

```

#include <Servo.h>
#include <SPI.h>
#include <MFRC522.h>
#include <LiquidCrystal_I2C.h>
#include <Wire.h>

int const ledPin2 = 5 ;
int const ledPin3 = 6 ;
int const ledPin4 = 7 ;

int const photoresistorPin2 = A1 ;
int const photoresistorPin3 = A2 ;
int const photoresistorPin4 = A3 ;

#define RST_PIN          9           // Configurable, see typical pin layout
above
#define SS_PIN           10          // Configurable, see typical pin layout
above

MFRC522 mfrc522(SS_PIN, RST_PIN); // Create MFRC522 instance
LiquidCrystal_I2C lcd(0x27,16,2);
Servo myServo; // Create a servo object
// E3 E6 AD A5 ID RFID
#define openBar 2
#define closeBar 3

int s2,s3,s4,Total;

void setup() {
  lcd.init();
  lcd.backlight();

  Serial.begin(9600);

  pinMode(photoresistorPin2, INPUT);
  pinMode(photoresistorPin3, INPUT);
  pinMode(photoresistorPin4, INPUT);
  // Set photoresistor pin as input

  pinMode(ledPin2, OUTPUT);
  pinMode(ledPin3, OUTPUT);
  pinMode(ledPin4, OUTPUT);

```

```

    // Set LED pin as output
    lcd.clear();
    lcd.setCursor(3,0);
    lcd.print("CAR PARKING");
    lcd.setCursor(5,1);
    lcd.print("SYSTEM");
    delay(3000);
    lcd.clear();

    //////////////////////////////////////
    myServo.attach(8);
    pinMode(closeBar, INPUT_PULLUP);
    pinMode(openBar, INPUT_PULLUP);
    SPI.begin();          // Init SPI bus
    mfrc522.PCD_Init();    // Init MFRC52
    //////////////////////////////////////
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("S1:----");
}

void loop() {
    lcd.setCursor(0,0);
    lcd.print("S1:----");

    int photoresistorValue2 = analogRead(photoresistorPin2);
    int photoresistorValue3 = analogRead(photoresistorPin3);
    int photoresistorValue4 = analogRead(photoresistorPin4);

    //////////// S2
    if (photoresistorValue2 <= 150) {
        digitalWrite(ledPin2, LOW);
        Serial.println("2 is OFF ");
        Serial.println(photoresistorValue2);
        //delay(1000);
        //lcd.clear();
        lcd.setCursor (8,0);
        lcd.print("S2:FULL");
        //delay(100);
        //lcd.clear();
        s2=1;
    }
    else {

        digitalWrite(ledPin2, HIGH);
        Serial.println("2 is ON");
        Serial.println(photoresistorValue2);
    }
}

```

```

    //delay(1000);
    //lcd.clear();
    lcd.setCursor(8,0);
    lcd.print("S2:-----");
    //delay(100);
    //lcd.clear();
    s2=0;
}
////////// S3
if (photoresistorValue3 <= 150) {
    digitalWrite(ledPin3, LOW);
    Serial.println("3 is OFF ");
    Serial.println(photoresistorValue3);
    //delay(1000);
    //lcd.clear();
    lcd.setCursor (0,1);
    lcd.print("S3:FULL");
    //delay(100);
    //lcd.clear();
    s3=1;
}
else {

    digitalWrite(ledPin3, HIGH);
    Serial.println("3 is ON");
    Serial.println(photoresistorValue3);
    //delay(1000);
    //lcd.clear();
    lcd.setCursor(0,1);
    lcd.print("S3:----");
    //delay(100);
    //lcd.clear();
    s3=0;
}
////////// S4
if (photoresistorValue4 <= 150) {
    digitalWrite(ledPin4, LOW);
    Serial.println("4 is OFF ");
    Serial.println(photoresistorValue4);
    //delay(1000);
    //lcd.clear();
    lcd.setCursor (8,1);
    lcd.print("S4:FULL");
    //delay(100);
    //lcd.clear();
    s4=1;
}
else {

```

```

    digitalWrite(ledPin4, HIGH);
    Serial.println("4 is ON");
    Serial.println(photoresistorValue4);
    //delay(1000);
    //lcd.clear();
    lcd.setCursor(8,1);
    lcd.print("S4:-----");
    //delay(100);
    //lcd.clear();
    s4=0;

}

delay(100);
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
Total = s2 + s3 + s4 ;
if(Total==3 && digitalRead(openBar) == LOW) {
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("PARKING IS ");
    lcd.setCursor(4, 1);
    lcd.print("FULL");
    delay(3000);
    lcd.clear();

}
else {
    if(digitalRead(openBar) == LOW){
        myServo.write(0);
        delay(5000);
        myServo.write(120);
    }
    if(digitalRead(closeBar) == LOW){
        myServo.write(0);
        delay(5000);
        myServo.write(120);
    }
}
// Look for new cards
if ( ! mfrc522.PICC_IsNewCardPresent())
{
    return;
}
// Select one of the cards
if ( ! mfrc522.PICC_ReadCardSerial())
{
    return;
}
}

```

```

//Show UID on serial monitor
Serial.print("UID tag :");
String content= "";
byte letter;
for (byte i = 0; i < mfrc522.uid.size; i++)
{
    Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
    Serial.print(mfrc522.uid.uidByte[i], HEX);
    content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
    content.concat(String(mfrc522.uid.uidByte[i], HEX));
}
Serial.println();
Serial.print("Message : ");
content.toUpperCase();
if (content.substring(1) == "AA DA D6 81") //change here the UID of the
card/cards that you want to give access
{
    Serial.println("Authorized access");
    Serial.println();
    myServo.write(0);
    delay(5000);
    myServo.write(120);
}
}

```