

基于 FPGA 的六级流水线 MIPS 处理器设计

孙巧稚, 施慧彬

(南京航空航天大学 计算机科学与技术学院, 江苏 南京 210016)

摘 要: 设计出了一种兼容 MIPS 指令集的 32 位六级流水线嵌入式处理器. 六级流水线的划分平衡了各个阶段的任务. 并详细介绍了数据冲突和控制冲突的解决方法. 该处理器使用 FPGA 实现, 在 DE2 芯片上的运行时钟频率可达 81.7 MHz. 最后给出了设计的综合结果, 并对该设计进行了软件仿真和硬件验证.

关键词: MIPS; 流水线处理器; 冲突; FPGA

中图分类号: TN402

文献标识码: A

文章编号: 1000-7180(2015)04-0031-04

Design of a Six-stage Pipelined MIPS Processor Based on FPGA

SUN Qiao-zhi, SHI Hui-bin

(Department of Computer Science and Technology, Nanjing University of
Aeronautics and Astronautics, Nanjing 210016, China)

Abstract: A 32-bit embedded six-stage pipelined processor is designed in this paper, which is compatible with MIPS instruction set. The six stages make the task of each stage balanced. The solutions to data hazards and control hazards in detail are given out. The processor is implemented in FPGA, and its clock frequency can be up to 81.7 MHz in DE2 development board. The comprehensive results of the design are presented, and the software simulation and hardware verification results prove the correctness of the design.

Key words: MIPS; pipelined processor; hazards; FPGA

1 引言

RISC 概念的广泛应用, 促成了 MIPS 技术的建立^[1], 并使 MIPS 处理器在工业控制领域得到了广泛应用. 本文在 MIPS 体系结构的基础上, 利用 FPGA 设计了一种兼容 MIPS32 指令集的具有六级流水线结构的高性能嵌入式处理器.

2 设计与实现

2.1 六级流水线的设计

经典的 MIPS 流水线处理器是五级流水线, 把数据路径分为取指、译码、执行、访存和写回五个阶段. 经典五级流水线的译码级不仅要读取写入寄存器组还要加入比较器等器件导致周期过长. 执行级由于 ALU 前端要放置多路选择器, 也使该阶段的周期较长. 由于流水线处理器的周期时间是取五个

阶段中周期的最大值, 进而可以通过缩短五级中周期较长的阶段来实现处理器周期的缩短. 本文设计出一种六级的流水线, 即在五级流水线的译码级和执行级中间加了一级选择级. 把译码级的比较器和执行级的多路选择器放到这一级, 同时缩短了这两级的周期, 进而提高整个处理器的主频.

本文设计的处理器的结构图如图 1 所示, 处理器分为六个阶段. (1) 在取指级, 处理器从指令存储器中读取指令. (2) 在译码级, 处理器从寄存器文件中读取源操作数并对指令译码以产生控制信号. (3) 在选择级, 对要进入执行阶段 ALU 两个输入端口的运算数预选择来化解数据冲突, 并且计算出转移指令的目标地址. (4) 在执行级, 处理器使用 ALU 进行运算. (5) 在存储器访问级, 处理器向数据存储寄存器中读取或写入数据. (6) 在写回级, 如果需要, 处理器将结果写入到寄存器.

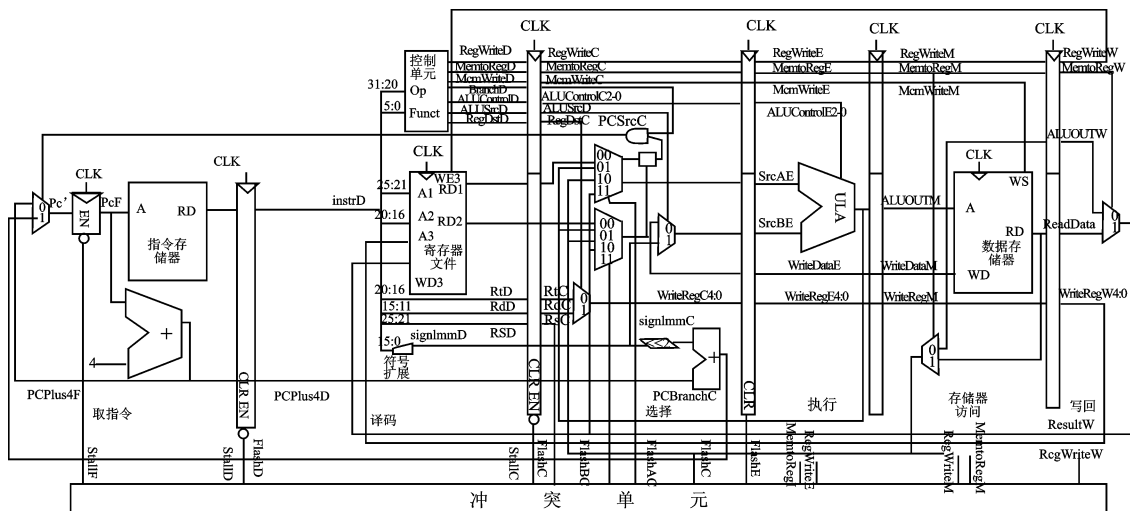


图 1 六级流水线结构图

2.2 流水线中冲突的解决方法

流水线处理器的核心问题是化解冲突. 因此本文将介绍六级流水线处理器设计中冲突产生的原因和解决方法. 在流水线系统中, 许多条指令同时执行, 当一条指令执行需要还没有结束指令的操作结果时, 将发生冲突^[2]. 冲突可以分为数据冲突和控制冲突: 当一条指令尝试读取前条指令还没写回寄存器的值时将发生数据冲突; 在取指令时还未确定下一条应取指令的地址将发生控制冲突^[3-4].

2.3 数据冲突

图 2 中指出了一条指令写入寄存器 \$s0, 后续指令读取这个寄存器, 这将产生冲突. 这种冲突称为读后写 (Read after Write, RAW). add 指令在第 6 周期的前半部分向 \$s0 写入结果. 然而, and 指令需要在第 3 个周期读出, or 指令需要在第 4 个周期读出, sub 指令需要在第 5 个周期读出, 都将得到一个错误的值. sub 后续的指令都将得到正确的值, 这是因为正确的值在第 6 个周期的前半部分已经写入.

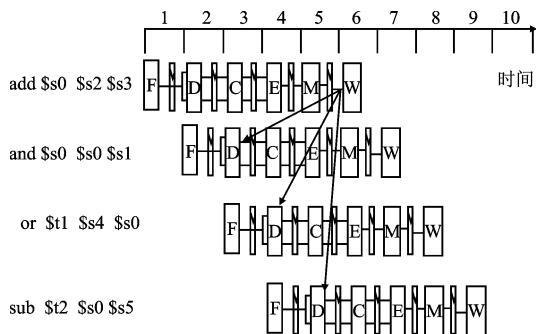


图 2 说明冲突的抽象流水线示意图

对这个问题进一步分析发现: add 指令的计算结果在第 4 个周期就由 ALU 得到, 并且直到第 5 个周期 and 指令才在 ALU 中使用到. 因此可以将

前条指令的结果旁路到后条指令来解决 RAW 冲突. 图 3 中显示了用旁路化解数据冲突的基本的设计原理. 在第 4 个周期中, \$s0 从 add 指令的执行级的结果旁路到 and 指令的选择级. 在第 5 个周期中, \$s0 从 add 指令的存储器级旁路到 or 指令的选择级. 在第 6 个周期中, \$s0 从 add 指令的写回级旁路到 sub 指令的选择级.

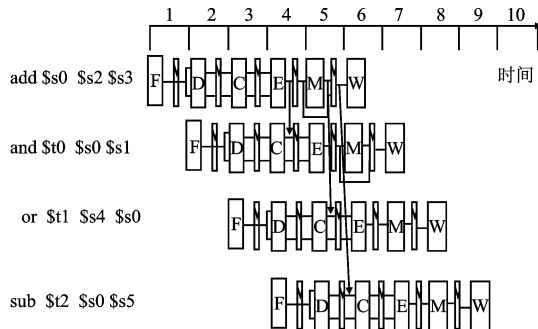


图 3 说明数据旁路的抽象流水线示意图

由于存储器级得到的数据有两种情况: 一种是由前一级 ALU 的计算结果传递过来作为要写入寄存器的值, 一种是从数据存储器下载的值作为要写入寄存器的值, 因此在存储器级加一个两路选择器, 来确定要旁路的值.

如果执行级、存储器级和写回级同时包含了匹配的目的寄存器, 则执行级拥有最高优先级, 存储器级次之, 写回级最后. 因为执行级包含了最近执行指令的结果, 另外两个次之.

图 4 中显示了一个新问题. lw 指令在第 5 个周期的最后才从存储器中接收到数据. 但是 and 指令在第 5 个周期开始时就需要这个数据作为源操作数. 因此使用旁路的方法无法解决这种问题.

我们采用的解决方法是阻塞流水线: 将操作挂

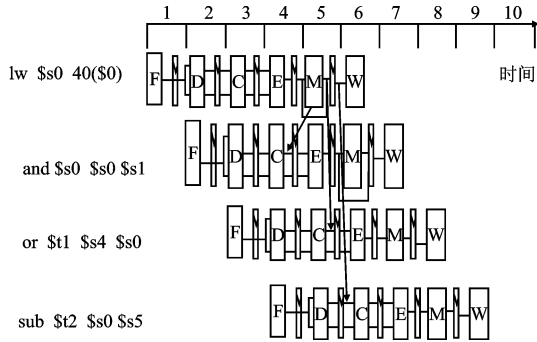


图4 说明lw旁路问题抽象流水线示意图

起直至数据有效。图5显示了在选择级阻塞相关的and指令。该指令在第4个周期进入选择级,并一直阻塞到第5个周期位置。在此过程中,后续的or指令也必须保持在译码级,sub指令必须保持在取指级。在第5个周期时,lw指令的结果从存储器级旁路到and指令的选择级,从写回级旁路到or指令选择级。由于要等到阻塞结束时,执行级才能得到两个正确的源操作数,因此在阻塞时要对相应的ALU源操作数进行刷新。

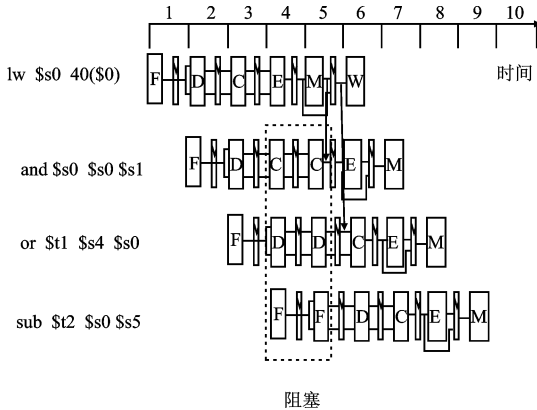


图5 通过阻塞方式解决冲突的抽象流水线示意图

对于lw指令,memtoRegE信号有效。因此计算阻塞和刷新的逻辑如下所示:

$$lwstallC = memtoRegE \& (writeregE == rsC | writeregE == rtC)$$

$$stallC = lwstallC$$

$$stallF = stallD = stallC$$

$$flushE = stallC$$

2.4 控制冲突的处理

beq指令和Jump指令将产生控制冲突,因为在取指时是否发生转移还尚未确定,所以流水线处理器并不知道取哪条指令^[5-6]。我们假设所有分支都不会发生而只是简单地继续按照顺序执行程序。如果分支的确发生,分支指令的后4条指令将通过清除

这些指令流水寄存器的方式来刷新(抛弃)。

图6中显示了这样的机制,其中从地址20~64的分支将发生。这个分支在第5个周期中确定,此时后续的and,or,sub和add指令(地址分别是24,28,2C和30)已经被取出。这些指令必须刷新,并且在第6个周期中从地址64取出slt指令。这在某些时刻已经有所改进,但是一旦分支发生需要刷新很多指令从而依然降低了性能。

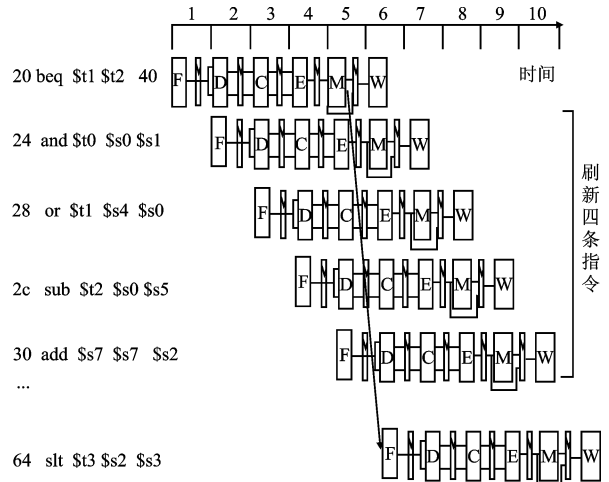


图6 当分支发生时刷新操作的抽象流水线示意图

如果能尽早确定是否发生分支,可以减少刷新多条指令的开销。确定分支只需要比较两个寄存器是否相等。使用一个专门的相等比较器可以将执行减法和0检测变得很快。考虑到五级流水线中把比较器放在译码级会延长流水线的周期,我们采用把比较器放在选择级的设计方式,这样可以直接获得要比较的两个寄存器的值,而且不必阻塞。

图7中显示了在第3个周期中尽早完成分支判断的流水线操作。在第4个周期,刷新了and指令和or指令而且取出了slt指令。此时,分支预测错误开销已经从4条指令减少到2条。

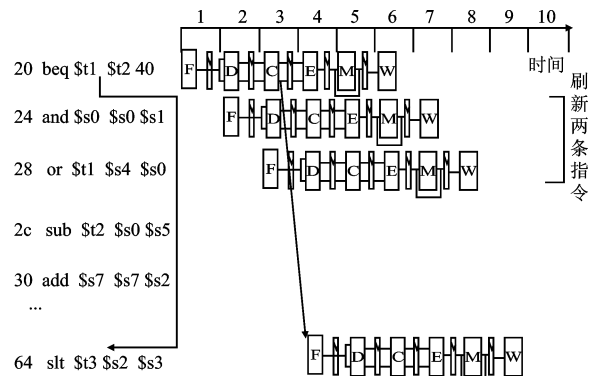


图7 尽早确定分支的抽象流水线示意图

此外我们把 jump 指令的转移地址也在选择阶段计算出来,这样在选择级就可以确定两种转移是否会发生. equalC 作为比较器的结果. 刷新的逻辑如下所示:

```
pcsrcC= branchC&equalC
flushC=pcsrcC|jumpC
flushD=flushC
```

3 仿真与综合

六级流水的设计使用 Verilog 语言描述,并用

ModelSim 进行仿真. 图 8 为在 ModelSim 中仿真的波形. 其中 instrF 为取址阶段按照指令地址取出的指令,由图 8 看出每个周期都有一条指令被取出并执行,并且最终的运行结果也与预期一致. 证明了设计的正确性.

本文给出了的六级流水线在 Altera 公司的 DE2 开发板上的验证结果,表 1 中显示了综合后开发板的资源利用率. 综合后的频率可达到 81. 7 MHz. 并且片内资源使用率不足 3%. 此外,本设计在性能上也有一定的提升,如表 2 所示.

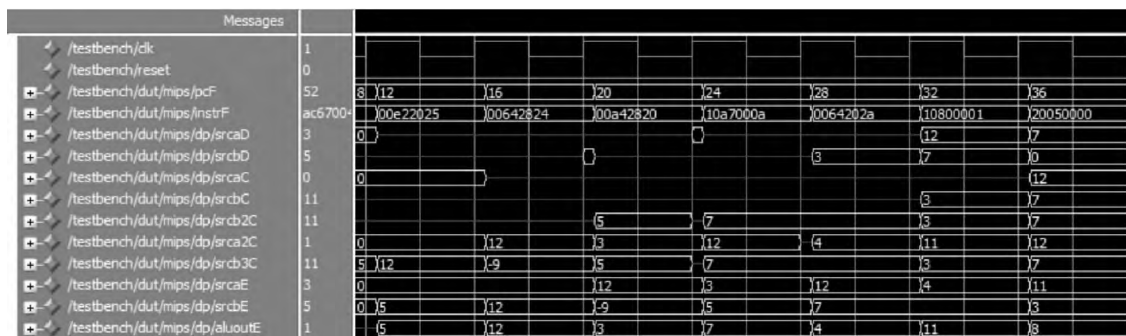


图 8 六级流水线的仿真结果图

表 1 综合后开发板的资源利用率

Complication Report-Flow Summary	
Familiy	Cyclone
Device	EP2C35F672C6
Met timing requirements	Yes
Total logic elements	914/33,216(3%)
Total combinational functions	759/33,216(2%)
Dedicated logic registers	539/33,216(2%)
Total registers	539
Total pins	68/475(14%)
Total memory bits	4096/483,840(<1%)

表 2 性能对比

器件名称	其他同类设计/ MHz ^[7-10]	本文设计/ MHz	性能提升/%
DE2	-	81. 7	-
Spartan-II	26	34. 5	32. 69
Virtex-E	120	139. 8	16. 5
Spartan-IIIE	25	32. 53	30. 12
XC2v4000	121	145. 47	20. 23

4 结束语

本文创造性地设计出一种六级流水线的处理

器. 它把处理器按功能划分为适当的六个阶段,平衡分配了每个阶段的任务. 并且在 FPGA 开发板上进行了有效的验证,保证了设计的正确性. 且经过综合后能满足较高的时钟频率. 该设计的实现为以后研究更加复杂高效的处理器奠定了基础.

参考文献:

- [1] 张英武,袁国顺. 32 位嵌入式 RISC 处理器的设计与实现[J]. 微电子学与计算机, 2008, 25(6): 14-17.
- [2] 刘览,郑步生,施慧彬. 基于 FPGA 的 32 位 RISC 微处理器设计[J]. 数据采集与处理, 2011, 26(3): 367-373.
- [3] David Money Harris. 数字设计和计算机体系结构[M]. 陈虎,译. 北京:机械工业出版社, 2009.
- [4] 白中英,戴志涛,周峰. 计算机组成原理[M]. 北京:科学出版社, 2007.
- [5] 杨全胜,王晓蔚,张志政. 计算机系统综合课程设计[M]. 北京:清华大学出版社, 2008.
- [6] 邹志斌. 基于 MIPS 指令集的 RISC 微处理器控制模块的设计与实现[D]. 武汉:华中科技大学, 2008.
- [7] Mane P S, Gupta I, Vasantha M K. Implementation of RISC processor on FPGA[C]// 2006 IEEE International Conference on Industrial Technology. Mum-bai, India, 2006: 2096-2100.
- [8] 江艳,廉殿斌,李勇. 64 位 RISC 微处理器的结构设计[J]. 微电子学与计算机, 2005, 22(4): 72-77.

(下转第 39 页)

7 结束语

本文提出了基于主题聚类的Web资源个性化推荐方法.该方法在推荐过程中考虑了Web资源的主题信息,并基于主题信息对Web资源聚类;然后,基于用户的浏览行为,隐式获取用户的兴趣,考虑到用户兴趣自身特点,在算法中加入时效的概念,真正动态获取用户偏好,在此基础上,实现Web资源的动态推荐;最后通过实验验证了该算法的有效性和稳定性.本文提出的方法局限于文本型Web资源的推荐,对音频、视频等Web资源的推荐还需要进一步研究.

参考文献:

- [1] Mojtaba Salehi. Application of implicit and explicit attribute based collaborative filtering and BIDE for Web resource recommendation[J]. Data & Knowledge Engineering, 2013, 87(9): 130-145.
- [2] Manouselis N, Vuorikari R, Van Assche F. Collaborative recommendation of e-Web resources: an experimental investigation[J]. Journal of Computer Assisted Web, 2010, 5(3/4): 347-370.
- [3] Li Dun, Ma Yongtao, Guo Jian li. Words semantic orientation classification based on HowNet[J]. The Journal of China University of Post and Telecommunications, 2009, 16(1): 106-110.
- [4] Holanda Olavo, Ferreira Rafael, Costa Evandro. Educational resources recommendation system based on agents and semantic web for helping students in a virtual Web environment[J]. International Journal of Web Based Communities, 2012, 8(1): 333-353.
- [5] 刘志勇, 刘磊, 刘萍萍, 等. 一种基于语义网的个性化Web资源推荐算法[J]. 吉林大学学报:工学版, 2009, 39(2): 391-395.
- [6] Perugini Saverio, Marcos Andre Goncalves, Edward A Fox. Recommender systems research: a connection-centric survey[J]. Journal of Intelligent Information Systems, 2004, 23(2): 107-143.
- [7] Matias Nicoletti, Silvia Schiaffino, Daniela Godoy. Mining interests for user profiling in electronic conversations[J]. Expert Systems with Applications, 2013, 40(2): 638-645.
- [8] Li Jun, Zhang Peng. Mining explainable user interests from scalable user behavior data[J]. Procedia Computer Science, 2013(17): 789-796.
- [9] Liu Xiaozhong, Turtle Howard. Real-time user interest modeling for real-time ranking[J]. Journal of the American Society for Information Science and Technology, 2013, 64(8): 1557-1576.
- [10] Li Zhengwei, Xia Shixiong, Niu Qiang. Research on the uUser interest modeling of personalized search engine[J]. 武汉大学学报:自然科学版, 英文版, 2007, 53(5): 893-896.
- [11] 尹春晖, 邓伟. 基于用户浏览行为分析的用户兴趣获取[J]. 计算机技术与发展, 2008, 18(5): 37-39.
- [12] Cheng Chen, Gui Pingliao, Xiao Huishi, et al. The model of user interest update and user classification in personal information push service[J]. Procedia Environmental Sciences, 2010, 2(10): 262-268.
- [13] Pluis Martinez, Luis G Perez, Manuel Barranco. A multigranular linguistic content-based recommendation model: research articles[J]. International Journal of Intelligent Systems, 2007, 22(5): 419-434.

作者简介:

张万山 男, (1973-), 硕士, 讲师. 研究方向为 web 信息挖掘.

肖 瑶(通讯作者) 女, (1987-), 硕士研究生. 研究方向为 web 信息挖掘. E-mail: hubuxiaoyao@163.com.

梁俊杰 女, (1974-), 博士, 副教授. 研究方向为数据分析与云计算.

余敦辉 男, (1974-), 博士后, 副教授. 研究方向为服务计算与大数据.

(上接第34页)

- [9] Kislyakov M, Mosin S. A processor development in-programmable logic basis [C]// The Experience of Designing and Application of CAD Systems in Microelectronics-CADSM 2007. Lviv-Polyana, Ukraine, 2007: 182-185.
- [10] 冯海涛, 王永纲, 石江涛, 等. 基于FPGA的32位整数微处理器的设计与实现[J]. 小型微型计算机系统, 2005, 26(6): 1113-1117.

作者简介:

孙巧稚 女, (1988-), 硕士研究生. 研究方向为计算机系统结构. E-mail: 404914613@qq.com.

施慧彬 男, (1966-), 博士, 副教授. 研究方向为计算机体系结构、多核微处理器设计、可重构计算、多核技术、嵌入式系统、编译技术、软件工程.