

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**  
**KHOA KỸ THUẬT MÁY TÍNH**



**BÁO CÁO THỰC HÀNH LAB 03**  
**MÔN THỰC HÀNH VI XỬ LÝ – VI ĐIỀU KHIỂN**

**Họ và tên sinh viên : Nguyễn Hữu Tứ**

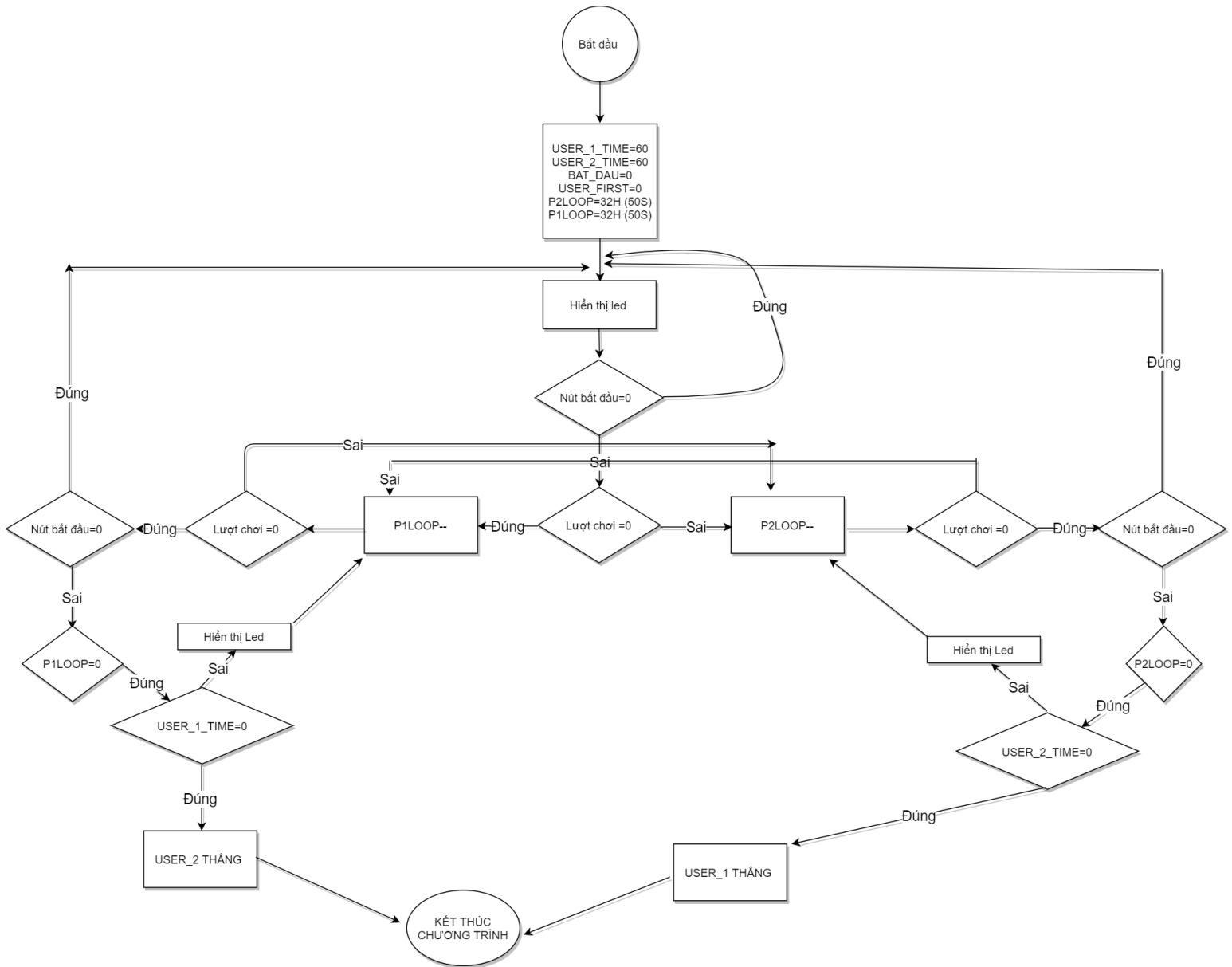
**Mã số sinh viên: 19522453**

**GIẢNG VIÊN HƯỚNG DẪN**

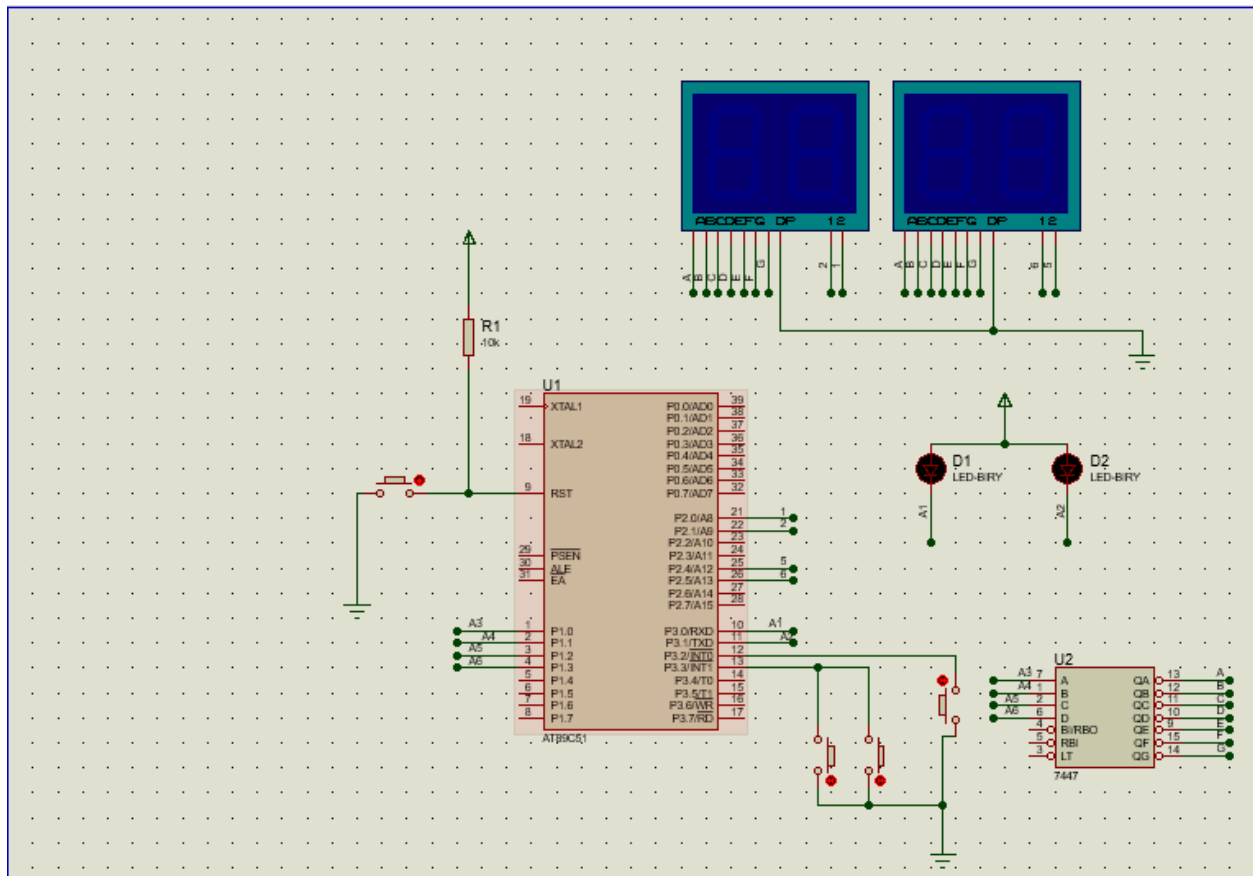
**Bùi Phùng Hữu Đức**

***TP. HỒ CHÍ MINH, 5/2021***

# 1. Sơ đồ giải thuật :



Hình 1 :Sơ đồ giải thuật



Hình 2 : Schematic mô phỏng

Link video:

<https://drive.google.com/file/d/1JlirOiQYOitb16l7PqP2vYRyoN4v3fTp/view?usp=sharing>

## 2. Chức năng các phần cứng được sử dụng:

- 2 đèn led 7 đoạn thể hiện số giờ của 2 người chơi
- 2 đèn led đơn để báo hiệu người chiến thắng
- PORT1 được nối với 7447 và vi xử lý để điều khiển chữ số hiển thị cho led 7 đoạn
- PORT2 được nối với 7447 và vi xử lý để điều khiển nguồn cho led 7 đoạn
- 2 chân P3.0 và P3.1 dùng để điều khiển 2 led đơn báo hiệu người chơi nào thắng cuộc.

- Chân P3.2(INT0) dùng để điều khiển tín hiệu Start/Stop.
- Chân P3.3(INT1) dùng để điều khiển tín hiệu lượt chơi của người chơi.

### 3. Giải thích chức năng của từng module:

Code	Giải thích nguyên lý
ORG 0000H	; Bảng vector ngắt
LJMP MAIN	
ORG 0013H	;ngat ngoai 1
LJMP NGAT_NGOAI_1	
ORG 0003H	;ngat ngoai 0
LJMP NGAT_NGOAI_0	
ORG 0030H	
MAIN:	
MOV TCON,#05H	; Set IT0 và IT1 ở thanh ghi TCON
MOV IE,#85H	; Cho phép ngắt toàn cục và INT0, INT1
	;Thời gian của người chơi 1 là 60s
MOV R1, #60 ;SECOND	;Thời gian của người chơi 2 là 60s
MOV R2, #60 ;SECOND	
	;LUOT CHOI (0: NGUOI CHOI 1, 1:
CLR 00H	NGUOI CHOI 2)
	;Cờ để xác định định tín hiệu ở nút bấm
;Flag	Start/ Stop ( 0: Start 1: Stop )
CLR 01H	
	; Vòng lặp của người chơi 1 32H (50s)
MOV R4, #032H	; Vòng lặp của người chơi 2 32H (50s)
MOV R5, #032H	

<p>BAT_DAU:</p> <p>JNB 01H, DUNG_STOP</p> <p>;THOI GIAN CUA MOI NGUOI CHOI</p> <p>JNB 00H, LUOT_NGUOI_CHOI_1</p> <p>JMP LUOT_NGUOI_CHOI_2</p> <p>;DUNG_STOP DEM XUONG</p> <p>DUNG_STOP:</p> <p>CALL HIEN_THI_THOI_GIAN</p> <p>JMP BAT_DAU</p> <p>LUOT_NGUOI_CHOI_1:</p> <p>LOOP1:</p> <p>CALL THOI_GIAN_NGUOI_CHOI</p> <p>JB 00H, LUOT_NGUOI_CHOI_2</p> <p>JNB 01H, DUNG_STOP</p> <p>DJNZ R4, LOOP1</p> <p>DEC R1</p> <p>MOV R4, #32H</p> <p>CJNE R1, #0, BAT_DAU</p> <p>;nếu R1 = 0 → người chơi 2 thắng</p> <p>CLR P3.1</p> <p>CLR 01H</p> <p>JMP DUNG_STOP</p>	<p>;BAT_DAU DEM XUONG</p> <p>;Nếu tín hiệu ở 01H = 1 thì bắt đầu chạy đèn led của người chơi 1</p> <p>; Nếu tín hiệu ở 01H = 0 thì tới lượt của người chơi 1</p> <p>; Nếu 01H=1 thì tới lượt của người chơi 2</p> <p>;Nếu tín hiệu ở 01H = 0 thì dừng cả 2 đèn</p> <p>; tiếp tục đợi tín hiệu</p> <p>; tới lượt lần người 1</p> <p>;Hiện thị thời gian đếm của người chơi</p> <p>;Khi mà nhấn nút thì nhảy tới hàm của người chơi 2</p> <p>;Nếu 01H=0 thì dừng màn hình</p> <p>; So sánh vòng lặp nếu khác 0 thì tiếp tục người chơi 1</p> <p>;Giảm đi giây của người chơi 1</p> <p>;Hết vòng lặp nạp lại giá trị của R4</p> <p>;R1 != 0 thì tiếp tục, nếu hết thời gian</p> <p>;=&gt; người chơi 2 thắng .</p> <p>;Bật Led ở chân P3.1 để báo hiệu</p> <p>;Bắt đầu lại</p> <p>; Dừng lại màn hình kết thúc.</p>
--	---

<pre> LUOT_NGUOI_CHOI_2: LOOP2: CALL THOI_GIAN_NGUOI_CHOI JNB 00H, LUOT_NGUOI_CHOI_1 JNB 01H, DUNG_STOP DJNZ R5, LOOP2 DEC R2 MOV R5, #32H CJNE R2, #0, BAT_DAU  CLR P3.0 CLR 01H JMP DUNG_STOP THOI_GIAN_NGUOI_CHOI: CALL HIEN_THI_THOI_GIAN CALL DELAY_1000us CALL DELAY_1000us CALL DELAY_1000us CALL DELAY_1000us CALL DELAY_500us RET  ; 28h ; 25h HIEN_THI_THOI_GIAN: MOV 28h, #01D MOV 25h, R1 </pre>	<pre> ;Nếu R2 = 0 → người chơi 1 thắng  ; dùng làm thời gian hiển thị số ; gọi hàm hiển thị thời gian sẽ có 15000us ;4 hàm delay 1000us =&gt; 4000us ;1 hàm delay 500us =&gt; 500us ;=&gt;195000us ;=&gt; 195000*50=975000us ~1s( Tính xấp xỉ)  ; Sử dụng thanh ghi ở Ram có địa chỉ 28h, 25h để hiển thị led 7 đoạn. </pre>
--	--

CALL HIEN_THI	
CALL DICH_LED	;Hàm hiển thị thời gian
CALL HIEN_THI	; Bật 2 led 7 đoạn
CALL DICH_LED	; Giây của người chơi 1
MOV 25h, R2	; Hiện thị số
CALL HIEN_THI	; Dịch led để hiển thị số tiếp theo
RET	
HIEN_THI:	;Giây của người chơi 2
CALL TACH_BCD	;Hiện thị số
CALL LEDON	
CALL DELAY_1000us	
;HIGH BYTE	; Biểu thị số BCD, ta tách 4 bit để nạp vào
CALL LAY_BIT_CAO	hàng đơn vị led 7 đoạn
CALL DICH_LED	; Dịch led 7 đoạn để biểu thị số tiếp theo
CALL LEDON	; Lấy 4 bit cao để biểu thị hàng chục của
CALL DELAY_1000us	Led 7 đoạn
RET	;Dịch led 7 đoạn để biểu thị số tiếp theo.
TACH_BCD:	
MOV A, 25h ; gán giá trị vào A	
MOV B, #10 ; Gán b =10	
DIV AB ; thương lưu vào A, dư lưu vào B	; Hàm tách hàng chục và hàng đơn vị , để
	biểu diễn qua số BCD
RL A	; Với A là số thương
RL A	;B là số dư sau khi thực hiện phép chia
RL A	





```

DELAY_1000us:
;DELAY_1000us 1000us
MOV TMOD, #01
MOV TH0,#HIGH(-1000)
MOV TL0,#LOW(-1000)
SETB TR0
HERE0:
JNB TF0, $
CLR TF0
CLR TR0
RET

```

; Hàm tạo delay 1000us

```

DELAY_500us:
;DELAY_1000us 500US
MOV TMOD, #01
MOV TH0,#HIGH(-500)
MOV TL0,#LOW(-500)
SETB TR0
HERE1:
JNB TF0, $
CLR TF0
CLR TR0
RET

```

; Hàm tạo delay 500us

```

NGAT_NGOAI_0:
;BAT_DAU/DUNG_STOP
CPL 01H
RETI

```

NGAT_NGOAI_1: ;Chuyen luot choi cho user CPL 00H RETI  END	; Hàm ngắt ngoài để chuyển lượt chơi
---	--------------------------------------