

ĐIỀU KHIỂN ĐỘNG CƠ ĐIỆN BẰNG HỘ VI ĐIỀU KHIỂN ATmega32

Chương 1: TỔNG QUAN

1.1 Tổng quan tình hình nghiên cứu và ứng dụng vi điều khiển:

Vi điều khiển là một tổ hợp các phần tử điện tử được tích hợp trên một mạch có nền chung gọi là chip, nó thường được sử dụng để điều khiển các thiết bị điện tử. Vi điều khiển, thực chất, là một hệ thống bao gồm một vi xử lý kết hợp với các khối ngoại vi như bộ nhớ, các module vào/ra, các module biến đổi số sang tương tự và tương tự sang số. Ở máy tính thì các module thường được xây dựng bởi các chip và mạch ngoài.

Vi điều khiển thường được dùng để xây dựng các hệ thống nhúng. Nó xuất hiện khá nhiều trong các thiết bị điện, điện tử, máy giặt, lò vi sóng, điện thoại, đầu đọc DVD, thiết bị đa phương tiện, dây chuyền tự động,...

Trên thế giới và tại Việt Nam, vi điều khiển được ứng dụng khá nhiều trong các thiết bị gia đình cũng như công nghiệp. Trên ô tô các vi xử lý ngày càng được sử dụng nhiều hơn để điều khiển các thiết bị nhờ khả năng ưu việt của nó.

Tại các trường đại học, cao đẳng trong khu vực, vi xử lý được đưa vào giảng dạy, nghiên cứu tạo ra các ứng dụng và các sân chơi mang tính sáng tạo cho người học. Đồng thời giúp người học tiếp cận với kiến thức mới.

1.2 Lý do chọn đề tài

Khoa ô tô trường cao đẳng Lý Tự Trọng chưa có chương trình học về vi điều khiển, sinh viên khoa ô tô của trường chưa có sân chơi sáng tạo về lĩnh vực vi điều khiển so với các trường lân cận và trong khu vực.

1.3 Mục tiêu nghiên cứu

Nghiên cứu khả năng khai thác các ứng dụng của vi điều khiển và các hệ thống trên ô tô.

1.4 Đối tượng và phương pháp nghiên cứu

1.4.1 Đối tượng nghiên cứu

Sử dụng chip điều khiển Atmega của Atmel Corporation vào các hệ thống trên ô tô

1.4.2 Phương pháp nghiên cứu

Nghiên cứu các công trình đã được công bố. Học tập kinh nghiệm thiết kế, sử dụng chip Atmega. Tra cứu Datasheet của họ Atmega. Nghiên cứu các phần mềm hỗ trợ viết code. Nghiên cứu phần mềm mô phỏng. Thiết kế và vận hành các ứng dụng đơn giản nhằm xem xét khả năng đưa vi điều khiển tiếp cận người học.

1.5 Ý nghĩa khoa học và thực tiễn của đề tài

1.5.1 Ý nghĩa khoa học

Tạo ra các ứng dụng sử dụng vi điều khiển vào các hệ thống trên ô tô

1.5.2 Ý nghĩa thực tiễn

Tạo sân chơi và môi trường học tập sáng tạo cho học sinh, sinh viên khoa công nghệ Động lực Trường Cao đẳng Lý Tự Trọng

Chương 2: CƠ SỞ LÝ THUYẾT

2.1 Tổng quan về vi điều khiển

Hầu hết các vi điều khiển ngày nay được xây dựng dựa trên kiến trúc Harvard, dùng để chỉ những kiến trúc máy tính mà trong đó phân biệt rõ ràng bộ nhớ dữ liệu và bộ nhớ chương trình, chúng có những đường truyền (bus) riêng để truy cập vào bộ nhớ dữ liệu và bộ nhớ chương trình, trong một máy tính kiến trúc Harvard, CPU có thể vừa đọc một lệnh, vừa truy cập dữ liệu từ bộ nhớ cùng lúc, kiến trúc này định nghĩa bốn thành phần cần thiết của một hệ thống nhúng. Những thành phần này là lõi CPU, bộ nhớ chương trình (thông thường là ROM hoặc bộ nhớ flash), bộ nhớ dữ liệu (RAM), một hoặc vài bộ định thời và các cổng vào/ra để giao tiếp với các thiết bị ngoại vi và các môi trường bên ngoài - tất cả các khối này được thiết kế trong một vi mạch tích hợp. Vi điều khiển khác với các bộ vi xử lý đa năng ở chỗ là nó có thể hoạt động chỉ với vài vi mạch hỗ trợ bên ngoài.

2.1.1 Các vi điều khiển thông dụng

Họ vi điều khiển AMCC do tập đoàn "Applied Micro Circuits Corporation" sản xuất.

- 403 PowerPC CPU
- PPC 403GCX
- 405 PowerPC CPU
- PPC 405EP
- PPC 405GP/CR

Họ vi điều khiển Atmel

- Dòng 8051 (8031, 8051, 8751, 8951, 8032, 8052, 8752, 8952)
- Dòng Atmel AT91 (Kiến trúc ARM THUMB)
- Dòng AT90, Tiny & Mega – AVR (Atmel Norway design)
- Dòng Atmel AT89 (Kiến trúc Intel 8051/MCS51)
- Dòng MARC4

Họ vi điều khiển Cypress Microsystems

- CY8C2xxxx (PSoC)

Họ vi điều khiển Freescale Semiconductor. Từ năm 2004, những vi điều khiển này được phát triển và tung ra thị trường bởi Motorola.

- Dòng 8-bit
 - 68HC05 (CPU05)
 - 68HC08 (CPU08)
 - 68HC11 (CPU11)
- Dòng 16-bit
 - 68HC12 (CPU12)
 - 68HC16 (CPU16)
 - Freescale DSP56800 (DSPcontroller)
- Dòng 32-bit
 - Freescale 683XX (CPU32)
 - MPC500
 - MPC 860 (PowerQUICC)
 - MPC 8240/8250 (PowerQUICC II)
 - MPC 8540/8555/8560 (PowerQUICC III)

Họ vi điều khiển Fujitsu

- F²MC Family (8/16 bit)
- FR Family (32 bit)
- FR-V Family (32 bit RISC)

Họ vi điều khiển Intel

- Dòng 8-bit
 - 8XC42
 - MCS48
 - MCS51
 - 8061
 - 8xC251
- Dòng 16-bit
 - 80186/88
 - MCS96
 - MXS296
- Dòng 32-bit
 - 386EX
 - i960

Họ vi điều khiển Microchip

- PIC 8-bit (xử lý dữ liệu 8-bit, 8-bit data bus)
 - Từ lệnh dài 12-bit (Base-line): PIC10F, PIC12F và một vài PIC16F
 - Từ lệnh dài 14-bit (Mid-Range và Enhance Mid-Range): PIC16Fxxx, PIC16F1xxx
 - Từ lệnh dài 16-bit (High Performance): PIC18F
- PIC 16-bit (xử lý dữ liệu 16-bit)
 - PIC điều khiển động cơ: dsPIC30F
 - PIC có DSC: dsPIC33F
 - Phổ thông: PIC24F, PIC24E, PIC24H
- PIC 32-bit (xử lý dữ liệu 32-bit): PIC32MX

Họ vi điều khiển National Semiconductor

- COP8
- CR16

Họ vi điều khiển STMicroelectronics

- ST 62
- ST7
- STM8
- STM32 (Cortex-Mx)

Họ vi điều khiển Philips Semiconductors

- LPC2000
- LPC900
- LPC700

2.1.2 Vi điều khiển Atmega



Atmel AVR ATmega8 [PDIP](#).

Vi điều khiển AVR do hãng Atmel sản xuất được giới thiệu lần đầu năm 1996. AVR có rất nhiều dòng khác nhau bao gồm dòng Tiny AVR (như AT tiny 13, AT tiny 22...) có kích thước bộ nhớ nhỏ, ít bộ phận ngoại vi, rồi đến dòng AVR (chẳng hạn AT90S8535, AT90S8515,...) có kích thước bộ nhớ vào loại trung bình và mạnh hơn là dòng Mega (như ATmega32, ATmega128,...) với bộ nhớ có kích thước vài Kbyte đến vài trăm Kb cùng với các bộ ngoại vi đa dạng được tích hợp trên chip, cũng có dòng tích hợp cả bộ LCD trên chip (dòng LCD AVR). Tốc độ của dòng Mega cũng cao hơn so với các dòng khác. Sự khác nhau cơ bản giữa các dòng chính là cấu trúc ngoại vi, còn nhân thì vẫn như nhau.

Vi điều khiển Atmega AVR có công suất cao, tiêu thụ năng lượng thấp, cấu trúc RISC tiến với 130 lệnh với chu kỳ thực hiện đơn xung lớn nhất, 32 thanh ghi đa mục đích 8 bit, 16 MIPS tại tần số đặt 16 MHz, bộ nhân 2 chu kỳ On-chip, Power-on Reset và Brown-out Detection có thể lập trình, bộ dao động RC bên trong có thể lập trình các mức, 5 Mode ngủ (Idle, ADC Noise Reduction, Power-save, Power-down và Standby), có khả năng Reset khi bật nguồn, khả năng dò lỗi Brown out lập trình được, có nguồn ngắt trong và ngắt ngoài.

Cốt lõi của AVR là sự kết hợp các câu lệnh phong phú với 32 thanh ghi đa mục đích. Tất cả 32 thanh ghi đều trực tiếp kết nối tới bộ xử lý logic số học - Arithmetic Logic Unit (ALU), cho phép truy nhập 2 thanh ghi độc lập trong một câu lệnh đơn được thực

hiện trong một chu kỳ xung. Kết quả của cấu trúc trở nên gọn nhẹ, hiệu quả hơn, trong khi vẫn đạt được thời gian xử lý nhanh hơn gấp 10 lần các vi điều khiển CISC thông thường khác.

Một ưu điểm khác của chip Atmega là giá thành rẻ và dễ tiếp cận với người mới bắt đầu. Các mạch ứng dụng và mạch hỗ trợ phong phú, dễ tìm kiếm, cấu trúc nguồn mở cho phép truy xuất dữ liệu rộng.

2.2 Mạch tích hợp arduino với vi điều khiển ATmega32



Mạch tích hợp arduino uno

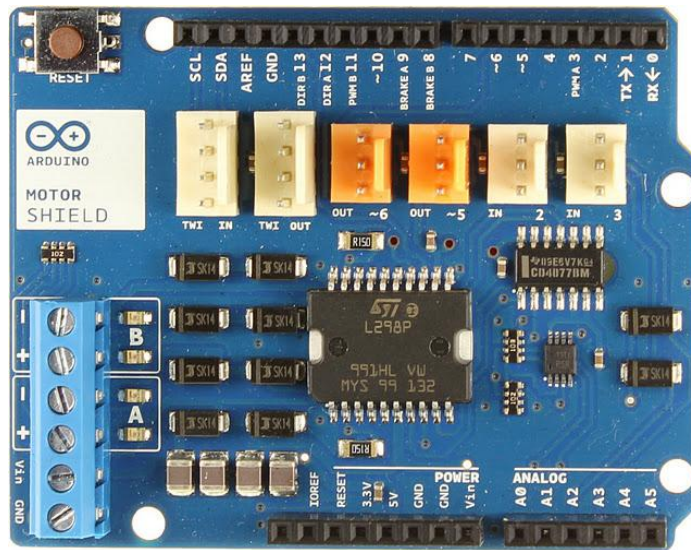
Arduino đã và đang được sử dụng rất rộng rãi trên thế giới, và ngày càng chứng tỏ được sức mạnh của chúng thông qua vô số ứng dụng độc đáo của người dùng trong cộng đồng nguồn mở (open-source). Tại Việt Nam Arduino được biết đến nhiều trong thời gian gần đây thông qua các ứng dụng mạnh mẽ của nó.

2.2.1 Các loại bo mạch Arduino

Về mặt chức năng, các bo mạch Arduino được chia thành hai loại: loại bo mạch chính có chip Atmega ví dụ như UNO hay Pro và loại mở rộng thêm chức năng cho bo mạch chính (thường được gọi là shield).

Các bo mạch chính về cơ bản là giống nhau về chức năng, tuy nhiên về mặt cấu hình như số lượng I/O, dung lượng bộ nhớ, hay kích thước có sự khác nhau. Một số bo có trang bị thêm các tính năng kết nối như Ethernet và Bluetooth.

Các bo mở rộng chủ yếu mở rộng thêm một số tính năng cho bo mạch chính ví dụ như tính năng kết nối Ethernet, Wireless, điều khiển động cơ v.v...



Bo mạch mở rộng Motor Shield dùng để điều khiển động cơ DC hoặc Stepper



Các mạch arduino thông dụng

2.2.2 Các ứng dụng nổi bật của bo mạch Arduino

Arduino được chọn làm bộ não xử lý của rất nhiều thiết bị từ đơn giản đến phức tạp. Trong số đó có một vài ứng dụng thực sự chứng tỏ khả năng vượt trội của Arduino do chúng có khả năng thực hiện nhiều nhiệm vụ rất phức tạp. Sau đây là danh sách một số ứng dụng nổi bật của Arduino.

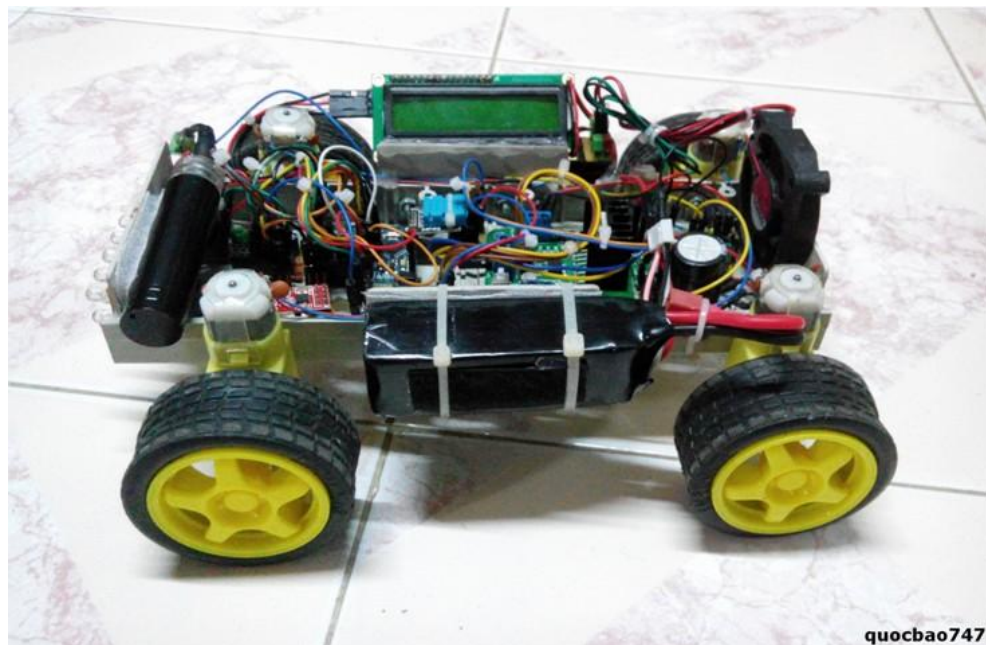
Máy in 3D

Một cuộc cách mạng khác cũng đang âm thầm định hình nhờ vào Arduino, đó là sự phát

triển máy in 3D nguồn mở Reprap. Máy in 3D là công cụ giúp tạo ra các vật thể thực trực tiếp từ các file CAD 3D. Công nghệ này hứa hẹn nhiều ứng dụng rất thú vị trong đó có cách mạng hóa việc sản xuất cá nhân.



Xe điều khiển



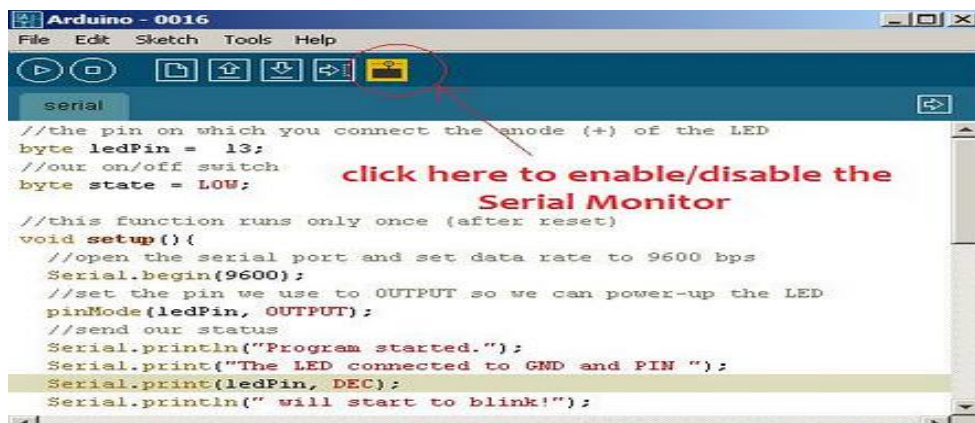
Robot

Do kích thước nhỏ gọn và khả năng xử lý mạnh mẽ, Arduino được chọn làm bộ xử lý trung tâm của rất nhiều loại robot, đặc biệt là robot di động.



2.2.3 Môi trường lập trình bo mạch Arduino

Thiết kế bo mạch nhỏ gọn, trang bị nhiều tính năng thông dụng mang lại nhiều lợi thế cho Arduino, tuy nhiên sức mạnh thực sự của Arduino nằm ở phần mềm. Môi trường lập trình đơn giản dễ sử dụng, ngôn ngữ lập trình dễ hiểu và dựa trên nền tảng C/C++ rất quen thuộc với người làm kỹ thuật. Và quan trọng là số lượng thư viện code được viết sẵn và chia sẻ bởi cộng đồng nguồn mở là cực kỳ lớn.

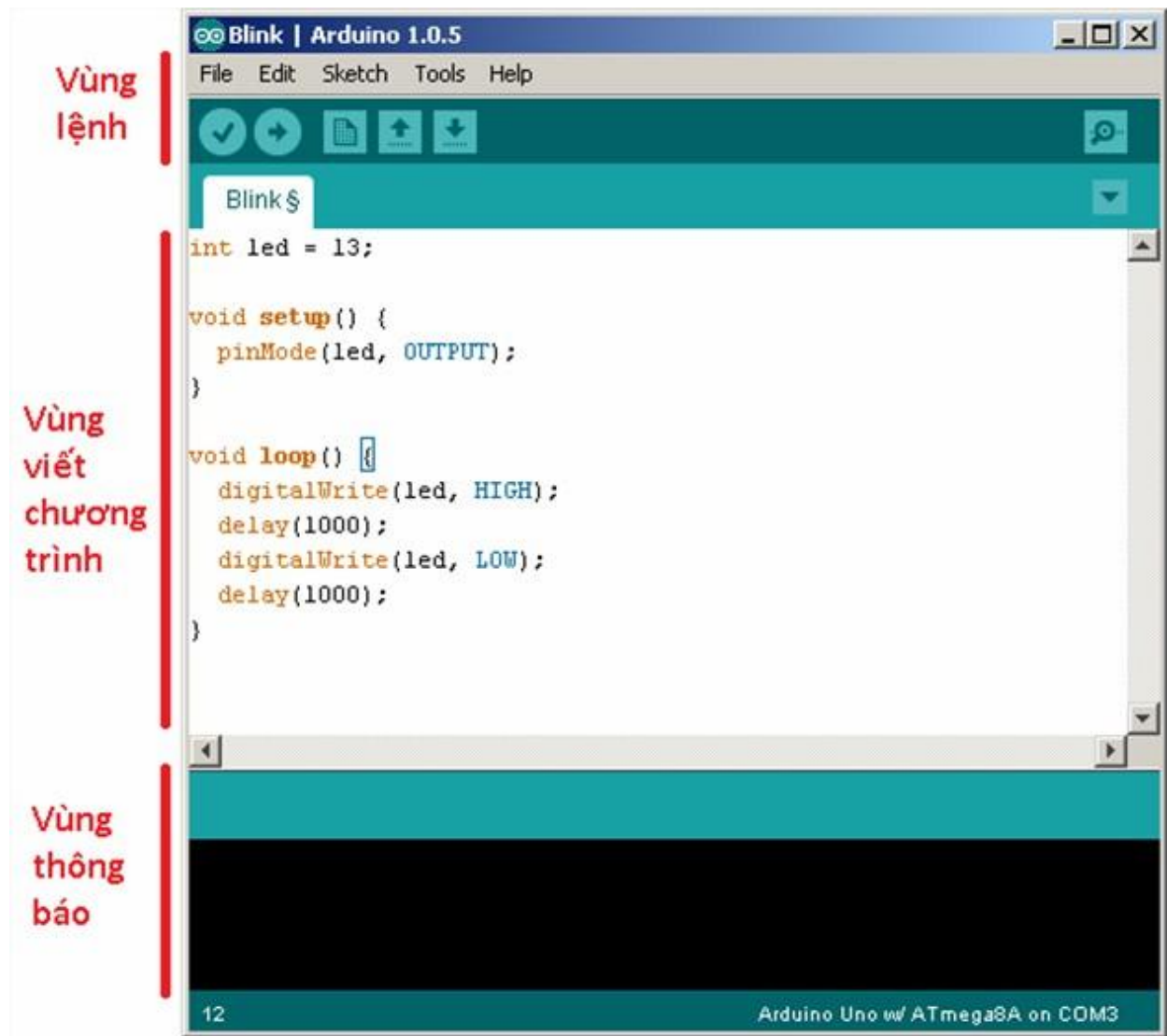


2.3 Phần mềm lập trình và hỗ trợ lập trình

2.3.1 Phần mềm lập trình cho arduino

Arduino IDE

Intergrated Development Environment (IDE) là công cụ được sử dụng để lập trình cho các board Arduino.



Vùng lệnh

Bao gồm các nút lệnh menu (File, Edit, Sketch, Tools, Help) và các icon cho phép sử dụng nhanh các chức năng thường dùng của IDE.

Vùng viết chương trình

Là nơi mà bạn sẽ viết các đoạn mã của mình tại đây.

Vùng thông báo

Những thông báo từ IDE sẽ được hiển thị tại đây.

Ưu điểm của IDE là giao diện đơn giản, dễ sử dụng và thư viện hỗ trợ khá đầy đủ. Nó thích hợp cho người mới bắt đầu.

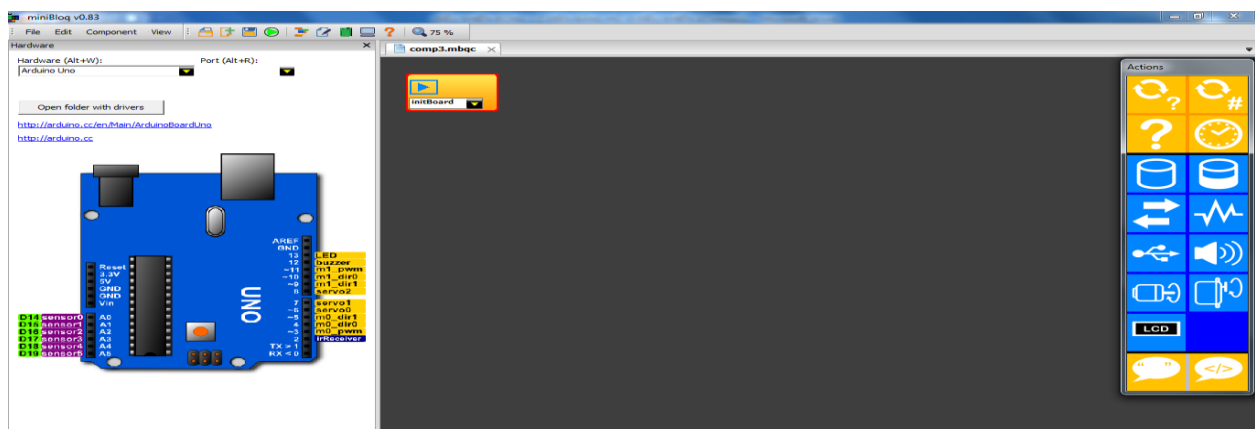
C/C++

Công cụ lập trình C/C++ là công cụ mạnh mẽ khi viết chương trình cho vi điều khiển. Tuy nhiên nó chỉ thích hợp với những người hiểu biết sâu hơn về vi điều khiển. Nói cách khác, chỉ khi thật sự có kiến thức phong phú về vi điều khiển thì mới sử dụng C/C++ để tạo ra các ứng dụng phong phú hơn cho vi điều khiển.

2.3.2 Phần mềm hỗ trợ lập trình cho arduino

Do tính phổ dụng của arduino nên có nhiều phần mềm hỗ trợ cho người dùng khi sử dụng. Các phần mềm này có ưu điểm là kéo thả và giao diện trực quan, nó giúp người dùng không mất nhiều thời gian viết code và gặp các lỗi logic khi viết code. Điều này hỗ trợ tích cực cho người mới làm quen với vi điều khiển, tuy nhiên, thư viện của nó có xu hướng nhàm chán khi các ứng dụng có tính rập khuôn và hạn chế sự mở rộng khả năng của người học cũng như mở rộng khả năng sử dụng vi điều khiển và những mục đích có tính phức tạp hay đặc thù.

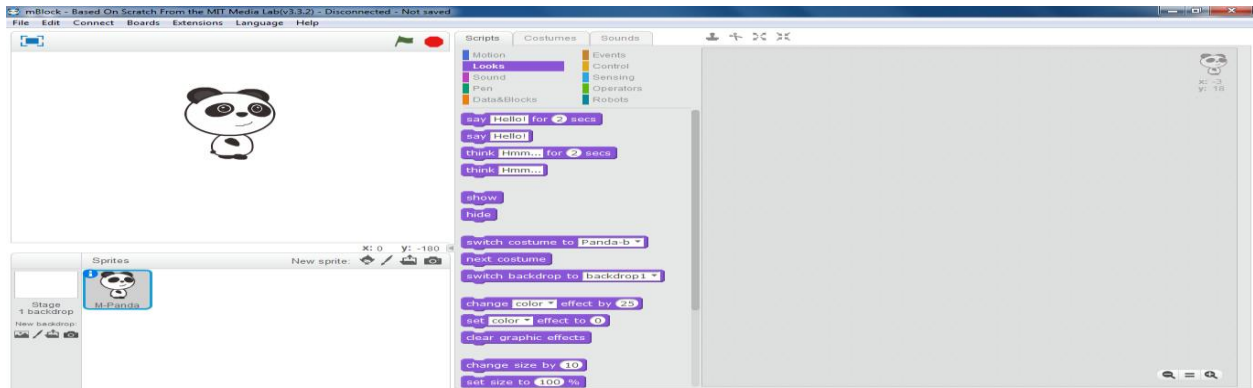
Phần mềm miniBloq



MiniBloq là phần mềm hỗ trợ lập trình cho người mới bắt đầu khá phổ biến. Giao diện đơn giản, dễ sử dụng và hỗ trợ tốt cho arduino từ thư viện mở. Các công cụ của

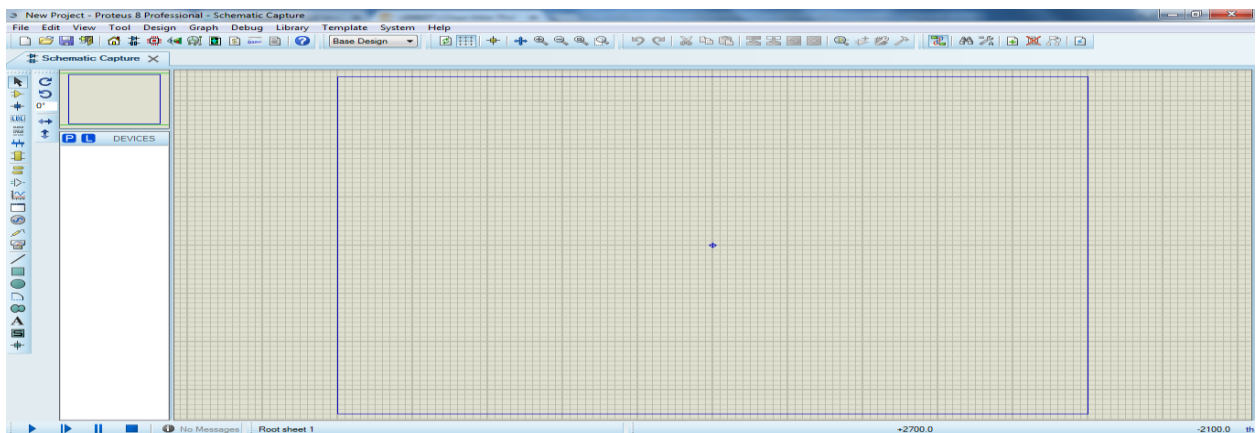
miniBloq có tính trực quan và chỉ cần kéo-thả nên người mới làm quen có thể dễ dàng thiết lập hoạt động của bo mạch arduino mà không cần hiểu biết về ngôn ngữ lập trình.

Phần mềm mBlock



Tương tự như miniBloq, mBlock cũng là phần mềm hỗ trợ lập trình cho arduino. Giao diện thân thiện, dễ sử dụng, chức năng kéo thả giúp nhanh chóng đạt được yêu cầu mà không cần quan tâm đến khả năng lỗi logic trong quá trình viết code

Phần mềm proteus



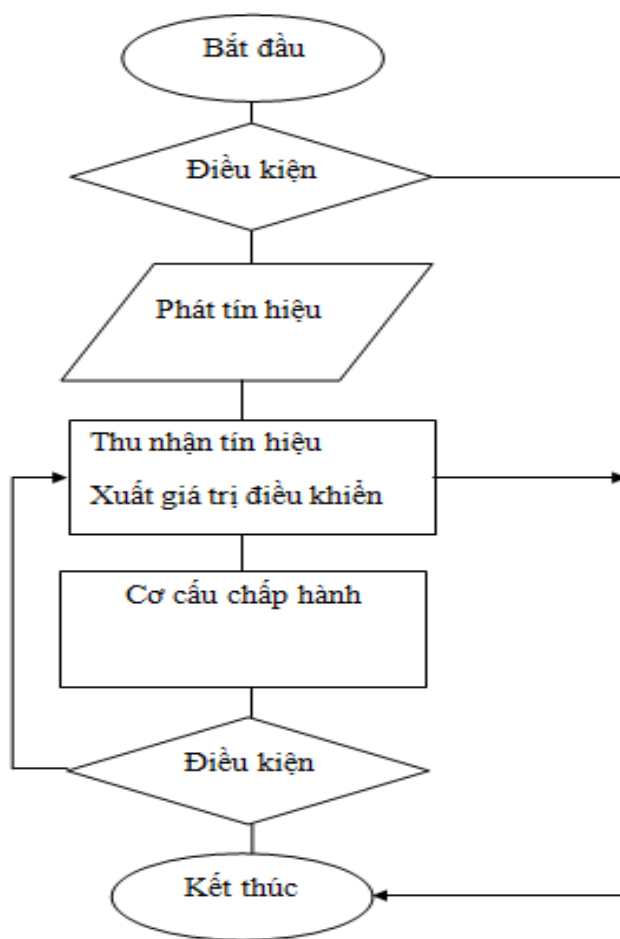
Proteus là phần mềm mô phỏng, phần mềm hỗ trợ rất nhiều linh kiện cũng như bo mạch với thư viện mở. Tuy khó sử dụng nhưng là công cụ đắc lực cho người học khi có khả năng mô phỏng với độ chính xác lên đến hơn 90%. Môi trường trong proteus cho phép thiết lập các phép thử với nhiều linh kiện và mạch ngoại vi hỗ trợ. Kết quả của mô phỏng là cơ sở cho thiết lập các hệ thống và giảm chi phí cho quá trình thiết kế.

CHƯƠNG 3: XÂY DỰNG MÔ HÌNH ĐIỀU KHIỂN ĐỘNG CƠ ĐIỆN BẰNG HỘ VI ĐIỀU KHIỂN ATmega32

3.1 Phương pháp xây dựng mô hình

Để xây dựng mô hình điều khiển động cơ DC có thể ứng dụng vào các hệ thống trên ô tô. Cần tiến hành xây dựng giải thuật với các đặc điểm tương quan với các hệ thống trên ô tô.

Giải thuật được xây dựng theo sơ đồ như sau



Như vậy hệ thống muốn hoạt động cần có hai điều kiện. Cần phải có tín hiệu ban đầu ứng với nguồn cấp cho hệ thống và phải có điều kiện sau cơ cấu chấp hành ứng với cơ cấu chấp hành thực sự có hoạt động.

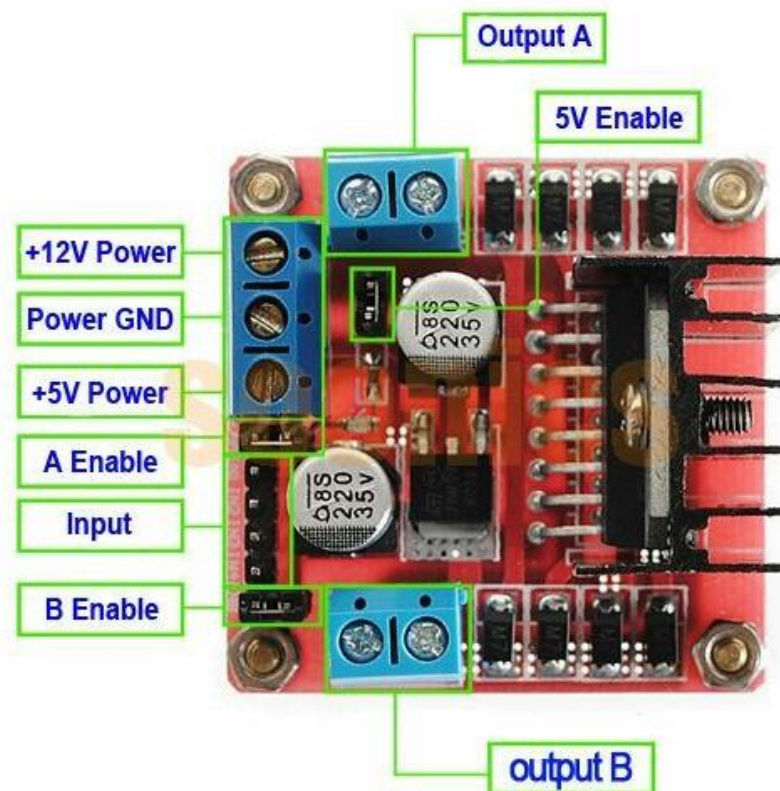
Điều này tương ứng với tín hiệu nguồn từ công tắc máy và tín hiệu phản hồi từ hoạt động hệ thống. Tín hiệu nguồn dùng để tránh các hoạt động ngoài ý muốn khi công tắc máy chưa được bật. Còn tín hiệu phản hồi từ hệ thống giúp tự chuẩn đoán khi hệ thống không hoạt động cũng như hệ thống không thể hoạt động.

3.1.1 Xây dựng mô hình

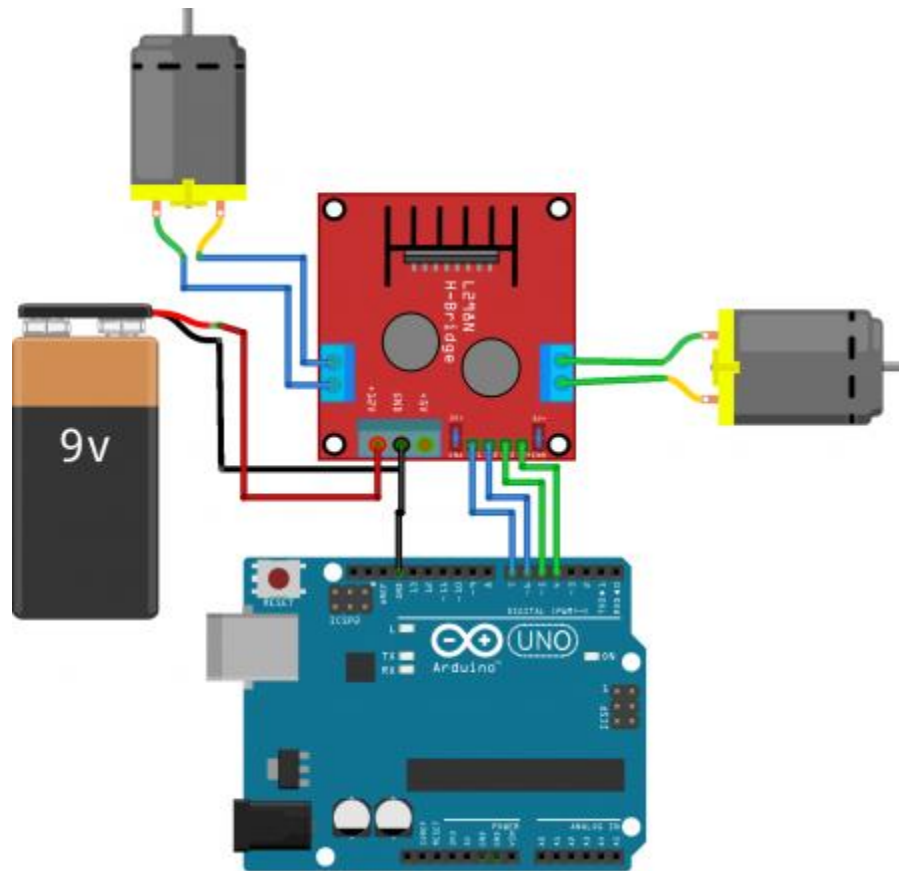
Đơn giản, mô hình được xây dựng với chức năng điều khiển hệ thống là chủ yếu, tạm thời loại bỏ các điều kiện. Nếu mô hình hoạt động tốt, các chức năng khác sẽ được phát triển thêm.

Mô tả

Điều khiển động cơ DC dùng arduino UNO và module L298N.



Sơ đồ nối dây



Để điều khiển 2 động cơ DC, cần chú ý đầu nối cực +,- của động cơ tương ứng với chân +,- của OUTPUT X.

Cấp nguồn cho Module L298 như hình trên.

Nếu dùng động cơ 5V và động cơ dưới 1A có thể dùng chân 5V của Arduino. Nếu không, nguồn cấp cho động cơ ở L298 phải là nguồn riêng để không làm hỏng Arduino.

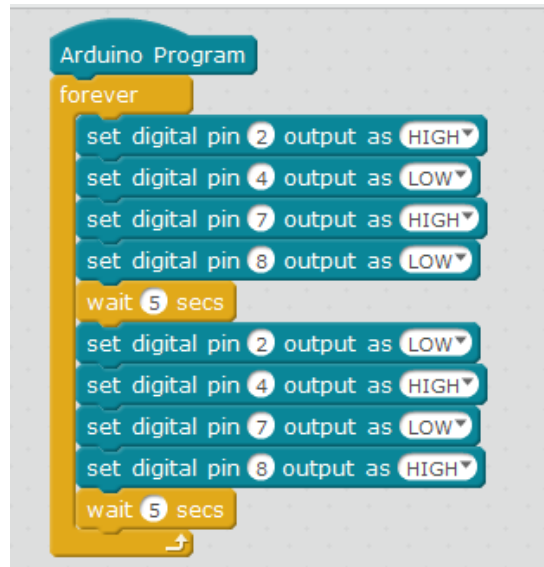
Các chân số D2, D4, D7 và D8 của Arduino sẽ nối tương ứng với IN1, IN2, IN3 và IN4 của L298.

Chiều quay của động cơ được điều khiển bằng cách xuất các đầu ra HIGH hoặc LOW tại các chân INx.

Ví dụ với Động Cơ A: Logic HIGH ở IN1 và IN2 Logic LOW sẽ làm động cơ quay một hướng nếu đặt Logic ngược lại sẽ làm động cơ quay theo hướng ngược lại.

3.1.2 Điều khiển động cơ DC

Sử dụng phần mềm hỗ trợ mBlock.



3.2 Kết quả

Mạch hoạt động ổn định. Không có hiện tượng trễ. Quá trình hoạt động kéo dài không nhận thấy lỗi phát sinh do hệ thống. Thời gian đảo chiều ổn định, tốc độ động cơ không thay đổi sau nhiều chu kỳ.

CHƯƠNG 4: KẾT LUẬN VÀ KIẾN NGHỊ

4.1 Kết luận

Vi điều khiển nói chung và Atmega với bo mạch tích hợp arduino UNO có thể dùng để điều khiển động cơ điện DC. Kết quả cho thấy có thể ứng dụng vào nhiều hệ thống trên ô tô như nâng hạ kính, điều chỉnh gương chiếu hậu, hệ thống gạt nước mưa, hệ thống điều chỉnh ghế...

Khả năng tiếp cận vi điều khiển không quá khó đối với sinh viên khoa ô tô trường cao đẳng lý tự trọng trong khi môi trường nghiên cứu rất phong phú, khả năng sáng tạo là rất đáng quan tâm.

Hiểu biết về vi điều khiển giúp người học dễ hiểu hơn về các hệ thống trên ô tô như hệ thống phun xăng điện tử EFI, hệ thống chuẩn đoán, hệ thống ABS...

Nắm được một kiến thức mới giúp tâm thế người học tự tin hơn, bản lĩnh hơn và đam mê nghề nghiệp hơn.

4.2 Kiến nghị

Nên giới thiệu với người học như là một sân chơi mở. Tạo điều kiện cho người học tiếp cận kiến thức mới. Tăng khả năng kích thích sáng tạo cho người học.

Mở rộng các ứng dụng và tăng mức độ khó của hệ thống là công việc cần làm để hệ thống gần sát với hệ thống thực tế trên ô tô.

Phát triển hệ thống độc lập, tương tác, tương thích và có thể thay thế hệ thống trên ô tô là mục tiêu của những dự án tiếp theo.

Tài liệu tham khảo

https://vi.wikipedia.org/wiki/Vi_điều_khiển

<http://arduino.vn/>

<http://st.com/>

https://www.arduino.cc/en/uploads/Main/Arduino_Uno_Rev3-schematic

<http://arduino.cc/en/Tutorial/HomePage>

<https://www.arduino.cc/en/Main/Software>

