

实验作业

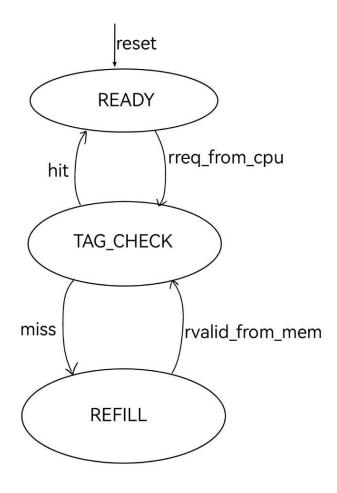
开课学期:	2022 春季
课程名称:	计算机组成原理(实验)
实验名称:	直接映射 Cache 设计
实验性质:	综合设计型
实验学时:	4 地点: T2506
学生班级:	
学生学号:	200111205
学生姓名:	 李聪
作业成绩:	

实验与创新实践教育中心制 2022年5月

一、 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。)

读的状态转移图:

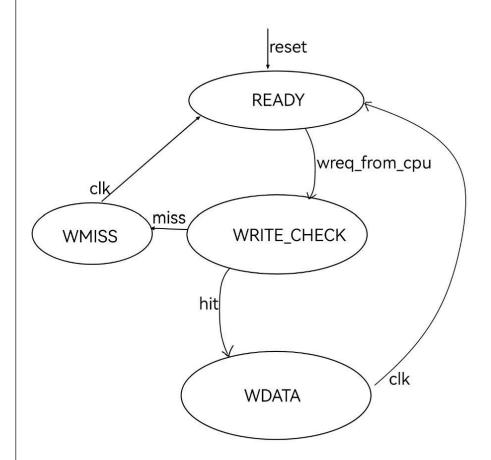


READY 状态: Cache 处在就绪状态

TAG_CHECK 状态: 检查是否命中 Cache

REFILL 状态:从内存中取出需要的块,加上标签与有效位标记后存入 Cache 的存储体。

写状态转移图:



READY 状态: Cache 处在就绪状态

WRITE_CHECK 状态: 检查是否命中 Cache

WDATA 状态:写命中,向内存和 Cache 写入数据。

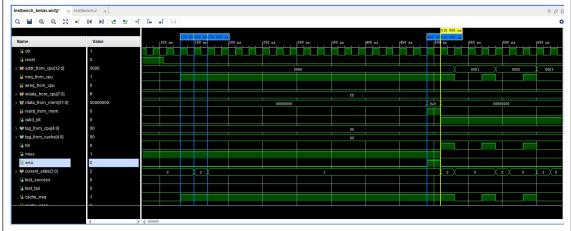
WMISS 状态:写缺失,hit 保持低电平,不修改内存。

二、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。)

Current_state 的表示: READY = 0000、TAG_CHECK = 0010、 REFILL= 0001、 WRITE_CHECK = 0011、WMISS= 0100、WDATA= 0101;

读缺失例 1:



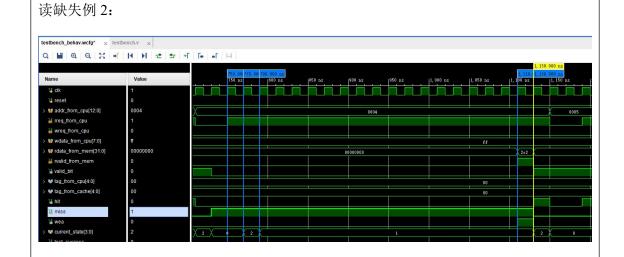
130ns 时,rreq_from_cpu 变为 1,cpu 发出读请求

150ns, current_state 变为 0010, 进入 TAG_CHECK 状态, 由于此时 valid_bit 值为 0, 故读缺失, 符合预期。

170ns, current state 变为 0001, 进入 REFILL 状态。

490ns,主存读取完成标记 rvalid_from_mem 为 1,说明主存读取完毕,于是将主存读取的数据加上标记和有效位后写入 cache, wea 变为高电平,符合预期。

510ns, cache 更新完毕,进入 TAG_CHECK 状态,此时 tag_from_cpu=tag_from_cache,且 valid_bit=1,故命中,符合预期。



750ns 时,rreq_from_cpu 变为 1,cpu 发出读请求

770ns, current_state 变为 0010, 进入 TAG_CHECK 状态,由于此时 valid_bit 值为 0,故 读缺失,符合预期。

790ns, current state 变为 0001, 进入 REFILL 状态。

1100ns,主存读取完成标记 rvalid_from_mem 为 1,说明主存读取完毕,于是将主存读取的数据加上标记和有效位后写入 cache, wea 变为高电平,符合预期。

1300ns, cache 更新完毕, current_state 变为 0001, 进入 TAG_CHECK 状态, 此时 tag_from_cpu=tag_from_cache, 且 valid_bit=1, 故命中,符合预期。

读命中例1 Name Value 550 ns ¼ clk ¼ reset 0 > W addr_from_cpu[12:0] 0002 If rreq_from_cpu 0 wreq_from_cpu > Www.cpu[7:0] 00000000 > W rdata_from_mem[31:0] If rvalid_from_mem 00 > W tag_from_cpu[4:0] 00 > W tag_from_cache[4:0] valid_bit le hit i miss wea ₩ wreq_to_mem 0000 > W waddr_to_mem[12:0] wdata_to_mem[7:0] 00 0010 current_state[3:0] 0010 X 0000 > W next_state[3:0] 0010 > W current_state[3:0] READ_CACHE RES REA READ_CACHE RES

RES_COMPARE

570ns 时,rreq_from_cpu 变为 1,cpu 发出读请求

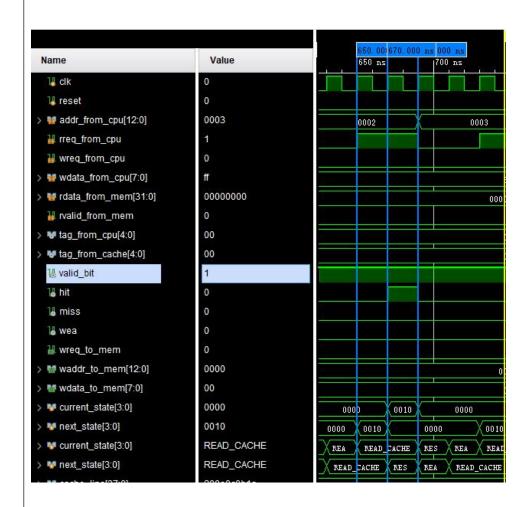
> W next_state[3:0]

590ns, current_state 变为 0010, 进入 TAG_CHECK 状态, tag_from_cpu=tag_from_cache, 且 valid bit=1, 故读命中, hit 变为 1, 符合预期。

REA

READ_CACHE | RES | REA

读命中例 2:



650ns 时, rreq from cpu 变为 1, cpu 发出读请求

670ns, current_state 变为 0010, 进入 TAG_CHECK 状态, tag_from_cpu=tag_from_cache, 且 valid_bit=1,故读命中,hit 变为 1,符合预期。

写缺失例 1: x testbench.v x testbench_behav.wcfg* Q 🖺 Q 🔀 📲 H N 🚾 🖭 + F 🕞 🛶 Name Value 1, 392, 750 ns 1, 392, 650 ns ₩ clk ™ reset 0000 addr_from_cpu[12:0] 1ffe 1fff # rreq_from_cpu wreq_from_cpu wdata_from_cpu[7:0] rdata_from_mem[31:0] 00000000 000000 rvalid_from_mem 00 ₩ tag_from_cpu[4:0] ₩ tag_from_cache[4:0] a valid_bit wea wreg to mem 0000 waddr_to_mem[12:0] 00 wdata_to_mem[7:0] 0100 current_state[3:0] 0000 (0010) 0011 0100 0000 0010 0000 0011 0100 0000 0010 0 w next_state[3:0] 0010 0000 w current_state[3:0] FLUSH_CLINE READ_CACHE RES RD_ WRI w next_state[3:0] FLUSH_CLINE REA RES RD_ WRI

1392690ns, wreq_from_cpu 变为 1, cpu 发出写请求

1392710ns, current_state 变为 0011, 进入 WREIT_CHECK 状态, 由于 tag_from_cpu=00, tag from cache=1f, 二者不相等, 故写缺失, hit=0, miss=1, 符合预期

1392730ns, current_state 变为 0100, 进入 WMISS 状态, hit 保持低电平, wreq_to_mem = 0, 不写主存, 符合预期

写缺失例 2:



1393250ns, wreq from cpu 变为 1, cpu 发出写请求

1393270ns, current_state 变为 0011, 进入 WREIT_CHECK 状态, 由于 tag_from_cpu=00, tag from cache=1f, 二者不相等, 故写缺失, hit=0, miss=1, 符合预期

1393290ns, current_state 变为 0100, 进入 WMISS 状态, hit 保持低电平, wreq_to_mem = 0, 不写主存, 符合预期

Name Value ¼ clk ₩ reset addr_from_cpu[12:0] 1f00 1f04 If rreq_from_cpu wreq_from_cpu wdata_from_cpu[7:0] ff rdata_from_mem[31:0] 00000000 00000000 In rvalid_from_mem ₩ tag_from_cpu[4:0] ₩ tag_from_cache[4:0] 00 avalid_bit le hit M miss ₩ wreq_to_mem 0000 waddr_to_mem[12:0] 1f04 00 wdata_to_mem[7:0] 00 ff 0000 current_state[3:0] 0011 0101 0000 0010 0000 w next_state[3:0] 0000 0011 0101 0000 0010 current_state[3:0] IDLE READ_C RES WRI WR_ REC w next_state[3:0] READ_TRACE RES WRI

写命中例 1:

1393810ns, wreq_from_cpu 变为 1, cpu 发出写请求

1393830ns, current_state 变为 0011, 进入 WREIT_CHECK 状态,由于 tag_from_cpu=1f, tag_from_cache=1f, 二者相等,且 valid_bit=1,故写命中,hit=1,符合预期

1393850ns, current_state 变为 0101, 进入 WDATA 状态, wreq_to_mem 变为 1, 向主存写入数据,同时 wea 变为 1,向 cache 写入数据,符合预期

写命中例 2: Value Name 1, 394, 650 ns ₩ clk 🗸 reset 0008 > W addr_from_cpu[12:0] 1f04 1f08 If rreq_from_cpu 0 wreq_from_cpu > Www.cpu[7:0] > W rdata_from_mem[31:0] 00000000 00000000 Invalid_from_mem 00 > W tag_from_cpu[4:0] 00 > W tag_from_cache[4:0] 1f avalid_bit 0 14 hit le miss wea wreq_to_mem > 1 waddr_to_mem[12:0] 0000 1f08 00 > **1** wdata_to_mem[7:0] 00

1394690ns, wreq_from_cpu 变为 1, cpu 发出写请求

0001

0001

FLUSH_CLINE

FLUSH_CLINE

> W current_state[3:0]

> W current_state[3:0]

next_state[3:0]

> W next_state[3:0]

1394710ns, current_state 变为 0011, 进入 WREIT_CHECK 状态,由于 tag_from_cpu=1f, tag_from_cache=1f, 二者相等,且 valid_bit=1,故写命中,hit=1,符合预期

0010

0000

RES X WRI

READ RES

0011

0101

WR_

REC

0011

WRI

0101

0000

REC

0000 | 0010

0010

1393730ns, current_state 变为 0101, 进入 WDATA 状态, wreq_to_mem 变为 1, 向主存写入数据, 同时 wea 变为 1, 向 cache 写入数据,符合预期