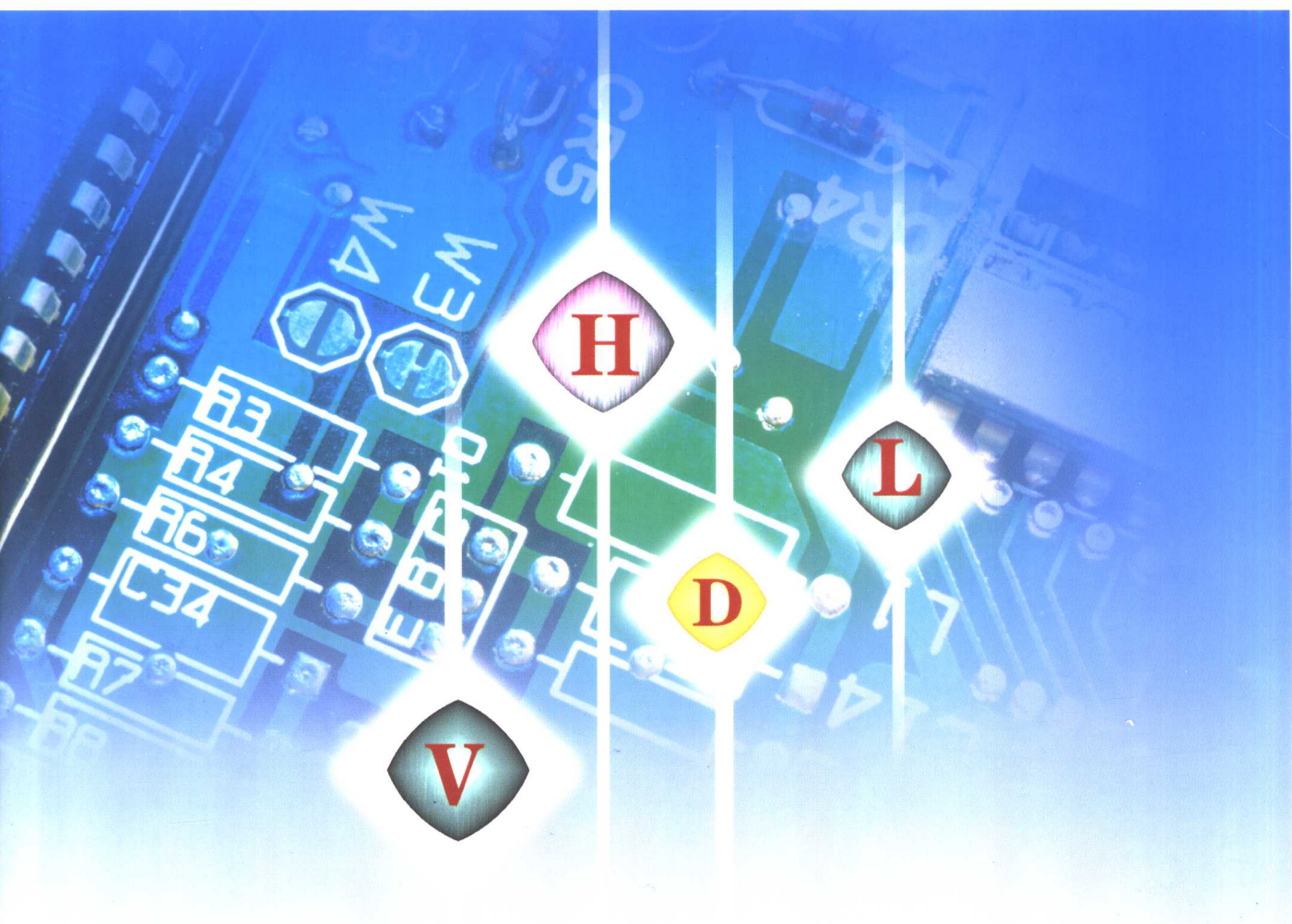


VHDL 电路设计

实用教程

齐洪喜 陆颖 编著



清华大学出版社

VHDL 电路设计实用教程

齐洪喜 陆颖 编著

清华大学出版社
北 京

内 容 简 介

本书分四部分对 VHDL 进行了全面、详细、系统的介绍。第一部分主要介绍 VHDL 语言的编程环境及开发软件 MAX+ plusII 的有关知识。第二部分主要介绍 VHDL 语言的基本知识和使用该语言设计简单逻辑电路的基本方法。第三部分主要通过实例介绍用 VHDL 语言设计大型复杂数字逻辑电路的方法。第四部分介绍 VHDL 语言中的编码规则。本书可使读者由浅入深地学习 VHDL 语言,并最终成为熟练使用该语言设计数字电路的专业人员。

本书结构清晰,易读易懂,非常适合大、中专院校相关专业作为教材和参考书,同时也可作为数字电路设计人员的自学参考书。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

VHDL 电路设计实用教程/齐洪喜,陆颖编著. —北京:清华大学出版社,2004.6

ISBN 7-302-08515-3

I. V… II. ①齐…②陆… III. ①电子电路—电路设计—教材②硬件描述语言,VHDL—程序设计—教材 IV. ①TN702②TP312

中国版本图书馆 CIP 数据核字(2004)第 035423 号

出 版 者:清华大学出版社 地 址:北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编:100084

社 总 机:010-62770175 客 户 服 务:010-62776969

责任编辑:宋 韬

封面设计:付剑飞

印 装 者:北京市清华园胶印厂

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:16.75 字数:385千字

版 次:2004年6月第1版 2004年6月第1次印刷

书 号:ISBN 7-302-08515-3/TP·6117

印 数:1~3000

定 价:27.00元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770175-3103 或(010)62795704。

前 言

20 世纪 80 年代以来,随着电子技术的不断发展,对数字电路设计的要求越来越高。同时,电路设计的复杂程度也在不断地提高,其主要的影晌就是直接推动了 EDA(Electronic Design Automatic, 电子设计自动化)的发展。随着 EDA 的发展,硬件电子电路的设计几乎全部可以依靠计算机来完成,这样就大大缩短了硬件电子电路设计的周期,从而使电子厂家可以迅速开发出品种多、批量小的产品,以便满足市场的需求。

EDA 要求必须使用硬件描述语言来描述电子电路。目前,硬件描述语言包括 VHDL、VerilogHDL、ABEL-HDL、AHDL 和硬件 C 语言。目前最流行且应用最广的是 VHDL,其他语言都存在这样或那样的缺陷,因此,是否掌握 VHDL 语言已经成为衡量每个电子工程师是否合格的标准。

VHDL 的英文全名是 Very-High-Speed Integrated Circuit Hardware Description Language,翻译成中文就是“超高速集成电路硬件描述语言”,它诞生于 1982 年。1987 年底,VHDL 被 IEEE 和美国国防部确认为标准集成电路硬件描述语言。自 IEEE 公布了 VHDL 的标准版本 IEEE-1076(简称 87 版)之后,1993 年又公布了新的 VHDL 版本,即 IEEE 标准的 1076-1993 版本(简称 93 版),目前主要使用该版本。现在,VHDL 作为 IEEE 的工业标准硬件描述语言,得到了众多 EDA 公司的支持,因此其在电子工程领域已成为事实上的通用硬件描述语言。专家认为,将来 VHDL 语言会承担起大部分数字系统的设计任务,因此掌握 VHDL 语言已经成为硬件设计的一种时尚。

本书分四部分对 VHDL 进行了全面、详细、系统的介绍。

第一部分包括第 1 章,主要介绍 VHDL 语言的编程环境及开发软件 MAX+plusII 的有关知识。第二部分包括第 2~10 章,主要介绍 VHDL 语言的基本知识和使用该语言设计简单逻辑电路的基本方法。第三部分包括第 11~12 章,主要通过实例介绍用 VHDL 语言设计大型复杂数字逻辑电路的方法。第四部分包括第 13 章,介绍了作者在实践中总结出来的 VHDL 语言中的编码规则。本书可使读者由浅入深地学习 VHDL 语言,并最终成为熟练使用该语言设计数字电路的专业人员。

本书主要由齐洪喜、陆颖编著,另外,赵磊、尹建民、徐日强、于秋生、于丙超、张海霞、周松建、陈策、仇谷峰、管佩森、李志伟、梁斌、张小刚、刘红、乔丰立、张万春、冉林仓也编写了部分内容,在此一并表示感谢。

由于作者的水平有限,在编著过程中难免有错误及不当之处,敬请读者不吝赐教。

作 者
2004 年 3 月

目 录

第 1 章 VHDL 开发工具 MAX + plusII	1
1.1 MAX + plusII 的安装	1
1.2 MAX + plusII 设计的输入	2
1.2.1 MAX + plusII 的启动	2
1.2.2 新的图形文件的建立	2
1.2.3 保存文件	4
1.2.4 指定项目名称	5
1.2.5 创建默认符号	6
1.3 项目的编译	6
1.3.1 打开并指向要编译的文件	6
1.3.2 打开编译器窗口	7
1.3.3 器件的选择及管脚的分配	7
1.3.4 全局逻辑综合方式的选择	9
1.3.5 FLEX 器件的进位/级联链的设置	9
1.3.6 定时的设置	10
1.3.7 编译	11
1.4 定时分析	11
1.5 时间仿真	13
1.6 器件的编程	13
1.6.1 利用 Altera 编程器对 MAX 和 EPROM 系列器件进行编程	13
1.6.2 通过 JTAG 实现在系统编程	14
1.6.3 利用 ByteBlaster 配置 FLEX 系列器件	15
第 2 章 VHDL 语言程序的基本结构	16
2.1 实体	16
2.1.1 实体说明	17
2.1.2 结构体	19
2.2 结构体的子结构描述	21
2.2.1 块(BLOCK)语句结构	21
2.2.2 进程(PROCESS)语句结构	23
2.3 子程序(SUBPROGRAM)语句结构	27
2.3.1 过程	27
2.3.2 函数	29
2.4 子程序的重载	32
2.4.1 函数的重载	33
2.4.2 过程的重载	37

第 3 章 程序包及库和配置	39
3.1 程序包	39
3.1.1 程序包首	39
3.1.2 程序包体	40
3.2 库	45
3.3 配置	47
3.3.1 默认配置	48
3.3.2 元件配置	50
3.3.3 结构体的配置	52
3.3.4 块的配置	53
第 4 章 VHDL 语言的对象和数据类型及运算操作符	55
4.1 VHDL 的对象	55
4.1.1 常量	55
4.1.2 变量	56
4.1.3 信号	57
4.1.4 信号与变量的区别	59
4.2 VHDL 的数据类型	61
4.2.1 标准的数据类型	61
4.2.2 用户自定义的数据类型	65
4.2.3 用户定义的子类型	71
4.3 IEEE 预定义的标准	72
4.3.1 标准逻辑位 STD_LOGIC 数据类型	72
4.3.2 标准逻辑矢量 STD_LOGIC_VECTOR 数据类型	73
4.3.3 其他预定义标准数据类型	73
4.4 数据类型的转换	74
4.4.1 函数转换法	74
4.4.2 类型标记转换法	76
4.4.3 常数转换法	76
4.5 VHDL 语言的运算操作符	77
4.5.1 逻辑运算符	78
4.5.2 算术运算符	78
4.5.3 并置运算	79
4.5.4 关系运算符	80
4.5.5 移位运算符	80
4.5.6 运算符的重载	82
第 5 章 VHDL 结构体的描述方式	88
5.1 结构体的行为描述方式	88
5.2 结构体的数据流描述方式	89
5.3 结构体的结构描述方式	90

5.4 结构体的混合描述方式	92
第6章 VHDL 语言中的顺序语句	94
6.1 WAIT 语句	95
6.1.1 WAIT ON 语句	95
6.1.2 WAIT UNTIL 语句	96
6.1.3 WAIT FOR 语句	97
6.1.4 多条件 WAIT 语句	98
6.1.5 超时等待处理	98
6.2 变量赋值语句	100
6.3 信号代入语句	100
6.4 IF 语句	101
6.4.1 门控控制语句	102
6.4.2 二选一控制语句	102
6.4.3 IF 语句的多选择控制语句	103
6.5 CASE 语句	105
6.6 LOOP 语句	108
6.6.1 FOR LOOP 语句	108
6.6.2 WHILE LOOP 语句	109
6.7 NEXT 语句	110
6.8 EXIT 语句	111
6.9 RETURN 语句	112
6.10 NULL 语句	113
6.11 过程调用语句	114
6.12 断言(ASSERT)语句	115
6.13 REPORT 语句	117
第7章 VHDL 语言中的并行语句	119
7.1 进程语句	119
7.2 并行信号代入语句	122
7.2.1 并发信号代入语句	124
7.2.2 条件信号代入语句	125
7.2.3 选择信号代入语句	126
7.3 并行过程调用语句	127
7.4 块语句	129
7.5 并行断言语句	132
7.6 元件例化语句	133
7.7 生成语句	135
7.7.1 FOR 格式的生成语句	135
7.7.2 IF 格式的生成语句	138

第 8 章 组合逻辑电路设计	140
8.1 门电路	140
8.1.1 二输入与门	140
8.1.2 二输入或门	142
8.1.3 二输入与非门	143
8.1.4 二输入或非门	145
8.1.5 反相器	146
8.1.6 二输入异或门	147
8.1.7 四输入与非门	149
8.2 编码器和译码器	151
8.2.1 编码器	151
8.2.2 译码器	155
8.3 多路选择器	157
8.3.1 二选一多路选择器	157
8.3.2 四选一多路选择器	158
8.4 比较器	160
8.5 加法器	162
8.5.1 半加器	162
8.5.2 全加器	163
8.6 求补器	164
8.7 三态门	165
第 9 章 时序逻辑电路设计	167
9.1 时钟信号和复位信号	167
9.1.1 时钟信号	167
9.1.2 复位信号	169
9.2 触发器	171
9.2.1 D 触发器	171
9.2.2 JK 触发器	175
9.2.3 T 触发器	177
9.2.4 锁存器	178
9.3 寄存器	180
9.4 移位寄存器	181
9.4.1 串行输入、串行输出移位寄存器.....	181
9.4.2 循环移位寄存器	182
9.5 计数器	184
9.5.1 同步计数器	184
9.5.2 异步计数器	189
第 10 章 有限状态机的设计	192
10.1 一个有限状态机的描述实例.....	193

10.1.1 功能要求	193
10.1.2 状态分析	194
10.1.3 交通灯控制器的三进程描述方式	195
10.1.4 交通灯控制器的双进程描述方式	197
10.1.5 交通灯控制器的单进程描述方式	204
10.2 有限状态机的复位	206
10.2.1 同步复位信号	206
10.2.2 异步复位信号	210
第 11 章 VHDL 语言的设计实例一:频率计的设计	214
11.1 频率计实现的功能	214
11.2 频率计各部分的分析	214
11.3 频率计各部分的设计和实现	214
11.3.1 时基进程的设计和实现	215
11.3.2 计数器的设计和实现	215
11.3.3 七段译码器的设计和实现	217
11.4 频率计的综合设计	219
第 12 章 VHDL 语言的设计实例二:计算器的设计	225
12.1 计算器实现的功能	225
12.2 计算器各个组成部分的分析	225
12.2.1 计算器的计算部分	225
12.2.2 计算器的存储部分	226
12.2.3 计算器的显示部分	226
12.2.4 计算器的输入部分	226
12.3 计算器各部分的设计和实现	226
12.3.1 计算器计算部分的设计和实现	227
12.3.2 计算器输入部分的设计和实现	231
12.3.3 计算器显示部分的设计和实现	232
12.4 计算器的综合设计	235
第 13 章 VHDL 语言的编码格式	245
13.1 文件头的编码格式	245
13.2 修改说明的格式	245
13.3 源代码的注释	246
13.4 VHDL 语言中实体的命名	247
13.5 VHDL 语言中的结构体的命名	248
13.6 程序包的书写格式	248
13.7 函数和过程的编码格式	249
13.8 信号的编码格式	250
13.9 变量的编码格式	251
13.10 进程的编码规则	252

13.11 时钟信号的编码规则	253
13.12 测试工作台的编码规则	253
13.13 其他的编码规则	253
附录 A VHDL 语言的保留字	255
参考文献	256

第 1 章 VHDL 开发工具 MAX + plusII

MAX + plusII 是由 ALTERA 公司开发的,现在被业界公认为是最优秀的 CPLD 开发平台之一。其全称是 Multiple Array Matrix and Programmable Logic User Systems。本章将向读者介绍 MAX + plusII 的安装和使用方法。

1.1 MAX + plusII 的安装

MAX + plusII 的运行和安装对系统没有太高的要求。在当前的 Windows 95、Windows 98、Windows ME、Windows 2000、Windows NT、Windows XP 上都可以安装运行此软件。

MAX + plusII 现在共有三种版本:商业版、基本版、学生版。

商业版:主要是针对电子生产企业而设计的。所有的 MAX + plusII 的功能都能使用,但是,在使用时要验证用户的合法性。

基本版:在商业版的基础上对时序分析,对 VHDL 语言的综合等方面进行了限制。在使用时,要输入 ALTERA 的授权码。

学生版:主要是 ALTERA 公司专门为在校的大学生设计的,也是在商业版的基础上对某些功能进行了限制,支持的芯片也较商业版少。

另外,就安装所占用的空间而言,商业版占的最多,基本版次之,学生版最少。

安装 MAX + plusII 的具体步骤如下:

(1) 运行 \pc \ maxplusII \ install. exe 文件,然后进入安装界面,按照屏幕上的提示进行操作。在安装过程中,如果需要帮助,则选择 Help 按钮。

(2) 当 MAX + plus II 安装成功后,readme 文件将自动出现。它含有一些重要的信息,用户可以仔细阅读它。

(3) 通过 Windows 系统菜单“开始”\“程序”\ MAXPLUSII \ MAXPLUSII,运行 MAXPLUSII。

如果安装的是 MAX + plusII 学生版,那么可直接使用。但是如果安装的是 MAX + plusII 的基本版,现在并不能马上使用,必须向 ALTERA 公司申请授权码然后将授权码输入到 MAX + plus II 中,才可以使用。

向 ALTERA 公司申请授权码有两种方式,如果你没有网络连接,可以使用你的硬盘卷序列号上网申请,ALTERA 公司将把你的授权码发到你指定的电子邮箱里。如果你有网络,便可以直接连接到 ALTERA 公司网站(<http://www.altera.com/>)上进行授权码的申请。在申请的过程中需要输入你的 NIC 号和电子邮件地址,然后回答 ALTERA 公司的几个问题,完成后 ALTERA 公司将发给你。

设置授权码的方法如下:启动 MAX + plusII 后,在管理器窗口的 Options 菜单中,选择 Lincese setup 菜单项,然后输入你的授权码,这样 MAX + plusII 基本版就可以使用了。

1.2 MAX + plusII 设计的输入

MAX + plusII 设计的输入有三种方式:图形输入、文本输入、波形输入。其中最简单、最明了的是图形输入方式,这里先简要介绍一下图形输入方式的操作步骤和需要注意的问题,以使大家对 MAX + plusII 的环境先有一个大体的了解。

1.2.1 MAX + plusII 的启动

在 Windows 的桌面上有一个 MAX + plusII 的图标,双击即可启动,进入 MAX + plusII 的管理器窗口,如图 1-1 所示。这个窗口共有五个主菜单选项,分别是 MAX + plusII、File、Assign、Options、Help 选项。在该窗口的标题栏中显示的是当前用户操作的工程的保存路径以及工程名。

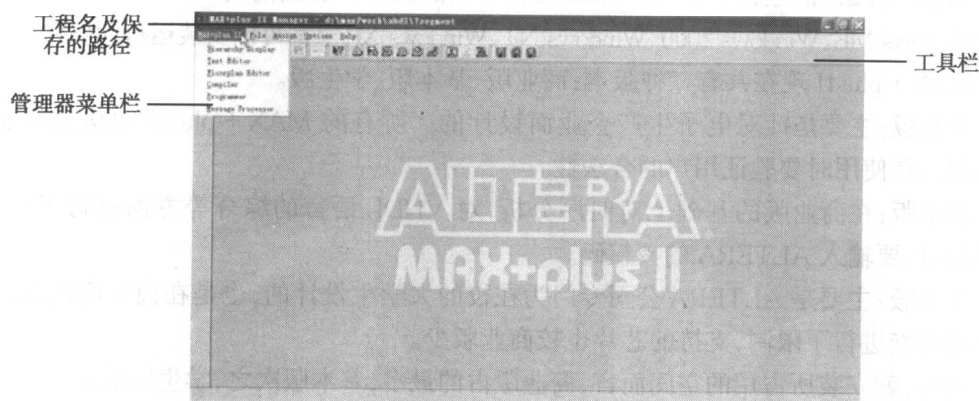


图 1-1 MAX + plusII 的管理器窗口

1.2.2 新的图形文件的建立

新的图形文件的建立步骤如下:

(1) 在 File 菜单中,选择 New 菜单项,出现如图 1-2 所示的对话框。在对话框中选择 Graphic Editor File,然后单击 OK 按钮,将会出现一个无标题的图形编辑窗口,如图 1-3 所示。

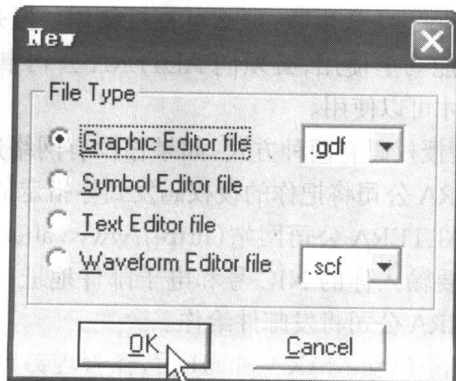


图 1-2 新建文件类型对话框

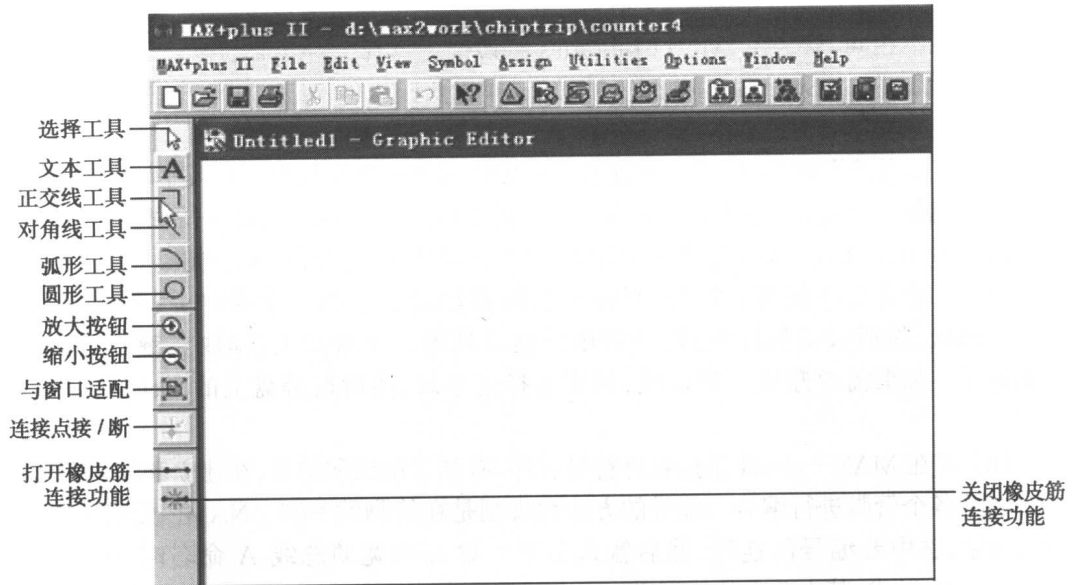


图 1-3 图形编辑窗口

(2) 在图形编辑窗口中输入 ALTERA 图元。当选择工具按钮有效时,在图形编辑器窗口的空白处单击以确定输入位置,将出现一个闪烁的小黑点来表示选择的位置,然后在 Symbol 菜单中,选择 Enter Symbol;或双击,将出现一个 Enter Symbol 对话框,如图 1-4 所示。在 Symbol Libraries 列表框中选择 .. \maxplus2 \max2lib \prim 路径。在图 1-4 的 Symbol Files 列表框中,所有的 ALTERA 图元都以列表的方式显现出来,选择想要输入的图元,然后单击 OK 按钮。

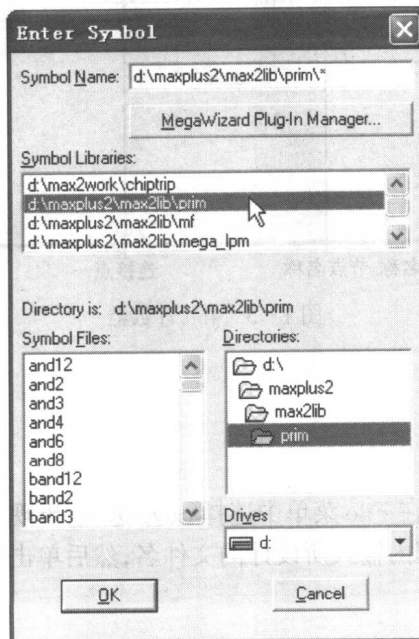


图 1-4 输入符号对话框

(3) 在图形编辑窗口中输入 74 系列的符号。其输入方法与步骤(2)中提到的输入图元的方法基本相同,只是在 Symbol Libraries 对话框中选择 .. \ maxplus2 \ max2lib \ mf 路径。在图 1-4 的 Symbol Files 列表框中,选择想要输入的 74 系列符号,然后单击 OK 按钮。

(4) 在图形编辑窗口中输入 LPM 符号的方法与前面介绍的图元以及 74 系列符号的输入方法相似。只是在 Symbol Libraries 对话框中选择 .. \ maxplus2 \ max2lib \ mega _ lpm, 再在 Symbol Files 里面选择自己需要的 LPM 符号,然后单击 OK 按钮。

(5) 如果需要连接两个端口,只需将鼠标移动到其中的一个端口上,鼠标自动变成“+”形状,然后按下鼠标左键,并将鼠标拖动到第二个端口上,释放左键,一条连线就画好了。如果需要删除一条连线,只需选择这条线,然后按键盘上的 Del 键,就可以删除。

(6) 要在 MAX+ plusII 里使各种符号有序,有利于调试和仿真,在建立和编辑图形文件时要给各个管脚进行编号。编号的方法和规则是在管脚的 PIN_NAME 处双击,然后输入名字,选中要编号的连线,然后输入名字。对 n 位宽的总线 A 命名时,可以采用 A[n-1..0]形式,其中单个信号用 A0, A1, A2, ..., An 形式,如图 1-5 所示。

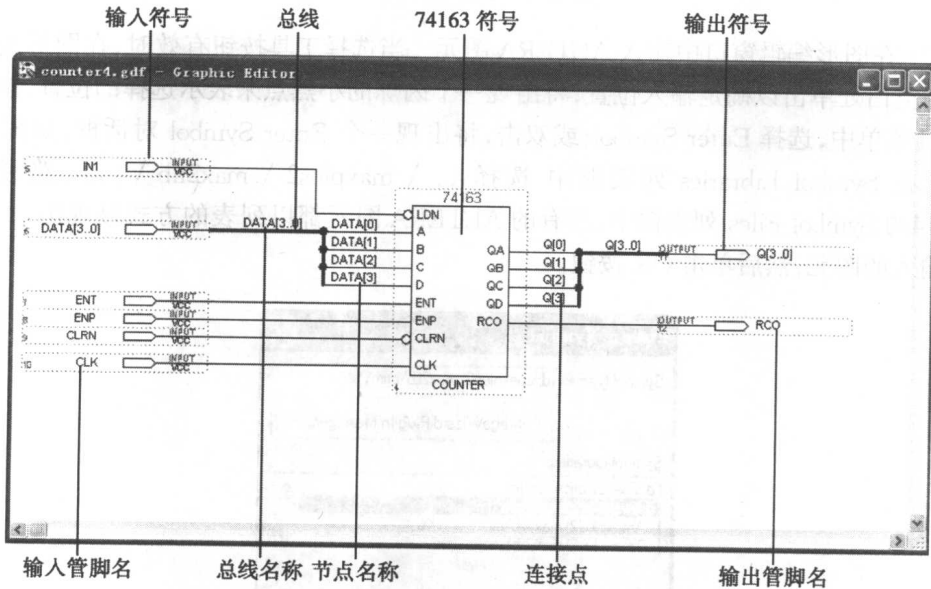


图 1-5 4bit 计数器

1.2.3 保存文件

如果需要保存文件,则在 File 菜单中选择 Save 选项,出现 Save As 对话框,如图 1-6 所示。在 File Name 对话框内输入所设计的文件名,然后单击 OK 按钮即可保存文件。

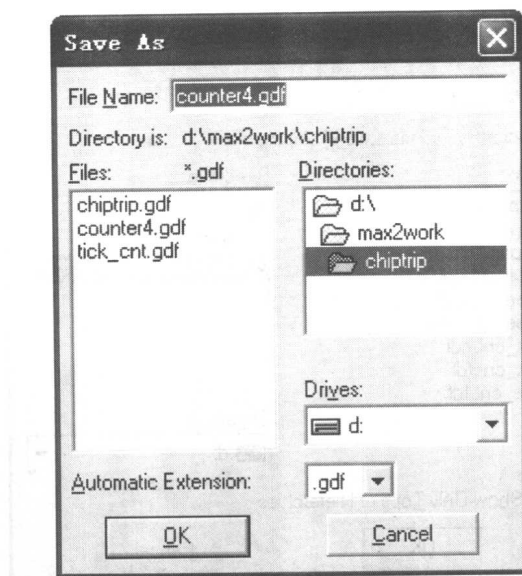


图 1-6 Save As 对话框

1.2.4 指定项目名称

在 MAX + plusII 中,编译一个项目前,必须确定一个设计文件作为当前项目。按下列步骤确定项目名:

(1) 在 File 菜单中选择 Project 菜单项,然后选择 Name 菜单项如图 1-7 所示。将出现 Project Name 对话框,如图 1-8 所示。

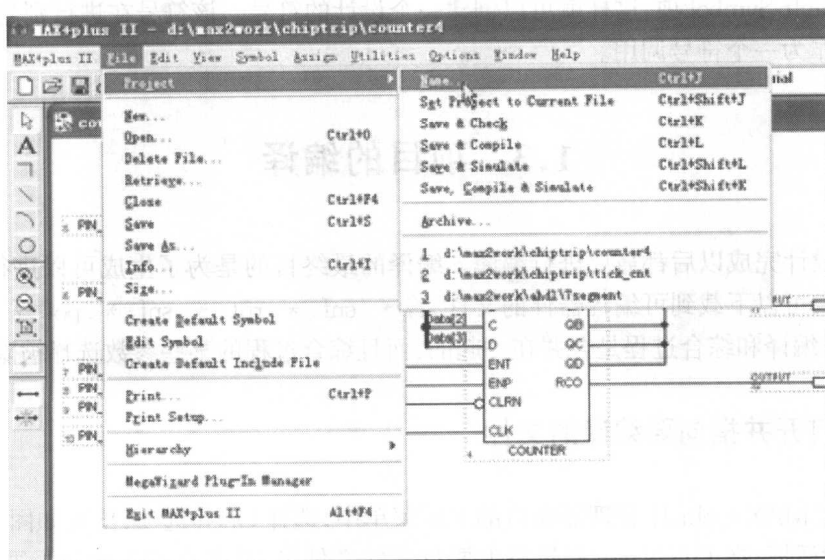


图 1-7 指定项目名称

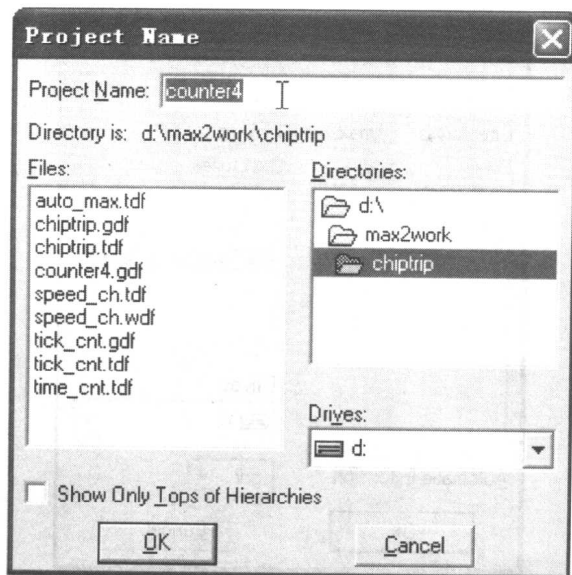


图 1-8 Project Name 对话框

(2) 在 Files 框内,选择设计文件。单击 OK 按钮。MAX + plusII 的标题条将显示新项目存储的目录和名字。

1.2.5 创建默认符号

在 File 菜单的 Project 菜单项中,选择 Save & Check 项,检查设计是否有错误。如果有错误,须根据提示进行修改,直到没有错误为止。如果没有,可以在 File 菜单中选择 Create Default Symbol 项,这样就可以创建一个设计的符号。该符号在进行别的高层设计时可以被作为一个符号调用。

1.3 项目的编译

项目设计完成以后都需要进行编译。编译的最终目的是为了生成可以进行仿真、定时分析以及可以下载到可编程器件的文件,像 *.cnf、*.rpt、*.snf、*.pof 等。在 MAX + plusII 里编译和综合过程是合并在一起的,而且综合过程的某些参数选择被隐藏了。

1.3.1 打开并指向要编译的文件

(1) 在 MAX + plusII 管理器窗口的 File 菜单中,选择 Open 选项,出现如图 1-9 所示的 Open 对话框,在 File Name 栏里添上要打开的文件名,或者在列表中选择要打开的文件名,然后单击 OK 按钮。

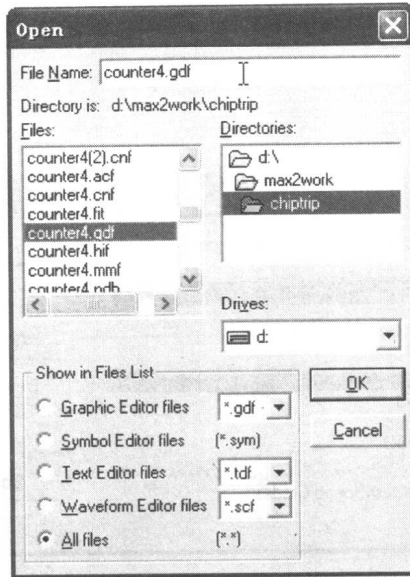


图 1-9 Open 对话框

(2) 选择 File 菜单,在 Project 菜单项的下一级菜单中选择 Set Project to Current File 项,使编译器指向已设定的现行文件,为下一步的编译做准备。

1.3.2 打开编译器窗口

在 MAX + plusII 菜单内选择 Compiler 菜单项,则出现编译器窗口,如图 1-10 所示,单击 Start 按钮即可开始编译。由于 MAX + plusII 编译器在编译时将检查项目是否有错,并对项目进行逻辑综合,然后配置到一个 Altera 器件中,同时将产生报告文件、编程文件和用于时间仿真用的输出文件,因此,在进行编译以前,应该选择最终要下载芯片。

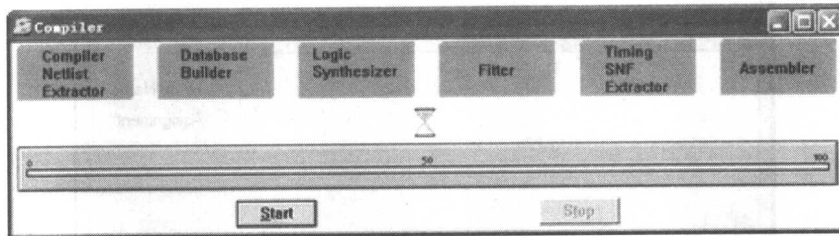


图 1-10 编译窗口

1.3.3 器件的选择及管脚的分配

在 Assign 菜单中选择 Device 菜单项,会出现如图 1-11 所示的 Device 对话框。首先在 Device Family 下拉菜单中为项目指定一个器件系列,然后,可以在 Devices 栏中选