

## # 编码

1.2421 码：4 位二进制数，大于等于 5 的高位为 1：5 = 1011，小于 5 的最高位为 0 ：4 = 0100

2.有权码：8421、2421、5421 码这种每一位带权值的码

无权码：余三码，余三循环码，格雷码这种不是权值的码

余三码相加的修正问题：

如果两数相加没有进位产生，则结果需要减三，如果有进位产生，则结果需要加三。

3.自补码：余三码中 2=0101，各位取反的 1010 = 7 的余三码；2，7 是相对于 9 自补的

4.运算优先法则：括号---非---与---异或---或

## # 数字集成电路的分类：

（一）根据采用的半导体器件分类：

### 一类是双极型半导体器件作为元件的双极性集成电路

特点：速度快，负载能力强，但功耗较大。结构较复杂，因此集成规模收到一定限制

### 第二类是采用金属-氧化物-半导体场效应管 (MOSFET) 作为元件的单极型集成电路，简称为 MOS 电路

特点：结构简单，制造方便，集成度高，功耗低，但速度较慢

## (二) 根基集成规模的大小分类

小规模集成电路 (SSI)、中规模集成电路 (MSI)、大规模集成电路 (LSI)、超大规模集成电路 (VLSI)

分类依据：一片集成电路芯片上的元器件数目：

$n < 100$  : SSI

$100 < n < 999$  : MSI

$1000 < n < 9999$  : LSI

$N > 10000$  : VLSI

# # 半导体器件的开关特性

## (一) 晶体二极管的开关特性

静态开关特性：由二极管的单向导电特性决定，从伏安特性曲线可知，二极管的电压与电流是非线性的。

正向特性：

外加正向电压处于导通状态--->类似于开关的接通状态--->

使二极管处于导通状态的最小电压:导通电压

### 反向电压:

反向电压在一定范围内时, 电阻很大, 电压变化几乎不引起电流变化, 此时处于截至状态, 截至状态的反向电流被称为**反向饱和电流**

当反向电压超过某个极限值时, 反向电流突然猛增, 致使二极管被击穿, 此电压成为**击穿电压**

正向电压时, 应控制电流避免烧坏; 反向电压时, 应控制电压避免击穿

动态开关特性: 指二极管在导通和截至两种状态切换过程的特性, 它表现为完成两种状态的转换需要一定的时间。**通常把二极管从正向导通到反向截至的时间成为反向恢复时间; 而把二极管从反向截至到正向导通的时间成为开通时间。**

### (二) 晶体三极管的开关特性:

静态特性: 三极管由集电结发射结两个 PN 结组成, 有截至, 放大, 饱和三种工作状态。

**截至状态:** 输入电压  $v_1 \leq 0$ , 发电结和集电结均处于反偏状态 ( $V_b < V_e, V_b < V_c$ ), 此时工作在截至状态。

**放大状态:** 发电结正偏, 集电结反偏 ( $V_b > V_e, V_b < V_c$ ), 此时工作在放大状态。

**饱和状态:** 发电结和集电结都正偏 ( $V_b > V_e, V_b > V_c$ ), 三极管工

作在饱和状态。

动态特性：三极管在饱和和截至两种状态之间转换的过程具有的特性我们称作三极管的动态特性。

开通时间：三极管从截至到饱和导通的时间，等于延迟时间和上升时间之和。

关闭时间：三极管从饱和导通到截至的时间，等于存储时间和下降时间之和。

## # 逻辑门电路的功能（功能，外部特性）

TTL 集成逻辑门电路：

典型 TTL 与非门：三输入一输出 与非门

主要外部参数（特性）：

输出逻辑电平：

输出高电平：对于与非门，至少有一个是 0，他就输出高电平

输出低电平：对于与非门，只有全部是 1，才能输出低电平

开门电平和关门电平：

开门电平：指确保与非门输出为低电平的所允许的最小输入高电平。它表示使与非门开通的输入高电平最小值。

关门电平：指确保与非门输出为高电平时所允许的最大输入低电平，他表示使与非门关断的输入低电平最大值

扇入系数和扇出系数：

**扇入系数  $N_1$ ：**与非门允许的最大输入端数目

**输出系数  $N_0$ ：**与非门输出端连接同类门的最多个数，它反映了

**与非门的带负载能力**

平均传输时延：

指一个矩形波信号从与非门输入端传到与非门输出端所延迟的时间

与非门的功耗：指在空载条件下工作时所消耗的电功率

## # 两种特殊的门电路（OC 门，三态门）

### （一）**集电极开路门（OC 门）：**

集电极开路与非门：将两个与非门的输出端直接对接，实现逻辑与功能，此时并没有使用与门，而是由线与逻辑实现。

### （二）三态门：三态输出门（TS 门）

其输出有三种状态：高电平，低电平和**高阻状态**（相当于开路），前两种是工作状态，**后一种是禁止状态**。

为保证数据传送的正确性，在实现单项数据从=传输时， $N$  个三态门只有一个的输出能为 1，其余均为 0

# # 触发器：存储以为二进制信息的理想器件

1."0"状态：  $Q = 0, \quad \neg Q = 1$ ; "1"状态：  $Q = 1, \quad \neg Q = 0$ ;

次态是现态和输入的函数

2.与非门基本 R-S 触发器：

两个与非门交叉耦合形成，低电平有效

R 称为复位端，S 称为置位端；

以低电平有效为例：

$S = 1, R = 1$  此时相当于两个输入都是高电平无效，触发器保持原来的状态

$S = 0, R = 1$  此时 S 有效，次态  $Q=1$

$S = 1, R = 0$  此时 R 有效，次态  $Q=0$

约束条件：不允许  $S = R = 0$

次态方程：  $Q_2 = \neg S + R \cdot Q$ ;

激励表：状态从  $Q$  到  $Q_1$  时对输入的要求

利用 RS 触发器消除机械开关抖动：

原理：与非门构成的 R-S 触发器，当输入端连续出现多次置 1 信号，或者输入端连续输入多个清 0 信号时，只有第一个信号会改变触发器的状态，具体原因见真值表。

效果：避免阻尼振荡带来的干扰信号

## 1.2 或非门基本 R-S 触发器

与与非门 R-S 触发器相反，其两个输入端均为高电平

次态方程： $Q_2 = S + \neg R * Q$ ;

约束方程： $S * R = 0$ ;

# 我们所讲的几个钟控触发器均是与非门组成的

## 3. 时钟控制触发器

与触发器的功能是一样的，区别在于触发条件需要等待时钟电平。

钟控 R-S 触发器：

CLK = 0 不工作，触发器状态保持不变

CLK = 1:

R = S = 0, 状态不变

R = 0, S = 1, 置 1

R = 1, S = 0, 清 0

R = S = 1, 不允许

状态方程： $Q_1 = S + \neg R * Q$

钟控 D 触发器：

CLK = 0 不工作，触发器保持不变

CLK = 1:

D = 0, Q = 0

$$D = 1, \quad Q = 1$$

状态方程:  $Q1 = D$ ; 由于这种特性, D 触发器也被称作“锁存器”

钟控 J-K 触发器:

CLK = 0 不工作, 触发器状态保持不变

CLK = 1:

J = K = 0, 保持不变

J = 1, K = 0, 置 1

J = 0, K = 1, 清 0

J = K = 1, 相反

状态方程:  $Q1 = J * !Q + !K * Q$

钟控 T 触发器: 将 J,K 连起来

CLK = 0 不工作, 触发器保持不变

CLK = 1:

T = 0, 状态不变

T = 1, 状态相反

状态方程:  $Q1 = !T * Q + T * !Q$

#### 4.空翻现象

当触发器 CP = 1 时, 输入信号多次发生改变, 导致触发器状态跟着发生变



化, 导致了状态的不确定性和系统工作的混乱

## 5.主从钟控触发器

主从 R-S 触发器:

一般是下降沿触发

(1) 触发器的状态变化发生在 CP 由 1 变 0 的时候, 因为 CP=0 时触发器被封锁, 其状态不再受 R,S 的影响, 因此不会发生两次以上的偏转, 从而克服了空翻现象;

(2) 触发器的状态实际上取决于 CP 由 1 变 0 之前的 R,S 的值

(3) 次态方程:  $Q_1 = S + !R * Q;$

约束方程:  $R * S = 0;$

主从 J-K 触发器:

下降沿触发

其是对主从 R-S 触发器的改造, 使得  $R = K * Q, S = J * !Q;$

次态方程:  $Q_1 = J * !Q + !K * Q;$

其逻辑功能和简单结构和 J-K 触发器完全相同, 但他克服了空翻现象;

问题:

主从 J-K 触发器存在“一次翻转”现象:

在一个时钟脉冲作用期间, 主触发器的状态只能更改一次, 导致其如果受到干扰, 就不能恢复到正常工作状态。

解决: 采用“窄脉冲”

## 6. 维持-阻塞钟控触发器

一般是在上升沿触发

边沿触发器仅在时钟脉冲的上升沿或者下降沿时刻响应信号，从而大大提高了触发器的抗干扰能力。

不仅克服了空翻，而且由于是边沿触发，抗干扰能力强，因而应用十分广泛  
输入：

维持阻塞 D 触发器有时具有几个 D 输入端，其相互之间是相与的关系

$$D = D1 \& D2 \& D3;$$

## # 各触发器之间的相互转换

方法：列出两触发器的状态方程，进行待定系数法

# # 组合逻辑电路

组合逻辑电路的设计：

选取与非门：对与或表达式整体取两次非，再按照德摩根展开下面一层

非

选取或非门：对与或表达式的每一项取两次非，在按照德摩根拆开下面

## 一层

多个输出函数的设计：

为了使表达式尽可能简化，我们要考虑多个函数之间的公共项。可以用

卡诺图圈出相同的位置的大小的圈圈

竞争：门的级数，逻辑门的延迟，导线的长短等因素导致输入信号经过不同路径到达输出端的时间存在先后。

险象：由竞争导致的错误输出信号。

非临界竞争：不会导致错误的竞争

临界竞争：导致错误的竞争

如何判断险象：

- (1) 代数法：化简出  $A + !A$  的形式
- (2) 卡诺图，如果存在相切的卡诺圈，则存在险象

如何消除险象：

- (1) 增加冗余项
- (2) 加一个滤波电容---惯性时延
- (3) 选通法

## # 时序逻辑电路

输出函数表达式：

Mealy 型:  $Z = F(x, y)$

Moore 型:  $Z = F(y)$

激励函数表达式：

$Y = F(x, y)$

次态函数表达式：

$y_1 = F(Y, y)$       激励函数和现态的函数

## # 脉冲异步时序电路

输入脉冲信号的约束条件：

不允许在两个或两个以上输入端同时出现脉冲。

分析时：

脉冲信号也是激励函数；

先分析脉冲信号固定变化的那个触发器，后分析脉冲信号需要靠输出变

化判断的触发器；

## # 中规模集成电路及其应用

**模 N 计数器：指计数器有 N 中状态，并不是说是从 0 到 N-1**

串行进位二进制并行加法器：全加器级联而成，他的被加数和加数能够同时到达各位的输入端，但是进位却需要逐级传递。

全加器：

$$S = A \oplus B \oplus C;$$

$$C_1 = A \& B + (A \oplus B) * C;$$

由此我们发现：

当  $A \& B = 1$  时，无论上一个进位如何，必定产生进位，因此我们定义  $G = A \& B$  为进位产生函数；

而当  $A \oplus B = 1$  时，上位的进位能够直接传递到本位，因此我们称  $P = A \oplus B$  为进位传递函数；

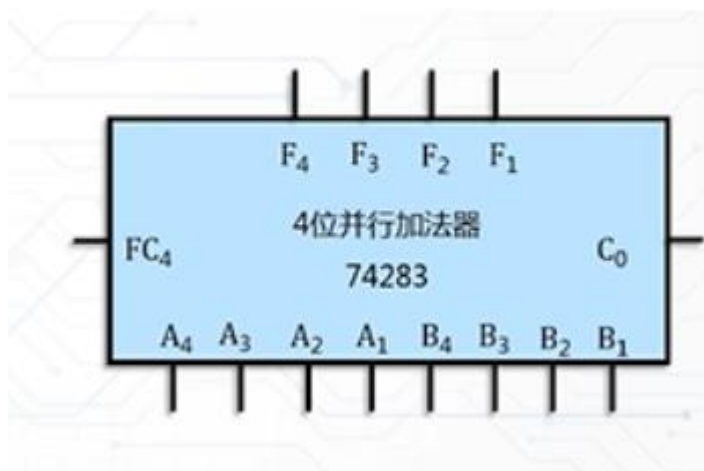
将这两个函数带入全加器，不难发现：

$$F_i = A_i \oplus B_i \oplus C_{i-1} = P_i \oplus C_{i-1}$$

$$C_i = A_i \& B_i + (A_i \oplus B_i) * C_{i-1} = G + P * C_{i-1}$$

然后将进位函数分别带到每一位，即令  $i = 1, 2, 3 \dots$  就可以得到每一位的进位表达式，实现超前进位加法器

常见的并行加法器：74283 其有 16 条引脚， $A_1, A_2, A_3, A_4$   $B_1, B_2, B_3, B_4$  是两个四位二进制加数， $F_4, F_3, F_2, F_1$  是四位和， $C_0$  是最低位进位输入（可用作级联）， $F_4$  是最高位进位输出。



设计：

- (1) 四位二进制并行加法器实现 8421 码转余三码

思路：余三码 = 8421 + 0011

- (2) 四位二进制并行加法器实现四位二进制并行加/减法器

思路：如下图：

数 A 直接加到并行加法器的  $A_4, A_3, A_2, A_1$  输入端, 4 位二进制数 B 通过加法器的  $B_4, B_3, B_2, B_1$  输入端。将功能选择变量 M 作为异或门的另一个输入器的  $C_0$  进位输入端。使之, 当  $M=0$  时,  $C_0=0, b_i \oplus M = b_i \oplus 0 = b_i$ , 加法: 1 时,  $C_0=1, b_i \oplus M = b_i \oplus 1 = \bar{b}_i$ , 加法器实现  $A + \bar{B} + 1$ , 即  $A - B$ 。其逻辑电

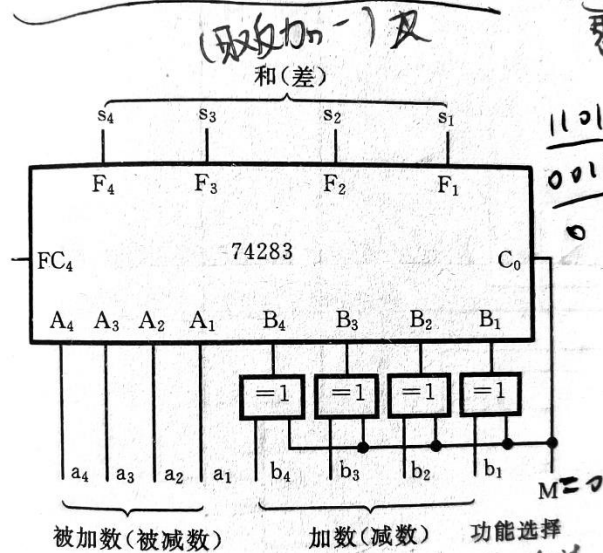


图 7.4 逻辑电路

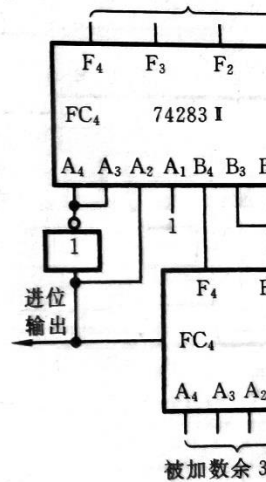


图 7.5 逻辑

例 7.3 用 4 位二进制并行加法器设计一个用余 3 码表示的 1 位十进制数相加。根据余 3 码的特点, 两个余 3 码表示的十进制数相加时, 需要解

我们增加一个功能输入 M, 再加上四个异或门:

当  $M=0$  时,  $b_i \wedge 0 = b_i$ , 此时进位  $C_0 = 0$ , 实现加法

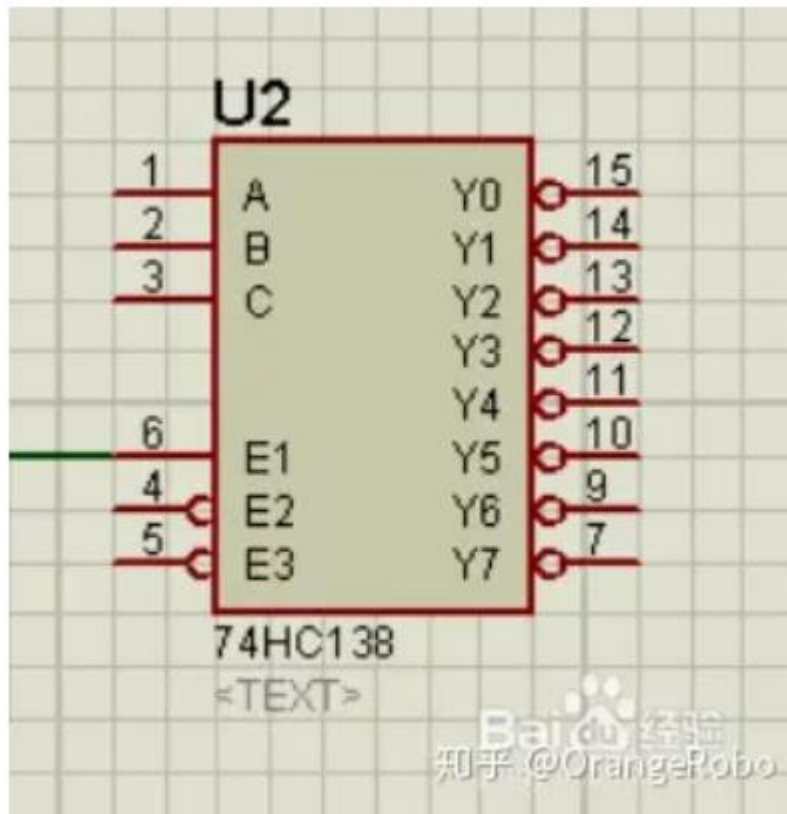
当  $M=1$  时,  $b_i \wedge 1 = \bar{b}_i$ , 此时进位  $C_0 = 1$ , 结果是  $F = A + \bar{B} + 1 = A$

- B;

## # 译码器和编码器

译码是一个少对多的过程; 编码是一个多对少的过程

常见 74138 译码器：



A,B,C 是三个输入端，Y0,...Y7 是 8 个输出端；

E1,E2,E3 是三个使能端，只有同时有效才能译码

设计：用 3-8 译码器 74138 和与非门实现全减器

思路：和全加器类似，写出其本位差和借位，列出真值表，然后取出最小项化为与非表达式即可。

## (2) 2-10 进制译码器

将 BCD 码转换为十进制输出，类似于 74138

7442 是将 8421 码转换成十进制



### (3) 七段显示译码器

就是七段数码管对应 abcdefg, 当四位二进制输入时, 对应点亮即可。

## # 编码器

### (1) 二-十进制编码器

74147 十进制-BCD 码, 加上使用输出标志是为了区分按下 I0 和全都不按都输出 0000 的两种状态; 每个按键不按时输入为 1, 按下输入为 0:

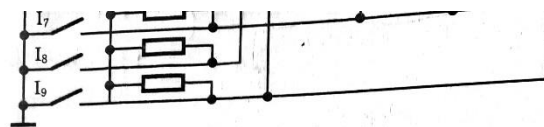


图 7.13 按键式 8421 码编码器

表 7.5 8421 码编码器真值表

输 入										输 出				
I <sub>9</sub>	I <sub>8</sub>	I <sub>7</sub>	I <sub>6</sub>	I <sub>5</sub>	I <sub>4</sub>	I <sub>3</sub>	I <sub>2</sub>	I <sub>1</sub>	I <sub>0</sub>	A	B	C	D	S
1	1	1	1	1	1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	1
1	1	1	1	1	1	1	1	0	1	0	0	0	1	1
1	1	1	1	1	1	1	0	1	1	0	0	1	0	1
1	1	1	1	1	1	0	1	1	1	0	0	1	1	1
1	1	1	1	1	0	1	1	1	1	0	1	0	0	1
1	1	1	1	0	1	1	1	1	1	0	1	0	1	1
1	1	1	0	1	1	1	1	1	1	0	1	1	1	1
1	1	0	1	1	1	1	1	1	1	1	0	1	1	1
1	0	1	1	1	1	1	1	1	1	1	0	0	0	1
0	1	1	1	1	1	1	1	1	1	1	0	0	1	1

常用的十进制-BCD 码编码器有中规模集成电路芯片 74147 等, 有关详细介绍可查阅集成电路手册。

### (2) 优先编码器

优先编码器是数字系统中实现优先权管理的一个重要逻辑部件。它与上述二-十进制

### (2) 优先编码器

区别在于可以同时有多个输入, 但是存在优先级的关系, 优先对等级高的输入进行编码。

## # 多路选择器

输入 2 的 N 次方路信号，通过 N 个控制信号选出其中的一个信号。

### 74153 多路选择器

用法：

(1) 选择数据

(2) 用具有 N 个控制变量的 MUX 实现 m 个变量的函数：

$m=n$ : 直接将 m 个变量输入到多路器的 n 个变量选择端，

$m=n+1$ , 任选 n 个接入输入端，则输入数据  $D_0, D_1, D_2, D_3$  只能是 0,

1, X, ! X, 如下：

图 7.17 例 7.8 的两种方案

案 2 采用 4 路数据选择器实现

定采用 4 路 MUX 74153 实现，由于 4 路 MUX 具有 2 个选择控制变量，所以实现函数功能时，应该首先从函数的 3 个变量中任选 2 个作为选择控制变量，然后将第 3 个变量作为数据输入。假定选 A、B 与选择控制端  $A_1, A_0$  相连，则可将函数 F 的表达式：

$$\begin{aligned} F(A, B, C) &= \overline{A}\overline{B}\overline{C} + \overline{A}B\overline{C} + A\overline{B}\overline{C} + ABC \\ &= \overline{A}\overline{B} \cdot 0 + \overline{A}B \cdot (\overline{C} + C) + A\overline{B} \cdot C + AB \cdot \overline{C} \\ &= \overline{A}\overline{B} \cdot 0 + \overline{A}B \cdot 1 + A\overline{B} \cdot C + AB \cdot \overline{C} \end{aligned}$$

然，要使 4 路 MUX 的输出 Y 与函数 F 相等，只需  $D_0 = 0, D_1 = 1, D_2 = C, D_3 = \overline{C}$ 。用 4 路 MUX 实现给定函数功能的逻辑电路，如图 7.17(b) 所示。类似地，若选 A、C 或 B、C 作为选择控制变量，选择控制变量不同，数据输入也不同。例如选 A、C 相连，则应有  $D_0 = B, D_1 = \overline{B}, D_2 = B, D_3 = \overline{B}$ 。

由两种方案表明：用具有 n 个选择控制变量的 MUX 实现 n 个变量的函数或函数时，无须任何辅助电路，可由 MUX 直接实现。

$M \geq n + 2$ ，借助逻辑门构造，和上面的情况差不多

## # 多路分配器

与多路器相反，是单输入，多输出的逻辑部件

功能：控制输入从哪一端输出

## # 集成计数器

4 位二进制同步可逆计数器 74193 的功能，引脚 P202

构成模大于、小于 16 的计数器：合理利用清零端和置位端。

## # 集成异步计数器

二-五-十进制加法计数器 74290，P204

## # 继承寄存器

74194 四位双向移位寄存器：

# # 可编程逻辑器件

1. PLD（全称：Programmable Logic Device）可编程逻辑器件，早期主要包括：

可编程只读存储器（Programmable Read Only Memory）-> PROM

可编程逻辑阵列（Programmable Logic Array）-> PLA

可编程阵列逻辑（Programmable Logic Array）-> PAL

他们三者都是由一个与阵列和一个或阵列组成，区别在于：

**PROM 的与阵列是固定的，而或阵列是可编程的**

PLA 的与阵列和或阵列都可以编程

PAL 的与阵列可编程，或阵列固定

2. PLD 的一般结构：输入电路，与阵列，或阵列，输出电路；

3. PLD 的电路表示方法：实点“·”表示固定连接；“X”表示可编程连接；没有则表示不连接

4. PLD 的分类：低密度可编程器件：集成度小于 1000 门： PROM PLA PAL GAL

高密度可编程器件：集成度大于 1000 门： CPLD FPGA

## # 时序逻辑电路设计难点

### # 利用隐含表进行状态化简

横格：从左往右是第一个状态到倒数第二个状态

纵格：从上向下是第二个状态到最后一个状态

寻找等效对：

首先进行顺序比较：按照从上向下，从左向右的顺序进行比对，如果两种状态是等效的，就打上对号，如果是明确不等效的，就打上错号，如果是和其他状态相关联的，就写上相关联的状态号。

其次进行关联比较：所谓关联比较，就是在顺序比较之后尚未确定是否等效的状态。进行关联比较时，首先应该看待检查的状态对的次态对是否等效，只要有一个次态对不等效，就不能等效。在方格中加“/”，若是等效的，就不用加任何处理。

求出最大等效对----->做出状态表

### # 状态编码：

- (1) 在相同输入条件下，具有相同次态的现态分配相邻的二进制编码
- (2) 在相邻输入条件下，同一现态的次态分配相邻的二进制编码
- (3) 输出完全相同的现态分配相邻的二进制编码

得到相邻关系后，用卡诺图进行编码

# 集成电路分类：CMOS 和 TTL

# 检查电路自启动：

如果不存在无关状态，则状态机都是可以正常工作的

如果存在无关状态，我们需要将无关状态的状态取值带入状态方程，看看能不能得到正常状态，能的话就可以正常自启动，否则就不行。