

Q_2 端输出的串行输出信号就应当是所要求的序列信号。

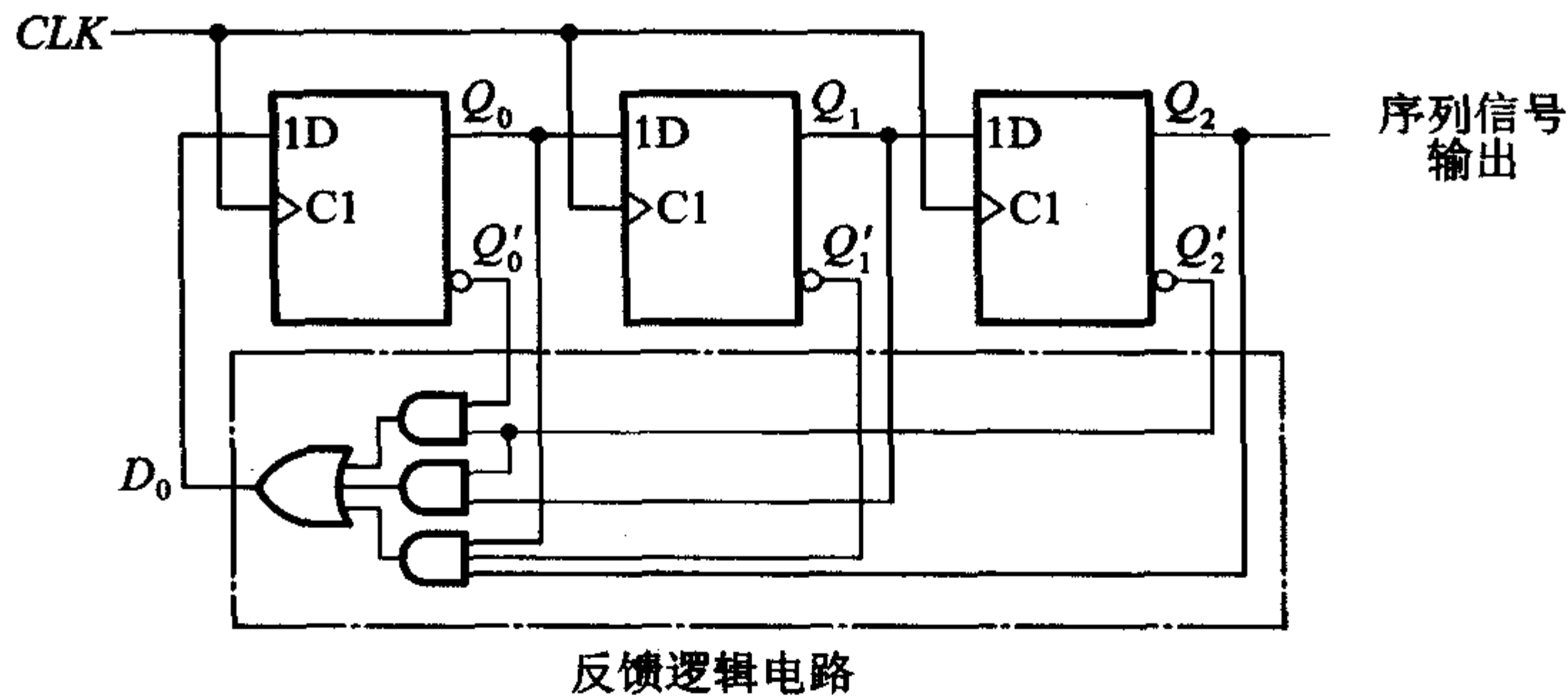


图 6.3.56 用移位寄存器构成的序列信号发生器

根据要求产生的序列信号,即可列出移位寄存器应具有的状态转换表,如表 6.3.9 所示。再从状态转换的要求出发,得到对移位寄存器输入端 D_0 取值的要求,如表 6.3.9 中所示。表中也同时给出了 D_0 与 Q_2 、 Q_1 、 Q_0 之间的函数关系。利用图 6.3.57 所示的卡诺图将 D_0 的函数式化简,得到

$$D_0 = Q_2 Q_1' Q_0 + Q_2' Q_1 + Q_2' Q_0' \quad (6.3.16)$$

图 6.3.56 中的反馈逻辑电路就是按式(6.3.16)接成的。

表 6.3.9 图 6.3.56 电路的状态转换表

CLK 顺序	Q_2	Q_1	Q_0	D_0
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	1	0	1	1
4	0	1	1	1
5	1	1	1	0
6	1	1	0	0
7	1	0	0	0
8	0	0	0	1

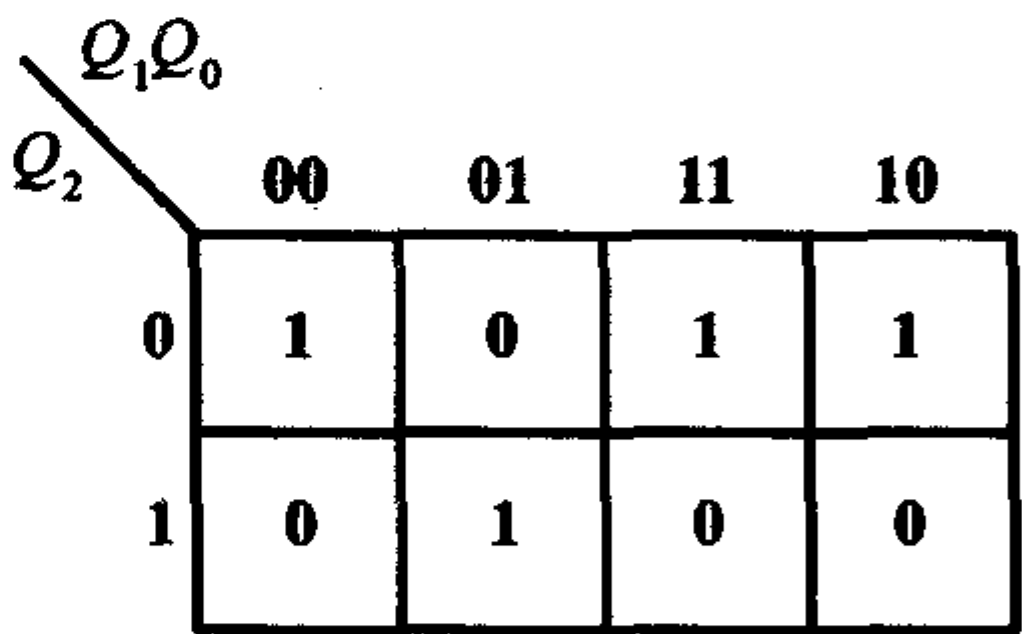


图 6.3.57 图 6.3.56 中 D_0 的卡诺图

6.4 时序逻辑电路的设计方法

6.4.1 同步时序逻辑电路的设计方法

在设计时序逻辑电路时,要求设计者根据给出的具体逻辑问题,求出实现这

一逻辑功能的逻辑电路。所得到的设计结果应力求简单。在这一小节里我们首先讨论简单时序电路的设计。这里所说的简单时序电路,是指用一组状态方程、驱动方程和输出方程就能完全描述其逻辑功能的时序电路。

当选用小规模集成电路做设计时,电路最简的标准是所用的触发器和门电路的数目最少,而且触发器和门电路的输入端数目也最少。而当使用中、大规模集成电路时,电路最简的标准则是使用的集成电路数目最少,种类最少,而且互相间的连线也最少。

设计同步时序逻辑电路时,一般按如下步骤进行:

一、逻辑抽象,得出电路的状态转换图或状态转换表

就是将要求实现的时序逻辑功能表示为时序逻辑函数,可以用状态转换表的形式,也可以用状态转换图或状态机流程图的形式。这就需要:

(1) 分析给定的逻辑问题,确定输入变量、输出变量以及电路的状态数。通常都是取原因(或条件)作为输入逻辑变量,取结果作输出逻辑变量。

(2) 定义输入、输出逻辑状态和每个电路状态的含意,并将电路状态顺序编号。

(3) 按照题意列出电路的状态转换表或画出电路的状态转换图。

这样,就把给定的逻辑问题抽象为一个时序逻辑函数了。

二、状态化简

若两个电路状态在相同的输入下有相同的输出,并且转换到同样一个次态去,则称这两个状态为等价状态。显然,等价状态是重复的,可以合并为一个。电路的状态数越少,设计出来的电路就越简单。

状态化简的目的就在于将等价状态合并,以求得最简的状态转换图。

三、状态分配

状态分配又称状态编码。

时序逻辑电路的状态是用触发器状态的不同组合来表示的。首先,需要确定触发器的数目 n 。因为 n 个触发器共有 2^n 种状态组合,所以为获得时序电路所需的 M 个状态,必须取

$$2^{n-1} < M \leq 2^n \quad (6.4.1)$$

其次,要给每个电路状态规定对应的触发器状态组合。每组触发器的状态组合都是一组二值代码,因而又将这项工作称为状态编码。在 $M < 2^n$ 的情况下,从 2^n 个状态中取 M 个状态的组合可以有多种不同的方案,而每个方案中 M 个状态的排列顺序又有许多种。如果编码方案选择得当,设计结果可以很简单。反之,编码方案选得不好,设计出来的电路就会复杂得多,这里面有一定的技巧。

此外,为便于记忆和识别,一般选用的状态编码和它们的排列顺序都遵循一定的规律。

四、选定触发器的类型, 求出电路的状态方程、驱动方程和输出方程

因为不同逻辑功能的触发器驱动方式不同, 所以用不同类型触发器设计出的电路也不一样。为此, 在设计具体的电路前必须选定触发器的类型。选择触发器类型时应考虑到器件的供应情况, 并应力求减少系统中使用的触发器种类。

根据状态转换图(或状态转换表)和选定的状态编码、触发器的类型, 就可以写出电路的状态方程、驱动方程和输出方程了。

五、根据得到的方程式画出逻辑图

六、检查设计的电路能否自启动

如果电路不能自启动, 则需采取措施加以解决。一种解决办法是在电路开始工作时通过预置数将电路的状态置成有效状态循环中的某一种。另一种解决方法是通过修改逻辑设计加以解决。具体的作法将在下一小节中介绍。

至此, 逻辑设计工作已经完成。图 6.4.1 用方框图表示了上述设计工作的大致过程。不难看出, 这一过程和分析时序电路的过程正好是相反的。

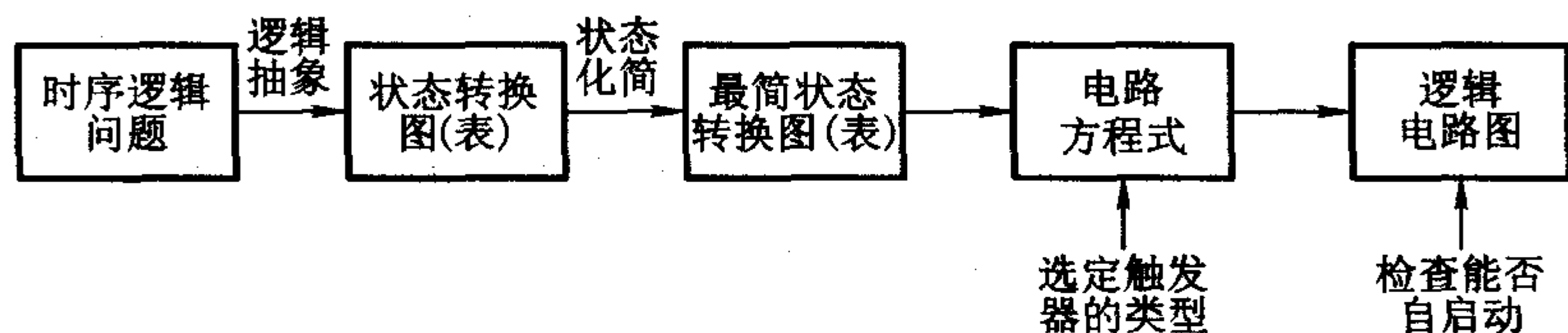


图 6.4.1 同步时序逻辑电路的设计过程

下面通过三个不同类型的具体例子进一步深入说明上述设计方法。

【例 6.4.1】 试设计一个带有进位输出端的十三进制计数器。

解: 首先进行逻辑抽象。

因为计数器的工作特点是在时钟信号操作下自动地依次从一个状态转为下一个状态, 所以它没有输入逻辑变量, 只有进位输出信号。因此, 计数器是属于穆尔型的一种简单时序电路。

取进位信号为输出逻辑变量 C , 同时规定有进位输出时 $C = 1$, 无进位输出时 $C = 0$ 。

十三进制计数器应该有十三个有效状态, 若分别用 $S_0, S_1, \dots, S_{12}$ 表示, 则按题意可以画出如图 6.4.2 所示的电路状态转换图。

因为十三进制计数器必须用 13 个不同的状态表示已经输入的脉冲数, 所以状态转

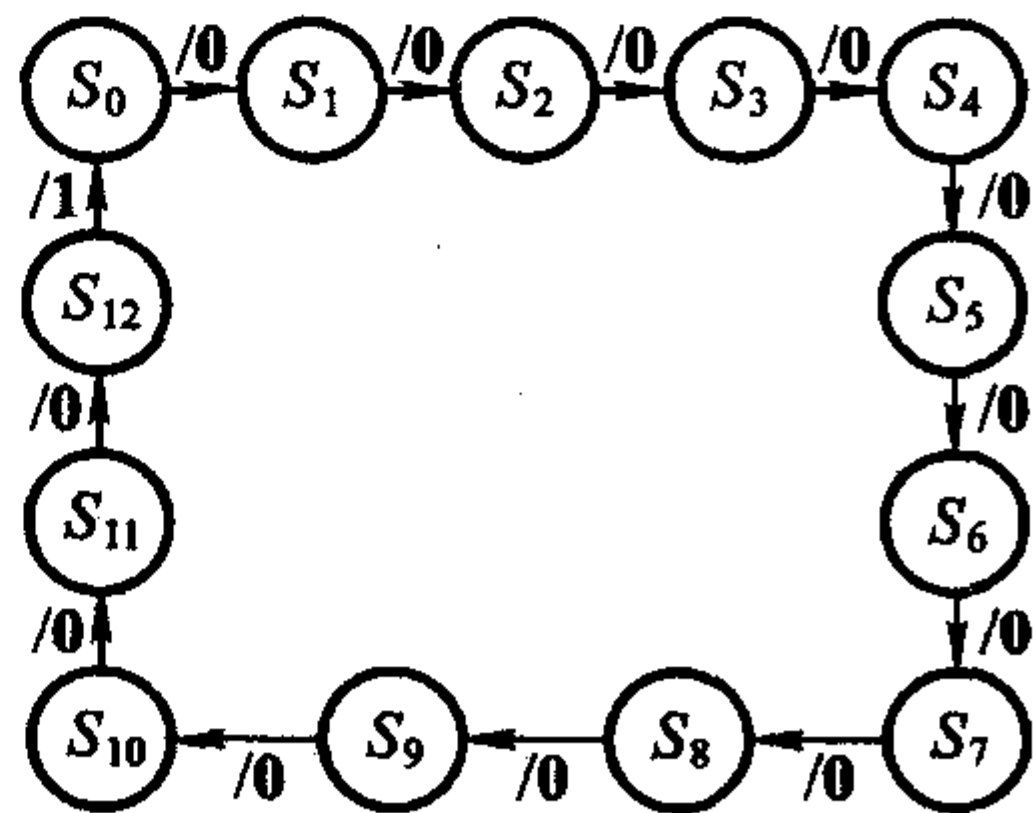


图 6.4.2 例 6.4.1 的状态转换图

换图已不能再化简。

根据式(6.4.1)知,现要求 $M=13$,故应取触发器位数 $n=4$,因为

$$2^3 < 13 < 2^4$$

假如对状态分配无特殊要求,可以取自然二进制数的 **0000 ~ 1100** 作为 $S_0 \sim S_{12}$ 的编码,于是得到了表 6.4.1 中的状态编码。

表 6.4.1 例 6.4.1 电路的状态转换表

状态变化顺序	状态编码				进位输出 C	等效 十进制数
	Q_3	Q_2	Q_1	Q_0		
S_0	0	0	0	0	0	0
S_1	0	0	0	1	0	1
S_2	0	0	1	0	0	2
S_3	0	0	1	1	0	3
S_4	0	1	0	0	0	4
S_5	0	1	0	1	0	5
S_6	0	1	1	0	0	6
S_7	0	1	1	1	0	7
S_8	1	0	0	0	0	8
S_9	1	0	0	1	0	9
S_{10}	1	0	1	0	0	10
S_{11}	1	0	1	1	0	11
S_{12}	1	1	0	0	1	12
S_0	0	0	0	0	0	0

由于电路的次态 $Q_3^* Q_2^* Q_1^* Q_0^*$ 和进位输出 C 唯一地取决于电路现态 $Q_3 Q_2 Q_1 Q_0$ 的取值,故可根据表 6.4.1 画出表示次态逻辑函数和进位输出函数的卡诺图,如图 6.4.3 所示。因为计数器正常工作时不会出现 **1101**、**1110** 和 **1111** 三个状态,所以可将 $Q_3 Q_2 Q_1' Q_0$ 、 $Q_3 Q_2 Q_1 Q_0'$ 和 $Q_3 Q_2 Q_1 Q_0$ 三个最小项作约束项处理,在卡诺图中用 $\times$ 表示。

为清晰起见,可将图 6.4.3 所示的卡诺图分解为图 6.4.4 所示的五个卡诺图,分别表示 Q_3^* 、 Q_2^* 、 Q_1^* 、 Q_0^* 和 C 这五个逻辑函数。从这些卡诺图得到电路的状态方程为

$$\begin{cases} Q_3^* = Q_3 Q_2' + Q_2 Q_1 Q_0 \\ Q_2^* = Q_3' Q_2 Q_1' + Q_3' Q_2 Q_0' + Q_2' Q_1 Q_0 \\ Q_1^* = Q_1' Q_0 + Q_1 Q_0' \\ Q_0^* = Q_3' Q_0' + Q_2' Q_0' \end{cases} \quad (6.4.2)$$

输出方程为

$Q_3Q_2 \backslash Q_1Q_0$		00	01	11	10
		00	01	11	10
00	00	0001/0	0010/0	0100/0	0011/0
01	01	0101/0	0110/0	1000/0	0111/0
11	11	0000/1	××××/×	××××/×	××××/×
10	10	1001/0	1010/0	1100/0	1011/0

图 6.4.3 例 6.4.1 电路次态/输出
($Q_3^* Q_2^* Q_1^* Q_0^* / C$) 的卡诺图

$Q_3Q_2 \backslash Q_1Q_0$		00	01	11	10
		00	01	11	10
00	00	0	0	0	0
01	01	0	0	1	0
11	11	0	×	×	×
10	10	1	1	1	1

(a) Q_3^*

$Q_3Q_2 \backslash Q_1Q_0$		00	01	11	10
		00	01	11	10
00	00	0	0	1	0
01	01	1	1	0	1
11	11	0	×	×	×
10	10	0	0	1	0

(b) Q_2^*

$Q_3Q_2 \backslash Q_1Q_0$		00	01	11	10
		00	01	11	10
00	00	0	1	0	1
01	01	0	1	0	1
11	11	0	×	×	×
10	10	0	1	0	1

(c) Q_1^*

$Q_3Q_2 \backslash Q_1Q_0$		00	01	11	10
		00	01	11	10
00	00	1	0	0	1
01	01	1	0	0	1
11	11	0	×	×	×
10	10	1	0	0	1

(d) Q_0^*

$Q_3Q_2 \backslash Q_1Q_0$		00	01	11	10
		00	01	11	10
00	00	0	0	0	0
01	01	0	0	0	0
11	11	1	×	×	×
10	10	0	0	0	0

(e) C

图 6.4.4 图 6.4.3 卡诺图的分解

$$C = Q_3 Q_2 \quad (6.4.3)$$

如果选用 JK 触发器组成这个电路,则应将式(6.4.2)的状态方程变换成 JK 触发器特性方程的标准形式,即 $Q^* = J Q' + K' Q$,然后就可以找出驱动方程了。为此,将式(6.4.2)改写为

$$\begin{cases} Q_3^* = Q_3 Q_2' + Q_2 Q_1 Q_0 (Q_3 + Q_3') = (Q_2 Q_1 Q_0) Q_3' + Q_2' Q_3 \\ Q_2^* = (Q_0 Q_1) Q_2' + (Q_3' (Q_1 Q_0)') Q_2 \\ Q_1^* = Q_0 Q_1' + Q_0' Q_1 \\ Q_0^* = (Q_3' + Q_2') Q_0' + 1' \cdot Q_0 = (Q_3 Q_2)' Q_0' + 1' Q_0 \end{cases} \quad (6.4.4)$$

在变换 Q_3^* 的逻辑式时,删去了约束项 $Q_3 Q_2 Q_1 Q_0$ 。将式(6.4.4)中的各逻辑式与 JK 触发器的特性方程对照,则各个触发器的驱动方程应为

$$\begin{cases} J_3 = Q_2 Q_1 Q_0, & K_3 = Q_2 \\ J_2 = Q_1 Q_0, & K_2 = (Q_3' (Q_1 Q_0)')' \\ J_1 = Q_0, & K_1 = Q_0 \\ J_0 = (Q_3 Q_2)', & K_0 = 1 \end{cases} \quad (6.4.5)$$

根据式(6.4.3)和式(6.4.5)画得计数器的逻辑图如图 6.4.5 所示。

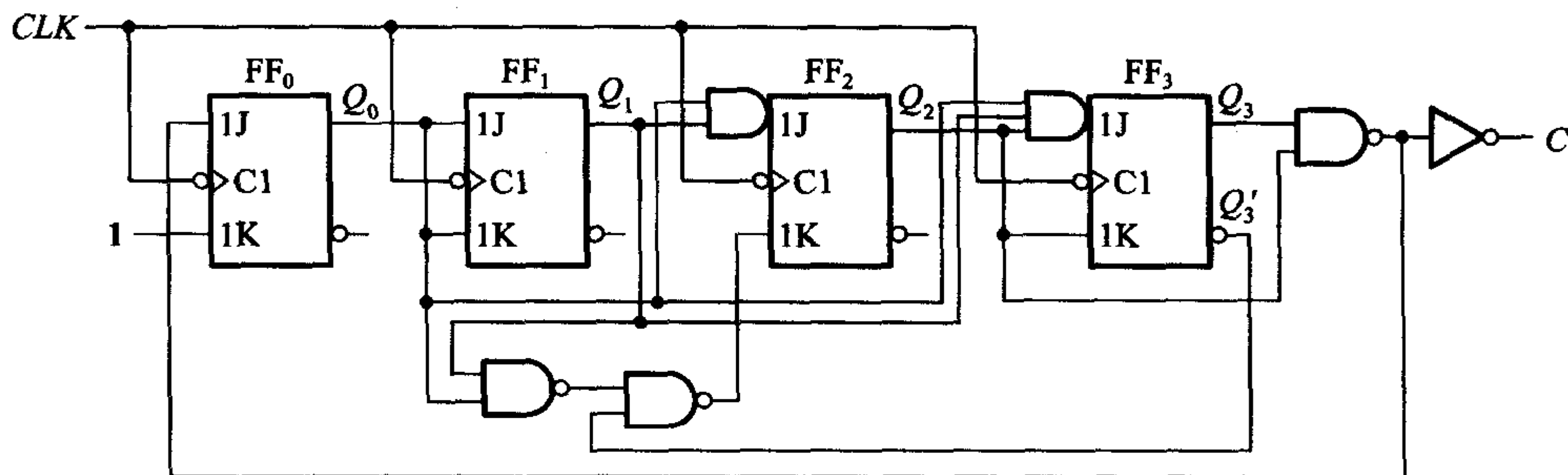


图 6.4.5 例 6.4.1 的同步十三进制计数器

为验证电路的逻辑功能是否正确,可将 **0000** 作为初始状态代入式(6.4.4)的状态方程依次计算次态值,所得结果应与表 6.4.1 中的状态转换表相同。

最后还应检查电路能否自启动。将 3 个无效状态 **1101**、**1110** 和 **1111** 分别代入式(6.4.4)中计算,所得次态分别为 **0010**、**0010** 和 **0000**,故电路能自启动。

图 6.4.6 是图 6.4.5 电路完整的状态转换图。

【例 6.4.2】 设计一个串行数据检测器,对它的要求是:连续输入 3 个或 3 个以上的 1 时输出为 1,其他输入情况下输出为 0。

解: 首先进行逻辑抽象,画出状态转换图。

取输入数据为输入变量,用 X 表示;取检测结果为输出变量,以 Y 表示。

设电路在没有输入 1 以前的状态为 S_0 ,输入一个 1 以后的状态为 S_1 ,连续输入两个 1 以后的状态为 S_2 ,连续输入 3 个或 3 个以上 1 以后的状态为 S_3 。若

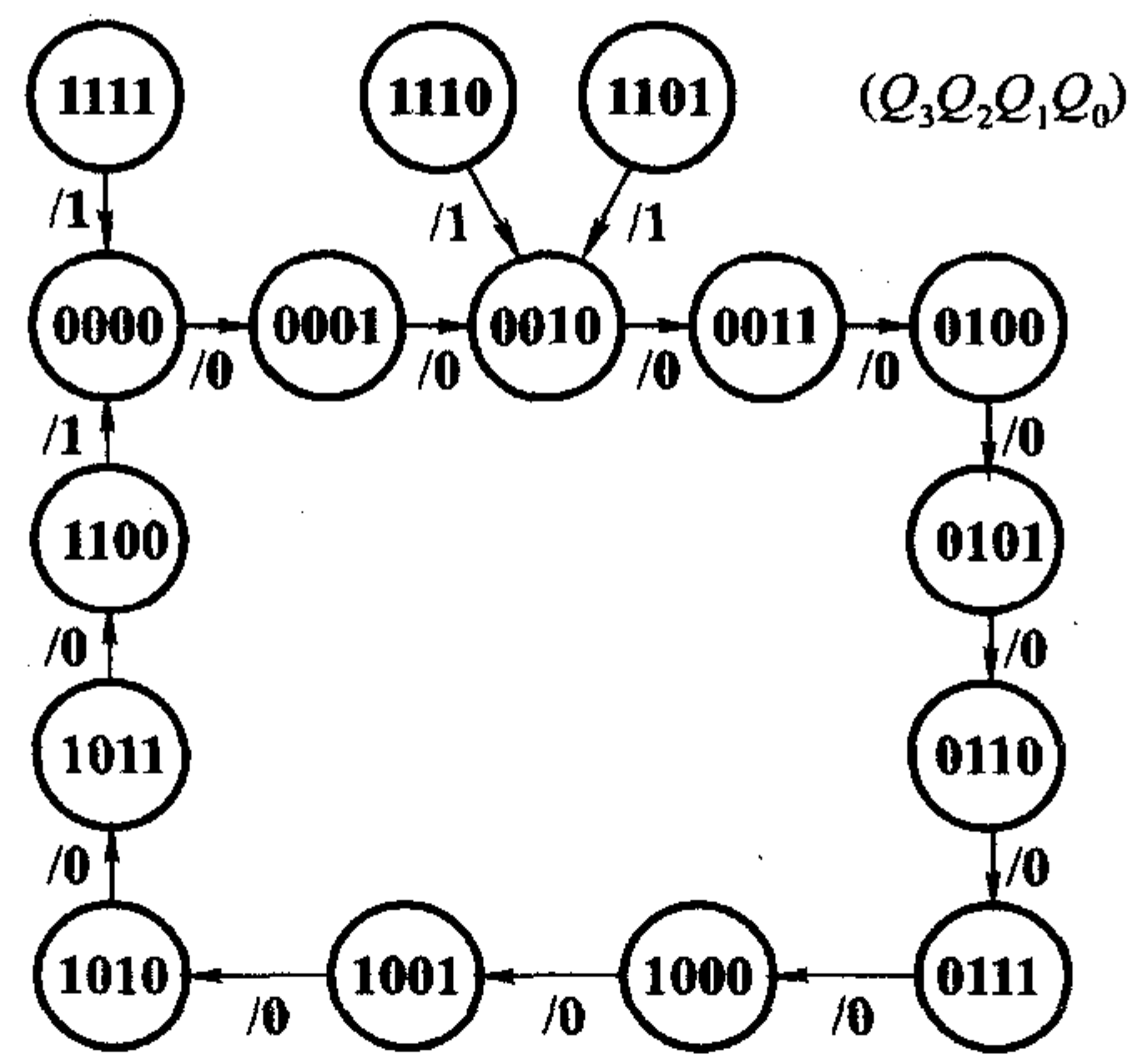


图 6.4.6 图 6.4.5 电路的状态转换图

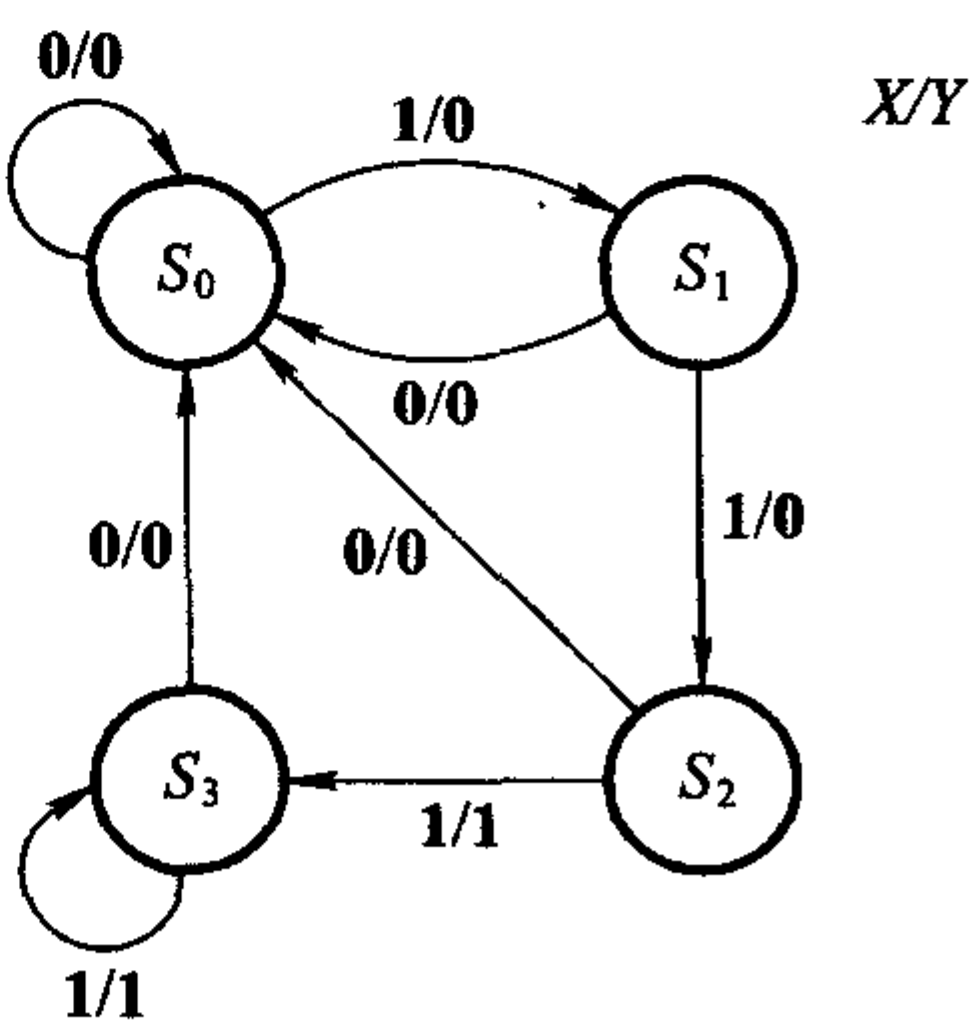
以 S 表示电路的现态,以 S^* 表示电路的次态,依据设计要求即可得到表 6.4.2 所示的状态转换表和图 6.4.7 所示的状态转换图。

表 6.4.2 例 6.4.2 的状态转换表

S^*/Y X	S	S_0	S_1	S_2	S_3
0		$S_0/0$	$S_0/0$	$S_0/0$	$S_0/0$
1		$S_1/0$	$S_2/0$	$S_3/1$	$S_3/1$

然后进行状态化简。比较一下 S_2 和 S_3 这两个状态便可发现,它们在同样的输入下有同样的输出,而且转换后得到同样的次态。因此 S_2 和 S_3 是等价状态,可以合并为一个。

从物理概念上也不难理解,当电路处于 S_2 状态时表明已经连续输入了两个 1。如果在电路转换到 S_2 状态的同时输入也改换为下一位输入数据(当输入数据来自移位寄存器的串行输出,而且移位寄存器和数据检测器由同一时钟信号操作时,就工作在这种情况下),那么只要下个输入为 1,就表明连续输入 3 个 1 了,因而无需再设置一个电路状态。于是就得到了图 6.4.8 所示化简后的状态转换图。



在电路状态 $M = 3$ 的情况下,根据式 图 6.4.7 例 6.4.2 的状态转换图

(6.4.1)可知,应取触发器的位数 $n=2$ 。

如果取触发器状态 Q_1Q_0 的00、01和10分别代表 S_0 、 S_1 和 S_2 ,并选定JK触发器组成这个检测电路,则可从状态转换图画出电路次态和输出的卡诺图,如图6.4.9所示。

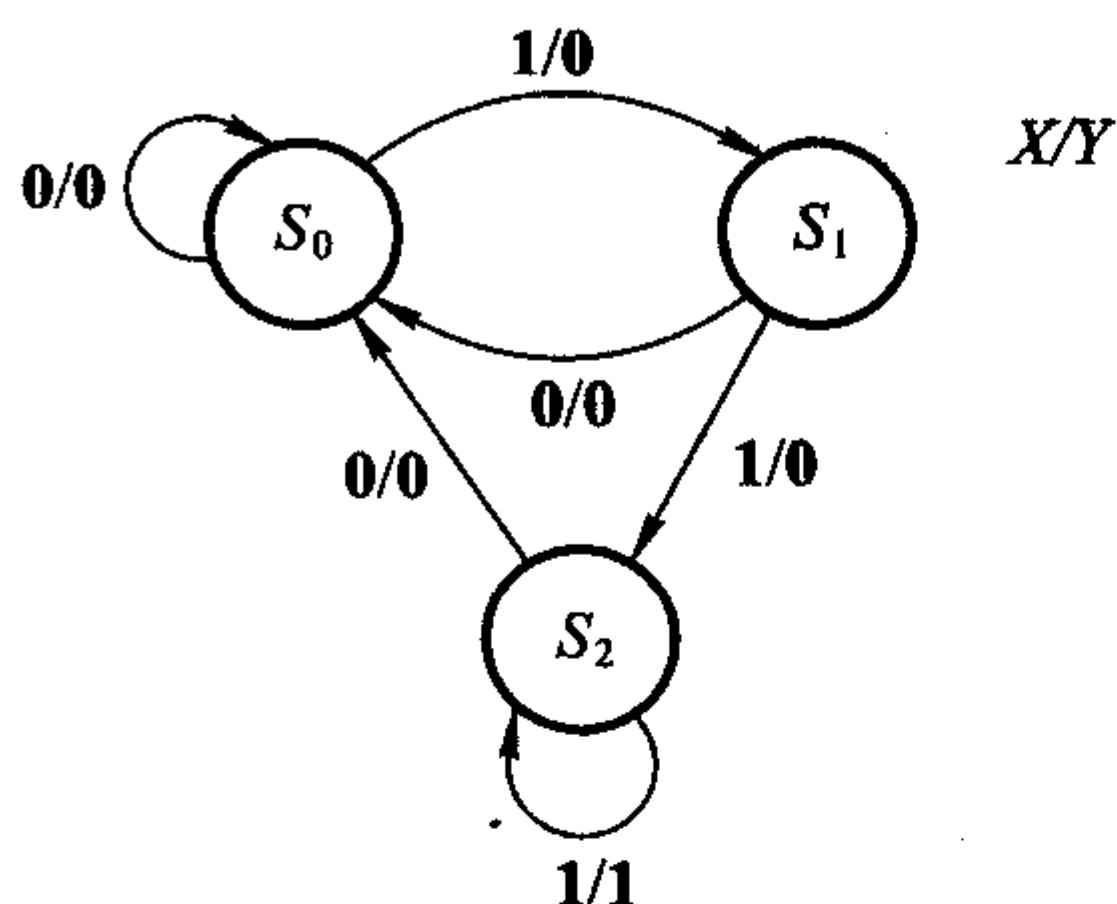


图 6.4.8 化简后的例 6.4.2 的状态转换图

Q_1Q_0		X			
		00	01	11	10
X	0	00/0	00/0	××/×	00/0
	1	01/0	10/0	××/×	10/1

图 6.4.9 例 6.4.2 电路次态/输出 ($Q_1^*Q_0^*/Y$) 的卡诺图

将图 6.4.9 所示的卡诺图分解为图 6.4.10 中分别表示 Q_1^* 、 Q_0^* 和 Y 的 3 个卡诺图,经化简后得到电路的状态方程为

Q_1Q_0		X			
		00	01	11	10
X	0	0	0	×	0
	1	0	1	×	1

(a) Q_1^*

Q_1Q_0		X			
		00	01	11	10
X	0	0	0	×	0
	1	1	0	×	0

(b) Q_0^*

Q_1Q_0		X			
		00	01	11	10
X	0	0	0	×	0
	1	0	0	×	1

(c) Y

图 6.4.10 图 6.4.9 卡诺图的分解

$$\begin{cases} Q_1^* = XQ_1 + XQ_0 = XQ_1 + XQ_0(Q_1 + Q_1') \\ \quad = (XQ_0)Q_1' + XQ_1 \\ Q_0^* = XQ_1'Q_0' = (XQ_1')Q_0' + 1'Q_0 \end{cases} \quad (6.4.6)$$

由上式得驱动方程

$$\begin{cases} J_1 = XQ_0, & K_1 = X' \\ J_0 = XQ_1', & K_0 = 1 \end{cases} \quad (6.4.7)$$

由图 6.4.10(c)得输出方程

$$Y = XQ_1 \quad (6.4.8)$$

根据式(6.4.6)、(6.4.7)、(6.4.8)所画出的逻辑图和电路状态转换图如图 6.4.11 和图 6.4.12 所示。状态转换图表明,当电路进入无效状态 11 后,若 $X =$

1则次态转入10;若 $X=0$ 则次态转入00,因此这个电路是能够自启动的。

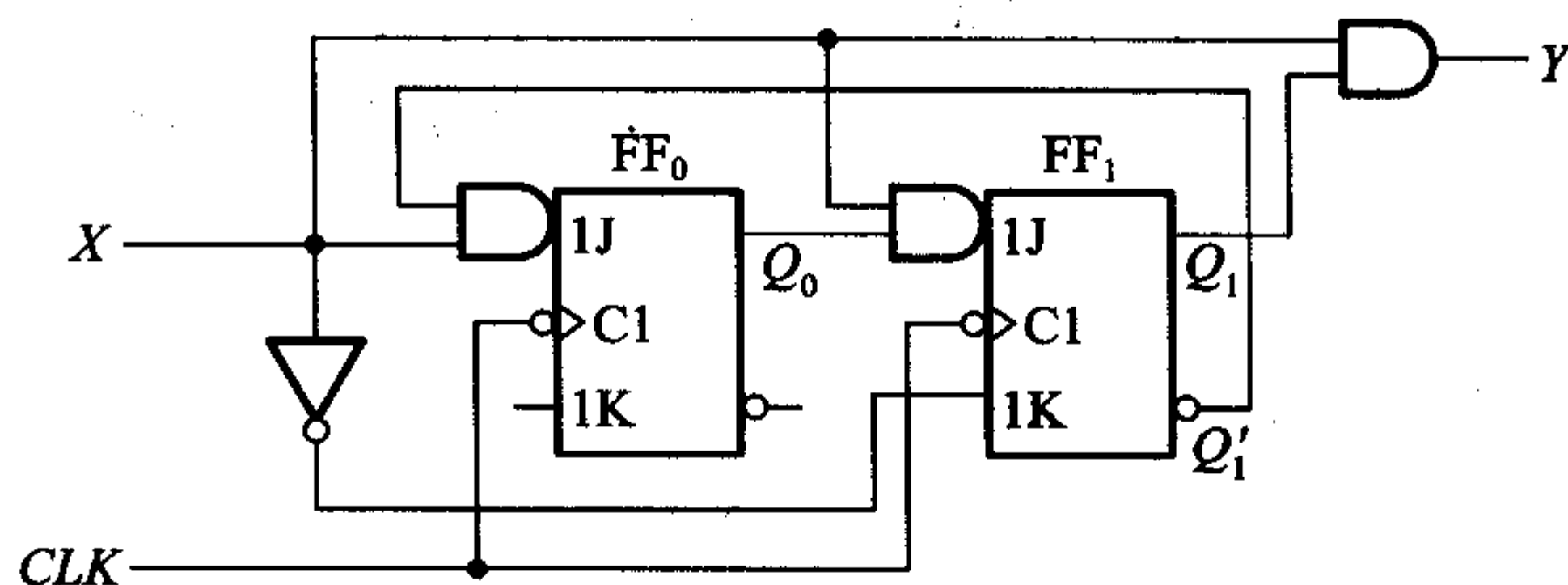


图 6.4.11 用 JK 触发器设计的例 6.4.2 电路

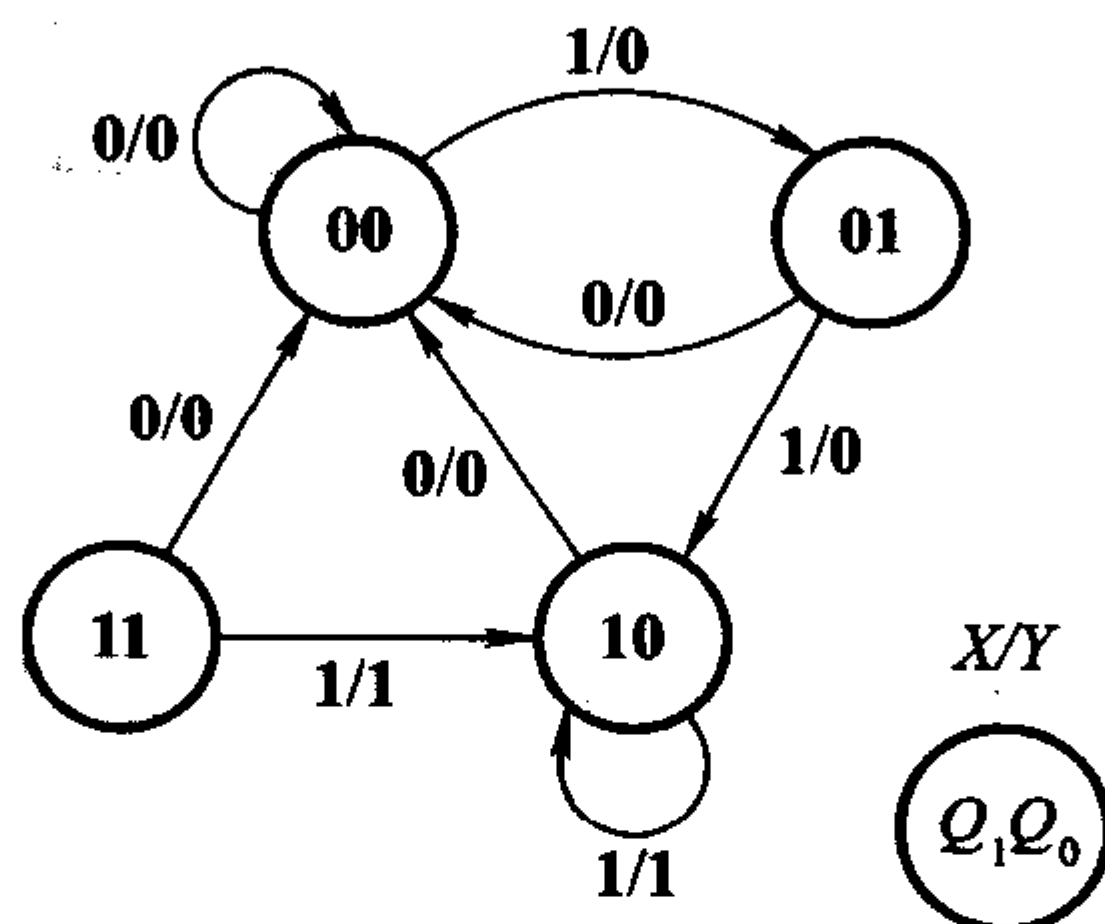


图 6.4.12 图 6.4.11 电路的状态转换图

本例中若改用 D 触发器,则应将式(6.4.6)的状态方程与 D 触发器的特性方程 $Q^* = D$ 对照,找出 D 端对应的逻辑式来,此即 D 触发器的驱动方程。于是得到

$$\begin{cases} D_1 = XQ_1 + XQ_0 = X(Q'_1Q'_0)' \\ D_0 = XQ'_1Q'_0 \end{cases} \quad (6.4.9)$$

而输出方程不受影响。

根据式(6.4.9)和式(6.4.8)得到的逻辑图如图 6.4.13 所示。它的状态转换图与图 6.4.12 相同。

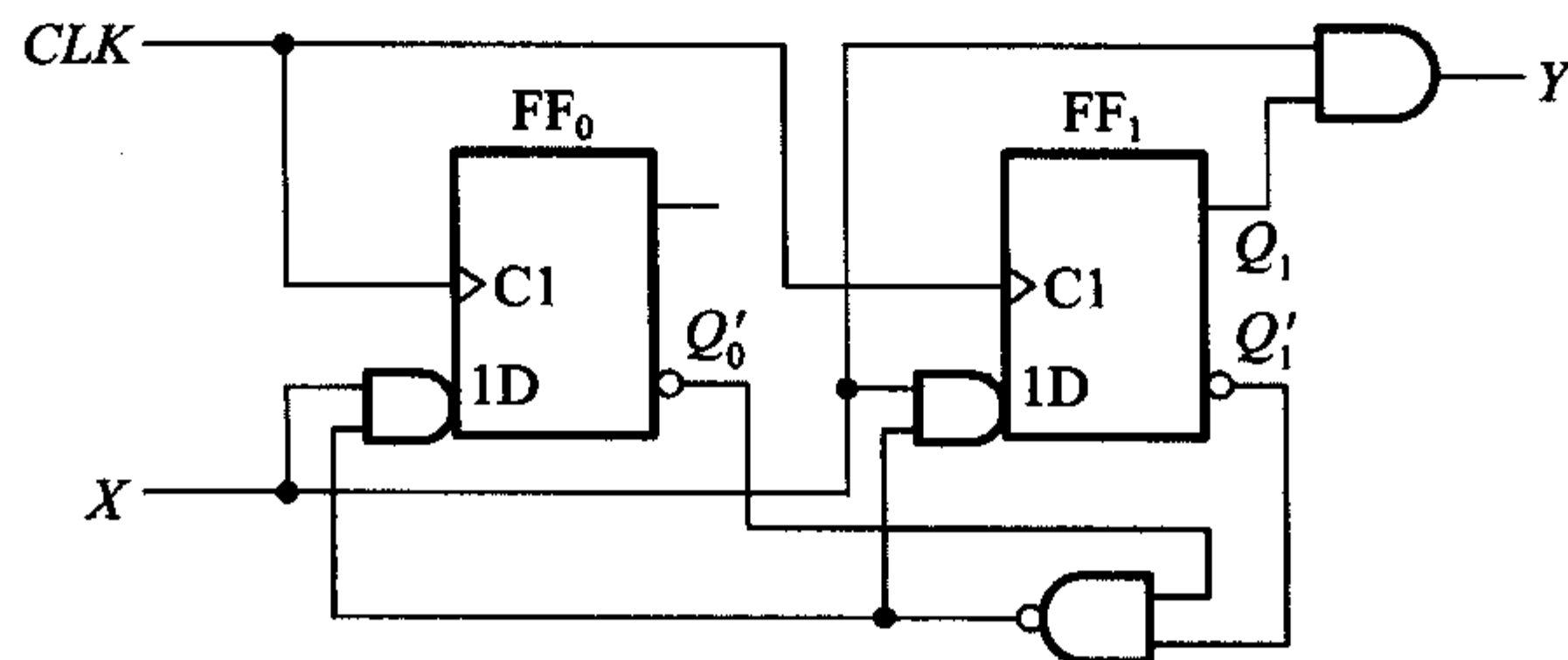


图 6.4.13 用 D 触发器设计的例 6.4.2 电路

【例 6.4.3】 设计一个自动售饮料机的逻辑电路。它的投币口每次只能投入一枚五角或一元的硬币。投入一元五角钱硬币后机器自动给出一杯饮料；投入两元（两枚一元）硬币后，在给出饮料的同时找回一枚五角的硬币。

解： 取投币信号为输入逻辑变量，投入一枚一元硬币时用 $A=1$ 表示，未投入时 $A=0$ 。投入一枚五角硬币用 $B=1$ 表示，未投入时 $B=0$ 。给出饮料和找钱为两个输出变量，分别以 Y 、 Z 表示。给出饮料时 $Y=1$ ，不给时 $Y=0$ ；找回一枚五角硬币时 $Z=1$ ，不找时 $Z=0$ 。

假定通过传感器产生的投币信号（ $A=1$ 或 $B=1$ ）在电路转入新状态的同时也随之消失，否则将被误认作又一次投币信号。

设未投币前电路的初始状态为 S_0 ，投入五角硬币以后为 S_1 ，投入一元硬币（包括投入一枚一元硬币和投入两枚五角硬币的情况）以后为 S_2 。再投入一枚五角硬币后电路返回 S_0 ，同时输出为 $Y=1$ 、 $Z=0$ ；如果投入的是一枚一元硬币，则电路也应返回 S_0 ，同时输出为 $Y=1$ 、 $Z=1$ 。因此，电路的状态数 $M=3$ 已足够。依据题意可列出如表 6.4.3 所示的状态转换表，并画出如图 6.4.14 所示的状态转换图。

表 6.4.3 例 6.4.3 的状态转换表

$S^* \backslash YZ$ AB	00	01	11	10
S_0	$S_0/00$	$S_1/00$	$\times/\times\times$	$S_2/00$
S_1	$S_1/00$	$S_2/00$	$\times/\times\times$	$S_0/10$
S_2	$S_2/00$	$S_0/10$	$\times/\times\times$	$S_0/11$

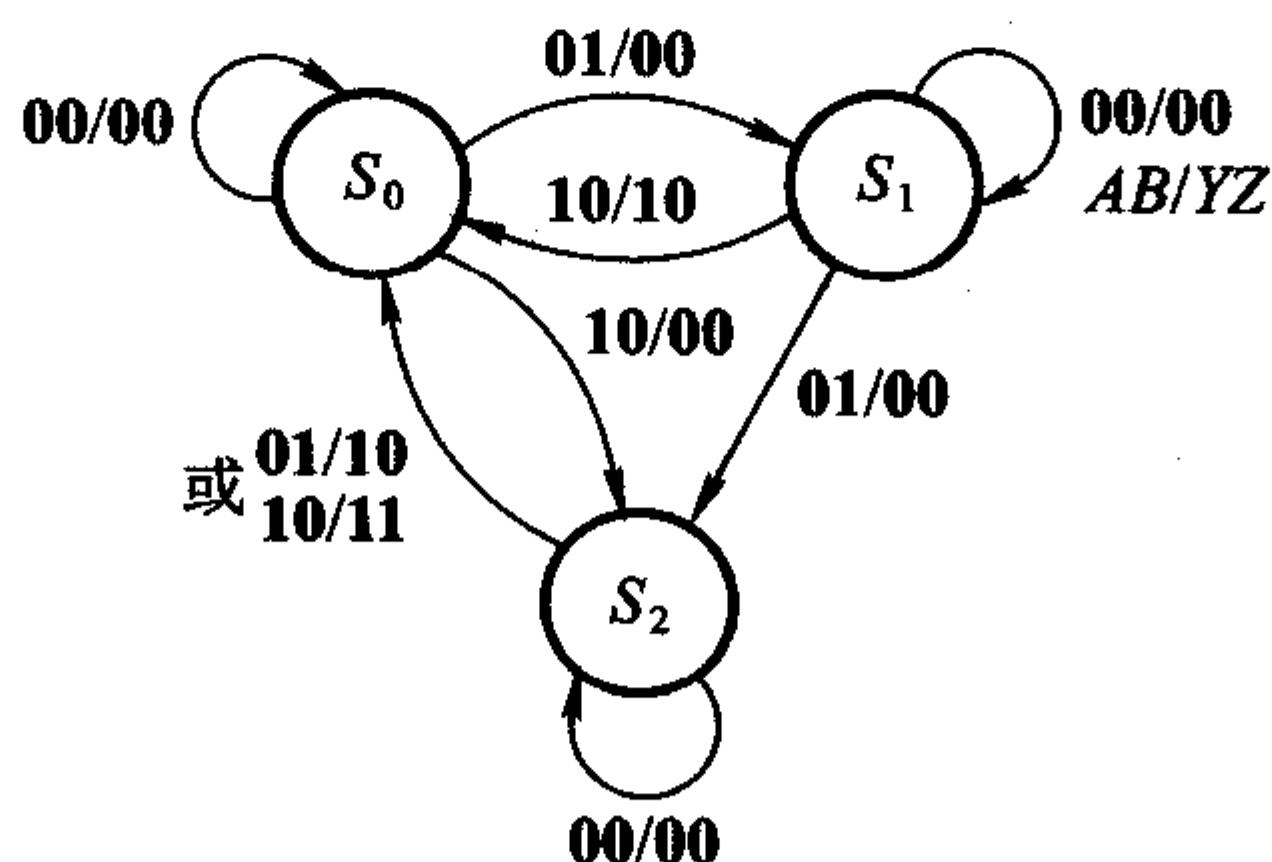


图 6.4.14 例 6.4.3 的状态转换图

因为正常工作中不会出现 $AB=11$ 的情况，所以与之对应的 S^* 、 Y 、 Z 均作约束项处理。

取触发器的位数 $n=2$ ，则 $2^1 < 3(M) < 2^2$ ，故符合要求。今以触发器状态