计算机系统基础

2022秋季学期

PA1

院系：人工智能学院

姓名：胡涛

学号：211300046

班级：ics2022-AI学院

邮箱：1011393914@qq.com

实验进度：

实验中完成了单步执行、打印寄存器扫描内存、表达式求值、监视点等要求的所有内容。

必答题：

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)→.....(k+2,r1,k-1)→(3,r1,k)→...→(102,4950,99)→(3,4950,100)→(,5050,100)

`PC=0 :mov r1 0;PC=1 :mov r2 0;PC=2 :mov r2 1;PC=3 :addi r1 r1 r2;PC=4 :mov r2 2`

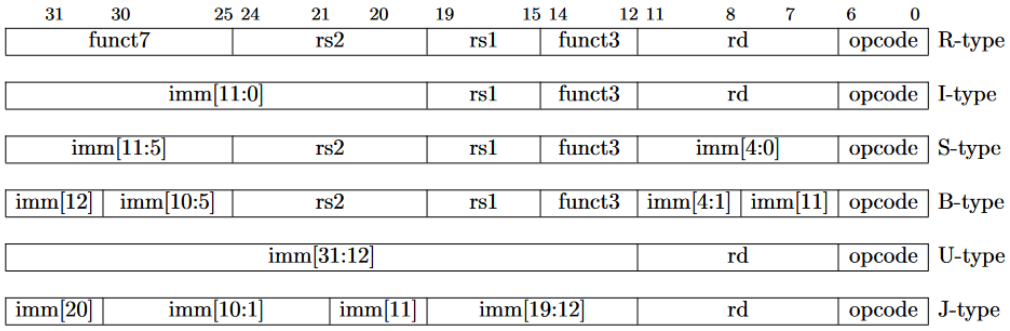
`PC=k+2 :mov r2 k k⩾4`

理解基础设施：

使用gdb在调试上花费4500分钟约187.5小时，使用简易调试器将可以少花费大概3000分钟约50小时，花费的时间确实是大大的节约了。从这个简单的例子中可以直观的看出简易调试器可以极大的节约时间也更方便后续的pa的完成。

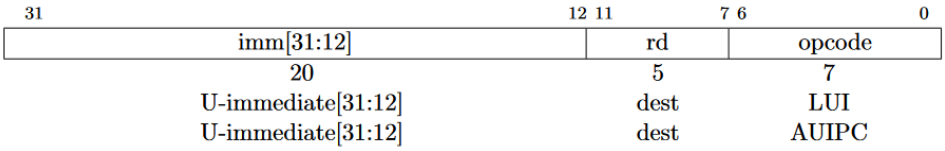
RTFM:

- riscv32有哪几种指令格式?



  阅读范围在riscv-spec.pdf的P16页.

- LUI指令的行为是什么?



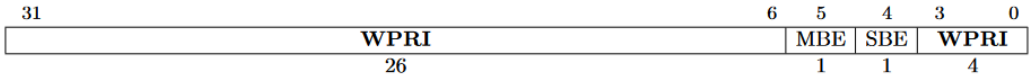
LUI (load upper immediate) is used to build 32-bit constants and uses the U-type format. LUI places the 32-bit U-immediate value into the destination register *rd*, filling in the lowest 12 bits with zeros.

  LUI用来创建一个32位u型常数。使用时LUI将imm值赋给ra的高20位同时给低十二位补零。

  阅读范围在riscv-spec.pdf的P19页.

- mstatus寄存器的结构是怎么样的?

  mstatus寄存器是一个32-bit的读/写寄存器，结构如下。



  阅读范围在riscv-privileged.pdf的P20页.

- shell命令 完成PA1的内容之后， `nemu/` 目录下的所有.c和.h和文件总共有多少行代码? 你是使用什么命令得到这个结果的? 和框架代码相比，你在PA1中编写了多少行代码? (Hint: 目前 `pa0` 分支中记录的正好是做PA1之前的状态，思考一下应该如何回到"过去"?) 你可以把这条命令写入 `Makefile` 中，随着实验进度的推进，你可以很方便地统计工程的代码行数，例如敲入 `make count` 就会自动运行统计代码行数的命令. 再来个难一点的，除去空行之外， `nemu/` 目录下的所有 `.c` 和 `.h` 文件总共有多少行代码?

  PA1中代码计算了空行的有23951行排除空行的是20780行，框架代码中计算了空行的有23468行排除空行的是20338行。使用find . -name "*.c" -o -name " .h"|xargs cat|wc -l命令在nemu目录下得到带空行的结果。使用find . -name "*.c" -o -name " .h"|xargs cat|grep -v ^$|wc -l得到无空行的结果。与框架代码相比PA1增加了483行代码。
  在makefile中添加

```
COUNT_LINES := $(shell  find . -name "*.h" -o -name "*.c" | xargs grep -v "^$$" | wc -l)

count:
	@echo There are $(COUNT_LINES) lines in nemu
```

  计算不带空行的.c和.h代码行数。执行效果如下:

```
hutao@hutao-virtual-machine:~/ics2022/nemu$ make count
There are 20780 lines in nemu
```

- RTFM 打开 `nemu/scripters/build.mk` 文件，你会在 `CFLAGS` 变量中看到gcc的一些编译选项. 请解释gcc中的 `-Wall` 和 `-Werror` 有什么作用? 为什么要使用 `-Wall` 和 `-Werror` ?

  -Wall是为了打开gcc的所有警告。而-Werror，它要求gcc将所有的警告当成错误进行处理。使用了-Wall和-Werror保证了make的时候一旦出现警告则使得程序认作错误而停止，使得我们可以尽可能地避免错误，减少了程序出错的可能性。

思考

在使用cmd_x,cmd_p等指令获取参数的时候中使用strtok错误导致可能存在‘ ’的expr表达式中断被取出而不能完整现实expr表达式，同时在只运行TODO()的时候运行完毕会使得程序异常退出，原因在于TODO中会运行定义的panic函数直接assert(0)。计算完成表达式的值后一定要清空tokens数组否则在下一次表达式计算的时候则可能会使用到上一次计算表达式存储在tokens中的token或者使得tokens下的str名称发生改变。