

# LS-DSP 数字信号处理器总线的低功耗设计\*

车德亮<sup>1</sup> 李剑川<sup>1,2</sup> 沈绪榜<sup>1</sup>

( 1. 西安微电子技术研究所 陕西 西安 710054 ; 2. 国防科技大学 机电工程与自动化学院 湖南 长沙 410073 )

**摘要** 随着数字信号处理器制造工艺的进步,总线的功耗已经成为数字信号处理器功耗的主要组成部分。研究了 LS-DSP 数字信号处理器地址总线、数据总线、内部数据总线三类总线的低功耗设计技术,主要采用了总线编码的方法实现总线的低功耗。实验结果表明,LS-DSP 采用总线低功耗技术后,有效地降低了总线的功耗。

**关键词** 总线编码 海明距离 串扰 数字信号处理器

**中图分类号** TP302 **文献标识码** A

## The Low Power Design of the Bus for Digital Signal Processor LS-DSP

CHE De-liang<sup>1</sup> ,LI Jian-chuan<sup>1,2</sup> ,SHEN Xu-bang<sup>1</sup>

( 1. Xi 'an Microelectronics Technology Institute , Xi 'an 710054 ,China ;

2. College of Mechatronics Engineering and Automation , National Univ. of Defense Technology , Changsha 410073 ,China )

**Abstract** With the development of fabricate technology for digital signal processo( DSP ), the power dissipation of the bus is the main part of the whole power dissipation in DSP. This paper researches the power saving ways for address bus, data bus, internal bus in LS-DSP . Bus coding technology has been used for that purpose. The experimental results show that the LS-DSP bus power can reduce effectively when it uses the power saving ways discussed in this paper are used.

**Key words** bus coding ;Hamming distance ;cross talk ;DSP

LS-DSP 是面向航天数字信号处理系统而开发的 32 位嵌入式数字信号处理器。由于航天数字信号处理系统的特殊运行环境,LS-DSP 的低功耗设计成为设计成败的关键。文献 [1] 的研究表明 VLSI CMOS 制造工艺特征尺寸发展到深亚微米后,总线的功耗将占到数字信号处理器功耗的 70%。因此,总线低功耗技术研究已成为数字信号处理器低功耗研究的重点。根据图 1 的 LS-DSP 体系结构,按总线的功能和位置进行划分,可将 LS-DSP 的总线分为地址总线、数据总线和内部数据总线三类。

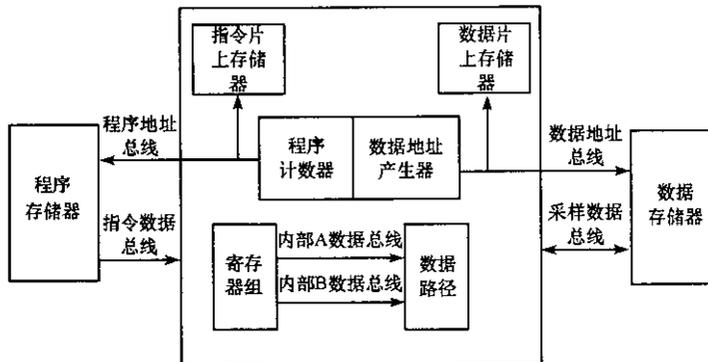


图 1 LS-DSP 的体系结构

Fig.1 The mainframe of LS-DSP

CMOS 集成电路的功耗由动态功耗、短路功耗、静态功耗三部分组成,其中功耗的主体是动态功耗

\* 收稿日期 2004 - 09 - 29

基金项目 国家 863 高技术研究发展计划资助项目( 2002AA714022 ) , 国家部委资助项目( 41308010203 )

作者简介 车德亮( 1975— ) , 男, 博士生。

( $P_d$ ) 其值可用集成电路中门的功耗表示如下:

$$P_d = \sum_{i=1}^N \frac{1}{2} C_i V_{dd}^2 f_i \quad (1)$$

其中,  $C_i$  为第  $i$  个门的输出电容,  $V_{dd}$  是电源电压,  $f_i$  是节点  $i$  的翻转频率,  $N$  是集成电路中门的总数。总线的功耗也可用 (1) 式估算, 由于总线通常是集成电路与外部系统的接口, 其  $C_i$  值通常比片内门  $C_i$  值大几个数量级, 因此具有大负载电容的总线功耗成为 CMOS 集成电路功耗的主体。以往认为总线的负载电容值相对稳定, 通常只考虑通过减少总线的  $\sum f_i$  来减小总线的功耗。文献 [1] 的研究表明总线内相邻信号间可变的串扰电容 ( $C_c$ ) 所引起的功耗是总线功耗的重要组成部分, 因此在进行总线低功耗设计时应综合考虑减少总线的  $\sum C_c f_i$ , 其中总线的翻转频率  $f_i$  以总线上连续信号值的海明距离 (Hamming distance) 来度量,  $C_c$  以总线内相邻信号向相反方向跳变的次数度量。

本文是通过增加硬件电路实现减小 LS-DSP 总线的  $\sum C_c f_i$ , 采用的技术有格雷码 (gray code) 编码技术<sup>[2,4,6]</sup>、反向编码技术 (Bus Invert Coding)<sup>[1,5]</sup> 和总线位序重排技术<sup>[2]</sup>。

## 1 LS-DSP 地址总线的低功耗设计

LS-DSP 的地址总线包括程序地址总线与数据地址总线。程序地址总线上相继变化的值是数字信号处理应用程序中指令的地址, 应用程序中 90% 的指令是顺序执行, 即程序地址总线上的值主要是顺序变化<sup>[1]</sup>; 数据地址总线上相继变化的值是 DSP 处理数据的地址, 在数字信号处理计算过程中, 被处理的采样数据通常放入一个循环缓冲区中, 在循环缓冲区内采样数据是顺序访问, 因此数据地址总线上的值也主要是顺序变化。

对总线值顺序变化的低功耗技术主要有 T0 编码技术与格雷码编码技术。T0 编码技术可实现地址顺序变化时地址总线  $\sum C_c f_i$  为 0, 但其编码电路与解码电路开销较大, 同时需要一条由总线始端到末端含时序信息的控制线, 当 CMOS 电路采用深亚微米设计时, 线延迟作用明显, T0 编码技术容易引起总线的误动作。LS-DSP 拟采用 0.18  $\mu\text{m}$  CMOS 工艺制造, 因此本文以格雷码编码技术进行 LS-DSP 地址总线的低功耗设计。

格雷码序列的定义如下:

如果  $N_i$  表示  $i$  位格雷码序列

1 位格雷码序列,  $N_1 = 0, 1$ ;

$k$  位格雷码序列,  $N_k = n_0, n_1, \dots, n_{2^k-2}, n_{2^k-1}$ ;

那么,  $k+1$  位格雷码序列,  $N_{k+1} = 0n_0, 0n_1, \dots, 0n_{2^k-2}, 0n_{2^k-1}, 1n_{2^k-1}, 1n_{2^k-2}, \dots, 1n_1, 1n_0$ ;

由格雷码序列的定义可知, 在格雷码序列中任意两个相邻格雷码的海明距离为 1, 并且格雷码内不会发生相邻位向相反方向跳变的情况。如果用顺序变化的格雷码表示顺序变化的二进制地址, 则地址总线在任意二个顺序地址之间海明距离为 1, 并且地址总线内相邻位的串扰为 0。  $N$  位二进制数与  $N$  位格雷码之间的转换算法如下:

$B$  表示  $N$  位二进制数,  $B = (b_{n-1}, b_{n-2}, \dots, b_1, b_0)_2$ ;

$G$  表示  $N$  位的格雷码,  $G = (g_{n-1}, g_{n-2}, \dots, g_1, g_0)_g$ ;

二进制数转换为格雷码 ( $b2g$ ):  $g_n = 0, g_j = b_{j+1} \oplus b_j$  ( $j = (n-1), \dots, 0$ );

格雷码转换为二进制数 ( $g2b$ ):  $b_n = 0, b_j = b_{j+1} \oplus g_j$  ( $j = (n-1), \dots, 0$ );

当地址总线上顺序变化的二进制值步进为 1 时, 表示地址总线上顺序变化的二进制值是字节地址, 直接采用转换算法就可达到任意二个顺序地址之间只有 1 位翻转, 并且地址总线内相邻位的串扰为 0。可是, LS-DSP 地址总线上还存在顺序变化的字地址或双字地址, 即二进制值还存在步进为 2 或为 4 的情况, 此时若直接采用转换算法, 总线上顺序变化的二进制值对于变换后的格雷码是不连续的, 此时总线顺序地址之间海明距离大于 1, 并且产生地址总线内相邻位的串扰。为了解决此问题, 需要在  $N$  位二进制数与  $N$  位格雷码之间的转换算法中考虑步进的影响。具体的解决方法是将地址值分成高低两部分, 两部分划分的界限由步进 ( $i$ ) 决定, 高低地址值分别用转换算法转换后, 再合成为一个地址值。本文

用  $g[n-1:i]$  表示步进为  $i$  的  $n$  位的格雷码,  $b[n-1:i]$  表示步进为  $i$  的  $n$  位的二进制地址值,  $b2gh$  表示二进制地址值高位部分转化为格雷码高位部分,  $g2bl$  表示格雷码低位地址部分转化为二进制地址低位部分, 则改进的  $N$  位二进制数与  $N$  位格雷码之间的转换算法如下:

$B$  表示  $N$  位二进制数,  $B = (b_{n-1}, b_{n-2}, \dots, b_1, b_0)_2$ ;

$G$  表示  $N$  位的格雷码,  $G = (g_{n-1}, g_{n-2}, \dots, g_1, g_0)_g$ ;

当步进  $i \neq 1$  且  $i = 2^k$ :

二进制数转换为格雷码

$$g[n-1:k] = b2gh(b[n-1:k]), g[k-1:0] = b2g(b[k-1:0]);$$

格雷码转换为二进制数

$$b[n-1:k] = g2bh(g[n-1:k]), b[k-1:0] = g2b(g[k-1:0]);$$

表 1 是 4 位地址值应用改进的转换算法, 对不同步进情况下的格雷码编码方案, 从表 1 中易得出, 采用不同步进的格雷码编码方案, 可实现总线顺序地址之间海明距离为 1, 并且无地址总线内相邻位的串扰。

表 1 4 位二进制地址对不同步进情况下的格雷码编码方案

Tab. 1 Gray code mapping for different steps

4 位二进制地址的十进制值	4 位二进制地址值	格雷码编码方案				
		步进 $i = 1$	步进 $i = 2$		步进 $i = 4$	
		$H$	$H$	$L$	$H$	$L$
0	0000	0000	000	0	00	00
1	0001	0001	000	1	00	01
2	0010	0011	001	0	00	11
3	0011	0010	001	1	00	10
4	0100	0110	011	0	01	00
5	0101	0111	011	1	01	01
6	0110	0101	010	0	01	11
7	0111	0100	010	1	01	10
8	1000	1100	110	0	11	00
9	1001	1101	110	1	11	01
10	1010	1111	111	0	11	11
11	1011	1110	111	1	11	10
12	1100	1010	101	0	10	00
13	1101	1011	101	1	10	01
14	1110	1001	100	0	10	11
15	1111	1000	100	1	10	10

## 2 LS-DSP 数据总线的低功耗设计

LS-DSP 数据总线包括指令数据总线与采样数据总线, 数据

总线上相继出现的数据值是随机的, 数据值之间取值是相互独立的并且取任何值的概率相同。因此, 数据总线上出现的数据值可看做服从均匀分布(uniformly distributed)的随机序列, 即在单位时间片内,  $n$ (不妨设  $n$  为偶数)位宽的数据总线上有可能出现的数据值有  $2^n$  个, 并且这  $2^n$  个有可能出现的数据值出现的概率相同。根据数据总线上数据的分布特性可知, 单位时间片内  $n$  位宽的数据总线上数值变化的平均海明距离为  $n/2$ ,  $n$  位宽的数据总线的平均功耗与  $n/2$  成正比; 单位时间片内  $n$  位宽的数据总线上峰值功耗与  $n$  成正比, 其发生的概率为  $1/(2^n)$ 。根据以上数据总线功耗特点的分析, 在 LS-DSP 数据总线的低功耗设计中, 本文研究总线反向编码技术降低数据总线的平均海明距离和峰值时的海明距离。

用  $n$  表示数据总线宽度; 用 Data\_value 表示数据总线上将要传送的值; 用 Bus\_value 表示当前数据总线上的值; 用 inv

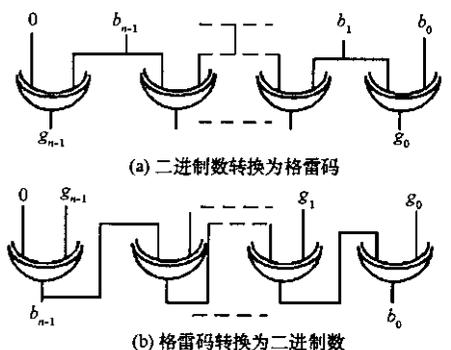


图 2 二进制数与格雷码的相互转换  
Fig. 2 Binary and gray code conversions

inv\_ctrl 表示反向编码控制信号,如果 inv\_ctrl 为 0,则 Data\_value 的值直接由数据总线传输,如果 inv\_ctrl 为 1,则 Data\_value 的值按位反向后,再由数据总线传输;用 HMD 表示海明距离计算函数,用 INVERT 表示按位求反函数。根据以上的约定,反向编码算法(Inv\_code Algorithm)如下:

```

Inv_code Algorithm :if( HMD( Data_value ,Bus_value )> n/2 )
    begin
        inv_ctrl = 1 ;
        Bus_value = INVERT( Data_value );
    end
else
    begin
        inv_ctrl = 0 ;
        Bus_value = Data_value ;
    end
end
    
```

根据反向编码算法,容易得出,数据总线采用反向编码后,单位时间片内  $n$  位宽的数据总线上峰值海明距离可减少一半,即数据总线的峰值功耗与  $n/2$  成正比。采用反向编码技术对  $n$  位宽数据总线平均海明距离的影响分析如下:Data\_value 经过反向编码算法优化后与 Bus\_value 的海明距离为 0 的可能性是  $C_{n+1}^0$  ( $n+1$  为  $n$  位宽的数据总线与 1 位的反向编码的控制线 inv\_ctrl);Data\_value 经过反向编码算法优化后与 Bus\_value 的海明距离为 1 的可能性是  $C_{n+1}^1$  (其包括  $HMD(Data\_value, Bus\_value) = 1$  且  $inv\_ctrl = 0$  的  $n$  种情况与  $HMD(Data\_value, Bus\_value) = n$  且  $inv\_ctrl = 1$  的 1 种情况);依次类推,Data\_value 经过反向编码算法优化后与 Bus\_value 的最大海明距离为  $n/2$ ,其出现的可能性为  $C_{n+1}^{n/2}$ 。 $n$  位宽数据总线的 Data\_value 值共有  $2^n$  种,采用反向编码技术后与 Bus\_value 的平均海明距离

$$HD_{average} = \frac{C_{n+1}^0 \cdot 0 + C_{n+1}^1 \cdot 1 + C_{n+1}^2 \cdot 2 + \dots + C_{n+1}^{n/2} \cdot n/2}{2^n} \tag{2}$$

$$C_{n+1}^0 + C_{n+1}^1 + C_{n+1}^2 + \dots + C_{n+1}^{n/2} = 2^n \tag{3}$$

参考式(3),容易得出式(2)的值小于  $n/2$ ,即说明采用反向编码技术可有效减小数据总线的平均海明距离。数据总线采用反向编码的电路结构如图 3 所示。

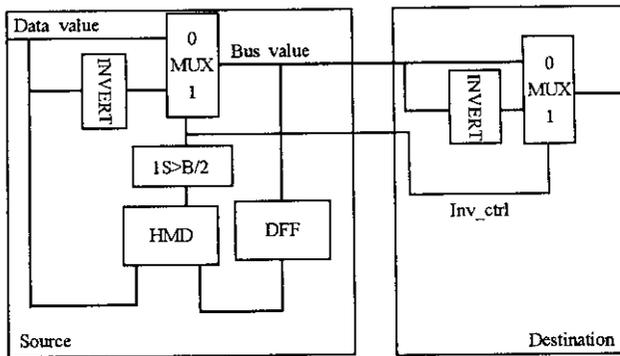


图 3 数据总线反向编码的电路结构

Fig.3 The architecture of bus-invert coding circuit

### 3 LS-DSP 内部数据总线的低功耗设计

LS-DSP 的内部数据总线主要指为数据路径 (ALU 等) 提供操作数的 A、B 内部数据总线。A、B 内部数据总线可用反向编码技术降低其功耗,但鉴于内部数据总线功耗组成的特点(内部数据总线总功耗中的串扰功耗比重远大于外部数据总线总功耗中的串扰功耗比重),本文研究 A、B 内部数据总线对应位交换以减少 A、B 内部总线  $\sum C_f f_i$  的方法。

LS-DSP 的 ALU 操作包括 ADD、AND、OR、XOR 等,这些操作的共同点是满足位交换率,即两个操作数的对应位置上的数据可任意交换,操作的结果值不变。例如,两个 4 位二进制数  $A(a_3 a_2 a_1 a_0)$  与  $B(b_3 b_2 b_1 b_0)$  相加的计算表达式可有以下几种:

$$\begin{aligned} a_3 a_2 a_1 a_0 + b_3 b_2 b_1 b_0 &= b_3 a_2 a_1 a_0 + a_3 b_2 b_1 b_0 \\ &= a_3 b_2 a_1 a_0 + b_3 a_2 b_1 b_0 \\ &= a_3 a_2 b_1 b_0 + b_3 b_2 a_1 a_0 \\ &= \dots \end{aligned}$$

可以利用操作满足位交换率的性质,减少内部数据总线的功耗。例如,连续的两个加法操作:  $Y_1 = A(0011) + B(1100)$ ,  $Y_2 = A(0100) + B(1011)$ , 计算  $Y_2$  时, A 总线上相继两个数据的海明距离为 3 (位  $a_2: 0 \rightarrow 1$ , 位  $a_1: 1 \rightarrow 0$ , 位  $a_0: 1 \rightarrow 0$ ), 并且有一对相邻信号 ( $a_2 - a_1$ ) 发生相反方向的翻转, 同时 B 总线上相继两个数据的海明距离也为 3 (位  $b_2: 1 \rightarrow 0$ , 位  $b_1: 0 \rightarrow 1$ , 位  $b_0: 0 \rightarrow 1$ ), 并且有一对相邻信号 ( $b_2 - b_1$ ) 发生相反方向的翻转, 即计算  $Y_2$  时, 内部数据总线的总体代价是内部数据总线变化的海明距离为 6, 并且有 2 次相邻位的串扰。利用加法满足位交换率的性质, 将  $Y_2$  中两个操作数的 2、1、0 位数据进行交换, 即  $Y_2 = A(0100) + B(1011) = A(0011) + B(1100)$ , 再计算  $Y_2$  时, A、B 内部数据总线变化的海明距离为 0, 并且无相邻位的串扰。利用对应位交换方法减少内部总线功耗的一种电路实现方式如图 4 所示。

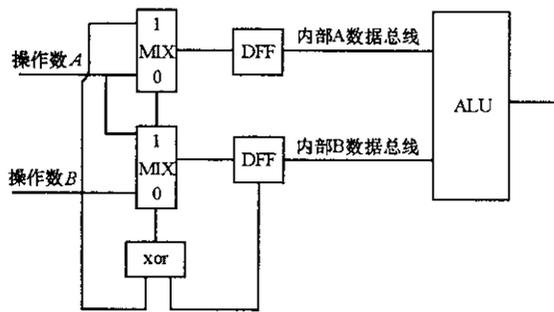


图 4 内部数据总线对应位交换电路结构

Fig.4 The architecture of internal data bus swap circuit

## 4 实验与分析

### 4.1 实验一

实现总线低功耗的硬件电路的门功耗模型引用 VeriSilicon SMIC 0.18 $\mu\text{m}$  High-Density Standard Cell Library Databook (version 2.1, 2004-01-13) 中的功耗参数, LS-DSP 的工作频率为 100MHz, 内核电压 1.8V, 其内门的平均负载电容为 0.1pF, LS-DSP 总线位宽为 32 位, 则其三种总线低功耗电路的功耗统计值如表 2 所示。

表 2 总线低功耗逻辑的功耗

Tab.2 Bus low power logical dissipation

电路类型	最大动态功耗( $\mu\text{W}$ )	静态功耗( $\text{nW}$ )
地址总线低功耗电路	102.912	12.86
数据总线低功耗电路	647.019	78.769
内部数据总线低功耗电路	417.537	56.16

分别对没有总线低功耗逻辑和有总线低功耗逻辑的 LS-DSP 的物理版图进行仿真后,执行 FIR 滤波、FFT 变换、矩阵乘三种不同处理算法。根据文献 [1] 的研究,总线整体的功耗按数字信号处理器功耗(芯片功耗)的 70% 进行估算,得到总线整体功耗变化情况如表 3 所示。

表 3 增加总线低功耗逻辑后的 LS-DSP 总线功耗变化情况

Tab.3 The effect of power for LS-DSP bus with bus low power logical

LS-DSP 执行的 处理算法名称	没总线低功耗逻辑的 LS-DSP 功耗(W)		有总线低功耗逻辑的 LS-DSP 功耗(W)		LS-DSP 总线整体 功耗减少率
	芯片功耗	总线功耗	芯片功耗	总线功耗	
FIR 滤波	1.75	1.225	1.73	1.211	1.1%
FFT 变换	2.16	1.512	2.04	1.428	5.56%
矩阵乘	1.86	1.302	1.81	1.267	2.7%

实验一是从整体出发,采用对比的方法验证总线低功耗逻辑的作用,从实验一的结果可以得到以下两个结论:

(1) LS-DSP 采用总线低功耗逻辑后,总线功耗明显下降。由于总线结构、处理算法、被处理数据集的特点对总线低功耗逻辑的效果都有着重要的影响,因此表 3 的结果只是定性反映了总线低功耗逻辑的作用。可预见,当总线负载较大、处理算法中多包含可进行对应位交换的运算、被处理数据集的均方差较大时,总线低功耗逻辑的优化效果会明显提高。

(2) 在以深亚微米设计制造的数字信号处理器内,增加总线低功耗逻辑,不会引起数字信号处理器功耗的显著增加。表 2 的结果表明,LS-DSP 内总线低功耗逻辑的最大动态功耗为  $1\ 168\ \mu\text{W}$ ,与 LS-DSP 的处理功耗相差 3 个数量级,所占比重很小。可预见,当以深亚微米设计制造的数字信号处理器整体设计集成度越高,工作频率越快时,总线低功耗逻辑的功耗可忽略不计。

#### 4.2 实验二

表 4 为 LS-DSP 顺序变化的数据地址总线值采用格雷码地址后,对数据地址总线功耗的影响。

表 4 格雷码地址减小 LS-DSP 数据地址总线功耗的效果

Tab.4 The effect of gray address for low power LS-DSP address bus

数据区长度	二进制地址		格雷码地址		功耗减少率
	海明距离	串扰次数	海明距离	串扰次数	
128	250	63	127	0	49.2%
256	504	130	255	0	49.4%
512	1 007	267	511	0	49.3%
1024	2 034	512	1 023	0	49.7%

表 5 为 LS-DSP 的数据总线和内部数据总线增加反向编码和对应位交换逻辑后,分别传送 500 个随机数和 1 000 个随机数时,两类总线的功耗变化情况。

表5 LS-DSP 两类数据总线低功耗电路的效果

Tab.5 The effect of low power circuit for LS-DSP data bus

总线类型	随机数量	不含总线低功耗逻辑		含有总线低功耗逻辑		总线功耗 减少率
		海明距离	串扰次数	海明距离	串扰次数	
数据总线	500	5 072	1 183	4 322	1 029	14.8%
	1 000	11 905	2 743	10 679	2 413	10.3%
内部数据总线	500	7 883	1 397	5 873	531	25.5%
	1 000	15 841	3 617	11 753	1 410	25.8%

试验二是从局部出发,采用直接验证的方法,观测三种总线低功耗逻辑的作用。从实验二的结果可以得到以下结论:

(1)LS-DSP 地址总线、数据总线、内部数据总线三类总线的低功耗设计技术可有效减少 LS-DSP 总线的功耗。采用格雷码地址可减少地址总线 49%的功耗;采用反向编码可减少数据总线 10%的功耗;采用对应位交换技术可减少内部数据总线 25%的功耗。

(2)实验二的结果是三类总线的低功耗设计技术的降低总线功耗的理想效果,其值基本与数学估算的理论值相符。

## 5 结束语

讨论了 LS-DSP 地址总线、数据总线、内部数据总线三类总线的低功耗设计技术,通过整体与局部两类实验证实,本文的总线低功耗方法对降低总线的功耗是有效的。

## 参考文献:

- [1] Sungpack Hong, Umni Narayanan, Ki-Seok Chung, et al, Bus-Invert Coding for Low-Power I/O-A Decomposition Aaraoach[J]. IEEE Midwest Symp on Circuits and Systems, Lansing MI, Aug 8 - 11, 2000.
- [2] Mehendale M, Sherlekar S D, Venkatesh G. Extensions to Programmable DSP Architectures for Reduced Power Dissipation[J]. International conference on VLSI Design, VLSI Design, 1998, 37 - 42.
- [3] Benini L, et al, Asymptotic Zero Transition Activity Encoding for Address Busses in Low-power in Processors for Signal Processing[J]. IEEE Transactions on VLSI systems, December 1997, 417 - 424.
- [4] Famehta H, et al. Some Issues in Gray Code Addressing[J]. Gls-VLSI '96, 6th Great Lakes symposium on VLSI, 1996, 178 - 181.
- [5] Stan M R, Burleson W P. BUS Invert Coding for Low Power I/O[J]. IEEE Transactions on VLSI systems, MARCH 1995, 49 - 58.
- [6] Ching-Long SU, Chi-ying tsui, Alvin M D. Saving Power in the Control Path of Embedded Processors[J]. IEEE Design and Test of Computers, Winter 1994, 24 - 30.
- [7] Panda P R, Dutt N D. Reducing Address Bus Transitions for Low Power Memory Mapping[J]. European Design and Test Conference, ED&TC, 1996.
- [8] Mehendale M, Sherlekar S D, Venkatesh G. Coefficient Optimization for Low Power Realization of FIR Filters[J]. IEEE Workshop on VLSI Signal Processing, 1995.
- [9] Zhu D, Li T, Wan H, et al. Automatic Circuit Extraction Using Program Slicing[J]. J. of National Univ. of Defense Technology, 25(6) 2003.



