

信号与系统实验报告

名 称: FIR 数字滤波器的设计及应用

学 院: 计软智学院

专 业: 人工智能专业

学 号: 58122310

姓 名: 唐梓烨

日期: 2024 年 05 月 29 日

一、实验目的

1. 进一步理解 FIR 滤波器的线性相位特性, 熟悉 4 种线性相位 FIR 滤波器的幅频特性、相频特性。
2. 掌握用窗函数法设计 FIR 数字滤波器的原理及方法, 了解各种窗函数对滤波器性能的影响。
3. 掌握用频率采样法设计 FIR 数字滤波器的原理及方法, 了解不同采样点数和不同的过渡带点数对滤波器性能的影响。
4. 结合学过的信号分析的基本概念、基本理论及方法, 将数字滤波器设计基本原理、基本方法进行综合运用, 以解决实际信号分析与处理问题。

二、实验任务

1. 完成实验内容全部题目, 分析解决调试代码过程中出现的问题。
2. 认真完成本次实验小结, 思考 FIR 数字滤波器的设计方法及其应用。

三、主要设备、软件平台

1. 硬件: 计算机
2. 软件: Matlab

四、实验内容

1. 选择合适的窗函数设计一个线性相位 FIR 低通滤波器, 通带截止频率 $\omega_p = 0.2\pi rad$, 阻带截止频率 $\omega_s = 0.4\pi rad$, 阻带最小衰减 $\alpha_s = 50dB$ 。要求分别绘制理想单位脉冲序列响应曲线、窗函数曲线、实际单位脉冲序列响应曲线、FIR 低通滤波器的幅频响应曲线和相频响应曲线。

实验思路:

我们的主要目的是设计一个基于窗函数法的低通 FIR 滤波器, 并分析其频率响应。首先我们可以根据题目给出的通带截止频率和阻带截止频率以及阻带最小衰减, 计算出过渡带的带宽, 并估算滤波器的阶数, 并确保滤波器阶数为奇数。接着, 利用 Sa 函数计算理想滤波器的单位脉冲响应。为了得到实际滤波器的单位脉冲响应, 我们还要应用窗函数对理想响应进行加权处理, 在这里我选择了汉明窗。

实验代码:

```
% 参数设置
wp = 0.2 * pi; % 通带截止频率
ws = 0.4 * pi; % 阻带截止频率
As = 50; % 阻带最小衰减 (dB)

delta_w = ws - wp; % 过渡带宽
N = ceil((As - 8) / (2.285 * delta_w)); % 估算滤波器阶次

if mod(N, 2) == 0
```

```

N = N + 1;
end

n = 0:N-1;
wc = (wp + ws) / 2; % 截止频率
hd = sin(wc * (n - (N-1)/2)) ./ (pi * (n - (N-1)/2));
hd((N-1)/2 + 1) = wc / pi;

% 汉明窗
w = 0.54 - 0.46 * cos(2 * pi * n / (N-1));

% 实际滤波器的单位脉冲响应
h = hd .* w;

% 计算频率响应
N_fft = 1024;
H = fft(h, N_fft); % 计算频率响应
H = H(1:N_fft/2+1);
W = (0:N_fft/2) * (2*pi/N_fft); % 归一化频率

% 绘制图形
figure;

subplot(5, 1, 1);
stem(n, w, 'filled');
title('汉明窗');
xlabel('n');
ylabel('w[n]');

subplot(5, 1, 2);
stem(n, hd, 'filled');
title('理想单位脉冲响应');
xlabel('n');
ylabel('hd[n]');

subplot(5, 1, 3);
stem(n, h, 'filled');
title('实际单位脉冲响应');
xlabel('n');
ylabel('h[n]');

subplot(5, 1, 4);
plot(W/pi, 20*log10(abs(H)));

```

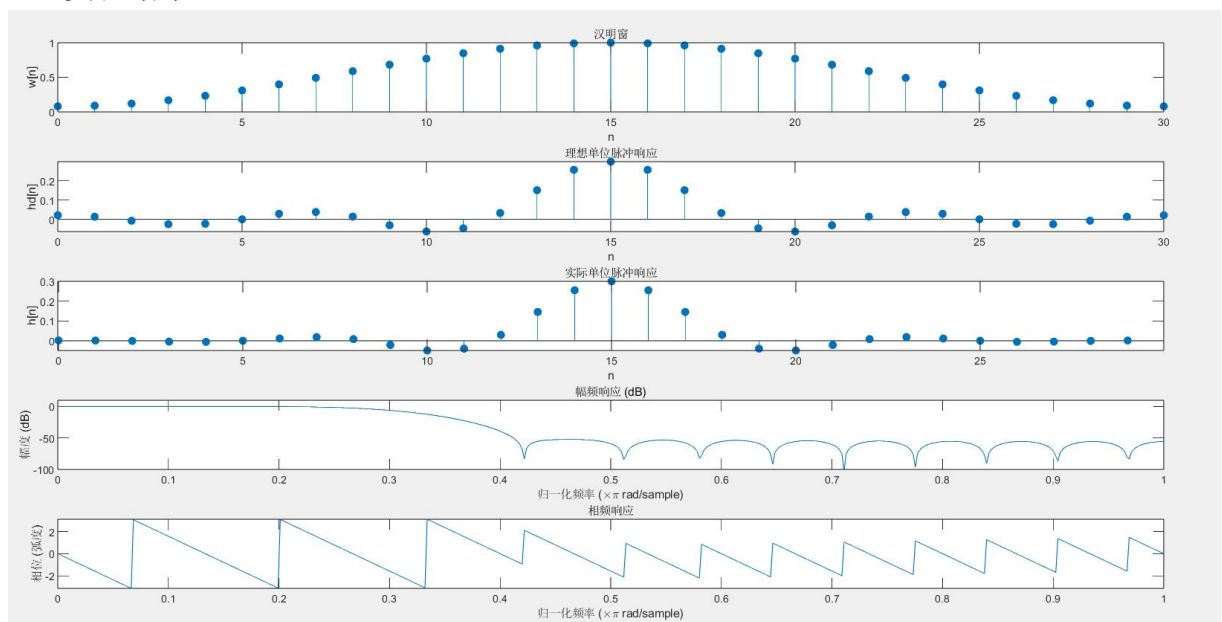
```

title('幅频响应 (dB)');
xlabel('归一化频率 (\times\pi rad/sample)');
ylabel('幅度 (dB)');
ylim([-100 10]);

subplot(5, 1, 5);
plot(W/pi, angle(H));
title('相频响应');
xlabel('归一化频率 (\times\pi rad/sample)');
ylabel('相位 (弧度)');

```

实验结果：



2. 利用频率采样法设计线性相位低通滤波器

1) 要求 3dB 截止频率 $\omega_c = \pi/2 \text{ rad}$, 采样点数分别取 $N = 33$ 和 $N = 63$, 选用 $h(n) = h(N - 1 - n)$ 的情况。分别显示理想幅频特性的采样序列和实际滤波器的幅频响应, 观察采样点数对滤波器特性的影响。

实验思路：

和实验 1 类似, 我首先设计了一个低通滤波器, 在代码中的低通滤波器函数给出。然后, 我通过快速傅里叶变换 (FFT) 计算理想幅频特性和实际滤波器的幅频响应。为了比较不同采样点数的影响, 我们选择两组不同的采样点数来设计低通滤波器, 分别计算和绘制其理想幅频特性和实际幅频响应。

实验代码：

```

%参数
wc = pi / 2; % 截止频率
N1 = 33; % 滤波器 1 阶数为 33
N2 = 63; % 滤波器 2 阶数为 63

```

```

% 设计两个滤波器
h1 = design_filter(N1, wc);
h2 = design_filter(N2, wc);

% 计算频率响应
fftLength = 1024;
H1 = fft(h1, fftLength);
H2 = fft(h2, fftLength);
W = linspace(0, 1, fftLength / 2) * pi;

% 绘图
figure;
subplot(2,1,1);
plot(W, 20*log10(abs(H1(1:fftLength/2))));
title('N=33 的 FIR 低通滤波器幅度响应');
xlabel('归一化频率 ( $\times \pi$  rad/sample)');
ylabel('幅度 (dB)');

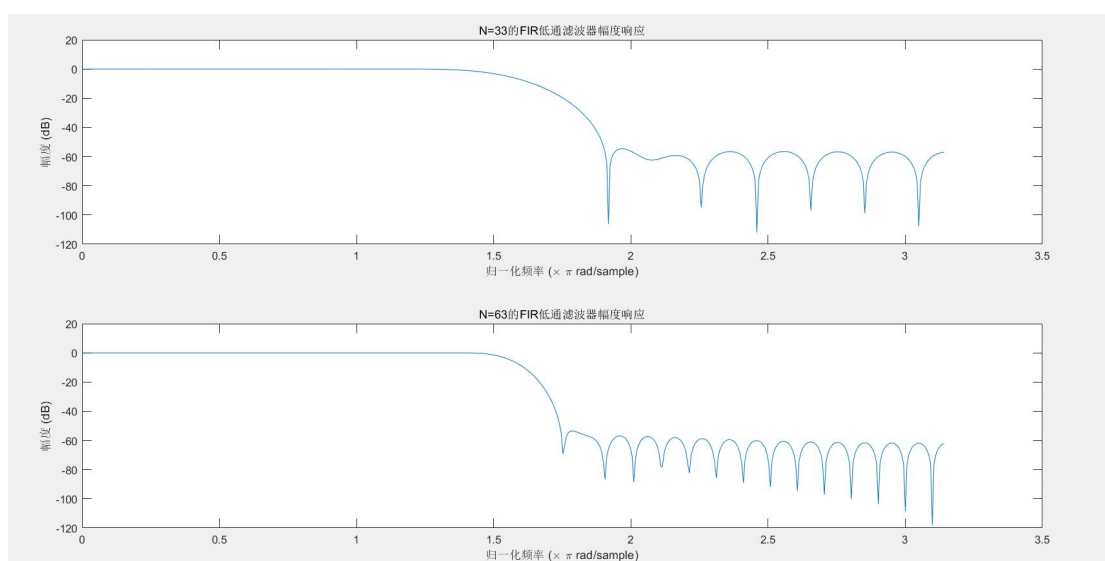
subplot(2,1,2);
plot(W, 20*log10(abs(H2(1:fftLength/2))));
title('N=63 的 FIR 低通滤波器幅度响应');
xlabel('归一化频率 ( $\times \pi$  rad/sample)');
ylabel('幅度 (dB)');

% 手动实现 Hamming 窗函数
function w = manual_hamming(N)
n = 0:N-1;
w = 0.54 - 0.46 * cos(2 * pi * n / (N-1));
end

% 设计滤波器函数
function h = design_filter(N, wc)
n = 0:N-1;
sinc_part = sin(wc * (n - (N-1)/2)) ./ (pi * (n - (N-1)/2)); % 理想的冲击
响应
sinc_part((N-1)/2 + 1) = wc / pi; % 中心点, 处理除零情况
w = manual_hamming(N);
h = sinc_part .* w; % 实际的冲击响应
end

```

实验结果：



实验分析：

通过比较这两组结果，我们可以观察到采样点数的变化如何影响滤波器的幅频特性和相频特性。一般而言，较多的采样点数会使滤波器的过渡带更陡峭，频率响应更接近理想滤波器，但同时也增加了计算量和滤波器长度。通过这个实验，我们可以直观地了解采样点数对滤波器设计的影响。

2) 要求 3dB 截止频率 $\omega_c = \pi/2 \text{ rad}$, 采样点数取 $N = 33$, 选用 $h(n) = h(N - 1 - n)$ 的情况。分别插入一个过渡点和两个过渡点时滤波器的幅频响应，观察增加过度点对滤波器特性的影响。

实验思路：

3) 和前面的实验思路类似，我首先设计了一个阶数为 33 的低通滤波器，并应用汉明窗来生成实际单位脉冲响应，再利用 FFT 将时域响应转换为幅频响应。为了观察增加过度点对滤波器特性的影响。我分别设计了两个低通滤波器，第一个的过渡点在 $t_1=0.5$ 处，第二个的过渡点在 $t_2=0.3, t_3=0.6$ 处。

实验代码：

```
% 参数设置
N = 33; % 滤波器的长度
wc = pi/2; % 截止频率
M1 = fix(wc/(2*pi/N));
M2 = N - 2 * M1 - 1;
theta = -pi * (0:N-1) * (N-1) / N;
resolution = 512; % 设置频率响应的分辨率

% 设计第一个滤波器
T1 = 0.5;
A1 = [ones(1, M1), T1, zeros(1, M2), T1, ones(1, M1-1)];
h1 = computeFilterResponse(A1, theta);
[H1, w1] = manualFreqz(h1, resolution);
```

```

% 设计第二个滤波器
T2 = 0.3; T3 = 0.6;
A2 = [ones(1, M1), T2, T3, zeros(1, M2-2), T3, T2, ones(1, M1-1)];
h2 = computeFilterResponse(A2, theta);
[H2, w2] = manualFreqz(h2, resolution);

% 绘制两个滤波器的幅度响应图
subplot(1, 2, 1);
plot(w1/pi, abs(H1), w2/pi, abs(H2), 'r');
grid on;
xlabel('频率(单位\pi)');
ylabel('幅度');

% 计算以分贝表示的归一化幅度响应
dbH1 = 20 * log10(abs(H1) / max(abs(H1)));
dbH2 = 20 * log10(abs(H2) / max(abs(H2)));

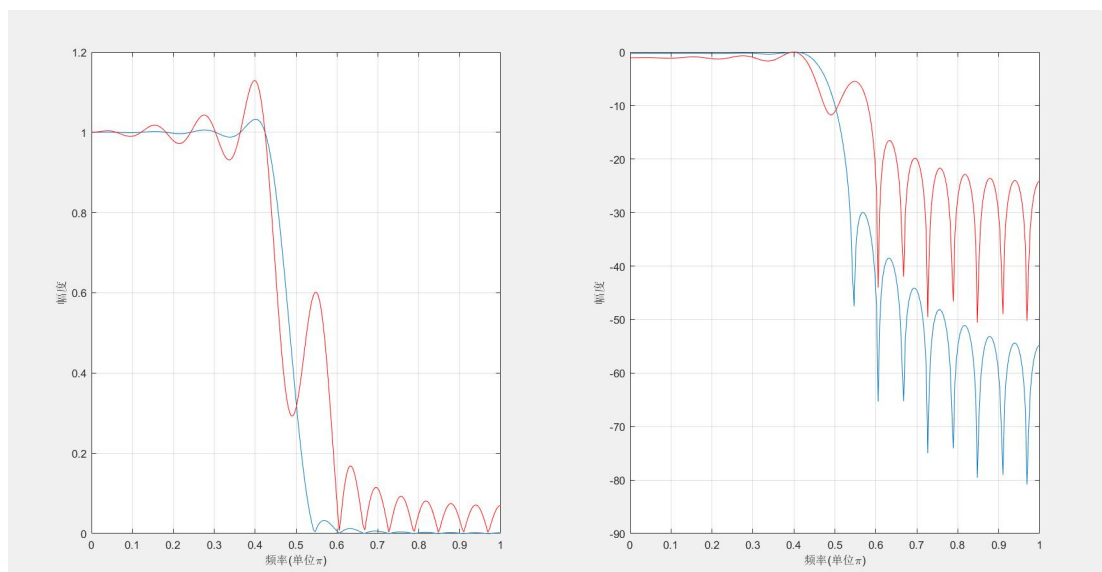
% 绘制两个滤波器的分贝幅度响应图
subplot(1, 2, 2);
plot(w1/pi, dbH1, w2/pi, dbH2, 'r');
grid on;
xlabel('频率(单位\pi)');
ylabel('幅度');

function [H, w] = manualFreqz(h, N)
H_full = fft(h, N); % 使用 FFT 计算频率响应
H = H_full(1:N/2+1);
w = 2 * pi * (0:N/2) / N;
end

function h = computeFilterResponse(A, theta)
Hk = A .* exp(1i * theta);
h = real(ifft(Hk));
end

```

实验结果：



实验分析：

图中红色为插入一个过渡点时滤波器的幅频响应，蓝色为插入两个过渡点时滤波器的幅频响应。可以看出增加过渡点可以使频率响应曲线更加平滑，减少过渡带内的波动。这是因为过渡点的引入相当于在滤波器设计中引入了更多的零点，使得频率响应在过渡带内更加均匀分布。而且增加过渡点可以提高阻带衰减。由于过渡点的引入阻带衰减变得更迅速，这有助于滤除不需要的频率成分。

3. 利用计算机的声音编辑工具录制一段语音信号，生成.wav 文件。对语音信号的时域波形和频谱进行观察和分析，设计 FIR 数字滤波器，并用设计的滤波器对语音信号进行滤波，比较滤波前后语音信号的波形及频谱。

- 1) 低通滤波器性能指标: $f_p = 1000\text{Hz}$, $f_s = 1200\text{Hz}$, $A_s = 100\text{dB}$, $A_p = 1\text{dB}$;
- 2) 高通滤波器性能指标: $f_s = 4800\text{Hz}$, $f_p = 5000\text{Hz}$, $A_s = 100\text{dB}$, $A_p = 1\text{dB}$ 。

实验思路：

首先，读取音频文件并转换为单声道，计算其 FFT。然后，设计一个低通和一个高通 FIR 滤波器，分别设置其通带和阻带截止频率。通过手动实现的 FIR 滤波器设计函数，计算理想冲激响应并应用汉明窗。接着，分别对音频信号进行低通和高通滤波。最后，绘制原始信号和滤波后信号的时域波形和频谱，以比较不同滤波器对信号的影响。

实验代码：

```
clc; clear;

% 读取音频文件并转换为单声道
[x, fs] = audioread('music.wav');
x = (x(:, 1))'; % 转置为行向量

% 获取信号长度
FS = length(x);

% 计算 FFT
```



```

X = fft(x);

% 时间轴
t = (0:FS-1) / fs;

% 低通滤波器设计参数
Fp_lp = 1000; % 低通滤波器通带截止频率 (Hz)
Fs_lp = 1200; % 低通滤波器阻带截止频率 (Hz)
Ft = fs; % 采样频率
wp_lp = 2*pi*Fp_lp/Ft;
N_lp = 100; % 滤波器阶数

% 设计低通滤波器
h_lp = fir1_lp(N_lp, wp_lp);

% 应用低通滤波器
y1 = filter(h_lp, 1, x);

% 高通滤波器设计参数
Fp_hp = 5000; % 高通滤波器通带截止频率 (Hz)
Fs_hp = 4800; % 高通滤波器阻带截止频率 (Hz)
wp_hp = 2*pi*Fp_hp/Ft;
N_hp = 100; % 滤波器阶数

% 设计高通滤波器
h_hp = fir1_hp(N_hp, wp_hp);

% 应用高通滤波器
y2 = filter(h_hp, 1, x);

% 绘制原始和滤波后的信号时域波形和频谱
figure;

% 原始信号时域波形
subplot(3, 2, 1);
plot(t, x);
title('原始语音信号时域波形');
xlabel('时间');
ylabel('幅值');
grid on;

% 原始信号频谱
subplot(3, 2, 2);

```

```
yy = abs(X);  
f = (0:floor(FS/2)-1)*(fs/FS); % 计算频率轴  
plot(f, yy(1:floor(FS/2)));  
title('原始语音信号频谱');  
xlabel('频率 (Hz)');  
ylabel('幅度');  
grid on;
```

```
% 低通滤波后信号时域波形  
subplot(3, 2, 3);  
plot(t, y1);  
title('低通滤波后信号时域波形');  
xlabel('时间');  
ylabel('幅值');  
grid on;
```

```
% 低通滤波后信号频谱  
subplot(3, 2, 4);  
yy_lp = abs(fft(y1));  
plot(f, yy_lp(1:floor(FS/2)));  
title('低通滤波后信号频谱');  
xlabel('频率 (Hz)');  
ylabel('幅度');  
grid on;
```

```
% 高通滤波后信号时域波形  
subplot(3, 2, 5);  
plot(t, y2);  
title('高通滤波后信号时域波形');  
xlabel('时间');  
ylabel('幅值');  
grid on;
```

```
% 高通滤波后信号频谱  
subplot(3, 2, 6);  
yy_hp = abs(fft(y2));  
plot(f, yy_hp(1:floor(FS/2)));  
title('高通滤波后信号频谱');  
xlabel('频率 (Hz)');  
ylabel('幅度');  
grid on;
```

```
% 手动实现低通 FIR 滤波器设计函数
```

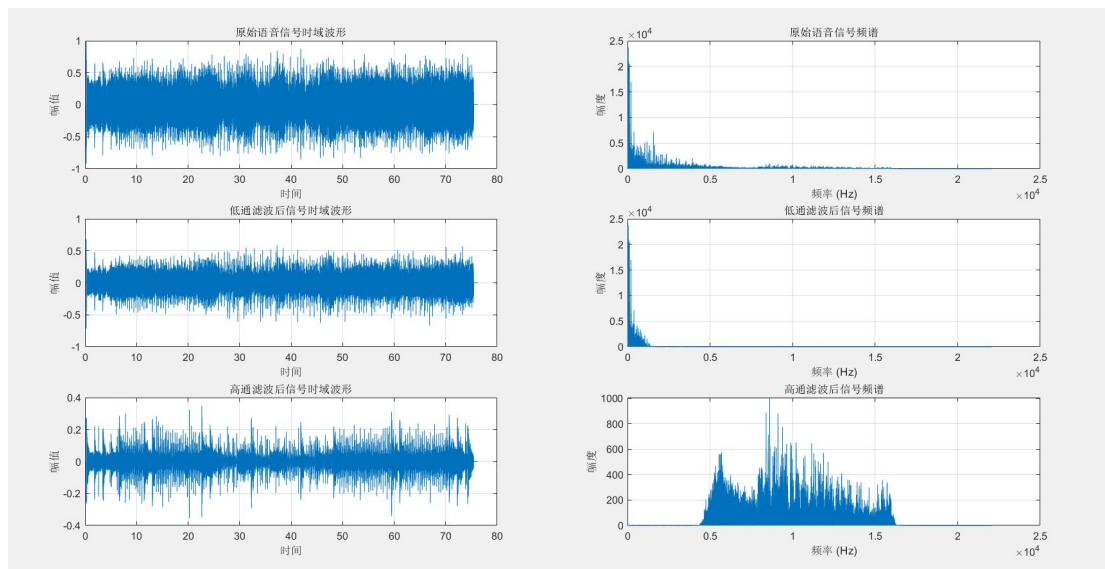
```

function h = fir1_lp(N, wc)
n = 0:N;
alpha = N / 2;
m = n - alpha;
h = sin(wc * m) ./ (pi * m); % 理想低通滤波器冲激响应
h(alpha + 1) = wc / pi;
h = h .* (0.54 - 0.46 * cos(2 * pi * n / N)); % Hamming 窗
end

% 手动实现高通 FIR 滤波器设计函数
function h = fir1_hp(N, wc)
n = 0:N;
alpha = N / 2;
m = n - alpha;
h = -sin(wc * m) ./ (pi * m); % 理想高通滤波器冲激响应
h(alpha + 1) = 1 - wc / pi;
h = h .* (0.54 - 0.46 * cos(2 * pi * n / N)); % Hamming 窗
end

```

实验结果:



实验分析:

左上角的图显示了原始音频信号的时域波形。右上角的图展示了原始信号的频谱。可以看到信号的频谱能量集中在低频部分，随着频率增加，能量迅速衰减。这表明原始信号主要由低频成分组成，高频成分较少。

而在右边中间和右下角的图分别为经过低通、高通滤波器滤过的音频信号的频谱。可以发现低通滤波器的信号仅保留的低频信号，高通滤波器仅保留的高通信号，滤波器设计正确。

左边中间的图显示了经过低通滤波后的信号时域波形。相比原始信号，低通滤波后的信号振幅显得更加平滑，表明高频噪声已经被滤除。

左下角的图显示了经过高通滤波后的信号时域波形。与原始信号相比，高通

滤波后的信号振幅减小,尤其是低频部分的成分被大幅削弱,只保留了高频成分。

五、实验小结

在本实验中,我通过使用 **Matlab** 成功完成了 **FIR** 数字滤波器的设计与实现。在第一个实验中,我们熟悉了 **FIR** 滤波器设计的基本方法,理解了如何通过窗函数法来设计滤波器,并观察其频率响应特性。在第二个实验中,我们采用频率采样法设计了线性相位低通滤波器,再在这个基础上加入了 **FFT** 对滤波器频率响应进行观察,深入分析了增加采样点数和过渡点数对滤波器特性的影响,进一步加深了对 **FIR** 滤波器设计原理的理解。在第三个实验中,我们从滤波器设计、滤波处理,到滤波后信号与原始信号的对比展示,系统地体验了滤波器对数字信号处理的全过程。

通过这些实验,我对信号与系统课程中所学的理论知识在数字信号处理中的实际应用有了更加深入的认识。例如,通过实验可以直观地看到不同滤波器参数对信号处理效果的影响,理解了设计合理的滤波器参数对实现特定信号处理目标的重要性。此外,实验中频域和时域分析方法的结合,使我掌握了从多个角度评估滤波器性能的技能。这不仅提高了我的实际动手能力,还增强了我对数字信号处理原理的理解。