

Lab2 实验报告

王嵘晟

PB17111614

1. 初始化过程

把寄存器 R6 作为栈指针,开辟栈从 XF000 开始。所以把 XF000 存入 R6。
寄存器 R4 存储 0 的 ASCII 码值的复数补码,用以做 GETC()-0 操作。JSR 过后
R7 存储 RET 指回地址,把 JSR 后的 R7 内容存储到栈里。R5 是帧指针,指向
程序的活动地址,在 main 函数中先把 R5 中本来的值存入栈,再赋初始值 R6-
1,用 TRAP 读取输入并入栈,将 a 到 f 依次存入栈中,完成初始化

2. 调用规则

GETC () 用 TRAP x23 来实现,它会改变 R0 的值,令 R0 存入输入内容的
ASCII 码,输入的值不能超出 ASCII 码的范围。而如果栈上溢则会输出字符
串 You idoit,the stack overflows! 用 TRAP x22 来实现,起始地址由 R0 指定,
执行这条 TRAP 指令时 R0 存储内容不能被改变。HALT 用 TRAP x25 实现。

3. 其他准则

在调用 Func 函数前,需要将 a,b,c,d,e,f 还有 n 都存储在同一个工作栈中,
栈指针指向栈顶。帧指针 R5 在栈指针 R6 下面,中间的内存空间存储实参。
并且在执行 Func 函数前,还要对栈的状态进行判断,如果栈上溢了,则不执
行 Func 函数,程序直接结束

4. 错误处理

当栈上溢出时,不调用 Func 函数,在屏幕上输出字符串" You idoit,the
stack overflows!"