

（深圳）

实验作业

开课学期： 2022春季

课程名称：计算机组成原理（实验）

实验名称： 直接映射Cache设计

实验性质： 综合设计型

实验学时： 4 地点： T2615

学生班级： 计算机6班

学生学号： 200110619

学生姓名： 梁鑫嵘

作业成绩：

实验与创新实践教育中心制

2022年5月

|  |
| --- |
| 1. Cache模块设计 |
| （画出读、写的状态转移图，并描述状态之间的转移关系和转移条件、以及每个状态需要完成什么操作。）  有限状态自动机：   1. READY：待机准备状态    1. r/waddr\_to\_mem <= 0    2. r/wreq\_to\_mem <= 0 2. TAG\_CHECK：当前周期检查是否命中Cache    1. waddr\_to\_mem <= addr\_from\_cpu; 写入写地址    2. wreq\_to\_mem <= 0; 检查的时候不能确定是否写缺失，所以不会写    3. raddr\_to\_mem <= addr\_from\_cpu; 写入读地址    4. rreq\_to\_mem <= miss && rreq\_from\_cpu && !rvalid\_from\_mem; 产生缺失 3. REFILL：读内存到Cache或者写内容到内存    1. raddr\_to\_mem <= addr\_from\_cpu; 写入读地址    2. rreq\_to\_mem <= rreq\_from\_cpu && !rvalid\_from\_mem; 读完适时拉低rreq信号    3. waddr\_to\_mem <= addr\_from\_cpu; 写入写地址    4. wdata\_to\_mem <= wdata\_from\_cpu; 是否写入    5. wreq\_to\_mem <= !miss && wreq\_from\_cpu; 填入Cache后拉低wreq |
| 1. **调试报告** |
| （仿真截图及时序分析，要求包含读命中、读缺失、写命中及写缺失共四种情况的分析，且每种情况需列举2个测试用例进行分析。）  读命中：   1. 每次读命中在请求的下一周期就能返回读到的数据 2. 第一次读命中：读地址为0x0101 3. State为：READY→TAG\_CHECK→READY 4. 这三次读取命中同一个Cache Line，Tag为0x01 5. Offset依地址变化为0x1,0x2,0x3，故依次返回Cache Line的第2,3,4字节   读缺失：   1. 读缺失的时候在请求拉高后18周期才从内存取回数据 2. 取到数据的那一周期写入Cache，取到数据为0xcdcccbca，写入Cache Line为0x21cdcccbca 3. 取到数据之后的下一周期将Cache内容返回CPU 4. State变化为：READY→TAG\_CHECK→REFILL→TAG\_CHECK→READY   写命中：   1. 写访存情况下，CPU写信号拉高后的下一周期就能将是否写命中返回给CPU 2. 返回是否写命中后的下一周期拉高内存写入信号并写入内存指定地址 3. 总共需要消耗3周期时间（假设内存1周期写入完毕） 4. State：READY→TAG\_CHECK→REFILL→TAG\_CHECK→READY 5. 因为写入后的下一周期立刻开始读取，所以TAG\_CHECK状态被复用而没有回到READY   写缺失：   1. 拉高写信号后下一周期返回了写缺失 2. 写缺失则什么也不做，不更新Cache也不更新内存内容 3. State：READY→TAG\_CHECK→READY |