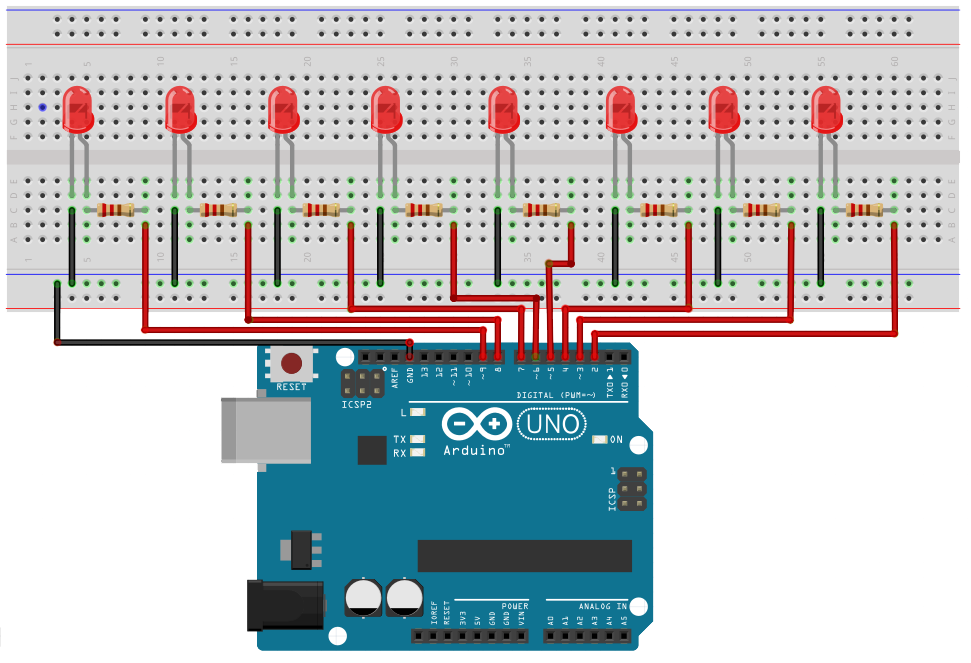
**ĐIỀU KHIỂN 8 ĐÈN LED SÁNG THEO Ý MUỐN**

## 1.Phần cứng

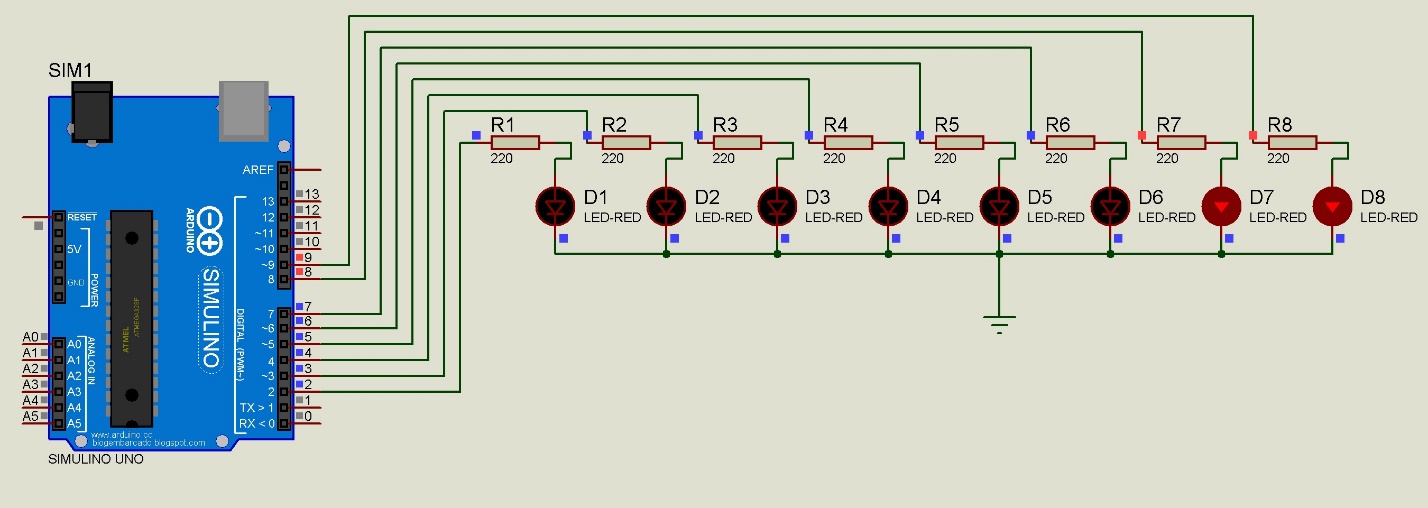
* Arduino UNO R3
* Breadboard
* Dây cắm breadboard
* 08 LED siêu sáng 5mm
* 08 điện trở 330 OhM

## 2.Điều khiển 8 LED bằng 8 chân digital của Arduino

## Bạn lắp mạch như hình vẽ nhé



Hoặc bạn có thể lắp theo cách này



Sau đây là đoạn code

byte ledPin[] = {2,3,4,5,6,7,8,9}; // Mảng lưu vị trí các chân Digital mà các đèn LED sử dụng theo thứ tự từ 1->8. Bạn có thể thêm các LED bằng cách thêm các chân digital vào mảng này

byte pinCount; // Khai báo biến pinCount dùng cho việc lưu tổng số chân LED

void setup() {

pinCount = sizeof(ledPin); //Xem thêm thông tin về hàm sizeof tại http://arduino.vn/reference/sizeof

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() {

/\*

Bật tuần tự các đèn LED

\*/

for (int i=0; i < pinCount; i++) {

digitalWrite(ledPin[i],HIGH); //Bật đèn

delay(500); // Dừng để các đèn LED sáng dần

}

/\*

Tắt tuần tự các đèn LED

\*/

for (int i = 0;i < pinCount; i += 1) {

digitalWrite(ledPin[i],LOW); // Tắt đèn

delay(500); // Dừng để các đèn LED tắt dần

}

}

## 3.Điều khiển 8 LED với IC 74hc595

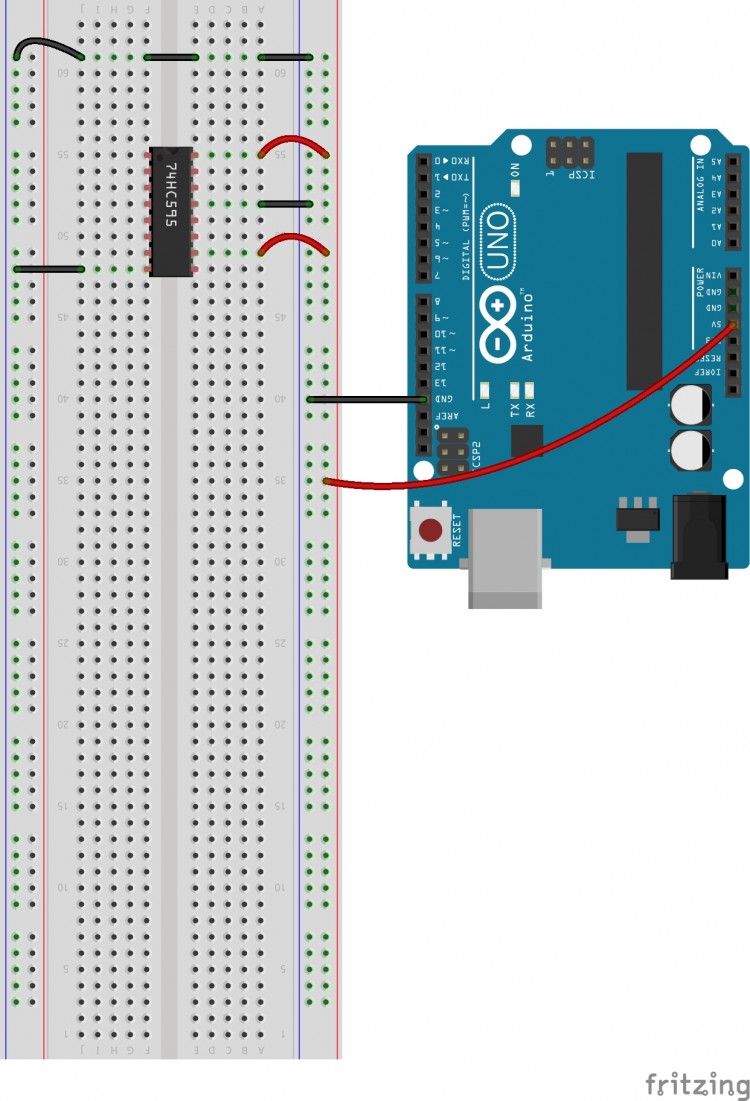
Chúng ta cần tìm hiểu về các chân của IC 74hc595 nhé:

|  |  |  |  |
| --- | --- | --- | --- |
| http://arduino.cc/en/uploads/Tutorial/595_pin_diagram.png | PINS 1-7, 15 | Q0 đến Q7 | các chân xuất tín hiệu, giống như các chân Digital được cài đặt là OUTPUT |
| PIN 8 | GND | Ground, Cực âm |
| PIN 9 | Q7" | Chân xuất ra tín hiệu Serial |
| PIN 10 | MR | Master Reclear, nối cực dương để bật IC hoạt động |
| PIN 11 | SH\_CP | Shift register clock pin |
| PIN 12 | ST\_CP | Storage register clock pin (latch pin) |
| PIN 13 | OE | Output enable, nối cực âm để các đèn LED có thể sáng được. |
| PIN 14 | DS | Serial data input |
| PIN 16 | Vcc | Cấp nguồn cho IC và LED. |

Lưu ý nho nhỏ với các bạn là các bạn hãy giữ nguyên các chân cắm của mạch 8 LED nhấp nháy chúng ta vừa làm ở trên nhé, chỉ tháo các khớp nối giữa các dây breadboard với Arduino thôi nha.

**Đầu tiên**, chúng ta cần bật IC 595 lên. Bạn hãy nối mạch như sau:

* GND (pin 8) nối đến cực âm
* Vcc (pin 16) nối đến chân 5V
* OE (pin 13) nối đến cực âm
* MR (pin 10) nối đến chân 5V

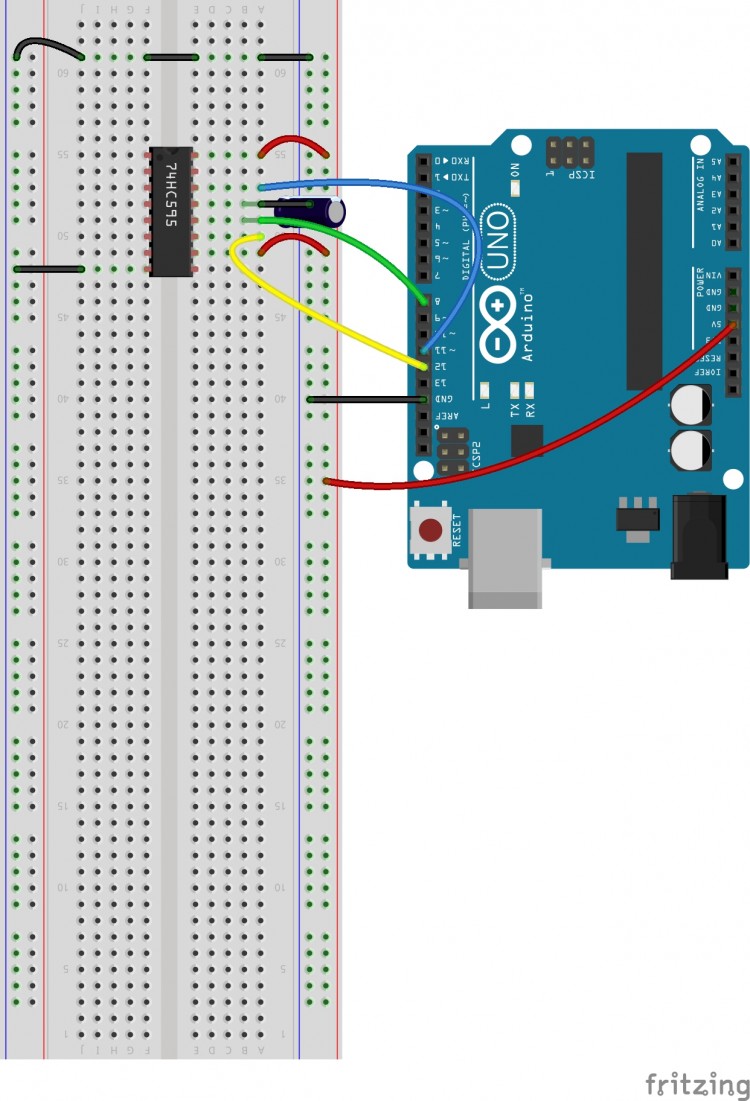
[](http://k3.arduino.vn/img/2014/06/03/0/502_8121-1401786154-0-hc-595-bb.jpg)

Tại sao lại mắc như vậy? Đó là bởi vì trong datasheet (tài liệu thông tin vi mạch) của 595 yêu cầu như vậy. Ngoài ra, bạn cũng có thể mắc các chân OE, MR bằng các chân digital của Arduino (để bật tắt các IC 595) mà thôi tôi nghĩ điều này không cần thiết đâu, với lại như vậy lại tốn thêm nhiều chân digital smiley.

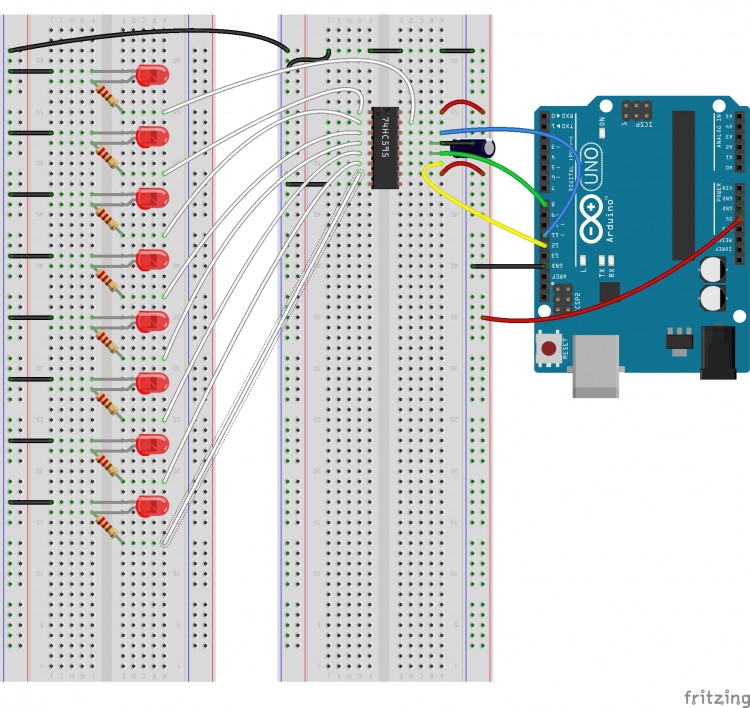
**Tiếp theo**, chúng ta sẽ nối Arduino với IC HC595.

* DS (pin 14) đến Arduino DigitalPin 11 (dây xanh nước biển)
* SH\_CP (pin 11) đến Arduino DigitalPin 12 (dây màu vàng)
* ST\_CP (pin 12) đến Arduino DigitalPin 8 (dây màu xanh lá).

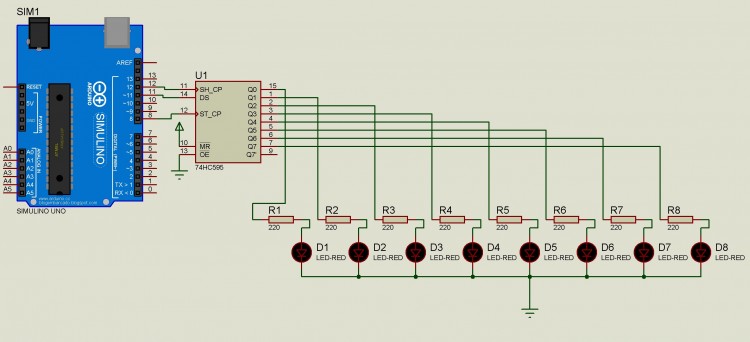
Lưu ý, vì một số IC HC 595 bị nhiễu trong quá trình shiftout. Nếu gặp tình trạng như vậy thì bạn đừng lo, chúng ta đã có cách, bạn chỉ mắc một tụ điện khoảng 0,1 uF (micro pha ra) (cực dương của tụ gắn với chân digital HC 595 và cực âm vào cực âm của nguồn).

[](http://k2.arduino.vn/img/2014/06/03/0/501_8821-1401789642-0-hc-595.fzz-step2-bb.jpg)

**Sau đó**, bạn sử dụng 8 đèn LED của chúng ta đã mắc ban đầu và mắc theo như sơ đồ dưới đây!

[](http://k1.arduino.vn/img/2014/06/03/0/536_1231-1401790521-0-hc-595.fzz-step2.fzz-step3-bb.jpg)

bạn có thể thích cách mắc này hơn

[](http://k2.arduino.vn/img/2014/08/20/0/735_8821-1408543070-0-led82.zip)

**Cuối cùng**, phần quan trọng nhất của buổi học hôm nay của chúng ta, đó là lập trình để điều khiển những con LED này!

Để điều khiển được LED qua IC HC 595, chúng ta phải làm quen với một kỹ thuật, được gọi là shiftOut. Nói vậy thôi chứ không có khó đâu, bạn chỉ cần hiểu đơn giản, shiftOut là việc gửi tín hiệu cho 1 IC có hỗ trợ shiftOut (ví dụ HC 595 này), cứ mỗi lần gửi nó gửi 1 bye (không hơn không kém), mỗi 1 bit (có tổng cộng 8 bit trong 1 byte) sẽ quản lý giá trị điện tại chân tín hiệu của HC 595 (các chân có tên là Q0-Q7).

Bây giờ để dễ hình dung, chúng ta sẽ tưởng tương như thế này:

1. Các LED sẽ sáng nếu được cấp điện thế dương (5V chẳng hạn) vào cực dương và cực âm của LED được gắn vào cực âm của nguồn (hiển nhiên rồi), vì vậy chúng ta sẽ gắn cực âm của LED vào nguồn và nếu muốn điều khiển cho đèn LED sáng thì chỉ quản lý điện thế gắn vào cực dương của LED, nếu là âm thì đèn tắt, nếu là dương thì đèn sáng.
2. Còn IC 595 sẽ nhận lệnh shiftOut của Arduino và sẽ quản lý 8 chân tín hiệu của nó (các chân có tên Q0-Q7) bằng 1 byte ⇔ 8 bit (từ bit 0 đến bit 7). Nếu tại bit 0 có giá trị là 1 ⇒ Q0 được nối với nguồn 5V, nếu bit 0 có giá trị là 0 thì Q0 sẽ được nối tới cực âm, tương tự với các 7 bit còn lại.
3. Bây giờ, chúng ta sẽ dùng shiftOut để quản lý hiệu điện thế các chân từ Q0 đến Q7 của IC HC595, từ đó, điều khiển việc bật / tắt đèn.

Lợi ích của việc này, đó là bạn chỉ cần 3 chân digital như đã hướng dẫn mắc ở trên là có thể điều khiển được bao nhiêu LED tùy ý (cứ mỗi con HC 595 điều khiển được tối ta 8 LED). Để điều khiển nhiều LED, chúng ta chỉ việc mắc thêm các bé 595 vào thôi, mắc như thế nào chúng ta sẽ biết ngay thôi smiley.

Bạn copy đoạn code sau và dán vào bên trong Arduino IDE, trong này, tôi có hướng dẫn bạn shiftOut để thực hiện lại quá trình đèn LED như lúc chúng ta sử dụng 8 chân digital.

/\*

shiftOut với 8 LED bằng 1 IC HC595

\*/

//chân ST\_CP của 74HC595

int latchPin = 8;

//chân SH\_CP của 74HC595

int clockPin = 12;

//Chân DS của 74HC595

int dataPin = 11;

//Trạng thái của LED, hay chính là byte mà ta sẽ gửi qua shiftOut

byte ledStatus;

void setup() {

//Bạn BUỘC PHẢI pinMode các chân này là OUTPUT

pinMode(latchPin, OUTPUT);

pinMode(clockPin, OUTPUT);

pinMode(dataPin, OUTPUT);

}

void loop() {

/\*

Trong tin học, ngoài các phép +, -, \*, / hay % mà bạn đã biết trên hệ cơ số 10.

Thì còn có nhiều phép tính khác nữa. Và một trong số đó là Bit Math (toán bit) trên hệ cơ số 2.

Để hiểu những gì tôi viết tiếp theo sau, bạn cần có kiến thức về Bit Math.

Để tìm hiểu về Bit Math, bạn vào mục Tài liệu tham khảo ở bảng chọn nằm phía trên cùng trang web và chạy xuống khi bạn kéo chuột trên trang Arduino.VN

\*/

//Sáng tuần tự

ledStatus = 0;//mặc định là không có đèn nào sáng hết (0 = 0b00000000)

for (int i = 0; i < 8; i++) {

ledStatus = (ledStatus << 1) | 1;//Đẩy toàn bộ các bit qua trái 1 bit và cộng bit có giá trị là 1 ở bit 0

/\*\*

Bắt buộc phải có để shiftOut

\*\*/

digitalWrite(latchPin, LOW); //các đèn LED sẽ không sáng khi bạn digital LOW

//ShiftOut ra IC

shiftOut(dataPin, clockPin, MSBFIRST, ledStatus);

digitalWrite(latchPin, HIGH);//các đèn LED sẽ sáng với trạng thái vừa được cập nhập

/\*\*

Kết thúc bắt buộc phải có

\*\*/

delay(500); // Dừng chương trình khoảng 500 mili giây để thấy các hiệu ứng của đèn LED

}

//Tắt tuần tự

for (int i = 0;i<8;i++) {

ledStatus <<= 1; //Đẩy tất cả các bit qua bên trái 1 bit

digitalWrite(latchPin, LOW);

shiftOut(dataPin, clockPin, MSBFIRST, ledStatus);

digitalWrite(latchPin, HIGH);

delay(500);

}

}

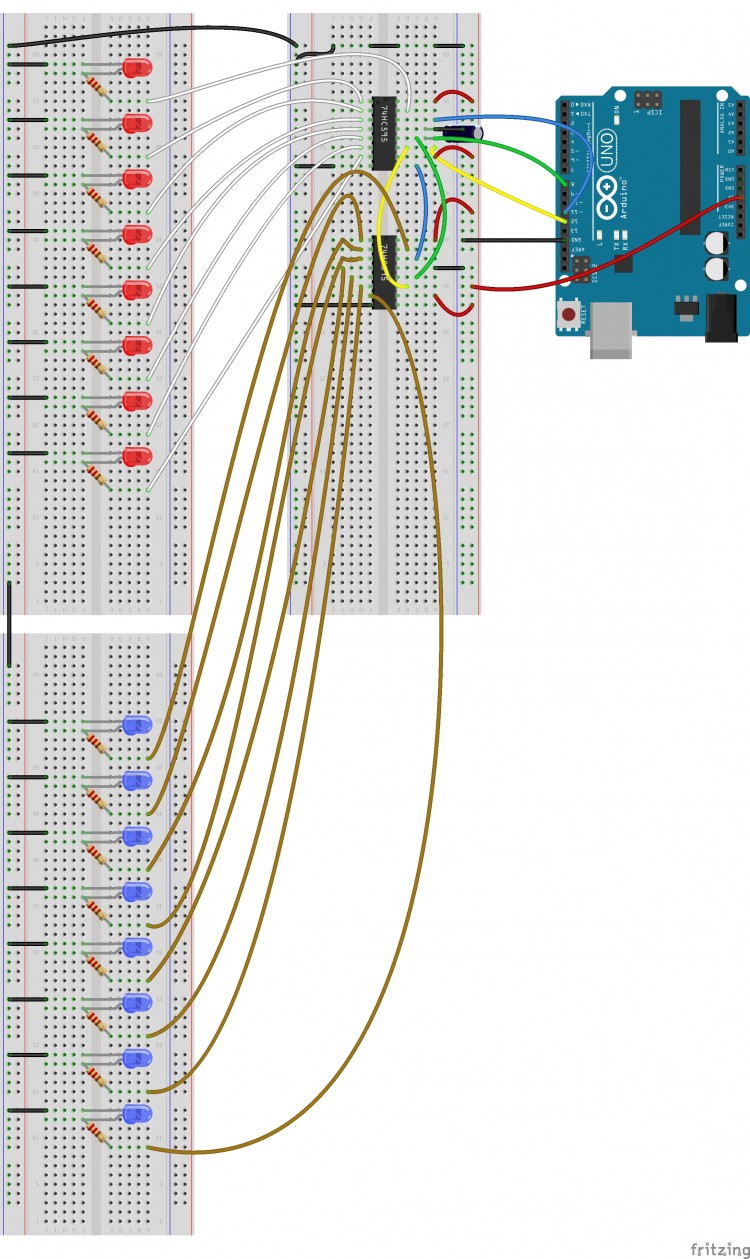
Bạn thấy đấy, đoạn bắt buộc phải có để shiftOut chúng ta có thể thay thế bằng một hàm tự tạo void.

### **3. Điều khiển nhiều đèn LED (số lượng LED chỉ phụ thuộc vào mức độ chịu khó của bạn) bằng Arduino với nhiều IC HC 595**

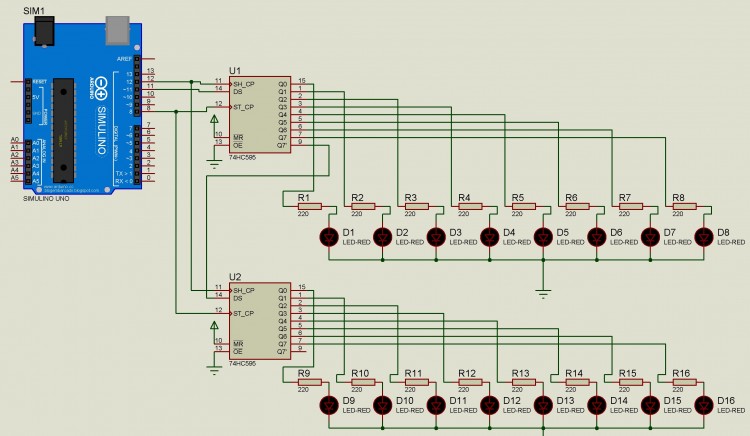
Như bạn đã biết, cứ mỗi một con HC 595 thì chúng ta sẽ điều khiển được tối đa 8 LED (mà tôi khuyên bạn nên thiết kế với số LED chia hết cho 8). Vậy theo lý thuyết và thực tế là như vậy, chỉ cần nhiều 595 là chúng ta làm được nhiều LED hơn. Cái khó, mà không, cũng không khó, chỉ cần hiểu sơ sơ là bạn sẽ làm được LED như mấy tấm biển quảng cáo rồi! Trước hết, bạn cần đọc nguyên lý mà tôi đã soạn để bạn - một người mới nghiên cứu có thể hiểu và làm được.

1. Chỉ IC HC595 đầu tiên mắc như những gì nói ở trên, kể từ IC HC595 thứ hai trở đi, các bạn mắc như những gì tôi nói ở đây và theo trình tự như sau: IC 2 nối với IC 1, IC 3 nối với IC 2, IC 4 nối với IC 3, và cứ như thế IC mắc sau cùng sẽ mắc vào IC vừa được mắc trước đó.
2. Cứ mỗi lần shiftOut, thì byte đầu tiên sẽ đến IC HC 595 cuối cùng, byte thứ hai sẽ đến IC HC595 thêm trước IC HC595 đó và cứ như thế nhé.
3. Cứ mỗi lần shiftOut ra trạng thái các LED, bạn buộc phải shiftOut hết cho đến shiftOut đầu tiên (vì shiftOut lần 1 sẽ tới cái IC cuối cùng). Nếu chưa rành, bạn hãy sử dụng hàm void shiftOutHC595 để shiftOut mỗi khi bạn muốn cập nhập trạng thái của mình.
4. Mệt đừng nản, hãy kiên trì!

Nào, chúng ta cùng lắp mạch thôi, những IC HC 595 thêm mới chúng ta vẫn lắp như mục thứ 3 ở trên. Nhưng có một thay đổi nhỏ xíu, đó là thay vì nối chân số 14 của HC 595 với chân digital 11 của Arduino mà chúng ta phải nối vào chân số 9 của IC 595 trước đó. Bạn hãy tham khảo mạch sau để rõ hơn.

[](http://k3.arduino.vn/img/2014/06/03/0/504_8121-1401808991-0-hc-595.fzz-step2.fzz-step3.fzzstep4-bb.jpg)

hoăc bạn có thể thích cách mắc này hơn

[](http://k2.arduino.vn/img/2014/08/20/0/736_8821-1408543707-0-led8-3.jpg)

Việc cuối cùng bạn cần làm là dán đoạn code này vào Arduino IDE, sau đó đọc từng dòng tôi comment trong này nhé devil!

1. /\*
2. shiftOut với 8 LED bằng 1 IC HC595
3. \*/
4. //chân ST\_CP của 74HC595
5. int latchPin = 8;
6. //chân SH\_CP của 74HC595
7. int clockPin = 12;
8. //Chân DS của 74HC595
9. int dataPin = 11;
11. //Trạng thái của LED, hay chính là byte mà ta sẽ gửi qua shiftOut
12. const int HC595\_COUNT = 2;//Nếu bạn dùng nhiều hơn thì thay bằng một số lớn hơn 2.
13. byte ledStatus[HC595\_COUNT]= {0};
14. void setup() {
15. //Bạn BUỘC PHẢI pinMode các chân này là OUTPUT
16. pinMode(latchPin, OUTPUT);
17. pinMode(clockPin, OUTPUT);
18. pinMode(dataPin, OUTPUT);
19. }
21. void fillValueToArray(byte value) {
22. for (int i = 0;i < HC595\_COUNT; i += 1) {
23. ledStatus[i] = value;
24. }
25. }
27. void shiftOutHC595(int dataPin, int clockPin, byte ledStatus[]) {
28. digitalWrite(latchPin, LOW);
30. for (int i = 0; i < HC595\_COUNT; i++) {
31. shiftOut(dataPin,clockPin,LSBFIRST,ledStatus[i]); // Chúng ta dùng LSBFIRST thay cho MSBFIRST là có lý do của nó, cái này tôi không biết trình như thế nào để bạn hiểu rõ nhất
32. // Tốt nhất bạn hãy thay LSBFIRST thành MSBFIRST để rõ ràng những gì tôi muốn nói.
33. }
35. digitalWrite(latchPin, HIGH);
36. }
38. void loop() {
39. /\*
40. Trong tin học, ngoài các phép +, -, \*, / hay % mà bạn đã biết trên hệ cơ số 10.
41. Thì còn có nhiều phép tính khác nữa. Và một trong số đó là Bit Math (toán bit) trên hệ cơ số 2.
42. Để hiểu những gì tôi viết tiếp theo sau, bạn cần có kiến thức về Bit Math.
43. Để tìm hiểu về Bit Math, bạn vào mục Tài liệu tham khảo ở bảng chọn nằm phía trên cùng trang web và chạy xuống khi bạn kéo chuột trên trang Arduino.VN
44. \*/
45. //Sáng tuần tự
47. //vì ledStatus là một mảng vì vậy để mặc định tất cả đèn tắt thì chúng ta phải for đến từng giá trị của mảng rồi đặt giá trị là 0.
48. fillValueToArray(0);
49. //Bật tuần tự
50. for (int i = 0; i < HC595\_COUNT; i++) {
51. for (byte j=0;j<8;j++) {
52. ledStatus[i] = (ledStatus[i] << 1) | 1;
53. shiftOutHC595(dataPin,clockPin,ledStatus);
54. delay(100); // Dừng chương trình khoảng 500 mili giây để thấy các hiệu ứng của đèn LED
55. }
56. }
58. //Tắt tuần tự
59. for (int i = 0; i < HC595\_COUNT; i++) {
60. for (byte j=0;j<8;j++) {
61. ledStatus[i] = (ledStatus[i] << 1);
62. shiftOutHC595(dataPin,clockPin,ledStatus);
63. delay(100); // Dừng chương trình khoảng 500 mili giây để thấy các hiệu ứng của đèn LED
64. }
65. }
67. /\*
68. Một số thuật toán nhấp nháy khác, tôi chia sẻ với các bạn đây, hãy làm một cái gì đấy tặng người thân yêu mình nhé.
69. Cộng đồng của chúng ta không chỉ hướng tới Arduino, mà còn hướng tới cuộc sống tinh thần của các bạn nữa :)
70. \*/
72. //Nhấp nháy nhanh
73. for (byte k = 0; k < 20; k++) {
74. fillValueToArray(0b10101010);
75. shiftOutHC595(dataPin,clockPin,ledStatus);
76. delay(50);
77. fillValueToArray(0b01010101);
78. shiftOutHC595(dataPin,clockPin,ledStatus);
79. delay(50);
80. }
82. //sáng 1 đèn rồi cách 1 đèn ko sáng rồi lại sáng những đèn chưa bật
83. fillValueToArray(0);
84. for (int i = 0; i < HC595\_COUNT; i++) {
85. for (byte j = 0;j<8;j += 2) {
86. ledStatus[i] |= 1 << j;
87. shiftOutHC595(dataPin,clockPin,ledStatus);
88. delay(200);
89. }
90. }
91. for (int i = 0; i < HC595\_COUNT; i++) {
92. for (byte j = 1;j<8;j += 2) {
93. ledStatus[i] |= 1 << j;
94. shiftOutHC595(dataPin,clockPin,ledStatus);
95. delay(200);
96. }
97. }
99. //Tắt dần theo thứ tự trên
100. for (int i = HC595\_COUNT - 1; i >= 0; i--) {
101. for (int j = 7;j >= 0;j -= 2) {
102. ledStatus[i] &= ~(1 << j);
103. shiftOutHC595(dataPin,clockPin,ledStatus);
104. delay(200);
105. }
106. }
107. for (int i = HC595\_COUNT - 1; i >= 0; i--) {
108. for (int j = 6;j >= 0;j -= 2) {
109. ledStatus[i] &= ~(1 << j);
110. shiftOutHC595(dataPin,clockPin,ledStatus);
111. delay(200);
112. }
113. }
115. /// Hãy khám phá thế giới lập trình này nhé :)
116. }