Semaphored Intersection

**Dondera Ionut Catalin gr. 1.2**

**Gut Ioan Filip gr. 1.2**

1. **Theme objectives and utility in real world**

When living in a big city, a semaphore is not on our list of unknown things. No matter where we go, may it be : work, school, shopping or just taking a walk, we have to get from one side of the street to the other (as pedestrians) or from point A to point B through many difficult situations in traffic (as drivers), we encounter the three colors that have to make it easy for us in any of the situations above. Thus, a semaphore is essential to keep the order and discipline in traffic, so that everyone arrives safe and sound to where they planned.

Our semaphored intersection is built in such a way that drivers and pedestrians have enough time to harmonize with each other in traffic without anyone getting hurt. The project contains two types of semaphores adapting to each of the traffic participants’ needs: one for the cars (drivers), including an emergency solution in case of earthquake and also an intermitent green which allows them at some time to go to the right even if the semaphore to go straight is red and one for the pedestrians, with an „extension” for people with disabilities.

The car semaphore has the red color for 10 seconds, the cars stop, the pedestrians being allowed to cross the street for 8 seconds, time in which the pedestrian semaphore is green. 10 seconds later, the car semaphore is green for 15 seconds, the cars move, the pedestrians stop for 20 seconds because their semaphore is red this time, meaning they have to allow the cars to pass. In the end, the car semaphore becomes yellow for 3 seconds and after that the entire story goes from begining to end over and over again.

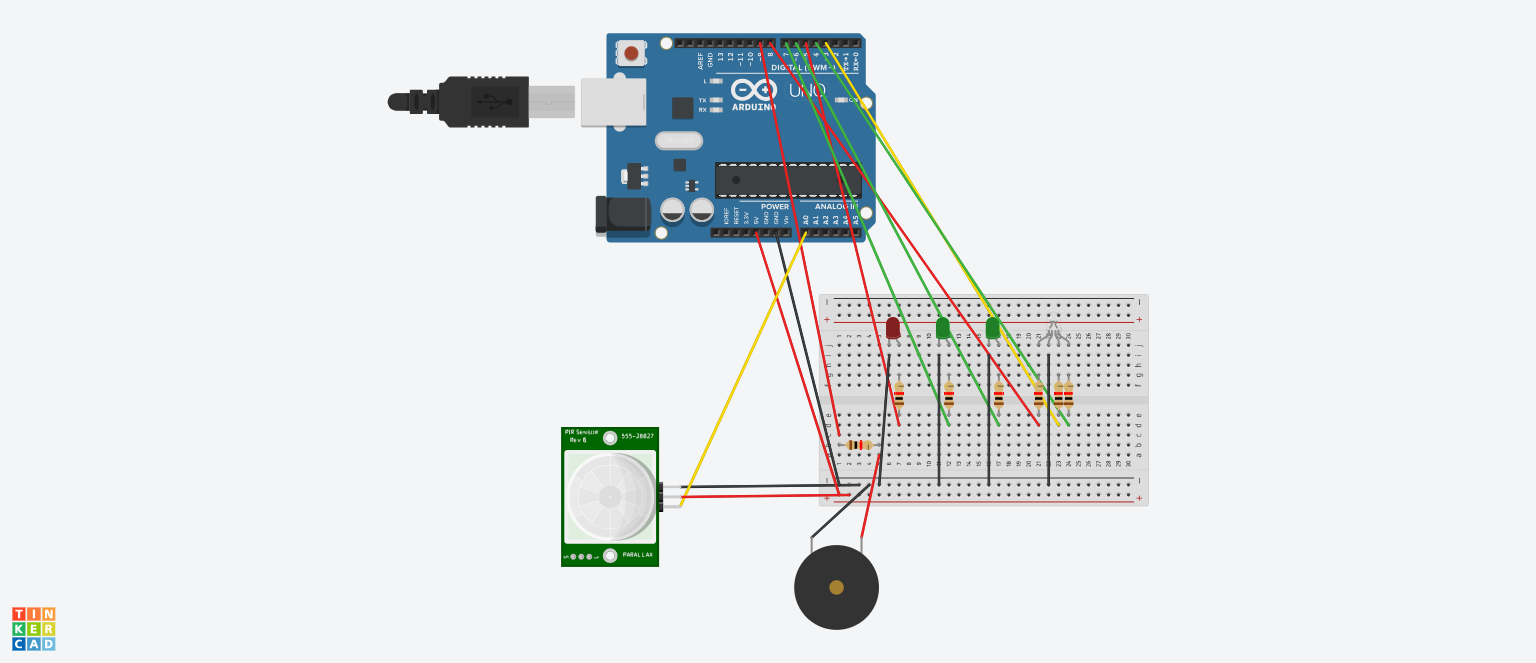
As we mentioned earlier, there is an emergency feature in case of eartquake : the car semaphore starts to blink, so that everyone is aware of danger and has to find a place to keep them safe.

As for the pedestrian semaphore’s „extension” , added especially for the less fortuned ones who can’t see, it relies in the buzzer that changes its sound (frequency) when there is a transition of the semaphore from red to green and green to red, so they can safely cross the street.

1. **Assembly components**

* Arduino UNO board;
* 1 RGB LED for the car semaphore;
* 1 Green LED and 1 Red LED for the pedestrian semaphore;
* 1 Green LED for the intermittent to the right feature of the car semaphore;
* 1 Buzzer for the pedestrian semaphore “extension” (people who can’t see);
* 1 Earthquake sensor (to detect earthquake as the name itself says).

1. **Associated actions and logical scheme**



1. **Source code in C language**

int RGBblue=10;

int RGBgreen=3;

int RGBred=4;

int BIred=5;

int BIgreen=11;

int LED=6;

int Eqk=A0;

int Buzzer=8;

int Button=9;

void setup()

{

Serial.begin(115200);

pinMode(RGBblue,OUTPUT);

pinMode(RGBgreen,OUTPUT);

pinMode(RGBred,OUTPUT);

pinMode(BIred,OUTPUT);

pinMode(LED,OUTPUT);

pinMode(Eqk,INPUT);

pinMode(Button,INPUT);

pinMode(Buzzer,OUTPUT);

pinMode(BIgreen,OUTPUT);

// initialisation of the serial connection

start();

}

void loop()

{

//delay(1000);

//float r= analogRead(Eqk);

//Serial.print(r);

//Serial.println();

eqK();

cars();

//pede();

}

void eqK(){

float r;

while((r=analogRead(Eqk))>10){

for(int i=0;i<5;i++){

Serial.println(r);

digitalWrite(BIred,LOW);

Serial.println("BIred OFF");

digitalWrite(BIgreen,LOW);

Serial.println("BIgreen OFF");

digitalWrite(RGBgreen,HIGH);

Serial.println("RGBgreen HIGH");

digitalWrite(RGBred,HIGH);

Serial.println("RGBred HIGH");

//digitalWrite(RGBblue,HIGH);

digitalWrite(LED,HIGH);

Serial.println("LED HIGH");

delay(500);

digitalWrite(RGBgreen,LOW);

Serial.println("RGBgreen LOW");

digitalWrite(RGBred,LOW);

Serial.println("RGBred LOW");

digitalWrite(RGBblue,LOW);

Serial.println("RGBblue LOW");

digitalWrite(LED,LOW);

Serial.println("LED OFF");

delay(500);

}

}

}

void start(){

digitalWrite(RGBred,HIGH);

Serial.println("RGBred HIGH");

digitalWrite(BIred,HIGH);

Serial.println("BIred HIGH");

delay(10000);

digitalWrite(RGBred,LOW);

Serial.println("RGBred LOW");

digitalWrite(BIred,LOW);

Serial.println("BIred LOW");

}

void cars(){

eqK();

digitalWrite(RGBred,HIGH);

Serial.println("RGBred HIGH");

eqK();

rightG();

eqK();

digitalWrite(LED,LOW);

Serial.println("LED LOW");

//blink green while red

eqK();

digitalWrite(RGBred,LOW);

Serial.println("RGBred LOW");

eqK();

digitalWrite(RGBgreen,HIGH);

Serial.println("RGBgreen HIGH");

for(int i=0;i<30;i++){

eqK();

digitalWrite(RGBgreen,HIGH);

Serial.println("RGBgreen HIGH");

delay(500);

}

eqK();

digitalWrite(RGBred,HIGH);

Serial.println("RGBred HIGH");

eqK();

delay(3000);

eqK();

digitalWrite(RGBgreen,LOW);

Serial.println("RGBgreen LOW");

eqK();

}

void rightG(){

for(int i=0;i<10;i++){

eqK();

if(i==1){

tone(Buzzer,150);

digitalWrite(BIred,LOW);

Serial.println("BIred LOW");

digitalWrite(BIgreen,HIGH);

Serial.println("BIgreen HIGH");

eqK();

}

if(i==9){

eqK();

digitalWrite(BIgreen,LOW);

Serial.println("BIgreen LOW");

digitalWrite(BIred,HIGH);

Serial.println("BIred HIGH");

tone(Buzzer,300);

}

digitalWrite(LED,HIGH);

Serial.println("LED HIGH");

delay(500);

eqK();

digitalWrite(LED,LOW);

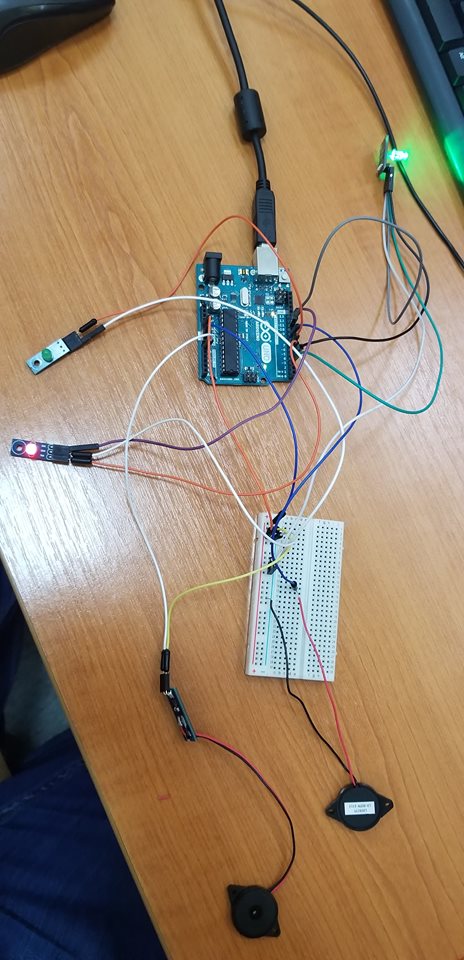
Serial.println("LED LOW");

delay(500);

}

eqK();

1. **Photo during project activity**



1. **Test Plan**

In the beginning we tested each component one by one to see if everything works as it should.

We assembled the leds, buzzer and earthquake on the arduino and tested the first code we wrote, just turning on and off the leds. Then we started modifying the code and testing different ideas until we reached the final solution. We took every assignment step by step, synchronizing the events with each other.

1. **Conclusions**

We enjoyed working on this project and we have learned many things. At the begining we had a bit of a hard time when we had to implement the delays between leds colors in order to build exactly what the requirements told us, but after some testing we managed to do so. We wrote our Arduino code as simple and as clear for a reader as possible. We have both worked and understood equally what we had done in our code. Overall, we have learned about the concept of delay, about working together on solving problems and organizing our work.