

计算机系统结构

第四讲 多指令流出技术

谢长生

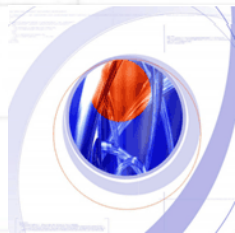
武汉光电国家研究中心



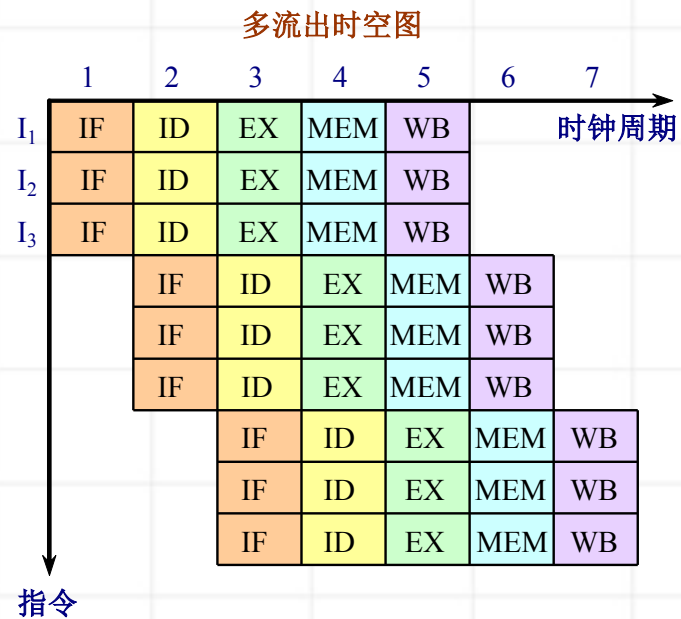
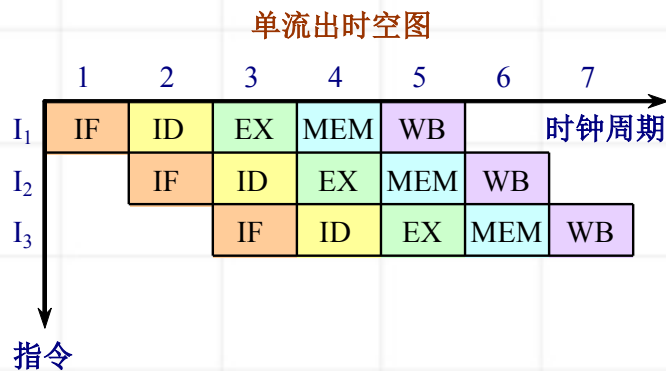
Computer Architecture

1.工作动机

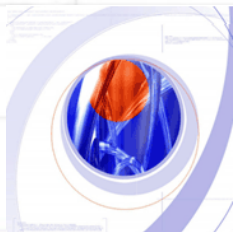
- 实际CPI=理想CPI+各种冲突引入的停顿
 - 前面讲的技术主要集中后者（消除/减少各种冲突引入的停顿）
 - 也可以考虑通过提升前者，改进实际CPI
- 前面讲的所有流水线都是单流出的，即一个时钟周期内至多流出一条指令，理想CPI=1
- 本节描述的技术：一个时钟周期内流出多条指令，理想CPI<1。



2.单流出与多流出对比

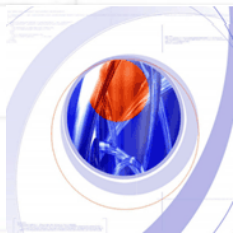


单流出和多流出处理器执行指令的时空图



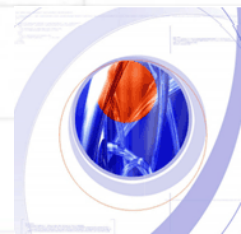
3. 多流出处理机有两种基本风格

- 超标量 (Superscalar)
 - 在每个时钟周期流出的指令条数不固定，依代码的具体情况而定。
(有个上限)
 - 设这个上限为 n ，就称该处理机为 n -流出。
 - 可以通过编译器进行静态调度，也可以基于Tomasulo算法进行动态调度。
- 超长指令字VLIW (Very Long Instruction Word)
 - 在每个时钟周期流出的指令条数是固定的，这些指令构成一条长指令或者一个指令包。
 - 指令包中，指令之间的并行性是通过指令显式地表示出来的。
 - 指令调度是由编译器静态完成的。



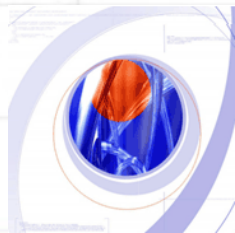
4. 超标量处理机的优势

- 超标量结构对程序员是透明的，处理机能自己检测下一条指令能否流出，不需要由编译器或专门的变换程序对程序中的指令进行重新排列；
- 即使是没有经过编译器针对超标量结构进行调度优化的代码或是旧的编译器生成的代码也可以运行，当然运行的效果不会很好。
 - 要想达到很好的效果，方法之一：使用动态超标量调度技术。



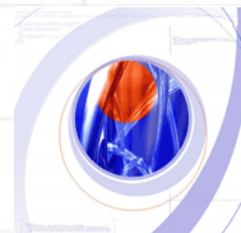
5. 基于静态调度的超标量流水线技术

- 在典型的超标量处理器中，每个时钟周期可流出1到8条指令。
- 指令按序流出，在流出时进行冲突检测。
 - 由硬件检测当前流出的指令之间是否存在冲突以及当前流出的指令与正在执行的指令是否有冲突。



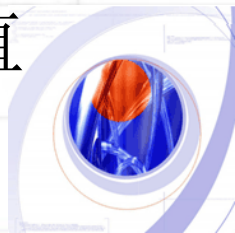
6-1. 基于静态调度的超标量流水线技术

指令类型	流水线工作情况							
整数指令	IF	ID	EX	MEM	WB			
浮点指令	IF	ID	EX	EX	MEM	WB		
整数指令		IF	ID	EX	MEM	WB		
浮点指令		IF	ID	EX	EX	MEM	WB	
整数指令			IF	ID	EX	MEM	WB	
浮点指令			IF	ID	EX	EX	MEM	WB
整数指令				IF	ID	EX	MEM	WB
浮点指令				IF	ID	EX	EX	MEM



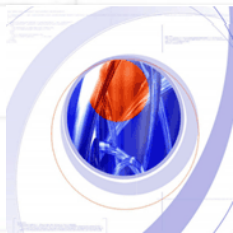
6-2. 限制超标量流水线的性能发挥的障碍

- Load指令
 - Load后续3条指令都不能使用其结果，否则就会引起停顿。
- 分支延迟
 - 如果分支指令是流出包中的第一条指令，则其直接影响后续3条指令；
 - 否则就是流出包中的第二条指令，其其直接影响后续2条指令。



7. 超长指令字技术

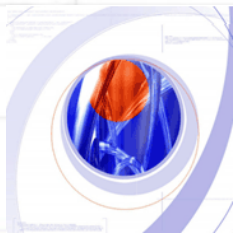
- 把能并行执行的多条指令组装成一条很长的指令；
(100多位到几百位)
- 设置多个功能部件；
- 指令字被分割成一些字段，每个字段称为一个操作槽，直接独立地控制一个功能部件；
- 在VLIW处理机中，在指令流出时不需要进行复杂的冲突检测，而是依靠编译器全部安排好了。



8. VLIW存在的一些问题

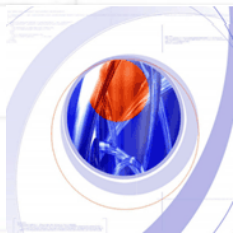
- 程序代码长度增加了
 - 提高并行性而进行的大量的循环展开；
 - 指令字中的操作槽并非总能填满。
解决：采用指令共享立即数字段的方法，或者采用指令压缩存储、调入Cache或译码时展开的方法。
- 采用了锁步机制

任何一个操作部件出现停顿时，整个处理机都要停顿。
- 机器代码的不兼容性

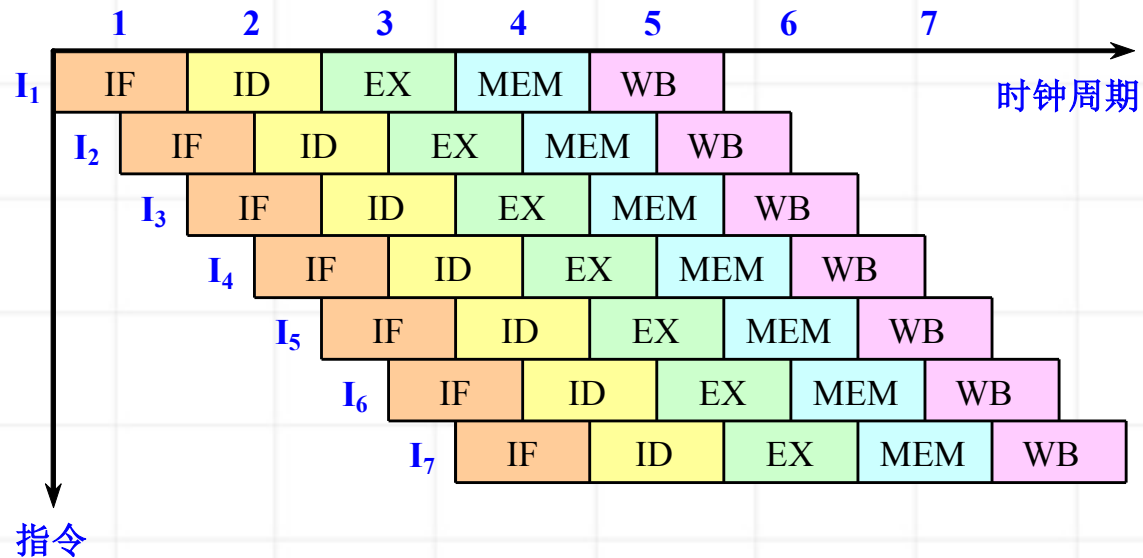


9. 超流水线处理机

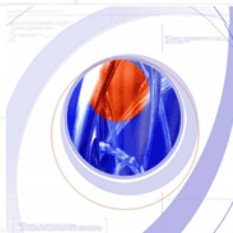
- 将每个流水段进一步细分，这样在一个时钟周期内能够分时流出多条指令。这种处理机称为超流水线处理机。
- 对于一台每个时钟周期能流出 n 条指令的超流水线计算机来说，这 n 条指令不是同时流出的，而是每隔 $1/n$ 个时钟周期流出一条指令。
 - 实际上该超流水线计算机的流水线周期为 $1/n$ 个时钟周期。



10. 超流水线处理机时空图



一台每个时钟周期分时流出两条指令的超流水线计算机的时空图





谢谢大家

