

Hiệu đính từ slide của thầy Hồ Trung Mỹ
Bộ môn Điện tử - DH BK TPHCM

CHƯƠNG 3

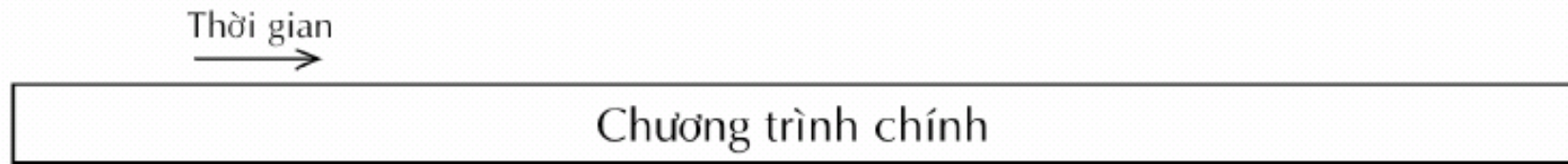
HỘ VI ĐIỀU KHIỂN 8051

3.7 Ngắt (Interrupt)

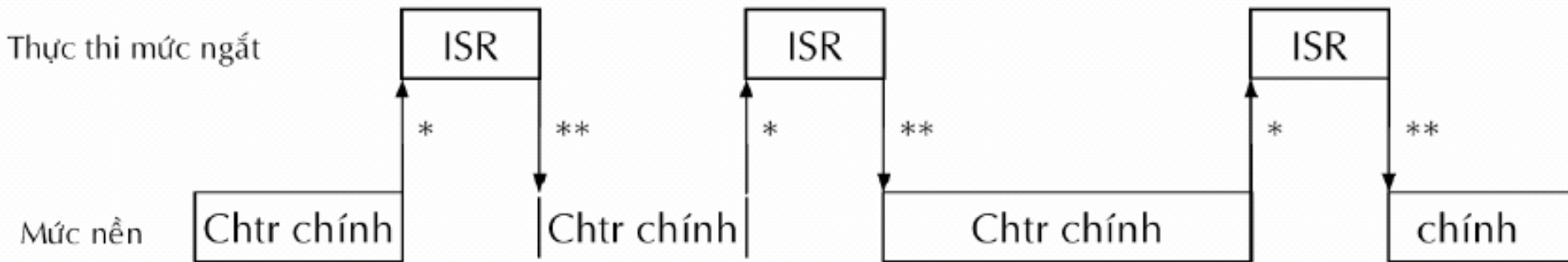
Giới thiệu ngắt

- Ngắt là sự xảy ra một điều kiện-sự kiện làm cho treo tạm thời chương trình trong khi đó điều kiện này được phục vụ bởi một chương trình khác
- Hệ thống được điều khiển bằng ngắt – làm nhiều việc đồng thời
 - Đa nhiệm (vụ) dựa trên Hardware
 - Không có hồi vòng
 - Khi có các sự kiện cụ thể (ngắt) xảy ra, CPU sẽ nhảy đến 1 chương trình con cụ thể – chương trình phục vụ ngắt (ISR), xử lý ngắt
 - Công việc mức nền (ở Foreground) với công việc mức ngắt (ở Background)

Thực thi chương trình



a) Thực thi chương trình không có ngắt.



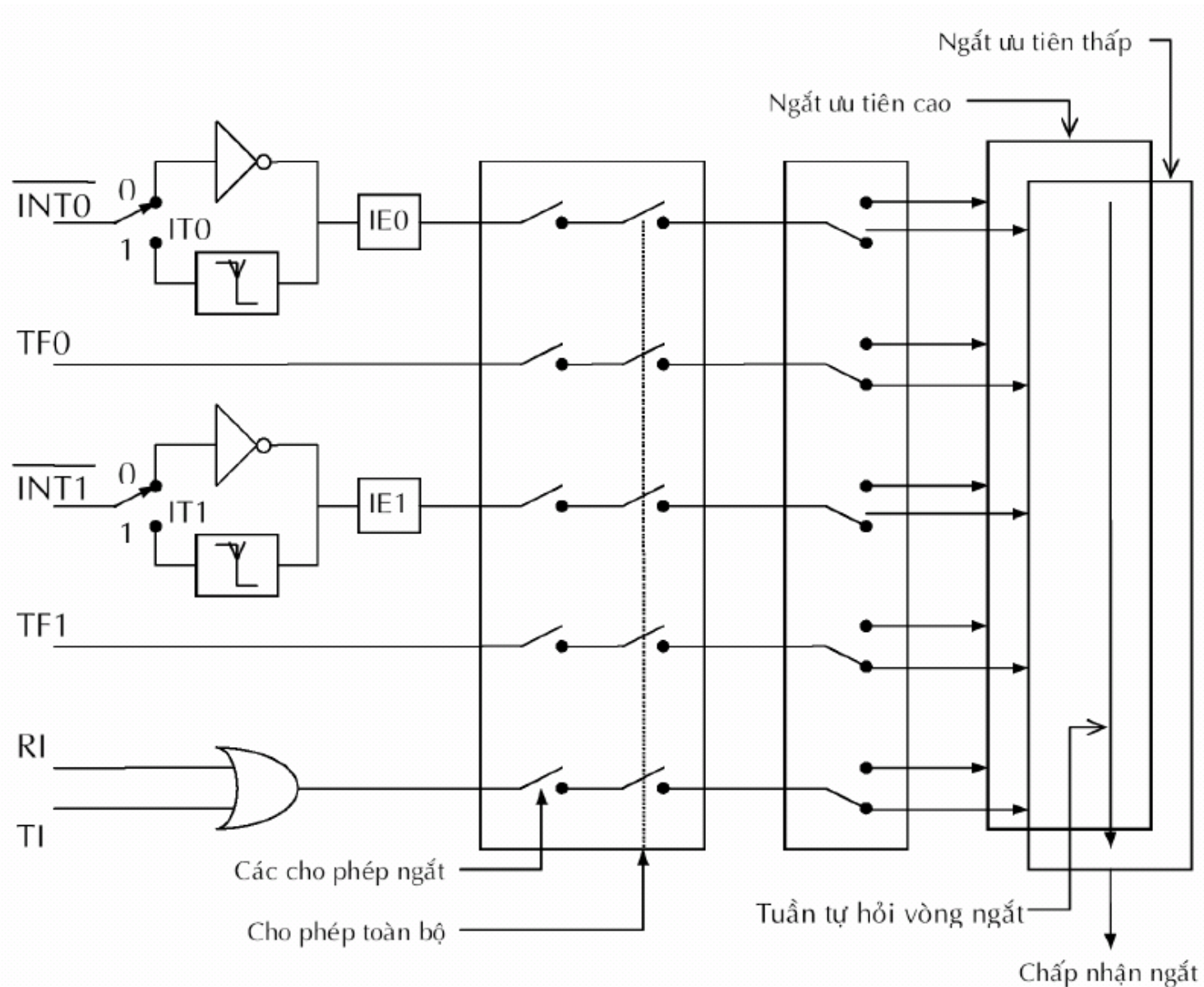
b) Thực thi chương trình có ngắt.

* : gọi ngắt.

** : quay về từ ngắt.

Hình 3.36 Thực thi chương trình có ngắt và không có ngắt.

Tổng quan về cấu trúc ngắt của 8051



Các nguồn ngắt

Bảng 3.26 Các bit cờ ngắt.

Ngắt	Cờ	Bit trong SFR
Bên ngoài 0	IE0	TCON.1
Bên ngoài 1	IE1	TCON.3
Timer 1	TF1	TCON.7
Timer 0	TF0	TCON.5
Cổng nối tiếp	TI	SCON.1
Cổng nối tiếp	RI	SCON.0
Timer 2	TF2	T2CON.7 (8052)
Timer 2	EXF2	T2CON.6 (8052)

Thanh ghi cho phép ngắt IE

- Mặt nạ ngắt

Bảng 3.24 Tóm tắt thanh ghi cho phép ngắt IE (Interrupt Enable).

Bit	Ký hiệu	Địa chỉ bit	Mô tả (1 = cho phép, 0 = cấm)
IE.7	EA	AFH	Cho phép/cấm toàn bộ.
IE.6	–	AEH	Không được định nghĩa.
IE.5	ET2	ADH	Cho phép ngắt từ Timer 2 (8052).
IE.4	ES	ACH	Cho phép ngắt của cổng nối tiếp.
IE.3	ET1	ABH	Cho phép ngắt từ Timer 1.
IE.2	EX1	AAH	Cho phép ngắt ngoài 1.
IE.1	ET0	A9H	Cho phép ngắt từ Timer 0.
IE.0	EX0	A8H	Cho phép ngắt ngoài 0.

- Nếu có nhiều nguồn ngắt xảy ra?
 - Mã hóa ưu tiên được sử dụng.
 - Chỉ có 2 cấp ưu tiên trong 8051 \Rightarrow Vẫn có nhiều ngắt có thể có cùng ưu tiên ngắt

Thứ tự ưu tiên ngắt

Bảng 3.25 Tóm tắt thanh ghi ưu tiên ngắt IP (Interrupt Priority) .

Bit	Ký hiệu	Địa chỉ bit	Mô tả (1 = mức cao hơn, 0 = mức thấp hơn)
IP.7	–	–	Không được định nghĩa.
IP.6	–	–	Không được định nghĩa.
IP.5	PT2	BDH	Ưu tiên cho ngắt từ Timer 2 (8052).
IP.4	PS	BCH	Ưu tiên cho ngắt của cổng nối tiếp.
IP.3	PT1	BBH	Ưu tiên cho ngắt từ Timer 1.
IP.2	PX1	BAH	Ưu tiên cho ngắt ngoài 1.
IP.1	PT0	B9H	Ưu tiên cho ngắt từ Timer 0.
IP.0	PX0	B8H	Ưu tiên cho ngắt ngoài 0.

Xử lý ngắt

Khi ngắt xảy ra và được CPU chấp nhận, chương trình chính bị ngắt quãng. Các hoạt động sau sẽ xảy ra:

- Lệnh hiện hành hoàn tất việc thực thi
- CPU sẽ cất PC vào ngăn xếp (**địa chỉ quay về**)
- CPU sẽ nhảy đến ISR khác nhau theo nguồn ngắt khác nhau
- $PC \leftarrow \text{địa chỉ vector ngắt} (= 3 + (\# \text{ of nguồn ngắt}) * 8)$
- Thực thi ISR

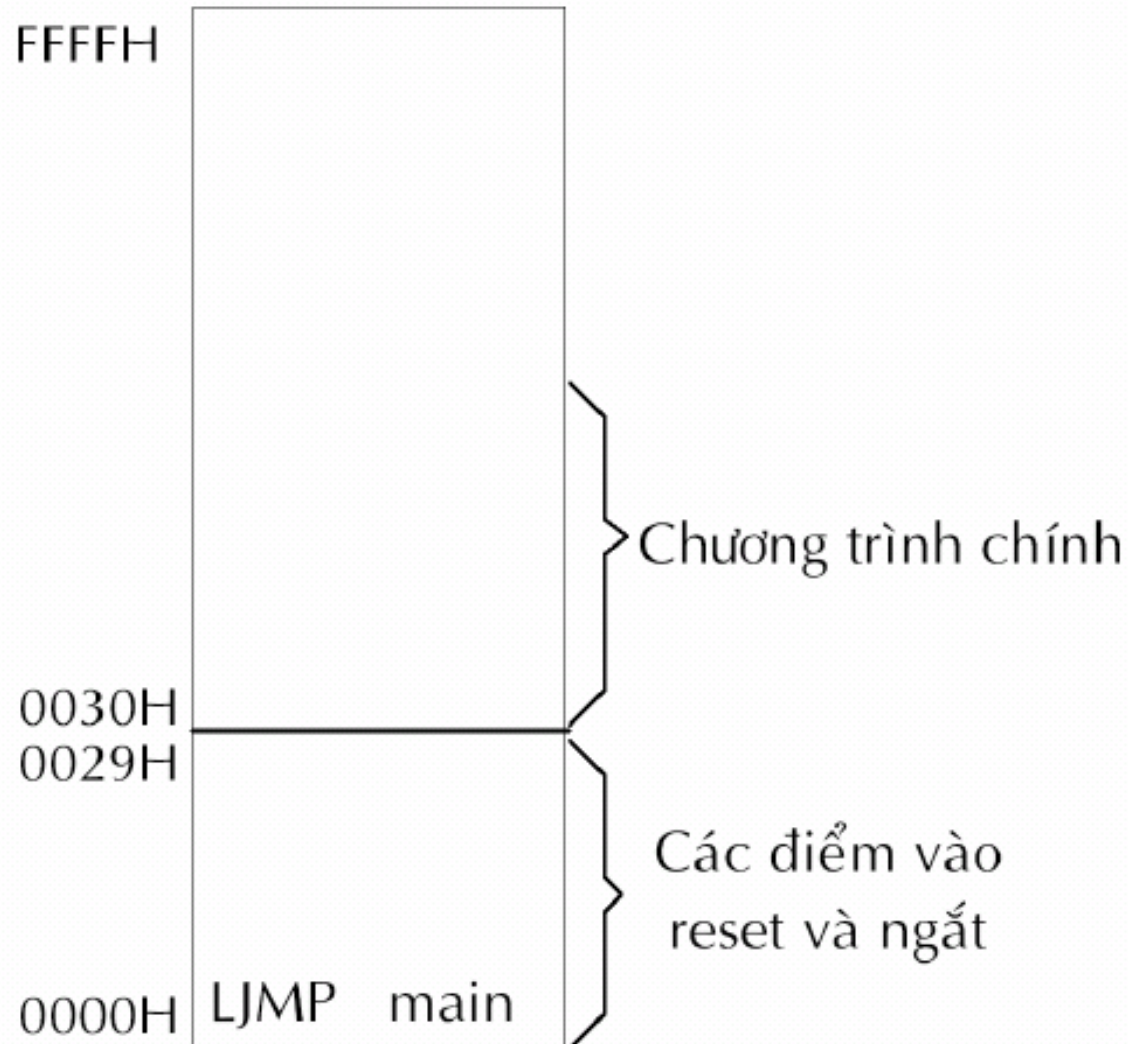
Các vector ngắt

Bảng 3.27 Các vector ngắt.

Ngắt	Cờ	Địa chỉ vector
Reset hệ thống	RST	0000H
Bên ngoài 0	IE0	0003H
Timer 0	TF0	000BH
Bên ngoài 1	IE1	0013H
Timer 1	TF1	001BH
Cổng nối tiếp	TI hoặc RI	0023H
Timer 2	TF2 hoặc EXF2	002BH

Tổ chức bộ nhớ khi sử dụng ngắt

Bộ nhớ chương trình bên ngoài



Khung chương trình dùng ngắt

```
ORG 0000H ; Điểm vào reset
```

```
LJMP MAIN
```

```
. ; Các điểm vào ISR
```

```
.
```

```
.
```

```
ORG 0030H ; Điểm vào chương trình chính
```

```
MAIN :
```

```
.
```

```
. ; Chương trình chính bắt đầu.
```

Lệnh thứ nhất nhảy đến địa chỉ 0030H, vừa trên các vị trí vector mà các ISR bắt đầu

Chương trình phục vụ ngắt có kích thước nhỏ

Nếu chỉ có một nguồn ngắt được sử dụng, ví dụ Timer 0, thì có thể sử dụng khung chương trình sau :

```
                ORG  0000H                ; Reset
                LJMP MAIN
                ORG  000BH                ; Điểm vào ISR cho Timer 0
T0ISR:          .                        ; Bắt đầu ISR cho Timer 0
                .
                RETI                    ; Quay về chương trình chính
MAIN:          .                        ; Chương trình chính
                .
                .
```

Nếu sử dụng nhiều ngắt, thì cẩn thận, phải bảo đảm là chúng bắt đầu ở vị trí đúng và không chạy lố sang ISR kế. Vì chỉ có một ngắt được sử dụng trong ví dụ trên, chương trình chính có thể bắt đầu ngay sau lệnh RETI.

Chương trình phục vụ ngắt có kích thước lớn

Thí dụ lúc này chỉ xét Timer 0, có thể sử dụng khung sau :

```
ORG 0000H           ; Điểm vào reset
LJMP MAIN
ORG 000BH           ; Điểm vào ISR của Timer 0
LJMP T0ISR
ORG 0030H           ; Các vector ngắt tiếp theo
MAIN:               ; Chương trình chính
.
.
T0ISR:              ; Bắt đầu ISR cho Timer 0
.
.
RETI                ; Quay về chương trình chính.
```

Để đơn giản, chương trình của chúng ta sẽ chỉ làm một việc lúc ban đầu. Chương trình khởi động timer, công nối tiếp và các thanh ghi ngắt cho thích hợp và rồi không làm gì cả. Công việc hoàn toàn được làm trong ISR. Sau các lệnh khởi động, chương trình chính chứa lệnh sau :

```
HERE: SJMP HERE
```

hay dạng viết gọn như sau:

```
SJMP $
```

a) Ngắt timer

- Ngắt timer xảy ra khi các cờ tràn TFX được bật lên một
- Cờ tạo ra ngắt này được xóa bằng phần cứng khi CPU chuyển đến chương trình phục vụ ngắt (ISR)

Thí dụ: Tạo sóng vuông bằng ngắt của Timer

Các ngắt của Timer xảy ra khi các thanh ghi timer TLx/THx tràn và đặt cờ báo tràn lên 1 (TFx). Ta có chương trình như sau:

Địa chỉ Mã máy	Dòng lệnh	Nhãn	Lệnh	Chú thích
0000	5		ORG	0000H
0000	020030	6	LJMP	MAIN
000B	7		ORG	000BH
000B	B290	8	T0ISR: CPL	P1.0
000D	32	9	RETI	
0030	10		ORG	0030H
0030	758902	11	MAIN: MOV	TMOD, #02H
0033	758CCE	12	MOV	TH0, #-50
0036	D28C	13	SETB	TR0
0038	75A882	14	MOV	IE, #82H
003B	80FE	15	SJMP	\$
	16		END	

Phân tích CT Tạo sóng vuông bằng ngắt Timer

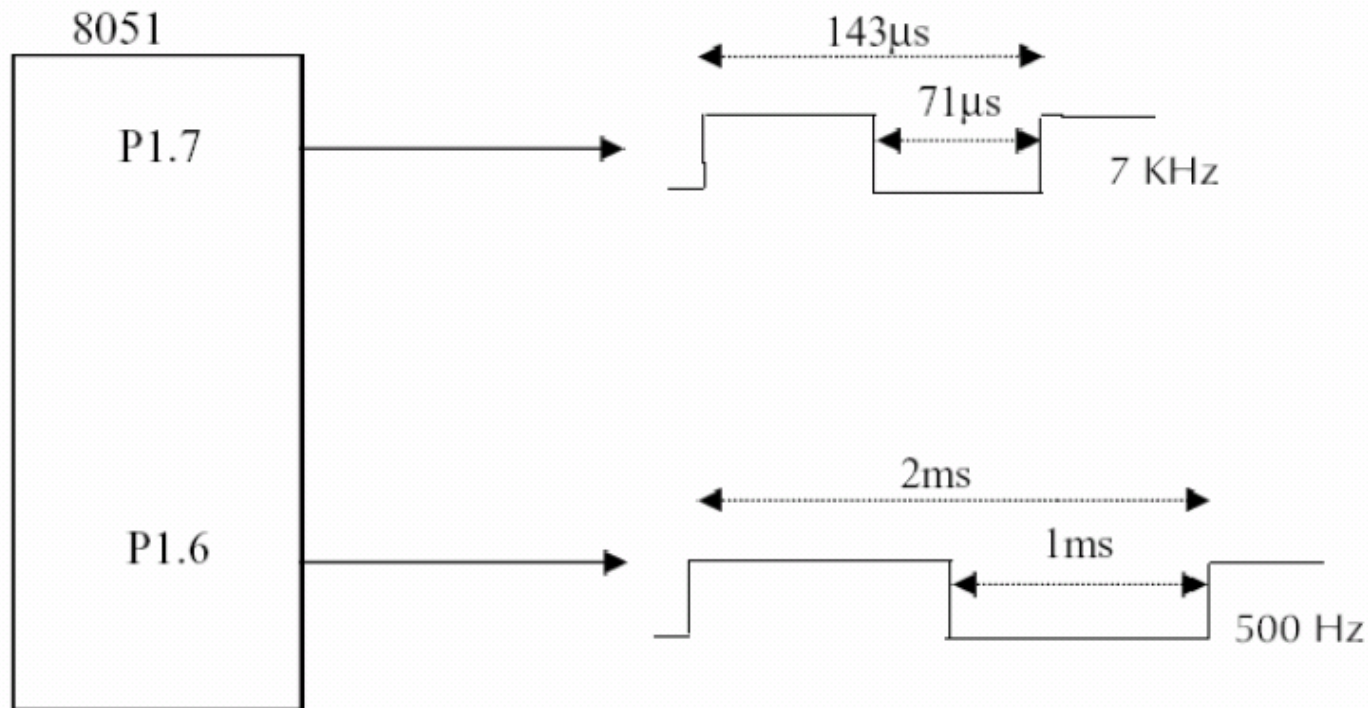
- Ngay sau khi reset, PC được nạp trị 0000H.
- Lệnh đầu tiên được thực thi LJMP MAIN mà rẽ nhánh bỏ qua ISR của Timer 0 đến địa chỉ 0030H trong bộ nhớ mã.
- Ba lệnh kế (các dòng 11–13) khởi trị Timer 0 chế độ 2 tràn sau 50 μ s.
- Lệnh MOV IE, #82H cho phép ngắt của Timer 0. Dĩ nhiên tràn thứ nhất sẽ không xảy ra trong vì có trì hoãn ở các lệnh khởi tạo trị...
- Cứ sau 50 μ s một ngắt xảy ra; chương trình chính bị ngắt và ISR (của Timer 0) thực thi. ISR này đảo trạng thái bit cổng và quay về chương trình gọi nó (vòng lặp tại chỗ) và tiếp tục như vậy với 50 μ s kế.
- **Chú ý là cờ timer TF0 không bị xóa bằng phần mềm. Khi các ngắt được cho phép, TF0 tự động bị xóa bằng phần cứng khi CPU chỉ đến ngắt.**
- Tình cờ địa chỉ quay về trong chương trình chính là địa chỉ của lệnh SJMP. Địa chỉ được cất vào ngăn xếp bên trong trước khi chỉ đến mỗi ngắt và được lấy lại từ ngăn xếp khi thực thi lệnh RETI ở cuối ISR. Vì SP đã không được khởi tạo trị, do đó mặc nhiên nó có giá trị reset là 07H. Tác vụ cất (PUSH) để địa chỉ quay về trong các ô nhớ RAM nội 08H (PCL = byte thấp của PC) và 09H (PCH = byte cao của PC).

Thí dụ: Tạo hai sóng vuông dùng ngắt (1/2)

Viết chương trình dùng các ngắt để tạo ra các sóng vuông đồng thời 7 KHz và 500 Hz ở các chân P1.7 và P1.6.

Bài giải.

Cấu hình phần cứng với những định thì cho các dạng sóng mong muốn được cho trong hình sau:



Hình 3.39 Dạng sóng 7 KHz và 500 Hz.

Thí dụ: Tạo hai sóng vuông dùng ngắt (2/2)

Địa chỉ	Mã máy	Dòng lệnh	Nhãn	Lệnh	Chú thích
0000		5		ORG	0000H
0000	020030	6		LJMP	MAIN
000B		7		ORG	000BH
000B	02003F	8		LJMP	T0ISR
001B		9		ORG	001BH
001B	020042	10		LJMP	T1ISR
0030		11		ORG	0030H
0030	758912	12	MAIN:	MOV	TMOD, #12H
		13			
0033	758CB9	14		MOV	TH0, #-71
0036	D28C	15		SETB	TR0
0038	D28F	16		SETB	TF1
003A	75A88A	17		MOV	IE, #8AH
003D	80FE	18		SJMP	\$
		19			
003F	B297	20	T0ISR:	CPL	P1.7
0041	32	21		RETI	
0042	C28E	22	T1ISR:	CLR	TR1
0044	758DFC	23		MOV	TH1, #HIGH(-1000)
0047	758B18	24		MOV	TL1, #LOW(-1000)
004A	D28E	25		SETB	TR1
004C	B296	26		CPL	P1.6
004E	32	27		RETI	
		28		END	

Ngắt timer

VD: Tạo sóng vuông có tần số 100Hz trên chân P1.2, duty cycle = 25% dùng ngắt timer 1, XTAL = 6MHz

b) Ngắt cổng nối tiếp

- Ngắt cổng nối tiếp xảy ra khi các cờ ngắt TI hoặc RI được bật lên 1
 - Ngắt phát xảy ra khi truyền xong một ký tự vừa được ghi vào SBUF (TI = 1)
 - Ngắt thu xảy ra khi một ký tự đã nhận xong và đang đợi trong SBUF để được đọc (RI = 1)
- Cờ gây ngắt của cổng nối tiếp (RI hoặc TI) không được xóa bởi phần cứng mà phải được xóa bằng phần mềm

b) Ngắt công nối tiếp

Thí dụ: **Xuất ký tự bằng ngắt**

Viết chương trình bằng cách dùng ngắt để liên tục phát bộ mã ASCII (ngoại trừ các mã điều khiển) đến thiết bị đầu cuối gắn ở cổng nối tiếp 8051.

Bài giải.

Trong bộ mã ASCII có 95 mã đồ họa (20H đến 7EH) và 33 mã điều khiển (00H đến 1FH, và 7FH). Ta có chương trình như sau

```
ORG 0
LJMP MAIN
ORG 0023H
LJMP SPISR
ORG 0030H
MAIN: MOV TMOD, #20H
      MOV TH1, #-26
      SETB TR1
      MOV SCON, #42H

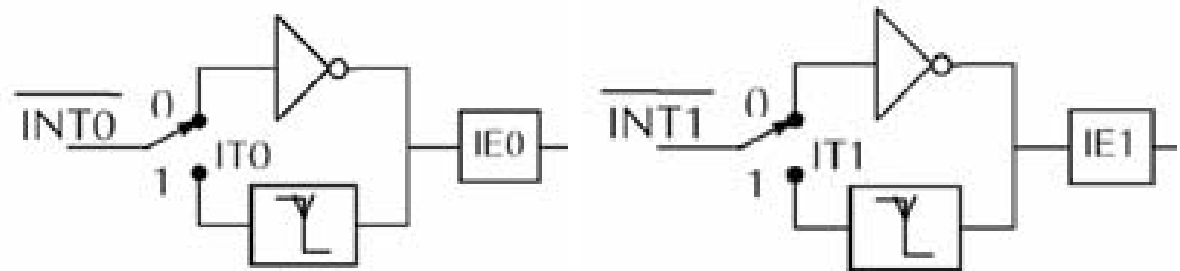
      MOV A, #20H
      MOV IE, #90H
      SJMP $
;
SPISR: CJNE A, #7FH, SKIP
      MOV A, #20H
SKIP:  MOV SBUF, A
      INC A
      CLR TI
      RETI
      END
```

Ngắt cổng nối tiếp

VD: Viết chương trình nhận một ký tự ASCII từ cổng nối tiếp, tốc độ baud 9600bps, XTAL = 12MHz, SMOD = 1

c) Ngắt ngoài

- Các ngắt ngoài được tạo ra khi có cạnh xuống hoặc mức thấp trên chân $\overline{INT0}$ hoặc $\overline{INT1}$
- Sự lựa chọn ngắt tích cực theo cạnh hay mức được lập trình qua bit $IT0$ hay $IT1$ trong $TCON$



- Các cờ tạo ngắt là các bit $IE0$ và $IE1$ trong $TCON$

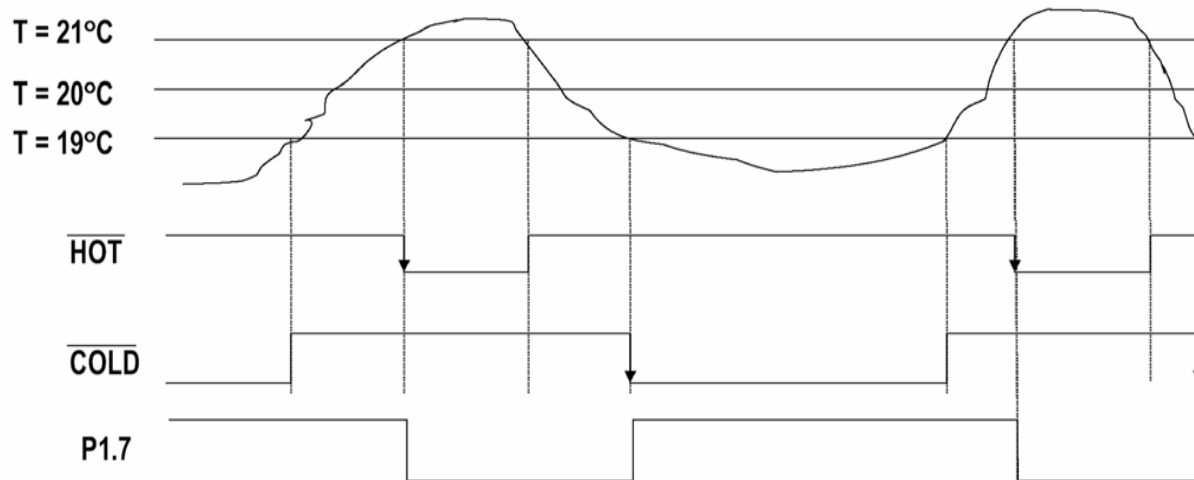
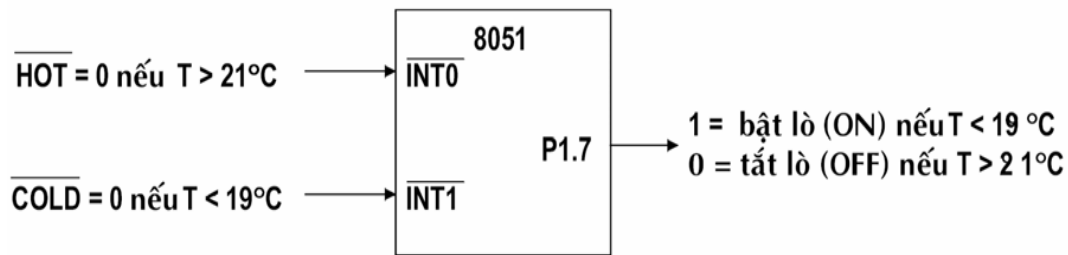
Thí dụ: Mạch điều khiển lò

Hãy sử dụng ngắt 8051 để thiết kế mạch điều khiển lò giữ cho nhiệt độ trong nhà ở $20^{\circ}\text{C} \pm 1^{\circ}\text{C}$.

Bài giải.

Giao tiếp sau được giả sử cho thí dụ này. Solenoid để ON/OFF (mở/tắt) lò được nối vào P1.7 sao cho

P1.7 = 1 để cấp năng lượng cho solenoid (lò ON)



```

                ORG    0
                LJMP   MAIN
                ; vector EXT 0 ở 0003H
EX0ISR:        CLR    P1.7      ; tắt lò (OFF)
                RETI
                ORG    0013H
EX1ISR:        SETB   P1.7      ; bắt lò (ON)
                RETI
                ORG    30H
MAIN:          MOV    IE, #58H   ; cho phép ngắt ngoài
                SETB   IT0       ; kích cạnh xuống
                SETB   IT1
                SETB   P1.7      ; tắt lò (OFF)
                JB     P3.2, SKIP ; nếu T > 21 °C thì
                CLR    P1.7      ; tắt lò (OFF)
SKIP:         SJMP   $          ; lặp tại chỗ
                END

```