

**课 程 实 验 报 告**

**课程名称： 计算机系统基础**

**实验名称： ARM指令系统的理解**

**院 系 ：计算机科学与技术**

**专业班级 ： 图灵2301**

**学 号 ： U202311239**

**姓 名 ： 刘星佳**

**指导教师 ： 王多强**

**2024 年 11 月 5 日**

**一、实验目的与要求**

通过在ARM虚拟环境下调试执行程序，了解 ARM的指令系统。

实验环境：ARM 虚拟实验环境 QEMU

工具：gcc, gdb 等

**二、实验内容**

**任务1、C与汇编的混合编程**

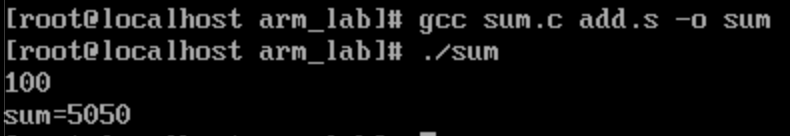
**任务2、内存拷贝及优化实验**

程序及操作方法 见 <ARM实验任务.pdf>

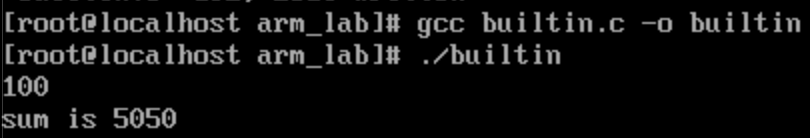
**三、实验记录及问题回答**

**（1）实验任务的实验结果记录**

1. C语言调用汇编实现累加和求值

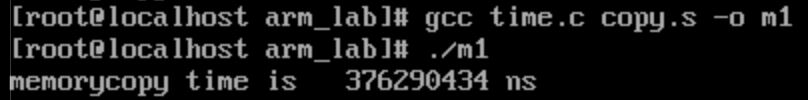


1. C语言内嵌汇编

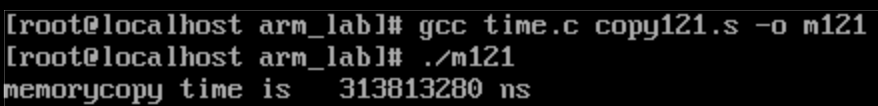


1. 循环展开优化

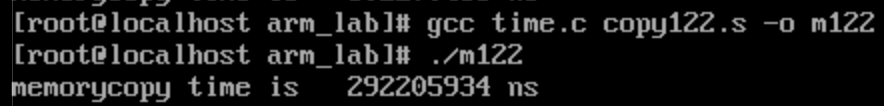
原始代码：



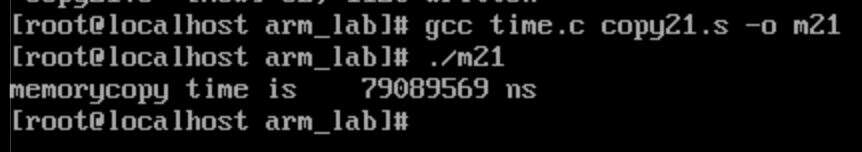
2倍展开循环优化：



4倍展开循环优化：



1. 内存突发传输方式优化



**（2）ARM 指令及功能说明**

查阅《ISA\_A64\_xml\_A\_profile-2023-09.pdf》，指出10条不同指令（存数、取数、算术运算、转移指令、函数调用等都应覆盖）的功能。

1. LDR X0, [X1]

将内存地址X1中存储的值加载到寄存器X0中。

1. LDP X0, X1, [X2]

从X2的内存地址连续加载两个值存储到寄存器X0和X1中。

1. STR X0, [X1]

将寄存器X0的值存储到内存地址X1中。

1. STP X0, X1, [X2]

将寄存器X0和X1的值分别存储到X2指定的内存地址及其下一个地址中。

1. ADD X2, X0, X1

将寄存器X0和寄存器X1相加的值存储到寄存器X2中。(无符号加法）

1. SUB X2, X0, X1

将寄存器X0和寄存器X1相减的值存储到寄存器X2中。（无符号减法）

1. B label

无条件跳转到标签label的地址。

1. BL label

跳转到标签label的地址并保存返回地址到X30寄存器。（用于实现类似函数调用的功能）

1. CMP X0, X1

比较寄存器X0和X1的值并更新条件标志。

1. MOV X0, X1

将一个寄存器X0的值复制到另一个寄存器X1。

**四、体会**

通过本次实验，我对 ARM 指令系统及其应用有了更直观的认识。通过在 QEMU 环境中调试程序，深入理解了 ARM 指令的功能及其优化策略，特别是指令级优化对性能提升的重要性。在 C 与汇编混合编程的尝试中，我感受到底层编程的灵活性，熟悉了寄存器操作与内存管理的细节。内存拷贝及循环展开的优化让我明白高效代码编写不仅依赖算法，还需要针对体系结构特点精心设计。在完成实验的过程中，我的动手能力和问题分析能力也得到了提升。这次实验让我体会到理论与实践结合的重要性，进一步激发了我对底层系统开发的兴趣。