**Điều khiển LCD1602 bằng Arduino UNO và giao tiếp qua I2C**

**I. Giới thiệu**

Hôm nay, các bạn sẽ tìm cách điều khiển màn hình LCD bằng Arduino mà không sử dụng module I2C LCD. Vì chỉ cần Arduino và LCD thì bạn sẽ tốn khá nhiều dây để lập trình điều khiển được LCD.

**II. Mục tiêu cần nắm**

Bạn sẽ biết được:

* Pinout của một số chân quan trọng với module LCD
* Điều khiển nó bằng Arduino UNO R3

**III. Phần cứng**

1. Arduino UNO
2. Breadboard
3. Dây cắm breadboard
4. Module LCD( 16x02, 20x04,…)
5. Một biến trở( 1k hoặc 10k)

**IV. Giới thiệu sơ lược về LCD và thư viện điều khiển LCD của Arduino**

**1. Thư viện điều khiển LCD của Arduino**

Thư viện LiquidCrystal là thư viện điều khiển LCD trên Arduino, nó được xây dựng để cho các bạn có thể lập trình điều khiển các module LCD ô vuông một cách nhanh chóng mà không cần phải lập trình nhiều. Thư viện này được viết để phù hợp với con IC HD44780 (con điều khiển module LCD), tuy nhiên, trên thị trường mình toàn thấy các con LCD của Trung Quốc và thư viện này vẫn hoạt động tốt. Nghĩa là, bạn chỉ cần mua module LCD về và gắn vào Arduino, nạp code là chạy được, không cần quan tâm đến IC điều khiển LCD.

[](http://k1.arduino.vn/img/2014/10/10/0/827_1231-1412942819-0-lcd-1602.jpg)

**2. LCD pinout - sơ đồ chân của LCD**

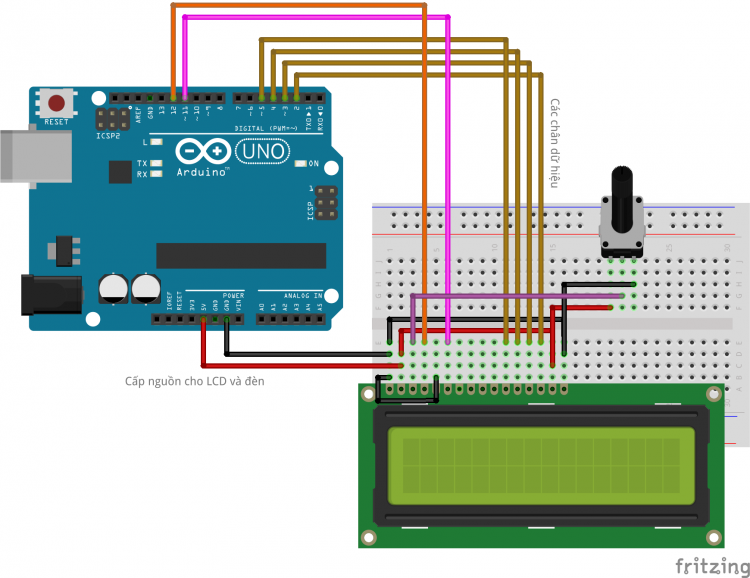
**[](http://k3.arduino.vn/img/2014/10/10/0/777_8121-1412945242-0-lcd.jpg)**

1. VSS: tương đương với GND - cực âm
2. VDD: tương đương với VCC - cực dương (5V)
3. Constrast Voltage (Vo): điều khiển độ sáng màn hình
4. Register Select (RS): điều khiển địa chỉ nào sẽ được ghi dữ liệu
5. Read/Write (RW): Bạn sẽ đọc (read mode) hay ghi (write mode) dữ liệu? Nó sẽ phụ thuộc vào bạn gửi giá trị gì vào.
6. Enable pin: Cho phép ghi vào LCD
7. D0 - D7: 8 chân dư liệu, mỗi chân sẽ có giá trị HIGH hoặc LOW nếu bạn đang ở chế độ đọc (read mode) và nó sẽ nhận giá trị HIGH hoặc LOW nếu đang ở chế độ ghi (write mode)
8. Backlight (Backlight Anode (+) và Backlight Cathode (-)): Tắt bật đèn màn hình LCD.

**3. Cách dùng thư viện với LCD**

Module LCD có thể được điều khiển ở chế độ: 4-bit điều khiển và 8-bit điều khiển. Với cách dùng ở chế độ 4-bit, bạn cần 7 chân ở Arduino, và nếu muốn dùng hết khả năng của LCD thì bạn cần 7 + 4 = 11 chân ở Arduino (không nên tí nào). Ngoài ra, mình đã thử hết các hàm trong thư viện LiquidCrystal của Arudino  ở chế độ 4-bit thì mọi thứ đều hoạt động tốt cả. Vì vậy, mình sẽ mắc mạch ở chế độ 4-bit điều khiển và hướng dẫn theo hướng này.

**V. Lắp mạch**

[](http://k2.arduino.vn/img/2015/11/23/0/1759_8822246-1448275799-0-lcd-bb.png)

**VI. Lập trình**

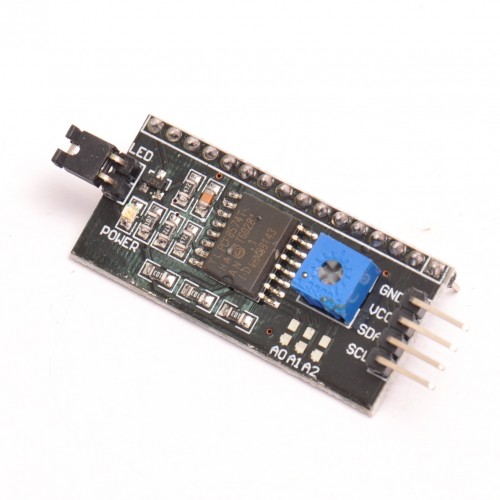
1. //Thêm thư viện LiquitCrystal - nó có sẵn vì vậy bạn không cần cài thêm gì cả
2. #include <LiquidCrystal.h>
4. //Khởi tạo với các chân
5. LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
7. void setup() {
8. //Thông báo đây là LCD 1602
9. lcd.begin(16, 2);
10. //In ra màn hình lcd dòng chữ Toi yeu Arduino
11. lcd.print("Toi yeu Arduino!");
12. }
14. void loop() {
15. // đặt con trỏ vào cột 0, dòng 1
16. // Lưu ý: dòng 1 là dòng thứ 2, lòng 0 là dòng thứ 1. Nôm na, nó đếm từ 0 từ không phải từ 1
17. lcd.setCursor(0, 1);
18. // In ra dong chu
19. lcd.print(" Arduino.VN");
20. }

## ****V. Màn hình LCD****

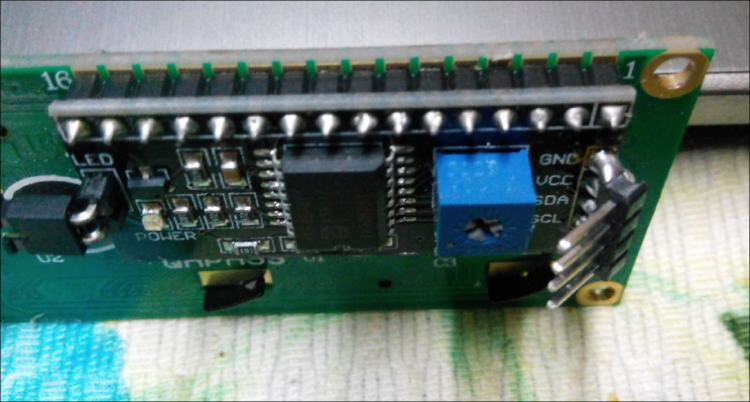
[](http://k1.arduino.vn/img/2014/05/27/0/509_12320-1401186400-0-lcd16x2-truoc-500x500.jpg)

Thông thường, để sử dụng màn hình LCD, bạn sẽ phải mất rất nhiều chân trên Arduino để điều khiển. Tham khảo tại [http://arduino.cc/en/Tutorial/Liquid...](http://arduino.cc/en/Tutorial/LiquidCrystal).

Do vậy, để đơn giản hóa công việc, người ta đã tạo ra một loại mạch điều khiển màn hình LCD sử dụng giao tiếp I2C. Nói một cách đơn giản, bạn chỉ tốn ... 2 dây để điều khiển màn hình, thay vì 8 dây như cách thông thường.

[](http://k2.arduino.vn/img/2014/05/28/0/493_88220-1401264055-0-mach-dieu-khien-man-hinh-lcd-1602-1-500x500.jpg)

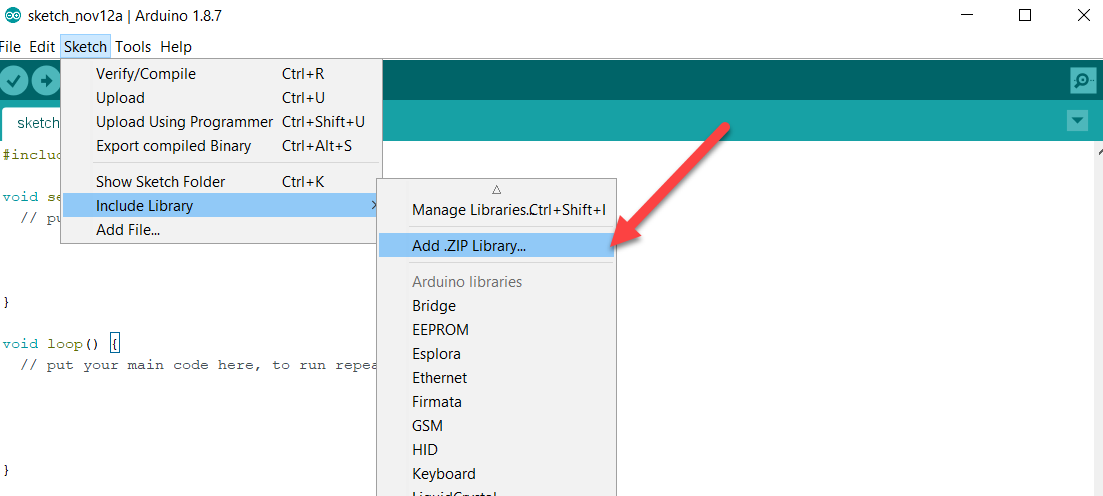
Bạn chỉ việc hàn mạch vào như thế này là xong.

[](http://k2.arduino.vn/img/2014/05/28/0/494_88220-1401265279-0-eugffgj.750x402.png)

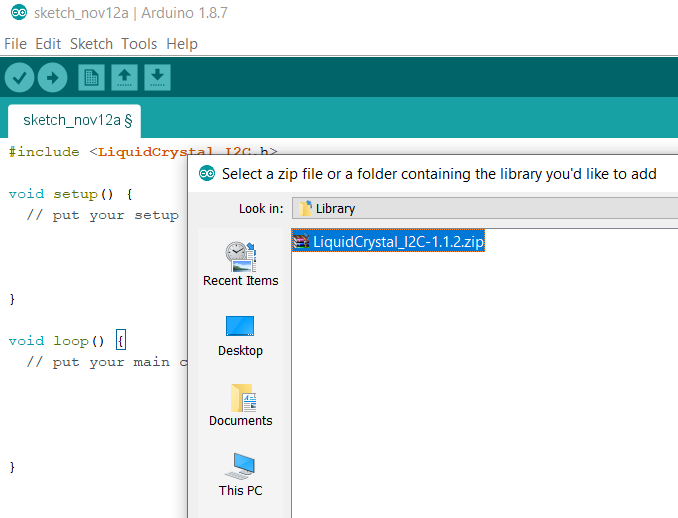
2 chân SDA và SCL là 2 chân tín hiệu dùng cho giao tiếp I2C.

**Download và cài đặt thư viện hỗ trợ sử dụng màn hình LCD qua giao tiếp I2C: *https://www.arduinolibraries.info/libraries/liquid-crystal-i2-c***

**Cách cài đặt thư viện**

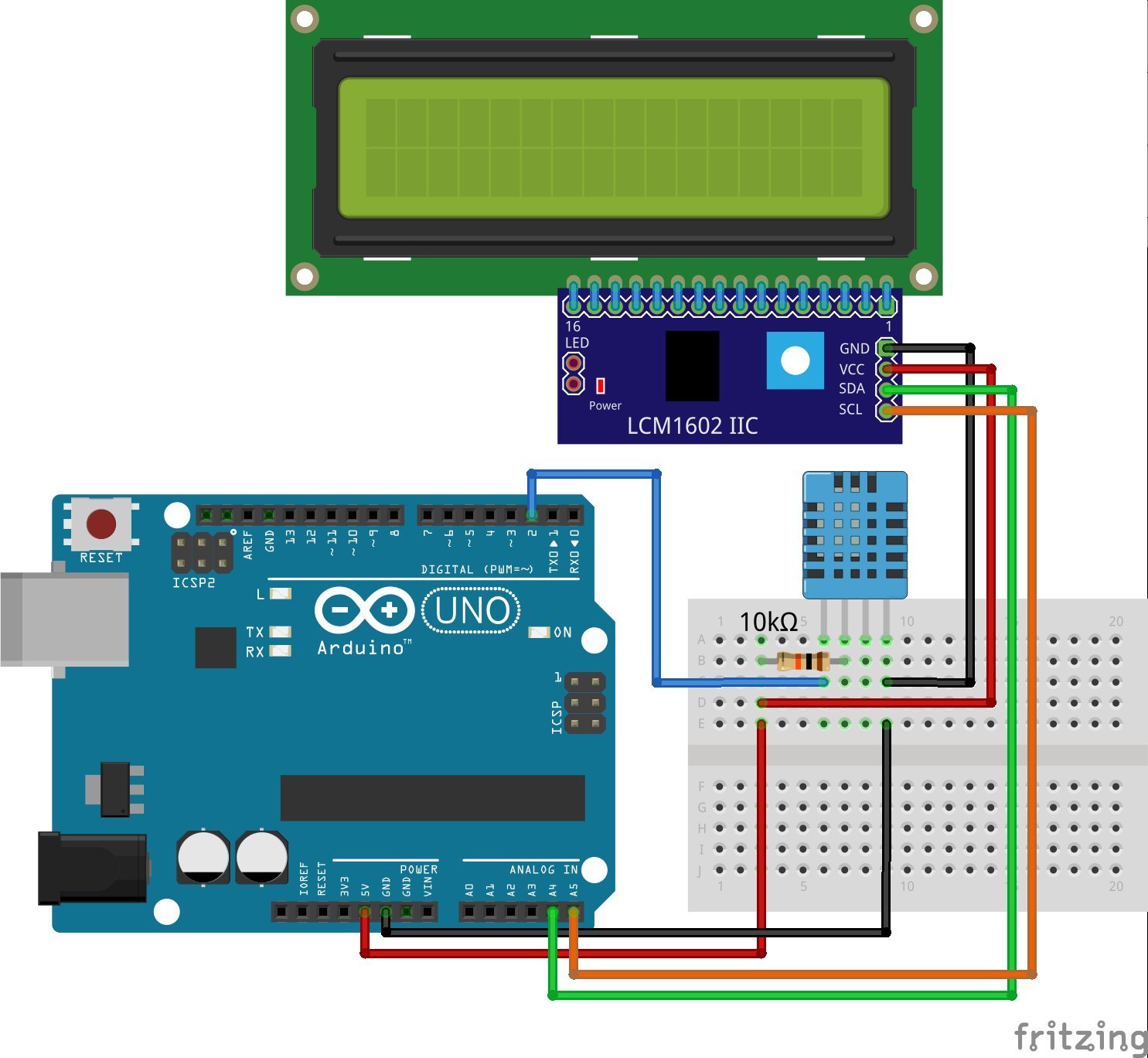


**Sau đó tìm đến file zip thư viện đã tải về**



**Kết nối module màn hình với Arduino**

|  |  |
| --- | --- |
| **Module màn hình LCD (16x2)** | **Arduino** |
| GND | GND |
| Vcc | 5V |
| SDA | A4 |
| SCL | A5 |

[](http://k2.arduino.vn/img/2016/03/02/0/2112_882450-1456920088-0-original.jpg)

**Lập trình**

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27,16,2);

//0x27 là địa chỉ màn hình trong bus I2C. Phần này chúng ta không cần phải quá bận tâm vì hầu hết màn hình (20x4,...) đều như thế này!

//16 là số cột của màn hình (nếu dùng loại màn hình 20x4) thì thay bằng 20

//2 là số dòng của màn hình (nếu dùng loại màn hình 20x4) thì thay bằng 4

void setup() {

lcd.init(); //Khởi động màn hình. Bắt đầu cho phép Arduino sử dụng màn hình, cũng giống như dht.begin() trong chương trình trên

lcd.backlight(); //Bật đèn nền

lcd.print("Hello world"); //Xuất ra chữ Hello world, mặc định sau khi init thì con trỏ tại cột 0 hàng 0 (trong C, khác với quy ước của tiếng Việt, mọi chỉ số đều bắt đầu bằng số 0, vì vậy bạn cần hiểu rằng, nếu ta kẻ một bảng có 2 hàng và 16 cột thì ô góc trên cùng bên trái là ô (0,0) tương tự với các ô khác, ta cứ tăng dần giá trị lên!

lcd.setCursor(0,1); //Đưa con trỏ tới hàng 1, cột 0

lcd.print("I love Arduino !");// Bạn thấy trên màn hình rồi chứ?

}

void loop() {

}