
上海交通大学

SHANGHAI JIAO TONG UNIVERSITY

实验报告



课程名称: 数字逻辑设计

项目名称: Lab1

学院(系): 电子信息与电气工程学院

专 业: 微电子科学与工程

学生姓名: 王赟恺 学号: 522031910274

2024 年 04 月 19 日

目录

1 计数器与 RELU	3
1.1 实验电路功能	3
1.2 实验电路模块端口图	3
1.3 代码分析	3
1.4 测试代码分析	4
1.5 波形图	5
2 实现电路功能	5
2.1 电路原理图	5
2.2 状态转移方程	6
2.3 代码分析	6
2.4 波形分析	7

1 计数器与 ReLU

1.1 实验电路功能

计数器功能：在 `rst_n` 有效时 (该信号为低电平有效)，对计数器中的值进行清零复位，其他情况下每个时钟 (`clk`) 上升沿计数值加 1，当经过设定的阈值 (Threshold) 数的时钟周期后，输入一个信号宽度为一时钟周期的 `out_en` 信号 (高电平有效)。

ReLU 功能：在 `rst_n` 有效时 (该信号为低电平有效)，将输出结果清零；只有当 `out_en` 有效时在时钟上升沿输出 ReLU 计算该周期 `input_data` 作为输入数据的结果，否则将一直保持为上周期的结果。

1.2 实验电路模块端口图

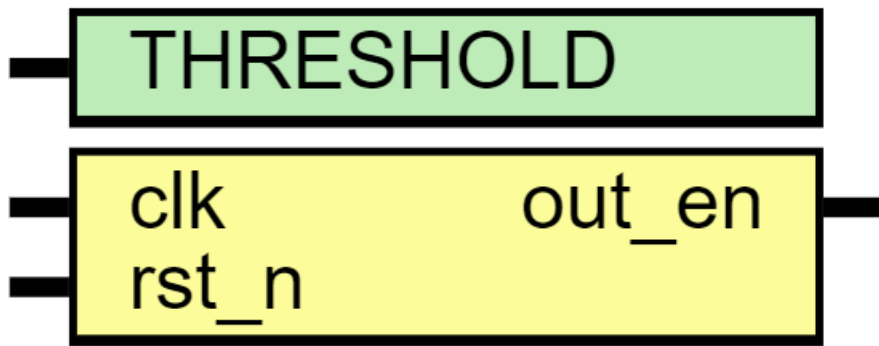


图 1 计数器端口

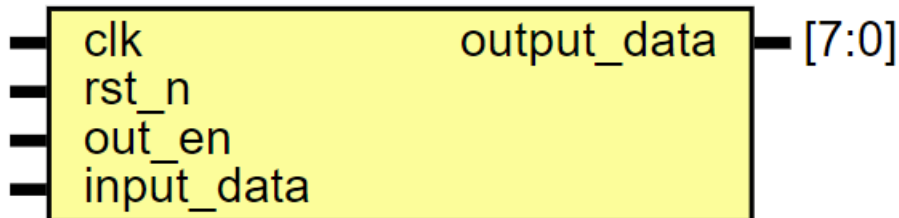


图 2 ReLU 端口

1.3 代码分析

```
localparam LOG_THRESHOLD = $clog2(THRESHOLD);  
  
reg [LOG_THRESHOLD - 1 : 0] counter;
```

使用 `$clog2` 函数，得到 10 进制数的二进制位宽

```

always @(posedge clk or negedge rst_n)
begin
if(~rst_n)
counter = 1'b0;
else
    if(counter < THRESHOLD)
        counter = counter + 1;
    else
        counter = 1'b0; // Reset value when
threshold reached
end

```

计数器时序逻辑部分，rst_n 复位信号低电平有效。

```

always@(*)
if(counter == THRESHOLD)
    out_en = 1'b1;
else
    out_en = 1'b0;

```

计数器组合逻辑部分，counter 从 THRESHOLD-1 更新到 THRESHOLD 的周期，out_en 输出高电平

```

assign cal_data = input_data[7] ? 8'b0 : input_data;

```

ReLU 组合逻辑部分，根据有符号数最高位为 1 为负数，为 0 为正数来判断

1.4 测试代码分析

```

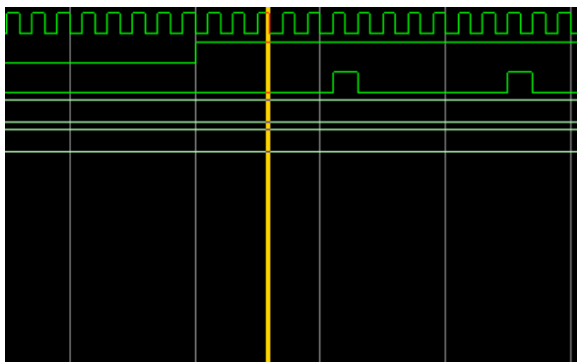
initial begin
clk_reg = 0;
resetsn_reg = 0;
relu_input_data = 8'b10011010;
#100 resetsn_reg = 1;
#2200 relu_input_data = 8'b00101110;

```

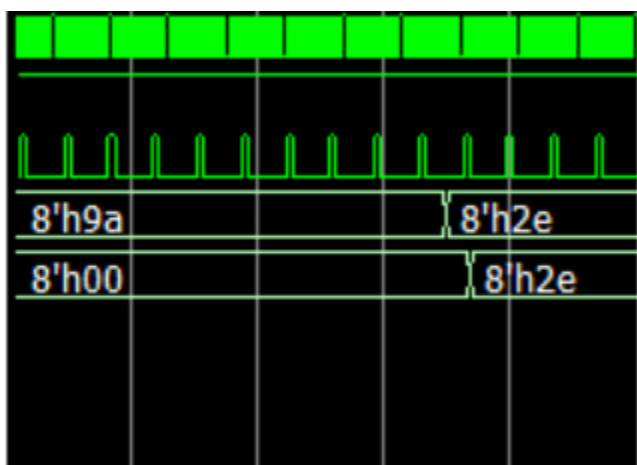
end

relu_input_data 覆盖了正数和负数的情况

1.5 波形图



置位后计数器功能正常，THRESHOLD=6，每 7 个时钟周期后 out_en 变为高电平一个周期

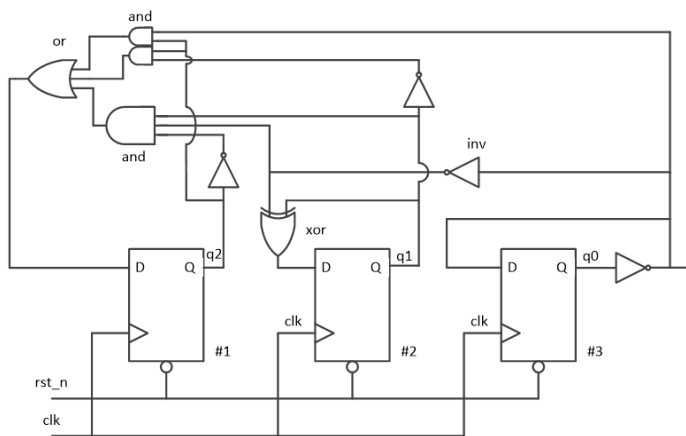


输入 0x9a,输出 0x00，输入 0x2e,输出 0x2e，ReLU 功能正常

2 实现电路功能

2.1 电路原理图

rst_n 有效时，将三个触发器输出 0，D 触发器上升沿触发



2.2 状态转移方程

$$\begin{cases} Q_0(t+1) = \overline{Q_0(t)} \\ Q_1(t+1) = Q_0(t) \oplus Q_1(t) \\ Q_2(t+1) = [Q_0(t)Q_1(t)] \oplus Q_2(t) \end{cases}$$

2.3 代码分析

```
reg q2, q1, q0;
wire q1_inp, q2_inp_or1, q2_inp_or2, q2_inp_or3,
q2_inp;

assign not_q0 = ~q0;
assign q1_inp = q1 ^ q0;
assign q2_inp_or1 = not_q0 & q2;
assign q2_inp_or2 = q2 & (~q1);
assign q2_inp_or3 = (~q2) & q1 & q0;
assign q2_inp = q2_inp_or1 | q2_inp_or2 |
q2_inp_or3;
```

寄存器和线的声明部分

```
always@(posedge clk or negedge rst_n)
begin
```

```

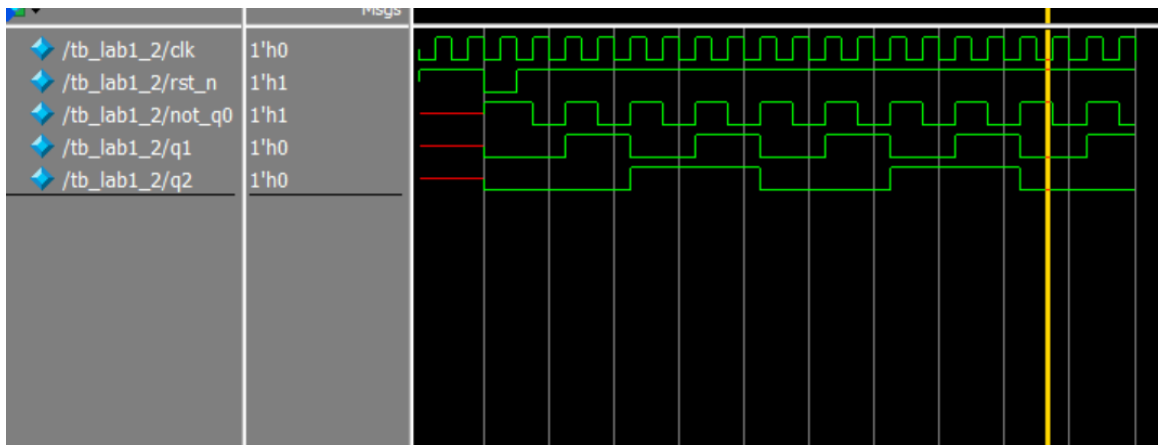
if(!rst_n)
begin
    q2 <= 0;
    q1 <= 0;
    q0 <= 0;
end

else
begin
    q2 <= q2_inp;
    q1 <= q1_inp;
    q0 <= not_q0;
end
end
end

```

寄存器部分的连线

2.4 波形分析



分析波形可知，该电路实现了计数器效果。置位低电平有效后，q0,q1,q2 输出均为 0。