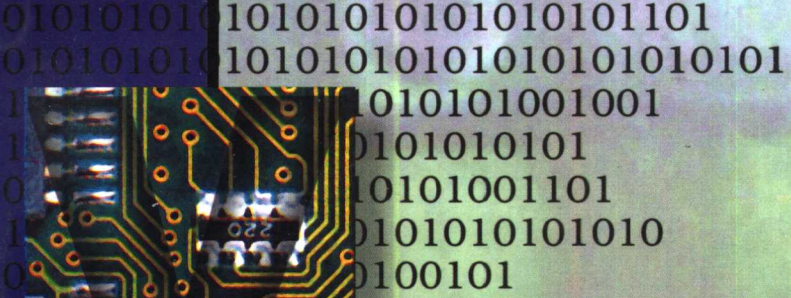
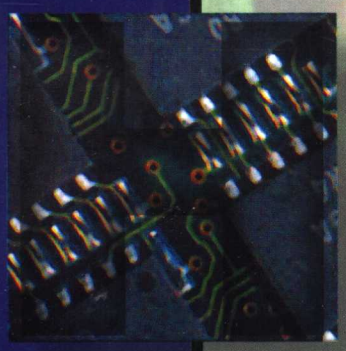


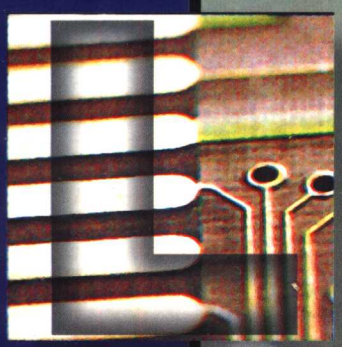
VHDL 简明教程

王小军 编著

乔长阁 边计年 薛宏熙 译

边计年 审校





```
ENTITY one_bit_adder IS
  PORT ( a, b, c_in : IN Bit
         c_out, sum : OUT Bit )
END one_bit_adder ;

ARCHITECTURE logic OF one_
  SIGNAL int : Bit ;
BEGIN
  int <= a XOR b AFTER 10 ns
  c_out <= (a AND b) OR (int AN
  sum <= int XOR c_in AFTER
END logic ;
```


清华大学出版社

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

VHDL 简明教程

王小军　　编著

乔长阁　边计年　薛宏熙　译

边计年　　审校

清华大学出版社

(京)新登字 158 号

内 容 提 要

VHDL 是国际标准硬件描述语言,在电子系统自动设计中已十分流行,成为主要的硬件描述工具,将成为数字系统设计领域中所有技术人员必须掌握的一种语言。

本书是根据王小军博士 3 年来在都柏林城市大学电子系给本科生四年级和研究生授课时的英文讲稿翻译而成的,主要介绍 VHDL 语言的基础及其应用。书中给出了一些 VHDL 描述实例,一些 VHDL 的标准程序包,练习题,实验,自测题,以及从 Internet 网上得到有关 VHDL 的帮助及最新信息的途径等。书中的实例都是经过微机上的 VHDL 模拟软件 V－System/Windows 编译和模拟的,用于逻辑综合的实例都经过 SUN SPARC 工作站上的 VHDL 模拟软件 Quick VHDL 编译和模拟,然后作为逻辑综合软件 AutoLogic 的输入而自动生成逻辑电路图。

本书全面而系统,可作为电子类和计算机类各专业的高年级本科生和研究生的入门教材和教学参考书,也可作为数字电路设计人员了解 VHDL 的读物。

图书在版编目(CIP)数据

VHDL 简明教程/王小军编著;乔长阁等译. －北京:清华大学出版社,1997
ISBN 7-302-02647-5

Ⅰ. V… Ⅱ. ①王… ②乔… Ⅲ. 程序语言,VHDL-教材 Ⅳ. TP312

中国版本图书馆 CIP 数据核字 (97) 第 18087 号

出 版 者:清华大学出版社(北京清华大学校内,邮编 100084)
因特网地址:www.tup.tsinghua.edu.cn
责任编辑:马瑛珺
印 刷 者:北京市清华园胶印厂
发 行 者:新华书店总店北京科技发行所
开 本:787×1092 1/16 印张:12.25 字数:285 千字
版 次:1997 年 10 月 第 1 版 1998 年 4 月 第 2 次印刷
书 号:ISBN 7-302-02647-5/TP・1365
印 数:4001～8000
定 价:19.50 元

译 者 序

硬件描述语言是硬件设计者和电子设计自动化工具之间的界面，用于数字电路与系统的描述、模拟和自动设计。目前，电子系统向集成化、大规模和高速度等方向发展，电子系统设计自动化(EDA)软件工具在市场上大量出现，以硬件描述语言和逻辑综合为基础的自顶向下的电路设计方法在工业界已十分流行。自从 1987 年 VHDL 被 IEEE 确定为标准硬件描述语言(IEEE 标准 1076)以来，VHDL 迅速出现在各 EDA 系统中，取代了各自的非标准语言，并很快为数字系统的硬件设计者所接受。1993 年，IEEE 又推出 VHDL 新的版本 IEEE 标准 1076-1993，IEEE 还制定了与 VHDL 语言有关的标准逻辑系统程序包 Std_Logic_1164、标准数学程序包 Math，标准综合用程序包 Numeric_Std，Numeric_Bit，面向 ASIC 模型设计的 VHDL 基准 VITAL 等，并正在制定数字系统与模拟电路结合的描述语言 VHDL-A。我国自行开发的熊猫系统(PANDA)中，也已经把 VHDL 语言作为行为描述的输入手段。近年来，我国高等院校也开始重视 VHDL，许多学校已经把它列入电子、计算机等专业的教学大纲。VHDL 在硬件设计领域的作用将与 C 和 C++在软件设计领域的作用一样，在大规模数字系统的设计中，它将逐步取代如逻辑状态表和逻辑电路图等级别较低的、繁琐的硬件描述方法，而成为主要的硬件描述工具，它将成为数字系统设计领域中所有技术人员必须掌握的一种语言。

国外介绍 VHDL 的书籍非常多，而我国此方面的书还非常少。本书是根据王小军博士几年来在都柏林城市大学电子系给本科生四年级和研究生授课的英文讲稿翻译而成的，主要介绍 VHDL 语言的基础及其应用。作为 VHDL 语言的基础，它可以作为电子类和计算机类各专业的高年级本科生和研究生的入门教材和教学参考书，也可以作为数字电路设计人员了解 VHDL 的读物。读者可在阅读此书的基础上进一步阅读其它有关 VHDL 的书籍，以便掌握更高级的 VHDL 特性和应用。我们愿意将此书介绍给读者，以促进 VHDL 在我国的迅速普及。

参加本书翻译的有以下 3 位同志：薛宏熙译 1～4 章，边计年译 5～13 章，乔长阁译 14～23 章。全书由乔长阁负责统稿，边计年对全书进行了审阅。我们对全书中的程序部分进行了规范化处理，加入了图注、表注及其说明等，统一了程序中的大小写，并标识了 1987 年和 1993 年 2 个版本的区别。程序中关键词一律用大写表示，各标准中规定的标准标识符首字符大写。有关 1993 年版本新添加的关键词用黑体标识。除此之外，我们还对原书中的错误进行了修改。书末附了主要术语的英汉对照表。

我们在翻译过程中力求译文准确，风格统一，但由于时间仓促和译者的水平有限，不足之处在所难免。欢迎读者提出批评改进意见。

译者感谢原书作者王小军博士给了我们这个机会，并感谢北京邮电大学胡健栋教授向我们推荐这本书。

译 者

1997 年 4 月于清华大学

前　言

VHDL 是美国电气和电子工程师协会制定的标准硬件描述语言(IEEE 标准 1076)。VHDL 可用于数字电路与系统的描述、模拟和自动设计。

以硬件描述语言和逻辑综合为基础的自顶向下的电路设计方法可用于复杂的集成电路设计以缩短设计周期。这种自顶向下的电路设计方法在欧美工业界已十分流行。大的集成电路厂家,如摩托罗拉等,已开始使用 VHDL。美国国防部明确要求,自 1988 年 9 月 30 日起,所有为军方研制的专用集成电路必须用 VHDL 描述。教育界也开始重视并逐步把硬件描述语言 VHDL 列入大学电子系和计算机系的教学大纲。包括美国加利福尼亚大学戴维斯分校和英国曼彻斯特大学在内的许多高校的电子系和计算机系已经把 VHDL 作为本科生或研究生的课程。从帝国理工医学院的 Internet 资料库里可以找到世界各地有关的 VHDL 检索资料。VHDL 在硬件设计领域的作用将与 C 和 C++在软件设计领域的作用一样。在软件设计领域,C 和 C++等高级语言早已取代低级汇编语言,尤其是在大的软件设计项目中。同样的情况正在硬件设计领域发生。高级硬件设计语言 VHDL 将逐步取代低级硬件描述语言,如逻辑状态表和逻辑电路图等。

作者希望把 VHDL 介绍给国内的读者。本书是以作者 3 年来在都柏林城市大学电子系给本科生四年级和研究生授课的讲稿为基础写成的。该书的英文原稿曾于 1995 年 5 月在都柏林用作一家属于菲利普集团的硅软公司的 VHDL 培训教材。本书可作为电子系和计算机系的高年级本科生和研究生的入门教材,也可以作为数字电路设计人员了解 VHDL 的读物。书中的实例都经过微机上的 VHDL 模拟软件 V-System/Windows[1]编译和模拟的。用于逻辑综合的实例都经过 SUN SPARC 工作站上的 VHDL 模拟软件 QuickVHDL[2]编译和模拟,然后作为逻辑综合软件 AutoLogic[3]的输入而自动生成逻辑电路图。自动生成的逻辑电路图经过电路模拟软件进行模拟,已证实其实现了原 VHDL 模型的功能。

由于作者水平有限,不足之处在所难免。欢迎读者提出批评改进意见。作者的电子邮件地址为:wangx@eeng.dcu.ie。

Xiaojun Wang　　王小军
Dublin City University　　都柏林城市大学
Dublin 9, Ireland　　爱尔兰

〔1〕 V-System/Windows 是 Model Technology 的注册商标
〔2〕 Quick VHDL 是 Mentor Graphics 的注册商标
〔3〕 AutoLogic 是 Mentor Graphics 的注册商标

目　录

1 为什么要用 VHDL

1.1 为什么要用硬件描述语言 HDL

技术的飞速发展使集成电路的设计规模日益增大，复杂程度日益增高。伴随着设计规模的增大，门级描述变得难以管理，不得不采用更抽象层次的描述方法，并接受高层次的、自顶向下的设计方法。逻辑图和布尔方程曾经是描述硬件的方法，但随着系统复杂程度的增加，这种描述变得过于复杂，不便于使用。在高于逻辑级的抽象层次上，这种方法很难以用简练的方式提供精确的描述，在自顶向下的设计方法中不能再把它当作通常的描述手段，而硬件描述语言 HDL(hardware description language)则逐渐成为满足以上要求的新方法。HDL 与高层次的软件程序设计语言类似，同时它又提供了以下功能：

1. 在希望的抽象层次上，可以对设计进行精确而简练的描述；
2. 易于产生用户手册、服务手册等文件；
3. 在不同层次上都易于形成用于模拟和验证的设计描述；
4. 在自动设计系统中(例如高层次综合工具和硅编译器)作为设计输入；
5. 可以作硬件和软件的联合设计，消除了硬件和软件开发时间上的间隔；
6. 易于作设计的修改，易于把相应的修改并入设计文件中；
7. 在希望的抽象层次上，可以建立设计者和用户(老师和学生)的通信界面。

和通常的软件程序设计语言不同，HDL 的主要目的是用来编写设计文件并建立硬件器件的模拟模型。硬件系统的基本性质和硬件设计的方法决定了 HDL 的主要特性。HDL 的语法和语义的定义是为了能描述硬件的行为，它应当能自然地描述硬件中并行的、非递归的特性以及时间关系。

正如汇编语言被高级程序设计语言所代替那样，门级电路框图必然被 HDL 所代替。软件设计者很喜欢使用高级语言程序的编译方法，HDL 和硅编译器使得硬件设计者也能使用这种高层次设计的编译方法。

1.2 各种 HDL 的浏览

自从 Iverson 于 1962 年提出 HDL 以来，已经出现许多 HDL。其中，绝大多数是专有产品，包括 Silvar-lisco 公司的 HHDL(hierarchical HDL)，Zycad 公司的 ISP，Gateway Design Automation 公司的 Verilog 以及 Mentor Graphics 公司的 BLM。高等学校和科研单位中也有上百种 HDL[IDTC92]，例如 AHPL，MIMOLA 以及 SCHOLAR。此外，一些大型计算机制造商也都有其内部使用的设计语言。例如 TIHDL 就是德克萨斯仪器公司的硬件描述语言，在该公司内部以其客户中采用 TIHDL 来描述其设计。

某些 HDL 是从已有的软件程序设计语言发展而来，例如：BLM，MIMOLA 和 SCHOLAR 是由 PASCAL 发展而来，而 Silicon Compiler 公司的 M 以及 Gateway 公司的

Verilog 则以 C 语言为基础。英国国防部在皇家信号和雷达研究所于 1979 年开发了 ELLA，在 VHDL 出现之前广泛应用于欧洲。UDL / I 在日本以标准 HDL 的形式出现。多年以来，设计者一直使用这些专用的 HDL 来描述他们的军用或民用芯片设计。为了能继续生存，Verilog 和 ELLA [MoC194]已经公开，因为它们最后将成为为数不多的标准 HDL 之一，而根据定义，这样的标准不能是专用语言。

EDIF 经常被误认为是一种硬件描述语言。事实上，EDIF 不是一种语言，而是一种格式，正如它的名字：电子设计交换格式 EDIF(electronic design interchange format)。我们注意到这个细小的差别是因为：EDIF 原来就不准备被人们用于阅读或书写。EDIF 是由 CAD 工具制造商们鼓动起来的，用于工作于不同数据格式上的 CAD 工具之间交换设计数据。EDIF 可能适合于纯描述性文件，但不适合于重新设计的过程，因为 EDIF 中未包含行为描述。

1.3 VHDL 的诞生

美国国防部的项目有众多的承包人，他们使用过多的设计语言，使得承包人甲的设计不能被承包人乙再次利用，这就造成了信息交换困难和设计维护困难。为了解决这个问题，美国国防部为他们的超高速集成电路计划 VHSIC(very high speed integrated circuit)提出了硬件描述语言 VHDL(VHSIC hardware description language)，这个任务交给了德克萨斯仪器公司、IBM公司和Intermetrics公司。1987年12月IEEE接受VHDL[Leib89]为标准HDL，这就是今天我们所了解的IEEE Std 1076-1987[LRM87]。此后又作了若干修改，增加了一些功能，新的标准版本记作IEEE Std 1076-1993[LRM93]。严格地说，VHDL'93 和 VHDL'87 并不完全兼容(例如，增加了一些保留字并删去了某些属性)，但是，对 VHDL'87 的源码只作少许简单的修改就可以成为合法的 VHDL'93 代码[BFMR93]。

表 1.1 VHDL 的发展过程

年	第一季度	第二季度	第三季度	第四季度
1983	美国国防部提出需求	和 TI，IBM 及 Intermetrics 公司签约	开始工作	
1984	V2.0		V5.0	V6.0
1985			V7.2	IEEE 开始标准化工作
1986	VHDL 分析和标准化小组第一次会议	语言参考手册第一次草稿		语言参考手册第二次草稿 1076//A
1987	评审 1076 / A	语言参考手册 1076 / B	反对票的出现	IEEE Std 1076-1987 获批准
1992				再次投票通过
1993		第二次投票通过		IEEE Std 1076-1993 获批准

1988年9月30日之后，美国国防部要求开发ASIC的合同文件一律采用VHDL文档。VHDL逐渐演变为工业标准的过程示于表1.1。

1.4 设计的表示方法

设计的表示方法涉及两方面的问题：领域和层次。在表1.2中，垂直方向表示抽象层次，水平方向表示领域。表1.2中有行为、结构和物理等3个领域。对于每一个领域，都可以分为5个抽象层次：系统级(architectural 或 system)、算法级(algorithmic，sub-system 或 chip level)、寄存器传输级(register-transfer，functional block 或 micro - architecture)、逻辑级(logic 或 gate level)和电路级(circuit)。

表1.2 设计的表示方法：领域和抽象层次

层 次	领 域		
	行 为	结 构	物 理
系统级	性能描述	CPU、存储器、开关、控制器以及总线之间的逻辑连接	芯片、模块、电路板以及子系统的物理划分
算法级（子系统级或芯片级）	I/O应答算法（数据结构操作）	硬件模块 数据结构	部件之间的物理连接（board，floor-plan）
寄存器传输级	并行操作，寄存器传输，状态序列（状态表）	ALU、多路器、寄存器、总线、微定序器、微存储器等功能块的物理连接	芯片、宏单元等
逻辑级	布尔方程	门、触发器、锁存器	标准单元布图
电路级	微分方程	晶体管、电阻、电容	晶体管布图

不幸的是，在一些人的概念中，许多抽象层次名字的含义早已和某个特定的领域有了联系，而不是像表1.2所示的那样和3个领域都有联系。例如，某些人可能认为逻辑级和电路级只是结构领域的元件；而算法级只是行为领域的1个级别，某些作者也把物理领域称作几何领域。尽管如此，更多的人还是接受了表1.2所示的名字。

行为领域描述1个设计的基本功能，或者说该电路或设计应该做什么。从概念上讲，纯行为是输入和输出关系的描述，例如布尔方程组就是组合逻辑网络的行为描述。

结构领域描述逻辑结构，或者说描述设计的抽象实现，典型的是抽象模块相互连接的网表。例如，在寄存器传输级，抽象模块是ALU、多路选择器、寄存器等。

物理领域描述设计的物理实现，或者说把结构领域里的抽象元件代之以真正的物理元件。例如在寄存器传输级，物理领域描述实现ALU、多路选择器和寄存器等所需的平面布图。在所有的抽象级别里，速度、功耗以及面积约束等通常是物理领域的一部分。

无论是领域还是层次级别，他们的边界都是可以重叠的。1个设计的描述，通常在领域方面以及在层次级别方面都可以混合表示。1个典型的寄存器传输级描述，通常既包含行为特性又包含结构特性，既包含寄存器级元件又包含逻辑级元件。任何设计的最终实现是物理

实现，换句话说，集成电路的制造数据库应当由许多多边形来表示。

1.5 VHDL 的能力范围

VHDL 既可以被计算机阅读又可以被人阅读，它支持硬件的设计、验证、综合和测试。此外，它还支持硬件设计数据的交换、维护、修改和硬件的实现[LRM87][WaSC89]。

VHDL 支持行为领域和结构领域的硬件描述，并且可以从最抽象的系统级一直到最精确的逻辑级。VHDL 主要优点之一是：在描述数字系统时，可以使用前后一致的语义和语法跨越多个层次，并且使用跨越多个级别的混合描述模拟该系统。因此，可以对由高层次行为描述子系统及低层次详细实现子系统所组成的系统进行模拟。

上述特性使得系统的原始描述能够方便地反映设计者的意图。在维护系统、重新设计或更改部分设计时，可以用原来的测试集对修改过的 VHDL 描述重新模拟。

VHDL 描述能力的范围示于图 1.1 [Hand90]。

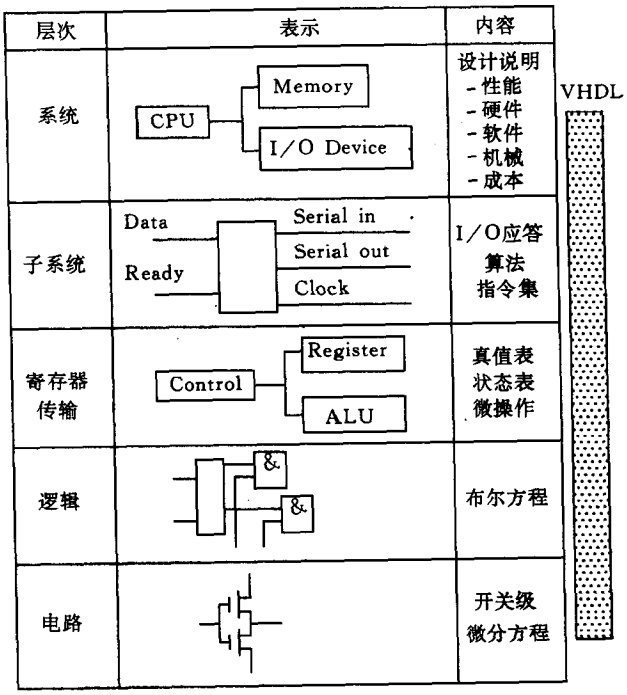


图 1.1 VHDL 描述能力的范围

请注意，VHDL 并不具有描述模拟电路的能力。虽然某些研究的结果以实例表明，VHDL 的描述能力可以扩展到电路级[HaSt91]，但是在这个级别上，VHDL 不是一种理想的语言。把 VHDL 的描述能力扩展到描述模拟电路由 IEEE 的 1076.1 小组进行。该小组正在设计一种新语言，以便既能描述模拟电路又能描述数模混合电路。这种新语言和我们即将

提到的描述数字电路的 VHDL'93 非常接近。该小组计划将其成为标准[VTfa94][1]。

1.6 一个 VHDL 的描述实例

图 1.2 所示是 1 位加法器的逻辑图,其 VHDL 的描述如下所示 :

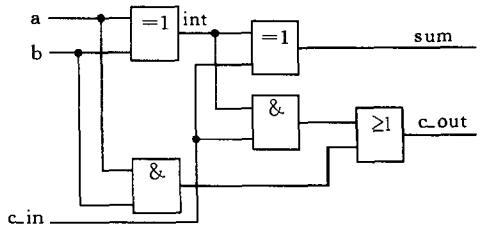


图 1.2 1 位加法器逻辑图

```
ENTITY one_bit_adder IS
  PORT ( a, b, c_in : IN Bit ;
         c_out, sum : OUT Bit );
END one_bit_adder;

ARCHITECTURE logic OF one_bit_adder IS
  SIGNAL int : Bit;
BEGIN
  int <= a XOR b AFTER 10 ns;
  c_out <= ( a AND b) OR ( int AND c_in ) AFTER 20 ns;
  sum <= int XOR c_in AFTER 10 ns;
END logic;
```

1.7 VHDL 的 CAD 工具

许多公司都提供 VHDL 工具,其中包括 VHDL 的创始者(例如 Intermetrics 公司,不过该公司的 VHDL 分部已被 Valid 公司收购,而此后不久,Valid 公司又并入 Cadence 公司),也包括一些大型 EDA 的公司(例如 Cadence, Mentor Graphics, ViewLogic, Synopsys 等),还包括一些专门领域的公司(例如 CLSI, Model-Technology, Vantage 等)。VHDL 模拟器和综合器已经是商业软件,可以在多种工作平台上运行(VHDL 描述不是 100%可以被综合的,不同公司提供的工具软件在描述风格和构造方面会有不同的指南和限制,满足这些要求的 VHDL 描述才能被该公司提供的工具所综合)。

VHDL 技术小组(http : //www. vhdl. com/VHDLVendors/Vendors. htm)已对许多 VHDL 商家的工具软件作过编译。

欧洲有 1 个促进元件、子系统及微系统技术发展的组织 EUROPRACTICE (http : //

[1] 目前仍在制定中。——译者注

www. te. rl. ac. uk/europractice)，通过他可以得到和 VHDL 有关的工具。图 1. 3 列举的仅是 VHDL 工具的一部分。

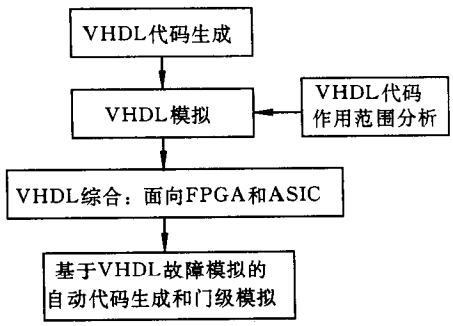


图 1.3　VHDL 的 CAD 工具

Altera

Altera 公司的 PLSM-VHDL(适用于个人计算机)和 PLSM-VHDLWS(适用于 SUN4 和 HP 9000 工作站)是 VHDL 的设计输入工具，支持该公司的 Classic，Max 5000，Max 7000，Flex 8000 等可编程器件系列。Altera 还提供和 Mentor-Graphics，Cadence 以及 Synopsys 工具的接口和库。

Cadence

该公司宣称他们的 VHDL 模拟器(Leapforg)是工业界最快的 VHDL 模拟器，并且和逻辑综合工具的 Synergy 系列紧密相连[VTfa94]。

Mentor Graphics

该公司提供 VHDL 模拟器 QuickVHDL 和 QuickSim II，以及 VHDL 综合工具 Autologic VHDL。

Synopsys

该公司提供 VHDL 系统模拟器(VSS Expert)和 VHDL 编译器。该模拟器有 1 条命令，可以告知程序的其中 1 行执行了多少次。留意观察这些数字，就可以知道该模型或测试基准的哪一部分执行的次数最少。

根据设计者的输入数据以及设计者目前所处的设计阶段，Synopsys 的 Design Power 可以快速给出比较精确的功耗反馈信息。在设计的早期阶段，Design Power 根据设计者对电路状态转换频率的粗略估计，可以快速给出功耗反馈信息，设计者利用此信息可以对设计指标作出折衷选择。在设计的后期，在门级优化阶段，Design Power 利用门级模拟器提供的精确的电路状态转换数据，可以给出精确的功耗反馈信息。

Model-Technology

该公司提供在个人计算机 Windows 环境下运行的 VHDL 模拟器(V-System/Windows)，它也有在工作站上运行的版本。

此外，VEDA 和 XILINX 公司也支持 VHDL 设计输入。VEDA 有 1 个叫作 VHDL Path 的工具，它可以计算语句的作用范围、分支的作用范围、条件的作用范围、路径的作用范围和信号的作用范围，并给出统计报告。

习题

1. 什么是 VHDL？简述 VHDL 的发展史。
2. 用领域和层次的概念简述 VHDL 的描述能力。
3. VHDL 适合于设计流程中的哪些部分？

2 基本的 VHDL 模型结构

2.1 设计实体

设计实体是 VHDL 中的基本单元和最重要的抽象，它可以代表整个系统、1 块电路板、1 个芯片、1 个单元或 1 个门电路。它可以代表像微处理器那样复杂的电路，也可以代表像单个逻辑门那样简单的电路，对于设计实体可以代表什么几乎没有限制。实际上，1 个设计实体由 1 个实体说明和 1 个结构体组成，如图 2.1 所示。在 VHDL 中，设计实体是 1 个模块，这个模块可以是设计中的 1 个元件，也可以是设计的顶层模块。

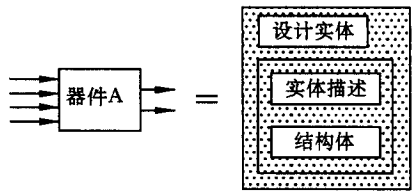


图 2.1 设计实体的表示

2.2 实体说明

设计实体是 1 个初级设计单元，它可以单独编译并且可以被并入设计库。它给实体命名并给实体定义 1 个接口，接口信息用于和其它模块通信。实体说明是 1 个器件的外部视图，即从器件外部看到的器件外貌，其中包括该器件的端口。实体说明也可以定义参数，并把参数从外部传入模块内部。下面是实体说明的一般格式，其中的黑体字选项是 VHDL'93 语法的要求。

```
ENTITY 实体名 IS
        [GENERIC ( 类属表 );]
        [PORT ( 端口表 );]
        实体说明部分;
[ BEGIN
        实体语句部分 ;]
END [ ENTITY ] [实体名];
```

2.2.1 类属和端口说明

类属表和端口表是实体说明的头，它们说明用于设计实体和其外部环境通信的对象。

类属为设计实体和其外部环境通信的静态信息提供通道，特别是用来规定端口的大小、实体中子元件的数目、实体的定时特性等等。

GENERIC ([CONSTANT] 名字表: [IN] 子类型标识 [:= 静态表达式],…);

端口为设计实体和其外部环境的动态通信提供通道，每个端口必须有 1 个名字、1 个通信模式和 1 个数据类型。名字是该端口的标识符，模式说明数据通过该端口的流动方向，类