Electrical Engineering Department,

Fourth Year - Communications & Electronics.

EE484 COMMUNICATION SYSTEMS

Experiment 1: Bluetooth Protocol

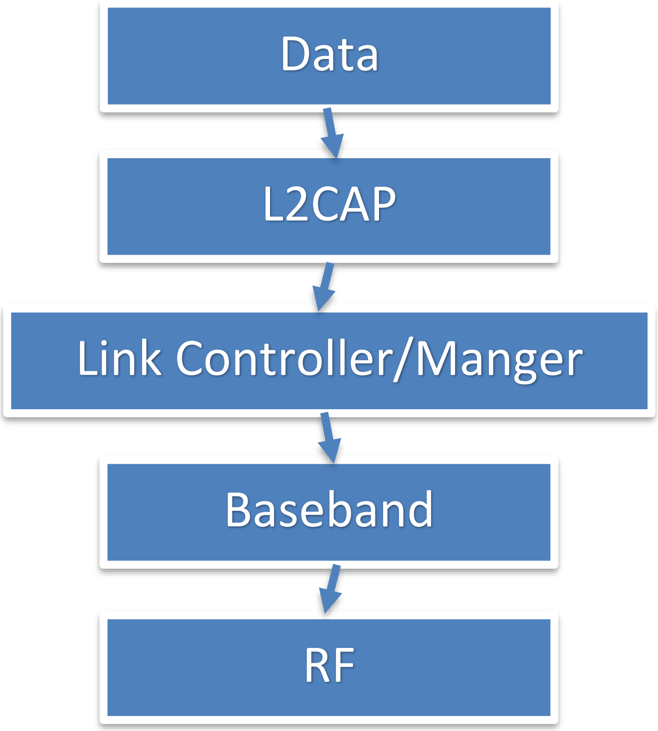
PREPARED BY	SECTION	SEAT.NO.
Mahmoud Mohamed Kamal Ismail	7	250
Mahmoud Alaa-Elden Mahmoud	7	249
Mahmoud AbdElHady Mahmoud	7	248
Mahmoud Adel Hassan Mohamed	7	247
Mostafa Khaled Gaber Rahal	7	262

➢ Contents

https://github.com/MahmoudFierro98/EE484_Communication_Systems_Lab

1. Problem 1: Bluetooth Standard

1.1. Bluetooth protocol stack



- L2CAP (Logical Link Control and Adaptation Protocol):
  - Multiplexing of higher layer protocols, which allows them to use the links provided by the lower layers.
  - Segmentation and reassembly of data packets of the upper layer that are larger than the capacity of the radio layer underneath.
- Link Controller/Manger (LMP):
  - LMP establishes logical links between Bluetooth devices and maintains the links for enabling communications. The other main functions of LMP are device authentication, message encryption, and negotiation of packet sizes.
  - Master-Slave role switch.
- Baseband:
  - Framing of data packets (ACL – SCO - eSCO).
  - Timing.
- Radio (RF):
  - Send-Receive modulated signal.

1.2. Bluetooth evolution

Bluetooth 1.0 – 1.2

After the Bluetooth Special Interest Group (also known as SIG) was formed in 1998, the Bluetooth 1.0 specification was released a year later in 1999.

The first phone to use Bluetooth was the Ericsson T36, which was unveiled in 2000. However, it wasn't actually available for purchase until 2001 when Ericsson released the revised T39 mobile phone. The Ericsson T39 used Bluetooth 1.0b, which was a mild successor to Bluetooth 1.0a. Compared to the Bluetooth we know today, Bluetooth 1.0b was *very* slow and unreliable. It had a maximum data transfer speed of 732.2kbs and could only stay connected to devices within a range of 33 feet. It can be easy to complain about Bluetooth being a pain to use today, but back then, the experience was substantially worse.

Also, with such slow data transfer speeds, Bluetooth wasn't really designed for listening to music the way it is now. Instead, its main purpose was for wireless Bluetooth headsets for making/receiving phone calls.

In February 2001, Bluetooth 1.1 hit the market. It set out to fix a lot of the usability complaints with Bluetooth 1.0, and while it still had the same max data transfer speed, it introduced support for up to seven simultaneous connections and made it possible to have Bluetooth connections on non-encrypted channels — one of the biggest drawbacks to Bluetooth 1.0.

Fast forward a couple more years, and Bluetooth 1.2 was ready to shine in November 2003. It was the first Bluetooth update that increased the data transfer speed, kicking things up to 1Mbs. Bluetooth 1.2 also benefited from making devices faster and easier to discover, was backward compatible with Bluetooth 1.1, and introduced something called "Adaptive Frequency Hopping" that made Bluetooth more resistant to interference from radio frequencies.

Another big win for Bluetooth 1.2 was its use of Extended Synchronous Connections. The main purpose of this was to improve the quality of phone calls over Bluetooth, with it giving users an option to increase latency in favor of better data transfer for high-quality audio.

Bluetooth 2.0 and 2.1

Bluetooth 2.0 was the first main number update for Bluetooth, and it was well-deserving of the big name change.

This is when EDR — Enhanced Data Rate — came to Bluetooth, enabling data transfer speeds that were three times as fast. EDR boasted a bit rate of 3/Mbs, but in real-world use, Bluetooth 2.0 was only capable of 2.1/Mbs. Even so, it was a huge step forward for the standard.

Along with better wireless performance, thanks to the increased speed, Bluetooth 2.0 and its use of EDR also allowed for better battery life on devices that used it compared to Bluetooth 1.2. In 2005, former Executive Director of the Bluetooth SIG, Mike Foley, noted that a wireless headset with Bluetooth 1.2 may only last 90 minutes on a charge, whereas one with Bluetooth 2.0 could be used for over four hours on a single charge.

Bluetooth 2.1 was a modest follow-up to 2.0, and it was unveiled by the SIG in July 2007.

The main draw to Bluetooth 2.1 was that it offered a simpler pairing process between devices, utilizing its "secure simple pairing" system. Similarly, Bluetooth 2.1 introduced the option to pair devices using NFC (the same technology used by Google Pay and Apple Pay for contactless payments).

Bluetooth 3.0

There was nearly a two-year gap between Bluetooth 2.1 and 3.0, with the latter being adopted by the SIG in April 2009. Where EDR was the main draw to the 2.0 and 2.1 days, Bluetooth 3.0's claim to fame was its HS — high-speed channel.

In theory, Bluetooth 3.0 could achieve data transfer speeds up to a whopping 24/Mbs. However, those speeds didn't happen solely over Bluetooth. Instead, Bluetooth 3.0 established a link connection to the 802.11 protocol (AKA, Wi-Fi).

With such vastly-improved speeds, Bluetooth was finally ready to go beyond audio transfers. Now, it was fast enough to stream video wirelessly between devices.

This is one of the biggest speed bumps Bluetooth has ever seen, and it was also the last iteration of the wireless standard before we were introduced to the world of Bluetooth Low Energy.

Bluetooth 4.0 – 4.2

That brings us nicely to June 2010. This is when the SIG formally adopted Bluetooth 4.0, and in October 2011, the iPhone 4S debuted as the first phone to use the new wireless protocol.

With Bluetooth 4.0, the big focus here was on its new Bluetooth Low Energy technology — also referred to as BLE. BLE was designed to offer more efficient connections to smaller wireless devices — often seen in fitness trackers and other smart wearables that don't require a lot of power.

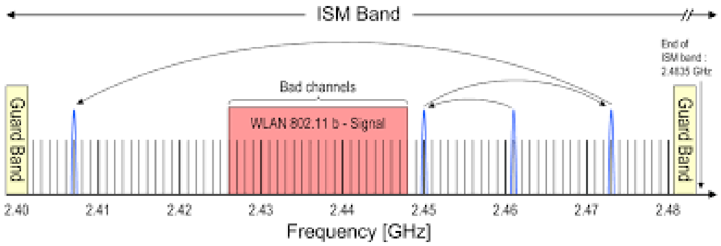
When BLE was first introduced, there was a waiting period for consumers to see its full usefulness as we waited for devices to adopt the standard. Today, however, it's become an expected feature in any gadget that can benefit from it.

Bluetooth 4.0 retained the same 24/Mbs speed when used with non-BLE devices, along with adding 128-bit encryption for enhanced security.

2. Problem 2: Questions Related to the Specifications

1- What is the frequency range of the Bluetooth? Is Bluetooth the only communication system that uses this range?

- Bluetooth uses channels in the 2.4GHz (About 79 Channels), ISM band with bandwidth of 1MHz.
- No, there's Bad channels (for WLAN).



2- What type of modulation does the basic rate of Bluetooth use? For this modulation scheme, mention the bandwidth-bit period product and the modulation index.

- For basic rate (BR) transmission, Bluetooth uses binary Gaussian frequency shift keying (GFSK) modulation scheme with a bandwidth bit period product BT=0.51 and 0.32 as nominal modulation index

3- How does the enhanced data rate mode achieve about three times the basic rate mode?

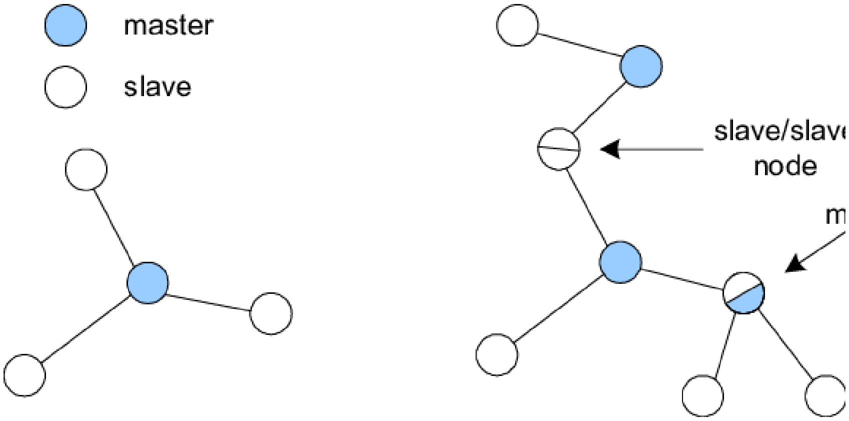
- As in Bluetooth 2.0 using modulation (DQPSK,8SPSK), Increase modulation order.

4- How many channels are used in Bluetooth and what is the Bandwidth of each one?

- Bluetooth uses channels in the 2.4GHz (About 79 Channels), ISM band with bandwidth of 1MHz.

5- What is the slot duration of the master-slave configuration? Sketch a multi-slot packet transmission for this configuration.

- Channel is divided into time slots of 625µsec, and all devices use the same channel but a signed time slots at different time.
- For large amount of data: 3TS – 5TS.
- For small amount of data: 1TS.



Network configuration of Bluetooth (a) piconet and (b) scatternet.

6- What are the Bluetooth power classes?

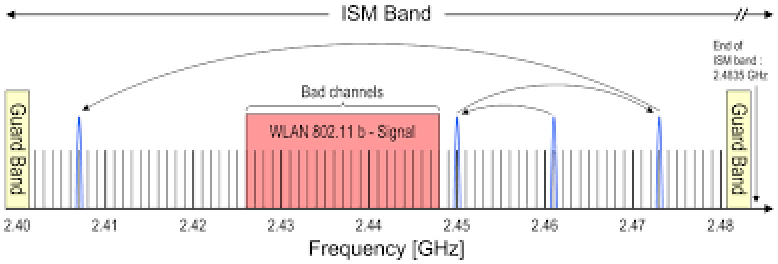
Class	Max. permitted power (mW)	Range (m)	Uses
1	100	~100	USB Bluetooth sticks for PCs.
1.5	10	~20	
2	2.5	~10	
3	1	~1 (10 in lab1 PDF)	Battery Drives (Mobile phone).
4	0.5	~0.5	

7- What is the difference between master and slave devices, and what is the max number of slave devices within the piconet?

- Master device is the device that initiate connection and it makes channel assessment to measure interference on each channel to select channels to be used in the hopping sequence.
- A single master device can be connected to up to seven different slave devices. Any slave device in the piconet can only be connected to a single master.
- The max number of slave devices: 7.

8- Describe the frequency hopping spread spectrum concept.

- Bluetooth share 2.4GHz ISM frequency band with other wireless technologies, so it uses frequency hopping (FHSS) multiple access to minimize interference.
- There are 79 channels available for Bluetooth, Carrier frequency of transmission (Channel) changes after each packet is transmitted.
- Each piconet has each own frequency hopping sequence, so they don't interfere with each other.



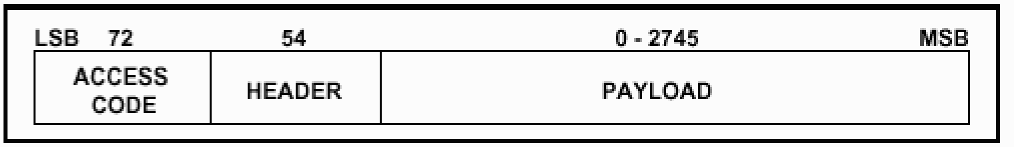
- If packet length is 1TS then hopping frequency = ${}^{1}/_{625\mu} = 1600\ Hop/sec$
- If packet length is 3TS then hopping frequency = ${}^{1}/_{(3 \times 625\mu)} = 533.33\ Hop/sec$
- If packet length is 5TS then hopping frequency = ${}^{1}/_{(5 \times 625\mu)} = 320\ Hop/sec$

9- In the baseband layer, discus the Composition of an ACL packet? the function of each group of bits in the packet? and how the transmission errors can be detected and corrected in it?

- Asynchronous Connection-Less (ACL) is a communication protocol, it's used as a transmission link used for data communication in the Bluetooth system or as a definition with access code (72 bit) + packet header (54 bit) + payload (0-2745 bits)
  - Access Code: Access code are used for timing synchronization, offset compensation, paging and inquiry. There are three different types of Access code: Channel Access Code (CAC), Device Access Code (DAC) and Inquiry Access Code (IAC). The channel access code identifies a unique piconet

while the DAC is used for paging and its responses. IAC is used for inquiry purpose.

  - Header: The header contains information for packet acknowledgement, packet numbering for out-of-order packet reordering, flow control, slave address and error check for header.
  - Payload: The packet payload can contain either voice field, data field or both. It has a data field, the payload will also contain a payload header.



LSB 72	54	0 - 2745 MSB
ACCESS CODE	HEADER	PAYLOAD

- There are three kinds of error correction schemes used in the baseband protocol: 1/3 rate FEC, 2/3 rate FEC and ARQ scheme.
  - In 1/3 rate FEC every bit is repeated three times for redundancy.
  - In 2/3 rate FEC a generator polynomial is used to encode 10-bit code to a 15-bit code.
  - In the ARQ scheme, DM, DH and the data field of DV packets are retransmitted till an acknowledgement is received (or timeout is exceeded). Bluetooth uses fast, unnumbered acknowledgement in which it uses positive and negative acknowledgements by setting appropriate ARQN values. If the timeout value is exceeded, Bluetooth flushes the packet and proceeds with the next.

10- What is Bluetooth low energy (BLE)? Describe the main features of it.

- Bluetooth Low Energy hit the market in 2011 as Bluetooth 4.0. When talking about Bluetooth Low Energy vs. Bluetooth, the key difference is in Bluetooth 4.0's low power consumption. Although that may sound like something negative, it is actually extremely positive when talking about M2M communication. With Bluetooth LE's power consumption, applications can run on a small battery for four to five years. Although this is not ideal for talking on the phone, it is vital for applications that only need to exchange small amounts of data periodically.
- Just like Bluetooth, BLE operates in the 2.4 GHz ISM band. Unlike classic Bluetooth, however, BLE remains in sleep mode constantly except for when a connection is initiated. The actual connection times are only a few mS, unlike Bluetooth which would take ~100mS. The reason the connections are so short, is that the data rates are so high at 1 Mb/s.
- BLE’s M2M/IoT Applications:

  Blood pressure monitors
  Fibit-like devices
  Industrial monitoring sensors
  Geography-based, targeted promotions (iBeacon)
  Public transportation apps
- Features: Easy to use – Use in short distance (connection range) – Low power consumption – Security - Compatible.

3. Problem 2: Questions Related to the Experiment

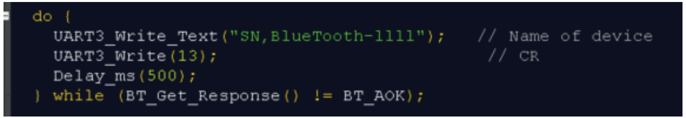
1- What are the specification of RN41/RN41N Bluetooth module (data rate, coverage distance, power class, …….).

- Fully qualified Bluetooth® version 2.1 module, supports version 2.1 + Enhanced Data Rate (EDR).
- Baud rate speeds: 1,200 bps up to 921 Kbps, non-standard baud rates can be programmed.
- Class 1 radio, 330’ (100 m) range, 15 dBm output transmitter, -80 dBm typical receive sensitivity.
- Frequency 2,402 ~ 2,480 MHz.

2- In the following questions, identify the line of code that needs to be changed and write the new commands.

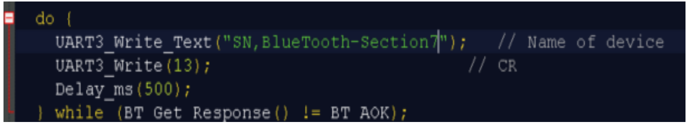
a. How to change the Bluetooth module name in the given code.

- From BT_Routines.c



```
do {
  UART3_Write_Text("SN,BlueTooth-1111");   // Name of device
  UART3_Write(13);                          // CR
  Delay_ms(500);
} while (BT_Get_Response() != BT_AOK);
```

-

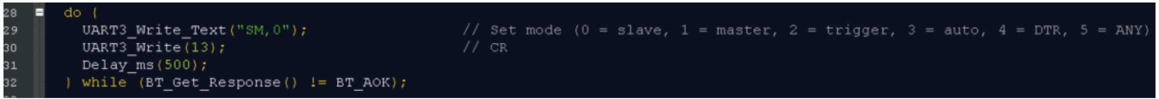


```
do {
  UART3_Write_Text("SN,BlueTooth-Section7");   // Name of device
  UART3_Write(13);                             // CR
  Delay_ms(500);
} while (BT Get Response() != BT AOK);
```

-
- Changed to: BlueTooth-Section7

b. How to change the Bluetooth module from Slave to Master?

- From BT_Routines.c



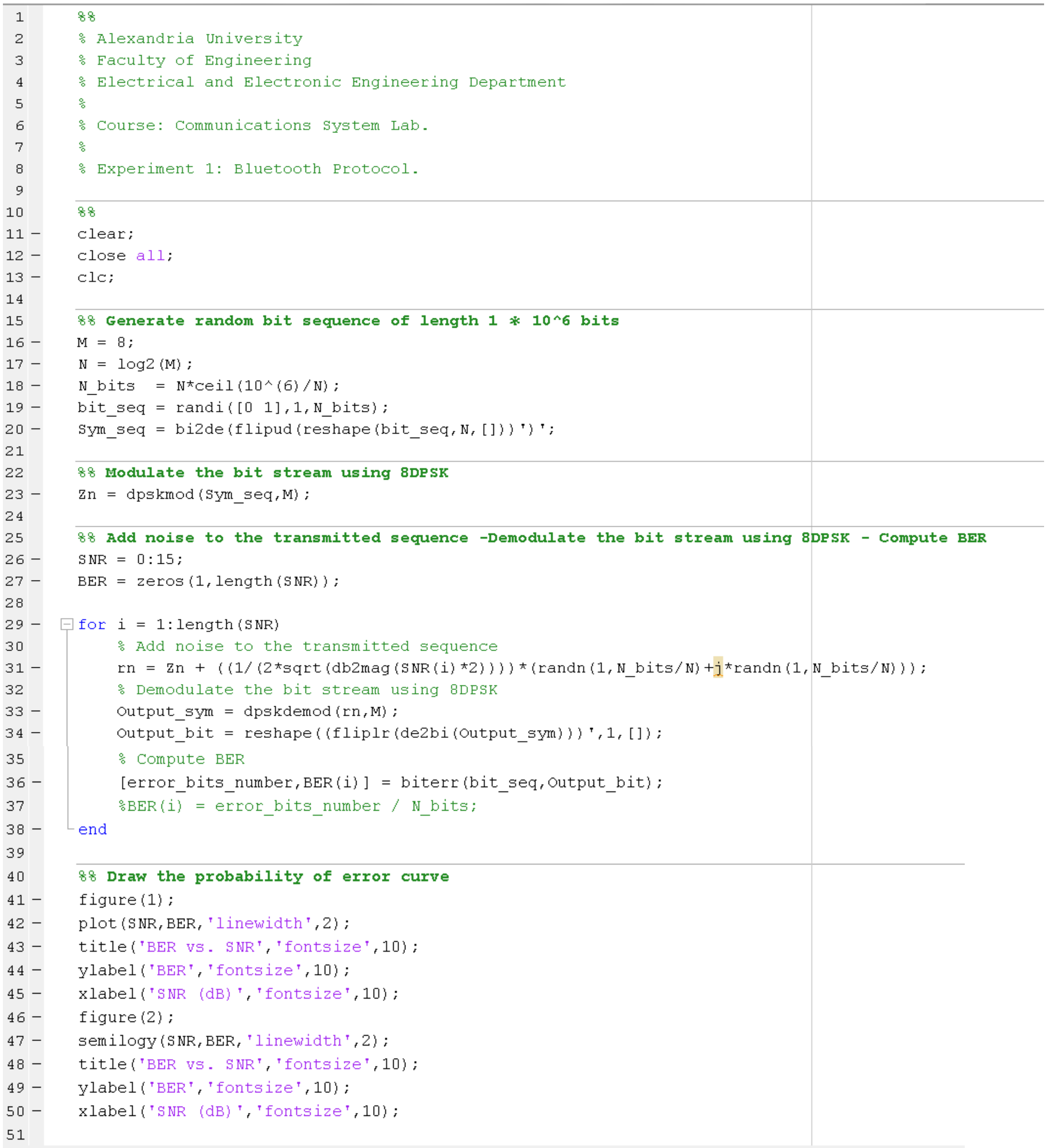
```
do {
  UART3_Write_Text("SM,0");                 // Set mode (0 = slave, 1 = master, 2 = trigger, 3 = auto, 4 = DTR, 5 = ANY)
  UART3_Write(13);                          // CR
  Delay_ms(500);
} while (BT_Get_Response() != BT_AOK);
```

-

0	Slave
1	Master

4. Problem 4: Mini-Simulations

- Code:



```
%%
% Alexandria University
% Faculty of Engineering
% Electrical and Electronic Engineering Department
%
% Course: Communications System Lab.
%
% Experiment 1: Bluetooth Protocol.

%%
clear;
close all;
clc;

%% Generate random bit sequence of length 1 * 10^6 bits
M = 8;
N = log2(M);
N_bits  = N*ceil(10^(6)/N);
bit_seq = randi([0 1],1,N_bits);
Sym_seq = bi2de(flipud(reshape(bit_seq,N,[]))')';

%% Modulate the bit stream using 8DPSK
Zn = dpskmod(Sym_seq,M);

%% Add noise to the transmitted sequence -Demodulate the bit stream using 8DPSK - Compute BER
SNR = 0:15;
BER = zeros(1,length(SNR));

for i = 1:length(SNR)
    % Add noise to the transmitted sequence
    rn = Zn + ((1/(2*sqrt(db2mag(SNR(i)*2))))*(randn(1,N_bits/N)+j*randn(1,N_bits/N)));
    % Demodulate the bit stream using 8DPSK
    Output_sym = dpskdemod(rn,M);
    Output_bit = reshape((fliplr(de2bi(Output_sym)))',1,[]);
    % Compute BER
    [error_bits_number,BER(i)] = biterr(bit_seq,Output_bit);
    %BER(i) = error_bits_number / N_bits;
end

%% Draw the probability of error curve
figure(1);
plot(SNR,BER,'linewidth',2);
title('BER vs. SNR','fontsize',10);
ylabel('BER','fontsize',10);
xlabel('SNR (dB)','fontsize',10);
figure(2);
semilogy(SNR,BER,'linewidth',2);
title('BER vs. SNR','fontsize',10);
ylabel('BER','fontsize',10);
xlabel('SNR (dB)','fontsize',10);
```

- **Graphs:**

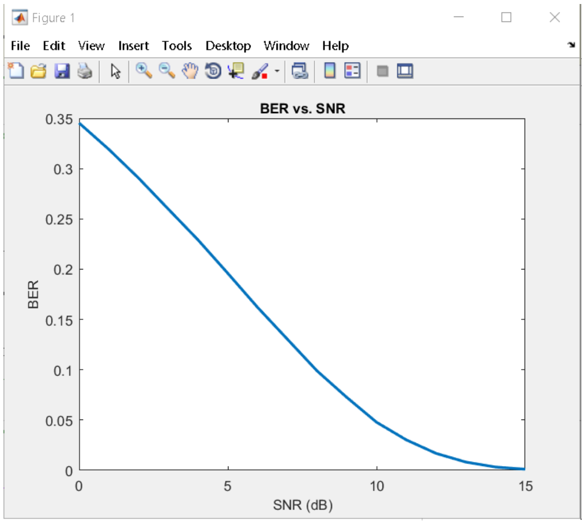


Figure 1
File Edit View Insert Tools Desktop Window Help
BER vs. SNR
BER
SNR (dB)
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
5
10
15

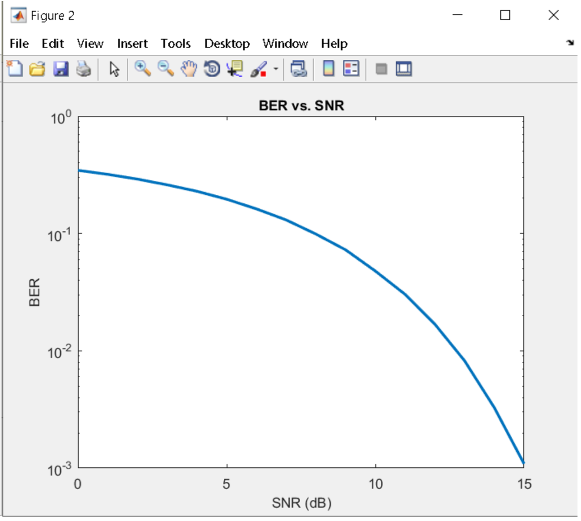


Figure 2
File Edit View Insert Tools Desktop Window Help
BER vs. SNR
BER
SNR (dB)
10^{0}
10^{-1}
10^{-2}
10^{-3}
0
5
10
15