

分支图在时序逻辑电路设计中的应用

钱碧云

(电气工程系)

摘 要 首先以序列检测器为例,分析了时序逻辑电路的设计过程,指出求最简状态图是其主要困难之一,并讨论了如何定义最简状态和利用分支图求出状态图的方法。其次,讨论了使用计算机辅助设计的可能性,指出采用分支图法可以使计算机求状态图的工作量由无穷高阶降低到二阶。最后指出,分支图法也适用于其他有输入信号的时序逻辑电路的设计。

关键词 分支图;时序逻辑电路;设计;应用

分类号 TN402

1 状态化简的困难

通常认为,时序逻辑电路的设计步骤如下:^[1~2]

1. 根据实际问题确定状态转换图或状态转换表;
2. 进行状态化简;
3. 进行状态分配;
4. 触发器选型,决定驱动方程和输出方程(若是异步时序逻辑电路的设计,还应选择各个触发器的时钟脉冲);
5. 检查能否自启动,若不能自启动,则应修改第3,4步;
6. 画逻辑图。

在确定状态转换表之后,关键就是状态化简这一步。

以设计“0010”序列检测器为例:当依次把两个“0”电平、一个“1”电平、一个“0”电平输入到序列检测器时,检测器的输出为“1”电平,在其他情况下检测器的输出为“0”电平。

对于字长为4的串行码的检测,由历史上最后输入的三位代码决定其处于何状态,可有 $2^3=8$ 个状态(表1),又可作出其状态转换表(表2)。

收稿日期:1995-01-20.

钱碧云,女,1938年生,副教授。

表 1

状 态	I_{n-2}	I_{n-1}	I_n
s_0'	0	0	0
s_1'	0	0	1
s_2'	0	1	0
s_3'	0	1	1
s_4'	1	0	0
s_5'	1	0	1
s_6'	1	1	0
s_7'	1	1	1

表 2

现态 (Y_n)	输入	次态/输出		Y_{n+1}/Z_n
		$X=0$	$x=1$	
s_0'		$s_0' / 0$	$s_1' / 0$	
s_1'		$s_2' / 1$	$s_3' / 0$	
s_2'		$s_4' / 0$	$s_5' / 0$	
s_3'		$s_6' / 0$	$s_7' / 0$	
s_4'		$s_0' / 0$	$s_1' / 0$	
s_5'		$s_2' / 0$	$s_3' / 0$	
s_6'		$s_4' / 0$	$s_5' / 0$	
s_7'		$s_6' / 0$	$s_7' / 0$	

为了使检测器具有更简单的电路结构,必须对表 2 进行化简. 表 2 的蕴涵表如图 1 所示 (参见文献[3]).

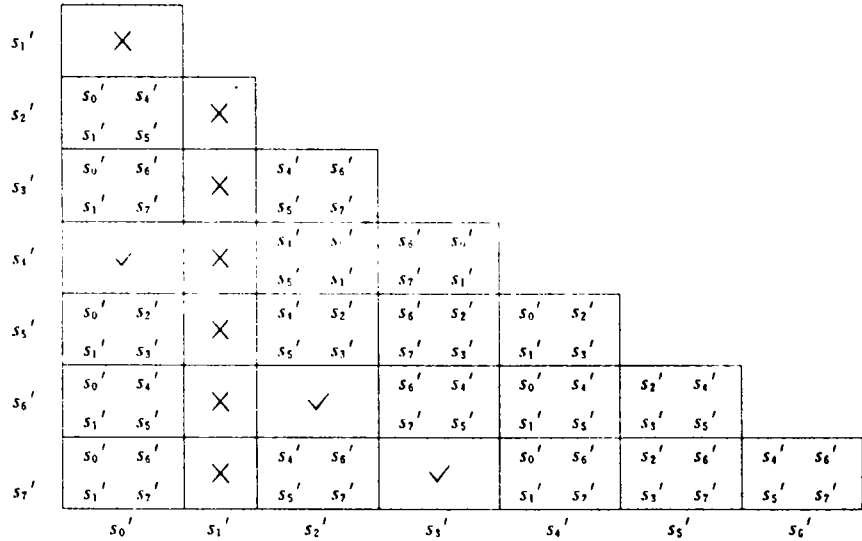


图 1 蕴涵表

分析表 2, $Y_n=s_1'$ 且 $X_n=0$ 时的输出为 1, 而其他状态的输出皆为 0, 因此 s_1' 肯定和其余七个状态皆不等价 (效), 所以, 在蕴涵表中的有关七个单元中应填入“X”号. 单元 (s_0', s_4') 、 (s_2', s_6') 、 (s_3', s_7') 中的两个状态在输入 X_n 相同时转换到相同的状态, 且输出相同, 因此, 在这几个单元应填“✓”号. 其余单元不能立即判断是否等价 (效), 可先将下一状态填入, 上半部填 $X_n=0$ 时的下一状态, 下半部填 $X_n=1$ 时的下一状态. 再继续观察: 若当前状态的下一状态是不等价的, 则这些当前状态的单元中均应填入“X”号; 若下一状态是等价的, 则这些状态的单元中应填入“✓”号. 依此原则可得到化简后的蕴涵表, 如图

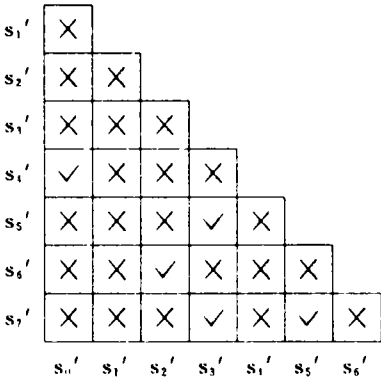


图 2 简化后的蕴涵表

2 所示.

由图 2, 可写出所有的等价状态对为: (s_0', s_4') , (s_2', s_6') , (s_3', s_5') , (s_3', s_7') , (s_5', s_7') . 从而可求出检测器的四个最大等价类: (s_3', s_5', s_7') , (s_2', s_6') , (s_0', s_4') , (s_1') , 分别命名为 s_0, s_1, s_2, s_3 , 得到最小化状态表如表 3 所示. 最简状态转换图如图 3 所示.

表 3

$Y_n \backslash X_n$	Y_{n+1}/Z_n	
	0	1
s_0	$s_1/0$	$s_0/0$
s_1	$s_2/0$	$s_0/0$
s_2	$s_2/0$	$s_3/0$
s_3	$s_1/1$	$s_0/0$

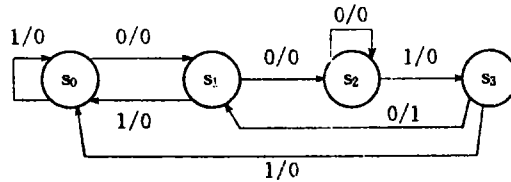


图 3 最简状态转换图

由上可见, 利用蕴涵表化简时, 一是繁, 二是易发生错误. 对于 n 位序列检测器, 当 n 增加时, 其工作量呈指数关系上升. 如当 $n=100$ 时, 求次态和比较的次数超过 10^{60} , 因此, 即使是利用电子计算机来做这一工作, 也颇为困难.

2 跳过状态化简后的困难

如果在定义时序逻辑电路的状态时, 使用的状态数恰如其份, 则在设计过程中, 化简这一步是可以跳过的. 仍以设计“0010”序列检测器为例.

我们定义未输入任何有效代码的状态为状态 s_0 , 已输入且仅输入一位有效代码的状态为状态 s_1 , 已输入且仅输入两位有效代码的状态为状态 s_2 , 已输入三位有效代码的状态为状态 s_3 . 各个状态已输入的有效代码见表 4 所示.

表 4

状 态	t_{n-2}	t_{n-1}	t_n
s_0	?	?	?
s_1	?	?	0
s_2	?	0	0
s_3	0	0	1

表 5

t_{n-2}	t_{n-1}	t_n	状 态
0	0	0	s_2
0	0	1	s_3
0	1	0	s_1
0	1	1	s_0
1	0	0	s_2
1	0	1	s_0
1	1	0	s_1
1	1	1	s_0

我们可对当前时刻最后输入的三位代码的各种可能组合, 与表 4 中的代码自下而上逐一比较, 判断其属于何状态, 得到表 5.

由表 5, 可归纳出集合 s_0, s_1, s_2, s_3

$$\begin{cases} s_0 = (011, 101, 111); \\ s_1 = (010, 110); \\ s_2 = (000, 100); \\ s_3 = (001). \end{cases} \quad (1)$$

对上式进行化简, 可以得到

$$\begin{cases} s_0 = (X11, 101); \\ s_1 = (X10); \\ s_2 = (X00); \\ s_3 = (001). \end{cases} \quad (2)$$

必须指出,在化简时,若前一位未合并,则后一位不允许合并,例如

$(011, 111) = (X11)$ 是可以的,但

$(101, 111) = (1X1)$ 是不允许的.

由式(2)可以得到表3所示的状态转换表和图3所示的状态转换图.

显然,这种方法不必进行状态化简,因而其工作量大大减少.但这种方法必须对表5所示的 2^n 个代码与表4作比较判断,当 n 增大时,其工作量仍呈指数上升.当 $n=100$ 时,其比较的代码数超过 10^{29} .这是一个多么大的数字!

3 分支图法求最简状态转换图

仔细分析各状态的定义,我们发现可以首先对 t_n 时刻输入的代码进行分析.可分为 $XX0$ 和 $XX1$ 两种情况,先将 $XX0$ 与表5自下而上逐个比较,显见,它不可能属于状态 s_3 ,但可能属于 s_2 ;也可能不属于 s_2 .必须进一步分析 t_{n-1} 时刻输入的代码,亦分为 $X00$ 和 $X10$ 两种情况,又将 $X00$ 与表5自下而上逐一比较,可见它属于 s_2 .再比较 $X10$,它属于 s_1 .然后再分析 $XX1$,显见,它可能属于 s_3 ,也可能不属于 s_3 ,必须进一步分析 t_{n-1} ,仍分为 $X01$ 和 $X11$ 两种情况…….如上一一步步进行分析,可以得到分支图如图4所示.显然,这是一个有根二分图,图中虚线箭头则为按深度优先的原则,每一个代码被处理的顺序.

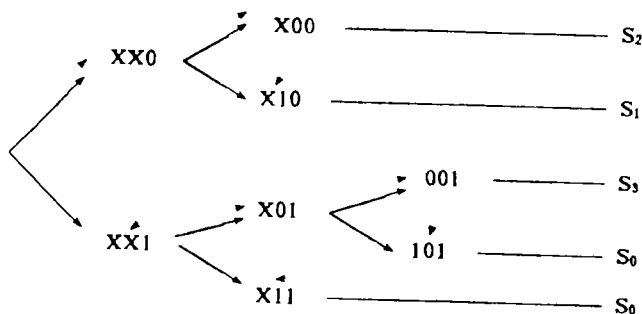


图4 分支图

很明显,由于采用分支图法,省去了由式(1)化简得到式(2)这一步骤.

为了说明分支图法的优越性,我们再看一个例子:设计一个“101110”序列检测器.

首先,定义 $s_0 \sim s_5$ 为输入,且仅输入0~5位有效代码时的状态,各个状态已输入的有效代码见表6所示,然后作分支图如图5所示.

表 6

状 态	t_{n-4}	t_{n-3}	t_{n-2}	t_{n-1}	t_n
s_0	?	?	?	?	?
s_1	?	?	?	?	1
s_2	?	?	?	1	0
s_3	?	?	1	0	1
s_4	?	1	0	1	1
s_5	1	0	1	1	1

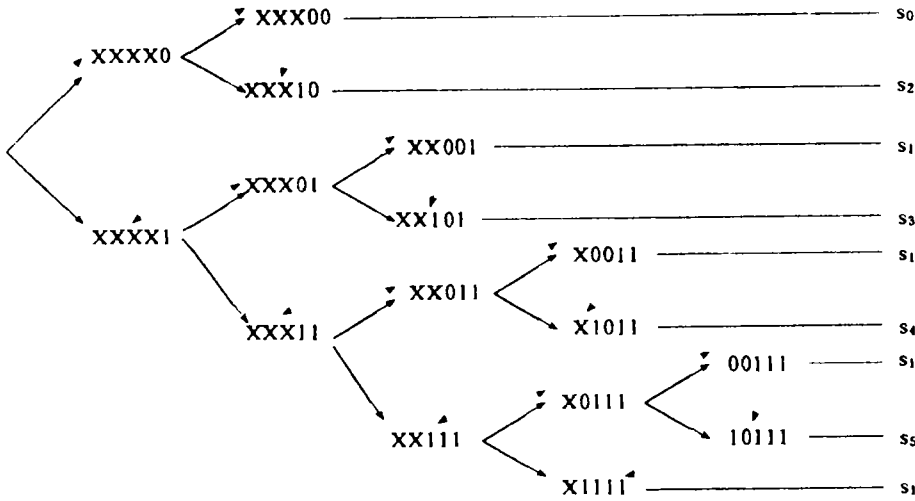


图5 “101110”序列检测器的分支图

由图 5 可以立即得到(3)式,从而进一步分析得到状态转换图和状态转换表如图 6 和表 7 所示. 读者可自己试试,如果不用分支图,该电路应如何设计.

$$\left\{ \begin{array}{l} s_0 = (XXX00); \\ s_1 = (XX001, X0011, 00111, X1111); \\ s_2 = (XXX10); \\ s_3 = (XX101); \\ s_4 = (X1011); \\ s_5 = (10111). \end{array} \right. \quad (3)$$

表 7

X_n Y_n	Y_{n-1}/Z_n	
	0	1
s_0	$s_0/0$	$s_1/0$
s_1	$s_2/0$	$s_1/0$
s_2	$s_0/0$	$s_3/0$
s_3	$s_2/0$	$s_4/0$
s_4	$s_2/0$	$s_5/0$
s_5	$s_2/1$	$s_1/0$

4 分支图法需比较的代码数

由上可见,利用分支图法,需要与表 4 比较的代码数大幅度地减少. 然而,要设计 n 位串行码检测器,究竟需要对多少个代码进行判断呢?

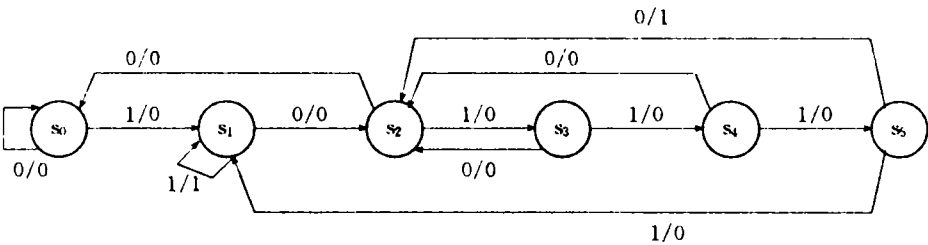


图6 “101110”序列检测器的状态转换图

分析分支图结构,可以看出在分支图中,对应于图中每一分支点,就要对两个代码进行比较判断,而最后得到的代码数比分支点的数目多一个.那么,有多少个分支点呢?

由于 n 位串行码检测器有 n 个状态,因此,得到至少 n 个代码(即每个状态至少一个代码),也即至少有 $n-1$ 个分支点.例如“11...1”序列检测器.

进一步分析可以得到, n 位串行码检测器最多有 s 个分支点,这里

$$s = \sum_{i=1}^{n-1} a_i.$$

(4)

上式中,

$$a_i = \{2^{i-1}, n-i\}_{\min},$$

显然,

$$S = \sum_{i=1}^{n-1} a_i < \sum_{i=1}^{n-1} (n-i) = \frac{(n-1)n}{2} < \frac{1}{2}n^2.$$

可见,分支图法需比较的代码数小于 n^2 .表 8 中列出了 n 位序列检测器采用分支图法和不采用分支图法需比较的代码数的一组数据.非常明显,由于采用了分支图法,工作量大为减少,由无穷高阶(指数形式)下降到二阶.加上分支图法非常规则,可以沿图 4 和图 5 中虚线箭头按深度优先的原则顺序处理,因而便于利用计算机辅助设计.

表. 8

n	需比较的代码数	
	不用分支图法	采用分支图法(下限/上限)
4	8	6/8
10	512	18/56
30	5.37×10^4	58/662
100	6.34×10^{19}	198/8718
300	1.02×10^{50}	598/85412

在本文结束时,特别要指出的是:分支图法不仅能用于设计序列检测器,也可以用来设计任何有输入信号的其它时序逻辑电路.所不同的是,若仅有一个输入变量,则采用有根二分支图;若有两个输入变量,则采用有根四分支图.一般情况下,若有 K 个输入变量,则采用有根 2^K 个分支图.这时,需比较判断的代码,则由 $(2^K)^{n-1}$ 个剧减到小于 $2^{K-1}n^2$ 个.例如,对于有根四分图,由 4^{n-1} 剧减到 2^n ,对简化的意义就更明显了.

参 考 文 献

1 顾德仁. 脉冲与数字电路(下). 北京:人民教育出版社,164~178

2 闫石. 数字电子技术基础(上). 北京:人民教育出版社,286~300

3 尤德斐. 数字化测量技术(上). 北京:机械工业出版社,227~245 (下转第 54 页)

从上述实例证明,用倒谱分析,反射信号明显,可进一步提高桩基的检测质量,防止桩身缺陷的漏判和误判.

参 考 文 献

- 1 汪风泉. 基础结构动态诊断. 南京:江苏科学技术出版社,1992
- 2 H. G. 纳特克. 时间序列和模态分析的理论与实践导论. 北京:电子工业出版社,1988

The Principle and Applications of the Cepstrum Analyses Method to the Judgement of the Pile body Quality

Lǚ Shaodi

Abstract After a brief review of the principle of the Cepstrum analyses, this paper throws its focus on applications of this method to the judgement of the pile integrity quality and three pile cepstrum are caculated. The result shows that the cepstrum method is better compared with the traditionally - used reflected wave method at some location. By using the cepstrum analysis, it is easily distinguish the reflected signals from the incident(signals).

Key words Cepstrum Analyses; Pile integrity quality

(上接第 49 页)

The Application of Branch Map in Sequential Logic Circuit Design

Qian Biyun

Abstract The design process of sequential logic circuit, which's one of the major difficulties is to seek the simplest state transform diagram, and how to define the transform diagram and the method of getting state transform diagram with branch map are discussed by instancing the sequence detector. Then the possibility of using computer aided design is also discussed. The amount of work would reduce from infinite order to 2 order by branch map, using the computer to seek state transform diagram. Finally it is pointed out that branch map method is also suitable for designing other sequential logic circuit which has input signal.

Key words Branch map, Sequential logic circuit; Design; Application