



## 第六章

# 异步时序逻辑电路





异步时序逻辑电路中没有统一的时钟脉冲信号，电路状态的改变是外部输入信号变化直接作用的结果。

根据电路结构和输入信号形式的不同，异步时序逻辑电路可分为**脉冲异步时序逻辑电路**和**电平异步时序逻辑电路**两种类型。

两类电路均有Mealy型和Moore型两种结构模型。



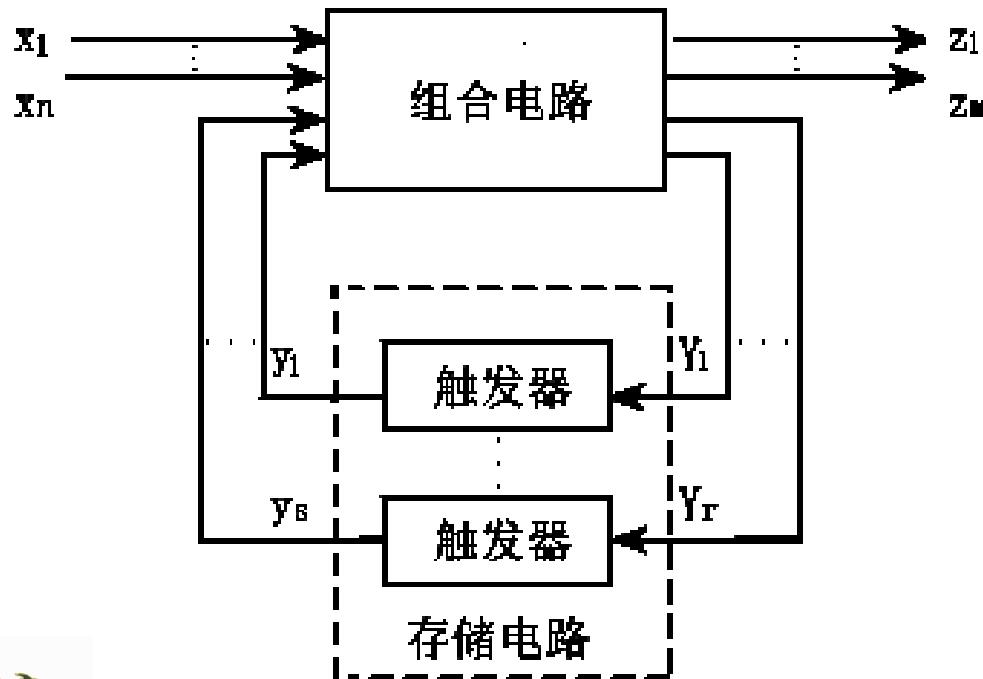


# 6.1 脉冲异步时序逻辑电路

## 6.1.1 概述

### 一、结构

脉冲异步时序电路的一般结构如下图所示。



图中，存储电路可由时钟控制触发器或非时钟控制触发器组成。





## 二、输入信号的形式与约束

1. 输入信号为脉冲信号；
2. 输入脉冲的宽度必须保证触发器可靠翻转；
3. 输入脉冲的间隔必须保证前一个脉冲引起的电路响应完全结束后，后一个脉冲才能到来；
4. 不允许两个或两个以上输入端同时出现脉冲。

对n个输入端的电路，其一位输入只允许出现 $n+1$ 种取值组合，其中有效输入种取值组合为n种。

## 三、输出信号的形式

脉冲异步时序逻辑电路的输出信号可以是脉冲信号也可以是电平信号。





## 6.1.2 脉冲异步时序逻辑电路的分析

### 一、分析方法与步骤

#### 1. 分析方法

脉冲异步时序逻辑电路的分析方法与同步时序逻辑电路大致相同。

**注意两点：**

(1)当存储元件采用时钟控制触发器时，对触发器的时钟控制端应作为激励函数处理。

仅当时钟端有脉冲作用时，才根据触发器的输入确定状态转移方向，否则，触发器状态不变。

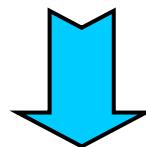
(2)根据对输入的约束，分析时可以排除两个或两个以上输入端同时出现脉冲以及输入端无脉冲出现情况，从而使图、表简化。



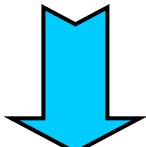


## 2. 分析步骤

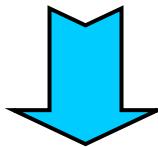
(1) 写出电路的输出函数和激励函数表达式；



(2) 列出电路次态真值表或次态方程组；



(3) 作出状态表和状态图；



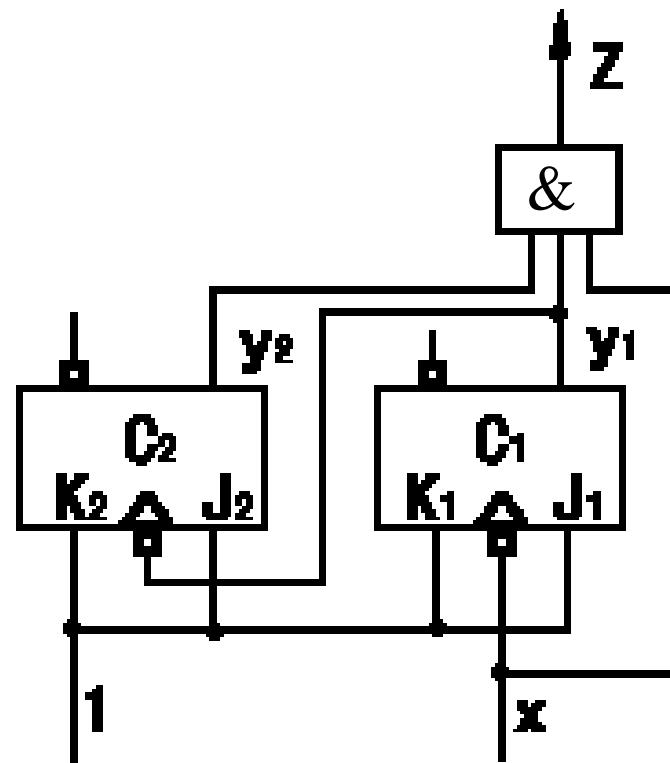
(4) 用文字描述电路的逻辑功能（必要时画出时间图）。

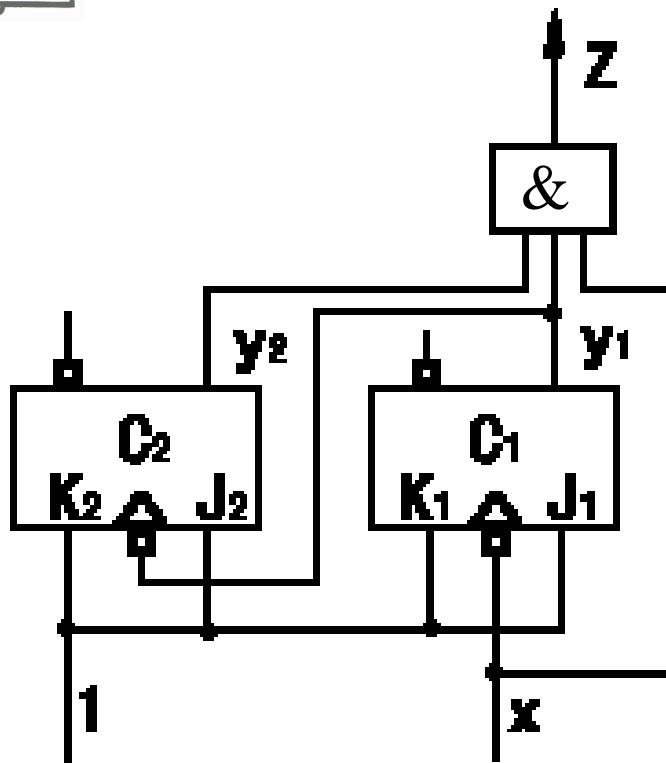




## 二、分析举例

例 分析下图所示脉冲异步时序逻辑电路，指出该电路功能。





**解:**该电路由两个J-K触发器和一个与门组成，有一个输入端x和一个输出端Z，输出是输入和状态的函数，属于Mealy型脉冲异步时序电路。

### (1) 写出输出函数和激励函数表达式

$$Z = xy_2y_1$$

$$J_2 = K_2 = 1; \quad C_2 = y_1$$

$$J_1 = K_1 = 1; \quad C_1 = x$$





## (2) 列出电路次态真值表

J-K触发器的状态转移发生在时钟端脉冲负跳变的瞬间，为了强调在触发器时钟端  $C_1$ 、 $C_2$ 何时有负跳变产生，在次态真值表中用“↓”表示下跳。仅当时钟端有“↓”出现时，相应触发器状态才能发生变化，否则状态不变。

| 输入<br>$x$ | 现态<br>$y_2y_1$ | 激励函数  |       |       |       |       |       | 次态<br>$y_2^{(n+1)}y_1^{(n+1)}$ |
|-----------|----------------|-------|-------|-------|-------|-------|-------|--------------------------------|
|           |                | $J_2$ | $K_2$ | $C_2$ | $J_1$ | $K_1$ | $C_1$ |                                |
| 1         | 0 0            | 1     | 1     |       | 1     | 1     | ↓     | 0 1                            |
| 1         | 0 1            | 1     | 1     | ↓     | 1     | 1     | ↓     | 1 0                            |
| 1         | 1 0            | 1     | 1     |       | 1     | 1     | ↓     | 1 1                            |
| 1         | 1 1            | 1     | 1     | ↓     | 1     | 1     | ↓     | 0 0                            |

用“1”表示输入脉冲

| J | K | $Q^{(n+1)}$ |
|---|---|-------------|
| 0 | 0 | $Q$         |
| 0 | 1 | 0           |
| 1 | 0 | 1           |
| 1 | 1 | $\bar{Q}$   |

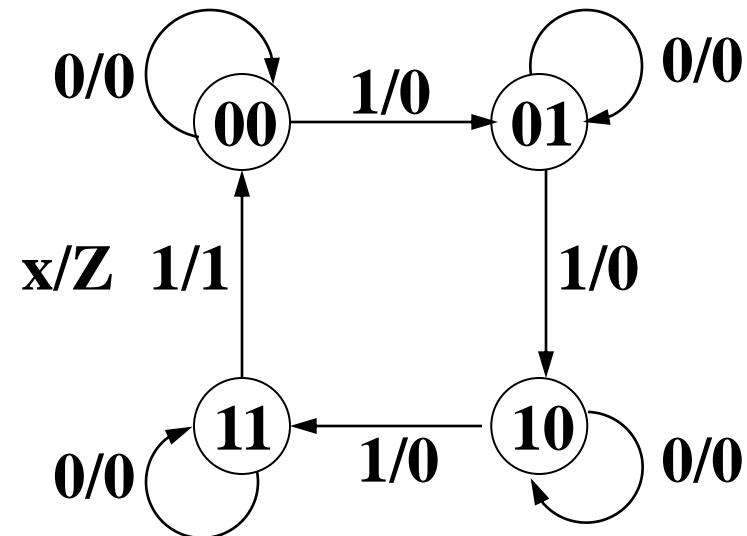




### (3) 作出状态表和状态图

根据次态真值表和输出函数表达式 ( $Z = xy_2y_1$ )，可作出该电路的状态表和状态图如下。

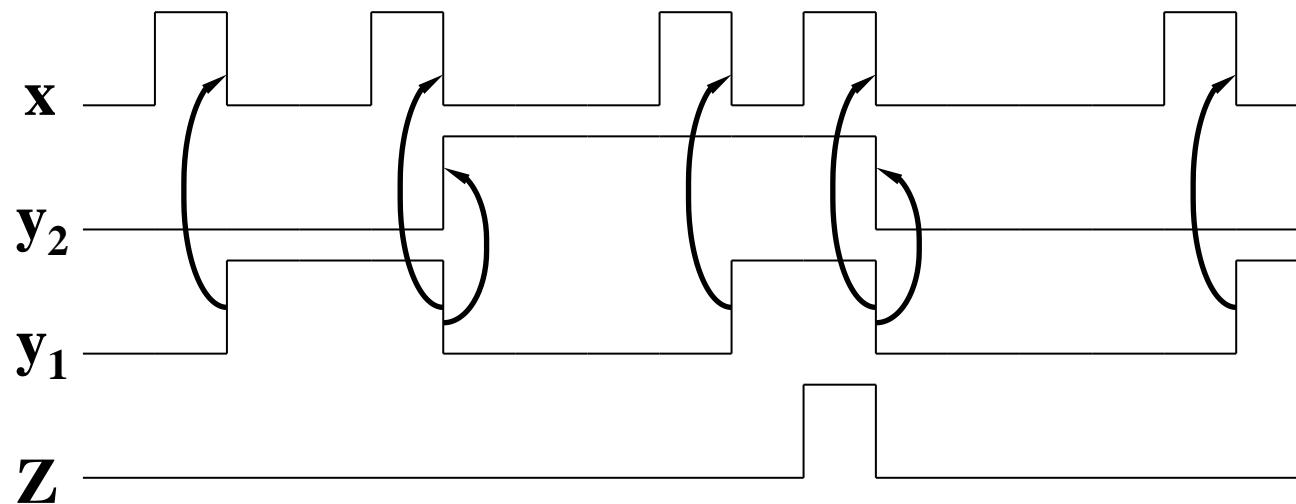
| 现态<br>$y_2 y_1$ | 次态 $y_2^{n+1}y_1^{n+1}$ / 输出 Z |
|-----------------|--------------------------------|
|                 | $x = 1$                        |
| 0 0             | 0 1 / 0                        |
| 0 1             | 1 0 / 0                        |
| 1 0             | 1 1 / 0                        |
| 1 1             | 0 0 / 1                        |





#### (4)画出时间图并说明电路逻辑功能。

为了进一步描述该电路在输入脉冲作用下的状态和输出变化过程，可根据状态表或状态图画出该电路的时间图如下图所示。



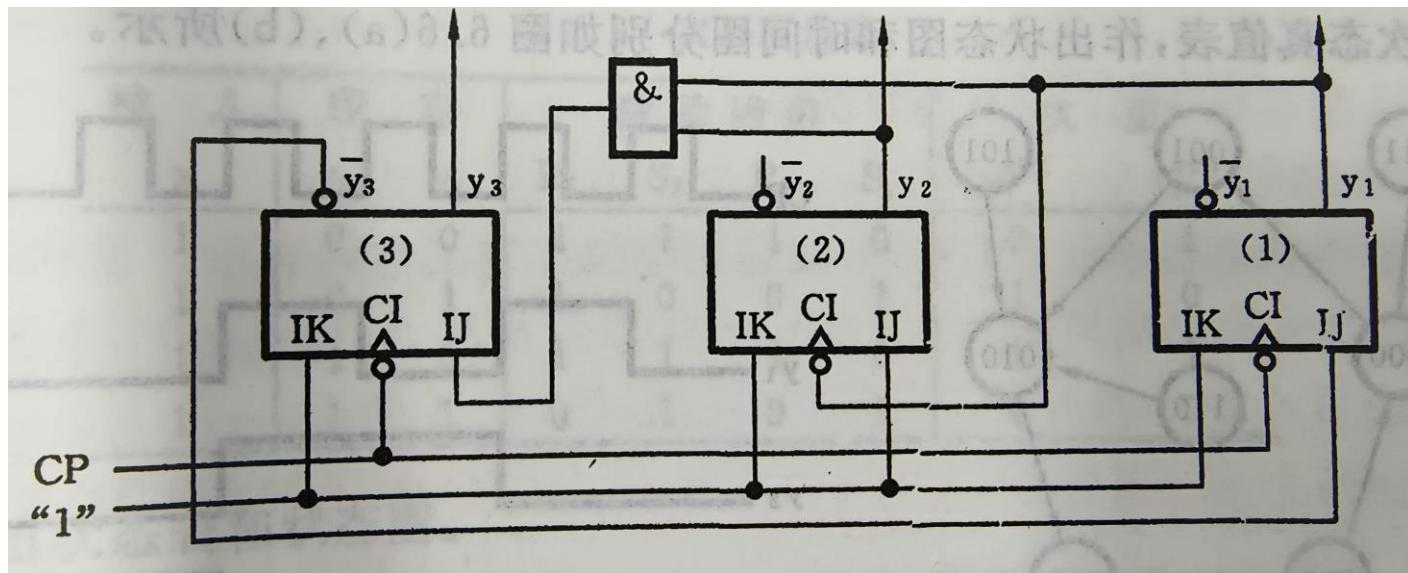
由状态图和时间图可知，该电路是一个**模4加1计数器**，当收到第四个输入脉冲时，电路产生一个进位输出脉冲。

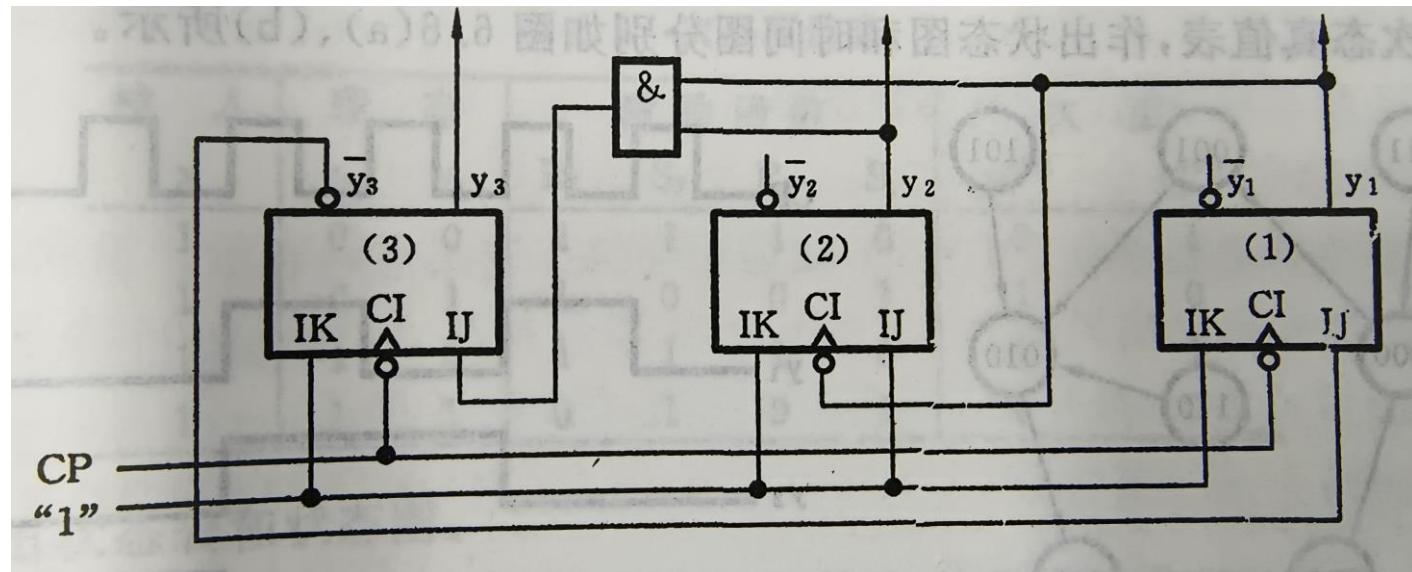




例题：分析下图所示时序逻辑电路，要求：

- (1) 指出该电路属于同步时序电路还是异步时序电路？属于Mealy型电路还是Moore型电路？
- (2) 做出该电路的状态图和时间图，说明该电路的功能。

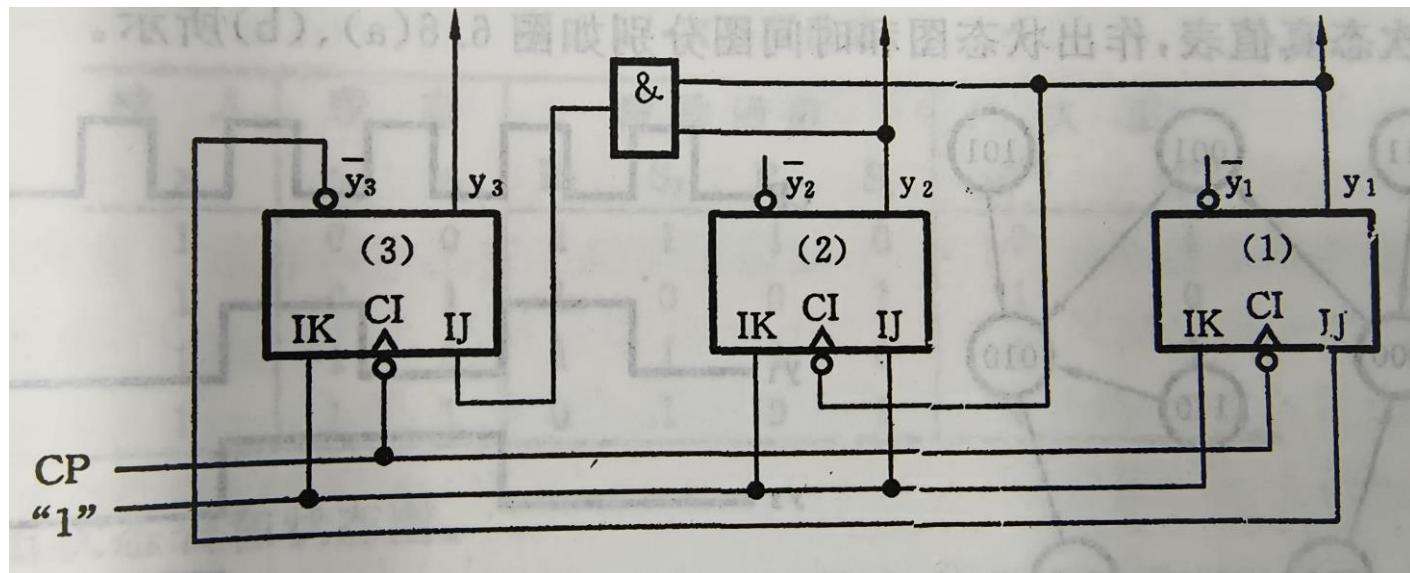




(1) 指出该电路属于同步时序电路还是异步时序电路？属于Mealy型电路还是Moore型电路？

解：由于电路中3个时钟控制触发器的时钟端有两个与CP相连，而另一个与 $y_1$ 相连，因此3个触发器不受统一的时钟控制，该电路的输出即为触发器状态，因此，该电路属于**Moore型脉冲异步时序逻辑电路**。



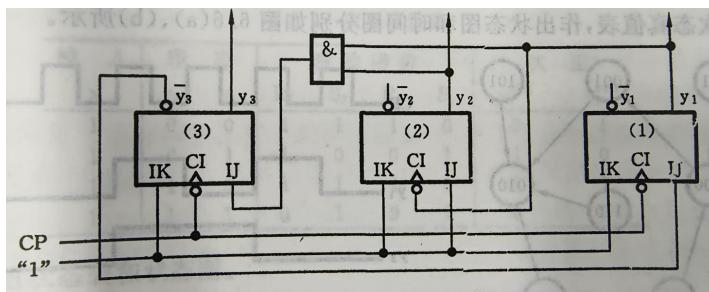


(2) 做出该电路的状态图和时间图，说明该电路的功能。

(a) 写出激励函数表达式：

$$\begin{array}{lll} J_3 = y_2 y_1, & K_3 = 1, & C_3 = CP \\ J_2 = 1, & K_2 = 1, & C_2 = y_1, \\ J_1 = \bar{y}_3, & K_1 = 1, & C_1 = CP, \end{array}$$





(a) 写出激励函数表达式:

$$J_3 = y_2 y_1, \quad K_3 = 1, \quad C_3 = CP$$

$$J_2 = 1, \quad K_2 = 1, \quad C_2 = y_1,$$

$$J_1 = \bar{y}_3, \quad K_1 = 1, \quad C_1 = CP,$$

(b) 列出电路次态真值表:

表 6.1 次态真值表

| 输入 | 现 状态 |       |       | 激 励 函 数 |       |       |       |       |       |       |       | 次 状态  |       |             | J           | K           | $Q^{(n+1)}$ |   |  |
|----|------|-------|-------|---------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------------|-------------|-------------|-------------|---|--|
|    | CP   | $y_3$ | $y_2$ | $y_1$   | $J_3$ | $K_3$ | $C_3$ | $J_2$ | $K_2$ | $C_2$ | $J_1$ | $K_1$ | $C_1$ | $y_3^{n+1}$ | $y_2^{n+1}$ | $y_1^{n+1}$ |             |   |  |
| 1  | 0    | 0     | 0     | 0       | 1     | ↓     |       | 1     | 1     |       | 1     | 1     | ↓     | 0           | 0           | 1           | 00          | Q |  |
| 1  | 0    | 0     | 1     | 0       | 1     | ↓     |       | 1     | 1     | ↓     | 1     | 1     | ↓     | 0           | 1           | 0           | 01          | 0 |  |
| 1  | 0    | 1     | 0     | 0       | 1     | ↓     |       | 1     | 1     |       | 1     | 1     | ↓     | 0           | 1           | 1           | 10          | 1 |  |
| 1  | 0    | 1     | 1     | 1       | 1     | ↓     |       | 1     | 1     | ↓     | 1     | 1     | ↓     | 1           | 0           | 0           | 11          | 0 |  |
| 1  | 1    | 0     | 0     | 0       | 0     | 1     | ↓     | 1     | 1     |       | 0     | 1     | ↓     | 0           | 0           | 0           | 00          | 1 |  |
| 1  | 1    | 0     | 1     | 0       | 0     | 1     | ↓     | 1     | 1     | ↓     | 0     | 1     | ↓     | 0           | 1           | 0           | 01          | 0 |  |
| 1  | 1    | 1     | 0     | 0       | 1     | ↓     |       | 1     | 1     |       | 0     | 1     | ↓     | 0           | 1           | 0           | 10          | 1 |  |
| 1  | 1    | 1     | 1     | 1       | 1     | ↓     |       | 1     | 1     | ↓     | 0     | 1     | ↓     | 0           | 0           | 0           | 11          | 0 |  |

注意：由于状态  $y_2$  对应的触发器时钟端与  $y_1$  相连，而且图中所示 J-K 触发器仅当时钟端信号产生负跳变时才发生翻转，因此仅当  $y_1$  从 1 变为 0 时， $y_2$  才能发生状态转移。



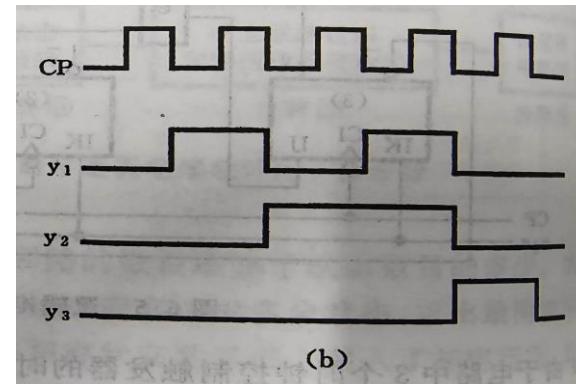
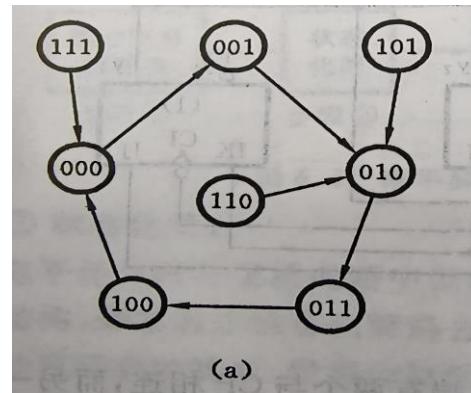


(b) 列出电路次态真值表:

表 6.1 次态真值表

| 输入<br>CP | 现 状   |       |       | 激 励 函 数 |       |       |       |       |       |       |       | 次 状   |             |             |             |
|----------|-------|-------|-------|---------|-------|-------|-------|-------|-------|-------|-------|-------|-------------|-------------|-------------|
|          | $y_3$ | $y_2$ | $y_1$ | $J_3$   | $K_3$ | $C_3$ | $J_2$ | $K_2$ | $C_2$ | $J_1$ | $K_1$ | $C_1$ | $y_3^{n+1}$ | $y_2^{n+1}$ | $y_1^{n+1}$ |
| 1        | 0     | 0     | 0     | 0       | 1     | ↓     | 1     | 1     | 1     | 1     | 1     | 1     | 0           | 0           | 1           |
| 1        | 0     | 0     | 1     | 0       | 1     | ↓     | 1     | 1     | 1     | 1     | 1     | 1     | 0           | 1           | 0           |
| 1        | 0     | 1     | 0     | 0       | 1     | ↓     | 1     | 1     | 1     | 1     | 1     | 1     | 0           | 1           | 1           |
| 1        | 0     | 1     | 1     | 1       | 1     | ↓     | 1     | 1     | 1     | 1     | 1     | 1     | 1           | 0           | 0           |
| 1        | 1     | 0     | 0     | 0       | 1     | ↓     | 1     | 1     | 0     | 0     | 1     | 1     | 0           | 0           | 0           |
| 1        | 1     | 0     | 1     | 0       | 1     | ↓     | 1     | 1     | 1     | 0     | 1     | 1     | 0           | 0           | 1           |
| 1        | 1     | 1     | 0     | 0       | 1     | ↓     | 1     | 1     | 0     | 0     | 1     | 1     | 0           | 1           | 0           |
| 1        | 1     | 1     | 1     | 1       | 1     | ↓     | 1     | 1     | 1     | 0     | 1     | 1     | 0           | 0           | 0           |

(c) 做出状态图和时间图:



由状态图和时间图可知，该电路是一个脉冲异步模5加1计数器，而且具有自启动功能。





## 6.1.3 脉冲异步时序逻辑电路的设计

### 一、方法与步骤

**1. 方法:** 脉冲异步时序逻辑电路设计的方法与同步时序逻辑电路设计大致相同，主要应注意**两个问题**。

(1)由于不允许两个或两个以上输入端同时为1（用1表示有脉冲出现），设计时可以作如下处理：

→ 当有多个输入信号时，只需考虑多个输入信号中仅一个为1的情况；

→ 在确定激励函数和输出函数时，可将两个或两个以上输入同时为1的情况作为无关条件处理。

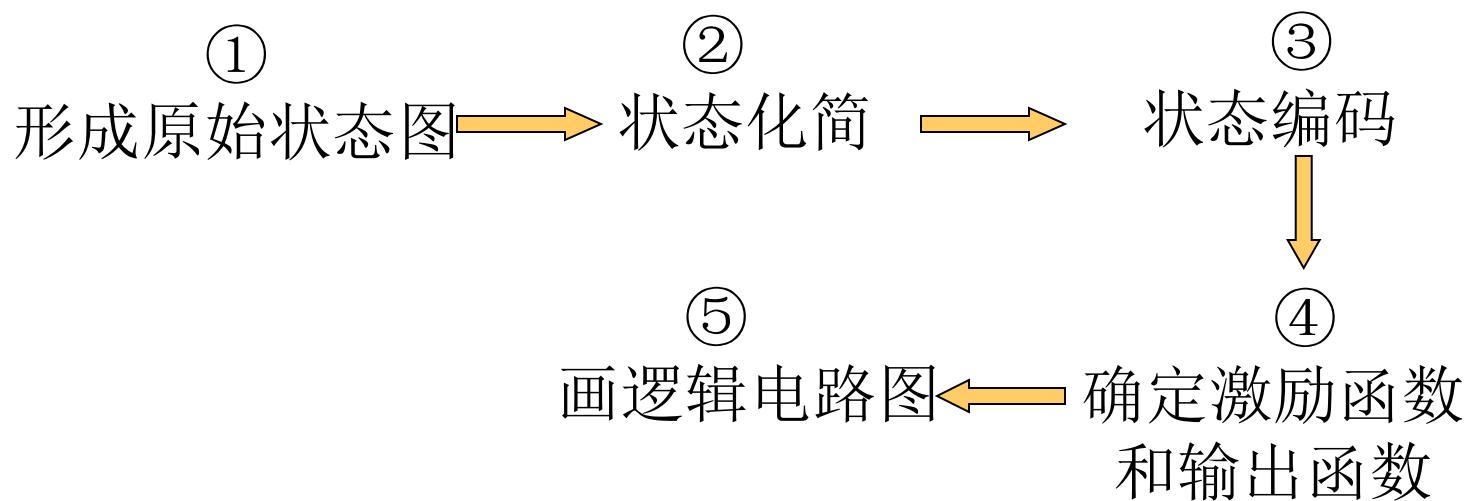
(2)当存储电路采用带时钟控制端的触发器时，触发器的时钟端应作为激励函数处理。设计时通过对触发器的时钟端和输入端综合处理，有利于函数简化。





## 2. 步骤

设计过程与同步时序电路相同，具体如下：





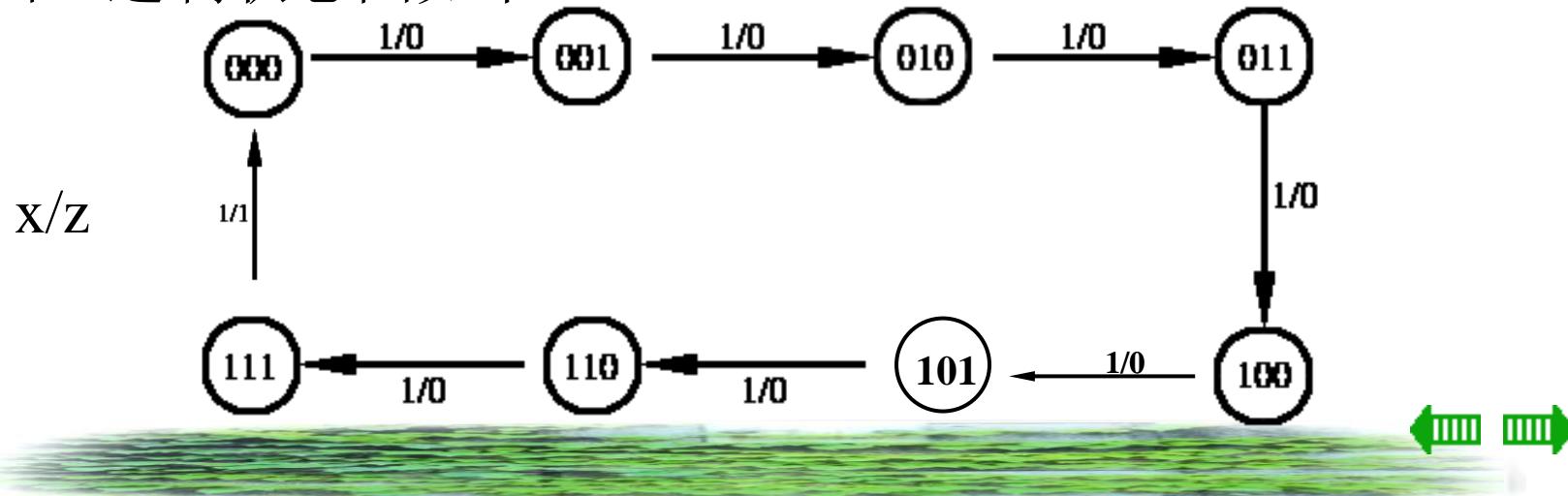
## 二、举例

例 用T触发器作为存储元件，设计一个异步模8加1计数器，电路对输入端x出现的脉冲进行计数，当收到第八个脉冲时，输出端Z产生一个进位输出脉冲。

解 由题意可知，该电路模型为Mealy型（输出受输入直接影响）。由于状态数目和状态转换关系非常清楚，可直接作出二进制状态图和状态表。

### (1) 作出状态图和状态表

设电路初始状态为“000”，状态变量用 $y_2$ 、 $y_1$ 、 $y_0$ 表示，可作出二进制状态图如下。





相应二进制状态表为：

| 现态<br>$y_2\ y_1\ y_0$ | 次态 $y_2^{n+1}y_1^{n+1}y_0^{n+1}$ / 输出Z<br>$x = 1$ |
|-----------------------|---|
| 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                 | 1 0 1 / 0   |
| 1 0 1                 | 1 1 0 / 0   |
| 1 1 0                 | 1 1 1 / 0   |
| 1 1 1                 | 0 0 0 / 1   |





## (2) 确定激励函数和输出函数

假定状态不变时，令相应触发器的时钟端为0，输入端T任意；而状态需要改变时，令相应触发器的时钟端为1(有脉冲出现)，T端为1。

根据状态表，可得到x为1时激励函数和输出函数真值表：

| 输入脉冲<br><b>x</b> | 现 态<br>$y_2 \ y_1 \ y_0$ | 次 态<br>$y_2^{(n+1)} \ y_1^{(n+1)} \ y_0^{(n+1)}$ | C <sub>2</sub> T <sub>2</sub> C <sub>1</sub> T <sub>1</sub> C <sub>0</sub> T <sub>0</sub> | 激 励 函 数 | 输出<br><b>Z</b> |
|------------------|--------------------------|--|---|---------|----------------|
| 1                | 0 0 0                    | 0 0 1  | 0 d 0 d   | 1 1 1 1 | 0              |
| 1                | 0 0 1                    | 0 1 0  | 0 d 1 1   | 1 1 1 1 | 0              |
| 1                | 0 1 0                    | 0 1 1  | 0 d 0 d   | 1 1 1 1 | 0              |
| 1                | 0 1 1                    | 1 0 0  | 1 1 1 1   | 1 1 1 1 | 0              |
| 1                | 1 0 0                    | 1 0 1  | 0 d 0 d   | 1 1 1 1 | 0              |
| 1                | 1 0 1                    | 1 1 0  | 0 d 1 1   | 1 1 1 1 | 0              |
| 1                | 1 1 0                    | 1 1 1  | 0 d 0 d   | 1 1 1 1 | 0              |
| 1                | 1 1 1                    | 0 0 0  | 1 1 1 1   | 1 1 1 1 | 1              |





根据激励函数和输出函数真值表，并考虑到x为0时(无脉冲输入，电路状态不变)，可令各触发器时钟端为0，输入端T随意。可得到简化后的激励函数和输出函数表达式如下：

$$C_2 = xy_1y_0 \quad ; \quad T_2 = 1$$

$$C_1 = xy_0 \quad ; \quad T_1 = 1$$

$$C_0 = x \quad ; \quad T_0 = 1$$

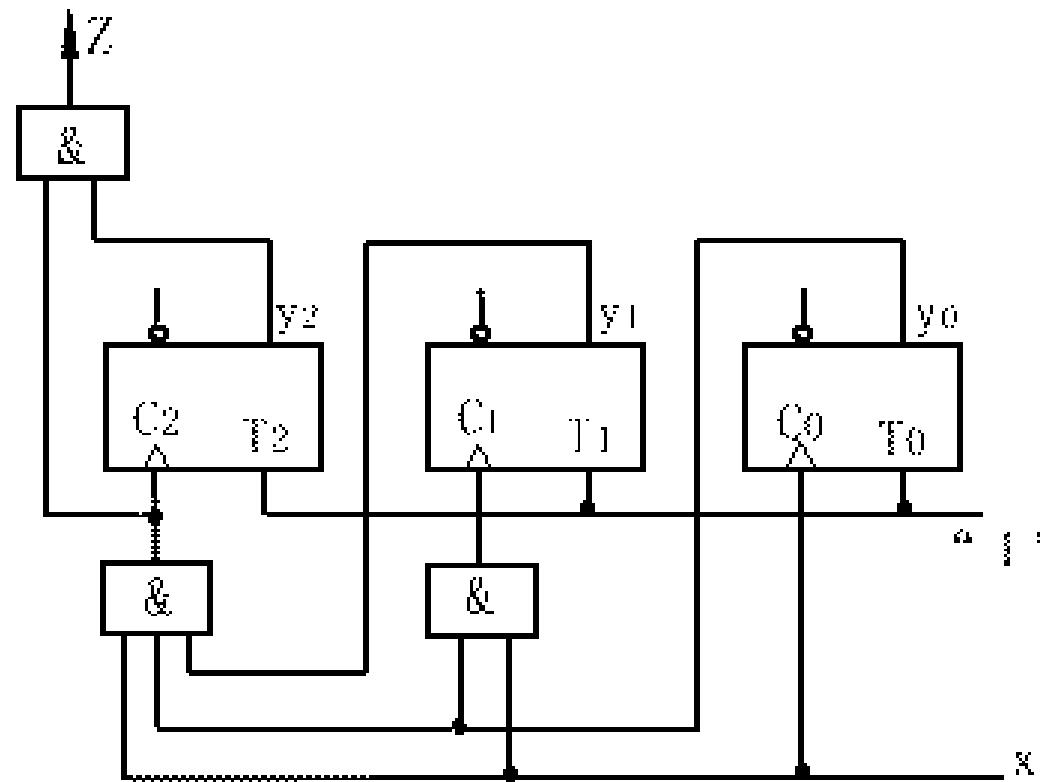
$$Z = xy_2y_1y_0$$





## (3)画出逻辑电路图

根据激励函数和输出函数表达式，可画出实现给定要求的逻辑电路如下图所示。





## 6.2 电平异步时序逻辑电路

### 6.2.1 概述

前面所述同步时序电路和脉冲异步时序电路有两个共同的特点：

- ☆ 电路状态的转换是在脉冲作用下实现的；
- ☆ 电路对过去输入信号的记忆由触发器的状态体现。





事实上，对上述特点可进一步理解如下：

● 脉冲信号只不过是电平信号的一种特殊形式。

● 电路中的触发器，不管是哪种类型，都是由逻辑门加反馈回路构成的。

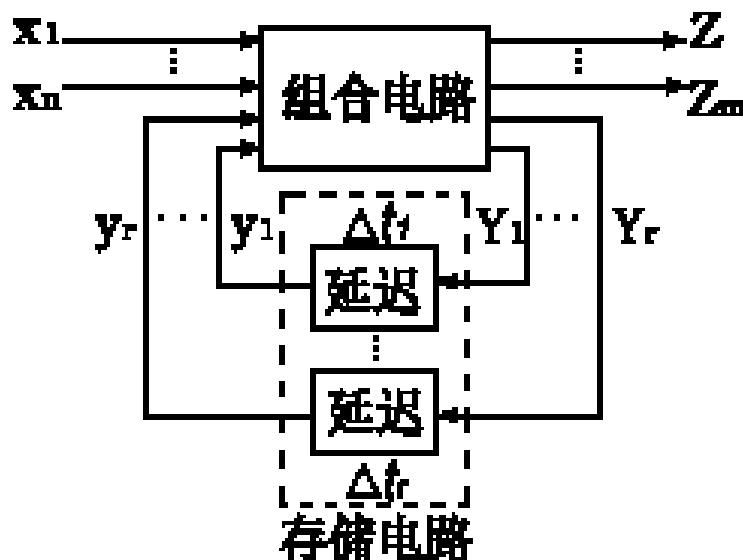
将上述两个特点一般化，便可得到时序逻辑电路中更具一般性的另一类电路——**电平异步时序逻辑电路**。





# 一、电平异步时序逻辑电路的结构特点

## 1. 结构框图



电平异步时序逻辑电路的结构模型

图中：

$x_1, \dots, x_n$ : 外部输入信号;

$Z_1, \dots, Z_m$ : 外部输出信号;

$Y_1, \dots, Y_r$ : 激励状态;

$y_1, \dots, y_r$ : 二次状态;

$\Delta t_1, \dots, \Delta t_r$ : 反馈回路中的时间延迟。

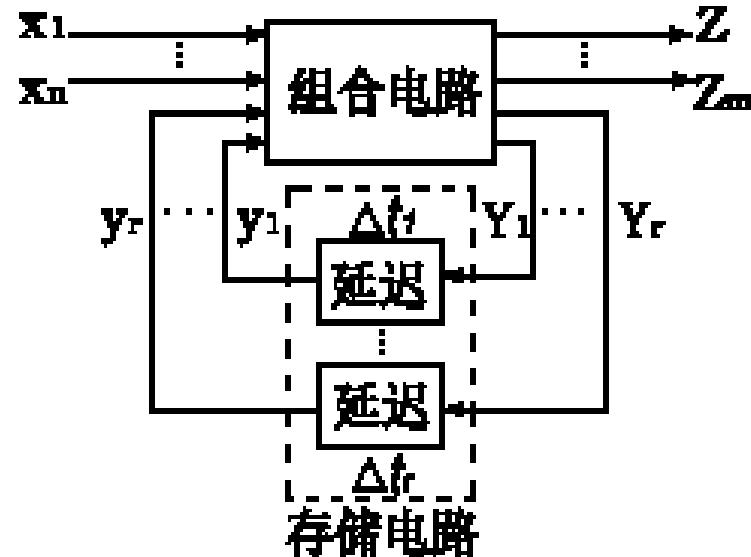
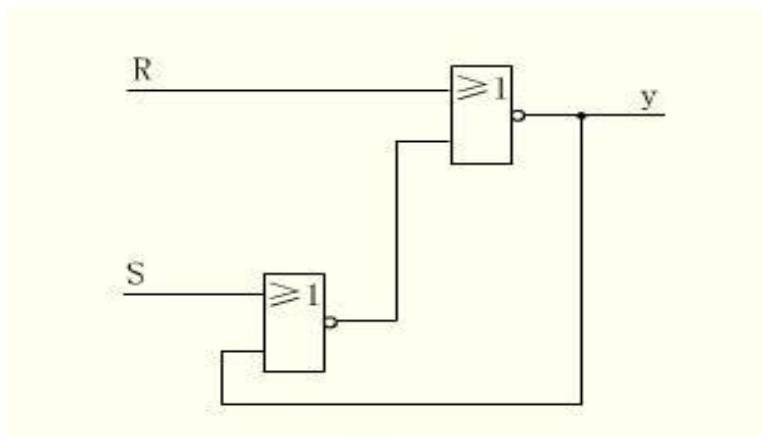




## 2. 组成

电平异步时序逻辑电路可由逻辑门加反馈组成。

例如：用“或非”门构成的R-S触发器。



电平异步时序逻辑电路的结构模型

## 3. 逻辑方程

电路可用以下逻辑方程组描述：

$$Z_i = f_i(x_1, \dots, x_n, y_1, \dots, y_r) \quad i=1, \dots, m$$

$$Y_j = g_j(x_1, \dots, x_n, y_1, \dots, y_r) \quad j=1, \dots, r$$

$$y_j(t + \Delta t_j) = Y_j(t)$$





## 4. 电平异步时序逻辑电路的特点

电平异步时序电路具有如下特点：

(1) 电路输出和状态的改变是由输入信号电位的变化直接引起的，工作速度较高；

(2) 电路的二次状态和激励状态仅仅相差一个时间延迟。

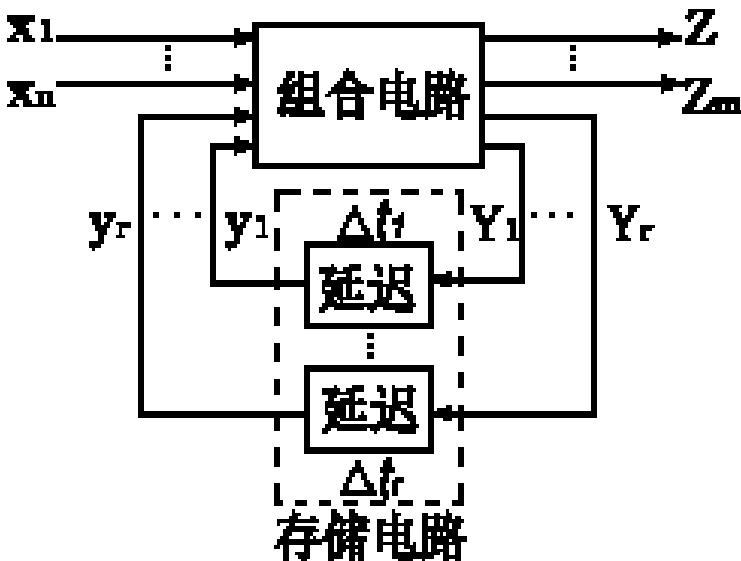
二次状态 $y$ 是激励状态 $Y$ 经过延迟 $\Delta t$ 后的“重现”。

(3) 输入信号的一次变化可能引起二次状态的多次变化。

(4) 电路在状态转换过程中存在稳定状态和非稳定状态。

稳定状态：  $Y=y$

非稳定状态：  $Y \neq y$



电平异步时序逻辑电路的结构模型

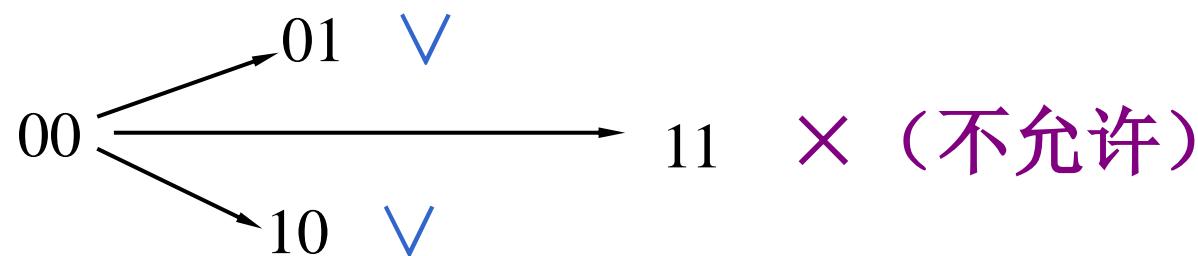




## 5. 输入信号的约束

(1) 不允许两个或两个以上输入信号同时发生变化。

例如，



(2) 输入信号变化引起的电路响应必须完全结束后,才允许输入信号再次变化。换句话说, 必须使电路进入稳定状态后, 才允许输入信号发生变化。





## 二. 电平异步时序逻辑电路的描述方法

### 1. 用逻辑方程描述

电路可用以下逻辑方程组描述：

$$Z_i = f_i(x_1, \dots, x_n, y_1, \dots, y_r) \quad i=1, \dots, m$$

$$Y_j = g_j(x_1, \dots, x_n, y_1, \dots, y_r) \quad j=1, \dots, r$$

$y_j(t + \Delta t_j) = Y_j(t)$  (二次状态y是激励状态Y  
经过延迟 $\Delta t$ 后的“重现”)

### 2. 流程表

**流程表：**是一种以卡诺图的格式反映电路输出信号、激励状态与电路输入信号、二次状态之间关系的一种表格。





流程表的一般格式如下表所示。

Mealy型流程表格式

| 二次<br>状态 | 激励状态 / 输出 |  |
|----------|-----------|--|
|          | 输入 x      |  |
|          |           |  |
|          |           |  |
| y        | y/z       |  |
|          |           |  |

Moore型流程表格式

| 二次<br>状态 | 激励状态 |   | 输出 |
|----------|------|---|----|
|          | 输入 x |   |    |
|          |      |   |    |
|          |      |   |    |
| y        |      | y | z  |
|          |      |   |    |

构造流程表应注意两点：

- 将表中与二次状态相同的激励状态加上圆圈，以表示电路处于稳态，否则处于非稳态。
- 将一位输入的各种取值按代码相邻的关系排列(与卡诺图相同)，以表示输入信号只能在相邻位置上发生变化。





例如，用或非门构成的基本R-S触发器是一个最简单的电平异步时序逻辑电路。该电路的状态即输出，属于Moore型电平异步时序逻辑电路的特例。其激励方程为

$$Y = S + \bar{R}y$$

根据激励方程和约束条件 $RS = 0$ ，可作出相应流程表如下表所示。

R-S触发器流程表

| 二次状态<br>$y$ | 激励状态 Y |       |       |       | 输出 |
|-------------|--------|-------|-------|-------|----|
|             | RS=00  | RS=01 | RS=11 | RS=10 |    |
| 0           | ①      | 1     | d     | ①     | 0  |
| 1           | ①      | ①     | d     | 0     | 1  |





### 3. 总态图

电平异步时序逻辑电路在输入信号作用下存在稳态和非稳态，而且在同一输入信号作用下，可能有一个稳态也可能有多个稳态，为了对电路的工作状态和逻辑功能作出确切的说明，除了流程表和常用的时间图外，引入了**总态**和**总态图**的概念。

**总态：指电路输入和二次状态的组合，记作 $(x,y)$**

在流程表中，代表某种输入取值的一列和代表某个二次状态的一行的交叉点对应一个总态。

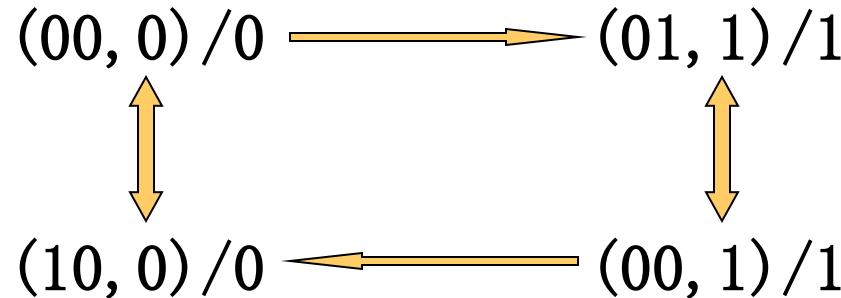
**总态图：反映稳定总态之间转移关系及相应输出的一种有向图**





一个电平异步时序逻辑电路的逻辑功能，是由该电路在输入作用下各**稳定总态**之间的转移关系以及各时刻的输出来体现的。总态图能够清晰地描述一个电路的逻辑功能。

例如，R-S触发器的流程表所对应的总态图如下图所示。



R-S触发器流程表

| 二次状态<br>Y | 激励状态 Y |       |       |       | 输出 |
|-----------|--------|-------|-------|-------|----|
|           | RS=00  | RS=01 | RS=11 | RS=10 |    |
| 0         | ①      | 1     | d     | ①     | 0  |
| 1         | ①      | ①     | d     | 0     | 1  |





## 6.2.2 电平异步时序逻辑电路的分析

### 一、一般步骤

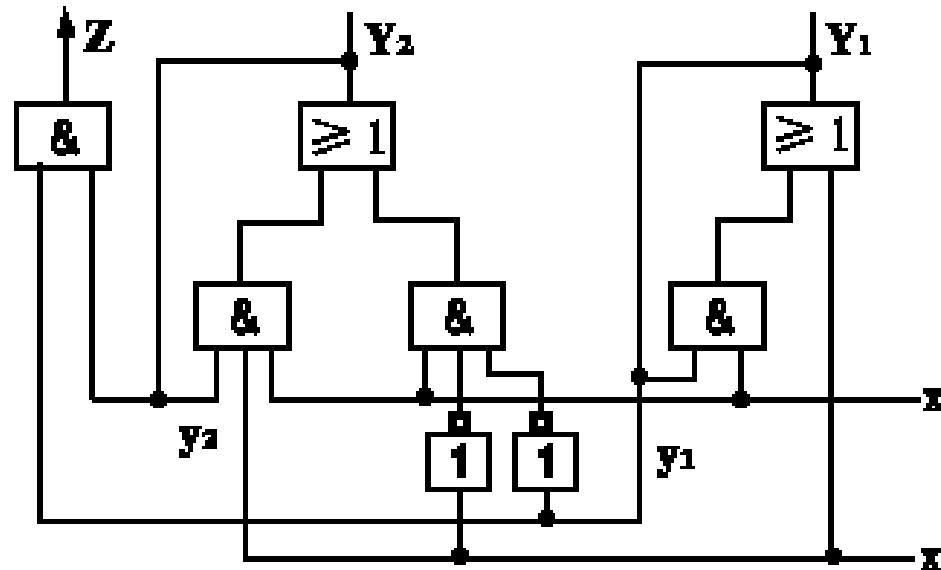
- (1) 根据逻辑电路图写出输出函数和激励函数表达式;  
↓
- (2) 作出流程表;  
↓
- (3) 作出总态图或时间图;  
↓
- (4) 说明电路逻辑功能。





## 二、举例

例 分析下图所示电平异步时序逻辑电路。



逻辑电路图

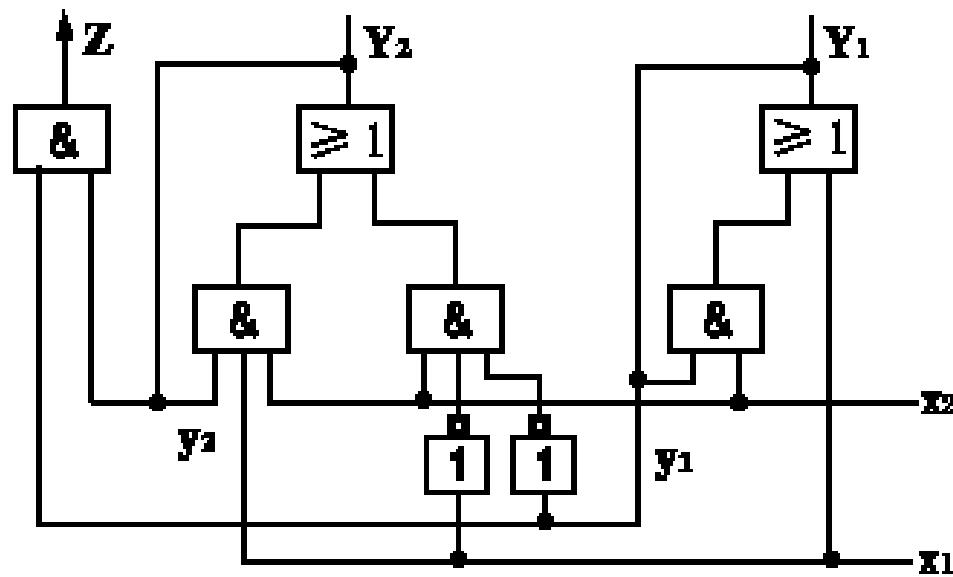
解 该电路有两个外部输入 $x_1$ 、 $x_2$ ；两条反馈回路，对应的激励状态为 $Y_1$ 、 $Y_2$ ，二次状态为 $y_1$ 、 $y_2$ ；一个外部输出 $Z$ 。输出仅仅是状态的函数，属于Moore模型。





## (1) 写出输出函数和激励函数表达式

根据逻辑电路图可写出输出函数和激励函数表达式如下。



逻辑电路图

$$Z = y_2 y_1$$

$$Y_2 = X_2 X_1 y_2 + X_2 X_1 y_1$$

$$Y_1 = X_2 y_1 + X_1$$





## (2) 作出流程表

$$Z = y_2 y_1$$

$$\overline{Y_2} = \overline{X_2 X_1 y_2} + \overline{X_2 X_1 y_1}$$

$$\overline{Y_1} = \overline{X_2 y_1} + \overline{X_1}$$

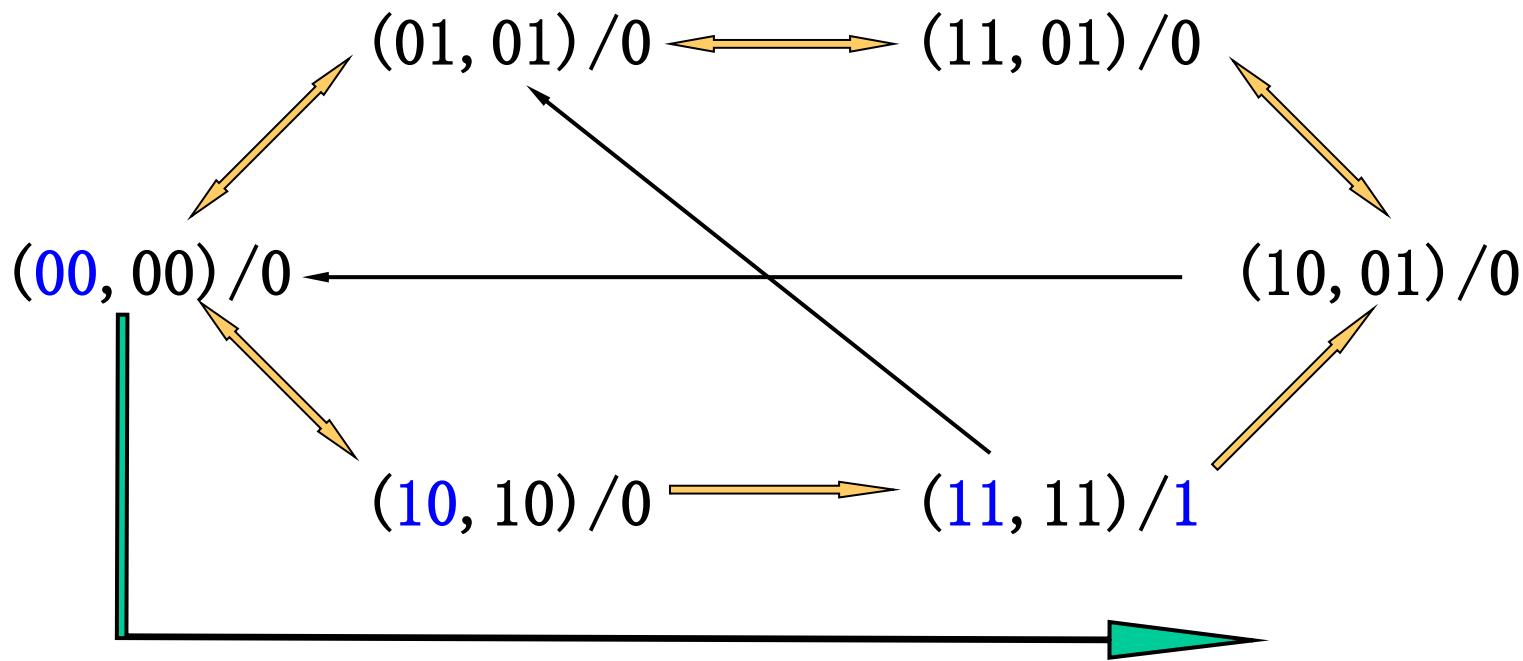
| 二次状态<br>$y_2$ $y_1$ |   | 激励状态 $\overline{Y_2} \overline{Y_1}$ |                |                |                | 输出<br>$Z$ |
|---------------------|---|--------------------------------------|----------------|----------------|----------------|-----------|
|                     |   | $X_2 X_1 = 00$                       | $X_2 X_1 = 01$ | $X_2 X_1 = 11$ | $X_2 X_1 = 10$ |           |
| 0                   | 0 | 00                                   | 01             | 01             | 10             | 0         |
| 0                   | 1 | 00                                   | 01             | 01             | 01             | 0         |
| 1                   | 1 | 00                                   | 01             | 11             | 01             | 1         |
| 1                   | 0 | 00                                   | 01             | 11             | 10             | 0         |





## (3) 作出总态图

| 二次状态<br>$y_2 \quad y_1$ | 激励状态 $Y_2 Y_1$ |                |                |                | 输出<br>Z |
|-------------------------|----------------|----------------|----------------|----------------|---------|
|                         | $X_2 X_1 = 00$ | $X_2 X_1 = 01$ | $X_2 X_1 = 11$ | $X_2 X_1 = 10$ |         |
| 0 0                     | (00)           | 01             | 01             | 10             | 0       |
| 0 1                     | 00             | (01)           | (01)           | (01)           | 0       |
| 1 1                     | 00             | 01             | (11)           | 01             | 1       |
| 1 0                     | 00             | 01             | 11             | (10)           | 0       |



当电路收到输入序列 “ $00 \rightarrow 10 \rightarrow 11$ ”时，才产生一个高电平输出信号，其他情况下均输出低电平。





#### (4) 说明电路功能

从总态图可以看出，仅当电路收到输入序列“ $00 \rightarrow 10 \rightarrow 11$ ”时，才产生一个高电平输出信号，其他情况下均输出低电平。因此，该电路是一个“ $00 \rightarrow 10 \rightarrow 11$ ”序列检测器。





## 6.2.3 电平异步时序逻辑电路反馈回路间的竞争

### 一、竞争现象

前面对电路进行分析时，是在假定各回路之间延迟时间相同的情况下对电路的工作过程进行分析的。事实上，各反馈回路的延迟时间往往各不相同。当电路中存在多条反馈回路，而各回路之间的延时又互不相同时，则可能由于输入信号的变化在反馈回路之间引起竞争。

**竞争：**是指当输入信号变化引起电路中**两个或两个以上**状态变量发生变化时，由于各反馈回路延迟时间的不同，使状态的变化有先有后而导致不同状态响应过程的现象。





## 1. 竞争的两种类型

根据竞争对电路状态转移产生的影响，可将竞争分为非临界竞争和临界竞争两种类型。

**非临界竞争：**若竞争的结果最终能到达预定的稳态，则称为非临界竞争。

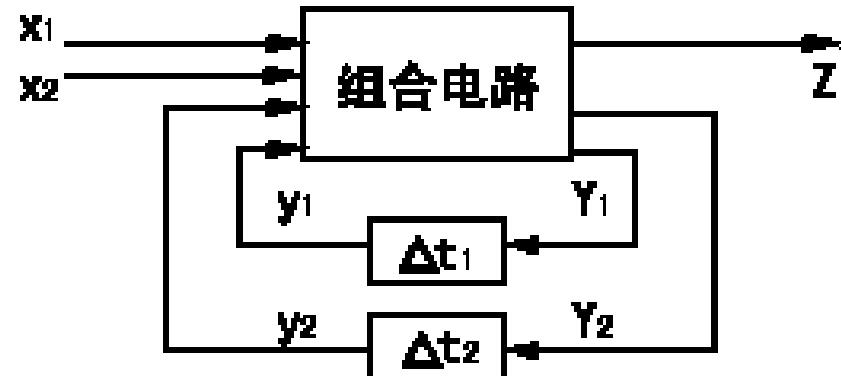
**临界竞争：**若竞争的结果可能使电路到达不同的稳态，即状态转移不可预测，则称为临界竞争。





## 2. 实例分析

例如, 右图所示某电平异步时序电路的流程表如下。



某电平异步时序电路框图

| 二次状态<br>$y_2y_1$ | 激励状态 $Y_2Y_1$ /输出 $Z$ |             |             |             |
|------------------|-----------------------|-------------|-------------|-------------|
|                  | $x_2x_1=00$           | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 00               | (00)/0                | (00)/0      | 01/0        | 11/0        |
| 01               | 00/0                  | (01)/0      | (01)/0      | (01)/0      |
| 11               | 00/0                  | 00/0        | 10/0        | (11)/0      |
| 10               | 00/0                  | 00/0        | (10)/0      | (10)/0      |

当电路处在稳定总态 (00, 00) 和 (10, 11), 输入发生变化时, 电路状态响应过程将如何呢?





| 二次状态<br>$y_2 \quad y_1$ |   | 激励状态 $Y_2Y_1$ /输出 Z |             |             |             |
|-------------------------|---|---------------------|-------------|-------------|-------------|
|                         |   | $X_2X_1=00$         | $X_2X_1=01$ | $X_2X_1=11$ | $X_2X_1=10$ |
| 0                       | 0 | (00)/0              | (00)/0      | 01/0        | 11/0        |
| 0                       | 1 | 00/0                | (01)/0      | (01)/0      | (01)/0      |
| 1                       | 1 | 00/0                | 00/0        | 10 /0       | (11)/0      |
| 1                       | 0 | 00/0                | 00/0        | (10)/1      | (10)/0      |

从表可以看出，当电路处于稳定总态(00, 00)，输入 $x_2x_1$ 由 $00 \rightarrow 10$ 时，引起激励状态 $Y_2Y_1$ 从00 $\rightarrow$ 11；当电路处于稳定总态(10, 11)、输入 $x_2x_1$ 由10 $\rightarrow$ 00时，激励状态 $Y_2Y_1$ 从11 $\rightarrow$ 00。即两个状态变量均发生变化，所以，当电路中两条反馈回路的延迟时间 $\Delta t_1$ 和 $\Delta t_2$ 不相等时，电路中将产生竞争。





(1) 当电路处于稳定总态(00, 00)、输入 $x_2x_1$ 由00→10时：

| 二次状态<br>$y_2 \quad y_1$ |   | 激励状态 $Y_2Y_1$ /输出Z |             |             |             |
|-------------------------|---|--------------------|-------------|-------------|-------------|
|                         |   | $x_2x_1=00$        | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 0                       | 0 | 00/0               | 00/0        | 01/0        | 11/0        |
| 0                       | 1 | 00/0               | 01/0        | 01/0        | 01/0        |
| 1                       | 1 | 00/0               | 00/0        | 10 /0       | 11/0        |
| 1                       | 0 | 00/0               | 00/0        | 10 /1       | 10/0        |

分析：

$\Delta t_2 = \Delta t_1$ : 到达预定的稳定总态(10, 11)。

$\Delta t_2 < \Delta t_1$ : 电路到达了一个非期望的稳定总态(10, 10)

$\Delta t_2 > \Delta t_1$ : 电路到达了一个非期望的稳定总态(10, 01)。

结论：本次竞争为临界竞争！





(2) 当电路处于稳定总态(10, 11)、输入 $x_2x_1$ 由10→00时，其状态响应过程如下。

| 二次状态<br>$y_2 \quad y_1$ |   | 激励状态 $Y_2Y_1$ /输出Z |             |             |             |
|-------------------------|---|--------------------|-------------|-------------|-------------|
|                         |   | $x_2x_1=00$        | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 0                       | 0 | 00/0               | 00/0        | 01/0        | 11/0        |
| 0                       | 1 | 00/0               | 01/0        | 01/0        | 01/0        |
| 1                       | 1 | 00/0               | 00/0        | 10 /0       | 11/0        |
| 1                       | 0 | 00/0               | 00/0        | 10 /1       | 10/0        |

分析如下：

$\Delta t_2 = \Delta t_1$ : 到达预定的稳定总态(00, 00)。

$\Delta t_2 < \Delta t_1$ : 到达预定的稳定总态(00, 00)。

$\Delta t_2 > \Delta t_1$ : 到达预定的稳定总态(00, 00)。

结论如下：本次竞争属于非临界竞争！





## 用流程表检查电路竞争的一般法则：

☆当从某一稳态出发，输入信号发生允许变化、引起两个或两个以上激励状态同时发生变化时，由于反馈回路之间延迟时间的不同会使电路产生竞争。

☆若输入信号变化所到达的列只有一个稳态，则该竞争属于非临界竞争；若输入信号变化所到达的列有两个或两个以上稳态，则该竞争属于临界竞争。

非临界竞争的存在不会影响电路的正确工作，但临界竞争的存在却将导致电路状态转换的不可预测。为了确保电平异步时序电路能可靠地实现预定功能，电路设计时必须避免发生临界竞争！



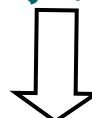


## 6.3 电平异步时序逻辑电路的设计

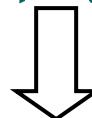
### 6.3.1 设计的一般步骤和方法

一般步骤如下：

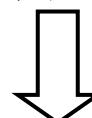
1. 根据设计要求，建立原始流程表；



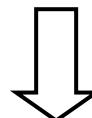
2. 化简原始流程表，得到最简流程表；



3. 状态编码，得到二进制流程表；



4. 确定激励状态和输出函数表达式；



5. 画出逻辑电路图。





## 一、建立原始流程表

原始流程表是对设计要求的一种最原始的抽象。建立原始流程表时通常借助**时间图**或**原始总态图**。即首先根据题意画出典型输入、输出时间图或作出原始总态图。

根据时间图建立原始流程表的过程如下。

### 1. 画出典型输入、输出时间图并设立相应状态

由于电平异步时序电路约定对于每次输入信号变化，必须保证电路进入稳定状态后才允许输入信号再次变化，所以，应根据题意设立与各时刻输入、输出对应的稳定状态。





## 2. 建立原始流程表

根据时间图和所设立的状态建立原始流程表，一般分为3步进行。

### (1) 画出原始流程表，并填入稳定状态和相应输出

由于根据时间图设立状态时，对不同的输入取值总是设立不同的状态进行区分的，这就使得原始流程表中每一行只有一个稳定状态。显然，时间图上设立了多少个状态，原始流程表便有多少行。





## (2) 填入非稳定状态并指定相应的输出，完善流程表

由于表中每行只有一个稳定状态，所以，在稳态下输入信号发生允许变化时，电路不可能直接进入另一个稳态。

假定每次输入信号发生变化时，电路总是经过一个非稳定状态后进入另一个稳定状态，根据时间图中的状态转移关系，可在原始流程表中填入相应的非稳定状态。

**注意：**当从某一稳态出发，输入信号发生允许变化所引起的状态转移，不能用时间图中所设立的状态来表示时，则应根据题意**补充新的状态**，以便无遗漏地反映设计要求。





## 非稳定状态下输出指定的法则为：

若转换前后两个稳定状态的输出相同，则指定非稳定状态下的输出与稳态下的输出相同；若转换前后两个稳定状态的输出不同，则可指定非稳定状态下的输出为任意值“d”。

### (3) 填入无关状态和无关输出

对稳态下输入不允许到达的列，在相应处填入任意状态和任意输出，用“d”表示，即作为无关处理。



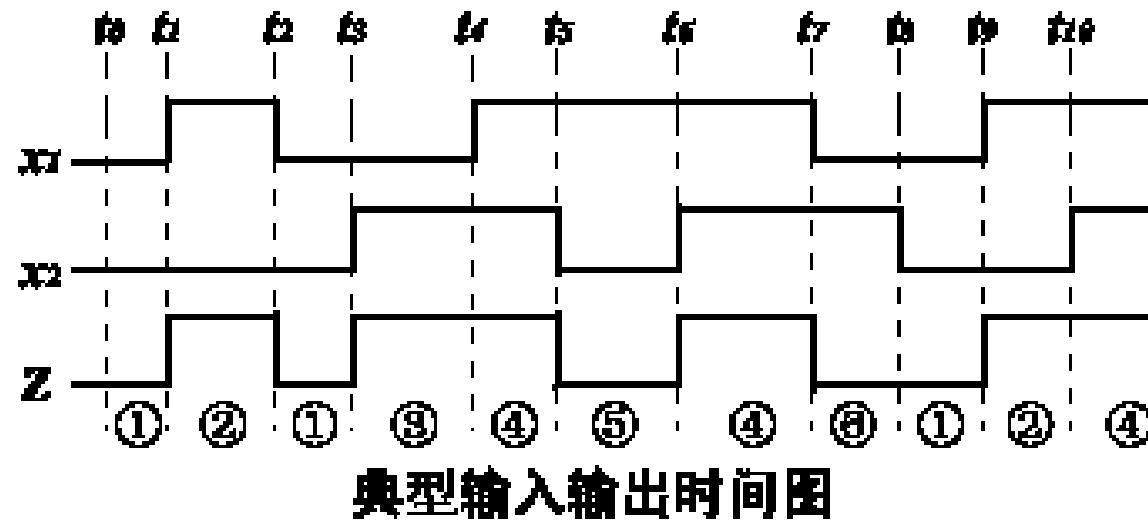


**例** 某电平异步时序逻辑电路有两个输入端 $x_1$ 和 $x_2$ , 一个输出端Z。输出与输入之间的关系为: 若 $x_1x_2=00$ , 则 $Z=0$ , 之后当 $x_1x_2=01$ 或 $10$ 时,  $Z=1$ ; 若 $x_1x_2=11$ , 则 $Z=1$ , 之后当 $x_1x_2=01$ 或 $10$ 时,  $Z=0$ 。作出该电路的原始流程表。

**解** 借助时间图形成该电路原始流程表的过程如下。

### (1) 画出典型输入、输出时间图并设立相应状态

根据题意, 可画出该电路典型输入、输出时间图并设立状态如下。

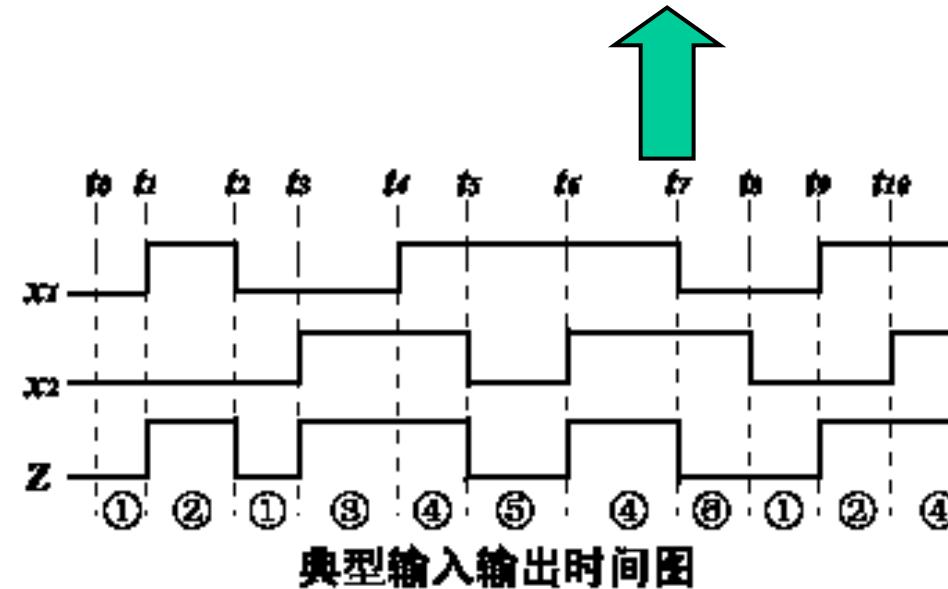




## (2)建立原始流程表

根据波形图中设立的状态和填写原始流程表的步骤，可构造出该问题的原始流程表。

| 二次状态<br>y | 激励状态Y/输出Z                         |                                   |                                   |                                   |
|-----------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|
|           | X <sub>2</sub> X <sub>1</sub> =00 | X <sub>2</sub> X <sub>1</sub> =01 | X <sub>2</sub> X <sub>1</sub> =11 | X <sub>2</sub> X <sub>1</sub> =10 |
| 1         | ①/0                               | 3/d                               | d/d                               | 2/d                               |
| 2         | 1/d                               | d/d                               | 4/1                               | ②/1                               |
| 3         | 1/d                               | ③/1                               | 4/1                               | d/d                               |
| 4         | d/d                               | 6/d                               | ④/1                               | 5/d                               |
| 5         | 1/0                               | d/d                               | 4/d                               | ⑤/0                               |
| 6         | 1/0                               | ⑥/0                               | 4/d                               | d/d                               |





## 二、化简原始流程表

目的-----简化电路结构。

在进行电平异步时序逻辑电路设计时，流程表中的状态数目决定了电路中反馈回路的数目，即状态数目的多少与电路的复杂程度直接相关。为了获得一种经济、合理的设计方案，必须对原始流程表进行化简，求出最简流程表。

由于原始流程表中含有不确定的状态和输出，因而原始流程表的化简是建立在状态相容这一概念基础之上的。





## 1. 相容行的概念

原始流程表中的每一行代表一个稳定状态，因而相容状态的概念被引申为**相容行**的概念。

**相容行：**对于原始流程表中的某两行，如果每一列给定的输出相同，且给定的激励状态**相同、交错、循环、相容或为各自本身**，则这两行为相容行。

**注意：**

☆在检查输出时，对于一个给定而另一个任意，或者两个均任意的情况，作为相同情况处理。





☆在检查激励状态时，按以下原则确定稳定状态、非稳定状态和任意状态的相容性。

- (1) 稳定状态  $i$  和非稳定状态  $i$  相容；
- (2) 若稳定状态  $i$  和  $j$  相容，则稳定状态  $i$  和非稳定状态  $j$  相容；
- (3) 若稳定状态  $i$  和  $j$  相容，则非稳定状态  $i$  和  $j$  相容；
- (4) 稳定状态  $i$  和非稳定状态  $i$  均与任意状态 “d” 相容，任意状态 “d” 与任意状态 “d” 相容。





## 2.化简的方法与一般步骤

化简原始流程表与化简不完全给定状态表的过程类似。

(1) 作**隐含表**, 找出**相容行对**;



(2) 作**合并图**, 求出**最大相容行类**;



(3) 从相容行类中选择一个**最小闭覆盖**;



(4) 状态行**合并**, 作出最简流程表。

注意:

对各相容行类中的相容行进行合并时,当输出存在给定值和任意值“d”时, 合并后取给定值; 当激励状态存在稳定状态和非稳定状态时, 合并时取稳定状态; 当激励状态存在给定状态和任意状态“d”时, 合并时取给定状态。





例 化简如下原始流程表。

| 二次<br>状态<br>$y$ | 激励状态 Y/输出 Z |             |             |             |
|-----------------|-------------|-------------|-------------|-------------|
|                 | $X_2X_1=00$ | $X_2X_1=01$ | $X_2X_1=11$ | $X_2X_1=10$ |
| 1               | ①/0         | 3/d         | d/d         | 2/d         |
| 2               | 1/d         | d/d         | 4/1         | ②/1         |
| 3               | 1/d         | ③/1         | 4/1         | d/d         |
| 4               | d/d         | 6/d         | ④/1         | 5/d         |
| 5               | 1/0         | d/d         | 4/d         | ⑤/0         |
| 6               | 1/0         | ⑥/0         | 4/d         | d/d         |



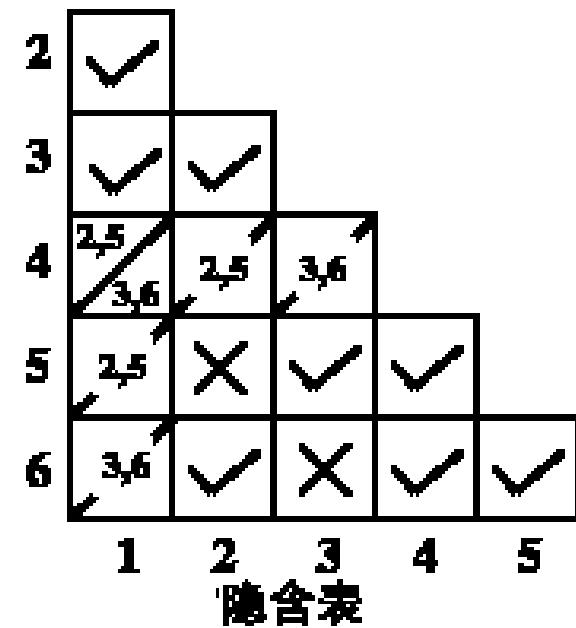


解 根据化简原始流程表的方法和步骤，化简过程如下：

### (1) 作隐含表，找相容行

原始流程表对应的隐含表如右下图所示。

| 二次<br>状态<br>$y$ | 激励状态 Y/输出 Z |             |             |             |
|-----------------|-------------|-------------|-------------|-------------|
|                 | $X_2X_1=00$ | $X_2X_1=01$ | $X_2X_1=11$ | $X_2X_1=10$ |
| 1               | ①/0         | 3/d         | d/d         | 2/d         |
| 2               | 1/d         | d/d         | 4/1         | ②/1         |
| 3               | 1/d         | ③/1         | 4/1         | d/d         |
| 4               | d/d         | 6/d         | ④/1         | 5/d         |
| 5               | 1/0         | d/d         | 4/d         | ⑤/0         |
| 6               | 1/0         | ⑥/0         | 4/d         | d/d         |



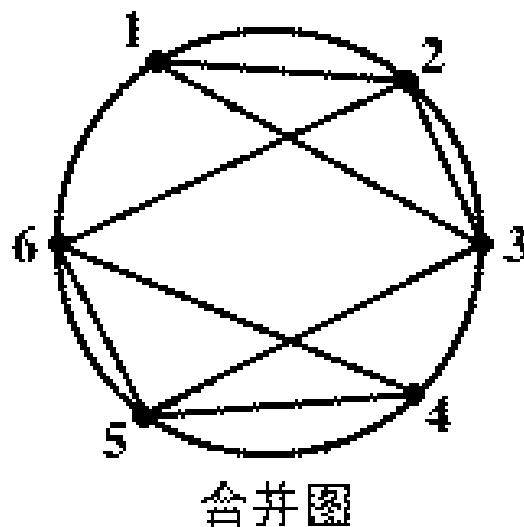
根据相容行的判断规则，可找出相容行对：(1, 2), (1, 3), (2, 3), (2, 6), (3, 5), (4, 5), (4, 6), (5, 6)。





## (2)作合并图，求最大相容行类

根据所得出的相容行对，可作出合并图如下图所示。



由合并图可知，最大相容行类为：

$(1, 2, 3)$ ,  $(4, 5, 6)$ ,  
 $(3, 5)$ ,  $(2, 6)$ 。

## (3)选择一个最小闭覆盖

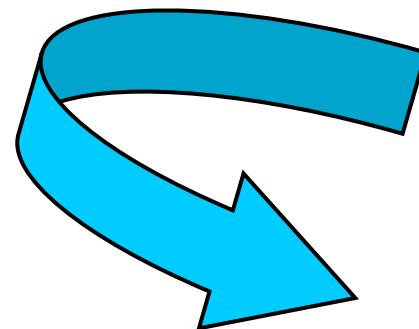
选择最大相容行类构成的集合 $\{(1, 2, 3), (4, 5, 6)\}$ ，便可满足覆盖、闭合和最小3个条件。所以，该集合即为原始流程表的最小闭覆盖。





#### (4)作出最简流程表

令: (1, 2, 3) —— A,  
(4, 5, 6) —— B, 可得到  
最简流程表如下表所示。



| 二次状态<br>$y$ | 激励状态 Y/输出 Z |             |             |             |
|-------------|-------------|-------------|-------------|-------------|
|             | $X_2X_1=00$ | $X_2X_1=01$ | $X_2X_1=11$ | $X_2X_1=10$ |
| 1           | ①/0         | 3/d         | d/d         | 2/d         |
| 2           | 1/d         | d/d         | 4/1         | ②/1         |
| 3           | 1/d         | ③/1         | 4/1         | d/d         |
| 4           | d/d         | 6/d         | ④/1         | 5/d         |
| 5           | 1/0         | d/d         | 4/d         | ⑤/0         |
| 6           | 1/0         | ⑥/0         | 4/d         | d/d         |

最简流程表

| 二次状态<br>$y$ | 激励状态 Y/输出 Z |             |             |             |
|-------------|-------------|-------------|-------------|-------------|
|             | $X_2X_1=00$ | $X_2X_1=01$ | $X_2X_1=11$ | $X_2X_1=10$ |
| A           | ④/0         | ④/1         | B/1         | ④/1         |
| B           | A/0         | ⑩/0         | ⑩/1         | ⑩/0         |





## 习题6-1：简化下面的原始流程表

例 6-8 简化表 6.13 所示原始流程表。

表 6.13 原始流程表

| 二次状态 | 激励状态/输出 Z      |                |                |                |
|------|----------------|----------------|----------------|----------------|
|      | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| 1    | ①/0            | 5/0            | d/d            | 2/d            |
| 2    | 1/0            | d/d            | 3/d            | ②/0            |
| 3    | d/d            | 5/d            | ③/1            | 4/1            |
| 4    | 1/d            | d/d            | 3/1            | ④/1            |
| 5    | 1/0            | ⑤/0            | 6/0            | d/d            |
| 6    | d/d            | 5/0            | ⑥/0            | 4/d            |





### 三、状态编码

状态编码的任务：

- (1) 根据化简后的状态数目确定二进制代码的位数；
- (2) 选择一种合适的状态分配方案，将每个状态用一个二进制代码表示。

**注意：**确定分配方案时应考虑的主要问题是如何避免反馈回路之间的临界竞争，保证电路可靠地实现预定功能。

常用的几种方法：

#### 1. 相邻状态，相邻分配

**相邻状态：**是指稳态下输入取值作相邻变化时，需要直接发生转换的状态。

**相邻分配：**是指分配给相邻状态的代码为相邻代码。





通常借助状态相邻图确定流程表中各状态的相邻关系。

**状态相邻图:**将流程表中的每一个状态加圈表示，并从每一个稳态出发，找出输入取值作相邻变化时的下一个稳态，用有向线段将其连接起来，表示这两个状态为相邻状态。

**例** 对如下流程表进行状态编码，求出二进制流程表。

流程表

| 二次状态<br>Y | 激励状态 Y      |             |             |             |
|-----------|-------------|-------------|-------------|-------------|
|           | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| A         | (A)         | (A)         | B           | C           |
| B         | A           | (B)         | (B)         | (B)         |
| C         | (C)         | A           | D           | (C)         |
| D         | C           | (D)         | (D)         | (D)         |

**解** 根据“相邻状态，相邻分配”的法则，首先根据流程表作出状态相邻图。



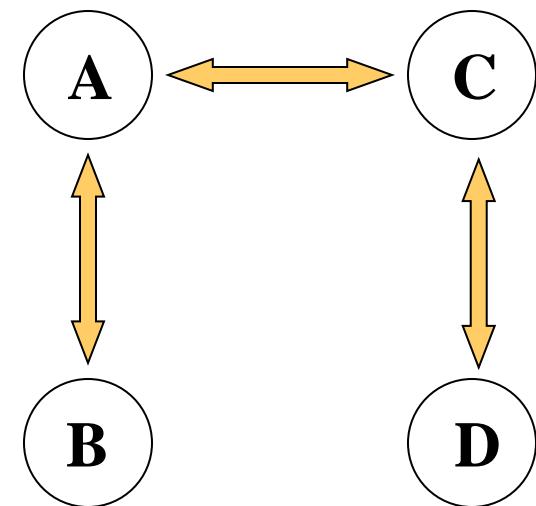


根据流程表作出状态相邻图如右下图所示。

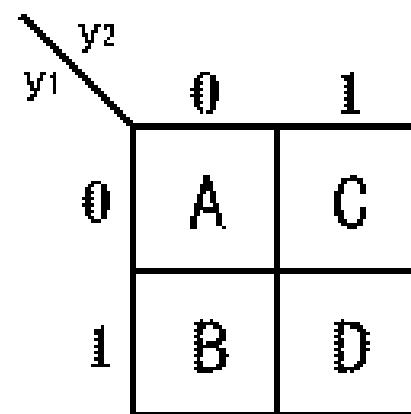
| 二次状态<br>$y$ | 激励状态 Y    |           |           |           |
|-------------|-----------|-----------|-----------|-----------|
|             | $xx_1=00$ | $xx_1=01$ | $xx_1=11$ | $xx_1=10$ |
| A           | B         | B         | C         | C         |
| B           | A         | D         | D         | D         |
| C           | C         | A         | D         | C         |
| D           | C         | D         | D         | D         |

流程表中共有4个状态，需两位代码，设二次状态用 $y_2$ 、 $y_1$ 表示。根据相邻图，可选择如右卡诺图所示的状态分配方案。

即用00表示A，01表示B，10表示C，11表示D。



状态相邻图

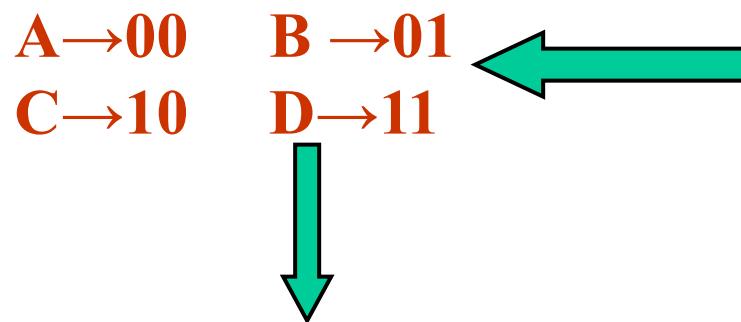


状态邻接图





将给定流程表中的状态用相应二进制编码表示，即可得到相应二进制流程表。



流程表

| 二次状态<br>$y$ | 激励状态 $Y$    |             |             |             |
|-------------|-------------|-------------|-------------|-------------|
|             | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| A           | ④           | ④           | B           | C           |
| B           | ⑧           | ⑩           | ⑩           | ⑪           |
| C           | ⑥           | ⑧           | D           | ⑨           |
| D           | ⑦           | ⑪           | ⑪           | ⑦           |

| 二次状态<br>$y_2 \quad y_1$ | 激励状态 $Y_2Y_1$ |             |             |             |
|-------------------------|---------------|-------------|-------------|-------------|
|                         | $x_2x_1=00$   | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| 0 0                     | ⑥             | ⑥           | ①           | ⑩           |
| 0 1                     | ⑥             | ⑩           | ⑩           | ⑩           |
| 1 1                     | ⑩             | ⑪           | ⑪           | ⑪           |
| 1 0                     | ⑩             | ⑥           | ⑪           | ⑩           |

该二进制流程表，消除了竞争现象。





## 2. 增加过渡状态，实现相邻分配

对于某些流程表，尽管相邻图上状态的最大相邻状态数L不大于状态分配的最小代码位数m，但状态之间的相邻关系形成由奇数个状态构成的闭环，因而无法直接实现状态的相邻分配。解决这类问题的一种常用的方法是通过增加过渡状态，实现相邻分配，得到一个无竞争的二进制流程表。

**例** 对如下流程表进行状态编码，得到二进制流程表。

| 二次<br>状态<br>$y$ | 激励状态Y       |             |             |             |
|-----------------|-------------|-------------|-------------|-------------|
|                 | $X_2X_1=00$ | $X_2X_1=01$ | $X_2X_1=11$ | $X_2X_1=10$ |
| A               | (A)         | B           | C           | (A)         |
| B               | A           | (B)         | (B)         | C           |
| C               | A           | B           | (C)         | (C)         |

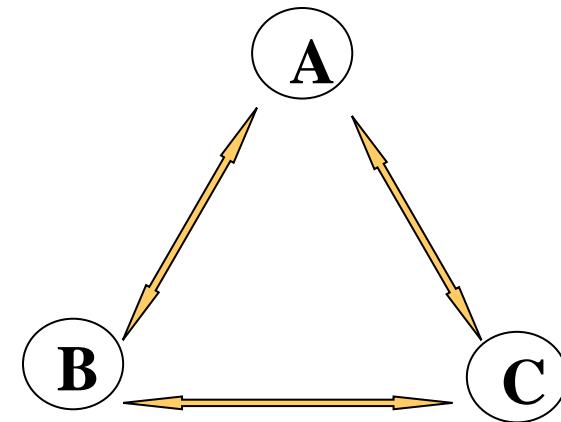




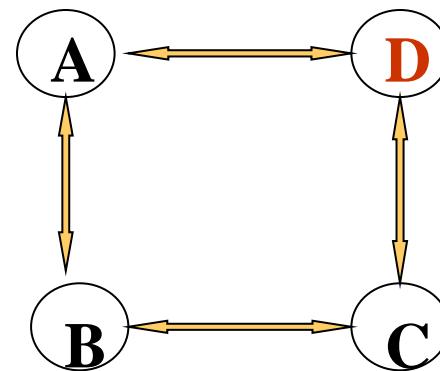
解 根据给定流程表可作出状态相邻图如右下图所示。

尽管相邻图上每个状态只有两个相邻状态，但由于3个状态之间的相邻关系构成一个闭环，所以，用两位代码无法满足其相邻关系。

| 二次<br>状态<br>$y$ | 激励状态Y       |             |             |             |
|-----------------|-------------|-------------|-------------|-------------|
|                 | $X_2X_1=00$ | $X_2X_1=01$ | $X_2X_1=11$ | $X_2X_1=10$ |
| A               | (A)         |             |             | (A)         |
| B               | A           | (B)         |             | C           |
| C               | A           | B           | (C)         | (C)         |

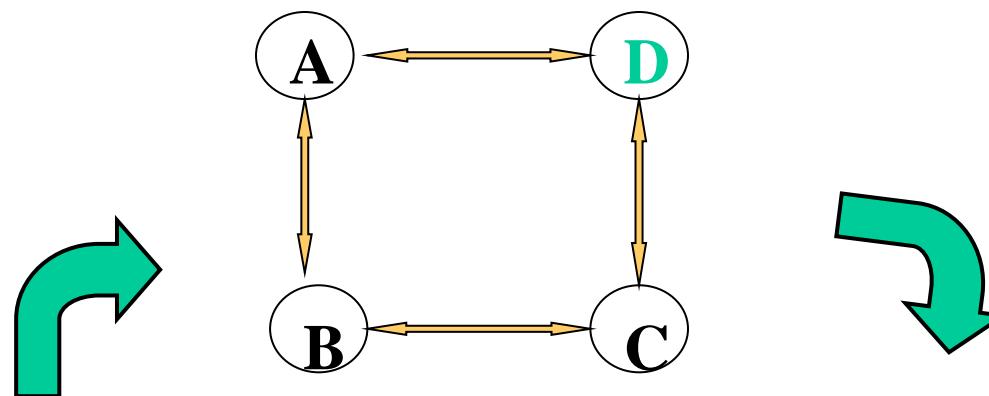


如果在状态A和C之间增加过渡状态D，将 $A \rightarrow C$ 改为 $A \rightarrow D \rightarrow C$ , $C \rightarrow A$ 改为 $C \rightarrow D \rightarrow A$ ,即如右图所示:





在状态A和C之间增加过渡状态D后，可将流程表修改成如右下表所示。修改后的流程表中增加了新的一行，但该行没有稳定状态，因为状态D仅在稳态A和C发生转换时完成过渡作用。



| 二次<br>状态<br><br>y | 激励状态 Y      |             |             |             |
|-------------------|-------------|-------------|-------------|-------------|
|                   | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| A                 | (A)<br>A    | B           | C           | (A)<br>C    |
| B                 |             | (B)<br>B    | (B)<br>C    |             |
| C                 | A           |             | (C)         | (C)         |

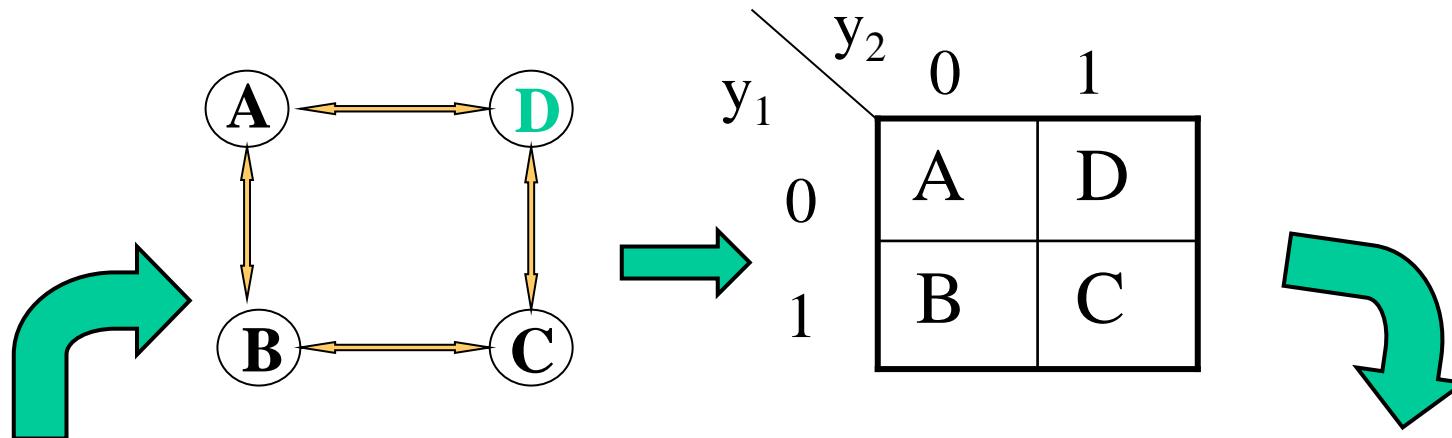
增加过渡状态后的流程表

| 二次状态<br><br>y | 激励状态 Y      |             |             |             |
|---------------|-------------|-------------|-------------|-------------|
|               | $x_2x_1=00$ | $x_2x_1=01$ | $x_2x_1=11$ | $x_2x_1=10$ |
| A             | (A)         |             | D           | (A)         |
| B             | A           | (B)         | (B)         | (B)         |
| C             | D           | B           | (C)         | (C)         |
| D             | A           | d           | C           | d           |





设二次状态用 $y_2$ 、 $y_1$ 表示，相应激励状态用 $Y_2$ 、 $Y_1$ 表示，令 $y_2y_1$ 取值  
**00表示A，01表示B，10表示D，11表示C**，即可得到相应二进制流程表  
 如右下表所示，该流程表描述的电路中不存在竞争。



增加过渡状态后的流程表

| 二次状态<br>$y$ | 激励状态 Y   |          |          |          |
|-------------|----------|----------|----------|----------|
|             | $xxx=00$ | $xxx=01$ | $xxx=11$ | $xxx=10$ |
| A           | (A)      | B        | D        | (A)      |
| B           | A        | (B)      | (D)      | C        |
| C           | D        | B        | (C)      | (C)      |
| D           | A        | d        | C        | d        |

二进制流程表

| 二次状态<br>$y_2\ y_1$ | 激励状态 $Y_2Y_1$ |          |          |          |
|--------------------|---------------|----------|----------|----------|
|                    | $xxx=00$      | $xxx=01$ | $xxx=11$ | $xxx=10$ |
| 0 0                | (0)           | 01       | 10       | (0)      |
| 0 1                | 00            | (1)      | (1)      | 11       |
| 1 1                | 10            | 01       | (1)      | (1)      |
| 1 0                | 00            | d        | 11       | d        |





### 3. 允许非临界竞争，避免临界竞争

对于有的流程表，虽然无法用最少位数的代码实现无竞争的状态分配，但可以通过将竞争限制在只有一个稳态的列，即允许非临界竞争，从而实现无临界竞争的状态分配。

**例** 对如下流程表进行状态编码，得到二进制流程表。

**流程表**

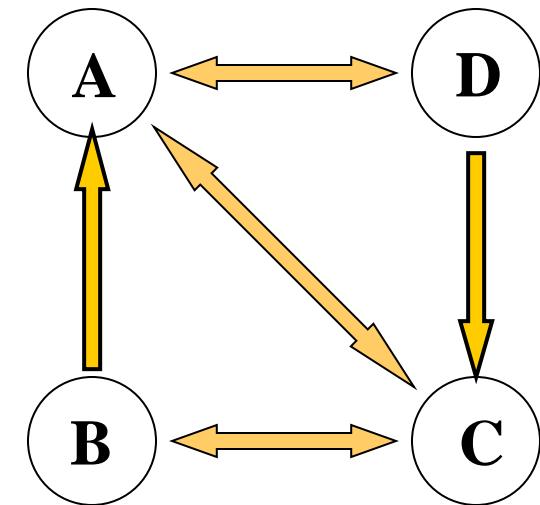
| 二次状态<br>Y | 激励状态 Y |        |        |        | 输出<br>Z |
|-----------|--------|--------|--------|--------|---------|
|           | xxm=00 | xxm=01 | xxm=11 | xxm=10 |         |
| A         | (A)    | C      | D      | (A)    | 0       |
| B         | A      | C      | (B)    | (B)    | 0       |
| C         | A      | (C)    | (C)    | B      | 0       |
| D         | A      | C      | (D)    | A      | 1       |





解 给定流程表的状态相邻图如右下图所示。

| 流程表       |                                   | 激励状态 Y                            |                                   |                                   |   | 输出<br>Z |
|-----------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|---|---------|
| 二次状态<br>Y | x <sub>2</sub> x <sub>1</sub> =00 | x <sub>2</sub> x <sub>1</sub> =01 | x <sub>2</sub> x <sub>1</sub> =11 | x <sub>2</sub> x <sub>1</sub> =10 |   |         |
| A         | (A)                               | C                                 | D                                 | (A)                               | 0 |         |
| B         | (A)                               | C                                 | (B)                               | (B)                               | 0 |         |
| C         | A                                 | (C)                               | (C)                               | B                                 | 0 |         |
| D         | A                                 | C                                 | (D)                               | A                                 | 1 |         |



显然，用两位二进制代码无法实现相邻状态相邻分配。但在该流程表中，状态A和C之间的转换只发生在 $x_2x_1=00$ 和 $x_2x_1=01$ 这两列，而这两列各只有一个稳定状态，这就意味着A和C发生转换时，即使产生竞争也属于非临界竞争。即在状态分配时A和C可以不相邻。排除A和C的相邻关系后，状态编码只需满足A和B、A和D、D和C、C和B相邻即可。



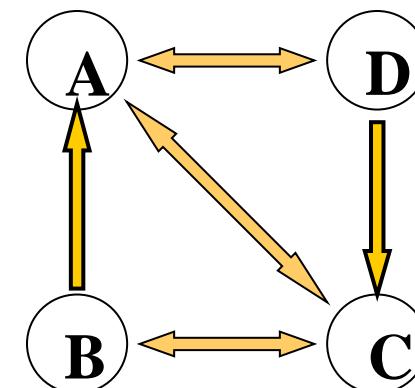


设二次状态用 $y_2$ 、 $y_1$ 表示，令 $y_2$ 、 $y_1$ 取值00表示A，01表示B，10表示D，11表示C，将其代入给定流程表，即可得到相应二进制流程表。该流程表描述的电路不会产生临界竞争。

$$y_2y_1 = 00, 01, 10, 11$$

↓    ↓    ↓    ↓

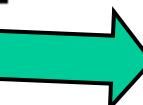
A    B    D    C



二进制流程表

流程表

| 二次状态<br>Y | 激励状态 Y |       |       |       | 输出<br>Z |
|-----------|--------|-------|-------|-------|---------|
|           | xx=00  | xx=01 | xx=11 | xx=10 |         |
| A         | Ⓐ      | C     | D     | Ⓐ     | 0       |
| B         | Ⓐ      | C     | Ⓑ     | Ⓑ     | 0       |
| C         | Ⓐ      | Ⓒ     | Ⓓ     | Ⓑ     | 0       |
| D         | Ⓐ      | C     | Ⓓ     | Ⓐ     | 1       |



| 二次状态<br>y <sub>2</sub> y <sub>1</sub> | 激励状态 Y <sub>2</sub> Y <sub>1</sub> |       |       |       | 输出<br>Z |
|---------------------------------------|------------------------------------|-------|-------|-------|---------|
|                                       | xx=00                              | xx=01 | xx=11 | xx=10 |         |
| 0 0                                   | Ⓐ                                  | 11    | 10    | Ⓑ     | 0       |
| 0 1                                   | 00                                 | 11    | Ⓑ     | Ⓑ     | 0       |
| 1 1                                   | 00                                 | Ⓑ     | Ⓑ     | 01    | 0       |
| 1 0                                   | 00                                 | 11    | Ⓓ     | 00    | 1       |





## 四、确定激励函数和输出函数

根据流程表可作出激励状态、输出函数的卡诺图，化简后即可得到激励状态和输出函数的最简表达式。

**例** 求如下流程表的激励函数和输出函数最简表达式。

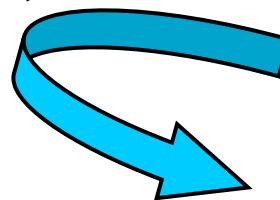
**二进制流程表**

| 二次状态  |       | 激励状态 $Y_2 Y_1$ |           |           |           | 输出<br>$Z$ |
|-------|-------|----------------|-----------|-----------|-----------|-----------|
| $Y_2$ | $Y_1$ | $xx_x=00$      | $xx_x=01$ | $xx_x=11$ | $xx_x=10$ |           |
| 0     | 0     | 00             | 11        | 10        | 01        | 0         |
| 0     | 1     | 00             | 11        | 01        | 01        | 0         |
| 1     | 1     | 00             | 01        | 01        | 01        | 0         |
| 1     | 0     | 00             | 11        | 10        | 00        | 1         |





根据二进制流程表可作出 $Y_2$ 、 $Y_1$ 和输出Z的卡诺图如下图所示。



二进制流程表

| 二次状态<br>$y_2 \ y_1$ | 激励状态 $y_2y_1$ |           |           |           | 输出<br>Z |
|---------------------|---------------|-----------|-----------|-----------|---------|
|                     | $xx_1=00$     | $xx_1=01$ | $xx_1=11$ | $xx_1=10$ |         |
| 0 0                 | 00            | 11        | 10        | 00        | 0       |
| 0 1                 | 00            | 11        | 01        | 01        | 0       |
| 1 1                 | 00            | 11        | 01        | 01        | 0       |
| 1 0                 | 00            | 11        | 00        | 00        | 1       |

| $x_2x_1$ | 00 | 01 | 11 | 10 |
|----------|----|----|----|----|
| $y_2y_1$ | 00 | 1  | 1  |    |
|          | 01 | 1  |    |    |
|          | 11 | 1  | 1  |    |
|          | 10 | 1  | 1  |    |

| $x_2x_1$ | 00 | 01 | 11 | 10 |
|----------|----|----|----|----|
| $y_2y_1$ | 00 | 1  |    |    |
|          | 01 | 1  | 1  | 1  |
|          | 11 | 1  | 1  | 1  |
|          | 10 | 1  |    |    |

| $x_2x_1$ | 00 | 01 | 11 | 10 |
|----------|----|----|----|----|
| $y_2y_1$ | 00 |    |    |    |
|          | 01 |    |    |    |
|          | 11 |    |    |    |
|          | 10 | 1  | 1  | 1  |

激励函数和输出函数表达式为：

$$Y_2 = \overline{x_2}x_1 + x_1y_2 + x_1\overline{y}_1; \quad Y_1 = \overline{x_2}x_1 + x_2y_1; \quad Z = y_2\overline{y}_1$$





**习题6-2：**对下表所示的最简流程表进行无临界竞争的状态编码，并确定激励状态和输出函数表达式。

表 6.15 最简流程表

| 二次状态 | 激励状态/输出 Z      |                |                |                |
|------|----------------|----------------|----------------|----------------|
|      | $x_2 x_1 = 00$ | $x_2 x_1 = 01$ | $x_2 x_1 = 11$ | $x_2 x_1 = 10$ |
| A    | Ⓐ/0            | Ⓐ/0            | Ⓐ/0            | C/0            |
| B    | Ⓑ/0            | A/0            | C/d            | Ⓑ/0            |
| C    | B/0            | A/d            | Ⓒ/1            | Ⓒ/0            |



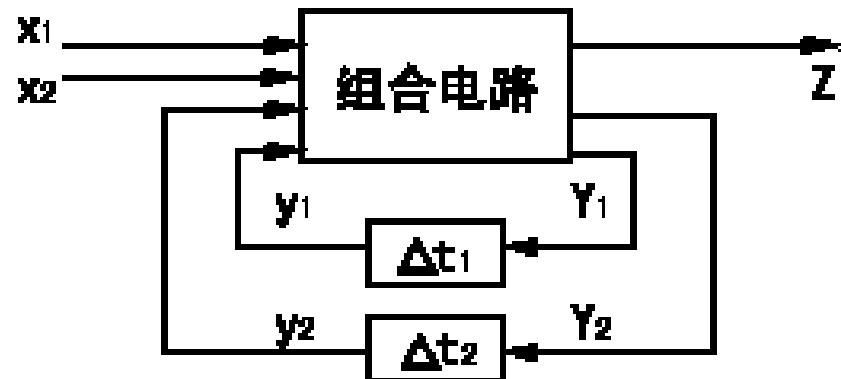


**习题6-3：**某电平异步时序逻辑电路的结构框图如下图所示，图中：

$$Y_2 = x_2y_2 + \bar{x}_1y_2 + x_2\bar{x}_1y_1$$

$$Y_1 = x_2x_1 + \bar{x}_2\bar{x}_1y_2 + x_1y_2\bar{y}_1$$

$$Z = y_2y_1$$



某电平异步时序电路框图

试判断该电路中是否存在竞争？若存在，请指出在什么情况下会发生竞争，属于何种类型的竞争。





根据激励函数和输出函数表达式可画出逻辑电路图。  
(略)

