

## 本节主题

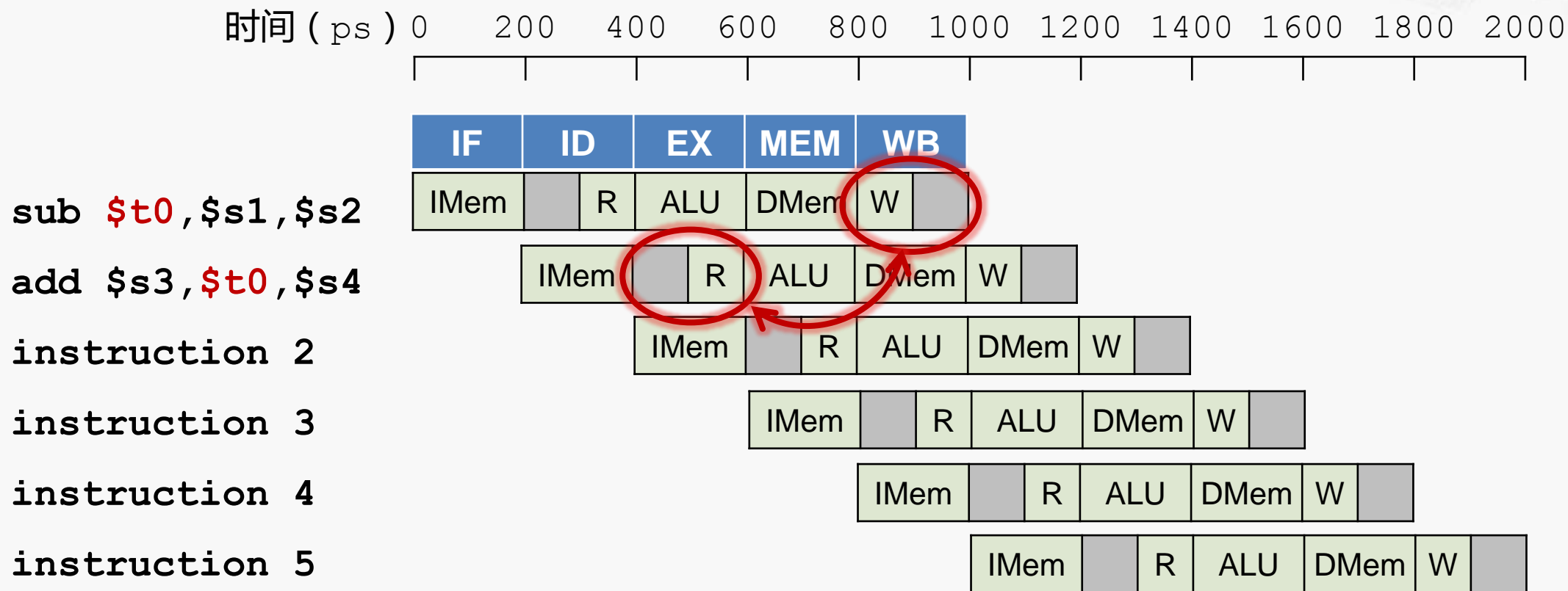


# 数据冒险的处理

北京大学·慕课  
计算机组成  
制作人：陆俊林

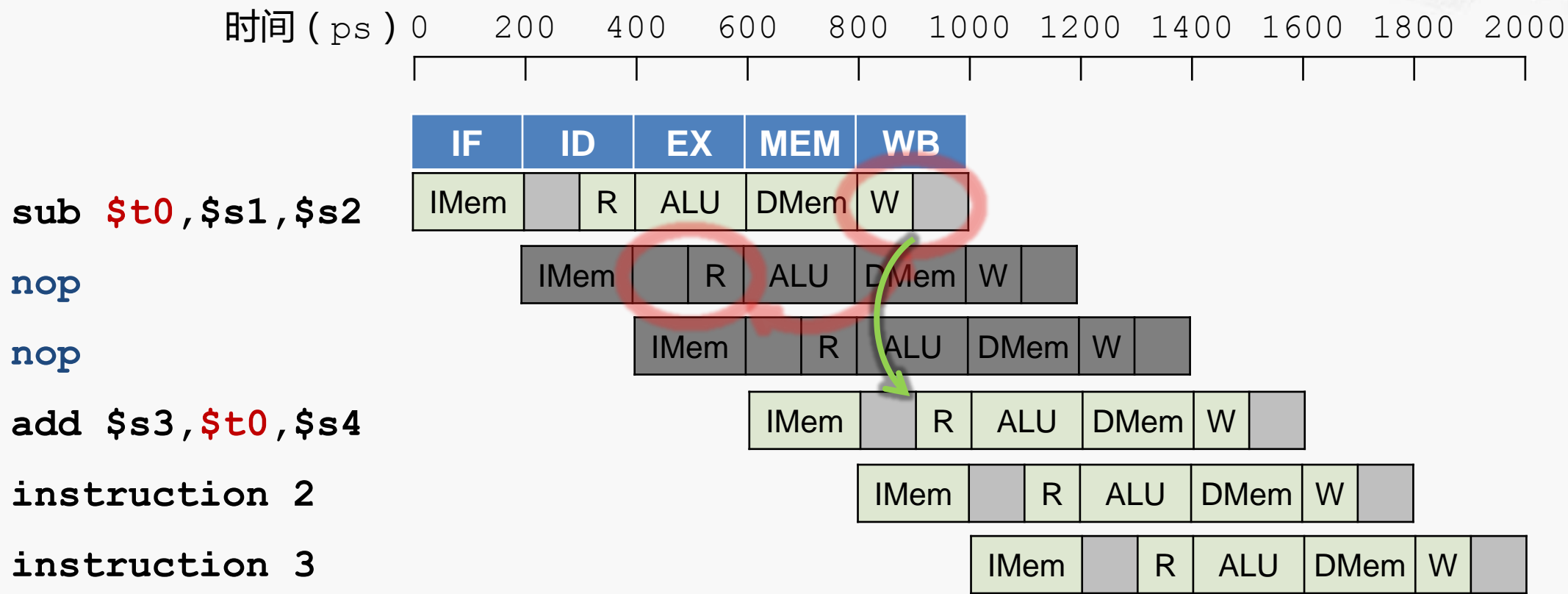


# 数据冒险 ( 示例1 )



一条指令需要使用之前指令的运算结果，但是结果还没有写回

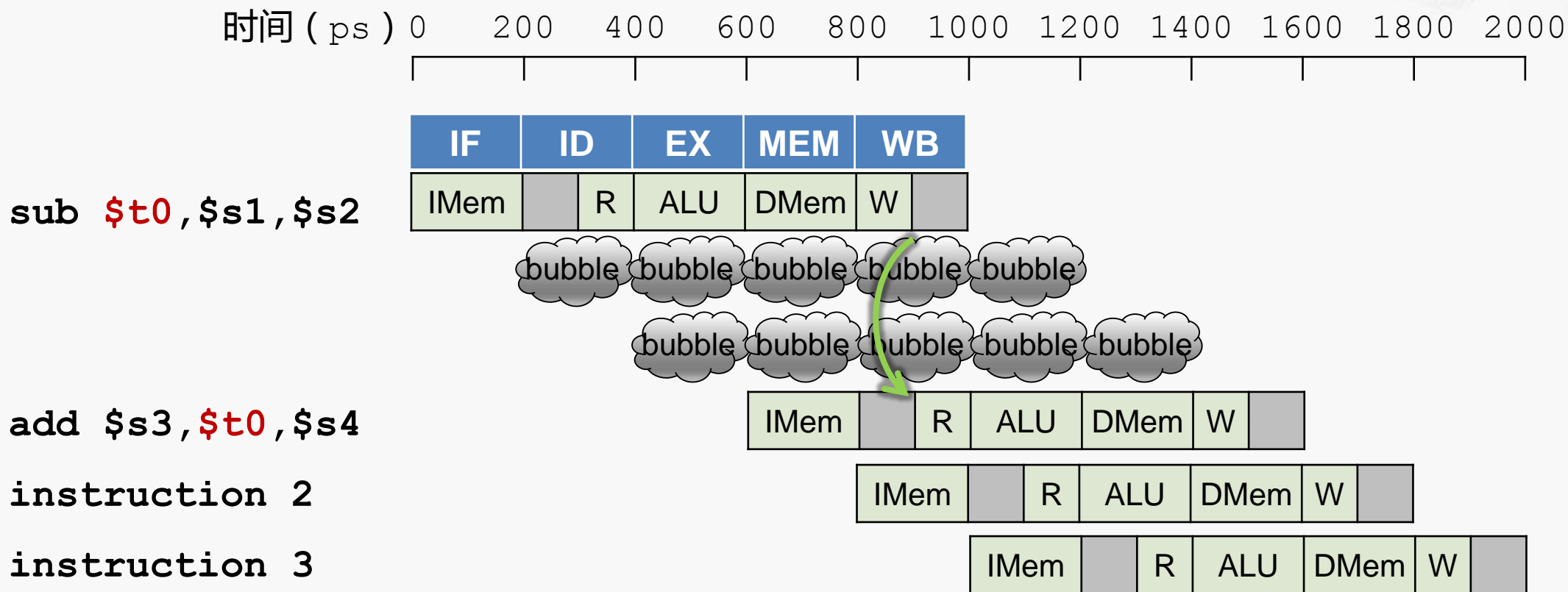
# 数据冒险 ( 示例1 )



一条指令需要使用之前指令的运算结果，但是结果还没有写回

软件解决方案：插入nop指令

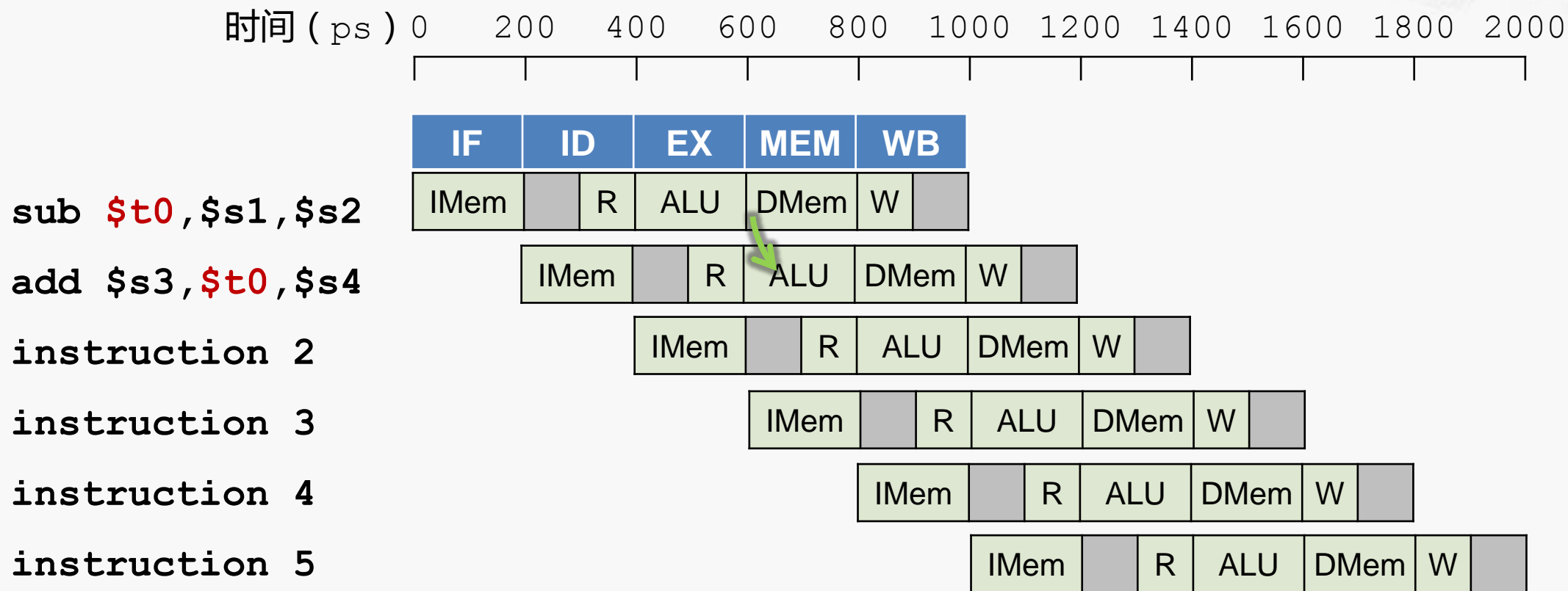
# 数据冒险（示例1）



一条指令需要使用之前指令的运算结果，但是结果还没有写回

解决方案1：流水线停顿（stall），产生空泡（bubble）

# 数据冒险 ( 示例1 )

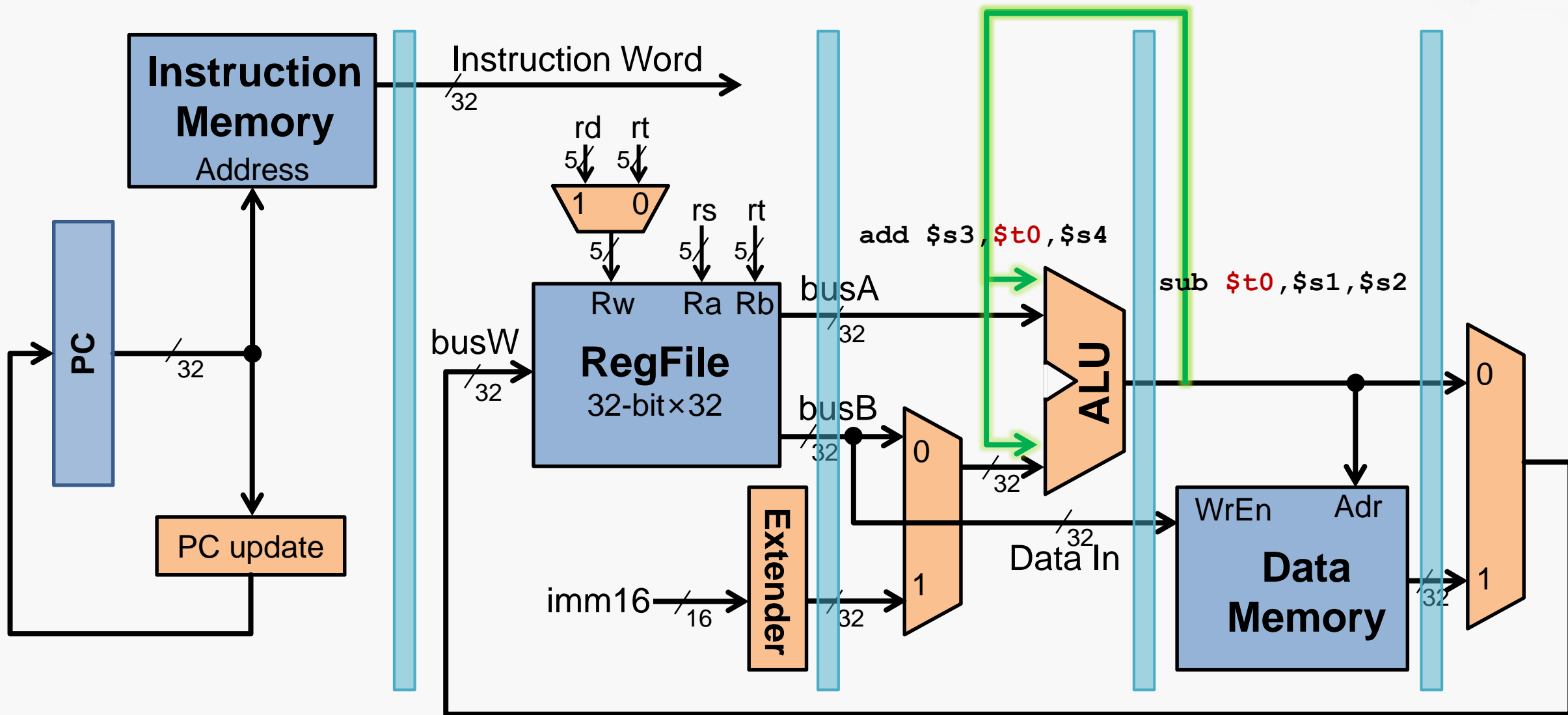


一条指令需要使用之前指令的运算结果，但是结果还没有写回

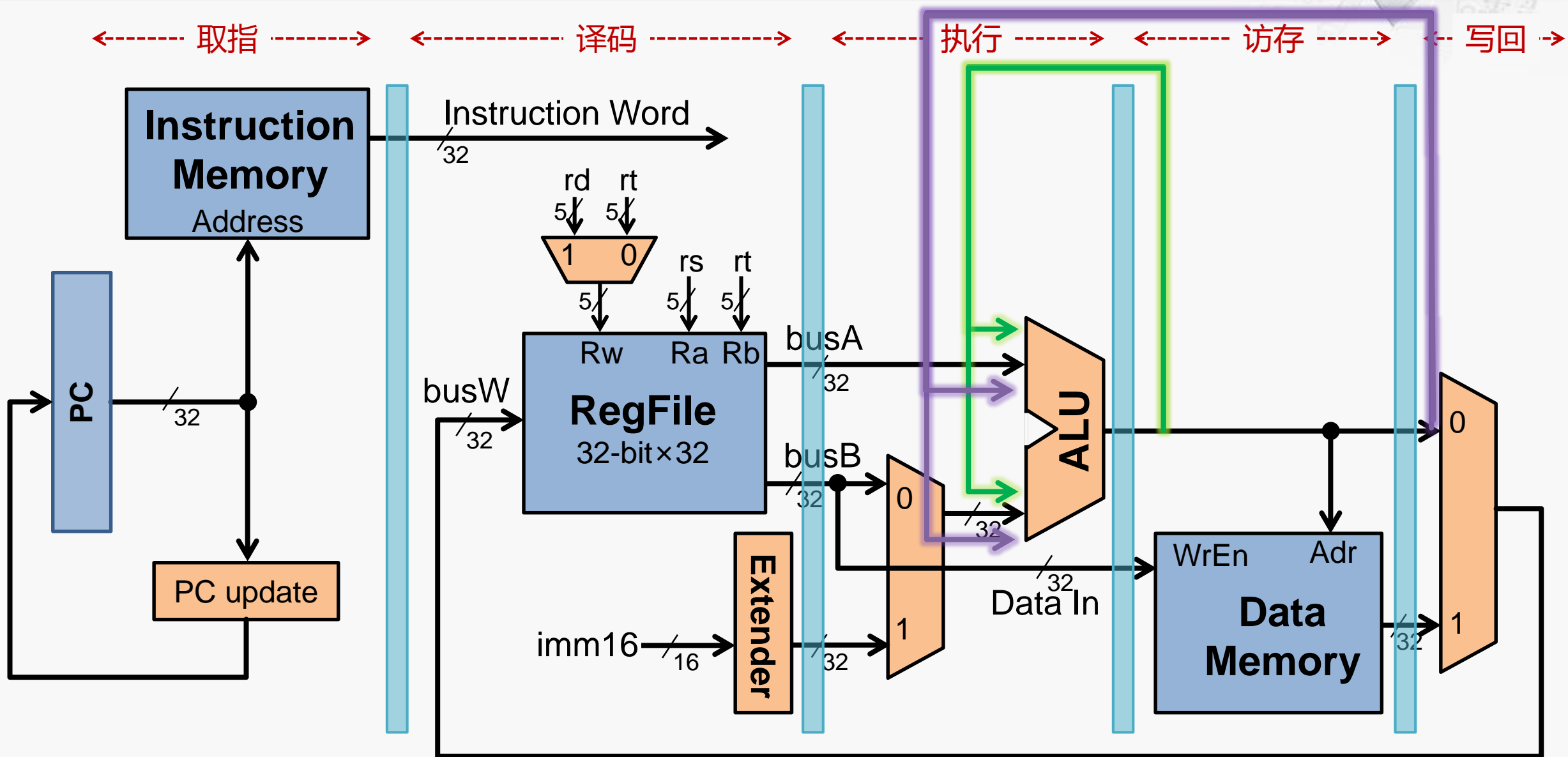
解决方案2：数据前递 ( Forwarding )

# 前递 ( Forwarding )      旁路 ( Bypass )

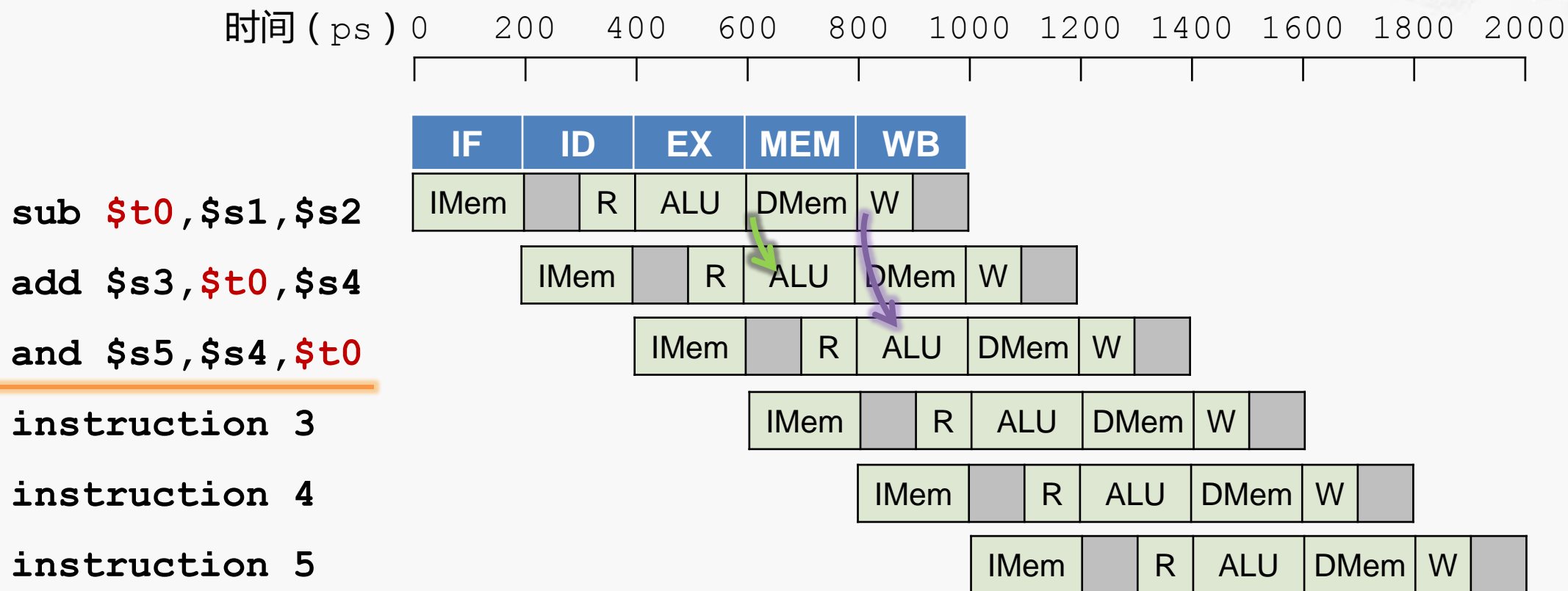
←----- 取指 ----->    ←----- 译码 ----->    ←----- 执行 ----->    ←----- 访存 ----->    ←----- 写回 ----->



# 前递 ( Forwarding )      旁路 ( Bypass )



# 数据冒险 ( 示例2 )

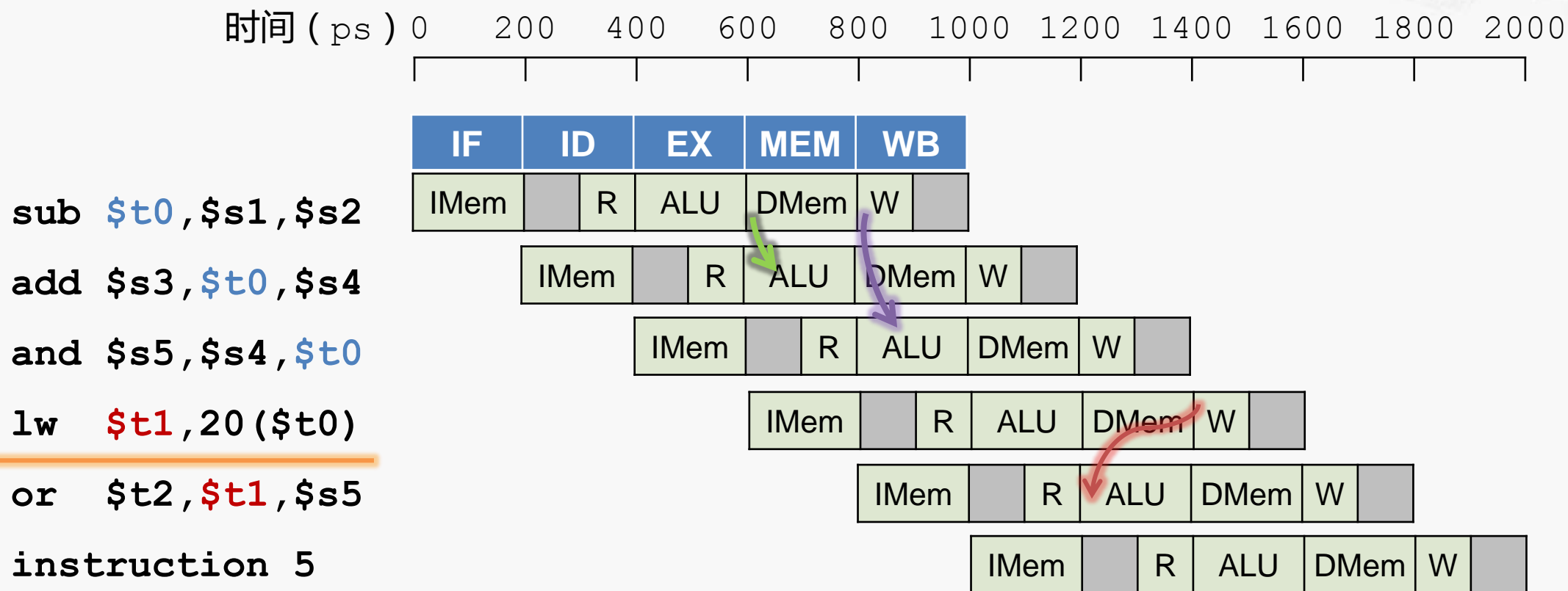


一条指令需要使用之前指令的运算结果，但是结果还没有写回

解决方案2：数据前递 ( Forwarding )



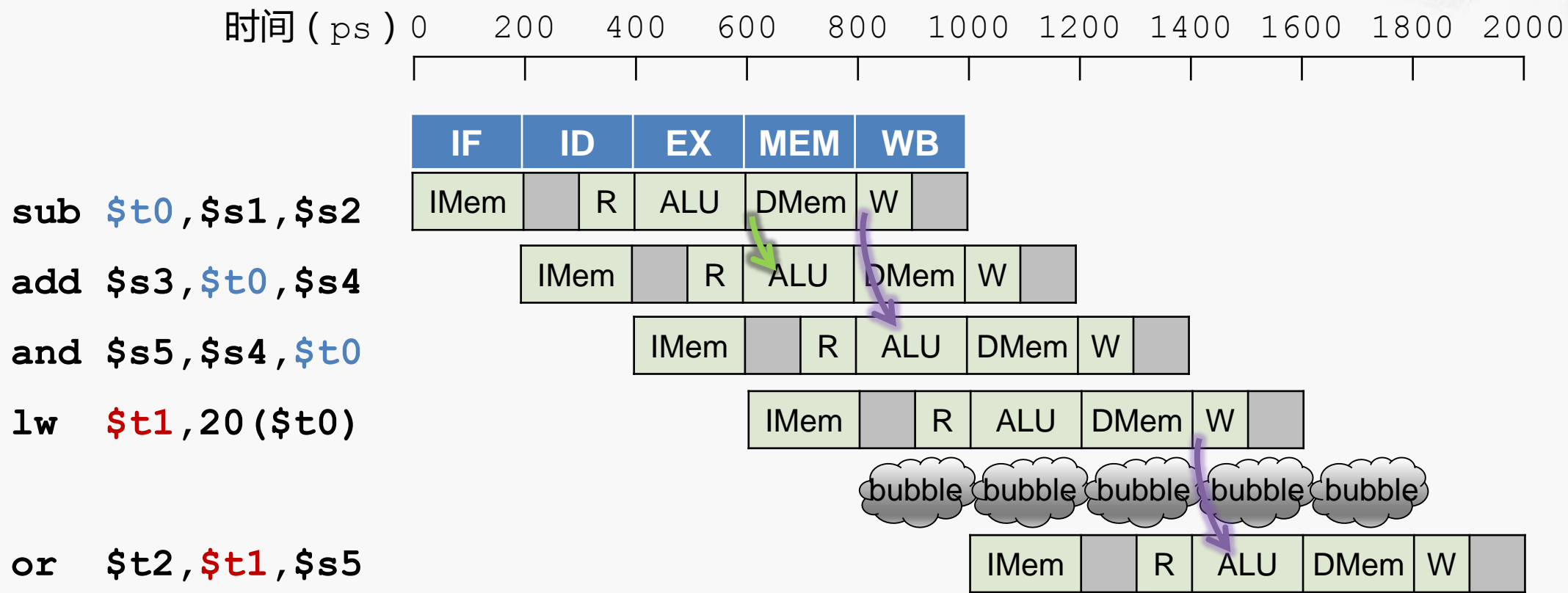
# 数据冒险 ( 示例3 )



一条指令需要使用之前指令的访存结果 ( Load-Use Harzard )

数据前递 ( Forwarding ) 也无法解决

# 数据冒险 ( 示例3 )



一条指令需要使用之前指令的访存结果 ( Load-Use Harzard )

解决方案：流水线停顿 + 数据前递

## 本节小结



# 数据冒险的处理

北京大学·慕课  
计算机组成  
制作人：陆俊林

