国防科技大学计算机学院

# 计算机原理

COMPUTER PRINCIPLE

第四章 第二节 (2) 单周期处理器的控制器设计

# 1. 设计步骤

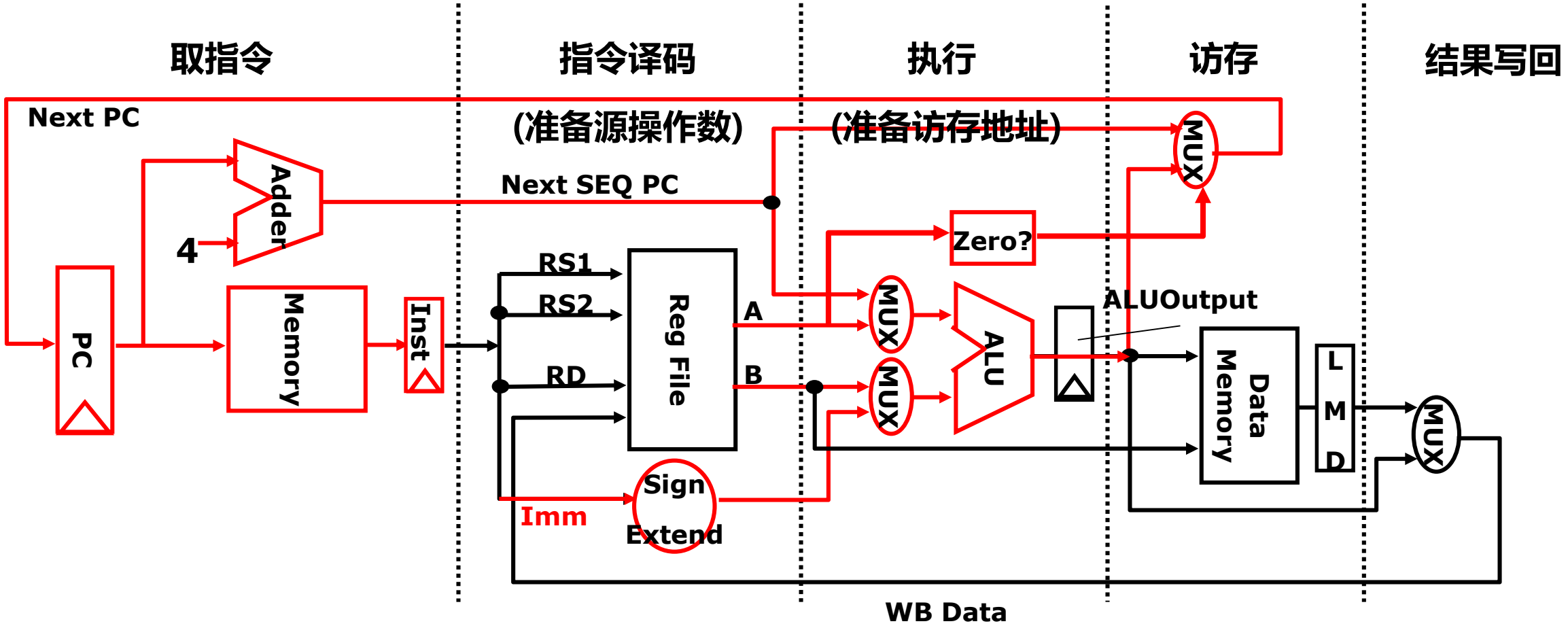
- 根据指令功能和数据通路，

① 确定数据通路上每个元件所需的控制信号以及控制信号的取值

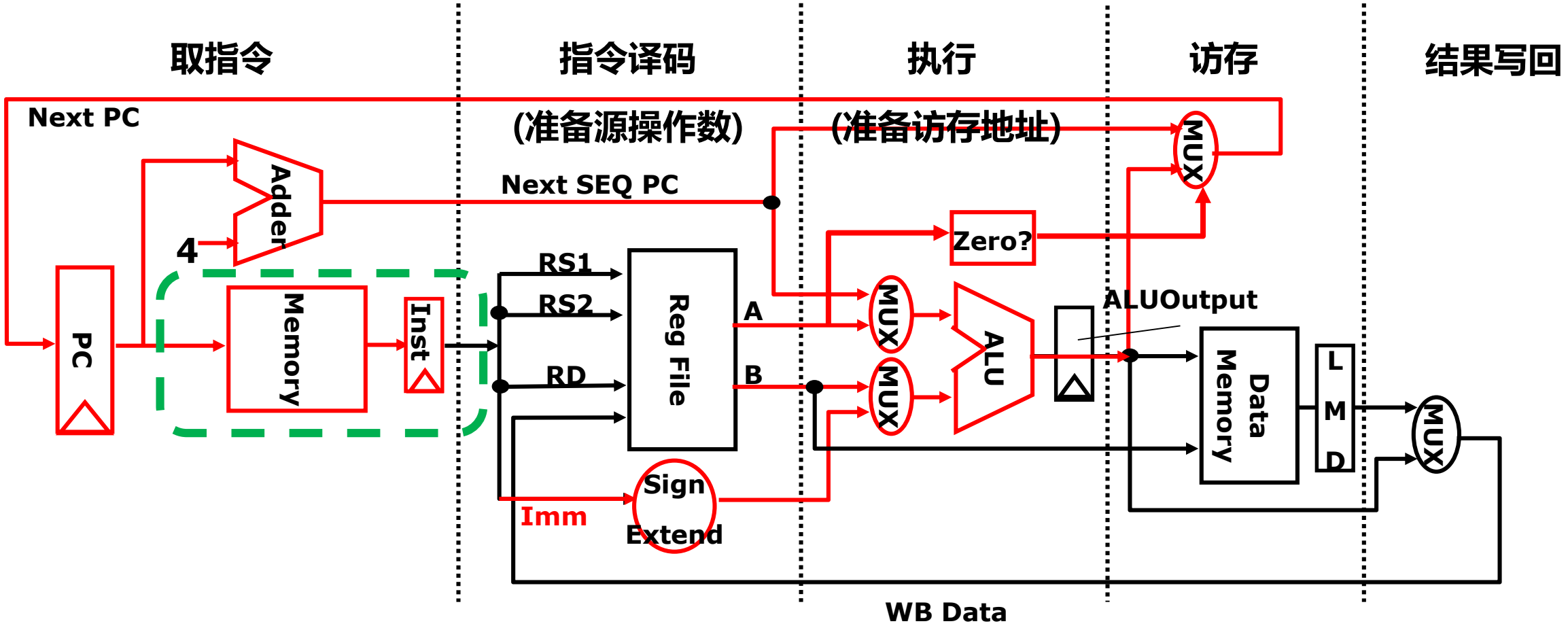
② 汇总所有指令涉及到的控制信号，生成一张反映指令与控制信号之间的关系表

③ 根据关系表得到每个控制信号的逻辑表达式，据此设计控制器电路

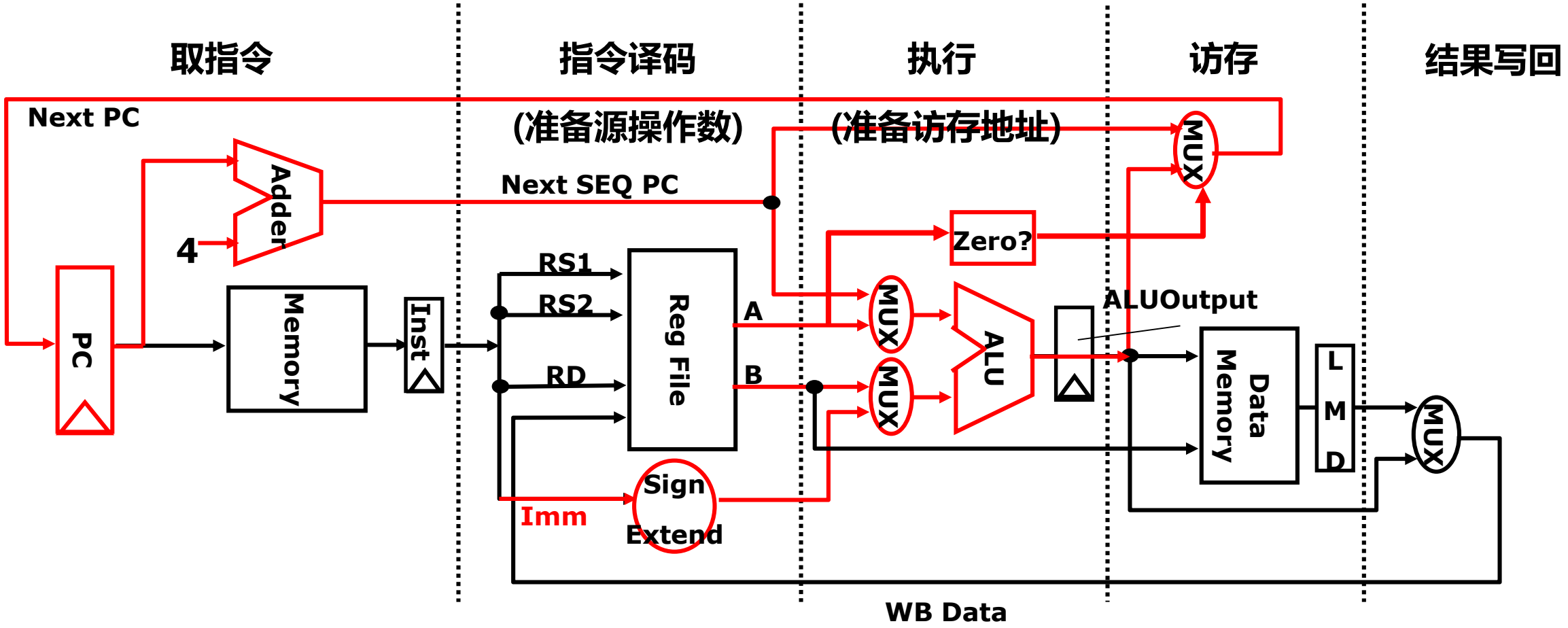
# 2. MIPS指令的取指周期



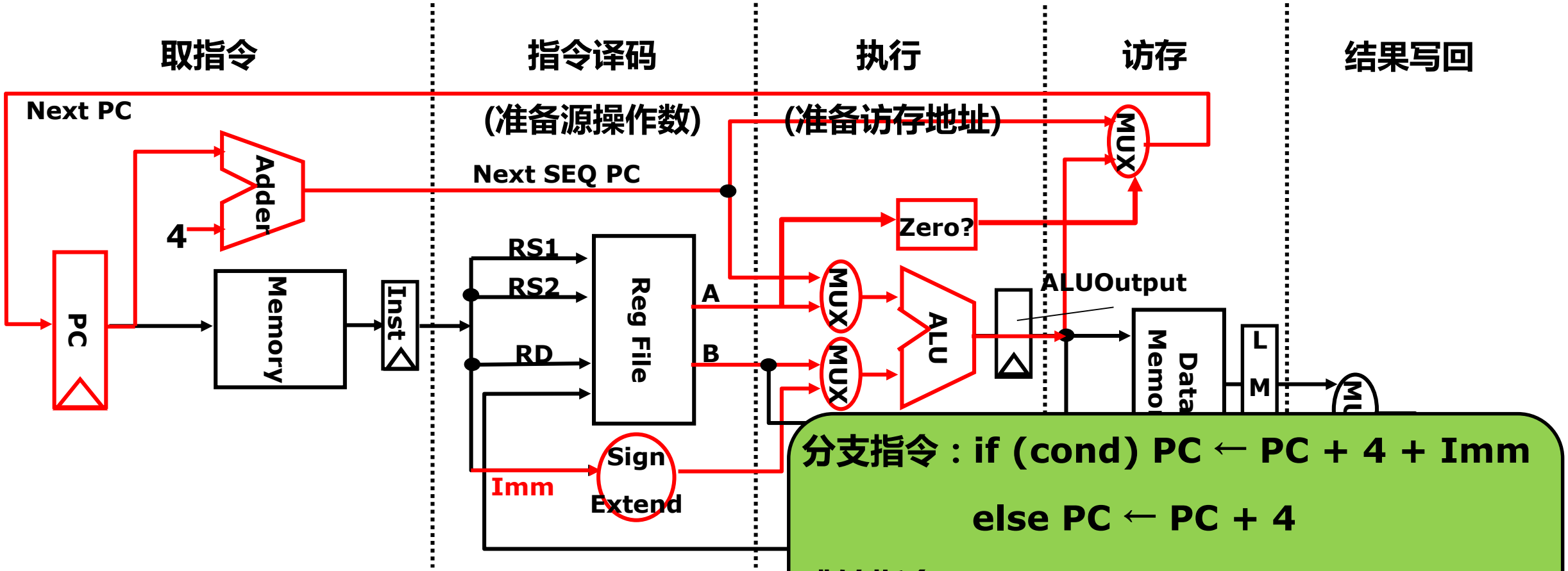
# 2. MIPS指令的取指周期



# 2. MIPS指令的取指周期



# 2. MIPS指令的取指周期



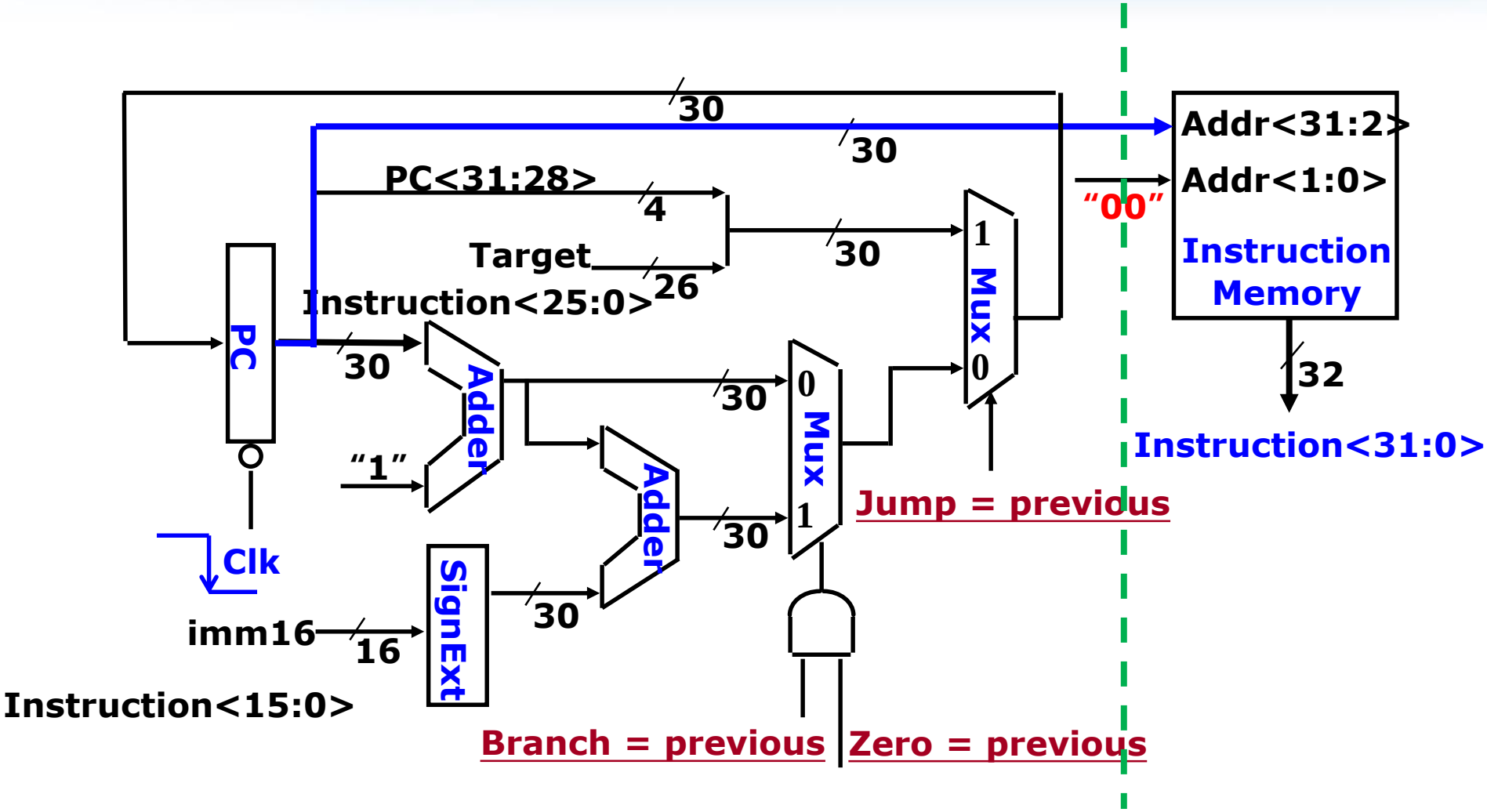

分支指令：if (cond) PC ← PC + 4 + Imm

else PC ← PC + 4

跳转指令：PC ← PC + 4 + Target

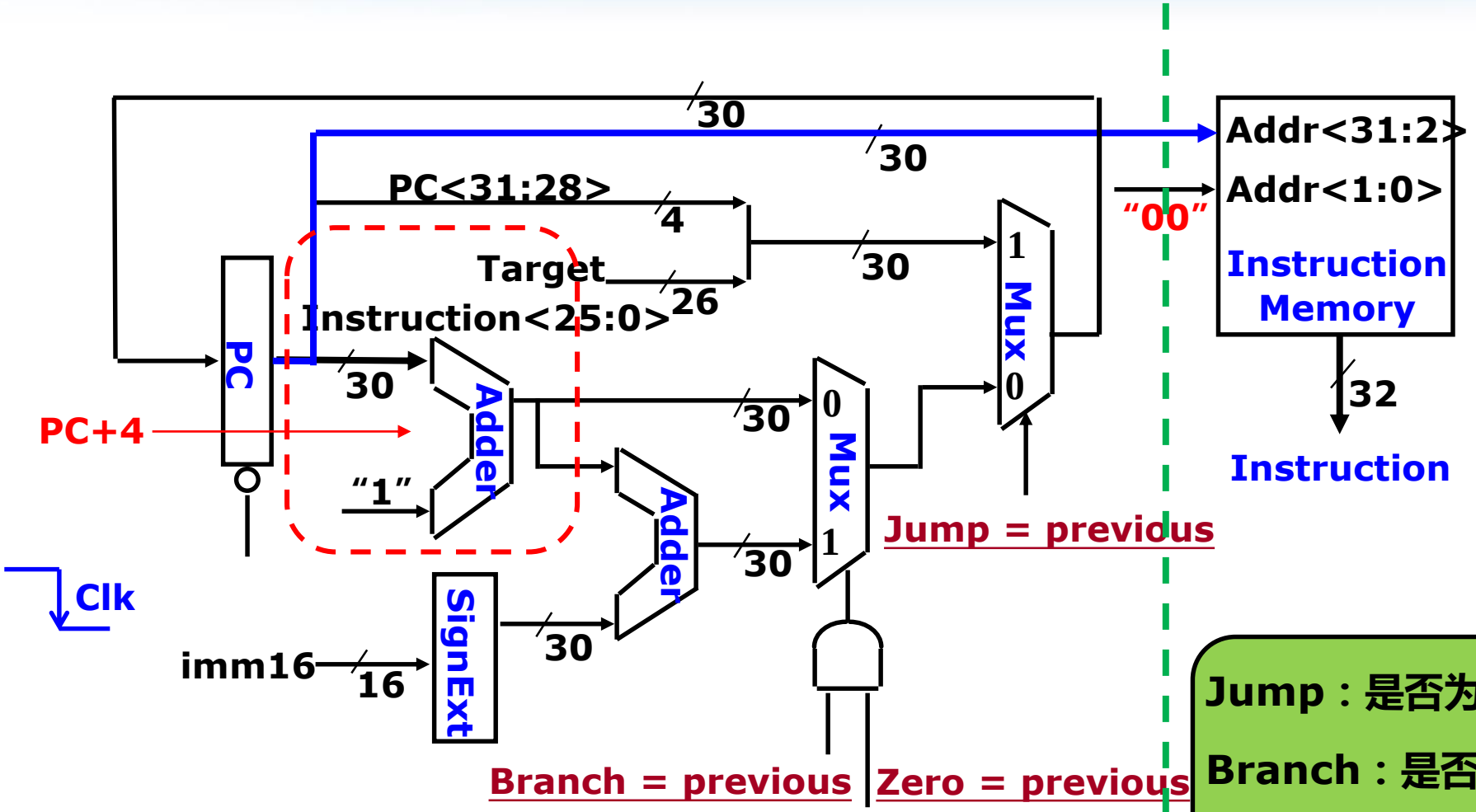
其他指令：PC ← PC + 4

# 2. MIPS指令的取指周期




图：数据通路取指令部分的另一种图示

# 2. MIPS指令的取指周期



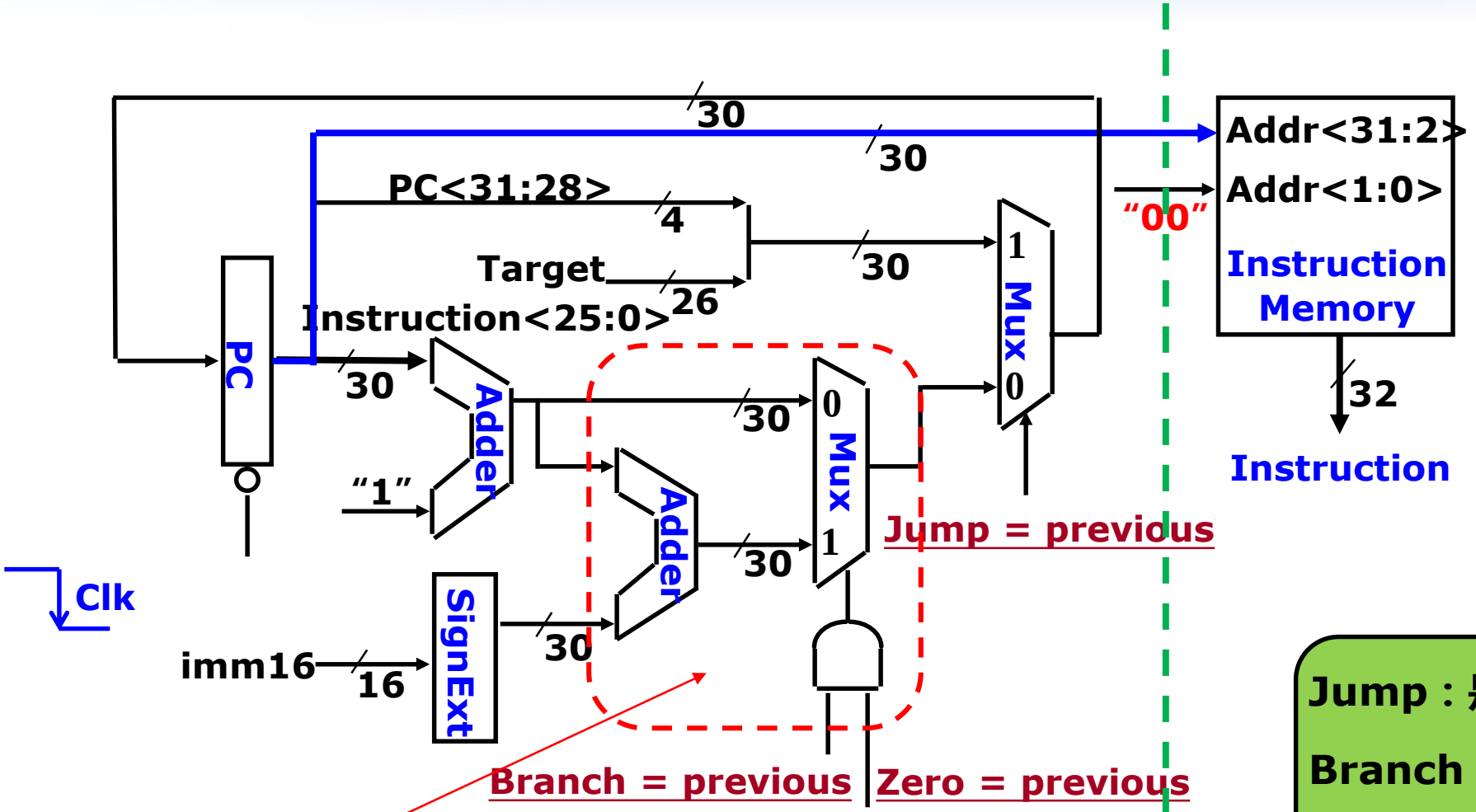

Jump：是否为跳转指令，1-是，0-否

Branch：是否为分支指令，1-是，0-否

Zero：分支转移条件是否为真，1-是，0-否

图：数据通路取指令部分的另一种图示

# 2. MIPS指令的取指周期




if (Branch & zero) PC = PC+4

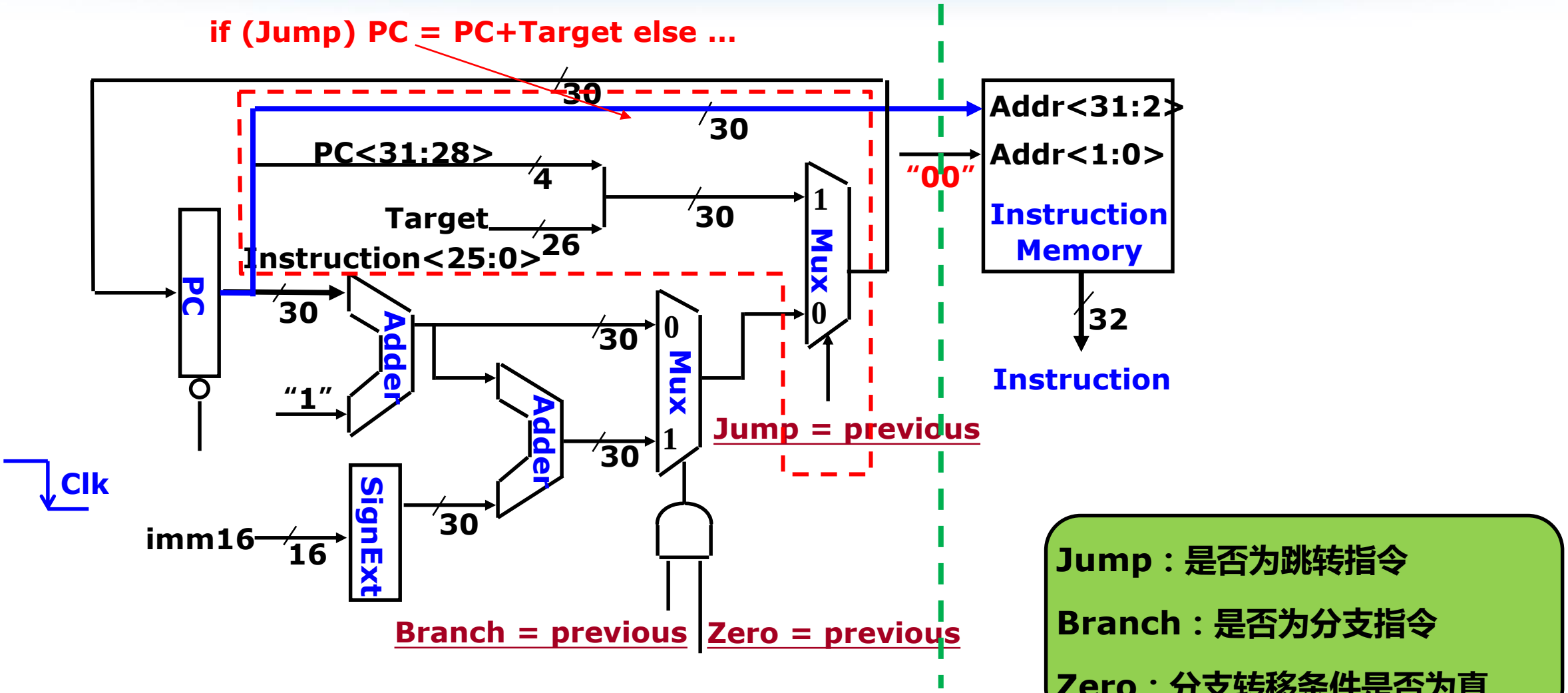
else PC = PC+4+Imm 图：数据通路取指令部分的另一种图示

Jump：是否为跳转指令

Branch：是否为分支指令

Zero：分支转移条件是否为真

# 2. MIPS指令的取指周期



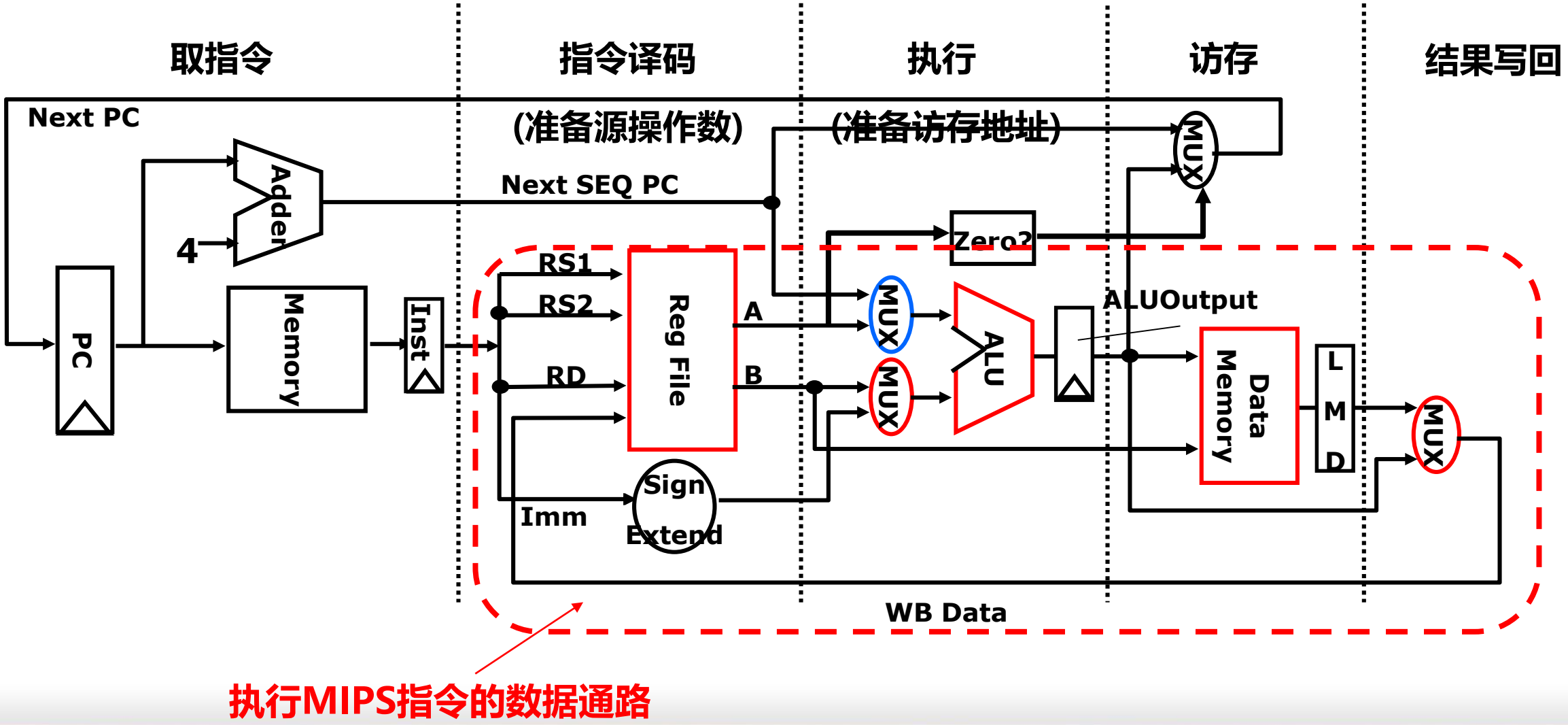

图：数据通路取指令部分的另一种图示

Jump：是否为跳转指令

Branch：是否为分支指令

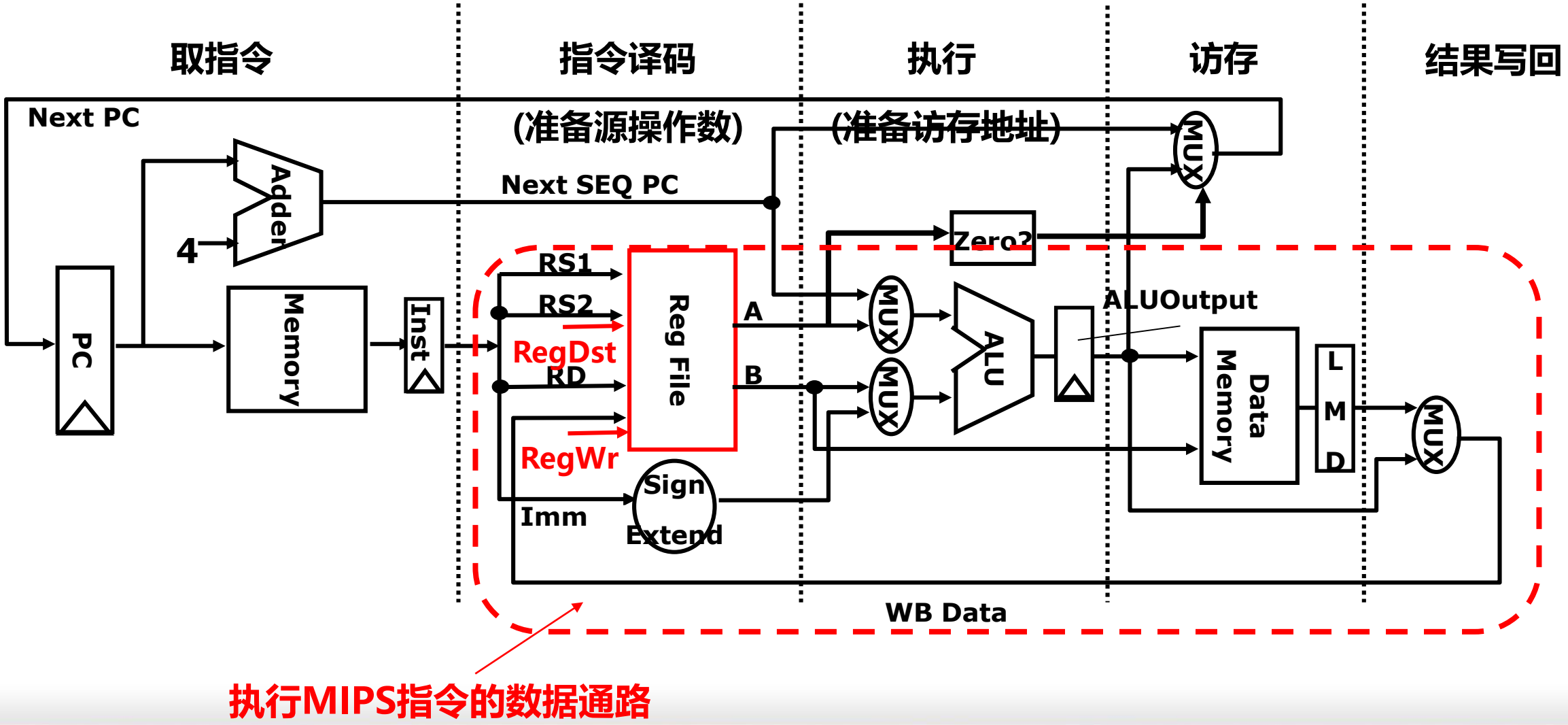
Zero：分支转移条件是否为真

# 3. MIPS指令的执行周期



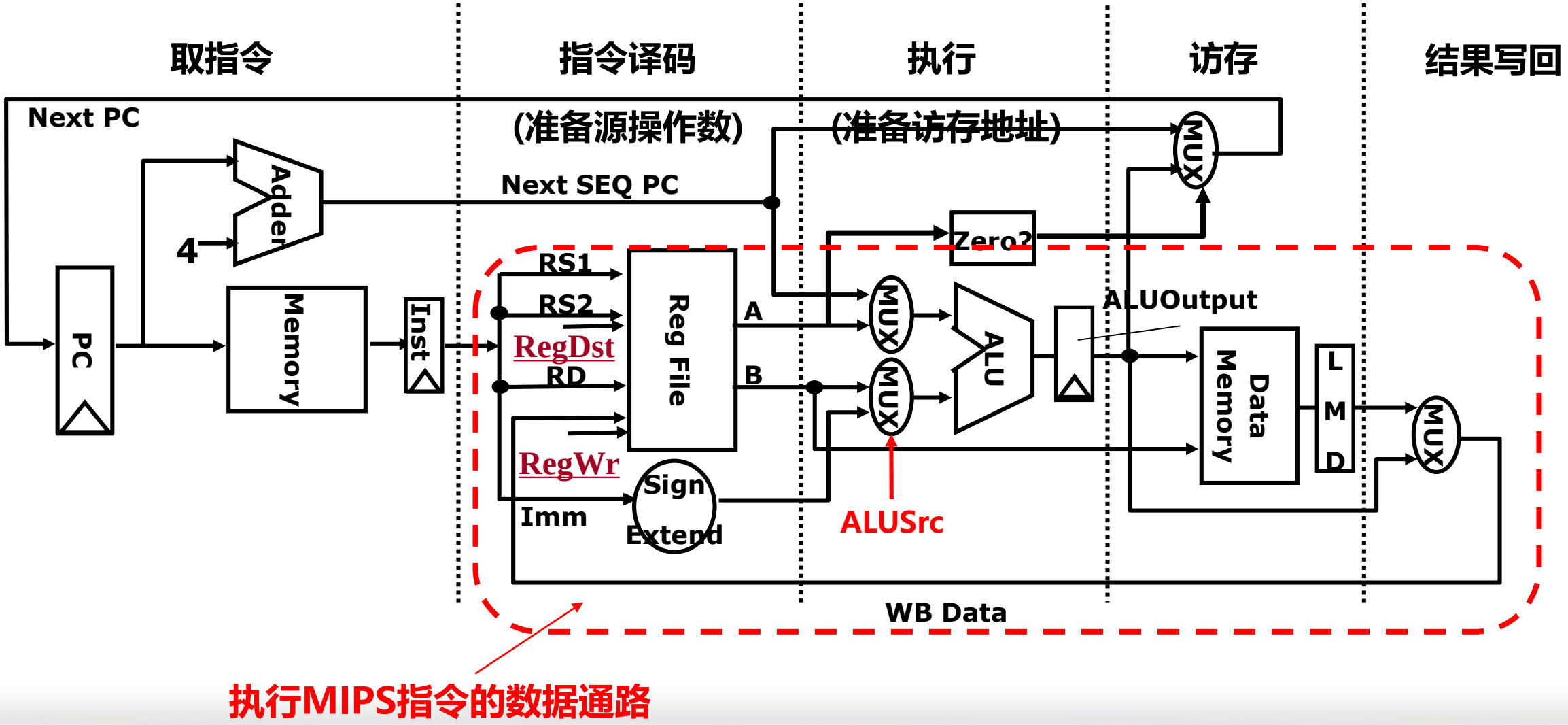

执行MIPS指令的数据通路

# 3. MIPS指令的执行周期

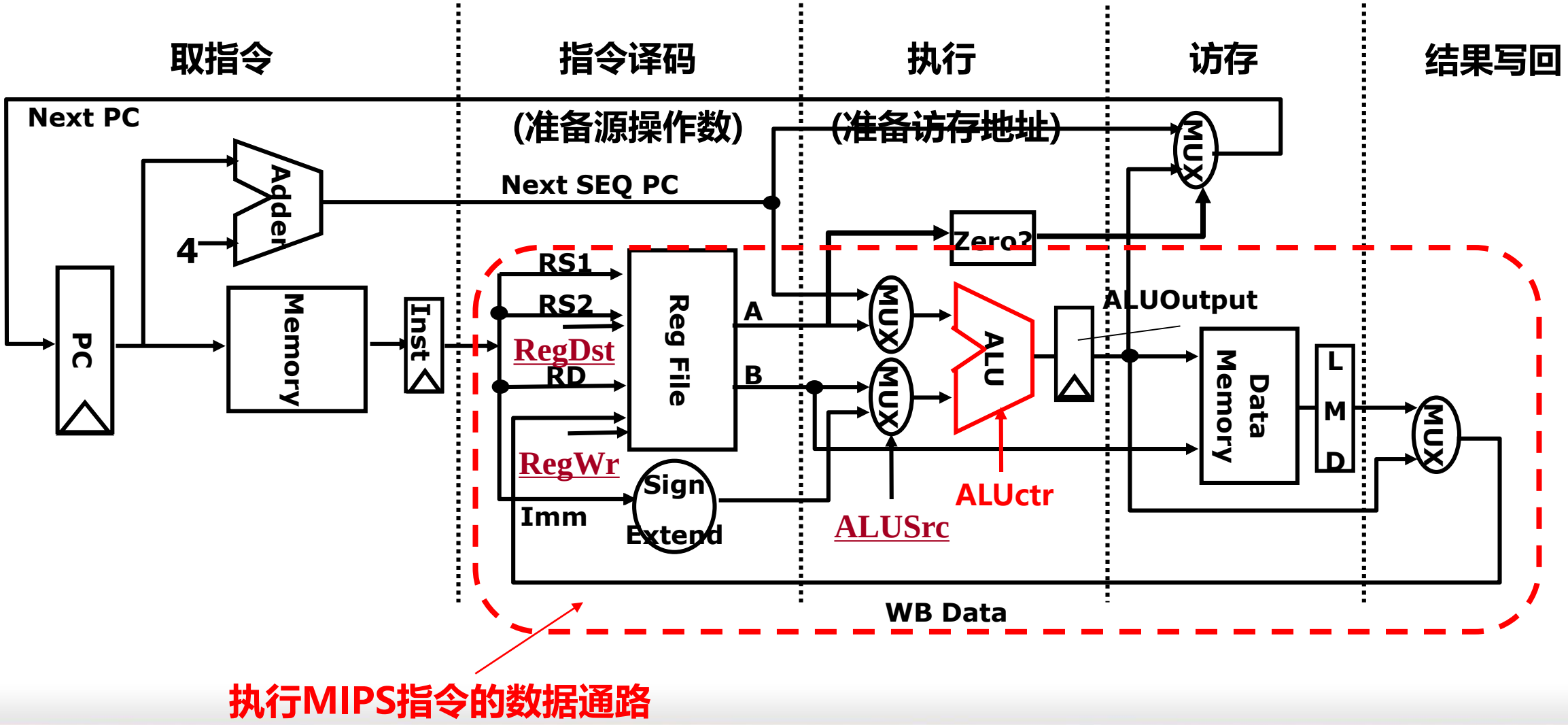



执行MIPS指令的数据通路

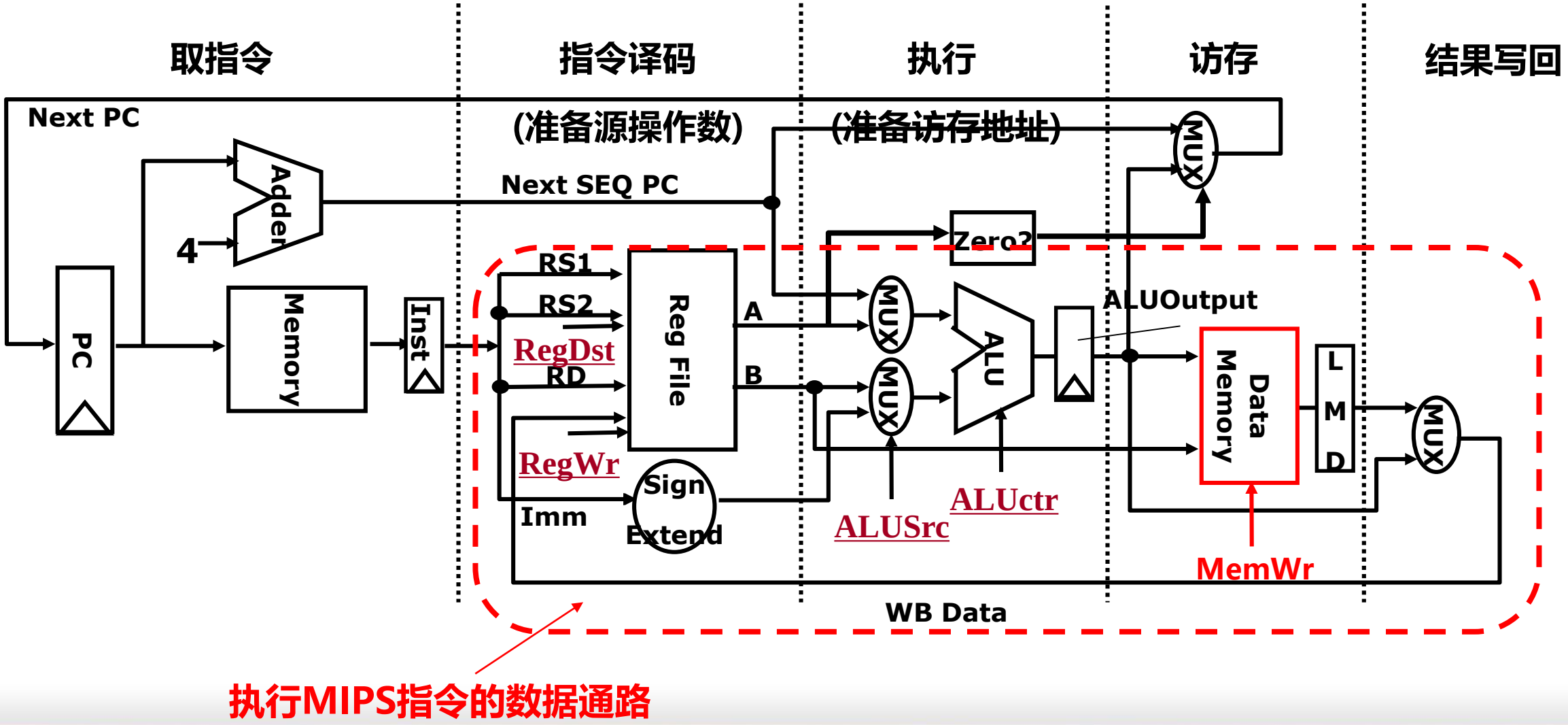
# 3. MIPS指令的执行周期



# 3. MIPS指令的执行周期

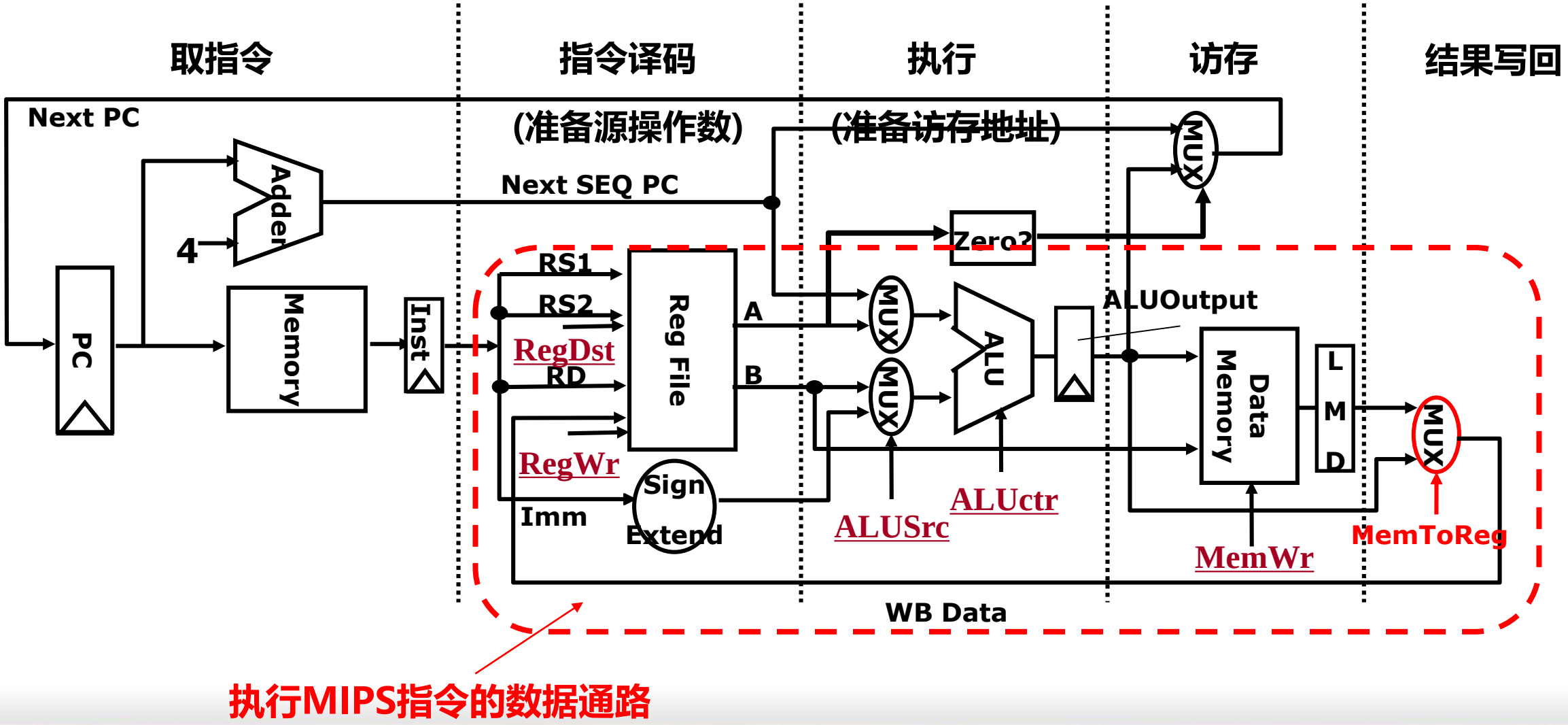


# 3. MIPS指令的执行周期



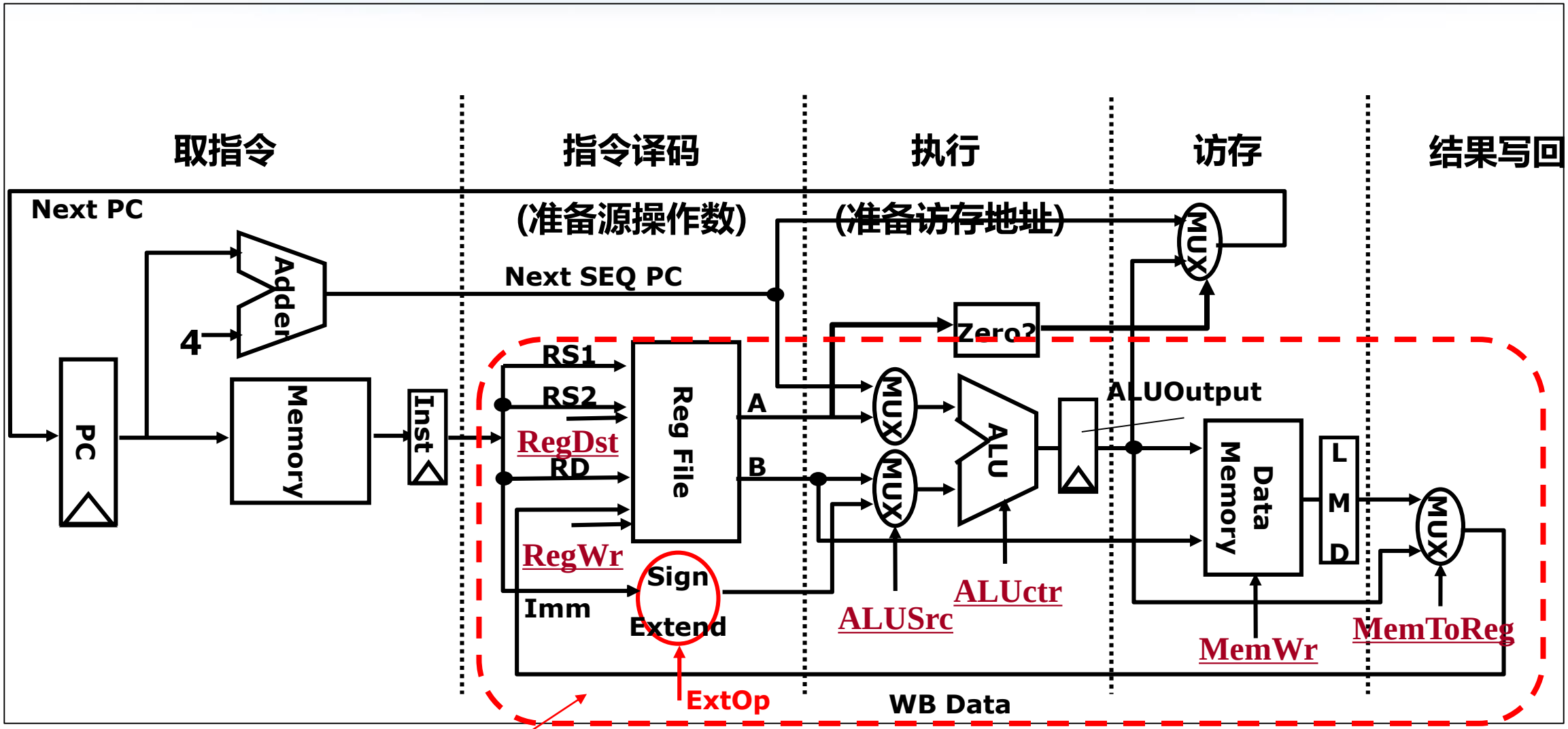

执行MIPS指令的数据通路

# 3. MIPS指令的执行周期



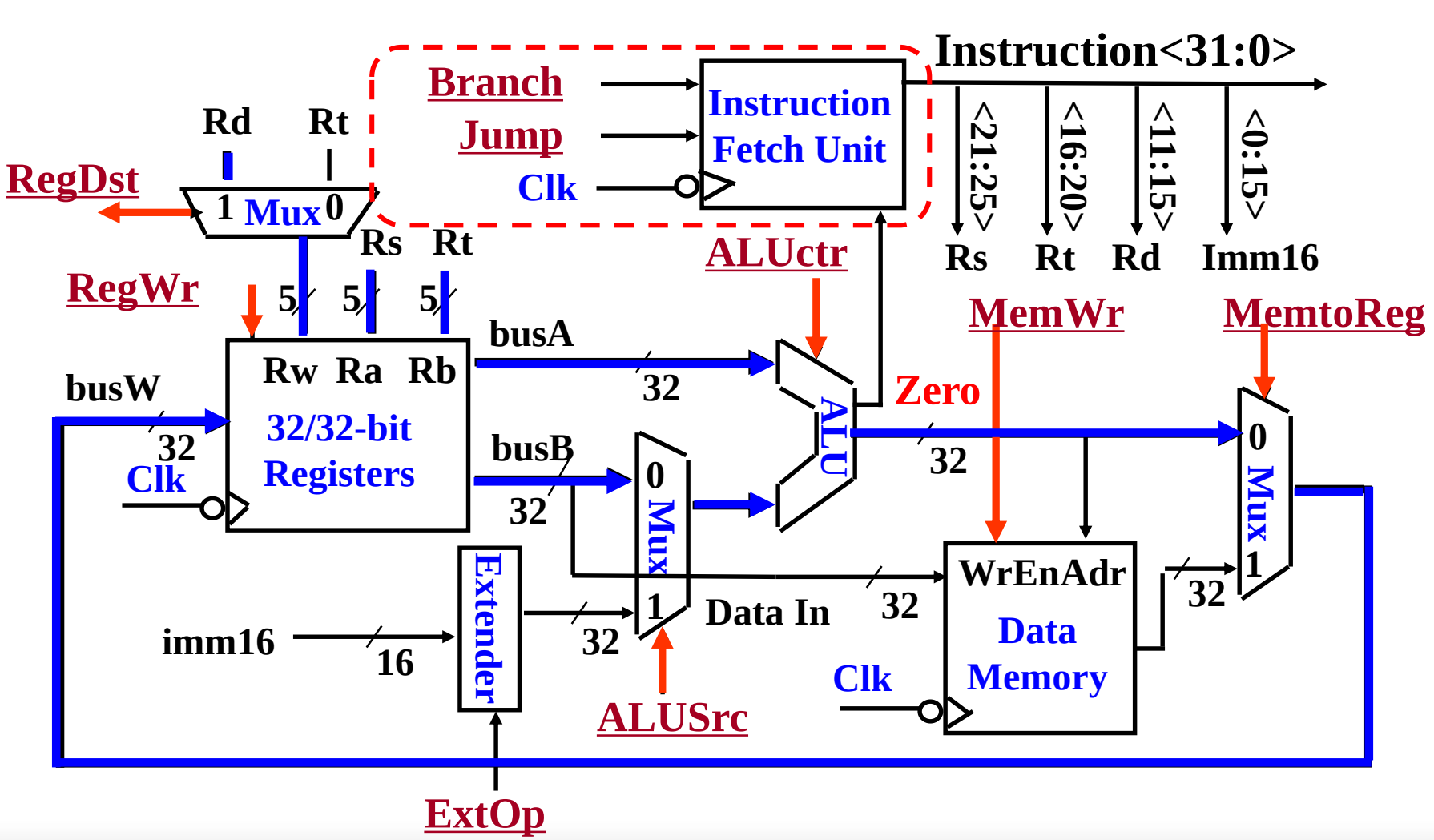

执行MIPS指令的数据通路

# 3. MIPS指令的执行周期




执行MIPS指令的数据通路　　实例指令：ORI R1, R1, #3

# 3. MIPS指令的执行过程



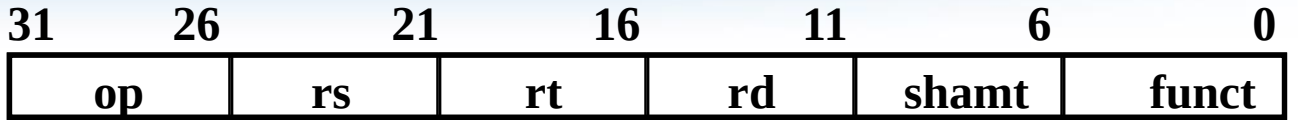

| 名称 | 控制对象 |
|---|---|
| RegDst | 选择目的寄存器 |
| RegWr | 读或写寄存器 |
| ExtOp | 立即数扩展方式 |
| ALUSrc | ALU单元的输入 |
| MemWr | 读或写存储器 |
| MemTo Reg | 选择要写回寄存器的数据 |
| ALUctr | ALU单元的操作 |
| Branch | 是否分支指令 |
| Jump | 是否跳转指令 |
| Zero | 分支转移条件是否为真 |

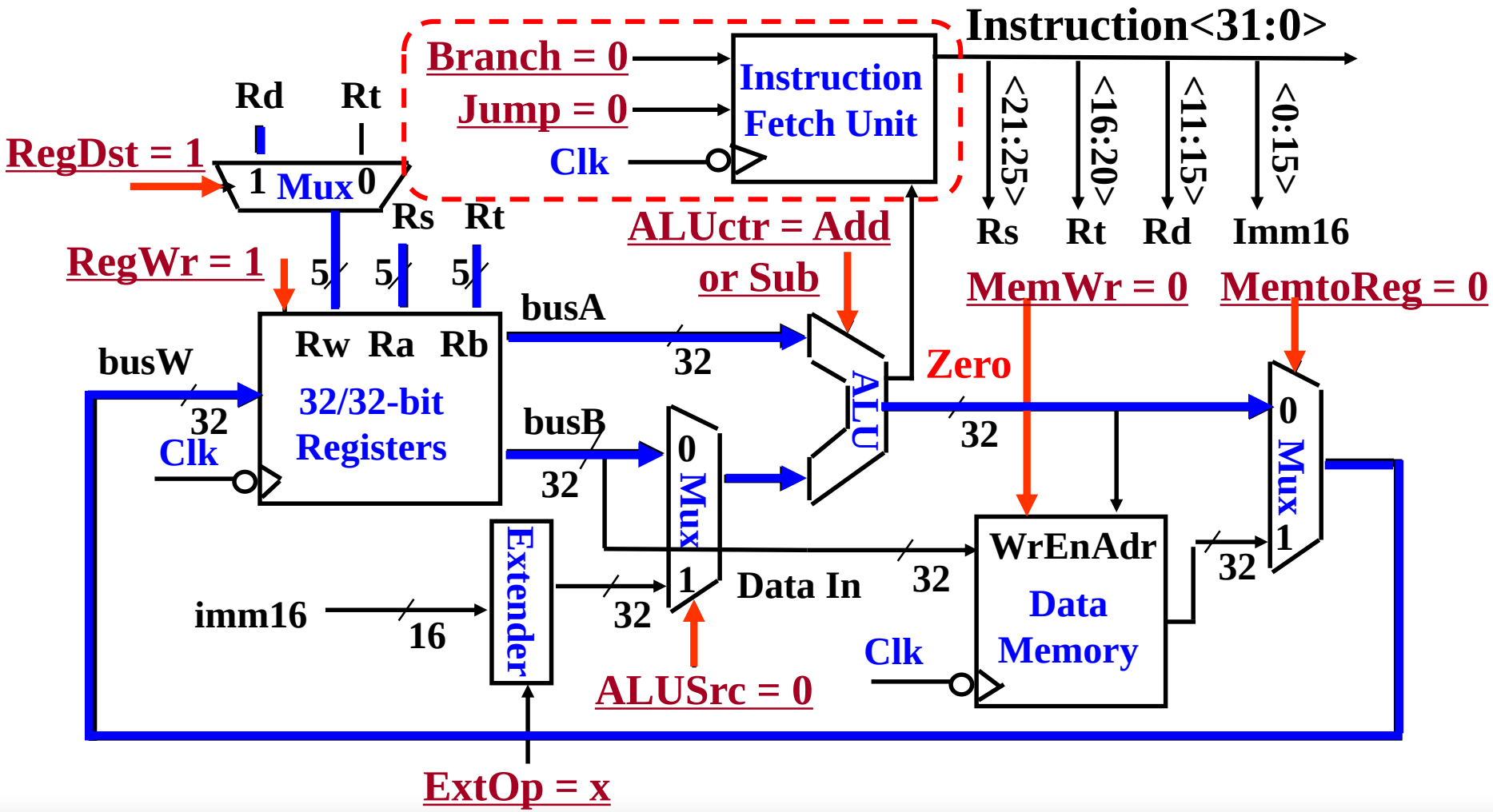
# 3. MIPS指令的执行过程

- ADD/SUB
  - R[rd] ← R[rs] +/− R[rt]
- ORI
  - R[rt] ← R[rs]  or  ZeroExt[imm16]
- Load
  - R[rt] ← M[ R[rs] + SignExt[imm16] ]
- Store
  - M[ R[rs] + SignExt[imm16] ] ← R[rt]
- Branch
  - if  (R[rs]−R[rt]==0) then Zero←1; else Zero←0

# 3. MIPS指令的执行过程——ADD/SUB指令

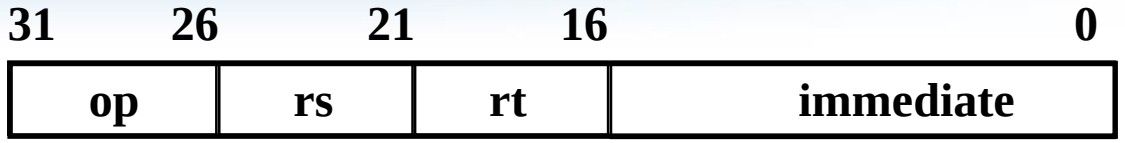
❑R[rd] ← R[rs] +/− R[rt]

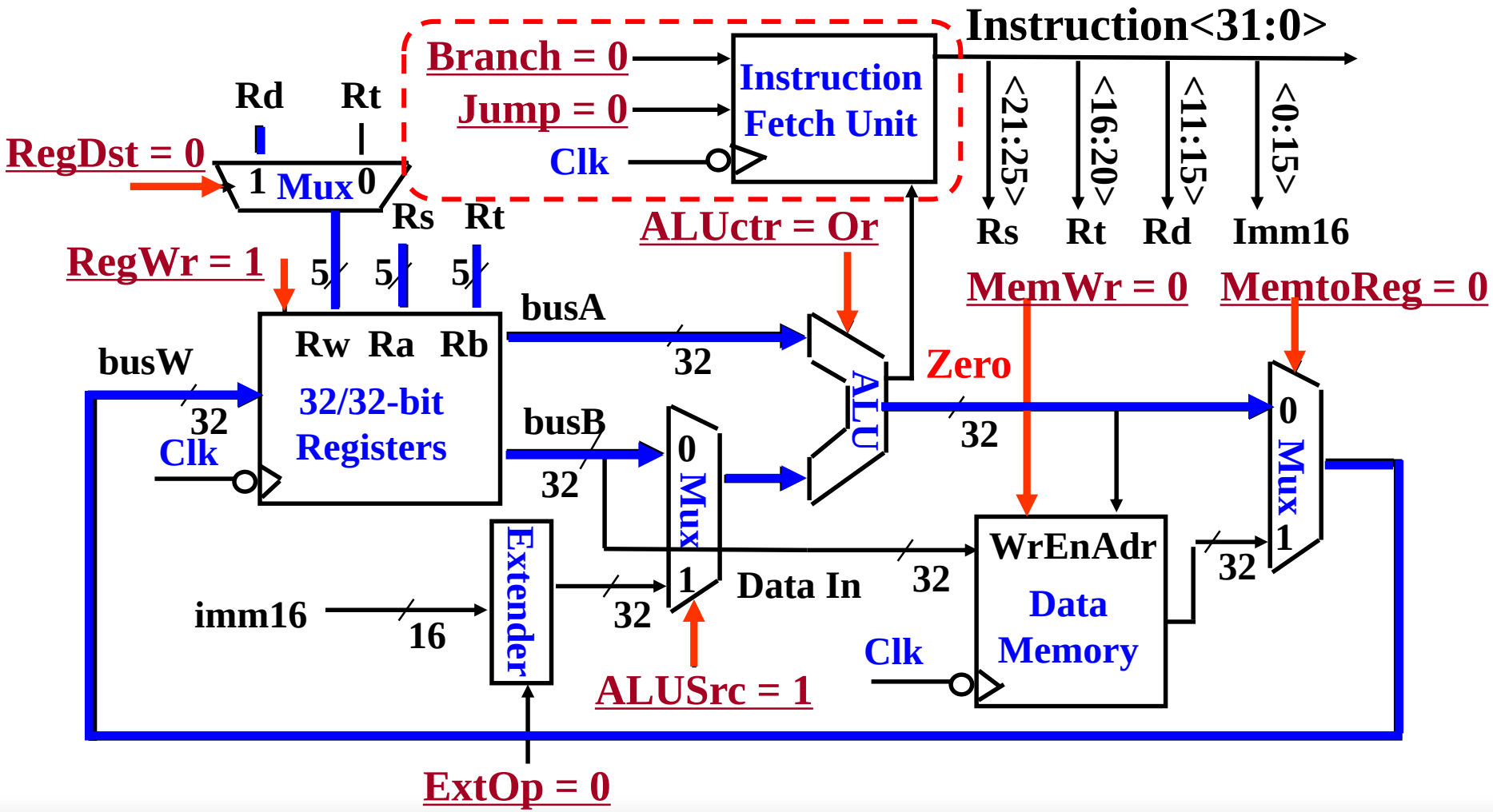


# 3. MIPS指令的执行过程——ORI指令

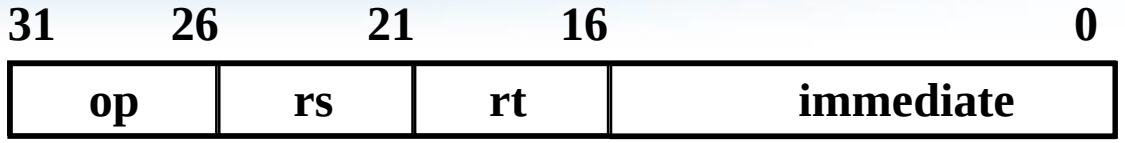
❑R[rt]←R[rs] or ZeroExt[imm16]

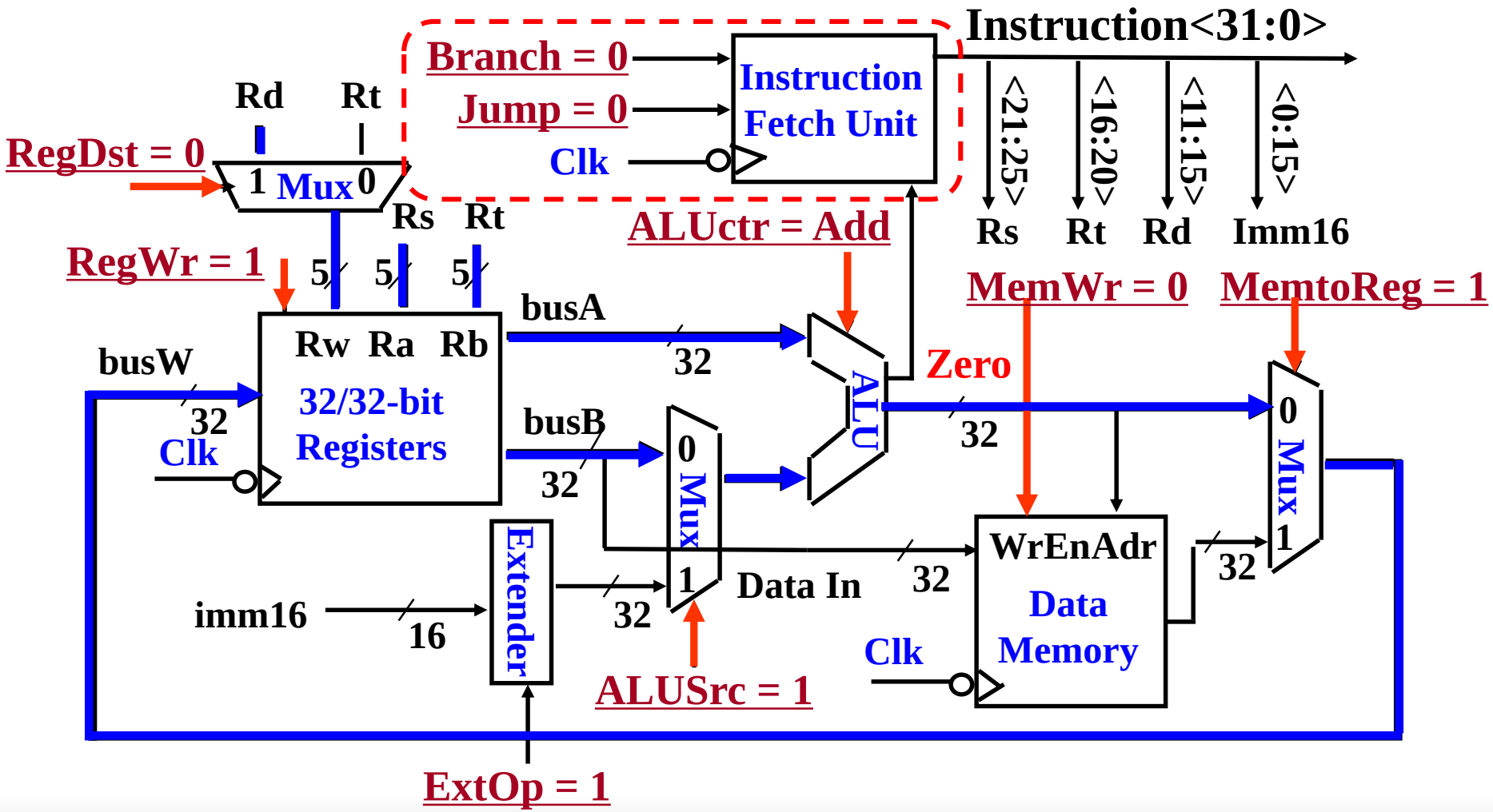


# 3. MIPS指令的执行过程——Load指令

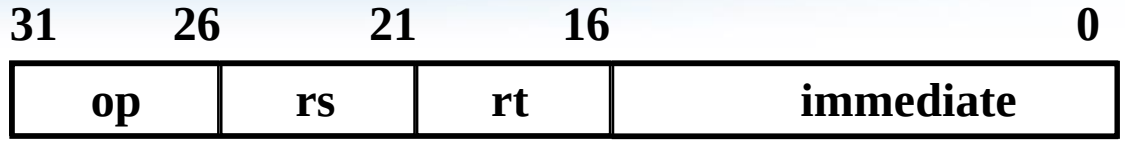
❑R[rt]←M[R[rs]+SignExt[imm16]]

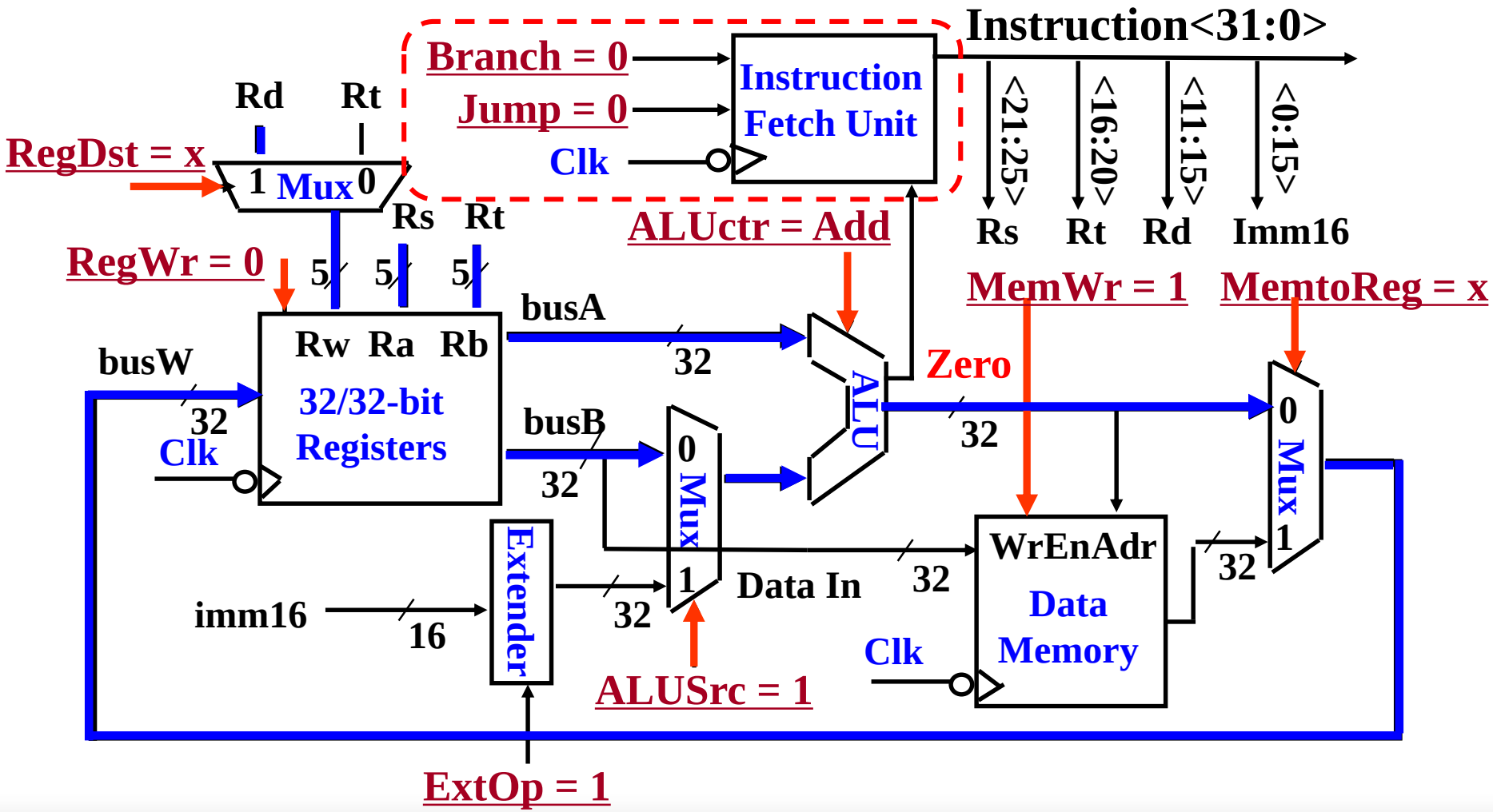


# 3. MIPS指令的执行过程——Store指令

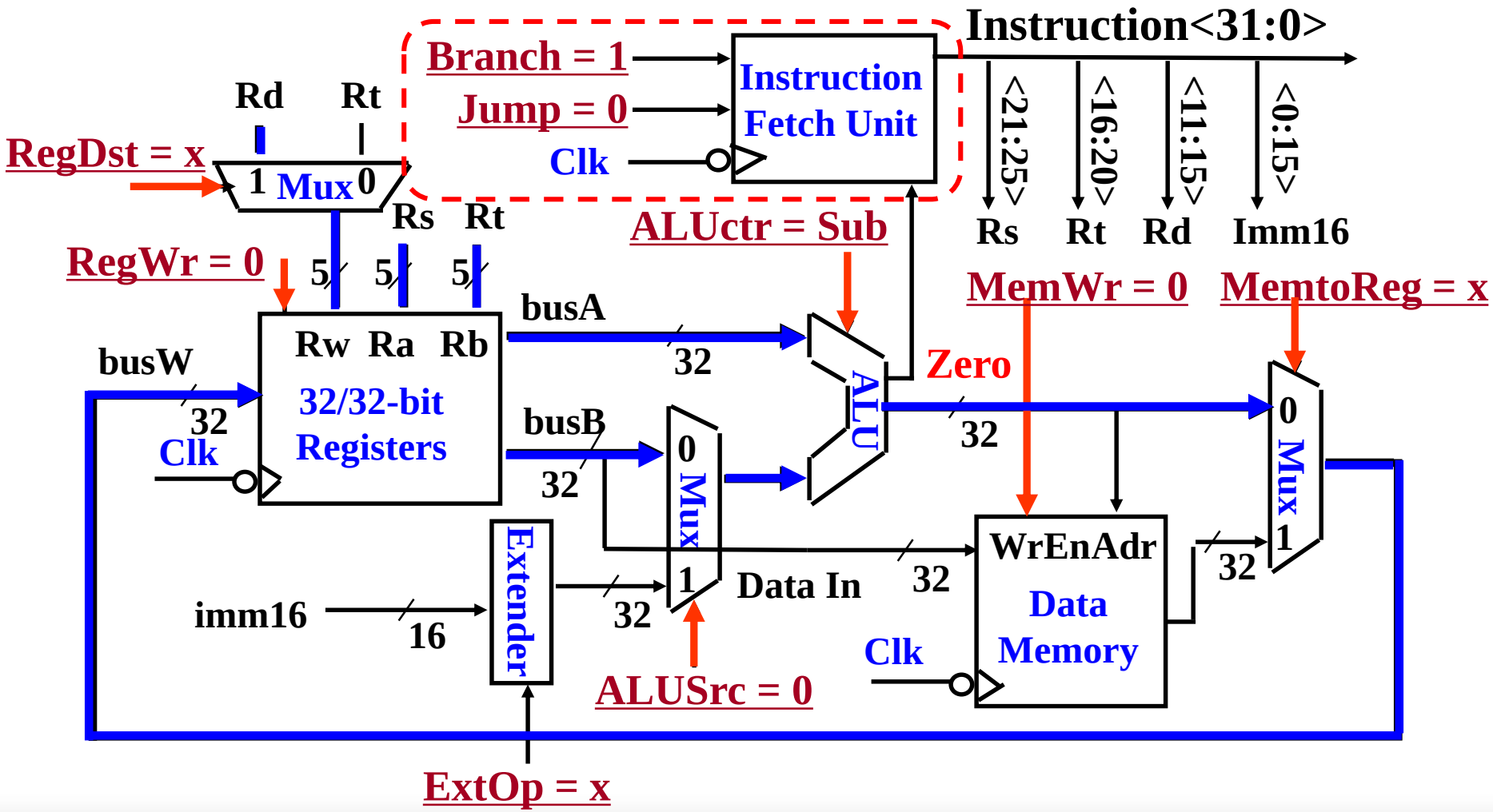
❑M[R[rs]+SignExt[imm16]]←R[rt]



# 3. MIPS指令的执行过程——Branch指令

- if (R[rs]－R[rt]==0) then Zero←1; else Zero←0



# 4. 控制器生成

❑综合分析结果，得到反映指令和控制信号取值的关系表

| func / op | 10 0000 / 00 0000 | 10 0010 / 00 0000 | We Don't Care / 00 1101 | We Don't Care / 10 0011 | We Don't Care / 10 1011 | We Don't Care / 00 0100 | We Don't Care / 00 0010 |
|---|---|---|---|---|---|---|---|
| | add | sub | ori | lw | sw | beq | jump |
| RegDst | 1 | 1 | 0 | 0 | x | x | x |
| ALUSrc | 0 | 0 | 1 | 1 | 1 | 0 | x |
| MemtoReg | 0 | 0 | 0 | 1 | x | x | x |
| RegWrite | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| MemWrite | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| Branch | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Jump | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| ExtOp | x | x | 0 | 1 | 1 | x | x |
| ALUctr<2:0> | Add | Subtract | Or | Add | Add | Subtract | xxx |