

考虑时间相关性的功耗估计熵算法

吴一品¹, 周 锋¹, 陈春鸿², 严 伟¹, 李 舜¹, 彭云峰¹

(1. 复旦大学专用集成电路与系统国家重点实验室, 上海 201203;

2. Department of Electrical and Computer Engineering, University of Windsor, Ontario, Canada N9B 3P4)

摘 要: 熵估计是一种在高层次估计功耗的方法, 但已经提出的熵算法无法考虑输入信号在时间上的相关性. 本文提出了改进熵的概念, 在传统熵中加入条件翻转因子, 使改进后的熵能够有效估计时间上有相互关联性的信号的翻转率. 理论证明和大量 BENCHMARK 实验结果都表明我们提出的改进熵算法具有合理性和可靠性.

关键词: 功耗估计; 熵; 时间相关性

中图分类号: TP331; TN431 **文献标识码:** A **文章编号:** 0372-2112 (2007) 05 0933-04

An Improved Entropy to Estimate Power for Considering Temporal Correlation at High Level

WU Yir pin¹, ZHOU Feng¹, CHEN Chun hong², YAN Wei¹, LI Shun¹, PENG Yur feng¹

(1. State Key Lab of ASIC & System, Fudan University, Shanghai 201203, China;

2. Department of Electrical and Computer Engineering, University of Windsor, Ontario, Canada N9B 3P4)

Abstract: The traditional entropy is an efficient method for high level power estimation, but it doesn't work when the input signals are temporal correlated, as is always the case for video and audio streams. This paper aims at this problem. We put forward a new definition of entropy. With the help of the conditional transition probabilities, the proposed algorithm can bring us the estimations with adequate accuracy for temporal correlated inputs. The theoretical proofs and the BENCHMARK experimental results verify the efficiency of our algorithm.

Key words: temporal correlation; power estimation; entropy

1 引言

随着集成电路工艺的发展, 电路的集成度和复杂度不断提高, 使得功耗成为 VLSI 电路与系统设计首要考虑的因数之一. 为了避免重复设计, 高层次的功耗估计也越来越迫切. 门级翻转的功耗估计方法可以分为两大类: 统计方法(动态方法)和概率方法(静态方法)^[1]. 在超大规模电路的高层次上, 门级翻转的估计方法将因其时间和空间上的复杂度而无法使用, 同时在电路综合之前作的系统级功耗估计对精确度的要求相对宽松, 所以我们需要研究新的算法来提高翻转率的估计速度. 用熵建立的模型来估计功耗就是一种在高层次估计功耗的方法.

在 Shannon 信息论中用熵的概念来衡量数字信号的复杂度, 数字电路实现了将输入信号转换为输出信号的过程. M Nemni 和 F N Najm 提出了基于熵的布尔逻辑电路高层次功耗分析估计方法^[2], 仅利用一个模块的输入和输出节点的翻转信息, 就能估计整个模块的功耗. 这种方法计算速度快, 精确度较高, 适合在高层次系统级作功耗估计. 但传统熵算法有一个限制, 要求每个输入信号在时间上没有相关性. 这样如果一个模块的输入

信号具有很强的时间相关性, 如声音和图像信号等, 传统熵算法就不能应用. 本文对传统熵模型进行了改进, 通过引入条件翻转因子来解决这个问题. 实验结果表明, 我们提出的改进熵算法能够以不亚于传统熵算法的精确度估计具有时间相关性信号的模块的功耗.

2 动态功耗的熵估计方法

在数字电路中功耗的主要来源是节点对电容的充放电, 从文[2]可以得到式(1),

$$P_{avg} \propto D_{avg} \quad (1)$$

其中
$$D_{avg} = \frac{1}{N} \sum_{i=1}^N D(x_i) \quad (2)$$

N 表示电路节点总数(包括电路内部全部节点), $D(x_i)$ 表示第 i 个节点的翻转率, D_{avg} 表示电路的平均翻转率.

传统熵算法将电路中的某一节点 x_i 的熵定义为^[2]:

$$H(x_i) = -p_i \log p_i - (1-p_i) \log(1-p_i) \quad (3)$$

其中 p_i 表示节点 x_i 上的信号为 1 的概率. 文献[2]表明 $H(x_i)$ 非常接近曲线 $4p_i(1-p_i)$, 所以也可以把熵近似为

$$H(x_i) = 4p_i(1-p_i) \quad (4)$$

假设信号在时间上没有相关性, 则

$$D(x_i) = 2p_i(1 - p_i) \quad (5)$$

由式(4), (5) 我们得到熵和翻转率的关系

$$H(x_i) \approx 2D(x_i) \quad (6)$$

由式(2), (6) 有熵和功耗的关系

$$P_{avg} \propto H_{avg} \quad (7)$$

其中

$$H_{avg} = \frac{1}{N} \sum_{i=1}^N H(x_i) \quad (8)$$

P_{avg} 表示每个节点的平均功耗, H_{avg} 表示电路中所有节点平均熵, N 表示电路节点总数, 第 i 个节点和第 i 个 $H(x_i)$ 熵相对应. 由文[2], 式(8)可以近似为

$$H_{avg} = \frac{2}{3(n+m)}(H_i + 2H_o) \quad (9)$$

其中 m 和 n 分别表示电路输出和输入的端口数, H_i 和 H_o 分别表示输入熵的总和输出熵的总和, 记为

$$H_i = \sum_{i=1}^n H(x_i), \quad H_o = \sum_{j=1}^m H(x_j)$$

可以看出, 传统的熵只需计算输入和输出节点的熵, 利用式(9)就可求出包括内部节点在内的电路全部节点平均熵(由此可推出电路全部节点的总翻转率). 和经典概率方法相比, 极大地降低了算法的时间和空间复杂度, 适合在高层次估计功耗. 但这样定义的熵只有信号在时间上没有相关性时才和翻转率有线性关系, 因此有很大的局限性.

3 改进熵估计翻转率方法

在本节中, 首先推导由条件翻转因子表达的时间相关信号的翻转率 D 以及由输入信号的条件翻转因子求输出信号条件翻转因子的方法, 然后给出改进熵的定义, 最后证明改进熵和翻转率 D 之间的关系.

3.1 条件翻转因子和翻转率

对某一信号 x_i , 已知 p_{01} 和 p_{10} 分别表示在已知上一个时钟周期为 0 时, 这个时钟周期为 1 的概率和已知上一个时钟周期为 1 时, 这个时钟周期为 0 的概率, 用 t 表示这个时钟周期, $t-T$ 表示上一个时钟周期, p_{01} 和 p_{10} 定义成 $P(x_i(t) = 1 | x_i(t-T) = 0)$ 和 $P(x_i(t) = 0 | x_i(t-T) = 1)$, 简单记为 $P(x_i(t) | \bar{x}_i(t-T))$ 和 $P(\bar{x}_i(t) | x_i(t-T))$.

假设 x_i 是 Markov 链平稳分布^[3,4], 其状态转移图如图 1 所示.

从文[4]得:

$$\begin{bmatrix} p(x_i = 0) \\ p(x_i = 1) \end{bmatrix} = \begin{bmatrix} p_{00} & p_{10} \\ p_{01} & p_{11} \end{bmatrix} \begin{bmatrix} p(x_i = 0) \\ p(x_i = 1) \end{bmatrix} \quad (10)$$

由概率的定义:

$$p(x_i = 1) + p(x_i = 0) = 1 \quad (11)$$

$$p_{00} = 1 - p_{01} \quad (12)$$

$$p_{11} = 1 - p_{10} \quad (13)$$

从式(10), (11), (12), (13) 得到 p_i 的表达式

$$p_i = P(x_i = 1) = P(x_i(t-T)) = P(x_i(t)) = \frac{p_i 01}{p_i 01 + p_i 10} \quad (14)$$

由条件概率公式得到式(15), (16):

$$P(\bar{x}_i(t-T) | x_i(t)) = p_{01} p(x_i = 0) \quad (15)$$

$$P(x_i(t-T) | \bar{x}_i(t)) = p_{10} p(x_i = 1) \quad (16)$$

所以得到 D 的表示式如

$$\begin{aligned} D_i &= P(\bar{x}_i(t-T) | x_i(t)) + P(x_i(t-T) | \bar{x}_i(t)) \\ &= \frac{2p_i 01 p_i 10}{p_i 01 + p_i 10} \end{aligned} \quad (17)$$

对于一个 n 输入的布尔逻辑 $y = f(x_1, x_2, \dots, x_n)$, 如果知道每个输入的 p_{10} 和 p_{01} , 就可以根据式(15), (16) 求出每个输入的 P 和 D . 然后根据文[5]中的 BDDs 方法重复使用式(18) n 次即可求出输出的 P .

$$P(y = 1) = P(x_i = 1) * P(f_{x_i=1}) + P(x_i = 0) * P(f_{x_i=0}) \quad (18)$$

根据文[6]中方法用式(19) 就可以求出输出的 D .

$$D(y) = 2(P(y) - P(y(t) | y(t-T))) \quad (19)$$

最后再根据式(14), (17) 求出的输出的 p_{10} 和 p_{01} :

$$p_{y10} = \frac{D(y)}{2P(y=1)} \quad (20)$$

$$p_{y01} = \frac{D(y)}{2(1 - P(y=1))} \quad (21)$$

这样就能够从已知的每个输入的 p_{10} 和 p_{01} 求出每个输出的 p_{10} 和 p_{01} .

3.2 改进熵的定义

我们定义改进熵:

$$H_c = (-p \log p - (1-p) \log(1-p)) * (p_{01} + p_{10}) \quad (22)$$

其中 p 是信号等于 1 的概率, 根据式(14)可知,

$$p = \frac{p_{01}}{p_{01} + p_{10}} \quad (23)$$

3.3 改进熵的证明

从定义式(22) 可以看出改进熵和传统熵的关系为

$$H_c = H * (p_{01} + p_{10}) \quad (24)$$

由式(4), (15) 得:

$$H \approx 4p(1-p) = \frac{4p_{01}p_{10}}{(p_{01} + p_{10})^2} \quad (25)$$

所以改进熵可以由式(24), (25) 用 p_{01} 和 p_{10} 表示:

$$H_c = \frac{4p_{01}p_{10}}{p_{01} + p_{10}} \quad (26)$$

比较式(17), (26) 最终可以得到改进熵和考虑时间相关

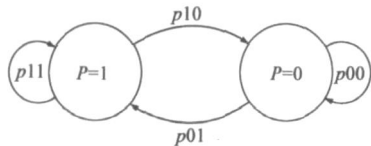


图 1 状态转移图

性的翻转率也呈线性关系:

$$H_c = 2D \quad (27)$$

当信号在时间上相互不关联时, 即当

$$p_{01} = p(x_i = 1 | x_i = 0) = p(x_i = 1) \quad (28)$$

$$p_{10} = p(x_i = 0 | x_i = 1) = p(x_i = 0) \quad (29)$$

时, 可以得到:

$$p_{01} + p_{10} = p(x_i = 1) + p(x_i = 0) = 1 \quad (30)$$

$$D = \frac{2p_{01}p_{10}}{p_{01} + p_{10}} = 2p(1-p) \quad (31)$$

$$H = H_c \quad (32)$$

可以看到 D 蜕化成式(5), 改进熵也蜕化成为传统熵. 由此可见, 传统熵只是改进熵的一种特例而已, 而改进熵能够估计信号在时间上相互关联时的翻转率, 有比传统熵更广泛的应用性.

4 时间复杂度分析

经典的求翻转率的方法需要对电路中每个节点求翻转率 D . 假设电路中某个节点由 r 个输入节点决定逻辑值 $y = f(x_1, x_2, \dots, x_r)$, 那么求 $P(y)$ 的复杂度是 $r^{[7]}$, 而求 $P(y(t) | y(t-T))$ 复杂度是 $L^2 r^{[7]}$, 其中 L 表示乘积项的个数. 由式(19)可以知道, 求这个节点的 D 的复杂度是 $(L^2 + 1)r$. r 的最坏情况就是电路的输入节点数目 n .

现在我们来估算电路中的节点数目. 首先对电路采用分级的技术^[7], 大部分逻辑电路也是可以看成这样的, 输入的可以看成第 0 级, 输出看成第 K 级. 则电路第 i 级的节点数目 $W(i)$ 可用(33)式表示^[2],

$$W(i) = m + (n - m) \left(1 - \frac{i}{K} \right) \quad (33)$$

其中 n 是输入节点的数目, m 是输出节点的数目. 电路节点的总数 N 可以表示为式(34),

$$N = \sum_{i=0}^K W(i) = \sum_{i=0}^K \left[m + (n - m) \left(1 - \frac{i}{K} \right) \right] = \frac{(n + m)(K + 1)}{2} \quad (34)$$

假设电路的结点数随着级数的变化而线性变化^[2], K 可以近似为:

$$K \approx |n - m| \quad (35)$$

N 的表达式(34)可以表示成

$$N = \frac{(n + m)(|n - m| + 1)}{2} \quad (36)$$

所以经典的求翻转率的方法的复杂度是:

$$\begin{aligned} & \frac{(n + m)(|n - m| + 1)}{2} (L^2 + 1)n \\ & \approx \frac{(n + m)(|n - m| + 1)}{2} L^2 n \end{aligned} \quad (37)$$

下面分析传统熵的算法复杂度. 从式(9)可以看

出, 传统熵的计算量与电路的输入、输出节点总数成正比, 而求每一个输出的 p 的复杂度是 n (简单假设每一个输出都和每一个输入有关, 这是最差情况), 所以, 传统熵的复杂度为

$$mn \quad (38)$$

复杂度很低.

改进熵在传统熵的基础上增加了每个输出节点的 p_{01} 和 p_{10} 的计算. 由文献[7]可知, 增加的部分计算量与 $L^2 n$ 成正比 (也简单假设每一个输出都和每一个输入有关), 其中 L 表示乘积项的个数. 综之, 改进熵复杂度是:

$$L^2 mn \quad (39)$$

由此可见, 改进熵的时间复杂度比传统熵高, 但仍然比经典方法优越. 改进熵以较小的时间复杂度的代价, 扩大了熵算法的应用范围, 使其能作时间相关的输入信号的高层次功耗估计.

5 实验结果

以 ISCAS85 Benchmark 中 10 个电路为实验样本电路. 从式(20)和式(21)可以看出, 对于任意一组输入概率 P_{in1}, P_{in2}, \dots 和输入翻转率 D_{in1}, D_{in2}, \dots 有唯一的一组 $(p_{01}, p_{10})_1, (p_{01}, p_{10})_2, \dots$ 和其相对应. 图 2 显示的是当 $P_{in} (= P_{in1} = P_{in2} = \dots) = 0.5$ 时, 各个电路的改进熵随着输入翻转率 $D_{in} (= D_{in1} = D_{in2} = \dots)$ 变化的曲线图. 图 2 说明, 即便 P_{in} 固定不便, 改进熵仍然会变化. 图 3 显示的是当 $D_{in} = 0.5$ 时, 各个电路的改进熵随着 P_{in} 变化的曲线图 (其中 P_{in} 值只能取 0.25 到 0.75 之间, 因为受到限制^[8]). 图 3 更清楚地表明当输入翻转率 D_{in} 固定时, 改进熵随输入概率 P_{in} 的变化很小. 图 2、图 3 说明, 改进熵主要由输入翻转率 D_{in} 决定, 不再和传统熵一样由输入概率 P_{in} 决定.

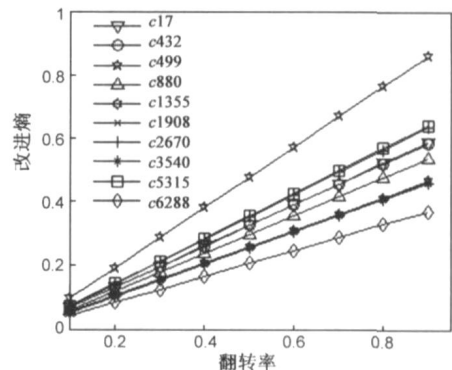


图 2 当 $p_{in} = 0.5$ 时输入的翻转率和改进熵的关系

为验证改进熵求出的平均翻转率和实际节点平均翻转率有一致关系, 我们为 benchmark 中 10 个电路各用 50 组不同的 P_{in} 和 D_{in} 做了测试, 结果如图 4 所示. 横轴表示通过改进熵求得的平均翻转率, 竖轴表示用经典

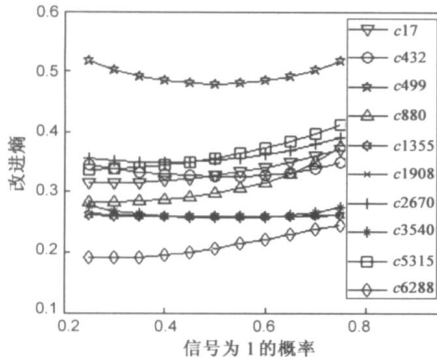


图 3 当 $Din = 0.5$ 时, 输入信号为 1 概率和改进熵的关系

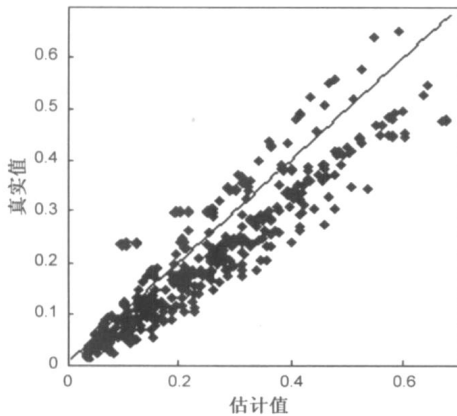


图 4 改进熵和实际的翻转率的关系

方法求出的平均翻转率。从图 4 中可以看出它们之间具有很好的一致性, 说明改进熵能有效的估计整个电路的翻转率。图 4 也显示通过改进熵得到的估计值总体而言要比电路的实际翻转率要大一些, 这是由熵估计本身的误差引起的, 因为每个节点的传统熵 H 比节点翻转率的两倍 $4p(1-p)$ 要略大一些, 只有在 $p_i = 0.5$ 时两者才相等^[4], 如文[2]所示。

最后分析图 4 中数据的绝对误差, 分别求出 benchmark 中 10 个电路 50 组数据由改进熵得到的翻转率和电路的实际翻转率的绝对误差, 进而得到 10 个电路 50 个绝对误差的平均值, 得到图 5。可以看到它们的绝对误差都在 0.1 以下, 这对高层次估计来说是完全可以接受的。

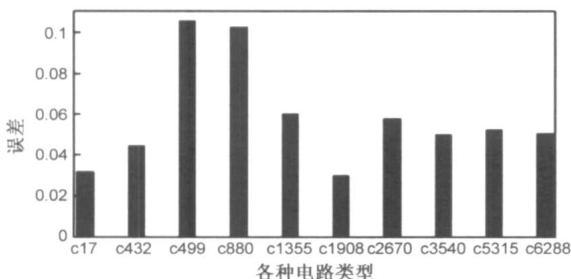


图 5 各个电路的绝对误差值

6 结论

熵的方法可以以极低的时间度和极高的准确度估算电路节点的总翻转率, 因而特别适用于大规模电路的高层次功耗估计。但传统熵不能处理输入信号具有时间相关性的情况。本文针对这个问题, 引入条件翻转因子, 提出了改进熵。理论和实验结果证明了我们提出的改进熵算法的有效性, 它比传统熵具有更广泛的适用范围。

参考文献:

- [1] F N Najm. A survey of power estimation techniques in VLSI circuits[J]. IEEE Trans. on VLSI Systems, 1994, 2(4): 446-455.
- [2] M Nemani, F N Najm. Toward a high level power estimation capability [J]. IEEE Trans. Computer Aided Design Integr Syst, 1996, 15(6): 588-598.
- [3] T M Cover, J A Thomas. Elements of Information Theory[M]. New York: Wiley, 1991.
- [4] P H Schneider, U Schlichtmann, B Wurth. Fast power estimation of large circuits[J]. Design & Test of Computers IEEE, 1996, 13(1): 70-78.
- [5] S Minato, N Ishiura, S yajima. Shared binary decision diagram with attributed edges for efficient Boolean function manipulation[A]. 27th Design Automation Conference [C]. Orlando, Florida: June, 1990. 52-57.
- [6] T L Chou, K Roy, S Prasad. Estimation of circuit considering signal correlations and simultaneous switching[A]. ACM/ IEEE International Conference on Computer Aided Design [C]. San Jose: November, 1994. 300-303.
- [7] K Roy, S Prasad. Low Power CMOS VLSI Circuit Design [M]. Wiley, 2000.
- [8] S Gupta, F N Najm. Power modeling for high level power estimation[J]. IEEE Trans VLSI Syst, 2000, 8(1): 19-28.

作者简介:



吴一品 男, 1982 年 6 月出生在浙江省浦江, 现为复旦大学微电子学与固体电子学专业研究生。目前主要从事电路的低功耗设计和研究。E-mail: ypwu@fudan.edu.cn

周锋 男, 1966 年出生在浙江杭州, 1999 年在复旦大学固体电子学与微电子学专业取得理学博士学位, 此后在复旦大学微电子系从事科研、教学工作至今, 其中 2001 年底至 2002 年底在美国加州大学圣地亚哥分校做访问学者。主要研究方向是模拟集成电路设计、数模混合集成电路设计和低功耗设计方法学。

E-mail: fengzhou@fudan.edu.cn