CAIRO UNIVERSITY
FACULTY OF ENGINEERING

ECTRONICS AND COMMUNICATIONS DEPT

3RD YEAR

Digital Communication s

Project # 1

Important notes:

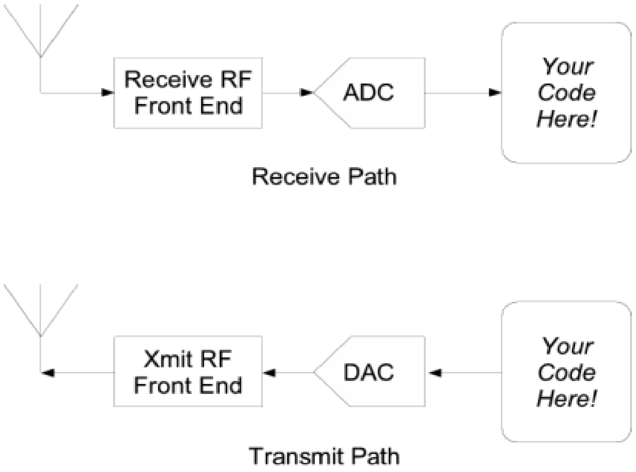
Groups of students with a maximum of 5 students per group

- The report should start with code for each line code then the results for that line code.
- All figures should be correctly labeled.
- The cheater and cheatee of any part of the project will get 0 marks.
- The project is due 6/4/2023 midnight.

If any of the instructions above is not followed, points will be deducted.

Project details

Software radio is the technique used to get the programmer's code as close to the antenna as possible. It turns radio hardware problems into software problems. The fundamental characteristic of software radio is that software defines the transmitted waveforms, and software demodulates the received waveforms. This is in contrast to most radios in which the processing is done with either analog circuitry or analog circuitry combined with digital chips**.**



Your code at the transmitter defines the the modulation technique, the coding used, … etc. The software can also be used to send data through cables, simply remove the antenna and use line codes. The software converts the files into a bit stream, makes framing, coding and then line coding. To know how this can happen, consider the following simplified example:

If you want to transmit binary data 0 1 0 0 1 1 0 1 using polar NRZ signaling, with pulse width=70 ms, and the DAC is activated every 10 ms, the software needs to feed the DAC with new data every 10ms. For polar NRZ , logic '1' is transmitted as voltage level +A, logic '0' is transmitted as voltage level –A. The code in MATLAB that may help in building the array that will be transmitted is:

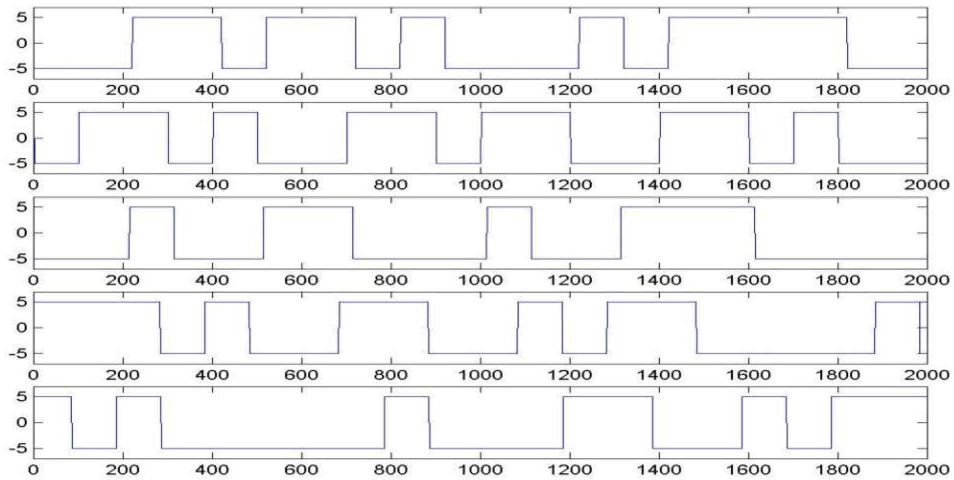
```
A=4;
Data=[0 1 0 0 0 1 1 0 1];
Tx=((2*Data)-1)*A; % maping for 0 to be –A, 1 to be A
% the above array contains 9 elements, corresponding to the levels that will be
%transmitted in every pulse
Tx2=repmat(Tx,7,1);
Tx_out=reshape(Tx2,size(Tx2,1)* size(Tx2,2),1);
```

The output of the last instruction is:

```
Tx_out= -4 -4    -4      -4      -4      -4      -4      4       4       4       4       4
   4      4      -4      -4      -4              -4      -4      -4      -4      -4      -4
  -4     -4      -4      -4      -4      -4      -4      -4      -4      -4      -4      -4
   4      4       4       4       4       4       4       4       4       4       4       4
   4     -4      -4      -4      -4      -4      -4      -4       4       4       4       4
   4      4       4
```

The first -4 was repeated 7 times, this corresponds to activating the DAC for 70 ms by certain voltage, then +4 for 7 times corresponding to activating the DAC for 70 ms with the other voltage, and so on.

The output of the DAC can be considered as a random process, due to the randomness of the transmitted bits and exact timing of the beginning of transmission. Consider the following waveforms:

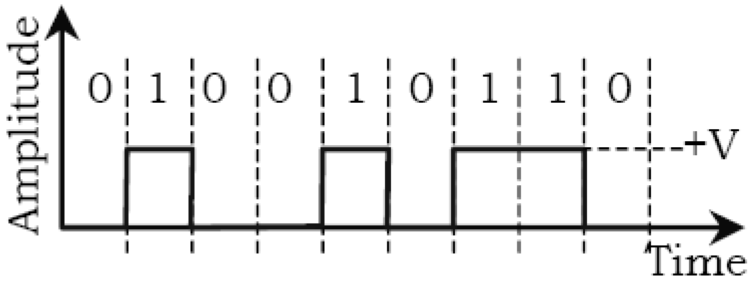


All the waveforms were generated from a Polar NRZ transmitter, so they constitute an **ensemble**. The randomness is in the binary data transmitted and in the initial time shift, so each waveform, or realization, should start from a random initial start It is required to generate an ensemble that consists of 500 waveforms, each containing 100 bits, for the line codes mentioned below. It is required to compute:

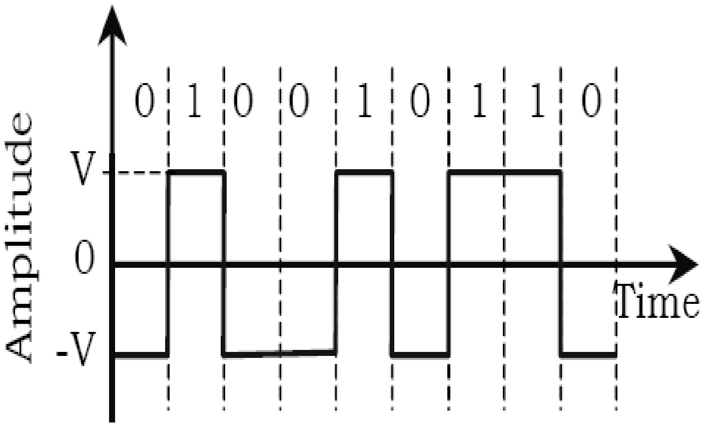
1- The statistical mean.
2- Is the random process stationary?
3- Determine the ensemble autocorrelation function $R_x(\tau)$.
4- The time mean and autocorrelation function for one waveform.
5- Is the random process Ergodic?
6- What is the bandwidth of the transmitted signal.

The line codes required to be investigated are:

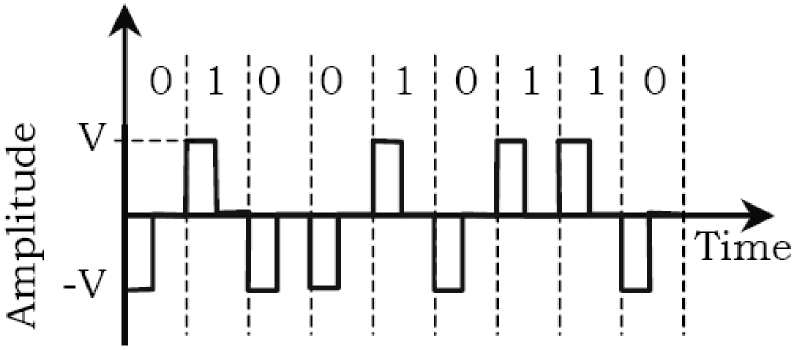
1- Unipolar Signaling



2- Polar Non Return to Zero



3- Return to Zero


Amplitude
0
1
0
0
1
0
1
1
0
V
-V
Time