s



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

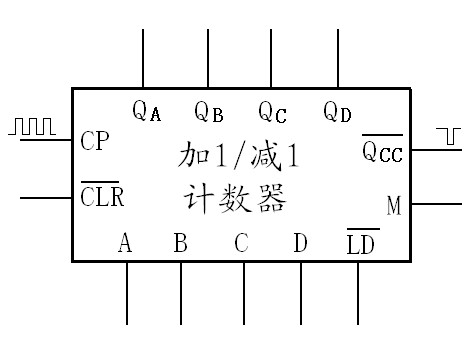
**1. 4** 位二进制计数器（必选）

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 \_4bit\_binary\_counter(

input CP,

input M,

input [3:0] D,

input LD\_n,

input CLR\_n,

output reg [3:0] Q,

output reg Qcc\_n

);

always @(negedge CLR\_n or negedge LD\_n or posedge CP) begin

if (Qcc\_n == 0) begin

Qcc\_n = 1;

end

if (!CLR\_n) begin

{Qcc\_n, Q} <= 5'b10000;

end

else if (!LD\_n) begin

{Qcc\_n, Q} <= {1'b1, D};

end

else if(CP) begin

if (M) begin

{Qcc\_n, Q} <= {Qcc\_n,Q} + 1;

end

else begin

{Qcc\_n, Q} <= {Qcc\_n,Q} - 1;

end

end

end

endmodule

（b）仿真程序

`timescale 1ns / 1ps

module \_4bit\_binary\_counter\_tb(

);

parameter COUNT = 85;

parameter DELAY = 10;

reg CP, M, LD\_n, CLR\_n;

reg [3:0] D;

wire [3:0] Q;

wire Qcc\_n;

integer i;

\_4bit\_binary\_counter DUT (.CP(CP), .M(M), .D(D), .LD\_n(LD\_n), .CLR\_n(CLR\_n), .Q(Q), .Qcc\_n(Qcc\_n));

initial begin

#(COUNT \* DELAY) $finish;

end

initial begin

CP = 0;

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

#DELAY CP = ~CP;

end

end

initial begin

M = 1;

#(COUNT\*DELAY/2) M = 0;

end

initial begin

D = 4'b1111; LD\_n = 1; CLR\_n = 0;

#DELAY CLR\_n = 1;

#(5\*DELAY) CLR\_n = 0;

#(3\*DELAY) CLR\_n = 1;

#(5\*DELAY) LD\_n = 0;

#(2\*DELAY) D = 4'b0111;

#(4\*DELAY) LD\_n = 1;

end

endmodule

（c）引脚约束（绑定）程序

# Nexys4 Pin Assignments

############################

# On-board Slide Switches #

############################

# CP, M, LD\_n, CLR\_n, D[3:0]

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 CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets M\_IBUF]

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

set\_property IOSTANDARD LVCMOS33 [get\_ports M]

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

set\_property PACKAGE\_PIN U8 [get\_ports LD\_n]

set\_property IOSTANDARD LVCMOS33 [get\_ports LD\_n]

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

set\_property PACKAGE\_PIN R7 [get\_ports CLR\_n]

set\_property IOSTANDARD LVCMOS33 [get\_ports CLR\_n]

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets D[0]\_IBUF]

set\_property PACKAGE\_PIN R6 [get\_ports D[0]]

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

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets D[1]\_IBUF]

set\_property PACKAGE\_PIN R5 [get\_ports D[1]]

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

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets D[2]\_IBUF]

set\_property PACKAGE\_PIN V7 [get\_ports D[2]]

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

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets D[3]\_IBUF]

set\_property PACKAGE\_PIN V6 [get\_ports D[3]]

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

############################

# On-board led #

############################

# Q[3:0], Qcc\_n

set\_property PACKAGE\_PIN T8 [get\_ports Q[0]]

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

set\_property PACKAGE\_PIN V9 [get\_ports Q[1]]

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

set\_property PACKAGE\_PIN R8 [get\_ports Q[2]]

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

set\_property PACKAGE\_PIN T6 [get\_ports Q[3]]

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

set\_property PACKAGE\_PIN T5 [get\_ports Qcc\_n]

set\_property IOSTANDARD LVCMOS33 [get\_ports Qcc\_n]

（**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

`timescale 1ns / 1ps

module \_4bit\_binary\_counter(

input CP,

input M,

input [3:0] D,

input LD\_n,

input CLR\_n,

output reg [3:0] Q,

output reg Qcc\_n

);

always @(negedge CLR\_n or negedge LD\_n or posedge CP) begin

if (Qcc\_n == 0) begin

Qcc\_n = 1;

end

if (!CLR\_n) begin

{Qcc\_n, Q} <= 5'b10000;

end

else if (!LD\_n) begin

{Qcc\_n, Q} <= {1'b1, D};

end

else if(CP) begin

if (M) begin

{Qcc\_n, Q} <= {Qcc\_n,Q} + 1;

end

else begin

{Qcc\_n, Q} <= {Qcc\_n,Q} - 1;

end

end

end

endmodule

（b）仿真程序

`timescale 1ns / 1ps

module \_4bit\_mod8\_counter\_tb(

);

parameter COUNT = 200;

parameter DELAY = 10;

parameter TIME = COUNT\*DELAY;

reg CP,reset,M;

wire [3:0] Q;

wire Qcc\_n;

integer i;

\_4bit\_mod8\_counter DUT (.CP(CP), .reset(reset), .M(M), .Q(Q), .Qcc\_n(Qcc\_n));

initial begin

#TIME $finish;

end

initial begin

CP = 0;

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

#DELAY CP = ~CP;

end

end

initial begin

reset = 1;

#(2\*DELAY) reset = 0;

#(COUNT\*DELAY/5) reset = 1;

#(COUNT\*DELAY/5) reset = 0;

#(COUNT\*DELAY/4) reset = 1;

#(2\*DELAY) reset = 0;

end

initial begin

M = 1;

#(COUNT\*DELAY/2) M = 0;

end

endmodule

（c）引脚约束（绑定）程序

# Nexys4 Pin Assignments

############################

# On-board Slide Switches #

############################

# CP, reset, M

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 CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets reset\_IBUF]

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

set\_property IOSTANDARD LVCMOS33 [get\_ports reset]

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

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

set\_property IOSTANDARD LVCMOS33 [get\_ports M]

############################

# On-board led #

############################

# Q[3:0], Qcc\_n

set\_property PACKAGE\_PIN T8 [get\_ports Q[0]]

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

set\_property PACKAGE\_PIN V9 [get\_ports Q[1]]

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

set\_property PACKAGE\_PIN R8 [get\_ports Q[2]]

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

set\_property PACKAGE\_PIN T6 [get\_ports Q[3]]

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

set\_property PACKAGE\_PIN T5 [get\_ports Qcc\_n]

set\_property IOSTANDARD LVCMOS33 [get\_ports Qcc\_n]

**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

`timescale 1ns / 1ps

module \_1bit\_binary\_comparator(

input A,

input B,

output [2:0] F

);

assign F[2] = ~A & B; // A < B

assign F[1] = A ~^ B; // A == B

assign F[0] = A & ~B; // A > B

endmodule

（b）仿真程序

`timescale 1ns / 1ps

module \_1bit\_binary\_comparator\_tb(

);

parameter COUNT = 4;

parameter DELAY = 10;

reg A, B;

wire [2:0] F;

integer i;

\_1bit\_binary\_comparator DUT (.A(A), .B(B), .F(F));

initial begin

#(COUNT \* DELAY) $finish;

end

initial begin

{A, B} = 2'b00;

for (i = 1; i < COUNT; i = i + 1) begin

#DELAY {A, B} = i;

end

end

endmodule

（c）引脚约束（绑定）程序

Nexys4 Pin Assignments

############################

# On-board Slide Switches #

############################

# A, B

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

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

set\_property IOSTANDARD LVCMOS33 [get\_ports A]

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

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

set\_property IOSTANDARD LVCMOS33 [get\_ports B]

############################

# On-board led #

############################

# F[2:0]

set\_property PACKAGE\_PIN T8 [get\_ports F[0]]

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

set\_property PACKAGE\_PIN V9 [get\_ports F[1]]

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

set\_property PACKAGE\_PIN R8 [get\_ports F[2]]

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

（**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

`timescale 1ns / 1ps

module \_2bit\_binary\_comparator(

input [1:0] A,

input [1:0] B,

output reg [2:0] F

);

wire [2:0] F1, F0;

\_1bit\_binary\_comparator COM1 (.A(A[1]), .B(B[1]), .F(F1));

\_1bit\_binary\_comparator COM0 (.A(A[0]), .B(B[0]), .F(F0));

always @ \* begin

if (F1 == 3'b010) F <= F0;

else F <= F1;

end

endmodule

（b）仿真程序

`timescale 1ns / 1ps

module \_2bit\_binary\_comparator\_tb(

);

reg [1:0] A, B;

wire [2:0] F;

integer i;

\_2bit\_binary\_comparator DUT (.A(A), .B(B), .F(F));

initial begin

#200 $finish;

end

initial begin

{A, B} = 4'b0000;

for (i = 1; i < 16; i = i + 1) begin

#10 {A, B} = i;

end

end

endmodule

（c）引脚约束（绑定）程序

# Nexys4 Pin Assignments

############################

# On-board Slide Switches #

############################

# A, B

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets A[0]\_IBUF]

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

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

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets A[1]\_IBUF]

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

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

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets B[0]\_IBUF]

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

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

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets B[1]\_IBUF]

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

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

############################

# On-board led #

############################

# F[2:0]

set\_property PACKAGE\_PIN T8 [get\_ports F[0]]

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

set\_property PACKAGE\_PIN V9 [get\_ports F[1]]

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

set\_property PACKAGE\_PIN R8 [get\_ports F[2]]

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

**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

5

8

4

0

1

6

2

3

7

12

9

11

15

14

10

13

0

1

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

（B）源程序

`timescale 1ns / 1ps

module binary\_counter

#(parameter LEN = 32)

(

input clk\_src,

input reset,

output reg [(LEN-1): 0] clk\_group

);

always @(posedge clk\_src or posedge reset) begin

if (reset) clk\_group <= {(LEN){1'b0}};

else clk\_group <= clk\_group + 1;

end

endmodule

module flow\_led(

input clk\_src,

input reset,

input [4:0] clk\_sel, // select switch for clock signal

input M,

output reg [3:0] state

);

parameter LEN = 32; // length of binary counter(different types of clock signal)

parameter l0 = 0, // // state enums

l1 = 1,

l2 = 3,

l3 = 2,

l4 = 6,

l5 = 7,

l6 = 5,

l7 = 4,

l8 = 12,

l9 = 13,

l10 = 15,

l11 = 14,

l12 = 10,

l13 = 11,

l14 = 9,

l15 = 8;

reg [3:0] nextState;

wire clk; // truly clock signal

wire [(LEN-1):0] clk\_group; // divided clock signal from clk, by binary counter

assign clk = clk\_group[clk\_sel];

binary\_counter DIVIDER (.clk\_src(clk\_src), .reset(reset), .clk\_group(clk\_group));

always @(posedge clk or posedge reset) begin

if (reset) state <= l0;

else state <= nextState;

end

always @(M or state) begin

case (state)

l0: if (!M) nextState = l1;

else nextState = l15;

l1: if (!M) nextState = l2;

else nextState = l0;

l2: if (!M) nextState = l3;

else nextState = l1;

l3: if (!M) nextState = l4;

else nextState = l2;

l4: if (!M) nextState = l5;

else nextState = l3;

l5: if (!M) nextState = l6;

else nextState = l4;

l6: if (!M) nextState = l7;

else nextState = l5;

l7: if (!M) nextState = l8;

else nextState = l6;

l8: if (!M) nextState = l9;

else nextState = l7;

l9: if (!M) nextState = l10;

else nextState = l8;

l10: if (!M) nextState = l11;

else nextState = l9;

l11: if (!M) nextState = l12;

else nextState = l10;

l12: if (!M) nextState = l13;

else nextState = l11;

l13: if (!M) nextState = l14;

else nextState = l12;

l14: if (!M) nextState = l15;

else nextState = l13;

l15: if (!M) nextState = l0;

else nextState = l14;

endcase

end

endmodule

（C）仿真程序

`timescale 1ns / 1ps

module flow\_led\_tb();

parameter COUNT = 1000; // number of T

parameter DELAY = 5; // T = 10ns, f = 100MHz

reg clk\_src, reset, M;

reg [4:0] clk\_sel;

wire [3:0] state;

integer i;

flow\_led DUT (.clk\_src(clk\_src), .reset(reset), .clk\_sel(clk\_sel), .M(M), .state(state));

initial begin

#(COUNT\*DELAY) $finish;

end

initial begin

clk\_src = 0;

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

#DELAY clk\_src = ~clk\_src;

end

end

initial begin

M = 0;

#(COUNT\*DELAY/2) M = 1;

end

initial begin

reset = 0;

#DELAY reset = 1;

#DELAY reset = 0;

#((COUNT-5)\*DELAY/2) reset = 0;

#DELAY reset = 1;

#DELAY reset = 0;

end

initial begin

clk\_sel = 5'b0000;

#(COUNT\*DELAY/5) clk\_sel = clk\_sel + 1;

#(COUNT\*DELAY/5) clk\_sel = clk\_sel + 1;

#(COUNT\*DELAY/5) clk\_sel = clk\_sel + 1;

#(COUNT\*DELAY/5) clk\_sel = clk\_sel + 1;

#(COUNT\*DELAY/5) clk\_sel = clk\_sel + 1;

end

endmodule

（D）引脚约束（绑定）程序

# Nexys4 Pin Assignments

##########################################

# On-board Slide Switches/Button/Signal #

##########################################

# Clk, reset, clk\_sel[4:0], M

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

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

set\_property IOSTANDARD LVCMOS33 [get\_ports clk\_src]

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

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

set\_property IOSTANDARD LVCMOS33 [get\_ports reset]

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

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

set\_property IOSTANDARD LVCMOS33 [get\_ports M]

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets clk\_sel[0]\_IBUF]

set\_property PACKAGE\_PIN U8 [get\_ports clk\_sel[0]]

set\_property IOSTANDARD LVCMOS33 [get\_ports clk\_sel[0]]

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets clk\_sel[1]\_IBUF]

set\_property PACKAGE\_PIN R7 [get\_ports clk\_sel[1]]

set\_property IOSTANDARD LVCMOS33 [get\_ports clk\_sel[1]]

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets clk\_sel[2]\_IBUF]

set\_property PACKAGE\_PIN R6 [get\_ports clk\_sel[2]]

set\_property IOSTANDARD LVCMOS33 [get\_ports clk\_sel[2]]

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets clk\_sel[3]\_IBUF]

set\_property PACKAGE\_PIN R5 [get\_ports clk\_sel[3]]

set\_property IOSTANDARD LVCMOS33 [get\_ports clk\_sel[3]]

set\_property CLOCK\_DEDICATED\_ROUTE FALSE [get\_nets clk\_sel[4]\_IBUF]

set\_property PACKAGE\_PIN V7 [get\_ports clk\_sel[4]]

set\_property IOSTANDARD LVCMOS33 [get\_ports clk\_sel[4]]

############################

# On-board led #

############################

# state[3:0]

set\_property PACKAGE\_PIN T8 [get\_ports state[0]]

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

set\_property PACKAGE\_PIN V9 [get\_ports state[1]]

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

set\_property PACKAGE\_PIN R8 [get\_ports state[2]]

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

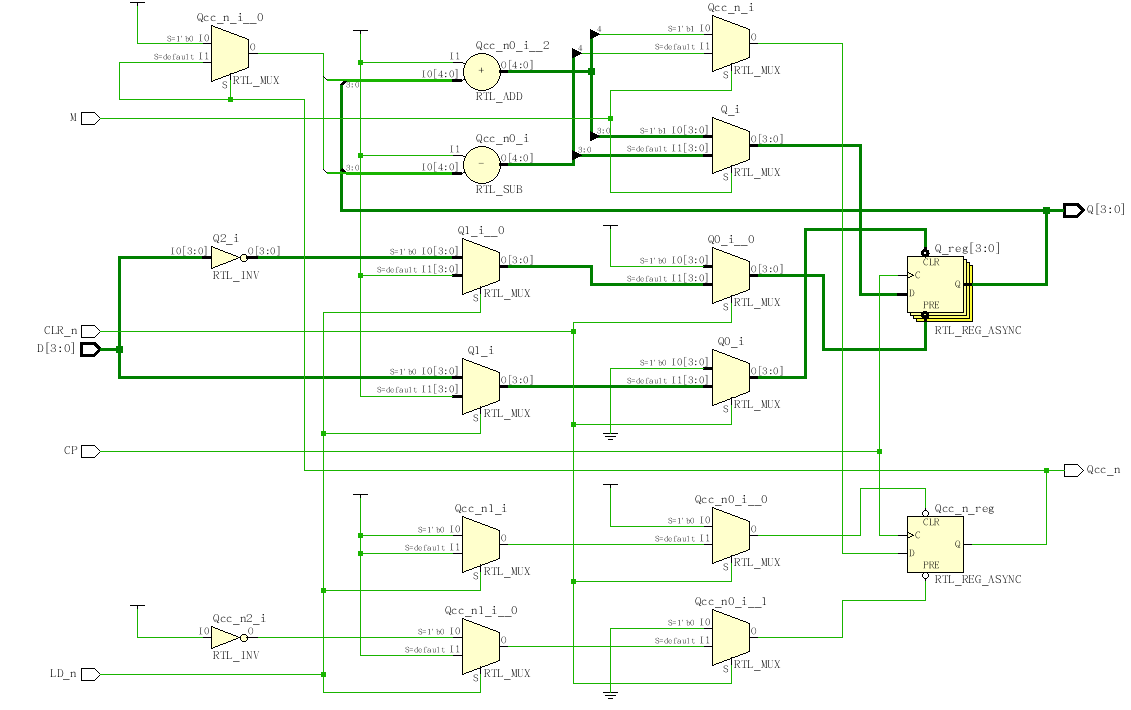
set\_property PACKAGE\_PIN T6 [get\_ports state[3]]

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

六、实验结果记录

**1. 4** 位二进制计数器的实验结果记录

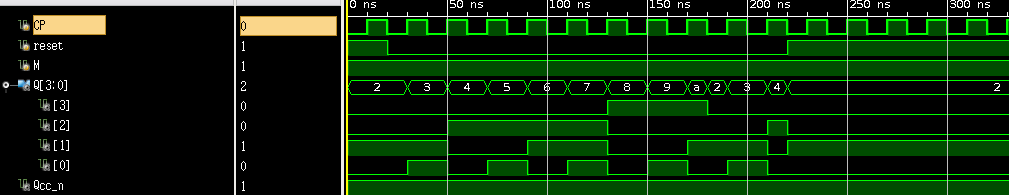
（**1**）给出 **Verilog** 设计的模 **16** 加 **1/**减 **1** 计数器的电路图（RTL Analysis 下的 “Schematic”）

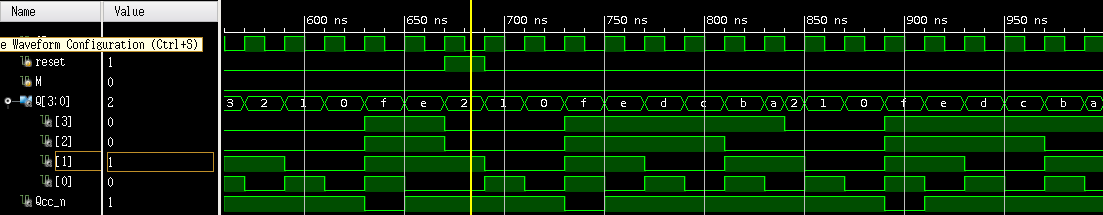


（**2**）初值为 **2** 的模 **8** 计数器仿真结果（波形含 **CP**，**M**，**Qa**，**Qb**，**Qc**，**Qd**，

**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）

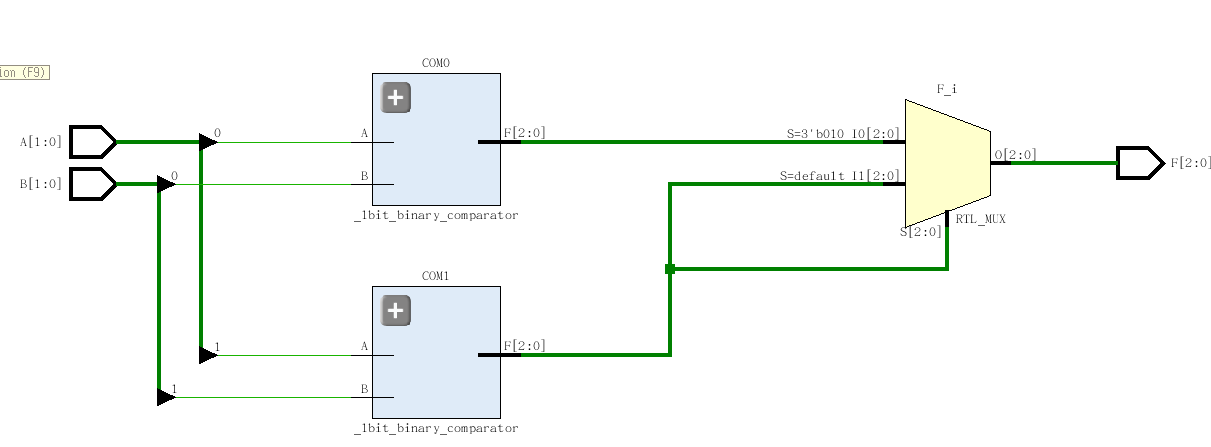




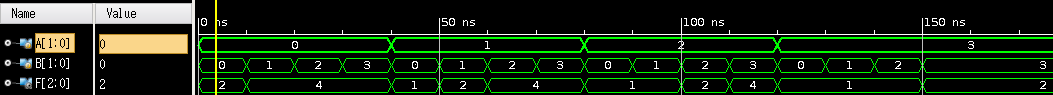
1. 开发板上电路的验证情况（主要记录：验证过程和结论）

**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”）



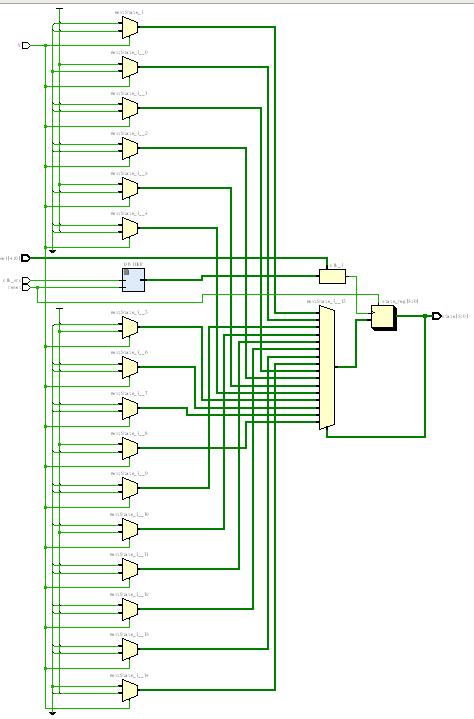
（**2**）仿真结果（波形含 **F1**，**F2**，**F3**，**A1**，**A0**，**B1**，**B0** 等）



（**3**）开发板上电路的验证情况（主要记录：验证过程和结论）

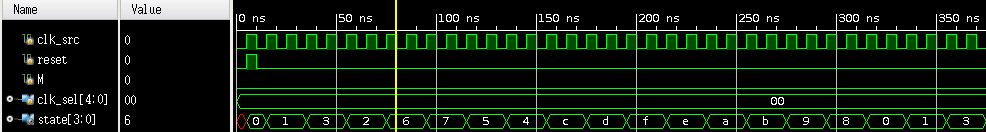
**3.** 采用状态机的流水灯控制电路实验结果记录

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

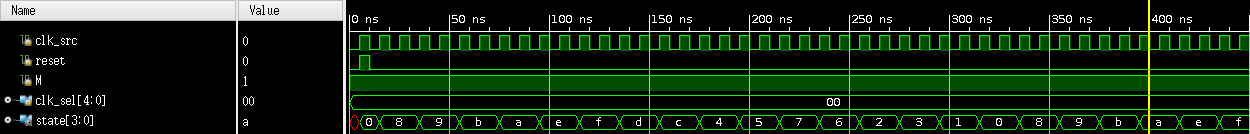


（**2**）仿真结果（波形含 **4** 盏灯，控制开关，时钟等）

（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