

TRƯỜNG ĐH CÔNG NGHIỆP THỰC PHẨM TP. HCM

KHOA ĐIỆN - ĐIỆN TỬ

BỘ MÔN: ĐỒ ÁN HỌC PHẦN 2



ĐOÀN PHƯƠNG NAM

ĐỒ ÁN HỌC PHẦN 2

MÔ HÌNH XE ĐIỀU KHIỂN TỪ XA QUA BLUETOOTH

GVHD: TRẦN HOÀN

TP. HỒ CHÍ MINH, tháng 6 năm 2018

TRƯỜNG ĐH CÔNG NGHIỆP THỰC PHẨM TP. HCM

KHOA ĐIỆN - ĐIỆN TỬ

BỘ MÔN: ĐỒ ÁN HỌC PHẦN 2



ĐOÀN PHƯƠNG NAM

ĐỒ ÁN HỌC PHẦN 2

MÔ HÌNH XE ĐIỀU KHIỂN TỪ XA QUA BLUETOOTH

GVHD: TRẦN HOÀN

TP. HỒ CHÍ MINH, tháng 6 năm 2018

TP. HCM, ngày 12 tháng 6 năm 2018

NHẬN XÉT ĐỒ ÁN HỌC PHẦN 2 CỦA GIẢNG VIÊN HƯỚNG DẪN

Tên đồ án:

MÔ HÌNH XE ĐIỀU KHIỂN TỪ XA QUA BLUETOOTH

Sinh viên thực hiện:

Đoàn Phương Nam

2002150267

Giảng viên hướng dẫn:

Trần Hoàn

Đánh giá Đồ án

1. Về cuốn báo cáo:

Số trang _____ Số chương _____

Số bảng số liệu _____ Số hình vẽ _____

Số tài liệu tham khảo _____ Sản phẩm _____

Một số nhận xét về hình thức cuốn báo cáo:

2. Về nội dung đồ án:

3. Về tính ứng dụng:

4. Về thái độ làm việc của sinh viên:

Đánh giá chung:

Điểm từng sinh viên:

Đoàn Phương Nam:...../10

Người nhận xét

LỜI CẢM ƠN

Em xin cảm ơn quý thầy, cô trường Đại học Công nghiệp Thực Phẩm TP. HCM đã tận tình dạy dỗ trong suốt thời gian qua. Trong đó phải kể đến quý thầy, cô Khoa Công Nghệ Điện Tử đã tạo điều kiện cho chúng em thực hiện đồ án 2 này.

Em xin cảm ơn giáo viên hướng dẫn, thầy Trần Hoàn đã tận tình giúp đỡ em trong quá trình chọn và thực hiện đề tài. Với thời gian thực hiện đề tài ngắn, kiến thức hạn hẹp, dù em cố gắng nhưng không thể thiếu những sai sót, em rất mong nhận được sự chỉ dẫn thêm của thầy.

Em xin chân thành cảm ơn.

TP. Hồ Chí Minh, ngày 12 tháng 6 năm 2018

Tác giả

Đoàn Phương Nam

TP. HCM, ngày 12 tháng 6 năm 2018

ĐỀ CƯƠNG CHI TIẾT

TÊN ĐỒ ÁN: MÔ HÌNH XE ĐIỀU KHIỂN TỪ XA QUA BLUETOOTH	
Giảng viên hướng dẫn: TRẦN HOÀN	
Thời gian thực hiện: Từ ngày 19/3/2018 đến ngày 13/6/2018	
Sinh viên thực hiện: Đoàn Phương Nam	
Nội dung đề tài: <ul style="list-style-type: none">- Thiết kế mô hình xe điều khiển qua bluetooth	
Kế hoạch thực hiện: <ul style="list-style-type: none">- Từ ngày 21/3/2018 đến ngày 28/3/2018: Tìm hiểu về đề tài- Từ ngày 28/3/2018 đến ngày 1/4/2018: Chuẩn bị phần cứng- Từ ngày 1/4/2018 đến ngày 15/5/2018: Hoàn thiện phần cứng và chương trình Arduino, chạy thử trên app điện thoại- Từ ngày 15/5/2018: Nghiên cứu về cách thiết kế chương trình điều khiển	
Xác nhận của giảng viên hướng dẫn	TP. HCM, ngày 12 tháng 6 năm 2018 Sinh viên

Contents

DANH MỤC KÝ HIỆU, CỤM TỪ VIẾT TẮT	iii
DANH MỤC BẢNG BIỂU	iv
DANH MỤC HÌNH ẢNH.....	v
CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI	1
1.1 Đặt vấn đề	1
1.2 Mục tiêu đề tài	1
CHƯƠNG 2: THÔNG TIN LINH KIỆN	3
2.1 Arduino Uno R3	3
2.1.1 Định nghĩa về Arduino	3
2.1.2 Các dòng Arduino đã ra đời	3
2.1.3 Phần mềm IDE	4
2.1.4 Arduino UNO R3	5
2.2 Arduino Motor Shield L293D	8
2.3 Module Bluetooth HC-06	10
2.4 Động cơ DC giảm tốc	12
2.5 Nguồn cấp.....	14
Một số hãng sản xuất pin sạc Li-ion lớn:	15
2.6 Khung xe	16
Tìm hiểu về phần mềm SketchUp.....	16
Một vài đặc điểm nổi bật:	16
CHƯƠNG 3: CƠ SỞ THỰC HIỆN	18
3.1 Thiết kế phần cứng	18
3.1.1 Khối xử lí	19
3.1.2. Thiết kế khung xe.....	19
3.2: Thiết kế phần mềm	20
3.2.1: Phần mềm trên Androi	20
3.3 Lưu đồ thuật toán.....	21

3.3.1: Thuật toán khối xử lý trung tâm Arduino	21
3.3.2: Thuật toán phần mềm điều khiển	22
CHƯƠNG 4: KẾT QUẢ THỰC NGHIỆM.....	23
CHƯƠNG 5: KẾT LUẬN VÀ ĐỊNH HƯỚNG ĐỀ TÀI.....	26
5.1 Kết quả đạt được:	26
5.2 Hạn Chế :	26
5.3 Hướng Phát Triển:	26
PHỤ LỤC.....	27
Code chương trình:	27
TÀI LIỆU THAM KHẢO.....	31

DANH MỤC KÝ HIỆU, CỤM TỪ VIẾT TẮT

KÝ HIỆU	THUẬT NGỮ
VCC	Nguồn Cấp
GND	Ground (nối đất)
Vin	Voltage Input
IC	Integrated Circuit (Chip)
UART	Universal Asynchronous Receiver/Transmitter

DANH MỤC BẢNG BIỂU

Bảng 1: Hướng dẫn nối dây.....	22
Bảng 2: Thông số kỹ thuật của pin 18650.....	26

DANH MỤC HÌNH ẢNH

Hình 2.1: Giao diện IDE Arduino	16
Hình 2.2: Mạch Arduino Uno R3.....	17
Hình 2.3: Arduino Moto Shield L293D	21
Hình 2.4: Module Bluetooth HC06.....	22
Hình 2.5: Motor DC giảm tốc	23
Hình 2.6: Cell Pin 18650.....	24
Hình 3.1: Kết nối Arduino với HC06.....	31
Hình 3.2: Khung xe	31
Hình 3.3: Giao diện điều khiển	32

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1 Đặt vấn đề

Ngày nay, hệ thống điều khiển đóng vai trò quan trọng trong sự phát triển và tiến bộ của khoa học kỹ thuật công nghệ, văn minh hiện đại. Trong thực tế, mỗi khía cạnh của hoạt động hàng ngày đều bị chi phối bởi một vài loại hệ thống điều khiển. Ta dễ dàng tìm thấy hệ thống điều khiển máy, công cụ, kỹ thuật không gian, hệ thống vũ khí, điều khiển máy tính, các hệ thống giao thông, hệ thống năng lượng.

Trong sinh hoạt hàng ngày của con người như những trò chơi giải trí (robot, xe điều khiển từ xa, ..) cho đến những ứng dụng gần gũi với con người cũng được cải tiến cho phù hợp với việc sử dụng và đạt mức tiện lợi nhất. Việc điều khiển từ xa đã thâm nhập vào tất cả các lĩnh vực của cuộc sống.

Chính vì tầm quan trọng đó, em đã lựa chọn đề tài “Thiết kế mô hình xe điều khiển từ xa qua bluetooth”

1.2 Mục tiêu đề tài

Với sự phát triển ngày càng mạnh mẽ của khoa học công nghệ, vi điều khiển ngày càng thông dụng và hoàn thiện hơn, nhưng có thể nói sự xuất hiện của Arduino vào năm 2005 tại Italia đã mở ra một hướng đi mới cho vi điều khiển. Sự xuất hiện của Arduino đã hỗ trợ cho con người rất nhiều trong lập trình và thiết kế, mà không cần quá nhiều kiến thức về vi xử lý. Phần cứng của thiết bị đã được tích hợp nhiều chức năng cơ bản và là mã nguồn mở. Ngôn ngữ lập trình trên nền Java lại vô cùng dễ sử dụng tương thích với ngôn ngữ C và hệ thư viện rất phong phú và được chia sẻ miễn phí. Chính vì những lý do như vậy nên Arduino hiện đang dần phổ biến và được phát triển ngày càng mạnh mẽ trên toàn thế giới. Điện tử đang trở thành một ngành khoa học đa

nhiệm vụ . Nó đã đáp ứng được những nhu cầu cần thiết trong hoạt động đời sống hằng ngày. Để có thể tìm hiểu sâu hơn về chức năng, ứng dụng của Arduino và công nghệ Bluetooth, em đã chọn đề tài “ Thiết kế mô hình xe điều khiển từ xa qua bluetooth “ là đồ án học phần 2.

CHƯƠNG 2: THÔNG TIN LINH KIỆN

2.1 Arduino Uno R3

2.1.1 Định nghĩa về Arduino

Arduino là một board mạch vi xử lý, nhằm xây dựng các ứng dụng tương tác với nhau hoặc với môi trường được thuận lợi hơn. Phần cứng bao gồm một board mạch nguồn mở được thiết kế trên nền tảng vi xử lý AVR Atmel 8bit, hoặc ARM Atmel 32-bit. Những Model hiện tại được trang bị gồm 1 cổng giao tiếp USB, 6 chân đầu vào analog, 14 chân I/O kỹ thuật số tương thích với nhiều board mở rộng khác nhau.

2.1.2: Các dòng Arduino đã ra đời

Dựa vào mục đích sử dụng mà các dòng Arduino được chia ra với nhiều thiết kế khác nhau với kết cấu phần cứng khác nhau để thực hiện tốt mục tiêu sử dụng hướng tới. Ở đây chúng tôi sẽ liệt kê các dòng Arduino hiện hành mà không đi vào chi tiết của từng dòng board Arduino, chỉ tập trung vào phân tích phần cứng của Arduino UNO R3 vì đây là board mà đồ án hướng tới sử dụng.

Các loại Arduino:

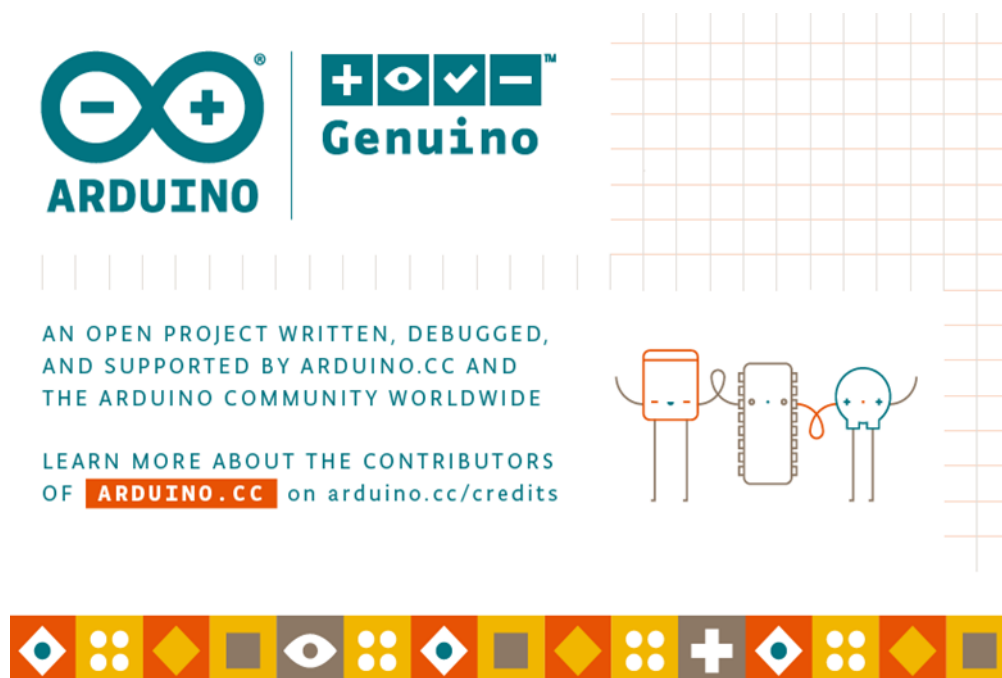
- +Arduino Diecimila
- +Arduino Duemilanove
- +Arduino UNO
- +Arduino Leonardo
- +Arduino Mega
- +Arduino Nano
- + ...

2.1.3 Phần mềm IDE

Môi trường phát triển tích hợp (IDE) của Arduino là một ứng dụng cross-platform (nền tảng) được viết bằng Java, và từ IDE này sẽ được sử dụng cho Ngôn ngữ lập trình xử lý (Processing programming language) và project Wiring. Nó được thiết kế để dành cho các nghệ sĩ và những người mới tập tành làm quen với lĩnh vực phát triển phần mềm. Nó bao gồm một chương trình code editor với các chức năng như đánh dấu cú pháp, tự động brace matching, và tự động canh lề, cũng như compile(biên dịch) và upload chương trình lên board chỉ với 1 cú click chuột. Một chương trình hoặc code viết cho Arduino được gọi là một sketch.

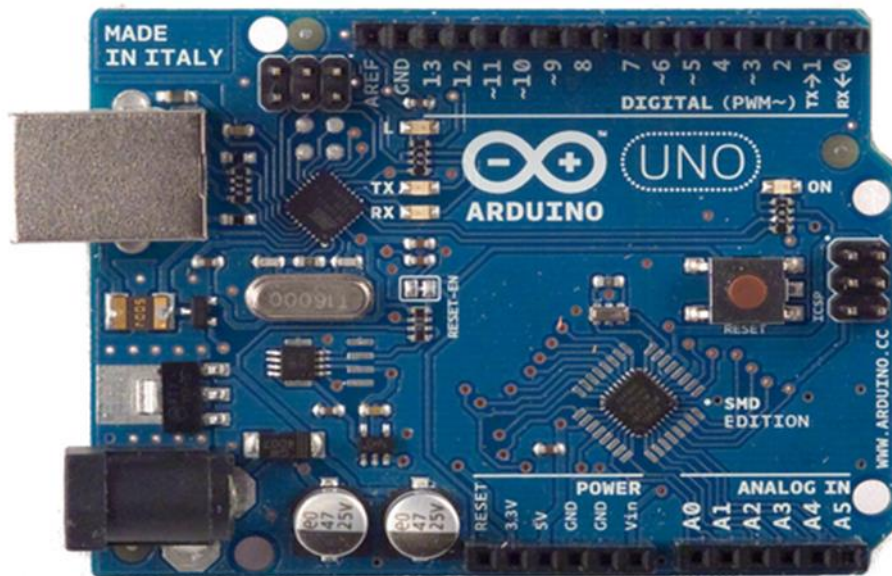
Các chương trình Arduino được viết bằng C hoặc C++. Arduino IDE đi kèm với một thư viện phần mềm được gọi là "Wiring", từ project Wiring gốc, có thể

giúp các thao tác input/output được dễ dàng hơn. Người dùng chỉ cần định nghĩa 2 hàm để tạo ra một chương trình vòng thực thi (cyclic executive) có thể chạy được



2.1.4 Arduino UNO R3

Nhắc tới dòng mạch Arduino dùng để lập trình, đầu tiên mà người ta thường nói tới chính là dòng Arduino UNO. Hiện dòng mạch này đã phát triển tới thế hệ thứ 3 (R3).



Hình 2.2: Mạch Arduino Uno R3

Arduino UNO có thể sử dụng 3 vi điều khiển họ 8bit AVR là ATmega8, ATmega168, ATmega328. Bộ não này có thể xử lý những tác vụ đơn giản như điều khiển đèn LED nhấp nháy, xử lý tín hiệu cho xe điều khiển từ xa, làm một trạm đo nhiệt độ - độ ẩm và hiển thị lên màn hình LCD,... hay những ứng dụng khác như trong đồ án này là thực hiện một mạch khóa số điện tử.

Về mặt năng lượng, Arduino UNO có thể được cấp nguồn 5V thông qua cổng USB hoặc cấp nguồn ngoài với điện áp khuyến dùng là 7-12V DC và giới hạn là 6-20V. Thường thì cấp nguồn bằng pin vuông 9V là hợp lý nhất nếu không có sẵn nguồn từ cổng USB. Nếu cấp nguồn vượt quá ngưỡng giới hạn trên, sẽ làm hỏng Arduino UNO.

Các chân năng lượng:

- GND (Ground): cực âm của nguồn điện cấp cho Arduino UNO. Khi 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 UNO, 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 UNO có thể được đo ở chân này. Và dĩ nhiên nó luôn là 5V. Mặc dù vậy, 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.

Các chân 3.3V và 5V trên Arduino là các chân dùng để cấp nguồn ra cho các thiết bị khác, không phải là các chân cấp nguồn vào. Việc cấp nguồn sai vị trí có thể làm hỏng board. Điều này không được nhà sản xuất khuyến khích.

Cường độ dòng điện vào/ra ở tất cả các chân Digital và Analog của Arduino UNO nếu vượt quá 200mA sẽ làm hỏng vi điều khiển.

Cấp điện áp trên 5.5V vào các chân Digital hoặc Analog của Arduino UNO sẽ làm hỏng vi điều khiển.

Cường độ dòng điện qua một chân Digital hoặc Analog bất kỳ của Arduino UNO vượt quá 40mA sẽ làm hỏng vi điều khiển. Do đó nếu không dùng để truyền nhận dữ liệu, phải mắc một điện trở hạn dòng.

Về bộ nhớ : vi điều khiển Atmega328 tiêu chuẩn cung cấp cho người dùng 32KB bộ nhớ Flash, những đoạn lệnh lập trình sẽ được lưu trữ trong bộ nhớ Flash của vi điều khiển. Thường thì sẽ có khoảng vài KB trong số này sẽ được dùng cho bootloader nhưng hiếm khi nào cần quá 20KB bộ nhớ này.

2KB cho SRAM (Static Random Access Memory): giá trị các biến khai báo khi lập trình sẽ lưu ở đây. Khai báo càng nhiều biến thì càng cần nhiều bộ nhớ RAM. Tuy vậy, thực sự thì cũng hiếm khi nào bộ nhớ RAM lại trở thành mối lo lắng của người dùng. Khi mất điện, dữ liệu trên SRAM sẽ bị mất.

Ngoài ra , Arduino UNO có 14 chân digital dùng để đọc hoặc xuất tín hiệu. Chúng chỉ có 2 mức điện áp là 0V và 5V với dòng vào/ra tối đa trên mỗi chân là 40mA. Ở mỗi chân đều có các điện trở pull-up từ được cài đặt ngay trong vi điều khiển ATmega328 (mặc định thì các điện trở này không được kết nối).

Một số chân digital có các chức năng đặc biệt như sau : 2 chân Serial là 0 (RX) và 1 (TX) dùng để gửi (transmit – TX) và nhận (receive – RX) dữ liệu TTL Serial. Arduino Uno 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, không nên sử dụng 2 chân này nếu không cần thiết.

Chân PWM (~): 3, 5, 6, 9, 10, và 11: cho phép xuất ra xung PWM với độ phân giải 8bit (giá trị từ 0 → 255 tương ứng với 0V → 5V) bằng hàm analogWrite(). Nói một cách đơn giản, 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.

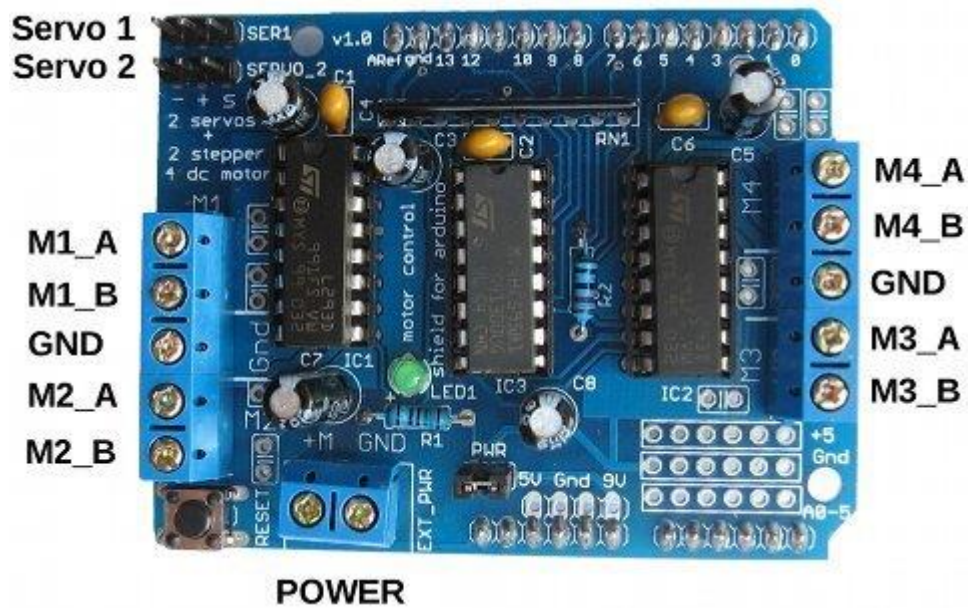
Chân giao tiếp SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ngoài các chức năng thông thường, 4 chân này còn dùng để truyền phát dữ liệu bằng giao thức SPI với các thiết bị khác.

LED 13: trên Arduino UNO có 1 đèn led màu cam (kí hiệu chữ L). Khi bấm nút Reset, 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 UNO có 6 chân analog (A0 \rightarrow A5) cung cấp độ phân giải tín hiệu 10bit (0 \rightarrow 2¹⁰-1) để đọc giá trị điện áp trong khoảng 0V \rightarrow 5V. Với chân AREF trên board, có thể để đưa vào điện áp tham chiếu khi sử dụng các chân analog. Tức là nếu cấp điện áp 2.5V vào chân này thì có thể dùng các chân analog để đo điện áp trong khoảng từ 0V \rightarrow 2.5V với độ phân giải vẫn là 10bit.

2.2 Arduino Motor Shield L293D

Arduino Motor Shield là một trong những Shield nổi bật nhất và phổ biến nhất hỗ trợ cho họ hàng nhà Arduino. Shield tương thích tốt với board Arduino Uno R3, Arduino Mega 2560, Arduino Leonardo. Motor Shield sử dụng 2 IC cầu H L293D và 1 IC 74HC595 điều khiển. Do đó, Motor Shield có thể điều khiển nhiều loại động cơ khác nhau với mức áp lên đến 36V, dòng tối đa 600mA mỗi kênh điều khiển



Hình 2.3: Arduino Moto Shield L293D

Các thành phần motor shield có thể điều khiển được như sau:

+ 2 servo

+ 4 motor điện DC (M1, M2, M3, M4) theo các hướng khác nhau, mỗi motor có thể chạy với 1 hướng tới/lùi bất kì thông qua 4 cầu H của L293D dòng 600mA

+ 2 stepper motor loại đơn cực hoặc lưỡng cực

Motor Shield có sẵn các điện trở nối GND nhằm tránh các động cơ tự quay khi khởi động.

Có sẵn nút Reset để khởi động lại board Arduino.

Các chân trên Shield được kết nối với board Arduino như sau:

+ 2 dây điều khiển 2 servo kết nối với chân số 9 và 10. Nguồn nuôi lấy trực tiếp từ board Arduino.

+ Motor 1 nối với chân 11

Motor 2 nối với chân 3

Motor 3 nối với chân 5

Motor 4 nối với chân 6

Chân 4, 7, 8, 12 dùng điều khiển motor thông qua IC 74HC595

+ Các chân chưa sử dụng: 2, 13, A0, A1, A2, A3, A4, A5

+ Trên shield có 1 jumper màu vàng PWR được sử dụng: nếu kết nối nguồn ngoài cho board Arduino thông qua jack DC (dùng pin 9V chẳng hạn) thì nguồn nuôi motor được lấy luôn từ jack này, không cần nối nguồn với EXT-PWR. Nếu ngắt jumper, cần nối 1 nguồn riêng vào terminal EXT-PWR để nuôi motor.

2.3 Module Bluetooth HC-06

Bluetooth module SLAVE cho phép vi điều khiển kết nối với các thiết bị ngoại vi: smartphone, laptop, usb bluetooth... thông qua giao tiếp Serial gửi và nhận tín hiệu 2 chiều.



Hình 2.4. Module Bluetooth HC06

Module bluetooth được tích hợp trên board cho phép bạn sử dụng nguồn từ 3.5V đến 6V cung cấp cho board mà không cần lo lắng về chênh lệch điện áp 3V - 5V gây hỏng board.

Bluetooth module gồm 6 chân theo thứ tự: KEY, VCC, GND, TX, RX, STATE.

Đây là module bluetooth SLAVE nghĩa là bạn không thể chủ động kết nối bằng vi điều khiển, mà cần sử dụng smartphone, laptop, bluetooth usb... để dò tín hiệu và kết nối (pair) từ smartphone, laptop, bluetooth usb... Sau khi pair thành công, có thể gửi và nhận tín hiệu từ vi điều khiển đến các thiết bị này.

Hướng dẫn nối dây

Bluetooth module	Arduino
VCC	5V
GND	GND
TX	RX
RX	TX

Thông số kỹ thuật:

- Điện áp hoạt động: 3.3 ~ 5VDC
- Điện áp giao tiếp: TTL tương thích 3.3VDC và 5VDC
- Baudrate UART có thể chọn được: 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
- Sử dụng CSR mainstream bluetooth chip, bluetooth V2.0 protocol standards.
- Dòng điện khi hoạt động: khi Pairing 30 mA, sau khi pairing hoạt động truyền nhận bình thường 8 mA.
- Kích thước của module chính: 28 mm x 15 mm x 2.35 mm
- Dải tần sóng hoạt động: Bluetooth 2.4GHz

Thiết lập mặc định

Baud rate: 9600, N, 8, 1

Pairing code: 1234

Tính năng:

- + Thu phát không dây
 - Độ nhạy (Tốc độ lỗi bit) có thể đạt tới -80dBm.
 - Phạm vi thay đổi của công suất đầu ra: -4 - + 6dBm
- + Mô tả chức năng
 - Có một mô-đun EDR và phạm vi thay đổi của độ sâu điều chế: 2Mbps - 3Mbps.
 - Có ăng-ten 2.4GHz tích hợp, người dùng không cần ăng-ten kiểm tra.
 - Có FLASH 8Mbit bên ngoài.
 - Có thể làm việc ở điện áp thấp (3.1V ~ 4.2V).
 - Khi Pairing 30 mA, sau khi pairing hoạt động truyền nhận bình thường 8 mA
 - Cổng HCI tiêu chuẩn (UART hoặc USB)
 - Giao thức USB: USB1.1 tốc độ cao, tuân thủ 2.0
 - Mô-đun này có thể được sử dụng trong SMD.
 - Nó được thực hiện thông qua quá trình RoHS
- + Sự tiêu thụ ít điện năng
- + Có hệ thống thu phát không dây hiệu suất cao
- + Giá thấp

2.4 Động cơ DC giảm tốc

Động cơ giảm tốc V1 là loại động cơ DC phổ thông được sử dụng nhiều nhất nhờ sự tiện dụng, giá thành rẻ. Thiết kế rất phù hợp cho các ứng dụng xe robot, kết hợp với các loại bánh xe thông dụng



Hình 2.5: Motor DC giảm tốc

Thông số kỹ thuật

- Điện áp hoạt động : 3~9 VDC
- Dòng điện tiêu thụ: 110 - 140mA
- Tỉ số truyền: 1:48
- Số vòng/phút:
 - + 125 vòng/ 1 phút tại 3 VDC.
 - + 208 vòng/ 1 phút tại 5 VDC.

2.5 Nguồn cấp

Cell Pin 18650



Hình 2.6 : Cell Pin 18650

Ý nghĩa của con số 18650 là gì?

- Hai số đầu là đường kính viên pin: ở đây là **18mm**
- Hai số tiếp theo là chiều dài viên pin, khoảng: **65mm**
- Số **0** cuối cùng để chỉ viên pin có hình trụ

Tuổi thọ của pin 18650 phụ thuộc vào số lần sạc. Nói chung nó chịu được 500 lần xả và sạc đầy. Tuy nhiên con số này chỉ là tương đối, nó còn phụ thuộc nhiều vào nhiệt độ hoạt động của pin, chất lượng bộ sạc pin. Ví dụ

như pin dùng trong các sản phẩm laptop chơi game thường bị “nung nóng” bởi laptop với card đồ họa chuyên dụng, nó không thể bền bằng viên pin sử dụng trong laptop thường. Pin được sạc bằng bộ sạc tốt cũng sẽ bền hơn pin sử dụng cục sạc kém chất lượng.

Một số hãng sản xuất pin sạc Li-ion lớn:

Có nhiều hãng sản xuất pin sạc 18650, tuy nhiên phải kể ra những hãng lớn như: Panasonic (Sanyo), LG chemical, Samsung SDI, Sony Energy Devices.

Top các nhà sản xuất Trung Quốc có: Tianjin Lishen Battery, SHENZHEN BAK BATTERY (hãng này có liên hệ mật thiết với gã khổng lồ Samsung), Pulead Technology, Beijing Easpring Material Technology.

Với pin dùng cần chú ý:

- Nếu dùng pin cell nên dùng của hãng nổi tiếng như Panasonic, Samsung, LG...
- Không để pin ở nơi nóng, có ánh sáng mặt trời chiếu thẳng hoặc ở nơi nhiệt độ cao (như trong xe hơi dưới trời nắng) hoặc nơi quá lạnh. Nhiệt độ lý tưởng khoảng 20-25 độ
- Pin để lâu sẽ tự xả. Nên sạc pin khi để lâu một thời gian dài không dùng.
- Không nên để pin quá lâu mà không dùng, khoảng 3-6 tháng nên lôi pin ra sạc một lần
- Để xa tầm tay trẻ em
- Nên sạc với bộ sạc tốt, có tự ngắt. Khi sạc, hãy để pin và bộ sạc cách xa

Thông số kỹ thuật của pin 18650

3. Nominal Specifications

Item	Specification
3.1 Typical Capacity	3,200mAh (0.2C, 2.50V discharge)
3.2 Minimum Capacity	3,100mAh (0.2C, 2.50V discharge)
3.3 Charging Voltage	4.2V
3.4 Nominal Voltage	3.65V (0.2C discharge)
3.5 Charging Method	CC-CV (constant voltage with limited current)
3.6 Charging Current	Standard charge: 960mA

2.6 Khung xe

Khung xe làm từ nhựa hoặc mica tùy theo sở thích và phong cách người thiết kế. Chúng ta có thể sử dụng phần mềm Sketchup hoặc AutoCad để thiết kế khung xe, sau đó đem file ra các chỗ cắt laser và cắt khung. Ở đây em sử dụng phần mềm SketchUp để thiết kế khung xe.

Tìm hiểu về phần mềm SketchUp

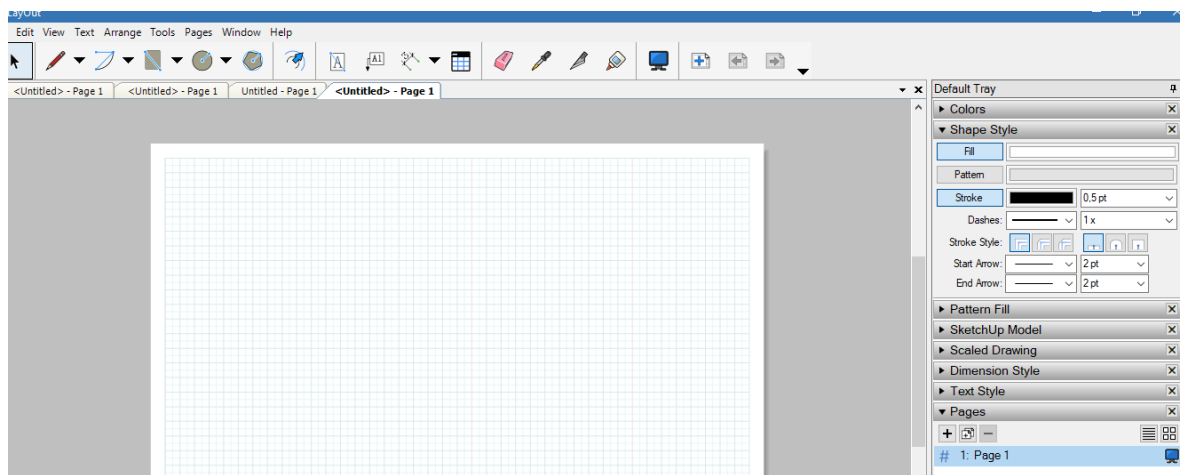
Là một phần mềm mô hình hóa 3D, dành cho các kiến trúc sư, các kỹ sư, nhà phát triển trò chơi điện tử, các đạo diễn điện ảnh và các ngành nghề có liên quan. Phần mềm này khởi đầu được phát triển bởi công ty @Last Software, có trụ sở tại Boulder, Colorado, Mỹ. Phần mềm này nổi bật như một công cụ diễn tả ý tưởng đơn giản, nhanh gọn với giao diện đồ họa cho người sử dụng.

Một vài đặc điểm nổi bật:

- Không cần phần cứng mạnh như các phần mềm mô hình hóa khác như 3D Max, FormZ, Maya.
- Hệ thống giao diện với con trỏ đồ họa thông minh cho phép người sử dụng dựng hình vẽ ba chiều trong không gian hai chiều của màn hình.

- Các mặt, diện (face) được định nghĩa đơn giản dựa trên một miền khép kín.
- Tạo khối đơn giản nhanh gọn bằng công cụ "kéo-đẩy" (push-pull tool)
- Công cụ chỉnh sửa khối (extrude - widen) và tạo khối theo đường sinh cho trước (follow me tool)
- Khả năng cho phép mô phỏng, hiệu chỉnh góc chiếu của mặt trời vào tất cả các thời điểm trong năm cũng như bao quát các góc nhìn cho hiệu quả gần như tức thời.
- Bản vẽ được kết xuất (render) ở tốc độ cao dựa trên tốt giả hệ mô hình đa giác thấp (low-poly), có phong cách trình bày độc đáo.
- Khả năng giao tiếp rộng rãi với các phần mềm mô hình khác.
- Có thể kết hợp với các trình kết xuất ngoài (Renderer) để cho ra những hình ảnh tốt hơn (etc. IRender, Podium, Indigo, Kerkythea...)

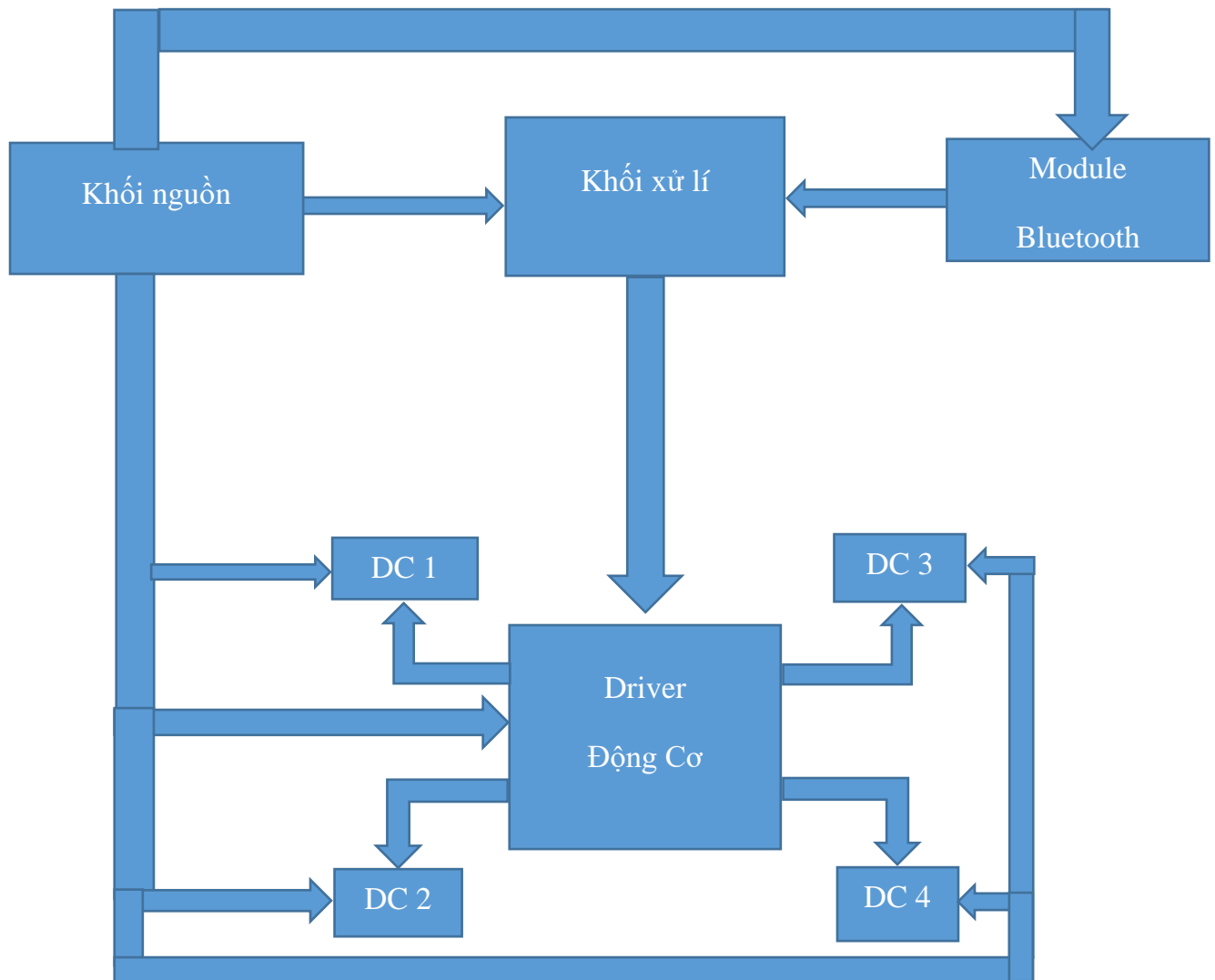
Phiên bản đầu tiên của SketchUp được phát hành đầu năm 2001 với mục đích tạo các đối tượng ba chiều đơn giản, tuy nhiên SketchUp đã sớm tìm được vị trí trong thị trường kiến trúc và thiết kết, sau khi có một vài sửa đổi nhỏ để phù hợp với đặc điểm của các ngành chuyên môn. Chìa khóa quyết định cho thành công của phần mềm nằm ở độ nhận biết các đường cong, tuy không sâu như các phần mềm mô hình hóa ba chiều trên thị trường nhưng đã đem lại tốc độ xử lý nhanh. Tuy nhiên chính điều này cũng hạn chế khả năng mô phỏng thực tế của SketchUp.



Hình 2.7: Giao diện của LayOut SketchUp dùng để vẽ 2D

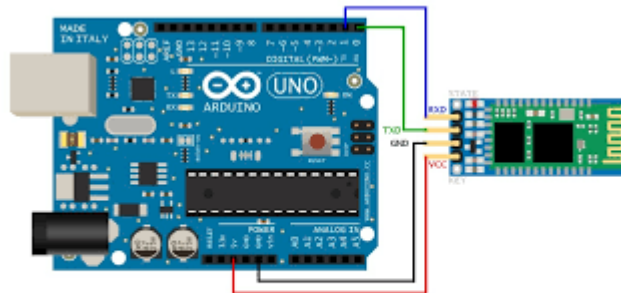
CHƯƠNG 3: CƠ SỞ THỰC HIỆN

3.1 Thiết kế phần cứng



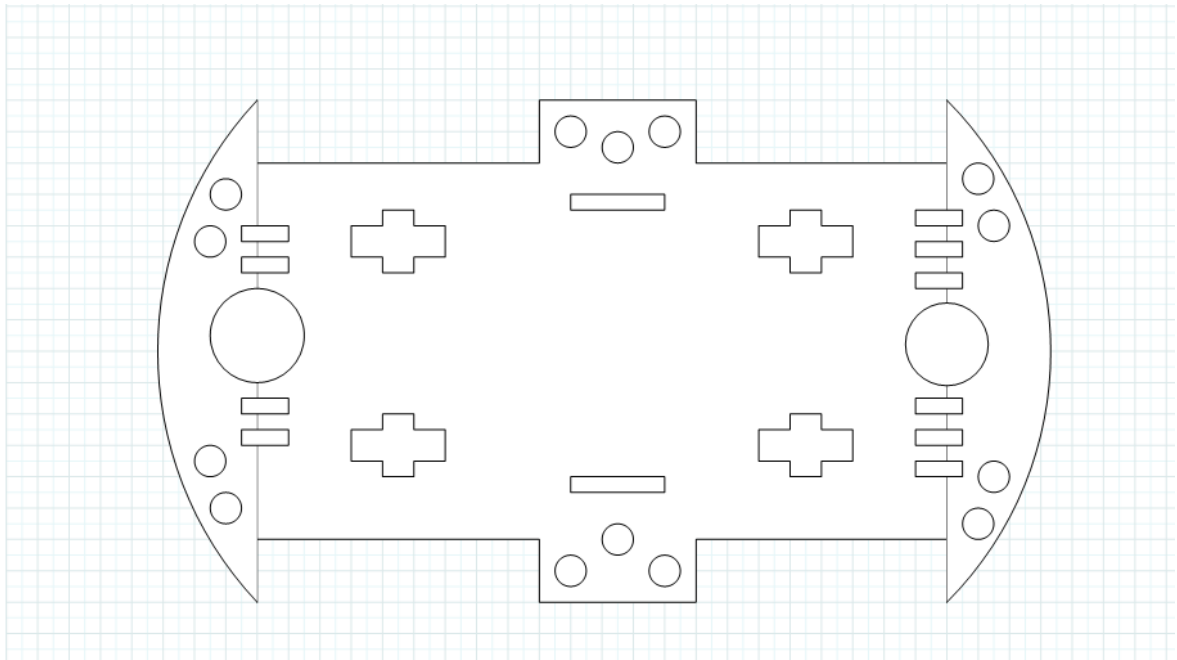
3.1.1 Khối xử lý

Khối xử lý sử dụng bo mạch Arduino UNO. Có nhiệm vụ nhận tín hiệu từ module Bluetooth, xử lý và xuất dữ liệu cho module Driver động cơ



Hình 3.1: Kết nối Arduino với HC-06

3.1.2. Thiết kế khung xe



Hình 3.2: Thiết kế khung xe bằng SketchUp 2D

3.2: Thiết kế phần mềm

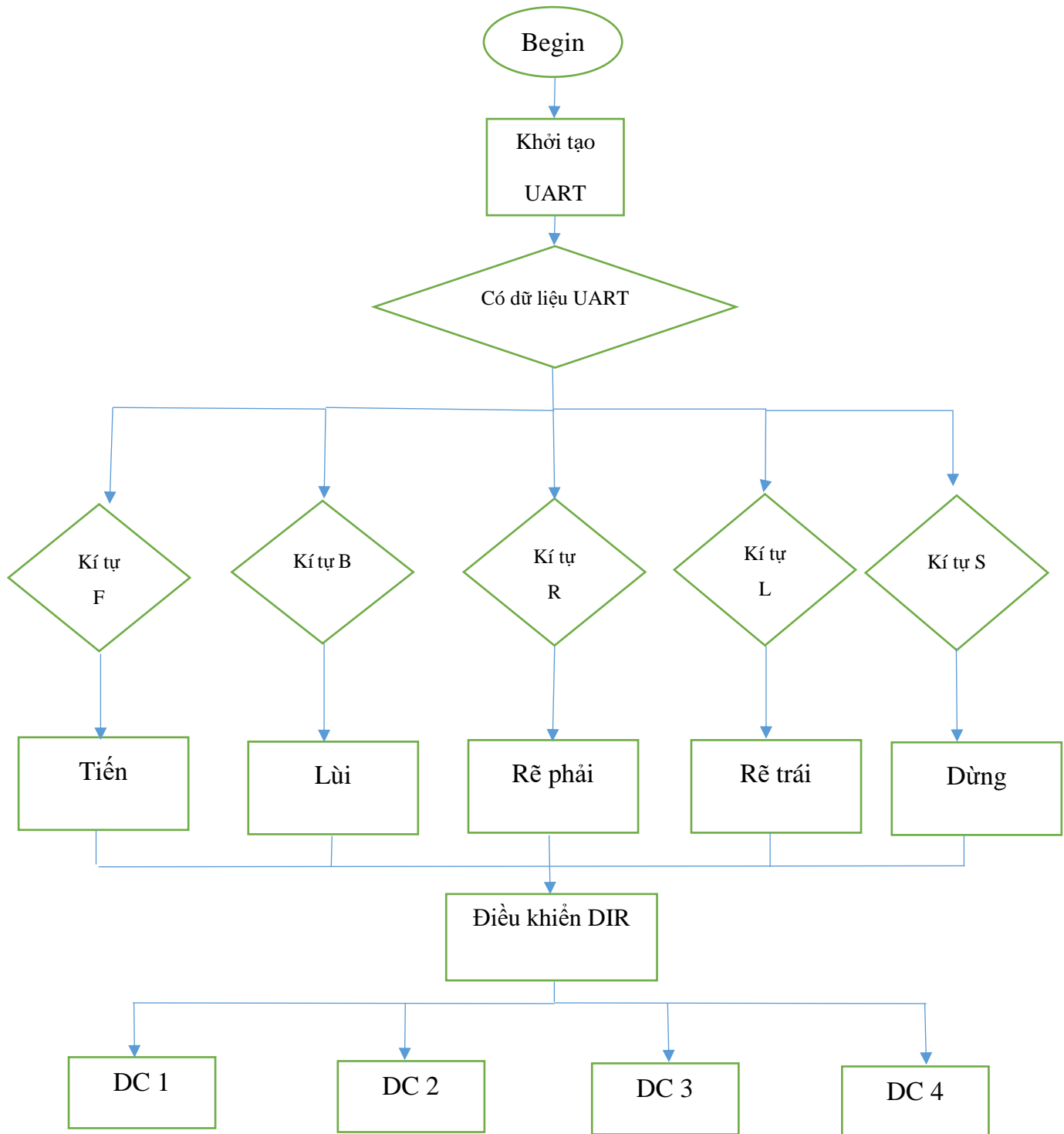
3.2.1: Phần mềm trên Androi



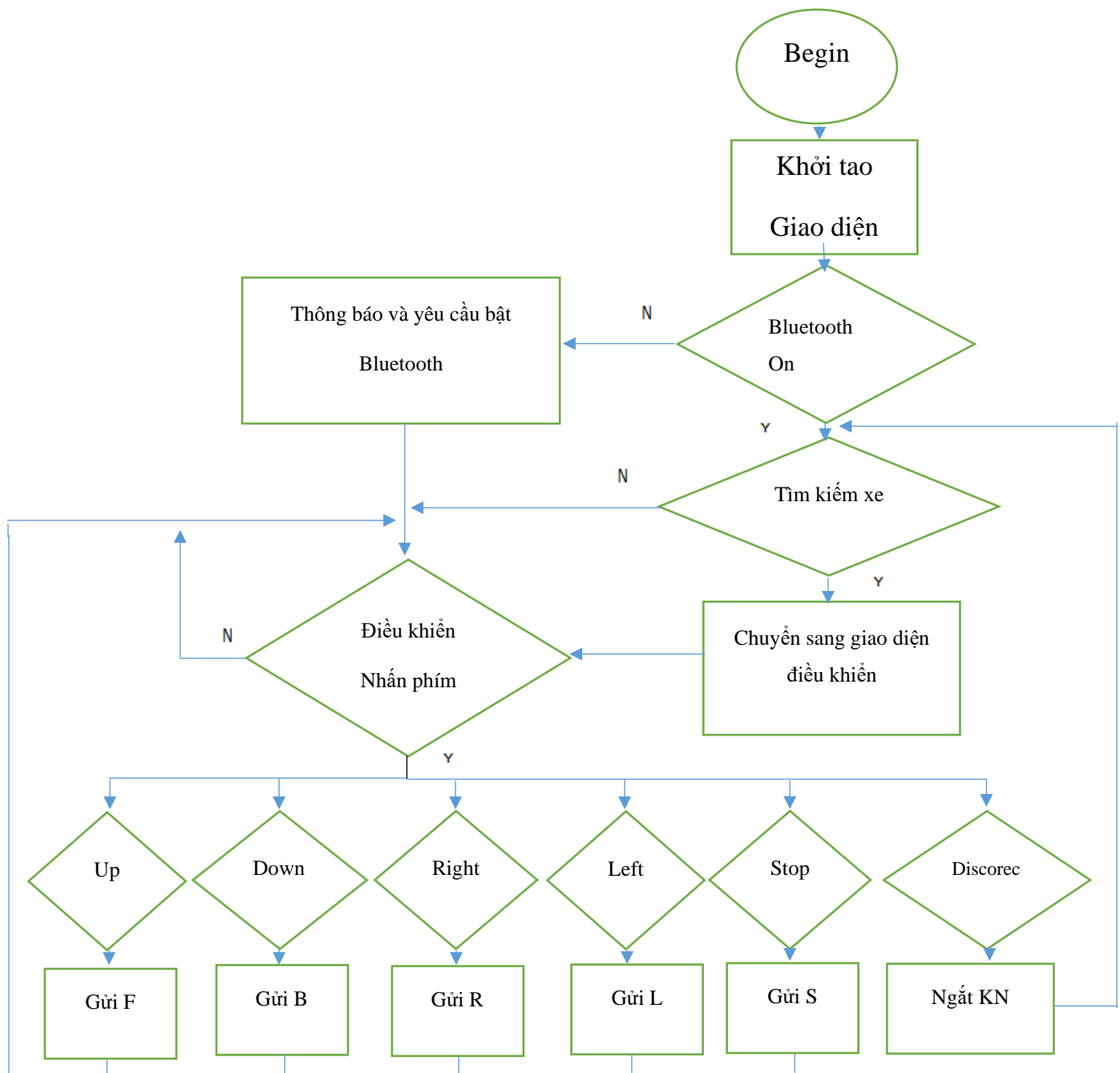
Hình 3.3: Giao diện phần mềm điều khiển

3.3 Lưu đồ thuật toán

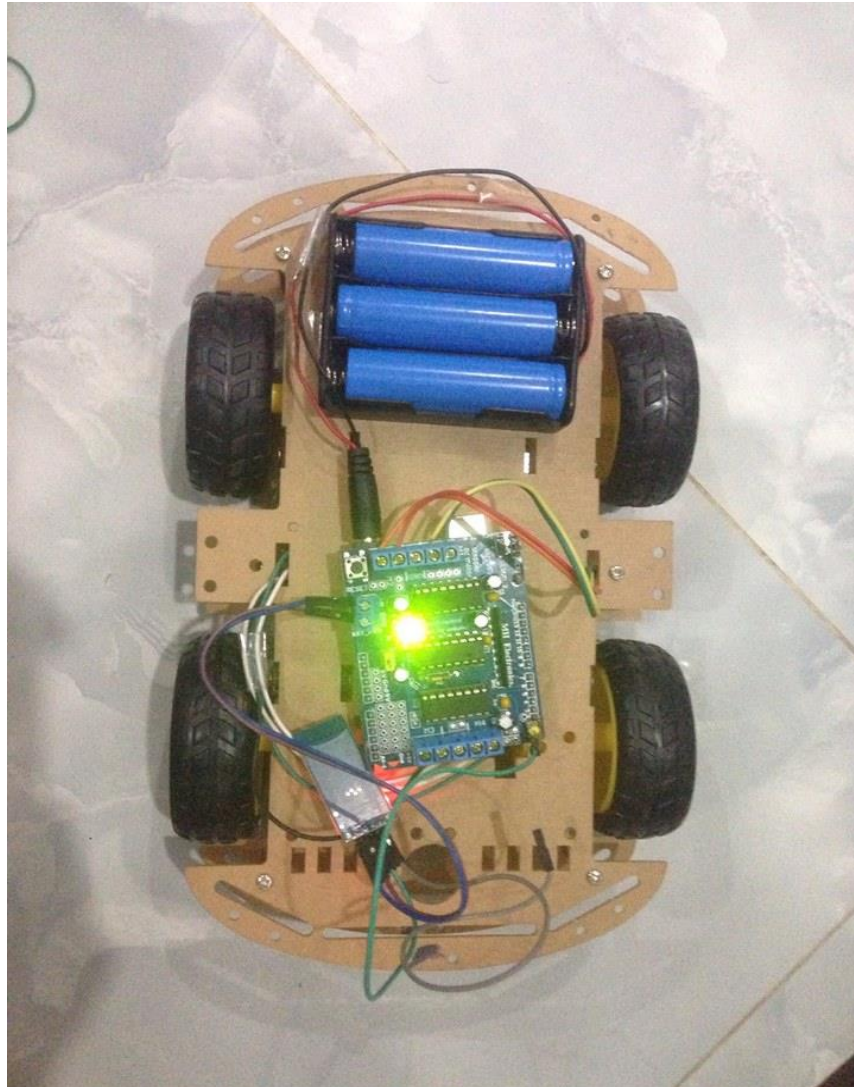
3.3.1: Thuật toán khối xử lí trung tâm Arduino



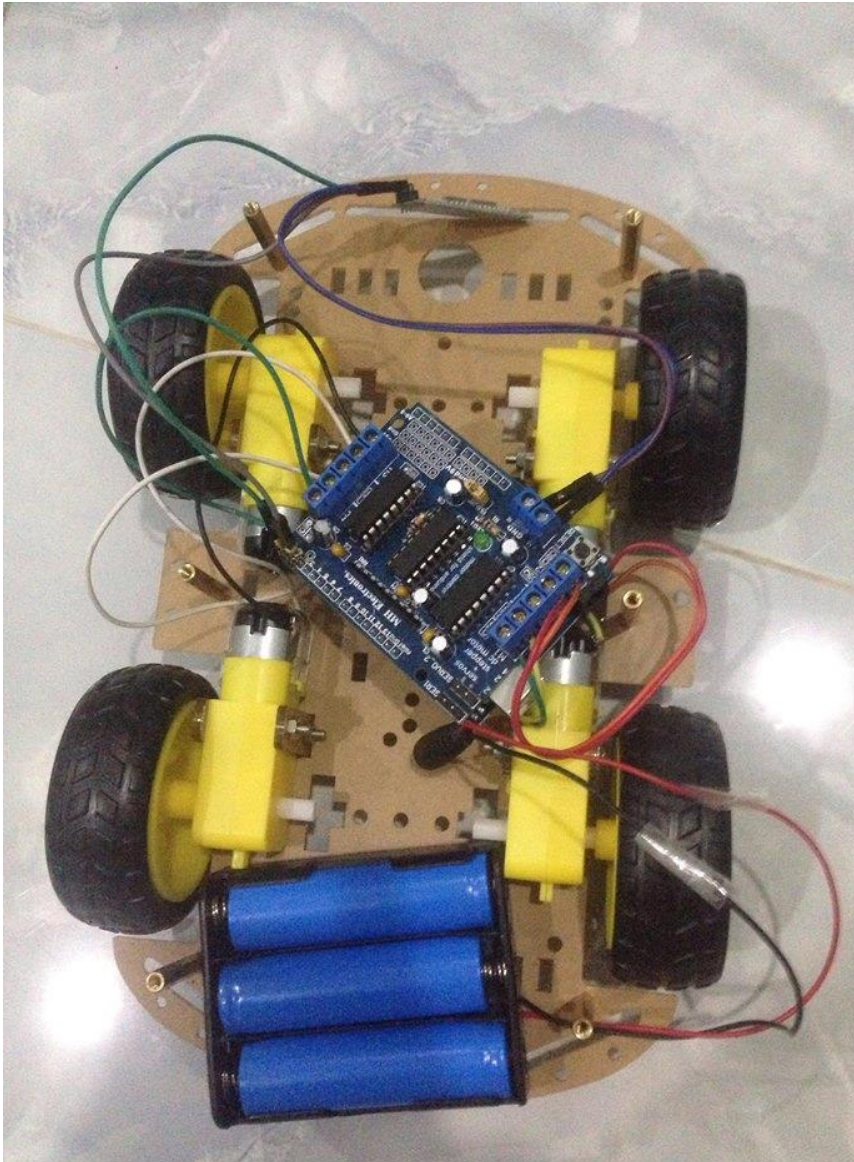
3.3.2: Thuật toán phần mềm điều khiển



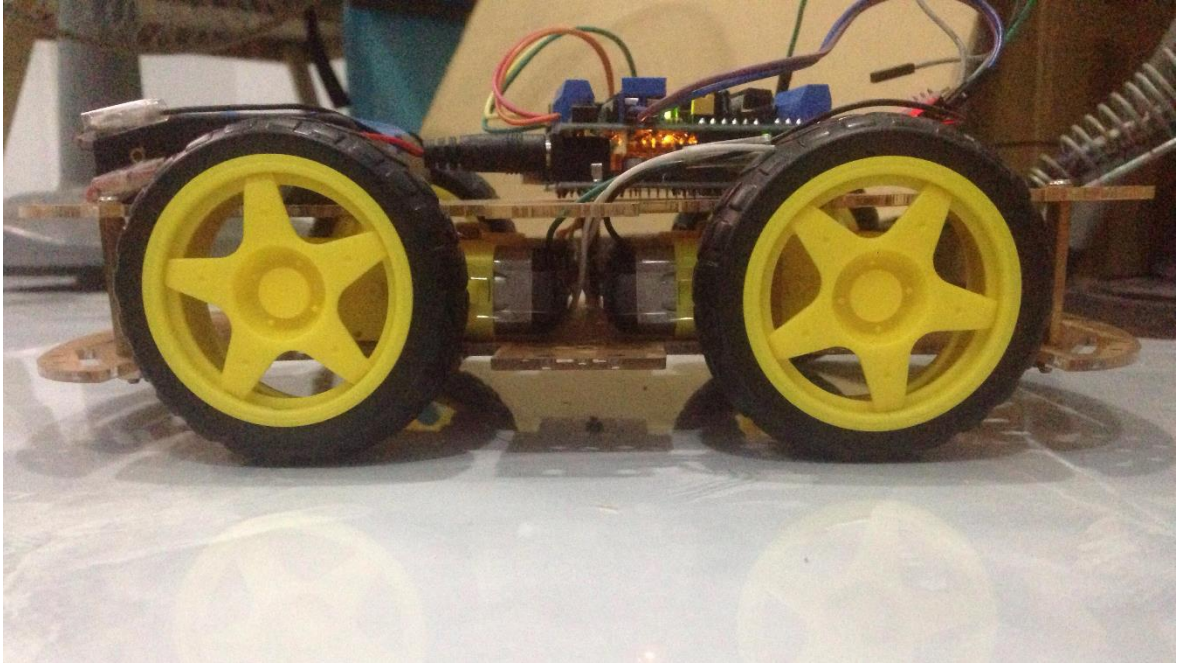
CHƯƠNG 4: KẾT QUẢ THỰC NGHIỆM



Hình ảnh mô hình nhìn từ trên xuống



Hình ảnh động cơ của mô hình



Mô hình khi nhìn từ 1 bên

CHƯƠNG 5: KẾT LUẬN VÀ ĐỊNH HƯỚNG ĐỀ TÀI

5.1 Kết quả đạt được:

Trên cơ sở lí thuyết đã học và qua tìm hiểu, em đã thiết kế mô hình xe điều khiển từ xa qua bluetooth với ứng dụng điều khiển có trên điện thoại Androi. Để có thể được ứng dụng nhiều hơn trong đời sống, mô hình cần có thêm nhiều bổ sung.

Qua quá trình thực hiện đồ án, em đã phần nào nắm rõ được giao tiếp Bluetooth, tìm hiểu về bo mạch Arduino, biết thiết kế kết cấu cơ khí cho khung xe, thực hiện kết nối và trao đổi dữ liệu giữa thiết bị cầm tay và Arduino UNO qua module Bluetooth.

Đây là một đề tài rất có ứng dụng trong thực tế nên em sẽ cố gắng để hoàn thiện đề tài và phù hợp với nhu cầu thực tế.

5.2 Hạn Chế :

Mô hình còn chỉ mới dừng lại ở mức độ đơn giản, điều khiển chạy tới lui, rẽ thông qua ứng dụng cài trên điện thoại hệ điều hành Androi, nhưng chưa có trên hệ điều hành iOS. Vì điều khiển qua Bluetooth nên khoảng cách điều khiển vẫn còn hạn chế, khoảng tầm ≤ 10 mét.

Trong quá trình làm đề tài do kiến thức còn hạn hẹp nên không thể tránh khỏi sai sót. Em mong rằng sẽ nhận được nhiều đóng góp ý kiến của thầy để sản phẩm ngày càng được hoàn thiện hơn.

5.3 Hướng Phát Triển:

Để mô hình xe điều khiển trở nên sinh động, thú vị, nhiều chức năng hơn thì ta có thể tích hợp thêm như truyền hình ảnh, đo nhiệt độ, độ ẩm, đo khoảng cách vật cản, đo độ nghiêng...Hoặc xe có thể phản hồi các sự cố về điện thoại di động.

PHỤ LỤC

Code chương trình:

```
#include<AFMotor.h>

AF_DCMotor motor1(1);
AF_DCMotor motor2(2);
AF_DCMotor motor3(3);
AF_DCMotor motor4(4);

char command;

void setup()
{
    Serial.begin(9600);
}

void loop() {
    if (Serial.available()) {
        command = Serial.read();
        Serial.println(command);
        Stop();
        switch (command) {
            case 'F':
                forward();
                break;
            case 'B':
```

```
        back();

        break;

    case 'L':

        left();

        break;

    case 'R':

        right();

        break;

    case 'S' :

        Stop();

        break;

    }

}

}

void right()

{

    motor1.setSpeed(255);

    motor1.run(FORWARD);

    motor2.setSpeed(255);

    motor2.run(FORWARD);

    motor3.setSpeed(255);

    motor3.run(BACKWARD);

    motor4.setSpeed(255);

    motor4.run(BACKWARD);
```

```
}  
  
void left()  
{  
    motor1.setSpeed(255);  
    motor1.run(BACKWARD);  
    motor2.setSpeed(255);  
    motor2.run(BACKWARD);  
    motor3.setSpeed(255);  
    motor3.run(FORWARD);  
    motor4.setSpeed(255);  
    motor4.run(FORWARD);  
}  
  
void forward()  
{  
    motor4.setSpeed(255);  
    motor4.run(FORWARD);  
    motor2.setSpeed(255);  
    motor2.run(FORWARD);  
    motor3.setSpeed(255);  
    motor3.run(FORWARD);  
    motor1.setSpeed(255);  
    motor1.run(FORWARD);  
}
```



```
void back()
{
    motor3.setSpeed(255);
    motor3.run(BACKWARD);
    motor1.setSpeed(255);
    motor1.run(BACKWARD);
    motor4.setSpeed(255);
    motor4.run(BACKWARD);
    motor2.setSpeed(255);
    motor2.run(BACKWARD);
}

void Stop()
{
    motor1.setSpeed(0);
    motor2.run(RELEASE);
    motor2.setSpeed(0);
    motor2.run(RELEASE);
    motor3.setSpeed(0);
    motor4.run(FORWARD);
    motor4.setSpeed(0);
    motor4.run(FORWARD);
}
```

TÀI LIỆU THAM KHẢO

- [1] <http://arduino.vn/bai-viet/639-du-xe-dieu-khien-tu-xa-qua-bluetooth>
- [2] <http://cee.duytan.edu.vn/vi-vn/tin-tuc/khoa-hoc-cong-nghe/gioi-thieu-ve-mach-arduino-va-cac-ung-dung/>