**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：\_ \_李远航\_\_ \_\_\_

学生学号： \_PB20000137\_ \_

完成日期： 2021.11.03\_ \_

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及底层结构
* 能够用基本逻辑门搭建各类时序逻辑器件
* 区分组合逻辑电路与时序逻辑电路
* 能够理解同步置位、异步置位、同步复位、异步复位之间的区别
* 能够使用Verilog HDL设计简单逻辑电路
* 进一步了解Verilog相关语法

【实验环境】

* PC一台
* Windows 10 操作系统
* Java 8（64 bit）
* Logisim 仿真工具

【实验过程】

1. 搭建双稳态电路，搭建如下图所示的电路：

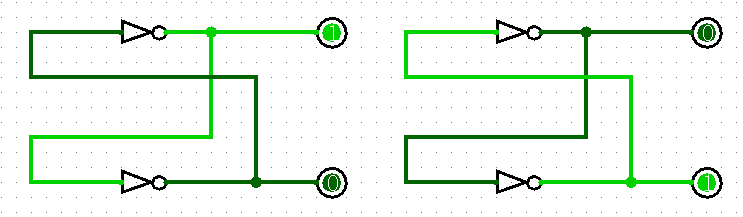


图1.1

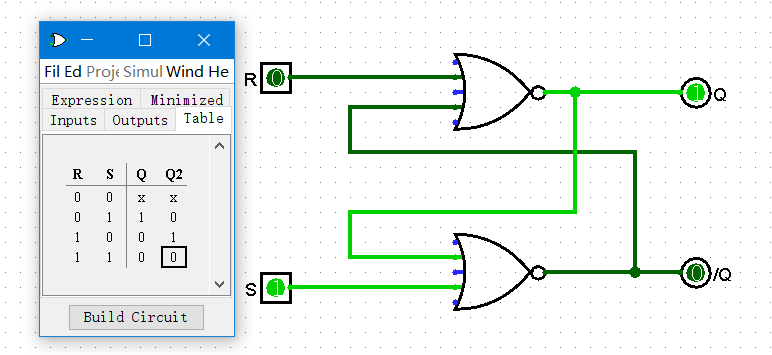
1. 搭建SR锁存器，观察电路特性

图2.1

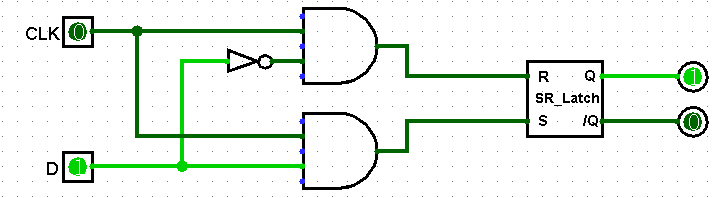
1. 搭建D锁存器

图3.1

观察发现当CLK为高电平时,Q信号会随D的变化而变化,CLK为低电平时，Q保持之前的值不变

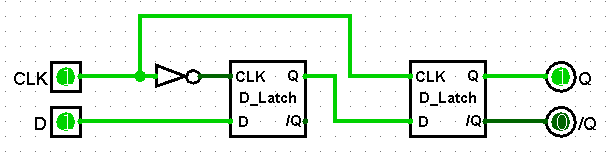
1. 搭建D触发器

图4.1

**观察电路特性发现，在每一个CLK的上升沿，Q信号会变为此时D的值，其他时刻，Q的值保持不变，对应的Verilog代码如下：**

module d\_ff(input clk, d,

            output reg q);

    always@(posedge clk)

        q <= d;

endmodule

**搭建同步复位的D触发器及其Verilog代码**

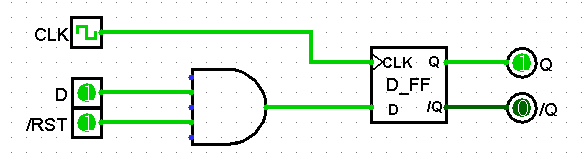


图4.2

module d\_\_ff\_r (input clk, rst\_n, d,

                output reg q);

    always @(posedge clk)

    begin

        if (rst\_n == 0)

            q <= 1'b0;

        else

            q <= d;

    end

endmodule

**异步复位的D触发器Verilog代码：**

module d\_ff\_r(input clk, rst\_n, d,

              output reg q );

    always@(posedge clk or negedge rst\_n)

    begin

        if (rst\_n == 0)

            q <= 1’b0;

        else

            q <= d;

    end

endmodule

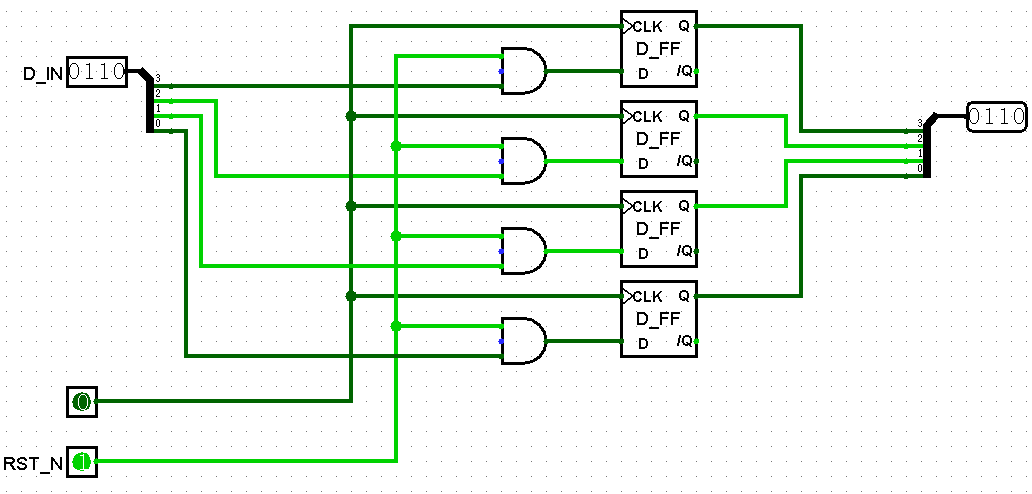
1. 搭建寄存器

图5.1

该寄存器能够存储4bit的数据，在时钟上升沿，复位信号为高电平时，将输入值存储到输出端，其Verilog代码如下：

module REG4 (input CLK, RST\_N,

             input [3:0] D\_IN,

             output reg [3:0] D\_OUT);

    always @(posedge CLK)

    begin

        if (RST\_N == 0)

            D\_OUT <= 4'b0;

        else

            D\_OUT <= D\_IN;

    end

endmodule

1. 搭建简单时序逻辑电路

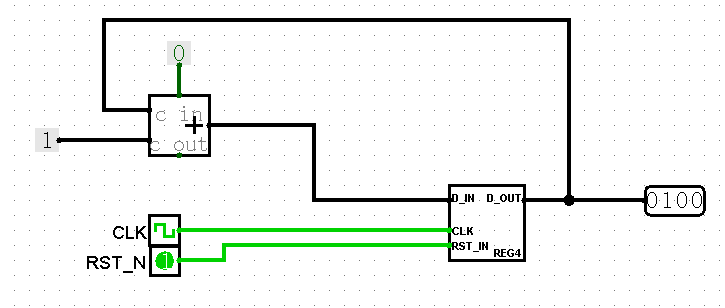
在Logism中利用寄存器搭建如下所示的循环计数器：

图6.1

**当复位信号高电平，随着CLK 的变化，输出端从0~15再变回0**

**Verilog代码如下：**

module REG4 (input CLK, RST\_N,

             output reg [3:0] CNT);

    always @(posedge CLK) begin

        if (RST\_N == 0)

            CNT <= 4'b0;

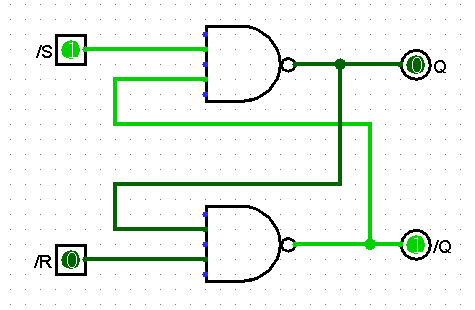
        else

            CNT <= CNT+4'b1;

    end

endmodule

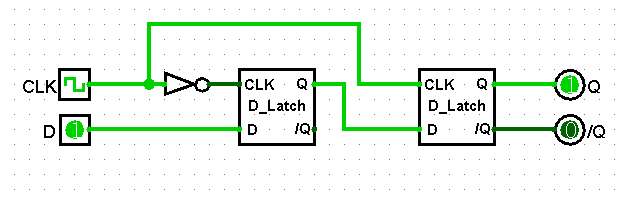
【实验练习】

1.  使用与非门搭建SR锁存器

题图1.1

|  |  |  |  |
| --- | --- | --- | --- |
| S | R | Q | ~Q |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | X | X |

表1：电路特性

1.  使用D锁存器搭建出如图所示的D触发器，以及对应的Verilog代码

题图2.1

module D\_FF (input CLK, D,

             output reg Q);

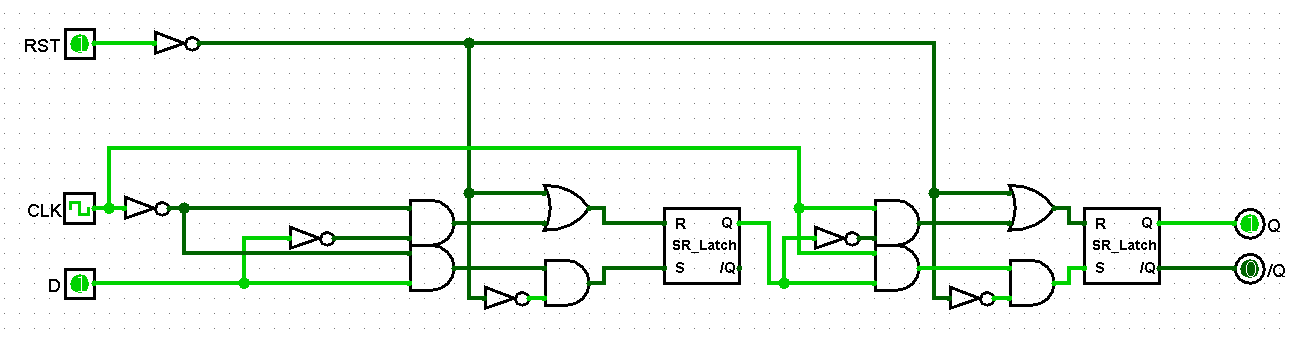
    always @(posedge CLK)

    begin

        Q <= D;

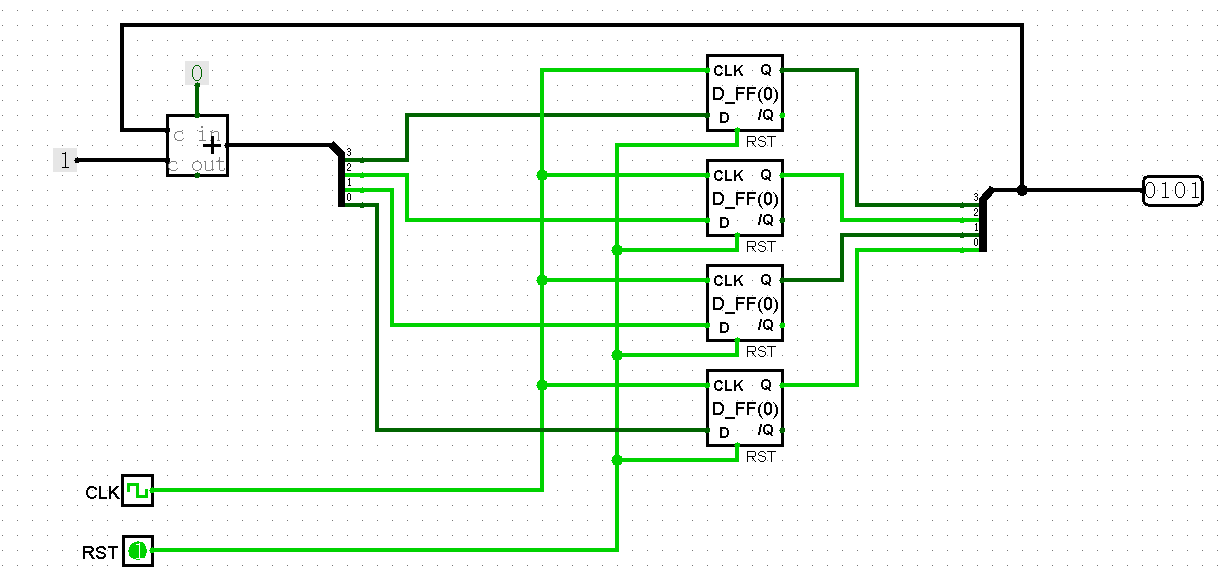
    end

endmodule

1. 异步复位的D触发器如图所示，使用SR锁存器，RST从1变成0时，Q置0

题图3.1

**0~15循环计数的4bit计数器如图所示，RST从1变0时，输出端置0：**



题图3.2

**对应循环计数器的Verilog代码：**

module fun (input CLK, RST,

             output reg [3:0] OUT);

    always @(posedge CLK or negedge RST)

    begin

        if (RST == 0)

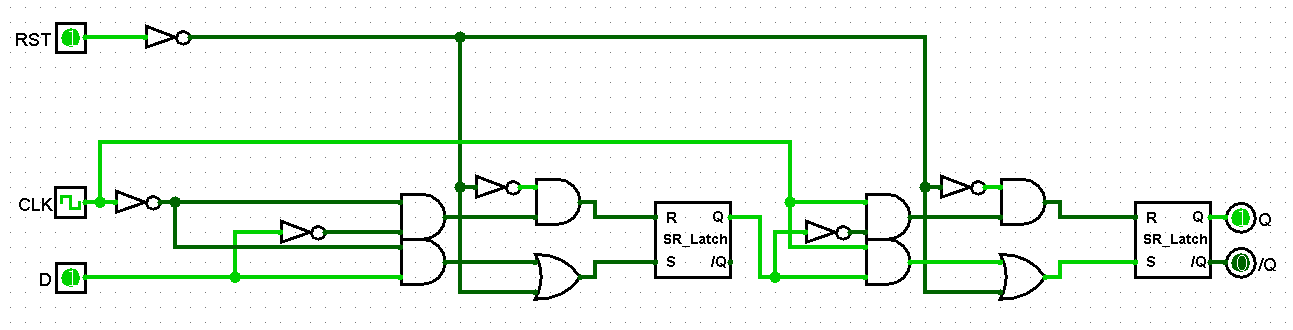
            OUT <= 4'b0;

        else

            OUT <= OUT+4'b1;

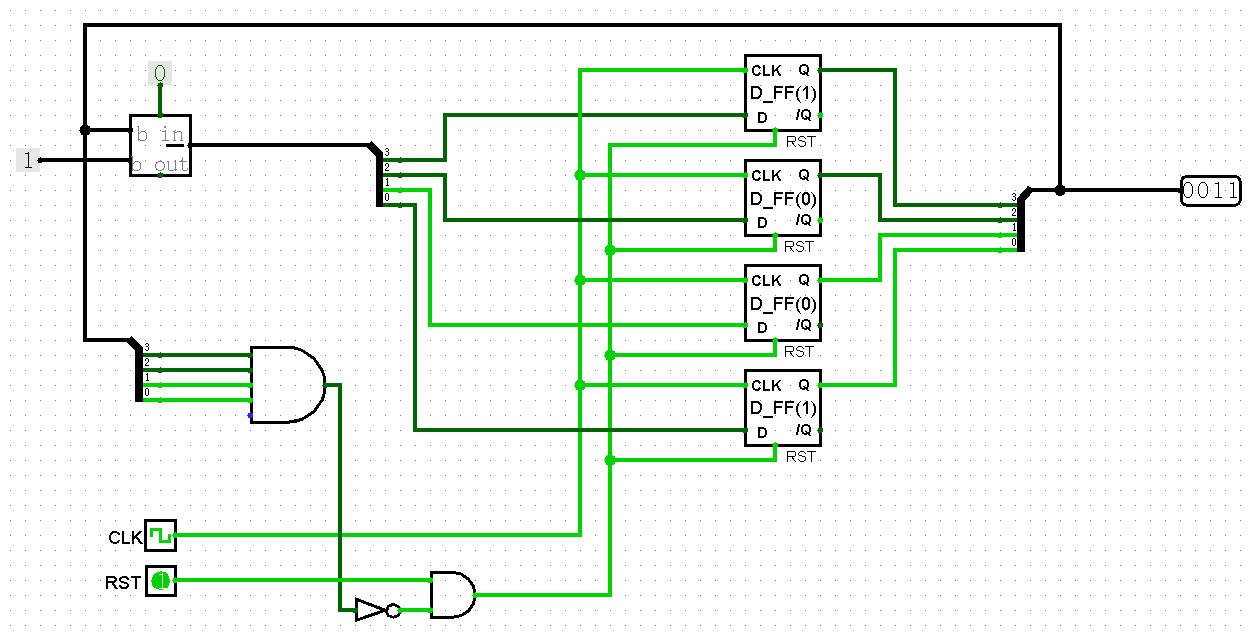
    end

endmodule

1.  循环递减的计数器需要用到异步复位为1的D触发器，如图所示：

题图4.1

**如图所示对于不同位选用复位值不同的触发器，以此达到复位9的目的，同时在检测到输出值变成4’b1111时同时触发复位信号，可以得到符合题意的电路，虽然输出端经历短暂的4’b1111的状态，但是属于复位为9的时钟脉冲：**

**如图所示**

题图4.2

**Verilog代码如下：**

module fun (input CLK, RST,

             output reg [3:0] CNT);

    always @(posedge CLK or negedge RST)

    begin

        if (RST == 0 || CNT == 4'b0)

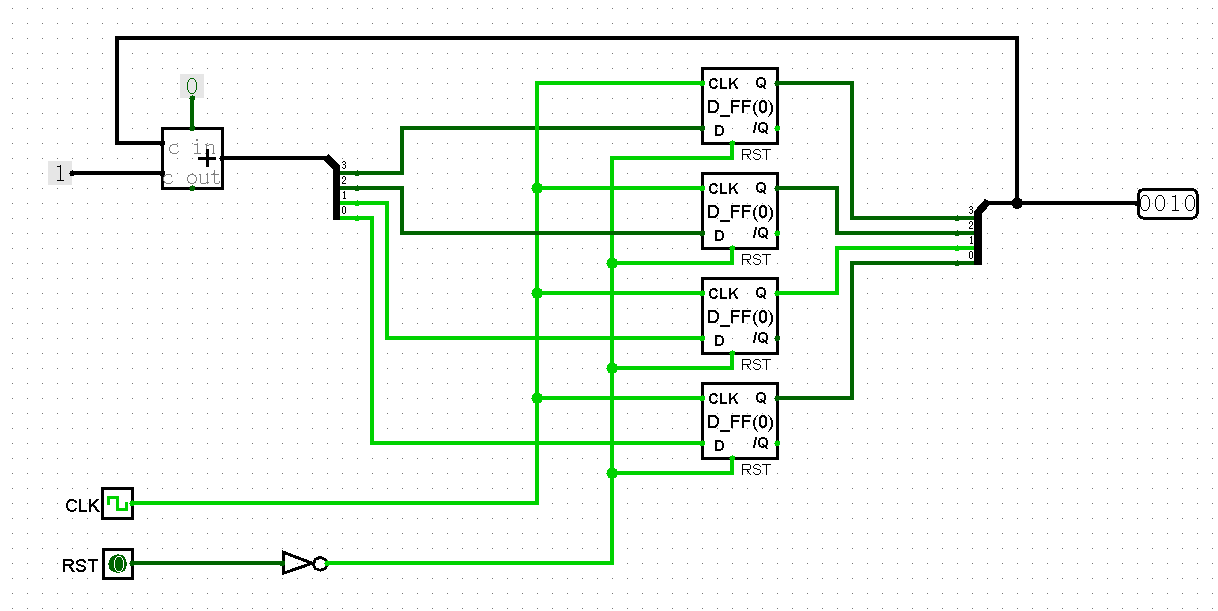
            CNT <= 4'b1001;

        else

            CNT <= CNT-4'b1;

    end

endmodule

1. 在复位端加上一个非门就可以使得复位信息高电平有效，选用题目3的电路和Verilog代码进行修改：

题图5.1

module fun (input CLK, RST,

             output reg [3:0] OUT);

    always @(posedge CLK or posedge RST)

    begin

        if (RST == 1)

            OUT <= 4'b0;

        else

            OUT <= OUT+4'b1;

    end

endmodule

【总结与思考】

* 学习了时序逻辑相关器件的原理及底层结构
* 学会使用基本逻辑门搭建各类时序逻辑器件
* 学会使用Verilog HDL设计简单逻辑电路
* 更加深刻认识到组合逻辑电路与时序逻辑电路的区别
* 本次实验较之前几次实验任务量稍大，难度有提高