

实验一

实验目的:

1. 初步熟悉 MAX+PLUS II 软件。
2. 熟悉掌握 CPLD 的开发流程。
3. 初步熟悉数字实验系统。

实验内容:

1 用原理图输入的办法设计 1~2 个简单组合逻辑电路, 如 3 输入与门、异或门等等。

要求: 走完 PLD 开发的全部流程, 包括原理图输入、设置器件管脚、设置逻辑综合选项、编译、仿真、定时分析以及下载到实验板进行验证。注意仿真的测试矢量要尽可能充分。

验证: 利用实验板的 LED 显示进行验证。实验板 LED 使用说明见后面。

2 用 VHDL 语言设计 1~2 个简单的组合逻辑电路, 不能与 1 相同。

要求: 与 1 相同。

验证: 与 1 相同。

3 自己随意发挥。

实验步骤:

● 原理图方式输入进行简单的组合逻辑电路设计

我们按步骤用原理图输入一个简单的设计: 一个 2 输入与非门和异或门

1 在计算机 C: 盘的 Max2Work 子目录下, 开设自己的目录, 以后的设计文件将放在此目录中。

2 打开并生成一个新的图形设计文件:

File 菜单下, 选 New 命令, 在 File Type 对话框中选 Graphic Editor file, 按 **OK** 后, 出现一个新的图形设计文件窗口, 文件名为 Untitled1, 在 File 菜单下, 选 Save As 命令, 将此文件存在自己的目录中, 并起名为 myfirst.gdf。

或者直接由 MAX+PLUS II 菜单下打开 Graphic Editor, 然后将 untitled1 保存为自己的文件。

3 设定 myfirst.gdf 为当前的 Project。

在 File 菜单下, 选 Project 子菜单, 选 Set Project to current file 命令。

4 开始编辑设计文件: 输入元件

首先输入一个二输入与非门: 在 Symbol 菜单下, 选 Enter Symbol 命令或在编辑窗口中, 双击鼠标左键, 这时出现 Enter Symbol 对话框, 这时在 Symbol Libraries 的列表框中一般有 5 个目录选项:

你的设计文件所在目录

你自己生成的元件

c:\maxplus2\max2lib\prim	基础元件（门电路，输入输出管脚等）
c:\maxplus2\max2lib\mf	中小型元件库（TTL 器件等）
c:\maxplus2\max2lib\mega_lpm	大型模块库，参数化模块库
c:\maxplus2\max2lib\edif	EDIF 标准格式库

选定目录，在 Symbol Files 列表框中将列出此目录中的所有元件。我们需要的二输入与非门在 Prim 目录中，在 Symbol Libraries 中选定 c:\maxplus2\max2lib\prim 目录后，在 Symbol Files 中选定 nand2（也可以直接输入 nand2 选择该器件），在 Symbol Name 编辑框中出现你选的 nand2，按 OK 后，二输入与非门即显示在 myfirst.gdf 文件编辑窗口中。用同样的方法，输入一个异或门。

5 在编辑窗口中操作元件：

- 选择元件： 在元件上单击鼠标左键，元件被红框包围，即表示选中。
- 移动元件： 用鼠标点在元件上，按住左键，拖动鼠标，即可随意移动元件。
- 删除元件： 选择元件后，按 Delete 键即可。
- 其他操作： 在元件上单击鼠标右键，将出现一个菜单，选择不同的命令即可。

6 缩放窗口

在 View 菜单下选择各种缩放命令或者窗口左边按钮中的两个放大镜，观察效果，选择适当的比例。

7 输入管脚

按输入元件的方法输入 *input* 元件，此元件也在 c:\maxplus2\max2lib\prim 中。这时管脚的缺省名字是 PIN_NAME，为了更改名字，用鼠标左键在 PIN_NAME 单击一下，当 PIN_NAME 被红框包围后，直接敲入想要的名字如 a，然后，用键盘的左右箭头键和 Delete 键，删除其他字符即可。

同样，我们输入另一个输入管脚，并命名为 b，还有两个输出管脚：x, y.

这时，我们在窗口中有五个元件，一个二输入与门，一个二输入或门，两个输入管脚，两个输出管脚。在下面的操作中，我们将按要求，用线将他们连接起来。

8 连接元件

当鼠标移到元件的管脚上或线端上时，鼠标的光标变成十字，按下左键，拖动鼠标，直到另一个元件的管脚或线端上，这样相应的管脚就被连接起来。

也可以从元件的管脚上引出线后，对此线加一个名字，方法是在线上用鼠标单击一下，当此线变红被选中后，直接敲入名字，具有同样名字的线在逻辑上就连在一起了。

当所有的连线连接完毕后，设计的输入就完成了。

9 功能编译（如不进行功能仿真，则可以由 8 直接跳到第 12 步）

在 Max+Plus II 菜单下，选 Compiler 命令，或按工具栏上的编译快捷按钮，出现编译的界面。选择 Processing 菜单，如果当前已经是 Functional SNF Extractor 被选中，则不做操作；如果是 Timing SNF Extractor 被选中，则用鼠标点击它一次，则切换到 Functional SNF Extractor 被选中的情况。此时进入功能编译状态，编译器显示出编译设计的各个步骤：Netlist Extractor, Database Builder, Functional SNF Extractor 按 **Start**，开始编译。

如果编译不成功，将出现一个信息处理窗口（Message—Compiler），中间将列出各项错误，你可以用 Locate 来定位错误的出处，用 Help on Message 得到错误出现的原因和解决错误的方法。

如果编译成功，则可以进行功能仿真。

10 编辑生成仿真输入文件：myfirst.scf

在 File 菜单下，选 New 命令，在 File Type 对话框中选 Waveform Editor file，按 **OK** 后，出现一个新的波形文件窗口，文件名为 Untitled1，在 File 菜单下，选 Save As 命令，将此文件存在自己的目

录中，并起名为 myfirst.scf。文件的确省时间长度为 1 us,如果需要更长的仿真时间，在 File 菜单下，选 End Time...命令，直接输入长度即可。

首先加入节点，包括各个输入管脚，输出管脚和感兴趣的内部节点。在 Node 菜单下，选 Enter node from SNF 命令，出现一个对话框，按 **List** 后，在左边的 Available Nodes&Groups 列表框中将列出输入节点，输出节点和内部节点的名字，用鼠标左键可以将输入节点，输出节点和想观察的内部节点选入右边的列表框中，按 **OK** 键后，myfirst.scf 文件中，将出现选择的各个节点的波形，这时，输入节点的波形为低电平，输出节点为不定。

然后，按设计要求编辑输入节点的波形，编辑波形时，一般以一个时间间隔（Grid）为单位来进行，这个时间间隔可以用 Options 菜单下的 Grid Size...命令修改，当然也可以不按间隔来编辑波形，只要关闭 Options 菜单下的 Snap to Grid 即可。

编辑波形时，开始要选定编辑的范围，用鼠标左键单击节点名字，此节点的整个范围被选中，也可以在节点的某个时间点按下用鼠标左键，拖动鼠标，在另一时间点放开鼠标左键，两个时间点之间的范围被选中，选中的区域为黑色。

选中范围后，用 Edit 菜单下的命令可以编辑波形，一般使用窗口左边的快捷按钮，如按 **1** 键，则选中范围中的信号波形变为高电平。常用的快捷按钮还有低电平改写，三态改写，不定电平改写，反相，时钟生成，记数序列生成，总线信号生成等。

输入节点的波形应能覆盖设计的所有范围，这样才能保证仿真没有遗漏。

为了仿真 myfirst.gdf 设计，我们按输出信号 x, y 的逻辑要求，我们编辑信号 a 和 b，使之出现所有可能的四种组合。

11 功能仿真

在 Max+Plus II 菜单下，选 Simulator 命令或按工具栏上的仿真快捷按钮，出现功能仿真 (Functional Simulation) 的窗口。

窗口中，显示仿真输入 (Simulation Input :) 是 myfirst.scf 文件。按 **Start** 后，开始仿真。

如果是第一次进行仿真或 myfirst.scf 文件不存在，仿真将无法进行，这时需要先编辑生成仿真激励文件：myfirst.scf。

如果 myfirst.scf 文件已存在，仿真结束后，输出信号的波形将写入到仿真激励文件 myfirst.scf 中。我们可以打开此文件，观察输出波形是否满足我们的设计要求。

如果输出波形不满足要求，说明设计存在逻辑问题，我们要根据错误的情况，分别回到前面的相应的步骤，进行修改，修改完成后，必须重新经过编译，功能仿真，直到满足设计要求。

如果满足设计要求，则说明前面的设计从功能上来说是正确的，我们接下来可以进行时序编译/仿真。

注意到功能仿真中输出信号是很“工整”的，没有毛刺，冒险。

12 选择时序编译

打开编译器，在 Processing 菜单下，鼠标再点一次 Functional SNF Extractor（参照 9 中说明）。

参照 9 中说明进行初编译。（如果设计在最近一次修改后进行过功能/时序编译，则该步省略）

13 选定器件

在 Assign 菜单下，选择 Device...命令，出现一个对话框，我们在 Device Family: 中选 FLEX8000，在 Device 中选 EPF8282ALC84-4（如果选择了”show only fastest speed grades”，则不会出现该型号，选择 EPF8282ALC84-2 也可以。）

14 设置管脚

在 Assign 菜单下，选择 Pin/Location/Chip...命令，出现 Pin/Location/Chip 对话框。点击 Search 按钮，出现 Search Nodes Database 对话框。在该对话框中，点击 List，然后在 Name in Database 框中选

择欲分配的管脚。点击 OK 返回 Pin/Location/Chip 对话框后，在 Pin 下拉菜单中选择相应的管脚号，然后点 ADD 按钮，则在下面的 Existing Pin/Location/Chip Assignments 框中出现与该管脚设置相对应的一行，此时完成了该管脚的设置。所有管脚设置完后选择 Ok 返回。

管脚设置也可以在 FloorPlan 中拖动实现或者直接修改.acf 文件实现。

14 其他设置

Assign->Global Project Device Options:

Configuration Scheme 框选 Passive Serial;

Pin----Reserve Tri-State:列表中所有信号都不选中。

Assign->Global Project Logic Synthesis (本例子中不必要)

Global Project Synthesis Style: 选 Normal 或者 Fast

点击 Define Synthesis Style 设置进位链和级联链。

Assign->Global Project Timing Requirements (设置定时要求, 本例子中不必要)

Processing->Design Doctor : 选中有助于发现不好的设计。

Processing->Total Recompile 或者 Smart Recompile

15 编译设计

在 Max+Plus II 菜单下，选 Compiler 命令，或按工具栏上的编译快捷按钮，出现编译的界面，显示出编译设计的各个步骤：Netlist Extractor, Database Builder, Logic Synthesizer, Partitioner, Fitter, Timing SNF Extractor, Assembler, 按 **Start**，开始编译。

如果编译不成功，将出现一个信息处理窗口 (Message—Compiler), 中间将列出各项错误，你可以用 Locate 来定位错误的出处，用 Help on Message 得到错误出现的原因和解决错误的方法。

如果编译成功，信息处理窗口 (Message—Compiler) 中间将显出软件自己选择的芯片。我们可以用鼠标左键双击 Fitter 下的 rpt 图标，打开 myfirst.rpt 文件，查看编译的结果，包括资源消耗情况，管脚分布，逻辑综合后的最底层的逻辑方程等信息。

16 静态时序分析

在 Max+Plus II 菜单下，选 Timing Analyzer 命令或按工具栏上的时序分析快捷按钮，出现时序分析的窗口。

在 Analysis 菜单下，选择 Delay Matrix, 窗口中出现一张表格，按 **Start**后，表格中列出组合逻辑每个输出到每个输入的延时。

在 Analysis 菜单下，选择 Registered Performance, 窗口中出现一个频率计，按 **Start**后，窗口中将列出时序电路中每个时钟的最大的工作频率。

在 Analysis 菜单下，选择 Setup/Hold Matrix, 窗口中出现一张表格，按 **Start**后，表格中列出将列出时序电路中每个输入相对每个时钟的要求的最小输入信号建立和保持时间。

由于当前的设计是一个组合设计，所以后两项分析不能进行。作完静态时序分析后，我们进入时序仿真。

17 编辑生成仿真输入文件：myfirst.scf

根据实际情况，继续沿用 10 中功能仿真的波形输入文件或者参照 10 中编辑新的仿真输入文件。

18 时序仿真

在 Max+Plus II 菜单下, 选 Simulator 命令或按工具栏上的仿真快捷按钮, 出现仿真分析 (Timing Simulation) 的窗口。

窗口中, 显示仿真输入 (Simulation Input:) 是 myfirst.scf 文件。按 **Start** 后, 开始仿真。

仿真结束后, 输出信号的波形将写入到仿真激励文件 myfirst.scf 中。我们可以打开此文件, 观察输出波形是否满足我们的设计要求。

如果输出波形不满足要求, 我们要根据错误的情况, 分别回到前面的相应的步骤, 进行修改, 修改完成后, 必须重新经过编译, 时序分析, 时序仿真的过程, 直到满足设计要求。

如果满足设计要求, 则说明前面的设计是正确的, 我们可以按 myfrist.rpt 中的器件和管脚分布来设计电路印制板 (PCB), 在一般情况下, 由于封装, 速度, 容量的考虑, 加上价格, 购买方便性等因素, 我们的要采用的芯片不一定与软件自己选择的器件一致, 而且为了 PCB 的设计方便, 我们必须选定一个芯片, 包括封装形式, 速度级别, 而且要将信号设定到需要的器件管脚上 (必须先选定芯片)。然后重新经过编译, 时序分析, 时序仿真的过程, 直到满足设计要求。

19 查看 Floorplan

在 Max+Plus II 菜单下, 选 FloorPlan Editor 命令或按工具栏上的 Floorplan 快捷按钮, 出现 Floorplan 窗口, 选择显示当前的设置图 (Layout->Current Assignments Floorplan), 则显示出当前的 Floorplan。编译前没有设定的管脚将会出现在上方的 Unassigned Nodes & Pins 框中。从框中, 用鼠标左键按住待设定的节点, 拖动鼠标到下面的器件平面图中, 在所需要设定的器件引脚处放开, 此时该管脚将变蓝, 则我们已经成功地将该管脚设定到芯片的某一引脚处。陆续的可以将其他未设定的管脚设定好。

如果重新进行了设定, 则需要重新经过编译才可能生效。此时需要重新进行时序分析, 时序仿真的过程, 确认仍能满足设计要求。打开 myfirst.rpt 文件, 确认管脚分布与单元电路实验板的电路图一致。

完成以上过程后, 软件生成的 myfirst.pof 文件就包含了目标器件的配置信息, 此配置信息用于下载到 PLD 芯片上加电进行硬件功能验证。

20 下载程序

确认 CIC-310 试验系统的下载板已经插好、与计算机用串口线已经连好后, 打开电源。运行桌面上的 DNLD102.exe 程序。出现 CPLDEXP-Sinosonic 面板。

在 CPLDEXP 面板上, 选择 Options->Com Port->选择计算机与下载板相连所使用的串口, 一般是 COM1。Options 下的 Reset Com 用于对串口进行复位, 当串口工作不正常时可以使用该选项。Options 菜单下的 Initial EEPROM 将擦除下载板上存储的所有下载程序——慎用。

在面板上的 Directories 框和 Files 框中选择所下载设计的目录以及下载文件 (与项目名同)。选中下载文件后选择 Add。则该文件出现在左边的 SEEPROM 框中。在该框中再次选中该文件, 点击 Config 则将该程序下载到 PLD 芯片中。

面板上各按钮功能说明: Files 框下面的 Config 是直接下载到 PLD 器件, 而不保存到 SEEPROM 中去; Add 用于将所选文件写入 SEEPROM; Del 用于从 SEEPROM 中删除所选文件; Act 是将 SEEPROM 中某个文件激活 (文件被激活的意思是下载板上单片机被复位或者系统初始上电时该文件会被自动下载到 PLD); Dir 显示 SEEPROM 中保存的下载文件; Dir 旁边的 Config 是下载 SEEPROM 中的选定文件。

21 验证设计

通过试验板上的按键及 LED 等验证对于不同输入逻辑电平得到的输出逻辑电平是否满足要求。

如果验证成功, 则我们就完成了一个 PLD 设计。

● 用 VHDL 语言设计 1~2 个简单的组合逻辑电路

与前面差别仅限于设计的输入方式和工具不一样。其他过程都一样。可参考前面的各个过程，但以下几步有所变化

2 打开并生成一个新的文本设计文件：

File 菜单下，选 New 命令，在 File Type 对话框中选 Text Editor file，按 **OK** 后，出现一个新的图形设计文件窗口，文件名为 Untitled1，在 File 菜单下，选 Save As 命令，将此文件存在自己的目录中，并起名为 myfirst.vhd。

或者直接由 MAX+PLUS II 菜单下打开 Text Editor，然后将 untitled1 保存为自己的文件。

4~8 编辑源程序文件。

小技巧：

Templates->VHDL Template 可以选择 VHDL 语言的模板。

写完后 File->Save & Check 可以在提示信息的帮助下找到并修改语法错误。

CIC-310 实验板：

本试验可能需要使用试验板上的 LED 作为验证显示用。试验板上有两排 LED，每个都连接与 8k 系列的 PLD 芯片的某个引脚，其引脚号与 LED 旁边的编号对应（如 P55 对应芯片之 55 脚）。其中上面的一排 LED 所连接的 PLD 引脚可以作为专门的输出引脚（因为没有其他源加在这些引脚上），而下面的一排 LED 所连接的 PLD 引脚可以作为输入或者输出引脚使用，作为输入引脚时，其电平有相应的拨位开关确定。

试验板右下方的三排每排 8 个的拨位开关是用来产生逻辑输入电平的，拨位开关拨上时对应引脚输入通过上拉电阻被上拉为高电平，拨下时则通过下拉电阻被拉到低电平。左边的 2 个拨位开关同时与下面的一排 LED 并联。每个拨位开关下面的标号（如 P01）表示该开关所连接的芯片引脚。