

实验 5-3 报告

2016K8009909006

刘杰

一、实验任务（10%）

使用已实现的类 sram 接口到 AXI 接口的转接桥，为 myCPU 添加 AXI 接口，并运行功能测试通过，上板时要求“随意切换拨码开关按复位”，CPU 均通过 94 个功能点的测试。

二、实验设计（30%）

已实现的 CPU 访存接口是 SRAM 接口，如果想要复用 AXI 接口转接桥，首先要将 SRAM 接口改为类 SRAM 接口。以数据端为例，

```
assign data_req = data_sram_en;
assign data_wr =(data_sram_wen == 4'd0)? 1'd1:1'd0;
assign data_size = {2{data_sram_wen == 4'b1111}}&2'd2
                  |{2{data_sram_wen == 4'b0011|data_sram_wen == 4'b1100}}&2'd1;
assign data_addr = data_sram_addr;
assign data_wdata = data_sram_wdata;
assign data_sram_rdata = data_rdata;
```

将 CPU 接口修改后，就可以访问 AXI 接口的存储设备。由于 AXI 接口的存储设备接受访问地址，返回访问数据的延迟不定。所以在取指请求发出后，需要阻塞取指级直到 inst_data_ok 拉高，同理在读数据和写数据请求发出后，需要阻塞取指级到访存级的流水线直到 data_data_ok 拉高。此外，在取指请求发出后，在 AXI 接口接受请求前需保持请求不变，只有当 inst_addr_ok 或者 data_addr_ok 拉高时，AXI 才成功接受请求。

由于上次 AXI 转接桥设计时，同一时间只能处理一个事务，也就是不能同时使用读写通道。所以 CPU 的效率明显偏低。正在考虑通过把状态机分拆成 2 个，分别表示读写通道的状态来实现同时处理读写事务。

三、实验过程（60%）

（一）实验流水账

2018.12.10 15:00-16:25 阅读讲义完成代码编写

2018.12.10 16:40-17:30 仿真测试，读指令出错，debug 中

2018.12.10 18:00-18:30 找出并修改前一个 bug 后仿真，lb 指令出错，继续 debug

2018.12.10 18:50-19:30 修改 bug，继续仿真，例外出错，debug

2018.12.10 19:30-20:25 修改 bug，继续仿真，仿真测试通过，上板出错，单色灯开始就全亮

2018.12.10 20:30-22:00 debug，查找 vivado log，改 warning，多次测试失败

2018.12.10 22:10-23:00 阅读 piaaza 的提问，得到提示，找出问题后解决，上板通过

（二）错误记录

1、错误 1

（1）错误现象

取指时取回的指令有误。

（2）分析定位过程

对比 ref，发现相同 PC 时写回寄存器值以及写回目标寄存器不正确，根据经验，推测为取指错误，找到此时指令寄存器的值，与 test.s 进行比对，发现不匹配。

（3）错误原因

因为使用 wire 型信号作为取指地址，发起指令请求后，在请求尚未被 AXI 通道接受前，取指的地址就发生改变，导致正确的地址没有接受到。

（4）修正效果

在发起请求时，同时用一个寄存器存储请求的地址，以改寄存器作为请求地址输出给 AXI 接口。

（5）归纳总结（可选）

在一个请求置起后，如果这个请求未被响应，就不能修改请求。

2、错误 2

（1）错误现象

LB 指令存入的数据有误。

（2）分析定位过程

根据控制台错误信息找出错误指令为 LB，对比 ref，仅写入寄存器的数据发生错误，而且仅一字节的数据不相同，推测控制写入字节的信号有问题，查看相应信号，发现正确的写选通信号要慢一拍。

（3）错误原因

由于阻塞，读取数据返回时，相应的指令仍然在访存级，而不是之前实验中实现的写回级，但是此时控制信号却使用写回级信号，导致错误。

（4）修正效果

把控制信号从写回级改为访存级，或者将读取的数据传入写回级。

3、错误 3

（1）错误现象

例外处理时，mfc0 指令查看 epc 寄存器的值有误。

(2) 分析定位过程

根据控制台错误信息找出错误指令为 mfc0，对比 ref，发现写会寄存器的值比正确值大 0x4，且此时读目标寄存器为 epc，推测为例外指令在延迟槽，但未成功标记。

(3) 错误原因

由于取指级可能发生阻塞，所以当延迟槽中指令返回时，译码级已经不是跳转指令，而我判断是否在例外是否是在延迟槽是根据译码级是否为分支指令，所以发生错误。

(4) 修正效果

当指令到达译码级时，根据是否为分支指令，把取指级指令在延迟槽的信号寄存器拉高，然后随流水线传递下去即可。

4、错误 4

(1) 错误现象

仿真测试通过，但是上板复位后单色灯就全亮。

(2) 分析定位过程

根据 log 查找综合中的 warning，把能修改的更正后仿真通过，再次上板依旧失败。翻阅 piaaza 的提问，发现由同学提到上板和仿真时 AXI 返回数据的行为不一样，上板时只有 data_ok 拉高时数据才有效，其他时候都为 0。

(3) 错误原因

由于 IR 寄存器更新不是在 inst_data_ok 拉高后的第一个时钟上升沿，所以在上板时当 IR 更新时，inst_rdata 已经为 0。

(4) 修正效果

当 inst_data_ok 拉高时，用寄存器先暂存指令，IR 每次更新都把这个暂存指令的寄存器作为来源。重新测试，上板通过。

四、实验总结（可选）

这次总线实验结束，体系结构实验课也基本接近尾声，虽然做实验调 bug 有时候耗时间，很累，但是坚持下来发现这个学期的实验对个人能力的提升非常显著。在这里感谢老师的帮助和付出。