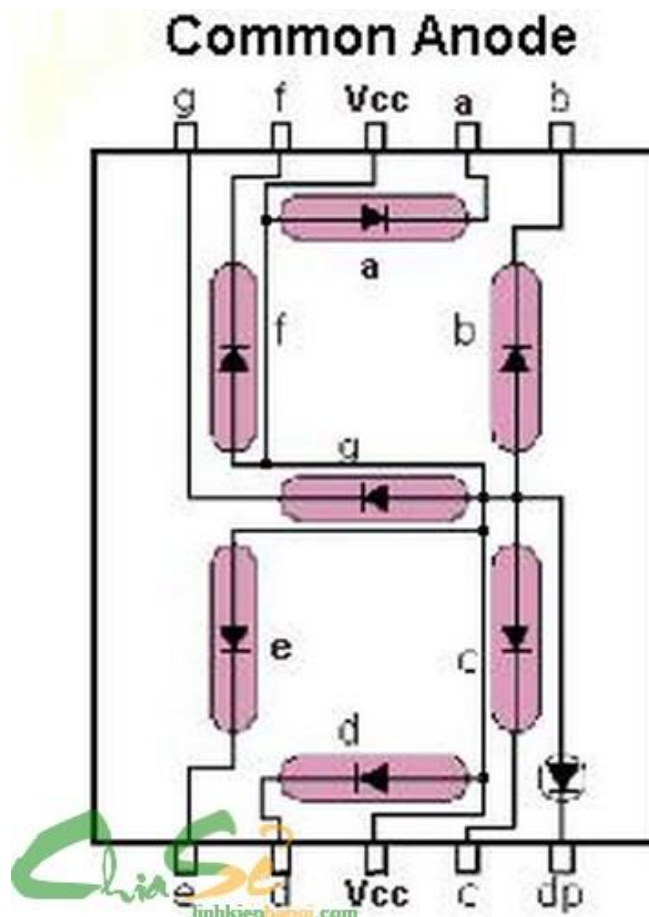


[BÀI 5]:

Lập trình LED 7 thanh với 8051

Bài trước chúng học lập trình giao tiếp để lập trình giao tiếp điều khiển Output theo Pin và theo Port. Ở bài này chúng ta sẽ áp dụng những kiến thức đó để thực hiện lập trình điều khiển hiển thị trên LED 7 thanh theo để hiển thị theo ý mình.

Sau đây là hình ảnh và chi tiết về 1 con LED 7 thanh và bảng dữ liệu mã hóa với các số từ 0-9 theo kiểu LED Anot chung.



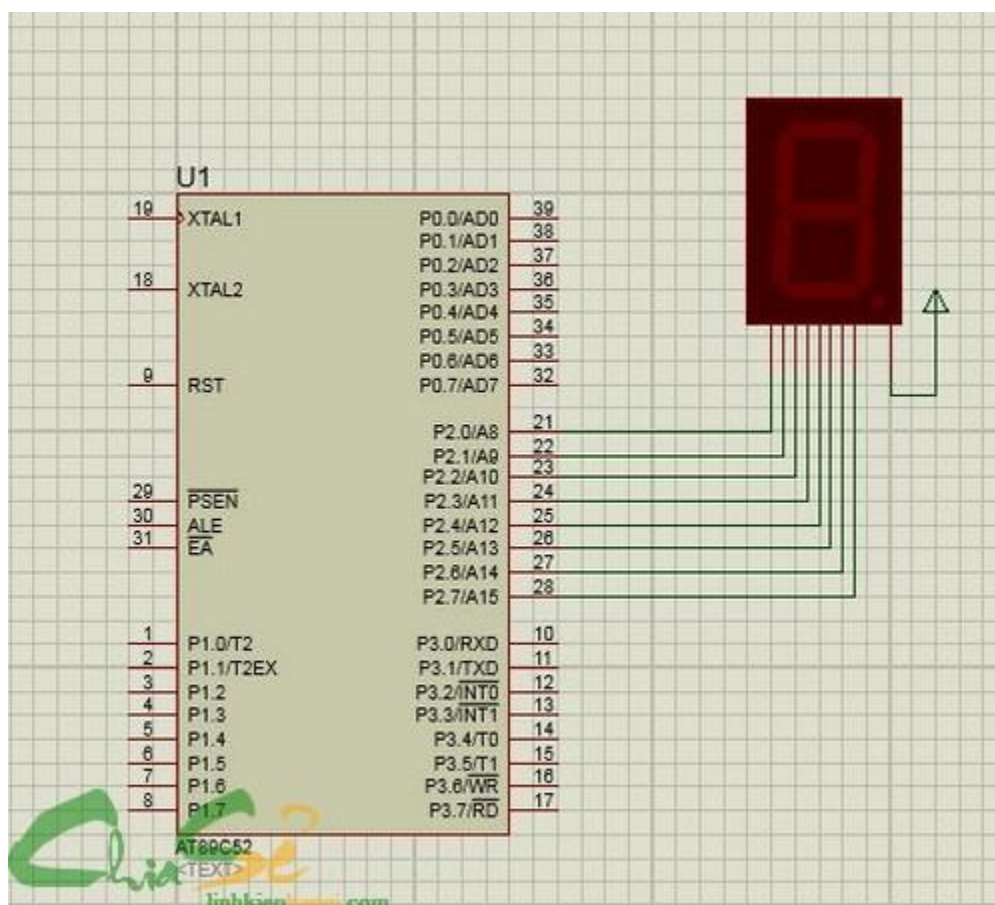
| Chữ số | A | B | C | D | E | F | G | Dp | Mã |
|--------|---|---|---|---|---|---|---|----|------|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0xC0 |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0xF9 |
| 2 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0xA4 |
| 3 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0xB0 |
| 4 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0x99 |
| 5 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0x92 |
| 6 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0x82 |
| 7 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0xF8 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0x80 |
| 9 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0x90 |

Các bạn ghi nhớ bảng trên để có thể áp dụng trong các bài toán liên quan đến LED 7 thanh mà không cần lặp lại việc này.

Chúng ta sẽ lần lượt tìm hiểu theo 2 bước sau:

Bước 1: Điều khiển 1 LED 7 thanh.

Để thực hiện bài tập này các bạn thực hiện vẽ như mạch:



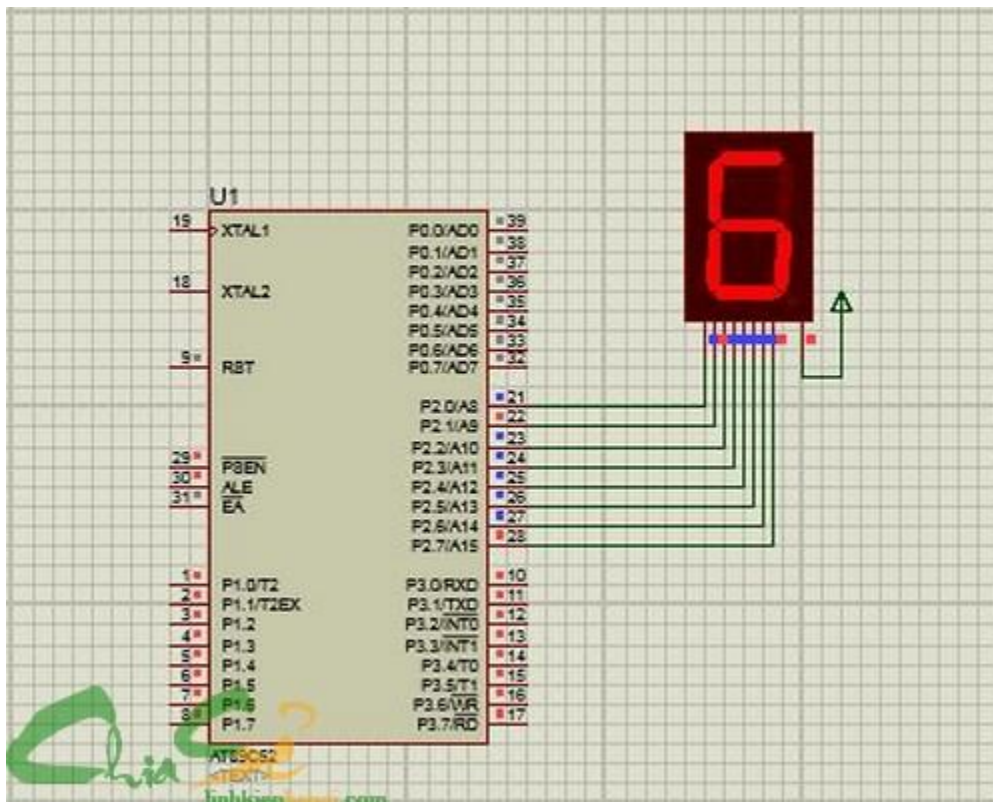
Sau đây mình giới thiệu chương trình điều khiển hiển thị lần lượt các số từ 0 đến 9 trên LED 7 thanh. Chương trình có nội dung như sau:

```
1 #include <REGX51.H>
2
3 /* Khai báo mảng 10 phần tử */
4 unsigned char LED7SEG[10] =
5 {0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90};
6
7 /* Trien khai hàm tạo độ trễ */
8 void Fn_Delay (unsigned int _vrui_Time)
9 {
10     while(_vrui_Time--);
11 }
12
13 /* Chương trình chính */
14 void main (void)
15 {
16     unsigned char i;
17     while(1)
18     {
19         /* Biến i chạy từ 0 - 9 */
20         for(i = 0; i < 10; i++)
21         {
22             P2 = LED7SEG[i]; // Xuất dữ liệu ra port P2
23             Fn_Delay(50000);
24         }
25     }
```

Ở đây chúng ta thực hiện lưu mảng dữ liệu LED 7 thanh mình đã nhắc ở bảng trên vào một mảng (*Bạn nào chưa rõ về vấn đề mảng có thể tìm hiểu thêm trong ngôn ngữ lập trình C*).

Khi đó phần tử LED7SEG[x] chứa dữ liệu hiển thị của số x trên LED thanh ở trên mạch với điều kiện x nằm từ 0-9.

Ở vòng lặp for bắt đầu từ dòng số 19 (trên code) chúng ta thực hiện đẩy dữ liệu theo của P2 lần lượt theo các mã của các số từ 0 đến 9 ra. Chạy mô phỏng chúng ta sẽ thấy các số này lần lượt hiển thị trên LED 7 thanh.



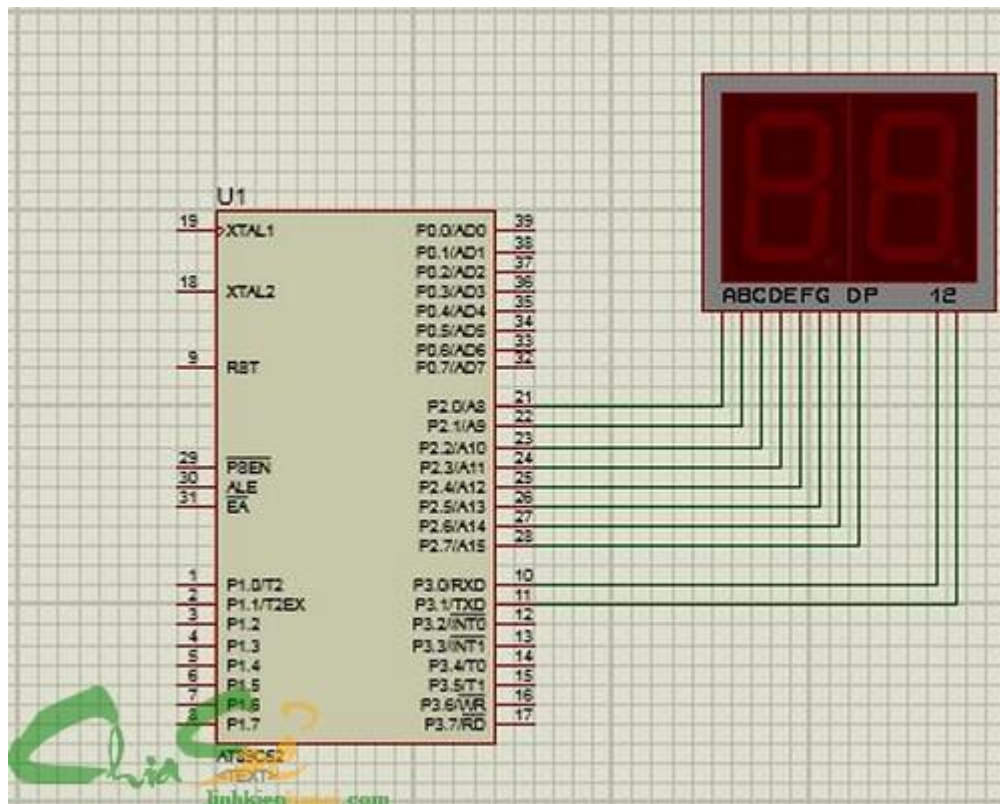
Bước 2: Điều khiển nhiều LED 7 thanh.

Ở bài toán điều khiển LED 7 thanh chúng ta có thể áp dụng theo bài toán bước 1 nối mỗi con LED 7 thanh vào 8 chân data độc lập. Tuy nhiên việc này sẽ gây lãng phí số chân điều khiển LED. Với số LED tăng lên đủ lớn số chân cần cũng tăng lên rất nhiều. Để giải quyết bài này có một kỹ thuật nêu ra là kỹ thuật “*Quét LED*”.

Kỹ thuật Quét LED thực hiện theo nguyên tắc một thời điểm chỉ bật một LED 7 thanh với dữ liệu nó cần hiển thị, các LED còn lại được tắt. Việc quét LED thực hiện luân phiên sáng các LED với yêu cầu trên. Có một hiện tượng hay xảy ra với những người mới thực hiện lập trình quét LED là hiện tượng “bóng ma” đó là hiện tượng xuất hiện các bóng mờ LED không mong muốn do quá trình điều khiển. Quá trình quét LED chuẩn được thực hiện theo các bước sau:

1. Xuất ra mã hiển thị.
2. Cấp nguồn cho LED muốn hiển thị.
3. Trễ 1 khoảng thời gian để duy trì sáng.
4. Cắt nguồn LED vừa hiển thị.

Các bạn chú ý thực hiện việc quét LED trong code của mình theo quy trình để tránh gặp những rắc rối không mong muốn. Sau đây mình xin giới thiệu một đoạn chương trình hiển thị các số từ 00-99 trên cơ sở mạch điện như sau:



Các bạn thực hiện chương trình code như sau:

```

1 #include <REGX51.H>
2
3 sbit LED1 = P3^0;
4 sbit LED2 = P3^1;
5 unsigned char LED7SEG[10] =
6 {0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x90};
7
8 void Fn_Delay (unsigned int _vrui_Time)
9 {
10     while(_vrui_Time--);
11 }
12
13 void main (void)
14 {
15     unsigned char i;
16     unsigned int j;
17     LED1 = 0;
18     LED2 = 0;
19     while(1)
20     {

```

```

21      /* Hien thi so i chay tu 00 den 99 */
22      for(i = 0; i < 100; i++)
23      {
24          /* Voi moi so i hien thi 200 lan */
25          for (j = 0; j < 200; j++)
26          {
27              /* Hien thi phan chuc cua so */
28              P2 = LED7SEG[i/10];
29              LED1 = 1;
30              Fn_Delay(100);
31              LED1 = 0;
32              /* Ket thuc hien thi phan chuc */
33
34              /* Hien thi phan don vi cua so */
35              P2 = LED7SEG[i%10];
36              LED2 = 1;
37              Fn_Delay(100);
38              LED2 = 0;
39              /* Ket thuc hien thi phan don vi */
40          }
41      }
42  }

```

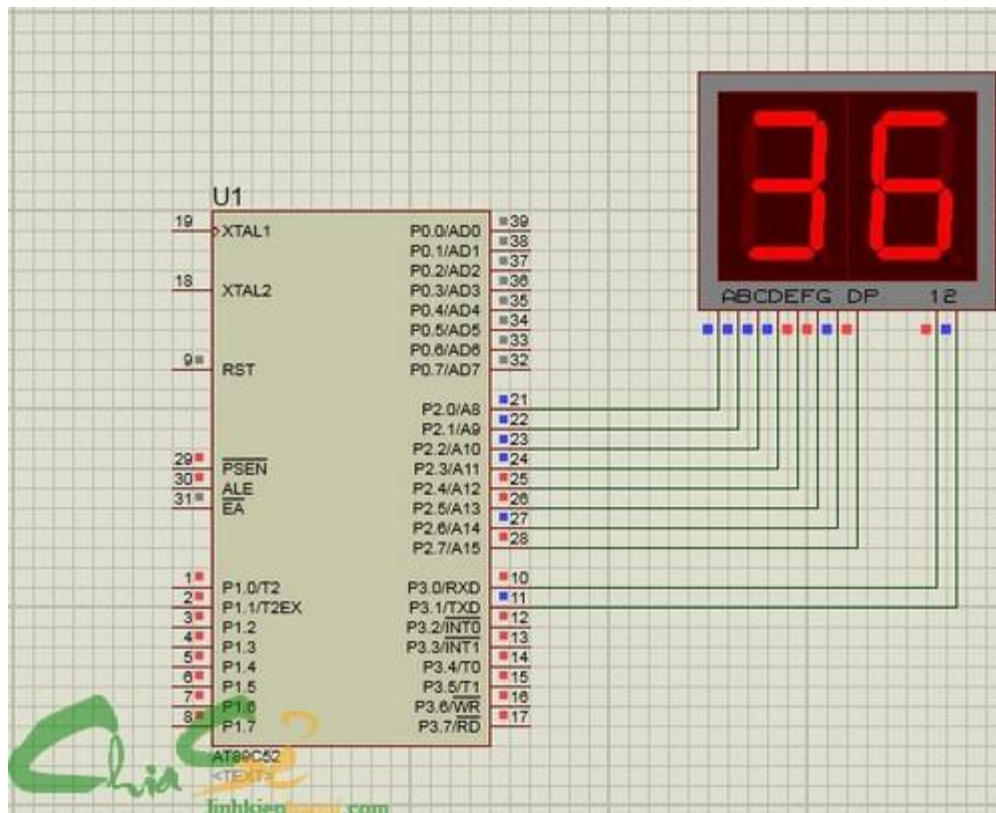
Trên đây chúng ta đã thực hiện theo quy trình 4 bước quét LED đã được nhắc ở trên cụ thể như sau:

```

1 P2 = LED7SEG[i/10]; // Xuất dữ liệu LED
2 LED1 = 1;           // Bật LED hiển thị
3 Fn_Delay(100);      // Trễ một khoảng thời gian
4 LED1 = 0;           // Tắt LED đang hiển thị

```

Ở đây ngoài vòng for thực hiện đếm từ 00-99 chúng ta cần thêm một vòng for cho biến j chạy từ 0-200 để đảm bảo mỗi số được hiển thị trong một khoảng thời gian đủ lâu để chúng ta có thể nhìn thấy. Kết quả của chương trình như sau:



Như vậy là chúng ta đã nắm được quy trình hiển thị nhiều LED 7 thanh theo phương pháp quét LED. Từ những kiến thức cơ bản này các bạn có thể áp dụng để hiển thị cho nhiều LED hơn cũng như hiển thị thông tin theo ý muốn của các bạn. Sau này khi tìm hiểu về Timer các bạn có thể kết hợp với Timer để thực hiện quét LED tránh sử dụng các hàm delay gây lãng phí thời gian xử lý của vi điều khiển.

3. Bài tập:

Bài 1: Thực hiện hiển thị 4 LED 7 thanh hiển thị lần lượt các số từ 0000-9999.
Chúc các bạn thành công !