

数字电路四位二进制全加器实验报告_数字电路笔记

[weixin_39721924](#) 于 2020-10-31 04:10:11 发布 3391 收藏 27
文章标签: [数字电路四位二进制全加器实验报告](#) [相邻位数字差值的绝对值不能超过](#)

数字电路

写给读者

作者是华中科技大学本科生,这是我的数字电路笔记

感谢你阅读本文。本文不允许任何形式的转载。有任何问题请联系: wnn2000@hust.edu.cn

第一章 数字电路概论

1.二进制

LSB和MSB

通常,MSB位于二进制数的最左侧,LSB位于二进制数的最右侧。

传输

串行:在计算机总线或其他数据通道上,每次传输一个bit

并行:所有bit同时传送

BCD编码

意义:用4个bit表示0-9

有权码

8421码:4个bit的权值分别是8、4、2、1

2421码:

5421码:

无权码

余3码:在8421码的基础上,将每个代码都加上0011而形成

余3循环码:

格雷码:

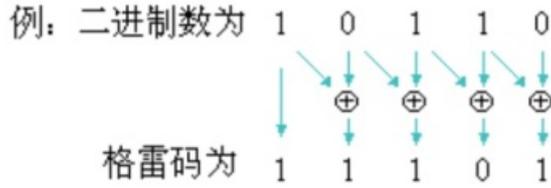
某二进制数为 $B_{n-1}B_{n-2} \cdots B_2B_1B_0$

其对应的格雷码为 $G_{n-1}G_{n-2} \cdots G_2G_1G_0$

异或运算：
相同为0
相异为1

其中：最高位保留—— $G_{n-1} = B_{n-1}$

其他各位—— $G_i = B_{i+1} \oplus B_i \quad i=0, 1, 2, \dots, n-2$



知乎 @楠木

2.二值逻辑变量与基本逻辑运算

输入输出

仅有0、1

几种逻辑运算类型

序号	名称	GB/T 4728.12-1996		国外流行图形符号	曾用图形符号
		限定符号	国标图形符号		
1	与门	&			
2	或门	≥ 1			
3	非门	 逻辑非入和出			

由真值表写逻辑表达式

把每个输出为1的一组输入变量组合状态以逻辑乘形式表示（原变量表示取值1，反变量表示取值0），再将所有的这些逻辑乘进行逻辑加。

第二章 逻辑代数与硬件描述语言基础

1.逻辑代数

基本公式和恒等式

这部分太简单了

对偶规则

对于逻辑函数式，若将其中的与换成或，或换成与；并将1换成0，0换成1；所得的新的函数式就是L的对偶式

例
$$L = (A + \bar{B})(A + C) \quad L' = A\bar{B} + AC$$

2.逻辑函数的卡诺图化简法

最小项

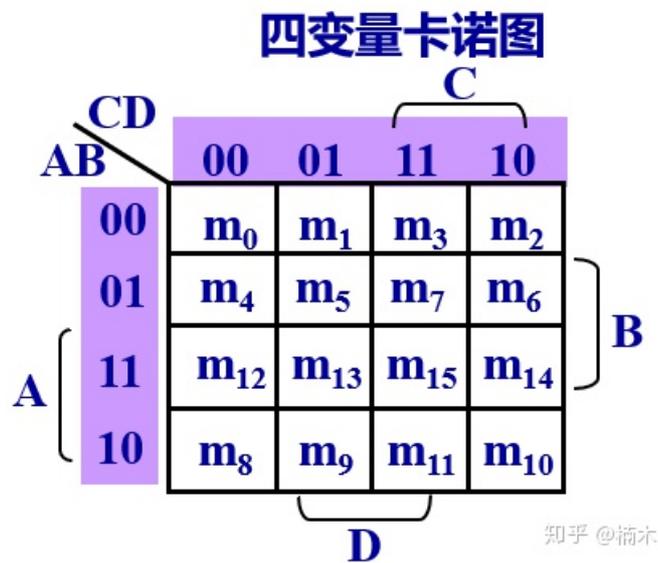
n个变量(X_1, X_2, \dots, X_n)的最小项就是n个因子的乘积，在该乘积中每个变量都以它的原变量或非变量的形式出现一次，且仅出现一次。

如三变量逻辑函数 $f(A, B, C)$

$$\overline{A}\overline{B}\overline{C} \quad \overline{A}\overline{B}C \quad ABC \quad \overline{A}BC \quad \text{-----最小项}$$

$$\overline{A}\overline{B}\overline{C}A \quad \overline{A}\overline{B} \quad A(B+C) \quad \text{-----不是最小项}$$

卡诺图

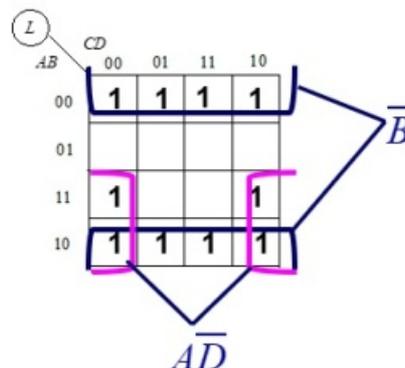


卡诺图横纵二进制码为格雷码。使用格雷码的意义在于：使得相邻的格子只有一个bit不同，方便寻找共同bit，从而画圈圈。

化简过程就是画圈圈的过程，把相邻的元素圈起来。（注意包括上下底相邻，左右边相邻和四角相邻）。包围圈内的方格数一定是 2^n 个，包围圈的方格数要尽可能多。

例 2 将逻辑函数 $L = \overline{A} \overline{B}\overline{C} + A\overline{C} \overline{D} + A\overline{B} + ABC\overline{D} + \overline{A} \overline{B}C$ 化简为最简与或表达式。

$$L = \overline{B} + A\overline{D}$$



无关项

在有些逻辑问题中，有些变量的取值要加以约束。

如A=1表示电机正转；B =1表示电机反转；C =1表示电机停转。ABC不能等于000、011、101、111中的任何一组。

填函数的卡诺图时在无关项对应的格内填符号“×”。逻辑函数式中用“Φ”或、“d”表示无关项。

化简时可根据需要视为“1”也可视为“0”，使函数化到最简。(无视无关项即可)

例1、试用卡诺图化简逻辑函数

$$L = m(5, 6, 7, 8, 9) + d(10, 11, 12, 13, 14, 15)$$

a、画出逻辑函数的卡诺图

b、化简逻辑函数

	CD			
AB	00	01	11	10
00	0	0	0	0
01	0	1	1	1
11	×	×	×	×
10	1	1	×	×

	CD				
AB	00	01	11	10	
00	0	0	0	0	
01	0	1	1	1	BD
11	×	×	×	×	BC
10	1	1	×	×	A

$$L = A + BC + BD$$

3.Verilog HDL基础

这部分等以后再补吧（逃）

第三章 逻辑门电路

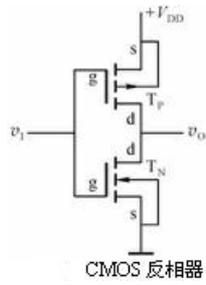
1.逻辑门电路分类



2.CMOS电路

MOS管的特性在模电中已经学过，不再赘述。

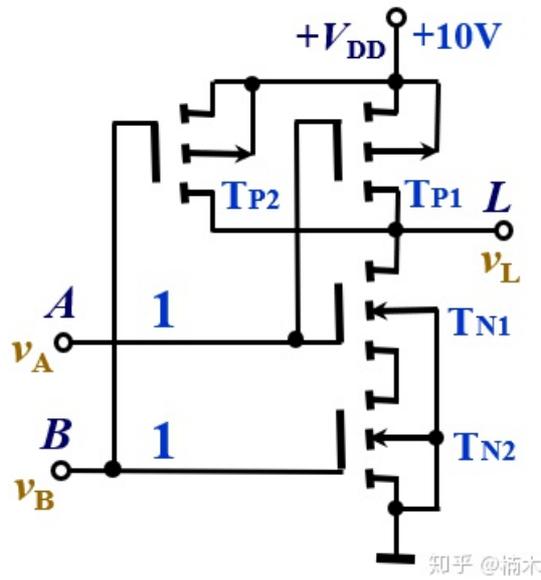
CMOS反相器(即非门)



当输入为低电位时，下方的NMOS的栅源电压为0，截止；上方的PMOS的栅源电压为 $-V_{DD}$ ，导通；输出为高电位。

当输入为高电位时，下方的NMOS的栅源电压为 V_{DD} ，导通；上方的PMOS的栅源电压为0，截止；输出为低电位。

CMOS与非门

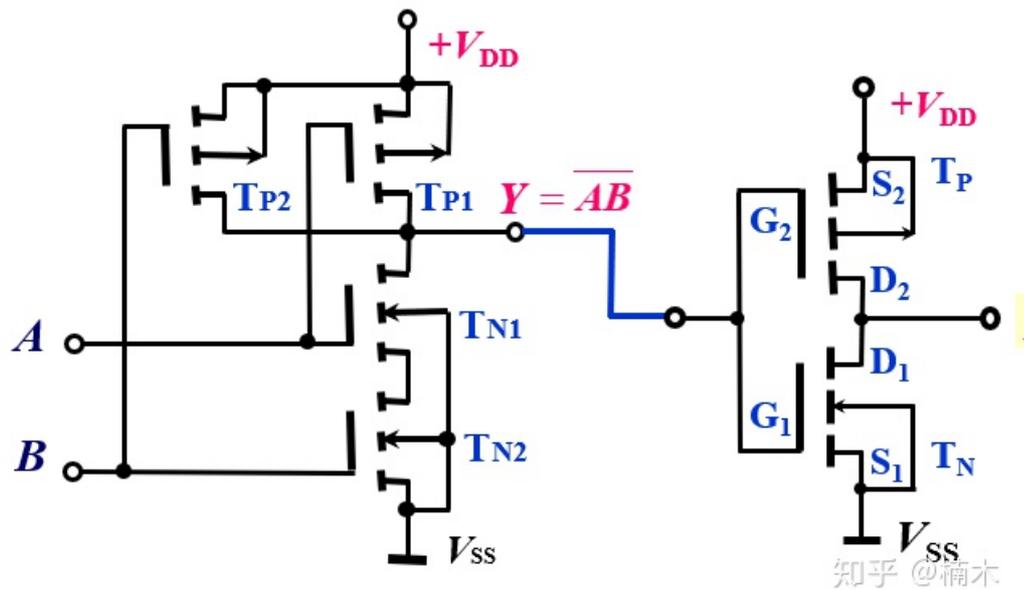


直接列出真值表

A	B	T_{N1}	T_{P1}	T_{N2}	T_{P2}	L
0	0	截止	导通	截止	导通	1
0	1	截止	导通	导通	截止	1
1	0	导通	截止	截止	导通	1
1	1	导通	截止	导通	截止	0

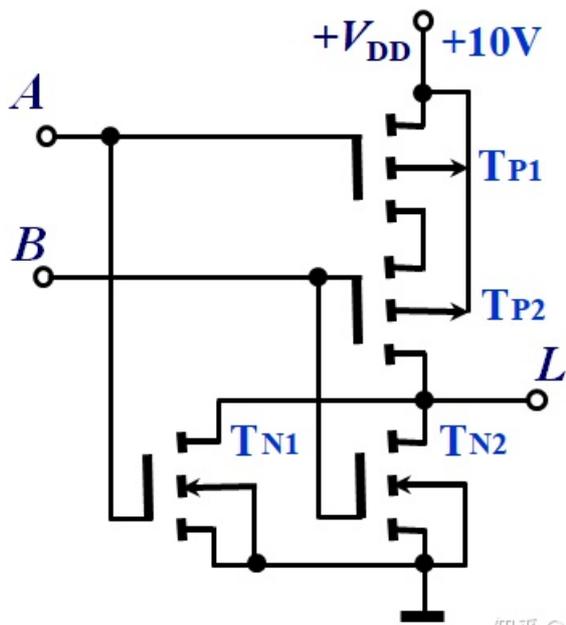
知乎 @楠木

CMOS与门



先与非再非即可得与

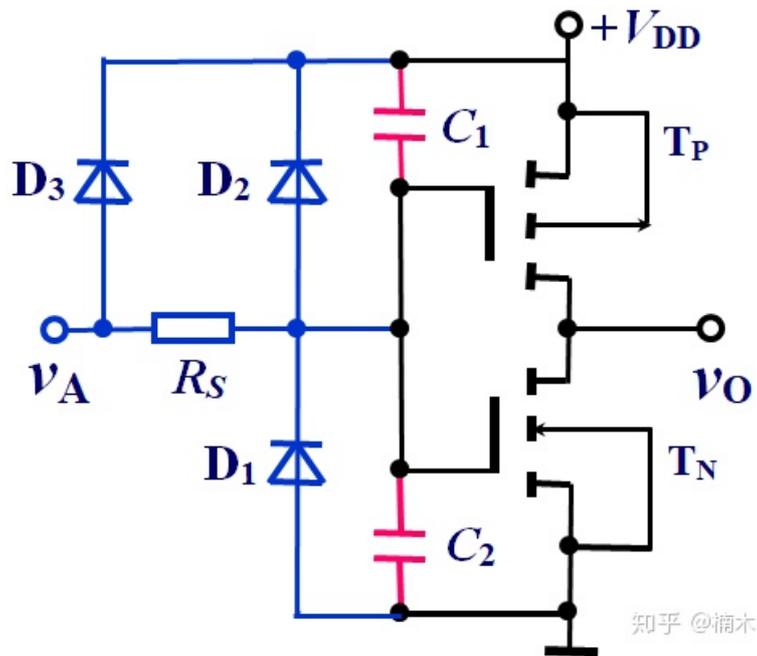
CMOS或非门



做类似与非门的分析即可

输入保护电路和缓冲电路

输入端保护电路



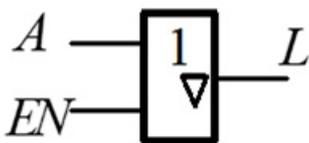
当输入电压不在正常电压范围时,二极管导通,限制了电容两端电压的增加,保护了输入电路。
 R_S 和MOS管的栅极电容组成积分电路,即使是大信号的输入,C充电也需要时间,保证漏极电位的正常

CMOS漏极开路 (OD) 门

这部分等以后再补吧 (逃)

三态输出门电路

加入一个“使能”输入,来决定输出是否为第三态——高阻态。



使能 EN	输入 A	输出 L
1	0	0
1	1	1
0	×	高阻

逻辑功能: 高电平同相逻辑门

知乎 @楠木

EN 上无圈,表示高电平有效;有圈,表示低电平有效
 还有其他的三态门,比如三态与非门,不加赘述

3.TTL逻辑门

非重点

第四章 组合逻辑电路

简单理解就是没有反馈的电路

1.组合逻辑电路的分析

- 1.由电路图写出各输出端的逻辑表达式
- 2.化简逻辑表达式,比如卡诺图法
- 3.根据化简后的表达式写出功能

2.组合逻辑电路的设计

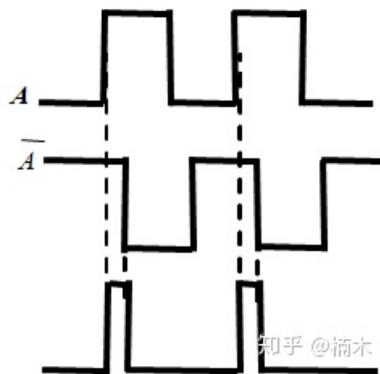
- 1.由功能抽象出逻辑表达式
- 2.化简逻辑表达式,比如卡诺图法
- 3.根据化简后的表达式画出电路

3.竞争-冒险现象

竞争：由于每级逻辑门存在延迟,信号经由不同的路径达到某一会合点的时间有先有后的现象

冒险：由于竞争而引起电路输出发生瞬间错误现象称为冒险，比如毛刺

产生条件：只要有互补变量进行与运算、或运算



解决办法

法一：增加乘积项,避免互补项相加

$$L = AC + B\bar{C}$$

知乎 @楠木

当A=B=1时，出现竞争-冒险现象

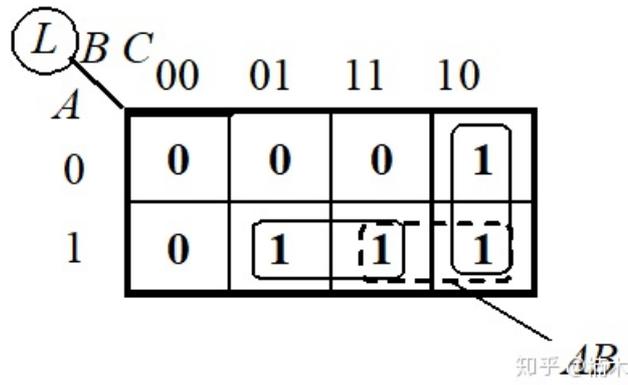
由卡诺图可知，这个表达式也可以写成

$$L = AC + B\bar{C} + AB$$

知乎 @楠木

当A=B=1时，L恒等于1，不会出现竞争-冒险现象

如何增加乘积项？把卡诺图相邻的圈圈圈起来即可



法二：输出端并联电容

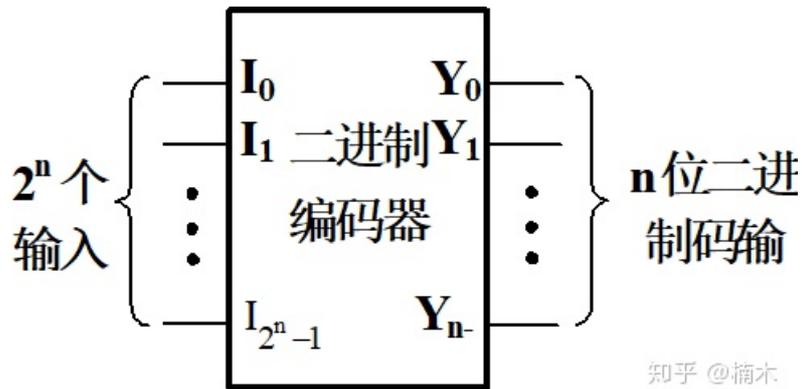
竞争-冒险现象相当于瞬间的高频信号，用电容滤波即可

4. 常见集成组合逻辑电路（重难点）

编码器

能将输入信号变换为不同的二进制的代码输出。如BCD编码器

普通编码器：只有一个输入信号为1，其他都是0，此时 2^n 个输入对应 n 个输出（多输入，少输出）



这种编码器致命缺点在于，不能同时有两个1输入。用这种编码器做键盘，只能每次按一个键.....

优先编码器：规定优先级，解决上述问题

输 入									输 出				
EI	I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	Y_2	Y_1	Y_0	GS	EO
L	×	×	×	×	×	×	×	×	L	L	L	L	L
H	L	L	L	L	L	L	L	L	L	L	L	L	H
H	H	×	×	×	×	×	×	×	H	H	H	H	L
H	L	H	×	×	×	×	×	×	H	H	L	H	L
H	L	L	H	×	×	×	×	×	H	L	H	H	L
H	L	L	L	H	×	×	×	×	H	L	L	H	L
H	L	L	L	L	H	×	×	×	L	H	H	H	L
H	L	L	L	L	L	H	×	×	L	H	L	H	L
H	L	L	L	L	L	L	H	×	L	L	H	H	L
H	L	L	L	L	L	L	L	H	L	L	L	H	L

优先编码器的功能表里有"×"符号，用来表示优先级。比如当I7为高电平时，其他信号为什么都不重要了，说明I7相对后面优先级最高。

这其实是集成电路CD4532的功能表。

EI是使能信号，它为低电平时，整个电路不工作，输出全为低电平

若EI为高电平，但是没有I信号输入，相当于键盘通电，但是没有按键被按下，此时只有EO输出为1

当有按键被按下时，GS=1,EO=0，按照优先级顺序处理输入信号I，从而输出相应的信号

译码器

译码器是普通编码器的逆过程，*可以将n个输入转变为2n个输出，少输入，多输出*。输出信号只有一个是1，其他都是0。所以译码器又叫唯一地址译码器

输 入						输 出							
E_3	\bar{E}_2	\bar{E}_0	A_2	A_1	A_0	\bar{Y}_0	\bar{Y}_1	\bar{Y}_2	\bar{Y}_3	\bar{Y}_4	\bar{Y}_5	\bar{Y}_6	Y_7
×	H	×	×	×	×	H	H	H	H	H	H	H	H
×	X	H	×	×	×	H	H	H	H	H	H	H	H
L	×	×	×	×	×	H	H	H	H	H	H	H	H
H	L	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	L	H	H	H	H	H	L	H	H	H	H
H	L	L	H	L	L	H	H	H	H	L	H	H	H
H	L	L	H	L	H	H	H	H	H	H	L	H	H
H	L	L	H	H	L	H	H	H	H	H	H	L	H
H	L	L	H	H	H	H	H	H	H	H	H	L	H

译码器也有使能信号，只有使能信号全都满足时，电路才会正常工作。还要特别注意输出端是否取非。*译码器的输出实际上是输入的信号变量最小项*，利用这点可以通过控制输入来生成逻辑函数

数据选择器

有多个信号输入，需要选择一个输出

靠什么来选择呢？靠"选择信号"

比如有四个信号输入，那么"选择信号"可以用2位二进制表示，2位二进制可以表示四种状态，即可以确定是4个中的哪一个信号输出"选择信号"就是译码器！，n位"选择信号"对应2n种状态（少对应多），就可以选择2n个输入信号中的一个进行输出

输 入				输 出	
使能 \overline{E}	选 择			Y	\overline{Y}
	S ₂	S ₁	S ₀		
H	X	X	X	L	H
L	L	L	L	D ₀	$\overline{D_0}$
L	L	L	H	D ₁	$\overline{D_1}$
L	L	H	L	D ₂	$\overline{D_2}$
L	L	H	H	D ₃	$\overline{D_3}$
L	H	L	L	D ₄	$\overline{D_4}$
L	H	L	H	D ₅	$\overline{D_5}$
L	H	H	L	D ₆	$\overline{D_6}$
L	H	H	H	D ₇	$\overline{D_7}$

"选择信号"译码后，实际上是"选择信号"的逻辑函数最小项。众多最小项中只有一个最小项 m_i 是1，所以与输入信号D相乘后，输出信号 $Y=m_1D_1+m_2D_2+\dots+m_iD_i+\dots=D_i$

从另一个角度看，由于 $Y=m_1D_1+m_2D_2+\dots+m_iD_i+\dots$ ，那么通过控制 D_i 就可以得到多个最小项的或

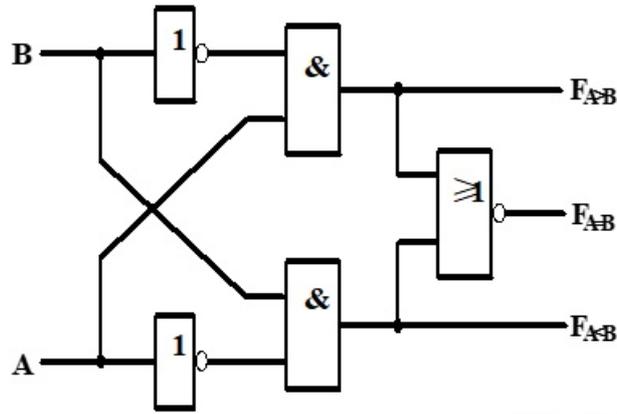
数值比较器（比较简单）

一位数值比较器

直接列出真值表

输 入		输 出		
A	B	$F_{A>B}$	$F_{A<B}$	$F_{A=B}$
0	0	0	0	1
0	1	0	1	0
1	0	1	0	0
1	1	0	0	1

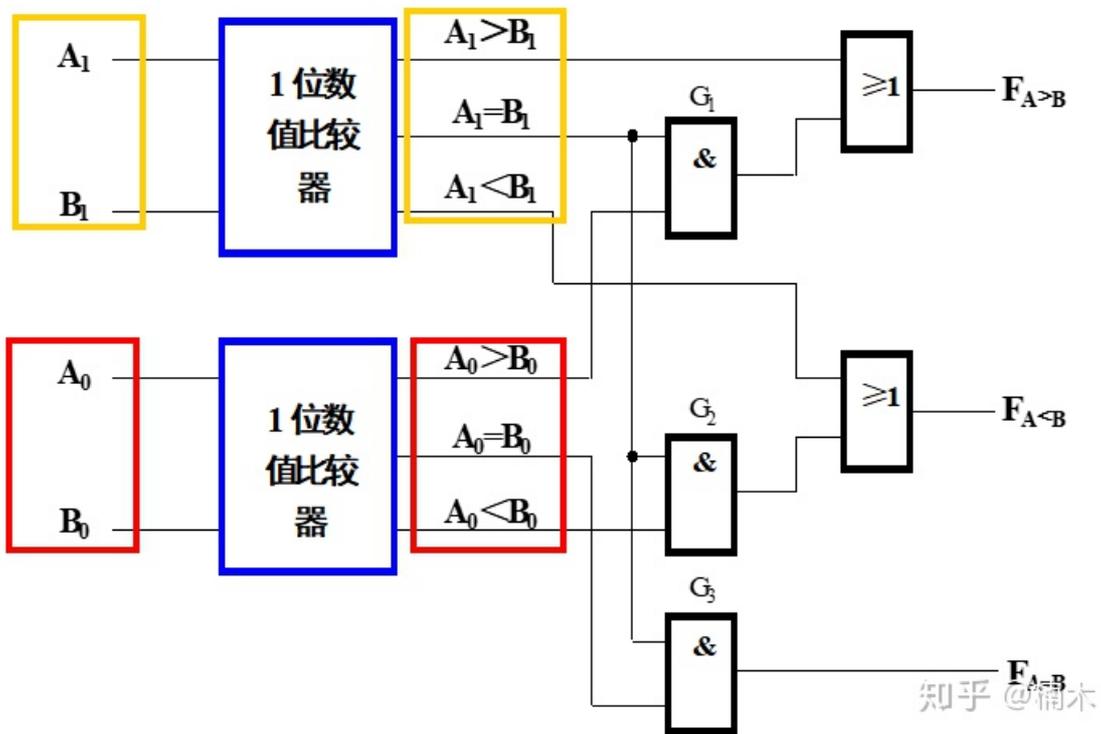
根据真值表画出电路即可



知乎 @楠木

两位数值比较器

先比高位，再比低位。可以用一位数值比较器拓展而成



知乎 @楠木

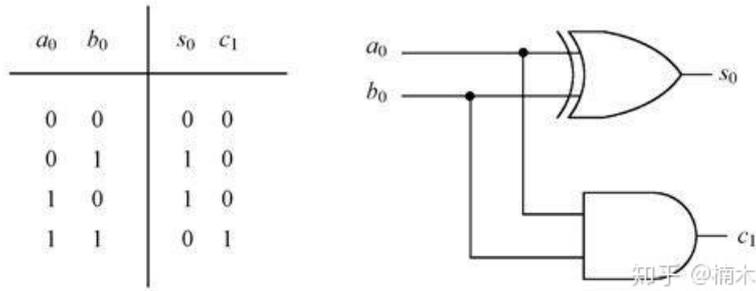
多位数值比较器

可以通过串联或者并联形式，前者节约门电路，后者节约时间，太简单不再赘述。

加法器（比较简单）

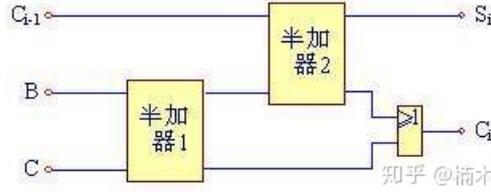
半加器

直接列出真值表画出电路



全加器

比较简单，先加低位再进位到高位



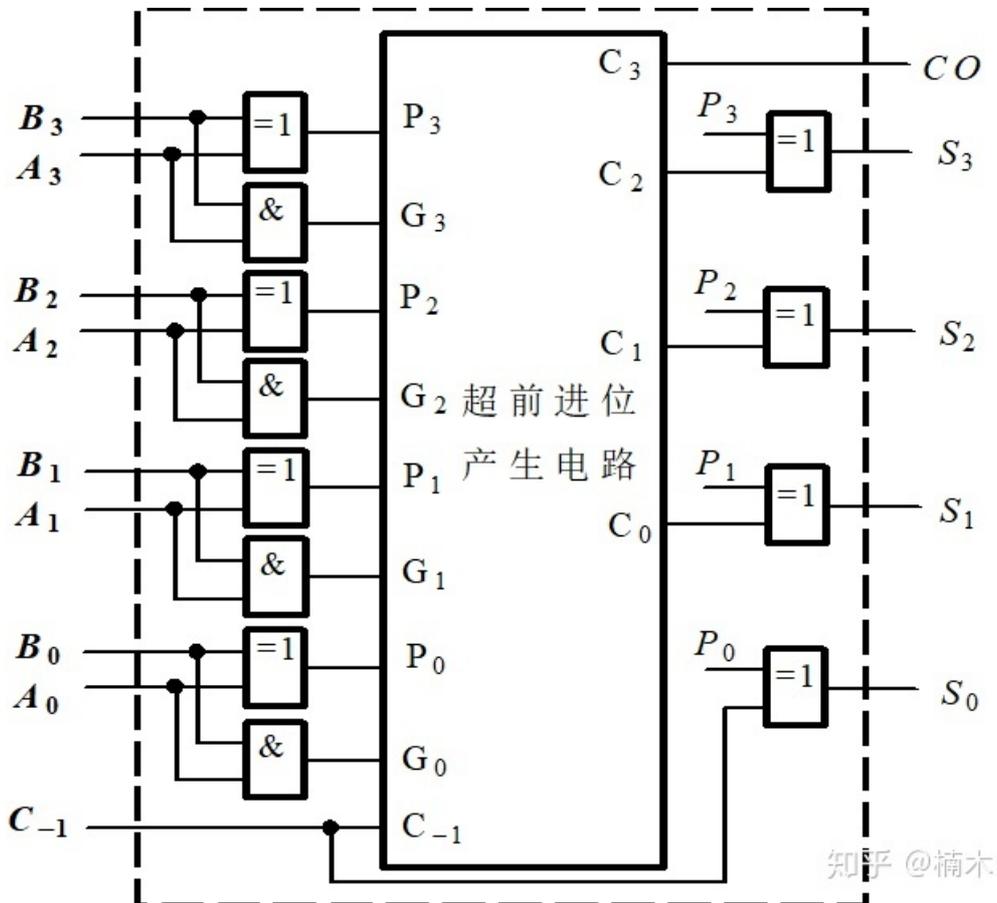
多位加法器

串联法：过于简单不赘述

超前进位加法器：不需要等低位进位，因为加数输入时，进位数就已经确定，直接求得进位数即可显然，第*i*位的进位数为：

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

求出carry bit，剩下的就相当于无进位的全加器

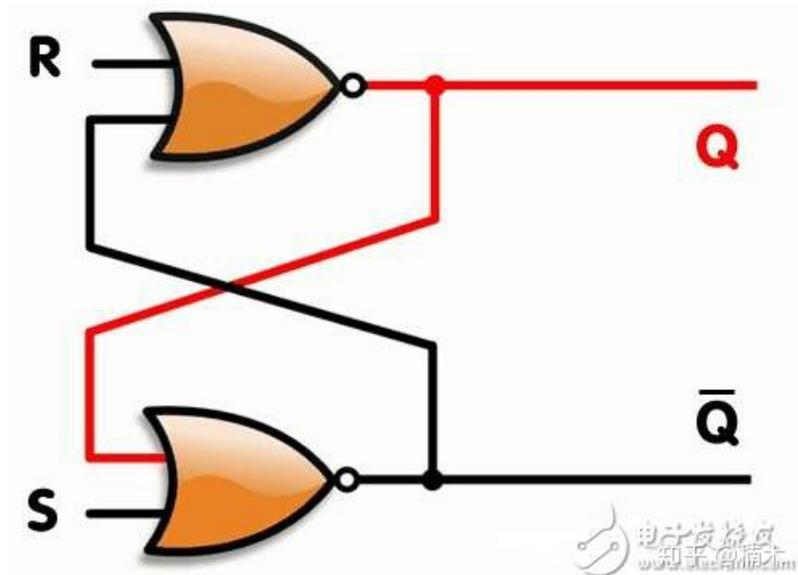


第五章 锁存器和触发器（重点）

这类电路具有“记忆功能”，是组合逻辑电路没有的

1.锁存器

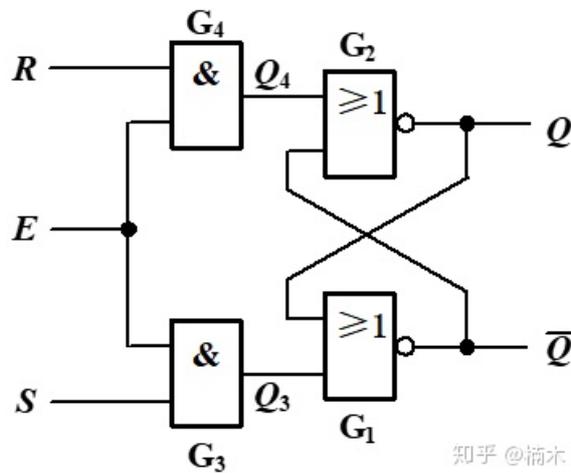
SR锁存器（两个或非门组成）



锁存器的真值表如下

R(Reset)S(Set)Q00保持原来的Q01110011状态不定，不允许这种状态

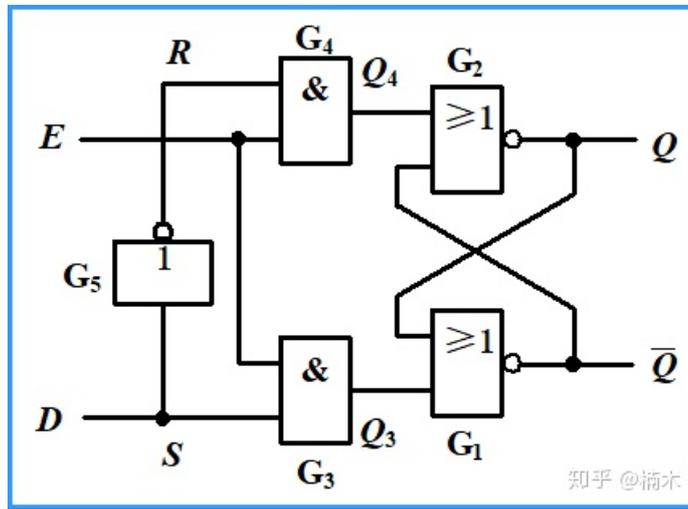
逻辑门控SR锁存器（加个使能信号）



D(Data)锁存器

逻辑门控D锁存器

其实没必要上面那么复杂，当E=0时就是保持不变的状态。所以R和S信号就是相反的关系，可以用一个非门接在一起。这就是D锁存器



只要电路工作，即 $E=1$ ，那么它的原理就和RS锁存器一样。

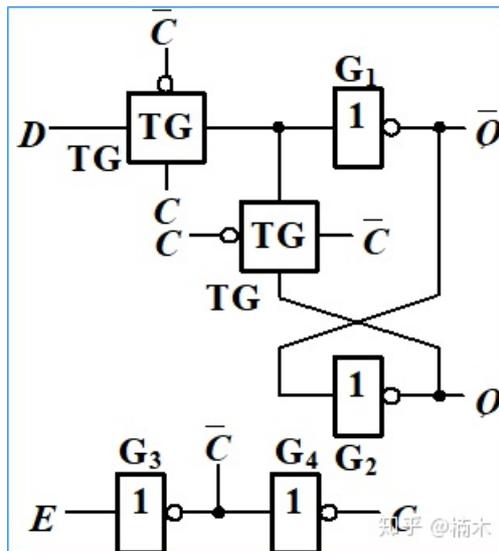
如果 $D=1$ ，相当于 $R=0$ 且 $S=1$ ，那么 $Q=1$ (置1)
 如果 $D=0$ ，相当于 $R=1$ 且 $S=0$ ，那么 $Q=0$ (置0)

有很多方式可以描述电路的功能，比如真值表、波形图等。这一章引入特性方程这一新方式

D锁存器的特性方程： $Q=D$ (电路工作时)

之后还会引入状态转换图来描述电路

传输门控D锁存器



传输门TG的作用就是，当电压给对了，D信号就会传过去

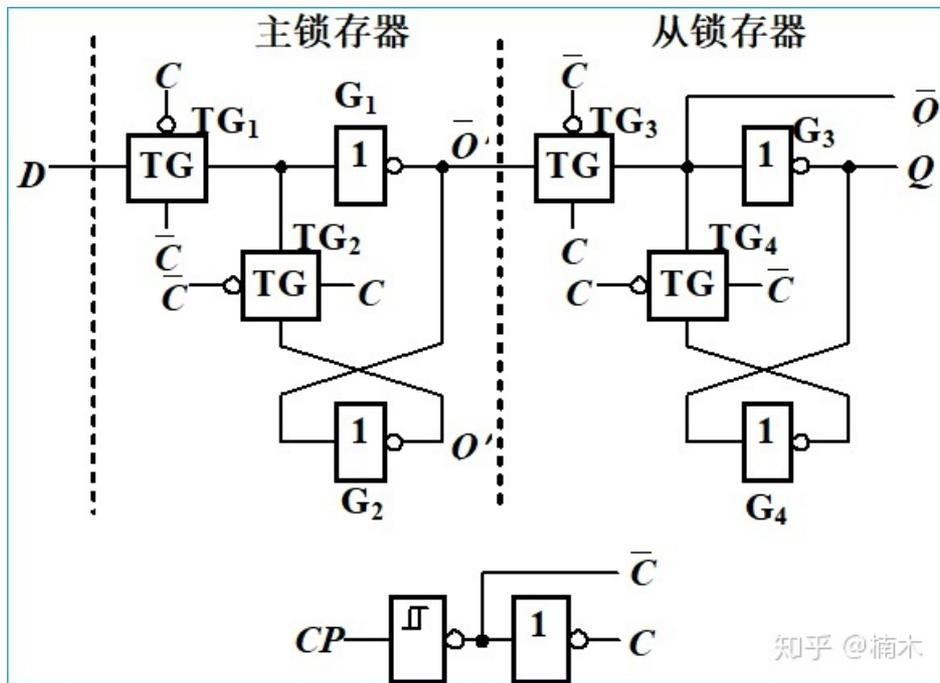
$E=1$ 时，TG1导通，TG2断开， $Q=D$

$E=0$ 时，TG2导通，TG1断开，Q保持不变

2. 触发器

刚才讲了锁存器，锁存器是在某个信号处于高电平或者低电平的时候工作，比如使能信号E

但是触发器不一样，触发器是在某个信号处于上升沿或者下降沿的时候工作，这里一般是信号CP



分析如下

CP=0时，输入信号D送入主锁存器，使Q'=D

CP由0跳变到1时，从锁存器TG3打开，Q'信号送Q到端

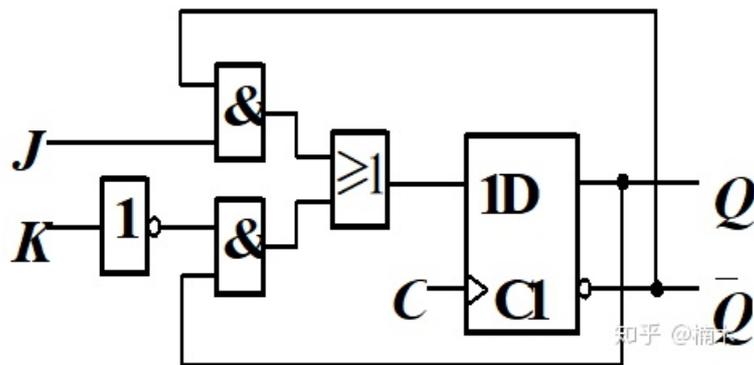
Q'由CP跳变前的电路决定。CP跳变后，Q=Q'。换句话说，触发器的状态仅仅取决于CP信号上升沿到达前瞬间的D信号

D触发器

由两个D锁存器构成的触发器，就是上图，这里不再分析

JK触发器

我们把D触发器封装一下，输入信号是D和C，输出Q和Q。再加上几个门电路，就构成了JK触发器



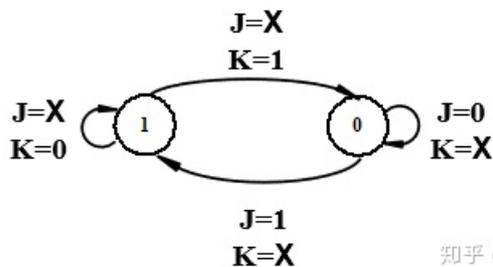
如果C信号有效时，比如上升沿到来
那么可以写出JK触发器的状态方程

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

当J和K取不同值的时候，电路就可以完成不同的功能。

例如，上升沿到来前，J=K=1，那么有D=Q非。上升沿到来后，由D触发器的特性可知，D=Q，那么此时Q=Q非，实现翻转。特别注意，如果上升沿迟迟不到来，那么D也就只能卡在或门的后面，不能进入D触发器从而改变Q

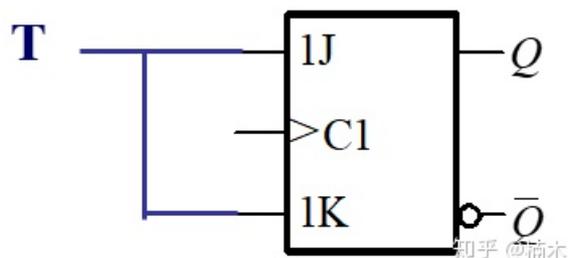
这边我们可以用状态转换图来描述这个电路



知乎 @楠木

T触发器和T'触发器

T触发器只需要将JK触发器的J和K接在一起就好了



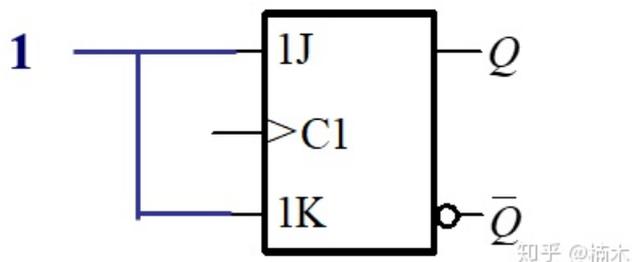
知乎 @楠木

写出方程如下

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

知乎 @楠木

T'触发器更简单了，直接把T触发器中的T接到1



知乎 @楠木

写出方程如下

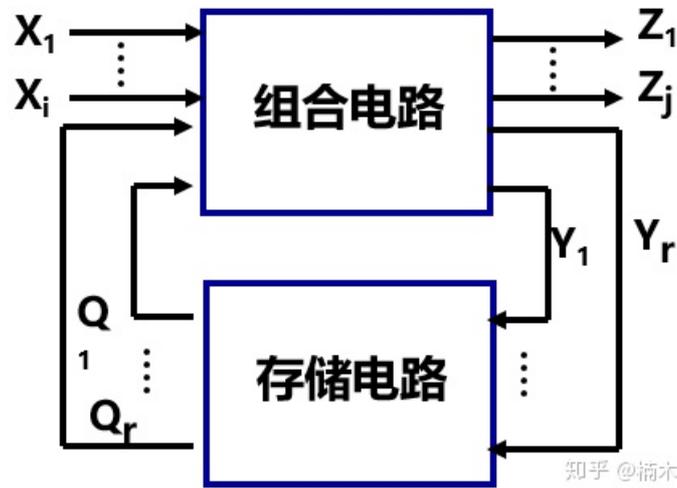
$$Q^{n+1} = \overline{Q}^n$$

当上升沿来之后，Q翻转，保持住，直到下一次上升沿到来

总结一下，本章的内容分为锁存器和触发器。内容不难，特别要注意电路工作的条件！

第六章 时序电路

1. 时序电路基本组成和分类



上图是时序电路的基本电路结构，存储部分由触发器构成

在时序电路中，要弄懂电路的功能，需要三组方程，分别是输出方程、状态方程、激励方程。输出方程描述输出信号 Z 与输入信号 X 和当前电路状态 Q 的关系，即 $Z = F(X, Q_n)$

如果输出信号 Z 是 Q 的单值函数，该电路是摩尔型电路；如果 Z 是二元函数，该电路是米里型电路。这本教材讨论摩尔型电路激励方程描述触发器的输入 Y 与 X 和 Q 的关系，即 $Y = F(X, Q_n)$ 。状态方程描述触发器下一状态与当前状态和 Y 的关系，即 $Q_{n+1} = F(Y, Q_n)$ 。激励方程和状态方程可以通过触发器特性方程联系起来。

2. 同步时序电路分析

1. 大致了解电路构成，判断触发器类型，写出特性方程
2. 写出激励方程和输出方程（也许不需要）
3. 把激励方程代入特性方程，就可以写出状态方程，即 Q_{n+1} 的表达式
4. 画出状态转换表
5. 画出状态转换图，确定功能
6. 可以关心一下电路的自启动问题

3. 同步时序逻辑电路的设计

1. 逻辑抽象，建立最初的状态图或者状态表
2. 状态化简，合并等价状态（如果在相同的输入下输出相同或者相互变化就叫等价）
3. 状态分配， n 个触发器可以确定 2^n 个状态 (highly encoding)
4. 选择触发器的类型
5. 求出电路的激励方程和输出方程
6. 画出逻辑图并检查自启动能力