指令集并行就是流水化

流水化的主要阻碍：

结构冒险：各种不同的指令同时执行，导致资源的竞争而产生的停顿。（实现硬件成本高，结构冒险如果很罕见，就不用实现的；把指令缓存和数据缓存分开可以减少结构冒险）

数据冒险：旁路（转发，短路）不能旁路的用检测互锁流水线来停顿。

控制冒险：静态分支预测判断（靠程序员统计自己的代码劲量不跳转），动态预测判断（用缓冲区记录分支预测是否正确，如果错误就反转记录位）。

指令级并行

1 概念与挑战

2 基本编译技术

循环展开：减少停顿，减少分支。汇编指令，（减少相关，避免停顿）

3 用高级分支预测降低分支成本

两位预测器错误率 > 相关分支预测错误率 > 竞赛预测器错误率

4 动态调度克服数据冒险

乱序执行。

乱序执行的算法（tomasulo算法）跟踪指令的操作数何时可用（降低RAW）寄存器重命名降低WAR和WAW冒险

5 动态调度：示例和算法

充分理解tomasulo算法

6 基于硬件的推测

克服控制冒险而产生的流水线技术。原理跟上一节类似，都是靠记录数据来完成。

7 以多发射和静态调度来开发ILP

在发射的时候都是动态的，因为这要考虑数据冒险的问题，一般应用于嵌入式设备。

8以多发射和动态调度和推测来开发ILP

I7是这个功能的代表，其中包括推测（克服控制冒险）动态调度（克服数据冒险）产生的流水线。并包含二位分支预测。

9 用于指令传送和推测的高级技术

用于增加指令发射带宽，推测增强《基于硬件的推测》课的内容。

10 ILP局限性研究

人们现在放弃了ILP并行开发的投入，因为回报太低了。

11 交叉问题，ILP与存储器

？

12 多线程。

这个的多线程，不是操作系统提供的多线程，而是常说的双核四线程中的线程。