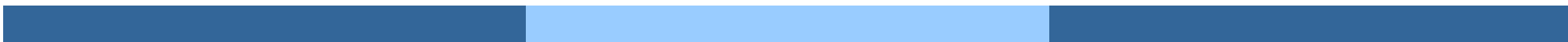


Chương 4:

ĐƠN VỊ XỬ LÝ TRUNG TÂM



Nội dung môn học

- Chương 1: Giới thiệu chung
- Chương 2: Tổng quan về hệ thống máy tính
- Chương 3: Biểu diễn dữ liệu và số học
- **Chương 4: Đơn vị xử lý trung tâm**
- Chương 5: Bộ nhớ máy tính
- Chương 6: Hệ thống vào ra
- Tổng kết – ôn tập



Nội dung

- Tổ chức của CPU
- Thanh ghi
- Hoạt động của chu trình lệnh
- Kỹ thuật pipeline
- Giới thiệu chung về tập lệnh



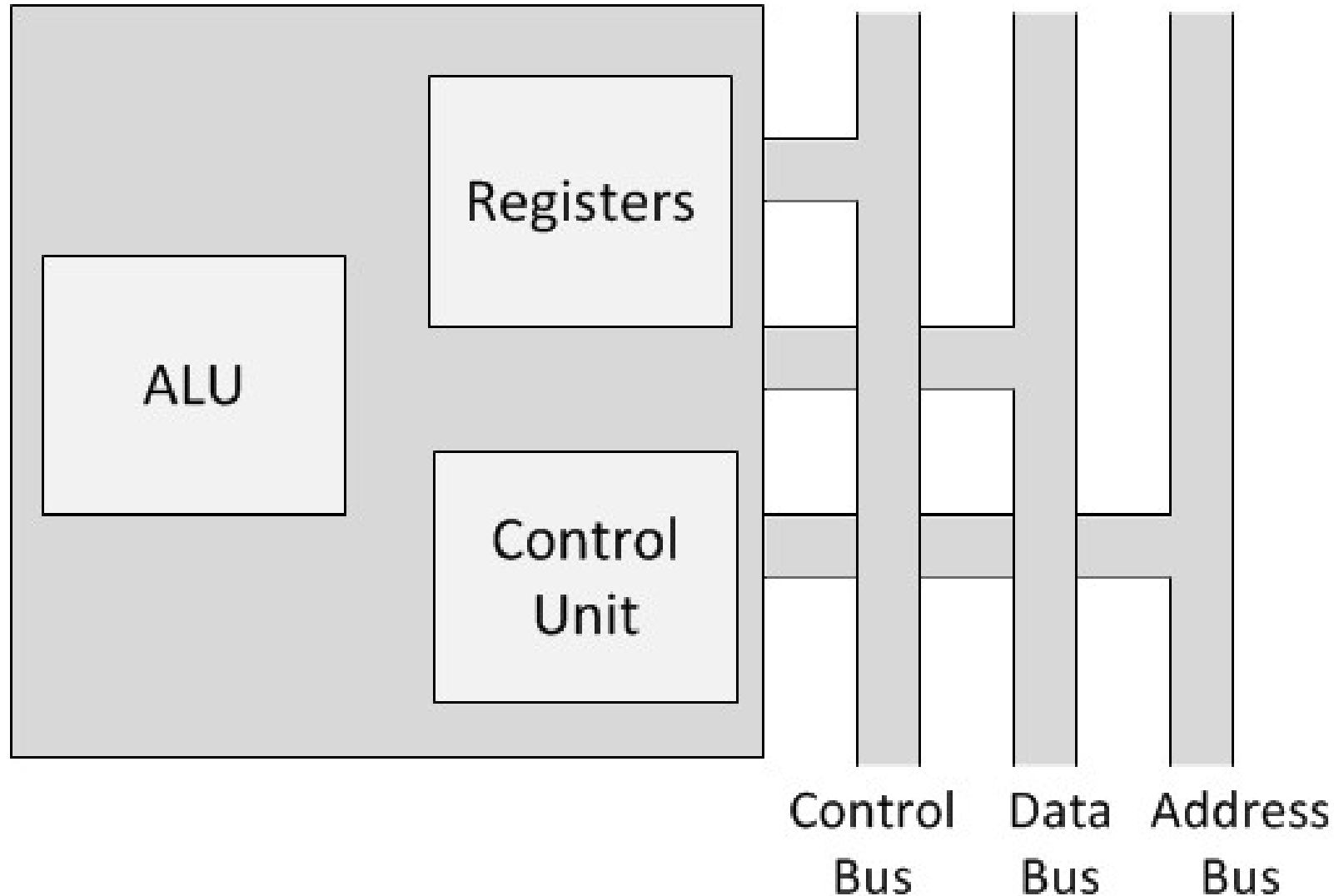
Cấu trúc của CPU

- ❑ Các công việc mà CPU phải thực hiện:
 - ❑ Nạp lệnh (fetch instruction): Bộ xử lý đọc một câu lệnh từ bộ nhớ (thanh ghi, bộ nhớ chính, cache).
 - ❑ Thông dịch lệnh (Interpret instruction): câu lệnh được giải mã để quyết định hoạt động nào được yêu cầu.
 - ❑ Nạp dữ liệu (fetch data): nạp dữ liệu để thực thi câu lệnh. Dữ liệu có thể được nạp từ bộ nhớ hoặc I/O.
 - ❑ Xử lý dữ liệu (process data): thực hiện các phép toán số học và luận lý trên dữ liệu.
 - ❑ Lưu dữ liệu (write data): dữ liệu sau khi được xử lý được lưu lại. Dữ liệu có thể được lưu trên bộ nhớ hoặc lưu trên I/O.



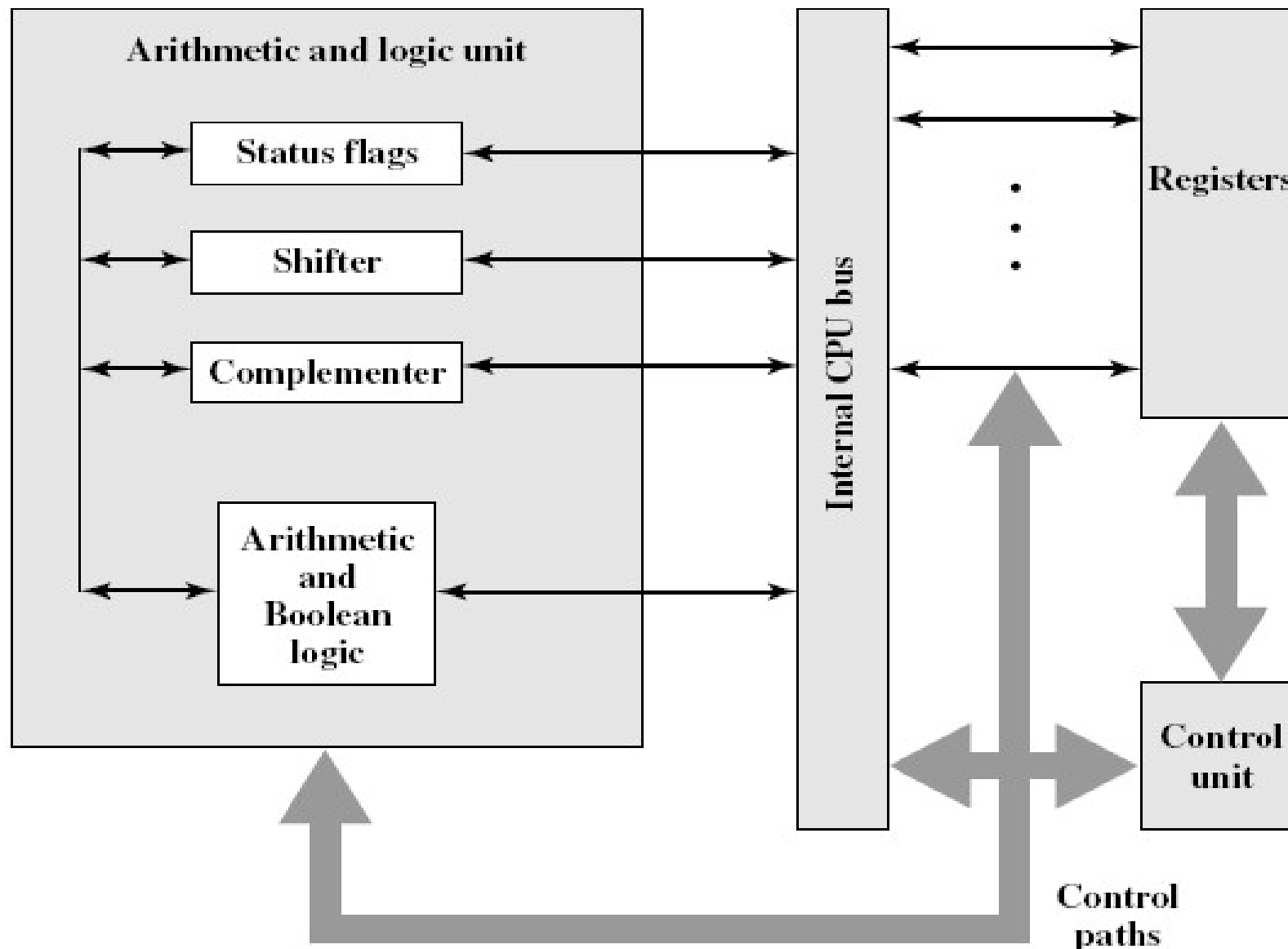
Cấu trúc của CPU (tt)

□ Cấu trúc cơ bản của CPU



Cấu trúc của CPU (tt)

□ Một mức chi tiết hơn của CPU

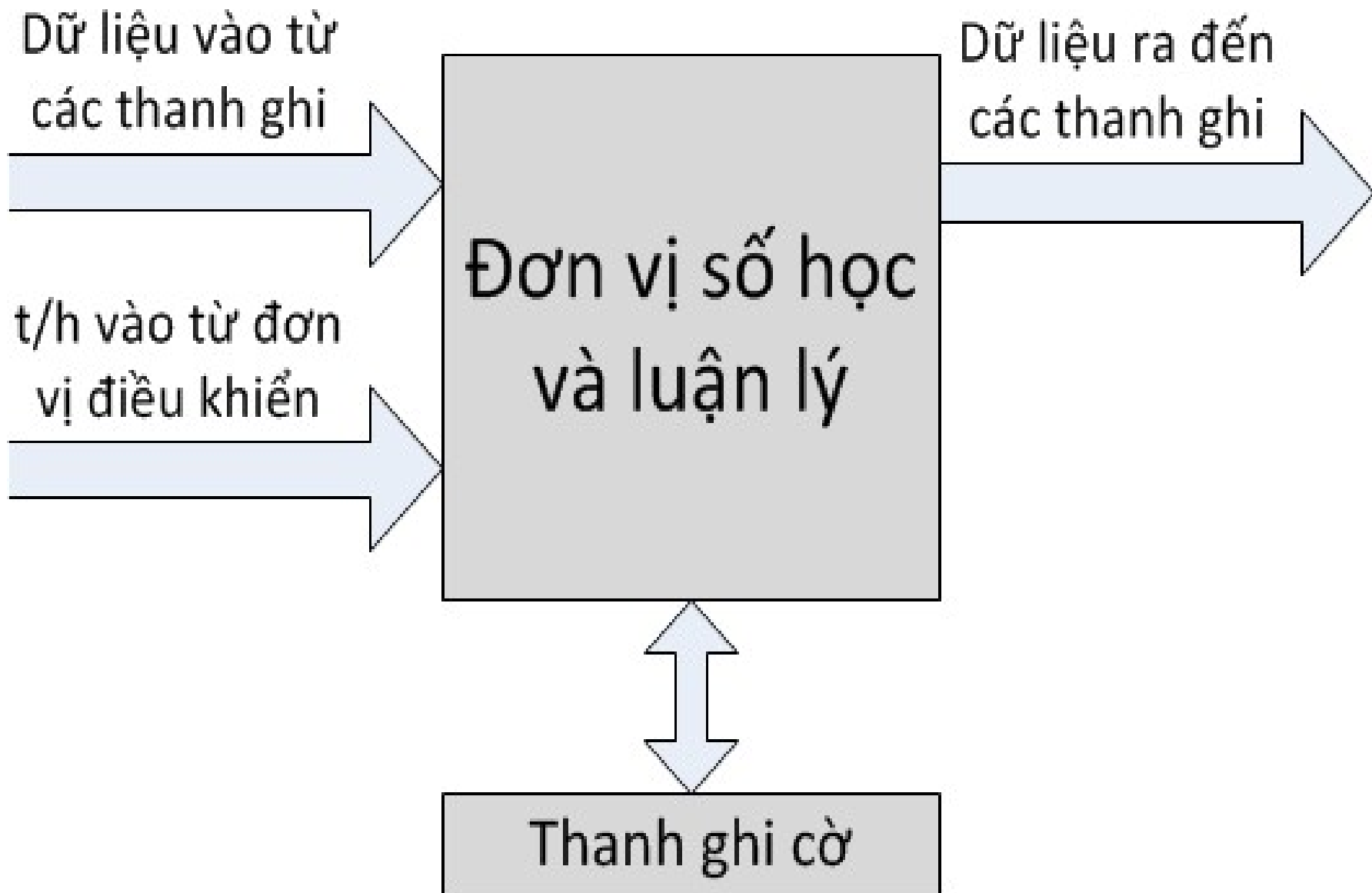


Đơn vị số học và luận lý

- ❑ Chức năng: Thực hiện các phép toán số học và phép toán luận lý.
 - ❑ Số học: cộng, trừ, nhân, chia, tăng, giảm, đảo dấu
 - ❑ Luận lý: AND, OR, XOR, NOT, phép dịch bit.



Mô hình kết nối đơn vị số học và luận lý

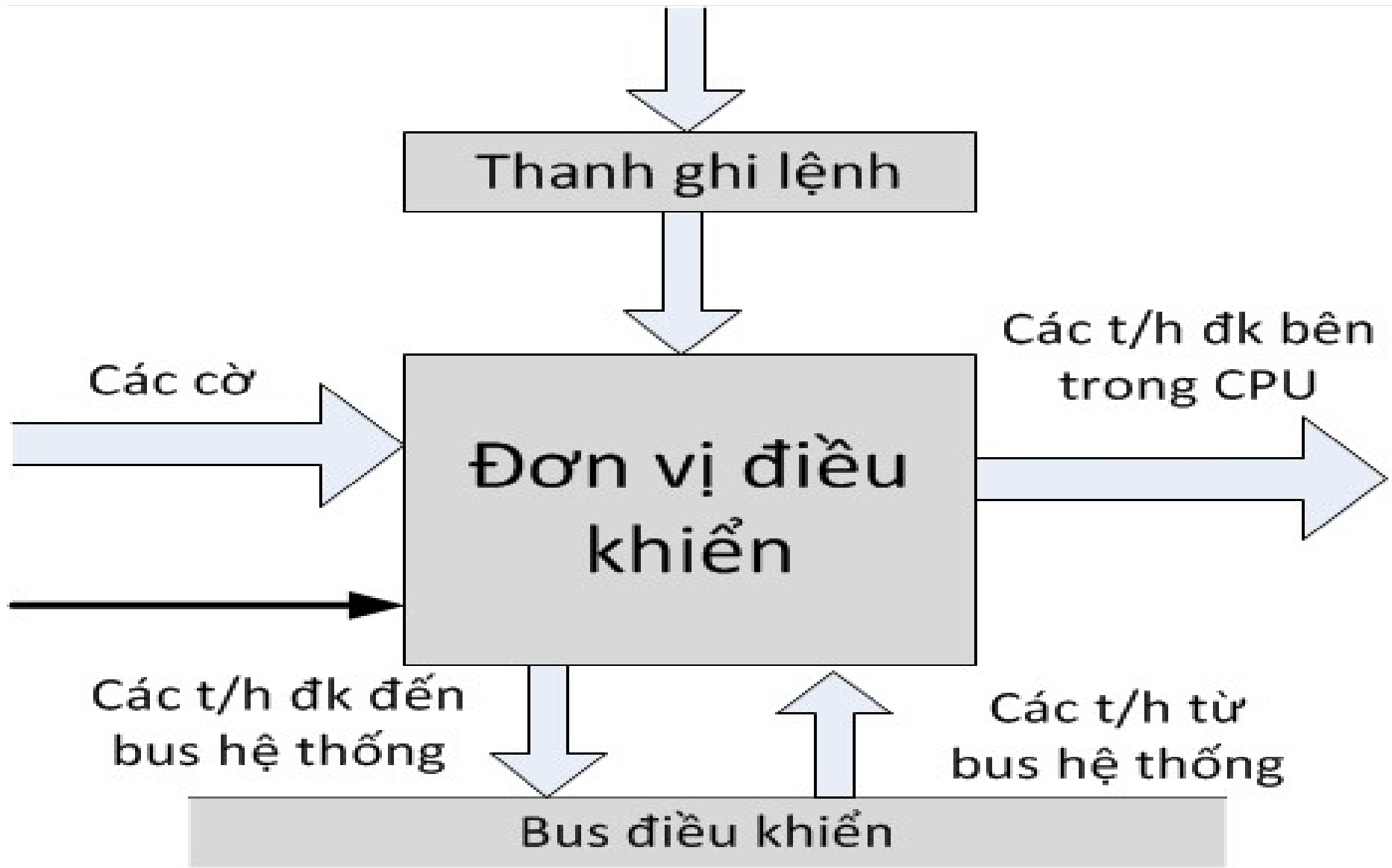


Đơn vị điều khiển

- ❑ Điều khiển nhận lệnh từ bộ nhớ đưa vào thanh ghi lệnh
- ❑ Tăng nội dung của PC để trở sang lệnh kế tiếp
- ❑ Giải mã lệnh đã được nhận để xác định thao tác mà lệnh yêu cầu
- ❑ Phát ra các tín hiệu điều khiển thực hiện lệnh
- ❑ Nhận các tín hiệu yêu cầu từ bus hệ thống và đáp ứng với các yêu cầu đó.



Mô hình kết nối đơn vị điều khiển



Các tín hiệu đến đơn vị điều khiển

- ❑ Clock: tín hiệu nhịp từ mạch tạo dao động bên ngoài.
- ❑ Mã lệnh từ thanh ghi lệnh đưa đến để giải mã.
- ❑ Các cờ từ thanh ghi cờ cho biết trạng thái của CPU.
- ❑ Các tín hiệu yêu cầu từ bus điều khiển



Các tín hiệu đến đơn vị điều khiển

- Các tín hiệu điều khiển bên trong CPU:
 - Điều khiển các thanh ghi
 - Điều khiển ALU
- Các tín hiệu điều khiển bên ngoài CPU:
 - Điều khiển bộ nhớ
 - Điều khiển các môđun vào-ra



Thanh ghi

- Chứa các thông tin tạm thời phục vụ cho hoạt động ở thời điểm hiện tại của CPU, gồm hai loại:
- Thanh ghi người sử dụng nhìn thấy (User-visible registers):
 - Cho phép người lập trình truy cập và sử dụng thanh ghi thay vì tham khảo bộ nhớ liên tục.
- Thanh ghi điều khiển và trạng thái (Control and status registers):
 - Sử dụng bởi đơn vị điều khiển để điều khiển hoạt động của đơn vị xử lý và sử dụng bởi các đặc quyền của hệ điều hành để điều khiển sự thực thi của chương trình.



User-visible registers

- Thanh ghi mục đích chung:
 - Được gán cho từng chức năng riêng biệt bởi người lập trình.
- Thanh ghi dữ liệu:
 - Dùng để chứa dữ liệu
- Thanh ghi địa chỉ
 - Dùng để chứa địa chỉ
- Thanh ghi điều khiển (thanh ghi cờ)
 - Liên quan việc bật các bit giá trị của bộ xử lý khi có một kết quả hay điều kiện liên quan.



Thanh ghi điều khiển và trạng thái

- Thanh ghi điều khiển: Điều khiển các hoạt động của đơn vị xử lý. Một số thanh ghi thông dụng như:
 - Bộ đếm chương trình: Chứa địa chỉ câu lệnh kế tiếp sẽ được nạp.
 - Thanh ghi lệnh: Chứa câu lệnh được nạp gần nhất.
 - Thanh ghi địa chỉ bộ nhớ (Memory Address Register: MAR): Chứa địa chỉ của một nơi trong bộ nhớ, thường được kết nối trực tiếp với bus địa chỉ.
 - Thanh ghi đệm dữ liệu (Memory Buffer Register: MBR): Chứa một từ dữ liệu sẽ được lưu vào bộ nhớ, thường được kết nối trực tiếp với bus dữ liệu.



Thanh ghi điều khiển và trạng thái (tt)

- Thanh ghi trạng thái (cờ trạng thái):
 - Dấu (sign): chứa bit dấu của kết quả của thao tác số học mới nhất.
 - Zero: được bật lên khi kết quả là 0.
 - Carry: được bật lên nếu một thao tác có nhớ.
 - Equal: được bật lên nếu so sánh bằng là đúng.
 - Overflow: được bật lên nếu có tràn số học
 - Interrupt Enable/Disable: được sử dụng để cho phép hay không cho phép ngắt.



Ngăn xếp (stack)

- ❑ Ngăn xếp là vùng nhớ có cấu trúc LIFO (FILO)
 - ❑ LIFO: Last In First Out
- ❑ Ngăn xếp thường dùng để phục vụ cho chương trình con
- ❑ Đáy ngăn xếp là một ngăn nhớ xác định
- ❑ Đỉnh ngăn xếp là thông tin nằm ở vị trí trên cùng trong ngăn xếp
- ❑ Đỉnh ngăn xếp có thể bị thay đổi



Ngăn xếp (stack)

- SP chứa địa chỉ của ngăn nhớ đỉnh ngăn xếp
- Khi cất một thông tin vào ngăn xếp:
 - Nội dung của SP tự động tăng
 - Thông tin được cất vào ngăn nhớ được trỏ bởi SP
- Khi lấy một thông tin ra khỏi ngăn xếp:
 - Thông tin được đọc từ ngăn nhớ được trỏ bởi SP
 - Nội dung của SP tự động giảm
- Khi ngăn xếp rỗng, SP trở vào đáy

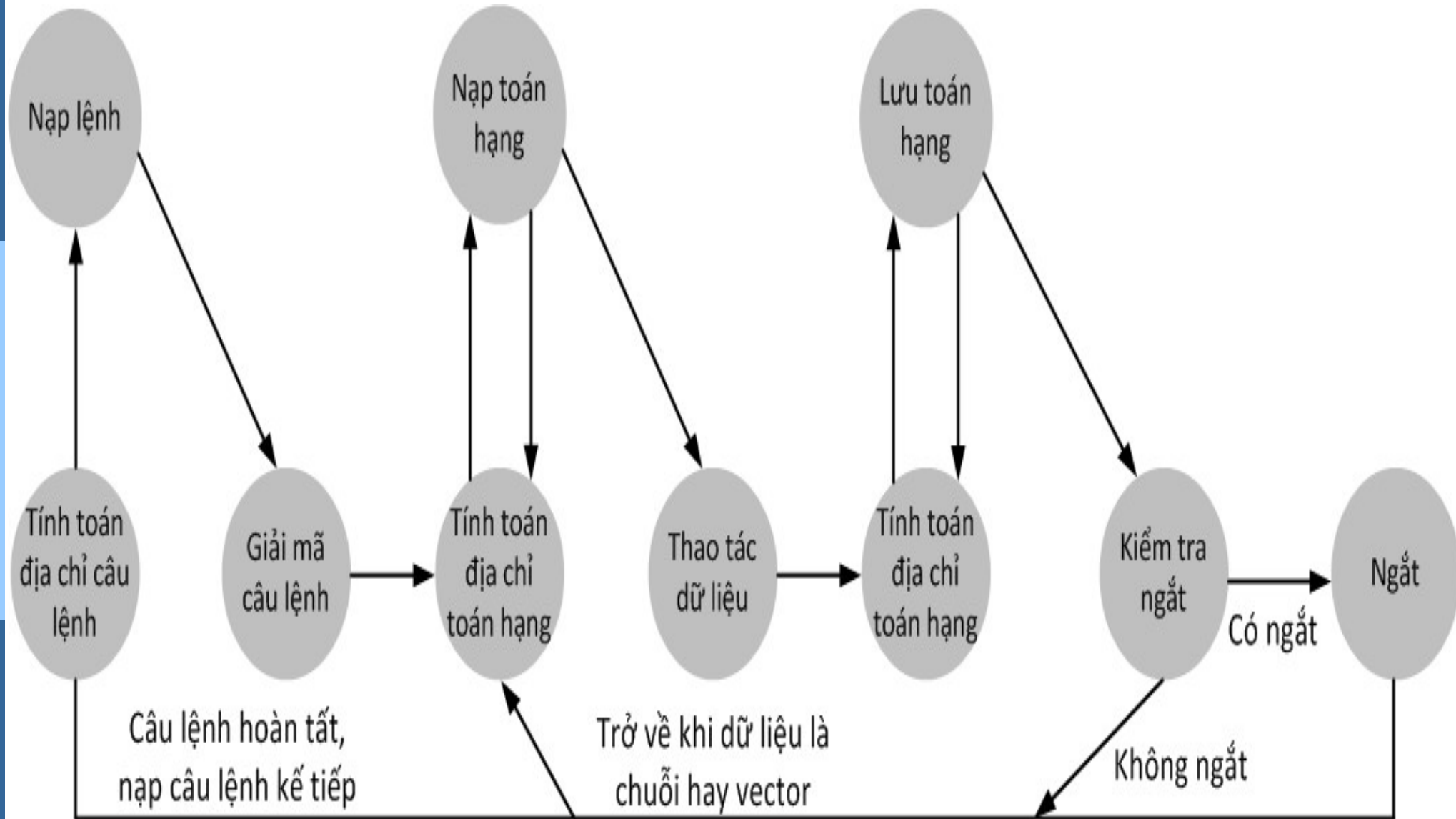


Hoạt động của chu trình lệnh

- Chu trình lệnh:
 - Nạp lệnh
 - Giải mã lệnh
 - Nạp toán hạng
 - Thực hiện lệnh
 - Cất toán hạng
 - Ngắt.



Lược đồ trạng thái chu trình lệnh

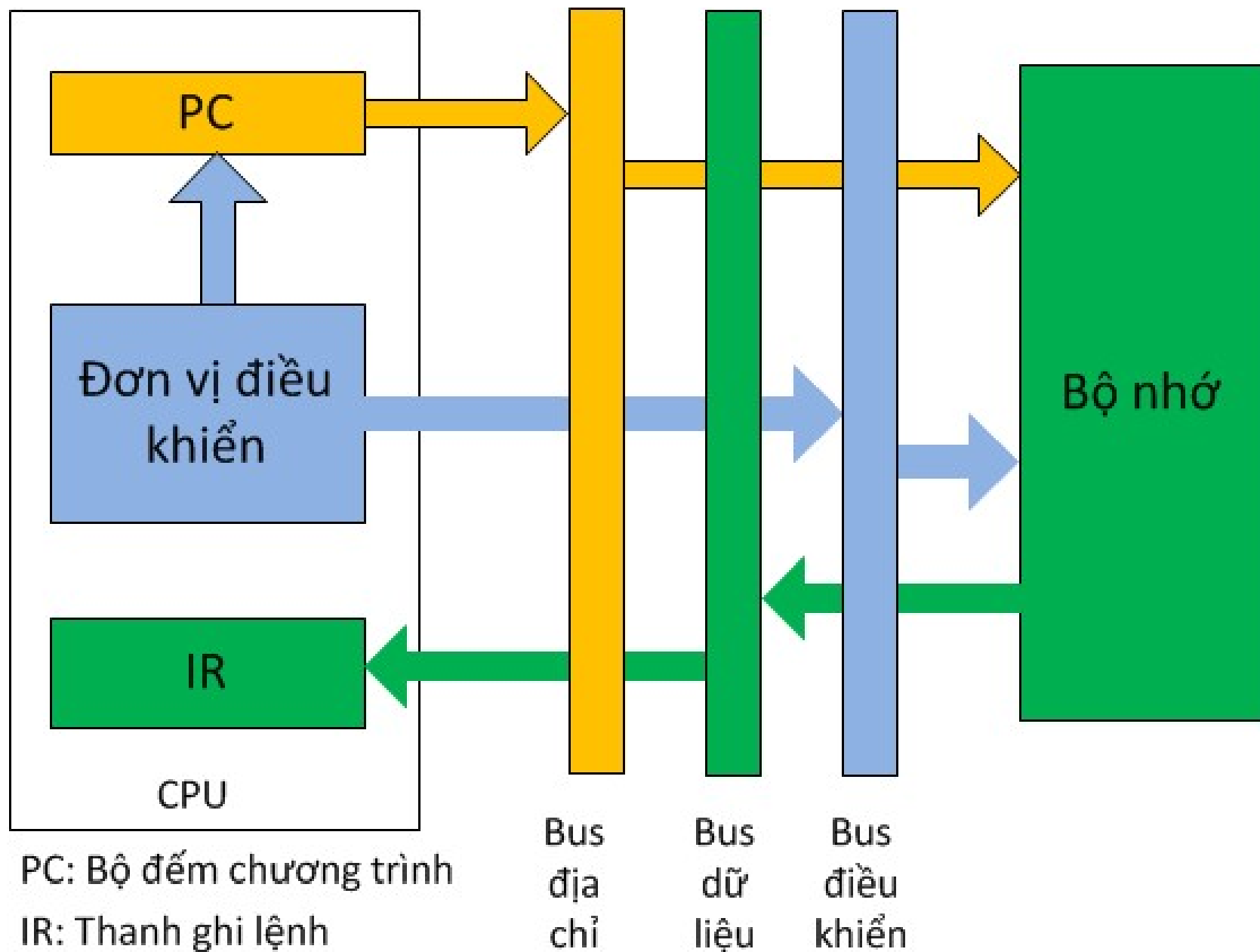


Nạp lệnh

- ❑ CPU đưa địa chỉ của lệnh cần nạp từ bộ đếm chương trình (PC) ra bus địa chỉ
- ❑ CPU phát tín hiệu điều khiển đọc bộ nhớ
- ❑ Lệnh từ bộ nhớ được đặt lên bus dữ liệu, chuyển đến CPU và được CPU copy vào thanh ghi lệnh IR
- ❑ CPU tăng nội dung PC để trở sang lệnh kế tiếp



Nạp lệnh



Giải mã lệnh

- ❑ Lệnh từ thanh ghi lệnh IR được đưa đến đơn vị điều khiển.
- ❑ Đơn vị điều khiển tiến hành giải mã lệnh để xác định thao tác phải thực hiện.
- ❑ Giải mã lệnh xảy ra bên trong CPU



Tính toán địa chỉ toán hạng

- Dựa vào câu lệnh được giải mã. CPU xác định được các toán hạng trong câu lệnh.
- CPU tính toán và lưu địa chỉ cần đọc/ lưu dữ liệu của toán hạng trong các thanh ghi đệm địa chỉ (MAR).

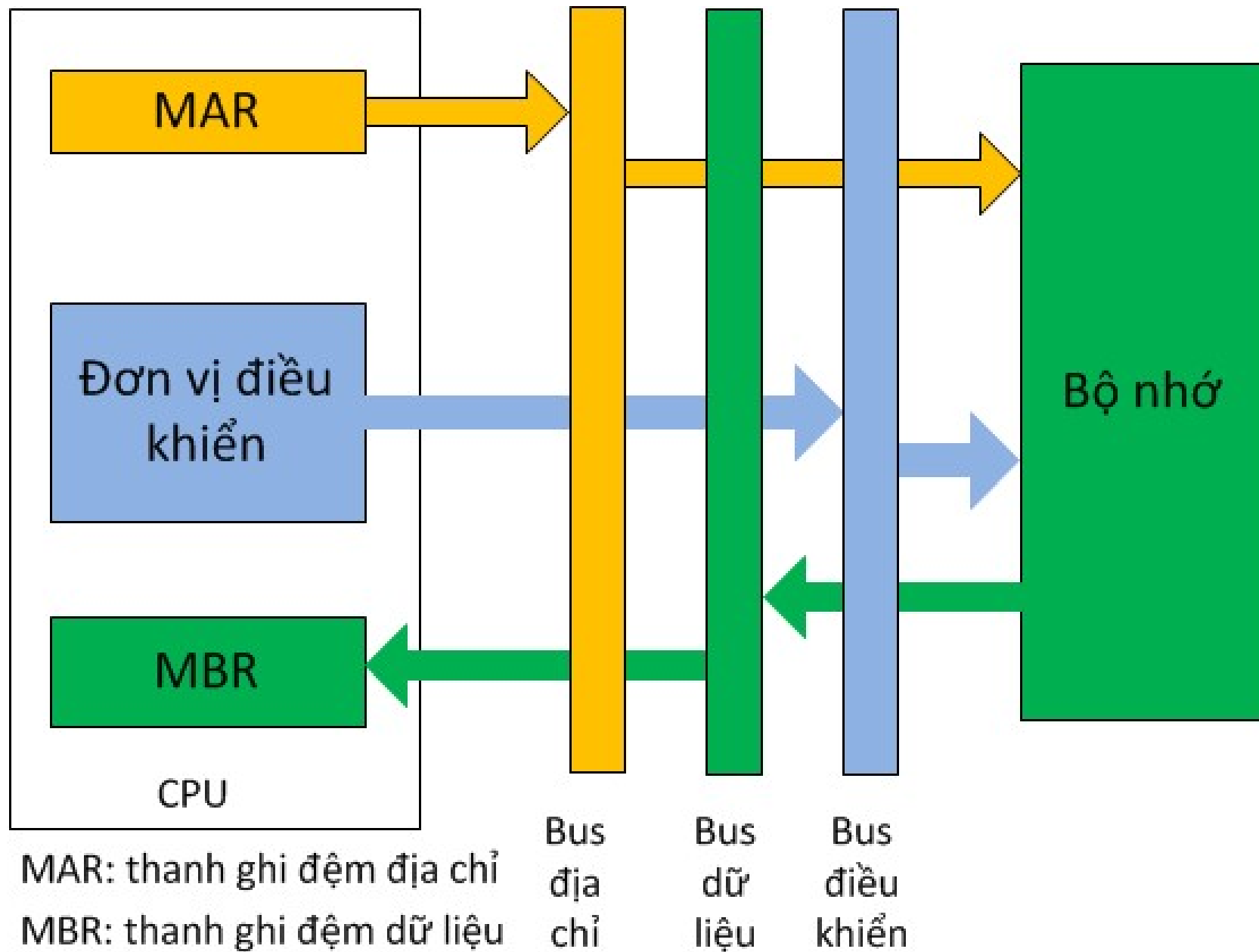


Nạp toán hạng

- ❑ CPU đưa địa chỉ của toán hạng ra bus địa chỉ
- ❑ CPU phát tín hiệu điều khiển đọc
- ❑ Toán hạng được đọc vào CPU
- ❑ Tương tự như nhận lệnh



Nạp toán hạng



Thực hiện lệnh

- Câu lệnh có nhiều dạng tùy thuộc vào lệnh, các câu lệnh có thể là:
 - Đọc/Ghi bộ nhớ
 - Vào/Ra
 - Chuyển giữa các thanh ghi
 - Thao tác số học/ logic
 - Chuyển điều khiển (rẽ nhánh)
 - ,...

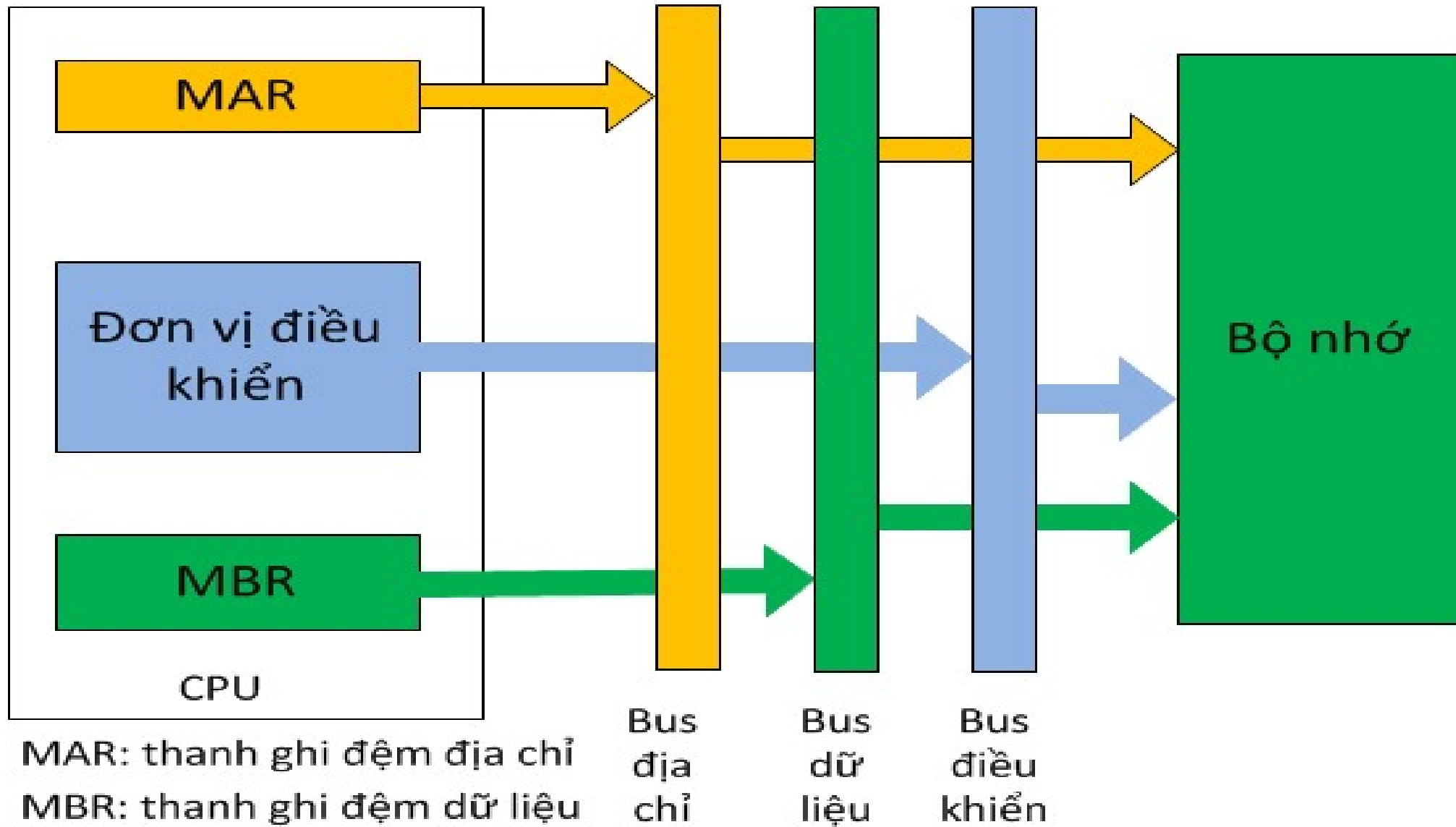


Lưu toán hạng (kết quả)

- CPU đưa địa chỉ ra bus địa chỉ
- CPU đưa dữ liệu cần ghi ra bus dữ liệu
- CPU phát tín hiệu điều khiển ghi
- Dữ liệu trên bus dữ liệu được copy đến vị trí xác định



Lưu toán hạng

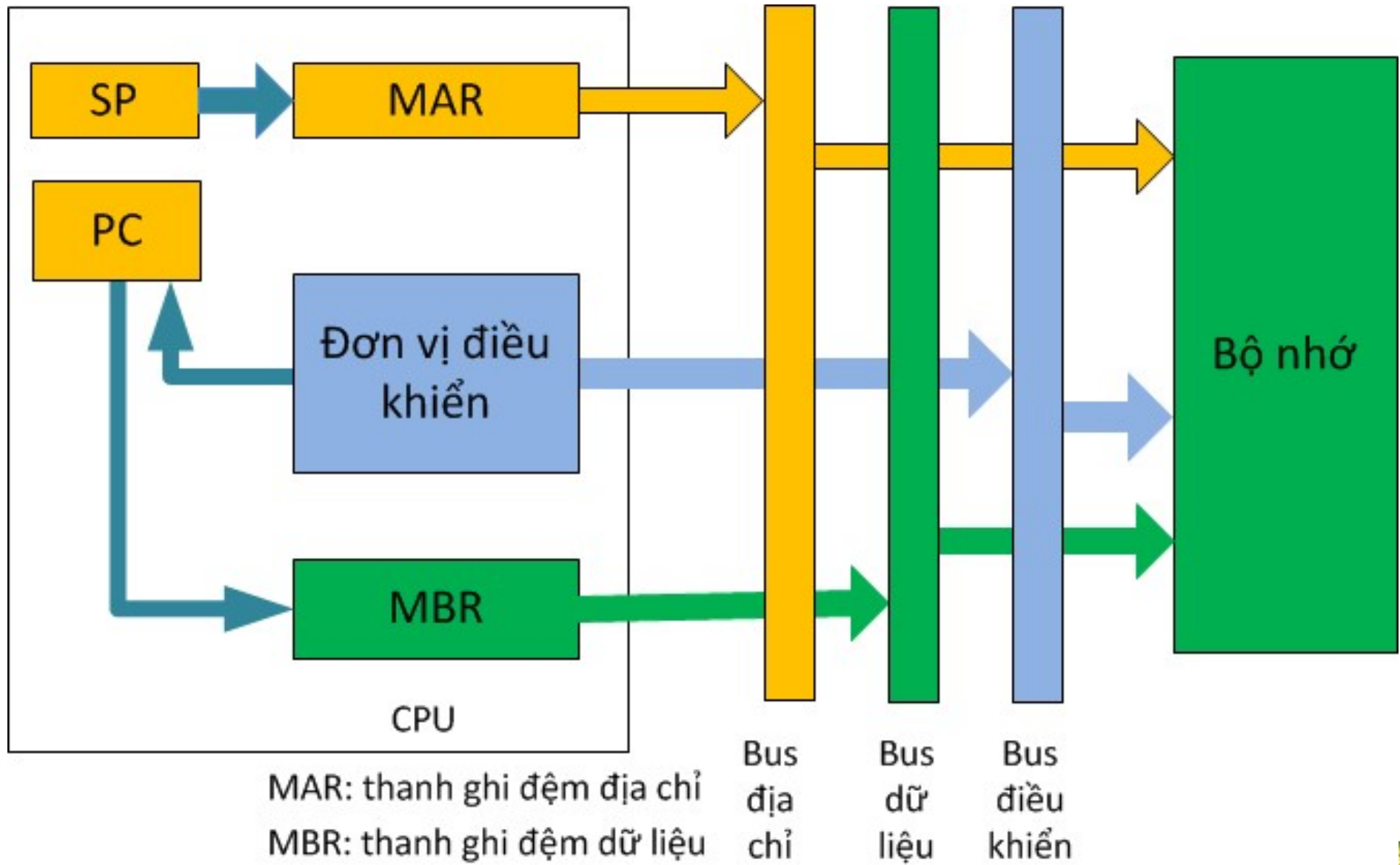


Ngắt

- ❑ Nội dung của bộ đếm chương trình PC (địa chỉ trở về sau khi ngắt) được đưa ra bus dữ liệu
- ❑ CPU đưa địa chỉ (thường được lấy từ con trỏ ngăn xếp SP) ra bus địa chỉ
- ❑ CPU phát tín hiệu điều khiển ghi bộ nhớ
- ❑ Địa chỉ trở về trên bus dữ liệu được ghi ra vị trí xác định (ở ngăn xếp)
- ❑ Địa chỉ lệnh đầu tiên của chương trình con điều khiển ngắt được nạp vào PC



Ngắt



Nội dung

- Tổ chức của CPU
- Thanh ghi
- Hoạt động của chu trình lệnh
- **Kỹ thuật pipeline**
- Giới thiệu chung về tập lệnh



Vấn đề của thực hiện lệnh tuần tự

- ❑ Các câu lệnh được thực hiện một cách tuần tự từ câu lệnh này đến câu lệnh khác.
- ❑ Không tận dụng được toàn bộ hiệu suất của hệ thống.
- ❑ Thời gian đáp ứng và throughput.



Kỹ thuật đường ống - pineline

- Ví dụ: quy trình có nhiều công đoạn.
 - Nhà máy SX sẽ đợi 1 sản phẩm hoàn chỉnh ra lò?
 - Hay từng công đoạn liên tục lấy sản phẩm vào để hoàn thiện?

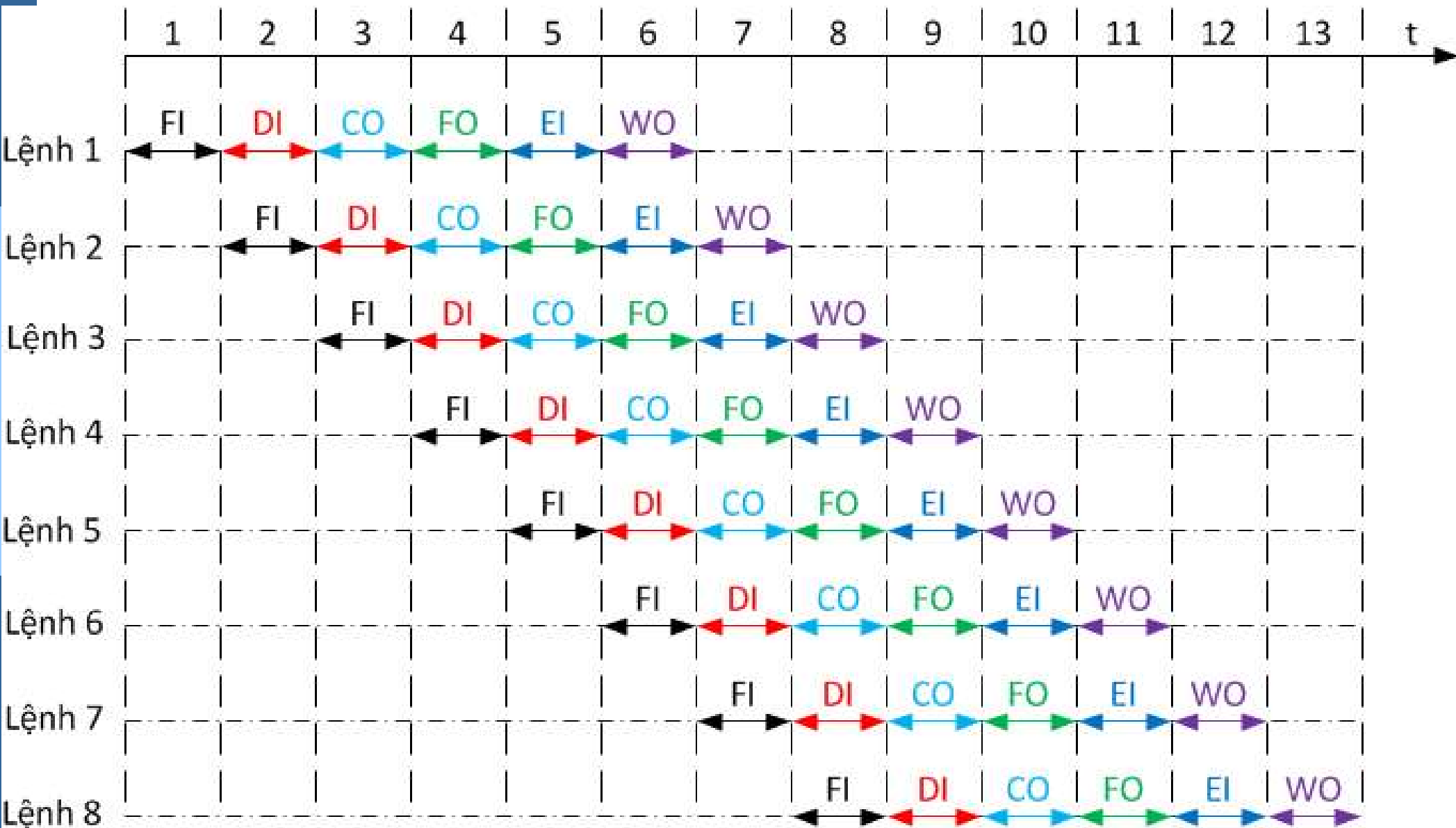


Kỹ thuật đường ống - pipeline

- ❑ Kỹ thuật đường ống lệnh (Instruction Pipelining): Chia chu trình lệnh thành các công đoạn và cho phép thực hiện gộp lên nhau (dây chuyền lắp ráp)
- ❑ Chẳng hạn có 6 công đoạn:
 - ❑ Nhận lệnh (Fetch Instruction - FI)
 - ❑ Giải mã lệnh (Decode Instruction - DI)
 - ❑ Tính địa chỉ toán hạng (Calculate Operand Address-CO)
 - ❑ Nhận toán hạng (Fetch Operands - FO)
 - ❑ Thực hiện lệnh (Execute Instruction - EI)
 - ❑ Ghi toán hạng (Write Operands - WO)



Biểu đồ thời gian của đường ống lệnh



Kỹ thuật đường ống - pipeline

- Giả sử thời gian để thực hiện một lệnh là 30s và mỗi công đoạn trong quá trình lệnh theo pipeline là 5s.
- Thực thi 8 lệnh ở trên:
 - Với hệ thống tuần tự từng lệnh 1 sẽ mất 240s
 - Với thực thi theo pipeline mất 65s
 - Nếu thực thi theo tuần tự, Lệnh cuối cùng được thực thi bắt đầu ở giây sau 210. Trong khi với pipeline, thời gian ở giây 35.
 - Với thực thi tuần tự, lệnh cuối cùng hoàn tất ở giây 240. Với pipeline, lệnh cuối cùng hoàn tất ở giây 65



Kỹ thuật đường ống - pipeline

- Giả sử thời gian để thực hiện một lệnh là 30s và mỗi công đoạn trong quá trình lệnh theo pipeline là 5s, các lệnh được sẵn sàng thực thi ở thời điểm ban đầu.
- Thực thi 8 lệnh ở ví dụ trên:
 - Tính thời gian xử lý trung bình (throughput)
 - ▶ Xử lý tuần tự: $(30+60+90+120+150+180+210+240)/8=135$
 - ▶ Xử lý theo pipeline: $(30+35+40+45+50+55+60+65)/8=47.5$
 - Tính thời gian đợi trung bình (response time)
 - ▶ Xử lý tuần tự: $(0+30+60+90+120+150+180+210)/8=105$
 - ▶ Xử lý theo pipeline: $(0+5+10+15+20+25+30+35)/8=17.5$



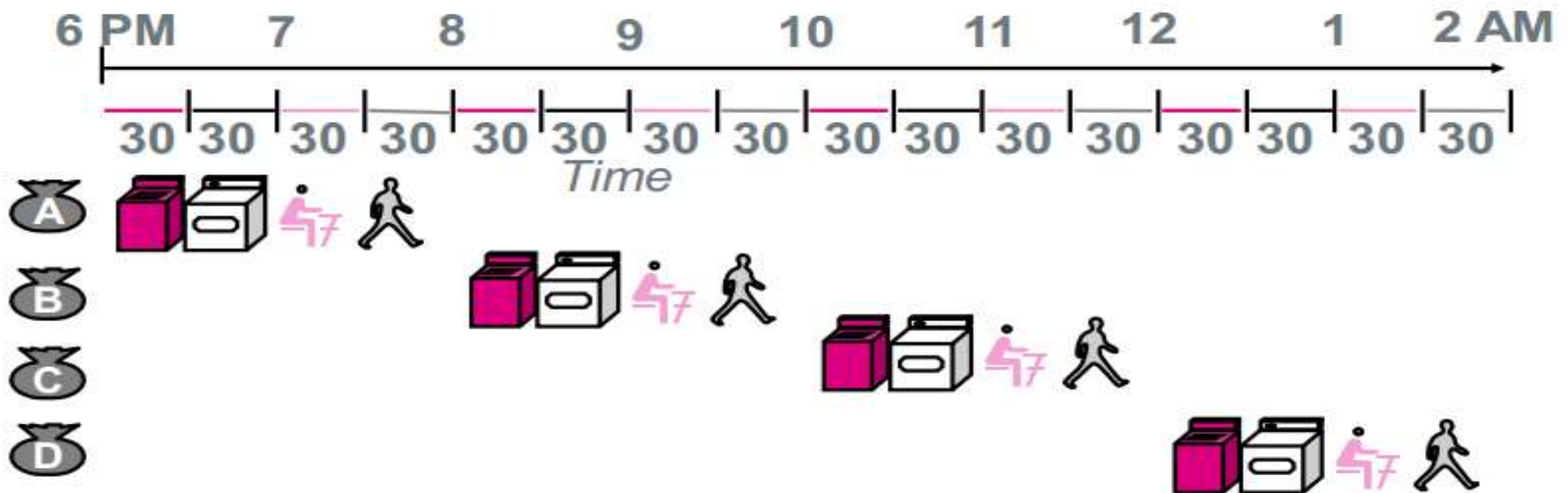
Pineline ví dụ

- An, Bình, Châu, Dũng cần giặt quần áo của từng người.
 - Máy giặt cần 30 phút.
 - Máy sấy cần 30 phút.
 - Ủi thẳng cần 30 phút.
 - Cắt vào ngăn tủ cần 30 phút.
- Chỉ có 1 máy giặt, 1 máy sấy, 1 bàn ủi cho mọi người.



Pineline ví dụ

□ Thực hiện tuần tự

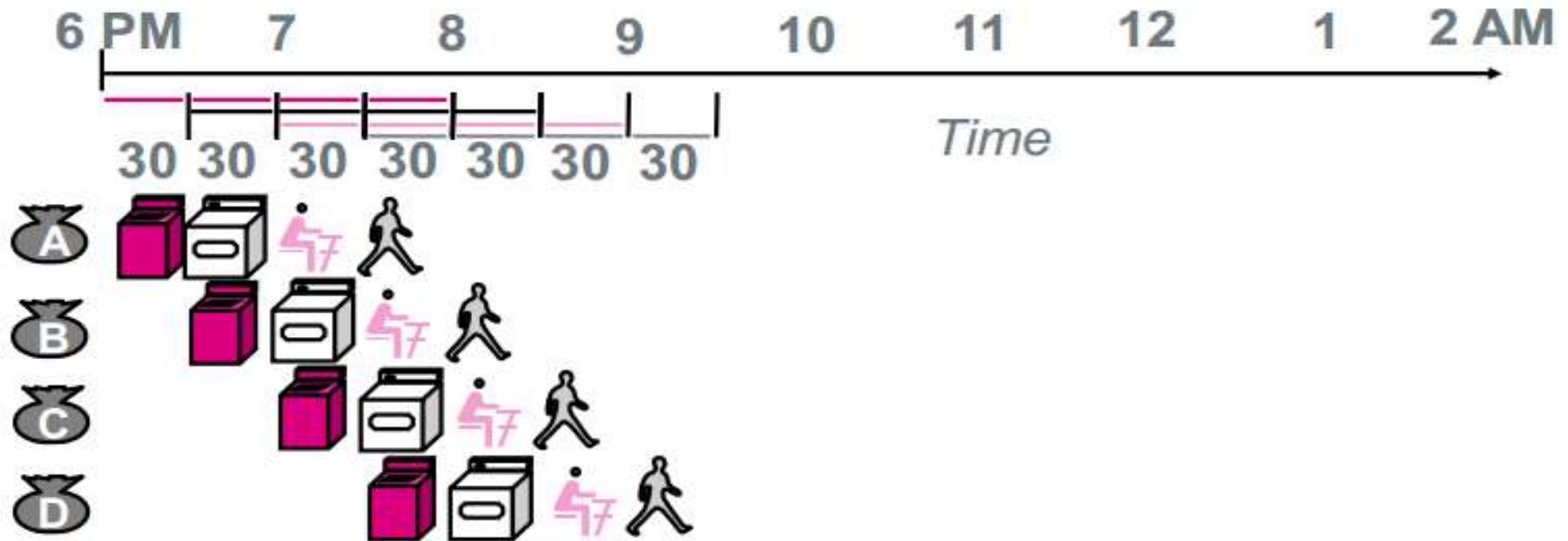


□ Tổng thời gian: 8 giờ



Pipeline ví dụ

Thực hiện theo pipeline



Tổng thời gian: 3.5 giờ



Giới thiệu chung về tập lệnh

- Mỗi bộ xử lý có một tập lệnh xác định
- Tập lệnh thường có hàng chục đến hàng trăm lệnh
- Mỗi lệnh là một chuỗi số nhị phân mà bộ xử lý hiểu được để thực hiện một thao tác xác định.
- Các lệnh được mô tả bằng các ký hiệu gọi nhớ dạng text → chính là các lệnh của hợp ngữ (assembly language)



Các thành phần của lệnh máy

Mã thao tác	Địa chỉ toán hạng
-------------	-------------------

- ❑ Mã thao tác (operation code → opcode): mã hóa cho thao tác mà bộ xử lý phải thực hiện
- ❑ Địa chỉ toán hạng: chỉ ra nơi chứa các toán hạng mà thao tác sẽ tác động
 - ❑ Toán hạng nguồn (source operand): dữ liệu vào của thao tác
 - ❑ Toán hạng đích (destination operand): dữ liệu ra của thao tác



Các kiểu thao tác thông dụng của tập lệnh

- Các lệnh chuyển dữ liệu
- Các lệnh xử lý số học
- Các lệnh xử lý logic
- Các lệnh chuyển điều khiển (rẽ nhánh, nhảy)



Định địa chỉ toán hạng

- Toán hạng của lệnh có thể là:
 - Một giá trị cụ thể nằm ngay trong lệnh
 - Nội dung của thanh ghi
 - Nội dung của ngăn nhớ hoặc cổng vào-ra
- Phương pháp định địa chỉ (addressing modes) là cách thức địa chỉ hóa trong trường địa chỉ của lệnh để xác định nơi chứa toán hạng



Các phương pháp định địa chỉ thông dụng

- Định địa chỉ tức thì
- Định địa chỉ thanh ghi
- Định địa chỉ trực tiếp
- Định địa chỉ gián tiếp qua thanh ghi
- Định địa chỉ dịch chuyển



Định địa chỉ tức thì

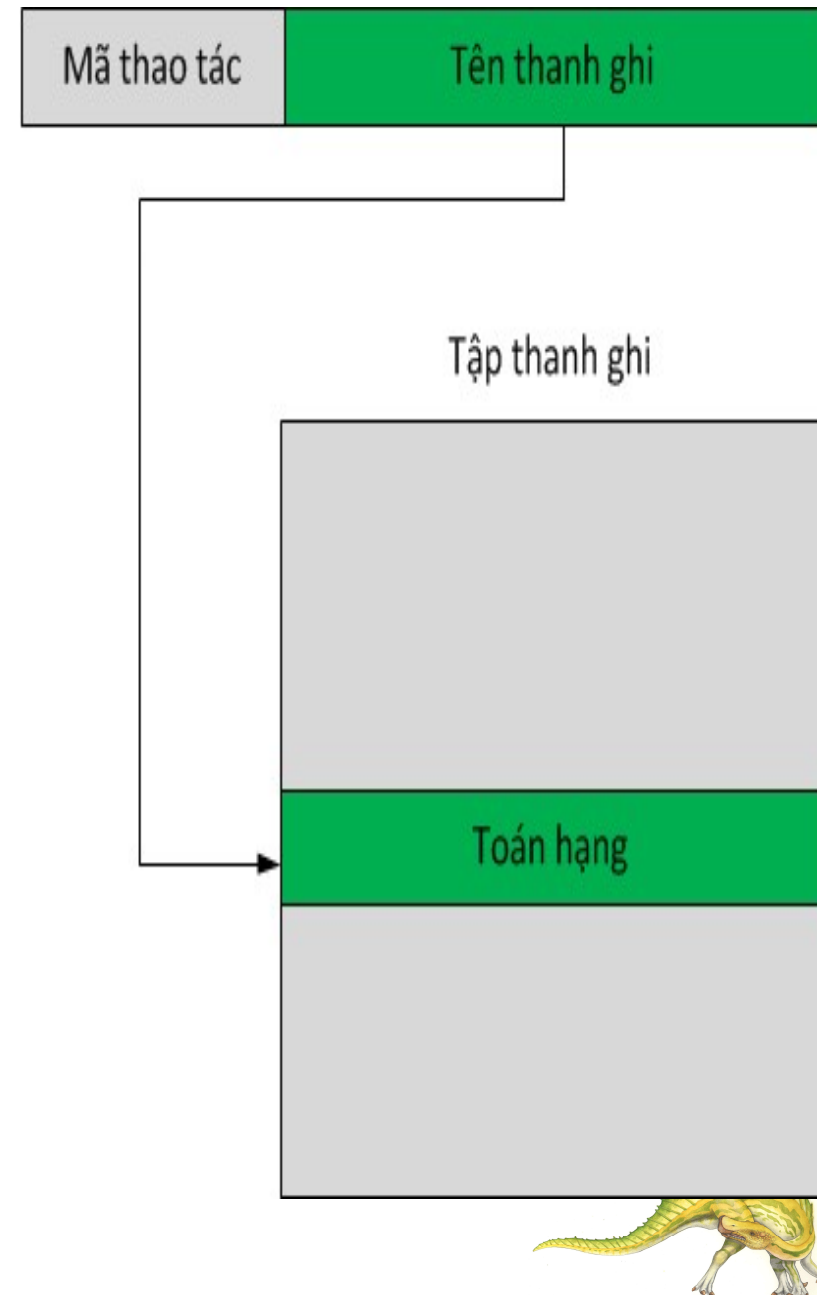
Mã thao tác		Toán hạng
-------------	--	-----------

- Toán hạng là hằng số nằm ngay trong lệnh
- Chỉ có thể là toán hạng nguồn
- Ví dụ:
 - `ADD R1, 5 ; R1 ← R1 + 5`
 - Không tham chiếu bộ nhớ
 - Truy nhập toán hạng rất nhanh
 - Dải giá trị của toán hạng bị hạn chế



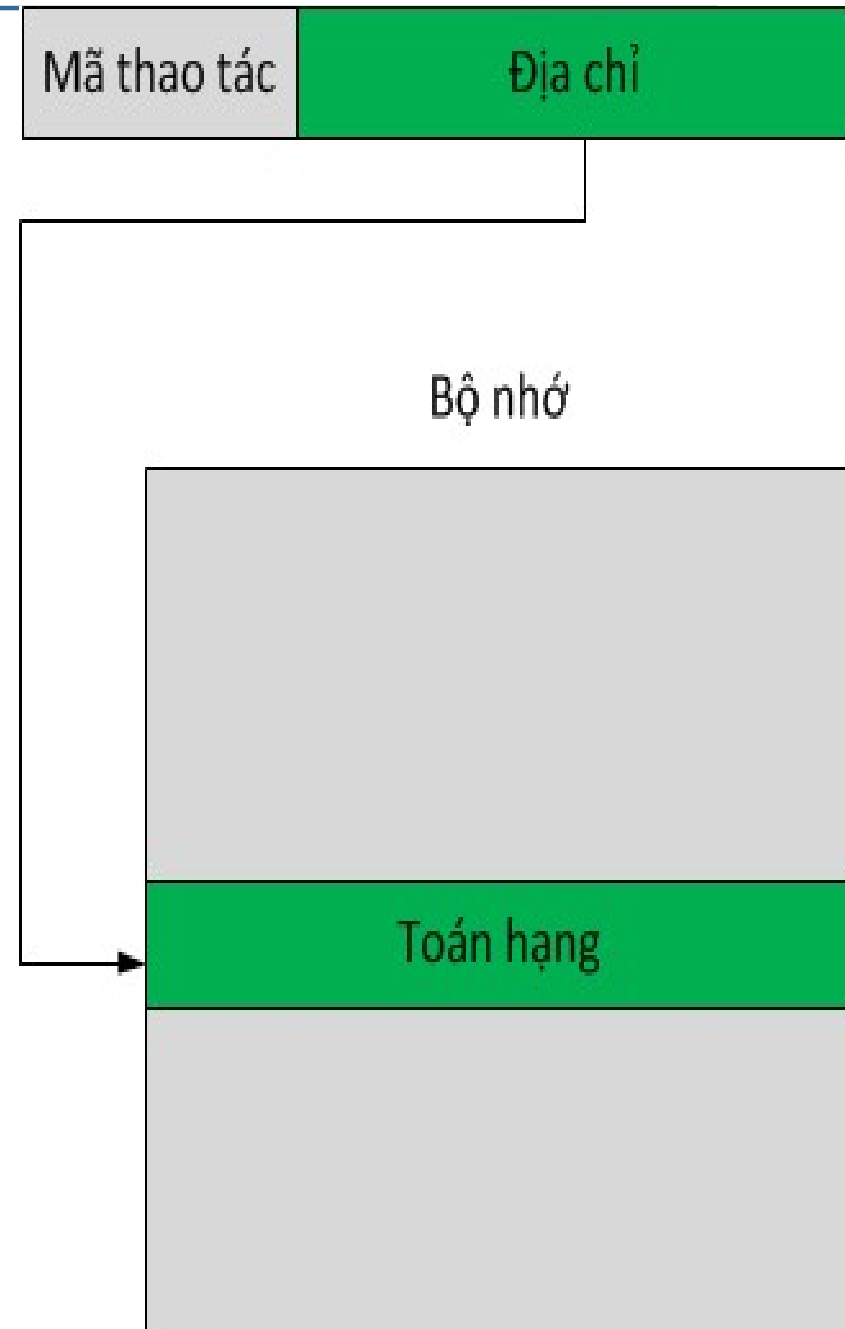
Định địa chỉ thanh ghi

- Toán hạng nằm trong thanh ghi có tên được chỉ ra trong lệnh
- Ví dụ:
 - $\text{ADD R1, R2 ; R1} \leftarrow \text{R1} + \text{R2}$
 - Số lượng thanh ghi ít \rightarrow Trường địa chỉ toán hạng chỉ cần ít bit
 - Không tham chiếu bộ nhớ
 - Truy hạng cập toán hạng nhanh
 - Tăng số lượng thanh ghi \rightarrow hiệu quả hơn



Định địa chỉ trực tiếp

- Toán hạng là ngăn nhớ có địa chỉ được cho trực tiếp trong lệnh
- Ví dụ:
 - $\text{ADD R1, A ; R1} \leftarrow \text{R1} + (\text{A})$
 - ▶ Cộng nội dung thanh ghi R1 với nội dung của ngăn nhớ có địa chỉ là A
 - ▶ Tìm toán hạng trong bộ nhớ ở địa chỉ A
- CPU tham chiếu bộ nhớ một lần để truy nhập dữ liệu

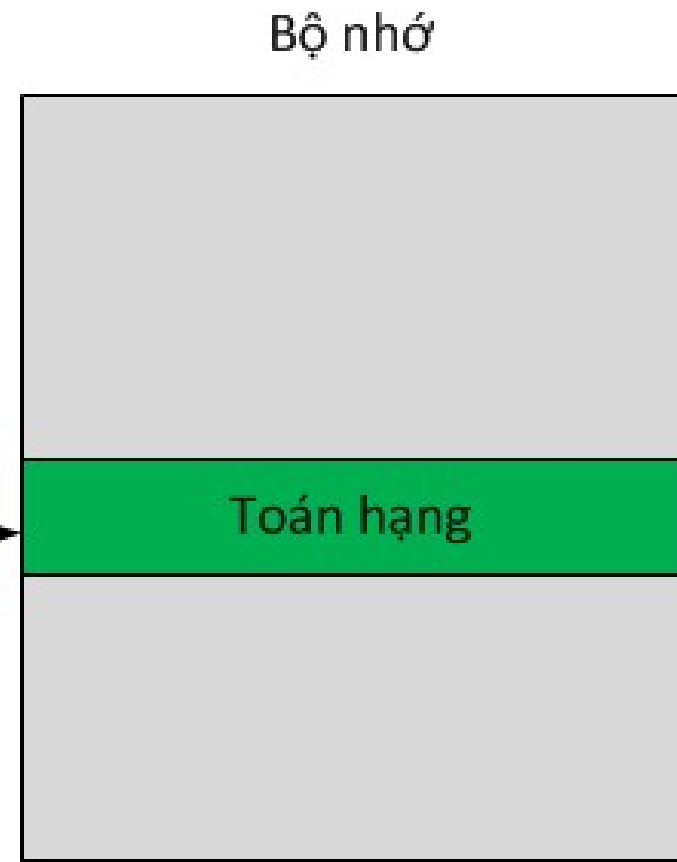
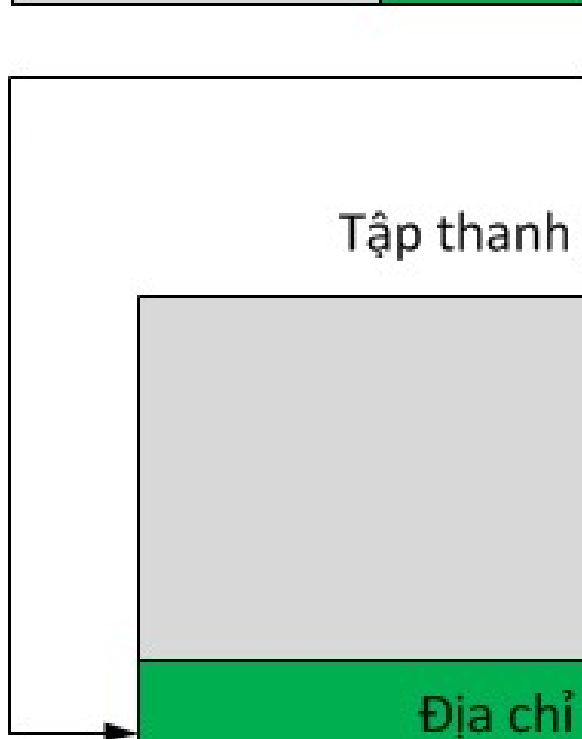


Định địa chỉ gián tiếp qua thanh ghi

- ❑ Toán hạng nằm ở ngăn nhớ có địa chỉ trong thanh ghi
- ❑ Trường địa chỉ toán hạng cho biết tên thanh ghi đó
- ❑ Thanh ghi có thể là ngầm định
- ❑ Thanh ghi này được gọi là thanh ghi con trỏ
- ❑ Vùng nhớ có thể được tham chiếu là lớn (2^n), (với n là độ dài của thanh ghi)



Định địa chỉ gián tiếp qua thanh ghi

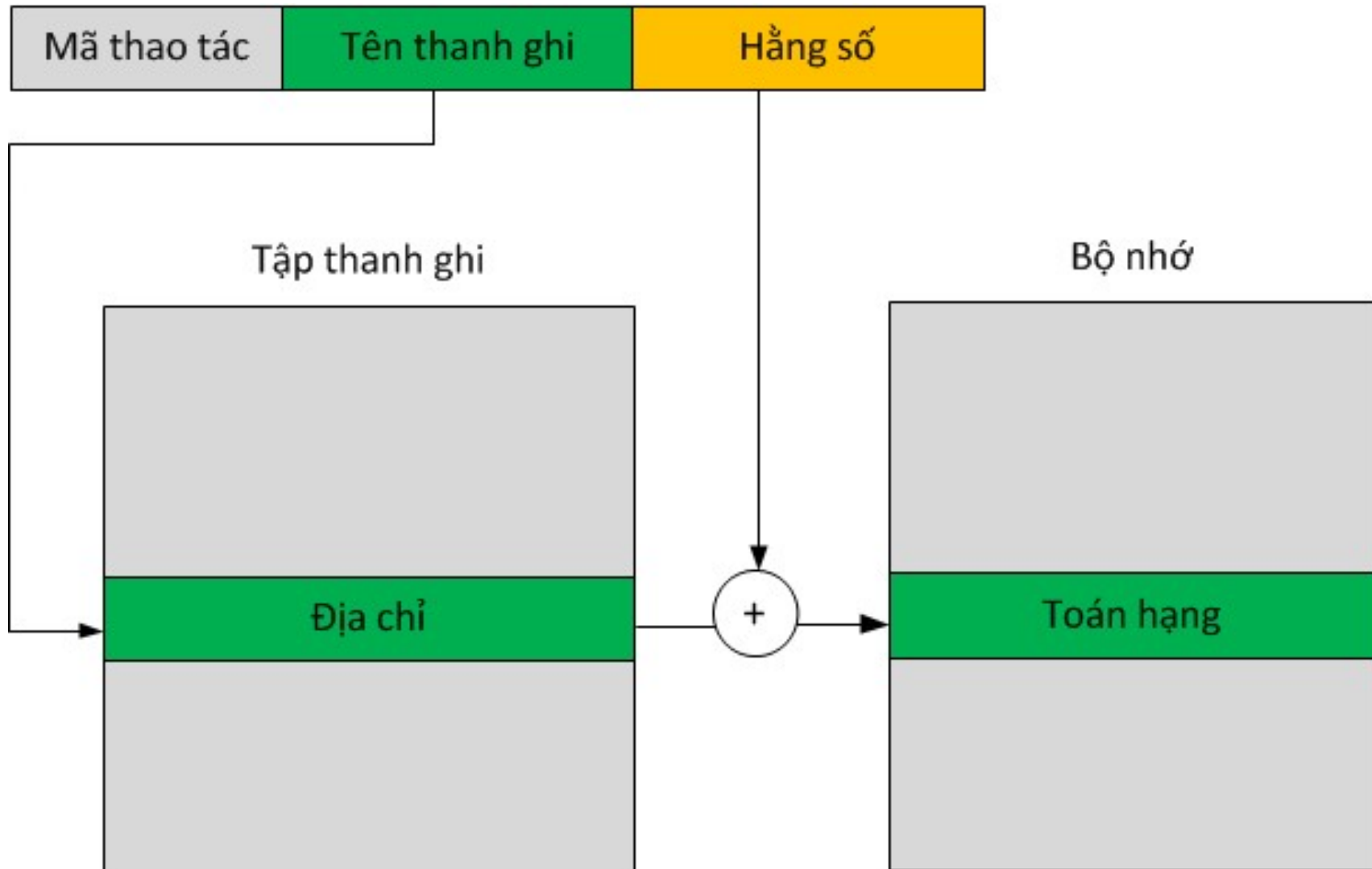


Định địa chỉ dịch chuyển

- Để xác định toán hạng, Trường địa chỉ chứa hai thành phần:
 - Tên thanh ghi
 - Hằng số (offset)
- Địa chỉ của toán hạng = nội dung thanh ghi + hằng số
- Thanh ghi có thể được ngầm định



Định địa chỉ dịch chuyển



Số lượng toán hạng trong lệnh

- Ba địa chỉ toán hạng:
 - 2 toán hạng nguồn, 1 toán hạng đích
 - $c = a + b$
 - Từ lệnh dài vì phải mã hoá địa chỉ cho cả ba toán hạng
 - Được sử dụng trên các bộ xử lý tiên tiến



Số lượng toán hạng trong lệnh (tt)

□ Hai địa chỉ toán hạng:

- Một toán hạng vừa là toán hạng nguồn vừa là toán hạng đích; toán hạng còn lại là toán hạng nguồn
- $a = a + b$
- Giá trị cũ của 1 toán hạng nguồn bị mất vì phải chứa kết quả
- Rút gọn độ dài từ lệnh
- Phổ biến



Số lượng toán hạng trong lệnh (tt)

- Một địa chỉ toán hạng:
 - Một toán hạng được chỉ ra trong lệnh
 - Một toán hạng là ngầm định → thường là thanh ghi (thanh ghi tích lũy –accumulator)
 - Được sử dụng trên các máy ở các thế hệ trước



Số lượng toán hạng trong lệnh (tt)

- 0 địa chỉ toán hạng:
 - Các toán hạng đều được ngầm định
 - Sử dụng Stack
 - Ví dụ:
 - ▶ push a
 - ▶ push b
 - ▶ add
 - ▶ pop c
 - ▶ Có nghĩa là : $c = a + b$
 - không thông dụng



Tổ chức bộ nhớ chính

- Bộ nhớ chính: phần 1 MB RAM đầu tiên (phần còn lại là bộ nhớ mở rộng)



Các kiểu dữ liệu thường dùng

- Bit : là đơn vị lưu trữ nhỏ nhất
- Byte : là đơn vị truy xuất của chương trình
- Word : đơn vị truy xuất của máy tính (có kích thước phụ thuộc vào CPU và lưu ngược theo đơn vị Byte)
- Chuỗi ký tự : lưu trữ theo thứ tự bình thường
- Số BCD: lưu trữ mỗi chữ số của 1 số thập phân bằng một (hoặc $\frac{1}{2}$) Byte



Nội dung

- Tổ chức của CPU
- Thanh ghi
- Hoạt động của chu trình lệnh
- Kỹ thuật pipeline
- Giới thiệu chung về tập lệnh



Tổ chức thanh ghi

+ Khái niệm thanh ghi:

- Là các vùng nhớ nhỏ lưu dữ liệu trong các chip xử lý
- Có tốc độ truy xuất nhanh và tần suất sử dụng cao

+ Kích thước thanh ghi:

Tính theo đơn vị bit – tùy thuộc chức năng của chip

VD: Các thanh ghi trong CPU 8 bit có kích thước 8 bit, trong CPU 16 bit có kích thước 16 bit, trong CPU 32 bit có kích thước 32 bit đồng thời có luôn các thanh ghi của CPU 16 bit

+ Số lượng thanh ghi :

Thường rất ít, tùy thuộc vào mức độ xử lý và thiết kế của chip.

VD: CPU Intel 16 bit có 14 thanh ghi, phân thành 5 nhóm



TỔ CHỨC THANH GHI CỦA CPU 16 bit

Nhóm thanh ghi đoạn (Segment register):

- CS (Code Segment): lưu chỉ số của segment chứa CT ngôn ngữ máy.
- DS (Data Segment): lưu chỉ số segment chứa dữ liệu của CT
- ES (Extra Segment): lưu chỉ số segment chứa dữ liệu bổ xung của CT
- SS (Stack Segment): lưu chỉ số segment chứa ngăn xếp của CT

(Trên CPU 32 bit còn có thêm 2 thanh ghi FS, GS có chức năng tương tự như ES)

** Chương trình muốn truy xuất 1 vùng nhớ thì phải đưa chỉ số segment của vùng nhớ đó vào 1 trong 4 thanh ghi trên*



TỔ CHỨC THANH GHI CỦA CPU 16 bit

Nhóm thanh ghi con trỏ và chỉ mục (Pointer & Index Reg.)

- IP (Instruction Pointer): lưu offset của ô nhớ chứa lệnh kế tiếp
- BP (Base Pointer): lưu offset của ô nhớ cần truy xuất
- SP (Stack Pointer): lưu offset đỉnh ngăn xếp.
- SI (Source Index): lưu offset vùng nhớ nguồn cần đọc lên
- DI (Destination Index): lưu offset vùng nhớ đích cần ghi xuống.

Mỗi thanh ghi trong nhóm này phải đi kèm với 1 thanh ghi trong nhóm thanh ghi đoạn mới biểu thị được vùng nhớ / ô nhớ cần truy xuất



TỔ CHỨC THANH GHI CỦA CPU 16 bit

- + Nhóm thanh ghi đa dụng (General Register)
 - AX (Accumulator Register): lưu các dữ liệu số, giá trị mặc định...
 - BX (Base Register): đóng vai trò chỉ số mảng, cũng có thể lưu trữ dữ liệu
 - CX (Count Register): có thể dùng để định số lần lặp
 - DX (Data Register): lưu dữ liệu/kết quả tính toán (~AX)

Mỗi thanh ghi trong nhóm này thường đều cho phép sử dụng đến từng Byte, bằng cách thay chữ “X” thành chữ “H” để chỉ Byte cao, hoặc “L” để chỉ Byte thấp (VD: AH, BL, CL...)



Bài tập

Xác định giá trị thanh ghi AX sau mỗi lệnh trong chuỗi lệnh

$AX = 2011h$

$AH = 20$

$AL = 11$

$AH = 20h$

$AL = 11h$



Bài tập 2

Xác định giá trị của DH, DL, DX sau mỗi lệnh trong chuỗi lệnh sau :

$$DX = 1234h + 5678$$

$$DL = 0$$

$$DH = 1$$

$$DL = DL - DH$$



TỔ CHỨC THANH GHI CỦA CPU 16 bit

Thanh ghi cờ (Flag Register)

Không có tên, mỗi bit là 1 cờ, biểu diễn trạng thái của lệnh vừa thực hiện hoặc đặt trạng thái cho lệnh thực hiện

