

针对嵌入式系统的低功耗存储器管理单元设计

朱贺飞, 陆 超, 周晓方, 闵 昊, 周 电

(复旦大学专用集成电路与系统国家重点实验室, 上海 201203)

摘 要: 针对 Linux 操作系统, 实现了面向 32 位 RSIC 嵌入式处理器的低功耗存储器管理单元。通过在指令快表中增加预比较电路, 提高了处理器连续访问同一虚拟页面时的地址转换效率, 降低指令快表命中时的功耗 37.07%。两级比较结构的内容寻址存储器与传统结构相比, 在失效和命中时分别可以取得 44.98% 和 74.94% 的功耗节省。该文设计的存储器管理单元能够很好地和 Linux 配合, 完成地址映射及存储权限管理。

关键词: 嵌入式系统; 低功耗; 存储器管理单元; 快表

Design of Low Power Memory Management Unit for Embedded System

ZHU Hefei, LU Chao, ZHOU Xiaofang, MIN Hao, ZHOU Dian

(ASIC & Systems State Key Laboratory, Fudan University, Shanghai 201203)

【Abstract】 To support Linux operating system, a low power memory management unit for a 32-bit embedded processor is proposed. Since the added pre-comparing circuits can greatly improve the efficiency of address translation when the processor needs to access the same virtual address page, the power consumption of ITLB can be reduced by 37.07%. Two-level hierarchical content addressable memory (CAM) also has a power reduction of 44.98% (miss) and 74.94% (hit) compared with the conventional fully parallel comparing architecture. Experiments show that RSIC embedded processor with MMU can run Linux perfectly.

【Key words】 Embedded system; Low power; Memory management unit; TLB

移动通信等依靠电池供电的设备对于嵌入式处理器提出了苛刻的功耗要求, 这些设备中的处理器通常都具有支持虚拟内存系统的存储器管理单元(Memory Management Unit, MMU), 通过MMU将虚拟地址转换为物理地址, 用来访问Cache或者主存。由于处理器在每一个指令周期和数据周期都将访问MMU, 因此MMU的功耗在一定程度上影响着整个系统的功耗。以Intel比较成功的低功耗处理器StrongArm SA-110^[1]为例, 存储器管理单元占整个处理器功耗的 17%。因此, 降低MMU的功耗对于设计低功耗嵌入式系统有着重要的现实意义。

本文针对Linux的内存管理机制, 为家庭网络核心SoC平台中使用的 32 位嵌入式RISC处理器^[2]设计了支持虚、实地址转换和存储权限控制的MMU。为了提高处理器访问指令和数据的效率, MMU中包括了彼此独立的指令快表(Instruction Translation Lookaside Buffer)和数据快表(Data Translation Lookaside Buffer), 用来实现地址映射关系的存储和比较。在实现MMU基本功能的基础上, 通过在指令快表中增加指令预比较电路来减小对指令快表的访问频率, 从而大大降低了指令快表命中时的访问功耗; 通过改进快表(Translation Lookaside Buffer, TLB)中内容寻址存储器的结构为两级并行比较, 在满足系统所需地址映射速度的前提下, 进一步降低了由快表进行虚拟地址比较时引入的动态功耗。

1 家庭网络 SoC 平台及 Linux 的内存管理体系

家庭网络 SoC 平台中使用的嵌入式系统架构如图 1 所示, 包括自行设计的 32 位 RSIC 处理器、MMU、Cache 以及片外主存。处理器给出的虚拟地址经过 MMU 地址映射后

得到物理地址。处理器中运行的 Linux 操作系统采用页式内存机制, 内存空间分为内核(Kernel)空间和用户(User)空间, 处在内核空间的地址使用线性映射的方式, 处在用户空间的地址使用页表映射的方式。

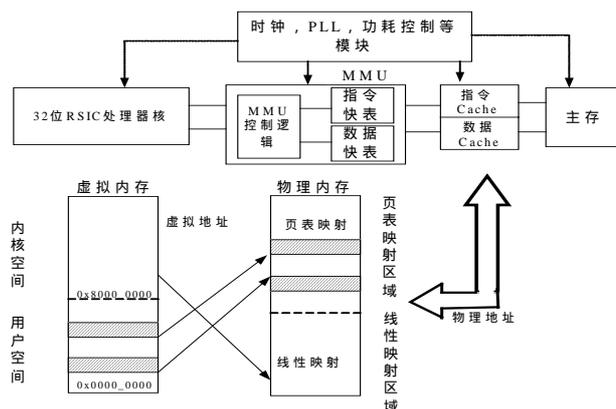


图 1 MMU 的功能及其在系统中的位置

2 MMU 硬件电路结构及工作原理

如图 1 所示, MMU 由控制逻辑、指令快表和数据快表 3

基金项目: 国家“863”计划基金资助项目(2003AA1Z1120); 上海市科委SDC计划基金资助项目(037062020)

作者简介: 朱贺飞(1982-), 男, 硕士生, 主研方向: 专用集成电路设计; 陆 超, 学士; 周晓方, 高工; 闵 昊、周 电, 博导、教授

收稿日期: 2006-07-23 **E-mail:** hfzhu@fudan.edu.cn

部分组成。控制逻辑采用Verilog硬件描述语言实现，主要负责对地址映射的时序及访问权限进行控制。指令快表和数据快表采用中芯国际 0.18 μm工艺进行全定制设计，为 32 路全相联形式^[4]。虚拟页号(虚拟地址高 20 位)和物理页号(物理地址高 20 位)分别存储在快表的CAM^[5](Content Addressable Memory)阵列和SRAM阵列中。虚拟地址的页面映射过程如图 2 所示，当Linux运行至用户空间后，处理器给出的虚拟页号将同CAM阵列中存储的虚拟页号比较，如果有一路命中就将相应一路SRAM单元中的物理页号读出，与虚拟地址的低位拼接在一起形成物理地址^[6]；如果快表中的 32 路虚拟地址都没有命中(称之为快表失效)，MMU将会查询两级页表取得的物理地址。

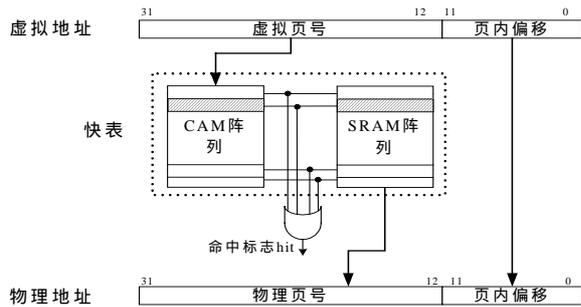


图 2 页面映射原理

3 低功耗指令快表设计

传统结构的指令快表需要在每次访问指令空间时都进行虚拟页号的比较，而存储虚拟页号的 32 路CAM阵列进行全并行比较不仅效率低下，还将带来大量的功耗损失^[7]。根据程序的临近性原理，60%~80%的指令都是顺序执行的^[8]。当指令顺序执行时，处理器给出的指令地址很有可能处于同一个虚拟页面的范围内(虚拟地址高 20 位相同)。这时完全可以省略指令快表的查找，直接利用上一次访问的结果取得物理页号，从而减少 32 路CAM并行比较的次数，大大降低指令快表命中时的功耗。文献[9]利用上述原理，在main TLB之前增加一级Filter TLB，Filter TLB命中时可以省略对main TLB的查找，当filter TLB具有较高命中率时可以在很大程度上降低功耗。但这种方法将两级TLB工作在串行模式下，Filter TLB失效时会给整个处理器的关键路径增加延时。

本文设计的指令快表充分考虑了连续访问同一虚拟页面对于降低快表功耗的积极作用，同时不以牺牲MMU进行地址映射的速度为代价。如图 3 所示，指令快表首先对每一次访问的虚拟页号和读出的物理页号都进行锁存，当处理器给出新的虚拟地址时，通过比较器来判断前后连续两次访问的虚拟页号是否一致。如果比较器比较命中，Hit_previous信号置高，使CAM比较使能信号CAM_EN无效，避免了 32 路CAM阵列的并行比较以及对SRAM阵列的读取，而直接从锁存的结果中取得所需的物理页号。由图 3 可以看出，只有当指令预比较电路与快表的输入信号锁存及时序控制部分并行工作时，才能避免对系统关键路径的负面影响，这就对指令预比较电路中使用的比较器提出了苛刻的设计要求。本文采用中芯国际 0.18μm 工艺对 20 位的指令预比较电路进行全定制设计，仿真结果表明，所采用的动态比较电路结构^[10]完成一次比较的最大延时为 0.43ns，完全能够将指令预比较与输入地址锁存并行进行，不再引入额外延时。采用Synopsys公司Hspice对指令快表命中时的功耗进行仿真表明，指令快表命

中的功耗为 26.64mW，而省略比较虚拟页号的情况下仅为 16.76mW，可以降低 37.07%。

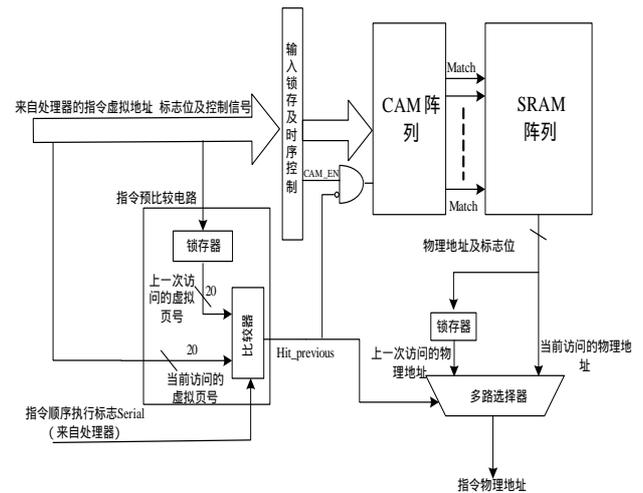


图 3 指令快表电路结构

通过对 Linux 操作系统运行过程中的指令进行统计，图 4 比较了 MMU 对传统结构指令快表和本文设计的低功耗指令快表各自所需的访问次数。图中同时给出了文献[9]中介绍的filter-TLB运行若干种测试程序时的命中情况(若filter-TLB命中，则可以省略对 main TLB 的查找，其平均命中概率为 80%~90%)。由于处理器连续访问同一个虚拟页面的概率很大，而本文指令快表和文献[9]中的快表都只有在虚拟页号变化时才进行 TLB 的访问，因此具有相似的访问频率，可以省略 90%左右的指令快表比较，从而大大降低指令快表的访问功耗。

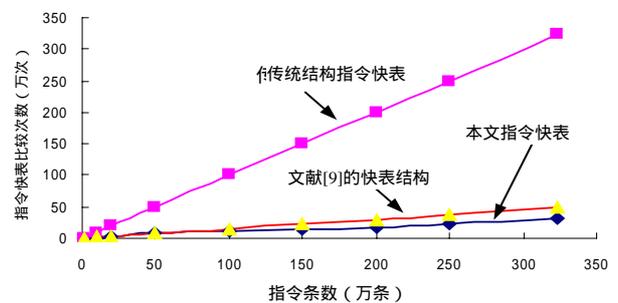


图 4 指令快表访问频率比较

4 低功耗 CAM 结构

数字集成电路中的动态功耗由电容的充放电引起，可用下式进行计算^[5]。其中 V_{dd} 为电源电压， C 为被充放电的电容， α 为活动因子(即每个时钟周期内输出翻转的次数)， f 为电路的工作频率。可见，在不改变电路工作电压和频率的前提下，减小活动因子 α 是降低功耗的一种有效措施。

$$P_{dynamic} = \alpha CV_{dd}^2 f$$

如图 2 所示，快表中一般采用内容寻址存储器(CAM)来实现虚拟页号的存储和比较。传统结构的全并行内容寻址存储器由 9 管CAM单元构成^[5]。由于每次进行虚拟地址比较时，未命中CAM单元的Match线都要进行一次充放电，因此其主要功耗来源是Match线的活动因子 α 过大。文献[11]提出一种两级比较CAM结构来降低由未命中CAM单元的Match线引入的动态功耗。处理器给出的虚拟页号低位首先与CAM阵列中存储的虚拟页号的低位进行比较，低位比较全部命中时才进行高位比较，从而减小了CAM单元虚拟地址高位Match线的

活动因子。由于文献[11]采用与非(NAND)形式CAM单元进行第一级比较时,命中信号串行传递,影响了虚拟页号比较的速度。

本文提出的低功耗 CAM 单元及 Match 线结构如图 5 所示。将每一路 CAM 都分成两级,图中的 EN1 和 EN2 分别是两级 CAM 的比较使能信号。当且仅当第一级的 K 个单元都命中后才打开第二级的 MN2 管,使第二级进入比较状态,开始第二级的比较;否则第二级的比较完全被省略,减小了第二级 CAM 单元 Match 线的活动因子,从而节省了虚拟地址并行比较的功耗。不同于文献[11]中的第一级串行比较结构,本文第一级中的 CAM 单元都采用或非(NOR)形式, K 个 CAM 单元并行参与比较,大大提高了 TLB 的比较速度。

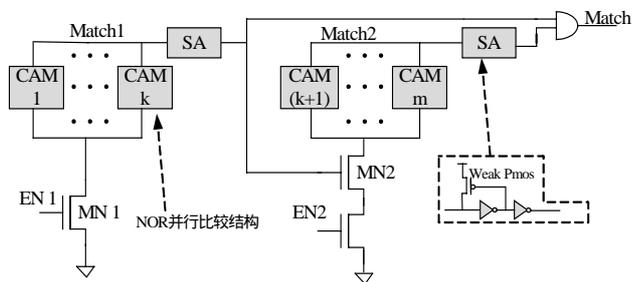


图 5 改进型低功耗 CAM 结构

为进行性能比较,本文采用相同的集成电路工艺(SMIC 0.18 μm)对文献[11]中的两级比较结构及全并行结构的 CAM 进行了设计。图 6 所示的是 3 种不同结构 CAM 的功耗仿真结果,仿真电压 1.8V,频率 200MHz。可见,文献[11]的两级结构可以取得最大的功耗节省,却以 909ps 的延时为代价;而本文的两级比较结构均选用并行比较,可在 323ps 的延时下取得 44.98% (失效时)和 74.94% (命中时)的功耗节省。

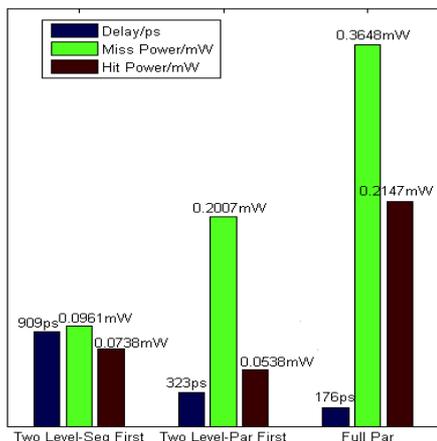


图 6 3 种不同结构 CAM 的功耗、延时比较

5 Linux 移植验证

家庭网络核心 SoC 平台上运行的 Linux 内核版本为 MIPS 公司提供的 Linux-2.4.18,并且已经针对本文所使用的 RSIC 处理器进行了移植。

采用 Mentor Graphics 公司的 Modelsim5.8 对 RSIC 处理

器运行 Linux 操作系统进行仿真,MMU 进行地址转换时的波形如图 7 所示。当处理器的程序指针 PC 为 0x00426d7c 时,执行的指令为 0x001318c0,是一条需要访问外存的 store 操作,此时处理器给出的数据虚拟地址 cpu_daddr 为 0x00000108,经过 MMU 进行地址映射后得到的物理地址 dpaddr 为 0x0043f108,处理器将利用该物理地址完成对 Cache 的访问。目前本文所设计的低功耗 MMU 已经通过了 FPGA 验证,FPGA 上运行的全定制快表电路采用 RTL 级模型代替。

图 7 MMU 地址映射仿真波形

6 结论

针对 Linux 操作系统的内存管理机制,为家庭网络核心 SoC 平台中使用的 32 位嵌入式处理器设计了一款低功耗存储器管理单元。通过在指令快表中增加指令预比较电路,以及改进型两级比较 CAM 结构,在降低 MMU 进行地址映射功耗的同时,兼顾了降低功耗对于系统速度的负面影响。本文硬件设计中的两项核心技术已经申请专利(专利申请号:200410067586.X 和 200510026403.4)。

参考文献

- 1 Montanaro J. A 160-MHz, 32-b, 0.5-W CMOS RISC Microprocessor [J]. IEEE Journal of Solid-state Circuits, 1996, 31(11): 1703-1717.
- 2 李 侠,周晓方,张海清,等. 可自适应变频嵌入式微处理器核的设计[J]. 小型微型计算机系统, 2006, 27(2): 335-338.
- 3 毛德操,胡希明. Linux 内核源代码情景分析[M]. 杭州: 浙江大学出版社, 2001: 38-39.
- 4 Hennessy J L, Patterson D A. Computer Architecture: A Quantitative Approach[M]. 3rd ed. San Francisco: Morgan Kaufmann Publishers, 2003.
- 5 Rabaey J M, Chandrakasan A. Digital Integrated Circuits: A Design Perspective[M]. 2nd ed. Upper Saddle River, New Jersey: Pearson Education, 2003: 670-672.
- 6 Juan T, Lang T, Navarro J J. Reducing TLB Power Requirements[C]// Proc. of the International Symposium on Low Power Electronics and Design. 1997: 196-201.
- 7 Hsiao H Y L, Wang D H, Jen C W. Power Modeling and Low-power Design of Content Addressable Memories[C]//Proc. of IEEE International Symposium on Circuits and Systems. 2001: 926-929.
- 8 Clark L T, Byungwoo C H. Reducing Translation Lookaside Buffer Active Power[C]//Proc. of IEEE International Symposium on Low Power Electronics and Design. 2003: 10-13.
- 9 Lee J H, Park G H. A Selective Filter-bank TLB System (Embedded Processor MMU for Low Power)[C]//Proc. of IEEE International Symposium on Low Power Electronics and Design. 2003: 25-27.
- 10 Edmondson J H. Impact of Physical Technology on Architecture[M]// Design of High-performance Microprocessor Circuits. Chandrakasan, Bowhil, Fox. New York: IEEE Press, 2001: 7-8.
- 11 Yang B D, Kim L S. A Low-power CAM Using Pulsed NAND-NOR Match-line and Charge-recycling Search-line Driver[J]. IEEE Journal of Solid-state Circuits, 2005, 40(8): 1736-1744.