山东大学 计算机科学与技术 学院

计算机体系结构 课程实验报告

学号：201900130133	姓名：**施政良**	班级： **四班**
实验题目：结构相关		
实验学时：2	实验日期： 2022-05-12	

实验目的：

（1）通过本实验，加深对结构相关的理解，了解结构相关对 CPU 性能的影响。

硬件环境：

WinDLX (一个基于 Windows 的 DLX 模拟器)

软件环境：

Windows 7

实验步骤与内容：
实验内容

本次实验主要涉及流水线的结构相关,具体的实验步骤可以划分为如下几个步骤

（1） 用 WinDLX 模拟器运行程序 structure_d.s 。

（2） 通过模拟，找出存在结构相关的指令对以及导致结构相关的部件。

（3） 记录由结构相关引起的暂停时钟周期数,计算暂停时钟周期数占总执 行周期数的百分比。

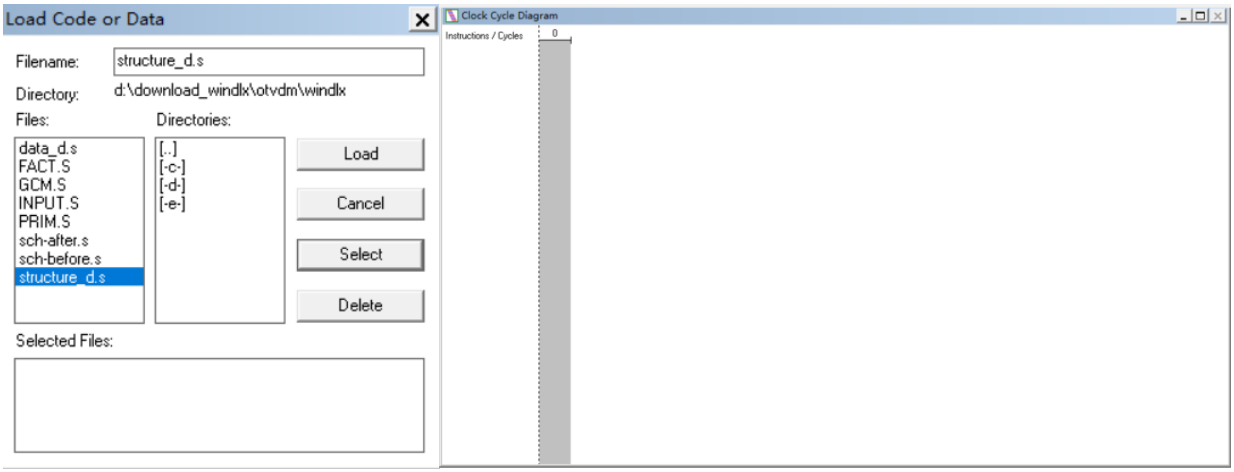
（4） 论述结构相关对 CPU 性能的影响，讨论解决结构相关的方法。

具体实验过程如下所示。

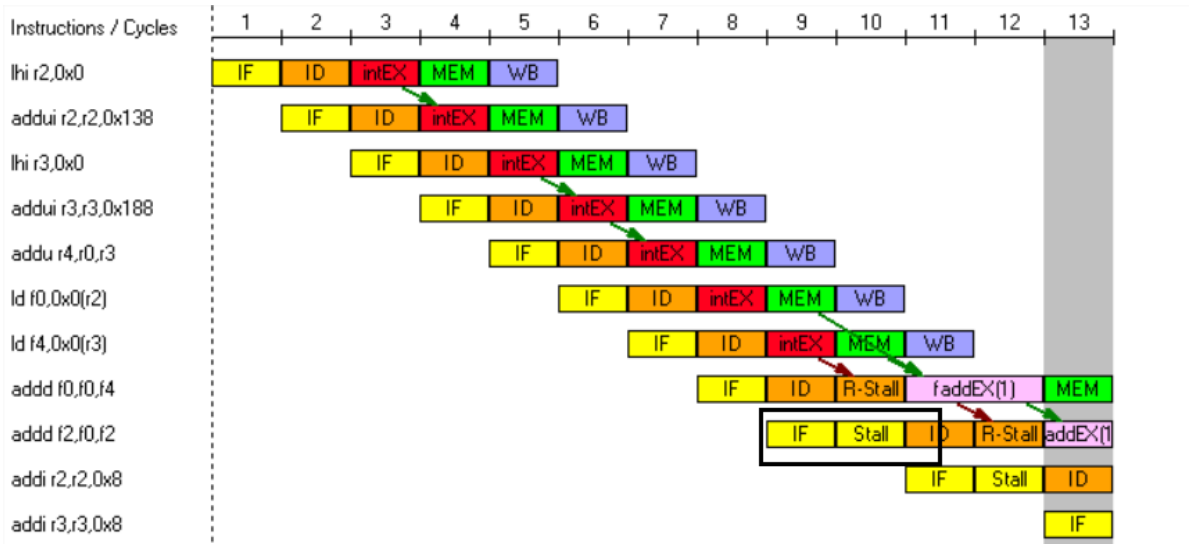
具体实验过程

（1） 实验前的准备：首先加载汇编文件 structure_d.s，采用单步跟踪观察指令执

行时流水线的变化。之后在 Clock Cycle Diagram 内观察执行部件的执行情况。

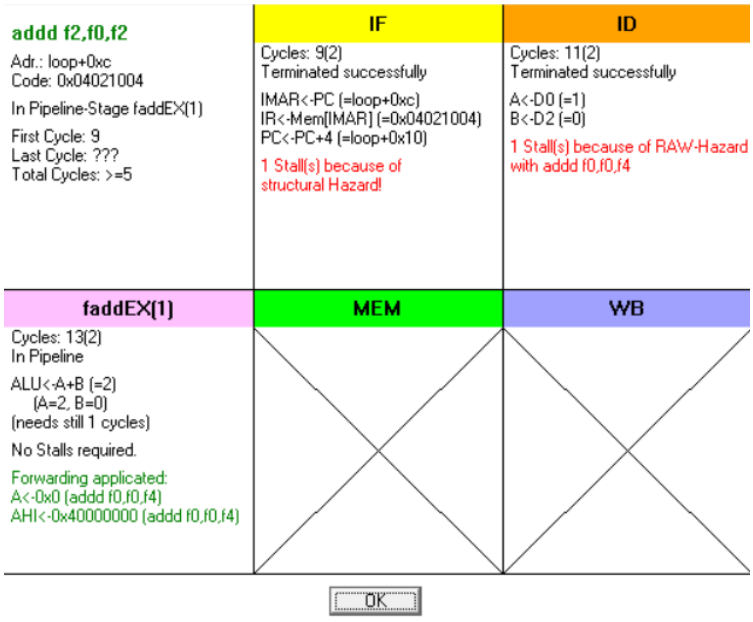


（2） 按下 F7 单步跟踪，知道遇到第一个结构相关，此时在 Clock Cycle Diagram 窗口中出现 stall 字段。

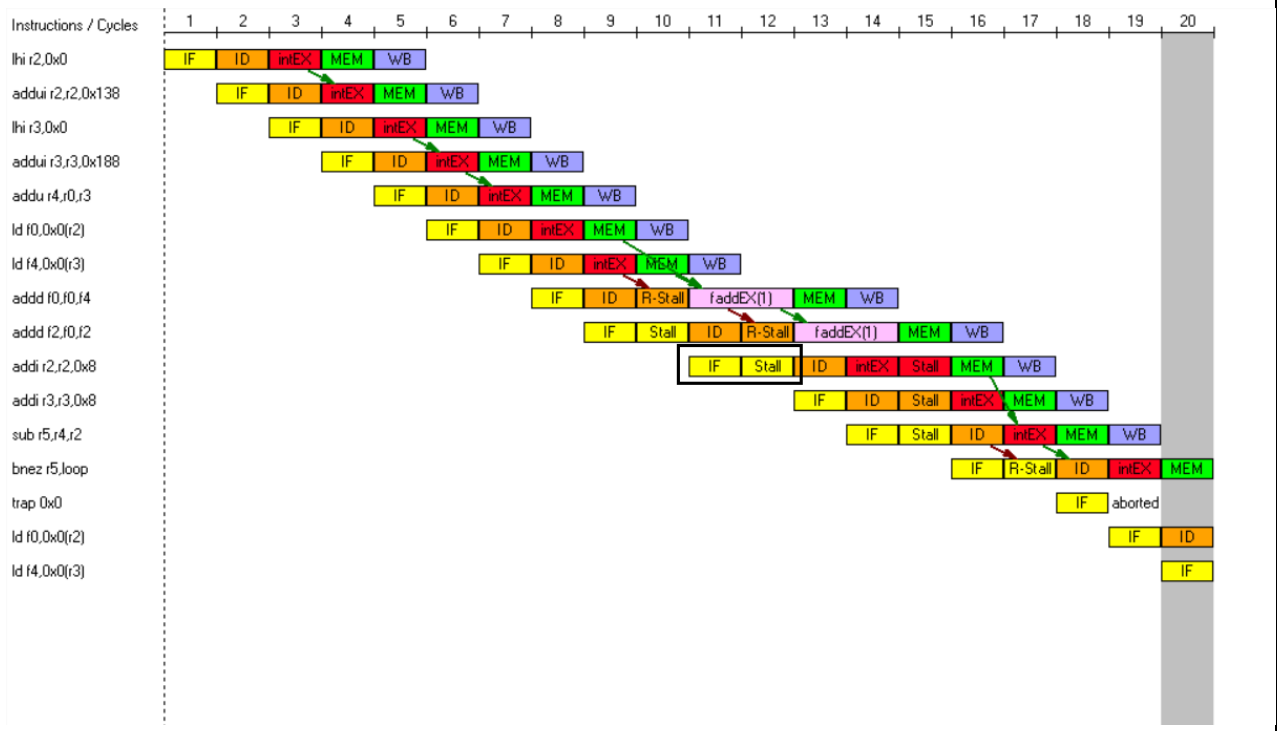


此时对应的指令为$add\ f2\ f0,f2$. 此处结构相关是由于之前的数据相关所导致的。由于在上一条指令中$f0$的计算依赖于$f4$, 而$f4$在计算时并未产生，因此会停留在译码阶段，而不能正常进入执行阶段。对于执行$add\ f2\ f0\ f2$来说，由于译码部件被上一条指令占用，导致本条指令产生译码器的结构相关，本阻塞在取值阶段。

双击该行，从弹出的窗口中可以看见如下提示：

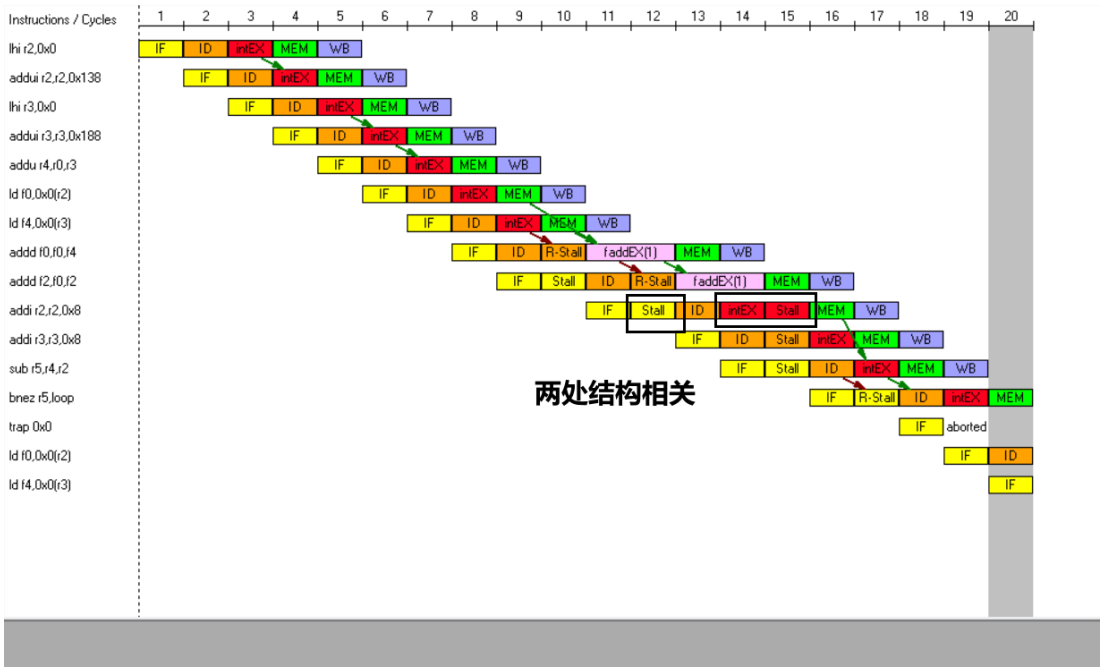


（3） 继续单步跟踪，直到指令运行至$addi\ r2\ r2\ 0x8$, 此时产生第二个结构相关。

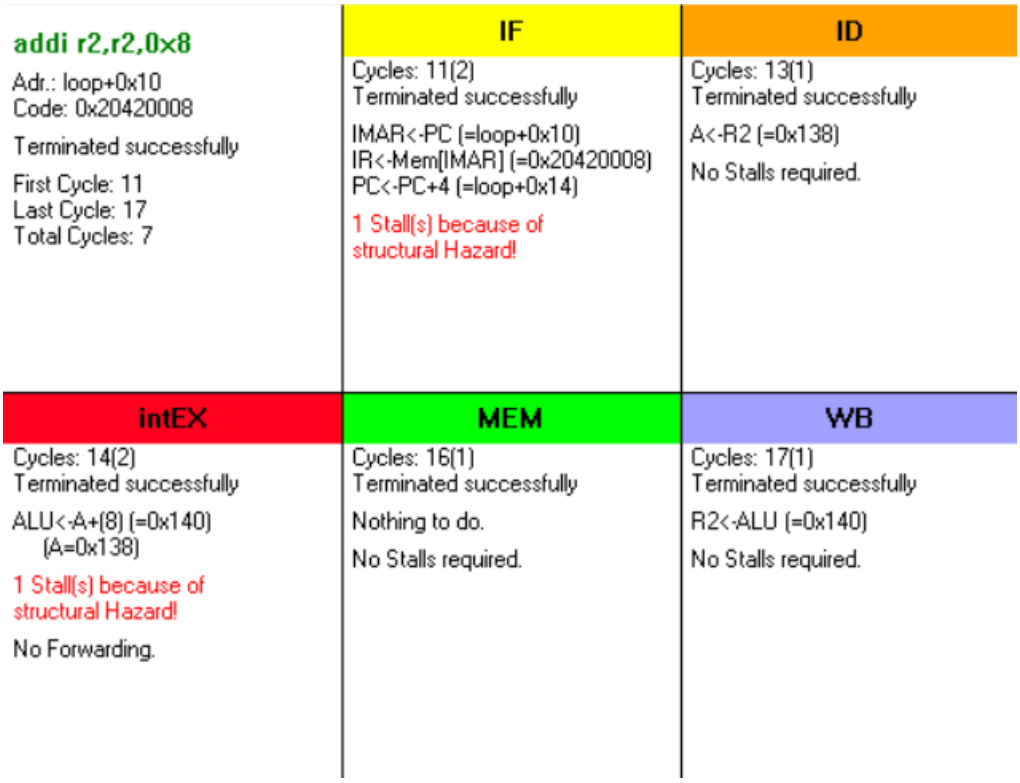


该指令也存在译码器结构相关，其产生原因与第一个结构相关类似，都是由于紧邻的上一条指令无法正常进入执行周期，从而延长占用译码器，导致本条指令无法译码，从而产生结构相关。

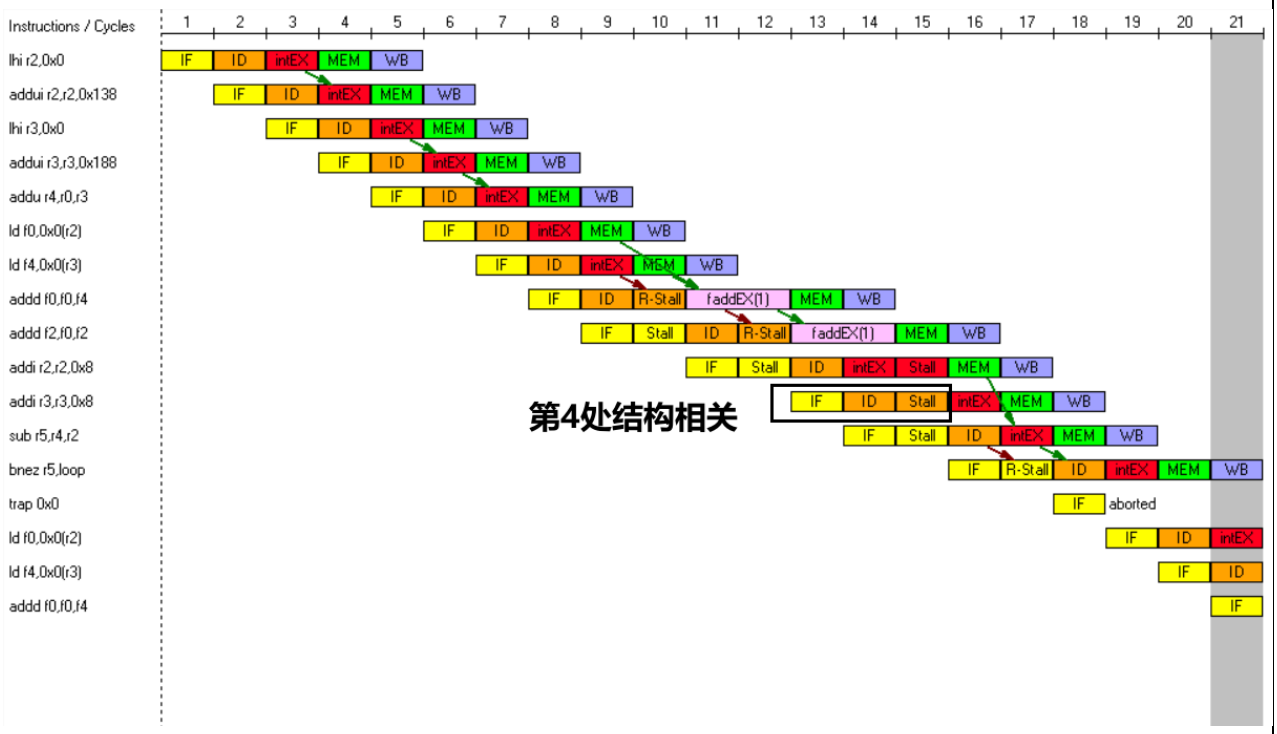
同时，该条指令也存在执行部件的结构相关，如下图所示



分析其原因可知，由于指令$addf$为双精度浮点数加法，因此指令执行的周期较长，导致该条指令的访存结构延迟一个周期与下一条指令的访存周期重合。因此当指令$addi\ r2\ r2\ 0x8$执行完毕之后，访存单元被上一条指令占用，因此产生结构相关。同样的，双击指令，可以看到如下提示：

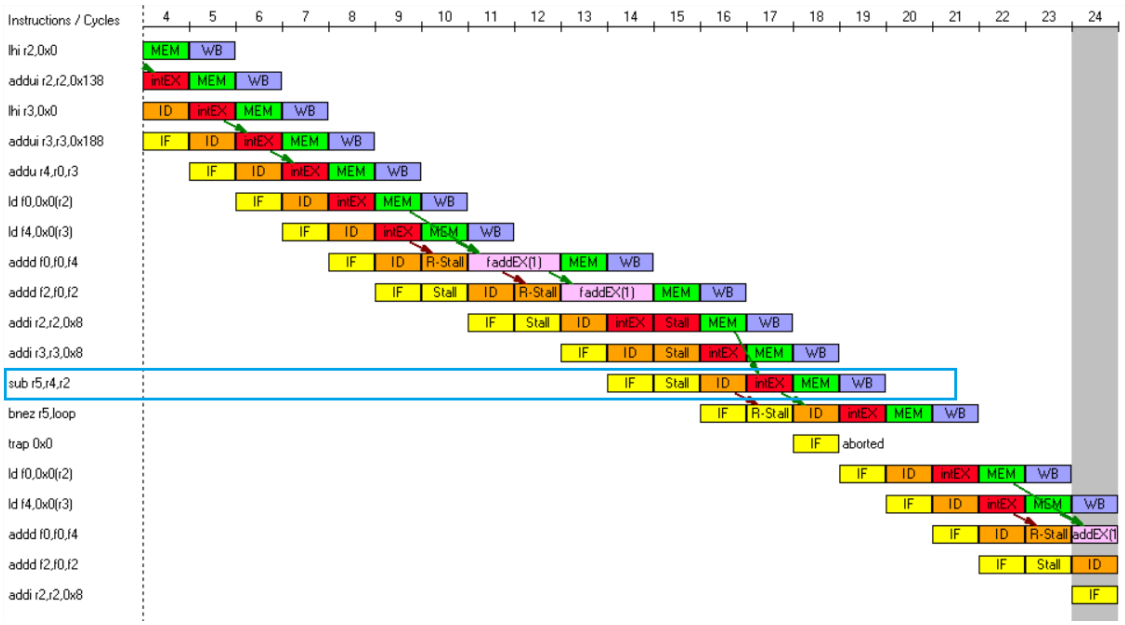


（4） 单步跟踪，直到执行到$addi\ r3\ r3\ 0x8$指令。

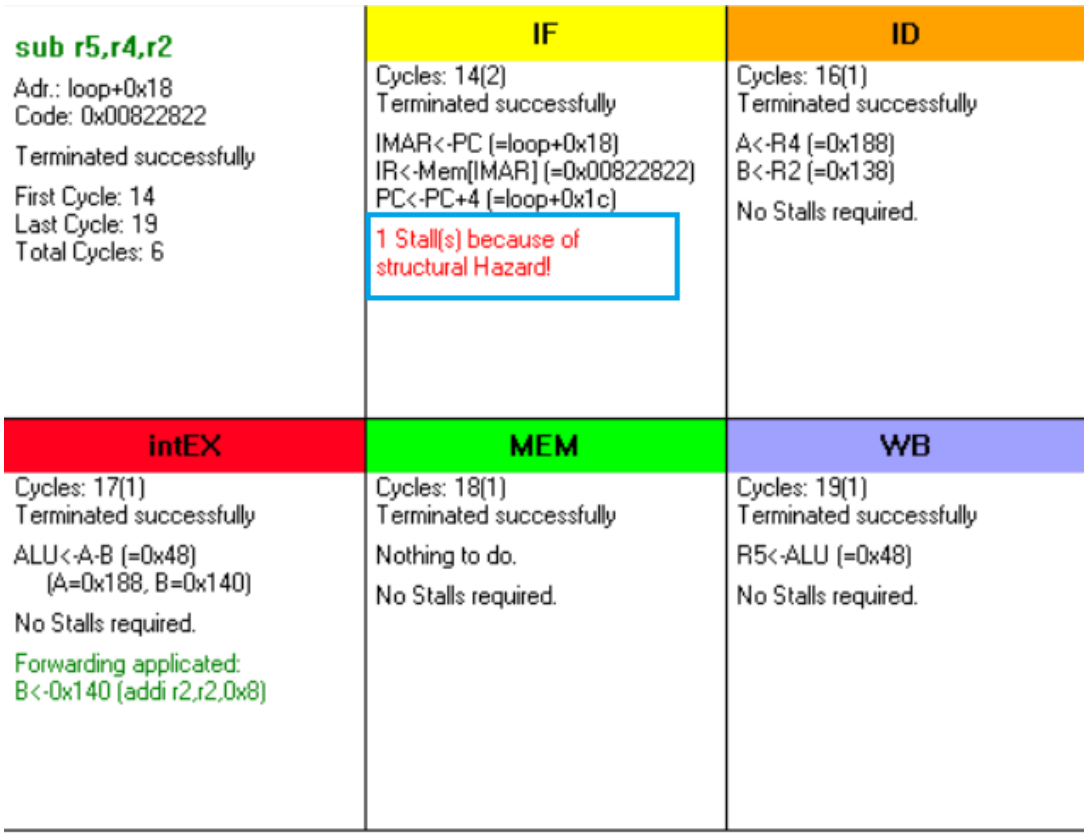


该条指令存在执行部件的结构相关，即由于上一条指令正在占用执行部件，因此本条指令在译码之后无法调用执行部件进行计算，与第三处结构相关的产生原因相同。

（5） 继续单步跟踪，直到执行到$sub\ r5\ r4\ r2$. 此时 Clock Cycle Diagram 窗口的指令流水如下图所示：

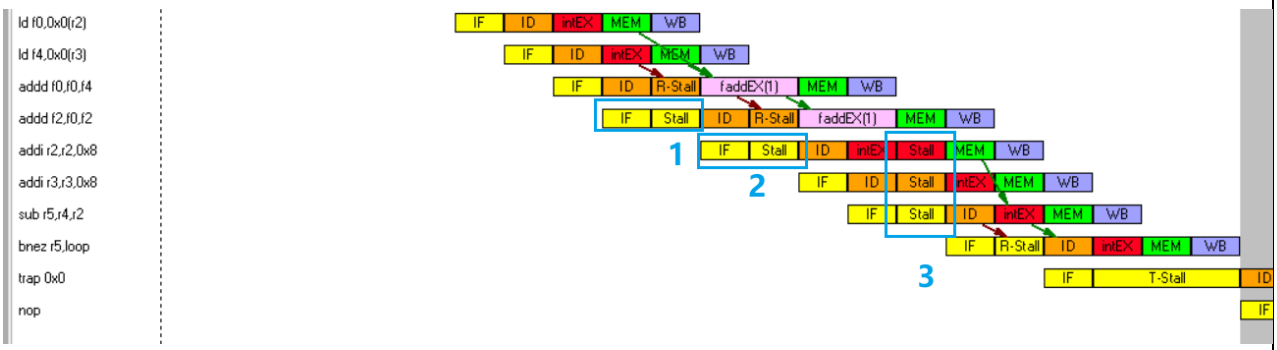


可以看到由于上一条指令被阻塞在译码阶段，因此，当取出指令$sub\ r5\ r4\ r2$之后无法译码，因而产生译码器结构相关。此时，双击指令，有如下的提示：



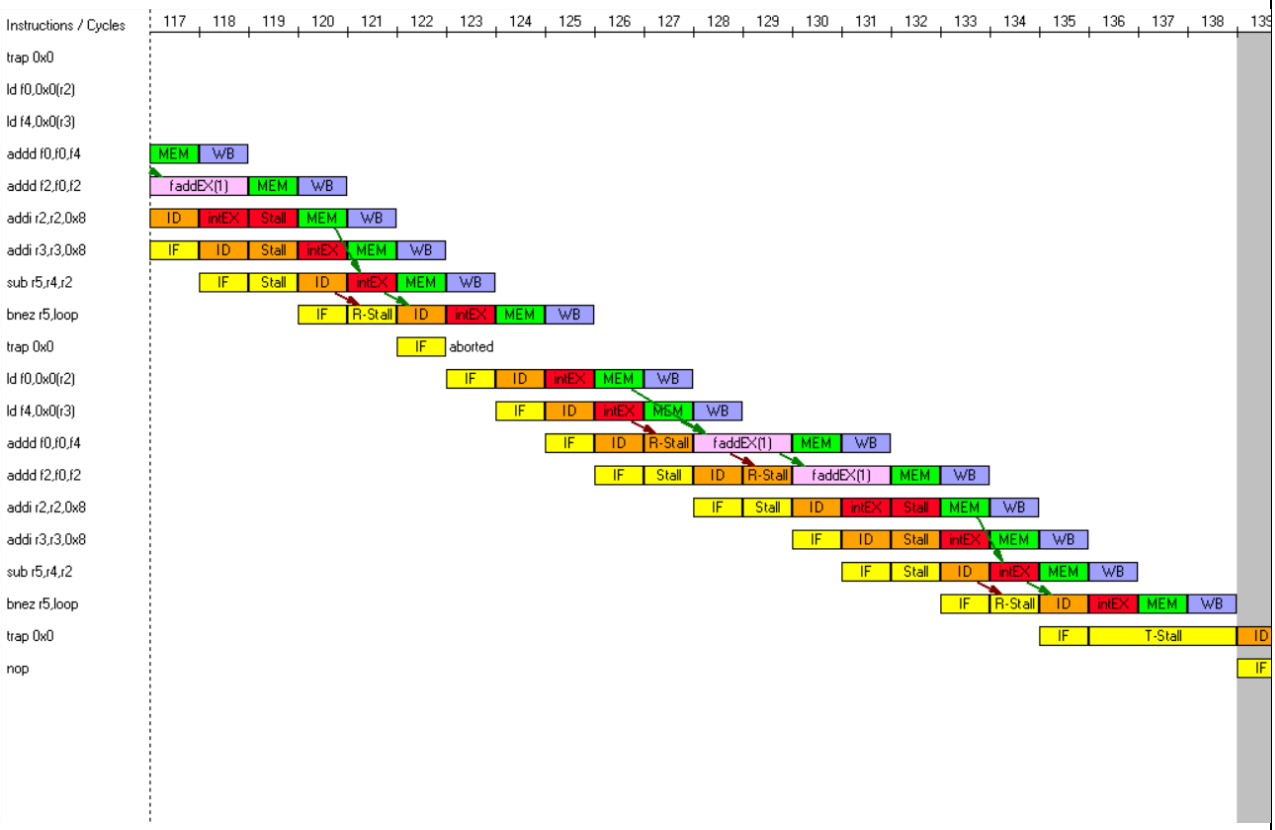
可以看到，指令$sub\ r5\ r4\ r2$在取值之后暂停一个周期。

（6） 重复上述过程：分析汇编代码可知，程序会不断重复上述循环，当循环次数达到 10 次之后即可退出。由于每一次循环都会产生五次结构相关，且后三次结构相关都在同一个周期内发生，如下图所示：



因此在 10 次循环中，因为结构相关而引起的时钟周期的暂停次数为 30 次，占总时钟周期数的$\frac{30}{139} = 21.58\%$.

（7） 按下 F5 连续执行至程序结束。此时各个执行部件的指令流水如下所示：



观察 Statistics 窗口内的统计信息，可以发现程序一共执行了 139 条指令，且数据相关的比例为21.58%，与（5）中的分析相同，说明实验中的分析正确。

```
Total:
    139 Cycle(s) executed.
    ID executed by 86 Instruction(s).
    2 Instruction(s) currently in Pipeline.

Hardware configuration:
    Memory size: 32768 Bytes
    faddEX-Stages: 1, required Cycles: 2
    fmulEX-Stages: 1, required Cycles: 5
    fdivEX-Stages: 1, required Cycles: 19
    Forwarding enabled.

Stalls:
    RAW stalls: 30 (21.58% of all Cycles), thereof:
        LD stalls: 10 (33.33% of RAW stalls)
        Branch/Jump stalls: 10 (33.33% of RAW stalls)
        Floating point stalls: 10 (33.33% of RAW stalls)
    WAW stalls: 0 (0.00% of all Cycles)
    Structural stalls: 0 (0.00% of all Cycles)
    Control stalls: 9 (6.47% of all Cycles)
    Trap stalls: 3 (2.16% of all Cycles)
    Total: 42 Stall(s) (30.22% of all Cycles)

Conditional Branches):
    Total: 10 (11.63% of all Instructions), thereof:
        taken: 9 (90.00% of all cond. Branches)
        not taken: 1 (10.00% of all cond. Branches)

Load-/Store-Instructions:
    Total: 20 (23.26% of all Instructions), thereof:
        Loads: 20 (100.00% of Load-/Store-Instructions)
        Stores: 0 (0.00% of Load-/Store-Instructions)

Floating point stage instructions:
    Total: 20 (23.26% of all Instructions), thereof:
        Additions: 20 (100.00% of Floating point stage inst.)
        Multiplications: 0 (0.00% of Floating point stage inst.)
        Divisions: 0 (0.00% of Floating point stage inst.)

Traps:
    Traps: 1 (1.16% of all Instructions)
```

结论分析与体会：

结论分析

1. **关于流水线相关的分类**

分析：一般来说，流水线中的相关主要分为如下三种类型：

(1) 结构相关：当硬件资源满足不了指令重叠执行的要求，而发生资源冲突时，就发生了结构相关。

(2) 数据相关：当一条指令需要用到前面指令的执行结果，而这些指令均在流水线中重叠执 行时，就可能引起数据相关。

(3) 控制相关：当流水线遇到分支指令和其它能够改变 PC 值的指令时，就会发生控制相关。一旦流水线中出现相关，必然会给指令在流水线中的顺利执行带来许多问题，如果不能很好 地解决相关问题，轻则影响流水线的性能，重则导致错误的执行结果。消除相关的基本方法是让 流水线暂停执行某些指令，而继续执行其它一些指令。

2. **分析为什么不可以禁止结构相关**

分析：允许结构相关的存在的原因主要分为两方面：

（1） 为了减少硬件代价

（2） 为了减少功能单元的延迟。如果为了避免结构相关而将流水线中的 所有功能单元完全流水化，或者设置足够的硬件资源，那么所带来的硬件代价必定很大。

3. 解决结构相关的方法

分析：发生结构相关时，必然会导致流水线效率的降低，如果处理不当甚至会发生错误。解决的方法主要包括：

（1）流水化功能单元

（2）资源重复

（3）暂停流水线。

体会

本次实验主要涉及到流水线中的结构相关的基本知识，需要在跟踪指令执行的基础上观察各个部件的执行情况。一般来说，如果某些指令组合在流水线中重叠执行时，产生资源冲突，则称该流水线有结构相关。为了能够在流水线中顺利执行指令的所有可能组合，而不发生结构相关，通常需要采用流水化功能单 元的方法或资源重复的方法。

同时，在本次实验，需要分析因为结构相关而导致的周期暂停的次数。由于多处结构相关可能发生在同一个周期内，因此结构相关的次数不一定等于周期的暂停次数。

作为最后的总结，本次实验通过流水线的结构图直观的展现了结构相关产生的原因，使我对于程序执行的底层逻辑有了更加深入的理解，

附录

实验汇编代码

```
1.        LHI      R2, (A>>16)&0xFFFF
2.        ADDUI    R2, R2, A&0xFFFF
3.        LHI      R3, (B>>16)&0xFFFF
4.        ADDUI    R3, R3, B&0xFFFF
5.        ADDU     R4, R0, R3
6.  loop:
7.        LD     F0, 0(R2)
8.        LD     F4, 0(R3)
9.        ADDD   F0, F0, F4
10.       ADDD   F2, F0, F2 ;; <- A stall is found (an example of how to answer your
    questions)
11.       ADDI   R2, R2, #8
12.       ADDI   R3, R3, #8
13.       SUB    R5, R4, R2
14.       BNEZ   R5, loop
15.       TRAP   #0         ;; Exit <- this is a comment !!
16.A:     .double 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
17.B:     .double 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
```