



21 世纪高等院校电气工程与自动化规划教材

Circuit Design with VHDL

VHDL 数字电路 设计教程

苗丽华 主编

潘莹 周雪妍 副主编



人民邮电出版社
POSTS & TELECOM PRESS



21 世纪高等院校电气工程与自动化规划教材

21 century institutions of higher learning materials of Electrical Engineering and Automation Planning

Circuit Design with VHDL

VHDL 数字电路 设计教程

苗丽华 主编

潘莹 周雪妍 副主编

人民邮电出版社

北京

图书在版编目 (CIP) 数据

VHDL数字电路设计教程 / 苗丽华主编. — 北京 :
人民邮电出版社, 2012. 11
21世纪高等院校电气工程与自动化规划教材
ISBN 978-7-115-27214-0

I. ①V… II. ①苗… III. ①
VHDL语言—程序设计—高等学校—教材②数字电路—电路设计—高等学校—教材 IV. ①TP301.2②TN79

中国版本图书馆CIP数据核字(2012)第236783号

内 容 提 要

本书主要讲述硬件描述语言 VHDL 及其在数字电路设计中的建模和仿真, 内容包括 VHDL 语言学习基础、语法规则和模型结构、数字电路的 VHDL 建模、VHDL 硬件描述等。本书在内容结构的组织上有独特之处, 如将并行描述语句、顺序描述语句、数据类型与运算操作符和属性等独立成章, 使读者更容易清晰准确地掌握这些重要内容。本书注重设计实践, 给出了大量完整设计实例的电路图、相关基本概念、电路工作原理以及仿真结果, 从而将 VHDL 语法学习和如何采用它进行电路设计有机地结合在一起, 并通过深入浅出的操作点拨, 提升学生对 VHDL 的理性认识, 提高学生的研究能力。

本书可作为高等院校电子工程、通信工程、自动化、电气工程等专业的教材及实验指导, 也可作为相关专业技术人员的自学参考书。

21 世纪高等院校电气工程与自动化规划教材

VHDL 数字电路设计教程

-
- ◆ 主 编 苗丽华
副 主 编 潘 莹 周雪妍
责任编辑 李海涛
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子邮件 j15@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京艺辉印刷有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 11.5 2012 年 11 月第 1 版
字数: 283 千字 2012 年 11 月北京第 1 次印刷

ISBN 978-7-115-27214-0

定价: 28.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223
反盗版热线: (010)67171154

随着 VLSI 的发展,电子产品市场运作节奏的进一步加快,涉及诸多领域(如计算机应用、通信、智能仪表、医用设备、军事、民用电器等领域)的现代电子设计技术已迈入一个全新的阶段,其特点如下。

(1) 电子器件及其技术的发展将更多地趋向于支持 EDA (Electronic Design Automation)。

(2) 借助于硬件描述语言,硬件设计与软件设计技术得到了有机的融合。

(3) 就电子设计的技术、开发过程和目标器件的形式与结构来说,将从过去的“群雄并起”的局面向规范化、标准化发展。

(4) 应用系统的设计已从单纯的 ASIC 设计走向了系统设计和单片系统 (System On a Chip, SOC) 设计。

一些专家预言,未来的电子技术将是 EDA 的时代。为了适应这一趋势,国外各大 VLSI 厂商纷纷推出各种系列的大规模和超大规模 FPGA 和 CPLD 产品。其产品性能提高之快,品种之多让人应接不暇。Xilinx 公司和 Altera 公司推出了多种高性能的 FPGA/CPLD 芯片, FPGA 器件的规模已进入了千万门的行列。

作为世界最早发明 GAL 等可编程逻辑器件的 Lattice 公司,在已成熟的 PLD/CPLD 产品的基础上,首次开创了可编程逻辑器件革命性的编程下载方式,即系统可编程下载方式。并相继推出了多种系列各具特色的 ISP 下载方式的 CPLD 器件以及大规模系统可编程模拟器件。最近几年,编程逻辑器件的开发生产和销售规模以惊人的速度增长,统计资料表明,其平均年增长率高达 23%。与此相适应, CadenceData I/O Mentor Graphics OrCAD Synopsys 和 Viewlogic 等世界各大 EDA 公司亦相继推出各类高性能的 EDA 工具软件。在现代电子设计技术高速推进浪潮的多层因素促进下, CAD、CAM、CAT 和 CAE 技术发生了进一步融合与升华,形成了更为强大的 EDA 和 ESDA (Electronic System Design Automation) 技术,从而成为当代电子设计技术发展的总趋势。

在 21 世纪,电子技术的发展将更加迅猛,电子设计的自动化程度将更高,电子产品的上市节奏将更快。传统的电子设计技术、工具和器件将在更大程度上被 EDA 所取代。因此,EDA 技术和 VHDL 势必成为广大电子信息工程类各专业领域工程技术人员的必修课。

与一般的高级语言相比, VHDL 的学习具有更强的实践性,它的学习和应用所涉及的内容和工具比较多,类似传统软件编程语言的语法语句和编程练习的学习已不足以掌握

VHDL。本书从实际应用出发，以实用和可操作为基点，以初步掌握 EDA 技术和培养基于 VHDL 的开发能力为目标，始终围绕一个主题——学以致用。

本书共分 12 章，第 1 章绪论，介绍了 VHDL 有关的基础概念；第 2 章 VHDL 语法基础，介绍了结构体、数据对象和进程；第 3 章数据类型，介绍了基本数据类型、自定义数据类型和类型间的转换；第 4 章 VHDL 操作符及属性，介绍了 VHDL 操作符、属性和一些设计实例；第 5 章 VHDL 中的顺序语句，介绍了各种语句的用法；第 6 章 VHDL 中的并行语句，介绍了各种并行语句；第 7 章 VHDL 子程序和元件例化，介绍了子程序、函数和过程等；第 8 章组合逻辑电路设计，介绍了门电路、编码器、译码器和运算器的设计；第 9 章时序逻辑电路的设计，介绍了触发器、寄存器、计数器和分频器的设计；第 10 章有限状态，介绍了状态机的建模、状态编码和有限状态机的设计实例；第 11 章 VHDL 在通信和 DSP 中的应用，介绍了分频器、循环冗余校验和快速加法器；第 12 章 SAP-1 CPU 电路设计，介绍了整个 CPU 设计的过程及实现。

参加本书编写和程序调试的有苗丽华、潘莹和周雪妍。第 1 章至第 5 章、第 9 章由苗丽华编写，第 6 章至第 8 章由周雪妍编写，第 10 章至第 12 章由潘莹编写。其中苗丽华任主编，潘莹、周雪妍任副主编。本书由苗丽华、潘莹统稿并主审。我们的学生王月、江弘扬、战瑜、赵斐、别燕娜、刘婉、孟平平、董洁冰、郭晓婧、李盛旗、张冶等在资料收集、插图描绘、文字校对等方面做了大量工作，在此表示衷心的感谢。

由于编者水平有限，书中错误之处在所难免，期待读者能对书中的错误给予指正。

编 者

2012 年 6 月

目 录

第 1 章 绪论	1	2.4 VHDL 数据对象	11
1.1 VHDL 的发展	1	2.4.1 信号	11
1.2 VHDL 的特点	1	2.4.2 变量	11
1.3 VHDL 设计流程	2	2.4.3 常量	11
1.3.1 接受设计任务	3	2.4.4 TO 和 DOWNTO 关键字	12
1.3.2 确定功能要求	3	第 3 章 数据类型	13
1.3.3 进行设计	3	3.1 基本数据类型	13
1.3.4 源代码模拟	3	3.1.1 整型数据 (INTEGER)	13
1.3.5 综合、优化和布局布线	3	3.1.2 实型数据 (REAL & FLOATING)	13
1.3.6 布局布线后的设计模拟	3	3.1.3 位和位向量 (BIT & BIT_VECTOR)	14
1.3.7 器件编程	3	3.1.4 布尔类型数据 (BOOLEAN)	14
1.4 VHDL 综合工具	4	3.1.5 字符与字符串 (CHARACTER & STRING)	14
1.4.1 Quartus II 综合工具	4	3.1.6 物理类型 (PHYSICAL)	14
1.4.2 ISE 综合工具	6	3.1.7 自然数和正整数 (NATURAL & POSITIVE)	15
1.4.3 ModelSim 仿真工具	6	3.2 自定义数据类型	15
1.5 CPLD 和 FPGA 概述	6	3.2.1 枚举类型	16
1.5.1 GAL 元件	6	3.2.2 整数类型	16
1.5.2 CPLD	6	3.2.3 实数类型和浮点类型	16
1.5.3 FPGA	7	3.2.4 数组类型	16
第 2 章 VHDL 语法基础	8	3.2.5 存取类型	17
2.1 VHDL 描述结构	8	3.2.6 文件类型	17
2.1.1 库描述	8	3.2.7 记录类型 (RECORD)	18
2.1.2 实体	8	3.2.8 时间类型 (TIME)	18
2.1.3 结构体	8	3.3 数据类型转换	19
2.2 结构体的描述	9		
2.3 进程	10		
2.3.1 进程语句的格式	10		
2.3.2 进程的敏感表	10		

3.3.1 使用转换函数.....	19	5.2.5 超时等待的处理.....	41
3.3.2 使用类型标记法转换数据 类型	20	5.3 IF 语句	41
第 4 章 VHDL 操作符及属性	21	5.4 CASE 语句.....	44
4.1 VHDL 操作符	21	5.5 LOOP 语句	46
4.1.1 赋值操作符.....	21	5.5.1 FOR LOOP 循环.....	47
4.1.2 逻辑操作符.....	21	5.5.2 WHILE LOOP 循环.....	47
4.1.3 算术操作符.....	22	5.6 NEXT 语句	48
4.1.4 关系操作符.....	23	5.7 EXIT 语句.....	49
4.1.5 移位操作符.....	24	5.8 RETURN 语句.....	49
4.1.6 连接操作符.....	24	5.9 NULL 语句.....	50
4.1.7 操作符重载.....	25	5.10 过程调用语句.....	51
4.2 VHDL 的属性	26	5.11 断言语句.....	52
4.2.1 预定义属性.....	26	5.12 REPORT 语句.....	53
4.2.2 值类型属性.....	26	第 6 章 VHDL 中的并行语句	56
4.2.3 函数类型属性.....	27	6.1 进程语句.....	56
4.2.4 信号类型属性.....	30	6.2 块语句.....	58
4.2.5 TYPE 类型属性.....	31	6.3 并行信号代入语句.....	60
4.2.6 RANGE 类型属性.....	31	6.3.1 并发信号代入语句	61
4.2.7 用户自定义属性.....	32	6.3.2 条件信号代入语句	62
4.3 GENERIC 参数传递	32	6.3.3 选择信号代入语句	63
4.4 VHDL 设计实例	33	6.4 并行断言语句.....	64
4.4.1 十六进制 7 段译码器.....	33	6.5 并行过程调用语句.....	65
4.4.2 16 位乘 16 位的乘法器.....	34	6.6 参数传递语句.....	66
4.4.3 波形发生器.....	35	6.7 元件例化语句.....	67
第 5 章 VHDL 中的顺序语句	38	6.8 生成语句.....	69
5.1 信号赋值语句和变量赋值语句.....	38	6.8.1 FOR-GENERATE 模式的生成 语句	69
5.2 WAIT 语句	39	6.8.2 IF-GENERATE 模式	71
5.2.1 WAIT ON.....	39	第 7 章 VHDL 子程序和元件例化	74
5.2.2 WAIT UNTIL.....	40	7.1 程序包.....	74
5.2.3 WAIT FOR.....	40	7.1.1 程序包的说明	75
5.2.4 多条件等待语句.....	41	7.1.2 程序包体的描述	75

7.1.3 程序包的使用.....	76	8.4.3 求补器.....	107
7.2 层次化建模和元件例化.....	76	8.5 三态门.....	107
7.2.1 层次化建模.....	77	第9章 时序逻辑电路的设计	109
7.2.2 元件声明.....	77	9.1 触发器.....	109
7.2.3 端口映射.....	78	9.1.1 D触发器.....	109
7.2.4 GENERIC映射.....	79	9.1.2 JK触发器.....	110
7.3 函数和过程.....	80	9.1.3 T触发器.....	112
7.3.1 函数.....	80	9.1.4 锁存器.....	113
7.3.2 全局函数和局部函数.....	81	9.2 寄存器和移位寄存器.....	114
7.3.3 过程.....	81	9.2.1 寄存器.....	114
7.3.4 全局过程和局部过程.....	82	9.2.2 移位寄存器.....	115
7.3.5 子程序的重载.....	82	9.3 计数器.....	118
7.3.6 函数和过程的比较.....	83	9.3.1 同步计数器.....	119
7.4 元件配置和子程序应用实例.....	83	9.3.2 异步计数器.....	120
7.4.1 元件例化实例.....	83	9.4 分频器.....	122
7.4.2 函数应用实例.....	86	第10章 有限状态机	129
第8章 组合逻辑电路设计	88	10.1 有限状态机概述.....	129
8.1 门电路.....	88	10.2 有限状态机的建模.....	129
8.1.1 二输入与门电路.....	88	10.2.1 状态的处理.....	130
8.1.2 二输入或门电路.....	89	10.2.2 模型的构建.....	130
8.1.3 二输入与非门电路.....	90	10.3 状态编码.....	131
8.1.4 二输入或非门电路.....	92	10.3.1 顺序码.....	131
8.1.5 反相门电路.....	94	10.3.2 枚举类型的编码.....	131
8.1.6 二输入异或门电路.....	95	10.3.3 一位有效编码.....	131
8.1.7 三输入与非门电路.....	96	10.3.4 综合工具的设置.....	131
8.2 编码器和译码器.....	97	10.3.5 定义编码方式的语法格式.....	132
8.2.1 编码器.....	97	10.3.6 初始化有限状态机.....	132
8.2.2 译码器.....	101	10.4 有限状态机的设计实例.....	133
8.3 多路选择器.....	102	10.4.1 Moore有限状态机.....	133
8.4 运算器.....	105	10.4.2 Mealy有限状态机.....	135
8.4.1 比较器.....	105	10.4.3 交通信号灯.....	137
8.4.2 加法器.....	106	10.4.4 硬币兑换机.....	140

第 11 章 扩频通信技术	144	11.3.2 流水线加法器	156
11.1 分频器	144	第 12 章 SAP-1 CPU 电路设计	159
11.1.1 偶数分频器的设计	144	12.1 SAP-1 CPU 及计算机的组成	
11.1.2 奇数分频器的设计	145	结构简介	159
11.1.3 半整数分频器的设计	148	12.2 SAP-1 CPU 指令、寻址法、	
11.1.4 大数目分频器的设计	149	程序设计	161
11.2 循环冗余校验 (CRC)	151	12.3 16×8 的 ROM 设计与仿真	162
11.2.1 CRC 的基本原理	151	12.4 SAP-1 CPU 设计与仿真	164
11.2.2 CRC 的实现方法	152	12.4.1 SAP-1 指令周期	164
11.2.3 CRC 程序的实现	152	12.4.2 SAP-1 CPU 设计	165
11.3 快速加法器	153	12.4.3 CPU 设计的完整代码	172
11.3.1 并行加法器	154	参考文献	175

VHDL 语言是一种用于电路设计的高级语言，在 20 世纪 80 年代的后期出现。最初是由美国国防部开发出来供美军用来提高设计的可靠性和缩减开发周期的一种使用范围较小的设计语言。

VHDL 的中文含义是超高速集成电路硬件描述语言，主要应用在数字电路的设计中。目前，它在中国主要应用在 FPGA/CPLD/EPLD 的设计中。当然在一些实力较为雄厚的单位，它也被用来设计 ASIC。

VHDL 主要用于描述数字系统的结构、行为、功能和接口。除了含有许多具有硬件特征的语句外，VHDL 的语言形式、描述风格以及语法十分类似于一般的计算机高级语言。VHDL 的程序结构特点是将一项工程设计，或称设计实体（可以是一个元件，一个电路模块或一个系统）分成外部（或称可视部分）和内部（或称不可视部分），即涉及实体的内部功能和算法完成部分。在对一个设计实体定义了外部界面后，一旦其内部开发完成后，其他的设计就可以直接调用这个实体。这种将设计实体分成内外部分的概念是 VHDL 系统设计的基本点。

1.1 VHDL 的发展

20 世纪 70 年代末和 80 年代初，提出 VHSIC 计划；1981 年，提出了 VHDL 语言，电路文本化；1986 年，IEEE 成立 VHDL 语言标准化小组；1987 年，VHDL 语言被接纳为 IEEE 1076 标准；1993 年，该标准被修订为 IEEE 1164 标准；1996 年，IEEE 1076.3 称为 VHDL 的综合标准；1995 年，中国国家技术监督局制定《CAD 通用技术规范》推荐 VHDL 语言成为硬件描述语言的国家标准。

1.2 VHDL 的特点

1. 功能强大、设计灵活

VHDL 具有功能强大的语言结构，可以用简洁明确的源代码来描述复杂的逻辑控制。它具有多层次的设计描述功能，层层细化，最后可直接生成电路级描述。VHDL 支持同步电路、异步电路和随机电路的设计，这是其他硬件描述语言所不能比拟的。VHDL 还支持各种设计方法，既支持自底向上的设计，又支持自顶向下的设计；既支持模块化设计，又

支持层次化设计。

2. 支持广泛、易于修改

由于 VHDL 已经成为 IEEE 标准所规范的硬件描述语言，目前大多数 EDA 工具几乎都支持 VHDL，这为 VHDL 的进一步推广和广泛应用奠定了基础。在硬件电路设计过程中，主要的设计文件是用 VHDL 编写的源代码，因为 VHDL 易读和结构化，所以易于修改设计。

3. 强大的硬件描述能力

VHDL 具有多层次的设计描述功能，既可以描述系统级电路，又可以描述门级电路。而描述既可以采用行为描述、寄存器传输描述或结构描述，也可以采用三者混合的混合级描述。另外，VHDL 支持惯性延迟和传输延迟，还可以准确地建立硬件电路模型。VHDL 支持预定义和自定义的数据类型，给硬件描述带来较大的自由度，使设计人员能够方便地创建高层次的系统模型。

4. 独立于器件的设计，与工艺无关

设计人员用 VHDL 进行设计时，不需要首先考虑选择完成设计的器件，就可以集中精力进行设计的优化。当设计描述完成后，可以用多种不同的器件结构来实现其功能。

5. 具有很强的移植能力

VHDL 是一种标准化的硬件描述语言，同一个设计描述可以被不同的工具所支持，使得设计描述的移植成为可能。

6. 易于共享与复用

VHDL 采用基于库 (Library) 的设计方法，可以建立各种可再次利用的模块。这些模块可以预先设计或使用以前设计中的存档模块，将这些模块存放到库中，就可以在以后的设计中进行复用，进而使设计成果在设计人员之间进行交流和共享，减少硬件电路设计。

1.3 VHDL 设计流程

VHDL 是 IEEE 所确认的一种标准化硬件描述语言，它在设计描述的过程中有一定的设计流程可以遵循，一般来讲，VHDL 的设计流程（见图 1-1）主要包括以下几个步骤。

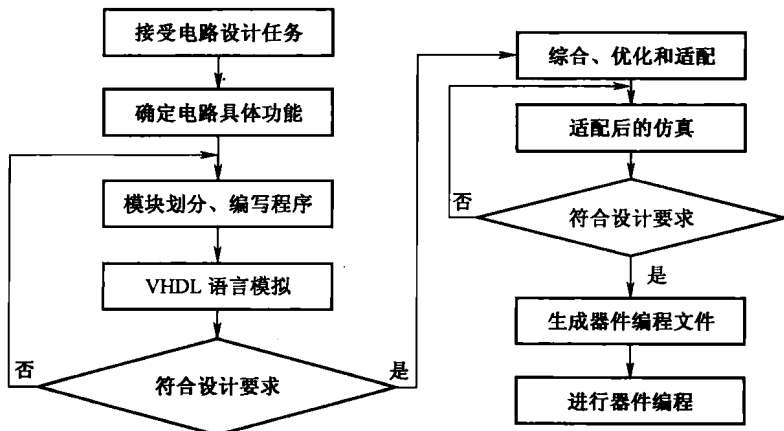


图 1-1 VHDL 设计流程示意图

1.3.1 接受设计任务

采用 VHDL 进行设计描述之前,设计人员首先要对电子系统的设计目的和设计要求有一个明确、清晰的认识,然后形成设计规范的定义,这一步骤对以后的设计来说是非常重要的,在此基础上接受设计任务才能保证设计的完成。

1.3.2 确定功能要求

采用 VHDL 进行设计描述主要包括设计规划和程序编写两个步骤。设计规划的主要任务是进行设计方式的选择以及确定是否进行模块划分。通常情况下,设计中采用的设计方式包括直接设计、自顶向下设计和自底向上的设计方法。

模块划分是设计过程中一个非常重要的步骤,模块划分的好坏将直接影响到最终的电路设计,因此设计人员在这一步应该花费一定的时间,从而保证模块划分的最优化。

1.3.3 进行设计

进行设计规划后,设计人员就可以按照模块划分来编写各个模块的 VHDL 程序,然后将各个模块的 VHDL 程序组合在一起,从而完成整个设计的 VHDL 描述。

1.3.4 源代码模拟

在设计流程中,综合、优化和适配(布局、布线)等后续操作往往需要花费大量的时间。一旦在后续工作中发现设计错误,设计人员往往需要先修改 VHDL 描述,再重新进行综合、优化和适配(布局、布线)等后续操作,如此反复,将会浪费大量的时间和人力、物力。因此,设计人员常常在完成编码后采用仿真器对 VHDL 设计描述进行仿真(有时称作前仿真),这样可以提早发现设计错误,节省时间,从而缩短开发周期。

1.3.5 综合、优化和布局布线

综合是指较高层次的抽象描述转化到低级别抽象的一种方法,简单来说,就是将设计的描述转化成底层电路进行表示。通常,综合的结果是一个网表或者是一组逻辑方程。

优化是指将设计的时延缩到最小和有效利用资源。几乎所有的高级 VHDL 综合工具都可以使用约束条件对设计进行优化。约束条件的设置主要包括时间的约束和面积的约束。

对设计描述进行综合和优化操作后,EDA 工具软件需要将综合和优化的逻辑结果分配到一个逻辑器件中。

1.3.6 布局布线后的设计模拟

与 VHDL 程序仿真不同,适配(布局、布线)后的仿真不仅要设计描述的逻辑功能进行验证,而且还要对设计描述的时序功能进行验证。如果时序不能满足,就需要回到前面的步骤重新进行操作。通常,适配(布局、布线)后的仿真被称为后仿真。

1.3.7 器件编程

器件编程就是将设计描述经过编译、综合、优化和适配(布局、布线)后的结果,经过一定的映射,转化成编程所需要的数据文件格式,然后通过烧片器或者下载电缆将数据下载

到器件中的过程。

1.4 VHDL 综合工具

1.4.1 Quartus II 综合工具

Altera 公司的 Quartus II 是一个完全集成化、易学易用的单芯片可编程系统 (SOPC) 设计平台, 它将设计、综合、布局和验证以及第三方 EDA 工具软件接口集成在一个无缝的环境中, 其界面友好, 使用便捷, 灵活高效, 深受设计人员的欢迎。下面简单介绍一下 Quartus II 工具的使用。

首先建立自己的工作文件夹, 用来存放所有的设计工程及文件。在工作文件夹中还可以再建立设计工程的文件夹, 不同的设计项目放在不同的文件夹中。

具体操作步骤如下。

(1) 新建工程, 出现图 1-2 后选择目录。

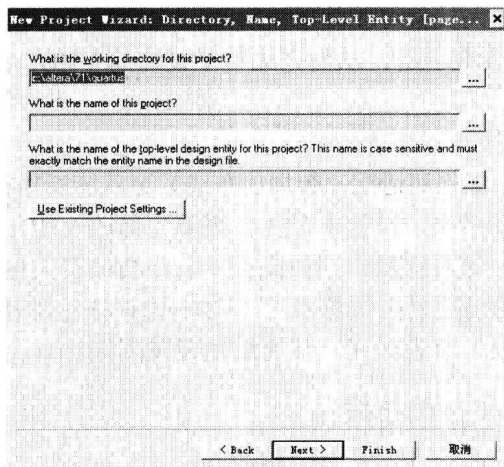


图 1-2 新建工程向导

(2) 添加文件, 如果没有已经创建好的文件直接单击 NEXT 按钮跳过, 如图 1-3 所示。

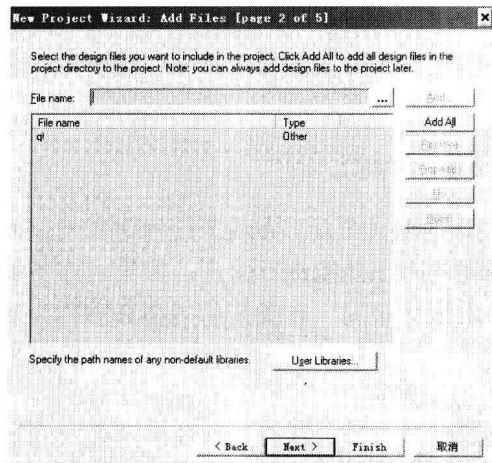


图 1-3 添加文件

(3) 在 Family 栏中选择目标芯片系列——Cyclone，然后选择此系列中的具体芯片型号为 EP1C6Q240C8，如图 1-4 所示。

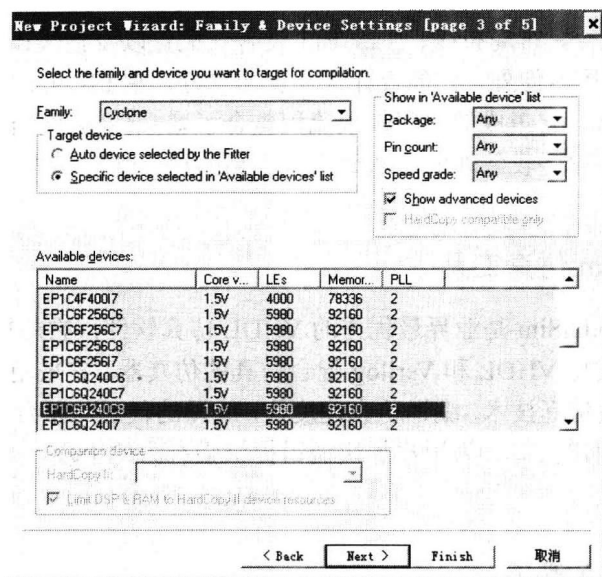


图 1-4 选择目标芯片

(4) 单击 Next 按钮进入 EDA 工具设置窗口（图略），勾选要用的第三方 EDA 工具。本次操作不采用第三方工具，因此单击 Next 按钮进入最后的总结窗口（图略）。在这个窗口中列出了所有前面设置的结果。如果有错误可以单击 Back 按钮返回逐一修改，否则单击 Finish 按钮结束。

(5) 新建 VHDL 设计文件如图 1-5 所示。

(6) 编辑文件，直接在空白处输入文件，如图 1-6 所示。

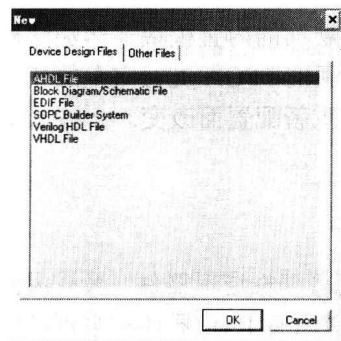


图 1-5 新建文件

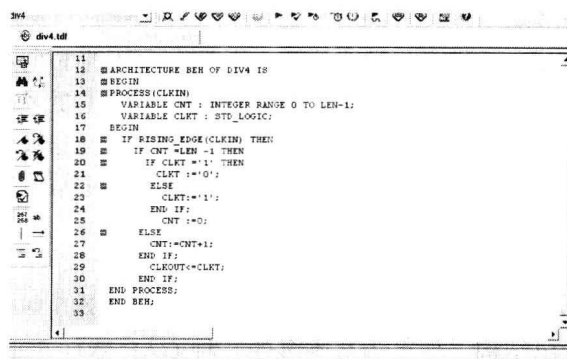


图 1-6 编辑文件

最后保存文件名为“div.vhd”。注意：确认文件保存在本工程文件夹下，实体名和 VHDL 设计文件名一致，而且在保存时要勾选“保存为”对话框中下方的“Add file to currentproject”选项。

1.4.2 ISE 综合工具

ISE 是使用 XILINX 的 FPGA 的必备设计工具，它可以完成 FPGA 开发的全部流程，包括设计输入、仿真、综合、布局布线、生成 BIT 文件、配置以及在线调试等，功能非常强大。ISE 除了功能完整、使用方便外，它的设计性能也非常好，拿 ISE 9.x 来说，其设计性能比其他解决方案平均快 30%，它集成的时序收敛流程整合了增强性物理综合优化，提供最佳的时钟布局、更好的封装和时序收敛映射，从而获得更高的设计性能。先进的综合和实现算法将动态功耗降低了 10%。

1.4.3 ModelSim 仿真工具

Mentor 公司的 ModelSim 是业界最优秀的 VHDL 仿真软件，它能提供友好的仿真环境，是业界唯一的单内核支持 VHDL 和 Verilog 混合仿真的仿真器。它采用直接优化的编译技术、Tcl/Tk 技术和单一内核仿真技术，编译仿真速度快，编译的代码与平台无关，便于保护 IP 核，个性化的图形界面和用户接口，为用户加快调错提供强有力的手段，是 FPGA/ASIC 设计的首选仿真软件。

1.5 CPLD 和 FPGA 概述

20 世纪 80 年代中期，FPGA 已经在 PLD 设备中扎根。CPLD 和 FPGA 包括了一些数量相对较大的可编辑逻辑单元。CPLD 逻辑门的密度在几千到几万个逻辑单元之间，而 FPGA 通常是在几万到几百万之间。

CPLD 和 FPGA 的主要区别是它们的系统结构。CPLD 是一个有点限制性的结构，这个结构由一个或者多个可编辑的结果之和的逻辑组列和一些相对少量的锁定的寄存器组成。这样的结果是缺乏编辑灵活性，但是却有可以预计延迟时间和逻辑单元对连接单元高比率的优点。而 FPGA 却是由很多的连接单元组成，这样虽然让它可以更加灵活地编辑，但是结构却复杂得多。

CPLD 和 FPGA 另外一个区别是大多数的 FPGA 含有高层次的内置模块（如加法器和乘法器）以及内置的记忆体。因此一个有关的重要区别是很多新的 FPGA 支持完全的或者部分的系统内重新配置。允许它们的设计随着系统升级或者动态重新配置而改变。一些 FPGA 可以让设备的一部分重新编辑而其他部分继续正常运行。

1.5.1 GAL 元件

GAL (Generic Array Logic) 是一种电擦除可重复编程的可编程逻辑器件，具有灵活的可编程输出结构，使得为数不多的几种 GAL 器件几乎能够代替所有 PAL 器件和数百种中小规模标准器件。而且，GAL 器件采用先进的 EECMOS 工艺，可以在几秒钟内完成对芯片的擦除和写入，并允许反复改写，为研制开发新的逻辑系统提供方便，因此，GAL 器件得到了广泛的应用。

1.5.2 CPLD

CPLD 主要是由可编程逻辑宏单元 (Macro Cell, MC) 围绕中心的可编程互连矩阵单元

组成。其中 MC 结构较复杂, 并具有复杂的 I/O 单元互连结构, 可由用户根据需要生成特定的电路结构, 完成一定的功能。由于 CPLD 内部采用固定长度的金属线进行各逻辑块的互连, 所以设计的逻辑电路具有时间可预测性, 避免了分段式互连结构时序不完全预测的缺点。

它具有编程灵活、集成度高、设计开发周期短、适用范围宽、开发工具先进、设计制造成本低、对设计者的硬件经验要求低、标准产品无需测试、保密性强、价格大众化等特点, 可实现较大规模的电路设计, 因此被广泛应用于产品的原型设计和产品生产(一般在 10 000 件以下)之中。几乎所有应用中小规模通用数字集成电路的场合均可应用 CPLD 器件。CPLD 器件已成为电子产品不可缺少的组成部分, 它的设计和应用成为电子工程师必备的一种技能。

1.5.3 FPGA

FPGA 一般来说比 ASIC(专用集成芯片)的速度要慢, 无法完成复杂的设计, 而且消耗更多的电能。但是它们也有很多优点, 如可以快速成品, 可以被修改来改正程序中的错误和更便宜的造价。厂商也可能会提供便宜的但是编辑能力差的 FPGA, 因为这些芯片有比较差的可编辑能力, 所以这些设计的开发是在普通的 FPGA 上完成的, 然后将设计转移到一个类似于 ASIC 的芯片上。

小 结

本章讲述了硬件描述语言, 特别是 VHDL 语言的产生、发展、特点及其开发流程, 以及对采用 VHDL 语言设计数字电路和数字系统的目标芯片——可编程逻辑器件做了简单介绍。

习 题

1. 什么是 EDA 技术?
2. 什么是硬件描述语言? 目前流行的硬件描述语言主要有几种?
3. 简述 VHDL 语言的历史、特点和发展趋势。
4. 采用 VHDL 语言开发数字电路的流程是什么?
5. 什么是 FPGA 和 CPLD?

第 2 章 VHDL 语法基础

VHDL 的语言要素是 VHDL 编程语句的基本元素，反映了 VHDL 重要的语言特性。准确无误地理解和掌握 VHDL 语言要素的基本含义和用法，并在编程中认真遵循，对于正确完成 VHDL 程序设计是十分重要的。

2.1 VHDL 描述结构

实体说明、结构体和库是每一个 VHDL 程序必不可少的三大部分，而配置说明和程序包则是选项，它们的取舍视具体情况而定。

2.1.1 库描述

库是一种 VHDL 设计单元的特定集合，用于存放可重复使用的元件和数据类型说明。它可由用户生成，也可由 ASIC 芯片制造商提供。程序包是属性选项，用于存放各模块均能共享的数据类型、常数、元件和子程序等，它通过编译后成为库。

2.1.2 实体

在 VHDL 中，一个设计单元被称为一个设计实体。它可以是一个简单的电路（如一个与门、一个译码器或加法器等），也可以是一个复杂的电路（如一个微处理器或一个系统），描述设计实体的外部接口信号。在 VHDL 中，任何一个设计实体都至少由实体说明和结构体说明两部分构成。

实体说明是对设计实体与外部电路的接口描述，它规定了设计单元的输入/输出接口信号或引脚，是设计实体对外的一个通信界面。

2.1.3 结构体

描述设计实体的内部结构和功能。结构体用于描述此设计实体的逻辑结构和逻辑功能。每个实体可以有多个结构体，每个结构体对应着实体不同的结构和算法实现方案；其间的各个结构体的地位是同等的，它们完整地实现了实体的行为，但同一结构体不能为不同的实体所拥有，而且结构体不能单独存在，它必须有一个界面说明，即一个实体。对于具有多个结构体的实体，必须用 CONFIGURATION 配置语句进行说明。