

1、ASIC 设计流程以及用到的工具？

前端设计：

需求分析（文档设计）—— 电路设计（RTL 代码设计）—— 功能仿真 —— FPGA

验证 —— 综合

后端设计：

布局布线 —— 时序分析 —— 逻辑优化

物理设计：

硅片设计 —— 生产

2、ASIC 前端设计流程以及用到的工具？

前端设计：

需求分析 —— 电路设计（RTL 代码设计） —— 功能仿真 —— FPGA 验证 —— 综合

<https://wenku.baidu.com/view/c2b8bc6448d7c1c708a145db.html>

3、建立时间和保持时间问题？

4、FIFO 最小深度计算问题?

5、格雷码计数器问题?

```
module gray_count(  
    input clk,  
    input rst_n,  
    output [3:0] count_gray  
);  
    reg [3:0] count_bin;  
    always@(posedge clk or negedge rst_n)begin  
        if(!rst_n)  
            count_bin <= 0;  
        else  
            count_bin <= count_bin + 1'b1;  
        end
```

```
assign count_gray = (count_bin >> 1) ^ count_bin;
```

```
endmodule
```

6、case 条件没列全生成锁存器的电路如何画?

```
reg [1:0] sel;
```

```
reg out;
```

```
always@(*) begin
```

```
case(sel)
```

```
2'b00: out = a;
```

```
2'b01: out = b;
```

```
2'b10: out = c;
```

```
2'b11: out = d;
```

```
endcase
```

```
end
```

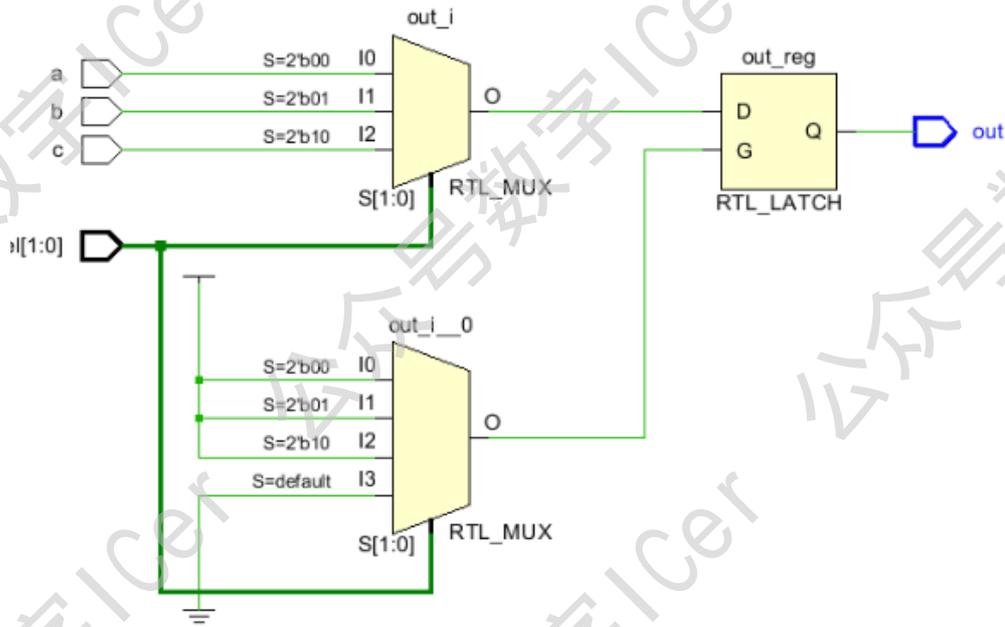
```
reg [1:0] sel;  
  
reg out;  
  
always@(*) begin  
  
case(sel)  
  
2'b00: out = a;  
  
2'b01: out = b;  
  
endcase  
  
end
```

先确定必定产生锁存器

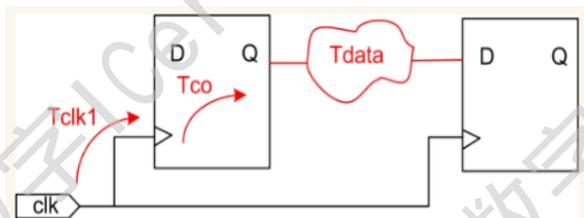
然后还是case

那就用四选一多路选择器

列了两种情况，那就肯定有两种情况到不了，那就保持上一个原值呗，如何保持呢？再多用一个多路选择器



7、数据到达时间、数据要求时间以及建立时间裕量和保持时间裕量？

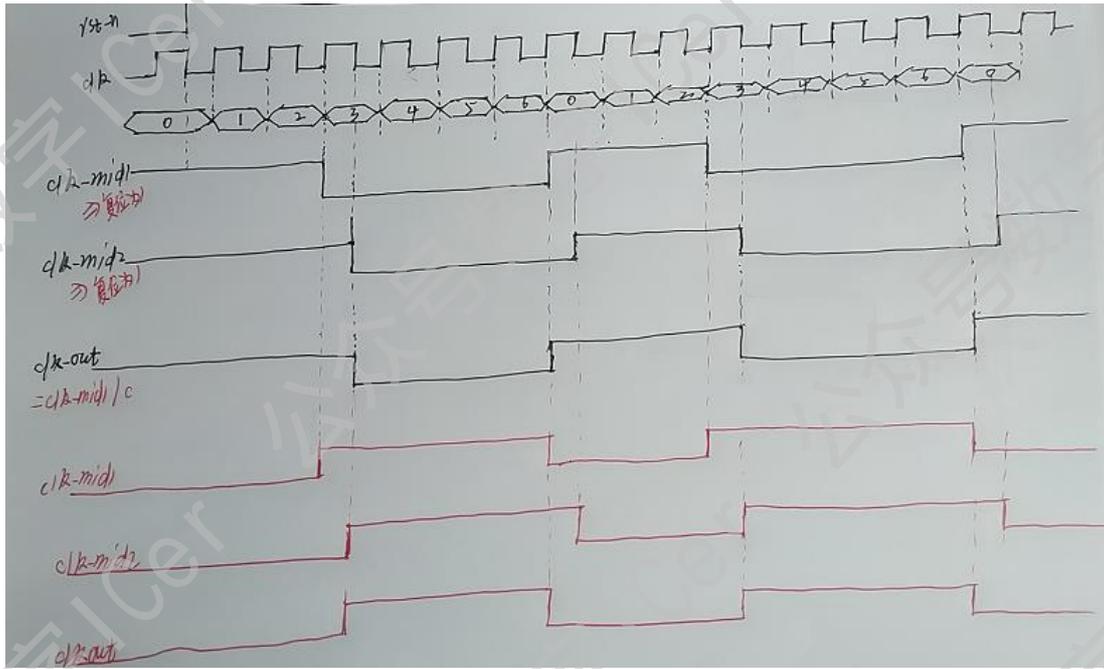


数据到达时间：

数据要求时间：

//这个题目 就不做了

8、写出一个 7 分频的电路？或画出波形？



代码:

```

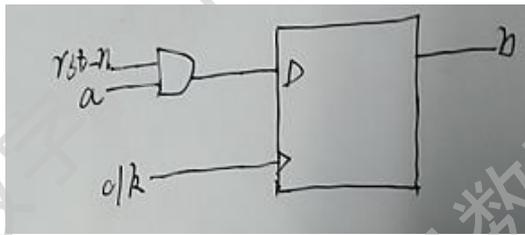
module div_7(
input clk,
input rst_n,
output clk_out
);

```

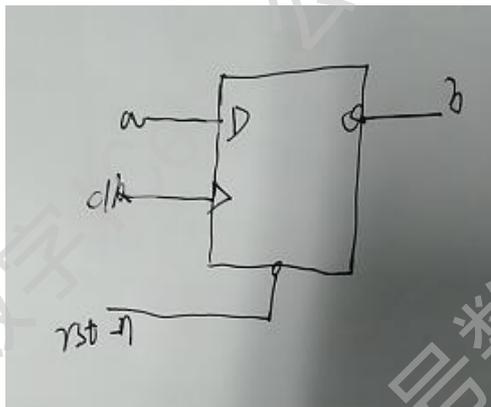
endmodule

9、同步复位和异步复位? 异步复位同步释放? 以4bit 为例的 DFF 画出, 电路图

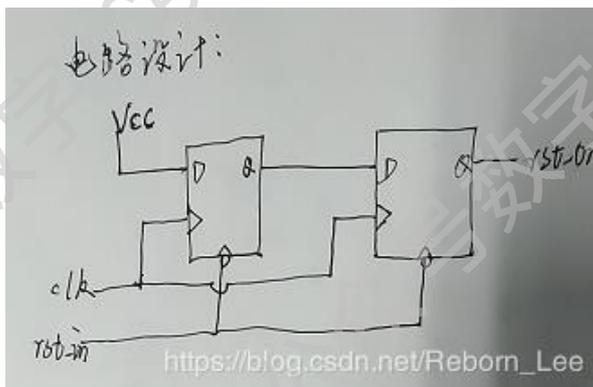
同步复位:



异步复位:



异步复位, 同步释放



如果是 4bit 的触发器:

什么是阻塞赋值？什么是非阻塞赋值？并举例说明：

例 1、（美满电子）解释下面的语句：

a: X = #5 B+C;

解答：语句内部延迟，可以把右式的值赋值给左式的目标之前延迟一段时间；

等同于：

```
begin
```

```
Mid = B+C;
```

```
#5
```

```
X = Mid;
```

```
end
```

b: X <= #5 B + C;

对于这种非阻塞过程性赋值，上面的语句表示，在第 5 个时间单位，X 被赋值为 B+C；

例 2、（综合例子）画出波形？

阻塞赋值：

```
initial begin
```

```
a = #5 0;
```

```
a = #4 1;
```

```
a = #10 0;
```

```
end
```

非阻塞赋值:

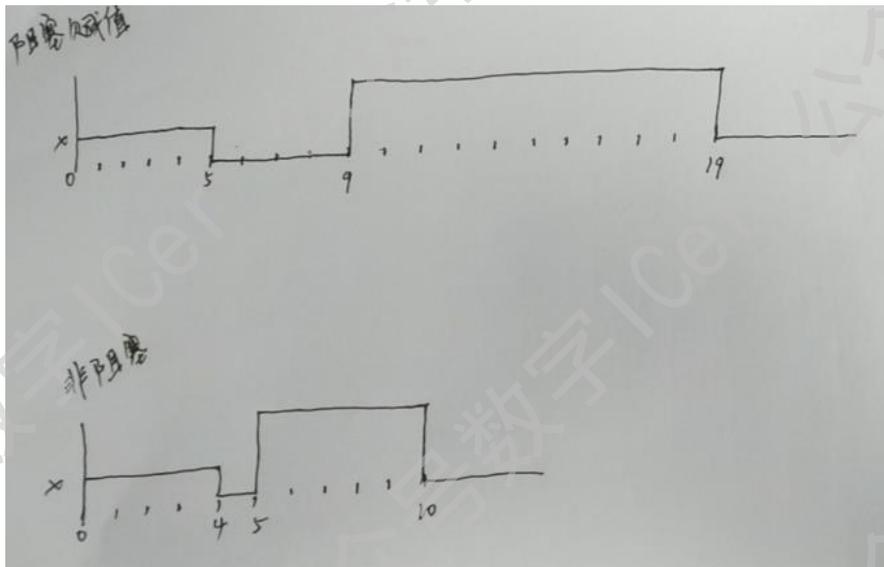
```
initial begin
```

```
  a <= #5 1;
```

```
  a <= #4 0;
```

```
  a <= #10 0;
```

```
end
```



范范总结:

“
一个8bit宽的AFIFO，输入时钟为100MHz，输出时钟为95MHz，设一个package为4Kbit，且两个package之间的发送间距足够大。求AFIFO的深度？”

已知量：

传输的最大package = 4 Kbit
读时钟速率 = 95 MHz
写时钟速率 = 100 MHz

套用公式：

$t = \text{写入的最大package} / \text{写时钟速率} = 4 \text{ Kbit} / 100\text{MHz}$
写入的最大package = FIFO容量 + 读出的数据量 $\rightarrow 4 \text{ Kbit} = x + t * 95\text{MHz}$
 $\Rightarrow \text{FIFO容量} x = 205\text{bit}$ 约 26 x 8bit

即FIFO深度为26

形而上者谓之道 形而下者谓之器

1、建立时间与保持时间的公式，以及裕量公式