实验九 4位Johnson计数器的设计

 （一）同步复位Johnson计数器

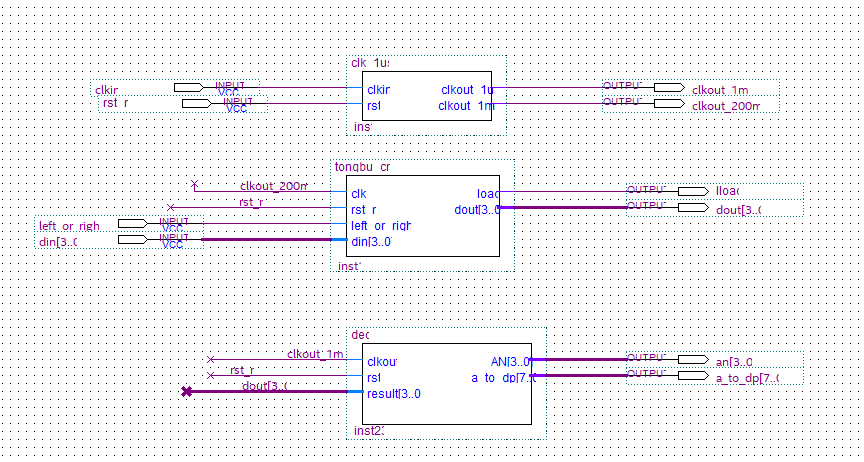
控制信号：

rst：同步复位信号，高电平有效；

load：同步加载信号，低电平有效；

left\_or\_right：同步方向控制信号，高电平左循环，低电平右循环。

1. 顶层文件



1. 计数器模块代码

module tongbu\_cnt(clk,rst\_n,left\_or\_right,load,din,dout);

input clk,rst\_n,left\_or\_right;

input [3:0] din;

output reg [3:0] dout;

output reg load;

reg [7:0] cnt;

always @ (posedge clk )

if (rst\_n) //同步复位信号，高电平有效

dout<=4'b0000;

else if (!load)

dout<=din;

else

begin

if (left\_or\_right==1)

begin

dout[0]<=~dout[3];

dout[3:1]<=dout[2:0];

end

else

begin

dout[3]<=~dout[0];

dout[2:0]<=dout[3:1];

end

end

always @ (posedge clk)

if (rst\_n)

begin

cnt<=0;

load<=1;

end

else if(cnt>=7)

begin

cnt<=0;

load<=0;

end

else

begin

load<=1;//同步加载信号，低电平有效

cnt<=cnt+1;

end

endmodule

1. 测试文件代码

`timescale 1 ns/ 1 ps

module john\_tongbu\_cnt\_vlg\_tst();

// constants

// general purpose registers

//reg eachvec;

// test vector input registers

reg clkin;

reg [3:0] din;

reg left\_or\_right;

reg rst\_n;

// wires

wire [7:0] a\_to\_dp;

wire [3:0] an;

wire clkout\_1ms;

wire clkout\_200ms;

wire [3:0] dout;

wire lload;

// assign statements (if any)

john\_tongbu\_cnt i1 (

// port map - connection between master ports and signals/registers

.a\_to\_dp(a\_to\_dp),

.an(an),

.clkin(clkin),

.clkout\_1ms(clkout\_1ms),

.clkout\_200ms(clkout\_200ms),

.din(din),

.dout(dout),

.left\_or\_right(left\_or\_right),

.lload(lload),

.rst\_n(rst\_n)

);

initial

begin

// code that executes only once

// insert code here --> begin

clkin=0;

rst\_n=0;

left\_or\_right=0;

din=4'b1111;

#10000000

rst\_n=1;

#100000000

rst\_n=0;

din=4'b1101;

din=4'b1001;

din=4'b0111;

#20000000

left\_or\_right=1;

din=4'b1101;

#20000000

din=4'b1001;

#60000000

din=4'b0111;

// --> end

$display("Running testbench");

end

always

// optional sensitivity list

// @(event1 or event2 or .... eventn)

begin

// code executes for every event on sensitivity list

// insert code here --> begin

#10 clkin=~clkin;

//@eachvec;

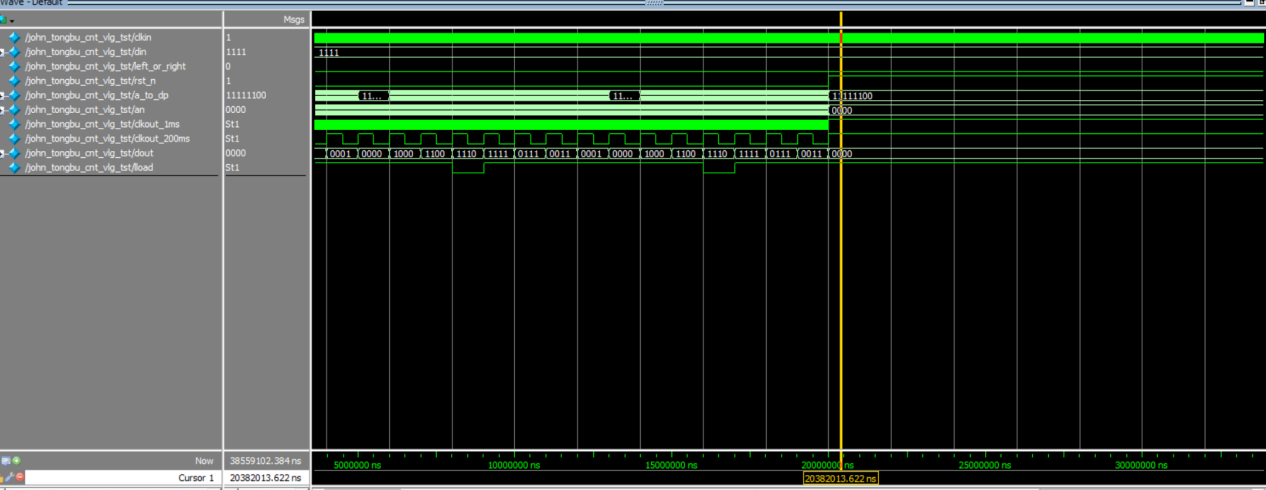
// --> end

end

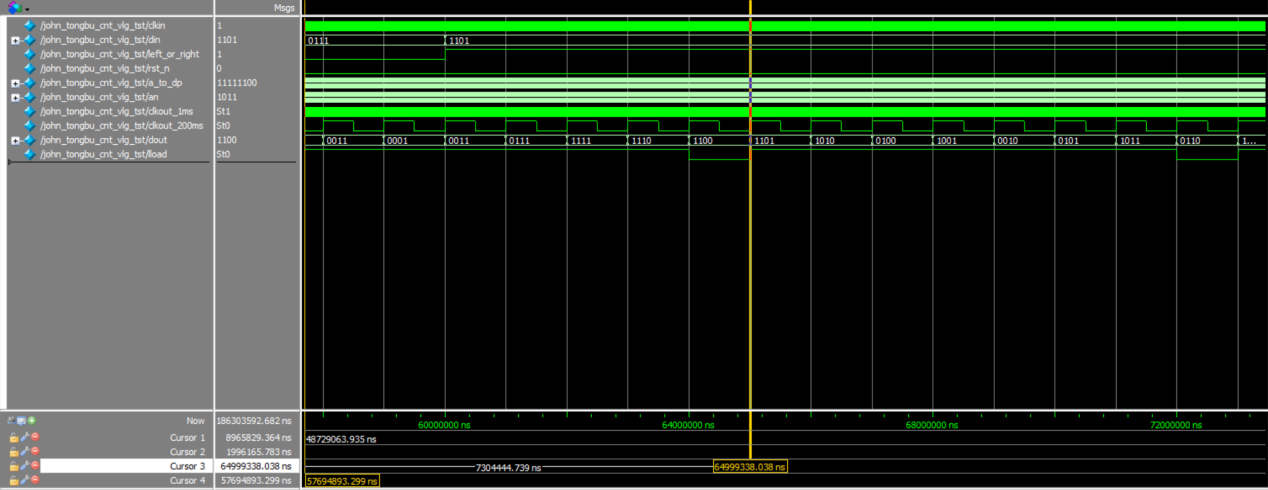
endmodule

1. 仿真波形

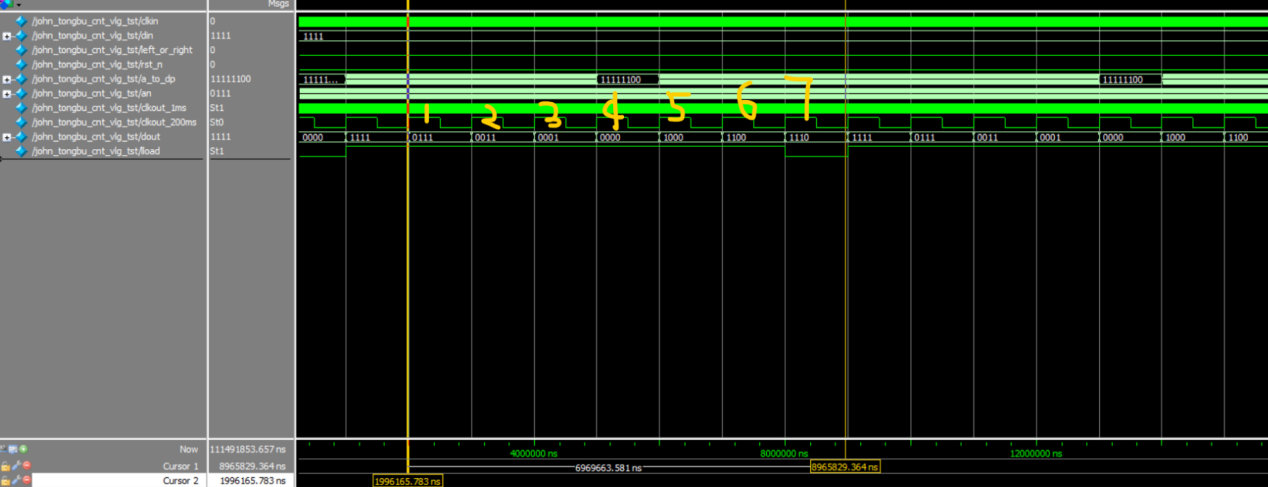
复位有效



加载有效



左移

右移

 （二）异步复位Johnson计数器

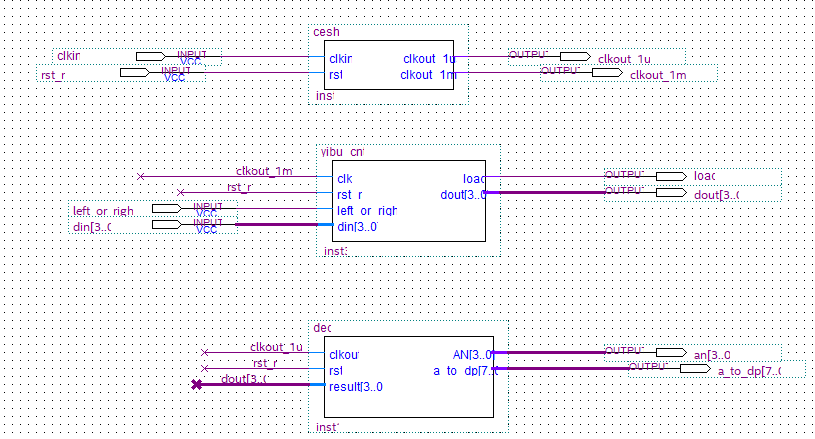
控制信号：

rst：异步复位信号，高电平有效；

load：同步加载信号，低电平有效；

left\_or\_right：同步方向控制信号，高电平左循环，低电平右循环。

1. 顶层文件



1. 计数器模块代码

module yibu\_cnt(clk,rst\_n,left\_or\_right,load,din,dout);

input clk,rst\_n,left\_or\_right;

input [3:0] din;

output reg [3:0] dout;

output reg load;

reg [7:0] cnt;

always @ (posedge clk , posedge rst\_n )

if (rst\_n) //异步复位信号，高电平有效

dout<=4'b0000;

else if (!load)

dout<=din;

else

begin

if (left\_or\_right==1)

begin

dout[0]<=~dout[3];

dout[3:1]<=dout[2:0];

end

else

begin

dout[3]<=~dout[0];

dout[2:0]<=dout[3:1];

end

end

always @ (posedge clk,posedge rst\_n)

if (rst\_n)

begin

cnt<=0;

load<=1;

end

else if(cnt>=7)

begin

cnt<=0;

load<=0;

end

else

begin

load<=1;//同步加载信号，低电平有效

cnt<=cnt+1;

end

endmodule

1. 测试文件代码

`timescale 1 ns/ 1 ps

module John\_yibu\_cnt\_vlg\_tst();

// constants

// general purpose registers

//reg eachvec;

// test vector input registers

reg clkin;

reg [3:0] din;

reg left\_or\_right;

reg rst\_n;

// wires

wire [7:0] a\_to\_dp;

wire [3:0] an;

wire clkout\_1ms;

wire clkout\_1us;

wire [3:0] dout;

wire load;

// assign statements (if any)

John\_yibu\_cnt i1 (

// port map - connection between master ports and signals/registers

.a\_to\_dp(a\_to\_dp),

.an(an),

.clkin(clkin),

.clkout\_1ms(clkout\_1ms),

.clkout\_1us(clkout\_1us),

.din(din),

.dout(dout),

.left\_or\_right(left\_or\_right),

.load(load),

.rst\_n(rst\_n)

);

initial

begin

// code that executes only once

// insert code here --> begin

clkin=0;

rst\_n=0;

left\_or\_right=0;

din=4'b1111;

#10000000

rst\_n=1;

#1000000

rst\_n=0;

din=4'b1101;

din=4'b1001;

din=4'b0111;

#20000000

left\_or\_right=1;

din=4'b1101;

#20000000

din=4'b1001;

#60000000

din=4'b0111;

// --> end

$display("Running testbench");

end

always

// optional sensitivity list

// @(event1 or event2 or .... eventn)

begin

// code executes for every event on sensitivity list

// insert code here --> begin

#10 clkin=~clkin;

//@eachvec;

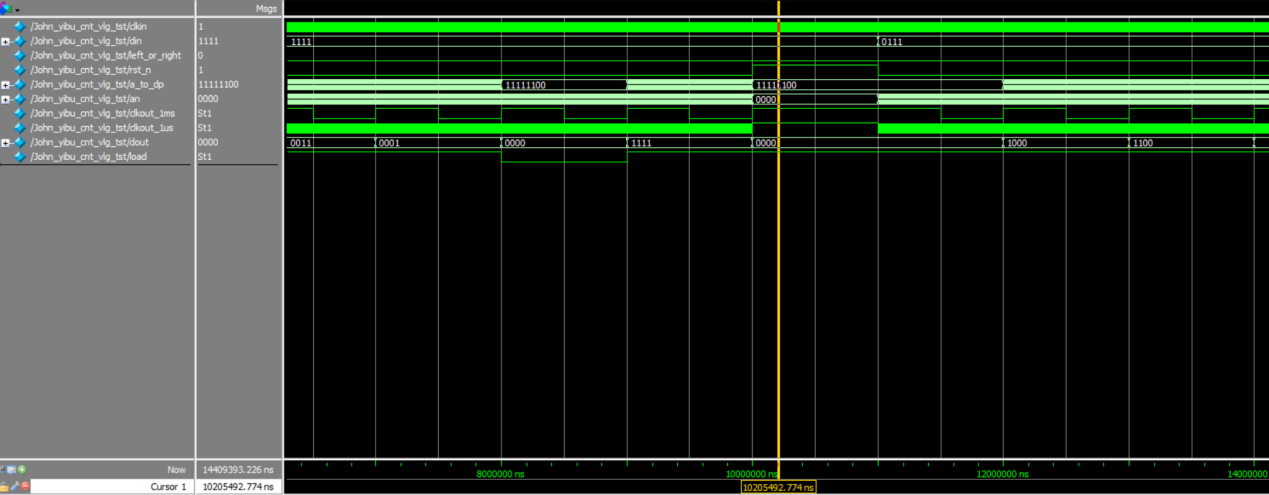
// --> end

end

endmodule

1. 仿真波形

复位有效



加载有效



左移

右移