

苏州大学电子信息学院
设计性实验报告

实验名称：基于 DFT 的信号识别系统

实验者姓名：王旭

合作者姓名：无

专业：通信工程

班级：22 通信 2 班

学号：221504064

指导老师：朱哲辰

实验日期：2024.11.13

一. 实验目的

1. 通过实验加深对离散傅里叶变换 DFT 的认识和理解
2. 掌握应用 DFT 进行频谱分析的方法
3. 初步学习 DFT 在实际生活中的应用

二. 实验背景

DFT 在语音识别、雷达信号处理、生物医学信号检测与识别等领域有广泛的应用。在无线通信系统中，无论是频分双工(FDD)、频分多址(FDMA)、跳频技术(FHSS)还是移动通信中小区之间的信道分配，都会利用分布在不同频率上的信道进行频谱资源的划分。现有一台利用跳频技术的发射机，所捕捉的瞬时频谱(上半部分)以及一端时间内的时频谱图(下半部分)。接收机所接受信号已通过采样得到实验文件中的三个信号本文件。利用 DFT 相关知识，完成对信号样本文件的分析与识别。

三. 实验原理

任何周期函数都可以表示为不同频率的正弦波和余弦波的和。非周期函数可以通过扩展周期性来分析，即通过将其视为周期函数的一个周期来处理。在实际应用中，我们处理的信号通常是离散的，即在时间上以固定间隔采样得到的。

对于一个长度为 N 的离散序列 x_0, x_1, \dots, x_{N-1} ，其 DFT 定义为：

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-j\frac{2\pi}{N}kn}$$

其中, X_k 是序列 x_n 的第 k 个 DFT 系数, $k = 0, 1, \dots, N - 1$, j 是虚数单位, $e^{-j\frac{2\pi}{N}kn}$ 是复指数函数。

DFT 通过对有限长时域离散信号的频谱进行等间隔采样，将信号从时域转换到频域，实现信号的频谱分析、滤波、特征提取等操作。

四. 实验内容与分析

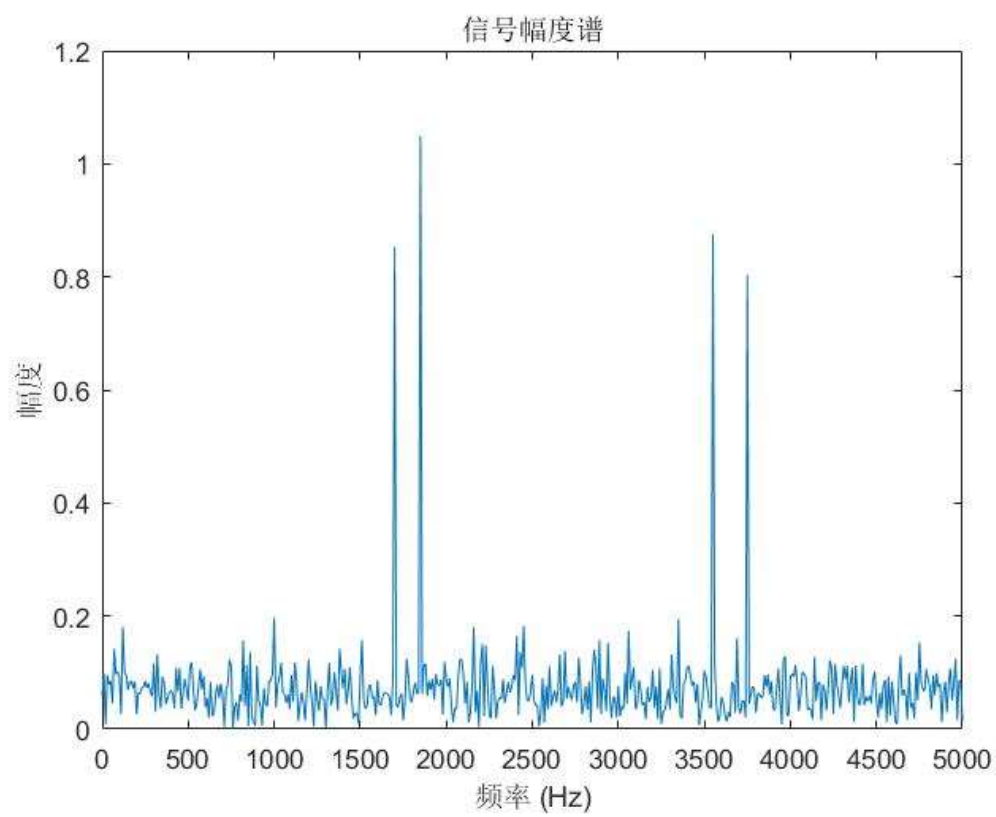
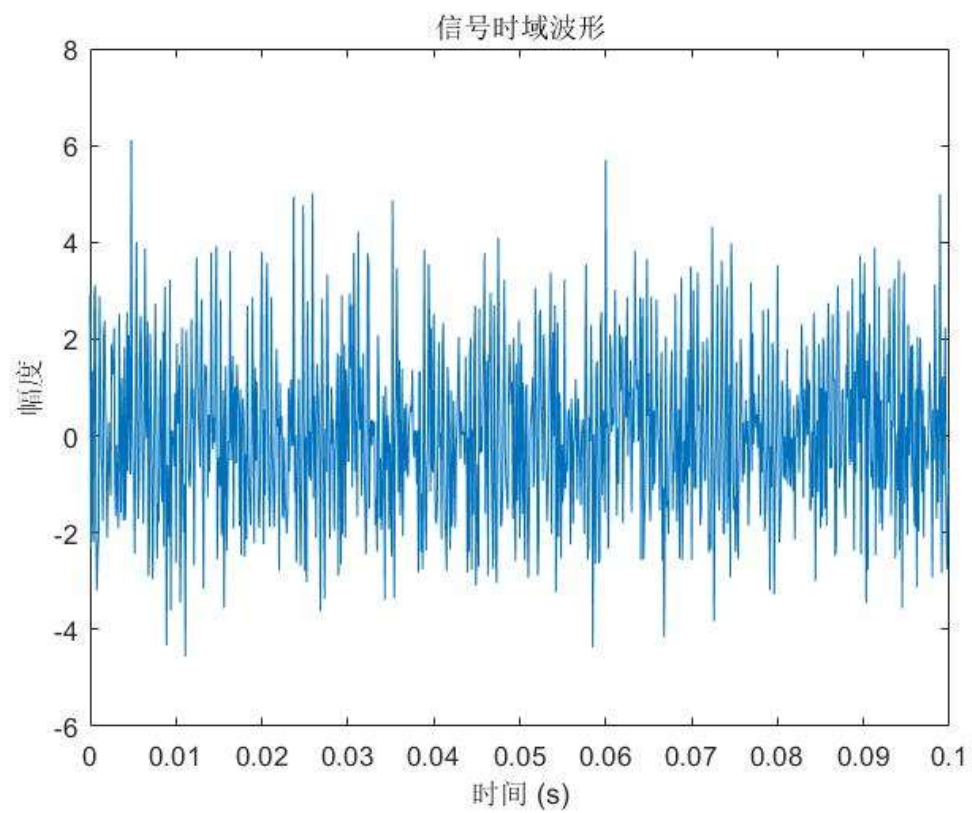
1. 实验一

设窗口长度为 1000，编写 MATLAB 程序，画出信号时域波形、信号幅度谱，并自动识别信号中 4 个最为主要的频率成分，返回其频率值（Hz）。



加载存储在 `x1.mat` 文件中的信号数据，并将其赋值给变量 `signal`。根据设定的窗口长度 $N=1000$ ，对信号进行裁剪或补零，以确保信号长度与窗口长度一致。假设信号的采样频率 F_s 为 10000 Hz ，并根据这个采样频率计算时间向量 t 。绘制信号的时域波形，显示信号随时间的变化。对信号进行傅里叶变换，得到信号的频域表示。计算信号的双边谱 $P2$ 和单边谱 $P1$ ，并对单边谱进行调整，以反映实际的幅度。根据采样频率和窗口长度计算频率向量 f 。绘制信号的幅度谱，显示不同频率成分的幅度。对单边谱 $P1$ 进行降序排序，找到幅度最大的四个频率成分。提取这四个频率成分对应的频率值。显示四个最为主要的频率成分，这些成分代表了信号中最主要的频率信息。

实验结果如下：



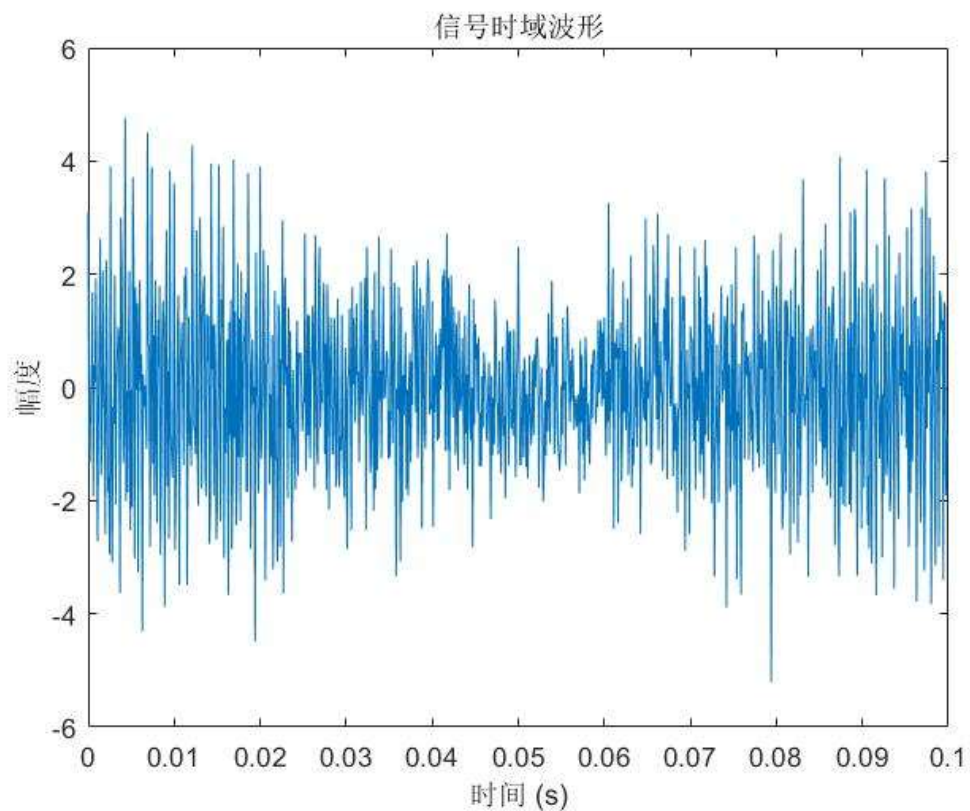
```
命令窗口
>> x111111
四个最主要的频率成分 (Hz) :
    1850    3550    1700    3750
fx>>
```

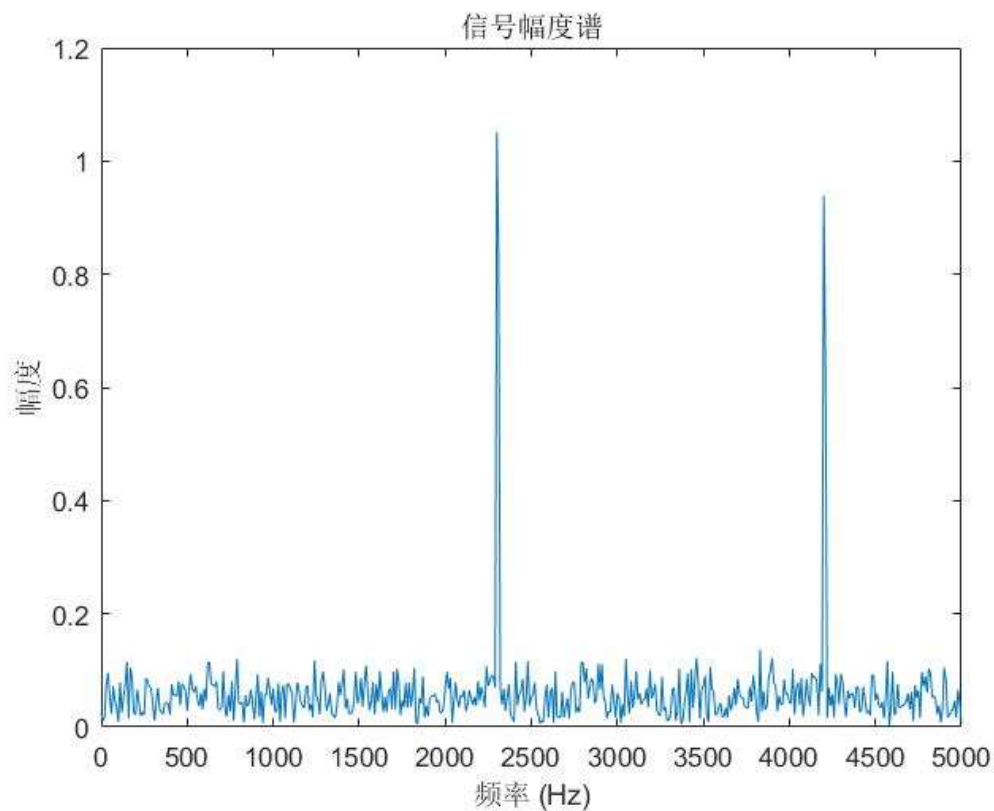
通过分析，得出四个最主要的频率成分分别为 1700Hz，1850Hz，3550Hz，3750Hz。

2. 实验二

(1) 已知信号二中包含 4 个主要频率成分，用 1 的程序对信号二进行分析，是否能准确识别信号二中所有频率分量？（要求检测峰值间有谷点）

使用实验一中程序，将加载数据由 x1.mat 更改为 x2.mat，运行程序，得到以下结果：





发现四个主要频率分量分别为



(2) 选取合适的窗口长度，画出信号幅度谱，并自动识别信号中 4 个最为主要的频率成分，返回其频率值 (Hz)。分析窗口长度与频率分辨率的关系。

我们更改窗口长度分别为 200, 500, 1000, 2000, 4000，并分别计算四个主要频率分量，结果如下图所示：

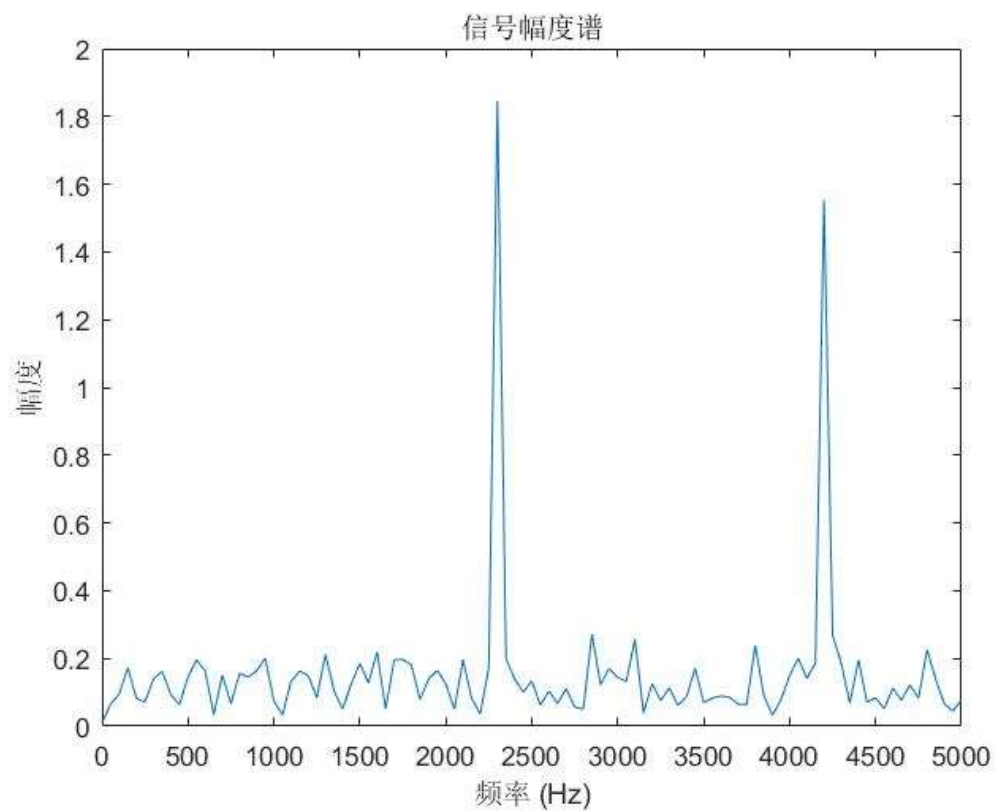


表 1 窗口长度 $N=200$

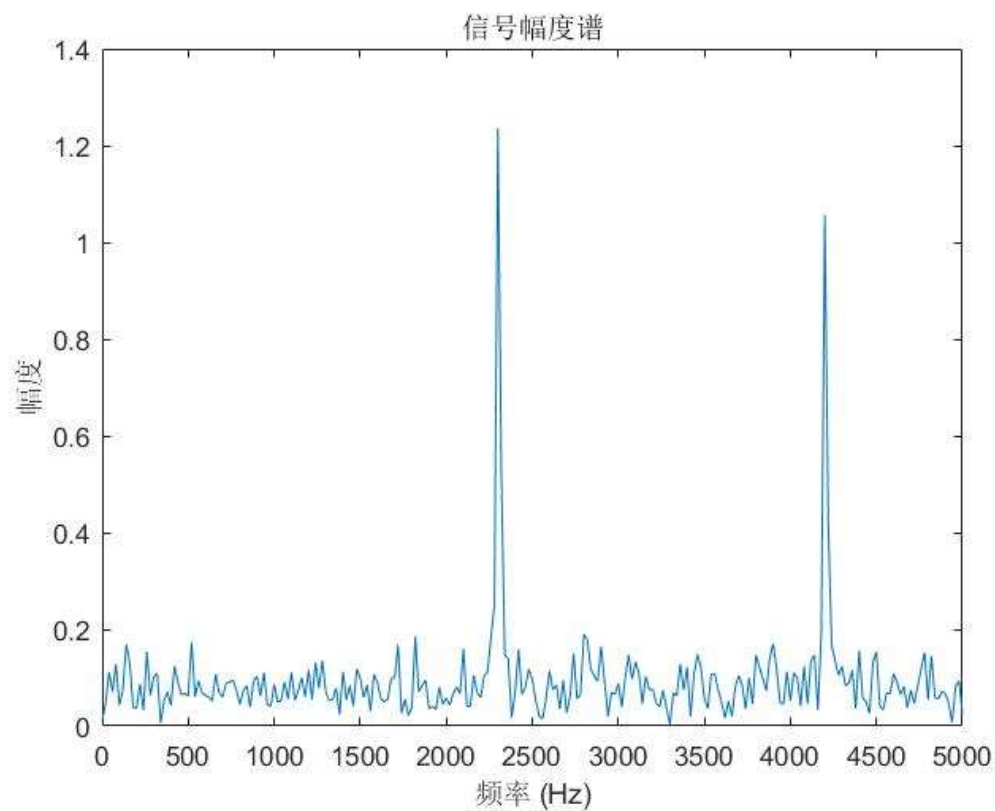


表 2 窗口长度 $N=500$

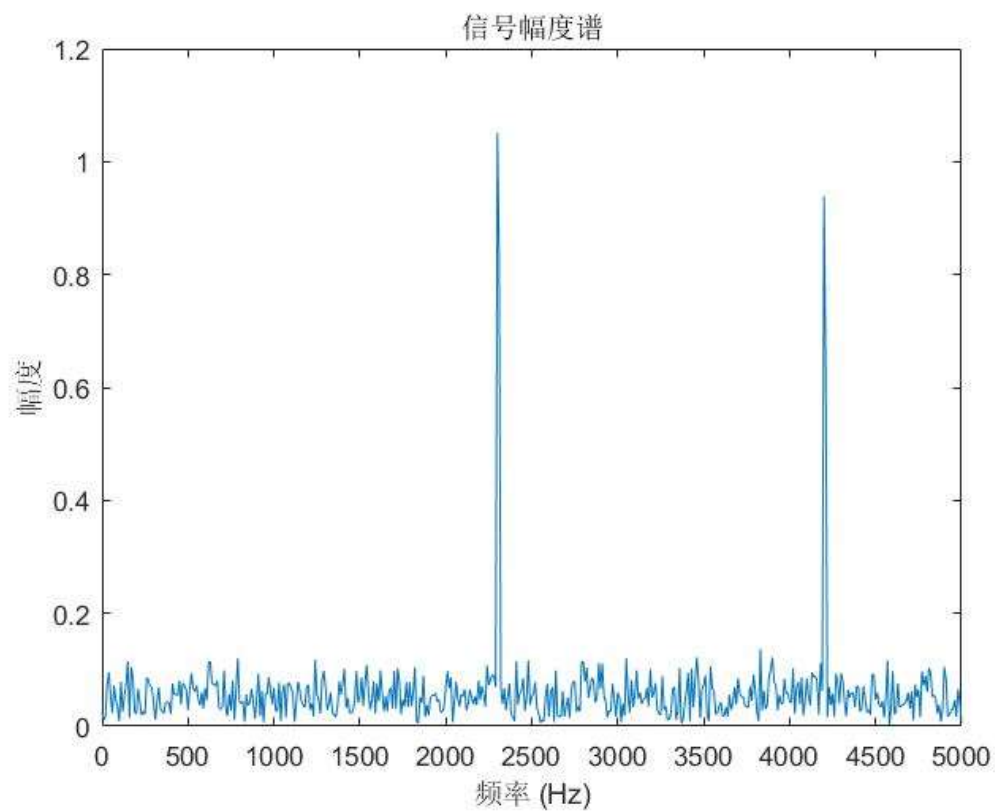


表 3 窗口长度 $N=1000$

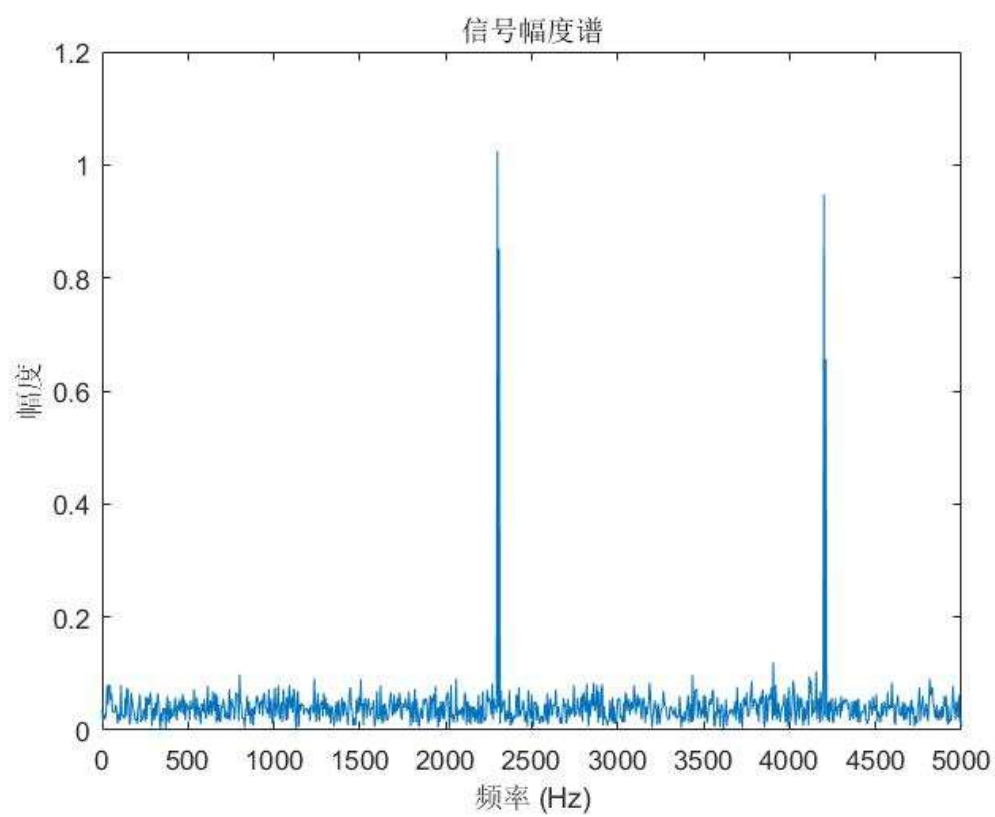


表 4 窗口长度 $N=2000$

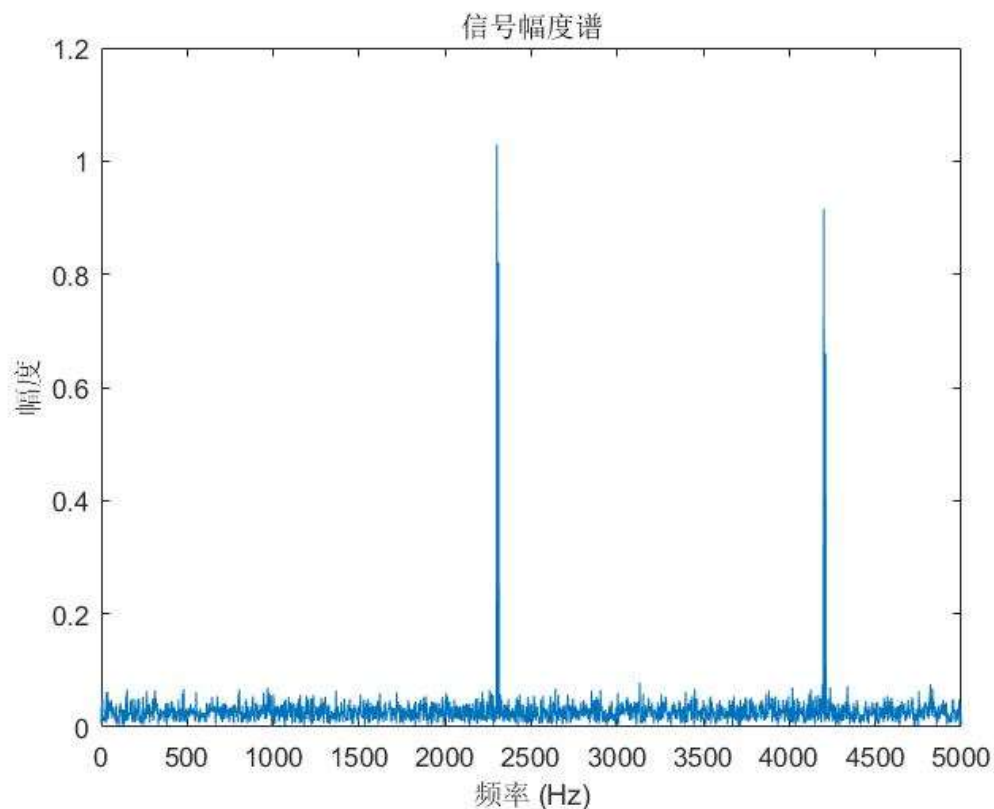


表 5 窗口长度 N=4000

窗口长度	主要频率 1	主要频率 2	主要频率 3	主要频率 4
200	2300	2850	4200	4250
500	2300	2320	4200	4220
1000	2300	2310	4200	4210
2000	2300	2310	4200	4210
4000	2300	2310	4200	4210

表格 1 窗口长度与主要频率的关系

观察图表能够发现，随着窗口长度的增大，主要频率趋向于一个恒定值，这体现了采样范围越大，频率的识别越精确，但这样运算量也越大，而根据频率分辨率和窗口值间的关系：

$$\text{频率分辨率} = \frac{\text{采样率}}{\text{窗口长度}}$$

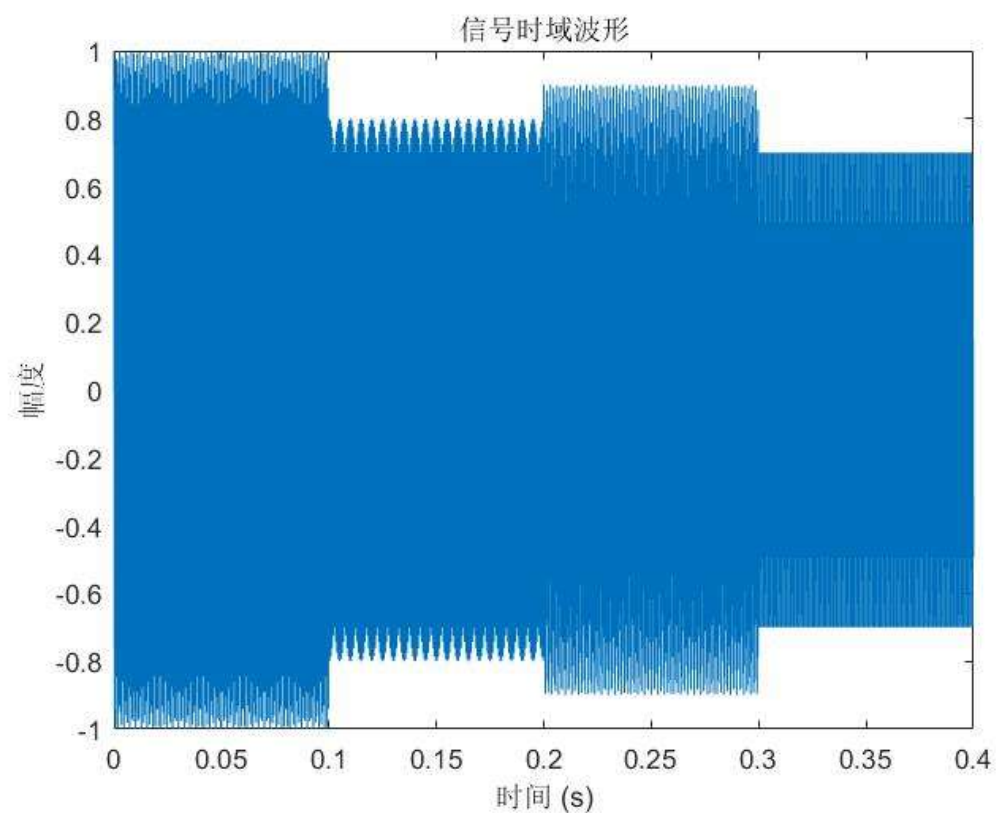
由此可见在采样率为 10kHz 不变的情况下，窗口长度越长，频率分辨率越低，

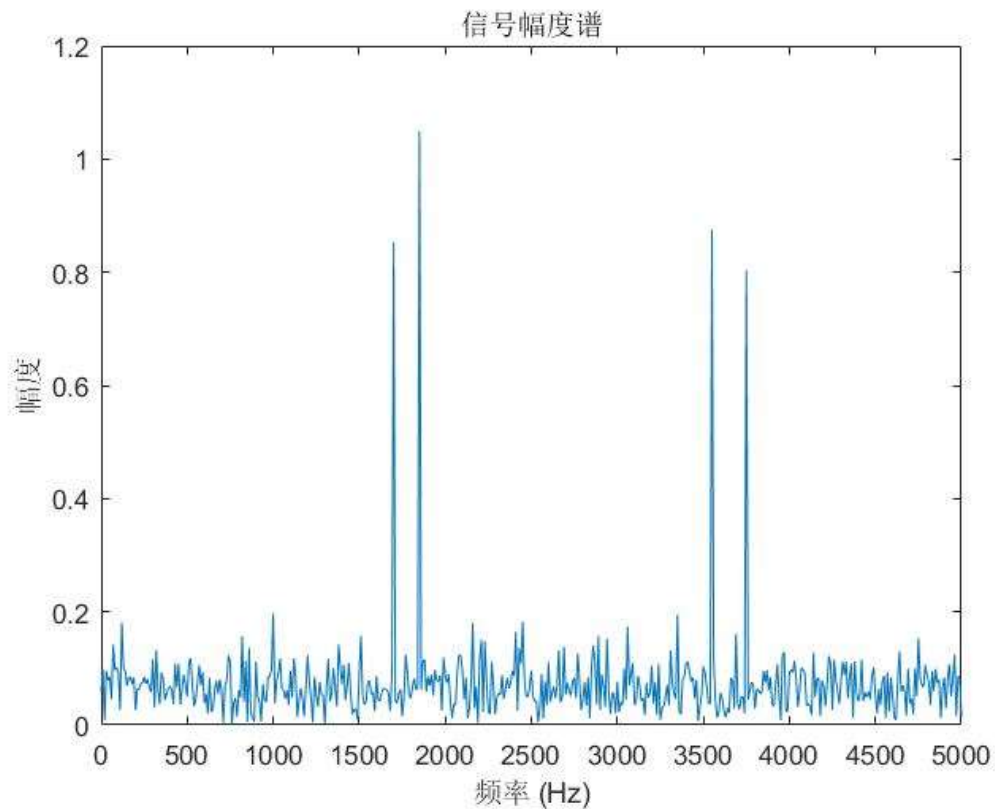
频率成分识别越准确。

3. 实验三

(1) 现有一较长信号 ($N=4000$)，信号随时间发生频率跳变，画出信号时域波形、信号幅度谱，与信号一进行比较分析

使用实验一中程序，将加载数据由 `x1.mat` 更改为 `x3.mat`，运行程序，得到以下结果：





(2) 绘制时频谱图来体现该信号频谱随时间的变化情况。(设置不同参数，画出宽带、窄带谱图，进行对比分析)

在宽带谱图中，窗口大小设置为 $\text{window} = 256$ 。

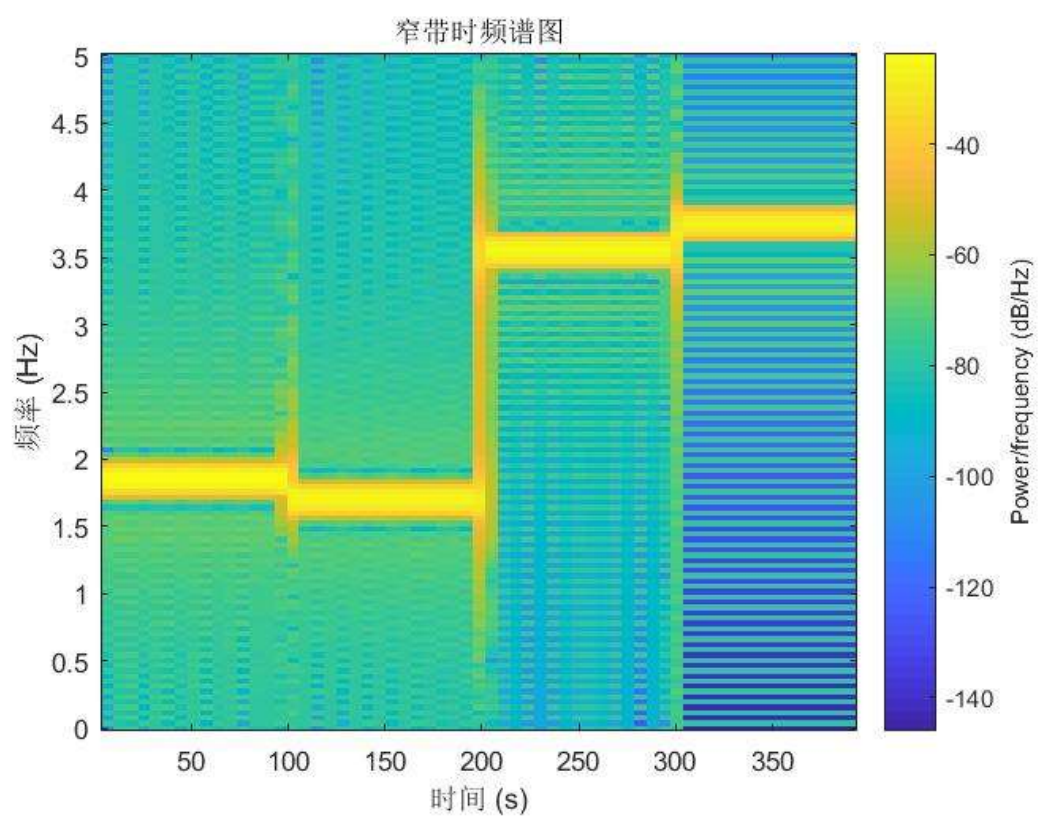
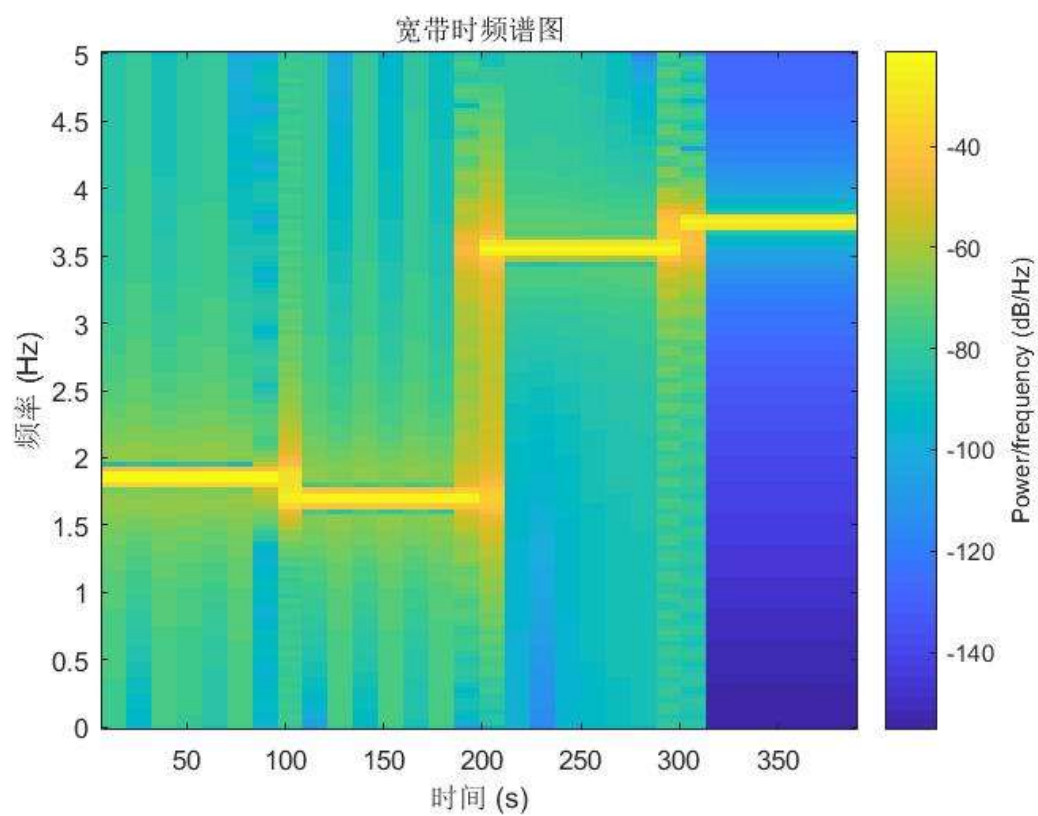
在窄带谱图中，窗口大小设置为 $\text{window} = 128$ 。

在宽带谱图中，重叠大小设置为 128。

在窄带谱图中，重叠大小设置为 64。

在两个谱图中，FFT 点数都设置为 256。

采样频率设置为 $F_s = 10000 \text{ Hz}$ (10 kHz)。



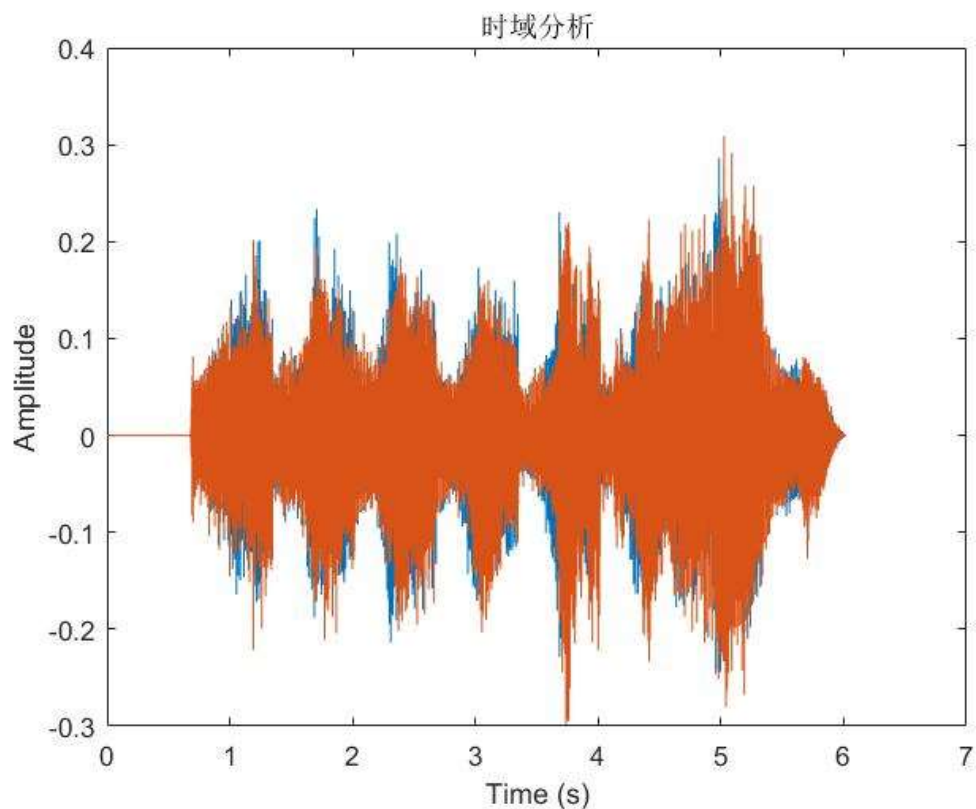
相同点：原信号是由多个不同的频率分量叠加而成。从频谱图中可以看出，随时间推移，信号的主频在发生较大变化。

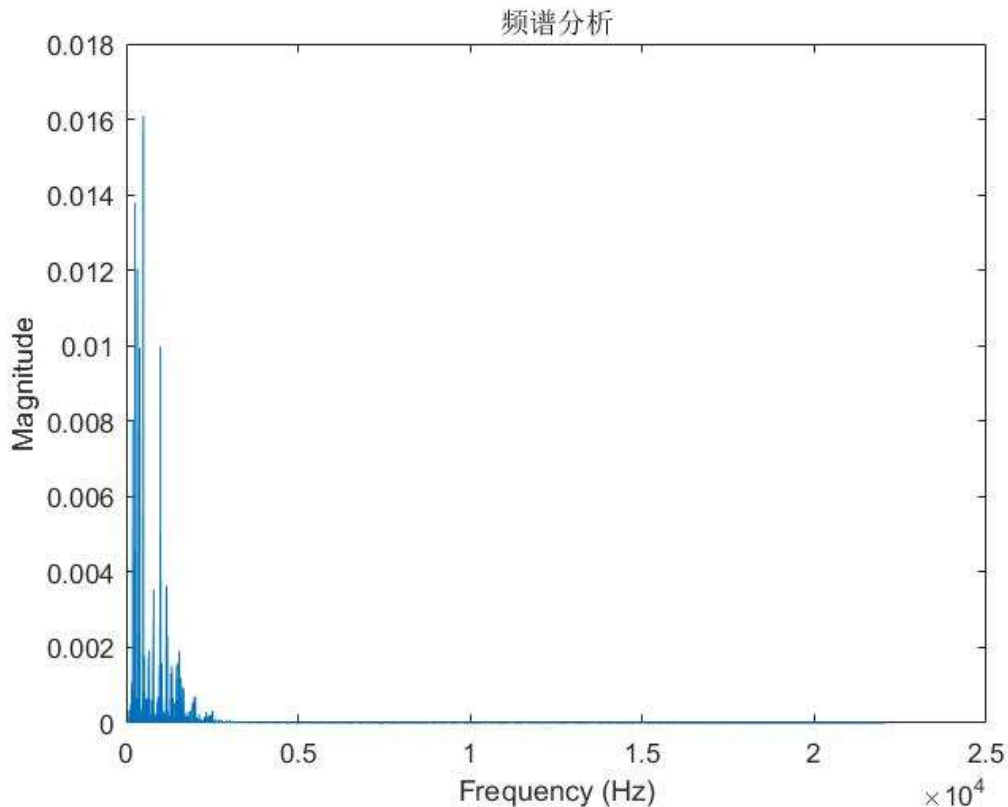
不同点：宽带信号包含较宽的频率范围，通常包含多个频率成分，可以覆盖从非常低频到高频的广泛频带。例如，音频信号、视频信号和许多通信信号通常被认为是宽带信号。窄带信号包含较窄的频率范围，通常只有一个或少数几个频率成分。这类信号的频谱较为集中，例如，某些无线电通信中的信号可能只在一个非常窄的频带内传输。宽带使用较大的窗口大小进行时频分析，可以提供更好的频率分辨率，但时间分辨率较低。这意味着可以更精确地识别信号中的频率成分，但难以捕捉快速变化的事件。窄带使用较小的窗口大小进行时频分析，可以提供更好的时间分辨率，但频率分辨率较低。这有助于捕捉信号中的快速变化，但对于精确识别频率成分则不够精确。

4. 实验四

请尝试获取一种一维时域信号样本（如音频、通信、震动、温度等），尝试利用 MATLAB 编程获得其频谱并分析其特点。

截取一段歌曲前几秒，命名为“music.mp3”，对此段音频进行频域分析，得到时域图和频谱图：





频谱图显示了音频信号在不同频率上的幅度分布。图中可以看到，大部分能量集中在低频区域，尤其是接近 0 Hz 的部分。在低频区域，频谱图中有多个明显的峰值，这些峰值代表了音频信号中的主要频率成分。这些频率可能是音频信号中基频或谐波的体现。随着频率的增加，幅度迅速下降，这表明音频信号在高频区域的能量较少。这种特性在许多自然声音中都很常见，因为大多数声音的高频成分相对较弱。在高频区域，频谱图的幅度非常低，这可能表明音频信号的噪声水平较低，或者高频成分在信号中不占主导地位。

五. 结论与总结

本次实验的目的在于运用离散傅里叶变换（DFT）对信号开展频谱分析，以此来深化对 DFT 在信号处理中应用的理解，特别是在无线通信领域，如频谱资源的分配和跳频技术等方面。实验内容广泛，包括对信号在时域、频域以及时频域的分析，并通过 MATLAB 程序来实现对信号频率成分的提取和识别。

在实验的第一部分，我们将使用 DFT 来绘制信号的时域波形，计算其幅度谱，并自动识别信号中的前四个主要频率成分，这样的练习有助于我们更好地理解频谱分析的原理。第二部分实验将研究不同窗口长度对频谱分析的影响，特别是窗口长度如何影响频率分辨率，我们将发现较长的窗口可以提高频率识别的精确度，但同时也会提升计算的复杂性。第三部分实验将通过时频谱分析来比较宽带和窄带信号之间的差异，这将帮助我们理解在频率分辨率和时间分辨率之间需要做出的权衡。在实验的最后一部分，我们将对音频信号进行频谱分析，探讨音高与频率之间的联系，并通过实际的音频样本来验证这一关系，这将加深我们对频谱分析在音频处理领域应用的认识。

综合实验结果，DFT 作为一种在信号处理中极为重要的工具，能够有效地提取信号的频率成分，对于无线通信和音频分析等关键领域提供了不可或缺的技术支持。

六. 参考文献

- [1]俞一彪,孙兵.数字信号处理.东南大学出版社
- [2]和二斌,马庆修,张文杰.Matlab 在信号与系统课程教学中的应用研究[J].电脑知识与技术,2024,20(08):138-142.DOI:10.14004/j.cnki.ckt.2024.0328.
- [3]杨桃丽,于瀚雯.基于 MATLAB 的数字信号处理综合课程实验[J].实验科学与技术,2024,22(01):57-61+67.

附录:

实验一代码 x1.m

```
1. % 加载数据文件
2. load('x1.mat');
3. signal = x1;
4.
5. N = 1000; % 窗口长度
6. if length(signal) > N
7.     signal = signal(1:N); % 如果信号长度超过窗口长度，截取前 N 个样本
8. elseif length(signal) < N
9.     signal = [signal; zeros(1, N - length(signal))]; % 如果信号长度不足窗口长度，补零
10. end
11.
12. % 画出信号时域波形
13. Fs = 10000;
14. t = (0:length(signal)-1)/Fs;
15. figure;
16. plot(t, signal);
17. title('信号时域波形');
18. xlabel('时间 (s)');
19. ylabel('幅度');
20.
21. % 计算信号的幅度谱
22. Y = fft(signal); % 快速傅里叶变换
23. P2 = abs(Y/length(signal)); % 双边谱
24. P1 = P2(1:N/2+1); % 单边谱
25. P1(2:end-1) = 2*P1(2:end-1); % 由于对称性，只取一半
26. f = Fs*(0:(N/2))/N; % 频率向量
27.
28. % 画出信号幅度谱
29. figure;
30. plot(f, P1);
31. title('信号幅度谱');
32. xlabel('频率 (Hz)');
33. ylabel('幅度');
34.
35. % 识别四个最为主要的频率成分
36. [~, idx] = sort(P1, 'descend');
37. main_freqs = f(idx(1:4)); % 返回四个最大的频率值
38.
39. % 输出四个最为主要的频率成分
40. disp('四个最为主要的频率成分 (Hz) :');
```



```
41. disp(main_freqs);
```

实验三代码 x3.m

```
1. % 加载数据文件
2. load('x3.mat');
3. signal = x3;
4. Fs = 10000;
5. N = 4000;
6.
7. % 时间向量
8. t = (0:N-1)/Fs;
9.
10. % 画出信号时域波形
11. figure;
12. plot(t, signal);
13. title('信号时域波形');
14. xlabel('时间 (s)');
15. ylabel('幅度');
16.
17. % 计算信号的 FFT
18. Y = fft(signal);
19. P2 = abs(Y/N);
20. P1 = P2(1:N/2+1);
21. P1(2:end-1) = 2*P1(2:end-1);
22. f = Fs*(0:(N/2))/N; % 频率向量
23.
24. % 画出信号幅度谱
25. figure;
26. plot(f, P1);
27. title('信号幅度谱');
28. xlabel('频率 (Hz)');
29. ylabel('幅度');
30.
31. % 绘制宽带谱图
32. window = 256; % 窗口大小
33. noverlap = window - 128; % 重叠大小
34. figure;
35. spectrogram(signal, window, noverlap, 256, Fs, 'yaxis');
36. title('宽带时频谱图');
37. xlabel('时间 (s)');
38. ylabel('频率 (Hz)');
39.
```

```
40. % 绘制窄带谱图
41. window = 128; % 窗口大小
42. noverlap = window - 64; % 重叠大小
43. figure;
44. spectrogram(signal, window, noverlap, 256, Fs, 'yaxis');
45. title('窄带时频谱图');
46. xlabel('时间 (s)');
47. ylabel('频率 (Hz)');
```

实验四代码 x4.m

```
1. % 读取音频文件
2. [audioIn, Fs] = audioread('music.mp3');
3.
4. % 绘制时域波形
5. t = (0:length(audioIn)-1)/Fs; % 时间向量
6. figure;
7. plot(t, audioIn);
8. title('时域分析');
9. xlabel('Time (s)');
10. ylabel('Amplitude');
11.
12. % 计算频谱
13. Y = fft(audioIn);
14. L = length(audioIn); % 信号长度
15. P2 = abs(Y/L); % 双边频谱
16. P1 = P2(1:L/2+1); % 单边频谱
17. P1(2:end-1) = 2*P1(2:end-1); % 由于对称性，只取一半
18.
19. % 绘制频谱图
20. f = Fs*(0:(L/2))/L; % 频率向量
21. figure;
22. plot(f, P1);
23. title('频谱分析');
24. xlabel('Frequency (Hz)');
25. ylabel('Magnitude');
```