**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**--🙢🕮🙠--**

A close up of a sign

Description automatically generated

**Báo cáo môn học: Hệ Nhúng**

**Đề tài: Xe dò đường kết hợp điều khiển bằng bluetooth**

**Giảng viên hướng dẫn**: TS. Ngô Lam Trung

**Họ tên** **Mssv**

Trần Văn Khoa 20173197

Đàm Việt Dũng 20173047

Nguyễn Xuân Cừ 20172985

*Hà Nội, Tháng 6-2020*

**Mục lục**

[**1. Mở đầu 2**](#_Toc45573422)

[**1.1 Giới thiệu 2**](#_Toc45573423)

[**1.2 Sơ lược các bước thực hiện 2**](#_Toc45573424)

[**1.3 Mục tiêu 2**](#_Toc45573425)

[**2. Yêu cầu chức năng 3**](#_Toc45573426)

[**2.1 Chức năng của sản phẩm: 3**](#_Toc45573427)

[**2.2 Ràng buộc 3**](#_Toc45573428)

[**3. Thiết kế phần cứng 3**](#_Toc45573429)

[**3.1 Lắp ráp 3**](#_Toc45573430)

[**3.1.1 Module motor driver shield L298 3**](#_Toc45573431)

[**3.1.2 Arduino Nano 5**](#_Toc45573432)

[**3.1.3 Cảm biến hồng ngoại 7**](#_Toc45573433)

[**3.1.4 Modul Bluetooth HC-05 8**](#_Toc45573434)

[**3.2 Sơ đồ mạch 9**](#_Toc45573435)

[**4. Thiết kế phần mềm 9**](#_Toc45573436)

[**5. Kết quả 10**](#_Toc45573437)

[**6. Tài liệu tham khảo 11**](#_Toc45573438)

**Yêu cầu của thầy giáo**

**Báo cáo project cuối kỳ:**

**- Mô tả bài toán: nhóm làm gì**

**- Yêu cầu chức năng: sản phẩm có tính năng gì, ràng buộc gì**

**- Thiết kế phần cứng**

**- Thiết kế phần mềm**

**- Kết quả + hạn chế (nếu có)**

# **1. Mở đầu**

## 1.1 Giới thiệu

Tự động hóa đang dần trở nên quen thuộc với con người, cũng như sự phổ biến của những thiết bị tự động ngày càng rộng. Tự động hóa có vai trò quan trọng và to lớn đối với sự phát triển công nghiệp hóa- hiện đại hóa như hiện nay. Từ việc giúp ích trong những dây chuyền sản xuất quy mô lớn, tới những robot ứng dụng trí thông minh,…

Việc chọn đề tài robot dò đường trong môn học này để có thể tìm hiểu và áp dụng những kiến thức đã học, tạo nên một sản phẩm có thể còn đơn giản, nhưng là bước đà để sau này có thể tìm hiểu những thứ phức tạp hơn trong lĩnh vực này.

## 1.2 Sơ lược các bước thực hiện

Trước tiên ta phải chế tạo được khung xe của robot. Khung xe phải đảm bảo bền chắc và đạt độ chính xác nhất định về việc bố trí các bánh xe và động cơ. Sau khi đã có xong mô hình thì việc tiếp theo là chọn vi xử lí, khối óc của robot, cũng như các mạch điện tử cần thiết cho việc điều khiển xe. Và cuối cùng là công đoạn lập trình dựa trên những kiến thức đã học được.

## 1.3 Mục tiêu

Các mục tiêu đặt ra cho sản phẩm:

- Robot đi theo vạch được chỉ định

- Có thể điều chỉnh tốc độ

- Hoạt động tốt và ổn định

# **2. Yêu cầu chức năng**

## 2.1 Chức năng của sản phẩm:

Sản phẩm có 2 chức năng chính tương ứng với 2 chế độ có thể chuyển qua lại trên giao diện ứng dụng.

- Có thể điều khiển bằng bluetooth thông qua ứng dụng trên di động hoặc trên PC

- Dò đường theo vạch đen nền trắng và vạch trắng nền đen

## 2.2 Ràng buộc

- Bản đồ rõ ràng giữa vạch và nền, vạch đen đủ để cảm biến có thể nhận biết được, được để trên nền phẳng không gập gềnh.

- Pin đủ khỏe để chạy ổn định trong thời gian trình bày

- Các module cảm biến, chip điều khiển hoạt động ổn định

# **3. Thiết kế phần cứng**

## 3.1 Lắp ráp

Các linh kiện sử dụng:

- Motor driver shield L298 (x1)

- Arduino nano (x1)

- Module thu phát hồng ngoại (x5)

- Động cơ, bánh xe (x2)

- Khung xe 3 bánh (x1)

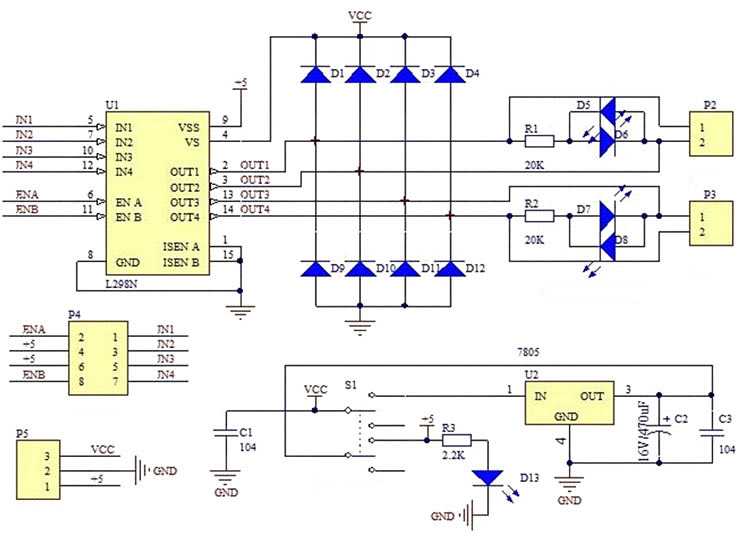
- Module thu phát bluetooth HC-05 (x1)

- Pin 18650 (x3)

-Dây cắm, ốc vít,…

### **3.1.1 Module motor driver shield L298**

Datasheet



Thông số kỹ thuật :

- Driver: L298N tích hợp hai mạch cầu H

- Điện áp điều khiển : +5V ~ +12 V

- Dòng tối đa cho mỗi cầu H là :2A

- Điện áp của tín hiệu điều khiển : +5 V ~ +7 V

- Dòng của tín hiệu điều khiển : 0 ~ 36Ma

- Công suất hao phí : 20W (khi nhiệt độ T = 75 °C)

- Nhiệt độ bảo quản : -25°C ~ +130

Mô tả:

- Module điều khiển motor L298N sử dụng IC điều khiển L298N có thể điều khiển 2 động cơ một chiều hoặc 1 động cơ bước 4 pha:

+ Được thiết kế chắc chắn, có sẵn chỗ bắt ốc vào mô hình.

+ Có gắn tản nhiệt chống nóng cho IC, giúp IC có thể điều khiển với dòng đỉnh đạt 2A.

+ IC L298N được gắn với các đi ốt trên board giúp bảo vệ vi xử lý chống lại các dòng điện cảm ứng từ việc khởi động/ tắt động cơ.

Các chân tín hiệu:

- Chân VCC dùng để cấp nguồn 12V cho Module

- Chân GND để cắm GND của nguồn vào

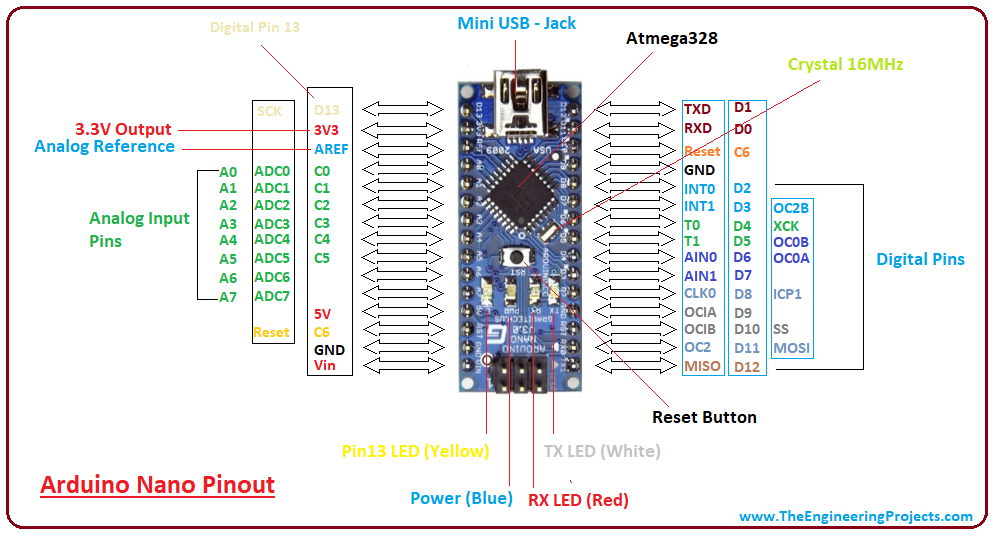
- Chân 5v là ngõ ra nguồn 5v

- Hai chân ENA và ENB là 2 chân cho phép động cơ hoạt động, có thể điều chỉnh tốc độ của động cơ bằng cách băm xung. Nếu nối vào nguồn 5v động cơ sẽ chạy với tốc độ tối đa

- Bốn chân INA , INB, INC, IND là chân input tín hiệu từ vi điều khiển dùng để điều khiển motor chạy thuận , nghịch hoặc dừng thông qua vi điều khiển

- OUTA, OUTB, OUTC, OUTD dùng nối với động cơ. Các chân này có thể điều khiển ở mức thấp hoặc cao thông qua các chân INPUT, từ đó có thể điều chỉnh động cơ theo ý muốn.

### **3.1.2 Arduino Nano**



Thông số kỹ thuật:

- Vi điều khiển ATmega328 (họ 8bit)

- Điện áp hoạt động: 5V – DC

- Tần số hoạt động: 16 MHz

- Dòng tiêu thụ: 30mA

- Điện áp vào khuyên dùng: 7-12V – DC

- Điện áp vào giới hạn: 6-20V – DC

- Số chân Digital I/O: 14 (6 chân PWM)

- Số chân Analog: 8 (độ phân giải 10bit)

- Dòng tối đa trên mỗi chân I/O: 40 mA

- Dòng ra tối đa (5V): 500 mA

- Dòng ra tối đa (3.3V): 50 mA

- Bộ nhớ flash 32 KB (ATmega328) với 2KB dùng bởi bootloader

- SRAM 2 KB (ATmega328)

- EEPROM 1 KB (ATmega328)

- Kích thước: 1.85cm x 4.3cm

Các chân trên arduino nano

- GND (Ground): cực âm của nguồn điện cấp cho Arduino NANO. Khi bạn dùng các thiết bị sử dụng những nguồn điện riêng biệt thì những chân này phải được nối với nhau.

- 5V: cấp điện áp 5V đầu ra. Dòng tối đa cho phép ở chân này là 500mA.

- 3.3V: cấp điện áp 3.3V đầu ra. Dòng tối đa cho phép ở chân này là 50mA.

- Vin (Voltage Input): để cấp nguồn ngoài cho Arduino NANO, bạn nối cực dương của nguồn với chân này và cực âm của nguồn với chân GND.

- IOREF: điện áp hoạt động của vi điều khiển trên Arduino NANO có thể được đo ở chân này. Và dĩ nhiên nó luôn là 5V. Mặc dù vậy bạn không được lấy nguồn 5V từ chân này để sử dụng bởi chức năng của nó không phải là cấp nguồn.

- RESET: việc nhấn nút Reset trên board để reset vi điều khiển tương đương với việc chân RESET được nối với GND qua 1 điện trở 10KΩ.

- 2 chân Serial: 0 (RX) và 1 (TX): dùng để gửi (transmit – TX) và nhận (receive – RX) dữ liệu TTL Serial. Arduino Nano có thể giao tiếp với thiết bị khác thông qua 2 chân này. Kết nối bluetooth thường thấy nói nôm na chính là kết nối Serial không dây. Nếu không cần giao tiếp Serial, bạn không nên sử dụng 2 chân này nếu không cần thiết

- Chân PWM (~): 3, 5, 6, 9, 10, và 11: cho phép bạn xuất ra xung PWM với độ phân giải 8bit (giá trị từ 0 → 28-1 tương ứng với 0V → 5V) bằng hàm analogWrite(). Nói một cách đơn giản, bạn có thể điều chỉnh được điện áp ra ở chân này từ mức 0V đến 5V thay vì chỉ cố định ở mức 0V và 5V như những chân khác.

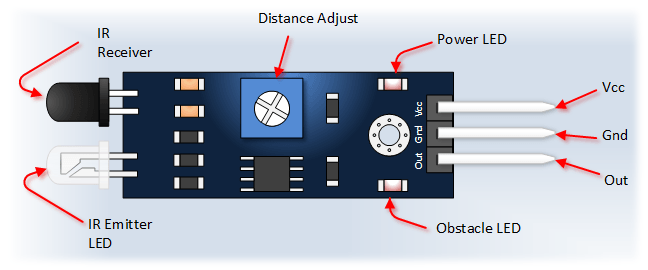
- Chân giao tiếp SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ngoài các chức năng thông thường, 4 chân này còn dùng để truyền phát dữ liệu bằng giao thức SPI với các thiết bị khác.

- LED 13: trên Arduino NANO có 1 đèn led màu cam (kí hiệu chữ L). Khi bấm nút Reset, bạn sẽ thấy đèn này nhấp nháy để báo hiệu. Nó được nối với chân số 13. Khi chân này được người dùng sử dụng, LED sẽ sáng.

- Arduino NANO có 6 chân analog (A0 → A5) cung cấp độ phân giải tín hiệu 10bit (0 → 210-1) để đọc giá trị điện áp trong khoảng 0V → 5V. Với chân AREF trên board, bạn có thể để đưa vào điện áp tham chiếu khi sử dụng các chân analog. Tức là nếu bạn cấp điện áp 2.5V vào chân này thì bạn có thể dùng các chân analog để đo điện áp trong khoảng từ 0V → 2.5V với độ phân giải vẫn là 10bit.

- Đặc biệt, Arduino NANO có 2 chân A4 (SDA) và A5 (SCL) hỗ trợ giao tiếp I2C/TWI với các thiết bị khác.

### **3.1.3 Cảm biến hồng ngoại**



Thông số kỹ thuật:

- Điện áp sử dụng: 3.3~5vDC

- Nhận biết vật cản bằng ánh sáng hồng ngoại.

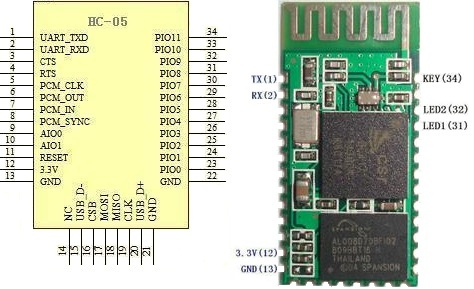
- Ngõ ra: Digital TTL

- Tích hợp biến trở chỉnh khoảng cách nhận biết vật cản.

- Kích thước: 3.2 x 1.4cm

Cảm biến vật cản hồng ngoại IR Infrared Obstacle Avoidance được sử dụng để nhận biết vật cản bằng ánh sáng hồng ngoại, cảm biến có cách sử dụng đơn giản với biến trở chỉnh khoảng cách nhận biết vật cản, ngõ ra dạng Digital dễ dàng giao tiếp và lập trình với Vi điều khiển, thích hợp để làm các ứng dụng Robot tránh vật cản, báo trộm, mô hình cửa tự động,...

### **3.1.4 Modul Bluetooth HC-05**



Thông số kỹ thuật:

Điện áp hoạt động: 3.6 ~ 6VDC

Mức điện áp chân giao tiếp: TTL tương thích 5VDC.

Dòng điện khi hoạt động: khi Pairing 30 mA, sau khi pairing hoạt động truyền nhận bình thường 8 mA.

Baudrate UART có thể chọn được: 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200

Support profiles: Bluetooth serial port (master and slave)

Bluetooth protocol: Bluetooth specification v2.0 + EDR

Frequency: 2.4 GHz ISM band

Modulation: GFSK (Gaussian frequency shift keying)

Transmit power: =4 dBm, class 2

Sensitivity: =-84 dBm at 0.1% BER

Rate: Asynchronous: 2.1 Mbps (max.)/160 kbps

Synchronous: 1 Mbps/1 Mbps

Security features: authentication and encryption

Kích thước: 15.2 x 35.7 x 5.6mm

Thiết lập mặc định:

Thiết lập UART mặc định: Baudrate 9600.

[**Module thu phát Bluetooth HC-05**](https://nshopvn.com/product/module-thu-phat-bluetooth-hc-05/) dùng để thiết lập kết nối Serial giữa 2 thiết bị bằng sóng bluetooth.

Các chế độ hoạt động

HC-05 có hai chế độ hoạt động là Command Mode và Data Mode. Ở chế độ Command Mode ta có thể giao tiếp với module thông qua cổng serial trên module bằng tập lệnh AT quen thuộc. Ở chế độ Data Mode module có thể truyền nhận dữ liệu tới module bluetooth khác. Chân KEY dùng để chuyển đổi qua lại giữa hai chế độ này. Có hai cách để bạn có thể chuyển module hoạt động trong chế độ Data Mode.

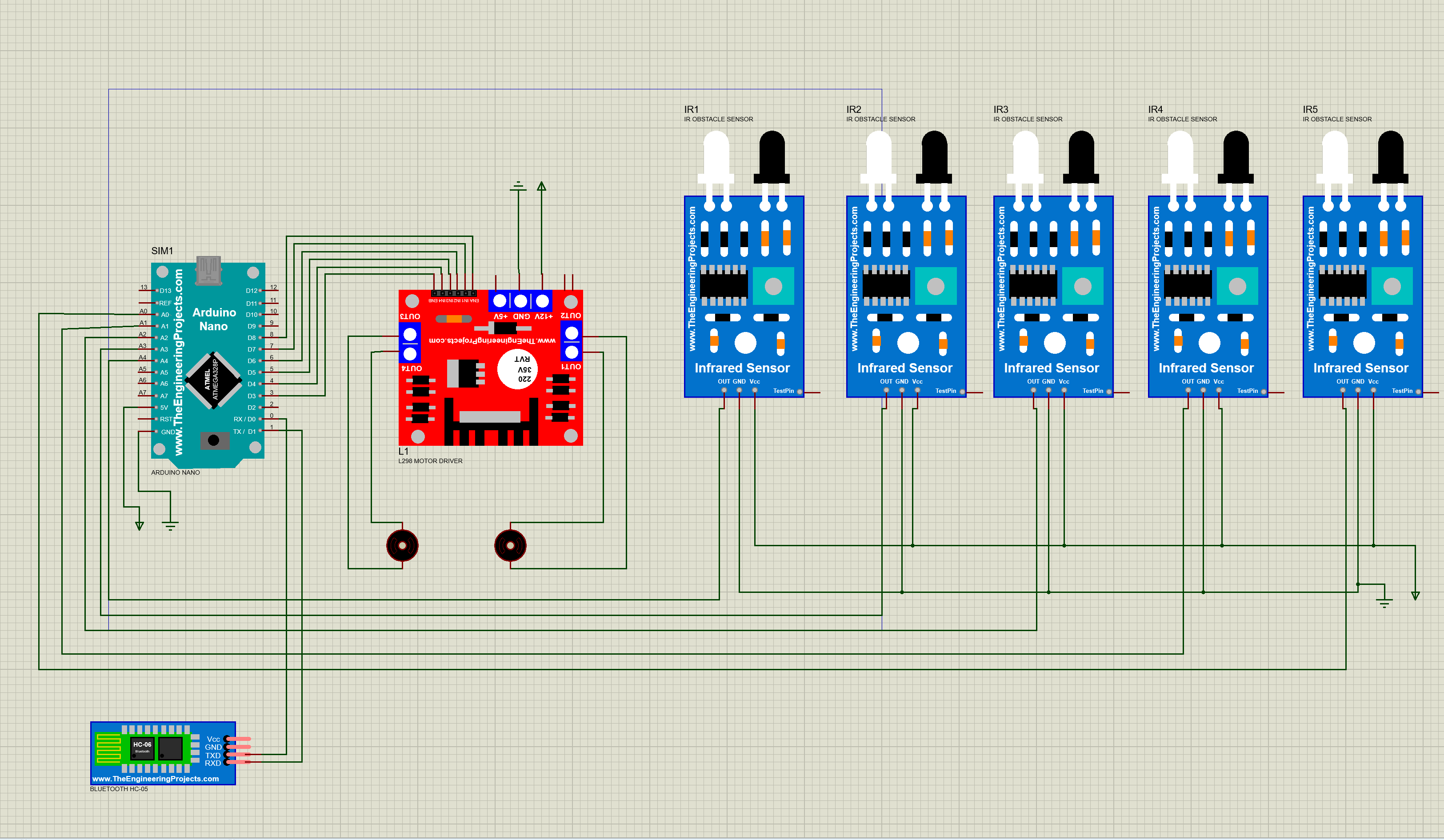
Ở đây chúng ta sử dụng chế độ hoạt động Command Mode , với thiết lập mặc định là

· Baudrate 9600, data 8 bits, stop bits 1, parity : none, handshake: none

· Passkey: 1234

· Device Name: HC-05

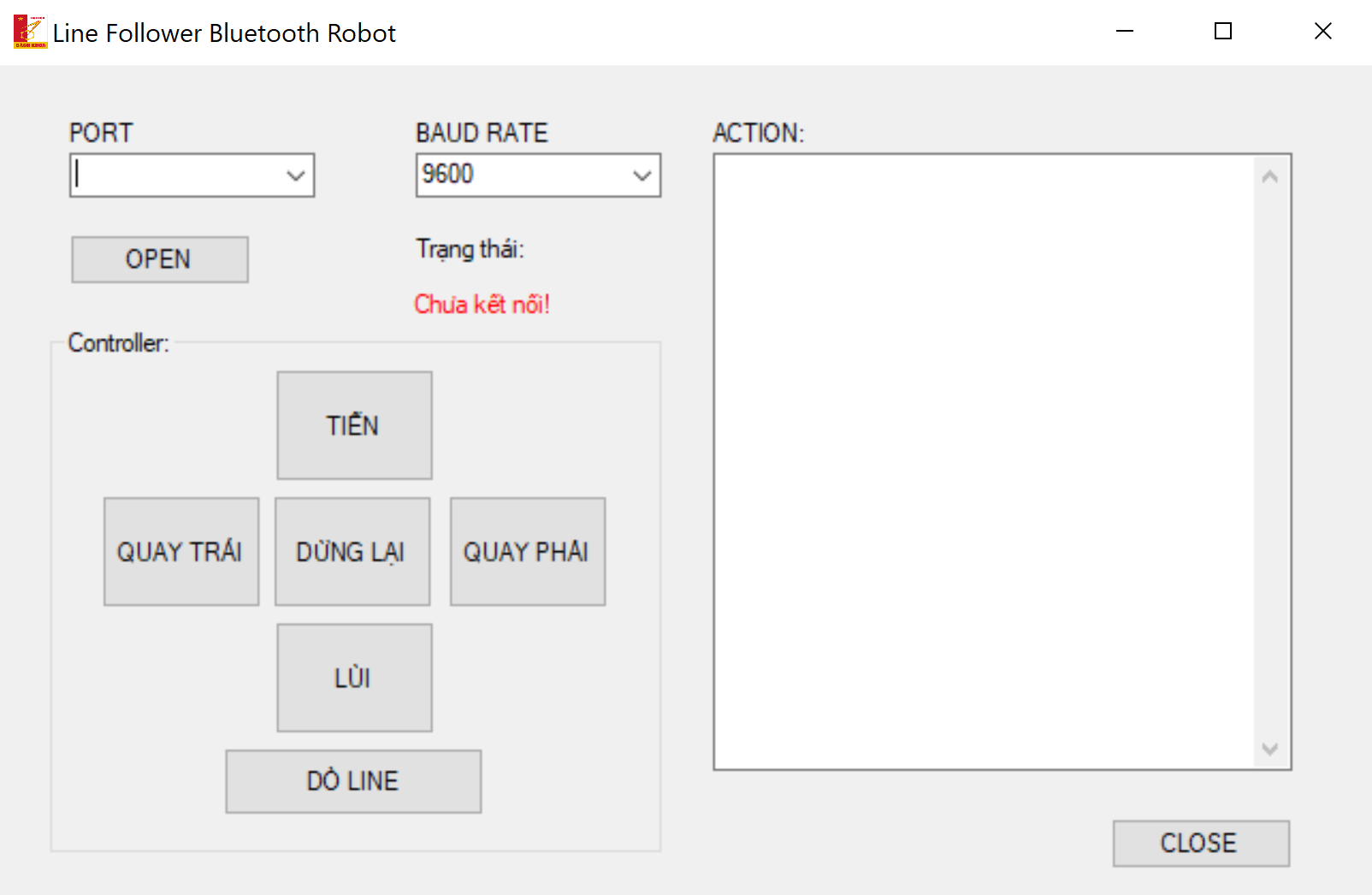
## 3.2 Sơ đồ mạch

****

# **4. Thiết kế phần mềm**

Phần mềm điều khiển:

Được lập trình bằng C++ winform trên Visual có giao diện như sau:



Phần mềm được viết trên ngôn ngữ C++ winform, có chức năng điều khiển xe chạy thông qua bluetooth

Nguyên lý hoạt động:

Port là cổng COM dùng để kết nối module bluetooth, với baud rate mặc định của module hc-05 là 9600. Nếu kết nối thành công sẽ hiển thị thông báo kết nối thành công.

Khi điều khiển sẽ hiển thị trạng thái của xe lên ô ACTION.

Khi ấn vào các nút TIẾN, LÙI, TRÁI, PHẢI sử dụng chế độ mousedown và mouseup để gửi dữ liệu , nên khi ấn giữ xe sẽ chạy còn buông thì sẽ tự dừng ( tránh trường hợp ấn 1 nút sẽ tự chạy, đến khi ấn nút khác thì mới đổi trạng thái sẽ rất khó điều khiển )

Đối với nút DÒ LINE sử dụng chế độ click, gửi tín hiệu đến module bluetooth, arduino đọc tín hiệu và thực hiện chuyển chế độ khi nút được click 1 lần mà không cần giữ.

(tham khảo thêm về chế độ mouseup, mousedown, click trong phần tài liệu tham khảo)

# **5. Kết quả**

- Xe có thể điều khiển bằng bluetooth ổn định, dò đường ổn định nếu đảm bảo ràng buộc

- Nhóm đã tìm hiểu được nguyên lý hoạt động của một số loại module, có thể ứng dụng trong những project đơn giản , từ đó có kiến thức hơn trong học tập và thực hành

Hạn chế : Có thể xe chạy sai khi dò đường do một số lý do nào đấy, tuy nhiên khả năng này rất ít

# **6. Tài liệu tham khảo**

<https://linhkienst.com/blogs/huong-dan-su-dung-cac-module/1000125965-module-bluetooth-hc-05-phan-1>

<https://www.arduino.cc/>

<https://social.msdn.microsoft.com/Forums/en-US/a9529502-6304-4aa6-90ee-0757ab258d87/create-c-windows-forms-application-in-visual-studio-2017?forum=winforms>

<https://www.quirksmode.org/dom/events/click.html>