Alexandria University
Faculty of Engineering
Computer and Systems Engineering Dept.

CS 232 Digital Systems II
Second Year
Spring 2019
Assigned: **Friday, May 3rd**
Due: **Wednesday, May 15th**

Lab 4

Uniquely Decipherable Codes

Objectives

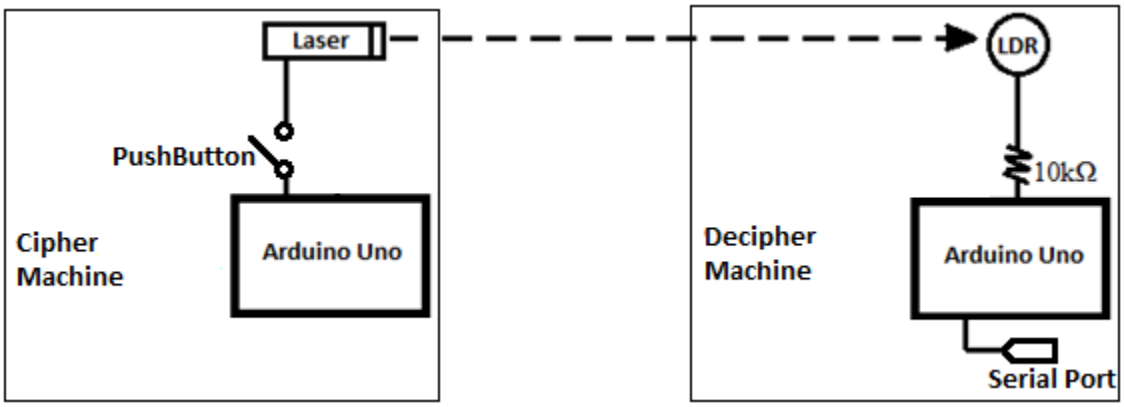
The objective of this lab is twofold: to practice some of the basic issues in coding theory and to demonstrate the applicability of ciphering techniques to the area of information transmission and codes. In this lab, you are required to design two circuits to transmit, receive, chipper and decipher the data.

Circuit Diagram

In this lab you will need the following components

- 2 x Arduino Uno kits
- 1 x Light Dependent Resistor (LDR)
- 1 x 10kΩ Resistors
- 1 x 5Volt Laser source
- 1 x Push Buttons (or Switches)

The layout of the circuits will be as follows



How it works?

A. Data Transmission

Similar to the previous lab the two circuits will use the light as a way to represent the transmitted data. The output of one of the circuits will be emitted through the 5Volt Laser, and on the other hand, the other circuit input will be using LDR. The LDR is a special type of resistor that allows higher voltages to pass through it (low resistance) whenever there is a high intensity of light, and passes a low voltage (high resistance) whenever it is dark. We can take advantage of this LDR property and use it as an input to our circuit.

B. Cipher/Decipher

Let the symbols {A, B, C, D, E} denote a finite source alphabet, and let a code alphabet L = {0, 1}. A concatenation of a finite number of code symbols is referred to as a ***code word***. A code consists of a finite number of distinct code words of finite length, each representing a source symbol. For example, we can use the following assignments

Symbol	Code Word	ASCII
A	11	65
B	011	66
C	001	67
D	01	68
E	00	69

A coded message is constructed by concatenating code words without spacing or any other punctuation. For example, the representation of the string "CABEBDEA" in this coding is

0011101100011010011

Hint: *pick suitable range for High and Low according to the readings of the LDR.*

C. Input/Output

The input for your machine will be through the push-buttons, and the output (i.e., the deciphered message) can be printed to the Arduino serial port using ***Serial.println(value)*** with the ASCII code of the detected symbols. So for example the representation of the string "CABEBDEA" that will be printed to the serial port will be: 67, 65, 66, 69, 66, 68, 69, 65

Hint: *You will need to synchronize the two machines together using suitable delay that acts as a clock. Use the delay() function to handle this.*

Problem Statement

1. Design a Ciphering/Deciphering machine and try it out to send & receive messages (e.g., try out the given example above, and any other example with 10 symbols)
2. Add an extra symbol to represent the "SPACE" to your code, and make sure it is still uniquely decipherable
3. Change the code assignment of the symbol C to be "101". Is your new implementation valid? If not find a counter example.

Watch out the usage of the resistors with the LDR otherwise you may burn them. To avoid any problems, check the detailed guide in the References section.

Grading Policies

- You should work in groups of 4 students.
- You are required to submit a video recording showing your experiment (checking both the valid state machine and the invalid state machine), and the code for all the circuits.
- The penalty of late submission is 20% of delivery grades for the first week and 50% afterwards.
- Your credits will be significantly affected by the cost of the final implementation of your design and code.
- Plagiarizing is not acceptable. Sharing code fragments between groups is prohibited and all the groups that are engaged in this action will be severely penalized. Not delivering the assignment will be much better than committing this offence.

References:

- https://learn.adafruit.com/adafruit-arduino-lesson-2-leds?view=all
- https://maker.pro/arduino/tutorial/how-to-use-an-ldr-sensor-with-arduino
- https://www.arduino.cc/en/tutorial/pushbutton