Homework 1

1. A PCM system has 16 levels in its Quantizer with step size of Δ Volts, and each sample is represented by 4 bits. Then, bits are pulse-shaped using NRZ and sent through channel. Assume that the channel causes burst errors of 2 bits at a time. In other words, when an error happens in channel, it causes errors in 2 successive bits.
   a. Calculate the expected noise power on uncoded sample when one channel error occurs in random location in time.
   b. What is the maximum error in Volts that can occur on a sample?
   c. Suggest a technique (a new mapping or bit reordering) to reduce the maximum possible error on a sample, and find the reduced max error in Volts.
   d. Repeat (a) with the suggested technique and compare it with the original result. Do you see an improvement on expected noise power?

2. Show that Raised Cosine pulse can be written using only the sinc functions as:

$$p_{\mathrm{RC}}(t) = \frac{\pi}{4} \cdot \mathrm{sinc}\left(\frac{t}{T}\right) \cdot \left[\mathrm{sinc}\left(\alpha \frac{t}{T} - \frac{1}{2}\right) + \mathrm{sinc}\left(\alpha \frac{t}{T} + \frac{1}{2}\right)\right]$$

   Hint: Express $P_{\mathrm{RC}}\left(f\right) = \mathscr{F}\{p_{\mathrm{RC}}(t)\}$ as convolution of a rectangular function with a windowed (time-limited) cosine function of period $2\alpha/T$.

3. Design a 2-PAM (NRZ) system in MATLAB using rectangular pulse in time at transmitter by following steps below:
   a. Generate 1000 random bits (Binary vector of size 1000)
   b. Map those bits to 2-PAM symbols, a_k
   c. Generate a vector of length 5000 representing $m(t) = \sum_k a_k \delta(t - kT_{sym})$ by padding 4 zeros after every a_k (e.g., $\delta(t - T_{sym}) \rightarrow$ [0 0 0 0 1 0 0 0 0])
   d. Generate a vector representing $x_c(t) = m(t) * g_T(t)$ by using a rectangular pulse for transmitter filter. $g_T(t) \rightarrow$ [1 1 1 1 1]. (Use conv() function in MATLAB)
   e. Generate noisy received vector per $y(t) = x_c(t) + n_W(t)$ where $\sigma_{n_W}^2 = 1$. Use randn() function to generate the noise vector.
   f. Directly sample $y(t)$ vector by factor of 5 (take every 5th sample in vector) and get 1000 samples back
   g. Decode 1000 bits from noisy symbols with an appropriate decision rule for NRZ signaling, and calculate the BER (# bits in error / # total bits)
   h. Now, apply an appropriate matched filter to $y(t)$ before sampling, then repeat (f) and (g). Compare the BER with the one found in (g).

```
N = 10000; % number of bits

bits  = randn(N,1) > 0; % randomly generated vector of bits

a_k = …
```