



HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

BÀI GIẢNG MÔN

KIẾN TRÚC MÁY TÍNH

ThS. Nguyễn Trọng Huân

Khoa Kỹ thuật Điện tử 2

2021

CHƯƠNG 2 CENTRAL PROCESSING UNIT (CPU)

Nội dung

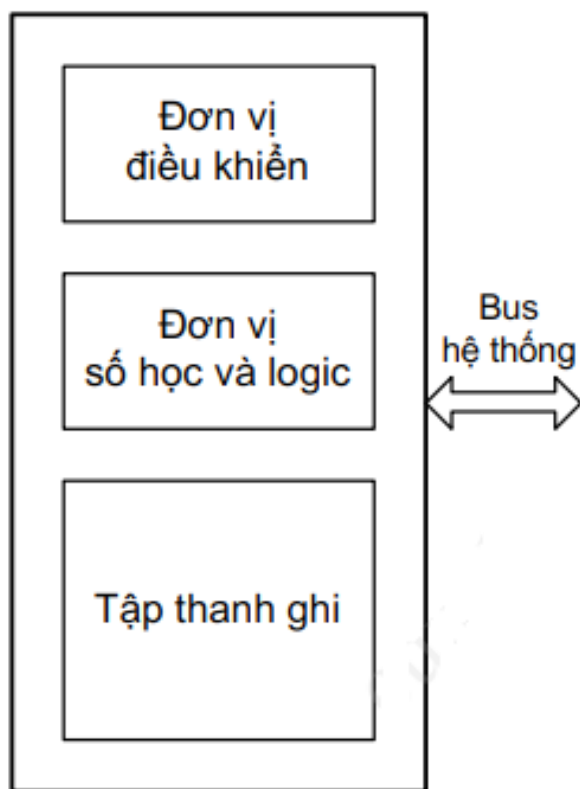
- 1. Sơ đồ khối tổng quát và các thành phần chức năng của CPU**
- 2. Khối logic và số học (ALU)**
- 3. Khối điều khiển**
- 4. Các thanh ghi**
- 5. Bus trong CPU**
- 6. Hiệu năng CPU**

1. SƠ ĐỒ KHỐI TỔNG QUÁT VÀ CÁC THÀNH PHẦN CHỨC NĂNG CỦA CPU

- Chức năng CPU:
 - Điều khiển hoạt động của máy tính
 - Xử lý dữ liệu
- Nguyên tắc hoạt động cơ bản:
CPU hoạt động theo chương trình nằm trong bộ nhớ chính.
- Là thành phần nhanh nhất trong hệ thống



CÁC THÀNH PHẦN CƠ BẢN CỦA CPU



➤ Đơn vị điều khiển - Control Unit (CU)

Điều khiển hoạt động của máy tính theo chương trình đã định sẵn

➤ Đơn vị số học và logic - Arithmetic and Logic Unit (ALU)

Thực hiện các phép toán số học và phép toán logic

➤ Tập thanh ghi - Register File (RF)

Gồm các thanh ghi chứa các thông tin phục vụ cho hoạt động của CPU

HOẠT ĐỘNG CỦA CPU

- **Nhận lệnh (fetch instruction):** đọc lệnh từ bộ nhớ.
- **Giải mã lệnh (decode instruction):** xác định tác vụ lệnh yêu cầu.
- **Nhận dữ liệu (fetch data):**
 - Nhận dữ liệu mới từ bộ nhớ hoặc các cổng I/O,
 - Xử lý dữ liệu (process data): thực hiện các phép toán số hoặc logic với các dữ liệu.
- **Ghi dữ liệu (write data):** ghi dữ liệu ra bộ nhớ hoặc các cổng I/O

HOẠT ĐỘNG CỦA CPU TRONG MỘT CHU KỲ LỆNH

- Nhận lệnh
- Giải mã lệnh
- Nhận toán hạng
- Thực hiện lệnh
- Cát toán hạng
- Ngắt

NHẬN LỆNH

- Bắt đầu mỗi chu trình lệnh, CPU nhận lệnh từ bộ nhớ chính
- Bộ đếm chương trình PC (Program Counter) là thanh ghi của CPU dùng để giữ địa chỉ của lệnh sẽ được nhận vào CPU phát ra địa chỉ từ bộ đếm chương trình PC tìm ra ngăn nhớ chứa lệnh
- Lệnh được đọc từ bộ nhớ đưa vào thanh ghi lệnh IR (Instruction Register)
- Sau khi lệnh được nhận vào, nội dung PC tự động tăng để trở đến lệnh kế tiếp.

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**

NHẬN DỮ LIỆU

➤ **Trực tiếp:**

- CPU đưa địa chỉ của toán hạng ra bus địa chỉ
- CPU phát tín hiệu điều khiển đọc
- Toán hạng được đọc vào CPU

➤ **Gián tiếp:**

- CPU đưa địa chỉ ra bus địa chỉ
- CPU phát tín hiệu điều khiển đọc
- Nội dung ngăn nhớ được đọc vào CPU, đó chính là địa chỉ của toán hạng (gián tiếp)
- Địa chỉ này được CPU phát ra bus địa chỉ để tìm ra toán hạng
- CPU phát tín hiệu điều khiển đọc
- Toán hạng được đọc vào CPU

THỰC THI LỆNH

- Tùy thuộc vào loại lệnh mà sử dụng ALU để tính :
- Đọc/Ghi bộ nhớ
 - Vào/Ra
 - Chuyển giữa các thanh ghi
 - Thao tác số học/logic
 - Chuyển điều khiển (rẽ nhánh)
 - ...

CẤT TOÁN HẠNG

- 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 (địa chỉ) xác định

NGẮT

➤ Chu trình lệnh có ngắt

- Nội dung của bộ đếm chương trình PC (địa chỉ trả 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 trả 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

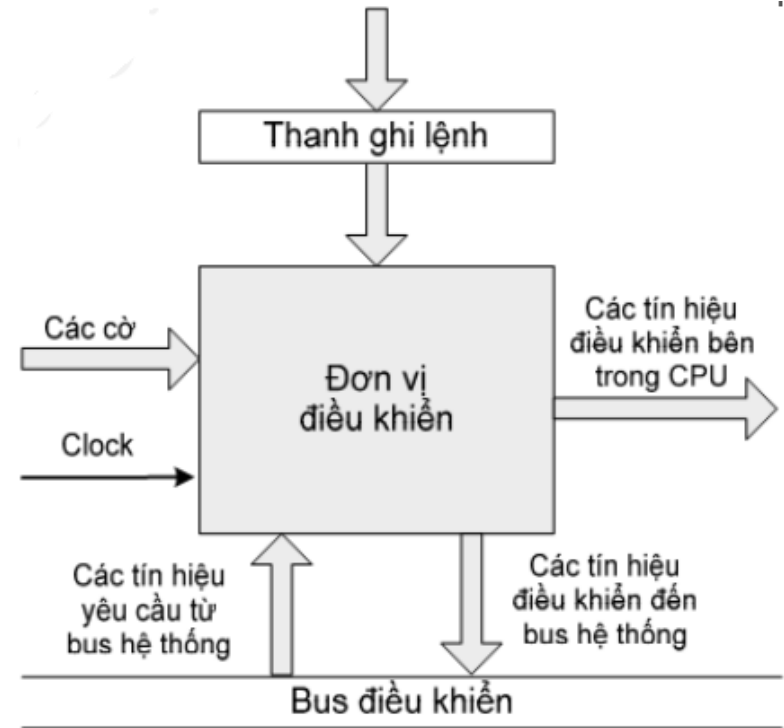
2. KHỐI LOGIC VÀ SỐ HỌC (ALU)

- Là thành phần quan trọng của CPU, thực hiện các phép toán luận lý và số học.
- Số học: cộng, trừ, nhân, chia, tăng, giảm, ...
- Logic: AND, OR, XOR, NOT, dịch bit...

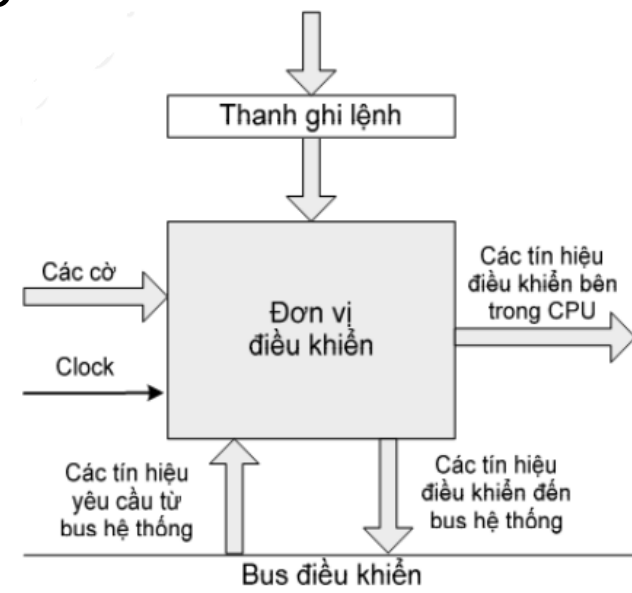


2. KHỐI ĐIỀU KHIỂN (CU)

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



- Clock: Tín hiệu xung nhịp từ mạch tạo dao động.
- Instruction register: Mã lệnh từ thanh ghi lệnh đưa đến để giải mã.
- Flags: Các cờ từ thanh ghi cờ cho biết trạng thái của CPU và kết quả các phép toán từ khối ALU trước đó.
- Control signals from control bus: Các tín hiệu yêu cầu từ bus điều khiển.
- Control signals within the processor: Điều khiển các thanh ghi, và điều khiển ALU
- Control signals to control bus: Điều khiển bộ nhớ và điều khiển các module I/O



3. CÁC THANH GHI (REGISTER)

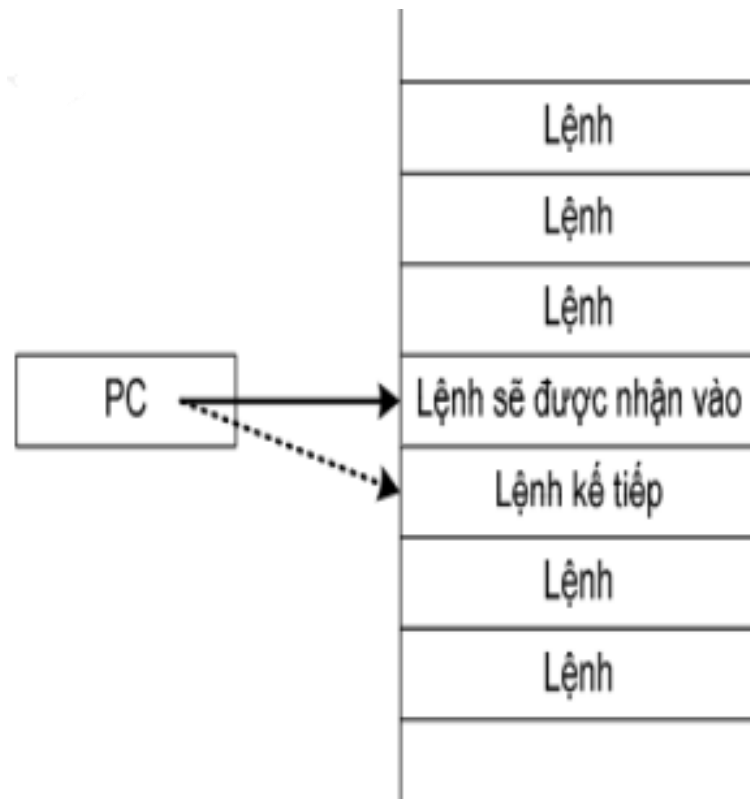
- Là các thanh ghi nằm trong CPU.
- Chứa các thông tin tạm thời phục vụ cho hoạt động ở thời điểm hiện tại của CPU.
- Được coi là mức đầu tiên của hệ thống nhớ.
- Số lượng thanh ghi nhiều → tăng hiệu năng của CPU.
- Có hai loại thanh ghi:
 - Các thanh ghi lập trình được
 - Các thanh ghi không lập trình được

Phân loại thanh ghi

- Thanh ghi địa chỉ: quản lý địa chỉ của ô nhớ hay cổng vào/ra.
- Bộ đếm chương trình PC (Program Counter)
- Con trỏ dữ liệu DP (Data Pointer)
- Con trỏ ngăn xếp SP (Stack Pointer)
- Thanh ghi cơ sở và thanh ghi chỉ số (Base Register & Index Register)
- Thanh ghi dữ liệu: chứa tạm thời các dữ liệu.
- Thanh ghi đa năng: có thể chứa địa chỉ hoặc dữ liệu.
- Thanh ghi điều khiển/trạng thái: chứa các thông tin điều khiển và trạng thái của CPU.
- Thanh ghi lệnh: chứa lệnh đang được thực hiện.

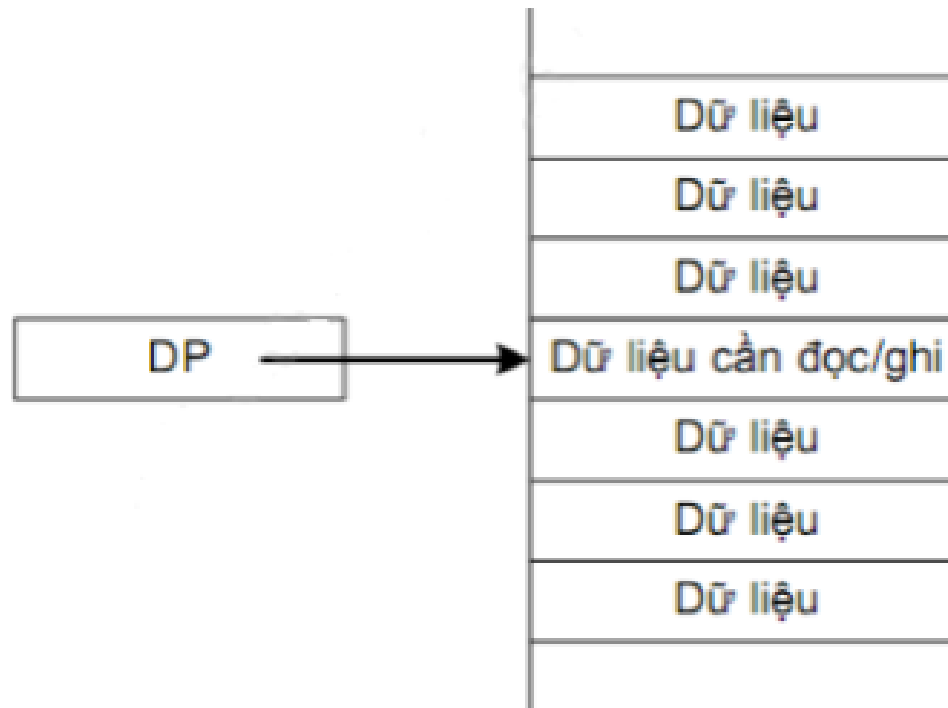
Bộ đếm chương trình PC (Program Counter)

- Còn được gọi là con trỏ lệnh IP
- Giữ địa chỉ của lệnh tiếp theo được nhận vào.
- Sau khi một lệnh được nhận vào, nội dung PC tự động tăng để trỏ sang lệnh kế tiếp.



Con trỏ dữ liệu DP (Data Pointer)

- Chứa địa chỉ của ngăn nhớ dữ liệu mà CPU muốn truy nhập

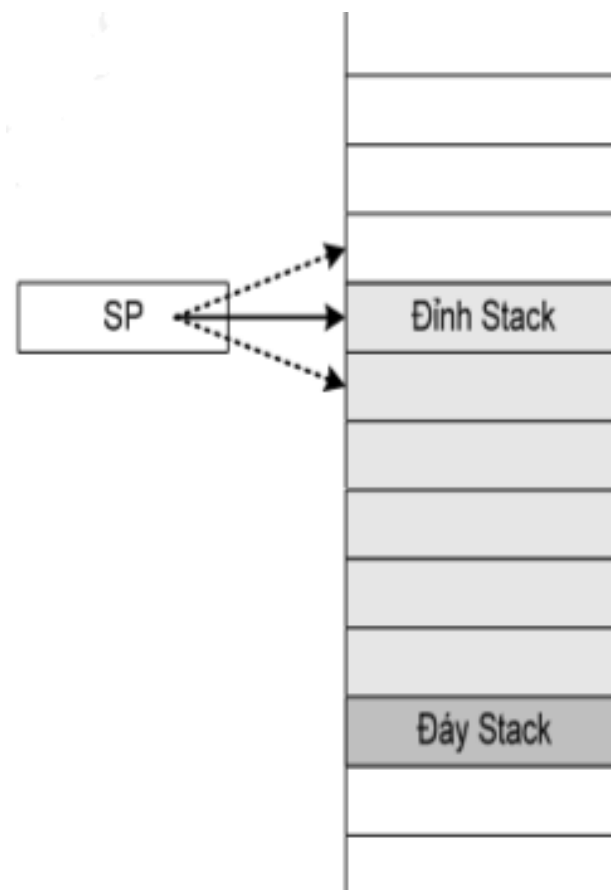


Ngăn xếp (Stack)

- Vùng nhớ có cấu trúc LIFO (Last In – First Out).
- Thường dùng để phục vụ chương trình con.
- Đáy ngăn xếp là một ô 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.

Con trỏ ngăn xếp (Stack Pointer)

- SP chứa địa chỉ của ngăn nhớ đỉnh ngăn xếp
- Khi cất một thông tin vào ngăn xếp: nội dung của SP tự động giảm, 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 tăng
- Khi ngăn xếp rỗng, SP trở vào đáy.



Thanh ghi cơ sở và thanh ghi chỉ số

- Thanh ghi cơ sở: chứa địa chỉ của ngăn nhớ cơ sở (địa chỉ cơ sở)
- Thanh ghi chỉ số: chứa độ lệch địa chỉ giữa ngăn nhớ mà CPU cần truy nhập so với ngăn nhớ cơ sở (chỉ số)
- Địa chỉ của ngăn nhớ cần truy nhập (vật lý) = địa chỉ cơ sở + chỉ số.



Thanh ghi dữ liệu

- Chứa các dữ liệu tạm thời hoặc các kết quả trung gian.
- Cần có nhiều thanh ghi dữ liệu.
- Các thanh ghi dữ liệu có thể chứa: 8, 16, 32, 64 bit.

Thanh ghi trạng thái

- Còn gọi là thanh ghi cờ (Flag Register)
- Chứa các thông tin trạng thái của CPU
- Các cờ phép toán: báo hiệu trạng thái của kết quả phép toán
- Các cờ điều khiển: biểu thị trạng thái điều khiển của CPU

Thanh ghi trạng thái

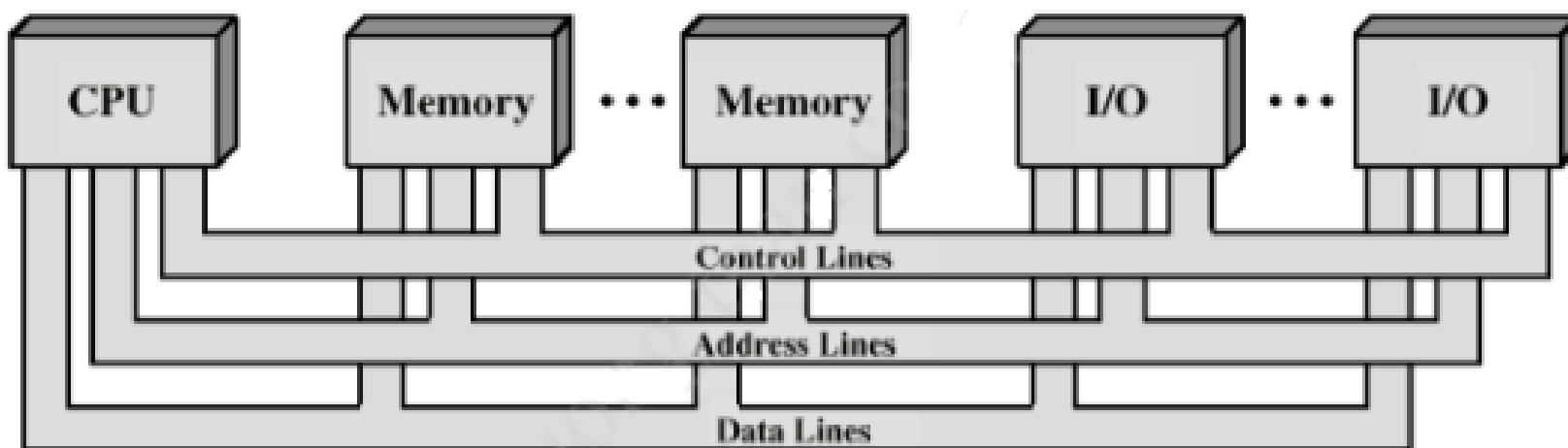
- Cờ phép toán
 - Cờ Zero (cờ rỗng): được thiết lập lên 1 khi kết quả của phép toán bằng 0.
 - Cờ Sign (cờ dấu): được thiết lập lên 1 khi kết quả phép toán nhỏ hơn 0
 - Cờ Carry (cờ nhớ): được thiết lập lên 1 nếu phép toán có nhớ ra ngoài bit cao nhất → cờ báo tràn với số không dấu.
 - Cờ Overflow (cờ tràn): được thiết lập lên 1 nếu cộng hai số nguyên cùng dấu mà kết quả có dấu ngược lại → cờ báo tràn với số có dấu.

- Cờ điều khiển:
 - Cờ Interrupt (Cờ cho phép ngắt)
Nếu $IF = 1 \rightarrow$ CPU ở trạng thái cho phép ngắt với tín hiệu yêu cầu ngắt từ bên ngoài gửi tới
Nếu $IF = 0 \rightarrow$ CPU ở trạng thái cấm ngắt với tín hiệu yêu cầu ngắt từ bên ngoài gửi tới
 - Cờ Direction (cờ hướng)
 - Cờ Trap
 - ...

5. HỆ THỐNG BUS TRONG CPU

➤ Data bus:

- Chuyển tải dữ liệu, lệnh.
- Độ rộng của Bus tương ứng khả năng chuyển tải dữ liệu của CPU: 8, 16, 32, 64 bit



➤ Address bus:

- Cung cấp địa chỉ nguồn và đích của dữ liệu cho CPU.
- Độ rộng Bus xác định khả năng nhớ của hệ thống. Ví dụ: 8088 dùng 16 bit địa chỉ => không gian địa chỉ là 64K.

➤ Control bus: truyền các tín hiệu điều khiển và định thời các thông tin

- Tín hiệu đọc/ghi bộ nhớ,
- Tín hiệu điều khiển ngắt,
- Tín hiệu điều khiển bus.

Ví dụ: Memory Read (MEMR): Tín hiệu điều khiển đọc dữ liệu từ một ngăn nhớ có địa chỉ xác định đưa lên bus dữ liệu.

Memory Write (MEMW): Tín hiệu điều khiển ghi dữ liệu có sẵn trên bus dữ liệu đến một ngăn nhớ có địa chỉ xác định

6. HIỆU NĂNG CPU

- Định nghĩa hiệu năng P (Performance):

Hiệu năng = $1/(\text{thời gian thực hiện})$ hay là: $P = 1/t$

- VD1: “Máy tính A nhanh hơn máy B k lần”

$$\rightarrow P_A / P_B = t_B / t_A = k$$

- VD2: Thời gian chạy chương trình:

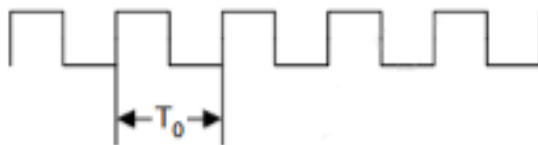
10s trên máy A, 15s trên máy B

$$t_B / t_A = 15s / 10s = 1.5$$

→ Vậy máy A nhanh hơn máy B 1.5 lần

Tốc độ xung nhịp CPU

- Về mặt thời gian, CPU hoạt động theo một xung nhịp (clock) có tốc độ xác định f_0



- Chu kỳ xung nhịp T_0 (Clock period): thời gian của một chu kỳ
- Số chu kỳ trong 1 giây: $f_0 = 1/T_0$

VD: Bộ xử lý có $f_0 = 4\text{GHz} = 4 \times 10^9\text{Hz}$

$$t_0 = 1/(4 \times 10^9) = 0.25 \times 10^{-9} \text{ s} = 0.25\text{ns}$$

Thời gian thực hiện của CPU

- Thời gian thực hiện của CPU (CPU time)
= Số chu kỳ xung nhịp x Thời gian một chu kỳ

$$t_{CPU} = n \times T_0 = \frac{n}{f_0}$$

n : số chu kỳ xung nhịp

- Tăng hiệu năng CPU bằng cách:
- Giảm số chu kỳ xung nhịp n
 - Tăng tốc độ xung nhịp f_0

Ví dụ: Hai máy tính A và B cùng chạy một chương trình

➤ Máy tính A:

- Tốc độ xung nhịp của CPU: $f_A = 2\text{GHz}$
- Thời gian CPU thực hiện chương trình: $t_A = 10\text{s}$

➤ Máy tính B:

- Thời gian CPU thực hiện chương trình: $t_B = 6\text{s}$
 - Số chu kỳ xung nhịp khi chạy chương trình trên máy B (n_B) nhiều hơn 1.2 lần số chu kỳ xung nhịp khi chạy chương trình trên máy A (n_A)
- Hãy xác định tốc độ xung nhịp cần thiết cho máy B (f_B)?

- Số chu kỳ xung nhịp khi chạy chương trình trên máy A:
$$nA = tA \times fA = 10s \times 2GHz = 20 \times 10^9$$
- Số chu kỳ xung nhịp khi chạy chương trình trên máy B:
$$nB = 1.2 \times nA = 24 \times 10^9$$
- Tốc độ xung nhịp cần thiết cho máy B

$$f_B = \frac{n_B}{t_B} = \frac{24 \times 10^9}{6} = 4 \times 10^9 Hz = 4GHz$$

Chu kỳ thực hiện chương trình

- Số chu kỳ xung nhịp của chương trình:

Số chu kỳ = Số lệnh của chương trình x Số chu kỳ trên một lệnh

$$n = IC \times CPI$$

n - số chu kỳ xung nhịp

IC - số lệnh của chương trình (Instruction Count)

CPI - số chu kỳ trên một lệnh (Cycles per Instruction)

- Thời gian thực hiện của CPU:

$$t_{CPU} = IC \times CPI \times T_0 = \frac{IC \times CPI}{f_0}$$

- Trong trường hợp các lệnh khác nhau có CPI khác nhau, cần tính CPI trung bình

➤ Ví dụ: Hai máy tính A và B có cùng kiến trúc tập lệnh

Máy tính A có:

Chu kỳ xung nhịp: $T_A = 250\text{ps}$

Số chu kỳ/ lệnh trung bình: $CPI_A = 2.0$

Máy tính B:

Chu kỳ xung nhịp: $T_B = 500\text{ps}$

Số chu kỳ/ lệnh trung bình: $CPI_B = 1.2$

Hãy xác định máy nào nhanh hơn ?

- Hai máy cùng kiến trúc tập lệnh, vì vậy số lệnh của cùng một chương trình trên hai máy là bằng nhau:

$$ICA = ICB = IC$$

- Thời gian thực hiện chương trình đó trên máy A và máy B:

$$t_A = ICA \times CPI_A \times T_A = IC \times 2.0 \times 250ps = IC \times 500ps$$

$$t_B = ICB \times CPI_B \times T_B = IC \times 1.2 \times 500ps = IC \times 600ps$$

- Từ đó ta có:

$$\frac{t_B}{t_A} = \frac{IC \times 600ps}{IC \times 500ps} = 1.2$$

- Kết luận: máy A nhanh hơn máy B 1.2 lần

➤ Tính CPI trung bình:

- Nếu loại lệnh khác nhau có số chu kỳ khác nhau, ta có tổng số chu kỳ

$$n = \sum_{i=1}^K (CPI_i \times IC_i)$$

- CPI trung bình:

$$CPI_{TB} = \frac{n}{IC} = \frac{1}{IC} \sum_{i=1}^K (CPI_i \times IC_i)$$

Ví dụ:

- Cho bảng chỉ ra các dãy lệnh sử dụng các lệnh thuộc các loại A, B, C. Tính CPI trung bình?

Loại lệnh	A	B	C
CPI theo loại lệnh	1	2	3
IC trong dãy lệnh 1	20	10	20
IC trong dãy lệnh 2	40	10	10

- Dãy lệnh 1: Số lệnh = 50

$$\text{Số chu kỳ} = 1 \times 20 + 2 \times 10 + 3 \times 20 = 100$$

$$\text{CPITB} = 100/50 = 2.0$$

- Dãy lệnh 2: Số lệnh = 60

$$\text{Số chu kỳ} = 1 \times 40 + 2 \times 10 + 3 \times 10 = 90$$

$$\text{CPITB} = 90/60 = 1.5$$

➤ **Hiệu năng phụ thuộc vào:**

- Thuật giải
- Ngôn ngữ lập trình
- Chương trình dịch
- Kiến trúc tập lệnh
- Phần cứng

Millions of Instructions Per Second (MIPS)

- MIPS: số triệu lệnh trong 1 giây cũng được xem là thước đo hiệu năng.

$$\text{MIPS} = \frac{\text{Instruction count}}{\text{Execution time} \times 10^6} = \frac{\text{Instruction count}}{\frac{\text{Instruction count} \times \text{CPI}}{\text{Clock rate}} \times 10^6} = \frac{\text{Clock rate}}{\text{CPI} \times 10^6}$$

$$\text{MIPS} = \frac{f_0}{\text{CPI} \times 10^6}$$

$$\text{CPI} = \frac{f_0}{\text{MIPS} \times 10^6}$$

➤ Ví dụ 1: Tính MIPS của bộ xử lý với:

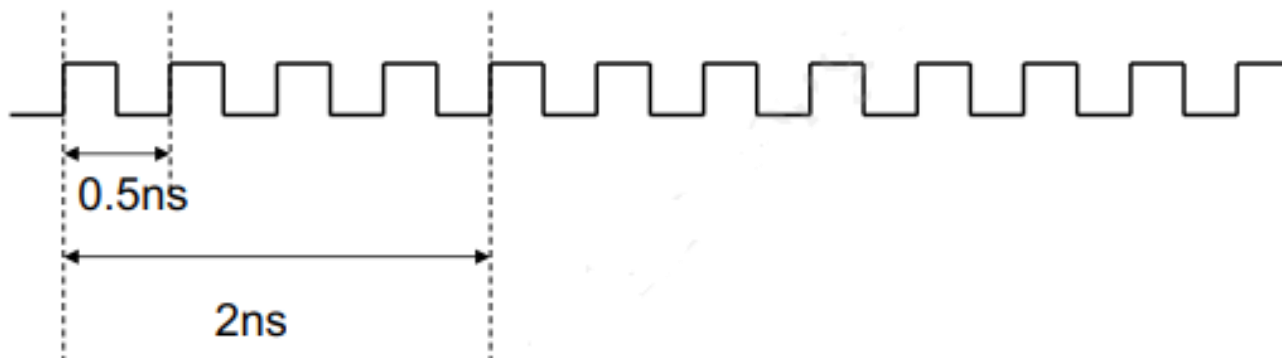
Clock rate = 2GHz và CPI = 4.

Chu kỳ $T_0 = 1/(2 \times 10^9) = 0.5\text{ns}$

CPI = 4 là thời gian thực hiện 1 lệnh = $4 \times 0.5\text{ns} = 2\text{ns}$

Số lệnh thực hiện trong 1s = $(10^9 \text{ ns})/(2\text{ns}) = 5 \times 10^8$ lệnh

Vậy bộ xử lý thực hiện được 500 MIPS



➤ Ví dụ 2: Tính CPI của bộ xử lý với:

Clock rate = 1GHz và 400 MIPS.

Chu kỳ $T_0 = 1/(10^9) = 1\text{ns}$

Số lệnh thực hiện trong 400 MIPS = 4×10^8 lệnh

Thời gian thực hiện 1 lệnh: $1/(4 \times 10^8)\text{s} = 2.5\text{ns}$

Vậy $\text{CPI} = 2.5$

HẾT CHƯƠNG 2