# **PROJECT IOT102**

# **Bãi đỗ xe quản lý tự động**

| Nhóm 7 | * Nguyễn Anh Đức * Vũ Đức Kiên * Trần Nhật Minh | HE153720  HE151210  HE161729 |
| --- | --- | --- |

## **I. Ý tưởng**

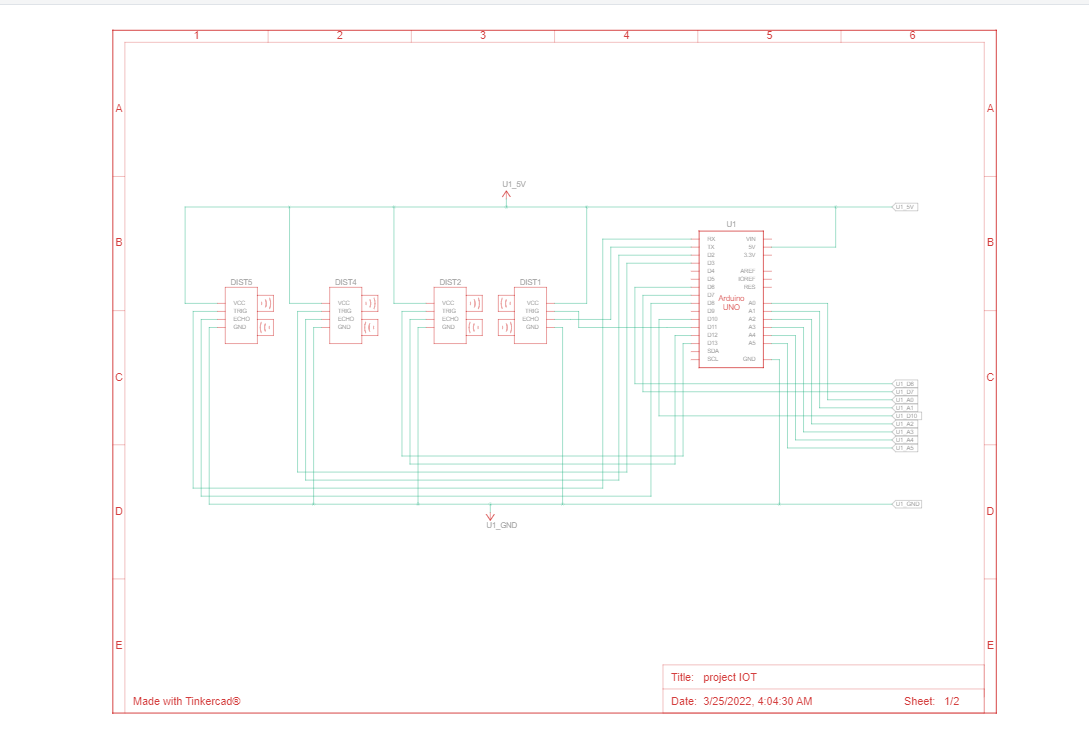
Ý tưởng của nhóm em mang đến giải pháp giúp người di chuyển phương tiện bằng ô tô có thể tìm được chỗ đậu xe nhanh chóng trong các chung cư hoặc khu đô thị lớn, giải pháp này giúp người lái tiết kiệm thời gian khi tìm chỗ trống hoặc không tốn công sức tìm chỗ gửi trong khi bãi đậu lại không còn chỗ. Giải pháp này có thể kết hợp với thẻ từ dành riêng cho cư dân trong chung cư, hoặc các bãi đỗ xe có nhiều tầng.

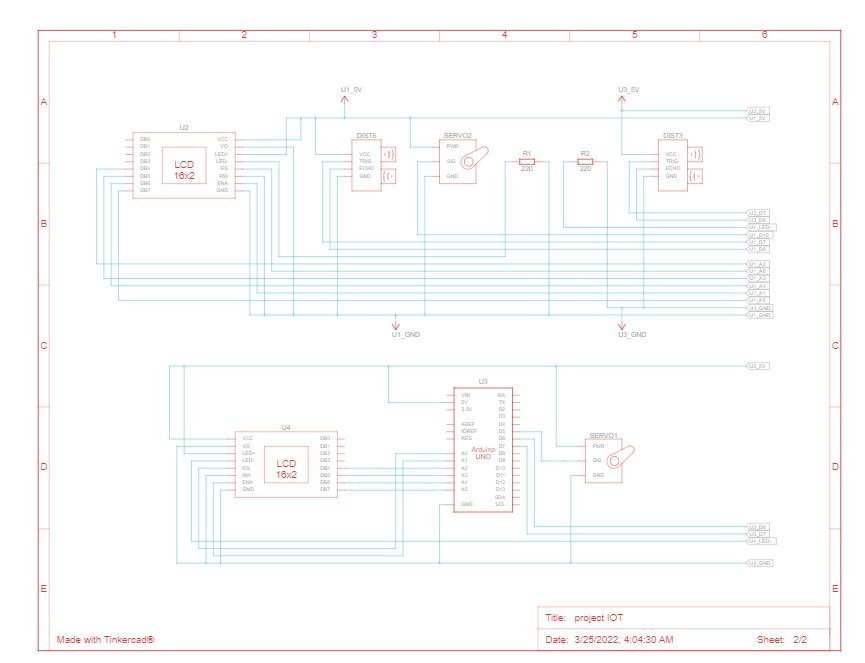
## **II. Nội dung**

### ***1. Linh kiện (Hardware Required )***

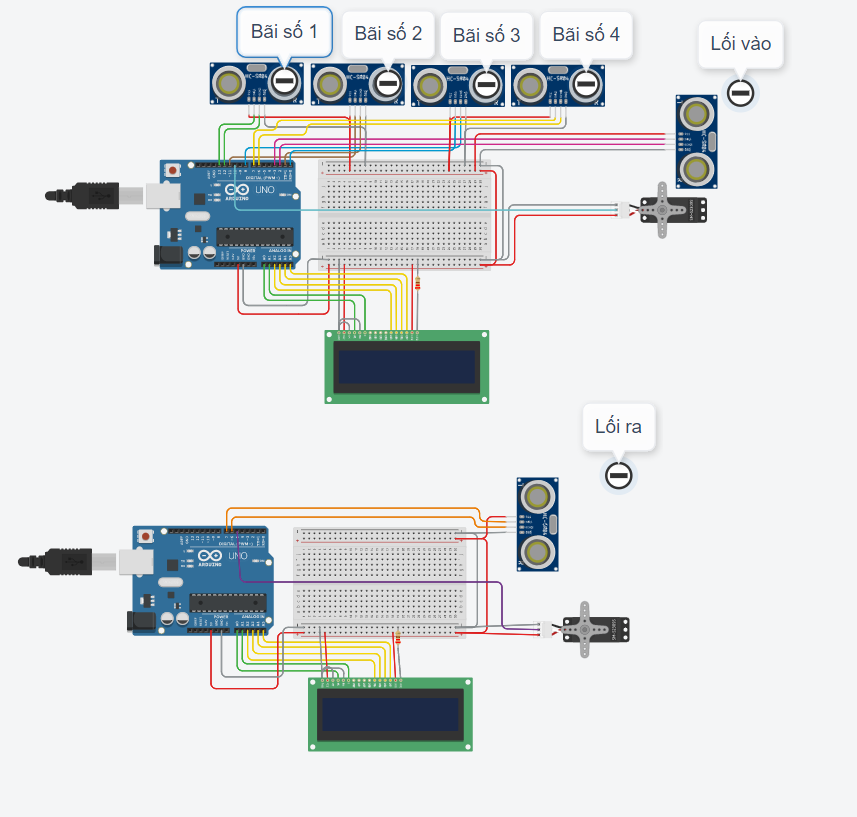
* 2 Arduino Uno R3
* 2 LED Display LCD 16 x 2
* 6 Ultrasonic Distance Sensor
* 2 Resistor 220Ω
* 2 Micro Servo

### ***2. Mạch nguyên lý (Schematic)***

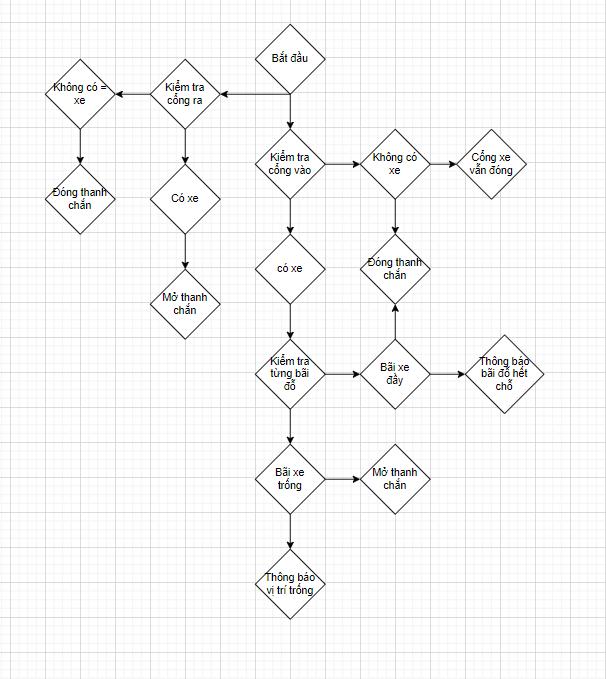




### ***3. Mạch kết nối (Circuit )***



### ***4. Lưu đồ thuật toán (Flow chart)***



## **III. Kết quả đạt được**

Ý tưởng có thể áp dụng thực tiễn nhằm quản lý bãi đỗ xe hiệu quả và có trật tự đối với các nơi đông dân cư để tránh tình trạng để xe lộn xộn. Với việc liên tục kiểm tra bãi trống mỗi khi có xe đến sẽ giúp tài xế nhanh chóng tìm đc bãi đỗ xe mà không tốn quá nhiều thời gian tìm chỗ. Thanh chắn barrier sẽ được mở khi có bãi đỗ trống và sẽ đóng khi không có xe ra vào hoặc khi bãi đỗ hết chỗ đồng thời thông báo tình trạng về bãi đỗ xe. Mỗi khi có xe ra thanh chắn barrier sẽ mở tự động mà không cần check bãi đỗ xe.

## **IV. Phụ lục**

### ***1. Link Tinkercad:*** https://www.tinkercad.com/things/hRcNT4Ihhyl-project-iot/editel?sharecode=MyTRZXmX7jURerisaGKbLmbsiwelNDdr5auXHVd6YMg&fbclid=IwAR3ej4g1Wbn4UuNUcQSzVxjX0qRbgPgEVdjoz7wi3v8yFeH-3ePNFIlFB58

***2. Code***

Arduino 1(dành cho cổng vào):

#include <LiquidCrystal.h>

#include <Servo.h>

Servo myservo;

LiquidCrystal lcd(A0, A1, A2, A3, A4, A5);

long duration;

int distance;

int pin = 1;

#define IN5 10

void setup()

{

pinMode(13, OUTPUT);

pinMode(11, OUTPUT);

pinMode(0, OUTPUT);

pinMode(7, OUTPUT);

pinMode(3, OUTPUT);

pinMode(5, OUTPUT);

pinMode(12, INPUT);

pinMode(1, INPUT);

pinMode(8, INPUT);

pinMode(6, INPUT);

pinMode(2, INPUT);

pinMode(4, INPUT);

lcd.begin(16,2);

lcd.setCursor(0,0);

lcd.print("Stop here! ");

myservo.attach(10);

}

void loop(){

boolean CarIn=false;

boolean CarOut=true;

int slot[] = {1,2,3,4};

CarIn = checkCarIn();

if(CarIn){

for(int i=0;i<4; i++){

boolean haveCar = false;

if(i==0){

haveCar = checkCar1();

}

if(i==1){

haveCar = checkCar2();

}

if(i==2){

haveCar = checkCar3();

}

if(i==3){

haveCar = checkCar4();

}

if(haveCar){

slot[i]=0;

}else{

slot[i]=i+1;

}

}

if(slot[0]==0&&slot[1]==0&&slot[2]==0&&slot[3]==0){

lcd.setCursor(0,0);

lcd.print("No place left! ");

myservo.write(0);

}

else{

for(int i=0;i<4; i++){

int pos = slot[i];

if(pos!=0){

lcd.setCursor(0,0);

lcd.print("Go to position ");

lcd.print(pos);

myservo.write(90);

break;

}

}

}

}

else{

lcd.setCursor(0,0);

lcd.print("Stop here! ");

myservo.write(0);

}

}

boolean checkCar1(){

digitalWrite(13, LOW);

delayMicroseconds(2);

digitalWrite(13, HIGH);

delayMicroseconds(10);

digitalWrite(13, LOW);

duration = pulseIn(12, HIGH);

distance= duration\*0.034/2;

if(distance>=331){

return false;

}

if(distance<330&distance>0){

return true;

}

}

boolean checkCar2(){

digitalWrite(11, LOW);

delayMicroseconds(2);

digitalWrite(11, HIGH);

delayMicroseconds(10);

digitalWrite(11, LOW);

duration = pulseIn(1, HIGH);

distance= duration\*0.034/2;

if(distance>=331){

return false;

}

if(distance<330&distance>0){

return true;

}

}

boolean checkCar3(){

digitalWrite(0, LOW);

delayMicroseconds(2);

digitalWrite(0, HIGH);

delayMicroseconds(10);

digitalWrite(0, LOW);

duration = pulseIn(8, HIGH);

distance= duration\*0.034/2;

if(distance>=331){

return false;

}

if(distance<330&distance>0){

return true;

}

}

boolean checkCar4(){

digitalWrite(7, LOW);

delayMicroseconds(2);

digitalWrite(7, HIGH);

delayMicroseconds(10);

digitalWrite(7, LOW);

duration = pulseIn(6, HIGH);

distance= duration\*0.034/2;

if(distance>=331){

return false;

}

if(distance<330&distance>0){

return true;

}

}

boolean checkCarIn(){

digitalWrite(3, LOW);

delayMicroseconds(2);

digitalWrite(3, HIGH);

delayMicroseconds(10);

digitalWrite(3, LOW);

duration = pulseIn(2, HIGH);

distance= duration\*0.034/2;

if(distance>=331){

return false;

}

if(distance<330&distance>0){

return true;

}

}

Arduno 2(dành cho cổng ra):

#include <LiquidCrystal.h>

#include <Servo.h>

Servo myservo;

LiquidCrystal lcd(A0, A1, A2, A3, A4, A5);

long duration;

int distance;

void setup()

{

pinMode(7, OUTPUT);

pinMode(6, INPUT);

lcd.begin(16,2);

lcd.setCursor(0,0);

lcd.print("Stop here! ");

myservo.attach(5);

}

void loop(){

boolean CarOut=false;

CarOut = checkCarOut();

if(CarOut){

lcd.setCursor(0,0);

lcd.print("Barrie open! ");

myservo.write(90);

}

else{

lcd.setCursor(0,0);

lcd.print("Stop here! ");

myservo.write(0);

}

}

boolean checkCarOut(){

digitalWrite(7, LOW);

delayMicroseconds(2);

digitalWrite(7, HIGH);

delayMicroseconds(10);

digitalWrite(7, LOW);

duration = pulseIn(6, HIGH);

distance= duration\*0.034/2;

if(distance>=331){

return false;

}

if(distance<330&distance>0){

return true;

}

}