

实验二：使用Verilog编写设计32位全加器

目标

1. 使用Verilog编写设计带有超前进位功能的32位全加器。你可以先使用4位全加器练手。
2. 安装并使用Vivado软件对编写的Verilog文件进行仿真。

要求

1. 请不要抄袭，可以与同学讨论，但不要直接抄袭同学的代码和实验报告。
2. 请认真完成实验报告，并在你认为关键的位置插入屏幕截图。
3. 请在截止日期（初定为2024.10.08日23: 55）前将Verilog源代码和实验报告提交至Unicourse+上。

步骤

1、Vivado软件的安装

- Vivado是Xilinx的一款开发套件EDA工具。我们将使用Vivado 2018.3版本完成本次及后续实验。也可以从官网下载免费的WebPACK版本。

1.1 在线安装

- 在 Xilinx-Vivado网页中找到最新版本的Vivado。

根据自己的操作系统选择“Windows Self Extracting Web Installer”或“Linux Self Extracting Web Installer”，在后续的界面中根据引导注册账户及验证后下载安装程序即可。

- 下载后双击运行安装程序
 - 填写你下载时注册的账号
 - 同意用户许可协议
 - 在版本选择界面选择“Vivado HL WebPACK”版本（第一项）
 - 选择设计工具、支持的器件界面维持默认选项即可
 - 选择安装路径及快捷方式设置
 - 确认配置后点击“Install”开始下载安装

1.2 本地安装

- 跟随PDF指引激活安装。

2、使用Vivado软件的Project Mode进行开发实验

2.1 新建工程

- 通过双击桌面快捷方式或开始菜单的“Xilinx Design Tools - Vivado”打开Vivado，在界面上的“Quick Start”下选择“Create Project”。
- 打开新建工程向导后点击“Next”，然后输入工程名称并选择工程的文件位置。“Create project subdirectory”选项表示在指定的存储路径下为项目创建独立的文件夹。工程名称和存储路径中**不能出现中文和空格**，设置完成后选择“Next”。
- 选择“RTL Project”一项，如果在新建工程时不需要添加设计源文件，勾选“Do not specify sources at this time”选项，点击“Next”。
- 向新建工程中添加文件，如果在上一步中勾选了“Do not specify sources at this time”选项则会跳过这一步，推荐勾选“Copy sources into project”选项，将所选的源文件拷贝到工程目录下。点击“Next”后向导提示向工程中添加约束文件，直接点击“Next”即可。
- 选择工程使用的器件，使用默认的即可，点击“Next”。
- 检查无误后点击“Finish”完成新建工程。

2.2 添加设计文件

- 在左侧导航窗口的“Project Manager”下点击“Add Sources”，或者点击“Sources”窗口下的加号按钮添加源文件。
- 选择“Add or create design sources”来添加或新建Verilog源文件，点击“Next”。
- 如添加已有设计文件或者添加包含已有设计文件的文件夹，选择“Add Files”或者“Add Directories”。如创建新的设计文件，选择“Create File”，并在窗口中选择类型、名称和位置，名称和位置中**不能出现中文和空格**。添加完成后点击“Finish”。
- 如果之前创建了新文件，需要进行模块端口设置，在“Module Definition”中的“I/O Port Definitions”，输入设计模块所需的端口，并设置端口方向，如果端口为总线型，勾选“Bus”选项，并通过“MSB”和“LSB”确定总线宽度。完成后点击“OK”。端口设置也可以在编辑源文件时完成，即可以在这一步直接点“OK”跳过。

2.3 添加激励文件

- 在左侧导航窗口的“Project Manager”下点击“Add Sources”，或者点击“Sources”窗口下的加号按钮添加源文件。
- 选择“Add or create simulation sources”来添加或新建Verilog源文件，点击“Next”。
- 选择“Create File”，并在窗口中选择类型、名称和位置，激励测试文件的命名通常为“待测试模块名_tb(test bench)”，比如“adder_tb”。点击“Finish”，之后在弹出的端口设置窗口中直接点击“OK”跳过。
- 在“Sources”窗口中的“Simulation Sources”下找到新创建的激励文件，双击并在右侧的窗口中对其进行编辑。
- 下面我给出一个本实验的激励文件，为了使激励文件工作，我们统一本次实验加法器的接口为

```
module adder(input [31:0] a,input[31:0] b,output[31:0] sum);
```

无论你的加法器是32位还是4位，我们的输入输出都使用32位。

其中a与b是两个代表输入的32位总线，sum为代表输出结果的32位总线，不需给出运算结束后的进位结果。

对于四位全加器，你需要给第4-31位输出赋值为零。也就是

```
assign c[31:4] = 28'b0;
```

同时，你需要在激励文件中取消定义宏 ADDER_32BIT。

2.4 进行仿真

- 在左侧导航窗口的“Simulation”下点击“Run Simulation”，选择“Run Behavioral Simulation”，这时会编译你的设计文件和激励文件，如果没有错误会进入仿真界面，否则你需要根据Log窗口中的错误信息对你的设计文件或激励文件进行修改。
- 在仿真界面，可以通过左侧“Scope”一栏中的目录结构定位到想要查看的module内部信号，在“Objects”对应的信号名称上右击选择“Add To Wave Window”，将信号加入波形图中。仿真器默认显示I/O信号。
- 可通过选择工具栏中的选项来进行波形的仿真时间控制。
- 在波形显示窗口上侧是波形图控制工具，由左到右分别是：查找、保存波形配置、放大、缩小、缩放到全显示、缩放到光标、转到时间0、转到时间的最后、前一个跳变、下一次跳变、添加标记、前标记、下一个标记、交换光标。
- 可通过右键选中信号在“Radix”选项中改变信号的显示形态。同时当改变光标位置（拖动黄线）时在“Object”窗口中相应信号的值也会改变，也可以右键选中信号在“Radix”选项中改变信号值的显示方式。

2.5 进行分析

- 对波形图进行分析，确定模块的功能符合预期，进行截图并完成实验报告。

温馨提示

- 在激励文件中两个输入分别使用了若干随机值，可以充分验证加法器的正确性。实验报告截图时应充分体现这一点。
- 实验报告中可以包含遇到的问题以及解决方法。
- 超前进位的表达式可能十分复杂，你可以选择编写一个程序来帮你生成Verilog文件。