**实验报告**

2019 年 4 月 2 日 成绩：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 姓名 | 於文卓 | 学号 | 17061833 | 班级 | 17052317 |
| 专业 | 计算机科学与技术 | | 课程名称 | 计算机组成原理课程设计 | |
| 任课老师 | 冯建文 | 指导老师 | 冯建文 | 机位号 |  |
| 实验序号 | 2X | 实验名称 | 乘法器/除法器实验 | | |
| 实验时间 | 4月2日 | 实验地点 | 1-225 | 实验设备号 | #18 |
| **一、实验目的与要求** | | | | | |
| 1. 实验目的： 2. 学习掌握如何配置IP核 3. 学习使用IP核构建乘法器 4. 掌握灵活运用Verilog HDL语言进行各种描述与建模的技巧和方法 5. 实验要求：   （1）使用IP核构建乘法器，要求输入大于等于8位  （2）通过板级验证 | | | | | |
| **二、实验设计与程序代码** | | | | | |
| 1. 模块设计说明 2. 设计IP核，过程如下 3. 在New Source页面选择创建IP核选项      1. 在Math Function中选择乘法器      1. 在配置页面选择位宽为8位无符号数      1. 选择乘法器的构成（使用片上逻辑or内部乘法器），这里使用内部乘法器Mults      1. 定义输出的位宽     （2 ）构建顶层模块mul，并调用IP核   1. 实验程序源代码及注释等 2. IP核模块代码 3. -- Interfaces: 4. --    a\_intf 5. --    clk\_intf 6. --    sclr\_intf 7. --    ce\_intf 8. --    b\_intf 9. --    zero\_detect\_intf 10. --    p\_intf 11. --    pcasc\_intf 13. -- The following code must appear in the VHDL architecture header: 15. ------------- Begin Cut here for COMPONENT Declaration ------ COMP\_TAG 16. COMPONENT multiply 17. PORT ( 18. clk : IN STD\_LOGIC; 19. a : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0); 20. b : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0); 21. p : OUT STD\_LOGIC\_VECTOR(15 DOWNTO 0) 22. ); 23. END COMPONENT; 24. -- COMP\_TAG\_END ------ End COMPONENT Declaration ------------ 26. -- The following code must appear in the VHDL architecture 27. -- body. Substitute your own instance name and net names. 29. **------------- Begin Cut here for INSTANTIATION Template ----- INST\_TAG** 30. **your\_instance\_name : multiply** 31. **PORT MAP (** 32. **clk => clk,** 33. **a => a,** 34. **b => b,** 35. **p => p** 36. **);** 37. 顶层模块mul代码 38. `timescale 1ns / 1ps 39. ////////////////////////////////////////////////////////////////////////////////// 40. // Company: 41. // Engineer: 42. // 43. // Create Date:    18:40:13 03/31/2019 44. // Design Name: 45. // Module Name:    mul 46. // Project Name: 47. // Target Devices: 48. // Tool versions: 49. // Description: 50. // 51. // Dependencies: 52. // 53. // Revision: 54. // Revision 0.01 - File Created 55. // Additional Comments: 56. // 57. ////////////////////////////////////////////////////////////////////////////////// 58. module mul( 59. input clk, 60. input [7:0] A, 61. input [7:0] B, 62. output [15:0]P 63. ); 64. multiply demo( 65. .clk(clk), 66. .a(A), 67. .b(B), 68. .p(P) 69. ); 71. endmodule | | | | | |
| **三、实验仿真** | | | | | |
| 1. 仿真代码 2. `timescale 1ns / 1ps 3. //////////////////////////////////////////////////////////////////////////////// 5. module test; 7. // Inputs 8. reg clk; 9. reg [7:0] A; 10. reg [7:0] B; 12. // Outputs 13. wire [15:0] P; 15. // Instantiate the Unit Under Test (UUT) 16. mul uut ( 17. .clk(clk), 18. .A(A), 19. .B(B), 20. .P(P) 21. ); 23. initial 24. begin 25. clk = 0; 26. A = 0; 27. B = 0; 28. end 29. always #10 clk=~clk; 30. initial 31. begin 32. #20;A=1;B=1; 33. #20;A=1;B=2; 34. #20;A=1;B=20; 35. #20;A=1;B=64; 36. #20;A=3;B=3; 37. #20;A=4;B=4; 38. #20;A=5;B=5; 39. #20;A=6;B=6; 40. #20;A=10;B=10; 41. #20;A=14;B=12; 43. end 45. endmodule 46. 仿真波形      1. 仿真结果分析   可以看到当A和B赋值后，clk经过一个上跳沿输入结果到P  当A为1时，P的值就为B的值  当A为3，B为3时，结果为9  当A为4，B为4时，结果为16  A为10，B为10，结果为100  A为14，B为12，结果为168  A为20，B为15，结果为300  仿真结果符合乘法法则，仿真结果正确 | | | | | |
| **四、电路图** | | | | | |
|  | | | | | |
| **五、引脚配置（约束文件）** | | | | | |
| NET "clk" LOC = N17;  NET "A[7]" LOC = V5;  NET "A[6]" LOC = T4;  NET "A[5]" LOC = V6;  NET "A[4]" LOC = T5;  NET "A[3]" LOC = T6;  NET "A[2]" LOC = V7;  NET "A[1]" LOC = R8;  NET "A[0]" LOC = U9;  NET "B[7]" LOC = V10;  NET "B[6]" LOC = R10;  NET "B[5]" LOC = U11;  NET "B[4]" LOC = R11;  NET "B[3]" LOC = U12;  NET "B[2]" LOC = T13;  NET "B[1]" LOC = V14;  NET "B[0]" LOC = T14;  NET "P[0]" LOC = V17;  NET "P[1]" LOC = T16;  NET "P[2]" LOC = V16;  NET "P[3]" LOC = T15;  NET "P[4]" LOC = U14;  NET "P[5]" LOC = R13;  NET "P[6]" LOC = U13;  NET "P[7]" LOC = R12;  NET "P[8]" LOC = V12;  NET "P[9]" LOC = T11;  NET "P[10]" LOC = V11;  NET "P[11]" LOC = T10;  NET "P[12]" LOC = V9;  NET "P[13]" LOC = T8;  NET "P[14]" LOC = U8;  NET "P[15]" LOC = R7;  NET "clk" IOSTANDARD = LVCMOS18 |CLOCK\_DEDICATED\_ROUTE = FALSE;  NET "A[7]" IOSTANDARD = LVCMOS18;  NET "A[6]" IOSTANDARD = LVCMOS18;  NET "A[5]" IOSTANDARD = LVCMOS18;  NET "A[4]" IOSTANDARD = LVCMOS18;  NET "A[3]" IOSTANDARD = LVCMOS18;  NET "A[2]" IOSTANDARD = LVCMOS18;  NET "A[1]" IOSTANDARD = LVCMOS18;  NET "A[0]" IOSTANDARD = LVCMOS18;  NET "B[7]" IOSTANDARD = LVCMOS18;  NET "B[6]" IOSTANDARD = LVCMOS18;  NET "B[5]" IOSTANDARD = LVCMOS18;  NET "B[4]" IOSTANDARD = LVCMOS18;  NET "B[3]" IOSTANDARD = LVCMOS18;  NET "B[2]" IOSTANDARD = LVCMOS18;  NET "B[1]" IOSTANDARD = LVCMOS18;  NET "B[0]" IOSTANDARD = LVCMOS18;  NET "P[15]" IOSTANDARD = LVCMOS18;  NET "P[14]" IOSTANDARD = LVCMOS18;  NET "P[13]" IOSTANDARD = LVCMOS18;  NET "P[12]" IOSTANDARD = LVCMOS18;  NET "P[11]" IOSTANDARD = LVCMOS18;  NET "P[10]" IOSTANDARD = LVCMOS18;  NET "P[9]" IOSTANDARD = LVCMOS18;  NET "P[8]" IOSTANDARD = LVCMOS18;  NET "P[7]" IOSTANDARD = LVCMOS18;  NET "P[6]" IOSTANDARD = LVCMOS18;  NET "P[5]" IOSTANDARD = LVCMOS18;  NET "P[4]" IOSTANDARD = LVCMOS18;  NET "P[3]" IOSTANDARD = LVCMOS18;  NET "P[2]" IOSTANDARD = LVCMOS18;  NET "P[1]" IOSTANDARD = LVCMOS18;  NET "P[0]" IOSTANDARD = LVCMOS18;  //NET "SD1\_" IOSTANDARD = LVCMOS18 |CLOCK\_DEDICATED\_ROUTE = FALSE;//下跳沿要用FALSE | | | | | |
| **六、思考与探索** | | | | | |
| 1. 实验结果记录：   在板级验证中  N17按钮控制时钟信号  开关V5到U9为A[7]到A[0]  开关V10到T14为B[7]到B[0]  LED V17到R7共16位，其中V17表示P的低位  当T4和U12拨下，表示A=4 B=4 ，按下N17 U14LED灯亮起 表示P=16  当T6V7R8拨下 表示A=14 ，U12T13拨下表示B=12，此时T15R13R12灯亮起 表示P=168  若不拨下表示A的开关，拨下任何B的开关，灯均不亮，此时A=0  若不拨下表示B的开关，拨下任何A的开关，灯均不亮，此时B=0   1. 实验结论：   实验结果也符合乘法法则，说明实验结果正确   1. 问题与解决方案： 2. 不知道如何配置IP核 3. 不知道在顶层模块中调用配置好的IP核   解决方案：通过网上搜索，查询了IP核的文档后，成功配置好了IP核，    使用这种形式调用IP核   1. 思考题： | | | | | |