

LSIS-II 布图设计系统中的联接 关系正确性验证子系统

高 春 华 庄 文 君

(中国科学院半导体研究所)

1985年4月1日收到

本文介绍了我所研制的 LSIS-II 布图设计系统中的联接关系正确性验证子系统。此子系统系启发式验证系统,以十项验证原则为算法基础,并采用了分级检查的方法,以及为提高错误命中率,运用了定向自锁和区域自锁技术,获得了令人满意的效果,为利用 LSIS-II 系统进行 LSI/VLSI 的设计奠定了可靠的基础。

一、引言

单元(包括伪单元)间联接关系的描述是布图设计对象描述的重要组成部分,联接关系的正确性与否将直接影响设计质量及设计效率。从电路的设计、描述到输入等各个过程中,只要存在人工干预,就不可避免地产生各种随机错误,若不通过某种手段发现其中的错误,势必导致后续布图设计以及制板过程的一系列连锁错误,最终造成废品,对于规模较大的电路,就将带来巨大损失。因而,联接关系的正确性验证是必不可少的。错误的发现并消除是整个系统其它后续部分进行设计的基础与前提。

本验证子系统是根据电路设计的基本原则及人的设计经验,归纳为十项验证原则,对于联接关系中的错误进行检查。理论和实际都充分说明,此启发式验证子系统具有很高的功效,其错误覆盖率达 95% 以上。子系统可以覆盖用户在人工干预各个阶段出现的随机错误,包括输入错误、描述错误以及部分设计上的错误(逻辑错误或电学错误),并对已覆盖的错误具有很高的错误定位功能,错误信息可以十分详细直观,易于用户修改。由于算法中采用了分级检查以及定向自锁和区域自锁技术,使得伪错误和派生错误信息大为减少,有效地提高了错误命中率,从而提高了修改效率,并使子系统具有模块化结构和较大的灵活性。

二、验证子系统的算法基础

1. 验证原则的建立及错误覆盖率的分析

一个验证系统应将验证对象中的错误尽可能多地检查出来,即使得错误覆盖率接近以至达到 100%,而错误覆盖率的高低将取决于用什么方法验证以及所用方法的完备性与否。

我们运用启发性原则, 构造了启发式验证系统。根据电路设计的基本原则和人的设计经验, 归纳为十项验证原则。凡是违反其中任一原则的, 即为错误的。下面我们来讨论一下验证原则的建立与错误覆盖率。

A. 建立验证原则必备的条件

设已建立的验证原则为 A_1, A_2, \dots, A_{n-1} , 新建立原则 A_n ; 设 $A_1, A_2, \dots, A_{n-1}, A_n$ 各自所能覆盖的错误集为 M_1, M_2, \dots, M_n , 则应有:

- i) $M_n \neq \emptyset$,
- ii) M_n 不是 $\bigcup_{i=1}^{n-1} M_i$ 的子集。

我们用右面的集合图表示验证原则覆盖错误的情况。设所有可能出现的错误集合为 E 。

一个验证原则效能的高低, 可以由下面三条来考察, 愈符合者, 效能愈高。

- i) M_n 尽可能大,
- ii) $M_n \cap \bigcup_{i=1}^{n-1} M_i$ 尽可能小,
- iii) 定位程度尽可能高。

那么, 一组验证规则 A_1, A_2, \dots, A_k , 其效能的高低, 可以由下面几点来考察, 愈符合者, 效能愈高。

- i) $\bigcup_{i=1}^k M_i$ 尽可能大。
- ii) 对任意 $1 \leq i \leq k$, $M_i \cap \left(\bigcup_{\substack{j=1 \\ j \neq i}}^k M_j \right)$ 尽可能小。
- iii) 所含原则的个数即 k 尽可能小。
- iv) 定位程度尽可能高。

本验证子系统的十项验证原则如下:

- i) 单元存在性
- ii) 接点存在性
- iii) 元素不唯一性
- iv) 线网独立性
- v) 特殊接点的存在性及唯一性
- vi) 特殊接点的必须连接性
- vii) 接点与线网的对应性
- viii) 单元选择合理性
- ix) 常规原则
- x) 语法规则

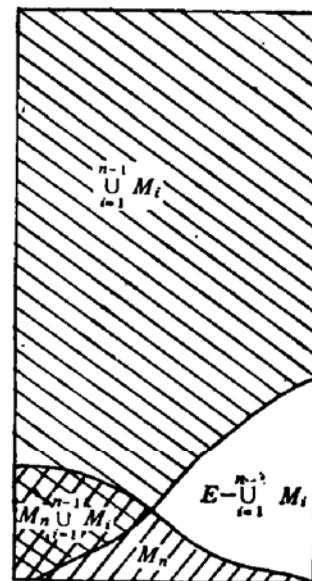


图 1 验证原则有效性示意图

B. 错误覆盖率的分析

考察错误覆盖率须将所有可能出现的错误与由验证规则所能查出的错误做以比较。我们以对线网表中错误的分析为例。

如果我们对于线网表中可能出现的错误加以分析，则可以将其分为若干类。如：

i) 语法错误

ii) 元素号(单元号或接点号)错误

这里又可以分为错写和误写。例如：

错写：100-201 错写成 1000-201。而 1000 号单元不存在。

误写：20-201 误写成 21-201，21 号单元是存在的。

iii) 漏点(特殊情形为漏网)

iv) 多点(特殊情形为多网)

v) 除上述错误以外的设计上的连接性错误。

线网表中出现的随机错误，绝大部分都属上述几类错误，而每类错误一般都相应地可被某些验证规则的并集所覆盖。

如：元素号错误可以由单元存在性，接点存在性，线网独立性及特殊接点存在性及唯一性等原则的并集覆盖。元素号错写可以由单元存在性和接点存在性查出；元素号误写可以由线网独立性及特殊接点存在性及唯一性查出。

如上，由对各种错误及验证规则的功效的分析中可以看出，本子系统的验证规则具有很高的错误覆盖率。

在由十项验证规则所能覆盖的错误域外，还存在如下两种情形：一种情形是，从设计原则和经验规则出发，有些个别类型的错误在理论上就不能被覆盖。如：在两条互相独立的线网中，同样电学性质的两个点被互换。这种错误一般情况下不能被验证规则所覆盖。但实际上这种错误出现的概率非常小。实验结果也证明了这一点。另一种情形是，有个别类型的错误，由设计原则和经验规则出发，其验证结果具有不确定性，而与人的设计意图和具体对象的特点直接有关，最终需人为确定验证结果。

为了进一步提高错误覆盖率，针对第二种情形，除十项原则外，我们还设计了某些临界原则，对处于临界状态的可能的错误，给出警告 WARNING，提醒人们去检查正确与

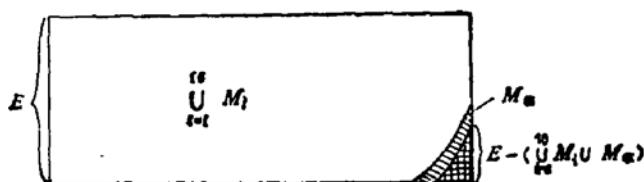


图 2 临界原则提高错误覆盖率的示意图

否，从而发现存在的错误，因而提高了错误覆盖率。如图 2 显示出运用临界原则后覆盖错误的情况。

2. 错误定位

对于验证系统，人们不仅希望能够发现存在的错误，而且希望能给出错误的确切位

置,以便于修改,减少用户的麻烦,提高效率。

本验证子系统,由于验证规则本身大多数具有较高的定位功能,因而在子系统的验证结果中,能够比较精确地给出错误的位置。如:在含有 52 个随机错误的验证实例中,输出错误信息,有 69.2% 定位到一个节点,25% 定位到两个节点,5.8% 定位到一条线网(错误覆盖率为 100%)。

3. 分级检查

由于 LSIS-II 系统对电路的描述采用分级描述,针对这个特点,我们在验证子系统中,采用了分级检查的方法。

分级检查即按照伪单元之间的嵌套关系形成先后次序,从最低级伪单元开始,逐级进行检查(采用伪单元概念进行分级设计,整板图可视为最大的,最高一级的伪单元)。分级检查使得错误检查的效率大为提高,避免了许多重复性检查,也使采用分级描述语言的设计对象在检查时具有更大的灵活性,同时,也使子系统结构十分清晰,成为模块化结构。

4. 提高错误命中率的方法

在通常的验证系统中,为了修改的方便,常常不仅关心错误覆盖率,并且希望有较高的错误命中率,即尽量减少株连错误信息——派生错误和伪错误信息的出现。

由于本子系统的诸验证规则并不是互相独立的,它们各自所能覆盖的错误之间存在交集。如图 3 所示。有时,对于一个错误,用几个原则各自都可查出,但错误定位程度有高低之分。

如: 21-101 是一个输出接点,它被错写为 121-101,而 121 号单元不存在,此错误用①单元存在性②特殊接点的存在性及唯一性③特殊接点必须连接性三个原则均可查出。如果顺序用三条规则查一遍,则相应于这个错误可打出三条错误信息,其中两条为派生错误信息。当这个错误被改正之后,随之三条错误信息就都可消除。

对于不同级伪单元,低一级伪单元中出现的错误可能会影响高一级伪单元中的某些验证,这时就有可能产生伪错误信息。另外,输入信息中某些重要部分产生错误,可能会引起各部分连锁错误,带来大量伪错误信息,有时对于一个错误甚至产生几倍以至几十倍的伪错误信息。例如:当单元号类型对照表中漏掉某个单元号,那么线网表中与此单元相连的接点,其单元号就会被认为不存在,而带来大量伪错误。

为减少派生错误和伪错误信息,提高错误命中率,我们对诸验证原则相应的各类错误信息之间的株连关系进行了研究分析,采用了定向自锁和区域自锁技术,使错误命中率大为提高。

定向自锁是根据各验证原则间的关系,当用某原则发现了某一错误后,对某些与其有派生关系的验证原则是否进行检查或在某范围上是否进行检查加以控制,我们称为“加锁”,从而避免派生错误信息的产生。

区域自锁是根据各描述区域的错误间的株连关系,当发现某一区域内的某种错误后,

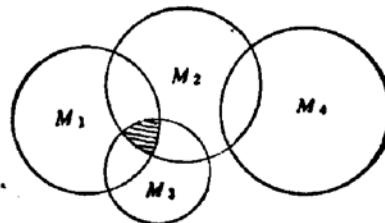


图 3 验证原则不独立性示意图

对受此区域错误严重影响的描述区域是否进行检查加以控制，从而避免伪错误信息的产生。

采用定向自锁及区域自锁技术，既减少了对同一错误的重复的不必要检查，又减少了多余的派生错误和伪错误信息，提高了修改的效率。但另一方面，由于加入“锁”，也有可能将真正的错误锁住，但当前一次运行的错误信息指出的错误被正确修改后，再次检查时，相应于被修改的错误所加的锁就被打开，这个锁所锁住的真正错误就会释放出来，所以，自锁的方法能够保证不降低错误覆盖率，所不同的是错误是分批次查出，分批修改的。前一批改正后，再运行检查出下一批，如此反复。直至无错误信息输出。这使得每批错误信息都具有很高的命中率，大大减少了用户识别是否真错误的工作量，方便了修改。

我们用下边的图 4 来形象表示采用自锁技术后的错误间的自锁关系。

● 表示真错误

○ 表示派生错误

◎ 表示伪错误

图中两点点 A 到 B 之间的有向边表示错误 A 与 B 之间具有自锁关系，即只有当 A 错误被发现并消除后，B 才有可能被检查。

i) A 到 B 的实线有向边记为 W_1 , $A \rightarrow B$ 表示:

客观上，B 是 A 的派生错误信息，我们在 A 错误发现并消除后，才有可能去检查 B（这时，B 错误信息可能由于 A 错误的消除而消失）

ii) A 到 B 的虚线有向边记为 W_2 , $A \rightarrow B$ 表示:

客观上，B 并不是 A 错误的派生错误信息，而由于在检查中加入“锁”的影响，使 A 与 B 之间也产生了“自锁”，即当 A 错误被发现并消除后，B 错误才有可能被检查和发现。

由图我们可以看出，如果不采用自锁技术，本来只需要修改 ● 圈个数的错误（图中 13 个）就可消除全部错误信息，却可能打出所有圈个数的错误信息（图中 49 个）。用户根据 49 个错误信息，最后只需修改 13 个，显然增加了许多识别是否真错误的工作量，而加入了“锁”，就会使 ○ 和 ◎ 信息即派生错误和伪错误信息大为减少，给用户修改带来了很大方

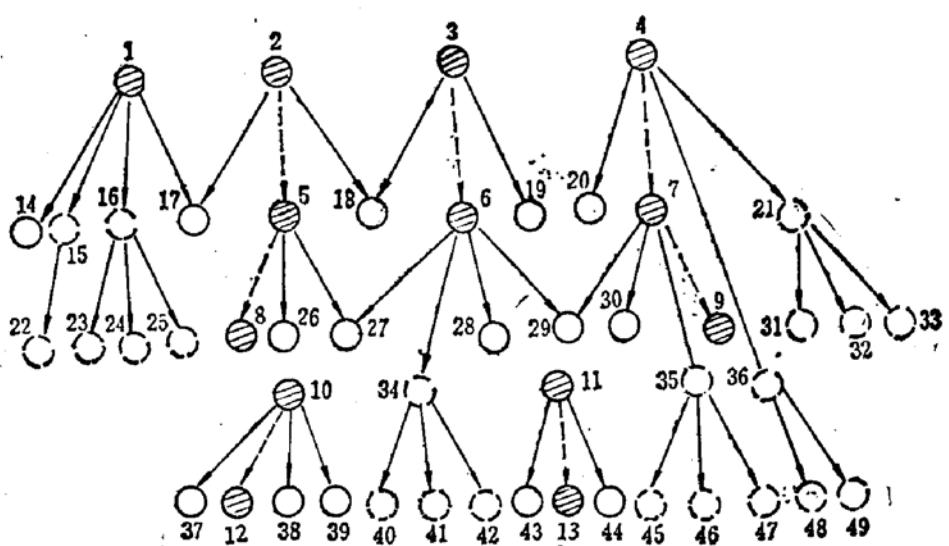


图 4 错误自锁树

便。

错误自锁技术的运用,提高了错误命中率,但一般情况下却有可能增加运行次数。运行次数和错误命中率是一对矛盾,若提高命中率,就有可能增加运行次数,而减少运行次数,又可能使命中率降低。所以,如何兼顾好这两方面,是个很重要的问题,它直接关系到子系统的处理效率。解决这个问题,与检查中如何“加锁”有着密切关系。

下面我们讨论一下如何加锁的问题。定义:最大链长 L_{\max}

设错误自锁图中入度为 0 的各顶点为 $V_{i_1}, V_{i_2}, \dots, V_{i_n}$; 出度为 0 的各顶点为 $V_{j_1}, V_{j_2}, \dots, V_{j_m}$; 设各有向边的边权重为 1; 设任一 V_{i_k} 到 V_{j_p} 的路径为 L_{kp} , 则

$$L_{\max} = \max_{\substack{k=1, \dots, n \\ p=1, \dots, m}} \{L_{kp}\}.$$

那么,若设运行次数为 q (即第 q 次运行后无错误信息输出), 则

$$q = L_{\max} + 2.$$

如何加锁,才能保证有较高的处理效率呢? 我们应使得在加锁后;

- i) L_{\max} 尽可能小,
- 且 ii) 图顶点间的 W_1 尽可能少。(每次运行输出的错误信息为图顶点)

在实际问题中,具体计算 L_{\max} 的长实际上是很困难的。因此,我们在实际处理时,是以上述 i) ii) 为指导,采取对各验证原则及错误的性质及关系进行逐点分析的办法来决定如何加锁。如我们考虑了下列一些因素:

- i) 各原则相应的错误出现的概率,
- ii) 各类错误引起的派生错误和伪错误的概率,
- iii) 对某原则施以加锁后, 它所锁住的派生错误数量和真错误数量的概率的比较。如果 W_1 的个数(即株连错误数量)远多于 W_2 的个数(即锁住的真错误数量)则考虑加锁,反之,不考虑加锁。等等。

如上,由于我们在子系统中,进行了综合分析考虑,采取了较为适度的手段,兼顾了两方面,既保证了有较高的命中率,又使得运行次数在一般情况下增加得很少或不增加。

三、验证子系统的基本算法

1. 验证原则的检查顺序

由于本子系统的十项验证原则间并不是完全独立的, 它们本身存在着一定的内在联系, 从而, 它们各自所覆盖的错误之间也有着一定的联系; 同时, 同一错误可被某几个原则覆盖的错误定位程度也不相同, 因而, 以不同的组合顺序检查, 其产生的效率是不一样的。我们对十项原则以及它们相应覆盖的错误间的联系进行了详尽研究, 安排了较合理、高效的检查顺序, 使算法更为简洁, 时间也更加快。

另外, 在算法实现时, 还兼顾了验证原则间的联系对数据结构设计及内存空间利用率的影响, 尽可能依据它们之间的联系采用更合理的数据结构, 减少存储空间的冗余度。

本子系统的处理时间较快。如在线网表的连接性错误检查中, 其时间复杂性分析如

下：

设 N_p , N_c , N_n 分别为所检查一级伪单元描述的接点数, 单元数和线网数。则各验证原则检查时的时间复杂性为:

原则 i) ii) iv) viii) 为 $O(N_p)$

原则 iii) vi) viii) 为 $O(N_c)$

原则 v) 为 $O(N_n)$

综上, 其整个检查的时间复杂性与验证对象的规模成线性关系。

由于在算法中采用了自锁技术, 可能增加运行次数, 因而对处理效率有一定的影响。

2. 错误局部化处理

为了提高处理效率, 一个合理的验证系统在发现了错误后, 不应立即中断, 而应采取某些措施, 使得能继续检查下去, 从而在一次运行中检查出更多的错误, 提高处理效率。在本子系统中采用了错误局部化方法, 即把错误局限在一个局部化范围内, 避免这种错误影响其它部分的分析和检查。如在算法中以接点为局部化单位即是, 当发现接点的单元号或接点号错误时, 则跳过这一接点, 从下一接点开始做各项检查。算法中有时也以单元或线网为局部化单位。

表 1 实验结果

1. 此实验结果是线网表中的连接性错误的检查结果

电路图序号		第一种电路			第二种电路	
电 路 规 模	单元类型种数	11 种			17 种	
	伪单元类型数	2 种			2 种	
	单元总个数	65 个			100 个	
	线网总数	115 条			224 条	
	有效结点个数	850 左右			1350 左右	
实验序号		一	二	三	一	二
结 果 数 据	随机错误个数	52	44	53	30	40
	查出错误个数	52	44	51	29	40
	错误覆盖率	100%	100%	96%	96.7%	100%
	输出错误信息数	63	50	57	33	44
	派生和伪错误数	11	6	6	4	4
	错误命中率	82.5%	88%	89.5%	87.9%	90.9%
	处理时间(秒)	25.55	28.02	31.03	39.63	28.06

不采用自锁技术运行结果

输出错误信息总数	88	101	100	61	87	68
含派生和伪错误数	36	57	49	32	47	36
错误覆盖率	100%	100%	96.2%	96.7%	100%	97%
错误命中率	59.1%	43.6%	51%	47.5%	46%	47.1%

2. 实例: 八位双向移位寄存器

单元个数 91 错误覆盖率 100%

线网数 113 错误命中率 100%

随机错误数 6 处理时间 2.5 秒

四、实验结果及讨论

本子系统已用一些实验进行了验证,其结果是令人满意的,实验表明,子系统具有很高的错误覆盖率和错误命中率,处理时间也很快。值得提出的是,实验中未被覆盖的错误皆是我们人为故意设置的,其它随机错误都是以 100% 的覆盖率查出。其实,本子系统验证原则不能覆盖的某些错误,在随机错误中出现的概率非常小,所以,一般情况下,本子系统的错误覆盖率几乎达 100%。

实验结果见表 1。

联接关系正确性验证一般有两种手段:逻辑模拟和启发性验证。一般说来,逻辑模拟的方法,虽然能够由人工给定的输入波形,比较精确地给出输出波形,但处理效率较低,规模越大的电路,效率越低。另外错误定位也比较困难,一般需要人工辅助定位。而采用启发性验证,其特点是;处理效率高,定位程度高。而且,事实证明,只要设计合理的话,错误覆盖率也是十分令人满意的。所以,对于实际工程系统,有着重要意义。

在子系统研究与实现过程中,所计算站的同志给予了大力支持,特致感谢。

参 考 文 献

- [1] Gate Mark Manual, 1983.
- [2] Frederick J. hill et al., *IEEE Computer* 7, Dec., 1974.
- [3] W. M. Vancleemput, "An Hierarchical Language for the Structural Description of Digital System," *Computer-Aided Design Tools for Digital System*, 2nd edition, 1979.
- [4] 庄文君等, LSIS-II 布图设计系统,全国第三届 CAD 年会, 1985 年 10 月。
- [5] 高建国等 LSIS-II 分级布图设计描述语言及其编译系统,全国第三届 CAD 年会, 1985 年 10 月。

A Verification Subsystem for the Connectionship of Contacts in LSIS-II Layout System

Gao Chunhua and Zhuang Wenjun
(Institute of Semiconductors, Academia Sinica)

Abstract

The verification subsystem for the connection of contacts in LSIS-II layout system is introduced. The algorithm of this subsystem is based on ten heuristic principles. In the subsystem the hierarchical checking, the directional and regional locking techniques are used for increasing the percentage of hits. Up to now all experiment results are satisfactory. The percentage of covering and hitting to errors is very high. The verification system is an important and necessary part of LSIS-II layout system.