

基于非冗余排序的地址总线的功耗优化编码

孙海珺, 邵志标

(西安交通大学 电子与信息工程学院, 陕西 西安 710049)

摘要: 提出了一种新的低功耗非冗余排序总线编码方法, 通过对改进的偏移地址线的动态重排以降低具有高负载的地址总线的功耗. 该编码方法根据偏移地址的值域对地址总线的低位进行优化重排, 通过高位地址总线传送排序矢量至存储器的地址接收端, 相对于传统的地址总线编码方法, 具有更低的总线跳变率. 实验结果表明, 采用所提出的非冗余排序总线编码, 地址总线的跳变率降低了 88.2%, 功耗减少了 76.1%, 有效降低了地址总线的功耗.

关键词: 低功耗; 偏移地址; 地址总线; 总线编码; 跳变

中图分类号: TP302 文献标识码: A 文章编号: 1001-2400(2006)06-0990-05

Power-optimal encoding of the address bus based on irredundant sorting

SUN Hai-jun, SHAO Zhi-biao

(School of Electronics and Information Engineering, Xi'an Jiaotong Univ., Xi'an 710049, China)

Abstract: This paper presents a novel low-power address bus encoding method to reduce the transition activity on address buses and hence reduce power dissipation. The irredundant sorting bus encoding method reduces the power dissipation of highly capacitive memory address bus based on the dynamic reordering of the modified offset address bus lines. This method reorders the ten least significant bits of offset address according to the value of offset address, and the optimal sorting pattern is transmitted through the high bits of bus without the need for redundant bus lines. As compared to the conventional encoding methods, the proposed encoding method is superior in terms of transition activity reduction on the address bus. Experimental results by using an instruction set simulator and SPEC2000 benchmarks show that the irredundant sorting bus encoding method can reduce signal transitions on the address bus by 88.2%, and that the power dissipation of the address bus is reduced by 76.1%, which indicates that the proposed encoding method is very practical for power optimization of the address bus.

Key Words: low-power; offset address; address bus; bus encoding; transition activity

随着集成度和工作频率的不断提高, 芯片的功耗也变得越来越大, 导致了散热和可靠性等问题, 使得便携式电子设备中 SOC 芯片的广泛应用遇到了困难, 低功耗已经成为 SOC 特别是处理器类芯片设计的关键技术之一. 研究表明, 随着 CMOS 工艺进入深亚微米阶段以后, 在系统级芯片中微处理器内部总线功耗占到系统功耗的 50%^[1]. 因此, 总线的低功耗设计非常重要. 由于地址总线要驱动片内外的存储器, 负载很大, 造成了非常大的系统功耗, 为了降低地址总线的功耗, 陆续提出了 T0, T0-Xor 和 Offset-Xor 等编码^[2~4].

笔者提出了一种新的非冗余排序总线编码, 与已提出的各种地址总线编码相比, 具有更低的总线功耗. 当指令顺序执行时, 地址总线不跳变, 总线动态功耗为零, 当出现转移指令时, 如果偏移地址处于 $(-2^{10}, 2^{10})$, 则可对改进的偏移地址的低 10 位进行动态重排, 高位地址线传送排序矢量, 否则, 改进的偏移地址被直接传送到存储器的地址接收端. 采用 0.18 μm CMOS 工艺, 设计实现了各种总线编码器. 模拟验证表明, 采用非冗余排序

收稿日期: 2005-12-28

基金项目: “863”计划课题资助(2005AA1Z1100); 国家部委预研基金资助项目(0105TJ003)

作者简介: 孙海珺(1976-), 男, 西安交通大学博士研究生.

总线编码,性能优于传统地址编码,有效降低了地址总线的跳变率,实现了低功耗地址总线。

1 地址总线功耗

CMOS 电路的功耗主要来自负载电容的充放电电流,即动态功耗

$$P = \sum_{i=1}^N C_i V^2 \alpha_i f \quad (1)$$

其中 C_i 是电路第 i 个节点的负载电容; V 是电源电压; α_i 是节点 i 的跳变率; f 是工作频率; N 是节点数^[5]。对于设计的系统级芯片,设参数 $C_{in,i}$ 是电路内部第 i 个节点的负载电容, $C_{b,i}$ 是地址总线第 i 位的负载电容; N_{in} 是内部节点数, N_b 是地址总线位数。那么式(1)可转化为

$$P = \sum_{i=1}^{N_{in}} C_{in,i} V^2 \alpha_i f + \sum_{i=1}^{N_b} C_{b,i} V^2 \alpha_i f \quad (2)$$

式(2)中地址总线由于要驱动片内外存储器,它的负载电容比电路内部节点的负载电容大几个数量级,因此地址总线的跳变率具有很大的权重,在工作频率、电源电压一定的情况下,降低地址总线的跳变率,可显著降低电路功耗。

2 非冗余排序总线编码

处理器执行的应用程序中 90% 的指令顺序执行^[1],即地址总线上的值主要是连续变化。非冗余排序总线编码基于指令地址的连续性和偏移地址的动态重排,有效降低了总线功耗。当指令顺序执行时,地址总线不发生跳变,总线动态功耗为 0,当出现转移指令时,对改进的偏移地址的低位进行动态重排,降低总线功耗。

在指令跳转时,超过 95% 的分支程序的偏移地址可以用 10 位二进制数编码^[6]。当地址总线传送的是偏移地址时,如果是正的偏移地址,由于偏移地址的高位通常为 0,总线的功耗降低,但是,应用程序中通常存在大量的向后跳转指令,会产生负的偏移地址,负的偏移地址虽然值小,但是当用二进制补码表示时,偏移地址的高位有多个 1,导致地址总线产生大量跳变,增加了总线功耗。在非冗余排序总线编码中,LSBInv(x) 函数被应用以减少负的偏移地址的总线跳变,可表示为

$$M(t) = \text{LSBInv}(b(t) - b(t-1)) \quad (3)$$

式中 $M(t)$ 是 t 时刻改进的偏移地址, $b(t)$ 是 t 时刻的指令地址。当偏移地址是负值时,LSBInv(x) 函数对偏移地址的最高位以外的所有位取反,当偏移地址是正值时,偏移地址保持不变。这样,负的偏移地址的高位 1 的数目被减少,从而减少总线跳变。

由于大多数分支程序的偏移地址可以用 10 位二进制数编码,总线的跳变主要发生在低 10 位。为了进一步减少地址总线的跳变率,当偏移地址处于 $(-2^{10}, 2^{10})$ 时,对改进的偏移地址的低 10 位进行动态重排,高位地址线用来传送排序矢量,否则,改进的偏移地址被直接传送到存储器的地址接收端。低位总线重排时将被分成 m 组,每组 n 位,组内地址线根据排序矢量进行重排,使得地址总线的跳变率最小,总线功耗最低。排序算法如下:

$X(t)$ 表示 n 位二进制数 $\{x(t)_{n-1}, x(t)_{n-2}, \dots, x(t)_0\}$,其中 $x(t)_{n-1}$ 表示 $X(t)$ 的第 $n-1$ 位。排序矢量 $S(t)$ 是 n 个指针的有序集合 $\{s(t)_{n-1}, s(t)_{n-2}, \dots, s(t)_0\}$,其中 $s(t)_{n-1}$ 表示第 $n-1$ 个指针,指向二进制数的第 $s(t)_{n-1}$ 位。 F 是组合函数去产生重排的 n 位二进制数 $Y(t)$:

$$Y(t) = \{y(t)_{n-1}, y(t)_{n-2}, \dots, y(t)_0\} = F(X(t), S(t)) = \{x_{s(t)_{n-1}}, x_{s(t)_{n-2}}, \dots, x_{s(t)_0}\} \quad (4)$$

例如,当 $X(t) = \{0, 1, 0, 1\}$, $S(t) = \{3, 1, 2, 0\}$,则 $Y(t) = \{x_3, x_1, x_2, x_0\} = \{0, 0, 1, 1\}$ 。每一个排序矢量 $S(t)$ 有一个惟一的逆矢量 $S^{-1}(t)$,并且由 $S^{-1}(t)$ 和 $Y(t)$ 可推出 $X(t)$ 为

$$X(t) = F(Y(t), S^{-1}(t)) = F(F(X(t), S(t)), S^{-1}(t)) \quad (5)$$

对于 $S(t)$ 和 $S^{-1}(t)$ 的关系,可推证出如下定理。

定理 1(互逆定理) 若一个包含 n 个指针的排序矢量 $S(t)$ 的第 i 个指针等于 j ,则其逆矢量 $S^{-1}(t)$ 的第

j 个指针等于 i ($0 \leq i \leq n-1, 0 \leq j \leq n-1$).

证明 若 $S(t)$ 的第 i 个指针等于 j , 即 $s(t)_i = j$, 则

$$y(t)_i = x(t)_{s(t)_i} = x(t)_j \quad (6)$$

由逆矢量 $S^{-1}(t)$ 可得

$$x(t)_j = y(t)_{s^{-1}(t)_j} \quad (7)$$

由式 (6),(7) 可得

$$y(t)_i = y(t)_{s^{-1}(t)_j} \quad (8)$$

即 $s^{-1}(t)_j = i$.

例如, 当 $S(t) = \{0, 2, 3, 1\}$, 则 $S^{-1}(t) = \{1, 2, 0, 3\}$.

n 位的地址线经过 n 次动态重排, 才能找到最佳排序地址使得地址总线的跳变率最小. 当 n 值比较大时, 总线重排的次数增多, 造成排序延时增大, 降低了总线性能. 为了减少总线重排的次数, 非冗余排序总线编码在重排地址之前判断地址中含有 1 的个数, 然后根据 1 的数目对地址进行重排, 从而减少了重排次数. 例如 n 为 4 时, 当总线中含有 1 个 1 时, 需要 4 次排序, 当含有 2 个 1 时, 需要 6 次排序. 经过地址重排, 找到 t 时刻的最佳排序地址 $y(t)$, 它与地址总线上 $t-1$ 时刻的地址 $B(t-1)$ 之间的 Hamming 距离 (H) 最小, 从而使地址总线的跳变最少. 如图 1 所示, 有限状态机根据地址中 1 的数目产生排序地址, 经过多次排序比较, 找到最佳排序地址使得地址线的跳变最少. 在生成最佳排序地址时, 根据原始地址, 生成相应的最佳排序矢量. 在地址总线传送重排地址的同时, 排序矢量也传送到存储器的地址接收端. 地址接收端根据接收的地址和排序矢量, 译出原始的存储器地址. 对于 n 位地址线, 设总线中含有 m 个 1, 对应于输出的排序地址, 共有 C_n^m 种排序矢量. 由于 $C_n^m \leq C_n^{n/2}$, 因此 n 位地址线至多需要 $C_n^{n/2}$ 条总线来传送排序矢量. 每一种排序矢量对应一条总线, 当采用该排序矢量得到最佳排序地址时, 其对应的总线发生跳变, 从而通过跳变信号传送排序矢量. 当地址不需要重排直接传送时, 用于传送排序矢量的总线保持不变. 由此可见, 每当一组地址总线经过排序后传送时, 会额外多一条总线发生跳变以传送排序矢量. 因此, 在生成最佳排序地址时, 只有当原始地址传送比排序后的地址传送造成的总线跳变多 1 次以上, 地址排序才会有效减少总线跳变, 否则, 原始地址应被直接传送到存储器的地址接收端.

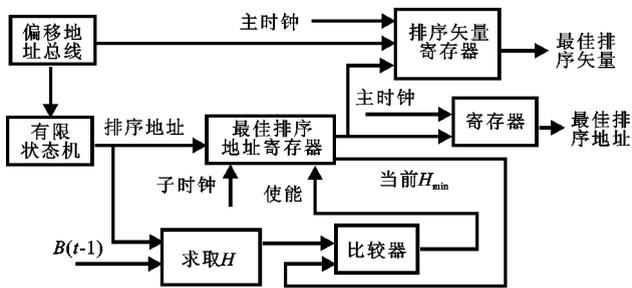


图 1 最佳排序地址生成

图 2 是 n 为 4 时的排序矢量图. 如果前一个时刻地址线上是 $\{0, 0, 1, 1\}$, 下一时刻的地址为 $\{1, 0, 0, 0\}$, 当地址直接传送时, 地址线有 3 根发生跳变, 如果地址经排序矢量 $\{0, 2, 1, 3\}$ 重新排序后成为 $\{0, 0, 0, 1\}$ 发送, 地址线只有 1 根发生跳变, 加上传送排序矢量 $\{0, 2, 1, 3\}$ 的总线发生跳变, 共有两根总线发生跳变, 因此地址经排序后传送减少了总线跳变. 排序地址 $\{0, 0, 0, 1\}$ 对应的排序矢量包括 $\{3, 2, 1, 0\}, \{3, 0, 1, 2\}, \{3, 2, 0, 1\}, \{0, 2, 1, 3\}$, 其中排序矢量 $\{3, 2, 1, 0\}$ 对应的排序地址是原始地址本身, 在传送该排序矢量时总线不需要跳变, 因此传送排序地址 $\{0, 0, 0, 1\}$ 对应的排序矢量时只需要 3 根总线.

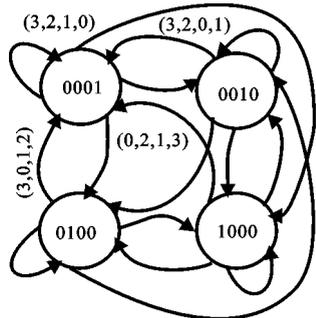


图 2 $n = 4$ 时排序矢量图

表 1 是 n 为 3, 前一个时刻地址总线上是 $\{0, 1, 0\}$, 在当前时刻非冗余排序总线编码器的输出地址. 表 2 是假设前后时刻总线上每一位地址都是随机的, 不同 n 值的排序输出后的总线跳变的减少比率. 从表 2 可以看出, 随着 n 值的增大, 总线跳变减少的比例越大. 非冗余排序总线编码函数为

$$(B(t), DS(t), ES(t)) = \begin{cases} (B(t-1), 1, 0) & , \quad \text{如果 } (b(t) = b(t-1) + R) \quad , \\ (y(t), 0, 0) & , \quad \text{如果 } (|b(t) - b(t-1)| < 2^{10}) \quad , \\ (M(t), 0, 1) & , \quad \text{其他} \quad , \end{cases}$$

式中 $B(t)$ 是地址总线上 t 时刻的值; DS, ES 是附加的地址状态线, DS 传送地址的连续性信息到存储器的地址接收端, ES 传送地址的值域信息到存储器的地址接收端; R 是连续地址间的步长; $y(t)$ 是编码器的输出

地址; $M(t)$ 是 t 时刻改进的偏移地址. 编码器如图 3 所示. 根据非冗余排序总线编码, 存储器的地址接收端的解码函数为

$$b(t) = \begin{cases} b(t-1) + R & , & (\text{DS 为 } 1, \text{ES 为 } 0) & , \\ b(t-1) + \text{LSBInv}^{-1}(F(B(t), s_{\text{op}}^{-1}(t))) & , & (\text{DS 为 } 0, \text{ES 为 } 0) & , \\ b(t-1) + \text{LSBInv}^{-1}(B(t)) & , & (\text{DS 为 } 0, \text{ES 为 } 1) & , \end{cases}$$

式中 s_{op} 是最佳排序矢量, $\text{LSBInv}^{-1}(x)$ 函数是 $\text{LSBInv}(x)$ 函数的逆函数, 解码器结构如图 4 所示.

表 1 $n = 3$ 时非冗余排序总线编码器地址输出

当前时刻地址	原始地址总线跳变数	最佳排序后总线跳变数	编码器输出	编码器输出后总线跳变数
000	1	2	原始地址	1
001	2	1	排序地址	1
010	0	1	原始地址	0
011	1	2	原始地址	1
100	2	1	排序地址	1
101	3	2	排序地址	2
110	1	2	原始地址	1
111	2	3	原始地址	2

表 2 不同 n 值的总线跳变减少比率

M	原始地址总线跳变数	编码输出后总线跳变数	总线跳变减少比率/%
3	96	78	18.75
4	512	390	23.83
5	2 560	1 830	28.52
6	12 288	8 246	32.89
7	57 344	36 162	36.93

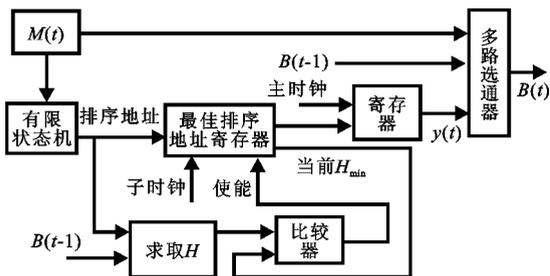


图 3 非冗余排序总线编码器

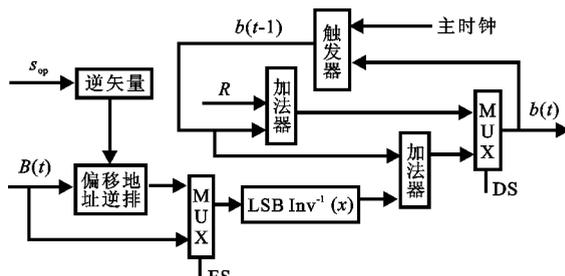


图 4 非冗余排序总线解码器

3 实验结果及分析

通过模拟器 SimpleScalar 产生了 SPEC2000 的测试程序 gzip, gcc, mcf, equake 和 ammp 的地址, 分别使用 T0, T0-Xor, Offset-Xor 与非冗余排序总线编码方法对地址进行了编码. 非冗余排序总线编码在低 10 位地址重排时 n 取 5, 各编码器的总线跳变率的模拟结果如表 3 所示, 由结果可见非冗余排序总线编码优于其他编码方法, 地址总线的跳变率降低了 88.2%. 采用 $0.18 \mu\text{m}$ CMOS 工艺完成了上述各种编码器的设计, 在 120 MHz 频率下对各编码器分别进行了模拟, 非冗余排序总线编码器的面积为 $17731 \mu\text{m}^2$, 动态功耗为 $1383 \mu\text{W}$, 在系统级芯片中, 编码器的面积和功耗实际上是可以忽略的.

各种编码技术的功耗降低与存储器地址总线的负载关系如图 5. 相对于传统地址编码, 非冗余排序总线编码方法有效降低了地址总线的功耗, 并且, 随着总线的负载电容的增加, 地址总线功耗的降低比例越大. 将以上各种编码器应用在 32 位浮点 RISC 微处理器中^[7], 对地址总线的功耗分别进行了模拟, 结果如表 4 所

示,采用非冗余排序总线编码器,地址总线的功耗降低了 76.1%。

表 3 不同编码器的地址总线跳变率

测试程序	原始地址	T0	T0-Xor	Offset-Xor	非冗余排序编码
Gzip	1	0.362	0.185	0.195	0.092
Gcc	1	0.253	0.192	0.211	0.128
Mcf	1	0.274	0.179	0.208	0.123
Equake	1	0.382	0.175	0.188	0.101
Ampmp	1	0.342	0.166	0.190	0.146
平均	1	0.323	0.179	0.198	0.118

表 4 采用编码器的 RISC 处理器的地址总线功耗

	动态功耗/mW	功耗减少率/%
原始地址	98.3	0
T0	52.3	46.8
T0-Xor	37.1	62.3
Offset-Xor	40.8	58.5
非冗余排序编码	23.5	76.1

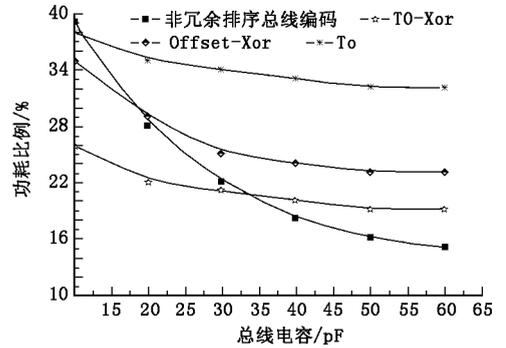


图 5 不同编码器的地址总线的功耗降低与负载的关系

4 结束语

提出了非冗余排序总线编码.当指令顺序执行时,地址总线不跳变,动态功耗为零,当出现转移指令时,如果偏移地址处于 $(-2^{10}, 2^{10})$,则对改进的偏移地址的低 10 位进行动态重排,高位地址线传送排序矢量,否则,改进的偏移地址被直接传送到存储器的地址接收端.实验结果表明,与原始地址直接传送相比,采用非冗余排序总线编码,地址总线的跳变率降低了 88.2%,功耗减少了 76.1%,性能优于传统的地址编码,特别适合于低功耗地址总线的实现.

参考文献:

- [1] Sunpack H, Ki S C, Taewhan K. Bus-invert Coding for Low-power I/O-a Decomposition Approach [A]. IEEE Midwest Symposium on Circuits and Systems[C]. Michigan: IEEE, 2000. 750-753.
- [2] Benini L, Micheli D G, Macii E. Asymptotic Zero-transition Activity Encoding for Address Buses in Low-power Microprocessor-based Systems[A]. IEEE 7th Great Lakes Symposium on VLSI[C]. Illinois: IEEE, 1997. 77-82.
- [3] Mamidipaka M N, Hirschberg D S, Dutt N D. Adaptive Low-power Address Encoding Techniques Using Self-organizing Lists [J]. IEEE Trans on VLSI Systems, 2003, 11(5): 827-834.
- [4] Olivieri M, Pappalardo F, Visalli G. Bus-switch Coding for Reducing Power Dissipation in Off-chip Buses [J]. IEEE Trans on VLSI Systems, 2004, 12(12): 1374-1377.
- [5] Tsui C Y, Pedram M. Accurate and Efficient Power Simulation Strategy by Compacting the Input Vector Set [J]. IEEE Trans on VLSI Systems, 1998, 25(1): 37-52.
- [6] Hennessy J L, Patterson D A. A Quantitative Approach on Computer Architecture [M]. San Francisco: Morgan Kaufmann Publishers, 1996.
- [7] Sun Haijun, Shao Zhibiao. Research on 32-bit Floating-point RISC Microprocessor with High Performance and Low Power Consumption [J]. Journal of Xi'an Jiaotong University, 2005, 39(5): 607-610.

(编辑: 齐淑娟)