数字逻辑

第四章 同步时序电路

学习要求：

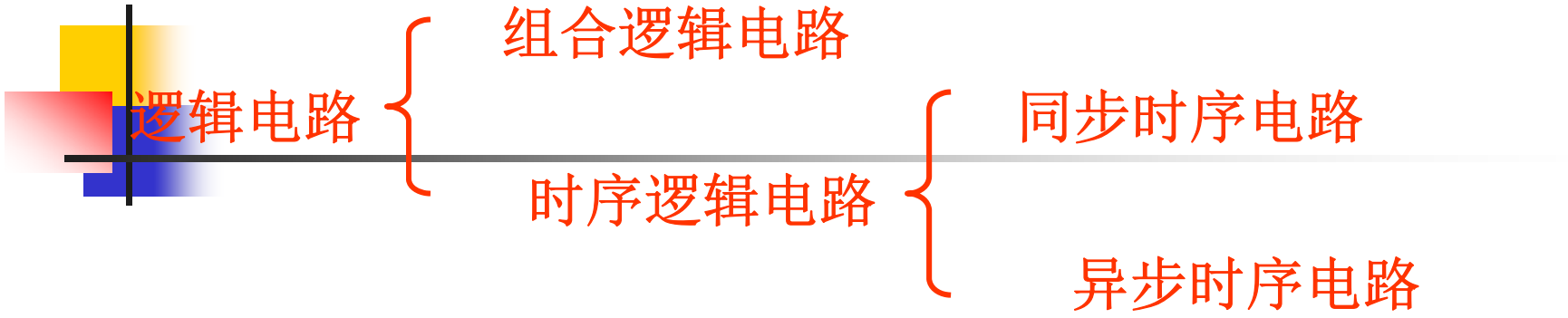
- 了解时序电路的基本结构、分类和常用的描述方法；
- 熟悉各种触发器的功能和使用；
- 熟练掌握同步时序电路分析和设计的基本方法；
- 熟悉状态图的建立，状态简化和状态分配的各个重要环节。

数字系统模型概述 （重点）

对任何一个数字系统来讲，它的组成结构只有两种模式，一种是组合逻辑模式，另一种就是时序逻辑模式。因此，无论多么复杂的数字系统，实际就是这两种模式的综合。

数字系统 { 组合逻辑
时序逻辑

逻辑电路的分类：



组合逻辑电路特点：在任何时刻电路产生的稳定输出信号仅与该时刻电路的输入信号有关。

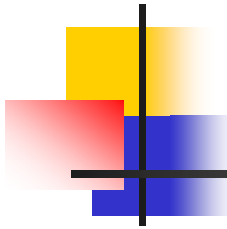
时序逻辑电路特点：在任何时刻电路的稳定输出信号不仅与该时刻电路的输入信号有关，而且与该电路过去的输入有关，或者说，某时刻电路的稳定输出与该时刻的输入和电路的状态有关。

在组合逻辑中，对组合逻辑网络来讲，由于它的输出不受到过去发生事件的影响，所以它必须满足两个条件：

1) 输入信号的“真 – 假”值，只能取决于当前输入信号的
“真 – 假”值；

2) 从输入到输出不能有反馈作用。

由于这两个条件的存在，如果输入条件一旦发生变化，组合逻辑网络的输出端将出现新值，这些新值出现的时间是由产生输出信号的组合网络所固有时延所决定的，这样输出变化将滞后于输入的变化。所以，在一定时间内输出可能与输入所需要的结果不一致。



若输入信号在一个足够长的时间内保持不变，这些信号最后将使输出成为正常。另外，因组合逻辑网络不具有反馈作用，就保证了输入信号的每次变化，只能引起输出信号的单一变化，不可能引起组合逻辑网络的不稳定性和不确定性。

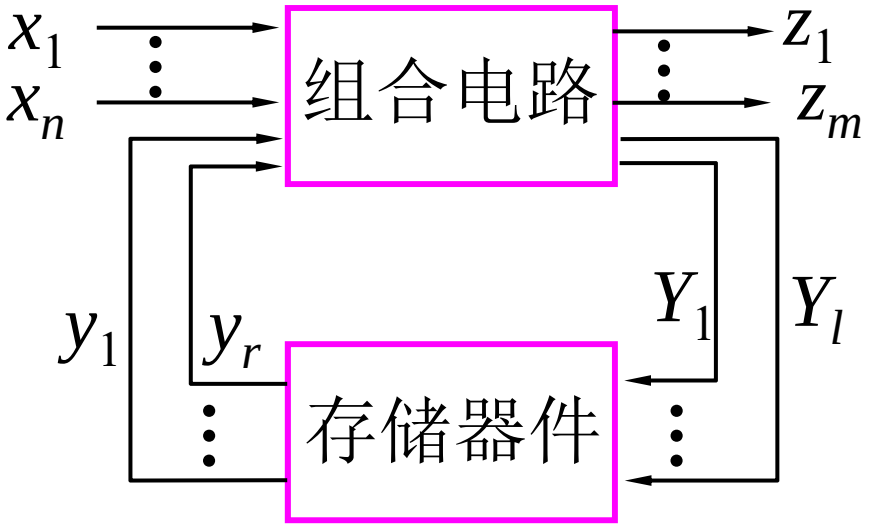
正因为这一特点，在组合逻辑设计中，为设计者带来很多方便，设计时只需要将2^n个不同输入组合值所对应的m个输出函数值列成一张表格，然后根据这张表格就可以进行组合逻辑网络的设计了。

4.1 时序逻辑电路 结构与类型

时序电路：一个电路在任何时刻的稳定输出不仅与该时刻电路的输入信号有关，而且与该电路过去的输入有关，这样的电路称为"时序电路"。

时序电路由组合电路和存储(记忆)器件及反馈回路三部分组成，见下图(b)

(a)



(b)

$x_1 \ldots x_n$：时序电路的输入或外部输入；

$z_1 \ldots z_m$：时序电路的输出或外部输出；

$y_1 \ldots y_r$：时序电路的状态或内部输入；

$Y_1 \ldots Y_l$：时序电路的激励或内部输出；

状态：过去的输入已不存在，但可以通过存储器件把它们记录下来，称之为状态。记录下来的信息可能和过去的输入完全一样，也可能是经过了组合电路加工处理后的结果。我们把某一时刻之前的状态称为"现态"，把这一时刻之后的状态称为"次态"，"现态"和"次态"是一个相对的概念，分别用$y^{(n)}$(或y)和$y^{(n+1)}$表示。

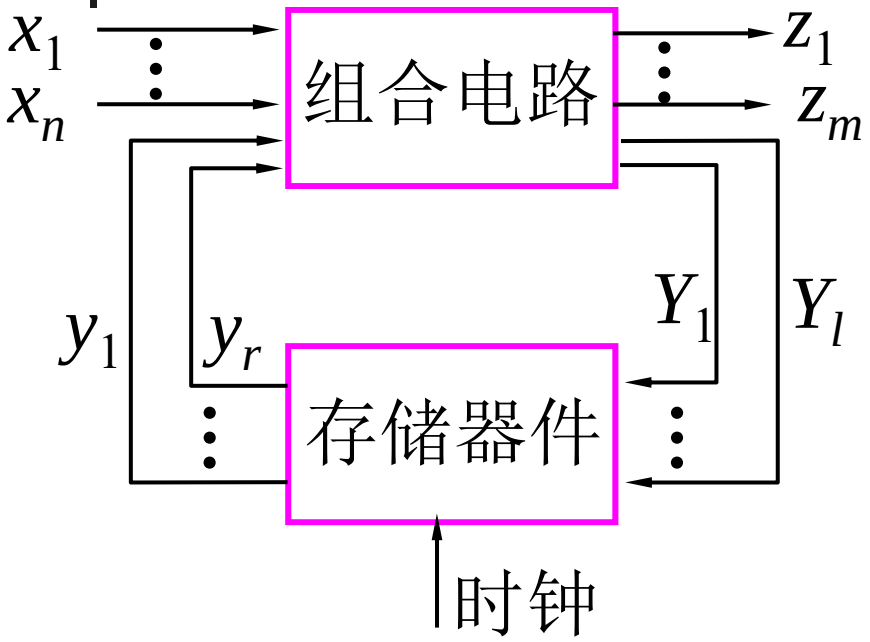
时序电路的逻辑函数由3组方程组成：

$$Z_i = f_i(x_1, \ldots, x_n; y_1, \ldots, y_r), i = 1, \ldots, m$$

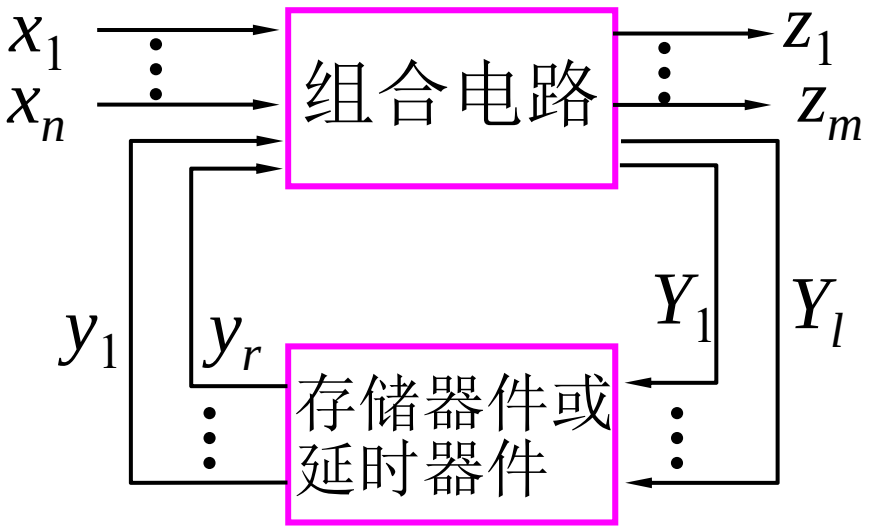
$$Y_j = g_j(x_1, \ldots, x_n; y_1, \ldots, y_r), j = 1, \ldots, l$$

$$y^{(n+1)} = h_k(y_1, \ldots, y_r; Y_1, \ldots, Y_r), k = 1, \ldots, r$$

两种时序电路类型见下图：

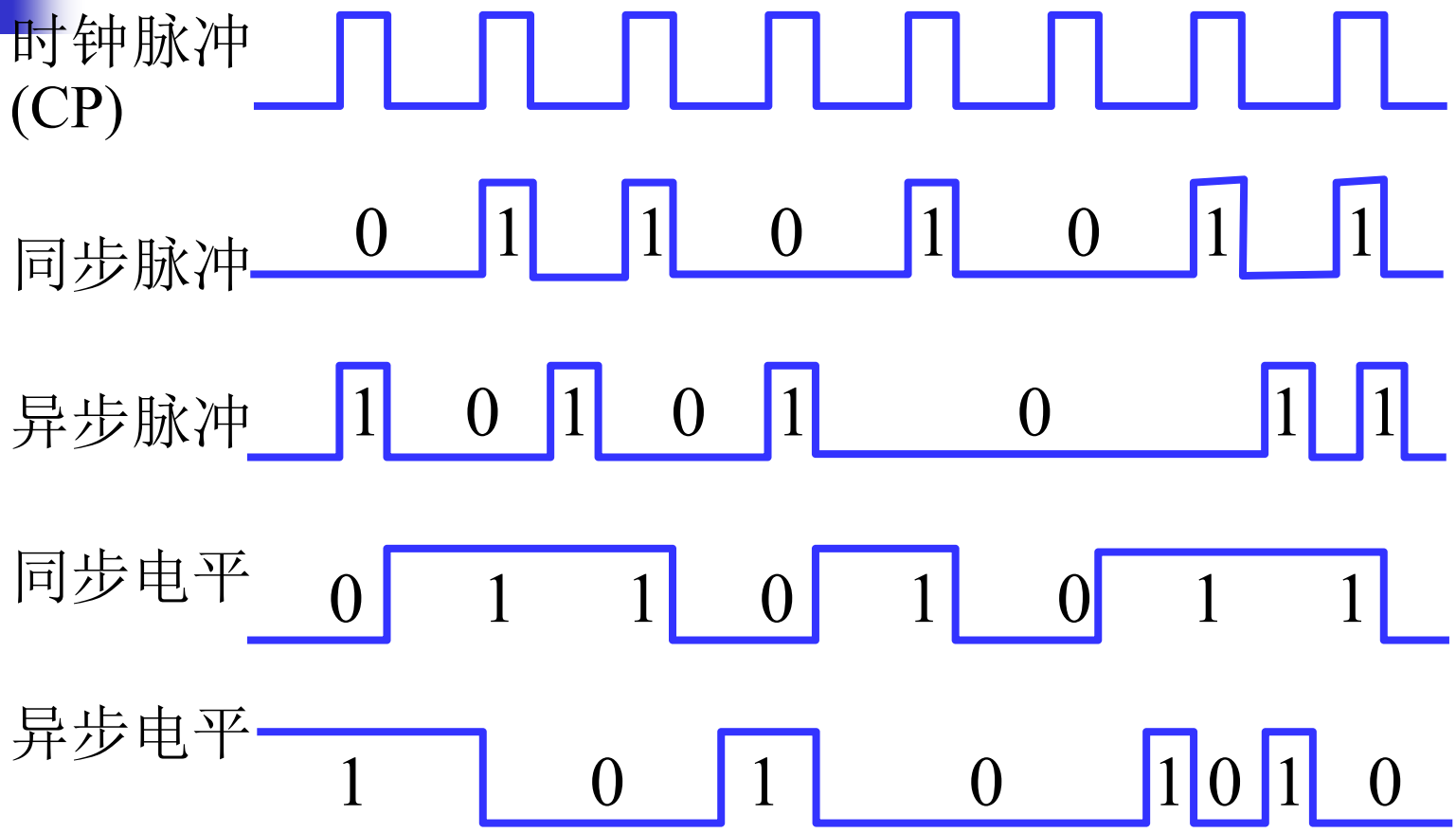


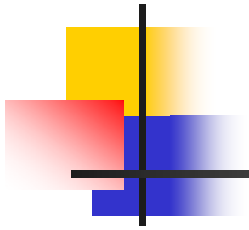
(a) 同步时序电路



(b) 异步时序电路

时序电路输入信号的波形图：





4.2 状态表和状态图

状态表与状态图是用来表示同步时序电路的输入、输出、现态、次态之间转移关系的两种常用工具。

4.2.1 Mealy 型状态表和状态图

如果同步时序电路的输出是输入和现态的函数，即$Z_i = f_i(x_1,\ldots, x_n; y_1,\ldots, y_r)$，$i=1,\ldots, m$，则称该电路为Mealy型电路。

一、状态表

Mealy 型电路状态表

现态	次态 / 输出		
		输入x	
y		$y^{(n+1)}/Z$	

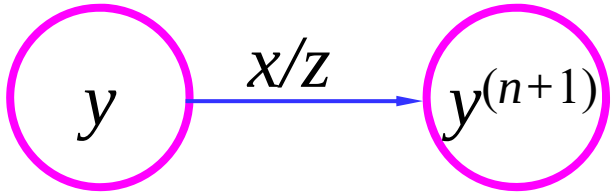
该表表明：处在状态y的同步时序电路，当输入为x时，输出为z，且在时钟脉冲作用下，电路进入次态$y^{(n+1)}$。

某Mealy 型电路状态表

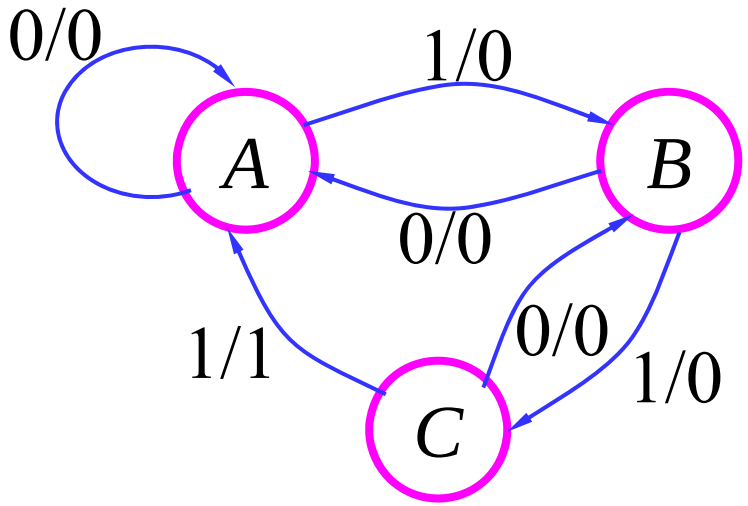
现 态 y	次态 / 输出$(y^{(n+1)}/Z)$	
	$x = 0$	$x = 1$
A	$A/0$	$B/0$
B	$A/0$	$C/0$
C	$B/0$	$A/1$

二、状态图

状态图是一种反映同步时序电路状态转移规律和输入、输出取值关系的有向图。



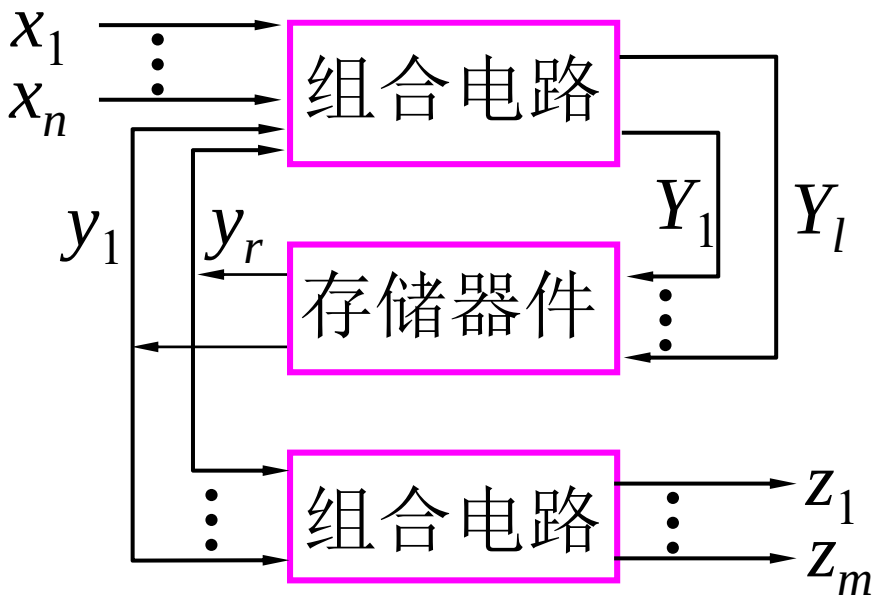
Mealy型电路状态图



某Mealy型电路状态图

4.2.2 Moore 型状态表和状态图

如果同步时序电路的输出仅是现态的函数，即$Z_i = f_i(y_1, \ldots, y_r)$, $i=1, \ldots, m$，则称电路为Morre型电路。它的电路结构图可表示为:



一、状态表

Moore 型电路状态表

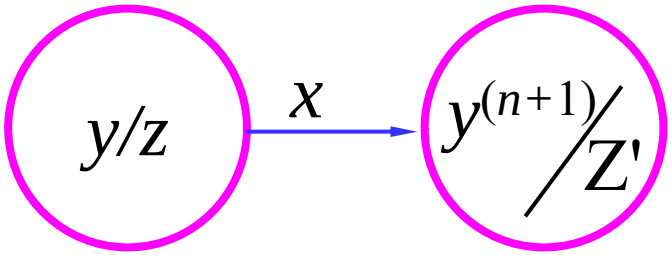
现 态	次 态		输 出
	输入x		
y	$y^{(n+1)}$		Z

该表表明：当电路处于状态y时，输出为z，若输入为x，则在时钟脉冲作用下，电路进入次态$y^{(n+1)}$。

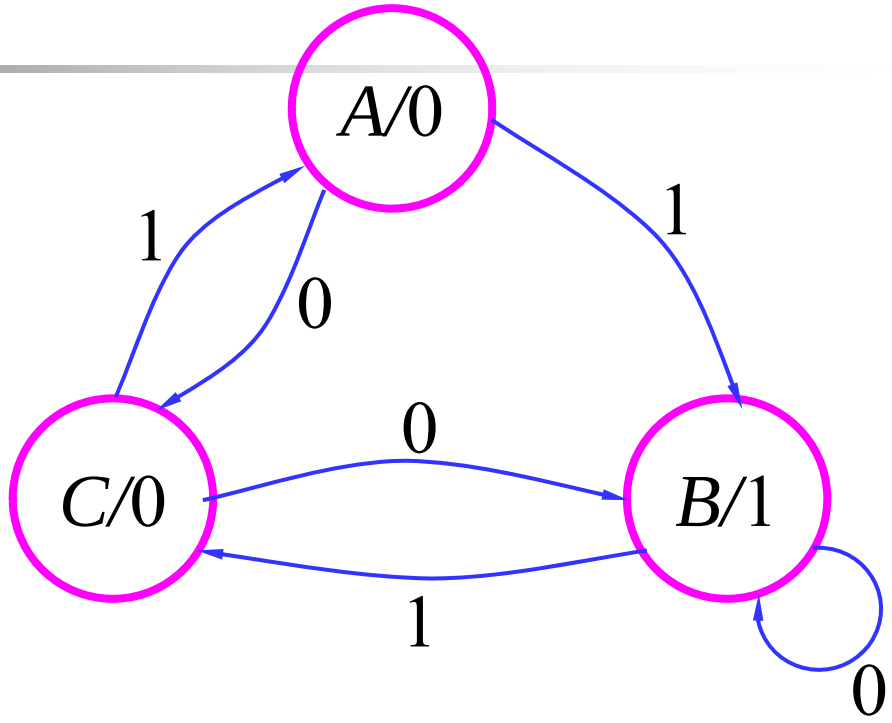
某Moore 型电路状态表

现 态 y	次 态 $y^{(n+1)}$		输 出 Z
	$x=0$	$x=1$	
A	C	B	0
B	B	C	1
C	B	A	0

二、状态图

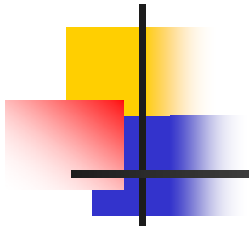


Moore型电路状态图



某Moore型电路状态图

4.3 触发器



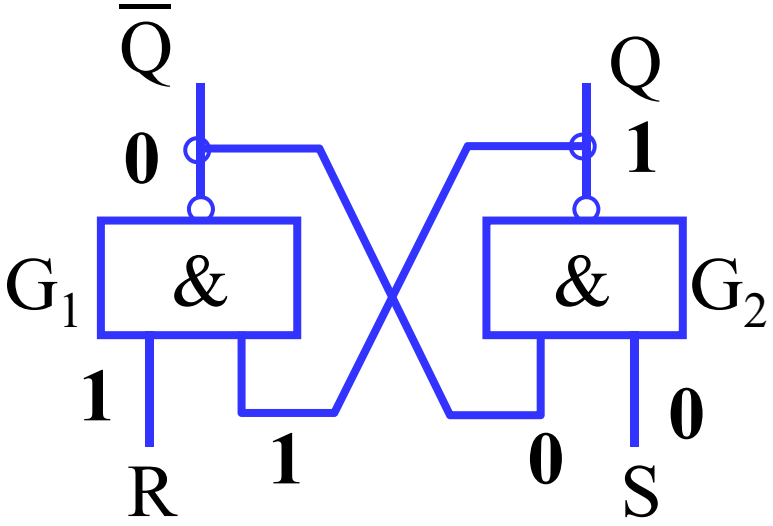
触发器是一种存储元件，在电路中用来"记忆"电路过去的输入情况。

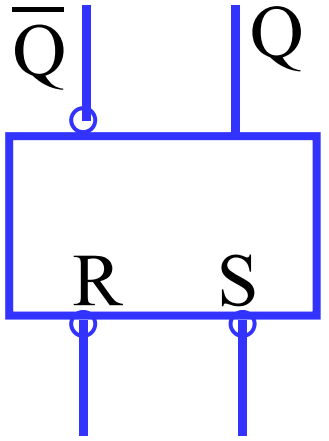
一个触发器具有两种稳定的状态,一个称之为 "0"状态，另一种称之为"1"状态。在任何时刻,触发器只处于一个稳定状态,当触发脉冲作用时,触发器可以从一种状态翻转到另一种状态。

常用的触发器有R–S触发器, D触发器J – K触发器和T触发器。

4.3.1 基本R－S触发器

基本R－S触发器可 由两个"与非"门交叉耦合组成，其逻辑图和逻辑符号如下：





基本R－S触发器的输入与状态之间的逻辑关系可用触发器的功能表来描述。

基本R－S触发器功能表

R	S	$Q^{(n+1)}$	功能说明
0	0	d	不定
0	1	0	置 0
1	0	1	置 1
1	1	Q	不变

基本R－S触发器的次态方程为：

$$Q^{(n+1)}=\overline{S}+RQ$$

其约束条件为：

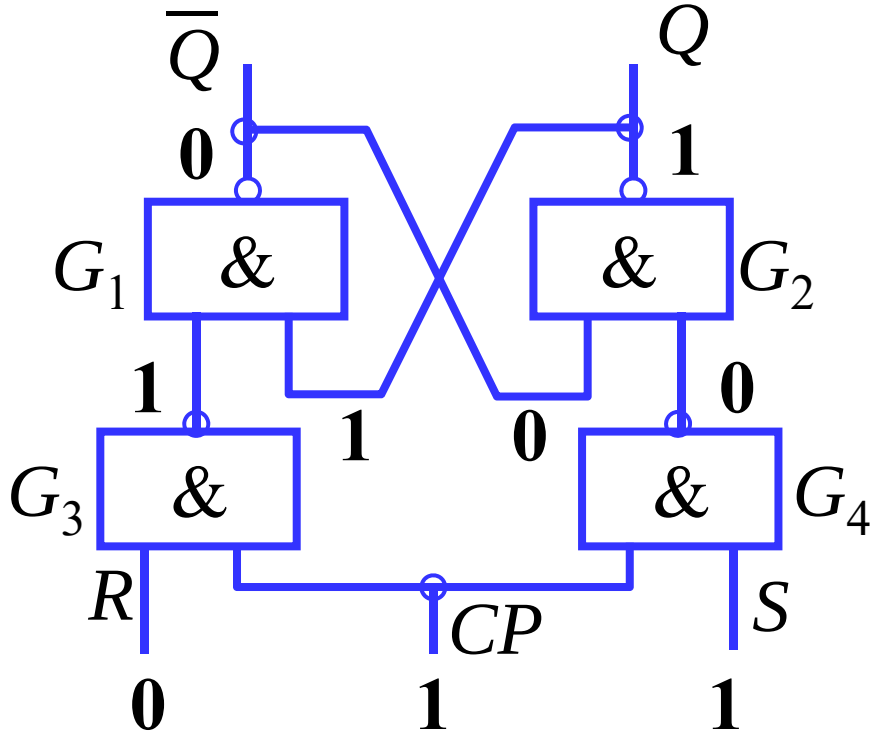
$$R+S=1$$

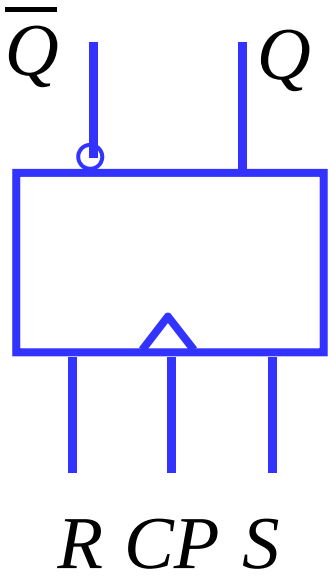
基本R－S触发器的一个重要特性：如果连续出现多个置0或置1信号，只有第一个置0或置1信号起作用。利用这一特性可消除机械开关的触点抖动。

基本R－S触发器也可由"或非"门组成。

4.3.2 时钟控制*R*—*S*触发器

在数字系统中，通常要求触发器按一定的时间节拍动作，即让输入信号的作用受到时钟脉冲的控制，为此出现了带时钟控制的*R*—*S*触发器，其逻辑图和逻辑符号如下：





当CP为0时，不论R，S为何值，触发器的状态保持不变；当CP为1时，触发器的状态取取决于R和S，工作原理与R－S触发器相同。

钟控触发器的功能表和状态表如下：

钟控R－S触发器功能表

R	S	$Q^{(n+1)}$	功能说明
0	0	Q	不变
0	1	1	置 1
1	0	0	置 0
1	1	d	不定

钟控R－S触发器状态表

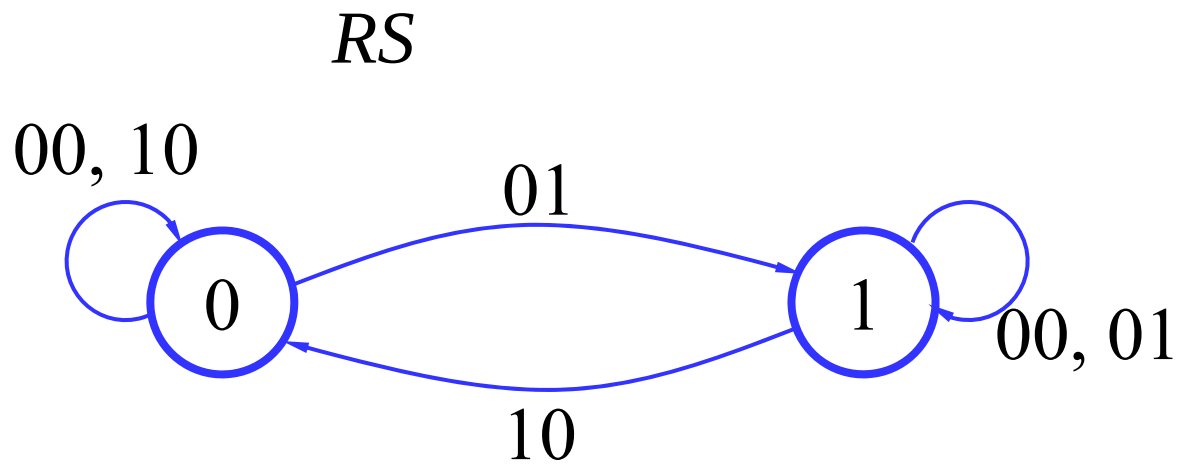
现 态	次 态 $Q^{(n+1)}$			
Q	RS=00	0 1	1 1	1 0
0	0	1	d	0
1	0	1	d	0

钟控R－S触发器的状态方程为：

$$Q^{(n+1)}=S+\overline{R}Q$$

$$R\cdot S=0 \quad \text{(约束条件)}$$

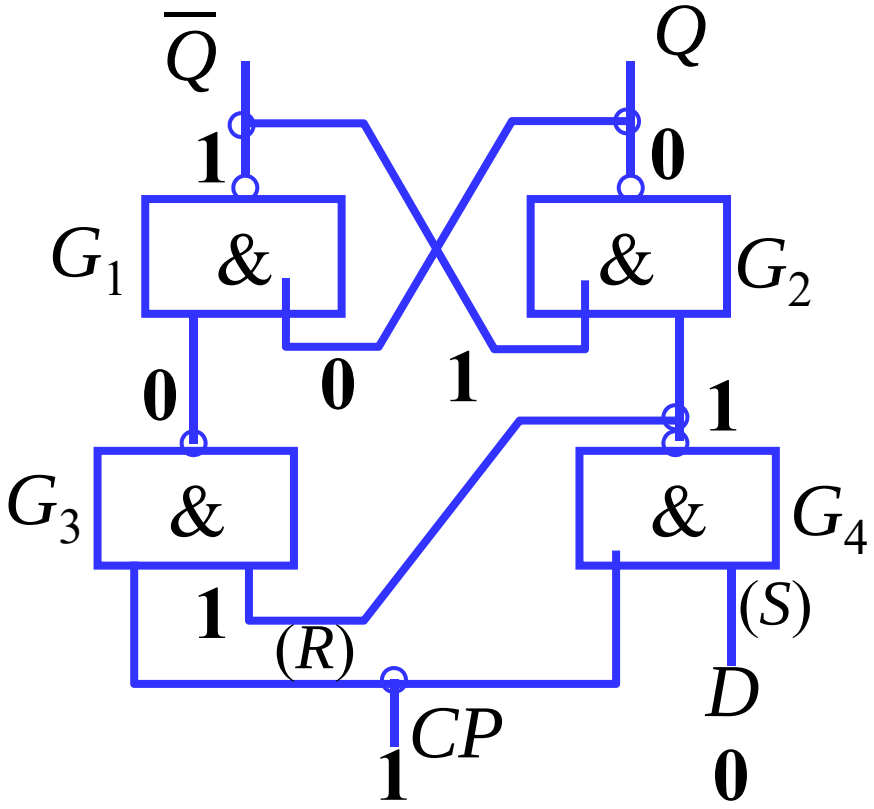
状态图为：

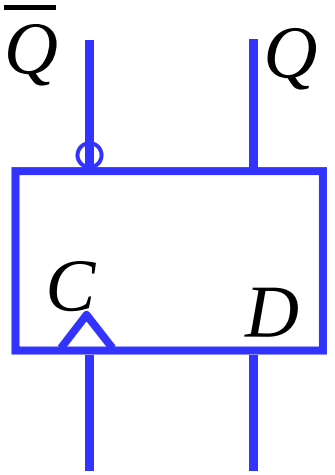


钟控R－S触发器存在次态不能确定和"空翻"两个问题。

4.3.3　D触发器

如果使钟控R－S触发器的R和S端始终处于互补状态, 则可消去次态不能确定的问题, 这就形成了所谓的D触发器, 其逻辑图的逻辑符号如下:





当CP=0时，D触发器的状态不变；当CP=1时，D触发器的状态取决于D。

D触发器的功能表和状态表如下：

D触发器功能表

D	$Q^{(n+1)}$
0	0
1	1

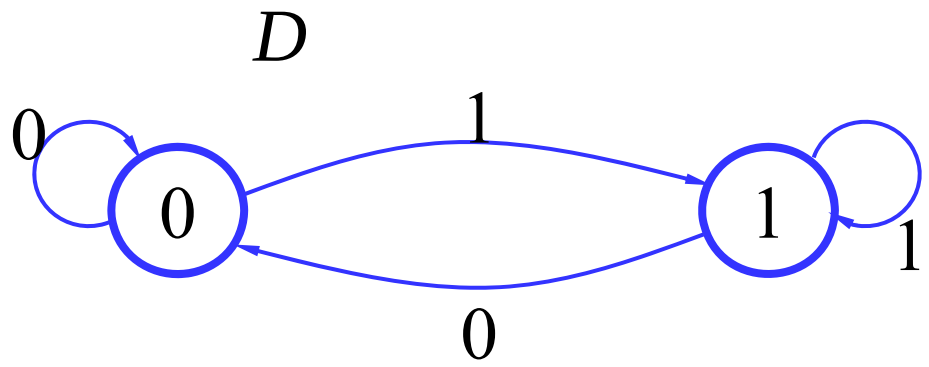
D触发器状态表

Q	$Q^{(n+1)}$	
	D＝0	D＝1
0	0	1
1	0	1

D触发器的次态方程为：

$$Q^{(n+1)}=D$$

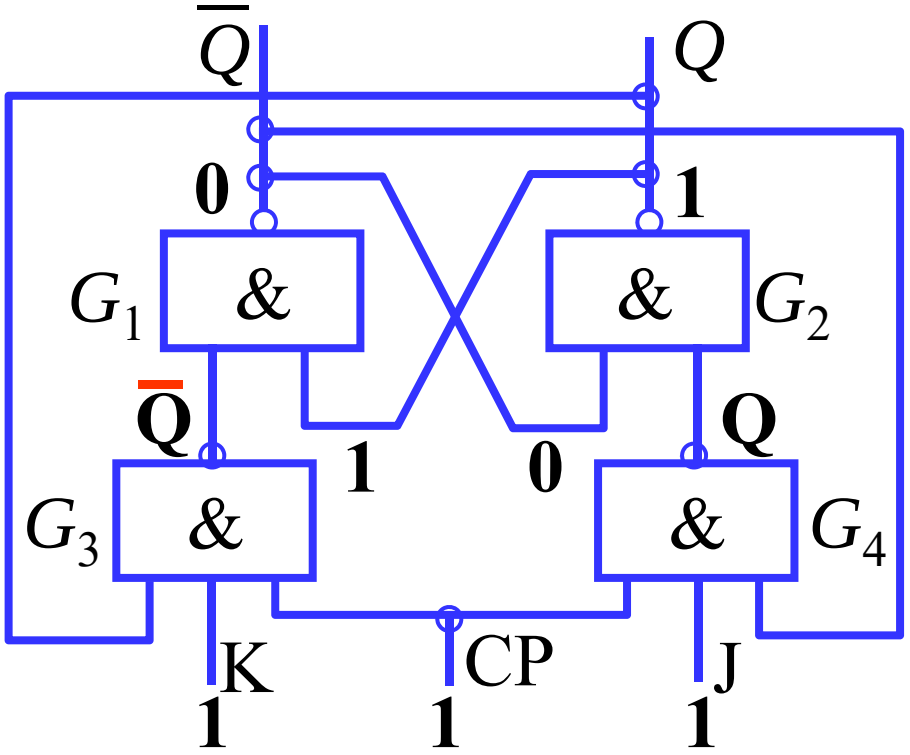
状态图为：



D触发器结构简单，但仍然存在"空翻"现象。实际使用的D触发器是一种维持阻塞型D触发器，可以防止"空翻"的发生。由于D触发器只有一个输入端，在某些情况下使用感到不便。

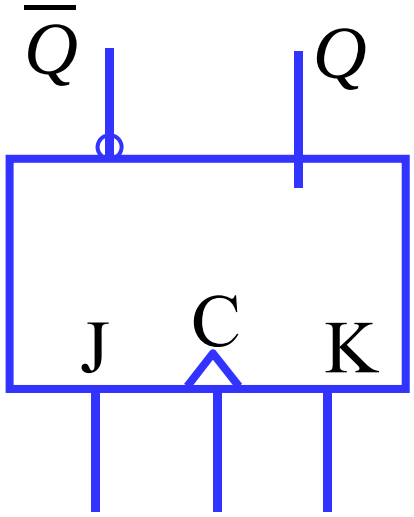
4.3.4　J－K触发器

J－K触发器有两个输入端，即克服了R－S触发器的"约束"问题，使用上又比D触发器灵活。其逻辑图与逻辑符号如下：



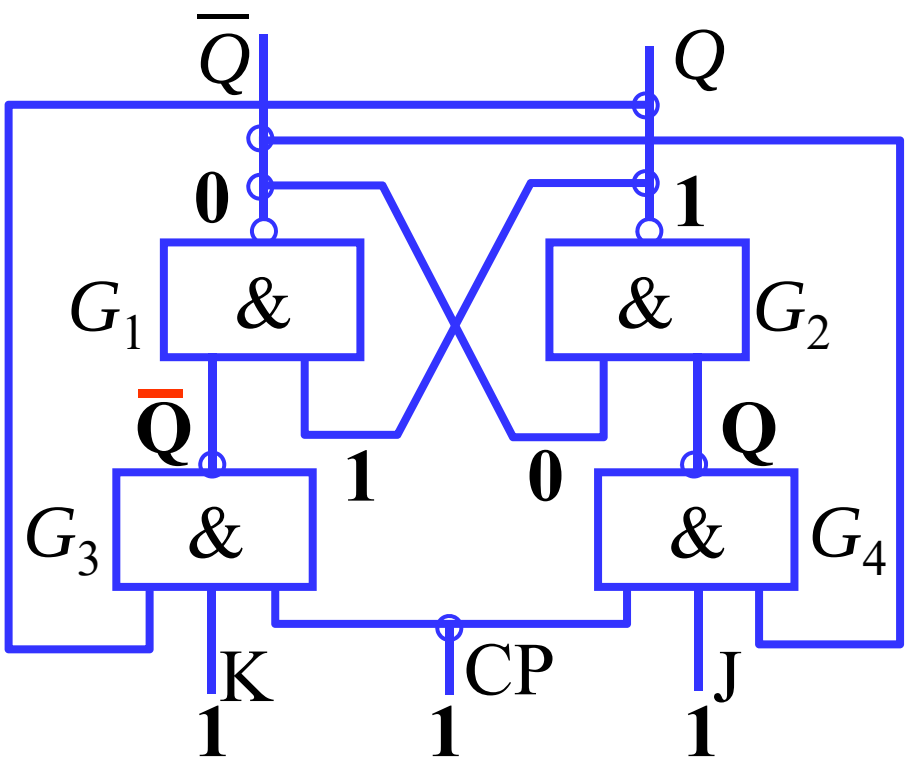
Q＝0

$Q^{(n+1)}$=1



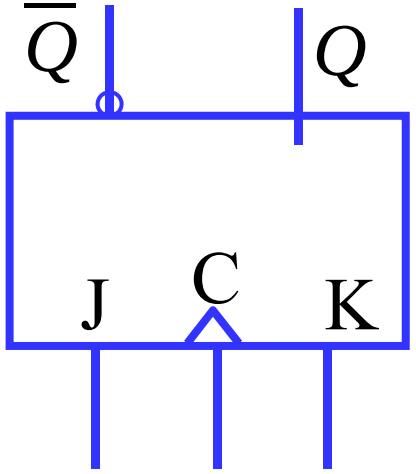
4.3.4　J－K触发器

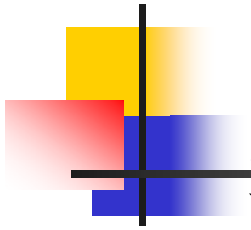
J－K触发器有两个输入端，即克服了R－S触发器的"约束"问题，使用上又比D触发器灵活。其逻辑图与逻辑符号如下：



Q＝0

$Q^{(n+1)}$=1





当CP＝0时，J－K触发器的状态保持不变；

当CP＝1时，

- 若J=K=0, 则G3=G4=1, 触发器保持原状态；
- 若J=1, K=0, 则G3=1, G4=Q, 使触发器置1；
- 若J=0, K=1, 则G3=$\overline{Q}$, G4=1, 使触发器置0；
- 若J=K=1, 则G3=$\overline{Q}$, G4=Q, 使触发器翻转；

J－K触发器功能表和状态表如下：

J－K触发器功能表

J	K	$Q^{(n+1)}$	功能说明
0	0	Q	不变
0	1	0	置 0
1	0	1	置 1
1	1	$\overline{Q}$	翻转

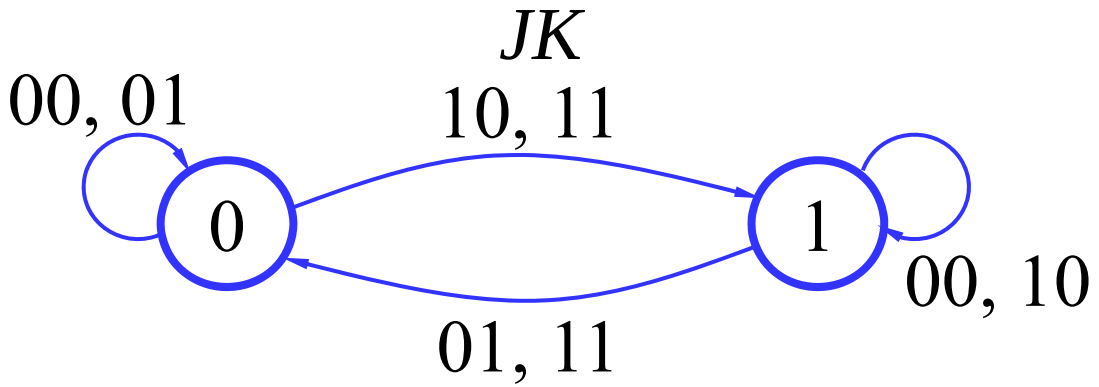
J－K触发器状态表

Q	$Q^{(n+1)}$			
	JK=00	0 1	1 1	1 0
0	0	0	1	1
1	1	0	0	1

J－K触发器的次态方程为：

$$Q^{(n+1)}=J\overline{Q}+\overline{K}Q$$

状态图为：



为防止"空翻"，实际数字电路中使用的J－K触发器是主从式集成J－K触发器，它使用"前沿采样，后沿定局"的方式，无空翻，功能较全，使用灵活。

4.3.5 T触发器

T触发器实际上是J－K触发器的一种特殊形式。如果把J－K触发器的J－K端连在一起就形成了T触发器。因此T触发器的次态方程为：

$$Q^{(n+1)}=T\overline{Q}+\overline{T}Q$$

其功能表和状态表是J－K触发器功能表和状态表的一部分。

T触发器又称"计数"触发器。

4.3.6 不同类型触发器之间的转换

转换方法：

利用令已有触发器和待求触发器的特性方程相等的原则，求出转换逻辑。

转换步骤：

（1）写出已有触发器和待求触发器的特性方程。
（2）变换待求触发器的特性方程，使之形式与已有触发器的特性方程一致。
（3）比较已有和待求触发器的特性方程，根据两个方程相等的原则求出转换逻辑。
（4）根据转换逻辑画出逻辑电路图。

1、将JK触发器转换为RS、D、T和T′触发器

JK触发器→RS触发器

变换*RS*触发器的特性方程，使之形式与*JK*触发器的特性方程一致：

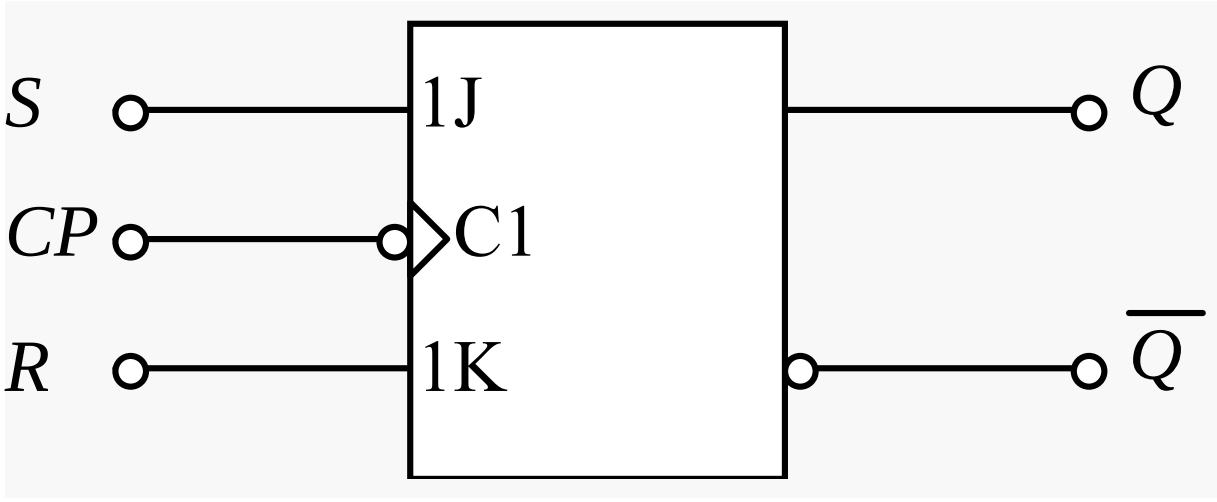
*RS*触发器特性方程

$$\begin{cases} Q^{n+1} = S + \overline{R}Q^n \\ RS = 0 \end{cases}$$

$$\begin{aligned} Q^{n+1} &= S + \overline{R}Q^n = S(\overline{Q}^n + Q^n) + \overline{R}Q^n \\ &= S\overline{Q}^n + SQ^n + \overline{R}Q^n \\ &= S\overline{Q}^n + \overline{R}Q^n + SQ^n(\overline{R} + R) \\ &= S\overline{Q}^n + \overline{R}Q^n + \overline{R}SQ^n + RSQ^n \\ &= S\overline{Q}^n + \overline{R}Q^n \end{aligned}$$

$$\left.\begin{aligned} Q^{n+1} &= J\overline{Q}^{n} + \overline{K}Q^{n} \\ Q^{n+1} &= S\overline{Q}^{n} + \overline{R}Q^{n} \end{aligned}\right\}$$ 比较，得：$$\begin{cases} J = S \\ K = R \end{cases}$$

电路图



JK触发器→D触发器

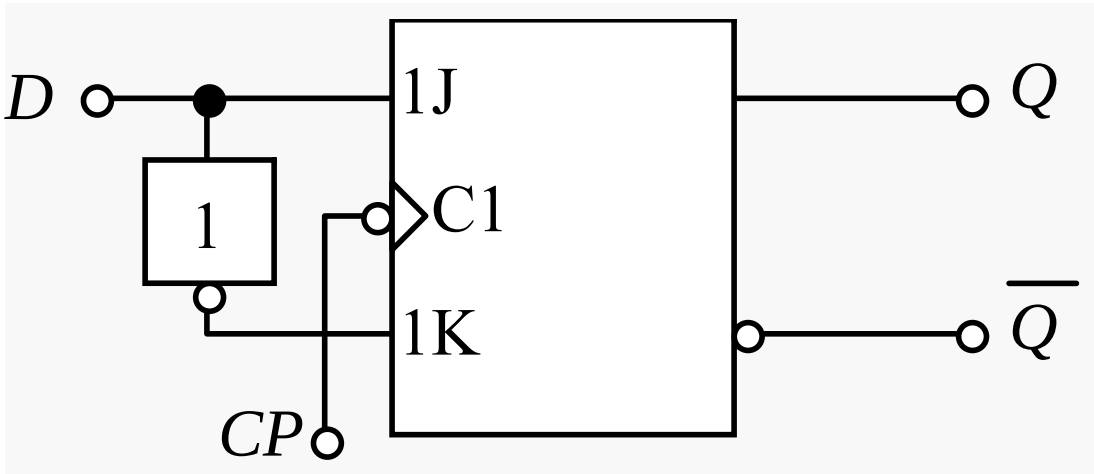
写出*D*触发器的特性方程，并进行变换，使之形式与*JK*触发器的特性方程一致：

$$Q^{n+1} = D = D(\overline{Q}^n + Q^n) = D\overline{Q}^n + DQ^n$$

与*JK*触发器的特性方程比较，得：

$$\begin{cases} J = D \\ K = \overline{D} \end{cases}$$

电路图



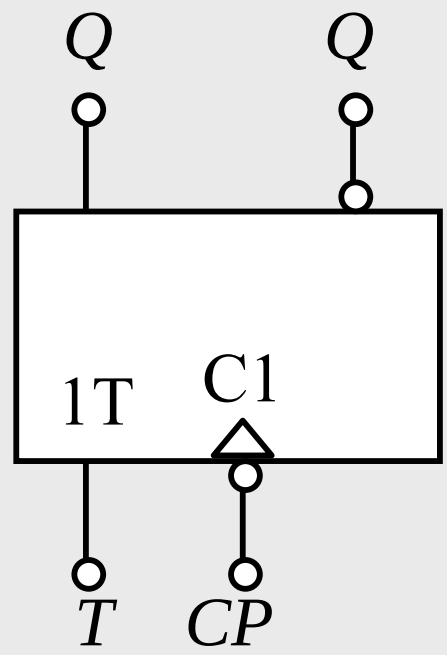
JK触发器→T触发器

在数字电路中，凡在*CP*时钟脉冲控制下，根据输入信号*T*取值的不同，具有保持和翻转功能的电路，即当*T*＝0时能保持状态不变，*T*＝1时一定翻转的电路，都称为*T*触发器。

特性表

T	Q^n	Q^{n+1}	功能
0	0	0	$Q^{n+1}=Q^n$ 保持
0	1	1	
1	0	1	$Q^{n+1}=\overline{Q}^n$ 翻转
1	1	0	

逻辑符号



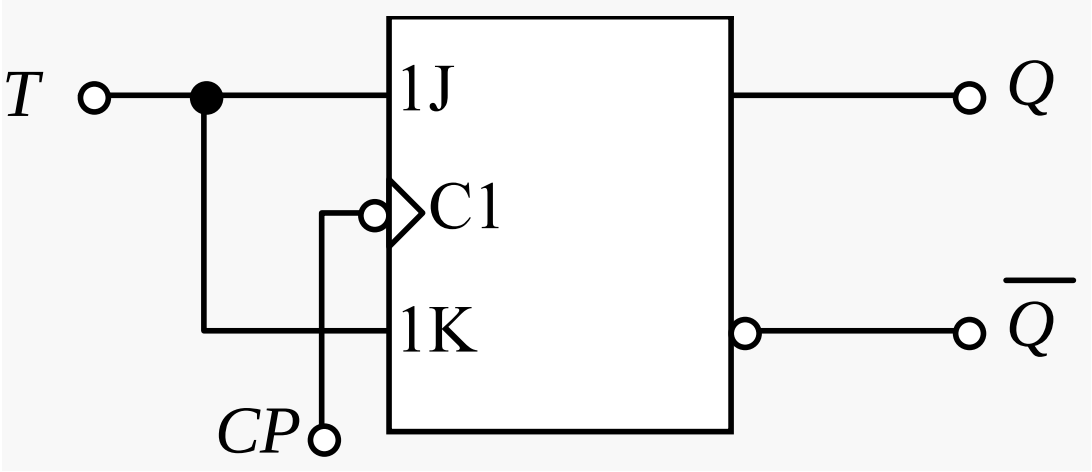
T触发器特性方程：

$$Q^{n+1} = T\overline{Q}^{n} + \overline{T}Q^{n} = T \oplus Q^{n}$$

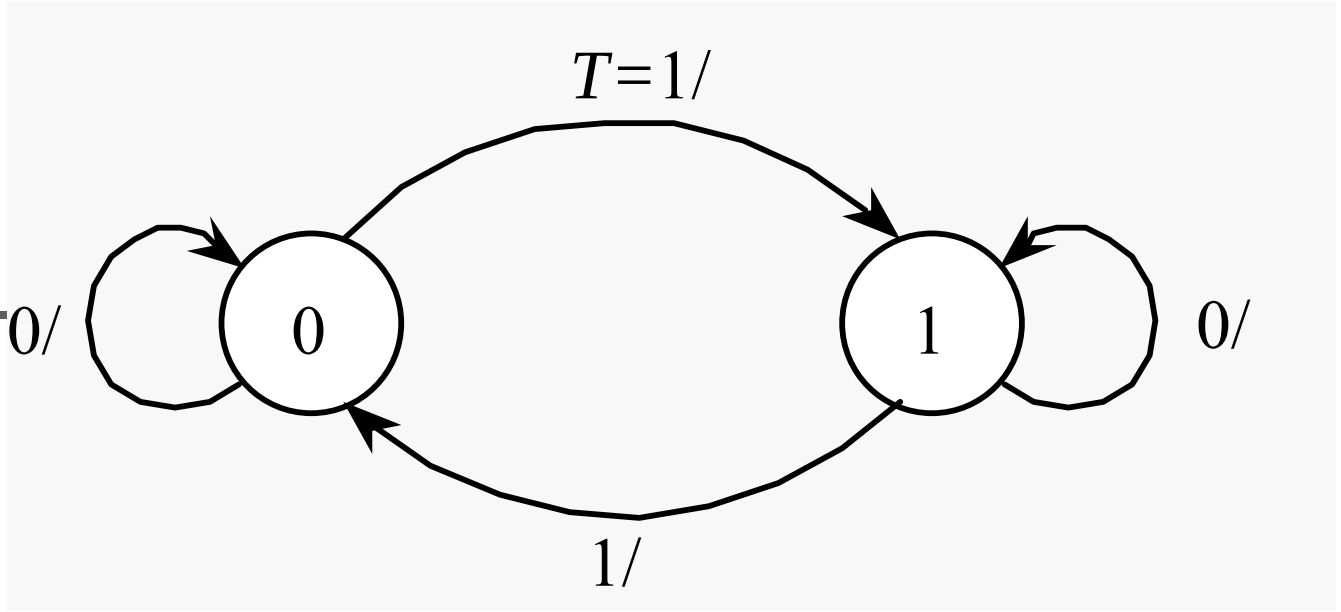
与JK触发器的特性方程比较，得：

$$\begin{cases} J = T \\ K = T \end{cases}$$

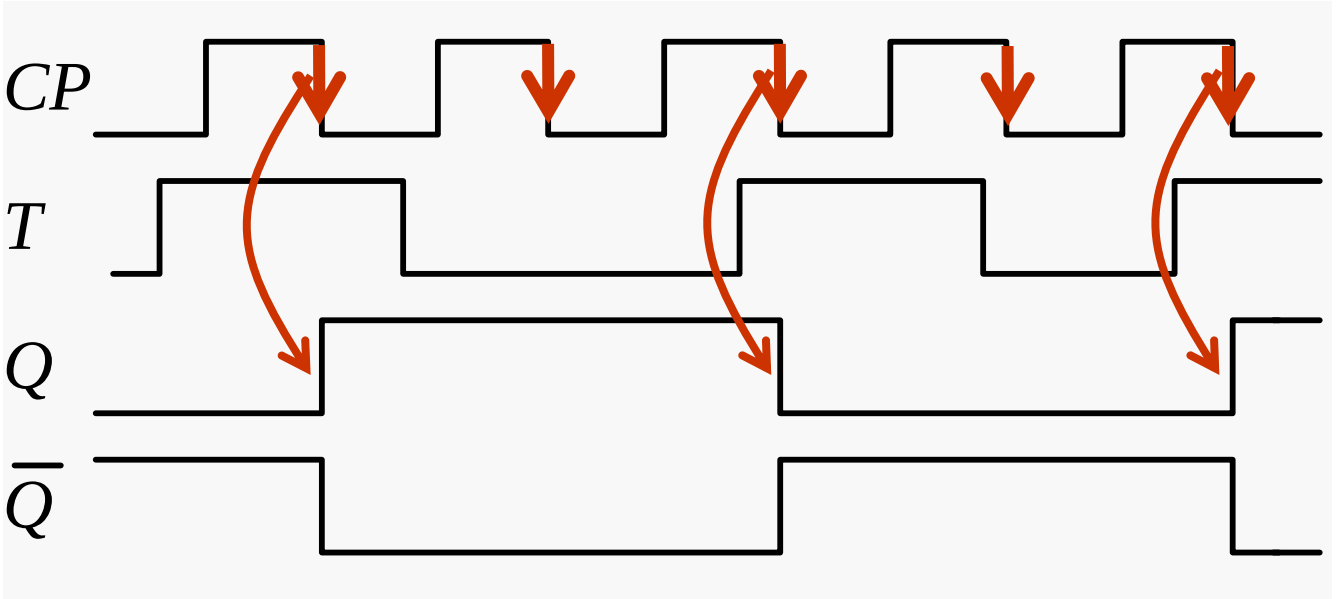
电路图



状态图



时序图



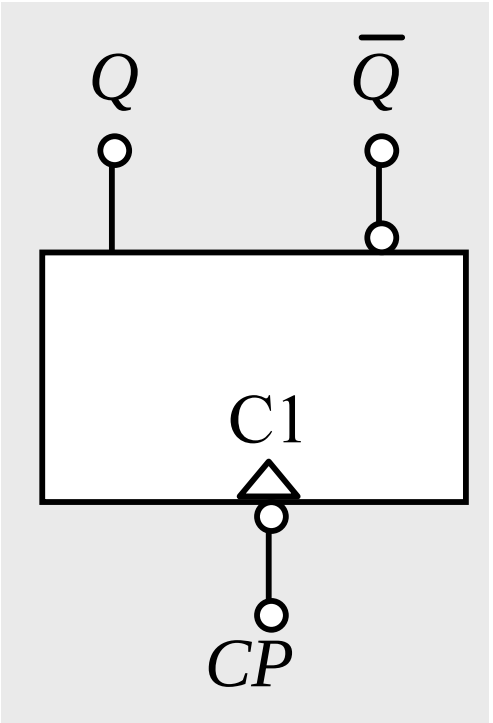
JK触发器→T′触发器

在数字电路中，凡每来一个时钟脉冲就翻转一次的电路，都称为T′触发器。

特性表

Q^n	Q^{n+1}	功能
0	1	$Q^{n+1}=\overline{Q}^n$
1	0	翻转

逻辑符号



T' 触发器特性方程：

$$Q^{n+1}=\overline{Q}^{n}$$

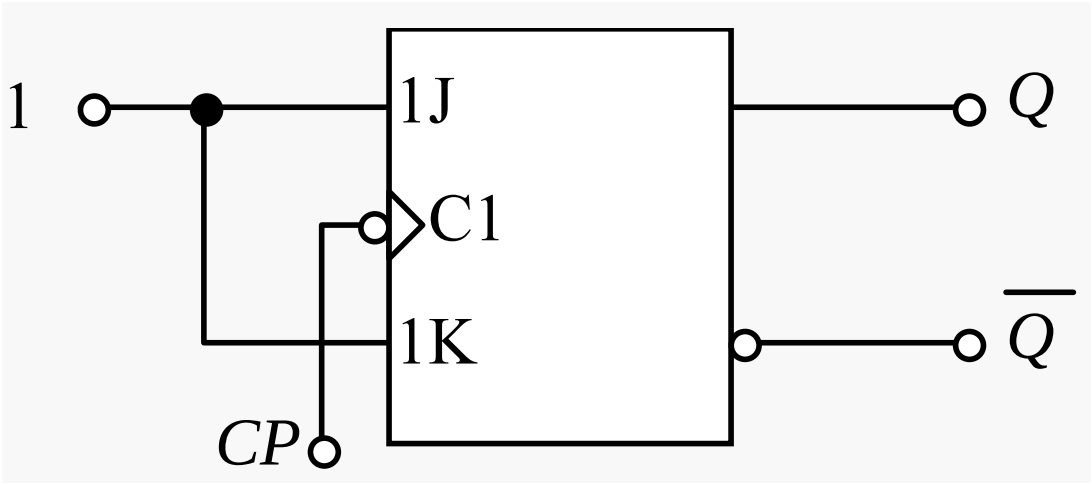
变换T'触发器的特性方程：

$$Q^{n+1}=\overline{Q}^{n}=1\cdot\overline{Q}^{n}+\overline{1}\cdot Q^{n}$$

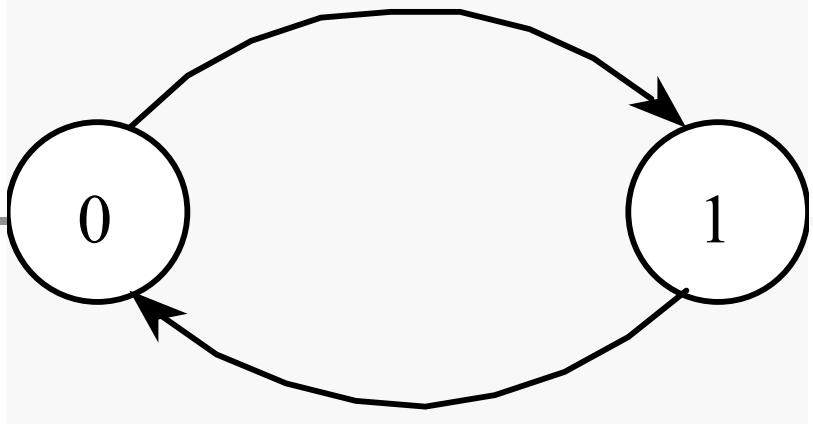
与JK触发器的特性方程比较，得：

$$\begin{cases}J=T\\K=T\end{cases}$$

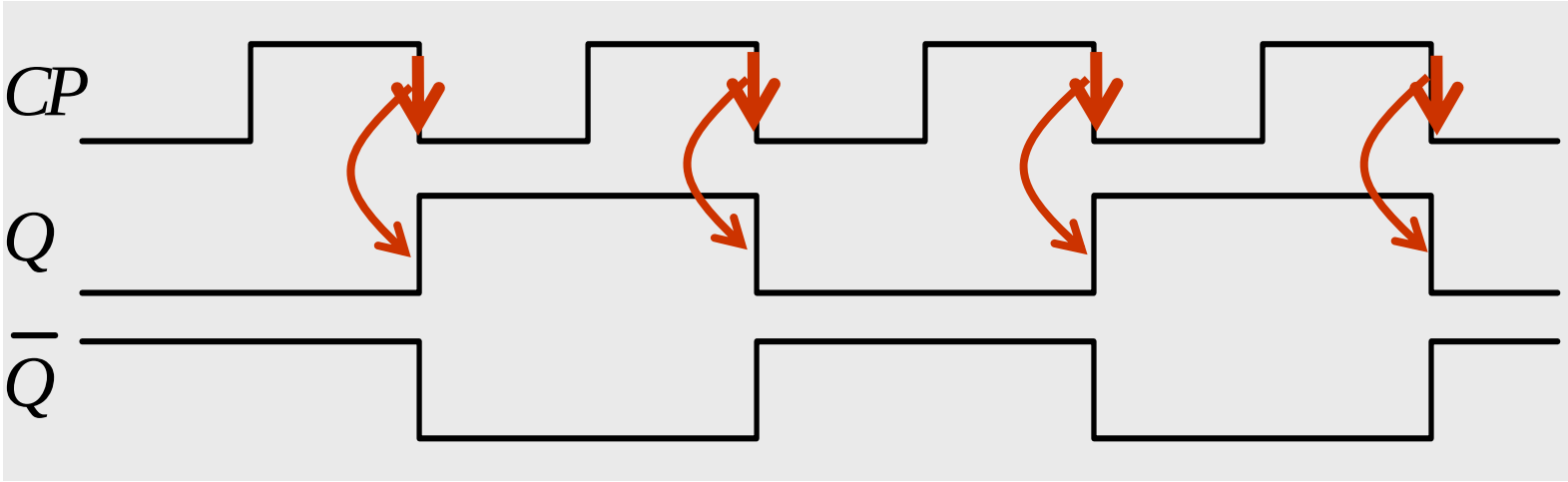
电路图



状态图



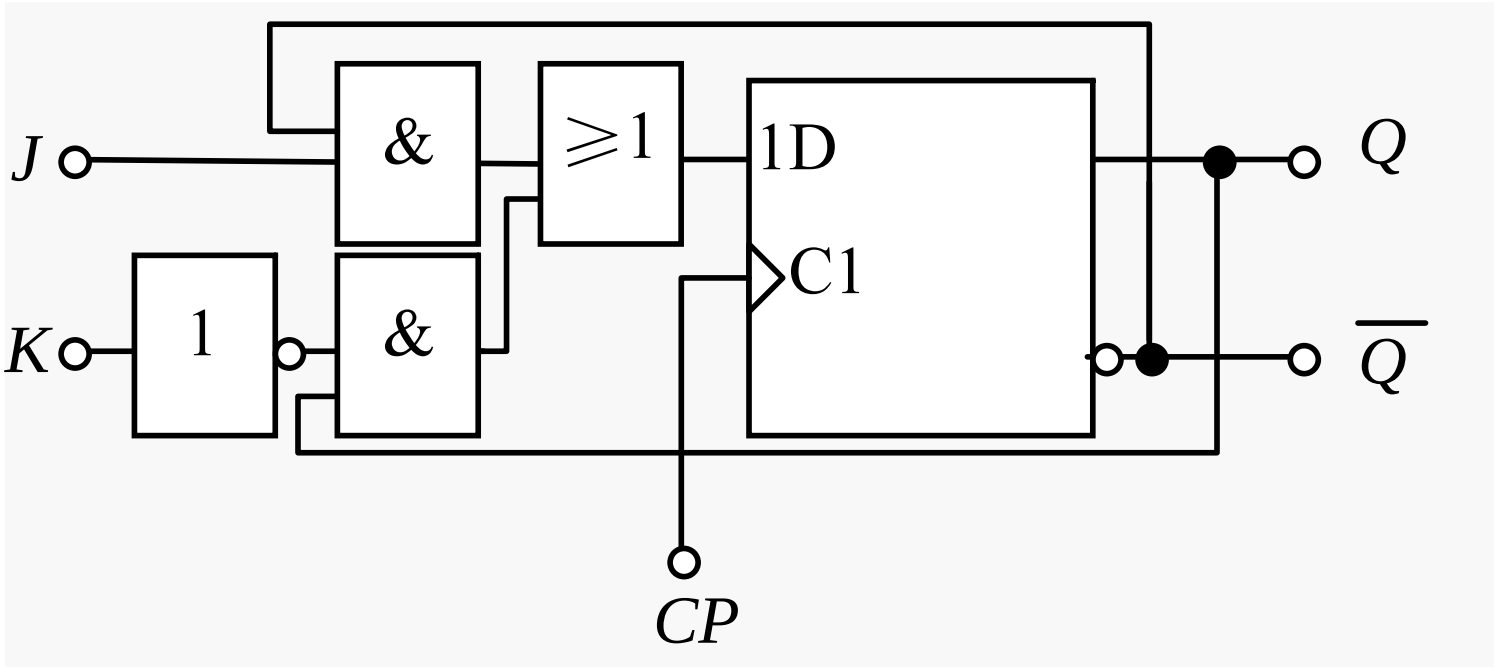
时序图



2、将D触发器转换为JK、T和T′ 触发器

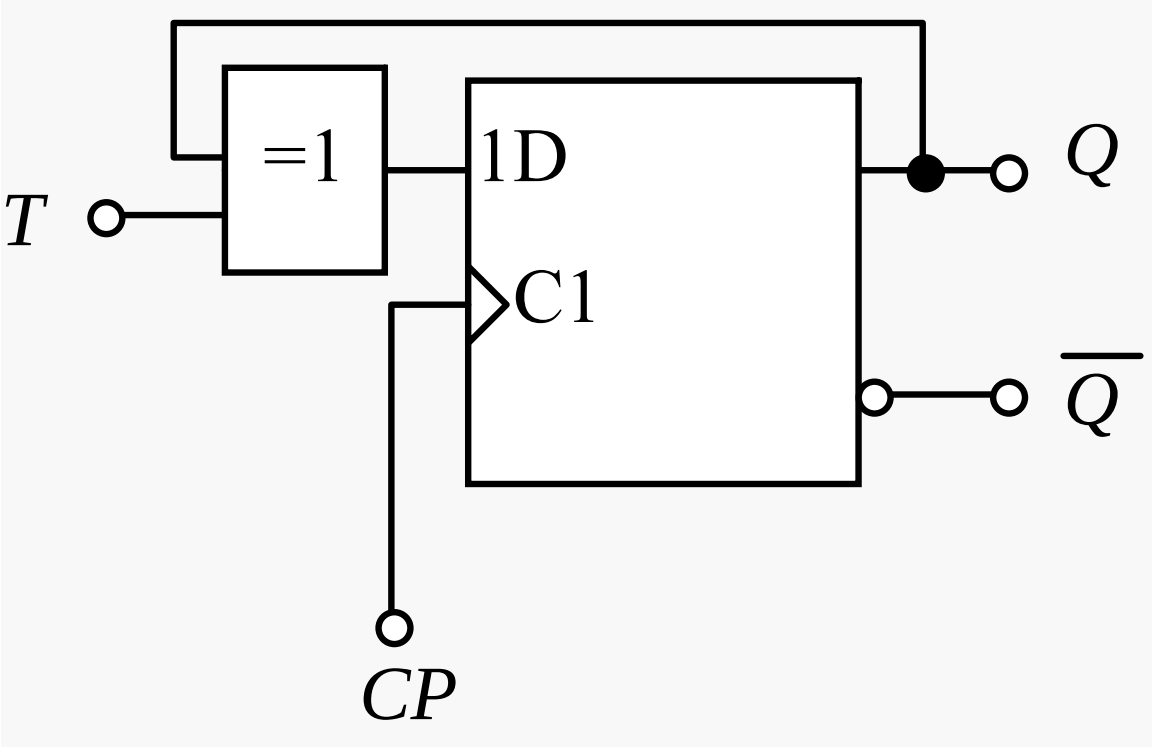
D触发器→JK触发器

$$D = J\overline{Q}^n + \overline{K}Q^n$$



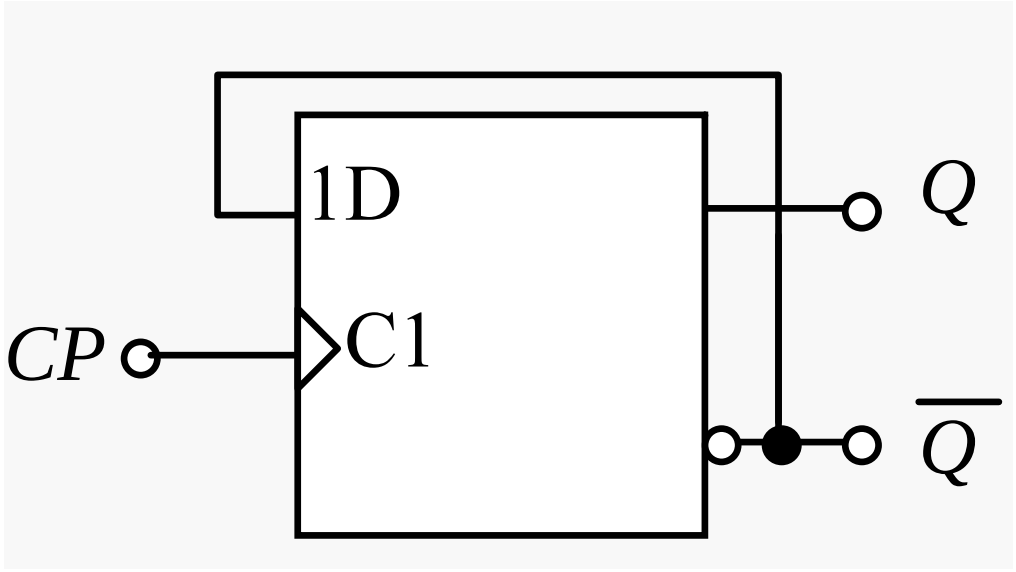
D触发器→T触发器

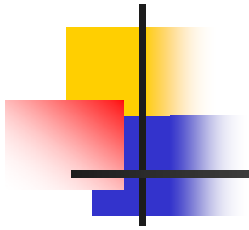
$$D = T \oplus Q^n$$



D触发器→T′触发器

$$D = \overline{Q}^{n}$$



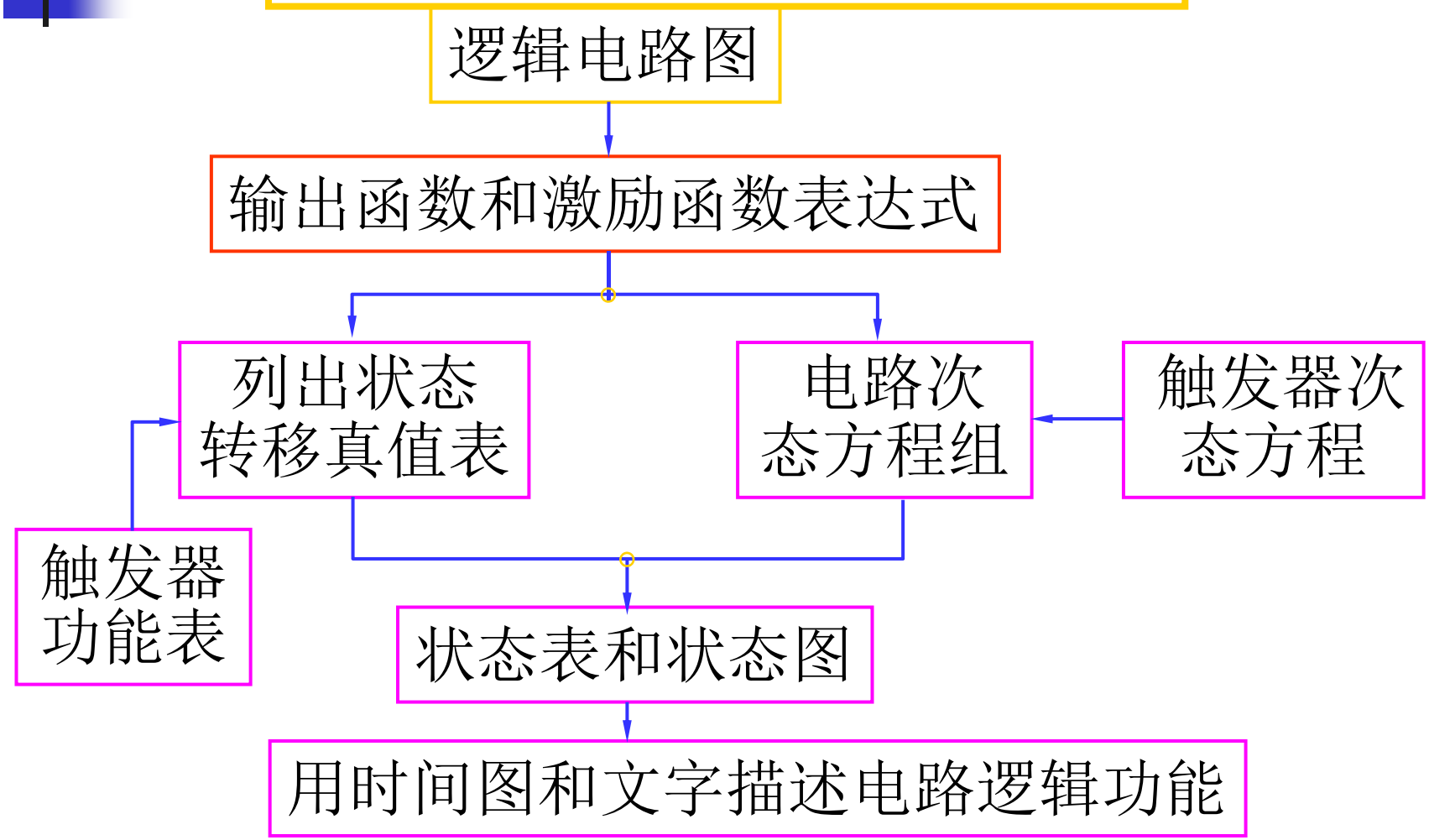


4.4 同步时序逻辑电路的分析

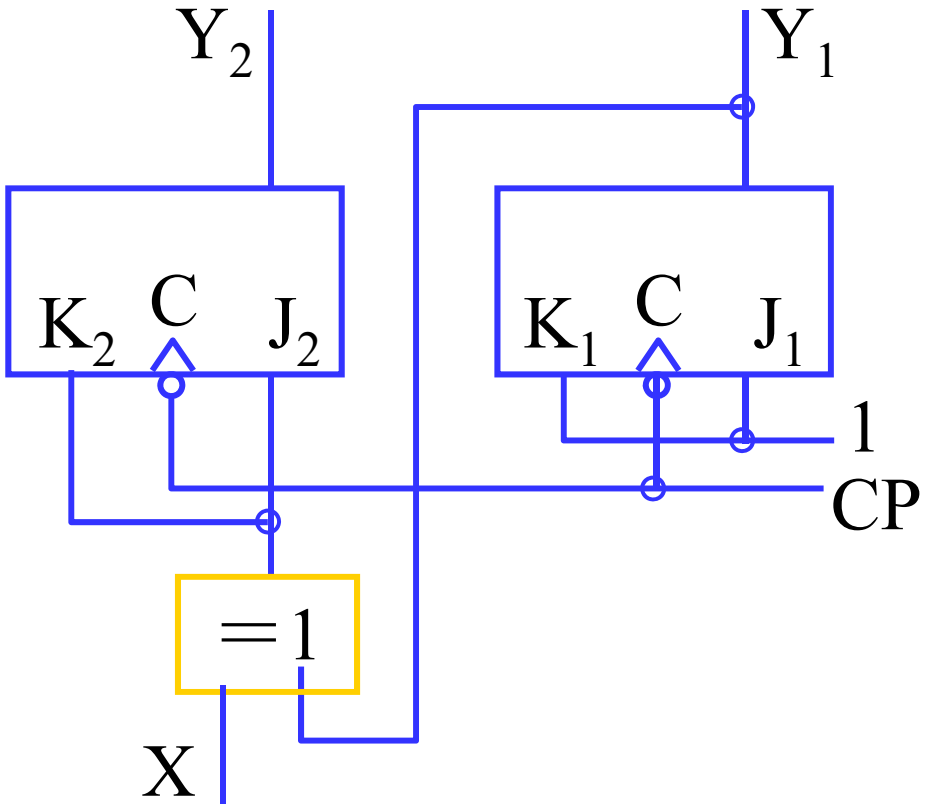
分析的任务：对一个给定的时序逻辑电路，研究在一系列输入信号作用下，电路将会产生怎样的输出，进而说明该电路的逻辑功能。实际上是要求出电路的状态表、状态图或时间图，并作出功能评述。

4.4.1 同步时序逻辑电路的分析方法

常用的方法有表格化和代数法。



例：用表格法分析下图所示的同步是序逻辑电路



解：

第一步：写出输出函数和激励函数表达式。

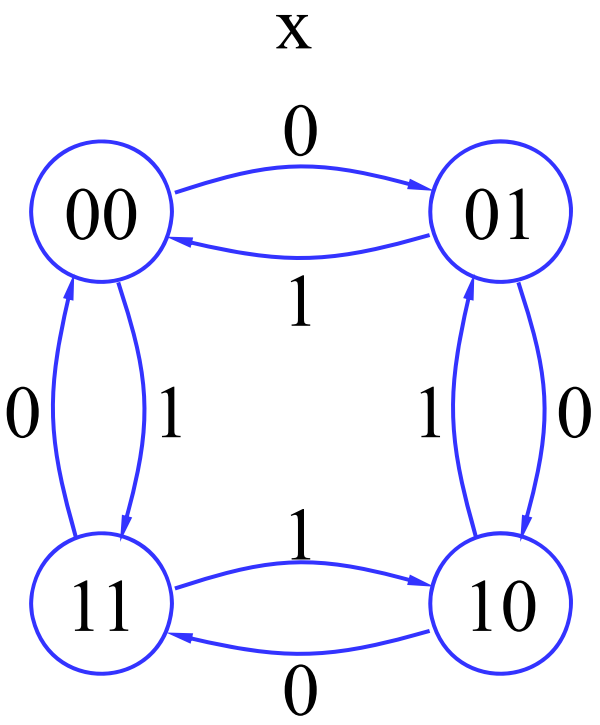
$J_1=K_1=1$　　$J_2=K_2=x \oplus y_1$

第二步：列同状态转移真值表。

输入	现态		激励函数				次态	
x	y_2	y_1	J_2	K_2	J_1	K_1	$y_2^{(n+1)}$	$y_1^{(n+1)}$
0	0	0	0	0	1	1	0	1
0	0	1	1	1	1	1	1	0
0	1	0	0	0	1	1	1	1
0	1	1	1	1	1	1	0	0
1	0	0	1	1	1	1	1	1
1	0	1	0	0	1	1	0	0
1	1	0	1	1	1	1	0	1
1	1	1	0	0	1	1	1	0

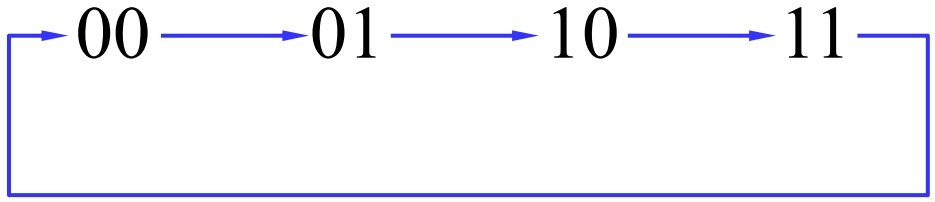
第三步：作出状态表和状态图。

现 态	次 态 $y_2^{(n+1)}y_1^{(n+1)}$	
y_2 y_1	x = 0	x = 1
0 0	0 1	1 1
0 1	1 0	0 0
1 0	1 1	0 1
1 1	0 0	1 0

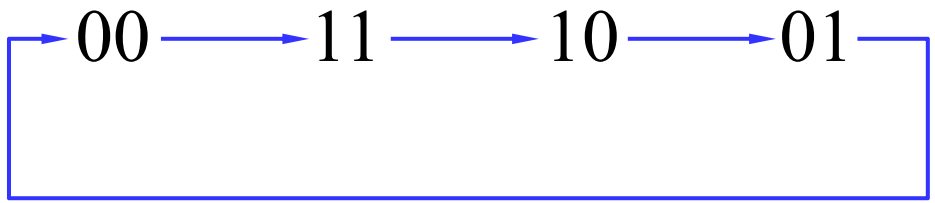


第四步：用时间图和文字描述电路和逻辑功能。

当x=0时,该电路进行加1计数,计数序列为:



当x=1时,该电路进行减1计数,计数序列为:



时间图的作法：

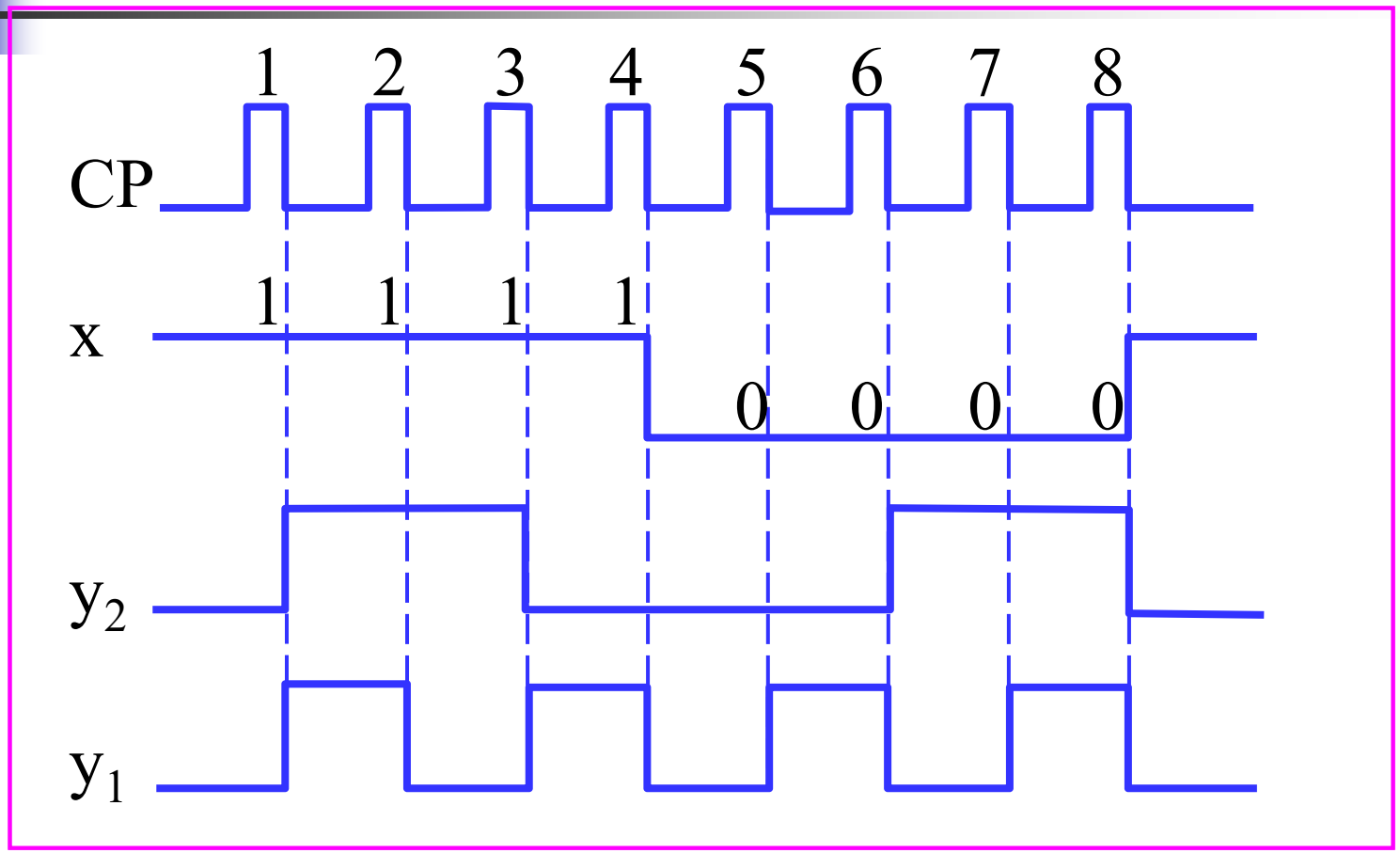
选定一个典型的输入序列；

输入序列为：x=11110000,初态： y_2y_1=00

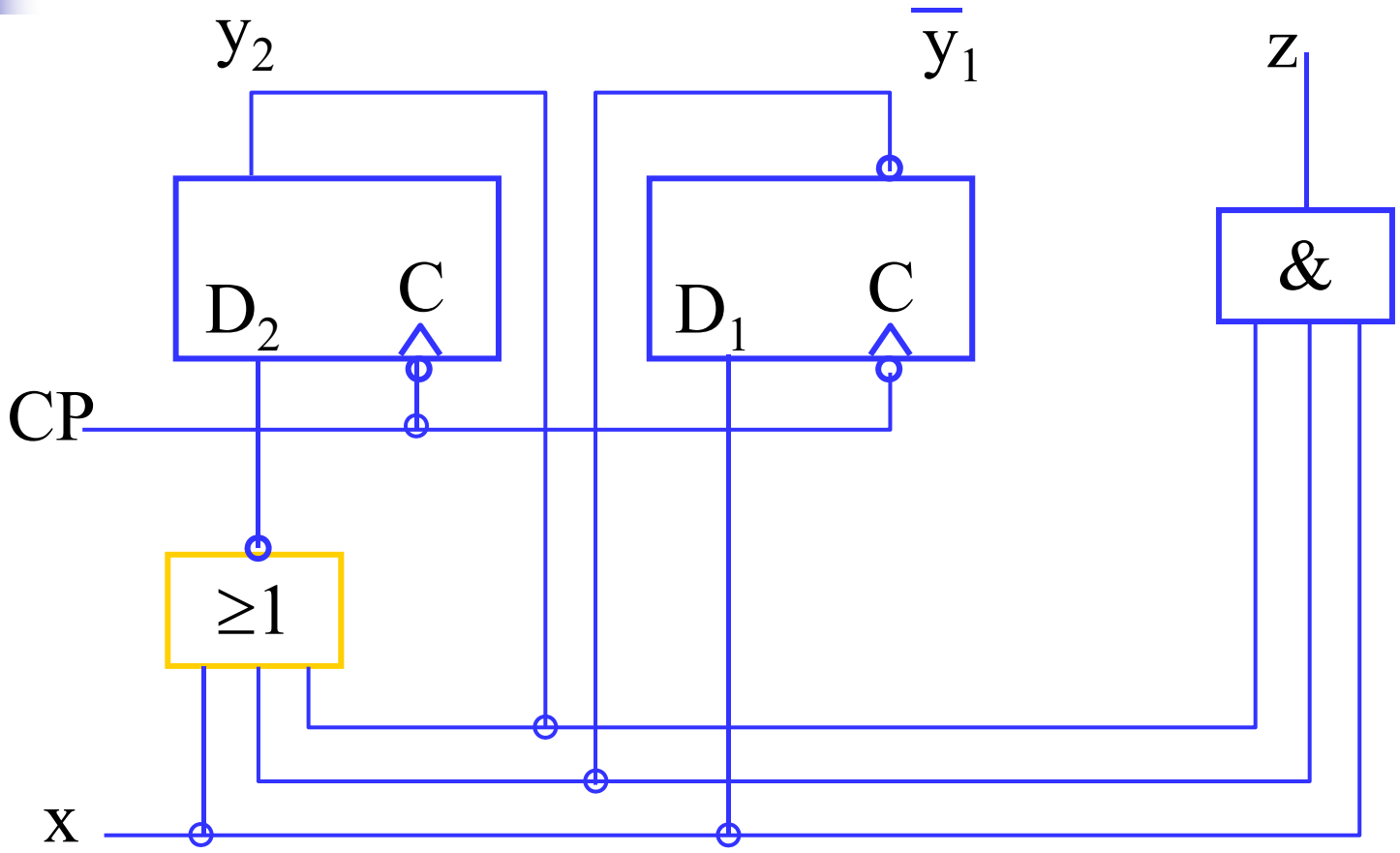
根据选定的典型输入序列，求出状态响应序列(和输出响应序列)

CP:	1	2	3	4	5	6	7	8
x:	1	1	1	1	0	0	0	0
y_2:	0	1	1	0	0	0	1	1
y_1:	0	1	0	1	0	1	0	1
$y_2^{(n+1)}$:	1	1	0	0	0	1	1	0
$y_1^{(n+1)}$:	1	0	1	0	1	0	1	0

画时间图。



例：试有代数法分析下图所示的同步时序逻辑电路。



解：

第一步：写出输出函数和激励函数表达式。

$$Z=x\,y_2\,\bar{y}_1$$

$$D_2=\overline{x+y_2+\bar{y}_1}=\bar{x}\,\bar{y}_2\,y_1$$

$$D_1=x$$

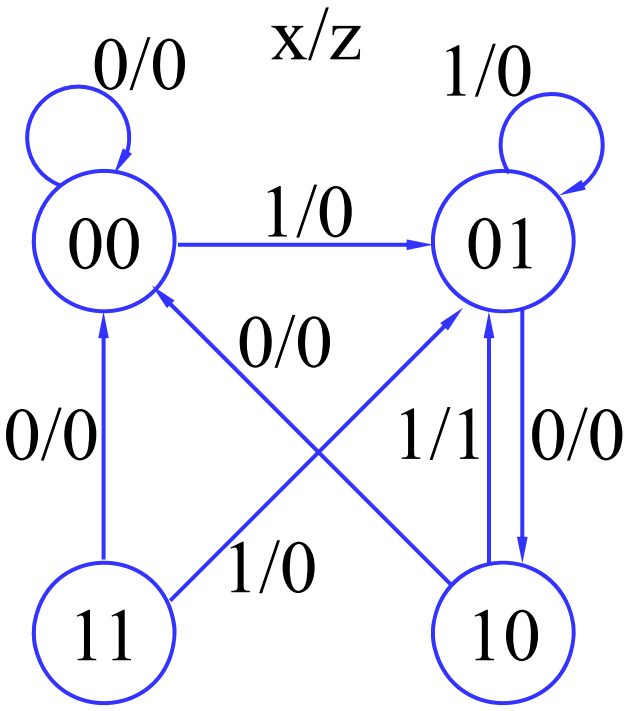
第二步：把激励函数表达式代入触发器的次态方程，得到电路的次态方程组。

$$Q_2^{(n+1)}=D_2=\bar{x}\,\bar{y}_2\,y_1$$

$$Q_1^{(n+1)}=D_1=x$$

第三步：根据次态方程组和输出函数表达式作出状态表和状态图。

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



第四步：作出时间图，并说明电路的逻辑功能。

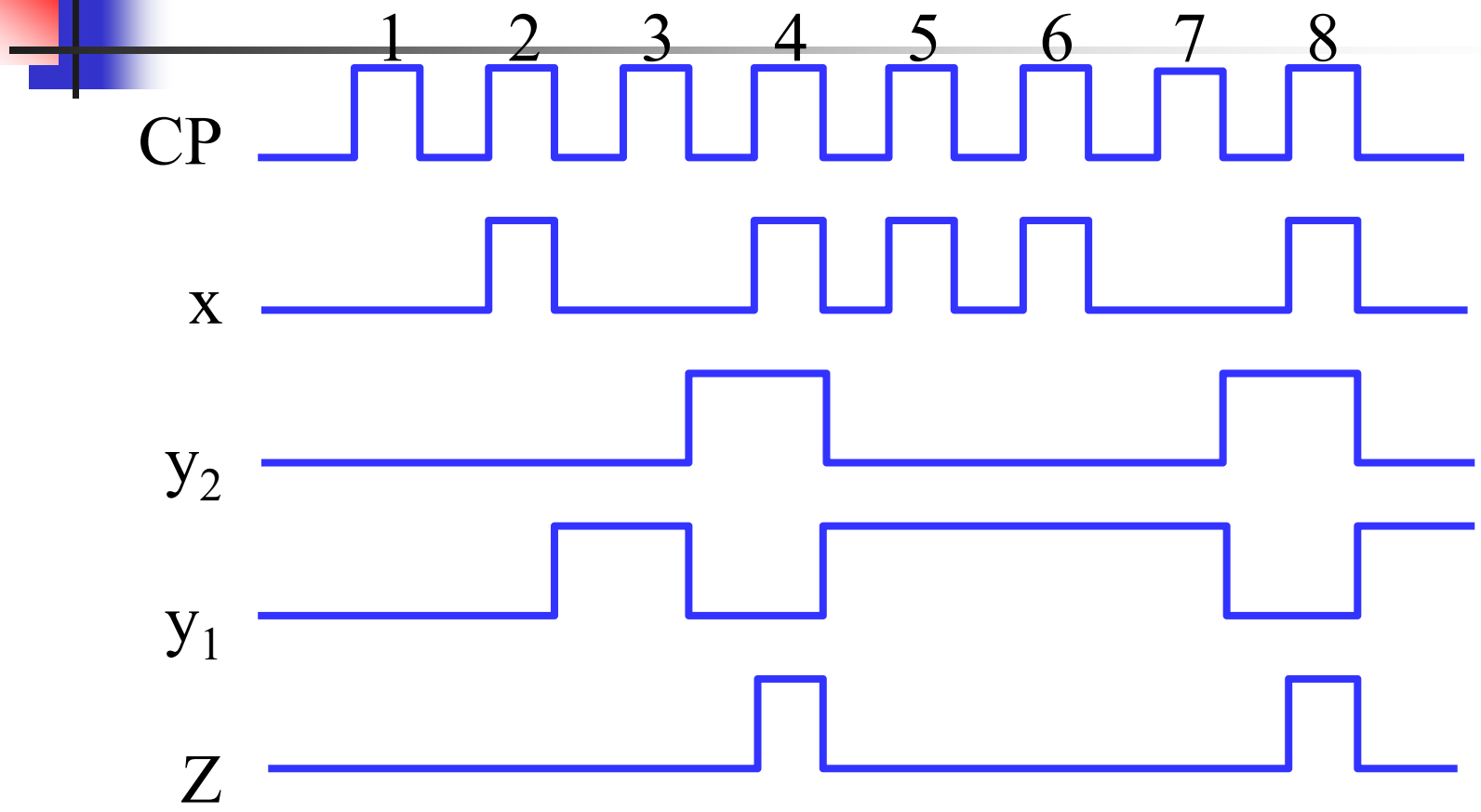
典型输入序列：x = 01011101

初态：$y_2\,y_1 = 00$

状态响应序列和输出响应序列为：

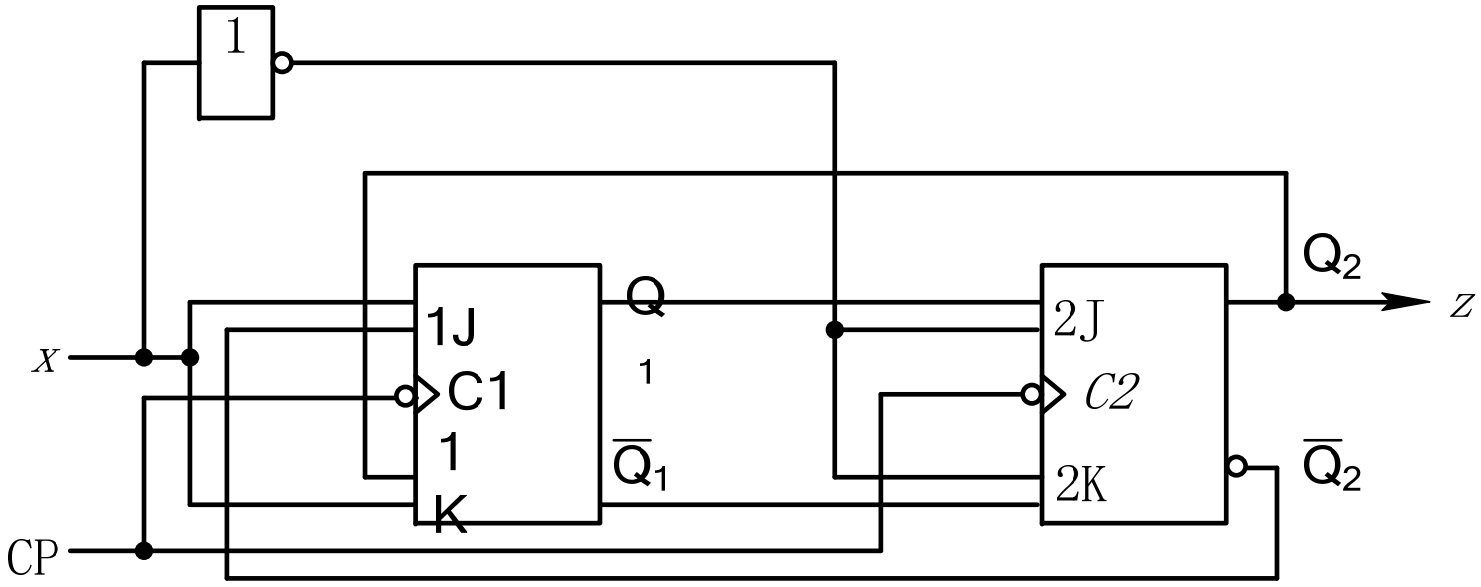
CP:	1	2	3	4	5	6	7	8
x:	0	1	0	1	1	1	0	1
y_2:	0	0	0	1	0	0	0	1
y_1:	0	0	1	0	1	1	1	0
$y_2^{(n+1)}$:	0	0	1	0	0	0	1	0
$y_1^{(n+1)}$:	0	1	0	1	1	1	0	1
Z:	0	0	0	1	0	0	0	1

时间图：



功能说明：该电路是一个"101"序列检测器。

例 1 时序电路如图 所示，试分析其功能，并画出x序列为1010 1100 的时序图，设起始态Q_2Q_1=00。



解 该电路中，时钟脉冲接到每个触发器的时钟输入端，故为同步时序电路。

(1) 写出方程。

① 激励方程、输出方程如下：

$$J_1 = x\overline{Q_2^n},\quad K_1 = xQ_2^n,\quad J_2 = \bar{x}Q_1^n,\quad K_2 = \bar{x}\overline{Q_1^n}$$

$$z = Q_2^n$$

② 次态方程。

将上述激励函数代入触发器的特性方程中， 即得每一触发器的次态方程。

$$Q_1^{n+1} = J_1\overline{Q_1^n} + \overline{K}_1 Q_1^n = x\overline{Q_2^n}\,\overline{Q_1^n} + \overline{xQ_2^n}\,Q_1^n$$

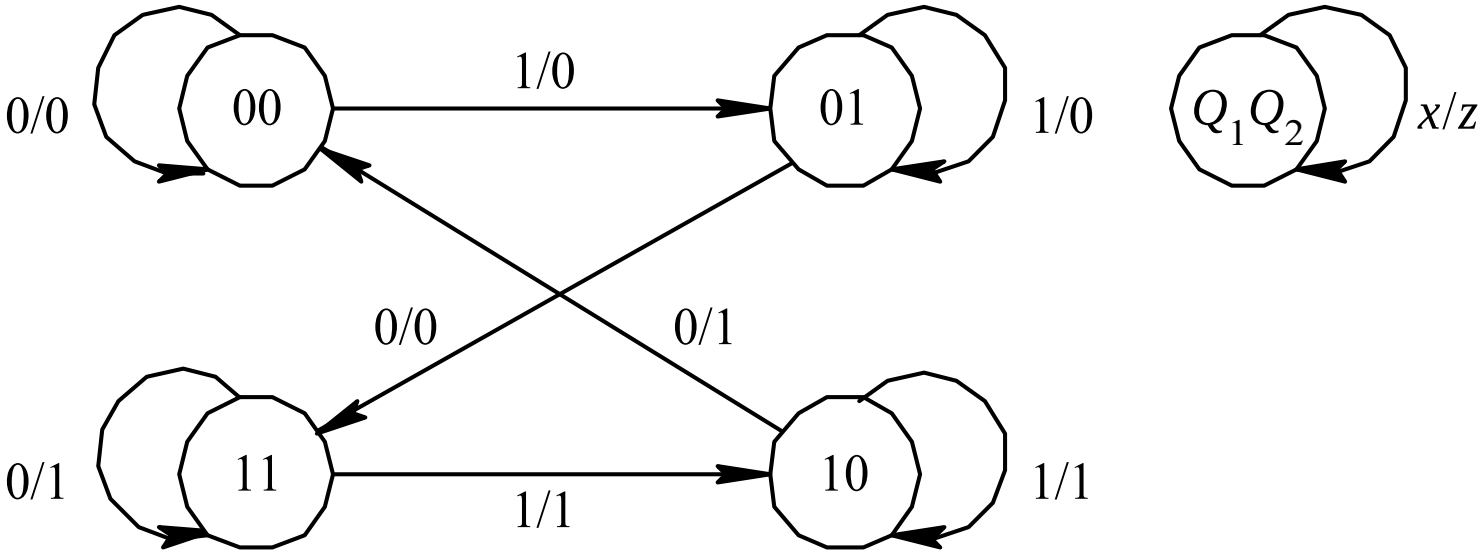
$$Q_2^{n+1} = J_2\overline{Q_2^n} + \overline{K}_2 Q_2^n = \overline{x}\,Q_1^n\overline{Q_2^n} + \overline{xQ_1^n}\,Q_2^n$$

(2) 列出状态真值表。

状态真值表

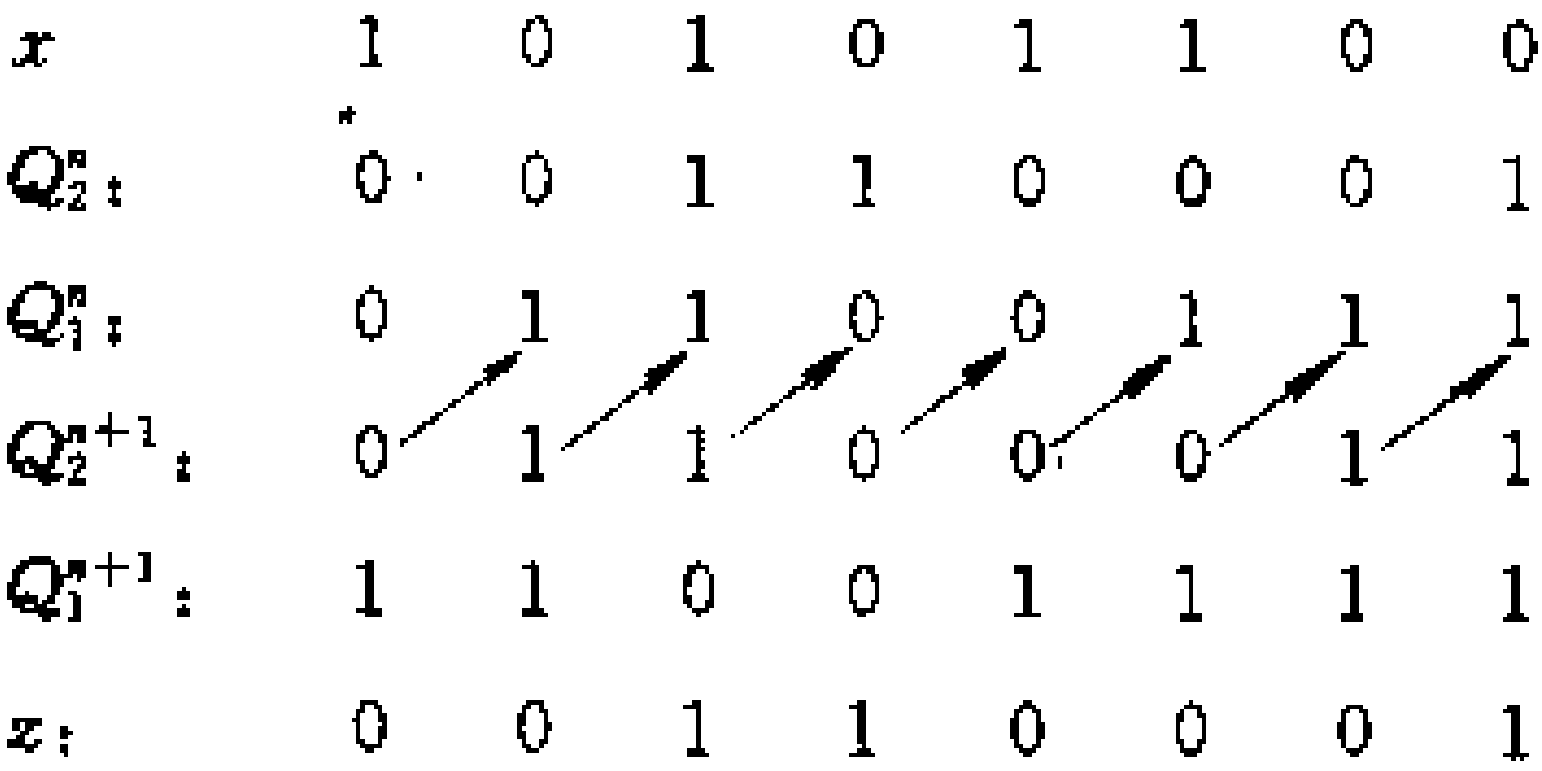
Q_2^n	x	Q_1^n	Q_2^{n+1}	Q_1^{n+1}	z
0	0	0	0	0	0
0	0	1	1	1	0
0	1	0	0	0	0
0	1	1	1	1	0
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	0	1

(3) 画出状态迁移图。

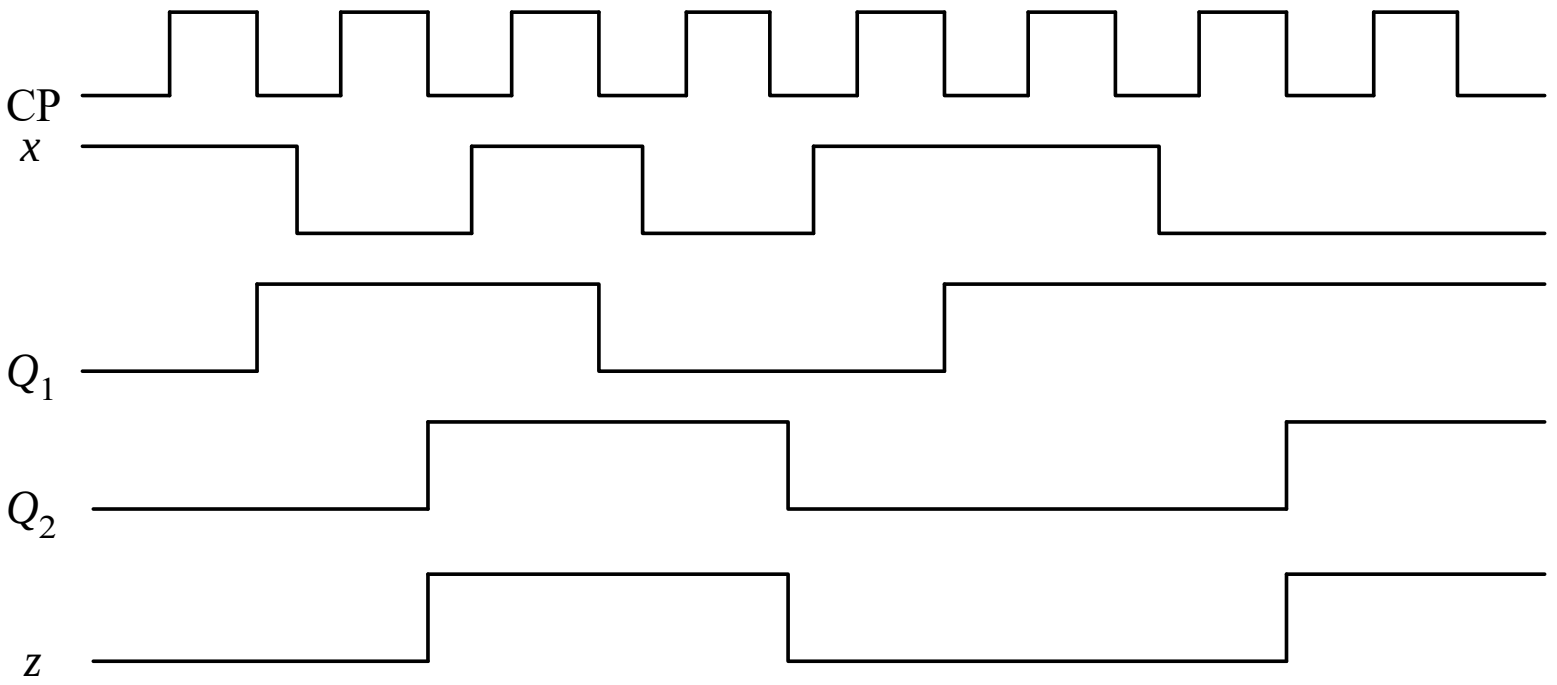


状态图

(4) 画出给定输入x序列的时序图。

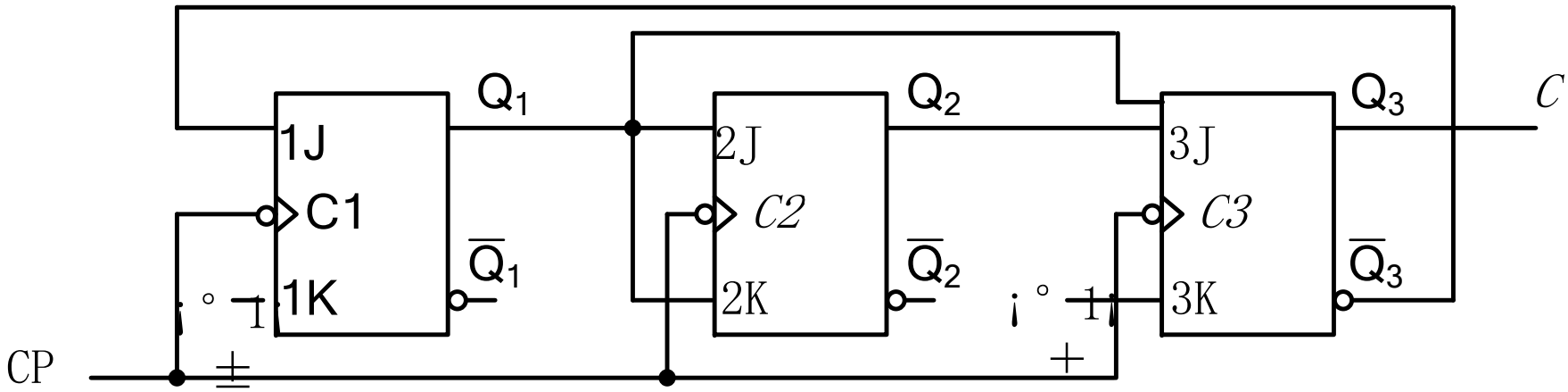


根据上述时序关系作出时序图， 如图 所示。



时序波形图

例 2　时序电路如图 所示，分析其功能。



例 2 图

解 该电路为同步时序电路。

从电路图得到每一级的激励、输出方程如下：

$$J_1 = \overline{Q_3^n} \quad K_1 = 1$$

$$J_2 = Q_1^n \quad K_2 = Q_1^n$$

$$J_3 = Q_1^n Q_2^n \quad K_3 = 1$$

$$C = Q_3^n$$

其次态方程为

$$Q_1^{n+1} = \overline{Q_3^n}\ \overline{Q_1^n}$$

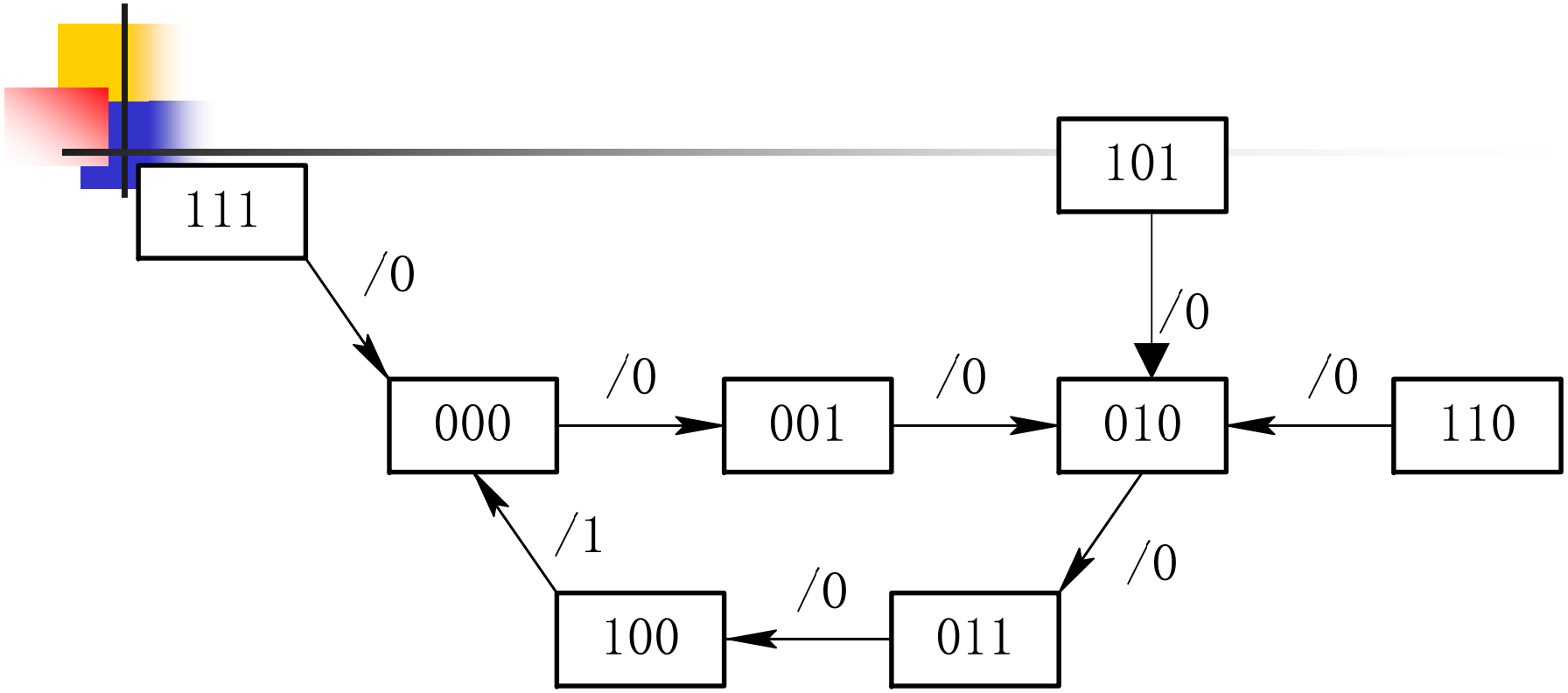
$$Q_2^{n+1} = Q_1^n\ \overline{Q_2^n} + \overline{Q_1^n}\ Q_2^n$$

$$Q_3^{n+1} = Q_1^n Q_2^n\ \overline{Q_3^n}$$

根据方程可得出状态迁移表，如表所示，再由表得状态迁移图，如图 所示。

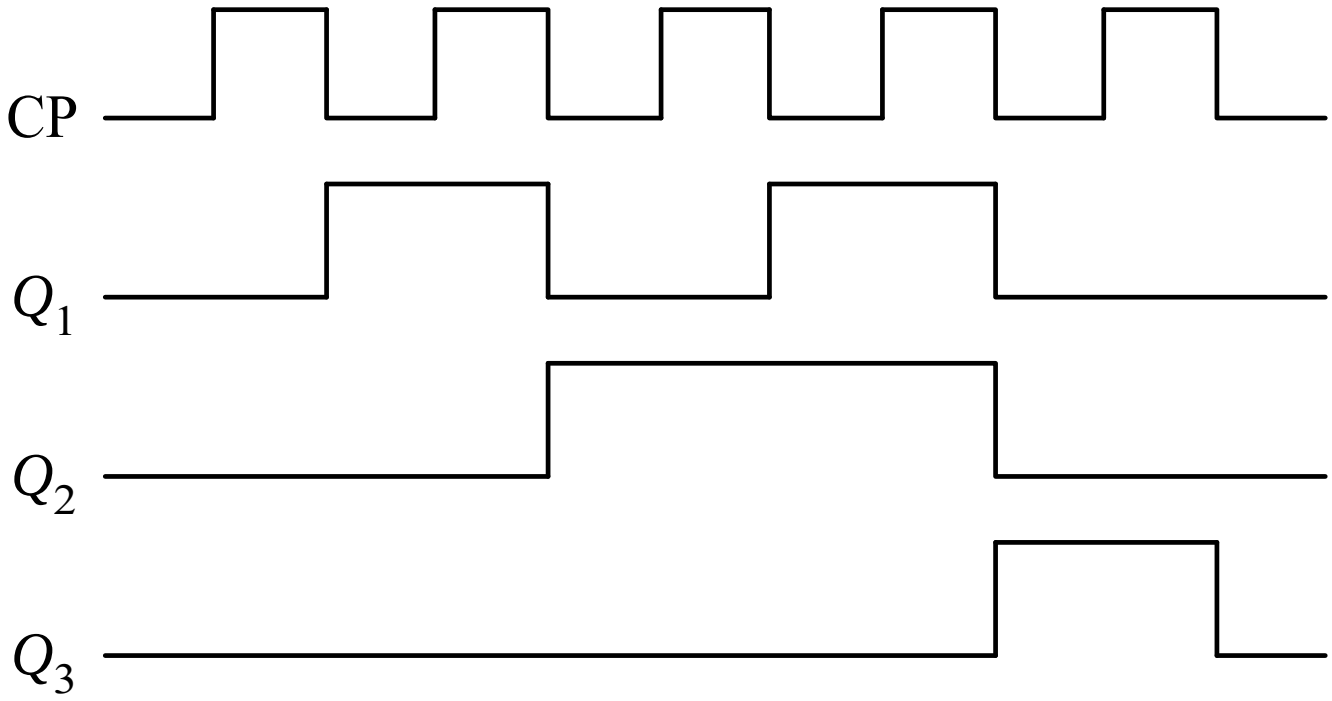
例 2 状态表

Q_3^n	Q_2^n	Q_1^n	Q_3^{n+1}	Q_2^{n+1}	C	Q_1^{n+1}
0	0	0	0	0	1	0
0	0	1	0	1	0	0
0	1	0	0	1	1	0
0	1	1	1	0	0	0
1	0	0	0	0	0	1
1	0	1	0	1	0	0
1	1	0	0	1	0	0
1	1	1	0	0	0	0



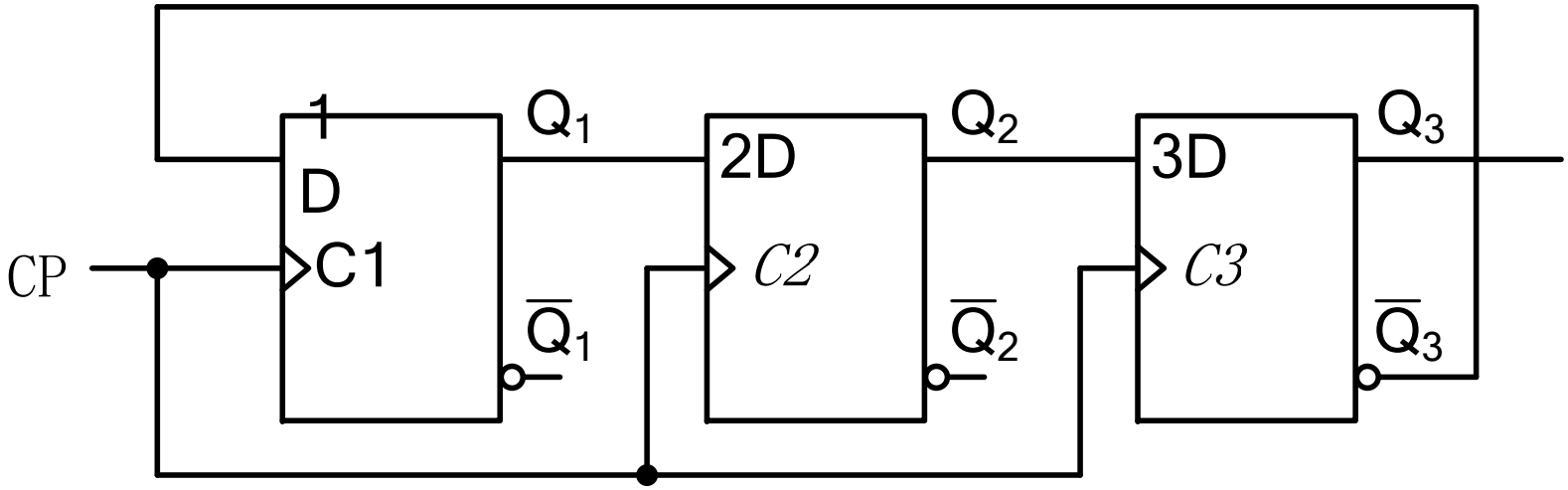
例 2 状态图

该电路的波形图如图 所示。



例 2 波形图

例 3 时序电路如图 所示，试分析其功能。



例 3 图

解 该电路仍为同步时序电路。

电路的激励方程为

$$D_1 = \overline{Q_3^n};\quad D_2 = Q_1^n;\quad D_3 = Q_2^n$$

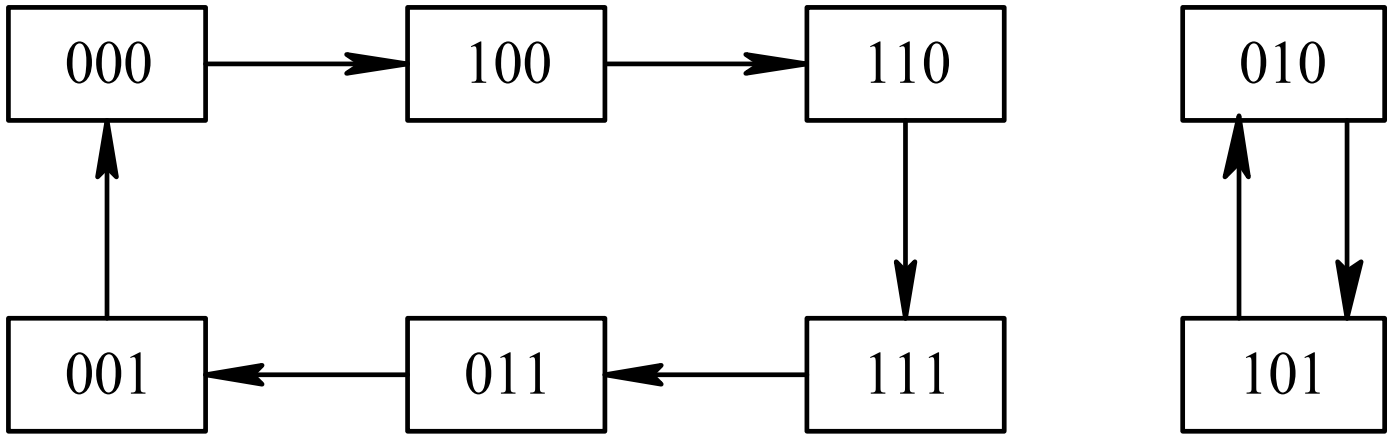
次态方程为

$$Q_1^{n+1} = \overline{Q_3^n};\quad Q_2^{n+1} = Q_1^n;\quad Q_3^{n+1} = Q_2^n$$

由此得出如表所示的状态真值表和如图所示的状态图。

例 3 状态真值表

Q_1^n	Q_2^n	Q_3^n	Q_1^{n+1}	Q_2^{n+1}	Q_3^{n+1}
0	0	0	1	0	0
0	0	1	0	0	0
0	1	0	1	0	1
0	1	1	0	0	1
1	0	0	1	1	0
1	0	1	0	1	0
1	1	0	1	1	1
1	1	1	0	1	1



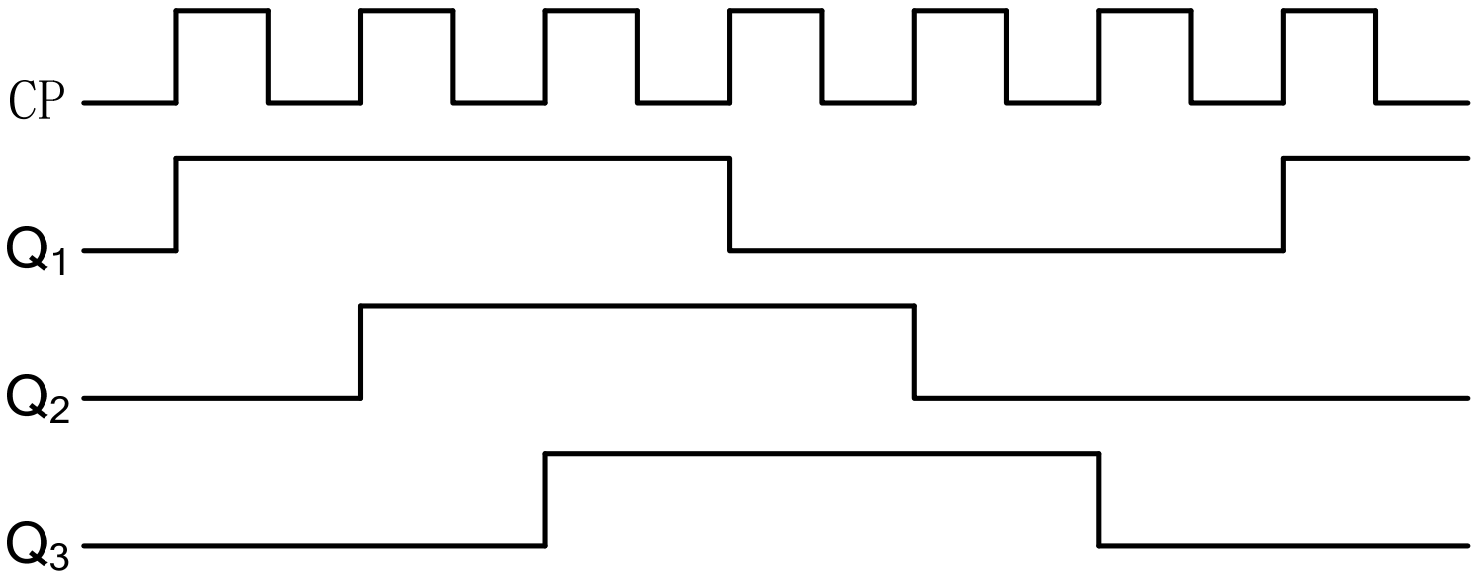
例 3 状态图

由状态迁移图可看出该电路为六进制计数器，又称为六分频电路，且无自启动能力。所谓分频电路是将输入的高频信号变为低频信号输出的电路。六分频是指输出信号的频率为输入信号频率的六分之一，即

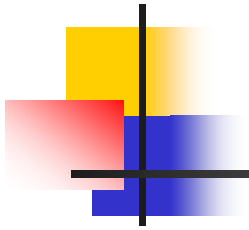
$$f_o = \frac{1}{6} f_{cp}$$

所以有时又将计数器称为分频器。

其波形图如图 所示。

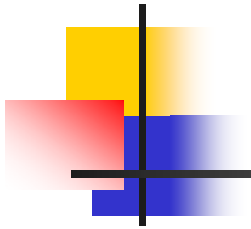


例 3 波形图



4.5 同步时序逻辑电路的设计

本节讨论的设计方法是一种经典的方法，即由小规模集成电路构成时序逻辑电路的设计方法，其基本指导思想是用尽可能少的触发器和门电路来完成设计。



同步时序电路设计的一般步骤为：

1. 作原始状态图和状态表；

2. 对原始状态表化简；

3. 状态分配；

4. 选定触发器，求出输出函数和激励函数表达式；

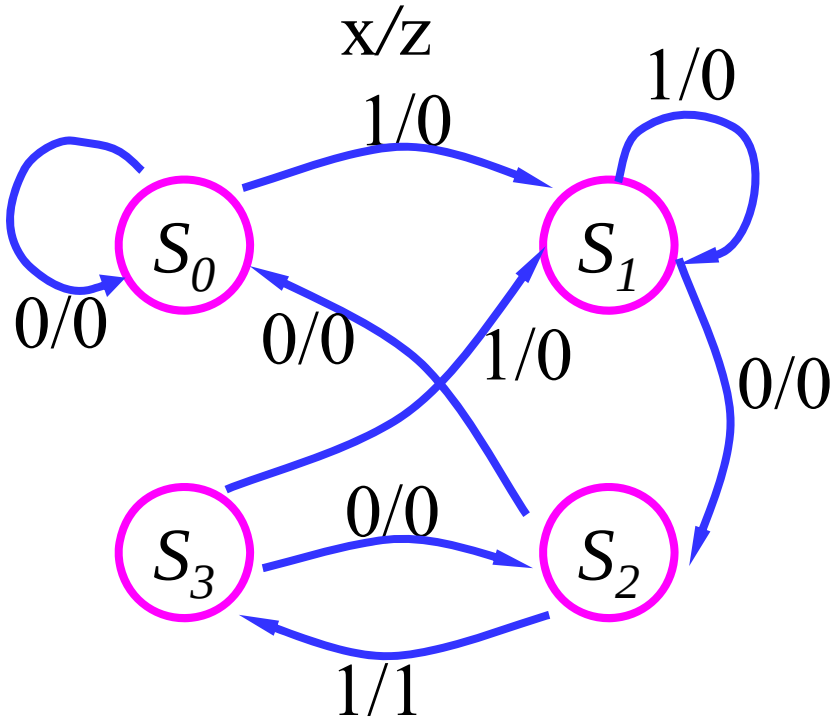
5. 画出逻辑电路图。

4.5.1 建立原始状态图

状态图是同步时序电路设计的依据，它必须正确反映设计要求。状态图的作法没有一成不变的方法，关键是要充分正确地理解设计要求。 一个正确的状态图是指状态图中的状态个数不能少，状态之间的转移关系不能错。

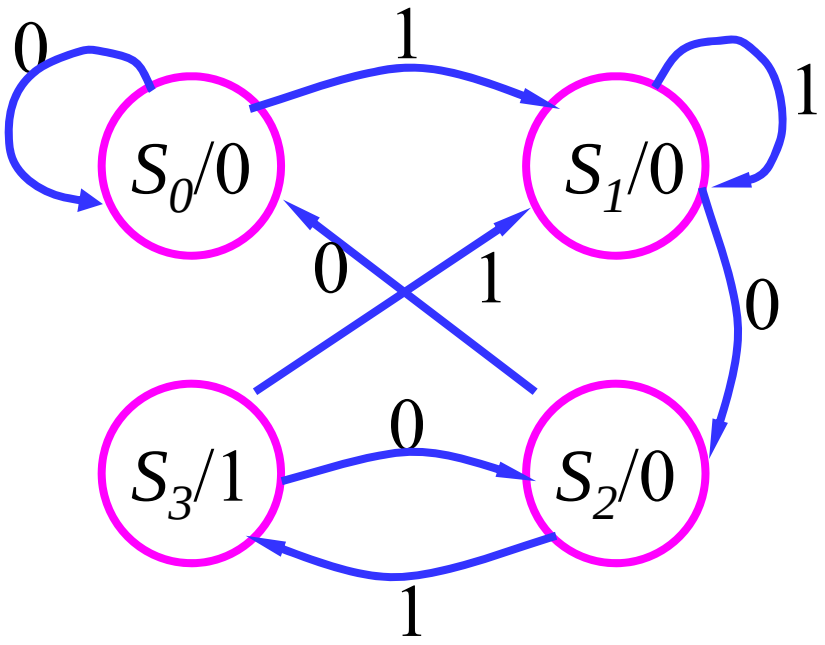
作状态图的一个基本方法是“对于感兴趣的东西应区分对待，对于不感兴趣的东西可不予区分”

例：　某序列检测器有一个输入端x和一个输出端Z。从x端输入一组按时间顺序排列的串行二进制码。当输入序列中出现101时，输出Z＝1，否则Z＝0。试作出该序列检测器的Mealy型和Moore型原始状态图和状态表。


x/z
1/0
1/0
S_0
S_1
0/0
0/0
1/0
0/0
S_3
0/0
S_2
1/1

电路的Mealy 型状态表

现态	次态/输出	
	x=0	x=1
S_0	S_0/0	S_1/0
S_1	S_2/0	S_1/0
S_2	S_0/0	S_3/1
S_3	S_2/0	S_1/0

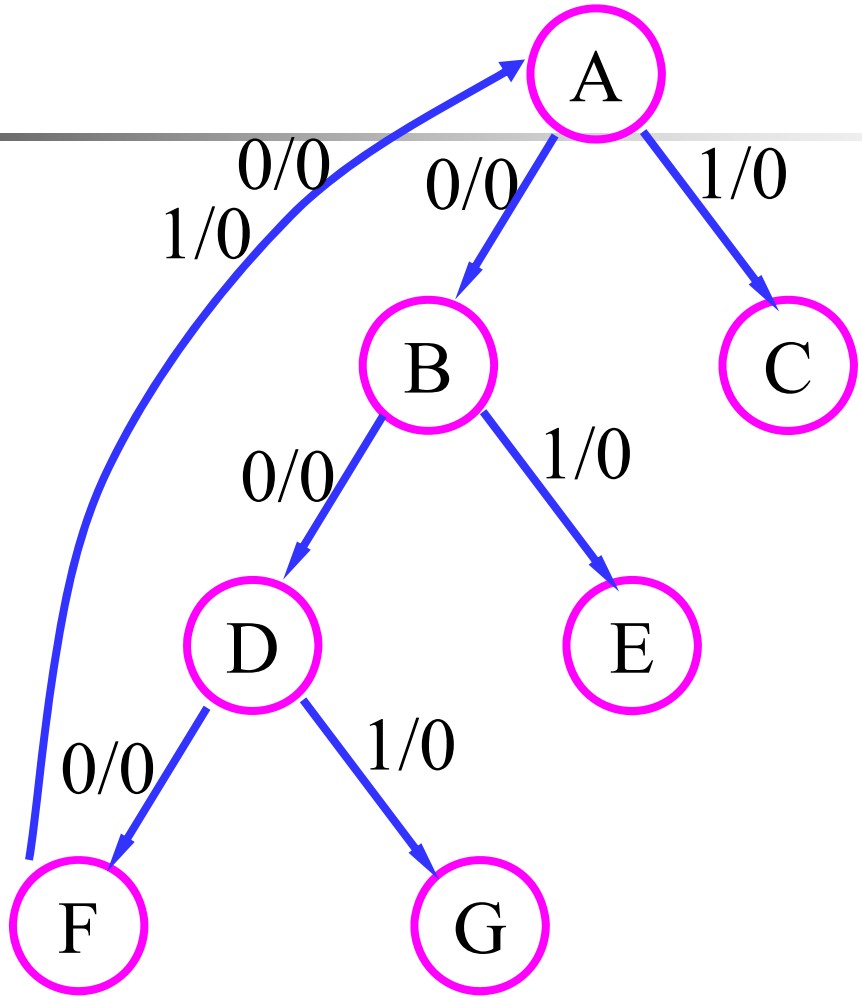

0
1
1
$S_0/0$
$S_1/0$
0
1
0
$S_3/1$
0
$S_2/0$
1

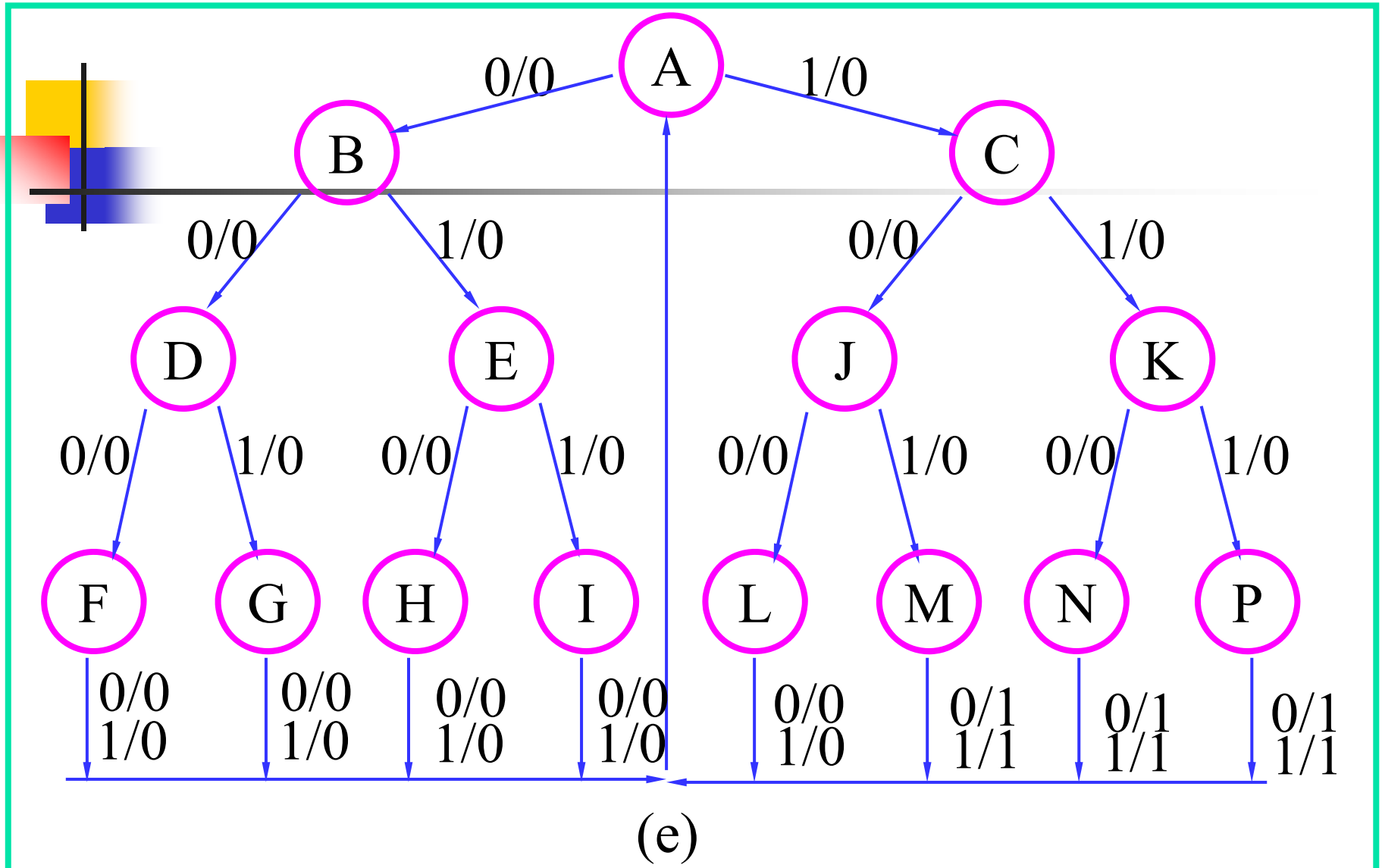
电路的 Moore 型状态表

现态	次　态		输　出 Z
	x=0	x=1	
S_0	S_0	S_1	0
S_1	S_2	S_1	0
S_2	S_0	S_3	0
S_3	S_2	S_1	1

例：　假设某同步时序电路，用于检测串行输入的8431BCD码，其输入的顺序是先高位后低位，当出现非法数字(即输入1010，1011，1100，1101，1110，1111)时，电路的输出为1。试作出该时序电路的Mealy型原始状态图和状态表。

解：





(e)

电路的原始状态图

电路的原始状态图

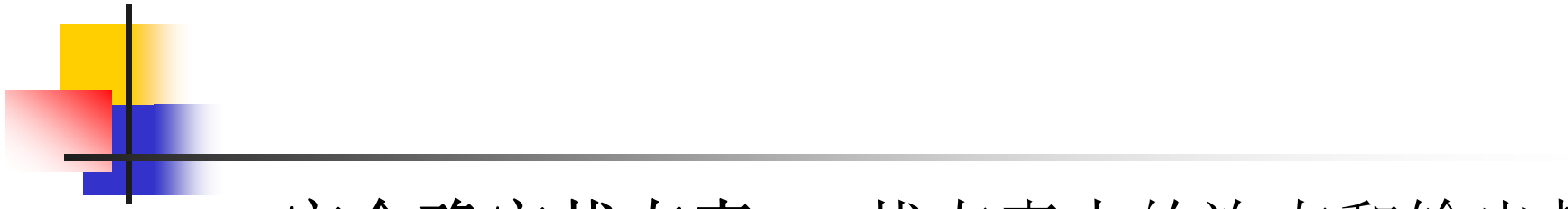
现态	次态/输出	
	x=0	x=1
A	B/0	C/0
B	D/0	E/0
C	J/0	K/0
D	F/0	G/0
E	H/0	I/0
F	A/0	A/0
G	A/0	A/0
H	A/0	A/0
I	A/0	A/0
J	L/0	M/0
K	N/0	P/0
L	A/0	A/0
M	A/1	A/1
N	A/1	A/1
P	A/1	A/1

4.5.2 状态简化

一般情况下，原始状态图和原始状态表中存在着多余的状态。状态个数越多，电路中所需的触发器的数目也越多，制造成本就越高。为降低制造成本，需要去掉多余的状态，即要进行**状态简化**。

所谓**状态简化**，就是要获得一个最小化的状态表。这个表不仅能正确地反映设计的全部要求，而且状态的数目最少。

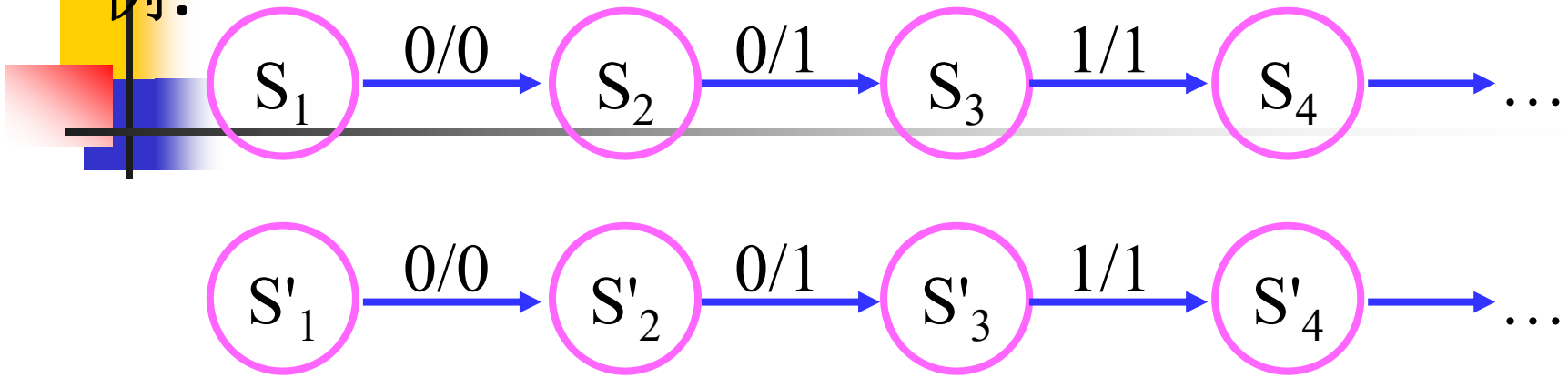
一、完全确定状态表的简化



完全确定状态表： 状态表中的次态和输出都有确定的状态和确定的输出值。

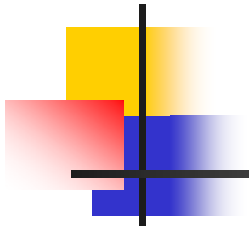
等效状态：设状态S1和S2是完全确定状态表中的两个状态,如果对于所有可能的输入序列，分别从状态S1和状态S2出发，所得到的输出响应序列完全相同，则状态S1和S2是等效的，记作(S1, S2)， 或者说，状态S1和S2是等效对。等效状态可以合并。这里“ 所有可能的输入序列” 是指长度和结构是任意的，它包含无穷多位，且有无穷多种组合。

例：



S_1, S_2等效

- 等效状态传递性：(S_1,S_2), (S_2,S_3)->(S_1,S_3)
- 等效类： 彼此等效的状态集合
- 最大等效类： 不被其它等效类所包含的等效类。一个状态也可能是一个最大等效类。



状态简化的任务是要在原始状态表中找出全部最大等效类(最大等效类集合)，并将每一个最大等效类用一个状态来表示。

直接根据定义来寻找等效状态或等效类是不现实的，实际上是从状态表中来寻找的，因为原始状态表已在总体上反映了各个状态在任意输入序列下的输出。

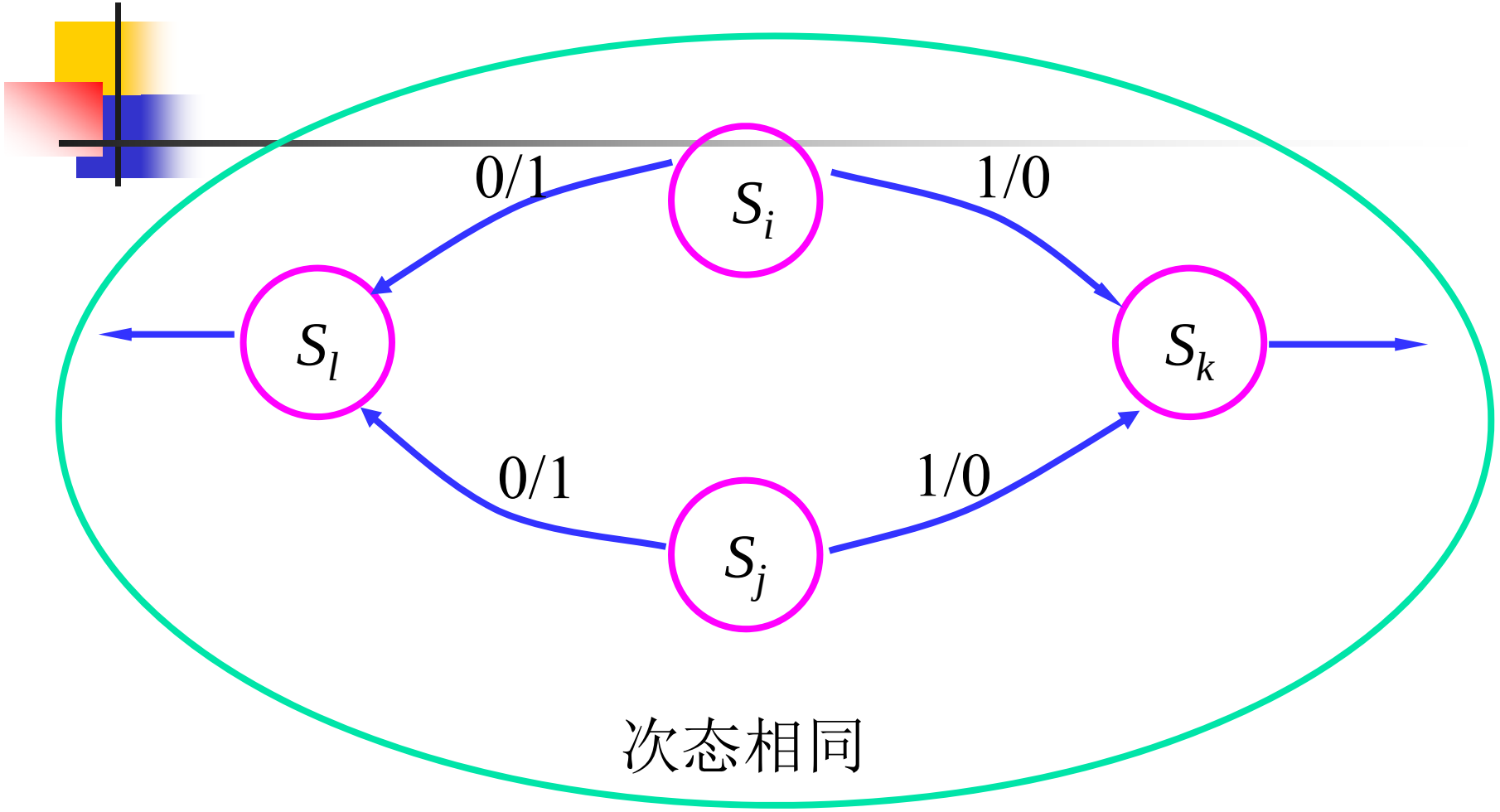
判别方法：

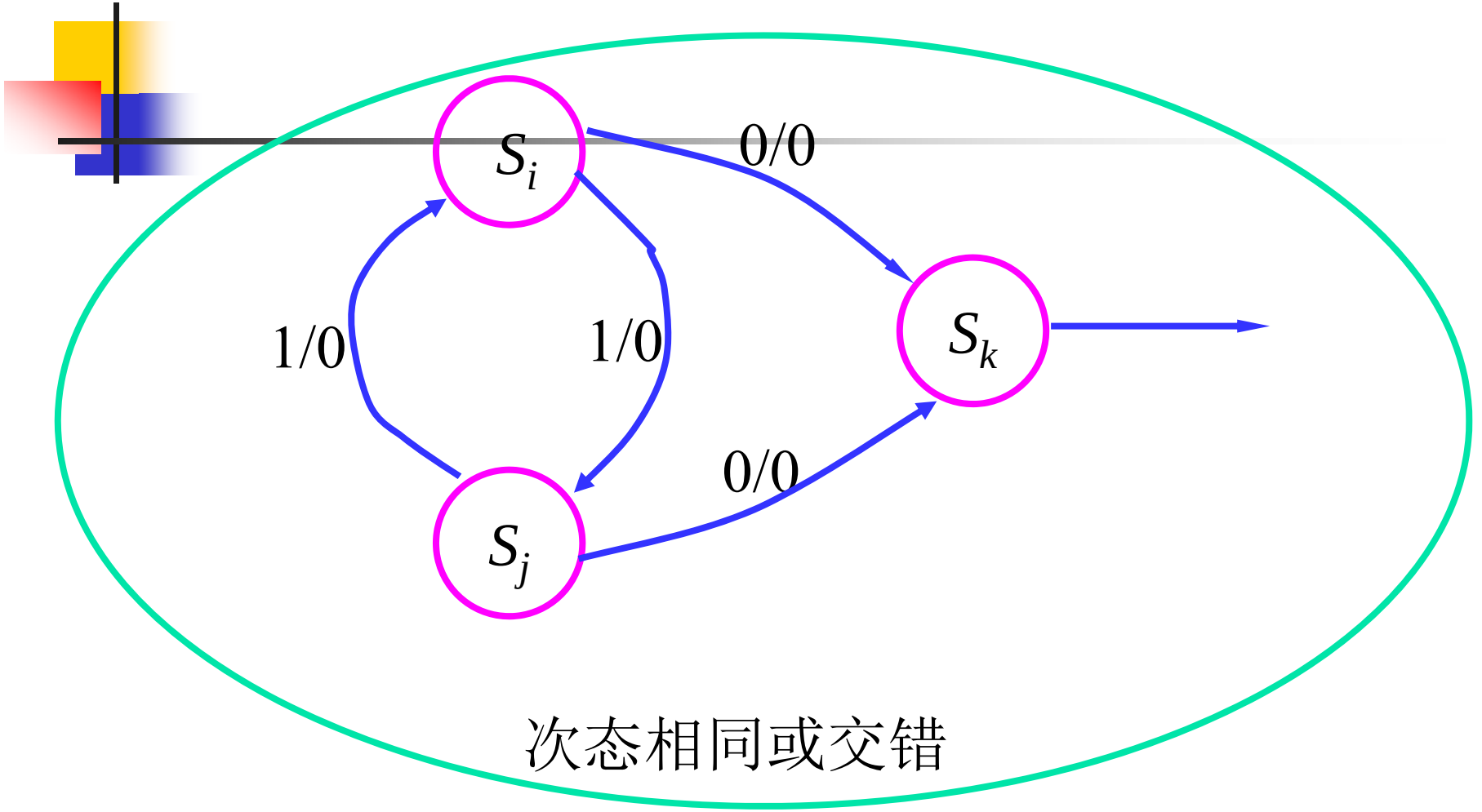
假定状态S_1和S_2是完全确定原始状态表中的两个现态，那么S_1和S_2等效的条件可归纳为在输入的各种取值组合下：

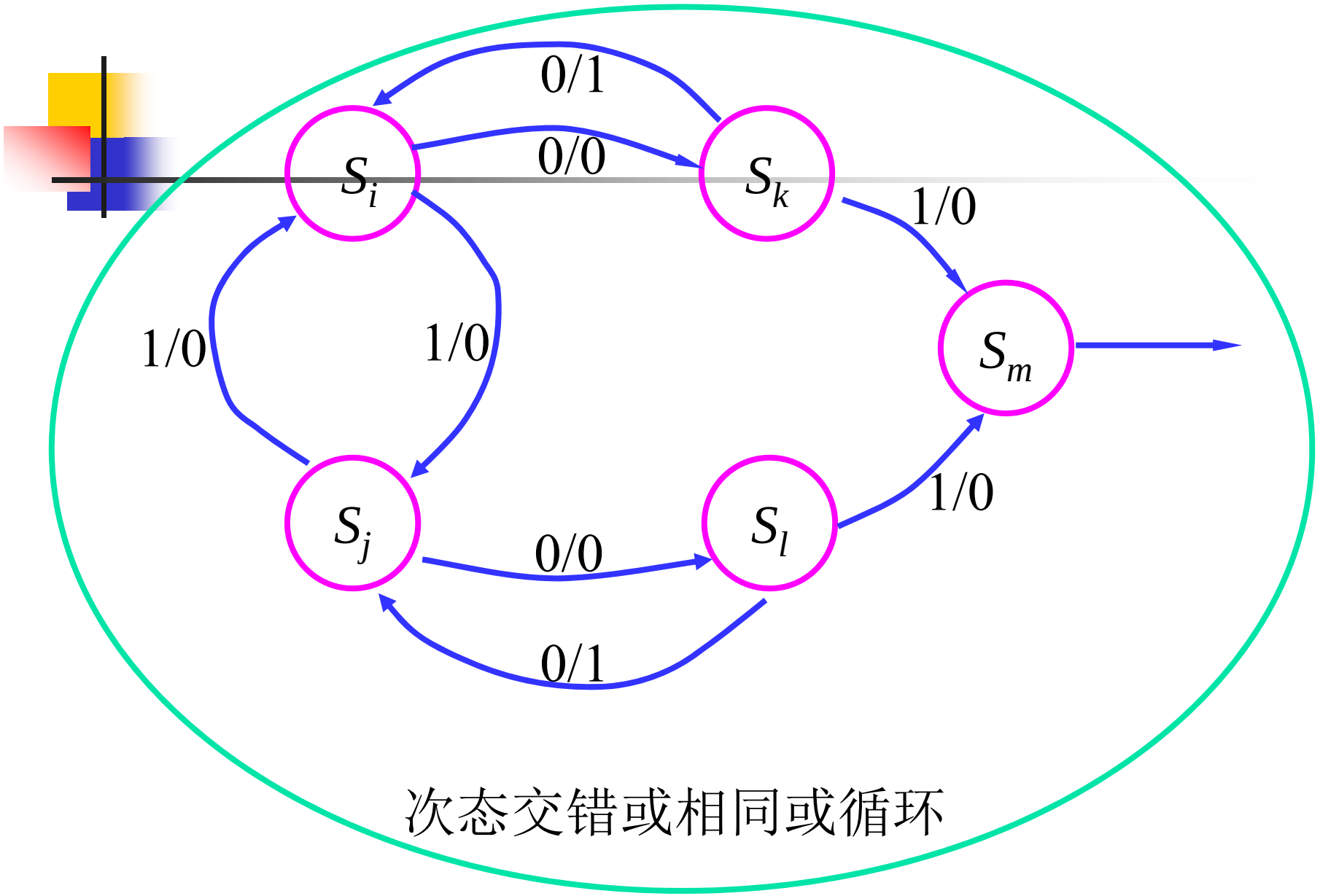
第一、它们的输出完全相同；

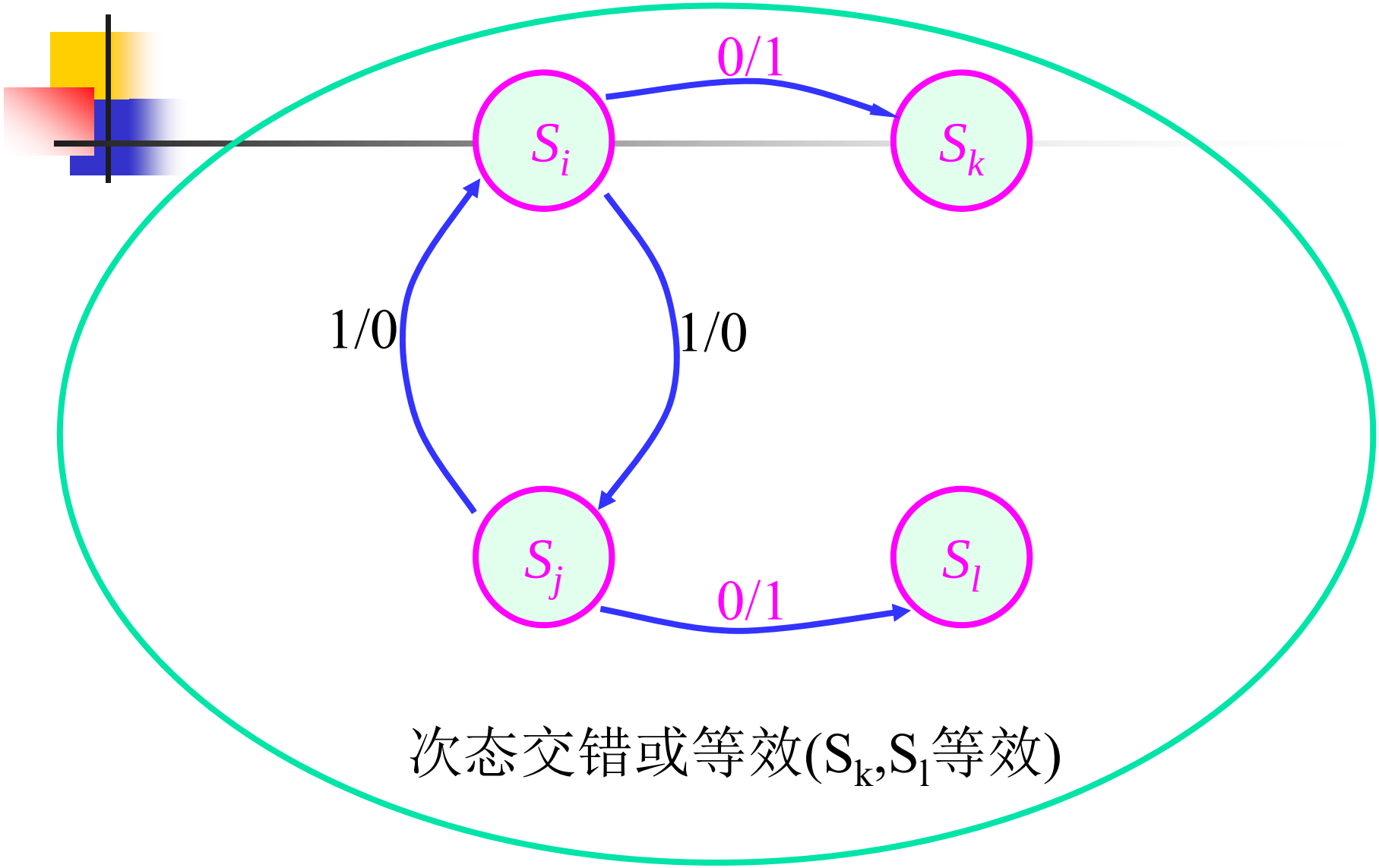
第二、它们的次态满足下列条件之一，即

(1) 次态相同；(2) 次态交错；(3) 次态循环；(4) 次态对等效。


0/1
1/0
S_i
S_l
S_k
0/1
1/0
S_j
次态相同


S_i
0/0
S_k
1/0
1/0
0/0
S_j
次态相同或交错


0/1
0/0
S_i
S_k
1/0
1/0
1/0
S_m
S_j
S_l
1/0
0/0
0/1
次态交错或相同或循环



次态交错或等效(S_k,S_l等效)

1. 观察法化简

例：简化下表所示的状态表

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

解：

A和B，C和D的输出完全相等；

C和D在输入的各种取值组合下，次态相同，因此C和D等效；

A和B在x=1时的次态不满足四条件之一，因此A和B不等效;

最大等效类为{A}，{B}，{C,D},分别用A', B', C'表示；

最小化状态表为

现态	次态/输出	
	x=0	x=1
A'	A'/0	B'/0
B'	A'/0	C'/0
C’	A'/0	C'/1

1. 隐含表法化简

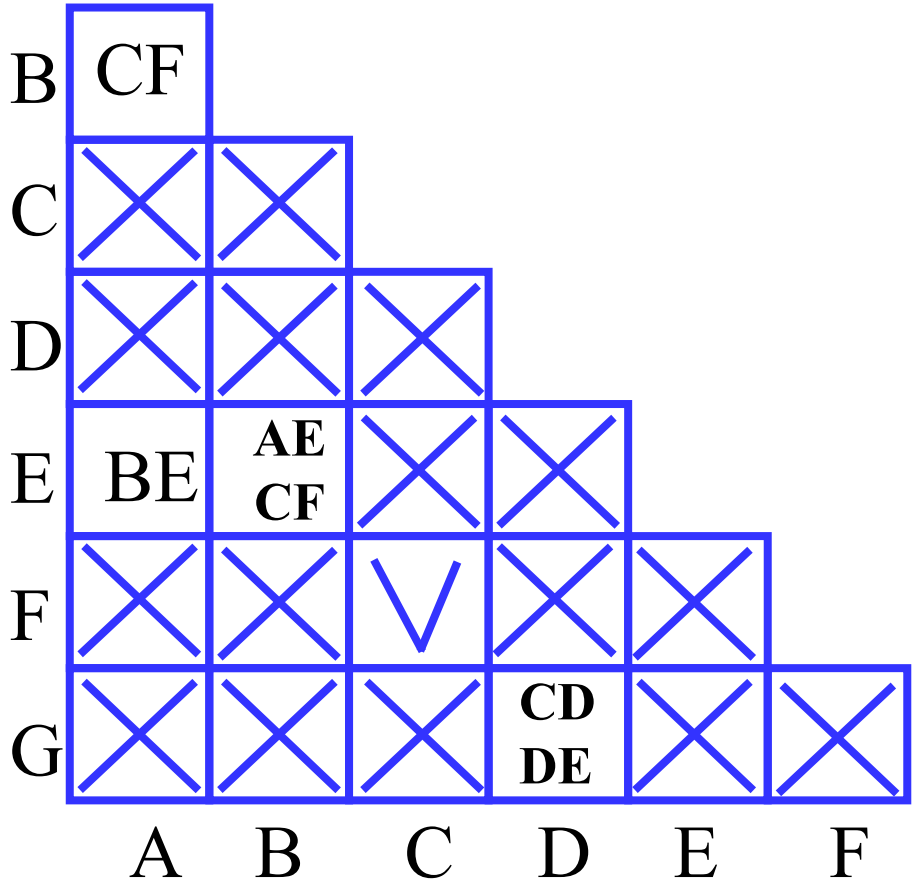
例：简化下表所示的状态表

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

解： 作隐含表

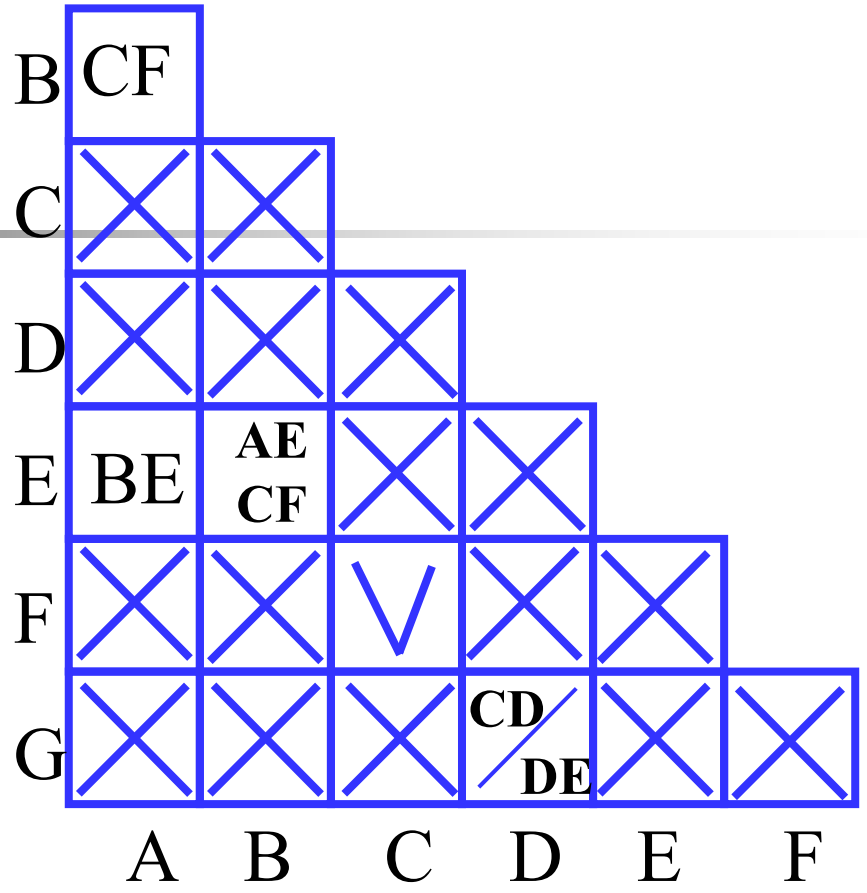
顺序比较，寻找等效状态对

- 状态对等效，打“ √”；
- 状态对不等效，打“ X”;
- 状态对是否等效需进一步检查，则标记次态对。



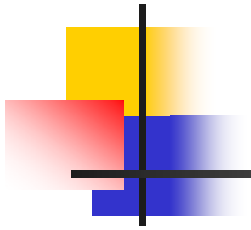
关联比较，

确定等效状态对



AE->BE->CF√

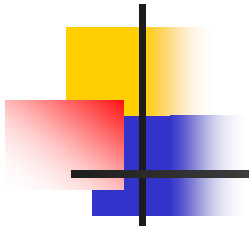
处于循环链中的每一个状态对都是等效状态对。



确定最大等效类，作最小化状态表

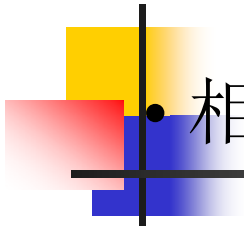
- 四个等效对 (A,B), (A,E), (B,E), (C,F)
- 四个最大等效类 (A,B,E), (C,F), (D), (G)
- 令以上四个最大等效类依次为a, b, c, d.

现态	次态/输出	
	x=0	x=1
a	b/0	a/1
b	c/0	d/0
c	c/1	a/0
d	b/1	c/0

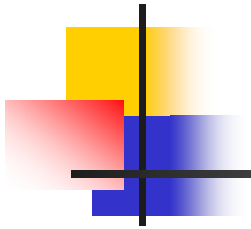


二、不完全确定状态表的简化

不完全确定状态表： 状态表中存在不确定的次态或输出，这些不确定的次态或输出将有利于状态简化。

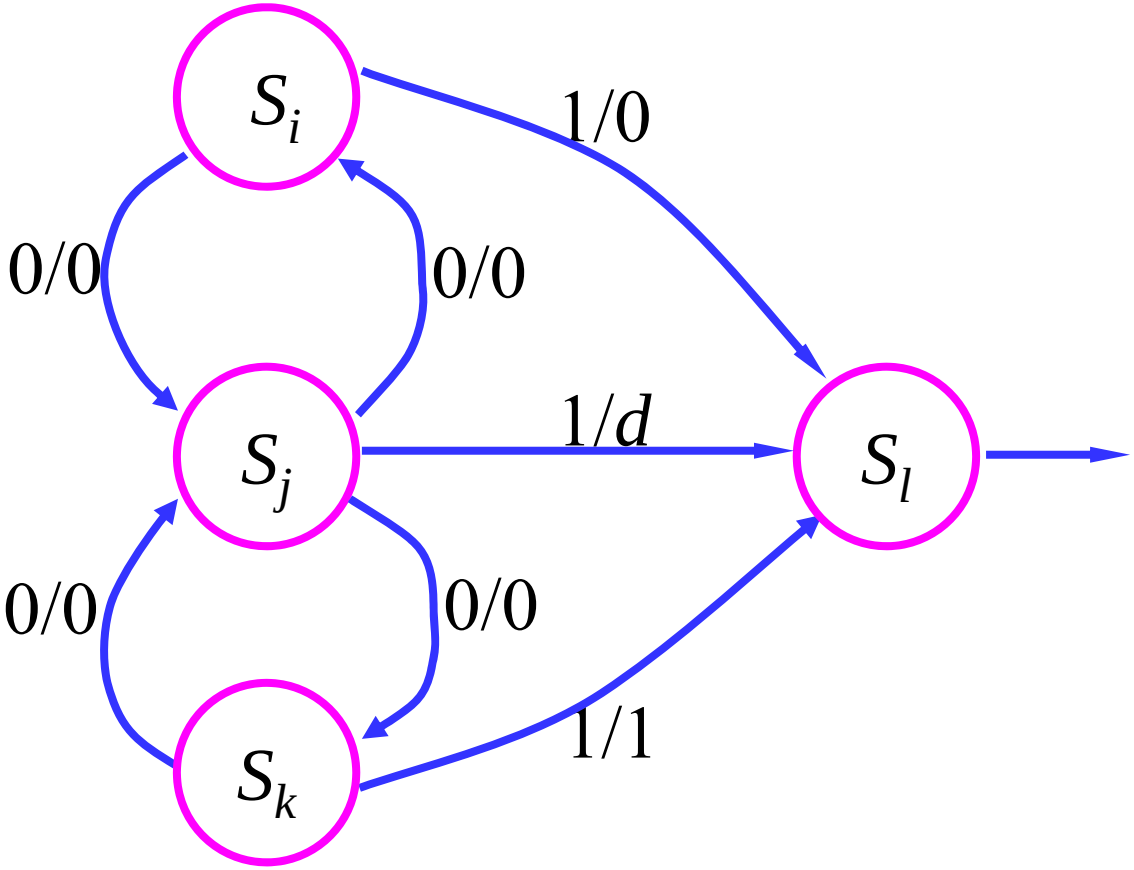


- 相容状态：设状态S_1和S_2是不完全确定状态表中的两个状态，如果对于所有的有效输入序列，分别从状态S_1和S_2出发，所得到的输出响应序列(除不确定的那些位之外)是完全相同的，那么状态S_1和S_2是相容的，或者说状态S_1和S_2是相容对，记作(S_1,S_2)。相容状态可以合并。



这里所谓有效输入序列的含义是：从状态表中的状态S出发，如果给定某输入序列所得的状态响应序列除最后一个次态外，其它状态都是确定的，那么这个输入序列对状态S是有效的。而所有的有效输入序列，则意味着有效输入序列的长度和结构是任意的。

- 相容状态无传递性：



S_i和S_j相容；

S_j和S_k相容；

但S_i和S_k不相容。

- 相容类：彼此相容的状态集合
- 最大相容类：不被其它相容类所包含的相容类

判别方法：

在不完全确定状态表中判断两个状态是否相容也是根据表中给出的次态和输出来决定的。假定状态S_i和S_j是不完全确定状态表中的两个现态，那么状态S_i和S_j相容的条件可归纳为在输入的各种取值组合下。

第一、它们的输出完全相同，或者其中的一个(或两个) 输出为任意值。

第二、它们的次态满足下列条件之一：

(1) 次态相容；

(2) 次态交错；

(3) 次态循环；

(4) 其中的一个(或两个)为任意状态

(5) 次态对相容；

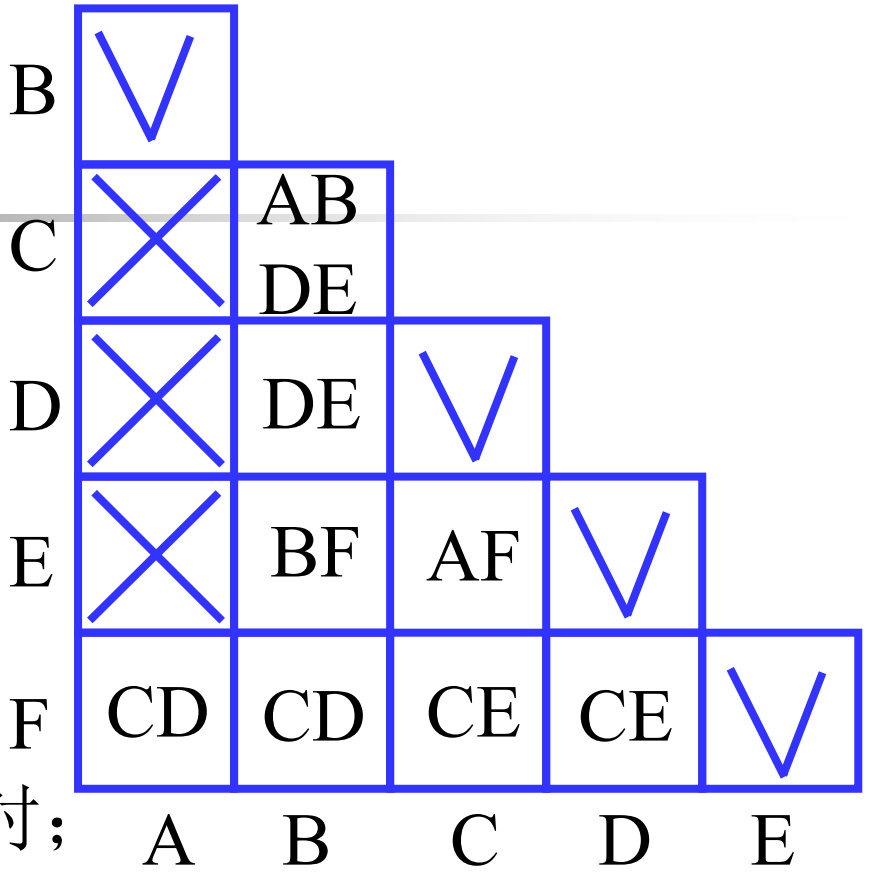
例：简化下表所示的状态表

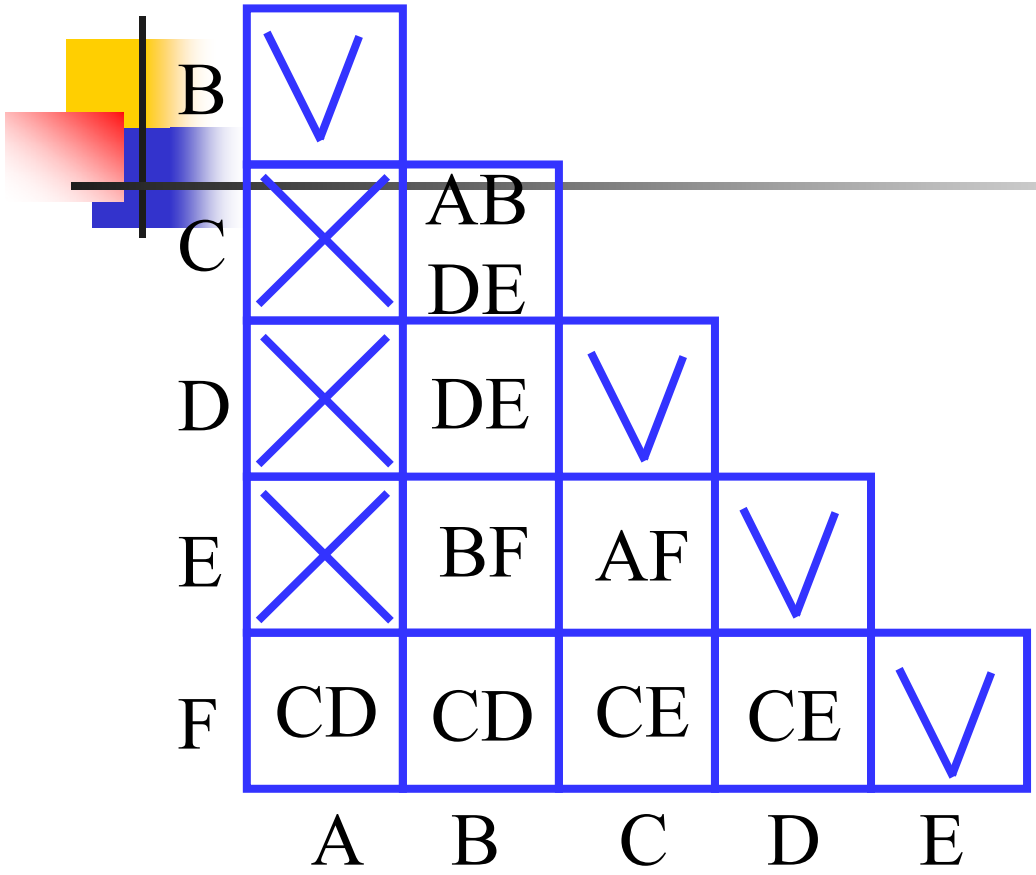
现 态	次 态		输 出
	x=0	x=1	
A	B	D	0
B	B	D	d
C	A	E	1
D	d	E	1
E	F	d	1
F	d	C	d

解：

作隐含表；

顺序比较，寻找相容对；





AF->CD √

BC->AB √

DE √

BD->DE √

BE->BF->CD √

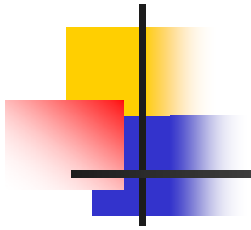
CE->AF √

CF->CE √

DF->CE √

关联比较，确定相容对；

以上三步与确定状态表的化简相同



全部相容状态对：

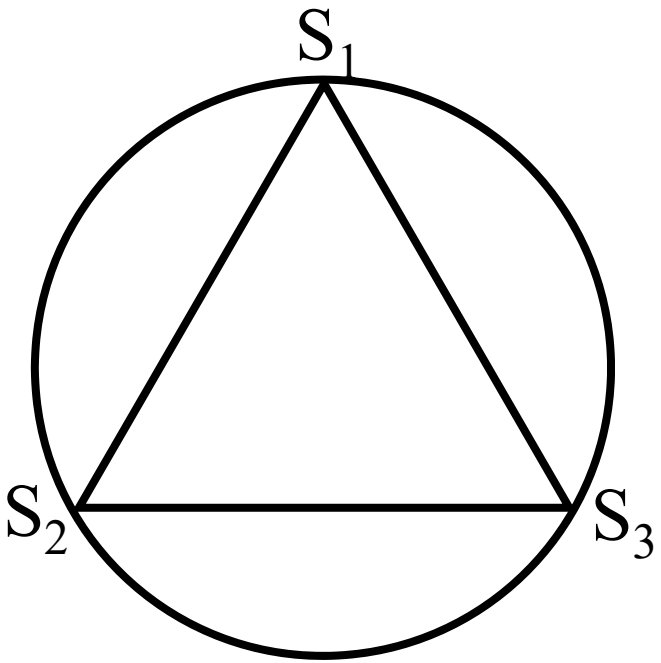
(A,B), (A,F), (B,C),

(B,D), (B,E), (B,F),

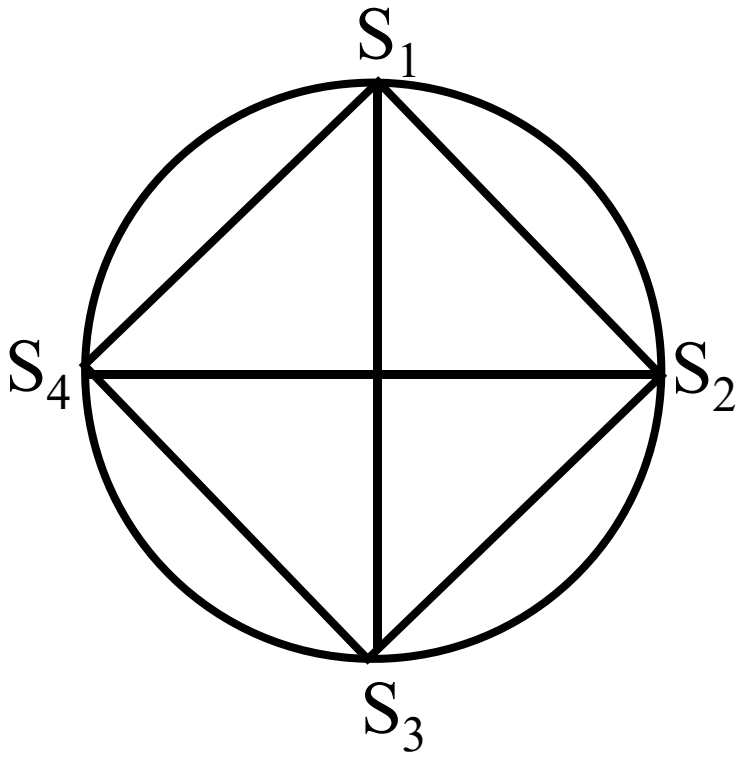
(C,D), (C,E), (C,F),

(D,F), (D,E), (E,F)。

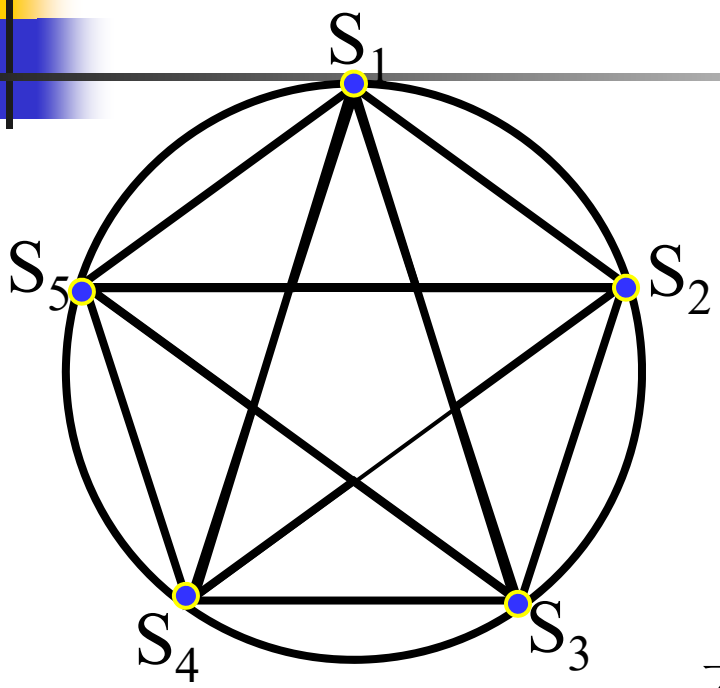
作状态合并图，求最大相容类。



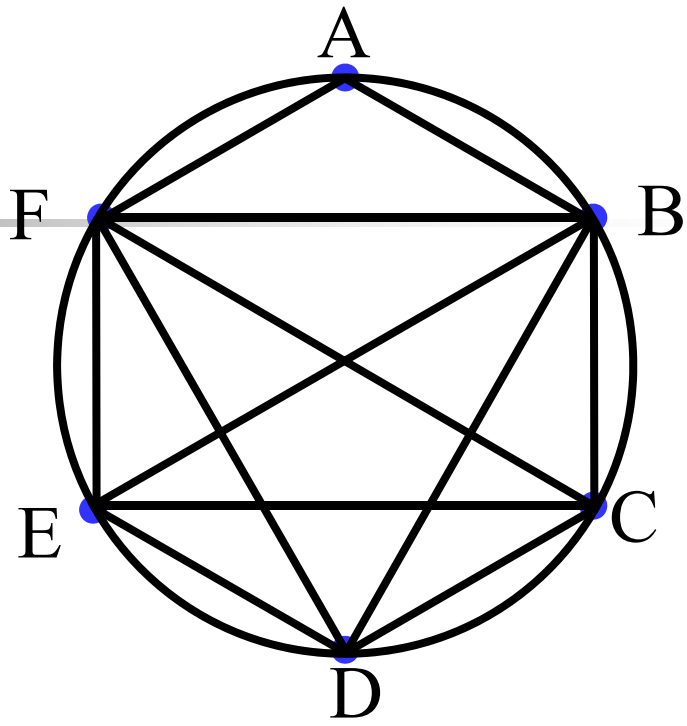
3状态相容



4状态相容



5状态相容



本例状态合并图，最大相容类是(A,B,F), (B,C,D,E,F)。

作最小化状态表。最小化状态表(又称最小闭覆盖) 应满足下列三个条件：

- 覆盖性——所选相容类集合应包含原始状态表中的全部状态。
- 最小性——所选相容类集合中相容类的个数应最少。
- 闭合性——所选相容类集合中的任一相容类，在原始状态表中任一输入条件下产生的次态应该属于该集合中的某一个相容类。

我们常采用闭覆盖表来反映所选相容类集合的覆盖和闭合情况。本例的闭覆盖表为

最大相容类	覆盖						闭合	
	A	B	C	D	E	F	x=0	x=1
ABF	√	√				√	B	CD
BCDEF		√	√	√	√	√	ABF	CDE

所选相容类集合{(A,B,F), (B,C,D,E,F)} 满足最小闭覆盖条件，令A表示(A,B,F)，C表示(B,C,D,E,F)可得：

现 态	次 态		输 出
	x=0	x=1	
A	A,C	C	0
C	A	C	1

由于该表中只有两个状态，进一步可以得到：

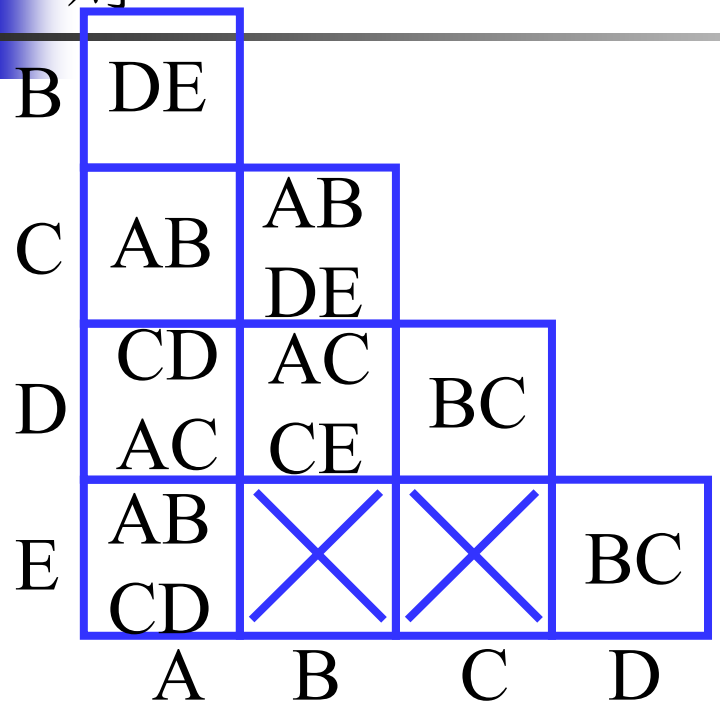
现 态	次 态		输 出
	x=0	x=1	
A	d	C	0
C	A	C	1

例：化简下表所示的状态表

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

解：　作隐含表，寻找相容状态

对



AB->DE ->BC √

AC->AB √

AD-> CD ->BC √
AC √

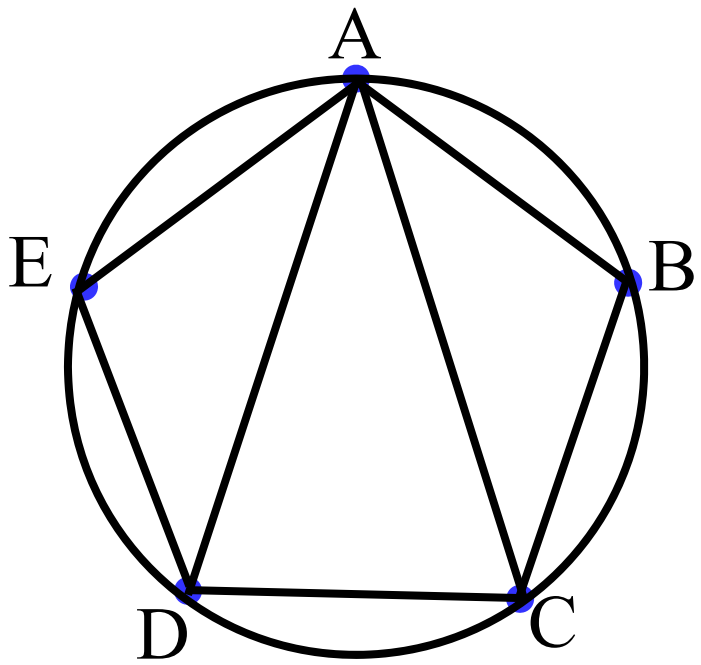
AE-> AB √
CD √

BD-> AC √
CE X

由上图得相容状态对为(A,B), (A,C), (A,D), (A,E),
(B,C), (C,D), (D,E)

(A,B), (A,C), (A,D), (A,E), (B,C), (C,D), (D,E)

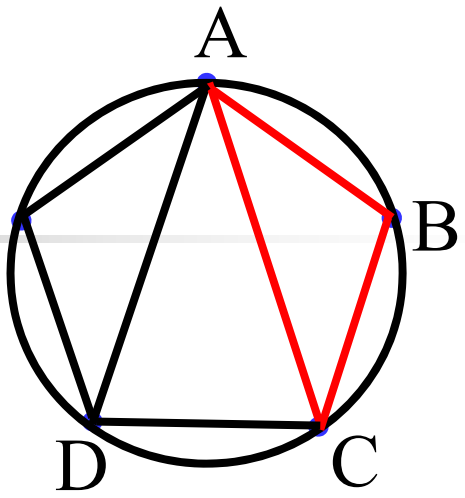
作状态合并图，寻找最大相容类



得最大相容类为(A,B,C), (A,C,D), (A,D,E)

作最小化状态表

若选相容类集合为{(A,B,C), (A,C,D)}则下表表明它不满足闭合要求

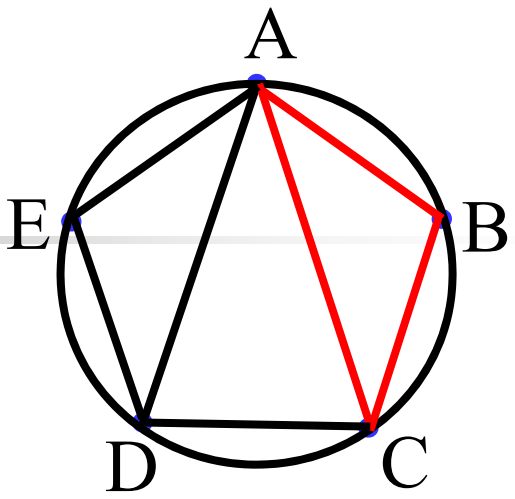


相容类	覆盖					闭合	
	A	B	C	D	E	x=0	x=1
ABC	√	√	√			DE	AB
ADE	√			√	√	CD	ABC

但如果选相容类

(A,B,C)和 (D,E)则能满

足最小闭覆盖的要求

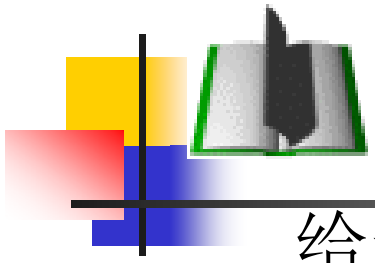


相容类	覆　盖					闭　合	
	A	B	C	D	E	x=0	x=1
ABC	A	B	C			DE	AB
DE				D	E	C	BC

令A'=(A,B,C), B'=(D,E), 进一步可得：

现态	次态/输出	
	x=0	x=1
A'	B' /0	A'/d
B'	A' /1	A'/d

寻找最小闭覆盖通常不是一件容易的事情，其结果往往不唯一。



4.5.3 状态编码(状态分配)

给最小化状态表中的每一个状态指定一个二进制代码，形成二进制状态表。

通常情况下，状态编码的方案不一样，所得到的输出函数和激励函数的表达式也不同，由此而设计出来的电路复杂度也不同。状态分配的任务是：　　决定编码的长度

寻找一种最佳的或接近最佳的状态分配方案。

设最小化状态表中的状态数为N，编码长度为n，N和n的关系为 $2^{n-1}<N<2^{n}$

用2^{n}种组合来对N个状态进行分配时，可能出现的分配方案的总数K_s为

$$K_s = A_{2^n}^{N} = \frac{2^n!}{(2^n - N)!}$$

例如，n=2, N=4时 有

状态	方案											
	1	2	3	4	5	6	7	8	9	10	11	12
A	00	10	01	11	00	01	10	11	00	10	01	11
B	01	11	00	10	10	11	00	01	11	01	10	00
C	11	01	10	00	11	10	01	00	01	11	00	10
D	10	00	11	01	01	00	11	10	10	00	11	01

状态	方案											
	1	2	3	4	5	6	7	8	9	10	11	12
A	00	10	01	11	00	01	10	11	00	10	01	11
B	01	11	00	10	10	11	00	01	11	01	10	00
C	11	01	10	00	11	10	01	00	01	11	00	10
D	10	00	11	01	01	00	11	10	10	00	11	01

但是，在K_s种方案中只有三种是独立的(真正不相同的)方案

$$K_u = \frac{(2^n - 1)!}{(2^n - N)!n!}$$

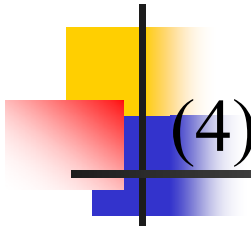
然而，当n较大时，K_u仍然很大，要真正找到最佳的分配方案是十分困难的，况且分配方案的好坏还与所采用的触发器的类型有关。因此，实际应用时都是采用工程的方法，依据以下四条件原则来进行状态分配。

状态分配的基本原则有四条：

(1) 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) 在相邻输入条件，同一现态的次态应尽可能分配相邻的二进制代码；

(3) 输出完全相同的现态应尽可能分配相邻的二进制代码；



(4) 最小化状态表中出现次数最多的状态或初始状态应分配逻辑0。

一般情况下，第一条原则较为重要，需优先考虑，其次要考虑由前三条原则得到的应分配相邻代码的状态对出现的次数，次数多的状态对应优先分配相邻的二进制代码。

例：对下表所示的状态表进行状态分配

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

解：

确定n=2

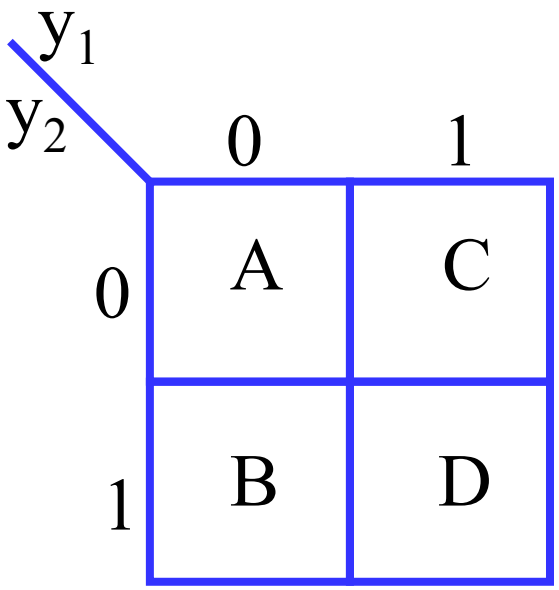
确定分配

由规则(1) 得 A和B，A和C 应相邻；

由规则(2)得 C和D，C和A，B和D，A和B应相邻；

由规则(3)得 A，B，C 三者应相邻，即A和B，A和C，B和C应相邻；

由规则(4)得A分配为逻辑0

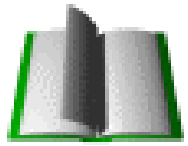


	y_1	y_2
A:	0	0
B:	0	1
C:	1	0
D:	1	1

最后我们可以得到二进制状态表

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

注意：有时满足分配原则的分配方案不唯一，这时可任选一种。



4.5.4 确定激励函数和输出函数

1. 触发器的激励表

触发器的激励表反映触发器从某种现态转换到某种次态时，对触发器输入(激励)的要求。在这种表中，现态和次态作为自变量，输入(激励)作为因变量。触发器的激励表可由触发器的状态表直接推出。

R-S触发器激励表

$Q->Q^{(n+1)}$		R	S
0	0	d	0
0	1	0	1
1	0	1	0
1	1	0	d

D触发器激励表

$Q->Q^{(n+1)}$		D
0	0	0
0	1	1
1	0	0
1	1	1

J-K触发器激励表

$Q->Q^{(n+1)}$		J	K
0	0	0	d
0	1	1	d
1	0	d	1
1	1	d	0

T触发器激励表

$Q->Q^{(n+1)}$		T
0	0	0
0	1	1
1	0	1
1	1	0

2. 确定激励函数

两种方法：根据次态方程来确定和通过激励表来确定。常采用后一种方法。

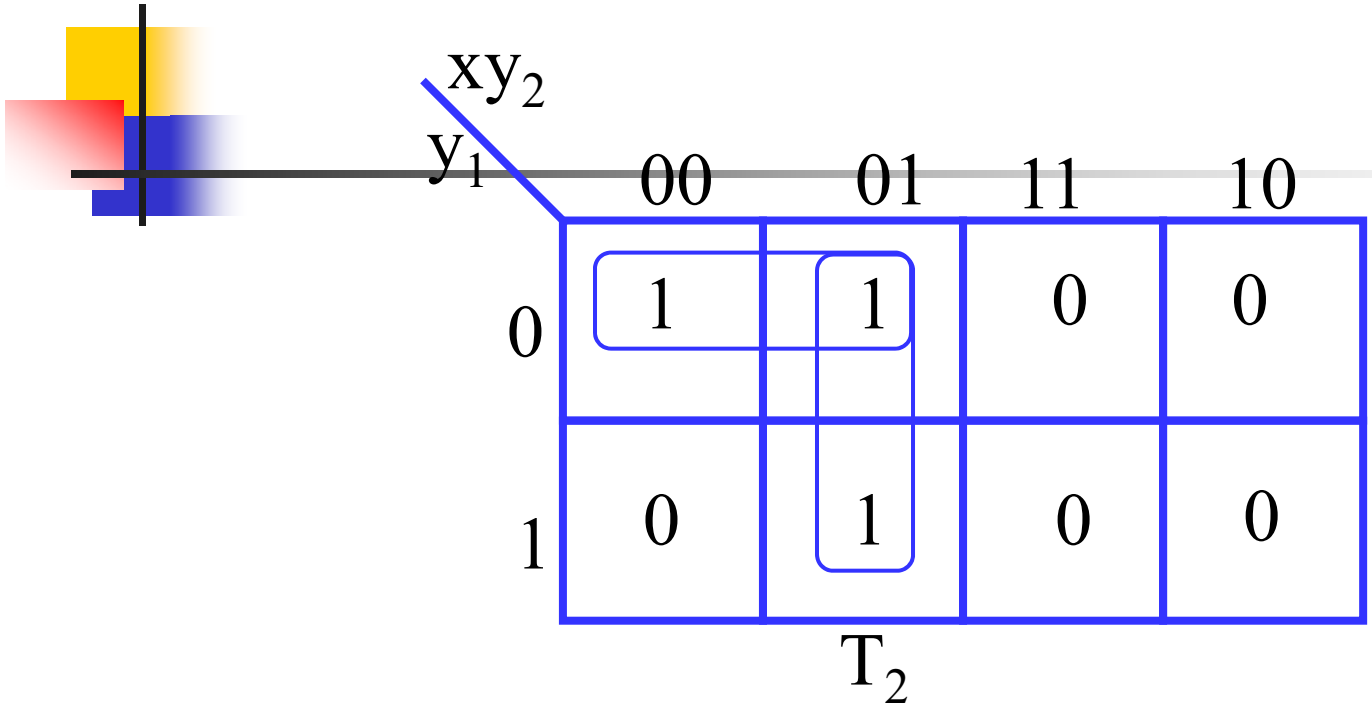
例：若用T触发器实现下表所示的二进制状态表，试写出激励和输出函数。

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

解： 确定激励函数

$Q->Q^{(n+1)}$		T
0	0	0
0	1	1
1	0	1
1	1	0

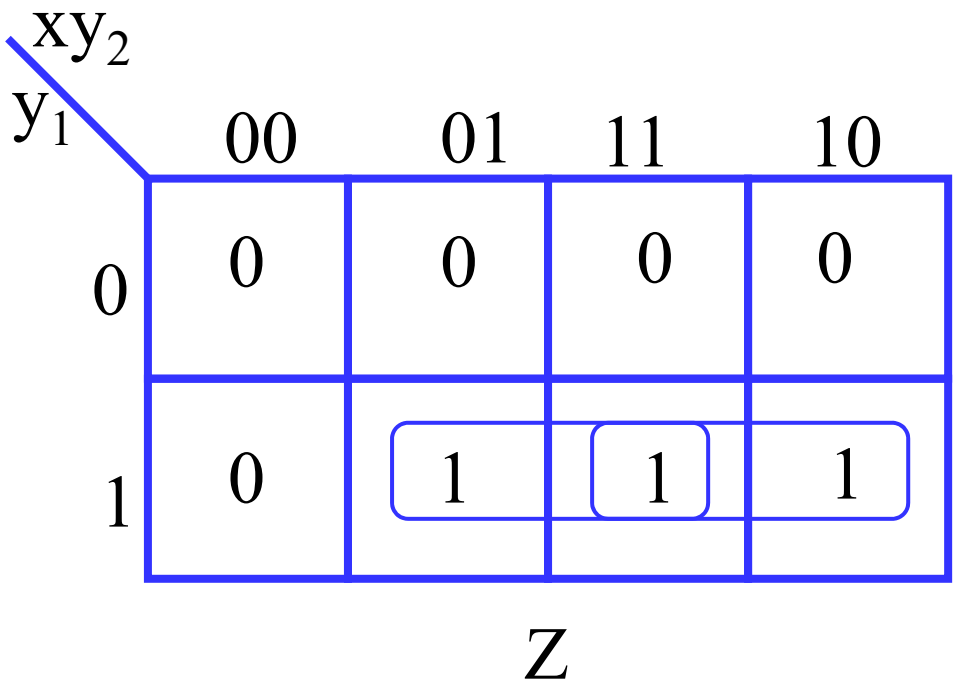
输入	现态		次态		激励函数	
x	y_2	y_1	$y_2^{(n+1)}$	$y_1^{(n+1)}$	T_2	T_1
0	0	0	1	1	1	1
0	0	1	0	0	0	1
0	1	1	0	0	1	1
0	1	0	0	1	1	1
1	0	0	0	1	0	1
1	0	1	0	0	0	1
1	1	1	1	0	0	1
1	1	0	1	1	0	1



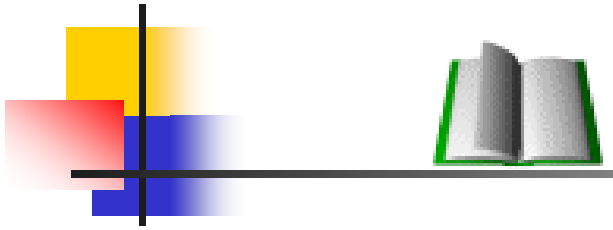
$$T_2 = \overline{x}\,\overline{y_1} + \overline{x}\,y_2$$

$$T_1 = 1$$

确定输出函数

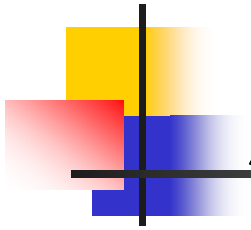


$Z=y_2y_1+xy_1$



4.5.5 画出逻辑电路图

先画出触发器并给触发器编号，再根据激励函数和输出函数画出组合逻辑部分的电路，最后画出同步时钟信号线。

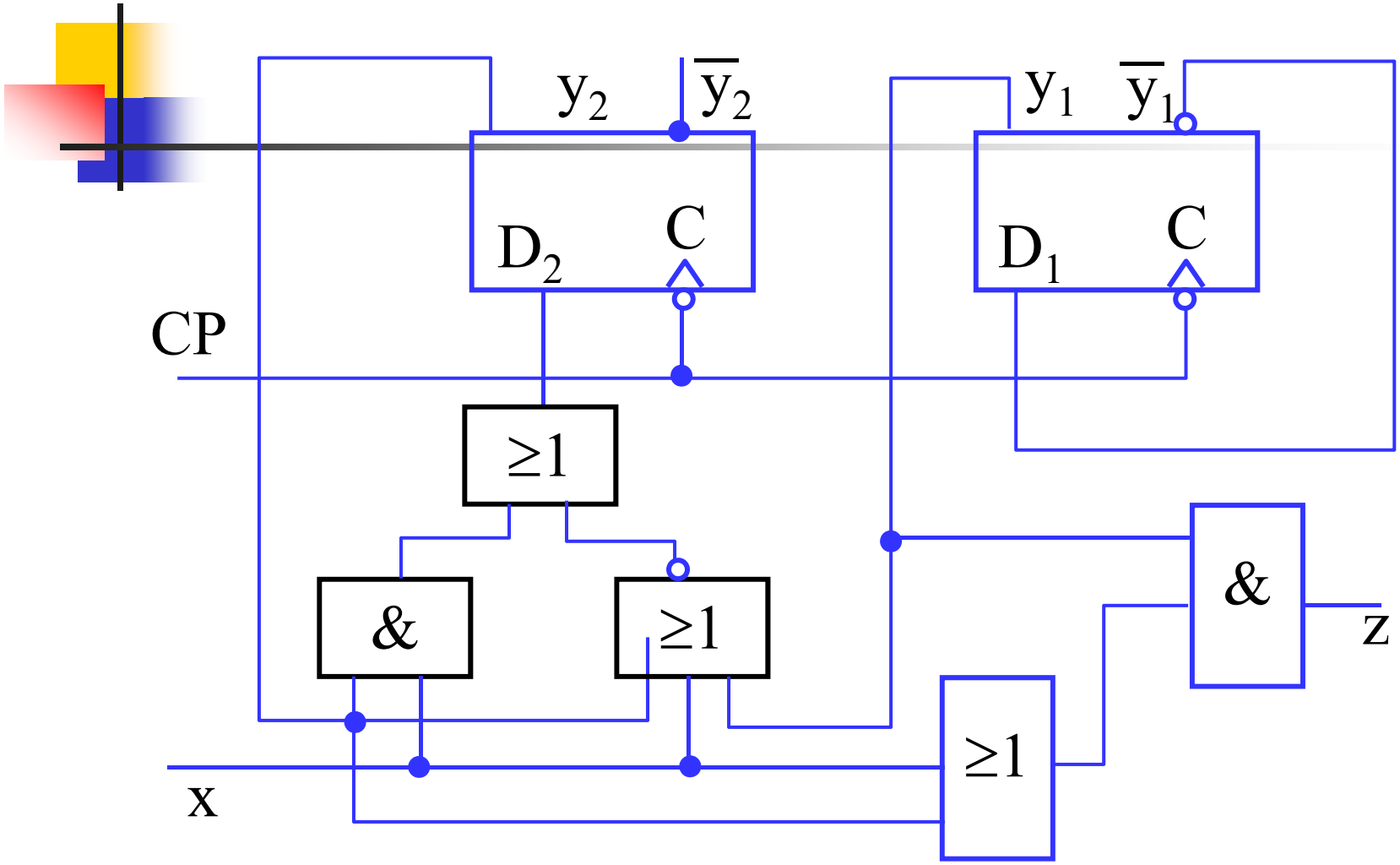


例如：

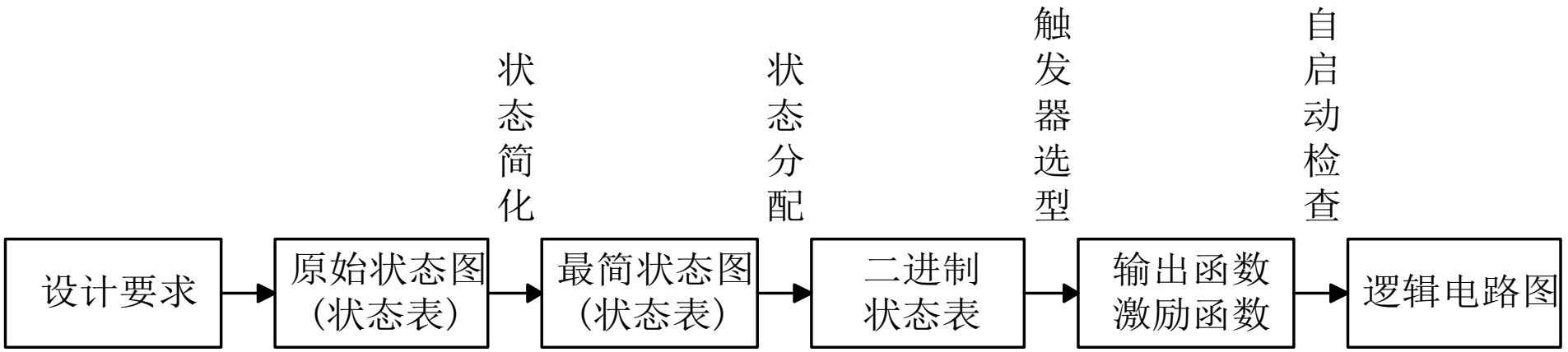
$$D_2 = \overline{x y_2} \bar{y}_1 + x \bar{y}_2 = \overline{x + y_2 + y_1} + x y_2$$

$$D_1 = \bar{y}_1$$

$$Z = x y_1 + y_2 y_1 = y_1 (x + y_2)$$


y2
$\overline{y}_2$
y1
$\overline{y}_1$
D2
C
D1
C
CP
≥1
&
≥1
&
z
≥1
x

同步时序电路设计过程


设计要求
原始状态图
(状态表)
状态简化
最简状态图
(状态表)
状态分配
二进制
状态表
触发器选型
输出函数
激励函数
自启动检查
逻辑电路图

4.6 同步时序逻辑电路设计举例

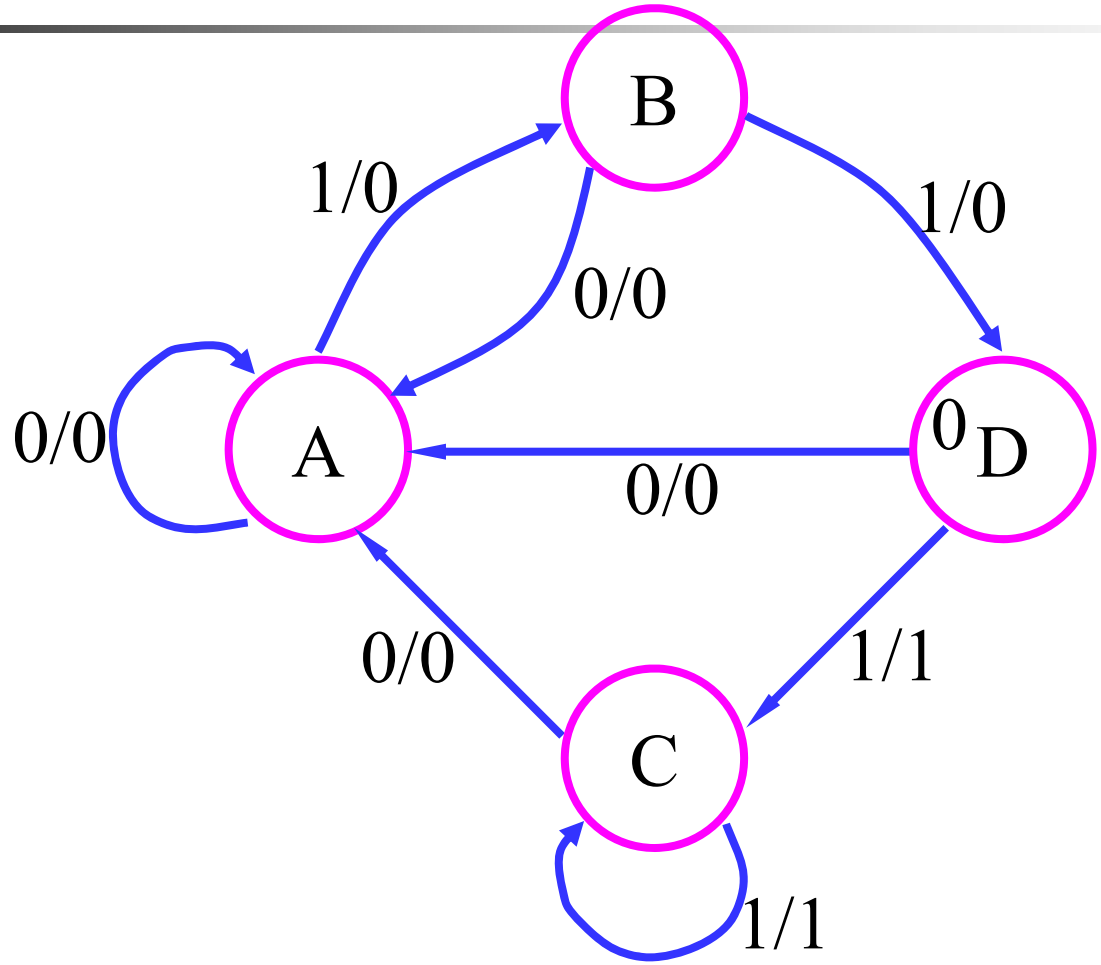
例：

设计一个“111…”序列检测器，用来检测串行二进制序列，要求每当连续输入3个(或3个以上)1时，检测器输出为1，否则输出为0。其典型输入输出序列如下：

输入x：0 1 1 1 0 1 1 1 1 0

输出Z：0 0 0 1 0 0 0 1 1 0

解：　　作状态图和状态表



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

- 状态化简

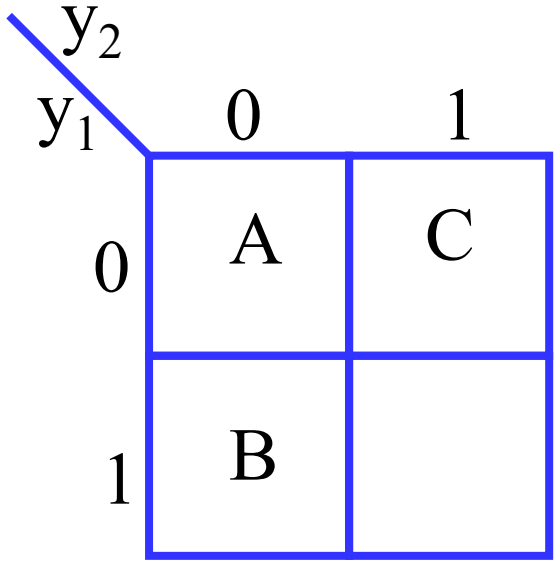
用观察法可得最大等效类为：(A), (B), (C,D)

•令C＝(C,D),可得下列最简状态表

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

状态分配

- AB，BC，AC应相邻
- AB，AC应相邻
- AB应相邻
- A应为逻辑0



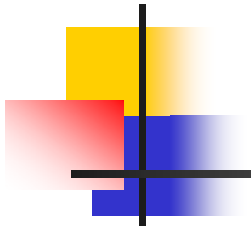
现　　态		次态$y_2^{(n+1)}y_1^{(n+1)}$/输出Z	
y_2	y_1	x=0	x=1
0	0	00/0	01/0
0	1	00/0	10/0
1	0	00/0	10/1

确定激励函数和输出函数表达式

选用J－K触发器作为存储元件

$Q->Q^{(n+1)}$		J	K
0	0	0	d
0	1	1	d
1	0	d	1
1	1	d	0

输 入	现 态		次 态		激励函数				输 出
x	y_2	y_1	$y_2^{(n+1)}$	$y_1^{(n+1)}$	J_2	K_2	J_1	K_1	Z
0	0	0	0	0	0	d	0	d	0
0	0	1	0	0	0	d	d	1	0
0	1	1	d	d	d	d	d	d	d
0	1	0	0	0	d	1	0	d	0
1	0	0	0	1	0	d	1	d	0
1	0	1	1	0	1	d	d	1	0
1	1	1	d	d	d	d	d	d	d
1	1	0	1	0	d	0	0	d	1



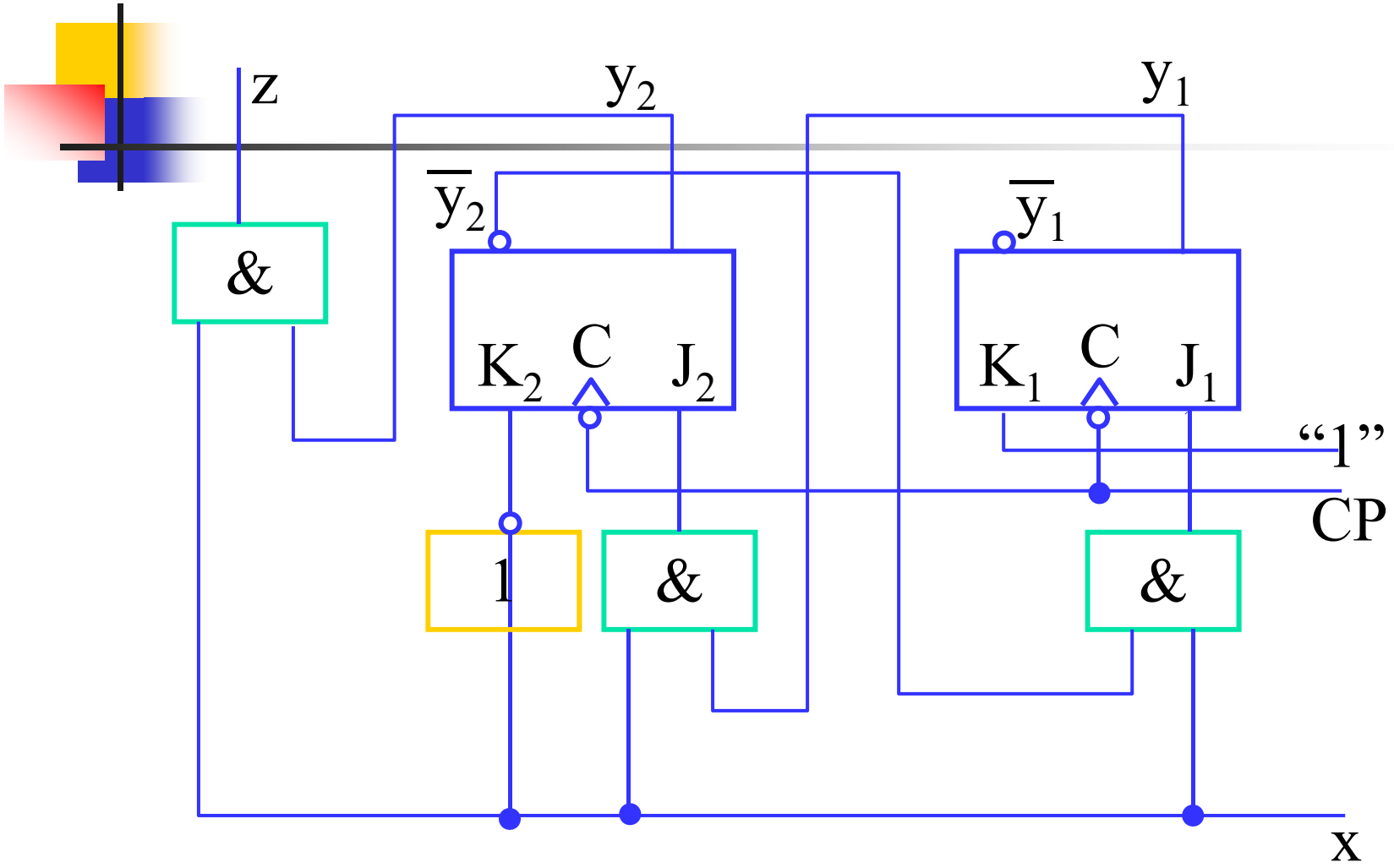
通过卡诺图化简，最后可得：

$J_2=xy_1$，　　$k_2=\bar{x}$

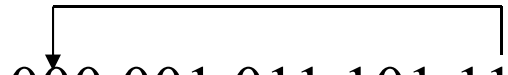
$J_1=\bar{y}_2x$，　　$k_1=1$

$Z=xy_2$

画电路图


z
y2
y1
&
y2
y1
K2
C
J2
K1
C
J1
“1”
CP
1
&
&
x

【例】 用JK触发器设计一个五进制同步计数器，要求状态转换关系为



解：本例属于给定状态时序电路设计问题。

① 列状态表。

根据题意，该时序电路有三个状态变量，设状态变量为Q_2、Q_1、Q_0，可作出二进制状态表表1所示，它是一个非完全描述时序电路的设计。

表 1状态表一

Q_2	Q_1	Q_0	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	×	×	×
0	1	1	1	0	1
1	0	0	×	×	×
1	0	1	1	1	0
1	1	0	0	0	0
1	1	1	×	×	×

② 确定激励函数和输出函数。

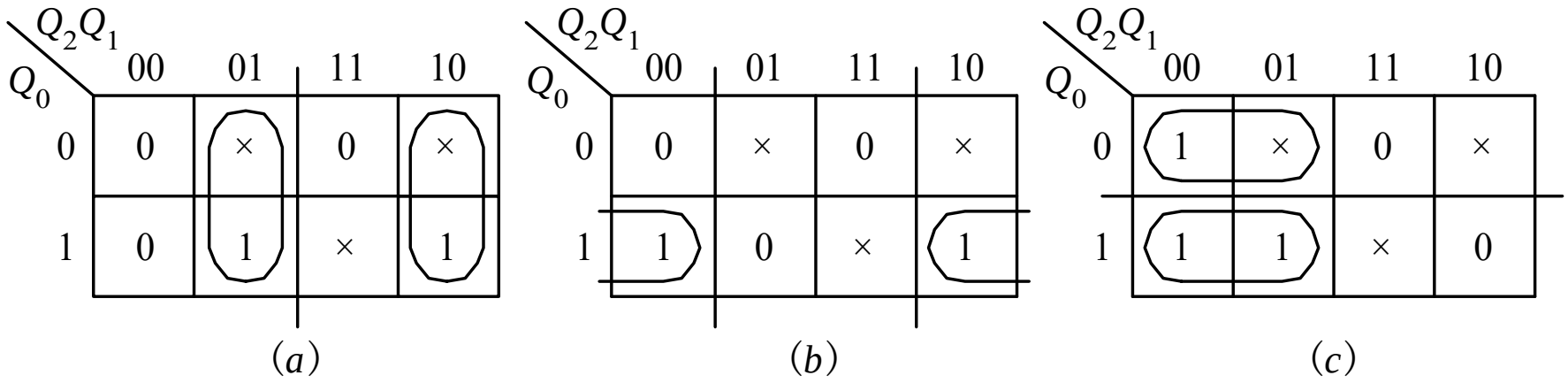


图1 次态卡诺图

$$(a)Q_2^{n+1};(b)Q_1^{n+1};(c)Q_0^{n+1}$$

由次态卡诺图求出其状态方程和激励函数如下：

$$Q_2^{n+1} = Q_1\overline{Q}_2 + \overline{Q}_1 Q_2, \quad J_2 = Q_1, \quad K_2 = Q_1$$

$$Q_1^{n+1} = Q_0\overline{Q}_1, \quad J_1 = Q_0, \quad K_1 = 1$$

$$Q_0^{n+1} = \overline{Q}_2\overline{Q}_0 + \overline{Q}_2 Q_0, \quad J_0 = \overline{Q}_2, \quad K_0 = Q_2$$

③ 自启动检查。

根据以上状态方程，检查多余状态的转移情况如表2所示，其完整的状态图如图2所示。

表 2 多余状态转移表

Q_2	Q_1	Q_0	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
0	1	0	1	0	1
1	0	0	1	0	0
1	1	1	0	0	0

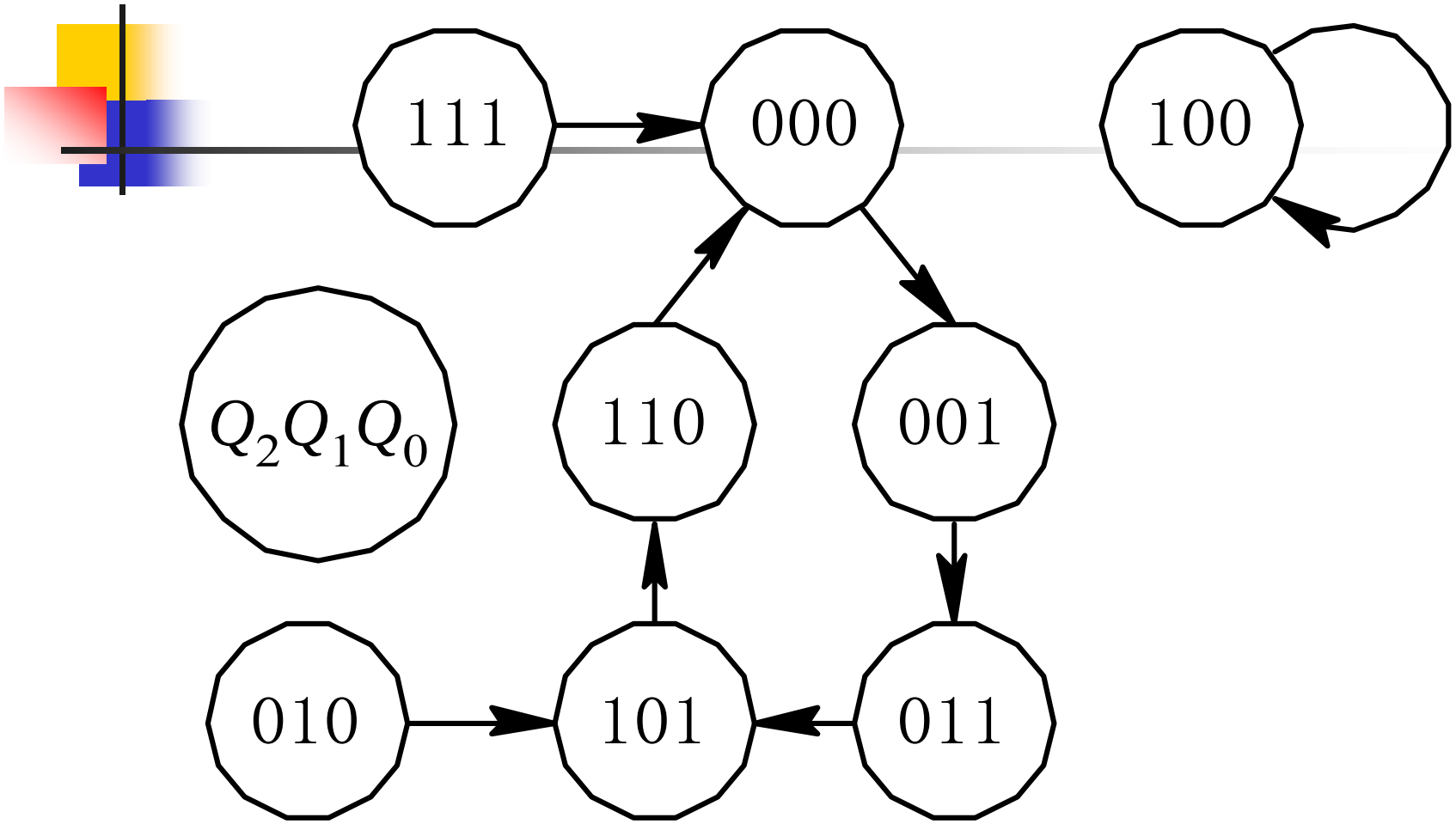


图 2 状态图

从图2可以看出，该电路一旦进入状态100，就不能进入计数主循环，因而该电路不能实现自启动，需要修改设计。

在非完全描述时序电路中，由于存在无效状态，使得在激励函数的获取过程中出现了任意项。在求取激励函数时，如果某任意项被圈入，则该任意项被确认为1，否则被确认为0。 由于圈法的随意性，故无效状态的转移可能出现死循环而使电路不能自启动。当电路不能自启动时，解决的方法有多种。

第一种方法，将原来的非完全描述时序电路中没有描述的状态的转移情况加以定义，使其成为完全描述时序电路。如将表1状态表中的无效状态的转移方向均定义为000，则可得到一个完全描述时序电路的状态表如表3。显然，按照表3设计的时序电路，不存在死循环问题，因为它是完全描述的。这种方法由于失去了任意项，会增加电路的复杂程度。

表 3　状态表二

Q_2	Q_1	Q_0	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	0	0	0
0	1	1	1	0	1
1	0	0	0	0	0
1	0	1	1	1	0
1	1	0	0	0	0
1	1	1	0	0	0

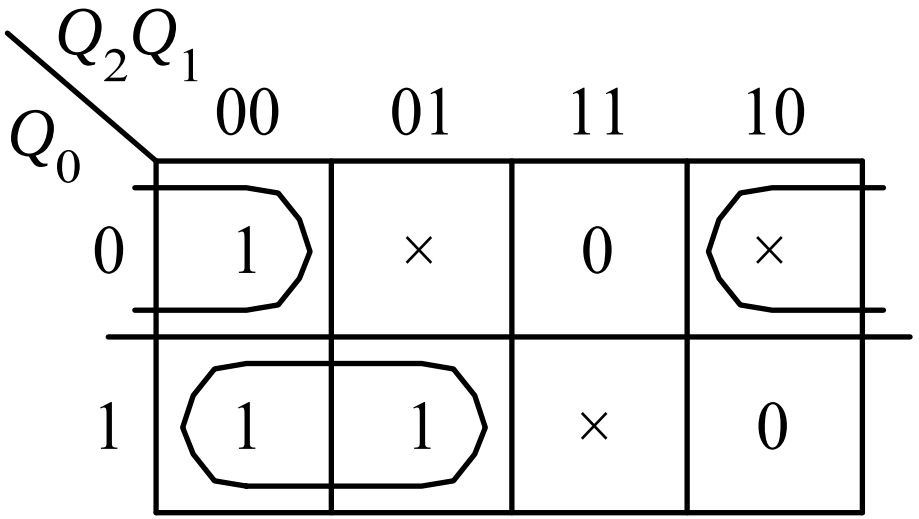
第二种方法，改变原来的圈法。如果盲目地改变所有激励函数的圈法，则其工作量大，效果差。若在分析观察的基础上改变某激励函数的圈法，则能获得较满意的效果。观察图1 次态卡诺图，如果希望能尽量使用任意项，只能对(*a*)和(*c*)的圈法作修改。现对(*c*)的圈法作修改，它仅改变Q_0的转移，新的圈法如图3所示。

分析新圈法可知：状态010将转移到100（原转移到101，现在最后一位Q_0转为0），状态100将转移到101（原转移到100，现最后一位Q_0转为1）。 由分析可以看出，新圈法将克服死循环，也不增加激励函数的复杂程度。

由新圈法得

$$Q_0^{n+1} = \overline{Q}_1\overline{Q}_0 + \overline{Q}_2 Q_0$$

$$J_0 = \overline{Q}_1, K_0 = Q_2$$



（c） Q_0^{n+1}

图 3 修整后圈法

重新检查多余状态的转移情况如表4所示，其状态图如图4所示，可以看到该电路具有自启动能力。如果修改图1 (*a*)的圈法，可以得到同样的效果。

表 4 多余状态转移表

Q_2	Q_1	Q_0	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
0	1	0	1	0	0
1	0	0	1	0	1
1	1	1	0	0	0

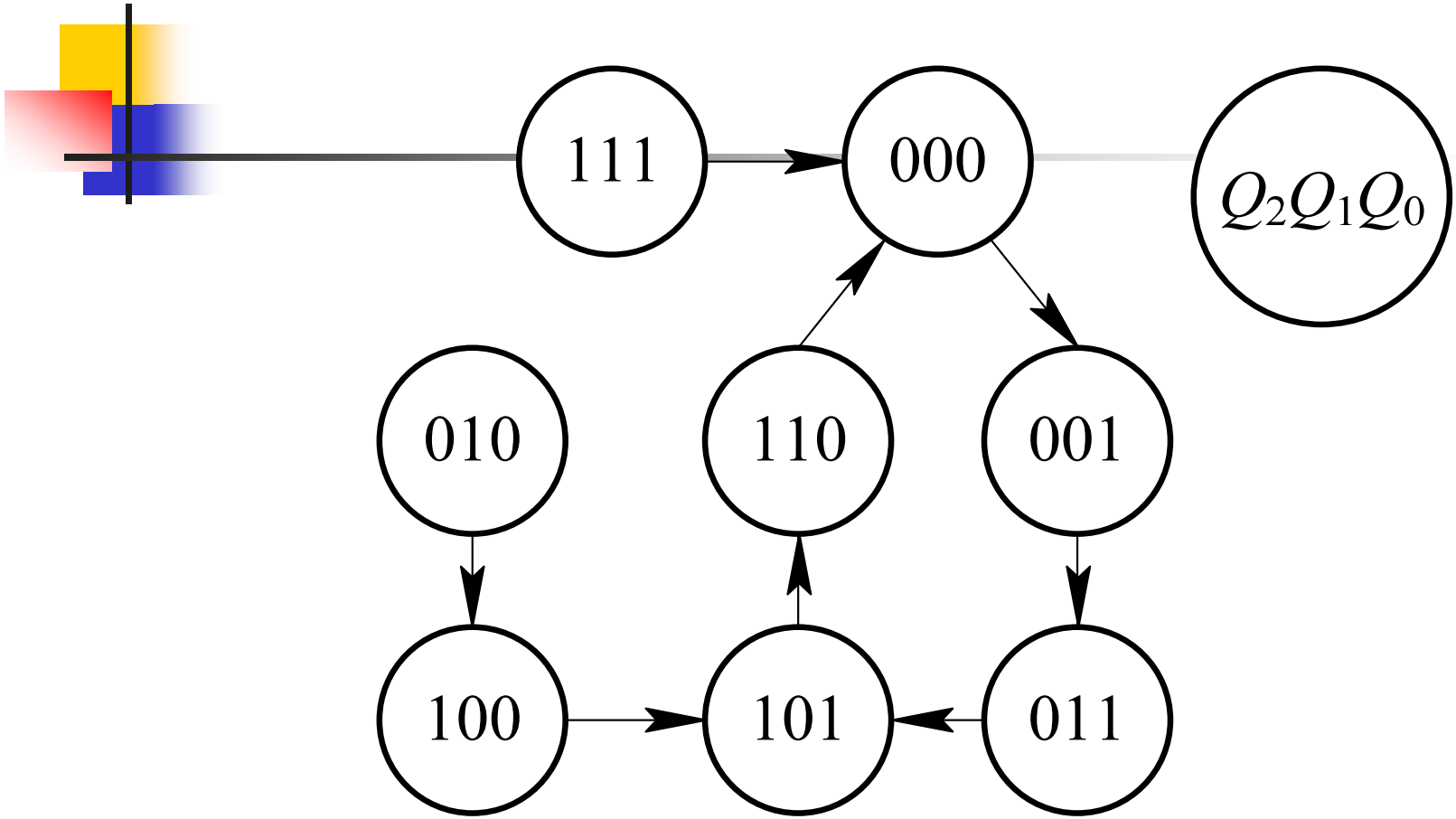


图 4

④ 画逻辑图。

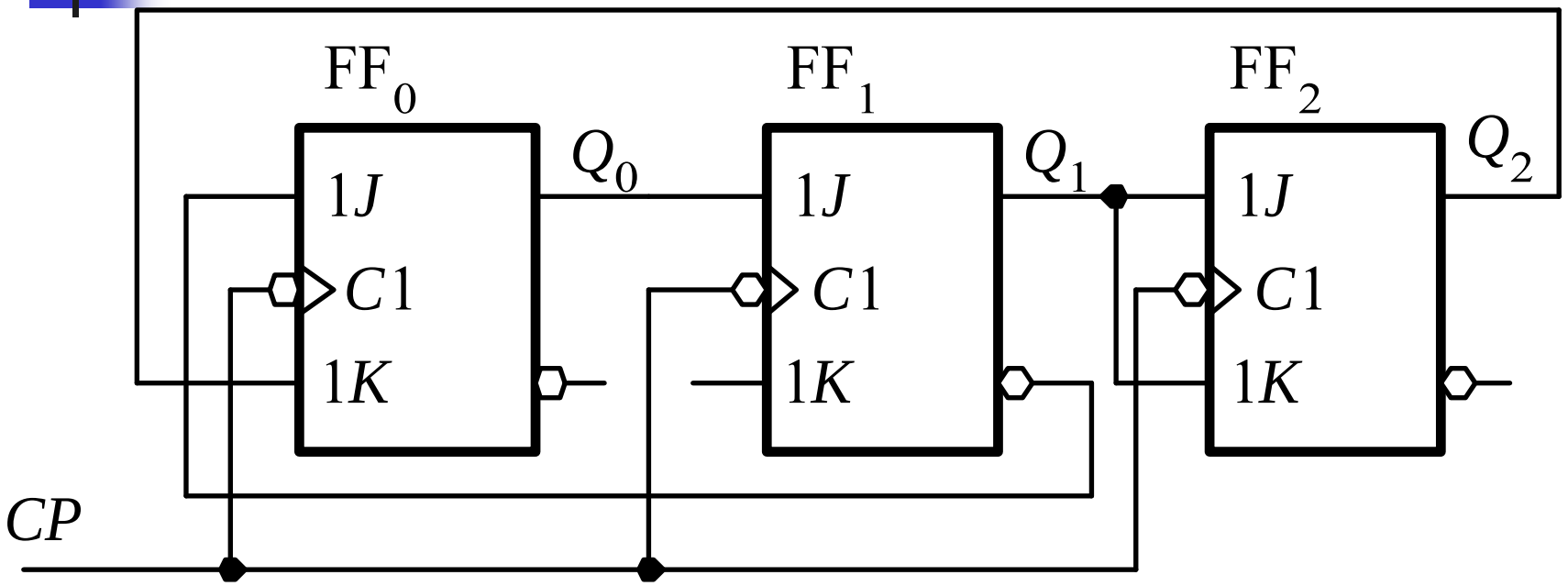


图 5 逻辑图

【 例 】用D触发器设计一个模七同步加法计数器。

解： 本例属于给定状态时序电路设计问题。

① 列状态表。

根据题意，该时序电路有三个状态变量。设状态变量为Q_2、 Q_1、Q_0，可作出二进制状态表如表6-28所示，它是一个非完全描述时序电路的设计。

② 确定激励函数和输出函数。

由表1状态表分别画出Q_2、Q_1、Q_0的次态卡诺图如图1(*a*)、(*b*)、 (*c*)所示。

表 1　状态表

Q_2	Q_1	Q_0	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	0	0	0
1	1	1	×	×	×

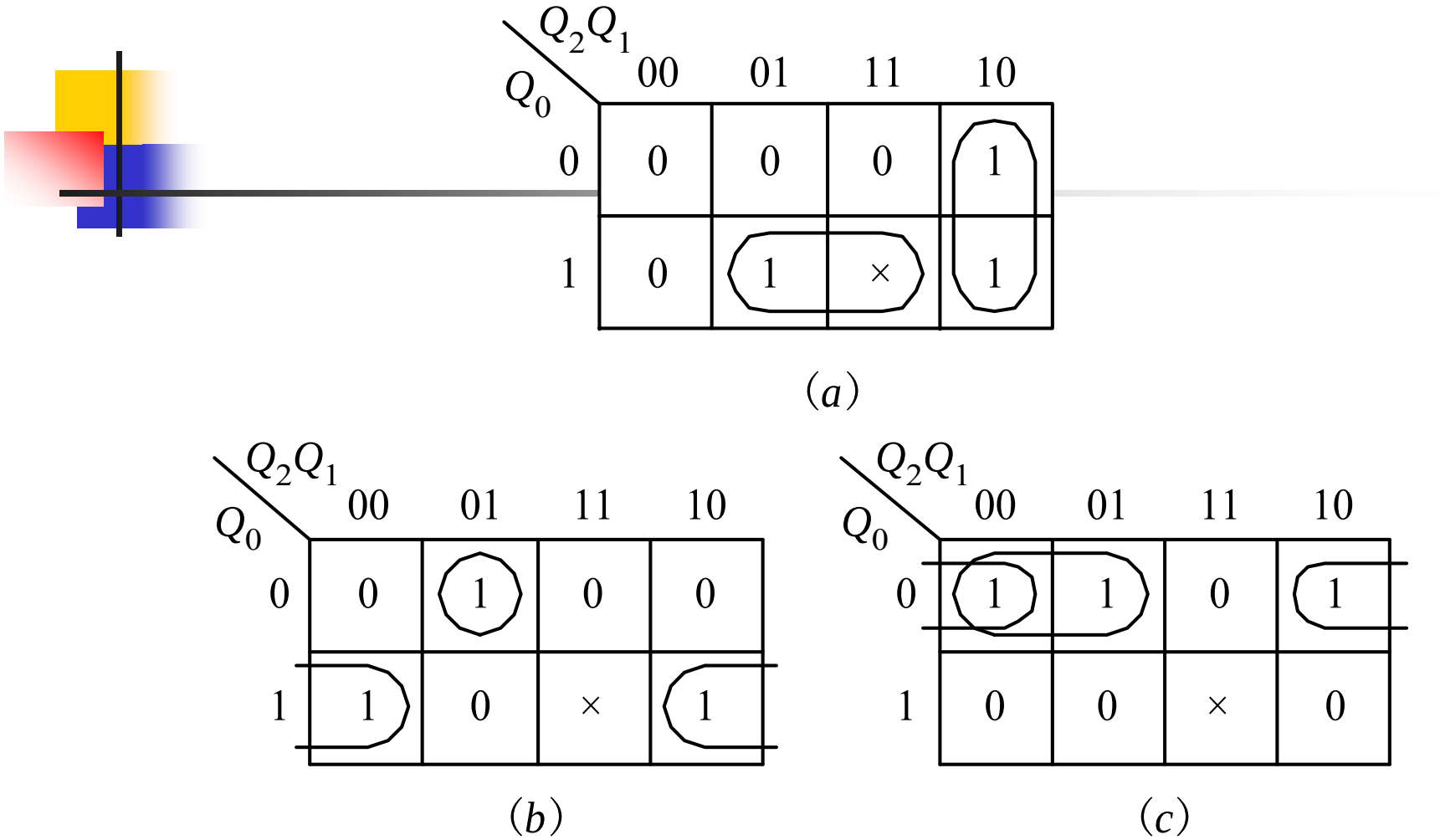


图 1 次态卡诺图

$(a)Q_2^{n+1};(b)Q_1^{n+1};(c)Q_0^{n+1}$

当使用D触发器实现时序电路时，由于D触发器的特征方程为Q^{n+1}=D，因此，可从次态卡诺图直接求出D触发器的激励函数：

$$D_2 = Q_1 Q_0 + Q_2 \overline{Q}_1$$

$$D_1 = \overline{Q}_2 Q_1 \overline{Q}_0 + \overline{Q}_1 Q_0$$

$$D_0 = \overline{Q}_2 \overline{Q}_0 + \overline{Q}_1 \overline{Q}_0$$

③ 自启动检查。

观察次态卡诺图激励函数的圈法，多余状态111的新状态为100，电路的状态图如图2所示，该电路具有自启动能力。

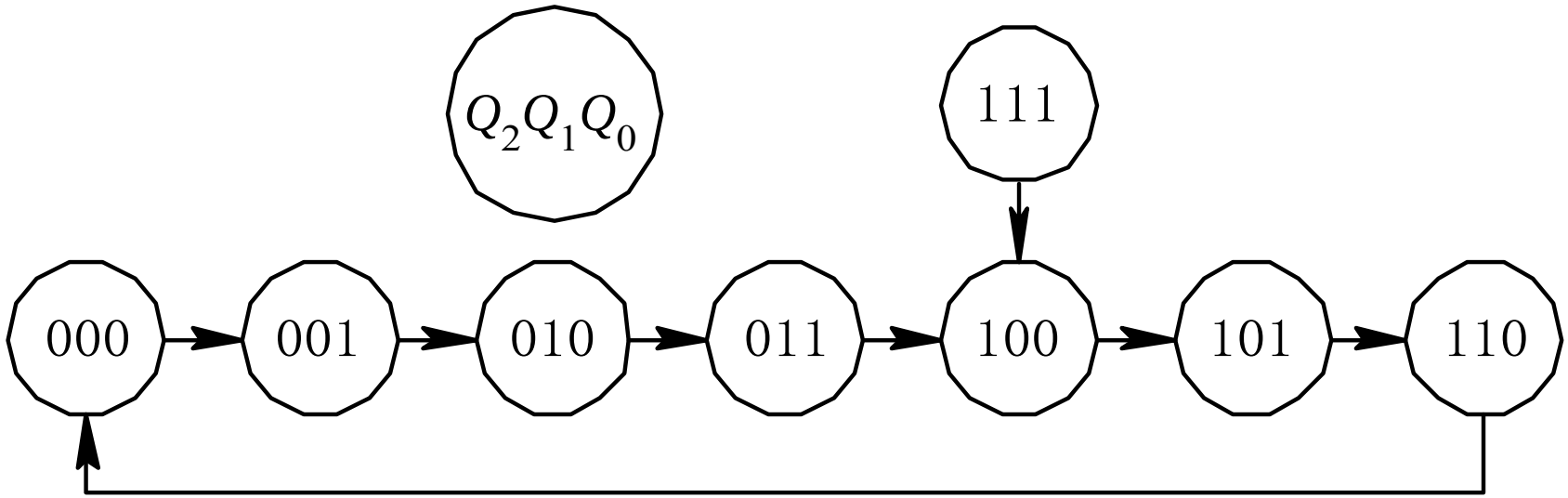


图 2 状态图

④ 画逻辑图。

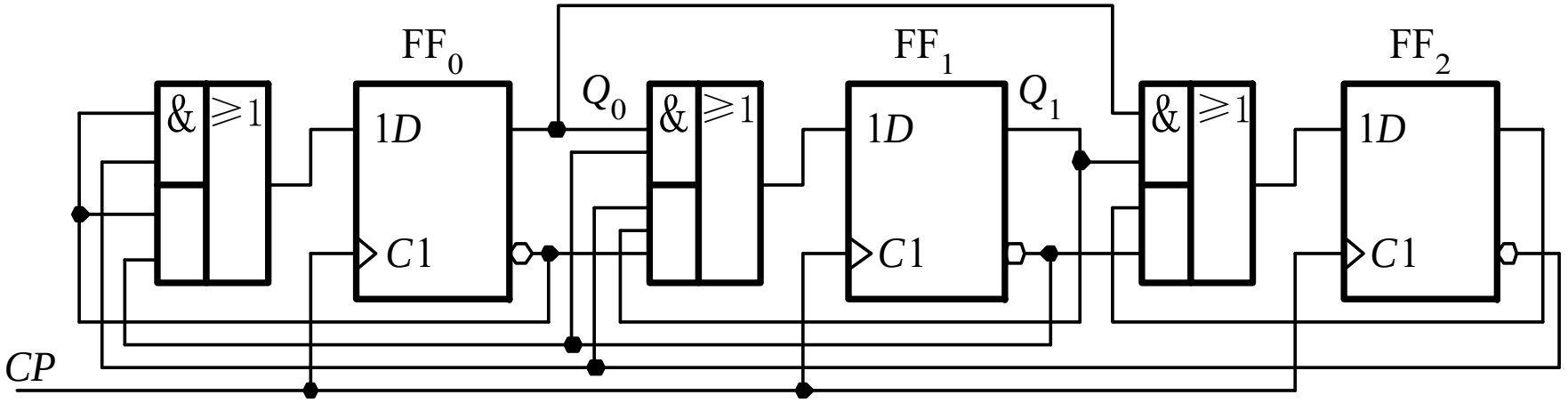


图 3 逻辑图

例 5 设计一个串行数据检测器，该电路具有一个输入端x和一个输出端z。输入为一连串随机信号，当出现“1111″序列时，检测器输出信号z=1，对其它任何输入序列，输出皆为 0。

解 (1) 建立原始状态图。

① 起始状态S0，表示没接收到待检测的序列信号。当输入信号x=0 时，次态仍为S_0，输出z为 0；如输入 x=1，表示已接收到第一个“1″，其次态应为 S_1，输出为0。

② 状态为S_1，当输入x=0 时，返回状态S_0，输出为0； 当输入x=1 时，表示已接收到第二个“1”，其次态应为S_2， 输出为 0。

③ 状态为S_2，当输入x=0 时，返回状态S_0，输出为0； 当输入x=1 时，表示已连续接收到第三个“1”，其次态应为S_3，输出为 0。

④ 状态为S_3，当输入x=0 时，返回状态S_0，输出为0；当输入x=1 时，表示已连续接收到第四个“1”，其次态为 S_4，输出为“1”。

⑤ 状态为S_4，当输入x=0时，返回状态S_0，输出为0；当输入x=1 时，则上述过程的后三个“1”与本次的“1”， 仍为连续的四个“1”， 故次态仍为S_4，输出为“1”。

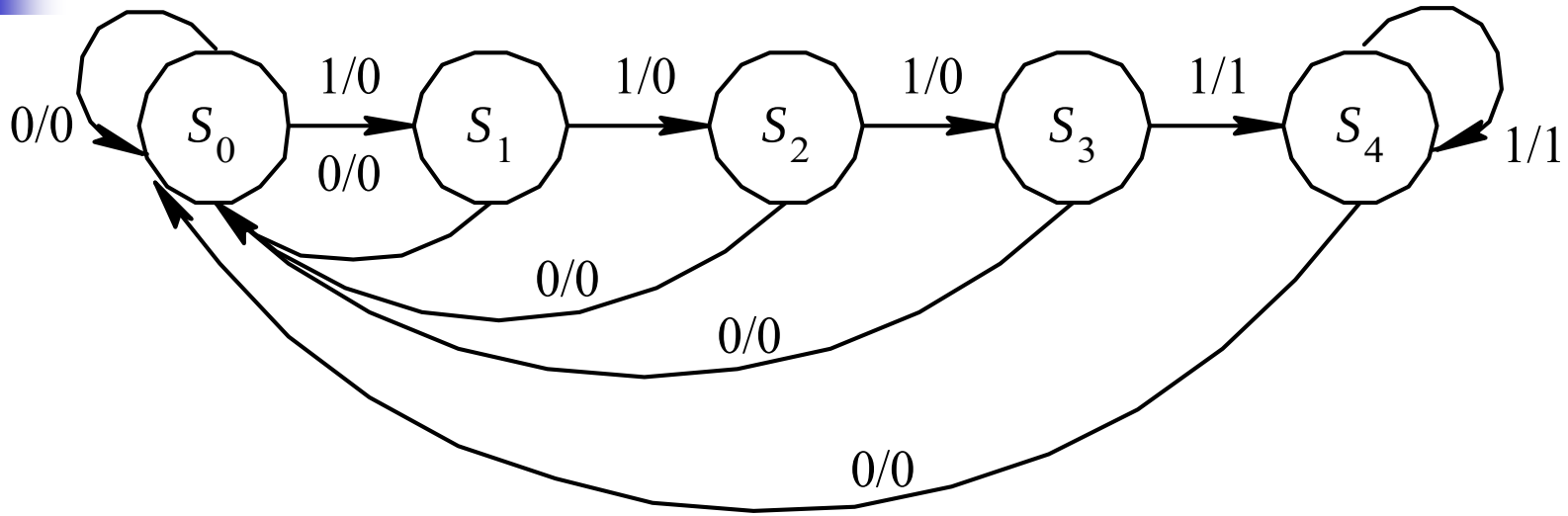


图 6 – 12　例 5 原始状态图

表 6－5 例 5 状态表

现态 \ 输入	次态/输出	
	0	1
S_0	$S_0/0$	$S_1/0$
S_1	$S_0/0$	$S_2/0$
S_2	$S_0/0$	$S_3/0$
S_3	$S_0/0$	$S_4/1$
S_4	$S_0/0$	$S_4/1$

(2) 状态化简。

在做原始状态图时，为确保功能的正确性，遵循“宁多勿漏”的原则。因此，所得的原始状态图或状态表可能包含有多余的状态，使状态数增加，将导致下列结果：

① 系统所需触发器级数增多；

② 触发器的激励电路变得复杂；

③ 故障增多。

因此， 状态化简后减少了状态数对降低系统成本和电路的复杂性及提高可靠性均有好处。

(3) 状态分配。

状态分配是指将化简后的状态表中的各个状态用二进制代码来表示，因此，状态分配有时又称为状态编码。电路的状态通常是用触发器的状态来表示的。

由于$2^2=4$，故该电路应选用两级触发器Q^2和Q^1，它有 4 种状态：“00″、“01″、“10″、“11″，因此对S_0、S_1、S_2、S_3 的状态分配方式有多种。对该例状态分配如下：

S_0—00　　S_1—10

S_2——01　　S_3 —11

则状态分配后的状态表如表 6 - 6 所示。

表 6 – 6　例 5 状态分配后的状态表

$Q_2^nQ_1^n$ \ x	$Q_2^{n+1}Q_1^{n+1}/z$	
	0	1
0 0	0 0/0	1 0/0
0 1	0 0/0	1 1/0
1 0	0 0/0	0 1/0
1 1	0 0/0	1 1/1

(4) 确定激励方程和输出方程。

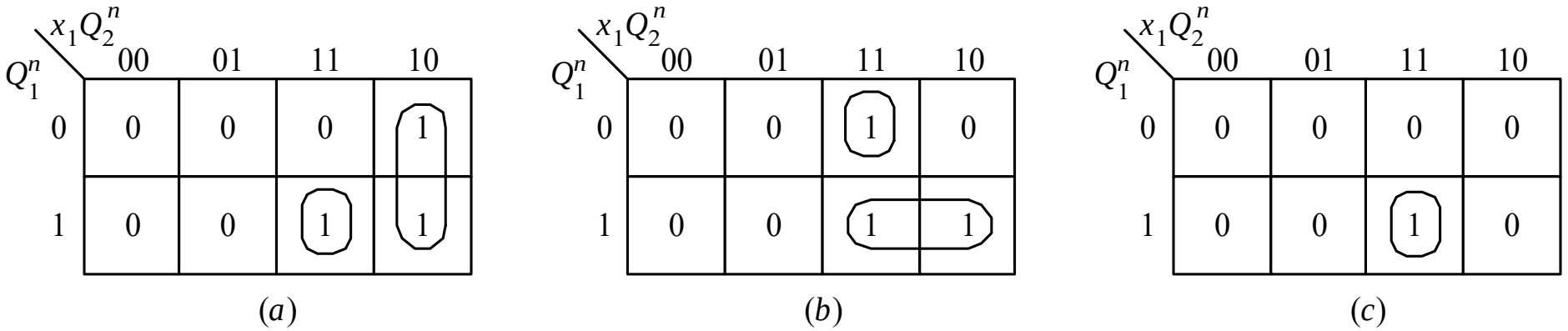


图 6 – 13 例 5 激励方程、输出方程的确定

在求每一级触发器的次态方程时，应与标准的特征方程一致， 这样才能获得最佳激励函数。如JK触发器标准特征方程为

$$Q^{n+1} = J\overline{Q^n} + \overline{K}Q^n$$

则求 Q_2^{n+1} 时应得

$$Q_2^{n+1} = a\overline{Q_2^n} + \beta Q_2^n$$

两式相比得

$$J = \alpha, \qquad K = \overline{\beta}$$

故

$$Q_2^{n+1} = x\overline{Q_2^n} + xQ_1^nQ_2^n$$

$$J_2 = x \qquad K_2 = \overline{xQ_1^n}$$

$$Q_1^{n+1} = xQ_2^n\,\overline{Q_1^n} + xQ_1^n$$

$$J_1 = xQ_2^n \qquad K_1 = \overline{x}$$

输出方程由卡诺图得

$$z = xQ_2^nQ_1^n$$

(5) 画出逻辑图。

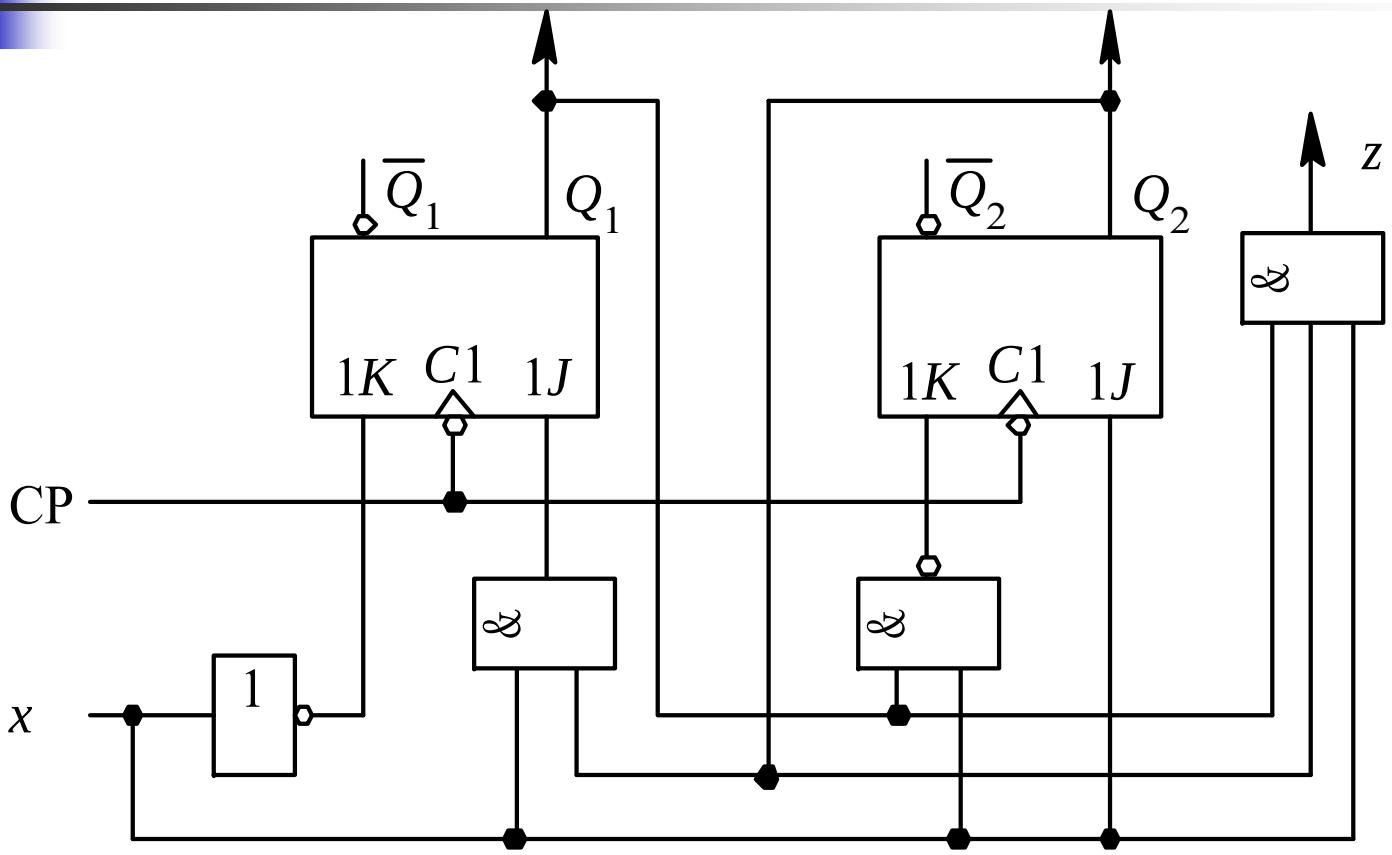


图 6 – 14 例 5 逻辑图

例 6 用JK触发器设计一个 8421BCD码加法计数器。

解 该题的题意中即明确有10个状态，且是按8421BCD 加法规律进行状态迁移，因为 $2^3<10<2_4$，所以需要四级触发器，其状态迁移表如表 6 - 7 所示，由状态表做出每一级触发器的卡诺图。

表 6 – 7　例 6 状态迁移表

Q_4^n	Q_3^n	Q_2^n	Q_1^n	Q_4^{n+1}	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}
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	0	0	0	0
1	0	1	0	×	×	×	×
1	0	1	1	×	×	×	×
1	1	0	0	×	×	×	×
1	1	0	1	×	×	×	×
1	1	1	0	×	×	×	×
1	1	1	1	×	×	×	×

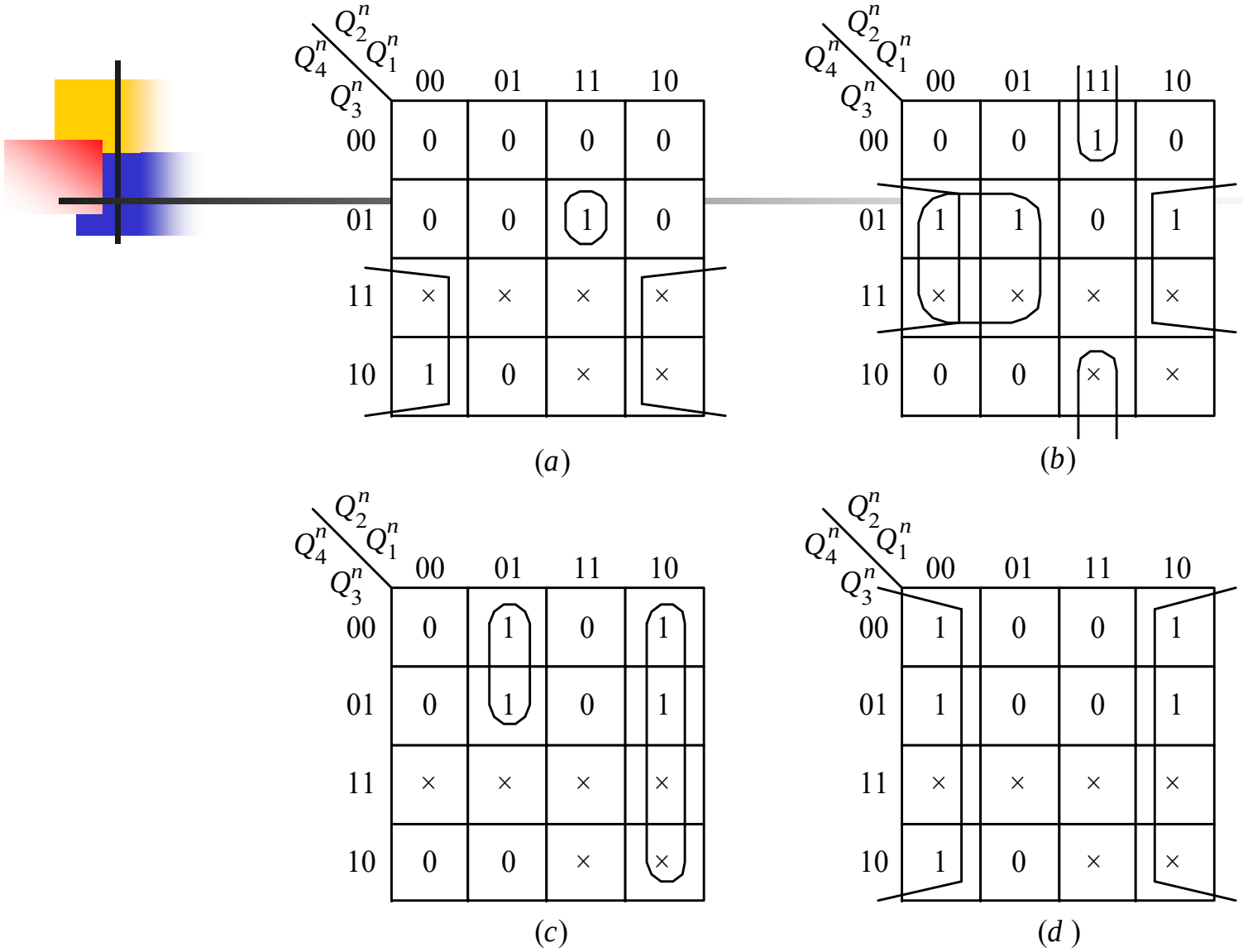


图 6 –15 确定激励函数的次态卡诺图

由图 6 - 15(*a*)~(*d*)可得

$$Q_4^{n+1} = Q_1^n Q_2^n Q_3^n \overline{Q_4^n} + \overline{Q_1^n}\, Q_4^n$$

$$Q_3^{n+1} = Q_1^n Q_2^n \overline{Q_3^n} + \overline{Q_1^n}\, Q_3^n + \overline{Q_2^n}\, Q_3^n$$

$$= Q_1^n Q_2^n \overline{Q_3^n} + \overline{Q_1^n Q_2^n}\, Q_3^n$$

$$Q_2^{n+1} = Q_1^n\, \overline{Q_4^n}\, \overline{Q_2^n} + \overline{Q_1^n}\, Q_2^n$$

$$Q_1^{n+1} = \overline{Q_1^n}$$

由此得各触发器的激励函数为

$$J_4 = Q_1^n Q_2^n Q_3^n \qquad K_4 = Q_1^n$$

$$J_3 = Q_1^n Q_2^n \qquad K_3 = Q_1^n Q_2^n$$

$$J_2 = Q_1^n \overline{Q_4^n} \qquad K_2 = Q_1^n$$

$$J_1 = K_1 = 1$$

由激励方程得逻辑图, 如图 6 - 16 所示。

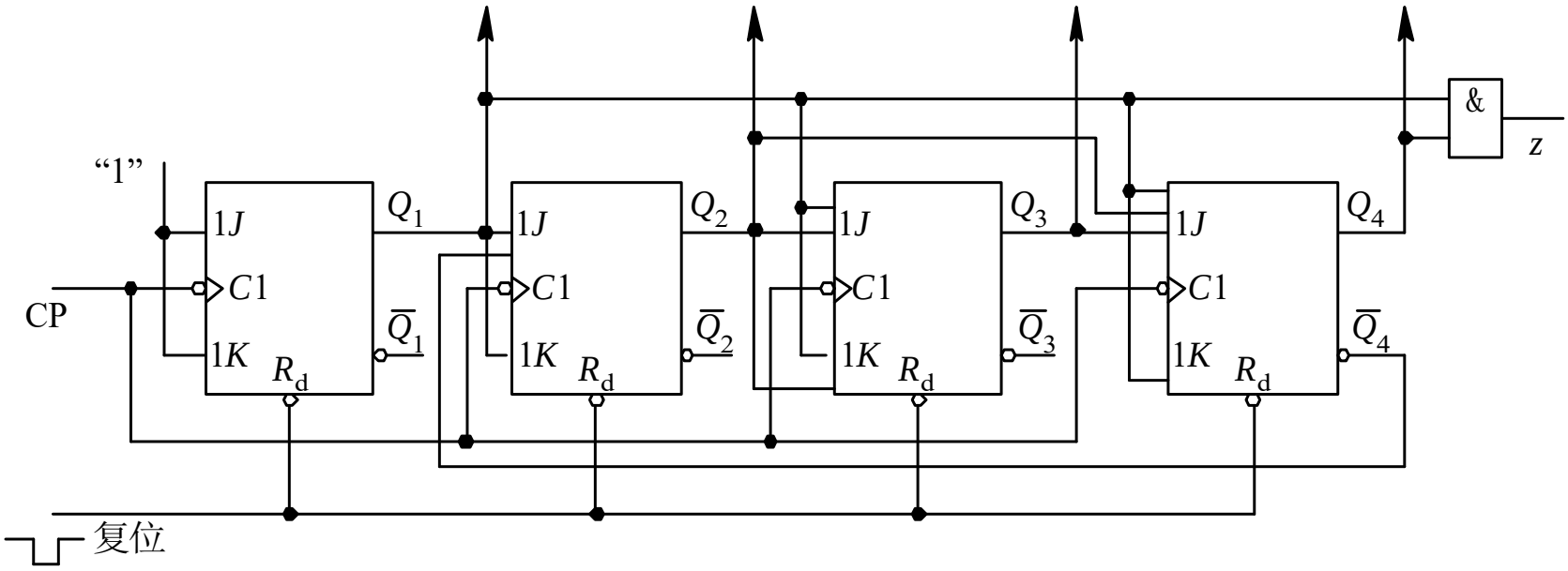


图 6 – 16 8421BCD码加法计数器逻辑图

表 6－8 检查自启动问题

Q_4^n	Q_3^n	Q_2^n	Q_1^n	Q_4^{n+1}	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}
1	0	1	0	1	0	1	1
1	0	1	1	0	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	0	1	0	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0

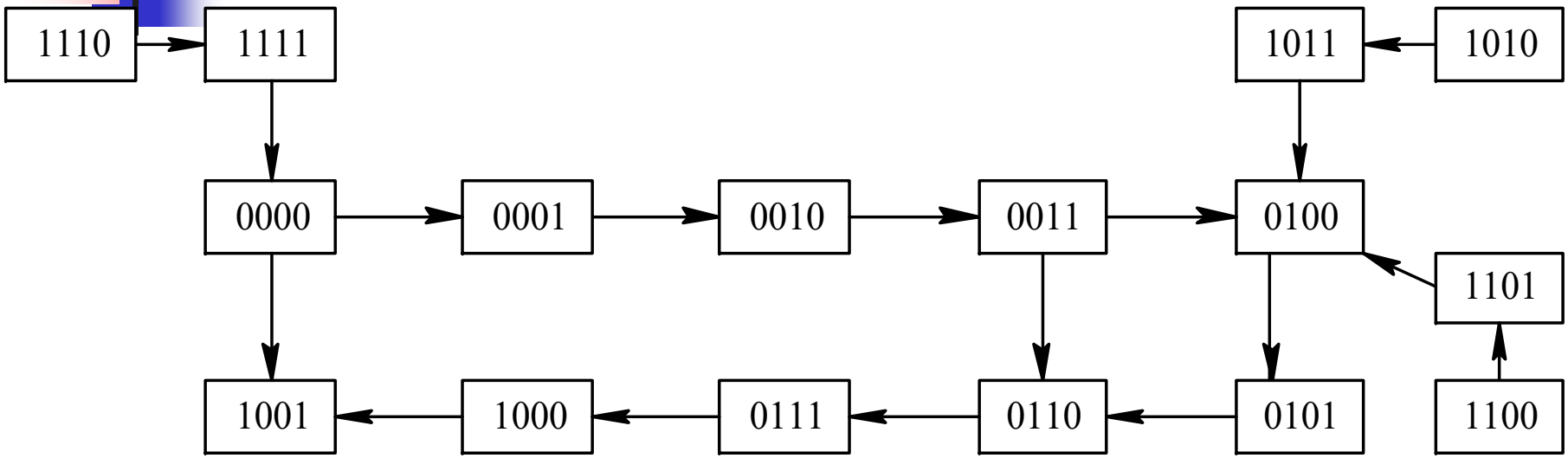


图 6 – 17 检查自启动能力

例 7 用JK触发器设计模 6 计数器。

由于$2^2<6<2^3$，所以模6计数器应该由三级触发器组成。三级触发器有 8 种状态，从中选 6 种状态，方案很多。我们按图 6 - 18 选取，其状态表如表 6 -9 所示。进位关系也在图中表示出来了。

表 6 –9 状态表

Q_3^n	Q_2^n	Q_1^n	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	C
0	0	0	1	0	0	0
1	0	0	1	1	0	0
1	1	0	1	1	1	0
1	1	1	0	1	1	0
0	1	1	0	0	1	0
0	0	1	0	0	0	1

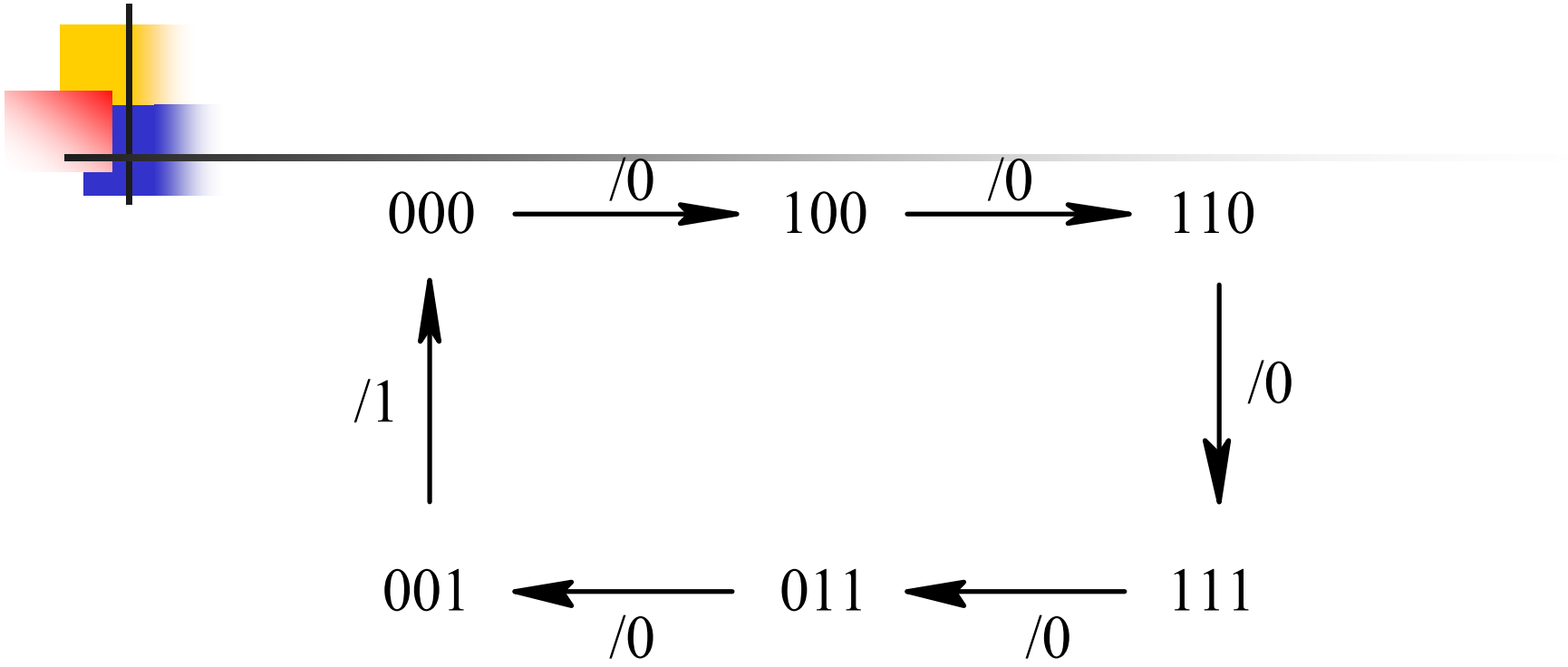


图 6 – 18　模 6 计数器状态迁移图

$$Q_3^{n+1} = \overline{Q_1^n Q_3^n} + \overline{Q_1^n}\, Q_3^n$$

$$Q_2^{n+1} = Q_3^n\, \overline{Q_2^{n+}} + Q_3^n Q_2^n$$

$$Q_1^{n+1} = Q_1^n\, \overline{Q_1^n} + Q_2^n\, Q_1^n$$

$$C = \overline{Q_2^n}\, Q_1^n$$

激励方程为 $J_3 = \overline{Q_1^n}, \qquad K_3 = Q_1^n$

$$J_2 = Q_3^n, \qquad K_2 = \overline{Q_3^n}$$

$$J_1 = Q_2^n, \qquad K_1 = \overline{Q_2^n}$$

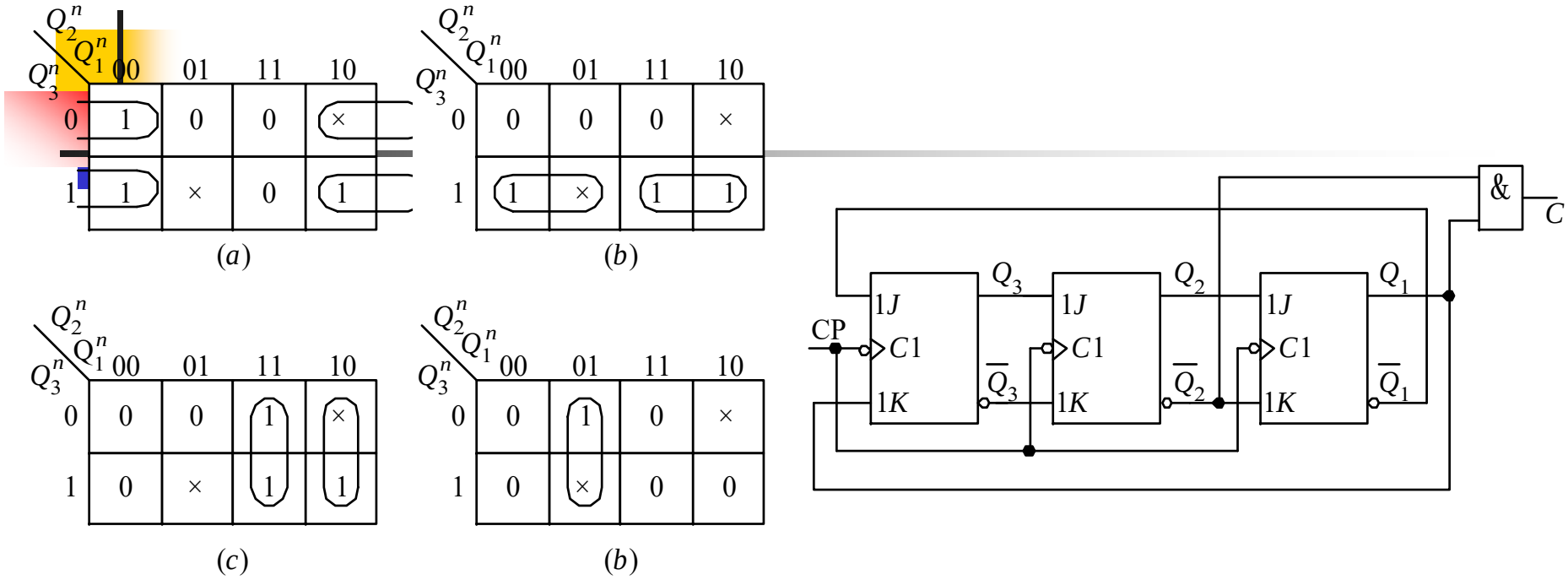


图6-19 模6计数器激励函数的确定和逻辑图

$$(a)Q_3^{n+1};(b)Q_2^{n+1};(c)Q_1^{n+1};(d)C$$

$$J_1 = Q_2^n, K_1 = \overline{Q_2^n}$$

检查自启动能力，把未用状态(010，101)代入上述次态方程，得到它们的状态变化情况，如表 6 - 10 和图 6-20 所示。

表 **6 – 10** 未用状态迁移关系

Q_3^n	Q_2^n	Q_1^n	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	C
0	1	0	1	0	1	0
1	0	1	0	1	0	0

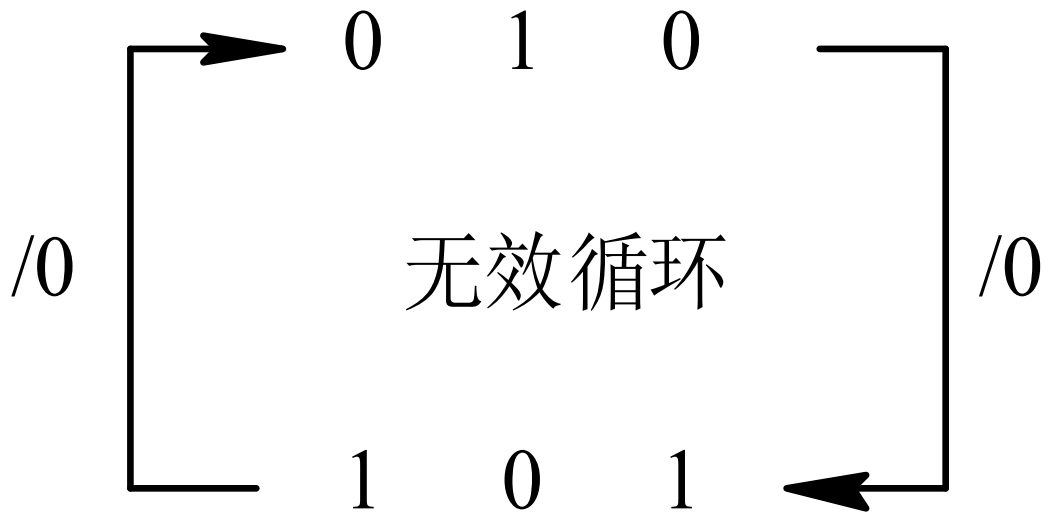


图 6 – 20　例 7 自启动能力检查

为了使电路具有自启动能力，可以修改状态转换关系，即切断无效循环，引入有效的计数循环序列。我们切断101→010 的转换关系，强迫它进入110。根据新的状态转换关系，重新设计。由于 Q_2^{n+1}和 Q_2^{n+1} 的转换关系没变， 只有改变了，故只要重新设计Q_3 级即可，如图 6 - 21(a)所示。

$$Q_3^{n+1} = \overline{Q_2^n}\, Q_3^n + \overline{Q_1^n}\, Q_3^n + \overline{Q_1^n}\, \overline{Q_3^n}$$

$$= \overline{Q_1^n}\, \overline{Q_3^n} + \overline{Q_1^n Q_2^n}\, Q_3^n$$

$$J_3 = \overline{Q_1^n}, \qquad K_3 = Q_1^n Q_2^n$$

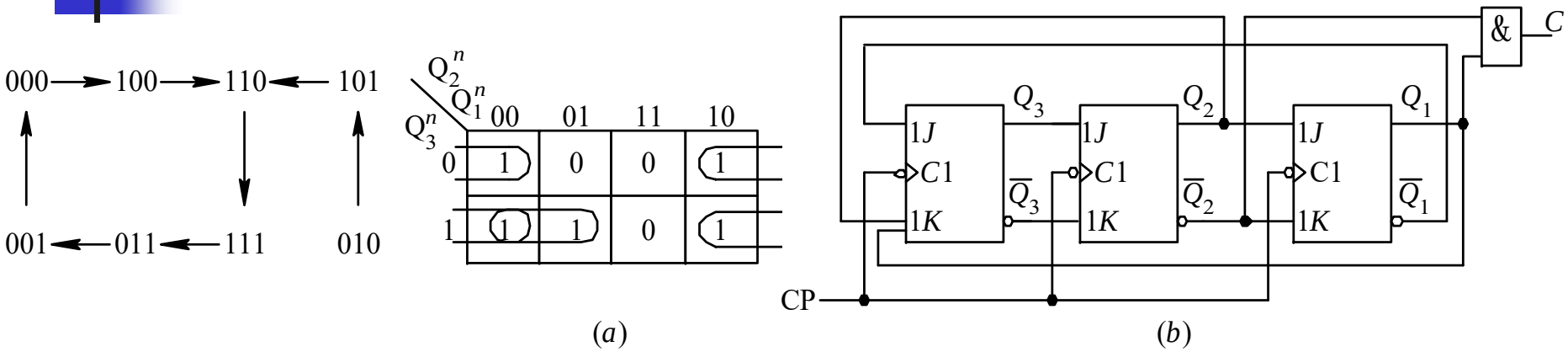


图 6 – 21　具有自启动能力的模 6 计数器