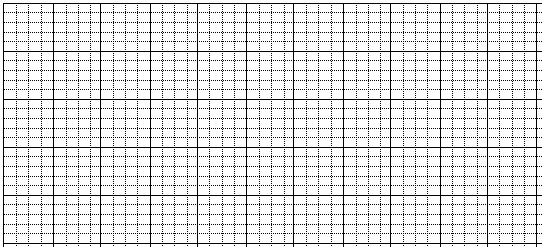
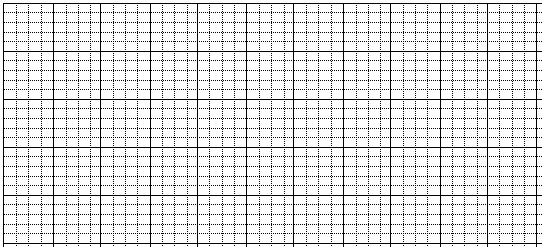
|  |  |
| --- | --- |
| **TRƯỜNG ĐH SƯ PHẠM KỸ THUẬT TPHCM**  **KHOA CƠ KHÍ CHẾ TẠO MÁY**  **BỘ MÔN CƠ ĐIỆN TỬ**  **Họ và tên: Võ Hồng Quân**  **Mã số SV: 22134012** | **BÁO CÁO**  **THỰC TẬP LẬP TRÌNH VI XỬ LÝ**  **Ngày:**  **Tuần học: 03**  **Nội dung: Giao tiếp thanh ghi dịch để điều khiển LED 7 đoạn** |

1. **Khảo sát sơ đồ nguyên lý:**

**Sinh viên vẽ sơ đồ nguyên lý khối LED 7 đoạn và giải thích**

****

**Sinh viên vẽ sơ đồ nguyên lý khối thanh ghi dịch và giải thích**

****

1. **Bảng kết nối input - output**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên phần cứng** | **Vị trí chân vi điều khiển tương ứng** | **Mô tả chức năng** |
| 1 | MOSI | PB5 | Chân dữ liệu đầu ra từ vi điều khiển |
| 2 | SCK | PB3 | Chân xung clock cho giao tiếp |
| 3 | RCL | PE2 | Chốt dữ liệu vào thanh ghi dịch |
| 4 | RCK | PE0 | Xung clock để đẩy dữ liệu ra các chân đầu ra |

1. **Lập trình**
   1. **Tính bảng mã LED 7 đoạn:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Hiển thị** | **QA**  **F** | **QB**  **A** | **QC**  **E** | **QD**  **D** | **QE**  **DP** | **QF**  **B** | **QG**  **G** | **QH**  **C** | **Mã nhị phân** | **Mã HEX** |
| **0** | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 11111101 | 0xFD |
| **1** | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 00001001 | 0x9 |
| **2** | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 10110111 | 0xB7 |
| **3** | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 10101111 | 0xAF |
| **4** | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 01001111 | 0x4F |
| **5** | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 11001011 | 0xCB |
| **6** | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 11011011 | 0xDB |
| **7** | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 10100001 | 0xA1 |
| **8** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 11111111 | 0xFF |
| **9** | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 11001111 | 0xCF |
| **(Tắt LED)** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 00000000 | 0x0 |
| **A** | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 11100111 | 0xE7 |
| **b** | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 11001101 | 0xCD |
| **C** | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 00001111 | 0xF |
| **d** | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 11101100 | 0xEC |
| **E** | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 01001111 | 0x4F |
| **F** | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 01000111 | 0x47 |
| **H** | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 11100101 | 0xE5 |
| **i** | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 00000101 | 0x5 |
| **J** | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 10101000 | 0xA8 |
| **L** | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 00001101 | 0xD |
| **N** | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 10100111 | 0xA7 |
| **O** | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 10101111 | 0xAF |
| **P** | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 01100111 | 0x67 |
| **S** | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 11001011 | 0xCB |
| **U** | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 10101101 | 0xAD |

* 1. **Tạo thư viện myLED:**
* Viết thư viện *myLED.h* và *myLED.c* với các hàm và macro sau đây:
* Hàm Data8\_put(uint8\_t d8) để đẩy số kiểu 1 byte d8 ra thanh ghi dịch. Chân QH là MSB của d8; chân QA là LSB của d8.
* Hàm Data16\_put(uint16\_t d16) để đẩy số kiểu 2 byte d16 ra 2 thanh ghi dịch nối tiếp. Chân QH của thanh ghi dịch cuối là MSB của d16; chân QA là LSB của thanh ghi dịch đầu là LSB của d16.
* LED\_data(3) : Hien thi so 3
* Hàm LED\_data(uint8\_t n) để giải mã hiển thị số n lên LED 7 đoạn.
* Hàm LED\_put(uint8\_t n, uint8\_t p) để giải mã hiển thị số n lên LED 7 đoạn tại vị trí p. với p như sau:
  + p = 0: Tắt hết các LED
  + p = 1: Hiển thị LED vị trí k1 (00000001)
  + p = 2: Hiển thị LED vị trí k2 (00000010)
  + p = 4: Hiển thị LED vị trí k3 (00000100)
  + p = 8: Hiển thị LED vị trí k4 (00001000)
  + p = 16: Hiển thị LED vị trí k5 (00010000)
  + p = 3: Hiển thị LED vị trí k1 và k2. (00000011)
  + p = 7: Hiển thị LED vị trí k1 và k2 và k3. (00000110)
* Hàm LED\_putstring(uint8\_t \* s) để đẩy một chuỗi lên LED 7 đoạn:
  + Chỉ hiển thị tối đa 5 ký tự đầu tiên của chuỗi s
  + Nếu ký tự nằm ngoài bảng mã thì tắt hiển thị.
* myLED.h

|  |
| --- |
|  |

* myLED.c

|  |
| --- |
|  |

* 1. **Lập trình ứng dụng 1**
* Khi mới mở điện đèn LED 7 đoạn hiển thị 5 số cuối của mã số sinh viên.
  1. **Lập trình ứng dụng 2**

Viết chương trình cấp dữ liệu cho IC74HC595 để giải mã hiển thị LED 7 đoạn

-         Mới mở điện lên LED hiển thị số 12345.

-         Nhấn nút P1 số hiển thị tăng lên 1 đơn vị (Ví dụ từ 12345 tăng lên 12346)

-         Nhấn nút P2 số hiển thị giảm đi 1 đơn vị (Ví dụ từ 12345 giảm thành 12344)

-         Nhấn nút P3 số hiển thị trở về 12345

-         Nhấn nút P4 số hiển thị tự đếm lên, mỗi giây tăng 1 đơn vị. Tăng đến 99999 thì dừng.

* 1. **Lập trình ứng dụng 3**

Lập trình hệ thống hiển thị LED 7 đoạn đồng hồ như sau:

* Khi mới mở điện, LED 7 đoạn hiển thị dạng phút và giây MM-SS . Ví dụ: 12-00 (12 phút 00 giây)
* Mỗi khi nhấn P1 thì bật/tắt chế độ đếm lên tự động (mỗi giây tăng 1 đơn vị giây hiển thị. (Lưu ý khi giây tăng 59 thì chuyển phút tăng 1 đơn vị).
* Mỗi khi nhấn P2 thì bật/tắt chế độ đếm xuống tự động (mỗi giây giảm 1 đơn vị giây hiển thị).
* Nút nhấn P3 và P4 sử dụng để điều chỉnh giá trị hiển thị phút và giây như sau:  
  + Mỗi lần nhấn P3 thì hiển thị LED 7 đoạn chuyển lần lượt qua 3 chế độ:
    - Chế độ chỉnh giây (2 LED giây nhấp nháy 0.5s liên tục);
    - Chế độ chỉnh phút (2 LED phút nhấp nháy 0.5s liên tục);
    - Chế độ nghỉ (LED hiển thị bình thường, không LED nào nhấp nháy hay tắt).
  + Mỗi lần nhấn P4:
    - Nếu đang ở chế độ chỉnh giây thì P4 làm tăng giá trị hiển thị giây 1 đơn vị;
    - Nếu đang ở chế độ chỉnh phút thì P4 làm tăng giá trị hiển thị phút 1 đơn vị;
    - Nếu đang ở chế độ nghỉ thì P4 không có tác dụng gì.
    - Lưu ý khi chỉnh giây hoặc phút tăng đến 59 thì lặp lại 00. Tăng giây từ 59 sang 00 không làm tăng giá trị của phút.