

**计算机系统结构**



**使用MIPS指令实现求两个数组的点积**

**姓 名 鄭毓恒**

**学 院 计算机学院**

**专 业 计算机科学与技术**

**班 级 2020211302**

**学 号 2020211262**

**任课教师 邝坚**

**2023年 4 月**

**实验目的**

1. 通过实验熟悉实验1和实验2的内容。
2. 增强汇编语言编程能力。
3. 学会使用模拟器中的定向功能进行优化。
4. 了解对代码进行优化的方法。

**实验原理**

自行编程一个计算两个向量点积的汇编程序，该程序要求可以实现求两个向量点积计算后的结果。两个向量元素使用数组进行数据存储，要求向量维度不得小于10。在使用和不使用的定向功能的情况下，对比程序的执行结果和效率。

向量的点积：假设有两个向量a、b，则a与b的点积为：



然后，使用静态调度方法重排指令序列，减少相关，优化程序。在本次实验，静态调度方法减少数据冲突发生的次数，减少因为数据冲突而停顿的时钟周期数。在使用和不使用的定向功能的情况下，和未优化程序比较执行结果和效率。

**向量点积程序代码清单及注释说明**

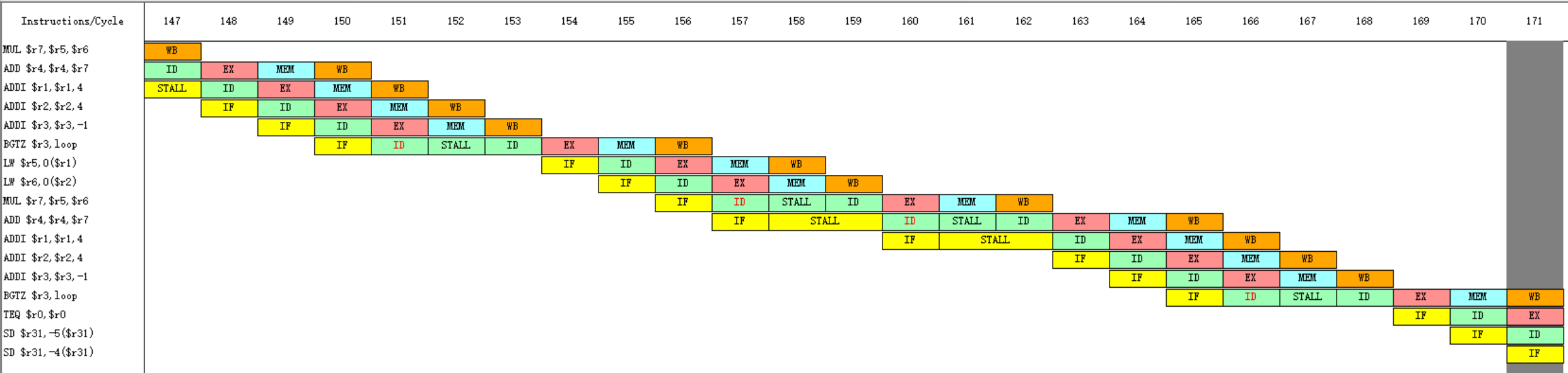
|  |
| --- |
| .text  main:  ADDIU $r1,$r0,a1 #向量1地址  ADDIU $r2,$r0,a2 #向量2地址  ADDI $r3,$r0,11 #以向量长度初始化循环变量  ADDI $r4,$r0,0 #存放结果  loop:  #加载当前相乘元素  LW $r5,($r1)  LW $r6,($r2)  MUL $r7,$r5,$r6 #相乘  ADD $r4,$r4,$r7 #加入点积结果  ADDI $r1,$r1,4 #下一个元素地址  ADDI $r2,$r2,4 #下一个元素地址  ADDI $r3,$r3,-1 #循环次数-1  BGTZ $r3,loop #循环判断  TEQ $r0,$r0  .data  a1:  .word -5,-4,-3,-2,-1,1,2,3,4,5,6  a2:  .word 5,4,3,2,1,1,2,3,4,5,6 |

**优化后的程序代码清单**

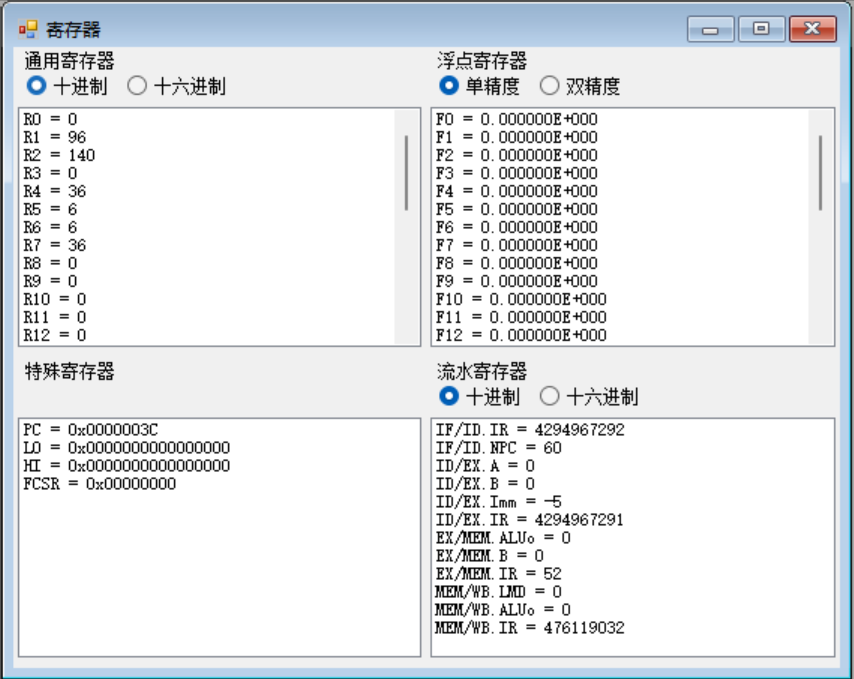
|  |
| --- |
| .text  main:  ADDIU $r1,$r0,a1 #向量1地址  ADDIU $r2,$r0,a2 #向量2地址  ADDI $r3,$r0,11 #以向量长度初始化循环变量  ADDI $r4,$r0,0 #存放结果  loop:  #加载当前相乘元素  LW $r5,($r1)  LW $r6,($r2)  ADDI $r3,$r3,-1 #循环次数-1  MUL $r7,$r5,$r6 #相乘  ADDI $r1,$r1,4 #下一个元素地址  ADDI $r2,$r2,4 #下一个元素地址  ADD $r4,$r4,$r7 #加入点积结果  BGTZ $r3,loop #循环判断  TEQ $r0,$r0  .data  a1:  .word -5,-4,-3,-2,-1,1,2,3,4,5,6  a2:  .word 5,4,3,2,1,1,2,3,4,5,6 |

**未优化代码和优化代码性能分析比较结果**

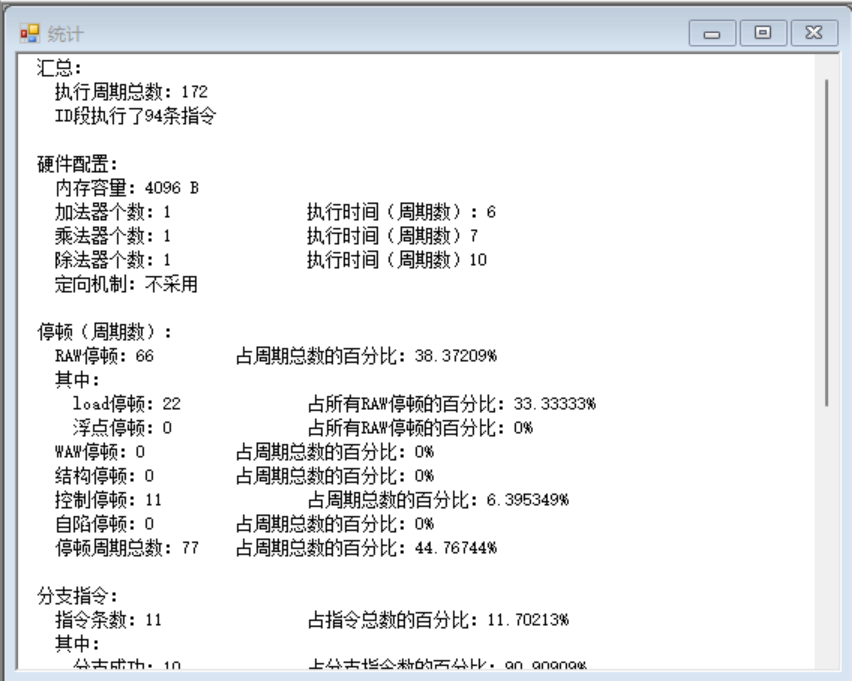
运行自己编写的程序，流水线的执行情况如下。



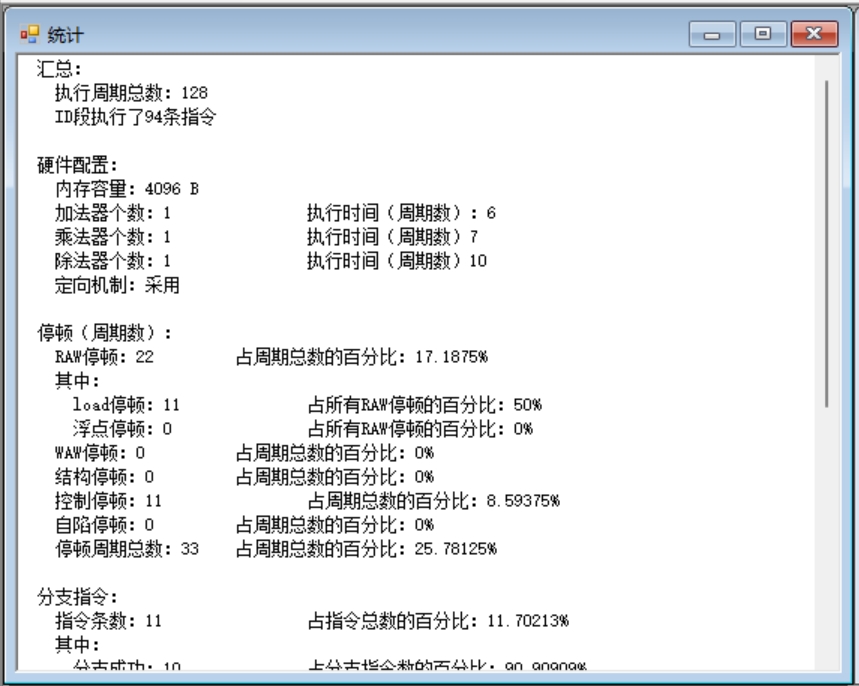
程序运行结果存放在寄存器R4，值为36，计算结果正确。



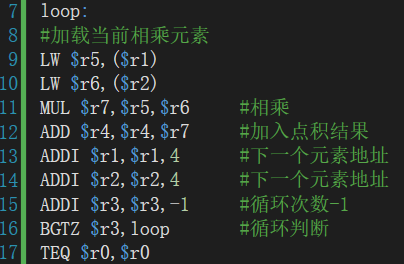
程序执行周期总数为172，RAW停顿周期数为66，占周期总数的38.37209%。



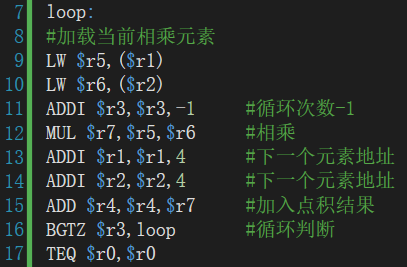
使用定向功能再次执行代码。程序执行周期总数降至128，RAW停顿周期数为22，占周期总数的17.1875%。与没有使用定向功能时相比，效率提升了1.34倍。



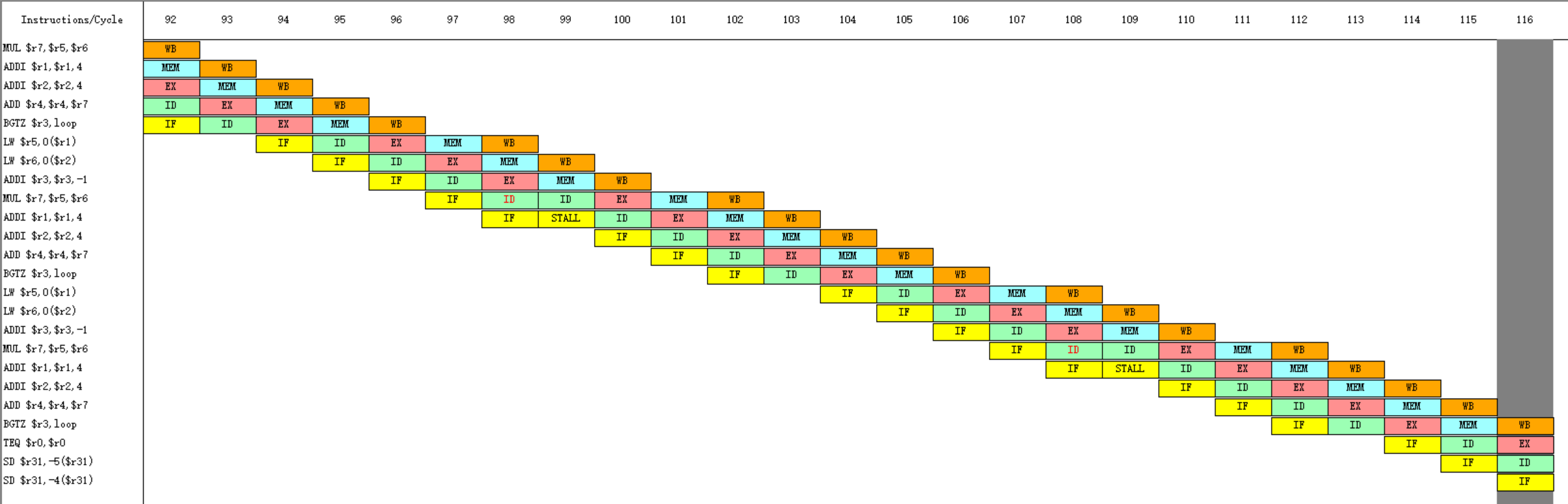
第11行指令和第9、10行指令有RAW相关。第16行指令和第15行指令有RAW相关。



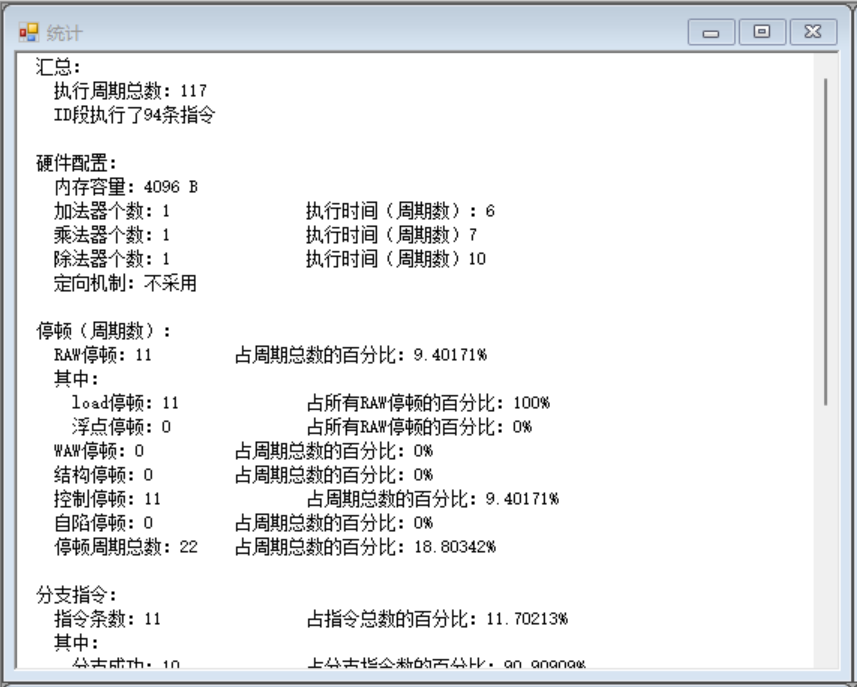
采用静态调度的方法将第13、14、15行指令进行重排，穿插在别的指令中，减少RAW冲突。



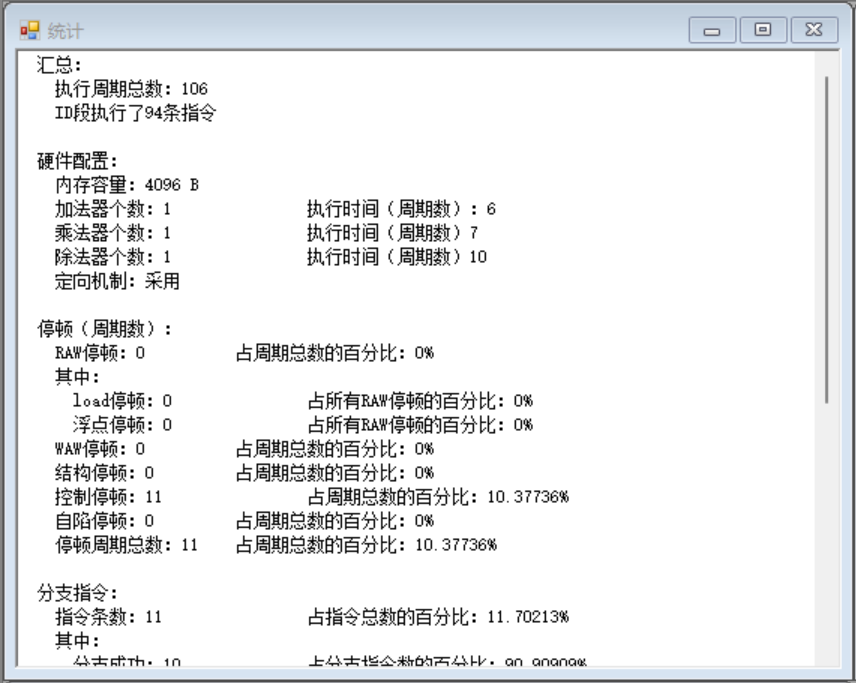
运行优化程序，流水线的执行情况如下。



程序执行周期总数为117，RAW停顿周期数为11，占周期总数的9.40171%。比没有使用定向功能的未优化代码，效率提升了1.47倍。



使用定向功能再次执行代码。程序执行周期总数降至106，RAW停顿周期数为0。比使用了定向功能的未优化代码，效率提升了1.21倍。



**实验总结**

通过编写汇编程序实现向量点积计算，增强了汇编语言编程能力，学习了如何使用静态调度优化代码。通过对比未优化和优化代码的性能，加深了我对数据冲突的理解，了解到定向功能和静态调度解决数据冲突的原理以及对提升CPU性能的帮助。