Lab 6 Report

Digital Modulation

Author: B08901049 Yuan-Chia Chang
Instructor: Professor Hao-Chung Cheng
TA: Yuan-Pon Chen
Created: 12/3/2021(Fri)
Last edited: 12/8/2021(Wed)
Contact: b08901049@ntu.edu.tw
Collaborator: B08901002 Chen-Han Lin, B08901209 Yu-Hsiang Lin

Q1.

Please check the file

https://github.com/yuanchiachang/CommLab/blob/main/Lab6/src/symbol_mapper.m

```
sym_seq = symbol_mapper(bin_seq, M, d, name);
```

bin_seq: the binary sequence we want to symbol mapping

M: the size of every symbol

d: the minimum distance among the constellation points

name: the name of modulations

For example, if the input is a 30-bit binary sequence

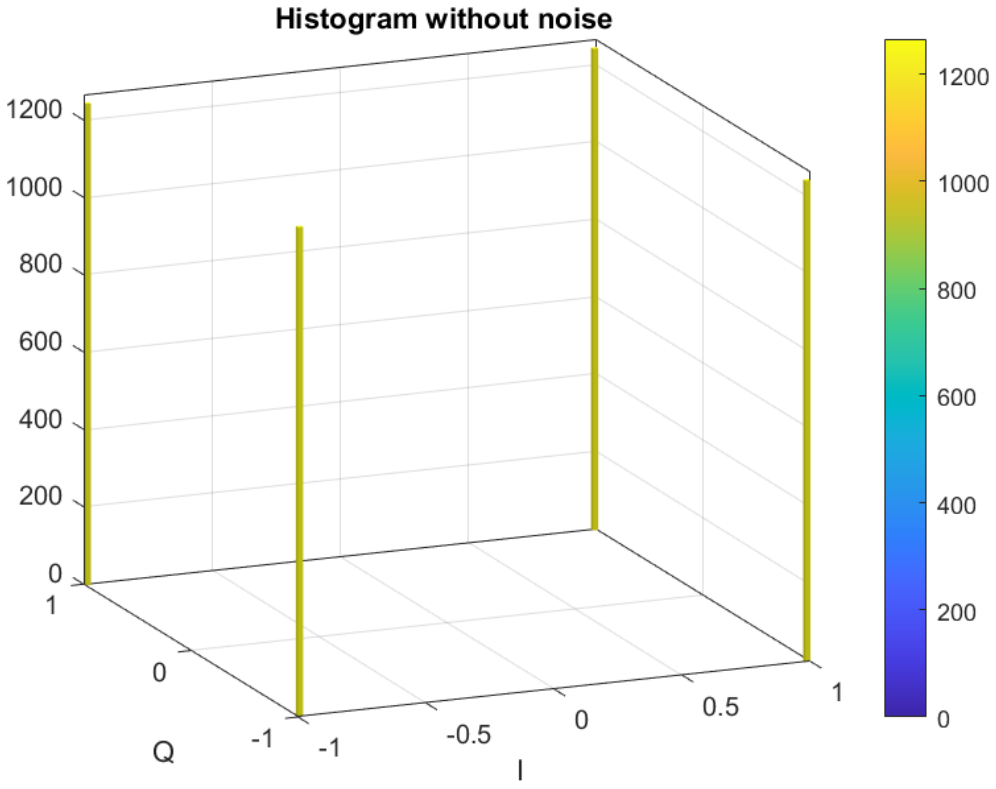
```
1   0   0   0   0   1   1   0   1   0   0   0   1   1   0   0   0   1   1   1   0   1   1   0   0   0   1   0   1   0
```

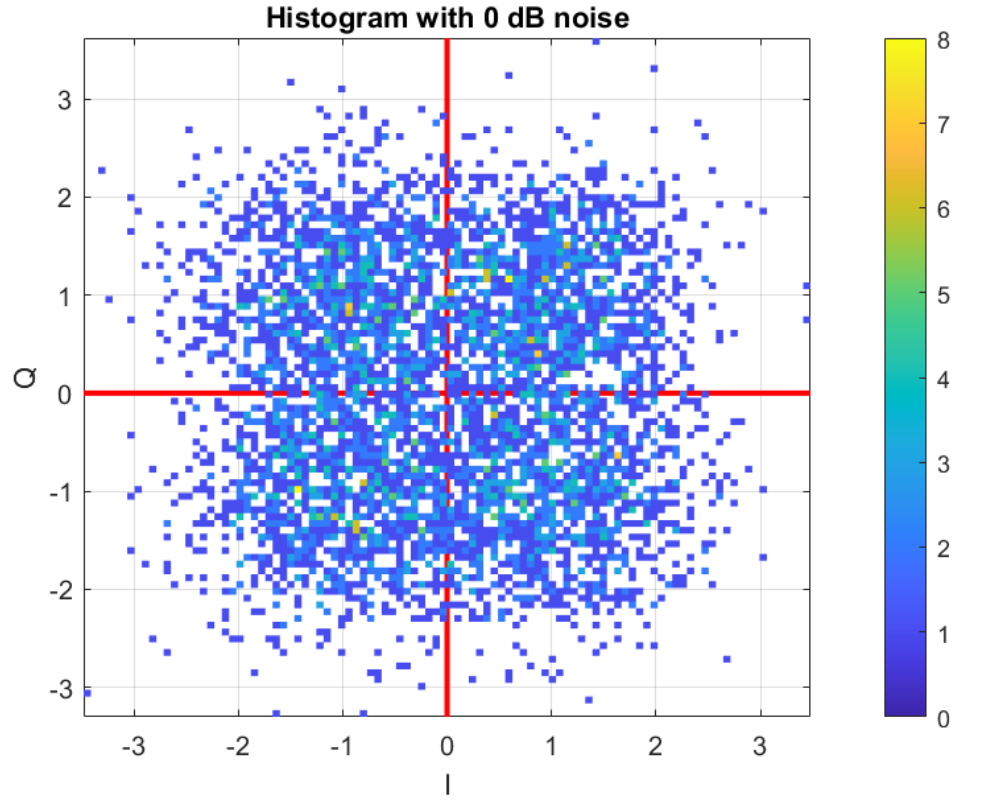
Then the output will be an array with 15 complex numbers

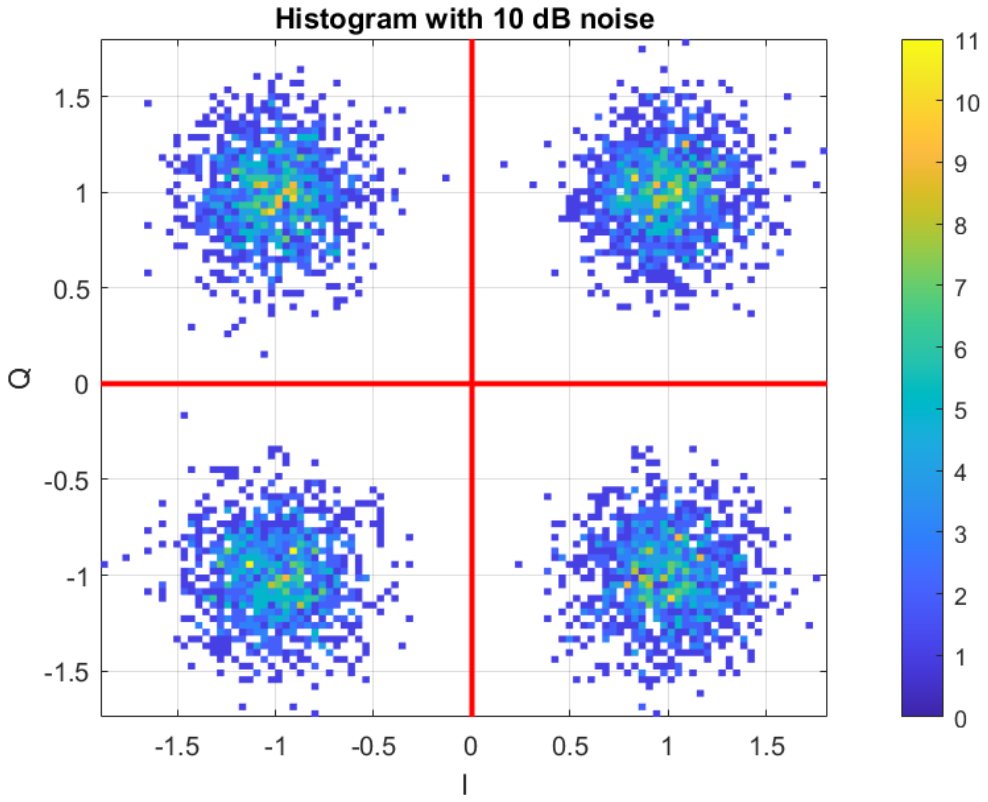
```
Columns 1 through 9

-0.0000 - 1.4142i   1.4142 + 0.0000i   0.0000 + 1.4142i  -0.0000 - 1.4142i  -0.0000 - 1.4142i   1.4142 + 0.0000i  -1.4142 + 0.0000i   1.4142 + 0.0000i   0.0000 + 1.4142i

Columns 10 through 15

-1.4142 + 0.0000i   0.0000 + 1.4142i  -0.0000 - 1.4142i   1.4142 + 0.0000i  -0.0000 - 1.4142i  -0.0000 - 1.4142i
```

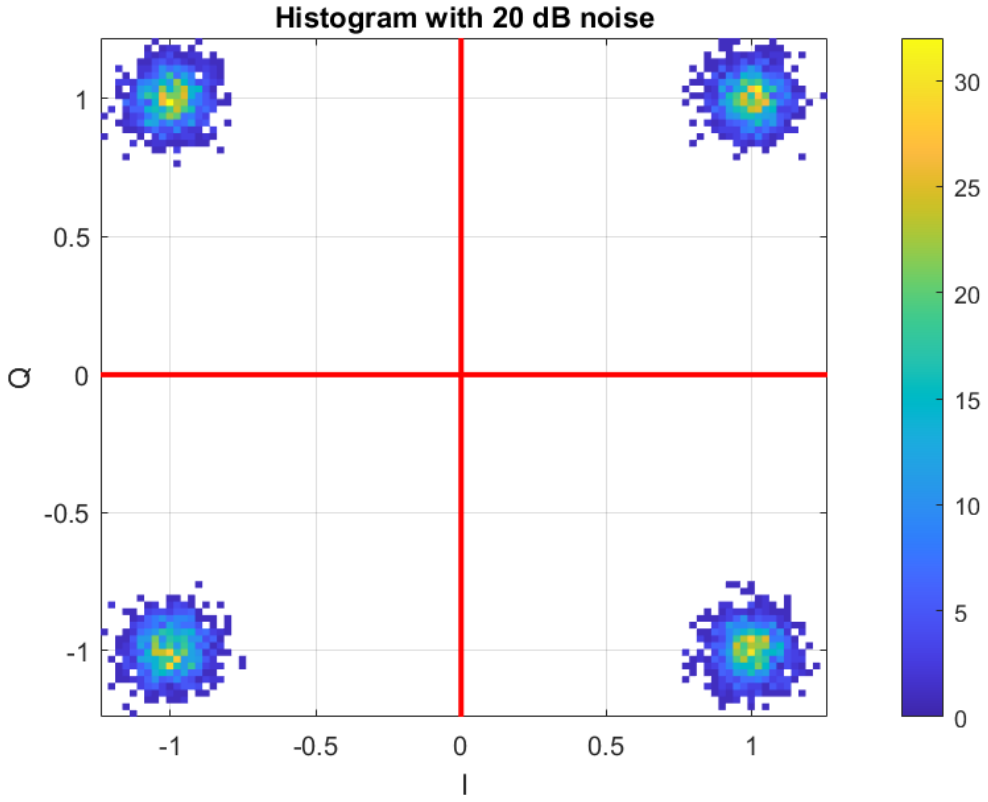
Q2

(a)


Histogram without noise
1200
1000
800
600
400
200
0
1
0
-1
Q
-1
-0.5
0
0.5
1
I
1200
1000
800
600
400
200
0


Histogram with 0 dB noise
3
2
1
0
-1
-2
-3
Q
-3
-2
-1
0
1
2
3
I
8
7
6
5
4
3
2
1
0


Histogram with 10 dB noise
1.5
1
0.5
0
-0.5
-1
-1.5
Q
-1.5
-1
-0.5
0
0.5
1
1.5
I
11
10
9
8
7
6
5
4
3
2
1
0


Histogram with 20 dB noise
1
0.5
0
-0.5
-1
Q
-1
-0.5
0
0.5
1
I
30
25
20
15
10
5
0

(b)

Please check the file
https://github.com/yuanchiachang/CommLab/blob/main/Lab6/src/symbol_demapper.m

In Q2(a)

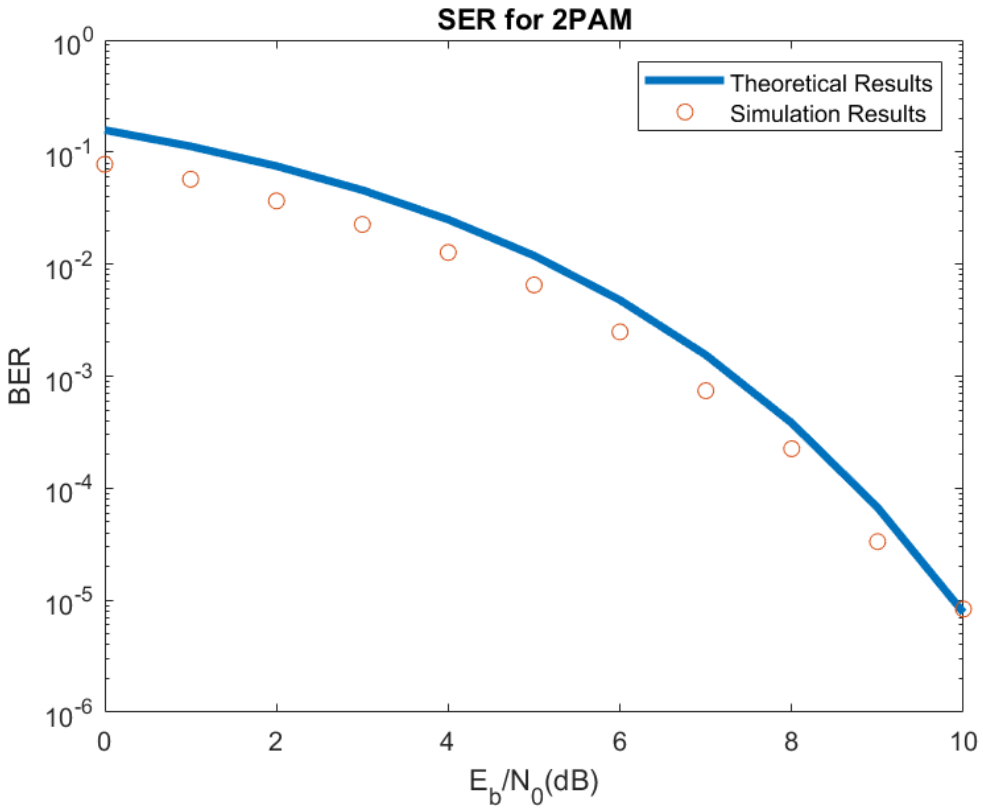
$\frac{E_b}{N_0} = 0dB$ SER:0.1446

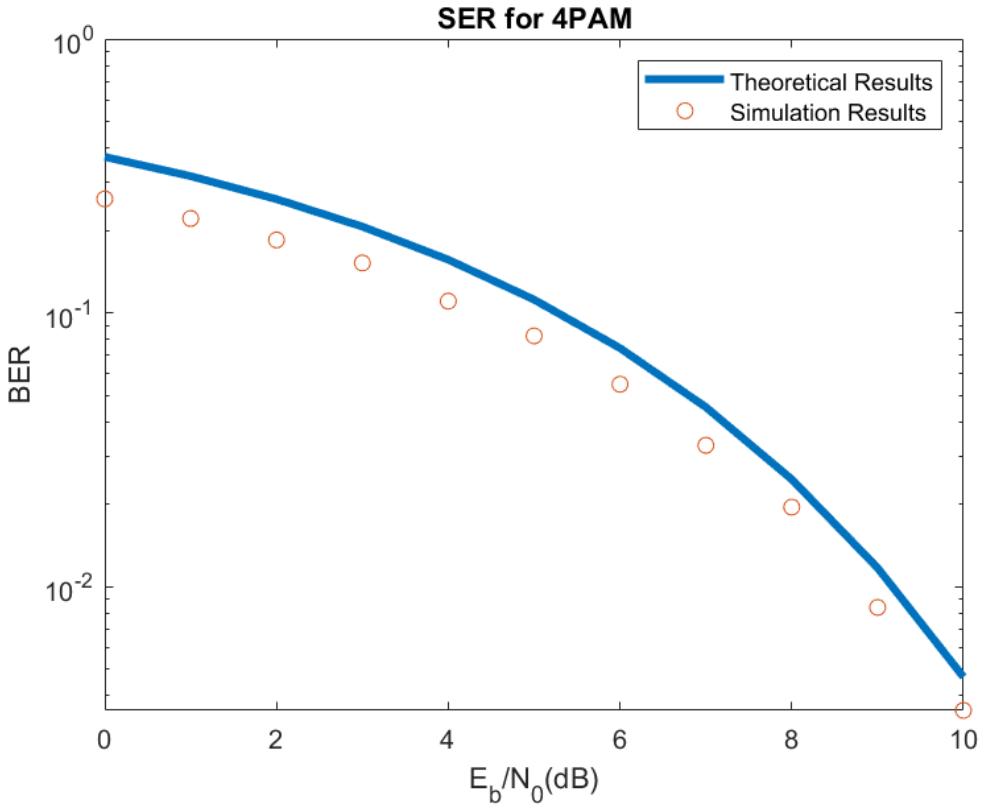
$\frac{E_b}{N_0} = 10dB$ SER:0(due to the lack of bits)

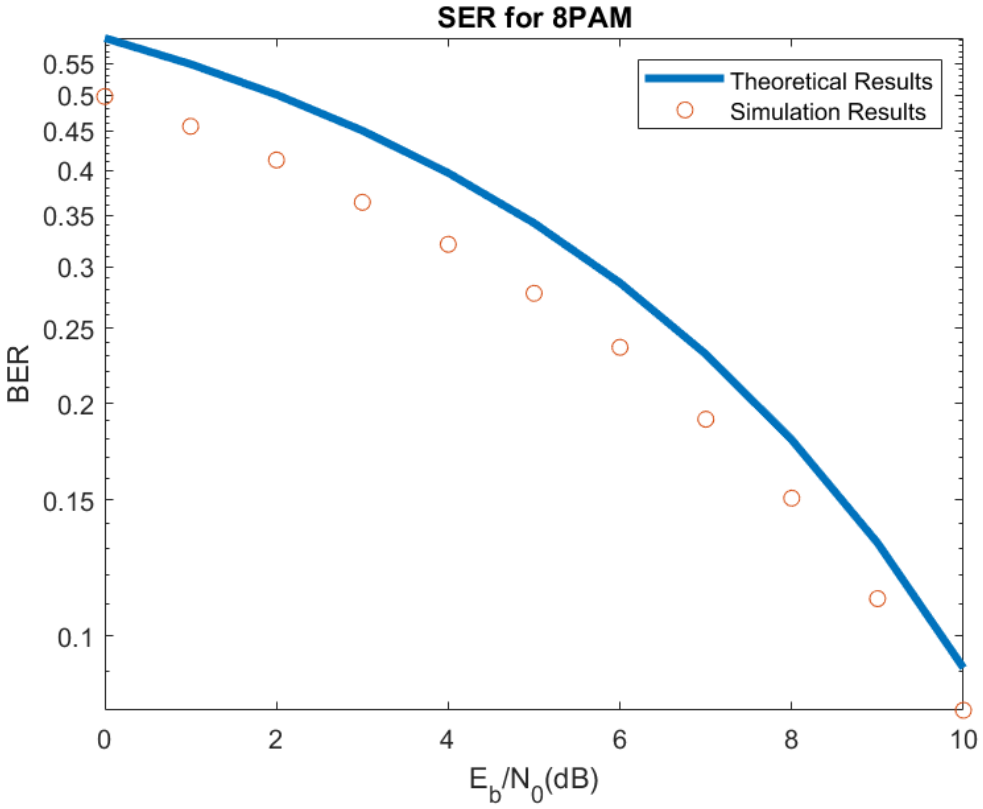
$\frac{E_b}{N_0} = 20dB$ SER:0(due to the lack of bits)

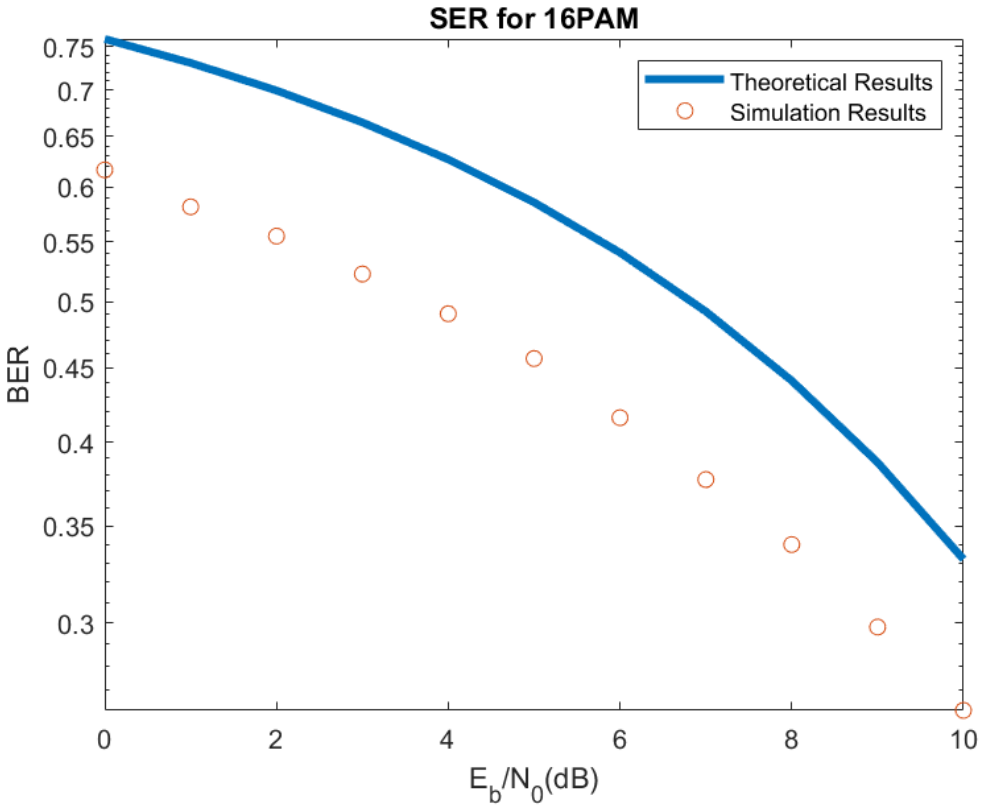
Q3

(a)

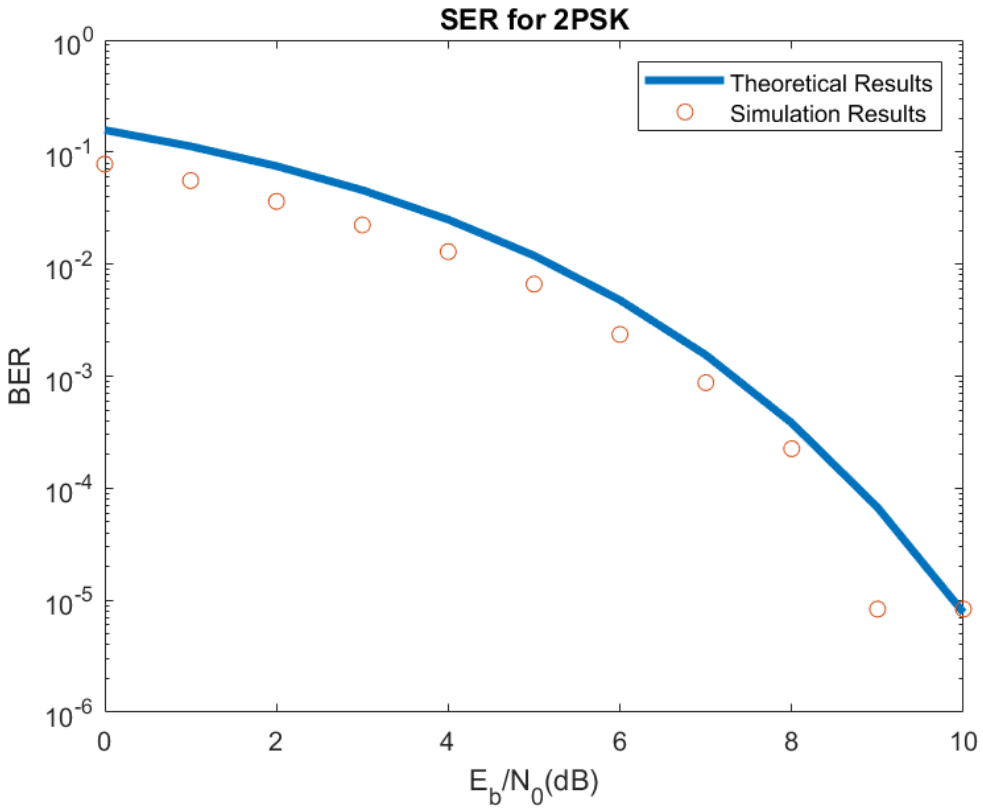


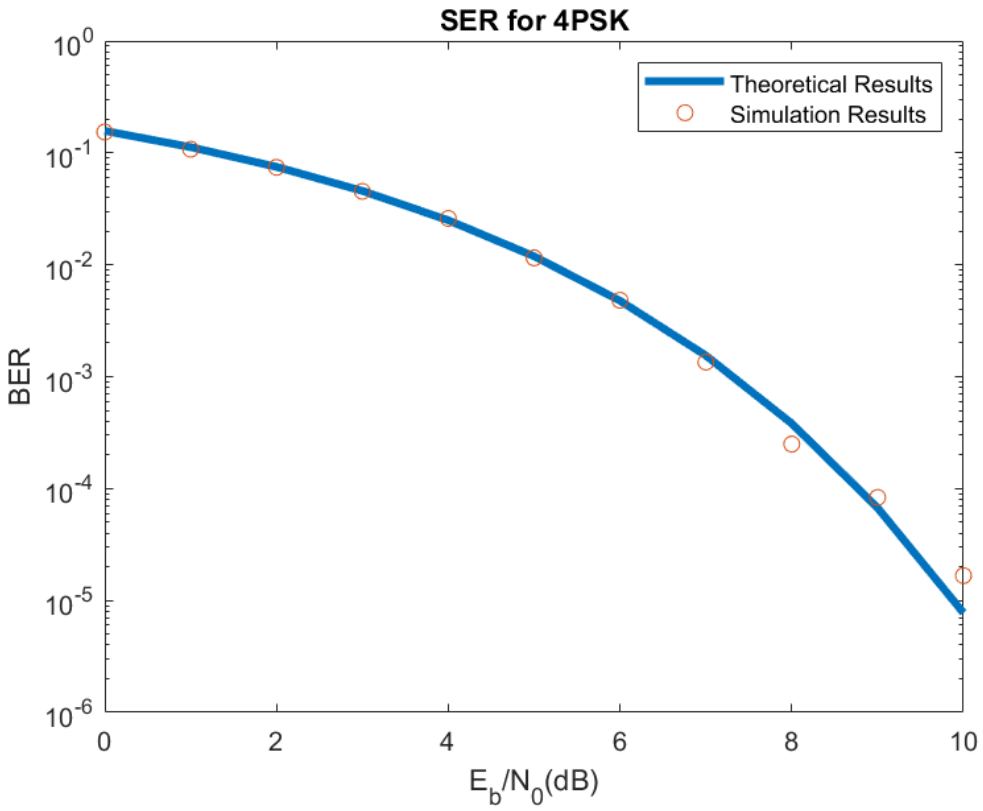

SER for 4PAM
Theoretical Results
Simulation Results
BER
10^0
10^{-1}
10^{-2}
0
2
4
6
8
10
E_b/N_0(dB)

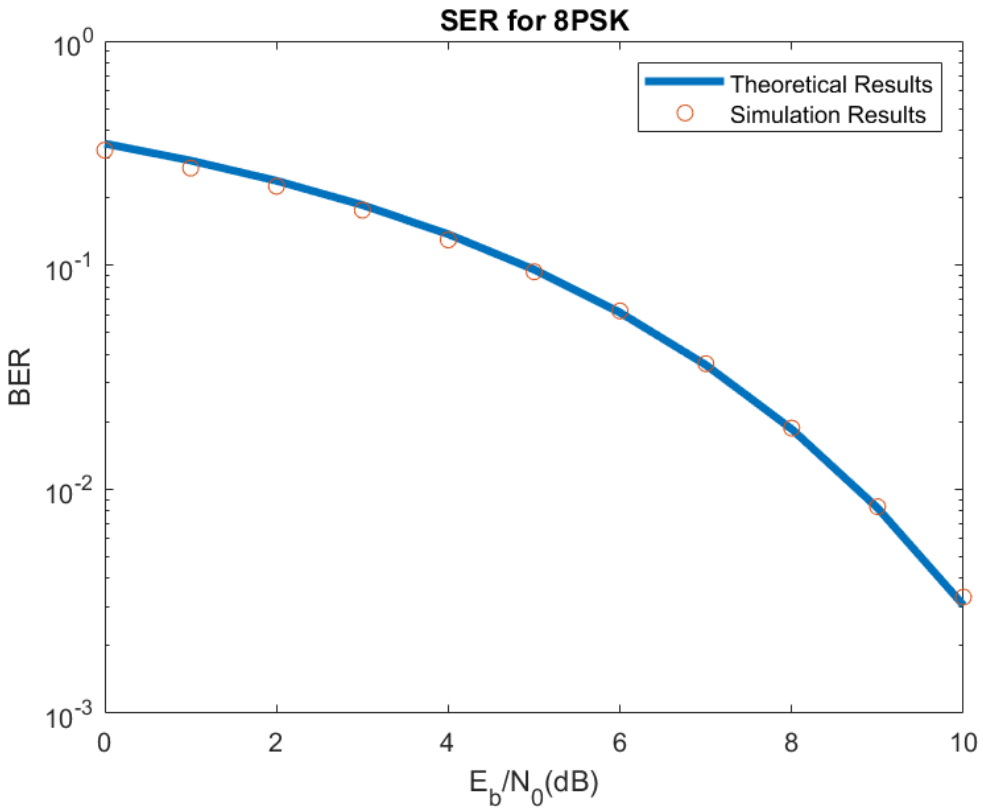

SER for 8PAM
Theoretical Results
Simulation Results
BER
0.55
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0
2
4
6
8
10
E_b/N_0(dB)

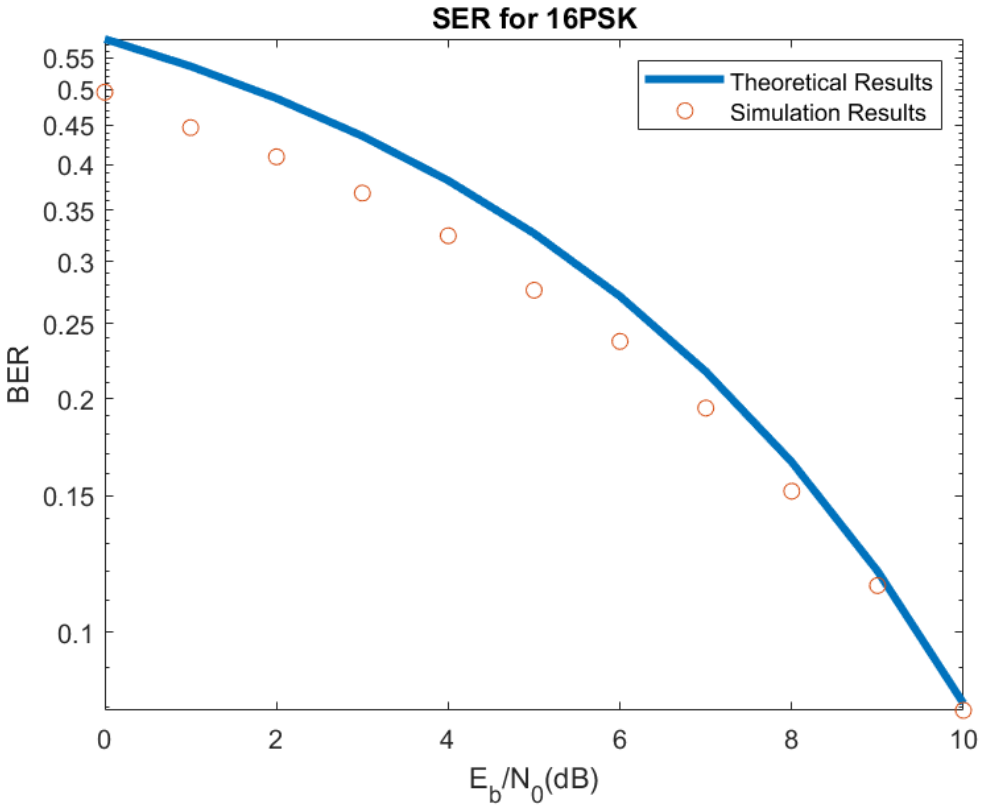

SER for 16PAM
Theoretical Results
Simulation Results
BER
0.75
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0
2
4
6
8
10
E_b/N_0(dB)

(b)

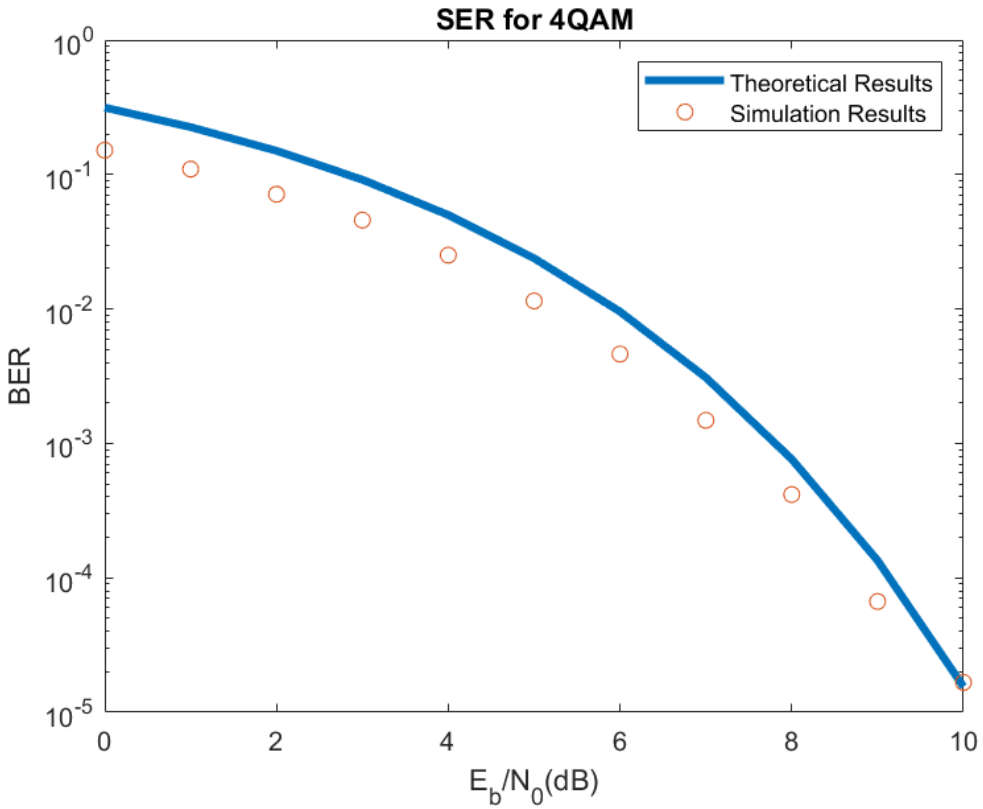

SER for 2PSK
Theoretical Results
Simulation Results
BER
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
0
2
4
6
8
10
E_b/N_0(dB)

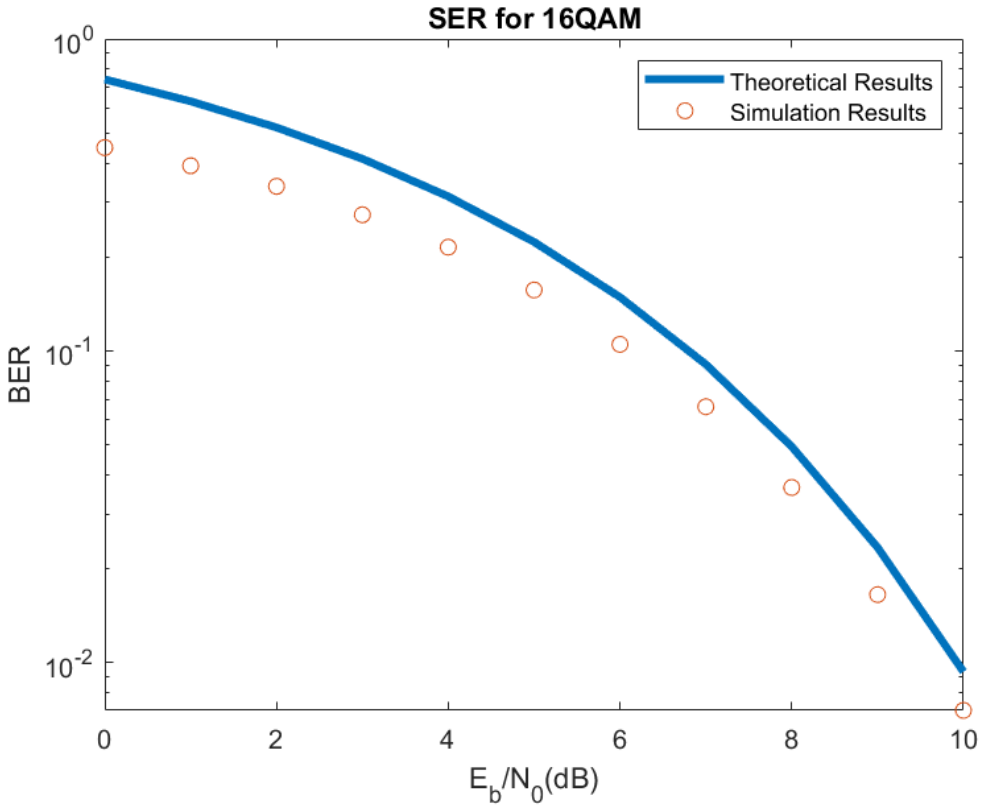

SER for 4PSK
Theoretical Results
Simulation Results
BER
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
0
2
4
6
8
10
E_b/N_0(dB)

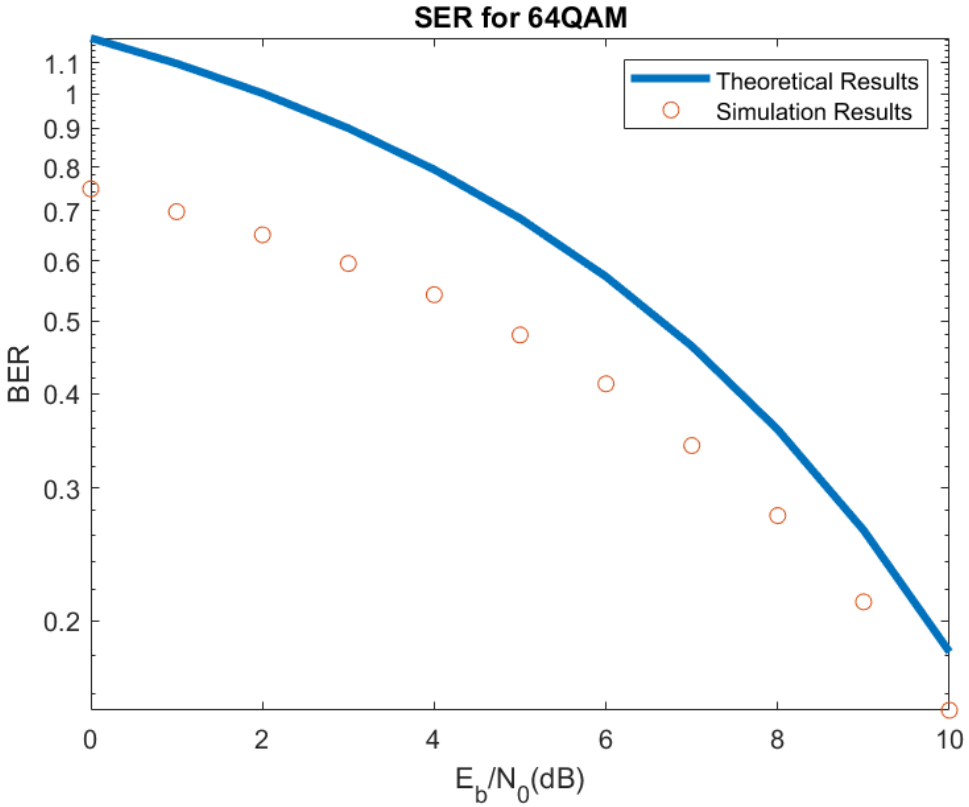

SER for 8PSK
Theoretical Results
Simulation Results
BER
10^{0}
10^{-1}
10^{-2}
10^{-3}
0
2
4
6
8
10
E_b/N_0(dB)


SER for 16PSK
Theoretical Results
Simulation Results
BER
0.55
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0
2
4
6
8
10
E_b/N_0(dB)

(c)


SER for 4QAM
Theoretical Results
Simulation Results
BER
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
0
2
4
6
8
10
E_b/N_0(dB)


SER for 16QAM
Theoretical Results
Simulation Results
BER
10^0
10^{-1}
10^{-2}
0
2
4
6
8
10
E_b/N_0(dB)


SER for 64QAM
Theoretical Results
Simulation Results
BER
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
2
4
6
8
10
E_b/N_0(dB)

(d)

1.Almost Every simulation result has lower error probability than the theoretical upper bound, which supports the correction of the code.

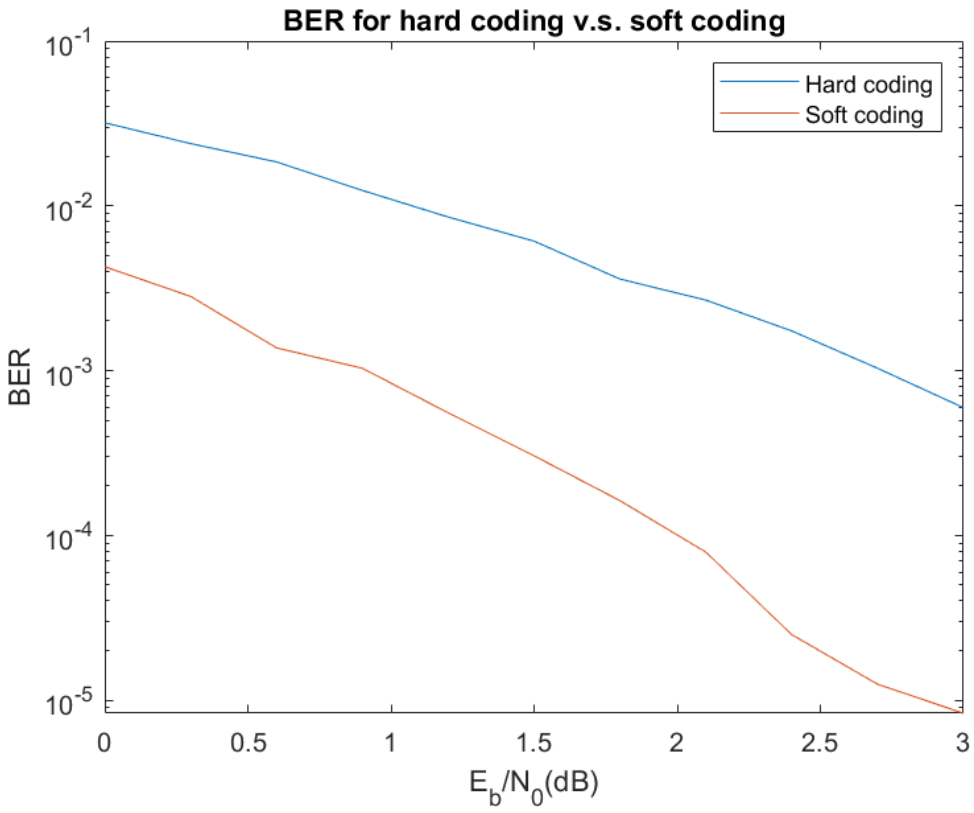
2. larger dB, smaller M will lead to smaller SER. Specifically, for 10dB, M=2, name= “PSK” or “PAM”, and for 10dB, M=4, name=”QAM”, the SER is approximately 10^{-5}

3. Larger M will lead to fewer running time. The reason is that for the same length of input sequence, larger M make the length of symbol sequence smaller.

4. the simulation result of 4-PSK and 8-PSK is closest to the theoretical upper bound. For 4-PSK 9dB and 10dB, the simulation result is even higher than the theoretical upper bound due to the small SER and the lack of the error bit, which leads to the statistical error.

Q4

(a)(b)



This figure shows that the performance of soft coding is better than hard coding. BER of soft coding is approximately 1% to 10% of BER of hard coding.

Appendix

Code
https://github.com/yuanchiachang/CommLab/blob/main/Lab6/src
Reference
https://openhome.cc/Gossip/AlgorithmGossip/GrayCode.htm#Java