# CE 3006: Course Project

## Aim: To develop a basic digital communication system using MATLAB.

**MATLAB®** is a high-level language and interactive environment for numerical computation, visualization, and programming. Using MATLAB, you can analyze data, develop algorithms, and create models and applications. The language, tools, and built-in math functions enable you to explore multiple approaches and reach a solution faster than with spread sheets or traditional programming languages, such as C/C++ or Java™.

You have to get familiarized with MATLAB before starting the project. A simple introduction for MATLAB and some additional reading materials are given in course project folder.

**The basic implementation of this project consists of three phases.**

- **Phase 1: Data generation**
- **Phase 2: Modulation for communication**
- **Phase 3: Basic error control coding to improve the performance**

The following section gives the details of the project implementation. Consider this as a guideline. You are free to improve/enhance your system implementation for better performance and demonstration. All additional inclusions will have extra weightage and bonus marks during the evaluation of the project.

## Phase 1: Data generation

This section is related to baseband modulation and demodulation. Here the binary data will be generated and will be transmitted through an additive white Gaussian noise channel with different SNR values. The bit error rate performance will be analysed and plotted against various SNR values.

**Basic steps for the first phase:**

- Assume the number of bits for transmission is 1024 (means, N= 1024).
- Generate random binary digits (0 or 1)
- Convert binary digits to ±1 (means 1 to +1 and 0 to -1). This is your data for transmission.
- Generate equal number of noise samples.
- The generated noise should have normal distribution with zero mean and unit variance (use the function randn in MATLAB).
- Change the noise variance with respect to SNR (signal to noise ratio) value.
- For that, fix the SNR value, (For example, let SNR = 10 dB)
- Use SNR value to generate noise variance.
- SNR (in dB) = $10\log_{10}$ (S/N) where S is the Signal power (or variance) and N is the Noise power (or variance)
- Assume signal (the input data) has unit power. That is, S=1. Obtain the noise variance (=N) from the previous relation and use it together with the noise samples to generate the required noise.
- Add noise samples with transmitted data. This is assumed as your received signal.
- Consider a threshold logic at the receiver.
- Fix the threshold value as 0 (the transmitted data is +1 and -1, and 0 is the mid value)
- If the received signal is above or equal to the threshold level, take it as 1
- If the received signal is below the threshold value, take it as 0.
- Compute the bit error rate during transmission.
- Compare the output values from the threshold logice with the input binary digits (1 or 0 format)
- Compute the bit error rate using the relation Bit error rate = (number of errors during transmission)/(total number of bits for transmission)
- Repeat the steps for different SNRs
- Repeat the steps from noise addition to bit error rate computation
- Do the simulation for the same data set with different SNRs
- Consider different SNR values from 0 dB to 50 dB (in multiples of 5 dB).
- Plot your result in a graph.
- X axis should be SNR values and Y axis should be bit error rate.

- Show all your results in the final report.

## Phase 2: Modulation for communication

This section is related to band-pass modulation and demodulation. You are asked to implement different band-pass modulation methods in the context. Simple modulation methods such as On-Off Keying and Binary Phase Shift Keying will give a basic start for this. All are encouraged to try out Binary Frequency Shift Keying and other M-ary modulation schemes. Any additional work in this direction will be rewarded with more bonus points during evaluation.

### Steps for the phase 2 (Baseband to band-pass conversion)

- Apply different modulation schemes to the generated baseband data.
- Consider On-Off Keying (OOK) and Binary Phase Shift Keying modulation (BPSK) schemes. Check the lecture slides on band-pass modulation and demodulation schemes for more details.
- For OOK the data signals can be considered 1 or 0 format
- For BPSK it has to be +1 or -1 format
  - Modulate the data samples with carrier signal ($\cos(2\pi f t)$) where $f$ is the carrier frequency.
- Let the carrier frequency be 10 KHz.
- Assume the carrier signal is 16 times oversampled (means one cycle contains 16 samples).
- Let us consider the baseband data rate as 1 kbps.
  - Generate the modulated signal (for both OOK and BPSK separately)
  - Add additive white noise with required SNR value to both the modulated signals
  - The received signal is modulated signal added with the noise.
  - Perform demodulation at the receiver.
- It can be done by multiplying twice the carrier signal (i.e., $2*\cos(2\pi f t)$)
  - Pass the resulting signal through a low pass filter.
- Use the function '*butter*' in MATLAB to generate the filter coefficients
- Assume a $6^{th}$ order filter with cut-off frequency 0.2 in the function.
- Use the filter coefficients in function '*filtfilt*' to do the filtering of the demodulated signal
  - Use the decision threshold logic for decoding of received signals
- Use decision logic after sampling at bit intervals (take the midpoint of the bits for decision logic).
  - Calculate the bit error rate performance.
  - Repeat the experiments for different SNR values as in phase 1 and plot the result using '*semilogy'* function.
  - In addition, plot the signals at different stages (data waveform, modulated signal, received signal, demodulated signal and decoded signal) for a selected SNR value (say, at 5 dB)
- Provide these plots in the final report

**Phase 3: Basic error control coding to improve the performance**

The performance of the proposed communication system can be improved further using error control codes. It will be experimented in this phase.

- You can use any codes available in MATLAB, or you can write your own channel codes to show the performance improvement of the system. The code design, implementation and its performance demonstration will be considered during evaluation.

- Include the resulting bit error rate performance in the final report.

Groups can work on this project for the addition of more features of a communication systems and integrate those with the basic implementation. All such modifications are counted for evaluation.

**The final project deadline is Friday of Week 12. You should submit your final report by the Friday of week 12. Your project (together with the report) will be evaluated during Week 13.**

Should you have any further queries on the project, please feel free to contact me.

Enjoy the project.