

# 数字逻辑与数字系统

---

## 数字逻辑与数字系统

### 第一章 开关理论基础

#### 数制与码制

##### BCD码

##### 有权BCD码

##### 8421码

##### 2421码

##### 无权码 (余3码)

##### 可靠性编码

##### 格雷码

##### 奇偶校验码

#### 逻辑函数及其描述工具

### 第二章 组合逻辑

### 第三章 时序逻辑

#### 双稳态触发器 (Flip-Flop)

##### RS触发器

##### 电平 (同步) D 触发器(锁存器)

##### 边沿D触发器 (寄存器)

##### 负边沿JK触发器

##### 上升沿 T 触发器

#### 时序电路的分析

##### 时序电路的特点

##### 同步时序电路的分析方法

#### 锁存器和寄存器

##### 锁存器

##### 寄存器

##### 移位寄存器

##### 移存型计数器

##### 环型计数器 (Ring Counter)

##### 扭环形计数器 (Johnson Counter)

##### 中规模 8位双向移位寄存器 - 74LS299

#### 中规模计数器 (Counter)

##### 集成同步二进制计数器

##### 四位二进制加计数器 (74LS161/163)

##### 容量扩展

##### 同步二进制可逆计数器 (74191)

##### 集成同步十进制计数器

##### 用中规模计数器构成任意进制计数器

##### 清零法

##### 计数器应用

##### 顺序脉冲 (节拍) 发生器

##### 序列信号发生器

#### 同步时序电路的设计

#### 附

## 第一章 开关理论基础

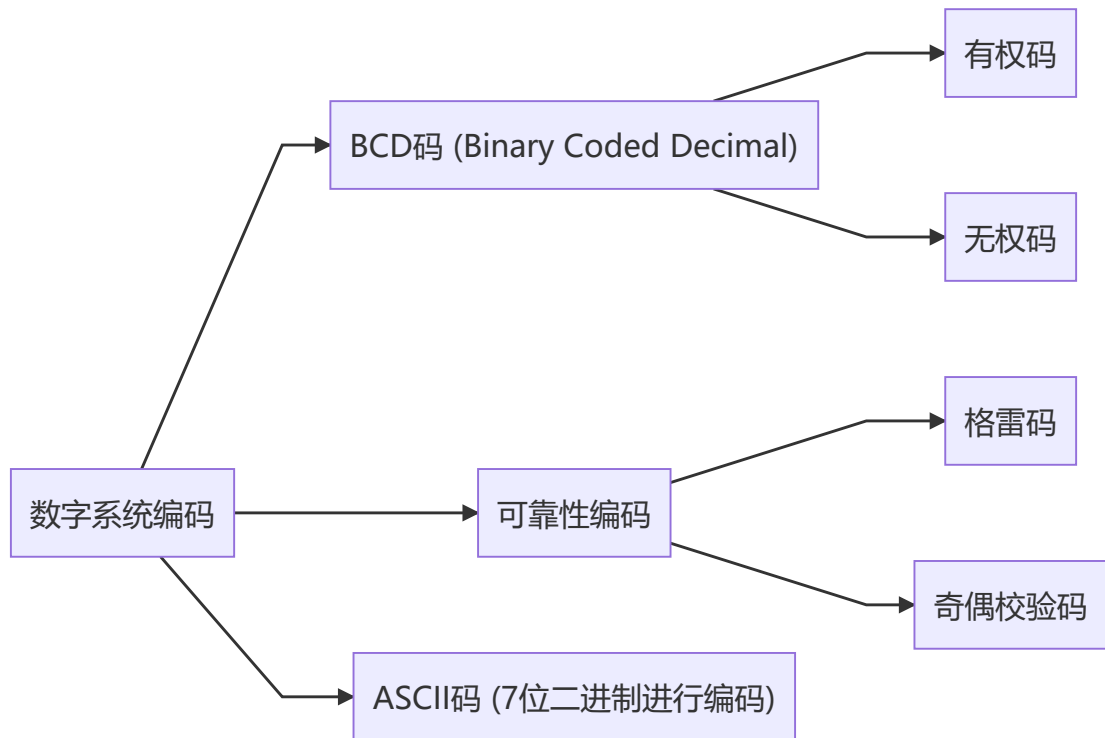
---

# 数制与码制

- 对N项信息编码，要求二进制代码位数n满足

$$2^n \geq N$$

- 常用编码：



## BCD码

### 有权BCD码

#### 8421码

- 定义：用0000 ~ 1001代表0 ~ 9，1010 ~ 1111冗余。

Decimal	Binary	BCD(8421)
0	0000	0000
1	0001	0001
2	0010	0010
3	0011	0011
4	0100	0100
5	0101	0101
6	0110	0110
7	0111	0111
8	1000	1000
9	1001	1001

- 8421为4位编码的权重，例：3的8421码为0011，那么有 $0 \times 8 + 0 \times 4 + 2 \times 1 + 1 \times 1 = 3$
- 对于多位数，直接按位转换，eg:  $(28)_{10} = (00101000)_{8421}$
- 运算如果出现进位或算出冗余码，需要加6(0110)

eg:

$$8 + 9 = 17$$

$$1000 + 1001 = 10001$$

计算中出现进位，需要加6

$$10001 + 0111 = 10111 = 00010111$$

$$(17)_{10} = (00010111)_{8421}$$

## 2421码

- 权值从高到低为2, 4, 2, 1
- 不允许出现0101 ~ 1010 的6个编码
- 取9的自补码，因此执行十进制数相加能正确产生进位信号

Decimal	2421码
0	0000
1	0001
2	0010
3	0011
4	0100
5	1011
6	1100
7	1101
8	1110
9	1111

**无权码（余3码）**

- 将8421每个码加0011即可的余3码
- 冗余码为：0000、0001、0010、1101、1110、1111
- 也是9的自补码

Decimal	余3码
0	0011
1	0100
2	0101
3	0110
4	0111
5	1000
6	1001
7	1010
8	1011
9	1100

**可靠性编码**

为减少错误的产生，或者能检测出错误 的发生，所设定的码制。

格雷码

- 性质：相邻两个编码只有一位不同，避免编码按顺序增加时多位变化产生的传输错误
- 公式：

$$\begin{cases} G_{n-1} = B_{m-1} \\ G_i = B_i \oplus B_{i+1} \end{cases}$$

eg: 将1101转化为格雷码

最高位1不变，第三位：1 ⊕ 1 = 0，第二位1 ⊕ 0 = 1，第一位0 ⊕ 1 = 1，则B: 1101，G: 1011

Decimal	Binary	Gray Code
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111

奇偶校验码

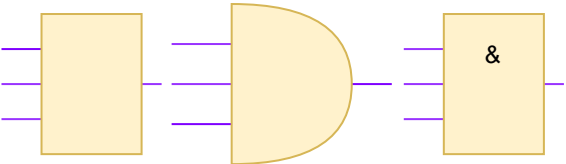
- 由两部分组成：信息位和检验位
- 编码方式：奇校验（信息位中1的个数为奇数，校验位为0）、奇校验（信息位中1的个数为偶数，校验位为0)
- 特点
  1. 编码简单
  2. 只有检错能力，没有纠错能力
  3. 只能发现单错，不能发现双错

逻辑函数及其描述工具

1. 与运算（逻辑乘）

函数式：  $F = A \cdot B \cdot C$

逻辑图

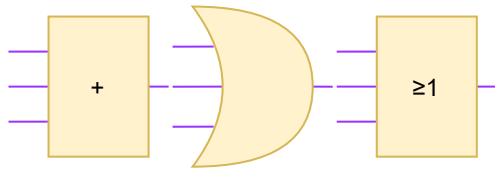


VHDL: `F <= A and B and C`

## 2. 或运算

函数:  $F = A + B + C$

逻辑图

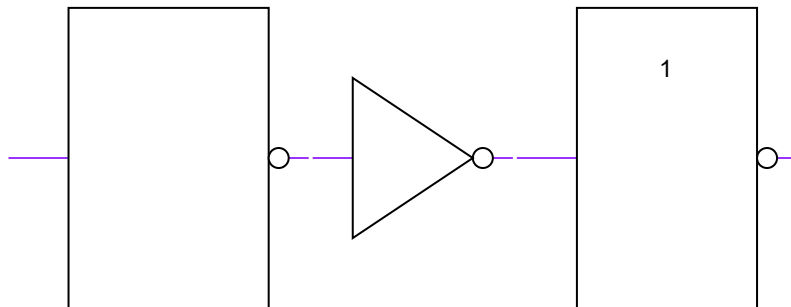


VHDL: `F <= A or B or C`

## 3. 非运算

函数:  $F = \overline{A}$

逻辑图



VHDL: `F <= not A`

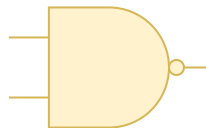
## 4. 复合逻辑运算

### 1. 与非

$F = \overline{AB}$

`F <= A nand B`

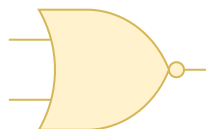
逻辑图



### 2. 或非

$F = \overline{A + B}$

`F <= A nor B`



### 3. 与或非运算

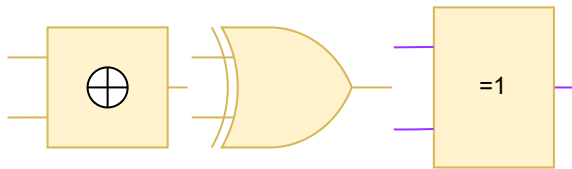
$F = \overline{AB + CD}$

### 4. 异或

$F = A \oplus B = \overline{A}B + A\overline{B}$

`F <= A xor B`

逻辑图



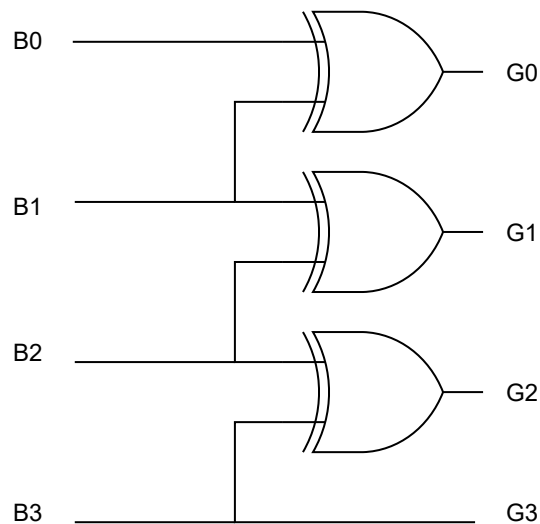
输入变量有奇数个1，则输出1，否则输出0，可以做奇偶校验

同时对于异或，其中一个输入C作为控制端，输出  $F = C \oplus A$

1. 当  $C = 0$  时,  $F = A$
2. 当  $C = 1$  时,  $F = \overline{A}$

可控制输出是否取反

下图为二进制码转格雷码



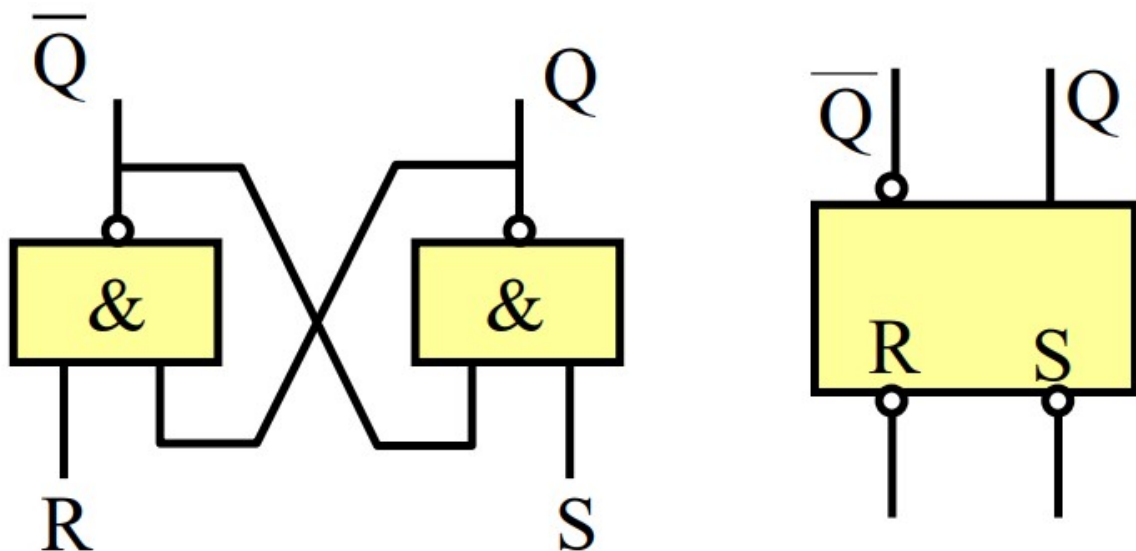
## 第二章 组合逻辑

没整

## 第三章 时序逻辑

### 双稳态触发器 (Flip-Flop)

RS触发器



- 低有效RS触发器：R、S同时为0称为不定状态（实际不定只会出现在RS=00转化到保持，即RS=11的时候才会出现）
- 次态方程（特征方程）

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

$$S + R = 1 \quad \text{约束条件}$$

$Q^n$	$Q^{n+1}$	R	S
0	0	x	1
0	1	1	0
1	0	0	1
1	1	1	x

- 交叉耦合的与非门，S和Q在同一个门对应
- 高有效RS触发器：类比低有效
- 次态方程：

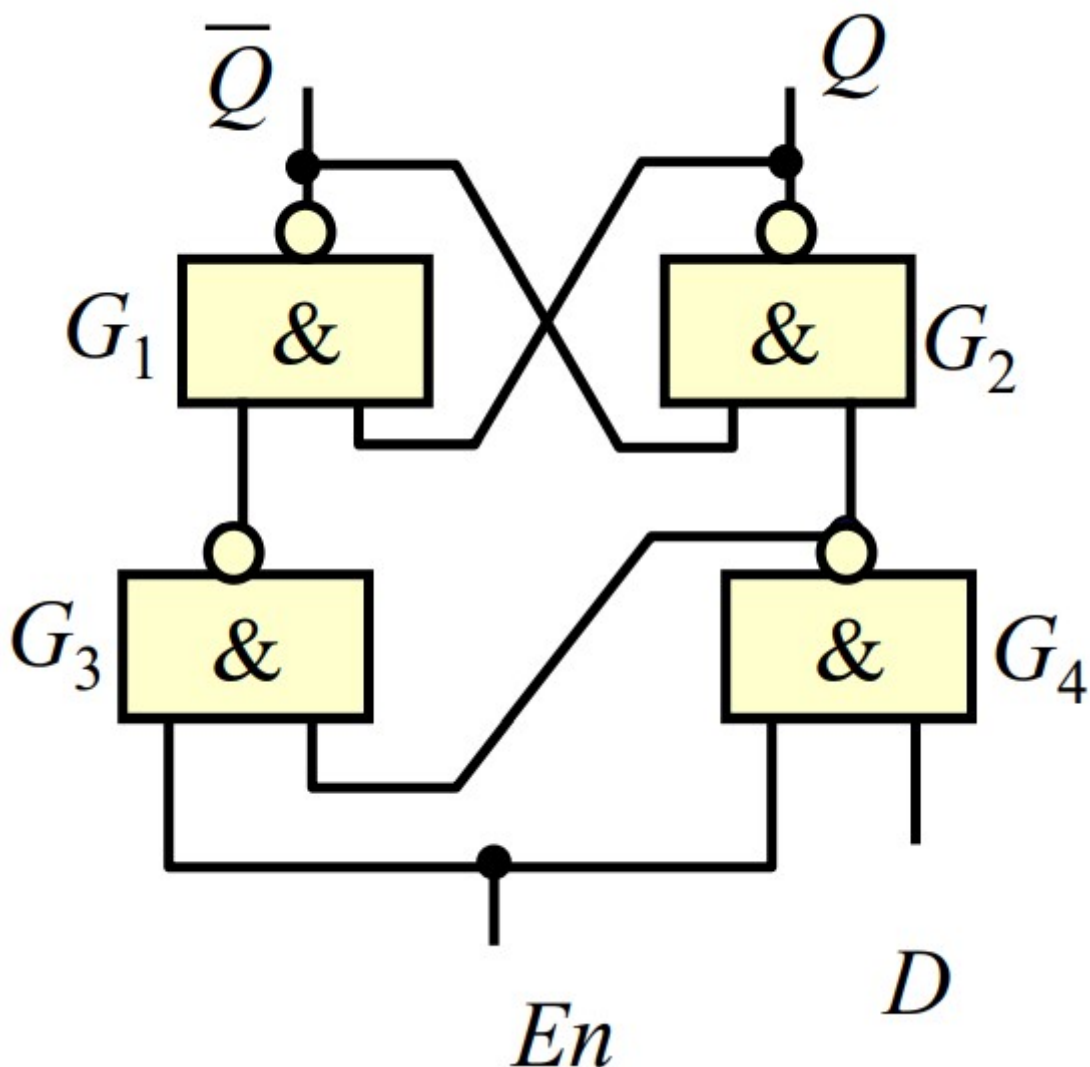
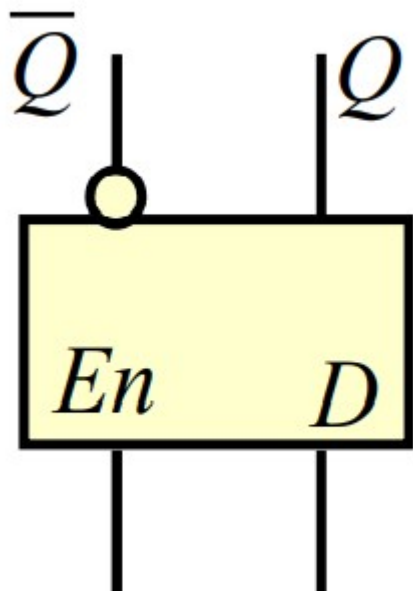
$$Q^{n+1} = S + \overline{R}Q^n$$

$$S \cdot R = 0 \quad \text{约束条件}$$

- 交叉耦合的或非门，S和 $\overline{Q}$ 在同一个门对应
- 问题：有不定状态(RS有约束)、不能定时控制

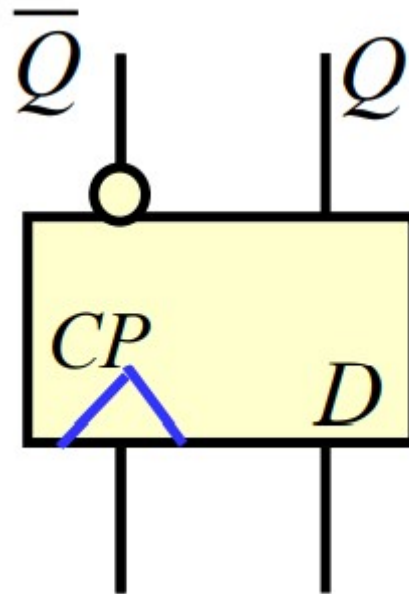
## 电平（同步）D 触发器(锁存器)





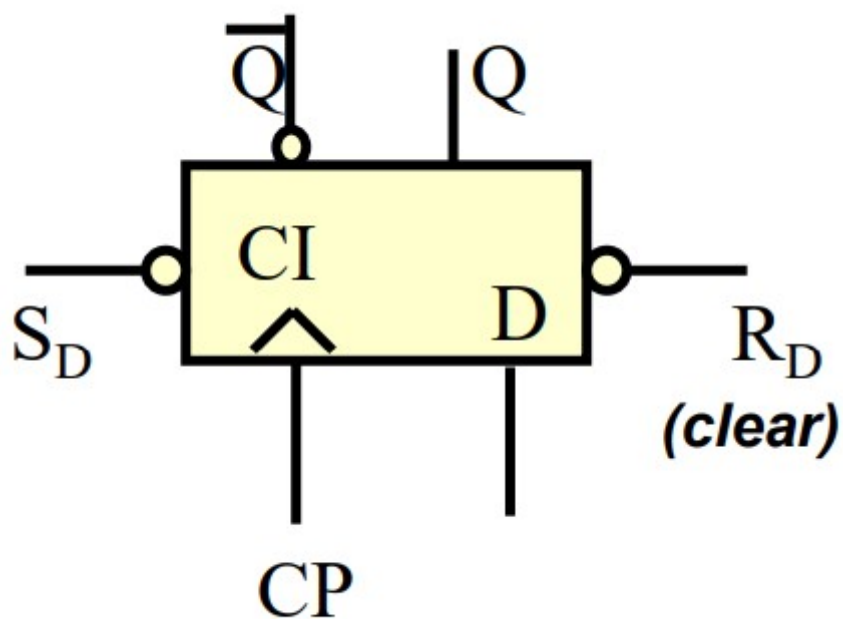
- $En = 0$ 时，维持原状态。 $En = 1$ 时，**次态方程**： $Q^{n+1} = D$
- 在RS触发器前加入两个与非门，防止RS同时有效出现不定状态
- 空翻：控制信号有效期间内，输出反转多于一次（出现空翻一般认为出现了干扰）
- 问题：有空翻现象、逻辑功能简单

## 边沿D触发器（寄存器）



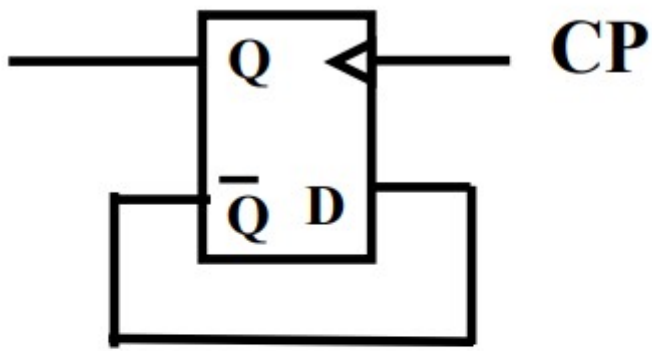
- 相对于锁存器，改电平使能为边沿触发
- 采用上升沿触发。触发器的次态取决于cp边沿前一瞬间的输入
- 双D边沿触发器，多出 $R_d$   $S_d$  功能如下

$R_d$	$S_d$	功能
1	1	工作
1	0	置位
0	1	清零
0	0	不定

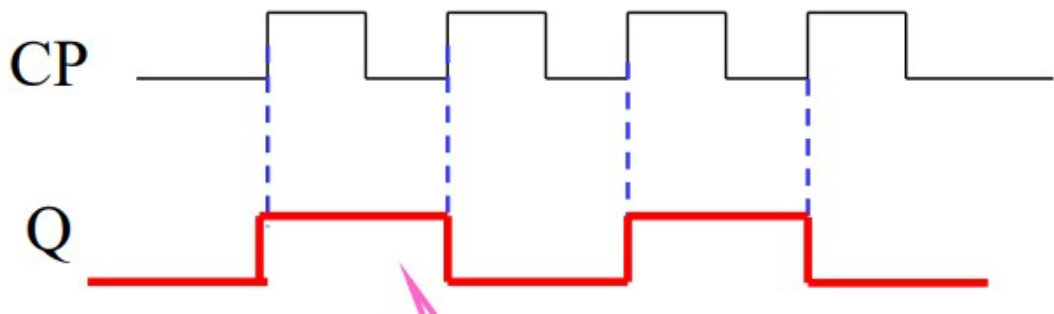


- 可以看出 $R_d$   $S_d$  优先级高，称为异步，不需要时钟配合

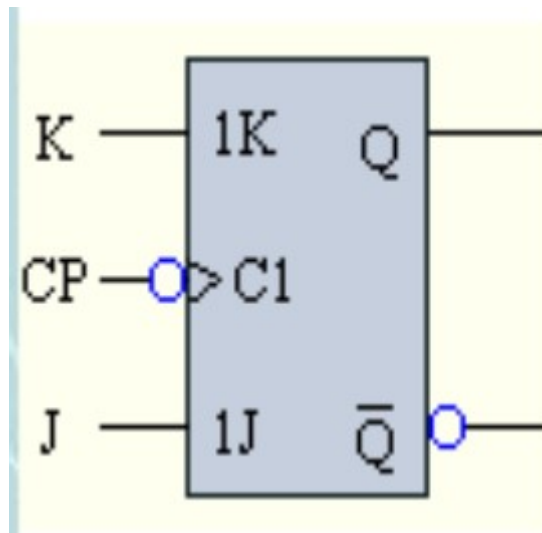
- 使用：二进制计数（或二分频）



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



### 负边沿JK触发器

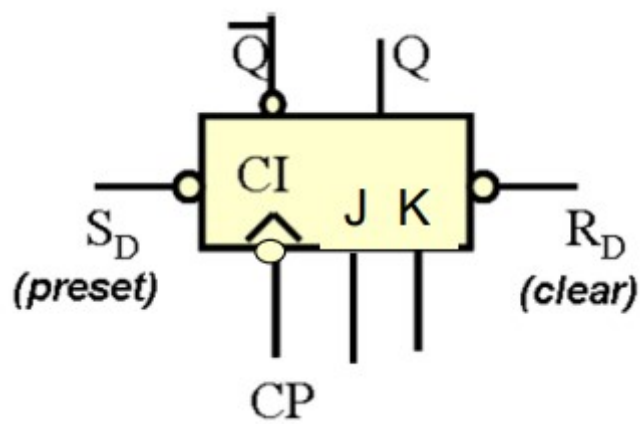


- 状态方程

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

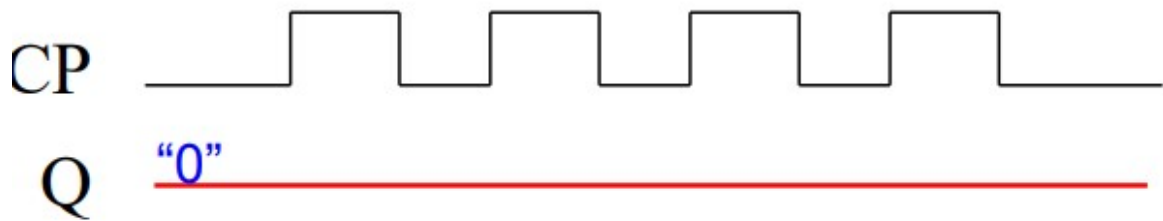
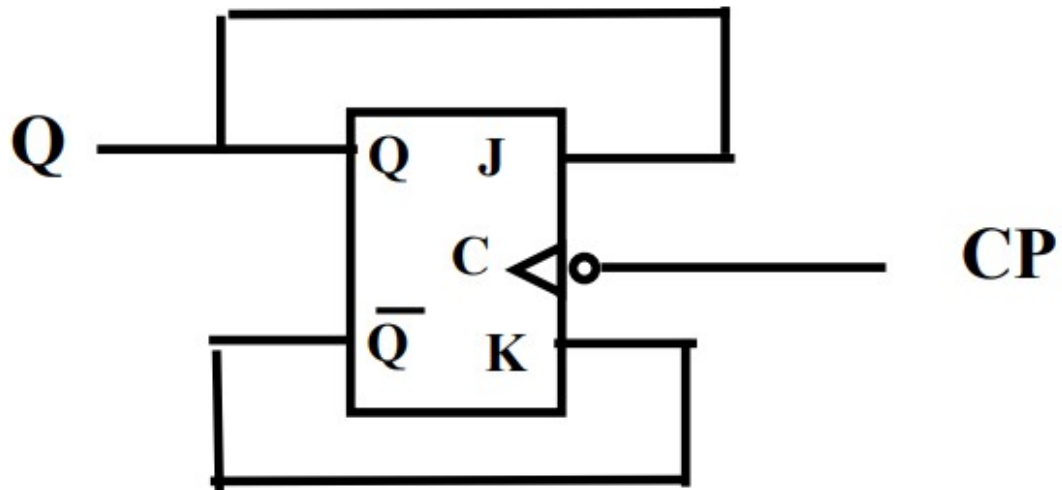
- 功能：J = K = 0 具有维持功能，J = K = 1 具有状态翻转功能（来一个时钟(CP) 下降沿，翻转一次），JK不等，不管现态如何次态与相同  
状态表：

J	K	$Q^n$	$Q^{n+1}$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0



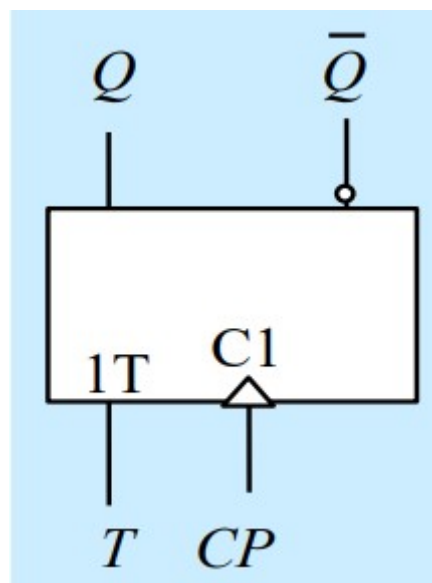
- 和[边沿触发器](#)类似，有清零和置位

使用例子：



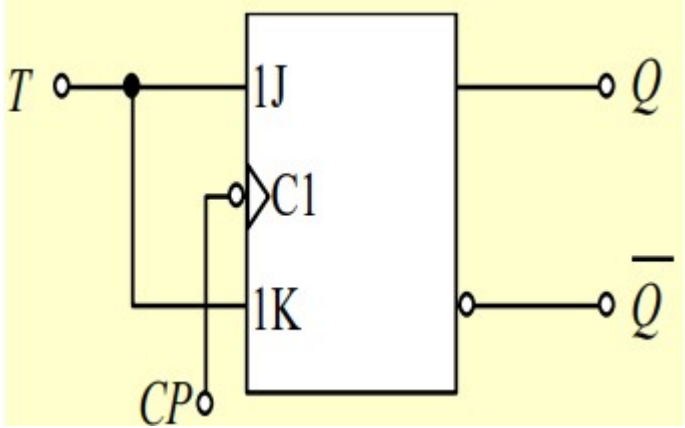
构成

上升沿 T 触发器

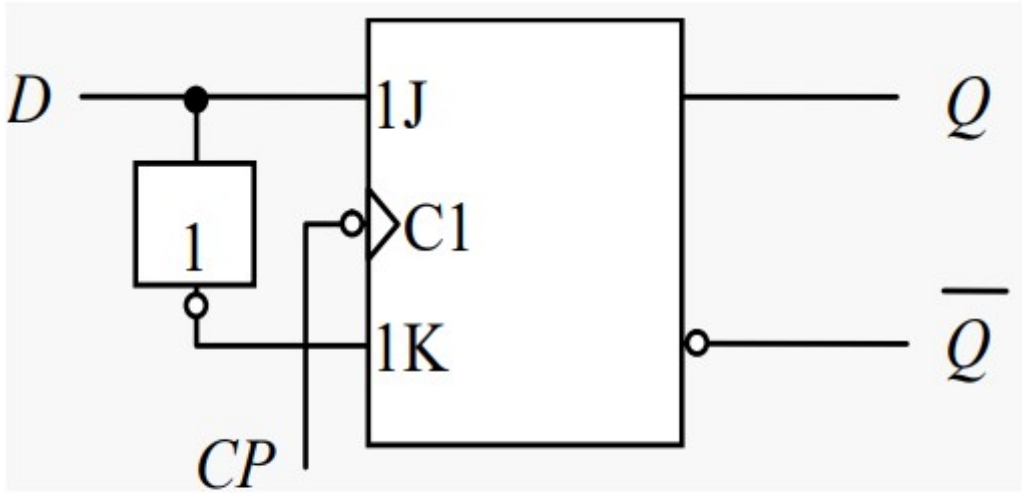


- 逻辑:  $Q^{n+1} = T\bar{Q}^n + \bar{T}Q^n = T \oplus Q^n$

- 对比JK，把JK的J和K并在一起接T（但触发不同，一个上升沿触发，一个下降沿）



类似的jk还可变为D触发器



JK  $\rightarrow$  D

- $T=0$  保持， $T=1$ 触发器在CP作用下翻转

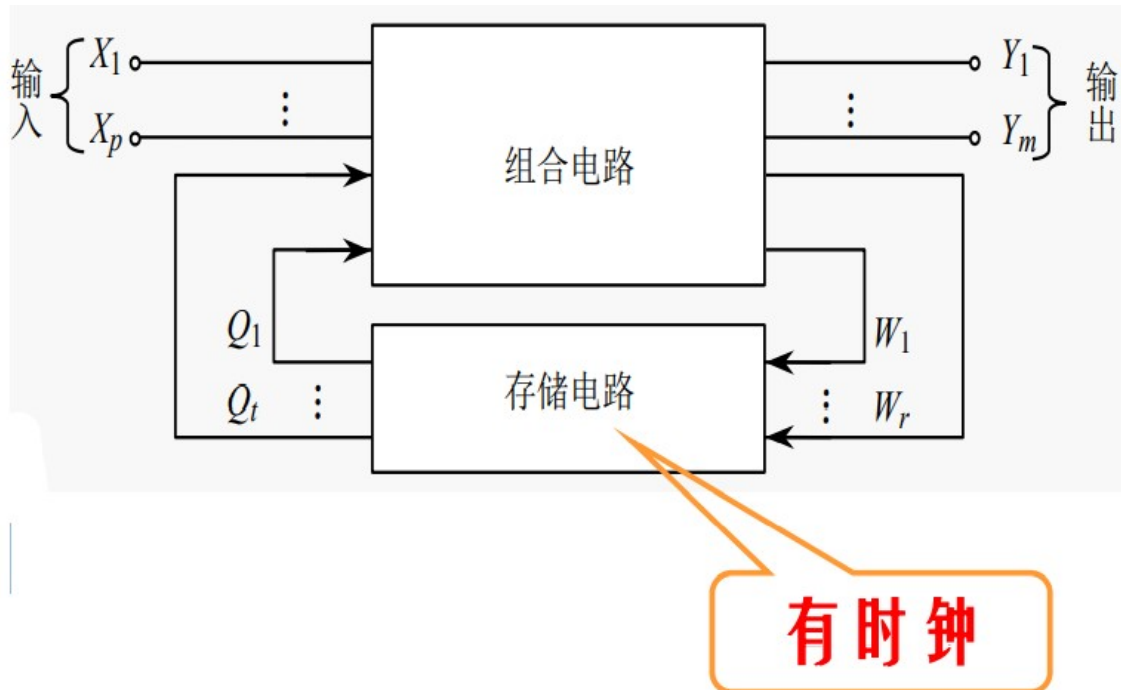
$T$	$Q^n$	$Q^{n+1}$
0	0	0
0	1	1
1	0	1
1	1	0

# 时序电路的分析

## 时序电路的特点

- 逻辑功能：**任一时刻的输出状态，不仅取决于当时的输入信号，而且与前一时刻电路的状态有关。

- **电路结构：**组合电路 + 触发器。



- 触发方式分类：
  - 同步时序电路：所有触发器共用一个时钟信号。
  - 异步时序电路：触发器无统一的时钟信号。
- 输出方式分类：
  - 莫尔型 (Moore)：输出 Y 仅与电路的现态有关 (式中无 x)
  - 米勒型 (Mealy)：输出 Y 是电路现态和外部输入量的函数。
- 时序电路可以用三个方程描述：

$$\begin{array}{ll}
 Y = F_1(X, Q^n) & \text{输出方程} \\
 W = F_3(X, Q^n) & \text{激励方程} \\
 Q^{n+1} = F_2(W, Q^n) & \text{状态方程}
 \end{array}$$

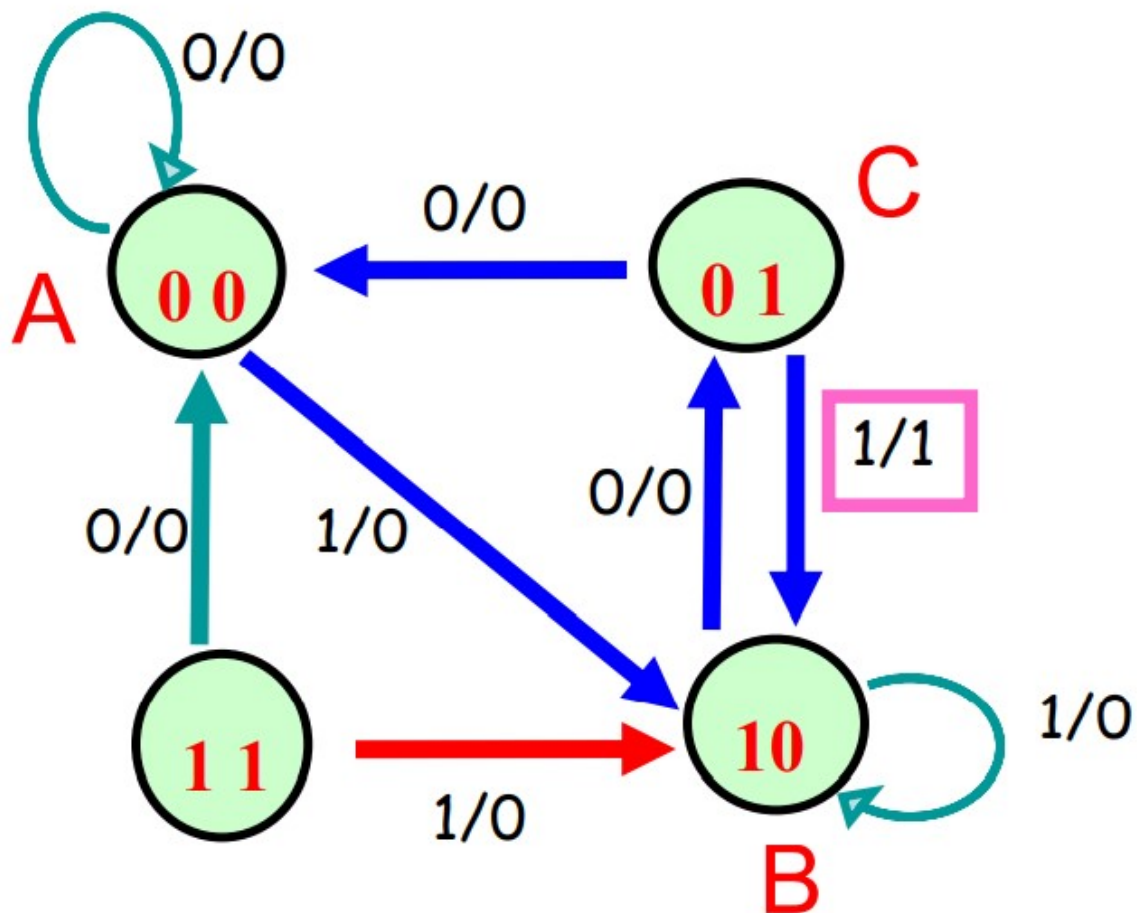
## 同步时序电路的分析方法

- 目的：得到电路状态、外部输出的变化规律，逻辑功能。
- 步骤
  1. 确定输入、输出信号
  2. 输出方程、驱动方程、状态方程
  3. 计算真值表
  4. 状态图
  5. 确定电路逻辑功能
- 状态图：
  - 主循环：状态图中状态多的循环
  - 有效状态：主循环中的状态。
  - 无效状态：不在主循环中的状态。
  - 自启动：无效状态在 CP 脉冲作用下能进入主循环。
  - 箭头旁 1/0 意义 (具体见 [状态图](#))：输入 1，输出 0
- 如果上电后只能在无效循环里称电路为**挂起**

- 时序图（波形图）：在CP和外部输入的作用下，电路状态、输出随时间变化的波形图。

$X$	$Q_2^n$	$Q_1^n$	$Q_2^{n+1}$	$Q_1^{n+1}$	$Z$
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	1	0
0	1	1	0	0	0
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	0	0
1	1	1	1	0	0

状态图



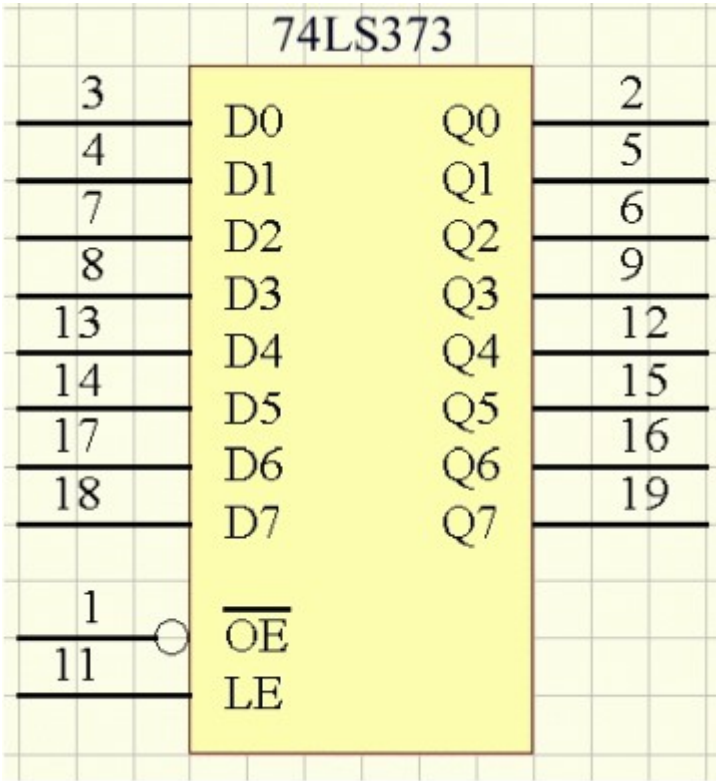
- 功能描述：序列信号检测器，该电路可重叠检测序列101(X输入依次为1、0、1后会被检测)
- 可重叠：如果输入1010101，会输出3个1，其中输入的第二、第三个1被重复使用检测



# 锁存器和寄存器

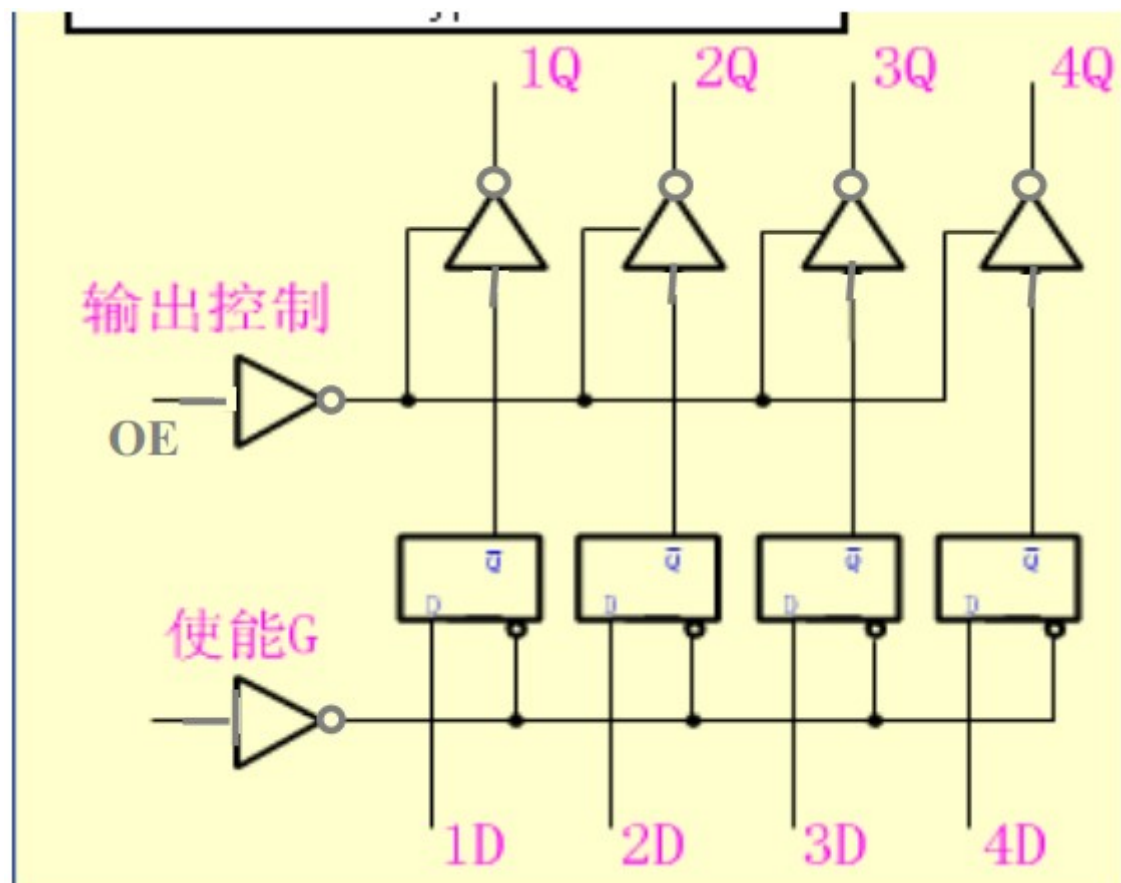
## 锁存器

- 用来暂时存放二进制代码或数据的电路。
- 典型芯片：集成三态输出8位锁存器74LS373

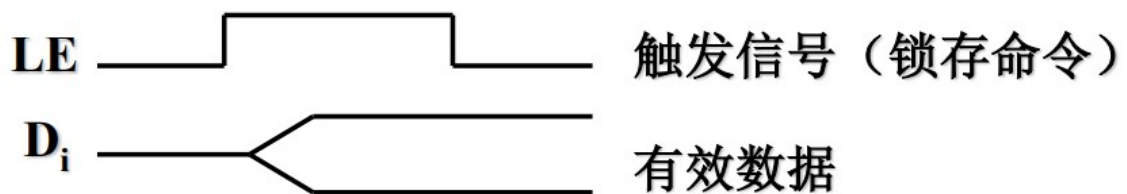


OE	LE	D(输入)	Q(输出)
0	1	D7~D0	D7~D0
0	0	X	输出保持
1	X	X	高阻态

- 引脚功能
  - D：数据输入
  - G：输入使能
  - Q(LE)：数据输出
  - OE：输出使能

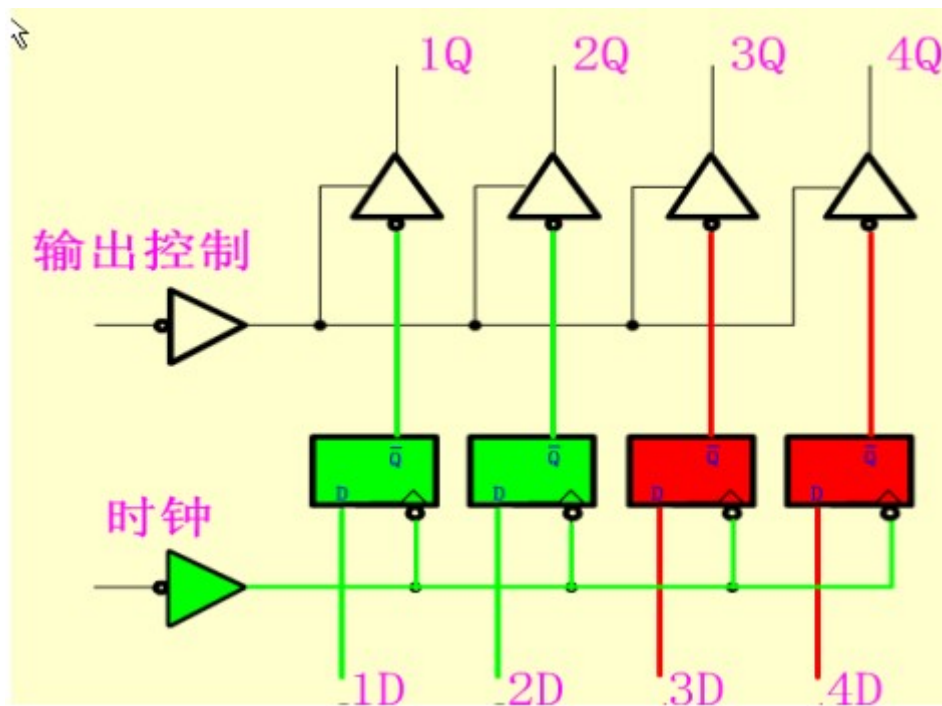


- 锁存器使用：输入有效数据的稳定滞后于锁存信号（门先打开，在打开的时间段内，输入信号达到稳定即可）

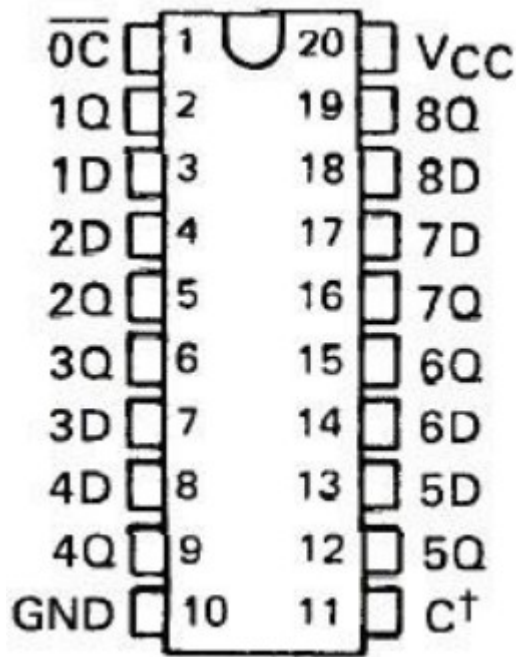


## 寄存器

- 用来暂时存放数据或代码

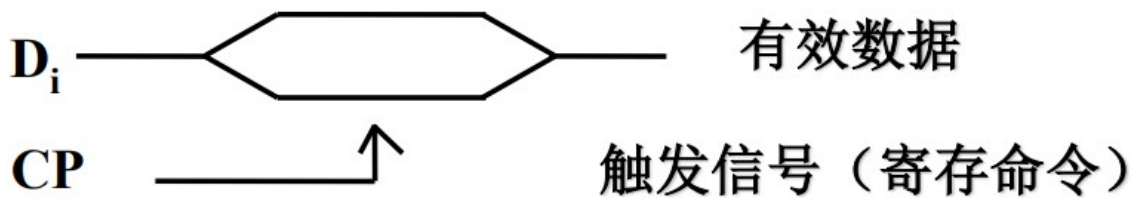


- 典型芯片：集成三态输出8位寄存器74LS374

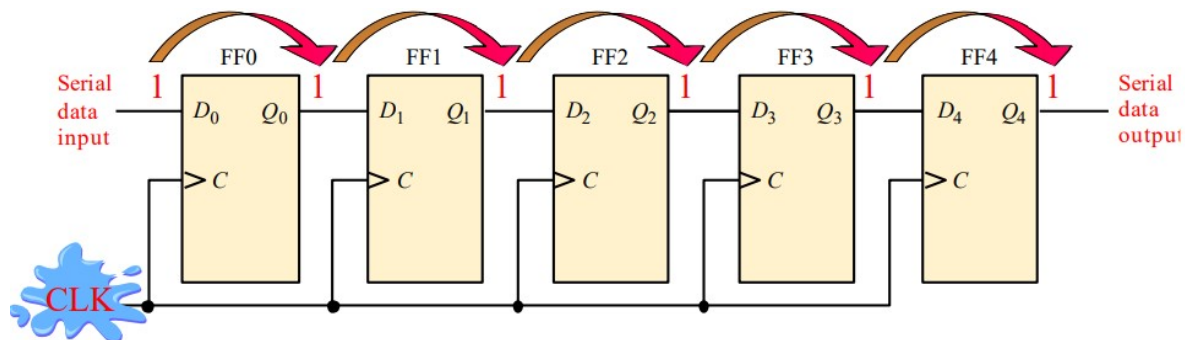


OC	CP	D(输入)	Q(输出)
0	↑	D7~D0	D7~D0
0	<del>↑</del>	X	输出保持
1	X	X	高阻态

- 寄存器使用：输入的有效数据的稳定先于打入脉冲



## 移位寄存器



- 在CP作用下，将所存的代码移至紧邻的左一位或右一位

$$Q_0^{n+1} = D \quad Q_1^{n+1} = Q_0^n \quad Q_2^{n+1} = Q_1^n \quad Q_3^{n+1} = Q_2^n$$

- 可用于串并转换
- 双四位移位寄存器CC4015

CC4015(1/2)			
15	D	Q0	13
		Q1	12
1		Q2	11
14	CLK	Q3	2
	RST		

D: 串行输入

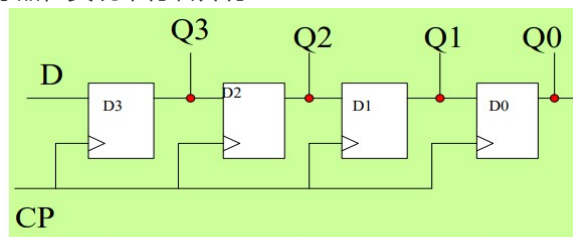
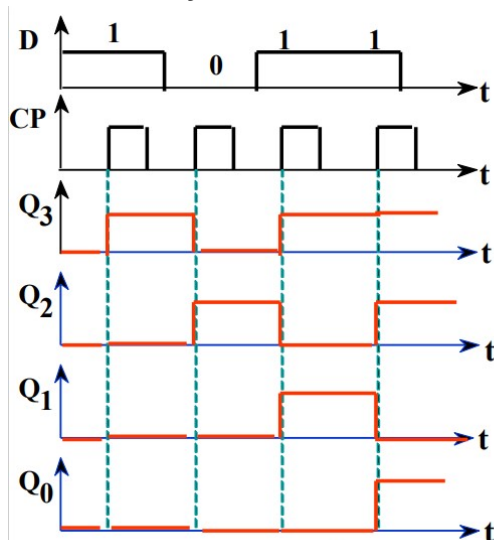
CLK: 时钟

RST: 异步清零

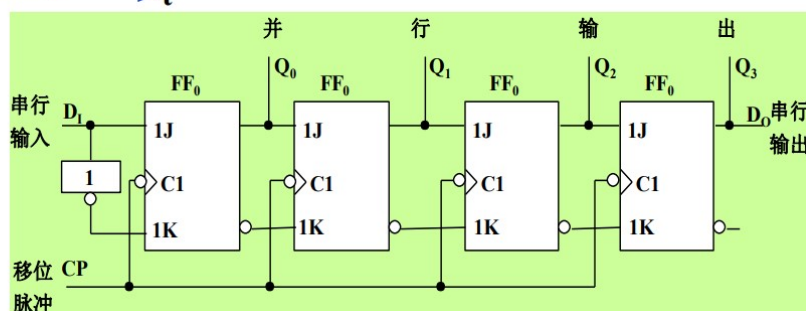
Q0~Q3: 输出(Q0 --> Q3)

## 双四位移位寄存器

- 工作示意：可用JK触发器或D触发器构成移位寄存器，实现串行转并行



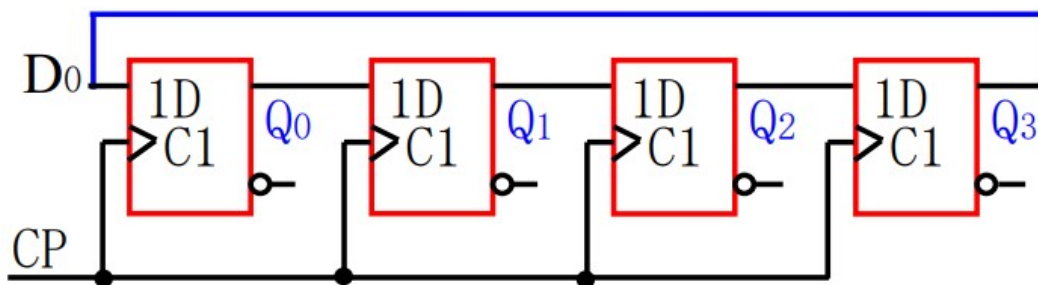
$$Q_3^{n+1} = D, Q_2^{n+1} = Q_3^n, \\ Q_1^{n+1} = Q_2^n, Q_0^{n+1} = Q_1^n$$



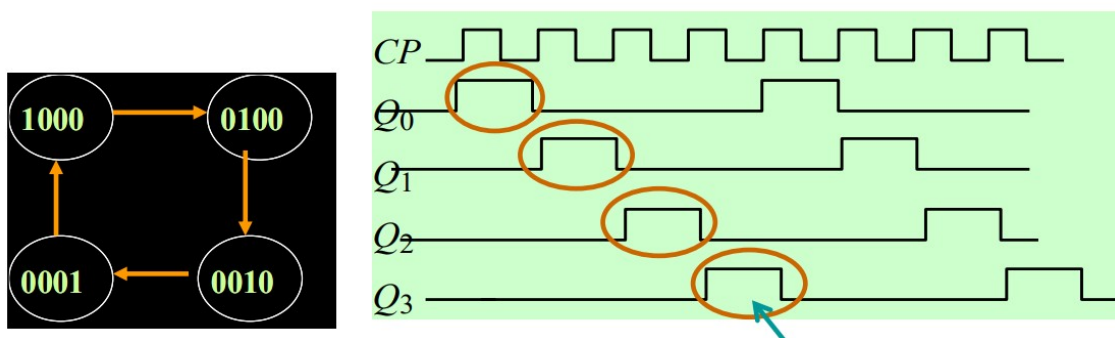
## 移存型计数器

### 环型计数器 (Ring Counter)

- 将移存器的最后一级输出反馈到第一级的输入



- 取状态只有一个 1 或 0 的为主循环，4 进制计数器。不需要译码



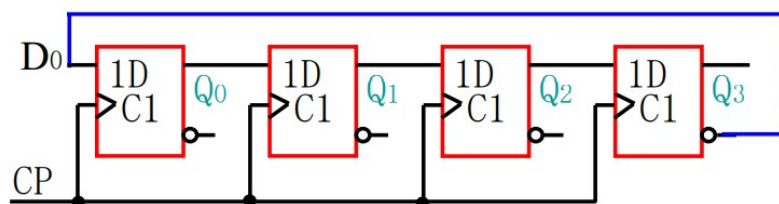
- 特点
  - n 位移存器，可构成 模  $M = N$  的环形计数器。
  - 缺点：无自启动能力，电路利用率不高。在计数开始之前，预置  $Q_0Q_1Q_2Q_3 = 1000$ ，进入主循环。
  - 优点：不需要译码器，输出顺序脉冲。

### 扭环形计数器 (Johnson Counter)

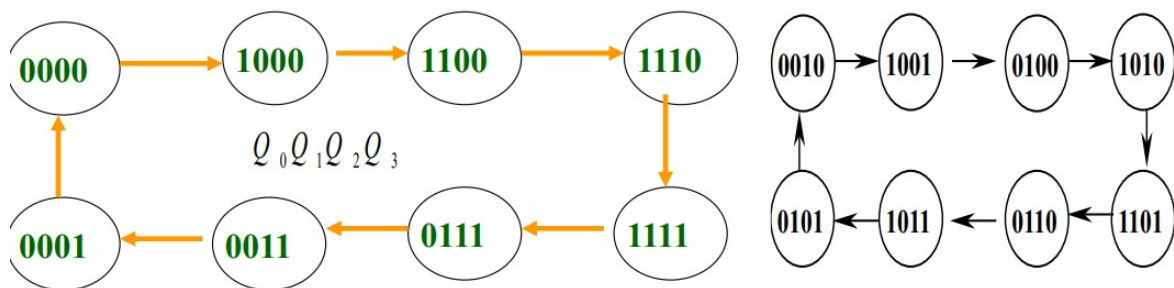
- 将移存器的最后一级输出反相后接到第一级的输入。

#### ※ 电路结构：

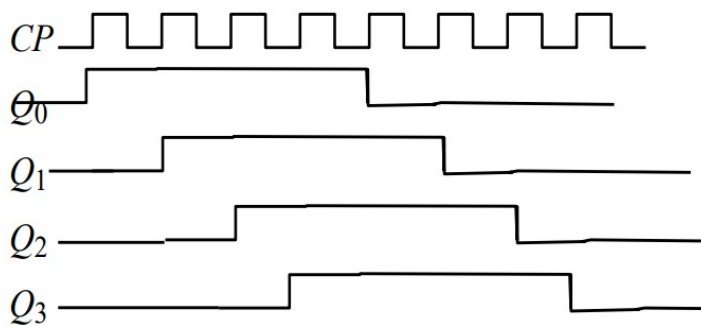
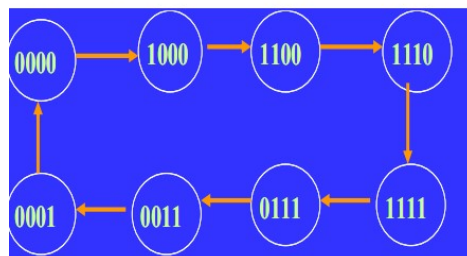
$$D_0 = \overline{Q_3^n}$$



- 取格雷码计数状态为主循环，模  $M = 8$  的计数。



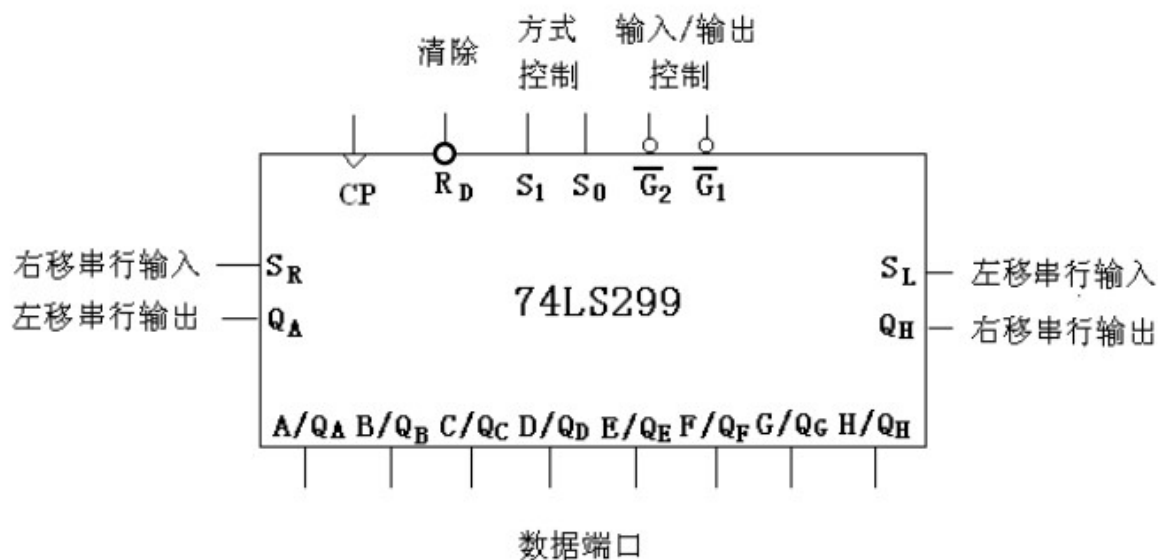
### ※主循环波形图:



- 电路特点
  - 优点:
    - 扭环计计数器的模  $M = 2N$ ，提高了电路利用率；
    - 电路状态译码时不会产生竞争 - 冒险现象。
  - 缺点:
    - 无自启动能力。在异步清零0000，进入主循环

### 中规模 8位双向移位寄存器 - - 74LS299

- 芯片示意



### 74LS299的逻辑符号

- 说明:
  - $R_D$  异步清零
  - $G_2$   $G_1$  控制数据端口是输入还是输出:
    - $G_2 = G_1 = 0$  输出引脚
  - CP: 上升沿触发
  - $S_1$   $S_0$  控制四种工作方式: 保持(00), 并行置数(11), 右移(01), 左移(10)



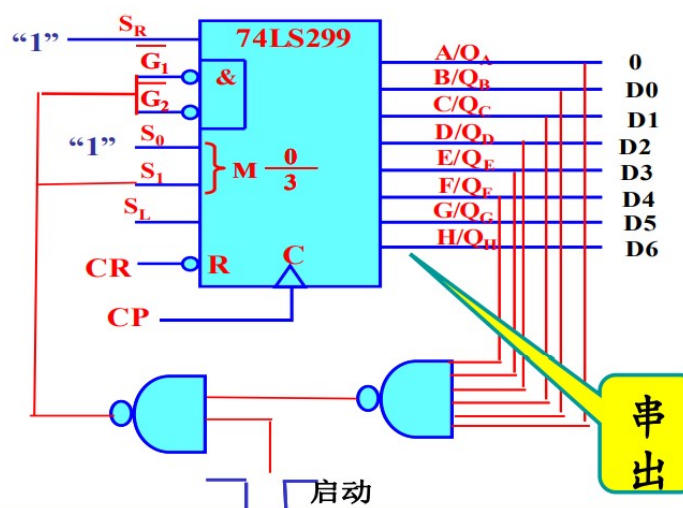
- 工作时：右移为例，数据端口统一右移，最右侧H通过右移串行输出 $Q_H$ 移出；同时，A通过左边右移串行输入 $S_R$ 移入
- 并行置数 时钟上升沿把数据端口八个数存储

输入信号						响应操作
Rd	G <sub>2</sub>	G <sub>1</sub>	S <sub>1</sub>	S <sub>0</sub>	CP	
0	0	0	×	×	×	清零， $Q_A \sim Q_H \rightarrow I/O_A \sim I/O_H$ 输出低电平
1	×	×	1	1	↑	并行置数， $I/O_n \rightarrow Q_n$
1	0	0	0	1	↑	右移， $S_R \rightarrow Q_A$ , $Q_A \rightarrow Q_B$ , $Q_H$ 右移出, $Q_A \sim Q_H \rightarrow I/O_A \sim I/O_H$
1	0	0	1	0	↑	左移， $S_L \rightarrow Q_H$ , $Q_H \rightarrow Q_G$ , $Q_A$ 左移出, $Q_A \sim Q_H \rightarrow I/O_A \sim I/O_H$
1	0	0	0	0	×	保持， $Q_A \sim Q_H$ 输出保持不变, $Q_A \sim Q_H \rightarrow I/O_A \sim I/O_H$

- 应用：7位并行转串行（图中  $D_0 \sim D_6$  七位按顺序  $D_6 D_5 D_4 D_3 D_2 D_1 D_0$ ）

### Parallel in/Serial out

实现7位并行 - 串行转换电路

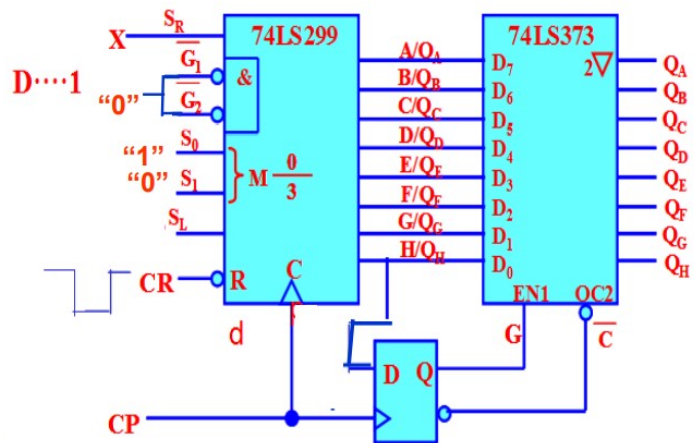


	CP	QA	QB	QC	QD	QE	QF	QG	QH
置数	1	0	D0	D1	D2	D3	D4	D5	D6
	2	1	0	D0	D1	D2	D3	D4	D5
	3	1	1	0	D0	D1	D2	D3	D4
	4	1	1	1	0	D0	D1	D2	D3
右移	5	1	1	1	1	0	D0	D1	D2
	6	1	1	1	1	1	0	D0	D1
	7	1	1	1	1	1	1	0	D0
	8	0	D0	D1	D2	D3	D4	D5	D6

- 应用：8位串行转并行

## 实现8位串行— 并行数据转换

### Serial-to-parallel Converter



CP	QA	QB	QC	QD	QE	QF	QG	QH
0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0
2	D0	1	0	0	0	0	0	0
3	D1	D0	1	0	0	0	0	0
4	D2	D1	D0	1	0	0	0	0
5	D3	D2	D1	D0	1	0	0	0
6	D4	D3	D2	D1	D0	1	0	0
7	D5	D4	D3	D2	D1	D0	1	0
8	D6	D5	D4	D3	D2	D1	D0	1
9	D7	D6	D5	D4	D3	D2	D1	D0

清零

右移

锁存

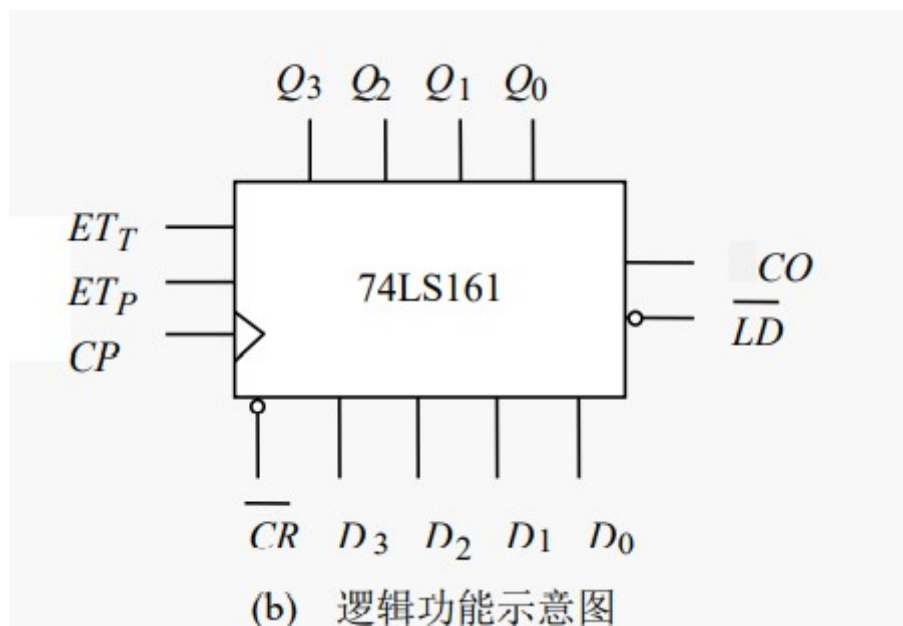
移位同样可以看作幂运算，左移 $n$ 位乘 $2^n$  右移除 $2^n$

## 中规模计数器 (Counter)

### 集成同步二进制计数器

#### 四位二进制加计数器 (74LS161/163)

- 不同
  - 74LS161采用异步清零方式。
  - 74LS163采用同步清零方式。(需要时钟触发)

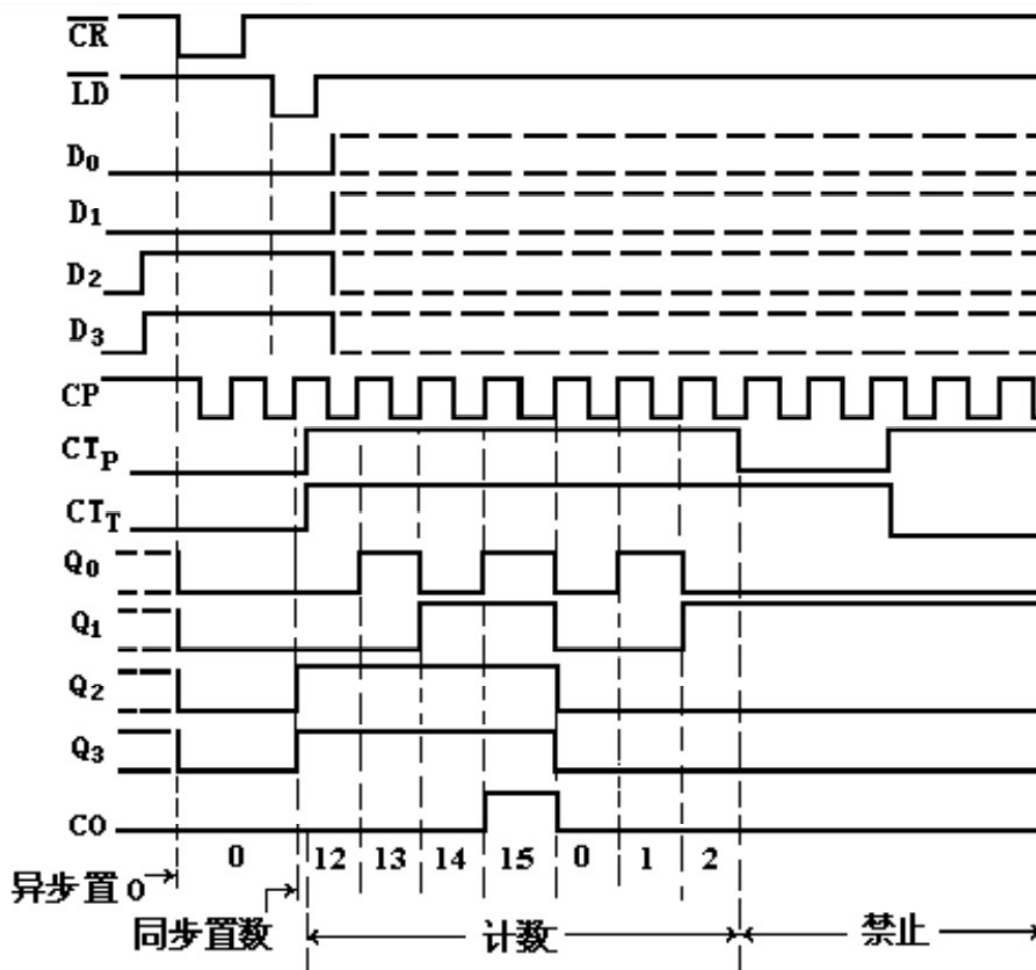


- 说明：
  - CR: 异步清零



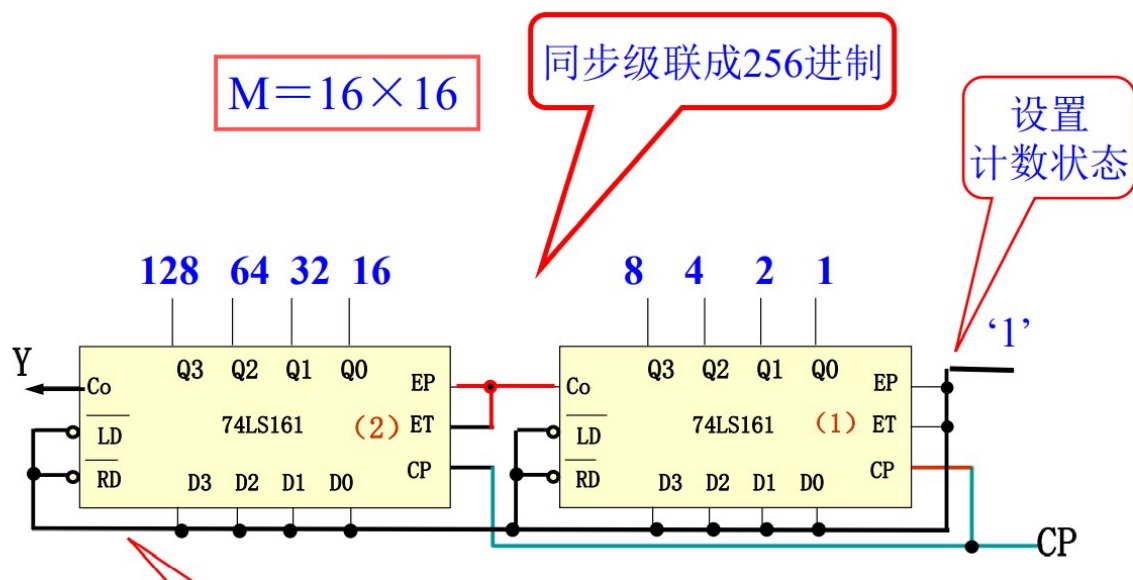
- LD(Load): 低有效, 并行置数端(同步置数), 把输入直接置到输出
- EP、ET: 使能, 高有效, 同时有效开始计数, 从0000到1111
- CO: 进位输出, 计数到1111后, 进位产生高电平1 (到1111就变成1)
- 使能端有一个为0, 进入保持状态, 不计数
- 记录时钟上升沿个数
- 对于特殊情况, 处于保持状态且输出为1111, 对于进位输出, 如果  $ET=1$ , 那么进位输出为1, 否则为0, 此时相当于  $CO = ET \cdot Q_3 Q_2 Q_1 Q_0$
- 计数器型号: 奇数结尾都为16进制, 偶数10进制

输 入				输 出				CT74LS161
$\overline{CR}$	$\overline{LD}$	EP	ET	CP	$D_3 D_2 D_1 D_0$	$Q_3 Q_2 Q_1 Q_0$	CO	
0	X	X	X	X	X X X X	0 0 0 0	0	异步置0
1	0	X	X	↑	$d_3 d_2 d_1 d_0$	$d_3 d_2 d_1 d_0$		$CO = E_T \cdot Q_3 Q_2 Q_1 Q_0$
1	1	1	1	↑	X X X X	计 数		$CO = Q_3 Q_2 Q_1 Q_0$
1	1	0	X	X	X X X X	保 持		$CO = E_T \cdot Q_3 Q_2 Q_1 Q_0$
1	1	X	0	X	X X X X	保 持	0	



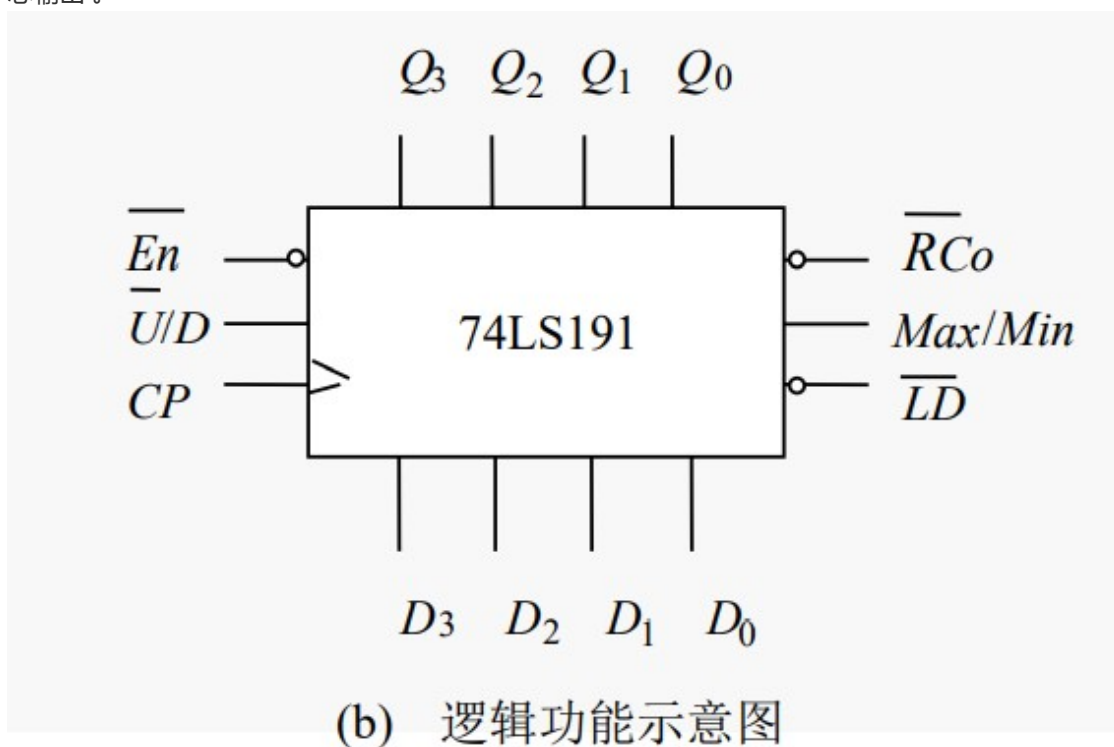
#### 容量扩展

- 用四位二进制计数器构成  $4 \cdot n$  位 二进制计数器
- **同步级联** (只适用于有使能端计数器)
- CP 同时接在各片计数器的时钟输入端
- 级联步骤:
  1. 排列计数器高低位的顺序;
  2. 找到低位向高位的进位信号;



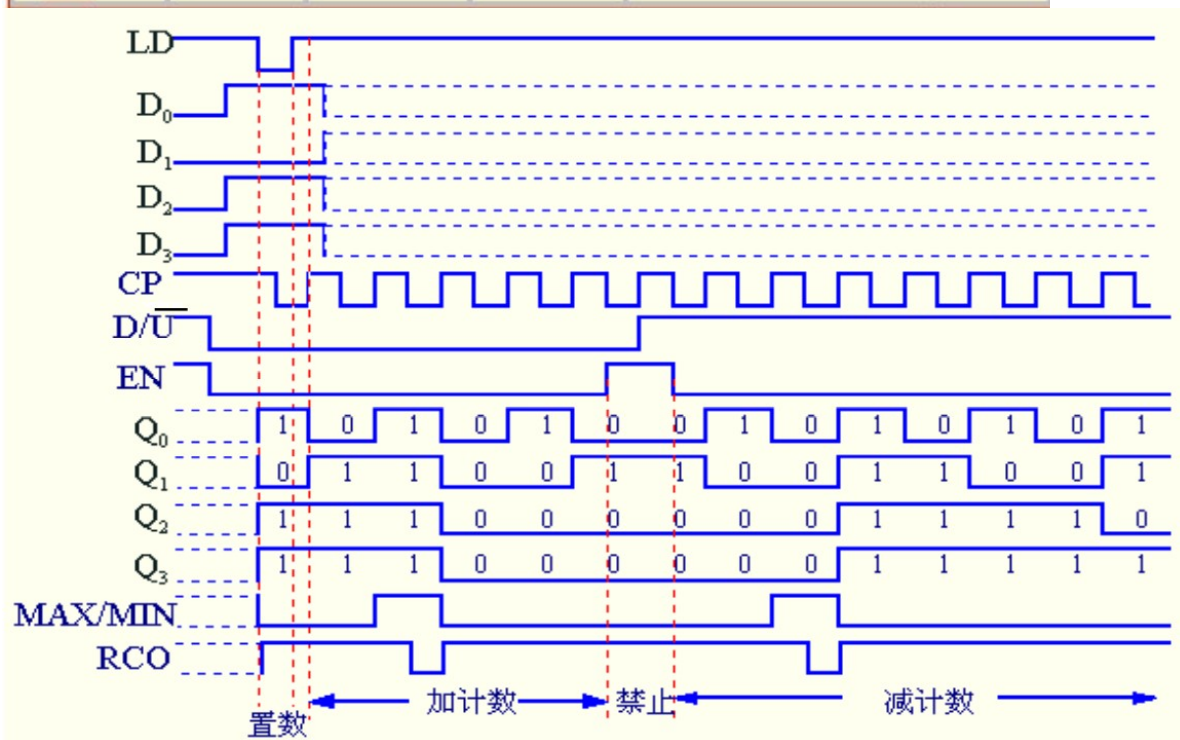
### 同步二进制可逆计数器 (74191)

- 逻辑功能：计数器可逆，异步置数，计数器状态可控，进位（借位）负脉冲输出，最大（最小）标志输出。



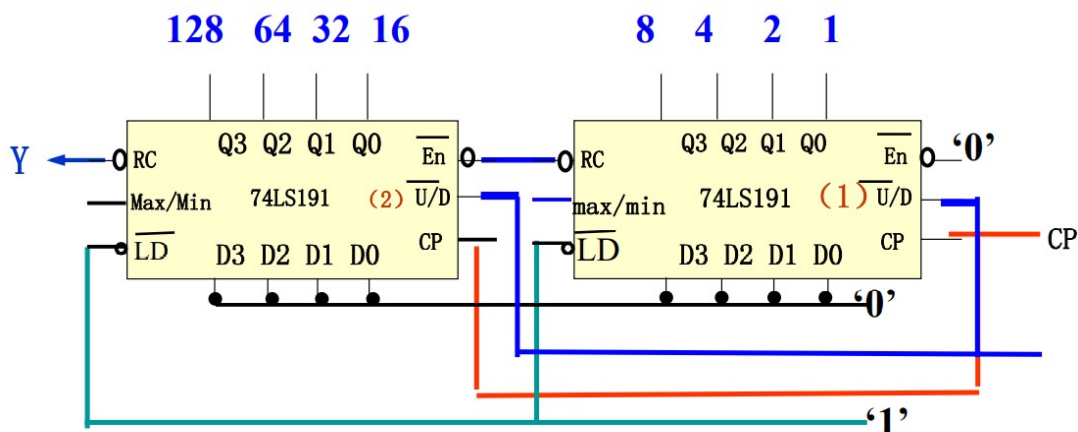
- 说明：
  - $En$ 使能，低有效
  - $Max/Min$  是最大或最小状态标志
  - $RCo$ ：进位或借位， $En$ 、 $RCo$ ：用于同步级联。：
    - 加法计数时，计数到1111，最大最小标志为1时，该位为0
  - $\overline{U/D}$ ：控制加计数或减计数，低电平为U(Up)有效 加法计数，反之D(Down) 有效，减法计数
  - $LD$ ：并行置数，异步
  - 特殊情况，如计数到1111后使能无效，不再计数或进位
    - $En = 0$ ,  $max/min = 1$ ,  $RCo = 0$
    - $En = 1$ ,  $max/min = 1$ ,  $RCo = 1$  停止计数，无效电平

CP	$\overline{\text{En}}$	$\overline{\text{LD}}$	$\overline{\text{U/D}}$	工作状态
X	1	1	X	保持
X	X	0	X	预置数
	0	1	0	加法计数
	0	1	1	减法计数



注意：RCO有效的只有半个时钟周期（实际是通过Max/Min和时钟与得到）

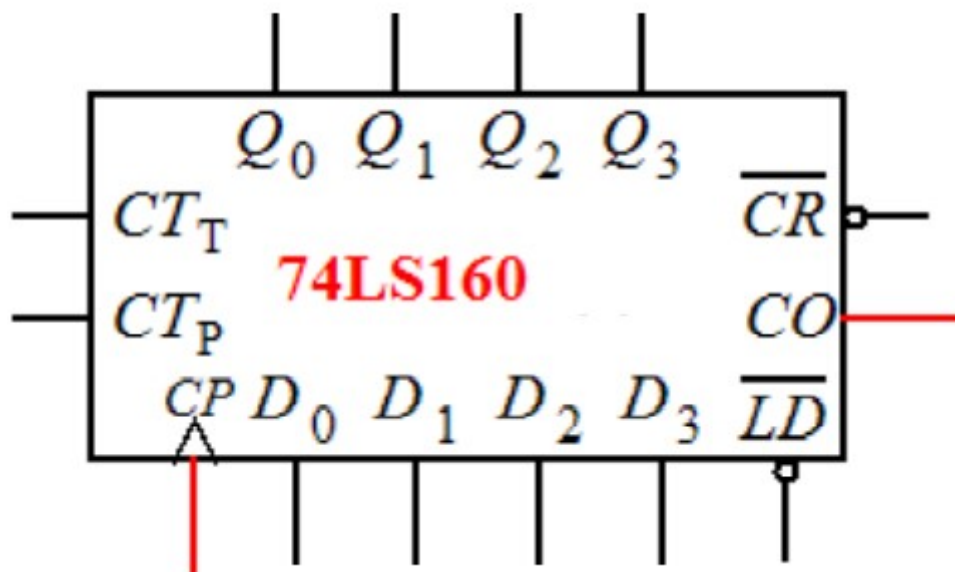
- 级联：构成  $4 \times n$  位计数器



不要使用Max/Min做级联

## 集成同步十进制计数器

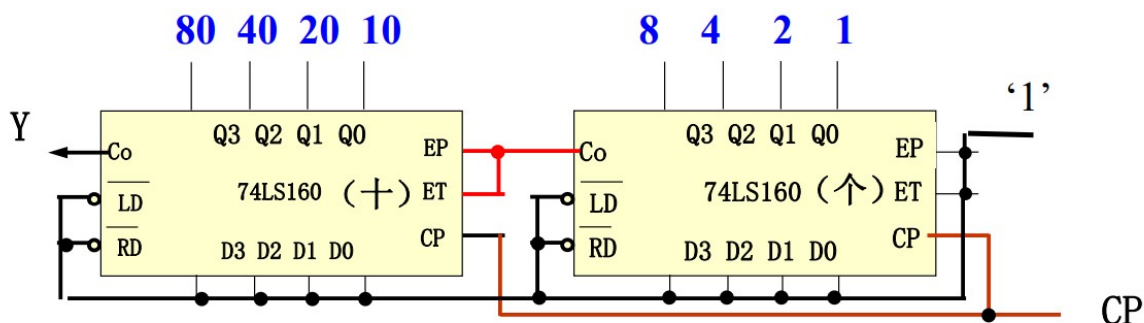
- 有三种：
  - 74160：异步清零方式。
  - 74162：同步清零方式。
  - 74190：单时钟十进制同步可逆计数器，其引脚排列图和逻辑功能示意图与74191相同。



- 端口功能可以类比161

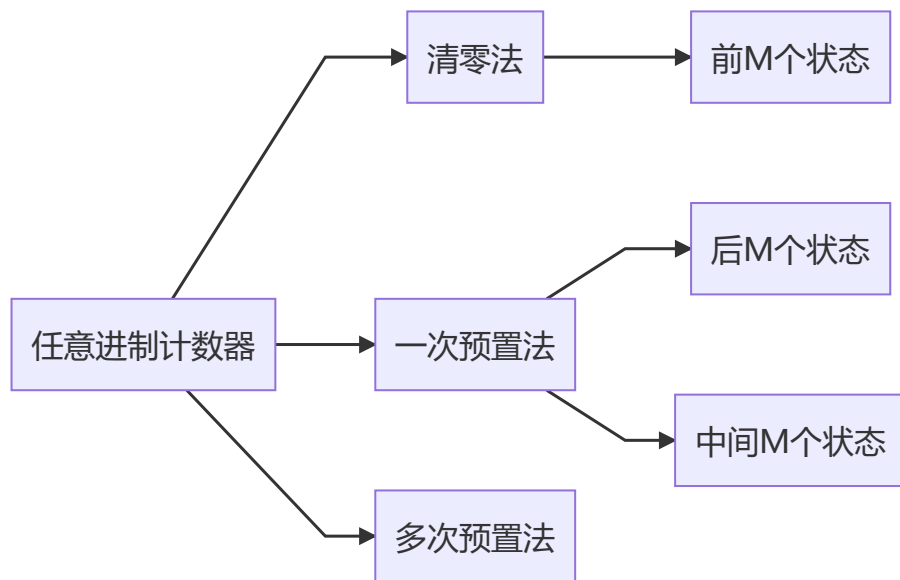
输 入				输 出				CT74LS160				
$\overline{CRLD}$	$P$	$T$	$\overline{CP}$	$D_3D_2D_1D_0$	$Q_3Q_2Q_1Q_0$	$CO$						
0	×	×	×	×	×	×	×	0	0	0	0	异步置 0
1	0	×	×	↑	$d_3d_2d_1d_0$	$d_3d_2d_1d_0$						$CO = T \overline{Q_3}\overline{Q_2}\overline{Q_1}\overline{Q_0}$
1	1	1	1	↑	×	×	×	×	计 数			$CO = Q_3\overline{Q_2}\overline{Q_1}\overline{Q_0}$
1	1	0	×	×	×	×	×	×	保 持			$CO = T \overline{Q_3}\overline{Q_2}\overline{Q_1}\overline{Q_0}$
1	1	×	0	×	×	×	×	×	保 持		0	

- 级联可组成8421码100进制



## 用中规模计数器构成任意进制计数器

方法：



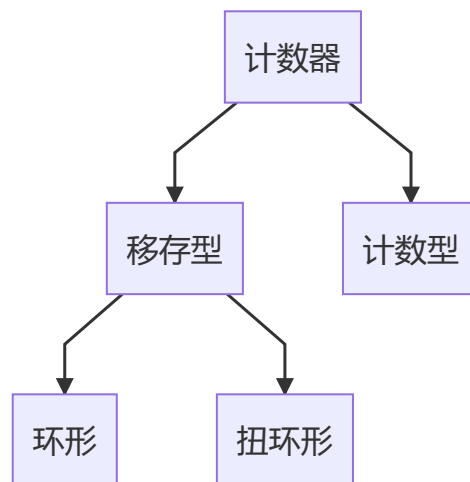
### 清零法

- 自然态序M进制计数器 ( $0 \sim M-1$ )，利用复位端和置数端(load 0000)实现归零。
  - 用同步复位端（置数端）归零（在M-1状态复位）
  - 用异步复位（置数端）归零（在M状态复位）：亚稳态。

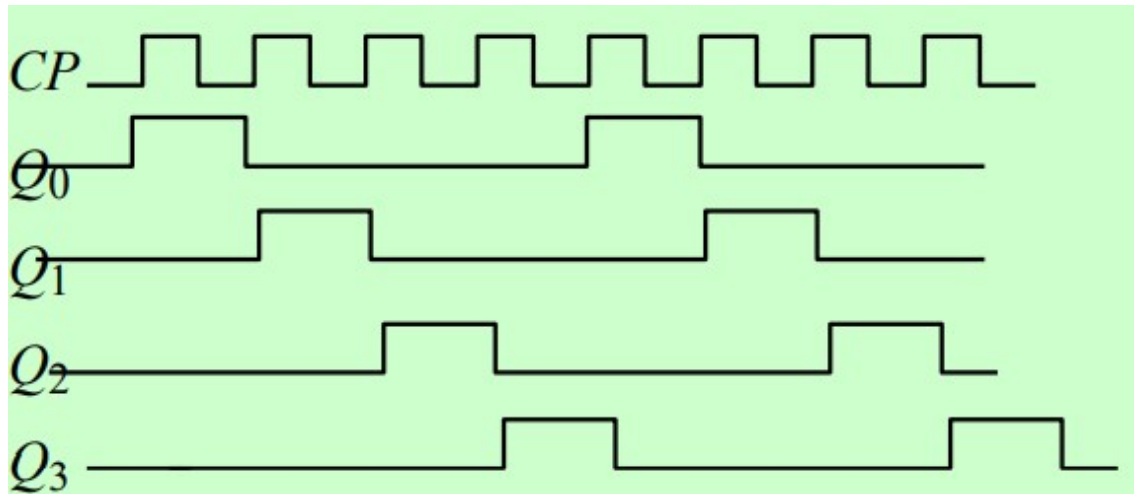
## 计数器应用

### 顺序脉冲（节拍）发生器

- 组成：计数器 + 译码电路



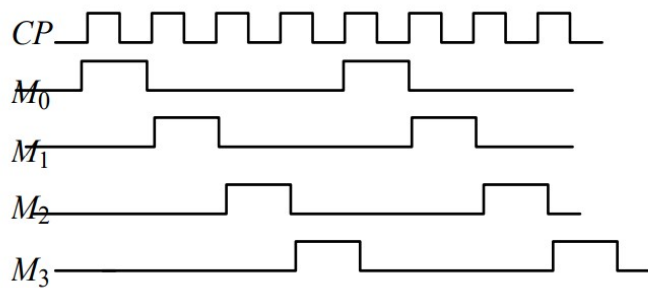
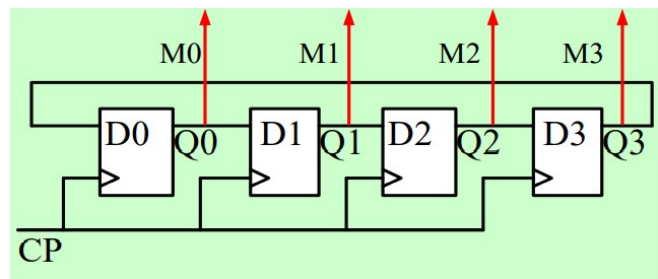
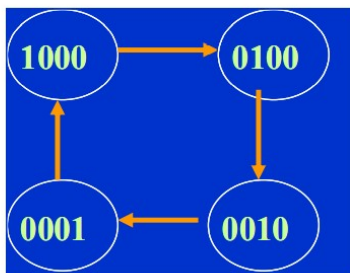
- 计数器的M个状态，产生M路脉冲



### 1. 环形计数器构成的节拍发生器

- 特点：
  - 结构简单，不必附加译码电路。
  - 但使用的触发器数目多，M路顺序脉冲用了M个触发器。

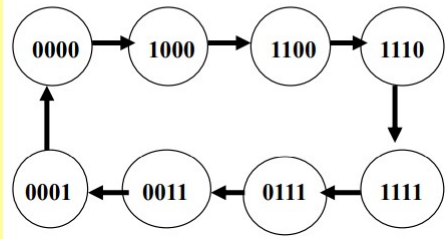
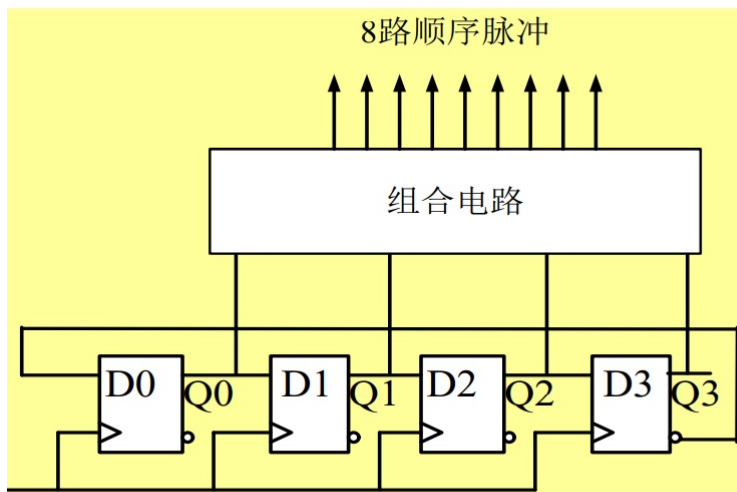
直接输出，不需译码



### 2. 扭环形计数器构成的节拍发生器

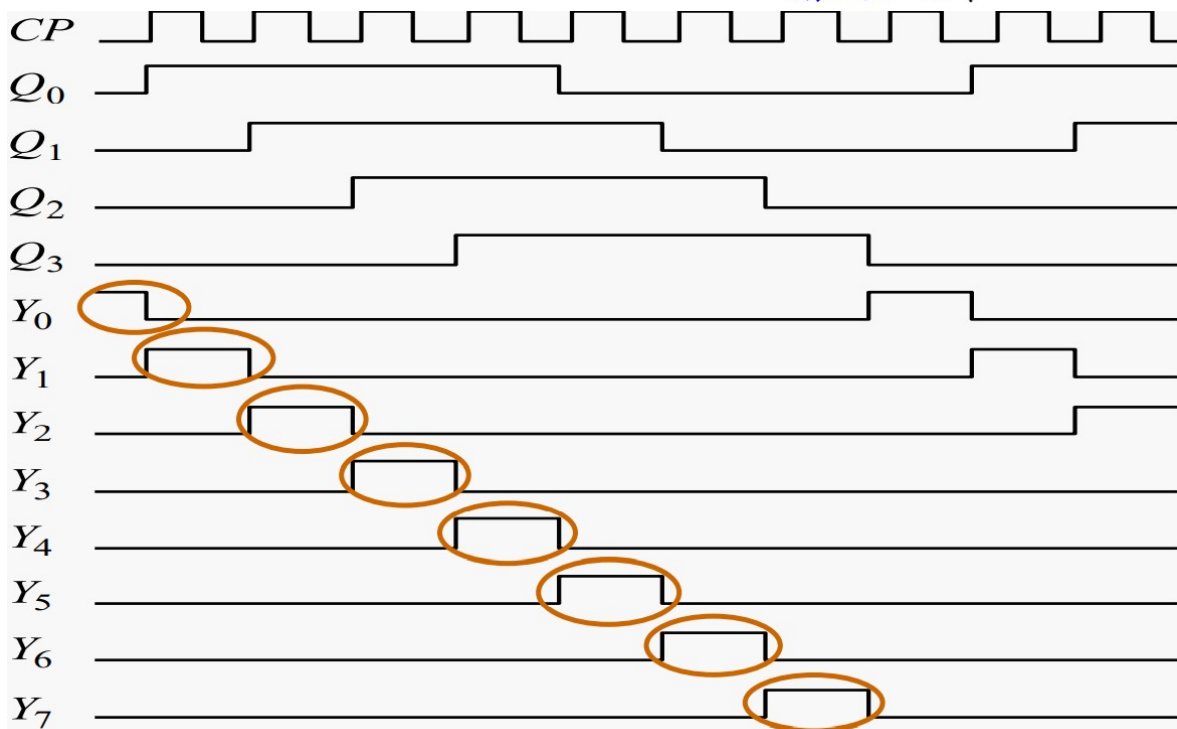


匹配的译码器应另行设计。



输入：现态

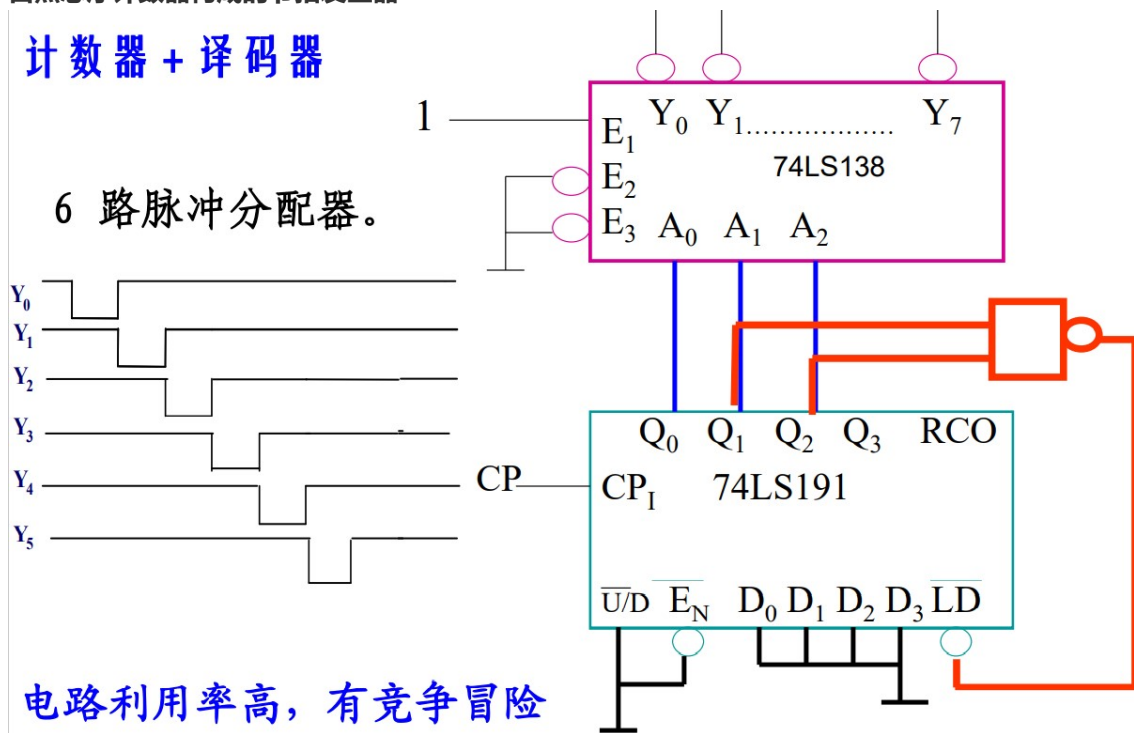
输出：M个



- 可以看出是格雷码计数，不会有竞争冒险，消除毛刺
- 特点：
  - 计数器采用扭环型计数器可从根本上消除竞冒险现象，
  - 译码电路简单。
  - 但电路利用率较低

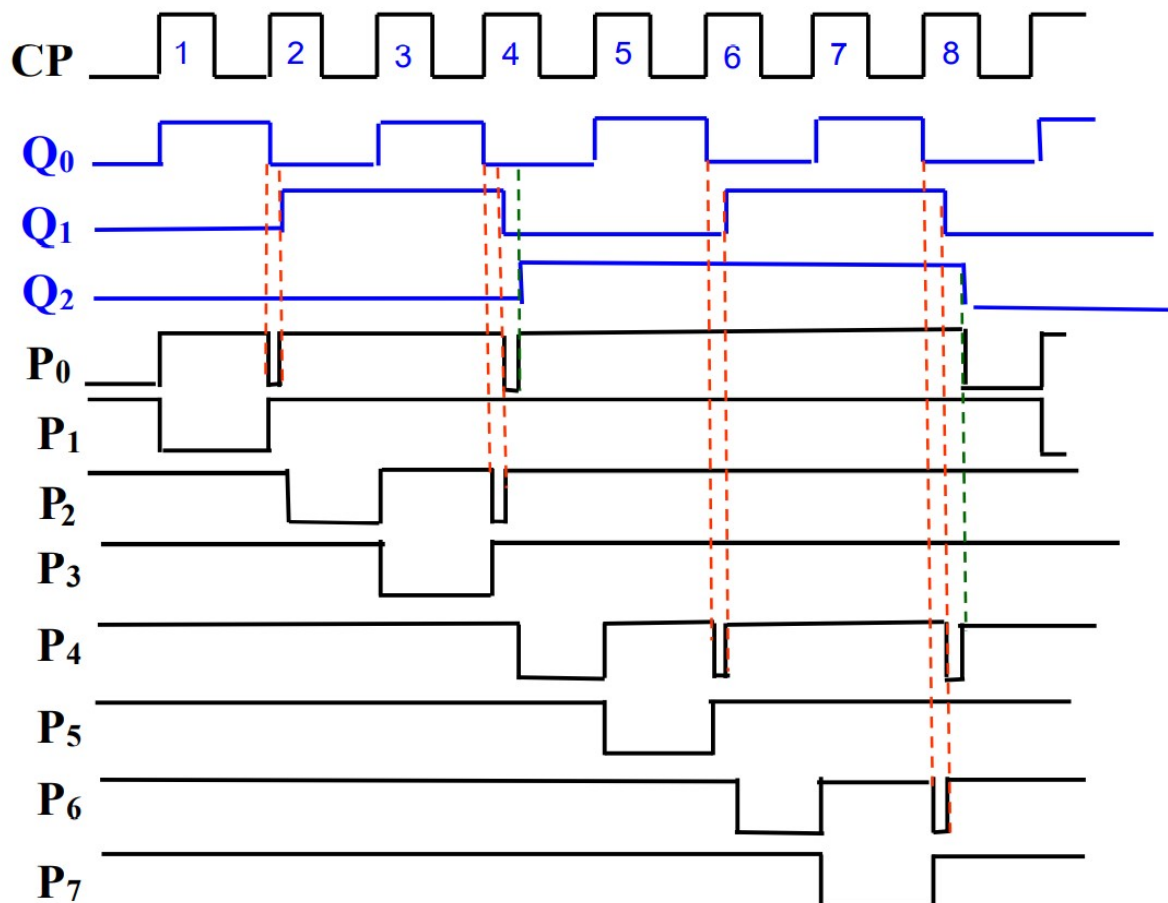
### 3. 自然态序计数器构成的节拍发生器

#### 计数器 + 译码器



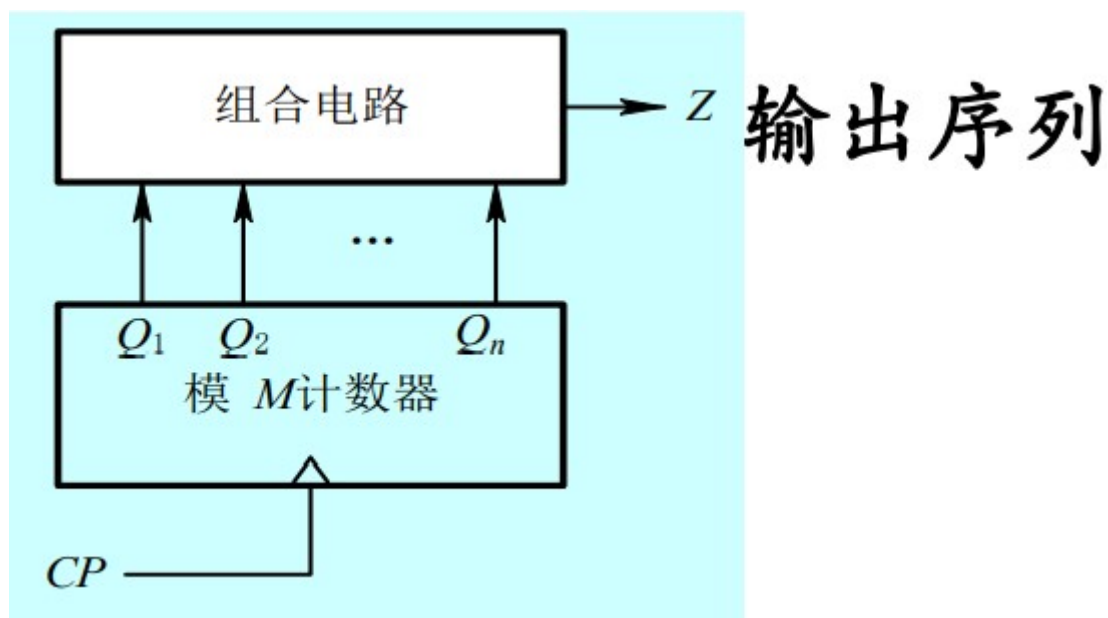
注：138 38译码器；191异步置数计数器

- 不是格雷码，会有毛刺

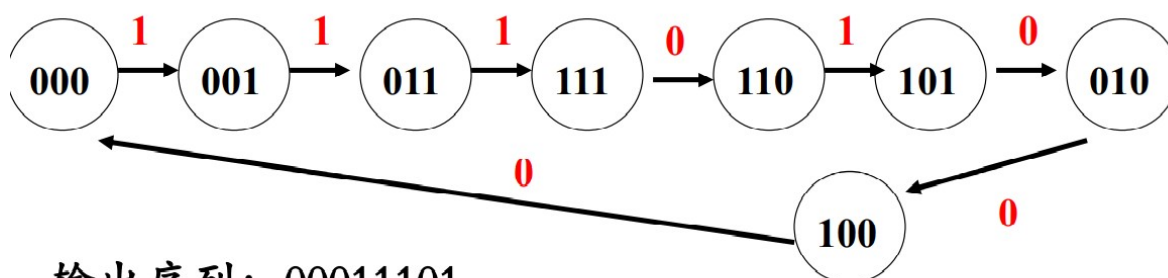
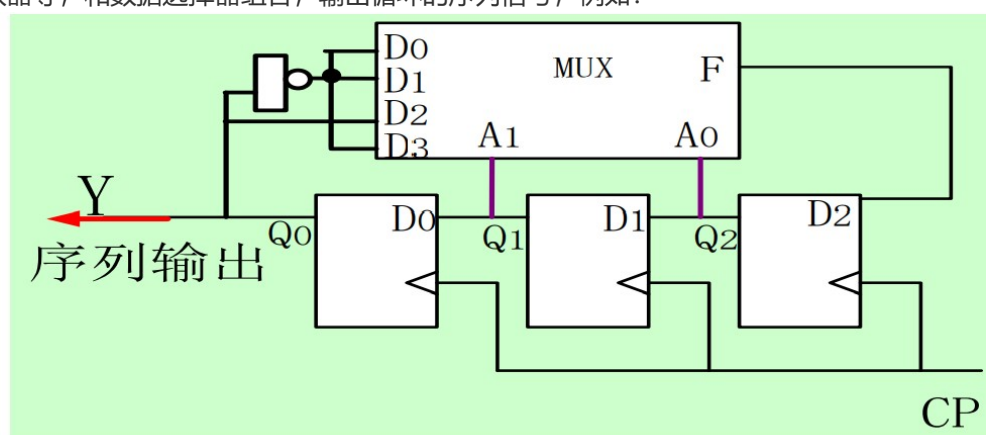


- 解决：
  - 控制使能端
  - 不用自然态序计数，用格雷码





利用计数器等，和数据选择器组合，输出循环的序列信号，例如：



输出序列：00011101

## 同步时序电路的设计

- 关键：
  - 确定应有的状态转换规律；
  - 根据状态转换规律，求得触发器的激励信号表达式。
- 步骤：
  1. 设计要求
  2. 原始状态图
  3. 化简为最简状态图
  4. 状态编码
  5. 选触发器，求输出、驱动方程
  6. 画电路图

## 7. 检查电路能否自启动

### 详细步骤说明

#### 1. 逻辑抽象，得状态图（表）：

- 输入输出
- 系统状态数
- 每一状态在规定条件下的转移方向

#### 2. 状态化简：

- 等价状态 —— 如果有两个或两个以上状态，在相同的外部输入下，各自的外部输出相同，次态相同，则这几个状态为等价状态。应把等价状态归并为一个。

#### 3. 状态编码，得真值表：

- 将化简后的各状态用二进制代码来表示。

1. 确定触发器个数
2. 状态分配
3. 写出编码形式的状态转换真值表

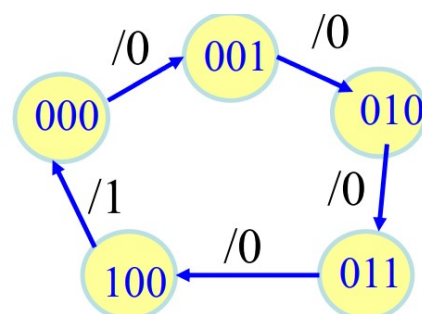
#### 4. 由状态图（表）得驱动方程和输出方程

1. 填卡诺图
2. 化简卡诺图得激励方程、输出方程

#### • 例：用D触发器设计一个五进制的加法计数器。

1. 原始状态图：无输入信号，有进位输出C，有5个状态分别表示0 — 4。
2. 状态化简：无需化简。
3. 状态编码：触发器个数  $n = 3$ ；状态分配：按自然态序编码；写出编码形式的状态图、真值表

现 态	次 态	输 出
$Q_2^n \ Q_1^n \ Q_0^n$	$Q_2^{n+1} \ Q_1^{n+1} \ Q_0^{n+1}$	c
0 0 0	0 0 1	0
0 0 1	0 1 0	0
0 1 0	0 1 1	0
0 1 1	1 0 0	0
1 0 0	0 0 0	1
1 0 1	x x x	x
1 1 0	x x x	x
1 1 1	x x x	x



#### 4. 写出输出方程和驱动方程：

- 输出方程：  $C = Q_2^n$

- 驱动方程：对于 D 触发器，要求状态方程最简

$Q_2^n$ \ $Q_1^n Q_0^n$				
	00	01	11	10
0	0	0	1	0
1	0	X	X	X

Q2 的次态卡诺图

$Q_2^n$ \ $Q_1^n Q_0^n$				
	00	01	11	10
0	1	0	0	1
1	0	X	X	X

Q0 的次态卡诺图

$Q_2^n$ \ $Q_1^n Q_0^n$				
	00	01	11	10
0	0	1	0	1
1	0	X	X	X

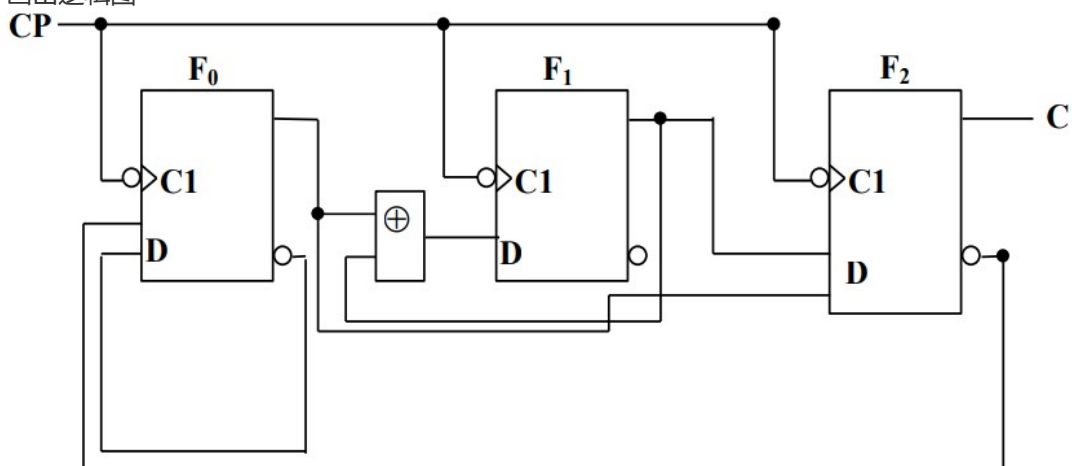
Q1 的次态卡诺图

$$Q_2^{n+1} = D_2 = Q_1^n Q_0^n$$

$$Q_1^{n+1} = D_1 = Q_0^n \overline{Q_1^n} + \overline{Q_0^n} Q_1^n$$

$$Q_0^{n+1} = D_0 = \overline{Q_2^n} \overline{Q_0^n}$$

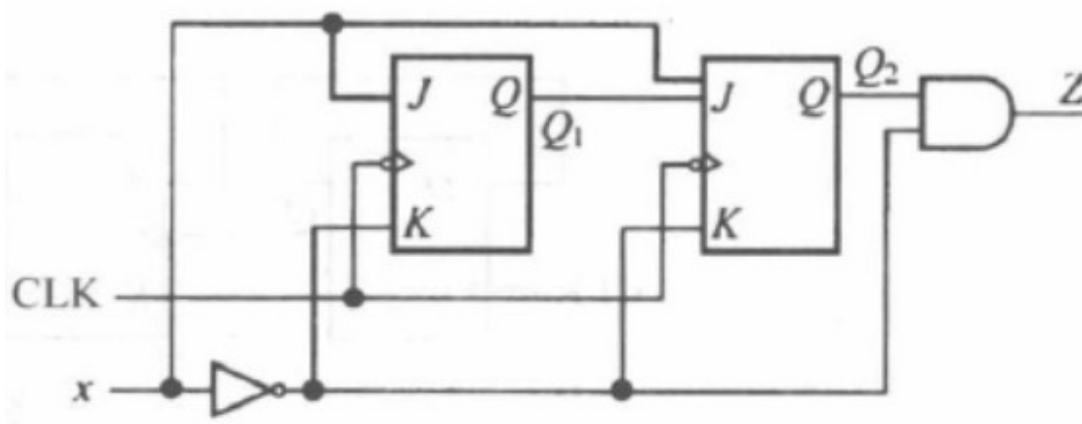
5. 画出逻辑图



6. 检查自启动：将无效状态101、110、111代入状态方程计算：电路能够自启动

## 附

- 一个端口接两根线表示两个信号与运算



- 端口悬空认为输入逻辑“1”

