

Experiment-6

AIM: Perform FFT and IFFT on discrete time signal.

Theory:

The Fast Fourier Transform (FFT) is an algorithm used to efficiently compute the Discrete Fourier Transform (DFT) of a sequence or signal. It converts a time-domain signal into its frequency-domain representation, revealing the signal's frequency components.

The Inverse Fast Fourier Transform (IFFT) is the reverse process of the FFT. It transforms the frequency-domain representation back into the time-domain signal.

The FFT and IFFT are widely used in various applications, including signal processing, image processing, audio processing, and communication systems.

Program:

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

signal = np.array([1, 2, 3, 4, 5, 6, 7, 8])

# Compute the FFT of the signal
fft_result = np.fft.fft(signal)

# Compute the magnitude and Phase spectrum of the FFT result
magnitude_spectrum = np.abs(fft_result)
phase_spectrum = np.angle(fft_result)

# Compute the IFFT of the FFT result
reconstructed_signal = np.fft.ifft(fft_result)

# Create frequency axis for plotting
N = len(signal)
frequencies = np.arange(N)

# Plot the results in a 2x2 subplot
plt.figure(figsize=(12, 10))

# Original Signal
plt.subplot(2, 2, 1)
plt.stem(frequencies, signal, basefmt='b-')
plt.title('Original Signal')
plt.xlabel('Sample Index')
plt.ylabel('Amplitude')
plt.grid(True)

# Magnitude Spectrum
plt.subplot(2, 2, 2)
plt.stem(frequencies, magnitude_spectrum, basefmt='r-')
plt.title('Magnitude Spectrum')
plt.xlabel('Frequency Bin')
plt.ylabel('Magnitude')
plt.grid(True)

# Phase Spectrum
plt.subplot(2, 2, 3)
```

```

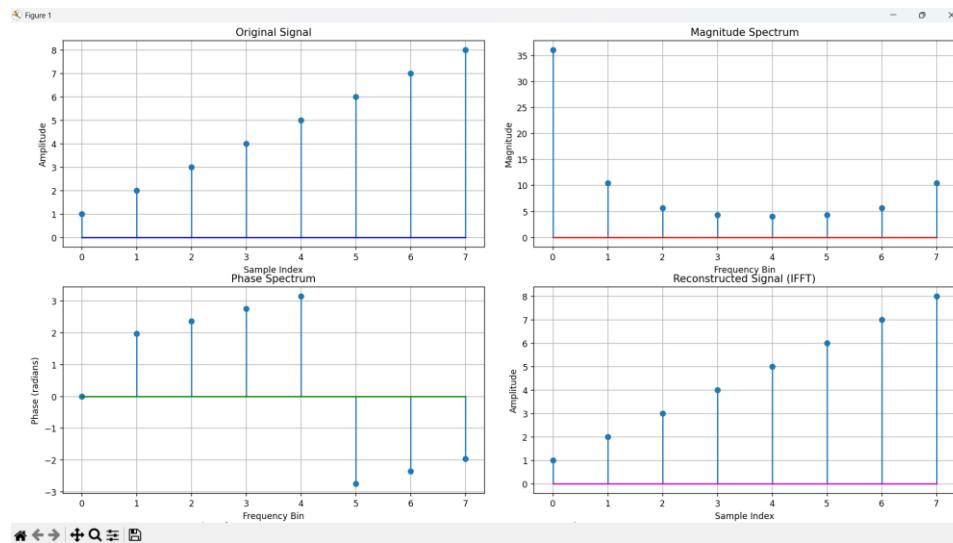
plt.stem(frequencies, phase_spectrum, basefmt='g-')
plt.title('Phase Spectrum')
plt.xlabel('Frequency Bin')
plt.ylabel('Phase (radians)')
plt.grid(True)

# Reconstructed Signal (Real part)
plt.subplot(2, 2, 4)
plt.stem(frequencies, np.real(reconstructed_signal), basefmt='m-')
plt.title('Reconstructed Signal (IFFT)')
plt.xlabel('Sample Index')
plt.ylabel('Amplitude')
plt.grid(True)

plt.tight_layout()
plt.show()

# Display numerical results
print("Original Signal:", signal)
print("FFT Result:", fft_result)
print("Magnitude Spectrum:", magnitude_spectrum)
print("Phase Spectrum:", phase_spectrum)
print("Reconstructed Signal (Real):", np.real(reconstructed_signal))
print("Reconstruction Error:", np.sum(np.abs(signal - np.real(reconstructed_signal))))
    
```

Output:



Conclusion:

- The FFT analysis demonstrates perfect signal reconstruction through the FFT-IFFT process.
- The magnitude spectrum shows the frequency content of the signal.
- The phase spectrum indicates the phase relationships between frequency components.
- The reconstructed signal matches the original, confirming the reversibility of the FFT operation.