Text

Description automatically generated with medium confidenceDigital Communication Systems

**Laboratory Report**

Fall 2021

|  |  |
| --- | --- |
| Laboratory Number: | **05** |
| Laboratory Title: | **Baseband Transmission and Detection Pt2** |
| Full Name: | **Robert Louis Bara** |
| TUID: | **915614617** |

**Description:**

This lab picks up from last lab which explores baseband transmission and detection. The lab utilizes waveform encodings by assigning a waveform to a symbol, so the waveform gets transmitted and then decoded to receive the waveform. The symbol may be encoded three different ways,

* Amplitude Shift Keying (ASK)
* Frequency Shift Keying (FSK)
* Phase Shift Keying (PSK)

Detection methods such as the correlation decoder can be used to detect as follows,

where is a detection, is the received signal with respect to time and is the base encoded signal.

The most common way to find the noise impact of the encoding/decoding process, is to generate White Gaussian Noise and transmit it with the transmitted signal. This lab transmits my TUID using a 16bit ASK encoding, then randomly generates White Gaussian Noise to determine the signal to noise ratio and mean absolute error rate. This data will then be compared to last week’s 4bit ASK data.

**Images:**

Diagram

Description automatically generated

Figure 1. Generated Sinusoid transmitting TUID

**Numerical Tables:**

Table 1. 4bit-ASK

|  |  |
| --- | --- |
| 4-ASK | |
| SNR | Mean Absolute Error |
| 20 | 0 |
| 16 | 0 |
| 8 | 0 |
| 4 | 0 |
| 2 | 0 |
| 0 | 0 |
| -2 | 0 |
| -4 | 0 |
| -8 | 0 |
| -16 | 0.004 |
| -20 | 0.087 |

Table 2. 16bit-ASK

|  |  |
| --- | --- |
| 16-ASK | |
| SNR | Mean Absolute Error |
| 20 | 0 |
| 16 | 0 |
| 8 | 0 |
| 4 | 0 |
| 2 | 0 |
| 0 | 0.002 |
| -2 | 0.013 |
| -4 | 0.044 |
| -8 | 0.173 |
| -16 | 0.66 |
| -20 | 1.048 |

Table 3.Percent difference between 4bit and 16bit ASK

|  |  |  |  |
| --- | --- | --- | --- |
| **SNR** | **4bit ASK Mean Absolute Error** | **16bit ASK Mean Absolute Error** | **Percent Difference** |
| **20** | 0 | 0 | 0 |
| **16** | 0 | 0 | 0 |
| **8** | 0 | 0 | 0 |
| **4** | 0 | 0 | 0 |
| **2** | 0 | 0 | 0 |
| **0** | 0 | 0.002 | 0.002 |
| **-2** | 0 | 0.013 | 0.013 |
| **-4** | 0 | 0.044 | 0.044 |
| **-8** | 0 | 0.173 | 0.173 |
| **-16** | 0.004 | 0.66 | 197.5903614 |
| **-20** | 0.087 | 1.048 | 169.339207 |

**Descriptive answers to tasks:**

By increasing the bit size from 4bits to 16bits, the increase for error stemmed much sooner when manipulating the signal to noise ratio (SNR) values. When comparing the percent differences, the error was much higher when SNR was inputted as -16 and -20. Decreasing the SNR is expected to result in a more noise, however it seems that by increasing a wider signal, the risk of getting a noisier signal is higher.

**Code:**

### Section 01

The initial parameters are usually defined at the beginning of the program.

clc; clear;

A = 1; % Signal amplitude

rb = 1000; % Fundamental frequency of signal

Tb = 1 / rb; % Period of signal

fs = 1000 \* rb; % Sampling frequency

Ts = 1 / fs; % Sampling period

Defining Symbol sets and encoding TUID: 915614617

symbol\_set = 0:15; % symbols' indices

symbol\_amplitude = [-8:-1, 1:8] \*A / 16; % symbols' amplitude

msg = [9, 1, 5, 6, 1, 4, 6, 1, 7]; % message: a set of symbols

msg\_amplitude = symbol\_amplitude(symbol\_set(msg+1)+1); % mepping the symbol to the amplitude

figure()

subplot(2, 1, 1); stem(msg); xlabel('time'); ylabel('Symbol Indices'); title('symbols indices');

subplot(2, 1, 2); stem(msg\_amplitude); xlabel('time'); ylabel('Amplitude'); title('symbols amplitude');

subplot(2, 1, 2); stem(msg\_amplitude); xlabel('time'); ylabel('Amplitude'); title('symbols amplitude');

Section 06 is then modified to take in 1000 samples

% generation of a random message with fixed length

rng(0);

Nm = 1000;

symbol\_set = 0:15; % symbols' indices

symbol\_amplitude = [-8:-1, 1:8] \*A / 16; % symbols' amplitude

msg = randi(M-1, [1, Nm]); % message: a set of symbols

figure(); stem(msg);

msg\_amplitude = symbol\_amplitude(symbol\_set(msg+1)+1); % mepping the symbol to the amplitude

SNR is then modified to take in the inputs to the table and compared using the sample code’s computation for error rate.

snr = -20;

noisy\_signal = awgn(signal, snr, 'measured');

figure(); plot(noisy\_signal);