

# 一种6读2写多端口寄存器堆的全定制实现

张 轩, 李兆麟

(清华大学信息科学技术学院, CPU & SoC 实验室, 北京 100084)

**摘要:**采用全定制设计方法实现了一种6读2写的32×32位的多端口寄存器堆,包括结构设计、电路设计、版图设计、仿真验证以及建模建库。该多端口寄存器堆的读写端口互相独立,在一个时钟周期内,能够同时读出6个32位数据,并写入2个32位数据。在电路实现上,采用高速SCL结构的地址译码和分组字线的方法来减少读写延迟。采用了0.18μm 6层金属P阱CMOS工艺来实现版图设计,通过了版图验证和后端仿真。

**关键词:**多端口寄存器堆;高速SCL结构;全定制设计;自顶向下的设计流程

## Full-custom Implementation of 6-read 2-write Multi-port Register File

ZHANG Xuan, LI Zhao-lin

(CPU & SoC Lab, Information Science and Technology College, Tsinghua University, Beijing 100084)

**【Abstract】**This paper presents a full-custom design of a multi-port 32×32 bits register file with 6 read ports and 2 write ports, including architecture design, circuit design, layout design, simulation and verification, modeling and library compiling. As all read/write ports are independent, it can read 6 32-bit data and write 2 32-bit data simultaneously in one clock cycle. The design employs high-speed SCL address-decoder and word-line grouping method to reduce access delay. Its layout is realized in 0.18μm 6-layer-metal P-trap CMOS technology and passes final verification and post simulation.

**【Key words】**multi-port register file; high-speed SCL structure; full-custom design; top-down design flow

寄存器堆(register file),或称寄存器文件,是RISC处理器的一个重要组成部分。例如MIPS微处理器具有32个通用寄存器,并且被配置成寄存器堆的形式使得寄存器资源得到充分的利用。由于传统的单端口寄存器堆已不能满足微处理器繁重的数据读写吞吐量的要求,因此人们研究并改进了寄存器堆的端口设计,提出了多端口寄存器堆的结构。目前,多端口寄存器堆已经成为超标量和多线程微处理器的一个核心部件和重要研究方向。

多端口寄存器堆具有多套读写端口,每套端口均配有独立的地址线,数据线和操作允许线。各套端口在不发生读写冲突的前提下可以同时工作。微处理器在一个时钟周期内可以同时从多个通用寄存器中完成多次读写操作,从而大大加快了微处理器的存取效率,降低了数据等待周期,使微处理器的并行操作性得到极大提高。但端口的增加必然带来配套读写电路复杂性的成倍增长,出于对使用效率、芯片面积、工作速度和产品成本等多方面因素的综合考虑,目前采用的多端口寄存器堆的端口数目都在10个以内。

由于寄存器堆的好坏对微处理器性能的影响至关重要,为达到最佳的性能,常常对寄存器堆采用全定制(full-custom design)的设计方法。此外,全定制的寄存器堆还具有重要的知识产权(intellectual property)价值,独立设计并实现一个全定制的寄存器堆对于开发具有自主知识产权的微处理器具有重要的意义。

### 1 寄存器堆设计方案

本文采用全定制设计方法实现了一个6读2写的多端口

寄存器堆,并最终将其封装成硬IP核。

寄存器堆的容量根据最通用的配置,采用32×32位的设计,即含有32个通用寄存器,每个寄存器的宽度均为32位。寄存器堆的存储单元采用静态随机存储器(SRAM)的形式,在此基础上进行了改进,并配备了多套端口。每套端口均配有各自独立的地址线,数据线和操作允许线,可以独立进行读写操作。

在读写端口的配置上,采用正反互补的设计思想,6组读出端口分为3个正读端口和3个反读端口,正读端口读出的是存储单元的数据本身,而反读端口读出的则是数据本身的反码;2组写入端口同样分为1个正写端口和1个反写端口,正写端口写入的是数据本身,而反写端口写入的则是数据本身的反码。这样的设计既加强了读写的灵活性又从结构上提高了通用寄存器的使用效率。

在I/O接口上,本文采用无握手机制的异步时序设计,数据端口则采用可以直接挂接总线的三态输出设计,既提高了通用型,又很容易在此基础上根据特殊需要改变成同步时序。

#### 1.1 原理和结构划分

寄存器堆的结构一般可以划分成4大部分:存储单元,地址译码逻辑电路,字选逻辑电路,数据输入输出电路<sup>[1-2]</sup>。寄存器堆的整体电路结构如图1所示。

**作者简介:**张 轩(1982-),男,硕士研究生,主研方向:超大规模集成电路与SoC系统芯片设计;李兆麟,副研究员

**收稿日期:**2006-11-19 **E-mail:** zhangx00@mails.tsinghua.edu.cn

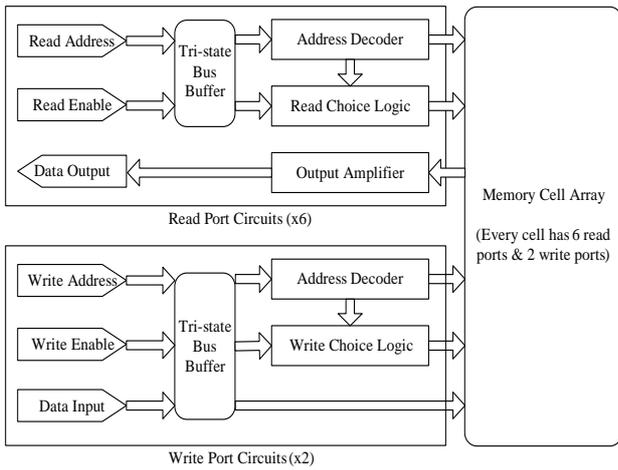


图1 寄存器堆的整体电路结构

### 1.2 存储单元的设计

本文采用读写分开的8管设计SRAM存储核<sup>[2]</sup>。读出方式采用传输门控制的缓冲驱动。写入方式采用预充电平衡的电流写方式<sup>[2-3]</sup>。存储单元的结构如图2所示,处于中心位置的一对反相器构成闭环反馈连接。正反输出通过一级反相器缓冲后送到左右互补的6个数据读出控制端。最上方的2个P管构成了写入预冲管。下方中心位置的传输管则用于写入时的预冲电压平衡。两侧的2个N管在写入时总是处于互补状态,用于控制写入时存储单元的状态转换。而下方外侧的电路则构成了2个互补的数据写入控制端。

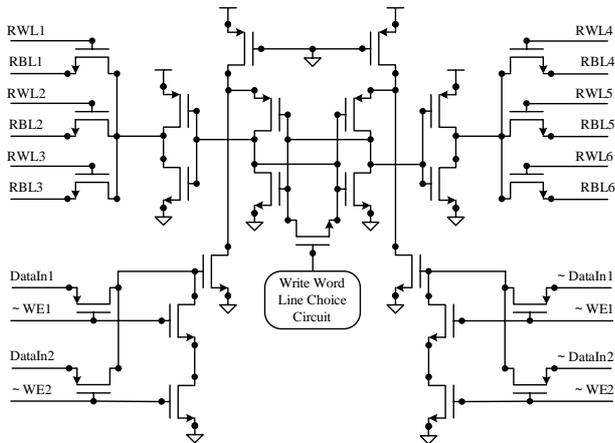


图2 存储单元的结构

### 1.3 地址译码电路的设计

寄存器堆的大小为 $32 \times 32$ 位,即地址的宽度为5位。由于地址线根数较少,因此采用高速SCL结构的一级译码器来缩短译码延迟<sup>[4]</sup>。其基本结构如图3所示,左方的5个N管为5位地址输入,CLK信号连接到译码允许输入,上方的4个P管构成预冲-反馈电路,输出信号经过反相器驱动后连接到字线输出。

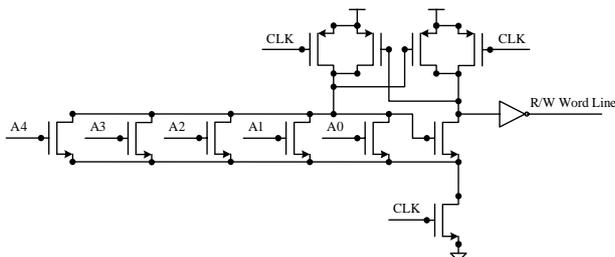


图3 地址译码电路的结构

### 1.4 字选电路的设计

读写的数据宽度均为32位,即每根字线需要驱动32个存储单元。字线较长,采用了分组字线(grouping word-line)的结构<sup>[5]</sup>。用分级的方式逐级驱动字线,把32位的存储单元分为8组,每组4个存储单元,从而降低了字线电容,减小了字线传输延迟。其基本电路如图4所示。

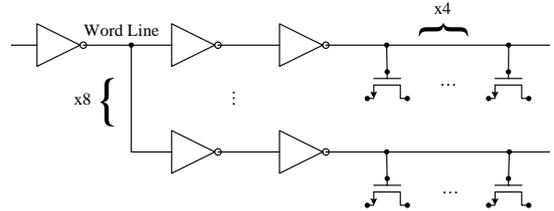


图4 字选电路的结构

### 1.5 数据输入输出电路的设计

存储单元采用单端输出,由于存储阵列的规模不大,不需要微弱信号检出-放大电路,此外由于读写控制逻辑是传输门,在关断时处于高阻状态,因此外部数据I/O接口可以直接与总线挂接。存储阵列的端口连接如图5所示,只画出了存储单元阵列的1个写入端口W1的具体连接,其他7个端口R1,R2,R3,R4,R5,R6和W2的连接与图中类似。

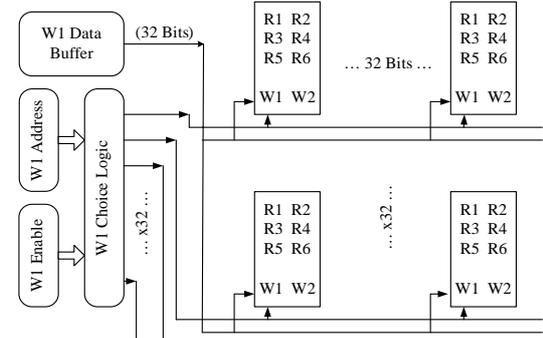


图5 存储阵列的端口连接

## 2 寄存器堆的原理图和前端仿真

寄存器堆的原理图设计使用cadence schematic composer完成。完成后的寄存器堆局部细节放大如图6所示。



图6 寄存器堆原理图局部细节放大

寄存器堆的前端仿真使用cadence Affirma analog artist & spectre simulator完成。

前端仿真得到的电路参数如下(每管脚负载为0.001pf,输入为0V~1.8V标准矩形波):

- (1)T<sub>wc</sub>(min): 2ns
- (2)T<sub>aw</sub>(min): 0.5ns
- (3)T<sub>oaw</sub>(min): 0.5ns
- (4)T<sub>dw</sub>(min): 0.5ns

- (5)  $T_{odw}(\min)$ : 0.5ns
- (6)  $T_{rc}(\min)$ : 2ns
- (7)  $T_{ar}(\min)$ : 0.5ns
- (8)  $T_{oar}(\min)$ : 0.5ns
- (9)  $T_{dr}(\max)$ : 1ns
- (10) 工作频率: 333MHz, 读出/写入周期 3ns
- (11) 工作功率: 127.93mW
- (12) 工作温度: -20 摄氏度 ~ +120 摄氏度
- (13) 输入高电平容限:  $1.8V \pm 0.2V$
- (14) 输入低电平容限:  $0V \pm 0.2V$
- (15) 输出高电平噪声:  $1.8V \pm 0.05V$
- (16) 输出低电平噪声:  $0V \pm 0.05V$

### 3 寄存器堆的版图设计

寄存器堆的版图设计采用 SMIC 的  $0.18\mu m$  6 层金属 P 阱 CMOS 工艺, 使用 cadence layout virtuoso 设计完成。单个存储单元的版图设计如图 7 所示。该版图由 50 个 MOS 管组成, 横向的 8 根长线为地址线, 纵向的 8 根长线为数据线。为减小天线效应, 使用在多层金属之间穿插的方法走线, 避免单层金属走线过长。通过合理配置大管和小管的组合摆放位置, 使整个单元成为规则的矩形并且左右对称, 从而使占用面积达到最小并同时便于组合排列。此外, 在每一个存储单元的上下左右四面均配置了围栏状的衬底接地保护, 用于减小各个单元之间的相互干扰和噪声。

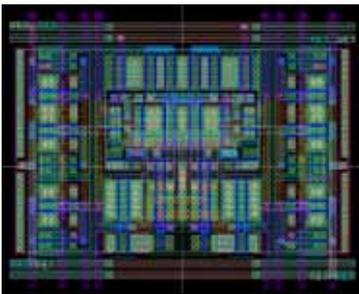


图 7 单个存储单元的版图设计

完成后的寄存器堆的版图局部细节放大如图 8 所示。1 024 个存储单元排列成  $32 \times 32$  的矩阵, 以横向 4 个存储单元为一组使用分组字线的方法级联字线, 而每一条位线则为纵向 32 个存储单元共享。6 个读出端口和 2 个写入端口设计成长条状, 对称排列在存储阵列的两侧, 便于引线输出并节省芯片面积。

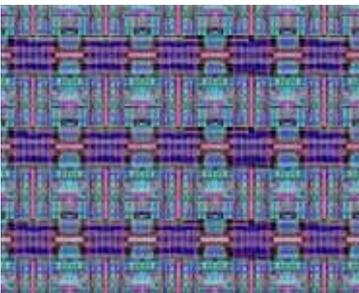


图 8 寄存器堆版图的局部细节放大

### 4 寄存器堆的后端仿真和验证

寄存器堆版图的设计规则检查(DRC)、版图/原理图对照

检查(LVS)以及寄生参数提取(LPE)均使用 cadence dracula verification 完成。寄存器堆的后端仿真采用 cadence spectre simulator 完成。寄存器堆的建模和建库使用 synopsys library compiler 完成。后端仿真得到的电路参数如下(每管脚负载为  $0.001pf$ , 输入为  $0V \sim 1.8V$  标准矩形波):

- (1)  $T_{wc}(\min)$ : 5ns
- (2)  $T_{aw}(\min)$ : 0.5ns
- (3)  $T_{oaw}(\min)$ : 0.5ns
- (4)  $T_{dw}(\min)$ : 0.5ns
- (5)  $T_{odw}(\min)$ : 0.5ns
- (6)  $T_{rc}(\min)$ : 5ns
- (7)  $T_{ar}(\min)$ : 0.5ns
- (8)  $T_{oar}(\min)$ : 0.5ns
- (9)  $T_{dr}(\max)$ : 1ns
- (10) 工作频率: 166MHz, 读出/写入周期 6ns
- (11) 工作功率: 103.49mw
- (12) 工作温度:  $-20\text{ }^{\circ}C \sim +120\text{ }^{\circ}C$
- (13) 输入高电平容限:  $1.8V \pm 0.2V$
- (14) 输入低电平容限:  $0V \pm 0.2V$
- (15) 输出高电平噪声:  $1.8V \pm 0.05V$
- (16) 输出低电平噪声:  $0V \pm 0.05V$

观察波形发现前后仿真主要的不同在于数据输出端口的电平上升和下降速度变缓。其原因主要是在写入和读出数据时, 要对字线和位线上寄生的分布电容进行充放电, 导致了上升和下降时间的变长。在不改动版图内部设计的情况下, 若想得到较高的工作频率, 只需在寄存器堆外围加入地址和数据寄存器, 将读写端口扩展为同步时钟控制的“地址 - 读写控制 - 数据”三段式流水线, 就可以将工作频率提高到 400MHz 以上。

### 5 结论

多端口寄存器堆作为超标量和多线程微处理器的核心部件, 在高性能微处理器的研究开发中占有重要地位。本文实现了一个 6 读 2 写多端口寄存器堆的全定制设计, 实现了自顶向下的全定制设计方法的一整套流程, 最终的设计结果验证了预期的设计目标。

### 参考文献

- 1 Yang Tsen Shau, Horowitz M, Wooley B. A 4-ns  $4K \times 1$ -bit Two-port BiCMOS SRAM[J]. IEEE Journal of Solid State Circuits, 1988, 23(5): 1030-1040.
- 2 Wang Fang, Ji Lijiu. Design of High Speed 2 Write / 6Read Eight Port Register File[C]//Proc. of ASICON'03, Beijing. 2001: 498-501.
- 3 Wang Jinn Shyan, Tseng Wayne, Li Hung Yu. Low Power Embedded SRAM with the Current Mode Write Technique[J]. IEEE Journal of Solid State Circuits, 2000, 35(1): 119-124.
- 4 Khellah M, Elmasry M. A Low Power High Performance Current Mode Multiport SRAM[J]. IEEE Transactions on Very Large Scale Integration Systems, 2001, 9(5): 590-598.
- 5 Khellah M, Elmasry M. Circuit Techniques for High-speed and Low-power Multi-port SRAMS[C]//Proc. of the ASIC Conference. 1998-09-13: 157-161.