

## 1. Xử lí tập lệnh và phát triển tập lệnh:

- RRR:

Ins	Opcode 15 13	Rs 12 10	Rt 9 7	Rd 6 4	Funct 3 0	Assembly-Code Format	Meaning
add	000	Rs	Rt	Rd	0000	add Rd, Rs, Rt	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} + \text{Reg[Rt]}$
inc	000	Rs	000	Rd	0001	inc Rd, Rs	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} + 1$
sub	000	Rs	Rt	Rd	0010	sub Rd, Rs, Rt	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} - \text{Reg[Rt]}$
dec	000	Rs	000	Rd	0011	dec Rd, Rs	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} - 1$
and	000	Rs	Rt	Rd	0100	and Rd, Rs, Rt	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} \& \text{Reg[Rt]}$
or	000	Rs	Rt	Rd	0101	or Rd, Rs, Rt	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]}   \text{Reg[Rt]}$
nand	000	Rs	Rt	Rd	0110	nand Rd, Rs, Rt	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} \sim\& \text{Reg[Rt]}$
xor	000	Rs	Rt	Rd	0111	xor Rd, Rs, Rt	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} \wedge \text{Reg[Rt]}$
shfl	000	Rs	000	Rd	1000	shfl Rd, Rs	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} \ll 1$
shfr	000	Rs	000	Rd	1001	shfr Rd, Rs	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} \gg 1$
sll	000	Rs	000	Rd	1010	sll Rd, Rs	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} \ll 2$
srr	000	Rs	000	Rd	1011	srr Rd, Rs	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} \gg 2$
sra	000	Rs	000	Rd	1100	sra Rd, Rs	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} \gg 1$ (giữ bit đầu)
mul	000	Rs	Rt	Rd	1101	mul Rd, Rs, Rt	$\text{Reg[Rd]} \leftarrow \text{Reg[Rs]} \times \text{Reg[Rt]}$
slt	000	Rs	Rt	Rd	1110	slt Rd, Rs, Rt	$\text{Reg[Rd]} \leftarrow (\text{Reg[Rs}] < \text{Reg[Rt]}) ? 1 : 0$
sgt	000	Rs	Rt	Rd	1111	sgt Rd, Rs, Rt	$\text{Reg[Rd]} \leftarrow (\text{Reg[Rs}] > \text{Reg[Rt]}) ? 1 : 0$

- RRI:

Ins	Opcode 15 13	Rs 12 10	Rt 9 7	Imm 6 0	Assembly-Code Format	Meaning
addi	001	Rs	Rt	Imm	addi Rt, Rs, Imm	$\text{Reg[Rt]} \leftarrow \text{Reg[Rs]} + \text{SignExtImm}$
subi	010	Rs	Rt	Imm	subi Rt, Rs, Imm	$\text{Reg[Rt]} \leftarrow \text{Reg[Rs]} - \text{SignExtImm}$
beq	011	Rs	Rt	Imm	beq Rs, Rt, Imm	$\text{PC} \leftarrow (\text{Reg[Rs]} == \text{Reg[Rt]}) ? \text{Imm} : \text{PC} + 1$

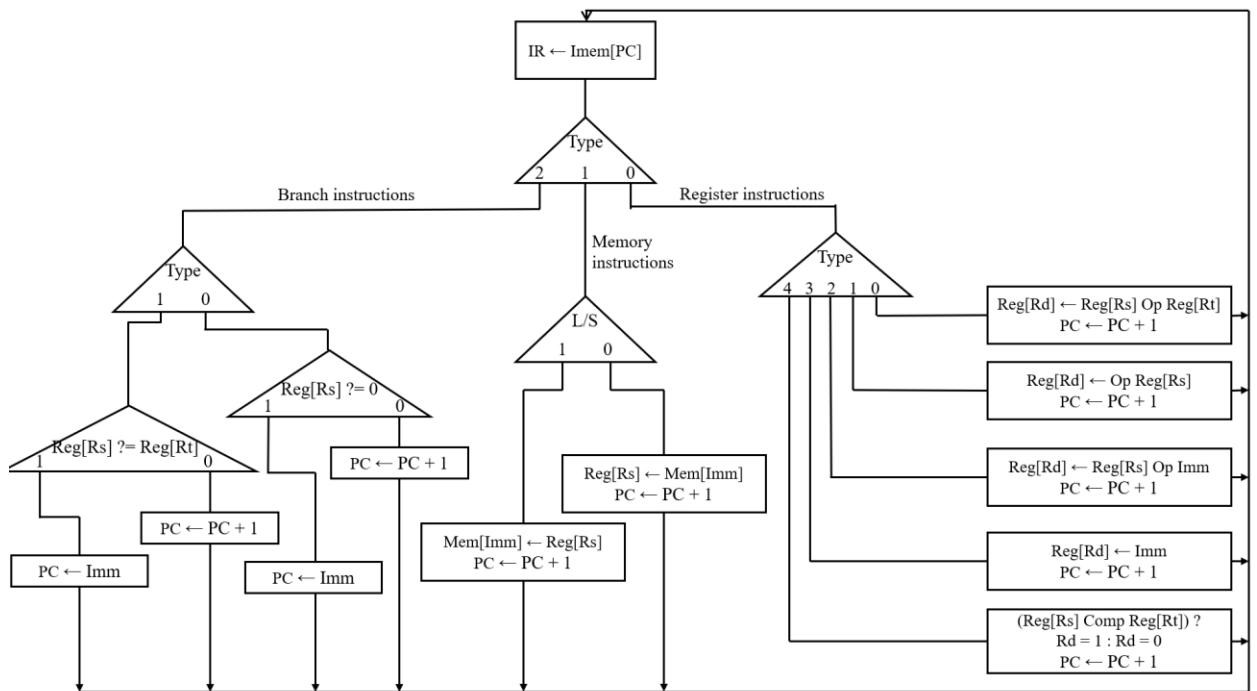
- RI:

Ins	Opcode 15 13	Rs 12 10	Imm 9 0	Assembly-Code Format	Meaning
lw	100	Rs	Imm	lw Rs, Imm	Reg[Rs] $\leftarrow$ Mem[Imm]
sw	101	Rs	Imm	sw Rs, Imm	Mem[Imm] $\leftarrow$ Reg[Rs]
jnz	110	Rs	Imm	jnz Rs, Imm	PC $\leftarrow$ (Reg[Rs] != 0) ? Imm : PC + 1
li	111	Rs	Imm	li Rs, Imm	Reg[Rs] $\leftarrow$ Imm

- Extend (Mã giả):

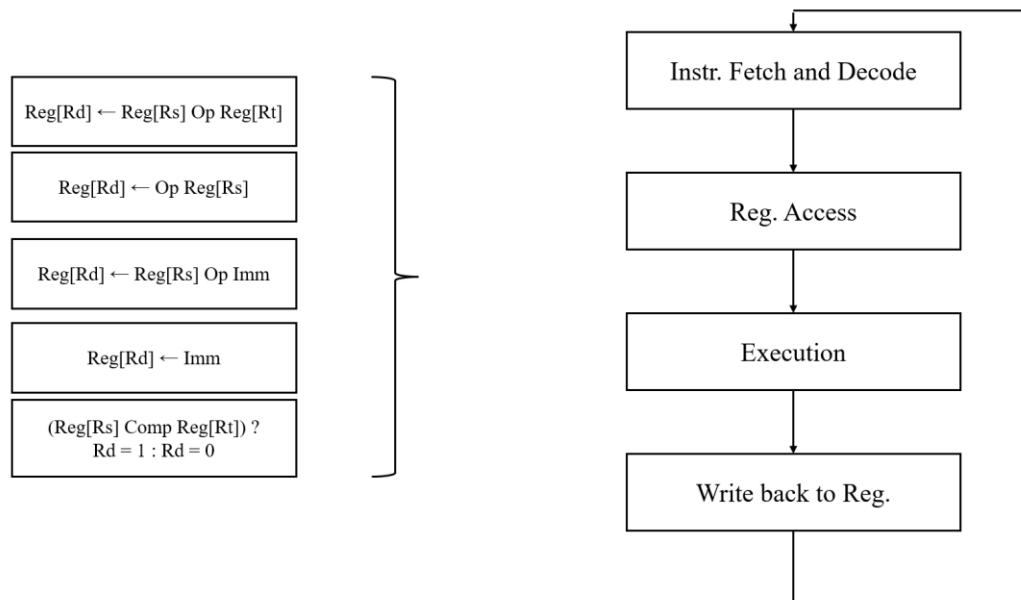
Ins	Opcode 15 13	Rs 12 10	Rt 9 7	Rd 6 4	Funct 3 0	Assembly-Code Format	Meaning
not	000	Rs	Rs	Rd	0110	not Rd, Rs	Reg[Rd] $\leftarrow$ Reg[Rs] ~& Reg[Rs]
mov	000	Rs	Rs	Rd	0100	mov Rd, Rs	Reg[Rd] $\leftarrow$ Reg[Rs] & Reg[Rs]
nop	001	Rs	Rs	0000000		nop	Reg[Rs] $\leftarrow$ Reg[Rs] + 0
jmp	011	Rs	Rs	Imm		jmp Imm	PC $\leftarrow$ (Reg[Rs] == Reg[Rs]) ? Imm : PC + 1

## 2. IS Flowchart:

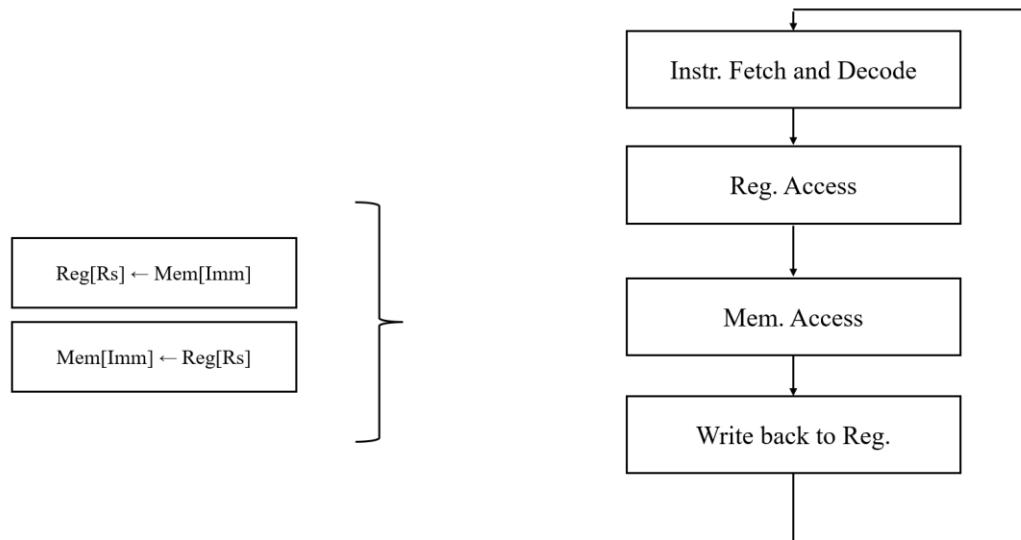


### 3. ASM Flowchart:

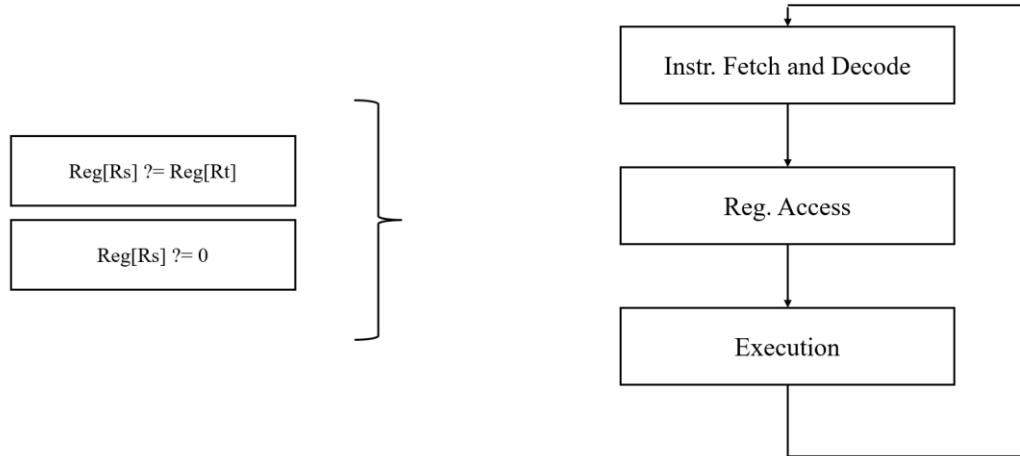
- Các lệnh không thao tác với DMem:



- Các lệnh thao tác với DMem:



- Các lệnh rẽ nhánh:



#### 4. Xác định các thành phần Datapath:

- **Program Counter (PC):** Thanh ghi để lưu địa chỉ của lệnh được thực thi kế tiếp.
- **Instruction Memory (IMem) 64x16:** Bộ nhớ lưu trữ các lệnh sẽ được thực thi (tối đa 64 lệnh).
- **Register File 8x16:** Tập 8 thanh ghi để thực thi các thao tác.
- **ALU:** Bộ tính toán số học (+, +1, -, -1, &, |, ~&, ^ và các tổ hợp luận lý cho ra tín hiệu so sánh).
- **Shifter:** Bộ dịch (No-shift, << 1, >> 1, >> 1 (số học)).
- **Data Address Decoder (DAD):** Bộ giải mã SignExtImm[15:0] thành DMemAddr[5:0]. DMemAddr[5] sử dụng cho I/O.
- **I/O Controller:** Bộ điều khiển chức năng DMem dựa theo tín hiệu điều khiển RW\_DMem từ Control Unit.
- **Data Memory (DMem) 32x16:** Vùng lưu trữ dữ liệu chương trình (RAM, tối đa 32 word).
- **Multiplexers:** Bộ chọn (4x1, 2x1).