电子科技大学信息与软件工程学院

**实 验 报 告**

学 号 2019091618017

姓 名 陈葳亘

（实验）课程名称《ARM处理器体系结构及

应用》课程实验

理论教师 兰 刚

实验教师 兰刚

**电 子 科 技 大 学**

**实 验 报 告**

**学生姓名： 陈葳亘 学号： 2019091618017 指导教师： 兰刚**

**实验地点： 信软楼西304 实验时间：**

1. 实验名称：ARM主程序调用ARM/C语言子程序
2. 实验学时：4
3. 实验目的：
4. 了解ARM应用程序框架。
5. 了解ARM汇编程序函数和C语言程序函数相互调用时，遵循的ATPCS标准；
6. 了解和掌握ARM汇编程序调用C语言程序函数的基本方法；
7. 了解和掌握ARM汇编程序调用C语言程序函数的参数传递过程。
8. 实验原理：
9. **ARM工程**

由于C语言便于理解，有大量的支持库，所以它是当前ARM程序设计所使用的主要编程语言。

对硬件系统的初始化、CPU状态设定、中断使能、主频设定 以及RAM控制参数初始化等C程序力所不能及的底层操作，还是要由汇编语言程序 来完成。

ARM工程 的各种源文件之间的关系，以及最后形成可执行文件的过程如下图1所示：

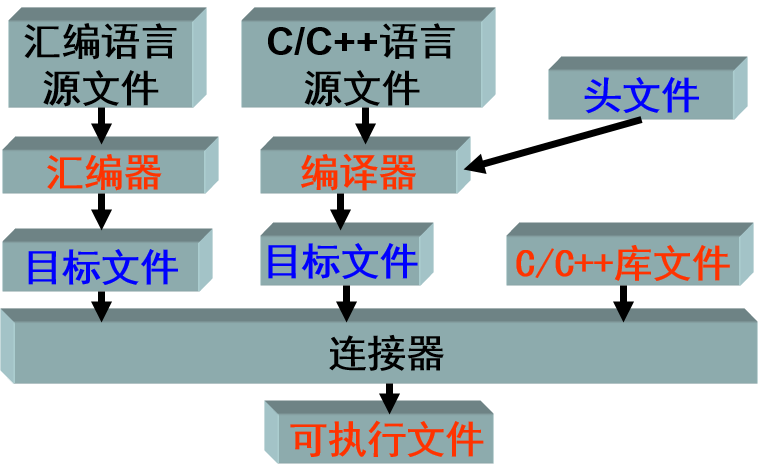


图1 汇编语言和C语言混合编译链接示意图

在应用系统的程序设计中，若所有的编程任务均用汇编语言 来完成，其工作量是可想而知的，这样做也不利于系统升级或应用软件移植。

通常汇编语言部分完成系统硬件的初始化；高级语言部分完成用户的应用。

执行时，首先执行初始化部分，然后再跳转到C/C++部分。整个程序结构显得清晰明了，容易理解。为方便工程开发，ARM公司的开发环境ARM ADS为用户提供了一个可以选用的应用程序框架。该框架把为用户程序做准备工作的程序分成了： 启动代码 和 应用程序初始化 两部分。

用于硬件初始化的汇编语言部分叫做 启动代码；用于应用程序初始化的C部分叫做初始化部分。整个程序如下图2所示：

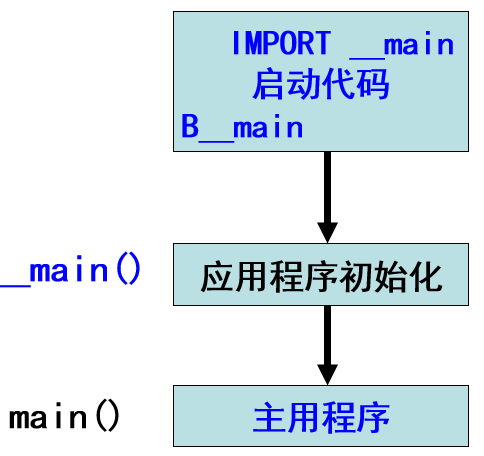


图2 ARM应用程序框架

1. **过程调用标准ATPCS**

在ARM工程中，C程序调用汇编函数和汇编程序调用C函数是经常发生的事情。为此人们制定了ARM-Thumb过程调用标准ATPCS（ARM-Thumb Procedure Call Standard）。

1. ATPCS规定，ARM的数据堆栈为FD型堆栈，即递减满堆栈。
2. ATPCS标准规定，对于参数个数不多于4的函数，编译器必须按参数在列表中的顺序，自左向右为它们分配寄存器R0~R3。其中函数返回时，R0还被用来存放函数的返回值。
3. 如果函数的参数多于4个，那么多余的参数则按自右向左的顺序压入数据堆栈，即参数入栈顺序与参数顺序相反。
4. 根据ATPCS的C语言程序调用汇编函数，参数由左向右依次传递给寄存器R0~R3的规则。
5. **子程序的调用与返回**

人们把可以多次反复调用的、能完成指定功能的程序段称为“子程序”。把调用子程序的程序称为“主程序”。

为进行识别，子程序的 第1条指令 之前必须赋予一个 标号，以便其他程序可以用这个标号调用子程序。

在 ARM 汇编语言程序中，主程序一般通过 BL 指令来调用子程序。该指令在执行时完成如下操作：将子程序的返回地址存放在连接寄存器 LR 中，同时将程序计数器 PC 指向子程序的入口点。

为使子程序执行完毕能 返回 主程序的调用处，子程序末尾 处应有 MOV、LDMFD 等指令，并在指令中将返回地址重新复制到 PC 中。

在调用子程序的同时，也可以使用 R0~R3 来进行 参数的传递 和从子程序返回 运算结果。

1. 实验内容：
2. ARM指令主程序调用ARM指令子程序；
3. ARM指令主程序调用C语言子程序，输入的6个参数为1、2、3、4、5、6;
4. 子程序的参数个数要求至少6个， C语言子程序实现的功能为：（i1+i2+i3+i4）\*i5-i6。
5. 分析通过反汇编得到的C程序的ARM指令代码段，了解参数传递过程。
6. 实验器材（设备、元器件）：
7. PC机一台；
8. Keil MDK-ARM uVision4开发工具。
9. 实验步骤：
10. 打开Keil MDK-ARM uVision4开发工具；
11. 新建一个工程文件；
12. 在新建的工程文件中，添加新的源程序文件
13. 编写代码
14. 选择“Build target”菜单对编写好的工程文件进行编译链接。
15. 点击““Start/Stop Debug Section””按键，对程序进行跟踪调试，在调试界面，单步执行，对CPU各寄存器的值的变化、以及相关内存的变化进行分析比较，判断程序的执行是否符合预期要求。
16. 实验结果与分析（含重要数据结果分析或核心代码流程分析）
17. **ARM指令主程序调用ARM指令子程序；**
18. **程序代码**

程序代码如代码1所示：

代码1 ARM指令主程序调用ARM指令子程序

|  |
| --- |
| AERA Init, CODE, READONLY  ENTRY  start LDR R0,=0X3FF5000  LDR R1,=0XFF  STR R1,[R0]  LDR R0,=0X3FF5008  LDR R1,=0X01  STR R1,[R0]  BL PR  PR:  MOV R2,#3  MOV PC,LR  END |

1. **运行过程及结果界面截图**

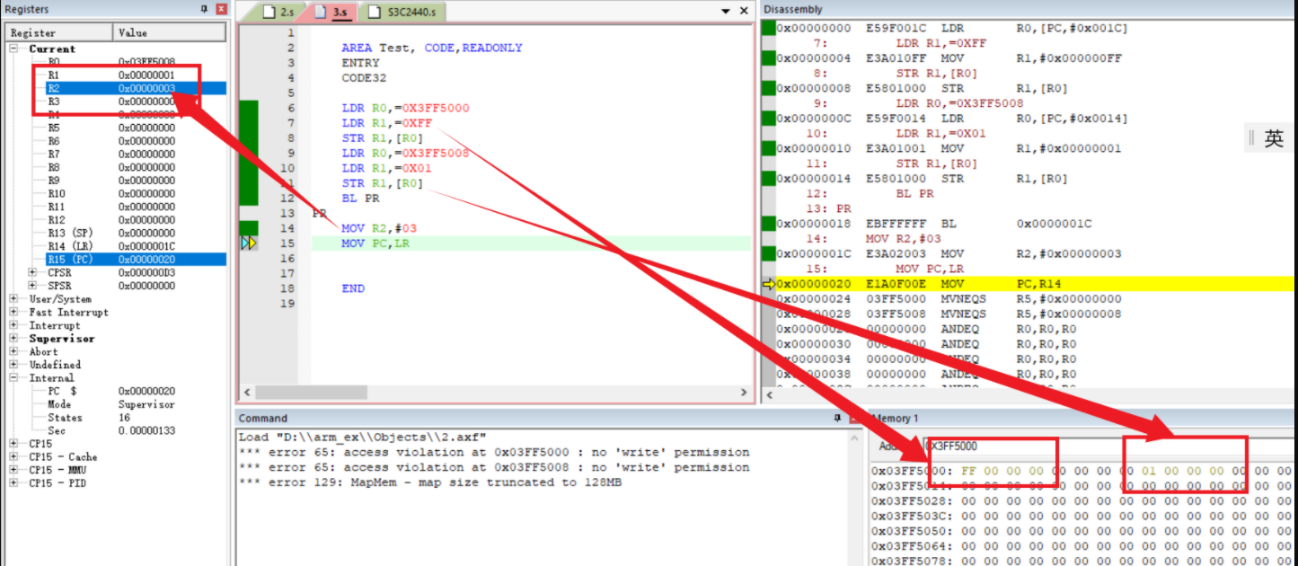


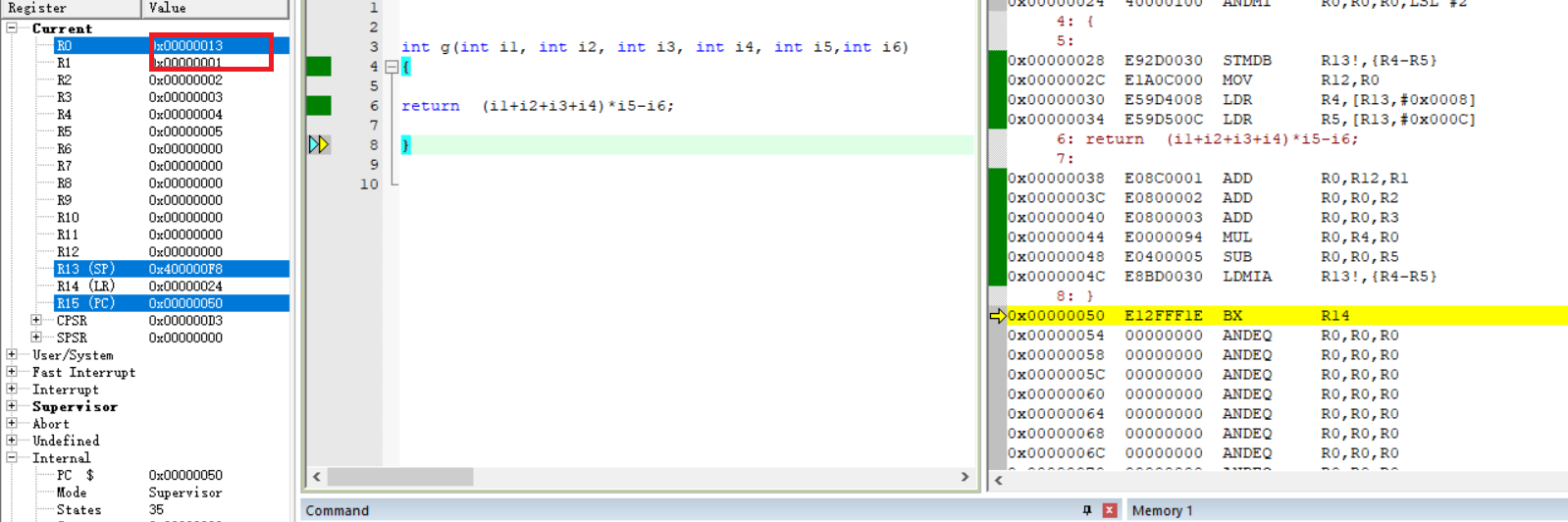
图1 ARM指令主程序调用ARM指令子程序运行结果截图

1. **实验结果分析**
2. **实验结论**
3. **ARM指令主程序调用C语言子程序；**
4. **程序代码**

ARM指令主程序如代码2所示

代码2 RM指令主程序调用C语言子程序的RM指令主程序

|  |
| --- |
| PRESERVE8    AREA tt, CODE, READONLY  IMPORT g ;  ENTRY    LDR SP,=0x40000100;  MOV R0,#0  MOV R1,#1;  MOV R2,#2  MOV R3,#3  MOV R4,#4  MOV R5,#5    stmfd sp!,{r5,r4}    BL g ;    END |

C语言子程序代码如代码3所示

代码3 RM指令主程序调用C语言子程序的C语言子程序

|  |
| --- |
| int g(int i1, int i2, int i3, int i4, int i5,int i6)  {    return (i1+i2+i3+i4)\*i5-i6;  } |

1. **运行过程及结果界面截图**

图2 程序执行完STR LR, [SP, #-4]!的界面截图

图3是程序即将调用C子程序的截图。

图3 程序即将调用C子程序的截图

图4是程序跟踪进入C语言子程序的截图。

图4 程序跟踪进入C语言子程序的截图

1. **实验结论**
2. 总结及心得体会：

根据自己试验心得总结

说明ARM汇编程序中，参数多余4个（比如为6个）时，参数是如何压栈传递的？通过C语言对应的汇编程序，分析C语言子程序是如何取得相关传入参数的？

1. 对本实验过程及方法、手段的改进建议：

根据实验过程，提出改建建议。

报告评分：

指导教师签字：