

# 低功耗总线编码技术

尹立群<sup>1</sup>, 冯庆<sup>2</sup>, 袁国顺<sup>1</sup>

(1. 中科院微电子研究所, 北京 100029; 2. 中科院电子学研究所, 北京 100080)

**摘要:** 为了降低总线翻转率通常对总线进行编码。B-I 编码是这些编码方法中比较简单实用的一种编码方法。但在连续传输时, 它对翻转率降低不明显。该文对 B-I 编码加以改进, 在编码前进行连续与非连续的判断。在连续传输时采用渐进零翻转编码, 从而使得总线进行连续传输时翻转率接近为零。改进后的编码对同一段程序, 相对于 B-I 编码翻转率进一步下降了 28.7%, 面积增加了 18 225 $\mu\text{m}^2$ 。

**关键词:** 先进微控制器总线结构; 低功耗设计; 总线编码; 翻转率; B-I 编码

## Low Power Bus Coding Approach

YIN Li-qun<sup>1</sup>, FENG Qing<sup>2</sup>, YUAN Guo-shun<sup>1</sup>

(1. Institute of Microelectronics of Chinese Academy of Sciences, Beijing 100029;

2. Institute of Electronics of Chinese Academy of Sciences, Beijing 100080)

**【Abstract】** The method of bus encoding is for cutting down the transition frequency. The Bus-Invert(B-I) coding is the most simple and practical. But the transition frequency reduction is not obviously during the sequential transfer. In this paper, it judges the type of sequential and non-sequential transfer and takes the method of asymptotic zero-transition activity encoding during the sequential transfer. Compared with the B-I coding method for the same program, it can cut down the transition frequency by 28.7% and area increasing by 18 225 $\mu\text{m}^2$ .

**【Key words】** Advanced Micro-controller Bus Architecture(AMBA); low-power design; bus coding; transition frequency; Bus-Invert(B-I) coding

### 1 概述

CMOS 集成电路电路的功耗可以分为动态功耗和静态功耗 2 种。由于 CMOS 电路的结构特点导致一个 CMOS 门的静态功耗在毫瓦级<sup>[1]</sup>可以忽略不计, 因此本文只讨论电路的动态功耗。芯片的动态功耗可表示为

$$P_{\text{chip}} \propto \sum_{i=1}^N C_{\text{load}} \cdot V_{\text{dd}}^2 \cdot f \cdot P_i \quad (1)$$

其中,  $i$  为电路的节点数;  $C_{\text{load}}$  为节点  $i$  处的负载电容;  $V_{\text{dd}}$  为节点  $i$  处的电压;  $f$  为频率;  $P_i$  为节点  $i$  的翻转因子。

目前随着便携式的设备不断普及, 对芯片高性能低功耗的要求不断提高。从式(1)中可以看出减小  $C_{\text{load}}$ ,  $V_{\text{dd}}$ ,  $f$ ,  $P_i$  都可以降低电路的功耗。文献[2-4]通过降低  $C_{\text{load}}$ ,  $V_{\text{dd}}$ ,  $f$  来实现电路的低功耗。本文主要从降低翻转因子的角度考虑降低电路的功耗。在此假设电路的每个节点处的电容都等于节点的平均电容, 这样就可以将式(1)转化为下式所示的形式:

$$P_{\text{chip}} \propto C_{\text{average}} \cdot V_{\text{dd}}^2 \cdot f \cdot N(\text{transitions}) \quad (2)$$

其中,  $N$  是总的节点数;  $\text{transitions}$  是各节点中翻转频率最高的翻转值。由式(2)可以看出在电源电压( $V_{\text{dd}}$ )和频率( $f$ )固定不变时, 电路的功耗只与  $C_{\text{average}}$  和  $N(\text{transitions})$  成正比。由此可以将电路的节点电容分为内部节点电容和外部 I/O 节点电容两部分, 将式(2)转化成下式的形式:

$$P_{\text{chip}} \propto C_{\text{int}} \cdot N(\text{transitions})_{\text{int}} + C_{\text{ext}} \cdot N(\text{transitions})_{\text{I/O}} \quad (3)$$

现在, 芯片的布局大部分都采用 I/O 压焊点在四周包围内部电路的布局方式。在电路设计时, 一方面为了提高器件的驱动能力需要通过增大压焊点的尺寸的方式来增加 I/O 节点的负载电容, 另一方面 I/O 节点处通常还与外部电路的电容或互连线相连。这 2 种情况导致了 I/O 节点处的电容远大

于电路内部节点处的电容。假设  $C_{\text{int}} = 6$ ,  $N(\text{transitions}) = 2\,000$ ,  $C_{\text{ext}} = 2\,000$ ,  $N(\text{transitions})_{\text{I/O}} = 6$ , 此时芯片的功耗值为 24 000, 若其他值不变,  $N(\text{transitions})_{\text{int}} = 1900$ , 则功耗减少值为 600, 约减少 2.5%。同样若其他值不变,  $N(\text{transitions})_{\text{I/O}} = 5$ , 则功耗减少值为 2 000, 大约减少 8.3%。可以看出在外部和内部节点的平均电容一定的情况下,  $N(\text{transitions})_{\text{I/O}}$  对电路功耗的影响更大。

### 2 Bus-Invert 编码方法

Bus-Invert 编码方法(简称 B-I 编码)<sup>[5]</sup>是由 Stan 和 Burleson 提出的。此编码方法是在总线上附加一条额外的标志信号线 INV, 用来标志总线前后 2 段连续的时间内的信号翻转情况。这种编码方法通过判断  $t-1$  时刻总线上经过编码的数据(包括 INV)与时间  $t$  时刻总线上需要传输的数据之间的 Hamming 距离(即总线发生变化的比特数)。当 Hamming 距离 ( $H$ )大于  $N/2$  ( $N$  为总线宽度)时, 总线上的数据取反并将 INV 置 1; 否则情况相反。B-I 编码可表示为

$$(B^{(t)}, INV^{(t)}) = \begin{cases} (b^{(t)}, 0) & H^{(t)} \leq N/2 \\ (\bar{b}^{(t)}, 1) & H^{(t)} > N/2 \end{cases} \quad (4)$$

其中,  $H^{(t)} = (B^{(t-1)} \oplus INV^{(t-1)}) \cdot b^{(t)} \cdot 0$ 。

在接收端, 当 INV 为“1”时, 将数据取反接收; 当 INV 为“0”时, 数据不作任何处理。

但是这种编码主要有以下 2 个缺点: (1)这种编码适合在  $t$  时刻和  $t-1$  时刻传输的数据为随机出现的数据, 而当  $t$  时刻

**作者简介:** 尹立群(1979 -), 男, 博士研究生, 主研方向: 超大规模集成电路设计; 冯庆, 硕士; 袁国顺, 硕士、博士生导师

**收稿日期:** 2007-09-07 **E-mail:** yinlq@casue.com.cn

与  $t-1$  时刻传输的数据具有相关性,尤其是当前后数据是连续数据时,这种编码对于低功耗所起的作用就非常小;(2)连续数据间的 Hamming 距离是按二项式分布的<sup>[5]</sup>,当总线宽度变大时,Hamming 距离分布更集中于  $N/2$ ,此时总线编码的效率随着宽度的增加而降低。

### 3 针对 AMBA 总线的应用

#### 3.1 AMBA 总线系统简介

Advanced Micro-controller Bus Architecture(AMBA)总线是 ARM 公司推出的应用于片上系统的总线协议。典型的 AMBA 总线架构一般来说包含了 Advanced High-performance Bus(AHB)和 Advanced Peripheral Bus(APB)。AHB 总线负责连接例如 ARM processor, Color LCD Controller, On-chip Memory, External Memory Interface 或其他需要 high bandwidth 的元件。而 APB 总线则是用来连接系统的周边低速元件如 UART, Timer, Keypad, GP I/O 等。图 1 是典型的 AMBA 总线系统。其中, External Memory Interface(EMI)是 AHB 总线的一个 slave。ARM 处理器通过 EMI 对片外的 RAM 进行读写操作。本文主要针对此接口部分进行改进,从而达到降低此处节点翻转频率的目的。

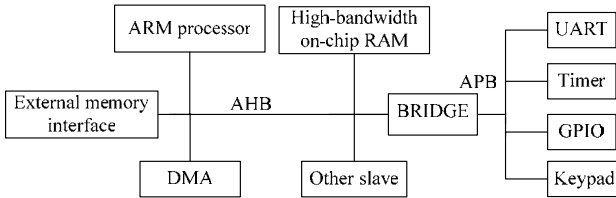


图 1 AMBA 总线系统

#### 3.2 根据 AMBA 总线协议对地址总线的编码

AHB 总线具有 2 级流水结构,一次完整的传输分为地址和数据 2 个阶段。地址阶段传输地址和控制信号,数据阶段传输读/写数据和反馈信号。AMBA 总线中规定 AHB 具有 4 种传输模式。它们分别是 IDLE, BUSY, NONSEQ(Non-sequential), SEQ(Sequential)。当采取 NONSEQ 传输方式时,总线上前后地址无序随机变化。而当采取 SEQ 传输时,总线上地址以一定的幅度连续变化。同时 SEQ 传输还分为 SINGLE, INCR 和 WRAP 3 种传输模式<sup>[6]</sup>。

因此在总线进行 NONSEQ 传输时对其进行 B-I 编码,当采取 SEQ 传输时,引入基于 Gray 编码思想的零翻转编码<sup>[7]</sup>。同样在总线上增加一条额外的标志信号线(INB)。当数据为不连续时,INB 置“0”,总线正常传输数据;当数据以固定幅度连续传输时,信号线 INB 置“1”,总线上除 INB 外的所有信号线全部锁死。因为在接收端收到 INB 为高的信号,所以接收端只需将上一数据与幅度值相加就可以得到当前的地址。因此当地址一直连续传输时,INB 就总是为“1”,总线始终处于锁定状态即处于零翻转状态。而在接收端根据接收到的 INB 电平高低来判断如何对地址进行解码。

渐进零翻转编码发送端可以表达为

$$(B^{(t)}, INB^{(t)}) = \begin{cases} (B^{(t-1)}, 1) & \text{if } t > 0 \text{ } b^{(t)} = b^{(t-1)} + S \\ (b^{(t)}, 0) & \text{otherwise} \end{cases} \quad (5)$$

在接收端的数据解码表示成:

$$b^{(t)} = \begin{cases} (b^{(t-1)} + S) & \text{if } INB = 1, t > 0 \\ B^{(t)} & \text{if } INB = 0 \end{cases} \quad (6)$$

其中,  $B^{(t)}$  为  $t$  时刻经编码后在接口处的地址输出;  $b^{(t)}$  为  $t$  时刻所要传输的实际地址;  $S$  为地址增加的幅度。

此方法大大减少了当地址总线在传输连续地址时的总线翻转率。进行改进后的 B-I 编码系统如图 2 所示。

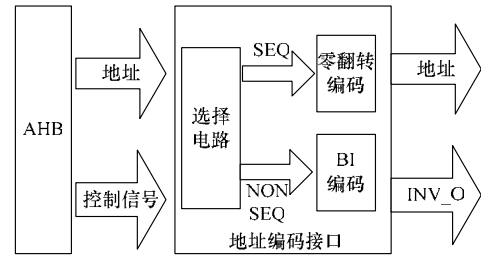


图 2 改进后的 B-I 编码系统

由于 AHB 总线地址传输模式由控制信号 HTRANS 决定。当进行 SEQ 传输时,无需对前后地址进行判断,直接对地址总线进行渐进的零翻转编码。因此,不必特意设计一块判断电路来判断前后地址是否连续,而只需对 HTRANS 信号进行判断,确定地址是否连续。这样既节省了编码电路的面积,也加快了编码速度。当然在 SEQ 传输中,还分为 SINGLE, INC 和 WRAP 传输(由 HBURST 信号决定),地址的增加幅度还根据 HSIZE(Transfer Size)不同而不同。因此,要根据控制信号的不同而采取不同的幅度值(表 1 所示)和不同的增长方式。

表 1 总线传输幅度

HSIZE/bit	幅度
8	1
16	2
32	4
64	8
128	16
256	32
512	64
1 024	128

将增长方式分为有界循环增加(即地址在一定的范围内循环增加)和无界增加(地址一直以一定幅度增加)。当采取 INC 传输时,采取无界增加模式;当采取 WRAP 传输时,采取有界循环增加模式。

图 3、图 4 为改进后的渐进零翻转发送端的编码电路和接收端的解码电路。在图 3 中,当 HTRANS 为 SEQ 传输时,INB 置“1”,同时输出地址 B 保持不变。当 HTRANS 为 NONSEQ 传输时,INB 置“0”,输出地址 B 等于输入地址 b。在接收端当收到 INB 为“1”时,地址自加并根据 BURST 和 HSIZE 信号来判断增加的类型和幅度。

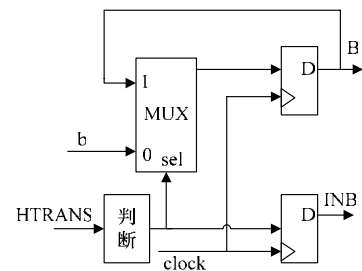


图 3 改进的渐进零翻转编码电路

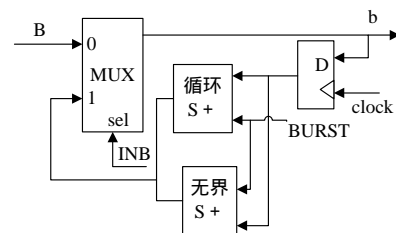


图 4 改进的渐进零翻转解码电路

由于总线还支持 NONSEQ 传输。此时地址随机变化, 渐进零翻转编码不起作用, 因此在这种情况下引入 B-I 编码。前面提到 B-I 编码对于宽总线效果不明显。因此, 在进行编码前, 要对地址总线先进行划分再编码。AHB 总线协议规定总线每次传输的地址宽度不能超过 1 Kb, 当超过 1 Kb 时要从起新的传输。可见地址变化最频繁的是地址总线的低十位。因此将前十位划分为一组进行 B-I 编码处理。编码电路如图 5 所示。

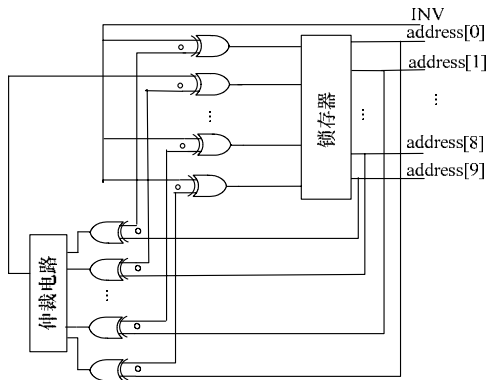


图 5 B-I 编码电路

在图 5 中, 仲裁电路由一组加法器组成, 当 Hamming 距离(前后地址的相异数)大于 5 时, INV 为“1”, 输出地址取反。反之则为“0”, 输出地址不变。B-I 接收端的解码电路相对简单(当接收端收到 INV 信号为“1”时, 地址取反, 反之不变), 在这里无需用电路图表示。

#### 4 仿真与比较

本设计采用 Verilog 硬件描述语言对各个模块进行寄存器级(RTL)的代码描述, 在 Verisilcon 公司的 0.18  $\mu\text{m}$  CMOS 标准单元工艺库支持下, 使用 Synopsys 公司的 Design Compiler 和 Power Compiler 工具进行逻辑综合并和功耗分析。

对测试程序产生的地址分别采用 B-I 编码、渐进零翻转编码和改进的 B-I 编码, 总线的翻转率如表 2 所示, 由结果可见, 使用 B-I 编码翻转率下降了 27.6%, 渐进零翻转编码翻转率下降了 47.6%, 改进的 B-I 编码翻转率下降了 56.3%。上述各种编解码器的面积如表 3 所示。可见各编码器的面积

(上接第 237 页)

幅值特性和相频特性的对比可见, 校正后的系统动态性能指标达到了系统的要求。

#### 3.2 时域性能分析

根据系统传递函数, 利用 Matlab 软件, 系统校正前(曲线 1)与校正后(曲线 2)的阶跃响应曲线如图 8 所示。

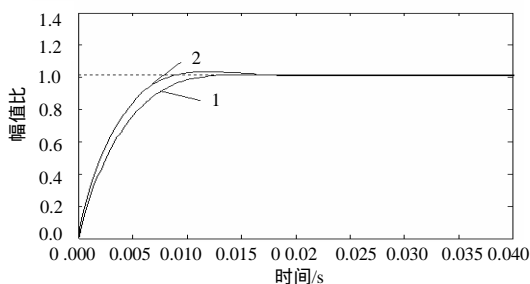


图 8 系统校正前后时域性能对比

都在同一数量级上。在系统芯片中, 编码器的面积可以忽略不计。

表 2 采用不同编码技术的总线翻转率 kHz

	原始地址	B-I 编码	渐进零翻转	改进 B-I 编码
程序	1 076	779	564	470

表 3 编码器的面积  $\mu\text{m}^2$

	B-I 编码	渐进零翻转	改进 B-I 编码
面积	12 860	28 714	31 085

#### 5 结束语

本文在分析了 B-I 编码的基础上, 针对 AMBA 总线的特点引入零翻转编码方法, 对原 B-I 编码进行了改进。克服了单纯用 B-I 编码的不足, 进一步降低了地址总线的翻转率。此方法不仅适用于 AMBA 总线系统, 也同样适用于其他对低功耗要求较高的 SoC 总线系统。

#### 参考文献

- [1] Weste N, Eshraghiam K. Principles of CMOS VLSI Design[M]. MA, USA: Addison-Wesley Publishing Company, 1998.
- [2] Chandrakasan A P, Sheng S, Brodersen R W. Low-power CMOS Digital Design[J]. IEEE J. Solid-State Circuit, 1992, 27(4): 473-484.
- [3] Devadas S, Keutzer K, White J. Estimation of Power Dissipation in CMOS Combinational Circuits[C]//Proc. of IEEE Custom Integrated Circuits Conf.. Boston, MA, USA: [s. n.], 1990: 1971-1976.
- [4] Chandrakasan A P, Shen S, Brodersen R W. Low-power Techniques for Portable Real-time DSP Application[C]//Proc. of the 5th International Conference on VLSI Design. Bangalore, India: [s. n.], 1992: 203-208.
- [5] Stan M R, Burlison W P. Bus-Invert Coding for Low-Power I/O[J]. IEEE Transactions on VLSI Systems, 1995, 3(1): 49-58.
- [6] ARM Corporation. AMBA Specification(Rev 2.0)[EB/OL]. (1999-05-24). <http://www.arm.com>.
- [7] Benini L, Micheli G D, Macii E, et al. Asymptotic Zero-transition Activity Encoding for Address Busses in Low-power Microprocessor-based Systems[C]//Proc. of the 7th Great Lakes Symposium on VLSI. Urbana Champaign, IL, USA: [s. n.], 1997: 77-82.

通过对比, 校正后系统的超调量稍许增加, 上升时间缩小, 稳定时间几乎不变, 系统的响应速度和稳态精度几乎不受损失。

#### 参考文献

- [1] 黄俊钦. 测试系统动力学[M]. 北京: 国防工业出版社, 1996.
- [2] 段广云. 微位移传感器的简便动态模型识别[J]. 青海大学学报, 2007, 25(2): 61-64.
- [3] 罗艳蕾, 王 玥, 刘正雷. 一种用 MATLAB 优化工具进行控制系统设计的方法[J]. 贵州工业大学学报, 2005, 34(6): 47-49.
- [4] 董景新, 赵长德. 控制工程基础[M]. 北京: 清华大学出版社, 2001.
- [5] 张红莲. 基于 MATLAB 的控制系统校正环节的优化设计[J]. 自动化与仪器仪表, 2000, 21(2): 22-23, 27.

