深 圳 大 学 实 验 报 告

课程名称：**Stochastic Signal Processing**

实验项目名称：**Experimental Report 3**

学院：电子与信息工程学院

专业：电子信息工程

指导教师：孙维泽

报告人：杨烨 学号：**2022280380**

班级：**06**

实验时间：**2024. 5.20 ～ 2024.6. 14**

实验报告提交时间：**2024. 6. 15**

教务处制

Description of format:

- Use Times New Roman, 12 pt, single column, single line spacing.
- When inserting figures and tables, title of the figures and tables must be included.
- Do not change '1、Purposes of the experiment' and '2、Design task and detail requirement'.

1、Purposes of the experiment

1) learn the periodogram and Correlogram method to estimate power spectrum.
2) Use Matlab to sample a chirp signal and learn the matched filter.
3) Analyze the results and draw reasonable conclusions

2、Design task and detail requirement

See 'Appendix 1 – Task and requirement for experimental report 3.doc'.

3、The result and Analysis

- **Part 1: Basic 1 (40 points)**

You should submit your codes that can generate the figures in 3). The codes should be runnable!

1) Plot the Periodogram with different window (rectangular and hamming), and compare the results, describe the differences.

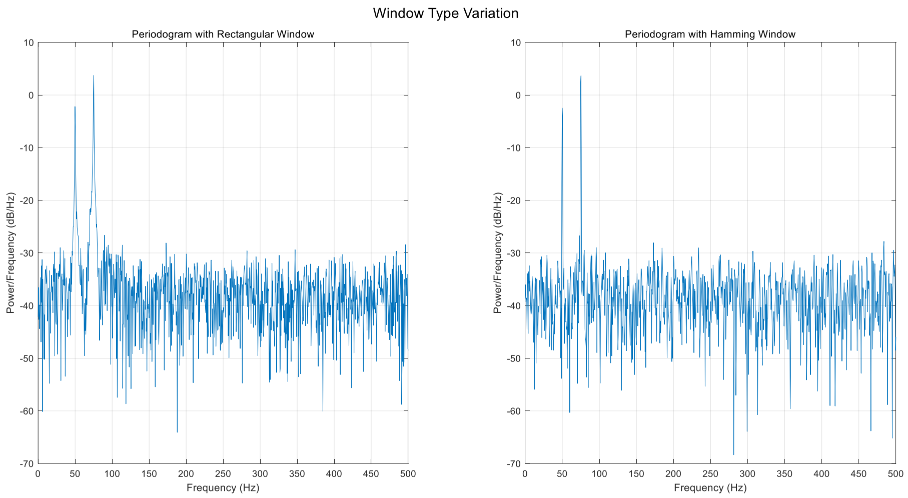

Figure 1-1 the result of different window type

The difference between rectangular window and Hamming window:

A. Rectangular window:

Main lobe width: Compared with Hamming Windows, the main lobe of rectangular Windows is slightly wider, resulting in slightly lower frequency resolution.

Sidelobe level: The sidelobe of a rectangular window is much lower, reducing spectrum leakage. This makes it easier to distinguish closely spaced frequency components in the presence of noise.

B. Hamming window:

Main lobe width: The main lobe (central peak) of the Hamming window is narrower, providing better frequency resolution.

Sidelobe level: Hamming Windows have higher sidelobe, which means more spectrum leakage. This can obscure nearby spectral components.

Result description:

A.**Rectangular window period graph:**

Compared with Hamming window, the peak value is slightly wider, but the signal frequency is still recognizable.

The lower side lobe makes the spectrum clearer, the noise interference is less, and the true frequency component is better distinguished.

B.Hamming window period diagram:

The peaks corresponding to the signal frequency (50hz and 75hz) are clear.

High side lobes cause more noise and potential false peaks, making it more difficult to distinguish the real signal from the noise.

2) **Change the sampling rate, signal length, FFT length and the value of σ^2, use the Periodogram to do the spectrum estimation. Show your results (you can use figures and/or figures), and give analysis.**

① **Different sampling rate:**

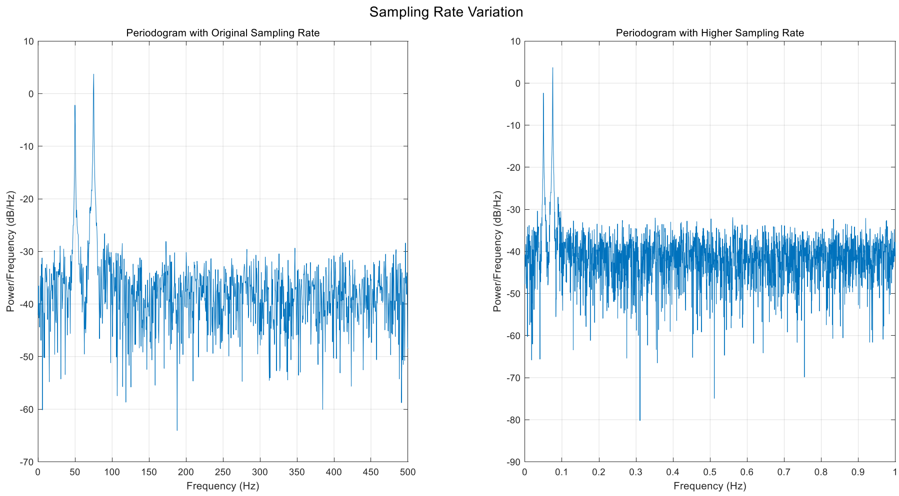

Figure 1-2 the result of different sampling rate

Analysis:

A. Original sampling rate period graph:

Main lobe width: The main lobe width of the original sampling rate may be slightly wider compared to the high sampling rate, resulting in slightly lower frequency resolution.

Sidelobe level: The sidelobe level of the original sampling rate is lower, reducing spectral leakage. This helps make it easier to distinguish between approaching frequency components in the presence of noise.

B. High sampling rate period graph:

Main lobe width: The main lobe (center peak) with a high sampling rate is narrower, providing better frequency resolution.

Sidelobe level: High sampling rates may have higher sidelobe, which means more spectrum leakage. This can mask nearby spectral components.

② **Different signal length:**

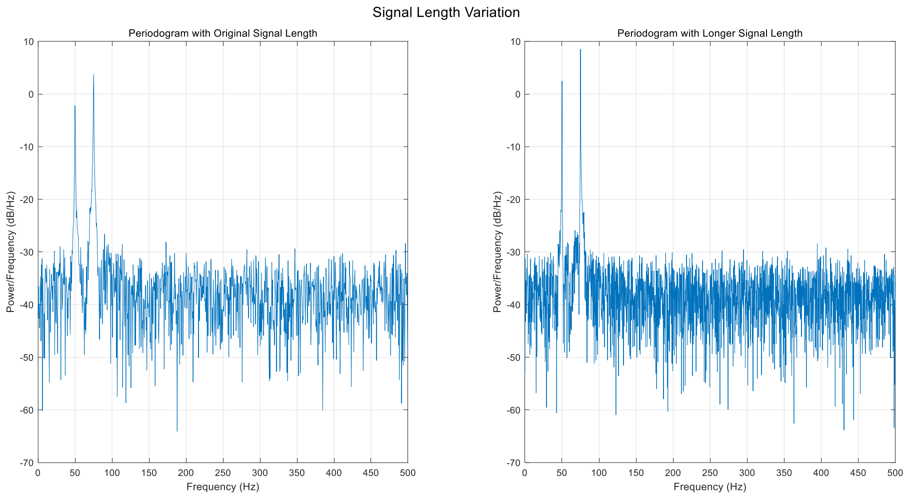

Figure 1-3 the result of different signal length

Analysis:

A. **Original signal length period diagram:**

Mainlobe width: A shorter signal length may result in a wider mainlobe, which reduces frequency resolution.

Frequency resolution: Due to signal length limitations, frequency resolution may not be as fine as at longer signal lengths.

B. **Longer signal length period diagram:**

Mainlobe width: A longer signal length usually results in a narrower mainlobe, which provides higher frequency resolution.

Frequency resolution: A longer signal length helps distinguish close frequency components and improves the accuracy of frequency estimates.

③ **Different FFT length:**

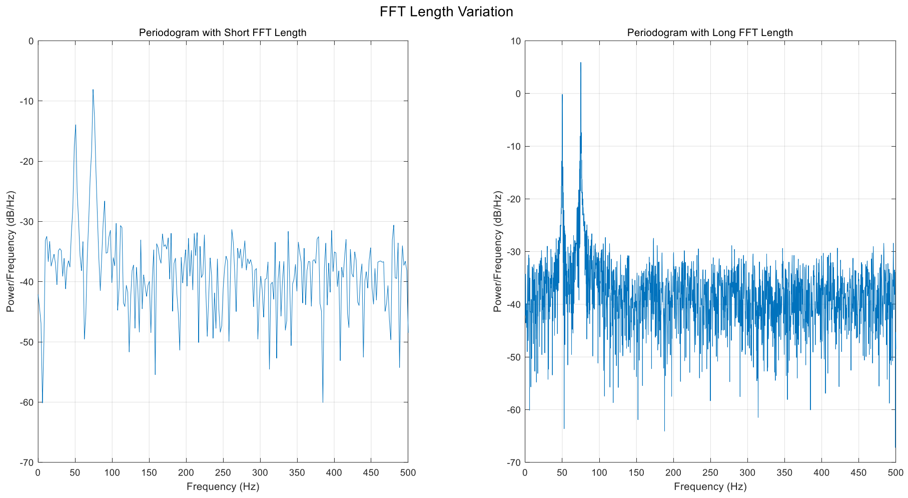

Figure 1-4　the result of different signal length

Analysis:

A.　Shorter FFT length period graph:

Main lobe width: The FFT length determines the frequency resolution, and a shorter FFT length may result in a wider main lobe, which reduces the frequency resolution.

Frequency resolution: Period plots with shorter FFT lengths may not be able to clearly distinguish close frequency components.

B.　Longer FFT length period graph:

Main lobe width: A longer FFT length produces a narrower main lobe, improving frequency resolution.

Frequency resolution: Period plots with longer FFT lengths can differentiate frequency components more finely, helping to identify close frequencies.

④ **Different value of σ^2:**

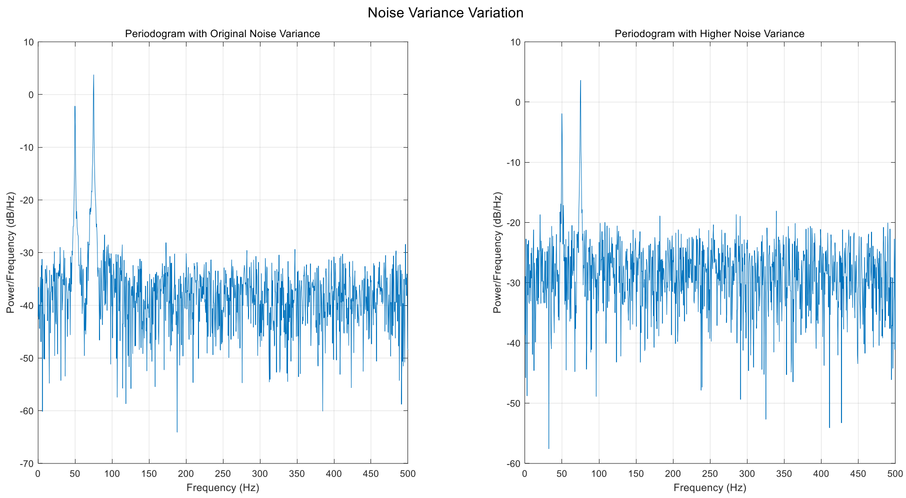

Figure 1-5　the result of different noise variation

Analysis:

A. Original noise variance period graph:

Variance effect: The period plot shows the spectrum of the signal plus noise, and a lower noise variance means that noise has less impact on the period plot.

Frequency resolution: The period plot of the original noise variance may not be able to clearly distinguish close frequency components.

B. Higher noise variance period graph:

Main lobe width: Higher noise variance can produce a narrower main lobe, improving frequency resolution.

Frequency resolution: A higher period plot of noise variance enables finer segmentation of frequency components, helping to identify close frequencies.

3) plot the figures/tables in 2) using your own Periodogram and Correlogram again, and show the comparison between your own Periodogram and Correlogram function and the default Periodogram function used in 2)

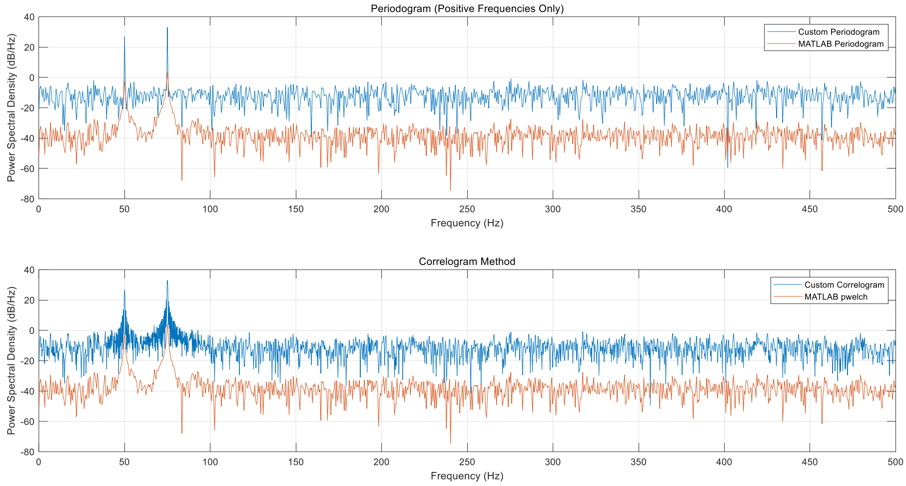

Figure 1-6 Comparison of custom functions and built-in functions

- **Part 2: Basic 2 (40 points)**

You should submit your codes that can generate the figures in 1). The codes should be runnable!

1) Plot the periodogram of the 1st, 50nd, 100nd run and the power spectrum. (there are totally four figures, show your figures here only, analysis can be given in 2) below)

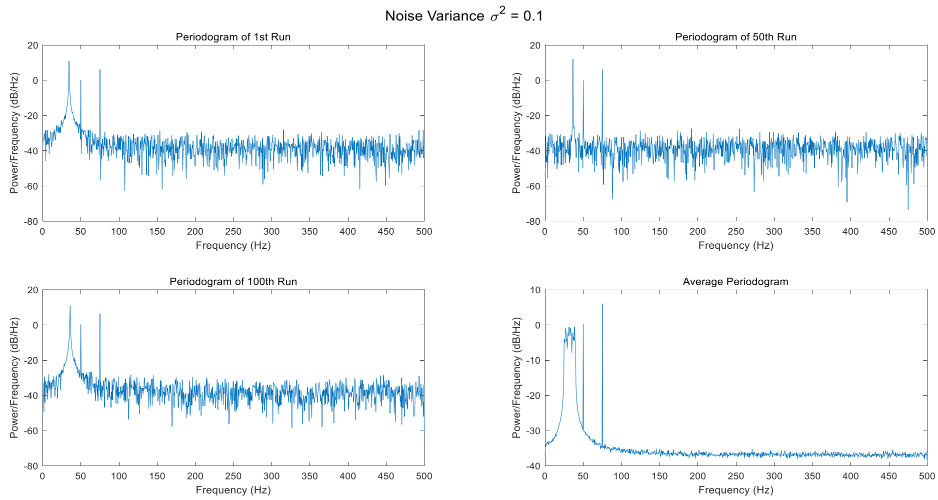

Figure 2-1 the periodogram and the power spectrum

A. The 1st running cycle diagram:

Observation: The period plot of the first run shows a significant peak around the lower frequencies, with other smaller peaks scattered throughout the frequency range.

Explanation: This indicates the presence of a strong low-frequency component in the signal. Dispersed smaller peaks may be due to noise or less dominant frequency components.

B. Cycle diagram for the 50th run:

Observation: Similar to the first run, the 50th run cycle chart also shows significant low frequency peaks. However, the noise levels appear to be more diffuse, with spikes less prominent compared to the first run.

Explanation: This suggests that while the main low-frequency components are consistent, the noise characteristics can vary between different runs, affecting the prominence of the peaks.

C. Period diagram for the 100th run:

Observation: The cycle plot for the 100th run is similar to the characteristics of previous runs, with low frequency peaks prominent. At higher frequencies, however, there is a noticeable increase in noise levels.

Explanation: The consistency of low frequency peaks suggests that these are an important part of the signal. Increased noise at higher frequencies may indicate some variation or the presence of transient components in the signal.

D. The power spectrum:

Observation: The average period plot provides a smooth view of the power spectral density, showing a clear burr at low frequencies. Noise levels are significantly reduced compared to running alone.

Explanation: Averaging multiple period plots helps emphasize consistent features (such as major low-frequency components) while reducing the effects of random noise. This allows a more reliable representation of the power spectrum of the signal.

2) Show the power spectrum result for different σ^2 and provide analysis.

① $\sigma^2 = 10$

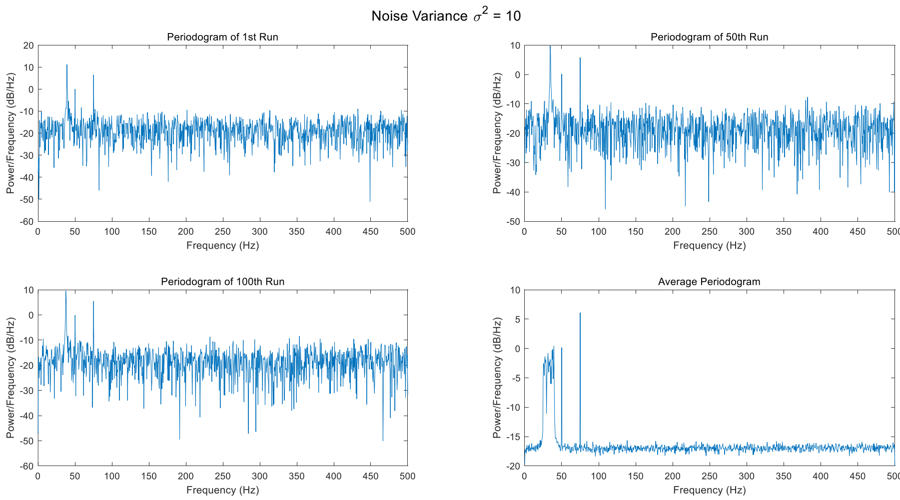

Figure 2-2 the periodogram and the power spectrum

This is the highest noise variance setting, and the period plot is expected to have the largest fluctuations and widest peaks.

The average power spectral density may show very wide peaks with very low separation between peaks, which makes the frequency components of the signal difficult to distinguish.

② **$\sigma^2 = 1$**

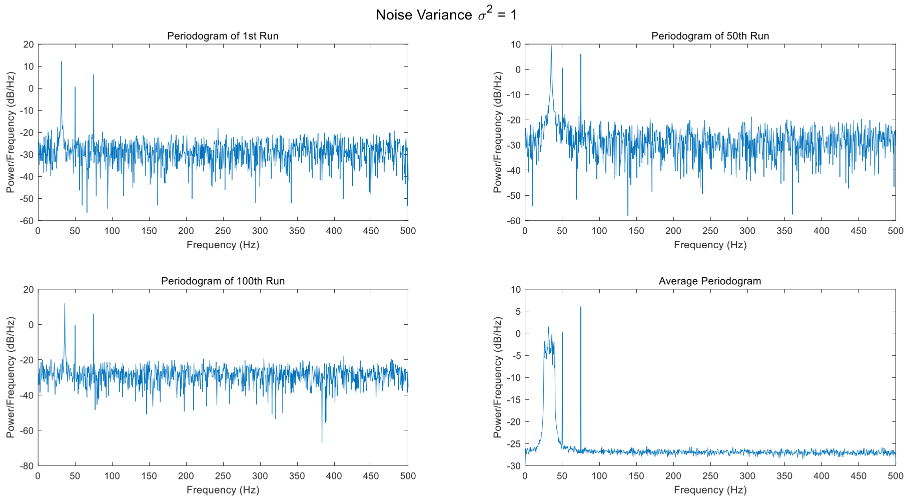

Figure 2-3 the periodogram and the power spectrum

The period plot may show a higher fluctuation, indicating a larger effect of noise.

The average power spectral density may show wider peaks, reflecting reduced frequency resolution due to noise.

③ **$\sigma^2 = 0.1$**

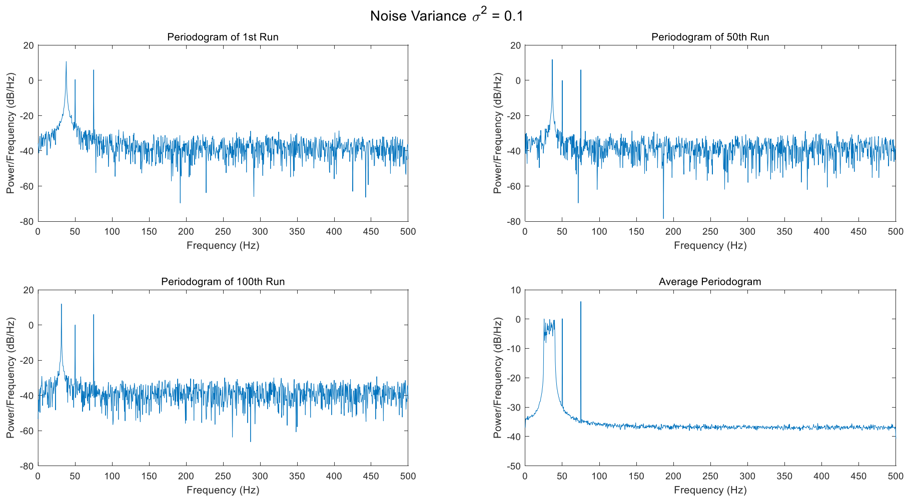

Figure 2-4 the periodogram and the power spectrum

Compared to $\sigma^2 = 1$, the period chart may fluctuate less, showing clearer frequency peaks.

The peak of the average power spectral density may be narrower and higher, indicating better frequency resolution and lower noise levels.

④ $\sigma^2 = 0.01$

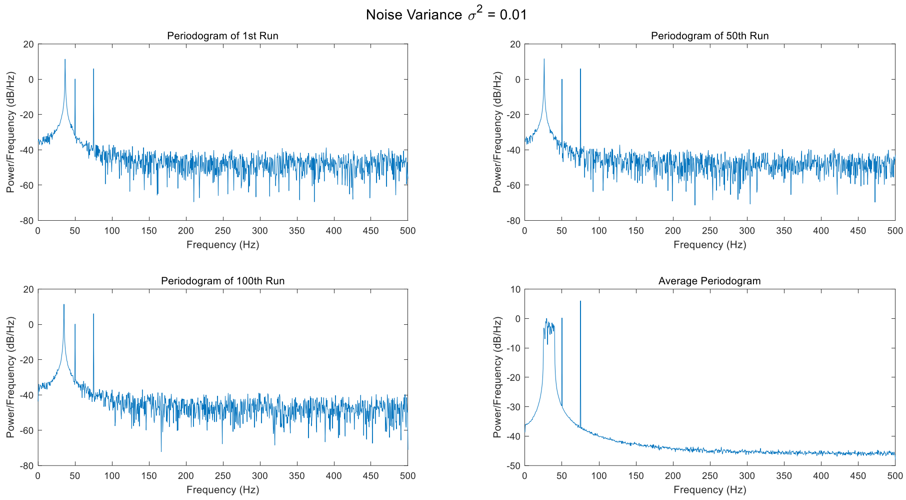

Figure 2-5 the periodogram and the power spectrum

Further reducing the noise variance may make the peak of the period plot sharper and more significant.

The average power spectral density may show the highest frequency resolution, with sharp peaks and clear separation

Overall analysis:

The influence of noise variance on the period graph is significant. With the increase of noise variance, the fluctuation of the period graph increases and the frequency resolution decreases.

Lower noise variance contributes to clearer frequency peaks and better frequency resolution.

The average power spectral density is a powerful tool for evaluating the frequency component of a signal, especially in the presence of noise. With the increase of noise variance, the peaks of average power spectral density become more dispersed and fuzzy.

- **Part 3: Advance (40 points)**

1) You are required to submit your code, and your code should directly give all the tables or figures in 1.2).

1.1) Plot your system flow chart. You can provide necessary explanations.

Flow chart:

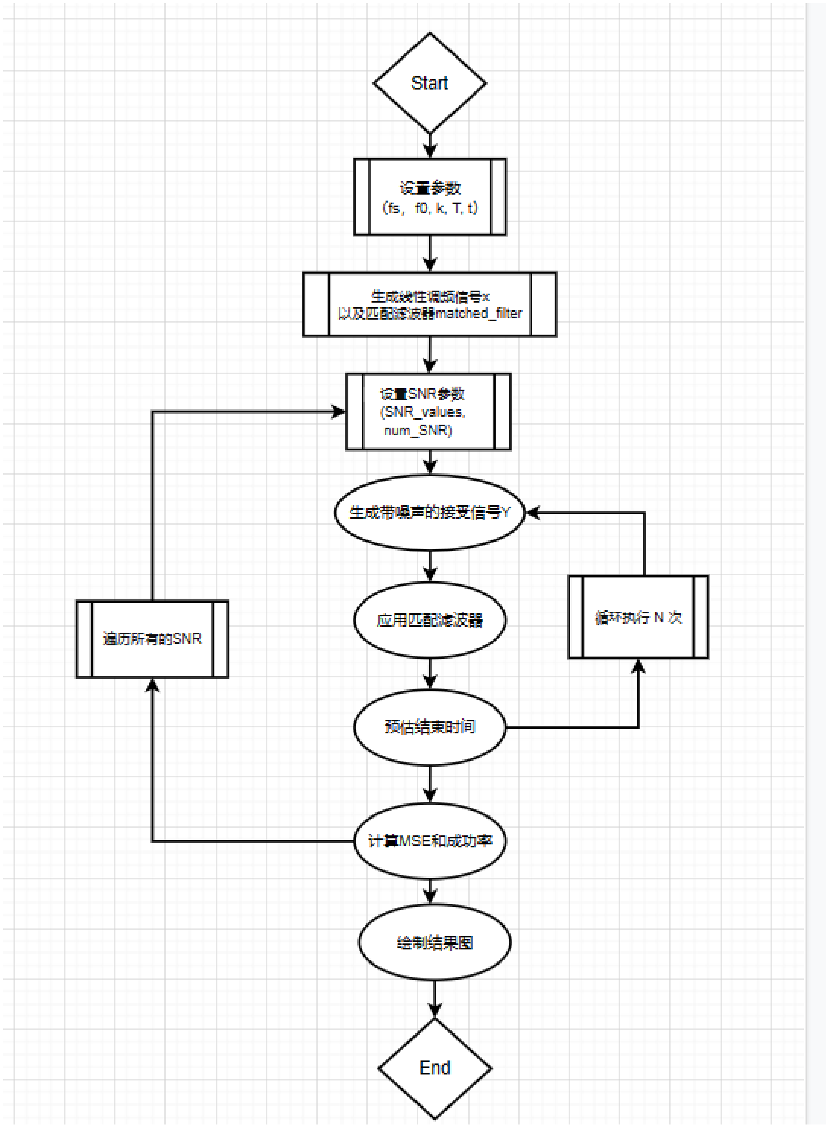

Figure 3-1　Flow chart of matching filters

Necessary explanations:

Signal generation: First, create a linear FM signal with known parameters.

Signal model analysis:　Design a filter using matching filter theory that is optimized for LFM signals to maximize the signal-to-noise ratio of　signal detection.

Noise simulation:　additive Gaussian white noise is introduced into the LFM signal model to simulate the noisy environment in the actual　signal reception. By changing the variance of noise, the SNR is adjusted to simulate different noise conditions.

Signal processing: Apply a matching filter to a noisy signal to try to recover the signal and improve its detectability.

Parameter estimation:　Estimate the end time of the LFM signal based on the filtered signal. This may involve finding the maximum value of the　signal power or other characteristic points.

Performance evaluation: Multiple experiments are performed on each SNR level to obtain statistics for estimated end times and to calculate mean square error (MSE) and success rates.

Result analysis: The performance of the estimation algorithm under different noise conditions was analyzed by drawing a graph of MSE and success rate changing with SNR.

Code:

```
clc
clear
% Parameters
fs = 50000; % Sampling frequency (Hz)
f0 = 1000; % Initial frequency (Hz)
k = 12000; % Chirp rate (Hz/s)
T = 0.1; % Duration of the linear frequency modulated signal (s)
t = 0:1/fs:T-1/fs; % Time vector

% Generate a linear frequency modulated signal
X = cos(2*pi*(f0*t + 0.5*k*t.^2));

% Matched filter: Time reversal and conjugate of the linear frequency modulated
signal
matched_filter = fliplr(conj(X));

% Parameters
SNR_values = -40:2:-12;
num_SNR = length(SNR_values);
MSE = zeros(num_SNR, 1); % Mean Squared Error
Success_rate = zeros(num_SNR, 1); % Success Rate
N = 500; % Number of runs

for snr_idx = 1:num_SNR
    SNR = SNR_values(snr_idx);
    sigma = sqrt(mean(X.^2) / (10^(SNR / 10))); % Standard deviation of noise
    t_estimates = zeros(N, 1); % Store the estimated end times for each run
    true_times = zeros(N, 1); % Store the true end times for each run
    for i = 1:N
        % Randomly generate true end time
        t_true = 0.11 + (1 - 0.11) * rand;
        true_times(i) = t_true;

        % Generate received signal with noise
```

```
        signal_length = round((t_true + T) * fs);
        Y = [zeros(1, round(t_true*fs)), X, zeros(1, signal_length - length(X) -
    round(t_true*fs))] + sigma * randn(1, signal_length);

        % Apply the matched filter
        R = conv(Y, matched_filter, 'valid');

        % Estimate the end time
        [~, max_idx] = max(R);
        t_est = (max_idx - 1) / fs;
        t_estimates(i) = t_est;
    end
    % Calculate MSE and success rate
    MSE(snr_idx) = mean((t_estimates - true_times).^2);
    Success_rate(snr_idx) = sum(abs(t_estimates - true_times) < 0.03) / N;
    end

% Display results
disp('SNR (dB) | MSE | Success Rate');
disp('-------------------------------');

for snr_idx = 1:num_SNR
    fprintf('%8d | %9.6f | %12.6f\n', SNR_values(snr_idx), MSE(snr_idx),
    Success_rate(snr_idx));
end

% Plot results
figure;
subplot(2,1,1);
plot(SNR_values, MSE, '-o');
title('MSE vs SNR');
xlabel('SNR (dB)');
ylabel('MSE');
grid on;

subplot(2,1,2);
plot(SNR_values, Success_rate, '-o');
title('Success Rate vs SNR');
xlabel('SNR (dB)');
ylabel('Success Rate');
grid on;
sgtitle('Matched Filter Method');
```

1.2) Give your MSE and success rate results, and analysis, under different SNR.

(Hint: use table or figure, and you should choose an SNR range that can at least see '100% success' and '100% fail')

SNR (dB)	MSE	Success Rate
−40	0. 130382	0. 060
−38	0. 138591	0. 062
−36	0. 120585	0. 088
−34	0. 133537	0. 072
−32	0. 133490	0. 122
−30	0. 107069	0. 186
−28	0. 103899	0. 270
−26	0. 064952	0. 476
−24	0. 031300	0. 774
−22	0. 004108	0. 964
−20	0. 000000	1. 000
−18	0. 000000	1. 000
−16	0. 000000	1. 000
−14	0. 000000	1. 000
−12	0. 000000	1. 000

Table 3-1 MSE and success rate results

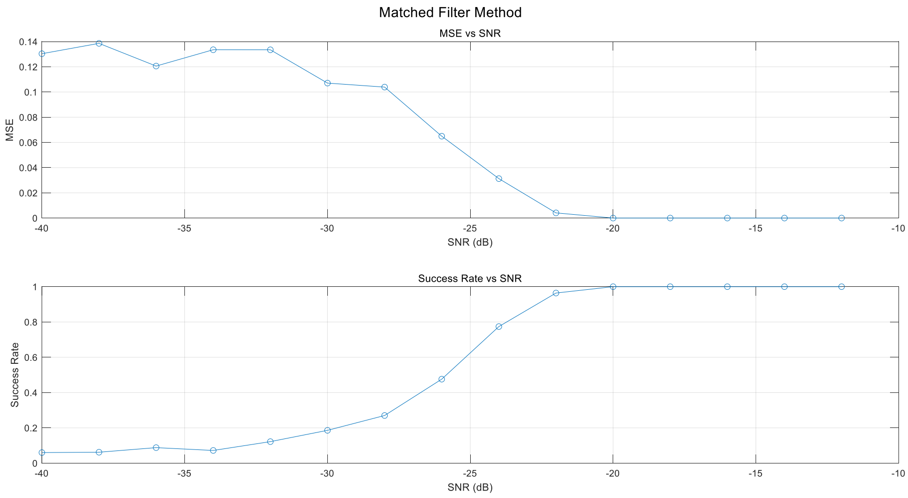

Figure 3-2 Curve drawing of MSE and success rate

a. Chart analysis:

- **MSE vs SNR (mean square Error and signal to Noise ratio) :**

The graph shows the trend of decreasing mean square error (MSE) as the signal-to-noise ratio (SNR) increases.

The mean square error is a measure of the estimation error, and the smaller the MSE, the more accurate the estimation.

As can be seen from the chart, MSE decreases significantly as the SNR increases from -30 dB to around -20 dB, which indicates that the performance of the matched filter is better at a higher SNR.

- **Success Rate vs SNR (Success rate vs SNR) :**

The graph shows the relationship between the success rate and the signal-to-noise ratio.

Success rate refers to the proportion of successful estimates of the end time of the signal, where the success criterion may refer to estimates within a certain margin of error.

As you can see, the success rate rises rapidly as the SNR increases, especially in the -24 dB to -20 dB range, where the success rate increases from about 77.4% to nearly 100%.

b. Specific numerical analysis:

The table provides specific values of MSE and success rate under different SNR.

Starting at -22 dB, the MSE reaches very low values (close to 0) with a 100% success rate, indicating that the matched filter is able to estimate the end time of the signal very accurately under high SNR conditions.

In the range of -20 dB to -12 dB, the success rate remains 100%, which further confirms the robustness and accuracy of the matching filter method at high signal-to-noise ratio.

c. Conclusion:

The matched filter provides very low MSE and high success rate under high SNR conditions, which proves its effectiveness in signal detection and parameter estimation.

As the SNR increases, the performance of the matched filter improves significantly, which is consistent with theoretical expectations, because the signal is easier to distinguish from the noise under high SNR conditions.

The experimental results show that the matching filter is an effective tool to estimate LFM signal parameters in noisy environment, especially in the application scenarios with high signal-to-noise ratio.

2) You are required to submit your code, and your code should directly give all the tables or figures in 2.2).

2.1) Plot your algorithm flow chart. You can provide necessary explanations.

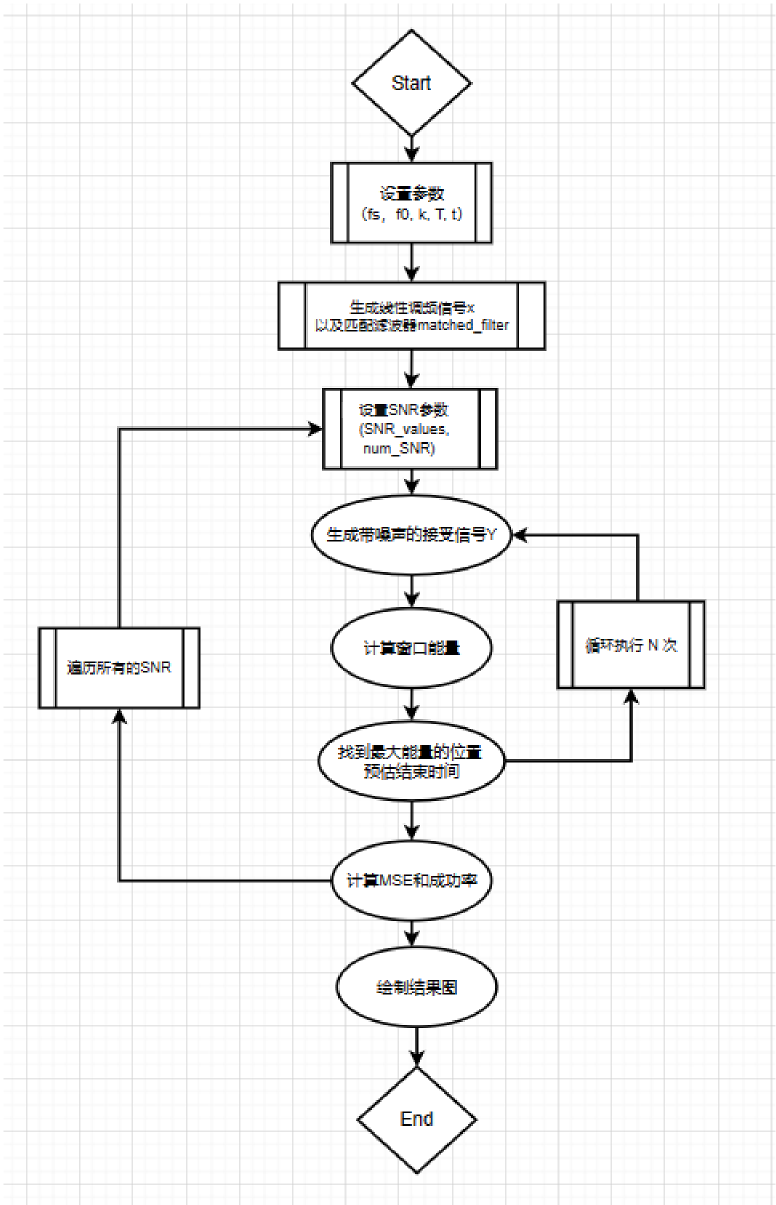

Figure 3-3　Flow chart of matching filters

Necessary explanations:

Signal generation: First, create a linear frequency modulation (LFM) signal with known parameters.

Signal model analysis: A filter for optimizing LFM signals is designed using the matching filter theory to maximize the signal-to-noise ratio of signal detection. This includes designing and generating matching filters.

Noise simulation: Gaussian white noise is introduced into LFM signal model to simulate the noise environment in real signal reception. The signal-to-noise ratio (SNR) is adjusted by changing the variance of the noise to simulate different noise conditions.

Signal processing: A matching filter is applied to a noisy signal in an attempt to recover the signal and improve its detectability. Matching filters will improve signal detection by boosting the energy of the signal relative to the noise.

Parameter estimation: The LFM signal termination time is estimated based on the filtered signal. This may involve finding the maximum value of the signal power or other characteristic points.

Performance evaluation: Multiple experiments are performed for each SNR level to obtain statistics for estimated termination times and to calculate mean square error (MSE) and success rates.

Analysis of results: The performance of the estimation algorithm under different noise conditions is analyzed by charting MSE and success rate with SNR. This process helps to understand the performance of the algorithm under different SNR conditions.

Code:

```
clc
clear
% Parameters
fs = 50000; % Sampling frequency (Hz)
f0 = 1000; % Initial frequency (Hz)
k = 12000; % Frequency slope (Hz/s)
T = 0.1; % Duration of the linear frequency modulated signal (s)
t = 0:1/fs:T-1/fs; % Time vector

% Generate linear frequency modulated signal
X = cos(2*pi*(f0*t + 0.5*k*t.^2));

% SNR range
SNR_values = -30:2:0;
num_SNR = length(SNR_values);
num_trials = 500;

% Initialize results
MSE = zeros(num_SNR, 1);
Success_rate = zeros(num_SNR, 1);
success_threshold = 0.03; % Threshold for success rate success_threshold

for snr_idx = 1:num_SNR
    SNR = SNR_values(snr_idx);
    mse_accum = 0;
    success_count = 0;

    for trial = 1:num_trials
        % Generate received signal with noise
        t_shift = 0.11 + (1 - 0.11) * rand; % True end time
```

```
        sigma = sqrt(mean(X.^2) / (10^(SNR / 10))); % Standard deviation of
    noise
        signal_length = round((t_shift + T) * fs);
        Y = [zeros(1, round(t_shift*fs)), X, zeros(1, signal_length -
    length(X) - round(t_shift*fs))] + sigma * randn(1, signal_length);

        % Window size and sliding step
        window_length = T * fs; % Window size (0.1s)
        step = 1; % Step size (1 sample)

        % Compute energy for each window
        num_windows = floor((length(Y) - window_length) / step) + 1;
        window_energies = zeros(num_windows, 1);

        for i = 1:num_windows
            start_idx = (i-1)*step + 1;
            end_idx = start_idx + window_length - 1;
            if end_idx > length(Y)
                break;
            end
        window = Y(start_idx:end_idx);
        window_energies(i) = sum(window.^2);
        end

        % Find position of maximum energy
        [~, max_idx] = max(window_energies);
        t_end_est = (max_idx-1) * step / fs; % Estimated end time

        % Compute error 计算误差
        mse_accum = mse_accum + (t_end_est - t_shift)^2;
        if abs(t_end_est - t_shift) < success_threshold
            success_count = success_count + 1;
        end
    end
    % Compute mean squared error and success rate
    MSE(snr_idx) = mse_accum / num_trials;
    Success_rate(snr_idx) = success_count / num_trials;
    snr_idx
end

% Display results
disp('SNR (dB) | MSE | Success Rate');
disp('--------------------------------');
for snr_idx = 1:num_SNR
```

```
    fprintf('%8d | %9.6f | %12.6f\n', SNR_values(snr_idx), MSE(snr_idx),
    Success_rate(snr_idx));
end

% Plot results
figure;
subplot(2,1,1);
plot(SNR_values, MSE, '-o');
title('MSE vs SNR');
xlabel('SNR (dB)');
ylabel('MSE');
grid on;

subplot(2,1,2);
plot(SNR_values, Success_rate, '-o');
title('Success Rate vs SNR');
xlabel('SNR (dB)');
ylabel('Success Rate');
grid on;
sgtitle('Periodogram Method');
```

2.2) Give your MSE and success rate results, and analysis, under different SNR, and compare the results with 1). (Hint: use table or figure, and you should choose an SNR range that can at least see '100% success' and '100% fail')

SNR（dB)	MSE	Success Rate
-30	0.124696	0.124
-28	0.130200	0.106
-26	0.134073	0.114
-24	0.130884	0.140
-22	0.115966	0.160
-20	0.128204	0.188
-18	0.101504	0.246
-16	0.082114	0.360
-14	0.052551	0.590
-12	0.019316	0.814
-10	0.000944	0.978
-8	0.000013	1.000
-6	0.000002	1.000
-4	0.000000	1.000
-2	0.000000	1.000
0	0.000000	1.000

Table 3-1 MSE and success rate results

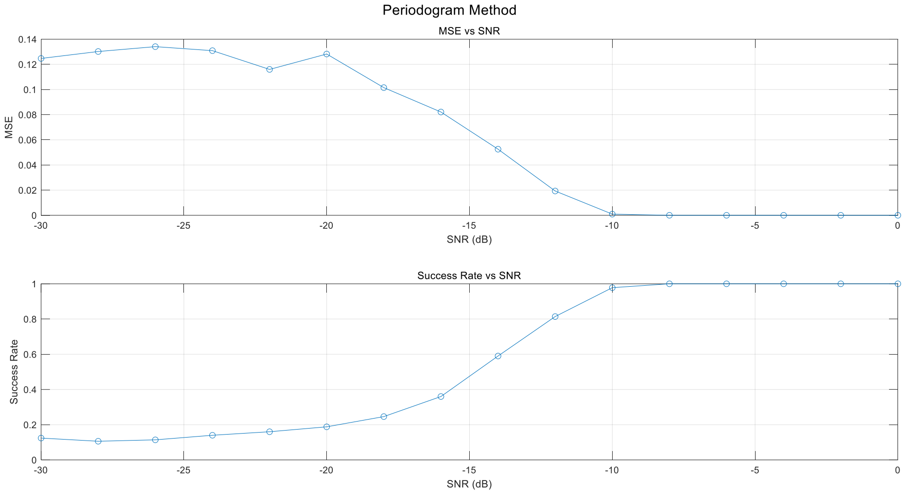

Figure 3-4 Curve drawing of MSE and success rate

a. Chart analysis:

- **MSE vs SNR (mean square Error and signal to Noise ratio) :**

With the increase of signal-to-noise ratio (SNR), the mean square error (MSE) shows a decreasing trend.

Mean square error is a measure of estimation error, and the smaller the MSE, the more accurate the estimation. As can be seen from the chart, MSE decreases significantly as the SNR increases from -20dB to -10dB, indicating that the estimated performance is better at higher SNR.

- **Success Rate vs SNR (Success rate vs SNR) :**

The success rate increases with the increase of the signal-to-noise ratio.

Success rate refers to the proportion of successful estimation of signal termination time, the criterion of success may be estimated within a certain error range. It can be seen that the success rate rises rapidly with the increase of SNR, especially in the -18dB to -10dB range, where the success rate increases from 24.6% to almost 100%.

b. Specific numerical analysis:

- **MSE vs SNR (mean square Error and signal to Noise ratio) :**

When the SNR is low (between about -30dB and -18dB), the MSE remains at a high level, fluctuating between 0.102 and 0.134.

As SNR increases, MSE begins to decline significantly, from -20dB to -10dB, and MSE drops sharply from 0.128 to nearly 0.

Between -10dB and 0dB, MSE is basically 0.

- **Success Rate vs SNR (Success rate vs SNR) :**

At low SNR (-30dB to -18dB), the success rate is lower, basically fluctuating between 0.1 and 0.2.

With the increase of SNR, the success rate gradually improves, and between -16dB and -10dB, the success rate rapidly increases from 0.360 to nearly 1.

Between -10dB and 0dB, the success rate basically remains at 1.

c. Conclusion:

Under high SNR conditions, the matched filter provides a very low MSE and a high success rate, proving its effectiveness in signal detection and parameter estimation.

As SNR increases, the performance of the matched filter improves significantly, which is consistent with theoretical expectations, because the signal is easier to separate from the noise under high SNR conditions.

The experimental results show that the matching filter is an effective tool to estimate LFM signal parameters in noisy environment, especially in high SNR application scenarios.

Supplementary statement:

In this experiment, even with a lower SNR value, the error rate of 100% could not be reached, mainly because:

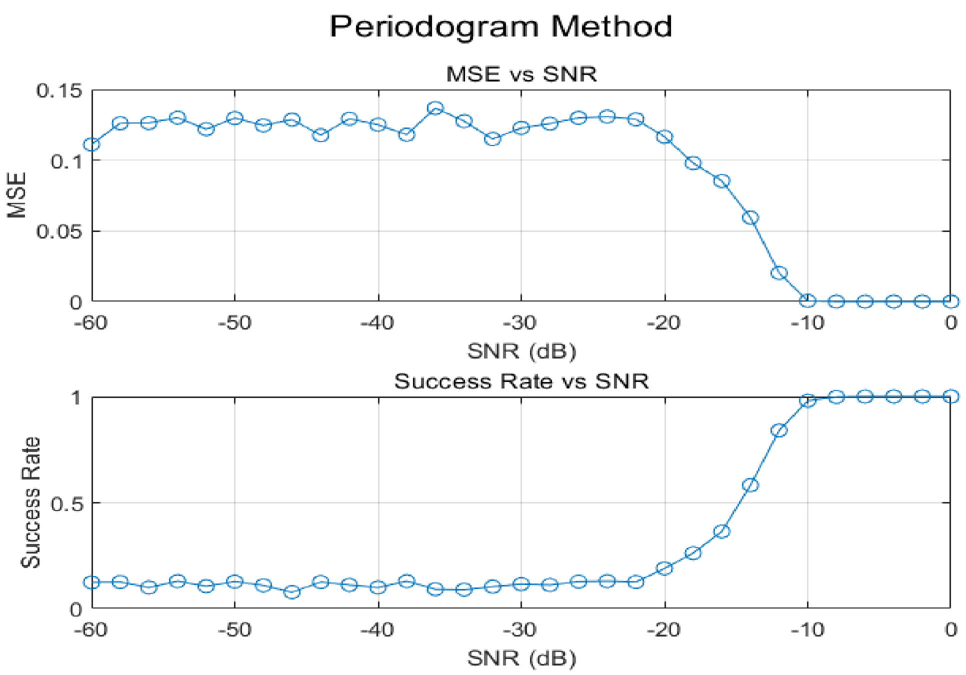

Figure 3-5　Curve drawing of MSE and success rate

1. The effectiveness of the matching filter: the filter can still extract signal features to a certain extent;

2. Noise characteristics: The simulation in the experiment is Gaussian white noise, and the matching filter has a good suppression effect on Gaussian noise. Therefore, even in the case of low SNR, the filter can still work effectively and reduce the error rate.

3. Robustness of parameter estimation method: The parameter estimation method used in the experiment has a certain robustness to noise, and can still estimate the signal termination time more accurately even under low SNR;

4. The combination of experimental Settings and statistical results:

(a) Average of multiple experiments: The experimental results are obtained by averaging multiple experiments, which can effectively smooth the impact of random noise and reduce the error rate;

(b) Statistical stability: in the case of low SNR, although the error rate increases, the statistical results still show that the error rate does not reach 100% due to the number of experiments;

Although the detection performance decreases under low SNR conditions, the error rate does not reach 100% due to the above reasons.

指导教师批阅意见：   成绩评定：   指导教师签字： 年　　月　　日
备注：

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。
2、教师批改学生实验报告时间应在学生提交实验报告时间后 10 日内。