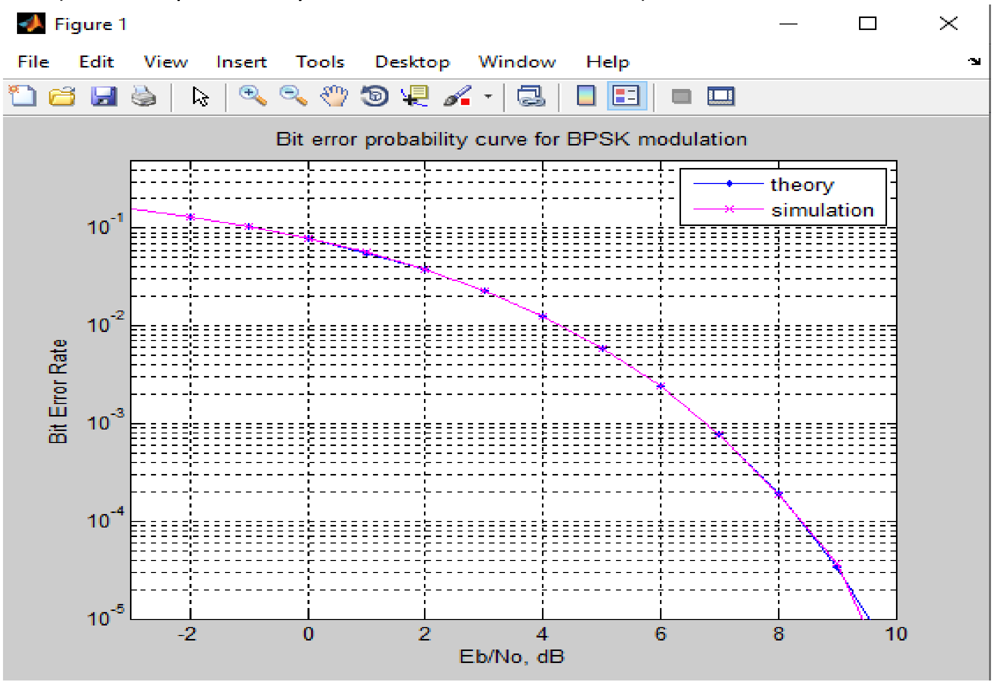
MC Pr 4 (29)

```
clear
N = 10^6 % number of bits or symbols
rand('state',100); % initializing the rand() function
randn('state',200); % initializing the randn() function
% Transmitter
ip = rand(1,N)>0.5; % generating 0,1 with equal probability
s = 2*ip-1; % BPSK modulation 0 -> -1; 1 -> 1
n = 1/sqrt(2)*[randn(1,N) + j*randn(1,N)]; % white gaussian noise, 0dB variance
Eb_N0_dB = [-3:10]; % multiple Eb/N0 values
for ii = 1:length(Eb_N0_dB)
% Noise addition
y = s + 10^(-Eb_N0_dB(ii)/20)*n; % additive white gaussian noise
% receiver - hard decision decoding
ipHat = real(y)>0;
% counting the errors
nErr(ii) = size(find([ip- ipHat]),2);
end
simBer = nErr/N; % simulated ber
theoryBer = 0.5*erfc(sqrt(10.^(Eb_N0_dB/10))); % theoretical ber
% plot
close all
figure
semilogy(Eb_N0_dB,theoryBer,'b.-');
hold on
semilogy(Eb_N0_dB,simBer,'mx-');
axis([-3 10 10^-5 0.5])
grid on
legend('theory', 'simulation');
xlabel('Eb/No, dB');
ylabel('Bit Error Rate');
title('Bit error probability curve for BPSK modulation');
```



```
N = 10^6; % number of bits or symbols
rand('state',100); % initializing the rand() function
randn('state',200); % initializing the randn() function
ip = rand(1,N)>0.5; % generating 0,1 with equal probability
s = 2*ip-1; % BPSK modulation: 0 -> -1; 1 -> 1
n = 1/sqrt(2)*[randn(1,N) + 1j*randn(1,N)]; % AWGN noise
 Eb_N0_dB = 0:35; % Updated Eb/N0 range to match the graph
 nErrAWGN = zeros(1, length(Eb_N0_dB));
nErrRayleigh = zeros(1, length(Eb_N0_dB));
 for ii = 1:length(Eb_N0_dB)
   yAWGN = s + 10^(-Eb_N0_dB(ii)/20)*n; % AWGN channel
   h = 1/sqrt(2) * (randn(1,N) + 1j*randn(1,N)); % Rayleigh channel
   yRayleigh = h .* s + 10^(-Eb_N0_dB(ii)/20)*n;  % Rayleigh + AWGN
   ipHatAWGN = real(yAWGN)>0;
   nErrAWGN(ii) = sum(ip ~= ipHatAWGN);
   yRayleigh_eq = yRayleigh ./ h; % Equalization
   ipHatRayleigh = real(yRayleigh_eq) > 0;
   nErrRayleigh(ii) = sum(ip ~= ipHatRayleigh);
end
simBerAWGN = nErrAWGN / N; % Simulated BER for AWGN
simBerRayleigh = nErrRayleigh / N; % Simulated BER for Rayleigh

theoryBerAWGN  = 0.5*erfc(sqrt(10.^(Eb_N0_dB/10)));  % Theoretical BER for AWGN
theoryBerRayleigh = 0.5 * (1 - sqrt(10.^(Eb_N0_dB/10) ./ (1 + 10.^(Eb_N0_dB/10))));
figure;
semilogy(Eb_N0_dB, theoryBerAWGN,  'c-d', 'LineWidth', 1.5); hold on; % AWGN-Theory  (cyan, diamond)
semilogy(Eb_N0_dB, theoryBerRayleigh, 'b-s', 'LineWidth', 1.5); % Rayleigh-Theory (blue, square)
semilogy(Eb_N0_dB, simBerRayleigh, 'm-x', 'LineWidth', 1.5); % Rayleigh-Simulation (magenta, x)
 axis([0 35 1e-5 0.5]);
grid on;
legend('AWGN-Theory',  'Rayleigh-Theory', 'Rayleigh-Simulation');
xlabel('Eb/No, dB');
ylabel('Bit Error Rate');
title('BER for BPSK modulation in Rayleigh channel');
```