**INTRODUCCIÓN A LA PROGRAMACIÓN DE MICROCONTROLADORES CON TECNOLOGÍAS LIBRES: TRABAJO FINAL**

**ESTACIONAMIENTO CON BARRERAS DE ENTRADA Y SALIDA**



**ALUMNO: *MAXIMILIANO MARITANO***

# MATERIALES UTILIZADOS

|  |  |
| --- | --- |
| **1x PLACA ARDUINO UNO** |  |
| **2x SENSORES DE PROXIMIDAD ULTRASONICOS** |  |
| **2x MICROSERVOMOTOR** |  |
| **1x PANTALLA LCD 16x2** |  |
| **1x POTENCIOMETRO 10k** |  |
| **3x RESISTENCIAS 220ohm** |  |
| **2x LEDS** |  |
| **1x BUZZER/ALARMA** |  |
| **CABLES MACHO-HEMBRA** |  |

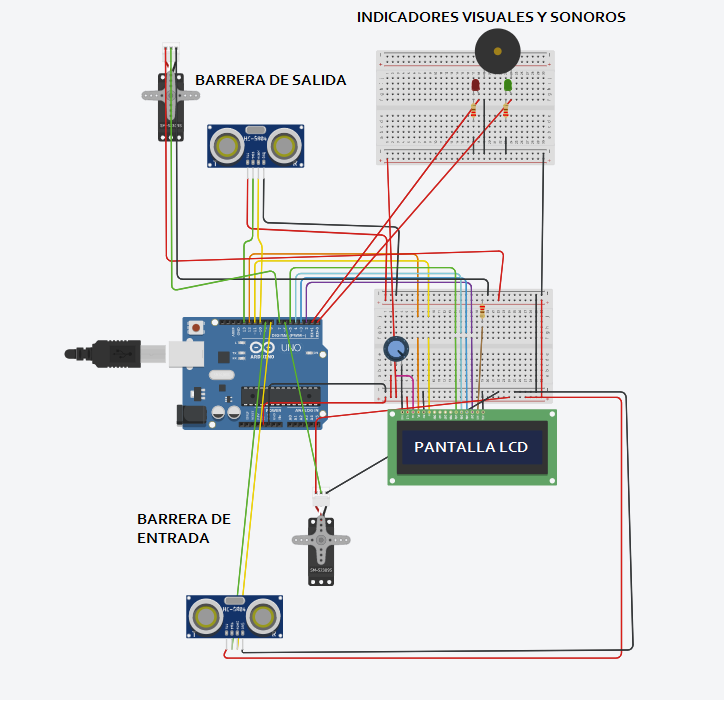
# ¿CÓMO Y DÓNDE SE CONSIGUIERON LOS MATERIALES?

Debido a los hechos de público conocimiento, relacionados con la pandemia COVID, se utilizó el simulador online TINKERCAD para realizar las pruebas y confeccionar el código de funcionamiento del proyecto.

# CÓDIGO FUENTE DEL PROGRAMA DEL MICRONTROLADOR

|  |
| --- |
| #define T\_SALIDA 13  #define E\_SALIDA 10  #define T\_ENTRADA 9  #define E\_ENTRADA 8  #define S\_ENTRADA 6  #define S\_SALIDA 7  #include <LiquidCrystal.h>  #include <Servo.h>  float sonido = 34300.0;  int adentro = 0;  bool entrando = false;  bool saliendo = false;  bool entroOSalio = false;  int segundosEntrando = 0;  int segundosSaliendo = 0;  LiquidCrystal lcd(12, 11, 5, 4, 3, 2);  Servo servoEntrada;  Servo servoSalida;  void setup() {  lcd.begin(16, 2);  servoEntrada.attach(S\_ENTRADA);  servoSalida.attach(S\_SALIDA);  servoEntrada.write(0);  servoSalida.write(0);  pinMode(T\_ENTRADA,OUTPUT);  pinMode(T\_SALIDA,OUTPUT);  pinMode(E\_ENTRADA,INPUT);  pinMode(E\_SALIDA,INPUT);  pinMode(1, OUTPUT);  pinMode(0, OUTPUT);  }  void loop() {  //lanzamos el pulso para medir las distancias  digitalWrite(T\_ENTRADA,HIGH);  delayMicroseconds(1);  digitalWrite(T\_ENTRADA,LOW);    //medimos la distancia del estacionamiento entrada  long tiempoEntrada = pulseIn(E\_ENTRADA,HIGH);  int distanciaEntrada = tiempoEntrada \* 0.000001 \* sonido / 2.0;    //lanzamos el pulso para medir las distancias  digitalWrite(T\_SALIDA,HIGH);  delayMicroseconds(1);  digitalWrite(T\_SALIDA,LOW);    //medimos la distancia del estacionamiento salida  long tiempoSalida = pulseIn(E\_SALIDA,HIGH);  int distanciaSalida = tiempoSalida \* 0.000001 \* sonido / 2.0;    //hay un vehículo que está tratando de entrar y en el ciclo anterior no había nadie  //tratando de entrar  if(distanciaEntrada > 25 && distanciaEntrada < 320 && entrando == false && adentro < 5){  digitalWrite(1, LOW);  digitalWrite(0, HIGH);  entrando = true;  servoEntrada.write(90);  lcd.clear();  lcd.setCursor(0,0);  lcd.print("BIENVENIDO");  lcd.setCursor(0,1);  lcd.print("ESPERE ");  lcd.setCursor(8,1);  lcd.print(segundosEntrando);  segundosEntrando = segundosEntrando + 1;  }  else{  if(distanciaEntrada > 25 && distanciaEntrada < 320 && entrando == true){  lcd.clear();  lcd.setCursor(0,0);  lcd.print("BIENVENIDO");  lcd.setCursor(0,1);  lcd.print("ESPERE ");  lcd.setCursor(8,1);  lcd.print(segundosEntrando);  segundosEntrando = segundosEntrando + 1;  }  if(adentro >= 5 && distanciaEntrada > 25 && distanciaEntrada < 320){  lcd.clear();  lcd.setCursor(0,0);  lcd.print("NO HAY MAS");  lcd.setCursor(0,1);  lcd.print("ESPACIO ADENTRO");  }  }    if(distanciaEntrada < 25 || distanciaEntrada > 320 && entrando == true && adentro < 5){  servoEntrada.write(0);  adentro = adentro + 1;  entrando = false;  entroOSalio = true;  segundosEntrando = 0;  digitalWrite(1, HIGH);  digitalWrite(0, LOW);  }    //hay un vehículo que está tratando de salir y en el ciclo anterior no había nadie tratando  //de salir  if(distanciaSalida > 25 && distanciaSalida < 320 && saliendo == false && adentro > 0){  digitalWrite(1, LOW);  digitalWrite(0, HIGH);  saliendo = true;  servoSalida.write(90);  lcd.clear();  lcd.setCursor(0,0);  lcd.print("NOS VEMOS");  lcd.setCursor(0,1);  lcd.print("ESPERE ");  lcd.setCursor(8,1);  lcd.print(segundosSaliendo);  segundosSaliendo = segundosSaliendo + 1;  }  else{  if(distanciaSalida > 25 && distanciaSalida < 320 && saliendo == true){  lcd.clear();  lcd.setCursor(0,0);  lcd.print("NOS VEMOS");  lcd.setCursor(0,1);  lcd.print("ESPERE ");  lcd.setCursor(8,1);  lcd.print(segundosSaliendo);  segundosSaliendo = segundosSaliendo + 1;  }  if(adentro <= 0 && distanciaSalida > 25 && distanciaSalida < 320){  lcd.clear();  lcd.setCursor(0,0);  lcd.print("NO HAY AUTOS");  lcd.setCursor(0,1);  lcd.print("ADENTRO");  }  }    if(distanciaSalida < 25 || distanciaSalida > 320 && saliendo == true && adentro > 0){  servoSalida.write(0);  adentro = adentro - 1;  saliendo = false;  entroOSalio = true;  segundosSaliendo = 0;  digitalWrite(1, HIGH);  digitalWrite(0, LOW);  }    if(entroOSalio == true){  lcd.clear();  lcd.setCursor(0,0);  lcd.print("AUTOS ADENTRO: ");  lcd.setCursor(0,1);  lcd.print(adentro);  entroOSalio = false;  }      delay(1000);    } |

# ESQUEMA DE CONEXIONES



# SOFTWARE Y LIBRERIAS UTILIZADAS

* **Simulador de Arduino online TINKERCAD**
* **Librería Servo.h: Para el control de servomotores**
* **Librería LiquidCrystal.h: Para el control de pantallas LCD**

# CONCLUSIONES

Se consiguió el objetivo del proyecto, de construir un sistema de barreras de estacionamiento, con indicadores visuales y sonoros de entrada y salida de vehículos. Surgieron algunos problemas menores de funcionalidad del código que fueron solucionados.