//分频器

module clk\_div(

input clk,

input reset,

output clk\_2ms

);

reg [19:0]count;

parameter maxcount= 100000

always@(posedge clk or posedge reset)

begin

if(reset==0) begin clk\_2ms<=0,count<=0;end

else if(count<maxcount) count<=count+1;

else if(count==maxcount) begin count=0;clk\_2ms=~clk\_2ms;end

end

仿真时 T=200ns maxcount=10

实际：T=2ms=2000000ns maxcount=100000

reg pick;

always@(posedge clk\_2ms)

begin

if(pick==3) pick<=0;

else if pick<=pick+1;

end

`timescale 1ns / 1ps

module lock\_tb;

// 输入信号

reg res;

reg reset;

reg b0;

reg b1;

reg clk;

// 输出信号

wire [3:0] display;

wire led;

wire an;

wire a;

wire b;

wire c;

wire d;

wire e;

wire f;

**module pin(**

**input [3:0]din,**

**output an,**

**output reg a,**

**output reg b,**

**output reg c,**

**output reg d,**

**output reg e,**

**output reg f,**

**output reg g**

**);**

**always @(\*) begin**

**case(din)**

**4'b0000 : {a,b,c,d,e,f,g}=7'b1111110;**

**4'b0001 : {a,b,c,d,e,f,g}=7'b0110000;**

**4'b0010 : {a,b,c,d,e,f,g}=7'b1101101;**

**4'b0011 : {a,b,c,d,e,f,g}=7'b1111001;**

**4'b0100 : {a,b,c,d,e,f,g}=7'b0110011;**

**4'b0101 : {a,b,c,d,e,f,g}=7'b1011011;**

**4'b0110 : {a,b,c,d,e,f,g}=7'b1011111;**

**4'b0111 : {a,b,c,d,e,f,g}=7'b1110000;**

**4'b1000 : {a,b,c,d,e,f,g}=7'b1111111;**

**4'b1001 : {a,b,c,d,e,f,g}=7'b1111011;**

**default : {a,b,c,d,e,f,g}=7'b1111110;**

**endcase**

**end**

**endmodule**

set\_property PACKAGE\_PIN P17 [get\_ports clk]

set\_property IOSTANDARD LVCMOS33 [get\_ports clk]

set\_property PACKAGE\_PIN R15 [get\_ports b0]

set\_property PACKAGE\_PIN R17 [get\_ports b1]

set\_property IOSTANDARD LVCMOS33 [get\_ports b0]

set\_property IOSTANDARD LVCMOS33 [get\_ports b1]

set\_property PACKAGE\_PIN F6 [get\_ports led]

set\_property IOSTANDARD LVCMOS33 [get\_ports led]

set\_property PACKAGE\_PIN P15 [get\_ports reset]

set\_property IOSTANDARD LVCMOS33 [get\_ports reset]

set\_property IOSTANDARD LVCMOS33 [get\_ports a]

set\_property IOSTANDARD LVCMOS33 [get\_ports an]

set\_property IOSTANDARD LVCMOS33 [get\_ports b]

set\_property IOSTANDARD LVCMOS33 [get\_ports c]

set\_property IOSTANDARD LVCMOS33 [get\_ports d]

set\_property IOSTANDARD LVCMOS33 [get\_ports e]

set\_property IOSTANDARD LVCMOS33 [get\_ports f]

set\_property IOSTANDARD LVCMOS33 [get\_ports g]

set\_property PACKAGE\_PIN G6 [get\_ports an]

set\_property PACKAGE\_PIN D4 [get\_ports a]

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

set\_property PACKAGE\_PIN D3 [get\_ports c]

set\_property PACKAGE\_PIN F4 [get\_ports d]

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

set\_property PACKAGE\_PIN E2 [get\_ports f]

set\_property PACKAGE\_PIN D2 [get\_ports g]

module roll(

input [3:0] in7,

input [3:0] in6,

input [3:0] in5,

input [3:0] in4,

input [3:0] in3,

input [3:0] in2,

input [3:0] in1,

input [3:0] in0

input clk,

input reset,

output reg [3:0] an,

output reg [3:0] out

output [3:0] d7,

output [3:0] d6,

output [3:0] d5,

output [3:0] d4,

output [3:0] d3,

output [3:0] d2,

output [3:0] d1,

output [3:0] d0

);

reg [20:0]count;

parameter maxcount=1000000;

reg clk\_2s;

always@(posedge clk or posedge reset)

begin

if(reset) begin clk\_2s<=0;count<=0;end

else if(count<maxcount) count<=count+1;

else if(count==maxcount) begin count<=0;clk\_2s<=~clk\_2s;end

end

reg [3:0]state;

always@(posedge clk\_2s or posedge reset)

begin

if (reset) begin state<=0;end

//else if(pick==7) state<=0;

else state<=state+1;

end

**case(state)**

**3'b000 :**

**begin d7=in7;d6=in6;d5=in5;d4=in4;d3=in3;d2=in2;d1=in1;d0=in0;end**

**3'b001 :**

**begin d7=in6;d6=in5;d5=in4;d4=in3;d3=in2;d2=in1;d1=in0;d0=in7;end**

**3'b010 :**

**begin d7=in5;d6=in4;d5=in3;d4=in2;d3=in1;d2=in0;d1=in7;d0=in6;end**

**3'b011 :**

**begin d7=in4;d6=in3;d5=in2;d4=in1;d3=in0;d2=in7;d1=in6;d0=in5;end**

**3'b100 :**

**begin d7=in3;d6=in2;d5=in1;d4=in0;d3=in7;d2=in6;d1=in5;d0=in4;end**

**3'b101 :**

**begin d7=in2;d6=in1;d5=in0;d4=in7;d3=in6;d2=in5;d1=in4;d0=in3;end**

**3'b110 :**

**begin d7=in1;d6=in0;d5=in7;d4=in6;d3=in5;d2=in4;d1=in3;d0=in2;end**

**3'b111 :**

**begin d7=in0;d6=in7;d5=in6;d4=in5;d3=in4;d2=in3;d1=in2;d0=in1;end**

**default :**

**begin d7=in7;d6=in6;d5=in5;d4=in4;d3=in3;d2=in2;d1=in1;d0=in0;end**

**endcase**

end

endmodule