

# Car Parking System

## Introducere

Anghelescu Vlad-Gabriel 1222A

## Descriere generală

Proiectul este reprezentat de un sistem automat de parcare. La fiecare intrare sau iesire a unei masini din parcare, sistemul afiseaza cu ajutorul unui LCD i2C cate locuri sunt libere in parcare, respectiv daca parcare este plina. Programul stie de la bun inceput cate locuri de parcare exista in total, iar la iesirea unei masini adauga un loc liber, iar la intrarea unei masini scade un loc de parcare. In momentul in care parcare este plina si nu mai exista locuri libere, LCD-ul afiseaza mesajul "Parking Full" si nu mai ridica bariera pentru a oferi acces in parcare. Modul de functionare: In momentul intrarii sau iesirii din parcare, masina trece prin 2 senzori, bariera fiind ridicata automat la trecerea prin primul senzor. In functie de senzorul prin care trece prima data masina, programul isi da seama daca masina intra sau iese din parcare si aduna sau scade un loc pentru a tine cont de cate locurile sunt libere.

## Hardware Design and Components List

Lista de componente:

-Arduino UNO -2x IR Proximity Sensor -16x2 LCD I2C -Servo -Jumpers

## Software Design

Codul folosit pentru realizarea proiectului:

```
Arduino Car Parking System #include <Wire.h> #include<LiquidCrystal_I2C.h> LiquidCrystal_I2C  
lcd(0x27,16,2); #include <Servo.h> Servo myservo1; int IR1 = 2; int IR2 = 4; int Slot = 4; int flag1 =
```

```
0; int flag2 = 0; void setup() { lcd.init(); lcd.begin(16,2); lcd.backlight(); pinMode(IR1, INPUT);
pinMode(IR2, INPUT); lcd.setCursor (0,0); lcd.print(" ARDUINO "); lcd.setCursor (0,1); lcd.print("
PARKING SYSTEM "); delay (2000); lcd.clear(); myservo1.attach(3); myservo1.write(100); } void
loop(){ if(digitalRead (IR1) == LOW && flag1==0){ if(Slot>0){flag1=1;
if(flag2==0){myservo1.write(0); Slot = Slot-1;} }else{ lcd.setCursor (0,0); lcd.print(" SORRY :( ");
lcd.setCursor (0,1); lcd.print(" Parking Full "); delay (3000); lcd.clear(); } } if(digitalRead (IR2) ==
LOW && flag2==0){flag2=1; if(flag1==0){myservo1.write(0); Slot = Slot+1;} } if(flag1==1 &&
flag2==1){ delay (1000); myservo1.write(100); flag1=0, flag2=0; } lcd.setCursor (0,0); lcd.print("
WELCOME! "); lcd.setCursor (0,1); lcd.print("Slot Left: "); lcd.print(Slot); } Arduino 16x2 LCD I2C
Scanner #include <Wire.h> void setup() { Wire.begin(); Serial.begin(9600); Serial.println("\nI2C
Scanner"); } void loop() { byte error, address; int Devices; Serial.println("Scanning..."); Devices = 0;
for(address = 1; address < 127; address++ ) {
```

```
Wire.beginTransmission(address); error = Wire.endTransmission(); if (error == 0) { Serial.print("I2C
device found at address 0x"); if (address<16) Serial.print("0"); Serial.print(address,HEX);
Serial.println(" !"); Devices++; } else if (error==4) { Serial.print("Unknown error at address 0x"); if
(address<16) Serial.print("0"); Serial.println(address,HEX); } } if (Devices == 0) Serial.println("No I2C
devices found\n"); else Serial.println("done\n"); delay(5000); }
```

Librariile folosite:

-Wire.h -LiquidCrystal\_I2C.h -Servo.h

## Rezultate Obținute

Proiectul functioneaza perfect si face exact ce scrie in descrierea generala.

## Concluzii

## Download

## Jurnal

Pasi pentru obtinerea proiectului. 1.Prima data a fost realizat si pus la punct codul. 2. Realizarea

machetei

## Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/cstan/10>



Last update: **2022/05/27 21:07**