|  |
| --- |
| **PRACTICE EXERCISES OF THE MICROPROCESSORS & MICROCONTROLLERS**  **Instructor: The Tung Than**  **Student's name: Nguyen Quoc Truong An**  **Student code: 21521810** |

**PRACTICE REPORT NO 3**

**LAB3: USING INTERRUPT**

1. **Student preparation**

− Knowledge of how to install and use Interrupt.

1. **Practice content**

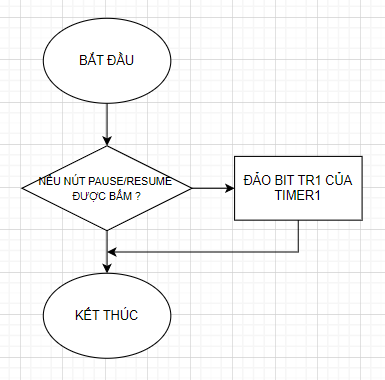
|  |
| --- |
| 1. Present and draw a flowchart to handle 2 buttons with the following functions: |

o Button A: Pause/Resume stopwatch

o Button B: Reset the stopwatch.

* Với nút A, ta nối với ngắt ngoài 0 (INT0) sử dụng ngắt ở chế độ ngắt cạnh. Ta thực hiện tính toán thời gian (giây, phần trăm giây) trong hàm ngắt của Timer1 khi giá trị Timer1 tràn và thực hiện quét led để hiển thị sử dụng hàm delay bằng vòng lặp.

Khi nút A được bấm, tức ngắt ngoài 0 (INT0) được hoạt động. Lúc này hàm xử lý ngắt ISR\_INT0 sẽ được tự động gọi. Trong hàm này ta thực hiện đảo bit TR1 để bật tắt Timer1 tương ứng với khi Resume và Pause.

* Với nút B, ta nối với ngắt ngoài 1 (INT1) sử dụng ngắt ở chế độ ngắt cạnh. Khi nút B được bấm, hàm xử lý ngắt ISR\_INT1 được tự động gọi. Trong hàm này ta thực hiện nạp lại giá trị của 2 thanh ghi lưu giây và lưu phần trăm giây R1 = 0, R0 = 0 ứng với chức năng reset.
* Lưu đồ thuật toán xử lý khi bấm nút A:
* Diagram

  Description automatically generatedLưu đồ thuật toán xử lý khi bấm nút B:

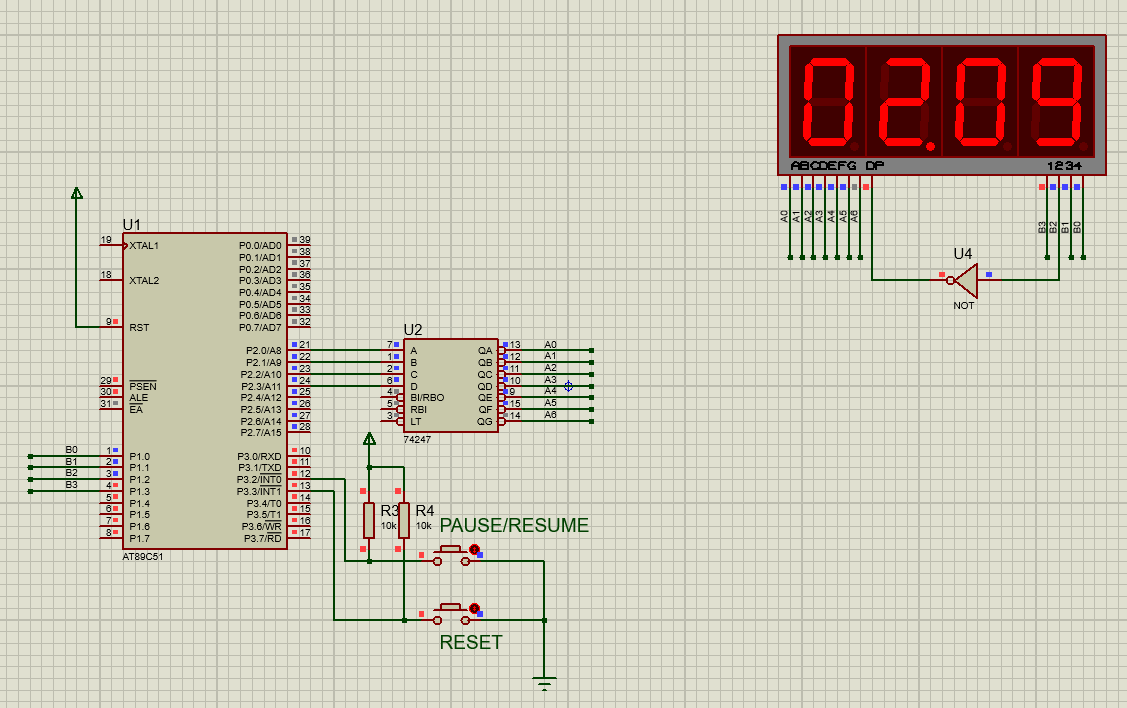
1. Using AT89C51/AT89C52 in combination with 4 7-Segment LED modules

and 2 buttons above, design a Sport clock circuit with the ability to count

accurately to 1% of seconds, counting range from 00.00 seconds to 99.99

seconds and has 2 buttons to control Pause/Resume and Reset.

-**Thiết kế:**



-**Code:**

$NOMOD51

$INCLUDE (8051.MCU)

*; RESET VECTOR*

ORG 0000H

JMP START

*; VECTOR NGAT TIMER1*

ORG 0001BH

LJMP ISR\_TIMER1

*; VECTOR NGAT INT0*

ORG 0003H

LJMP ISR\_INT0

*; VECTOR NGAT INT1*

ORG 0013H

LJMP ISR\_INT1

ORG 0100H

START:

MOV TMOD, #10H *; CHON TIME1 CHE DO 1*

MOV TL1, #0F0H *; NAP GIA TRI CHO TIMER1*

MOV TH1, #0D8H

SETB TR1 *; BAT TIMER1*

MOV IE, #10001101B *; CHO PHEP NGAT TIMER1, NGAT INT0, INT1*

SETB TCON.0 *; CHO PHEP NGAT THEO CANH INT0*

SETB TCON.2 *; CHO PHEP NGAT THEO CANH INT1*

MOV R0, #0 *; LUU 1%S*

MOV R1, #0 *; LUU GIAY*

LOOP:

CALL DISPLAY\_LOOP

JMP LOOP

DISPLAY\_LOOP: *; HAM HIEN THI QUET LED*

SETB P1.0 58 MOV A, R0

MOV B, #10

DIV AB

MOV P2, B

CALL DELAY\_5MS

CLR P1.0

SETB P1.1

MOV A, R0

MOV B, #10

DIV AB

MOV P2, A

CALL DELAY\_5MS

CLR P1.1

SETB P1.2 74 MOV A, R1

MOV B, #10

DIV AB

MOV P2, B

CALL DELAY\_5MS

CLR P1.2

SETB P1.3 82 MOV A, R1

MOV B, #10

DIV AB

MOV P2, A

CALL DELAY\_5MS

CLR P1.3

RET

DELAY\_10MS: *; HAM DELAY 10ms BANG TIMER1, DUNG DE TANG GIA TRI PHAN TRAM GIAY*

MOV TL1, #0F0H 93 MOV TH1, #0D8H

SETB TR1

JNB TF1, $

RET

DELAY\_5MS: *; HAM DELAY 5ms BANG VONG LAP, DUNG DE QUET LED*

MOV R4, #10

DELAY:

MOV R5, #250

DJNZ R5, $

DJNZ R4, DELAY

RET

ISR\_TIMER1: *; HAM XU LY BO DEM THOI GIAN KHI TIMER1 TRAN*

*; XU LY TANG DON VI THOI GIAN*

INC R0

MOV A, R0

MOV B, #100

CJNE A, B, NOT\_EQUAL

MOV R0, #0

INC R1

MOV A, R1

MOV B, #100

CJNE A, B, NOT\_EQUAL

MOV R1, #0

NOT\_EQUAL: *; NAP LAI GIA TRI CHO TIMER1*

MOV TL1, #0F0H 123 MOV TH1, #0D8H

RETI

ISR\_INT0: *; HAM XU LY NUT A*

CPL TR1 *; TAT TIMER1*

RETI

ISR\_INT1: *; HAM XU LY NUT B*

MOV R0, #0 *; GAN PHAN TRAM GIAY = 0*

MOV R1, #0 *; GAN GIAY = 0*

RETI

END

1. **Exercises**

Add 2 buttons to the Sport watch with the following function:

o Button C: Increase the number of seconds counting to 1 second

o Button D: Decrease the number of seconds to 1 second

-**Thiết kế:**

**Diagram, schematic

Description automatically generated**

**-Code: Ta chỉ cần xử lý thêm một đoạn code ở vòng lặp Loop**

LOOP:

JNB P0.0, INCREASE *; NEU NUT INCREASE ĐƯỢC BẤM THÌ NHAY TOI HAM INCREASE*

JNB P0.1, DECREASE *; NEU NUT DECREASE ĐƯỢC BẤM THÌ NHAY TOI HAM DECREASE*

JMP NO\_PRESS *; KHONG NUT NAO BAM*

INCREASE: *; HAM XU LY TANG 1s*

CALL DISPLAY\_LOOP

CLR TR1 *; TAT TIMER1*

JNB P0.0, INCREASE *; NHAY TAI CHO DE KIEM TRA CO GIU NUT NHAN*

INC R1 *; TANG 1 GIAY*

SETB TR1 *; BAT TIMER1*

JMP NO\_PRESS *; NHAY TOI NHAN NO\_PRESS*

DECREASE: *; HAM XU LY GIAM 1s*

CALL DISPLAY\_LOOP

CLR TR1 *; TAT TIMER1*

JNB P0.1, DECREASE *; NHAY TAI CHO DE KIEM TRA CO GIU NUT NHAN*

DEC R1 *; GIAM 1 GIAY*

SETB TR1 *; BAT TIMER1*

NO\_PRESS:

CALL DISPLAY\_LOOP *; GOI HAM QUET LED*

JMP LOOP

1. **Report**
2. Design result (screenshot and pasted in the report). **(1 point)**

**Diagram, schematic

Description automatically generated**

1. Explain the operating principle of the effects, accompanied by a video (send a Google Drive link) to demonstrate the circuit operation in case the instructor cannot run the design file. **(3 points)**

**\* Google Drive link:**

[**https://drive.google.com/drive/folders/1AAtaz9caai1kAWlxRj6GhLg12cMjXCMZ?usp=sharing**](https://drive.google.com/drive/folders/1AAtaz9caai1kAWlxRj6GhLg12cMjXCMZ?usp=sharing)

**\*\*Nguyên tắc hoạt động của mạch trên proteus**

**Diagram, schematic

Description automatically generated**

**-**Ta sử dụng điện trở kéo lên ở các nút bấm tại pin P0.0, P0.1, P3.2, P3.3. Khi không bấm, các chân này ở mức cao, khi các nút được bấm chân ở mức thấp.

**-**Pin P0.0 và P0.1 nối với các nút bấm tăng 1s, giảm 1s, được xử lí bằng phần mềm.

**-**Pin P3.2, P3.3 nối với các nút bấm Pause/Resume, Reset, được xử lí bằng ngắt ngoài 0 và ngắt ngoài 1.

**-**Sử dụng IC giải mã 74247 led-7-segments để hỗ trợ hiển thị led 7 đoạn.

**-**Các pin P1.0 -> P1.3 dùng để cấp cực dương cho các led.

**\*\*\*Chương trình đầy đủ hỗ trợ 4 nút bấm: PAUSE/RESUME, RESET, INCREASE, DECREASE**

$NOMOD51

$INCLUDE (8051.MCU)

*; RESET VECTOR*

ORG 0000H

JMP START

*; VECTOR NGAT TIMER1*

ORG 001BH

LJMP ISR\_TIMER1

*; VECTOR NGAT INT0*

ORG 0003H

LJMP ISR\_INT0

*; VECTOR NGAT INT1*

ORG 0013H

LJMP ISR\_INT1

ORG 0100H

START:

MOV TMOD, #10H *; CHON TIME1 CHE DO 1*

MOV TL1, #0F0H *; NAP GIA TRI CHO TIMER1*

MOV TH1, #0D8H

SETB TR1 *; BAT TIMER1*

MOV IE, #10001101B *; CHO PHEP NGAT TIMER1, NGAT INT0, INT1*

SETB TCON.0 *; CHO PHEP NGAT THEO CANH INT0*

SETB TCON.2 *; CHO PHEP NGAT THEO CANH INT1*

MOV R0, #0 *; LUU 1%S*

MOV R1, #0 *; LUU GIAY*

LOOP:

JNB P0.0, INCREASE

JNB P0.1, DECREASE

JMP NO\_PRESS

INCREASE:

CLR TR1

JNB P0.0, $

INC R1

SETB TR1

JMP NO\_PRESS

DECREASE:

CLR TR1

JNB P0.1, $

DEC R1

SETB TR1

NO\_PRESS:

ALL DISPLAY\_LOOP

JMP LOOP

DISPLAY\_LOOP: *; HAM HIEN THI QUET LED*

SETB P1.0

MOV A, R0

MOV B, #10

DIV AB

MOV P2, B

CALL DELAY\_5MS

CLR P1.0

SETB P1.1

MOV A, R0

MOV B, #10

DIV AB

MOV P2, A

CALL DELAY\_5MS

CLR P1.1

SETB P1.2

MOV A, R1

MOV B, #10

DIV AB

MOV P2, B

CALL DELAY\_5MS

CLR P1.2

SETB P1.3 82 MOV A, R1

MOV B, #10

DIV AB

MOV P2, A

CALL DELAY\_5MS

CLR P1.3

RET

DELAY\_10MS: *; HAM DELAY 10ms BANG TIMER1, DUNG DE TANG GIA TRI PHAN TRAM GIAY*

MOV TL1, #0F0H

MOV TH1, #0D8H

SETB TR1

JNB TF1, $

RET

DELAY\_5MS: *; HAM DELAY 5ms BANG VONG LAP, DUNG DE QUET LED*

MOV R4, #10

DELAY:

MOV R5, #250

DJNZ R5, $

DJNZ R4, DELAY

RET

ISR\_TIMER1: *; HAM XU LY BO DEM THOI GIAN KHI TIMER1 TRAN*

*; XU LY TANG DON VI THOI GIAN*

INC R0 *; TANG PHAN TRAM GIAY*

MOV A, R0

MOV B, #100

CJNE A, B, NOT\_EQUAL *; KIEM TRA NEU PHAN TRAM GIAY = 100 THI RESET PHAN TRAM GIAY VE 0*

MOV R0, #0

INC R1 *; TANG GIAY*

MOV A, R1

MOV B, #100

CJNE A, B, NOT\_EQUAL *; KIEM TRA NEU GIAY = 100 THI RESET GIAY VE 0*

MOV R1, #0

NOT\_EQUAL: *; NAP LAI GIA TRI CHO TIMER1*

MOV TL1, #0F0H

MOV TH1, #0D8H

RETI

ISR\_INT0: *; HAM XU LY NUT A*

CPL TR1 *; TAT TIMER1*

RETI

ISR\_INT1: *; HAM XU LY NUT B*

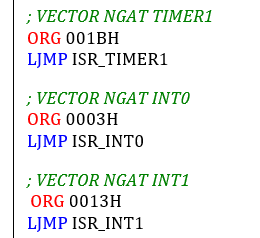
MOV R0, #0 *; GAN PHAN TRAM GIAY = 0*

MOV R1, #0 *; GAN GIAY = 0*

RETI

END

**\*\*\*\*Giải thích chi tiết code:**

****

-Khai báo vector ngắt của Timer1 tại địa chỉ 1BH

-Khai báo vector ngắt ngoài 0, ngắt ngoài 1 lần lượt tại địa chỉ 03H và 13H

Text, letter

Description automatically generated

-Để tạo hàm delay 10ms thì cần nạp giá trị cho Timer1 là D8F0H và bật Timer1 bằng bit TR1

-Set thanh ghi IE với giá trị 10001101B, ứng với bật cho phép ngắt, cho phép ngắt Timer1, cho phép ngắt ngoài INT1, INT0

-Set bit TCON.0, TCON.2 để chọn chế độ ngắt theo cạnh cho ngắt ngoài 0 và 1

-Khởi tạo giá trị R0 và R1 lần lượt lưu giây và phần trăm giây

A picture containing text, screenshot, font

Description automatically generated

-Kiểm tra nút bấm nối với pin P0.0 có bằng 0, nếu bằng 0, tức được bấm, thực hiện tắt Timer1 để ngừng bộ đếm thời gian, kiểm tra nhảy tại chỗ trong trường hợp người nhấn ấn giữ không thả nút bấm, sau đó tăng 1 giây và bật lại Timer1.

-Tiếp tục kiểm tra với pin P0.1, có bằng 0, nếu bằng 0, tức được bấm, thực hiện tắt Timer1 để ngừng bộ đếm thời gian, kiểm tra nhảy tại chỗ trong trường hợp người nhấn ấn giữ không thả nút bấm, sau đó giảm 1 giây và bật lại Timer1..

-Nếu 2 nút trên không nhấn, bỏ qua nó và thực hiện quét led và hiển thị.

A picture containing timeline

Description automatically generated

Text

Description automatically generated

-Thực hiện lần lượt bật tắt các led nối tại các pin P1.0 -> P1.3

-Ta dùng lệnh DIV để chia lấy phần đơn vị và phần chục của giây và phần trăm giây đưa vào port P2 để hiển thị ra led-7-segments.



-Hàm delay bằng Timer1, delay 10ms, khi Timer1 tràn, sẽ gọi hàm ngắt của Timer1 ở địa chỉ 1BH và thực hiện tính toán tăng giá trị giây và phần trăm giây.

Text

Description automatically generated with low confidence

-Hàm delay bằng vòng lặp, thời gian delay là 10\*250\*2us = 5ms. Hàm này dùng để delay khi quét led.

Text

Description automatically generated

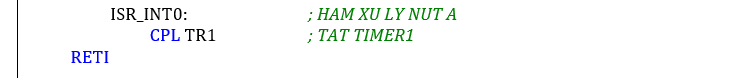
-Hàm ngắt tự động được gọi của Timer1 khi giá trị timer tràn.

-Hàm này thực hiện theo thứ tự sau:

+Tăng phần trăm giây

+Kiểm tra phần trăm giây có bằng 100, nếu bằng reset phần trăm giây về 0 và tăng giây

+Kiểm tra giây có bằng 100, nếu bằng reset giây về 0 và nạp lại giá trị cho Timer1



-Hàm phục vụ ngắt ngoài 0

-Trong hàm này, ta thực hiện đảo bit TR1 ứng với bật hoặc tắt Timer1, điều này sẽ giúp ta PAUSE/RESUME clock.

Text

Description automatically generated

-Hàm phục vụ ngắt ngoài 1

-Trong hàm này ta reset lại giá trị của R0, R1 về 0, ứng với reset giây và phần trăm giây về 0, tức reset clock về 0.

1. Exercise report. **(2 points)**

**\*Giải thích code:**

A picture containing graphical user interface

Description automatically generated

**-**Như đã trình bày trên phần thực hành, ta thực hiện kiểm tra 2 chức năng tăng 1s và giảm 1s bằng phần mềm như sau:

+Kiểm tra nút bấm nối với pin P0.0 có bằng 0, nếu bằng 0, tức được bấm, thực hiện tắt Timer1 để ngừng bộ đếm thời gian, kiểm tra nhảy tại chỗ trong trường hợp người nhấn ấn giữ không thả nút bấm, sau đó tăng 1 giây và bật lại Timer1.

+Tiếp tục kiểm tra với pin P0.1, có bằng 0, nếu bằng 0, tức được bấm, thực hiện tắt Timer1 để ngừng bộ đếm thời gian, kiểm tra nhảy tại chỗ trong trường hợp người nhấn ấn giữ không thả nút bấm, sau đó giảm 1 giây và bật lại Timer1.

+Nếu 2 nút trên không nhấn, bỏ qua nó và thực hiện quét led và hiển thị.

1. So sánh sự khác nhau giữa ngắt theo cạnh (Edge) và ngắt theo mức (Level)?

* Ngắt theo cạnh (Edge): là sử dụng ngắt dựa vào kích cạnh lên hoặc kích cạnh xuống của tín hiệu ngắt. Đối với vi điều khiển 8051, cạnh xuống được dùng để kích hoạt hàm phục vụ ngắt. Khi chân INTx đang ở mức cao 1 chu kỳ máy xuống mức thấp 1 chu kỳ máy thì ngắt được xác lập trong khoảng thời gian đó và bật bit IEx lên 1 sau đó bỏ qua các sườn xuống khác.
* Ngắt theo mức (Level): là sử dụng ngắt dựa vào mức cao hoặc mức thấp của tín hiệu ngắt. Đối với vi điều khiển 8051, tín hiệu ngắt INTx đang ở mức cao ít nhất 1 chu kỳ máy được chuyển xuống mức thấp liên tục 4 chu kỳ máy thì một ngắt theo mức sẽ được xác lập. Khi đó vi điều khiển sẽ dừng công việc đang thực hiện để thực hiện chương trình phục vụ ngắt. Tín hiện múc thấp phải được thả ra trước khi thực hiện lệnh cuối cùng của ISR (chương trình phục vụ ngắt), nếu không thì sẽ có thêm 1 ngắt khác được tạo ra. Điều này làm chúng ta sẽ khó xác định xem phải ấn nút bao lâu để thực hiện 1 ngắt, nếu ấn chưa đủ 4 chu kỳ máy thì chưa tạo ra ngắt, nếu ấn quá lâu thì sẽ tạo nhiều ngắt xảy ra gây nên kết quả không mong muốn.

1. **References**

[1] [Thuật toán tính toán thời gian clock](http://www.dientuvietnam.net/forums/forum/vi-%C4%91i%E1%BB%81u-khi%E1%BB%83n-mcu-b%E1%BB%99-%C4%91i%E1%BB%81u-khi%E1%BB%83n-t%C3%ADn-hi%E1%BB%87u-s%E1%BB%91-dsc/vi-%C4%91i%E1%BB%81u-khi%E1%BB%83n-h%E1%BB%8D-8051/47339-thu%E1%BA%ADt-to%C3%A1n-l%E1%BA%ADp-tr%C3%ACnh-%C4%91%E1%BB%93ng-h%E1%BB%93-d%C3%B9ng-8051)

[2] Tống Văn On – Hoàng Đức Hải, *HỌ VI ĐIỀU KHIỂN 8051,* Nhà xuất bản Lao Động – Xã Hội