



COMPUTER ENGINEERING

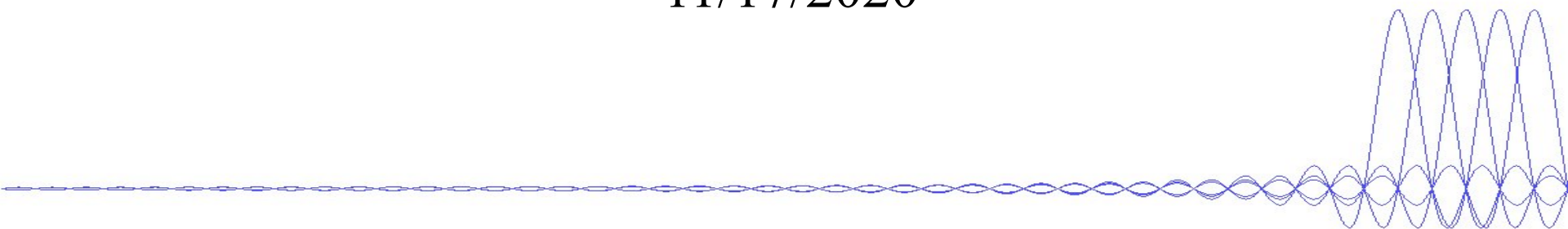


**UIT**  
TRƯỜNG ĐẠI HỌC  
CÔNG NGHỆ THÔNG TIN

# TỔ CHỨC VÀ CẤU TRÚC MÁY TÍNH II

## Chương 8 Bộ xử lý

11/17/2020



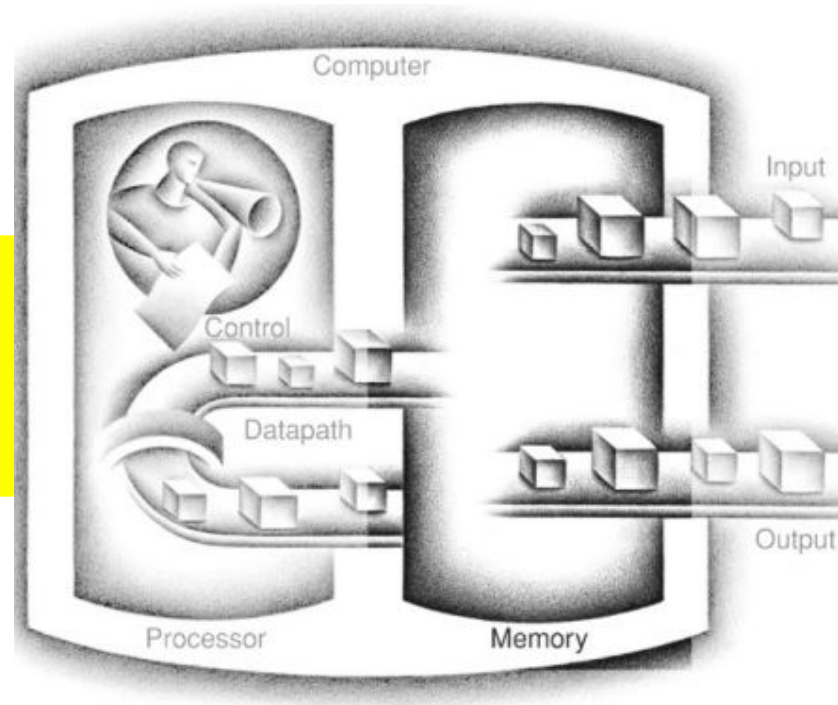


- Vi kiến trúc
- Datapath
- Thực thi lệnh
- Bài tập




# Vi kiến trúc (1/2)

- Kiến trúc Máy tính bao gồm 3 thành phần chính:
  - Kiến trúc tập lệnh (ISA): Quy định máy tính có thể làm những việc gì?
    - Lệnh
  - Vi kiến trúc (Tổ chức Phần cứng Máy tính): Quy định máy tính làm việc như thế nào?
    - Hiện thực ISA
  - Hệ thống Máy tính: Quy định các thành phần của máy tính phối hợp trong một hệ thống điện toán như thế nào?
    - Ảo hóa, Quản lý Bộ nhớ, Xử lý Đồ họa...





## Vi kiến trúc (2/2)

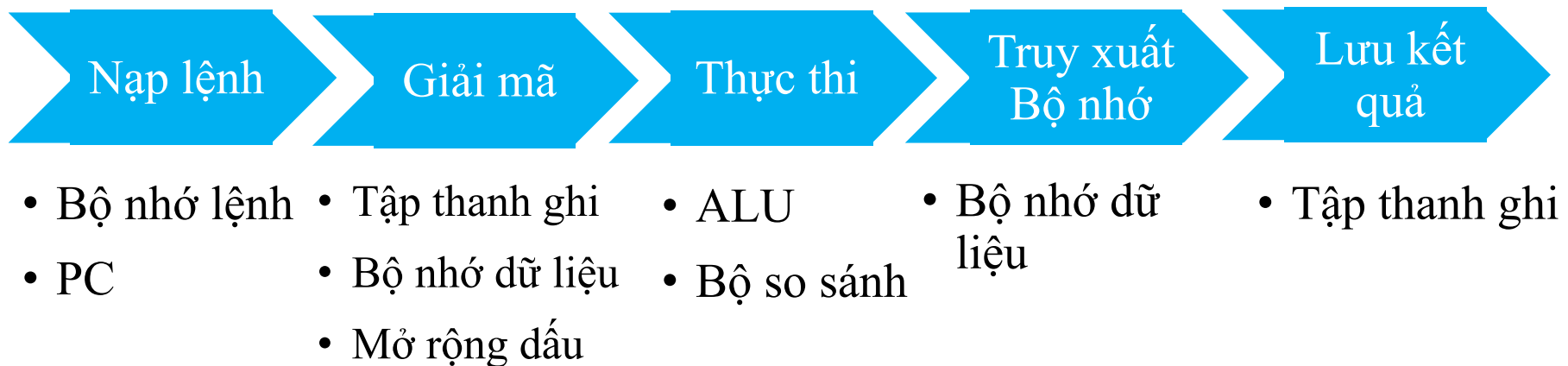
- Về chức năng, Vi kiến trúc là một tổ chức phần cứng dùng để hiện thực tập lệnh của một máy tính.
- Về cấu tạo, Vi kiến trúc được chia thành 2 khối:
  - Khối đường dữ liệu (datapath): Thực thi lệnh
    - Lưu trữ: Bộ nhớ lệnh, Bộ nhớ dữ liệu, Tập thanh ghi, ...
    - Truyền/nhận: Các đường tín hiệu dữ liệu, địa chỉ, điều khiển
    - Xử lý: ALU, Bộ so sánh, Mux, Bộ mở rộng dấu, Bộ dịch, ...
  - Khối điều khiển (control unit): Điều khiển datapath hoạt động
    - Dựa trên opcode của lệnh và trạng thái của datapath



# Datapath (1/9) – Chu kỳ thực thi lệnh

- Datapath dùng để thực thi lệnh! Một lệnh thực thi như thế nào?

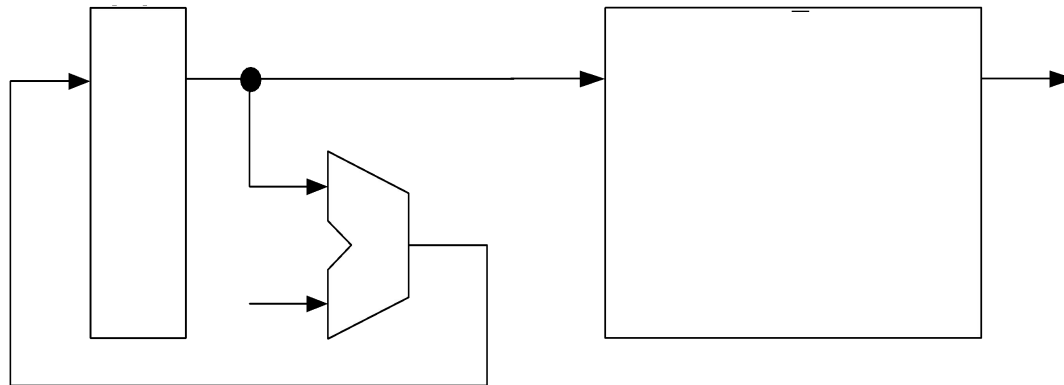
□ Chu kỳ thực thi lệnh!





## Datapath (2/9) – Nạp lệnh

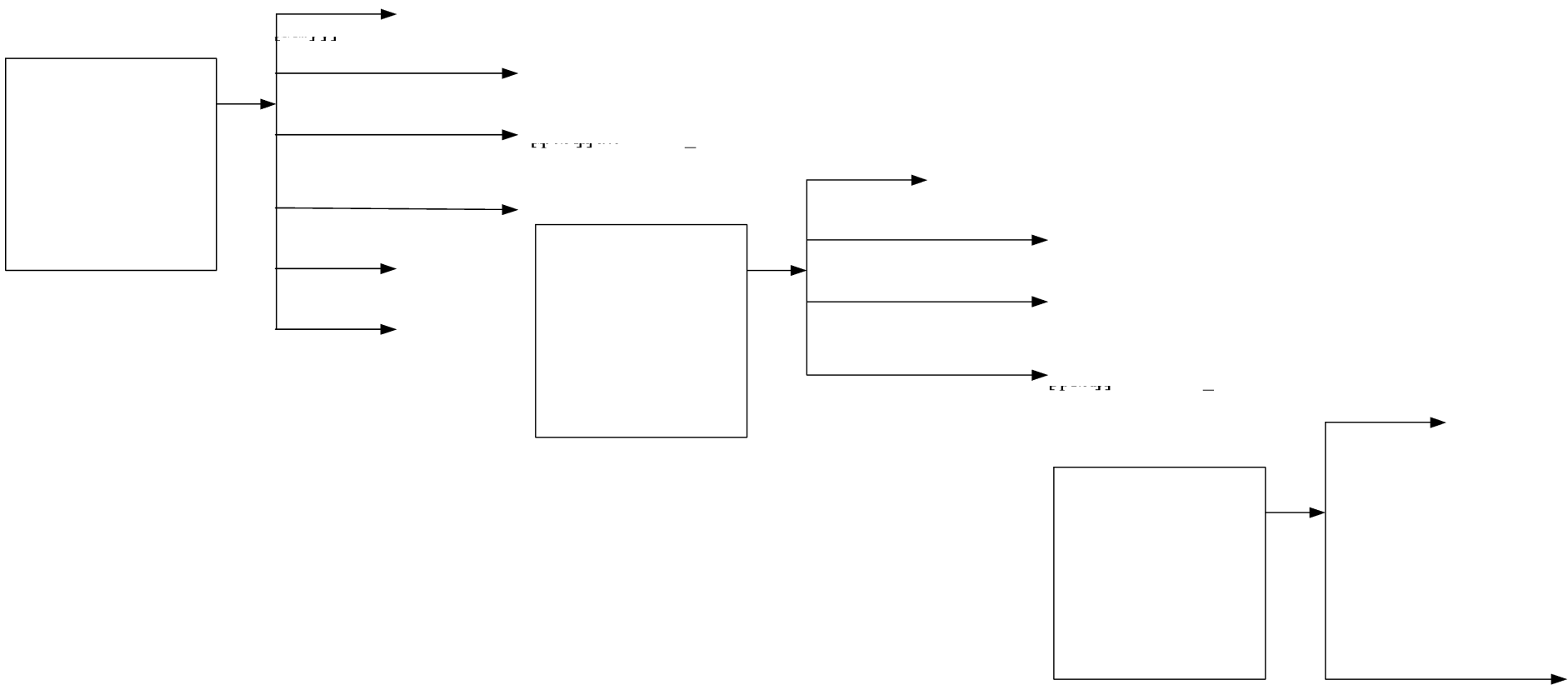
- Lệnh cần nạp lưu trong Bộ nhớ lệnh
- Địa chỉ của lệnh cần nạp lưu trong thanh ghi PC
  - Tăng PC lên 4 để chuẩn bị nạp lệnh tiếp theo -> cần thêm bộ cộng





# Datapath (3/9) – Giải mã lệnh – Định dạng lệnh

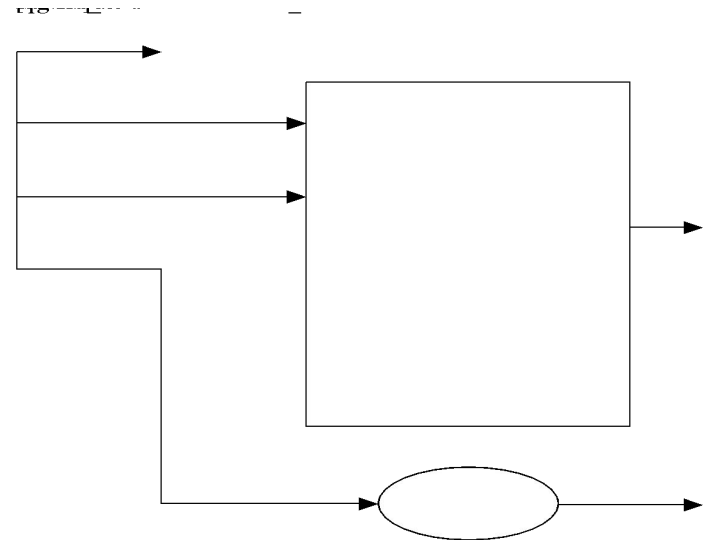
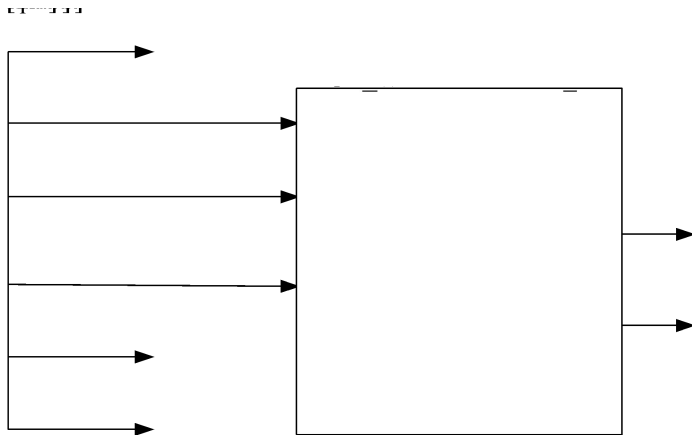
- Dựa vào opcode để xác định định dạng lệnh





# Datapath (4/9) – Giải mã lệnh - Nạp toán hạng

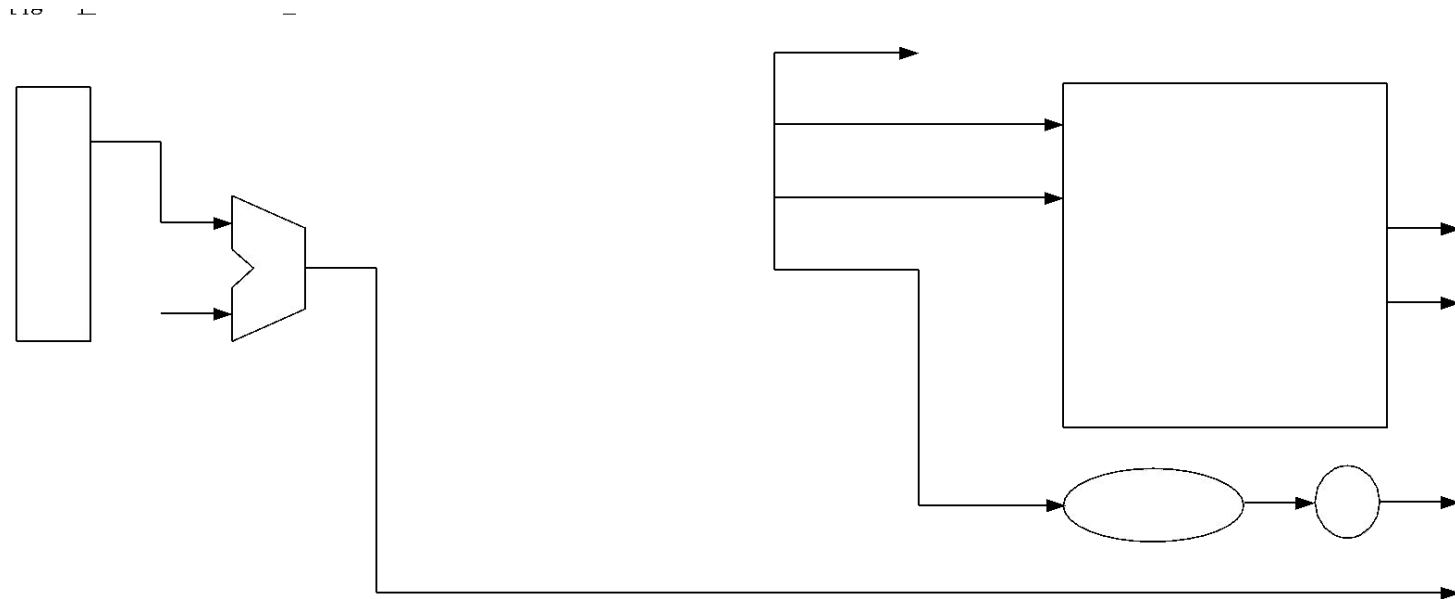
- Dựa vào định dạng lệnh mà nạp toán hạng tương ứng





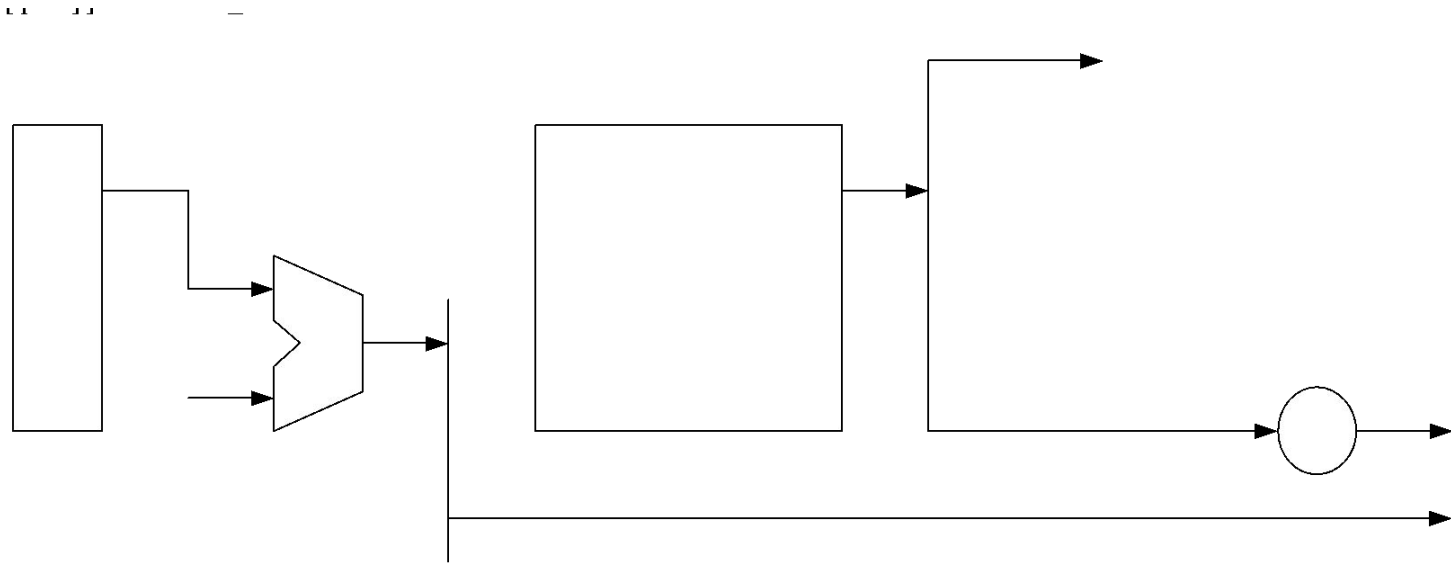


# Datapath (5/9) – Nạp toán hạng: beq/bne





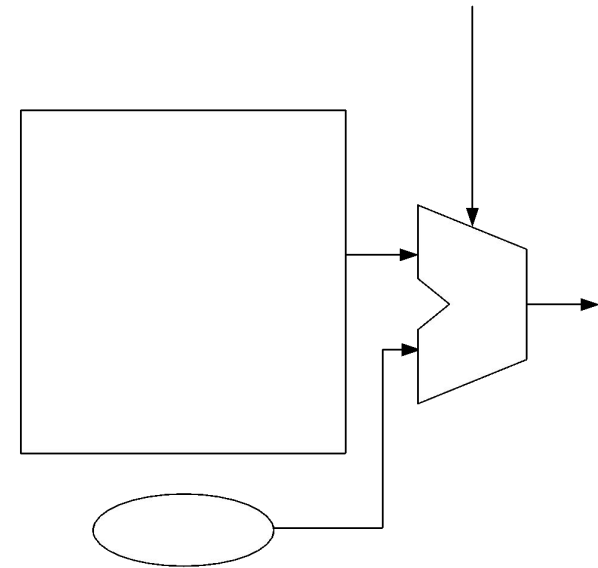
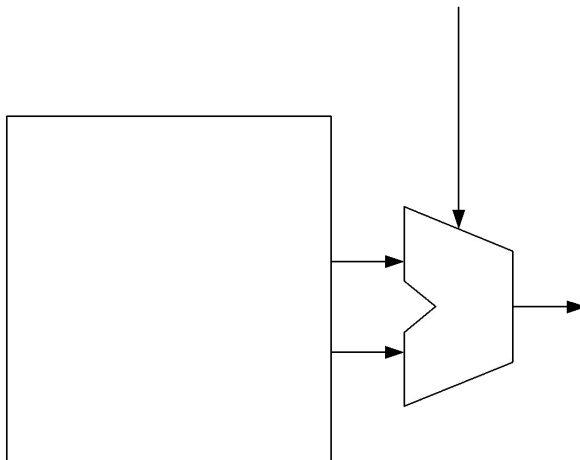
# Datapath (6/9) – Nạp toán hạng: j





# Datapath (7/9) – Thực thi

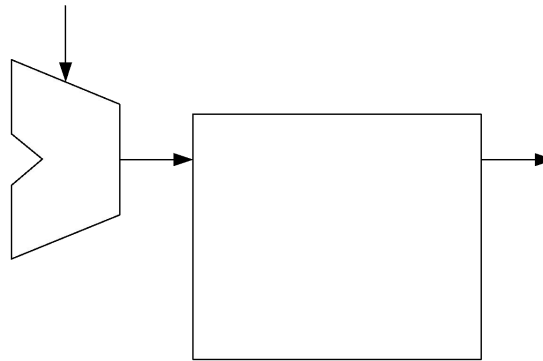
- Dựa vào opcode và funct để quyết định thao tác gì sẽ được thực thi



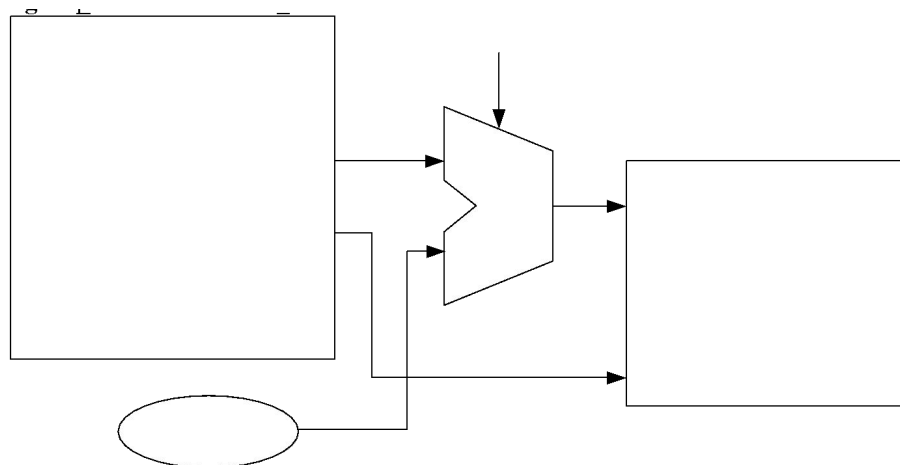


# Datapath (8/9) – Truy xuất bộ nhớ

- Đọc dữ liệu từ bộ nhớ dữ liệu đối với các lệnh nạp



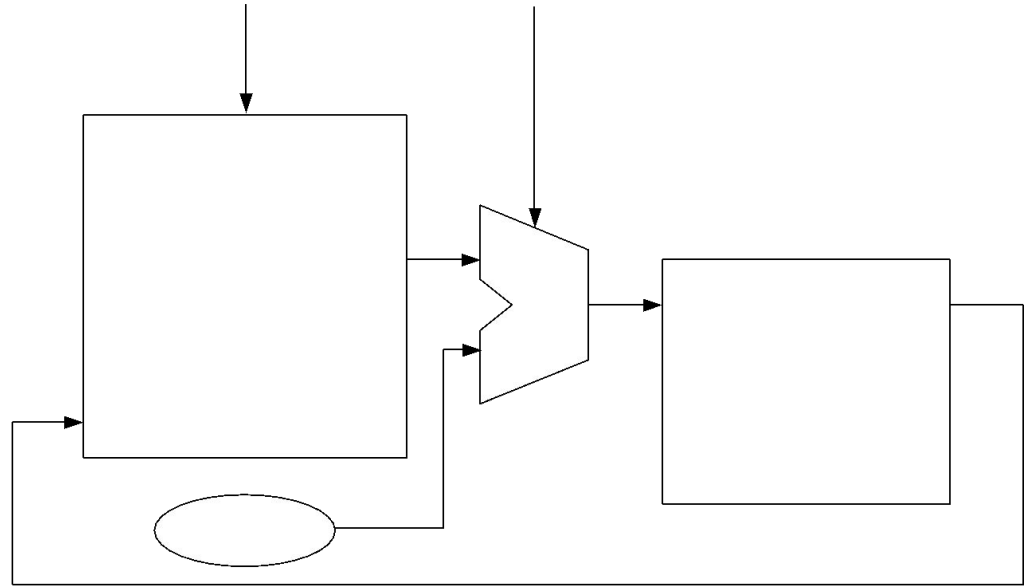
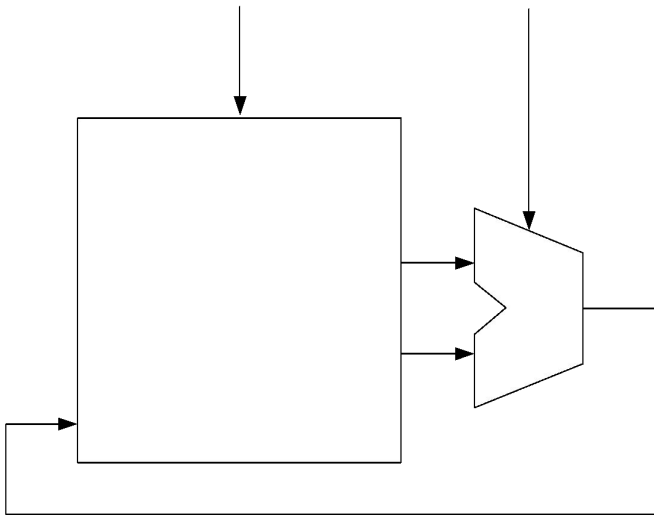
- Ghi dữ liệu tới bộ nhớ dữ liệu đối với các lệnh lưu





# Datapath (9/9) – Lưu kết quả

- Có thể ghi dữ liệu về lại Tập thanh ghi





# Thực thi nhóm lệnh luận lý & số học (1/2)

ALU\_Inst C, A, B

Nạp lệnh

$\text{Inst} = \text{IM}[\text{PC}]$

$\text{PC} = \text{PC} + 4$

Giải mã

$A = R[\text{rs}]$

$B = R[\text{rt}]$

$C = R[\text{rd}]$

Thực thi

$\text{ALU} = A \text{ op } B$

Truy xuất  
Bộ nhớ

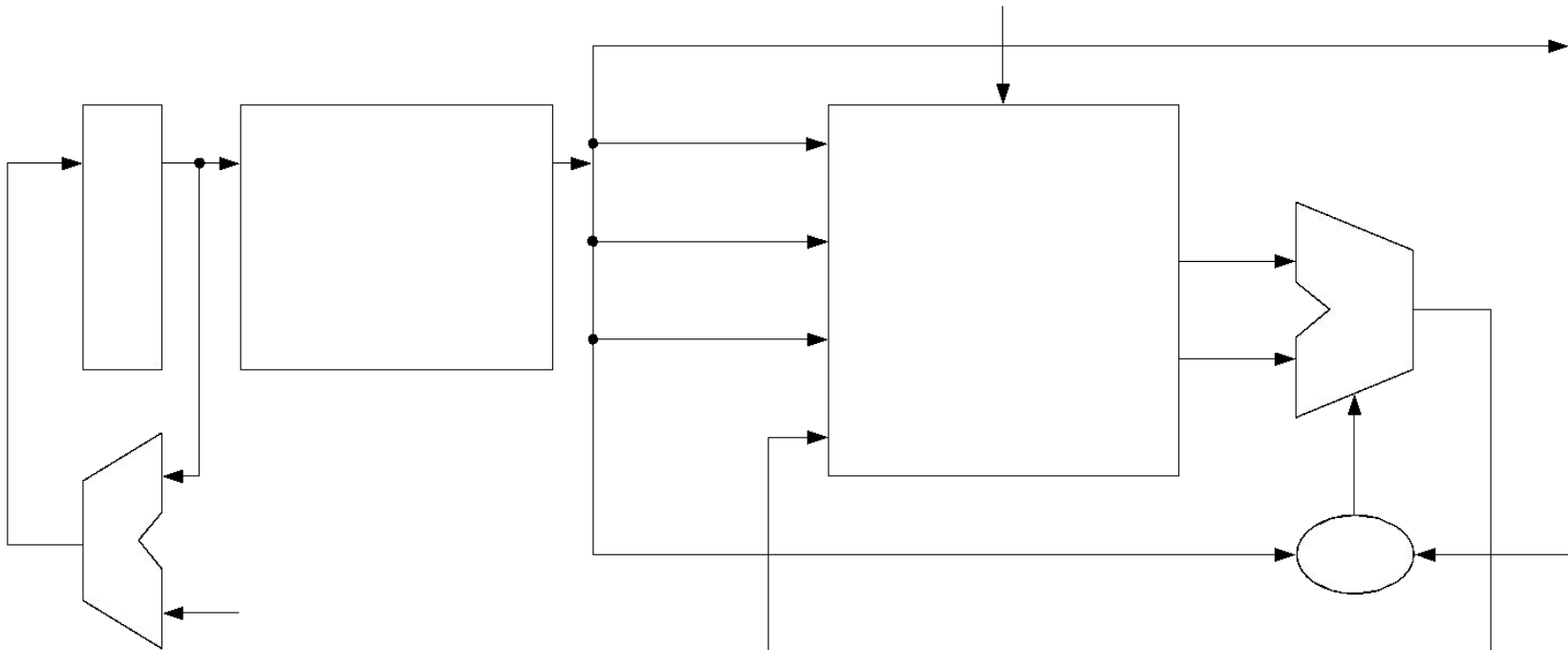
Lưu kết  
quả

$C = \text{ALU}$



# Thực thi nhóm lệnh luận lý & số học (2/2)

add, sub, and, or, slt





# Thực thi nhóm lệnh truyền dữ liệu (1/5) - lw

lw C, B(A)



$Inst = IM[PC]$

$A = R[rs]$

$ALU = A + B$

$D = DM[ALU]$

$C = D$

$PC = PC + 4$

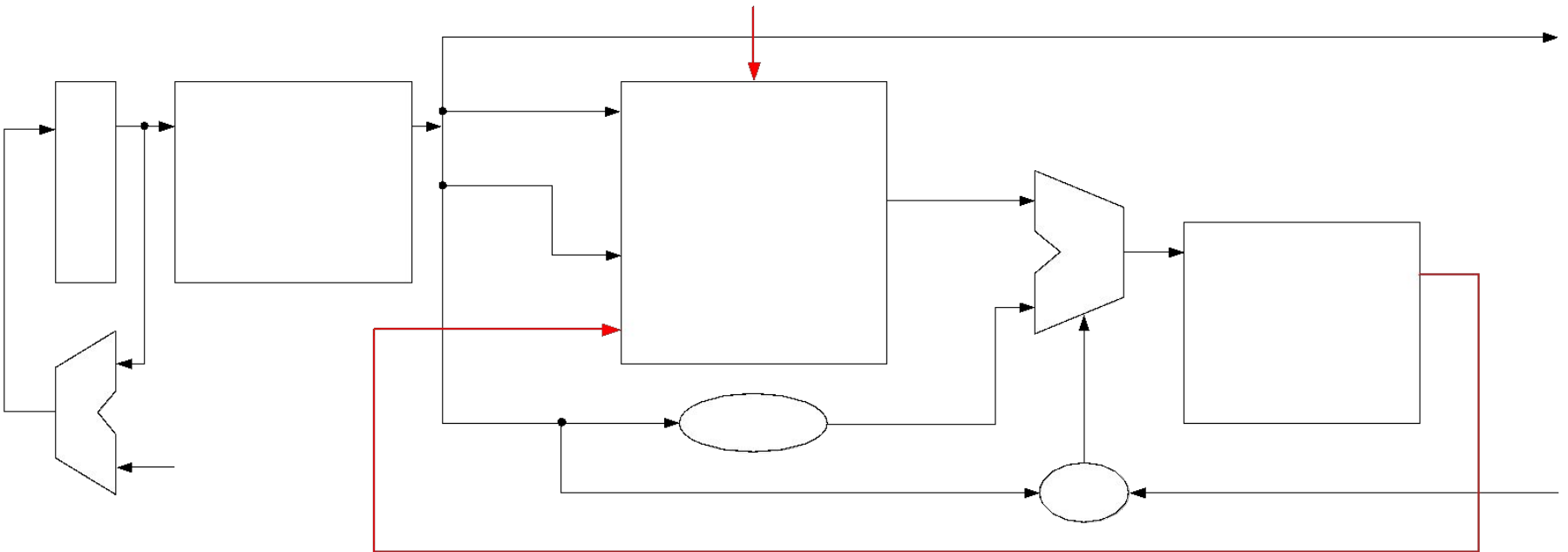
$B = \text{SigExt}(\text{imm})$

$C = R[rt]$





# Thực thi nhóm lệnh truyền dữ liệu (2/5) - lw





# Thực thi nhóm lệnh truyền dữ liệu (3/5) - sw

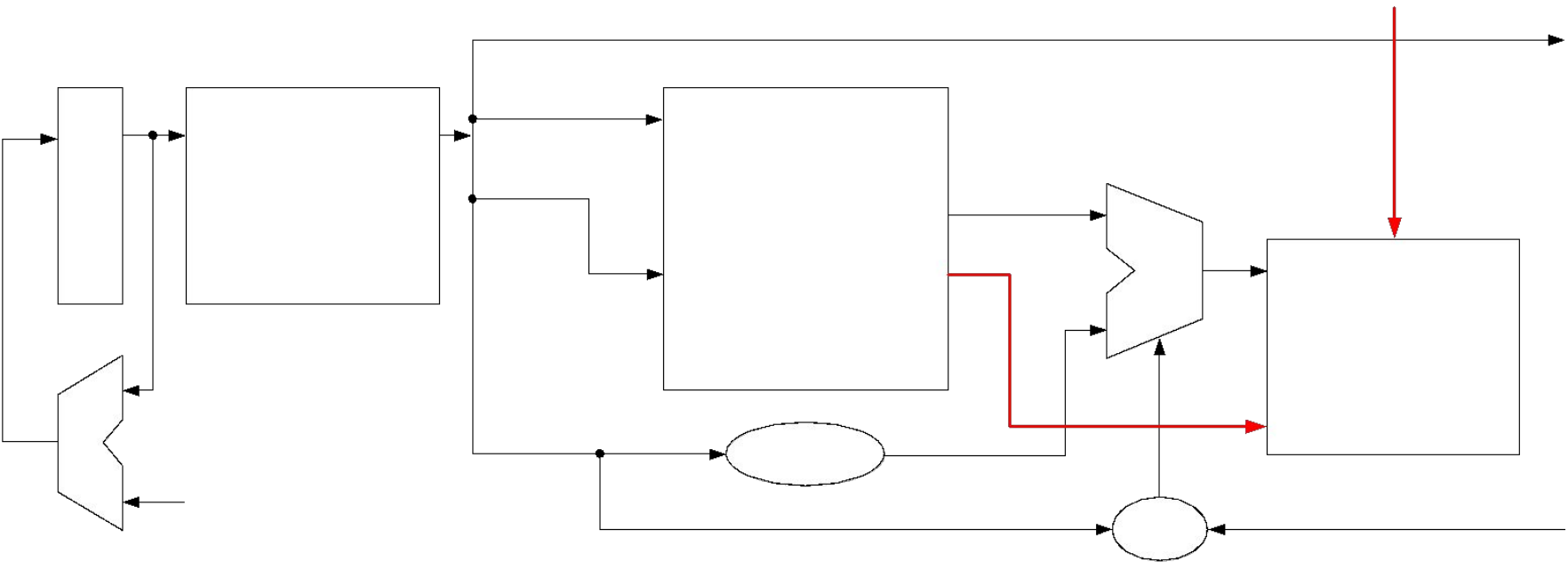
lw C, B(A)



$Inst = IM[PC]$        $A = R[rs]$        $ALU = A + B$        $DM[ALU] = C$   
 $PC = PC + 4$        $B = \text{SigExt}(\text{imm})$   
 $C = R[rt]$

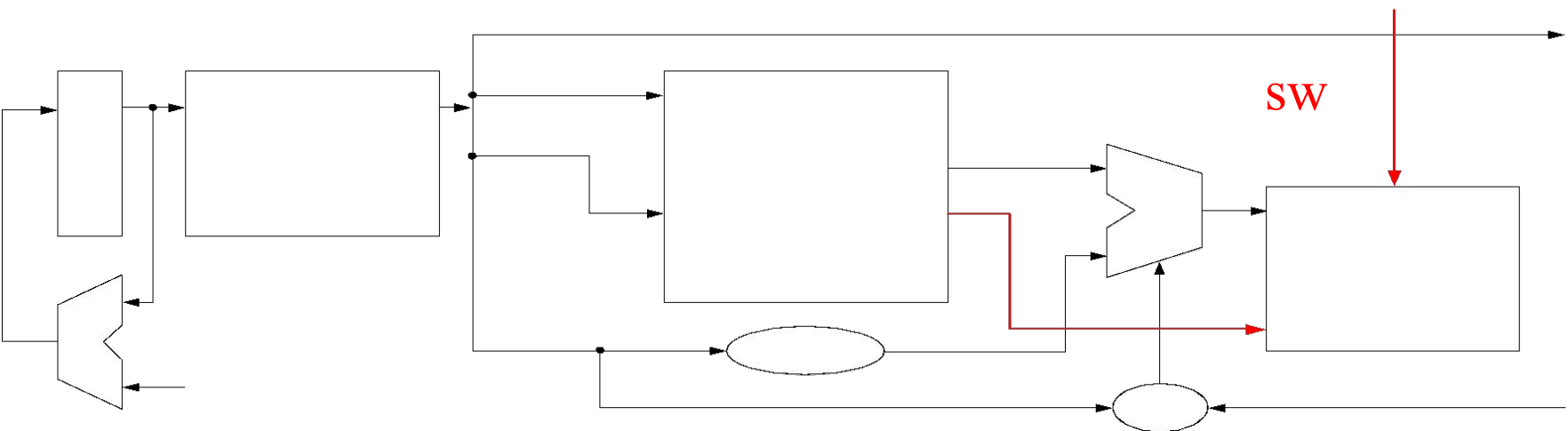
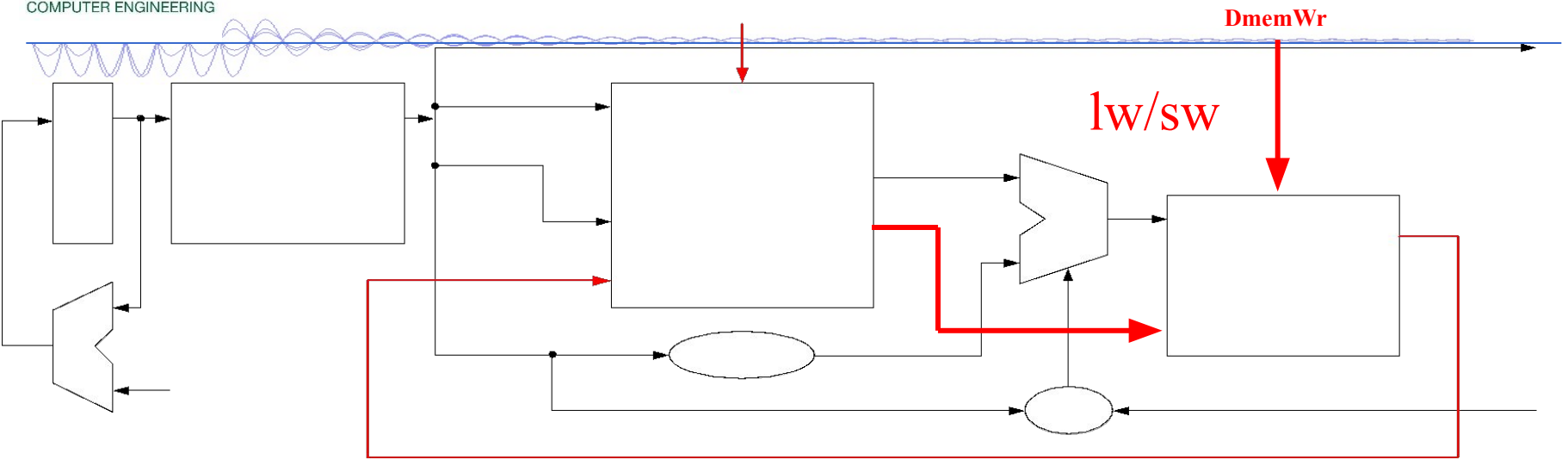


# Thực thi nhóm lệnh truyền dữ liệu (4/5) - sw





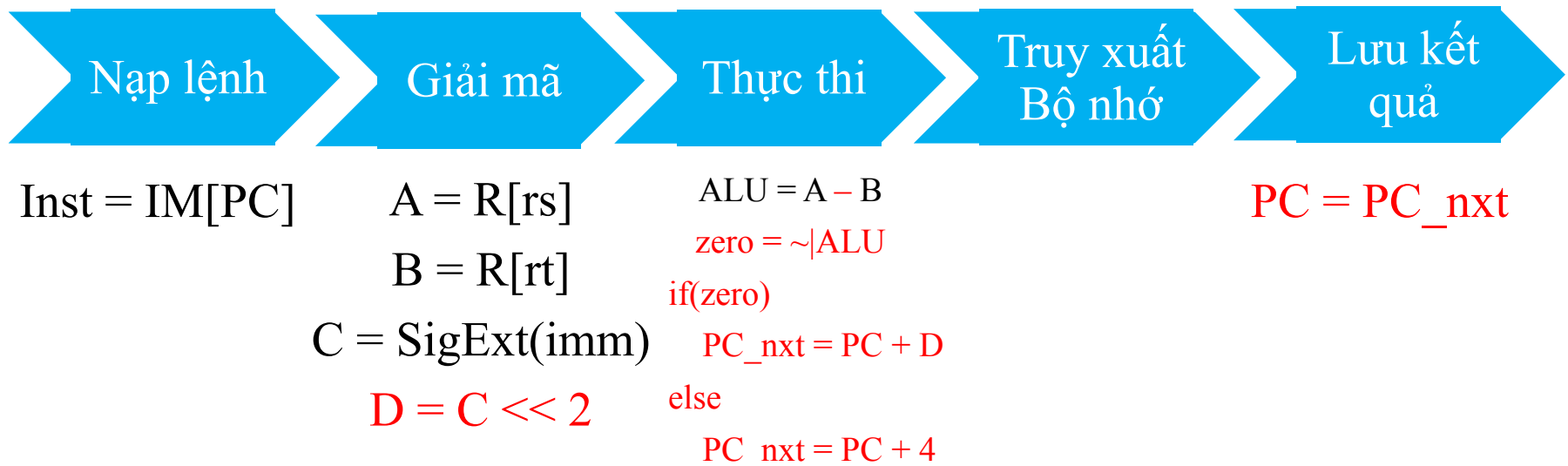
# Thực thi nhóm lệnh truyền dữ liệu (5/5)





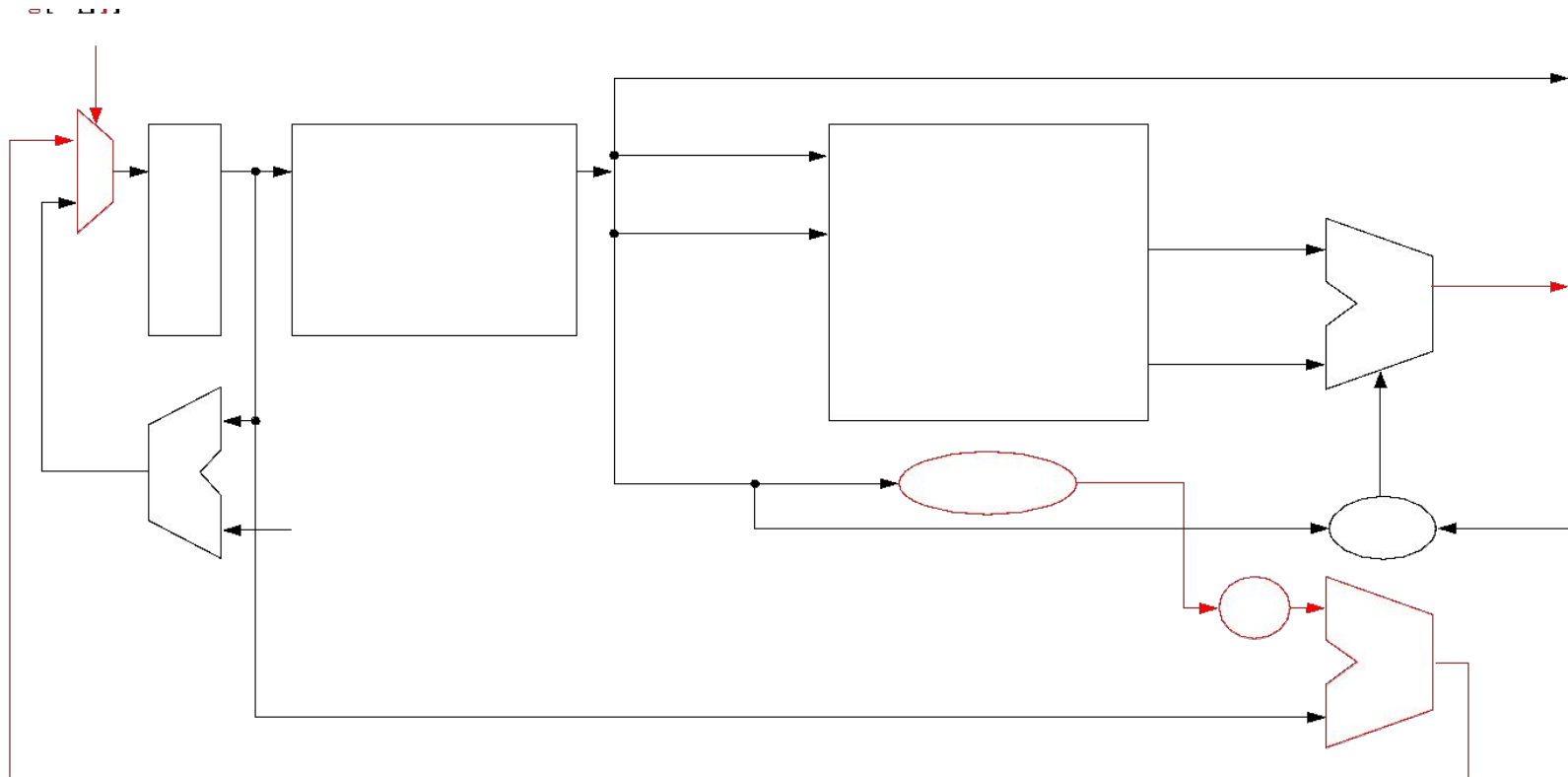
# Thực thi nhóm lệnh điều khiển (1/2) - beq

beq A, B, C



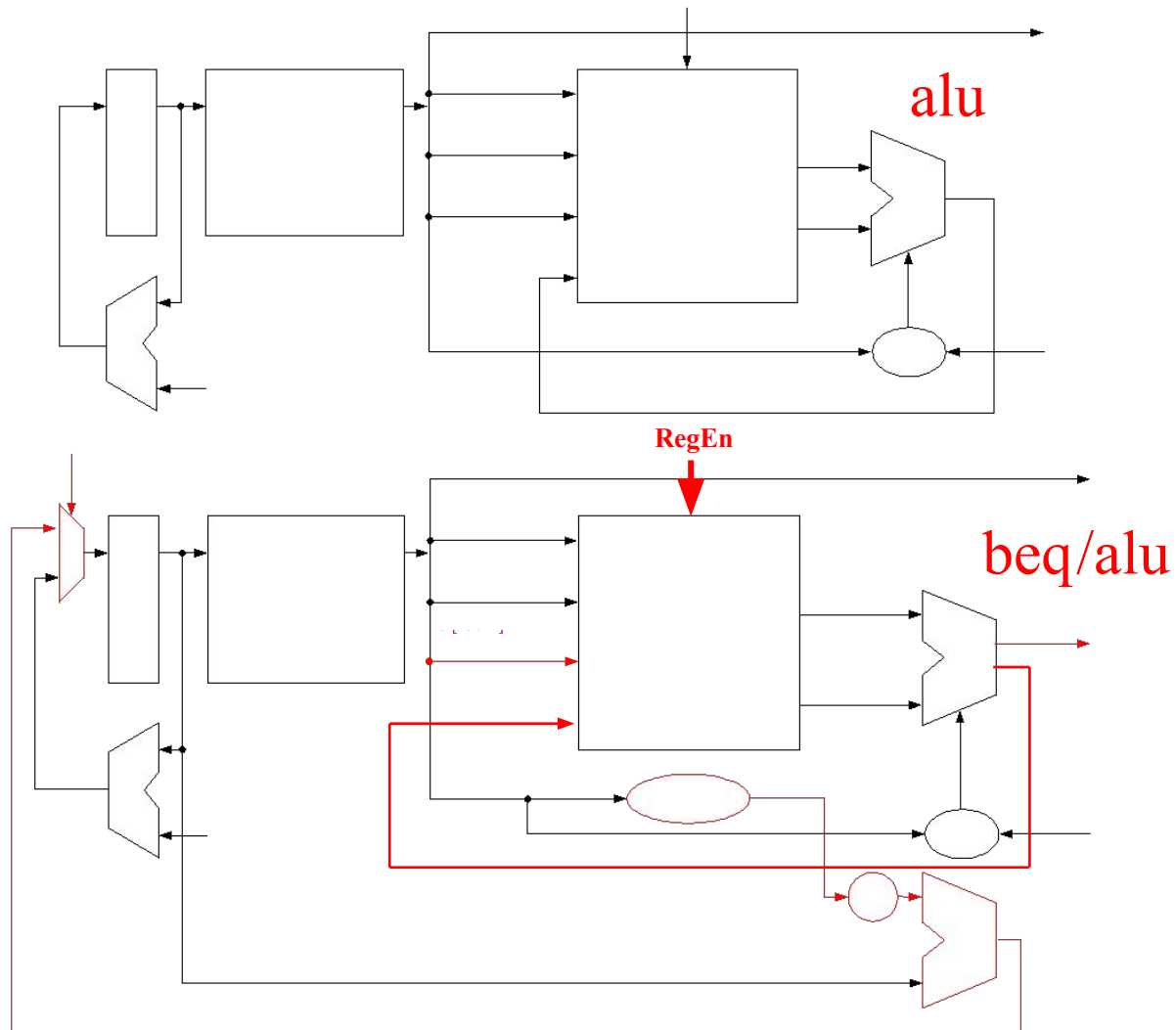


# Thực thi nhóm lệnh điều khiển (2/2) - beq





# Thực thi nhóm lệnh điều khiển + ALU

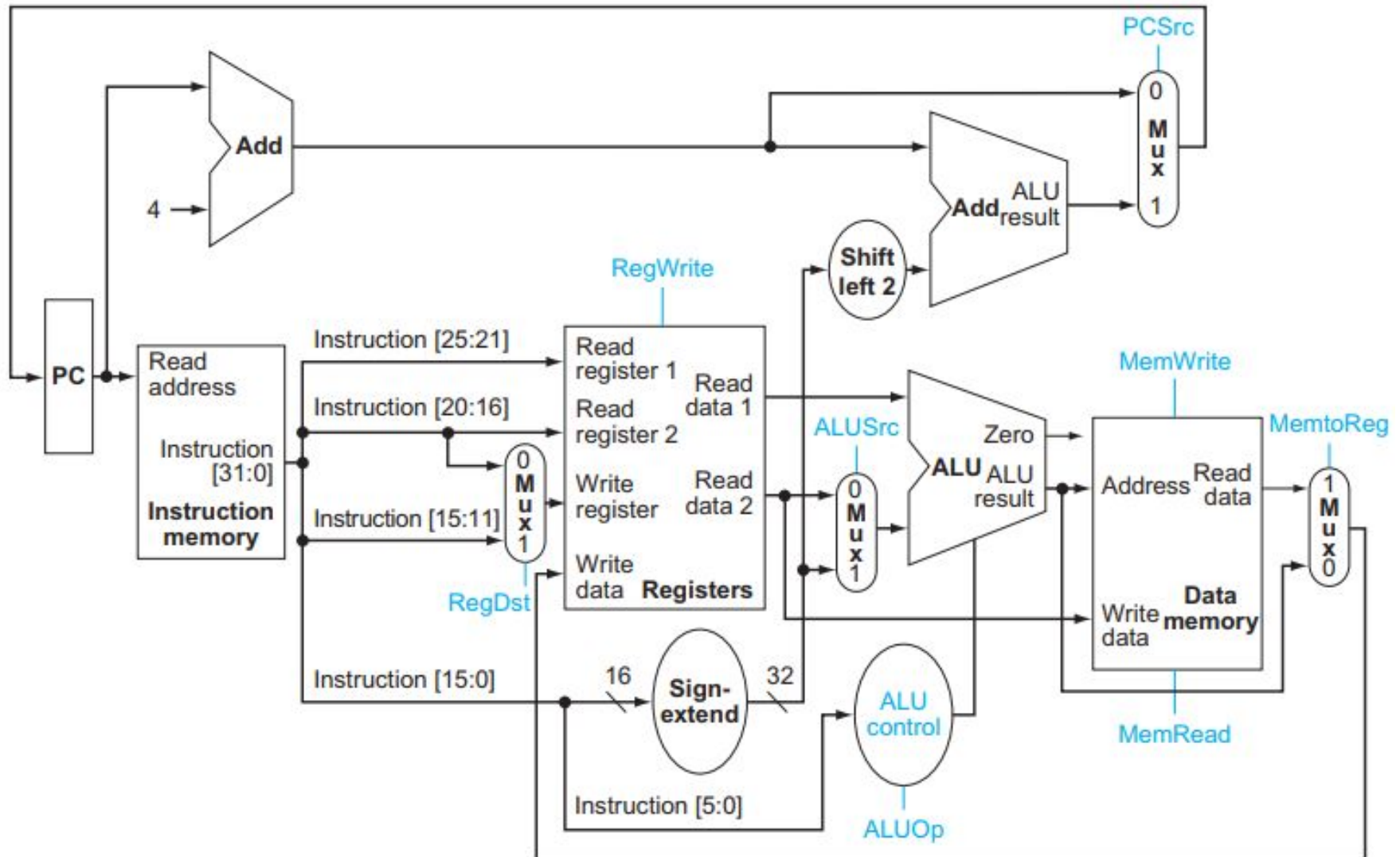








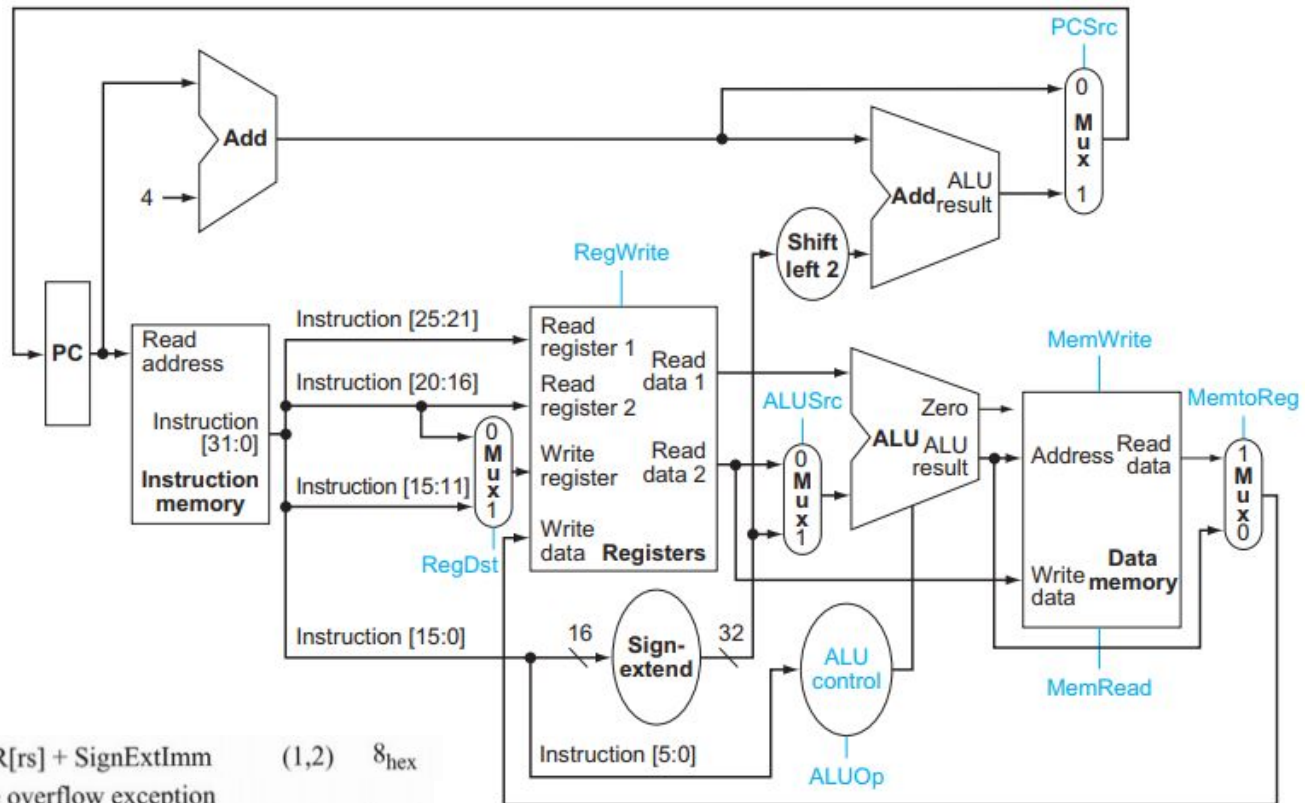
# Tổng kết: Datapath có thể thực thi beq/alu/lw/sw





# Bài tập (1/5)

Trình bày các khối chức năng được sử dụng khi thực thi lệnh addi?



Add Immediate    `addi`    `I`     $R[rt] = R[rs] + \text{SignExtImm}$     (1,2)     $8_{\text{hex}}$

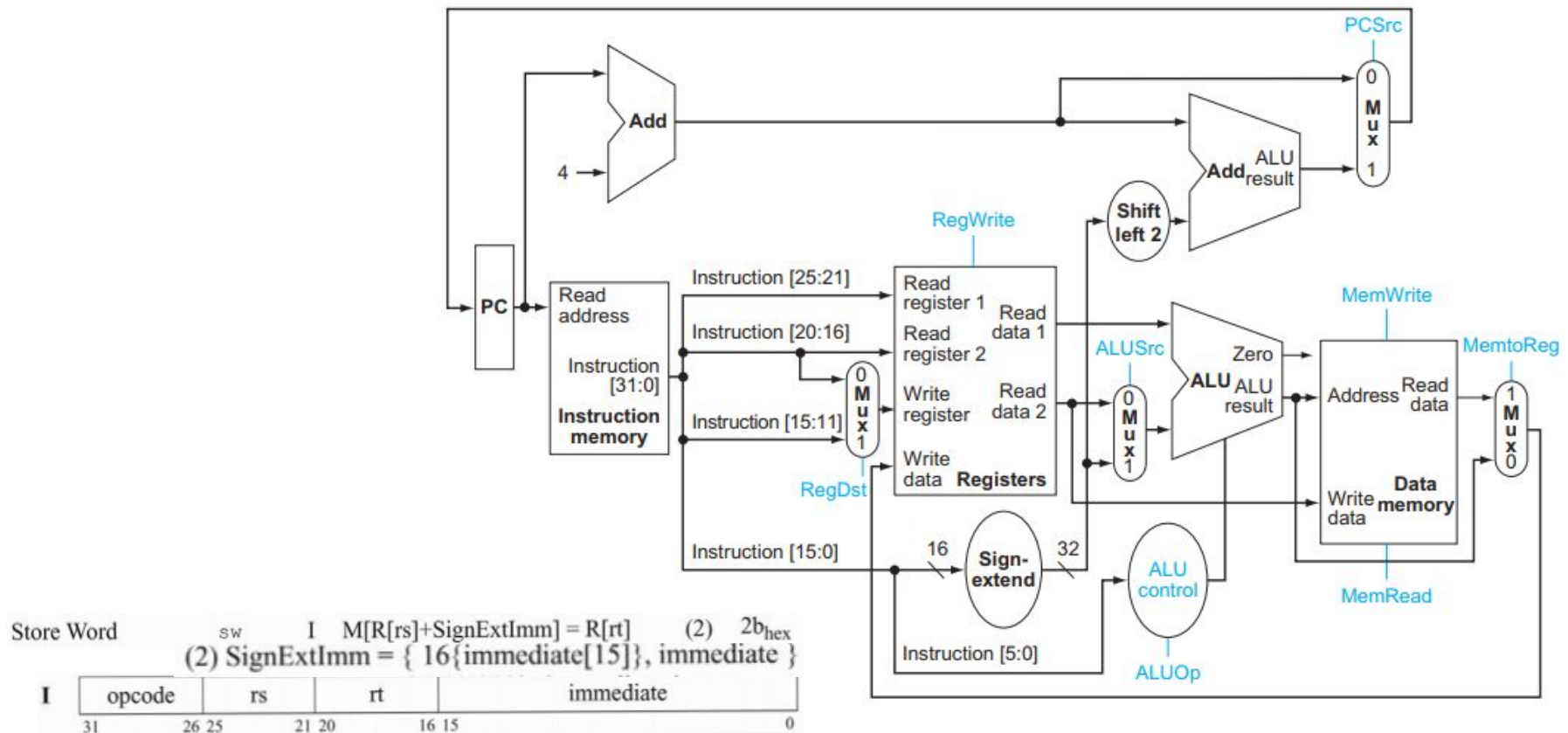
(1) May cause overflow exception  
(2)  $\text{SignExtImm} = \{ 16\{\text{immediate}[15]\}, \text{immediate} \}$





# Bài tập (2/5)

Trình bày các khối chức năng được sử dụng khi thực thi lệnh sw?



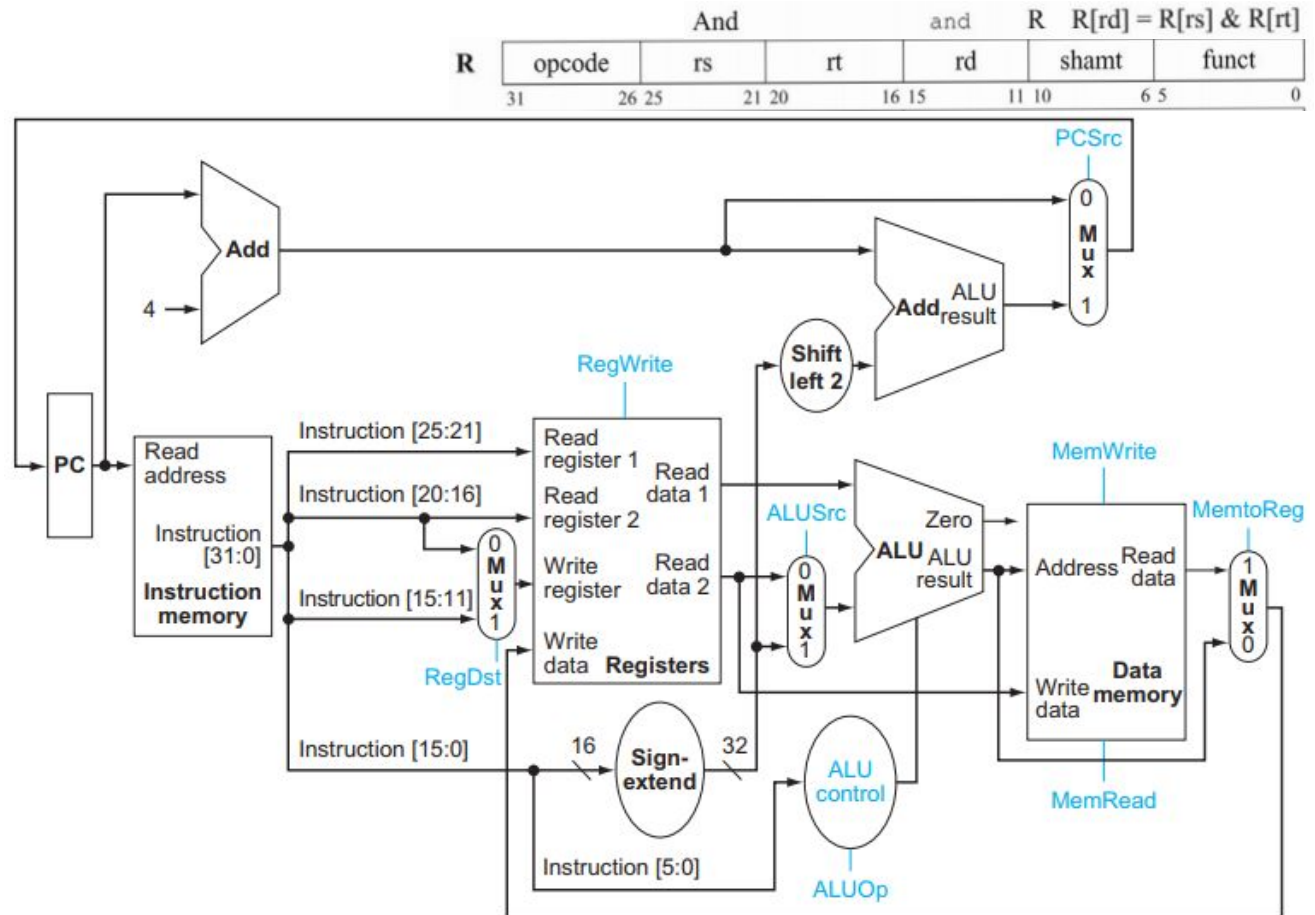


# Bài tập (3/5)

COMPUTER ENGINEERING

Tìm chu kỳ nhỏ nhất của CPU nếu chỉ thực thi lệnh and?

Quy ước: Việc đọc dữ liệu từ Register File phải thực hiện sau khi xác định đủ các thanh ghi cần đọc và ghi.



I-Mem	Add	Mux	ALU	Regs	D-Mem	Sign-Extend	Shift-Left-2
200ps	70ps	20ps	90ps	90ps	250ps	15ps	10ps

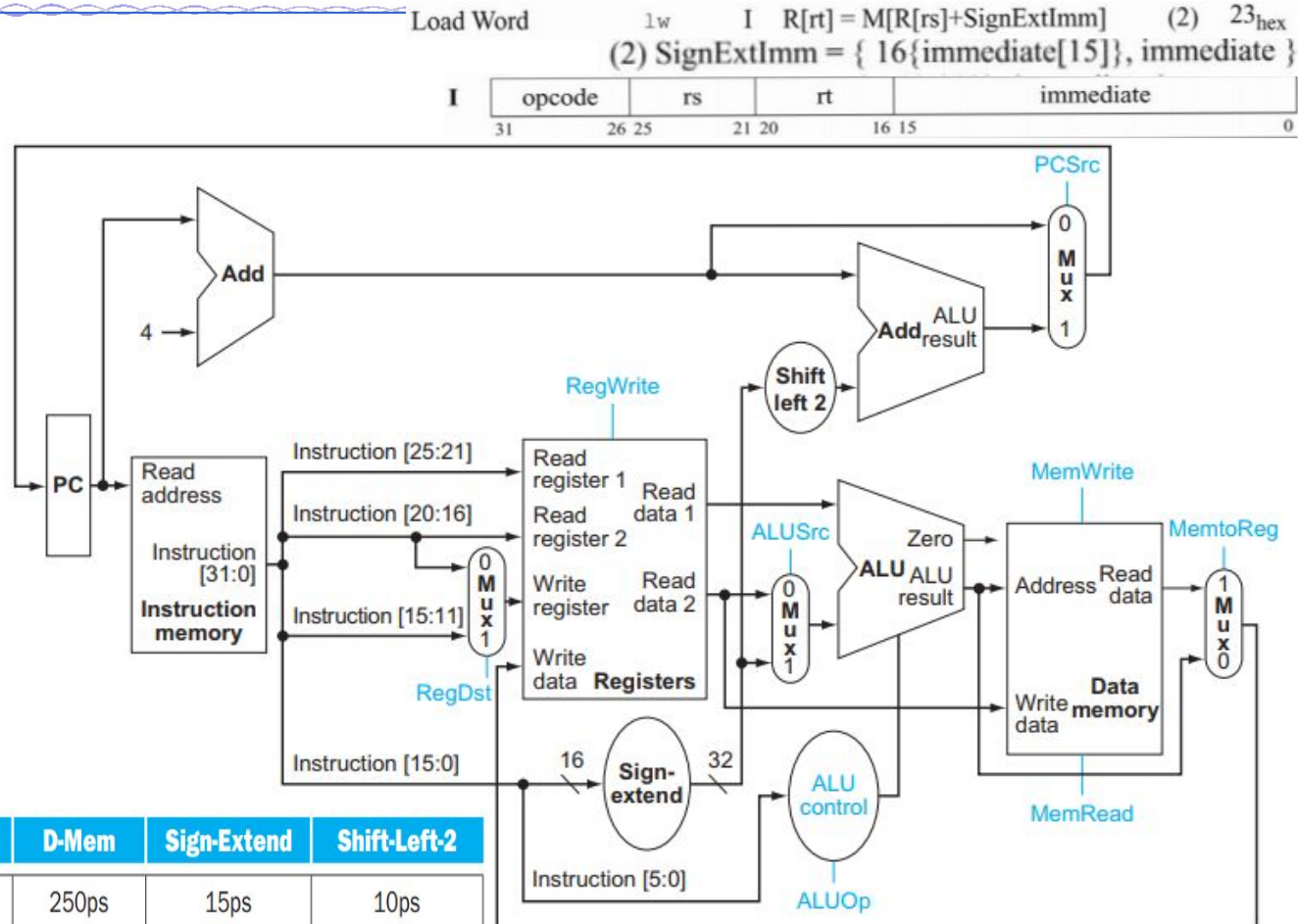


# Bài tập (4/5)

COMPUTER ENGINEERING

Tìm chu kỳ nhỏ nhất của CPU nếu chỉ thực thi lệnh lw?

I-Mem	Add	Mux	ALU	Regs	D-Mem	Sign-Extend	Shift-Left-2
200ps	70ps	20ps	90ps	90ps	250ps	15ps	10ps

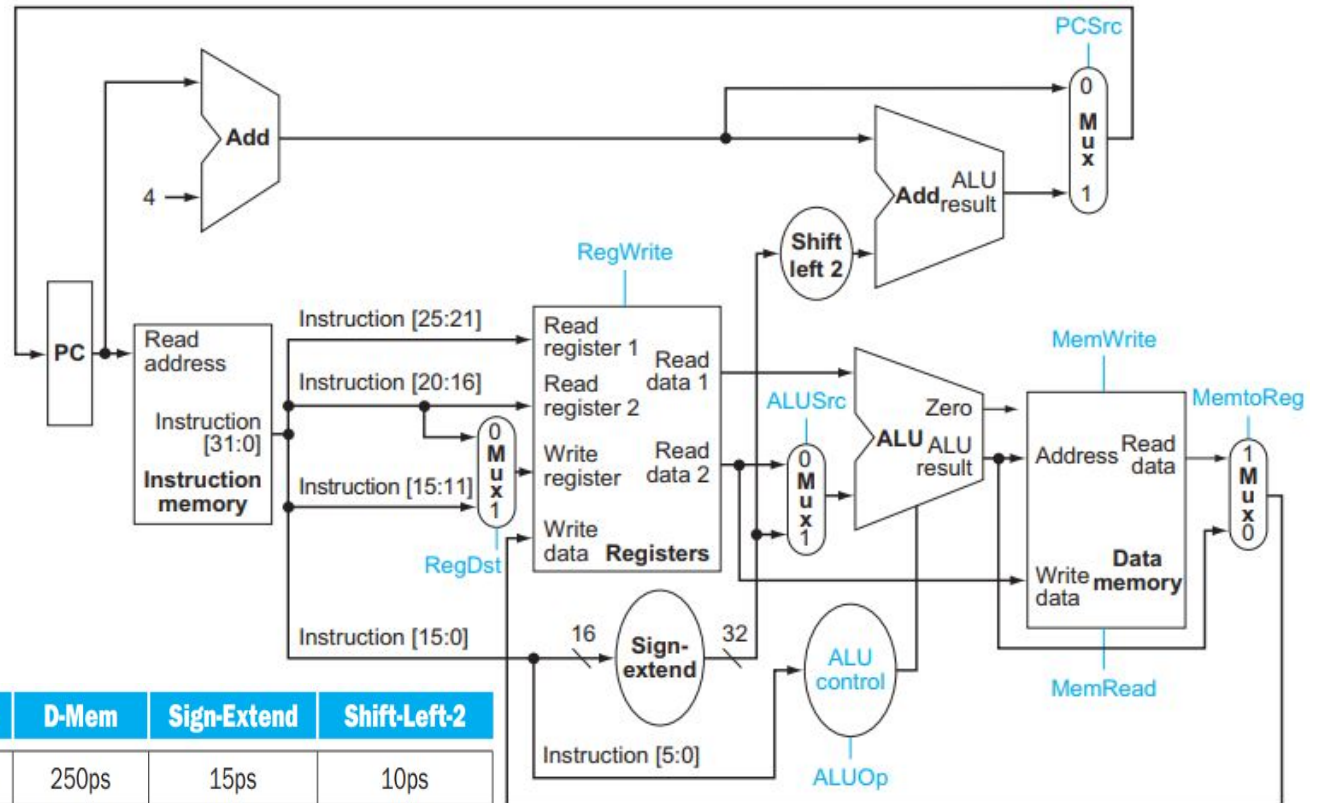
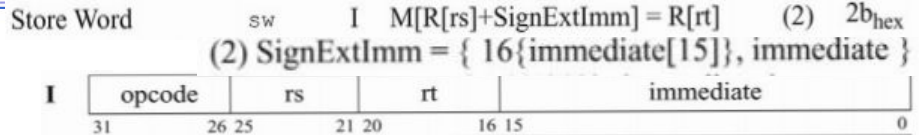






# Bài tập (5/5)

Tìm chu kỳ nhỏ nhất của CPU nếu chỉ thực thi lệnh sw?



I-Mem	Add	Mux	ALU	Regs	D-Mem	Sign-Extend	Shift-Left-2
200ps	70ps	20ps	90ps	90ps	250ps	15ps	10ps



COMPUTER ENGINEERING



**UIT**  
TRƯỜNG ĐẠI HỌC  
CÔNG NGHỆ THÔNG TIN

# THẢO LUẬN

