

文章编号:1001-5132 (2008) 04-0496-05

# 低功耗 XOR 门的快速分解技术

许 亮, 夏银水

(宁波大学 信息科学与工程学院, 浙江 宁波 315211)

摘要:为了在相对短的时间内得到基于 XOR 门电路的最优结构,并达到功耗优化的目的,以 XOR 门输入信号的概率为依据进行低功耗分解,提出了一种新的基于 XOR 门的电路功耗优化技术. 实验结果表明:提出的算法能在更短的时间内实现功耗优化,且比现有的方法最多可提高 8.9%,同时也证明了提出的算法在功耗节省方面比其他同类算法更有效.

关键词:功耗优化技术; Reed-Muller 逻辑电路; XOR 门

中图分类号: TN47

文献标识码: A

集成技术的迅速发展使得在单块芯片上所能集成的电路规模越来越大,运算速度也越来越快. 同时移动设备对低功耗芯片的旺盛需求,使得低功耗设计技术成为 VLSI 设计的热点之一. CMOS 由于本质上的低功耗,已成为集成电路的基准工艺. 而对 CMOS 电路来说,功耗主要是由于电路节点的信号翻转引起对节点电容充放电所造成的<sup>[1]</sup>,其功耗可以由下式决定:

$$Power = \frac{1}{2} V_{dd}^2 f_{clk} \sum_{i=1}^n C_i E_i, \quad (1)$$

其中,  $V_{dd}$  为电源电压;  $f_{clk}$  为时钟频率;  $V_{dd}$  及  $f_{clk}$  是常量,由所采用的工艺技术决定;  $i$  为电路中输出结点的个数;  $C_i$  为输出结点的负载电容;  $E_i$  代表每个时钟周期输出信号的平均跳变次数,称为开关活动性.

目前的低功耗技术往往是基于传统的布尔逻辑来实现,所有 IC 设计工具均是在此基础上发展

起来的<sup>[2]</sup>. 然而局限于传统布尔逻辑的集成电路设计技术面临着诸如面积、速度、功耗及可测试性四大挑战,因而迫切需要发展新的综合技术.

研究表明,采用了基于与异或的 Reed-Muller (RM)逻辑表达式可以对电路面积进行优化<sup>[3]</sup>. 近年来, RM 逻辑电路的低功耗技术正逐渐成为低功耗技术研究的新热点. Narayanan and Liu 提出了一种基于固定极性下 RM 函数(FPRM)功耗优化的算法<sup>[4]</sup>, Zhou and Wong 则分别针对固定极性下 AND 门 XOR 门的分解提出了 2 种功耗优化算法<sup>[5,6]</sup>,这些研究都开创了对 RM 逻辑电路功耗优化技术的研究. 然而,这些技术远没有达到满意的程度.

将多输入 XOR 门分解成 2 输入 XOR 门的树是技术映射前的重要一步. 发展多输入 XOR 门分解的低功耗技术,对于低功耗 XOR 门逻辑电路的获得具有重要的作用. 文献[7]讨论了动态逻辑的多输入 XOR 门分解的低功耗技术,文献[8]讨论了

静态逻辑的多输入 XOR 门分解的低功耗技术. 由于需要搜索所有  $2^{n-1} - 1$  种偶极性, 耗费的时间较多, 本文在文献[4]算法的基础上, 提出一种新的低功耗 XOR 门的快速分解方法.

### 1 相关研究

定义 1 低功耗 XOR 优化树: 将 1 个  $n$  输入的 XOR 门分解成  $n-1$  个 2 输入 XOR 门组成的电路结构, 其中功耗最小的结构称为低功耗 XOR 优化树.

定义 2 XOR 门分解树: 设  $T=(V, E)$  表示 1 棵树,  $p_r(v)(v \in V$  为树的结点), 代表每个 2 输入 XOR 门的输出信号  $v$  的信号概率; 内部结点间的连接关系称为树枝; 多输入 XOR 门的输入变量称为树叶, 输出变量称为树根.

定义 3 如果信号  $x(t)$  为 0~1 随机信号, 在连续  $N$  个周期( $N \times T$ )内, 在  $S$  个周期取 1, 那么该信号  $x(t)$  的信号概率为:  $p_r(v) = \lim_{N \rightarrow \infty} (S/N) \times 100\%$ . 假设每个 2 输入 XOR 门的  $C_i$  相同, 由式(1)可知, 需要计算的只有  $E_i$  参数. 对于在时间和空间上都独立的零延时模型电路, 已知 1 棵 XOR 分解树和初始输入信号概率, 就可计算出树中所有内部结点信号的开关活动性.

定义 4 开关活动性的估算: 设  $z = x \oplus y = \overline{xy} + \overline{\overline{xy}}$ ,  $x$  和  $y$  为输入变量, 则:

$$p_{r,z} = p_r(p_{r,x}, p_{r,y}) = p_{r,x} + p_{r,y} - 2p_{r,x}p_{r,y}. \quad (2)$$

文献[5]推导出由信号概率求开关活动性的方法, 其中对于动态逻辑电路, 结点  $i$  的开关活动性计算为式(3):

$$E_i = 2p_{r,i}. \quad (3)$$

总开关活动性计算为式(4):

$$E_T^{sw} = \sum_{i \in V-1} 2p_{r,i}. \quad (4)$$

对于静态逻辑电路, 结点  $i$  的开关活动性计算为式(5):

$$E_i = 2p_{r,i}(1 - p_{r,i}). \quad (5)$$

总开关活动性计算为式(6):

$$E_T^{sw} = \sum_{i \in V-1} 2p_{r,i}(1 - p_{r,i}), \quad (6)$$

式中,  $p_{r,i}$  为内部结点的信号概率.

对于动态和静态两种逻辑电路而言, 由式(3)和式(5), 可得如图 1 所示的图形. 图 1 中虚线表示动态逻辑, 实线表示静态逻辑, 从中可看出动态逻辑电路的开关活动性随着信号概率的增大而增大. 而对于静态逻辑电路, 信号概率越接近 0.5, 其开关活动性越大.

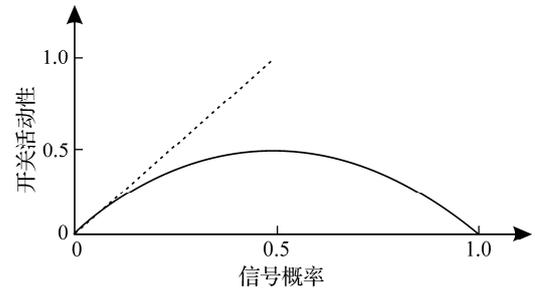


图 1 开关活动性与信号概率的关系

基于以上所述, 给定 1 个  $n$  输入的 XOR 门, 设其输入信号为  $I = \{x_{n-1}, x_{n-2}, \dots, x_0\}$ , 相应的信号概率为  $\{p_{r,x_{n-1}}, p_{r,x_{n-2}}, \dots, p_{r,x_0}\}$ , XOR 分解树的功耗估算过程如下:

- (1) 根据输入信号概率, 构造 1 棵 XOR 分解树  $T=(V, E)$ ,  $\{p_{r,x_{n-1}}, p_{r,x_{n-2}}, \dots, p_{r,x_0}\}$  为其树叶;
- (2) 根据式(2)计算其中每个 XOR 门的输出信号概率;
- (3) 根据式(3)或式(5)计算每个 XOR 门输出结点的开关活动性;
- (4) 根据式(4)或式(6)计算总的开关活动性.

根据上述功耗估算方法, 通过相应的搜索算法, 就可以得到低功耗的优化树. 研究者们已提出了一系列的算法, 其中较有代表性的算法如下.

Min-Huffman 算法: 对于所有的输入信号概率, 结合概率最小的 2 个信号, 用产生信号代替它们, 重复此过程<sup>[3]</sup>.

Narayana and Liu 的算法: 设  $k$  为信号概率大于 0.5 的变量个数, 若  $k$  为偶数, 将此  $k$  个信号分别取反; 若  $k$  为奇数, 将其中任意  $k-1$  个信号取反. 然后将得到的新的信号概率用 Min-Huffman 法构

造 XOR 树<sup>[4]</sup>.

Zhou and Wong 的算法: 首先找出信号概率大于 0.5 的输入信号, 然后由大到小两两组合, 最后用 Min-Huffman 算法构造 XOR 树<sup>[5]</sup>.

对于上述 3 种算法, Narayana and Liu 和 Zhou and Wong 的算法实际上是 Min-Huffman 算法的改良, 为比较上述 3 种算法的区别, 下面分别将其应用于同 1 个例子来看具体效果. 设 3 个输入信号概率分别为 0.1、0.45 和 0.9, 如图 2 所示; 3 种算法所生成的低功耗 XOR 优化树的结构相同, 如图 3 所示. 如果是动态逻辑电路, 可计算出该结构的开关活动性为 1.984(若为静态逻辑电路, 可计算出该结构的开关活动性为 0.994 8). 然而发现, 如果输入信号及其取反后的信号概率均已知, 则上述 3 种算法所生成的优化树结构并不是最优的. 例如, 若将  $x_1$  和  $x_2$  取反, 如图 4 所示, 则其逻辑功能不变, 在动态逻辑电路中, 它的总开关活动性却减小到 1.424, 在静态逻辑电路中, 可计算出它的开关活动性减小为 0.793 2. 由此可见, 现存的算法并非最优的, 还有待改进.

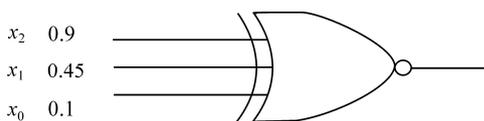


图 2 多输入 XOR 门

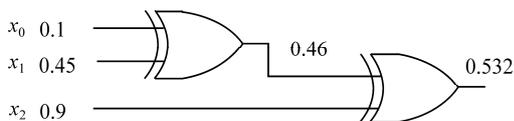


图 3 已提出的分解方法

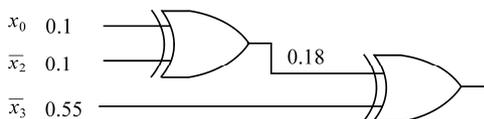


图 4 一种新的分解方法

## 2 提出的算法

### 2.1 理论基础

给定 1 个 XOR 表达式, 如果将其中偶数个变

量取反则表达式含义不变, 如果将其中奇数个变量取反则需增加 1, 此变换可以用  $x \oplus 1 = \bar{x}$  和  $x \oplus x = 0$  来解释.

以三输入变量为例, 有下面 2 种情况:

(1)  $f = x_0 \oplus x_1 \oplus x_2 = x_0 \oplus \bar{x}_1 \oplus \bar{x}_2$ ,

(2)  $f = x_0 \oplus x_1 \oplus x_2 = x_0 \oplus \bar{x}_1 \oplus x_2 \oplus 1$ ,

第 1 种情况的表达式  $x_1$  和  $x_2$  这 2 个变量同时取反, 而表达式的含义不变, 在这种情况下, 电路的面积保持不变. 第 2 种情况的表达式只有  $x_1$  取反, 与第 1 种情况相比, 要多 1 个输入端, 相应也就多 1 个 2 输入 XOR 门, 这样就会增加电路的面积, 显然, 第 2 种情况不可取.

给定 1 个  $n$  输入的 XOR 门, 设其输入信号为  $I = \{x_{n-1}, x_{n-2}, \dots, x_0\}$ , 按照其信号概率的大小排序为  $p_{r, x_0}, p_{r, x_1}, \dots, p_{r, x_n}$ , 存在以下 3 种情况:

情况 1:  $p_{r, x_{n-1}} = 0.5$ ;

情况 2:  $p_{r, x_0} < 0.5, p_{r, x_{n-1}} > 0.5$ ;

情况 3:  $p_{r, x_0} = 0.5$ .

根据输入信号概率所属的情况, 可以找出最佳的组合方式使得功耗降到最低. 对于情况 1, 可以直接用 Min-Huffman 算法对输入信号进行优化组合; 对于情况 2, 可以直接用 Zhou and Wong 的算法对输入信号进行优化组合; 而对于情况 3, 则需要一定的算法将其转化处理.

### 2.2 新的算法

对照以往的算法, 发现对信号概率大于 0.5 的变量的处理是关键所在, 设  $k$  为信号概率大于 0.5 的变量个数, 若  $k$  为偶数, 按照 Narayana and Liu 的算法即可得到最佳的 XOR 树; 若  $k$  为奇数, 将此  $k$  个信号分别取反, 然后将新组成的序列中信号概率最大的信号再取反, 这样就保证了用 Mini-Huffman 法构造 XOR 树时最小的信号概率两两结合, 而且表达式的含义不变, 从而按此变换可保证开关活动性最小.

根据以上讨论, 提出以下算法来搜索低功耗的 XOR 优化树:

(1) 若输入信号的概率属于情况 1,即所有输入信号的概率都不大于 0.5,则用 Min-Huffman 算法构造 XOR 分解树并计算其总的开关活动性;

(2) 若输入信号的情况属于情况 2 和 3,设  $k$  为信号概率大于 0.5 的变量个数,若  $k$  为偶数,将这  $k$  个信号分别取反;若  $k$  为奇数,将这  $k$  个信号分别取反,然后将概率最大的信号再取反.最后将得到的新信号概率用 Min-Huffman 法构造 XOR 树并计算其总的开关活动性;

(3) 输出树的结构以及总的开关活动性.

对于情况 1,与现有算法处理方法相同.对于输入概率分别为 0.2,0.3,0.9 的 3 输入 XOR 门,则属于情况 2,在动态逻辑下,Narayana and Liu 和 Zhou and Wong 的算法处理这种情况的方法一样,算出开关活动性为 1.952(若为静态逻辑电路,可计算出总的开关活动性为 0.9528),而用本文算法可以将其转化成 1 个输入概率为 0.1,0.2,0.7 的 3 输入 XOR 门,所算出的开关活动性为 1.328(若为静态逻辑电路,可计算出总的开关活动性为 0.8664).而对于输入概率为 0.6,0.7,0.8,0.8,0.9 的 5 输入 XOR 门,则属于情况 3,在动态逻辑下,用 Zhou and Wong 算法处理,算出的开关活动性为 3.352(若为静态逻辑电路,可计算出总开关活动性为 1.865),而用本文的算法所算出的开关活动性为 3.139(若为静态逻辑电路,可计算出总的开关活动性为 1.836).用 Narayana and Liu 的算法处理这种情况,只有 20%的概率找到最佳.

### 3 实验结果与分析

为将本文所提出的算法与 Narayana and Liu 和 Zhou and Wong 的算法进行比较,将上述 3 种算法都用 C 语言进行了实现,并用 VC++ 编译.并为验证所提出算法的有效性,进行相关实验.

实验是比较所提出的算法与 Narayana and Liu 算法和 Zhou and Wong 的算法的性能.对于 3 到 12

个输入变量,用随机产生的 100 组不同的信号概率对上述 3 种算法分别进行实现.设  $opt(x)$  为同组输入信号中开关活动性最小的 1 种算法.那么,算法  $s(x)$  的性能优劣可以用式(7)中的参数  $R$  来衡量:

$$R = (s(x) - opt(x)) / opt(x). \quad (7)$$

由式(7)可见,算法的性能随着  $R$  值得升高而变差.分别用  $NoOpt$ ,  $MR$  和  $AR$  表示 100 组不同实验中算法不是最佳的次数、 $R$  的最大百分率和平均百分率.在静态逻辑下,其比较结果如表 1 所示.

表 1 3 种算法在静态逻辑下性能比较的实验结果

In	N and L 算法			Z and W 算法			Proposed
	$NoOpt$	$MR/$ %	$AR/$ %	$NoOpt$	$MR/$ %	$AR/$ %	$NoOpt$
3	38	48.2	5.2	53	62.7	7.8	0
4	36	42.0	5.1	69	34.2	6.1	0
5	40	31.1	5.0	85	23.8	5.9	0
6	48	34.6	5.2	88	23.2	5.0	0
7	46	19.9	3.4	90	19.2	3.8	0
8	43	23.6	3.4	93	16.0	3.2	0
9	46	19.1	3.0	97	12.2	2.9	0
10	55	15.8	3.5	98	6.7	2.6	0
11	55	13.3	3.1	98	5.3	2.3	0
12	53	11.7	2.9	99	4.6	2.2	0

表 1 为在静态逻辑下以本算法作为比较基准的实验结果.从表 1 可看到,随着输入变量数目的增加,Zhou and Wong 的算法找到最佳的概率是越来越小,而 Narayana and Liu 的算法不是最佳的概率始终在 50%左右,这是因为当  $k$  为偶数时,使用该算法可以达到最佳.

在动态逻辑下,其比较结果如表 2 所示.表 2 给出了在动态逻辑下以本算法作为比较基准的实验结果.与表 1 结果类似,Narayana and Liu 的算法不是最佳的概率始终在 50%左右,而随着输入变量的增加,Zhou and Wong 的算法的性能有所改善.

从以上的比较可以看到本文所提出的算法在这 2 种逻辑下都能找到最佳,因此本文中的算法要比 Narayana and Liu 和 Zhou and Wong 的算法有所

改善.

表 2 3 种算法在动态逻辑下性能比较的实验结果

In	N and L 算法			Z and W 算法			Proposed
	NoOpt	MR/ %	AR/ %	NoOpt	MR/ %	AR/ %	NoOpt
3	41	46.0	7.8	49	58.0	5.8	0
4	34	40.1	3.1	68	40.6	8.9	0
5	44	35.8	8.6	88	20.7	4.3	0
6	45	34.7	2.6	91	28.2	7.4	0
7	51	25.0	6.8	90	15.9	2.6	0
8	42	21.0	1.4	96	20.0	4.9	0
9	48	22.5	5.9	96	10.6	1.9	0
10	53	14.1	1.3	98	7.3	4.0	0
11	54	12.0	1.1	99	5.3	3.6	0
12	54	13.4	5.9	99	4.8	1.3	0

## 4 结论

本文仅针对多输入 XOR 门的分解,对已发表的 Narayana and Liu 的算法加以改进,提出了一种新的算法.并通过实验结果表明,与 Narayana and Liu 算法和 Zhou and Wong 算法相比,本文算法可以更有效地降低功耗.

参考文献:

- [1] Najm F. Transition density: a new measure of activity in digital circuits[J]. IEEE Transactions on CAD, 1993, 12(2):310-323.
- [2] Panda R, Najm F. Technology decomposition for low-power synthesis[C]//IEEE Custom Integrated Circuits Conference, 1997:650-655.
- [3] Tsui C, Sandowska M M. Technology mapping for low power[C]//Proceedings of the 33rd IEEE/ACM International Conference DAC, USA: Las Vegas, 1996:68-73.
- [4] Narayanan U, Liu C L. Low power logic synthesis for XOR based circuits[C]//ICCAD, 1997:570-574.
- [5] Zhou Hai, Wong D F. Optimal low power XOR gate decomposition[C]//Design Automation Conference, 2000: 104-107.
- [6] Zhou Hai, Wong D F. An exact gate decomposition algorithm for low-power technology mapping[C]//Proceedings of IEEE/ACM, International Conference Computer-Aided Design(ICCAD97). CA: San Jose, 1997:575-580.
- [7] Xia Yinshui, Almaini A E A. Best polarity for low power XOR gate decomposition[J]. Proceedings of the Euromicro Symposium on Digital System Design, 2002(9):53-59.
- [8] 干雪, 叶锡恩, 夏银水. 基于 XOR 门的静态逻辑电路功耗优化技术[J]. 宁波大学学报: 理工版, 2006, 19(2): 155-160.

## Fast Decomposition Techniques for Low Power XOR Gate

XU Liang, XIA Yin-shui

( Faculty of Information Science and Technology, Ningbo University, Ningbo 315211, China )

**Abstract:** In order to fast minimize the power dissipation, we propose a novel method to optimize the low power decomposition for XOR-based circuits. The XOR gate decomposition algorithm is implemented in C programs. The experimental results indicate that the proposed method further reduces the power dissipation by additional up to 8.9% compared to the reduction that the current methods can achieve. The proposed algorithm is found to be more efficient than those of the same type in terms of power dissipation.

**Key words:** low power; reed-muller logic; XOR gates

**CLC number:** TN47

**Document code:** A

( 责任编辑 章践立 )