

# 浙江大学

## 本科实验报告

|       |            |
|-------|------------|
| 课程名称: | 计算机逻辑设计基础  |
| 姓名:   | 张晋恺        |
| 学院:   | 竺可桢学院      |
| 系:    | 所在系        |
| 专业:   | 计算机科学与技术   |
| 学号:   | 3230102400 |
| 指导教师: | 董亚波        |

2024 年 5 月 9 日

# 浙江大学实验报告

课程名称: 计算机逻辑设计基础 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: 张晋恺 专业: 计算机科学与技术 学号: 3230102400

同组学生姓名: 杨吉祥 指导老师: 董亚波

实验地点: 东 4-511 实验日期: 2024 年 5 月 8 日

## 一、实验目的和要求

- 掌握锁存器与触发器构成的条件和工作原理
- 掌握锁存器与触发器的区别
- 掌握基本 SR 锁存器, 门控 SR 锁存器, D 锁存器, SR 触发器, D 触发器的基本功能
- 掌握基本 SR 锁存器, 门控 SR 锁存器, D 锁存器, SR 触发器存在的时序问题

## 二、实验内容和原理

### 实验任务

- 实现基本  $\bar{S}\bar{R}$  锁存器, 验证功能和存在的时序问题
- 实现门控 SR 锁存器, 并验证功能和存在的时序问题
- 实现 D 锁存器, 并验证功能和存在的时序问题
- 实现 SR 主从触发器, 并验证功能和存在的时序问题
- 实现 D 触发器, 并验证功能

### 实验原理

#### 构成锁存器的充分条件

- 能长期保持给定的某个稳定状态
- 有两个稳定状态:  $Q = 0$  和  $Q = 1$
- 能在一定条件下随时改变逻辑状态, 即置 1 或置 0

最基本的锁存器有 SR 锁存器和 D 锁存器, 锁存器有两个稳定状态, 又称双稳态电路

#### SR 锁存器

将两个具有 2 输入端的反向逻辑器件的输出与输入端交叉连起来, 另一个输入端作为外部信息输出端, 就构成最简单的 SR 锁存器

SR 锁存器电路是由两个交叉耦合的或非门构成的。我们只需要把单环路存储元件中的反相器简单地替换成或非门就可以得到 SR 锁存器。锁存器有两个输入端 S 和 R, 其中 S 用于置位, R 用于复位。当输出  $Q = 1$  且  $\bar{Q} = 0$  时, 称锁存器处于置位状态 (set state); 当  $Q = 0$  且  $\bar{Q} = 1$  时, 称锁存器处于复位状态 (reset state)。输出 Q 和  $\bar{Q}$  通常是互反的, 当两个输入同时为 1 时, 两个输出都等于 0, 这是个未定义状态。

在通常情况下, 除非要改变状态, 否则锁存器的两个输入信号都将保持为 0。若输出端 S 变为 1, 锁存器进入置位状态。有两种输入条件可以使电路处在置位状态。

初始条件为  $S = 1, R = 0$  时, 电路进入置位状态。当  $S$  重置为 0 且使  $R = 0$ , 电路保持置位状态不变。当两个输入信号返回到 0 之后, 将输入  $R$  置为 1, 电路将进入复位状态。这时, 如果将  $R$  重置为 0, 电路将保持复位状态不变。所以, 当两个输入同时被置为 0 时, 锁存器可能处于置位状态, 也可能处于复位状态, 这取决于最近哪个输入为 1。

如果将锁存器的两个输入都置为 1, 那么两个输出都为 0, 这是衣蛾未定义状态, 因为它违背了两个输出互反的原则。同时, 如果两个输入同时返回 0, 会导致输出的下一个状态不确定或者不可预知。所以为了避免上述情况的发生, 应该保证两个输入不会同时为 1。

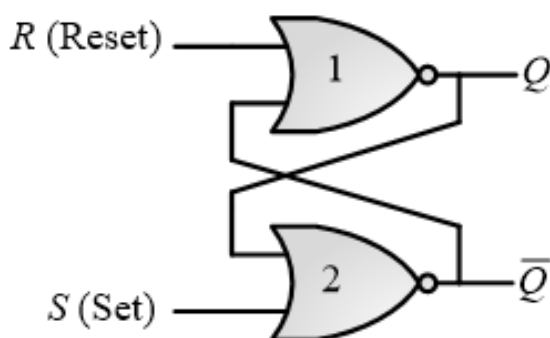


图 1: SR 锁存器

| $R S$ | $Q Q$ | 说明  |
|-------|-------|-----|
| 0 0   | $Q Q$ | 保持  |
| 0 1   | 1 0   | 置1  |
| 1 0   | 0 1   | 置0  |
| 1 1   | 0 0   | 未定义 |

图 2: SR 锁存器功能表

## $\bar{S}\bar{R}$ 锁存器

由两个与非门交叉耦合的锁存器叫做  $\bar{S}\bar{R}$  锁存器, 或者在一些别的资料中叫做 RS 锁存器。

除非要改变锁存器的状态, 否则在通常情况下, 两个输入信号都为 1。将输入  $S$  置为 0 时, 输出  $Q$  变为 1, 锁存器进入置位状态。当输入  $S$  变回 1 时, 电路保持置位状态不变。当两个输入信号都为 1 时, 将输入信号  $R$  的值变为 0, 锁存器的状态会随之

改变, 此时电路进入复位状态, 即两个输入信号都被置为 1 后电路状态仍保持此状态不变。两个输入信号同时等于 0 是未定义状态, 应该避免。

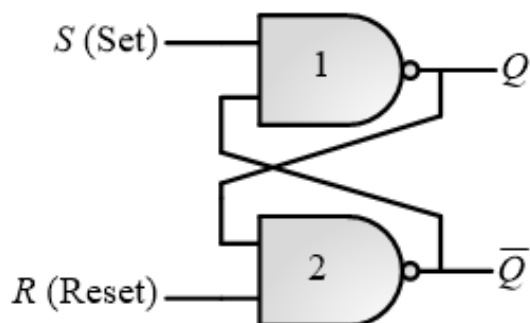


图 3:  $\bar{S}\bar{R}$  锁存器

| $RS$ | $QQ$ | 说明  |
|------|------|-----|
| 0 0  | 1 1  | 未定义 |
| 0 1  | 0 1  | 置0  |
| 1 0  | 1 1  | 置1  |
| 1 1  | $QQ$ | 保持  |

图 4:  $\bar{S}\bar{R}$  锁存器功能表

## 门控 SR 锁存器

门控 SR 锁存器是由两个输入端 S 和 R 以及一个控制端 C 构成的。控制端 C 和两个与非门构成了类似于非门的作用, 当控制端  $C=0$ , 两个与非门输出都为 1, 外部的锁存器处于保持状态, 当  $C=1$  时, 两个与非门输出都与输入为相反的状态, 总体构成了一个 SR 锁存器。

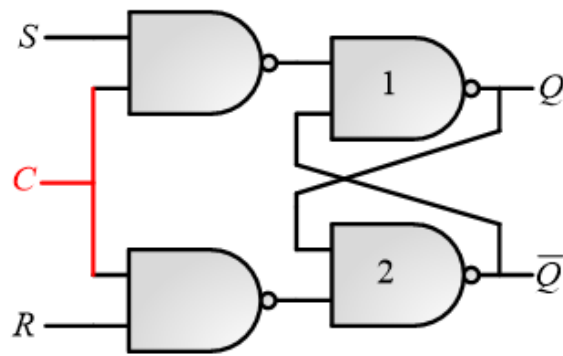


图 5: 门控 SR 锁存器

| $CRS$           | $Q\bar{Q}$ | 说明  |
|-----------------|------------|-----|
| $0\times\times$ | $Q\bar{Q}$ | 保持  |
| $100$           | $Q\bar{Q}$ | 保持  |
| $101$           | $10$       | 置1  |
| $110$           | $01$       | 置0  |
| $111$           | $11$       | 未定义 |

图 6: 门控 SR 锁存器功能表

## D 锁存器

基本锁存器缺点: 存在不确定状态

解决方法: 消除不确定状态所以, 我们就需要 D 锁存器

我们知道, 基本锁存器存在不确定状态是因为 SR 输入端有可能存在同时取 1 的情况, 那么, 只要让其无法同时取 1 即可, D 锁存器只有两个输入端, 数据输入端 D 和控制端 C, D 接入门控 SR 锁存器的 S 端,  $\bar{D}$  接入门控 SR 锁存器的 R 端, 当 C=1 时也类似于一个 SR( $D\bar{D}$ ) 锁存器, 输出端 Q 等于输入端 D。

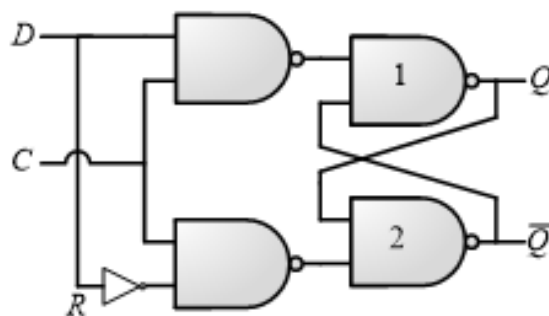


图 7: D 锁存器

| $CD$       | $Q \bar{Q}$ | 说明 |
|------------|-------------|----|
| $0 \times$ | $Q \bar{Q}$ | 保持 |
| $1 0$      | $0 1$       | 置0 |
| $1 1$      | $1 0$       | 置1 |

图 8: D 锁存器功能表

## 锁存器的空翻现象

**空翻现象**：又称为竞态现象，是数字电路中的一个术语，指在同一个时钟脉冲信号作用区间内，由于时钟脉冲的宽度过大，触发器出现在“0”“1”两逻辑信号中多次翻转的现象。

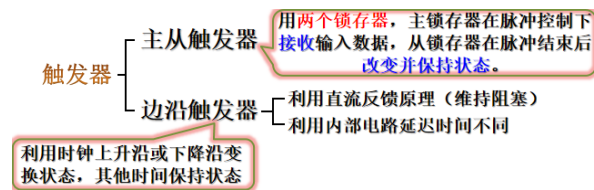
**D 锁存器的缺点**：存在空翻现象，即如果 D 锁存器直接用在时序电路中作为状态存储元件，当使能控制信号有效时，会导致该元件内部的状态值随时多次改变，而不是保持所需的原始状态值

**解决方法**：消除空翻现象，使每次触发仅使锁存器的内部状态仅改变一次

## 触发器

**触发**：外部输入使锁存器状态改变的瞬间状态

**触发器**：在锁存器的基础上使每次触发仅使状态改变一次的锁存电路（双稳态）



常见的触发器有：主从 SR 触发器、D 触发器、JK 触发器、T 触发器等

## SR 主从触发器

- 由两个钟控 S-R 锁存器串联构成，第二个锁存器的时钟通过反相器取反
- $C=1$  时，输入信号进入第一个锁存器（主锁存器）
- $C=0$  时，第二个锁存器改变输出（从锁存器）
- 从输入到输出的通路被不同的时钟信号所断开

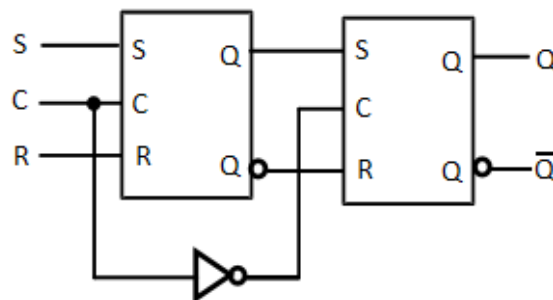


图 9: SR 主从触发器



正边沿维持阻塞型 D 触发器

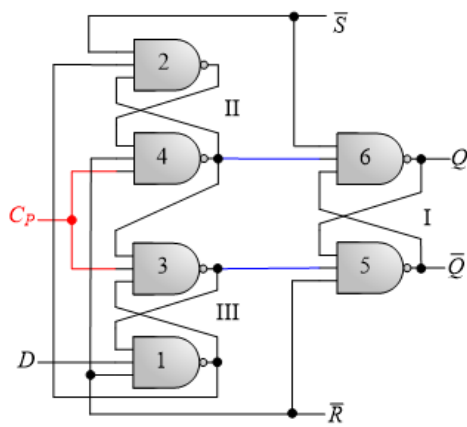


图 10: D 触发器

| 异步控制 |     | 上升沿触发 |     |     |           |
|------|-----|-------|-----|-----|-----------|
| $R$  | $S$ | $C_p$ | $D$ | $Q$ | $\bar{Q}$ |
| 0    | 1   | ×     | ×   | 0   | 1         |
| 1    | 0   | ×     | ×   | 1   | 0         |
| 1    | 1   | ↑     | 0   | 0   | 1         |
| 1    | 1   | ↑     | 1   | 1   | 0         |

图 11: D 触发器功能表

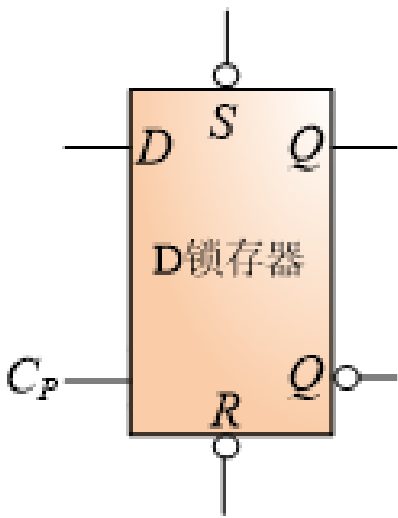


图 12: D 触发器逻辑图

分析电路图，可知

- SR=00 时，输出未定义
- 当 SR=01, 输出为 10
- 当 SR=10, 输出位 01, 类似于 S 非 R 非锁存器
- 当 SR=11, 时,  $C_p$  上升沿触发时, 输出更新为与 D 的输入一致, 其余情况就保持不变

### 三、实验步骤和结果记录

#### $\bar{S}\bar{R}$ 锁存器仿真

在 Digital 中画出原理图如图

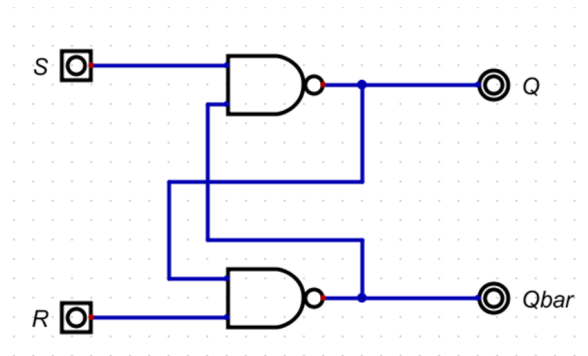


图 13:  $\bar{S}\bar{R}$  锁存器原理图

在 Vivado 仿真

仿真代码如下

```
1  module tb_lock(
2
3  );
4  reg S;
5  reg R;
6  wire Q;
7  wire Qbar;
8  SR_bar UUT(
9      .S(S),
10     .R(R),
11     .Q(Q),
12     .Qbar(Qbar)
13 );
14 initial begin
```

```

15         R=1;S=1; #50;
16         R=1;S=0; #50;
17         R=1;S=1; #50;
18         R=0;S=1; #50;
19         R=1;S=1; #50;
20         R=0;S=0; #50;
21         R=1;S=1; #50;
22     end
23 endmodule

```

仿真结果如图

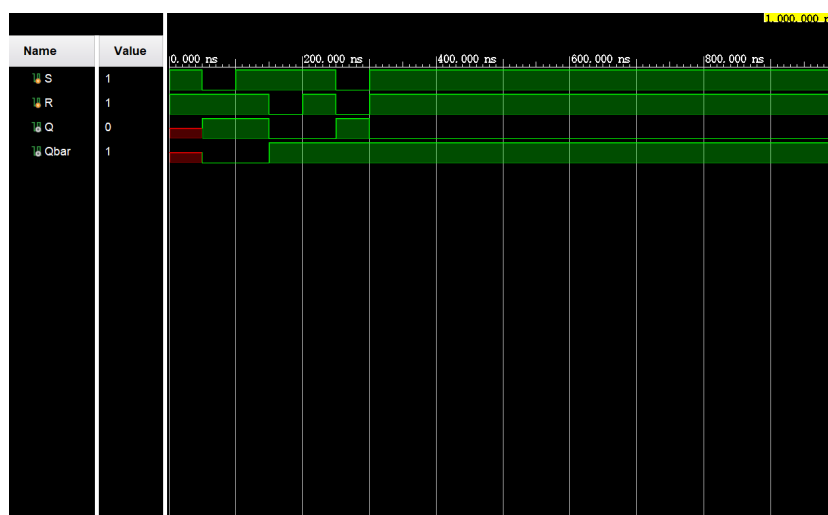


图 14: SR 锁存器仿真

可以看到, 最开始锁存器中的存储值是未知的, 当  $S=0$   $R=1$  时, 我们存入了 1, 因此 Q 变为 1; 然后保存一次, 当  $S=1$   $R=0$  时, 我们存入了 0; 然后保存一次, 当 S R 同时从 0 变为 1 时, Q 和 Q 非的结果是不定的, 取决于门的延迟。仿真结果和 SR 锁存器真值表相同, 符合预期

## 门控 SR 锁存器仿真

在 Digital 中画出原理图如图

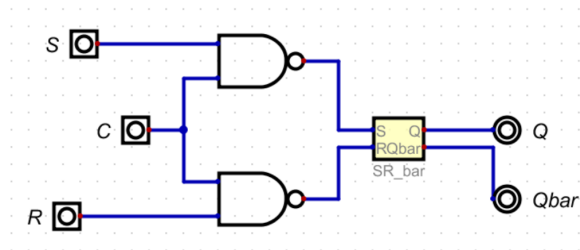


图 15: 门控 SR 锁存器原理图

在 Vivado 中仿真

仿真代码如下

```
1  module tb_lock(  
2  
3  );  
4  reg C;  
5  reg S;  
6  reg R;  
7  wire Q;  
8  wire Qbar;  
9  CSR UUT(  
10     .C(C),  
11     .S(S),  
12     .R(R),  
13     .Q(Q),  
14     .Qbar(Qbar)  
15  );  
16  initial begin  
17     C=1;S=0;R=0;#50;  
18     C=1;S=0;R=1;#50;  
19     C=1;S=0;R=0;#50;  
20     C=1;S=1;R=0;#50;  
21     C=1;S=0;R=0;#50;
```

```

22         C=1;S=1;R=1;#50;
23         C=1;S=0;R=0;#50;
24         C=0;S=0;R=0;#50;
25         C=0;S=0;R=1;#50;
26         C=0;S=0;R=0;#50;
27         C=0;S=1;R=0;#50;
28         C=0;S=0;R=0;#50;
29         C=0;S=1;R=1;#50;
30         C=0;S=0;R=0;#50;
31     end
32 endmodule

```

仿真结果如图

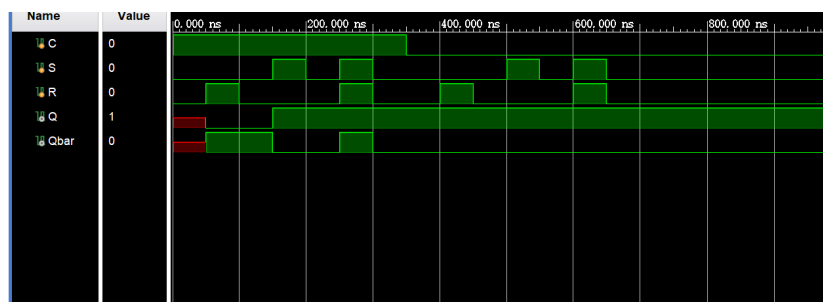


图 16: 门控 SR 锁存器仿真

可以看到, 最开始锁存器中的存储值是未定义的, 在  $C=1$  的情况下, 由仿真代码可知, 我们依次存入了 0, 保持, 1, 保持, 未定义, 保持, 当  $C=0$  时, 锁存器处于保持状态, 无论输入如何变化, 锁存器的状态都不会改变, 仿真结果和门控 SR 锁存器真值表相同, 符合预期

## D 锁存器仿真

在 Digital 中画出原理图如图

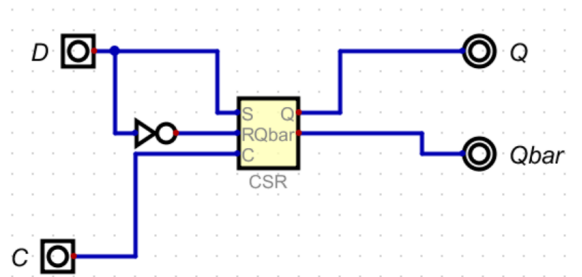


图 17: D 锁存器原理图

## 在 Vivado 中仿真

仿真代码如下

```

1      module tb_lock(
2
3      );
4      reg C;
5      reg D;
6      wire Q;
7      wire Qbar;
8      D UUT(
9          .C(C),
10         .D(D),
11         .Q(Q),
12         .Qbar(Qbar)
13     );
14     initial begin
15         C=1;D=1;#50;
16         C=1;D=0;#50;
17         C=0;D=0;#50;
18         C=0;D=1;#50;
19     end
20 endmodule

```

仿真结果如图

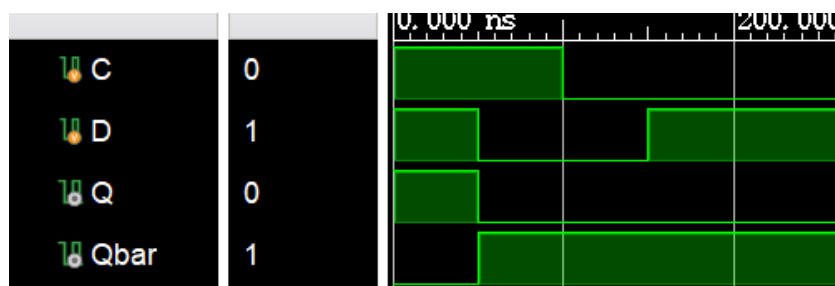


图 18: D 锁存器仿真

由仿真图可知，一开始  $C=1$ ， $Q$  随  $D$  而变化， $C=0$  之后， $Q\bar{Q}$  保持不变，仿真结果与真值表相同，符合我们的预期。

## 验证 D 锁存器的空翻现象

在 Digital 中画出原理图如图

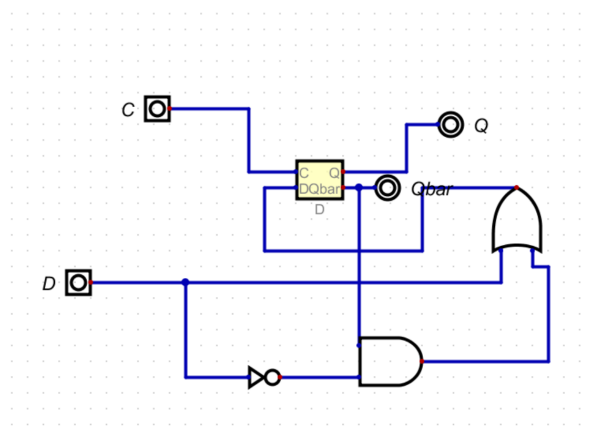


图 19: D 锁存器空翻现象原理图

基本想法为，一开始令输入端为 1，此时  $Qbar$  为 0，整个电路正常，当输入端变为 0 时， $Qbar$  变为 1，此时  $Q$  变为 0， $Qbar$  变为 1，相当于把  $Qbar$  接入到输入端，所以输入端的信号一直在变化，产生震荡。结果如图



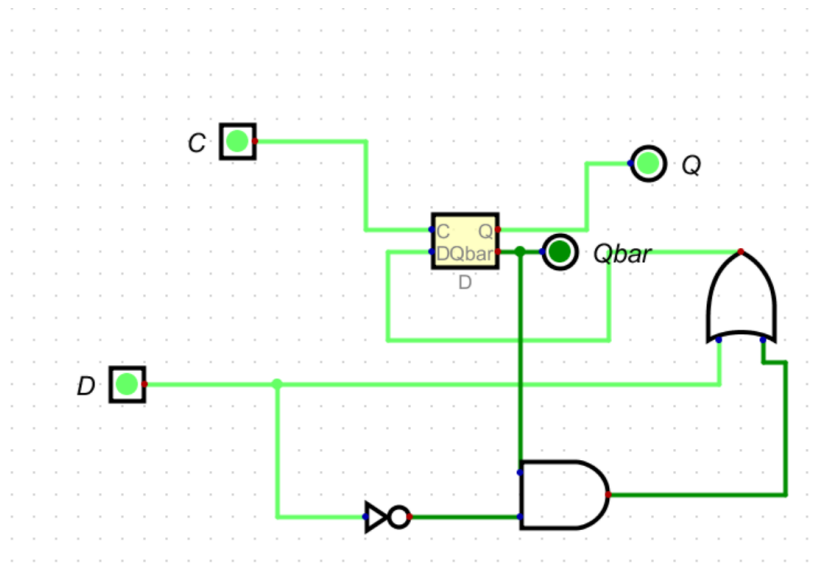


图 20: D 为 1 时正常

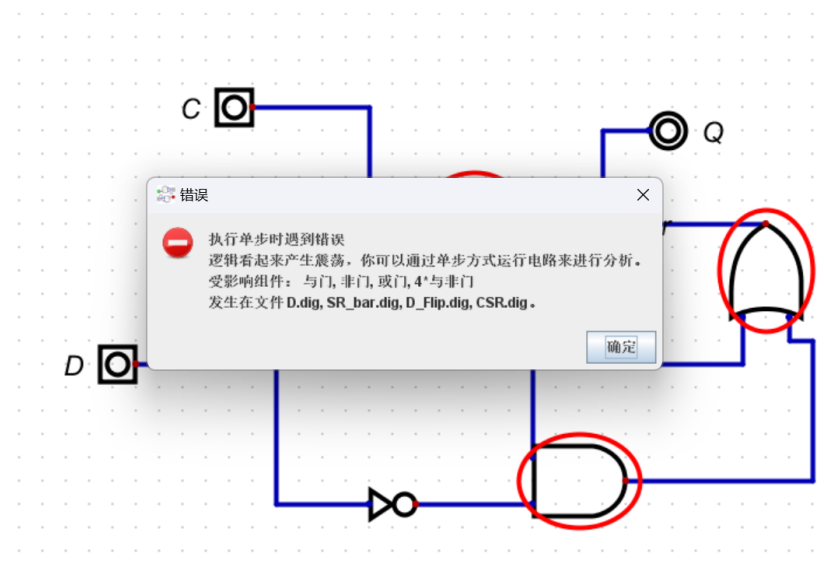


图 21: D 为 0 时空翻

导出文件在 Vivado 中仿真

仿真代码如下

```

1  module tb_lock(
2
3  );
4  reg C;

```

```

5     reg D;
6     wire Q;
7     wire Qbar;
8     D_Flip UUT(
9         .C(C),
10        .D(D),
11        .Q(Q),
12        .Qbar(Qbar)
13    );
14    initial begin
15        C=1;#10;
16        D=1;#10;
17        D=0;#10;
18    end
19 endmodule

```

仿真结果如图

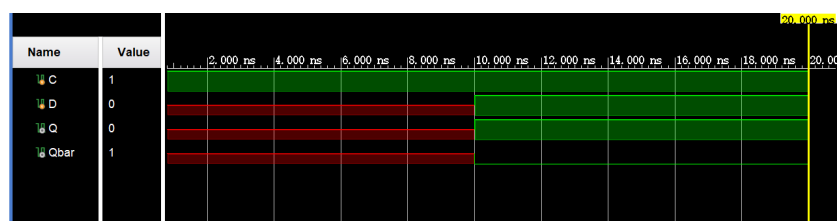


图 22: D 锁存器空翻现象仿真

可以看到，我们的仿真波形图停止在 18ns，之后不再生成，但是会看到各个数值仍然在不停变化，如下图

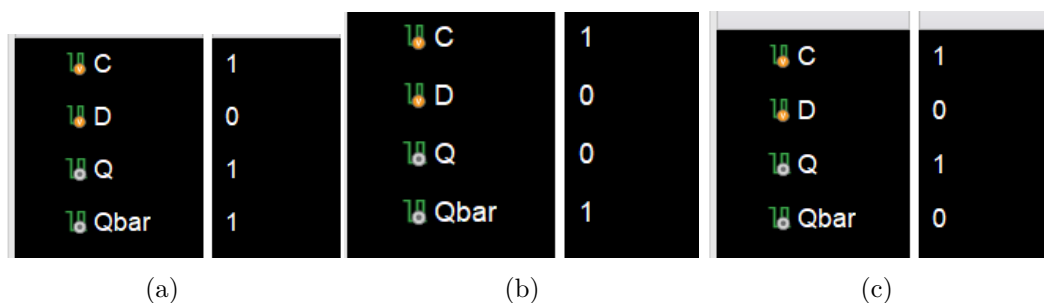


图 23: 震荡现象

说明出现了空翻现象

## SR 主从触发器仿真

在 Digital 中画出原理图如图

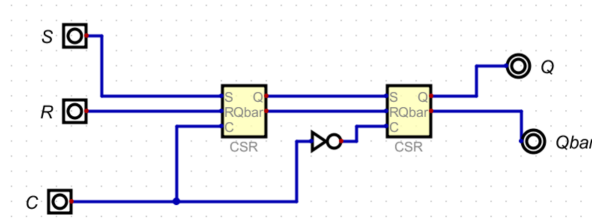


图 24: SR 触发器原理图

## 在 Vivado 中仿真

仿真代码如下

```
1  module tb_lock(  
2  
3  );  
4  reg C;  
5  reg S;  
6  reg R;  
7  wire Q;  
8  wire Qbar;  
9  TriSR UUT(  
10     .C(C),  
11     .S(S),  
12     .R(R),  
13     .Q(Q),  
14     .Qbar(Qbar)  
15 );  
16  
17     always begin  
18         C=0;#20;  
19         C=1;#20;  
20     end  
21  
22     initial begin
```

```

23      R=0;S=0; #50;
24      R=1;S=0; #50;
25      R=0;S=0; #40;
26      R=0;S=1; #5; //这里用于验证SR触发器的一次性采样问题
27      R=0;S=0; #5;
28      R=0;S=0; #50;
29      R=1;S=0; #40;
30      R=0;S=0; #50;
31      R=0;S=1; #50;
32      R=0;S=0; #50;
33      R=1;S=1; #50;
34  end
35  endmodule

```

这个代码设计了一个时钟信号，每 20ns 变化一次，并且在 140ns 的时候，产生了一个用于测试一次性采样的小脉冲仿真结果如图

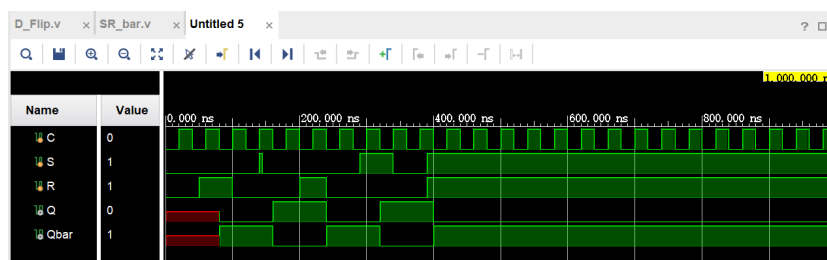


图 25: SR 触发器仿真

可以看到，小脉冲产生的时候，C 为 1，把这个小脉冲存进了主锁存器里面，即使后面 S 重新变为了 0，这个信号仍然还在，当 C 变为 0 时，把这个信号传给了从锁存器，造成了输出的变化，这就是 SR 主从触发器的一次性采样问题。

对于其他 SR 触发器的仿真结果和真值表相同，C=1 时存到主锁存器中，C=0 时体现在输出中。符合预期

## D 主从触发器仿真

在 Digital 中画出原理图如图

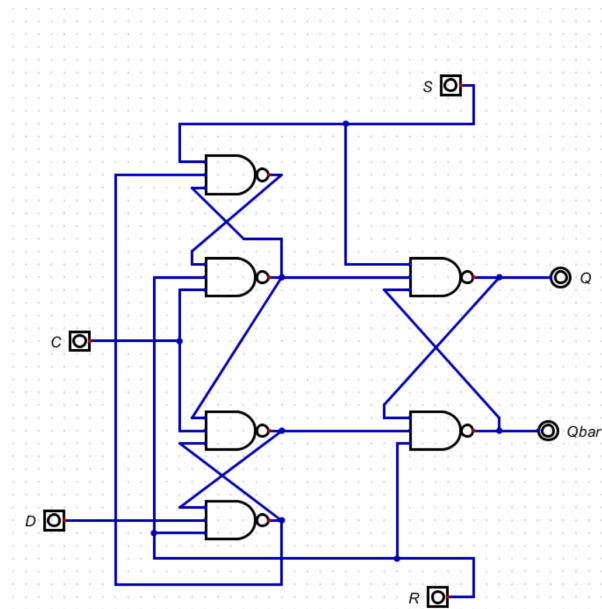


图 26: D 触发器原理图

## 在 Vivado 中仿真

仿真代码如下

```
1  module tb_lock(
2
3      );
4      reg C;
5      reg S;
6      reg R;
7      reg D;
8      wire Q;
9      wire Qbar;
10     TriD UUT(
11         .C(C),
12         .S(S),
13         .D(D),
14         .R(R),
```

```

15         .Q(Q),
16         .Qbar(Qbar)
17     );
18
19     always begin
20         C=0;#20;
21         C=1;#20;
22     end
23     integer i;
24     initial begin
25         R=0;S=1;D=0;#50;
26         R=0;S=1;D=1;#50;
27         R=1;S=0;D=0;#50;
28         R=1;S=0;D=1;#50;
29         R=1;S=1;D=1;
30     forever begin
31         #40 D=~D;
32     end
33 end
34 endmodule

```

仿真结果如图

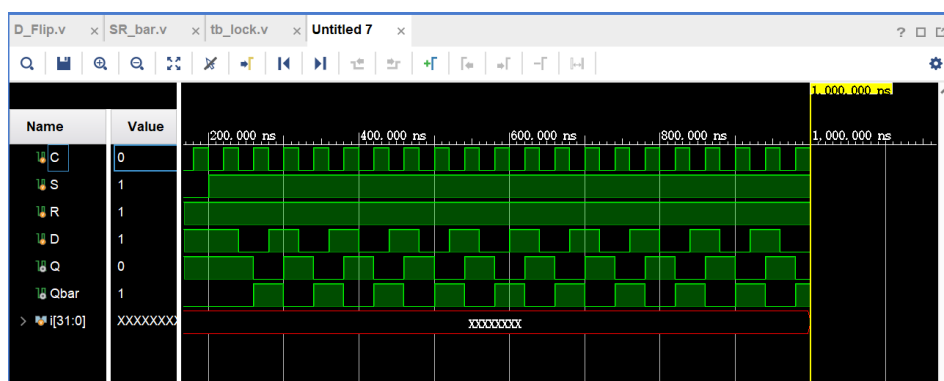


图 27: D 触发器仿真

可以看到, 当 RS 输入信号是 01 或者 10 时输出信号与  $\overline{SR}$  锁存器一致, 当 RS 输入信号是 11 时, 输出信号在 C 为上升沿时, 会把 D 的输入传递到输出, 符合我们的预期。故仿真正确

## 四、实验结果分析

本实验没有使用 SWORD 板，只是进行了仿真。相关结果和 Verilog 代码都已经在前文写出。实验结果基本符合要求：仿真激励波形与真值表都相对应。

## 五、讨论与心得

虽然本次实验没有上板，只需要画图以及仿真，但是需要自己设计仿真代码，还需要在 Digital 中画较多的电路图，在做最后一个 D 触发器的电路图的时候，我发现虽然画出的电路图看起来比较麻烦。但是导出的 Verilog 代码却很简单，只有几行。这让我感受到了 Verilog 的强大之处。

总的来说，本次实验让我对触发器以及锁存器有了更深的理解，也让我对 Vivado 和 Digital 的使用更加熟练，验收也十分顺利，希望在以后的实验中能够更好的运用 Verilog 进行设计。