Lab 6 – Binary Phase Shift Keying (BPSK)

Youssef Samwel

yo800238@ucf.edu

EEL4515 Fundamental of Digital Communications

Prof. Dr. George Atia - Section 0012

3/28/2024

1.0 Experiment Objective

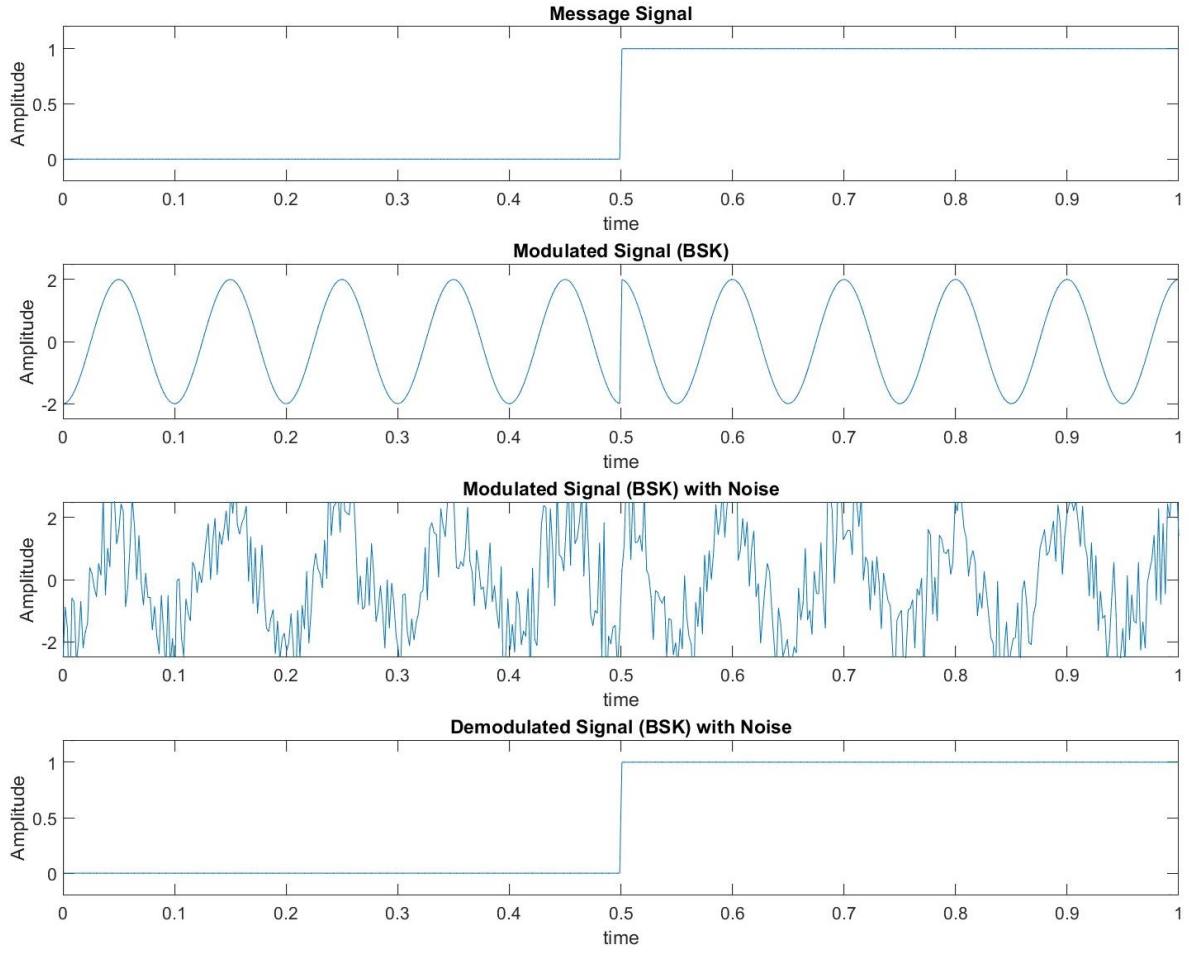
Understand the principles of Binary Phase Shift Keying (BPSK) digital modulation scheme, its error performance through simulation and hardware implementation of BPSK modulation.

2.0 About Laboratory Day and Equipment List

The laboratory session took place on the Thursday section between 9:00am and 11:50am on March 28th, 2024. My lab partner was Isiah. The equipment for the is experiment is listed below,

1. MATLAB
2. Rohde & Schwarz RTM 3034 Oscilloscope
3. Function Generator
4. 2N4392 NMOS
5. LF351N Operational Amplifier

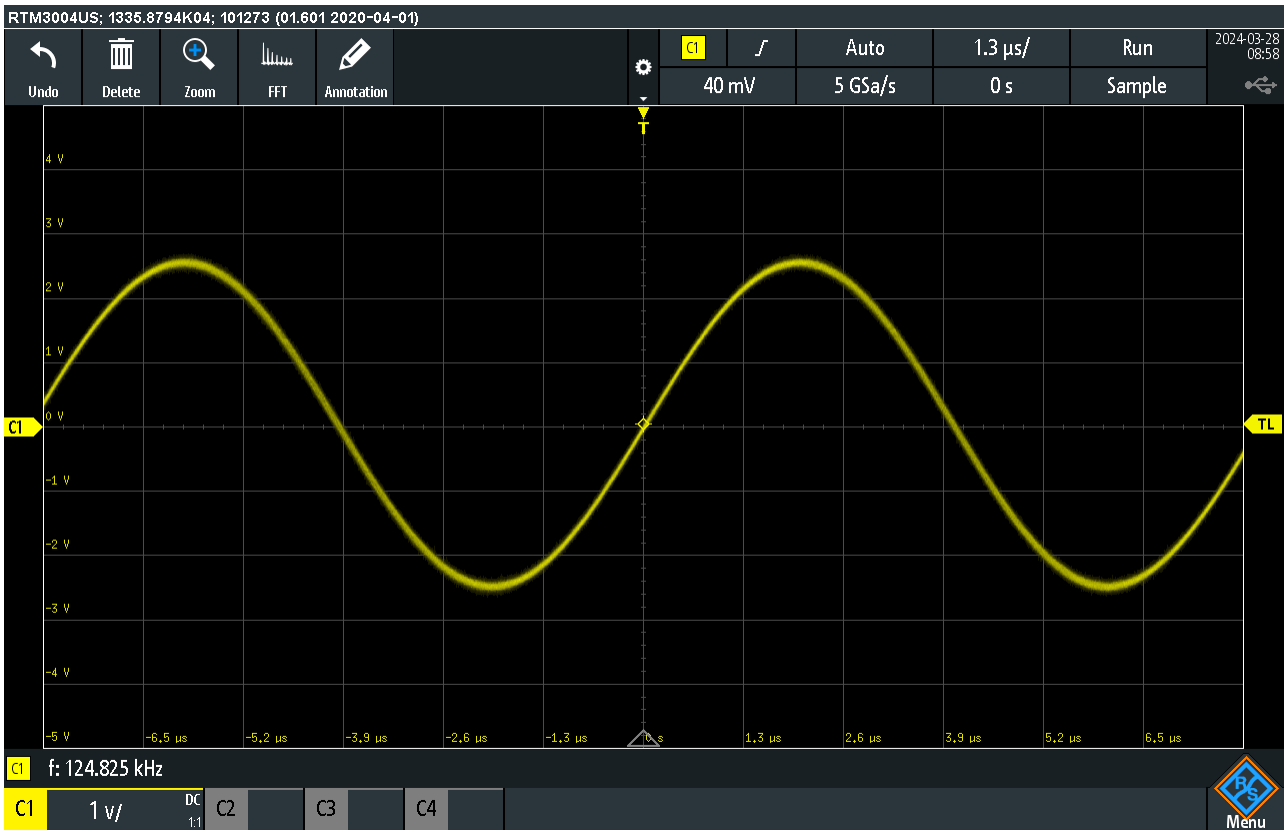
3.0 Simulation



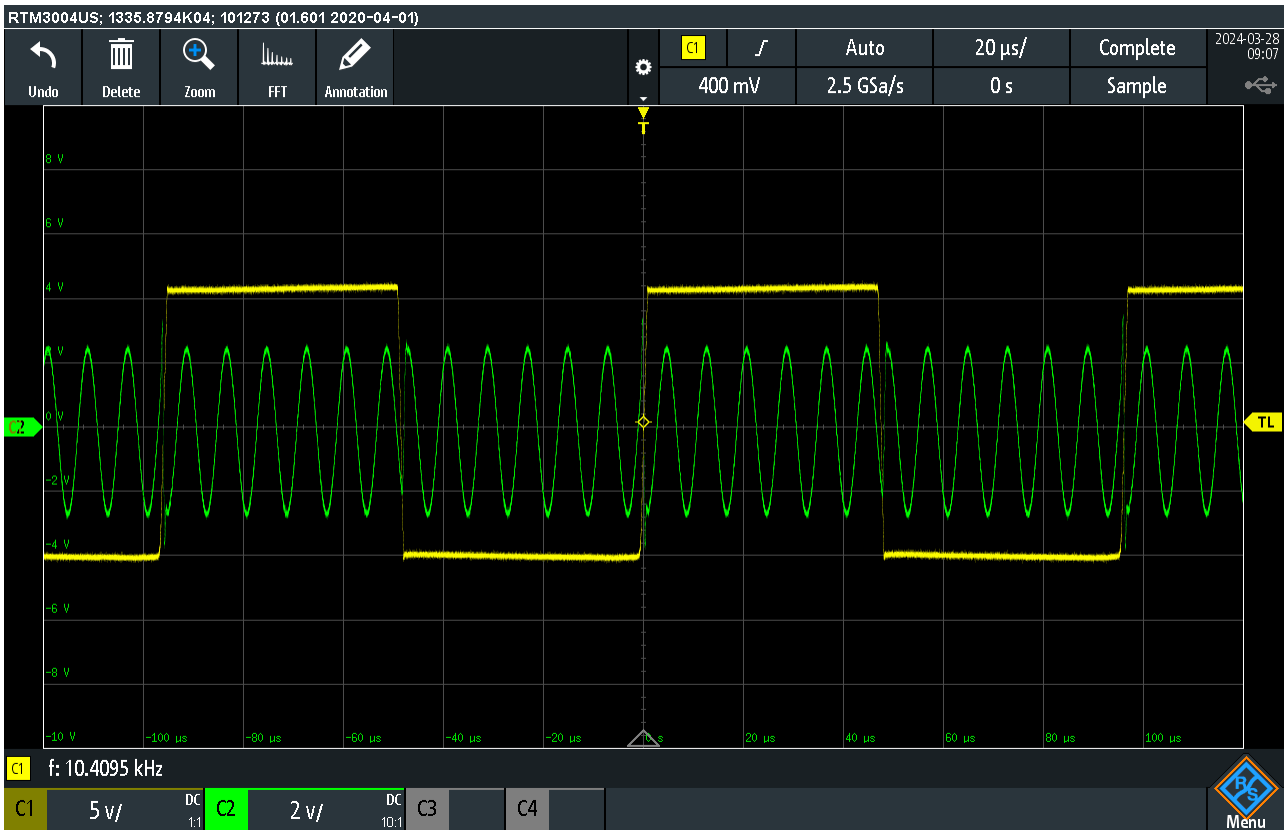
Manchester Encoding Simulation Results

See Section 5.0 for MATLAB code.

4.0 Implementation



Carrier Signal



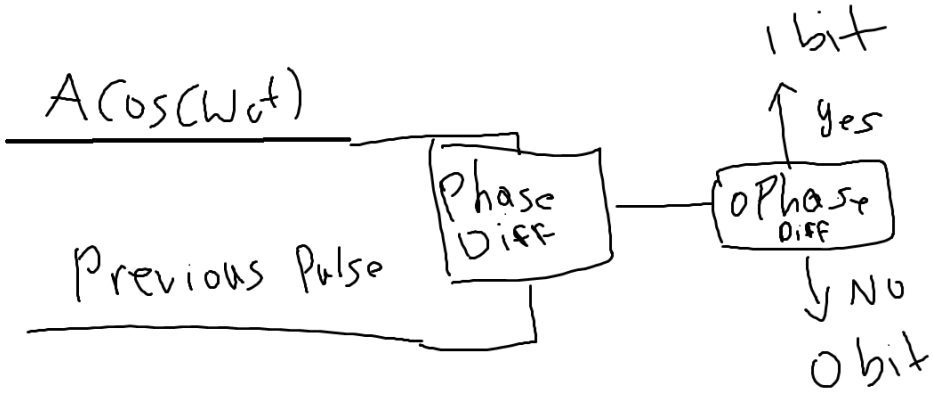
Modulated Signal with Demodulated Bitstream

4.5 Questions and Results

How BPSK modulated signal can be detected? Show the demodulation process for BPSK symbols through block diagrams for the case of

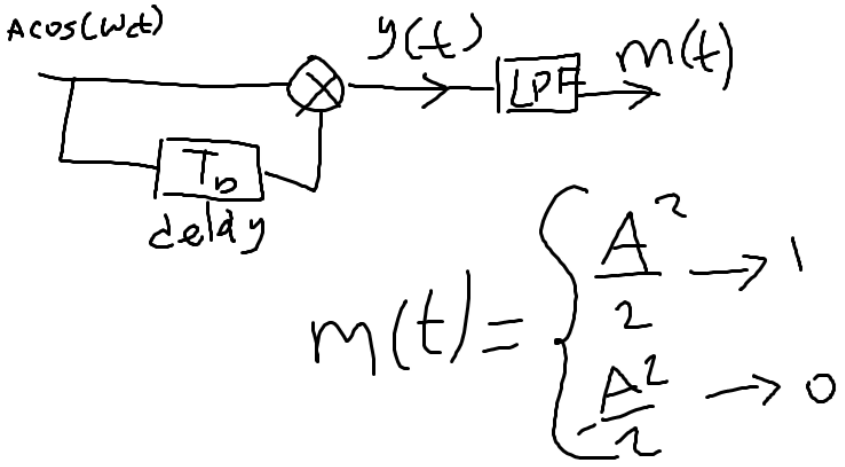
- Coherent detection, and
- Non-coherent detection.

If the previous bit and the current bit have the same polarity, then we demodulate the current bit as 1 otherwise we demodulate as 0.



Coherent Detection

If two consecutive pulses have are identical then we decode “1” otherwise “0”.



Incoherent Detection

5.0 MATLAB Code

Used to generate figure(s)

```
clear all;
close all;
clc;

Tb = 0.5;
```

```
N=500;
Bits = 2;

t = linspace(0, Bits*Tb, N);

last_bit = 1;
for b=1:1:Bits
    len = N / Bits;
    offset = ((b - 1) * len) + 1;
    last_bit = ~last_bit;
    for i=offset:1:min((len+offset), N)
        message(i) = last_bit;
    end
end

subplot(4,1,1);
plot(t, message);
ylim([-0.2, 1.2]);
xlim([0, 1]);
title("Message Signal");
xlabel("time");
ylabel("Amplitude");

%% Generating BSK signal
E = 1;
M = 2;
fc = 10;
phi = sqrt(2)*cos(2*pi*fc*t);
s1 = sqrt(2*E)*phi;
s2 = -sqrt(2*E)*phi;
% noise
N0 = (10^(3/10) * E);
sigma = sqrt(N0/2);

noise_matrix = randn(length(t), 1) * sigma;

modulated_signal = linspace(0, 1, N);
for i=1:1:length(modulated_signal)
    time = t(i);
    value = 0;
    if message(i) == 1
        value = s1(i);
    else
        value = s2(i);
    end
    modulated_signal(i) = value;
end
subplot(4,1,2);
plot(t, modulated_signal);
ylim([-2.5, 2.5]);
xlim([0, 1]);
title("Modulated Signal (BSK)");
xlabel("time");
ylabel("Amplitude");

modulated_signal_with_noise = modulated_signal + transpose(noise_matrix);

subplot(4,1,3);
plot(t, modulated_signal_with_noise);
ylim([-2.5, 2.5]);
xlim([0, 1]);
```

```
title("Modulated Signal (BSK) with Noise");
xlabel("time");
ylabel("Amplitude");

%% Demodulation

demodulated_signal = modulated_signal - modulated_signal;
bit_sum = 0;
time_counter = 0;
time_step = (Bits*Tb)/N;
demodulated_bits = [];
for i=1:1:length(demodulated_signal)
    time = t(i);
    time_counter = time_counter + time_step;
    r_mul_phi = modulated_signal_with_noise(i) * phi(i);

    bit_sum = bit_sum + r_mul_phi;
    %plot(time, bit_sum, '.');
    %hold on;
    if time_counter >= (Tb - 0.001)
        if(bit_sum > 1)
            bit_value = 1;
        else
            bit_value = 0;
        end
        demodulated_bits(length(demodulated_bits) + 1) = bit_value;
        bit_sum = 0;
        time_counter = 0;
    end
end

bit_index = 1;
for i=1:1:length(demodulated_signal)
    time = t(i);
    time_counter = time_counter + time_step;
    demodulated_signal(i) = demodulated_bits(min(bit_index, length(demodulated_bits)));
    if time_counter >= (Tb)
        bit_index = bit_index + 1;
        time_counter = 0;
    end
end

subplot(4,1,4);
plot(t, demodulated_signal);
ylim([-0.2, 1.2]);
xlim([0, 1]);
title("Demodulated Signal (BSK) with Noise");
xlabel("time");
ylabel("Amplitude");
```

6.0 Learned Objectives

- XR-2206 and XR-2211
- BPSK Modulation
- MATLAB Simulation

7.0 Conclusion

In this experiment, we successfully explored Binary Phase Shift Keying (BPSK) modulation, both through simulation in MATLAB and hardware implementation. By analyzing BPSK principles and error performance, we gained a deeper understanding of digital communication fundamentals. Through practical demonstrations and MATLAB simulations, we clarified concepts related to BPSK signal generation, demodulation processes, and detection methods. This experiment not only met its objectives but also provided valuable hands-on experience, laying a strong foundation for future exploration in digital modulation techniques and communication systems.