Part Ⅲ 触发器与时序逻辑*OK*

Lecture 10 同步时序逻辑电路设计 1

一般步骤：

①逻辑抽象：作出原始状态图和原始状态表；

②状态化简：求得最小化状态表；

③状态编码：得到二进制状态表；

④选定触发器的类型，并求出激励函数和输

出函数最简表达式；

⑤检查自启动能力；

⑥画出逻辑电路图。

一、逻辑抽象

原始状态图和原始状态表是对设计要求的最原始的抽象，建立正确的原始状态图和状态表是同步时序电路设计中最关键的一步。

建立原始状态图的一般思路：

1）确定电路模型

设计成 Mealy 型？Moore 型？

将电路设计成哪种模型？有的问题已由设计要求规定，有的问题可由设计者选择。不同的模型对应的电路结构不同，设计者在选择时，应根据问题中的信号形式、电路所需器件的多少等综合考

虑。

2）设立初始状态

时序逻辑电路在输入信号开始作用之前的状态称为**初始状态**。

在建立原始状态图时，应首先设立初始状态，然后从初始状态出发考虑在各种输入作用下的状态转移和输出响应。

3）根据需要记忆的信息增加新的状态

同步时序电路中状态数目的多少取决于需要记忆和区分的信息量。

一般来说，若在某个状态下出现的输入信号能用已有状态表示，则应转向已有状态。仅当某个状态下出现的输入信号不能用已有状态表示时，才令其转向新的状态。

4）确定各时刻电路的输出

在建立原始状态图时，必须确定各时刻的输出值。

在 Moore 型电路中，应指明每种状态下对应的输出；在 Mealy 型电路中应指明从每一个状态出发，在不同输入作用下的输出值。

注意：在描述一个逻辑问题的原始状态图和

原始状态表中，状态数目不一定能达到最少，这一点无关紧要，因为可以对它再进行状态化简。设计者应把清晰、正确地描述设计要求放在第一位。

【例 1】某**序列检测器**有一个输入端x和一个输出端z。

输入端x输入一串随机的二进制位，当输入序列中出现“011”时，输出z产生一个“1”输出，平时z输出“0”。

典型输入、输出序列如下：

输入x：1 0 1 0 1 1 1 0 0 1 1 0

输出z：0 0 0 0 0 1 0 0 0 0 1 0

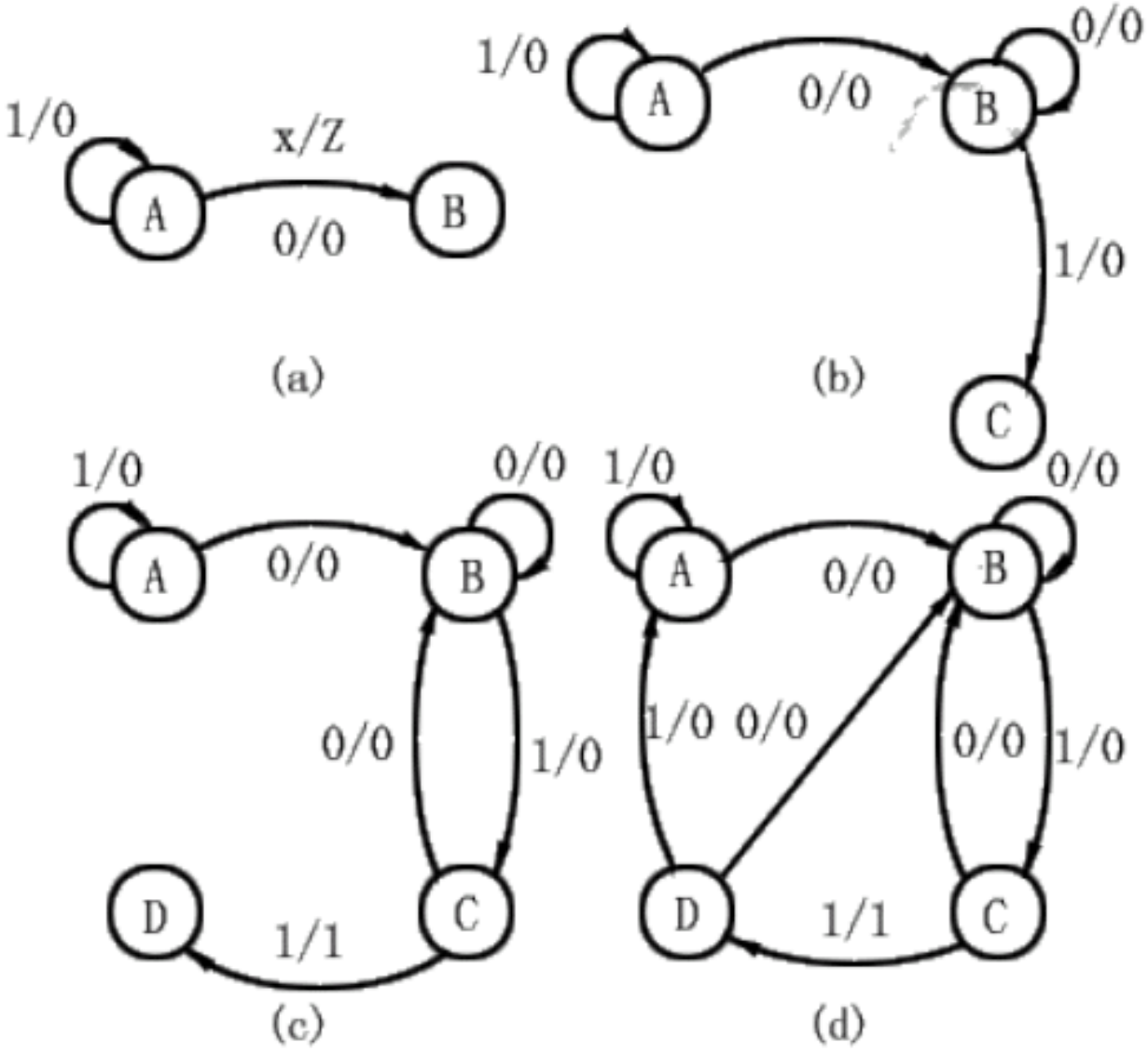
方案 1 采用 Mealy 型电路，设：

状态 A------电路的初始状态；

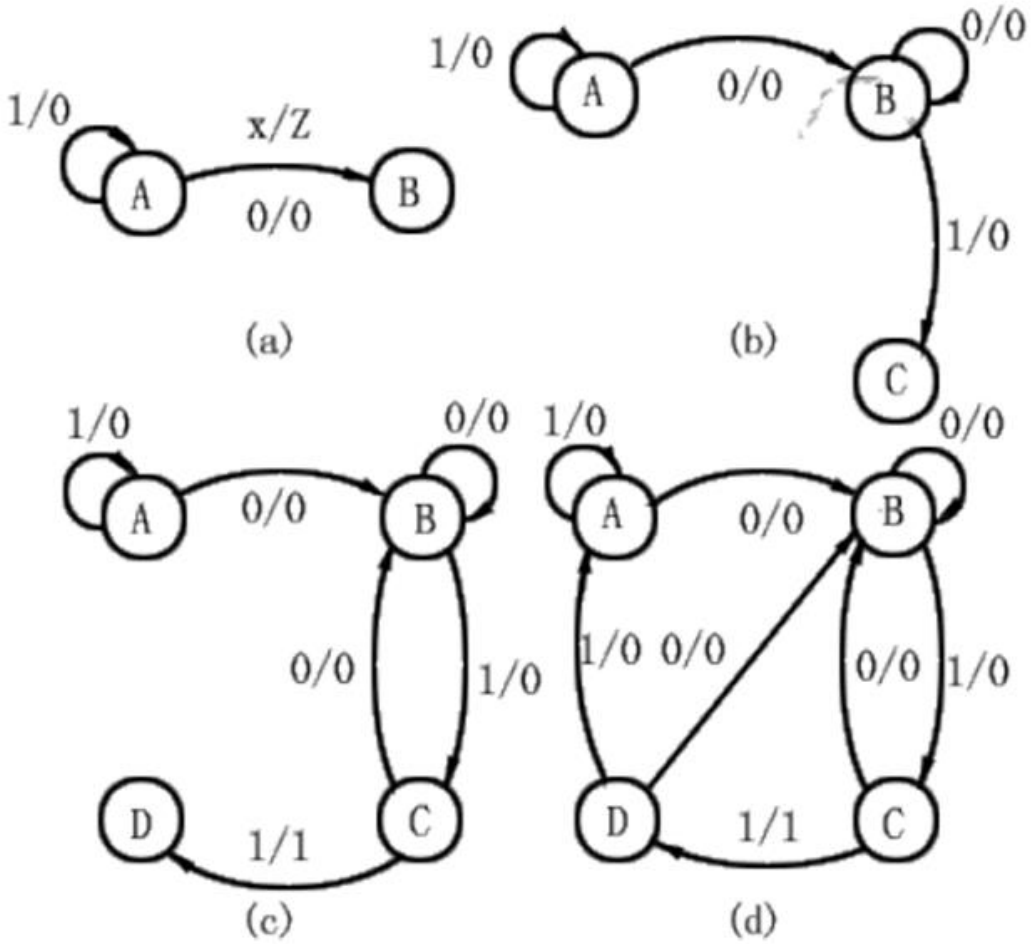
状态 B------表示收到了目标序列“011”中的第一位“0”；

状 态 C------ 表示收到了目标序列“011”中的前面两位“01”;

状 态 D------ 表示收到了完整的目标序列“011”。



原始状态图和原始状态表如下：



Mealy型状态表

现　态	次态/输出	
	x=0	x=1
A	B/0	A/0
B	B/0	C/0
C	B/0	D/1
D	B/0	A/0

方案 2 如果采用 Moore 型电路

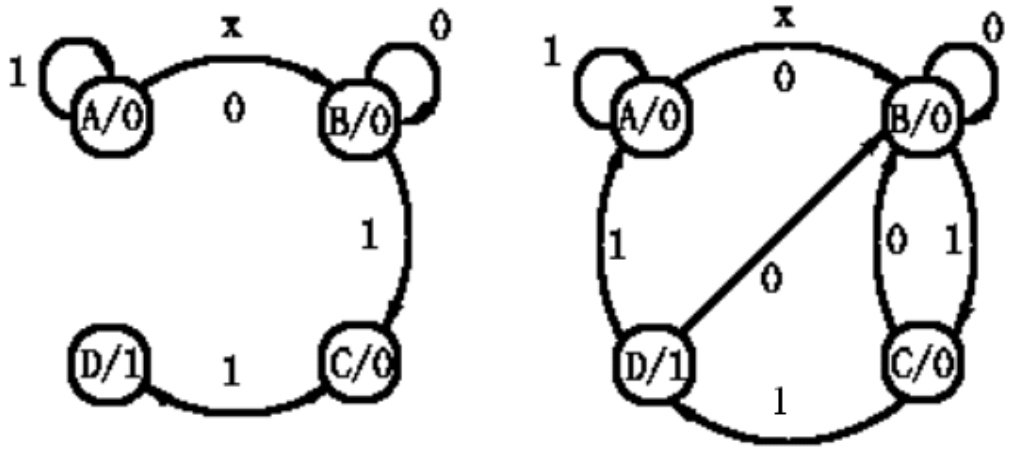
由于电路输出完全取决于状态，而与输入无直接联系。在作状态图时，应将输出标记在代表各状态的圆圈内。

设电路初始状态为 A，并用状态 B、C、D 分别表示收到了输入x送来的 0、01、011。

显然，仅当处于状态 D 时电路输出为 1，其他

状态下输出均为 0。

原始状态图和原始状态表如下：



Moore 型状态表

现 态	次 态		输 出
	x=0	x=1	Z
A	B	A	0
B	B	C	0
C	B	D	0
D	B	A	1

【例 2】模 5 加 1、加 2 **计数器**有一个输入x和一个输出z。

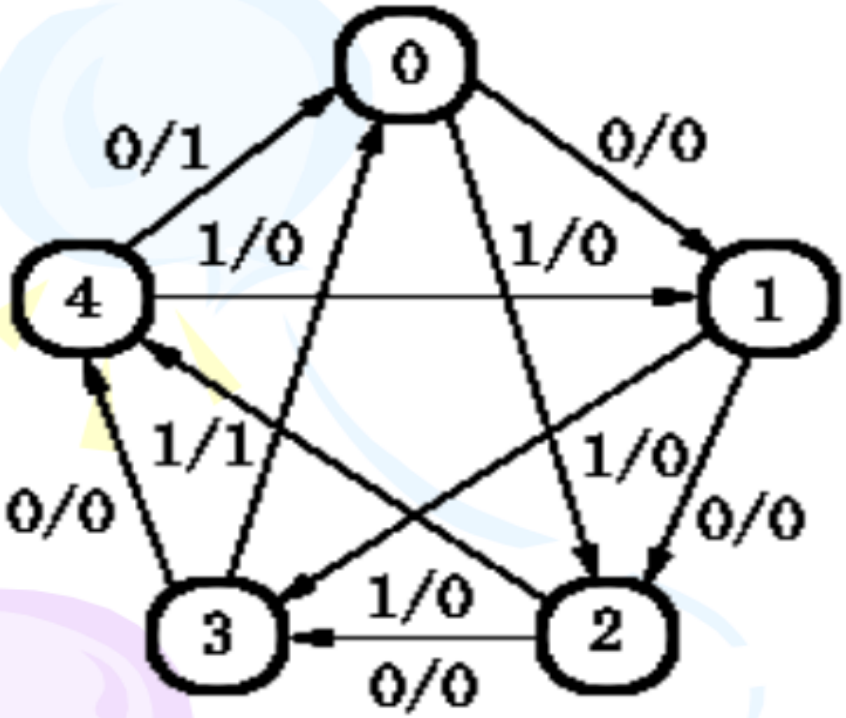
输入x为加 1、加 2 控制信号，当$x=0$时，计数器在时钟脉冲作用下进行加 1 计数；当$x=1$时，计数器在时钟脉冲作用下进行加 2 计数。

当电路计满 5 个状态后，输出z产生一个 1 信号作为进位输出，平时z输出为 0。

下面建立该计数器的 Mealy 型原始状态图和状态表。

假设模 5 计数器的 5 个状态分别用 0、1、2、3、4 表示，其中 0 为初始状态。

根据题意可作出原始状态图和原始状态表如下：



原始状态表

现　态	次态 /输出 Z	
	x=0	x=1
0	1/0	2/0
1	2/0	3/0
2	3/0	4/0
3	4/0	0/1
4	0/1	1/0

【例 3】设计一个**代码检测器**，该同步时序电路用于检测串行输入的 8421 码，其输入的顺序是先低位后高位，当出现非法数字（1010，1011，1100，1101，1110，1111）时，电路的输出为 1。

下面建立该时序电路的 Mealy 型原始状态图和状态表。

由于输入的 8421 码是先低位后高位，因此，判断输入是否为非法数字时，应从低位到高位查

看各位输入值。

设：状态 A 表示起始状态；

状态 B 和 C 分别表示最低一位代码的两种不同取值 0 和 1；

状态 D，E，F，G 分别表示低两位的码的四种不同取值 00~11；

状态 H、I、J、K、L、M、N、P 分别表示低三

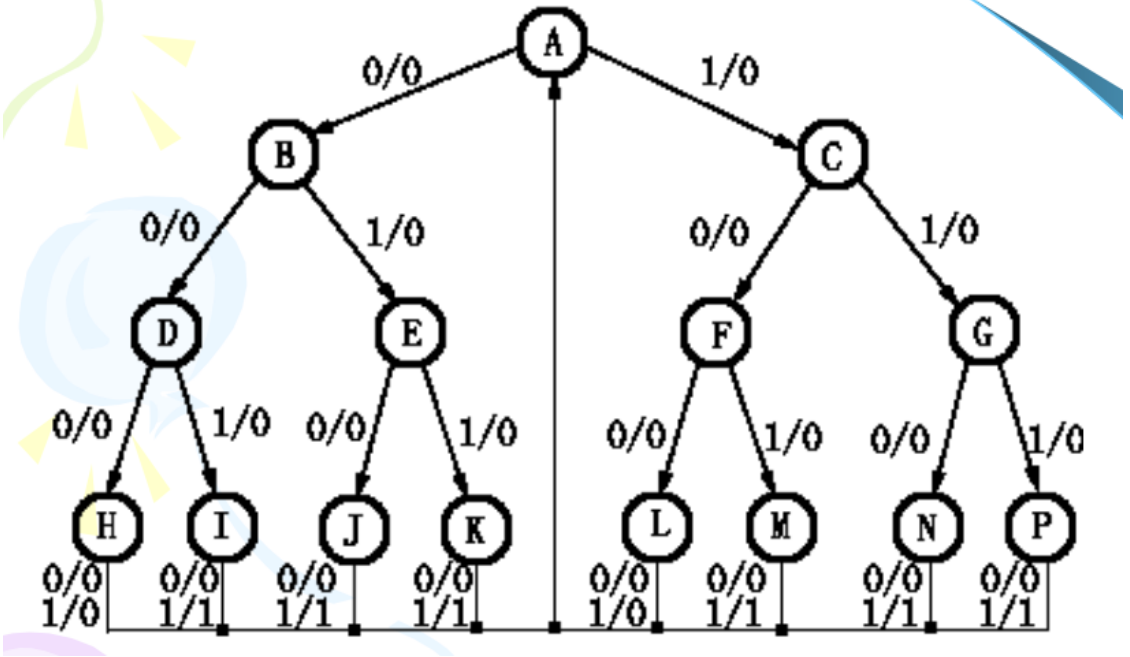
位代码的八种取值 000~111。

当x输入的第四位代码到来时，即可对输入代码进行判断，若出现非法数字，电路的输出为 1，否则为 0，并返回到起始状态 A。

根据以上分析，可以得到原始状态图和状态表如下：

Mealy模型原始状态图

现 态	次态 /输出		现 态	次态 /输出	
	x=0	x=1		x=0	x=1
A	B/0	C/0			
B	D/0	E/0	I	A/0	A/1
C	F/0	G/0	J	A/0	A/1
D	H/0	I/0	K	A/0	A/1
E	J/0	K/0	L	A/0	A/0
F	L/0	M/0	M	A/0	A/1
G	N/0	P/0	N	A/0	A/1
H	A/0	A/0	P	A/0	A/1



注意：图中，当 4 位代码检测完后，应转向初始状态 A，以便检查下一组代码。

思考：代码检测器与序列检测器的主要区别是什么？

二、状态化简

状态化简：是指采用某种化简技术从原始状态表中消去多余状态，得到一个既能正确地描述给定的逻辑功能，又能使所包含的状态数目达到最少的状态表，通常称这种状态表为最小化状态表。

目的：简化电路结构。状态数目的多少直接决定电路中所需触发器数目的多少。

设状态数目为n，所需触发器数目为m，则应满足如下关系：$2^m \geq n > 2^{m-1}$。

为了降低电路的复杂性和电路成本，应尽可能状态表中包含的状态数达到最少。

方法：隐含表法。

1、基本概念

等效状态：设状态S_i和S_j是状态表中的两个状

态，若对于所有可能的输入序列，分别从状态S_i和状态S_j出发，所得到的输出响应序列完全相同，则状态S_i和S_j是等效的，记作(S_i, S_j)，又称状态S_i和S_j为**等效对**。

判断方法：若状态S_i和S_j是原始状态表中的两个现态，则S_i和S_j等效的条件可归纳为在输入的各种取值组合下满足如下两条：

第一，输出相同；

第二，次态属于下列情况之一：

a. **次态相同**；

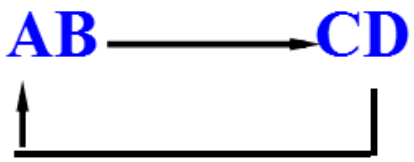
b. **次态交错**或为各自的现态；

c. **次态循环**或为等效对。

问题：什么叫交错、循环呢？

例如，在下表中：当$X=0$时，现态 A、B 的次态相同，现态 C、D 的次态交错；当$X=1$时，现态 A、B 的次态为 C、D，而现态 C、D 的次态为 A、B，构成次态循环，即

现态	次态/输出	
	X=0	X=1
A	B/0	C/1
B	B/0	D/1
C	D/0	A/0
D	C/0	B/0



性质：等效状态具有**传递性**，即假若S_1和S_2等效，S_2和S_3等效，那么，一定有S_1和S_3等效。

等效类：由若干彼此等效的状态构成的集合。在同一个等效类中的任意两个状态都是等效的。

例如：由(S_1,S_2)和(S_2,S_3)可以推出(S_1,S_3)，进而可知S_1、S_2、S_3属于同一等效类，记作$\{S_1,S_2,S_3\}$。

等效类是一个广义的概念，两个状态或多个

状态均可以组成一个等效类，甚至一个状态也可以称为等效类，因为任何状态和它自身必然是等效的。

最大等效类：是指不被任何别的等效类所包含的等效类。

注意：这里所指的最大，并不是指包含的状态最多，而是指它的独立性，即使是一个状态，只要它不被包含在别的等效类中，也是最大等效类。换

而言之，如果一个等效类不是任何其他等效类的子集，则该等效类为最大等效类。

说明：状态等效是状态之间的一种**等价关系**！

原始状态表的化简过程，就是寻找出表中的所有最大等效类，然后将每个最大等效类中的所有状态合并为一个新的状态，从而得到**最小化状态表**。

简化后的状态数等于最大等效类的个数！

2、状态化简

一般步骤：

①作隐含表

隐含表是一个直角三角形阶梯网格，表中每个方格代表一个状态对。

②寻找等效对

顺序比较：按照隐含表中从上至下、从左至右的顺序，对照原始状态表依次对所有“状态对”进行逐一检查和比较，并将检查结果标注在隐含表中的相应方格内。

比较结果标注如下：

等效 ------- 在相应方格内填上“∨”；

不等效----- 在相应方格内填上“×”；

与其他状态对相关 ---- 在相应方格内填上相关的状态对。

关联比较：指对那些在顺序比较时尚未确定是否等效的状态对作进一步检查，直到判别出状态对等效或不等效为止。

③求出最大等效类

在找出原始状态表中的所有等效对之后，可

利用等效状态的传递性，求出各最大等效类。

④状态合并，作出最小化状态表

将每个最大等效类中的全部状态合并为一个状态，即可得到和原始状态表等价的最小化状态表。

【例 1】化简下表所示原始状态表：

原始状态表

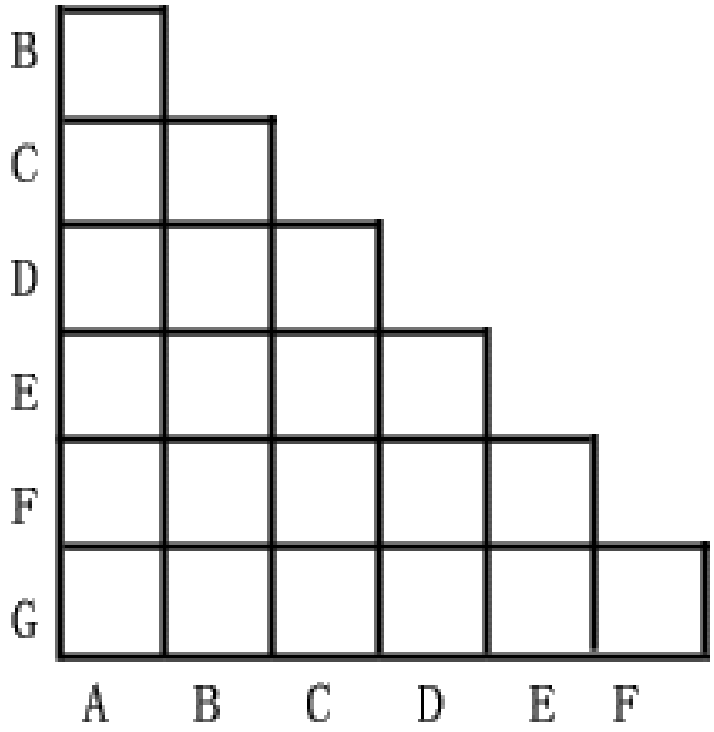
现　态	次态 / 输出	
	x=0	x=1
A	C/0	B/1
B	F/0	A/1
C	F/0	G/0
D	D/1	E/0
E	C/0	E/1
F	C/0	G/0
G	C/1	D/0

①作隐含表

给定原始状态表具有 7 个状态，根据画隐含表的规则，可画出隐含表框架如下：

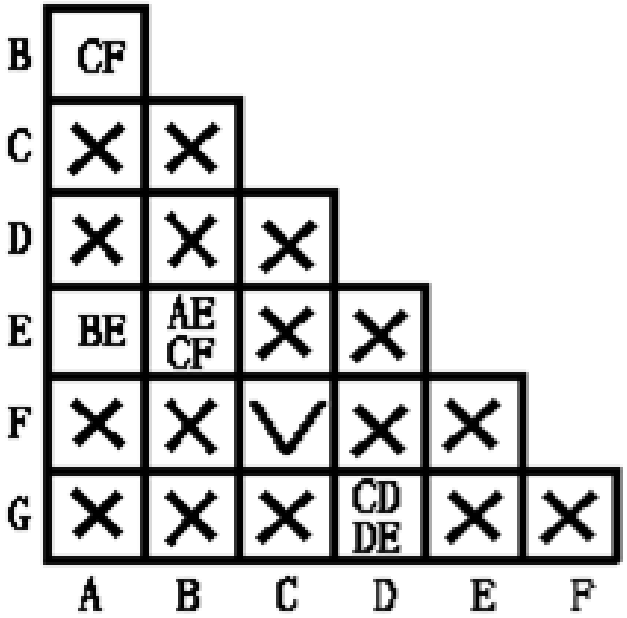
原始状态表

现　态	次态 / 输出	
	x=0	x=1
A	C/0	B/1
B	F/0	A/1
C	F/0	G/0
D	D/1	E/0
E	C/0	E/1
F	C/0	G/0
G	C/1	D/0



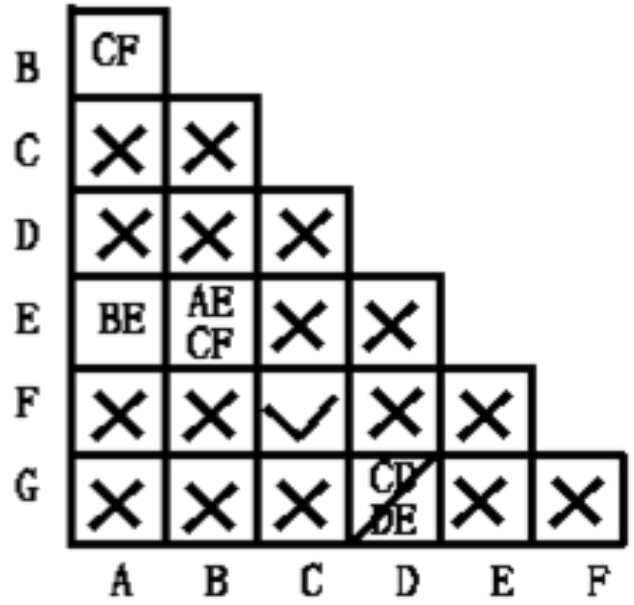
原始状态表

现　态	次态 / 输出	
	x=0	x=1
A	C/0	B/1
B	F/0	A/1
C	F/0	G/0
D	D/1	E/0
E	C/0	E/1
F	C/0	G/0
G	C/1	D/0

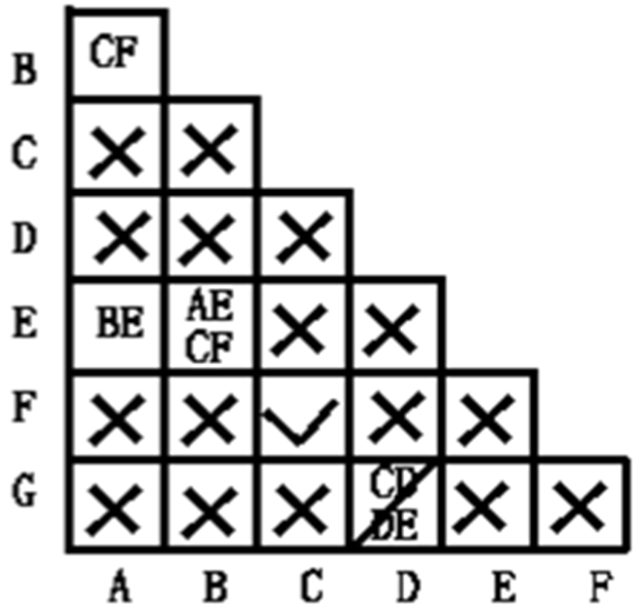


原始状态表

现　态	次态 / 输出	
	x=0	x=1
A	C/0	B/1
B	F/0	A/1
C	F/0	G/0
D	D/1	E/0
E	C/0	E/1
F	C/0	G/0
G	C/1	D/0



②寻找等效对



根据等效状态的判断标准，依次检查每个状态对，可得到顺序比较结果如上图（左）和关联比较结果如上图（右）。

由判断结果可知，原始状态表中的 7 个状态共有四个等效对：(A,B)、(A,E)、(B,E)、(C,F)

③求出最大等效类

由所得到的等效对和最大等效类的定义可知，原始状态表中的7个状态共构成四个最大等效类：

$\{A,B,E\}$、$\{C,F\}$、$\{D\}$、$\{G\}$

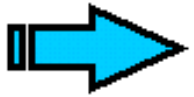
④状态合并，作出最小化状态表

令 $\{A,B,E\}\to a$ 、 $\{C,F\}\to b$ 、 $\{D\}\to c$ 、$\{G\}\to d$，并代入原始状态表中，即可得到化简后

的状态表如下边右表：

原始状态表

现　态	次态 /输出	
	x=0	x=1
A	C/0	B/1
B	F/0	A/1
C	F/0	G/0
D	D/1	E/0
E	C/0	E/1
F	C/0	G/0
G	C/1	D/0



最小化状态表

现　态	次态 /输出	
	x=0	x=1
a	b/0	a/1
b	b/0	d/0
c	c/1	a/0
d	b/1	c/0

$\{A,B,E\}\to a$、$\{C,F\}\to b$、$\{D\}\to c$、$\{G\}\to d$

三、状态编码

状态编码：也称状态分配，是指给最小化状态表中用字母或数字表示的状态，指定一个二进制代码，形成**二进制状态表**。

状态编码的任务是：

①确定状态编码的长度（即二进制代码的位数，或者说所需触发器个数）；

②寻找一种最佳的或接近最佳的状态分配方案，以便使所设计的时序电路最简单。

1、确定二进制代码的位数

二进制代码的位数是由最小化状态表中的状态个数来确定的。

设最小化状态表的状态数为 N，状态编码的长度为 m，则状态数 N 与状态编码长度 m 的关系为

$2^{m-1} < N \leq 2^m$。

例如：若某状态表的状态数 N=7，则状态分配时，二进制代码的位数应为 m=3，或者说状态变量个数为 3。

2、确定状态分配方案

状态与代码之间的对应关系可以有许多种。一般说来，用 m 位二进制代码的2^m种组合来对 N

个状态进行分配时，可能出现的状态分配方案数K_s为：$K_s = P_{2^m}^N$。

例如：当 N = 4，m= 2 时，$K_s = P_{2^2}^4 = 24$

随着状态数目的增加，分配方案的数目急剧增加。

问题：如何从众多的分配方案中寻找出一种最佳方案？

在实际工作中，工程技术人员通常按照一定的原则、凭借设计的经验去寻找相对最佳的编码方案。

一种常用方法称为**相邻分配法**，其基本思想是：在选择状态编码时，尽可能使激励函数和输出函数在卡诺图上的“1”方格处在相邻位置，从而有利于激励函数和输出函数的化简。

相邻分配法的状态编码**原则**如下：

①**次态相同，现态相邻**。在相同输入条件下，具有相同次态的现态应尽可能分配相邻的二进制代码；

②**同一现态，次态相邻**。在相邻输入条件下，同一现态的次态应尽可能分配相邻的二进制代码；

③**输出相同，现态相邻**。在每一种输入取值下

均具有相同输出的现态应尽可能分配相邻的二进制代码。

某些状态表常常出现不能同时满足 3 条原则的情况。此时，**可按从①至③的优先顺序考虑。**

此外，从电路实际工作状态考虑，一般将初始状态分配“0”状态。

【例如】对如下状态表进行状态编码（设 A 为初始状态）。

现态	次态/输出	
	x=0	x=1
A	C/1	B/0
B	A/0	A/1
C	A/1	D/1
D	D/1	C/0

所示状态表中，状态数 N = 4，故状态编码的长度应为 m=2。即实现该状态表的功能需要两个触发器。

根据相邻法的编码原则，4 个状态的相邻关系如下：

现态	次态/输出	
	x=0	x=1
A	C/1	B/0
B	A/0	A/1
C	A/1	D/1
D	D/1	C/0

根据原则①，状态 B 和 C 应分配相邻的二进制代码；

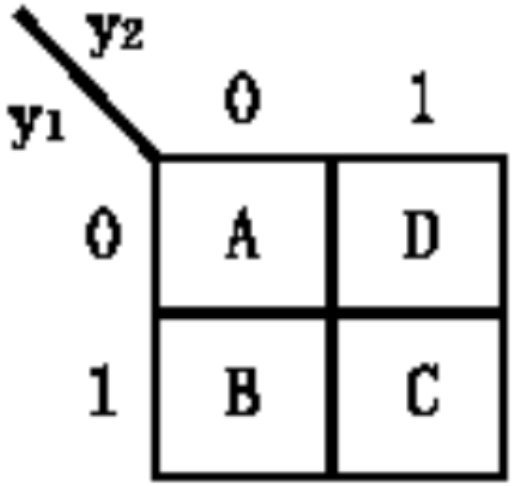
根据原则②，状态 B 和 C、A 和 D、C 和 D 应分配相邻的二进制代码；

根据原则③，状态 A 和 D 应分配相邻的二进制代码。

综合①—③可知，状态分配时要求满足 B 和 C、A 和 D、C 和 D 相邻。

在进行状态分配时，为了使状态之间的相邻关系一目了然，通常将**卡诺图**作为状态分配的工具。

假定状态变量用y_2y_1表示，并将 A 分配“0”，一种满足上述相邻关系的分配方案如下：



现态	次态/输出	
	x=0	x=1
A	C/1	B/0
B	A/0	A/1
C	A/1	D/1
D	D/1	C/0

现态 y_2y_1	次态$y_2^{(n+1)}y_1^{(n+1)}$/输出	
	x=0	x=1
00	11/0	01/0
01	00/0	00/1
11	00/1	10/1
10	10/0	11/0

状态 A、B、C、D 的状态编码依次为y_2y_1的取值00、01、11、10。

将状态表中的状态 A、B、C、D 分别用编码 00、01、11、10 代替，即可得到该状态表的二进制状

态表如右上表所示。

注意：满足分配原则的方案通常可以有多种，设计者可从中任选一种。