

数字逻辑实验报告（4）

团队成员：

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 班级 | 学号 | 贡献百分比 |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实验部分：实验完成结果、时间  （亮点、完成、基本完成、未完成） | | | | 总分  （实验部分 **70% +**报告 **30%**） |
|  | 第一个实验 | 第二个实验 | 第三个实验 |  |
| 检查结果 |  |  |  |
| 检查时间 |  |  |  |
| 检查老师 |  |  |  |

报告人 ：

实验指导教师：

报告批阅教师：

计算机科学与技术学院

**20** 年 月 日

**一、实验内容**

综合应用 Verilog HDL 设计电路。

**二、实验目的**

1．掌握综合应用 Verilog 设计电路的方法；

2．熟悉顶层文件与电路例化的设计；

3．掌握状态机实现控制电路的应用方法。

**三、实验所用组件**

Xilinx NEXYS 4 开发板（芯片为 XC7A100TCSG324-1，封装为 CSG324）1 套

**四、实验要求**

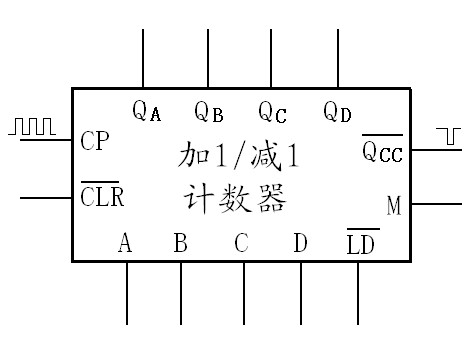
**1. 4 位二进制计数器（必选）**

设计一个能清零、置数和进位/借位输出的增 1/减 1 的 4 位二进制计数器，

其结构框图如图 1 所示。电路输入为计数脉冲 CP、工作模式选择 M、预制初值 D，C，B，A（其中 D 为高位，A 为低位）和预制控制 *LD* ，清零端*CLR* ；输出 为计数值 QD，QC，QB，QA（QD 为高位，QA 为低位）和进位/借位输出*Qcc* 。当 *CLR* 为 0 时，电路输出清零；预制控制 *LD* =0 时，将 D、C、B、A 的输入值送 到计数器中，并立即在 QD，QC，QB，QA 中输出。模式选择端 M=1 时加 1 计数，

当 M=0 时减 1 计数。当 CP 端输入一个上升沿信号时进行一次计数，计数有进

位/借位时*Qcc* 端输出一个负脉冲。



具体要求：

图1 4位二进制加法/减法计数器

（1）用 Verilog HDL 实现该计数器，将之下载到开发板中进行验证；

（2）借助该 4 位二进制计数器，用 Verilog HDL 实现一个初值为 2 的模 8

计数器，将之下载到开发板中进行验证。

**2. 两位二进制数值比较器（必选）**

设计一个二位二进制数值比较器。当 A>B 时，F1=1，F2=F3=0；当 A=B 时，

F2=1，F1=F3=0；当 A<B 时，F3=1，F1=F2=0。 具体要求：

（1）用 Verilog HDL 设计一个一位二进制数值比较器（采用结构级）；

（2）借助该一位二进制数值比较器，用 Verilog HDL 设计一个二位二进制 数值比较器（采用行为级）；

（3）将所设计的电路下载到开发板上进行验证。

**3. 有限状态机(FSM)（可选）**

采用 Gray 编码设计一个 4 位的流水灯控制电路，通过拨码开关控制，可以显

示两种流水灯。

**五、实验方案设计**

**1. 4 位二进制计数器的设计方案**

**（A）模 16 加 1/减 1 计数器**

（a）源程序

`timescale 1ns / 1ps

module lab4\_1(

input CP,M,D,C,B,A,LD,CLR,

output reg Qd,Qc,Qb,Qa,Qcc

);

reg [3:0] count;

always @(posedge CP or negedge CLR or negedge LD)

begin

if(CLR == 0)

begin

count = 0;

{Qd,Qc,Qb,Qa} = count;

Qcc = 1;

end

else if(LD == 0)

begin

count = {D,C,B,A};

{Qd,Qc,Qb,Qa} = count;

Qcc = 1;

end

else if(M == 1)

begin

if(count ==15)

begin

count = 0;

{Qd,Qc,Qb,Qa} = count;

Qcc = 0;

end

else

begin

count = count + 1;

{Qd,Qc,Qb,Qa} = count;

Qcc = 1;

end

end

else

begin

if(count == 0)

begin

count = 15;

{Qd,Qc,Qb,Qa} = count;

Qcc = 0;

end

else

begin

count =count - 1;

{Qd,Qc,Qb,Qa} = count;

Qcc = 1;

end

end

end

endmodule

（b）仿真程序

`timescale 1ns / 1ps

module lab4\_1\_tb(

);

reg CP,M,D,C,B,A,LD,CLR;

wire Qd,Qc,Qb,Qa,Qcc;

lab4\_1 dut(CP,M,D,C,B,A,LD,CLR,Qd,Qc,Qb,Qa,Qcc);

initial

begin

for(CP = 0;CP >= 0;CP = CP + 1)

begin

#5;

end

end

initial

begin

LD = 0;D = 0;C = 1;B = 0;A = 1;CLR = 1;M = 1;

#10 LD = 1;

end

endmodule

（c）引脚约束（绑定）程序

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets CP\_IBUF]

set\_property PACKAGE\_PIN U9 [get\_ports CP]

set\_property IOSTANDARD LVCMOS33 [get\_ports CP]

set\_property PACKAGE\_PIN U8 [get\_ports CLR]

set\_property IOSTANDARD LVCMOS33 [get\_ports CLR]

set\_property PACKAGE\_PIN R7 [get\_ports LD]

set\_property IOSTANDARD LVCMOS33 [get\_ports LD]

set\_property PACKAGE\_PIN R6 [get\_ports M]

set\_property IOSTANDARD LVCMOS33 [get\_ports M]

set\_property PACKAGE\_PIN R5 [get\_ports A]

set\_property IOSTANDARD LVCMOS33 [get\_ports A]

set\_property PACKAGE\_PIN V7 [get\_ports B]

set\_property IOSTANDARD LVCMOS33 [get\_ports B]

set\_property PACKAGE\_PIN V6 [get\_ports C]

set\_property IOSTANDARD LVCMOS33 [get\_ports C]

set\_property PACKAGE\_PIN V5 [get\_ports D]

set\_property IOSTANDARD LVCMOS33 [get\_ports D]

set\_property PACKAGE\_PIN T8 [get\_ports Qa]

set\_property IOSTANDARD LVCMOS33 [get\_ports Qa]

set\_property PACKAGE\_PIN V9 [get\_ports Qb]

set\_property IOSTANDARD LVCMOS33 [get\_ports Qb]

set\_property PACKAGE\_PIN R8 [get\_ports Qc]

set\_property IOSTANDARD LVCMOS33 [get\_ports Qc]

set\_property PACKAGE\_PIN T6 [get\_ports Qd]

set\_property IOSTANDARD LVCMOS33 [get\_ports Qd]

**（B）初值为 2 的模 8 计数器，要求：通过例化 A【调用（A）实现的计数**

**器】来实现。**

（a）源程序

module lab4\_1\_2(

input CP,M,

output wire Qd,Qc,Qb,Qa,

output reg Z

);

reg LD;

wire Qcc;

fb\_count cou1(CP,M,0,0,1,0,LD,1,Qd,Qc,Qb,Qa,Qcc);

initial

begin

LD = 0;

#2 LD = 1;

end

always @(Qd or Qc or Qb or Qa)

begin

LD = ~(Qd & (~Qc) & Qb & (~Qa));

Z = (~Qd) & (~Qc) & Qb & (~Qa);

end

endmodule

module fb\_count(

input CP,M,D,C,B,A,LD,CLR,

output reg Qd,Qc,Qb,Qa,Qcc

);

reg [3:0] count;

always @(posedge CP or negedge CLR or negedge LD)

begin

if(CLR == 0)

begin

count = 0;

{Qd,Qc,Qb,Qa} = count;

Qcc = 1;

end

else if(LD == 0)

begin

count = {D,C,B,A};

{Qd,Qc,Qb,Qa} = count;

Qcc = 1;

end

else if(M == 1)

begin

if(count ==15)

begin

count = 0;

{Qd,Qc,Qb,Qa} = count;

Qcc = 0;

end

else

begin

count = count + 1;

{Qd,Qc,Qb,Qa} = count;

Qcc = 1;

end

end

else

begin

if(count == 0)

begin

count = 15;

{Qd,Qc,Qb,Qa} = count;

Qcc = 0;

end

else

begin

count =count - 1;

{Qd,Qc,Qb,Qa} = count;

Qcc = 1;

end

end

end

endmodule

（b）仿真程序

module lab4\_1\_2\_tb(

);

reg CP,M;

wire Qd,Qc,Qb,Qa,Z;

lab4\_1\_2 dut(CP,M,Qd,Qc,Qb,Qa,Z);

initial

begin

M = 1;

#200 M = 0;

end

initial

begin

for(CP = 0;CP >= 0;CP = CP + 1)

begin

#5;

end

end

endmodule

（c）引脚约束（绑定）程序

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets CP\_IBUF]

set\_property PACKAGE\_PIN F15 [get\_ports CP]

set\_property IOSTANDARD LVCMOS33 [get\_ports CP]

set\_property PACKAGE\_PIN U9 [get\_ports M]

set\_property IOSTANDARD LVCMOS33 [get\_ports M]

set\_property PACKAGE\_PIN T8 [get\_ports Qa]

set\_property IOSTANDARD LVCMOS33 [get\_ports Qa]

set\_property PACKAGE\_PIN V9 [get\_ports Qb]

set\_property IOSTANDARD LVCMOS33 [get\_ports Qb]

set\_property PACKAGE\_PIN R8 [get\_ports Qc]

set\_property IOSTANDARD LVCMOS33 [get\_ports Qc]

set\_property PACKAGE\_PIN T6 [get\_ports Qd]

set\_property IOSTANDARD LVCMOS33 [get\_ports Qd]

set\_property PACKAGE\_PIN T5 [get\_ports Z]

set\_property IOSTANDARD LVCMOS33 [get\_ports Z]

**2. 两位二进制数值比较器的设计方案**

**（A）一位二进制数值比较器**

（a）源程序

module one\_comparator(

input a,b,

output f1,f2,f3

);

wire f21,f22;

and

ua1(f1,a,~b),

ua2(f21,~a,~b),

ua3(f22,a,b),

ua4(f3,~a,b);

or

uo1(f2,f21,f22);

endmodule

（b）仿真程序

module lab4\_2\_1\_tb(

);

reg a,b;

wire f1,f2,f3;

integer i,j;

one\_comparator dut(a,b,f1,f2,f3);

initial

begin

a = 0;b = 0;

for(i = 0;i <= 1;i = i + 1)

begin

for(j = 0;j <= 1;j = j + 1)

begin

#10;

b = b + 1;

end

a = a + 1;

end

end

endmodule

（c）引脚约束（绑定）程序

set\_property PACKAGE\_PIN U9 [get\_ports a]

set\_property IOSTANDARD LVCMOS33 [get\_ports a]

set\_property PACKAGE\_PIN U8 [get\_ports b]

set\_property IOSTANDARD LVCMOS33 [get\_ports b]

set\_property PACKAGE\_PIN T8 [get\_ports f1]

set\_property IOSTANDARD LVCMOS33 [get\_ports f1]

set\_property PACKAGE\_PIN V9 [get\_ports f2]

set\_property IOSTANDARD LVCMOS33 [get\_ports f2]

set\_property PACKAGE\_PIN R8 [get\_ports f3]

set\_property IOSTANDARD LVCMOS33 [get\_ports f3]

**（B）二位二进制数值比较器，要求：通过例化 A【调用（A）实现的一位**

**二进制数值比较器】来实现。**

（a）源程序

module lab4\_2\_1(

input [1:0] A,B,

output reg F1,F2,F3

);

wire F11,F12,F21,F22,F31,F32;

one\_comparator com1(A[1],B[1],F12,F22,F32);

one\_comparator com2(A[0],B[0],F11,F21,F31);

always @(F11,F12,F21,F22,F31,F32)

begin

if(F12 | (F22 & F11))

begin

F1 = 1;F2 = 0;F3 = 0;

end

else if(F22 & F21)

begin

F1 = 0;F2 = 1;F3 = 0;

end

else

begin

F1 = 0;F2 = 0;F3 = 1;

end

end

endmodule

module one\_comparator(

input a,b,

output f1,f2,f3

);

wire f21,f22;

and

ua1(f1,a,~b),

ua2(f21,~a,~b),

ua3(f22,a,b),

ua4(f3,~a,b);

or

uo1(f2,f21,f22);

endmodule

（b）仿真程序

module lab4\_2\_1\_tb(

);

reg [1:0] A,B;

wire F1,F2,F3;

integer i,j;

lab4\_2\_1 dut(A,B,F1,F2,F3);

initial

begin

A = 0;B = 0;

for(i = 0;i <= 3;i = i + 1)

begin

for(j = 0;j <= 3;j = j + 1)

begin

#10;

B = B + 1;

end

A = A + 1;

end

end

endmodule

（c）引脚约束（绑定）程序

set\_property PACKAGE\_PIN U9 [get\_ports A[0]]

set\_property IOSTANDARD LVCMOS33 [get\_ports A[0]]

set\_property PACKAGE\_PIN U8 [get\_ports A[1]]

set\_property IOSTANDARD LVCMOS33 [get\_ports A[1]]

set\_property PACKAGE\_PIN R7 [get\_ports B[0]]

set\_property IOSTANDARD LVCMOS33 [get\_ports B[0]]

set\_property PACKAGE\_PIN R6 [get\_ports B[1]]

set\_property IOSTANDARD LVCMOS33 [get\_ports B[1]]

set\_property PACKAGE\_PIN T8 [get\_ports F1]

set\_property IOSTANDARD LVCMOS33 [get\_ports F1]

set\_property PACKAGE\_PIN V9 [get\_ports F2]

set\_property IOSTANDARD LVCMOS33 [get\_ports F2]

set\_property PACKAGE\_PIN R8 [get\_ports F3]

set\_property IOSTANDARD LVCMOS33 [get\_ports F3]

**3.采用状态机的流水灯控制电路设计方案**

（A）4 位流水灯控制电路的状态图

（B）源程序

module lab4\_3\_1(

input m,clk,

output reg [3:0] Z

);

reg n;

reg [1:0] state,nextstate;

parameter S0 = 2'b00,S1 = 2'b01,S2 = 2'b11,S3 = 2'b10;

initial

begin

state = S0;

n = 0;

end

always @(posedge clk)

begin

if(n == 10000000)

begin

state = nextstate;

n = 0;

end

else

begin

n = n + 1;

end

end

always @(state)

begin

case(state)

S0:if(m) Z = 4'b0001;else Z = 4'b0001;

S1:if(m) Z = 4'b0010;else Z = 4'b1000;

S2:if(m) Z = 4'b0100;else Z = 4'b0100;

S3:if(m) Z = 4'b1000;else Z = 4'b0010;

endcase

end

always @(state)

begin

case(state)

S0: nextstate = S1;

S1: nextstate = S2;

S2: nextstate = S3;

S3: nextstate = S0;

endcase

end

endmodule

（C）仿真程序

module lab4\_3\_1\_tb(

);

reg m,clk;

wire [3:0] Z;

lab4\_3\_1 dut(m,clk,Z);

initial

begin

for(clk = 0;clk >= 0;clk = clk + 1)

begin

#5;

end

end

initial

begin

m = 0;

#300 m = 1;

end

endmodule

（D）引脚约束（绑定）程序

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets clk\_IBUF]

set\_property PACKAGE\_PIN E3 [get\_ports clk]

set\_property IOSTANDARD LVCMOS33 [get\_ports clk]

set\_property PACKAGE\_PIN T8 [get\_ports m]

set\_property IOSTANDARD LVCMOS33 [get\_ports m]

set\_property PACKAGE\_PIN V9 [get\_ports Z[0]]

set\_property IOSTANDARD LVCMOS33 [get\_ports Z[0]]

set\_property PACKAGE\_PIN R8 [get\_ports Z[1]]

set\_property IOSTANDARD LVCMOS33 [get\_ports Z[1]]

set\_property PACKAGE\_PIN T6 [get\_ports Z[2]]

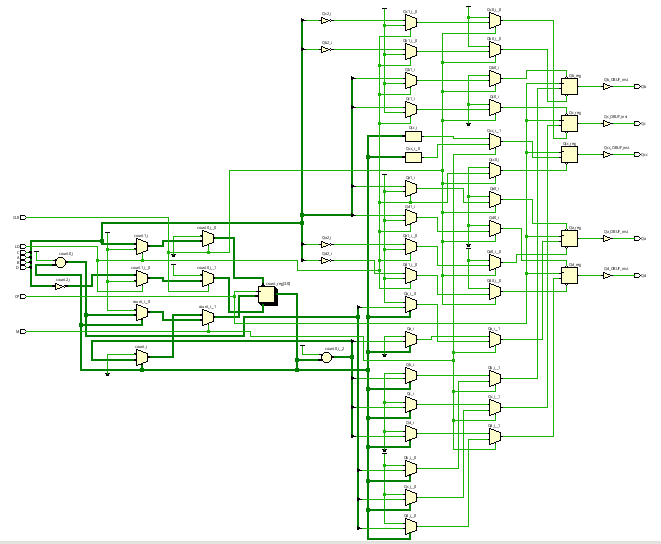
set\_property IOSTANDARD LVCMOS33 [get\_ports Z[2]]

set\_property PACKAGE\_PIN T5 [get\_ports Z[3]]

set\_property IOSTANDARD LVCMOS33 [get\_ports Z[3]]

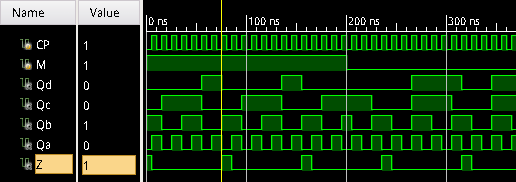
**六、实验结果记录**

**1. 4 位二进制计数器的实验结果记录**

**（1）给出 Verilog 设计的模 16 加 1/减 1 计数器的电路图**（RTL Analysis 下的 “Schematic”）

**（2）初值为 2 的模 8 计数器仿真结果（波形含 CP，M，Qa，Qb，Qc，Qd，**

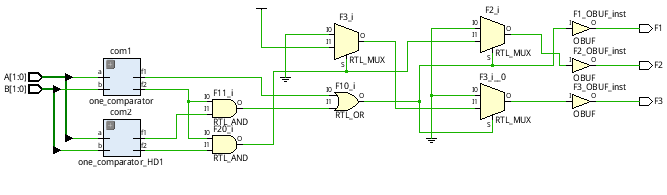
**Q 模 8 等）**

（仿真结果具有周期性的要在仿真图上将其“周期”加以标注和说明）

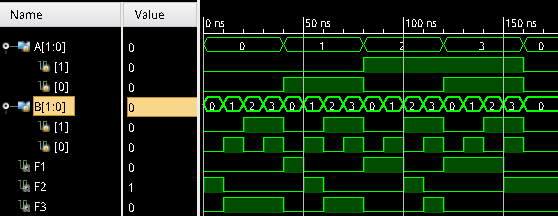
**（3）开发板上电路的验证情况（主要记录：验证过程和结论）**

**2. 两位二进制数值比较器的结果记录**

**（1）给出 Verilog 设计的两位二进制数值比较器的电路图**（RTL Analysis 下的 “Schematic”）



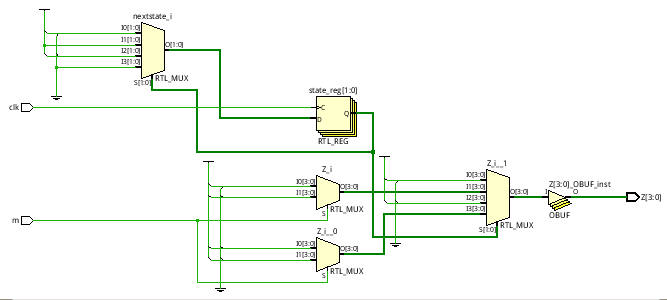
**（2）仿真结果（波形含 F1，F2，F3，A1，A0，B1，B0 等）**



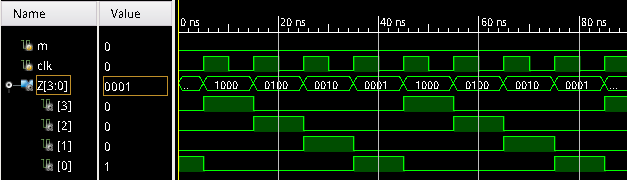
**（3）开发板上电路的验证情况（主要记录：验证过程和结论）**

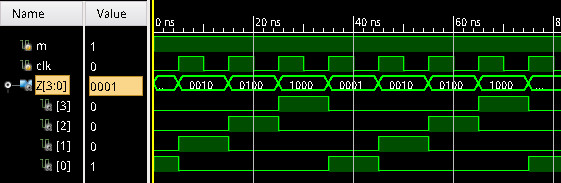
**3.采用状态机的流水灯控制电路实验结果记录**

**（1）给出 Verilog 设计的流水灯电路图**（RTL Analysis 下的“Schematic”）



**（2）仿真结果（波形含 4 盏灯，控制开关，时钟等）**

（a）流水灯显示方式 1

（b）流水灯显示方式 2

**（3）开发板上电路的验证情况（主要记录：验证过程和结论）**

**七．实验结果分析**

**八．思考题**

1．用 HDL 设计数字逻辑电路的设计步骤有哪些？

2．实验 2 中 HDL 的结构级描述和行为级描述各有什么特点？

3．实验 3 中如果采用边沿敏感时序，例如 always @（*posedge*.. , *negedge..*）设计中 要注意那些问题？

**九．心得体会**

**十．意见与建议**