

西 安 电 子 科 技 大 学

考试时间 _____ 分钟

试 题

题号	一	二	三	四	五	六	七	八	九	十	总分
分数											

1. 考试形式：闭（开）卷； 2. 本试卷共 四 大题，满分 100 分。

班级 _____ 学号 _____ 姓名 _____ 任课教师 _____

一、 选择题（每题 2 分，共 18 分）

1. 下面哪个是可以用 verilog 语言进行描述，而不能用 VHDL语言进行描述的级别？（ A ）

(A) 开关级 (B) 门电路级 (C) 体系结构级 (D) 寄存器传输级

2. 在 verilog 中，下列语句哪个不是分支语句？（ D ）

(A) if-else (B) case (C) casez (D) repeat

3 . 下列哪些 Verilog 的基本门级元件是多输出（ D ）

(A) nand (B) nor (C) and (D) not

4 . Verilog 连线类型的驱动强度说明被省略时，则默认的输出驱动强度为（ B ）

(A) supply (B) strong (C) pull (D) weak

5. 元件实例语句 “ notif1 #(1:3:4,2:3:4,1:2:4) U1(out,in,ctrl); ” 中截至延迟的典型值为（ B ）

(A) 1 (B) 2 (C) 3 (D) 4

6 . 已知 “ a=1b ' 1; b=3b'001; ” 那么 {a,b} = （ C ）

(A) 4b '0011 (B) 3b '001 (C) 4b '1001 (D) 3b '101

7. 根据调用子模块的不同抽象级别，模块的结构描述可以分为 (**ABC**)

(A) 模块级 (B) 门级 (C) 开关级 (D) 寄存器级

8. 在 verilog 语言中， $a=4b'1011$ ，那么 $\&a=$ (D)

(A) $4b'1011$ (B) $4b'1111$ (C) $1b'1$ (D) $1b'0$

9. 在 verilog 语言中整型数据与 (C) 位寄存器数据在实际意义上是相同的。

(A) 8 (B) 16 (C) 32 (D) 64

二、简答题 (2 题，共 16 分)

1. Verilog HDL 语言进行电路设计方法有哪几种 (8 分)

- 1、自上而下的设计方法 (Top-Down)
- 2、自下而上的设计方法 (Bottom-Up)
- 3、综合设计的方法

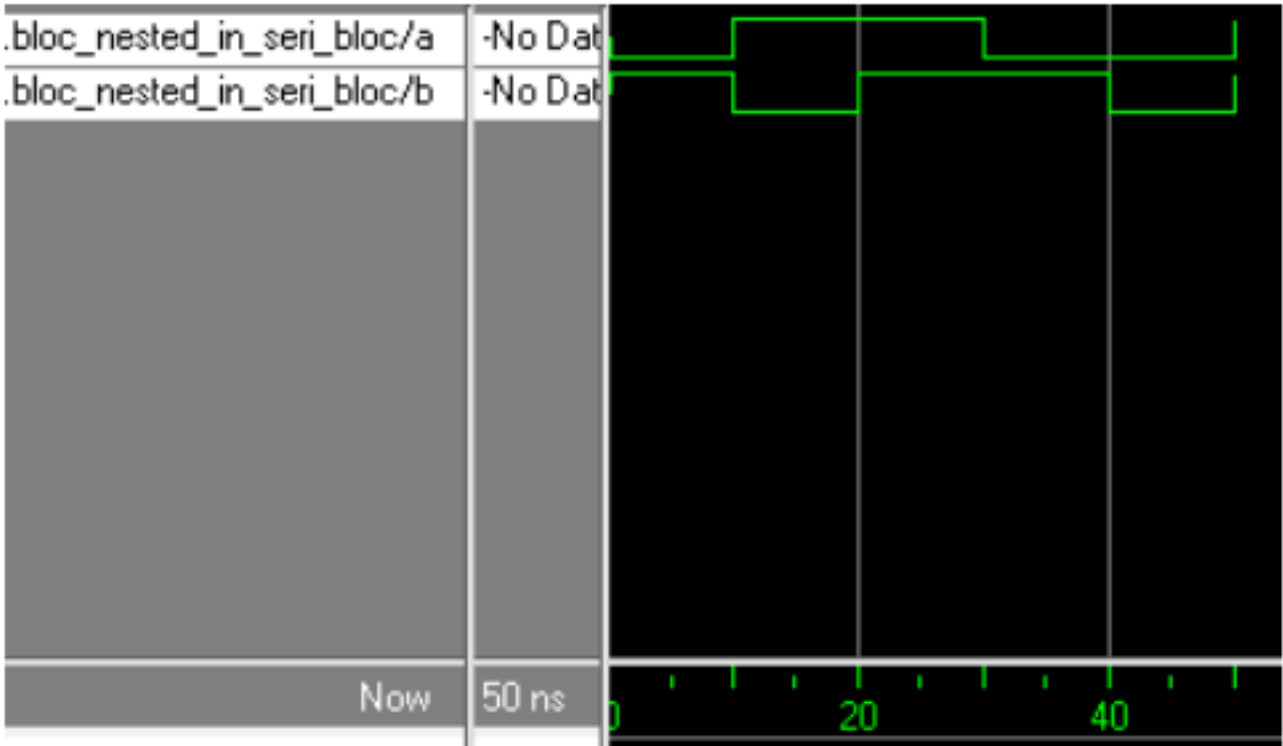
2. specparam语句和 parameter语句在参数说明方面不同之处是什么 (8分)。

- 1、. specparam 语句只能在延时的格式说明块 (specify 块) 中出现，而 parameter 语句则不能在延时说明块内出现
- 2、由 specparam 语句进行定义参数只能是延时参数，而由 parameter 语句定义参数则可以是任何数据类型的参数
- 3、由 specparam 语句定义的延时参数只能在延时说明块内使用，而由 parameter 语句定义参数则可以在模块内 (该 parameter 语句之后) 的任何位置说明

三、画波形题（每题 8 分，共 16 分）

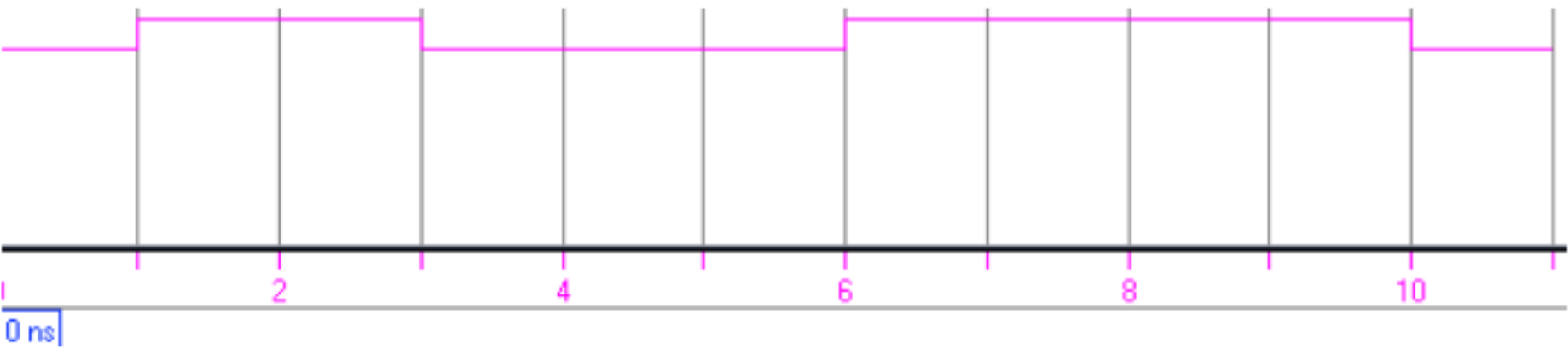
1. 根据下面的程序，画出产生的信号波形（ 8分）

```
module para_bloc_nested_in_seri_bloc(A,B);
  output A,B;
  reg a,b;
  initial
  begin
    A=0;
    B=1;
    #10 A=1;
    fork
      B=0;
      #10 B=1;
      #20 A=0;
    join
    #10 B=0;
    #10 A=1;
    B=1;
  end
endmodule
```



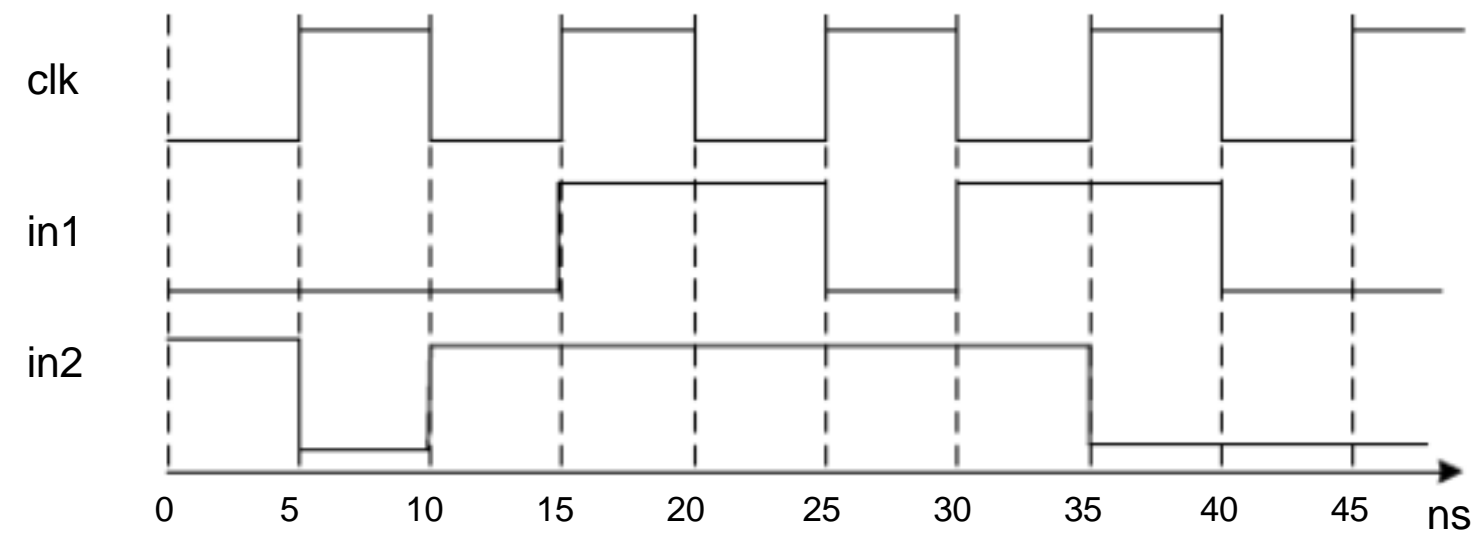
2. 根据下面的程序，画出产生的信号波形（ 8分）

```
module signal_gen1(d_out);
output d_out;
reg d_out;
initial
begin
    d_out=0;
    #1 d_out=1;
    #2 d_out=0;
    #3 d_out=1;
    #4 d_out=0;
end
endmodule
```



四、 程序设计（ 4 题，共 50 分）

1. 试用 verilog语言产生如下图所示的测试信号（ 12分）



```
module signal_gen9(clk,in1,in2);
output in1,in2,clk;
reg in1,in2,clk;
initial
begin
    in1=0;
    in2=1;
```

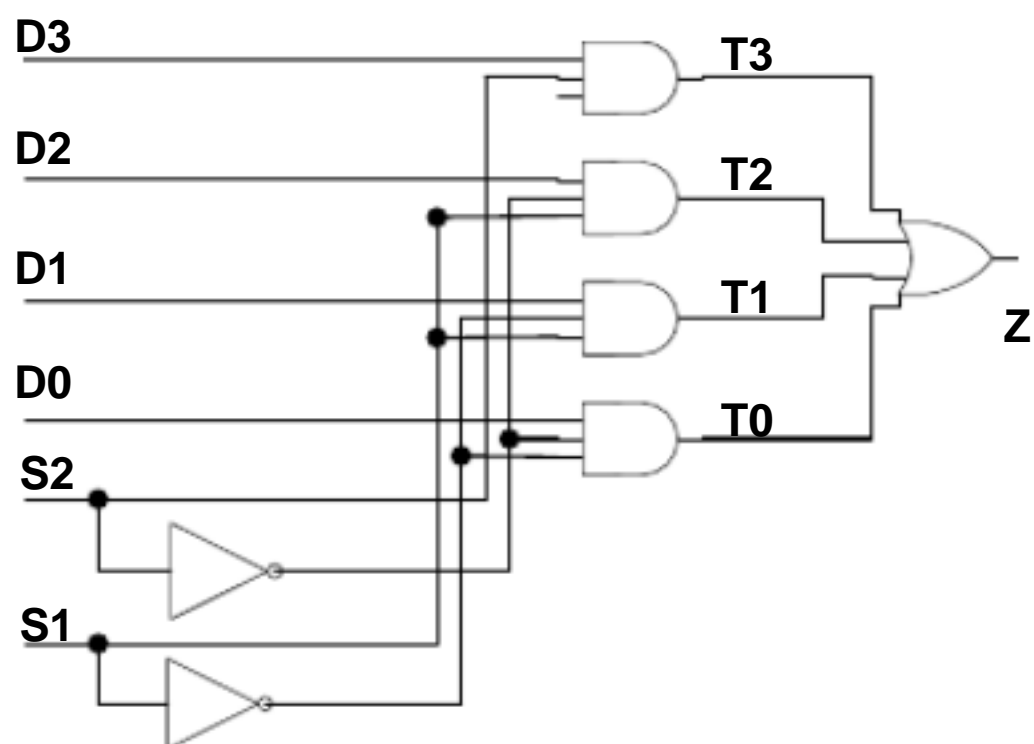
```

        clk=0;
    end
initial
    begin
        #15 in1=1
        #10 in1=0;
        #5 in1=1;
        #10 in1=0;

    end
initial
    begin
        #5 in2=0;
        #5 in2=1;
        #25 in2=0;
    end
    end
always
    begin
        #5 clk=~clk;
    end
endmodule

```

2 . 试用 **verilog** 语言，利用内置基本门级元件，采用结构描述方式生成如图所示的电路（ 12 分）



```

module MUX4x1(Z,D0,D1,D2,D3,S0,S1);

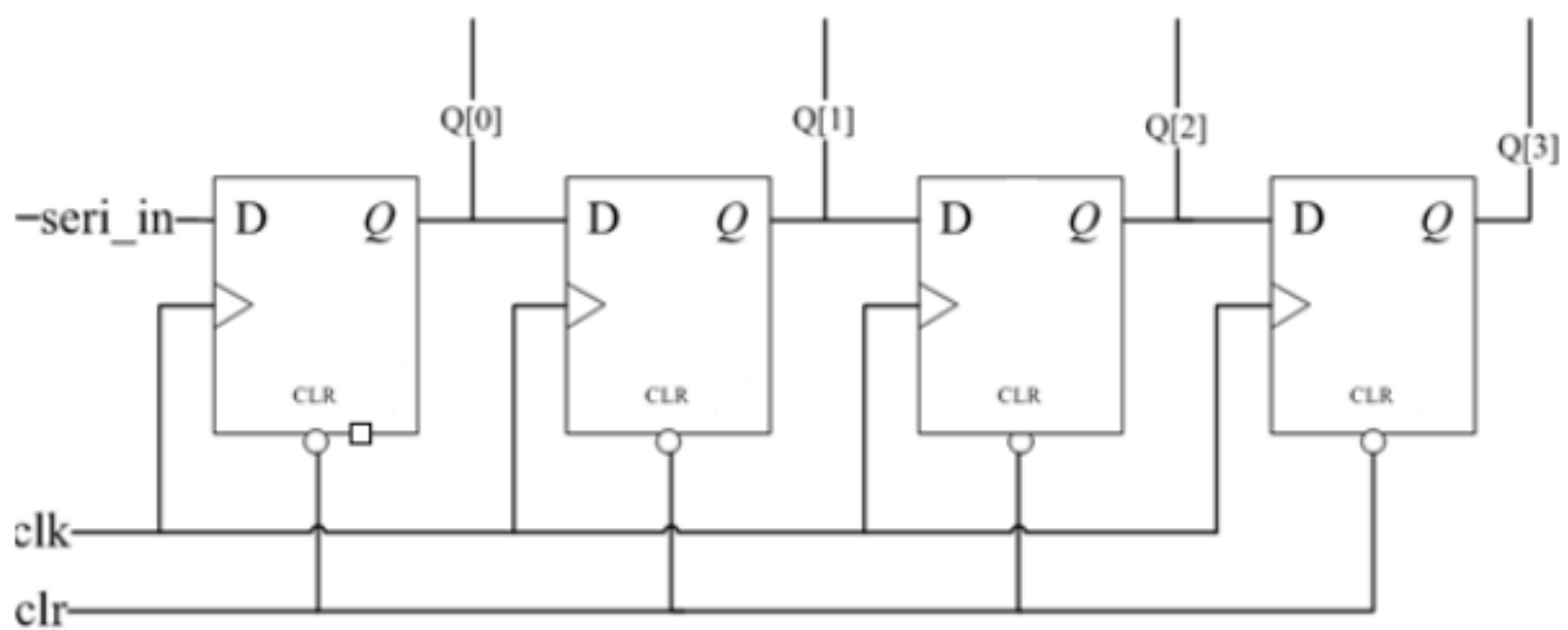
```

```

output Z;
inout D0,D1,D2,D3,S0,S1;
and u0 (T0, D0, S0bar, S1bar) ,
      u1 ( T1, D1, S0bar, S1),
      u2 (T2, D2, S0, S1bar),
      u3 (T3, D3, S0, S1) ,
not u4 (S0bar, S0),
    u5 (S1bar, S1);
or  u6 (Z, T0, T1, T2, T3);
endmodule

```

3. 试用 **verilog**语言描述：图示为一个 **4位移位寄存器**，是由四个**D**触发器（分别设为 **U1**，**U2**，**U3**，**U4**）构成的。其中 **seri_in** 是这个移位寄存器的串行输入； **clk**为移位时脉冲输入； **clr**为清零控制信号输入； **Q[1]~Q[3]**则为移位寄存器的并行输出。（14分）



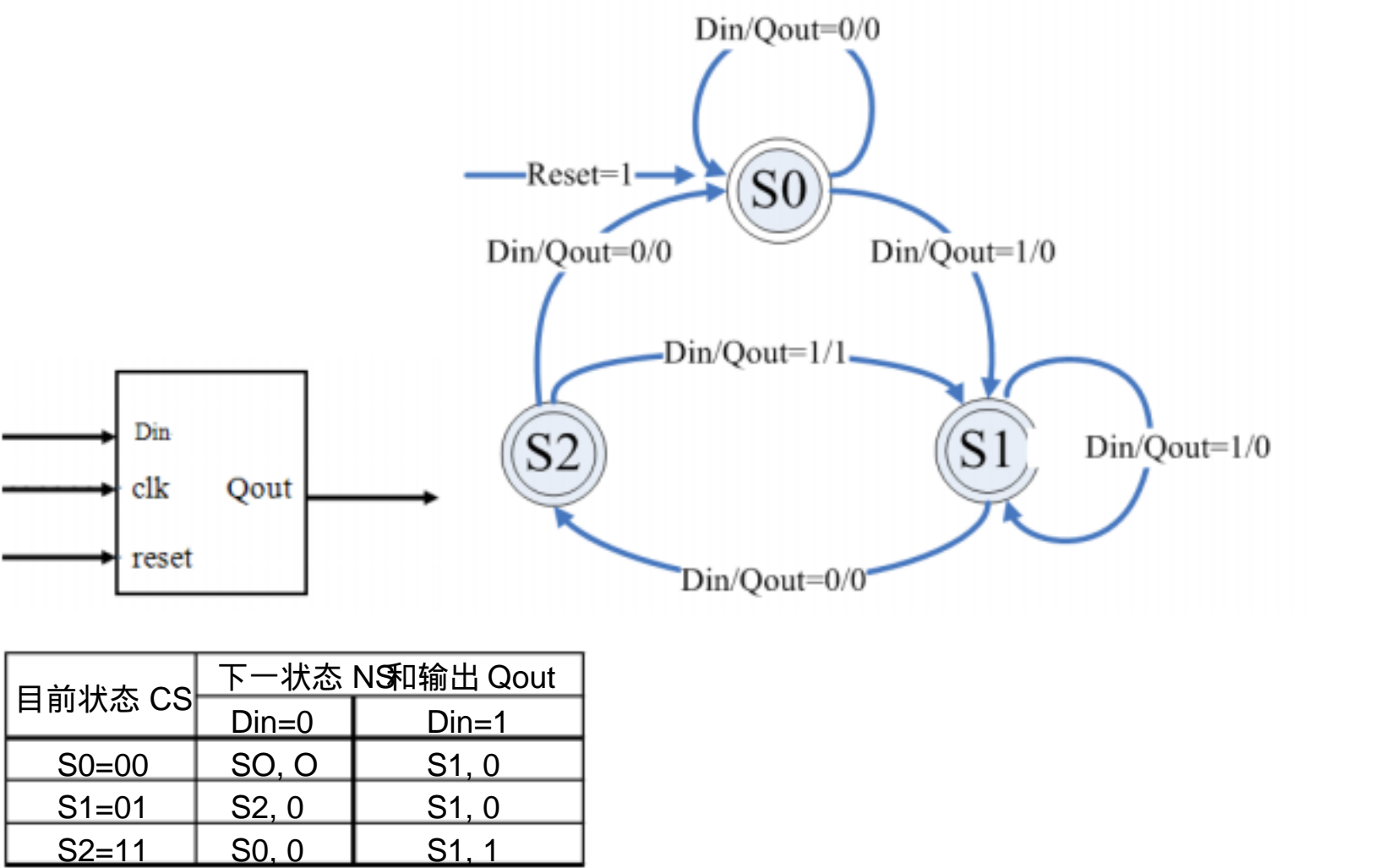
```

module d_flop(q,d,clr,clk);
    output q;
    input d,clr,clk;
    reg q;
    always @(clr)
    if(!clr) assign q=0;
    else deassign q;
    always @(negedge clk)
    q=d;
endmodule

```

```
module shifter(seri_in,clk,clrb,Q);
    input seri_in,clk,clrb;
    output[3:0] Q;
    d_flop U1(Q[0],seri_in,clrb,clk),
           U2(Q[1],Q[0],clrb,clk),
           U3(Q[2],Q[1],clrb,clk),
           u4(Q[3],Q[2],clrb,clk);
endmodule
```

4 . 利用有限状态机，以格雷码编译方式设计一个从输出信号序列中检测出 101 信号的电路图，其方块图、状态图和状态表如图表示。（ 12 分）



```
module melay(clk,Din,reset,Qout);
    input clk,reset;
    input Din;
    output Qout;
    reg Qout;
    parameter[1:0] S0=2'b00,S1=2'b01,S2=2'b11;
    reg[1:0] CS;
```

```

reg[1:0] NS;

always @(posedge clk or posedge reset)
begin
    if(reset==1'b01)
        CS=S0;
    else CS=NS;
end

always @(CS or Din)
begin
    case(CS)
        S0:begin
            if(Din==1'b0)
            begin
                NS=S0;
                Qout=1'b0;
            end
            else
            begin
                NS=S1;
                Qout=1'b0;
            end
        end
        S1:begin
            if(Din==1'b0)
            begin
                NS=S2;
                Qout=1'b0;
            end
            else
            begin
                NS=S1;
                Qout=1'b0;
            end
        end
        S2:begin
            if(Din==1'b0)
            begin
                NS=S0;
                Qout=1'b0;
            end
        end
    endcase
end

```



```
        end
    else
        begin
            NS=S1;
            Qout=1'b0;
        end
    end
end

endcase
end
endmodule
```