ \vec{x}_4 \end{pmatrix} \tag{7.73}$$

$$\vec{P}_4 \vec{x}_2 = \begin{pmatrix} \vec{x}_5 \\ \vec{x}_6 \end{pmatrix} \tag{7.74}$$

where we define $x_3(k) = x(4k)$, $x_4(k) = x(4k+2)$, $x_5(k) = x(4k+1)$, and $x_6(k) = x(4k+3)$ for $k = 0, 1$.

7.11 For

$$\vec{x} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 2 \\ 1 \end{pmatrix} \tag{7.75}$$

compte the DFT using (7.39)

Solution:
Download the following Python code that plots Fig. 7.11.

7.11.py

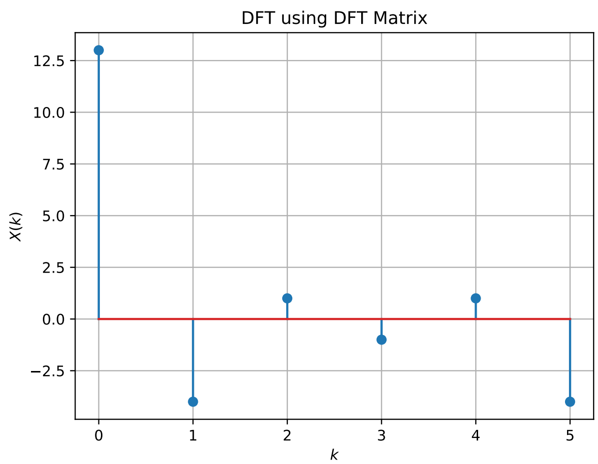


Fig. 7.11: Plot using DFT matrix

7.12 Repeat the above exercise using the FFT after zero padding $\vec{x}$.
Solution: Download the following Python code that plots Fig. 7.12.

7.12.py

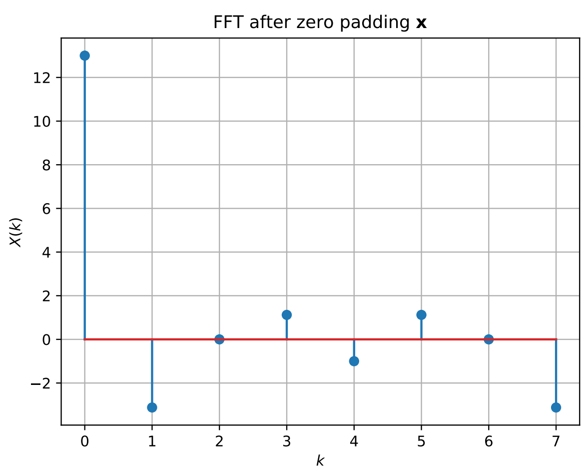


Fig. 7.12: Plot of FFT with zero padding

7.13 Write a C program to compute the 8-point FFT.
Solution:
Download the following C code that generates the values of $X(k)$ using 8-point FFT

8pointFFT.c

Download the following Python code that generates the plot of 8-point FFT from the C code

7_13.py

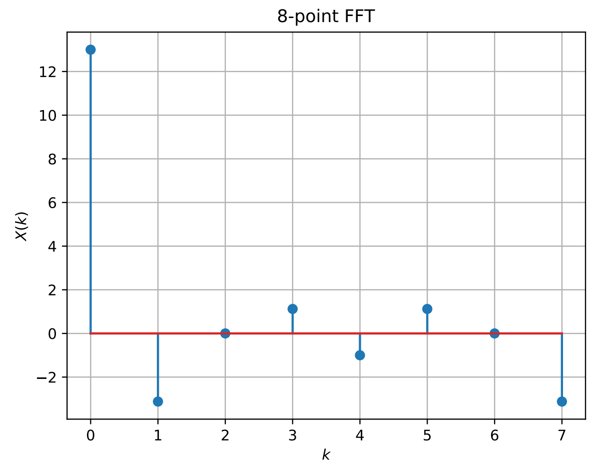
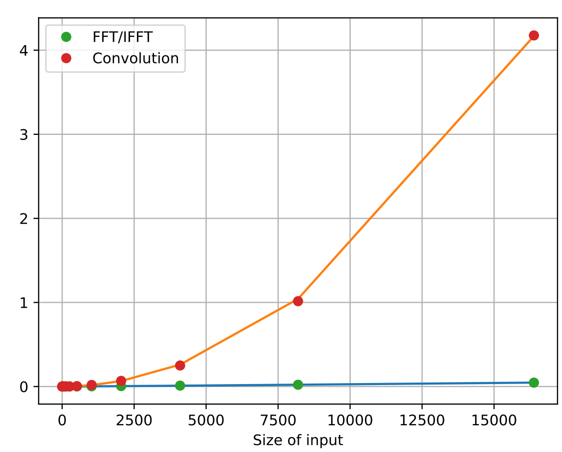


Fig. 7.13: Plot of FFT using C

7.14 Time complexity comparison of FFT and convolution
Solution:
The following C code notes the running times of FFT and convolution and the python code plots the results.

8pointFFT_times.c
7.14.py



8 Exercises

Answer the following questions by looking at the python code in Problem 2.3

8.1 The command

```
output_signal = signal.lfilter(b, a,
    input_signal)
```

in Problem 2.3 is executed through the following difference equation

$$\sum_{m=0}^{M} a(m)\, y(n-m) = \sum_{k=0}^{N} b(k)\, x(n-k) \quad (8.1)$$

where the input signal is $x(n)$ and the output signal is $y(n)$ with initial values all 0. Replace **signal.filtfilt** with your own routine and verify.
Solution: On taking the Z-transform on both sides of the difference equation

$$\sum_{m=0}^{M} a(m)\, z^{-m} Y(z) = \sum_{k=0}^{N} b(k)\, z^{-k} X(z) \quad (8.2)$$

$$\implies H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{N} b(k)\, z^{-k}}{\sum_{m=0}^{M} a(m)\, z^{-m}} \quad (8.3)$$

For obtaining the discrete Fourier transform, put $z = j\frac{2\pi i}{I}$ where I is the length of the input signal and $i = 0, 1, \ldots, I-1$
Download the following Python code

```
8_1.py
```

8.2 Repeat all the exercises in the previous sections for the above a and b
Solution: Using the previous routine, we can find the output signal using FFT/IFFT and also plot $h(n)$ using the following Python code

```
8_2.py
```

8.3 What is the sampling frequency of the input signal?
Solution: The sampling frequency of the input signal is 44.1 kHz

8.4 What is the type, order and cutoff frequency of the above Butterworth filter?
Solution:
It is a low-pass filter of order 4. The cutoff frequency is 4000Hz.

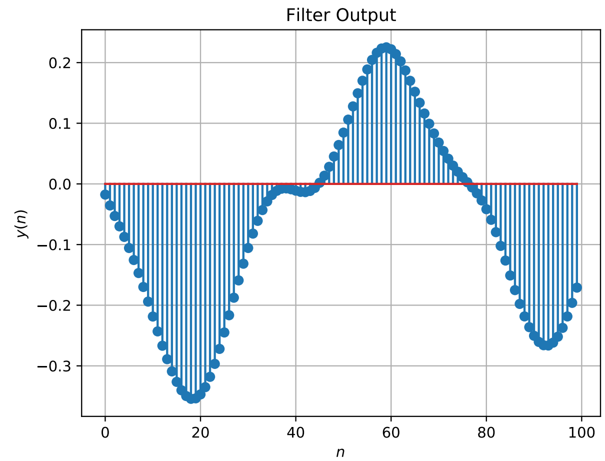


Fig. 8.2: Plot of $y(n)$

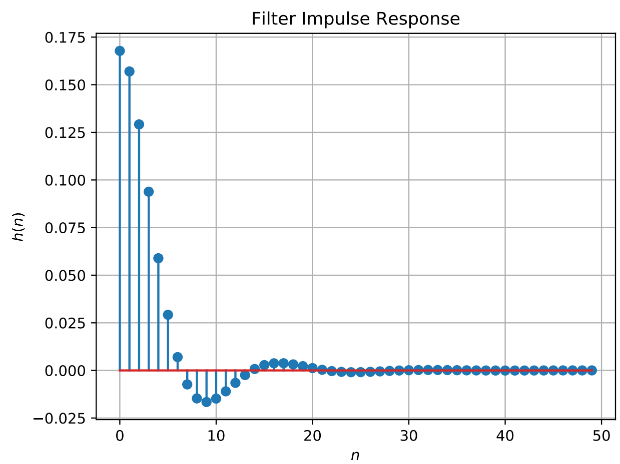


Fig. 8.2: Plot of $h(n)$

8.5 Modify the code with different input parameters to get the best possible output.
Solution:
Best possible output is obtained when order is 10 and cutoff frequency of 3000Hz.