

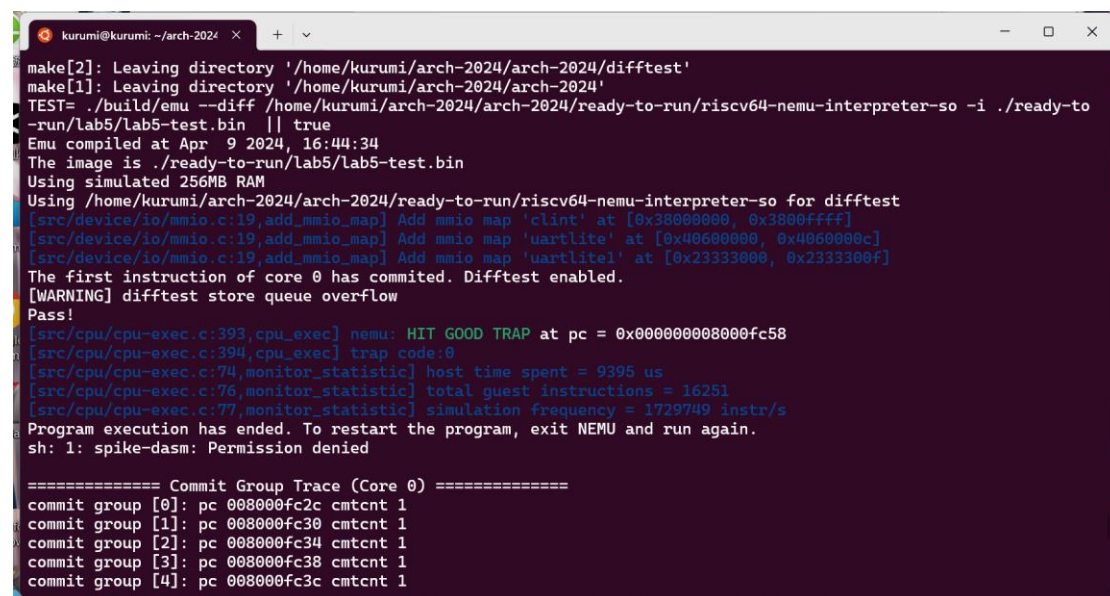
实验名称: CPU lab5

实验人: 谢志康

学号: 22307110187

实验时间: 2024. 4. 9

Pass 截图



```
kurumi@kurumi: ~/arch-2024
make[2]: Leaving directory '/home/kurumi/arch-2024/arch-2024/diffest'
make[1]: Leaving directory '/home/kurumi/arch-2024/arch-2024'
TEST= ./build/emu --diff /home/kurumi/arch-2024/arch-2024/ready-to-run/riscv64-nemu-interpretor-so -i ./ready-to-run/lab5/lab5-test.bin || true
Emu compiled at Apr 9 2024, 16:44:34
The image is ./ready-to-run/lab5/lab5-test.bin
Using simulated 256MB RAM
Using /home/kurumi/arch-2024/arch-2024/ready-to-run/riscv64-nemu-interpretor-so for diffest
[src/device/io/mmio.c:19,add_mmio_map] Add mmio map 'clint' at [0x38000000, 0x3800ffff]
[src/device/io/mmio.c:19,add_mmio_map] Add mmio map 'uartlite' at [0x40600000, 0x4060000c]
[src/device/io/mmio.c:19,add_mmio_map] Add mmio map 'uartlite1' at [0x23333000, 0x2333300f]
The first instruction of core 0 has committed. Diffest enabled.
[WARNING] diffest store queue overflow
Pass!
[src/cpu/cpu-exec.c:393,cpu_exec] nemu: HIT GOOD TRAP at pc = 0x000000000000fc58
[src/cpu/cpu-exec.c:394,cpu_exec] trap code:0
[src/cpu/cpu-exec.c:74,monitor_statistic] host time spent = 9395 us
[src/cpu/cpu-exec.c:76,monitor_statistic] total guest instructions = 16251
[src/cpu/cpu-exec.c:77,monitor_statistic] simulation frequency = 1729749 instr/s
Program execution has ended. To restart the program, exit NEMU and run again.
sh: 1: spike-dasm: Permission denied

===== Commit Group Trace (Core 0) =====
commit group [0]: pc 008000fc2c cmtcnt 1
commit group [1]: pc 008000fc30 cmtcnt 1
commit group [2]: pc 008000fc34 cmtcnt 1
commit group [3]: pc 008000fc38 cmtcnt 1
commit group [4]: pc 008000fc3c cmtcnt 1
```

计算模块: 本次加了 divw 和 mulw 等 word 指令, 其实和前面做的 div, mul 差不多。根据多周期原理和指令手册对应写即可。Divw 等 4 个指令, 我开了个新的 divw 模块, 就是大体上将 div 模块分 64 个周期算改为 32 个周期算, 没算完把 cpu 给 stall 住即可。Mulw 模块直接在 mul 模块后面加, 把结果低 32 位符号扩展即可。

遇到几个问题: 1. 特判问题, 有的除 0, 除 64' bf 这种, 各种奇怪的特殊情况, 就是一个一个加特判解决的, 具体我代码中特判部分有部分有标出在哪一条指令导致我加的特判。2. 模块冲突问题, 在我 lab5 跑到很后面的时候还突然 div 错了一个, 很难理解, 在谭学长的帮助下发现是前面 count 没有置 0。就是在 div 上两条有个 divw, count 要到 32 才算算完嘛 (分 32 个周期), 然后我原本 div 模块 count 不为 0 就开始算 div 中的数了, 导致到 div 的时候 count 并不是为 0, 就是没有完整算 64 个周期。后面加上不是 div 四个指令的时候 div 模块中 count<=0, divw 同理。3. 有点难理解的是阻塞与非阻塞赋值, mul 模块我用非阻塞过不了, 阻塞就对了。助教说想清楚组合逻辑还是上升沿逻辑。

内存模块: 和前面访存操作基本同理, 了解 strobe 的用法即可, 根据地址的低三位设置 offset 将 data 写到正确地方。Load 操作和 lab2 那个同理, 没搞完 (data_ok) 把 cpu 给 stall 住即可。