

第 5 章 双稳态存储器件的 VHDL 设计

本章内容提要

5.1 引言

5.2 S-R NOR 锁存器分析

5.3 S-R NAND 锁存器分析

5.4 设计一个简单的时钟

5.5 设计一个 D 锁存器

5.6 设计 D 触发器电路

习题

5.1 引言

本章将介绍双稳态存储器件 (bistable memory devices) 的设计。“双稳态”指的是设备具有两个稳定状态，可以用于写入、保持或读取逻辑 1 或逻辑 0 的值。本章讨论包括 S-R 锁存器 (set-reset (S-R) latches)、D 锁存器 (D latches)、边沿触发的 D 触发器 (edge-triggered D flip-flops) 等双稳态器件的设计、分析以及它们的工作特性。本章还介绍一个简单的时钟设计。在第 6 章同步时序逻辑电路 (synchronous sequential logic circuits) 的设计中会用到本章介绍的技术和器件。

5.2 S-R NOR 锁存器分析

寄存器是一种可以暂时存储单个或多个比特的数字单元电路。锁存器则是最简单的单比特寄存器。两个交叉耦合的 NOR 门或 NAND 门可以构成一个置位-复位 (S-R) 锁存器。首先以一个电灯开关为例，因为它的工作原理与交叉耦合 NOR 门构成的锁存器的工作原理相似。

5.2.1 简单的电灯开关

试想一个墙上的电灯开关，它有两个控制信号：一个叫 S 的信号，用于将灯泡置位或点亮，一个叫 R 的信号，用于将灯泡复位或熄灭。S 和 R 信号各自都有两种可能的取值。可以拨动开关使灯泡点亮 (S=1)，然后停止拨动开关，则灯保持在点亮状态 (S=0)。也可以把开关拨至灯熄灭的位置 (R=1)，然后停止拨动开关 (R=0)，则灯保持在熄灭状态。不

拨动开关使得灯亮 ($S=0$)，并且不拨动开关使得灯灭 ($R=0$)，则电灯保持其现态。这个思想很简单，大家不妨在房间里试试墙上的电灯开关。图 5.1 所示就是电灯开关（机械的锁存器）在不同 S 、 R 取值下的 5 种状态，其中开关往上拨电灯点亮，往下拨则电灯熄灭。

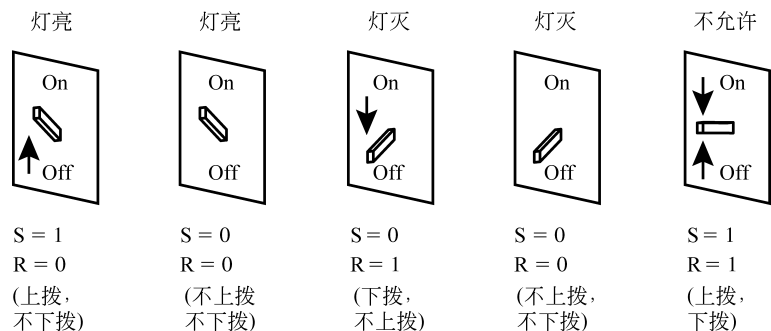


图 5.1 一个简单电灯开关的 5 种不同状态

图中的箭头表示的是为改变开关位置对其所施的力及方向。如果没有箭头，则没有对开关施加外力。一旦有力施加，电灯开关就保存了其所在的新的置位或复位位置，然后这个力就可以撤销了。由于其保存了上一次所施加力的方向，这种开关具有记忆功能。也正因为这种记忆功能，不用像瞬时按钮开关那样，要想灯亮就得站在它边上一直摁着。

图 5.1 中的电灯开关捕获并保存了其所在的灯亮位置，因此不用一直推着它来让灯泡点亮。类似地，开关捕获并保存了其所在的灯灭位置，因此不用一直压着它来让灯泡熄灭。

表 5.1 给出电灯开关的特性表。

表 5.1 电灯开关的特性表

S R	灯泡状态	S R	灯泡状态
0 0	保持（开或关）	1 0	开
0 1	关	1 1	禁止

电灯开关和 S - R NOR 锁存器（由两个交叉耦合的 NOR 门构成的数字电路）有相似的特性表。它们都具有记忆功能，都存储了一个值。

到目前为止，只涉及到了组合逻辑电路（combinational logic circuits），这些电路不存在反馈，因而均没有记忆能力。存在反馈的电路称为时序逻辑电路（sequential logic circuits），它们具有记忆能力。由于反馈的存在，时序逻辑电路的分析和设计较为困难。一个基本 S - R NOR 锁存器的分析方法包括：

- (1) 电路延迟模型。
- (2) 特性表。
- (3) 特征方程。
- (4) PN/NS（现态/次态）表。
- (5) 时序图。

5.2.2 S-R NOR 锁存器的电路延迟模型

图 5.2(a)所示是 S-R NOR 锁存器的门级电路，图 5.2(b)是其逻辑符号。

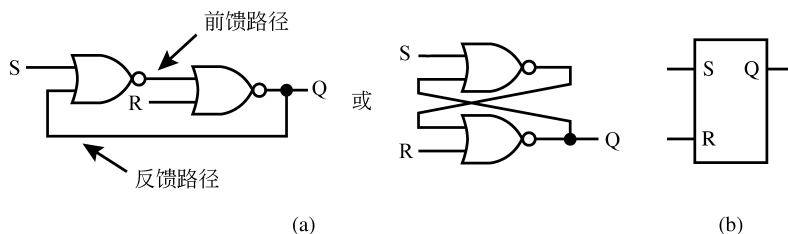


图 5.2 S-R NOR 锁存器：(a)门级电路；(b)逻辑符号

图 5.3 是 S-R NOR 锁存器的电路延迟模型。其门级电路是一个数字电路，现态的输出 Q (present-state output Q) 反馈至其输入。这是为什么 S-R NOR 锁存器能够保存数据的秘密。

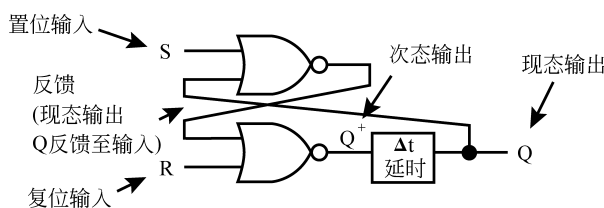


图 5.3 S-R NOR 锁存器的电路延迟模型

如果一个 S-R 锁存器被置位，其输出 Q 变为 1；如果其被复位，输出变为 0。每一个 NOR 门都有传播延时。最坏情况下该电路的延时记为 Δt ，表示从输入 S 到输出 Q 的最长延时或 $2t_p$ ，其中假设每一个门的延时都是 t_p 。在此延迟模型里，只是简单地把电路的最坏情况延时放在一个延迟框里，而所有的门都被认为是没有延时，因为电路整体的延时都由这个延时框所表示。这样把电路看作是集总参数电路（在这里参数是延时），而非分布参数电路，解释起来更容易些，虽然实际的电路是分布参数的。

在此延迟模型里，若 S 或 R 的值变为 1，则次态输出 Q^+ (next-state output Q^+) 立即响应 S 或 R 的变化。现态输出 Q 是在 S 和 R 变化前，电路的输出值。只有过了延时 Δt 后， Q 才会变化为新的现态输出值，即 $Q=Q^+$ 。如果用电灯开关来类比的话，就是灯泡需要经过一段时间才能从灭变到亮（就像 Q 需要 Δt 的时间才能变为 1），也要一定时间才能从亮变为灭（就像 Q 需要 Δt 的时间才能变为 0）。图 5.4(a)~图 5.4(e)所示的各个电路分别表示了图 5.1 中的各个状态。

图 5.4(e)所示的状态一般是不允许的，因为输出不可能同时被置位和复位。不过如果尝试同时置位和复位一个 S-R NOR 锁存器，那么电路的输出是复位状态 ($Q=0$)；也就是说复位输入优先于置位输入。也正因为此，S-R NOR 锁存器电路被称为是复位占优的。

初始启动，若 S-R NOR 锁存器电路刚上电时输入 S 和 R 均为 0，则输出 Q 暂时处在不确定状态，因为输出 Q 可能为 0 也可能为 1。

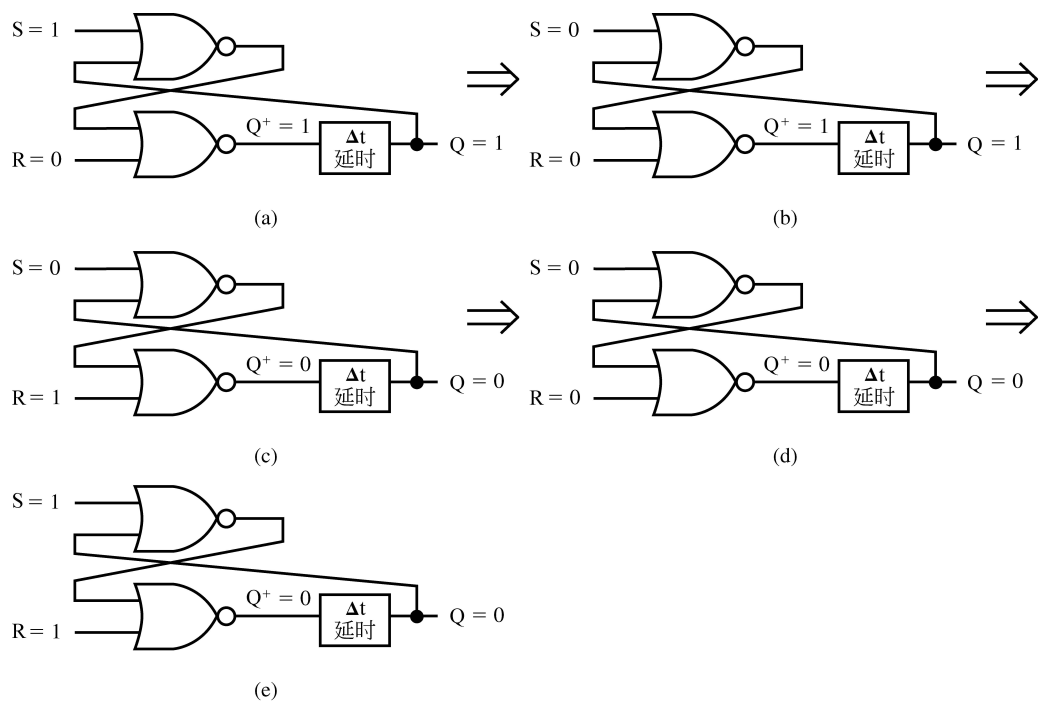


图 5.4 S-R NOR 锁存器的 5 种不同状态

5.2.3 S-R NOR 锁存器的特性表

S-R NOR 锁存器输入是高电平有效的。按照通常的逻辑表示，1=高（H），0=低（L）。S=1，R=0 时，Q 变为 1 或高。R=1，S=0 时，Q 变为 0 或低。表 5.2 所示是 S-R NOR 锁存器的特性表。

表 5.2 S-R NOR 锁存器的特性表

		备注
S R	Q^+	次态
0 0	Q	现态（高或低）
0 1	0	低
1 0	1	高
1 1	0	复位优先（正常情况下不允许）

观察可发现，如果同时对 S-R NOR 锁存器进行置位和复位，电路的输出将变为复位状态，因为该电路是复位占优的。

5.2.4 S-R NOR 锁存器的特征方程

锁存器电路所满足的布尔方程称为特征方程，因为它以方程的形式描述了一个锁存器是如何被构建的。根据 S-R NOR 锁存器的延迟模型，可以写出图 5.5 所示电路的特征方程。

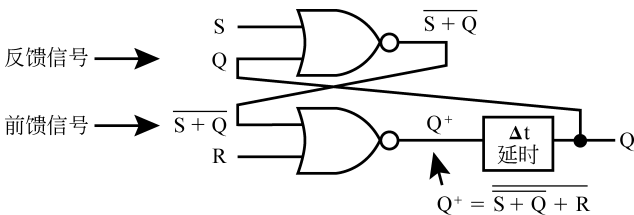


图 5.5 S-R NOR 锁存器的特征方程

分析这个电路，会发现 $Q^+ = \overline{\overline{S+Q} + R}$ 。方程右边的变量 Q 是现态输出，方程左边的 Q^+ 是次态输出。

一个组合逻辑方程总是有 $F(A,B,C,\dots)$ 的形式，表示函数只依赖于输入 $A,B,C,\dots$ 的取值。一个时序逻辑方程（用于具有记忆功能的电路的方程）则有 $F^+(F,A,B,C,\dots)$ 的形式，表示函数依赖于作为输入的其自身的输出，以及其他输入。函数 Q^+ 有形式 $Q^+(Q,S,R)$ ，表示该函数依赖于作为输入的其自身的输出，以及输入 S 和 R 。当 S-R NOR 锁存器的输出稳定后，现态输出信号 Q 在延时 Δt 后变为 Q^+ ，即在 Δt 后 $Q=Q^+$ 。

5.2.5 S-R NOR 锁存器的 PS/NS 表

时序逻辑电路的真值表称为现态/次态表或 PS/NS 表。之所以称之为现态/次态表是因为在这个表里，现态（PS）的输出信号 Q 被列为一个输入（反馈输入），紧接着的是次态（NS）的输出信号 Q^+ 。S-R NOR 锁存器 PS/NS 表的得到有两种方式：一种是对其特征性表进行展开；另一种是利用 SOP 形式的特征方程来填 PS/NS 表。我们把 SOP 形式（与/或形式）的特征方程重写如下：

$$Q^+ = \overline{\overline{S+Q} + R} = \overline{\overline{Q+S} \cdot \overline{R}} = (Q+S) \cdot \overline{R} = Q \cdot \overline{R} + S \cdot \overline{R}$$

从特性方程中可以发现只有当 $Q \cdot \overline{R} = 1$ 或者 $S \cdot \overline{R} = 1$ 时， $Q^+=1$ ，否则 $Q^+=0$ 。

表 5.3 所示是 S-R NOR 锁存器的 PS/NS 表。

表 5.3 S-R NOR 锁存器的 PS/NS 表

S	R	Q ⁺
0	0	Q
0	1	0
1	0	1
1	1	0

⇒

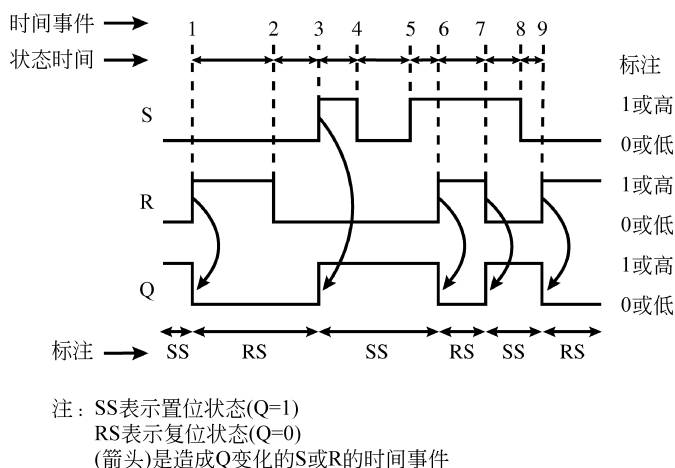
S	R	Q	Q ⁺
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

5.2.6 S-R NOR 锁存器的时序图

导致时序逻辑电路输出发生变化的事件称为时序事件 (timing events)。任意一个外部输入 S 或 R 的变化都构成 S-R NOR 锁存器的一个时序事件。理想时序图 (Ideal timing diagrams)，或波形图表征了电路相关信号的逻辑值 (1 和 0)。理想时序图中不包含上升时间 (rise time) 和下降时间 (fall times) 信息，每个门的门延时都认为是 0。这是画时序图最简单的方式。

一个信号的上升时间或下降时间通常被定义为信号转换过程中变化 10% 到变化 90% 所需的时间。一个正脉冲的脉冲宽度 (pulse width) 通常被定义为信号从上升沿 50% 的位置到下降沿 50% 的位置所经历的时间。负脉冲的脉冲宽度定义与此类似。为简化时序图，门延时或以平均传播延时 $t_p = (t_{PHL} + t_{PLH})/2$ 来表示，或以更简单的方式：门延时为 0。 t_{PHL} 表示输出从高 (H) 变为低 (L) 情况下通过一个门的延时， t_{PLH} 则表示输出从低 (L) 变为高 (H) 情况下通过一个门的延时。这两个延时不总是相等。

波形 5.1 所示是一个 S-R NOR 锁存器电路的理想时序图。



波形 5.1 S-R NOR 锁存器的理想时序图

在波形 5.1 中，首先画出了输入 S 和 R 的波形，随后是输出 Q 的波形。输出 Q 必须遵循 S-R NOR 锁存器的特性表，随输入信号的变化而变化。时间从左到右推演。每次外部输入的变化，无论是 1 变到 0 抑或是 0 变到 1，都构成一个时序事件。状态时间 (state time) 则是相邻两个时序事件之间的时间。

代码段 5.1(a) 给出一个 S-R NOR 锁存器的完整 VHDL 设计。

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity S_R_NOR_LATCH is port (
    S, R : in std_logic;
    Q : inout std_logic
);
```

```
end S_R_NOR_LATCH;

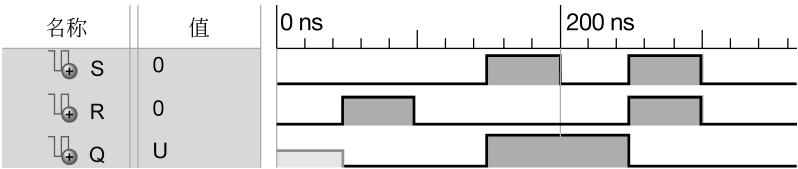
architecture dataflow of S_R_NOR_LATCH is
begin
    Q <= (Q and not R) or (S and not R);
end dataflow;
```

代码段 5.1a S-R NOR 锁存器的完整 VHDL 设计（工程：S_R_NOR_LATCH_Bool）

代码段 5.1a 的 VHDL 设计中，应注意：

- 输出信号 Q 在实体中应被标识为 inout 模式。
- 必须读取输出信号 Q，因为它出现在赋值语句的右边；因此它必须是 in 模式。由于输出信号 Q 同样出现在了赋值语句的左边，它必须是 out 模式。关键词 inout 用于标识既是输入又是输出的模式。关键词 buffer 也可替代关键字 inout 用于标识非双向的信号。
- 设计采用了含布尔方程的数据流风格。其中的布尔方程是 S-R NOR 锁存器的特征方程。

波形 5.2 是设计实体 S_R_NOR_LATCH 的正确仿真结果（行为仿真）。



波形 5.2 设计实体 S_R_NOR_LATCH 的正确仿真结果

记住：在初始启动时，如果没有设定 S 或 R（即不为 1），那么在 Q 的波形图里可以看到，Q 可能暂时性地不确定（输出介于 0 和 1 之间）。观察发现，波形 5.2 中的仿真结果服从表 5.2 中 S-R NOR 锁存器的特性表，证明这段 VHDL 代码的设计是正确的。

代码段 5.1(b)展示了另一种设计 S-R NOR 锁存器的方式。

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity S_R_NOR_LATCH_int is port (
    S, R : in std_logic;
    Q : out std_logic
);
end S_R_NOR_LATCH_int;

architecture dataflow of S_R_NOR_LATCH_int is
    signal Q_int: std_logic;
begin
    Q_int <= (Q_int and not R) or (S and not R);
    Q <= Q_int;
end dataflow;
```

代码段 5.1(b) 另一种设计 S-R NOR 锁存器的方式（工程：S_R_NOR_LATCH_Bool_int）

代码段 5.1(b)的 VHDL 设计中，应注意：

- 输出信号 Q 在实体中应被标识为 out 模式。

- 可以利用一个中间信号 Q_int （意为 Q 内部）来表示锁存器布尔方程中的信号 Q ，而不用 inout 模式。这个中间信号是一个内部信号，因此设计中必须在 architecture 和第一个 begin 之间用 Signal $Q_int:std_logic$ 来标识。
- 输出信号 Q 通过布尔方程 $Q \leq Q_int$ 来产生。

代码段 5.1(b)给出的仿真波形图与波形 5.2 中的波形图相同。Xilinx ISE 软件对代码段 5.1(a)和代码段 5.1(b)所生成的电路也是一样的。为使 VHDL 代码更简洁，本书中对那些既作为反馈输入又作为输出的变量，选择使用 inout 模式，因为这样可使代码行数更少，同时也可减少所需变量的数目。

5.3 S-R NAND 锁存器分析

本节将介绍一个基本 S-R NAND 锁存器的性质。读者会发现一个基本 S-R NAND 锁存器输入是低电平有效的，这与 S-R NOR 锁存器不同，S-R NOR 锁存器输入是高电平有效的，还会发现 S-R NAND 锁存器是置位占优的，而不像 S-R NOR 锁存器那样是复位占优的。

5.3.1 S-R NAND 锁存器电路延迟模型

具有多个 S 和 R 输入的 S-R NAND 锁存器的门级电路及逻辑符号如图 5.6(a)和图 5.6(b)所示。图中用反向输入的或门来表示与非门，这样等效的表示可以帮助我们想起 S-R NAND 锁存器输入是低电平有效的。这种锁存器有些时候也称为 $\bar{S}-\bar{R}$ NAND 锁存器。图 5.6(b)中的逻辑符号在本章还将用于构成 D 触发器电路。观察还将发现，输出信号 $\bar{Q}$ 是由前馈通路所产生的。

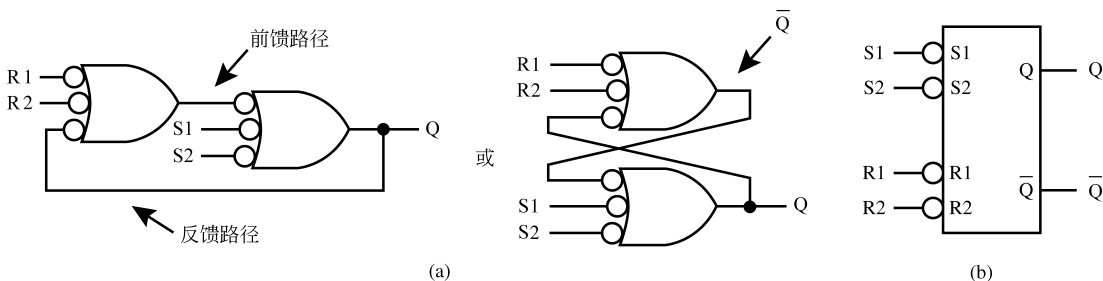


图 5.6 S-R NAND 锁存器：(a)门级电路；(b)逻辑符号

图 5.7 所示是 S-R NAND 锁存器的电路延迟模型。

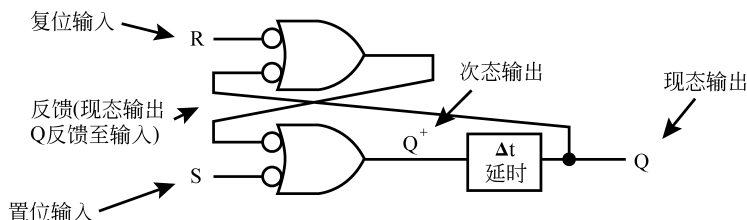


图 5.7 S-R NAND 锁存器的电路延迟模型

初始启动时，若 S-R NAND 锁存器电路刚上电时输入 S 和 R 均为 1，则电路输出 Q 暂时处在不确定状态，因为输出 Q 可能为 0 也可能为 1。

5.3.2 S-R NAND 锁存器的特性表

表 5.4 所示是 S-R NAND 锁存器的特性表。

表 5.4 S-R NAND 锁存器的特性表

		备注
S R	Q+	次态
0 0	1	置位占优（正常情况下不允许）
0 1	1	高
1 0	0	低
1 1	Q	现态（高或低）

观察可发现，如果同时对 S-R NAND 锁存器进行置位和复位，电路的输出将变为置位状态，因为该电路是置位占优的。

5.3.3 S-R NAND 锁存器的特征方程

根据图 5.7 中 S-R NAND 锁存器的延迟模型，可以写出其特征方程 $Q^+ = \overline{\overline{R} + \overline{S} + \overline{S}}$ 。方程右边的变量 Q 是现态输出，而方程左边的变量 Q^+ 则是次态输出。

5.3.4 S-R NAND 锁存器的 PS/NS 表

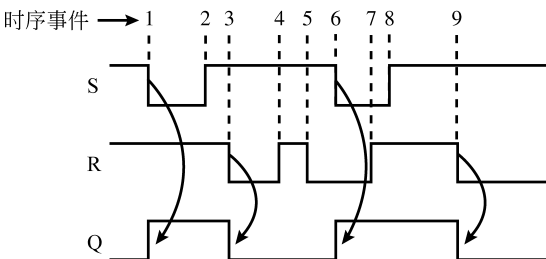
表 5.5 所示是 S-R NAND 锁存器的 PS/NS 表，它可以通过 S-R NAND 锁存器的特性表或者 SOP 形式的特征方程得到。

表 5.5 S-R NAND 锁存器的 PS/NS 表

S	R	Q ⁺	⇒	S	R	Q	Q ⁺
0	0	1		0	0	0	1
0	1	1		0	0	1	1
1	0	0		0	1	0	1
1	1	Q		0	1	1	1
				1	0	0	0
				1	0	1	0
				1	1	0	0
				1	1	1	1

5.3.5 S-R NAND 锁存器的时序图

波形 5.3 是 S-R NAND 锁存器的理想时序图。



波形 5.3 S-R NAND 锁存器的理想时序图

代码段 5.2 是一个 S-R NAND 锁存器的完整 VHDL 设计。

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity S_R_NAND_LATCH is port (
    S, R : in std_logic;
    Q : inout std_logic
);
end S_R_NAND_LATCH;

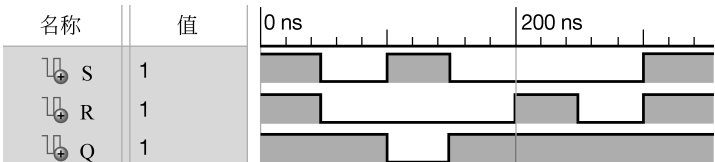
architecture dataflow of S_R_NAND_LATCH is
begin
    Q <= (R and Q) or not S;
end dataflow;
```

代码段 5.2 S-R NAND 锁存器的完整 VHDL 设计（工程 S_R_NAND_LATCH_Bool）

代码段 5.2 的 VHDL 设计中，应注意：

- 输出信号 Q 在实体中被标识为 inout 模式，以使 Q 既可以作为输入被读取，也可作为输出。
- 设计采用了含布尔方程的数据流风格。其中的布尔方程是 S-R NAND 锁存器的特征方程。

波形 5.4 是设计实体 S_R_NAND_LATCH 的正确仿真结果。



波形 5.4 设计实体 S_R_NAND_LATCH 的正确仿真结果

观察发现波形 5.4 中的仿真结果服从表 5.4 中 S-R NAND 锁存器的特性表，证明这段 VHDL 代码的设计是正确的。