

# 第五章 同步时序逻辑电路

## ● 时序逻辑电路

➤ 输出状态不仅和当时的输入状态有关，而且与过去的输入状态有关

➤ 特点：包含触发器（有反馈线）

⊕ 时序电路的结构特点：

※1、包含组合逻辑电路和存储电路两部分，存储器必不可少

※2、存储电路的状态必须反馈回输入端，与输入信号一起共同决定组合电路的输出

## 时序逻辑电路

### ➤ 分析

- ⊕ 对已有的时序逻辑电路研究其逻辑功能和工作特性

### ➤ 设计

- ⊕ 对已确定的功能要求，作出实现其功能的时序逻辑电路

### ➤ 常用时序逻辑功能器件

- ⊕ 计数器、寄存器

# 内容提要



山东大学

计算机科学与技术学院



## 5.1 时序逻辑电路概述



## 5.2 同步时序逻辑电路分析



## 5.3 同步时序逻辑电路设计概述



## 5.4 建立状态表



## 5.5 状态化简



## 5.6 状态分配和网络实现



# 5.1 时序逻辑电路概述



山东大学

计算机科学与技术学院

## ● 时序逻辑电路

- $X=(x_1, x_2, \dots, x_n)$ : 网络输入
- $Z=(z_1, z_2, \dots, z_m)$ : 网络输出
- $W=(w_1, w_2, \dots, w_p)$ : 网络激励, 即各触发器的输入变量
- $y=(y_1, y_2, \dots, y_q)$ : 网络状态, 即各触发器的输出变量

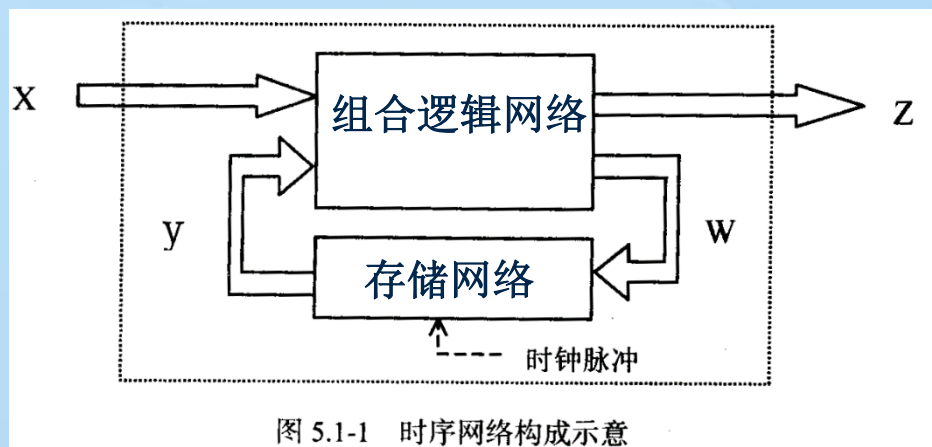


图 5.1-1 时序网络构成示意

输出方程 $Z=Z(y, X)$

激励方程 $W=W(y, X)$

组合函数

状态方程 $y'=F(y, W)=f(y, X)$ 描述时序电路的存储特性

- 分类方法：是否使用同一个时钟脉冲

- 同步时序网络

- ⊕ 有统一的时钟脉冲，各触发器同时变化

- ⊕ 工作速度快，可靠性高

- (即存储电路中所有存储单元状态的变化都是在同一个控制脉冲的操作下同时发生的)

- 异步时序网络

- ⊕ 无统一的时钟脉冲，各触发器不同时变化

- ⊕ 工作速度和可靠性稍差

# 时序逻辑电路的分类



山东大学

计算机科学与技术学院

- 分类方法：输入 $x$ 是否直接影响输出 $Z$   
( $y$ 为触发器电路输出)

- 米里型Mealy

- ⊕ 输入直接影响输出 $Z=Z(y, x)$

- 摩尔型Moore

- ⊕ 输入不直接影响输出 $Z=Z(y)$

不同类型的电路，其工作特性及分析、设计方法均有所不同。

# 5.2 同步时序逻辑电路分析



山东大学

计算机科学与技术学院

## ● 1. 分析步骤

### (1) 写组合函数

- ⊕ 网络的输出方程和激励方程

### (2) 写状态方程

- ⊕ 把激励方程代入到所用触发器的特征方程，得到网络的次态和现态及网络输入的函数表达式

### (3) 计算状态表

- ⊕ 由状态方程计算各状态在不同输入作用下的次态

### (4) 作状态图

- ⊕ 将状态表所反映的状态转换关系用图形直观形象的表示出来

### (5) 特性讨论

- ⊕ 对网络功能特点作出某些必要的说明，还可以作出时序图



# 分析举例



山东大学

计算机科学与技术学院

- 例5. 2-1分析右图的功能特性，并作出网络初态为 $y_1y_0=11$ ，输入时序为 $X=00001111$ 时的时序图

## ① 写组合函数

输出方程  $Z = \overline{y_1y_0}$

激励方程  $J_0 = 1 \quad J_1 = x \oplus y_0$

$K_0 = 1 \quad K_1 = x \oplus y_0$

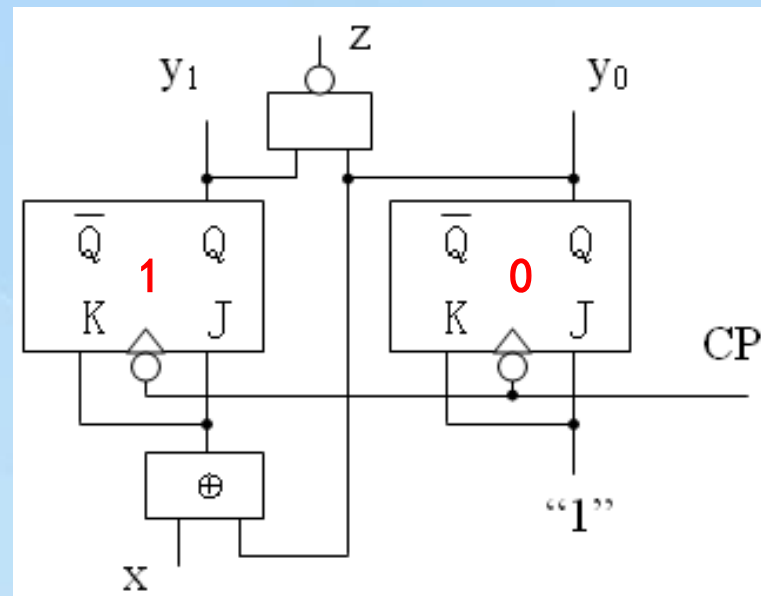
## ②写状态方程

$JK$ 触发器

$$Q' = J\bar{Q} + \bar{K}Q$$

$$y'_0 = J_0\bar{y}_0 + \bar{K}_0y_0 = 1\bar{y}_0 + 0y_0 = \bar{y}_0$$

$$y'_1 = J_1\bar{y}_1 + \bar{K}_1y_1 = X \oplus y_0 \oplus y_1$$



### ③状态表(卡诺图形式)

$$y'_0 = J_0 \bar{y}_0 + \bar{K}_0 y_0 = 1 \bar{y}_0 + 0 y_0 = \bar{y}_0$$

$$y'_1 = J_1 \bar{y}_1 + \bar{K}_1 y_1 = X \oplus y_0 \oplus y_1$$

$$Z = \overline{y_1 y_0}$$

状态转换表

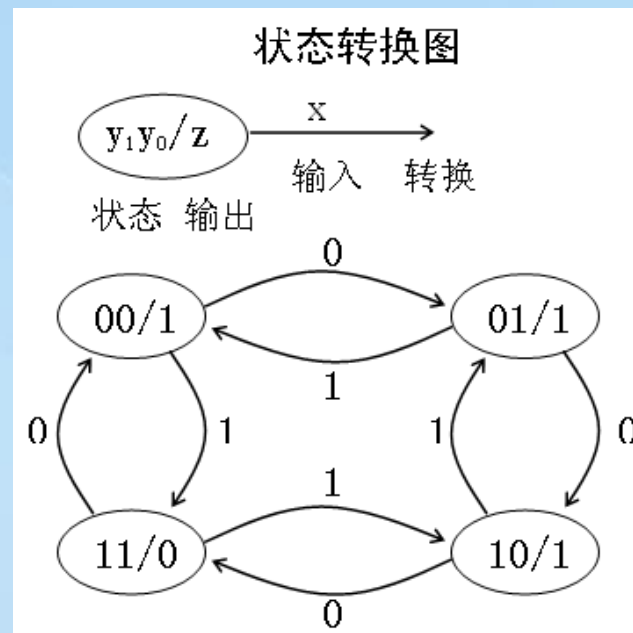
| $y_1 y_0$ | $X$ |     | $Z$ |
|-----------|-----|-----|-----|
|           | 0   | 1   |     |
| 00        | 0 1 | 1 1 | 1   |
| 01        | 1 0 | 0 0 | 1   |
| 11        | 0 0 | 1 0 | 0   |
| 10        | 1 1 | 0 1 | 1   |

$y_1' y_0'$

### ③状态表(卡诺图形式)

| 状态转换表     |             |     |     |  |
|-----------|-------------|-----|-----|--|
| $y_1 y_0$ | $x$         |     | $z$ |  |
|           | 0           | 1   |     |  |
| 00        | 0 1         | 1 1 | 1   |  |
| 01        | 1 0         | 0 0 | 1   |  |
| 11        | 0 0         | 1 0 | 0   |  |
| 10        | 1 1         | 0 1 | 1   |  |
|           | $y_1' y_0'$ |     |     |  |

### ④状态图



### ⑤讨论

状态以四为模构成循环

状态按二进制加1或减1规律变化，这种电路称为计数器，能计数CP脉冲中的个数

当 $x=0$ 时，按照加1规则计数，

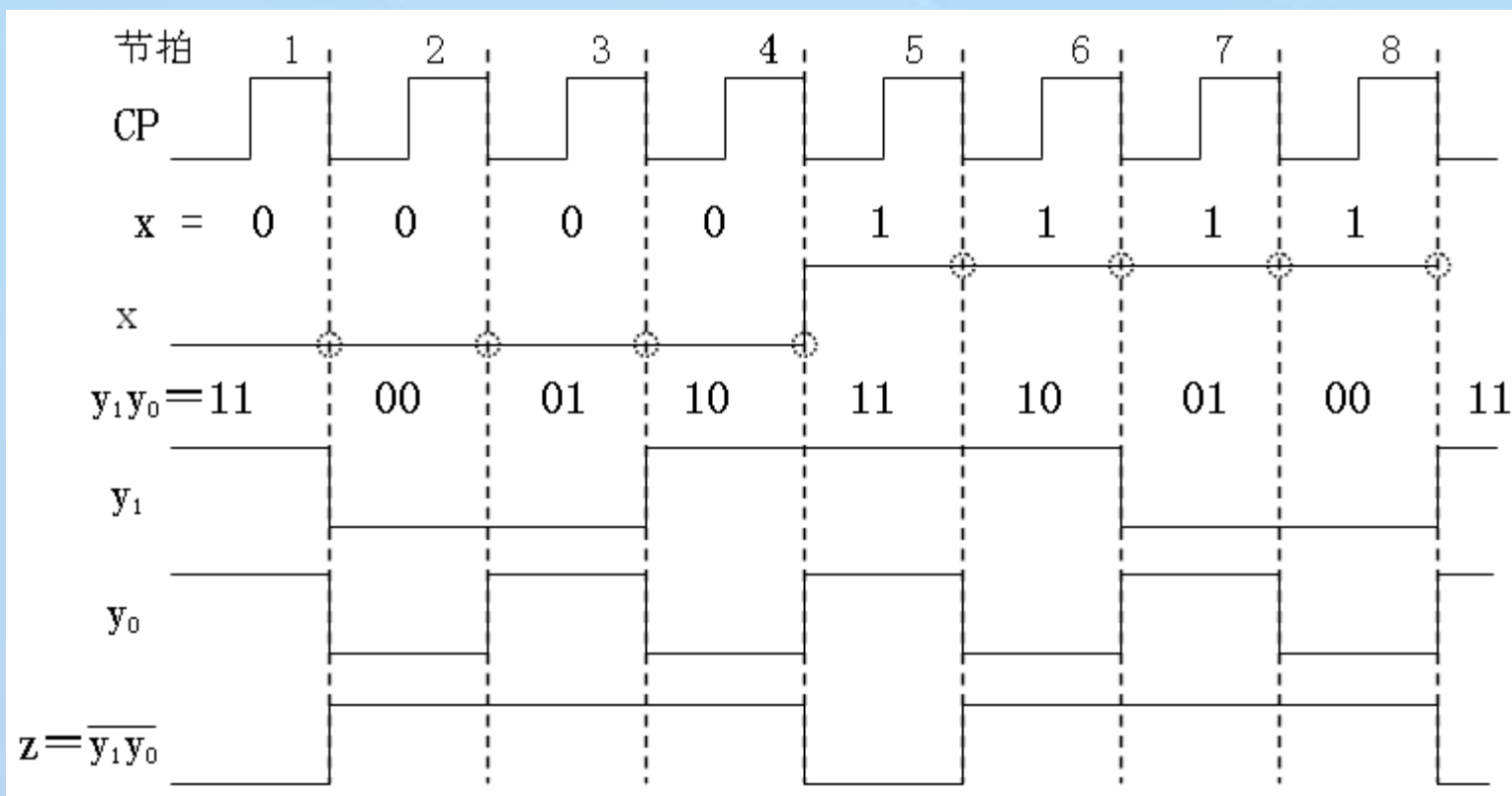
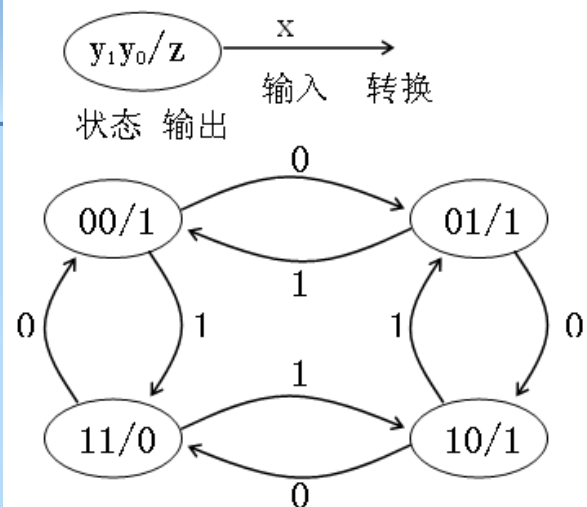
当 $x=1$ 时，按照减1规则计数。

## ⑥波形图(时序图)

作出网络初态为 $y_1y_0=11$ , 输入时序为 $X=00001111$ 时的时序图

说明: 使用负沿触发器, 以CP下降沿划分节拍  
CP下降沿时若输入变化, 其变化前的值有效

状态转换图



# 计数器



山东大学

计算机科学与技术学院

- 计数器是能对脉冲个数进行计数的电路
- 按计数的模分类，计数器可以分为：
  - $r$ 位二进制计数器：以 $2^r$ 为模
  - $N$ 进制计数器：以 $N$ 为模
- 按计数规则分类，计数器可以分为：
  - 加1计数器
  - 减1计数器
- 按时钟分类，计数器可以分为：
  - 同步计数器：各记忆部件受统一时钟信号控制
  - 异步计数器：无统一时钟信号

计数器可以没有外部输入，只对CP脉冲进行计数，也可以直接以电路状态作为输入。

# 例5. 2. 2分析电路—只对CP脉冲进行计数



山东大学

计算机科学与技术学院

图5. 2-4

$$Q' = J\bar{Q} + \bar{K}Q$$

## ①组合函数

激励方程

$$D_0 = \bar{y}_0$$

$$J_1 = \bar{y}_2 \cdot y_0$$

$$J_2 = y_0 \cdot y_1$$

$$K_1 = y_0$$

$$K_2 = y_0$$

## ②状态方程

触发器0

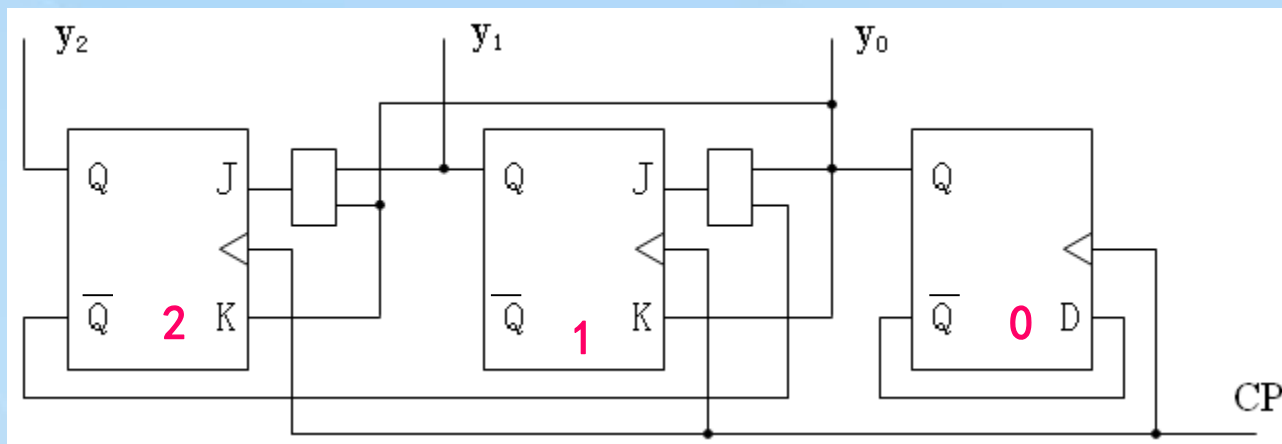
$$y_0' = D_0 = \bar{y}_0$$

触发器1

$$y_1' = J_1 \cdot \bar{y}_1 + \bar{K}_1 \cdot y_1 = \bar{y}_2 y_0 \bar{y}_1 + \bar{y}_0 y_1$$

触发器2

$$y_2' = J_2 \cdot \bar{y}_2 + \bar{K}_2 \cdot y_2 = y_0 y_1 \bar{y}_2 + \bar{y}_0 y_2$$



## 例5. 2. 2分析电路



山东大学

计算机科学与技术学院

### ③状态表

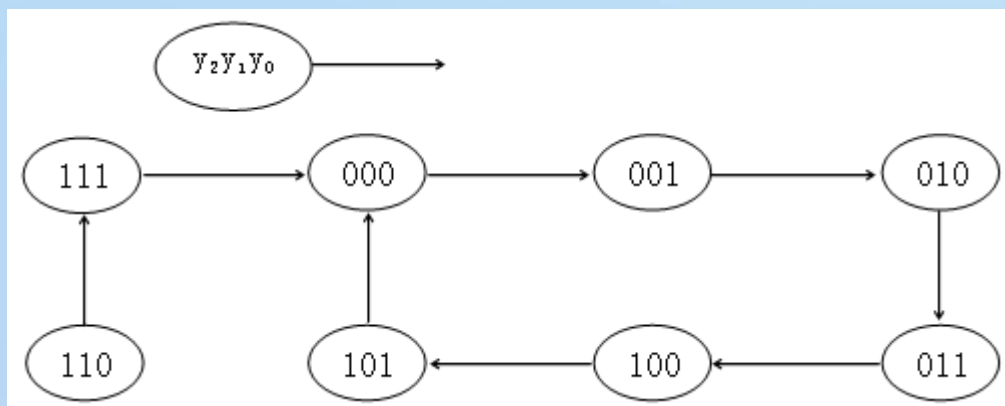
$$y_0' = D_0 = \overline{y_0}$$

$$y_1' = J_1 \cdot \overline{y_1} + \overline{K_1} \cdot y_1 = \overline{y_2} y_0 \overline{y_1} + \overline{y_0} y_1$$

$$y_2' = J_2 \cdot \overline{y_2} + \overline{K_2} \cdot y_2 = \overline{y_2} y_1 y_0 + y_2 \overline{y_0}$$

| $y_2 y_1 y_0$ | $y_2' y_1' y_0'$ |
|---------------|------------------|
| 000           | 001              |
| 001           | 010              |
| 010           | 011              |
| 011           | 100              |
| 100           | 101              |
| 101           | 000              |
| 110           | 111              |
| 111           | 000              |

### ④状态图



## 例5. 2. 2分析电路



山东大学

计算机科学与技术学院

### ⑤特性讨论

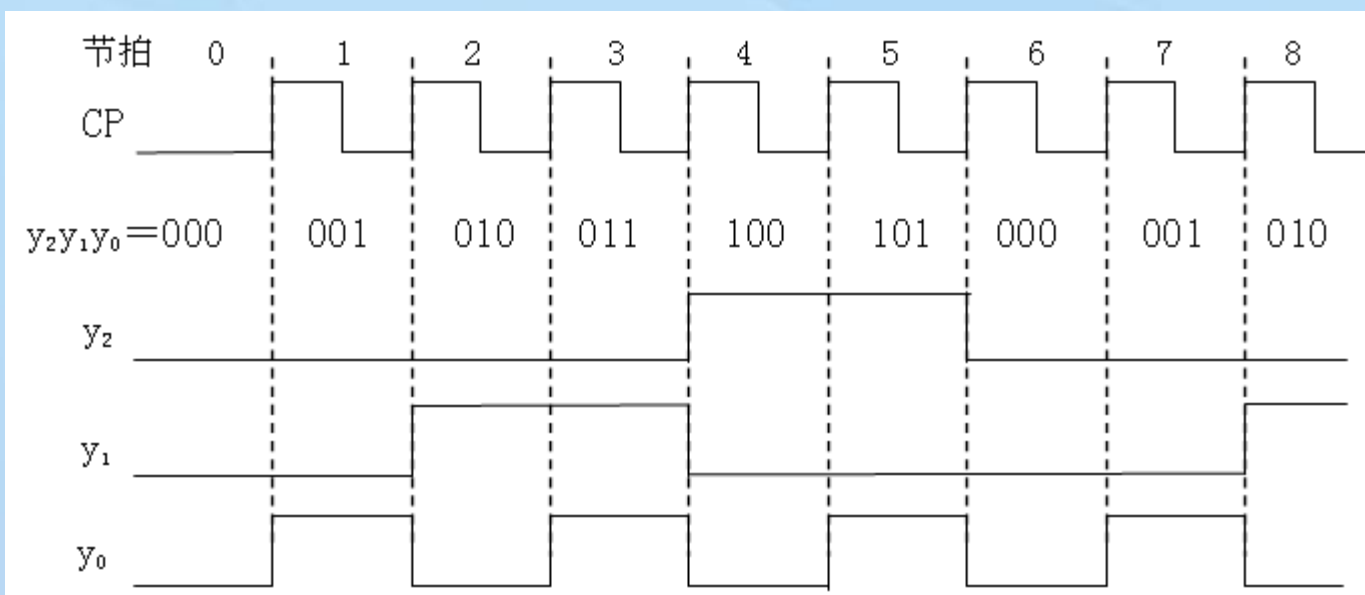
状态转换以6为模，按加1规则变化  
为同步六进制加法计数器

- 有效状态 正常工作时经历的状态  
如000, 001, , . . . , 101
- 无效状态 正常工作不经历的状态  
如110, 111
- 自启动 能由无效状态返回有效状态
- 不能自启动的电路工作不可靠



## ⑥时序图

正沿触发器，其状态在CP上升沿时变化。



# 分析举例

例5. 2-4分析右图的功能特性，  
并作出网络初态为 $y_1y_0=00$ ，  
输入时序为 $X=00011001$ 时的  
时序图

解： 同步时序，米里型(输入直接影响输出)

## ① 写组合函数

$$\text{输出方程 } Z = \overline{\bar{x} + y_1 + \bar{y}_0} = x \cdot \bar{y}_1 \cdot y_0$$

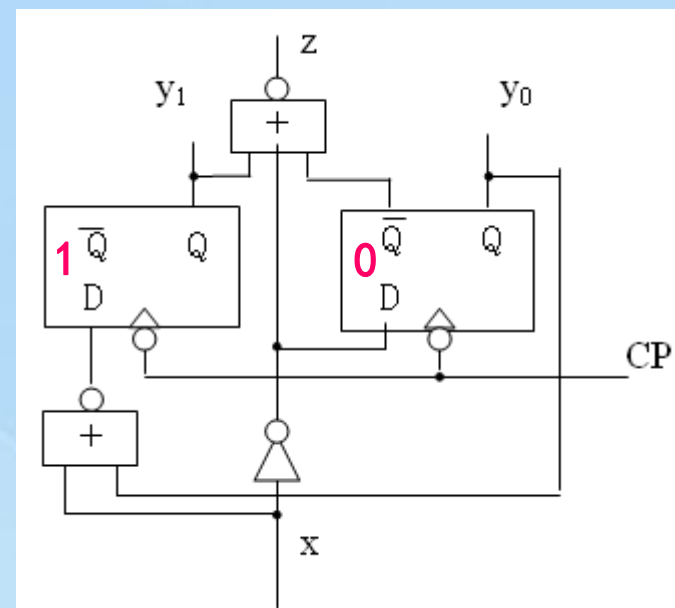
激励方程

$$D_0 = \bar{x} \quad D_1 = \overline{x + y_0} = \bar{x} \cdot \bar{y}_0$$

## ②写状态方程

$$Q' = D \quad y'_0 = D_0 = \bar{x}$$

$$y'_1 = D_1 = \bar{x} \cdot \bar{y}_0$$



## 例5. 2. 4分析电路

### ③状态转换表

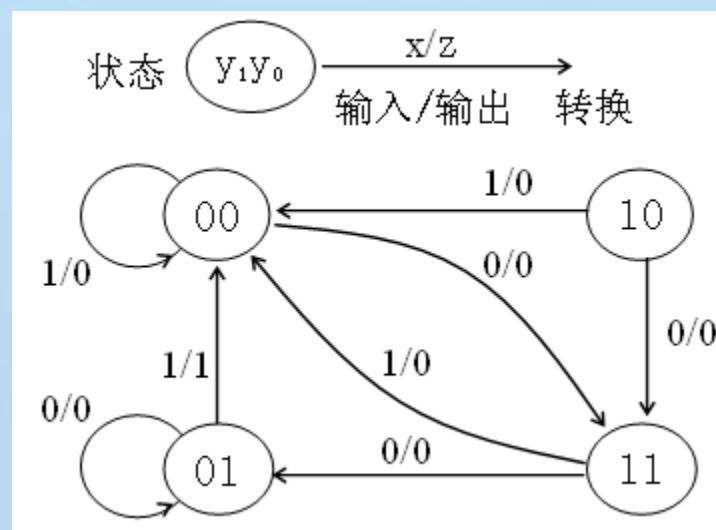
$$y'_0 = \bar{x}$$

$$y'_1 = \bar{x} \cdot \bar{y}_0$$

$$Z = x \cdot \bar{y}_1 \cdot y_0$$

| $y_1y_0$ \ $x$ | 0            | 1    |
|----------------|--------------|------|
|                | $y_1'y_0'/z$ |      |
| 00             | 11/0         | 00/0 |
| 01             | 01/0         | 00/1 |
| 11             | 01/0         | 00/0 |
| 10             | 11/0         | 00/0 |

### ④状态转换图



## 例5. 2. 4分析电路



山东大学

计算机科学与技术学院

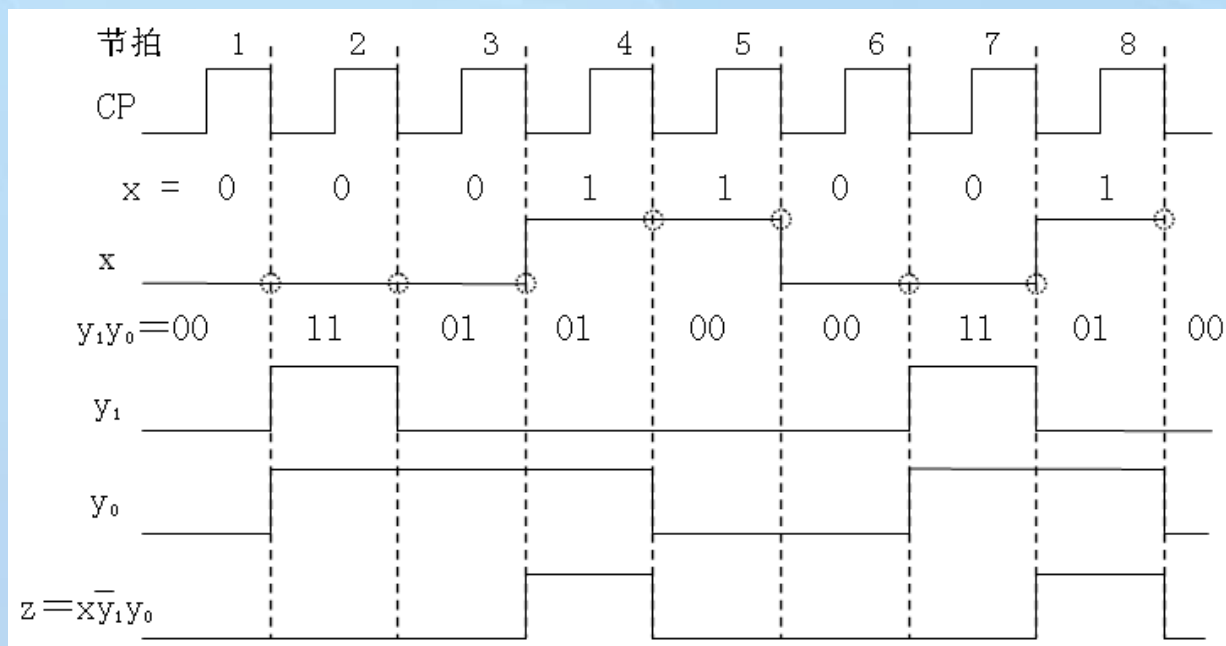
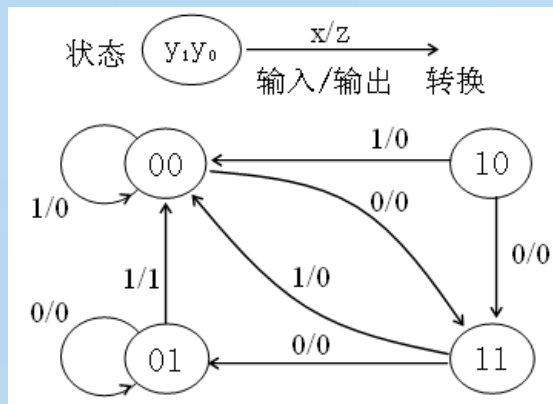
### ● ⑤特性讨论

3个有效状态，1个无效状态，能自启动

若初态为 $y_1y_0=00$ ，则仅当输入 $x$ 出现“001”序列时，输出才为1，可作为“001”序列监测器

### ⑥时序图

负沿触发器，以  
CP下降沿划分节拍



# 分析举例



山东大学

计算机科学与技术学院

## 例5. 2-5分析右图的功能特性

解：

### ① 写组合函数

输出方程  $Z = x \cdot y_1 \cdot y_0$

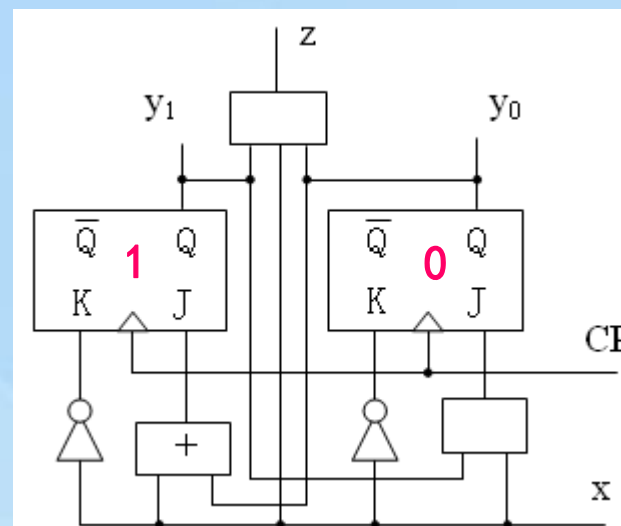
激励方程

$$J_0 = xy_1 \quad J_1 = x + y_0$$

$$K_0 = \bar{x} \quad K_1 = \bar{x}$$

### ②写状态方程

$$Q' = J\bar{Q} + \bar{K}Q \quad y'_0 = J_0\bar{y}_0 + \bar{K}_0y_0 = x \cdot y_1 \cdot \bar{y}_0 + \bar{x} \cdot y_0 = x \cdot y_1 + x \cdot y_0$$
$$y'_1 = J_1 \cdot \bar{y}_1 + \bar{K}_1 \cdot y_1 = (x + y_0)\bar{y}_1 + x \cdot y_1 = x + y_0 \cdot \bar{y}_1$$



## 例5. 2. 5分析电路



山东大学

计算机科学与技术学院

### ③状态转换表

$$y'_0 = x \cdot y_1 + x \cdot y_0$$

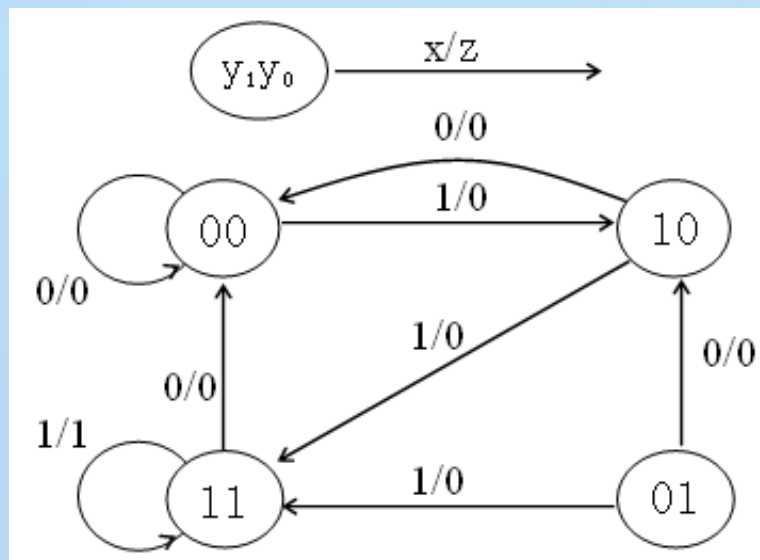
$$y'_1 = x + y_0 \cdot \bar{y}_1$$

$$Z = x \cdot y_1 \cdot y_0$$

| $y_1 y_0$ | $x$  |      |
|-----------|------|------|
|           | 0    | 1    |
| 00        | 00/0 | 10/0 |
| 01        | 10/0 | 11/0 |
| 11        | 00/0 | 11/1 |
| 10        | 00/0 | 11/0 |

$y_1' y_0' / z$

### ④状态转换图



## 例5. 2. 5分析电路



山东大学

计算机科学与技术学院

- ⑤特性讨论

3个有效状态，1个无效状态01，能自启动

仅当连续输入3个1时，输出才为1

可作为“111”序列监测器

## 5.2.3 寄存器

- 寄存器(register)是一种能够暂存二进制信息的器件。寄存器主要由触发器构成,一个触发器可以存放一位二进制信息,n个触发器可以构成能存放n位二进制信息的n位寄存器。

### (1) 基本寄存器

#### 电路构成

(右图)

#### 功能分析

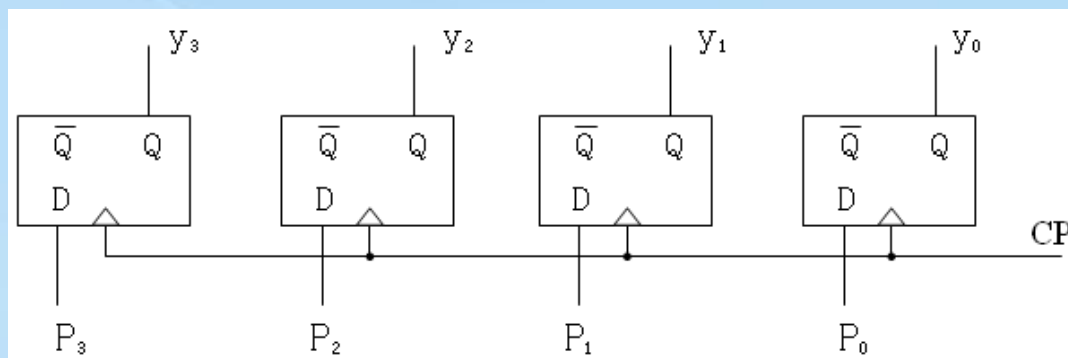
激励  $D_i = P_i$  其中  $P_i$  称为数据输入端

状态  $y'_i = D_i = P_i$ , CP上升沿

#### 工作情况

- 输入数据,即在CP脉冲作用下各  $P_i$  端数据送入相应触发器;
- 保存数据,即在无CP脉冲作用时各触发器的状态不变。

基本寄存器的数据是并行输入、并行输出的。





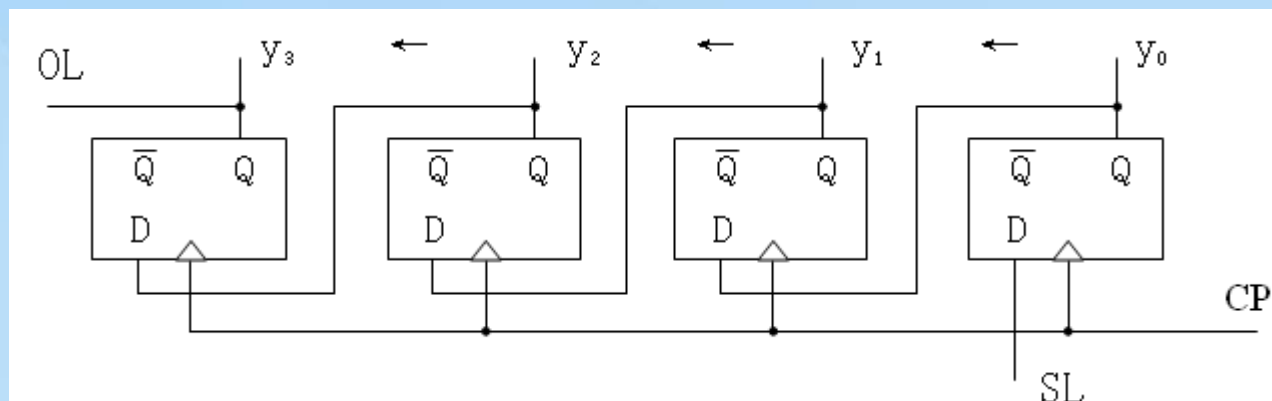
## (2) 移位寄存器



山东大学

计算机科学与技术学院

- 电路



- 功能分析

- $y_i' = D_i = y_{i-1}$  ( $i=3,2,1$ )
- $y_0' = D_0 = SL$  其中SL为外部数据输入端(串行输入)

- 工作情况

- 移位输入：在CP脉冲作用下，内部数据逐位移动，外部数据按位输入
  - 数据保持：无CP脉冲时，各触发器状态不变
- 通常称 由低向高位的移动为左移，由高向低位的移动为右移

- 串行输入/输出

各位数据由同一信号线分时传送，传送n位需要n个节拍

- 作用

- 串-并转换，算术移位等

### (3) 环形移位寄存器



山东大学

计算机科学与技术学院

- 电路

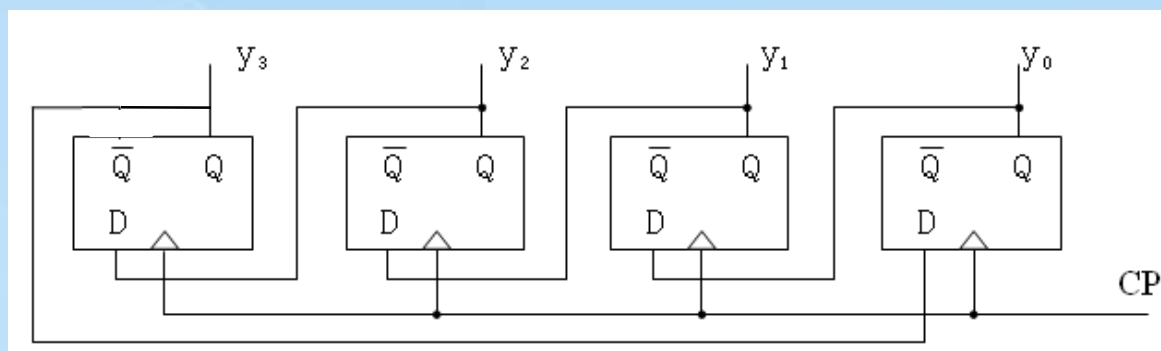
首位相连，无外部输入

- 功能分析

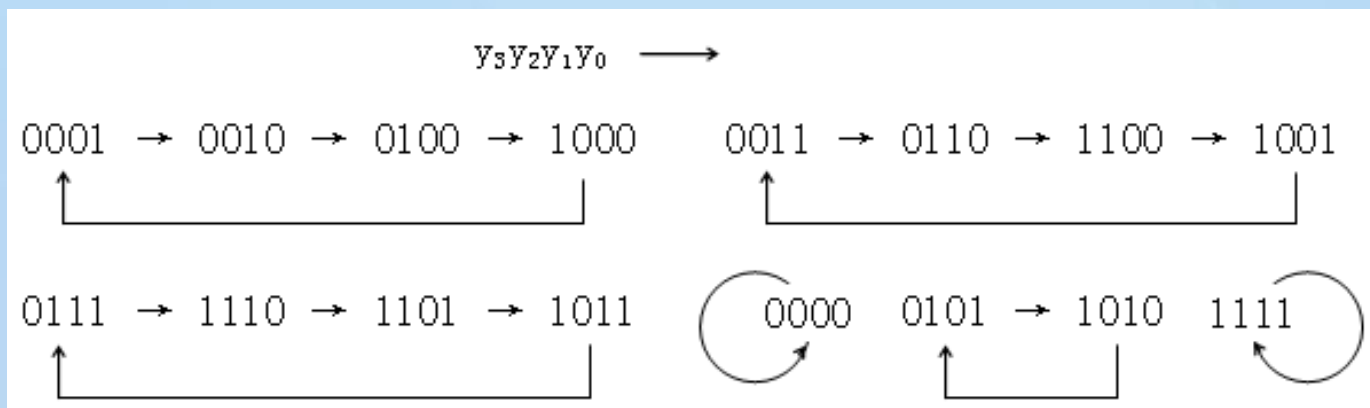
- $y_i' = D_i = y_{i-1} \ (i=3,2,1)$

- $y_0' = D_0 = y_3$

- 工作情况



- 状态转换图



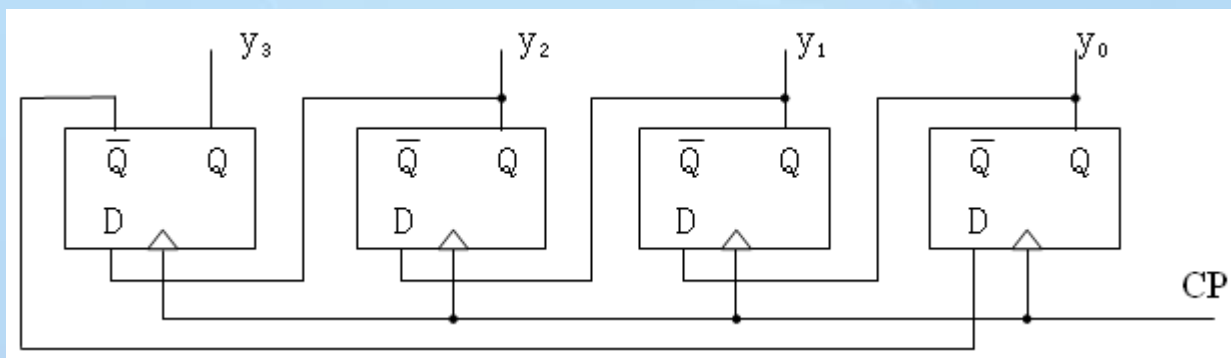
## (4) 扭环移位寄存器



山东大学

计算机科学与技术学院

- 若将移位寄存器的最高位触发器的输出反相后（实际是从 $\bar{Q}$ 输出）接到最低位触发器的输入端，即构成扭环计数器



# 作业



山东大学

计算机科学与技术学院

- 练习五

- 1, 4, 5

# 内容提要



山东大学

计算机科学与技术学院



## 5.1 时序逻辑电路概述



## 5.2 同步时序逻辑电路分析



## 5.3 同步时序逻辑电路设计概述



## 5.4 建立状态表



## 5.5 状态化简



## 5.6 状态分配和网络实现



## 5.3 同步时序逻辑电路设计概述



山东大学

计算机科学与技术学院

- 同步时序逻辑电路设计(又称为同步时序逻辑电路**综合**)

- 对给定功能要求, 做出电路实现

- 时序逻辑电路设计是时序逻辑电路分析的逆过程, 设计过程更复杂

- 设计步骤(4步)

- (1) 建立状态表: **确定状态及转换关系**

- 建立输入、输出以及状态之间的关系**

- (2) 状态化简: **消除多余状态**

- (3) 状态分配: **用二进制编码表示各状态**

- (4) 电路实现: **由赋值状态表确定各触发器的激励函数和电路输出, 做出电路图**

说明: 有的设计可能不需要经过全部步骤, 如计数器设计不需要化简、赋值, 可以直接写出赋值状态表。

下面分别介绍各部的具体做法。

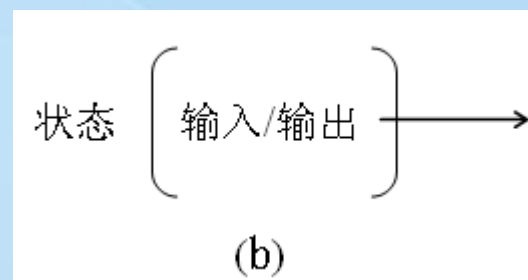
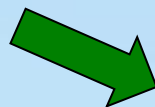
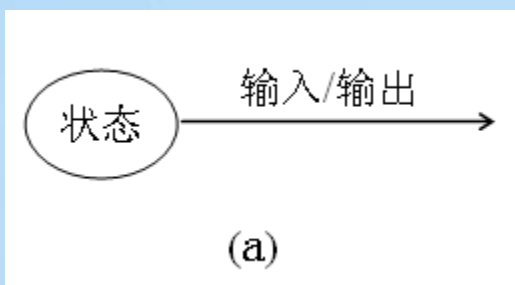
## 5.4 建立状态表

- 建立状态表就是对给定的设计要求，初步确定所需状态个数及其转换关系，写出描述给定功能要求的状态转换表。
- 两种基本方法
  - 信号序列法
    - ⊕ **信号序列**是指作用于网络的**输入信号序列**和相应的网络**输出信号序列**
    - ⊕ 信号序列法是先确定作用于网络的信号序列，再指定状态，作出状态转换表
  - 状态图法
    - ⊕ 先根据设计要求作出状态转换图，再由状态转换图写出状态转换表。
    - ⊕ 作状态转换图可以先设定一个初态，再根据所作用的输入信号和系统的功能要求逐步推导出相继状态及其转换关系，完成状态转换图。
  - 比较：状态图法中的各状态可以有特定的含义，比较容易理解和掌握

以下通过例题，介绍用状态图法建立状态转换表的具体做法。

## 5.4 建立状态表

- 状态图是用从状态引出的有向线段表示状态的转换关系，用标注在有向线段上的数码表示输入信号以及输出信号



- 这种表示方法在输入信号较多时，从状态引出的线段就比较多，看起来不够清晰

- 为此我们在建立状态表时采用图b所示的表示方法，即把作用于状态的全部输入信号以及相应的输出信号都列在状态右边的大括号中，括号右边引出的有向线段表示状态的转换关系。



## 5.4 建立状态表



### 同步时序逻辑电路作用1：进行信号检测

- 序列检测：序列的分组是任意的

- 例如：对信号 $X=101001100011$ 进行序列检测，则可有多种分组，其中有两组“100”序列

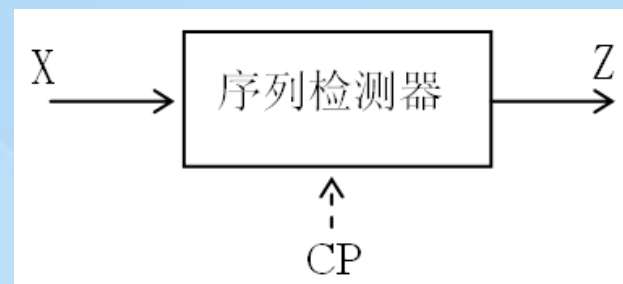
- 代码检测：代码的分组是固定的

- 例如：对信号 $X=101001100011$ 进行代码检测，则只有101，001，100，011四个分组，其中有一组“100”代码
- 代码检测器不仅要保存合乎要求的信号，还要“记住”代码的分组情况

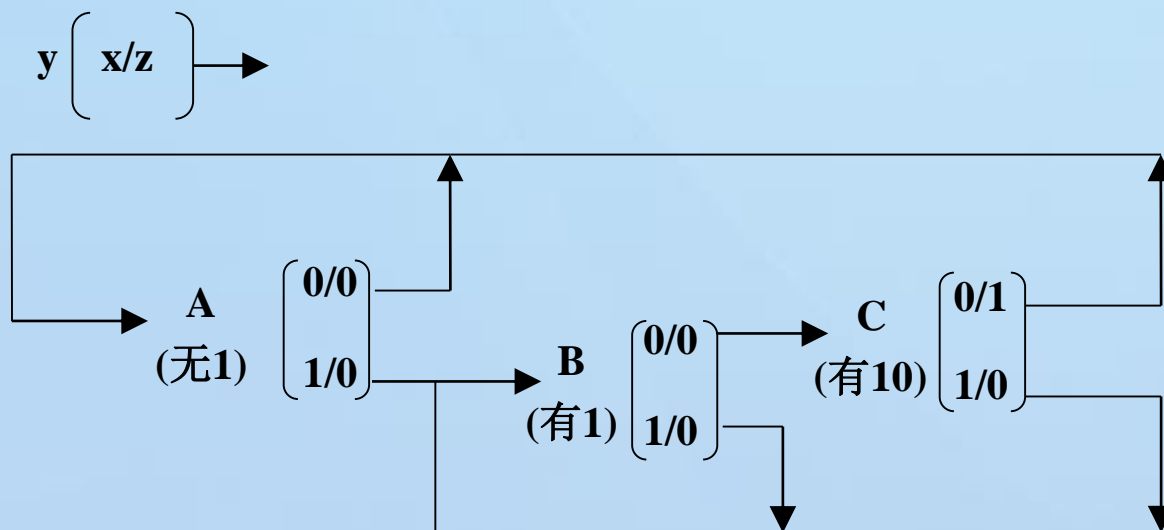
# 例1 建立“100”序列检测器的状态表

- 例5.4-1 建立“100”序列检测器的状态表。该检测器有一个串行输入信号X和一个输出信号Z(由CP脉冲划分节拍)。平时Z=0, 当输入信号X出现“100”序列时Z=1。

**分析：序列监测器需要记住前两拍的输入信号，因此该网络是具有存储功能的时序逻辑网络**



## 状态图



## 状态表

|   |   | x   |     |
|---|---|-----|-----|
|   |   | 0   | 1   |
| y | A | A/0 | B/0 |
|   | B | C/0 | B/0 |
|   | C | A/1 | B/0 |

y'/z

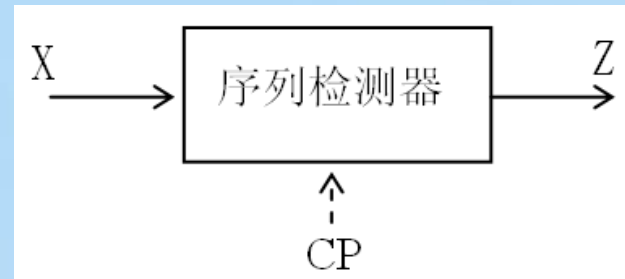
## 例2 建立“100”代码检测器的状态表



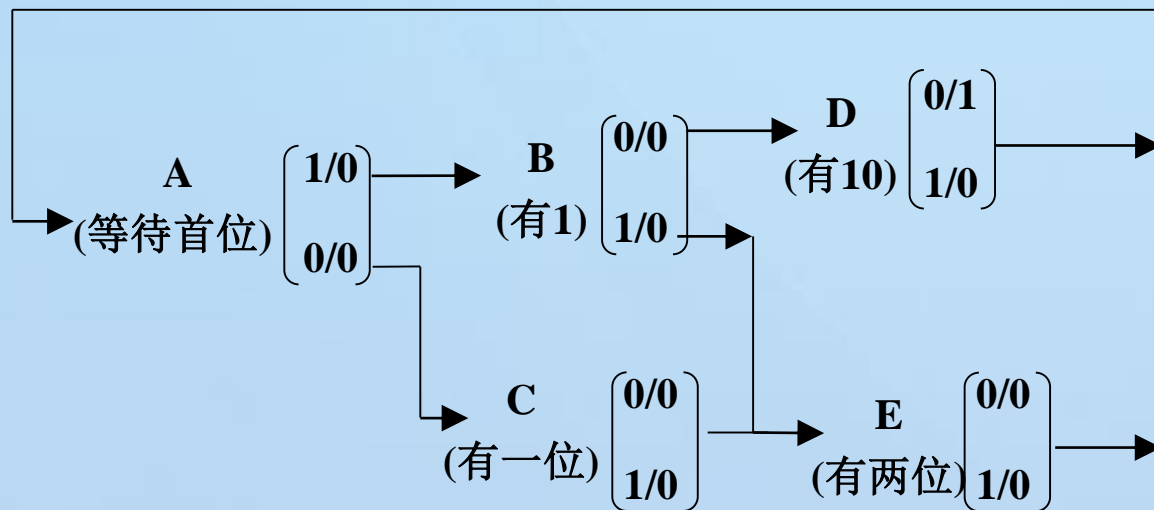
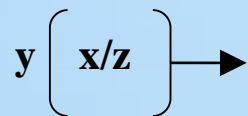
山东大学

计算机科学与技术学院

例5.4-2 建立“100”代码检测器的状态表。该检测器有一个串行输入端X和一个输出端Z(由CP脉冲划分节拍)，其功能是检测三位二进制代码是否是100。平时Z=0，当检测到“100”代码时 Z=1。平时Z=0，当检测到“100”代码时 Z=1。



状态图



状态表

|   |   | x   |     |      |  |
|---|---|-----|-----|------|--|
|   |   | 0   | 1   |      |  |
| y | A | C/0 | B/0 | y'/z |  |
|   | B | D/0 | E/0 |      |  |
|   | C | E/0 | E/0 |      |  |
|   | D | A/1 | A/0 |      |  |
|   | E | A/0 | A/0 |      |  |

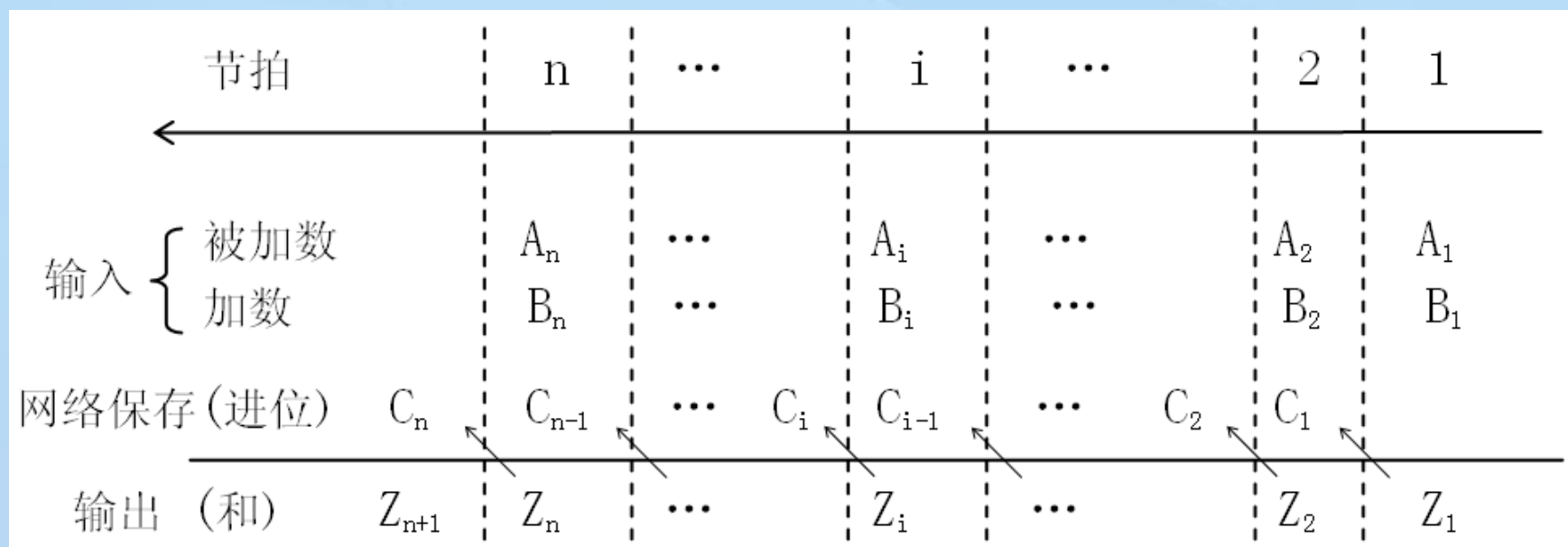
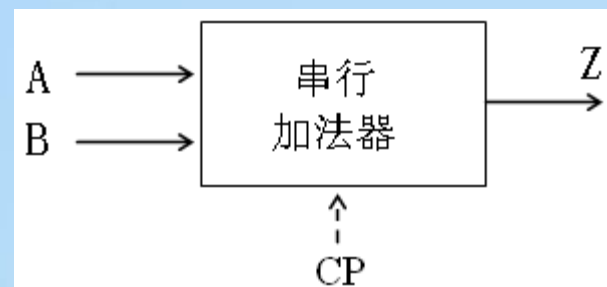
# 例3 建立串行加法器的状态表



山东大学

计算机科学与技术学院

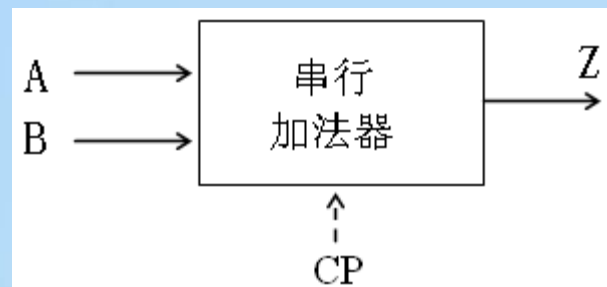
例5.4-3 建立串行加法器的状态表。串行加法器有两个串行信号输入端A, B和一个输出信号Z(由CP脉冲划分节拍)。A, B分别串行输入两个二进制数(先送低位), Z串行输出两数相加的“和”



# 例3 建立串行加法器的状态表

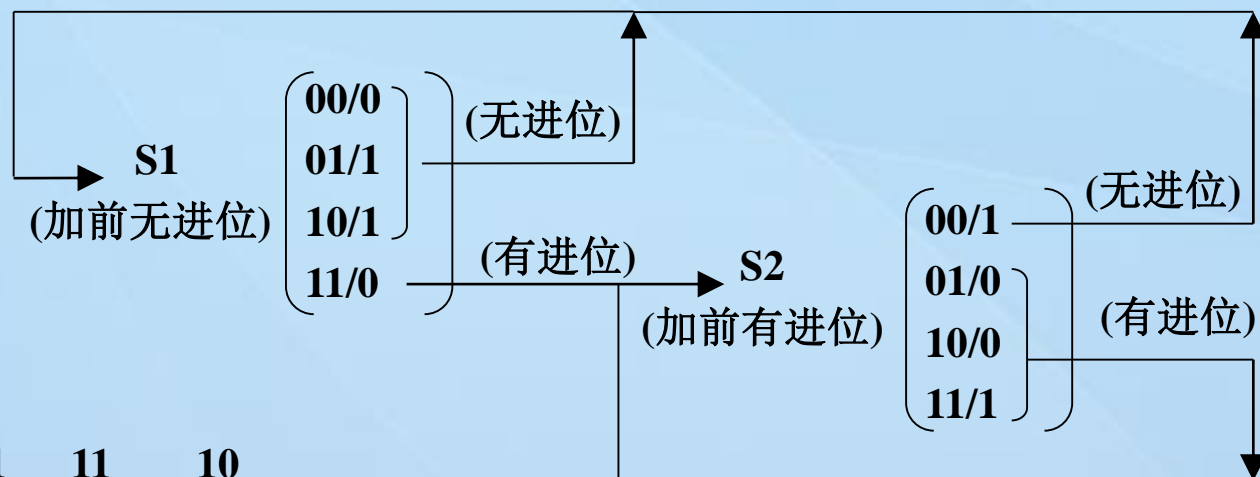


例5.4-3 建立串行加法器的状态表。串行加法器有两个串行信号输入端A, B和一个输出信号Z(由CP脉冲划分节拍)。A, B分别串行输入两个二进制数(先送低位), Z串行输出两数相加的“和”



状态图

$y \left[ \begin{matrix} A & B & Z \end{matrix} \right] \rightarrow$



状态表

| $y \backslash AB$ |    |      |      |      |      |
|-------------------|----|------|------|------|------|
|                   |    | 00   | 01   | 11   | 10   |
| y                 | S1 | S1/0 | S1/0 | S2/0 | S1/0 |
|                   | S2 | S1/0 | S2/0 | S2/1 | S2/0 |

$y' / Z$

## 例4 串行比较器



山东大学

计算机科学与技术学院

例5.4-4 建立一个串行数据比较器的状态表。串行数据比较器的功能是比较两个串行二进制数的大小。比较器有三个输入端 $x_3$ 、 $x_2$ 、 $x_1$ 和两个输出端 $z_2$ 、 $z_1$ （由CP脉冲划分节拍）。

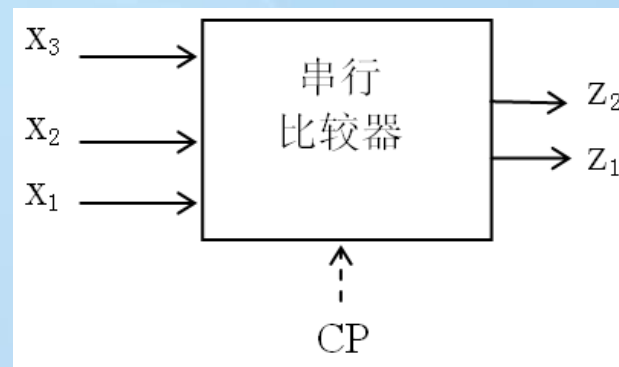
当 $x_3=0$ 时， $x_2$ 和 $x_1$ 分别串行输入两个二进制数（先送低位）；这时输出 $z_2z_1=00$ 。

当 $x_3=1$ 时， $x_2$ 和 $x_1$ 分别输入两个二进制数的最高位；这时 $z_2z_1$ 输出比较结果。  
即：

若 $x_2 < x_1$ ， 则 $z_2z_1=01$ ；

若 $x_2 > x_1$ ， 则 $z_2z_1=10$ ；

若 $x_2 = x_1$ ， 则 $z_2z_1=11$ ；



# 例4 串行比较器

状态图

当 $x_3=0$ 时， $x_2$ 和 $x_1$ 分别串行输入两个二进制数(先送低位)；这时输出 $z_2z_1=00$ 。

当 $x_3=1$ 时， $x_2$ 和 $x_1$ 分别输入两个二进制数的最高位；这时 $z_2z_1$ 输出比较结果。即：

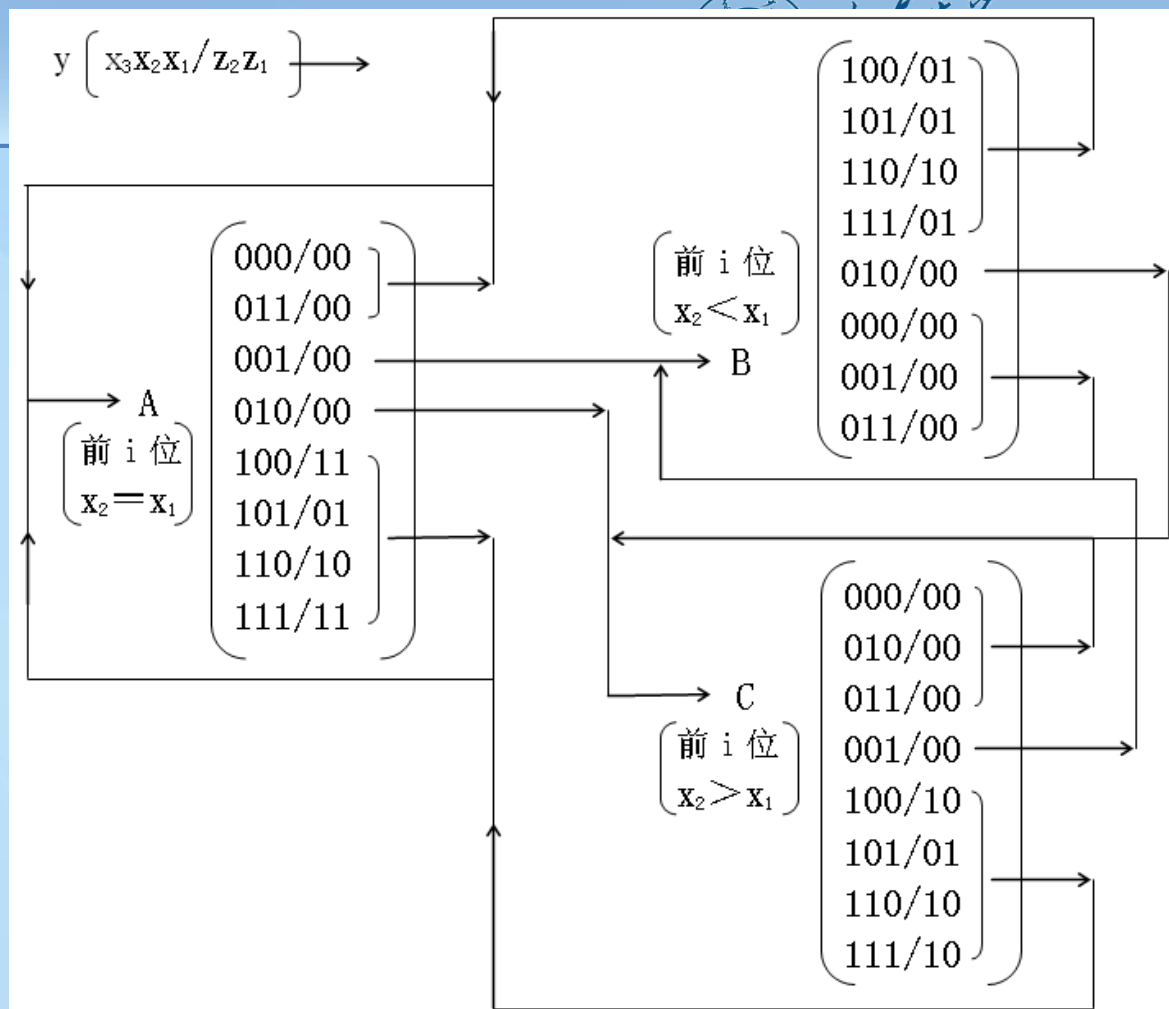
若 $x_2 < x_1$ ，则 $z_2z_1=01$ ；

若 $x_2 > x_1$ ，则 $z_2z_1=10$ ；

若 $x_2 = x_1$ ，则 $z_2z_1=11$ ；

状态表

| y \ x <sub>3</sub> x <sub>2</sub> x <sub>1</sub> |  |      |      |      |      |      |      |      |      |                               |
|--|--|------|------|------|------|------|------|------|------|-------------------------------|
|  |  | 000  | 001  | 011  | 010  | 110  | 111  | 101  | 100  |                               |
| A  |  | A/00 | B/00 | A/00 | C/00 | A/10 | A/11 | A/01 | A/11 | z <sub>2</sub> z <sub>1</sub> |
| B  |  | B/00 | B/00 | B/00 | C/00 | A/10 | A/01 | A/01 | A/01 |                               |
| C  |  | C/00 | B/00 | C/00 | C/00 | A/10 | A/10 | A/01 | A/10 |                               |



# 作业



山东大学

计算机科学与技术学院

## ● P167练习五

➤ 9, 10, 11





# 内容提要



山东大学

计算机科学与技术学院



## 5.1 时序逻辑电路概述



## 5.2 同步时序逻辑电路分析



## 5.3 同步时序逻辑电路设计概述



## 5.4 建立状态表



## 5.5 状态化简



## 5.6 状态分配和网络实现



## 5.5 状态化简

合并状态表中外特性相同的状态（消除多余状态）

外特性相同的状态常称为“相容状态”

- 外特性是指状态的输入/输出特性
- 化简步骤
  - (1) 找相容状态对（可以合并的状态对）
  - (2) 找最大相容集合（可以合并的个数最多的状态）
  - (3) 找最小闭覆盖（构成简化表的必要最大相容）
  - (4) 作简化状态表

## 5.5 状态化简



合并状态表中外特性相同的状态（消除多余状态）

外特性相同的状态常称为“相容状态”

- 外特性是指状态的输入/输出特性

| 现态 \ x | 0   | 1   |
|--------|-----|-----|
| A      | A/0 | B/0 |
| B      | A/0 | C/0 |
| C      | A/0 | D/0 |
| D      | A/0 | E/1 |
| E      | A/0 | E/1 |



| 现态 \ x | 0   | 1   |
|--------|-----|-----|
| A      | A/0 | B/0 |
| B      | A/0 | C/0 |
| C      | A/0 | D/0 |
| D      | A/0 | D/1 |

## 5.5.1 确定相容状态对

(概念定理)



山东大学

计算机科学与技术学院

### 1. 概念定理

(设 $y_i$ 和 $y_j$ 是状态表M中的状态)

#### ▲状态相容

对状态表M的任意输入序列 $X(n)$ 作用下,  $y_i$ 和 $y_j$ 产生的指定输出都相同, 则定义 $y_i$ 和 $y_j$ 为**状态相容**。 (对任意输入序列)

#### ▲状态不相容

如果存在某个输入序列 $X(n)$ , 使 $y_i$ 和 $y_j$ 产生的某个指定输出不相同, 则定义 $y_i$ 和 $y_j$ 为**状态不相容**。 (某个输入序列)

#### ▲输出相容

对状态表M的全部指定输入,  $y_i$ 和 $y_j$ 产生的指定输出都相同, 则定义 $y_i$ 和 $y_j$ 为**输出相容**。 (全部一次输入)

#### ▲输出不相容

如果在状态表M的某个指定输入作用下,  $y_i$ 和 $y_j$ 产生的指定输出不相同, 则定义 $y_i$ 和 $y_j$ 为**输出不相容**。 (某个一次输入)

## 5.5.1 确定相容状态对

(概念定理)



山东大学

计算机科学与技术学院

### 定理5.1

$y_i$ 和 $y_j$ 状态不相容的充分必要条件是： $y_i$ 和 $y_j$ 输出不相容或次态不相容。

例

| $y \backslash x$ | 0   | 1   |
|------------------|-----|-----|
| A                | A/0 | B/0 |
| B                | C/0 | B/0 |
| C                | A/1 | B/0 |

$y' / z$

AC输出不相容，AB次态不相容。

## 5.5.1 确定相容状态对 (隐含表法)



山东大学

计算机科学与技术学院

### 2. 找相容对的隐含表法

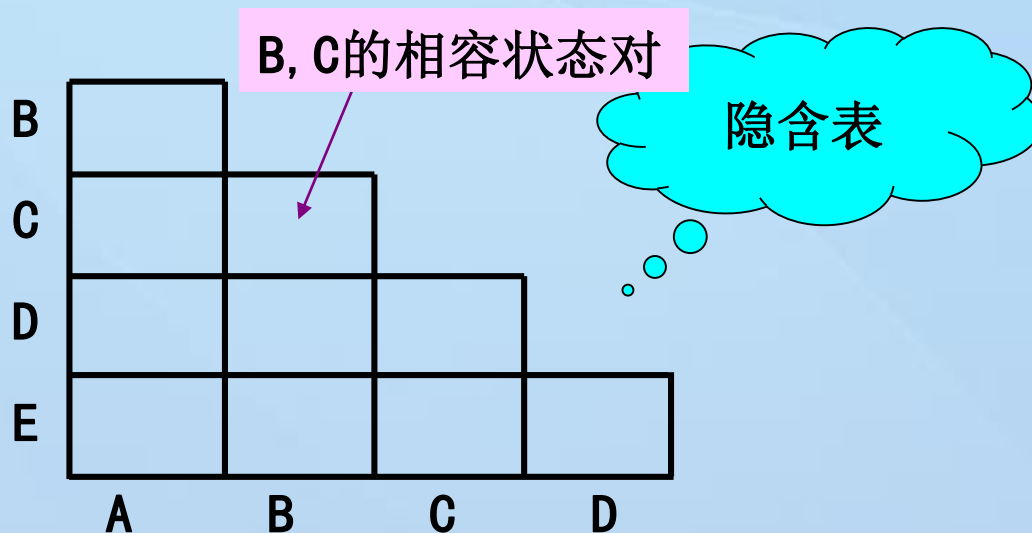
对状态表中的所有状态对，按定理5.1找出所有的不相容对，其余即为相容对。在查找过程中，用隐含表记录对各状态对的相容情况。

#### ▲隐含表

是一种阶梯状的表格，其中每个方格对应一个状态对。如

| y \ x | 0   | 1   |
|-------|-----|-----|
| A     | C/0 | B/0 |
| B     | D/0 | E/0 |
| C     | E/0 | E/0 |
| D     | A/1 | A/0 |
| E     | A/0 | A/0 |

$y' / z$



## 5.5.1 确定相容状态对 (隐含表法)



山东大学

计算机科学与技术学院

### ▲确定相容对的隐含表法

#### ①填隐含表

将各状态对的相容情况填入  
隐含表的相应方格中

|   |     |     |
|---|-----|-----|
| { | ×   | 不相容 |
|   | √   | 相容  |
|   | 次态对 | 隐含  |

#### ②查次态对

用“×”标出所有不相容次态  
对

#### ③确定相容对

无“×”方格所对应的状态对即为相容对

# 5.5.1 确定相容状态对

## 例5.5-1 确定图示状态表的相容状态对

|   |            |     |            |            |     |
|---|------------|-----|------------|------------|-----|
| 2 | 2,6        |     |            |            |     |
| 3 | ×          | ✓   |            |            |     |
| 4 | 1,2        | ×   | ✓          |            |     |
| 5 | 1,3<br>2,6 | ✓   | 1,5<br>1,6 |            |     |
| 6 | 3,4<br>2,6 | 4,5 | 4,5        | 1,5<br>1,6 | 1,4 |
|   | 1          | 2   | 3          | 4          | 5   |

隐含表

不完全指定状态表是指含有任意项的状态表，其中任意项可以是次态或输出。

| y \ x | 1   | 2   | 3   |
|-------|-----|-----|-----|
| 1     | 3/0 | *   | 2/* |
| 2     | *   | 4/0 | 6/* |
| 3     | 5/1 | *   | */0 |
| 4     | *   | 1/1 | 1/* |
| 5     | 1/* | *   | 6/* |
| 6     | 4/* | 5/* | 6/* |

$y' / z$

原始状态表

### 1. 填隐含表(按列填表)

对原始状态表进行顺序比较，

- 1) 输出不相同，打×，表示不等效
- 2) 输出完全相同，且（次态相同或呈交错），打✓，表示状态等效
- 3) 输出完全相同，但次态不相同且非交错，将相容状态对填入隐含表以待进一步比较

### 2. 查次态对

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

### 3. 定相容对

无“×”方格所对应的状态对就是相容对



## 5.5.2 确定最大相容集合



山东大学

计算机科学与技术学院

### 相容状态集合

设 $Q=\{y_1, y_2, \dots, y_k\}$ 是状态表 $M$ 的一个状态集合，如果对于每一个状态对 $(y_i, y_j) \in Q$ ，都有 $y_i y_j$  相容，则称 $Q$ 为相容状态集合。

### 最大相容集合

设 $Q$ 是状态表 $M$ 的一个相容状态集合，如果不存在一个状态 $y_p \in M$ ，但 $y_p \notin Q$ ，而 $y_p$ 与 $Q$ 中的每个状态都相容，则称 $Q$ 是最大相容集合，简称最大相容。

**最大相容就是状态个数最多并且相互相容的状态集合。**

用关系图法确定最大相容集合：

- 1) 用分布在圆周上的点表示各状态；
- 2) 用直线连接各相容状态对；
- 3) 找出各“最大完备多边形”，每个最大完整多边形对应一个最大相容集合。  
(完备多边形是指各顶点之间均有连线的广义多边形)



# 5.5.3 确定最大相容集合



山东大学

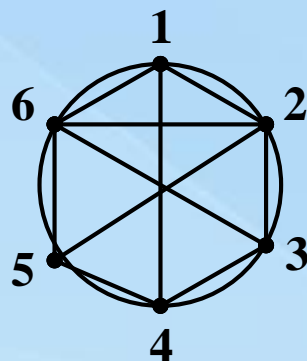
计算机科学与技术学院

## 例5.5-2 确定下列隐含表的最大相容

步骤:

- 1) 用分布在圆周上的点表示各状态;
- 2) 用直线连接各相容状态对;
- 3) 找出各“最大完备多边形”

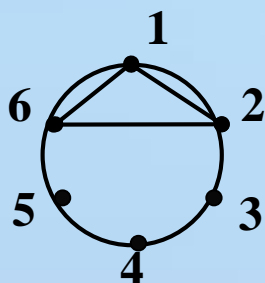
逐点按线查找,  
每条线都要用到。  
一条线可以用多次



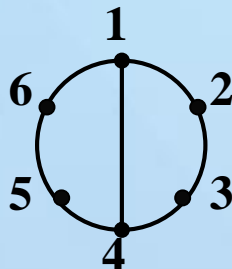
关系图

|   |            |     |     |            |     |
|---|------------|-----|-----|------------|-----|
| 2 | 2,6        |     |     |            |     |
| 3 | ×          | √   |     |            |     |
| 4 | 1,2        | ×   | √   |            |     |
| 5 | 1,3<br>2,6 | ×   | √   | 1,5<br>1,6 |     |
| 6 | 3,4<br>2,6 | 4,5 | 4,5 | 1,5<br>1,6 | 1,4 |
|   | 1          | 2   | 3   | 4          | 5   |

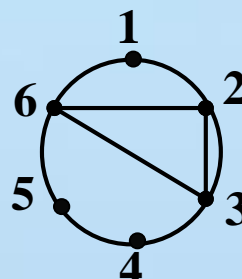
隐含表



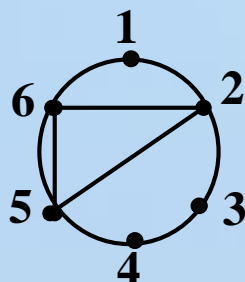
{1,2,6}



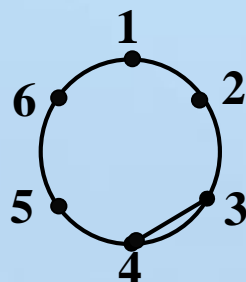
{1,4}



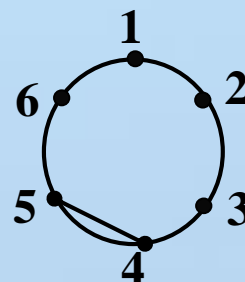
{2,3,6}



{2,5,6}



{3,4}



{4,5}

## 5.5.2 确定简化状态表



从最大相容中找出构成**简化状态表**的必要最大相容，即最小闭覆盖。

简化状态表

如果 $M'$ 是状态表 $M$ 的简化状态表，则 $M'$ 应满足：

- (a)覆盖性： $\forall$  对 $y_i \in M$ ，至少存在一个 $C_j \in M'$ ，使 $M'$ 覆盖了 $M$ 的全部状态，我们就称 $M'$ 满足覆盖性；
- (b)闭合性： $\forall$  对 $y_i \in M$ ，至少存在一个 $C_j \in M'$ ，使 $C_i$ 的次态 $\in C_j$ ，即 $M'$ 中状态的次态仍是 $M'$ 的状态，我们就称 $M'$ 满足闭合性；
- (c)状态个数最少。

### 1. 完全指定状态表的化简

**没有任意项的状态表，即状态表中次态和输出是完全指定的，没有任意项。**

### 2. 不完全指定状态表的化简

**不完全指定状态表是指含有任意项的状态表，其中任意项可以是次态或输出。**

## 5.5.2.1 完全指定状态表的化简



山东大学

计算机科学与技术学院

**定理5.2** 设 $M$ 是完全指定状态表,  $y_i, y_j, y_k \in M$ 。如果 $y_i$ 和 $y_j$ 相容,  $y_j$ 和 $y_k$ 相容, 则 $y_i$ 和 $y_k$ 相容。即对于完全指定状态表, 状态相容具有传递性。

**定理5.3** 对于完全指定状态表, 全部最大相容满足最小闭覆盖。

即对于构成完全指定状态表的简化状态表, 所有的最大相容都是必要的, 所以找出最大相容后, 就可以作简化状态表了。

# 完全指定状态表的化简举例



山东大学

计算机科学与技术学院

## 例5.5-3 简化下图所示的状态表

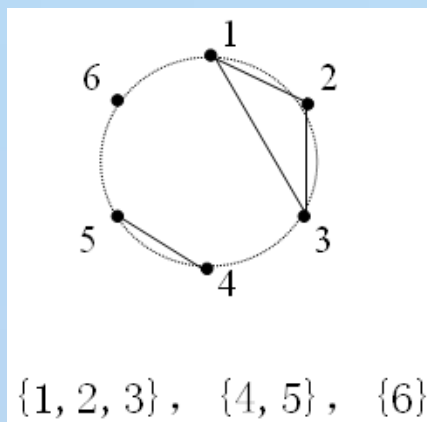
状态表

| y \ x |                |                |                |      |
|-------|----------------|----------------|----------------|------|
|       | I <sub>1</sub> | I <sub>2</sub> | I <sub>3</sub> |      |
| 1     | 3/0            | 4/1            | 3/0            | y'/z |
| 2     | 2/0            | 4/1            | 2/0            |      |
| 3     | 2/0            | 5/1            | 1/0            |      |
| 4     | 1/1            | 6/0            | 6/0            |      |
| 5     | 2/1            | 6/0            | 6/0            |      |
| 6     | 1/1            | 5/1            | 4/1            |      |

步骤1: 根据状态表确定隐含表, 找相容状态对

|   |                      |              |   |      |   |
|---|----------------------|--------------|---|------|---|
| 2 | 2, 3                 |              |   |      |   |
| 3 | 1, 3<br>2, 3<br>4, 5 | 1, 2<br>4, 5 |   |      |   |
| 4 | ×                    | ×            | × |      |   |
| 5 | ×                    | ×            | × | 1, 2 |   |
| 6 | ×                    | ×            | × | ×    | × |
|   | 1                    | 2            | 3 | 4    | 5 |

步骤2: 画出状态图, 求最大相容



关系图和最大相容

步骤3: 找最小闭覆盖

| y \ x   |                |                |                |      |
|---------|----------------|----------------|----------------|------|
|         | I <sub>1</sub> | I <sub>2</sub> | I <sub>3</sub> |      |
| 1, 2, 3 | 2, 3/0         | 4, 5/1         | 1, 2, 3/0      | y'/z |
| 4, 5    | 1, 2/1         | 6/0            | 6/0            |      |
| 6       | 1/1            | 5/1            | 4/1            |      |

简化状态表

步骤4: 合并状态, 做简化状态表

| y \ x     |                |                |                |      |
|-----------|----------------|----------------|----------------|------|
|           | I <sub>1</sub> | I <sub>2</sub> | I <sub>3</sub> |      |
| (1,2,3) A | A/0            | B/1            | A/0            | y'/z |
| (4,5) B   | A/1            | C/0            | C/0            |      |
| (6) C     | A/1            | B/1            | B/1            |      |

最终的简化状态表

## 5.5.3 确定简化状态表



山东大学

计算机科学与技术学院

### ▲简化状态表

是与原状态表外特性相同而状态个数最少的状态表。

覆盖

最小

闭合

即 简化状态表应满足

(a) 覆盖性

(b) 闭合性

(c) 最小

最小闭覆盖

## 1. 完全指定状态表的化简

▲完全指定状态表 是指状态表中无任意项。

### ▲定理5.3

对于完全指定状态表，全部最大相容满足最小闭覆盖。

即 对于完全指定状态表，所有最大相容都是必要的。

找出最大相容后就可将每个最大相容合并成一个新状态，  
据此作简化状态表。

## 5.5.3 确定简化状态表 (不完全指定表)



山东大学

计算机科学与技术学院

### 2. 不完全指定状态表的化简

▲不完全指定状态表 是指状态表中有任意项

▲对于不完全指定状态表，全部最大相容中可能有多余的。

需要从中找出必要的最大相容，即**最小闭覆盖**。

可利用闭合覆盖表查找必要最大相容（最小闭覆盖）

▲查找步骤：

①作闭合覆盖表（**闭覆表**）；

②找一个最小覆盖；

③检查闭合性，若满足转⑥；

④消去相交部分，再查闭合性，  
若满足转⑥；

⑤找一个较小覆盖，转③；

⑥作简化状态表。

闭合覆盖表

| 最大相容<br>(列出全部<br>最大相容) | 覆盖性<br>(标出原状态表的<br>全部状态) | 闭合性<br>(标出原状态表的<br>全部输入) |
|------------------------|--------------------------|--------------------------|
|                        |                          |                          |

闭合性：次态集合是否属于某一相容集合



## 例5.5-4 简化图示状态表

| $y \backslash x$ | $I_1$ | $I_2$ | $I_3$ |
|------------------|-------|-------|-------|
| 1                | 3/0   | *     | 2/*   |
| 2                | *     | 4/0   | 6/*   |
| 3                | 5/1   | *     | */0   |
| 4                | *     | 1/1   | 1/*   |
| 5                | 1/*   | *     | 6/*   |
| 6                | 4/*   | 5/*   | 6/*   |

$y' / z$

闭合覆盖表

| 最大相容    | 覆盖性 |   |   |   |   |   | 闭合性   |       |       |
|---------|-----|---|---|---|---|---|-------|-------|-------|
|         | 1   | 2 | 3 | 4 | 5 | 6 | $I_1$ | $I_2$ | $I_3$ |
| *       | ✓   | ✓ |   |   |   | ✓ | 3, 4  | 4, 5  | 2, 6  |
| 1, 2, 6 |     |   |   |   |   |   |       |       |       |
| 1, 4    |     |   |   |   |   |   |       |       |       |
| 2, 3, 6 |     |   |   |   |   |   |       |       |       |
| 2, 5, 6 |     |   |   |   |   |   |       |       |       |
| * 3, 4  |     |   | ✓ | ✓ |   |   | 5     | 1     | 1     |
| * 4, 5  |     |   |   | ✓ | ✓ |   | 1     | 1     | 1, 6  |

简化表

| $y \backslash x$ | $I_1$ | $I_2$ | $I_3$ |
|------------------|-------|-------|-------|
| (1, 2, 6) A      | B/0   | C/1   | A/*   |
| (3, 4) B         | C/1   | A/1   | A/0   |
| (4, 5) C         | A/*   | A/1   | A/*   |

$y' / z$

①作闭合覆盖表；

②找一个最小覆盖；

③检查闭合性，

若满足转⑥；

⑥作简化状态表。

用个数最少的集合  
包含所有原状态

所选集合的各次态集必须  
属于所选的某个集合

所选各集合并为一个状态  
，并用新名表示

# 5.5.3 确定简化状态表 (例5.5-5)



山东大学

计算机科学与技术学院

例5.5-5 简化图示状态表

| $y \backslash x$ | 0   | 1   |
|------------------|-----|-----|
| 1                | 5/0 | 1/0 |
| 2                | 4/0 | 2/0 |
| 3                | 5/* | 3/* |
| 4                | 1/1 | 1/1 |
| 5                | 1/* | 2/* |

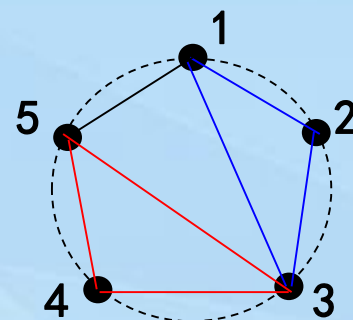
$y' / z$

无“×”方格所对应的状态对是相容对

(1) 隐含表

|   | 1    | 2      | 3            | 4    |
|---|------|--------|--------------|------|
| 2 | 4, 5 |        |              |      |
| 3 | √    | 4, 5   |              |      |
| 4 | ×    | ×      | 1, 3<br>1, 5 |      |
| 5 | 1, 2 | 1, 4 × | 1, 5<br>2, 3 | 1, 2 |

(2) 关系图



最大相容

$\{1, 2, 3\}$ ,  $\{1, 3, 5\}$   
 $\{3, 4, 5\}$

对原始状态表进行顺序比较,

- 1) 输出不相同, 打×, 表示不等效
- 2) 输出完全相同, 且 (次态相同或呈交错), 打√, 表示状态等效
- 3) 输出完全相同, 但次态不相同且非交错, 将相容状态对填入隐含表以待进一步比较

# 5.5.3 确定简化状态表 (例5.5-5)



山东大学

计算机科学与技术学院

| y \ x | 0   | 1   |
|-------|-----|-----|
| 1     | 5/0 | 1/0 |
| 2     | 4/0 | 2/0 |
| 3     | 5/* | 3/* |
| 4     | 1/1 | 1/1 |
| 5     | 1/* | 2/* |

$y' / z$

(3) 闭覆盖表

| 最大相容                | 覆盖性 |   |   |   |   | 闭合性  |         |
|---------------------|-----|---|---|---|---|------|---------|
|                     | 1   | 2 | 3 | 4 | 5 | 0    | 1       |
| #1, 2, 3<br>1, 3, 5 | ✓   | ✓ | ✓ |   |   | 4, 5 | 1, 2, 3 |
| #3, 4, 5            |     |   | ✓ | ✓ | ✓ | 1, 5 | 1, 2, 3 |
| 4, 5                |     |   |   | ✓ | ✓ | 1    | 1, 2    |

消去3

不闭合

(4) 简化表

| y \ x       | 0   | 1   |
|-------------|-----|-----|
| (1, 2, 3) A | B/0 | A/0 |
| (4, 5) B    | A/1 | A/1 |

$y' / z$

- ①作闭合覆盖表;
- ②找一个最小覆盖;
- ③检查闭合性, 若满足转⑥;
- ④消去相交部分, 再查闭合性, 若满足转⑥;
- ⑥作简化状态表。

## 例5.5-6 简化图示状态表

| $X_2X_1$<br>y | 00  | 01  | 11  | 10  |
|---------------|-----|-----|-----|-----|
| 1             | 3/0 | 1/* | *   | *   |
| 2             | 6/* | 1/0 | 1/* | *   |
| 3             | */1 | *   | 4/0 | *   |
| 4             | 1/0 | *   | *   | 5/1 |
| 5             | *   | 5/* | 2/1 | 1/1 |
| 6             | *   | 2/1 | 6/* | 4/1 |

$y' / z$

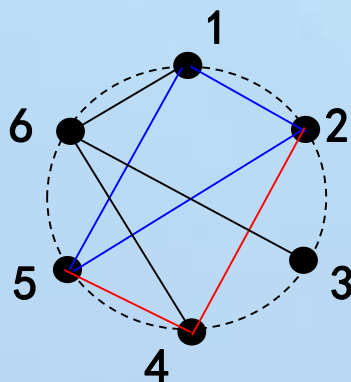
### (1) 隐含表

|   | 1    | 2            | 3    | 4    | 5                    |
|---|------|--------------|------|------|----------------------|
| 2 | 3, 6 |              |      |      |                      |
| 3 | ×    | 1, 4         |      |      |                      |
| 4 | 1, 3 | 1, 6         | ×    |      |                      |
| 5 | ✓    | 1, 2<br>1, 5 | ×    | 1, 5 |                      |
| 6 | 1, 2 | ×            | 4, 6 | 4, 5 | 1, 4<br>2, 5<br>2, 6 |

???

### (2) 关系图

???



### 最大相容

{1, 2, 5}, {1, 6}, {2, 4, 5}

{3, 6}, {4, 6}

## 例5.5-6 简化图示状态表

| $X_2X_1$ | 00  | 01  | 11  | 10  |
|----------|-----|-----|-----|-----|
| y        |     |     |     |     |
| 1        | 3/0 | 1/* | *   | *   |
| 2        | 6/* | 1/0 | 1/* | *   |
| 3        | */1 | *   | 4/0 | *   |
| 4        | 1/0 | *   | *   | 5/1 |
| 5        | *   | 5/* | 2/1 | 1/1 |
| 6        | *   | 2/1 | 6/* | 4/1 |

$y' / z$

### (3) 闭覆盖表

| 最大相容         | 覆盖性 |   |   |   |   |   | 闭合性              |            |            |      |
|--------------|-----|---|---|---|---|---|------------------|------------|------------|------|
|              | 1   | 2 | 3 | 4 | 5 | 6 | 00<br>10<br>3, 6 | 01<br>1, 5 | 11<br>1, 2 | 1    |
| *<br>1, 2, 5 | ✓   | ✓ |   |   | ✓ |   |                  |            |            |      |
| *<br>1, 6    |     |   |   |   |   |   |                  |            |            |      |
| *<br>2, 4, 5 |     | ✓ |   | ✓ | ✓ |   | 1, 6             | 1, 5       | 1, 2       | 1, 5 |
| #<br>3, 6    |     |   | ✓ |   |   | ✓ | *                | 2          | 4, 6       | 4    |
| +<br>4, 6    |     |   |   |   |   |   | 1                | 2          | 6          | 4, 5 |
| 4, 5         |     |   |   | ✓ | ✓ |   | 1                | 5          | 2          | 1, 5 |

简化状态集 (1, 2, 5)、(3, 6)、(4, 6)、(4, 5)

### ▲查找步骤:

- ①作闭合覆盖表（闭覆盖表）；
- ②找一个最小覆盖；
- ③检查闭合性，若满足转⑥；
- ④消去相交部分，再查闭合性，若满足转⑥；
- ⑤找一个较小覆盖，转③；
- ⑥作简化状态表。

## 例5.5-6 简化图示状态表

| $X_2X_1$<br>y | 00  | 01  | 11  | 10  |
|---------------|-----|-----|-----|-----|
| 1             | 3/0 | 1/* | *   | *   |
| 2             | 6/* | 1/0 | 1/* | *   |
| 3             | */1 | *   | 4/0 | *   |
| 4             | 1/0 | *   | *   | 5/1 |
| 5             | *   | 5/* | 2/1 | 1/1 |
| 6             | *   | 2/1 | 6/* | 4/1 |

$y' / z$

### (4) 简化表

| $X_2X_1$<br>y | 00  | 01  | 11  | 10  |
|---------------|-----|-----|-----|-----|
| (1, 2, 5) A   | B/0 | A/0 | A/1 | A/1 |
| (3, 6) B      | */1 | A/1 | C/0 | D/1 |
| (4, 6) C      | A/0 | A/1 | C/* | D/1 |
| (4, 5) D      | A/0 | A/* | A/1 | A/1 |

$y' / z$

可选 (C, D)

可选 (A, D)

可选 (B, C)

次态集的归并可能有多种选择，应按照“尽量与相邻单元相同，以便于化简合并”的原则来确定。

# 内容提要



山东大学

计算机科学与技术学院



## 5.1 时序逻辑电路概述



## 5.2 同步时序逻辑电路分析



## 5.3 同步时序逻辑电路设计概述



## 5.4 建立状态表



## 5.5 状态化简



## 5.6 状态分配和网络实现



## 5.6.1 状态分配

用二进制编码表示各状态（状态编码）

▲代码位数 $r$ 和状态个数 $n$ 应满足

$$2^r \geq n \quad \text{且} r \text{ 最小}$$

编码方案可有多种，一般采用经验分配原则和平滑转换原则。

### 1. 经验分配原则

按便于简化表达式，使设计的电路较简的目标进行状态分配。

▲经验原则

I. 次态相同，现态相邻；

①同列相同

②不同列相同

II. 同一现态，次态相邻；

III. 输出相同，状态相邻；

处理的优先顺序  
为 I，II，III

状态相邻是指两个状态的编码只有一位不同



# 5.6.1 状态分配 (例5.6-1)

例5.6-1 对图示状态表按经验原则进行状态编码

|   |   | x      |     |
|---|---|--------|-----|
|   |   | 0      | 1   |
| y | 1 | 1/0    | 3/0 |
|   | 2 | 4/0    | 3/0 |
|   | 3 | 2/0    | 3/0 |
|   | 4 | 1/1    | 3/0 |
|   |   | y' / z |     |

(1) 相邻关系

I. ① (1, 4), (1, 2, 3, 4)

② ---

II. (1, 3)\*, (3, 4), (2, 3)

III. (1, 2, 3)

(2) 状态分配

|                |   | y <sub>2</sub> |   |
|----------------|---|----------------|---|
|                |   | 0              | 1 |
| y <sub>1</sub> | 0 | 1              | 4 |
|                | 1 | 3              | 2 |

(3) 赋值状态表

|                               |        | x                                   |      |
|-------------------------------|--------|-------------------------------------|------|
|                               |        | 0                                   | 1    |
| y <sub>2</sub> y <sub>1</sub> | (1) 00 | 00/0                                | 01/0 |
|                               | (3) 01 | 11/0                                | 01/0 |
|                               | (2) 11 | 10/0                                | 01/0 |
|                               | (4) 10 | 00/1                                | 01/0 |
|                               |        | y' <sub>2</sub> y' <sub>1</sub> / z |      |

## ▲经验原则

I. 次态相同，现态相邻；

①同列相同

②不同列相同

II. 同一现态，次态相邻；

III. 输出相同，状态相邻；

## 5.6.1 状态分配 (例5.6-2)

例5.6-2 对图示状态表按经验原则进行状态编码

| $x \backslash y$ | 0   | 1   |
|------------------|-----|-----|
| 1                | 2/0 | 3/0 |
| 2                | 4/0 | 5/0 |
| 3                | 5/0 | 4/0 |
| 4                | 6/0 | 7/0 |
| 5                | 7/0 | 6/1 |
| 6                | 1/0 | 1/0 |
| 7                | 1/0 | 1/1 |

$y' / z$

### (1) 相邻关系

I. ① (6, 7)

② (2, 3), (4, 5)

II. (2, 3) (4, 5)\*, (6, 7)\*

III. (1, 2, 3, 4, 6), (5, 7)

### (2) 状态分配

| $y_3 y_2$ |  | 00 | 01 | 11 | 10 |
|-----------|--|----|----|----|----|
| $y_1$     |  |    |    |    |    |
| 0         |  | 1  | 2  | 4  | 6  |
| 1         |  |    | 3  | 5  | 7  |

# 5.6.1 状态分配 (例5.6-2)



山东大学

计算机科学与技术学院

(3) 赋值状态表

|         |   | x   |     |
|---------|---|-----|-----|
|         |   | 0   | 1   |
| y       |   |     |     |
| (000) 1 | 1 | 2/0 | 3/0 |
| (010) 2 | 2 | 4/0 | 5/0 |
| (011) 3 | 3 | 5/0 | 4/0 |
| (110) 4 | 4 | 6/0 | 7/0 |
| (111) 5 | 5 | 7/0 | 6/1 |
| (100) 6 | 6 | 1/0 | 1/0 |
| (101) 7 | 7 | 1/0 | 1/1 |

$y' / z$

|                               |  | xy <sub>3</sub> |       |       |       |
|-------------------------------|--|-----------------|-------|-------|-------|
|                               |  | 00              | 01    | 11    | 10    |
| y <sub>2</sub> y <sub>1</sub> |  |                 |       |       |       |
| 00                            |  | 010/0           | 000/0 | 000/0 | 011/0 |
| 01                            |  | *               | 000/0 | 000/1 | *     |
| 11                            |  | 111/0           | 101/0 | 100/1 | 110/0 |
| 10                            |  | 110/0           | 100/0 | 101/0 | 111/0 |

$y'_3 y'_2 y'_1 / z$

# 5.6.1 状态分配

(平滑原则)



山东大学

计算机科学与技术学院

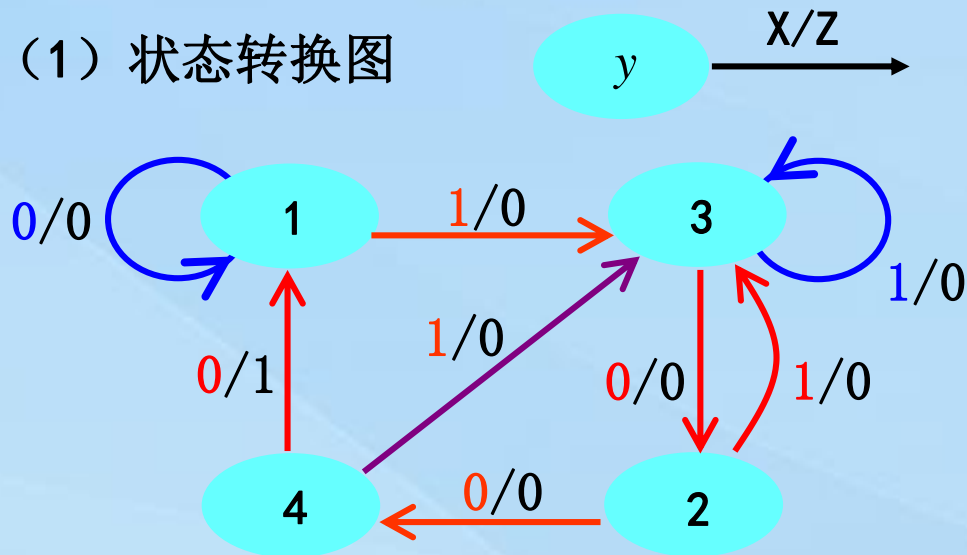
## 2. 平滑转换原则

以状态转换时没有竞争冒险为目标进行状态分配。

### ▲平滑转换原则

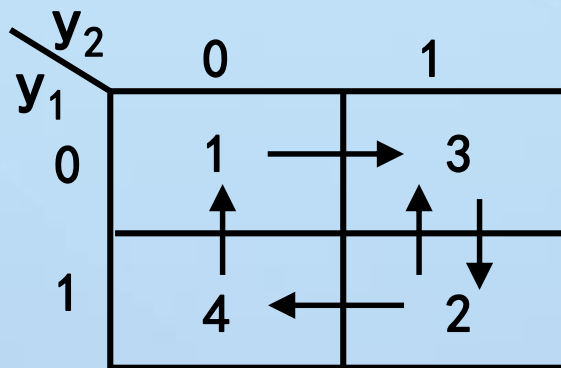
状态相继，编码相邻。

(1) 状态转换图



(2) 状态分配

|   |   | x      |     |
|---|---|--------|-----|
|   |   | 0      | 1   |
| y | 1 | 1/0    | 3/0 |
|   | 2 | 4/0    | 3/0 |
|   | 3 | 2/0    | 3/0 |
|   | 4 | 1/1    | 3/0 |
|   |   | y' / z |     |



状态4到3转换不能满足平滑性。

# 5.6.1 状态分配 (平滑原则)

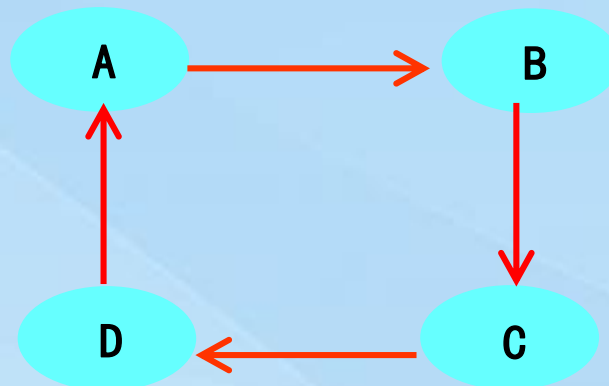


山东大学

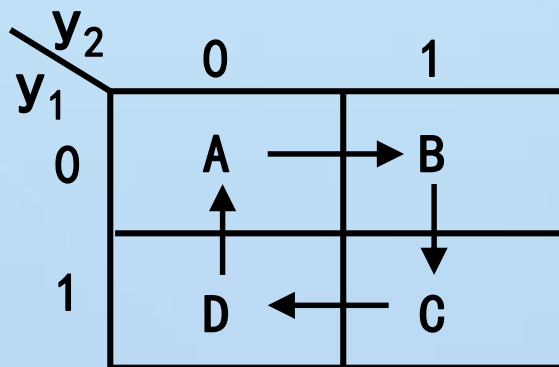
计算机科学与技术学院

状态相继，编码相邻。

(1) 状态转换图



(2) 状态分配



## 5.6.2 电路实现

- **定义：由赋值表作出逻辑电路图**

- 每位状态变量对应一个触发器
- 输出函数一般比较简单

- **电路实现的主要工作是确定各触发器的激励函数**

### 方法一、激励表法

- 方法：作激励表，写表达式，画电路图
- 特点：
  - 能处理约束条件，容易实现最简
  - 需作较多表格，每个触发器一个

### 方法二、状态方程法

- 方法：写方程式，定激励表，画电路图
  - ① 直接由赋值表写方程
  - ② 与特征方程比较系数确定激励
- 特点：
  - 较为简捷
  - 不便于处理约束项，有任意项需注意简化

# 例5. 6-5 用JK触发器实现下图状态表

方法一、激励表法

状态表

| x \ y2y1 | 0    | 1    |
|----------|------|------|
|          |      |      |
| 00       | 00/0 | 01/0 |
| 01       | 11/0 | 01/0 |
| 11       | 10/0 | 01/0 |
| 10       | 00/1 | 01/0 |

y2' y1' / z

(1) 作激励表:

JK触发器激励表

| Q → Q' | J K |
|--------|-----|
| 0 0    | 0 * |
| 0 1    | 1 * |
| 1 0    | * 1 |
| 1 1    | * 0 |

$$Q' = J\bar{Q} + \bar{K}Q$$

当Q=0时, Q'与J一致, 与K无关

当Q=1时, Q'与K相反, 与J无关

(2) 写表达式

$$J_2 = \bar{x}y_1$$

$$K_2 = x + \bar{y}_1 = \overline{\bar{x}y_1}$$

$$J_1 = x$$

$$K_1 = \bar{x}y_2$$

$$z = \bar{x}y_2\bar{y}_1$$

触发器2的激励表

(由y2的现态和次态决定)

| x \ y2y1 | 0  | 1  |
|----------|----|----|
|          |    |    |
| 00       | 0* | 0* |
| 01       | 1* | 0* |
| 11       | *0 | *1 |
| 10       | *1 | *1 |

J2 K2

触发器1的激励表

(由y1的现态和次态决定)

| x \ y2y1 | 0  | 1  |
|----------|----|----|
|          |    |    |
| 00       | 0* | 1* |
| 01       | *0 | *0 |
| 11       | *1 | *0 |
| 10       | 0* | 1* |

J1 K1

## (2) 写表达式

$$J_2 = \bar{x}y_1$$

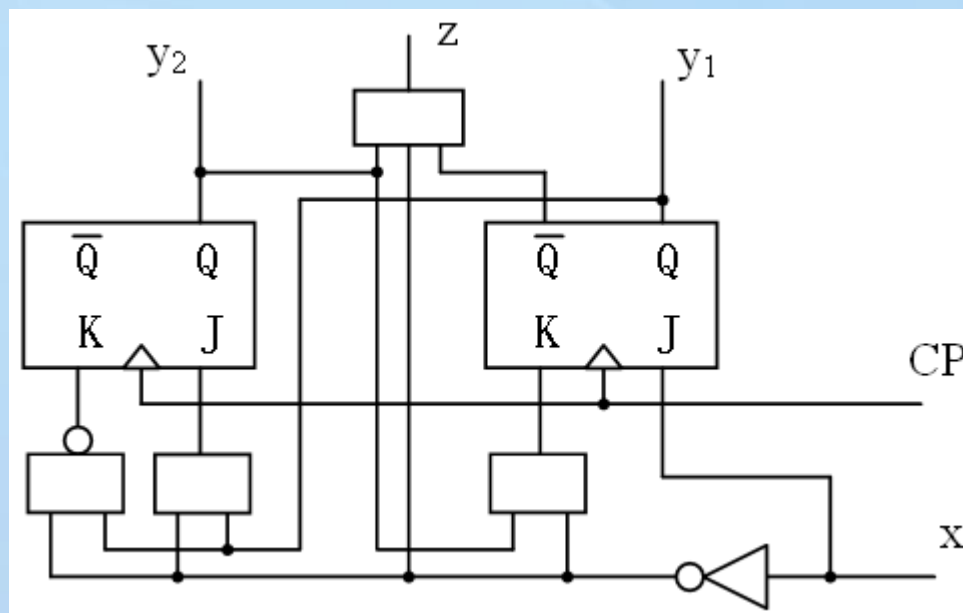
$$K_2 = x + \bar{y}_1 = \overline{\bar{x}y_1}$$

$$J_1 = x$$

$$K_1 = \bar{x}y_2$$

$$z = \bar{x}y_2\bar{y}_1$$

## (3) 画电路图





## 例5.6-5 实现下图状态表

方法二、状态方程法

状态表

|                               |    | x                                   |      |
|-------------------------------|----|-------------------------------------|------|
|                               |    | 0                                   | 1    |
| y <sub>2</sub> y <sub>1</sub> | 00 | 00/0                                | 01/0 |
|                               | 01 | 11/0                                | 01/0 |
|                               | 11 | 10/0                                | 01/0 |
|                               | 10 | 00/1                                | 01/0 |
|                               |    | y <sub>2</sub> ' y <sub>1</sub> '/z |      |

(1) 写方程

$$Q' = J\bar{Q} + \bar{K}Q$$

$$z = \bar{x}y_2y_1$$

$$y_2' = \bar{x}y_1 = \bar{x}y_1(\bar{y}_2 + y_2) = \bar{x}y_1\bar{y}_2 + \bar{x}y_1y_2$$

$$y_1' = x + \bar{y}_2y_1 = x\bar{y}_1 + (x + \bar{y}_2)y_1$$

(2) 定激励

若用D触发器，则

$$D_2 = \bar{x}y_1$$

$$D_1 = x + \bar{y}_2y_1$$

若用JK触发器，则

$$J_2 = \bar{x}y_1$$

$$J_1 = x$$

$$K_2 = \bar{\bar{x}y_1}$$

$$K_1 = \overline{x + \bar{y}_2} = \bar{x}y_2$$

(3) 画电路图

$$z = \bar{x}y_2\bar{y}_1$$

# 例5. 6-6 用JK触发器实现下图状态转换表 方法二、状态方程法

状态表

|                               |    | xy <sub>3</sub>                                      |       |       |       |
|-------------------------------|----|--|-------|-------|-------|
|                               |    | 00   | 01    | 11    | 10    |
| y <sub>2</sub> y <sub>1</sub> | 00 | 010/0  | 000/0 | 000/0 | 011/0 |
|                               | 01 | *  | 000/0 | 000/1 | *     |
|                               | 11 | 111/0  | 101/0 | 100/1 | 110/0 |
|                               | 10 | 110/0  | 100/0 | 101/0 | 111/0 |
|                               |    | y <sub>3</sub> ' y <sub>2</sub> ' y <sub>1</sub> '/z |       |       |       |

## (1) 写方程

$$Q' = J\bar{Q} + \bar{K}Q$$

$$z = xy_3y_1$$

$$y_3' = y_2 = y_2\bar{y}_3 + y_2y_3$$

$$y_2' = \bar{y}_3 = \bar{y}_3\bar{y}_2 + \bar{y}_3y_2$$

$$y_1' = \bar{x}y_2y_1 + xy_2\bar{y}_1 + x\bar{y}_3\bar{y}_1$$

## (2) 定激励

$$J_3 = y_2$$

$$J_2 = \bar{y}_3$$

$$J_1 = x(y_2 + \bar{y}_3)$$

$$K_3 = \bar{y}_2$$

$$K_2 = y_3$$

$$K_1 = \overline{\bar{x}y_2}$$

## 例5.6-6 用JK触发器实现下图状态转换表

### 方法二、状态方程法

(1) 写方程

$$z = xy_3y_1$$

(2) 定激励

$$J_3 = y_2$$

$$J_2 = \bar{y}_3$$

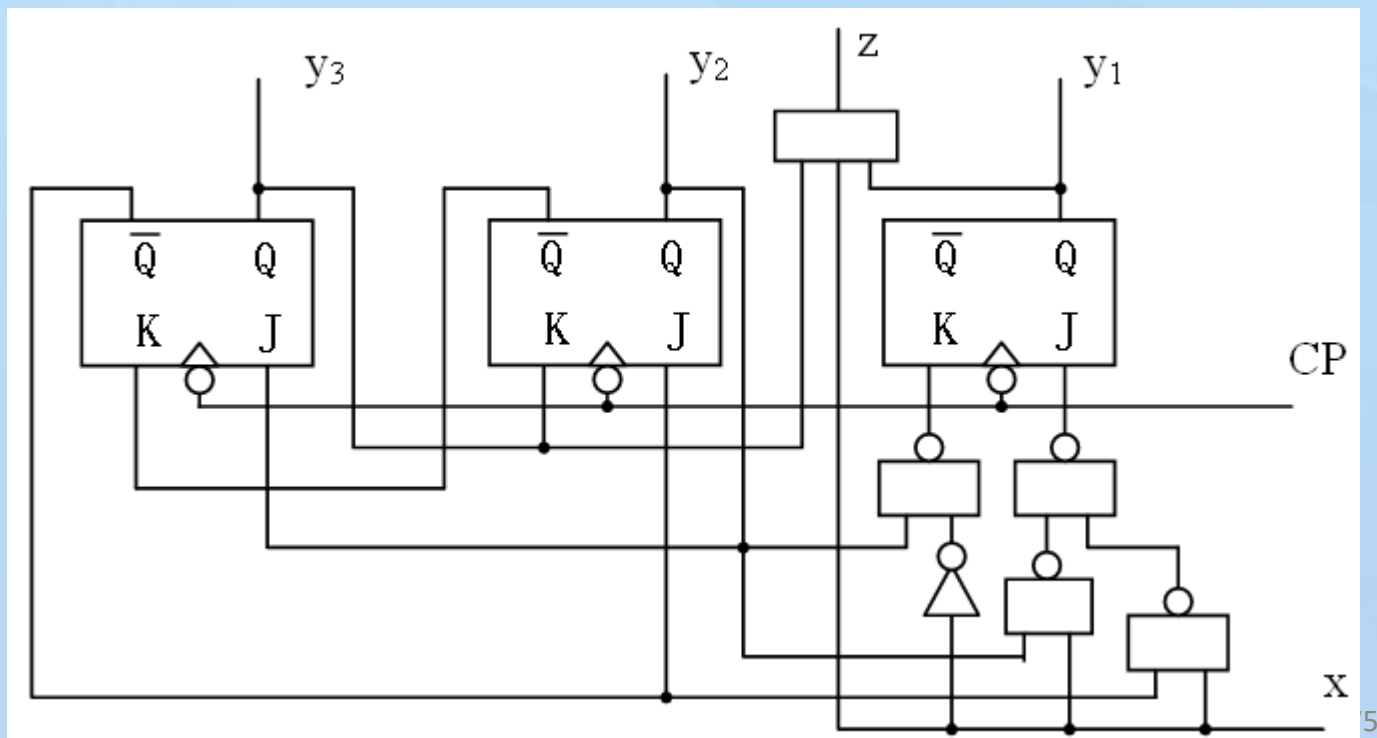
$$J_1 = x(y_2 + \bar{y}_3)$$

$$K_3 = \bar{y}_2$$

$$K_2 = y_3$$

$$K_1 = \bar{x}y_2$$

(3) 画电路图



# 状态方程法注意事项



山东大学

计算机科学与技术学院

- 状态方程法较为简练，但难以处理约束条件（如RS触发器中RS=0）。
- 方程法需注意：
  - 激励可为常数
    - ⊕ 若  $y_i' = \bar{y}_i$  则  $J_i=1, K_i=1$
    - ⊕ 若  $y_i' = y_i$  则  $J_i=0, K_i=0$
  - 若状态表中有任意项，配项后需注意化简

状态表

|                               |    | x                                     |      |
|-------------------------------|----|---------------------------------------|------|
|                               |    | 0                                     | 1    |
| y <sub>2</sub> y <sub>1</sub> | 00 | 01/0                                  | 01/0 |
|                               | 01 | 11/0                                  | 10/0 |
|                               | 11 | *                                     | *    |
|                               | 10 | 00/1                                  | 01/0 |
|                               |    | y <sub>2</sub> ' y <sub>1</sub> ' / z |      |

## 用JK触发器实现下面的状态表

$$Q' = J\bar{Q} + \bar{K}Q$$

(1) 写方程

$$y_2' = y_1 = y_1\bar{y}_2 + y_1y_2$$

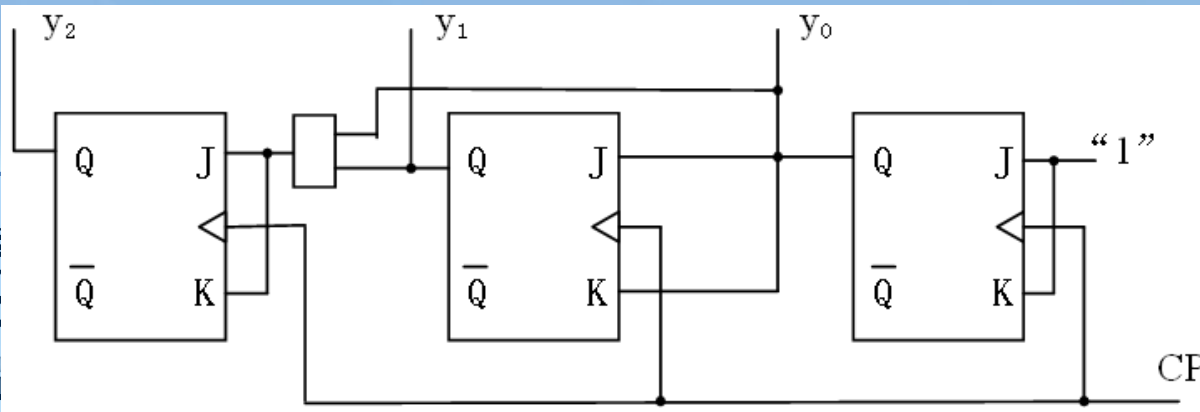
任意项

$$y_2' = y_1 = y_1\bar{y}_2 + 0 \cdot y_2$$

(2) 定激励

$$J_2 = y_1 \quad K_2 = 1 \quad \text{而不是} \quad K_2 = \bar{y}_1$$

• 计其电



• 例5.6-7 设计同步8进制加法计数器，用JK实现

(1) 状态转换表

(2) 方程

| y2y1 |   | y0          |     |     |     |
|------|---|-------------|-----|-----|-----|
|      |   | 00          | 01  | 11  | 10  |
| y0   | 0 | 001         | 011 | 111 | 101 |
|      | 1 | 010         | 100 | 000 | 110 |
|      |   | y2' y1' y0' |     |     |     |

$$y2' = \bar{y}_2 y_1 y_0 + y_2 \bar{y}_0 + y_2 \bar{y}_1$$

$$= y_1 y_0 \bar{y}_2 + (\bar{y}_0 + \bar{y}_1) y_2$$

$$y1' = y_0 \bar{y}_1 + y_1 \bar{y}_0$$

$$y0' = \bar{y}_0 = \bar{y}_0 + 0 \cdot y_0$$

(3) 定激励

$$J_2 = y_1 y_0$$

$$J_1 = y_0$$

$$J_0 = 1$$

$$K_2 = \bar{y}_1 + \bar{y}_0 = y_1 y_0$$

$$K_1 = y_0$$

$$K_0 = 1$$

(4) 画电路图

## ● 练习五

- 14
- 15(不完全指定状态表的化简)
- 16, 20, 22

