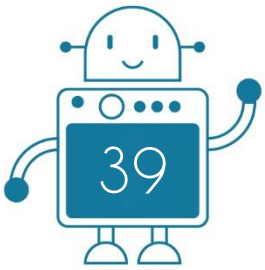


PARKING CONTROL

STEMJAM Teaching Guide

Developing make spaces to promote creativity around STEM in schools
Acronym: STEMJAM
Project no. 2016-1-ES01-KA201-025470

www.stemjam.eu

Co-funded by the Erasmus+ Programme of the European Union

PARKING CONTROL

ABSTRACT

The activity consists in creating a barrier like the parking. Combining arduino with mBot.

DIDACTIC OBJECTIVES

TECHNOLOGY

- Develop the code for the barrier.

ENGINEERING

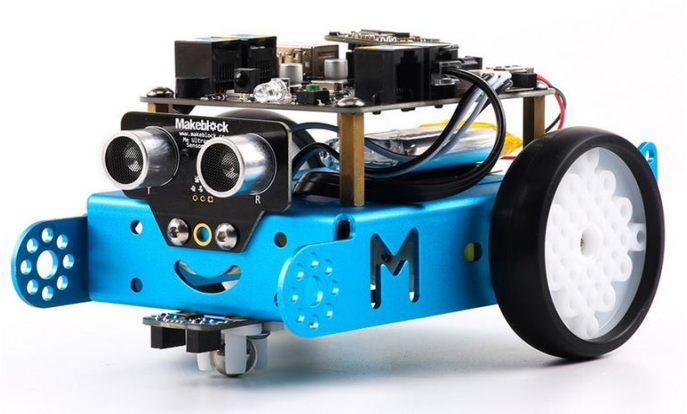
- Construct the system with Arduino and the barrier's house.

STEM Subject: Science☐ Technology ☒ Engineering☒ Mathematics☐

Education Level: 12-14 years☐ 14-16 years☒

BOM (Bill of Materials Needed)

- MBOT

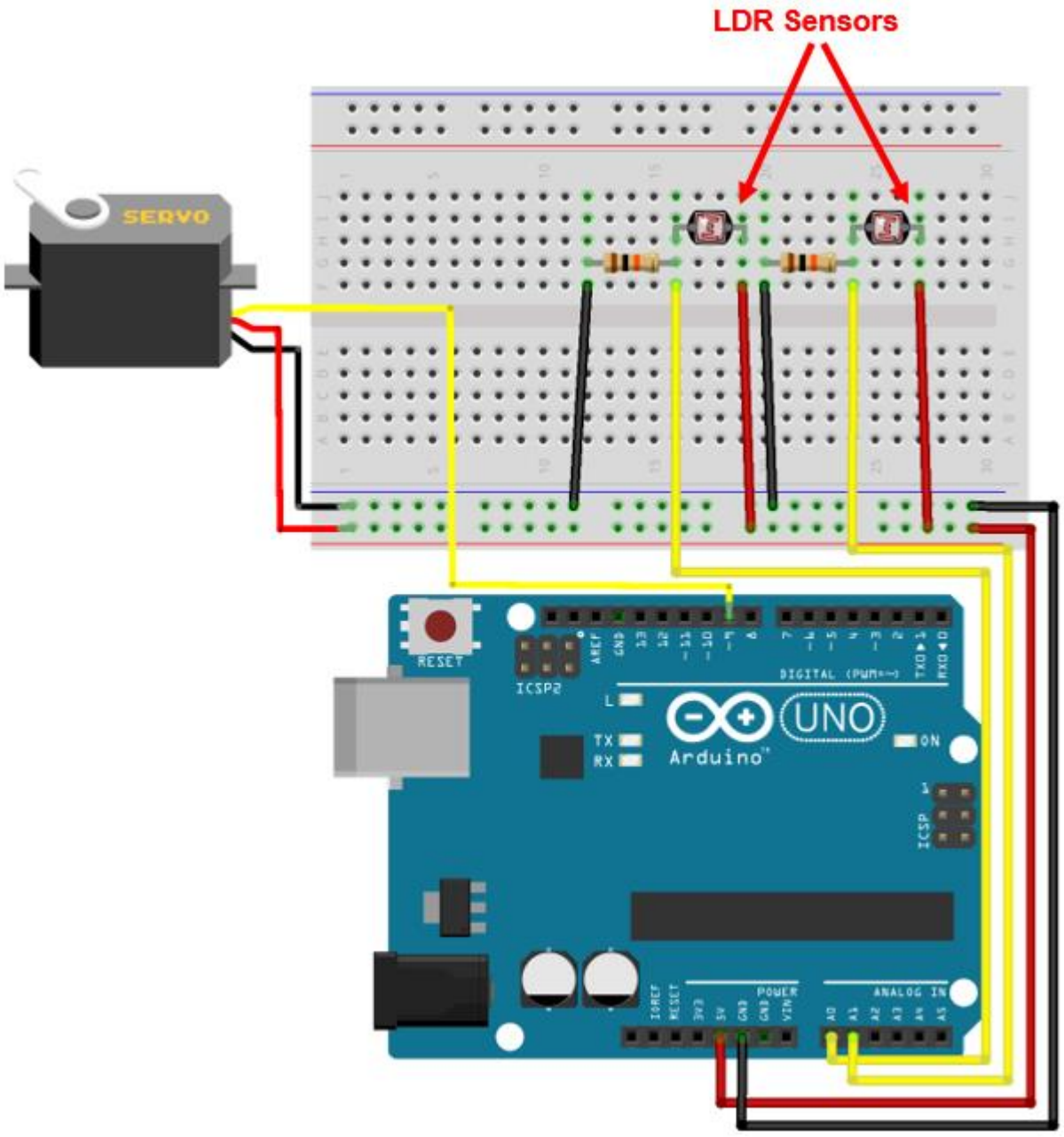


- Arduino Uno Shield.
- Protoboard.
- 2x LDR Sensors.
- 2x 10k ohms Resistor.
- Servo motor.
- Material to build the structure (i.e. Lego pieces).

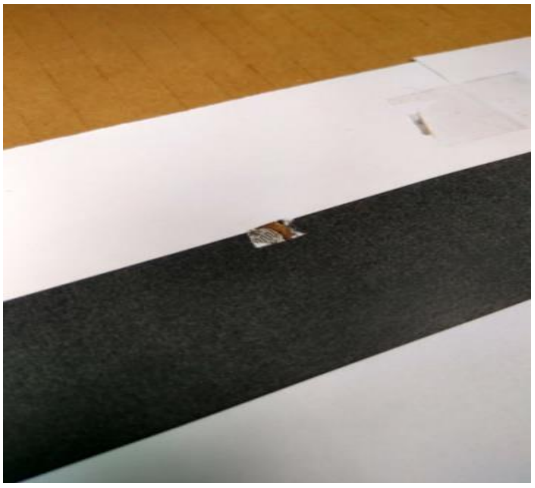
ACTIVITY DESCRIPTION

This project could be combined with MBOT's line follower. MBOT follow the black line, and when it arrives to the barrier, MBOT will stop and the barrier open it. The MBOT could follow the black line.

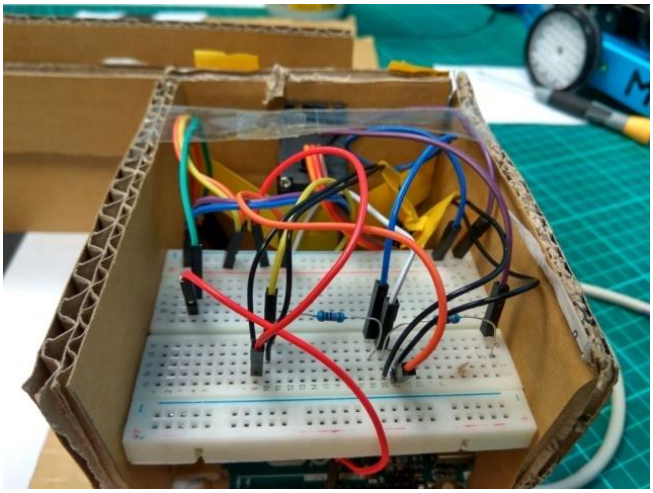
For visualize better the Arduino System works, we can view an example of the structure:

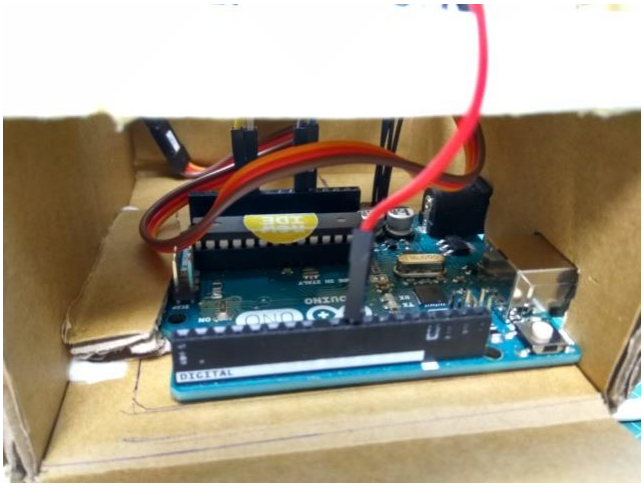


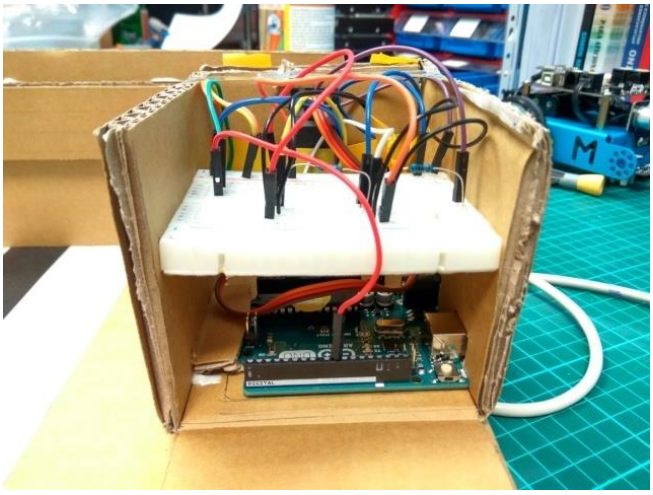
1 Black Line Follower

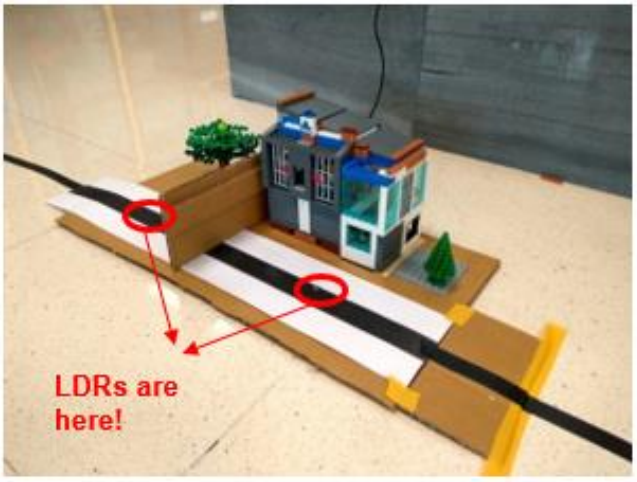


2 LDR Sensor



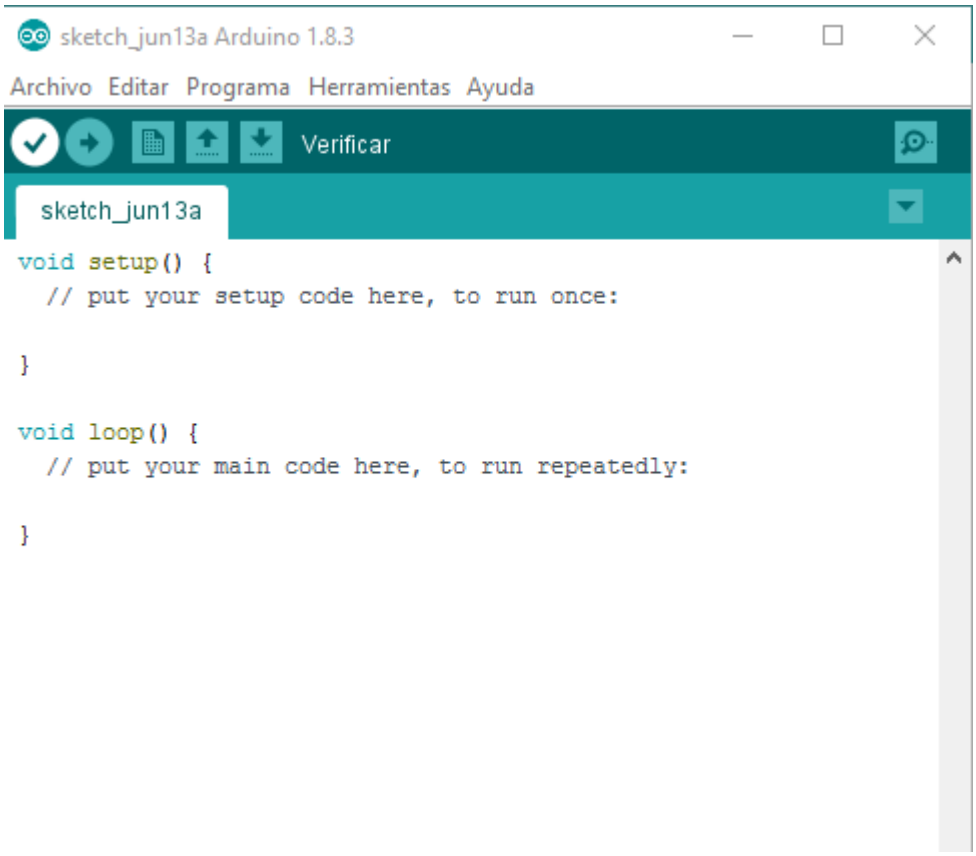






Once we have established the Arduino's Structure, we will develop the Barrier's code.

For do this, we use Arduino IDE software (https://www.arduino.cc/en/main/software)



We proceed to perform the programming:

1. Import the Servo's Library (you can download it in the Activity folder in Google Drive):

```
#include <Servo.h>
```

2. Create and initialize the variables:

```
Servo myservo;

int pos = 1;
int LDR1 = A0;
int LDR2 = A1;
int value1;
int value2;
```

3. Inside the “setup()” function we set the configuration of the shield:

```
void setup() {
  Serial.begin(9600);
  pinMode(LDR1, INPUT);
  pinMode(LDR2, INPUT);
  myservo.attach(9);
  myservo.write(1);
}
```

4. In “loop()” function, the code repeat all the instructions all the time:

```
void loop() {
  value1 = analogRead(LDR1);
  value2 = analogRead(LDR2);
  Serial.print(value1);
  Serial.print("|");
  Serial.println(value2);

  if (value1 < 780 && pos<=5 && value2 >900) {
    for (pos = 1; pos <= 90; pos += 1) { // goes from 0 degrees to 180 degrees
      // in steps of 1 degree
      myservo.write(pos);              // tell servo to go to position in variable 'pos'
      delay(15);                       // waits 15ms for the servo to reach the position
    }
  }

  if (value2 < 820 && pos>=85 && value1 >870) {
    for (pos = 90; pos >= 1; pos -= 1) { // goes from 180 degrees to 0 degrees
      myservo.write(pos);              // tell servo to go to position in variable 'pos'
      delay(15);                       // waits 15ms for the servo to reach the position
    }
  }
}
```

For the MBOT, we will program when it finds an obstacle to less than 10 meters it stops, and when it does not detect anything, it continues to advance.