

# BỘ NHỚ (Memory)



---

Mục tiêu :

1. Hiểu được cấu tạo của bộ nhớ, chức năng và hoạt động của bộ nhớ.
2. Nắm được quá trình đọc bộ nhớ & ghi bộ nhớ.
3. Vai trò của bộ nhớ Cache trong máy tính.



# Bộ nhớ (**Memory**)

---

Nội dung :

1. Tổ chức bộ nhớ của máy tính IBM PC
2. Phân loại bộ nhớ : Primary Memory và Secondary Memory.
3. Quá trình CPU đọc bộ nhớ.
4. Quá trình CPU ghi bộ nhớ.
5. Bộ nhớ Cache.



# Memory

---

- Bộ nhớ (Memory) là nơi chứa chương trình và dữ liệu.
- Đơn vị đo bộ nhớ :
- Bit : đơn vị bộ nhớ nhỏ nhất là bit. Mỗi bit có thể lưu trữ 1 trong 2 trạng thái là 0 và 1.
- Byte = 8 bits, được đánh chỉ số từ 0 đến 7 bắt đầu từ phải sang trái.
- Kbyte = 1024bytes =  $2^{10}$  bytes.
- Mbyte = 1024Kbytes =  $2^{10}$  Kbytes.
- Gbyte = 1024Mbytes =  $2^{10}$  Mbytes.



# Primary Memory

---

**Còn được gọi là bộ nhớ chính hay bộ nhớ trung tâm.**

**Chia làm 2 loại : RAM và ROM**

# RAM

RAM (Random Access Memory) bộ nhớ truy xuất ngẫu nhiên. Là nơi lưu giữ các chương trình và dữ liệu khi chạy chương trình. Đặc điểm của RAM :

Cho phép đọc/ ghi dữ liệu.

Dữ liệu bị mất khi mất nguồn.

Khi máy tính khởi động, Ram rỗng. Người lập trình chủ yếu là làm việc với Ram – vùng nhớ tạm để dữ liệu và chương trình.

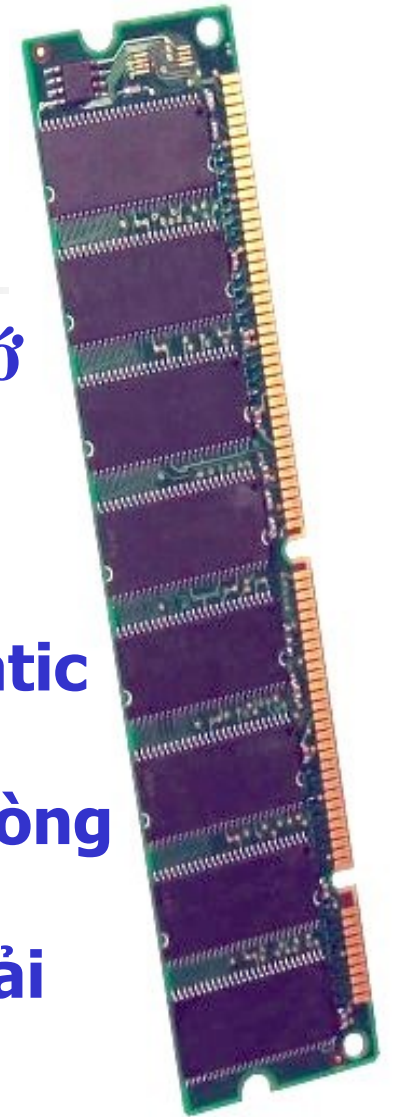


# RAM

Ram là vùng nhớ làm việc → nếu vùng nhớ này trở nên nhỏ so với nhu cầu sử dụng thì ta tăng thêm Ram (gắn thêm Ram).

**RAM có thể chia làm 2 loại : Dynamic và Static RAM**

- **Dynamic RAM** : phải được làm tươi trong vòng dưới 1 ms nếu không sẽ bị mất nội dung.
- **Static RAM** : giữ được giá trị không cần phải làm tươi.
- **RAM tĩnh có tốc độ cao, có tên là bộ nhớ CACHE nằm trong CPU.**



# RAM





# ROM

---

**ROM (Read Only Memory) : bộ nhớ chỉ đọc.**

**ROM BIOS chứa phần mềm cấu hình và chẩn đoán hệ thống, các chương trình con nhập/xuất cấp thấp mà DOS sử dụng. Các chương trình này được mã hoá trong ROM và được gọi là phần dẽo (firmware).**

**Một tính năng quan trọng của ROM BIOS là khả năng phát hiện sự hiện diện của phần cứng mới trong MT và cấu hình lại hệ điều hành theo Driver thiết bị.**





# ROM(cont)

---

## **Đặc điểm của ROM:**

- **Chỉ cho phép đọc không cho phép ghi.**
- **Dữ liệu vẫn tồn tại khi không có nguồn.**



# Các loại Rom

---

## **PROM (Programmable Read Only Memory) :**

Cho phép user có thể lập trình và ghi vào ROM bằng cách đốt.

## **EPROM (Erasable Programmable Read Only Memmory)**

Cho phép user viết ghi chương trình và xóa ghi lại. Việc xóa bằng cách dùng tia cực tím.

## **EEPROM (Electrically Erasable Programmable Read Only Memory)**

bộ nhớ có thể lập trình bằng xung điện đặc biệt



# Secondary Memory

---

**Là bộ nhớ phụ nằm ngoài hộp CPU.**

**Floppy disk, Tapes, Compact discs ... là secondary Memory.**



# Sơ lược về Cache

---

- **Cache cấp 1 (Level 1-cache)** : nằm trong CPU, tốc độ truy xuất rất nhanh, theo tốc độ của CPU.
- **Cache cấp 2 (Level 2-cache)** : thường có dung lượng 128K,256K là cache nằm giữa CPU và Ram, thường cấu tạo bằng Ram tĩnh (Static Ram), tốc độ truy xuất nhanh vì không cần thời gian làm tươi dữ liệu.
- **Cache cấp 3 (Level 3-cache)** : chính là vùng nhớ DRAM dùng làm vùng đệm truy xuất cho đĩa cứng và các thiết bị ngoại vi.  
Tốc độ truy xuất cache cấp 3 chính là tốc độ truy xuất DRAM.



# Cache (cont)

---

- **Tổ chức của Cache :liên quan đến chiến lược trữ đệm và cách thức lưu thông tin trong Cache.**
- **Loại lệnh phải thi hành : Cache chứa cả chương trình và dữ liệu, khi CPU truy xuất mà chúng có sẵn thì truy xuất nhanh. Khi CPU cần truy xuất bộ nhớ, cache sẽ kiểm tra xem cái mà CPU cần đã có trong cache chưa.**
- **Dung lượng cache : như vậy nếu 1 tập lệnh nằm gọn trong cache (vòng lặp chẳng hạn) thì thực thi rất nhanh.**



## Cấu trúc Cache

**Cache được cấu tạo thành từng hàng (cache lines) , 32 bit/hàng cho 386, 128 bit/hàng cho 486, 256 bit/hàng cho Pentium.**

**Mỗi hàng có kèm theo 1 tag để lưu trữ địa chỉ bắt đầu của đoạn bộ nhớ mà thông tin được đưa vào cache. Nếu là cache cấp 2 (SRAM), địa chỉ bắt đầu của đoạn bộ nhớ đã chuyển data vào cache còn được lưu trong 1 vùng nhớ riêng.**

**Một bộ điều khiển cache (cache controller) sẽ điều khiển hoạt động của cache với CPU và data vào/ra cache. Chính Cache controller phản ánh chiến lược trữ đệm của cache.**

**Với cache cấp 1, cache controller là 1 thành phần của CPU.**

**Với cache cấp 2, cache controller nằm trên Mainboard.**



# Hiệu suất của Cache

---

Cache dùng làm vùng đệm truy xuất nên nếu CPU truy xuất data mà có sẵn trong cache thì thời gian truy xuất nhanh hơn nhiều. Hiệu quả của cache ngoài việc cho tốc độ truy xuất nhanh còn phụ thuộc vào Cache hit hoặc Cache miss.

**Cache Hit** : tức data có sẵn trong Cache.

**Cache Miss** : tức data chưa có sẵn trong cache.

tỉ lệ cache hit và cache miss phụ thuộc vào 3 yếu tố :

tổ chức cache , loại lệnh phải thi hành và dung lượng của cache.

# Hiệu suất của Cache

## Tính toán hiệu suất thực thi của Cache :

Gọi  $c$  thời gian truy xuất của Cache

$M$  là thời gian truy xuất bộ nhớ

$h$  là tỉ lệ thành công (hit ratio), là tỉ số giữa số lần tham chiếu cache với tổng số lần tham chiếu.  $h = (k-1)/k$

Tỉ lệ thất bại (miss ratio)  $(1-h)$

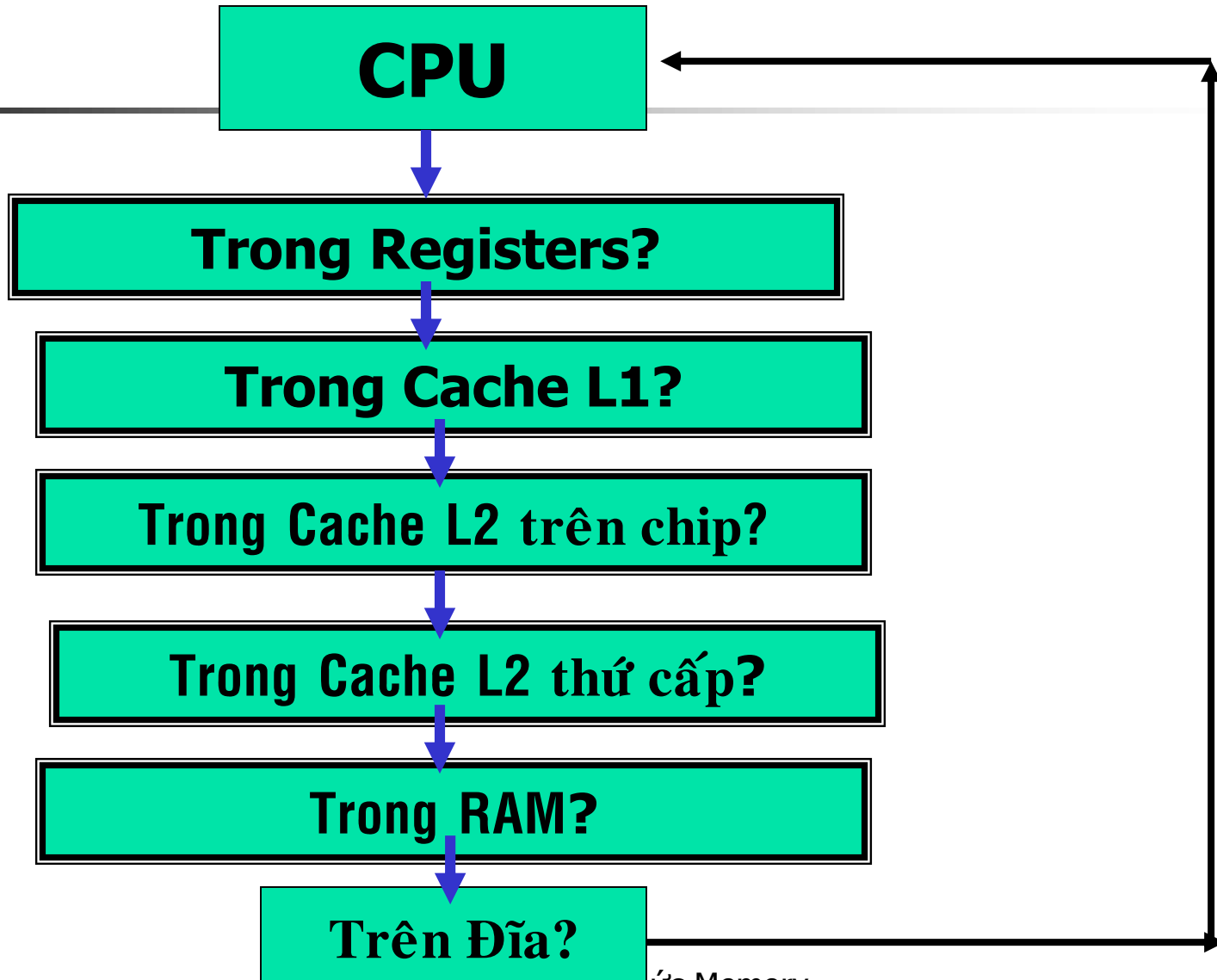
Thời gian truy xuất trung bình  $= c + (1-h)m$

Khi  $h \rightarrow 1$ , tất cả truy xuất đều tham chiếu tới Cache, thời gian truy xuất trung bình  $\rightarrow c$ .

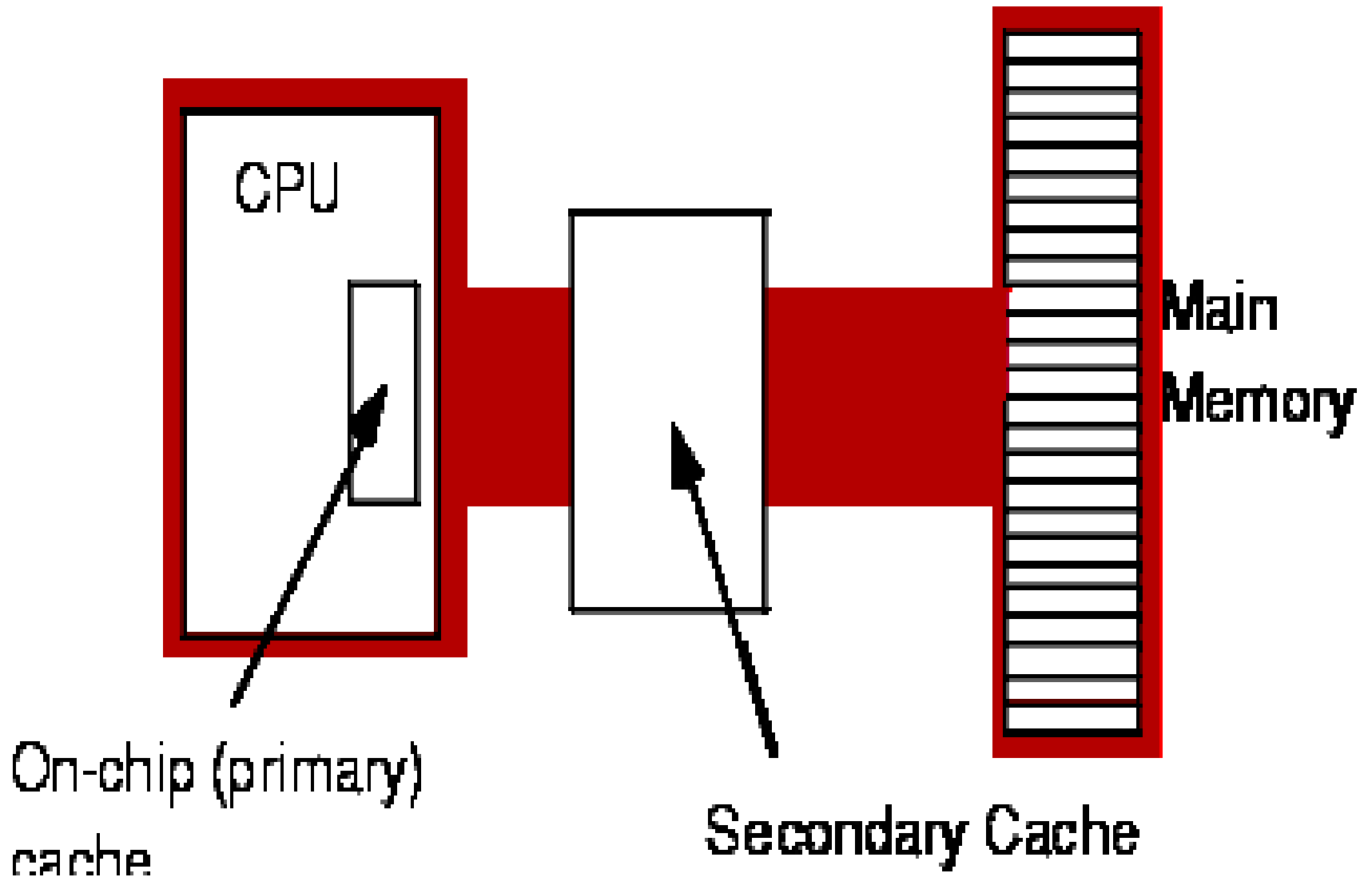
Khi  $h \rightarrow 0$ , cần phải tham chiếu bộ nhớ chính mọi lúc, thời gian truy xuất trung bình  $\rightarrow c+m$ .



# Hiệu suất của Cache (cont)



# A Two Level Caching System

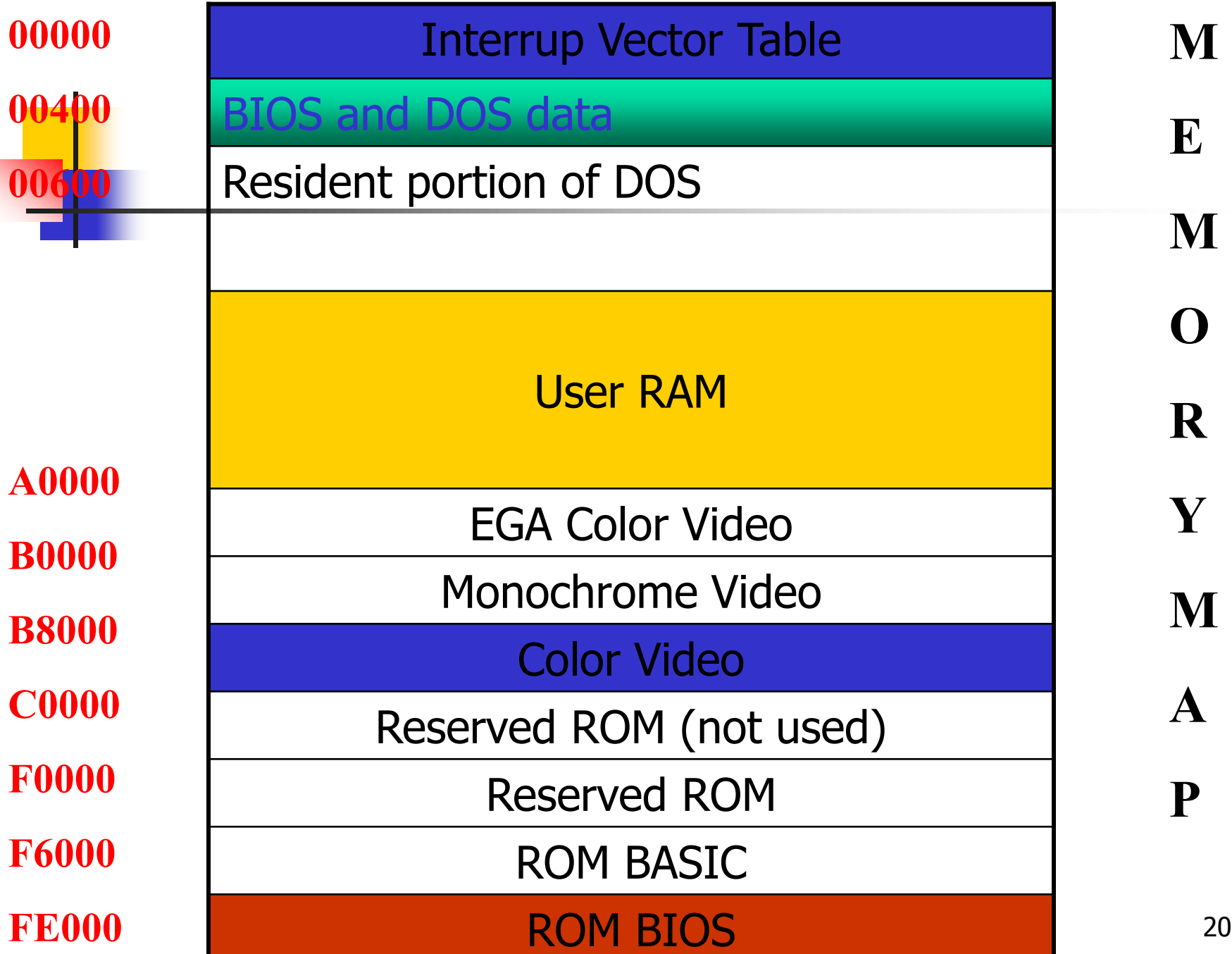




# Các chiến lược trữ đệm trong Cache

Các chiến lược trữ đệm liên quan đến tác vụ đọc ghi từ CPU. Có 2 loại :  
Writethrough Cache (WTC) và Writeback cache (WBC).

- **Khi CPU đọc từ bộ nhớ qui ước** thì WTC và WBC đều như nhau : sẽ đọc 1 đoạn nội dung trong bộ nhớ vào cache.
- **Khi CPU ghi ra bộ nhớ qui ước :**
  - **WTC** : CPU ghi data ra vùng đệm ghi (write buffer) rồi bỏ đó tiếp tục việc khác, cache sẽ lấy nội dung trong buffer rồi chịu trách nhiệm ghi ra bộ nhớ qui ước khi bus rảnh.
  - **WBC** : CPU ghi data vào cache, khi cache đầy thì đẩy thông tin ra bộ đệm (đệm castoff) rồi từ castoff, data chuyển sang bộ nhớ qui ước.



# Memory Map

**1024 bytes thấp nhất chứa bảng vector**

## **interrupt**

Dos data Area chứa các biến được DOS sử dụng như :

- Keyboard buffer : các phím nhấn được lưu cho đến khi được xử lý.
- Cờ chỉ tình trạng keyboard : cho biết phím nào đang được nhấn.
- Địa chỉ cổng printer.
- Địa chỉ cổng tuần tự
- Mô tả các thiết bị đang có trong hệ thống : tổng dung lượng bộ nhớ, số ổ đĩa, kiểu màn hình...



# Memory Map

---

User Ram : vị trí thường trú của DOS ở địa chỉ 0600H.  
Vùng nhớ trống nằm ngay dưới vùng nhớ Dos.

**Rom Area : từ C000H – FFFFH được IBM dành riêng cho Rom sử dụng chứa hard disk controller, Rom Basic.**

**Rom BIOS : từ F000H – FFFFH vùng nhớ cao nhất của bộ nhớ chứa các chương trình con cấp thấp của Dos dùng cho việc xuất nhập và các chức năng khác..**



# Quá trình Boot máy

---

- **Xãy ra khi ta power on hay nhấn nút Reset.**

Bộ VXL xóa tất cả ô nhớ của bộ nhớ trở về 0, kiểm tra chẩn lẻ bộ nhớ, thiết lập thanh ghi CS trở đến segment FFFFh và con trỏ lệnh IP trở tới địa chỉ offset bằng 0.

→ Chỉ thị đầu tiên được MT thực thi ở địa chỉ ấn định bởi nội dung cặp thanh ghi CS:IP, đó chính là FFFF0H , điểm nhập tới BIOS trong ROM.



## Trình tự tác vụ đọc ô nhớ

---

- ✓ CPU đưa địa chỉ ô nhớ cần đọc vào thanh ghi địa chỉ.
- ✓ Mạch giải mã xác định địa chỉ ô nhớ.
- ✓ CPU gửi tín hiệu điều khiển đọc → bộ nhớ. Nội dung ô nhớ cần đọc được đưa ra thanh ghi dữ liệu.
- ✓ CPU đọc nội dung của thanh ghi dữ liệu.





## Mạch giải mã địa chỉ ô nhớ

---

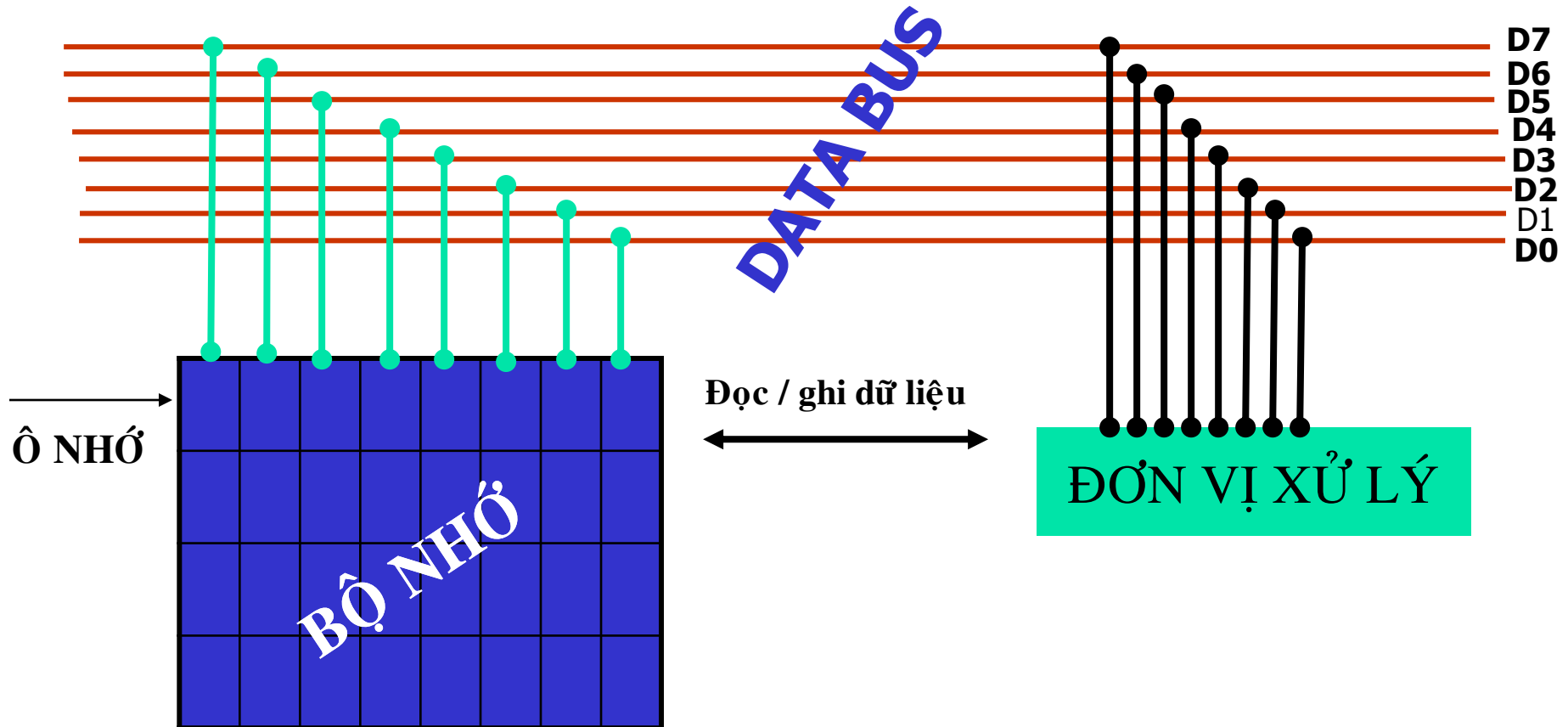
**Mạch điện có nhiệm vụ xác định đúng ô nhớ cần truy xuất đang có địa chỉ lưu trong thanh ghi địa chỉ.**

**Bộ nhớ làm việc được chia thành nhiều ô nhớ.**

**Kích thước mỗi ô nhớ thay đổi tùy theo máy, thường là 8 hay 16 bit tức 1 byte hay 1 word.**

**Nếu kích thước mỗi ô nhớ là 1 byte thì sẽ có 8 đường dữ liệu song song nối bộ nhớ làm việc với bộ VXL. Mỗi đường 1 bit, tất cả 8 đường tạo thành một tuyến dữ liệu (data bus)**

# Truy xuất bộ nhớ (cont)



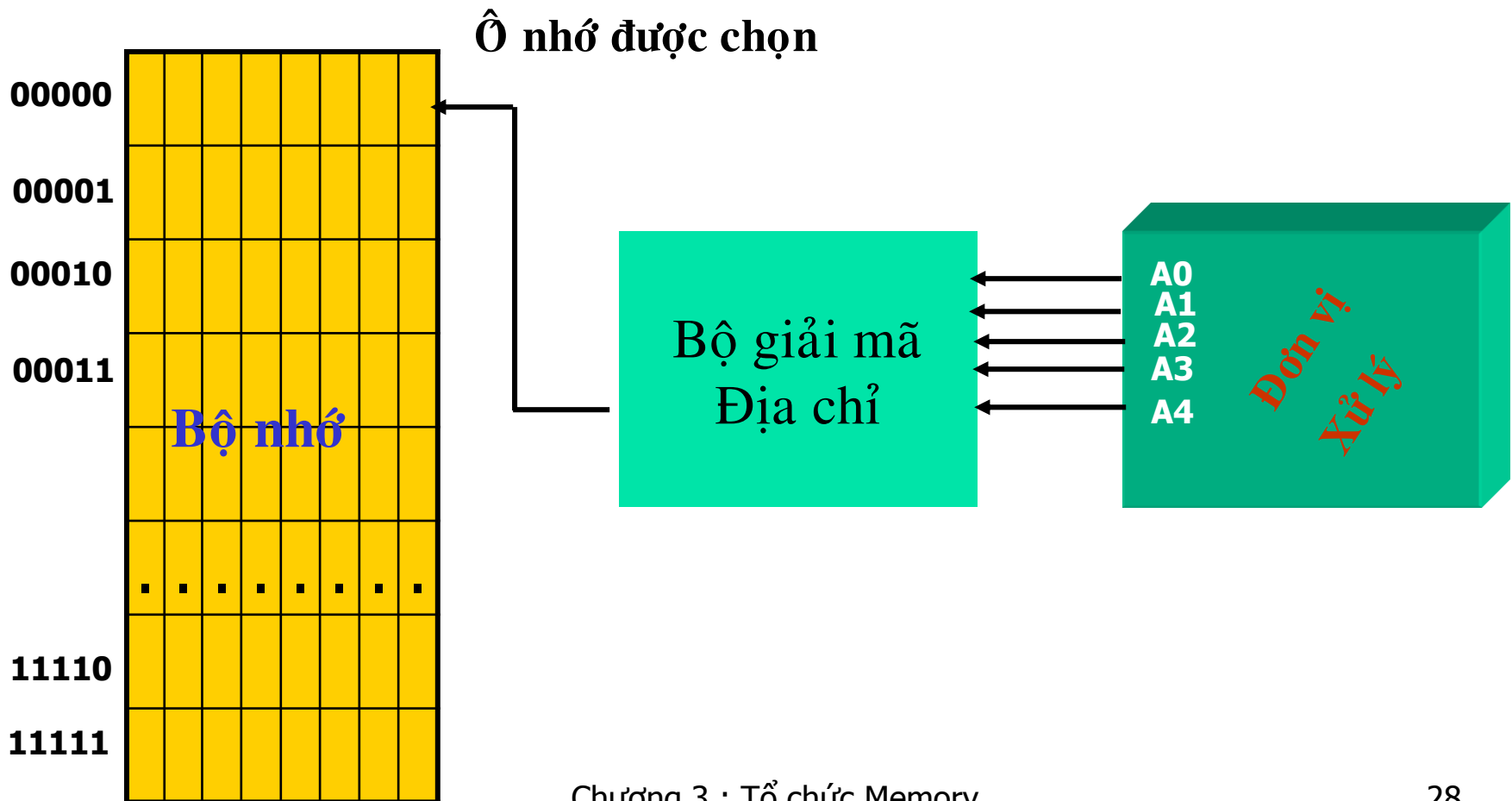


## Trình tự tác vụ ghi ô nhớ

---

- CPU đưa địa chỉ ô nhớ cần ghi vào thanh ghi địa chỉ của bộ nhớ.
- Mạch giải mã xác định địa chỉ ô nhớ.
- CPU đưa dữ liệu cần ghi vào thanh ghi dữ liệu của bộ nhớ.
- CPU gửi tín hiệu điều khiển ghi → bộ nhớ. Nội dung trong thanh ghi dữ liệu được ghi vào ô nhớ có địa chỉ xác định.

# Truy xuất bộ nhớ : ghi ô nhớ





# Stack

---

- **Stack là vùng nhớ đặc biệt dùng để lưu trữ địa chỉ và dữ liệu.**

**Stack thường trú trong stack segment. Mỗi vùng 16 bit trên stack được trỏ đến bởi thanh ghi SP, gọi là stack pointer.**

**Stack pointer lưu trữ địa chỉ của phần tử dữ liệu cuối mới được thêm vào (pushed lên stack.)**



# Stack

---

**phần tử dữ liệu cuối mới được thêm vào này lại là phần tử sẽ được lấy ra (popped trước tiên).**

**→ Stack làm việc theo cơ chế LIFO (Last In First Out).**

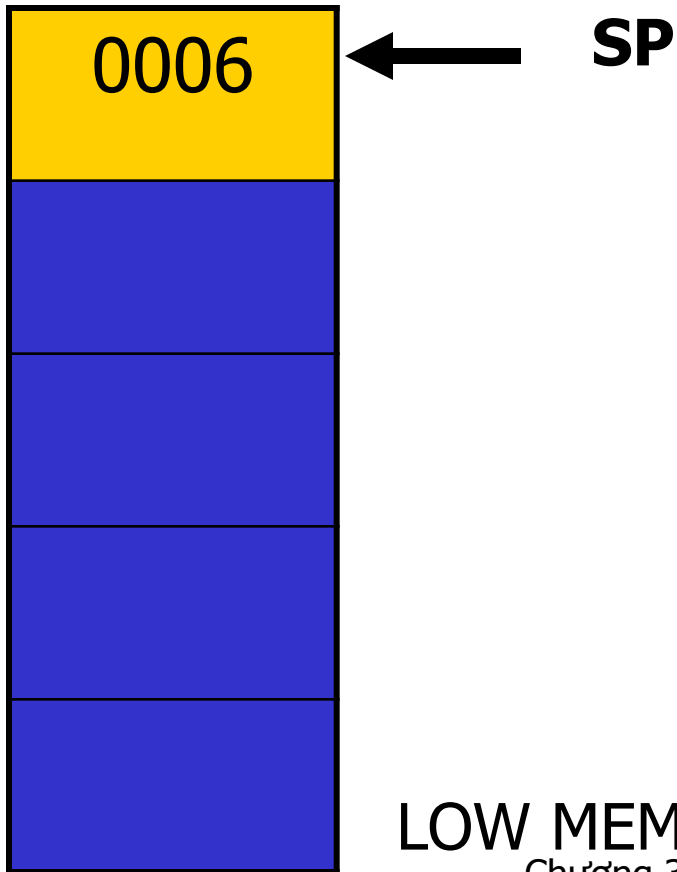
**Xét ví dụ sau : giả sử stack đang chứa 1 giá trị 0006**



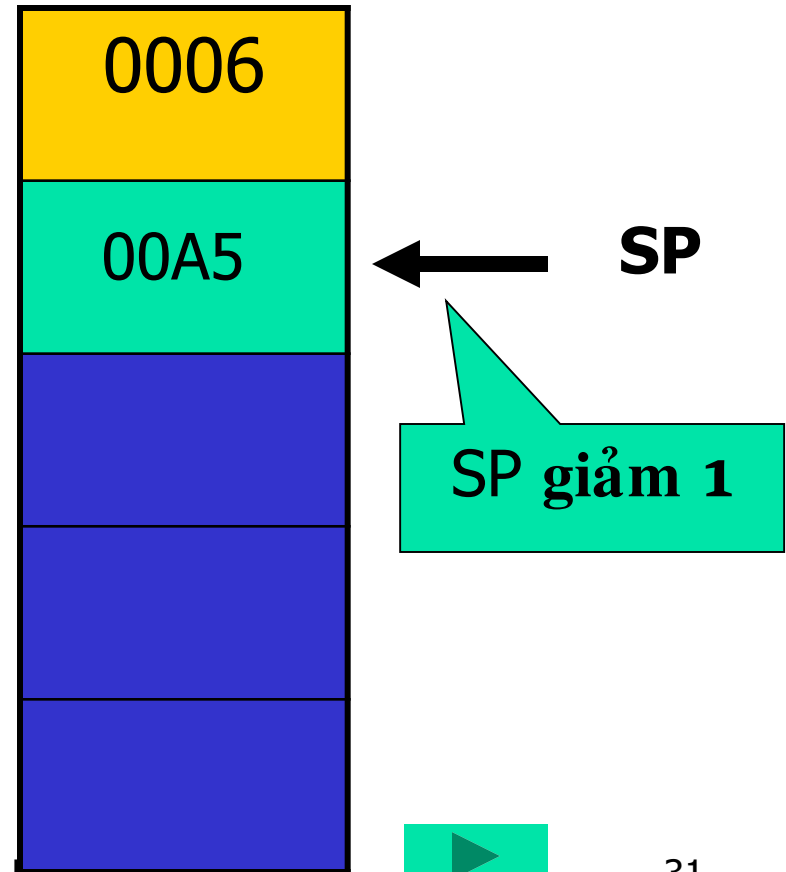
Sau đó ta đưa 00A5 vào stack

# Stack

BEFORE *HIGH MEM*



AFTER *HIGH MEM*





## Công dụng của Stack

---

- ❖ Dùng để lưu trữ dữ liệu tạm cho thanh ghi nếu ta cần sử dụng các dữ liệu này.
- ❖ Khi 1 chương trình con được gọi, stack sẽ lưu trữ địa chỉ trở về ngay sau khi chương trình con thực hiện xong.
- ❖ Các ngôn ngữ cấp cao thường tạo ra 1 vùng nhớ bên trong chương trình con gọi là stack frame để chứa các biến cục bộ.





# Summary Slide

---

- Cờ nào được thiết lập khi 1 phép tính số học không dấu quá rộng không vừa với đích?
- Hai thanh ghi nào được tổ hợp thành địa chỉ của lệnh sẽ được thực kế tiếp?
- Nêu quá trình đọc bộ nhớ. Tại sao quá trình đọc bộ nhớ lại chiếm nhiều chu kỳ máy hơn so với truy cập thanh ghi?
- Thanh ghi AH bị sửa đổi, tại sao thanh ghi AX cũng thay đổi theo.
- Nội dung nào chiếm 1024 bytes thấp nhất của bộ nhớ?



# Câu hỏi ôn tập

---

- Vai trò của Cache trong máy tính.
- Trình bày chiến lược trữ đệm của Cache.
- Phân biệt bộ nhớ RAM và ROM.
- Nêu trình tự quá trình thực hiện khi khởi động máy tính.



# Câu hỏi ôn tập

---

- Một bộ nhớ có dung lượng  $4K \times 8$ .
  - a) Có bao nhiêu đầu vào dữ liệu, đầu ra dữ liệu.
  - b) Có bao nhiêu đường địa chỉ.
  - c) Dung lượng của nó tính theo byte.



# Câu hỏi ôn tập

---

Bộ nhớ Cache nằm giữa :

- a) Mainboard và CPU
- b) ROM và CPU
- c) CPU và bộ nhớ chính.
- d) Bộ nhớ chính và bộ nhớ ngoài



# Câu hỏi ôn tập

---

Theo quy ước, người ta chia bộ nhớ thành từng vùng có những địa chỉ được mô tả bằng :

- a) số thập phân
- b) số thập lục phân
- c) số nhị phân
- d) số bát phân