**实验报告**

姓名：谌梓轩 专业：电子科学与技术 学号：3210105209

课程名称： 信息与电子工程导论 任课老师：周成伟

实验名称：基于MATLAB的信号频谱分析 实验日期：2022.3.1

**1 实验目的和要求**

**1.1 实验目的**

**（1）初步学习MATLAB的信号函数分析的流程和方法**

**（2）使用FFT函数进行频谱分析**

**（3）学会时域、频域之间的转换**

**1.2 实验要求**

**（1）构建信号，至少含有3个频率分量**

**（2）利用MATLAB分析频谱，并进行适当处理**

**（3）观察信号变化**

**（4）注意采样频率和采样点数的影响**

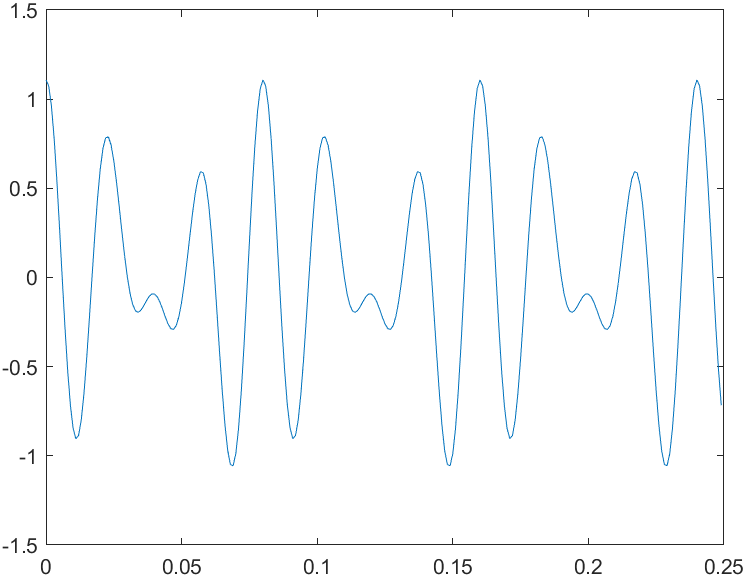
**2 实验原理**

1. **通过三角函数和其他函数相加构建信号**
2. **通过MATLAB绘制图像**
3. **利用randn函数与时间关联添加噪音**
4. **利用fft傅里叶变化得出频谱**

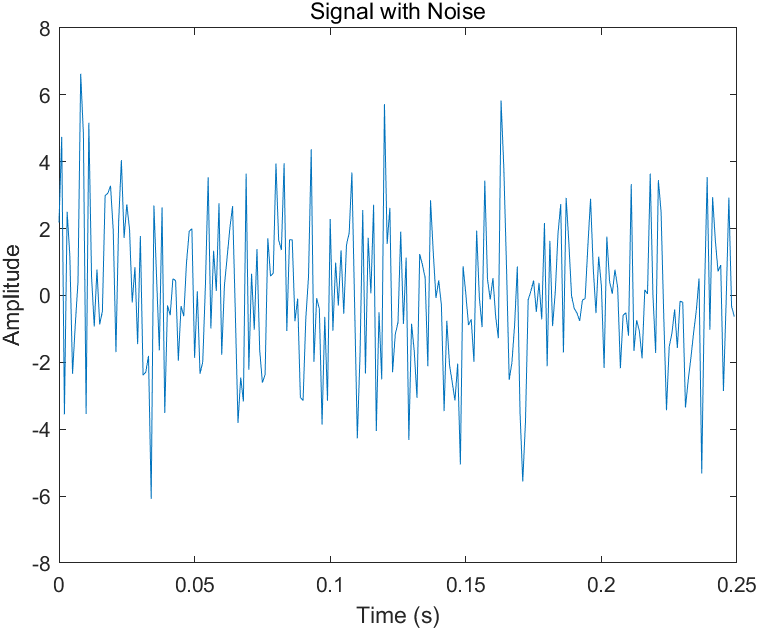
**3 实验内容**

1. **设置采样频率、周期及信号长度**
2. **构建信号函数**
3. **绘制图像**
4. **添加噪声**
5. **绘制添加噪声后的图像**
6. **调用fft函数**
7. **计算谱密度和单侧频谱**
8. **设置采样点数并绘制图像**
9. **对原信号调用fft函数**
10. **绘制图像**
11. **对添加噪音的信号进行傅里叶逆变换，绘制图像**
12. **去掉频谱中小于100Hz的部分**
13. **再次进行傅里叶逆变换并绘制图像**

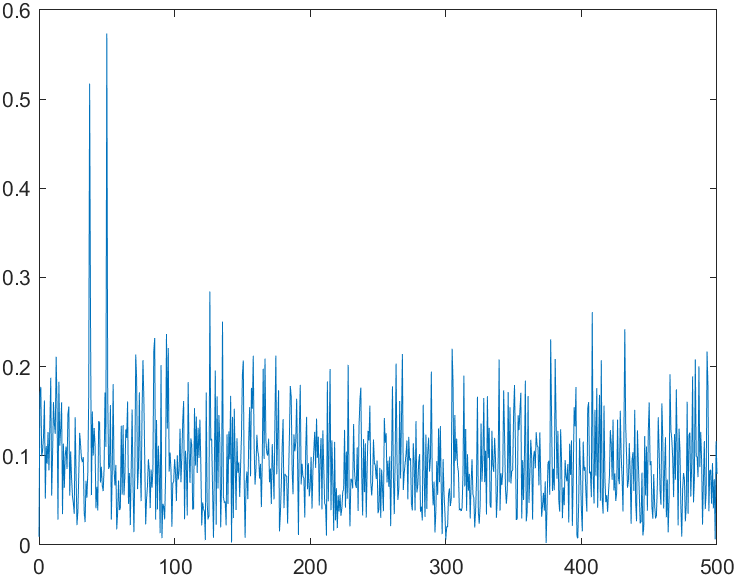
**4 实验结果和分析**



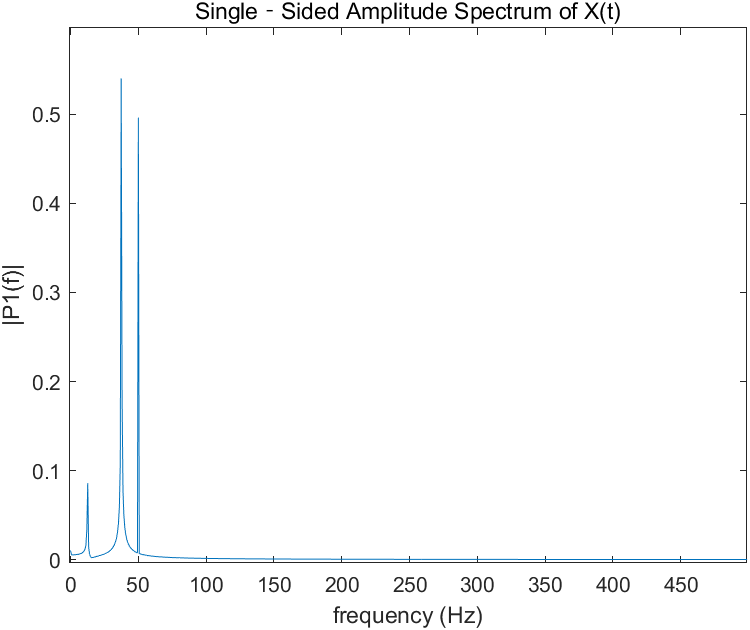
由函数s=0.5\*cos(100\*pi\*t)+0.9\*sin(50\*pi\*t)/t+0.1\*sin(25\*pi\*t)+0.6\*cos(75\*pi\*t)构成的信号，由于sin(50\*pi\*t)/t函数的存在，函数是非周期的



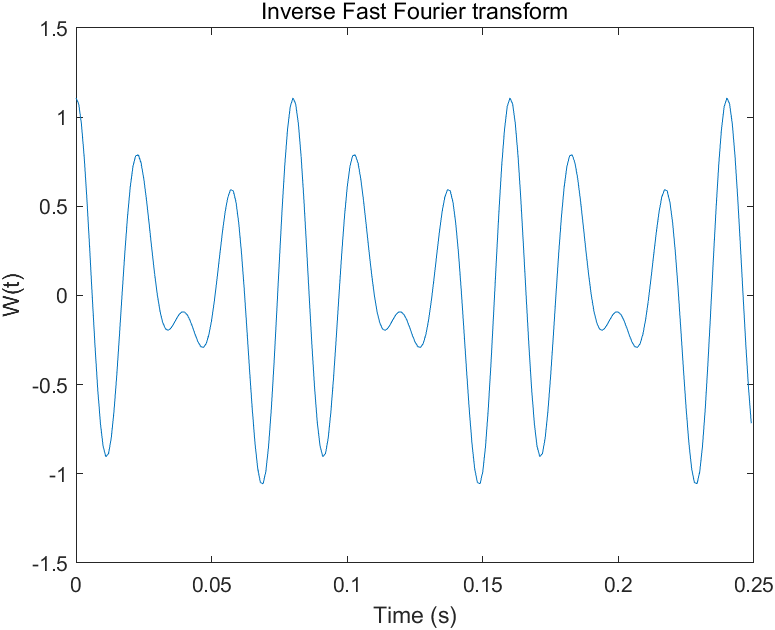
通过与时间相关联的噪声的干扰，信号变得非常复杂，与原信号相差极大。



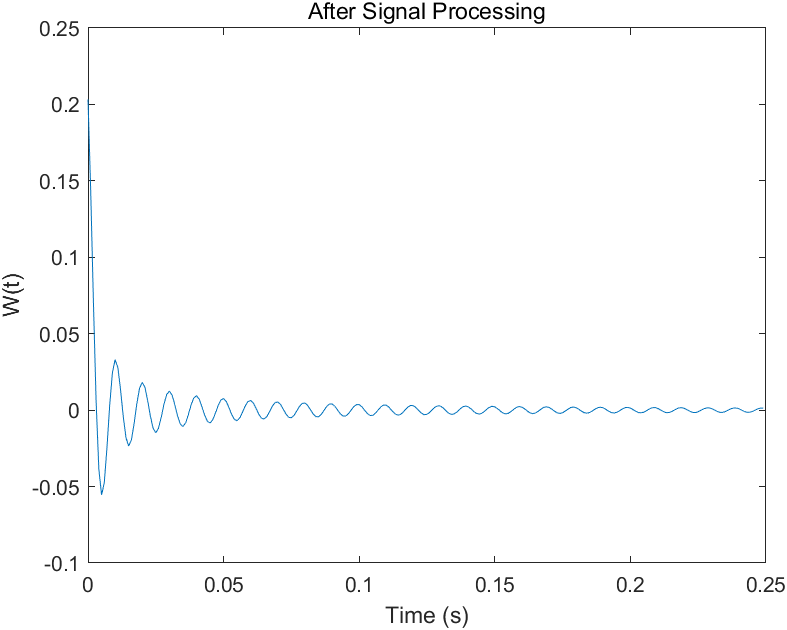
对受到干扰的信号进行傅里叶变换后，所产生频谱连续且相当复杂，并且，原信号中100Hz、75Hz频率等信号对应的幅值也收到了干扰。



对元信号进行傅里叶变化，可以发现25Hz、50Hz的幅值被很好地呈现出来，且由于sin(t)/t函数地存在，频谱也是连续地，并对其他频率地函数地幅值造成了影响。



对原函数的频谱进行傅里叶逆变换，得到了和原函数时域内信号1一样的信号图像



去除小于100Hz的部分后，取其实部绘制图像

**5 实验结论**

**对时域内的信号进行傅里叶变换后能够得到信号的频谱，且对信号的频谱进行傅里叶逆变换后，能够重新获得原信号的时域图像；干扰对于模拟信号的影响是非常大的，不仅会改变时域函数的各参数，也会改变频域上函数的状态；去掉信号频率较小的部分之后，函数会变得较为清晰，更加具有规律性。**

**6 源代码与分析**

>>Fs=1000;

>> T = 1/Fs;

>> N = 1500;

>> t = (0:N‐1)\*T;

>> t=(0:N-1)\*T;

>> s=0.5\*cos(100\*pi\*t)+0.9\*sin(50\*pi\*t)/t+0.1\*sin(25\*pi\*t)+0.6\*cos(75\*pi\*t);

>> figure(1) ***（原信号的构造及绘制）***

>> plot(t(1:250),s(1:250))

>> xlabel('Time (s)')

>> ylabel('Amplitude')

>> title('Time Domain Signal')

>> X = s + 2\*randn(size(t)); ***（添加噪声）***

>> figure(2)

>> plot(t(1:250),X(1:250))

>> xlabel('Time (s)')

>> ylabel('Amplitude')

>> title('Signal with Noise')

>> Y =fft(X); ***（对有噪声的信号进行傅里叶变化）***

>> P2 =abs(Y/N);

>> P1 = P2(1:N/2+1);

>> P1(2:end‐1) = 2\*P1(2:end‐1);

>> f = Fs\*(0:(N/2))/N;

>> figure(3)

>> plot(f,P1)

>> title('Single‐Sided Amplitude Spectrum of X(t)')

>> xlabel('frequency (Hz)')

>> ylabel('|P1(f)|')

>> Y =fft(s); ***（对原信号进行傅里叶变换）***

>> P2 =abs(Y/N);

>> P1 = P2(1:N/2+1);

>> P1(2:end‐1) = 2\*P1(2:end‐1);

>> f = Fs\*(0:(N/2))/N;

>> plot(f,P1)

>> title('Single‐Sided Amplitude Spectrum of X(t)')

>> xlabel('frequency (Hz)')

>> ylabel('|P1(f)|')

>> W =ifft(Y); ***（对原信号的频谱进行傅里叶逆变换）***

>> plot(t(1:250),W(1:250))

>> xlabel('Time (s)')

>> ylabel('W(t)')

>> title('Inverse Fast Fourier transform')

>> Y(1:151)=0;

>> Y(1350:1500)=0;

>> W =ifft(Y); ***（舍弃100Hz以下部分，进行傅里叶变换）***

>> W =real(W);

>> plot(t(1:250),W(1:250))

>> xlabel('Time (s)')

>> ylabel('W(t)')

>> title('After Signal Processing')