

用于信号完整性的 IBIS 建模与仿真方法研究

学 校:

院 系:

班 级:

学 号:

姓 名:

专业方向:

导 师:

此研究部分由国家自然科学基金 (No. 60572028) 和
上海市浦江人才基金 (07pj14053) 支持

2007 年 12 月

学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：

日期 年 月 日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权 xx 大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保密 ，在__年解密后适用本授权书。

本学位论文属于

不保密 。

(请在以上方框内打“√”)

学位论文作者签名：

日期： 年 月 日

指导教师签名：

日期： 年 月 日

摘 要

深亚微米设计带来了大量的挑战，当涉及到信号完整性时，高速的数字信号不能简单的用逻辑的 0 和 1 来表征，需要使用模拟量来描述，而晶体管级模型的仿真速度是进行系统级高速仿真的瓶颈。

本文描述了一种对数字电路输入输出端口电路进行建模的标准，IBIS。它可以替代晶体管级的模型进行电路仿真和信号完整性分析。文章首先对工作频率在 1GB 的 DDR-SDRAM 的数据输出端口建立 IBIS 模型，并通过特征点比较，差值比较的方法，对模型进行了验证。在此基础上，对输出缓冲器模型仿真方法进行研究，提出一种新的仿真方法，能够消除输出缓冲器 IBIS 模型的瞬态仿真中对伴随电容的重复计算误差。考虑到，经过传输线的信号将带有斜率和延迟，本文对非理想输入给 IBIS 仿真带来的影响做出分析，并且采用一种新的仿真方法，使得 IBIS 模型能够准确的响应非理想信号激励。

关键词：输入输出缓冲器信息标准，输出缓冲器建模，瞬态响应，信号完整性

IBIS MODELING AND SIMULATION METHOD FOR SIGNAL INTEGRITY

ABSTRACT

Deep submicron circuit design brings lots of challenges. When come to signal integrity, digital signal can not be simply treated as logic “1” or “0”. It is useful to describe them as analog signal. But simulation on transistor level is time-consuming which can not be used on system level.

Some basic concepts in IBIS are reviewed in this paper. The process to generate and verify an IBIS output model for a DDR-SDRAM which works on 1 GHz is described. A simulation method is proposed to eliminate the double count of company capacitance in transient simulation. Transition time should be considered when the signal transfer through transmission line, it brings slope to input signals. In this paper, a novel circuit structure is introduced which can improve the accuracy of the non ideal input IBIS simulation.

KEY WORDS: IO buffer information specification, output buffer modeling, transient response, Signal Integrity

目录

1	绪论	10
1.1	引言	10
1.2	信号完整性概述	10
1.3	仿真与建模	11
1.4	缓冲器建模	13
1.4.1	缓冲器电路.....	14
1.4.2	晶体管级模型.....	14
1.4.3	线性模型.....	14
1.4.4	行为模型.....	15
1.5	输入输出缓冲器信息标准	16
1.5.1	IBIS 文件头.....	17
1.5.2	IBIS 成员描述.....	17
1.5.3	IBIS 模型描述.....	18
2	高速 DDR-SDRAM 输出端口 IBIS 建模与验证	21
2.1	SPICE 模型	21
2.2	Power Clamp 和 GND Clamp 仿真	22
2.3	Pullup 和 Pulldown 仿真	23
2.4	Rising Waveform 和 Falling Waveform 仿真	25
2.5	C_comp 和 Ramp 仿真	28
2.6	数据表格精度控制	28
2.7	IBIS 驱动器模型的使用	30
2.7.1	特征阻抗与传输线.....	30
2.7.2	IBIS 模型特征阻抗提取.....	31
2.8	IBIS 模型验证	32
3	IBIS 驱动器瞬态响应仿真方法和改进	37
3.1	IBIS 仿真方法研究意义	37
3.2	输入缓冲器仿真方法	38
3.3	输出缓冲器仿真方法	41

3.4	输出缓冲器瞬态响应仿真方法	42
3.5	输出缓冲器瞬态响应仿真方法实现和改进	46
3.5.1	IBIS 数据预处理	47
3.5.2	IBIS 算法处理	47
3.5.3	宏电路网表实现	49
3.5.4	算法改进与结果比较	51
4	IBIS 驱动器非理想输入仿真方法和改进	55
4.1	仿真环境及方法	56
4.2	Element 仿真	56
4.3	改进方法和仿真结果	59
5	总结与研究展望	62
5.1	总结	62
5.2	研究展望	62
	参考文献	63
	致谢	错误! 未定义书签。
	攻读学位期间发表的学术论文	67

图片目录

图 1 缓冲器线性模型	15
图 2 驱动器 SPICE 模型.....	22
图 3 Power Clamp 与 GND Clamp 仿真电路和波形	23
图 4 Pullup 与 Pulldown 仿真电路和换算波形	24
图 5 参考 GND 的 Rising Waveform 仿真电路和波形.....	25
图 6 参考 GND 的 Falling Waveform 仿真电路和波形.....	26
图 7 参考 Vcc 的 Rising Waveform 仿真电路和波形.....	27
图 8 参考 Vcc 的 Falling Waveform 仿真电路和波形.....	27
图 9 Clamp 数据的区域取点	29
图 10 区域取点与平均取点结果比对	29
图 11 IBIS 驱动器特征阻抗提取和仿真.....	31
图 12 IBIS 模型与 SPICE 模型的仿真比较.....	32
图 13 Typical Case 的差错示意图	33
图 14 Fast Case 的差错示意图	34
图 15 Slow Case 的差错示意图.....	35
图 16 HSPICE 与 PAD 仿真电路	37
图 17 HSPICE 与 PAD 仿真结果比对.....	38
图 18 输入缓冲器仿真电路.....	39
图 19 L=0.2 输入缓冲器仿真结果	40
图 20 L=0.5 输入缓冲器仿真结果	40
图 21 IBIS 输出缓冲器仿真模型	42
图 22 0 到 1 翻转电流过渡曲线（引自参考文献[1]）	43
图 23 IBIS 输出电路模型	44
图 24 IBIS 仿真方法流程	46
图 28 公共索引计算	48
图 26 仿真电路实现	49
图 30 四波形方法仿真结果.....	51
图 28 消除重复电容计算的仿真电路结构	52

图 29 原始四波形方法仿真结果比较	53
图 30 消除电容重复计算后仿真结果比较	53
图 31 IBIS 改进模型仿真环境	56
图 32 B element 仿真电路	57
图 33 B-element 理想输入仿真波形	58
图 34 B-element 2ns 斜坡输入仿真波形	58
图 35 斜坡输入改进电路模型	60
图 36 改进结构 2ns 斜坡输入仿真波形	61

表格目录

表 1 IBIS 验证环境参数	32
表 2 IBIS 与 SPICE Typical Case 仿真结果比较 1	33
表 3 IBIS 与 SPICE Typical Case 仿真结果比较 2	33
表 4 IBIS 与 SPICE Fast Case 仿真结果比较	34
表 5 IBIS 与 SPICE Fast Case 仿真结果比较 2	34
表 6 IBIS 与 SPICE Slow Case 仿真结果比较	35
表 7 IBIS 与 SPICE Slow Case 仿真结果比较 2	35
表 8 传输线长度 $L=0.5$ 仿真结果	41
表 9 传输线长度 $L=0.2$ 仿真结果	41
表 10 四波形方法和改进后的仿真结果比较 1	53
表 11 四波形方法和改进后的仿真结果比较 2	54
表 12 B-element 斜坡输入仿真误差分析	58
表 13 改进结构的斜坡输入仿真误差分析	61

1 绪论

1.1 引言

随着集成电路设计和制造技术的飞速发展，如今主流集成电路工艺尺寸已经到达 65 纳米。系统的时钟频率以 GHz 为单位不断提高，在其中传播的高速信号的边沿则被控制在皮秒(ps)量级，并且该时序级别不仅须维持在芯片级，还须维持在物理尺寸更大的系统电路板(Print Circuit Board)上，此时，互连线的行为不再像简单的理想导线，驱动器和接受器的电气特性也将对系统的正常工作造成严重的影响。

为了使系统正常稳定的工作，必须分析这种高速信号对系统带来的影响，因此我们需要对系统建立合适的模型以进行仿真，建立模型的对象有三个，分别是驱动器(driver)，接受器(receiver)和传输线(transmission line)。对于传输线，传统的 RC 模型已经不能够胜任，取而代之的是更加精确的 RLC, RLCM 模型。对于驱动器和接收器，传统的方法是使用 SPICE 模型，或者线形模型来表示，其中 SPICE 模型的仿真速度不能满足系统仿真的需求，线形模型则太过于粗糙，不适用于高速信号的信号完整性分析，为了解决这个问题，芯片设计厂商与 PCB 供应商创立了一种标准 IBIS(IO Buffer Information Specification)，它规范了一种以数据表格形式表征和描述输入输出端口(I/O)行为的方法。因为 IBIS 可以高速和准确的仿真信号完整性问题，它逐渐成为板级仿真通用的标准。

1.2 信号完整性概述

低速电路板的级仿真，把各个输出输入端口看作是由导线直接相连的节点，数字信号可以无误的在这两个节点传输，但随着工作时钟速率上升，信号有可能在传输时，被干扰或者有反射叠加等情况，则造成一个信号有可能多次翻越阈值电平，造成接受端的误判断，因此，我们需要研究信号完整性问题。

信号完整性主要是指信号在信号线上传输的质量，当电路中信号能以要求的时序、持续时间和电压幅度到达接收芯片管脚时，该电路就有很好的信号完整性。当

信号不能正常响应或者信号质量不能使系统长期稳定工作时，就出现了信号完整性问题。信号完整性主要表现在延迟、反射、串扰、时序、振荡等几个方面[23][24]。

反射(reflection)是由传输线上的波的传输和反射叠加所形成的现象。信号(电压和电流)在信号源端阻抗和传输线阻抗的分压下，其中一部分传播到传输线上并达到负载出，但是有一部分会由于阻抗的匹配问题被反射，成为反射波，反射波沿传输线反向传播，在源端叠加后二次反射，这样不断的反射的过程，在波形中呈现出阶梯状上升下降，这种现象被称为反射。

过冲(overshoot)是第一个超过稳态电压的峰值，对于上升沿，过冲是超过其稳态高电平的高电压，对应的，下冲(undershoot)而对于下降沿是指低于其稳态电平的低电压。在电路中通常会使用二极管来保护器件不受过冲带来的过电压和过电流的冲击，但过多的过冲同时也会导致二极管过早的失效。

振荡(ringing)就是在反复出现过冲和下冲。信号的振铃通常由线上过度的电感和电容引起。

串扰(crosstalk noise)是最常见也是最重要的信号完整性问题。当两条或者两条以上连线相邻时，任意一条信号线上的电压变化会导致周围导线中产生感应电流 从而导致时序不确定 引起逻辑错误

同步开关噪声(Simultaneous Switch Noise)当大量输出同时开启时，将有一个较大的瞬态电流在芯片与板的电源平面流过，芯片封装与电源平面的电感和电阻会引发电源噪声，这样会在真正的地平面上产生电压的波动和变化，这个噪声会影响其它元器件的动作。

1.3 仿真与建模

高速的工作时钟频率不仅仅是在芯片内部，信号也从芯片级传递至电路板级，特别在通信应用中，高速串行接口，高速数字总线协议往往都超过了 GHz。比如时钟频率达到3GHz的串行SATA(Serial Advanced Technology Attachment)接口，最快速度达到10GHz的PCI Express(Peripheral Component Interconnect Express)接口。而即使是低速传输中，由于信号上升沿越来越窄，同样会遇到信号完整性的问题，因此针对这种高速数字电路的设计，新的设计方法学被逐渐的采用[25]。

新的设计方法学中涉及到信号完整性的主要包括以下部分，首先需要理解信号完整性问题的起源和能够解决这些问题的总体方针。然后将总体方针转化为具体的设计规则，在设计早期进行系统级的仿真，并且在整个设计周期进行参数的测量，

通过仿真和建模来优化设计的性能。因此这种新的设计方法学的关键是伴随设计周期的不同层次的仿真和建模。

建模是指为元器件创建一种电器表征与描述，仿真器对它进行解释，并且用计算结果描绘出电压和电流波形。从不同角度，有多种电气模型和仿真器可以用于高速电路信号完整性分析。

- 电磁场模型

绝大多数信号完整性问题可以归结到电磁场理论。电磁仿真使用麦克斯韦方程来分析信号完整性问题，互联线和所有的无源器件被看作是导体和介质，驱动器输入转换为入射电磁波，信号以电磁波的形式传输。由于场分析法过于复杂，因此需要一种简化的分析方法。S 参数即散射参数就是建立在入射波、反射波关系基础上的网络参数，适于微波电路分析，以器件端口的反射信号以及从该端口传向另一端口的信号来描述电路网络。同 N 端口网络的阻抗和导纳矩阵那样，用散射矩阵亦能对 N 端口网络进行完善的描述。阻抗和导纳矩阵反映了端口的总电压和电流的关系，而散射矩阵是反映端口的入射电压波和反射电压波的关系。散射参量可以直接用网络分析仪测量得到，可以用网络分析技术来计算。只要知道网络的散射参量，就可以将它变换成其它矩阵参量。

电磁场模型和仿真器是可以非常精确的仿真非均匀非规则的电路拓扑结构，但是这种仿真方法非常消耗时间，通常只对电路中的关键路径进行仿真。

- 电路模型

电路模型是对电磁场模型的近似，将电磁场模型合理的描述为电阻电容电感等无源器件和电压源电流源等有源器件的组合，在时域中，基于基尔霍夫定理列写电路的改进节点方程，并求解出各个节点的电压电流。常用的电路仿真器是 SPICE。

SPICE 是 Simulation Program with Integrated Circuit Emphasis 的缩写，是由加利福尼亚大学伯克利分校的计算机系创建的模拟电路仿真器。而 SPICE 模拟器支持的电路的网表描述已成为模拟电路和晶体管级电路描述的标准，称为 SPICE 模型。SPICE 的计算引擎非常高效，它仍然是目前众多商业化 SPICE 软件，如 HSPICE，PSPICE 的核心。SPICE 可分析的电路模型包括电阻、电容、电感、互感、独立电压源、独立电流源、各种线性受控源、传输线以及半导体器件等等。

采用 SPICE 模型进行电路信号完整性分析时，分析精度主要取决于模型参数的来源的精确性，这就需要集成电路设计者和制造商提供详细准确描述集成电路 I/O 单元子电路的 SPICE 模型和半导体特性的制造参数，而由于这些资料通常都属于设计者和制造商的知识产权和机密，所以只有较少的半导体制造商会在提供芯片产品

的同时提供相应的 SPICE 模型。同时 SPICE 模型的模拟需要通过求解巨大的矩阵方程，这样电路规模的微小增加就会导致模拟时间的倍增。

- 行为模型

行为模型通常在时域中分析，它是对电路模型的进一步简化，行为级模型非常灵活，根据不同的仿真需求，它可能只是一个匹配阻抗，也可能含有更多的行为描述数据。

在行为模型中元器件被看成‘黑盒子’，我们测量或者模拟其端口的电气特性，记录相应数据，而不涉及器件的工作原理，因此它不会泄露 I/O 单元的内部设计和晶体管制造参数，因而得到了半导体厂商的欢迎和支持。

与电路模型相比，使用行为模型在保持了精确性的同时，仿真时间大大的缩减，从而行为模型比电路模型更加适用于系统级的时序仿真和信号完整性分析。

- 混合信号模型

混合信号模型即 AMS(Analog Mix Signal)模型，通常使用扩展的 AMS 语言来描述，主要包括以下两种，VHDL-AMS 是 VHDL 扩展到描述模拟和混合信号的一种标准语言，Verilog-AMS 是 Verilog 扩展到描述模拟和混合信号的一种标准语言，它的仿真器即采用事件驱动的计算引擎，也采用模拟电路数值计算的计算引擎，从而使得用户获得更大的支持。

在 AMS 语言中由用户来编写描述元器件行为的方程式。采用 AMS 语言所支持的语法来对电路进行不同层次的建模，可以建立其 RTL(Register Transmission Level)模型，也可以建立门级模型，也可以进行 S 域传输函数的建模。

由于 Verilog-AMS 和 VHDL-AMS 都是一种新的标准，只有少数的半导体厂商能够提供 AMS 模型，目前能够支持 AMS 的仿真器也比较少。但 AMS 模型在 PCB 板级信号完整性分析中的可行性和计算精度毫不逊色于 SPICE 模型。

1.4 缓冲器建模

在高速电路的高速信号完整性分析中，我们通常需要对以下三个电路器件建立模型：驱动器，导线，接收器。

高速电路，主要包含两层含意，一是指系统工作频率高，当系统的工作频率很高的时候，导线的行为不再像是简单理想的连接，而会出现高频效应，像传输线一样把电信号传递到相邻器件，或者从相邻器件接受电信号，如果不能合理的分析这种传输线，它们就会破坏系统的时序。二是指信号的上升下降沿变化很快，如果源

端的信号在状态切换的过程中，终端的反射波已经到达，则叠加的波形会影响逻辑判断。

传输线理论和建模已有了广泛的研究，而作为发送端和接收端的缓冲器，如果模型不够准确，不但无法使得精确的传输线模型体现出来，更是会影响系统的信号完整性分析

缓冲器(Buffer)是数字集成电路器件的输入输出端口电路，输出缓冲器也称作驱动器，输入缓冲器也称作接收器。到目前为止，往往将驱动器和接收器当做是阻抗确定的器件，但是对于高速电路，缓冲器的阻抗值是动态变化的，因此需要对它建立更加精确的模型。在数字系统仿真中使用的缓冲器模型通常有三种，全晶体管级模型，行为模型，和线性模型[23][24][25]。

1.4.1 缓冲器电路

下面分析当一个 CMOS 反相器作为输出缓冲器时，我们所关注的行为。当输入为高电平时，NMOS 导通，输出为低电平，当输入为低电平时，PMOS 导通，输出为高电平。CMOS 驱动器的阻抗是动态变化的，它取决于 MOS 管的瞬时工作状态。当 $V_{ds} > V_{gs} - V_t$ 时，NMOS 工作在饱和区，PMOS 工作在非饱和区， $V_{ds} < V_{gs} - V_t$ 时，NMOS 工作在非饱和区，PMOS 工作在饱和区，而阻抗表示为 dV/dI ，既是 IV 曲线的斜率的倒数，也就是斜率越平缓阻抗越大，工作在非饱和区时，IV 曲线近似为一条斜率固定的直线，阻抗值为其斜率的倒数，而在饱和区，阻抗非常大，在考虑阻抗匹配时，需要避免其工作在饱和区。

1.4.2 晶体管级模型

晶体管级模型即为在上文中所提到的基于 SPICE 仿真器的网表模型，这种模型从晶体管级的角度，描述了构成缓冲器的电路结构，并且记录了该晶体管的各种参数，以便形成精确的等价电路，供 SPICE 仿真器调用仿真。

通常来说，这种模型是最精确的，但是由于其仿真时间长，复杂度高，不适用于系统仿真，而通常用来作为线性模型和行为模型构建的标准。

1.4.3 线性模型

在系统设计的初期，往往驱动缓冲器的模型和特性还没有完全的确立，而一个缓冲器的仿真牵涉的许多互相依赖的变量，所以在设计初期，忽略一些复杂的效应，

只对主要关注的仿真器特性进行假定，这样便于在设计早期对系统进行仿真。而在上文中提到过，我们主要关注的是缓冲器的阻抗和上拉和下拉的跃迁过程。

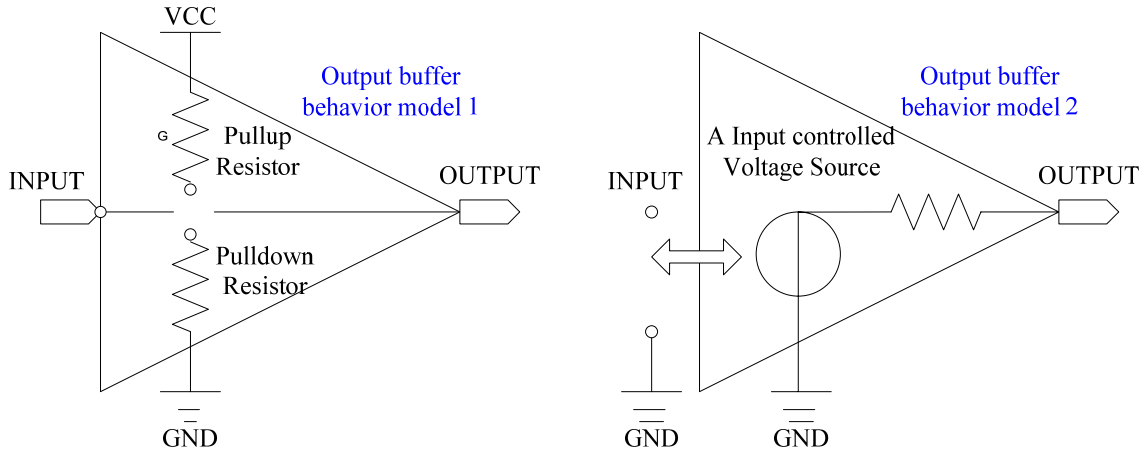


图 1 缓冲器线性模型

如图 1 所示，一个简单的缓冲器线性模型由一个开关和一个上拉电阻，一个下拉电阻构成，其中上拉电阻和下拉电阻的阻抗值分别对应了缓冲器上拉部分和下拉部分在工作区的阻抗值，这样随着开关状态的变化，就会发生上拉跃迁和下拉跃迁，这个模型的主要缺点是不会对边沿变化率进行控制，这只是适用于需要区分开上拉阻抗和下拉阻抗时的粗略计算。

还有一种缓冲器线性模型假定，上拉电阻和下拉电阻都等于传输线的阻抗 Z ，这种建模方法可以设计一个电压源来控制电压随时间变化的关系，从而对边沿变化率进行控制。

从以上两种模型分析，线性模型是建立在缓冲器在工作区时是线性的这一假设上，如果最终设计的缓冲器具有严重的非线性特性，那么使用线性缓冲器模型会得到错误的结果，其次使用线性缓冲器模型仿真时，边沿波形和上升下降时间是预先设定好的，这将无法得到最终系统的准确结果，这种模型通常用于工作在线性区的缓冲器的一阶近似。

1.4.4 行为模型

行为级模型与线性模型有所类似，不同的是它将使用与晶体管曲线类似的 I-V 曲线来表示上拉下拉的过程，而不是用直线来近似。

对于模型中的上拉和下拉部分，行为模型需要给出其导通和截止分别对应的电

流电压(I-V)曲线和电压时间曲线(V-T),这些曲线是指驱动假定负载时对应的实际电压波形,而在模型中使用的假定负载,不必与实际系统中要求的物理负载完全对应,不过如果模型中使用的负载与实际系统中的对应,可以得到更加精确的仿真效果。

实际电路中,上拉器件是不能输出下拉状态的,所以我们认为的在激励置为上拉时,把上拉器件的输出置为下拉状态,即接地或是一个低电平电压源,这样就得到了上拉器件所对应的“截止”行为,同样的我们在激励置为下拉时,把下拉器件的输出接高电平电压源,这样就得到了下拉器件所对应的“截止”行为。

电压电流曲线可以记录不同输出电压下电流的变化情况,仿真器可以结合 VT, IV 曲线,通过各种算法得到不同负载情况下各个时间点的 I-V 信息即瞬态响应。

以上所述的缓冲器的行为可以有各种各样的方法来描述,为了规范和统一这种行为描述方法,一些 EDA 厂商、半导体厂商和大学提出了 IBIS 标准。

1.5 输入输出缓冲器信息标准

如上文所述,缓冲器行为模型是多种多样的,而输入输出缓冲器信息标准 IBIS(IO Buffer Information Specification)规范了输入缓冲器电路和输出缓冲器电路行为的描述。在 1993 年,由一些 EDA 厂商、半导体厂商和大学组成的格式制定委员会推出了 IBIS 的第一个标准 Version 1.0,以后不断对其进行修订。

在 IBIS 标准中,元器件被看成“黑盒子”,我们测量或者模拟其端口的电气特性,记录相应数据,而不涉及器件的工作原理,因此 IBIS 是电路的行为级描述,它不会泄露 I/O 单元的内部设计和晶体管制造参数,因而得到了半导体厂商的欢迎和支持。

与电路模型相比,使用 IBIS 规范的行为模型在保持了精确性的同时,仿真时间大大的缩减,从而 IBIS 比 SPICE 模型更加适用于系统级的时序仿真和信号完整性分析。

需要注意的是,IBIS 规范只是一种方法学,它说明在一个标准的 IBIS 文件中如何记录一个芯片的驱动器和接收器的行为,但并不说明这些被记录的数据如何使用,这些数据需要由使用 IBIS 模型的仿真工具来读取。也就是说,对于已有的电路仿真器,我们需要通过一定的仿真方法来把 IBIS 所记录的数据转化为本电路仿真器所支持的模型。

IBIS 文件采用 ASCII 格式,便于直接阅读和修改。在 IBIS 文件里,记录了电路器件的输入端口和输出端口的行为,以便用于电路板级的信号完整性仿真或者时

序分析。IBIS 标准并不定义文件中得数据如何组成一个可供仿真的模型，它只是用一种规范的格式记录了特定的数据。下面以 IBIS 3.2 版本为例，简单介绍一个 IBIS 文件的组成[13]。

IBIS 文件主要由三个部分组成，分别是文件头，成员描述，模型描述。

1.5.1 IBIS 文件头

IBIS 的文件头中描述文件本身的基本信息。如下所示

```
[IBIS Ver]      3.2
[File Name]    lab_1.ibs
[File Rev]     0.00
[Date]         10/29/2003
[Source]       From silicon level SPICE model
[Notes]        The following information doesn't correspond to any real
                product of any company.
[Disclaimer]   This information is for educational purposes only.
[Copyright]    Copyright 2003, Noname Corporation, All Rights Reserved.
```

其中关键字[IBIS ver]表示 IBIS 的版本号，IBIS 规范具有向下兼容性，高版本的 IBIS 文件包括所有低版本 IBIS 文件所需的数据。

关键字[File Name]表示 IBIS 文件名。

关键字[File Rev]表示文件的版本号。

以上三项是 IBIS 文件头的必选项，下列关键字是可选项

[Data]: 创建日期

[Source]: 本文件中数据的来源，通常分为测量或者是 SPICE 模型仿真。

[Notes] [Disclaimer] [Copyright] 分别记录了一些版权信息和注意事项。

1.5.2 IBIS 成员描述

IBIS 的成员描述主要包含了本 IBIS 文件描述的元件名，引脚，封装参数信息。其格式如下。

```
[Component]    LAB_1
[Manufacturer] Noname Corporation
[Package]
|              typ          min          max
```

```

R_pkg          200.000mOhm    100.000mOhm    300.000mOhm
L_pkg          4.00nH              3.00nH          5.00nH
C_pkg          2.00pF              1.00pF          3.00pF
|
[PIN]  signal_name    model_name      R_pin  L_pin  C_pin
|
1      Vcc            Power
2      GND            GND
3      Input3_3       in33v
4      Input5_0       in50v
5      Data_IO3_3_1   io33v
6      Input5         io33v
7      Data_IO3_3_2   io50v

```

关键字[Component]表示元件描述的开始，每个元件有唯一的一个名字。

关键字[Manufacture]声明了元件生产商的名称

关键字[Package]表示了封装信息，在这里使用电阻电容电感对封装的参数提取，而 TYP，MIN，MAX，则分别对应了三种不同的环境信息。其中只有 TYP 的值是必须的。

关键字[Pin]定义了当前电路的所有引脚，包括了引脚名和此引脚对应的模型名称，不同引脚可能对应于相同的模型，比如上面的 IBIS 文件中，引脚 Data_IO3_3_1 与引脚 Data_IO3_3_2 对应着相同端口模型 io33v。而 io33v 以及其它模型的具体数据会在 IBIS 文件的第三部分记录。R_pin L_pin C_pin 是用来分别记录当前引脚的电气参数，它们不是必需的。

1.5.3 IBIS 模型描述

模型描述记录了缓冲器电路的行为，其格式如下

```

[Model]          in33v
Model_type       Input
|
Vinl = 800.000mV
Vinh = 2.000V
|
|               typ           min           max

```

C_comp	4.550pF	2.600pF	6.500pF
[Voltage Range]	3.300V	3.135V	3.465V
[Temperature Range]	50.0	100.0	0.0
[GND Clamp]			
	Voltage	I(typ)	I(min)
	-3.3E+0	-2.41574407E+0	-1.93394305E+0
			I(max)
			-3.28484653E+0

在模型描述中，IBIS 需要下列的关键字和参数。

[Model]标志着模型描述的开始并且声明一个唯一的模型名。模型名必须与引脚部分定义的名字相匹配。

[Model_type]是说该模型的电路类别，常见的有以下的类别：

Input

输入端口模型，相当于接收端口，在 Input 模型中必须定义其负载电容，和翻转的阈值电压，可以选择性的定义箝位二极管的 I-V 曲线

Output

输出端口模型，相当于驱动器端口，在 output 模型中必须定义其负载电容，稳态上拉下拉曲线，瞬态上拉下拉曲线，可以选择性的定义箝位二极管的 I-V 曲线。

I/O 输入输出端口模型

3-state 三态驱动器端口模型

[C_comp] 伴随电容，不包括封装电容

[Voltage Range] 定义了器件电源电压范围，包括了典型值(type)，最小值(min)，最大值(max)。

[Pullup] 定义上拉的 I-V 曲线

[Pulldown] 定义下拉的 I-V 曲线

[GND Clamp] 定义与参考地相接的箝位二极管的 I-V 曲线

[POWER Clamp] 定义与参考电源相接的箝位二极管的 I-V 曲线

[Ramp] 记录了驱动器模型逻辑状态转换时的斜率。[Ramp]描述了两个参数，dV/dt_r 说明上升时间，是信号上升沿从 20%变化至 80%所花费的时间。dV/dt_f 说明下降时间，是信号下降沿从 80%变化至 20%所花费的时间。

[Rising waveform] 记录了驱动器模型逻辑状态从低到高转换时输出电压随时间的变化关系。

[Falling waveform] 记录了驱动器模型逻辑状态从高到低转换时输出电压随时

间的变化关系。

对于 IV 和 VT 数据，IBIS 使用数据表格的方式来记录，数据用科学计数法表示。其中，每组数据都有三种值，典型 typ，最大 max，最小 min，对应了在不同的温度等外界环境作用下电路的不同行为。典型值是必须的，最大值，最小值是可选的。

2 高速 DDR-SDRAM 输出端口 IBIS 建模与验证

如今，片外存储器的工作频率通常都在 800MHz 以上，如此高速的信号在板级仿真时必须考虑信号完整性问题，而测试一个 RAM，通常要用海量的数据作读写，当仿真的量级在兆 bit 的情况下，采用 SPICE 模型仿真的时间已经无法接受了，此时我们需要对这样的存储器进行 IBIS 建模。

生成 IBIS 模型的方法有测量和仿真两种，测量是指以实际的电路器件为对象，在外围使用仪器加载激励和记录输出数据。仿真是指从一个精确的模型出发，加载适当的激励，仿真得出其输出数据。在本章中我们使用仿真的方法[5][14]，对 Micron 公司的 MT74H 系列的 DDR-SDRAM 的数据输出端口进行 IBIS 建模，并且比较 IBIS 模型和 SPICE 模型的仿真结果。建立一个 IBIS 模型的主要步骤如下：

构造 IBIS 文件结构，记录 IBIS 文件头，以及模型的封装电气参数。

准备好电路的 SPICE 模型网表，进行仿真提取 IBIS 所定义的数据。

将.LIS 文件中的仿真数据整理成 IBIS 格式并且写入 IBIS 文件。

检查 IBIS 语法。

验证 IBIS 模型的准确性。

2.1 SPICE 模型

通常 IBIS 所需要的数据来自与 SPICE 模型的仿真，芯片厂商所给的 SPICE 都是一个加密的电路参数库，和宏电路网表，我们只需要根据不同 IBIS 数据需求连接其电源，地，输入，输出，使能端口，而无需关注电路内部。

我们选用 Micron 公司的 MT74H 系列的 DDR-SDRAM 的数据输出端口作为操作对象，它的电气特性和功能参数在链接[28]中可以下载到。在下面的 IBIS 数据仿真过程中，该 SPICE 模型只是以一个子电路的模式被引用，因为通常从芯片厂商得到的 SPICE 网表都是加密的，从而无须，也无法对其内部进行观测。

在下面的仿真中，我们将以一个内部不可见的子电路来代替实际的加密的 SPICE 模型，如图 2 所示。

```
.SUBCKT IO_buf d_in pad power p_clamp ground g_clamp enable
```

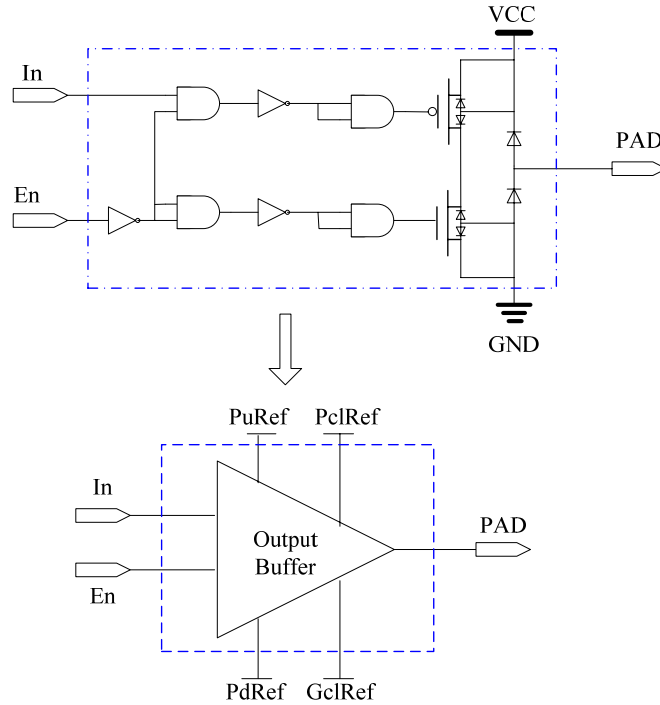


图 2 驱动器 SPICE 模型

2.2 Power Clamp 和 GND Clamp 仿真

电源箝制和电压箝制曲线是衡量输出端口的箝位二极管的响应电流的数据，在仿真时，将输出缓冲器电路的使能端置为无效，对于电源箝制曲线，在输出端加的直流扫描电压源是以 VCC 为参考点的，而对于地箝制曲线，在输出端加的直流扫描电压源是以 GND 为参考点的。

选择 VCC 作为电源箝制曲线的参考点而非接地是为了使得电源轨线噪声和地轨线噪声分别的进行建模，从而能够更加准确的仿真同步开关噪声等问题。

电路连接网表以及图示如下，仿真时，对负载端的直流电压源 Vrefpd 做直流扫描，所得到流过电压源的电流即是 Power clamp 或是 GND clamp。

```

BUFFER d_in pad puref PCLref pdref GCLref en
Vrefpd pdref 0 DC = PD_ref
VrefGNDcl GCLref 0 DC = GCL_ref
Vrefpu puref 0 DC = PU_ref

```


$V_{refPOWcl}$ PCLref 0 DC = PCL_ref
 V_{on} Von 0 DC = Ven
 V_{off} Voff 0 DC = 'PU_ref-(Ven-PD_ref)'

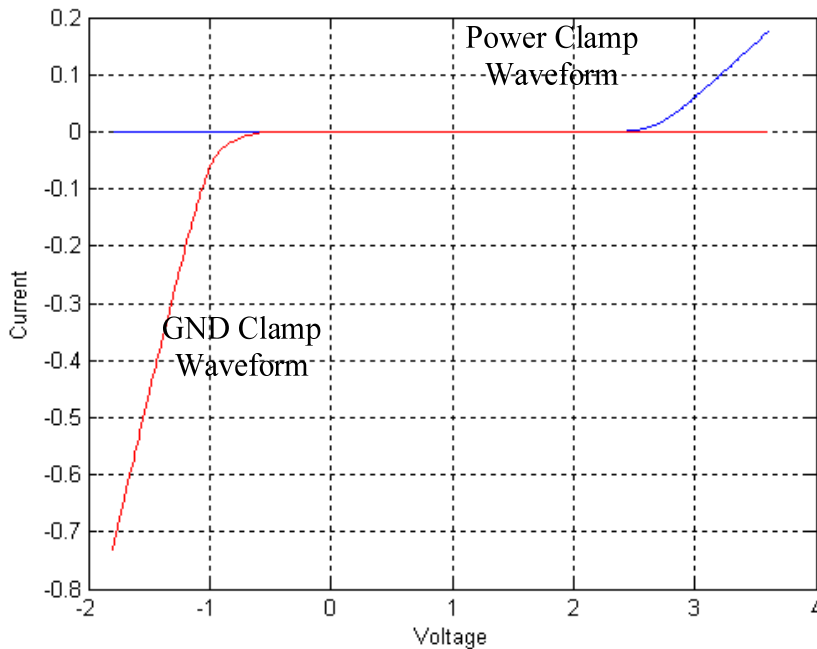
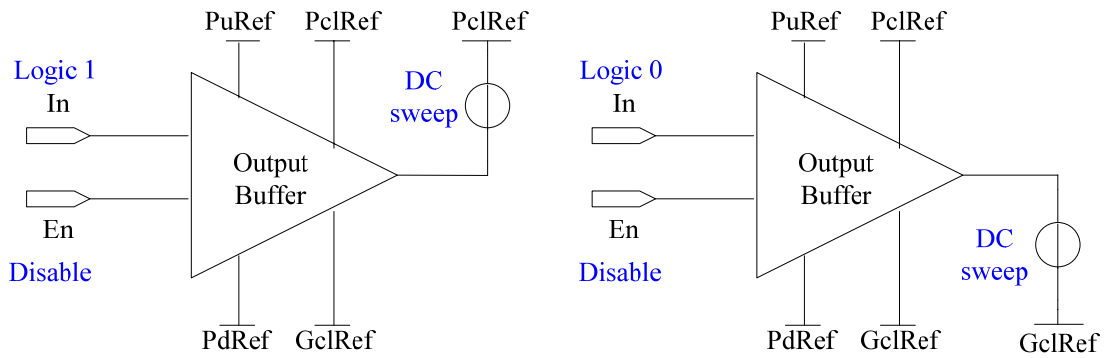


图 3 Power Clamp 与 GND Clamp 仿真电路和波形

需要注意的是，Power Clamp 波形是以电源电压为参考的，所以在同一个坐标系下 $V = V_{cc} - V_{power_clamp}$ 是对 x 轴互反。

2.3 Pullup 和 Pulldown 仿真

稳态上拉曲线和下拉曲线，需要对缓冲器电路的输出电路进行直流扫描。提取上拉曲线时，需要将输入置为高电平，而下拉曲线需要把输入置为低电平。在输出

端加上一个直流电压源，进行从-VCC 到 2VCC 的直流电压扫描。电路的连接方法和上一小节仿真箝制电流的方法一致，只是使能端需要被置为有效。

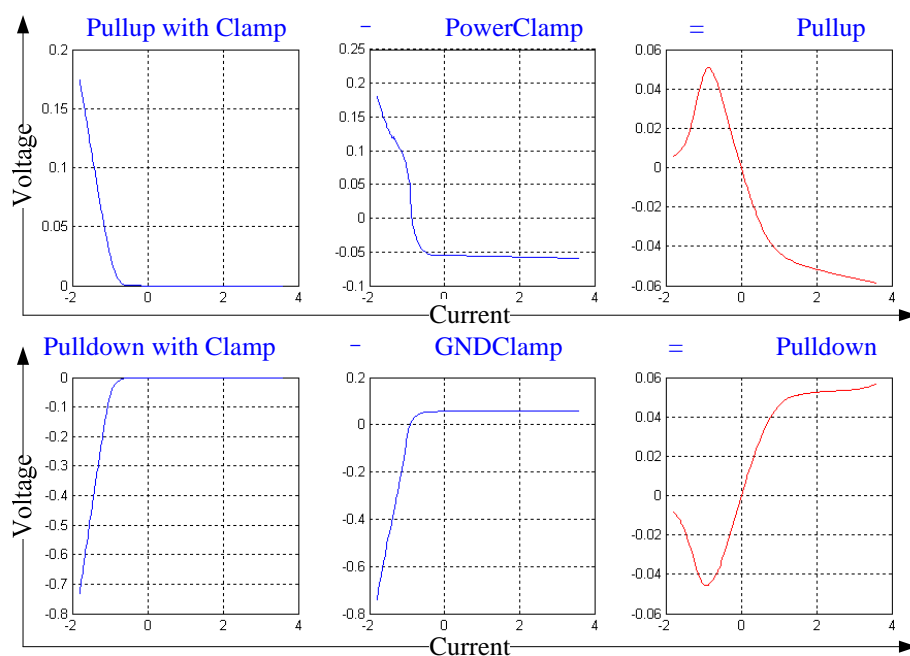
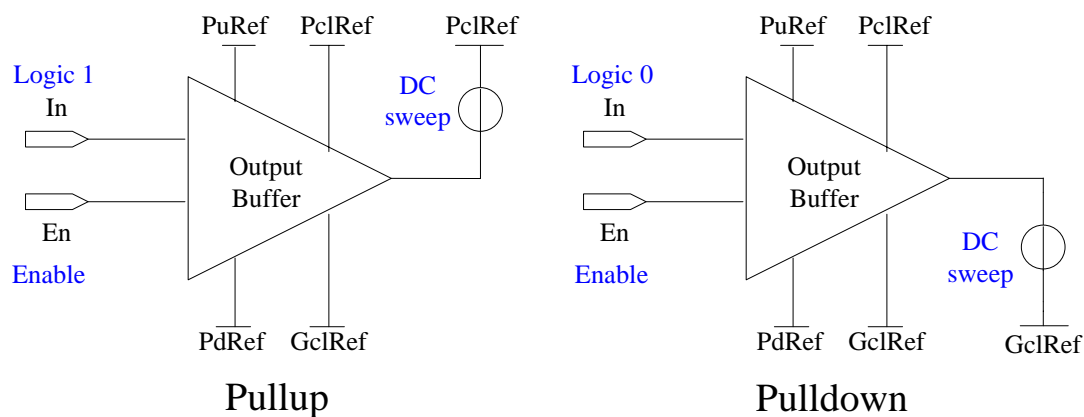


图 4 Pullup 与 Pulldown 仿真电路和换算波形

而这个数据并不是上拉下拉器件的准确稳态行为，它还包含了箝位二极管的电流值，因此从波形来看，上拉曲线和电源箝位曲线，下拉曲线和地下拉曲线是很相似的，为了得到上拉下拉曲线的准确值，我们需要从以上仿真得到的电流值中减去箝位二极管电流。

2.4 Rising Waveform 和 Falling Waveform 仿真

瞬态上拉曲线和瞬态下拉曲线描述了输出缓冲器的瞬态行为，根据不同的激励和负载组合，IBIS 规定了两种逻辑翻转状态，和两种负载连接，在仿真时，通常使用实际电路工作的可能负载作为测试负载，并且数据的记录的零时刻是激励信号进行状态翻转的时间，也就是说输入的激励信号是从“0”时刻的阶跃信号，仿真维持时间需要持续到状态稳定。

如果当前输出缓冲器电路是非反向的逻辑，输入激励是从低电平到高电平的翻转，而负载电路接地，则当电路完成状态翻转后，上拉 MOS 管导通，这就对应着一条 Rising Waveform，称为 Pullup_on。电路网表如下，其中 Rfxt_pu_on 和 Cfxt_pu_on 分别是电容和电阻负载，Vfxt_pu_on 是参考电压。

```
Xdrv_pu_on Pls_r Out_pu_on puref PCLref pdref GCLref Von BUFFER
Rfxt_pu_on V_pu_on Out_pu_on Rfx_pu_on
Cfxt_pu_on V_pu_on Out_pu_on Cfx_pu_on
Vfxt_pu_on V_pu_on 0 Vfxt_pu_on
```

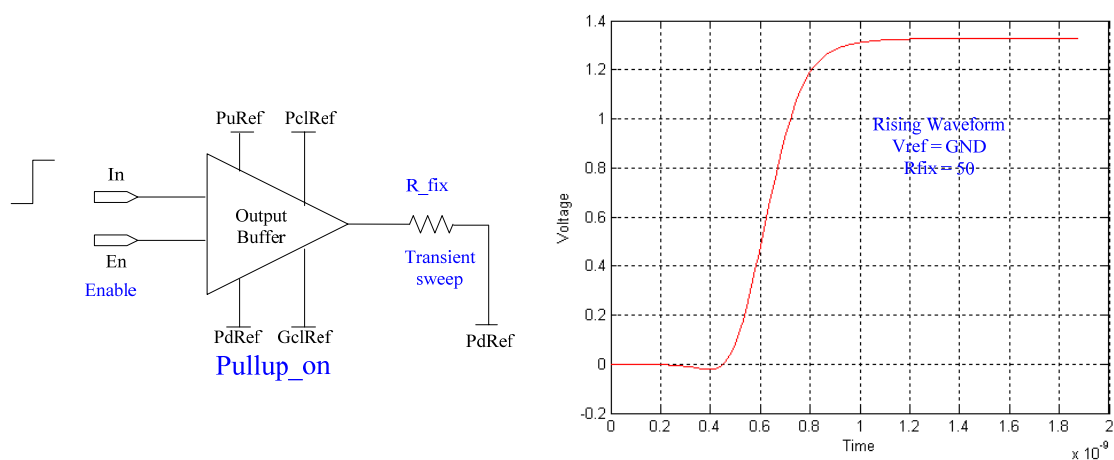


图 5 参考 GND 的 Rising Waveform 仿真电路和波形

输入激励是从高电平到低电平的翻转，负载电路接地，则当电路完成状态翻转后，上拉 MOS 管截止，这就对应着一条 Falling Waveform，称为 Pullup_off。电路网表如下，其中 Rfxt_pu_off 和 Cfxt_pu_off 分别是电容和电阻负载，Vfxt_pu_off 是参考电压。

```
Xdrv_pu_off Pls_f Out_pu_off puref PCLref pdref GCLref Von
```

BUFFER

```

Rfxt_pu_off  V_pu_off  Out_pu_off  Rfx_pu_off
Cfxt_pu_off  V_pu_off  Out_pu_off  Cfx_pu_off
Vfxt_pu_off  V_pu_off  0          Vfx_pu_off
  
```

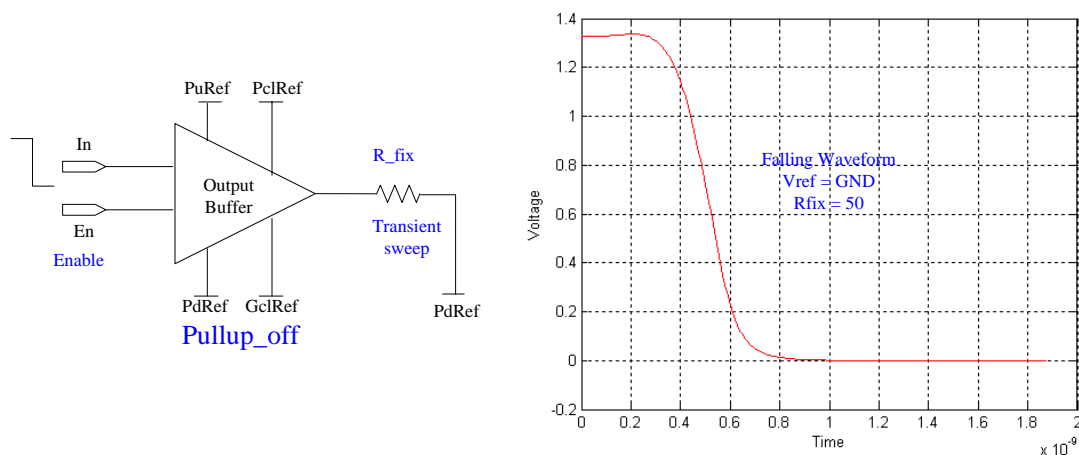


图 6 参考 GND 的 Falling Waveform 仿真电路和波形

输入激励是从低电平到高电平的翻转，负载电路接电源 VCC，则当电路完成状态翻转后，下拉 MOS 管截止，这就对应着一条 Rising Waveform，称为 Pulldown_off。电路网表如下，其中 Rfxt_pd_off 和 Cfxt_pd_off 分别是电容和电阻负载，Vfxt_pd_off 是参考电压。

```

Xdrv_pd_off  Pls_r  Out_pd_off  puref  PCLref  pdref  GCLref  Von
BUFFER
  
```

```

Rfxt_pd_off  V_pd_off  Out_pd_off  Rfx_pd_off
Cfxt_pd_off  V_pd_off  Out_pd_off  Cfx_pd_off
Vfxt_pd_off  V_pd_off  0          Vfx_pd_off
  
```

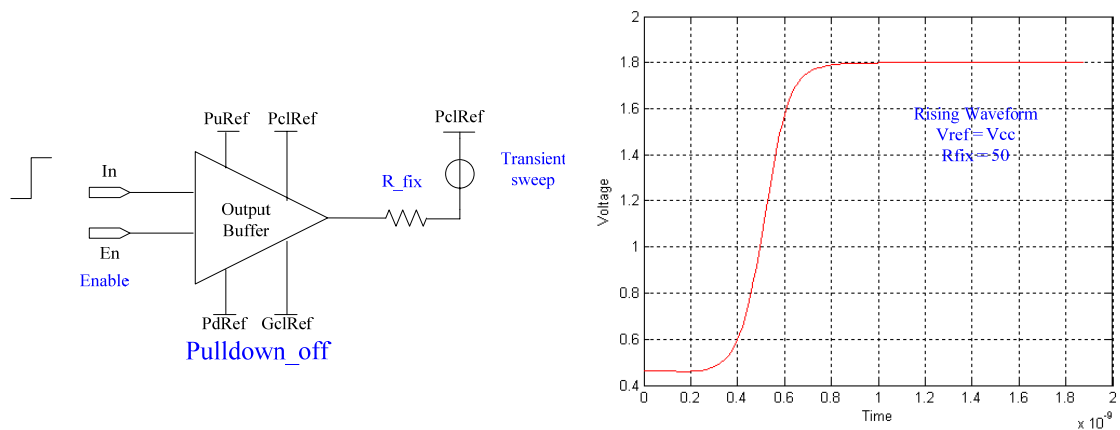


图 7 参考 Vcc 的 Rising Waveform 仿真电路和波形

输入激励是从低电平到高电平的翻转，负载电路接电源 VCC，则当电路完成状态翻转后，下拉 MOS 管导通，这就对应着一条 Falling Waveform，称为 Pulldown_on。电路网表如下，其中 Rfxt_pd_on 和 Cfxt_pd_on 分别是电容和电阻负载，Vfxt_pd_on 是参考电压。

Xdrv_pd_on	Pls_f	Out_pd_on	puref	PCLref	pdref	GCLref	Von
BUFFER							
Rfxt_pd_on	V_pd_on	Out_pd_on	Rfx_pd_on				
Cfxt_pd_on	V_pd_on	Out_pd_on	Cfx_pd_on				
Vfxt_pd_on	V_pd_on	0	Vfx_pd_on				

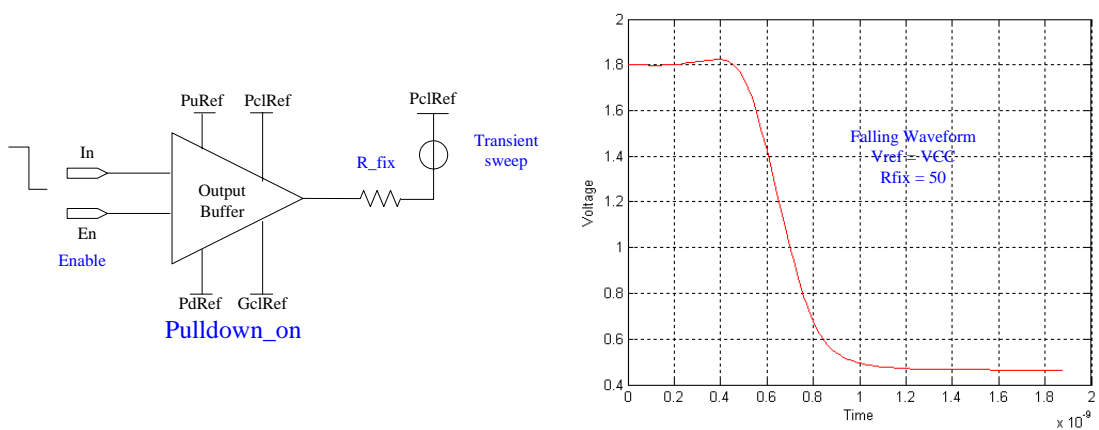


图 8 参考 Vcc 的 Falling Waveform 仿真电路和波形

2.5 C_comp 和 Ramp 仿真

在 IBIS 较早的版本里，使用瞬态响应曲线的平均斜率来描述上拉和下拉的行为，其中上拉 Ramp 值是当电压输出稳态值的 90% 减去电压输出稳态值的 10% 比上从输出电压到达稳态值的 10% 上升至 90% 的时间。其中下拉 Ramp 值是当电压输出稳态值的 90% 减去电压输出稳态值的 10% 比上从输出电压到达稳态值的 90% 下降至 10% 的时间。在 IBIS1.0 版本之后，已经普遍使用 Rising Waveform 和 Falling Waveform 来刻画瞬态行为。

C_comp 是指电路的伴随电容，根据输出端口的电流表达形式 $i = C \cdot dV/dt$ 可知， $C = i/(dV/dt)$ 。

2.6 数据精度分析与控制

IBIS 文件是 ASCII 格式，可以用所有文字处理软件将得到的以上仿真数据，按照 IBIS 规范的格式整理成一个完整的 IBIS 文件，通常主要步骤包括只包含 IBIS 关键字的标准 IBIS 文件模版生成和数据填充。

以符号 [] 代表其中间是关键字，另外，需要在关键字的相邻行标注出 typ min max，这一行必须以 IBIS 中的注释符号 | 其实。数据将被填写在相应的关键字和下一个关键字之间。在文件的最后，以关键字 [END] 表示结束。

需要注意的是 IBIS 使用数据表格的方式来记录仿真结果，而且规定每一个数据表格只允许选取 100 点数据，因此如何选取合适的数据点使其能够更加准确的刻画电路的行为是 IBIS 建立准确模型很关键的一步。

对输出缓冲器电路而言，从信号完整性的角度，我们更加关注的上冲，下冲，振铃，地弹等现象，以及在一些特殊点，比如 0 电压或 VCC 电压以及阈值翻转电压时仿真结果的数值。

通常，可以通过控制 SPICE 仿真的步长来得到这 100 个点，虽然可以通过线形插值来使得数据连续和平滑，但是在仿真的步长限制有可能丢失极值点。

对于 pullup 和 pulldown 曲线，IBIS 规范中，它们必须从 -VCC 扫描至 +2VCC，然而上拉下拉器件主要工作在 0 到 VCC 这个电压区间内，相反的，对于 POWER Clamp 和 GND Clamp 曲线而言，它的工作范围主要是在 -VCC 到 0 以及 VCC 到 2VCC

区域，根据这样的想法，我们可以把各条曲线按其工作区域划分，按照不同工作区域的重要性来确定步长，而非使用平均步长。对于 Clamp 曲线，我们主要对其工作区域取点，Power Clamp 的取点区域主要为 V_{cc} 到 $2V_{cc}$ ，而 GND Clamp 的取点区域为 $-V_{cc}$ 到 0，如下图所示。

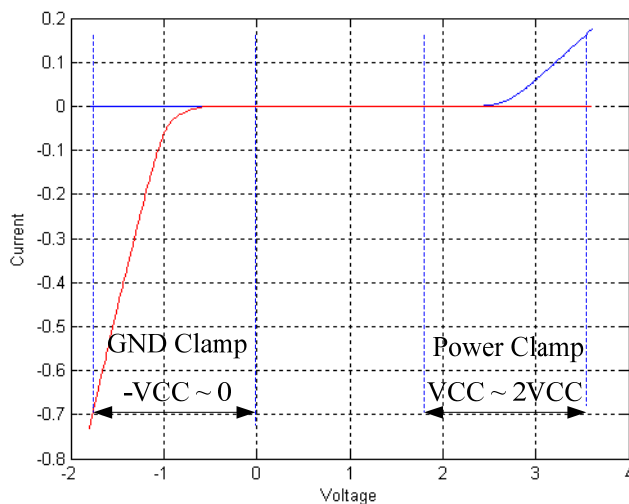


图 9 Clamp 数据的区域取点

对于输出缓冲器电路的 pullup 和 pulldown 曲线，我们将其分为三个区域，一是从 $-V_{CC}$ 到上拉 PMOS 阈值电压即 $-V_t$ ，第二个区域是从 PMOS 阈值电压值 $(-V_t)$ 到 NMOS 阈值电压 (V_t) ，第三个区域是 V_t 到 $2V_{CC}$ 。三个区域分别取 20 点 60 点与 20 点数据，如图 14 所示，如果平均取点，下冲电流的峰值会被丢失。

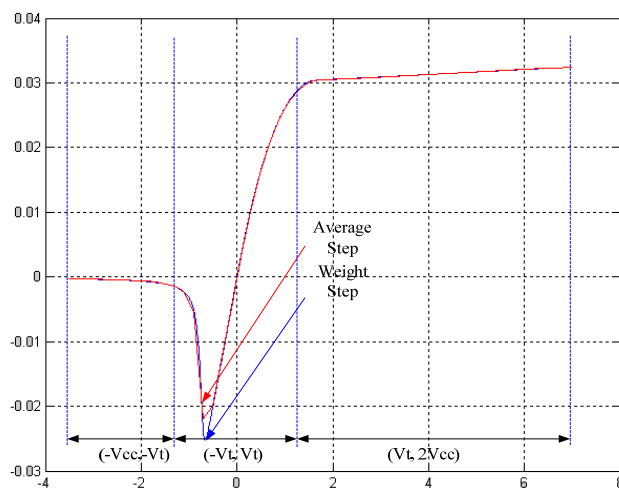


图 10 区域取点与平均取点结果比对

2.7 IBIS 驱动器模型的使用

IBIS 记录了缓冲器电路的行为，电路设计者可以根据其数据获得自己希望的信息。比如封装参数等电气特性，是直接记录在 IBIS 文件中的。还有翻转的阈值电压也是直接记录在 IBIS 数据中的，因此一个简单的驱动器行为可以用一个电压控制电压源来表征，即越过阈值电压时输出电平翻转。

而针对有一定精度需要的信号完整性仿真，我们还需要掌握缓冲器的其它数据，特别的对于传输线分析，我们需要知道缓冲器的阻抗值，从而以入射波和反射波的形式模拟信号的传输过程。

2.7.1 特征阻抗与传输线

在高速电路中，导线不能仅仅看作是连接关系还需要对其电气特性进行分析，即视其为传输线。导线会被看作传输线处理主要取决于导线的长度以及在此导线上传播的信号的速度以及信号的翻转时间(上升下降时间)。

对于信号的传播速度而言，虽然在金属导线中，电流是以电子的形式形成和流动的，但是信号的速度并不是电子迁移的速度。导线中的信号传播速度是在这种导线介质中电场的建立速度，与光速和介电常数有关。

在零时刻，当需要传播的信号加载在导线上时，会建立起电压差，随即这个电压差会在导线上相邻的区域形成电场，同时传播这个电压差，就好像是信号在这条传输线上传播一样。建立起电压差，就需要有电流流动，因此，信号在传播过程中建立这样的电压差会导致一定的电流，这个电压与电流的比值成为传输线的阻抗，特别的如果传输线介质均匀几何形状规整，则整条传输线的阻抗是恒定的，则把它称为传输线的特征阻抗。

在信号传输到终端时，会遇到一个终端的阻抗，这个时候为了保证在这个终端上建立起电压差，信号会对这个终端注入电流，而对于与这个终端阻抗连接的传输线而言，电流必须是连续的，所以当终端的阻抗不等于传输线阻抗时，会有反射出现，反射系数的计算如下：

$$\rho = \frac{V_{\text{reflected}}}{V_{\text{incident}}} = \frac{Z_t - Z_0}{Z_t + Z_0} \dots\dots\dots(2.1)$$

其中，Z₀ 是传输线的特征阻抗，Z_t 是终端的阻抗，如果 Z_t=Z₀ 则称作端接阻抗匹配，另外，特别的，如果终端开路时 Z_t 无穷大，则反射系数为 1，如果终端短路则反射系数为-1。

如上所分析的，当入射波达到终端时，信号的一部分会反方向的反射回去，并与入射波叠加，沿传输线反向传播，到达信号源时，在发生一次反射。这种反射和逆反射将不断的进行，直到传输线电压逼近稳态对应的电流值为止。

2.7.2 IBIS 模型特征阻抗提取

输出缓冲器的特征阻抗并没有被直接记录在 IBIS 数据中，但是相关信息可以从器件的稳态上拉下拉曲线中获得。对于上拉和下拉两条曲线，我们可以分别提取上拉特征阻抗和下拉特征阻抗。

根据上文中分析，特征阻抗是信号传输过程中电压和电流之比， $Z_{out} = \Delta V / \Delta I$ ，以上文中提取的输出缓冲器为例，计算其平均特征阻抗 $Z_{out} = |(V2 - V1) / (I2 - I1)|$ ，与上文中获得上拉下拉曲线相似的，在这里的电流 I 是除去了箝位二极管作用的电流，如果直接用上拉下拉电流计算平均值，将得到非常小的阻抗，是因为，箝位二极管在工作时将输出短路至电源轨线或是地。

以上文的 IBIS 模型为例，提取出其上拉特征阻抗为 34.3 欧姆，与微带线模型相连，终端开路，微带线模型网表如下：

```
Uload Out_pd_on 0 t_out 0 micro1 L=0.4
```

```
.model micro1 U LEVEL=3 PLEV=1 ELEV=1 wd=2m ht=2m th=0.25m KD=5
```

分析仿真波形可以看到，从 IBIS 数据中获得的特征阻抗信息可以较为准确的仿真出入射和反射波在传输线上的情形，仿真波形如图 15 所示。

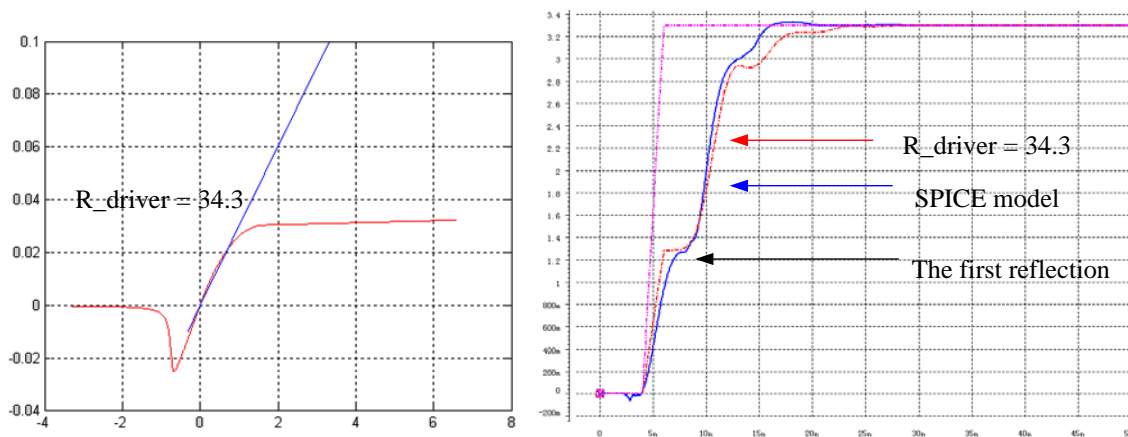


图 11 IBIS 驱动器特征阻抗提取和仿真

2.8 IBIS 模型验证

IBIS 在越来越广泛的被接受的同时，一些新的要求和问题也随着用户对 IBIS 的使用而显现出来，其中最重要的一个问题就是 IBIS 模型的验证。在将 IBIS 应用于大规模的系统仿真之前，必须要保证 IBIS 模型正确表达了电路，还需要验证其仿真精度能否达到系统仿真的需要。

IBIS 模型的验证，分为两个方面，一方面是对 IBIS 文件格式的检验，另一方面是对 IBIS 模型准确性的检验。下面我们通过将 IBIS 模型与 SPICE 模型同时进行仿真，对比其分析结果，来验证 IBIS 模型。

我们根据 Datasheet 中所提供的负载情况来做仿真，其中各参数如下表所示

表 1 IBIS 验证环境参数

仿真实例	温度	电源电压	负载
Typical Case	40	1.5v	R=25
Fast Case	0	1.55v	R=25
Slow Case	100	1.45v	R=25

使用 HSPICE 作为仿真器，仿真波形如下：

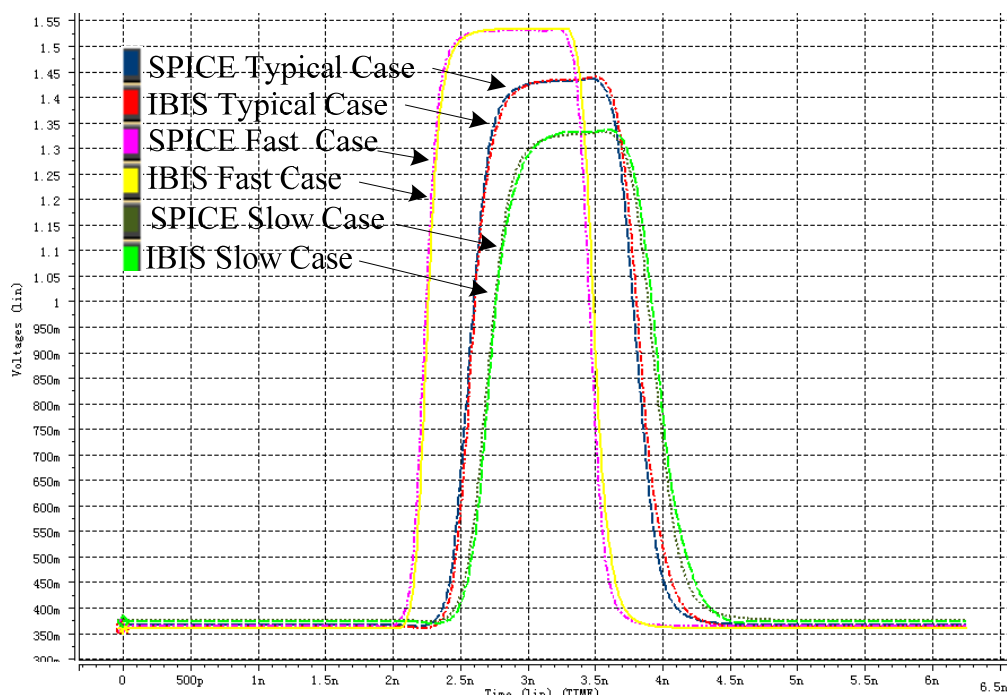


图 12 IBIS 模型与 SPICE 模型的仿真比较

一般的对于 IBIS 的验证而言，我们以 SPICE 仿真数据或者实际测量数据为基准，计算相同条件下，每个点上 IBIS 到这些被视作 golden 的数据的仿真误差，从而通过最大的峰值误差，以及平均误差来衡量此 IBIS 模型的准确性。另外，特别的对于 RAM 而言，我们会关注他的 transition time 和翻越阈值电压的时间点，我们把 SPICE 模型和 IBIS 模型的上述特征列表比较如下：

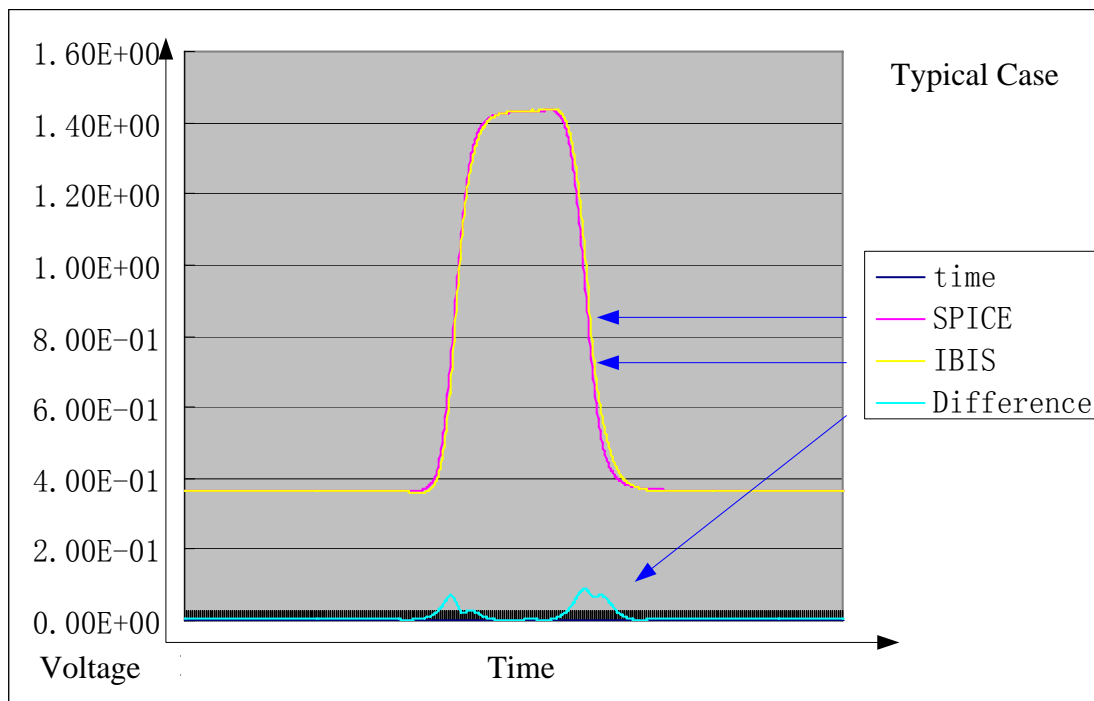


图 13 Typical Case 的差错示意图

表 2 IBIS 与 SPICE Typical Case 仿真结果比较 1

数据来源	上拉 Transition time	下拉 Transition time	阈值电压时间 上拉	阈值电压时间 下拉
SPICE 模型	0.2875ns	0.29875ns	2.53125ns	3.85625ns
IBIS 模型	0.28035ns	0.29325ns	2.55000ns	3.88000ns

表 3 IBIS 与 SPICE Typical Case 仿真结果比较 2

数据来源	峰值误差	峰值误差比	平均误差	平均误差比
IBIS 模型	0.090421v	8.427%	0.010077v	1.0004%

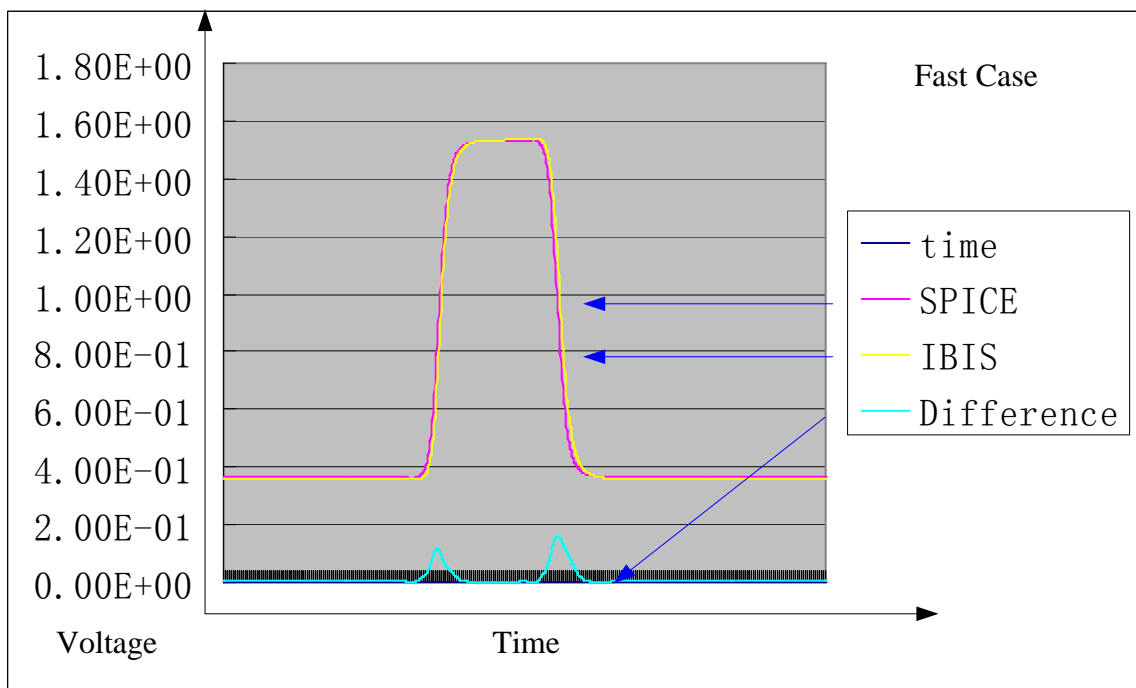


图 14 Fast Case 的差错示意图

表 4 IBIS 与 SPICE Fast Case 仿真结果比较

数据来源	上拉	下拉	阈值电压时间	
	Transition time	Transition time	上拉	下拉
SPICE 模型	0.25625ns	0.26250ns	2.19300ns	3.52325ns
IBIS 模型	0.25625ns	0.28125ns	2.21250ns	3.44025ns

表 5 IBIS 与 SPICE Fast Case 仿真结果比较 2

数据来源	峰值误差	峰值误差比	平均误差	平均误差比
IBIS 模型	0.16181v	13.807%	0.013065v	1.115%

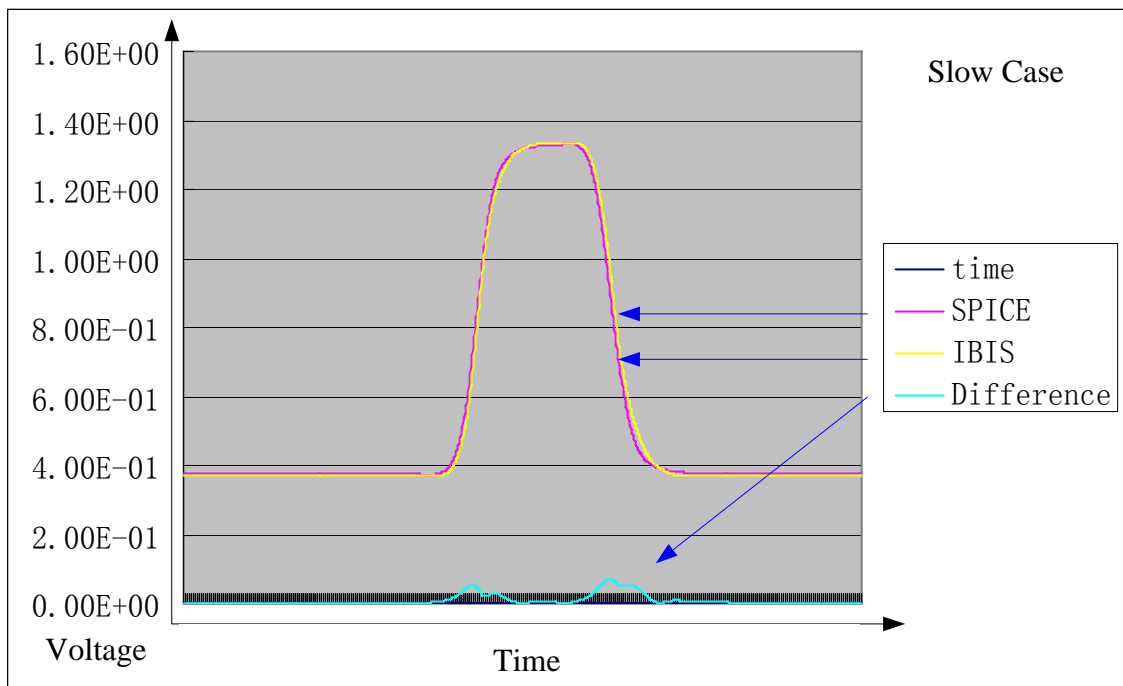


图 15 Slow Case 的差错示意图

表 6 IBIS 与 SPICE Slow Case 仿真结果比较

数据来源	上拉	下拉	阈值电压时间	
	Transition time	Transition time	上拉	下拉
SPICE 模型	0.42970ns	0.42500ns	2.65000ns	4.06250ns
IBIS 模型	0.42500ns	0.45000ns	2.65875ns	4.03125ns

表 7 IBIS 与 SPICE Slow Case 仿真结果比较 2

数据来源	峰值误差	峰值误差比	平均误差	平均误差比
IBIS 模型	0.07232v	7.530%	0.0097817v	1.019%

我们可以看到 SPICE 与 IBIS 模型的仿真精度差别在皮秒量级，最大的时刻误差是 Slow Case 下的下拉 transition time，误差有 0.025ns，最大的峰值误差是 Fast Case 中，误差达到 0.16181v。而在配置为 AMD Athlon 64 位 CPU 512M 内存的 PC 上运行时，跑一个 bit 的 SPICE 仿真花费了 45.63 秒，而跑 1 个 bit 的 IBIS 仿真花费了 1.37 秒。因此，对于需要测试海量数据的 DDR-SDRAM 而言，IBIS 模型是一个非常好的选择。

3 IBIS 驱动器瞬态响应仿真方法和改进

3.1 IBIS 仿真方法研究意义

IBIS 规范了一种输入输出缓冲器行为的描述方法，它统一了多种不同样的行为描述，然而如何根据 IBIS 记录的数据进行仿真，在 IBIS 规范中并没有定义。采用正确的方法，来构建出准确表征 IBIS 模型的电路，是 IBIS 仿真的关键所在。而对于不同的仿真器，它们对 IBIS 模型有着不同的解释，因此导致了即使对于相同的 IBIS 模型，在相同的激励和负载情况下，使用不同的电路仿真器也会有不同的仿真结果。比如上文中建立的 DDR-SDRAM 的 IBIS 模型，串联 75 欧姆电阻和 5pf 电容，分别使用 Synopsys 的 HSPICE 仿真器和 Mentor Graphic 的 PAD 仿真器对其仿真，电路分别如下图所示：

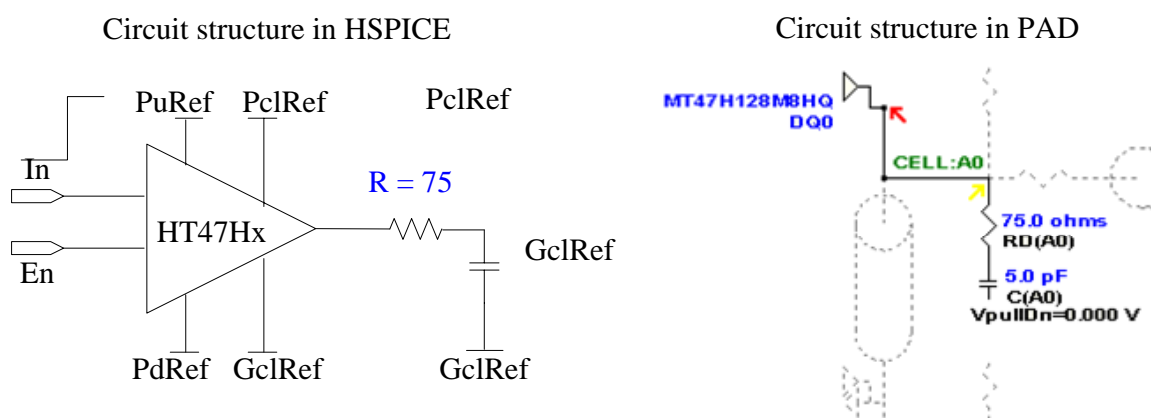


图 16 HSPICE 与 PAD 仿真电路

从仿真结果我们可以看到，即使简单的电阻负载，不同的仿真器的仿真波形都是不一致的，这与上面 SPICE 模型到 IBIS 模型之间的误差有所不同，前者是由于 IBIS 模型是对 SPICE 模型的简化，为了提高仿真速度而牺牲了一些精度，对于使用相同 IBIS 模型的不同仿真器之间的差别则是因为各个仿真器对 IBIS 模型的使用各

不相同导致的。

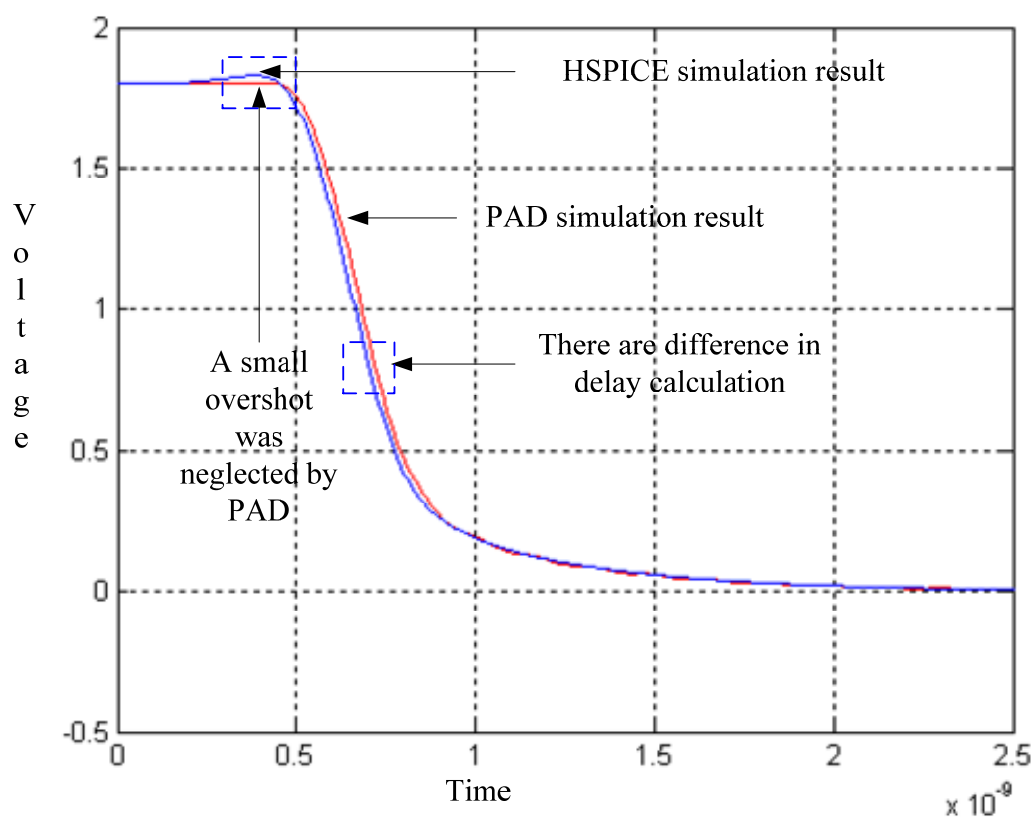


图 17 HSPICE 与 PAD 仿真结果比对

从 IBIS 标准执行，直到最近一次的 Tokyo 举行的 IBIS 峰会，关于 IBIS 模型在不同仿真器上会有不同的仿真结果这一问题始终是 IBIS Summit 的一个讨论主题。
(<http://www.vhdl.org/pub/ibis/summits/sep07b/wang.pdf>)

这是因为 IBIS 模型实质上是电路在某些特定负载情况下的仿真数据，数据不能直接被利用到不同仿真中，而需要重新组织成一个合理的，适用于当前仿真器的模型。而这个过程被称作基于该仿真器的 IBIS 仿真方法。

3.2 输入缓冲器仿真方法

IBIS 中定义了许多 IO 电路的描述方法，而在仿真中常常使用的是单端口非反向的输入缓冲器模型或称为接收器，和输出缓冲器模型或称为驱动器。在 IBIS 模型

中，主要记载了以下输入缓冲器的特征：负载电容(C load)，电源箝位二极管(Power clamp diode)以及地箝位二极管(Ground clamp diode)，翻转阈值(Vth)，这些特征的记录方法可以参考第二章的叙述。

因此，我们可以构建一个如下的子电路，使用一个接地的电容 C_load 来表示负载电容，我们不需要从记录的电压电流值去推导出两个箝位二极管的具体特征参数，可以用两个电压控制电流源 I_power_clamp，I_GND_clamp 来表征箝位二极管的行为，数值即是 IBIS 中[power clamp][GND clamp]两个数据表格的值。而翻转阈值可以用电压控制电压源 V_out 来表示，当节点 pin 的电压翻越阈值电压时，输出逻辑电平 Vcc 或 0。电路结构如下：

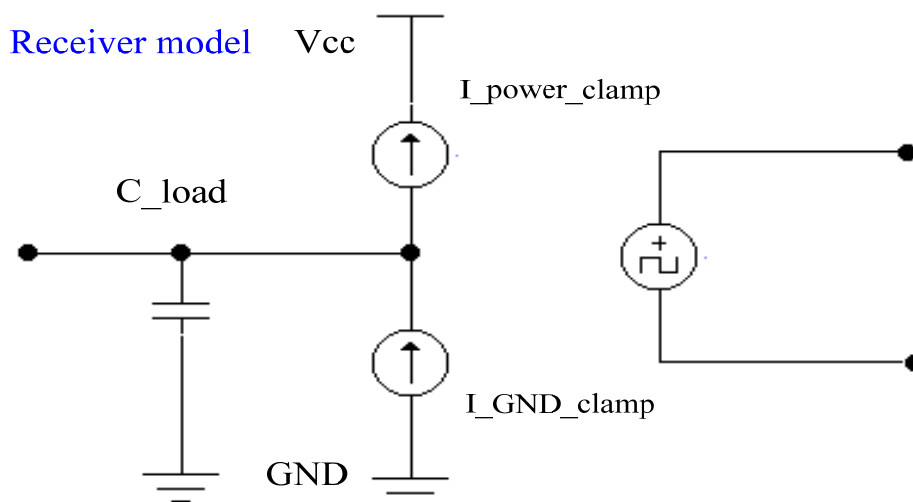


图 18 输入缓冲器仿真电路

将此接受器作为传输线终端，在 HSPICE 中进行仿真，仿真网表如下，通过改变传输线 U1 的长度，第一次仿真 L=0.2，第二次仿真 L=0.5。

```
X_IBIS_INPUT nd_pc nd_gc t_out nd_out_of_in
R_1 t_out 0 50
U1 nd_in 0 t_out 0 micro1 L=0.2
.model micro1 U LEVEL=3 PLEV=1 ELEV=1 wd=2m ht=2m th=0.25m
+ KD=5
```

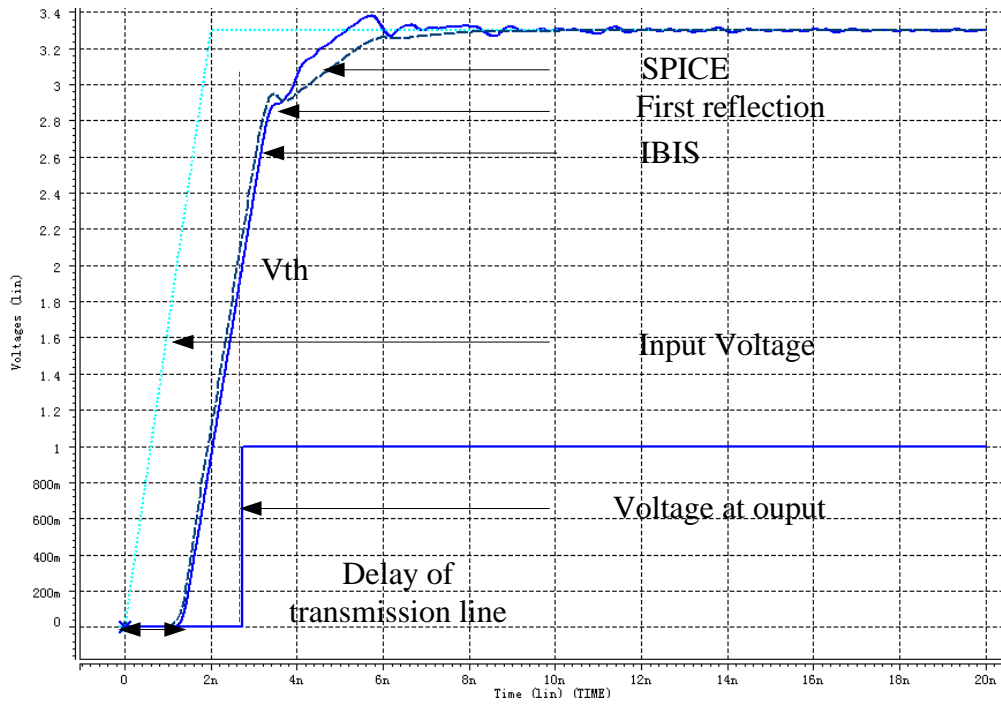


图 19 L=0.2 输入缓冲器仿真结果

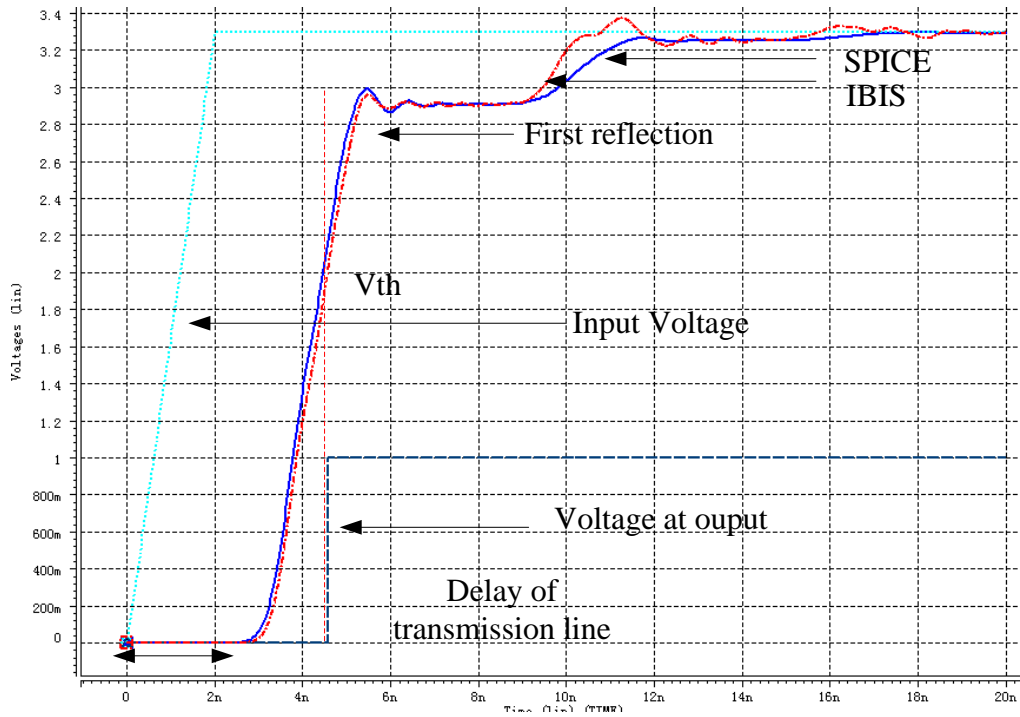


图 20 L=0.5 输入缓冲器仿真结果

SPICE 与 IBIS 模型仿真数据 transition time 和通过阈值电压的时间比较如下，我们可以观察到在两次仿真中，在第一次反射时，IBIS 输入缓冲器模型与 SPICE 模

型非常接近，但是在反射后，由于阻抗是在变化的，所以单一参数的 IBIS 模型便不能精确的描述电路行为了，这是由于 IBIS 本身记录的数据有限导致的。

表 8 传输线长度 L=0.5 仿真结果

模型	10% VCC	90% VCC	Vth
SPICE 模型	3.345ns	9.622ns	4.22ns
IBIS 模型	3.454ns	9.342ns	4.342ns

表 9 传输线长度 L=0.2 仿真结果

模型	10% VCC	90% VCC	Vth
SPICE 模型	1.456ns	4.041ns	2.623ns
IBIS 模型	1.571ns	3.867ns	2.492ns

3.3 输出缓冲器仿真方法

对于输出缓冲器，在稳态工作情况下，通常用恒定的电阻来表示，而更多的我们关注输出缓冲器在不同的负载和激励情况下的瞬态行为。

在 IBIS 规范的输出端口模型(output model)中,包含有以下的数据表格，上拉电流电压表格(pull up I-V table)，它记录了在输入为逻辑高电平时，输出端点的电流随输出端点电压的变化过程。下拉电流电压表格(pull down I-V table)，它记录了在输入为逻辑低电平时，输出端点的电流随输出端点的电压变化的过程。上拉波形电压时间表格(rising waveform V-T table)记录的是，一个从低电平到高电平翻转的激励下，输出端口电压随时间变化的过程。下拉波形电压时间表格(falling waveform V-T table)记录的是，一个从高电平到低电平翻转的激励下，输出端口电压随时间变化的过程。

我们用上拉和下拉器件来表征驱动器的上拉和下拉行为,用一个接地电容来表征该模型的输出负载。

IBIS 的输出缓冲器模型中，基于特定的负载情况(通常是 $R_{fix} = 50$ 欧姆)记录了一组瞬态响应的数据，负载不同时，这些数据是不能直接用于其它负载的仿真，需要结合稳态响应的数据对其重新计算。这个求解电路瞬态响应的方法通常称作 IBIS 瞬态响应仿真方法。

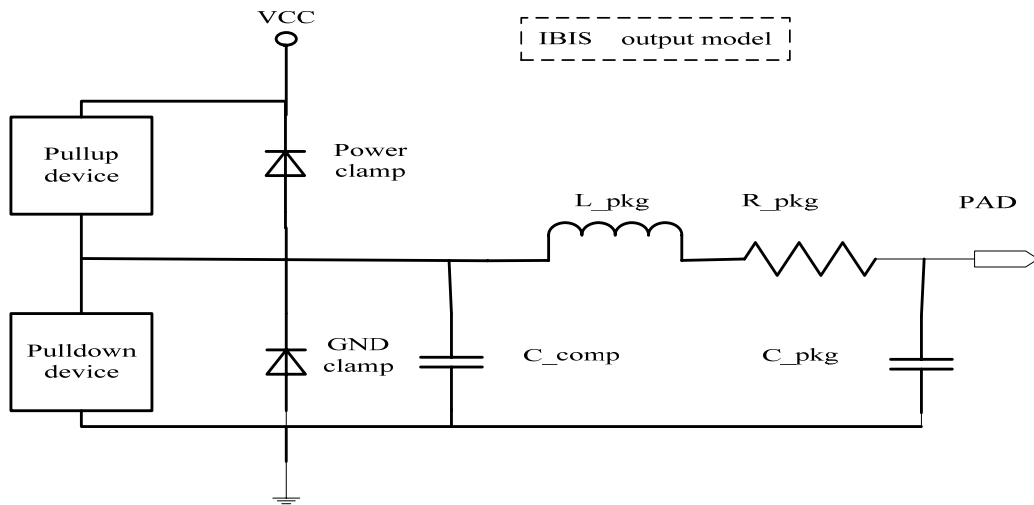


图 21 IBIS 输出缓冲器仿真模型

3.4 输出缓冲器瞬态响应仿真方法

在第三章中，我们从 IBIS 输出缓冲器模型数据中计算出当缓冲器处于正常工作电压阈值之内的平均特征阻抗，用一个恒定的电阻就可以表示这个输出缓冲器在正常工作电压区域下的瞬态仿真模型，但是这种粗略的电阻模型不能够发挥出 IBIS 模型的精度，事实上输出缓冲器的阻抗是根据激励段的电压实时变化的，对于信号完整性分析我们需要知道电路的瞬态响应。

参考文献[1]中提出了一种 IBIS 输出模型的瞬态响应仿真方法。它假设在瞬态响应的每个时刻，输出电流是相同电压下稳态电流的一部分，且不随负载的变化而变化，由于其使用了 IBIS 输出端口的两组数据，一条是 Rising Waveform 另一条是 Falling Waveform，通常把它称为二波形方法。

● IBIS 2 waveform 方法

IBIS 输出端口模型中所记录的电流电压表格，记录了在输出端为电压 V 时，流出输出端的电流 I 。当一个逻辑低电平到逻辑高电平的翻转发生时，上拉器件逐渐的开始工作，同时，下拉器件从完全打开状态逐渐开始关断。

把上拉器件和下拉器件的行为结合起来，可以看出从低电平到高电平翻转瞬态响应是从上拉稳态曲线到下拉稳态曲线的变化过程。

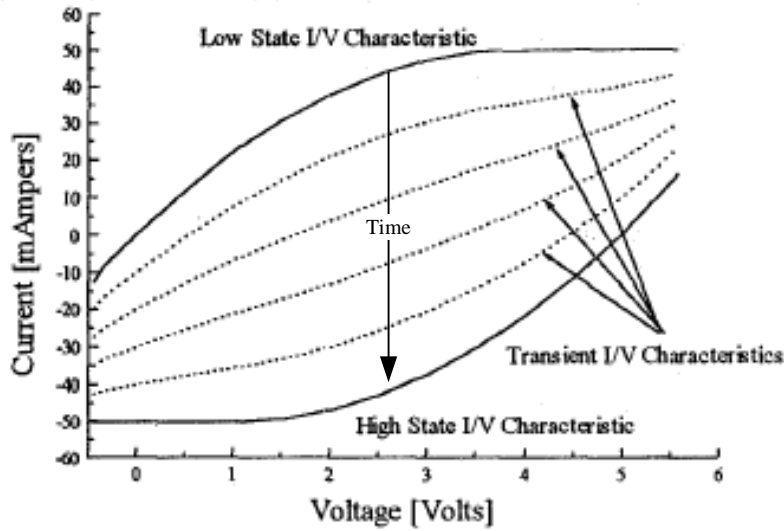


图 22 0 到 1 翻转电流过渡曲线 (引自参考文献[1])

而此时这个驱动器的瞬态响应可以用上拉器件下拉器件的瞬态响应曲线以及它的负载线的交点得到。

对于不同的负载的瞬态仿真，我们假设在相同的时间点上，上拉器件和下拉器件的电流于相同电压下稳态电流的比值是恒定的，即满足以下关系。

$$I_{pull_up_transient}(t) = \alpha(t) \cdot I_{Pull_up_steady}(V_{output}(t)) \dots \dots \dots (3.1)$$

$$I_{pull_down_transient}(t) = \beta(t) \cdot I_{Pull_down_steady}(V_{output}(t)) \dots \dots \dots (3.2)$$

因此从电路的角度来看，上拉器件和下拉器件对外负载特性表现为一个随时间变化的电阻，在同一时间点上，上拉和下拉器件的输出电流随输出端点的电压线性变化。而这样的电路特性，我们可以用一个由时间参数调制的电流源表示。从而我们可以假定 IBIS 的输出端口电路模型如下图所示，而上拉和下拉电流源的行为分别满足公式(1)(2)。这样的电路模型便可以使用常见的电路仿真器进行瞬态响应和稳态响应的仿真。

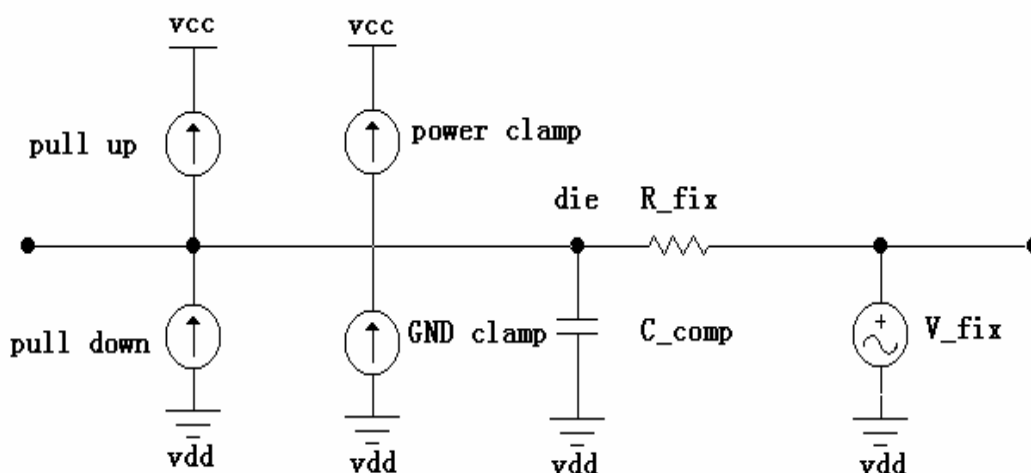


图 23 IBIS 输出电路模型

如何得到调制上拉电流源和下拉电流源的系数，便是整个 IBIS 输出端口电路模型的关键。

在 IBIS 的 1.0 版本中，对于驱动器模型的瞬态响应，只是记录了一组当负载电阻为 50 欧姆的情况下从低电平到高电平与从高电平到低电平的翻转发生时，输出电流随输出端电压变化的过程。

这样，我们只有一条负载曲线，而上拉下拉调制参数有两个，所以需要找到上拉下拉参数的一个近似的关系。

对于从低电平到高电平的翻转，在零时刻，上拉器件关断，下拉器件完全开启，在达到稳态后，上拉器件完全开启，下拉器件关断，并且在翻转过程中，上拉和下拉器件的瞬态电流不会大于同样电压下的稳态电流，这样我们假设：

$$\alpha(t) + \beta(t) = 1 \dots \dots \dots (3.3)$$

从而对于已知的随时间变化的输出端电压 V 和负载，以及稳态时相同输出端电压时的电流 $I_{pull_up_steady}(V)$ 和 $I_{pull_down_steady}(V)$ ，我们可以解出这个电流调制系数。

$$\alpha(t)g_{pull_up_steady}(V_out(t)) + \beta(t)g_{pull_down_steady}(V_out(t)) = I(V_out(t)) \dots \dots \dots (3.4)$$

但是由于公式(3)对上拉下拉器件瞬态电流调制参数的估计是非常的粗糙的，致使这种 two waveform 方法构成的电路在瞬态仿真时精度有所不足，并且在 IBIS 的

新版本中，为了更加精确的描述瞬态响应，IBIS 的输出端口模型记录了两个不同负载情况下的电压随时间变化的表格，对应于输出端从低电平到高电平和从高电平到低电平的两种不同翻转方式，一共有 4 个数据表格，根据这四个数据表格所提取瞬态电流调制系数的方法叫做四波形方法[2]。

● IBIS 4 waveform 方法

在业界最为广泛使用的 IBIS 3.2 版本中，输出端口模型的瞬态响应通过两组负载来描述。两组负载分别对应上拉下拉器件，有以下四种组合：

负载情况 1:

上拉曲线 Rising waveform:

$$R_{fix}=50\Omega \quad V_{fix}= Gnd$$

下拉曲线 Falling waveform:

$$R_{fix}=50\Omega \quad V_{fix}= Gnd$$

负载情况 2:

上拉曲线 Rising waveform:

$$R_{fix}=50\Omega \quad V_{fix}= VCC$$

下拉曲线 Falling waveform:

$$R_{fix}=50\Omega \quad V_{fix}= VCC$$

这样我们可以细化瞬态电流调制参数，对于每一组的负载情况，都有一组上拉下拉瞬态电流调制参数对应。而对应于 V_{fix} 是地的情况，从低到高翻转时，上拉器件开启，下拉器件关断，分别对应调制参数 K_{up_on} ，与 K_{dn_off} ，从 1 到 0 翻转时，上拉器件关断，下拉器件开启，分别对应调制参数 K_{up_off} ，与 K_{dn_on} 。而对应于 V_{fix} 是 VCC 的情况，亦有相同的论述，应此对于每一组负载列写下列方程，两组负载情况共有四个方程。

$$K_{up_on} * I_{up}(Vout_rise(t)) + K_{dn_off} * I_{dn}(Vout_rise(t)) = Vt_rise(t) / R_{fix} \dots\dots\dots(3.5)$$

$$K_{up_off} * I_{up}(Vout_fall(t)) + K_{dn_on} * I_{dn}(Vout_fall(t)) = Vt_fall(t) / R_{fix} \dots\dots\dots(3.6)$$

$$K_{up_on} * I_{up}(Vout_rise(t)) + K_{dn_off} * I_{dn}(Vout_rise(t)) = [Vt_rise(t) - VCC] / R_{fix} \dots\dots\dots(3.7)$$

$$K_{up_off} * I_{up}(Vout_fall(t)) + K_{dn_on} * I_{dn}(Vout_fall(t)) = [Vt_fall(t) - VCC] / R_{fix} \dots\dots\dots(3.8)$$

其中等式左边 $Vt_rise(t)$ $Vt_fall(t)$ 是输出端点处加在负载电阻上的电压值，这

个电压与负载电阻的比值即是流出输出端点的电流。

此电流为上拉器件和下拉器件的瞬态电流之和，而根据上文所述，上拉器件和下拉器件的瞬态电流等于当前输出电压下的稳态电流 $I(V_{out}(t))$ ，与调制系数 K 的乘积。从而我们根据方程(3.5)(3.6)(3.7)(3.8)求出四个调制参数，根据不同的起始激励情况，触发相应的调制参数工作，即去乘上拉下拉稳态曲线，得到最终的仿真结果。:

3.5 输出缓冲器瞬态响应仿真方法实现和改进

基于以上对 IBIS 输出模型的瞬态响应仿真方法的描述，我们编写程序处理 IBIS 文件，求解出用于仿真的数据，并在电路仿真器上实现仿真。

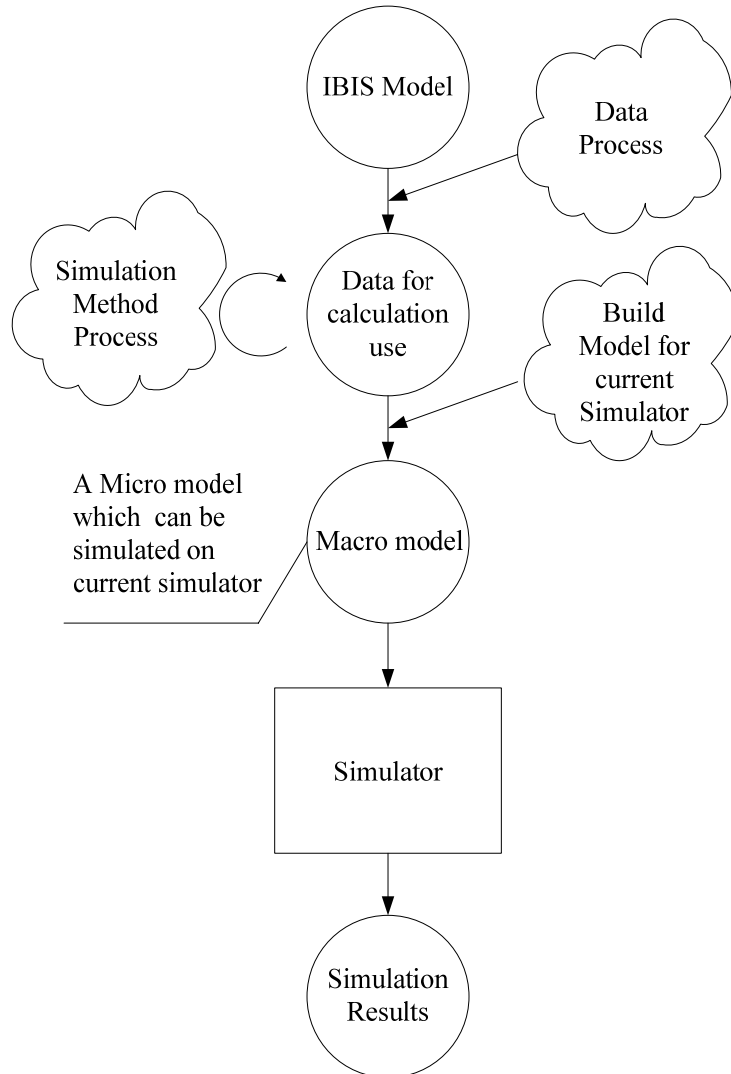


图 24 IBIS 仿真方法流程

上图描述了一个 IBIS 仿真方法的环境和流程，在这里我们以 Synopsys 公司的 HSPICE 作为电路仿真器(simulator)，选用 IBIS 官方提供的标准输出模型(http://www.vhdl.org/pub/ibis/training/IBIS_class_2003.zip), 在对 IBIS 模型进行规整和计算后，使用得到的数据，结合上面提到的 IBIS 模型的仿真结构，构建出 HSPICE 能够识别的宏电路(macro circuit)，并且连接传输线和负载，得到仿真结果。

下面我们分别描述主要的三个步骤。

3.5.1 IBIS 数据预处理

IBIS 预处理读取 IBIS 文件，根据命令行输入参数，将 IBIS 文件中所要处理的数据提取出来，并作整理。

IBIS 文件记录数据使用了大量的数据表格，因此需要根据其文件格式将数据提取出来，并在一个目录结构中进行存储，以便下面的算法调用，这个过程称作 IBIS 数据的预处理。这一步，我们基于 TCL(tool command language)语言来编写程序，因为 TCL 是一种跨平台的脚本语言，它对正则表达式有着很好的支持，从而有着强大并且方便的文本处理能力。

IBIS 数据预处理的实现细节如下：

首先根据命令行的输入，确定待处理的 IBIS 模型的端口类型，比如是输入端口(input)型或者是输出端口(output)型，不同的端口类型记录数据的方式不同，在数据预处理程序内部也会区别的对对待。

主要的输入参数是，文件路径，文件名，电路端口名，电路端口模型名。在抽取的时候，有些非强制要求的关键字和数据表格可能没有实现，则这些值需要以 0 代替。

下一步是建立合适的目录结构以便存放数据，根目录为 `extract_data`，第二层目录为当前处理的 IBIS 文件名，第三层目录是端口名，第四层目录是这个输出端口所包含的数据，根据 IBIS 3.2 的标准，一个输出端口模型必须包括伴随电容，上拉下拉瞬态稳态响应曲线，封装参数，参考电压这些值，我们把从 IBIS 中提取出的这些值分别存储于对应的目录下以供调用。

第三步是对数据进行规整，比如 IBIS 使用带单位的数据来描述器件参数的值，比如电容 5pf 电流 10ma，而在计算时，不同的计算环境对数据的表达有不同的要求，因此这一步根据具体的要求将 IBIS 中记录的数据转化成科学计数法或是有位数限制的小数表示。

3.5.2 IBIS 算法处理

IBIS 算法处理程序读取 IBIS 数据预处理的结果，根据上文所介绍的 IBIS 四波形算法计算瞬态响应电流调制参数。

这一步涉及到查找表中的插值运算等数值运算问题，通常可以选择使用 C, matlab 实现，我们在使用 matlab 验证算法的正确性的同时，把算法移植到 TCL 上面，以保证整个处理环境的一致性。

瞬态响应四波形提取算法以在上文进行描述，本算法即使要求解方程 (5)(6)(7)(8)。主要有两个问题，一是基于离散数据的方程计算，另外就是对于计算流过电容的电流时所需要的电压微分运算的差分近似。

IBIS 模型通过数据表格的方式记录数据，对于记录瞬态响应的四个电压时间表格(VT table)，我们首先对其索引(index)进行统一化，即把这四个电压时间表格的时间约束为相同的一个时间索引。为确保数值的准确性，我们将每一个时间点都纳入到这个索引中，计算方法如下图所示。

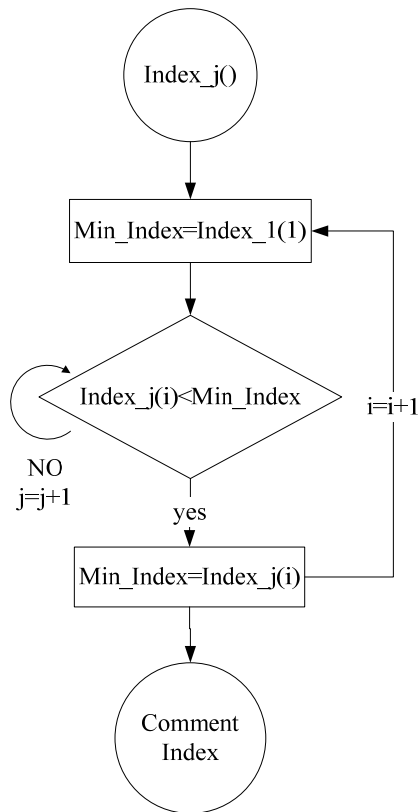


图 25 公共索引计算

在获得公共索引之后，我们更新使用这个索引的查找表，显然索引中包含了很多原本没有的数据点，因此需要对这个数据点进行插值，我们选择二阶均值的牛顿插值对新索引下的缺失点进行插值。

$$x < a < b$$

$$f(x) = f(a) + \frac{f(a) - f(b)}{a - b}(x - a) + \frac{1}{a - x} \cdot \left(\frac{f(a) - f(b)}{a - b} - \frac{f(a) - f(x)}{b - x} \right) \cdot (x - a)(x - b) \dots (9)$$

在以上两个步骤之后，我们就获得了相同查找索引的查找表，因此可以对于每个索引点求解方程(3.5)(3.6)(3.7)(3.8)。

因此，计算每个时间点 t 的瞬态电流调制系数的过程是：

在查找表中找到该时间点下的输出端口电压

查找该电压下稳态的上拉电流

查找该电压下稳态的下拉电流

带入方程求解

对于四组瞬态响应曲线都进行以上步骤，把这些值代入方程(3.5)(3.6)(3.7)(3.8)便可分别求出这个时间点上的四个瞬态电流调制系数。

算法结束后，将公共时间索引和求得的四个瞬态电流调制系数保存为一个新的查找表，以便供下面的程序调用。

3.5.3 宏电路网表实现

计算完以上所需的数据之后，需要把这些数据组织成为网表格式，根据上述的 IBIS 输出端口模型中的连接关系，使用 HSPICE 能够支持的元器件来组成宏电路，从而完成了将一个 IBIS 输出端口数据解释成一个可供 HSPICE 调用的电路的过程。电路结构如图所示：

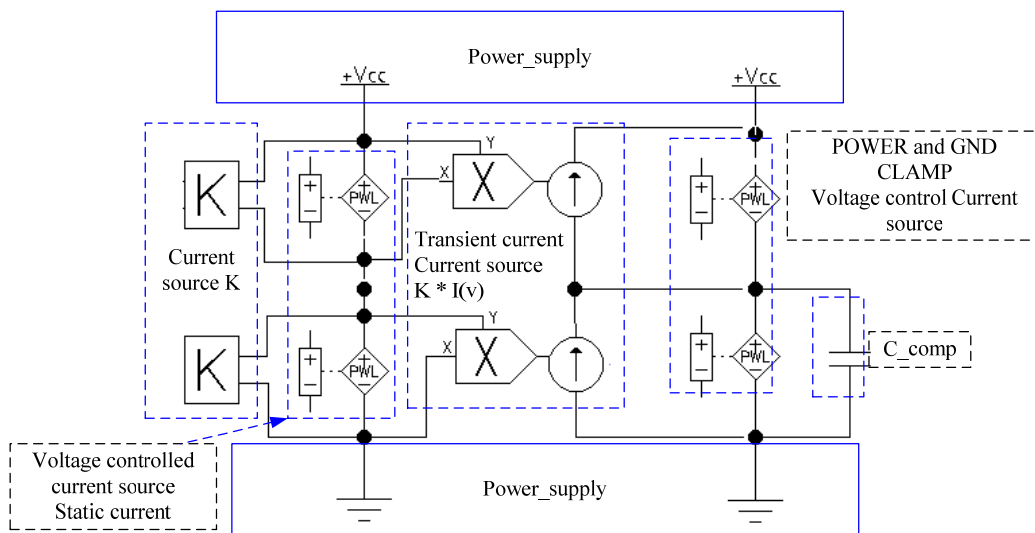


图 26 仿真电路实现

根据上面定义的输出端口仿真模型，我们使用压控电流源（voltage controlled current source）来表示上拉稳态电流和下拉稳态电流，其中电流大小跟随输出端点电压变化，而调制参数 K 可以用随时间变化的电流源表示，然后按照公式 (3.5)(3.6)(3.7)(3.8) 将相应的 K 值与稳态电流值相乘，就得到了当前时间点的瞬态电流输出，而电源箝位和地箝位亦可以用电压控制电流源来表示，伴随电容是一个常数，因此直接使用电容表示，综上所述网表文件如下：

```
* IBIS simulation
.INCLUDE out33v.lib
V_ref_up Vcc 0 3.3v
V_ref_dn GND 0 0v
Xkpuoff 10 0 K_pu_off
Vkpuoff 10 0 0
Xkpdon 20 0 K_pd_on
Vkpdon 20 0 0
Xkpuon 11 0 K_pu_on
Vkpuon 11 0 0
Xkpdoff 21 0 K_pd_off
Vkpdoff 21 0 0
Xhigh out Vcc tempout_1 pullup
Vup tempout_1 Vcc 0
Gup out Vcc CUR='-I(Vup)*(I(Vkpuoff)*(1-V(in)) + I(Vkpuon)*V(in))'
Xlow out GND tempout_2 pulldown
Vdn tempout_2 GND 0
Gdn out GND CUR='-I(Vdn)*(I(Vkpdon)*(1-V(in))+ I(Vkpdoff)*V(in))'
C_comp out GND 4.47pf
```

至此，我们已经将一个输出缓冲器的 IBIS 模型转化为一个由子电路构成的 SPICE 模型，可以对其进行瞬态仿真。我们把输出缓冲器接上一个 50 欧姆的电阻，仿真结果如下，可以看到使用以上方法作出的 IBIS 输出缓冲器瞬态响应曲线在 transition time 上有 0.28ns 的延迟。

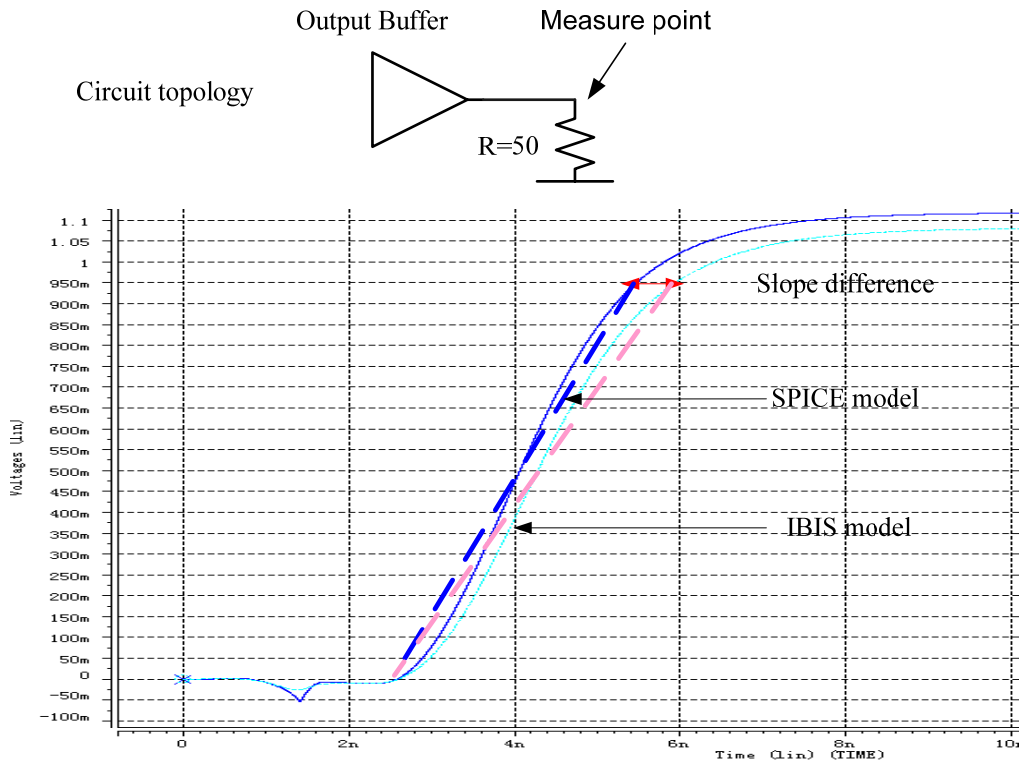


图 27 四波形方法仿真结果

3.5.4 算法改进与结果比较

注意到 IBIS 中 C_{comp} 的生成方法，它是在当前负载下，流过负载的电流除以输出瞬态响应曲线的平均斜率。 C_{comp} 主要影响两个方面，一方面是输出端的阻抗匹配，另外一个方面，它调制着输出波形的斜率。

在 IBIS 中生成瞬态响应曲线时，输出缓冲器连接上负载，做瞬态响应，并测量负载处的电压随时间变化的值，显然， C_{comp} 的作用此时被包括在了输出缓冲器内部，即在下图中的 measure point。从而 Rising waveform 和 Falling waveform 是包含了流过 C_{comp} 电流的结果。

在仿真实现时， C_{comp} 被作为负载又一次的被加入了电路，即在下图中的 simulation point，这就两次引入了 C_{comp} 的影响。

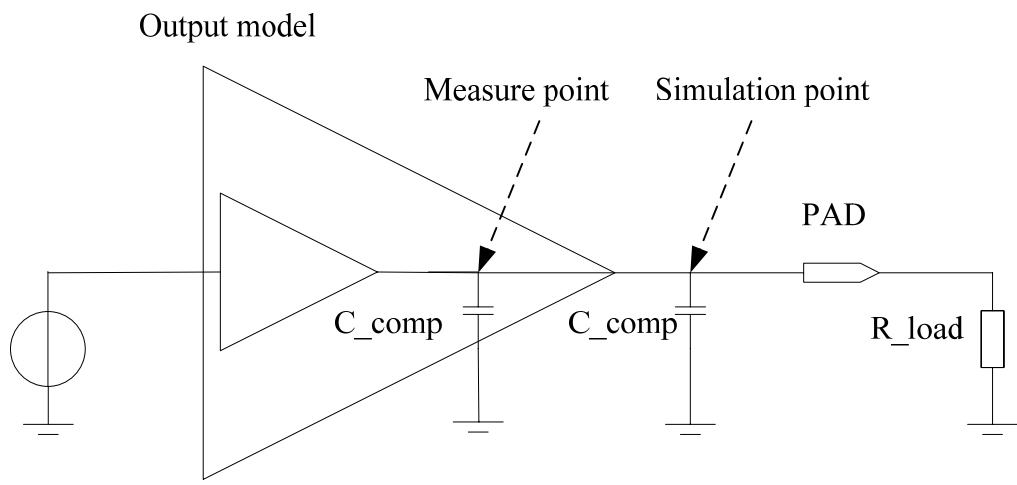


图 28 消除重复电容计算的仿真电路结构

C_comp 作为输出负载的作用必须被保留，但是我们必须去除被重复计算的 C_comp 带来的影响，对于上图，我们需要去除 measure point 处的 c_comp，而保留在 simulation point 处的 c_comp 带来的影响。因此我们将公式修改如下：

$$K_up_on * I_up(Vout_rise(t)) + K_dn_off * I_dn(Vout_rise(t)) = Vt_rise(t) / R_fix - C_comp(dv / dt) \dots\dots\dots(3.10)$$

$$K_up_off * I_up(Vout_fall(t)) + K_dn_on * I_dn(Vout_fall(t)) = Vt_fall(t) / R_fix - C_comp(dv / dt) \dots\dots\dots(3.11)$$

$$K_up_on * I_up(Vout_rise(t)) + K_dn_off * I_dn(Vout_rise(t)) = [Vt_rise(t) - VCC] / R_fix - C_comp(dv / dt) \dots\dots\dots(3.12)$$

$$K_up_off * I_up(Vout_fall(t)) + K_dn_on * I_dn(Vout_fall(t)) = [Vt_fall(t) - VCC] / R_fix - C_comp(dv / dt) \dots\dots\dots(3.13)$$

注意到计算流过电容的电流时，会需要用到电压的微分，对于离散的点而言，我们用相邻时间点上差分来代替电压的微分，即在 t 时刻的电压微分表示为：

$$t_0 < t < t_1$$

$$dv(t) = \frac{[(v(t) - v(t_0)) / (t - t_0) + (v(t) - v(t_1)) / (t - t_1)]}{2} \dots\dots\dots(3.14)$$

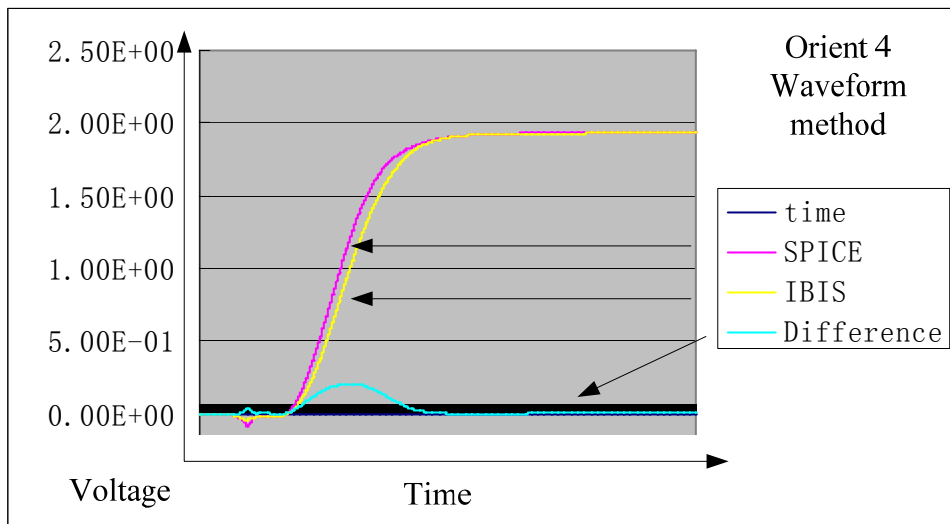


图 29 原始四波形方法仿真结果比较

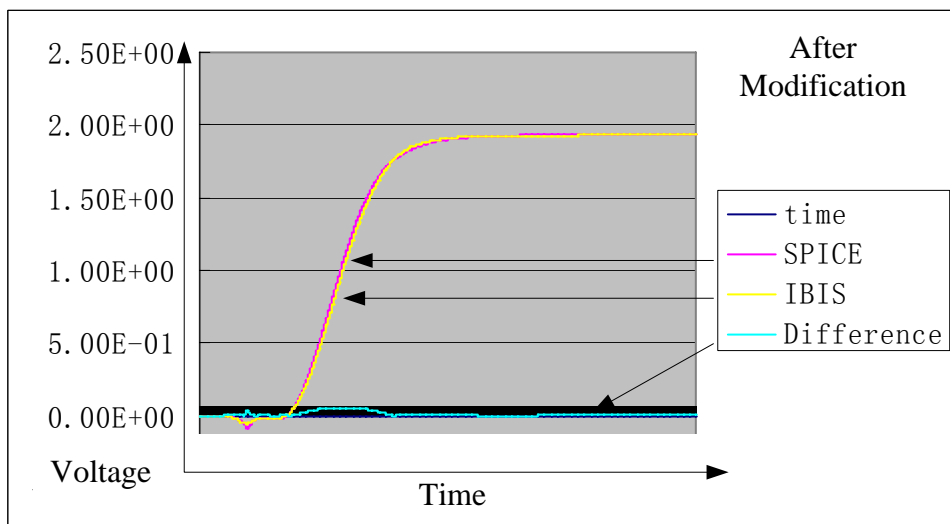


图 30 消除电容重复计算后仿真结果比较

表 10 四波形方法和改进后的仿真结果比较 1

数据来源	峰值误差	峰值误差比	平均误差	平均误差比
4 波形算法	0.20721	10.279%	0.040354	2.002%
改进 4 波形算法	0.055285v	2.743%	0.0116v	0.575%

表 11 四波形方法和改进后的仿真结果比较 2

数据来源	10% Vcc	90% Vcc	阈值电压点
SPICE	2.948ns	6.113ns	4.129ns
4 波形算法	3.3007ns	6.408ns	4.401ns
改进 4 波形算法	2.976ns	6.052ns	4.199v

如上述表格和仿真图所示，使用上一节的仿真环境对改进的方法进行仿真，驱动器的负载为 50 欧姆。改进前的上拉穿过阈值点的误差为 0.272ns，改进后的上拉 transition time 误差为 SPICE 仿真的 transition time 为 0.03ns。改进后的方法消除了重复计算伴随电容带来的误差。

4 IBIS 驱动器非理想输入仿真方法和改进

IBIS 仿真模型通常还存在以下的问题：

- 参数的使用

对于 IBIS 记录的数据，组织成一个合理的行为级模型，是正确使用 IBIS 模型的基础。但是由于没有一个标准的建立行为级模型的方法，所以在一些参数的使用上仍然存在着争议，比如对于伴随电容的使用等等，在对同步开关噪声，和功耗分析时均有不同的应用。

- 描述新的 IO 电路的特性

比如预加重 IO，它可以根据输入电平不同的持续时间，提供不同的电平输出，这样用简单的上拉下拉行为已经不能正确表征其行为了。不过，这个问题不仅仅是改变 IBIS 仿真电路模型能够解决，还需要新的 IBIS 版本，对此类 IO 的描述进行重新定义，从而记录更多的数据，再重新组织该 IO 的 IBIS 仿真电路模型。

- 缺乏输入激励影响输出的数据

IBIS 所记录的瞬态响应，它们的输入都是在 0 时刻翻转的理想阶越信号，然后记录下从 0 时刻到某一时刻 T 之间的输出电压。这就引起了两个问题，一个是非理想输入对输出的影响，另一个是当输入激励电平的维持时间小于 T 时，输出的失真。这类问题被统称为 over clocking。

IBIS 输出端口瞬态响应记录的是在零时刻翻转的理想阶越激励下的输出电压，因此许多 EDA 工具在实现 IBIS 仿真时，就没有考虑除理想阶越外其它任何可能输入，以 Mentor Graphic 的板级仿真工具 Hyperlinks 为例，它只是支持 Rising 和 Falling 这两种激励输入。

通常逻辑信号经由传输线传输，在输入端会造成一定的输入延迟和翻转延迟 (transition time)。所以简单的无翻转时间理想激励不能满足这种仿真的要求。下面我们就这种输入激励影响输出的问题，提出了一种新的仿真方法，它能够准确的对非理想输入做出响应。

4.1 仿真环境及方法

我们选择 HSPICE 作为仿真平台，在第 3.4 节的 IBIS 输出端口仿真模型的基础上，对电路结构作出改变，使其能够支持带翻转延迟的输入激励。并且，我们以原始 SPICE 电路模型为标准，比较 B element 的仿真结果和宏电路改进后的仿真结果，以说明改进效果。

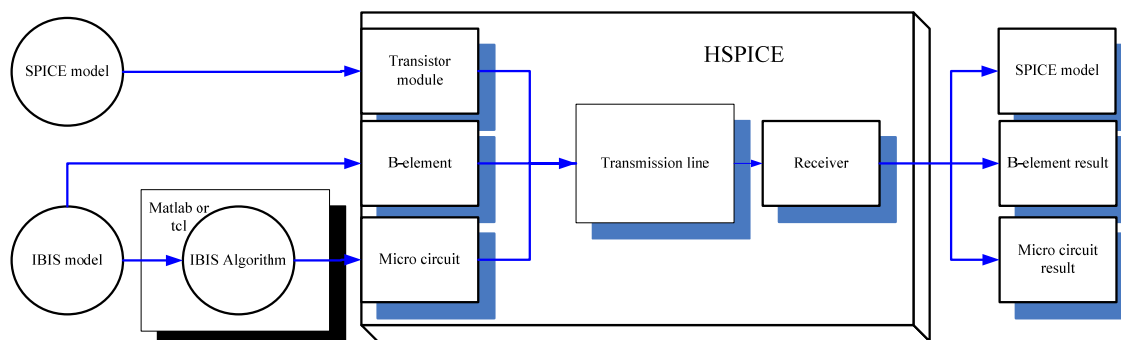


图 31 IBIS 改进模型仿真环境

4.2 B element 仿真

B element 是 HSPICE 用来支持 IBIS 仿真所提供的宏电路，它对 IBIS 的数据和端口类型进行封装打包和解释，对外表现为一个只含有输入输出接口和电源接口的黑盒。它的内部实现是不可见的，当我们使用 B element 的时候，我们既是默认它对 IBIS 模型的解释是准确的，而通过仿真，我们发现至少在非理想输入激励这一问题上，B-element 并没有能很好的反映出 IBIS 模型描述的电路的行为。

B-element 模型的网表示例如下

```
B_INPUT nd_pc nd_gc u_out nd_out_of_in
+ file = 'lab_1.ibs'
+ model = 'in33v'
+ typ = 'typ'
+ power = 'on'
+ buffer = 'input'
```

在电路网表中加入上述申明，便是实例化了一个 IBIS 描述的接收端口电路，此电路的数据是来源于 IBIS 文件 lab_1.ibs，这个电路的模型名称叫做 in33v，而仿真时，在最小(min)，最大(max)，典型(typ)的不同数据中选择典型(typ)数据。Power =

on 是指本接收端口采用内部电源供电，即按照 IBIS 文件 lab_1.ibs 中所记录的电源数据提供电源，不需要在网表里外接电源。此外本接收器的输出只能是非 0 既 1 的逻辑值，直接由电压源提供，输出电阻为无穷大。

B-element 在非理想输入下的仿真电路结构如下，输出端口接一根传输线，模型如下，接收端为容性负载。

```
.subckt Load pad
.PARAM vref='vref_typ'
T1load pad 0 end 0 Z0=50 TD='39.37/6*1e-9' L='6*ui*4*1e+9/39.37'
R1load end vref R=Rload
C1load end 0 C=Cload
Vref vref 0 DC=vref
.ends
```

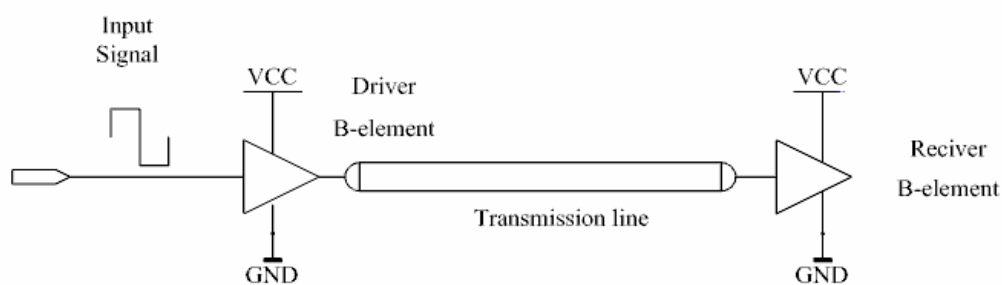


图 32 B element 仿真电路

我们测试 0 时刻从 0 到 3.3V 翻转的理想阶越信号和翻转时间为 2 ns 的斜率信号，分别观察进入传输线的波形和到达接收端的波形。仿真结果和分析如以下图表所示：

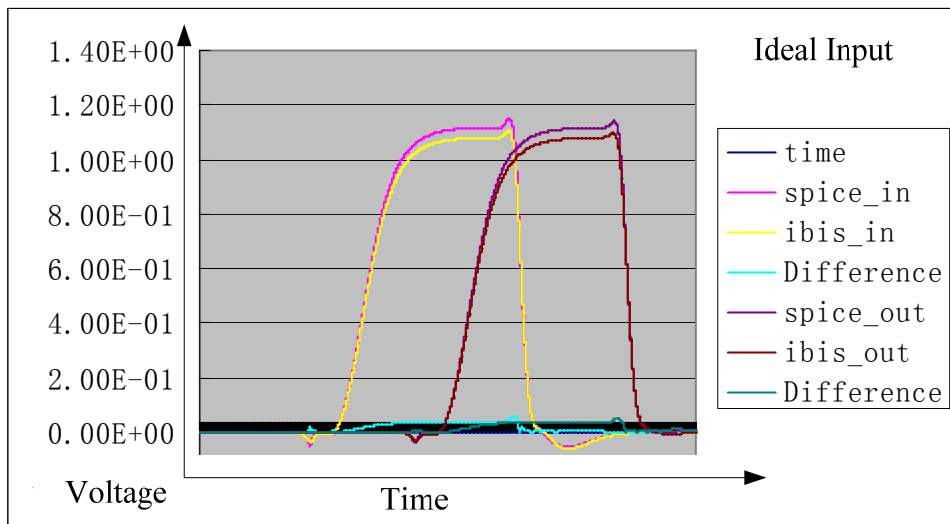


图 33 B-element 理想输入仿真波形

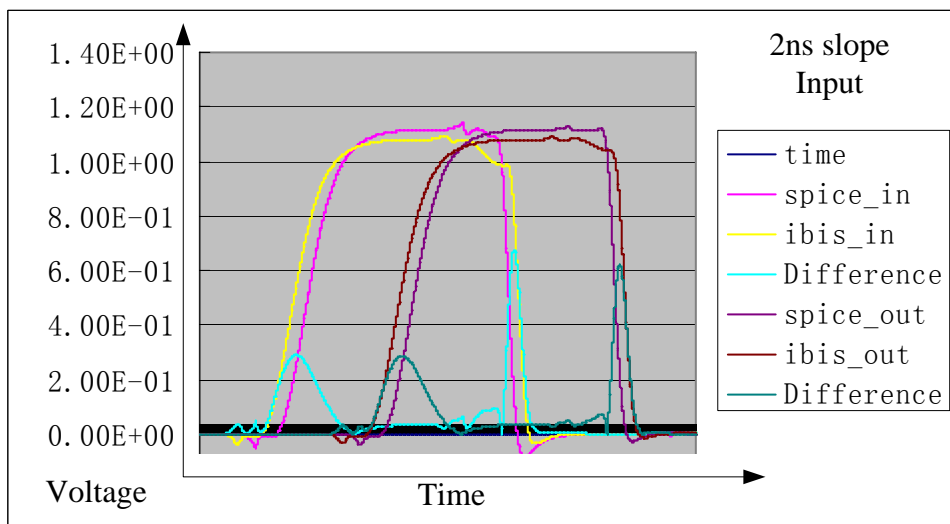


图 34 B-element 2ns 斜坡输入仿真波形

表 12 B-element 斜坡输入仿真误差分析

数据来源	峰值误差	峰值误差比	平均误差	平均误差比
理想输入	0.056235v	4.669%	0.019443v	1.614%
2ns 斜坡输入	0.67520v	55.123%	0.070182v	5.730%

以理想输入情况下的误差分析作为 IBIS 模型质量的标准，即是，我们认为理想输入下 IBIS 仿真的结果是该 IBIS 模型能够达到的最大精度。

由于 overclocking 主要是延迟时间上面的计算误差，所以斜坡输入下同一时间点的电压值相差很大，最大可达 0.6v，这种由于斜坡输入带来的误差必须作出纠正。

4.3 改进方法和仿真结果

我们以第 3 章所实现的 IBIS 输出缓冲器模型为基础，对其进行修改，使其能够支持带斜率输入的激励。

我们有四个瞬态电流调制参数在每个时刻点修正着上拉器件和下拉器件的瞬态电流，而理想输入是在 0 时刻即对四个参数进行分配，如果输入是一个上拉的激励，那么就是用 K_{dn_off} 与 K_{up_on} 这两个参数来调制电流。

那么对于带斜率的输入信号，输出产生“提前”的原因，是因为对电流调制参数的起始作用时间做出了误判断。因此我们对此结构做出修改，加入一个控制 K 的开关，当输入激励到达某个阈值时，调制参数开始工作。而在到达阈值电压之前，如果输入电压低于阈值电压，则上拉器件处于关断，下拉器件处于稳态，如果输入电压高于阈值电压，则下拉器件处于稳态，上拉器件处于关断。

基于以上想法，我们构造出一个函数来控制调制参数 $K_trigger$ ，在 HSPICE 仿真时的每一个时间节点，都由这个参数来判断和修正着拉和下拉器件的输出。 $K_trigger$ 的取值为 0 或 1，当输入向上翻越过阈值电压 V_{th} 后， $K_trigger$ 触发 K_{dn_off} 与 K_{up_on} 去调制下拉上拉电流源，当输出向下翻越过阈值电压 V_{th} 时， $K_trigger$ 触发 K_{dn_on} 与 K_{up_off} 来调制上拉下拉电流源，函数表达如下所示：

for rising

when $V < V_{th}$

$$K_{up} = 1 \quad K_{dn} = 0 \dots \dots \dots (4.1)$$

when $V > V_{th}$

$$K_{up} = (1 - K_{trigger})gK_{up_off} + K_{trigger}gK_{up_on} \dots \dots \dots (4.2)$$

$$K_{dn} = K_{trigger}gK_{dn_off} + (1 - K_{trigger})gK_{dn_on} \dots \dots \dots (4.3)$$

for falling

when $V < V_{th}$

$$K_{dn} = 0 \quad K_{up} = 1 \dots \dots \dots (4.4)$$

when $V > V_{th}$

$$K_{up} = (1 - K_{trigger})gK_{up_off} + K_{trigger}gK_{up_on} \dots \dots \dots (4.5)$$

$$K_{dn} = K_{trigger}gK_{dn_off} + (1 - K_{trigger})gK_{dn_on} \dots \dots \dots (4.6)$$

基于以上函数表示，在第三章介绍的驱动器仿真电路模型中加入这样的行为开关，电路结构如下图所示，在输出端接传输线，并在负载端接相当于 5pf 电容的接收器：

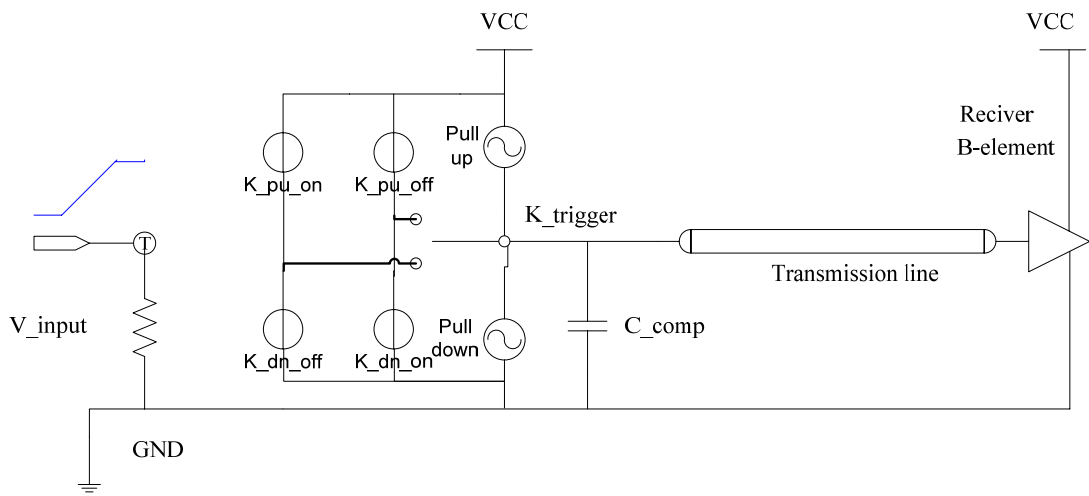


图 35 斜坡输入改进电路模型

在此结构上依次对 2 纳秒斜坡输入做出仿真，仿真波形如下：

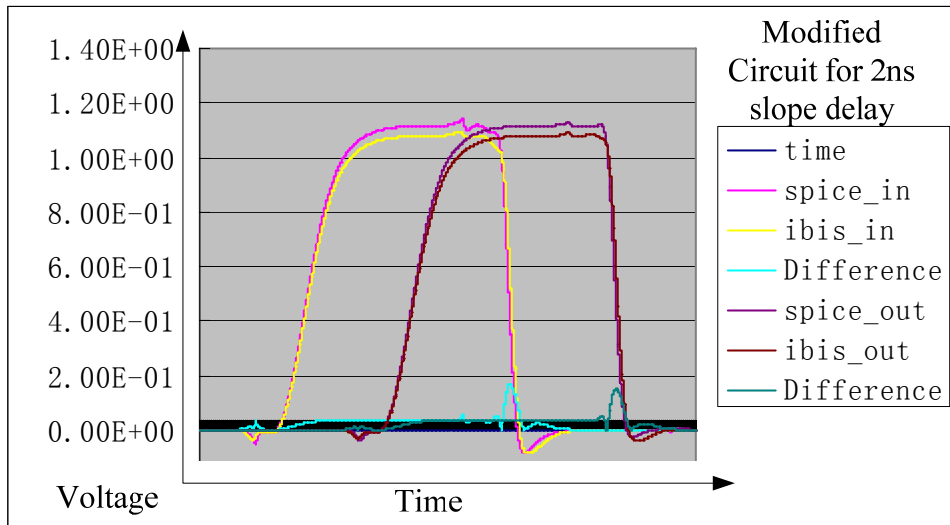


图 36 改进结构 2ns 斜坡输入仿真波形

表 13 改进结构的斜坡输入仿真误差分析

数据来源	峰值误差	峰值误差比	平均误差	平均误差比
理想输入	0.056235v	4.669%	0.019443v	1.614%
2ns 斜坡输入	0.67520v	55.123%	0.070182v	5.730%
改进结构 2ns 斜坡输入	0.16927v	13.813%	0.028002v	2.286%

从以上仿真数据对比可知，改进后的模型对带斜率输入的响应的峰值误差由原始的 55.123% 降低至 13.813%，平均误差从 5.730% 降低至了 2.286%。这样我们就可以对带斜坡输入进行准确的仿真。

5 总结与研究展望

5.1 总结

本文介绍了一种用于高速电路板级仿真的建模方法，**IBIS**。通过仿真一个高速 DDR-SDRAM 的数据输出电路的晶体管级模型，建立了这个输出端口的 **IBIS** 模型，通过横向采用特殊时刻点比较，纵向采用差值比较的方法，对建立的 **IBIS** 进行了验证，对于目前还没有规范的 **IBIS** 验证流程和方法是一种新的尝试

在此基础上，针对 **IBIS** 模型在不同仿真器中会有不同仿真结果的问题进行了研究，改进了 **IBIS** 模型瞬态响应仿真方法，提高了瞬态响应的仿真精度，并且针对信号经过传输线可能导致的边沿斜率，提出一种新的仿真方法，使得 **IBIS** 驱动器能够准确仿真非理想的信号激励。

5.2 研究展望

信号完整性问题是高速电路设计的瓶颈之一，随着时钟频率的上升，信号完整性必将越发影响电路的设计的传统流程。93 年提出的 **IBIS** 标准，作为一种用于高速电路板级仿真的方法，目前已广泛的被各大半导体厂商，系统设计公司和 EDA 公司接受和推广。而基于 **IBIS** 仿真的设计方法学和流程还没有完整的形成一个体系，因此有很多方向值得去研究和实现。

首先是 **IBIS** 模型的正确建立，模型的准确性直接影响到仿真结果的可信程度。还有 **IBIS** 模型的验证，目前 **IBIS** 发展最需要一个环节就是一套完整可信的验证方案，从而能够让更多习惯于传统设计方法的设计者足够信赖 **IBIS** 模型。第三点是 **IBIS** 模型的使用，只有采用正确的仿真方法才能发挥出 **IBIS** 模型快速准确的特点。当然随着电路设计的不断发展，亦需要对 **IBIS** 标准本身进行修改以适应新的仿真需求。

IBIS 对于板极信号完整性仿真依然算是一个新兴的事物，是一个充满了挑战，前景和机遇的领域。

参考文献

- [1] Peivand F. Tehrani, Yuzhe Chen, Jiayuan Fang, Department of Electrical Engineering, State University of New York at Binghamton, "***Extraction of Transient Behavioral Model of Digital I/O Buffers from IBIS***", 1996 Electronic Components and Technology Conference, PP:1009-1015,
- [2] Ying Wang, Han Ngee Tan, School of EEE, Nanyang Technological University, Singapore "***The Development of Analog SPICE Behavioral Model Based on IBIS Model***", 4-6 March 1999 , VLSI, 1999. Proceedings. PP:101 - 104
- [3] Bob Ross, Syed Huq, Jon Powell, EIA IBIS Open Forum Committee "***IBIS Models for Signal Integrity Applications***", EE Times, 1996, 18(9).38~43
- [4] A.Varma, M.Steer, P.Franzor "***Developing Improved IO Buffer Behavioral Modeling Methodology Based on IBIS***" 2006 IEEE Electrical Performance of Electronic Packaging P 69-72
- [5] Arpad Muranyi, Signal Integrity Engineering, Intel Corporation, "***Introduction to IBIS models and IBIS model making***", Intel Corporation Folsom, CA November 3-4, 2003
- [6] Arpad Muranyi, Signal Integrity Engineering, Intel Corporation, "***I-V curve linearity and buffer impedance***", IBIS Summit at DAC 2004 Manchester Grand Hyatt Hotel, Santa Diego, CA June 8, 2004
- [7] Bob Ross, Interconnectix Business Unit, Mentor Graphics Corporation, "***Introduction & Model Processing Algorithms***", IBIS Summit Meeting, Boxboro, MA October 15, 1998
- [8] Jinhua Chen, North East Systems Associates, Inc. "***Comparison between SPICE and IBIS I/O Device Simulations***" IBIS Summit at DAC 2004 Manchester Grand Hyatt Hotel, Santa Diego, CA June 8, 2004
- [9] Bob Ross, Interconnectix Business Unit, "***IBIS Die V-T Tables from Part or Board Measurements***" DesignConIBIS Summit Meeting Santa Clara, California February 2, 2004
- [10] Ambrish Varma, Michael Steer, Paul Franzon, "***A New Method to Achieve Improved Accuracy with IBIS Models***" IBIS Summit at DAC 2004 Manchester Grand Hyatt Hotel, Santa Diego, CA June 8, 2004

[11] Roland H.G.Cuny, Development Support, Siemens Nixdorf Informationsystems AG, Paderbom, Germany, “*SPICE and IBIS Modeling Kits the Basis for Signal Integrity Analyses*” Electromagnetic Compatibility 19-23 Aug. 1996 PP:204 - 208

[12] Jerry D Hayes and Larry Wissel, IBM Microelectronics Division, “*Behavioral Modeling for Timing, Noise, and Signal Integrity Analysis*”

[13] “*IBIS (I/O Buffer Information Specification)*”, The IBIS Open Forum, Version 3.2, August 1999.

[14] Peters S., “*IBIS Forum I/O Buffer Modeling Cookbook*”, the IBIS Open Forum, Revision 2.0X

[15] Lance Wang, ZhangMinZhong Cadence Design Systems, Inc. “*Using IBIS for SI Analysis*” Asian IBIS Summit, ShenZhen, P.R. China December 6, 2005

[16] Technology and Research Labs “*Code (Equation) Based Models, an accurate way to represent, and transfer simulation models*”

[17] Bob Ross, Interconnectix Business Unit, “*I-T Tables and BIRD42.3 Revisited*” IBIS Summit Meeting Design Automation Conference 2004 San Diego, California June 8, 2004

[18] P. Forstner, “*Input and Output Characteristics of Digital Integrated Circuits*”, Application Report, Texas Instruments, August 1994.

[19] Arpad Muranyi, Signal Integrity Engineering, Intel Corporation “*An algorithm to model overclocking more accurately*” IBIS Summit at DesignConEast 2003 Royal Plaza Hotel, Marlborough, MA June 23, 2003

[20] Bob Ross, “*Buffers for Advanced SPICE to IBIS Testing*” IBIS Summit Meeting Design Automation Conference 2006 San Francisco, California July 25, 2006

[21] F. G. Canavero, I. A. Maio, I. S. Stievano, “*Identification of Models for Digital Integrated Circuit Ports from Measured Transient Waveforms*” IEEE Instrumentation and Measurement Technology Conference Budapest, Hungary, May 21-23, 2001

[22] F. G. Canavero, I. A. Maio, I. S. Stievano, “*Black-Box Models of Digital IC ports for EMC simulations,*” in IEEE International Zurich Symposium on Electromagnetic Compatibility, Zurich, CH, February 20-22, 2001.

[23] Stephen, Garrett, James, “*High-Speed Digital System Design : A Handbook of Interconnect Theory and Design Practices*” H.Hall, W.Hall, A.McCall

[24] Brian Young “*Digital Signal Integrity-Modeling and Simulation with*

Interconnects and Package” Prentice-Hall PTR

[25] Eric Bogatin “**Signal Integrity: Simplified**” Pearson Education Prentice Hall

[26] “**HSPICE® Simulation and Analysis User Guide**”, SYNOPSIS

[27] “**HSPICE® Signal Integrity User Guide**”, SYNOPSIS

[28] **MT74Hx DDR-SDRAM Data Sheet**, Micron,

<http://download.micron.com/pdf/datasheets/dram/ddr2/1GbDDR2.pdf>

[29] 蔡兴建, 毛军发, 陈建华, 李征帆 **高速数字IO 缓冲器瞬态行为建模及其在同步开关噪声分析中的应用** 电子学报 2000 年第 11 期 PP.36-38,42

[30] 陈建华, 毛军发, 蔡兴建 王德东 **基于 IBIS 模型的 CMOS 电路同步开关噪声的计算和优化** 上海交通大学学报 2001 年 35 卷第六期 PP:848-851

[31] 方国华, 刘光斌, 余志勇 **基于 IBIS 模型的信号完整性仿真分析** 计算机科学与技术 2004 年 12 月第六期 PP:68-71

[32] 石章如, **IBIS 模型的仿真研究与应用分析** 武汉理工大学学报 2005 年第 5 期 PP 9-12

[33] 童智勇, 韩月秋 **基于 IBIS 模型的仿真分析在高速 DSP 系统设计中的应用** 电子技术应用 2000 年第 12 期 PP:13-15

符号说明

IBIS	输入输出缓冲器信息标准
C_comp	伴随电容
Rising_waveform	上拉瞬态曲线
Falling_waveform	下拉瞬态曲线
Pull_up	上拉稳态曲线
Pull_down	下拉稳态曲线
GND_clamp	地箝位曲线
Power_clamp	电源箝位曲线

攻读学位期间发表的学术论文

1. 第一作者,《非理想逻辑输入对 IBIS 仿真的影响及仿真方法》,微电子学与计算机,中文核心期刊