



Chương 5

IO VÀ NGẮT



Giới thiệu các phương pháp vào ra dữ liệu

- Vai trò của vào ra dữ liệu
 - Là phương tiện giúp CPU giao tiếp với thế giới bên ngoài
 - Cung cấp dữ liệu đầu vào cho CPU xử lý
 - Cung cấp phương tiện để CPU kết xuất dữ liệu đầu ra
- Các phương pháp vào ra chính:
 - Thăm dò (polling)
 - Ngắt (Interrupt)
 - Truy nhập trực tiếp bộ nhớ (DMA-Direct Memory Access)



- **Các cổng vào ra của máy tính**
 - PS/2: cổng ghép nối với bàn phím và chuột
 - COM: các cổng ghép nối nối tiếp
 - LPT: các cổng ghép nối song song
 - IDE, SATA, SCSI: các cổng ghép nối ổ đĩa
 - LAN: cổng ghép nối mạng cục bộ
 - Audio: cổng ghép nối âm thanh (speaker, mic và line-in)
 - Video: Cổng ghép nối với màn hình (tương tự)
 - DVI : Cổng ghép nối với màn hình (số)
 - USB: Cổng ghép nối theo chuẩn USB
 - USB 1.0: 12Mb/s
 - USB 2.0: 480Mb/s
 - USB 3.0: 1.5Gb/s (tương lai)



Vào ra bằng thăm dò

- **Cơ chế vào ra bằng thăm dò:**
 - CPU tổ chức một thanh ghi lưu trạng thái sẵn sàng làm việc của các thiết bị vào ra;
 - Mỗi bit của thanh ghi trạng thái được gán cho một thiết bị;
 - Các thiết bị định kỳ cập nhật trạng thái sẵn sàng làm việc của mình lên bit tương ứng;
 - CPU định kỳ lần lượt “quét” các bit trạng thái vào ra;
 - Nếu gặp một thiết bị sẵn sàng làm việc, 2 bên tiến hành trao đổi dữ liệu;
 - Trao đổi dữ liệu xong, CPU tiếp tục quét thiết bị khác.
 - CPU là bên chủ động trong quá trình trao đổi dữ liệu

■ Ưu điểm:

- Đơn giản, dễ cài đặt
- Có thể được cài đặt bằng phần mềm

■ Nhược điểm:

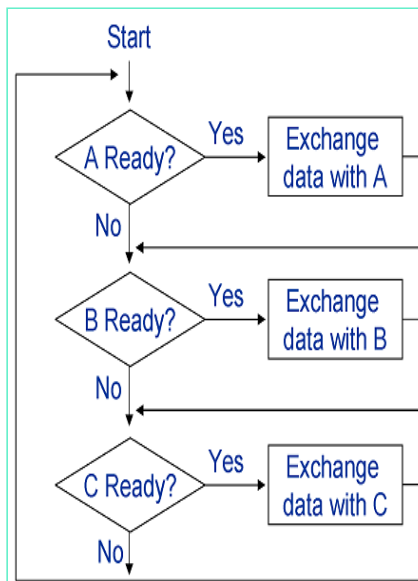
- Hiệu quả thấp do CPU tốn nhiều thời gian để thăm dò các thiết bị
- Không thực sự khả thi khi có nhiều thiết bị trong danh sách thăm dò

■ Ứng dụng của vào ra bằng thăm dò:

- Thăm dò thường được sử dụng khi hệ thống khởi động: CPU thăm dò hầu hết các t/bị để xác lập c/hình
- Thăm dò được sử dụng trong quá trình hoạt động với các thiết bị rời (removable) như ổ CD/DVD, ổ mềm,...

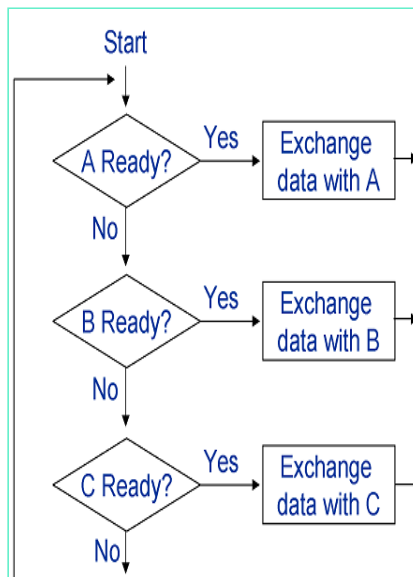
Vào ra bằng thăm dò – không ưu tiên

- Ba thiết bị A, B, C được thăm dò không ưu tiên
- CPU quét tất cả các thiết bị trong một chu trình thăm dò
- CPU có thể trao đổi dữ liệu với nhiều hơn 1 thiết bị trong một chu trình thăm dò
- Các thiết bị được “thăm” lần lượt, không phụ thuộc vào thiết bị đứng trước chu trình.
- CPU bắt đầu 1 chu trình thăm dò mới sau khi đã quét qua tất cả các thiết bị.



Vào ra bằng thăm dò – có ưu tiên

- Ba thiết bị A, B, C được thăm dò có ưu tiên theo thứ tự: A, B, C;
- CPU có thể quét tất cả các thiết bị trong một chu trình thăm dò
- CPU chỉ trao đổi dữ liệu với tối đa 1 thiết bị trong một chu trình thăm dò
- Các thiết bị có mức ưu tiên cao luôn được thăm trước;
- Các thiết bị có mức ưu tiên thấp chỉ được thăm nếu các thiết bị đứng trước nó không sẵn sàng.
- CPU bắt đầu 1 chu trình thăm dò mới ngay sau khi trao đổi dữ liệu với một thiết bị.



Ngắt và xử lý ngắt

- Ngắt là gì?
 - Ngắt (Interrupt) là một sự kiện mà CPU tạm dừng thực hiện một chương trình để thực hiện một đoạn chương trình khác theo yêu cầu từ bên ngoài;
 - Thông thường các yêu cầu từ bên ngoài thường xuất phát từ các thiết bị vào ra. Các yêu cầu này gọi là các yêu cầu ngắt;
 - Đoạn chương trình CPU thực hiện trong thời gian ngắt được gọi là chương trình con phục vụ ngắt (CTCPVN).

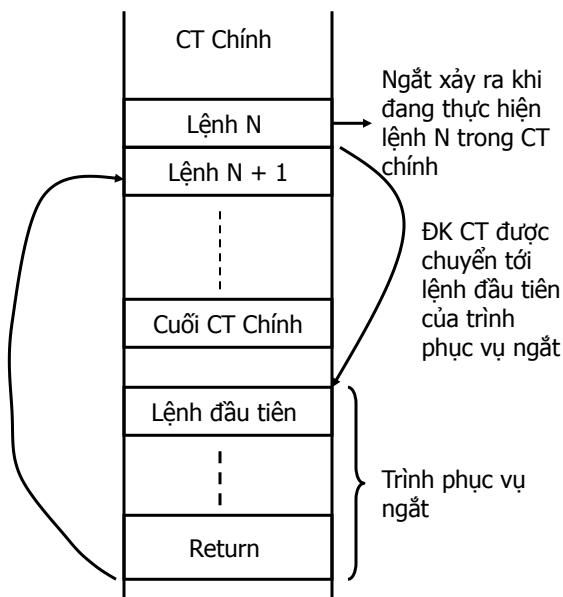


- Các CTCPVN là các đoạn chương trình:
 - Được viết sẵn và lưu trong ROM;
 - Mỗi CTCPVN có nhiệm vụ riêng và thường là đảm nhiệm việc trao đổi dữ liệu với thiết bị vào ra.
- Khi nào CPU kiểm tra và xử lý ngắt: CPU kiểm tra yêu cầu ngắt tại chu kỳ đồng hồ cuối cùng của chu kỳ lệnh.



Ví dụ về ngắt

Trở về chương trình chính (CT Điều khiển) từ cuối của CT phục vụ ngắt tới lệnh N+1 của chương trình chính





■ Phân loại ngắt :

- Ngắt cứng: là các ngắt được kích hoạt bởi các bộ phận phần cứng gửi đến chân NMI và INTR của CPU; gồm:
 - Ngắt không che được NMI (Non-Maskable Interrupt): ngắt gửi đến chân NMI của CPU, không chịu sự ảnh hưởng của cờ ngắt; VD: ngắt Reset;
 - Ngắt che được INTR (Maskable Interrupt): ngắt gửi đến chân INTR của CPU, chịu sự chi phối của cờ ngắt; Cờ $IF=1 \rightarrow$ cho phép ngắt, $IF=0 \rightarrow$ cấm ngắt.

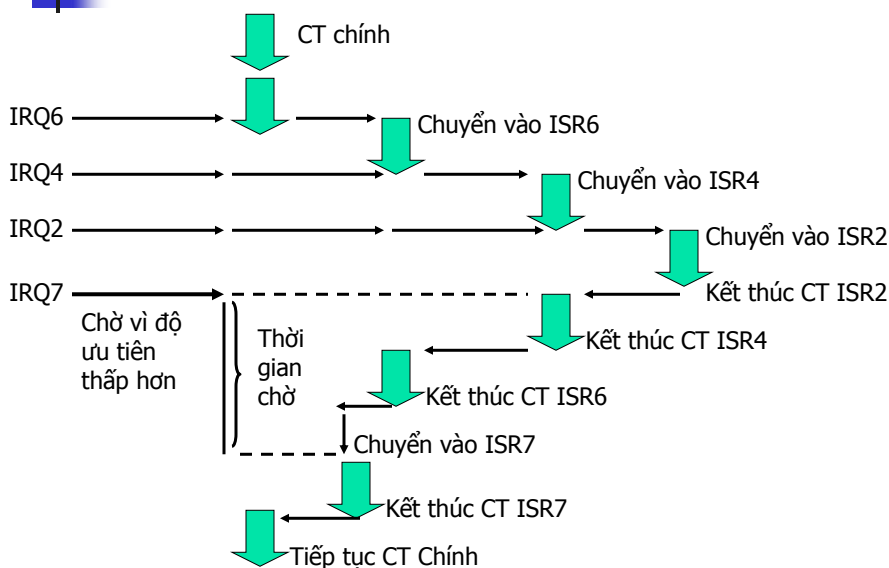



- Ngắt mềm: là các ngắt được kích hoạt bởi các chương trình thông qua lệnh gọi ngắt $INT <N>$. N là số hiệu ngắt, $N=0-255$.
- Các ngắt ngoại lệ: là các ngắt do các lỗi nảy sinh trong quá trình hoạt động của CPU:
 - Ngắt chia cho 0 (divide by zero)
 - Ngắt do tràn (overflow)

Nguyên tắc hoạt động các ngắt


- Các ngắt được phục vụ trên cơ sở độ ưu tiên
- Mỗi ngắt được gán 1 con số (loại ngắt) : số nhỏ ưu tiên cao, số lớn ưu tiên thấp
- Khi 1 chương trình phục vụ ngắt (loại n) đang được phục vụ thì chỉ có những ngắt có độ ưu tiên cao hơn (loại $< n$) mới được phép ngắt chương trình phục vụ ngắt đang hoạt động.
- Các thiết bị có độ ưu tiên thấp hơn phải chờ cho đến khi trình phục vụ ngắt hiện thời hoàn tất trước khi yêu cầu của nó được xem xét.

Ví dụ về ưu tiên ngắt





- Mức ưu tiên các yêu cầu ngắt (từ cao nhất đến thấp nhất)
 - 1. Ngắt nội bộ: INT 0 (chia cho 0), INT N ($N \neq 0$)
 - 2. Ngắt không che được NMI
 - 3. Ngắt che được INTR
 - 4. Ngắt chạy từng lệnh: INT 1
- Vi xử lý 8086/8088 có 256 ngắt được đánh số từ 0-255
- Một vector ngắt gồm các thông tin:
 - Số hiệu ngắt N, $N=0-255$ hoặc 00-FFH
 - Địa chỉ đầy đủ chương trình con phục vụ ngắt (CTCPVN) lưu trong bộ nhớ ROM. Địa chỉ đầy đủ gồm:
 - Địa chỉ đoạn (segment) (CS)
 - Địa chỉ lệch (offset) (IP)



Bảng Vector Ngắt

- Mỗi ngắt → có :
 - 1 chương trình con phục vụ ngắt cho nó, lưu trong bộ nhớ
 - Số hiệu ngắt
- Khi gọi ngắt → gọi thông qua con số → làm thế nào thông qua con số này để tìm tới vị trí đầu lưu chương trình phục vụ ngắt
- Thực hiện thông qua bảng vector ngắt :
- Là 1 bảng con trỏ địa chỉ (address pointer table) được dùng để liên kết (tìm tới) loại ngắt với vị trí của chương trình phục vụ ngắt trong bộ nhớ CT



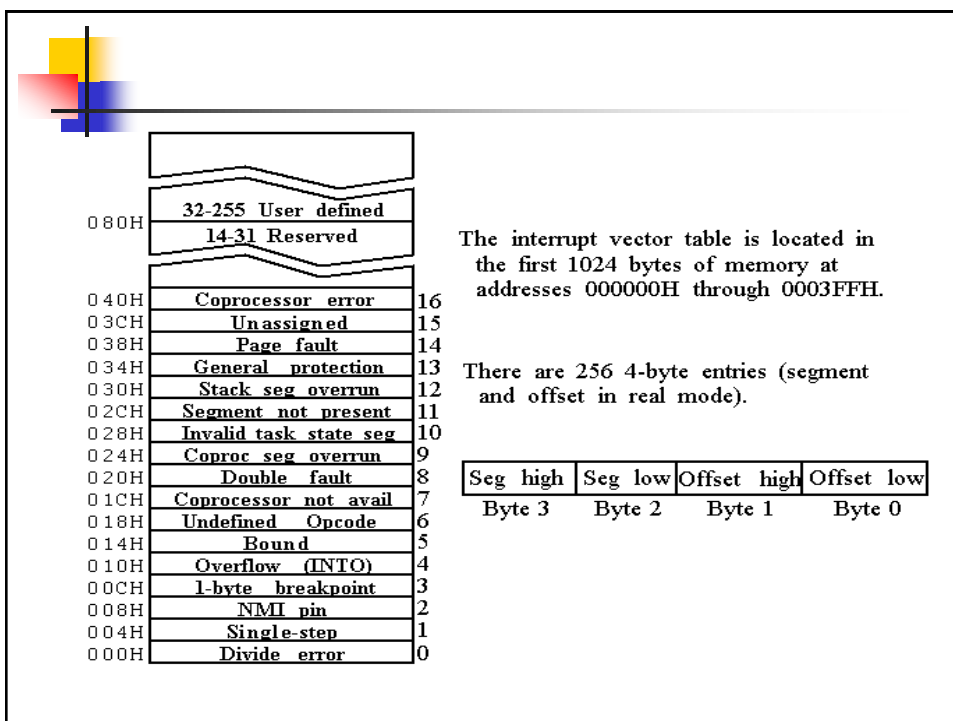
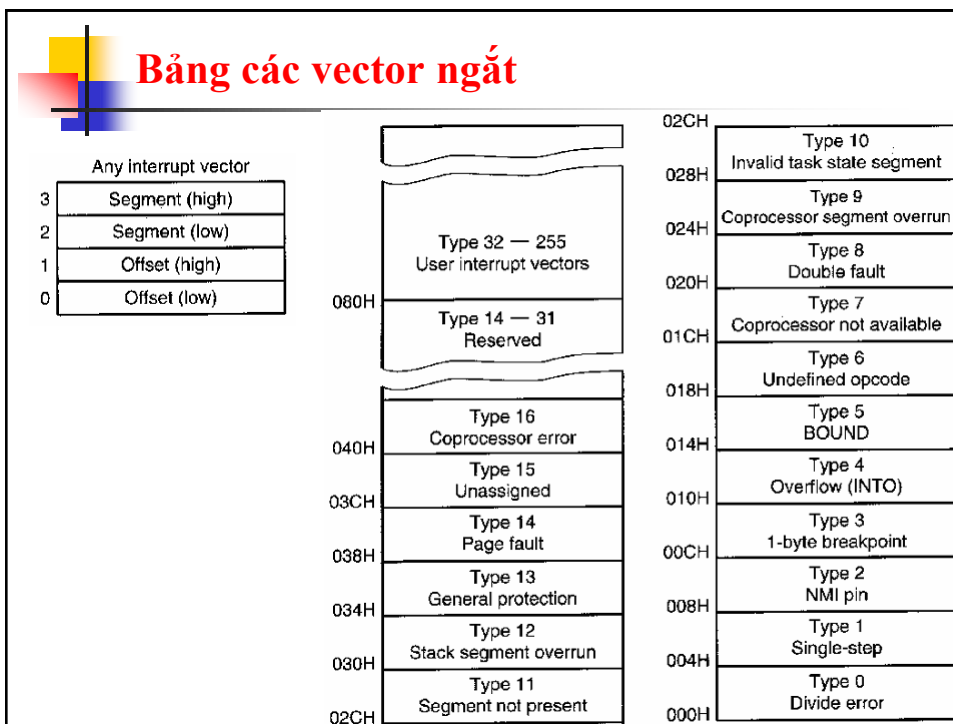
Bảng vector ngắt

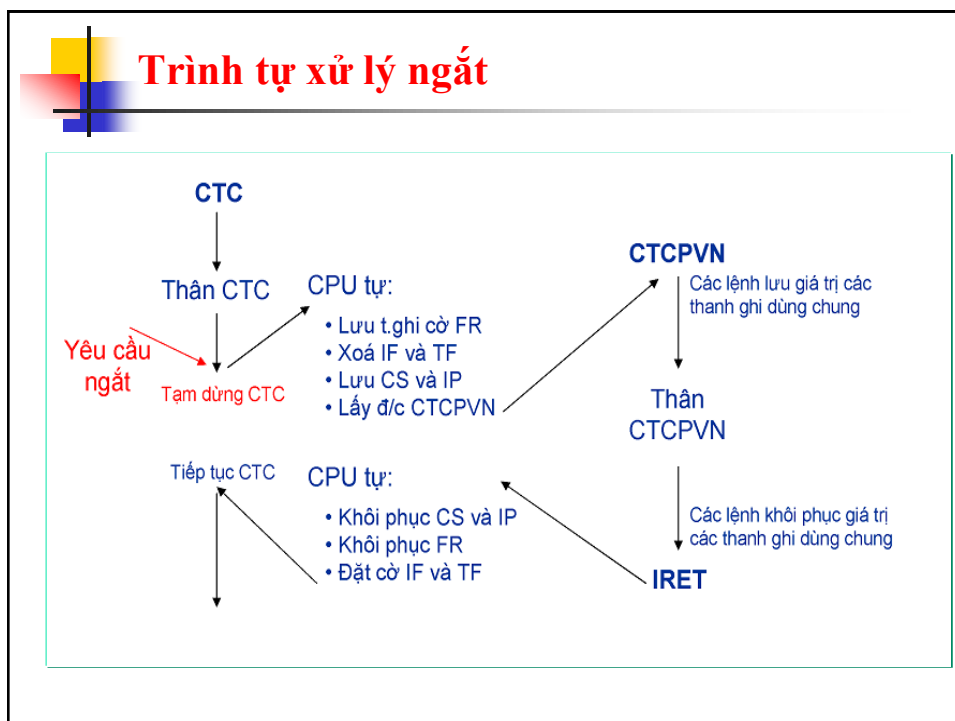
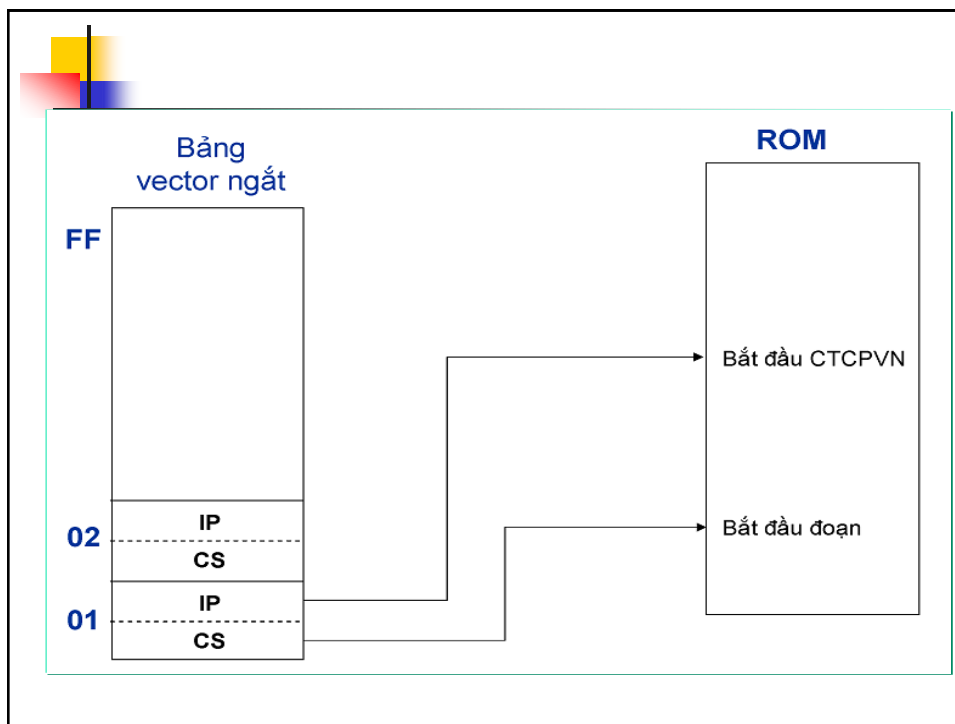
- Nội dung của bảng này có thể được lưu trong ROM, hoặc nạp vào RAM như 1 phần của chương trình khởi động hệ thống.
 - Bảng được đặt ở địa chỉ : 00000 → 003FF (1KB trong bộ nhớ)
 - Chứa 256 con trỏ địa chỉ (vector) từ 0 → 255 mô tả cho 256 ngắt từ 0 → 255
 - Mỗi con trỏ rộng 4 byte (2 word) và luôn đặt ở địa chỉ chẵn.
 - Word ở địa chỉ cao của vector gọi là base-address : xác định địa chỉ segment của bộ nhớ CT lưu trình phục vụ ngắt → nó được nạp vào thanh ghi CS



Bảng vector ngắt

- Word ở địa chỉ thấp chứa địa chỉ offset của lệnh đầu tiên của chương trình phục vụ ngắt tính từ đầu segment → được nạp vào IP
- 5 vector đầu tiên dùng cho các chức năng đặc biệt







- 1 Khi nhận được yêu cầu ngắt, CPU thực hiện các việc:
 - a. Hoàn tất lệnh đang thực hiện của chương trình chính (CTC)
 - b. Lưu giá trị của thanh ghi cờ FR vào ngăn xếp
 - c. Xóa cờ ngắt IF và cờ bẫy TF
 - d. Lưu giá trị của các t.ghi CS và IP vào ngăn xếp
 - e. Từ số hiệu ngắt N, lấy địa chỉ của CTCPVN từ bảng vector ngắt
- 2. Nạp địa chỉ của CTCPVN vào CS và IP, CPU thực hiện CPCPVN, gồm:
 - a. Lưu giá trị các thanh ghi dùng chung vào ngăn xếp
 - b. Thực hiện mã chính của CTCPVN
 - c. Khôi phục giá trị các thanh ghi dùng chung



- 3. Gặp lệnh IRET kết thúc CTCPVN, CPU thực hiện các việc:
 - a. Khôi phục giá trị của CS và IP
 - b. Khôi phục giá trị của thanh ghi cờ FR
 - c. Đặt cờ ngắt IF và cờ bẫy TF
- 4. CPU tiếp tục thực hiện lệnh tiếp theo của CTC (nằm sau lệnh xảy ra ngắt).



Ví dụ

- Tính địa chỉ cho CS₅₀ và IP₅₀ trong bộ nhớ
- Giải :
 - Mỗi vector yêu cầu 4 byte liên tục
 - Gọi n là số vector (n=50)
 - Địa chỉ đầu của vector = $n*4$
 - IP lưu địa chỉ = nội dung 2 byte ($n*4$ và $n*4+1$)
 - CS lưu đ/c = nội dung của 2 byte $n*4+2$ và $n*4+3$
 - Vậy địa chỉ đầu vector 50 là $50*4=200=C8_{16}$
 - IP₅₀ lưu nội dung ở địa chỉ 000C8h và 000C9h
 - CS₅₀ lưu nội dung ở địa chỉ 000CAh và 000CBh



Vào ra bằng ngắt – chu trình vào ra bằng ngắt

- 1. Thiết bị vào ra có nhu cầu trao đổi dữ liệu, gửi yêu cầu ngắt đến chân tín hiệu INTR của CPU;
- 2. Khi nhận được yêu cầu ngắt, CPU thực hiện các việc:
 - a. Hoàn tất lệnh đang thực hiện của chương trình chính (CTC)
 - b. Lưu giá trị của thanh ghi cờ FR vào ngăn xếp
 - c. Xóa cờ ngắt IF và cờ bẫy TF
 - d. Lưu giá trị của các t.ghi CS và IP vào ngăn xếp
 - e. Gửi tín hiệu xác nhận ngắt đến thiết bị vào ra qua chân tín hiệu INTA



- 3. Nhận được tín hiệu xác nhận ngắt của CPU, thiết bị vào ra gửi số hiệu ngắt N đến CPU
- 4. Nhận được số hiệu ngắt N, CPU lấy địa chỉ của CTCPVN tương ứng từ bảng vector ngắt
- 5. Nạp địa chỉ của CTCPVN vào CS và IP, CPU thực hiện CPCPVN, gồm:
 - a. Lưu giá trị các thanh ghi dùng chung vào ngăn xếp
 - b. Thực hiện mã chính của CTCPVN: đồng thời thực hiện việc trao đổi dữ liệu với thiết bị vào ra
 - c. Khôi phục giá trị các thanh ghi dùng chung



- 6. Gặp lệnh IRET kết thúc CTCPVN, CPU thực hiện các việc:
 - a. Khôi phục giá trị của CS và IP
 - b. Khôi phục giá trị của thanh ghi cờ FR
 - c. Đặt cờ ngắt IF và cờ bắt TF
- 7. CPU tiếp tục thực hiện lệnh tiếp theo của CTC (nằm ngay sau lệnh xảy ra ngắt).

■ Ưu điểm

- Hiệu quả hơn vào ra bằng thăm dò, do CPU không phải thăm dò từng thiết bị

■ Nhược điểm

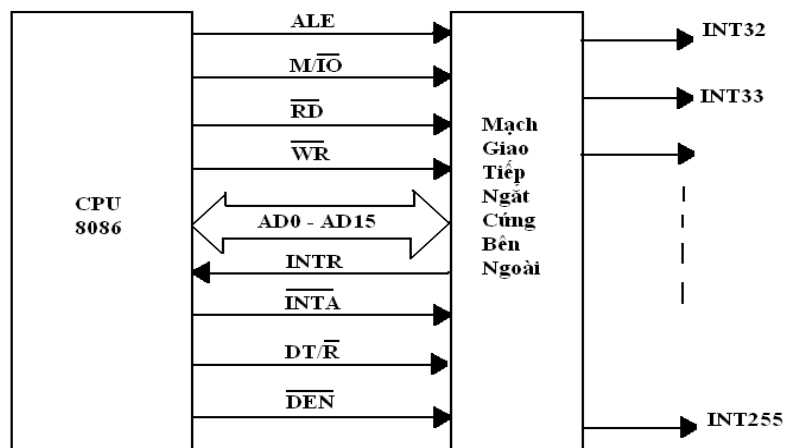
- Phức tạp hơn vào ra bằng thăm dò
- Cần mạch phần cứng để điều khiển ngắt

■ Bên chủ động trong vào ra bằng ngắt:

- Thiết bị vào ra

Các Tín Hiệu Giao Tiếp Ngắt Cứng

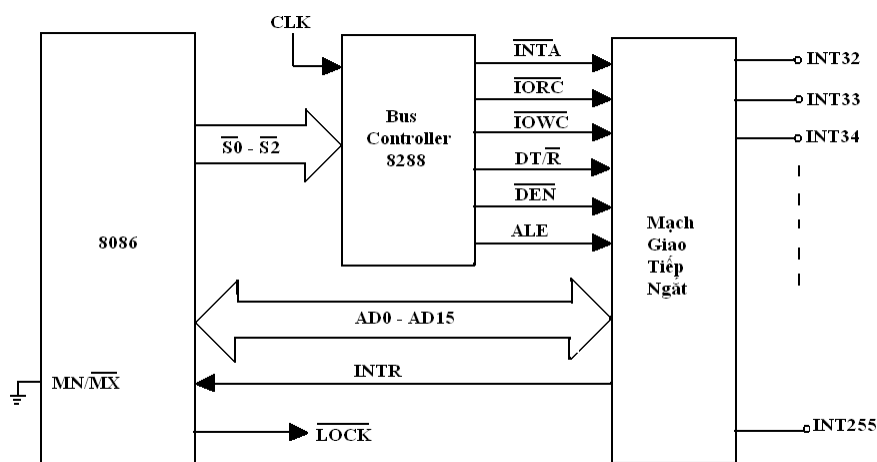
■ Chế độ Min



Các tín hiệu ngắt

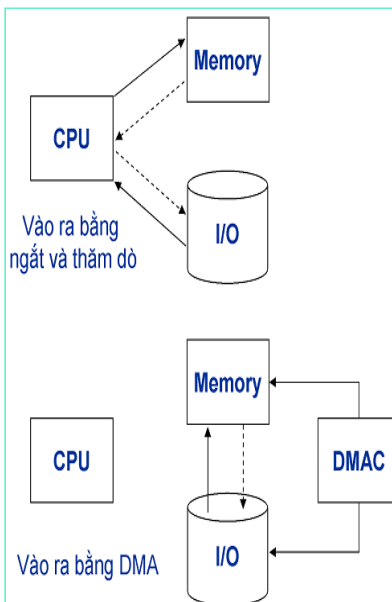
- INTR báo cho CPU biết có 1 thiết bị yêu cầu phục vụ
- 8086 lấy mẫu tín hiệu INTR ở chu kỳ cuối của xung clock trong mỗi chu kỳ lệnh để xác định có y/c ngắt ?
- Nếu chấp nhận, CPU báo cho mạch ngoài bằng INTA
- AD0 – AD7 để mạch giao tiếp truyền số hiệu ngắt 8 bit vào cho CPU, để CPU tính đ/c cho CS và IP

b. Chế độ Max



vào ra bằng DMA

- Trong các phương pháp vào ra bằng thăm dò và ngắt thiết bị vào ra trao đổi dữ liệu với bộ nhớ thông qua CPU.
- Phương pháp vào ra bằng DMA (Direct Memory Access) cho phép thiết bị vào ra trao đổi dữ liệu trực tiếp với bộ nhớ theo khối, không thông qua CPU;
- DMA thích hợp khi cần trao đổi dữ liệu với khối lượng lớn trong khoảng thời gian ngắn

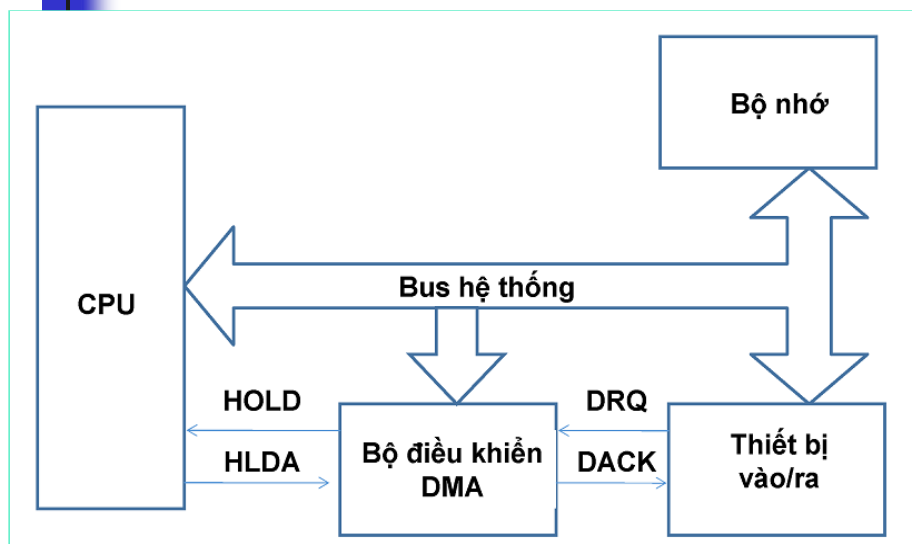


- DMAC (DMA Controller) thay mặt CPU điều khiển quá trình trao đổi dữ liệu trực tiếp giữa thiết bị vào ra và bộ nhớ;
- DMA có tốc độ cao hơn nhiều lần so với vào ra bằng thăm dò và ngắt. Ví dụ, với VXL 8088:
 - Vào ra bằng DMA mất 4 chu kỳ đồng hồ để chuyển 1 byte thiết bị ngoại vi vào bộ nhớ;
 - Vào ra thông qua CPU mất 39 chu kỳ đồng hồ để chuyển 1 byte thiết bị ngoại vi vào bộ nhớ:

	;Số chu kỳ đồng hồ
LAP: MOV AL, [SI];	10
OUT PORT, AL;	10
INC SI;	2
LOOP LAP;	17

: Công: 39 chu kỳ

Vào ra bằng DMA – hệ VXL với DMAC



Vào ra bằng DMA – Chu trình vào ra bằng DMA

- 1. Thiết bị vào ra có yêu cầu trao đổi dữ liệu gửi yêu cầu DRQ đến CPU thông qua DMAC;
- 2. DMAC chuyển yêu cầu DRQ thành HRQ và gửi đến chân tín hiệu HOLD của CPU;
- 3. Nhận được yêu cầu sử dụng bus HRQ, CPU:
 - a. Gửi các tham số điều khiển trao đổi dữ liệu và tín hiệu xác nhận yêu cầu sử dụng bus HACK cho DMAC qua chân tín hiệu HLDA;
 - b. Tự tách ra khỏi bus hệ thống (100% các tín hiệu của bus A và D và một số tín hiệu của bus C)
- 4. Nhận được HACK, DMAC chiếm quyền điều khiển bus hệ thống và gửi tín hiệu xác nhận DACK cho thiết bị vào ra;
- 5. DMAC điều khiển quá trình trao đổi dữ liệu trực tiếp giữa thiết bị vào ra và bộ nhớ;
- 6. Kết thúc quá trình DMA, DMAC trả quyền điều khiển bus cho CPU.



- Ưu điểm:

- Hiệu suất rất cao do dữ liệu được trao đổi trực tiếp theo khối giữa thiết bị vào ra và bộ nhớ không thông qua CPU

- Nhược điểm:

- Phức tạp hơn vào ra bằng thăm dò và ngắt
- Cần mạch phần cứng để điều khiển quá trình DMA

- Bên chủ động trong vào ra bằng DMA:

- Thiết bị vào ra