Design and Analysis of Null Filter

Syed Nazmus Sakib
ID: AE-172-009

January 20, 2026

Abstract

In this experiment, we explored the design and implementation of a discrete-time Linear Time-Invariant (LTI) system known as a "Null Filter." The primary objective was to remove a specific interference frequency (50 Hz) from a composite signal sampled at 1 kHz, which also contained a desired signal component at 100 Hz. By utilizing Z-transform theory, we calculated the necessary filter coefficients by placing zeros on the unit circle at the precise angle corresponding to the noise frequency. The system was simulated in MATLAB. The results demonstrated successful suppression of the 50 Hz noise; however, a detailed analysis of the output revealed a significant attenuation of the desired 100 Hz signal, highlighting the inherent selectivity limitations of low-order FIR filters.

Introduction and Objective

Digital filters are fundamental components in signal processing, used to enhance signals by suppressing unwanted components. A "Null Filter" (or Notch Filter) is a specific band-stop filter designed to reject one specific frequency while attempting to pass all others.

In this scenario, we simulate a common real-world problem: Power Line Interference. A useful signal (100 Hz) is contaminated by low-frequency hum (50 Hz). The specific objectives are:

- To derive the mathematical Transfer Function $H(z)$ for a filter that nulls 50 Hz.
- To verify the filter's operation using Frequency Domain analysis (FFT) in MATLAB.
- To analyze the impact of the filter on the desired signal component.

Theoretical Background and Design

Digital Frequency Calculation

The design process begins by converting the analog noise frequency into the discrete angular frequency domain.

- Sampling Frequency (F_s): 1000 Hz
- Noise Frequency (f_{noise}): 50 Hz

The normalized angular frequency ω_0 is calculated as:

$$\omega_0 = 2\pi \frac{f_{noise}}{F_s} \tag{1}$$

$$\omega_0 = 2\pi \frac{50}{1000} = \frac{100\pi}{1000} = \frac{\pi}{10} \text{ radians/sample} \tag{2}$$

Z-Plane Design (Placement of Zeros)

In the Z-domain, the magnitude of the frequency response $|H(e^{j\omega})|$ becomes zero if we place "Zeros" (roots of the numerator) exactly on the unit circle at the angle ω_0. Since the impulse response must be real-valued, the zeros must appear in complex conjugate pairs. Therefore, we place zeros at:

$$z_1 = e^{j\frac{\pi}{10}} \quad \text{and} \quad z_2 = e^{-j\frac{\pi}{10}} \tag{3}$$

Derivation of Transfer Function $H(z)$

The transfer function of an FIR filter is the product of its zero factors:

$$H(z) = (1 - z_1 z^{-1})(1 - z_2 z^{-1}) \tag{4}$$

$$H(z) = (1 - e^{j\frac{\pi}{10}} z^{-1})(1 - e^{-j\frac{\pi}{10}} z^{-1}) \tag{5}$$

Expanding this polynomial:

$$H(z) = 1 - (e^{j\frac{\pi}{10}} + e^{-j\frac{\pi}{10}})z^{-1} + (e^{j\frac{\pi}{10}} \cdot e^{-j\frac{\pi}{10}})z^{-2} \tag{6}$$

Using Euler's Identity ($e^{jx} + e^{-jx} = 2\cos(x)$) and ($e^{jx} \cdot e^{-jx} = e^0 = 1$):

$$H(z) = 1 - 2\cos\left(\frac{\pi}{10}\right) z^{-1} + z^{-2} \tag{7}$$

We calculate the coefficient $2\cos(18°)$:

$$2\cos(18°) \approx 1.902113 \tag{8}$$

Thus, the final system transfer function is:

$$H(z) = 1 - 1.902113z^{-1} + z^{-2} \tag{9}$$

This corresponds to the time-domain difference equation:

$$y[n] = x[n] - 1.902113x[n-1] + x[n-2] \tag{10}$$

Methodology (MATLAB Simulation)

The experiment was conducted using MATLAB to simulate the signals and the filtering process. The steps were as follows:

1. **Signal Generation:** A composite time-domain signal s was created by summing a 100 Hz cosine wave and a 50 Hz cosine wave.
2. **Frequency Analysis (FFT):** The Fast Fourier Transform was used to convert the time-domain signal into the frequency domain. This allows us to visualize the signal as "spikes" at specific frequencies rather than a wavy line.
3. **Filter Construction:** The filter vector h was defined using the coefficients derived in Section 3.3 ([1, -1.902..., 1]).
4. **Filtering (Frequency Domain):** Instead of using time-domain convolution, we utilized the property that Convolution in Time = Multiplication in Frequency.

$$Y(f) = S(f) \cdot H(f) \tag{11}$$

We computed the FFT of the signal (FFTs) and the FFT of the filter (FFTh), multiplied them element-wise, and obtained the output spectrum.

5. **Signal Reconstruction (IFFT):** The Inverse FFT was applied to the resulting spectrum to view the filtered signal in the time domain.

5. Results and Detailed Analysis

Input Signal Characteristics (Figures 1 & 2)

The input signal consisted of two distinct sinusoids. In the time domain (Figure 1), this resulted in a waveform with a "beating" pattern—amplitude variations caused by the constructive and destructive interference of the 100 Hz and 50 Hz waves.

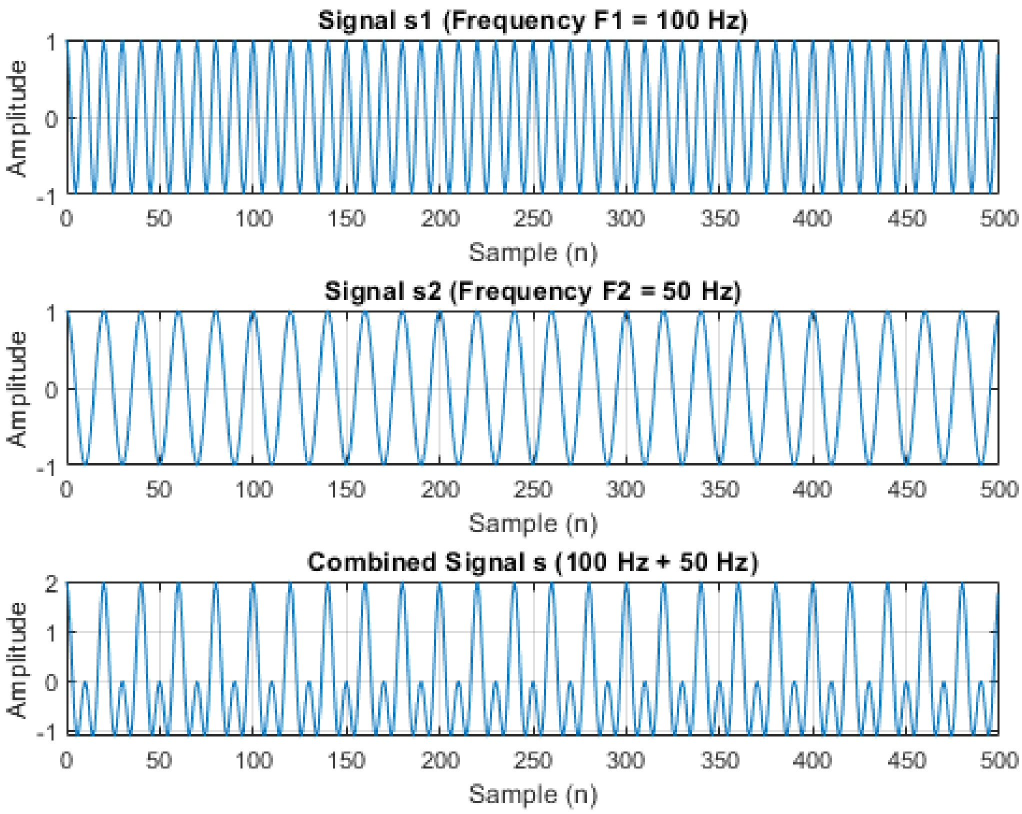

Figure 1: Input Signal - Time Domain

In the Frequency Domain (Figure 2), the spectrum clearly displayed two discrete impulses of equal height: one at 50 Hz and one at 100 Hz. This confirms the composition of the input.

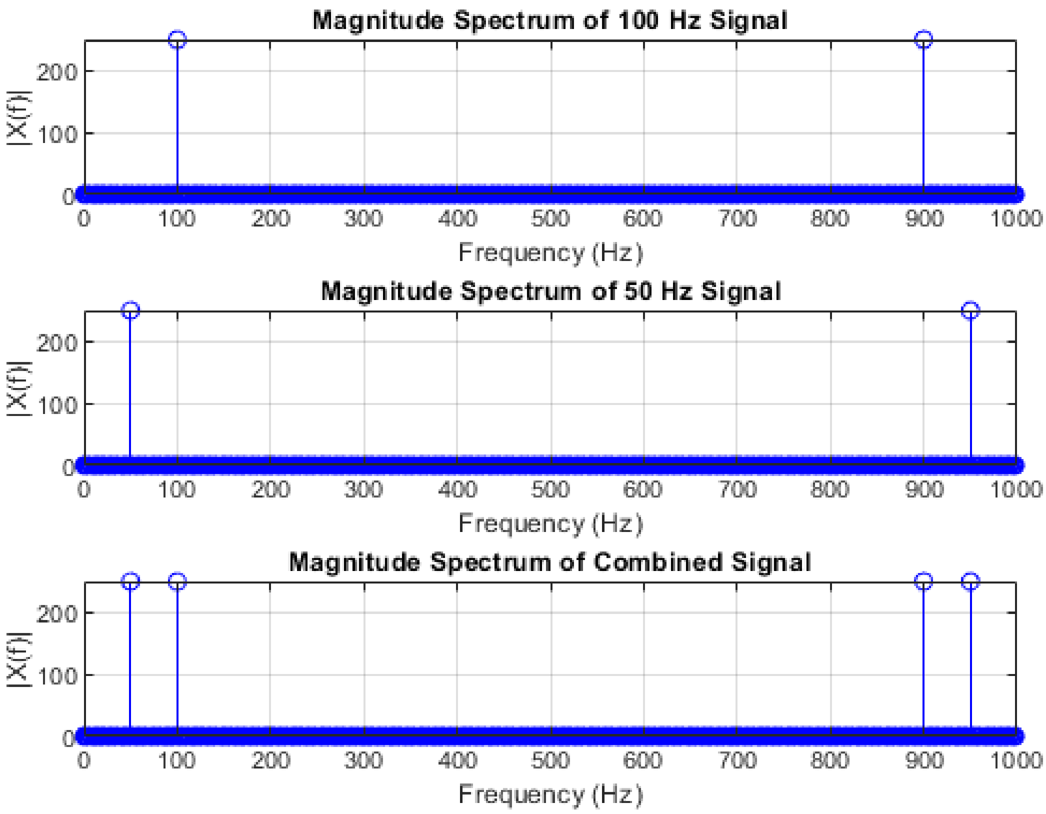

Figure 2: Input Signal - Frequency Domain

Filter Frequency Response (Figure 3)

Figure 3 displays the magnitude spectrum of the designed Null Filter $|H(f)|$.

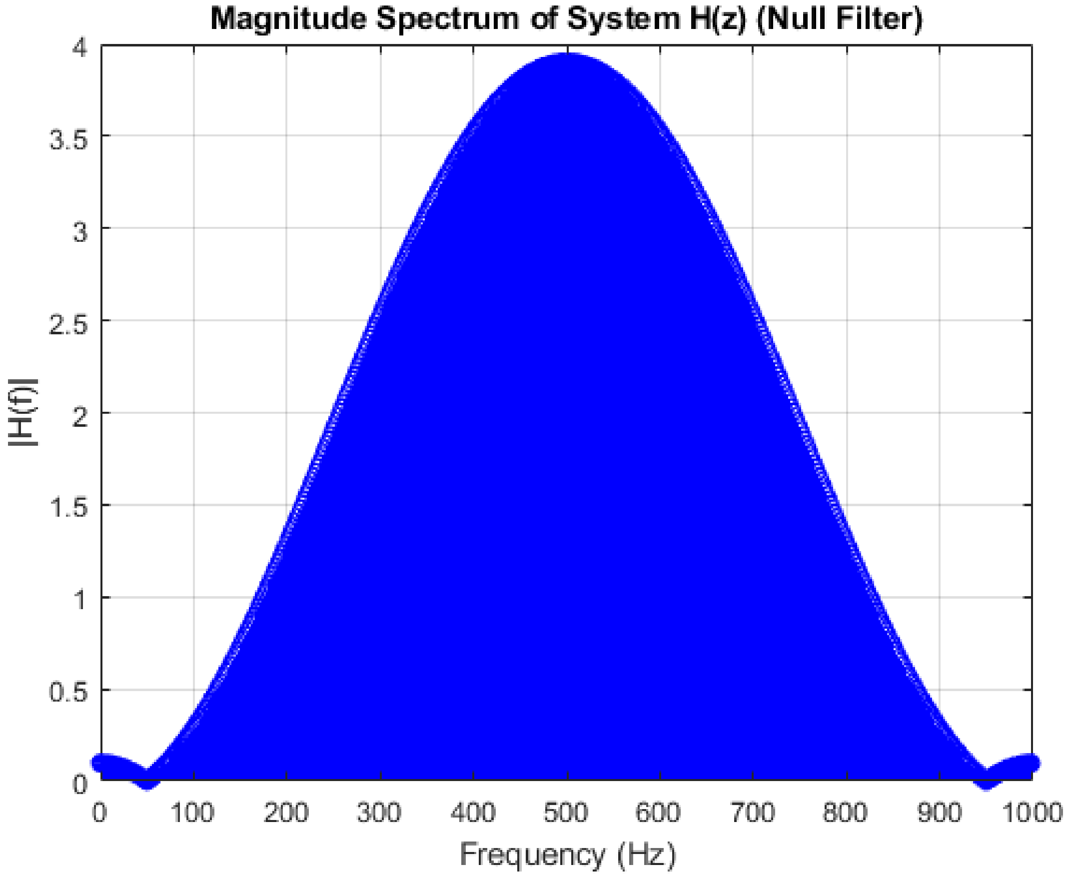

Figure 3: Filter Frequency Response

Shape: The response is not a sharp rectangular "notch." Instead, it resembles a wide, inverted parabolic curve (a "V" shape) that starts at 0 Hz, dips to zero at exactly 50 Hz, and rises back up.

The Null Point: At exactly 50 Hz, the magnitude is effectively 0. This confirms our calculation of the coefficients was correct.

The Slope: A critical observation is the gradual slope of the curve. It does not jump immediately from 0 gain back to 1 gain. It rises slowly.

Output Spectrum (Figure 4)

The output spectrum shows the result of multiplying the Input Spectrum by the Filter Response.

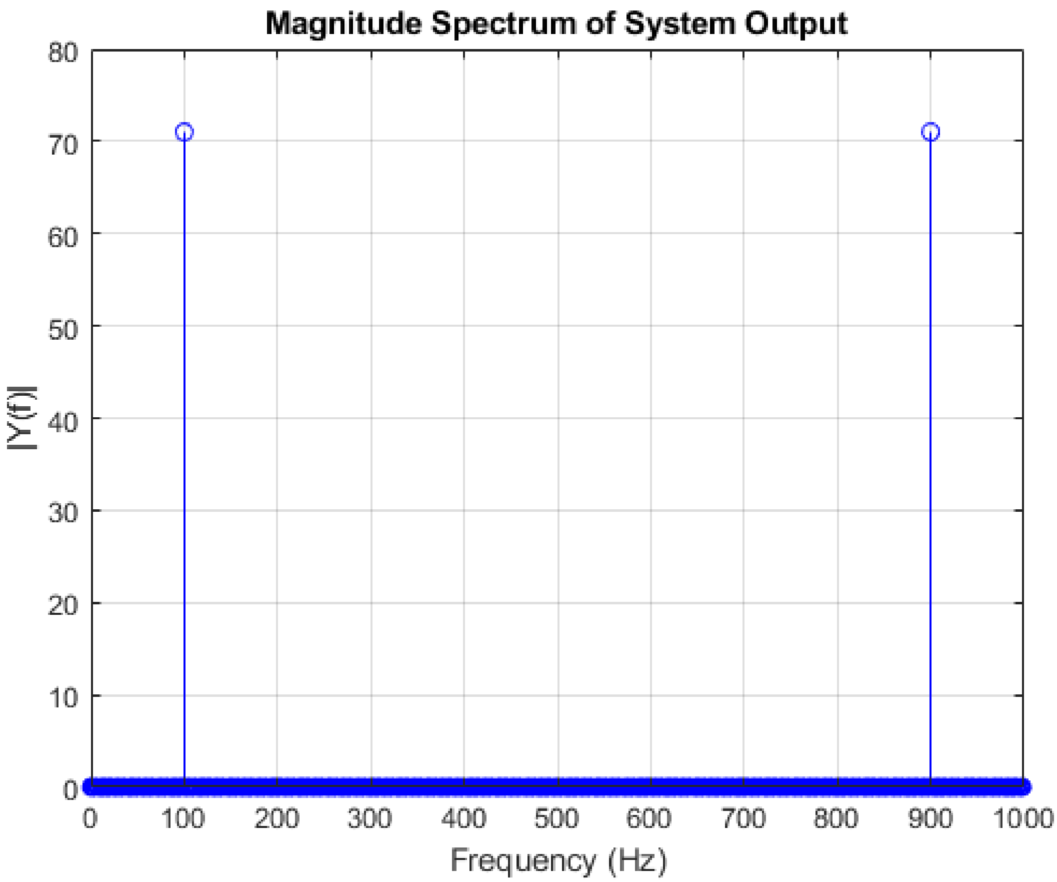

Figure 4: Output Signal - Frequency Domain

50 Hz Component: The input amplitude at 50 Hz was multiplied by the filter gain of 0. Result: The 50 Hz component is completely eliminated.

100 Hz Component: The 100 Hz component remains, but its magnitude is visible. It is the only remaining frequency spike.

5.4 Input vs. Output Comparison (Figure 5 - Critical Analysis)

Figure 5 overlays the original clean 100 Hz signal (Blue) with the filtered output (Red). This plot reveals the primary limitation of this FIR filter design.

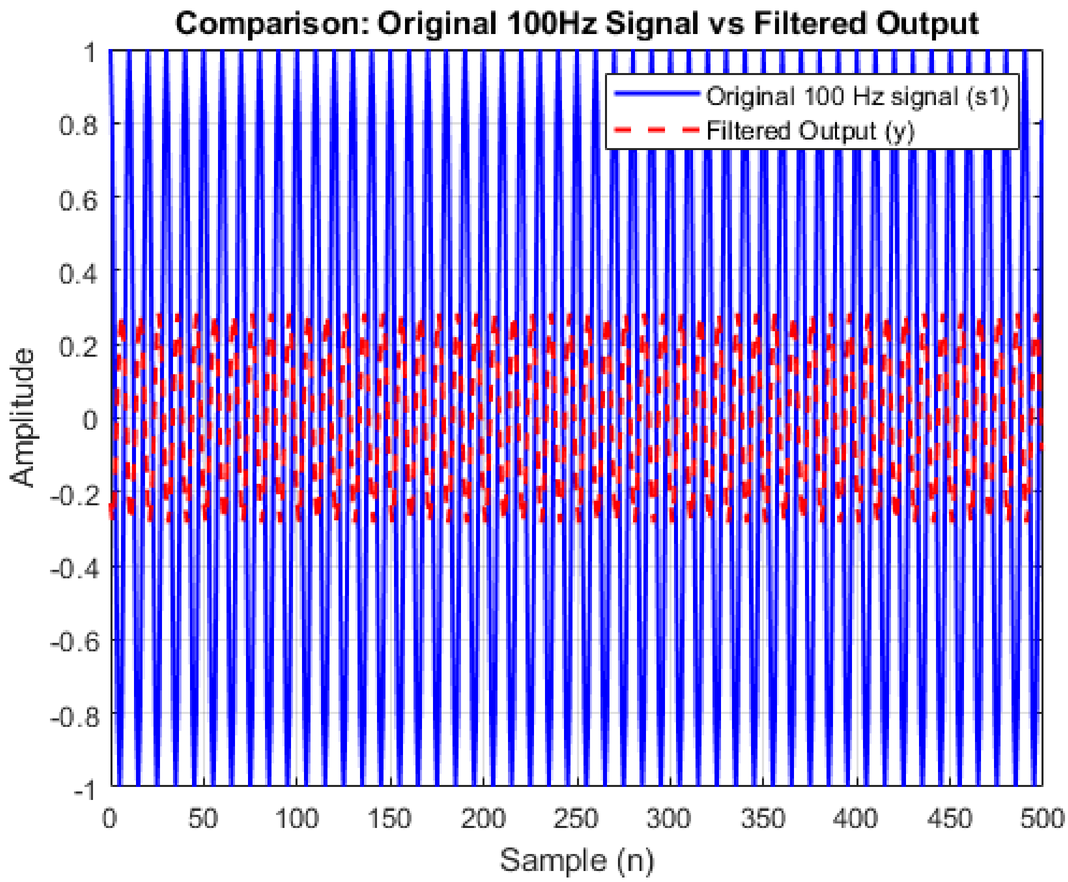

Figure 5: Input vs. Output Comparison – Time Domain

Observation: The Red line (Output) is a clean sine wave with no 50 Hz ripple, meaning the noise removal was successful.

Problem: The amplitude of the Red line is significantly lower than the Blue line. The original signal had an amplitude of 1.0, while the filtered signal has an amplitude of approximately 0.3.

Detailed Explanation of Amplitude Loss:

Why did we lose 70% of our desired signal? This is due to the Frequency Response shape observed in Figure 3.

- The filter forces the gain to be 0 at 50 Hz.
- Because this is a simple 2nd-order FIR filter (only 2 zeros, no poles), the transition from "Stop" (0) to "Pass" (1) is very slow.
- Our desired signal is at 100 Hz. This is relatively close to 50 Hz in the digital domain.
- At 100 Hz, the filter's curve has not yet recovered to a gain of 1. It is still climbing up the slope of the "V".
- Mathematically, if we plug $z = e^{j2\pi(100/1000)}$ into our transfer function, the magnitude $|H(z)|$ evaluates to approx 0.3.

Therefore, the filter attenuated the 100 Hz signal essentially as "collateral damage" while trying to kill the 50 Hz noise.

Discussion

The experiment successfully demonstrated the principle of Zero-Placement. By understanding the relationship between the unit circle angles and signal frequency, we were able to mathematically derive a system that eliminates a specific target frequency.

However, the experiment also highlighted a fundamental trade-off in DSP filter design: Complexity vs. Selectivity.

- The filter we designed was very simple (Order $M = 2$). It required very little computation (just 3 multiplications per sample).
- However, low-order FIR filters have very poor selectivity. They cannot distinguish sharply between frequencies that are close together. To remove 50 Hz without touching 100 Hz, we would need a filter with a much sharper "notch"—a narrow canyon rather than a wide valley.

To achieve this without amplitude loss, one would typically need either a much higher order FIR filter (more coefficients) or an IIR (Infinite Impulse Response) filter that utilizes Poles to boost the gain back up quickly.

Conclusion

In this lab, we designed a Null Filter to reject 50 Hz interference. The design was verified using MATLAB.

Success: The filter successfully completely removed the 50 Hz component from the signal.

Limitation: The desired 100 Hz signal was attenuated by approximately 70% due to the wide stopband width of the 2nd-order FIR filter.

Takeaway: While Zero-placement is an effective technique for removing frequencies, care must be taken regarding the filter's bandwidth. For applications where the noise and signal are close in frequency, simple FIR null filters may degrade the desired signal quality, necessitating more advanced filter designs.