

*Polytechnic University of Puerto Rico*  
*Electrical & Computer Engineering and Computer Sciences Department*



*COE 4341 Microcomputer Interfacing Laboratory*

## **Laboratory No 4: Using The LCD Display**

ID	LAST NAME	FIRST NAME
130541	Sierra Bello	Gabriel
109648	Salgado Rivera	Daniel E.
103733	Meléndez Nieves	Josué
116762	Madera Torres	Elian F.

SUPERVISOR SIGNATURE \_\_\_\_\_

DATE: **April 4, 2022**

**April 20, 2022**

## Table of Contents

<b><i>Introduction</i></b> .....	<b>3</b>
<b><i>Objective</i></b> .....	<b>4</b>
<b><i>Methods</i></b> .....	<b>5</b>
<b>Procedure C Language Program</b> .....	<b>5</b>
<b>Source Code for the C Language Program</b> .....	<b>6</b>
<b><i>Results</i></b> .....	<b>12</b>
<b><i>Conclusion</i></b> .....	<b>13</b>

## **Introduction**

In this laboratory we will be working with an LCD display connected to the Arduino Mega 2560 and the previous laboratory exercise. The LCD display that will be used in this exercise is a monochrome which contains 2 lines of 16 characters that uses a parallel interface. After connecting and initializing the display we will send messages to indicate the pedestrian when they can cross and when they can't.

## **Objective**

The objective of this laboratory is to connect an LCD display to the Arduino Arduino Mega 2560 and print “Pase” when the green LED turns on and “No pase” when the red LED turns on for the pedestrians of each side of the road.

## **Methods**

### **Procedure C Language Program**

In this experiment, we had to expand on the traffic light system that was implemented last time. Once our traffic light system worked, we added 7 segment displays to portray a countdown of the pedestrian light working and a LCD display that should display “Pase!” and “No pase!” based on the pedestrian light. We had to receive an input from a push button and display a countdown on the seven-segment that indicates how long the pedestrian light would be on. Then it would simply continue its traffic light loop until interrupted again by the push of one of the buttons. When this interruption occurs, the process will start once again.

## Source Code for the C Language Program

### *Semaforo con LCD:*

```
#define cross1 25
#define cross2 29
#define red1 22
#define red2 26
#define yellow1 23
#define yellow2 27
#define green1 24
#define green2 28
#define button1 2
#define button2 3

#include <LiquidCrystal.h>

LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

int start1 = 22;
int start2 = 26;

int i = 0;
int Flag1;
int Flag2;
int Light_State;
int current;
int back;
char
SevenSegL[16]={0b00001001,0b00000001,0b00011111,0b01000001,0b01001001,0b1001100
1, 0b00001101, 0b00100101, 0b10011111, 0b00000011};
char
SevenSegR[16]={0b11110110,0b11111110,0b11100000,0b10111110,0b10110110,0b0110011
0, 0b11110010, 0b11011010, 0b01100000, 0b11111100};
void setup() {
  lcd.begin(16, 2);
  pinMode(cross1, OUTPUT);
  pinMode(cross2, OUTPUT);
  pinMode(red1, OUTPUT);
  pinMode(red2, OUTPUT);
  pinMode(yellow1, OUTPUT);
  pinMode(yellow2, OUTPUT);
  pinMode(green1, OUTPUT);
  pinMode(green2, OUTPUT);
  pinMode(button1, INPUT);
```

```

pinMode(button2, INPUT);

DDRC = 0b11111111;
DDRL = 0b11111111;

attachInterrupt (0,SetFlag1,CHANGE);
attachInterrupt (1,SetFlag2,CHANGE);

Serial.begin (9600);
Flag1 = 0;
Flag2 = 0;
back = 0;

lcd.setCursor(0,1);
lcd.print("No pase!");
lcd.setCursor(0,0);
lcd.print("No pase!");

}

void loop() {

LOOP1:
//Green state 1
digitalWrite(green2, LOW);
digitalWrite(cross2, LOW);
digitalWrite(red2, HIGH);
greenState(start1);
waitClick(1, 3000, start1);
if (back == 1){
    back = 0;
    goto LOOP2;
}
else if (back ==2){
    back = 0;
    goto LOOP1;
}
//Yellow state 1
yellowState(start1);
waitClick(2, 2000, start1);
if (back == 1){
    back = 0;
    goto LOOP2;
}
else if (back ==2){
    back = 0;
    goto LOOP1;
}
}

```

```

}

//Red State 1
redState(start1);
waitClick(3, 3000, start1);
if (back == 1){
    back = 0;
    goto LOOP2;
}
else if (back ==2){
    back = 0;
    goto LOOP1;
}

LOOP2:
//Green state 2
greenState(start2);
waitClick(1, 3000, start2);
if (back == 1){
    back = 0;
    goto LOOP2;
}
else if (back ==2){
    back = 0;
    goto LOOP1;
}

//Yellow state 2
yellowState(start2);
waitClick(2, 2000, start2);
if (back == 1){
    back = 0;
    goto LOOP2;
}
else if (back ==2){
    back = 0;
    goto LOOP1;
}

//Red State 2
redState(start2);
waitClick(3, 3000, start2);
if (back == 1){
    back = 0;
    goto LOOP2;
}
else if (back ==2){
    back = 0;

```



```

    goto LOOP1;
  }
}

void SetFlag1 (){
  Flag1 = 1;
}

void SetFlag2 (){
  Flag2 = 1;
}

void greenState(int pin){
  digitalWrite(pin,LOW);
  digitalWrite(pin +1,LOW);
  digitalWrite(pin +2,HIGH);
  digitalWrite(pin +3,LOW);
}

void yellowState(int pin){
  digitalWrite(pin +2,LOW);
  digitalWrite(pin + 1,HIGH);
}

void redState(int pin){
  digitalWrite(pin + 1,LOW);
  digitalWrite(pin,HIGH);
}

void waitClick(int light, int onTime, int pin){
  for (i=0;i<=(onTime/1000);i++)
    {delay(1000);
    if(Flag1==1 and pin ==22) {
      Light_State =light;
      Pedestrian(pin);
      back = 2;
    }
    else if(Flag1==1 and pin !=22) {
    if(light != 3){
      yellowState(26);
      delay(1000);
    }
    redState(26);
    greenState(22);
    Light_State =1;
    Pedestrian(22);
    back = 2;
    }
}
}

```

```

else if(Flag2==1 and pin ==26) {
  Light_State =light;
  Pedestrian(pin);
  back = 1;
}
else if(Flag2==1 and pin !=26) {
  if(light != 3){
    yellowState(22);
    delay(1000);
  }
  redState(22);
  greenState(26);
  Light_State =1;
  Pedestrian(26);
  back = 1;
}
}
}

```

```

void Pedestrian(int pin){
  switch (Light_State){
    case 1: {
      digitalWrite(pin + 2,HIGH);
      digitalWrite(pin + 3,HIGH);
      delay(3000);
      break;
    }
    case 2:{
      delay (500);
      delay(1000);
      digitalWrite(pin + 1,LOW);
      digitalWrite(pin + 2,HIGH);
      digitalWrite(pin + 3,HIGH);
      delay(3000);
      break;
    }
    case 3:{
      digitalWrite(pin,LOW);
      digitalWrite(pin + 1,HIGH);
      delay(2000);
      digitalWrite(pin + 1,LOW);
      digitalWrite(pin + 2,HIGH);
      digitalWrite(pin + 3,HIGH);
      delay(3000);
      break;
    }
  }
}

```

```

default: {};

}

for(i = 0; i < 10; i++){
  if(pin == 22){
    lcd.setCursor(0,0);
    lcd.print("Pase! ");
    PORTC =~SevenSegL[i];
    digitalWrite(pin + 3,LOW);
    delay(500);
    digitalWrite(pin + 3,HIGH);
    delay(500);

  }
  else if (pin == 26){
    lcd.setCursor(0,1);
    lcd.print("Pase! ");
    PORTL =~SevenSegR[i];
    digitalWrite(pin + 3,LOW);
    delay(500);
    digitalWrite(pin + 3,HIGH);
    delay(500);

  }
}

lcd.setCursor(0,1);
lcd.print("No pase!");
lcd.setCursor(0,0);
lcd.print("No pase!");
Flag1 = 0;
Flag2 = 0;
}

```

## **Results**

[https://youtu.be/okfkFFr\\_MPE](https://youtu.be/okfkFFr_MPE)

## **Conclusion**

In conclusion, during this experiment we discovered how to work with an LCD display connected to the Arduino Mega 2560. Learned the use of a monochrome which contains 2 lines of 16 characters that uses a parallel interface. Then, we connected and initialize the display so it can display the message to the pedestrian. The experiment was a resounding success.