Practical- 5

To study CDMA spreading/despreading techniques and apply it on the Communication link in MATLAB

MULTIPLE ACCESS TECHNIQUES

- Multiple access techniques are used to allow a large number of mobile users to share the allocated spectrum in the most efficient manner.
  - As the spectrum is limited, so the sharing is required to increase the capacity of cell or over a geographical area by allowing the available bandwidth to be used at the same time by different users.
  - And this must be done in a way such that the quality of service doesn't degrade within the existing users.
  - A cellular system divides any given area into cells where a mobile unit in each cell communicates with a base station.
  - The main aim in the cellular system design is to be able to increase the capacity of the channel i.e. to handle as many calls as possible in a given bandwidth with a sufficient level of quality of service.

These includes mainly the following:

1. Frequency division multiple-access (FDMA)
2. Time division multiple-access (TDMA)
3. Code division multiple-access (CDMA)

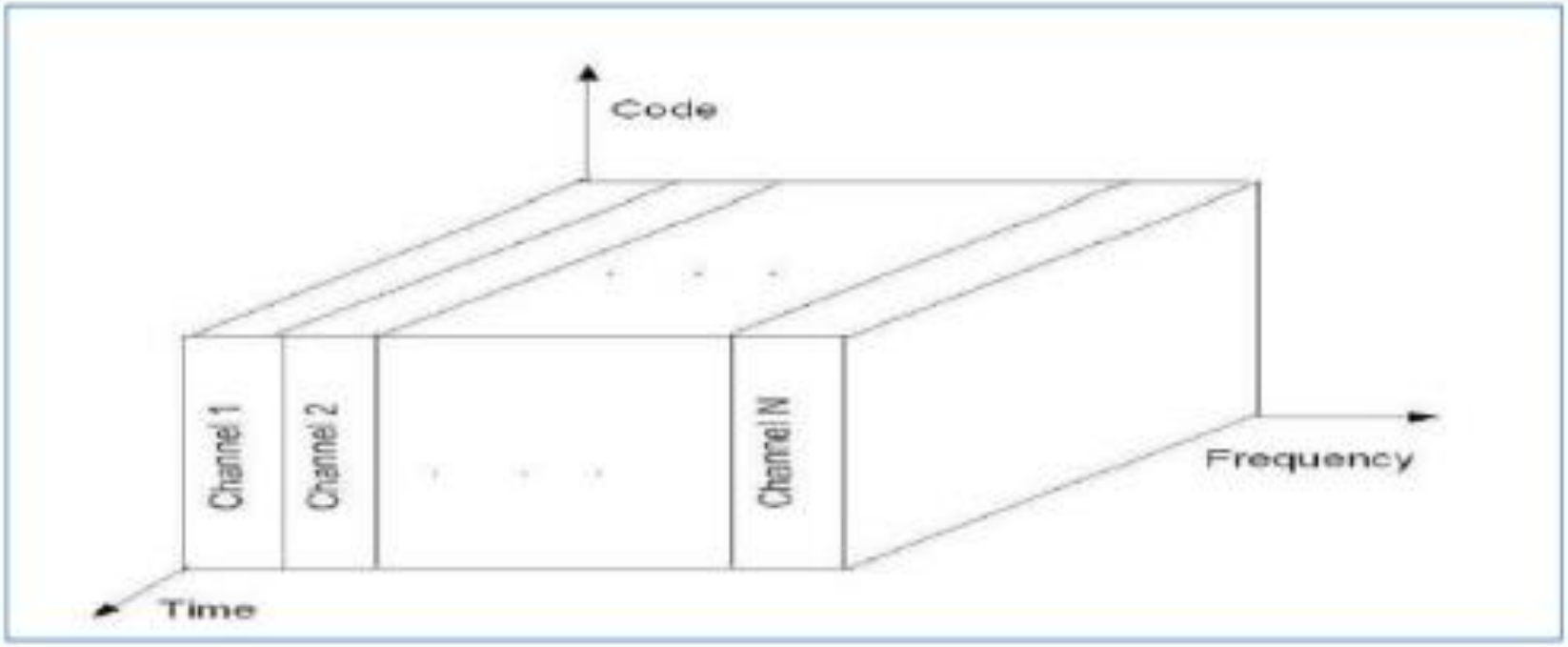


Figure : The basic concept of FDMA.

- Each individual user is assigned a pair of frequencies while making or receiving a call as shown in Figure.
- One frequency is used for downlink and one pair for uplink. This is called frequency division duplexing (FDD).

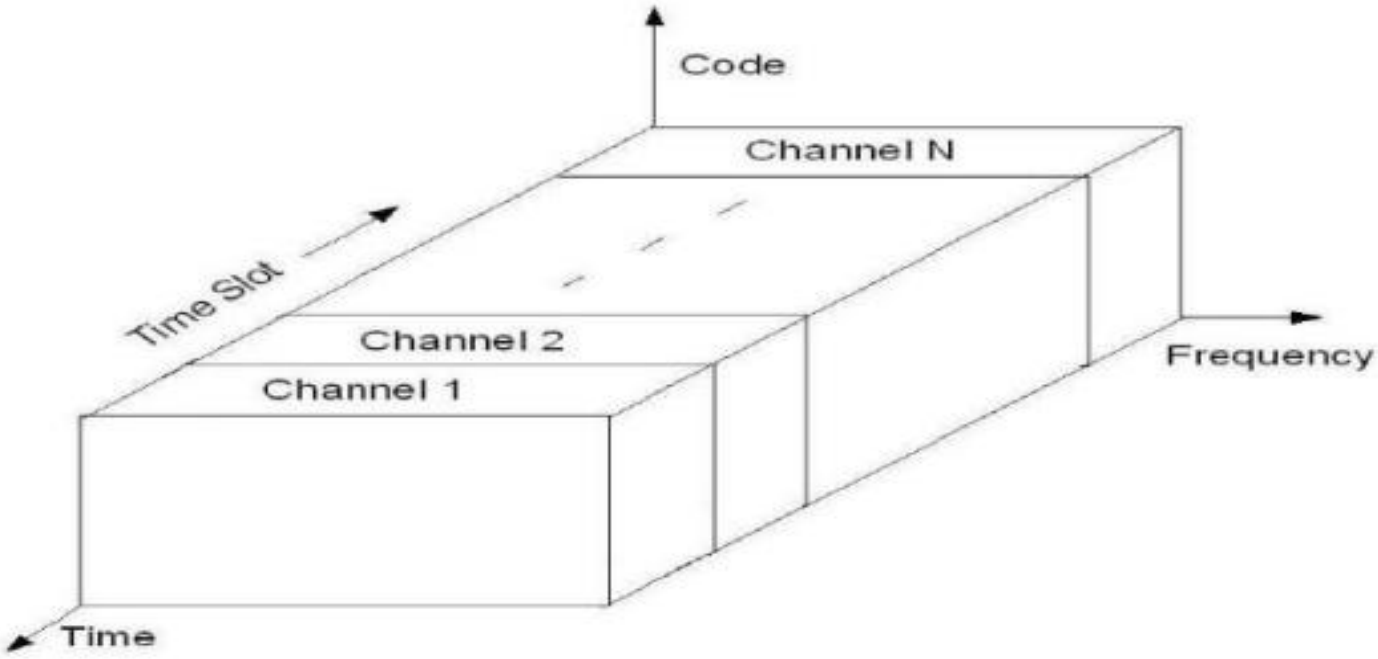


Figure : The basic concept of TDMA.

- In digital systems, continuous transmission is not required because users do not use the allotted bandwidth all the time. In such cases, TDMA is a complimentary access technique to FDMA.
- Global Systems for Mobile communications (GSM) uses the TDMA technique.
- In TDMA, the entire bandwidth is available to the user but only for a finite period of time. The users are allotted time slots during which they have the entire channel bandwidth at their disposal, as shown in Figure

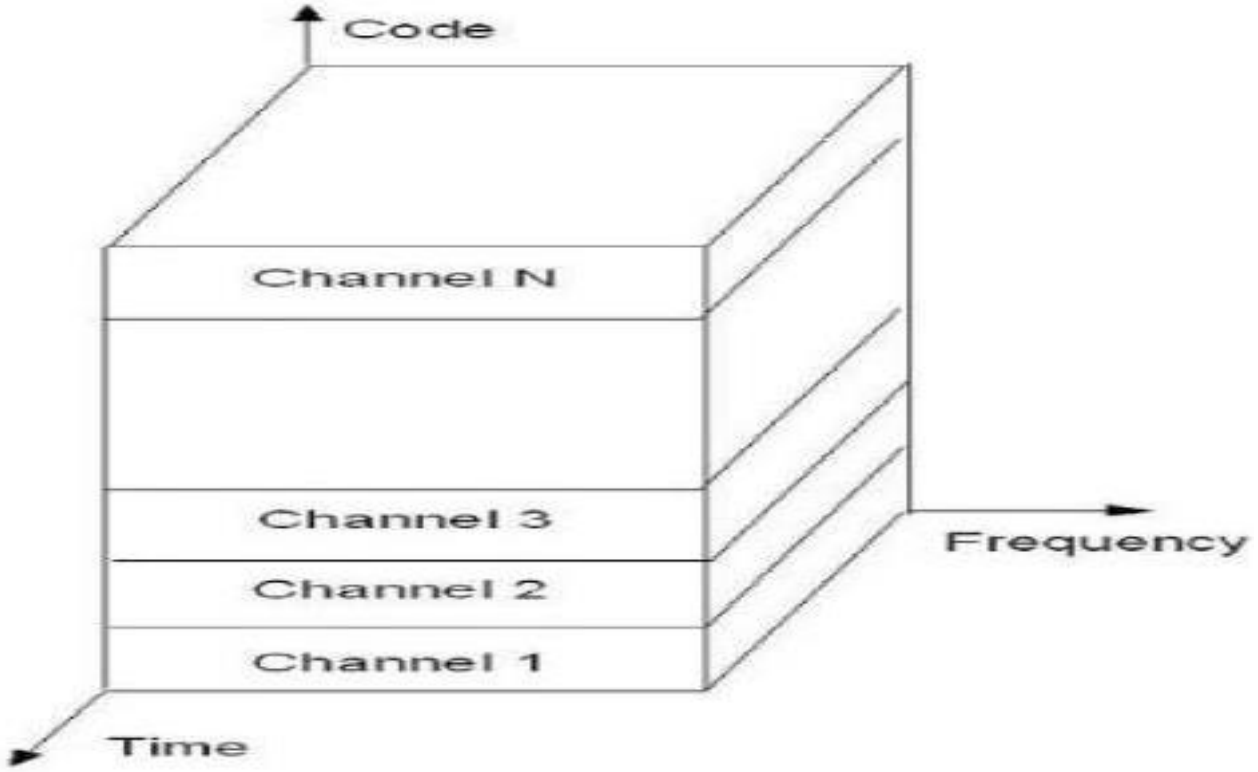
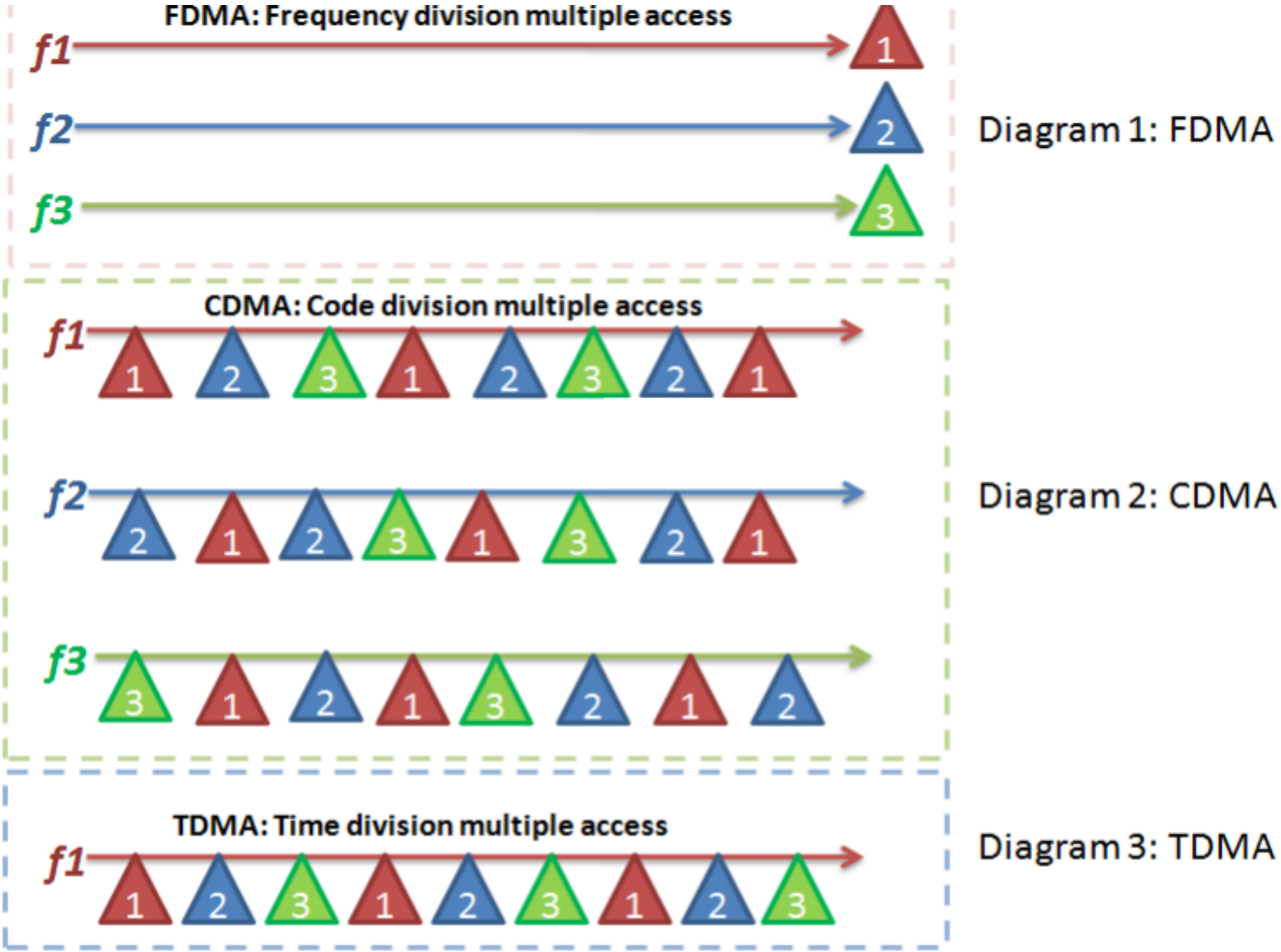


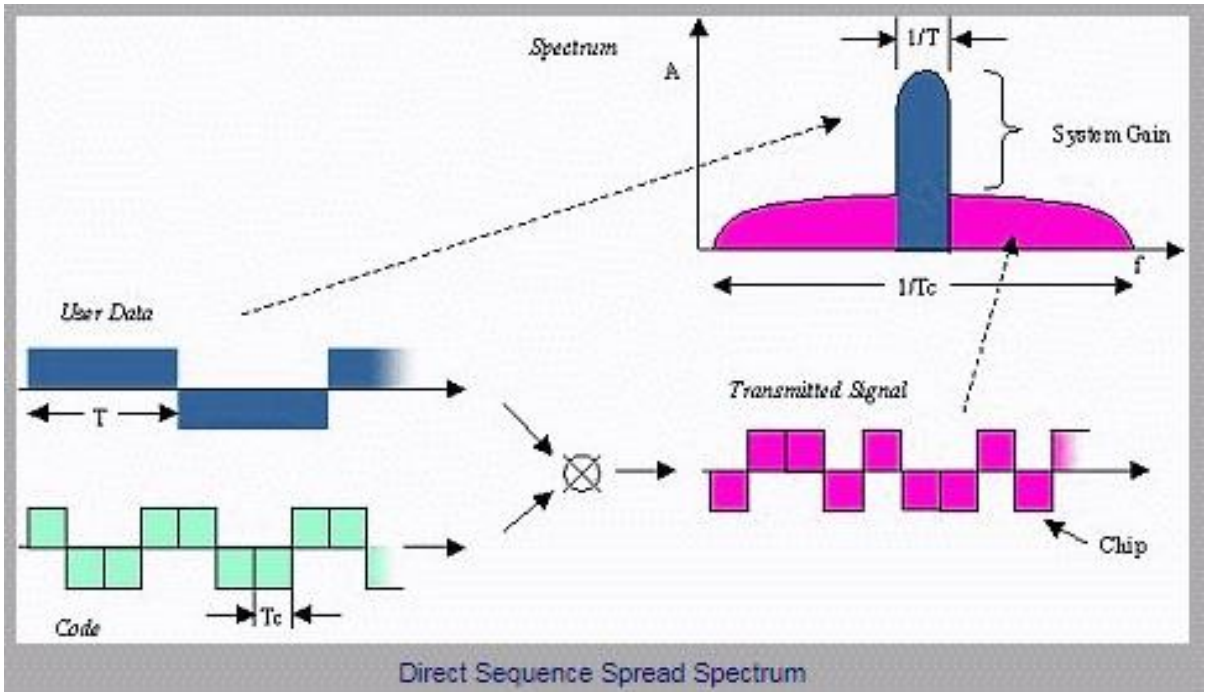
Figure : The basic concept of CDMA.

- In CDMA, the same bandwidth is occupied by all the users, however they are all assigned separate codes, which differentiates them from each other shown in Figure
- CDMA utilize a spread spectrum technique in which a spreading signal (which is uncorrelated to the signal and has a large bandwidth) is used to spread the narrow band message signal.

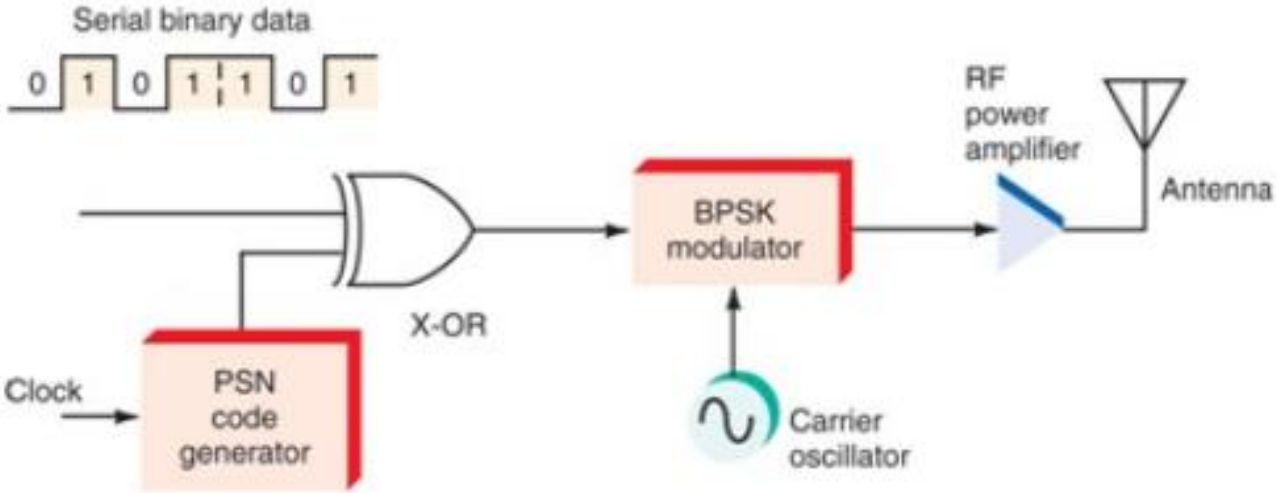

FDMA: Frequency division multiple access
f1
f2
f3
1
2
3
Diagram 1: FDMA
CDMA: Code division multiple access
f1
1 2 3 1 2 3 2 1
f2
2 1 2 3 1 3 2 1
f3
3 1 2 1 3 2 1 2
Diagram 2: CDMA
TDMA: Time division multiple access
f1
1 2 3 1 2 3 1 2 3
Diagram 3: TDMA

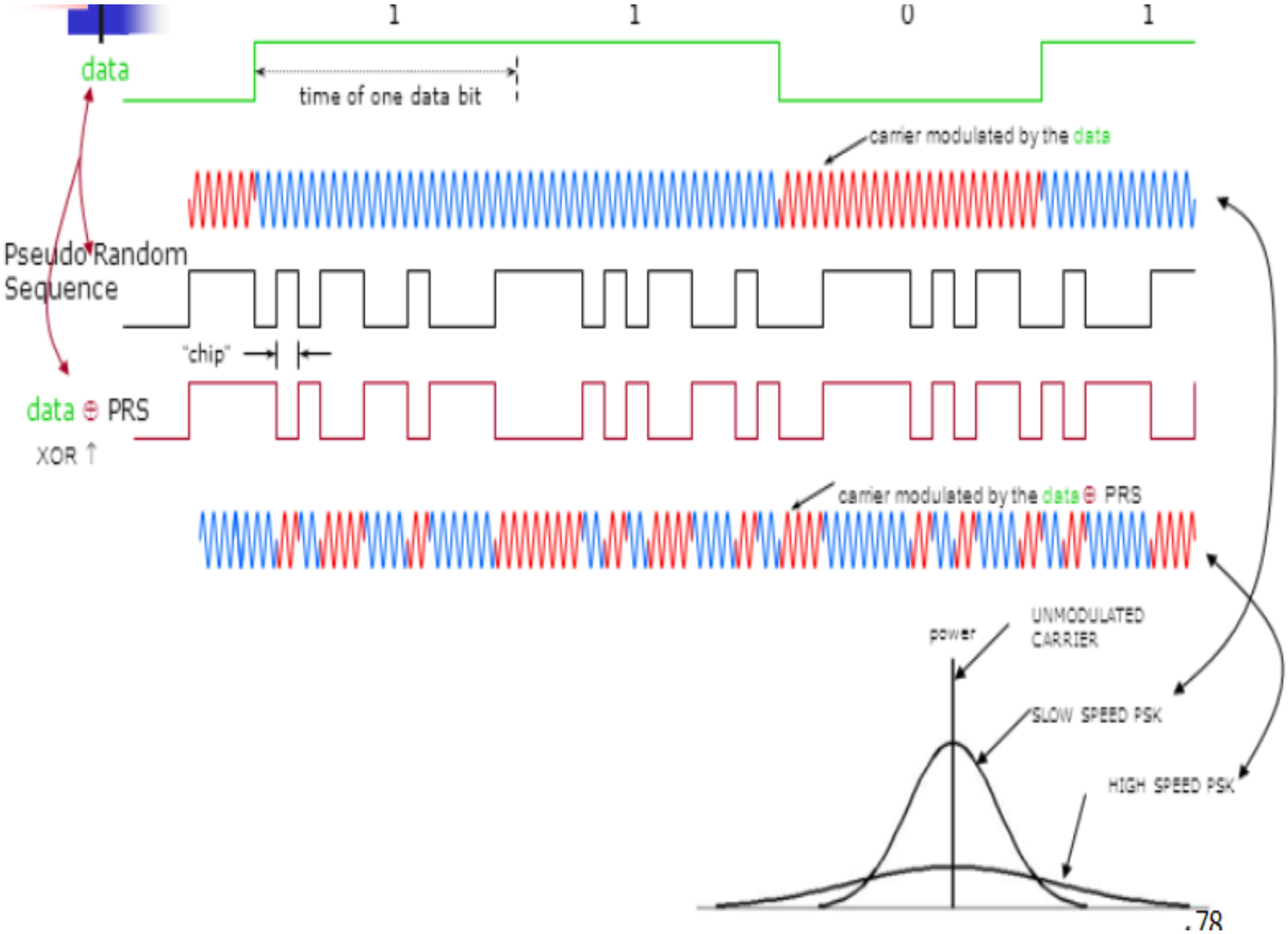
Direct Sequence Spread Spectrum (DSSS)

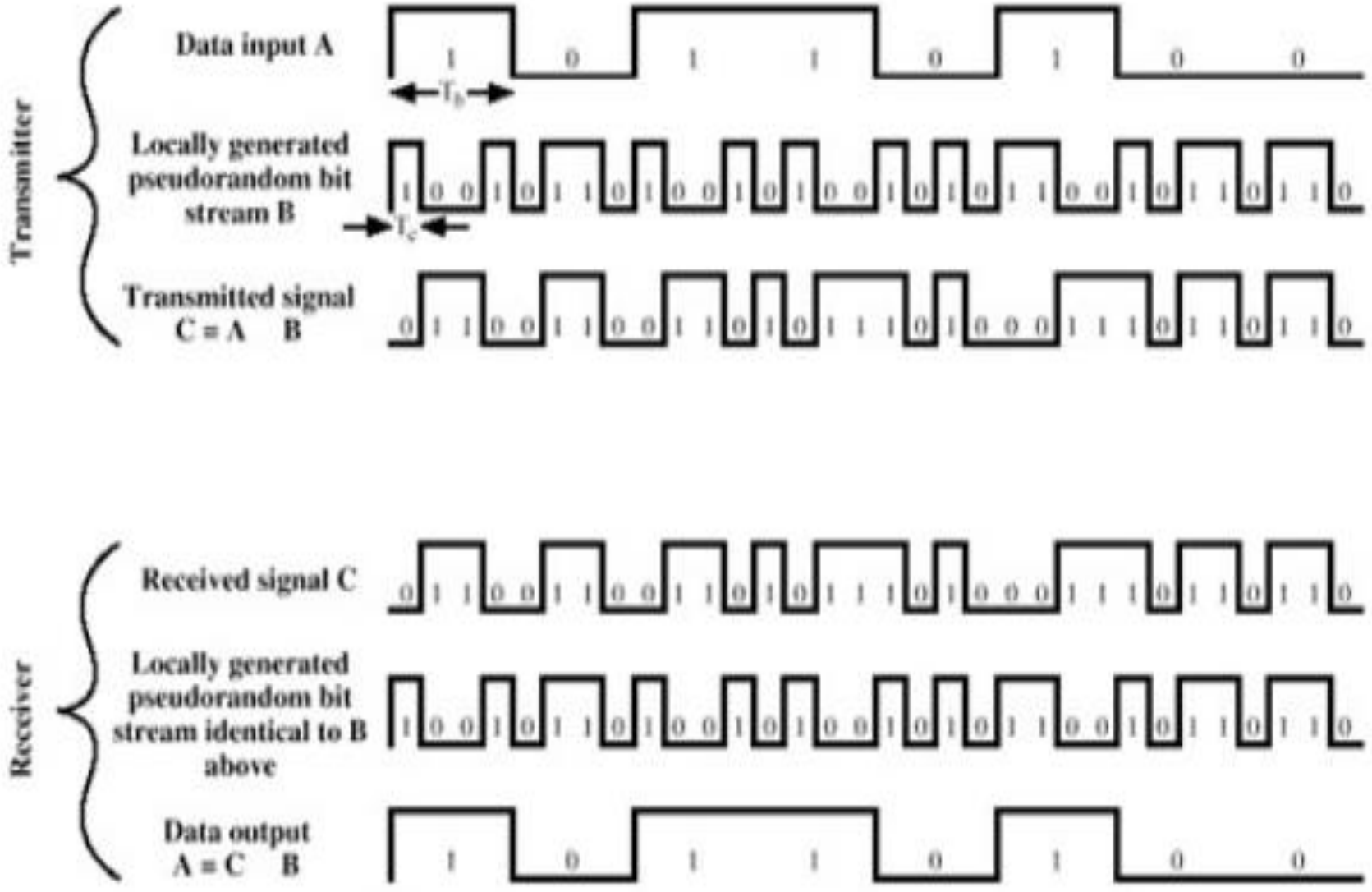
- In DS-SS, the message signal is multiplied by a Pseudo Random Noise Code.
- Each user is given his own codeword which is orthogonal to the codes of other users and in order to detect the user, the receiver must know the codeword used by the transmitter.

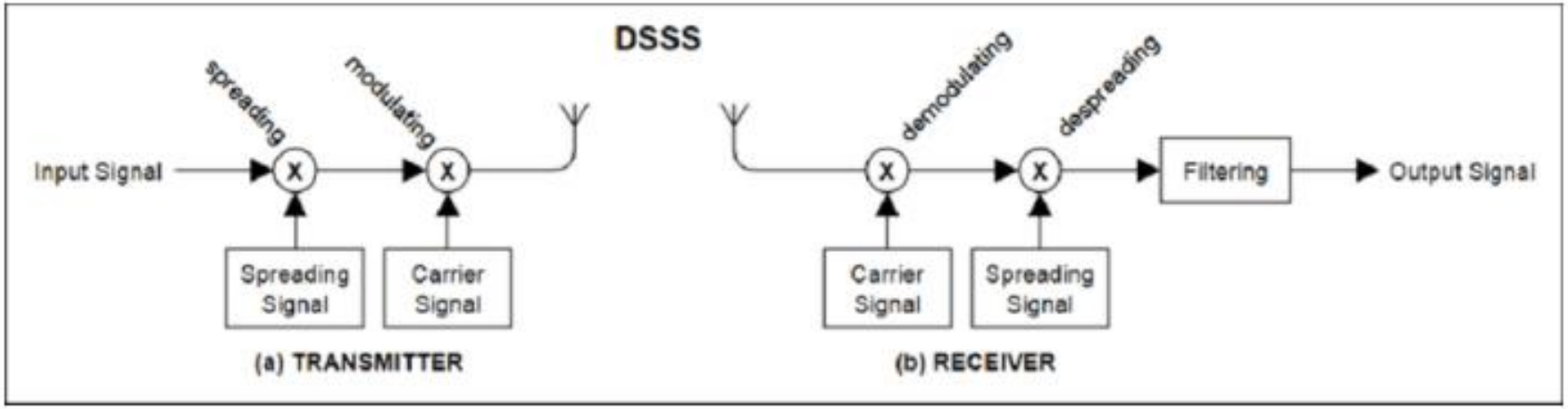


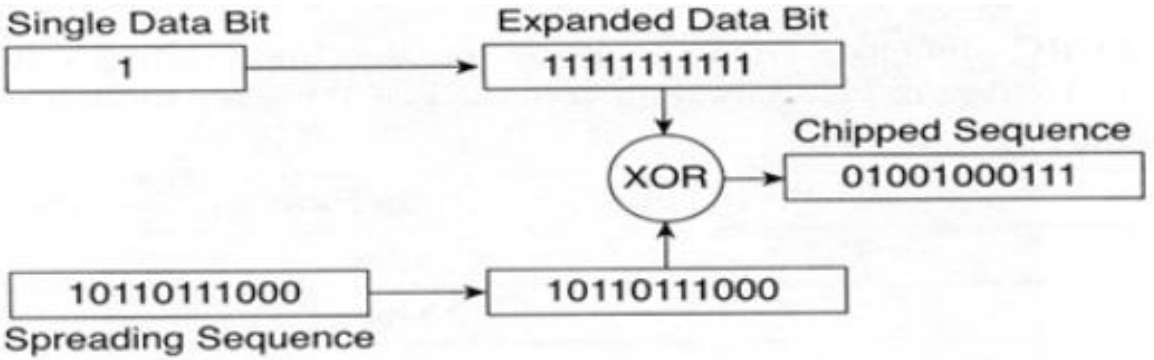
Direct Sequence Spread Spectrum

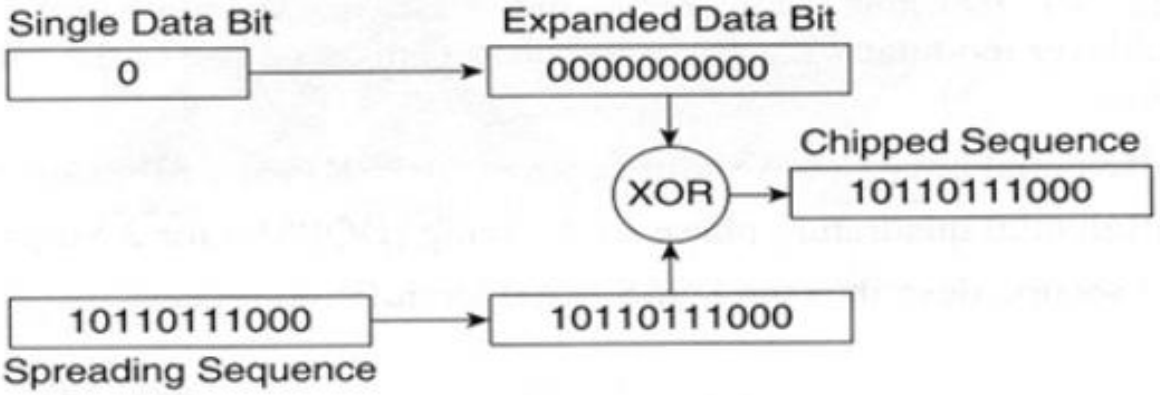



1
1
0
1
data
time of one data bit
carrier modulated by the data
Pseudo Random Sequence
"chip"
data ⊕ PRS
XOR
carrier modulated by the data ⊕ PRS
power
UNMODULATED CARRIER
SLOW SPEED PSK
HIGH SPEED PSK


Transmitter
Data input A
1 0 1 1 0 1 0 0
T_b
Locally generated pseudorandom bit stream B
1 0 0 1 0 1 1 0 1 0 0 1 0 1 0 0 1 0 1 0 1 1 0 0 1 0 1 1 0 1 1 0
T_c
Transmitted signal C = A B
0 1 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 0 1 0 0 0 1 1 1 0 1 1 0 1 1 0
Receiver
Received signal C
0 1 1 0 0 1 1 0 0 1 1 0 1 0 1 1 1 0 1 0 0 0 1 1 1 0 1 1 0 1 1 0
Locally generated pseudorandom bit stream identical to B above
1 0 0 1 0 1 1 0 1 0 0 1 0 1 0 0 1 0 1 0 1 1 0 0 1 0 1 1 0 1 1 0
Data output A = C B
1 0 1 1 0 1 0 0


DSSS
spreading
modulating
Input Signal
X
X
Spreading Signal
Carrier Signal
(a) TRANSMITTER
demodulating
despreading
X
X
Filtering
Output Signal
Carrier Signal
Spreading Signal
(b) RECEIVER


Single Data Bit
1
Expanded Data Bit
11111111111
Chipped Sequence
XOR
01001000111
10110111000
10110111000
Spreading Sequence


Single Data Bit
0
Expanded Data Bit
0000000000
Chipped Sequence
XOR
10110111000
10110111000
10110111000
Spreading Sequence

MATLAB IMPLEMENTATION

PART-I

- Generate a binary message signal using **rand** function. Take size as (1000,1) initially and later change it as per (1XXX,1), where XXX is your three digit roll No.
- Generate a PN sequence using **commsrc.pn** from communication system tool box of MATLAB.
  - Example: H = commsrc.pn('GenPoly', [3 2 0], ...
    'InitialStates', [0 0 1], ...
    'CurrentStates', [0 0 1], ...
    'Mask', [0 0 1], ...
    'NumBitsOut', 8);
    pn=generate(H1); size(pn)=[8, 1]

- Expand the message bit using **repmat**, so as to make it of size [1000, 8]
- Xor(msg, pn) size[1000,8]:
- Convert the vector into row vector: **Spreaded data [8000, 1]**
- Change the binary data into uint8 before modulation
- Choose mod=4, QPSK/QAM
- Add awgn for a single value of SNR
- Demodulate the signal using qamdemod /pskdemod

- Using reshape, convert the serial demodulated data vector to parallel before despreading. [1000, 8]
- Despread the signal by using xor function

  despread_data= xor(demod_data, pn)
- Contract the data to original format by

  msg_rx=round(mean(despread_data,1));

  mean(x,1)→row vector

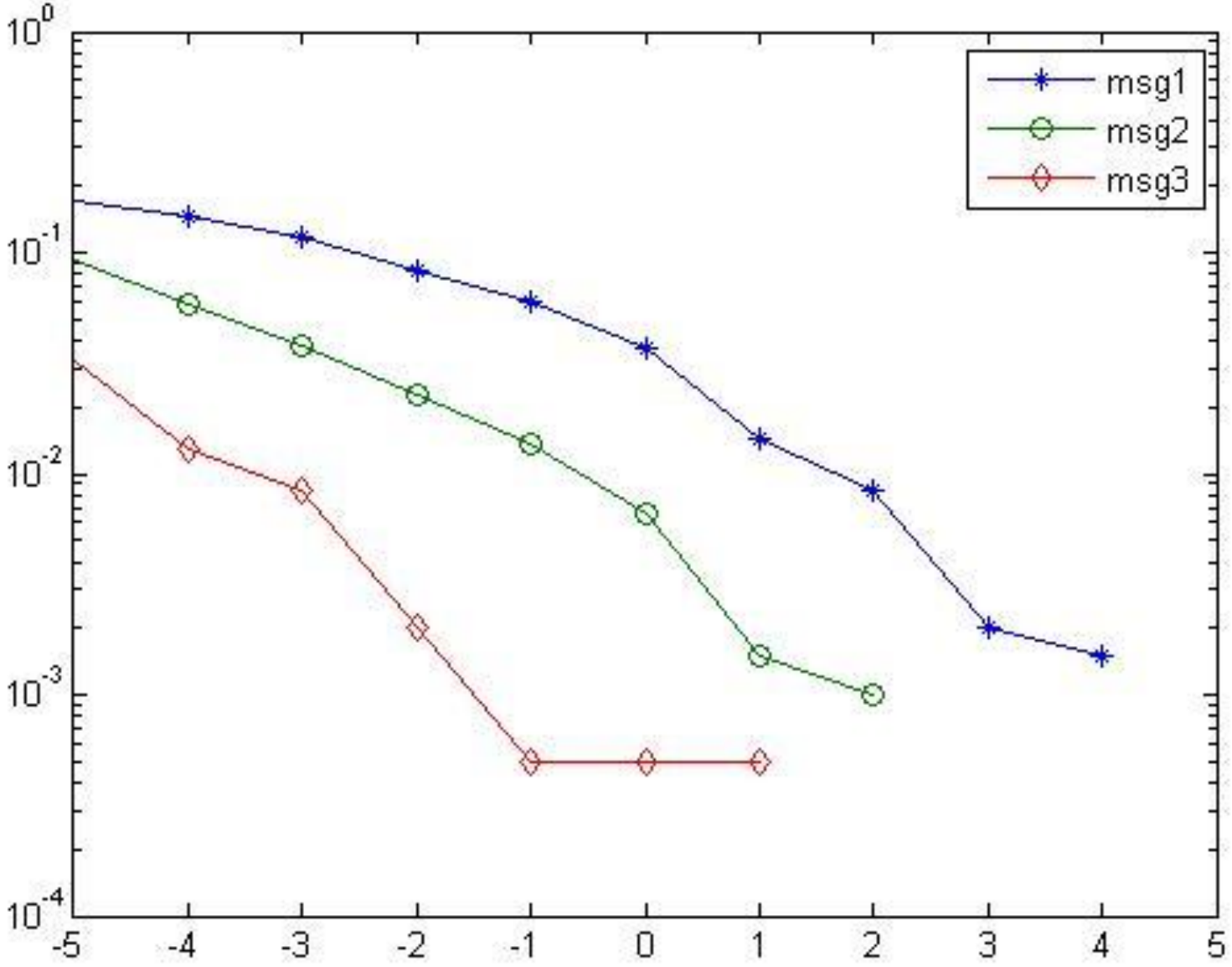
  mean(x,2)→column vector

  Example: [1 0 1 1 1 1 1 1] = 7/8 =0.875≈1

- Calculate the BER
  - BER=mean(abs(msg_rx-msg));
  - This is for a single value or SNR.
- Now, vary SNR [-10:1:10]
  - Plot BER vs SNR
  - **Output plots [choose msg bits=4]**
    - **Msg signal**
    - **PN sequence**
    - **Spreaded data**
    - **Modulated signal**
    - **Demodulated signal**
    - **Despreaded signal**
    - **Received message signal**
    - **SNR versus BER plot [msg bits=1000]**

Part –II

- Choose 3 different message signals of same length using rand function.
- Allocate three PN sequence for each message signal.
- Perform spreading, modulation, demodulation, despreading same as in Part-I for three messages and three PN sequence.
- Plot BER vs SNR for three messages.


msg1
msg2
msg3
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
-5
-4
-3
-2
-1
0
1
2
3
4
5