

中国科学技术大学计算机学院
《数字电路实验》报告



实验题目：简单时序逻辑电路

学生姓名：陈鸿绪

学生学号：PB21000224

完成日期：10.20

计算机实验教学中心制

2020 年 09 月

【实验题目】 简单时序逻辑电路

【实验目的】 掌握时序逻辑相关器件的原理、底层结构；

学会用基本逻辑门搭建各类时序逻辑器件；

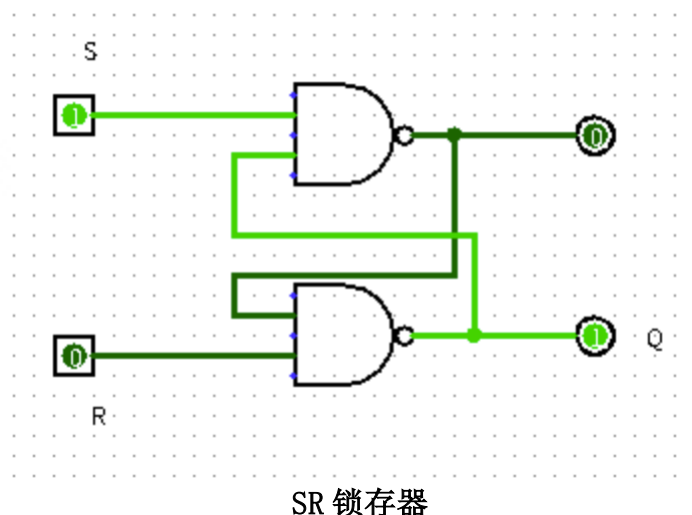
学会用 Verilog HDL 设计、搭建简单逻辑电路；

【实验环境】 在 vlab.ustc.edu.cn 上获取虚拟机，在其已配置好的环境 Logisim 里设计逻辑电路。在 Verilogoj.ustc.edu.cn 网站学习 Verilog 编程知识。

【实验练习】

T1. 在 Logisim 中用与非门搭建 SR 锁存器，画出电路图，并分析其行为特性，列出电路在不同输入时的状态。

[解]：在 Logisim 中用与非门搭建 SR 锁如下图所示：



分析：在图中 $S=0$ ， $R=1$ 时， Q 置 0； $S=1$ ， $R=0$ 时， Q 置 1； $S=R=1$ 时，可以发现 Q 仍然等于上一个状态时的值。若 $S=R=0$ 时， Q 和另

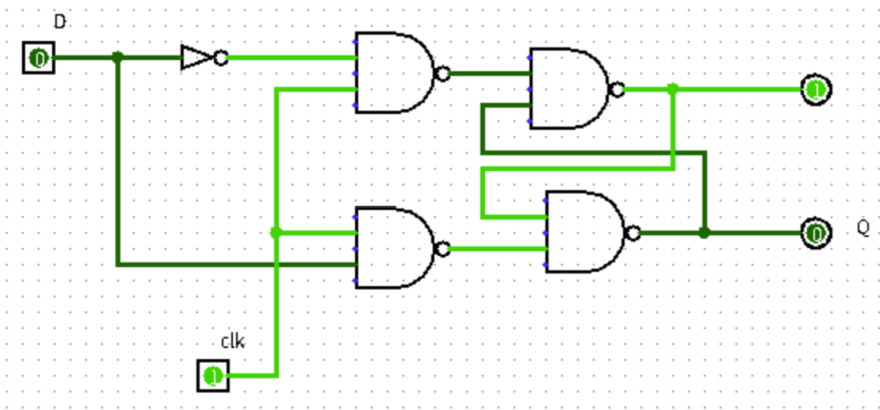
一个输出端都为 1，这导致若下一步 S、R 都取 1，即保持状态时，锁存器会变得不稳定，所以 $S=R=0$ 是非法状态。

综上，如下表所示：

S	R	Q	功能
0	1	0	置 0
1	0	1	置 1
1	1	不变	保持
0	0	1	非定义状态

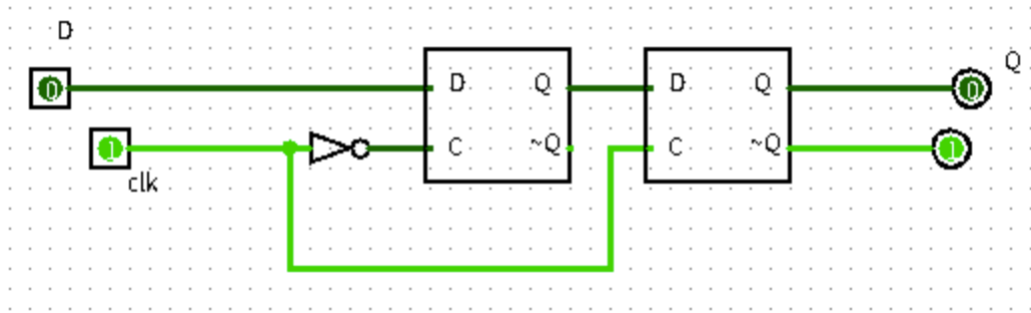
T2. 在 Logisim 中搭建一个支持同步置位功能的 D 触发器，画出其电路图，并编写对应的 Verilog 代码。

[解]：首先搭建 D 锁存器，如下图所示：



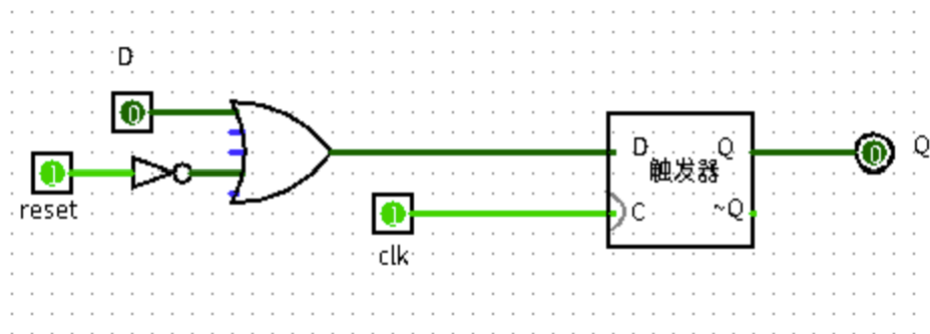
D 锁存器

封装 D 锁存器完成后搭建主从 D 触发器：



主从 D 触发器

封装 D 触发器后, 我们用其搭建同步**置位**D 触发器(reset 在 0 有效):



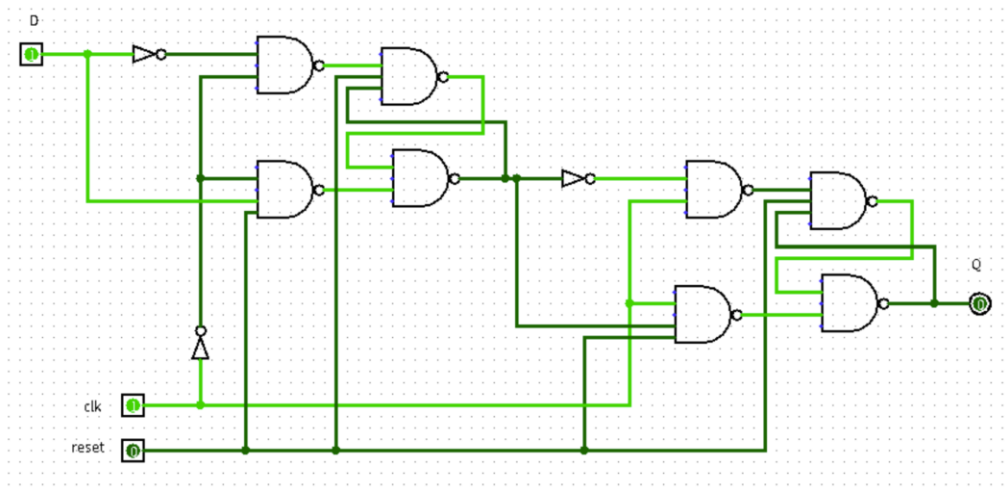
同步置位 D 触发器

根据同步置位 D 触发器的功能, 我们写出其相应的 Verilog 代码:

```
module d_ff_r(
    input clk, reset, d,
    output reg q);
    always@(posedge clk)
    begin
        if(reset==0)
            q <= 1'b1;
        else
            q <= d;
        end
    endmodule
```

T3. 在 Logisim 中搭建一个带有异步复位功能的 D 触发器，画出其完整电路图，并进一步调用该触发器设计一个从 0~15 循环计数的 4bit 计数器（可使用 Logisim 中的加法器模块，也可自行设计计数器），写出计数器的 Verilog 代码。

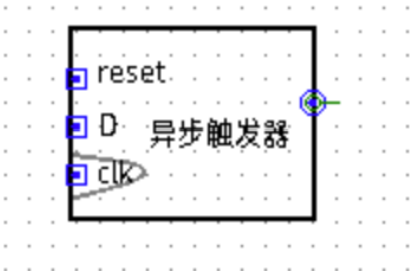
[解]：首先搭建一个带有异步复位功能的 D 触发器，如下图所示：



异步复位功能的 D 触发器

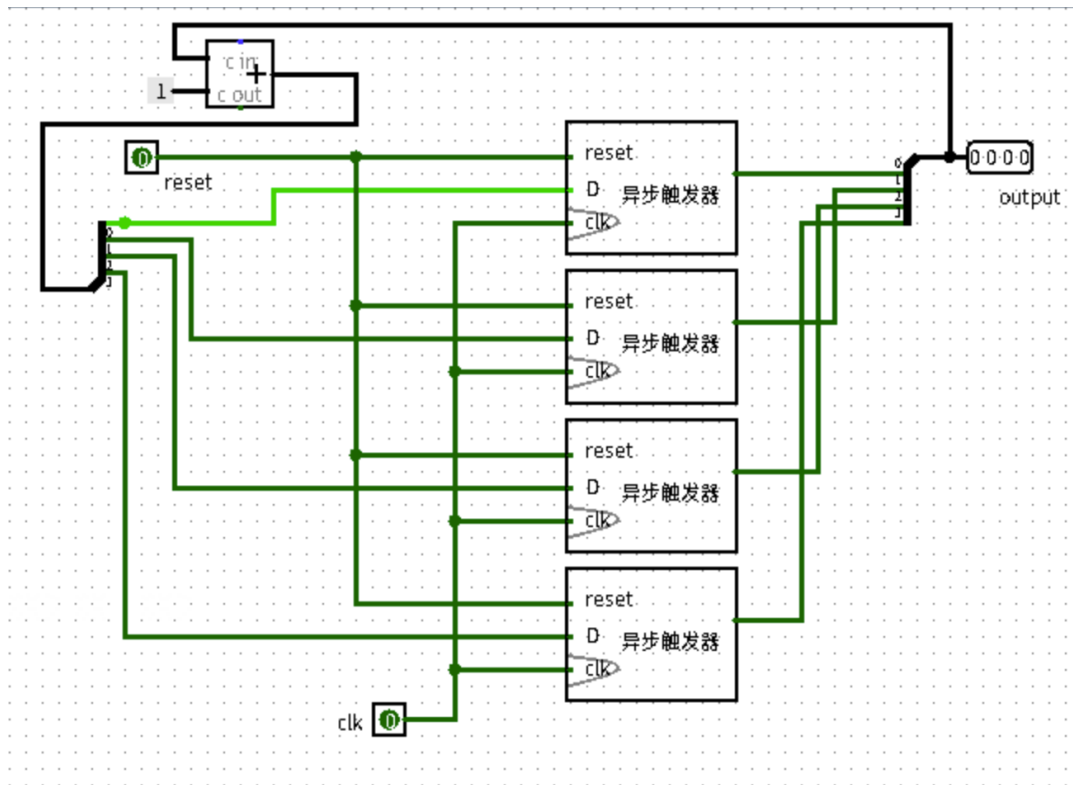
当不管时钟 clk 处于什么状态，reset 一旦取 0 时，Q 就会恒等于 0；reset 取 1 时，以上电路即为一个基本的主从 D 触发器；reset 从 0 到 1 时，由于电路特性，在时钟未达到上沿时，输出 Q 仍为 0，只有在始终过了一个上沿时后，Q 才会更新。这是满足异步复位的要求的。

封装好电路，如下图所示：



封装样式

首先可以用四个异步复位 D 触发器构造一个异步复位的寄存器，再由内置的加法器原件构造出 4bit 计数器，如下图所示。



4bit 计数器

该计数器在 0~15 之间循环计数，复位时立刻输出值为 0，reset 由 0 置 1 时，计数器会从 0 重新开始计数。

该计数器的 Verilog 代码如下所示：

```
module count_4bit(  
    input clk, reset,  
    output reg cnt);  
    always@(posedge clk or negedge reset)  
    begin  
        if(reset == 0)  
            cnt <= 4'b0;
```

```

else

    cnt <= cnt + 1;

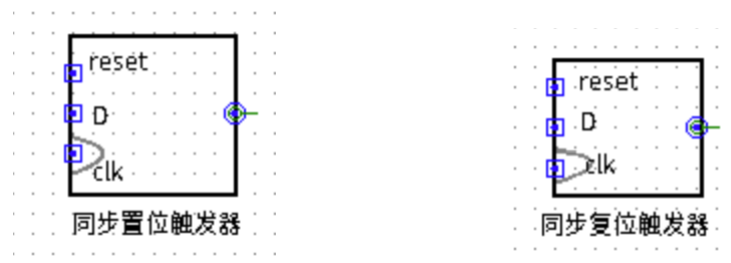
end

endmodule

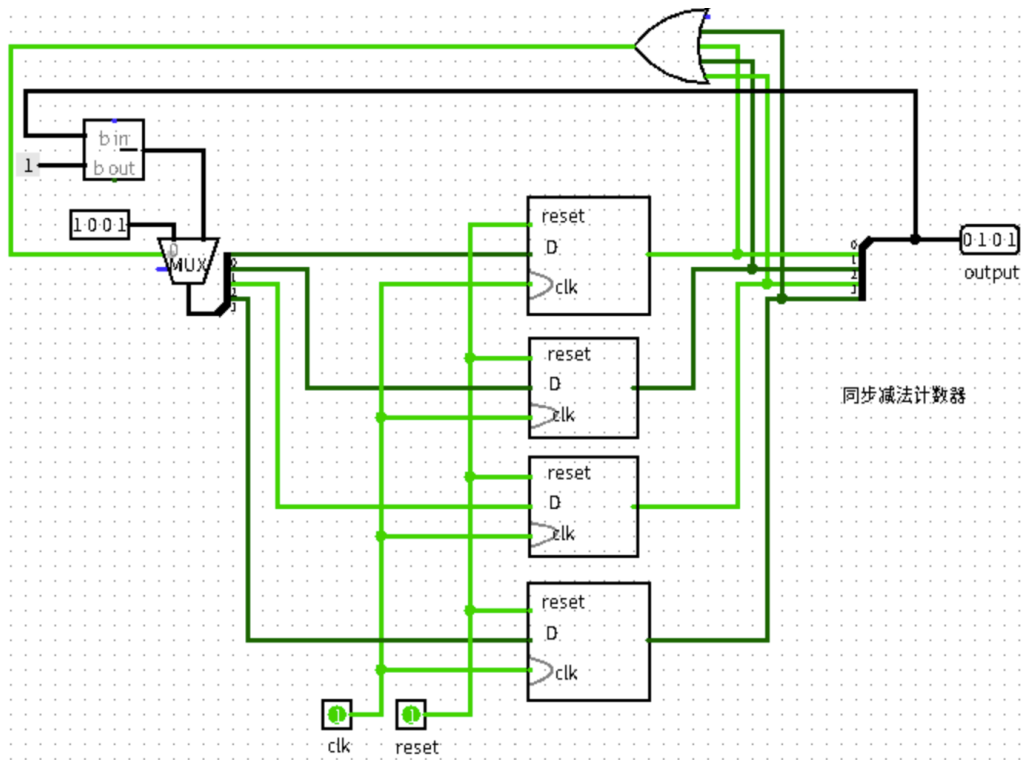
```

T4. 在 Logisim 中搭建一个 9~0 循环递减的计数器，复位值为 9，每个周期减一（可使用 Logisim 中的减法器模块，也可自行设计计数器），画出电路图，进行正确性测试，并写出其对应的 Verilog 代码。

[解]：由于题目中并没有明确要求，所以我们这里搭建同步复位的计数器。由于 9 的二进制表示为 1001，计数器整体复位时需要两个复位到 0，两个置位到 1，所以我们需要同步置位 D 触发器与同步复位 D 触发器，封装样式如下：（大为同步置位触发器，小为同步复位触发器）



通过以上两种触发器构造寄存器后，再利用内置减法单元，我们可以构造出减法计数器，但是该减法计数器尚且不能完成 0~9 的循环计数，在状态为 0000 时刻，若再减去 1，会使状态变成 1111，不符合功能，所以我们用一个 4bit 的选择器，让状态为 0000 时候，选择下一个状态到 1001，即为 9，由此就可以完成该减法计数器的循环计数的功能。下面是该计数器的逻辑电路：



9~0 循环递减的计数器

由该计数器电路，在 reset 取 0 时，可以发现计数器复位到 1001，即为 9，是符合该计数器的复位要求的。

以下为该计数器的 Verilog 代码：

```
module count_10(
    input clk,reset,
    output reg cnt);
always@(posedge clk)
begin
    if(reset == 0)
        cnt <= 4'b1001;
    else
        if(cnt == 4'b0)

```



```

        cnt == 4'b1001;

    else

        cnt <= cnt - 1;

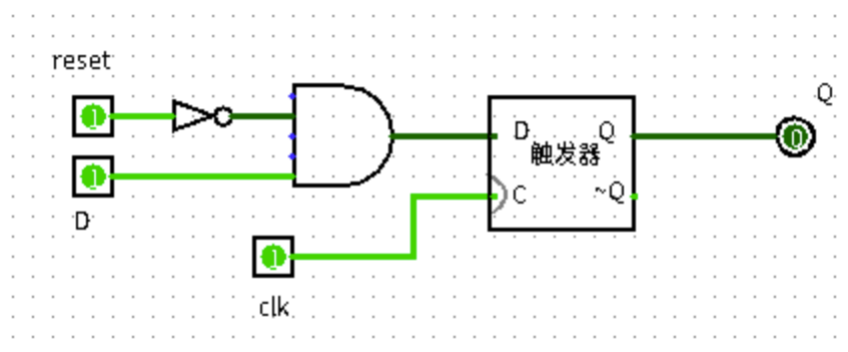
    end

endmodule

```

T5. 手册中给出的示例电路的复位信号都是低电平有效，如要使复位信号高电平有效，应如何实现？试用 Logisim 画出一个示例电路，并编写 Verilog 代码。

[解]:在Logisim构造复位信号高电平有效的D触发器,如下图所示:



复位信号高电平有效同步复位 D 触发器

在 reset 置 1 的时候，由上图可以分析输出 Q 始终为 0，实现了复位信号高电平有效。

Verilog 代码如下：

```

module d_ff_r_h(
    input d,clk,reset,
    output q);

    always@(posedge clk)

```

```
begin
    if(reset == 1)
        q <= 1'b0;
    else
        q <= d;
    end
endmodule
```

【总结与思考】

总结：在本次实验中，我学会了用 Logisim 构造简单的时序电路，熟悉了如何构造 SR 锁存器、D 触发器、计数器、寄存器等逻辑电路。同时我也掌握时序逻辑相关器件的原理、底层结构；