

第3章

MAX+plus II 软件概述

3.1 MAX+plus II 软件简介

MAX+plus II 10.2 界面友好,使用便捷,被誉为业界最易学易用的 EDA 软件。它支持原理图、VHDL 和 Verilog 语言文本文件,以及波形与 EDIF 等格式的文件作为设计输入,并支持这些文件的任意混合设计。MAX+plus II 具有门级仿真器,可以进行功能仿真和时序仿真,能够产生精确的仿真结果。在适配之后,MAX+plus II 生成供时序仿真用的 EDIF、VHDL 和 Verilog 三种不同风格的网表文件。

MAX+plus II 支持主流的第三方 EDA 工具,如 Synopsys、Cadence、Synplicity、Mentor、Viewlogic、Exemplar 和 Model Technology 等。MAX+plus II 支持除 APEX20K 系列之外的所有 Altera CPLD/FPGA 大规模逻辑元件。

3.1.1 MAX+plus II 开发软件特点

1. 开放的界面

MAX+plus II 支持 Cadence、Exemplarlogic、Mentor Graphics、Synplicity、Viewlogic 和其他公司所提供的 EDA 工具接口。

2. 与结构无关

MAX+plus II 系统的核心 Compiler 支持 Altera 公司的 FLEX10K、FLEX8000、FLEX6000、MAX9000、MAX7000、MAX5000 和 Classic 可编程逻辑元件,提供了世界上唯一真正与结构无关的可编程逻辑设计环境。

3. 完全集成化

MAX+plus II 的设计输入、处理与校验功能全部集成在统一的开发环境下,这样可以加快动态调试、缩短开发周期。

4. 丰富的设计库

MAX+plus II 提供丰富的库单元供设计者调用,其中包括 74 系列的全部元件和多种特殊的逻辑功能(Macro-Function)以及新型的参数化的兆功能(Mage-Function)。

5. 模块化工具

设计人员可以从各种设计输入、处理和校验选项中进行选择从而使设计环境用户化。

6. 硬件描述语言(HDL)

MAX+plus II 软件支持各种 HDL 设计输入选项,包括 VHDL、Verilog HDL 和 Altera 自己的硬件描述语言 AHDL。

7. Opencore 特征

MAX+plus II 软件具有开放核的特点,允许设计人员添加自己认为有价值的宏函数。

3.1.2 MAX+plus II 开发软件的主要功能

1. 支持的 PLD 元件

主要支持三大类近 8 个系列的 PLD 元件:

- 多阵列结构的 MAX9000、MAX7000、MAX3000 和 Classic 系列;
- 柔性阵列结构的 FLEX10K、FLEX8000 和 FLEX6000 系列;
- 先进阵列结构的 ACEX1K 系列。

2. 支持的设计输入方式

- 基于图形编辑器创建的原理图设计文件(.gdf 文件);
- 基于文本编辑器创建的 HDL 文本文件;
- 基于波形编辑器创建的波形输入设计文件(.wdf 文件);
- 基于其他常用的 EDA 工具产生的输入文件。

3. 提供设计编译

通过编辑器可完成设计项目的规则检查、逻辑综合与自动适配、多元件划分、错误自动定位、定时驱动编辑和开放核环境等功能。

4. 提供设计验证

通过定时分析器能对设计项目进行功能仿真、时序仿真、波形分析、定时分析等验证。

5. 提供元件的编程和配置

提供操作灵活、使用方便的元件编程(Programming)和配置(Configuration)工具。

3.1.3 MAX+plus II 运行环境需求

支持的操作系统: Windows 98/Me/2000 以及 Windows XP。

安装所需空间: 1GB。

内存要求: 可用 64MB, 推荐内存 64MB 以上。

3.2 MAX+plus II 的安装

3.2.1 MAX+plus II 软件安装

(1) 将 MAX+plus II 光盘放进光驱, 这里假设光驱的驱动器号为 F。选择 Windows 系统的“开始”→“运行”菜单, 输入“F:\ALTERA_MAXplusII_10.0_FULL\SETUP.EXE”, 然后出现欢迎界面, 如图 3.1 所示, 单击 Next 按钮, 即可开始安装过程。



图 3.1 Welcome 窗口

(2) 进入图 3.2 所示窗口, 单击 Yes 按钮, 表示接受此协议。此时出现提示, 告诉你需要一个 license 文件来运行程序, 单击此提示中的 Next 按钮。

(3) 进入图 3.3 所示窗口, 输入用户名和公司名, 单击 Next 按钮。

(4) 进入图 3.4 所示窗口, 选择完全安装即默认选项, 单击 Next 按钮。

(5) 进入图 3.5 所示窗口, 默认安装路径时, 单击 Next 按钮。若把软件安装在 C 盘, 单击 Browse 按钮进行路径设置。这里需要注意的是 MAX+plus II 软件的安装路径中不得含有中文。

(6) 接着在出现的窗口一直单击 Next 按钮, 直到如图 3.6 所示开始安装。



图 3.2 授权许可界面

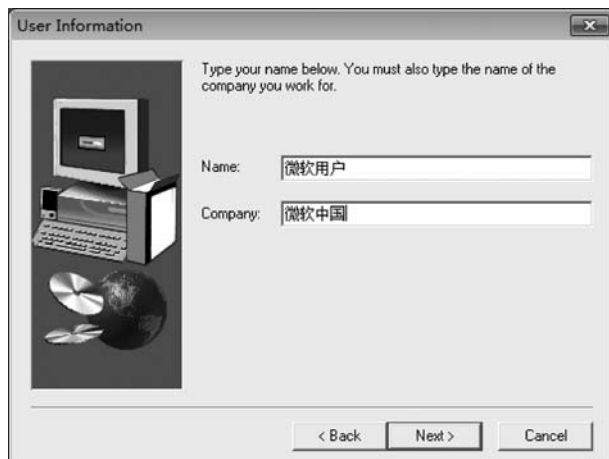


图 3.3 用户信息界面

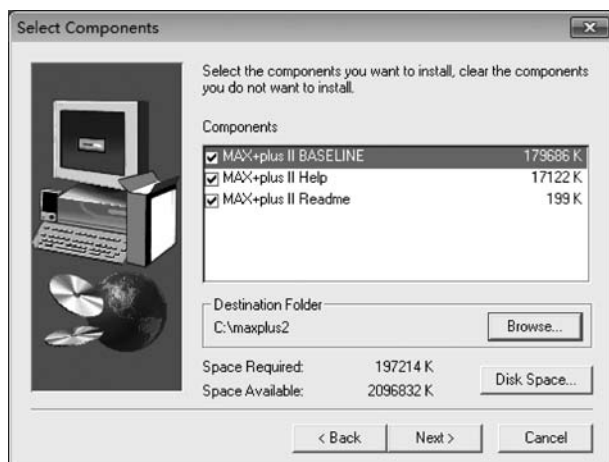


图 3.4 安装组件选择窗口

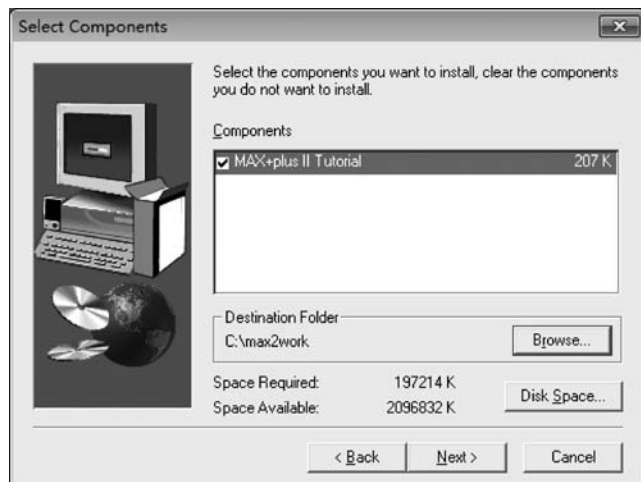


图 3.5 安装路径选择窗口



图 3.6 执行安装

3.2.2 MAX+plus II 软件授权

(1) 通过 Windows 系统菜单“开始”→“程序”→MAX+plus II→MAX+plus II, 运行 MAX+plus II。首次运行 MAX+plus II, 会出现 License Agreement(授权协议)对话框(见图 3.7), 按 Tab 键, 然后再单击 Yes 按钮即可。当出现对话框提示当前的软件保护号(Software Guard ID)时, 应到指定的 Internet 站点上申请授权号。

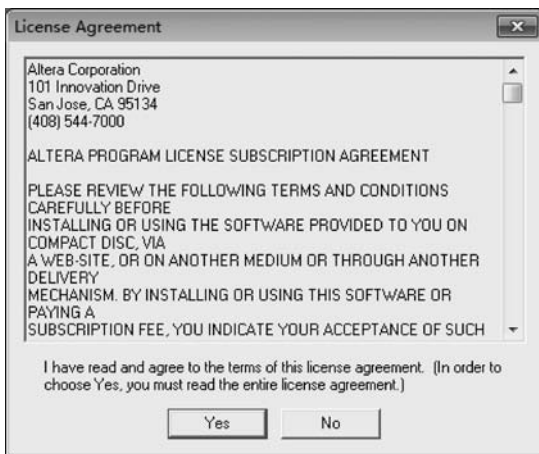


图 3.7 授权协议对话框

(2) 将申请到的授权号输入 Authorization Code 对话框的文本输入框中,单击 OK 按钮即可,也可先单击 Validate 按钮看一下授权号是否正确。

安装完 MAX+plus II 系统文件后,可通过运行光盘上的文件“Acroread\win\disk1\setup.exe”来安装 PDF 文件阅读器,以便阅读包括 Altera 公司的数据手册、应用笔记等文档内容。光盘根目录下的文件 Altera.pdf 是光盘上 Altera 所有文档的索引。

3.3 MAX+plus II 设计流程

3.3.1 设计流程

根据图 3.8,下面简述设计流程的几个重要步骤,后面章节实例中将进行详细说明。

1. 设计输入

提供图形、文本和波形编辑器实现图形、AHDL、VHDL、Verilog HDL 或波形的输入,也可输入网表文件。

2. 项目编译

提供了一个完全集成的编译器(Compiler),可直接完成从网表提取到最后编程文件的生成,包含时序模拟、适配的标准文件。

3. 项目校验

对设计项目的功能、时序进行仿真和时序分析,判断输入/输出间的延迟。

4. 项目编程

将设计下载/配置到所选择的元件中。

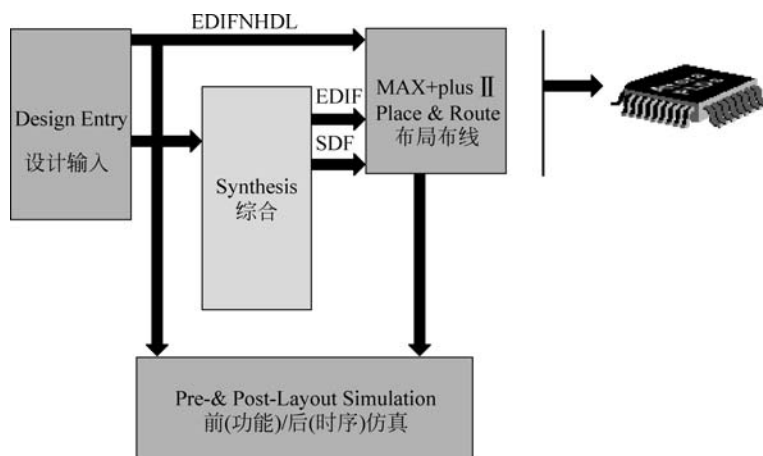


图 3.8 MAX+plus II 设计流程

3.3.1.1 MAX+plus II 工具按钮简介

1. 设计文件操作按钮(图 3.9)



图 3.9 文件操作按钮

2. 设计项目处理操作按钮(图 3.10)



图 3.10 项目处理操作按钮

3. 设计文件编辑操作按钮(图 3.11)

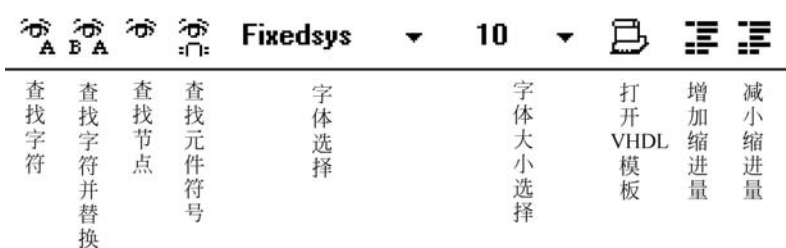


图 3.11 文件编辑操作按钮

4. 绘图操作和绘制波形操作按钮(图 3.12)



图 3.12 绘图操作和绘制波形操作按钮

3.3.1.2 基于原理图设计文件的输入

用图形编辑器所做的设计也叫作原理图设计,通常使用 MAX+plus II 提供的库元件和用户自定义的符号进行设计。采用这种方式时,应采用自顶向下(Top-Down)的设计方式,就是从系统级开始,把系统划分为若干个基本单元,然后再把每个基本单元划分为下一层次的基本单元,一直这样做下去,直到可以直接用库文件来实现为止。当对系统认识得很清楚并且能把它划分为一些基本单元,而且系统速率要求较高时,或者整个系统中有对时间特性要求较高的部分时,一般采用原理图设计的方法。尽管原理图设计的效率比较低,但是能够保证实现系统的技术指标,且便于系统的仿真和设计的改进。

下面通过一个实例介绍原理图设计的方法和技巧。所举例子是用 74161 设计一个模 12 计数器。

1. 项目的建立

1) 建立工作目录

用户的每个独立设计都对应一个项目,每个项目可包含一个或多个设计文件,其中有一个是顶层文件,顶层文件的名字必须与项目名相同。编译器是对项目中的顶层文件进行编译。项目还管理所有中间文件,所有项目的中间文件的文件名相同,仅后缀名(扩展名)不同。对于每个新的项目最好建立一个单独的文件夹,使设计有条理化。切记项目名不同于项目文件夹,项目文件夹可包含多个项目文件。我们在这里需要新建工程存储目录,在 D 盘下新建一个名为 mydesign 的文件夹,这个文件夹用来存放以后所有的 MAX+plus II 工程,在 mydesign 下新建一个名为 graph 的文件夹用来存储本次实验的工程。

2) 启动 MAX+plus II

启动 MAX+plus II 10.2,进入图 3.13 所示 MAX+plus II 管理器窗口。在 File 菜单中选择 Project 的 Name 选项。

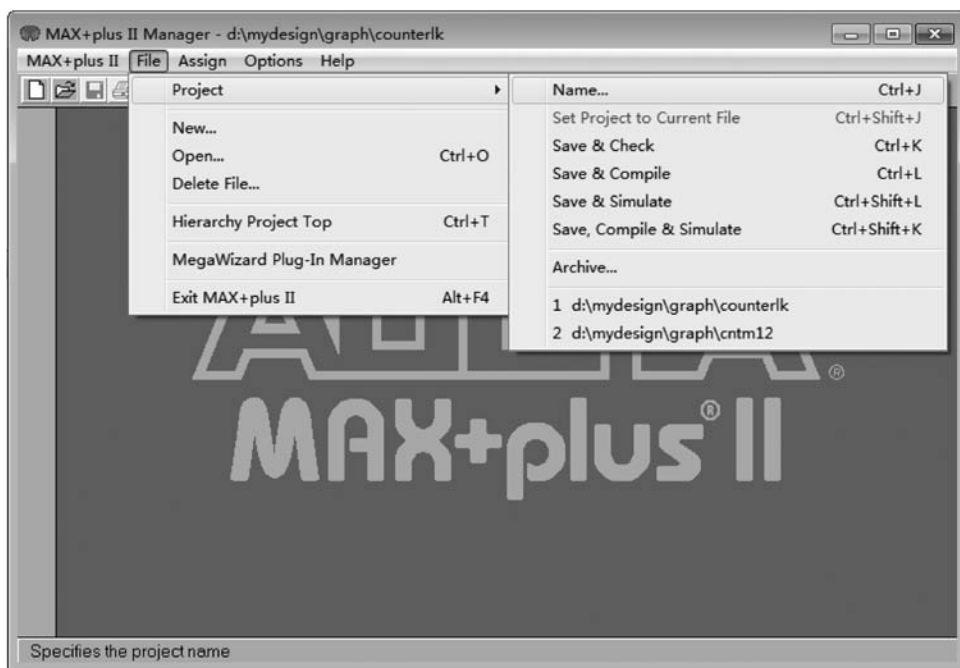


图 3.13 管理器窗口

3) 建立项目

出现如图 3.14 所示的对话框,在 Directories 区选中刚才为项目所建的目录;在 Project Name 区键入项目名,单击 OK 按钮即完成项目建立。需要注意的两点是,MAX+plus II 软件对大小写不敏感,而且不识别中文。

4) 目标元件选择

选择菜单命令 Assign→Device, 出现如图 3.15 所示的窗口。选择 ACEX1K 系列的芯片, Devices 可以选择 AUTO, 让软件自动匹配具体元件。也可以选择一个确定的型号, 这里以 EP1K30QC208-1 为例。单击 OK 按钮, 完成选择。



图 3.14 建立项目名

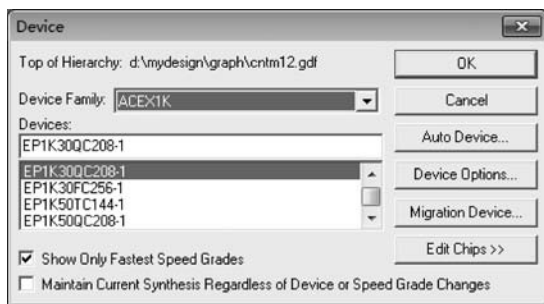


图 3.15 设定元件

完成以上步骤之后, 一个工程就新建完毕了。

2. 原理图设计

1) 创建图形输入文件

在 MAX+plus II 中要打开图形编辑器输入原理图有两种方法: 一种方法是选择 MAX+plus II→Graphic Editor file, 直接打开图形编辑器; 另一种方法是选择菜单命令 File→New, 出现如图 3.16 所示的新建文件窗口。

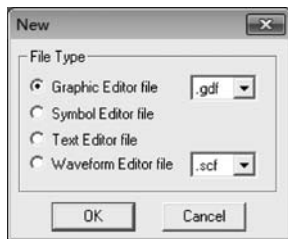


图 3.16 新建文件窗口

可以看到, 打开图形编辑器后, 如图 3.17 所示, 菜单栏和工具栏的命令选项增多了, 并且图形编辑器左侧出现一个工具条, 称它为绘图工具条。绘图工具条的含义上文已有介绍。

打开图形编辑器之后就新建了一个新的无标题图形文件, 可以在原理图设计完成后保存该文件, 也可以在设计之前保存文件。为了保证存盘的及时性, 推荐在设计之前保存文件。从 File 菜单下选择 Save, 出现文件保存对话框。单击 OK 按钮, 使用默认的文件名存盘。此处默认的文件名为 cntm12.gdf, 即项目名 cntm12 再加上图形文件的拓展名。

2) 调用库元件

设计原理图需要调用库元件, 设计之前需要根据所实现的功能选择库元件, 在使用原理图设计之前应该先将数字电路设计好。设计模 12 计数器, 就需要基本的计数元件, 在 mf 库里有 74 系列元件, 里面包含了多种计数器。

调用库元件的方法有很多种, 可以先在图形编辑区双击鼠标打开 Enter Symbol 对话框。也可以选择菜单命令 Symbol→Enter Symbol, 无论哪种方式均打开如图 3.18 所示窗口。

MAX+plus II 为实现不同的逻辑功能提供了大量的库文件, 每个库对应一个目录。这些库的具体分类及特点如表 3.1 所示。

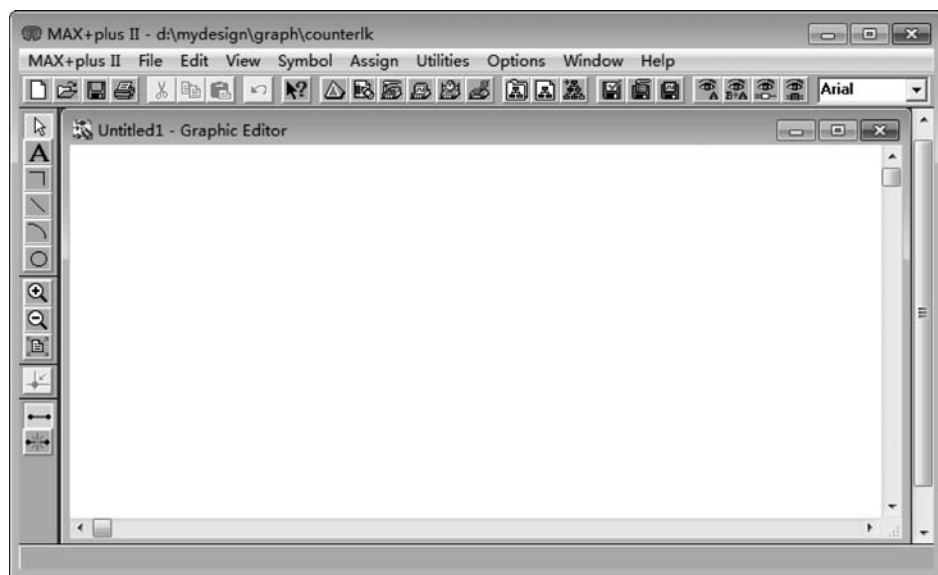


图 3.17 图形编辑窗口



图 3.18 加入库元件窗口

表 3.1 MAX+plus II 库简介

库 名	内 容
用户库	用户自建的元件,即一些底层设计
Prim(基本库)	基本的逻辑模块元件,如各种门、触发器等
mf(宏功能库)	所有 74 系列逻辑元件,如 74161
mega_lpm	包括参数化模块,功能复杂的高级功能模块,如可调模值的计数器、FIFO、RAM 等
Edif	与 mf 库类似

在库选择区双击“c:\maxplus2\max2lib\mf”，此时在元件列表区列出了该库中所有元件，找到 74161，并单击。此时 74161 出现在元件符号名输入区，如图 3.19 所示。

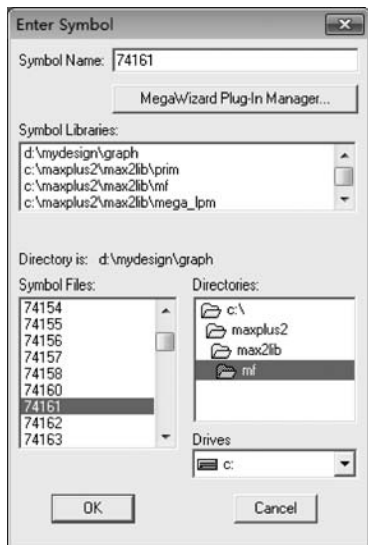


图 3.19 选择 74161 元件

双击它，或者单击 OK 按钮，就可以将 74161 添加到元件编辑区，如图 3.20 所示。

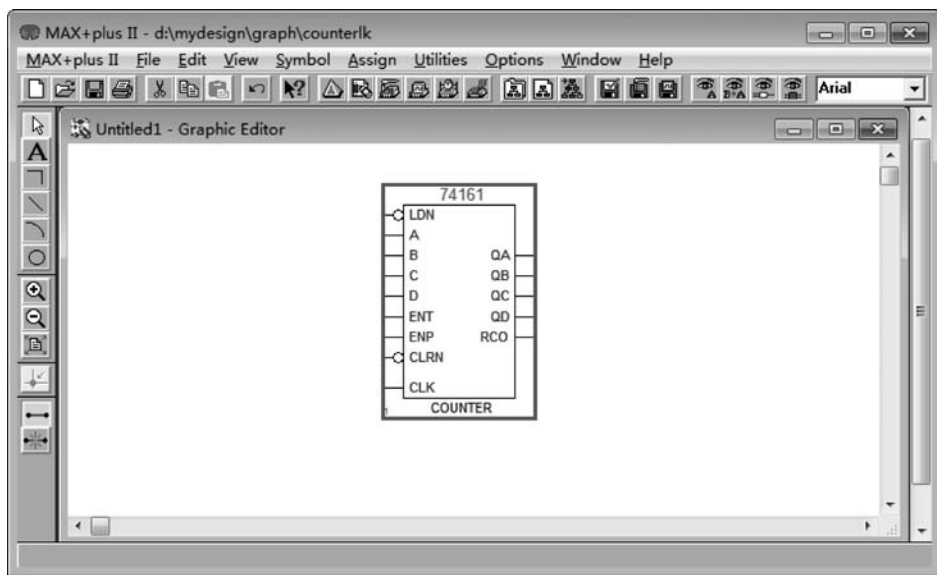


图 3.20 调入库文件后的图形编辑器窗口

对于库中调出的元件都可在帮助文件中找到相关功能说明及用法，下面从了解 74161 真值表的例子说明帮助文件的用法。在菜单栏单击 Help → Old-Style Macrofunctions，如图 3.21 所示。

在弹出的窗口下选择 Counters(74161 为计数器系列)，然后在接下来出现的窗口中选择 74161，如图 3.22 所示。

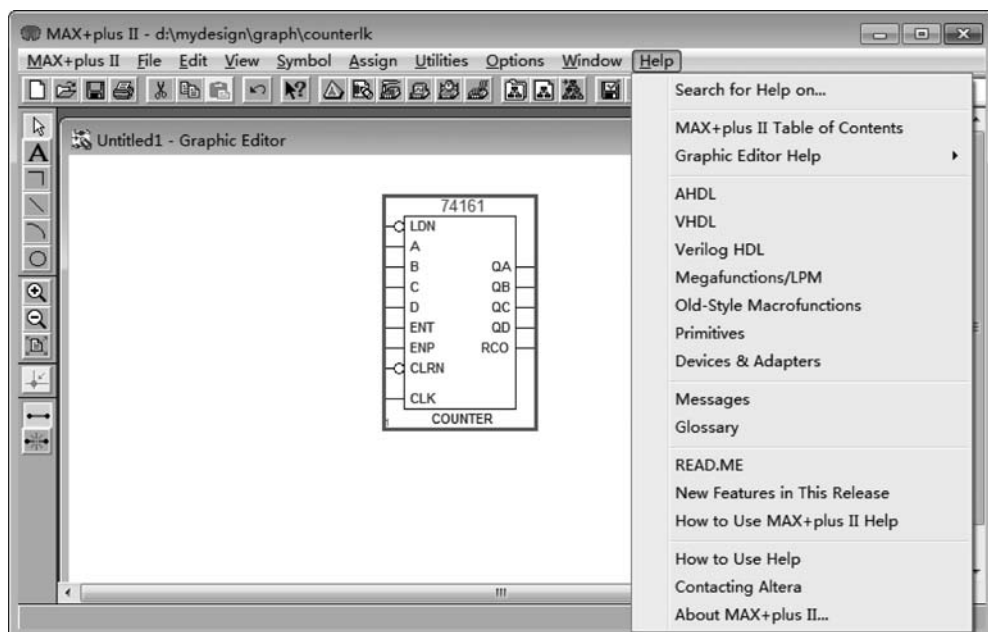


图 3.21 查看元件文档

MAX+PLUS II Version 10.2 Help															
文件(F) 编辑(E) 书签(M) 选项(O) 帮助(H)															
目录(C) 索引(I) 后退(B) 打印(P) Glossary															
74161 (Counter)															
Macrofunctions															
4-Bit Binary Up Counter with Synchronous Load and Asynchronous Clear <i>Note</i>															
Default Signal Levels: GND--A, B, C, D, CLK VCC--LDN, ENT, ENP, CLRN															
AHDL Function Prototype (port name and order also apply to Verilog HDL):															
FUNCTION 74161 (clk, ldn, clrn, enp, ent, d, c, b, a)															
RETURNS (qd, qc, qb, qa, rco);															
Inputs									Outputs						
CLK	LDN	CLRn	ENP	ENT	D	C	B	A	QD	QC	QB	QA	RCO		
X	X	L	X	X					L	L	L	L	L		
f	L	H	X	X	d	c	b	a	d	c	b	a	*		
f	H	H	X	L					QD	QC	QB	QA	*		
f	H	H	L	X					QD	QC	QB	QA	*		
f	H	H	H	H					L	L	L	L	L		
f	H	H	H	H					L	L	L	H	L		
f	H	H	H	H					L	L	H	L	L		
f	H	H	H	H					L	L	H	H	L		
f	H	H	H	H					L	H	L	L	L		
f	H	H	H	H					L	H	L	H	L		

图 3.22 74161 芯片资料

因为模 12 计数器还需要用到一些其他元件,添加的方法与上述相同,再添加一个 NAND3(三输入与非门)和一个 GND(电源地)。

3) 摆放元件

按照输入/输出关系把各元件摆放好,一般来说,元件的输入端在左边,输出端在右

边。但是这样有时候也不便于连线,需要把元件翻转一下方向。翻转元件的方法是,选中需要进行翻转的元件然后单击鼠标右键,选择快捷菜单 Flip Horizontal 或者选择快捷菜单 Rotate→180°。摆放好的元件如图 3.23 所示。

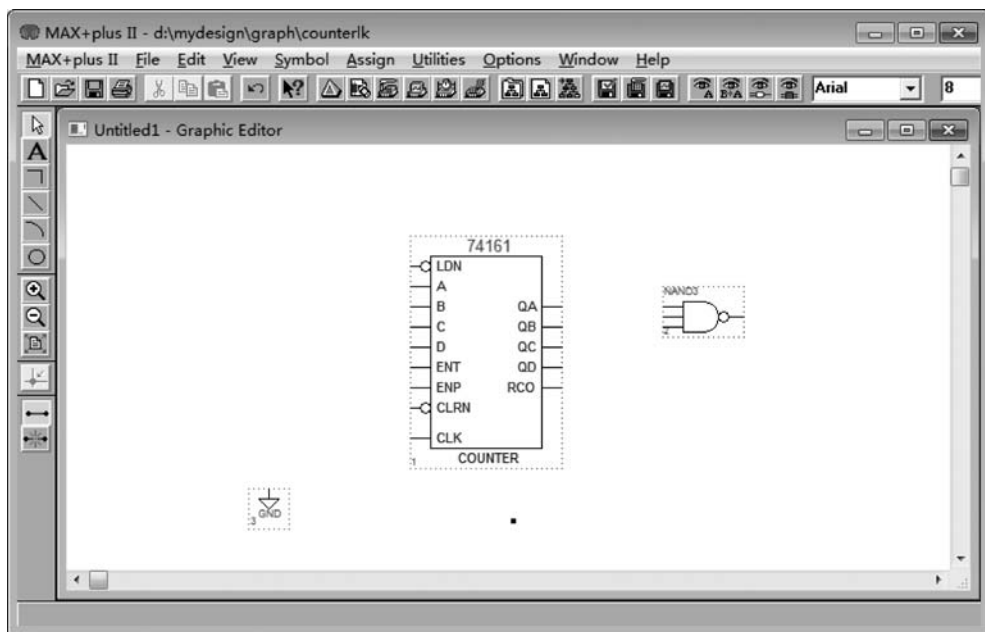


图 3.23 元件摆放

4) 连接元件

用绘图工具条里的直线和正交线把这些元件连接起来,在连线过程中会出现有交叉的地方。如果交叉的线需要连接,就在交叉点处单击已确定插入点位置,然后单击连接线接/断功能按钮,使交叉点接或者断。如果要删除一根连接线,单击这根连接线选中它,然后单击鼠标右键选择快捷菜单 Delete。连接完之后如图 3.24 所示。

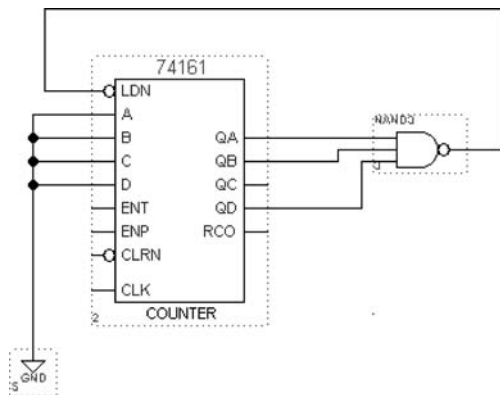


图 3.24 连接完毕

5) 定义输入/输出引脚

根据要实现的功能把原理图中的元件连接起来后,就要定义输入/输出节点。单击

绘图工具条中的文本工具,在定义输入/输出节点的连线上方单击,然后输入节点名,这一步是为定义输入/输出引脚做准备。输入/输出引脚是基本元件,在 prim 库里。从 prim 库里调出输入/输出引脚,修改引脚名,使引脚名和原理图中相应的节点名相同。输入引脚还要根据需要修改默认引脚值,引脚值有高电平 VCC 和低电平 GND。引脚如图 3.25 所示。



图 3.25 输入/输出引脚

根据模 12 计数器的原理,经过上述操作步骤后,一个完整的模 12 计数器的功能模块就设计好了,如图 3.26 所示。

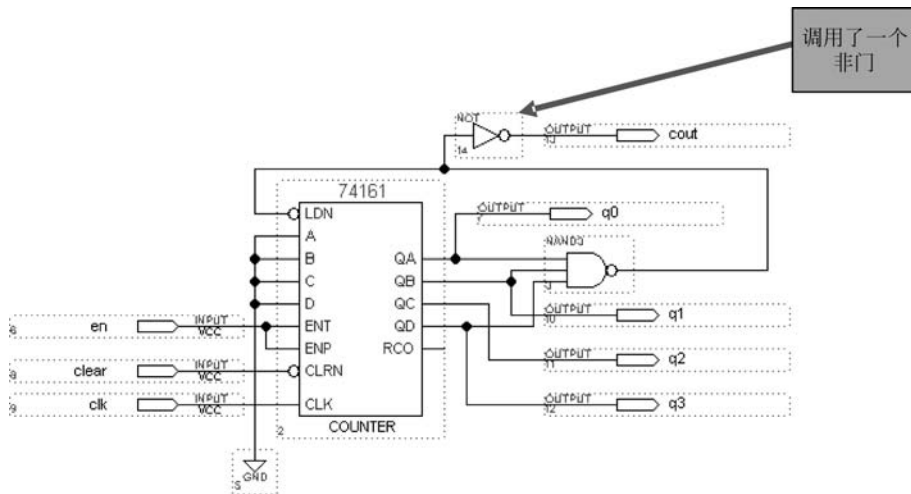


图 3.26 模 12 计数器

如果对元件摆放的位置不满意,可以单击打开橡皮筋连接功能按钮,然后移动元件,在移动的过程中,与元件连接的线不会断开。

3. 保存并检测错误

(1) 选择 File→Project→Save & Check 命令,或单击工具栏中的“保存”按钮,当前的设计被保存,编辑器窗口被打开,启动网表提取器模块检查设计文件的基本错误。检查完毕,会弹出错误与警告信息对话框。

(2) 如果错误与警告信息对话框显示无错误和警告,则单击“确定”按钮关闭对话框。如果设计有错误或警告,选择 MAX+plus II→Message Processor 命令,打开消息处理窗口,在其中获取并定位错误信息,然后在设计输入窗口中修改错误,直至检查没有错误和警告为止。

(3) 单击编译器窗口右上方的“关闭”按钮关闭编译器,返回图形编辑器窗口。

4. 创建功能模块

在设计无误的情况下,把图形设计文件创建成一个功能模块(文件拓展名为“.sym”),以供其他图形设计文件所调用,这是做复杂的设计常采用的方法。

选择 File→Create Default Symbol 命令,即把当前设计文件创建成一个同名的功能

模块 cntml2.sym。如果当前的设计已经被创建成一个功能模块,则执行此菜单命令会提示是否覆盖已创建的功能模块。如果设计有所改变,则单击“确定”按钮,即可将设计文件的改动保存到其功能模块内;如果设计没有改变,则单击“取消”按钮。

3.3.1.3 基本文本设计文件的输入

MAX+plus II 支持文本输入设计,即支持用硬件描述语言(HDL)来设计数字逻辑系统。文本编辑器是设计输入的工具。硬件描述语言有很多种,MAX+plus II 支持 AHDL、VHDL、Verilog HDL 等硬件描述语言。其中,AHDL 是 Altera 公司根据公司生产的 MAX 系列元件和 FLEX 系列元件的特点专门设计的一套完整的硬件描述语言;VHDL 是由美国军方组织开发的,1987 年成为 IEEE 标准;而 Verilog HDL 是 1983 年由 GateWay 公司首先开发成功的,经过诸多改进,于 1995 年 11 月正式被批准为 IEEE 标准。

一般来说,在系统速率较低、时间特性要求不是十分严格的情况下,往往采用硬件描述语言输入设计方式。硬件描述语言可以实现状态机、真值表、条件逻辑、布尔方程和算术操作等。复杂的项目容易通过硬件描述语言简练的、高层次的描述得以实现,所以运用硬件描述语言设计输入文件具有效率高的优点,但语言输入必须依赖综合器,好的逻辑综合器才能把语言转化成优化的电路。

下面使用 VHDL 语言设计一个十位二进制计数器,用来实现 1K 分频。

1. 创建设计输入文件

在前面的图形输入中是先创建项目后创建设计输入文件,也可以先创建设计输入文件,而后把项目指定为当前的输入文件。在模块化的设计中往往采用这种方法,即先把整个项目划分为若干模块。在各个模块的设计中,以上模块为当前项目进行设计、编译和校验,各个模块校验通过后,再把它们连接起来构成整个系统。下面在文本输入设计中介绍这种方法。

(1) 打开 MAX+plus II 设计管理器窗口,单击工具栏上的 New 按钮,选择 Text Editor file 选项,出现如图 3.27 所示的窗口。

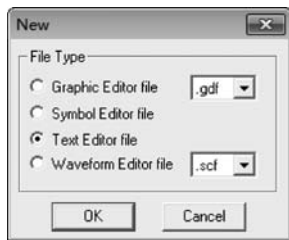


图 3.27 新建文本文件窗口

(2) 单击 OK 按钮,出现如图 3.28 所示的窗口。

(3) 在文本编辑器中输入 10 位二进制计数器的代码,如图 3.29 所示。

2. 保存并检测错误

(1) 选择 File→Project→Save&Check 命令,或单击工具栏中的“保存”按钮,当前的设计被保存,编译器窗口被打开,启动网表提取器模块检查设计文件的基本错误。检查完毕,会弹出“错误与警告信息”对话框,如图 3.30 所示。

(2) 如果“错误与警告信息”对话框显示无错误和警告,则单击“确定”按钮关闭对话框。如果设计有错误或警告,选择菜单栏命令 MAX+plus II →Message Processor 打开消息处理窗口,在其中获取并定位错误信息,然后在设计输入窗口中修改错误,直至检查

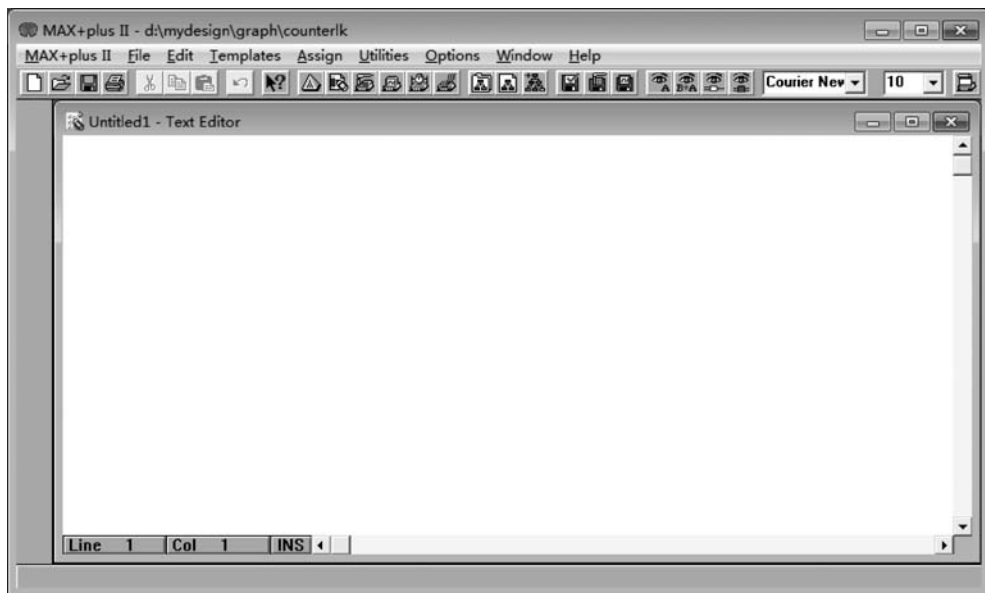


图 3.28 文件编辑器窗口

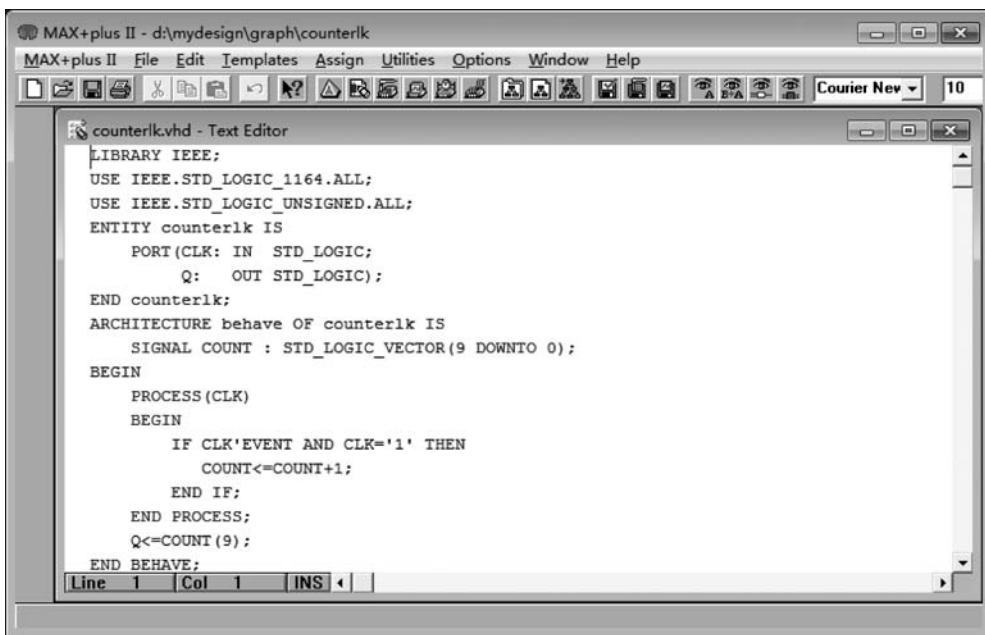


图 3.29 十位二进制计数器文本

没有错误为止(警告可以忽略)。

(3) 单击编译器窗口右上方的“关闭”按钮关闭编译器,返回图形编辑器窗口。

3. 创建功能模块

在设计无误的情况下,把文本设计文件创建成一个元件符号(文件拓展名为 .sym),

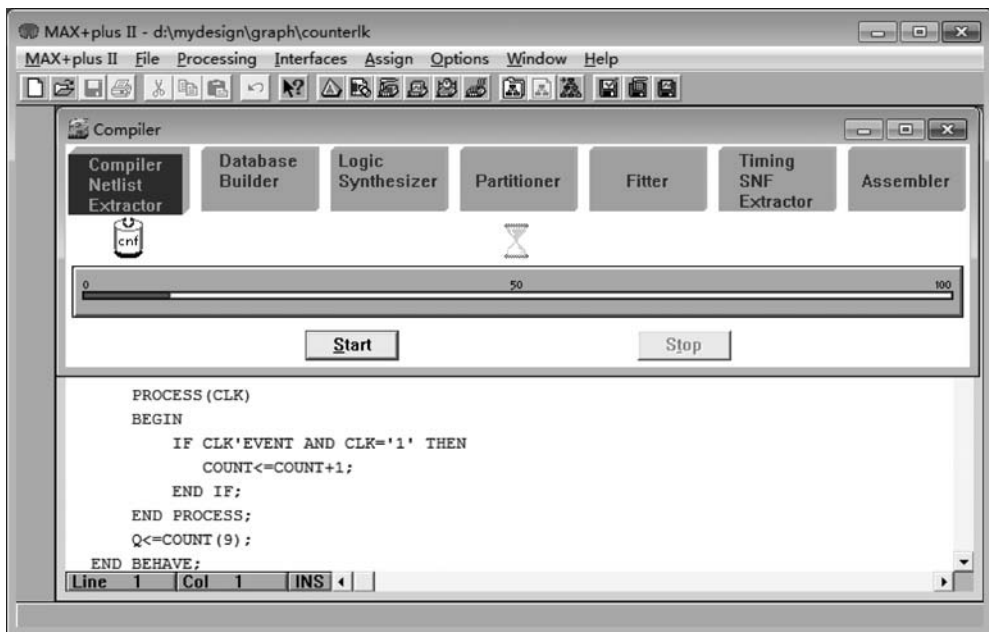


图 3.30 保存并检查窗口

以供其他图形设计文件所调用,这是做层次化设计常采用的方式。

选择菜单命令 File→Create Default Symbol,即把当前设计文件创建成一个同名的元件符号 counterlk. sym,则编译器会自动打开,并弹出元件符号成功产生的信息对话框。

选择菜单命令 File→Edit Symbol,可以打开刚刚创建的元件符号。

3.3.1.4 基于波形设计文件的输入

MAX+plus II 的波形编辑器(Waveform Editor)有两个功能:一个功能是用于时序仿真或功能测试;另一个功能是用于设计输入。在前面的入门实例中,波形编辑器只是用来做时序仿真。本节介绍用波形编辑器进行设计输入。做仿真用时,用波形编辑器创建的文件以“. scf”为后缀;而做设计输入使用时,创建的文件以“. wdf”为后缀。如果把 scf 类型的文件保存成 wdf 文件,并且编辑它,则可以生成一个设计输入文件。

设计者通过指定输入的逻辑电平和输出的逻辑电平来创建波形输入文件。波形输入文件适合于已完全确定了输入与输出之间的时序关系的数字逻辑设计,如状态机、计数器、寄存器等。

下面使用波形图来描述一个十二进制计数器。

1. 创建设计输入文件

1) 创建新文件

打开 MAX+plus II 设计管理器窗口,单击工具栏上的“新建”按钮,选择 Waveform Editor file 选项,在其下拉列表框中选择“. wdf”,如图 3.31 所示。

单击 OK 按钮,产生一个未命名的波形文件,如图 3.32 所示。

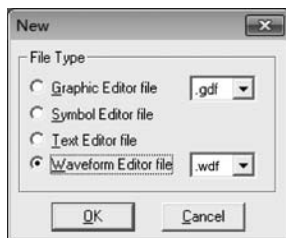


图 3.31 新建波形文件窗口

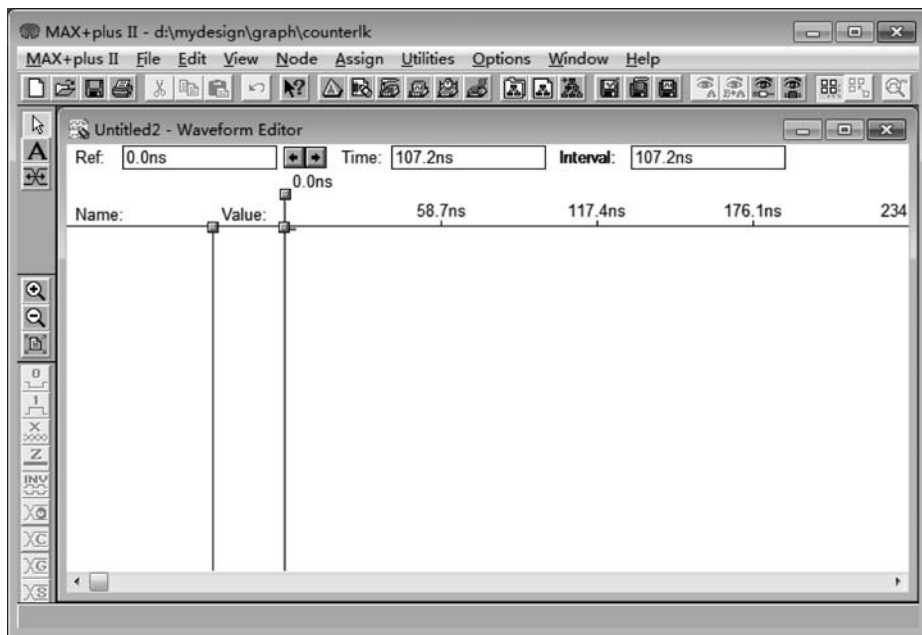


图 3.32 波形编辑窗口

单击工具栏上的“保存”按钮,或者选择 File→Save 命令,出现保存文件窗口,文件命名为 counter12.wdf,文件扩展名选择为.wdf,如图 3.33 所示。



图 3.33 保存波形设计文件

单击 OK 按钮,出现如图 3.34 所示的窗口。

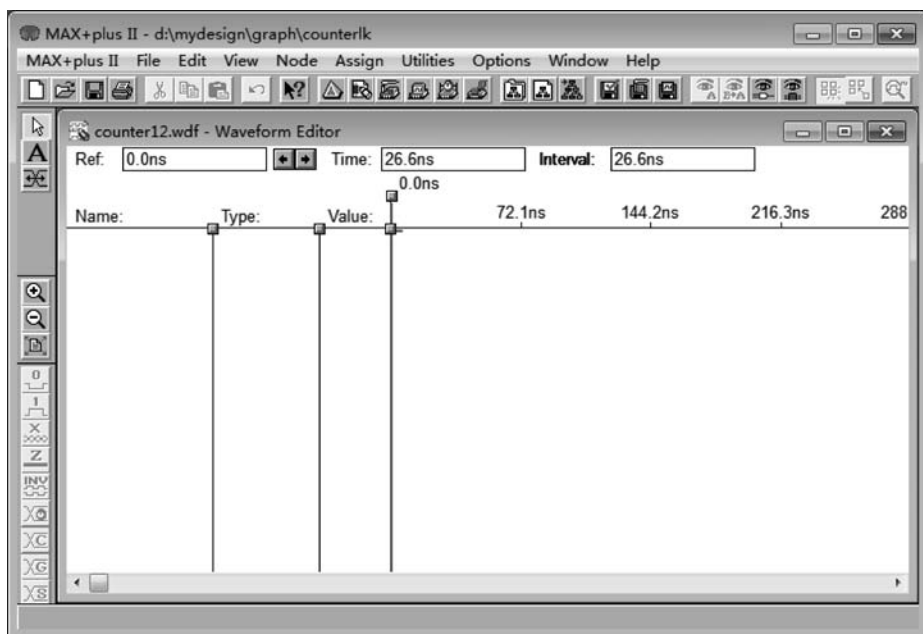


图 3.34 波形设计文件

2) 设定元件

选择 Assign→Device 命令,出现如图 3.35 所示的窗口。选择 ACEX1K 系列的芯片,Devices 可以选择 AUTO,让软件自动匹配具体元件。也可以选择一个确定的型号,这里以 EP1K30QC208-1 为例。单击 OK 按钮,完成选择。

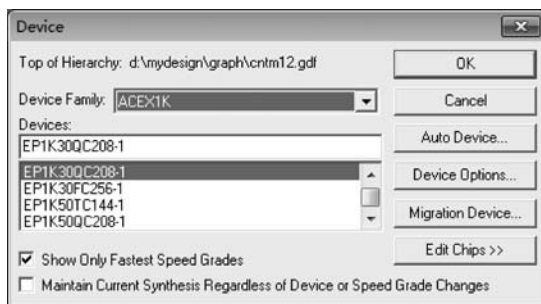


图 3.35 设定元件

2. 创建输入、输出节点和隐埋节点

由状态转换关系,可定义以下节点:输入节点 CLK 和 EN,隐埋节点 STATE,输出节点有 8 个,为 Q[7..0]。其中,输出节点的逻辑电平随着 CLK 变化而变化,EN 使能模块的节点,STATE 指示系统当前时刻所处的状态。下面就在波形编辑器中创建上述节点。

(1) 在波形编辑器中 Name 区双击,或者选择 Node→Insert Node 命令,出现如图 3.36

所示的对话框。

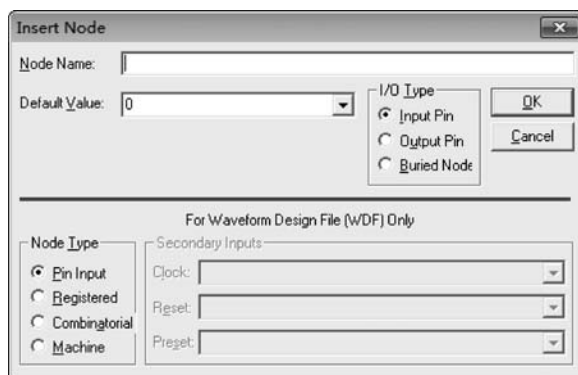


图 3.36 插入节点

在节点名称中输入 CLK,I/O 类型选择输入引脚,节点类型选择输入引脚,单击 OK 按钮确认。

(2) 按照前面的方法和选项创建输入节点 EN。

(3) 按照前面的方法打开插入节点窗口,在节点名称中输入 state。

(4) 按照前面的方法打开插入节点窗口,插入输出节点。

全部节点创建完毕后,波形编辑器如图 3.37 所示。

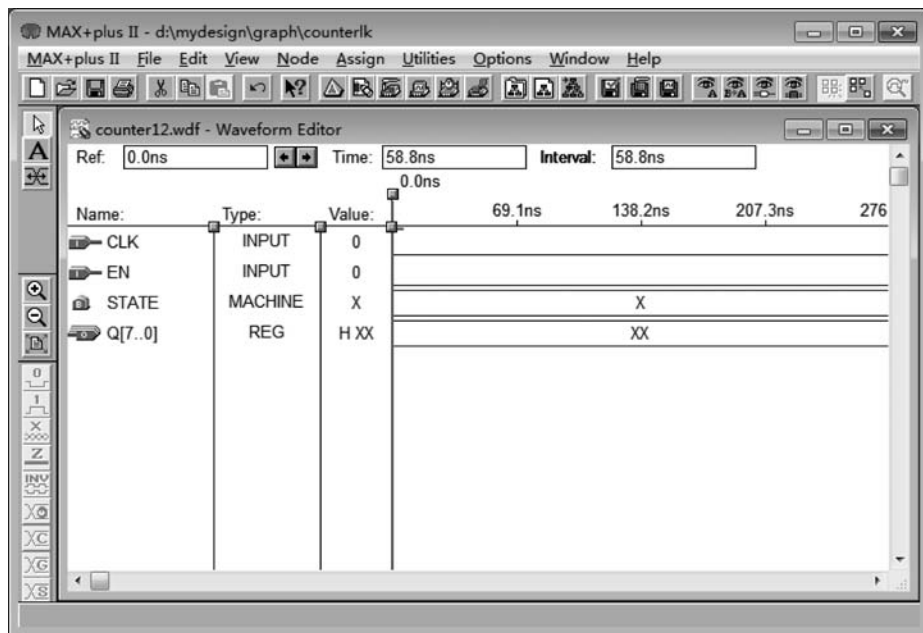


图 3.37 创建节点后的波形文件

3. 设置结束时间、栅格尺寸和显示方式

结束时间是波形设计的总共时间,栅格尺寸是输入信号中时钟信号变化的最小时间

单位,对这两个参数的设置是波形设计中的基本要求。

选择 File→End Time 命令,出现如图 3.38 所示的对话框,在对话框的文本框中输入结束时间 15.5s。

选择 Options→Grid Size 命令,出现如图 3.39 所示的对话框,在对话框的文本框中输入栅格尺寸 250.0ms。

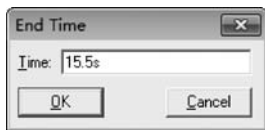


图 3.38 设置结束时间



图 3.39 栅格尺寸对话框

选择 Options→Show Grid 命令,则在波形区会显示出栅格,菜单命令如图 3.40 所示。

选择 View→Time Range 命令,在显示时间范围对话框中,输入起始时间和终止时间,则波形区就会显示这个范围内的波形,范围外的将不显示。

4. 设计节点波形

1) 设计 CLK 波形

因为 CLK 是时钟信号,对于时钟信号,波形编辑器提供了专门的设计方法。单击 CLK 节点,选择 Edit→Overwrite→Clock 命令,或在右键快捷菜单中选择 Overwrite→Clock 命令,则会出现如图 3.41 所示的窗口。



图 3.40 显示栅格

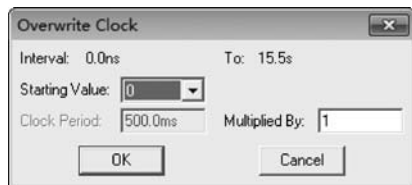


图 3.41 写时钟信号

2) 设计 EN 波形

EN 信号是使能信号,高电平有效。输出节点波形完成后如图 3.42 所示。

3) 设计隐埋节点波形

隐埋节点为 STATE,是状态机信号,指示了系统所处的状态。系统共有 12 个状态,用英文的序数表示: S0、S1、S2、S3、S4、S5、S6、S7、S8、S9、S10 和 S11。时钟信号的上升沿到来,则系统的状态随之改变,系统就在这 12 种状态之间转换。

可以通过上述的状态机赋值按钮对隐埋节点进行赋值。输出节点的状态随着隐埋节点的变化而变化。所有信号赋值完之后的波形图如图 3.43 所示。

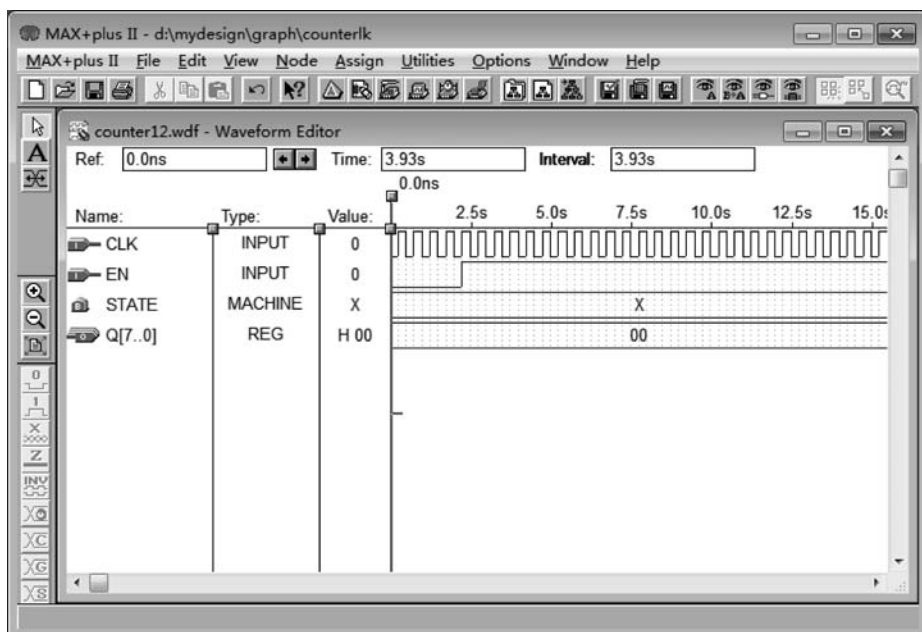


图 3.42 输入节点波形

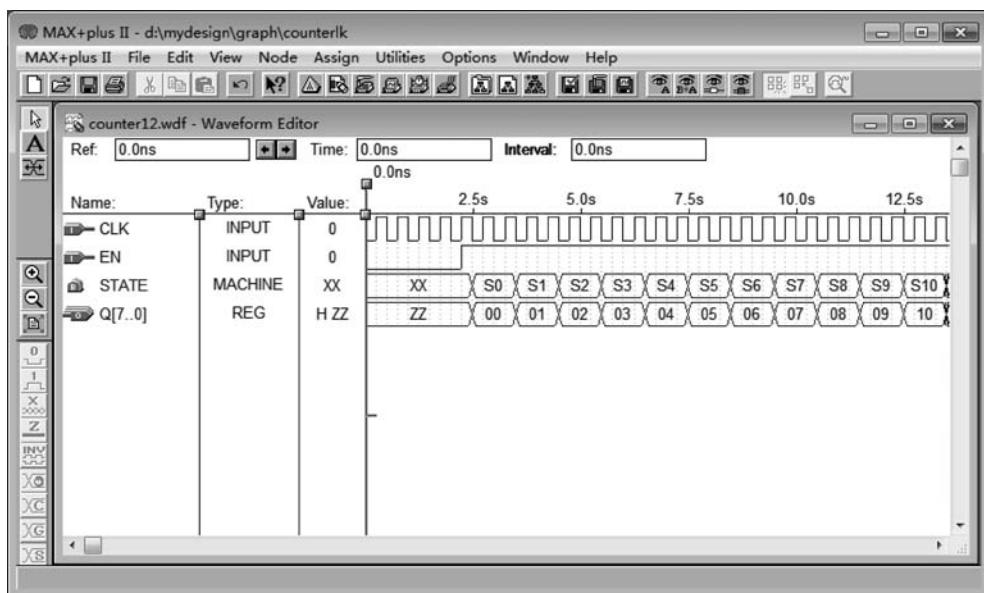


图 3.43 设计完成的波形编辑器

5. 保存并检查错误

选择 File→Project→Save&Check 命令,或单击工具栏中的“保存”按钮,当前的设计被保存,编辑器窗口被打开,启动网表提取器模块检查设计文件的基本错误。检查完毕,会弹出“错误与警告信息”对话框。

3.3.2 设计编译

3.3.2.1 编译选项设置

选择 MAX+plus II → Compiler 命令,弹出的窗口如图 3.44 所示。

编辑器由多个功能模块组成:

- 网表提取模块:用于生成设计的网表文件。
- 数据库建库模块:用于建立描述整个设计的数据库。
- 逻辑综合模块:用于对设计进行逻辑综合和优化。
- 逻辑分割模块:用于对设计进行逻辑分割。
- 逻辑适配模块:将已通过逻辑综合的设计映射到所选元件中。
- 时序仿真网表生成模块:生成用于时序仿真的各种文件。
- 装配模块:生成用于元件编程下载的各种文件。

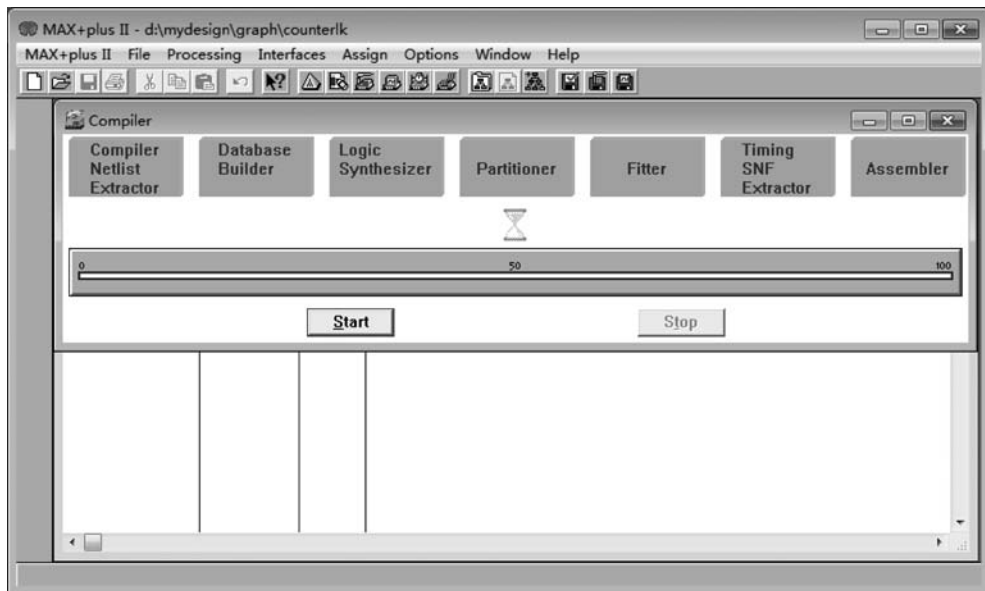


图 3.44 编译器图示

单击 Start 按钮开始编译。编译完成后如图 3.45 所示。

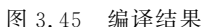
3.3.2.2 引脚的锁定

将设计文件中的输入、输出端口映射到所选元件指定引脚的过程称为引脚锁定。有两种锁定引脚的方法:

- 编译前锁定:在设计文件编译前,通过“引脚锁定”对话框来实现引脚的锁定。
- 编译后锁定:在平面布局编辑器上通过编辑适配结果来修改引脚锁定。

引脚锁定的步骤如下:

(1) 打开“引脚锁定”对话框。在 MAX+plus II 命令栏中选择 Assign → Pin/



Pin/Location/Chip

Top of Hierarchy: d:\mydesigns\graph\cncntm133.gdf

Node Name:

Chip Name:

Chip Resource

☒ Pin: Pin Type (Only for Special Cases):

☐ LC/IO/EC: ☐ Row:

☐ LAG/EAB: ☐ Column:

☒ Anywhere on this Chip

OK Cancel Search... Assign Device... Show Buried Assignments

Existing Pin/Location/Chip Assignments:

Sort By

☒ Node Name ☐ Assignment

Add Delete

(2) 选择输入节点 CLK, 节点名称(Node Name)文本框中将显示要分配的节点。例如, 如果要分配输入点 CLK, 单击 Search 按钮, 弹出如图 3.47 所示的对话框。

现在,按照资料把节点 CLK 分配给引脚 83。选择 Pin 选项,单击下拉按钮,选择引脚 83,如图 3.49 所示。

然后,单击 Add 按钮,把分配的结果添加到 Existing Pin/Location/Chip Assignments 列表中,如图 3.50 所示。

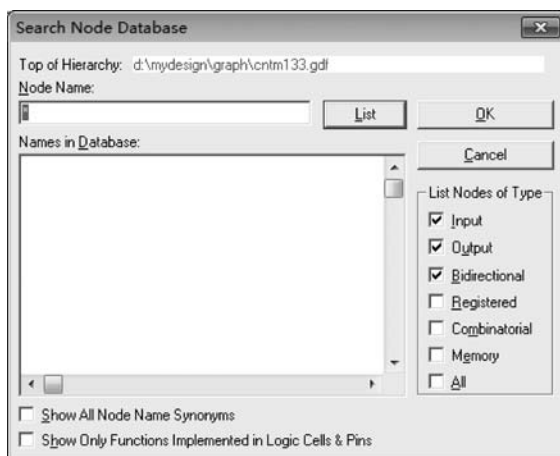


图 3.47 查询节点数据库

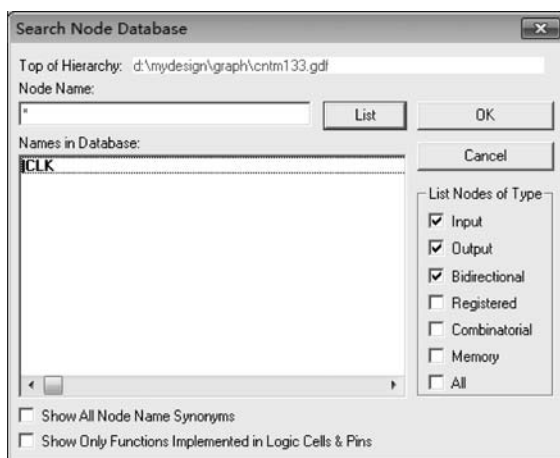


图 3.48 Search Node Database 对话框

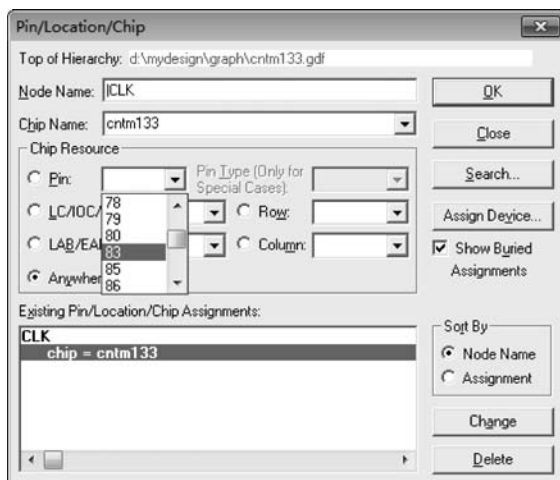


图 3.49 选择引脚 83

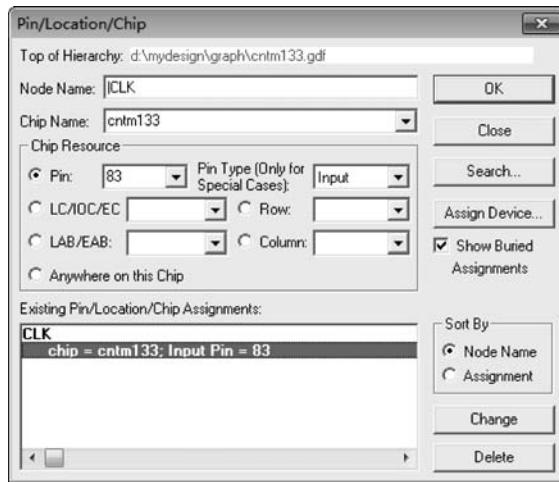


图 3.50 已存在的引脚分配

按照文档要求继续完成余下的引脚分配即可。