



“十二五”国家重点图书出版规划项目

中国科学技术大学 *精品* 教材



李 辉 / 编著

VHDL 与数字系统设计

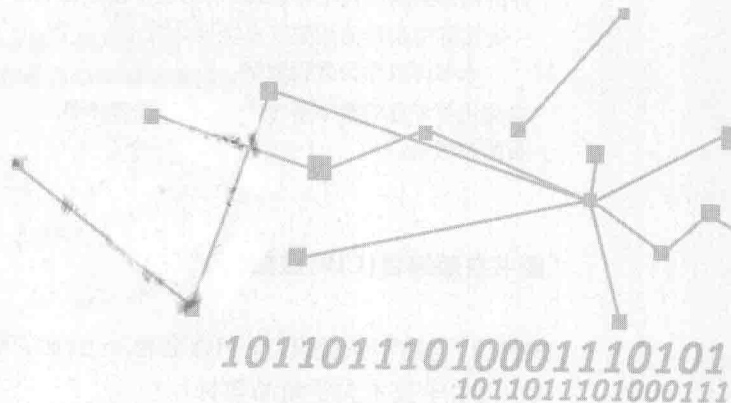
VHDL and Digital System Design

中国科学技术大学出版社



“十二五”国家重点图书出版规划项目

中国科学技术大学 **精品** 教材



李 辉 / 编著

VHDL and Digital System Design

VHDL 与数字系统设计

中国科学技术大学出版社

内 容 简 介

本书根据课堂教学的需要,以提高在校学生的实际数字系统设计能力为目标,介绍了采用硬件描述语言 VHDL 进行组合电路与时序电路设计的方法和可编程逻辑器件的内部结构及其工作原理、采用硬件描述语言 VHDL 和可编程逻辑器件实现具有一定实际应用价值的数字系统设计方法和实例。

本书可以作为开设数字系统设计实验和设计课程的大专院校电子工程、计算机和自动化等专业的教学参考书,也可以作为从事电子产品开发和生产的电子工程技术人员的参考书。

图书在版编目(CIP)数据

VHDL 与数字系统设计/李辉编著. —合肥:中国科学技术大学出版社,2015. 2
(中国科学技术大学精品教材)

“十二五”国家重点图书出版规划项目

ISBN 978-7-312-03693-4

I. V… II. 李… III. ①VHDL 语言—程序设计 ②数字系统—系统设计
IV. ①TP312 ②TP271

中国版本图书馆 CIP 数据核字(2015)第 024050 号

中国科学技术大学出版社出版发行

安徽省合肥市金寨路 96 号,230026

<http://press.ustc.edu.cn>

安徽省瑞隆印务有限公司

全国新华书店经销

开本:710 mm×960 mm 1/16 印张:18 插页:2 字数:342 千

2015 年 2 月第 1 版 2015 年 2 月第 1 次印刷

印数:1—3000 册

定价:39.00 元



编审委员会

主 任 侯建国

副主任 窦贤康 陈初升

张淑林 朱长飞

委 员 (按姓氏笔画排序)

方兆本 史济怀 古继宝 伍小平

刘 斌 刘万东 朱长飞 孙立广

汤书昆 向守平 李曙光 苏 淳

陆夕云 杨金龙 张淑林 陈发来

陈华平 陈初升 陈国良 陈晓非

周学海 胡化凯 胡友秋 俞书勤

侯建国 施蕴渝 郭光灿 郭庆祥

奚宏生 钱逸泰 徐善驾 盛六四

龚兴龙 程福臻 蒋 一 窦贤康

褚家如 滕脉坤 霍剑青

总 序

2008年,为庆祝中国科学技术大学建校五十周年,反映建校以来的办学理念和特色,集中展示教材建设的成果,学校决定组织编写出版代表中国科学技术大学教学水平的精品教材系列。在各方的共同努力下,共组织选题281种,经过多轮严格的评审,最后确定50种入选精品教材系列。

五十周年校庆精品教材系列于2008年9月纪念建校五十周年之际陆续出版,共出书50种,在学生、教师、校友以及高校同行中引起了很好的反响,并整体进入国家新闻出版总署的“十一五”国家重点图书出版规划。为继续鼓励教师积极开展教学研究与教学建设,结合自己的教学与科研积累编写高水平的教材,学校决定,将精品教材出版作为常规工作,以《中国科学技术大学精品教材》系列的形式长期出版,并设立专项基金给予支持。国家新闻出版总署也将该精品教材系列继续列入“十二五”国家重点图书出版规划。

1958年学校成立之时,教员大部分来自中国科学院的各个研究所。作为各个研究所的科研人员,他们到学校后保持了教学的同时又作研究的传统。同时,根据“全院办校,所系结合”的原则,科学院各个研究所在科研第一线工作的杰出科学家也参与学校的教学,为本科生授课,将最新的科研成果融入到教学中。虽然现在外界环境和内在条件都发生了很大变化,但学校以教学为主、教学与科研相结合的方针没有变。正因为坚持了科学与技术相结合、理论与实践相结合、教学与科研相结合的方针,并形成了优良的传统,才培养出了一批又一批高质量的人才。

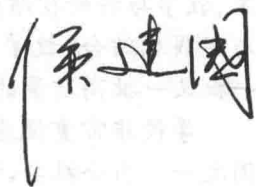
学校非常重视基础课和专业基础课教学的传统,这也是她特别成功的原因之一。当今社会,科技发展突飞猛进、科技成果日新月异,没有扎实的基础知识,很难在科学技术研究中作出重大贡献。建校之初,华罗庚、吴有训、严济慈等老一辈科学家、教育家就身体力行,亲自为本科生讲授基础课。他们以渊博的学识、精湛的讲课艺术、高尚的师德,带出一批又一批杰出的年轻教员,培

养了一届又一届优秀学生。入选精品教材系列的绝大部分是基础课或专业基础课的教材,其作者大多直接或间接受到过这些老一辈科学家、教育家的教诲和影响,因此在教材中也贯穿着这些先辈的教育教学理念与科学探索精神。

改革开放之初,学校最先选派青年骨干教师赴西方国家交流、学习,他们在带回先进科学技术的同时,也把西方先进的教育理念、教学方法、教学内容等带回到中国科学技术大学,并以极大的热情进行教学实践,使“科学与技术相结合、理论与实践相结合、教学与科研相结合”的方针得到进一步深化,取得了非常好的效果,培养的学生得到全社会的认可。这些教学改革影响深远,直到今天仍然受到学生的欢迎,并辐射到其他高校。在入选的精品教材中,这种理念与尝试也都有充分的体现。

中国科学技术大学自建校以来就形成的又一传统是根据学生的特点,用创新的精神编写教材。进入我校学习的都是基础扎实、学业优秀、求知欲强、勇于探索和追求的学生,针对他们的具体情况编写教材,才能更加有利于培养他们的创新精神。教师们坚持教学与科研的结合,根据自己的科研体会,借鉴目前国外相关专业有关课程的经验,注意理论与实际应用的结合,基础知识与最新发展的结合,课堂教学与课外实践的结合,精心组织材料、认真编写教材,使学生在掌握扎实的理论基础的同时,了解最新的研究方法,掌握实际应用的技术。

入选的这些精品教材,既是教学一线教师长期教学积累的成果,也是学校教学传统的体现,反映了中国科学技术大学的教学理念、教学特色和教学改革成果。希望该精品教材系列的出版,能对我们继续探索科教紧密结合培养拔尖创新人才,进一步提高教育教学质量有所帮助,为高等教育事业作出我们的贡献。



中国科学技术大学校长
中国科学院院士
第三世界科学院院士

前 言

随着计算机和大规模集成电路制造技术的迅速发展,现代的电子产品和复杂数字逻辑系统正朝着高集成度、小型化和低功耗的方向发展。传统的依赖电路原理图的设计方法已经不能够满足现代复杂数字系统设计的要求。

在现代电子系统设计中,采用硬件描述语言 VHDL 设计硬件电路相比采用传统的电路原理图设计硬件电路效率更高,设计的模块与使用哪一个公司生产的器件无关,设计不会因为芯片的工艺和结构的变化而变化,从而使已经设计成功的模块可以重复使用,可移植性好,提高了系统设计的效率。

目前可编程逻辑器件的功能,配合日益完善的电子设计自动化工具,可以反复修改,成为电路设计者首选的电子元件之一。

现场可编程或在系统可编程技术是指用户为了修改逻辑设计或重构数字系统,在已经设计和制作好的电路板上,直接对现场可编程或在系统可编程逻辑器件进行在线编程和反复修改,并进行现场调试和验证,使原来不容易改变的硬件设计变得像软件一样灵活而易于修改和调试。

本书共分 5 章介绍 VHDL、可编程逻辑器件和应用实例。第 1 章介绍 VHDL 的基本结构与描述语句;第 2 章介绍基本逻辑单元电路的设计;第 3 章介绍采用 VHDL 进行仿真程序的编写;第 4 章介绍常用的可编程逻辑器件的特点;第 5 章介绍实现具有一定应用价值的数字系统的设计实例,这些设计实例已经得到验证和通过。

本书在编写过程中得到了很多在校本科生和研究生的帮助,作者在此表示衷心的感谢。由于作者水平有限和时间短促,书中难免有一些错误,恳请各位专家批评指正。读者如果对本书有任何意见和建议,请发电子邮件到邮箱 hli@ustc.edu.cn 告诉作者。

作 者

2014 年于中国科学技术大学

目 次

总序	(i)
前言	(iii)
第 1 章 VHDL 的基本结构与描述语句	(1)
1.1 VHDL 的基本结构	(4)
1.2 VHDL 并发描述语句	(9)
1.3 VHDL 的运算操作符	(25)
1.4 标示符、数据对象、数据类型、属性和保留关键字	(26)
1.5 顺序描述语句	(42)
1.6 库、程序包	(56)
习题 1	(59)
第 2 章 基本逻辑单元电路的设计	(61)
2.1 组合电路的设计	(61)
2.2 时序电路的设计	(68)
习题 2	(107)
第 3 章 仿真	(108)
3.1 激励信号的产生	(108)
3.2 十进制计数器的仿真	(110)
3.3 仿真与综合	(114)
习题 3	(117)
第 4 章 可编程的逻辑器件	(118)
4.1 可编程的逻辑器件概述	(118)

第 1 章 VHDL 的基本结构与描述语句

随着现代电子系统设计的规模日益增大和复杂程度日益增高,大家熟悉的通过绘制逻辑电路图来设计复杂数字逻辑系统的方法已经远远不能满足现代复杂电子系统设计的要求,需要采用更抽象的层次描述和自顶向下的设计方法。

为了大大降低设计难度,越来越广泛地采用硬件描述语言设计方法。硬件描述语言是可以描述硬件电路的功能、信号连接关系和定时关系的语言,采用硬件描述语言设计数字系统是硬件设计领域的一次重大变革,与采用电路原理图设计硬件电路的方法相比,大大提高了设计效率。

目前,在进行数字系统设计时,广泛采用的硬件描述语言有 VHDL 和 Verilog, VHDL 的英文全名为 VHSIC Hardware Description Language,而 VHSIC 则是 Very High Speed Integrated Circuit 的缩写,所以 VHDL 的中文名为超高速集成电路的硬件描述语言。硬件描述语言 VHDL 是美国国防部在 20 世纪 80 年代初研究 VHSIC 计划时组织开发的,1987 年由 IEEE 标准化委员会确定为标准硬件设计语言,大部分电子设计自动化工具都支持 VHDL 语言。1993 年进一步修订,定为 IEEE 1076—1993 标准。IEEE 还制定了与 VHDL 语言相关的标准逻辑系统程序包 IEEE.STD_LOGIC_1164。

VHDL 描述能力很强,支持硬件的设计、验证、综合和测试,此外,还支持硬件设计数据的交换、维护、修改和硬件的实现。它可以描述抽象的系统级,也可以描述具体的逻辑级。许多计算机辅助设计公司新开发的电子设计软件都支持 VHDL 语言。

VHDL 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来模拟该系统。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 语言描述是分层次的,描述既可抽象(高层的行为描述),也可具体(底层的结构描述),其中比较重要的层次有系统级(System Level)、算法级(Algorithm Level)、寄存器传输级(RTL, Register Transfer Level)、逻辑级(Logic Level)、

采用寄存器硬件一一对应的直接描述,要么采用寄存器之间的功能描述。寄存器传输级描述表示行为(功能描述,也可称为 RTL 的行为描述),也隐含表示结构(与硬件对应的直接描述)。

结构描述是抽象模块相互连接的网表,是一种在多层次的设计中,高层次的设计模块调用低层次的设计模块,或者直接用门电路设计单元构成一个复杂的逻辑电路的描述方法。例如,在寄存器传输级,抽象模块是 ALU、多路选择器、寄存器等。

综上所述,采用 VHDL 设计数字逻辑系统的主要优点为:

(1) 具有多层次描述硬件功能的能力,支持自顶向下的设计方法,是系统设计领域中使用最多的硬件描述语言之一;支持大规模设计的分解和设计的共享。

(2) 设计可以重复使用。采用 VHDL 语言的设计与工艺技术无关,不会因为工艺的更新而过时。

(3) VHDL 语言标准、规范,是最早定为 IEEE 标准的硬件描述语言,使用广泛,绝大多数的 EDA 工具都支持 VHDL 语言,这为 VHDL 语言的进一步推广和应用创造了一个良好的环境。

然而,使用现场可编程门阵列(FPGA, Field Programmable Gate Array)或复杂可编程逻辑器件(CPLD, Complex Programmable Logic Device)来实现逻辑功能时,所采用的硬件描述语言是同软件语言(如 C, C++ 等)有本质区别的!在采用硬件描述语言设计硬件电路时,虽然硬件描述语言使用了计算机程序语言的形式,但是硬件描述语言描述是硬件的抽象,它的最终实现结果是芯片内部的硬件电路,而不是一条一条指令的执行过程。所以评判一段用硬件描述语言编写的代码的最终标准应该是其描述并实现的硬件电路的正确性及其性能(包括面积和速度两个方面)。对于初学者,特别是从事软件工程的初学者来说,采用硬件描述语言设计数字逻辑系统,有时会与所描述的具体硬件电路结构相脱节,片面追求代码的整洁、简短,无法达到设计的指标。

实际设计过程中,设计者主要是根据硬件描述语言 VHDL 的语法规则对系统目标的逻辑行为进行描述,然后通过综合工具进行电路结构的综合、优化,通过仿真工具进行逻辑功能仿真和系统时延仿真。由于每个设计者对语言规则、电路行为的理解程度不同,每个人的编程风格不同,往往同样的系统功能,描述的方式是不一样的,综合出来的电路结构更是大相径庭。因此,即使最后综合出的电路都能实现相同的逻辑功能,但是其电路的复杂程度和时延特性会有很大的差别,甚至某些臃肿的电路还会产生难以预料的问题。正确的编程方法是,首先要对该部分硬件的结构与连接十分了解和清晰,然后用成熟、合适的硬件描述语言表达

出来。

在使用硬件描述语言进行数字逻辑系统设计时,只会使用到 VHDL 所有语句中的一部分可以综合的语句。例如,使用 FPGA/CPLD 完成数字逻辑功能时,FPGA/CPLD 的制造商提供的开发工具就不支持一些 VHDL 的语句,如不支持“after”等语句。而且,FPGA/CPLD 的不同制造商提供的开发工具支持的硬件描述语言的语句都可能不一样,有时还会出现用硬件描述语言设计好的逻辑电路在有些开发工具中可以完成和实现所设计的逻辑功能,但是同样的设计在另外一些开发工具中就不能完成的情况,这是由于 FPGA/CPLD 的不同制造商提供的开发工具支持的硬件描述语言的语句不一样造成的,当出现这样的问题时,设计者要阅读使用的开发工具是否支持设计中使用到的语句。

目前,已经出版了许多非常好的中文版或英文版的关于 VHDL 方面的书籍,由于本书是以数字逻辑系统设计为目的的,所以本书没有全面介绍 VHDL 所有语句,只是介绍了那些在设计数字逻辑系统中经常使用到的一些语句。

由于 VHDL 或 Verilog 硬件描述语言不是为了逻辑系统的设计而设计的,而是为了仿真和测试设计的,所以对一个已经用 VHDL 设计好的逻辑电路,在编写仿真代码时,仿真工具会支持 VHDL 的所有语句。使用 VHDL 进行设计时,出现无法实现很多语句的现象也就不奇怪了。

1.1 VHDL 的基本结构

VHDL 语言描述的对象称为实体(Entity),可以将一个复杂的系统抽象成一个实体,可以描述像 CPU 那样复杂的电路,也可以描述一块芯片或一个门电路。

在设计时,采用自顶向下的层次化设计方法,各层的设计模块,都可以作为实体。高层次的设计实体可以调用低层次的设计实体。

VHDL 语言设计的基本单元,就是 VHDL 语言的一个基本设计实体(Design Entity)。一个基本设计实体由实体说明(Entity Declaration)和结构体(Architecture Body)两个部分组成。

可以把实体说明看作一个黑盒子,知道黑盒子的输入和输出,但不知道黑盒子里面的内容。黑盒子里面的内容由结构体来描述。例如,要描述一个逻辑元件 A,如图 1.1 所示。

结构体包含多条并发描述语句,但它包含的并发描述语句不允许对同一个信号进行多次赋值,语句执行的顺序与并发描述语句出现的先后次序无关,与纯软件程序不一样,不是按顺序一条一条指令执行的过程。

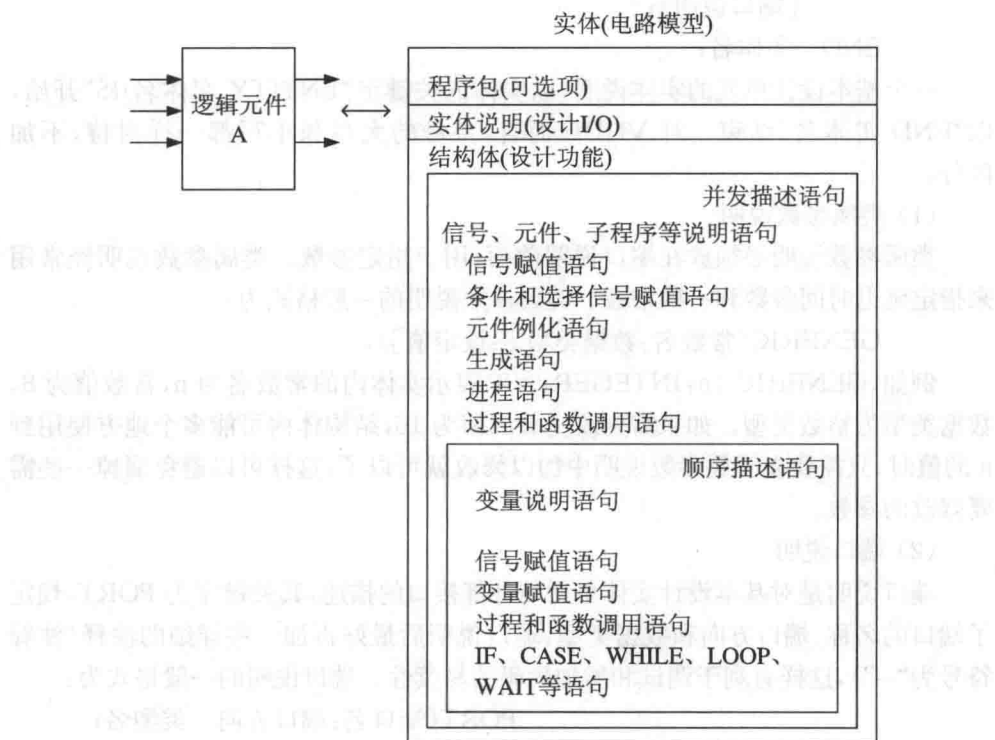


图 1.1 VHDL 的一个基本设计实体

顺序描述语句只能够出现在进程或子程序中,在进程或子程序中的顺序描述语句允许对同一个信号进行多次赋值,顺序描述语句像一般的高级语言一样,按语句出现的先后次序执行。

实体说明部分规定了基本设计单元的输入/输出端口,其作用相当于逻辑符号的引脚,信号通过端口在实体之间流入或流出。

实体说明部分定义了设计单元的输入/输出端口,而结构体部分定义了设计单元具体的逻辑功能。

1. 实体说明

实体说明类似于电路原理图中的符号,不描述模块的具体逻辑功能,主要描述该实体与外部电路的接口信号,确定了实体的每一个输入/输出信号的端口名称、

信号的方向和类型。实体说明的一般格式为：

```
ENTITY 实体名 IS
    [类属参数说明];
    [端口说明];
END 实体名;
```

一个基本设计单元的实体说明，以实体的关键字“ENTITY 实体名 IS”开始，以“END 实体名”结束。对 VHDL 而言，字母的大写和小写都一样对待，不加区分。

(1) 类属参数说明

类属参数说明必须放在端口说明前面，用于指定参数。类属参数说明经常用来指定延迟时间参数和一些常数。类属参数说明的一般格式为：

```
GENERIC(常数名:数据类型[:设定值]);
```

例如，GENERIC (n:INTEGER:=8)表示实体内的常数名为 n，常数值为 8，数据类型为整数类型。如果需要改变 n 的值为 16，结构体内可能多个地方使用到 n 的值时，只需要在类属参数说明中加以修改就可以了，这样可以避免漏掉一些需要修改的参数。

(2) 端口说明

端口说明是对基本设计实体单元与外部接口的描述，其关键字为 PORT，规定了端口的名称、端口方向和数据类型，端口说明后最好再加一些详细的注释(注释符号为“--”)，这样有利于调试和增加代码的易读性。端口说明的一般格式为：

```
PORT(端口名:端口方向 类型名;
```

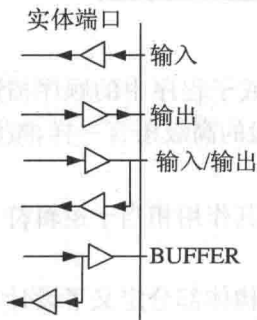


图 1.2 端口方向

```
...
    端口名:端口方向 类型名);
```

其中各端口名在实体中不能够重复。

端口方向的说明有以下几种模式：

- IN 输入
- OUT 输出
- INOUT 双向
- BUFFER 输出(结构体内部可以再使用)

当端口方向确定后，就会综合成如图 1.2 所示的电路原理图。

如果将端口信号定义成输入，在实体的结构体内，不能对该端口信号赋值，该

端口信号不能出现在赋值符号的左边。

如果将端口信号定义成输出,在实体的结构体内,不能将该端口信号传递给其他信号,该端口信号不能出现在赋值符号的右边。

如果将端口信号定义成输入/输出,在实体的结构体内,该端口信号可以作为输出信号出现在赋值符号的左边,也可以作为输入信号出现在赋值符号的右边。

如果将端口信号定义成 BUFFER,在实体的结构体内,该端口信号可以作为输出信号出现在赋值符号的左边,也可以出现在赋值符号的右边,此时该端口信号得到的是该实体输出信号的反馈信号,而不是来自该实体外部的其他输入信号。

VHDL 语言有 10 种数据类型,在逻辑电路设计中经常用到 2 种:位逻辑数据类型 BIT 和位向量类型 BIT_VECTOR。

当信号定义成 BIT 类型时,该信号的电平只能取为逻辑“0”或“1”,而没有高阻态,为了扩大信号的逻辑状态,一般可以用包含了多值逻辑的 STD_LOGIC 类型说明,因为 STD_LOGIC 类型不但包含了高阻态,还包含了弱“1”、弱“0”和不确定等多种状态。但是,在使用 STD_LOGIC 或 STD_LOGIC_VECTOR 时,应该在实体说明以前,增加两条语句:

```
LIBRARY IEEE; --IEEE 库
USE IEEE. STD_LOGIC_1164. ALL; --调用其中 STD_LOGIC_1164 程序包中所有(.ALL)的内容
```

在对 VHDL 语言程序编译时,从指定的程序包中,寻找数据类型的定义。其作用像 C 语言的 include< * .h>一样。STD_LOGIC_1164 是一个常用的程序包,它定义了一些常用的数据类型和函数。

例 1.1 实体的外部端口如图 1.3 所示。

图中信号 d[15:0]为 16 位输入总线,clk,reset,oe 为输入信号,q1[15:0]为 16 位三态输出总线,q2[15:0]为 16 位双向总线,opt 为输出信号。

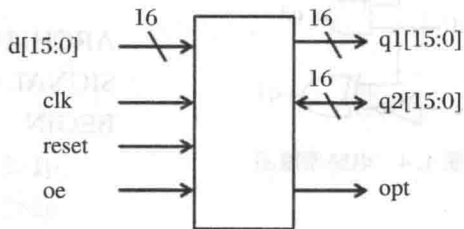


图 1.3 实体的外部端口

```
LIBRARY IEEE;
USE IEEE. STD_LOGIC_1164. ALL;

ENTITY example IS
PORT (d: IN STD_LOGIC_VECTOR (15 DOWNT0 0);
```



```

clk, reset, oe: IN STD_LOGIC;
q1: OUT STD_LOGIC_VECTOR (15 DOWNTO 0);
q2: INOUT STD_LOGIC_VECTOR (15 DOWNTO 0);
opt: OUT STD_LOGIC);

```

```
END example;
```

2. 结构体

结构体定义了设计单元具体的功能，一定要跟在实体的后面。

结构体可以分为两个部分：说明部分和描述体部分。说明部分在 ARCHITECTURE 与 BEGIN 之间，定义需要在结构体内部出现的信号。对具体电路的描述出现在结构体内的 BEGIN 与 END 之间。

一个结构体的具体结构描述如下：

```
ARCHITECTURE 结构体名 OF 实体名 IS
```

```
[定义语句]内部信号,常数,数据类型等的定义;
```

```
BEGIN
```

```
[并发语句];
```

```
END 结构体名;
```

例 1.2 编写描述如图 1.4 所示的程序。

```

ENTITY example IS
    PORT (a, b, c, d, e: IN STD_LOGIC;
          q1, q2, q3, q4: OUT STD_LOGIC);
END example;

ARCHITECTURE arch OF example IS
    SIGNAL tmp: STD_LOGIC;
BEGIN
    q1<= a AND b;
    q2<= b OR c;
    tmp<= NOT e;
    q4<= d XOR tmp;
    q3<= c NAND d;
END arch;

```

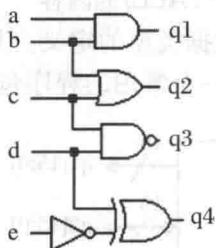


图 1.4 电路原理图

其中小于等于符号“<=”是信号赋值符号，表示将符号“<=”右边的值赋给左边的信号。