
```

%QPSK with Rayleigh fading & AWGN
clc;
close all;
clear all;

bit_count = 10000;
data =randi(2,bit_count)-ones(bit_count,bit_count);

Eb_No = -3: 1: 20;
SNR = Eb_No + 10*log10(2);
for aa = 1: 1: length(SNR)
    T_Errors = 0;
    T_bits = 0;

    while T_Errors < 100
        uncoded_bits = round(rand(1,bit_count));
        B1 = uncoded_bits(1:2:end);
        B2 = uncoded_bits(2:2:end);qpsk_sig =
        ((B1==0).*(B2==0)*(exp(1i*pi/4)))+(B1==0).*(B2==1)...
        *(exp(3*1i*pi/4)))+(B1==1).*(B2==1)*(exp(5*1i*pi/4))...
        +(B1==1).*(B2==0)*(exp(7*1i*pi/4)));
        ray =
        sqrt(0.5*((randn(1,length(qpsk_sig))).^2+(randn(1,length(qpsk_sig))).^2));
        rx = qpsk_sig.*ray;
        N0 = 1/10^(SNR(aa)/10);
        rx = rx +
        sqrt(N0/2)*(randn(1,length(qpsk_sig))+1i*randn(1,length(qpsk_sig)));
        rx = rx./ray;
        B4 = (real(rx)<0);
        B3 = (imag(rx)<0);
        uncoded_bits_rx = zeros(1,2*length(rx));
        uncoded_bits_rx(1:2:end) = B3;
        uncoded_bits_rx(2:2:end) = B4;
        diff = uncoded_bits - uncoded_bits_rx;
        T_Errors = T_Errors + sum(abs(diff));
        T_bits = T_bits + length(uncoded_bits);
    end
    figure; clf;
    plot(real(rx),imag(rx),'o'); % Scatter Plot
    title(['constellation of received symbols for SNR = ',
    num2str(SNR(aa))]);
    xlabel('Inphase Component'); ylabel('Quadrature Component');
    BER(aa) = T_Errors / T_bits;
    disp(sprintf('bit error probability = %f',BER(aa)));
end
figure(1);
semilogy(SNR,BER,'or');
hold on;
xlabel('SNR (dB)');
ylabel('BER');
title('SNR Vs BER plot for QPSK Modulation in Rayleigh Channel');

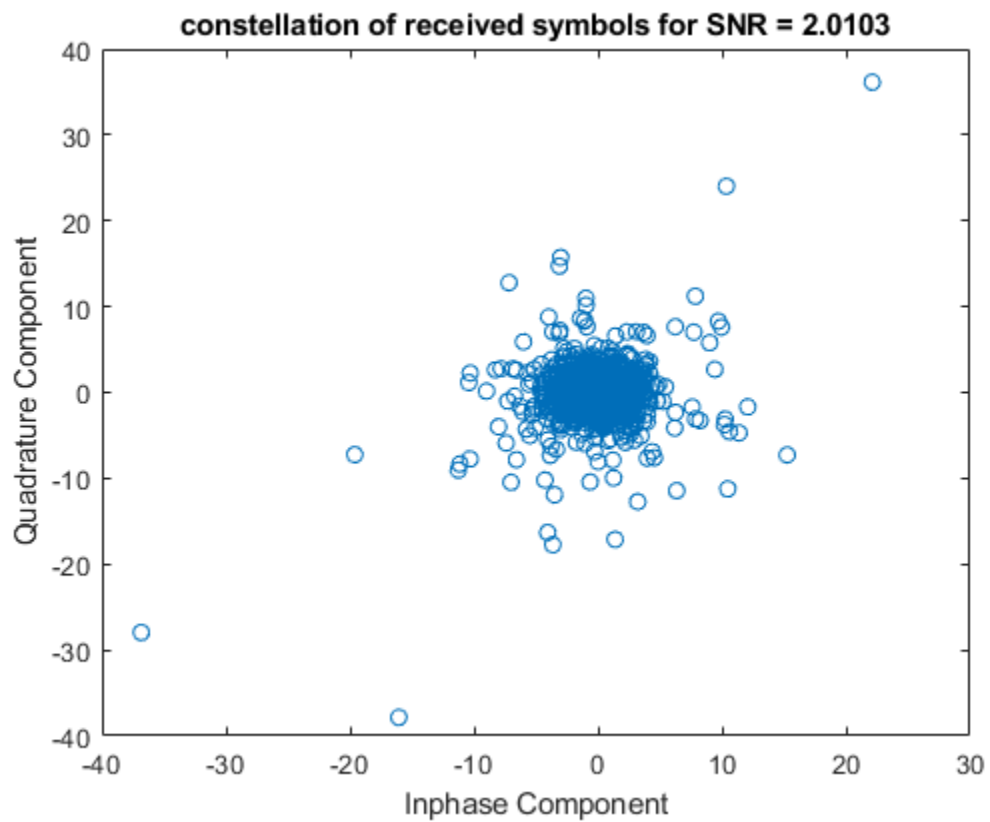
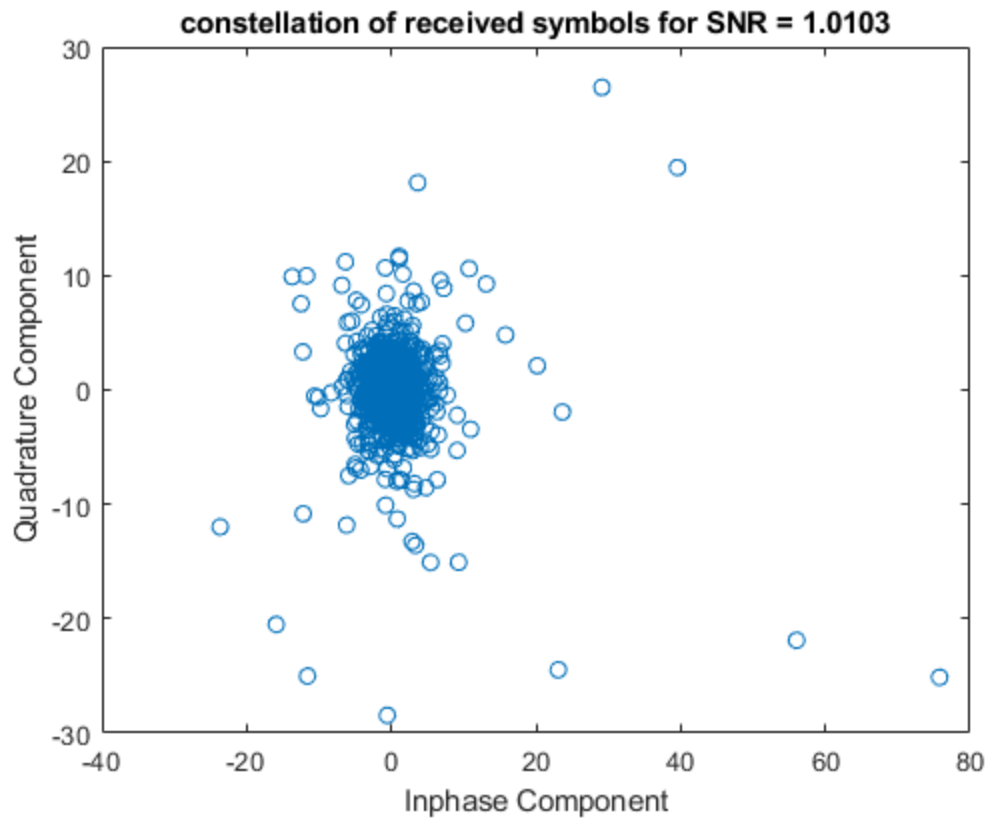
```

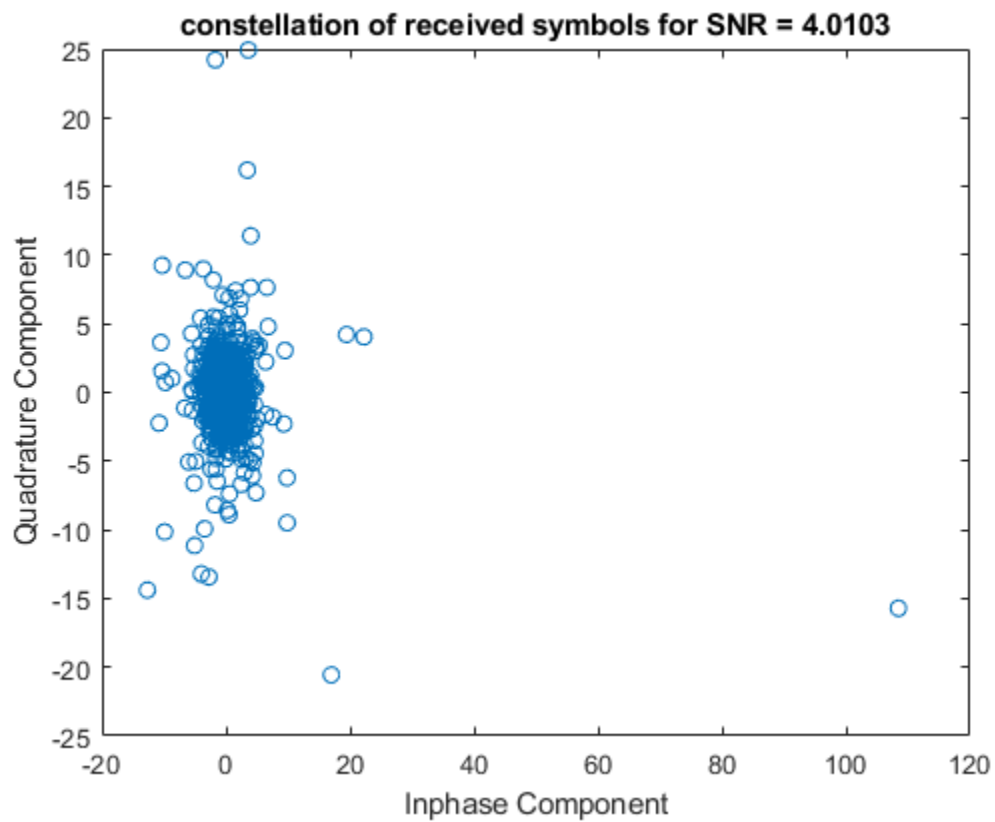
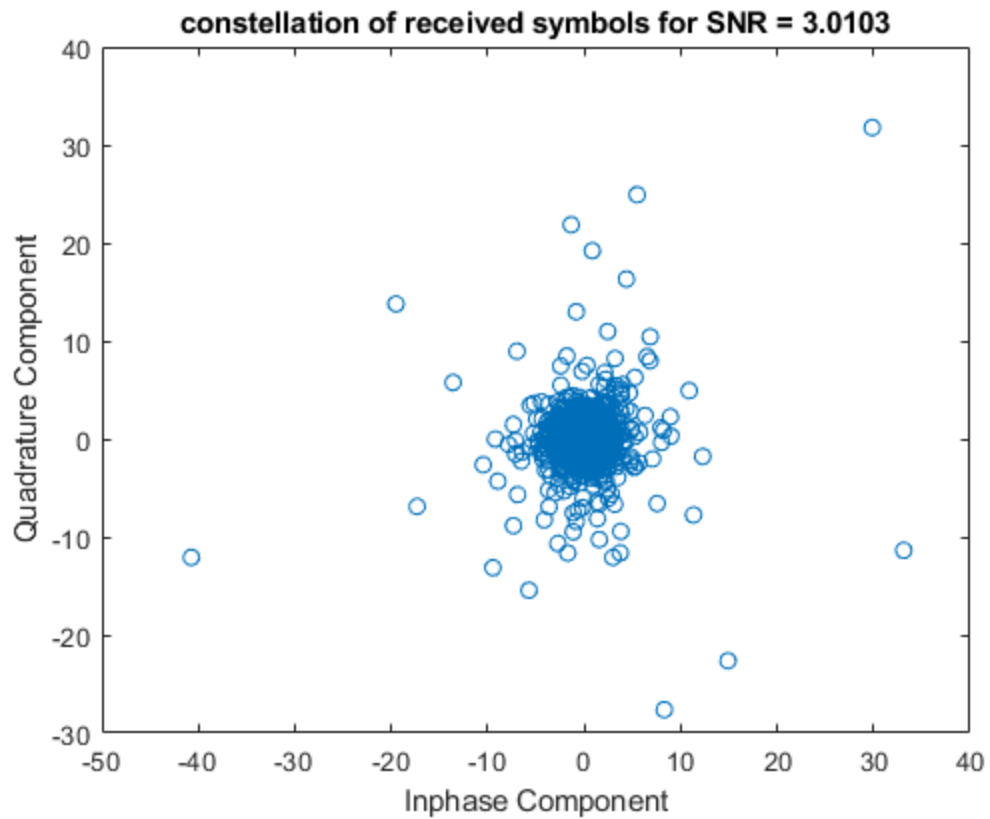
```

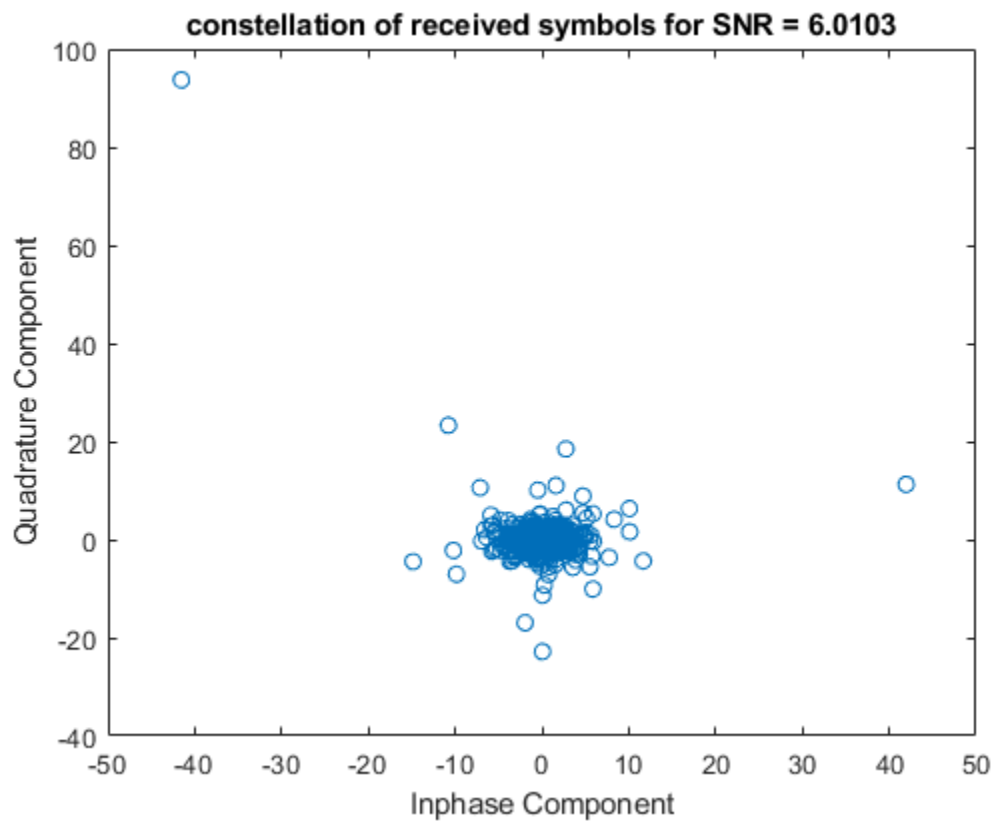
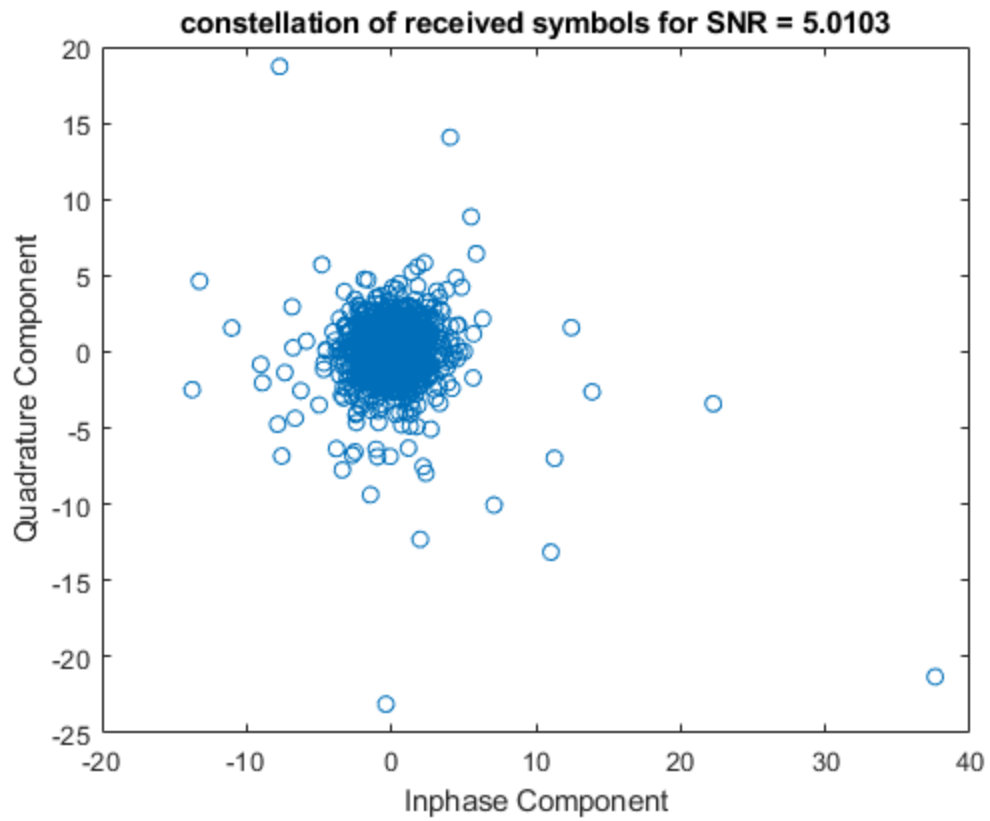
figure(1);
EbN0Lin = 10.^(Eb_No/10);
theoryBerRay = 0.5.*(1-sqrt(EbN0Lin./(EbN0Lin+1)));
semilogy(SNR,theoryBerRay);
grid on;
figure(1);
theoryBerAWGN = 0.5*erfc(sqrt(10.^(Eb_No/10)));
semilogy(SNR,theoryBerAWGN,'g-+'); grid on;
legend('Simulated', 'Theoretical Rayleigh', 'Theoretical AWGN');
axis([SNR(1,1) SNR(end-3) 0.00001 1]);

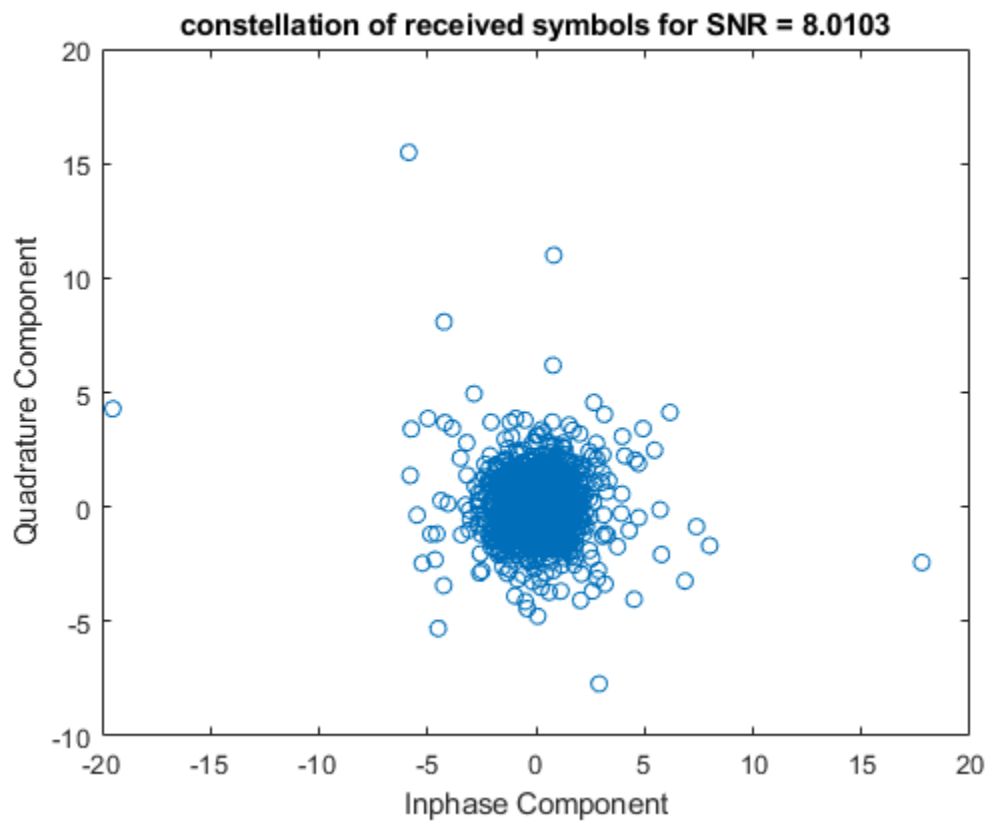
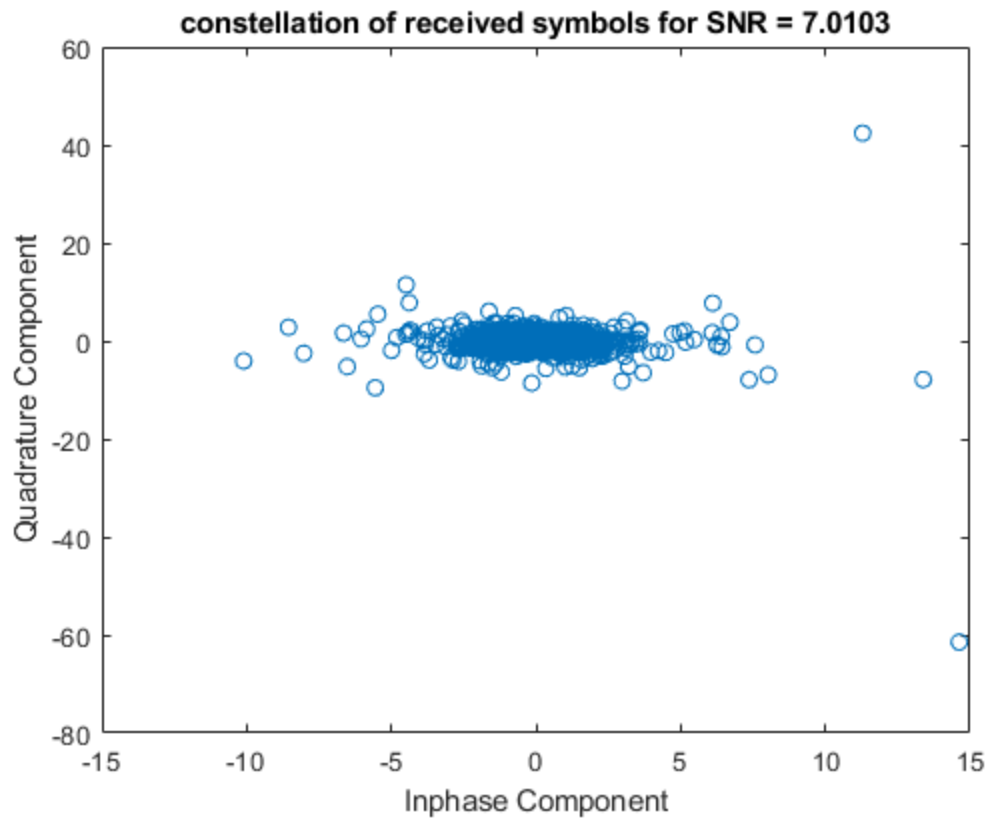
bit error probability = 0.213800
bit error probability = 0.184200
bit error probability = 0.166800
bit error probability = 0.146900
bit error probability = 0.119900
bit error probability = 0.108900
bit error probability = 0.099100
bit error probability = 0.080000
bit error probability = 0.062700
bit error probability = 0.056300
bit error probability = 0.044900
bit error probability = 0.037300
bit error probability = 0.027900
bit error probability = 0.023800
bit error probability = 0.016400
bit error probability = 0.016100
bit error probability = 0.013400
bit error probability = 0.008350
bit error probability = 0.007500
bit error probability = 0.005400
bit error probability = 0.004900
bit error probability = 0.003867
bit error probability = 0.003467
bit error probability = 0.002675

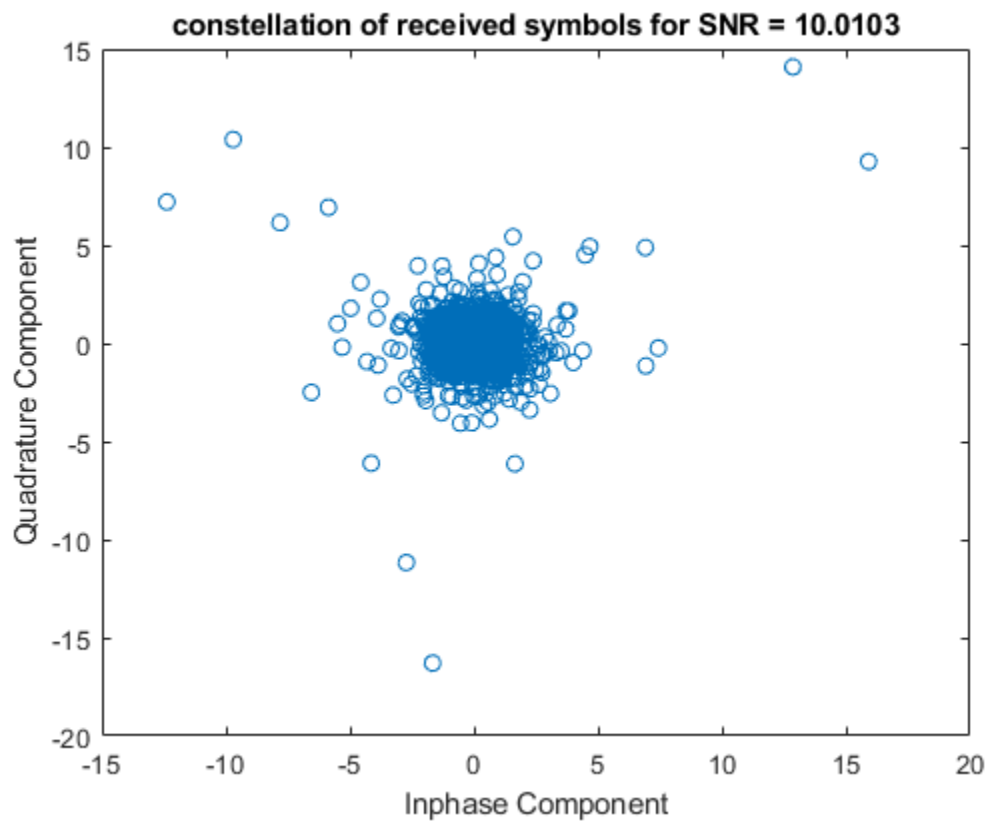
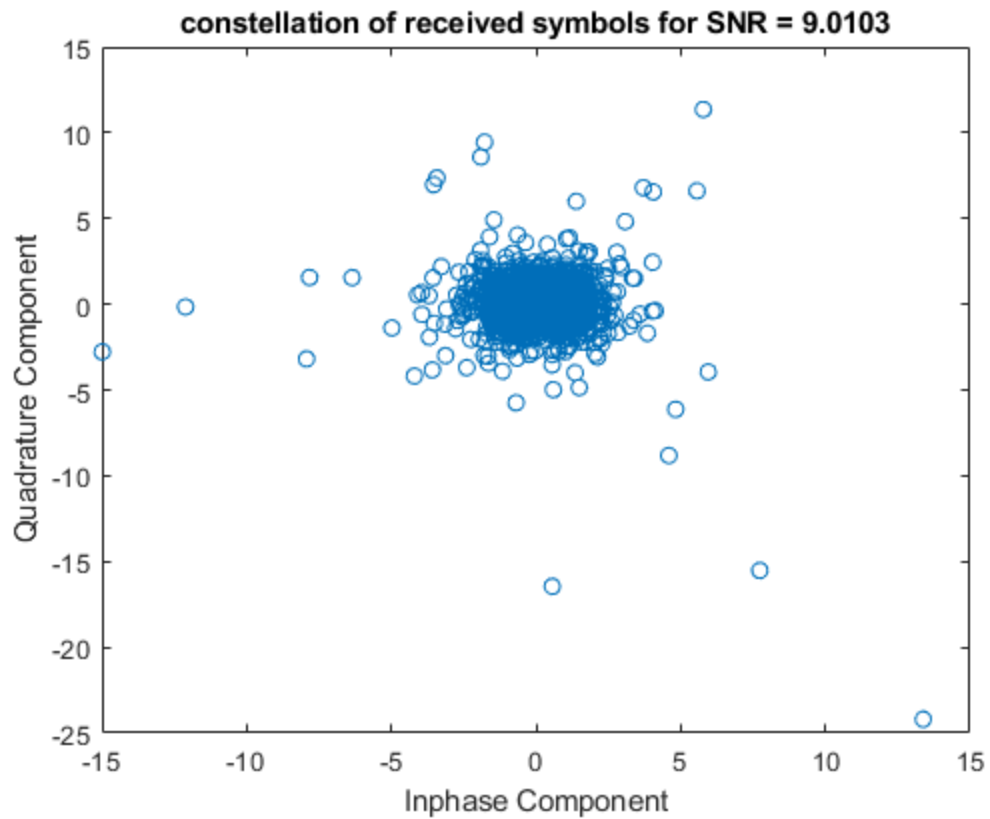
```

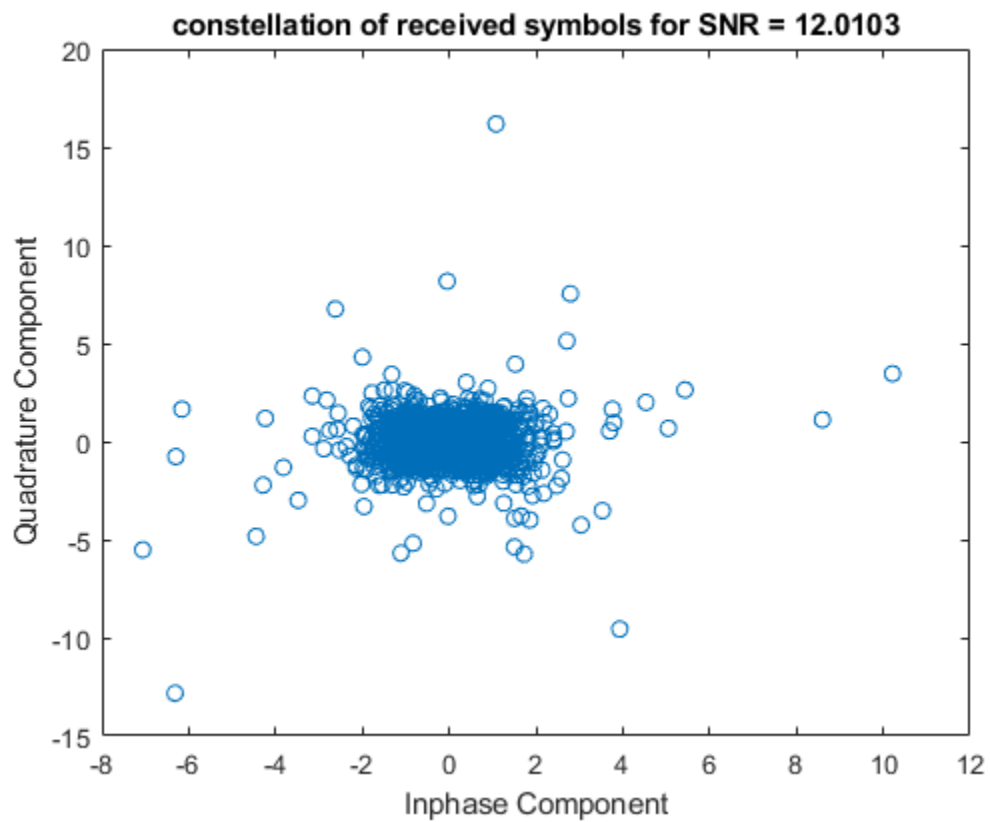
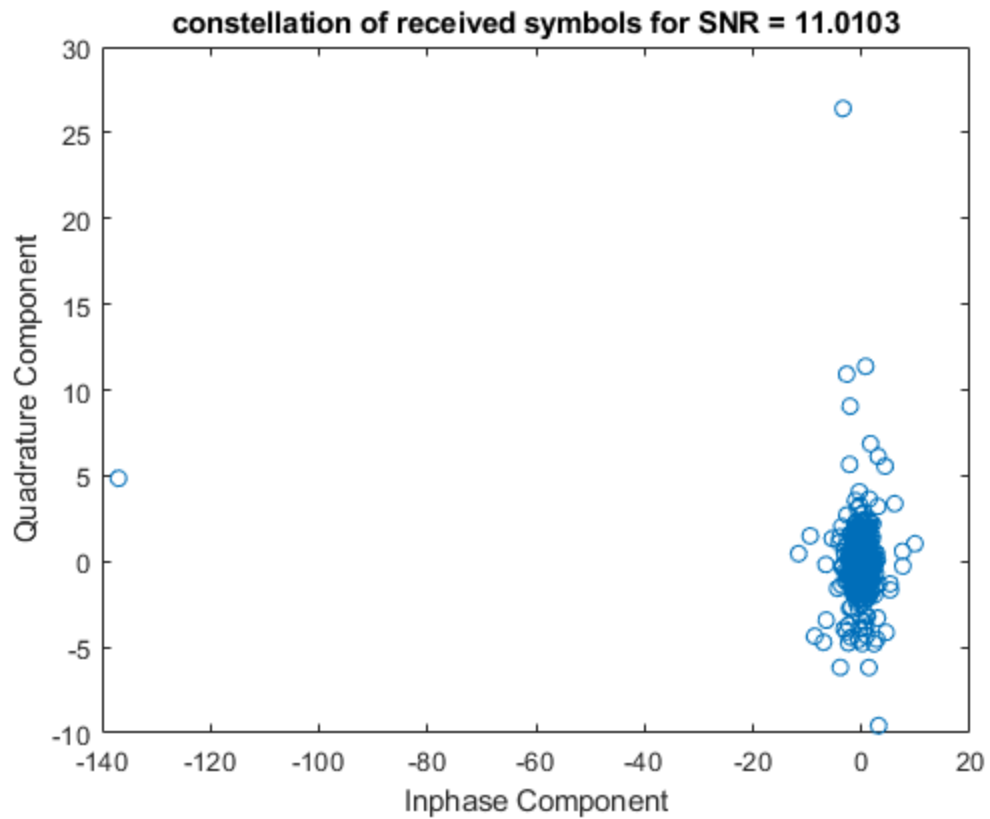


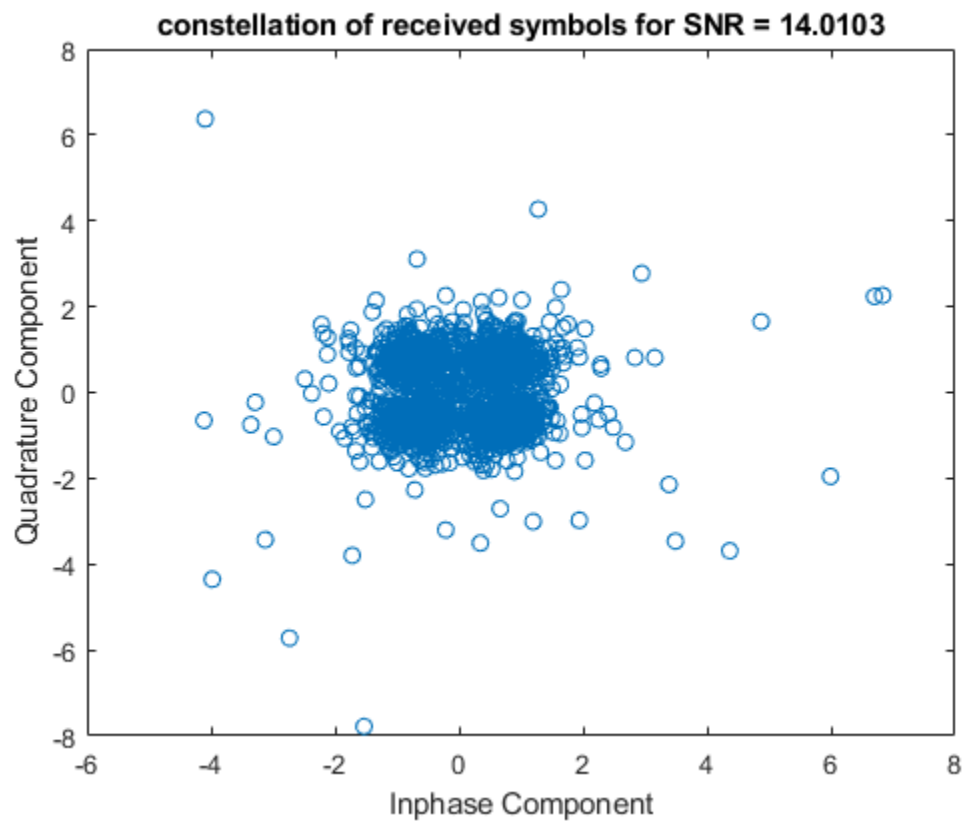
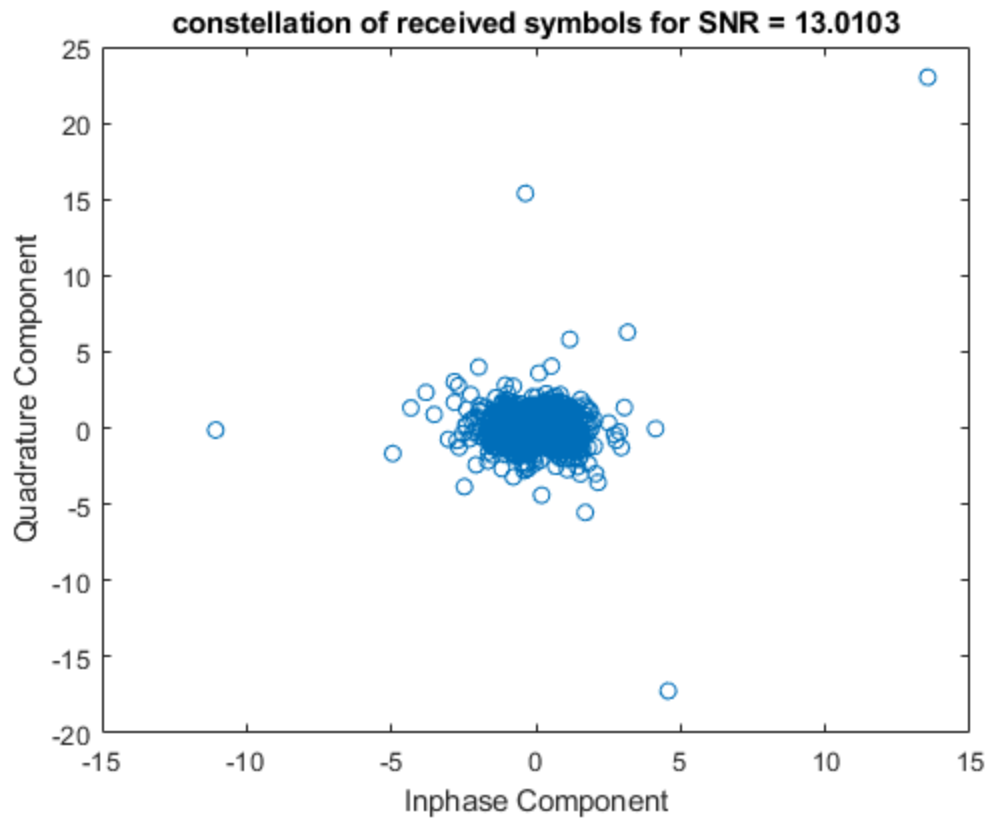


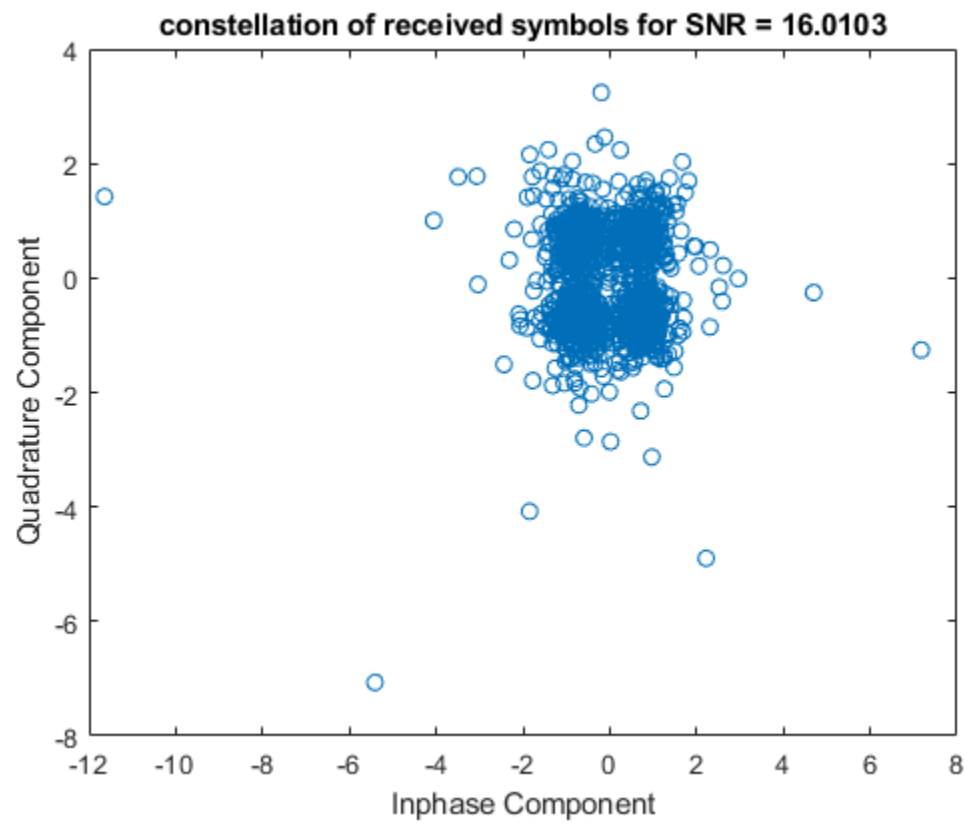
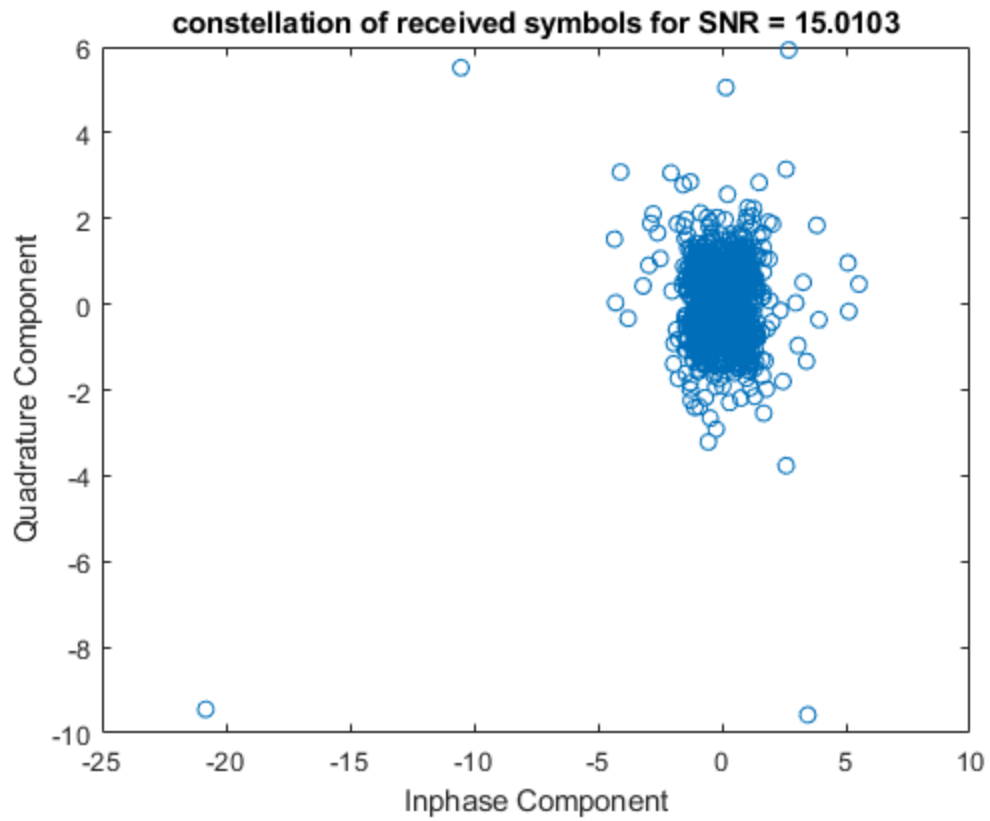


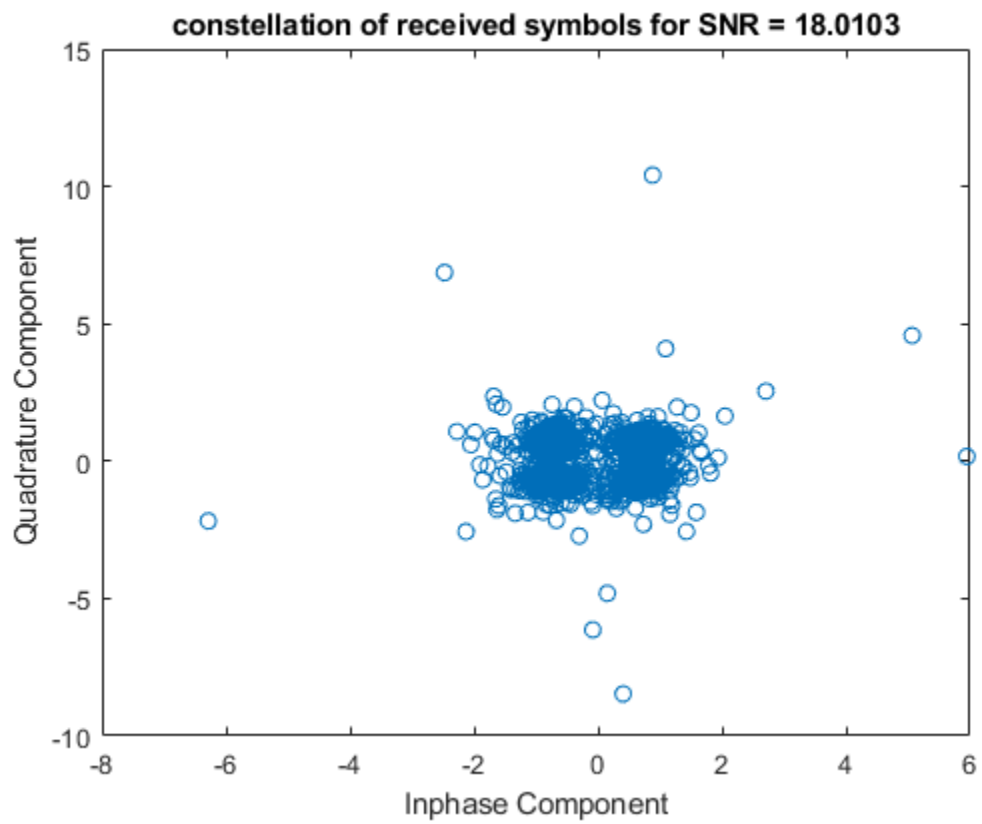
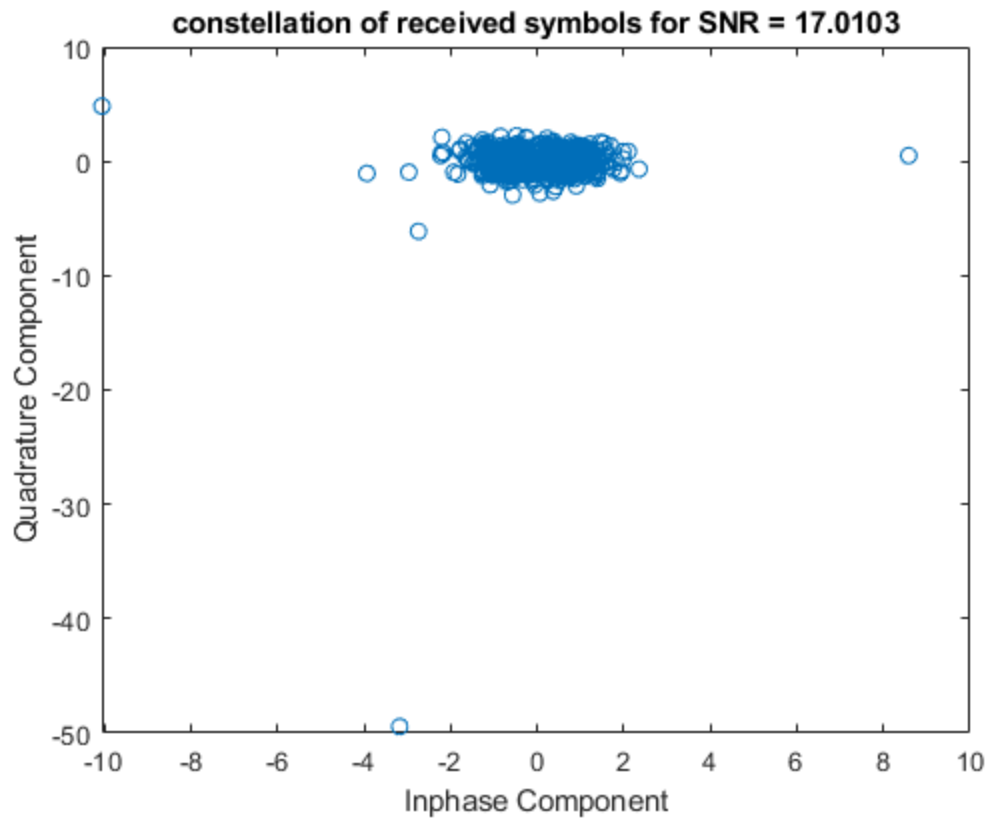


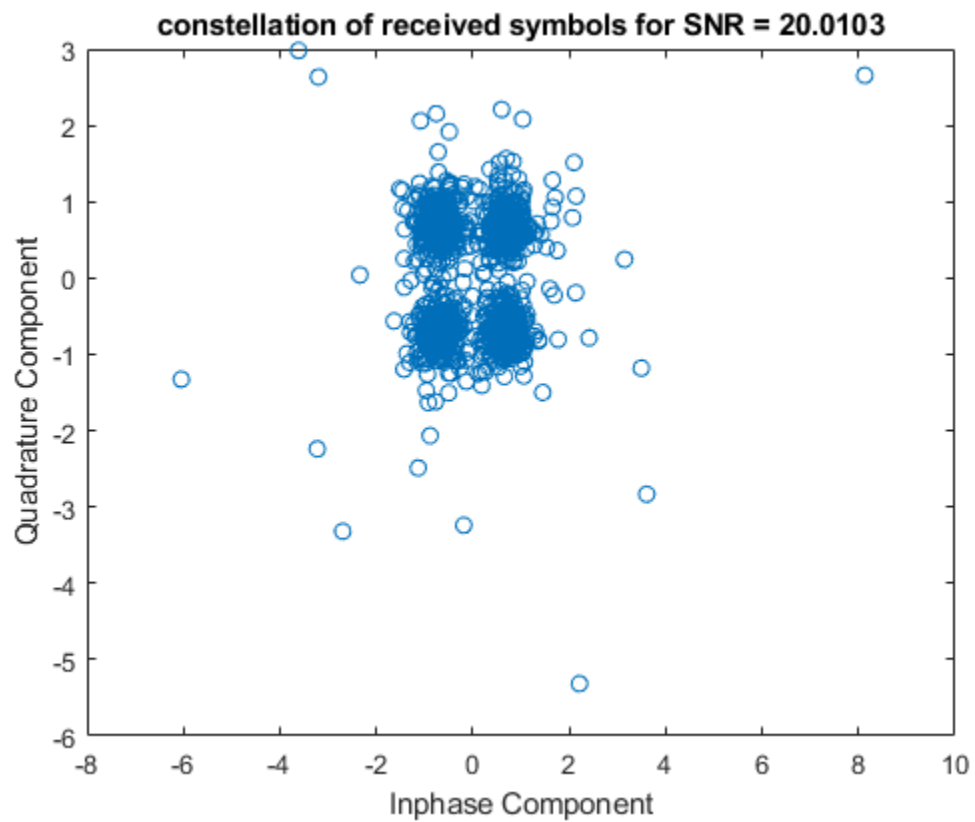
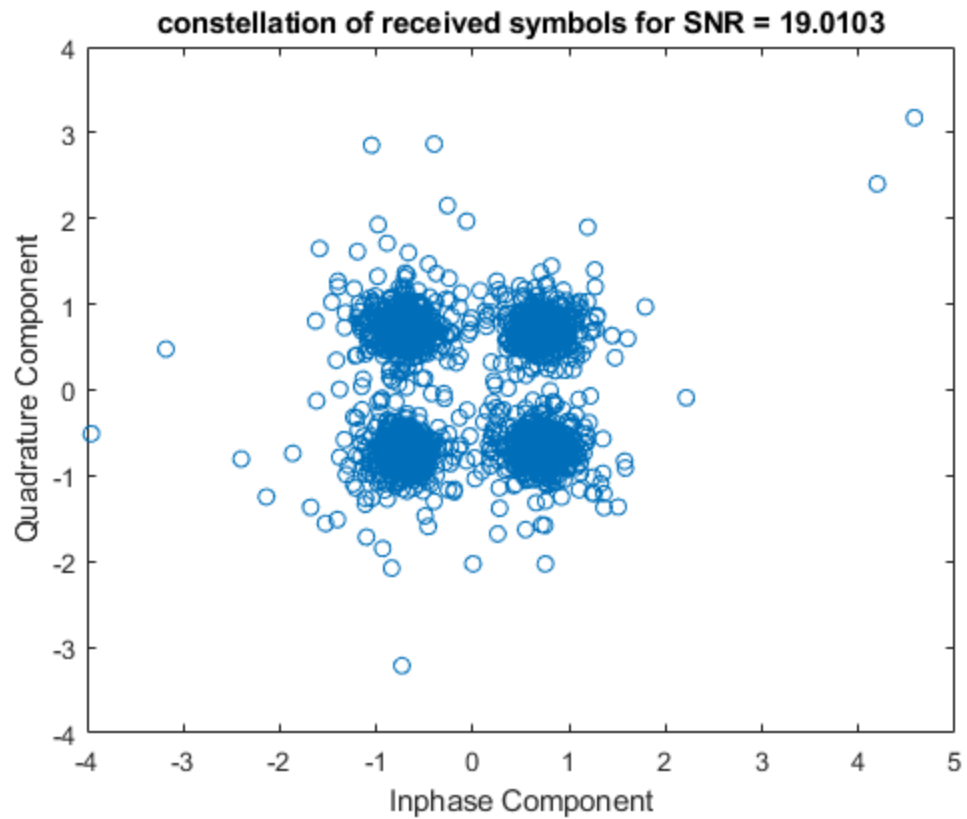


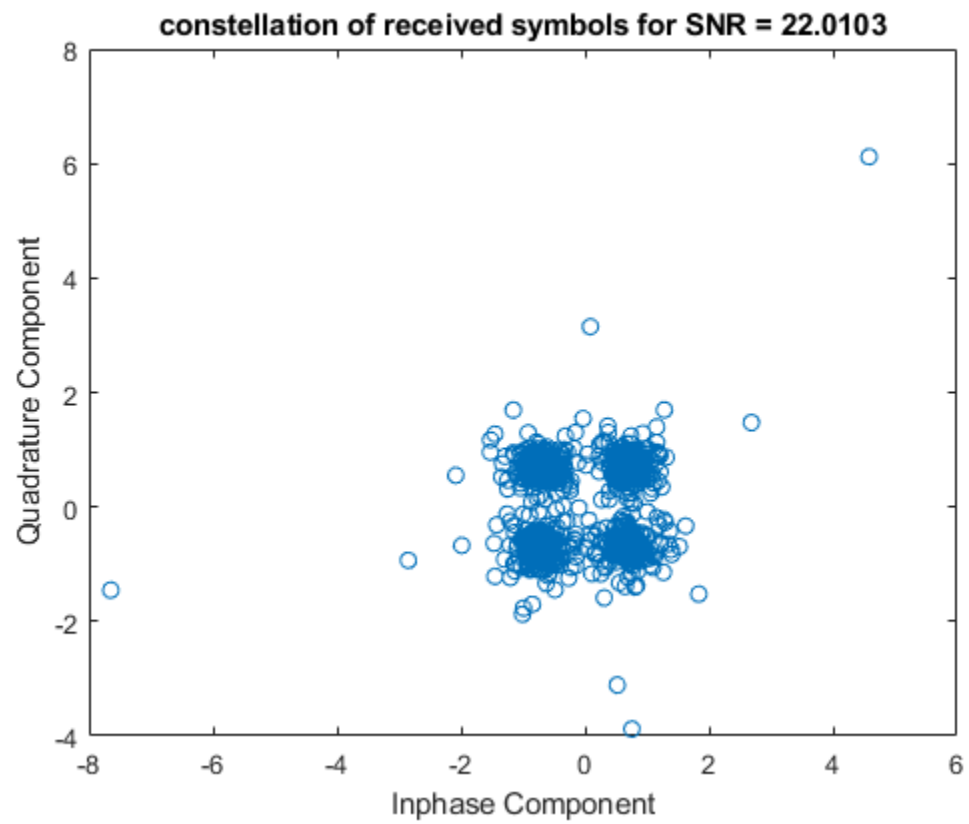
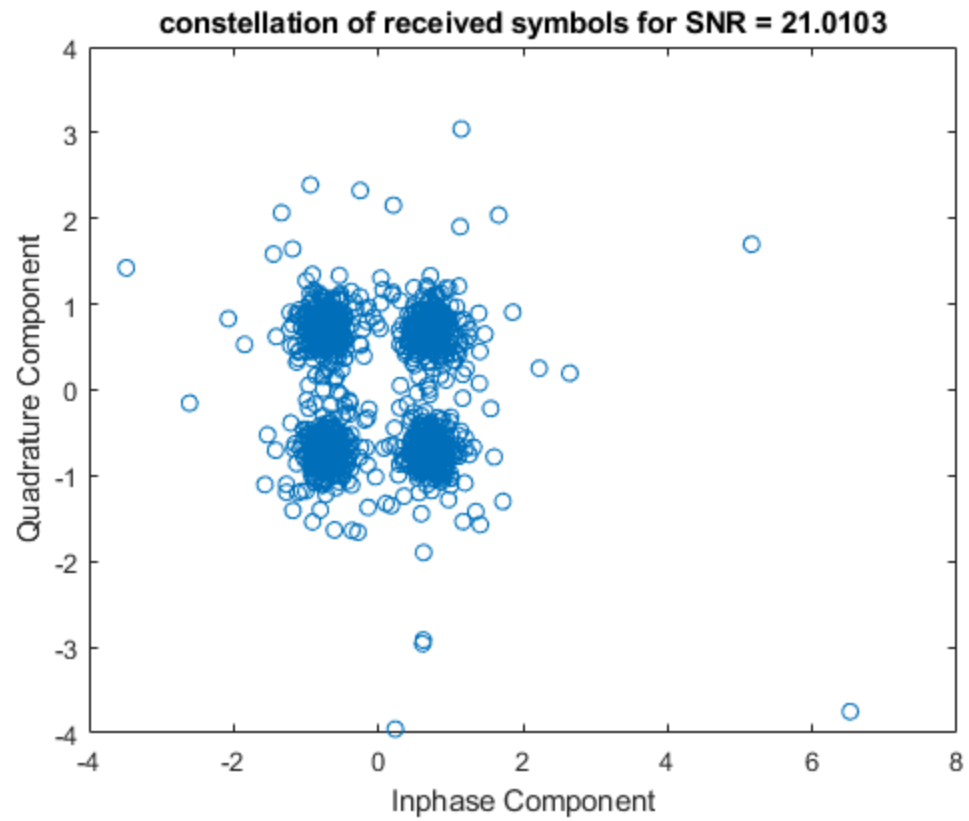


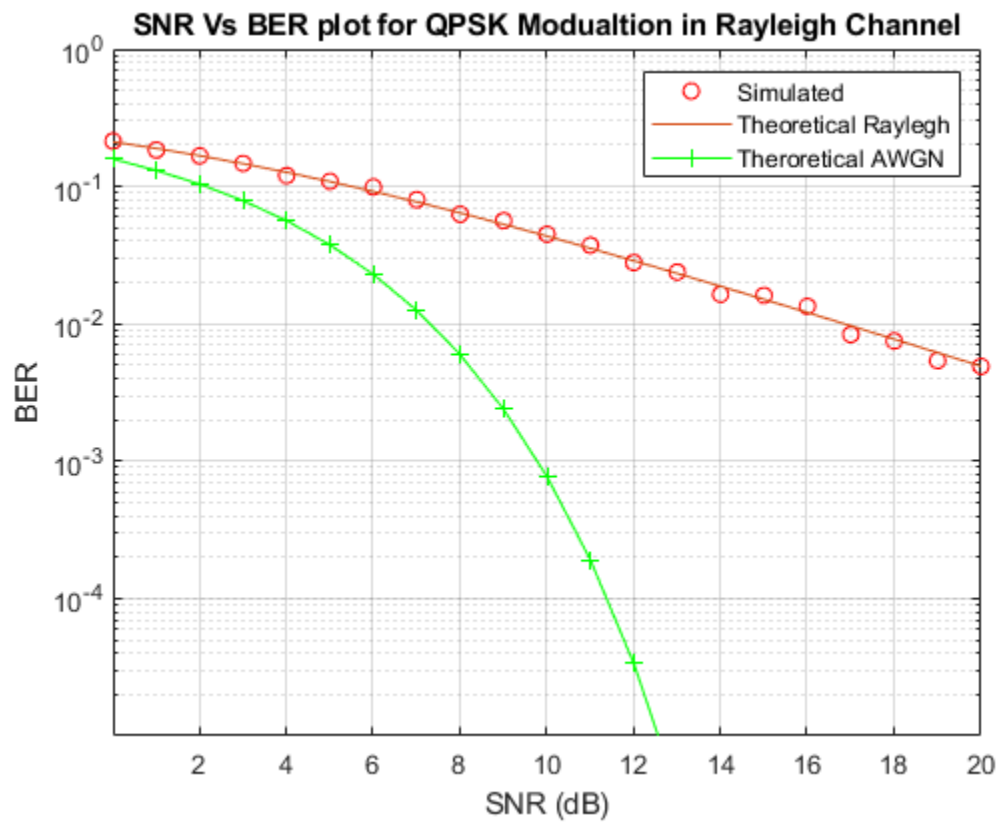
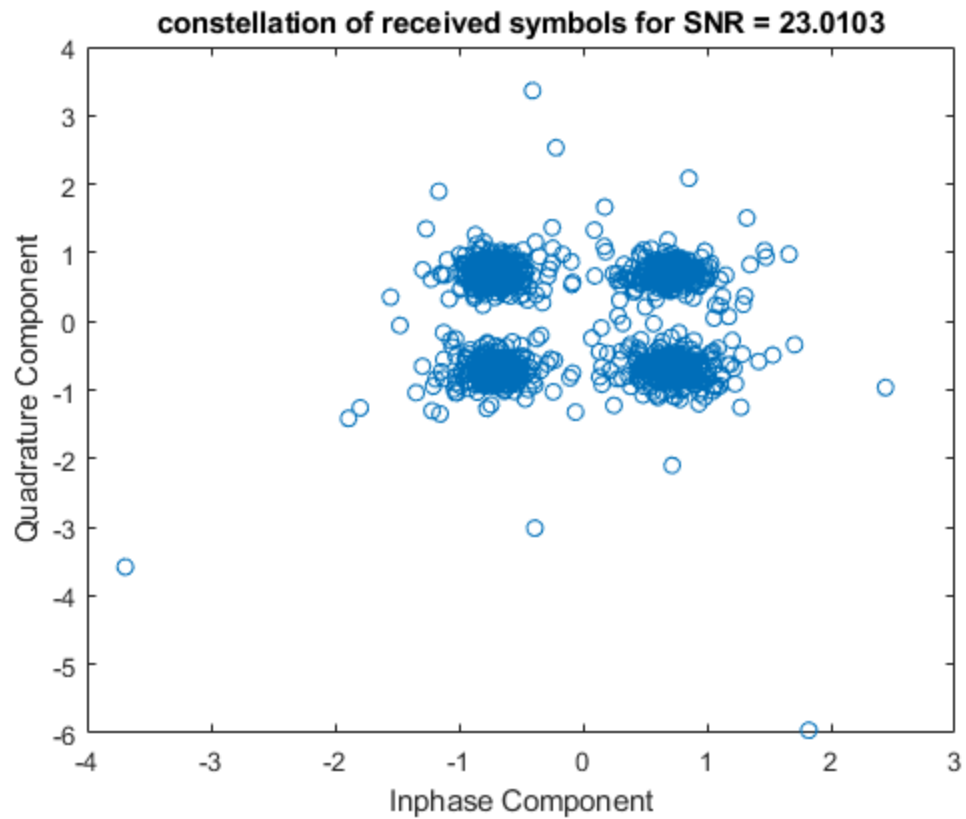












Published with MATLAB® R2019b