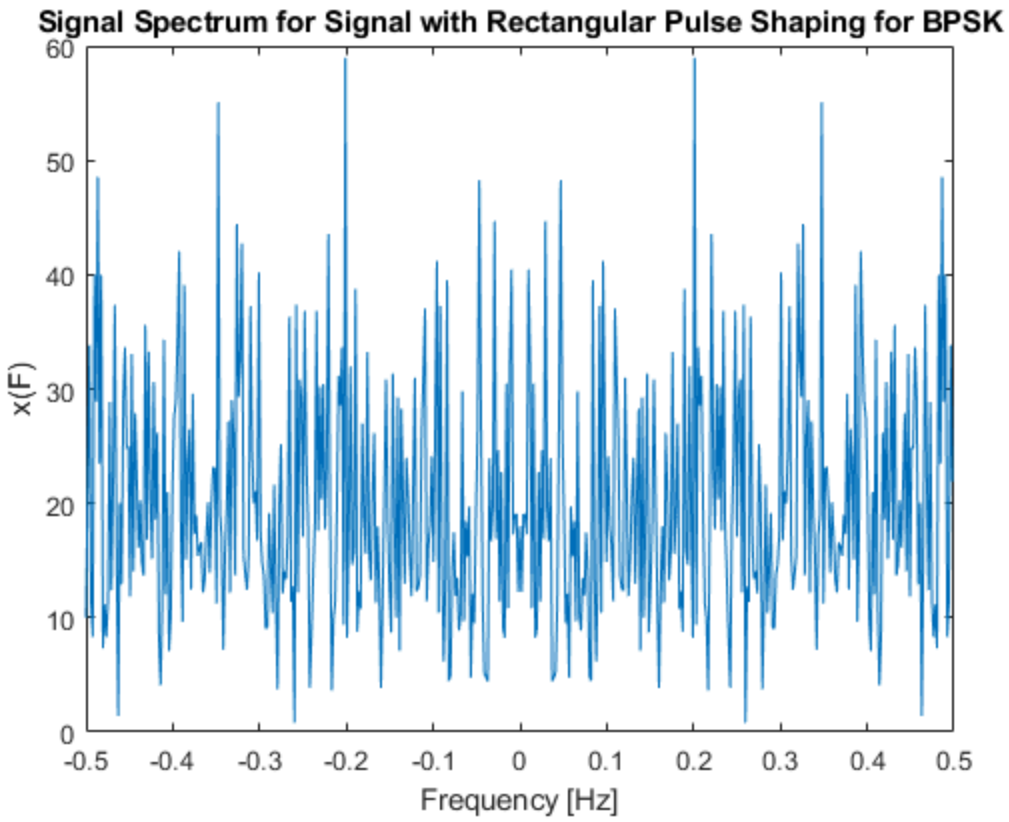
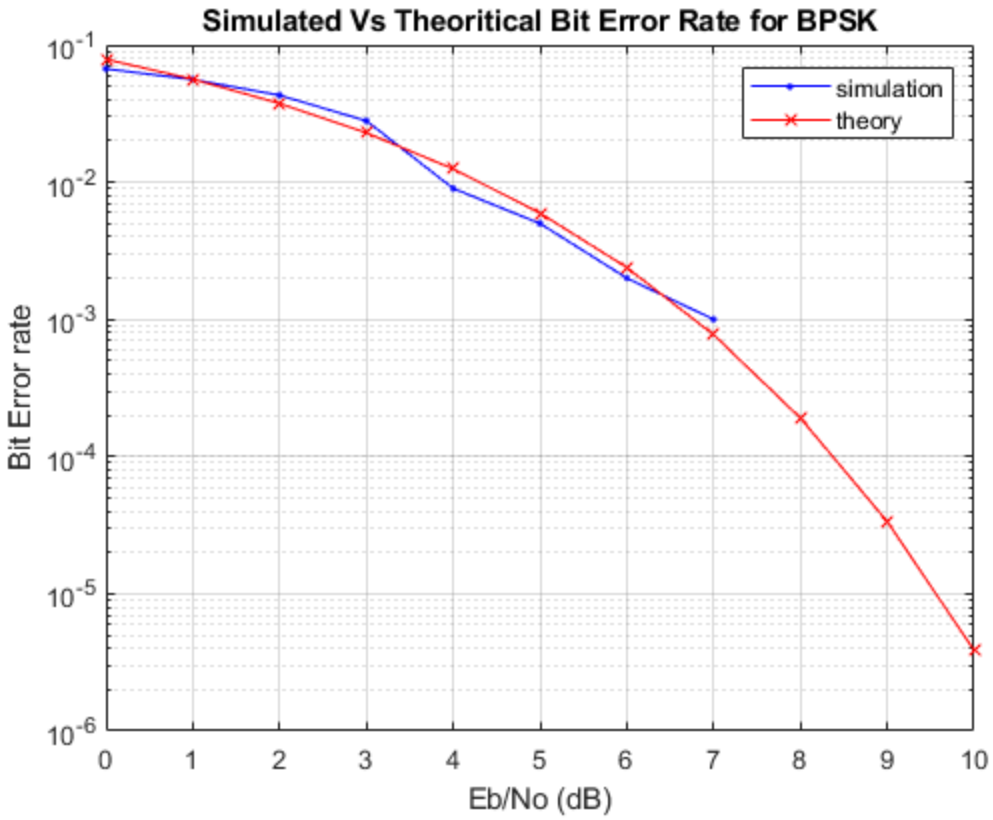
```
%BER Simulation of BPSK modulation
clc;
close all;
clear all;


bits=1000; %number of bit
data=double(randi(2,[1,bits])>1.5);%random bit generation (1 or 0)
ebno=0:10;

BER=zeros(1,length(ebno));
for i=1:length(ebno)
    %---Transmitter---------
    %mapping of bits into symbols
    symb=2.*data-1;
    %----Filter
    psf=ones(1,1);
    M=length(psf);
    % inserting zeros between the bits
    % w.r.t number of coefficients of
    % PSF to pass the bit stream from the PSF
    z=zeros(M-1,bits);
    upsamp=[symb;z];
    upsamp2=reshape(upsamp,1,(M)*bits);
    %Passing the symbols from PSF
    tx_symb=conv(upsamp2,psf);
    %--------CHANNEL-----------
    %Random noise generation and addition to the signal
    ebnos=10.^(ebno(i)/10);
    n_var=1/sqrt(2.*ebnos);
    rx_symb=tx_symb+n_var*randn(1,length(tx_symb));
    %xxxxxxxxxxxxxxxxxxxxxxxxx
    %-------RECEIVER-----------
    rx_match=conv(rx_symb,psf);
    rx=rx_match(M:M:length(rx_match));
    rx=rx(1:1:bits);
    recv_bits=(sign(rx)+1)./2;

    %xxxxxxxxxxxxxxxxxxxxxxxxxx
    %---SIMULATED BIT ERROR RATE----
    errors=find(xor(recv_bits,data));
    errors=size(errors,2);
    BER(i)=errors/bits;
    %xxxxxxxxxxxxxxxxxxxxxxxxxx
end
fs=1;
n_pt=2^9;
tx_spec=fft(tx_symb,n_pt);
f= -fs/2:fs/n_pt:fs/2-fs/n_pt;
figure;
plot(f,abs(fftshift(tx_spec)));
```

```
title('Signal Spectrum for Signal with Rectangular Pulse Shaping for
 BPSK');
xlabel('Frequency [Hz]');
ylabel('x(F)');

figure;
semilogy(ebno,BER,'b.-');
hold on;
thr=0.5*erfc(sqrt(10.^(ebno/10)));
semilogy(ebno,thr,'rx-');
xlabel('Eb/No (dB)');
ylabel('Bit Error rate');
title('Simulated Vs Theoritical Bit Error Rate for BPSK');
legend('simulation','theory');
grid on;
```





Published with MATLAB® R2019b