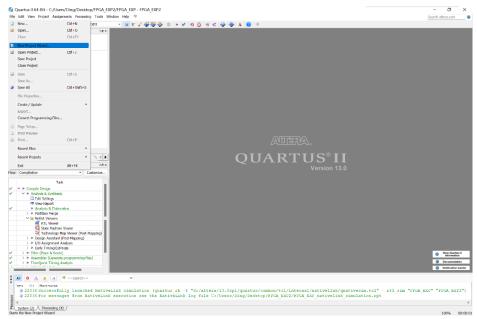
实验一: ALTERA FPGA 开发入门

一.实验流程:

1. 工程创建及配置:

打开 Quartus 软件,在"File"菜单中选择"New···"命令,然后选择"New Quartus II Project"即可创建项目,或者在"File"菜单中直接选择"New Project Wizard",也可进行项目的创建。如图所示:



之后会弹出工程的配置菜单:

- (1) 此处注意顶层实体名应与项目名称严格一致,否则编译会报错。
- (2) 之后按照教程配置正确的"Family & Device Settings", 注意选择正确的设备, 否则后续无法正常配置管脚。如果后续需要更改,可以在上方缩略图菜单栏中,点击"Device"进行修改。



(3) 设置仿真工具,在"Tool Type"中的"Simulation"栏选择仿真工具为 ModelSim, 硬件描述语言为 VHDL,项目根据此设置可以选择正确的编译器。

2. 添加文件和代码

单击"File"菜单,选择"New···"命令,在弹出的 New 窗口中选择"VHDL File"项,添加文件。同样,可以直接在菜单栏中点击文件图片,也可唤出菜单;仿真代码的添加步骤和设计代码相同。在按照语法添加完代码后,保存文件到工程路径下。注意文件名应与设计文件的实体名相对应。

然后对代码进行编译。单击"Processing"菜单,选择"Start"子菜单中的"Start Analysis & Synthesis"命令, 完成功能仿真前的分析和综合。(或者直接点击上方缩略图菜单栏中的对应图标,同样可以实现该功能)。

如果代码和工程设置均正确,则会弹出"Analysis & Synthesis was successful"窗口,且

在"Messages" 窗口显示"Quartus II 64-Bit Analysis & Synthesis was successful. 0 errors, 0 warnings …",如果报错,在下方 Message 栏查看报错信息,双击某处错误会自动跳转到对应代码,方便查找问题。重复以上过程,直至编译成功。

3. 功能仿真

在添加并保存过仿真代码后,对仿真工具 ModelSim 进行设置:

单击"Tools"菜单,选择"Options···"命令,在弹出"Options"窗口的左侧 单击"General"下的 "EDA Tool Options"。在对应 ModelSim 栏选择可执行文件的路径。

然后配置仿真任务:

单击"Assignments"菜单,选择"Settings···"命令,在弹出"Settings"窗口的左侧单 击"EDA Tool Settings"下的"simulation",配置 simulation 参数。并新建测试台,在设置中添加对应的仿真文件。

上述配置完成后,即可单击"Tools"菜单,选择"Run Simulation Tool"子菜单中的"RTL Simulation"命令来启动 Modelsim 实现功能仿真。(同样,可以点击快捷菜单栏中的缩略图唤出此功能)。

分析仿真结果。

4. 分配管脚

单击"Assignments"菜单,选择"Pin Planner"命令打开"Pin Planner"窗口(同样可以使用快捷菜单栏),并检查对应设备是否正确。查找实验指导书,为 FPGA 的内部节点分配外部管脚。

5. 全编译

单击"Processing"菜单,选择"Start Compilation"命令,实现 Analysis & Synthesis、Fitter 及 Assembler 等完整的全编译过程。

6. 硬件验证

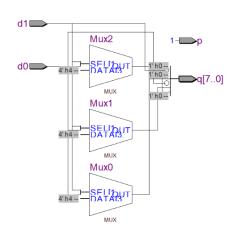
首先单击"Tools"菜单,选择"Programmer"命令,配置下载线。确保各处设置正确后,点击Start,将程序烧写进FPGA。使用实验箱上的硬件电路,验证实验结果是否符合预期的逻辑功能和效果。

二.实验结果:

设计二的仿真时序图:



该时序图表示: 仿真共持续 1200ns, d1,d0 成周期性取'00', '01', '10', '11', 'U0', '0X'这六种状态, q 的各个分量随之发生周期性的变化。 设计二的 RTL 结构图:



设计二对应的功能:

d1,d0 为两个输入端口, p和 q为两个输出端口, 其中 q 共有 8 个分量。d1,d0 对应外部端口 R11, Y11,即对应试验箱上的开关 DIP0 和 DIP1.p 端口对应外部管脚为 E14, 即数码管的片选信号。p 端口被置为 1, 代表选中数码管。实验箱使用的七段码 LED 是共阳极型, a~f和 dp这八个 LED 段。q端口的 8 个分别对应八段 LED 的阴极。当 FPGA 对应的 IO 引脚输出低电平时,对应的七段码 LED 中的 LED 点亮;当 FPGA 对应的 IO 引脚输出高电平时,对应的七段码 LED 中的 LED 熄灭。

根据两个开关输入的四种不同状态,数码管分别显示"5""6""7""8"四个数字。

三.实验中遇到的问题:

1. 程序文件名和实体名未对应, 导致编译报错。

解决办法:首先尝试对文件进行重命名,但是编译仍然会报错,显示无法找到文件。进一步了解后发现,对于该工程来说,其内部组织的文件结构设置中,保留的仍是原文件。在项目设置中,移除原有的文件,重新添加重命名后的文件,问题得到解决。

2. 语法错误.

在编译过程中,发现了多次语法错误,包括忘记符号,某些符号使用了中文输入法等。另外,通过测试,我知道 VHDL 对缩进并没有严格规定,但是为了提高代码的可读性和美观性,我仍然尽量保持了各语句之间的正确缩进形式。

3. Device 设置错误

在建立工程时,由于粗心,我错误的配置了工程的 Devide 类型。在准备为 FPGA 内部节点分配管脚时, 打开 Pin Planner 时, 发现没有显示正确的结构图, 查看上方 device 信息之后, 发现是 Device 配置错误。

解决办法:可以在项目设置中的 Device 中直接修改设置。