

Lab1报告

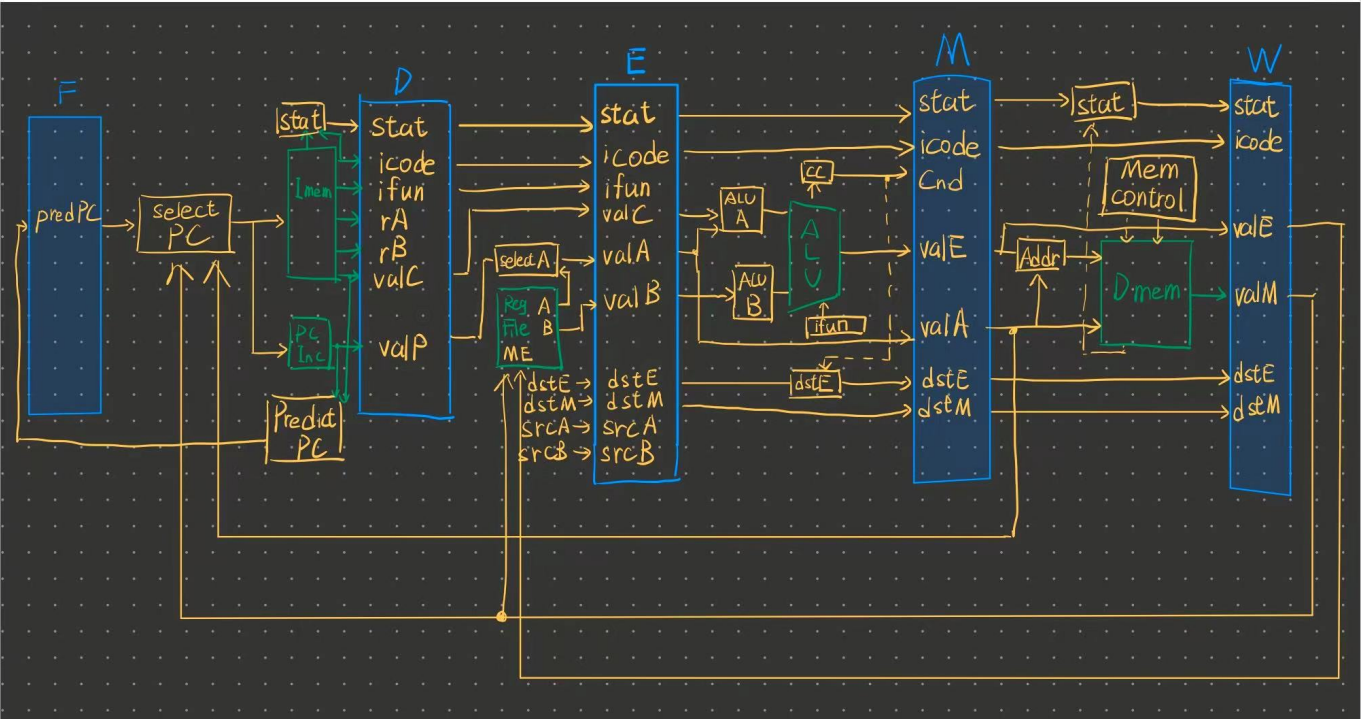
季雨昊 23300240010

实验目标

- 1.搭建实验环境
- 2.构建五级流水线CPU架构，CPU需要支持以下指令并通过lab1测试：
算术运算与逻辑运算：
addi, xori, ori, andi, add, sub, and, or, xor
扩展指令：
addiw, addw, subw

实验过程

1.电路设计图的绘制：



2.设计模块的介绍

- 1. core.sv : cpu顶层的实例化模块，实例化了取指到写回的各个模块，包括每个模块之间的流水线寄存器。
- 2. 流水线寄存器reg_FD.sv、reg_DE.sv、reg_EM.sv、reg_MW.sv：现在的功能只有简单的寄存器功能，捕获前一个阶段的数据，并在下一个时钟周期将其传递给下一个阶段，从而保持流水线的连续性和正确性。
- 3. 取指阶段fetch.sv、pcselect.sv：从指令存储器中取出指令，并将其送入指令寄存器。现在的指令都只需要pc+4就能更新到下一条指令，所以pcselect模块只需要简单地将pc+4送入指令寄存器即可。
- 4. 译码阶段decode.sv、decoder.sv：decode从指令中提取立即数和寄存器地址，并且示例化了decoder。decoder利用case条件分支语句从raw_instr中译码出不同的指令。
- 5. 执行阶段execute.sv、alu.sv：execute提取译码阶段输出的dataD信息，提取出alu的输入信号并实例化alu。alu根据指令进行具体的运算，如add\sub\and\or\xor等。execute还负责进行addw等指令的截断和符号位扩展。
- 6. 访存阶段memory.sv:由于现在的指令还不需要访存，故只进行了接口的设计，没有具体的内容。
- 7. 写回阶段writeback.sv：writeback调控regfile的写操作，把运算结构写回目标寄存器。

- 寄存器文件regfile.v : 负责寄存器的读写操作, 参照wiki用regs_nxt进行寄存器状态的更新, 便于测试平台的提交。

实验结果

hit good trap。能顺利执行测试指令。

```
Let REF run one more instruction.
[src/cpu/cpu-exec.c:393,cpu_exec] nemu: HIT GOOD TRAP at pc = 0x0000000080010004
[src/cpu/cpu-exec.c:394,cpu_exec] trap code:0
[src/cpu/cpu-exec.c:74,monitor_statistic] host time spent = 6031 us
[src/cpu/cpu-exec.c:76,monitor_statistic] total guest instructions = 16385
[src/cpu/cpu-exec.c:77,monitor_statistic] simulation frequency = 2716796 instr/s
sh: 1: spike-dasm: not found

===== Commit Group Trace (Core 0) =====
commit group [0]: pc 0080010000 cmtcnt 1 <--
commit group [1]: pc 008000ffc4 cmtcnt 1
commit group [2]: pc 008000ffc8 cmtcnt 1
commit group [3]: pc 008000ffcc cmtcnt 1
commit group [4]: pc 008000ffd0 cmtcnt 1
commit group [5]: pc 008000ffd4 cmtcnt 1
commit group [6]: pc 008000ffd8 cmtcnt 1
commit group [7]: pc 008000ffdc cmtcnt 1
commit group [8]: pc 008000ffe0 cmtcnt 1
commit group [9]: pc 008000ffe4 cmtcnt 1
commit group [a]: pc 008000ffe8 cmtcnt 1
commit group [b]: pc 008000ffec cmtcnt 1
commit group [c]: pc 008000fff0 cmtcnt 1
commit group [d]: pc 008000fff4 cmtcnt 1
commit group [e]: pc 008000fff8 cmtcnt 1
commit group [f]: pc 008000fffc cmtcnt 1
```