



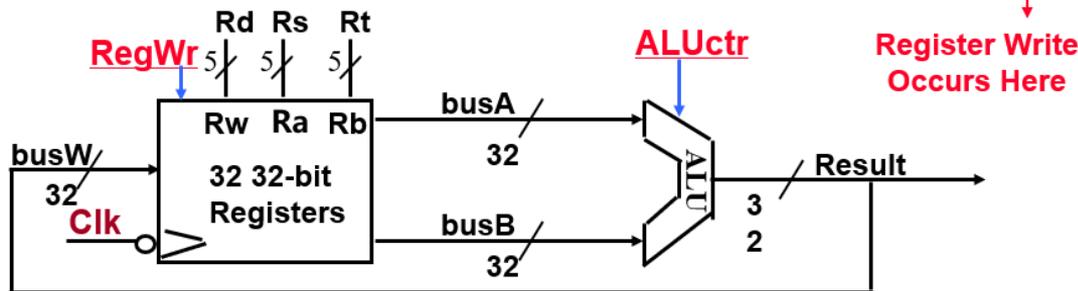
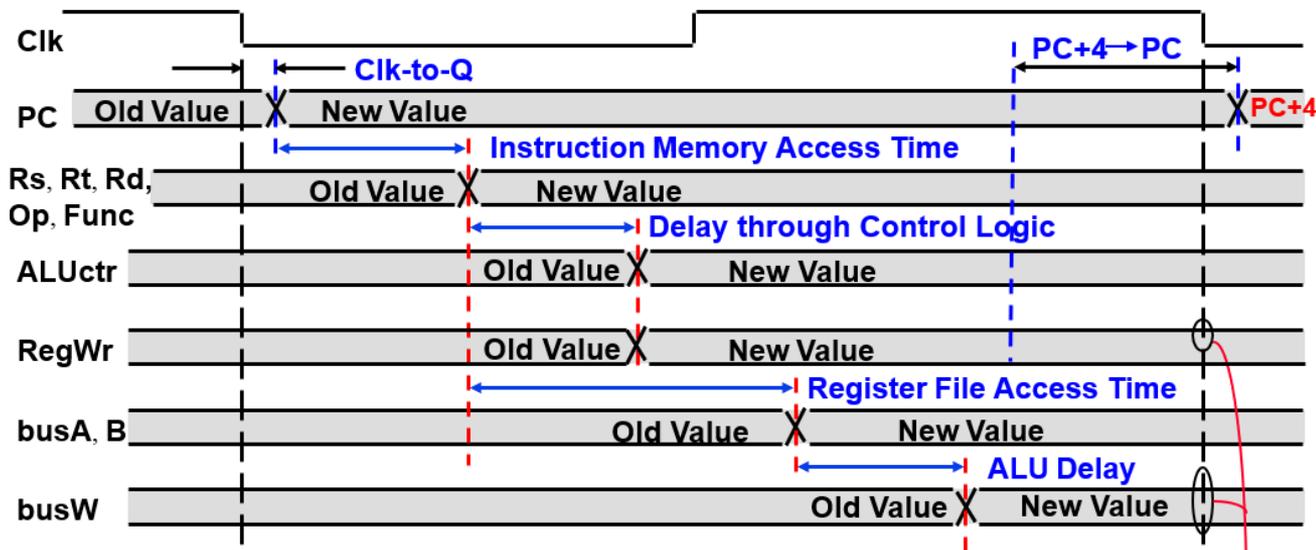
■ 第五讲 性能应用

翟象平

电邮: blueicezhaixp@nuaa.edu.cn

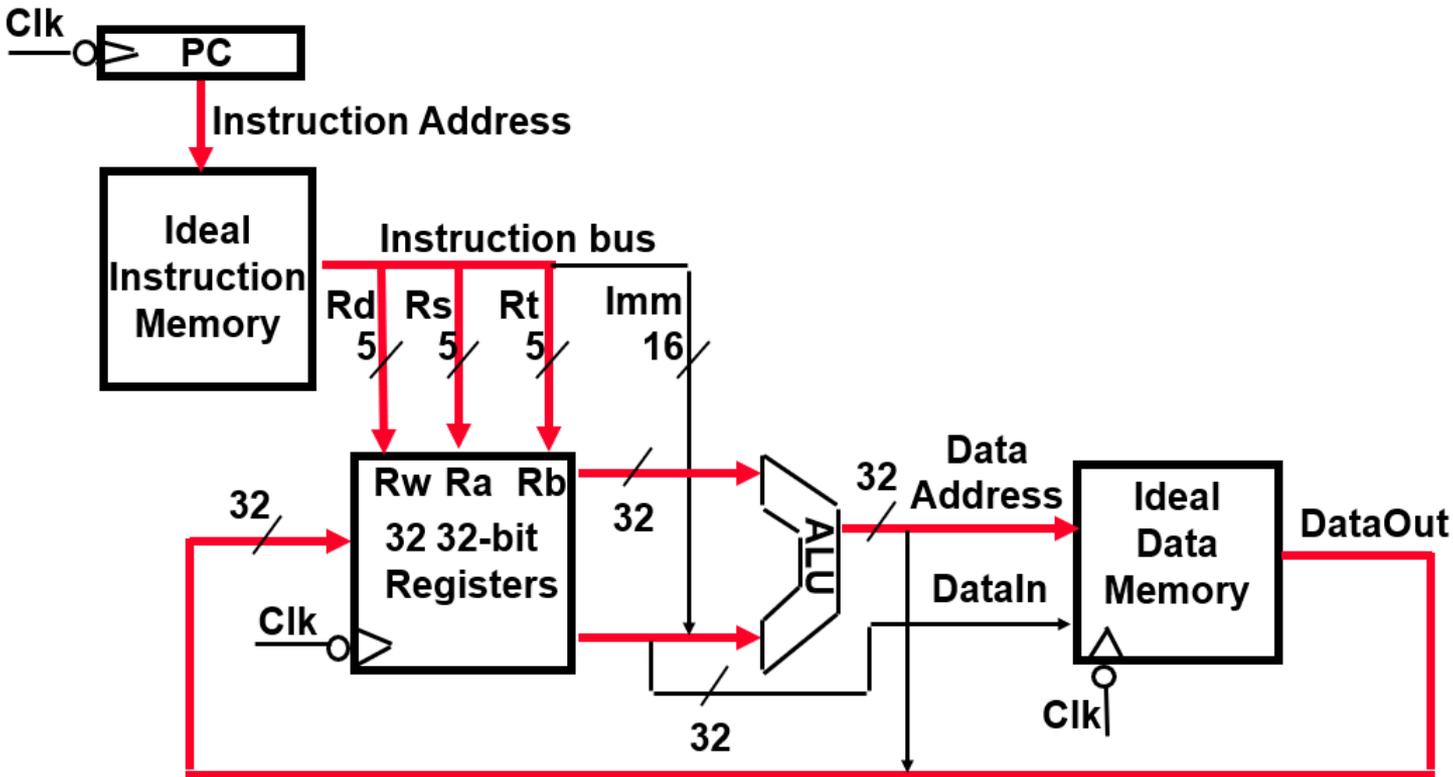
个人主页: <http://cyber.nuaa.edu.cn>

Register-Register (R型指令) 时序



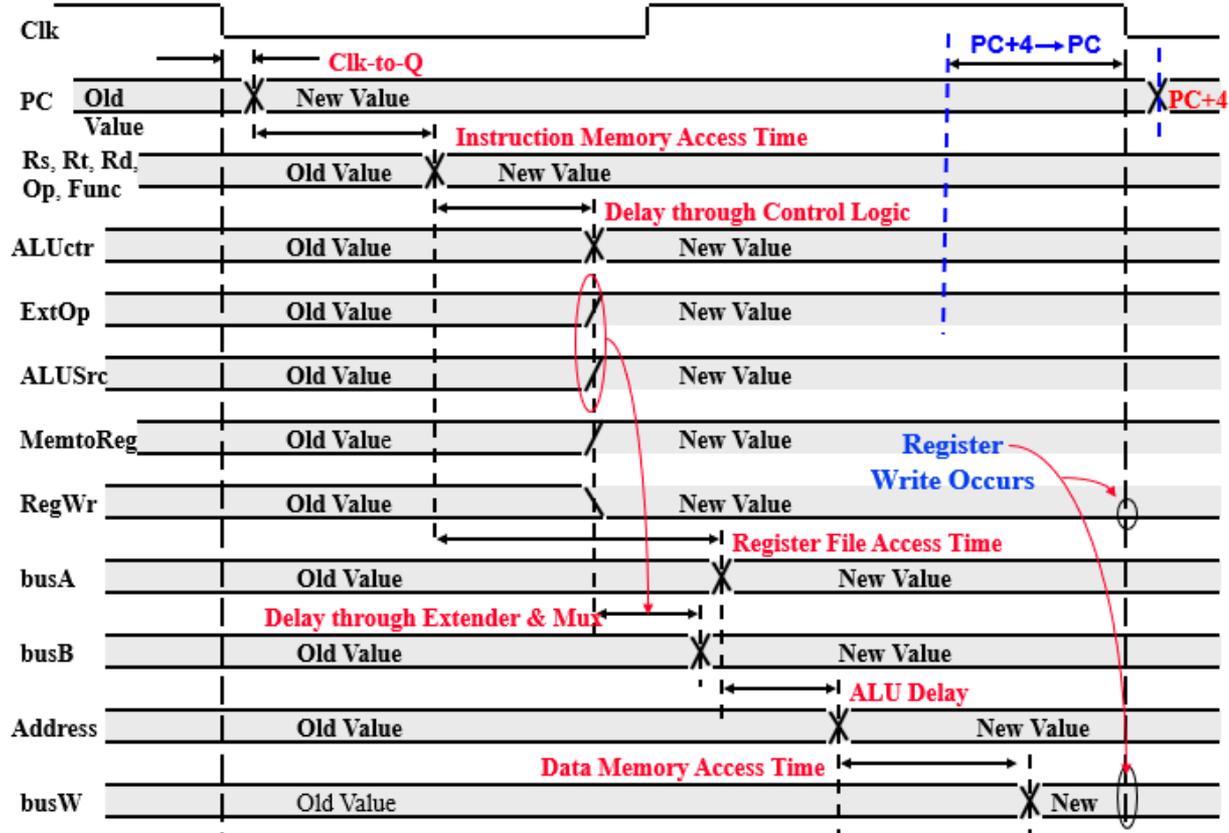


LW 指令执行过程





lw时间最长作为时钟周期





单周期计算机的性能

- 单周期处理器的CPI为多少？
 - 其他条件一定的情况下，CPI越小，则性能越好！
 - $CPI=1$ ，不是很好吗？
- 单周期处理器的性能会不会很好？为什么？
 - 计算机的性能除CPI外，还取决于时钟周期的宽度
 - 单周期处理器的时钟宽度为最复杂指令的执行时间
 - 很多指令可以在更短的时间内完成
 - 非load/store指令无需访问数据内存
 - J-指令无需访问寄存器

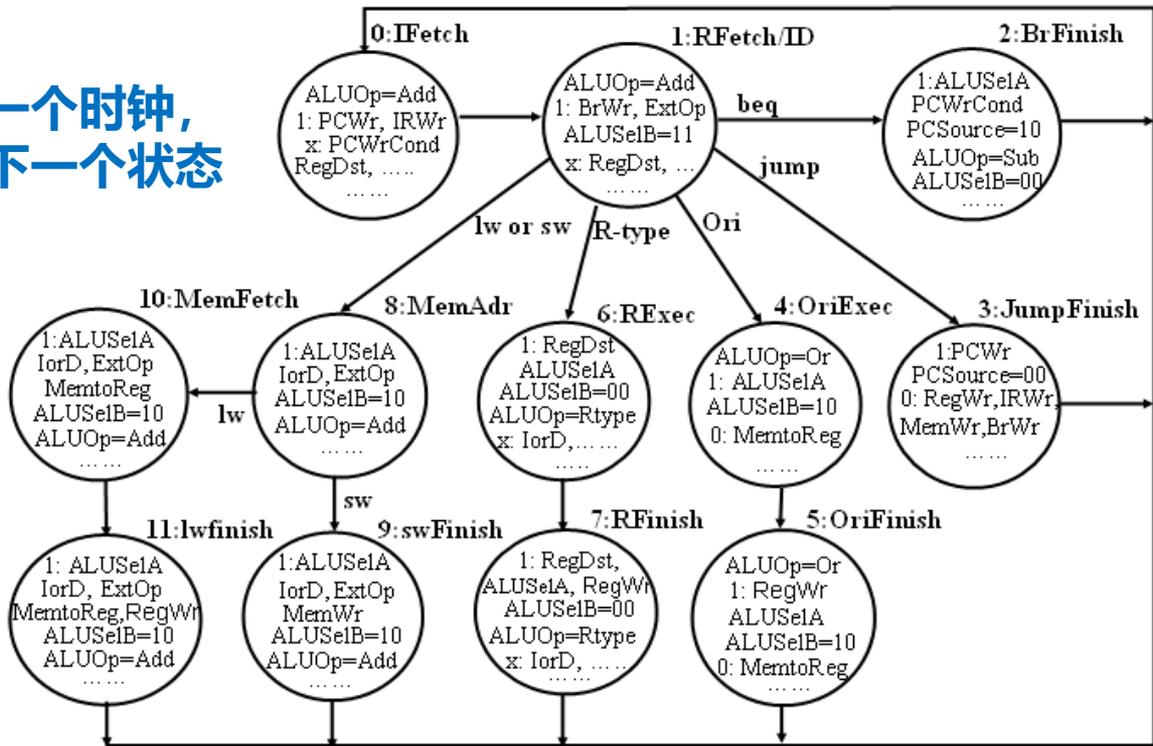


多周期状态转换图

时钟周期固定、时钟周期数可变

每个状态下，输出的控制信号有相应的不同取值！

每来一个时钟，
进入下一个状态



R-4,
ori-4,
beq-3,
Jump-3,
lw-5,
sw-4



计算机的性能举例

假设在单周期处理器中，各主要功能单元的操作时间为：

- 存储单元：200ps
- ALU和加法器：100ps
- 寄存器堆（读/写）：50ps

假设MUX、控制单元、PC、扩展器和传输线路都没有延迟，则下面实现方式中，哪个更快？快多少？

- (1) 每条指令在一个固定长度的时钟周期内完成
- (2) 每条指令在一个时钟周期内完成，但时钟周期仅为指令所需，也即为可变的（实际不可行，只是为了比较）

假设程序中各类指令占比：25%取数、10%存数、45%ALU、15%分支、5%跳转

计算机的性能举例



Instruction class	Functional units used by the instruction class				
R-type	Instruction fetch	Register access	ALU	Register access	
Load word	Instruction fetch	Register access	ALU	Memory access	Register access
Store word	Instruction fetch	Register access	ALU	Memory access	
Branch	Instruction fetch	Register access	ALU		
Jump	Instruction fetch				

CPU执行时间=指令条数 × CPI × 时钟周=指令条数 × 时钟周期

Instruction class	Instruction memory	Register read	ALU operation	Data memory	Register write	Total
R-type	200	50	100	0	50	400 ps
Load word	200	50	100	200	50	600 ps
Store word	200	50	100	200		550 ps
Branch	200	50	100	0		350 ps
Jump	200					200 ps



计算机的性能举例

- 对于方式（1），时钟周期由最长指令来决定，应该是load指令，为600ps
- 对于方式（2），时钟周期取各条指令所需时间，时钟周期从600ps至200ps

根据各类指令的频度，计算出平均时钟周期长度为：

CPU时钟周期=600x25%+550x10%+400x45%+350x15%+200x5%=447.5ps

$$\text{CPU性能比} = \frac{\text{方式(1)的CPU执行时间}}{\text{方式(2)的CPU执行时间}} = \frac{\text{方式(1)的CPU 时钟周期数}}{\text{方式(2)的CPU 时钟周期数}} = \frac{600}{447.5} = 1.34$$

可变时钟周期的性能是定长周期的1.34倍！但是，对每类指令采用可变长时钟周期实现非常困难，而且所带来的额外开销会很大



Q & A

THANKS