

面向多级逻辑的低功耗有限状态机状态分配

叶锡恩 夏银水 陶伟炯

(宁波大学电路与系统研究所 浙江 宁波 315211)

摘 要 状态分配在低功耗有限状态机设计中已经被证明是很有效的方法. 该文针对有限状态机多级组合逻辑实现提出了一个新的成本函数, 并利用整体退火遗传算法来进行状态码的搜索, 通过减少电路的开关活动性和组合逻辑部分的面积来达到功耗的降低. 对 25 个有限状态机标准测试电路进行面积和功耗的测试表明: 与已发表的针对面积和功耗优化的算法相比, 该文所提出的算法不但在功耗降低上具有较大的改进, 在面积改善上也具有一定的优势.

关键词 有限状态机; 低功耗; 状态分配; 整体退火遗传算法; 多级逻辑

中图法分类号 TP391

Low Power State Assignment of FSMs Targeting Multi-Level Logic Implementations

YE Xi-En XIA Yin-Shui TAO Wei-Jiong

(Institute of CAS, Ningbo University, Ningbo, Zhejiang 315211)

Abstract State assignment has been shown to be an effective method for low power FSM design. This paper presents a new cost function targeting multi-level combinatorial logic implementation. Using whole annealing genetic algorithm, low power state codes are obtained to reduce the switching activities and the area of the combinatorial logic, which reduce the average power of the FSM. The proposed algorithm is tested on 25 benchmarks. Compared with published area and power optimization algorithms, the proposed algorithm can save much more power dissipation and has advantage on area improvement.

Keywords FSM; low power; state assignment; whole annealing genetic algorithm; multi-level logic

1 引 言

近几年来, 低功耗设计在集成电路设计中变得越来越重要. 为了达到电路的低功耗, 可以使用两种策略: 第一种是通过改进集成电路的工艺来减少电路的功耗; 另一种是通过改进设计流程来达到功耗的降低, 这种方法是与工艺技术无关的. 在逻辑级低功耗设计中, 一种重要的设计方法是通过减少电路

的平均开关活动性来降低电路的平均功耗. 在有限状态机设计中, 通过改进状态分配在一定程度上不但可以改进电路的组合逻辑面积, 也可以降低电路的开关活动性, 从而降低电路的功耗. 但是, 寻找最佳状态分配是一个 NP 问题, 目前国际上利用状态分配进行 FSM 优化已有了不少研究结果^[1-5], 这些研究包括两方面的工作: 估算特定状态分配下功耗大小的成本函数; 在成本函数引导下的搜索算法.

就算法来说, 遗传算法由于搜索空间大, 并具有

收稿日期: 2006-10-16; 最终修改稿收到日期: 2007-05-08. 本课题得到国家自然科学基金(60676017)、浙江省自然科学基金人才专项基金(R105614)、浙江省自然科学基金(Y106818)、浙江省教育厅研究基金(20051732)、宁波市自然科学基金(2006A60091)和浙江省教育厅留学回国基金资助. 叶锡恩, 男, 1955 年生, 副教授, 主要研究方向为低功耗逻辑综合与优化、ASIC 设计技术. E-mail: yexieng@nbu.edu.cn. 夏银水, 男, 1963 年生, 博士, 教授, 主要研究领域为低功耗大规模集成电路综合和优化. 陶伟炯, 男, 1982 年生, 硕士研究生, 主要研究方向为低功耗逻辑综合与优化.

多目标优化的能力而受到青睐. 文献[6]发展了利用遗传算法对状态机面积和功耗同时优化的技术. 文献[7]则实现了基于遗传算法的状态机同时划分和状态分配的低功耗综合技术.

成本函数的建立与 FSM 的结构有密切的关系. FSM 组合逻辑有两种不同实现方式: 二级和多级. 文献[5]利用条件跳转概率作为衡量电路的开关活动性的因子, 通过状态分配使得条件跳转概率高的状态分配汉明距离小的编码, 从而降低电路状态跳转时的开关活动性, 进而降低了电路的功耗. 但是它既没有考虑状态本身的概率, 也没有考虑状态改变所引起的组合逻辑变化而导致的功耗抵消. 文献[8-9]改进了文献[5]的成本函数, 把状态本身的概率考虑在内, 但是它们同样没有考虑组合逻辑面积的影响. 为克服这一缺点, 文献[10]采用开关活动性与立方体数(或文字数)的凸组合作为成本函数, 组合系数为一些经验参数. 文献[6]则讨论了这种凸组合系数大小对综合结果的影响, 提出了两个面向二级逻辑的成本函数. 但这些方法都在一定程度上以面积的增加为代价. MUSTANG^[2], JEDI^[3]和 MUSE^[4]则是面向多级逻辑的面积优化的状态分配算法.

众所周知, 多级电路实现比二级电路实现具有更低功耗和面积, 所以本文着重研究面向多级电路实现的状态分配算法. 由于状态分配是个 NP 问题, 所以本文首先提出一个适于多级逻辑并同时考虑面积和功耗的成本函数, 然后利用整体退火算法作为状态码的搜索算法, 通过搜索最佳或几乎最佳的状态分配来实现 FSM 的优化设计.

2 基于整体退火遗传算法的 FSM 状态分配

遗传算法是模拟生物进化过程的计算模型. 它作为一种新的全局优化搜索算法, 具有简单通用、鲁棒性强、适于并行处理的特点. 它将实际问题中的每个可能解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, 将可能解的编码模拟为染色体. 这样, 从任意一个初始种群出发, 经过选择、交叉、变异三种运算, 产生新一代种群, 经过多次迭代后, 使其收敛于全局最优解或全局次最优解. 理论上已经证明, 遗传算法能从概率意义上收敛于问题的最优解, 但实践表明, 在遗传算法的应用中会出现一些不尽人意的地方, 如易出现早熟现象, 局部寻优能力差等. 模拟退火算法具有较好的局部搜索能力, 这一点

恰是遗传算法所缺乏的, 两者相结合则有可能产生出性能优良的新的全局搜索算法. 整体退火遗传算法就是基于这个思路将退火思想引入选择算子, 并允许父代参加竞争的新型遗传算法. 新算法所产生群体的每一个体都将以概率 1 收敛到整体最优解, 并有较快的收敛速度^[11].

2.1 编 码

编码是整体退火算法中一个重要的环节. 本文中采用十进制编码串来表示每种可能的分配, 也叫一个可能的解, 在遗传算法中称它为染色体. 染色体中每个十进制码叫做基因, 所有可能的染色体组合为一个种群, 一般来说, 这个种群的规模(染色体的个数)有一定的限制, 这是设计者根据问题的复杂度来指定的. 在有限状态机编码的设计中, 假设状态个数为 n , 则采用最小长度编码方式后, 最少需要 $\lceil \log_2 n \rceil$ 个状态变量, $\lceil \cdot \rceil$ 表示取上边界. 我们用 $pop[i][j]$ 来存储一个种群, 其中, i 表示第几个染色体, $i \in \{0, l-1\}$; l 为种群的规模; j 表示染色体中第几个基因; $j \in \{0, n-1\}$. 初始种群是通过随机算法来产生的, 但是得保证同一个染色体中的所有基因具有唯一性, 这是由相同编码不能分配给不同状态所决定的.

2.2 成本函数及适应度的计算

在优化算法中, 成本函数的设计是非常重要的. 这是因为由成本函数可直接获得适应度函数的设计, 而遗传原理则是根据适应度的大小来判断个体是否具有更大的生存机会. 所以, 如果适应度函数设计的不合理, 往往可能使最后获得的“优化”结果并不优化. 成本函数的设计目标是估算方便、快速, 并能客观地反映目标参数.

在 CMOS 电路中, 具有 m 个节点的电路, 动态功耗的计算公式为

$$P_{dynamic} = 0.5 \cdot f_{clk} \cdot V_{dd}^2 \cdot \sum_{i=1}^{i=m} \alpha_i \cdot C_i \quad (1)$$

式中 f_{clk} 表示时钟频率, V_{dd} 是电源电压, C_i 为节点 i 负载电容, α_i 为节点 i 的开关活动性. 在这里假设时钟频率和电源电压都是固定的, 因此影响功耗大小的主要参数为负载电容和节点的开关活动性.

在基于状态分配的 FSM 功耗优化中, 人们已提出了各种成本函数, 如条件跳转概率、总状态跳转概率、开关活动性以及开关活动性与反映面积的参数的组合^[5-14]. 文献[12]提出了一个物理意义明显的成本函数, 不过该成本函数在面积优化上是针对二级逻辑的. 文献[15]提出了两个成本函数, 其中一个利用 SIS^[16] 的 `power_estimate` 命令计算所得的

功耗值作为成本函数. 不过在具体的实现中时间复杂度很大, 对于状态数较大的测试电路几乎没有实用的价值. 另一个成本函数通过减少开关活动性来减少功耗, 但这仅最小化了寄存器部分的开关活动性, 并没有考虑因组合电路部分可能的面积增加而导致的功耗开销. 文献[17]在面向多级逻辑的成本函数中包含了面积因子. 它在成本函数中引入一个参数 $\alpha \leq 1$ 作为开关活动性和面积的权重因子. 但是这些算法实现的功耗降低都是在一定程度上以面积的增加为代价.

由式(1)可见, 在电源电压及时钟频率一定的情况下, 电路的功耗大小与电路内节点电容和节点开关活动性相关. 由于电路的面积与负载电容有较为直接的对应关系, 因此, 可用电路的面积来表示负载电容. 如采用最小编码长度进行编码, 则对一个给定的 FSM, 所需存储单元的个数是确定的, 因而相应的面积也是固定的. 因此, 仅组合逻辑电路部分面积会随着状态分配的变化而变化.

在多级 FSM 电路实现中, 常用文字数(literal)来表示面积. 因此最小化文字数能减少多级电路的节点个数, 也就是减少了多级电路的面积. MUSTANG 中通过最大化电路中公共立方体(cube)的个数来减少多级电路中文字的个数, 从而达到面积优化的目的. 因此, MUSTANG 提出了两种不同的策略, 分别为 *fan_out* 策略和 *fan_in* 策略. *fan_out* 策略考虑 FSM 现态和输出部分, 通过增大出现概率最高的公共立方体的大小来实现, 所以它对于输出变量个数较大的状态机比较有效. 而 *fan_in* 策略考虑 FSM 输入和次态部分, 通过增加最大公共立方体的个数来实现, 对输入变量个数较大的状态机比较有效.

由于状态间跳转概率是衡量开关活动性的重要参数, 且在多级电路中状态跳变对节点开关活动性影响很大, 所以, 本文中利用状态间的总跳转概率和汉明距离的乘积来表示开关活动性, 用 MUSTANG 算法计算得到的权重与汉明距离的乘积来表示节点电容, 这两部分分别用来估算电路的功耗和面积. 然后将这两部分的凸组合作为成本函数, 引导搜索算法实现功耗和面积的同时优化. 据此, 构造成本函数如下:

$$cost = \sum_{i=0}^{i=n-1} \sum_{j=i+1}^{j=n-1} (tp_{ij} + we_{ij}) \times HD(e(s_i), e(s_j)) \quad (2)$$

式中 n 是状态数; $e(s_i)$ 表示状态 s_i 的二进制编码; $HD(e(s_i), e(s_j))$ 表示两个状态编码之间的汉明距离. 汉明距离定义为两个码组对应位置上取值不同

(1 或 0)的位数, 称为码组的距离, 简称码距. tp_{ij} 是状态 i 和状态 j 之间的总状态跳转概率. we_{ij} 为状态 i 和状态 j 之间的面积权重, 定义如下:

$$we_{ij} = \begin{cases} we_{ij_fanout}, & n_o - n_i > v_{th} \\ we_{ij_fanin}, & n_i - n_o > v_{th} \\ we_{ij_fanin} + we_{ij_fanout}, & \text{其它} \end{cases} \quad (3)$$

其中 n_o 为状态机输出变量个数, n_i 为输入变量个数, v_{th} 为设定的阈值, 根据 FSM 输入和输出变量个数的大小来设定状态机该采取怎样的多级面积优化策略. 当状态机输出变量个数与输入变量个数之差大于设定的阈值 v_{th} 时, we_{ij} 取 MUSTANG 中 *fan_out* 策略计算得到的归一化权重值; 当状态机输入变量个数与输出变量个数之差大于设定的阈值时, we_{ij} 取 MUSTANG 中 *fan_in* 策略计算得到的归一化权重值; 其他情况采用两者的和作为 we_{ij} .

由上述成本函数, 可以用成本的倒数来评估适应度, 从而可获得适应度函数:

$$fitness = 1/cost \quad (4)$$

由上式可以看到, 给定状态分配, 成本越低, 适应度越高.

2.3 退火选择

选择操作的目的是为了从当前种群中选出优良的个体, 使它们有机会作为父代繁殖下一代. 判断个体优良与否的唯一准则就是适应度, 即适应度越高, 其被选择的机会就越多. 选择操作实现的方式很多, 这里采用退火选择. 假设对于第 k 代的种群, 用 $p(i)$ 表示个体 i 被选择的几率, $f(i)$ 表示染色 i 的适应度, 则个体 i 被选中的概率可以表示为

$$p(i) = \exp(f[i]/T_k) / \sum_{i=0}^{pop_size} \exp(f[i]/T_k) \quad (5)$$

T_k 为趋向 0 的退火温度, 且 $T_k = 1 / \ln\left(\frac{k}{T_0} + 1\right)$,

pop_size 表示种群规模. 其中 T_0 表示初始退火温度, 这里取 $T_0 = 2 \times gen_size$, $k = 1, 2, \dots, gen_size$, 其中 gen_size 为演化代数.

2.4 交叉

交叉是遗传算法中一个主要的算子, 它是通过交换父代信息来产生较好的下一代. 我们所采用的交叉算子是均匀交叉, 它是通过设定屏蔽字来决定子代的基因继承两个父代的哪个个体的对位基因^[12]. 均匀交叉的操作过程如下: 当屏蔽字中的位是 0 时, 子代继承父代 A 个体中对应的基因; 当屏蔽字为 1 时, 子代继承父代 B 个体中对应的基因. 由此可以产生一个新的子代个体 C, 同理可以产生另一个新的子代个体 D.

2.5 变 异

变异过程是改变染色体中的某些基因,它能增加群体多样性,克服有可能限于局部最优解的弊病.它需要和交叉算子妥善配合使用,使这些过程更具有普遍性.由于所采用的染色体编码是十进制编码,而且应该满足染色体中不能产生相同基因的限制,因此在算法的实现中采取了以根据随机产生的变异率与所设定变异率的比较来逐一检验染色体中每一基因,确定是否需要两两互换的变异方法^[12].

2.6 算法描述

所提出的算法流程如下所示:

1. 读取测试电路;
2. 初始化种群和适应度;
3. 计算状态转移概率 tp_{ij} .

Repeat

4. 通过赌轮方法选择染色体进行繁殖;
5. 通过退火选择、交叉算子产生后代;
6. 变异;
7. 状态分配并为每个染色体同时计算汉明距离和面积权重因子 $w_{e_{ij}}$;
8. 计算后代适应度;
9. 更新种群:根据适应度大小排序,保留好的个体;
10. 演化代数加 1;

Until 收敛或者达到最大演化代数.

为方便,上述算法称之为 WALPMSA (Whole

Annealing genetic algorithm based Low Power Multi-level State Assignment).

3 实验结果

本文所提出的算法已用 C 语言编程实现并通过 Linux 下 GCC 加以编译.算法对 25 个状态数为 6~121 的 MCNC91 标准测试电路进行测试.实验方法如下:首先,用 WALPMSA 算法获得状态分配.实验中,设置交叉概率是 50%,变异率为 6%,种群规模为 60,最大演化代数设为 150 代,初始温度为 300.然后,用 SIS 进行功耗和面积的估算.在功耗估算时,假设电源电压为 5V,时钟频率为 20MHz,并采用零延时模型.

为了验证所提出算法及成本函数的有效性,把由 WALPMSA 获得的实验结果与已发表论文的实验结果进行了比较.在用 JEDI 进行状态分配时,所用的命令为 state_assign jedi.对其他算法则是利用其成本函数来引导状态码的搜索.表 1 分别列出了用相应算法获取状态码后,利用 SIS 进行综合得到的面积和功耗.表中面积(area)单位是 lib2 库的面积单位,它是根据库关联后组合逻辑门单元的面积累加得到的数值.功耗(power)的单位是 μW .

表 1 用 SIS 进行综合后的面积和功耗

测试电路	状态数	JEDI ^[3]		IITG8 ^[15]		POW3 ^[17]		WALPMSA	
		area	power/ μW	area	power/ μW	area	power/ μW	area	power/ μW
dk27	7	36656	240.7	37584	190.6	35728	190.8	32016	167.9
bbtas	6	41296	100.4	38976	99.0	39904	72.3	38976	100.5
ex4	14	91872	466.0	84448	273.8	84912	340.8	77952	305.6
cse	16	227824	454.2	213904	433.7	207872	368.9	219008	406.8
s386	13	117392	409.9	126208	399.6	141984	387.2	119712	362.3
ex1	20	257056	660.5	252880	593.0	238496	599.1	194880	466.8
bbsse	16	126672	530.8	123888	385.3	124352	429.0	115072	354.1
sse	16	121104	511.9	123888	385.3	124352	429.0	115072	354.1
keyb	19	213904	657.2	247312	603.8	246848	662.4	212976	601.2
s27	6	42688	151.5	39440	152.4	37584	152.3	36656	140.7
opus	10	92800	295.0	93264	264.2	97440	261.1	90480	230.7
planet	48	491376	2354.1	586960	2031.1	599952	2189.5	532208	1990.1
sand	32	531280	1475.5	547520	1607.2	561904	1542.8	531280	1440.0
s1488	48	566080	1657.8	636144	1460.2	574432	1346.5	632896	1369.5
pma	24	212512	900.2	207872	787.0	209728	895.8	188848	751.2
s420	18	114608	282.6	117856	265.2	107648	233.1	98832	225.2
dk17	8	61712	372.8	75632	403.1	69136	351.4	59856	315.9
s208	18	110896	233.1	117856	265.2	105792	248.7	110432	189.8
s820	25	335008	1058.3	308096	811.1	295104	762.8	285360	649.3
s832	25	293712	764.6	302992	806.9	288608	708.6	283504	653.6
tbk	32	235248	809.4	230144	860.8	234784	768.2	243600	709.6
dk14	7	99760	509.2	94656	474.1	92336	493.2	88624	486.8
dk512	15	65424	449.2	88624	444.8	88624	400.4	75632	394.8
scf	121	838448	2258.5	956768	1894.2	922896	1576.7	911760	2067.4
mark1	15	81664	363.5	84912	297.0	91872	319.4	83056	284.3

表 2 则给出了各种算法面积和功耗改进的比较, 其定义如下:

$$\%red. = \frac{my_cost - other_cost}{other_cost} \times 100\%$$

(6)

上式中, $\%red.$ 表示面积或功耗的减少, 而 $other_cost$ 则表示由其他算法得到的面积或功耗, my_cost 则是用 WALPMSA 得到的面积或功耗. 从表中可以看出对于大多数测试电路, 在电路功耗方面, WALPMSA 比 JEDI 有明显的优势, 平均功耗比

JEDI 减少 17.4%. 当然, JEDI 是一个针对面积的状态分配工具, 但是即使如此, 利用所建议的算法进行优化不但没有增加面积开销, 而且平均还减少 4.0%. 与针对功耗优化的 IITG8^[15] 相比, 面积和功耗均有较明显的改进, 分别为 7.5% 和 8.8%. 与针对功耗和面积优化的 POW3^[17] 相比, WALPMSA 仍有一定的改善, 虽然面积和功耗的改善相对较少, 分别为 6.1% 和 5.8%.

表 2 各种算法面积和功耗改进的比较

测试电路	状态数	WALPMSA vs. JEDI		WALPMSA vs. IITG8		WALPMSA vs. POW3	
		$\%area_red.$	$\%power_red.$	$\%area_red.$	$\%power_red.$	$\%area_red.$	$\%power_red.$
dk27	7	12.7	30.2	14.8	11.9	10.4	12.0
bbtas	6	5.6	-0.1	0.0	-1.5	2.3	-39.0
ex4	14	15.2	34.4	7.7	-11.6	8.2	10.3
cse	16	3.9	10.4	-2.4	6.2	-5.4	-10.3
s386	13	-2.0	11.6	5.1	9.3	15.7	6.4
ex1	20	24.2	29.3	22.9	21.3	18.3	22.1
bbsse	16	9.2	33.3	7.1	8.1	7.5	17.5
sse	16	5.0	30.8	7.1	8.1	7.5	17.5
keyb	19	0.4	8.5	13.9	0.4	13.7	9.2
s27	6	14.1	7.1	7.1	7.7	2.5	7.6
opus	10	2.5	21.8	3.0	12.7	7.1	11.6
planet	48	-8.3	15.5	9.3	2.0	11.3	9.1
sand	32	0.0	2.4	3.0	10.4	5.5	6.7
s1488	48	-11.8	17.4	0.5	6.2	-10.2	-1.7
pma	24	11.1	16.6	9.2	4.5	10.0	16.1
s420	18	13.8	20.3	16.1	15.1	8.2	3.4
dk17	8	3.0	15.3	20.9	21.6	13.4	10.1
s208	18	0.4	18.6	6.3	28.4	-4.4	23.7
s820	25	14.8	38.6	7.4	19.9	3.3	14.9
s832	25	3.5	14.5	6.4	19.0	1.8	7.8
tbk	32	-3.6	12.3	-5.8	17.6	-3.8	7.6
dk14	7	11.2	4.4	6.4	-2.7	4.0	1.3
dk512	15	-15.6	12.1	14.7	11.2	14.7	1.4
scf	121	-8.7	8.5	4.7	-9.1	1.2	-31.1
mark1	15	-1.7	21.8	2.2	4.3	9.6	11.0
$\%red.$		4.0	17.4	7.5	8.8	6.1	5.8

4 结 论

本文根据 FSM 多级逻辑电路的特点, 提出了一个新的成本函数, 并利用整体退火遗传算法进行低功耗有限状态机的状态分配. 算法已用 C 语言加以实现. 本文通过对 25 个 MCNC91 标准测试电路进行效果验证, 表明所提出的算法与已发表的其他面向多级逻辑的功耗优化算法相比, 在功耗和面积上都具有一定改进.

参 考 文 献

[1] Villa T, Sangiovanni-Vincentelli A. NOVA: State assignment of finite state machines for optimal two-level logic implemen-

tation. IEEE Transactions on Computer-Aided Design, 1990, 9(9): 905-924

[2] Devadas, Ma H T, Newton A R, Vincentelli Sangiovanni. MUSTANG: State assignment of finite state machines for optimal multi-level logic implementations//Proceedings of the International Conference of Computer-Aided Design. 1987

[3] Lin B, Newton A R. Synthesis of multi-level logic from symbolic high-level description languages//Proceedings of the International Conference on Very Large Scale Integration. Munich, 1989: 187-196

[4] Du X, Hactel G, Lin B, Newton A R. MUSE: A multilevel symbolic encoding algorithm for state assignment. IEEE Transactions on Computer-Aided Design, 1991, 10(1): 28-38

[5] Roy K, Prasad S. Syclop: Synthesis of CMOS logic for low power application//Proceedings of the International Conference of Computer-Aided Design. Cambridge, 1992: 464-467

- [6] Xia Y, Almaini A E A. Genetic algorithm based state assignment for power and area optimization. *IEE Proceeding Computer Digital Techniques*, 2002, 149(4): 128-133
- [7] Venkataraman G, Reddy S, Pomeranz I. GALLOP: Genetic algorithm based low power FSM synthesis by simultaneous partitioning and state assignment//*Proceedings of the 16th International Conference on VLSI Design*. New Delhi, India, 2003: 533-538
- [8] Hong S, Pard S, Kyung C. State assignment in finite state machines for minimal switching power consumption. *Electronics Letter*, 1994, 30(8): 627-629
- [9] Wang S, Horng M. State assignment of finite state machines for low power applications. *Electronics Letter*, 1996, 32(25): 2323-2324
- [10] Olson E, Kang S. Low-power state assignment for finite state machines search//*Proceedings of the International Workshop on Low-Power Design*. Napa Valley, 1994: 63-68
- [11] Zhang Jiang-She et al. Whole annealing genetic algorithm and its convergence conditions. *Science in China (Series E)*, 1997, 27(2): 154-164(in Chinese)
(张讲社等. 整体退火遗传算法及其收敛充要条件. *中国科学 (E 辑)*, 1997, 27(2): 154-164)
- [12] Ye Xi-En, Xia Yin-Shui, Tao Wei-Jiong, Wang Lun-Yao. Genetic algorithm based low power FSM state assignment. *Journal of Computer Aided-Design and Graphics*, 2006, 18(12): 1861-1866(in Chinese)
(叶锡恩, 夏银水, 陶伟炯, 王伦耀. 基于遗传算法的低功耗有限状态机状态分配. *计算机辅助设计与图形学学报*, 2006, 18(12): 1861-1866)
- [13] Zhu Ning, Zhou Run-De, Yang Xing-Zi. Low power state encoding using genetic algorithm. *Acta Electronica Sinica*, 2000, 28(8): 124-126(in Chinese)
(朱宁, 周润德, 羊性滋. 应用遗传算法进行低功耗状态编码. *电子学报*, 2000, 28(8): 124-126)
- [14] Olson E, Kang S. State assignment for low-power FSM synthesis using genetic local search//*Proceedings of the IEEE Custom Integrated Circuits Conference*. San Francisco, 1994: 140-143
- [15] Chattopadhyay S, Reddy P. Finite state machine state assignment targeting low power consumption. *IEE Proceeding of Computer Digital Techniques*, 2004, 151(1): 61-70
- [16] Sentovich E, Singh K et al. SIS: A system for sequential circuit synthesis. *Electronic Research Laboratory, College of Engineering, University of California, Berkeley: Technical Report M92/41*, 1992
- [17] Benini L, Micheli G. State assignment for low power dissipation. *IEEE Journal of Solid-State Circuits*, 1995, 30(3): 32-40



YE Xi-En, born in 1955, associate professor. His research interests include low power logic synthesis and optimization, and ASIC techniques.

XIA Yin-Shui, born in 1963, Ph. D., professor. His research interests include low power VLSI synthesis and optimization.

TAO Wei-Jiong, born in 1982, M. S. candidate. His research interests include low power logic synthesis and optimization.

Background

This work is part of the project supported by the National Natural Science Foundation of China under grant No. 60676017. With the rapid increase of chip size, power dissipation has been a critical issue in VLSI design. Low power design technique is a critical need for today's VLSI industry. Since most of digital systems are sequential systems and sequential system is modeled by FSM, it is great important to develop low power FSM design techniques. In low power FSM design, state assignment is one of the important methods. Minimizing the transitions between FSM states can reduce power dissipation and good state assignment can reduce significantly state transitions. However, area tradeoff

may incur. Therefore, a good optimizing technique is required based on state assignment.

The research group has worked on this project for five years and published 11 academic papers in *IEE Proceedings of Computer Digital Techniques*, *Journal of Computer Science and Technology*, *Journal of Computer Aided-Design and Graphics*, and so on. The authors have proposed low power state assignment techniques targeting two level FSM design, partitioned based low power FSM design techniques, etc. This work aims to solve the problem targeting power and area optimization for multilevel FSMs.