



时序电路逻辑设计
与特殊组合函数

[美] FREDERICK J. HILL GERALD R. PETERSON 著

郑友明 江太辉 译

高等教育出版社



ISBN 7-04-001997-3/TN·106

定 价 4.95 元



时序电路逻辑设计 与 特殊组合函数

[美] Frederick J. Hill & Gerald R. Peterson 著

郑友明 江太辉译

高等教育出版社

内 容 简 介

本书译自(美) Frederick J. Hill & Gerald R. Peterson 著 "Introduction to Switching Theory & Logical Design" (3-rd Edition) 的 9~16章。主要内容有时序电路导论、时钟型时序电路的综合、时钟控制和脉冲型电路采用MSI和LSI组件的逻辑设计、非完全确定的时序电路、电位型时序电路、触发器和定时的再考察、特殊组合函数, 并有两个附录。

原书是一本既注重基本理论又强调工程实际的、颇受读者欢迎的逻辑设计方面的教科书, 自1968年出版以来, 已出三版。取材精练, 概念清楚, 尤其对时序电路(时钟型、脉冲型、电位型)分析清晰, 条理性强。并介绍了用硬件描述语言(AHPL)进行逻辑设计的方法。各章均有丰富的例题和习题。

译者节译了9~16章, 突出时序电路的设计, 故更书名为《时序电路逻辑设计与特殊组合函数》。

本书可供高等学校工科无线电、计算机、自动化等专业作选修课教材, 也可供各专业师生及工程技术人员参考。

责任编辑 姚玉洁

Introduction to Switching Theory & Logical Design (Chapter 9~16)

3-rd Edition (1981)

[美] Frederick J. Hill & Gerald R. Peterson

John Wiley & Sons

时序电路逻辑设计与特殊组合函数

[美] Frederick J. Hill & Gerald R. Peterson 著

郑友明 江太辉 译

高等教育出版社出版
新华书店北京发行所发行
天津新华印刷一厂印装

开本 787×1092 1/16 印张 17 字数 380 000

1990年7月第1版 1990年7月第1次印刷

印数 0 001—1500

ISBN 7-04-001997-3/TN·106

定 价 4.95 元

译 者 序

[美] Frederick J. Hill 和 Gerald R. Peterson 所著《Introduction to Switching Theory and Logical Design》(开关理论与逻辑设计导论)一书,是作者根据长年教学经验和研究成果而写成的。该书内容丰富,论述严谨,既注意基本理论,又强调工程实际,是一本颇受欢迎的逻辑设计方面的教科书。该书初版于1968年,经作者一再补充修订,1981年已出第三版。第三版共十六章,前八章介绍逻辑设计基础知识和组合逻辑电路设计方法;第九到十五章讲述时序电路设计;第十六章介绍几种特殊组合函数;书末尚有两个附录。

原书的篇幅比较长。考虑到逻辑设计基础知识及组合逻辑电路设计方面的参考资料比较多,而该书时序电路部分的内容组织得相当有特色,特别用硬件描述语言(AHPL)方法,使用中大规模集成电路设计逻辑电路部分更是具有较大的参考价值,为了减轻读者的负担,我们选译了该书第九到十六章及两个附录,现在的书名也是译者加的,对章节顺序也重新做了安排。译本的第一章为时序电路引论,介绍时序电路的基本概念。第二章讲述时钟型时序电路的综合。第三章讨论时钟控制和脉冲型电路。第四章讲述使用AHPL,用中大规模集成电路设计逻辑电路的基本原理和方法。第五章讨论非完全确定时序电路。第六章介绍电位型时序电路的设计。第七章再次对触发器和定时问题作更深入的讨论。第八章介绍几种具有某些特殊性质的组合函数。附录A介绍关于最小闭覆盖的选择方法。附录B简单介绍继电器电路的有关问题。只要具备数字电路和组合电路设计的基本知识,就能阅读本书。

本书第一、二、三、四和八章由郑友明翻译,第五、六、七章及两个附录由江太辉翻译,全书由郑友明统稿。

由于译者的业务水平和外语水平有限,错误和不当之处在所难免,欢迎读者指正。

译 者

1988.4

目 录

第一章 时序电路导论	(1)	(74)
1.1 一般特性	(1)	8.1 引言	(74)
1.2 触发器	(2)	8.2 一般脉冲型	(75)
1.3 为什么要用时序电路	(6)	8.3 重读计数器	(83)
1.4 移位寄存器和计数器	(9)	8.4 时钟输入控制	(85)
1.5 速度与成本的折衷选择	(10)	8.5 扩展的状态表	(88)
1.6 时序电路的一般模型	(12)	8.6 程序描述	(90)
1.7 时钟型和脉冲型时序电路	(13)	8.7 综合	(91)
1.8 触发器中的定时问题	(15)	8.8 向量运算	(96)
1.9 标准的边沿触发 D 触发器	(18)	8.9 条件转移	(99)
1.10 电位型时序电路	(21)	习题	(108)
1.11 小结	(22)	参考文献	(108)
习题	(22)	第四章 采用 MSI 和 LSI 组件的逻辑设计	(107)
参考文献	(24)	4.1 引言	(107)
第二章 时钟型时序电路的综合	(25)	4.2 术语的定义	(107)
2.1 时序电路的分析	(25)	4.3 互连	(109)
2.2 设计过程	(27)	4.4 逻辑本原与组合逻辑单元	(111)
2.3 状态图的综合	(27)	4.5 功能寄存器	(118)
2.4 有限存贮电路	(30)	4.6 标准组件的小程序库	(118)
2.5 等价关系	(35)	4.7 上升沿或下降沿触发	(119)
2.6 等价状态和等价电路	(36)	4.8 若干设计举例	(119)
2.7 不可区分状态类的确定	(38)	4.9 功能寄存器的链接	(125)
2.8 用隐含表化简	(43)	4.10 存贮器的使用	(130)
2.9 Mealy 电路和 Moore 电路	(47)	4.11 集成电路的经济问题	(132)
2.10 状态分配和存贮元件输入方程	(50)	4.12 模拟	(135)
2.11 划分和状态分配	(62)	4.13 测试序列的生成	(136)
2.12 小结	(68)	4.14 小结	(138)
习题	(69)	习题	(138)
参考文献	(72)	参考文献	(142)
第三章 时钟控制和脉冲型电路	(72)	第五章 非完全确定的时序电路	(144)
.....	(74)	5.1 引言	(144)

5.2 相容性.....	(146)	7.8 电位型模拟.....	(214)
5.3 一个完整的例子.....	(154)	7.9 小结.....	(216)
习题.....	(159)	习题.....	(216)
参考文献.....	(162)	参考文献.....	(217)
第六章 电位型时序电路.....	(164)	第八章 特殊组合函数.....	(219)
6.1 引言.....	(164)	8.1 引言.....	(219)
6.2 基本模式电路分析.....	(165)	8.2 对称函数.....	(219)
6.3 流程表的综合.....	(168)	8.3 对称函数的布尔联合.....	(221)
6.4 最小化.....	(168)	8.4 高阶型.....	(223)
6.5 转换表、激励图和输出图.....	(173)	8.5 简单不相交分解.....	(225)
6.6 一次操作与竞争.....	(175)	8.6 复杂的不相交分解.....	(231)
6.7 无竞争状态分配.....	(177)	8.7 迭代网络.....	(235)
6.8 时序电路中的冒险.....	(185)	8.8 次序关系.....	(238)
6.9 一般电位型电路.....	(189)	8.9 单边函数.....	(240)
习题.....	(195)	8.10 广义电阻-晶体管逻辑电路.....	(243)
参考文献.....	(199)	8.11 线性可分性.....	(245)
第七章 触发器和定时的再考察.....	(200)	习题.....	(248)
7.1 引言.....	(200)	参考文献.....	(250)
7.2 边沿触发 D 触发器.....	(200)	附录 A 最小闭合覆盖的选择.....	(252)
7.3 边沿触发和时钟的时滞.....	(203)	参考文献.....	(257)
7.4 主-从 $J-K$ 触发器的流程表.....	(204)	附录 B 继电器电路.....	(258)
.....	(204)	B.1 继电器电路的基本特性.....	(258)
7.5 电位型实现的另一种方法.....	(206)	B.2 对称函数的继电器实现.....	(261)
7.6 实现主-从 $J-K$ 触发器.....	(208)	B.3 时序电路中的继电器.....	(262)
7.7 竞争和冒险的分析.....	(210)	参考文献.....	(263)

第一章 时序电路导论

1.1 一般特性

研究图1.1开关系统的一般模型，这个模型有 n 个输入变量，即激励变量 $x_k(t)$ ， $k = 1, 2, \dots, n$ ； p 个输出变量，即响应变量 $Z_i(t)$ ， $i = 1, 2, \dots, p$ ，并且假定 x, Z 都是时间的函数。如果在任一特定的时间上，系统输出的现在值仅仅决定于输入的现在值，我们就说系统是组合系统，用一组形式为

$$Z_i = F_i(x_1, x_2, \dots, x_n) \quad (1.1)$$

的方程就能完全描述这样的系统，其中变量与时间的关系不需明确指出，这是因为认为所有变量的值都是在某一单独时间上的值。反之，如果输出的现在值不仅取决于输入的现在值，而且也取决于系统过去的历史，那么我们就说系统是时序系统。

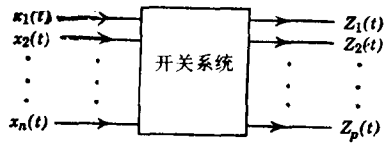


图 1.1

作为阐明这种区分的例子，让我们来考察两种组合锁*(图1.2)。第一种锁，普通的组合挂锁(图1.2 a)，只有一个拨盘；第二种锁，有时用在皮箱中，有几个拨盘(图1.2 b)。就这两种锁而言，输入是拨盘的位置，输出是锁的状态，即打开或者锁上。对于第一种锁来说，锁的现在状态(打开或者锁上)不仅取决于拨盘现在的位置，而且还取

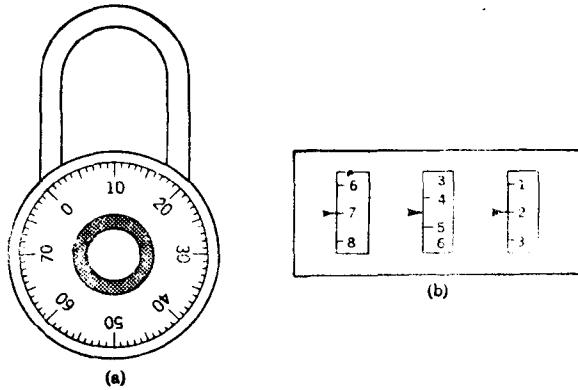


图 1.2 两种组合锁

决于前面拨动拨盘的方法。第二种锁，则只要把三个拨盘都置于正确的数字上，锁就打开，而与拨盘以前的位置或者是按什么次序拨动拨盘无关。很显然，单拨盘组合锁是一种时序装置，多拨盘组合锁是一种组合装置。

电话系统是时序系统的另一个实例。假如你已经拨了电话号码的前六位数，而现在正在拨最后一位，那么这第七位数就是系统的现在输入，输出就是使你与需要联系的一方接通的信号。显然，现在输入不是决定你得到哪一个号码的唯一的因素，因为其它六位数同样重

* “组合”一词当用于说明锁的类型时，与这里讨论中的专门术语“组合”并无联系。

数字计算机是非常重要的时序系统实例。在数字计算机中有许多级涉及顺序使用子系统的时序操作，而这些子系统本身又是时序系统。考察把一个数加到计算机累加器中的情况可知，产生的和既取决于加进去的数，也取决于以前已经放在累加器中的数。

1.2 触发器

在上述的时序装置和系统中，显然需要贮存关于以前事件的信息。在组合锁中，关于以前拨盘位置的信息贮存于某些零件的机械位置中。在自动电话系统中，每拨一个数，这个数就贮存在继电器或电子器件中。在计算机中，许多不同的操作都涉及贮存信息。

存贮器件有许多种。在电子时序电路或者系统中，最普通的存贮器件是触发器*。图1.3表示由两个或非门构成的触发器的电路，以及典型工作时序的定时图。为了便于解释电路的工作，我们在图中也画出了或非门的真值表。

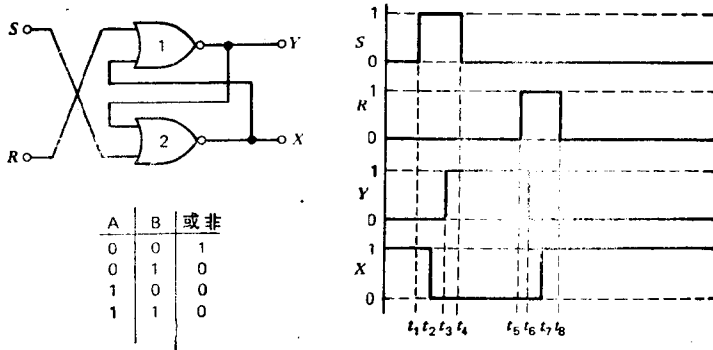


图 1.3 触发器的工作原理

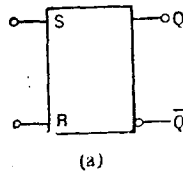
开始时，两个输入都是0，Y输出是0，X输出是1。因为输出反馈到门的输入，所以我们必须进行检验，务必使情况与所假定的一致。门1的输入是 $R=0$ 和 $X=1$ ，使输出 $Y=0$ ，因此检验无误。同样，门2的输入是 $S=0$ 和 $Y=0$ ，使输出 $X=1$ 。在时刻 t_1 ，输入S变为1，于是门2的输入从00变到01；经过延迟，在时刻 t_2 ，X从1变为0，使门1的输入从01变到00；这样，在时刻 t_3 ，Y从0变为1，这又使门2的输入从01变到11，然而这个变化对输出没有影响。类似地，在时刻 t_4 ，S从1变为0也不影响输出。当R变为1时，Y变为0，使X变为1，于是“锁住”Y，因此，当R回到0时，就不再影响电路的工作。

按上述方式进行工作的触发器称为复位-置位触发器或R-S触发器，其框图符号如图1.4所示。

在框图符号中，S和R输入分别对应于图1.3中的S和R，Q和 \bar{Q} 输出分别对应于图1.3中的Y

* 更正式的名称是双稳态多谐振荡器，触发器这个名称的由来不明，但已成标准。

和 X 。 S 输入上的脉冲将使触发器“置位”，即把 Q 输出驱动到1电位，而把 \bar{Q} 输出驱动到0电位。 R 线上的脉冲将使触发器“复位”，即把 Q 输出驱动到0电位，而把 \bar{Q} 输出驱动到1电位。在图1.4b所示的转换表中，以表格的形式概括表示出触发器的这种工作情况，表中 S^v 、 R^v 和 Q^v 表示在某一任意时刻 t_v 输入和输出的值，而 Q^{v+1} 表示由于在 t_v 时输入作用的结果输出将要达到的值。读者应该确信，可以用下面的方程表示这张表：



S^v	R^v	Q^v	Q^{v+1}
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	x
1	1	1	x

图1.4 R-S触发器

$$Q^{v+1} = S^v + Q^v \bar{R}^v \quad (\text{其中 } S^v \cdot R^v = 0) \quad (1.2)$$

在最后的两行中， Q^{v+1} 的随意值反映出这样的事实，即在正常工作时，不应允许两个输入同时为1。施加这种限制的原因有二：第一，如果两个输入都为1，则将把两个输出都驱动到0，这就违背触发器工作的基本定义，定义要求触发器的两个输出应该总是互补的。第二，如果两个输入都是1，而然后同时回到0，那么两个或非门的输入将都是00，而两个输出都将企图变为1，但是由于反馈，两个输出同时为1是不可能的（读者自己应该确信这一点），结果，触发器的转换就会不可预知，甚至会进入振荡。对输入施加这种限制后， R - S 触发器就可靠地起着存贮器件的作用，它的输出状态表示最末的1电位是在哪个输入。

在上面的讨论中，我们讲到 R 和 S 线上的脉冲。对于读者来说，确切理解脉冲一词的含义是重要的。在组合电路中，我们主要研究信号处于稳定时的情况，而对以前的值或者转换一般不感兴趣。时序电路的情况则不同。考察 R - S 触发器，并假定在特定的时刻上 $R=0$ 和 $S=0$ 。如果这是一个组合电路，那么我们就从这些信息确定出输出，然而在这个电路中，我们还需要知道，最末的1电位是在哪个输入。于是工作的时间特性就成为一个重要的因素。

再次假定 $R=S=0$ ，并且还假定 $Q=0$ （触发器复位）。假设我们想把触发器置位。我们从图1.3看到，这就要求 S 上升到1，而 R 保持在0。如果电路是组合的，我们就能使 S 在任一时刻上回到0；然而在目前的情况下，如果要可靠地完成触发器的状态转换， S 就必须在某一最短的时间内保持为1。如果 S 在时刻 t_1 （图1.3）之前回到0，那么门2的两个输入至少会瞬间再次都为0，这势必使 X 回到1，在这种情况下，电路的工作将不可预知。一旦到达置位状态的转换已经发生， S 就能在任一时刻上回到0，因为这个转换不会再引起进一步的变化。但是我们注意， S 必须在触发器能由 R 线上的1进行复位之前回到0。最简单的方法是使 S 为1电位的时间只足以保证可靠置位，然后使它回到0电位。

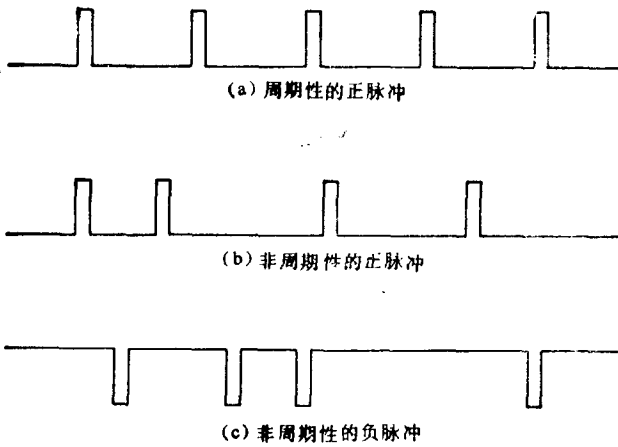


图 1.5 不同类型的脉冲信号

通常保持在一个电位而仅在短时间内到达另一电位的这样一种信号称为脉冲信号。脉冲可以是周期性的，即按规则的时间间隔出现（图1.5a），或者非周期性的（图1.5b和图1.5c）。它们可以是正的（图1.5a和图1.5b），也可以是负的（图1.5c）。上述定义的一个问题是难以详细说明“短时间”究竟是什么意思。“短”是与什么比较？正如在讨论R-S触发器时所指出的那样，脉冲的宽度可以是若干个门延迟的数量级，但也可以长得多，并且仍然有理由认为是脉冲。脉冲和电位之间的重要差别是按电路对它们的响应来区分的。尽管一个脉冲包含两次转换，不管持续时间有多长，仍然把它看作是引起单个响应的单独事件。响应可以在第一次转换（上升沿）之后出现，R-S触发器中情况通常就是这样，也可以在第二次转换（下降沿）后出现。

R-S触发器也可用与非门构成，如图1.6a所示。在图1.6c所示的定时图中我们看到，除了输入通常处于1电位和用负脉冲使触发器置位和复位外，电路的工作与图1.3所示的或非门触发器相同。框图符号（图1.6b）中输入端上的反相圈表示状态转换是在输入的负向转换之后出现，也即在输入从较正的值转换到较负的值之后出现。相反，在或非门触发器中（图1.3），输入端没有反相圈，表明状态转换是在输入的正向转换之后出现。

正如我们将在后面几章中看到的那样，如果我们能使状态的变化同步，也就是使触发器的状态转换同时发生，则能大大地简化包含触发器的系统的设计。使用公共的脉冲信号源同

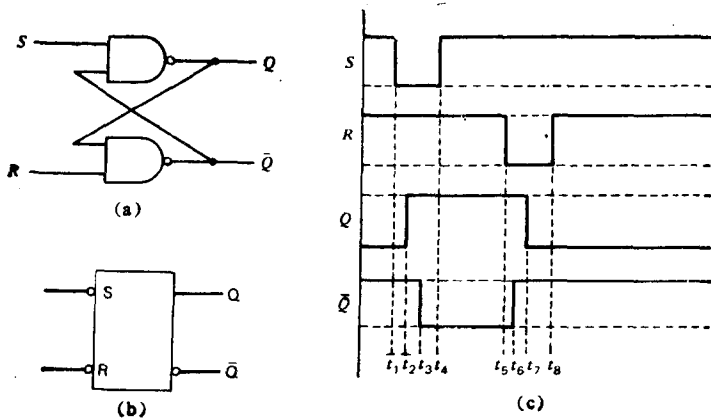


图 1.6 负输入的R-S触发器

步所有的状态转换即能实现这样的操作，这样的定时脉冲信号源称为时钟。时钟脉冲通常是周期性的。使用单独的定时信号意味着触发器必须有单独的输入，以便控制发生什么操作和控制何时发生这样的操作。图1.7a表示在或非门R-S触发器中怎样才能实现这样的单独控制，图1.7b表示一个典型的时间序列。请注意，我们所做的全部工作只是加进两个与门，这

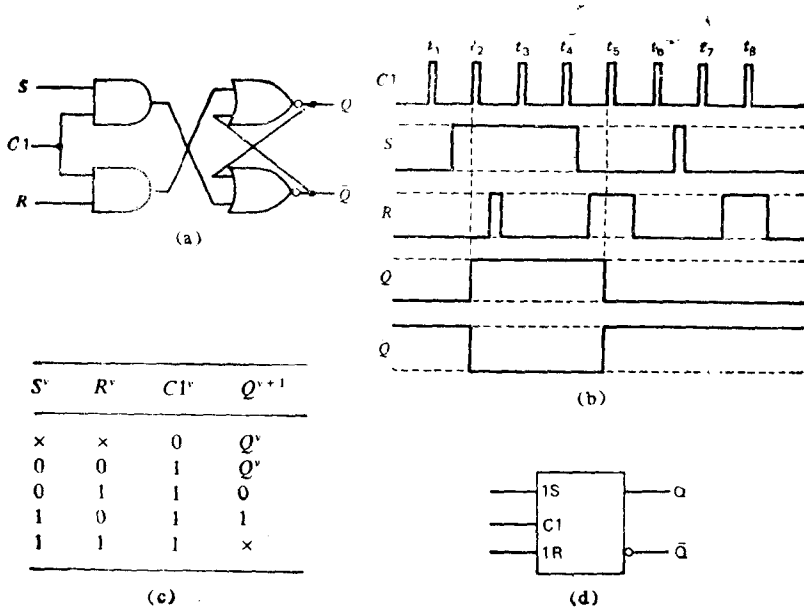
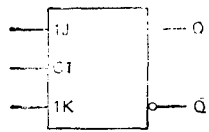


图 1.7 时钟 R-S 触发器

样一来，除非时钟脉冲出现，否则S和R对触发器就没有影响。在时刻 t_1 ，两个输入都是0，因此触发器的状态不变。在 t_2 ， $S=1$ ， $R=0$ ，于是 t_2 的时钟脉冲使触发器置位。在 t_2 和 t_3 之间，R暂时变为1，但因此时的时钟是0，所以它对电路的工作没有影响。在 t_4 和 t_5 之间，S回到0，R变为1，因此 t_5 的时钟脉冲使触发器复位。

图1.7c和1.7d分别表示这个触发器的转换表和框图符号。转换表的第一行表明，只要时钟是0，则不管S和R的状况如何，触发器的状态不变。当时钟为1时，输出由S和R确定，方法与图1.3的电路相同。在正常的时钟工作中，如图1.7b所示，假定在时钟脉冲持续时间内，S和R不变。框图符号（图1.7d）是相应关系符号表示法的一个实例。输入C1是控制输入，它控制以数字1开始作标记的任何其它输入的工作。在目前的情况下，输入1S和1R的影响取决于C1的状态。只要C1为低电位，1S和1R对触发器就没有影响；当C1为高电位时，1S和1R即按触发器的置位、复位输入的正常方式控制触发器的工作。

实践中很少使用时钟R-S触发器，它们几乎已完全为时钟J-K触发器所代替。在J-K触发器中取消了两个输入不能同时为1的限制，而这种限制对设计者来说会不方便。J-K触发器的转换表和框图符号示于图1.8。当J对应于S，K对应于R时，除了在时钟脉冲期间内 $J=K=1$ 时触发器改变状态这一点外，J-K触发器的工作情况与R-S触发器（图1.7c）相同。这种触发器的方程是



J^v	K^v	$C1^v$	Q^{v+1}
x	x	0	Q^v
0	0	1	Q^v
0	1	1	0
1	0	1	1
1	1	1	\bar{Q}^v

图 1.8 J-K 触发器

$$Q^{v+1} = \bar{K}^v Q^v + J^v \bar{Q}^v \quad (1.3)$$

请注意, J-K 触发器总是时钟控制的。

另一种触发器是 D 触发器, 其框图符号及转换表见图 1.9a 和图 1.9b。这种触发器只有一个数据输入 D, 它是存储器件的最直接

的实现。只要加上时钟, D 的现在值就存入触发器。D 触发器的方程很简单, 即

$$Q^{v+1} = D^v \quad (1.4)$$

集成 D 触发器可以买到, 也能用 J-K 触发器连接而成, 如图 1.9c 所示。

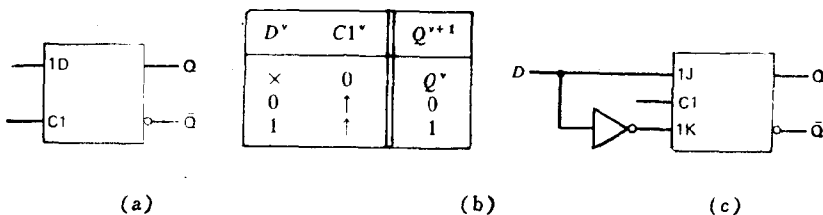


图 1.9 D 触发器

第四种触发器是 T 触发器 (图 1.10), 它只有一个输入 T。只要 T 加上脉冲, 触发器就改变状态, 因此触发器的方程很简单, 即

$$Q^{v+1} = \bar{Q}^v \quad (1.5)$$

T 触发器没有现成的集成电路产品, 用

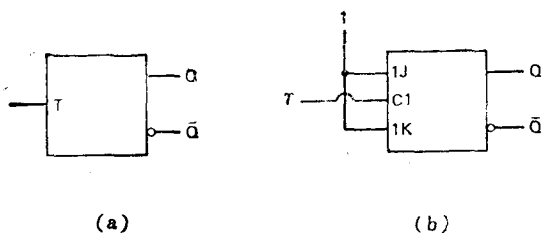


图 1.10 T 触发器

J-K 触发器按图 1.10b 进行连接就可实现 T 触发器。

1.3 为什么要用时序电路

我们已经考察了几个时序系统的例子。显然, 许多现有系统是以类似方式工作的。但是作为一个设计者, 我们的任务不仅是分析电路, 而且要选择最最好的方法综合电路。已知某一个逻辑任务需要完成, 我们就面临着对任务的实现作出明确的决定: 我们应该使用组合电路呢, 还是使用时序电路? 我们既然知道如何在时序电路中存储信息, 那么就让我们来研究做出这种决定时应该考虑的若干因素。



图 1.11 tic-tac-toe 游戏模型

* tic-tac-toe 系一种两人游戏。两人轮流在一有九个方格的盘上画 O 或 X, 以所画的三个符号先连成直线、横线或斜线者为胜。——译者注

我们来探讨一下 tic-tac-toe 机* 的设计问题。整个“游戏系统”可表示如图 1.11。游戏盘可由开关和显示灯组成，游戏者可用开关指示出他的下法，而灯光显示游戏的现在状态。机器人将由某种逻辑系统所组成，以便决定为响应游戏者的下法，机器人应该下哪一着。

显然，整个游戏过程是时序的，但这并不意味着机器人必须是时序装置。它可以是一个组合电路，在任一给定的时间上，电路从游戏盘上接收关于游戏盘状态的信息，并且仅在这些现有信息的基础上决定下哪一着。游戏盘有九个方格，每个方格有三种可能的状态（空白，O，X），所以一共有 $3^9 = 19683$ 种可能的游戏情况。这样一来，组合方法需要使用电路实现下述逻辑要求：给定 3^9 种可能输入组合中的任何一种组合，选择九种可能的响应中正确的一种响应。十分明显，满足这个要求势必要使用大量的组合逻辑电路。

作为另一种可能使用的方法，让我们使机器人随着游戏的进行而以某种方法改变它的结构，以便利用这样的事实：即在游戏中的任一特定点上，并非 3^9 种情况都能同时出现。让我们假定游戏者先下，因为只有九种可能的下法，所以无论游戏者下了哪一着，机器人都开始布局，目的是选择正确响应，并且当机器人选出它的响应时它就知道，游戏者下一着可以使用的响应是哪七个，于是它能够改变它的电路，以便考虑这一点。后续每一着都重复类似的过程。

在这种时序方法中，我们只是把一个大问题分解成若干个较小的问题。我们并不去组装一个能处理游戏中任一点上可能出现的所有情况的大型电路，而是组装若干较小的电路*，每个电路能够处理游戏中某些特定点上可能出现的有限种不同情况。因此，随着游戏的逐步进行，机器人有顺序地从一个较小的电路进入另一个较小的电路。

在当前的情况下，问题本身就是时序的，这是因为它涉及在游戏双方之间不断地交换信息。这种与问答形式十分相象的信息交换，在数字系统的各个单元之间是常见的。这种固有的时序过程自然导致采用时序设计方法，这种方法通常比纯粹的组合方法更为经济。后面几章引出的方法会使读者能够设计（如果他倾向于这样做的话）自己的 tic-tac-toe 机，并且能验证时序设计更为经济这样一个论点。

作为另一个例子，研究 8 位奇数奇偶校验器。如果可以同时得到全部 8 位，那么由方程

$$z = [(x_1 \oplus x_2) \oplus (x_3 \oplus x_4)] \oplus [(x_5 \oplus x_6) \oplus (x_7 \oplus x_8)] \quad (1.6)$$

描述的六级组合电路即能实现所需要的功能，这个电路若用与非门实现，共需 21 个门。

其次让我们假设，8 位字是必须远距离传送（例如在航空专用系统中）的一个数据单元。我们可用 8 根线，一位一位来进行传送，但这会太贵。相反，让我们使用一根线而按顺序传送 8 位， x_1 在时刻 t_1 进行传送， x_2 在时刻 t_2 传送，等

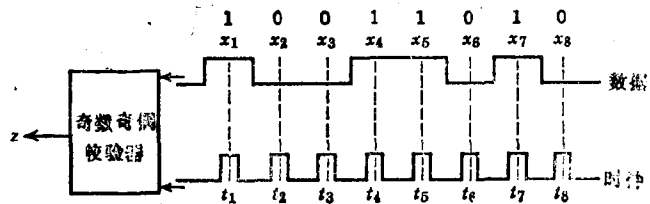


图 1.12 串行数据的奇偶校验器

* 正如将在后面各章中可以证明的那样，这些电路通常将不是完全独立的。

等。我们在第二根线上传送时钟脉冲即同步脉冲，以标明 t_1, t_2, \dots 这些时刻*。这两根线将向电路提供输入，如果每 8 位一组之中 1 的个数不是偶数，电路就必须发出信号（图 1.12）。

直到全部 8 位都出现为止，当每一位到达时可把它贮存起来，然后把它们加到一个六级组合电路上，但这种方法会非常昂贵，它需要用于贮存的八个触发器和一个 21 个门的网络。相反，我们注意到，**异或**是可结合的，从而把方程 (1.6) 改写成下面的形式：

$$z = (\{[\{[(x_1 \oplus x_2) \oplus x_3] \oplus x_4\} \oplus x_5] \oplus x_6\} \oplus x_7\} \oplus x_8) \quad (1.7)$$

从功能上讲，这个方程表示先检验前两位的奇偶性，然后根据第一次检验的结果检验第三位，等等。（读者应该用具体的例子试验这种方法，以确信它是行得通的。）

我们并不是建议采用组合方法实现方程 (1.7)，这样做效率会很底。相反，先在一个**异或**网络中检验前两位，并且在第三位到达之前，把检验结果贮存起来，然后将第三位与贮存的结果在同一个**异或**网络中进行比较，在第四位到达之前，这个结果又贮存起来，等等。图 1.13 表示实现这种方法的一个简单电路。

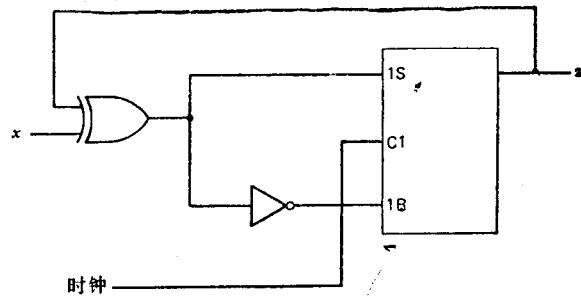


图 1.13 时序奇偶校验器

在 x_1 到达之前，触发器最初必须清零，即 $z = 0$ ，这样第一次工作情况是

$$x_1 \oplus z = x_1 \oplus 0 = x_1$$

当时钟脉冲在时刻 t_1 到达时，就把这个结果存入触发器，即当时钟脉冲到达时，如果 $x_1 \oplus z = 1$ ，脉冲就被选通加到 S 输入，使触发器置 1。同样，当时钟脉冲到达时，如果 $x_1 \oplus z = 0$ ，触发器就置 0。然后 x_2 到达，**异或**操作变为

$$x_2 \oplus z = x_2 \oplus x_1$$

由 t_2 时的时钟脉冲把结果存入触发器。这个过程一直继续到 t_n 时的时钟脉冲后为止，这时的输出 z 就是所需要的奇偶检验，即仅当 8 个数据位有奇数个 1 时才有 $z = 1$ 。事实上，因为最后对触发器清零，所以不论已经接收到多少个数据位，输出 z 都给出其奇数奇偶检验，因此，图 1.13 所示为一般的 n 位奇数奇偶校验器。

tic-tac-toe 机和串行奇偶校验器是使用时序方法往往最好这样一类情况的例子。在上述情况下，使用组合方法在速度上都没有好处。在 tic-tac-toe 机中，游戏的速度取决于游戏者下着的快慢。如果我们从全部 8 位都可使用时起测量组合电路的速度，那么时序奇偶校验器比组合电路慢，但是如果各个数据位无论如何将按顺序到达，那么在速度上就没有什么区别，然而时序电路比组合电路更简单，所以更便宜。

在某些场合中，数据总是顺序出现的，这通常是由于成本或者硬件的限制，而不是逻辑

* 使用 8 线方法时也需要同步线，以便指明数字何时出现。

上的需要。对于调制在载波上且远距离传送的数据来说，情况更是如此。磁盘和磁鼓存储器通常每次送出数据的一位。来自磁带或纸带的数据一般一次出现一个字符（即一次6位或8位）。来自穿孔卡片的数据一次可以出现一个字符或一次出现80个字符。磁芯存储器的输出将成批（几个字）出现，一次一批，每批从12位到100位。不管数据批量的大小如何，当它出现时即进行处理，这样当然不会损失速度，且可节约一定的成本。

1.4 移位寄存器和计数器

我们在图1.13中看到了使用单个触发器的时序电路。正如我们可以预料到的那样，在大多数的时序电路的应用中，需要贮存一位以上的二进制信息。通常，一个时序电路存贮信息的能力是由电路中所包含的触发器数来反映的，因此一个典型的时序电路会包含若干个触发器。

在后面五章中，我们将详细研究任意时序电路的设计。但在目前，让我们一般地介绍触发器的两种特殊的电路组态——移位寄存器和计数器，在时序电路的应用中，这两种电路组态的使用率非常高。在移位寄存器和计数器中，触发器数和信息贮存能力之间一一对应的关系特别清楚。

在数字计算机中，指令通常存于存储器的连续单元之中。例如，如果第一条指令存于第501号单元，那么第二条指令就应存于第502号单元，第三条指令存于503号单元，等等。为了控制程序的顺序，计算机使用指令计数器。程序开始时，将计数器的内容置到第一条指令的地址，随着每执行一条指令，计数器加1，于是把地址变到下一条指令地址。这仅仅是数字系统中使用计数器的一个例子。

尽管可以用任一种触发器设计计数器，但是T触发器计数器的功能最好理解。图1.14表示一个模-16的计数器，所谓模-16指的是计数器从0计数到15，然后回到0。在图1.14中， y_0 表示最低有效位。

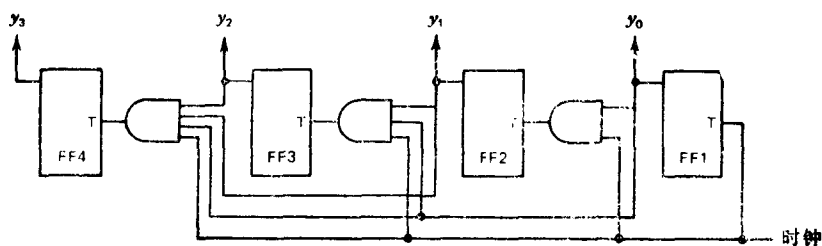


图 1.14 模-16计数器

假设开始时计数是0，即 $y_3y_2y_1y_0 = 0000$ 。当第一个输入脉冲出现时，触发器1改变状态。因为在与门的输出不会出现脉冲，所以其它三个触发器将保持在0状态，于是计数就进到 $y_3y_2y_1y_0 = 0001$ 。现在 y_0 是1，第二个输入脉冲将通过第一个与门触发 y_1 为1，同时把 y_0 触发为0，这样，计数就进到 $y_3y_2y_1y_0 = 0010$ 。通常如果有效位比某一位触发器低的所有触

发器都处于 1 状态，那么当输入脉冲出现时，这位触发器就改变状态，因此，计数过程如图 1.15 所示。

时钟脉冲前的计数				时钟脉冲后的计数			
y_3	y_2	y_1	y_0	y_3	y_2	y_1	y_0
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	1
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0

图 1.15

大多数计数器往往必须复位到零，或者置到某一特定的初始值，通常必须在图 1.14 所示的电路加上实现这种目的的电路。

除了计数外，计数器在输入脉冲的间隔之间也贮存当前的数。任何一组用于存贮有关信息位的触发器通常称为寄存器，因此计数器可以称为计数寄存器。寄存器常常用于收集和贮存来自串行信号源的信息。示于图 1.16 的电路，称为移位寄存器，就是用来完成这种任务的。假定在每个时钟脉冲出现时，可以从 x 线上对信息的新的一位进行抽样。如果第一位是 1，时钟脉冲就使触发器 1 置 1；如果第一位是 0，就把触发器 1 置为 0。第二个时钟脉冲将以相同的方法把第二位移入触发器 1，同时把第一位从触发器 1 移入触发器 2。当对后面的每一位进行抽样时，情况依此类推。

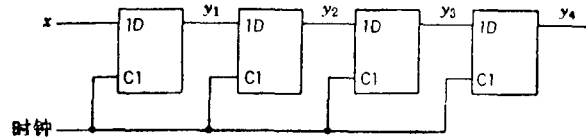


图 1.16 移位寄存器

1.5 速度与成本的折衷选择

在有如前面所讨论的那些场合中，将要进行处理的信息按时间顺序到达，通常自然选择时序电路。因为时序电路对接连的信息段常常重复使用同一设备，所以它们通常提供完成给