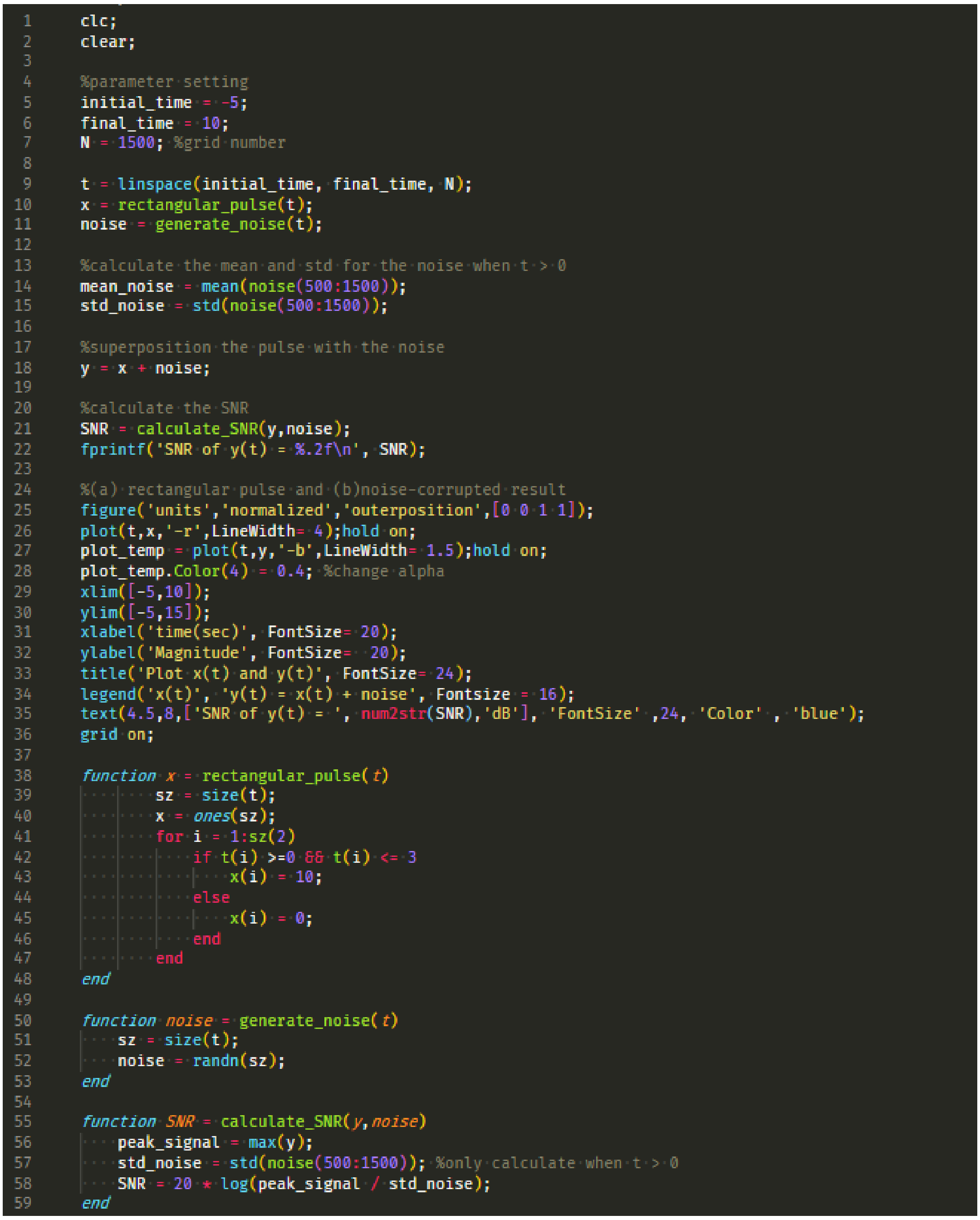
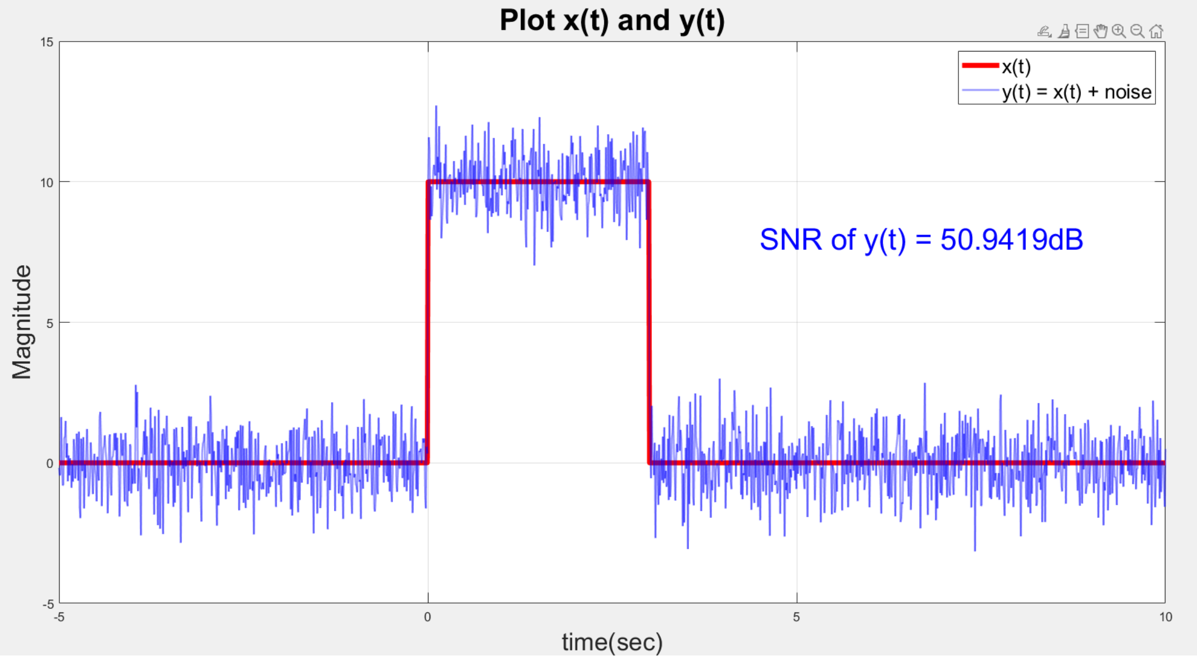


```matlab
clc;
clear;

%parameter setting
initial_time = -5;
final_time = 10;
N = 1500; %grid number

t = linspace(initial_time, final_time, N);
x = rectangular_pulse(t);
noise = generate_noise(t);

%calculate the mean and std for the noise when t > 0
mean_noise = mean(noise(500:1500));
std_noise = std(noise(500:1500));

%superposition the pulse with the noise
y = x + noise;

%calculate the SNR
SNR = calculate_SNR(y,noise);
fprintf('SNR of y(t) = %.2f\n', SNR);

%(a) rectangular pulse and (b)noise-corrupted result
figure('units','normalized','outerposition',[0 0 1 1]);
plot(t,x,'-r',LineWidth= 4);hold on;
plot_temp = plot(t,y,'-b',LineWidth= 1.5);hold on;
plot_temp.Color(4) = 0.4; %change alpha
xlim([-5,10]);
ylim([-5,15]);
xlabel('time(sec)', FontSize= 20);
ylabel('Magnitude', FontSize=  20);
title('Plot x(t) and y(t)', FontSize= 24);
legend('x(t)', 'y(t) = x(t) + noise', Fontsize = 16);
text(4.5,8,['SNR of y(t) = ', num2str(SNR),'dB'], 'FontSize' ,24, 'Color' , 'blue');
grid on;

function x = rectangular_pulse(t)
        sz = size(t);
        x = ones(sz);
        for i = 1:sz(2)
            if t(i) >=0 && t(i) <= 3
                x(i) = 10;
            else
                x(i) = 0;
            end
        end
end

function noise = generate_noise(t)
    sz = size(t);
    noise = randn(sz);
end

function SNR = calculate_SNR(y,noise)
    peak_signal = max(y);
    std_noise = std(noise(500:1500)); %only calculate when t > 0
    SNR = 20 * log(peak_signal / std_noise);
end
```


Plot x(t) and y(t)
x(t)
y(t) = x(t) + noise
SNR of y(t) = 50.9419dB
Magnitude
time(sec)