

浙江大学

本科实验报告

课程名称:	数字逻辑设计
姓 名:	
学 院:	计算机科学与技术学院
专 业:	计算机科学与技术
邮 箱:	
QQ 号:	
电 话:	
指导教师:	蔡铭
报告日期:	2023 年 12 月 16 日

浙江大学实验报告

课程名称：_____ 数字逻辑设计 _____ 实验类型：_____ 综合 _____

实验项目名称：_____ 锁存器与触发器基本原理 _____

学生姓名：_____ 学号：_____ 同组学生姓名：_____

实验地点：_____ 紫金港东四 509 室 _____ 实验日期：_____ 2023 _____ 年 _____ 11 _____ 月 _____ 29 _____ 日

一、操作方法和实验步骤

- 本次实验开始，我们将开始时序电路的设计。而时序电路中最基本的实现就有锁存器和触发器。
- **锁存器 (Latch)**以及 **触发器 (Flip-flop)**被称为双稳态多谐振荡器(Bistable multivibrator)，即拥有两个稳定状态并可以用来记录和表达两种状态信息的电子部件，可以用来存储 **1 比特**的数据（两种状态分别为 0 和 1）。

锁存器与触发器通常分为几种类型：**SR**("set-reset") **D**("data" or "delay") **T**("toggle") **JK** 等。

- 构成锁存器有以下条件：

能长期保持给定的某个稳定状态

有两个稳定状态：0、1

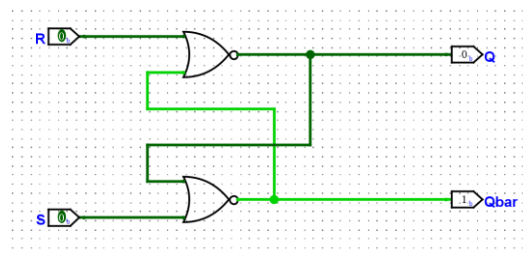
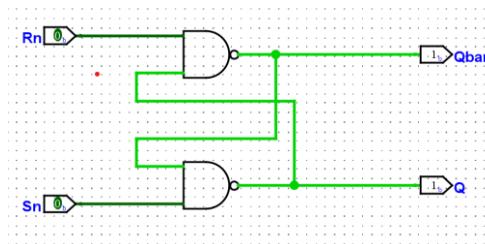
在一定的条件下能随时改变逻辑状态，即置 1 或 0.

SR 锁存器

- SR 锁存器通过两个输入端口(S, R)对锁存器保存数据进行修改或保持，S 含义为 set 即将数据设置为 1，R 含义为 reset 即将数据重置为 0。

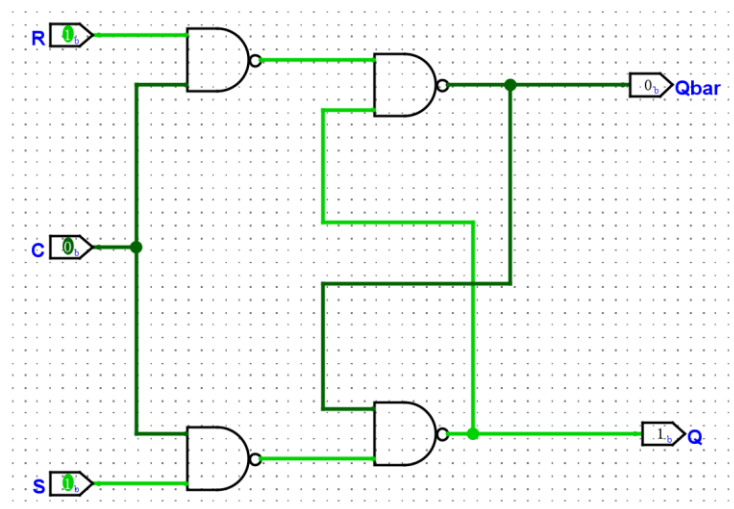
$R\ S$	$Q\ Qn$	说明
0 0	$Q\ Qn$	保持
0 1	1 0	置1
1 0	0 1	置0
1 1	0 0	未定义

- 在具体的实现上，根据使用逻辑门（输入端口有效电平）的不同分为 **SR NOR latch** 和 **\overline{SR} NAND latch**。后者名字中的“非”表示其值为假时（低电平）有效。



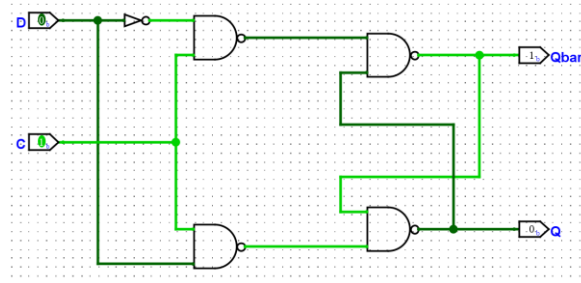
门控 SR 锁存器

- 为 SR 锁存器添加一个输入 **C**，只有它的值为 1 时才能对锁存器存储的值进行修改。



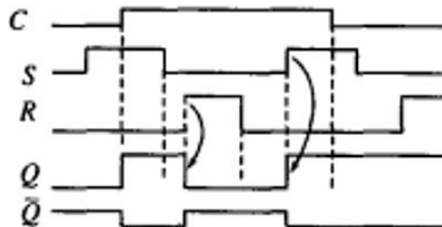
门控 D 锁存器

- 可以观察到，SR 锁存器存在**不稳定状态**这一问题，不稳定状态与其输入相关，而如果我们只用一个输入端口 D 来对存储内容进行修改，就可以避免不稳定状态。同时，我们需要加入一个门控，来实现保持状态。



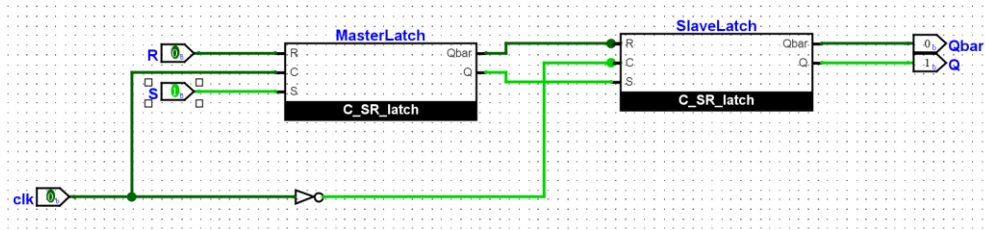
触发器

- 对于门控锁存器，当**门控有效**时，改变输入就会造成锁存器保存值的变化，**多次**改变输入就会导致锁存器输出**多次变化**。这种在一个有效周期（或时钟脉冲）可以多次改变状态的现象，叫做**空翻现象**（或称竞态现象）。为解决锁存器造成的空翻现象，提出了触发器。根据触发器的设计可以分为**主从触发器**(Master-slave flip-flop)和**边沿触发器**(Edge-triggered flip-flop)。



SR 主从触发器

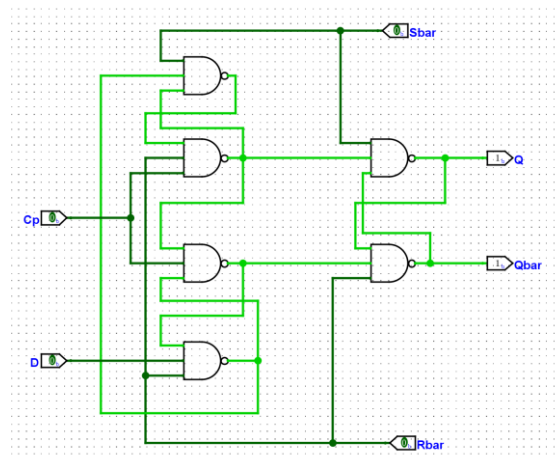
- 由两个门控 **SR Latch** 串联构成，第二个锁存器的时钟通过非门取反。
- clk 处于低位时，master 门控有效，slave 门控无效
 - 对 S R 进行修改，**master** 的输出会**改变**，但 slave 的输出（也就是触发器的输出）不会改变
- clk 上升沿后一小段时间内，master 门控变为无效，slave 门控有效
 - master 输出不会再改变，**slave** 接收到 master 的输出并**改变或维持自身输出**
- clk 处于高位时，master 门控无效，slave 门控有效
 - 对 S R 进行修改，**master** 的输出**不会改变**，因此 slave 的输出也不会改变



- **一次性采样问题**，主从触发器在主锁存器门控有效时，可以任意改变自身状态，因此我们需要将输入信号维持至少半个时钟周期（即时钟低电平时），如果有信号“毛刺”会导致主锁存器的状态有不期望的变化，因此产生了“一次性采样”问题。

边沿触发器

- 该触发器包括两个阶段：输入处理和输出处理。通过处理时钟信号 `clk` 和数据信号 `D` 来提供给输出处理部分锁存器正确的输入。



二、实验结果和分析

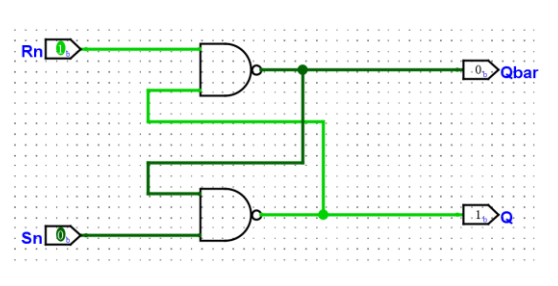
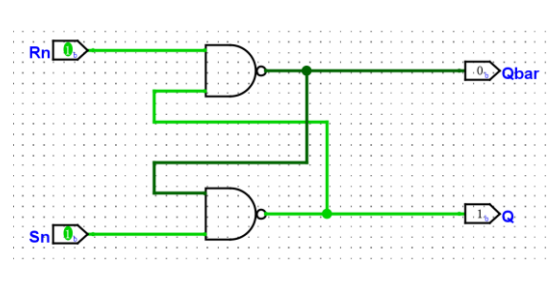
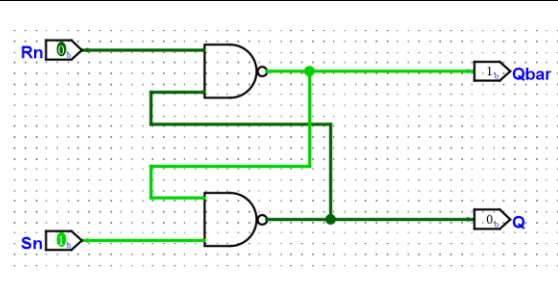
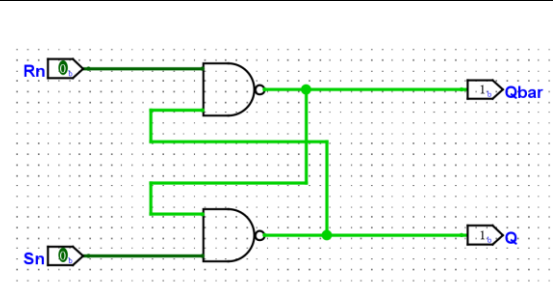
基本 SR 锁存器

- 值得注意的是，寄存器和锁存器都可以用简单的 `verilog` 代码实现，下面是 SR 锁存器的基本实现。

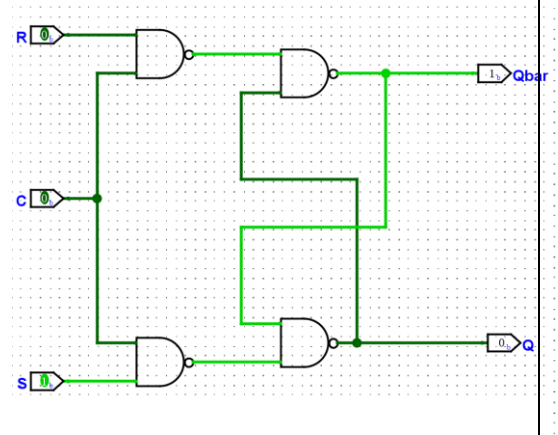
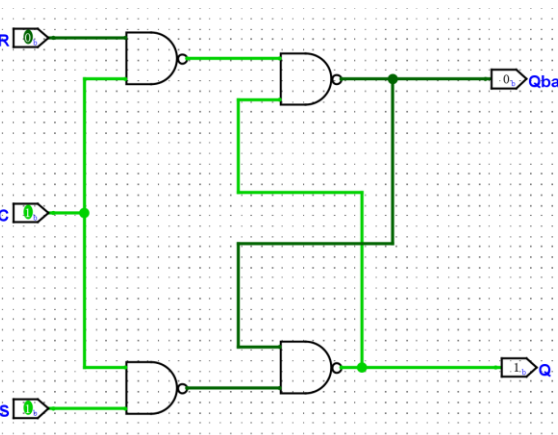
```

module SR (
    input wire S, R,
    output wire Q, InvQ
);
    nand m0 (Q, S, InvQ);
    nand m1 (InvQ, Q, R);
endmodule

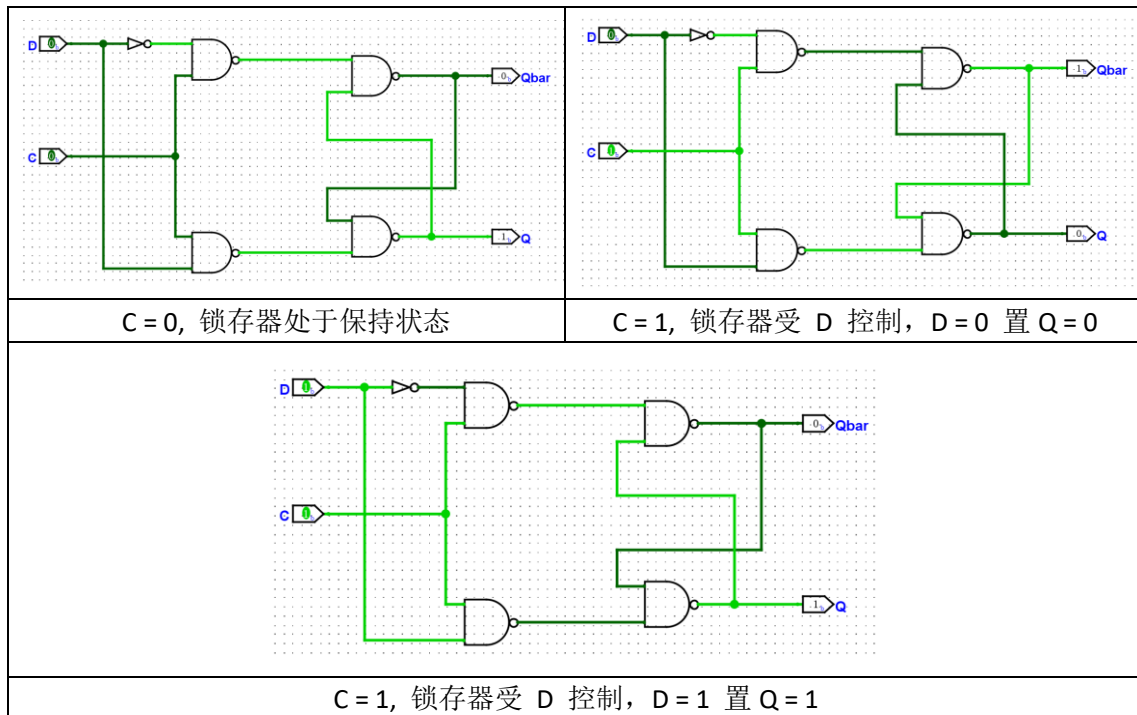
```

	
$S_n = 0, R_n = 1$, 将 Q 置 1	$S_n = 0, R_n = 1$, 保持
	
$S_n = 1, R_n = 0$, 将 Q 置 0	$S_n = 0, R_n = 0$, 不稳定 (例如出现 1 1)

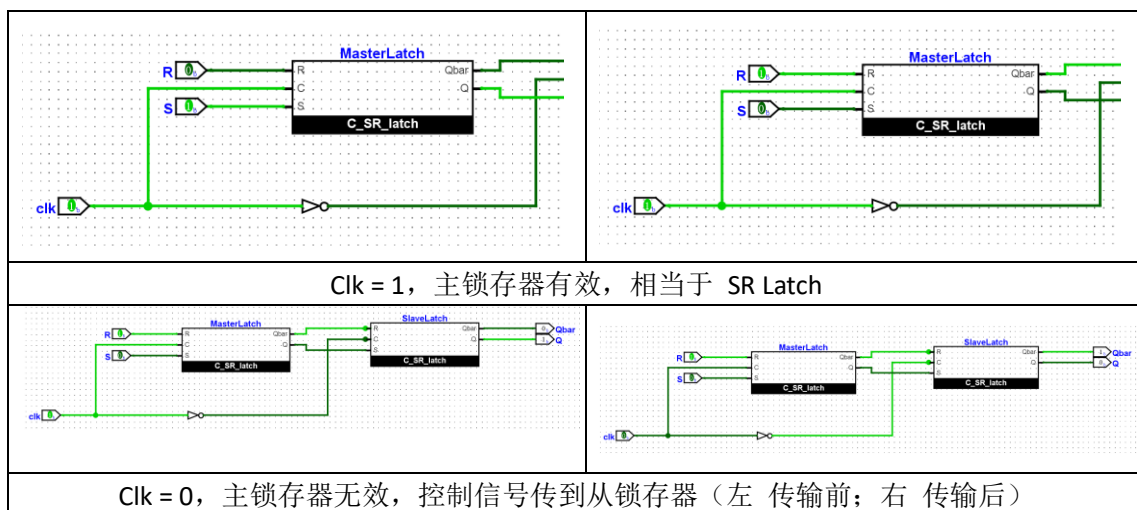
门控 SR 锁存器

	
$C = 0$, 此时锁存器始终处于保持状态	$C = 0$, 此时即为正常的 SR 锁存器

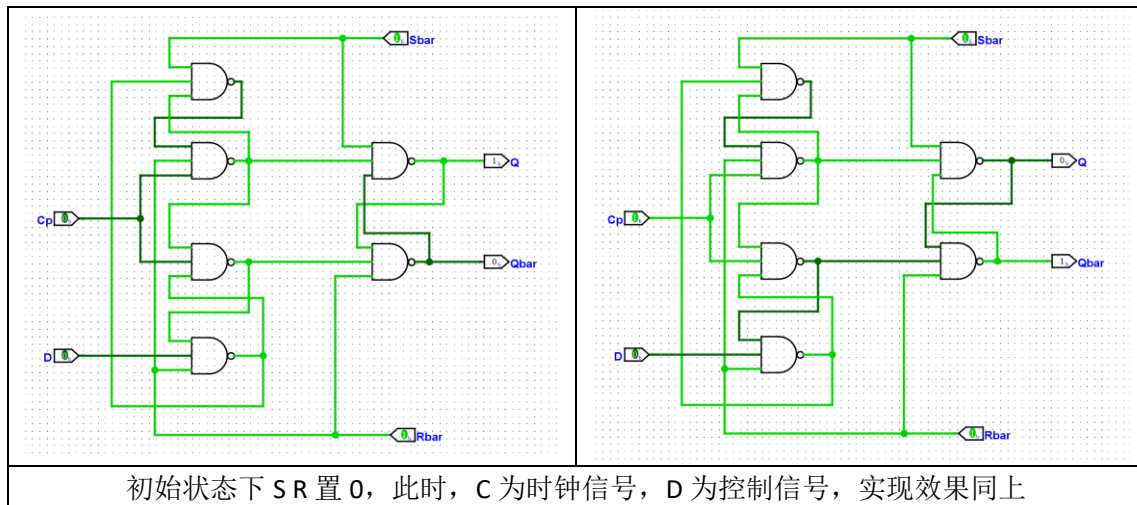
D 锁存器



SR 主从触发器



D 触发器



三、讨论、心得

本次实验并不复杂，主要是熟悉几种基本的时序电路的电路实现，并对它们的真值表、功能、优缺点有一定了解。我在画电路图和思考功能实现的过程中对时序电路也有了更加多的认识。