

Introduction to robotics HW1

Slide prototype

Mantvydas Skruodys
2311158

September 22, 2025

Contents

1	Problem	3
2	Design	3
3	Parts list	3
4	Wiring	4
5	Main ideas for coding	4
6	Arduino code	4
7	Future improvements	7

1 Problem

We are creating a small prototype for the water park.

The people should be allowed to enter the slide if there is nobody else already sliding.

There is a light system:

- The red light means that the slide is occupied and you can not enter
- The green light means the slide is clear.
- If there is an error, for example if two people entered the slide at the same time, the system enters "error state"(yellow and red light together) that needs to be turned off by pressing the reset button.

2 Design

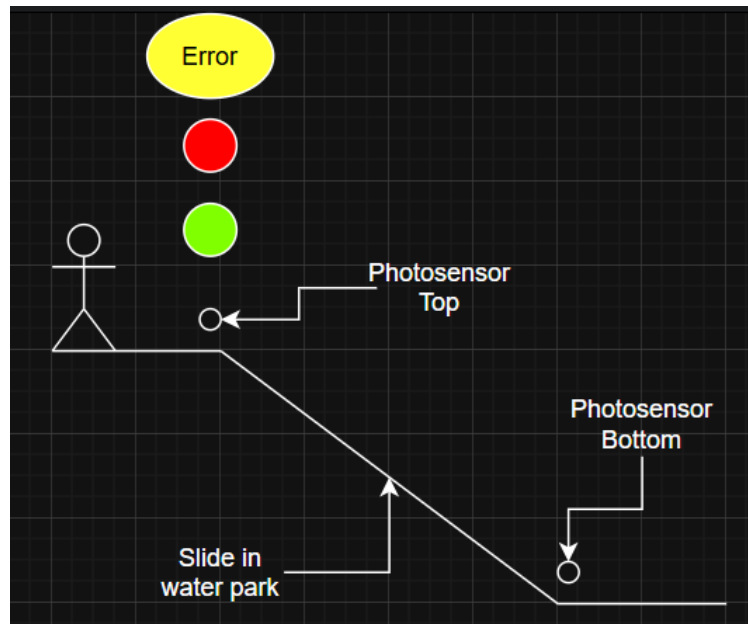


Figure 1: the instalation of the system on a slide

There are two photosensors at the top and bottom. They detect a person passing by. To have better reliability laser light should be pointed to the sensors.

3 Parts list

- 1x Arduino UNO R3
- 2x Photoresistor
- 2x resistor(10k Ω)
- 3x resistor(220 Ω)
- 1x red LED
- 1x green LED

- 1x yellow LED
- 1x button
- wires

4 Wiring

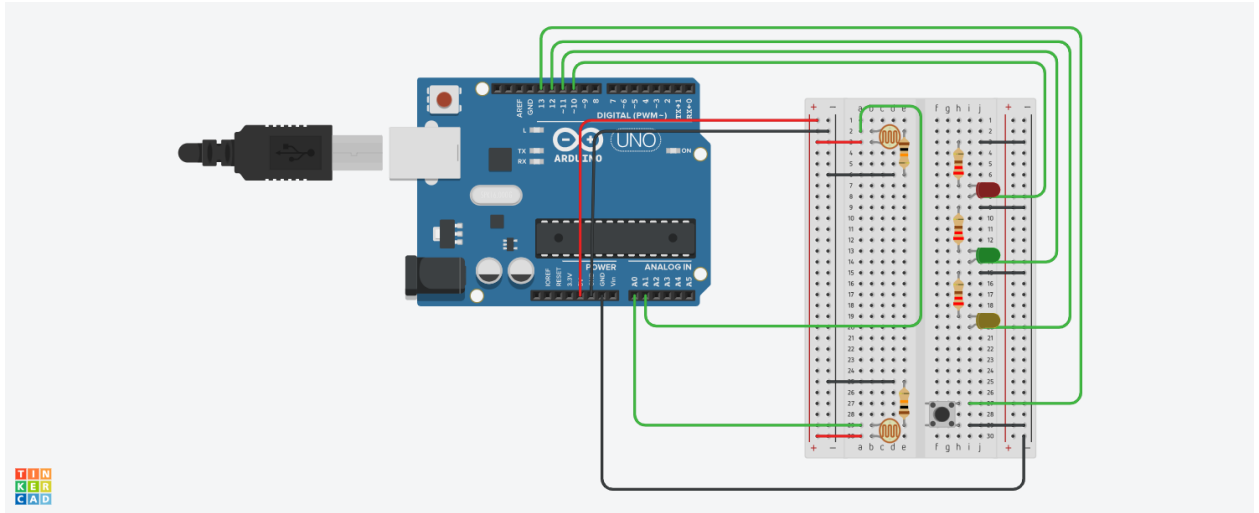


Figure 2: Arduino wiring

5 Main ideas for coding

Initial state of the system is when both of the photosensors detect light.

If a person blocks the top photoresistor, then we set the flag `topPassingStarted = true`. And when he leaves the photoresistor, so the light is not blocked - we set the flag `topPassed`.

The same happens for the bottom photoresistor.

The system handles these errors:

- a second person enters the slide before the first one reaches the bottom.
- somebody enters the slide from the bottom.
- somebody enters and does not reach the bottom during the giventime(for simulation we use 5s).
- when two people block both entrances.

6 Arduino code

```
//constants
const int topSensor = A1;
const int bottomSensor = A0;
```

```

const int redLight = 10;
const int greenLight = 11;
const int yellowLight = 12;
const int resetButton = 13;
const int initialBrightLight = 900;
const int passingLight = 100;
const unsigned long waitingTime = 5000; //5sec

bool begin = false;
int counter = 0;
bool topPassingStarted = false;
bool topPassed = false;
bool bottomPassingStarted = false;
bool bottomPassed = false;
bool error = false;

unsigned long timeTopPassed;

void setup() {
    Serial.begin(9600);
    pinMode(redLight, OUTPUT);
    pinMode(greenLight, OUTPUT);
    pinMode(yellowLight, OUTPUT);
    pinMode(resetButton, INPUT_PULLUP);
}

void loop() {
    int valueB = analogRead(bottomSensor);
    int valueT = analogRead(topSensor);

    //waiting until both sensors have enough light to initialize the system
    if(!begin && valueT >= initialBrightLight && valueB >= initialBrightLig
        begin = true;
        digitalWrite(greenLight, HIGH);
        digitalWrite(redLight, LOW);
        digitalWrite(yellowLight, LOW);

    }

    if(begin && !error) {
        if(valueT <= passingLight) {
            topPassingStarted = true;
        } else if(topPassingStarted) {
            topPassingStarted = false;
            topPassed = true;
            digitalWrite(greenLight, LOW);
            digitalWrite(redLight, HIGH);
            timeTopPassed = millis();
        }
    }
}

```

```

    if(topPassingStarted && topPassed) {
        error = true;
    }

    if(topPassed && millis() - timeTopPassed >= waitingTime) {
        error = true;
    }

    if(valueB <= passingLight) {
        bottomPassingStarted = true;
    } else if(bottomPassingStarted) {
        if(topPassed) {
            topPassed = false;
            digitalWrite(greenLight, HIGH);
            digitalWrite(redLight, LOW);
        }
        bottomPassingStarted = false;
        bottomPassed = true;
    }

    if(bottomPassingStarted && bottomPassed) {
        error = true;
    }

    if(!topPassed && bottomPassingStarted) {
        error = true;
    }

    if(topPassingStarted && bottomPassingStarted) {
        error = true;
    }
}

if(error) {
    digitalWrite(greenLight, LOW);
    digitalWrite(redLight, HIGH);
    digitalWrite(yellowLight, HIGH);
}

if(digitalRead(resetButton) == LOW){
    topPassed = false;
    topPassingStarted = false;
    bottomPassed = false;
    bottomPassingStarted = false;
    begin = false;
    error = false;
}
}

```

7 Future improvements

- Add sound allarm system when there is a error.
- Add lasers to point at the photoresistors.
- Add a counter of how many people passed(e.g. to do maintenance after 1000 people).