

**278** 为了最小化描述的复杂性, 最大化表示形式的灵活性, 允许使用缺省条件, 以及提供一个对实际设计更容易模型化的模型。另外, 利用这个模型可以用硬件描述语言对时序电路进行模型化。

除了逻辑图、状态图和状态表外, 时序电路还可以用 VHDL 或 Verilog 语言进行描述, 这两种描述方式为模拟和自动电路综合提供了有力而灵活的时序电路定义方式。它们都使用了进程这一新的概念, 进程比 VHDL 中的并发赋值语句和 Verilog 中的连续赋值语句有更强的描述能力。进程允许我们像一般程序一样进行编码, 允许使用 if-then-else 和 case 条件语句, 它还可以用来有效地描述组合逻辑。

最后介绍了与触发器相关的定时参数, 建立了时序电路通路延迟与时钟频率的关系。接着, 还讨论了异步信号的同步, 同步电路中的亚稳态现象一些重要话题。

## 参考文献

1. BHASKER, J. *A Verilog HDL Primer*, 2nd ed. Allentown, PA: Star Galaxy Press, 1999.
2. CILETTI, M. *Advanced Digital Design with Verilog HDL*. Upper Saddle River, NJ: Pearson Prentice Hall, 2003.
3. CILETTI, M. *Starter's Guide to Verilog 2001*. Upper Saddle River, NJ: Pearson Prentice Hall, 2004.
4. CLARE, C. R. *Designing Logic Systems Using State Machines*. New York: McGraw-Hill Book Company, 1973.
5. *High-Speed CMOS Logic Data Book*. Dallas: Texas Instruments, 1989.
6. *IEEE Standard VHDL Language Reference Manual (ANSI/IEEE Std 1076-1993; revision of IEEE Std 1076-1987)*. New York: The Institute of Electrical and Electronics Engineers, 1994.
7. *IEEE Standard Description Language Based on the Verilog Hardware Description Language (IEEE Std 1364-1995)*. New York: The Institute of Electrical and Electronics Engineers, 1995.
8. KATZ, R. H. AND G. BORRIELLO. *Contemporary Logic Design*, 2nd ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2005.
9. MANO, M. M. *Digital Design*, 3rd ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2002.
10. PALNITKAR, S. *Verilog HDL: A Guide to Digital Design and Synthesis*, 2nd ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2003.
11. PELLERIN, D. AND D. TAYLOR. *VHDL Made Easy!* Upper Saddle River, NJ: Prentice Hall PTR, 1997.
12. SMITH, D. J. *HDL Chip Design*. Madison, AL: Doone Publications, 1996.
13. STEFAN, S. AND L. LINDH. *VHDL for Designers*. London: Prentice Hall Europe, 1997.
14. THOMAS, D. AND P. MOORBY. *The Verilog Hardware Description Language*, 5th ed. New York: Springer, 2002.
15. WAKERLY, J. F. *Digital Design: Principles and Practices*, 4th ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2006.
16. YALAMANCHILI, S. *VHDL Starter's Guide*, 2nd ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2005.

**279**

## 习题



(+) 表明更深层次的问题, (\*) 表明在原书配套网站上有相应的解答。

4-1 像图 4-5 给出的那样, 构造一个输入序列, 对图 4-6 中的  $\overline{SR}$  锁存器进行人工或计算机逻辑模拟。切记, 此类锁存器的状态在响应 0 而不是 1 时发生变化。

- 4-2 像图 4-5 给出的那样，对图 4-7 中的带有控制输入信号  $C$  的 SR 锁存器进行人工或计算机逻辑模拟。特别要检查当  $C$  等于 1， $S$  和  $R$  发生变化时的电路行为。
- 4-3 图 4-47 给出了上升沿触发式 D 触发器的另外一种较流行的设计方式。通过人工或自动方式模拟电路，以便确定这个电路是否与图 4-10 所示的电路具有相同的功能行为。
- 4-4 图 4-48 所示为时钟 (Clock) 和 D 的输入波形，还有一个 D 锁存器以及一个边沿触发式 D 触发器。对图中的锁存器和触发器，仔细画出对应于输入波形的输出波形  $Q_i$ 。假设所有存储元件的传播时延忽略不计，同时假设所有存储元件的初始值为 0。

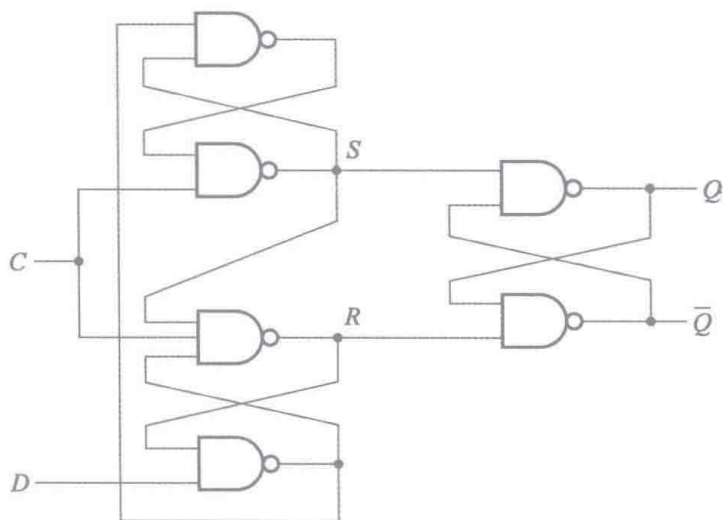


图 4-47 习题 4-3 的电路

280

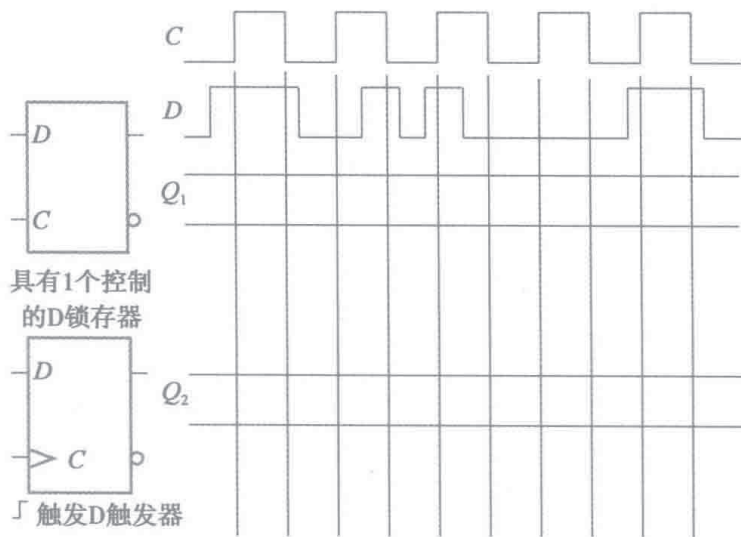


图 4-48 习题 4-4 的波形和存储元件

- 4-5 在一个时序电路中，有两个 D 触发器  $A$  和  $B$ ，一个输入  $Y$  以及一个输出  $Z$ 。这个时序电路可用下面的输入方程描述：

$$D_A = BY + \bar{A}Y, \quad D_B = \bar{Y}, \quad Z = \bar{A}B$$

- 画出电路的逻辑图。
- 作出状态表。
- 画出状态图。
- 这是个 Mealy 型还是 Moore 型机器？

- 4-6 在一个时序电路中，有两个 D 触发器  $A$  和  $B$ ，两个输入  $X$  和  $Y$ ，一个输出  $Z$ 。这个时序电路可以用下面的输入方程描述：

$$D_A = XA + \overline{XY}, \quad D\overline{B} = XB + \overline{X}A, \quad Z = \overline{X}B$$

- (a) 画出电路的逻辑图。  
 (b) 作出状态表。  
 (c) 画出状态图。  
 (d) 这是个 Mealy 型还是 Moore 型机器?

\*4-7 在一个时序电路中, 有三个 D 触发器 A、B 和 C, 一个输入 X。这个时序电路可以用下面的输入方程描述:

$$D_A = (\overline{B}\overline{C} + \overline{B}C)X + (BC + \overline{B}\overline{C})\overline{X}$$

$$D_B = A$$

$$D_C = B$$

- (a) 作出电路的状态表。  
 (b) 分别画出当 X=0 时和 X=1 时的状态图。

4-8 在一个时序电路中, 有一个触发器 Q, 两个输入 X 和 Y, 以及一个输出 S。这个电路包括一个输出为 S 的 D 触发器和一个实现函数

$$D = X \oplus Y \oplus S$$

的组合逻辑, D 触发器的输入为 D。画出这个时序电路的状态表和状态图。

4-9 在图 4-15a 所示的状态图中, 若从状态 00 开始, 加载输入序列 10011011110, 写出所得到的状态转换序列和输出序列。

4-10 画出由表 4-14 所示状态表所描述的时序电路的状态图。

表 4-14 习题 4-10 电路的状态表

当前状态		输入		下一状态		输出
A	B	X	Y	A	B	Z
0	0	0	0	1	0	0
0	0	0	1	1	1	1
0	0	1	0	1	1	0
0	0	1	1	1	1	1
0	1	0	0	0	1	1
0	1	0	1	0	0	0
0	1	1	0	0	0	1
0	1	1	1	0	0	0
1	0	0	0	1	1	1
1	0	0	1	0	1	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	0	0
1	1	0	1	0	1	0
1	1	1	0	1	0	1
1	1	1	1	1	1	1

4-11 在一个时序电路中, 有两个 D 触发器, 一个输入 X 和一个输出 Y, 电路的逻辑图如图 4-49 所示。画出这个电路的状态表和状态图。

4-12 图 4-13 所示为一个时序电路。

- (a) 在电路中增加必要的与 / 或连接逻辑, 使得当信号 Reset=0 时, 将电路异步复位为状态 A=



1、 $B=0$ 。

- (b) 在电路中增加必要的与/或连接逻辑,使得当信号  $\text{Reset}=1$  时,将电路同步复位为状态  $A=0$ 、 $B=0$ 。

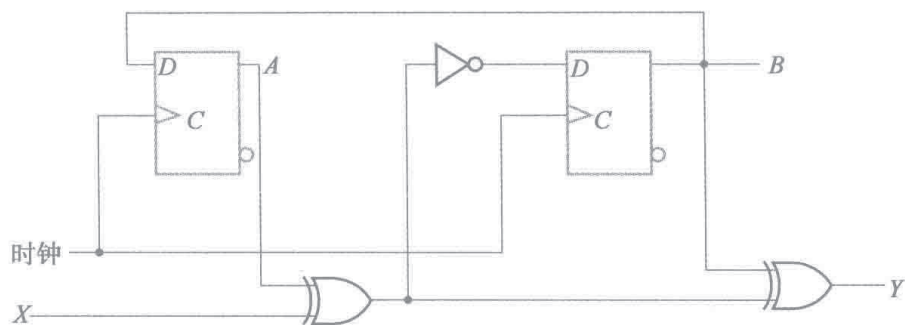


图 4-49 习题 4-11、习题 4-40、习题 4-41、习题 4-49、习题 4-50 和习题 4-59 的电路

- \*4-13 设计一个时序电路。该时序电路包含两个 D 触发器  $A$  和  $B$  及一个输入  $X$ 。当  $X=0$  时,电路保持状态不变;当  $X=1$  时,电路状态转换路线为  $00 \rightarrow 10 \rightarrow 11 \rightarrow 01 \rightarrow 00$ ,并按照这样的序列不断重复。

$X_1X_2/Z$  00/0, 11/0

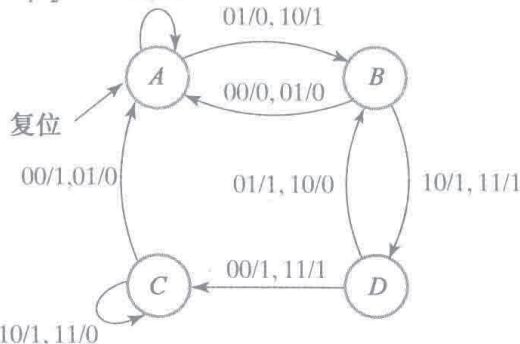


图 4-50 习题 4-14 的状态图

- 4-14 图 4-50 所示为一个时序电路的状态图。

- (a) 给出这个电路的状态表。  
(b) 用 2 位长的代码对电路状态进行赋值,给出编码后的状态表。  
(c) 使用 D 触发器、与非门和反相器,设计出最优的电路。  
(d) 用单热点编码对状态进行赋值,重复问题 (b) 和 (c)。

- 4-15 图 4-51 所示为一个时序电路的状态图。

- (a) 给出这个电路的状态表。  
(b) 用 3 位长的代码对电路的 6 个状态进行赋值,使码字中的一位等于输出值,以便节省成本,给出编码后的状态表。不考虑两个未使用的状态码字所对应的下一状态和输出。  
(c) 使用 D 触发器、与非门和反相器,设计出最优的电路。  
(d) 用单热点编码对状态进行赋值,重复问题 (b) 和 (c)。

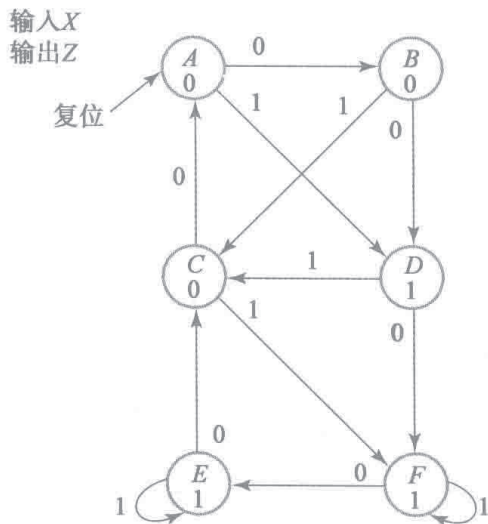


图 4-51 习题 4-15 的状态图

- 4-16 重新设计图 4-52 所示的电路以减少成本。

- (a) 给出这个电路的状态表,并将每个状态编码用一个字母来表示。在原设计中状态 100 和 111 未被使用。  
(b) 检查并合并等价的状态。  
(c) 对状态赋值,使得输出是某一个状态变量。  
(d) 计算原电路和新电路的门输入成本,假设一个 D 触发器的门输入成本为 14。新电路的成本减少了吗?

- 4-17 行李锁的时序电路有 10 个按钮,分别标记为 0、1、2、3、4、5、6、7、8、9。从 0 到 9 的每个按钮分别在相应的  $X_i$  ( $i=0, 1, \dots, 9$ ) 上产生 1,而在其他的变量  $X_j$  ( $j \neq i$ ) 上产生





否则如果  $E=0$ , 那么输出保持为 1。

- (a) 画出这个电路的 Moore 型状态图。
- (b) 给出电路的状态表并进行状态赋值。
- (c) 用 D 触发器和逻辑门设计该电路。该电路要包括复位信号以便将电路设置为适当的初始状态。在初始状态时, 会检查  $E$  的值以确定是否产生常数 1 序列。

4-21 假设在另一种通信网络协议中使用序列 01111110, 重做习题 4-20。



- +4-22 习题 4-21 中的序列在通信网络中用于标记消息的开始。这个标记必须是唯一的, 因此在消息序列的其他任何位置最多只能出现 5 个连续的 1。显然对于通常的消息而言, 这一要求是不实际的, 为解决这一问题, 可以使用零插入技术。包含多于 5 个连续 1 的消息从输入端  $X$  进入零插入时序电路, 该电路有两个输出  $Z$  和  $S$ 。当序列中的第 5 个 1 出现在输入端  $X$  时, 在输出端  $Z$  的输出流中插入一个 0 值, 同时输出  $S=1$ , 表明电路启动了零插入技术, 此时信号输入必须暂停, 也就是在一个时钟周期内不能输入新的信号, 因为在输出序列中插入 0 使得输出序列比没有停顿的输入序列长。下面的例子序列解释了零插入技术:

$X$  上没有任何停顿的序列: 0111110011111100001011110101

$X$  上带有停顿的序列: 01111100111111100001011110101

$Z$  上的序列: 0111110001111101100001011110101

$S$  上的序列: 000000100000001000000000000000

- (a) 给出电路的状态图。
- (b) 给出电路的状态表并进行状态赋值。
- (c) 使用 D 触发器和逻辑门实现这个电路。



- 4-23 在许多通信和网络系统中, 在通信线路上传输的信号采用不归零 (NRZ) 格式。USB 使用一种特殊的版本, 称为翻转不归零 (NRZI) 制。现在要设计一个电路, 把任何由 0 和 1 组成的消息序列转换为 NRZI 格式的消息序列。这个电路的映射方式为:

(a) 如果消息位为 0, 则 NRZI 消息发生翻转, 从 1 变为 0 或从 0 变为 1, 这取决于当前的 NRZI 消息值。

(b) 如果消息位为 1, 则 NRZI 消息保持为 0 或 1, 这取决于当前的 NRZI 消息值。

下面的例子说明了这种转换方式, 假设 NRZI 消息的初始值为 1:

消息: 10001110011010

NRZI 消息: 10100001000110

- (a) 画出这个电路的 Mealy 型状态图。
- (b) 作出这个电路的状态表并进行状态赋值。
- (c) 使用 D 触发器和逻辑门实现这个电路。



- +4-24 重做习题 4-23, 设计一个时序电路将 NRZI 消息转换为普通消息。这个电路的映射方式为:

(a) 如果 NRZI 消息中的相邻两位出现从 0 到 1 或从 1 到 0 的变化, 那么消息位为 0。

(b) 如果 NRZI 消息中的相邻两位没有发生变化, 则消息位为 1。

- 4-25 被称为请求信号 ( $R$ ) 和应答信号 ( $A$ ) 的一对信号用于协调 CPU 和 I/O 系统之间的传输, 这些信号之间的交互通常称为“握手”。在进行通信时, 这些信号与时钟同步, 它们按照如图 4-53 所示的顺序进行变化。设计一个握手检测器来验证变化发生的顺序。检测器有输入  $R$  和  $A$ 、异步复位信号 RESET, 以及错误输出  $E$ 。如果握手的变化是按顺序进行的, 则  $E=0$ 。如果变化没有按顺序进行, 则  $E$  变为 1, 并且在 CPU 加载异步复位信号 (RESET=1) 之前,  $E$  都将保持为 1。

- (a) 画出握手检测器的状态图。

286

287

(b) 作出握手检测器的状态表。

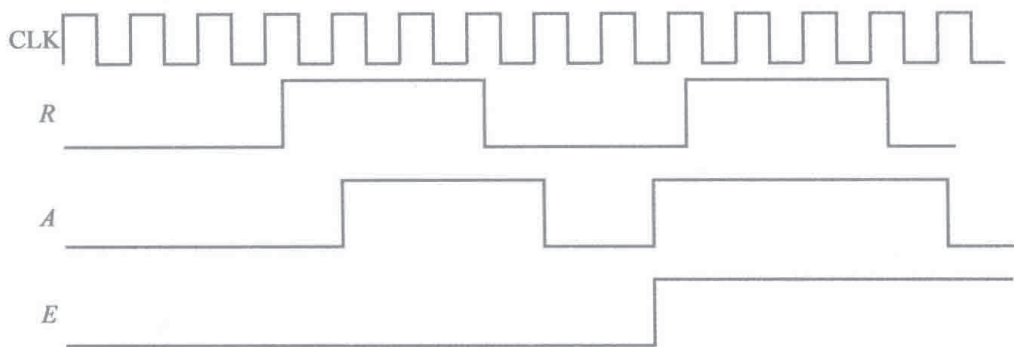


图 4-53 习题 4-25 的信号

- 4-26 设计一个串行的以 1 开头的序列检测器。一个任意长度的二进制整数以高位在前、低位在后的方式输入到这个串行的以 1 开头的序列检测器的输入端  $X$ 。输入端  $X$  出现一位输入时，输出端  $Z$  在同一时钟周期内产生相应的输出位。只要加载到  $X$  的位为 0，则  $Z=0$ 。当第一个 1 出现在输入端  $X$  时， $Z=1$ ，而对第一个 1 出现之后  $X$  输入端出现的任何值，都有  $Z=0$ 。为了表明一个序列已经输入完毕，同时电路将被初始化以便接收下一个序列，输入  $Y$  要持续一个时钟周期保持为 1，其他情况  $Y$  为 0。

(a) 画出这个串行的以 1 开头的序列检测器的状态图。

(b) 作出这个串行的以 1 开头的序列检测器的状态表。

- \*4-27 在一个时序电路中有两个触发器  $A$  和  $B$ ，一个输入  $X$  和一个输出  $Y$ ，其状态图如图 4-54 所示。用 D 触发器和单热点状态赋值方法设计这个电路。

4-28 用 D 触发器和格雷码赋值方法重做习题 4-27。

- +4-29 一个 3 位扭环形计数器 (twisted ring counter) 的状态表如表 4-15 所示。这个电路没有输入，其输出为触发器的非反相输出。由于没有输入，每当出现一个时钟脉冲时，它便简单地从一个状态转换到另一个状态。该电路有一个异步复位端，用来将其初始化为状态 000。

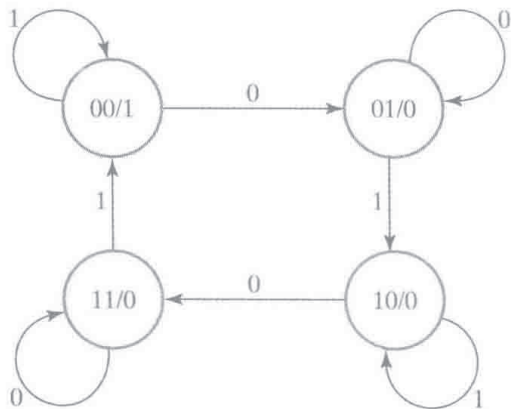


图 4-54 习题 4-27 的状态图

表 4-15 习题 4-29 的状态表

当前状态	下一状态
$ABC$	$ABC$
000	100
100	110
110	111
111	011
011	001
001	000

- (a) 用 D 触发器设计这个电路。假设未指定的下一状态是不必考虑的情况。
- (b) 在电路中增加必要的逻辑，使得在上电启动主复位时，能够使电路初始化为状态 000。
- (c) 4.5 节中关于“无效状态的设计”讨论了当电路意外进入无效状态时的 3 种处理技术。如果你所设计的电路用于儿童玩具，在处理 (a) 和 (b) 问题时，你将使用这 3 种技术中的哪一种？说明你的理由。
- (d) 根据你在 (c) 中所做的决定，如果有必要的话请重新设计这个电路。
- (e) 如果这个电路要应用到商用航空的引擎控制系统中，重做 (c)。说明你的理由。



(f) 根据你在 (e) 中所作的决定, 重做 (d)。

- 4-30 对你在习题 4-14 所作的设计, 用自动逻辑模拟方法进行验证。模拟所用的输入序列应该包括图 4-50 中所有的状态转换, 模拟输出应包括输入  $X$  和状态变量  $A$ 、 $B$  和输出  $Z$ 。
- \*4-31 针对状态表 4-14 所描述的电路, 产生一个验证序列。为了减少模拟序列的长度, 假设模拟器可以操控输入  $X$ , 并且可以在任何可能的情况下使用  $X$ 。假设复位输入可以将状态初始化为  $A=0$ 、 $B=0$ , 状态图中的所有状态转换都必须考虑。
- 4-32 设计表 4-14 所描述的电路, 并使用习题 4-31 中的序列 (你自己得出的或本书相关网站上给出的) 对你的设计用自动逻辑模拟方法进行验证。
- 4-33 一个时序电路的状态表如表 4-16 所示。

表 4-16 习题 4-33 的状态表

当前状态		输入	下一状态		输出
$A$	$B$	$Y$	$A$	$B$	$Z$
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	1	0	0
0	1	1	1	1	0
1	0	0	1	0	0
1	0	1	0	0	0
1	1	0	0	0	1
1	1	1	1	0	1

(a) 画出这个电路的状态图。

(b) 用 D 触发器实现该电路, 并使得每个触发器的输入函数最小。假设复位是异步的, 而且是低电平有效 ( $\text{RESET}=0$ ), 电路初始状态为  $A=0$ 、 $B=0$ 。

- 4-34 设计一种新型的正边沿触发的触发器——LH 触发器。它有一个时钟  $C$ 、一个数据输入  $D$  和一个装载输入  $L$ 。在  $C$  的正边沿如果  $L=1$ , 那么  $D$  上的数据存入触发器; 在  $C$  的正边沿如果  $L=0$ , 那么触发器当前存储的值保持不变。仅使用  $\overline{S}\overline{R}$  锁存器、与非门和反相器设计该触发器。
- 4-35 给出与图 4-55 所示状态图等价的状态机图。要尽可能地减小转换条件的复杂性, 尝试通过把 Mealy 型输出转变为 Moore 型输出来使输出变成无条件的。对状态机图进行状态赋值, 并利用 D 触发器、与门、或门及反相器实现相应的时序电路。

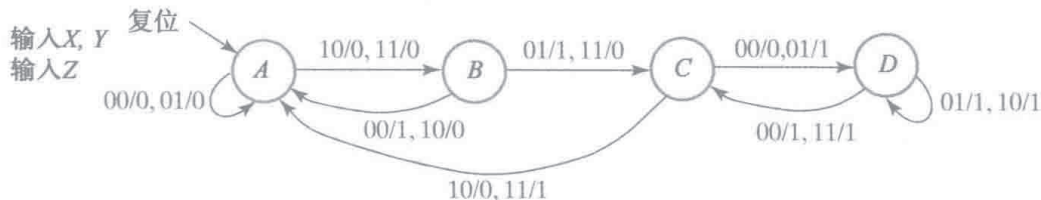




图 4-55 习题 4-35 的状态图



- 4-36 根据习题 4-35 给出的状态机图, 利用 D 触发器、与门、或门及反相器设计相应的时序电路, 要求使用单热点状态赋值。
- 4-37 (a) 验证图 4-27 所示状态机图中的转换满足状态图的两个转换条件。(b) 对于图 4-28 所示的状态机图, 重做问题 (a)。
- \*4-38 画出下面描述的电子售货机的状态机图。售货机销售糖果, 每颗糖果卖价为 25¢。售货机仅接收  $N$  (1nickels=5¢)、 $D$  (1dimes=10¢) 和  $Q$  (1quarters=25¢)。当连续投入的硬币



之和为 25¢ 或更多时,  $DJ=1$ , 售货机输出一颗糖果同时返回到初始状态, 售货机不找零。对于其他的所有状态  $DJ=0$ 。如果投入的硬币少于 25¢, 当按下 CR (Coin Return) 按钮时,  $RC=1$ , 投入的硬币通过硬币退还出口被退回, 然后售货机返回到初始状态。在其他的所有状态下  $RC=0$ 。请使用 Moore 型输出进行设计。

-  4-39 画出下面描述的电子售货机的状态机图。售货机销售汽水, 每瓶卖价为 1.50\$。售货机仅接收  $D$  (1 美元纸币) 和  $Q$  (1 quarters = 25¢)。如果投入的钱多于 1.50\$, 例如两张 1\$ 纸币, 售货机通过硬币出口退还零钱 (2 个  $Q$ )。当已经付了 1.50\$ 时, 机器亮起一个 LED, 表明可以选取里面的一种汽水。通过按钮可选择  $C$  (cola)、 $L$  (Lemon soda)、 $O$  (Orange soda) 和  $R$  (Root Beer)。当按下一个按钮时, 售货机送出被选择的汽水同时返回到初始状态。售货机还有另外一个特点, 即当已经投入两个 quarter 时会亮起一个 LED 灯以提醒用户, 如果再投入 1\$ 纸币则不会找回零钱。
- (a) 根据描述画出汽水售货机的状态机图。
- (b) 题目给出的规格说明对于用户而言并不是很方便。请考虑用户在使用机器过程中可能遇到的所有情况, 重新做这个题目。


 下面习题所涉及的所有文件, 在本书的相关网站上均有 ASCII 格式的版本可供编辑和仿真。对于一些需要进行模拟的问题或子问题, 要使用 VHDL 或 Verilog 编译器 / 模拟器。但对于那些没有使用编译器或模拟器的问题来说, 直接描述也是可以的。

- 4-40 写出习题 4-11 中所给电路的门级 VHDL 结构描述。对于 D 触发器使用图 4-29 所示的 VHDL 模型, 对于逻辑门元件要在库文件 `lcdf_vhdl` 中使用包 `func_prims`。
- 4-41 写出习题 4-11 中所给电路的 VHDL 行为级描述, 使用一个进程来描述状态图。
- \*4-42 尽管本章已经介绍了用 VHDL 进程来描述时序电路, 但组合电路也可以用进程来描述。用 VHDL 描述图 3-25 所示的多路复用器, 就像 3.7 节那样在进程中使用 case 语句而不是并发赋值语句。
- 4-43 在 VHDL 进程中使用 if-then-else 语句, 重做习题 4-42。
- +4-44 针对图 4-19d 所示状态图描述的时序电路, 写出 VHDL 描述。包含一个异步 RESET 信号, 可把电路初始化为 Init 状态。编译你的程序, 加载一个输入序列, 使得状态图中的每一个状态转换至少经历一次, 并通过与状态图进行比较, 验证状态序列和输出序列的正确性。
- 4-45 用 VHDL 描述习题 4-14 所定义的电路。
- 4-46 用 VHDL 描述习题 4-15 所定义的电路。
-  4-47 针对从例 4-10 中得到的批处理混合系统的状态机图, 给出 VHDL 描述。
-  4-48 针对从习题 4-38 中得到的糖果售货机的状态机图, 给出 VHDL 描述。你可以通过解习题 4-38 得到状态机图, 或者使用本书配套网站上给出的答案。
- 4-49 写出习题 4-11 中所给电路的门级 Verilog 结构描述。对于 D 触发器使用图 4-33 所示的 Verilog 模型。
- 4-50 写出习题 4-11 中所给电路的 Verilog 行为级描述, 使用一个进程来描述状态图。
- 4-51 尽管本章已经介绍了用 Verilog 进程来描述时序电路, 但组合电路也可以用进程来描述。用 Verilog 描述图 3-25 所示的多路复用器, 就像 3.7 节那样在进程中使用 case 语句而不是连续赋值语句。
- \*4-52 在 Verilog 进程中使用 if-else 语句, 重做习题 4-51。
- +4-53 针对图 4-19d 所示状态图描述的时序电路, 写出 Verilog 描述。包含一个异步 RESET 信号, 可把电路初始化为 Init 状态。编译你的程序, 加载一个输入序列, 使得状态图中的每一个状态转换至少经历一次, 并通过与状态图进行比较, 验证状态序列和输出序列的正确性。

4-54 用 Verilog 描述习题 4-14 所定义的电路。

4-55 用 Verilog 描述习题 4-15 所定义的电路。

 4-56 针对从例 4-10 中得到的批处理混合系统的状态机图，给出 Verilog 描述。

 4-57 针对从习题 4-38 中得到的糖果售货机的状态机图，给出 Verilog 描述。你可以通过解习题 4-38 得到状态机图，或者使用本书配套网站上给出的答案，并在参数语句中使用单热点状态赋值。

4-58 图 4-56 所示的是一组加载到两个 D 触发器的波形，这些加载到触发器的波形带有定时参数。

(a) 对于触发器 1 (原书有误——译者注)，给出信号 D1 出现定时冲突的时刻。

(b) 对于触发器 2，指出信号 D2 何时出现了定时冲突。

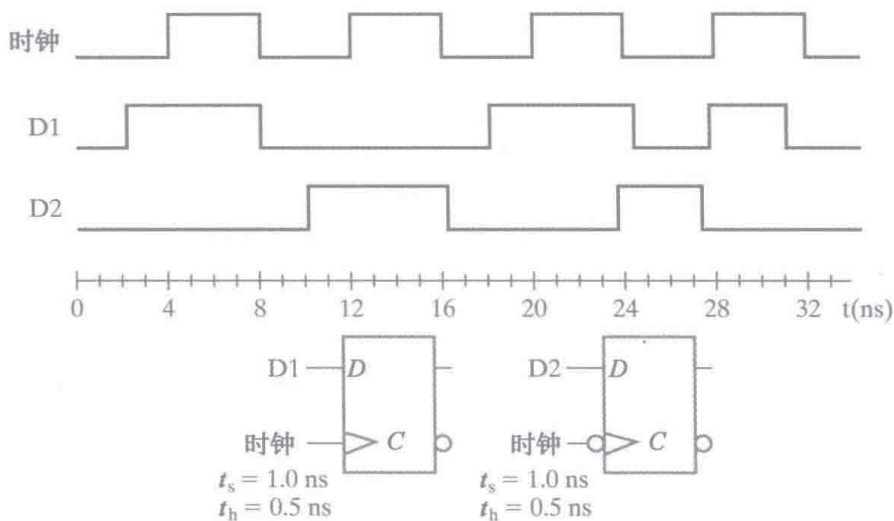


图 4-56 习题 4-58 的电路

\*4-59 一个时序电路如图 4-49 所示，其中门和触发器的定时参数为

反相器:  $t_{pd} = 0.01$  ns

异或门:  $t_{pd} = 0.04$  ns

触发器:  $t_{pd} = 0.08$  ns,  $t_s = 0.02$  ns,  $t_h = 0.01$  ns

(a) 给出从外部电路输入，经过一系列门，直接到达外部电路输出的最长路径的延迟。

293

(b) 给出从外部电路输入，到达时钟上升沿的最长路径的延迟。

(c) 给出从时钟上升沿，到达输出的最长路径的延迟。

(d) 给出从时钟上升沿，到达时钟上升沿的最长路径的延迟。

(e) 确定电路的最高工作频率，以兆赫兹 (MHz) 为单位进行计算。

4-60 重做习题 4-59。假设电路由两个如图 4-49 所示的电路构成，第二个电路的输入 X 由第一个电路的输出 Y 所驱动。

4-61 用 HDL 对习题 4-59 中给出的电路进行门级描述，电路中的每个元件均有延迟。当工作频率大于你在解答习题 4-59 时所得到的最高频率时，电路会出现错误，请对这一情况给予说明。

294