

# 多发射处理器

动态多发射处理器

# 什么是动态多发射？

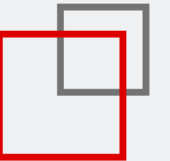
◆ 动态多发射： 超标量处理器

	超标量 (Superscalar)	超长指令字 (VLIW)
指令长度	固定大小	固定大小 (但更长)
指令格式	固定	固定
寄存器	通用寄存器、换名、 多个读写端口	很多个通用寄存器、 多个读写端口
访存方式	load/store	load/store
关键特性	动态发射结构、 硬件实现冲突检测	静态发射结构、编译器实现 冲突检测、指令调度

◆ 使用硬件、执行时动态决定同时发射的指令



# 超标量结构 (superscalar)

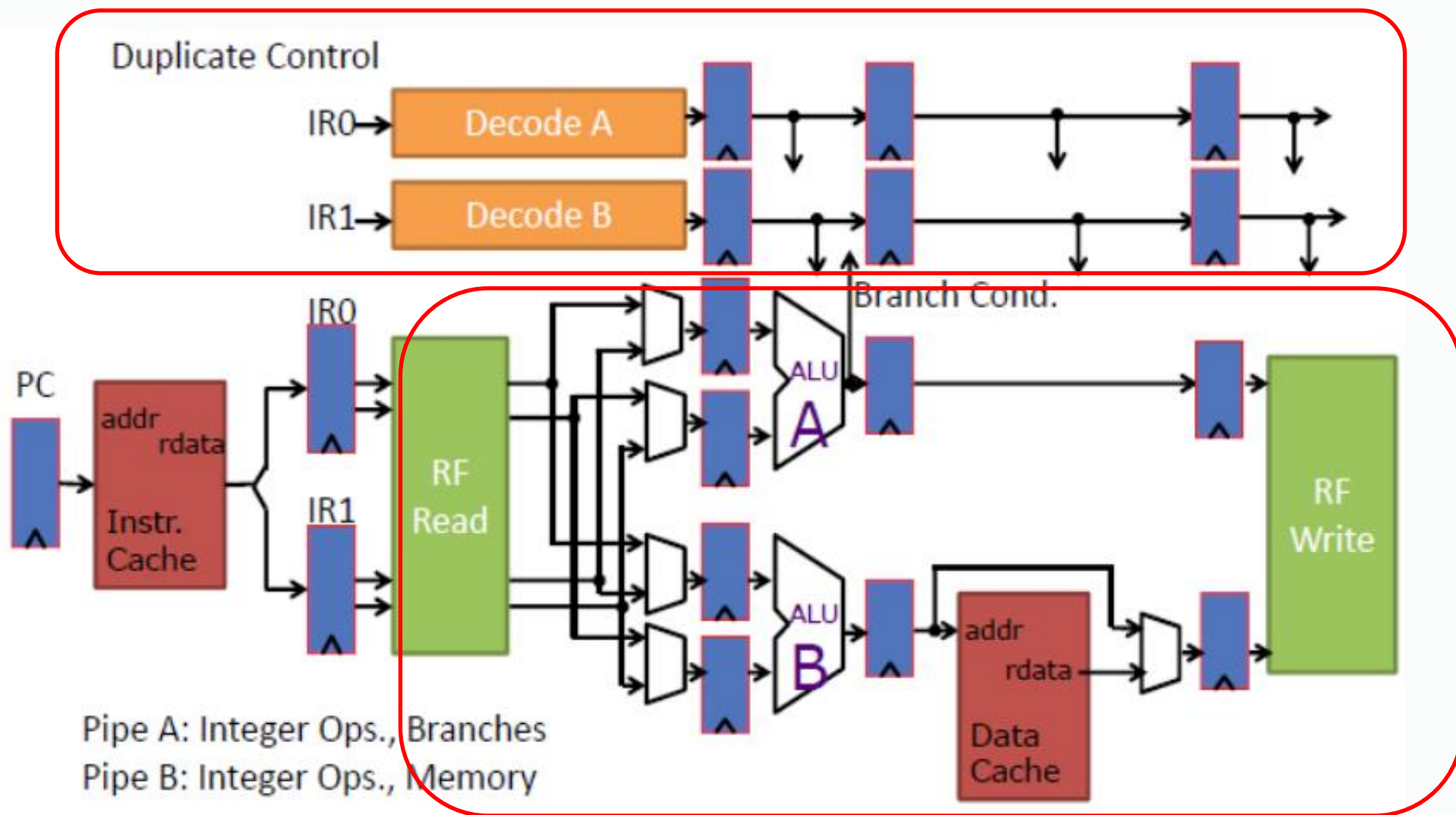




# 多发射处理器的分类

常用名	发射结构	冲突检测	调度方式	特点	处理器举例
静态超标量 superscalar (static)	动态	硬件	静态	按序执行	大部分嵌入式处理器, 例如ARM cortex-A8
动态超标量 superscalar (dynamic)	动态	硬件	动态	乱序执行	目前无
推测执行超标量 superscalar (speculative)	动态	硬件	带推测的动态	乱序、 推测执行	大部分通用处理器, 如Intel Core i3,i5,i7
超长指令字 (VLIW)	静态	主要由软件完成	静态	编译器 (隐式) 完成冲突检测、指令调度	某些特定领域,如信号处理器 TI C6x
显式并发指令运算(EPIC)	主要为静态	主要由软件完成	主要为静态	编译器 (显式) 完成冲突检测、指令调度	Intel 安腾 Itanium处理器

# 一个双发射超标量MIPS处理器



# 动态多发射：流水线时空图

OpA	F	D	A0	A1	W		
OpB	F	D	B0	B1	W		
OpC		F	D	A0	A1	W	
OpD		F	D	B0	B1	W	
OpE			F	D	A0	A1	W
OpF			F	D	B0	B1	W

ADDIU	F	D	A0	A1	W			
LW	F	D	B0	B1	W			
LW		F	D	B0	B1	W		
ADDIU		F	D	A0	A1	W		
LW			F	D	B0	B1	W	
LW			F	D	D	B0	B1	W

- 每次取指、译码、执行两条指令
- 理想情况：CPI=0.5

- 指令发射逻辑
  - 能调换指令所进入的功能部件
  - 能检测结构冒险、数据冒险等等



# 动态多发射：流水线时空图

No Bypassing: 无前向通路

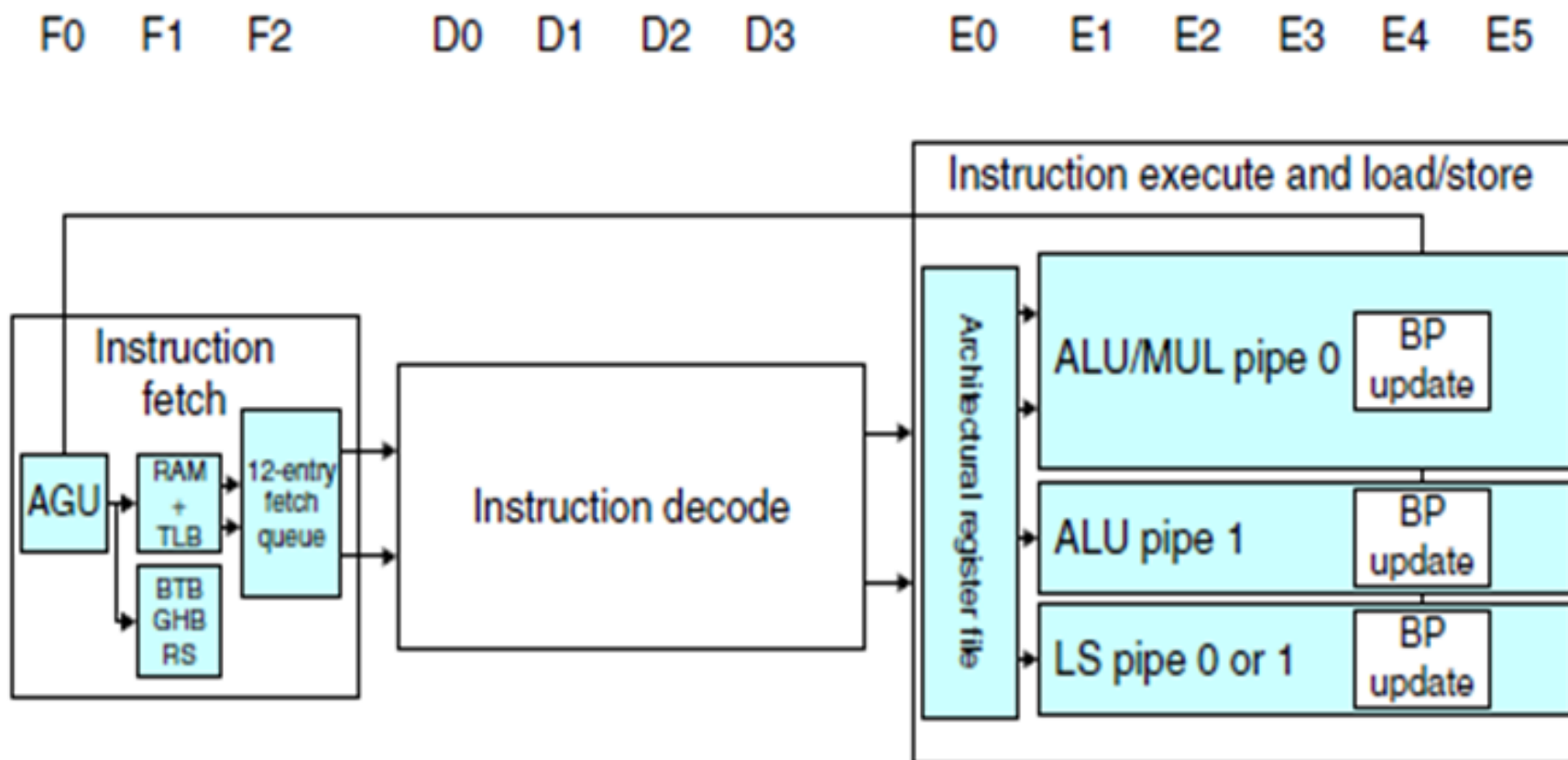
ADDIU R1,R1,1	F	D	A0	A1	W			
ADDIU R3,R4,1	F	D	B0	B1	W			
ADDIU R5,R6,1		F	D	A0	A1	W		
ADDIU R7,R5,1		F	D	D	D	D	A0	A1 W

Full Bypassing: 有前向通路

ADDIU R1,R1,1	F	D	A0	A1	W			
ADDIU R3,R4,1	F	D	B0	B1	W			
ADDIU R5,R6,1		F	D	A0	A1	W		
ADDIU R7,R5,1		F	D	D	A0	A1	W	



# 按序超标量实例：ARM Cortex-A8







# A8流水线处理器的特点

- 静态调度、动态发射、按序超标量
- 静态调度，编译器尽力做：
  - 避免两条相邻指令使用同一部件（结构冒险）
  - 避免相邻的指令有数据依赖关系（数据相关）
- 动态发射结构：
  - 每个周期由控制逻辑判断是发射一条还两条指令，还是不发射指令；
  - 结构冒险：编译器实在无法避免时，能检测出冒险，一次只发送一条；
  - 数据冒险：编译器实在无法避免时，能检测出冒险，如果检测到冒险，要么把两条都停顿、要么停顿两条中的一条；
  - 控制冒险：当转移预测错误时、清空流水线，从正确位置重新开始执行。



# 小结

## 动态多发射

超标量：硬件决定哪些指令可以同时执行

## 实例

双发射按序超标量MIPS处理器

## 实际产品

ARM Cortex-A8

## 下一节

乱序超标量技术



# 谢谢观看

上海交通大学