PA1 实验报告　　高歌 221240075

实验进度：全部完成

PA1.1：

如果没有寄存器，计算机还可以工作吗？如果可以，这会对硬件提供的编程模型有什么影响呢?

寄存器用于高速存储和访问数据，它们位于 CPU 内部，因此访问速度非常快。如果没有寄存器，计算机将不得不依赖于较慢的内存来存储和检索数据，这会显著降低性能，计算机不能正常工作。虽然理论上没有寄存器的计算机是可能的，但这种计算机将在实际应用中受到严重的限制，因为性能、编程复杂性和效率方面的问题。因此，现代计算机体系结构依赖于寄存器来提供高性能和有效的编程模型。

为什么全是函数?

全部使用函数引用，更加方便理解程序逻辑，代码编写更加优雅。

究竟要执行多久?

传入的类型是 uint64_t，即无符号数，传入-1 后会自动转为 uint64 的最大值，并且逐级减一，到 0 时停止

潜在的威胁：

不属于未定义行为。传入-1 对于无符号数来说即是最大的数，同上问，只要执行 18446744073709551615 步结束便可正常完成程序。

优雅的退出：

打开代码，比较 cmd_q() 与 is_exit_status_bad()函数，发现 cmd_q 只 return -1 便执行结束了，因此只需修改 nemu_state.state = NEMU_QUIT 即可优雅退出。

为什么要有 nemu trap 与 monitor?

nemu_trap： 这可能是指 NEMU 模拟器中的一种机制或函数，用于处理和模拟异常、中断或陷阱。在模拟实际硬件时，计算机系统需要处理不同类型的异常，如除以零、页面错误、系统调用等。它允许模拟器以与实际硬件相似的方式响应和处理异常情况。

monitor：是用户与模拟器进行交互的方式，通常用于加载程序、运行程序、监视寄存器和内存状态，以及调试程序。他能使用户能查看控制模拟器状态。

PA1.2

为什么 printf 要换行?

1. 方便查看
2. 不换行可能导致输出留在输出缓冲区中，看不到已经执行的输出。

必答题：

画出计算 1+2+...+100 的程序的状态机

(0,x,x)->(1,0,x)->(2,0,0)->(3,0,1)->(4,1,1)->(3,1,2)->(4,3,2)->(3,3,3)->(4,6,3)->(3,6,4)->(4,10,4)->.....->(3,4851,99)->(4,4950,99)->(3,4950,100)->(4,5050,100)->quit.

那么这个学期下来，简易调试器可以帮助你节省多少调试的时间?

500*0.9*30*20/3600=75h.

RTFM:

What are the instruction formats for riscv32?

riscv32 有四种指令格式 (R/I/S/U)。若考虑立即数，S 类型-> B 类型，U 类型-> J 类型。

What is the behavior of the LUI instruction?

LUI 指令为 U 类型，将高 20 位编码的立即数左移 12 位后，装入目的寄存器。

What is the structure of the mstatus register?

The `mstatus` register is an MXLEN-bit read/write register formatted as shown in Figure 3.6 for RV32 and Figure 3.7 for RV64. The `mstatus` register keeps track of and controls the hart's current operating state. A restricted view of `mstatus` appears as the `sstatus` register in the S-level ISA.

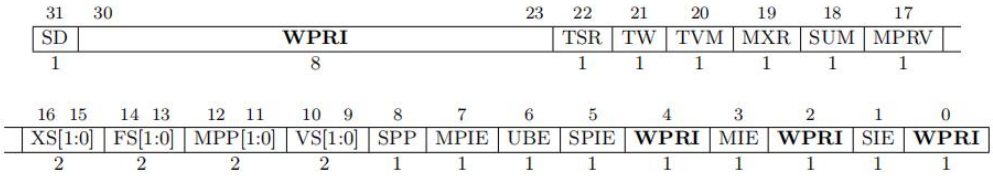

Figure 3.6: Machine-mode status register (`mstatus`) for RV32.

统计代码行数：

before edit:23799

after edit: 24171

. 请解释 gcc 中的-Wall 和-Werror 有什么作用？为什么要使用-Wall 与 -Werror?

这里-Wall 和-Werror 的目的是启用所有的警告消息，并将警告消息转换为错误消息，这样便于调试和减少 bug 的发生