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

第 2 页 共 8 页

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语句进行定义的参数只能是延时参数，而由parameter语句定义的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语句定义的参数则可以在模块内（该parameter语句之后）的任何位置说明

第 3 页 共 8 页

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

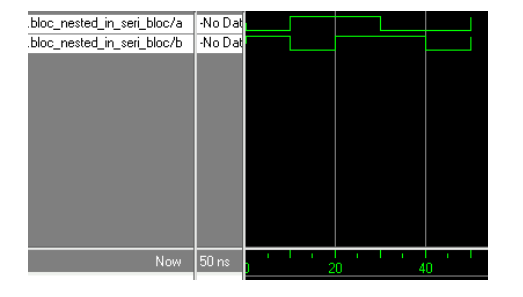
        #10 B=0;

   #10 A=1;

   B=1;

 end

   endmodule



第 4 页 共 8 页

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;

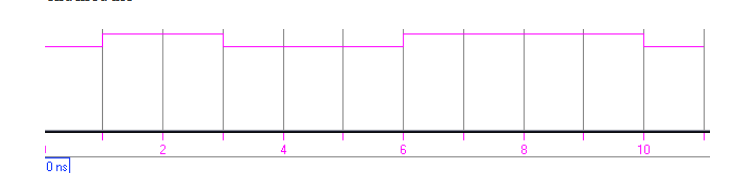
 #1 d\_out=1;

#2 d\_out=0;

#3 d\_out=1;

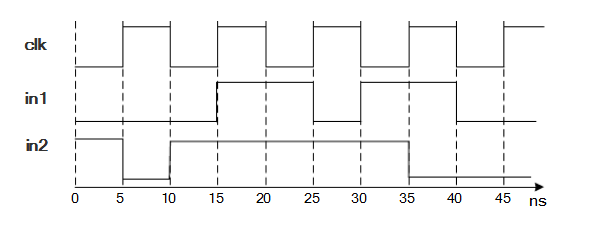
 #4 d\_out=0; end

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

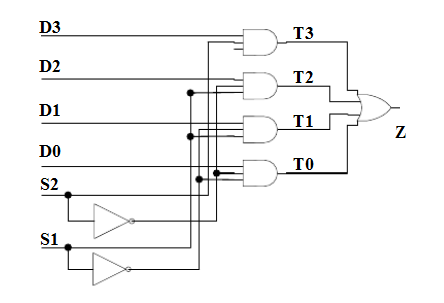
#5 in2=0;     #5 in2=1;     #25 in2=0;   end

always   begin

    #5 clk=~clk;   end

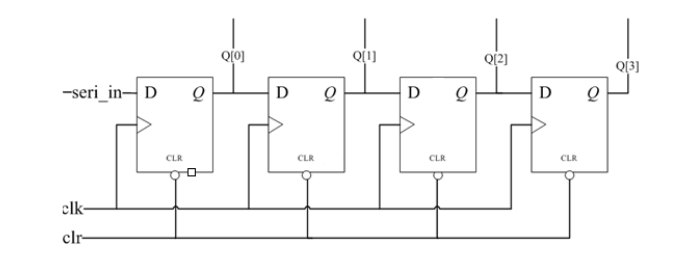
endmodule

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);

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;

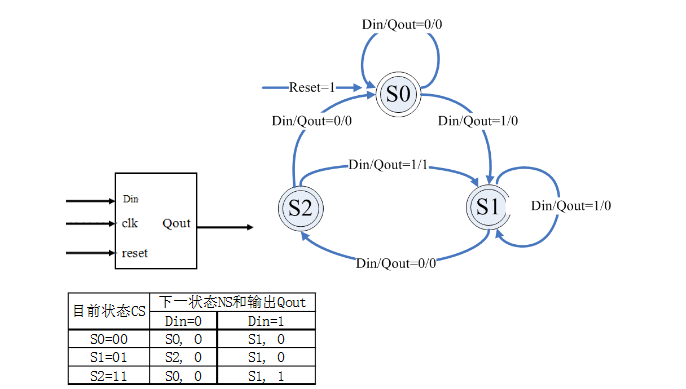
    always @(clr)     if(!clr) assign q=0;     else deassign q;

    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）



Din=0Din=1S0=00

SO, OS1, 0S1=01S2, 0S1, 0S2=11

S0, 0S1, 1

下一状态NS和输出Qout目前状态CS

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:beign

            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:beign

            if(Din==1'b0)               begin

                 NS=S0;                  Qout=1'b0;

              end             else               begin

                 NS=S1;                  Qout=1'b0;               end            end

        endcase         end

endmodule

、填空题（10分,每小题1分）

1. 用EDA技术进行电子系统设计的目标是最终完成                 的设计与实现。 2. 可编程器件分为                和                   。

3. 随着EDA技术的不断完善与成熟，                 的设计方法更多的被应用于Verilog HDL设计当中。

4. 目前国际上较大的PLD器件制造公司有               和                 公司。 5. 完整的条件语句将产生           电路，不完整的条件语句将产生          电路。 6. 阻塞性赋值符号为                  ，非阻塞性赋值符号为                   。  二、选择题 (10分，每小题2分)

1. 大规模可编程器件主要有FPGA、CPLD两类，下列对FPGA结构与工作原理的描述中，正确的是                。

A．FPGA全称为复杂可编程逻辑器件；

B．FPGA是基于乘积项结构的可编程逻辑器件；

C．基于SRAM的FPGA器件，在每次上电后必须进行一次配置；

D．在Altera公司生产的器件中，MAX7000系列属FPGA结构。

2. 基于EDA软件的FPGA / CPLD设计流程为：原理图/HDL文本输入 →  综合    →\_\_\_     \_\_→          →适配→编程下载→硬件测试。正确的是            。 ①功能仿真 ②时序仿真 ③逻辑综合 ④配置 ⑤分配管脚              A．③①  B．①⑤  C．④⑤  D．④②

3. 子系统设计优化，主要考虑提高资源利用率减少功耗（即面积优化），以及提高运行速度（即速度优化）；指出下列哪些方法是面积优化             。   ①流水线设计 ②资源共享 ③逻辑优化 ④串行化 ⑤寄存器配平

⑥关键

路径法

A．①③⑤ B．②③④     C．②⑤⑥  D．①④⑥

 4. 下列标识符中，\_\_\_\_\_\_\_\_\_\_是不合法的标识符。

A．9moon     B．State0    C． Not\_Ack\_0  D． signall

 5. 下列语句中，不属于并行语句的是：\_\_\_\_\_\_\_

A．过程语句     B．assign语句  C．元件例化语句 D．case语句

三、EDA名词解释（10分） 写出下列缩写的中文含义：

ASIC：                          RTL：                      FPGA：                          SOPC：                     CPLD：              LPM：                       EDA：                      IEEE：                      IP：                             ISP：

四、简答题（10分）

1. 简要说明仿真时阻塞赋值与非阻塞赋值的区别（本题4分）。

2. 简述有限状态机FSM分为哪两类？有何区别？有限状态机的状态编码风格主要有哪三种？FSM的三段式描述风格中，三段分别描述什么？（本题6分）

五、程序注解（20分,每空1分）

module  AAA  ( a ,b );

output  a

input [6:0] b

reg[2:0] sum;

integer i;

reg  a

always @ (b)

begin

sum = 0;

for(i = 0;i<=6;i = i+1)

if(b[i])

sum = sum+1;

if(sum[2])  a = 1;

else       a = 0;

end

endmodule

 本程序的逻辑功能是：                                                 。

四、VerilogHDL语言编程题（1、2小题10分，3小题20分）     要求：写清分析设计步骤和注释。

1.  试用Verilog HDL描述一个带进位输入、输出的8位全加器。 端口：A、B为加数，CI为进位输入，S为和，CO为进位输出

2. 编写一个带异步清零、异步置位的D触发器。

端口：CLK为时钟，D为输入，CLK为清零输入端，SET为置位输入端；Q输出端。

3. 设计一个带有异步复位控制端和时钟使能控制端的10进制计数器。 端口设定如下：

输入端口：CLK：时钟，RST：复位端，EN：时钟使能端，LOAD：置位控制端， DIN：置位数据端；

输出端口：COUT：进位输出端，DOUT：计数输出端。

4

一、填空题（每空2分，共20分） 1、   ASIC

2、     FPGA     和 CPLD   。 3、 自顶向下

4、   Altera  和  Xilinx   5、 组合     时序   6、   =    <=

二、选择题 (10分，每小题2分)

1、C   2、 B   3、B    4、 A   5、D  三、EDA名词解释（10分） ASIC      专用集成电路         RTL

       寄存器传输级

FPGA      现场可编程门阵列     SOPC         可编程片上系统

CPLD      复杂可编程逻辑器件   LPM              参数可定制宏模块库 EDA

 电子设计自动化       IEEE              电子电气工程师协会

IP       知识产权核           ISP                在线系统编程

四、简答题（10分）

 1、简要说明仿真时阻塞赋值与非阻塞赋值的区别（本题4分）。 答：非阻塞（non-blocking)赋值方式 ( b<= a)：

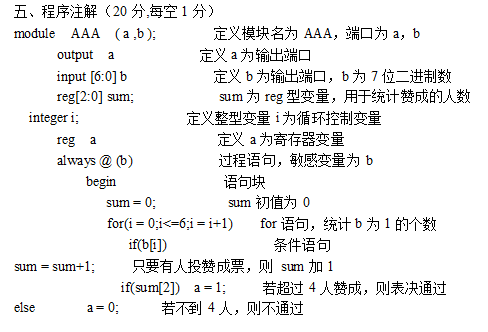
b的值被赋成新值a的操作, 并不是立刻完成的，而是在块结束时才完成； 块内的多条赋值语句在块结束时同时赋值； 硬件有对应的电路。

阻塞（blocking)赋值方式 ( b = a)： b的值立刻被赋成新值a；

完成该赋值语句后才能执行下一句的操作； 硬件没有对应的电路，因而综合结果未知。

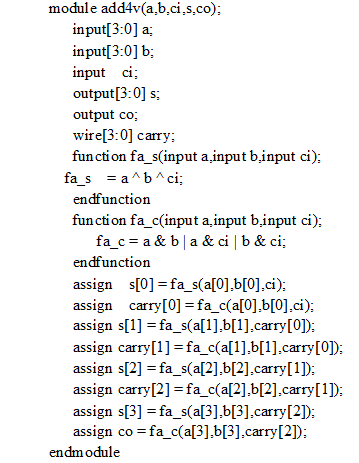
2、 简述有限状态机FSM分为哪两类？有何区别？有限状态机的状态编码风格主要有哪三种？FSM的三段式描述风格中，三段分别描述什么？（本题6分）

答：Mearly型，Moore型；前者与输入与当前状态有关，而后者只和当前状态有关；Binary，Gray，One-Hot编码；分别为状态保存，状态切换，输出； 五、程序注解（20分,每空1分）



 本程序的逻辑功能是：     7人投票表决器。

 六、VerilogHDL语言编程题（1、2小题10分，3小题20分）     要求：写清分析设计步骤和注释。 1.试用Verilog HDL描述一个带进位输入、输出的8位全加器。 端口：A、B为加数，CIN为进位输入，S为和，COUT为进位输出



module add4v(a,b,ci,s,co);    input[3:0] a;    input[3:0] b;    input  ci;    output[3:0] s;    output co;

   wire[3:0] carry;

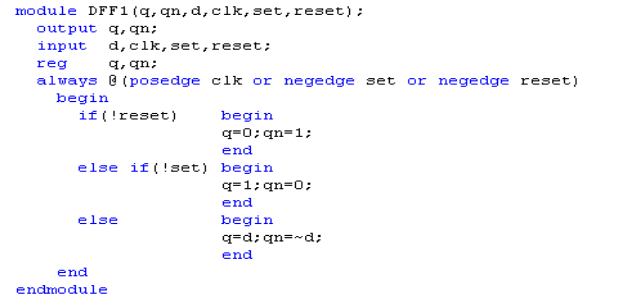
   function fa\_s(input a,input b,input ci);   fa\_s  = a ^ b ^ ci;    endfunction

   function fa\_c(input a,input b,input ci);       fa\_c = a & b | a & ci | b & ci;    endfunction

   assign  s[0] = fa\_s(a[0],b[0],ci);    assign  carry[0] = fa\_c(a[0],b[0],ci);    assign s[1] = fa\_s(a[1],b[1],carry[0]);    assign carry[1] = fa\_c(a[1],b[1],carry[0]);    assign s[2] = fa\_s(a[2],b[2],carry[1]);    assign carry[2] = fa\_c(a[2],b[2],carry[1]);    assign s[3] = fa\_s(a[3],b[3],carry[2]);    assign co = fa\_c(a[3],b[3],carry[2]);

endmodule

2.编写一个带异步清零、异步置位的D触发器。



 6

3.设计一个带有异步复位控制端和时钟使能控制端的10进制计数器。

mdule CNT10 (CLK,RST,EN,LOAD,COUT,DOUT,DATA); input CLK   ; input EN   ; input RST  ; input LOAD  ; input [3:0] DATA  ; output [3:0] DOUT  ; output COUT  ; reg [3:0] Q1  reg COUT

assign DOUT = Q1;

always @(posedge CLK or negedge RST) begin if (!RST)   Q1 <= 0; else  if (EN) begin

if (!LOAD)   Q1 <= DATA; else if (Q1<9)  Q1 <= Q1+1; else     Q1 <= 4'b0000; end end

always @(Q1)

if (Q1==4'h9)  COUT = 1'b1; else          COUT = 1'b0; endmodule

 6