

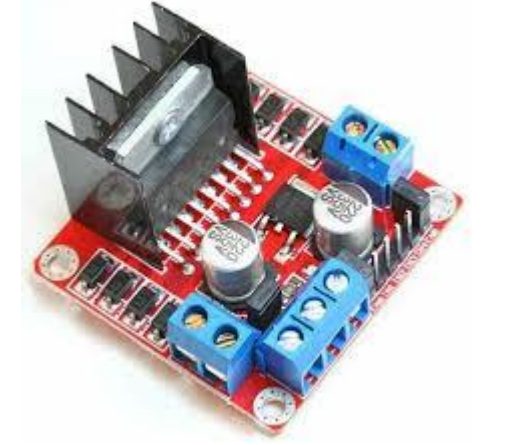


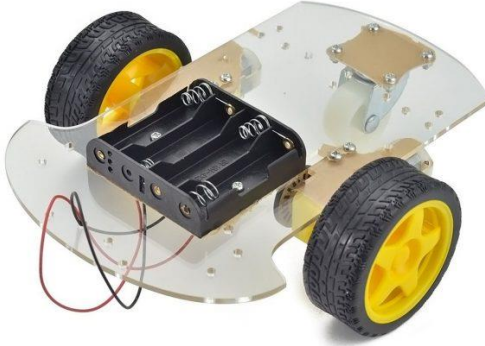


Đề tài xe điều khiển bằng Bluetooth

1. Thành phần.

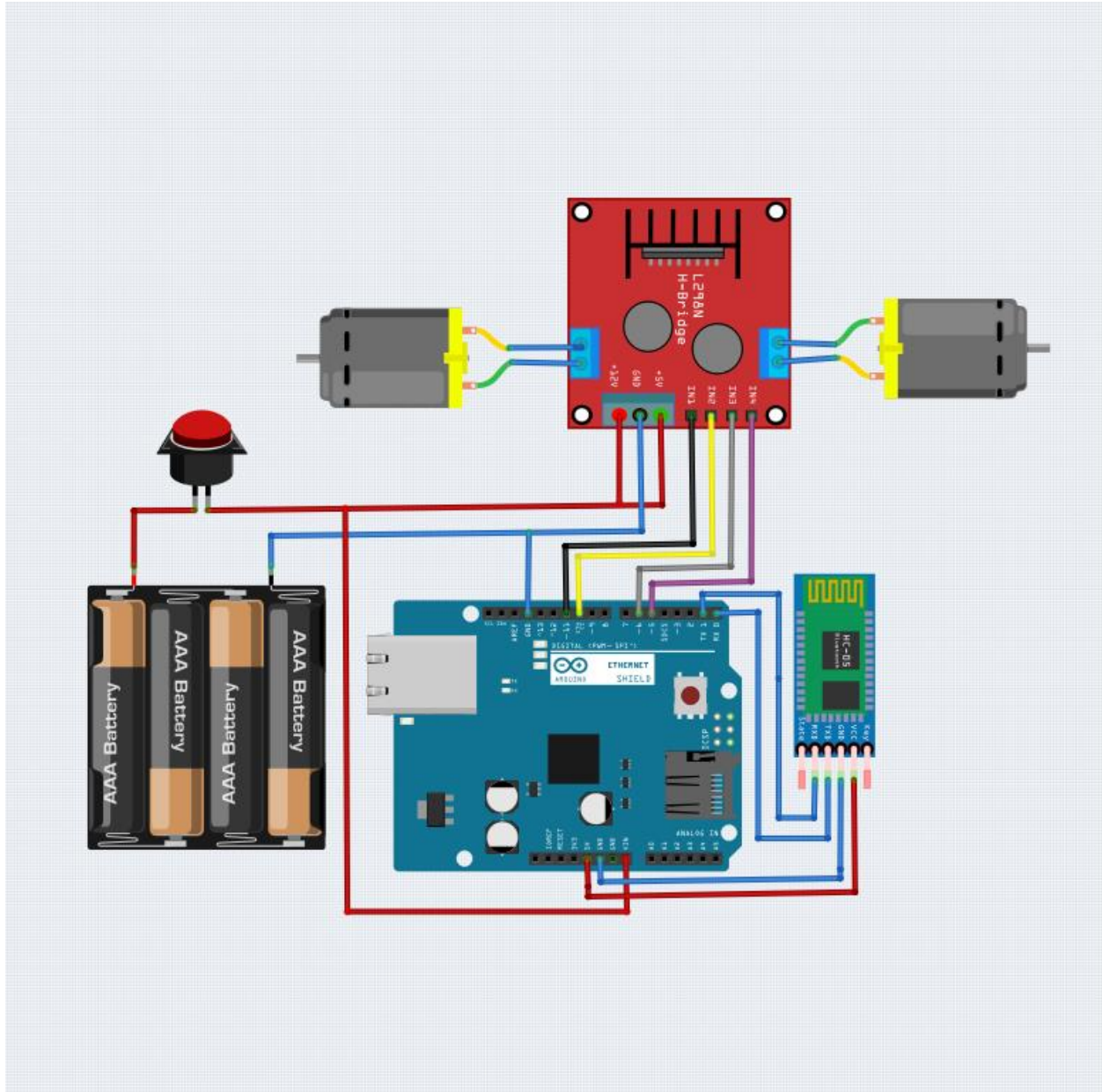
Tên linh kiện	Chức năng	Hình ảnh
Arduino UNO	Bộ vi điều khiển trung tâm để nhận, thu tín hiệu và điều khiển hệ thống	
Module HC-05	Giao tiếp truyền dữ liệu từ smartphone đến Arduino Uno	
Module L298	Điều khiển tốc độ, chiều quay của động cơ.	

LCD 16x2	Hiển thị dữ liệu 16 cột, 2 hàng	
Module I2C LCD	Giúp LCD giao tiếp I2C với Arduino	
Khung xe	Sử dụng khung xe có sẵn	

Công tắc		 <p>CÔNG TẮC ON-OFF 2 chế độ</p>
Khay pin		

2. Sơ đồ kết nối.

- Download phần mềm Fritzing



3. Code

```
#include <Wire.h>
#include<LiquidCrystal_I2C.h>;
#define motoLeft1 11
#define motoLeft2 10
#define motoRight1 6
#define motoRight2 5
LiquidCrystal_I2C lcd(0x27,16,2);
int val;

void setup() {
  Serial.begin(9600);
  pinMode(motoLeft1, OUTPUT);
  pinMode(motoLeft2, OUTPUT);
  pinMode(motoRight1, OUTPUT);
  pinMode(motoRight2, OUTPUT);
  lcd.init();
  lcd.backlight();
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" Tran Van Quyet ");
  lcd.setCursor(0,1);
  lcd.print("Lop: 08DHLDT_CT ");
  delay(2500) ;
  lcd.setCursor(0,0);
  lcd.print("** PROJECT **");
  lcd.setCursor(0,1);
  lcd.print(" BLUETOOTH CAR ");
  delay(2000);
  lcd.setCursor(0,0);
  lcd.print("** Status **");
  lcd.setCursor(0,1);
  lcd.print("   Stop   ");
}

void hc05(){
  if (Serial.available()>0){
    val=Serial.read();
    switch(val)
    {
```

```

case 'S' :
    lcd.setCursor(0,1);
    lcd.print("  Stop  ");
    ngung();
    break;
case 'F' :
    lcd.setCursor(0,1);
    lcd.print(" Forward ");
    forward();
    break;
case 'B' :
    lcd.setCursor(0,1);
    lcd.print(" Backward ");
    backward();
    break;
case 'L' :
    lcd.setCursor(0,1);
    lcd.print(" Left  ");
    turnLeft();
    break;
case 'R' :
    lcd.setCursor(0,1);
    lcd.print(" Right ");
    turnRight();
    break;
}

```

```

}}
void dongcotrai(int x, int y){
    analogWrite(motoLeft1,x);
    analogWrite(motoLeft2,y);
}
void dongcophai(int j,int k){
    analogWrite(motoRight1,j);
    analogWrite(motoRight2,k);
}
void forward(){
    dongcotrai(0,255);
    dongcophai(0,255);
}
void backward(){
    dongcotrai(255,0);
}

```

```
    dongcophai(255,0);  
}  
void ngung(){  
    dongcotrai(0,0);  
    dongcophai(0,0);  
}  
void turnLeft(){  
    dongcotrai(255,0);  
    dongcophai(0,255);  
}  
void turnRight(){  
    dongcotrai(0,255);  
    dongcophai(255,0);  
}
```

```
void loop() {  
    hc05();  
}
```