

```

clc; clear; close all;

% Input sequence (length must be power of 2)
x = [1 2 3 4 0 0 0 0];

% Call FFT function
X = radix2_fft(x);

% Display result
disp("FFT Output:");
disp(X);

% Plot Magnitude and Phase
N = length(x);

figure;

subplot(2,1,1);
stem(0:N-1, abs(X), 'filled');
title('Magnitude Spectrum');
xlabel('Frequency Index');
ylabel('|X(k)|');
grid on;

subplot(2,1,2);
stem(0:N-1, angle(X), 'filled');
title('Phase Spectrum');
xlabel('Frequency Index');
ylabel('Phase (radians)');
grid on;

sgtitle('Radix-2 FFT Result');

% ----- FUNCTION -----
function X = radix2_fft(x)

N = length(x);

% Check if N is a power of 2
if bitand(N, N-1) ~= 0

```

```

        error('Input length must be a power of 2');
end

% Base case
if N == 1
    X = x;
    return;
end

% Divide: even and odd elements
X_even = radix2_fft(x(1:2:end));
X_odd  = radix2_fft(x(2:2:end));

% Allocate memory
X = zeros(1, N);

% Combine (Butterfly Operation)
for k = 0:(N/2 - 1)
    W = exp(-1i * 2 * pi * k / N) * X_odd(k+1);

    X(k+1) = X_even(k+1) + W;
    X(k + N/2 + 1) = X_even(k+1) - W;
end

end

```