

BÀI THỰC HÀNH 8:

XÂY DỰNG HỆ THỐNG NHÚNG BẰNG NGÔN NGỮ LẬP TRÌNH ARDUINO

1. Giới thiệu về ngôn ngữ lập trình Arduino:

Arduino là một nền tảng mã nguồn mở bao gồm cả phần cứng (bo mạch vi điều khiển) và phần mềm (Arduino IDE). Nó được thiết kế để giúp lập trình viên, kỹ sư, sinh viên và người đam mê điện tử dễ dàng tạo ra các dự án nhúng, điều khiển thiết bị và tự động hóa.

Ngôn ngữ lập trình của Arduino dựa trên C/C++, với một số thư viện hỗ trợ giúp đơn giản hóa quá trình lập trình và giao tiếp với phần cứng.

2. Cấu trúc chương trình Arduino

Một chương trình Arduino (còn gọi là sketch) có hai phần chính:

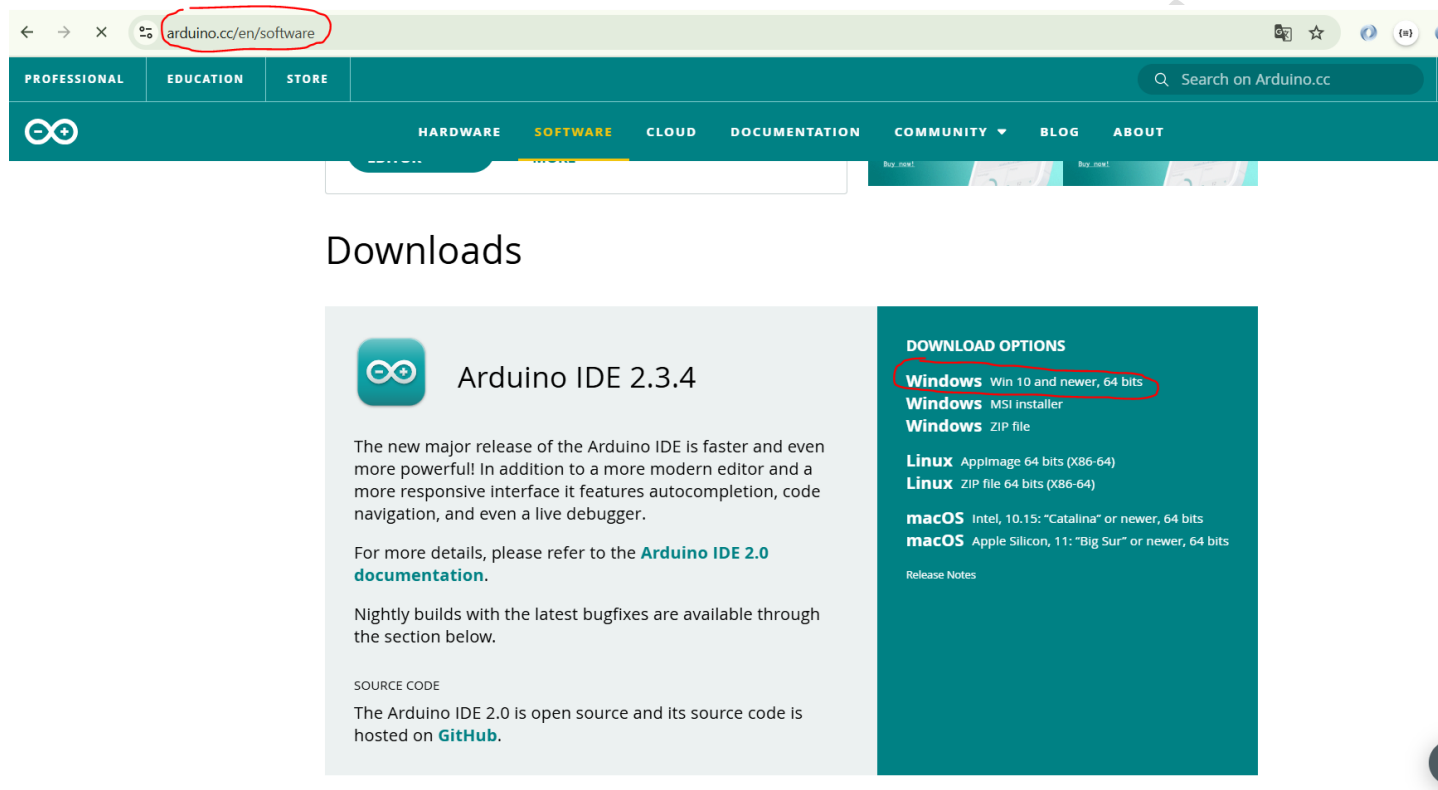
```
void setup() {  
    // Khối lệnh khởi tạo, chạy một lần khi Arduino khởi động  
}  
  
void loop() {  
    // Khối lệnh chạy lặp lại liên tục sau khi setup() thực thi xong  
}
```

Ví dụ đơn giản:

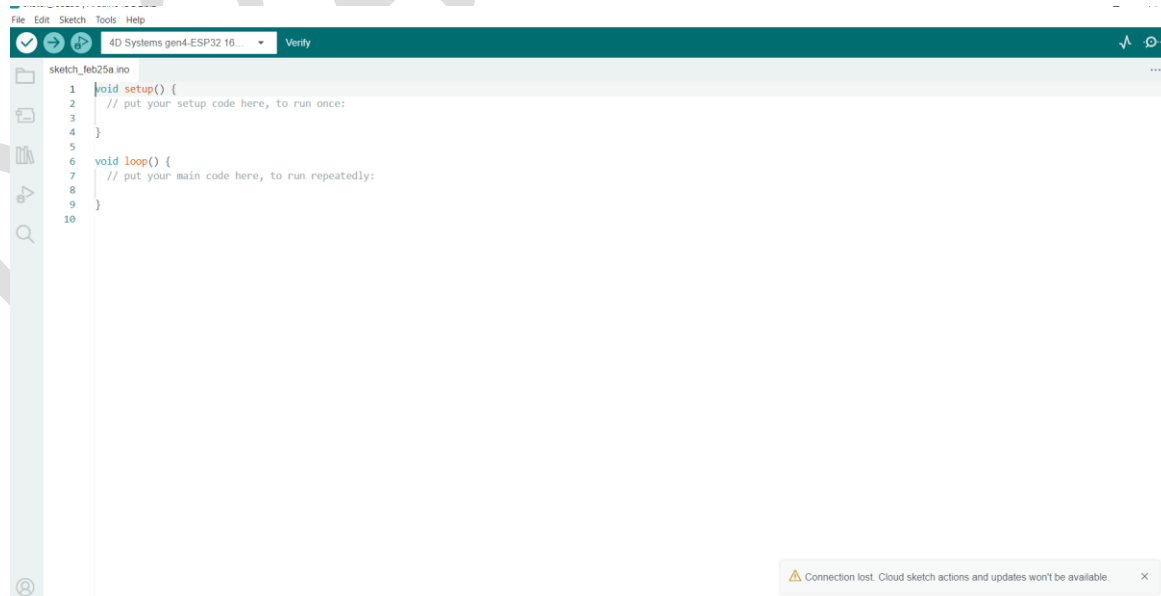
```
void setup() {  
    pinMode(13, OUTPUT); // Đặt chân số 13 là OUTPUT (xuất tín hiệu)  
}  
  
void loop() {  
    digitalWrite(13, HIGH); // Bật đèn LED  
    delay(1000);           // Đợi 1 giây  
    digitalWrite(13, LOW); // Tắt đèn LED  
    delay(1000);           // Đợi 1 giây  
}
```

Để cài đặt Arduino, chúng ta cần vào đường dẫn sau:

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

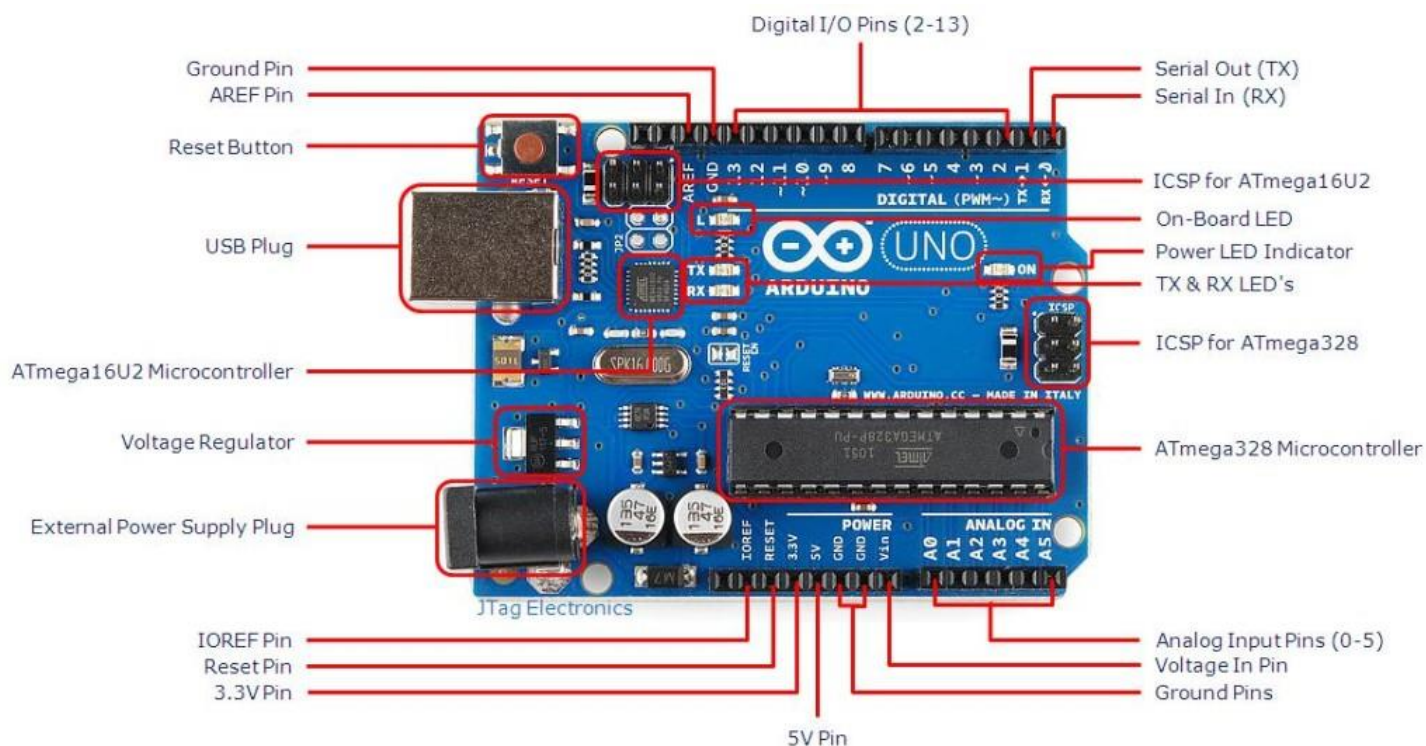


Sau khi cài đặt xong giao diện phần mềm sử dụng:



Còn giao diện phần cứng của Arduino:

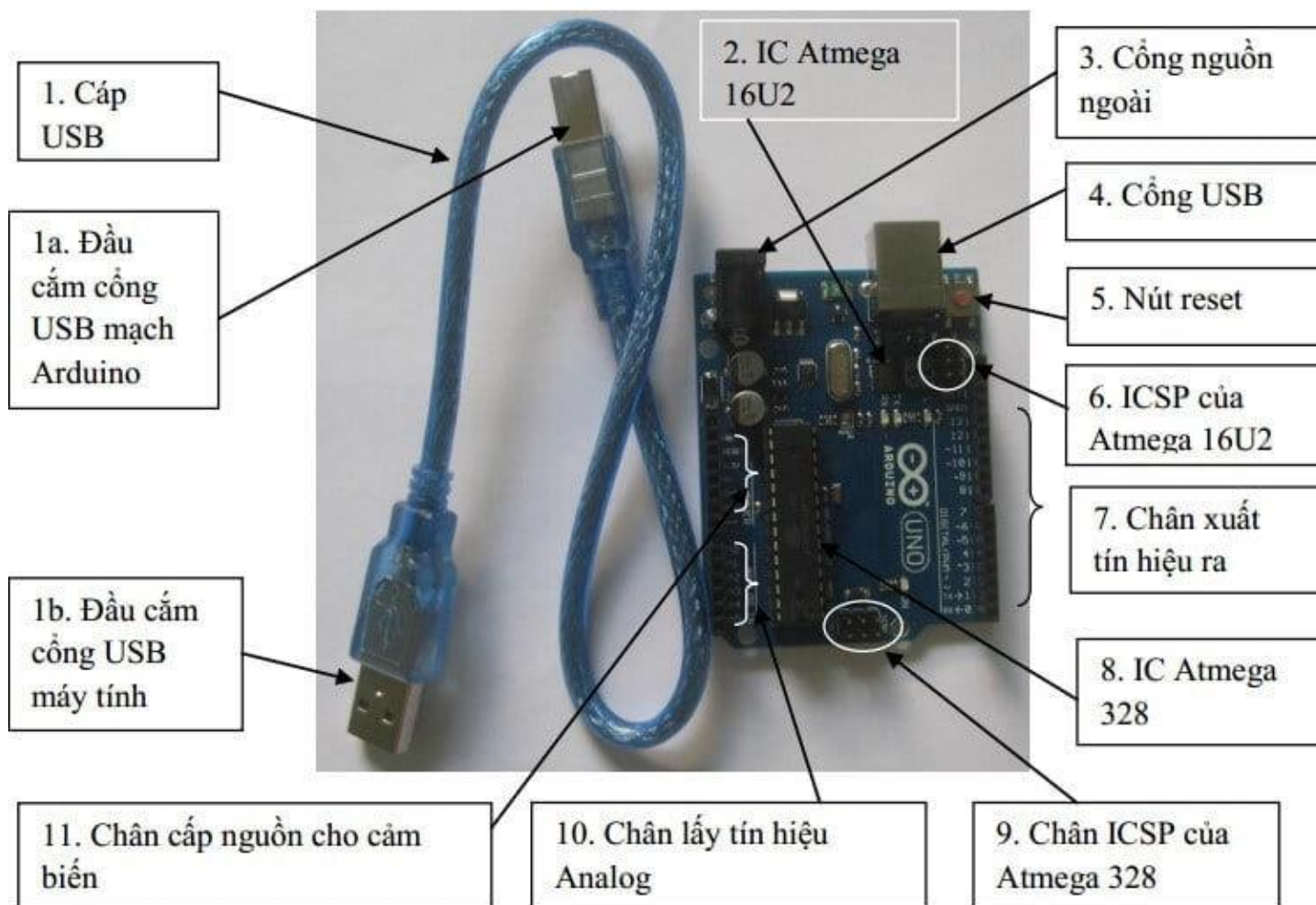
nvhach@hunre.edu.vn – 0934.345.777



Và thông số kỹ thuật của mạch

Vi điều khiển	Atmega328P
Điện áp hoạt động	5V
Điện áp cấp (hoạt động tốt)	7 – 12 V
Điện áp cấp (giới hạn)	6 – 12 V
Chân I/O digital	14 (có 6 chân xuất xung PWM)
Chân Input analog	6 (A0 – A5)
Dòng điện mỗi chân I/O	20 mA
Dòng điện chân 3.3V	50 mA
Bộ nhớ Flash	32 kB (Atmega328P) – trong đó 0.5 kB dùng cho bootloader.
SRAM	2 kB (Atmega328P)
EEPROM	1 kB (Atmega328P)
Tốc độ xung nhịp	16 MHz

Kích thước	68.6 x 53.4 mm
Trọng lượng	25 g



3. Một số biến và kiểu dữ liệu cơ bản trong lập trình Arduino

Arduino hỗ trợ các kiểu dữ liệu phổ biến như:

int: Số nguyên (VD: `int x = 10;`)

float: Số thực (VD: `float y = 3.14;`)

char: Ký tự đơn (VD: `char c = 'A';`)

String: Chuỗi ký tự (VD: `String text = "Hello";`)

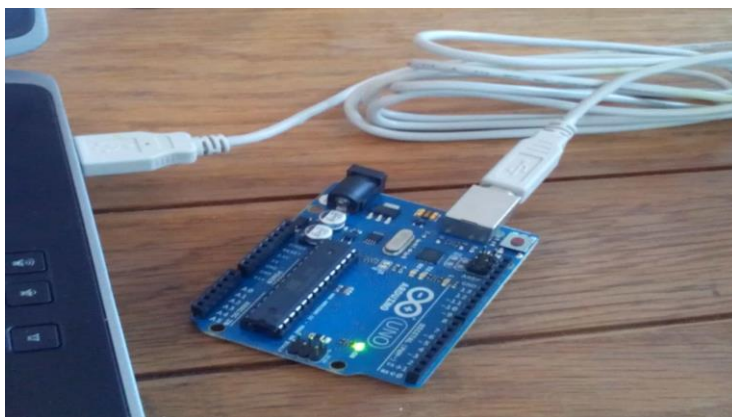
bool: Kiểu logic đúng/sai (true hoặc false)

4. Thư viện Arduino

- Arduino có nhiều thư viện giúp điều khiển cảm biến, động cơ, module giao tiếp, v.v.
- Thêm thư viện trong Arduino IDE: Vào **Sketch** → **Include Library** → **Manage Libraries**

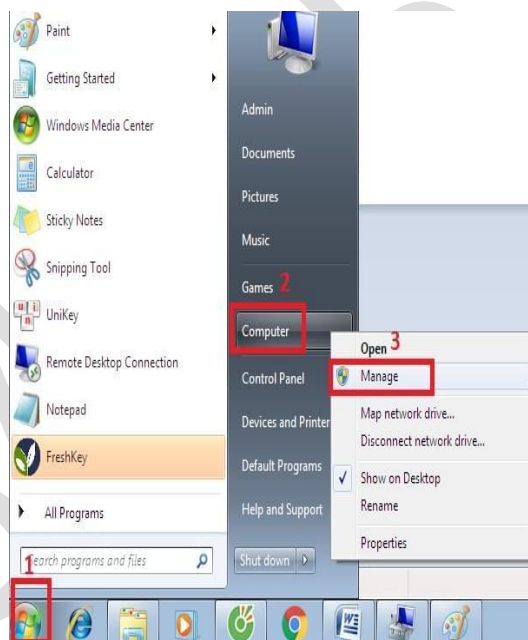
5. Nạp chương trình lên Arduino với máy tính

Bước 1: Kết nối arduino uno r3 với máy tính

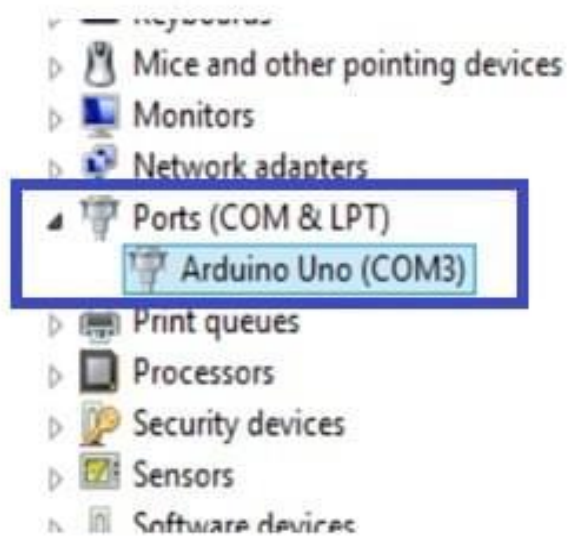


Bước 2: Tìm cổng kết nối của Arduino Uno R3 với máy tính.

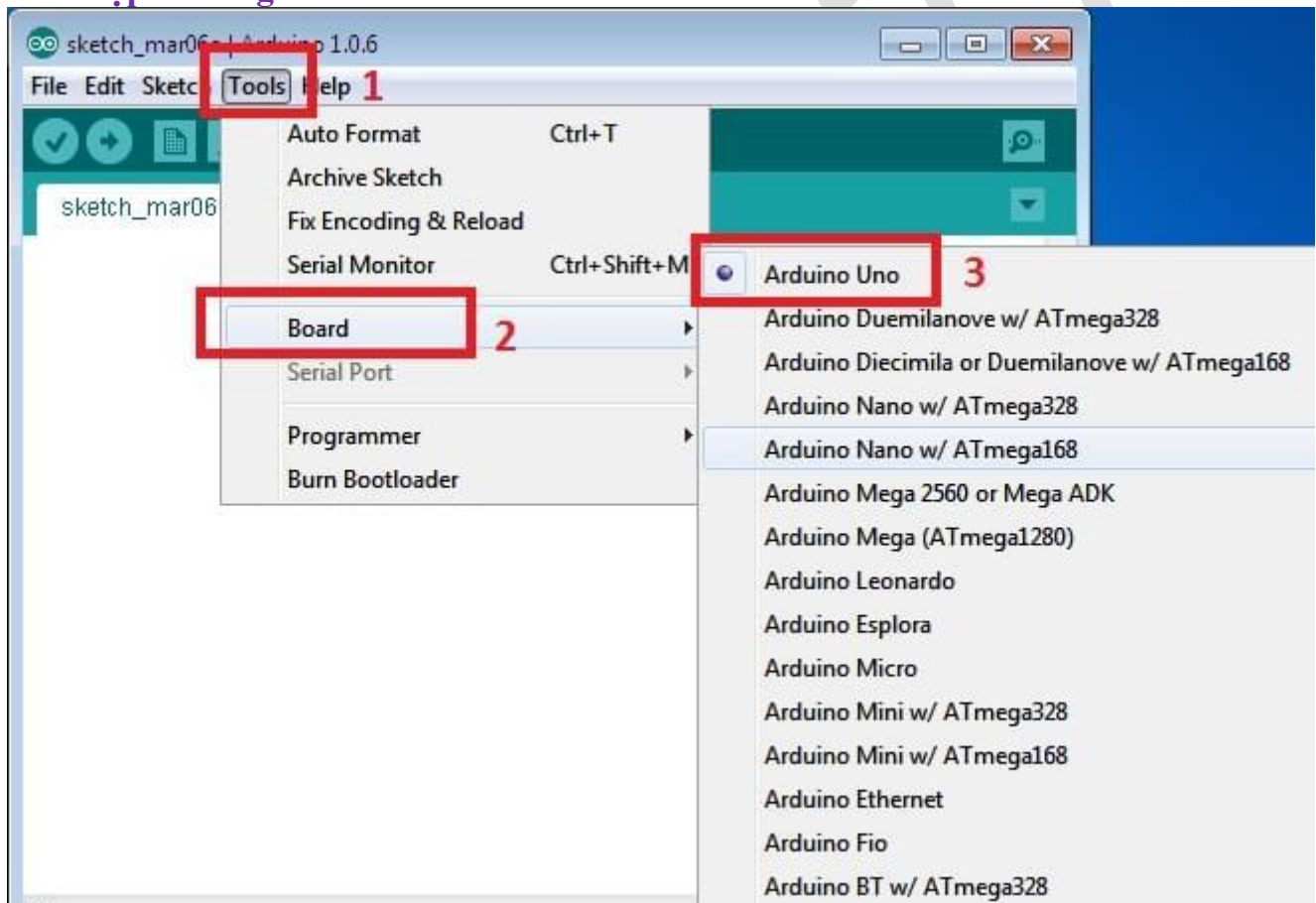
Khi Arduino Uno R3 kết nối với máy tính, nó sẽ sử dụng một cổng COM để máy tính và bo mạch có thể truyền tải dữ liệu qua lại thông qua cổng này.



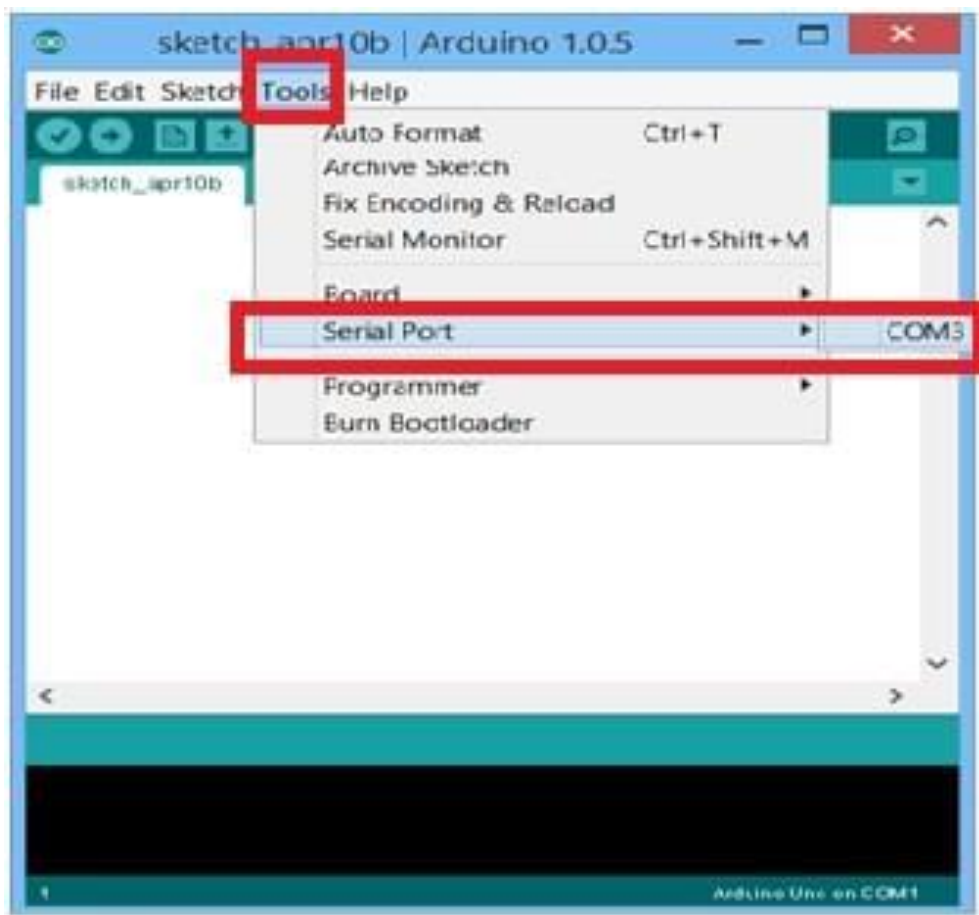
Mở mục **Ports (COM & LPT)**, chúng ta sẽ thấy cổng COM Arduino Uno R3 đang kết nối.



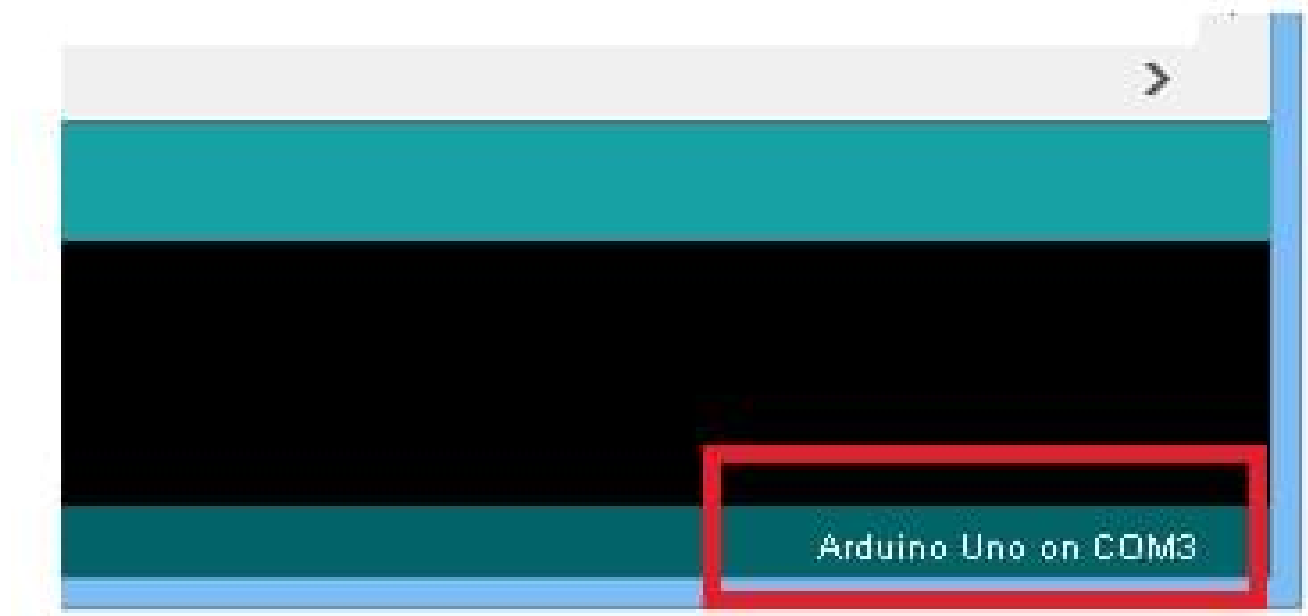
Bước 3: Nạp chương trình cho arduino uno r3.



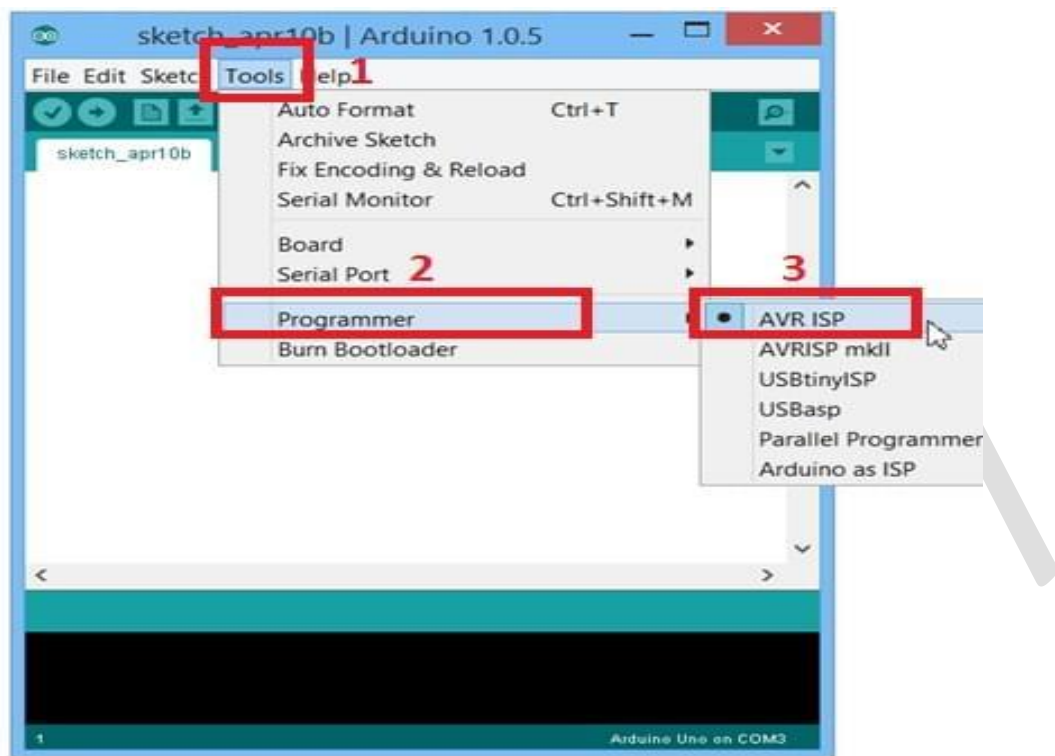
Tiếp theo vào menu Tools -> Serial Port -> chọn cổng Arduino đang kết nối với máy tính.



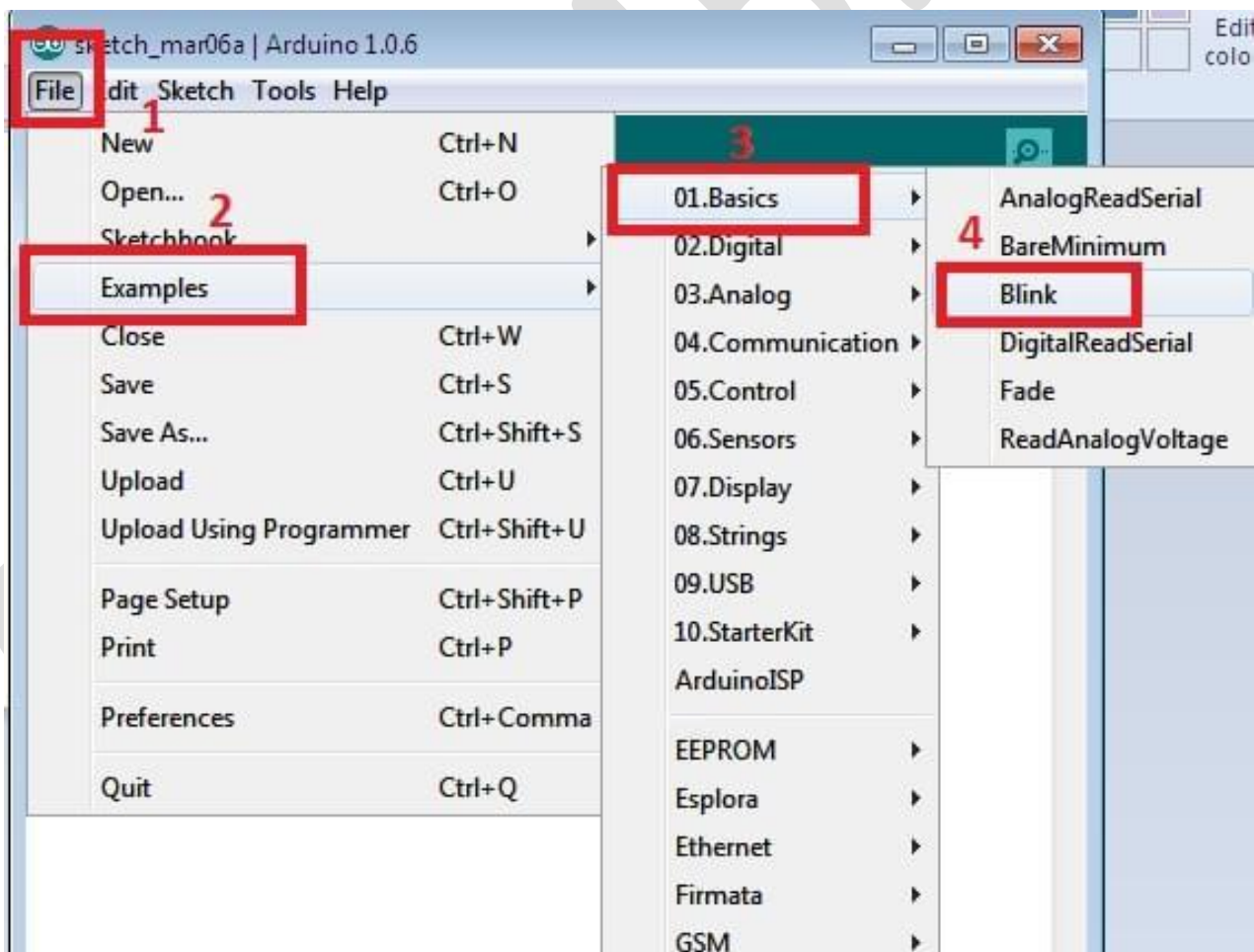
Xác nhận cổng COM của Arduino IDE ở góc dưới cùng bên phải cửa sổ làm việc.



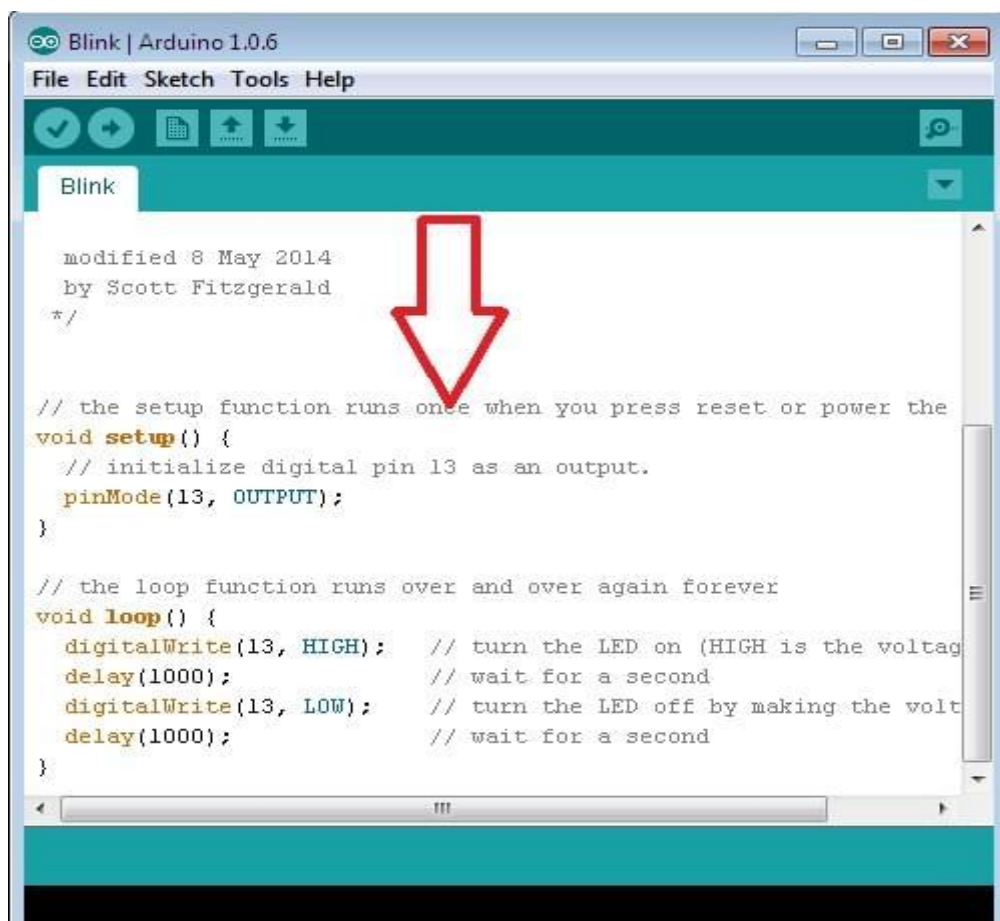
Vào menu Tools -> Programmer -> chọn AVR ISP



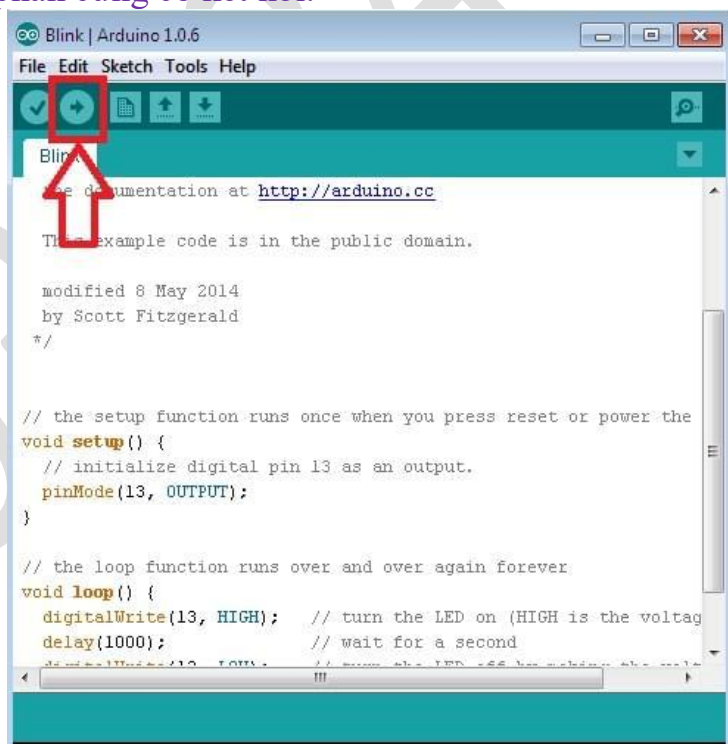
Tiếp theo









Giao diện chương trình Arduino.



Nạp chương trình cho phần cứng có kết nối.



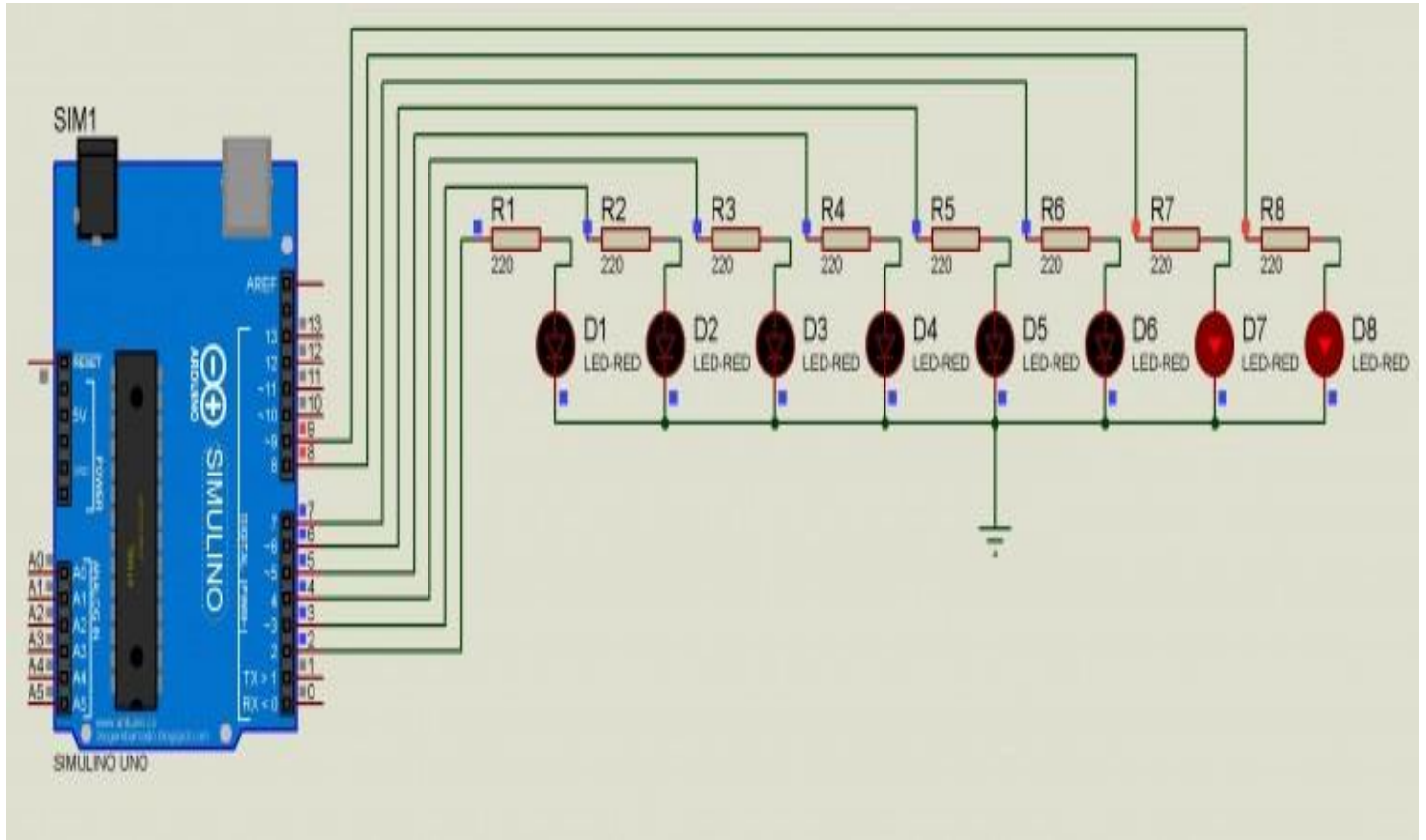
Vùng lệnh: Bao gồm các nút lệnh menu (**File, Edit, Sketch, Tools, Help**). Phía dưới là các icon cho phép sử dụng nhanh các chức năng thường dùng của IDE được miêu tả như sau:

Icon	Chức năng
	Biên dịch chương trình đang soạn thảo để kiểm tra các lỗi lập trình.
	Biên dịch và upload chương trình đang soạn thảo.
	Mở một trang soạn thảo mới.
	Mở các chương trình đã lưu.
	Lưu chương trình đang soạn.
	Mở cửa sổ Serial Monitor để gửi và nhận dữ liệu giữa máy tính và board Arduino.

THỰC HÀNH PHẦN BÀI TẬP

PHẦN A: Xây dựng hệ thống đèn Led chiếu sáng với các yêu cầu nhấp nháy cho bóng đèn (10 kiểu nhấp nháy)

(1). Thiết kế phần giao diện trên phần mềm Protues



(2) Viết Code trên phần mềm Arduino

Code tham khảo

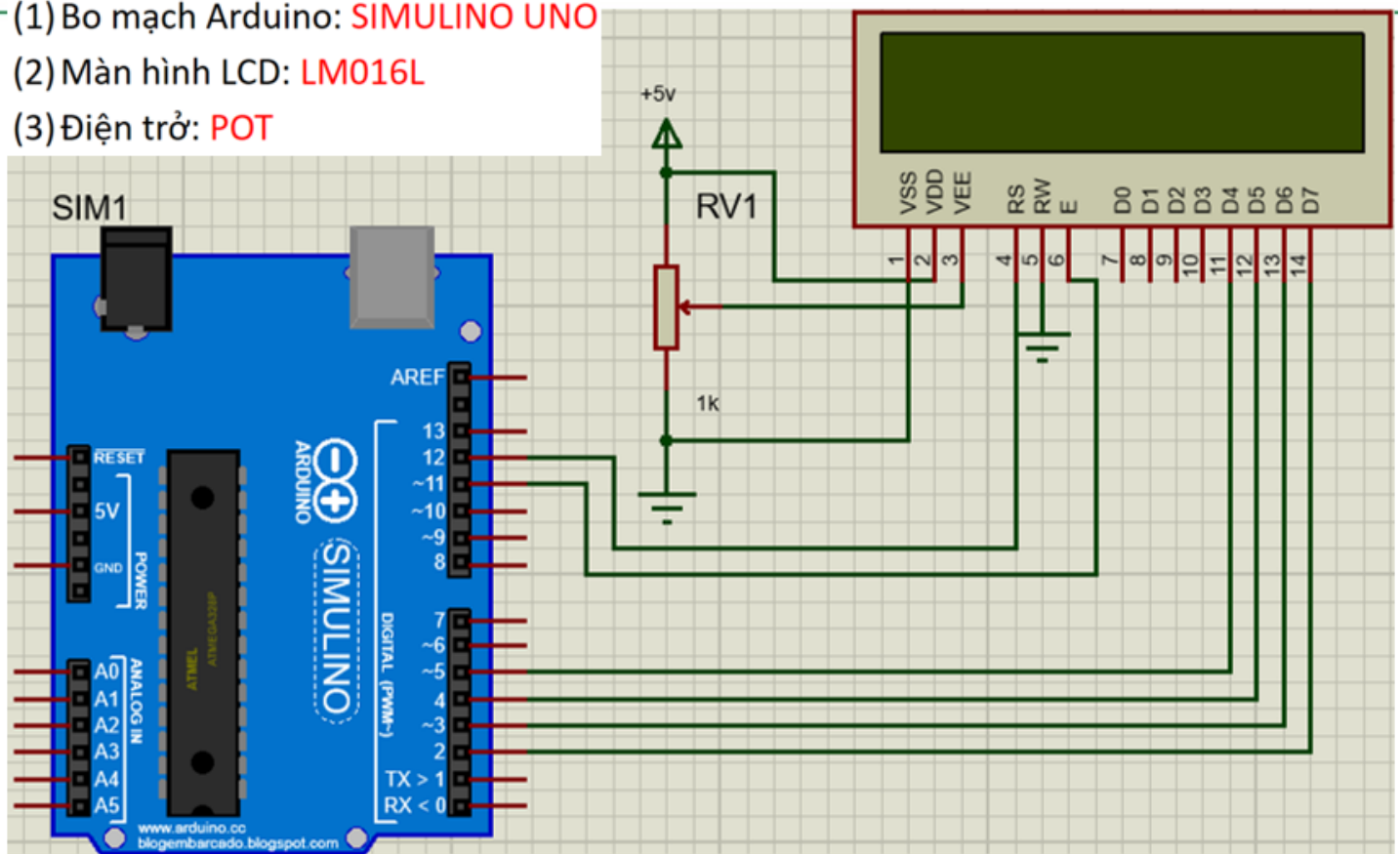
```
byte ledPin[] = {2,3,4,5,6,7,8,9,10,11};
byte pinCount; // Khai báo biến pinCount dùng cho việc lưu tổng số chân LED
void setup() {
    pinCount = sizeof(ledPin);
    for (int i=0;i<pinCount;i++) {
        pinMode(ledPin[i],OUTPUT); //Các chân LED là OUTPUT
        digitalWrite(ledPin[i],LOW); //Mặc định các đèn LED sẽ tắt
    }
}
void loop() {
    for (int i=0; i < pinCount; i++) {
        digitalWrite(ledPin[i],HIGH); //Bật đèn
        delay(500); // Dừng để các đèn LED sáng dần
    }
    for (int i = 0; i < pinCount; i++) {
```

```
digitalWrite(ledPin[i],LOW); // Tắt đèn
delay(500); // Dừng để các đèn LED tắt dần
}
}
```

PHẦN B: Xây dựng hệ thống nhúng hiển thị dữ liệu lên màn hình LCD

(1). Thiết kế phần giao diện trên phần mềm Protues

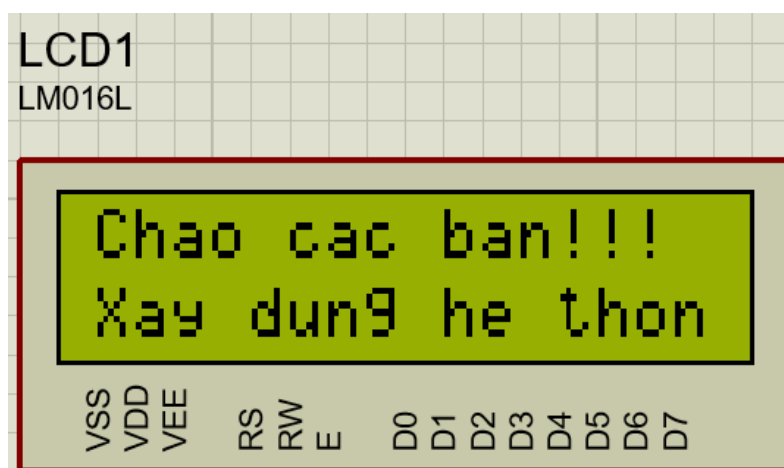
- (1) Bo mạch Arduino: **SIMULINO UNO**
- (2) Màn hình LCD: **LM016L**
- (3) Điện trở: **POT**



(2) Viết Code trên phần mềm Arduino

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12,11,5,4,3,2);
void setup() {
  lcd.begin(16,2);
  lcd.print("Chao cac ban!!!");
}
void loop() {
  lcd.setCursor(0,1);
  lcd.print("Xay dung he thong nhung");
}
```

Kết quả chạy chương trình



PHẦN C: Sinh viên xây dựng hệ thống nhúng cho hiển thị số bằng Arduino

