**BÀI THÍ NGHIỆM 5**

**THÍ NGHIỆM LẬP TRÌNH NGẮT**

Họ và tên : Dương Phúc Nguyên

MSSV : 1811109

Nhóm lớp : L10

1. **Thí nghiệm 1**

***Yêu cầu***: Viết chương trình làm song song 2 công việc sau: Nhận 1 byte từ máy tính và xuất ngược lại (echo) lên màn hình terminal. Bật LED gắn vào P1.1 khi SW 317 (gắn vào P1.0) được nhấn, và tắt LED khi phím được nhả.

ORG 0000H

LJMP MAIN

ORG 0023H

LJMP ISR\_SERIAL

MAIN:

MOV TMOD,#20H

MOV TH1,#-3

SETB TR1

MOV SCON,#01010010B

MOV IE,#10010000B

D:JB P1.0,TAT

CLR P1.1

SJMP D

TAT: SETB P1.1

SJMP D

ISR\_SERIAL:

JNB RI,NEXT

CLR RI

MOV A,SBUF

MOV SBUF,A

SJMP OUT

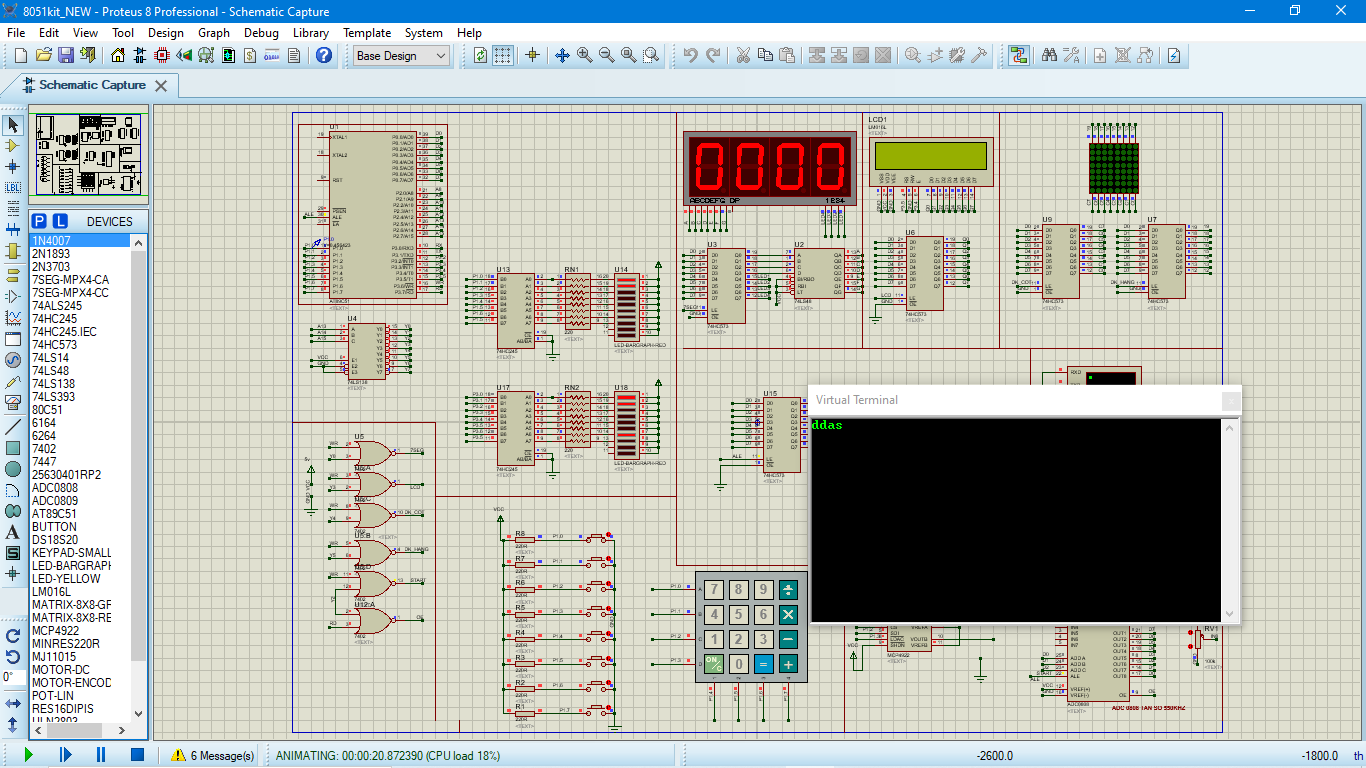
NEXT:

CLR TI

OUT: RETI

END

***Kiểm tra:***

* Biên dịch, thực thi và kiểm tra chương trình
* Tại sao phải xét xem cờ TI hay cờ RI là nguồn gây ra ngắt?
* Do sử dụng ngắt nối tiếp, các cờ TI và RI không được xóa bởi phần cứng.
* Nếu không sử dụng ngắt, có thể viết được chương trình thực thi đầy đủ hai nhiệm vụ như vậy được không?
* Không thể viết được chương trình thực thi đầy đủ hai nhiệm vụ như vậy mà không sử dụng ngắt được.

1. **Thí nghiệm 2**

***Yêu cầu***: Viết chương trình hiển thị lên 4 LED 7 đoạn một số có 4 chữ số, 4 ký số này chứa trong các ô nhớ 20H, 21H, 22H và 23H. Ô nhớ 20H chứa số hàng nghìn.

ORG 0000H

LJMP MAIN

ORG 000BH

LJMP ISR\_T0

ORG 0030H

MAIN:

MOV TMOD,#01H

SETB TF0

MOV R7,#0

MOV IE,#82H

SJMP $

ISR\_T0:

CJNE R7,#0,NEXT1

MOV R1,23H

SJMP TT

NEXT1:

CJNE R7,#1,NEXT2

MOV R1,22H

SJMP TT

NEXT2:

CJNE R7,#2,NEXT3

MOV R1,21H

SJMP TT

NEXT3:

MOV R1,20H

TT:

LCALL DISPLAYLED

INC R7

CJNE R7,#4,RUN

MOV R7,#0

RUN:

CLR TR0

MOV TH0,#HIGH(-9216)

MOV TL0,#LOW(-9216)

SETB TR0

RETI

DISPLAYLED:

MOV A,R7

MOV DPTR,#TABLE

MOVC A,@A+DPTR

ADD A,R1

MOV DPTR,#0000H

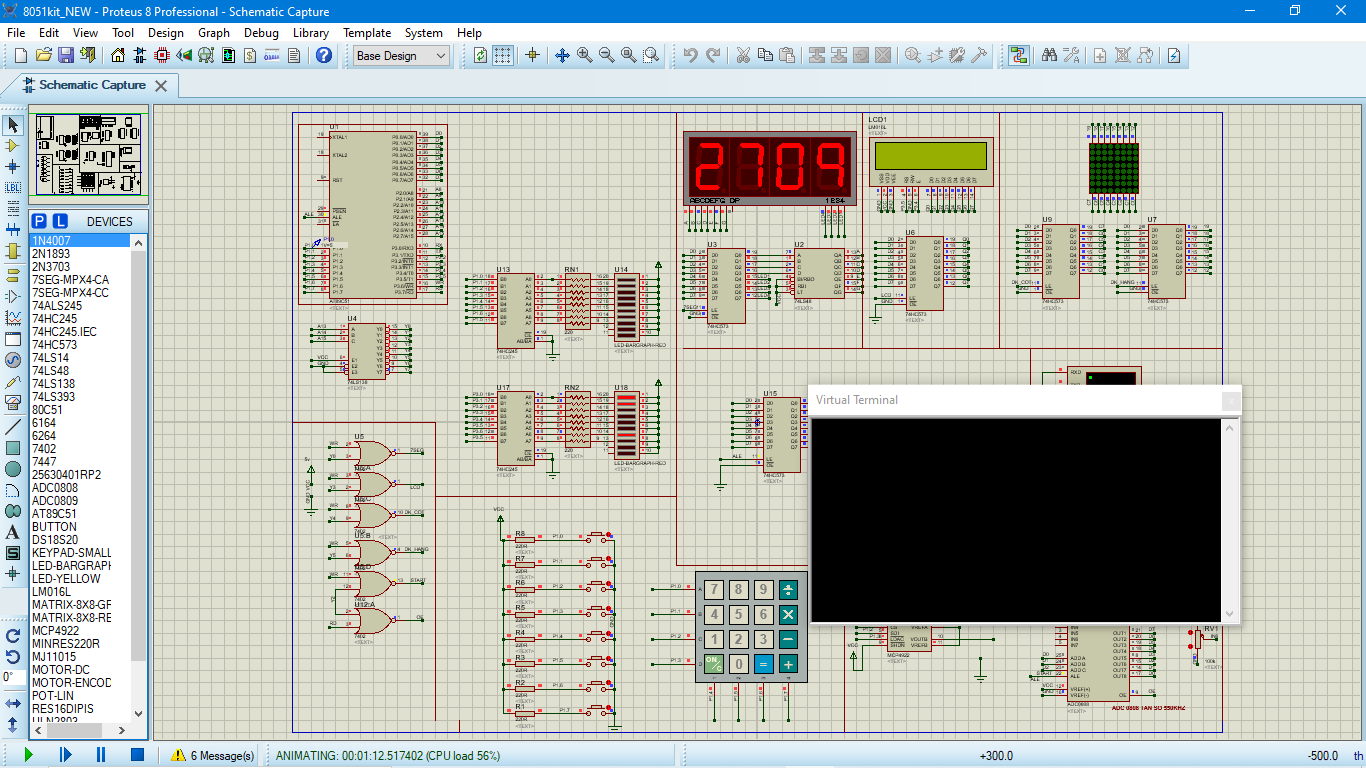
MOVX @DPTR,A

RET

TABLE: DB 0E9H,0D0H, 0B7H, 072H

END

***Kiểm tra:***

* Biên dịch, thực thi và kiếm tra chương trình
* Cờ ngắt của timer được xóa bằng phần mềm hay phần cứng?
* Cờ ngắt của timer được xóa bằng phần mềm.
* Trong thí nghiệm 2, chương trình chính làm nhiệm vụ gì?
* Thiết lập các chế độ của interupt.

1. **Thí nghiệm 3**

***Yêu cầu***: Viết chương trình đếm số lần nhấn phím SW317, hiển thị lên 4 LED 7 đoạn.

ORG 0000H

LJMP MAIN

ORG 000BH

LJMP ISR\_T0

ORG 0030H

MAIN:

MOV TMOD,#01H

SETB TF0

MOV R7,#0

MOV IE,#82H

MOV 23H,#1

MOV 22H,#0

MOV 21H,#0

MOV 20H,#0

LAP: JB P1.0,TIEP

INC 23H

MOV R2,23H

CJNE R2,#10,TIEP

MOV 23H,#0

INC 22H

MOV R2,22H

CJNE R2,#10,TIEP

MOV 23H,#0

MOV 22H,#0

INC 21H

MOV R2,21H

CJNE R2,#10,TIEP

MOV 23H,#0

MOV 22H,#0

MOV 21H,#0

INC 20H

TIEP:

MOV R5,#250

L: MOV R4,#250

DJNZ R4,$

DJNZ R5,L

SJMP LAP

ISR\_T0:

CJNE R7,#0,NEXT1

MOV R1,23H

SJMP TT

NEXT1:

CJNE R7,#1,NEXT2

MOV R1,22H

SJMP TT

NEXT2:

CJNE R7,#2,NEXT3

MOV R1,21H

SJMP TT

NEXT3:

MOV R1,20H

TT:

LCALL DISPLAYLED

INC R7

CJNE R7,#4,RUN

MOV R7,#0

RUN:

CLR TR0

MOV TH0,#HIGH(-9216)

MOV TL0,#LOW(-9216)

SETB TR0

RETI

DISPLAYLED:

MOV A,R7

MOV DPTR,#TABLE

MOVC A,@A+DPTR

ADD A,R1

MOV DPTR,#0000H

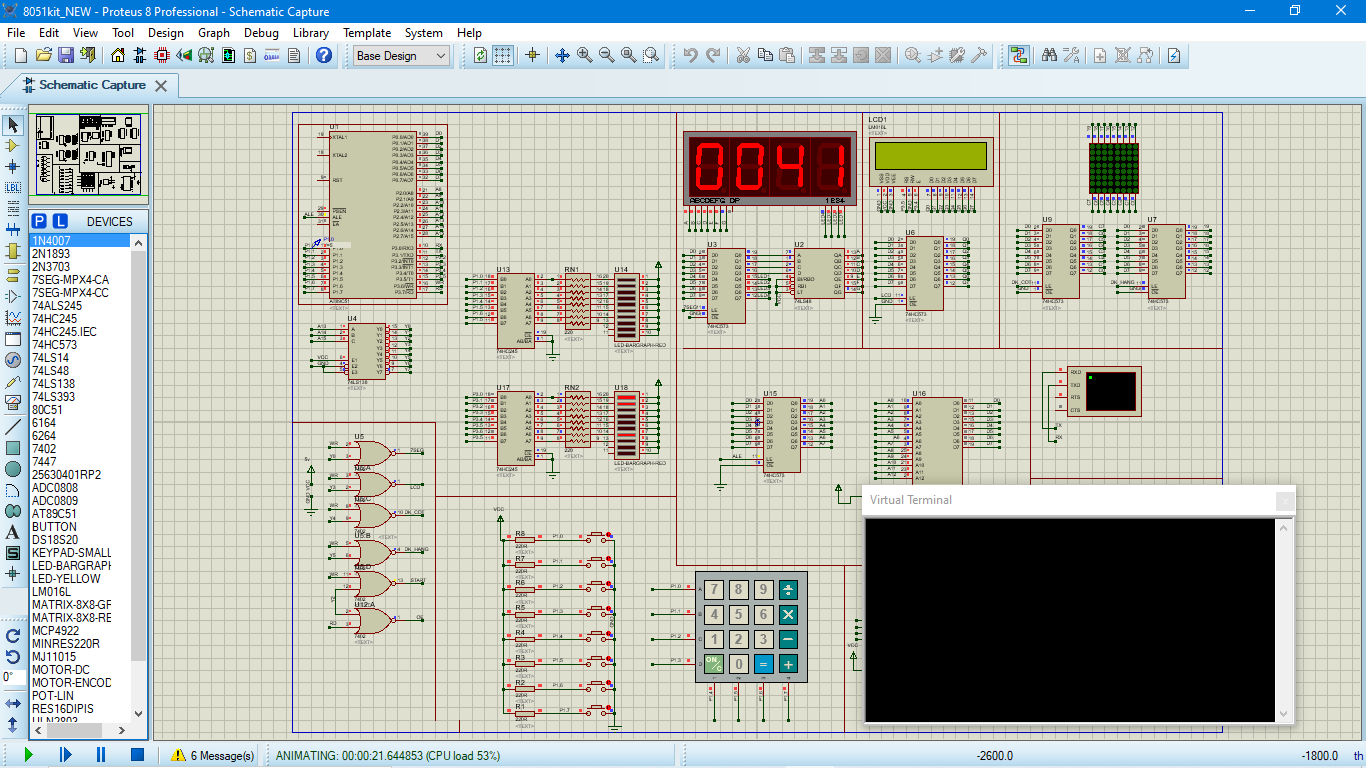
MOVX @DPTR,A

RET

TABLE: DB 0E0H,0D0H, 0B0H, 070H

END

***Kiểm tra:***

* Biên dịch, thực thi và kiếm tra chương trình.
* Trong thí nghiệm 3, chương trình chính làm nhiệm vụ gì?
* Cấu hình timer 0 và timer 1.
* Set các giá trị ban đầu là 0 cho các ô nhớ 20h, 21h, 22h, 23h.
* Đếm số lần nhấn switch.
* Daylay tạo cho quá trình chống rung của nút (nếu có) nên được tạo bằng cách nào?
* Chống rung của nút nhấn nên được tạo bằng phần mềm và dùng timer để tăng độ chính xác

1. **Thí nghiệm 4**

***Yêu cầu***: Viết chương trình thực hiện đồng hồ bấm giây, hiển thị lên LED 7 đoạn. Hai LED 3 và 2 thể hiện số giây, 2 LED 1 và 0 thể hiện phần trăm của giây.

Khi:

SW 310 nhấn, đồng hồ reset về 0.

SW 311 nhấn, đồng hồ bắt đầu chạy.

SW 312 nhấn, đồng hồ dừng lại.

ORG 0000H

LJMP MAIN

ORG 000BH

LJMP ISR\_T0

ORG 0030H

MAIN:

MOV TMOD,#01H

SETB TF0

MOV R7,#0

MOV IE,#82H

RE:

MOV 23H,#0

MOV 22H,#0

MOV 21H,#0

MOV 20H,#0

START:

JNB P1.2 ,RE; P1.2 RESET

JB P1.0,START; P1.0 START

LAP:

JNB P1.2 ,RE

JB P1.1,LA; P1.1 STOP

SJMP START

LA:

INC 23H

MOV R2,23H

CJNE R2,#10,TIEP

MOV 23H,#0

INC 22H

MOV R2,22H

CJNE R2,#10,TIEP

MOV 23H,#0

MOV 22H,#0

INC 21H

MOV R2,21H

CJNE R2,#10,TIEP

MOV 23H,#0

MOV 22H,#0

MOV 21H,#0

INC 20H

TIEP:

MOV R5,#20

L:MOV R4,#250

DJNZ R4,$

DJNZ R5,L

SJMP LAP

ISR\_T0:

CJNE R7,#0,NEXT1

MOV R1,23H

SJMP TT

NEXT1:

CJNE R7,#1,NEXT2

MOV R1,22H

SJMP TT

NEXT2:

CJNE R7,#2,NEXT3

MOV R1,21H

SJMP TT

NEXT3:

MOV R1,20H

TT:

LCALL DISPLAYLED

INC R7

CJNE R7,#4,RUN

MOV R7,#0

RUN:

CLR TR0

MOV TH0,#HIGH(-9216)

MOV TL0,#LOW(-9216)

SETB TR0

RETI

DISPLAYLED:

MOV A,R7

MOV DPTR,#TABLE

MOVC A,@A+DPTR

ADD A,R1

MOV DPTR,#0000H

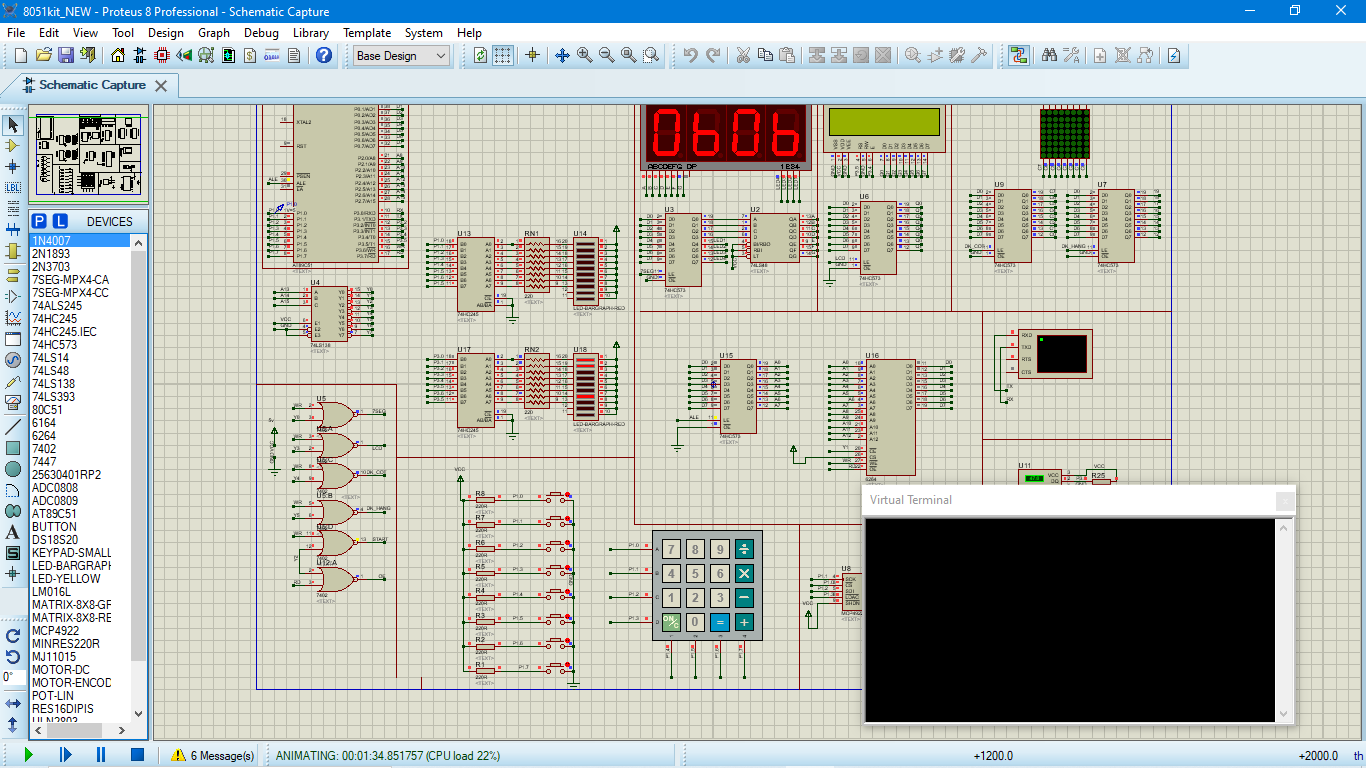
MOVX @DPTR,A

RET

TABLE: DB 0E0H,0D0H, 0B0H, 070H

END

***Kiểm tra:***

* Biên dịch và kiểm tra hoạt động của chương trình.
* Các nút nhấn SW310, SW311, SW312 được xử lý như thế nào? Trình bày cụ thể?
* SW310 dùng để reset về 0
* SW311 dùng để bắt đầu đếm
* SW312 dùng để kết thúc
* Đầu tiên khi chạy chương trình, các giá trị của LED 7 đoạn sẽ là 0000, sau khi SW311 được nhấn, chương trình bắt đầu đếm cho đến khi nhận được tín hiệu ngắt được khởi tạo từ nút nhấn SW312, sau đó chương trình sẽ tiếp tục đếm nếu nhấn SW 311 hoặc reset tất cả về 0 nếu nhấn SW310.