

# EXTRA

1/ Lệnh jal, jr có thực thi đối với kiến trúc trong hình 1 được không? Nếu được chỉ rõ datapath và các tín hiệu điều khiển kèm theo. Nếu không thì cần thêm những phần tử gì?

Lệnh `jal` (jump and link) và `jr` (jump register) có thể được thực thi trong kiến trúc MIPS. Tuy nhiên, để thực hiện chúng, chúng ta cần thêm một số thành phần vào datapath.

\*Lệnh `jal` (jump and link):

- Lệnh này lưu địa chỉ của lệnh kế tiếp vào thanh ghi `\$ra` (link) và nhảy tới nhãn (jump).
- Để thực hiện lệnh này, chúng ta cần thêm một đường dẫn từ PC+4 tới thanh ghi `\$ra`.
- Các tín hiệu điều khiển(Bảng 2).

\*Lệnh `jr` (jump register):

- Lệnh này nhảy tới địa chỉ nằm trong một thanh ghi.
- Để thực hiện lệnh này, chúng ta cần thêm một đường dẫn từ thanh ghi tới PC.
- Các tín hiệu điều khiển(Bảng 2).

	jal	jr
RegDst	X	X
ALUSrc	X	X
MemtoReg	X	X
RegWrite	1	0
MemRead	0	0
MemWrite	0	0
Branch	0	0
ALUOp	XX	XX
Jump	1	1

Bảng 2

(Lưu ý rằng "X" đại diện cho giá trị không quan trọng, nghĩa là nó có thể là bất kỳ giá trị nào và không ảnh hưởng đến thực thi lệnh.)

## 4.1: Instruction: AND Rd, Rs, Rt

#### 4.1.1

	AND
RegDst	1
ALUSrc	0
MemtoReg	0
RegWrite	1
MemRead	0
MemWrite	0
Branch	0
ALUOp	X
Jump	0

4.1.2 Which resources (blocks) perform a useful function for this instruction? → All (except Data memory and branch Add Unit)

4.1.3 Which resources (blocks) produce outputs, but their outputs are not used for this instruction? Which resources produce no outputs for this instruction?

- Outputs that are not used: Branch Add

- No outputs: Data memory

#### 4.2 Instruction: LWI Rt,Rd(Rs)

4.2.1 Which existing blocks (if any) can be used for this instruction? → This instruction uses instruction memory, both existing read ports of Registers, the ALU, and the write port of Register

4.2.2 Which new functional blocks (if any) do we need for this instruction?

→ Another read port in Register (to read Rx) and either a second ALU (to add Rx to Rs + Rt) or a third input to the existing ALU.

4.2.3 What new signals do we need (if any) from the control unit to support this instruction?

→ We need a control signal that tells the new ALU what to do, or if we extended the existing ALU, we need to add a new ADD operation.

#### 4.6

**4.6.1.** Để test lỗi kẹt bit 0 trên một dây, ta cần sử dụng một lệnh mà tín hiệu trên dây sẽ là 1 và sẽ cho ra kết quả khác nếu tín hiệu trên dây là 0

VD: Nếu tín hiệu trên dây là 0 thì một lệnh thực hiện việc ghi vào thanh ghi số lẻ sẽ thực hiện ghi vào thanh ghi số chẵn. VD: Nếu thanh ghi 30 có giá trị là 0, thanh ghi 31 có giá trị là 1, khi thực hiện lệnh Add \$31, \$30, \$30 thì giá trị thanh ghi 31 sẽ bằng 0. Nhưng khi input Write Register của khối Register bị kẹt tại 0 thì giá trị tính toán được sẽ ghi vào thanh ghi 30 và thanh ghi 31 vẫn có giá trị là 1.

**4.6.2.** Bài test cho lỗi kẹt bit 0 cần một lệnh mà tín hiệu là 1, bài test cho lỗi kẹt bit 1 cần 1 lệnh mà tín hiệu là 0. Vì tín hiệu không thể bằng 0 và 1 cùng lúc trong 1 chu kỳ, nên chúng ta không thể test 2 lỗi kẹt bit trên cùng một tín hiệu bằng 1 lệnh được.

Bài test cho lỗi kẹt bit 1 tương tự như cho kẹt bit 0:

Nếu thanh ghi 31 có giá trị là 0, thanh ghi 30 có giá trị là 1, khi thực hiện lệnh Add \$30, \$31, \$31 thì giá trị thanh ghi 30 sẽ bằng 0. Nhưng khi input Write Register của khối Register bị kẹt tại 1 thì giá trị tính toán được sẽ ghi vào thanh ghi 31 và thanh ghi 30 vẫn có giá trị là 1.

**4.6.3.** Ta cần viết lại một chương trình mà chỉ sử dụng những thanh ghi số lẻ

**4.6.4.** Để test lỗi này, ta cần một lệnh mà tín hiệu MemRead sẽ là 1, tín hiệu RegDst sẽ là 0, ở đây lệnh load là phù hợp nhất. Khi thực hiện lệnh load, nếu tín hiệu MemRead = 0 nghĩa là không đọc dữ liệu trong bộ nhớ, điều này dẫn đến việc lệnh sẽ ghi một giá trị bất kỳ vào thanh ghi Rt. Tuy nhiên, giá trị bất kỳ này có thể bằng chính giá trị của thanh ghi Rt trước đó, vì vậy bài test này không có kết luận.

**4.6.5.** Để test lỗi này, ta cần một lệnh mà tín hiệu Jump sẽ là 1. Ở đây lệnh jump phù hợp. Tuy nhiên khi lệnh jump thực hiện thì tín hiệu RegDst là “don’t care” vì không thực hiện ghi vào thanh ghi. Vì tín hiệu RegDst là “don’t care” nên có thể bằng 0 hoặc bằng 1, do đó bài test này sẽ không có một kết luận chắc chắn.