

收稿日期: 2002- 04- 03

利用 Cadence Allegro 进行 PCB 级的信号完整性仿真

Signal Integrity Simulation with Allegro for PCB Board Design

李 新

Li Xin

(西安电子科技大学 西安 710071)

(Xidian University, Xi'an, 710071, China)

张 琳

Zhang Lin

(西安大唐电信有限公司 西安 710075)

(Xi'an Datang Telecom, Xi'an, 710075, China)

摘 要 在高速 PCB 设计过程中, 仅仅依靠个人经验布线, 往往存在巨大的局限性。利用 Cadence 的 Allegro 软件包对电路进行 PCB 级的仿真, 可以最优化线路布局, 极大地提高电路设计质量, 从而缩短设计周期。本文结合作者的实际设计经验, 介绍使用 Cadence 的一般步骤并列举在使用过程中所发现的一些问题。

关键词 高速 PCB 布线 Allegro 文件转换 信号完整性仿真

随着信息宽带化和高速化的发展, 以前的低速 PCB 已完全不能满足日益增长信息化发展的需要, 而高速 PCB 的出现将对硬件人员提出更高的要求, 仅仅依靠自己的经验去布线, 会顾此失彼, 造成研发周期过长, 浪费财力物力, 生产出来的产品不稳定。一般认为高速 PCB 是指其数字信号边沿上升时间小于 4 倍信号传输时延, 这种高速 PCB 的信号线必须按照传输线理论去设计, 否则将会严重影响信号的完整性。Cadence 公司针对 PCB Design Studio 发布一个功能非常实用的高速电路设计及信号完整性分析的工具选件——ALLEGRO PCB。利用这个仿真软件能够根据叠层的排序, PCB 的介电常数, 介质的厚度, 信号层所处的位置以及线宽等等来判断某一 PCB 线条是否属于微带线、带状线、宽带耦合带状线, 并且根据不同的计算公式自动计算出信号线的阻抗以及信号的反射、串绕、电磁干扰等等, 从而可以对布线进行约束以保证 PCB 的信号完整性。下面根据我们的具体实践, 介绍其基本使用方法。

由于我们在实际设计过程中, 通常使用 ORCAD 进行电路前期设计, 得到的是电路的 max 文件。为了利用 Cadence 进行电路仿真, 首先需要将 ORCAD 的 max 文件转换为 ALLEGRO 的 brd 文件。完成这一转换的工具是 Cadence 公司提供的一组附件, 该附件包含 4 个文件, 分别是 Layout.ctl, Layout.fnt, layout.ini 和 T oallegro.exe。使用时, 需要设置好路径和环境变量, 然后运行 T oallegro.exe 文件, 在提示下输入所要转换的 max 文件名, 就可以将 max 文件转换为 brd 文件。但是需要特别注意的是, 转换后的 brd 文件与原来的 max 文件相比有一些隐蔽性的问题, 列举如下:

首先, 元件的焊盘名和封装名会出现问题, 在 ORCAD 中合法的命名规则在 ALLEGRO 中则可能不合法。例如: 在 ORCAD 中可这样定义一个封装名“SBGA\1.27M\131\W31\P304”, 但是在转换至 ALLEGRO 的 brd 文件时会转变为“SBGA127ML31W31P304”; 而在 ORCAD 中定义的焊盘名“9DTI0.038X0.032”在转换至 ALLEGRO 的 brd 文件时会转变为“9DTI0_038X0_032”。即, 它将封装名的“\”删掉, 而将焊盘名的“.”改为“_”。有的焊盘名如果与 ALLEGRO 中的一些关键字重名, 必须将其改名才能转换成功。

第二, 它会给 PCB 自动加上默认叠层, 然而 ORCAD 中没有叠层的选项。

第三, 它不能将原来的各种线宽、间距带到 ALLEGRO 中。

第四, 在将 ORCAD 的 max 文件转换至 ALLEGRO 的 brd 文件后, 有时会发现转换后的 brd 文件在 ALLEGRO 中虽然能够正常打开, 但是却不能正常存盘, 它只能将 PCB 的 brd 文件存为*.SAV 文件。解决的方法是在 DOS 环境下执行 dbfix 命令纠正该错误。

第五, 转换至 ALLEGRO 的 brd 文件的装焊层有些元件值没有带过来, 因此不能在 ALLEGRO 的 brd 文件中输出装焊图。

第六, 也是最重要的一点, 在 ALLEGRO 中自动加上叠层后, 将原来的通孔焊盘按照默认叠层结构自动改变, 如果要调整叠层结构, 就必须对每一个通孔焊盘进行修改, 否则会出现很严重的后果。因为在 ALLEGRO 中每一层都定义了各种焊盘, 根据不同的层

进行选择,相比之下,ORCAD 只在平面层上才定义热焊盘。

在成功转到 ALLEGRO 之后,还需要做些仿真前的准备。首先是根据器件的 Datasheet 对器件的 IBIS 进行检查,检查的内容包括:

①察看 IBIS 库是否有语法上的错误,这个可以在将 IBIS 文件转换成 dml 文件报告时看出,或是在转换后的 dml 文件上,用 ALLEGRO 的工具中的 dml check 选项进行检查;

②管脚的输入、输出类型是否正确;

③VI、VT 曲线是否有明显的非单调性、不连续性或其他明显的错误;

④对所有的管脚模型是否都有 Max, Min, Typical 值以及它们的关系是否正确,如果只有 Typical 值,那么仿真的时候只能用 Typical;

⑤所有的输出和双向管脚模型是否都有测试负载值,即 Cref, Rref, Vref 和 Vmeas,如果是纯容性测试负载,可以没有 Rref;

⑥在标准测试负载情况下,VT 的上升和下降的波形是否达到了 Vmeas 的值;

⑦IBIS 库的管脚是否与器件的 Datasheet 管脚相一致。做完这些检查之后,就要对相应的器件指定各自的 IBIS 库。而对离散器件,则要手工加上 SPICE 库。接着,需要指定 DC 网线的电压值,这是为以后抽取模型时,不至于把 DC 网线当成是信号网线。再下一步进行叠层编辑时,还需要同制板厂商联系,让他们给出满足需要的 PCB 各层的介电常数、介质厚度、铜皮厚度以及叠层的顺序,上述参数要填在叠层表中。

做完准备工作后,就可以抽取网线的拓扑结构进行信号完整性仿真了。仿真分为 2 种:一种是数据线仿真,一种是时钟线仿真,这都是在同步电路下进行的。仿真时,应该从器件的 Datasheet 中查找以下的参数: Tco.min, Tco.max, Tcycle, Tsetup.min, Thold.min; 从所提供的时钟的 Datasheet 中查找 Tskew.clk, Tjit; 另外还需要估计 PCB 的 Tskew.pcb, 给出余量 Tmargin。然后根据以下两个计算公式计算两个参数 Tsettle.delay.max 和 Tswitch.delay.min。这两个算式分别为:

$$T_{settle.delay.max} \leq T_{cycle} - T_{co.max} - T_{setup.min} - T_{skew.clk} - T_{skew.pcb} - T_{jit} - T_{adj};$$

$$T_{switch.delay.min} \geq T_{hold} - T_{co.min} + T_{skew.pcb} + T_{skew.clk}$$

经过仔细分析,发现该公式确实已考虑得非常周全,把上升和下降沿的细微时间以及沿的细小抖动也

包含在内,这可能也是 Cadence 的 PCB 软件中最核心最精华的部分。通过公式算出 Tsettle.delay.max, Tswitch.delay.min 后,需要和根据拓扑结构仿真出来的 Tsettle.delay.max, Tswitch.delay.min 值进行比较,以便用来调整参数。这些参数包括:布线长度,线阻抗,线速,增加减电阻、电容,节点的位置以及线的连接方式等。时钟线还需考虑它的单调性。参数调节是不能随心所欲的,不仅要找一定的规律,还要考虑到 PCB 叠层的参数设置及一些现实所允许的值。这样才能找到适合实际 PCB 的、比较理想的拓扑结构,否则如果在自动布线完成后才发现拓扑结构有问题,就需要来回进行修改。还要注意一点,就是所有以上参数都必须在最坏情况下找出它们的最大范围。所谓最坏情况是指,在仿真 Tsettle.delay.max 数据时,必须是在 slow 的环境下;而仿真 Tswitch.delay.min 数据时,必须在 fast 的环境下进行。在我们仿真的过程中,还发现有些器件的 Tco.min 和 Tco.max 是在不同的测试条件下测试的,在仿真时还要改变 dml 库的测试环境值,才能测出更准确的值。调整好拓扑结构后,千万别忘了填写拓扑结构的限制表,限制表里头才是最终起作用的限制条件。接下来只要将填好限制后的拓扑结构加在同类网线上,然后转到 SPECTRA 中进行自动布线。SPECTRA 会尽量根据限制自动布线,但是,最后在 ALLEGRO 中做检查时,会发现有一些 DRC 报告是因为没有按照限制完成布线,它会提示限制是多长,而实际只布了多长。这时,一种解决方法是删除所有线,改变器件位置,重新自动布线;再一种就需要改变拓扑结构,如果两种方法都失败了,就要考虑到换性能更好的器件。按照以上方法,一块高质量信号完整性的高速 PCB 就已基本完成。

通过使用 Cadence 软件进行信号完整性的仿真,我们体会到该软件具有很强的实用性,对于硬件设计人员来说,是一种不可多得的设计工具,对高速 PCB 来说尤为如此。当然,对于这样一个高智能仿真软件,使用起来也并不那么容易,如果对其内部计算不甚了解,很有可能会因为一两个参数或选项的错误而导致仿真结果失败。希望本文能够起到一定的帮助作用。

参 考 文 献

- 1 Cadence Corp, Allegro PCB Layout System Training Manual
- 2 Cadence Corp, Allegro online reference manuals
- 3 Roy Leventhal, Signal Integrity Board Design & Simulation Techniques

收稿日期: 2002- 04- 15

CORDIC 算法在 DSP 算法硬件实现中的应用进展

CORDIC Algorithm's Applications and Developments in DSP Hardware Realizing

李岩¹ 汪海明¹ 郭士德² 赵建业¹ 余道衡¹Li Yan¹ Wang Haiming¹ Guo Shide² Zhao Jianye¹ Yu Daoheng¹

(1 北京大学电子学系声场与声信息处理国家重点实验室 北京 100871; 2 北京大学遥感所 北京 100871)

(1 Dep. of Electronics, Nat. Lab. of Machine Perception, Peking University, Beijing, 1008712, China;

2 The Institute of Remote Sensing and GIS, Peking University, Beijing, 100871, China)

摘 要 CORDIC 算法被广泛应用于数字信号处理算法的硬件实现中。由于它将许多复杂的算术运算化成简单的加法和移位操作, 因此它在许多 DSP 算法的硬件实现中都有着极为重要的意义。有了它, 许多难于实现而又极具应用价值的算术函数的硬件实现成为了可能。本文首先介绍了 CORDIC 算法的理论概要, 然后给出了 CORDIC 算法在国内外的应用现状。最后, 给出了作者自行设计的基于 CORDIC 算法的可参数化的 FFT 模型。

关键词 数字信号处理 CORDIC 算法 硬件实现

CORDIC 算法 (The Coordinate Rotational Digital Computer) 是 Volder 等人于 1959 年在美国航空控制系统的设计中提出来的, 它是一种用于计算一些常用的基本运算函数和算术操作的循环迭代算法, 其基本思想是用一系列与运算基数相关的角度的不断偏摆从而逼近所需旋转的角度^[1]。从广义上讲它是一个数值性计算逼近的方法, 由于这些固定的角度与计算基数有关, 运算只有移位和加减。可用该算法来计算的函数包括乘、除、平方根、正弦、余弦、反正切、向量旋转 (即复数乘法) 以及指数运算等。1971 年, J. S. Walther 提出了统一的 CORDIC 算法形式^[2], 把圆周旋转、双曲旋转和直线旋转统一到同一个 CORDIC 迭代方程里, 为同一硬件实现多功能提供了前提。随着 VLSI 技术的不断发展, CORDIC 算法越来越受到研究与应用人员的重视, 更加展示出广泛的应用发展前景。

在传统的硬件算法设计中, 乘、除等基本数学函数运算是一种既耗时又占用面积大的运算, 甚至有时是难以实现的, CORDIC 算法正是为解决这种问题而产生的。它从算法本身入手, 将其分解成为一些简单的且在硬件中容易实现的基本算法, 如加法、移位等, 因此使

得这些算法在硬件上可以得到较好的实现。又由于该算法是一种规则化的算法, 它满足了硬件对算法的模块化、规则化的要求, 因此 CORDIC 算法可以充分发挥硬件的优势, 利用硬件的资源, 从而实现硬件与算法相结合的一种优化方案。正是由于上述原因, CORDIC 算法的原始思想一经提出, 就受到了人们的普遍关注, 40 年来人们不断地对其进行探索研究, 并提出了各种改进算法和优化方案以适应各种不同的需求^[3-5]。

我们觉得有必要介绍并讨论这一算法, 使国内注意到 CORDIC 算法的这一新的发展动向。

1 CORDIC 算法理论概要

下面简要介绍一下 CORDIC 算法理论。

设输入为 x_i, y_i, z_i , 参数为 m , 则 CORDIC 算法如式 (1) 所示:

$$\begin{aligned}x_{i+1} &= x_i + m\delta_i\alpha_i y_i \\y_{i+1} &= y_i + m\delta_i\alpha_i x_i \\z_{i+1} &= z_i + \delta_i\theta_i\end{aligned}\quad (1)$$

式(1)中 当 $m = -1$ 时, $\theta_i = \tanh^{-1}(\alpha_i)$;

当 $m = 0$ 时, $\theta_i = \alpha_i$;

当 $m = 1$ 时, $\theta_i = \tan^{-1}(\alpha_i)$ 。

Abstract With the increased demand of high density, high speed board designs, signal integrity simulation is becoming more important. Cadence Allegro is just the tool that can meet the need. It delivers an economic front to back advanced environment for the physical design of complex multi-layer PCB designs. This paper mainly introduces how to using this tools.

Keywords high speed PCB, Allegro file change, signal integrity simulation

作者简介 李 新 男, 西安电子科技大学研究生, 研究方向: 计算机系统结构。