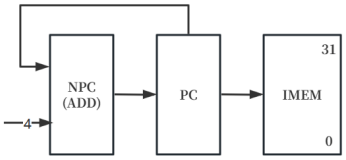
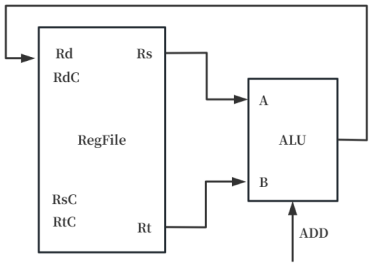
R型指令

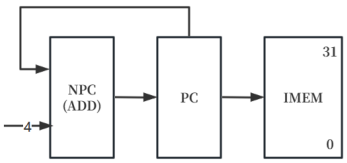
1.ADD

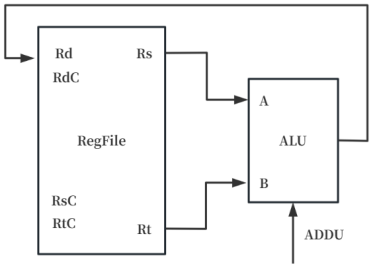





```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs → A , Rt → B
(A + B → RES)
RES → Rd
```

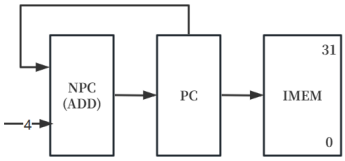
2.ADDU

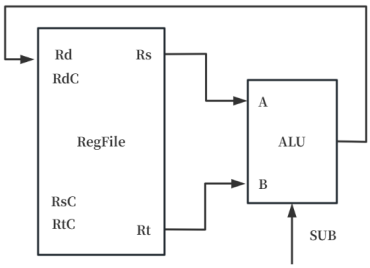





```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs → A , Rt → B
(A + B → RES)
RES → Rd
```

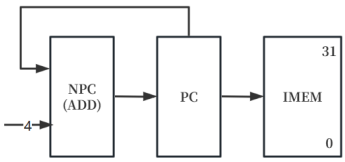
3.SUB

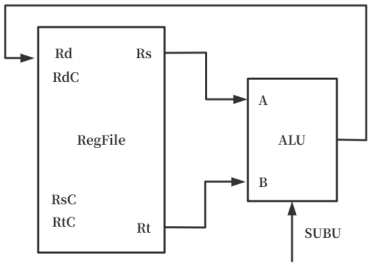





```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs → A , Rt → B
(A - B → RES)
RES → Rd
```

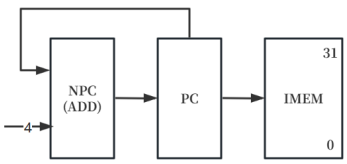
4.SUBU

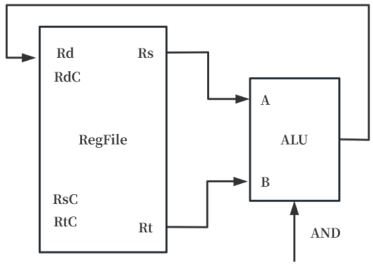





```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs → A , Rt → B
(A - B → RES)
RES → Rd
```

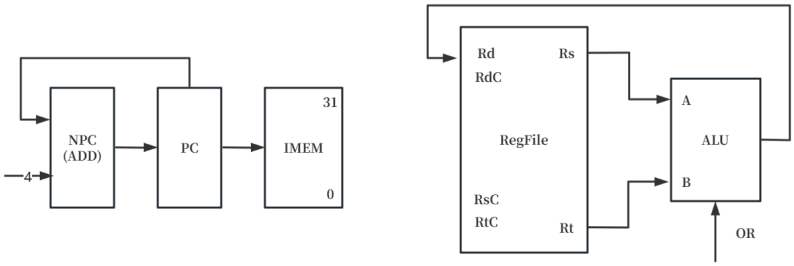
5.AND





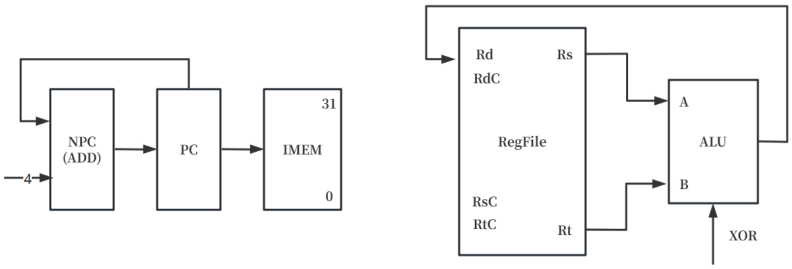
```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs → A , Rt → B
(A & B → RES)
RES → Rd
```

6.OR



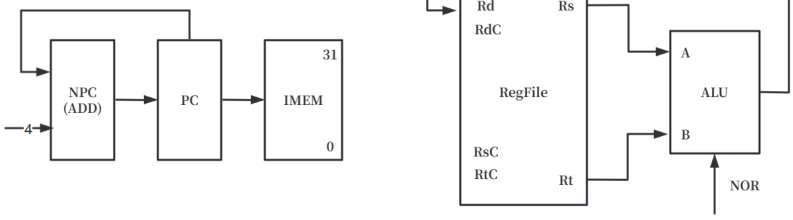

```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs → A , Rt → B
(A | B → RES)
RES → Rd
```

7.XOR



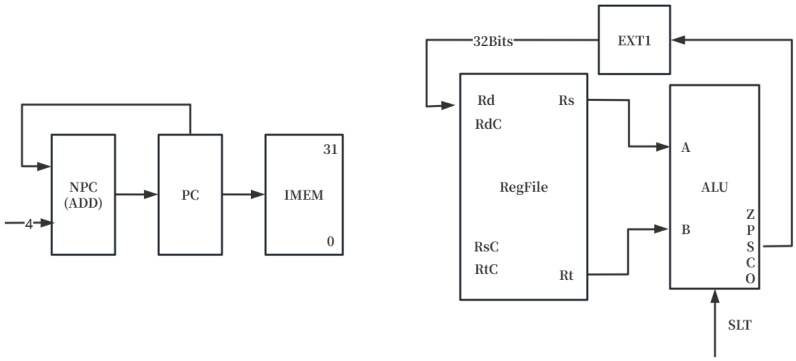

```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs → A , Rt → B
(A ⊕ B → RES)
RES → Rd
```

8.NOR



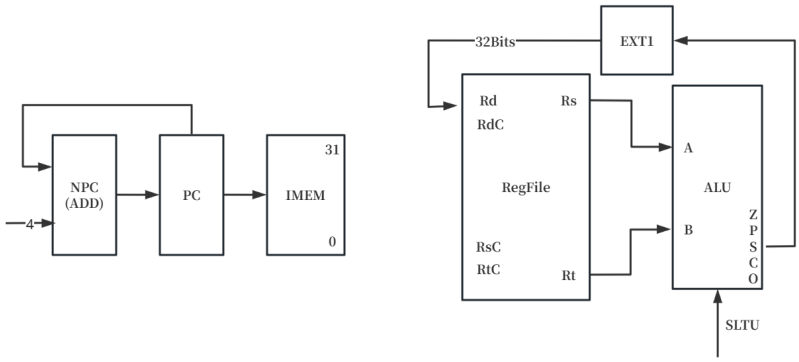

```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs → A , Rt → B
(A ⊙ B → RES)
RES → Rd
```

9.SLT



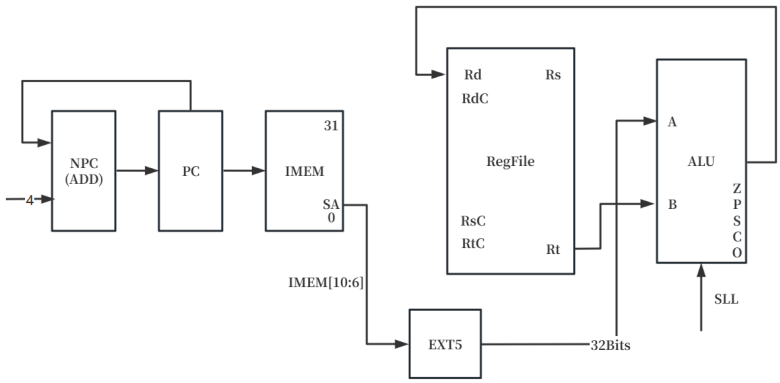

```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs → A , Rt → B
(A - B → RES)  //相减判断，负数则为Rs中数小
SF → EXT1      //注意要做扩展
EXT1_OUT → Rd
```

10.SLTU



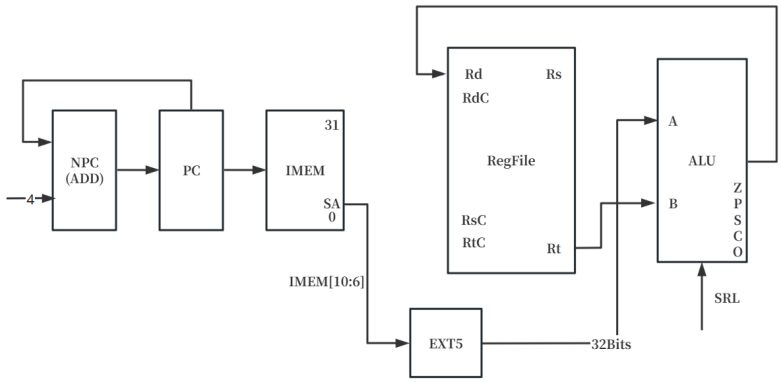

```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs → A , Rt → B
(A - B → RES)  //相减判断，负数则为Rs中数小
SF → EXT1      //注意要做扩展
EXT1_OUT → Rd
```

11.SLL



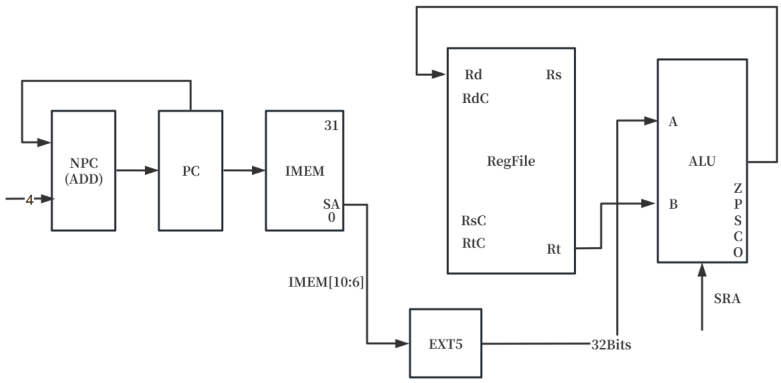

```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[10:6] → EXT5
EXT5_OUT → A

Rt → B
(B<<A → RES)
RES → Rd
```

12.SRL



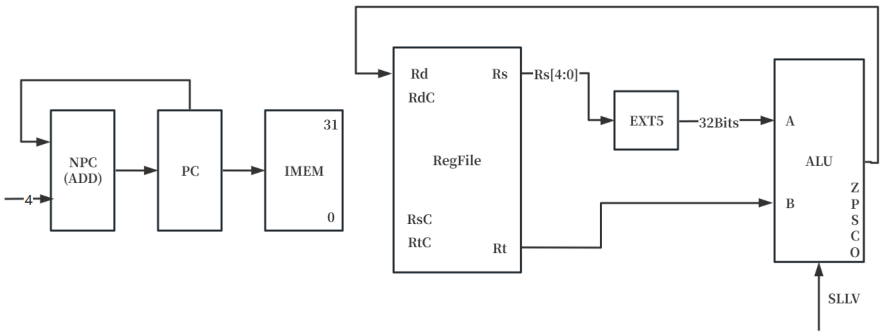

```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[10:6] → EXT5
EXT5_OUT → A

Rt → B
(B>>A → RES)
RES → Rd
```

13.SRA



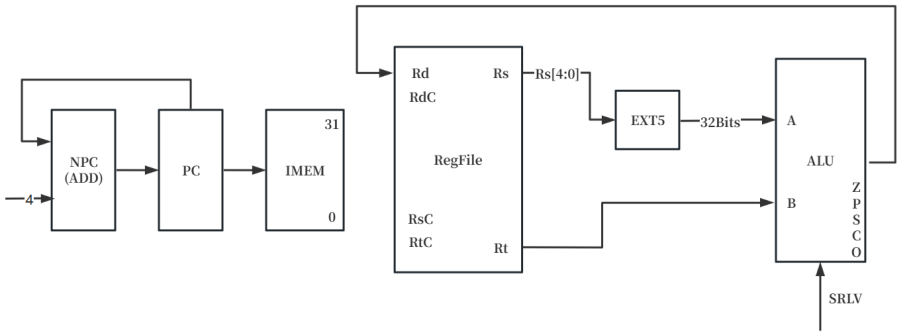

```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[10:6] → EXT5
EXT5_OUT → A

Rt → B
(B>>A → RES)
RES → Rd
```

14.SLLV



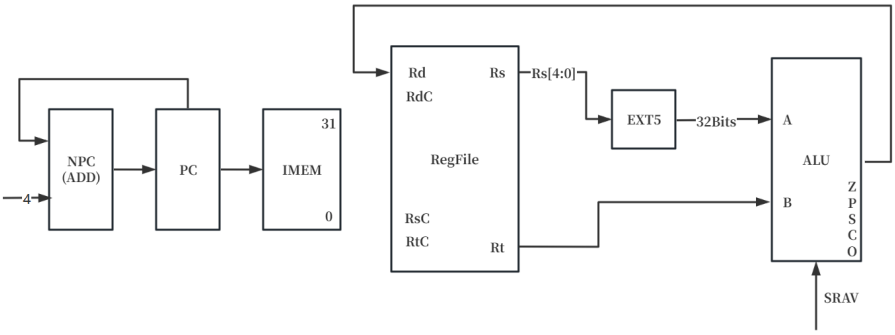

```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs[4:0] → EXT5
EXT5_OUT → A
Rt → B
(A<<B → RES)
RES → Rd
```

15.SRLV



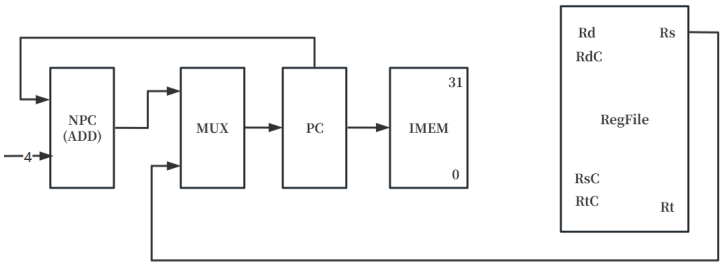

```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs[4:0] → EXT5
EXT5_OUT → A
Rt → B
(A>>B → RES)
RES → Rd
```

16.SRAV




```
PC → IMEM
PC + 4 → NPC
NPC → PC

Rs[4:0] → EXT5
EXT5_OUT → A
Rt → B
(A>>B → RES)
RES → Rd
```

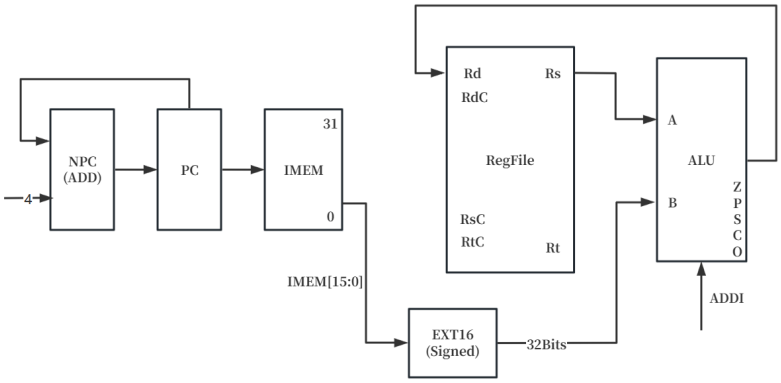
17.JR




```
PC → IMEM
PC + 4 → NPC    //无关指令
Rs → MUX
MUX_OUT → PC

NPC → MUX       //无关指令
```

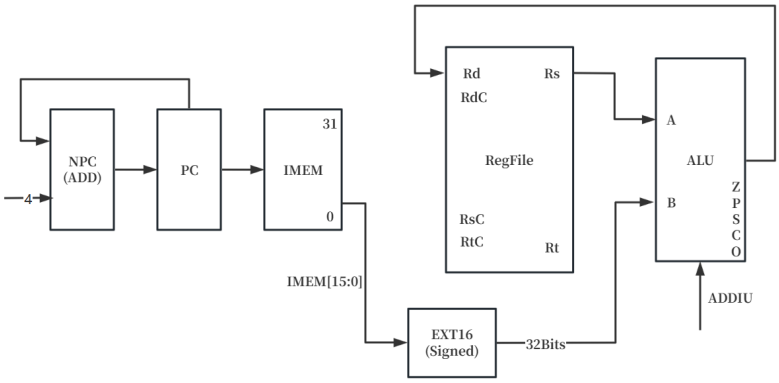
I型指令

18.ADDI



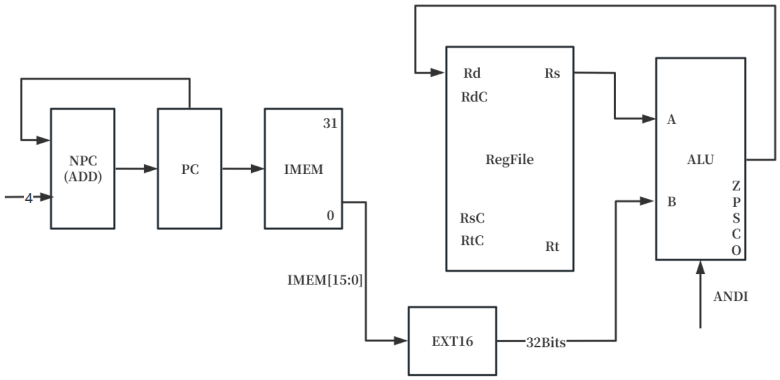

```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[15:0] → EXT16
EXT16_OUT → B
Rs → A
(A + B → RES)
RES → Rd
```

19.ADDIU



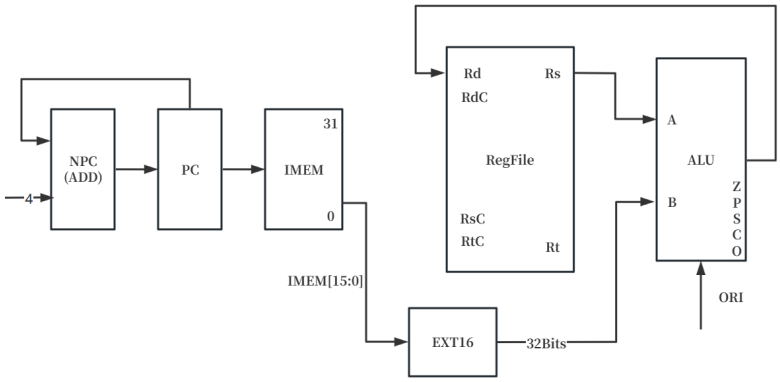

```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[15:0] → EXT16
EXT16_OUT → B
Rs → A
(A + B → RES)
RES → Rd
```

20.ANDI



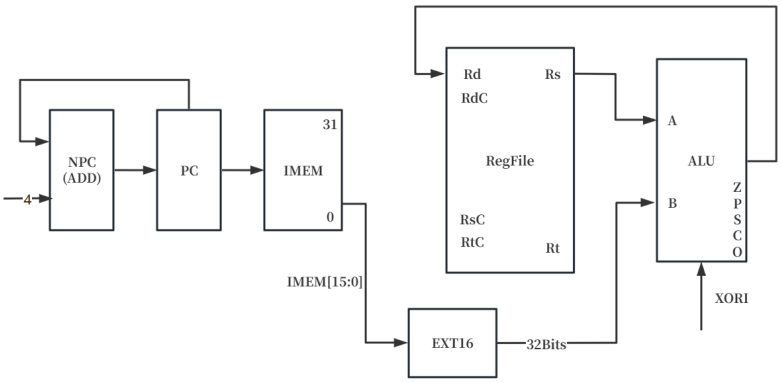

```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[15:0] → EXT16
EXT16_OUT → B
Rs → A
(A & B → RES)
RES → Rd
```

21.ORI



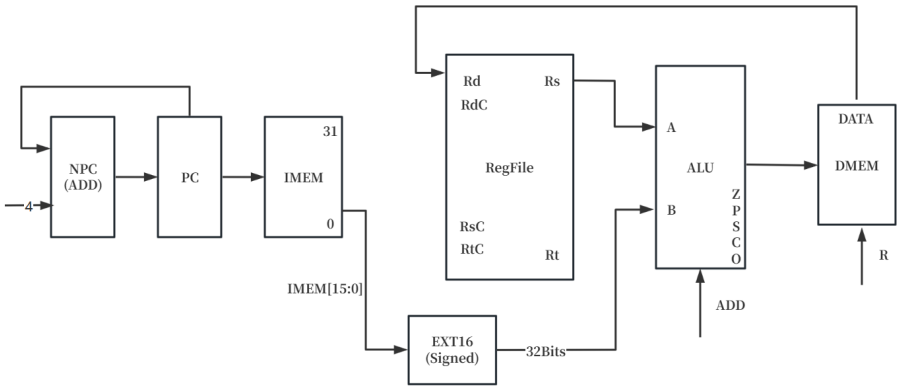

```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[15:0] → EXT16
EXT16_OUT → B
Rs → A
(A | B → RES)
RES → Rd
```

22.XORI



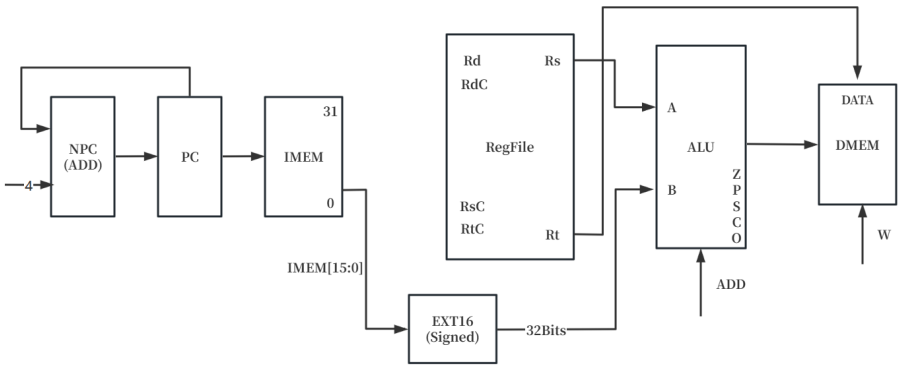

```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[15:0] → EXT16
EXT16_OUT → B
Rs → A
(A ⊕ B → RES)
RES → Rd
```

23.LW



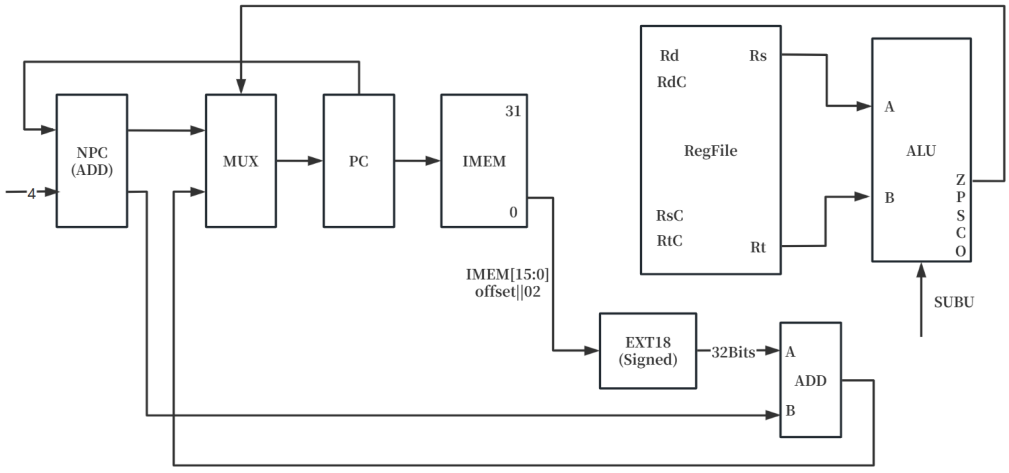

```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[15:0] → EXT16
EXT16_OUT → B
Rs → A
(A + B → RES)
RES → DMEM_ADDR
DMEM_OUT → Rd
```

24.SW




```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[15:0] → EXT16
EXT16_OUT → B
Rs → A
(A + B → RES)
Rt → DMEM
RES → DMEM_ADDR
```

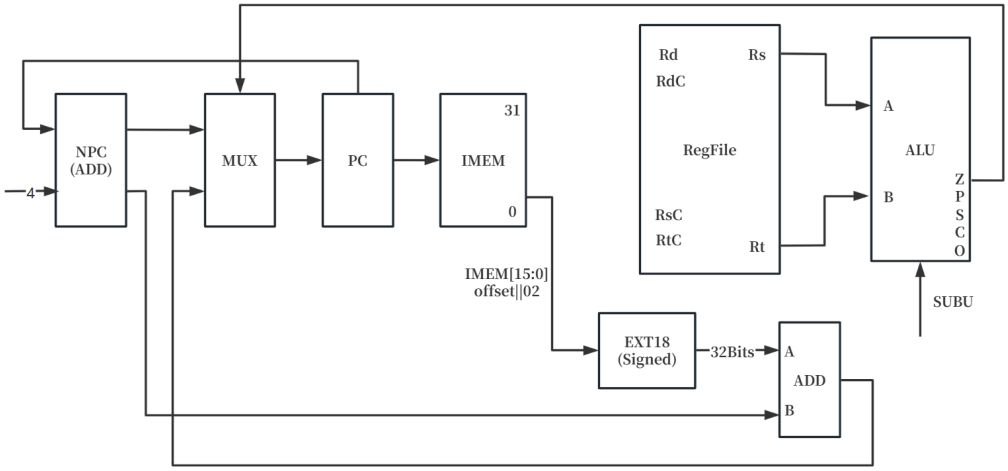
25.BEQ




```
PC → IMEM
PC + 4 → NPC
NPC → MUX

IMEM[15:0] || 02 → EXT18
EXT18_OUT → ADD_A
NPC → ADD_B
(ADD_A + ADD_B → ADD_OUT)
ADD_OUT → MUX

Rs → A
Rt → B
(A + B → RES)
```

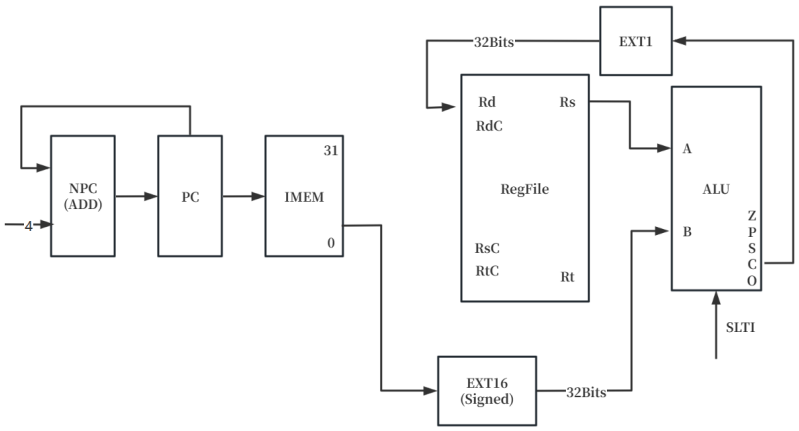
```
Z → MUX

MUX → PC
```

26.BNE



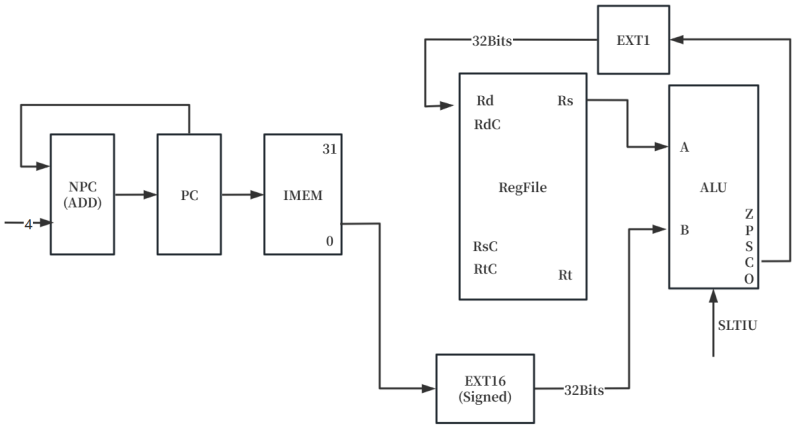

```
PC → IMEM
PC + 4 → NPC
NPC → MUX

IMEM[15:0] || 02 → EXT18
EXT18_OUT → ADD_A
NPC → ADD_B
(ADD_A + ADD_B → ADD_OUT)
ADD_OUT → MUX

Rs → A
Rt → B
(A + B → RES)
Z → MUX

MUX → PC
```

27.SLTI



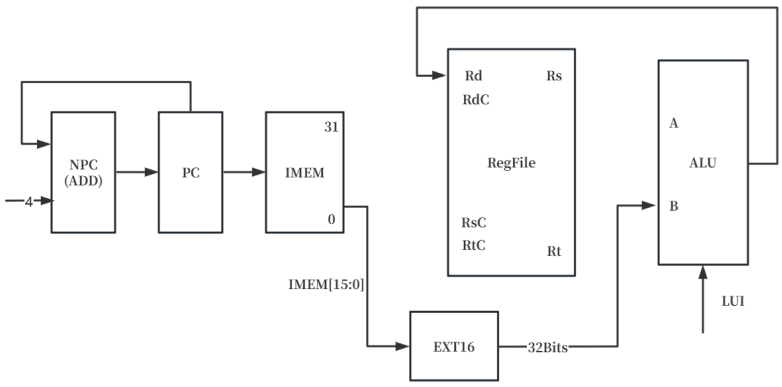
```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[15:0] → EXT16
EXT16_OUT → B
Rs → A
(A - B → RES)
CF → EXT1
EXT1_OUT → Rd
```

28.SLTIU




```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[15:0] → EXT16
EXT16_OUT → B
Rs → A
(A - B → RES)
CF → EXT1
EXT1_OUT → Rd
```

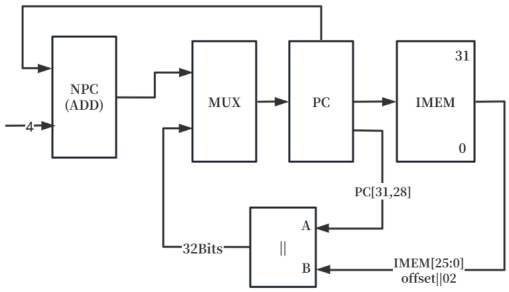
29.LUI



```
PC → IMEM
PC + 4 → NPC
NPC → PC

IMEM[15:0] → EXT16
EXT16_OUT → B
RES → Rd
```

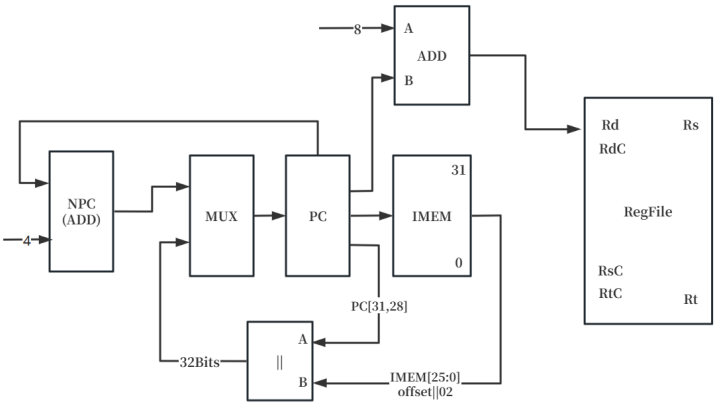
J型指令

30.J




```
PC → IMEM
PC + 4 → NPC
NPC → MUX

PC[31:28] → ||_A
IMEM[25,0] || 02 → ||_B
||_OUT → MUX

MUX_OUT → PC
```

31.JAL




```
PC → IMEM
PC + 4 → NPC
NPC → MUX
```

```
8 → ADD_A
PC → ADD_B
(ADD_A + ADD_B → ADD_OUT)
ADD_OUT → Rd

PC[31:28] → ||_A
IMEM[25,0] || 02 → ||_B
||_OUT → MUX

MUX_OUT → PC
```