

CMOS VLSI 电路最大功耗估计

卢君明, 林争辉

(上海交通大学大规模集成电路研究所, 上海 200030)

摘 要: 最大功耗分析对于设计高可靠性的 VLSI 芯片是非常重要的. 由于电路功耗强依赖于其输入模式, 对有大量管脚的 CMOS 组合或时序电路, 不能采用穷举搜索. 本文用遗传算法来选择具有高功耗的输入及内部状态模型, 在逻辑仿真基础上实现 CMOS 电路的最大功耗估算. 同时用逻辑仿真的统计方法来衡量获得最大功耗的质量. 基于 ISCAS85 和 ISCAS89 基准电路的仿真表明, 新方法在大规模门数时具有明显的优势, 估算精度较高. 而且新方法的计算时间基本上是电路逻辑门的线性关系.

关键词: CMOS 时序电路; 最大功耗估计; 遗传算法; 逻辑模拟

中图分类号: TN47 **文献标识码:** A **文章编号:** 0372-2112 (2001) 05-0630-04

Estimation of Maximum Power Dissipation of CMOS VLSI

LU Jun-ming, LIN Zheng-hui

(LSI Research Institute, Shanghai Jiaotong University, Shanghai 200030, China)

Abstract: Estimation of maximum power dissipation is important in designing highly reliable VLSI systems. However, maximum power estimation for CMOS circuits is essentially a combination optimization problem, which has exponential complexity in the worst case. In this paper, we propose a novel approach to obtain a lower bound of the maximum power dissipation using Genetic Algorithm (GA). Experiments with ISCAS-85 and ISCAS-89 benchmark circuits show that our approach generates the lower bound with the quality which cannot be achieved using simulation-based techniques. In addition, a statistics based technique is realized to serve as a comparison version for our GA approach and to generate a metric to measure the equality of a lower bound from a statistical point of view.

Key words: CMOS sequential circuits; dissipation estimation; genetic algorithm; logic simulation

1 引言

由于 VLSI 电路过高的瞬间功耗会降低电路的可靠性和性能, 随着当今 VLSI 设计的高可靠性要求, 有必要在 VLSI 电路的综合过程中进行最大功耗估计. 对 CMOS 电路来说, 功耗主要是开关功耗, 既由于开关翻转时负载电容充放电引起的功率损耗. 因此要精确估计 CMOS 组合电路的最大功耗, 就要穷举搜索所有可能的连续两个输入位矢量, 以获得电路内部尽可能多的开关翻转. 这一问题是一个完全 NP 问题, 搜索的时间复杂度是 $O(2^{2n})$, 其中 n 是电路原始输入管脚数. 对时序电路, 因为既要考虑电路的原始输入 (PI's, primary inputs), 又要考虑到电路内部锁存器状态, 搜索具有最大开关翻转数的原始输入和状态的时间复杂度为 $O(2^{2n+m})$ (m 为内部锁存器数目). 因此对 CMOS VLSI 的输入模式进行直接穷举搜索, 获取最大功耗估计是不现实的.

Wang^[1]等人首先提出了采用基于 ATPG 的方法估算组合电路最大功耗的下限, 然后又将该方法推广到时序电路中^[2]. 这类算法通过从内部节点的信号翻转分配, 从而得到一组输入测试向量, 使得电路在其驱动下产生的开关功耗最大.

基于 ATPG 生成测试向量来获得最大功耗估算的思想, 本文采用改进的遗传算法, 在电路原始输入模式中进行优化搜索, 求得一组原始输入模式, 在该输入模式下, 使电路获得最大功耗来实现对 CMOS VLSI 的最大功耗估计.

2 算法准备

2.1 CMOS 电路功耗因子

在 CMOS 电路中, 功耗主要是开关功耗^[3], 电路的瞬间功耗由下式决定:

$$P = 0.5 \times (V_{dd}^2 / T_{cyc}) \times \prod_{\text{for all gates}} T(g) \times C(g) \quad (1)$$

其中 $C(g)$ 是逻辑门 g 的负载电容, $T(g)$ 是个布尔函数, 代表两个连续时钟内逻辑门 g 是否实现信号翻转. V_{dd} 是电源电压, T_{cyc} 是全局时钟周期. $T(g)$ 取决于电路的逻辑功能和输入的测试向量. 因为门的负载电容与其扇出数是成正比的, 显然我们可以定义一个与电路瞬间功耗成正比的功耗因子 PF , 作为衡量电路开关功耗估计算法结果的参量^[4]:

$$PF = \prod_{\text{for all gates}} [G(V_1) \oplus G(V_2)] \times F_{out}(g) \quad (2)$$

V_1 和 V_2 是电路的两个连续位输入向量, $G(*)$ 是门 g 关于电路原始输入的布尔函数. 这是组合电路最大功耗估计中要最大化的直接目标函数.

2.2 时序电路时间相关性

在时序电路中, 采用米勒模型 (Mealy Model), 将内部锁存器的输出定义为伪原始输入, 锁存器的输入定义为伪原始输出. 对给定的时序电路将原始输入和伪原始输入组合成一个组合输入向量, 则最大功耗的估计就变成了这样一个过程: 寻求两个连续的组合输入向量, 获得最大电路开关翻转. 与式 (2) 相比, 时序电路和组合电路的唯一不同就是组合输入向量在时序电路中是相关的. 下面分析这种相关性.

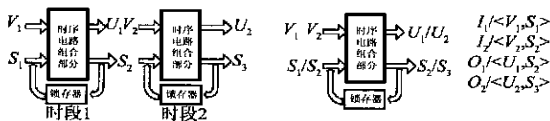


图 1 时序电路分解图 图 2 重合两个连续时段

将时序电路分解成两个连续的时间段, 如图 1 所示, 在时段 1 中, 原始输入向量 V_1 使电路状态从 S_1 转换到 S_2 . 而在时段 2 中 V_2 使状态 S_2 变到 S_3 . U_1 和 U_2 分别表示电路在两个时段的原始输出. 图 2 中, 同时考虑两个时段, 将两个时段的逻辑组成复合逻辑, 假设某个逻辑门在时段 1 时为逻辑 1 而在时段 2 时为逻辑 0, 则其复合逻辑为 1/0. 从图 2 中可以看出, 由于 V_1 和 S_1 在同一个时段, 可以包装成单个组合输入向量 $I_1 = \langle V_1, S_1 \rangle$. 相似的可以得到其它组合向量对: $I_2 = \langle V_2, S_2 \rangle$, $O_1 = \langle U_1, S_2 \rangle$ 和 $O_2 = \langle U_2, S_3 \rangle$. 如果时序电路有 n 个原始输入和 m 个状态位, 对组合向量 I_2 和 O_1 , 其第 $n+1$ 个到 $n+m$ 个元素必须相等.

由等式 (2) 和以上分析, 时序电路的最大功耗估计问题可描述如下:

$$\text{Max} (PF = \prod_{\text{for all gates}} [G(I_1) \odot G(I_2)] \times F_{\text{out}}(g)) \quad (3)$$

$$\text{约束条件: } I_j = O_j; j = n+1, n+2, \dots, n+m$$

此处 I_j 和 O_j 分别表示组合向量 I_2 和 O_1 的第 j 个元素. 由于组合输出 O_1 是相应组合输入向量 I_1 的输出, 因此 描述了连续输入组合向量 I_1 和 I_2 的时间相关性.

3 基于遗传算法的 COMS 电路最大功耗估计

3.1 遗传算法介绍

遗传算法^[5]结构简单, 并且对搜索空间 (目标函数的性质) 不需任何限制. 比起其他搜索方法, 如随机查找、梯度下降、模拟退火等, 主要特点是简单、鲁棒性强. 遗传算法实现全局并行搜索, 搜索空间大, 并且在搜索过程中不断地向可能包含最优解的方向调整搜索空间, 以便寻找到最优解或准最优解. 它将实际问题模拟成个体的生存环境, 将目标函数模拟成个体的生存能力, 将可能解模拟为个体. 这样, 从任一个初始解群出发, 经过选择 (selection)、杂交 (crossover)、变异 (mutation) 三种运算, 产生新一代解群, 从而经过多次迭代后, 使其收敛于全局最优解或全局次最优解. 由于组合电路是时序电路的一个特例, 即当电路中不存在锁存器时, 就成了组合电

路, 因此下面主要介绍时序电路最大功耗估计算法.

3.2 时序电路最大功耗估计

对时序电路最大功耗估计, 目标函数为式 (3), 目的是在电路的组合输入模式中搜索两个连续的组合输入向量 I_1 和 I_2 , 使 PF 最大化, 找到了 I_1 和 I_2 , 也就估计出了最大功耗. 基于以上目的, 采用了遗传算法进行搜索, 下面叙述时序电路最大功耗估计中的遗传算法.

3.2.1 编码

遗传算法进行优化操作的对象是所有需要优化变量的编码. 在本文的应用中, 采用了电路的组合输入向量 (由原始输入和状态位组合而成) 作为算法的编码, 连续的两个组合输入向量 (即一个复合逻辑向量) 编码为一个个体.

3.2.2 父本选择

本文遗传算法解群的群体规模为 $n = 100$, 即有 100 个个体, 匹配集中个体数为 $m = 10$, 即每次在解群中选择 10 个个体作为下一代迭代的父本. 本文选择的选择机制是这样的, 对每个个体用式 (3) 计算 PF , 然后对群体中的所有个体, 依照 PF 值的大小从大到小排列, 选择排在前十个个体作为下一代的父本. 给选中的个体依 PF 值相应的赋一个从 10 到 1 的整数值, 再采用线性归一化技术^[6], 计算每个个体的适合度值. 不采用实际功耗因子值作为适合度的原因是这样的, 通常选择技术采用轮盘机技术, 即个体按正比于适合度值的概率从匹配集被选择来杂交再生. 在这种方法下, 个体适合度越高产生下一代个体的数目越多. 然而当个体的适合度很接近时, 情况就不一样了. 考虑到某一代中, 最好的和最差的功耗分别为 617 和 619, 群体的平均功耗为 618. 则最好和最差的个体产生的下一代数目基本差不多, 这时就没有自然选择的效应了. 而采用上述的线性归一化技术, 就能始终使适应度最好的个体产生较多的下一代.

同时采用优秀个体保护策略^[7], 保护以前的最优解. 该策略是这样实现的, 保存上一代的最优个体, 与当前选出的父本个体进行比较, 如果它比最差的当前父本个体优秀, 则用上代的最优个体替换掉当前选出的最差父本. 如果它比最优的当前父本个体差, 则保存该最优当前父本, 以便下一代迭代. 该方法使父代中的最优个体总是能生存到下一代, 而且保持了群体规模的稳定性.

3.2.3 杂交再生

在匹配集中随机选取两个个体, 杂交概率选择为 80%, 杂交方法采用单点杂交, 但杂交的位置随机确定, 杂交的长度为电路组合输入数 (原始输入管脚加上状态位数) 的 10%. 并将由杂交产生的两个子代个体放到新的群体中, 重复该过程, 直到新的群体仍包括了 100 个个体. 以便以下变异处理.

3.2.4 变异

变异的作用是增强个体的多样性, 增加收敛到最优解的机会. 对杂交再生产生的新群体中的每个个体, 变异概率选择为 10%. 变异的位置也是随机确定. 本文中变异是这样操作的, 众所周知, 电路输入的高翻转率一般会导致电路内部节点和输出节点的高翻转率, 因此随机确定变异位置后, 分两种情况处理: 对在原始输入区, 比较该位置两个组合输入向量元素

是否相同,如相同,则第二个向量的元素取反,即该原始输入位置实现翻转,如不同,则互换该位置的元素值;对在状态输入区,由于状态的时间相关性,直接对第一个向量进行翻转,而第二个向量对应的元素由电路输出决定。

算法的流程采用经典遗传算法流程,其 C 语言的伪代码如下:

```

Max Power() {
    当前代数 = 1; 最大代数 = 100;
    群体规模 = 100;
    个体编码长度 = 原始输入数 + 内部状态数;
    初始化个体群;
Do {
    计算群体中个体的功耗因子;
    按功耗因子从大到小对个体排序;
    选择最好的 10 个个体作为父本;
    优秀个体保护;
    采用线性归一化技术计算适应度;
    用轮盘机技术选择父本进行杂交;
    重复杂交过程产生 100 个新个体;
    对新个体进行变异操作;
} While ( + 当前代数 < 最大代数)
}

```

4 随机搜索方法

在估计最大功耗时,很自然的方法就是对电路进行逻辑仿真,并监视仿真时产生的瞬间功耗的最大值,这个值我们定义为仿真最大功耗。为判断仿真最大功耗 X 的有效性,我们给每个 X 赋一个估计质量,该估计质量是瞬间功耗的均值和方差的函数。因此需要在电路仿真时采用统计技术估计瞬间功耗的均值和方差。在正态分布的假设下,均值和方差可以方便地求得。因此仿真最大功耗估计的质量可在仿真结束后求得。这个方法可应用于门级仿真器中,在电路原始输入中加高翻转率的信号来逻辑仿真,再进行估计均值和方差(通常认为,高翻转率的原始输入会使电路内部节点产生高翻转率)。

为了方便地估计瞬间功耗因子 P_c 的均值和方差,假设瞬间功耗因子是正态分布。因为根据式(1), P_c 是许多随机位变量的和,这些位变量表明该门在两个连续组合输入向量下的翻转情况。如果所有的翻转是独立且具有相同的分布特性,根据中心极限定理(Central Limit Theorem),当电路的门数足够大,则 P_c 的分布是接近正态分布的。

在 P_c 是正态分布的假设下, P_c 的分布函数,用 $DP_c(x)$ 表示,可以用 P_c 已知的均值和方差来计算。对一个具有较大分布值的仿真最大功耗 X ,继续仿真获得更大仿真最大功耗的概率就比较

低。因此我们可以用概率值 $DP_c(x)$ 作为仿真最大功耗 X 的估计质量。

如图 3 示, P_c 是瞬间功耗,是一个随机变量。 $DP_c(x)$ 是 X 的估计质量,此处 X 是仿真最大功耗。 $DP_c(x) = \Pr\{P_c < x\} = \text{阴影面积}$ $f(P_c)$: P_c 的概率密度函数。同理,用统计方法获得 P_c 的均值和方差后,也可以计算由遗传方法获得最大功耗的估计质量。

5 仿真和实验结果

本文提出的基于遗传算法的 CMOS 电路最大功耗估计方法(GaMax)和逻辑仿真方法(SimMax)均已在 SUN ULTRA 10 工作站上用 C 语言实现。时序电路以 ISCAS89 的基准电路为例,组合电路以 ISCAS85 为例,运行该两种最大功耗估算程序。

在 SimMax 仿真算法中,电路的原始输入信号概率和原始输入信号翻转概率在仿真前就已经确定。信号概率表示在仿真中信号为高("1")的时钟数目与仿真总时钟数目的比率。信号翻转概率表示在仿真中信号翻转次数与仿真总时钟数目的比率。在 SimMax 的仿真中,原始输入信号概率设定为 50%。由于电路输入的高翻转率一般会导致电路内部节点和输出节点的高翻转率,为了估算最大功耗,输入信号的翻转概率应该是尽可能的高。在 SimMax 中我们设定原始输入信号翻转概率为 90%,这是基于如下两个原因:(1)因为导致电路功耗最大的两个输入向量的位相关性未知,因此信号翻转概率不应设定为 1。(2)输入的信号翻转概率设为极高,对电路的某些门并不一定适合,如 XOR 门和 XNOR 门。

在时序 GaMax 仿真中,遗传算法的迭代次数为 100 代。初始解群如下产生:对每个个体的第一输入向量(包括原始输入和状态位)的元素位是随机产生的,其值为 0 或 1 的概率均为 50%。对第二个输入向量的元素,如果是原始输入元素,则以对应位的 90%翻转概率来获得,如为状态位,则不赋值,由逻辑仿真时状态输出决定。定翻转概率为 90%的理由如上所述。表 1、2 分别列出对 ISCAS89 和 ISCAS85 基准电路的仿真实验结果。表 1 中,2-5 列是 ISCAS89 时序电路的参数,6,9 列是最大功耗仿真结果。最大功耗估计均采用公式(3)计算,即用功耗因子来表示,遗传算法的各个参数由前述文中给定。对逻辑仿真方法,最大功耗和运行时间是对 100000 个仿真输入向量仿真后测量的。表 2 中,最大功耗估计采用式(2)计算,其他条件同表 1。从表 1、2 中可以看出本文提出基于遗传算法的最大功耗估计技术具有如下优点:

(1)从实验结果上看,本方法运行时间基本上是电路逻辑门的线性关系,所以比较适用于超大规模集成电路,而且能有一个可靠的功耗估计,而对逻辑仿真来说,电路门数和原始输入管脚数的增多,有限次仿真获得的结果可靠性降低。

(2)本方法对电路最大功耗估计的结果基本上比通过逻辑仿真获得的结果要好或接近,而且原始输入和状态位数越多,效果越明显。

同时应注意到,遗传算法中的几个参数均可修改,如能最优地给定参数,可以获得更进一步的结果。

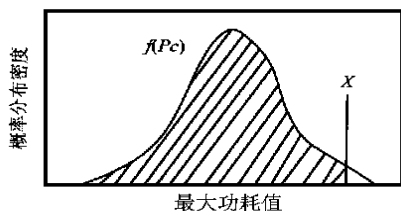


图 3 估计质量

表 1 ISCAS89 时序逻辑电路的最大功耗估计(以功耗因子计)

| 电路 | 电路门数 | 锁存器数 | 输入数 | 输出数 | 遗传算法结果 | | | 逻辑仿真结果 | | |
|--------|-------|------|-----|-----|--------|---------|-------|--------|-------|-------|
| | | | | | 最大功耗 | 时间(s) | 质量(%) | 最大功耗 | 时间(s) | 质量(%) |
| s298 | 119 | 14 | 3 | 6 | 212 | 5.32 | 99.90 | 212 | 57 | 99.90 |
| s344 | 160 | 15 | 9 | 11 | 214 | 6.93 | 99.91 | 206 | 72 | 99.82 |
| s349 | 161 | 15 | 9 | 11 | 215 | 7.02 | 99.91 | 206 | 73 | 99.80 |
| s382 | 158 | 21 | 3 | 6 | 260 | 7.10 | 99.97 | 257 | 76 | 99.96 |
| s386 | 159 | 6 | 7 | 7 | 241 | 6.83 | 99.68 | 239 | 73 | 99.62 |
| s510 | 211 | 6 | 19 | 7 | 227 | 9.42 | 98.07 | 225 | 102 | 97.88 |
| s526 | 194 | 21 | 3 | 6 | 358 | 8.89 | 99.98 | 353 | 96 | 99.97 |
| s820 | 289 | 5 | 18 | 19 | 539 | 13.39 | 99.37 | 532 | 151 | 99.24 |
| s832 | 287 | 5 | 18 | 19 | 545 | 13.47 | 99.34 | 538 | 150 | 99.20 |
| s1196 | 529 | 18 | 14 | 14 | 464 | 22.83 | 99.98 | 460 | 246 | 99.97 |
| s1238 | 508 | 18 | 14 | 14 | 467 | 22.30 | 99.99 | 464 | 241 | 99.98 |
| s1423 | 657 | 74 | 17 | 5 | 918 | 29.42 | 99.99 | 801 | 308 | 99.69 |
| s1488 | 653 | 6 | 8 | 19 | 903 | 28.03 | 98.71 | 898 | 304 | 98.67 |
| s1494 | 647 | 6 | 8 | 19 | 904 | 27.92 | 98.64 | 901 | 302 | 98.57 |
| s5378 | 2779 | 179 | 35 | 49 | 2836 | 122.41 | 99.99 | 2254 | 1294 | 99.99 |
| s9234 | 5597 | 211 | 36 | 39 | 4405 | 230.52 | 99.99 | 3288 | 2369 | 99.99 |
| s13207 | 7951 | 638 | 62 | 152 | 5612 | 337.47 | 99.99 | 3921 | 3418 | 99.99 |
| s15850 | 9772 | 534 | 77 | 150 | 6626 | 404.62 | 99.99 | 4705 | 4122 | 99.98 |
| s35932 | 16065 | 1728 | 35 | 320 | 15887 | 707.67 | 95.25 | 15501 | 7113 | 90.98 |
| s38417 | 22179 | 1636 | 28 | 106 | 18037 | 1036.49 | 99.99 | 13466 | 10004 | 99.99 |
| s38584 | 19253 | 1426 | 38 | 304 | 19003 | 916.32 | 99.84 | 15777 | 9000 | 94.18 |

表 2 ISCAS-85 组合电路的最大功耗估计结果

| 电路 | 电路门数 | 输入数 | 遗传算法结果 | | | 逻辑仿真结果 | | |
|-------|------|-----|--------|--------|-------|--------|---------|-------|
| | | | 最大功耗 | 时间(s) | 质量(%) | 最大功耗 | 时间(s) | 质量(%) |
| c432 | 160 | 36 | 181 | 6.48 | 99.99 | 175 | 70.03 | 99.97 |
| c499 | 202 | 41 | 193 | 8.70 | 99.98 | 190 | 89.91 | 99.96 |
| c880 | 383 | 60 | 349 | 16.88 | 99.99 | 333 | 181.62 | 99.99 |
| c1355 | 546 | 41 | 424 | 20.52 | 99.99 | 419 | 228.73 | 99.98 |
| c1908 | 880 | 33 | 946 | 35.27 | 99.94 | 953 | 364.84 | 99.96 |
| c2670 | 1193 | 233 | 1241 | 52.18 | 99.99 | 1184 | 585.15 | 99.96 |
| c3540 | 1669 | 50 | 1523 | 70.60 | 99.99 | 1497 | 811.97 | 99.98 |
| c5315 | 2307 | 178 | 2689 | 101.85 | 99.99 | 2572 | 1217.31 | 99.99 |
| c6288 | 2416 | 32 | 2666 | 122.93 | 99.99 | 2647 | 1249.88 | 99.99 |
| c7552 | 3512 | 207 | 3695 | 162.87 | 99.99 | 3412 | 1753.69 | 99.94 |

5 结论和今后的工作

本文给出了一种基于遗传算法的 CMOS VLSI 最大功耗估计算法,该算法通过对电路原始输入模式的搜索,以获得能最多或尽可能多的逻辑门实现翻转的原始输入和状态位模式,从而最大化电路工作时的瞬态功耗。本算法在不影响估算精度的前提下,减少电路逻辑仿真的次数,从而减少了运行时间。从实验的结果上看,运行时间基本上是电路逻辑门的线性关系,所以比较适用于大规模集成电路。

参考文献:

- [1] Chuan-Yu Wang, Kaushik Roy. Maximum power estimation for CMOS circuits using deterministic and statistical approaches [J]. IEEE Trans. VLSI Systems, March 1998, 6(1): 134 - 140.

- [2] Chuan-Yu Wang, et al. Maximum power estimation for sequential circuits using a test generation based technique [A]. IEEE Custom Integrated Circuits Conference [C]. 1996: 229 - 232.
- [3] A. P. Chandrakasan, et al. Low-power CMOS digital design [J]. Journal of Solid State Circuit, April, 1992, 27(4): 473 - 483.
- [4] K. Roy, S. Prasad. Circuit activity based logic synthesis for low power reliable operations [J]. IEEE Trans. VLSI System, December, 1993, 1(4): 503 - 513.
- [5] D. E. Goldberg. Genetic algorithms in search, optimization, and machine learning [M]. Mass. Addison-Wesley, 1989.
- [6] D. Lawrence. Handbook of genetic algorithm [M]. New York: Van Nostrand Reinhold, 1991.
- [7] Farzas G, Cecil A. Application of random restart to genetic algorithms [J]. Information Sciences, 1996, 85(1-2): 81 - 102.

作者简介:



卢君明 1974 年出生,1996、1999 年分别获西安电子科技大学工学学士和硕士学位,目前在上海交通大学大规模集成电路研究所攻读博士学位。研究方向为电子设计自动化。主要做 CMOS 数字电路功耗分析与低功耗设计方法等研究工作。

林争辉 1933 年出生,教授,博士生导师,主要研究方向为 ICCAD 及专用集成电路(ASIC)设计。