

Đại học Sư phạm Kỹ thuật Thành phố Hồ Chí Minh

Khoa Cơ khí Chế tạo Máy

Bộ môn Cơ Điện tử

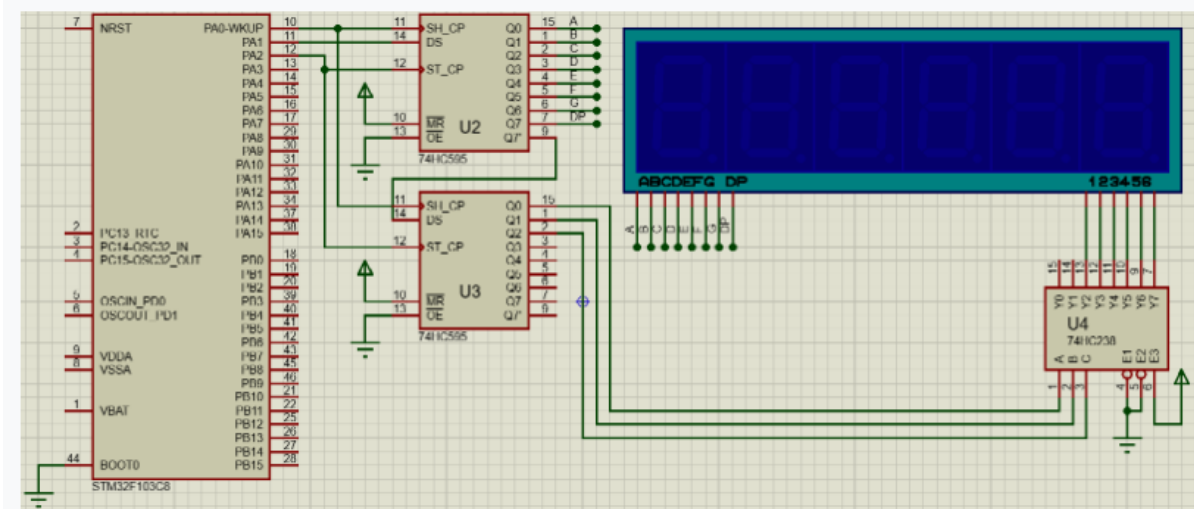
# BÁO CÁO DỰ ÁN

## 1. Thông tin người thực hiện:

STT	Người thực hiện	MSSV	Ngày
01	Nguyễn Hữu Chí	20146479	9/5/2023

## 2. Yêu cầu dự án

- Cho sơ đồ mạch điện như bên dưới (bản thiết kế bằng phần mềm Proteus được đính kèm).  
Yêu cầu viết chương trình điều khiển LED 7 đoạn sáng số kiểu đồng hồ đếm lên (ví dụ hiển thị 07-45-03 là <07 giờ>-<45 phút>-<03 giây>)

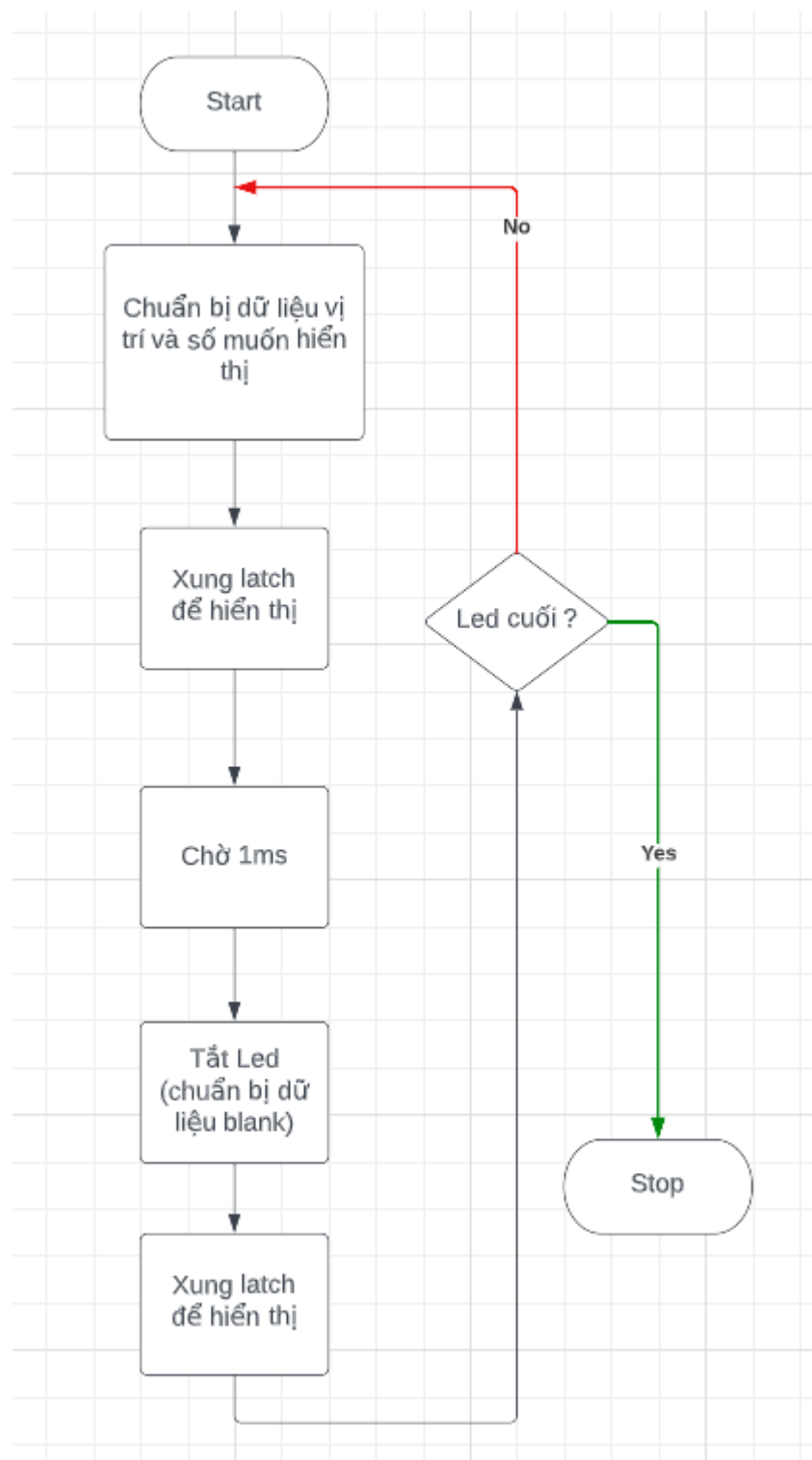


## 3. Nội dung:

### 3.1. Phân tích dự án:

- Ý tưởng bài này là đầu tiên ta giải quyết phần hiển thị
- Đầu tiên chuẩn bị 3 xung kích để chọn vị trí Led muốn sáng, tiếp theo là 8 xung để hiển thị số muốn hiển thị. Sau đó cho 1 xung latch để hiển thị lên, chờ 1 ms thì tắt led để chống lem. Lặp lại các bước trên 6 lần là ta hiển thị được thời gian hh.mm.ss
- Ta sẽ tạo 1 biến count đếm lên để có thể thay đổi thời gian hiển thị. Biến count này sẽ được đổi thành 6 giá trị tương ứng để có thể thay đổi và hiển thị lên 6 Led ⇔ hh.mm.ss

### 3.2. Lưu đồ lập trình:



### 3.3. Mã nguồn chương trình:

- Khai báo bộ mã số và các biến cần thiết

```
12
13 /* USER CODE BEGIN PV */
14 const uint8_t LED[] = {0x03,0x9F,0x25,0x0D,0x99,0x49,0x41,0x1F,0x01,0x09};
15 uint32_t count = 31200 ; // set h ban dau
16 uint8_t sec_donvi;
17 uint8_t sec_chuc;
18 uint8_t min_donvi;
19 uint8_t min_chuc;
20 uint8_t hour_donvi;
21 uint8_t hour_chuc;
22 /* USER CODE END PV */
23
```

- Hàm tạo xung clock

```
62 /* USER CODE BEGIN 0 */
63 void CLK(uint32_t n){
64     while(n>0){
65         HAL_GPIO_WritePin(CLK_GPIO_Port,CLK_Pin,1);
66         HAL_GPIO_WritePin(CLK_GPIO_Port,CLK_Pin,0);
67         n = n-1;
68     }
69 }
70
```

- Hàm này giúp hiển thị số nào tại vị trí nào (có cả xử lý dot để ngăn cách giờ phút giây)

```
71 void set_LED(uint8_t pos, uint8_t NUM, uint8_t dot){
72     HAL_GPIO_WritePin(DAT_GPIO_Port,DAT_Pin,(pos+1)>>2&lu);
73     CLK(1);
74     HAL_GPIO_WritePin(DAT_GPIO_Port,DAT_Pin,(pos+1)>>1&lu);
75     CLK(1);
76     HAL_GPIO_WritePin(DAT_GPIO_Port,DAT_Pin,(pos+1)&lu);
77     CLK(1);
78
79     for(int8_t k = 0; k <8;k++){
80         HAL_GPIO_WritePin(DAT_GPIO_Port,DAT_Pin,((LED[NUM]&(254 + dot))>>k&lu));
81         // 254 biến bit cuối thành 0 -> dấu chấm. 255 thì không có dấu chấm
82         CLK(1);
83     }
84
85     HAL_GPIO_WritePin(LAT_GPIO_Port,LAT_Pin,1);
86     HAL_GPIO_WritePin(LAT_GPIO_Port,LAT_Pin,0);
87
88     HAL_Delay(1);
89     // Chong lem
90     for(int8_t k = 0; k <8;k++){
91         HAL_GPIO_WritePin(DAT_GPIO_Port,DAT_Pin,1); // off all. chong lem
92         CLK(1);
93     }
94     HAL_GPIO_WritePin(LAT_GPIO_Port,LAT_Pin,1);
95     HAL_GPIO_WritePin(LAT_GPIO_Port,LAT_Pin,0);
96
97 }
```

- Chúng ta chuẩn bị dữ liệu blank để tắt, không quan tâm vị trí nào được tắt (do dữ liệu nối tiếp trước đó) thì cũng đáp ứng được việc tắt như mong muốn (khi latch lên thì chỉ có 1

trong 6 led được kích còn lại thì tắt, mà led kích cũng hiển thị blank đã chuẩn bị trước đó nên toàn bộ 6 led xem như tắt hết)

- Hiển thị hh.mm.ss bằng cách gọi hàm set\_LED 6 lần với các thông số tương ứng
- Vòng for tạo độ trễ giữa mỗi lần tăng:  $\text{delay} = 30 \times 6 \times 1\text{ms} = 180\text{ms}$  (không chính xác)
- Sau mỗi lần hiển thị biến count được tăng lên và quy đổi ra 6 giá trị để cập nhật vào phần hiển thị

```
136  /* USER CODE BEGIN WHILE */
137  while (1)
138  {
139      /* USER CODE END WHILE */
140      if (count > 86399) count = 0;
141      hour_chuc = (count/3600)/10;
142      hour_donvi = (count/3600)%10;
143      min_chuc = ((count/60)/10)%6; // <6
144      min_donvi = (count/60)%10;
145      sec_chuc = (count/10)%6;
146      sec_donvi = count%10;
147
148      for (uint8_t f = 0;f<30;f++) // tao do tre = delay
149      {
150          set_LED(1, hour_chuc, 1);
151          set_LED(2, hour_donvi, 0);
152          set_LED(3, min_chuc, 1);
153          set_LED(4, min_donvi, 0);
154          set_LED(5, sec_chuc, 1);
155          set_LED(6, sec_donvi, 1);
156      }
157
158      count = count+1;
159      .....
```

#### 4. Ghi chú khác (nếu có)