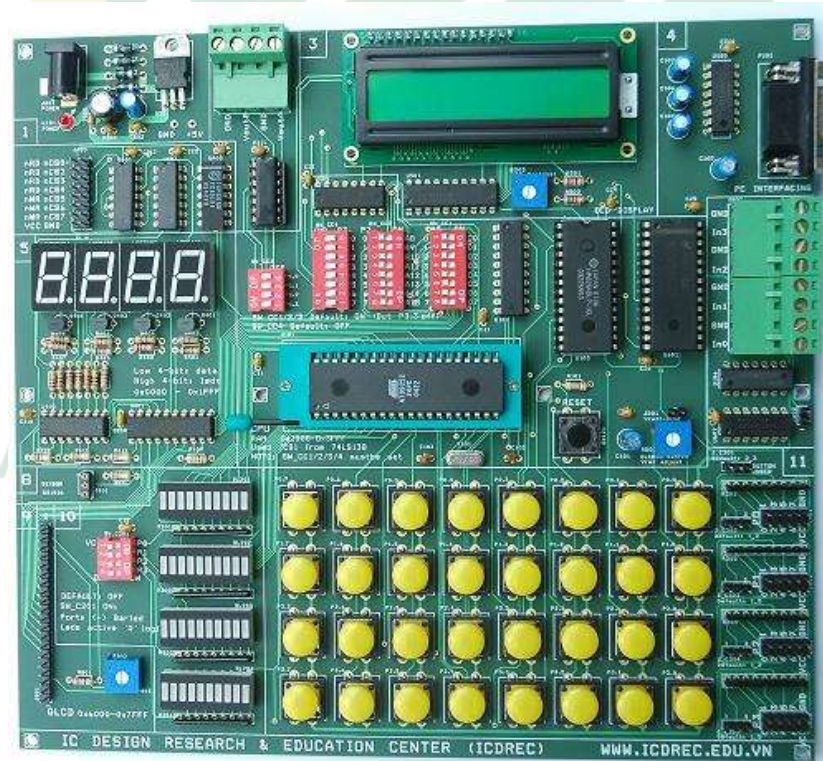


Chương 2: Họ vi điều khiển 8051 (Interrupt)



- Hiểu được định nghĩa và các interrupts trong vđk 8051
- Biết được cách sử dụng các ngắt ngoài
- Biết được cách sử dụng các ngắt Timer, UART
- Vận dụng để viết các chương trình đơn giản trên 8051

- Interrupts
 - External
 - Timer
 - UART

- Timer là gì, cách hoạt động?
- Có bao nhiêu Timer trong 8051?
- Có bao nhiêu mode hoạt động của Timer?
- Có bao nhiêu mode hoạt động của UART?
- Tính tốc độ BAUD như thế nào?

- Một ngắt là một sự kiện bên trong hoặc bên ngoài làm ngắt bộ vi điều khiển để báo cho nó biết rằng thiết bị cần dịch vụ của nó.
- Chương trình đi cùng với ngắt được gọi là trình dịch vụ ngắt (ISR) hoặc trình quản lý ngắt (IH)
- Nhóm các vị trí nhớ được dành riêng để gửi các địa chỉ của các ISR gọi là bảng vector ngắt



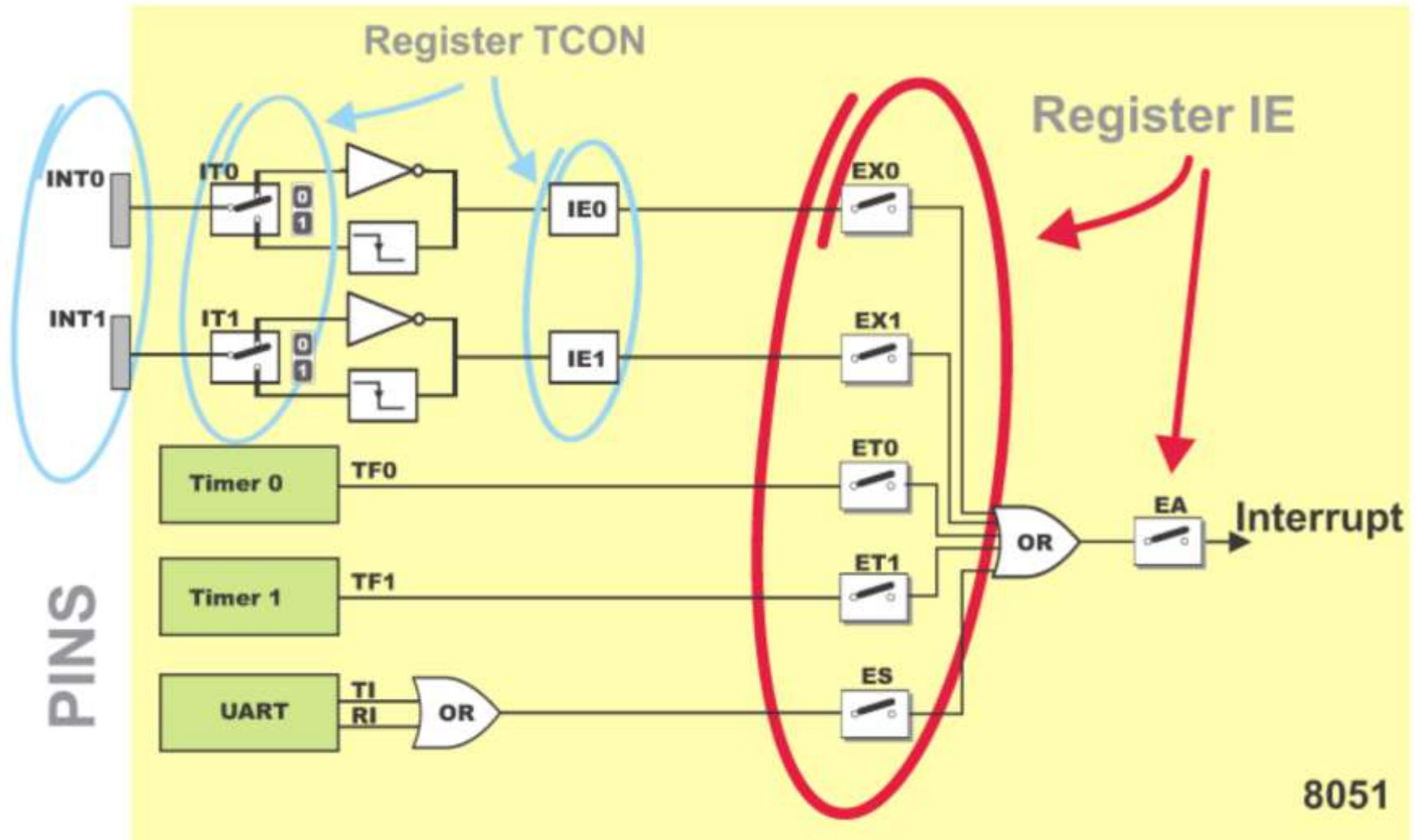
Interrupts (tt)



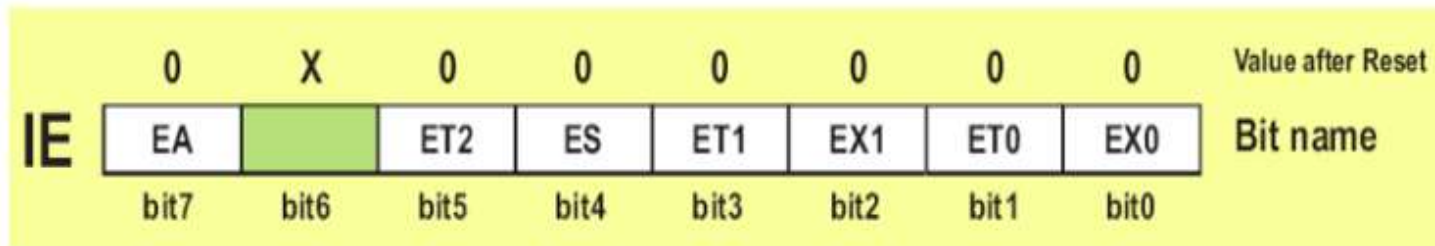
```
ORG 0
rom_start: LJMP
main_code
ORG 13H
int1_vec: LJMP
int1_isr
ORG 30H
main_code:    ;bla bla
; ....
int1_isr:     ;bla bla
```

Interrupt	ROM Location	Pin
Reset	0000H	9
INT0	0003H	P3.2
TF0	000BH	
INT1	0013H	P3.3
TF1	001BH	
S0	0023H	

Interrupts (tt)



Thanh ghi cho Interrupts

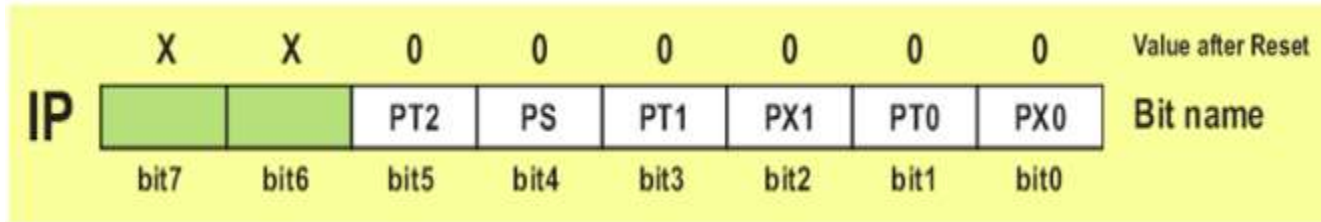


Trong đó:

Bit	Mô tả
EA	Cho phép/cấm ngắt toàn cục = 0: Cấm tất cả các ngắt = 1: Cho phép các ngắt
ES	= 0: Cấm ngắt truyền thông nối tiếp = 1: Cho phép ngắt truyền thông nối tiếp
ET1	= 0: Cấm ngắt Timer 1 = 1: Cho phép ngắt Timer 1
EX1	= 0: Cấm ngắt ngoại vi INT0 = 1: Cho phép ngắt ngoại vi INT0
ET0	= 0: Cấm ngắt Timer 0 = 1: Cho phép ngắt timer 0
EX0	= 0: Cấm ngắt ngoại vi INT1 = 1: Cho phép

- Kết thúc lệnh đang thực hiện và lưu địa chỉ của lệnh kế tiếp (PC) vào ngăn xếp
- Lưu lại tình trạng hiện tại của tất cả các ngắt
- Nhảy đến bảng vector ngắt tại địa chỉ của ISR
- Nhận địa chỉ ISR và thực hiện cho đến lệnh cuối cùng của ISR và RETI
- Nhận địa chỉ PC từ ngăn xếp và thực hiện các lệnh tiếp theo

- Ngắt reset có mức ưu tiên cao nhất, khi reset xảy ra tất cả các ngắt khác và chương trình đều bị dừng và vi điều khiển trở về chế độ khởi động ban đầu.
- Ngắt mức 1, chỉ có reset mới có thể cấm ngắt này
- Ngắt mức 0, các ngắt mức 1 và reset có thể cấm ngắt này.

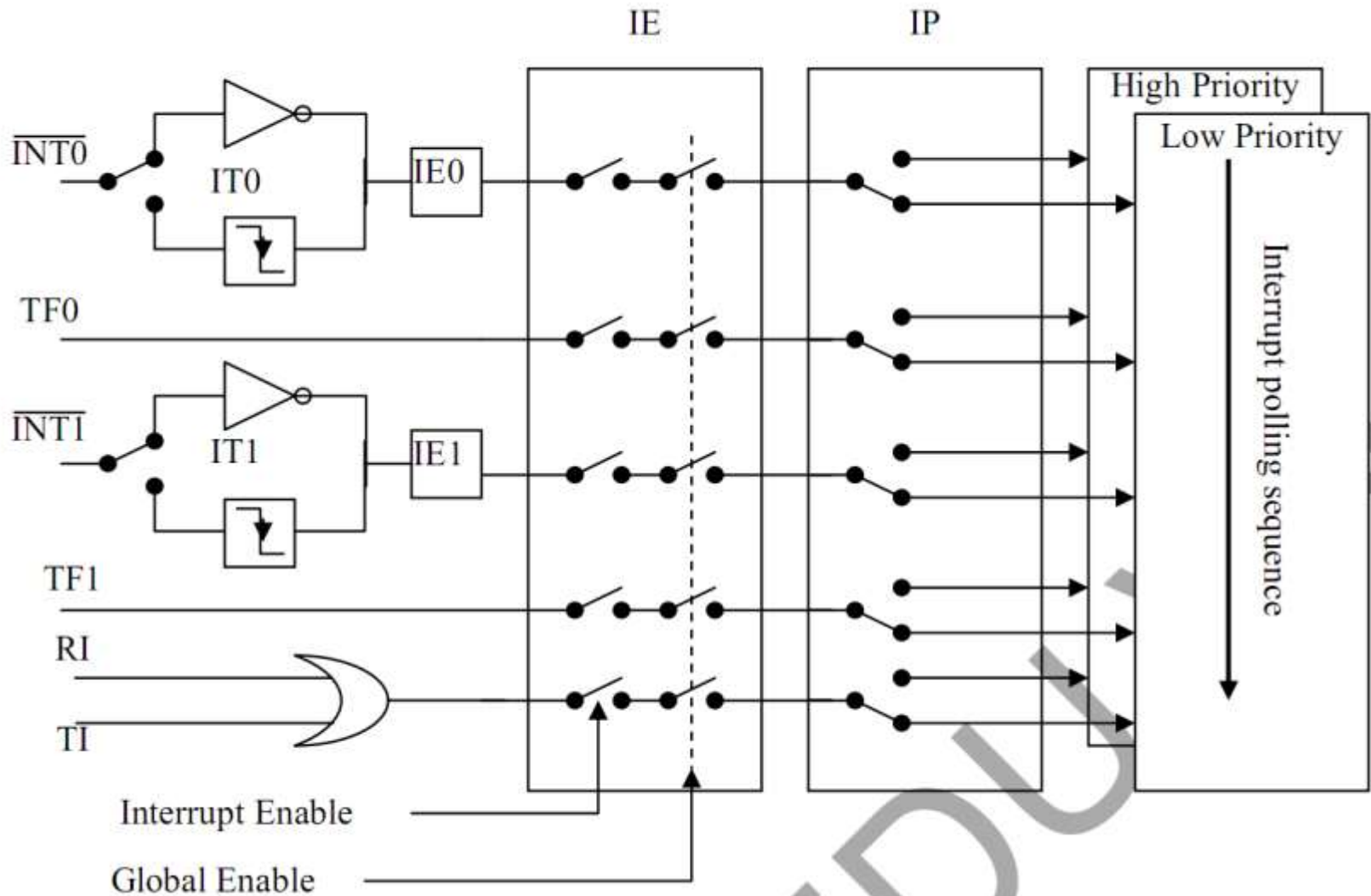


Bảng 3.25 Tóm tắt thanh ghi ưu tiên ngắt IP (Interrupt Priority) .

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.

- Nếu 1 có độ ưu tiên cao hơn một ngắt đang được xử lý xuất hiện thì, ngắt có ưu tiên thấp ngay lập tức bị dừng để ngắt kia được thực hiện.
- Nếu 1 có độ ưu tiên cao hơn một ngắt đang được xử lý xuất hiện thì, ngắt có ưu tiên thấp ngay lập tức bị dừng để ngắt kia được thực hiện
- Nếu 2 ngắt có cùng mức ưu tiên cùng yêu cầu vào 1 thời điểm thì thứ tự được chọn như sau: INTRO, Timer 0, INTR1, Timer 1, UART

Sơ đồ các interrupts



Các ngắt timer: có 2 ngắt timer

- Có địa chỉ vector ngắt là 000BH (Timer 0) và 001BH (Timer 1)
- Ngắt timer xảy ra khi TLx/THx tràn và lập cờ TFx lên 1
- Các cờ TFx tự xóa bằng phần cứng khi ISR thực hiện xong

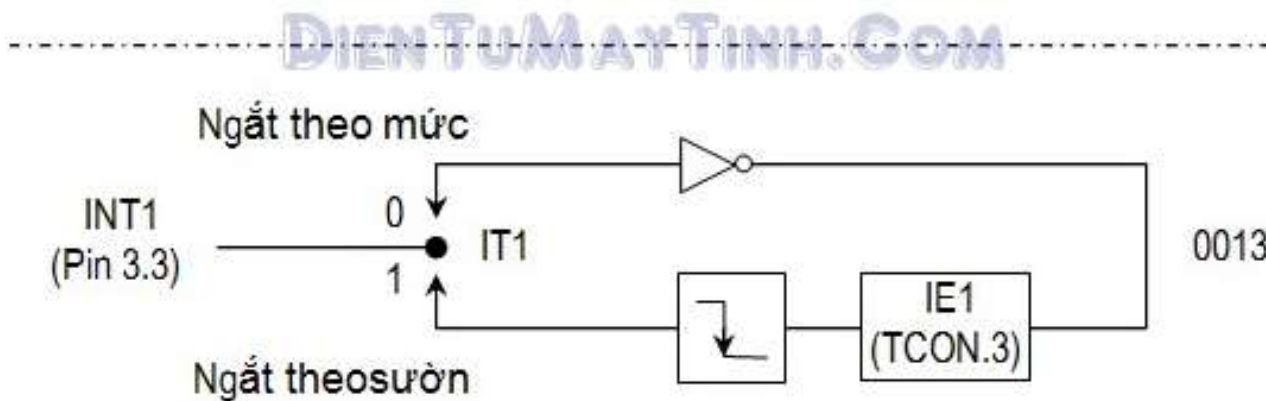
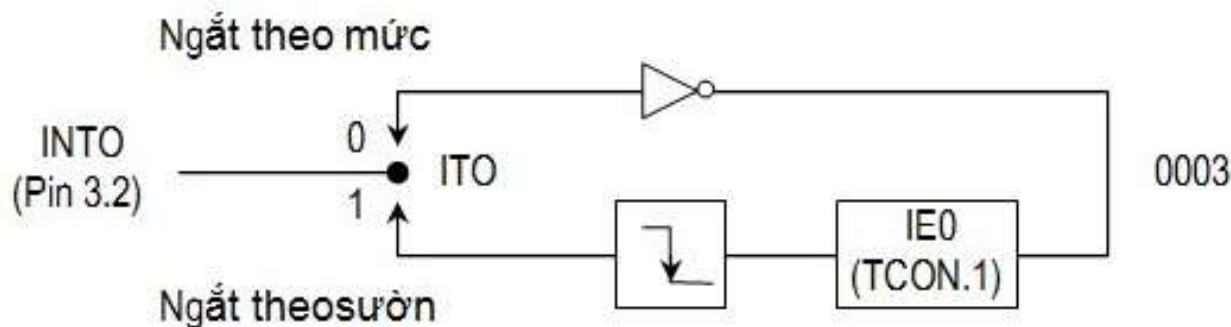


Ngắt UART: có 1 ngắt

- Có địa chỉ vector ngắt là 0023H
- Ngắt UART xảy ra khi TI hoặc RI được đặt lên 1
- Các cờ TI và RI không tự xóa bằng phần cứng

Các ngắt ngoài: có 2 ngắt ngoài

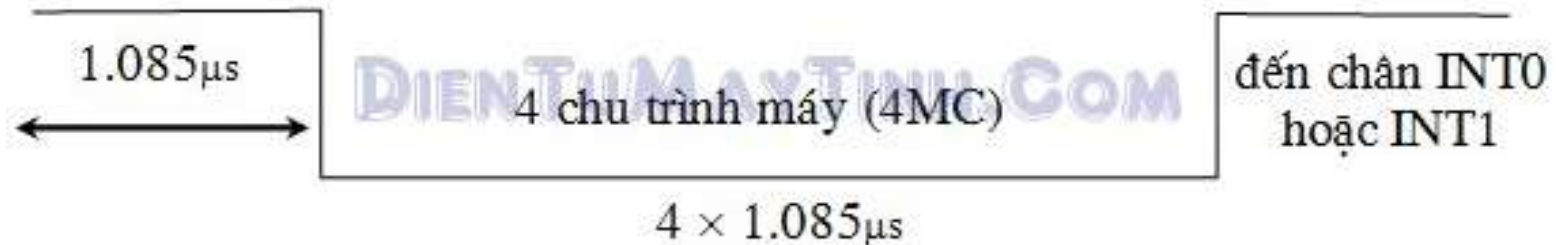
- Có địa chỉ vector ngắt là 0003H (INT 0) và 0013H (INT 1)



Các ngắt ngoài theo mức: $INT0$ và $INT1 = 0$

- $INT0$ và $INT1$ ở mức cao, nếu có 1 tín hiệu ở mức thấp thì xảy ra ngắt
- Tín hiệu mức thấp phải được thả trước khi thực hiện lệnh cuối cùng của ISR nếu không sẽ có 1 ngắt khác được tạo ra

1 chu trình máy

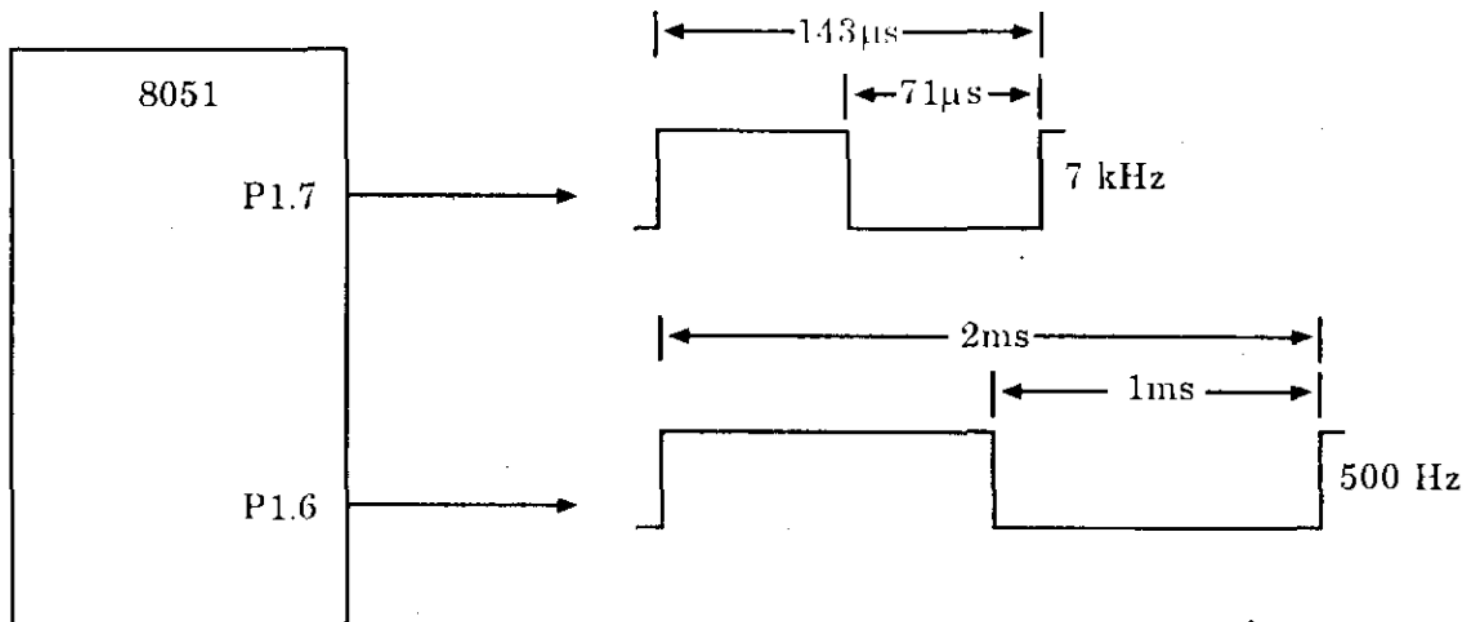


- Nguyên lý hoạt động của Timer
- Timer trong 8051 và các mode
- Nguyên lý hoạt động và các mode của UART trong 8051
- Nguyên lý hoạt động của interrupts, các loại interrup

- Hãy chỉ ra những lệnh để:
 - a) cho phép ngắt nối tiếp ngắt Timer0 và ngắt phần cứng ngoài 1 (EX1).
 - b) cấm (che) ngắt Timer0 sau đó
 - c) trình bày cách cấm tất cả mọi ngắt chỉ bằng một lệnh duy nhất.

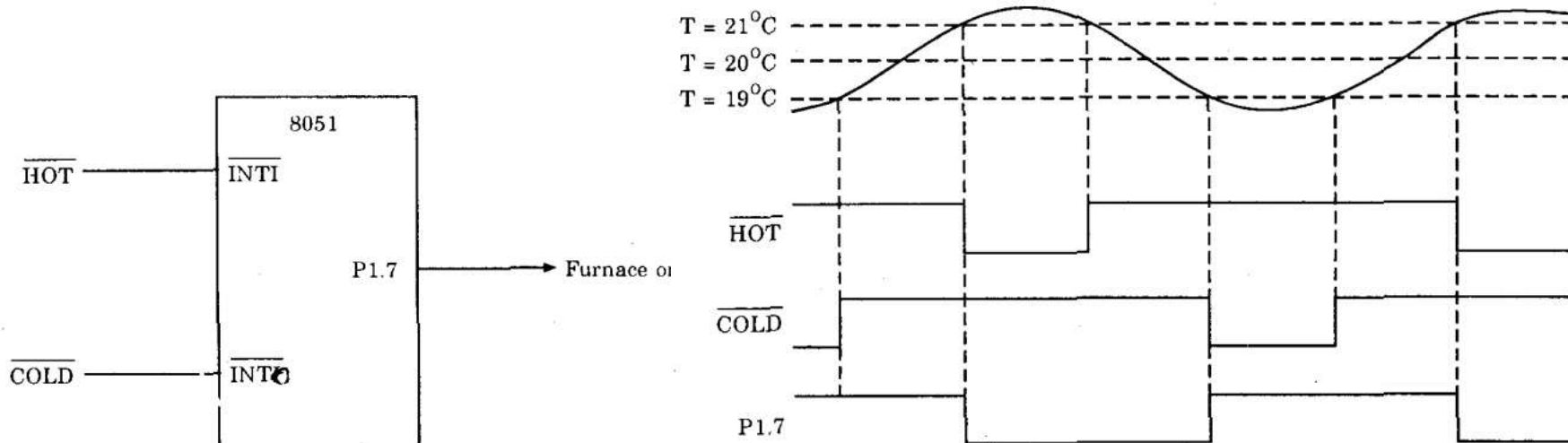
- Hãy viết chương trình nhân liên tục dữ liệu 8 bit ở cổng P0 và gửi nó đến cổng P1. Trong khi đó, nó cùng lúc tạo ra một sóng vuông chu kỳ 200us trên chân P2.1. Hãy sử dụng bộ Timer0 để tạo ra sóng vuông, tần số của 8051 là $XTAL = 11.0592MHz$.

- Hãy viết một chương trình sử dụng các ngắt để tạo đồng thời các dạng sóng vuông có tần số là 7KHz (timer 0, mode 2) và 500Hz (timer 1, mode 1) trên các chân P1.7 và P1.6



- Hãy viết 1 chương trình sử dụng các ngắt để liên tục phát đi tập mã ASCII đến một thiết bị khác qua cổng nối tiếp của 8051 (dùng UART mode 1, tốc độ baud 1200, thạch anh 12MHz)

- Hãy viết chương trình vi điều khiển sử dụng các ngắt để thiết kế bộ điều khiển lò nung sao cho nhiệt độ duy trì ở mức $200^{\circ}\text{C} \pm 10^{\circ}\text{C}$



- Hãy viết chương trình sử dụng các ngắt để thiết kế một hệ thống báo động tạo ra âm hiệu 600Hz trong 500 mili giây (sử dụng 1 loa nối với chân P3.1) (dùng timer 0 mode 3) đồng thời gửi ký tự "open" qua cổng UART (mode 1, tốc độ baud 4800) mỗi khi bộ cảm biến đặt ở cửa ra vào (được nối với chân INT1') tạo ra một chuyển trạng thái từ mức cao xuống mức thấp (thạch anh 12MHZ)

Kết thúc chương 2-4

