VHDL与集成电路设计丛书

用VHDL设计电子线路

For Designers

Stefan Sjöholm, Lennart Lindh 著

边计年 薛宏熙 译

清华大学出版社

http://www.tup.tsinghua.edu.cn

VHDL for Designers

用 VHDL 设计电子线路

Stefan Sjoholm, Lennart Lindh 著

边计年 薛宏熙 译

清华大学出版社
Prentice-Hall

（京）新登字 158 号

书　名：用 VHDL 设计电子线路
作　者：stefan Sjoholm，Lennart Lindh 著　边计年　薛宏熙 译
出版者：清华大学出版社（北京清华大学学研大厦，邮编 100084）
　　http://www.tup.tsinghua.edu.cn
印刷者：世界知识印刷厂
发行者：新华书店总店北京发行所
开　本：787×1092 1/16　　印张：25.5　字数：618 千字
版　次：2000 年 8 月第 1 版　2000 年 8 月第 1 次印刷
书　号：ISBN 7-302-03932-1/TP・2300
印　数：0001～5000
定　价：42.00 元

序　　言

这是我们的第三本关于 VHDL 的书。本书篇幅稍长，目的是覆盖 VHDL 新增加的部分（包括 VHDL-93 新标准），且尽量详细。我们很高兴得知一些大学已经在使用本书，而且工业界的技术决策者和开发工程师也在阅读本书。

本书的目的是教会读者用现有开发工具在设计电子系统的实际工作中如何使用 VHDL，也包括建立环境模型（测试基准）的方法。

本书包含丰富的示例，因而既可以作为各种水平的教科书，也可以作为今后工作的参考书。为此我们尽自己的努力使本书简明易懂。

VHDL 已经用于设计多年，本书反映了当前所提供的可能性。系统组成、开发方法、综合工具以及测试方法等都在迅速发展。譬如，行为综合已经被确认是一个最重要的领域。因而本书提供了对行为综合及其应用可能性的介绍。我们也准备再过若干年当新技术发展之后对本书进行再版。

VHDL 是一个新语言。在某些方面，VHDL 语法类似于其他流行语言如 C 和 Pascal，但其功能完全不同。C 和 Pascal 适应于一个 CPU，即一个时序机，在某一时刻只有一条指令在运行。而 VHDL 适应于硬件的一般结构。硬件的结构大部分是并行的。实际结果表明：若使用 VHDL，其性能总是比常规的 CPU 的程序设计语言在性能上要高几个数量级。这就是说，在使用软件语言的地方，VHDL 作为一个实现的语言会用得越来越多。我们相信，在将来大部分的小型微控制器将取代 CPU 程序设计语言和机器码，而用 VHDL 设计。

VHDL 模拟器已经在市场上出现多年，可以"执行"VHDL 代码，验证其功能。VHDL 代码可以自动被转换为 FPGA（field programmable gate array）或门阵列（gate matrix）。从 VHDL 代码到某一工艺技术实现的转换称为综合。综合所产生的硬件应当与已经模拟过的 VHDL 代码有相同的行为（功能），这一点是最重要的。写一个综合用的一般性 VHDL 代码是不容易的，不同的综合工具只能转换 VHDL 语言的不同部分。为此，本书的例子针对 3 个不同综合工具。这 3 个综合工具为：Synopsys，Autologic（Mentor Graphics）和 ViewLogic。ViewLogic 是一个基于 PC 或工作站的系统，而 Synopsys 和 Autologic 都是基于工作站的系统。

对于大多数设计者来说，用 VHDL 语言描述行为，而不是用门级设计来描述，需要从一开始就调整观念。但是，为了达到高设计能力和高质量，这种调整是必须的。通过用与工艺无关的高层次语言 VHDL 描述设计，设计者可以对其设计精细调整，使其做到功能正确。VHDL 所提供的"设计效益"和鲁棒性要比门级设计大得多。VHDL 与综合工具结合使用也意味着工艺的改变很快、很容易。譬如，用 FPGA 实现 ASIC（application specific ntegrated circuits）的所谓快速原型（rapid prototyping）就是利用这个特性。

要进行系统的、鲁棒的、结构化的、可测的和与工艺无关的设计，要求有扎实的 VHDL 知识，这就是本书所要提供的。进而，如果利用 VHDL 这样的高层次语言描述设计的优点，还需要好的设计方法和好的测试方法。因此，本书也包含关于设计方法和测试方法的章节。

VHDL 代码中所有的 VHDL 保留字都用黑体字印刷。

本书中有一些图是综合工具自动产生的，我们引用这些图的目的是为了说明自动综合的结果，因而没有为了改善可读性而修改原图中的文字。

如果读者对本书内容有任何意见，请告诉我们：

Stefan Sjoholm，ABB Industrial Systems，721 67 Vasteras，Sweden

(E-mail：stefan.j.sjoholm@se.abb.com 或 ssm@mdh.se)或

Lennart Lindh，Malardalens hogskola，Idt，Box 883，721 23 Vasteras，Sweden

(E-mail：lennart.lindh@mdh.se)

VHDL 是 VHSIC hardware description language 的缩写，而 VHSIC 是 very high speed integrated circuit 的缩写。

我们在这里要特别感谢 Synopsys，Viewlogic，Mentor Graphics，Motorola，Texas Instrument，Xilinx 以及其他单位的大力支持。

在这里我们还要对清华大学边计年教授、薛宏熙教授表示感谢，感谢他们把本书翻译成中文，介绍给中国读者。

Stefan Sjoholm，Lennart Lindh

1999 年 4 月

译 者 序

VHDL 语言是一种标准化的硬件描述语言，已经大量出现在各公司的商业电子设计自动化(EDA)系统中。许多硬件设计者用 VHDL 描述系统的行为，利用设计自动化工具达到自动设计的目的。VHDL 的优势正在迅速地显现出来。

VHDL 的特点是覆盖各个层次，包括行为描述、数据流描述和结构描述。它给电子线路设计者提供了很大的自由度。也正是这个原因，使得设计者在初学 VHDL 进行硬件描述时会遇到困难，往往会觉得无从下手。另外，并不是所有的 VHDL 语句都可以被综合器所接受，设计者在描述自己的设计时必须按照 EDA 工具的要求进行描述。因此，设计者除了要求学会 VHDL 语言的语法之外，还必须学会对不同硬件的具体描述方法。

我们为读者介绍的这本书正是为设计者解决这些问题而写的。作者以他们丰富的实践经验，介绍用于实际电子线路的描述方法，介绍许多种不同电路元件的描述方法，介绍针对不同综合工具的不同描述方法和注意事项。

利用设计自动化工具是高效进行电子线路设计的手段，已逐渐在我国推广开来。许多人已认识到设计自动化工具在进入 21 世纪后将成为硬件设计者所必须掌握的基本技能。许多高等学校已先后开设了电子线路设计自动化的课程。我们相信，把本书介绍给国内读者，必将对我国电子线路设计自动化的发展和普及起到重要作用。

本书根据 Prentice Hall 出版的英文版“VHDL for Designers”翻译。边计年译第 1～8 章和第 19 章，薛宏熙译第 9～18 章。在翻译过程中所发现原文中的错误都与原作者商讨后改正。鉴于译者的水平，一定存在错误和不足之处，望读者批评指正。

译 者

1999 年 8 月于清华大学

E-mail 地址：[bianjn | xuehx]@tiger.cs.tsinghua.edu.cn

目　　录

1 概　述

VHDL 在 80 年代初期出现，并且已经被接受为描述、验证和设计电子线路的最重要的标准语言之一。目前，许多高技术公司以 VHDL 作为数字系统的唯一描述语言。许多大学开设了 VHDL 语言课程，并且有几个 VHDL 组织为新用户提供支持。预计在几年之后，一些中小公司将有可能使用 VHDL。VHDL 和 ASIC(application specific integrated circuit，专用集成电路)也已经开始与单片控制电路竞争。VHDL 将使设计者不必再使用冯·诺伊曼结构，允许他们的设计是真正并行，而不再是时序机。这给设计者开辟了一种全新的设计电子线路的可能性。

使用 VHDL 以代替传统原理图进行电子系统设计有两条理由：缩短设计时间，简化维护工作。

VHDL 作为一个规范语言和建模语言，其第一个模拟器出现于 80 年代后期。80 年代末，则随着 VHDL 的标准化出现了一些工具，而又过了几年才开始在设计中使用 VHDL。目前，各主要的工具制造商都支持 VHDL 标准。现在，VHDL 已成为标准化的语言，优点是易于将 VHDL 代码在不同的商业平台和工具之间交换。这意味着在某一个工具制造商的模拟器用的 VHDL 程序可以不加修改地移到另一个工具制造商的模拟器。然而不幸的是 VHDL 对于设计(综合)还没有标准化，尽管对设计数字电路模型是标准化的。最近几年，对越来越多的语言构造部分的综合成为可能。

幸运的是，VHDL 工具，尤其是 VHDL 模拟器，近年来也已经在 PC 上开发出来，其价格戏剧性地下降，使得较小的公司也可以使用 VHDL 工具。也有 PC 综合工具，主要是用于 FPGA 和 PLD，但它们的功能要比工作站少一些。

一种可以改变电子设计者的世界的新的可能性，是将成千上万个门和触发器作为一个集成电路描述，在一台不太昂贵的 PC 设备上只用几分钟时间。这意味着，有可能通过描述电路自动产生一个文件；这种方法称为快速原型(rapid prototyping)。目前单个电路中已经包含多达 10 万个门。快速原型也已经用于小型产品系列中。

用 VHDL 代码，而不是用原理图(譬如门)，是一种新的设计方法。使用 VHDL 不只是意味着编写代码，而且也便于建立层次结构和用元件库进行设计。VHDL 也利于编写标准电路的代码(譬如 Motorola 68020)。现在已有一些公司出售模拟用的标准元件。这意味着使用标准元件的整个电路板的模型可以通过计算机模拟进行验证。如果模拟整个电路板，诸如访问时间、非法地址等可以用高强手段检测出来。通常可以对每个元件专门加一段相应的程序段，用来检测接口信号是否按时到达，所用地址是否合法等。

所谓用 VHDL 设计是指由设计者编写代码，然后用模拟器验证其功能，最后把这些代码综合成一个网表。综合可以与把代码编译成机器码的编译器相类比。在硬件设计中，VHDL 代码翻译成由门和触发器组成的原理图。

早在 1980 年，美国国防部开始进行 VHDL 的开发，因为美国军事工业需要描述电子系统的标准方法。1987 年由 IEEE(Institute of Electrical and Electronics Engineers)将 VHDL

制订为标准。参考手册称为 IEEE VHDL 语言参考手册标准草案 1076/B 版(IEEE VHDL Language Reference Manual Draft Standard version 1076/B)，于 1987 年批准，称为 IEEE1076-1987。应当注意，起初 VHDL 只是作为系统规范(system specification)的一个标准，而不是为设计而制订的。

VHDL 非常类似软件语言 ADA。部分原因是五角大楼委托研究新语言的公司(研究所)有大量的使用 ADA 的经验。目标是使 VHDL 标准像其他标准一样，应当每 5 年更新版本一次。第 2 个版本 VHDL-93 被延迟到 1993 年，它相对于 VHDL-87 没有什么大变化，只是增添了主要是对 VHDL 模型的某些新的 VHDL 命令和属性。本书把新增加的内容标以"VHDL 93"。

VHDL 最大的成功在于它是唯一制订为标准的硬件描述语言。事实上，ADA 比其他程序设计语言诸如 C 和 C++等要麻烦的多。

1.1 为什么要用 VHDL

用 VHDL 进行设计与传统的原理图设计技术相比有很多优点。本节介绍该语言的优点和缺点。

VHDL 支持数字电路的开发环境。VHDL 也支持各种设计方法：自顶向下、自底向上或混合的方法。当今许多电子产品的生命期大约 10 年，同时又必须多次重新设计，以利用新的技术。其最简单的方法是用与工艺无关的 VHDL 设计方法，即使用自动工具来改变工艺。一个电子产品的 10 年的生命期中，通常需要进行修改，增加新的功能。VHDL 支持可修改性，因为语言易读、层次化且结构化。

VHDL 语言支持层次化(框图)、元件再利用、出错处理和验证。层次化可以利用结构 VHDL、过程和函数描述。结构 VHDL 类似于框图。许多系统支持图形输入，并可以自动转换为结构 VHDL。VHDL 语言支持并行的和顺序的语言结构。它还支持从规范到门级描述的所有各级。

VHDL 元件的设计可以与工艺无关，或者或多或少与某一类工艺无关。元件可以保存在设计库中，在多个不同的设计中再利用。这样就可以在市场上买到商用集成电路标准元件的 VHDL 模型，当要验证整个电路板时，这是一个很重要的优点。元件也可以设计成可自动改变的。例如，FIFO(先进先出)元件可以用这样的方法设计：只需要唯一的一个元件来覆盖每一种可能性(大小和行数)。这种元件称为通用元件 (generic component)。

VHDL 的代码可以用模拟器验证其功能。模拟器施加输入信号进行模拟("执行")，并以元件为基础产生信号图和出错信息。输入信号的描述或者用 VHDL，或者用模拟器语言。当对 VHDL 代码进行模拟时，就实现了功能验证。其后可以再对设计作时序验证。

在传统的原理图设计中，设计者必须用手工检查与工艺有关的因素如时序、面积、驱动强度、元件选择和扇出。用 VHDL 进行设计的一个很大的优点是设计者可以专心致力于其功能，即需求规范的实现，而不需要对不影响功能的与工艺有关的因素花费过多的时间和精力。

VHDL 作为一个标准，其代码有可能在不同的系统之间交换以建模(模拟)。而对设计而言，因为还没有一个标准，难以交换。因此，本书指明对于 ViewLogic, Mentor Graphics

和 Synopsys 等公司的综合工具应当如何写可综合的 VHDL 代码。这 3 种工具体现了当前应用于工业、大学的综合工具有一个大市场。ViewLogic 是一个基于 PC 的系统(也适用于工作站),而且与另外两个公司的综合工具的功能不相同。Synopsys 和 Mentor Graphics 的 Autologic 2 是基于工作站的系统,比较贵,但其功能比 ViewLogic 强。90 年代 Synopsis 在 VHDL 综合方面一直是处于领先地位的公司,但也面临着一系列的挑战,首先是 Autologic 2 在综合方面的挑战。幸运的是,Synopsys 和 Autologic 2 都支持很大的同样的 VHDL 子集(99%),因而 VHDL 代码可以在这两个领先综合工具中不加修改而相互交换。

本书从简单的 PC 综合工具到高级的工作站综合工具介绍如何写可综合的 VHDL 代码,因而读者如果使用上述 3 种工具之外的系统工具时,使用本书的综合模型将不会有困难。

VHDL 还没有对模拟电路进行标准化。在 VHDL 基础上加上模拟电路的扩展(AHDL)正在进行*。这个新标准以整个 VHDL 标准为基础,并增加了许多描述模拟电路功能的内容。

1.2 开发流程

本节介绍开发过程的总体开发方法。它可以较好地用于小规模的学生实验和课题。

产品的开发阶段是产品生命周期中的第一阶段。在这个阶段给产品指定规范、进行设计并验证。通常的开发模型称为瀑布模型。它从指定规范开始,然后引导设计步骤形成功能原型。假如我们要在实验室里作一个实验,就需要一个写得很好的技术规范,自上而下的设计方法对此是一个理想的模型。

从规范到原型的开发流程可以划分成如下步骤(图 1.1):

开发阶段	结果	文档
分析	规范	要做什么?
设计	VHDL 代码	如何做?
工艺映射	网表	
建立原型	原型	结果如何?

图 1.1 设计流程概览

- 分析阶段包括写规范(specification)。规范可以用 VHDL 或自然语言来写。规范的目的是要确定"要做什么"。规范可以用 VHDL 描述,然后用 VHDL 模拟器验证。
- 设计阶段将规范转化为总体结构和 VHDL 代码。这一阶段还没有可能完全自动化。本阶段从定义总体结构(框图)开始。当总体结构准备好了,就可以将各种元件(方框)或从库里取出预先准备好的元件写成 VHDL 代码。然后用模拟器验证设计结果的功能。当结果与规范一致时,就完成了设计阶段。这个阶段的主要问题是"总体结构和元件应如何设计"。

* 1998 年制定了 IEEE 标准,称为 VHDL-AMS,标准编号为 IEEE std 1076.1。AMS 是"Analog and Mixed Signal"的缩写。——译者注