

应用型本科 电子及通信工程专业“十三五”规划教材

EDA技术实践教程

主 编 顾涵

副主编 夏金威 潘启勇 张惠国 张静亚 徐健

- 内容新颖：新知识、新技术、新工艺
- 特色鲜明：突出“应用、实践、创新”
- 定位准确：面向工程技术型人才培养
- 质量上乘：应用型本科专家全力打造



西安电子科技大学出版社
<http://www.xduph.com>

应用型本科 电子及通信工程专业“十三五”规划教材

EDA 技术实践教程

主 编 顾 涵

副主编 夏金威 潘启勇 张惠国

张静亚 徐 健

西安电子科技大学出版社

内 容 简 介

本书坚持“抓基础、重设计、培养创新实践能力”的宗旨,全面介绍 EDA 技术实践课程的主要内容。第一篇介绍了 VHDL 基本语法和结构,使读者掌握采用 VHDL 编程实现硬件描述功能的基本方法;第二篇介绍了电子线路 CAD 与仿真技术;第三篇介绍了 PCB 制板技术;第四篇介绍了 EDA 技术实践应用。全书内容由浅入深、取材新颖、内容丰富、实用为主、重点突出。与传统实践教程不同,本书给出了具体的设计思路、参考程序及所需的硬件知识,便于没有学习过 EDA 技术课程的学生快速入门。

本书可作为高等院校电气信息类专业的实验、课程设计指导用书,也可作为没有学习过 EDA 技术课程学生的开放性实践教材,还可作为大学生电子设计竞赛的入门培训教材。

图书在版编目(CIP)数据

EDA 技术实践教程 / 顾涵主编. —西安:西安电子科技大学出版社,2017.5

应用型本科 电子及通信工程专业“十三五”规划教材

ISBN 978-7-5606-4472-1

I. ① E… II. ① 顾… III. ① 电子电路—电路设计—计算机辅助设计—教材

IV. ① TN702.2

中国版本图书馆 CIP 数据核字(2017)第 075577 号

策 划 高 樱

责任编辑 祝婷婷 阎彬

出版发行 西安电子科技大学出版社(西安市太白南路2号)

电 话 (029) 88242885 88201467

邮 编 710071

网 址 www.xduph.com

电子邮箱 xdupfxb001@163.com

经 销 新华书店

印刷单位 陕西天意印务有限责任公司

版 次 2017年5月第1版

2017年5月第1次印刷

开 本 787毫米×1092毫米

1/16 印 张 18.5

字 数 435千字

印 数 1~3000册

定 价 42.00元

ISBN 978-7-5606-4472-1 / TN

XDUP 4764001-1

如有印装问题可调换

应用型本科 电子及通信工程专业规划教材

编审专家委员名单

主 任: 沈卫康(南京工程学院 通信工程学院 院长/教授)

副主任: 张士兵(南通大学 电子信息学院 副院长/教授)

陈 岚(上海应用技术学院 电气与电子工程学院 副院长/教授)

宋依青(常州工学院 计算机科学与工程学院 副院长/教授)

张明新(常熟理工学院 计算机科学与工程学院 副院长/教授)

成 员: (按姓氏拼音排列)

鲍 蓉(徐州工程学院 信电工程学院 院长/教授)

陈美君(金陵科技学院 网络与通信工程学院 副院长/副教授)

高 尚(江苏科技大学 计算机科学与工程学院 副院长/教授)

李文举(上海应用技术学院 计算机科学学院 副院长/教授)

梁 军(三江学院 电子信息工程学院 副院长/副教授)

潘启勇(常熟理工学院 物理与电子工程学院 副院长/副教授)

任建平(苏州科技学院 电子与信息工程学院 副院长/教授)

孙霓刚(常州大学 信息科学与工程学院 副院长/副教授)

谭 敏(合肥学院 电子信息与电气工程系 系主任/教授)

王杰华(南通大学 计算机科学与技术学院 副院长/副教授)

王章权(浙江树人大学 信息科技学院 副院长/副教授)

温宏愿(南京理工大学泰州科技学院 电子电气工程学院 副院长)

严云洋(淮阴工学院 计算机工程学院 院长/教授)

杨会成(安徽工程大学 电气工程学院 副院长/教授)

杨俊杰(上海电力学院 电子与信息工程学院 副院长/教授)

郁汉琪(南京工程学院 创新学院 院长/教授)

于继明(金陵科技学院 智能科学与控制工程学院 副院长/副教授)

前 言

随着电子技术、EDA 技术的快速发展,功能强大、开发周期短、便于修改及开发工具智能化的可编程逻辑器件已被广泛应用在各个领域。可编程逻辑器件的开发与应用已成为电子信息、计算机类各专业的必修课,同时相关知识也成为电子设计工程师的必备知识。多年的实践证明,全面使用 EDA 工具是电子设计技术的发展趋势,因为利用 EDA 工具可以帮助设计者完成电子系统设计中的大部分工作。

在使用 VHDL 硬件描述语言和 EDA 技术进行电路设计时,遵守的是自顶向下的设计流程。设计过程中可以充分地采用前人已经设计、验证过的模块,实现前人设计成果的复用。采用这样的设计方法设计电子系统,大大缩短了新产品的的设计周期,降低了设计成本,又快又好地满足了市场的需求,因此电子工程师与电子、计算机、通信、微电子等专业的学生都应该掌握这些方法和技术。

全书共 7 章。第 1 章介绍 VHDL 程序基本结构;第 2 章介绍 VHDL 基础;第 3 章介绍电路仿真技术;第 4 章介绍电子线路 CAD 技术;第 5 章介绍 PCB 制作流程及制作工艺;第 6 章介绍 EDA 技术基础实验;第 7 章介绍典型应用系统设计。

本书由常熟理工学院顾涵担任主编,夏金威、潘启勇、张惠国、张静亚、徐健担任副主编。其中顾涵编写了第一篇和第四篇,夏金威、潘启勇、张惠国、张静亚、徐健编写了第二篇和第三篇。全书由顾涵负责组织、统稿工作。

本书所列实验、课程设计项目的功能实现并不局限于某一种型号的开发板,开发板上只要有相应的接口都可以实现。VHDL 程序是共享的,因此本书没有列出全部引脚分配及下载过程。本书在编写过程中参考了大量资料,部分资料来源于互联网,无法一一列出,在此向所有作者深表谢意。

限于编者水平,书中难免有不足之处,恳请各位老师和读者不吝指正。

编 者
2017 年 3 月

目 录

第一篇 VHDL 基本语法和结构

第 1 章 VHDL 程序基本结构	3	2.1.5 VHDL 的属性语句	30
1.1 初识 VHDL 程序	3	2.2 VHDL 的顺序语句	32
1.2 VHDL 程序基本结构	5	2.2.1 赋值语句	33
1.2.1 VHDL 程序一般结构	5	2.2.2 转向控制语句	35
1.2.2 库、程序包	5	2.2.3 WAIT 语句	48
1.2.3 实体	9	2.2.4 子程序调用语句	51
1.2.4 结构体	11	2.2.5 返回语句(RETURN)	52
1.2.5 配置	11	2.2.6 断言语句(ASSERT)	53
1.3 结构体描述方式	12	2.2.7 REPORT 语句	53
1.3.1 行为描述方式	12	2.2.8 NULL 语句	54
1.3.2 数据流描述方式	12	2.3 VHDL 的并行语句	54
1.3.3 结构描述方式	13	2.3.1 并行信号赋值语句	55
本章小结	14	2.3.2 进程语句	57
习题	15	2.3.3 块语句(BLOCK)	61
		2.3.4 并行过程调用语句	63
第 2 章 VHDL 基础	16	2.3.5 元件例化语句	67
2.1 VHDL 的语言要素	16	2.3.6 生成语句(GENERATE)	71
2.1.1 VHDL 文字规则	16	本章小结	77
2.1.2 VHDL 数据对象	17	习题	77
2.1.3 VHDL 数据类型与转换	21		
2.1.4 VHDL 操作符	27		

第二篇 电子线路 CAD 与仿真技术

第 3 章 电路仿真技术	83	3.4.1 菜单栏简介	104
3.1 Multisim 软件使用简介	83	3.4.2 工具栏简介	108
3.2 Multisim 软件界面及通用环境变量	83	3.5 Multisim 软件的实际应用	109
3.3 Multisim 软件常用元件库分类	85	3.6 利用 Multisim 软件进行电阻、	
3.4 Multisim 软件界面菜单工具栏介绍	104	电容、电感的电原理性分析	114

3.6.1 电阻的分压、限流电阻演示.....	114	4.2.7 网络与网络标签.....	137
3.6.2 电容的隔直流、通交流特性的 演示和验证	116	4.2.8 生成 PCB 网表	137
3.6.3 电感的隔交流、通直流的特性 演示与验证	119	4.3 PCB 文件的设计.....	139
3.6.4 二极管特性的演示与验证.....	120	4.3.1 PCB 的相关概念	139
3.6.5 三极管特性的演示与验证.....	122	4.3.2 PCB 设计的流程和原则	140
4.3.3 PCB 编辑环境	141	4.3.4 PCB 文件的创建	143
4.3.5 PCB 设计环境的设置	149	4.3.6 原理图信息的导入.....	153
4.3.7 元件的放置及封装的修改	154	4.3.8 布线	158
4.3.9 PCB 设计的检查	161	4.3.10 PCB 图的打印及文件输出	161
4.4 Protel DXP 库的建立与元件制作	164		
4.4.1 创建原理图库.....	164		
4.4.2 创建 PCB 元件库	168		
4.4.3 元件封装库的管理.....	173		

第 4 章 电子线路 CAD 技术 124

4.1 Protel DXP 软件平台介绍..... 124

4.1.1 Protel DXP 概述..... 124

4.1.2 Protel DXP 界面..... 124

4.2 Protel DXP 电路原理图的绘制..... 127

4.2.1 电路原理图的绘制流程..... 127

4.2.2 新建工程设计项目..... 128

4.2.3 新建原理图文件 129 | | |

4.2.4 原理图图纸的设置..... 131

4.2.5 放置元件 131 | | |

4.2.6 连接电路 136 | | |

第三篇 PCB 制板技术

第 5 章 PCB 制作流程及制作工艺介绍.....	177	5.4.3 设备的使用及操作.....	201
5.1 PCB 制作流程	177	5.5 印刷线路油墨工艺介绍	203
5.2 钻孔工艺介绍	178	5.5.1 丝印的目的.....	203
5.2.1 工艺要求及注意事项.....	178	5.5.2 丝印工艺.....	203
5.2.2 HW 系列双面线路板制作机的 工作及操作过程	178	5.6 显影工艺介绍	205
5.3 沉铜工艺介绍	187	5.6.1 工艺要求及注意事项.....	205
5.3.1 电镀前处理(沉铜)工艺介绍	187	5.6.2 工艺原理及常见问题.....	205
5.3.2 沉铜工艺各溶液介绍.....	189	5.6.3 设备的使用及操作.....	207
5.3.3 设备的使用及操作.....	197	5.7 蚀刻工艺介绍	209
5.4 孔金属化(电镀)工艺介绍.....	199	5.7.1 工艺原理、操作规范及 常见问题.....	209
5.4.1 工艺要求及注意事项.....	199	5.7.2 设备的使用及操作.....	212
5.4.2 工艺原理及操作要求.....	200		

第四篇 EDA 技术实践应用

第 6 章 EDA 技术基础实验	217	7.3 交通灯信号控制器的设计	251
实验一 ISE 设计环境熟悉(1).....	217	7.3.1 设计要求.....	251
实验二 ISE 设计环境熟悉(2).....	220	7.3.2 设计实现.....	251
实验三 EDA 实验硬件熟悉.....	221	7.4 空调系统有限状态自动机的设计.....	253
实验四 寄存器电路设计仿真与下载	222	7.4.1 设计要求.....	253
实验五 层次化设计仿真与下载	223	7.4.2 设计实现.....	253
实验六 触发器电路设计仿真与下载	224	7.5 智力竞赛抢答器的设计	254
实验七 简单电路的 VHDL 描述	225	7.5.1 设计要求.....	254
实验八 七人表决器的设计	227	7.5.2 设计实现.....	255
实验九 数字秒表的设计	229	7.6 软核处理器 Picoblaze 的原理及应用 ...	259
实验十 Vivado 操作入门.....	231	7.6.1 Picoblaze 架构介绍	259
第 7 章 典型应用系统设计.....	244	7.6.2 Picoblaze 指令设置	262
7.1 多功能信号发生器的设计	244	7.6.3 Picoblaze 文件结构	272
7.1.1 设计要求	244	7.6.4 Picoblaze 汇编基础	272
7.1.2 设计实现	244	7.6.5 Picoblaze 汇编程序开发	278
7.2 序列检测器的设计	249	本章小结	283
7.2.1 设计要求	249	习题.....	284
7.2.2 设计实现	250	参考文献	286

第一篇 VHDL 基本语法和结构



第 1 章 VHDL 程序基本结构

1.1 初识 VHDL 程序

当使用一个集成芯片时，根据数字电子技术的知识，至少需要了解三个方面的信息：① 该芯片符合什么规范，是谁生产的，是否被大家认可；② 该芯片有多少管脚，每个管脚是输入还是输出，每个管脚对输入/输出有什么要求；③ 该芯片各管脚之间的关系，以及能完成什么逻辑功能。

相应地，当使用 VHDL 设计一个硬件电路时，至少需要描述三个方面的信息：① 使用的设计规范，亦即此设计符合哪个设计规范才能得到大家的认可，这就是库、程序包使用说明；② 所设计的硬件电路与外界的接口信号，这就是设计实体的说明；③ 所设计的硬件电路其内部各组成部分的逻辑关系及整个系统的逻辑功能，这就是该设计实体对应的结构体说明。

1. 设计思路

根据数字电子技术的知识，可以知道，74LS00 是一个 4—2 线输入与非门，亦即该芯片由 4 个 2 输入与非门组成，因此设计时可先设计一个 2 输入与非门(见图 1.1(a))，再由 4 个 2 输入与非门构成一个整体——MY74LS00(见图 1.1(b))。

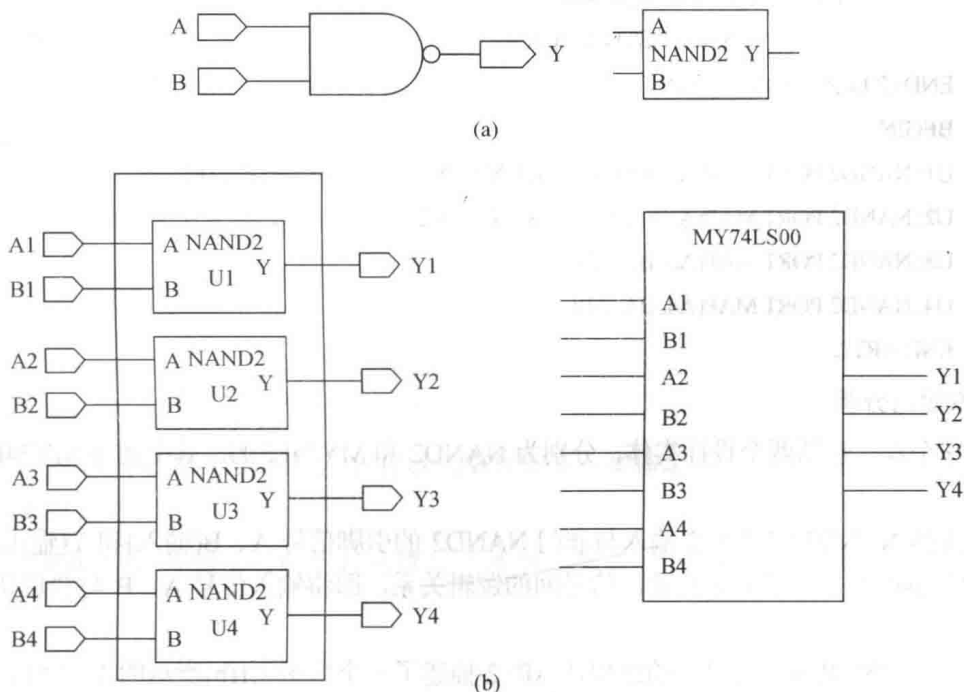


图 1.1 MY74LS00 的结构示意图



2. VHDL 源程序

(1) 2 输入与非门 NAND2 的逻辑描述, 程序如下:

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;           -- IEEE 库及其中程序包的使用说明
ENTITY NAND2 IS
    PORT(A, B: IN STD_LOGIC;
          Y: OUT STD_LOGIC);
END NAND2;                               -- 实体 NAND2 的说明
ARCHITECTURE ART1 OF NAND2 IS           -- 结构体 ART1 的说明
    BEGIN
    Y <= A NAND B;
END ART1;
```

(2) MY74LS00 的逻辑描述, 程序如下:

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;           -- IEEE 库及其中程序包的使用说明
ENTITY MY74LS00 IS
    PORT(A1, B1, A2, B2, A3, B3, A4, B4: IN STD_LOGIC;
          Y1, Y2, Y3, Y4: OUT STD_LOGIC);
END MY74LS00;
ARCHITECTURE ART2 OF MY74LS00 IS       -- 结构体 ART2 的说明
    COMPONENT NAND2 IS                 -- 元件调用声明
        PORT(A, B: IN STD_LOGIC;
              Y: OUT STD_LOGIC);
    END COMPONENT NAND2;
    BEGIN
    U1: NAND2 PORT MAP(A => A1, B => B1, Y => Y1);   -- 元件连接说明
    U2: NAND2 PORT MAP(A => A2, B => B2, Y => Y2);
    U3: NAND2 PORT MAP(A3, B3, Y3);
    U4: NAND2 PORT MAP(A4, B4, Y4);
END ART2;
```

3. 说明与分析

(1) 整个设计包括两个设计实体, 分别为 NAND2 和 MY74LS00, 其中实体 MY74LS00 为顶层实体。

(2) 实体 NAND2 定义了 2 输入与非门 NAND2 的引脚信号 A、B(输入)和 Y(输出), 其对应的结构体 ART1 描述了输入与输出信号间的逻辑关系, 即将输入信号 A、B 与非后传给输出信号端 Y。

(3) 实体 MY74LS00 及对应的结构体 ART2 描述了一个如图 1.1(b)所示的 4—2 线输入与非门。由其结构体的描述可以看到, 它是由 4 个 2 输入与非门构成的。



(4) 在 MY74LS00 接口逻辑 VHDL 描述中, 根据图 1.1(b) 右侧的 MY74LS00 原理图可知, 实体 MY74LS00 定义了引脚的端口信号属性和数据类型。

(5) 在结构体 ART2 中, COMPONENT-END COMPONENT 语句结构对所调用的 NAND2 元件作了声明。

(6) 实体 MY74LS00 引导的逻辑描述也是由三个主要部分(即库和程序包使用说明、实体说明和结构体)构成的。

1.2 VHDL 程序基本结构

1.2.1 VHDL 程序一般结构

一般地, 一个完整的 VHDL 源代码通常包括库(LIBRARY)、程序包(PACKAGE)、实体(ENTITY)、结构体(ARCHITECTURE)和配置(CONFIGURATION)5 个部分。

一个相对完整的 VHDL 程序(或称为设计实体)比较固定的结构如图 1.2 所示, 其至少应该包括 3 个基本组成部分: 库、程序包使用说明, 实体和结构体。

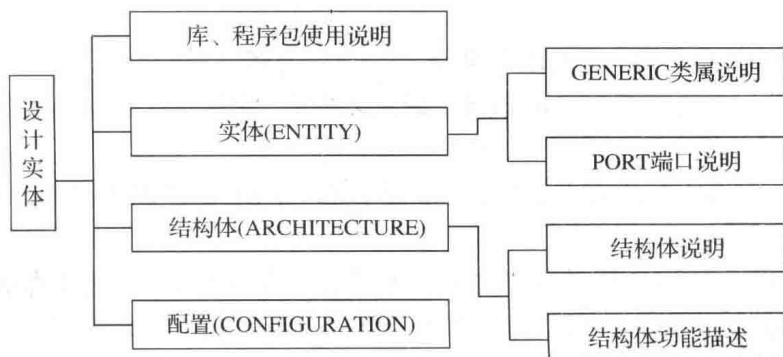


图 1.2 VHDL 程序设计基本结构

本书主要讨论 VHDL 设计的基本组成模块: 实体说明和结构体。

1.2.2 库、程序包

1. 库

1) 库的定义

库是经编译后的数据的集合, 它存放包集合定义、实体定义、结构体定义和配置定义等。

2) 库的使用

在 VHDL 中, 库的说明语句总是放在实体单元前面, 而且库语句一般必须与 USE 语句同时使用。

库的语句格式如下:

LIBRARY 库名;

这一语句相当于为其后的设计实体打开了以此库名所命名的库, 以便设计实体可以利用其中的



程序包。

例如：

```
LIBRARY IEEE;           --打开 IEEE 库
```

USE 语句指明库中的程序包。USE 语句的使用有两种常用格式：

```
USE 库名.程序包名.项目名;
```

```
USE 库名.程序包名.ALL;
```

第一语句格式的作用是，向本设计实体开放指定库中的特定程序包内所选定的项目；第二语句格式的作用是，向本设计实体开放指定库中的特定程序包内所有的内容。

库语句一般必须与 USE 语句同时使用，一旦说明了库和程序包，整个设计实体都可进入访问或调用。

例如：

```
LIBRARY IEEE; --打开 IEEE 库
```

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

```
--打开 IEEE 库中的 STD_LOGIC_1164 程序包的所有内容
```

```
USE IEEE. STD_LOGIC_UNSIGNED.ALL;
```

```
--打开 IEEE 库中的 STD_LOGIC_UNSIGNED 程序包的所有内容
```

3) 库的分类

VHDL 程序设计中常用的库有以下四种：

(1) IEEE 库。IEEE 库是 VHDL 设计中最常见的库，它包含有 IEEE 标准的程序包和其他一些支持工业标准的程序包。

(2) STD 库。VHDL 标准定义了两个标准程序包，即 STANDARD 和 TEXTIO 程序包，它们都收入在 STD 库中。

(3) WORK 库。WORK 库是用户的 VHDL 设计的现行工作库，用于存放用户设计和定义的一些设计单元和程序包，因此自动满足 VHDL 标准，在实际调用中，不必预先说明。

(4) VITAL 库。VITAL 库是 FPGA/CPLD 生产厂商提供的面向 ASIC 的逻辑门库。使用 VITAL 库，可以提高 VHDL 门级时序模拟的精度，因而只在 VHDL 仿真器中使用。

(5) 用户自定义的库。

2. 程序包

为了使已定义的常数、数据类型、元件调用说明及子程序能被更多的 VHDL 设计实体方便地访问和共享，可以将它们收集在一个 VHDL 程序包中。多个程序包可以并入一个 VHDL 库中，使之适用于更一般的访问和调用范围。这一点对于大系统开发，多个或多组开发人员并行工作显得尤为重要。

1) 预定义程序包

常用的预定义程序包有以下四种：

(1) STD_LOGIC_1164 程序包。它是 IEEE 库中最常用的程序包，是 IEEE 的标准程序包。其中包含了一些数据类型、子类型和函数的定义，这些定义将 VHDL 扩展为一个能描述多值逻辑(除具有“0”和“1”以外还有其他的逻辑，如高阻态“Z”、不定态“X”等)的硬件描述语言，很好地满足了实际数字系统的设计需求。



(2) STD_LOGIC_ARITH 程序包。它预先编译在 IEEE 库中，是 Synopsys 公司的程序包。此程序包在 STD_LOGIC_1164 程序包的基础上扩展了三个数据类型，即 UNSIGNED、SIGNED 和 SMALL_INT，并为其定义了相关算术运算符和转换函数。

(3) STD_LOGIC_UNSIGNED 和 STD_LOGIC_SIGNED 程序包。这两个程序包都是 Synopsys 公司的程序包，都预先编译在 IEEE 库中。这些程序包重载了可用于 INTEGER 型及 STD_LOGIC 和 STD_LOGIC_VECTOR 型混合运算的运算符，并定义了一个由 STD_LOGIC_VECTOR 型到 INTEGER 型的转换函数。

(4) STANDARD 和 TEXTIO 程序包。这两个程序包是 STD 库中的预编译程序包。STANDARD 程序包中定义了许多基本的数据类型、子类型和函数等。

程序包的具体内容介绍如下。

常数说明：主要用于预定义系统的宽度，如数据总线通道的宽度。

数据类型说明：主要用于说明在整个设计中通用的数据类型，例如通用的地址总线、数据类型的定义等。

元件定义：主要规定在 VHDL 设计中参与元件例化的文件(已完成的设计实体)对外的接口界面。

子程序说明：用于说明在设计中任一处可调用的子程序。

2) 自定义程序包

自定义程序包的一般语句结构如下：

```
--程序包首
PACKAGE 程序包名 IS           --程序包首开始
    程序包首说明部分;
END[PACKAGE][程序包名];      --程序包首结束
--程序包体
PACKAGE BODY 程序包名 IS      --程序包体开始
    程序包体说明部分以及包体内容;
END[PACKAGE BODY][程序包名]; --程序包体结束
```

3) 程序包首

程序包首的说明部分可收集多个不同的 VHDL 设计所需的公共信息，其中包括数据类型说明、信号说明、子程序说明及元件说明等。

程序包结构中，程序包体并非是必需的，程序包首可以独立定义和使用。

程序包首的主要定义程序如下：

```
PACKAGE PAC1 IS                -- 程序包首开始
    TYPE BYTE IS RANGE 0 TO 255; -- 定义数据类型 BYTE
    SUBTYPE BYTE1 IS BYTE RANGE 0 TO 15; -- 定义子类型 BYTE1
    CONSTANT C1: BYTE := 255;    -- 定义常数 C1
    SIGNAL S1: BYTE1;             -- 定义信号 S1
    COMPONENT BYTE_ADDER IS      -- 定义元件
        PORT(A, B: IN BYTE;
```



```

    C: OUT BYTE;
    OVERFLOW: OUT BOOLEAN);
END COMPONENT BYTE_ADDER;
FUNCTION MY_FUNCTION(A: IN BYTE)RETURN BYTE;
--定义函数
END PACKAGE PAC1;
--程序包首结束

```

下面是在现行 WORK 库中定义程序包并立即使用的示例。

```

PACKAGE SEVEN IS
--定义程序包
SUBTYPE SEGMENTS IS BIT_VECTOR(0 TO 6);
TYPE BCD IS RANGE 0 TO 9;
END PACKAGE SEVEN;
USE WORK.SEVEN.ALL;
--打开程序包，以便后面使用
ENTITY DECODER IS
PORT(SR: IN BCD;
      SC: OUT SEGMENTS);
END ENTITY DECODER;
ARCHITECTURE ART OF DECODER IS
BEGIN
WITH SR SELECT
SC <= B"1111110" WHEN 0,
      B"0110000" WHEN 1,
      B"1101101" WHEN 2,
      B"1111001" WHEN 3,
      B"0110011" WHEN 4,
      B"1011011" WHEN 5,
      B"1011111" WHEN 6,
      B"1110000" WHEN 7,
      B"1111111" WHEN 8,
      B"1111011" WHEN 9,
      B"0000000" WHEN OTHERS,
END ARCHITECTURE ART;

```

4) 程序包体

程序包体用于定义在程序包首已定义的子程序的子程序体。程序包体说明部分的组成可以是 USE 语句(允许对其他程序包的调用)、子程序定义、子程序体、数据类型说明、子类型说明和常数说明等。没有子程序说明的程序包体可以省去。

程序包常用来封装属于多个设计单元分享的信息，程序包定义的信号、变量不能在设计实体之间共享。



1.2.3 实体

设计实体是 VHDL 设计中的基本单元，可以描述完整系统、电路板、芯片、逻辑单元或门电路。它不仅可以描述像微处理器那样的复杂电路，也能描述像门电路那样简单的电路，体现了 VHDL 描述的灵活性。

不管是复杂的设计实体，还是简单的设计实体，一个设计实体总是由两部分组成的：实体和结构体。实体说明主要描述的是一个设计的外貌，即输入/输出接口及一些用于结构体的参数定义；结构体则描述设计的行为和结构，指定输入/输出之间的行为。

下面以一个 2 选 1 电路原理图或 VHDL 描述为例分别加以说明，如图 1.3 所示。

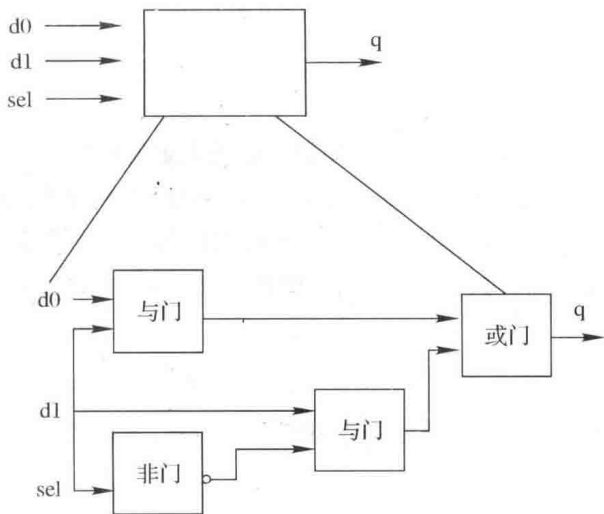


图 1.3 2 选 1 电路原理图

VHDL 源程序如下：

```
LIBRARY IEEE;                                --库
USE IEEE.STD_LOGIC_1164.ALL;
ENTITY ch0 IS
PORT(d0: IN STD_LOGIC;
      d1: IN STD_LOGIC;
      sel: IN STD_LOGIC;
      q: OUT STD_LOGIC);
END ch0;
ARCHITECTURE CONNECT OF ch0 IS
BEGIN
PROCESS(d0, d1, sel)
VARIABLE temp1, temp2, temp3: STD_LOGIC;    --结构体
BEGIN
TEMP1 := d0 AND sel;
TEMP2 := d1 AND (NOT sel);
```