

计算机系统结构

第五讲相关

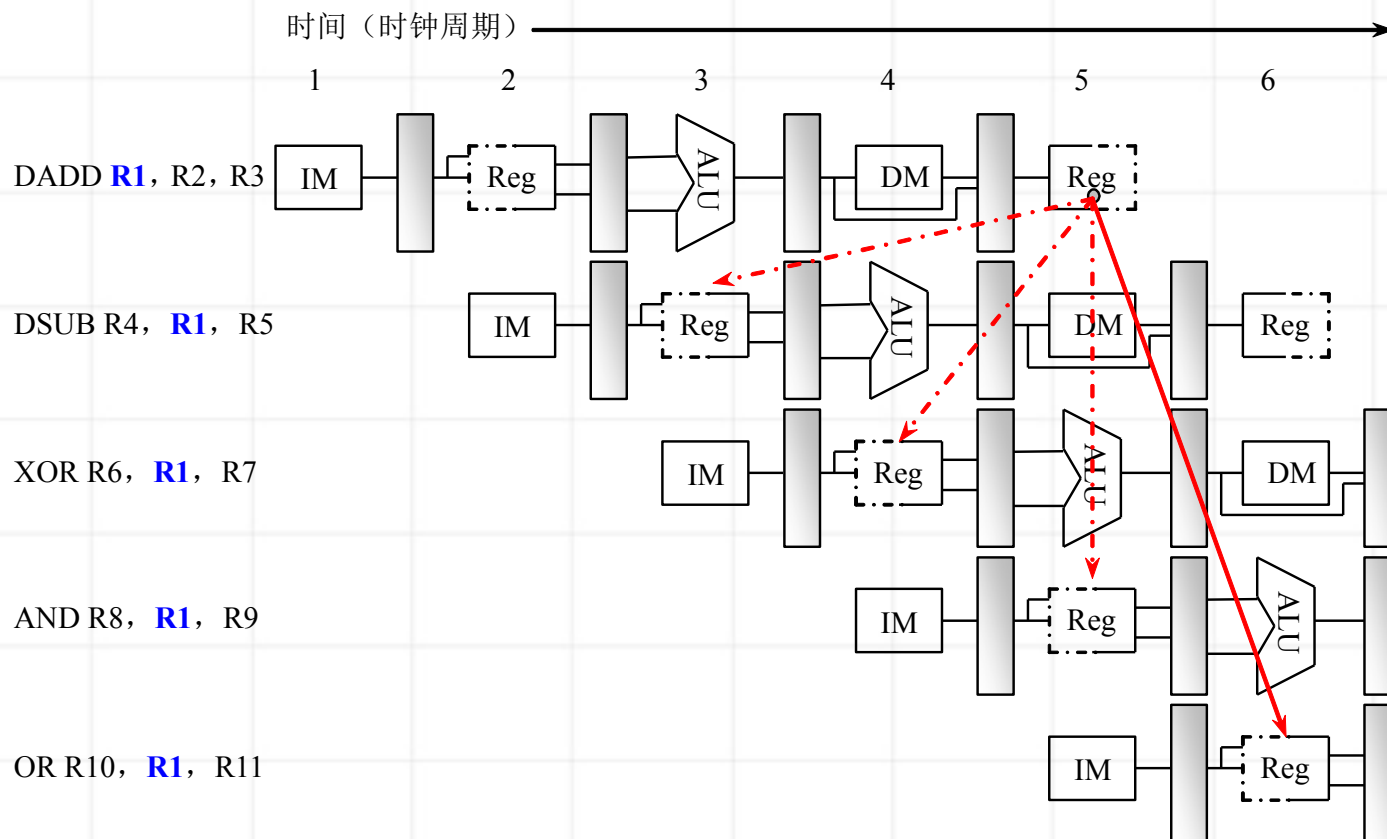
冯丹

武汉光电国家研究中心

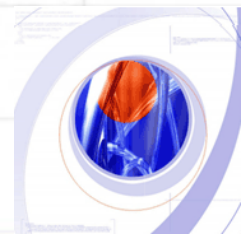


Computer Architecture

1. 流水线冲突实例



后续指令需要前面指令结果而导致的流水线冲突



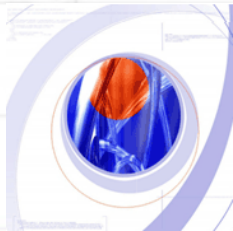
2. 相关

— 相关

- 两条指令之间存在某种依赖关系。
- 如果两条指令相关，则它们就有可能不能在流水线中重叠执行或者只能部分重叠执行

— 三类相关

- 数据相关（也称真数据相关）
- 名相关
- 控制相关



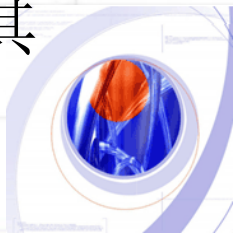
3-1. 数据相关

- 数据相关

- 对于两条指令*i*（在前，下同）和*j*（在后，下同），如果下述条件之一成立，则称指令*j*与指令*i*数据相关。
 - 指令*j*使用指令*i*产生的结果；
 - 指令*j*与指令*k*数据相关，而指令*k*又与指令*i*数据相关。

- 数据相关特点

- 数据相关具有传递性。
- 数据相关反映了数据的流动关系，即如何从其产生者流动到其消费者。



3-2. 数据相关实例

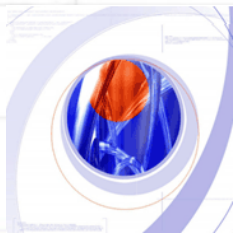
```
Loop:  L. D      F0, 0 (R1)    // F0为数组元素

        ADD. D   F4, F0, F2    // 加上F2中的值

        S. D     F4, 0 (R1)    // 保存结果

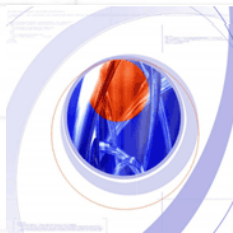
        DADDIU   R1, R1, -8    // 数组指针递减8个字节

        BNE      R1, R2, Loop  // 如果R1≠R2, 则分支
```



3-3. 数据相关检测

- 当数据的流动是经过寄存器时，相关的检测比较直观和容易。
- 当数据的流动是经过存储器时，检测比较复杂。
 - 相同形式的地址其有效地址未必相同；
 - 举例
 - 形式不同的地址其有效地址却可能相同
 - 举例



4-1. 名相关

— 名

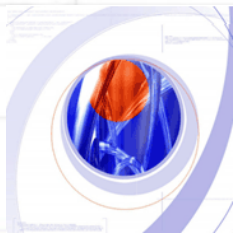
- 指令所访问的寄存器或存储器单元的名称
- 例如，R4，0x10000000等

— 名相关

- 如果两条指令使用相同的名，但是它们之间并没有数据流动，则称这两条指令存在名相关

— 细分

- 反相关
- 输出相关



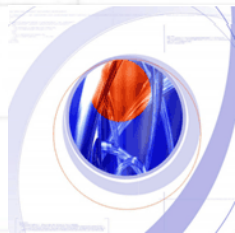
4-2. 名相关之反相关

— 反相关

- 如果指令j写的名与指令i读的名相同，则称指令i和j发生了反相关。
- 指令j写的名=指令i读的名

— 实例

- XOR R1, R2, R3
- SUB R2, R4, R5

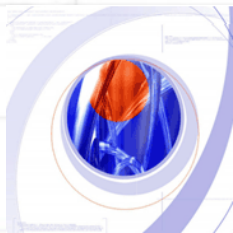


4-3. 名相关之输出相关

— 输出相关

- 如果指令j和指令i写相同的名，则称指令i和j发生了输出相关
- 指令j写的名=指令i写的名

— 实例



4-4. 名相关解决方案

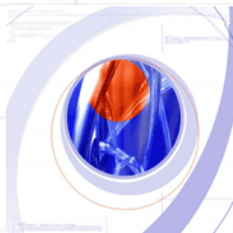
- 名相关特点

- 名相关的两条指令之间并没有数据的传送。
- 如果一条指令中的名改变了，并不影响另外一条指令的执行。

- 消除名相关的方法：换名技术

- 换名技术

- 通过改变指令中操作数的名来消除名相关。
- 对于寄存器操作数进行换名称为寄存器换名
 - 既可以用编译器静态实现，也可以用硬件动态完成。



4-5. 换名技术实例

例如：考虑下述代码：

DIV. D F2, F8, F4

ADD. D F8, F0, F12

SUB. D F10, F8, F14

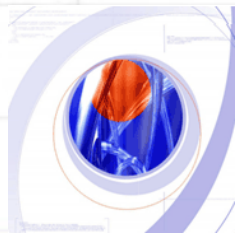
DIV. D和ADD. D存在反相关。

进行寄存器换名（F8换成S）后，变成：

DIV. D F2, F8, F4

ADD. D S, F0, F12

SUB. D F10, S, F14



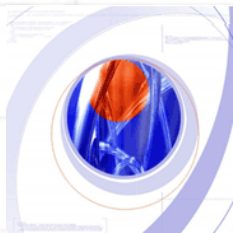
5. 控制相关

— 控制相关

- 控制相关是指由分支指令引起的相关。
- 为了保证程序应有的执行顺序，必须严格按控制相关确定的顺序执行。

— 控制相关实例

```
if p1 then {S1};  
S;  
if p2 then {S2; };
```





谢谢大家

