# 实验2 流水线及流水线中的冲突

## 1 实验目的

（1）加深对计算机流水线基本概念的理解。

（2）理解MIPS结构如何用5段流水线来实现，理解各段的功能和基本操作。

（3）加深对数据冲突和资源冲突的理解，理解这两类冲突对CPU性能的影响。

（4） 进一步理解解决数据冲突的方法，掌握如何应用定向技术来减少数据冲突引起的停顿。

## 2 实验平台

指令级和流水线操作级模拟器MIPSsim。

## 3 实验内容和步骤

（1）启动MIPSsim。

（2）进一步理解流水线窗口中各段的功能，掌握各流水寄存器的含义。（鼠标双击各段，即 可看到各流水寄存器的内容）

（3）载入一个样例程序（在本模拟器所在文件夹下的“样例程序”文件夹中），然后分别以单步执行一个周期、执行多个周期、连续执行、设置断点等方式运行程序，观察程序的执行情况，观察CPU中寄存器和存储器内容的变化，特别是流水寄存器内容的变化。

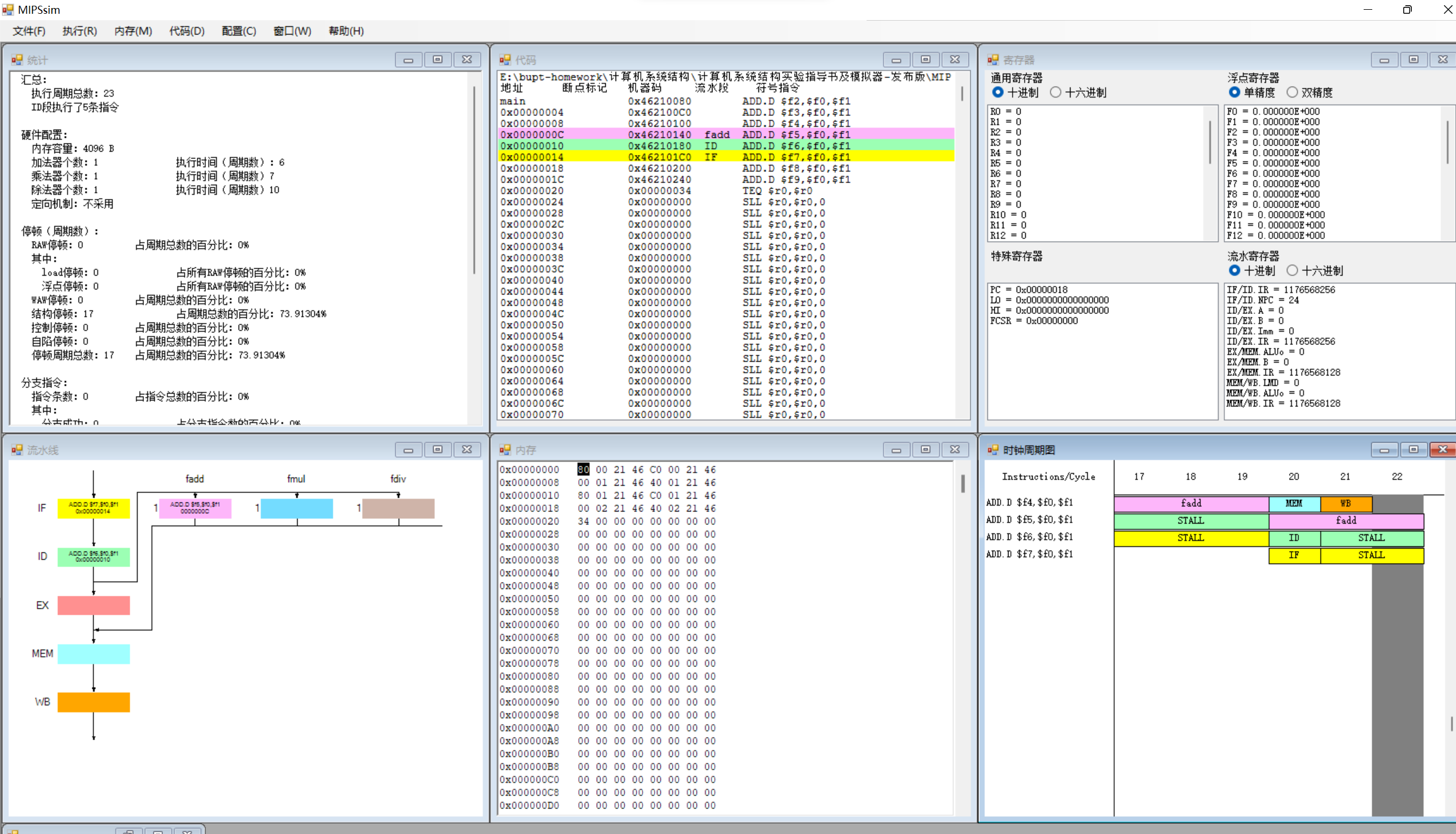
（4）选择配置菜单中的“流水方式”选项，使模拟器工作于流水方式下。

（5）观察程序在流水方式下的执行情况。

（6）观察和分析结构冲突对CPU性能的影响，步骤如下：

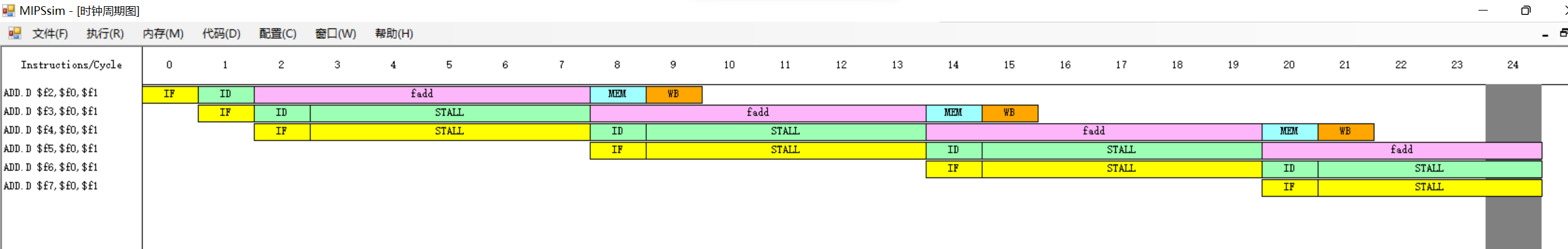
1) 加载structure\_hz.s（在模拟器所在文件夹下的“样例程序”文件夹中）。

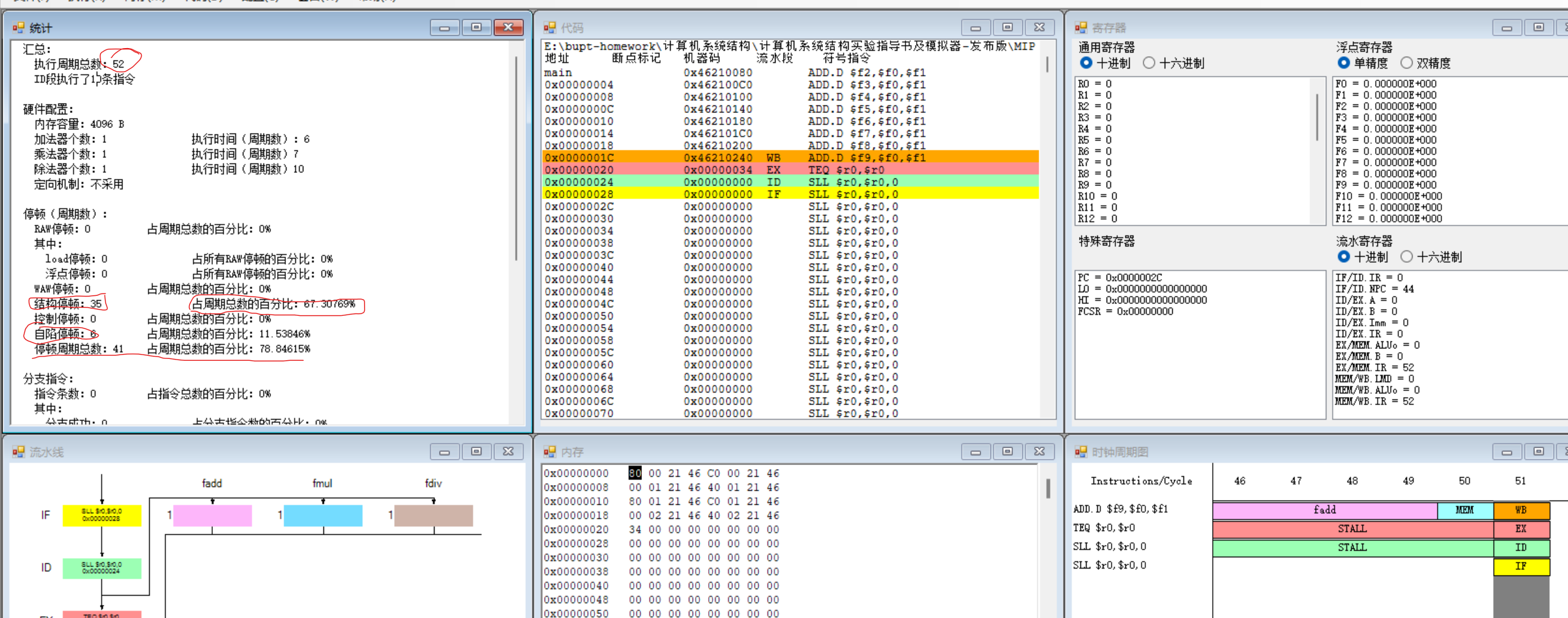
2) 执行该程序，找出存在结构冲突的指令对以及导致结构冲突的部件。



可以发现，存在冲突的部件为加法部件，因为在每个加法的执行EX阶段，都会长时间占用加法部件，导致后续指令停顿。

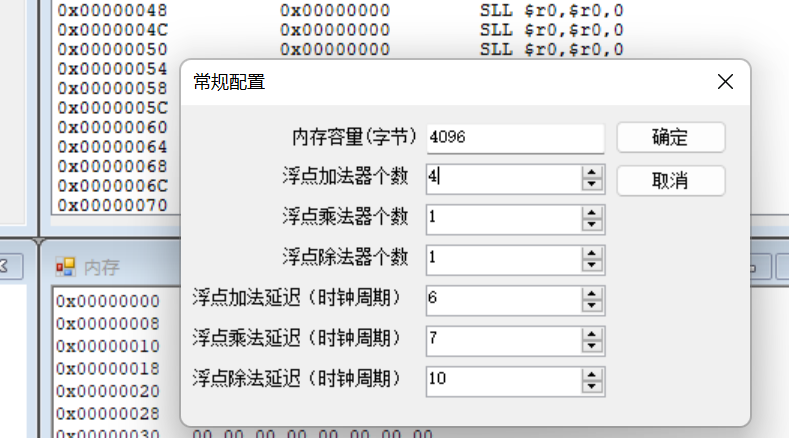
3) 记录由结构冲突引起的停顿周期数，计算停顿周期数占总执行周期数的百分比。



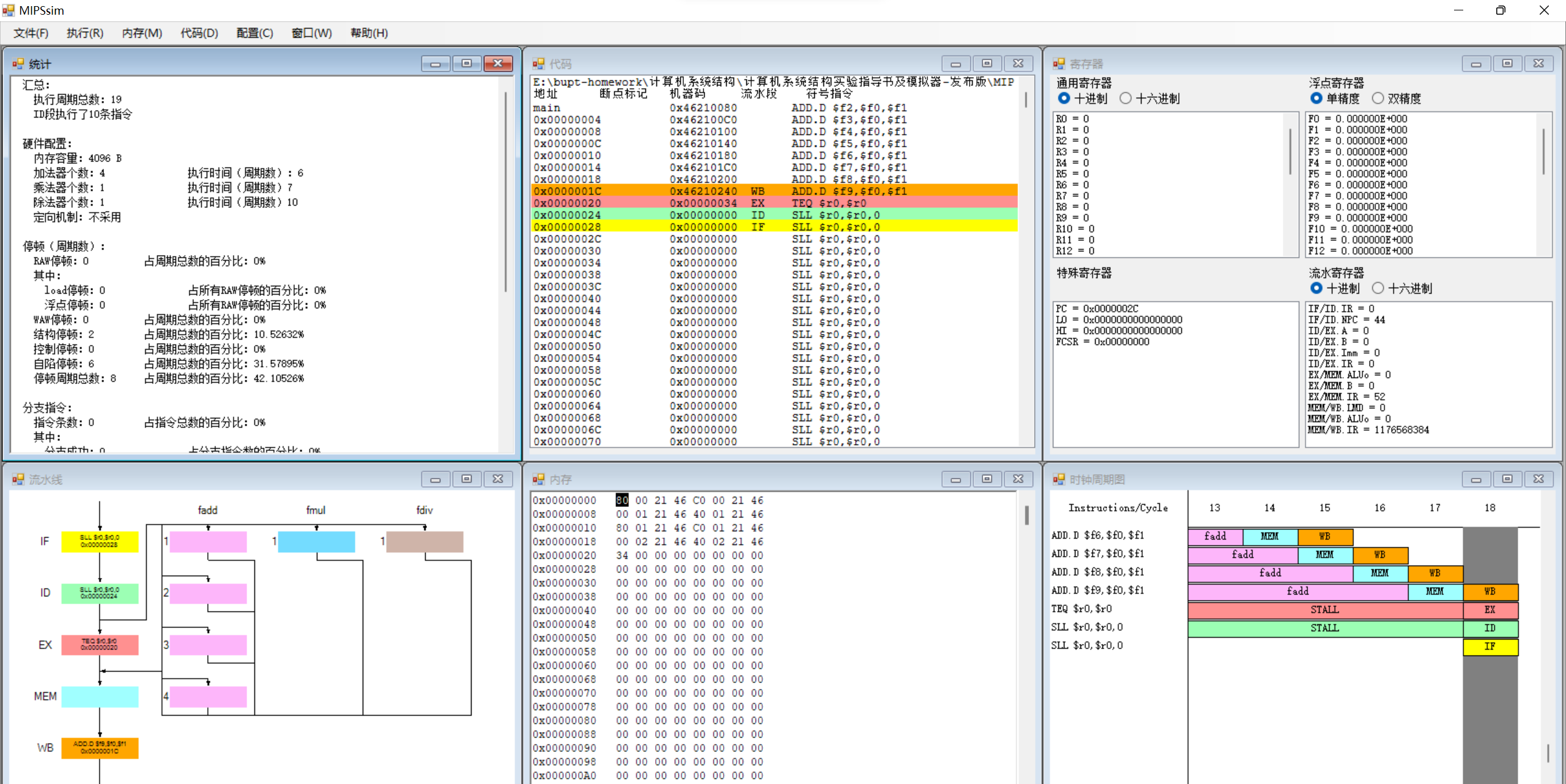


可以看到，总的执行周期为52，结构停顿周期数为35，占周期总数的百分比为：67.30769%。

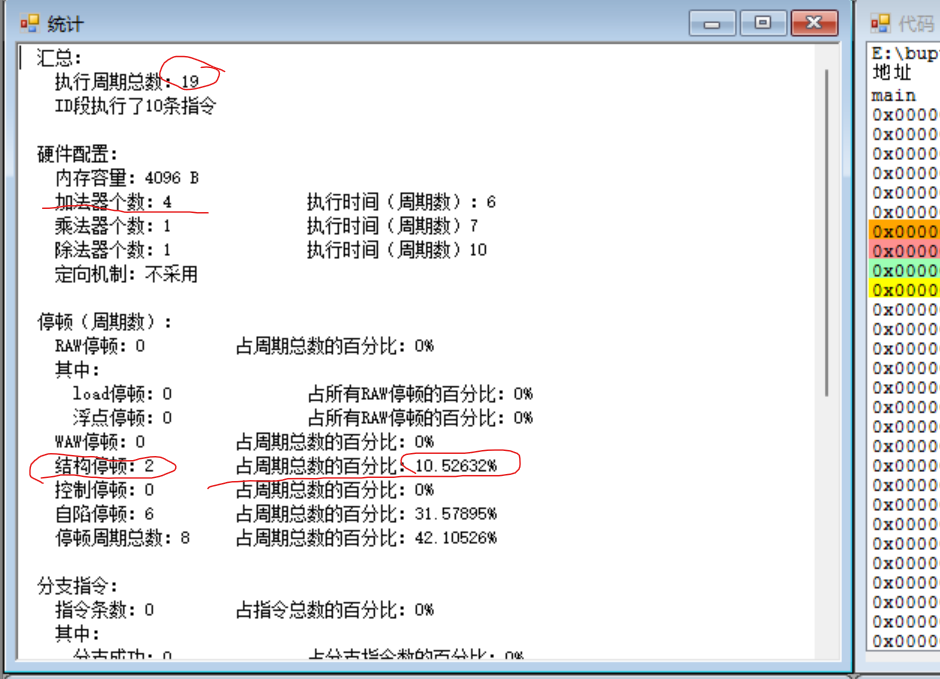
4) 把浮点加法器的个数改为4个。



5) 再重复1-3的步骤。



执行结果如上。停顿数如下，一共只有两个结构停顿，一共有19个执行周期，结构停顿占周期总数的10.52632%。



6) 分析结构冲突对CPU性能的影响，讨论解决结构冲突的方法。

结构冲突会降低CPU性能，导致停顿周期变多，增加了所有指令执行的时间。为了缓解、解决结构冲突，应该增加导致结构冲突的功能部件的数量。

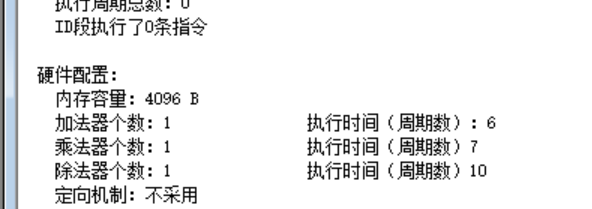
（7）观察数据冲突并用定向技术来减少停顿，步骤如下：

1) 全部复位。

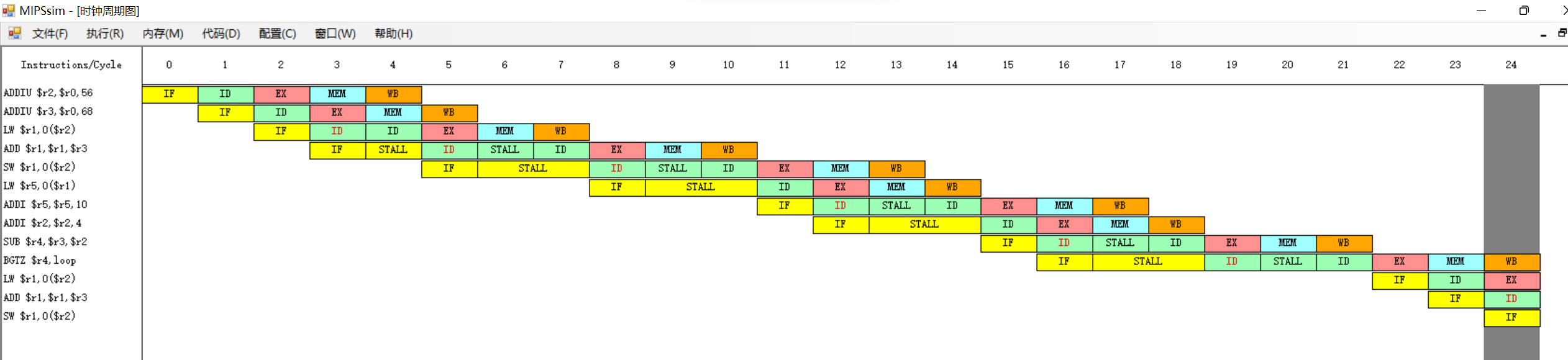
2) 加载data\_hz.s（在模拟器所在文件夹下的“样例程序”文件夹中）。

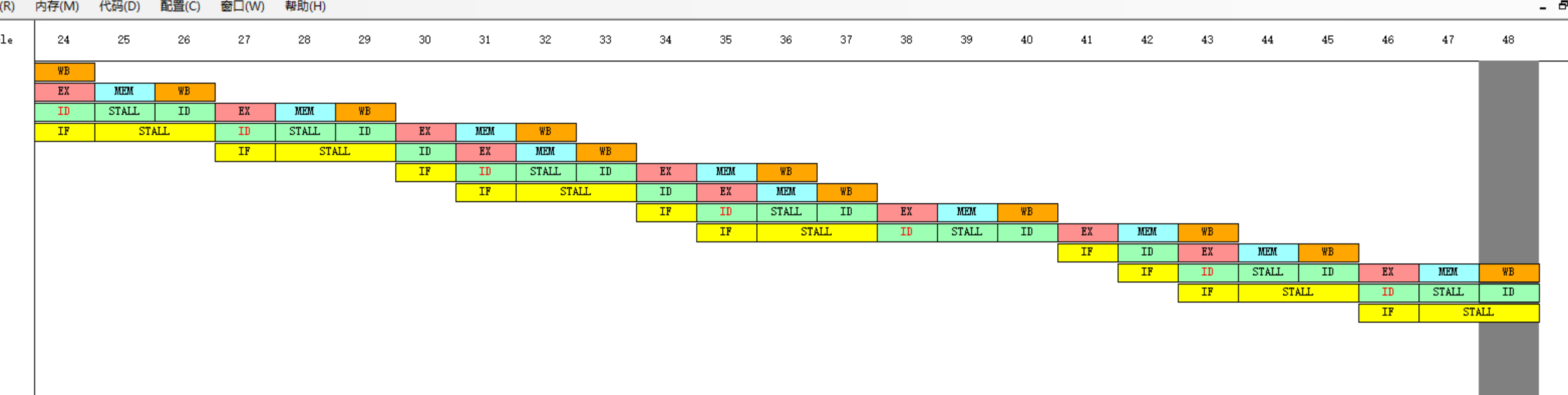


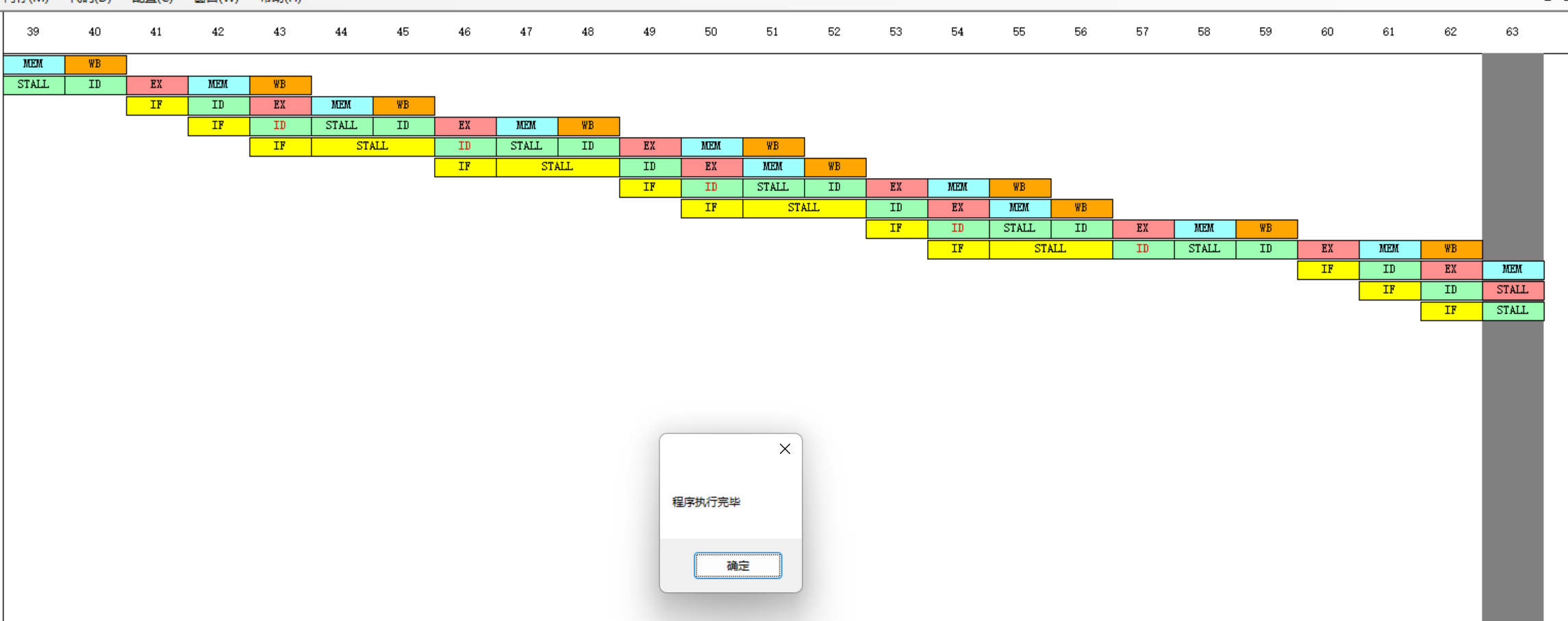
3) 关闭定向功能（在“配置”菜单下选择取消“定向”）。



4) 用单步执行一个周期的方式执行该程序，观察时钟周期图，列出什么时刻发生了 RAW冲突。



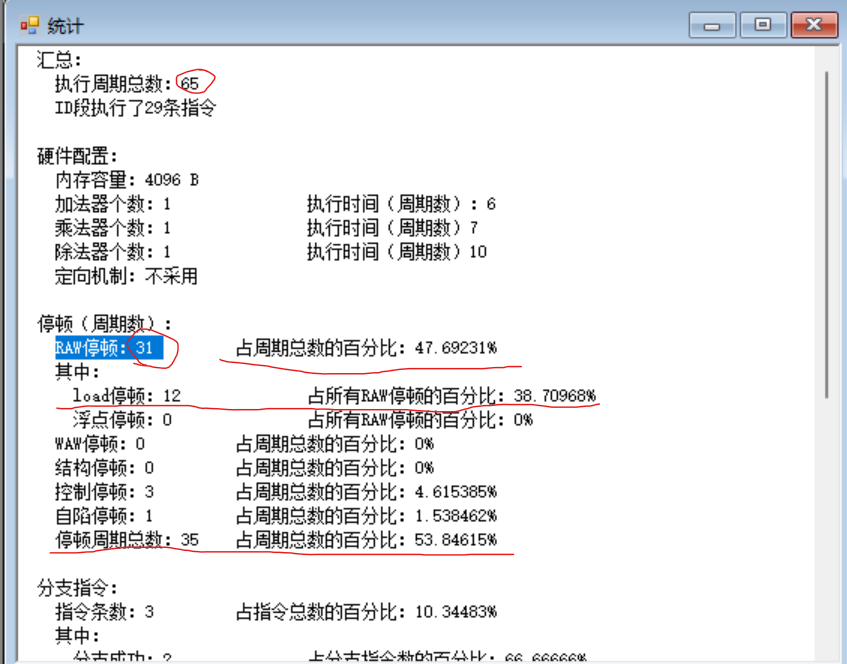




可以看到，当出现stall的时候说明出现了RAW冲突。所以出现冲突的周期为4、6、7、9、10、13、14、17、18、20、25、26、28、29、32、33、36、37、39、44、45、47、48、51、52、55、56、58、63周期

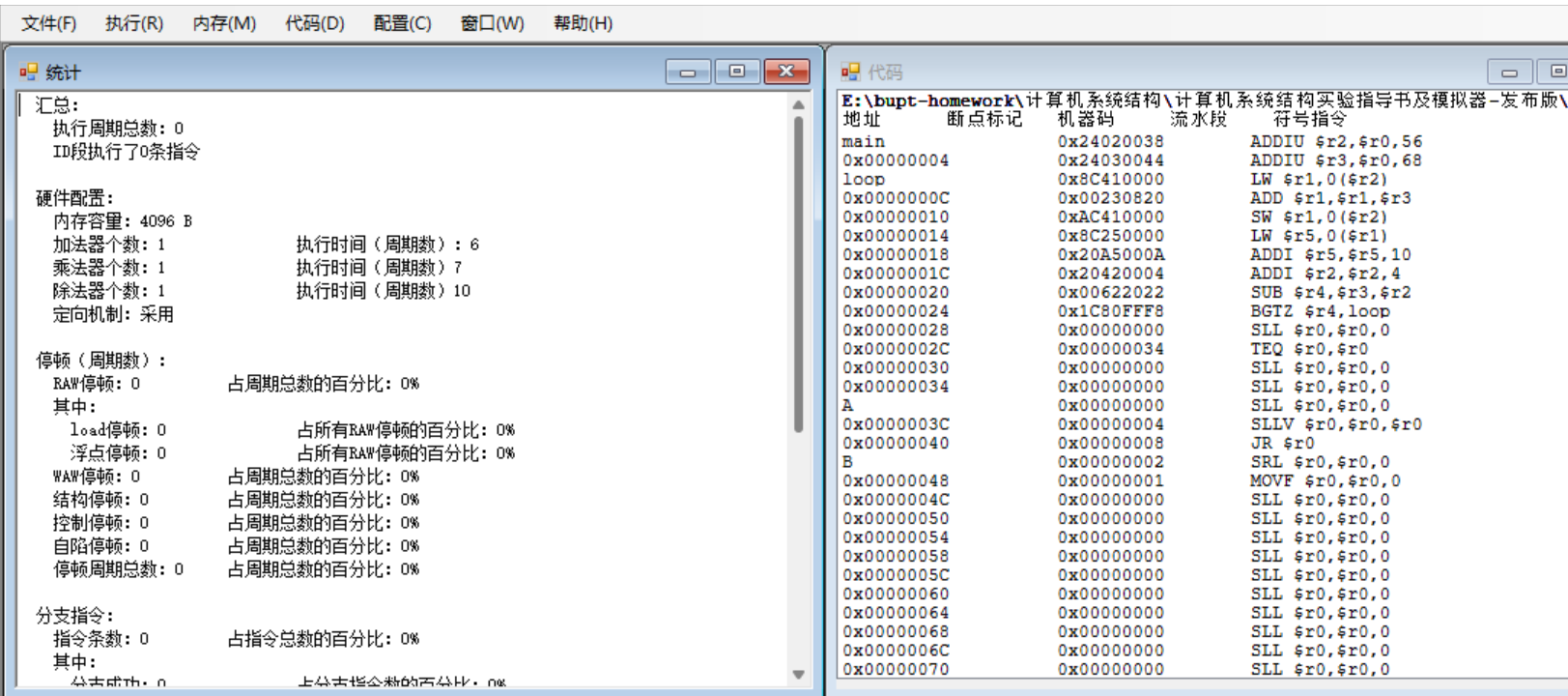
5) 记录数据冲突引起的停顿周期数以及程序执行的总时钟周期数，计算停顿时钟周期47数占总执行周期数的百分比。

如下图所示，数据冲突引起的停顿周期数为31，占总时钟周期数的47.69231%。总停顿数为35，占总时钟周期的53.84615%。

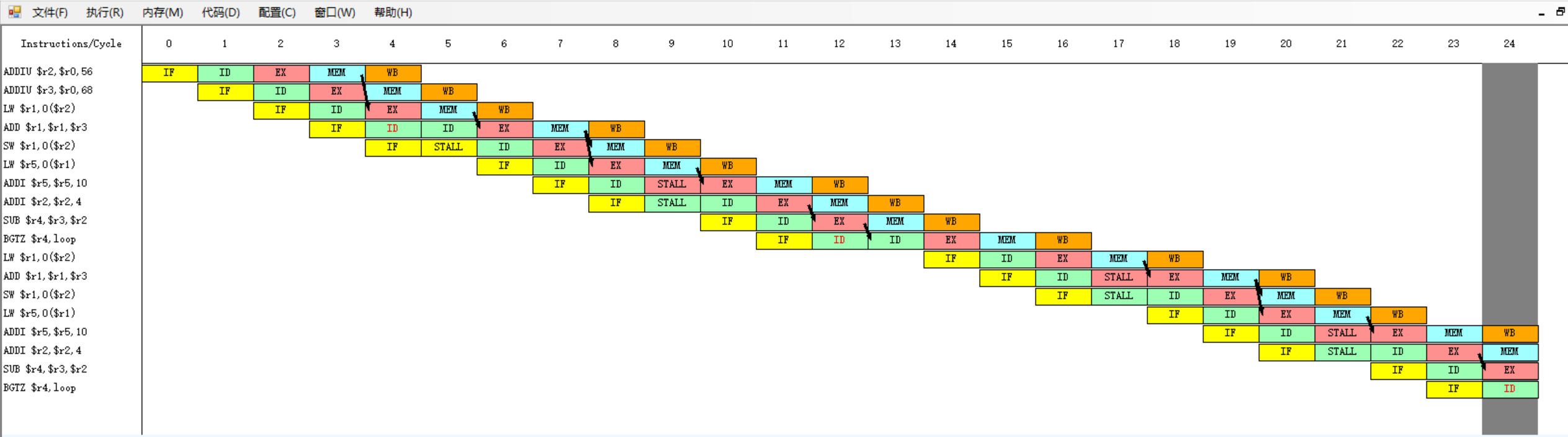


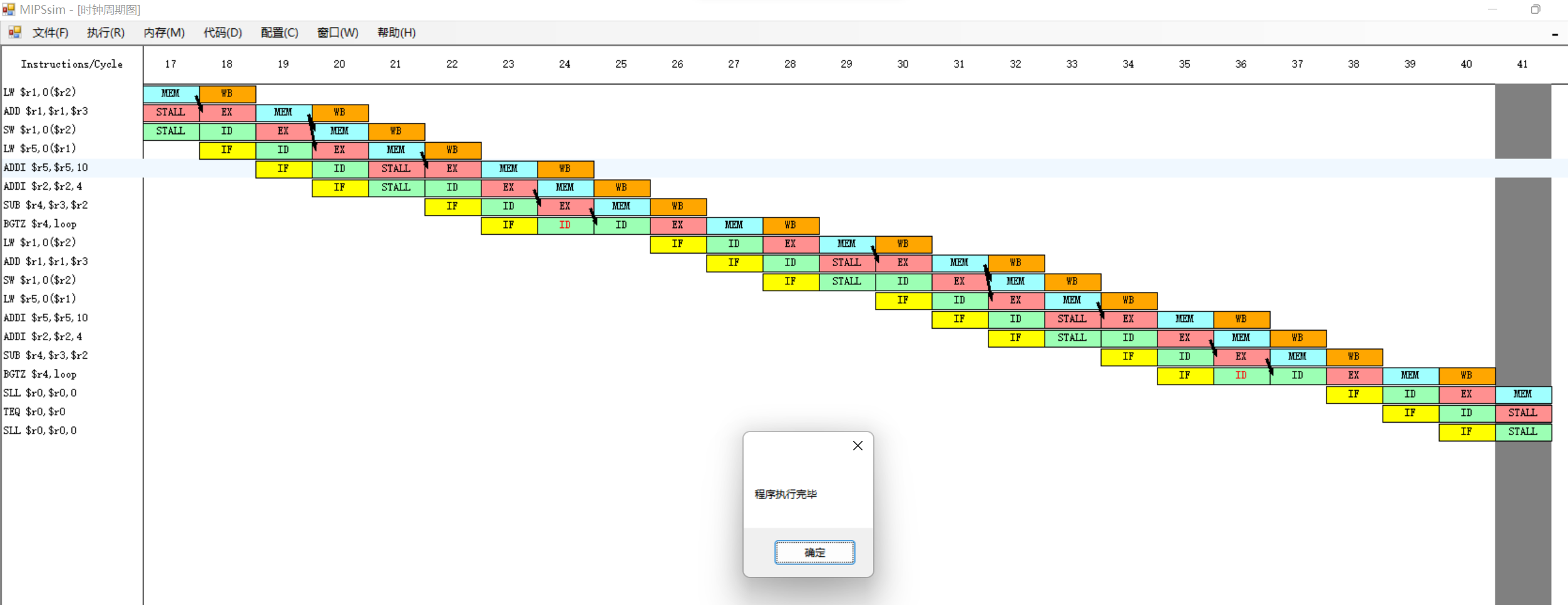
6) 复位CPU。

7) 打开定向功能。



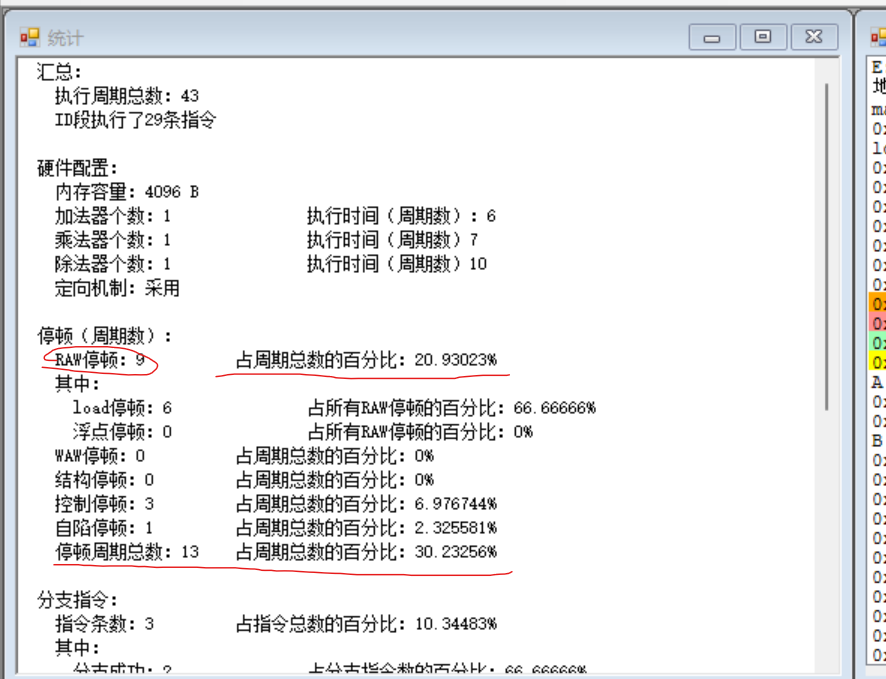
8) 用单步执行一个周期的方式执行该程序，查看时钟周期图，列出什么时刻发生了 RAW冲突，并与步骤3）的结果比较。 时钟周期图为：





发生RAW冲突的周期号为：5、9、12、17、21、24、29、33、36。

9) 记录数据冲突引起的停顿周期数以及程序执行的总周期数。计算采用定向以后性能比原来提高多少。



如上图所示，RAW停顿只有9个，总周期为43

RAW停顿占总周期数的20.93023%。

总停顿有13个，占总周期数的30.23256%。

采用定向技术后，性能提高到了原来的65/43≈1.512倍。