

TRƯỜNG CAO ĐẲNG LÝ TỰ TRỌNG TP.HCM

KHOA CÔNG NGHỆ ĐIỆN – ĐIỆN TỬ

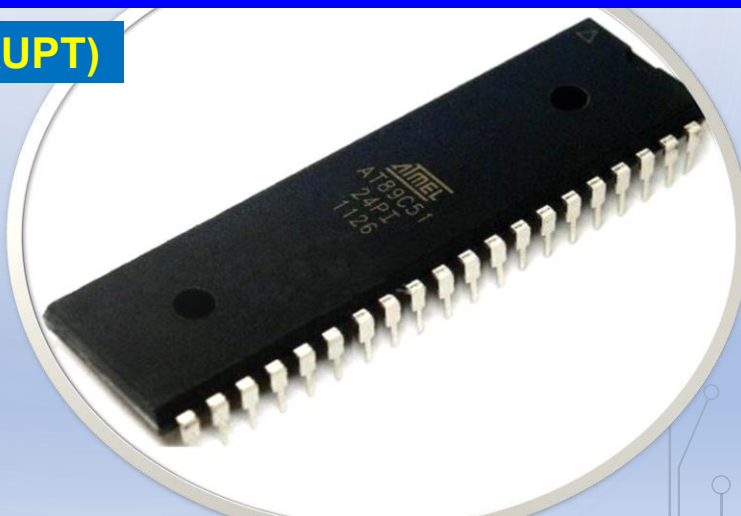


MÔN: VI XỬ LÝ

GV: TRẦN GIANG NAM

CHƯƠNG 3: CÁC CHỨC NĂNG ĐẶC BIỆT

3.2. NGẮT (INTERRUPT)



GIỚI THIỆU NỘI DUNG: NGẮT (INTERRUPT)



3.2.1. Tổ chức ngắt

3.2.2. Xử lý ngắt

3.2.3. Thiết kế chương trình dùng các ngắt

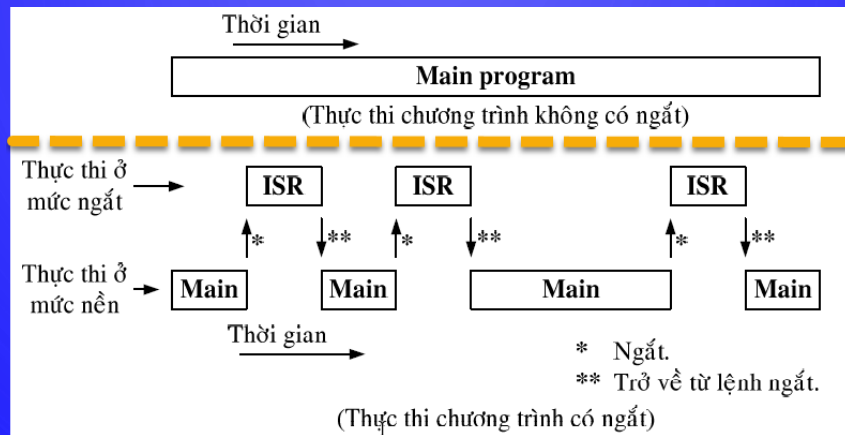
3.2.4. Chương trình ứng dụng ngắt

❑ KHÁI NIỆM NGẮT

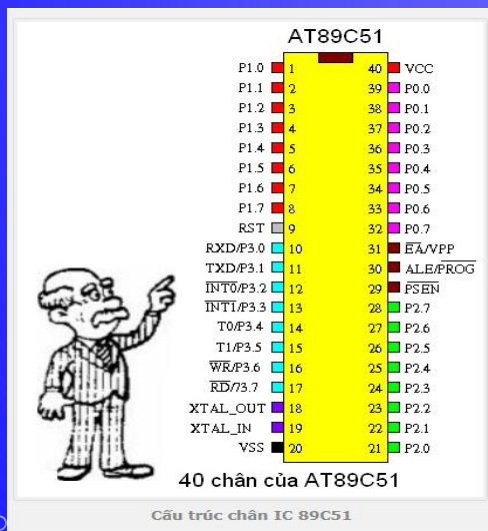
CON NGƯỜI	VI ĐIỀU KHIỂN
Đang xem youtube	Đang thực hiện công việc gì đó
Điện thoại gọi đến	Sự kiện ngắt xuất hiện (ngắt ngoài, timer tràn, uart....)
1. Dừng xem youtube	1. Dừng công việc đang làm
2. Nghe điện thoại	2. Làm công việc tương ứng với sự kiện ngắt (trình phục vụ ngắt)
3. Quay lại coi youtube tiếp	4. Quay lại thực hiện công việc đang làm trước khi ngắt xuất hiện

➤ Ngắt (Interrupt) là việc xảy ra một điều kiện (một sự kiện) làm cho chương trình đang thực thi (chương trình chính) bị tạm dừng để quay sang thực thi một chương trình khác (chương trình xử lý ngắt) rồi sau đó quay trở về thực thi tiếp chương trình đang bị tạm dừng.

BIỂU DIỄN VIỆC THỰC THI CHƯƠNG TRÌNH



3.2.1. Tổ chức ngắt



□ Các nguồn ngắt:

Ngắt do tín hiệu ngoài 0

Ngắt do tín hiệu ngoài 1

Ngắt do Timer 0

Ngắt do Timer 1

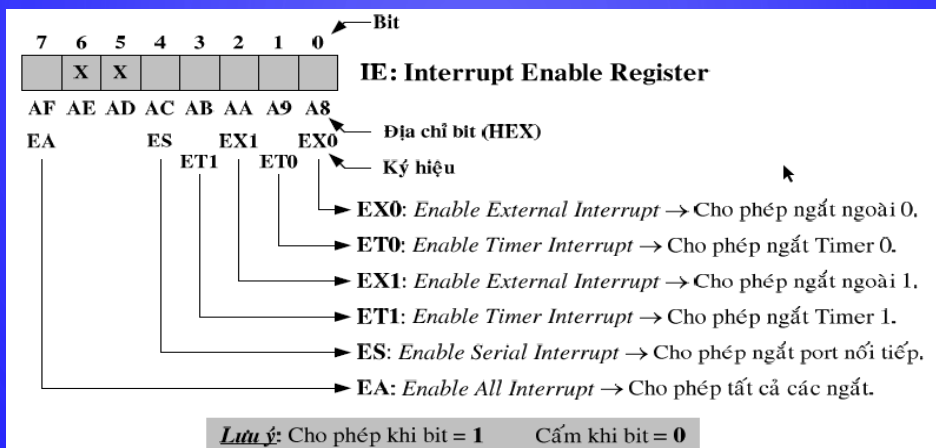
Ngắt do port nối tiếp

□ Các cờ ngắt của 8051

<i>Loại ngắt</i>		<i>Cờ ngắt</i>		<i>Vị trí của bit trong các thanh ghi</i>
Ngắt ngoài 0	→	IE0	→	TCON.1
Ngắt ngoài 1	→	IE1	→	TCON.3
Ngắt Timer 1	→	TF1	→	TCON.7
Ngắt Timer 0	→	TF0	→	TCON.5
Ngắt port nối tiếp	→	RI	→	SCON.0
Ngắt port nối tiếp	→	TI	→	SCON.1

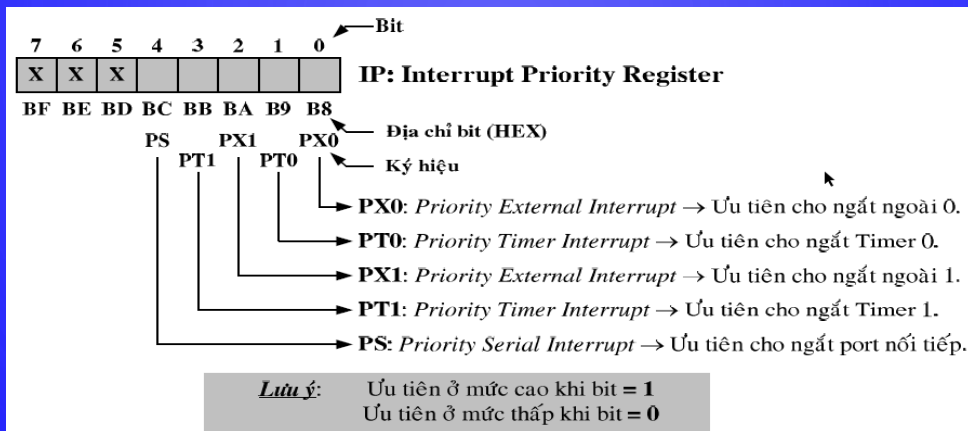
- Một ngắt xảy ra thì cờ ngắt tương ứng sẽ được set bằng 1
- Khi ISR của ngắt được thực thi thì cờ ngắt tương ứng sẽ tự động bị xóa về 0 bằng phần cứng (ngoại trừ cờ ngắt RI và TI phải được xóa về 0 bằng phần mềm).

□ Thanh ghi cho phép ngắt (IE):



- Thanh ghi cho phép ngắt (IE: Interrupt Enable): chứa các bit dùng để cho phép hoặc cấm các ngắt hoạt động

❑ Thanh ghi ưu tiên ngắt (IP):



- Thanh ghi ưu tiên ngắt (IP: Interrupt Priority): chứa các bit dùng để thiết lập mức độ ưu tiên cho từng ngắt riêng rẽ.

- Khái niệm ưu tiên ngắt giúp 8051 giải quyết vấn đề hai tín hiệu ngắt xuất hiện đồng thời và vấn đề một tín hiệu ngắt xuất hiện trong khi một ngắt khác đang được thực thi.

Câu hỏi:

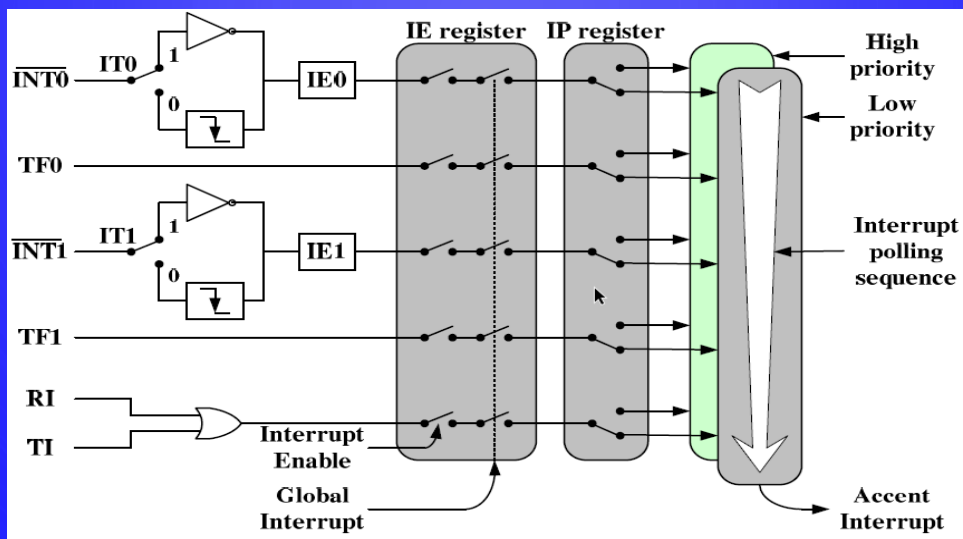
Nếu có hai hay nhiều tín hiệu ngắt có mức ưu tiên giống nhau xuất hiện đồng thời, khi đó 8051 sẽ xử lý ra sao ?

➡ **Chuỗi vòng ngắt theo thứ tự**

❑ Thứ tự chuỗi vòng ngắt (Interrupt Polling Sequence):



CẤU TRÚC NGẮT CỦA 8051:



3.2.2. Xử lý ngắt

Quy trình xử lý ngắt:

- ✓ Hoàn tất việc thực thi lệnh hiện hành.
- ✓ Giá trị của bộ đếm chương trình PC được cất giữ vào Stack (chính là địa chỉ của lệnh tiếp theo trong chương trình chính).
- ✓ Trạng thái ngắt hiện hành được lưu giữ lại.
- ✓ Các yêu cầu ngắt sẽ bị ngăn lại.
- ✓ Địa chỉ của chương trình phục vụ ngắt tương ứng sẽ được nạp vào bộ đếm chương trình PC.
- ✓ Bắt đầu thực hiện chương trình phục vụ ngắt ISR.

Trong chương trình phục vụ ngắt luôn kết thúc bằng lệnh *RETI*. Khi gặp lệnh *RETI* thì CPU sẽ lấy lại địa chỉ của lệnh tiếp theo trong ngăn xếp trả lại cho thanh ghi PC để tiếp tục thực hiện các công việc tiếp theo của chương trình chính.

□ Các vector ngắt:

- Vector ngắt là địa chỉ bắt đầu của chương trình phục vụ ngắt (ISR) của ngắt tương ứng.

<i>Loại ngắt</i>		<i>Cờ ngắt</i>		<i>Địa chỉ của vector ngắt</i>
Reset hệ thống	→	RST	→	0000H
Ngắt ngoài 0	→	IE0	→	0003H
Ngắt Timer 0	→	IT0	→	000BH
Ngắt ngoài 1	→	IE1	→	0013H
Ngắt Timer 1	→	IT1	→	001BH
Ngắt port nối tiếp	→	RI hoặc TI	→	0023H

3.2.3. Thiết kế chương trình dùng các ngắt

❑ Thiết kế các chương trình ISR kích thước nhỏ:

- Điều kiện: Khi ISR có kích thước không quá 8 byte.
- ISR được viết trong phạm vi điểm nhập tương ứng của nó trong bộ nhớ chương trình

```

ORG    0000H
LJMP   MAIN
ORG    000BH
.....
.....
ORG    0013H
.....
.....
ORG    0030H
MAIN:  .....
.....
.....
END

```

❑ Thiết kế các chương trình ISR kích thước lớn:

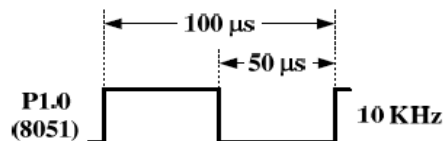
- Điều kiện: Khi ISR có kích thước vượt quá 8 byte.
- ISR không thể viết vào điểm nhập tương ứng của nó trong bộ nhớ chương trình (vì kích thước điểm nhập chỉ có 8 byte) vậy nên phải chuyển ISR này đến một nơi khác trong bộ nhớ chương trình hoặc có thể viết lần qua điểm nhập của ISR kế tiếp (nếu ISR đó không sử dụng).


```

ORG 0000H ; Điểm nhập của reset hệ thống
LJMP MAIN
ORG 000BH ; Điểm nhập cho ISR của Timer 0
LJMP T0ISR
ORG 0013H ; Điểm nhập cho ISR của ngắt ngoài 1
LJMP EX1ISR
ORG 0030H ; Điểm nhập của chương trình chính
MAIN: ..... ; Chương trình chính bắt đầu
      .....
      .....
T0ISR: ..... ; ISR của ngắt Timer 0
      .....
      RETI
EX1ISR:..... ; ISR của ngắt ngoài 1
      .....
      RETI
END

```

Ví dụ 1: Viết chương trình sử dụng Timer 0 và các ngắt để tạo ra một sóng vuông có tần số 10 KHz trên chân P1.0, $f_{Osc} = 12\text{MHz}$.



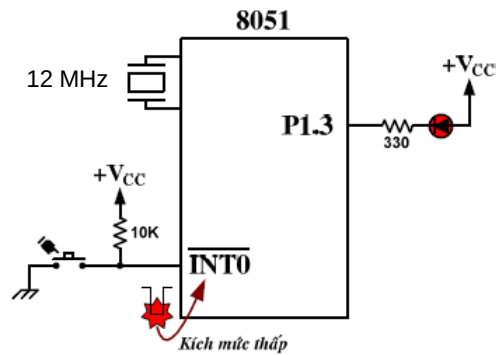
Giải:

```

ORG 0000H ;Điểm nhập reset.
LJMP MAIN ;Nhảy qua khỏi các vector ngắt.
ORG 000BH ;Điểm nhập ISR của Timer 0.
T0ISR:    ;ISR của Timer 0.
      CPL P1.0 ;Lấy bù.
      RETI ;Kết thúc ISR của Timer 0.
ORG 0030H ;Điểm nhập chương trình chính.
MAIN:    ;Chương trình chính bắt đầu.
      MOV TMOD, #02H ;Chọn chế độ 2 cho Timer 0.
      MOV TH0, #(-50) ;Định thời 50μs.
      SETB TR0 ;Cho Timer 0 hoạt động.
      MOV IE, #82H ;Cho phép ngắt Timer 0 .
      SJMP $ ;Không làm gì (nhảy tại chỗ).
      END ;Kết thúc chương trình.

```

Ví dụ 2 Ngắt ngoài kích khởi mức thấp - Cho mạch điện như hình vẽ. Chân INT0\ được nối với một công tắc bình thường ở mức cao, mỗi khi chân này có mức thấp (nhấn công tắc) thì điều khiển bật LED (bình thường thì LED tắt). Khi LED đã được bật thì phải sáng trong một khoảng thời gian (vài trăm μs) trước khi tắt, khi công tắc được nhấn và giữ thì LED phải sáng liên tục.



Giải:

ORG	0000H	;Điểm nhập reset.
LJMP	MAIN	;Nhảy qua khởi các vector ngắt.
ORG	0003H	;Điểm nhập ISR ngắt ngoài 0.
CLR	P1.3	;Bật LED.
MOV	R3, #255	;Thời gian LED duy trì trạng thái sáng là
DJNZ	R3, \$	;255.T _{Machine} .
SETB	P1.3	;Tắt LED.
RETI		;Kết thúc ISR của ngắt ngoài 0.
ORG	0030H	;Điểm nhập chương trình chính.
MAIN:		;Chương trình chính bắt đầu.
CLR	TCON.0	;Ngắt ngoài 0 kích khởi mức thấp.
MOV	IE, #81H	;Cho phép ngắt ngoài 0.
SJMP	\$	;Không làm gì (nhảy tại chỗ).
END		;Kết thúc chương trình

ORG	0000H	;Điểm nhập reset.
LJMP	MAIN	;Nhảy qua khỏi các vectơ ngắt.
ORG	0003H	;Điểm nhập ISR ngắt ngoài 0.
CLR	P1.3	;Bật LED.
MOV	R3, #255	;Thời gian LED duy trì trạng thái sáng là
DJNZ	R3, \$	;255.T _{Machine} .
SETB	P1.3	;Tắt LED.
RETI		;Kết thúc ISR của ngắt ngoài 0.
ORG	0030H	;Điểm nhập chương trình chính.
MAIN:		;Chương trình chính bắt đầu.
SETB	TCON.0	;Ngắt ngoài 0 kích khởi cạnh âm.
MOV	IE, #81H	;Cho phép ngắt ngoài 0.
SJMP	\$	;Không làm gì (<i>nhảy tại chỗ</i>).
END		;Kết thúc chương trình