# ELEG5663 Project 5
# Linear Block Code with Syndrome Decoding

**I.  Objectives**
   1. Understand the concepts of linear block code.
   2. Understand the procedure for syndrome decoding.
   3. Learn to simulate communication system with linear block code and syndrome decoding.

**II.  Theories**

An $(n, k)$ linear block code can be defined by a size $(k$ x $n)$ generation matrix $\mathbf{G}$ , or equivalently, and $n$ x $(n\text{-}k)$ parity check matrix $\mathbf{H}$. For a size-$(1$ x $k)$ information vector $\mathbf{x}$ , the size $(1$ x $n)$ codeword can be calculated as

$$\mathbf{c} = \mathbf{x} \cdot \mathbf{G}$$

The codeword is modulated and transmitted.

At the receiver, the received signal is demodulated. The output of the demodulator is denoted as $\mathbf{y}$. The syndrome of the received codeword is

$$\mathbf{s} = \mathbf{y} \cdot \mathbf{H} = \mathbf{e} \cdot \mathbf{H} ,$$

where $\mathbf{e}$ is a size $(1$ x n) error vector. A syndrome – error vector table can be built up for syndrome decoding. The table has $2^{n-k}$ entries, with each entry corresponding to one unique syndrome. Each syndrome entry is associated with an error vector. Syndrome decoding can be performed by finding the error vector based on the syndrome of the received codeword.

The block diagram of the communication system that is going to be simulated is shown as follows.
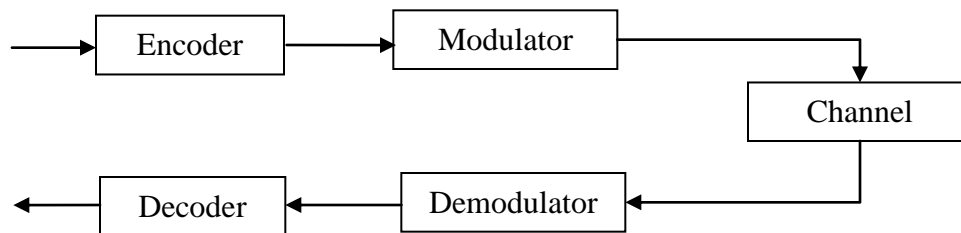


**III.  Procedures**

   1. Write a function, block_encode.m, to perform encoding of a (7, 4) linear block code. The input of the function is a length 4 information vector containing binary '0's and '1's. The output of the function is the corresponding length 7 codeword. The generation matrix of the linear block code is

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

2. Write a function, block_decode.m, to perform syndrome decoding of the (7, 4) linear block code. The input of the function is a length 7 codeword vector containing '1's and '0's, and the output of the function is the decoded length information vector.
3. What is the error correction capability of the code?
4. Test your encoder and decoder by passing one of the 16 possible information vectors through the encoder, and then pass the output of the encoder to the decoder. Compare the original binary sequence and the output of the decoder to make sure they are working properly.

5. In the simulation, the symbols are transmitted in slots (bursts). Each slot has 7 modulated symbols. The modulation scheme is BPSK. Use 200,000 slots for each SNR point for the BER calculation.

6. Use the Matlab code from Project 2 as the framework of this project. Put the linear block code encoder before the BPSK modulator; put the syndrome decoder after BPSK demodulator.
7. Since linear block code is employed, the relationship between noise variance and Eb/N0 can be expressed by the following equation

$$E_b / N_0 = \frac{E_s / r}{\text{noise\_vari} \quad \text{ance} \times \log_2 M}$$

where $E_s$ the energy of one symbol, $r$ is the coding rate, and $M$ is the modulation constellation size. What should be the noise_amplitude for $E_s = 1$, $M = 8$, $r = 4/7$, and $E_b / N_0 = 0 dB$ ?
8. The generation of information bits, fading, and noise is the same as project 2.

9. Plot the uncoded BER curve and the BER curve of the coded system in the same figure at $E_b / N_0 = [1:9]$ dB. Compare the results. Note the calculation of $E_b / N_0$ for the coded and uncoded systems are different.