

2020 数字逻辑试题回忆

一、填空题

1. 大项小项的性质：

$$\sum_{i=0}^n m_i = 1$$

$$\prod_{i=0}^n M_i = 0$$

所有小项的和是 1，所有大项的乘积是 0

2. 阻塞赋值和非阻塞赋值的表示($<=$, $=$)

3. 补码：给一个数，求补码；补码的范围；溢出；判断是否溢出

4. 德摩根定律（反演律）

5. 跟选择器功能相反的是：数据分配器

6. $4'b0111$ 最低位取反的操作： $4'b0111_4'b0001$ （横线上填符号）

二、画图题

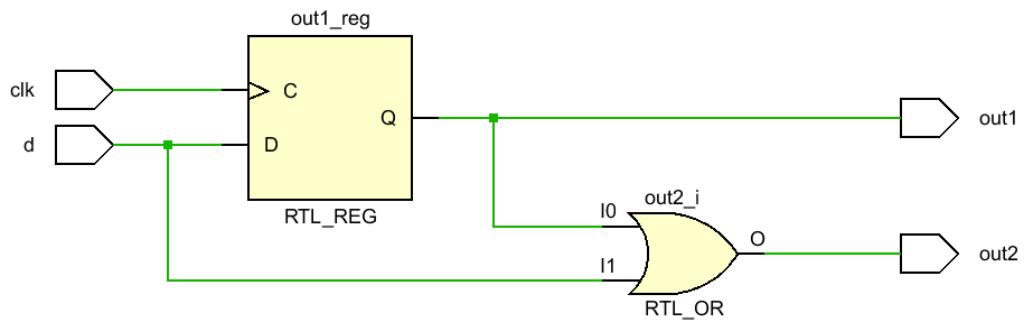
1. 一个 always 触发器赋值，一个 assign 语句赋值

代码大致长这样：

```
module test(clk,d,out1,out2);
    input clk,d;
    output reg out1;
    output out2;

    assign out2 = out1|d;
    always@(posedge clk)
        | out1 <= d;|
endmodule
```

RTL 图长这样：



2. 锁存器画法

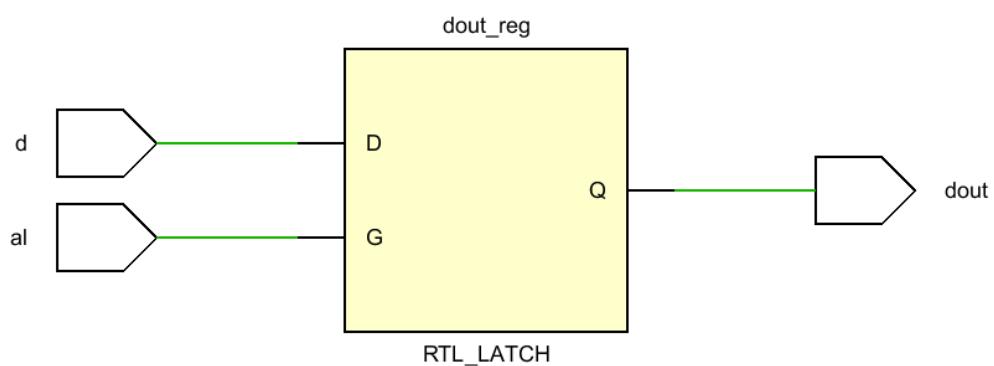
代码大概长这样

```

module test(al,d,dout);
    input al,d;
    output reg dout;
    always@(*)
        if (al)
            dout <= d;
endmodule

```

RTL 图大概长这样



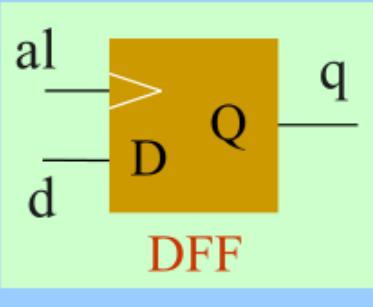
生成了不想要的锁存器：

always@ (al or d)

begin

if(al) q<=d;

end



当 al 为 0 时， q 保持原值！

三、电路设计

1. 组合逻辑设计

灯开关控制（类似于举重裁判控制电路）：三个开关，奇数个开关打开，灯亮；否则灯不亮

先画出真值表，求得输出函数

1) 用 74LS138 3-8 译码器实现

见 PPT 里面译码器如何实现逻辑函数

2) 2 选 1 多路选择器实现

3) 4 选 1 多路选择器实现

见 PPT 里面数据选择器如何实现逻辑函数

2.设计七进制计数器（我的方法有点奇怪）

输入：时钟信号 clk；使能信号 w, w=1 时计数, w=0 是不计数；复位信号
输出：计数 out, 进位 q

三、代码题

1.模 60 计数器

见这个 PPT

 10Verilog 组合电路与时序电路 (基本器...)

```
module count60(qout, cout,
  data, load, cin, reset, clk);
  output [7:0] qout;
  output cout;
  input [7:0] data;
  input load, cin, clk, reset;
  reg [7:0] qout;

  always @ (posedge clk)
  begin
    if (reset) qout <= 0;
    else if (load) qout <= data;
    else if (cin) begin
      if(qout[3:0] == 9) begin
        qout[3:0] <= 0;
        if(qout[7:4] == 5) qout[7:4] <= 0;
        else qout[7:4] <= qout[7:4]+1;
      end
      else qout[3:0] <= qout[3:0]+1;
    end
    end
    assign cout = ((qout == 8'h59)&cin)?1:0;
endmodule
```

```
module count60(qout, cout,
  data, load, cin, reset, clk);
  output [7:0] qout;
  output cout;
  input [7:0] data;
  input load, cin, clk, reset;
  reg [7:0] qout;
```

2.四位并入串出寄存器

有使能、载入、复位信号

大体跟这个差不多，有细节差异

```
module piso4 ( dout, clk,clr,din); //并入串出
    output dout;
    input clk, clr;
    input[3:0] din;
    reg dout;
    reg[1:0] cnt; //初始为0
    reg[3:0] q;
    always @ (posedge clk)
    begin
        if(clr) begin q <= 4'b0000; end
        else begin if(cnt > 0) begin q[3:1] <= q[2:0];
                    cnt <= cnt -1; end
                else if(cnt == 2'b00)
                    begin q <= din; cnt<=2'b11; end
                end
        dout <= q[3];
    end
endmodule
```

3.状态机

设计一个饮料机，每次投币 1、2、5 分，饮料价格 5 分。输出有是否找零、是否购买饮料。