

TỰ HỌC LẬP TRÌNH ARDUINO
Trong ngành Công nghệ điện điện tử
Giảng viên: Kỹ sư Châu Lê Sơn

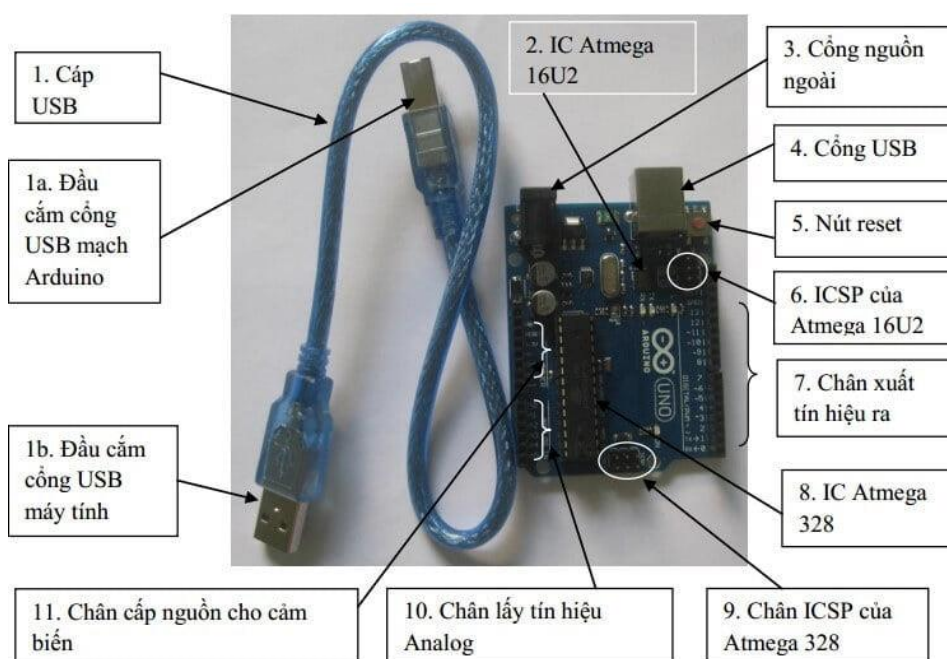
Arduino là gì?

Arduino là một bo mạch vi điều khiển do một nhóm giáo sư và sinh viên nước Ý thiết kế và đưa ra đầu tiên vào năm 2005. Mạch Arduino được sử dụng để cảm nhận và điều khiển nhiều đối tượng khác nhau. Nó có thể thực hiện nhiều nhiệm vụ lấy tín hiệu từ cảm biến đến điều khiển đèn, động cơ, và nhiều đối tượng khác. Ngoài ra mạch còn có khả năng liên kết với nhiều module khác nhau như module đọc thẻ từ, ethernet shield, sim900A, để tăng khả năng ứng dụng của mạch.

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,.... Hiện phần cứng của Arduino có tất cả 6 phiên bản, Tuy nhiên phiên bản thường được sử dụng nhiều nhất là Arduino Uno và Arduino Mega.

Phần mềm để lập trình cho mạch Arduino là phần mềm IDE.

Cấu tạo Arduino



Cấu tạo của Arduino Uno R3

Thông số cơ bản của Arduino Uno R3

Vi điều khiển	Atmega 328 (họ 8 bit)
Điện áp hoạt động	5V – DC (cấp qua cổng USB)
Tần số hoạt động	16 MHz
Dòng tiêu thụ	30mA
Điện áp vào khuyến dùng	7 – 12V – DC
Điện áp vào giới hạn	6 – 20V – DC
Số chân Digital I/O	14 (6 chân PWM)
Số chân Analog	6 (độ phân giải 10 bit)
Dòng tối đa trên mỗi chân I/O	30 mA
Dòng ra tối đa (5V)	500 mA
Dòng ra tối đa (3.3V)	50 mA
Bộ nhớ flash	32 KB (Atmega328) với 0.5KB dùng bởi bootloader
SRAM	2KB (Atmega328)
EEPROM	1KB (Atmega328)

Các loại Board Arduino phổ biến

Không giống như hầu hết các board mạch lập trình trước đó, Arduino không yêu cầu một phần cứng riêng để lập trình mã mới lên board mà bạn chỉ cần sử dụng cáp USB. Đồng thời, phần mềm Arduino IDE sử dụng phiên bản cơ bản của C ++, giúp việc học chương trình trở nên đơn giản hơn. Chúng ta có thể tổng hợp một số loại Arduino phổ biến như sau:

1. Arduino Uno

Đây chính là loại board đơn giản nhất nên rất phù hợp với những người mới bắt đầu tìm hiểu về lĩnh vực này. Dữ liệu số bao gồm 14 chân, đầu vào gồm 6 chân

5V, khả năng phân giải là 1024 mức, tốc độ 16MHz, điện áp từ 7V đến 12V. Kích thước của Board này là 5,5x7cm.

2. *Arduino Micro*

Bao gồm có đến 20 chân, trong đó có 7 chân có thể phát PWM. Loại này có thiết kế khá nhỏ gọn, kích thước chỉ 5x2cm.

3. *Arduino Nano*

Có thể nói đây chính là loại board có kích thước nhỏ nhất chỉ 2x4cm, việc lắp đặt được thực hiện dễ dàng.

4. *Arduino Pro*

Đây là một thiết kế mới mẻ khi chân số không có sẵn, tùy vào số chân bạn sử dụng để gắn trực tiếp và giúp tiết kiệm được khoảng không lớn, ta thường thấy hai loại có nguồn 3.3V và 5V.

5. *Arduino Mega*

Chân số lên đến 64, 14 chân có thể phát PWM, 4 cổng truyền tiếp cùng kích thước khá lớn 5x10cm.

6. *Arduino Leonardo*

Là board không có cổng nối USB dùng lập trình. Được thiết kế tại một chip nhỏ điều khiển. Kết nối qua COM ảo và có thể kết nối với chuột và bàn phím.

7 *Arduino LilyPad*

Board mạch Lily Pad Arduino là một công nghệ dệt điện tử có thể đeo được được mở rộng bởi Leah Sang Buechley, và được thiết kế một cách cẩn thận bởi dòng Lea Leah và SparkFun. Mỗi board được thiết kế một cách tương tự với các miếng kết nối không lò & một mặt sau mịn màng để cho chúng được khâu vào quần áo bằng chỉ. Arduino này cũng bao gồm I / O, nguồn và cả board cảm biến được chế tạo đặc biệt cho hàng dệt may điện tử.

8 *Arduino RedBoard*

Board mạch RedBoard Arduino có thể được lập trình bằng cáp USB Mini-B bằng Arduino IDE. Nó sẽ hoạt động trên Windows 8 mà không phải sửa đổi cài đặt bảo mật của bạn. Nó không đòi hỏi chip USB hoặc FTDI chúng tôi sử dụng và nó hoàn toàn phẳng ở mặt sau. Tạo nó rất đơn giản để sử dụng trong thiết kế dự án. Chỉ cần cắm board, chọn tùy chọn menu để chọn Arduino UNO và bạn đã sẵn sàng để tải lên chương trình. Bạn có thể điều khiển RedBoard qua cáp USB bằng giắc cắm thủng.

Ngoài ra, còn có thể kể đến: Arduino Diecimila, Arduino Duemilanove, Arduino Due, v.v.

Ứng dụng Arduino

Arduino có nhiều ứng dụng trong đời sống, trong việc chế tạo các thiết bị điện tử chất lượng cao. Một số ứng dụng có thể kể đến như:

Lập trình robot: Arduino chính là một phần quan trọng trong trung tâm xử lý giúp điều khiển được hoạt động của robot. Lập trình máy bay không người lái. Có thể nói đây là ứng dụng có nhiều kì vọng trong tương lai.

Game tương tác: chúng ta có thể dùng Arduino để tương tác với Joystick, màn hình,... để chơi các trò như Tetrix, phá gạch, Mario... và nhiều game rất sáng tạo nữa

Arduino điều khiển thiết bị ánh sáng cảm biến tốt. Là một trong những bộ phận quan trọng trong cây đèn giao thông, các hiệu ứng đèn nháy được cài đặt làm nổi bật các biển quảng cáo.

Arduino cũng được ứng dụng trong máy in 3D và nhiều ứng dụng khác tùy thuộc vào khả năng sáng tạo của người sử dụng.

Phần mềm lập trình Arduino IDE

Arduino cung cấp đến môi trường lập trình tích hợp mã nguồn mở hỗ trợ người dùng viết code và tải nó lên bo mạch Arduino. Đây là môi trường đa nền tảng, hỗ trợ một loạt các bo mạch Arduino cùng rất nhiều tính năng độc đáo. Ứng dụng lập trình này có giao diện được sắp xếp hợp lý, phù hợp với cả những người dùng chuyên nghiệp lẫn không chuyên.

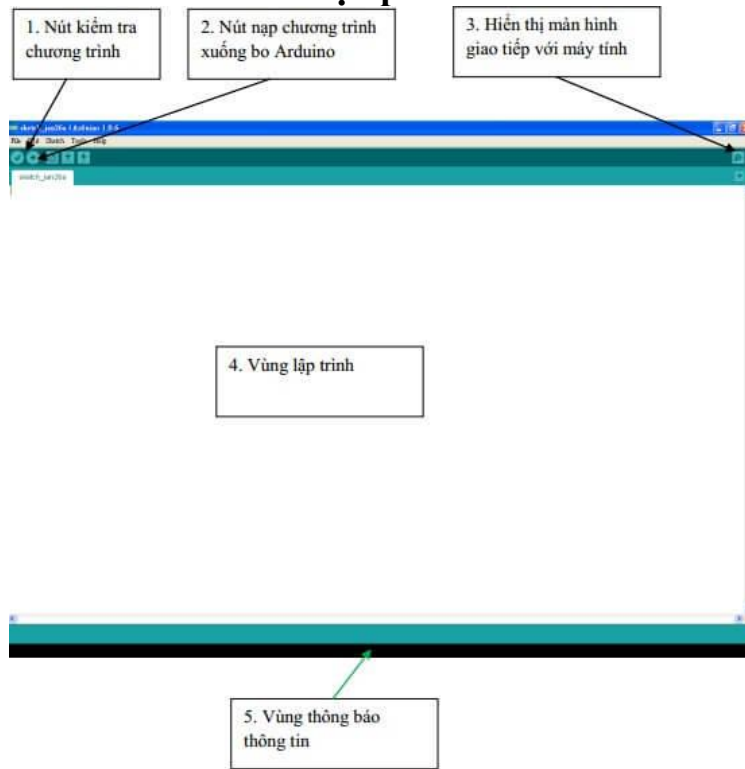
Arduino có môi trường lập trình được viết bằng java, hiện đang được sử dụng cho các bo mạch Arduino và Genuino, được nhiều công ty trên thế giới sử dụng để lập trình cho các thiết bị của họ. Java 2 Platform Standard Edition cũng là một IDE hỗ trợ Java. Hiện Java 2 Platform Standard Edition được rất nhiều người sử dụng.

Arduino là môi trường phát triển tích hợp đa nền tảng, hỗ trợ cho một loạt các bo mạch Arduino như Arduino Uno, Nano, Mega, Esplora, Ethernet, Fio, Pro hay Pro Mini cũng như LilyPad Arduino. Phần mềm này cũng phù hợp cho những lập trình viên C và C++ là thay thế hoàn hảo cho các IDE khác. Với những ai muốn học lập trình PHP, thì PHP Designer 2007 Personal là lựa chọn tốt. Phần mềm PHP Designer 2007 Personal cung cấp các giải pháp hiệu quả trong thiết kế website.

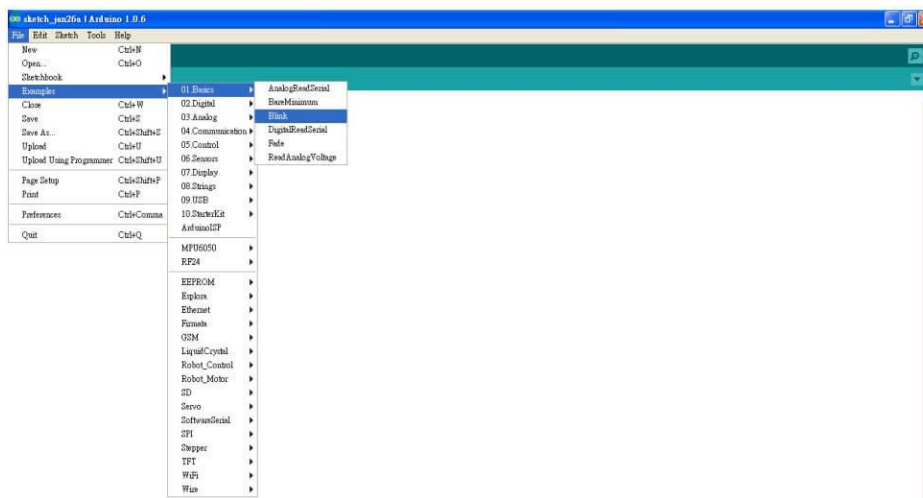
Các tính năng chính của Arduino IDE:

- Viết code cho bo mạch Arduino
- Hỗ trợ nhiều loại bo mạch Arduino
- Giao diện được sắp xếp hợp lý
- Bộ sưu tập các ví dụ mẫu
- Mảng thư viện hỗ trợ phong phú
- .v.v.

Giao diện phần mềm IDE



Giao diện của phần mềm IDE



Menu file trên phần mềm IDE

Menu file trên phần mềm IDE



Cách chọn bo trên màn hình IDE

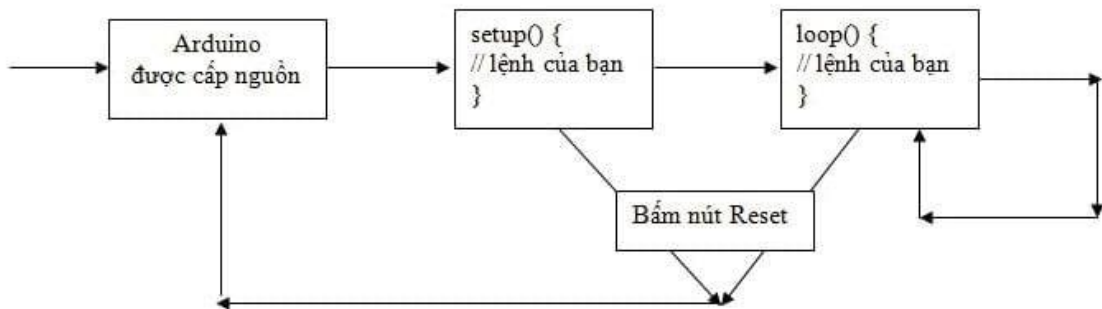
Cách trên board trên phần mềm IDE



Cách chọn cổng COM trên màn hình IDE

Cách chọn cổng COM trên màn hình IDE

Cấu trúc một chương trình trong phần mềm IDE



Tổng quan quá trình xử lý chương trình Arduino

Phần 1: Khai báo biến

Đây là phần khai báo kiểu biến, tên các biến, định nghĩa các chân trên board một số kiểu khai báo biến thông dụng: #define

Nghĩa của từ “define” là định nghĩa, hàm #define có tác dụng định nghĩa, hay còn gọi là gán, tức là gán một chân, một ngõ ra nào đó với 1 cái tên.

Ví dụ: #define led 13

Chú ý: sau #define thì không có dấu “,” (dấu phẩy)

Khai báo các kiểu biến khác như: int (kiểu số nguyên), float,...

Phần 2: Thiết lập (void setup())

Phần này dùng để thiết lập cho chương trình, cần nhớ rõ cấu trúc của nó

void setup()

```
{  
.....  
}
```

Cấu trúc của nó có dấu ngoặc nhọn ở đầu và ở cuối, nếu thiếu phần này khi kiểm tra chương trình thì chương trình sẽ báo lỗi.

Phần này dùng để thiết lập các tốc độ truyền dữ liệu, kiểu chân là chân ra hay chân vào. Trong đó:

Serial.begin(9600);	Dùng để truyền dữ liệu từ board Arduino lên máy tính
pinMode(biến, kiểu và hoặc ra) Ví dụ: pinMode(ChanDO, INPUT);	Dùng để xác định kiểu chân là đầu vào hay đầu ra

Phần 3: Vòng lặp

Dùng để viết các lệnh trong chương trình để mạch Arduino thực hiện các nhiệm vụ mà chúng ta mong muốn, thường bắt đầu bằng:

void loop()

```
{  
.....  
}
```

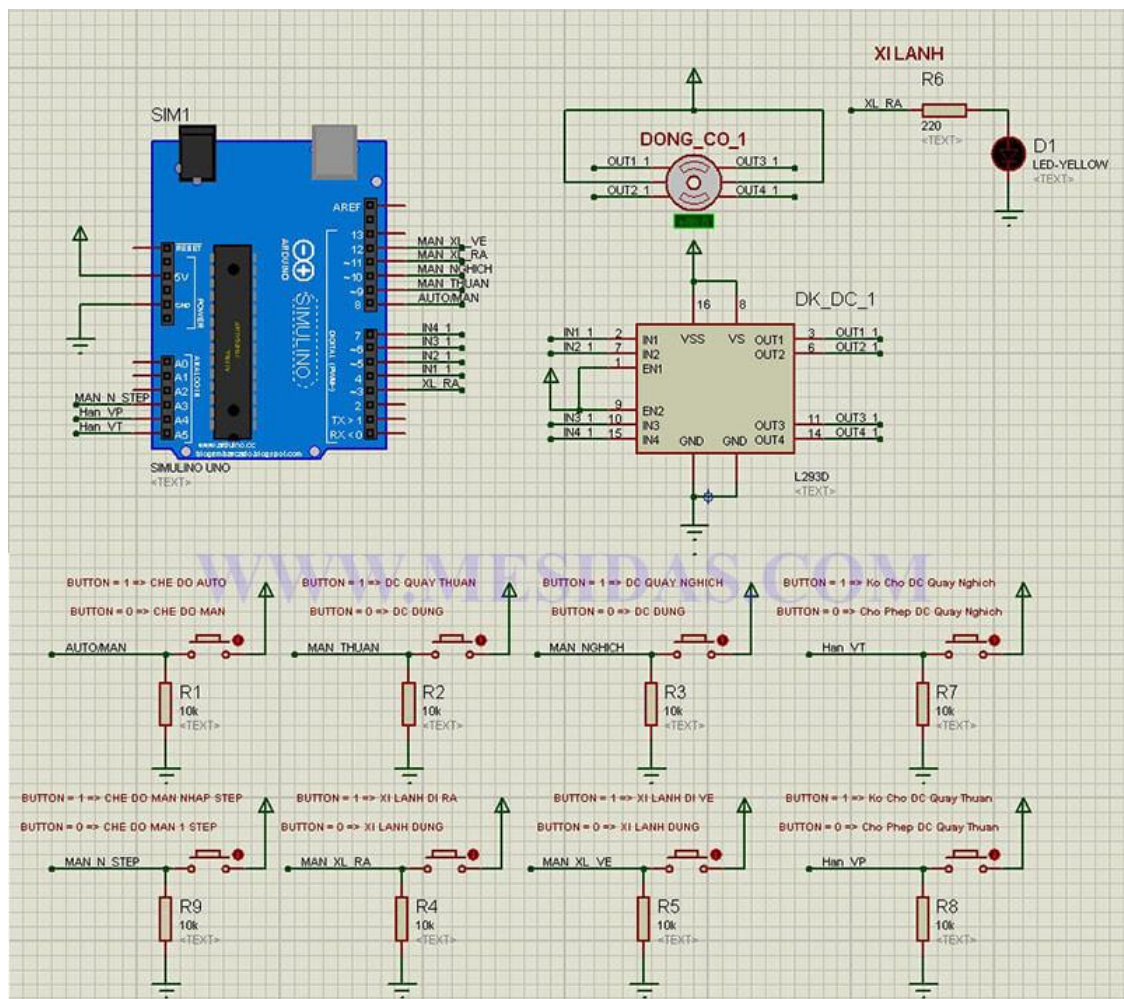
Một số ký hiệu và câu lệnh thường gặp

Ký hiệu, câu lệnh	Ý nghĩa
//	Dấu // dùng để giải thích, khi nội dung giải thích nằm trên 1 dòng, khi kiểm tra chương trình thì phần kiểm tra sẽ bỏ

	qua phần này, không kiểm tra
/* */	Ký hiệu này cũng dùng để giải thích, nhưng giải thích dành cho 1 đoạn, tức có thể xuống dòng được
#define biến chân	Define nghĩa là định nghĩa, xác định. Câu lệnh này nhằm gán tên 1 biến vào 1 chân nào đó. Ví dụ #define led 13
digitalWrite(chân, trạng thái);	Dùng để tắt, mở 1 chân ra. Cú pháp của nó là digitalWrite(chân, trạng thái chân);. Ở đây trạng thái chân có thể là HIGH hoặc LOW. Ví dụ: digitalWrite(led, HIGH); , hoặc digitalWrite(led, LOW); . Chú ý dấu chấm phẩy đằng sau câu lệnh.
analogWrite(chân, giá trị);	Có ý nghĩa dùng để băm xung (PWM), thường dùng để điều khiển tốc độ động cơ, độ sáng led,..
digitalRead(chân);	Read nghĩa là đọc, lệnh này dùng để đọc giá trị digital tại chân muốn đọc
analogRead(chân);	Read nghĩa là đọc, lệnh này dùng để đọc giá trị analog tại chân muốn đọc
delay(thời gian);	Delay nghĩa là chờ, trì hoãn, duy trì. Lệnh này dùng để duy trì trạng thái đang thực hiện chờ một thời gian. Thời gian ở đây được tính bằng mili giây, 1 giây bằng 1 ngàn mili giây.
if() { Các câu lệnh} else () { Các câu lệnh}	if nghĩa là nếu, sau if là dấu (), bên trong dấu ngoặc là một biểu thức so sánh. Ví dụ trong bài về cảm biến độ ẩm đất (phần 5) thì: if (giatriAnalog>500) //nếu giá trị đọc được của biến giatriAnalog lớn hơn 500 { digitalWrite(Led,HIGH); //Ra lệnh cho led sáng delay(1000);//chờ 1s} else nghĩa là ngược lại

Serial.print()	In ra màn hình máy tính, lệnh này in không xuống dòng
Serial.println()	In ra màn hình máy tính, in xong xuống dòng, giá trị tiếp theo sẽ được in ở dòng kế tiếp

Mô phỏng Arduino điều khiển Stepper Motor trên Proteus



Arduino điều khiển động cơ bước

Trên đây, MESIDAS đã chia sẻ cho các bạn những kiến thức tổng quan nhất về vi điều khiển Arduino. Hy vọng rằng, với những kiến thức mà MesidDas chia sẻ phía trên sẽ giúp ích được cho các bạn trong quá trình tìm hiểu, nghiên cứu, học tập và làm việc với vi điều khiển Arduino. Xin cảm ơn!

Tài liệu tham khảo:

https://mesidas.com/arduino-la/?srsltid=AfmBOooQeYetprluw8a4tuae6F4Z2K4JNJHcFlD-fAab2DvPwuqwnNLZ_