lab3_report

练习1:内核从完成必要的初始化到用户态程序的过程是怎么样的？尝试描述一下调用关系。

答：首先运行`_start`函数，并执行`init_c`，这一系列操作包括执行`clear_bss`清除kernel image的bss area，`early_uart_init`初始化uart，`init_boot_pt`初始化boot page table，`el1_mmu_activate`启用MMU，以及调用`start_kernel`跳转到高地址，进而跳转到内核的 main 函数。

接下来，执行`kernel/arch/aarch64/main.c`里的`main`函数来进入用户态程序。这一系列操作包括执行`uart_init`来初始化uart，`mm_init`初始化内存管理模块，`arch_interrupt_init`来初始化异常向量表，然后调用`create_root_thread`来创建`root_thread`，也就是第一个用户线程。

`create_root_thread`的一系列操作包括：

1. `create_root_cap_group`创建第一个用户进程
2. `__create_root_thread`在第一个用户进程中创建第一个用户线程root_thread，并获得其标识符`thread_cap`
3. `obj_get`获取`thread_cap`对应的线程对象，也就是`root_thread`
4. `obj_put`用于与`obj_get`相对应来改变`refcount`的值
5. `switch_to_thread`将`root_thread`设置为当前线程

在`main`函数的最后，使用`switch_context`获取栈堆指针寄存器的上下文指针，再执行`eret_to_thread`函数实现进程上下文切换，完成了从内核模式到用户模式的切换，并在用户模式下开始运行用户代码。

至此，我们实现了内核从完成必要的初始化到用户态程序的过程。

519021910594

陶昱丞