

点拨·FPGA之 至多1个乘法器延时的流水线练习思路

点透学习误区 拨出设计精髓

主讲：潘文明



明德扬
科技·教育

QQ群： 97925396

官 网： <http://www.mdy-edu.com>

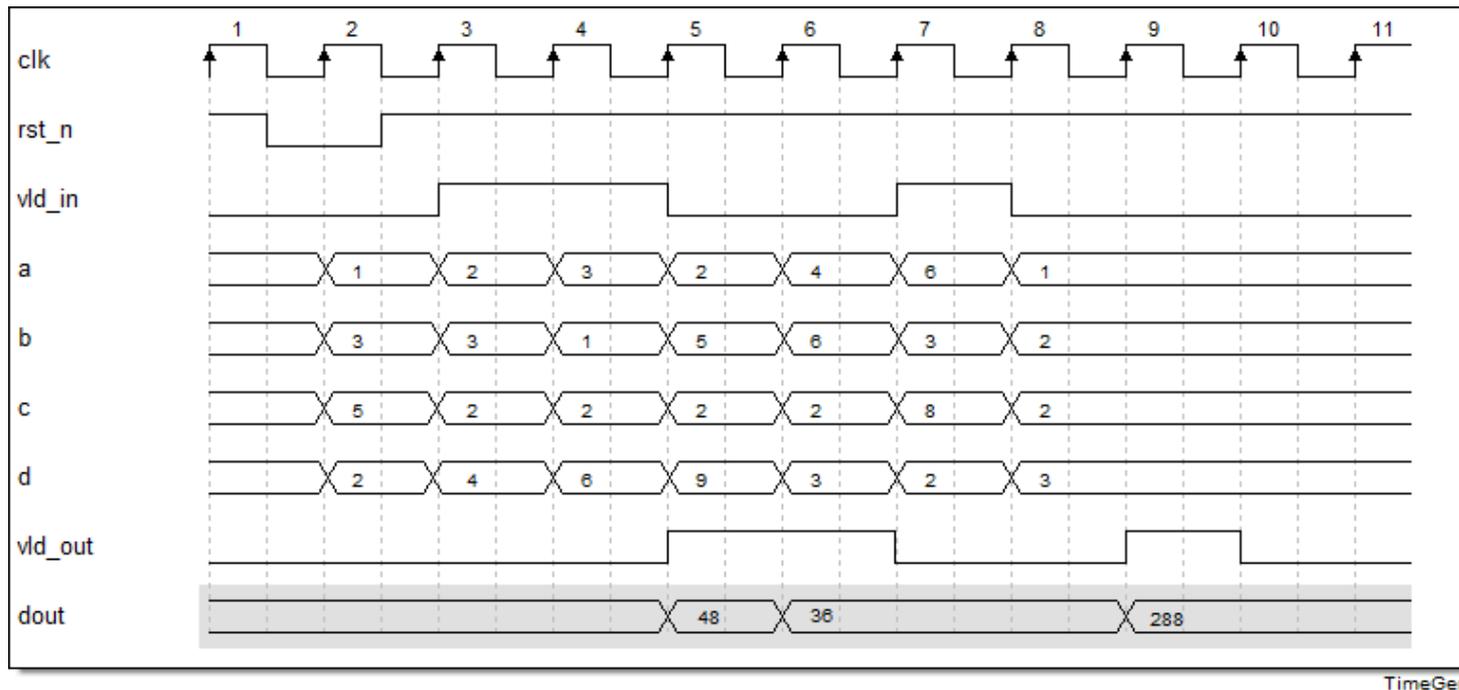
淘 宝： <http://mdy-edu.taobao.com>

课程大纲

1. 功能要求
2. 设计思路
3. 代码设计

一、功能要求

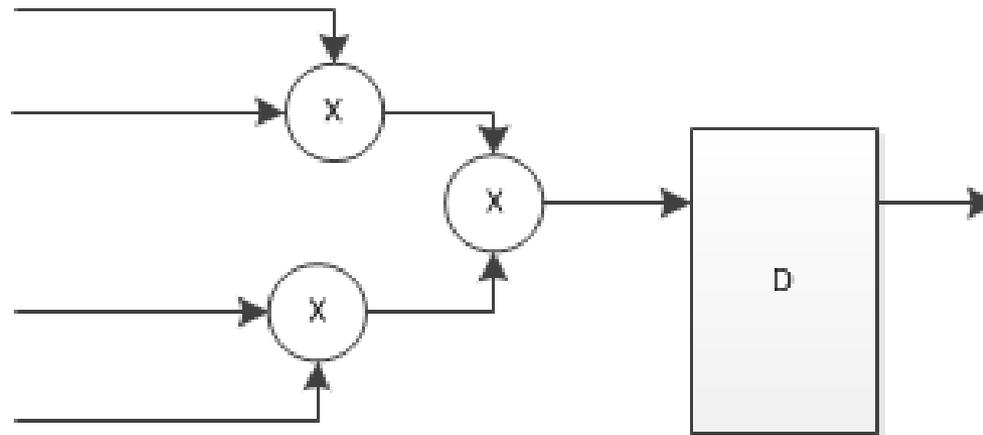
1. 实现一个四输入的乘法器，寄存器之间仅包含一个乘法器（注：2级流水线。每次输入有效（vld_in=1）后的第2个时钟周期输出有效（vld_out=1），并由dout输出结果）



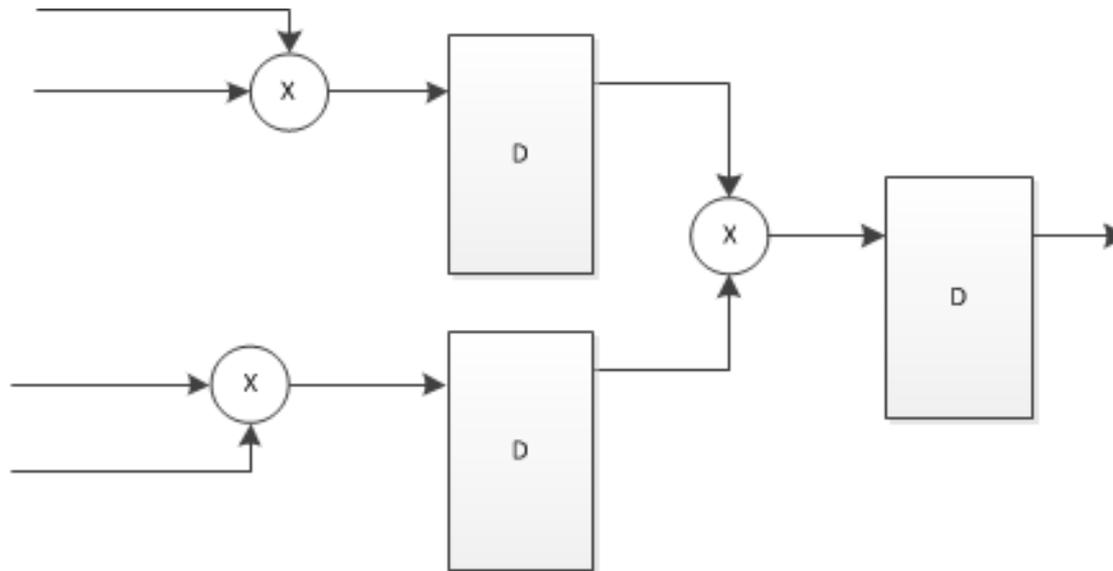
二、设计思路

1. 设计时要从输出信号倒推，现有条件不能直接得到的信号，先假设信号再设计；
2. 一个一个信号设计，逐个击破

三、设计思路



三、设计思路



三、设计思路—vld_out

1. 输入数据后两拍出结果

```
always @(posedge clk or negedge rst_n)begin
    if(rst_n==1'b0)begin
        vld_in_ff0 <= 1'b0;
    end
    else begin
        vld_in_ff0 <= vld_in;
    end
end

always @(posedge clk or negedge rst_n)begin
    if(rst_n==1'b0)begin
        vld_out <= 1'b0;
    end
    else begin
        vld_out <= vld_in_ff0;
    end
end
```

三、设计思路—dout

1. 实现一个四输入的乘法器，实现电路结构 $s=(a*b)*(c*d)$
 2. 一拍出结果
-
1. 可以先得到 $(a*b)$ 、 $(c*d)$ ，然后再将结果相乘
 2. 假设 $s1=a*b$ ， $s2=(c*d)$ ，注意 $s1$ 和 $s2$ 的位宽

```
always @(posedge clk or negedge rst_n)begin
    if(rst_n==1'b0)begin
        dout <= 0;
    end
    else begin
        dout <= s1 * s2;
    end
end
```

三、设计思路—s1和s2

1. $s1=a*b$, $s2=(c*d)$, 注意s1和s2的位宽
2. s1和s2是寄存器输出

```
always @(posedge clk or negedge rst_n)begin
    if(rst_n==1'b0)begin
        s1 <= 0;
    end
    else begin
        s1 <= a*b;
    end
end

always @(posedge clk or negedge rst_n)begin
    if(rst_n==1'b0)begin
        s2 <= 0;
    end
    else begin
        s2 <= c*d;
    end
end
```



明德扬
科技·教育

QQ群： 97925396

官 网： <http://www.mdy-edu.com>

淘 宝： <http://mdy-edu.taobao.com>

Thank You !

