第8章 逻辑代数和逻辑门电路

本章要求

1、熟练掌握二进制、十六进制与十进制的相互转换，8421BCD码

2、熟练掌握基本逻辑运算和逻辑问题的描述

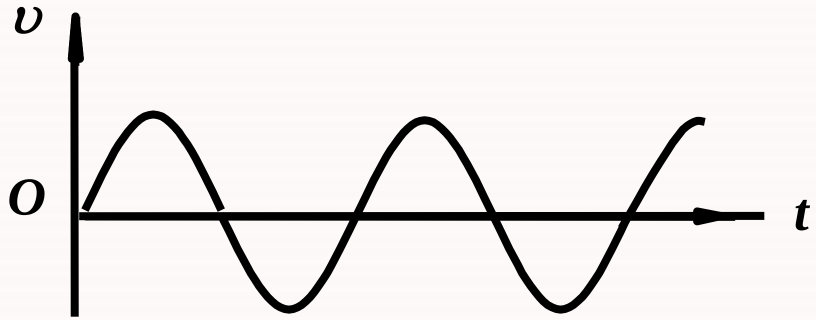
3、熟练掌握逻辑代数的基本定律和规则

4、熟练掌握逻辑函数的代数化简法和卡诺图化简法，逻辑函数的变换

8.1概 述

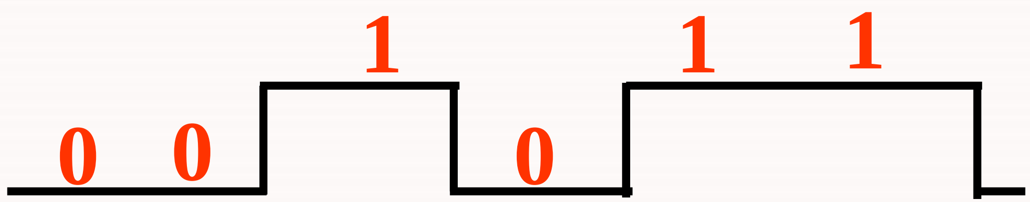
模拟信号

正弦波



- 研究模拟信号时，我们注重电路输入、输出信号间的大小、相位关系。
- 在模拟电路中，晶体管一般工作在放大状态。

数字信号——在时间上和数值上均是离散的。如电子表的秒信号，生产线上记录零件个数的记数信号等。



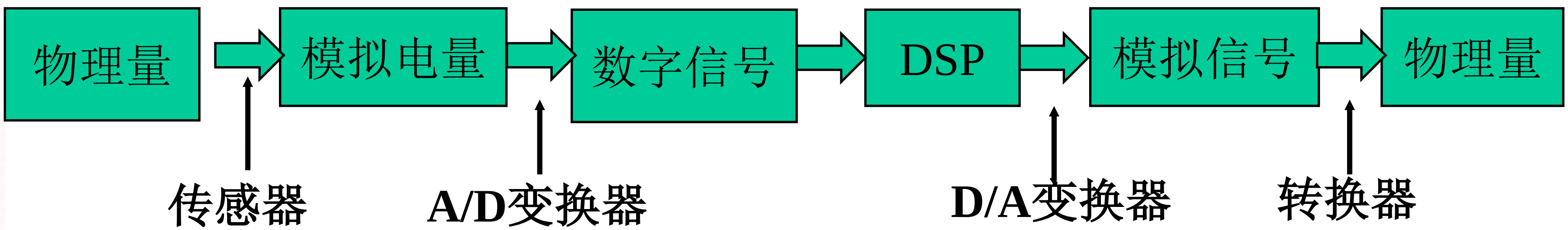
- 数字电路：处理数字信号的电路。
- 研究数字电路时注重电路输出、输入间的逻辑关系。
- 三极管工作在开关状态，即饱和和截止状态。

- 模拟电路的地位和作用

  1、在很高工作频率的信号还只能由模拟电路处理，因为数字电路的工作速度不够；

  2、微弱信号的放大数字电路不能完成；

  3、与物理世界的接口，我们经常接触的物理量是模拟的，即使采用数字信号处理，也必须经过一定的模拟信号处理；



  4、与传输介质接口，信号在各种传输介质中的传播，因为主要是载波传输，主要应用模拟信号；

  5、易于实现各种非线性电路，如相乘器等；

- 完整的电子系统是模拟—数字混合系统，模拟电路的性能严重地影响整个系统的性能。

数字电路的特点：

1、在数字电路中，只有高、低两种电平分别用1，0表示。

2、抗干扰能力强，可靠性和准确性高，对元件精度要求不高。

3、数字电路能够对输入的数字信号进行各种算术运算和逻辑运算，具有一定的“逻辑思维”能力，易于实现各种控制和决策应用系统。具可编程性,可实现硬件设计软件化

4、数字信号易于存储。

5、集成度高。

数字电路和数字信号应用远超过模拟电路和模拟信号

数字技术的发展及其应用

60~70代- IC技术迅速发展：SSI、MSI、LSI 、VLSI。10万个门/片。

80年代后- ULSI， 1 0 亿个门/片 、 ASIC 制作技术成熟

90年代后- 97年一片集成电路上有40亿个门。

目前-- 芯片内部的布线细微到纳米(10^{-9}μm)量级

酷睿 i5-8250U处理器的时钟频率高达3.8GHz（10^9Hz）

将来- 高分子材料或生物材料制成密度更高、三维结构的电路

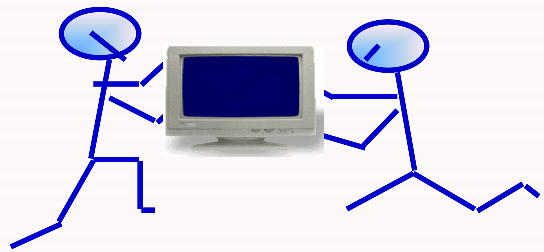
数字集成电路的集成度

一块芯片中含有等效逻辑门或元器件的个数

小规模集成电路 SSI (Small Scale Integration)	< 10 门/片 或 < 100 元器件/片
中规模集成电路 MSI (Medium Scale Integration)	10 ~ 99 门/片 或 100 ~ 999 元器件/片
大规模集成电路 LSI (Large Scale Integration)	100 ~ 9 999 门/片 或 1 000 ~ 99 999 元器件/片
超大规模集成电路 VLSI (Very Large Scale Integration)	> 10 000 门/片 或 > 100 000 元器件/片

计算机体系结构和组织

软件

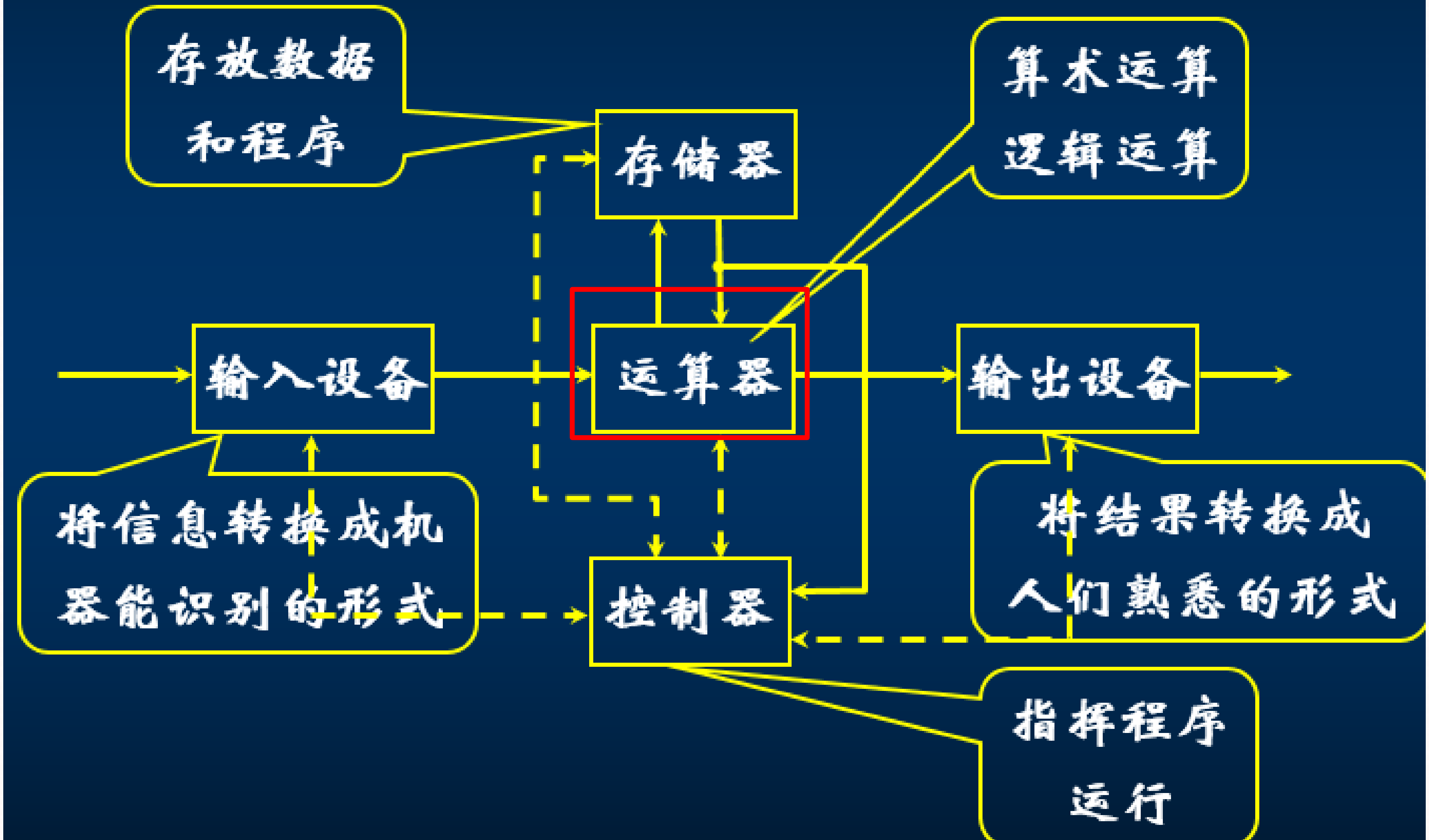


指令集（instruction set）

硬件

计算机体系结构概念的实质是计算机系统中**软硬件界面**的确定，其界面之上的是软件的功能，界面之下的是硬件和固件的功能。

冯·诺依曼计算机硬件框图



（1）控制器（Control unit）：是分析和执行指令的部件，也是统一指挥并控制计算机各部件协调工作的中心部件，所依据的是机器指令。控制器的组成包含程序计数器（PC）、指令寄存器（IR）、指令译码器、时序部件、微操作控制信号形成部件（PSW）和中断机构。

（2）运算器：也叫做算术逻辑单元（Arithmetic and Logic Unit，ALU），对数据进行算术运算和逻辑运算。通常由ALU（算术/逻辑单元，包括累加器、加法器等）、通用寄存器（不包含地址寄存器）、多路转换器、数据总线组成。

总结：计算机的硬件结构

- 归纳如下：
- 采用二进制形式表示数据和指令
- 采用存储程序方式
- 由运算器、存储器、控制器、输入设备和输出设备五大部件组成计算机硬件系统。

课程特点：数字电路是一门技术基础课程，它是学习微机原理、接口技术等课程的基础。既有丰富的理论体系，又有很强的实践性。

内容：（1）逻辑代数基础（逻辑运算，逻辑代数，化简逻辑函数）
（2）组合逻辑电路（编码器、译码器、加法器、比较器等）
（3）时序逻辑电路（计数器、寄存器）
（4）其它内容（555，A/D D/A)

课程重点：数字电路的分析、设计

分析方法：逻辑代数、真值表、卡诺图、时序图等等。

8.1.4 几种常用的数制和码制

数制：

$$M_R=\sum_{i=-\infty}^{\infty}a_iR^i$$

十进制　$N_D=\sum_{i=-\infty}^{\infty}K_i\times 10^i$

二进制　$N_B=\sum_{i=-\infty}^{\infty}K_i\times 2^i$

十六进制　$N_H=\sum_{i=-\infty}^{\infty}K_i\times 16^i$

其中，R成为计数的基数，a_i为第i位的系数，R i称为第i位的权。

应用场合：

二进制：　计算机内部运算；

十进制：　输入和输出显示；

十六进制：　二进制的书写。

1. 十进制（**Decimal**）-- 逢十进一

数码：$0 \sim 9$　　　位权：10^{i}

$$(143.75)_{10} = 1\times10^{2} + 4\times10^{1} + 3\times10^{0} + 7\times10^{-1} + 5\times10^{-2}$$

2. 二进制（**Binary**）-- 逢二进一

数码：0，1　　　位权：2^{i}

$$(1011)_{2} = 1\times2^{3} + 0\times2^{2} + 1\times2^{1} + 1\times2^{0}$$

$$(101.11)_{2} = 1\times2^{2} + 0\times2^{1} + 1\times2^{0} + 1\times2^{-1} + 1\times2^{-2}$$

3. 十六进制 (Hexadecimal) --逢十六进一

数码：0 ~ 9 , A(10) , B(11) , C(12) , D(13) , E(14) , F(15)

位权：16^{i}

$$(2A.7F)_{16} = 2\times16^{1} + 10\times16^{0} + 7\times16^{-1} + 15\times16^{-2}$$

任意(N)进制数展开式的普遍形式： $D = \sum k_i N^{i}$

k_i — 第 i 位的系数　　N^{i} — 第 i 位的权

4. 几种常用进制数之间的转换

(1) 二-十转换： 将二进制数按位权展开后相加

$$(101.11)_2 = 1\times2^2 + 0\times2^1 + 1\times2^0 + 1\times2^{-1} + 1\times2^{-2}$$

$$= 4 + 1 + 0.5 + 0.25 = (5.75)_{10}$$

(2) 十-二转换:

整数的转换--连除法

$$(26)_{10} = (11010\ )_2$$

$$\begin{array}{r|l l}
2 & 26 & \text{余数} \\ \hline
2 & 13 & 0 \\ \hline
2 & 6 & 1 \\ \hline
2 & 3 & 0 \\ \hline
2 & 1 & 1 \\ \hline
 & 0 & 1
\end{array}$$

除基数得余数作系数从低位到高位

小数的转换--连乘法

$$(0.8125)_{10} = (0.\ 1101\ )_2$$

$$\begin{array}{r l}
0.8125 & \text{取整} \\
\times\quad\quad 2 & \\
\hline
1.6250 & 1 \\
0.6250 & \\
\times\quad\quad 2 & \\
\hline
1.2500 & 1 \\
0.2500 & \\
\times\quad\quad 2 & \\
\hline
0.5000 & 0 \\
\times\quad\quad 2 & \\
\hline
1.0000 & 1
\end{array}$$

若小数在连乘多次后不为 0，一般按照精确度要求(如小数点后保留 *n* 位)得到 *n* 个对应位的系数即可。

乘基数取整数作系数从高位到低位

快速转换法：拆分法

$$(26)_{10} = 16 + 8 + 2 = 2^4 + 2^3 + 2^1 = (\overset{16}{1}\ \overset{8}{1}\ \overset{4}{0}\ \overset{2}{1}\ \overset{1}{0})_2$$

（3）二-十六转换：

每 4 位二进制数相当一位 16 进制数

$$(26)_{10} = (\underbrace{0001}_{1}\ \underbrace{1010}_{A})_2 = (1A)_{16}$$

$$(0001\ 1011\ 0110.0010)_2 = (1B6.2)_{16}$$

（4）十六-二转换：

每位 16 进制数换为相应的 4 位二进制数

$$(8FA.C6)_{16} = (1000\ 1111\ 1010.1100\ 0110)_2$$

$$(ED8.2F)_{16} = (1110\ 1101\ 1000.0010\ 1111)_2$$

三、二进制代码

编码： 用二进制数表示文字、符号等信息的过程。

二-十进制代码： 用二进制代码表示十个数字符号 0 ~ 9，又称为 **BCD** 码（**Binary Coded Decimal**）

从4 位二进制数16种代码中,选择10种来表示0~9个数码的方案有很多种。每种方案产生一种BCD码。

几种常见的BCD代码：
- **8421码**　**2421码**　**5211码**
- 余 3 码　余 3 循环码

其他代码： **ISO** 码，**ASCII**（美国信息交换标准代码）

十进制数	几种常见的 BCD 代码				
	8421 码	余 3 码	2421(A)码	5211 码	余3循环码
0	0 0 0 0	0 0 1 1	0 0 0 0	0 0 0 0	0 0 1 0
1	0 0 0 1	0 1 0 0	0 0 0 1	0 0 0 1	0 1 1 0
2	0 0 1 0	0 1 0 1	0 0 1 0	0 1 0 0	0 1 1 1
3	0 0 1 1	0 1 1 0	0 0 1 1	0 1 0 1	0 1 0 1
4	0 1 0 0	0 1 1 1	0 1 0 0	0 1 1 1	0 1 0 0
5	0 1 0 1	1 0 0 0	1 0 1 1	1 0 0 0	1 1 0 0
6	0 1 1 0	1 0 0 1	1 1 0 0	1 0 0 1	1 1 0 1
7	0 1 1 1	1 0 1 0	1 1 0 1	1 1 0 0	1 1 1 1
8	1 0 0 0	1 0 1 1	1 1 1 0	1 1 0 1	1 1 1 0
9	1 0 0 1	1 1 0 0	1 1 1 1	1 1 1 1	1 0 1 0
权	8 4 2 1		2 4 2 1	5 2 1 1	

例1: $(37.86)_{10} = (?)_{8421BCD}$

不能省略!

$= (0011,0111.1000,0110)_{8421BCD}$

一位十进制数，用四位二进制数表示。

例2: $(011000101000.10010101)_{8421BCD} = (?)_{10}$

四位二进制数,可以表示一位十进制数。

$= (0110，0010，1000.1001，0101)_{8421BCD}$

$= (628.95)_{10}$

[练习] 完成下列数制和码制之间的相互转换

1. $(37)_{10} = (\overset{32}{1}\,0\,0\,\overset{4}{1}\,0\,\overset{1}{1})_2$

2. $(2DE)_{16} = (\overset{512}{1}\,0\;\;\overset{128}{1}\,\overset{64}{1}\,0\,\overset{16}{1}\;\;\overset{8}{1}\,\overset{4}{1}\,\overset{2}{1}\,0)_2 = (734)_{10}$

3. $(151)_{10} = (\overset{128}{1}\,0\,0\,\overset{16}{1}\,0\,\overset{4}{1}\,\overset{2}{1}\,\overset{1}{1})_2 = (0001\;0101\;0001)_{8421BCD}$

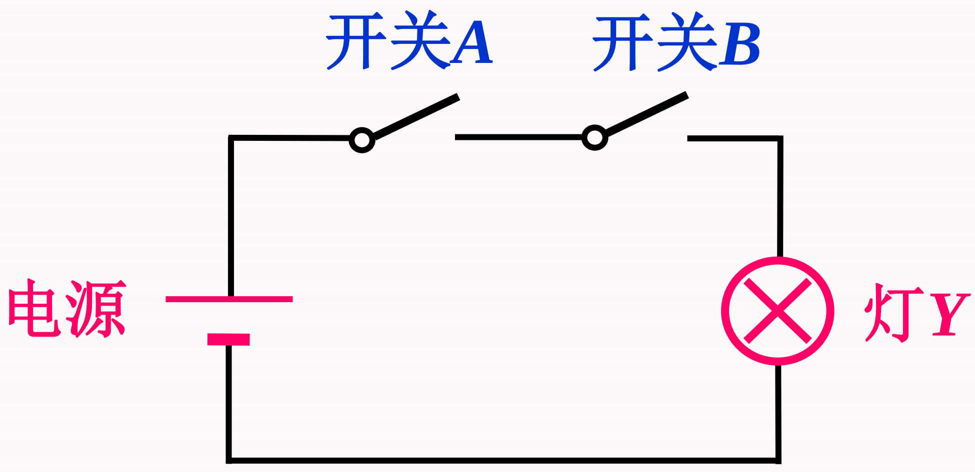
4. $(10\;\;1001)_{8421BCD} = (29)_D = (\overset{16}{1}\,\overset{8}{1}\,\overset{4}{1}\,0\,\overset{1}{1})_B$

8.2 逻辑运算及逻辑函数的表示方法

8.2.1 逻辑运算

一、三种基本逻辑运算

1. 与逻辑： 当决定一事件的所有条件都具备时，事件才发生的逻辑关系。



与逻辑关系

功能表

A	B	Y
断	断	灭
断	合	灭
合	断	灭
合	合	亮

与逻辑的表示方法：

真值表 （Truth table）

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

功能表

A	B	Y
断	断	灭
断	合	灭
合	断	灭
合	合	亮

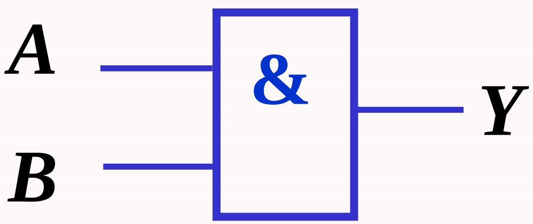
逻辑函数式

$$Y = A \cdot B = AB$$

输出逻辑变量

输入逻辑变量

逻辑符号



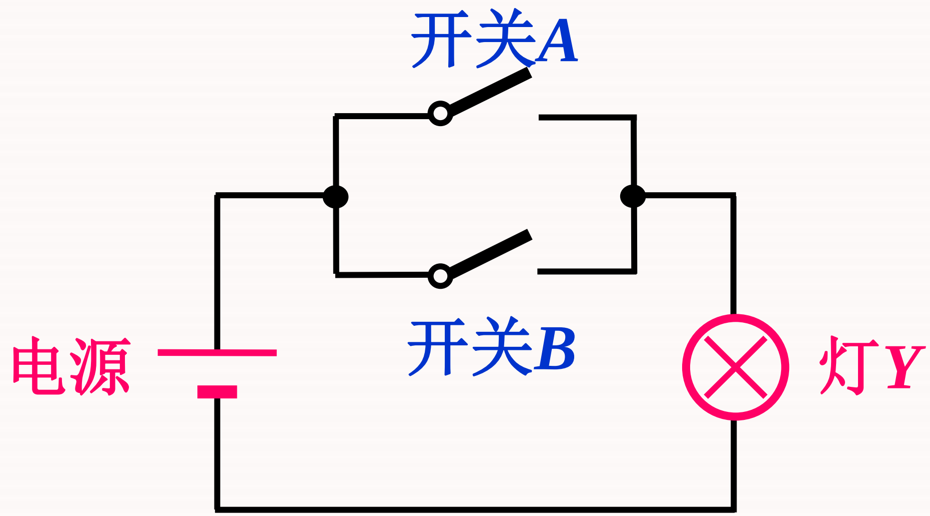
与门（AND gate)

2. 或逻辑：

决定一事件结果的诸条件中，只要有一个或一个以上具备时，事件就会发生的逻辑关系。

真值表

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

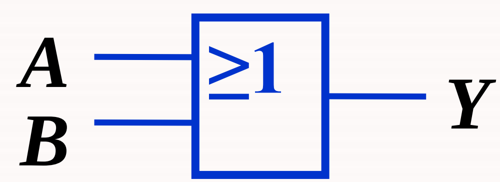


或逻辑关系

逻辑函数式

$$Y = A + B$$

逻辑符号



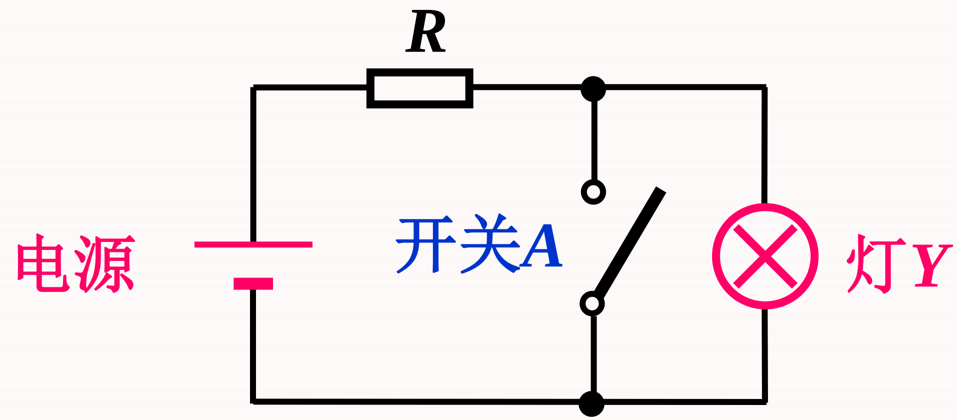
或门（OR gate）

3. 非逻辑：

只要条件具备，事件便不会发生；条件不具备，事件一定发生的逻辑关系。

真值表

A	Y
0	1
1	0



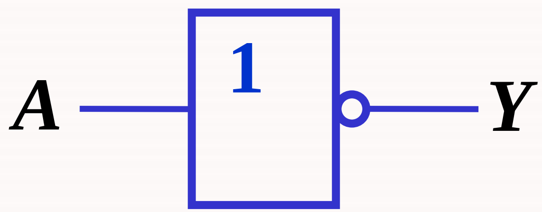
非逻辑关系

逻辑函数式

$$Y = \overline{A}$$

字母上面无反号的称为原变量，有反号的叫做反变量(非变量)。

逻辑符号



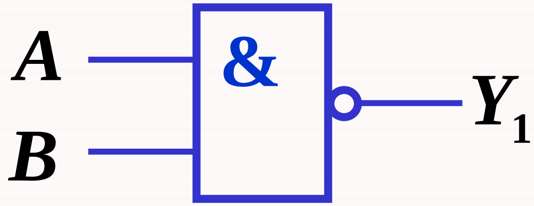
非门（NOT gate)

二. 几种常用复合逻辑运算

(1) 与非逻辑

(NAND)

$$Y_1 = \overline{AB}$$



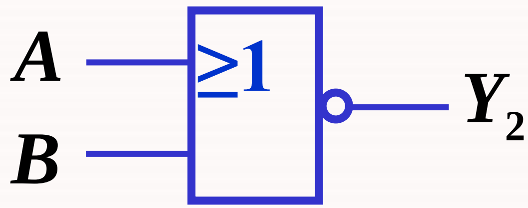
Y_1、Y_2的真值表

A	B	Y_1	Y_2
0	0	1	1
0	1	1	0
1	0	1	0
1	1	0	0

(2) 或非逻辑

(NOR)

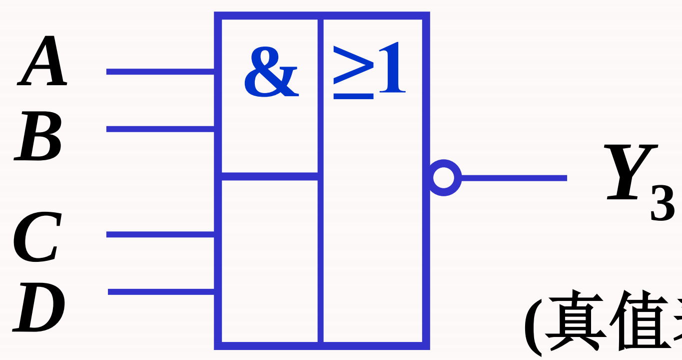
$$Y_2 = \overline{A + B}$$



(3) 与或非逻辑

(AND – OR – INVERT)

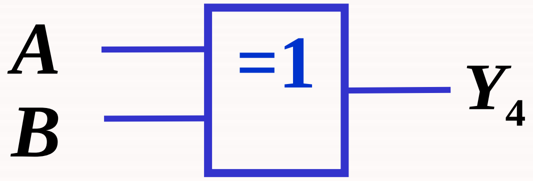
$$Y_3 = \overline{AB + CD}$$



(真值表略)

(4) 异或逻辑

(Exclusive—OR)



A	B	Y_4
0	0	0
0	1	1
1	0	1
1	1	0

$$Y_4 = A \oplus B = A\overline{B} + \overline{A}B$$

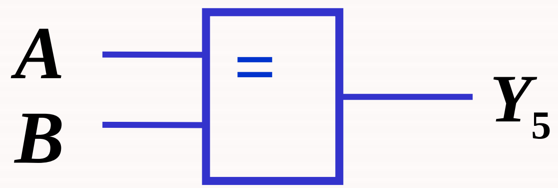
(5) 同或逻辑 （异或非）

(Exclusive—NOR)

$$Y_5 = \overline{A \oplus B}$$

$$= \overline{A}\,\overline{B} + AB$$

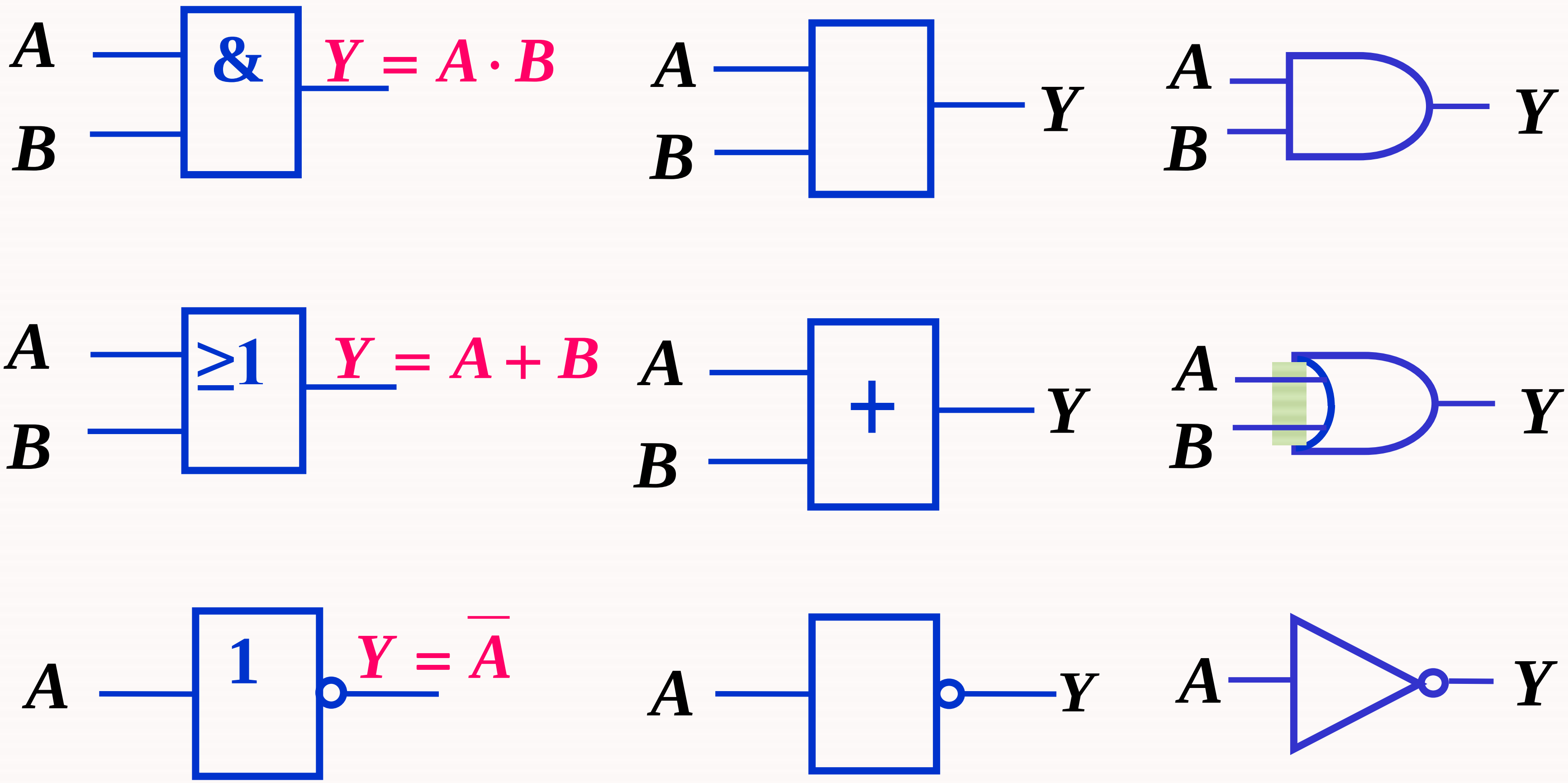
$$= A \odot B$$



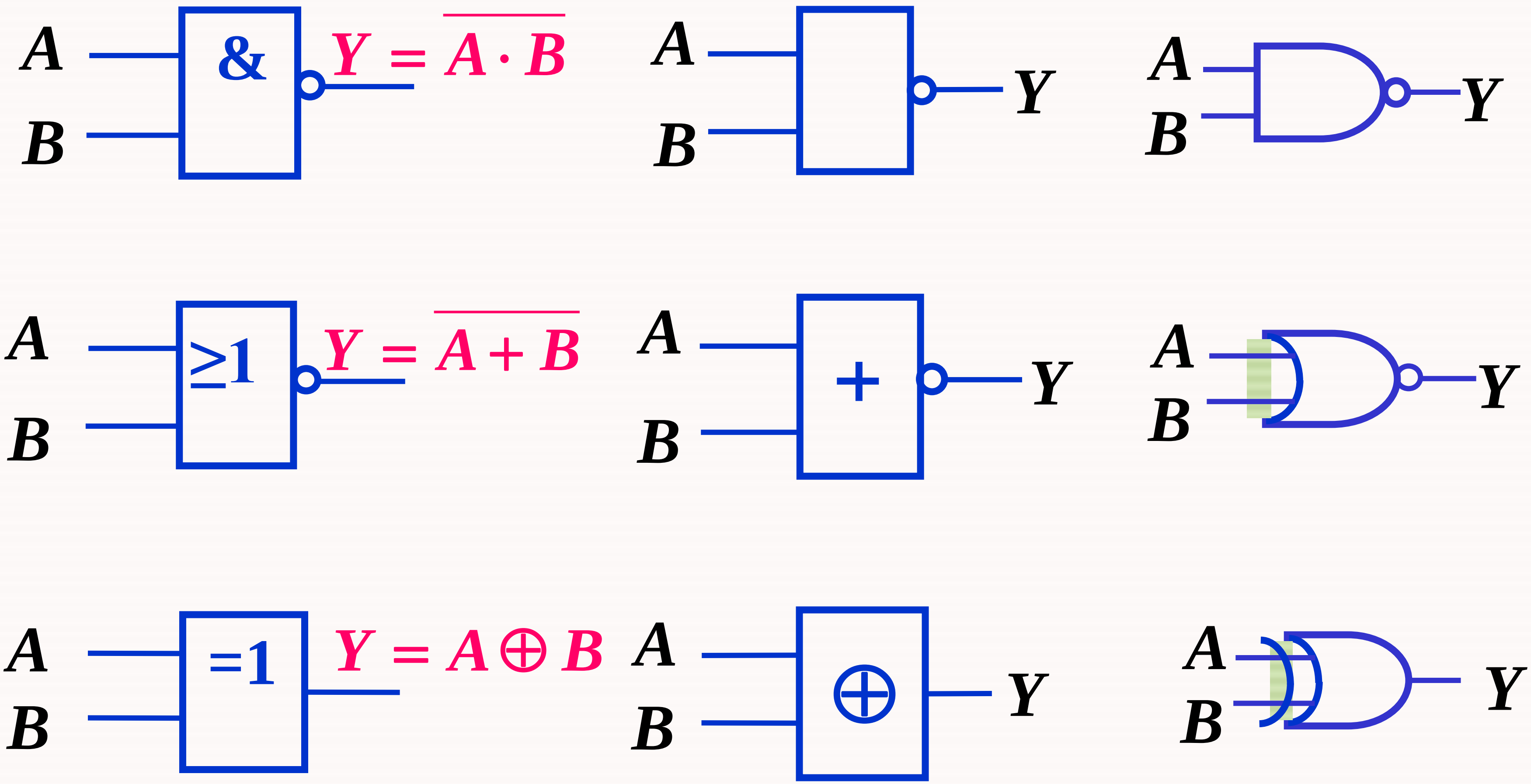
A	B	Y_5
0	0	1
0	1	0
1	0	0
1	1	1

3. 逻辑符号对照

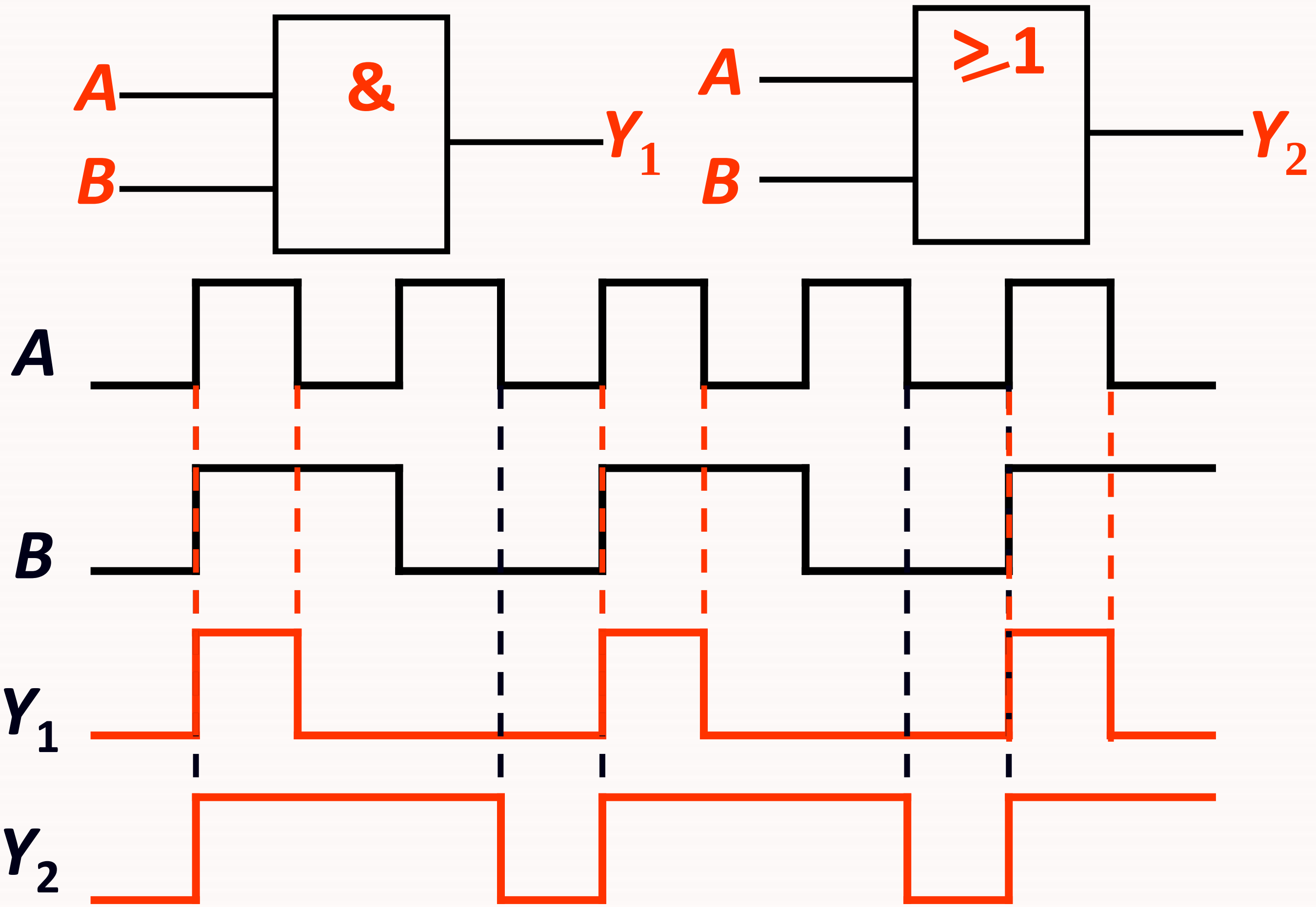
国标符号　　曾用符号　　美国符号



国标符号 曾用符号 美国符号



例：根据输入波形画出输出波形



有“1”出“1”，全“0”出“0”

8.2.2 逻辑函数的建立及其表示方法

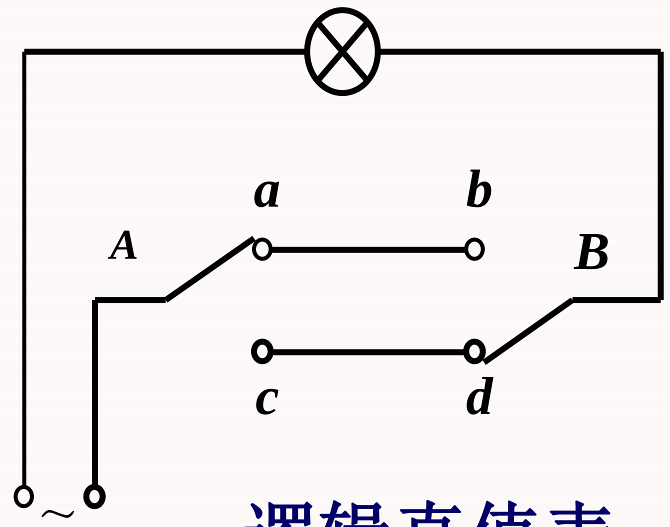
1. 真值表表示

逻辑抽象，列出真值表

开关状态表

开关 *A*	开关 *B*	灯
下	下	亮
下	上	灭
上	下	灭
上	上	亮

楼道灯开关示意图



确定变量、函数，并赋值

开关:　变量 A、B
灯　:　函数 L

A、B: 向上—1 向下--0
L　: 亮---1; 　灭---0

逻辑真值表

A	*B*	*L*
0	0	1
0	1	0
1	0	0
1	1	1

2、逻辑函数表达式表示

逻辑表达式是用与、或、非等运算组合起来，表示逻辑函数与逻辑变量之间关系的逻辑代数式。

例：已知某逻辑函数的真值表，试写出对应的逻辑函数表达式。

$$L = \overline{A}\,\overline{B} + AB$$

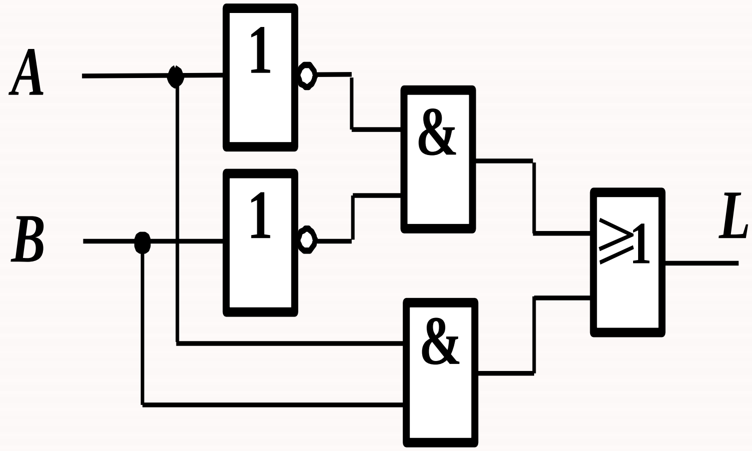
逻辑真值表

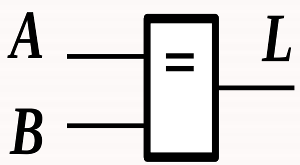
A	B	L
0	0	1
0	1	0
1	0	0
1	1	1

3. 逻辑图表示方法

用与、或、非等逻辑符号表示逻辑函数中各变量之间的逻辑关系所得到的图形称为逻辑图。

例：已知某逻辑函数表达式为 $L=\overline{A}\,\overline{B}+AB$ ，试画出其逻辑图



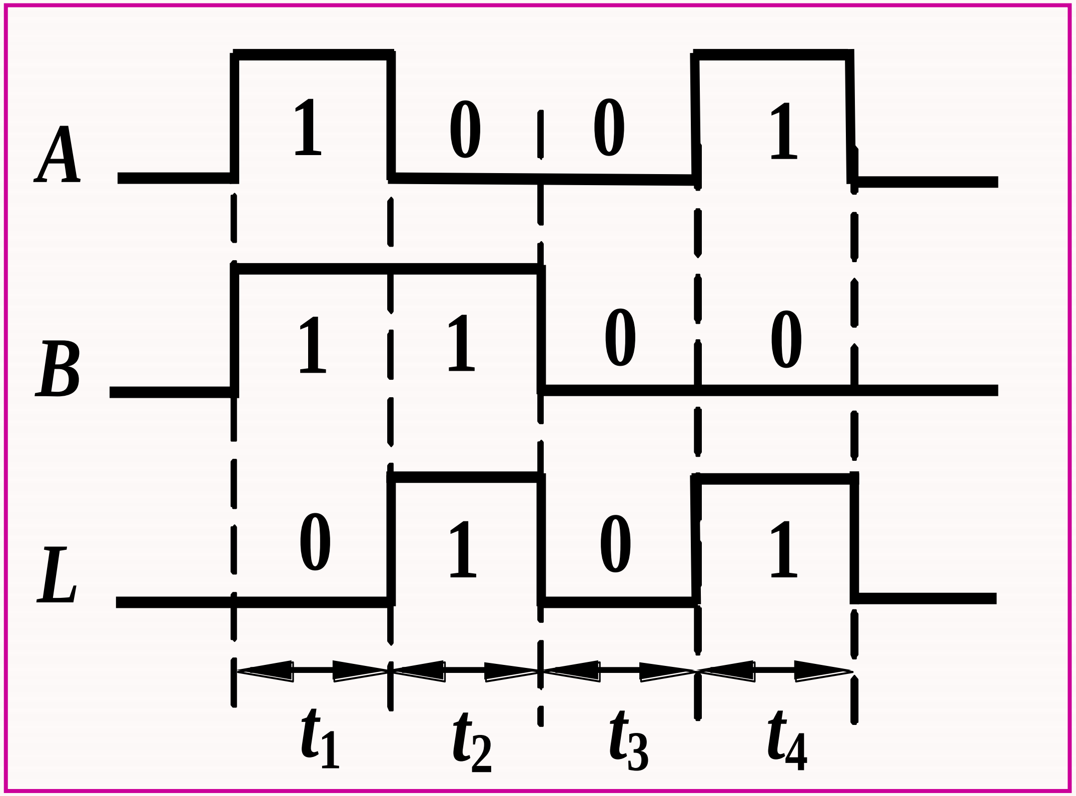


4. 波形图表示方法

用输入端在不同逻辑信号作用下所对应的输出信号的波形图，表示电路的逻辑关系。

真值表

A	B	L
0	0	0
0	1	1
1	0	1
1	1	0



8.3 逻辑代数

8.3.1 公式和定理

一、 常量之间的关系(常量：0 和 1)

与：$0 \cdot 0 = 0$ 或：$1 + 1 = 1$ 非：$\overline{0} = 1$

$0 \cdot 1 = 0$ $1 + 0 = 1$ $\overline{1} = 0$

$1 \cdot 1 = 1$ $0 + 0 = 0$

二、变量和常量的关系(变量：A、B、C...)

与：$A \cdot 1 = A$ 或：$A + 0 = A$ 非：$A \cdot \overline{A} = 0$

$A \cdot 0 = 0$ $A + 1 = 1$ $A + \overline{A} = 1$

逻辑函数与普通代数中的函数相比较，有两个突出的特点：

（1）逻辑变量和逻辑函数只能取两个值0和1。

（2）函数和变量之间的关系是由“与”、“或”、“非”三种基本运算决定的。

三、与普通代数相似的定理

交换律　$A \cdot B = B \cdot A$　　　$A + B = B + A$

结合律　$(A \cdot B) \cdot C = A \cdot (B \cdot C)$

$(A + B) + C = A + (B + C)$

分配律　$A(B + C) = AB + AC$

$A + BC = (A + B)(A + C)$

普通代数不适用！

[例 1. 1. 1] 证明公式 $A + BC = (A + B)(A + C)$

[解]　方法一：公式法

右式 $= (A + B)(A + C) = A \cdot A + A \cdot C + A \cdot B + B \cdot C$

$= A + AC + AB + BC = A(1 + C + B) + BC$

$= A + BC =$ 左式

证明公式　$A+BC=(A+B)(A+C)$

方法二：真值表法(将变量的各种取值代入等式两边，进行计算并填入表中)

A	B	C	$B\cdot C$	$A+BC$	$A+B$	$A+C$	$(A+B)(A+C)$
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

相等

四、逻辑代数的一些特殊定理

同一律　$A \cdot A = A$　　$A + A = A$

德·摩根定理　$\overline{A \cdot B} = \overline{A} + \overline{B}$　　$\overline{A + B} = \overline{A} \cdot \overline{B}$

还原律　$\overline{\overline{A}} = A$

［例 1. 1. 2］　证明：德 · 摩根定理

A	B	$A \cdot B$	$\overline{A \cdot B}$	$\overline{A}$	$\overline{B}$	$\overline{A} + \overline{B}$	$A + B$	$\overline{A + B}$	$\overline{A} \cdot \overline{B}$
0	0	0	1	1	1	1	0	1	1
0	1	0	1	1	0	1	1	0	0
1	0	0	1	0	1	1	1	0	0
1	1	1	0	0	0	0	1	0	0

相等　　相等

五、关于等式的三个规则

1. 代入规则： 等式中某一变量都代之以一个逻辑函数，则等式仍然成立。

例如，已知 $\overline{A+B}=\overline{A}\cdot\overline{B}$ （用函数 $A+C$ 代替 A）

则 $\overline{(A+C)+B}=\overline{A+C}\cdot\overline{B}=\overline{A}\cdot\overline{C}\cdot\overline{B}$

2. 对偶规则： 如果两个表达式相等，则它们的对偶式也一定相等。

对偶规则的应用：证明等式成立

将 Y 中 "." 换成 "+"，"+" 换成 "."
"0" 换成 "1"，"1" 换成 "0"

$\Rightarrow$ Y' （对偶式）

$$A(B+C)=AB+AC \qquad A+BC=(A+B)(A+C)$$

3. 反演规则： 应用：求逻辑函数的反函数

将Y式中“.”换成“+”,“+”换成“.”
“0”换成“1”,“1”换成“0”
原变量换成反变量，反变量换成原变量 ➡ $\overline{Y}$

例如：已知 $Y_1 = A(B+C)+CD$

$$Y_1' = (A+BC)(C+D)$$

$$\overline{Y_1} = (\overline{A}+\overline{B}\,\overline{C})(\overline{C}+\overline{D})$$

运算顺序：
括号→与→或

已知 $Y_2 = \overline{\overline{A\overline{B}+C}+D}+C$

$$Y_2' = \overline{\overline{(A+\overline{B})\cdot C}\cdot D}\cdot C$$

$$\overline{Y_2} = \overline{\overline{(\overline{A}+B)\cdot\overline{C}}\cdot\overline{D}}\cdot\overline{C}$$

不属于单个变量上的反号应保留不变

例： $L = \overline{A} \bullet \overline{B} + C \bullet D + 0$

注意括号

注意括号

$$L' = (\overline{A} + \overline{B}) \bullet (C + D) \bullet 1$$

$$\overline{L} = (A + B) \bullet (\overline{C} + \overline{D}) \bullet 1$$

在L’的基础上每变量求反。

8.4 用代数法化简逻辑函数

最简与或式：
- 乘积项的项数最少。
- 每个乘积项中变量个数最少。

$$Y=\overline{\overline{AB+\overline{A}C}}$$ ← 最简与或式

核心

最简与非-与非式

$$\overline{\overline{AB}\cdot\overline{\overline{A}C}}$$

最简或与非式

$$\overline{(\overline{A}+\overline{B})(A+\overline{C})}$$

最简与或非式

$$\overline{A\overline{B}+\overline{A}\,\overline{C}}$$

最简或与式

$$\overline{\overline{(\overline{A}+B)\,(A+C)}}$$

8.4.2 用代数法化简逻辑函数

（与或式 $\xrightarrow[\text{定理}]{\text{公式}}$ 最简与或式）

一、并项法： $AB + A\overline{B} = A$

二、吸收法： $A + AB = A$ （长中含短，留下短）

三、消去法： $A + \overline{A}B = A + B$ （长中含反，去掉反）

$$AB + \overline{A}C + BC = AB + \overline{A}C$$ **(正反相对,余全完)**

［例］ $Y = ABC + A\overline{B}\,\overline{C} + AB\overline{C} + A\overline{B}C$

$$= A\,(BC + \overline{B}\,\overline{C}) + A\,(B\overline{C} + \overline{B}C)$$

$$= A \cdot \overline{B \oplus C} + A(B \oplus C)$$

$$= A$$

综合练习：

$$L=\overline{A}\,\overline{B}+BD+\overline{A}BD+\overline{A}B\overline{C}D+A\overline{B}$$

合并项

$$=\overline{B}+BD+\overline{A}BD+\overline{A}B\overline{C}D$$

吸收消去

$$=\overline{B}+BD$$

(长中含短，留下短)

吸收消去

(长中含非，去掉非)

$$\therefore L=\overline{B}+D$$ (最简与或式)

[练习] 求下列函数的反函数（用摩根定理），并化简。

$$Y = A\cdot\overline{B+\overline{C}}+\overline{A}D$$

[解] $\overline{Y} = \overline{A\cdot\overline{B+\overline{C}}+\overline{A}D} = \overline{A\cdot\overline{B+\overline{C}}}\cdot\overline{\overline{A}D} = (\overline{A}+B+\overline{C})(A+\overline{D})$

$= \overline{A}\,\overline{D}+AB+B\overline{D}+A\overline{C}+\overline{C}\,\overline{D} = \overline{A}\,\overline{D}+AB+A\overline{C}$

8.5 用卡诺图化简逻辑函数

卡诺图： 最小项方格图

1. 最小项的概念：

包括所有变量的乘积项，每个变量均以原变量或反变量的形式出现一次。

$Y = F(A, B, C)$ （3 变量共有 8 个最小项）

$\bar{A}\bar{B}\bar{C}$ $\bar{A}\bar{B}C$ $\bar{A}B\bar{C}$ $\bar{A}BC$ $A\bar{B}\bar{C}$ $A\bar{B}C$ $AB\bar{C}$ ABC

$Y = F(A, B, C, D)$ （4 变量共有 16 个最小项）

$\bar{A}\bar{B}\bar{C}\bar{D}$ $\bar{A}\bar{B}\bar{C}D$ $\bar{A}\bar{B}C\bar{D}$ … … $ABC\bar{D}$ $ABCD$

（n 变量共有 2^n 个最小项）

2. 最小项的编号：

把与最小项对应的变量取值当成二进制数，与之相应的十进制数，就是该最小项的编号，用 m_i 表示。

对应规律：原变量 ⇔ 1　反变量 ⇔ 0

$\bar{A}\bar{B}\bar{C}$	$\bar{A}\bar{B}C$	$\bar{A}B\bar{C}$	$\bar{A}BC$	$A\bar{B}\bar{C}$	$A\bar{B}C$	$AB\bar{C}$	ABC
0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
0	1	2	3	4	5	6	7
m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7

3. 最小项是组成逻辑函数的基本单元

任何逻辑函数都是由其变量的若干个最小项构成，都可以表示成为最小项之和的形式。

[例] 写出下列函数的标准与或式：

$$Y = F(A,B,C) = AB + \overline{A}C$$

[解] $Y = AB(\overline{C} + C) + \overline{A}C(\overline{B} + B)$

$$= \underset{m_6}{AB\overline{C}} + \underset{m_7}{ABC} + \underset{m_1}{\overline{A}\,\overline{B}C} + \underset{m_3}{\overline{A}BC}$$

$$= m_6 + m_7 + m_1 + m_3$$

或 $= \sum_m (1,3,6,7)$

8.5.2 逻辑函数的卡诺图表示方法

一、逻辑变量的卡诺图(Karnaugh maps)

1. 二变量 的卡诺图(四个最小项)

$A \backslash B$	$\bar{B}$	B
$\bar{A}$	$\bar{A}\bar{B}$	$\bar{A}B$
A	$A\bar{B}$	AB

➡

$A \backslash B$	0	1
0	m_0	m_1
1	m_2	m_3

2. 变量卡诺图的画法

三变量 的卡诺图：八个最小项（按循环码排列）

逻辑相邻：

两个最小项只有一个变量不同

逻辑相邻的两个最小项可以合并成一项，并消去一个因子。

如：

$$\bar{A}\bar{B}C + \bar{A}BC = \bar{A}C$$

逻辑相邻

A \ BC	00	01	11	10
0	m_0	m_1	m_3	m_2
1	m_4	m_5	m_7	m_6

逻辑相邻

卡诺图的实质：

逻辑相邻 → 几何相邻 $\left\{ \begin{array}{l} \text{紧挨着} \\ \text{行或列的两头} \\ \text{对折起来位置重合} \end{array} \right.$

四变量 的卡诺图：

十六个最小项

AB \ CD	00	01	11	10
00	m_0	m_1	m_3	m_2
01	m_4	m_5	m_7	m_6
11	m_{12}	m_{13}	m_{15}	m_{14}
10	m_8	m_9	m_{11}	m_{10}

几何相邻

几何相邻

五变量 的卡诺图：

三十二个最小项

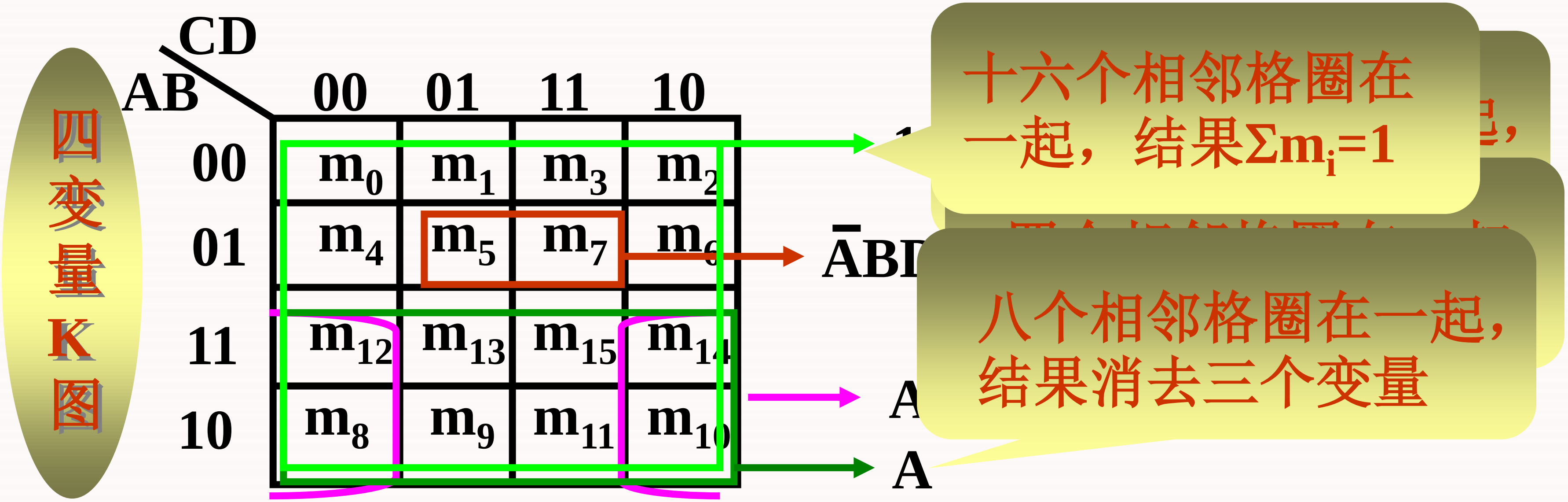
当变量个数超过六个以上时，无法使用图形法进行化简。

AB \ CDE	000	001	011	010	110	111	101	100
00	m_0	m_1	m_3	m_2	m_6	m_7	m_5	m_4
01	m_8	m_9	m_{11}	m_{10}	m_{14}	m_{15}	m_{13}	m_{12}
11	m_{24}	m_{25}	[illegible]	[illegible]	[illegible]	31	m_{29}	m_{28}
10	m_{16}	m_{17}	m_{19}	m_{18}	m_{22}	m_{23}	m_{21}	m_{20}

几何相邻

以此轴为对称轴（对折后位置重合）

3. 卡诺图中最小项合并规律：



● 几何相邻的2^i（i = 1、2、3…n）个小格可合并在一起构成正方形或矩形圈，消去i个变量，而用含（n - i）个变量的积项标注该圈。

三、 用卡诺图化简逻辑函数

基本步骤:

三步曲:
- 一、布阵（画法规则）
- 二、填项(用卡诺图表示逻辑函数)
- 三、勾圈化简(用卡诺图化简)

按L=1的与或式填项

1. 最小项直接填入;
2. 刷项（填公因子所包含的项）;
3. 按Σ（m_0 ,~ m_{15}）编号填入。

例1: $L_1(A,B,C,D)=\bar{A}\bar{B}+BD+\underline{\bar{A}BD}+\bar{A}B\bar{C}D+A\bar{B}$

$(\bar{C}+C)$

直接填入

AB \ CD	00	01	11	10
00				
01		1	1	
11				
10				

公因子:

$\bar{A}BD$

有重复“1”者，只填一个“1”。

例1：$F_1(A,B,C,D)=\overline{A}\,\overline{B}+BD+\overline{A}BD+\overline{A}B\overline{C}D+A\overline{B}$

刷项：

填公因子包含的项

AB \ CD	00	01	11	10
00				
01		1	1	
11		1	1	
10				

公因子：BD

有重复“1”者，只填一个“1”。

例1: $L_1(A,B,C,D) = \overline{A}\,\overline{B} + BD + \overline{A}BD + \overline{A}B\overline{C}D + A\overline{B}$

刷项:

填公因子包含的项

AB \ CD	00	01	11	10
00	1		1	
01	1	1	1 1	
11		1	1	
10	1		1	

1 1

有重复“1”者，只填一个“1”。

例1: $F_1(A,B,C,D)=\overline{A}\,\overline{B}+BD+\overline{A}BD+\overline{A}B\overline{C}D+A\overline{B}$

AB \ CD	00	01	11	10
00	1	1	1	1
01		1	1	
11		1	1	
10	1	1	1	1

F=1的项全部填完以后，填项结束；

不填者自动为“0”。

画包围圈的原则：

1.圈的个数应最少

2.每个“圈”要最大

3.每个“圈”至少要包含一个未被圈过的最小项。

4. 认真检查，防止有重复圈，写出最简与或式。

例1: $L_1(A,B,C,D)=\overline{A}\,\overline{B}+BD+\overline{A}BD+\overline{A}B\overline{C}D+A\overline{B}$

$$\therefore\ L_1(A,B,C,D)=\overline{B}+D$$

AB \ CD	00	01	11	10
00	1	1	1	1
01		1	1	
11		1	1	
10	1	1	1	1

保留公因子：

D

保留公因子：

$\overline{B}$

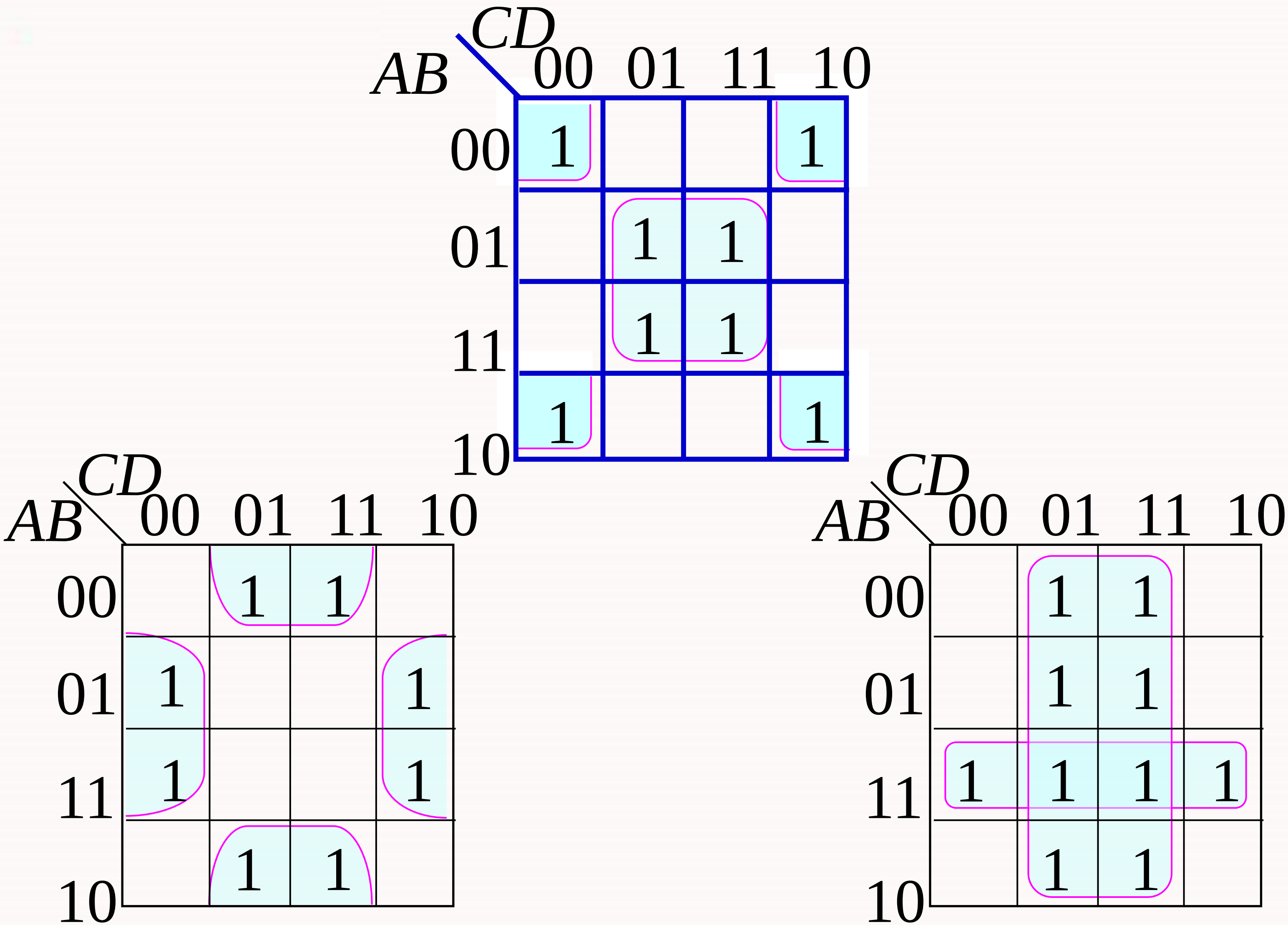
合理重叠（“1”可以重复使用）。

也可以取L=0的项化简： ∴ $L_1(A,B,C,D) = \overline{B\overline{D}}$

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

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

$$\overline{L_1}(A,B,C,D) = B\overline{D}$$



四变量卡诺图的典型合并情况

相邻单元的个数是2^n个，并组成矩形时，可以合并。

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

$$F_2(A,B,C,D) = A\overline{B} + \overline{B}\,\overline{C}D + ABD + \overline{A}B\overline{C}D$$

填项：

AB \ CD	00	01	11	10
00				
01	1			
11	1			
10	1	1	1	1

$$F_2(A,B,C,D) = A\bar{B} + B\bar{C}\bar{D} + ABD + \bar{A}B\bar{C}D$$

AB \ CD	00	01	11	10
00				
01	1	1		
11	1	1	1	
10	1	1	1	1

F=1的项全部填完以后，填项结束。

$$F_2(A,B,C,D) = A\overline{B} + B\overline{C}\,\overline{D} + ABD + \overline{A}B\overline{C}D$$

勾圈化简

AB \ CD	00	01	11	10
00				
01	1	1		
11	1	1	1	
10	1	1	1	1

$B\overline{C}$

AD

$A\overline{B}$

$$\therefore F_2(A,B,C,D) = A\overline{B} + B\overline{C} + AD$$

例：用公式化简法得到下式，问是否最简，若不是请化简之。

$$F_3(A,B,C)=A\overline{B}+\overline{A}C+\overline{A}B+B\overline{C}$$

填项:

A \ BC	00	01	11	10
0		1	1	
1	1	1		

例：用公式化简法得到下式，问是否最简，若不是请化简之。

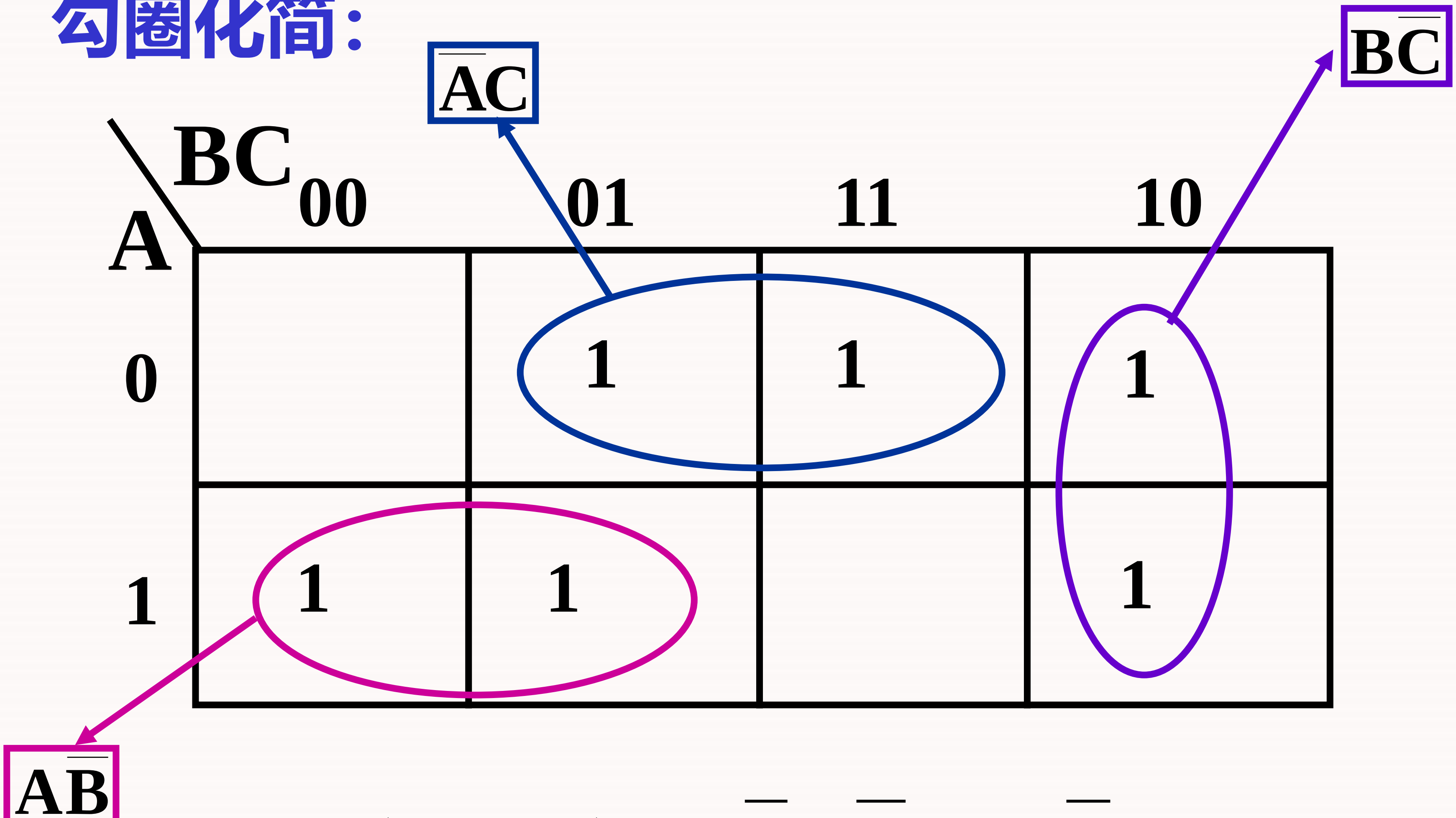
$$F_3(A,B,C)=A\overline{B}+\overline{A}C+\overline{A}B+B\overline{C}$$

A \ BC	00	01	11	10
0		1	1	1
1	1	1		1

F=1的项全部填完以后，填项结束。

$$F_3(A,B,C) = A\overline{B} + \overline{A}C + \overline{A}B + B\overline{C}$$

勾圈化简:



$$\therefore F_3(A,B,C) = A\overline{B} + \overline{A}C + B\overline{C}$$

$$F_3(A,B,C)=A\overline{B}+\overline{A}C+\overline{A}B+B\overline{C}$$

$$[\ \ F_3(A,B,C)=A\overline{B}+\overline{A}C+B\overline{C}\ \ ]$$

$\boxed{B\overline{C}}$ $\boxed{\overline{A}B}$

A \ BC	00	01	11	10
0		1	1	1
1	1	1		1

$\boxed{A\overline{C}}$

$$\therefore\ F_3(A,B,C)=A\overline{C}+B\overline{C}+\overline{A}B$$

A \ BC	00	01	11	10
0		1	1	1
1	1	1		1

A \ BC	00	01	11	10
0		1	1	1
1	1	1		1

说明：化简结果不唯一。

8.5. 4　含有约束项的逻辑函数及其化简

一、　约束的概念和约束条件

(1) 约束：输入变量取值所受的限制

(2) 约束项：不会出现的变量取值所对应的最小项。

在真值表和卡诺图中，用：　× 或 Φ表示约束项；
在逻辑式中，用 Σd 来表示约束项之和。

(don’t care)

十进制数	8421 码
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

二—十进制编码（8421 BCD）

四变量⇒A, B, C, D

六个约束项：

$m_{10}, m_{11}, m_{12}, m_{13}, m_{14}, m_{15}$

二、具有约束的逻辑函数的化简

[解] 化简步骤：

(1) 画函数的卡诺图，顺序为：先填 1 ⟶ 后×

(2) 合并最小项，画圈时 × 既可以当 1，又可以当 0

(3) 写出最简与或表达式

$$\begin{cases} Y = D \\ \sum_d(10,11,12,13,14,15) \end{cases}$$

或$AB + AC = 0$

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

[例] 化简逻辑函数 $F(A,B,C,D)=$

$$\sum_m(1,7,8)+\sum_d(3,5,9,10,12,14,15)$$

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

$$\begin{cases} Y=\overline{A}D+A\overline{D} \\ \sum_d(3,5,9,10,12,14,15)=0 \end{cases}$$

8.6 TTL逻辑门电路

门电路

实现基本逻辑运算和常用复合逻辑运算的电子电路

基本逻辑运算	常用复合逻辑运算
与 门	与 非 门
或 门	或 非 门
非 门	与或非门
	异或门

逻辑门电路的分类

- 逻辑门电路
  - 分立门电路
    - 二极管门电路
    - 三极管门电路
  - 集成门电路
    - MOS门电路
      - NMOS门
      - PMOS门
      - CMOS门
    - TTL门电路

•数字集成电路：在一块半导体基片上制作出一个完整的逻辑电路所需要的全部元件和连线。使用时接：电源、输入和输出。数字集成电路具有体积小、可靠性高、速度快、而且价格便宜的特点。

目前普遍使用的数字集成电路主要有两大类，一类由**NPN**型三极管组成，简称**TTL**集成电路；另一类由**MOSFET**构成，简称**MOS**集成电路。

•TTL电路型：输入和输出端结构都采用了半导体晶体管，称之为: **Transistor— Transistor Logic**。

数字集成电路简介

1.CMOS集成电路:

广泛应用于超大规模、甚大规模集成电路

4000系列 → 74HC 74HCT → 74VHC 74VHCT → 74LVC 74VAUC

4000系列	74HC 74HCT	74VHC 74VHCT	74LVC 74VAUC
速度慢 与TTL不兼容 抗干扰差 功耗低	速度加快 与TTL兼容 负载能力强 抗干扰 功耗低	速度两倍于74HC 与TTL兼容 负载能力强 抗干扰 功耗低	低(超低)电压 速度更加快 与TTL兼容 负载能力强 抗干扰功耗低

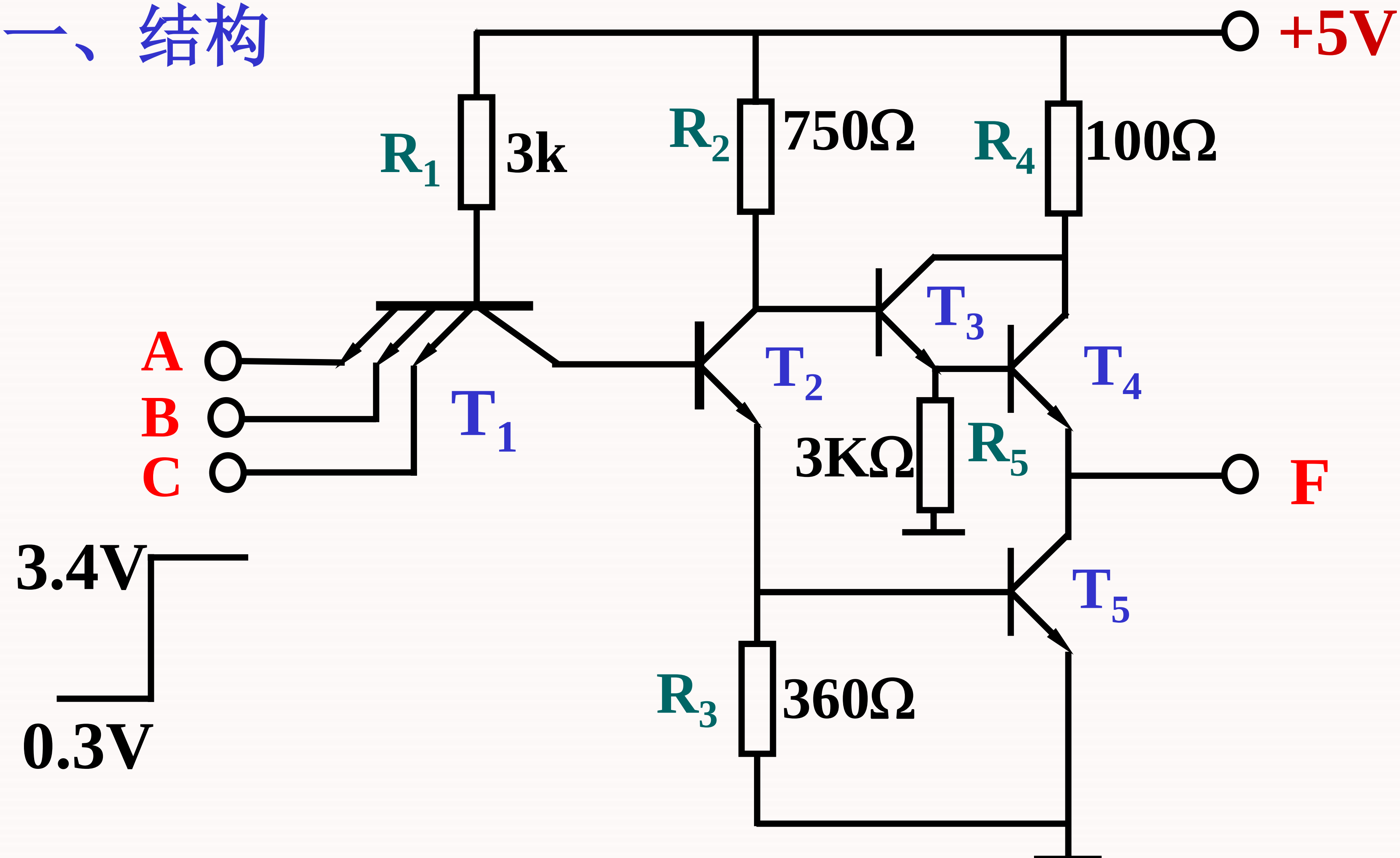
2.TTL 集成电路:

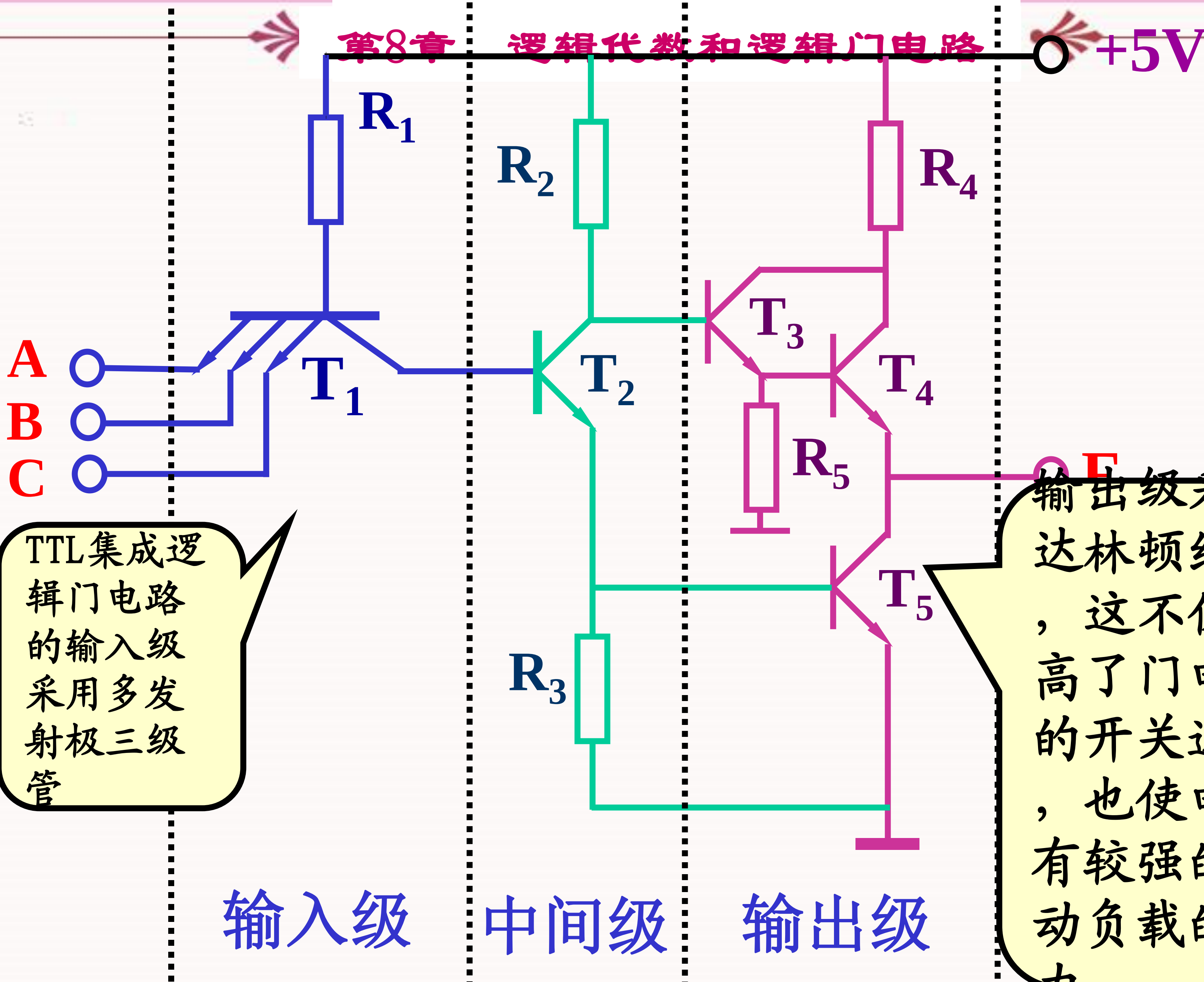
广泛应用于中大规模集成电路

74系列 → 74LS系列 → 74AS系列 → 74ALS

§ 8.6 TTL逻辑门电路
Transistor—Transistor Logic）

一、结构

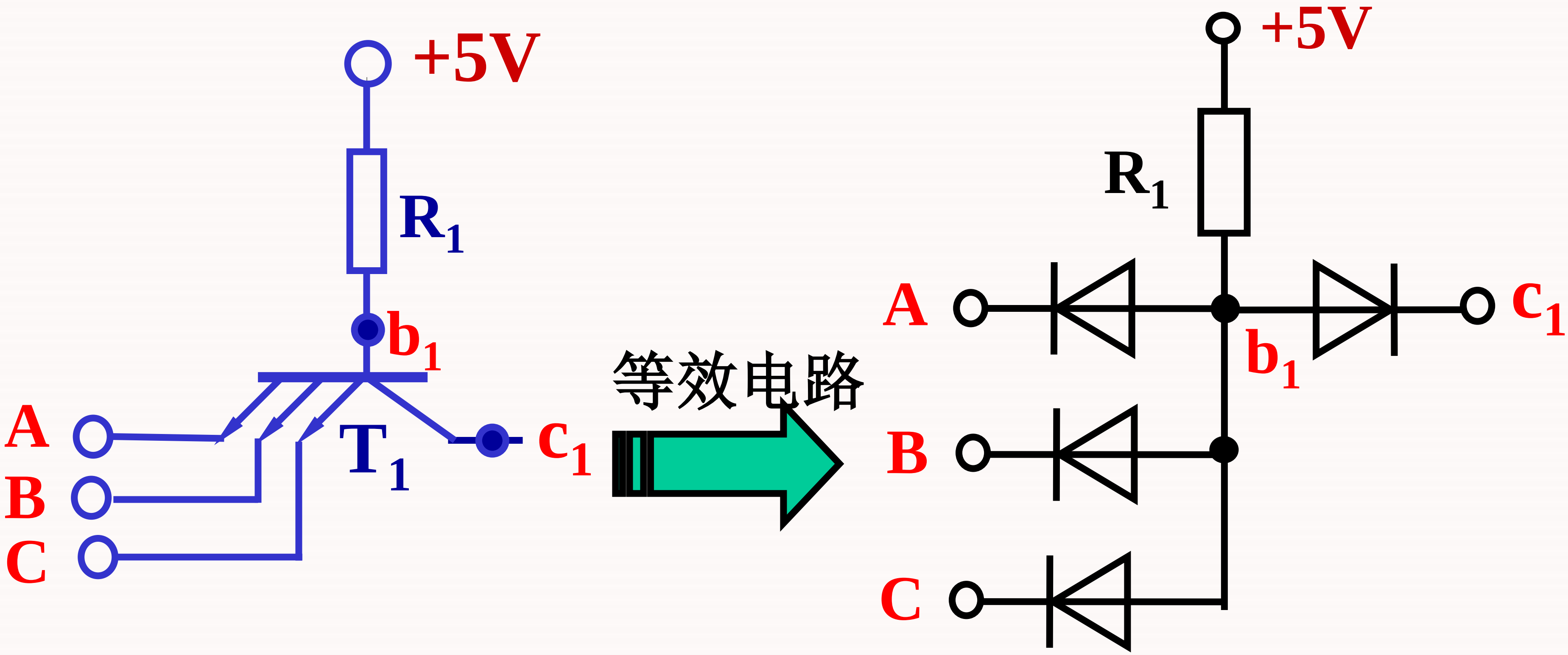




TTL集成逻辑门电路的输入级采用多发射极三级管

输出级采用达林顿结构，这不仅提高了门电路的开关速度，也使电路有较强的驱动负载的能力。

T_1 与R_1组成输入级：



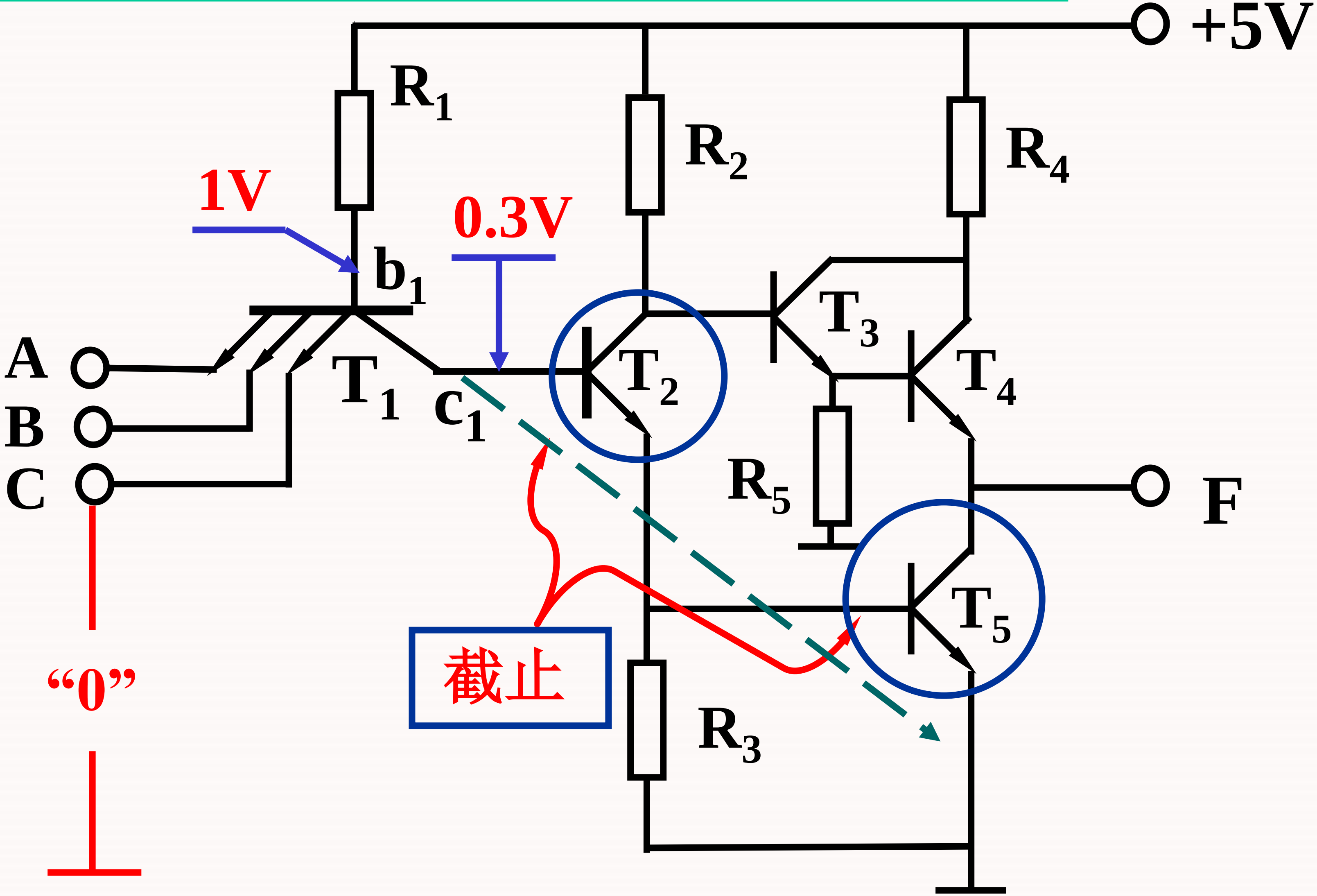
T_1—多发射极晶体管：

实现“与”运算。

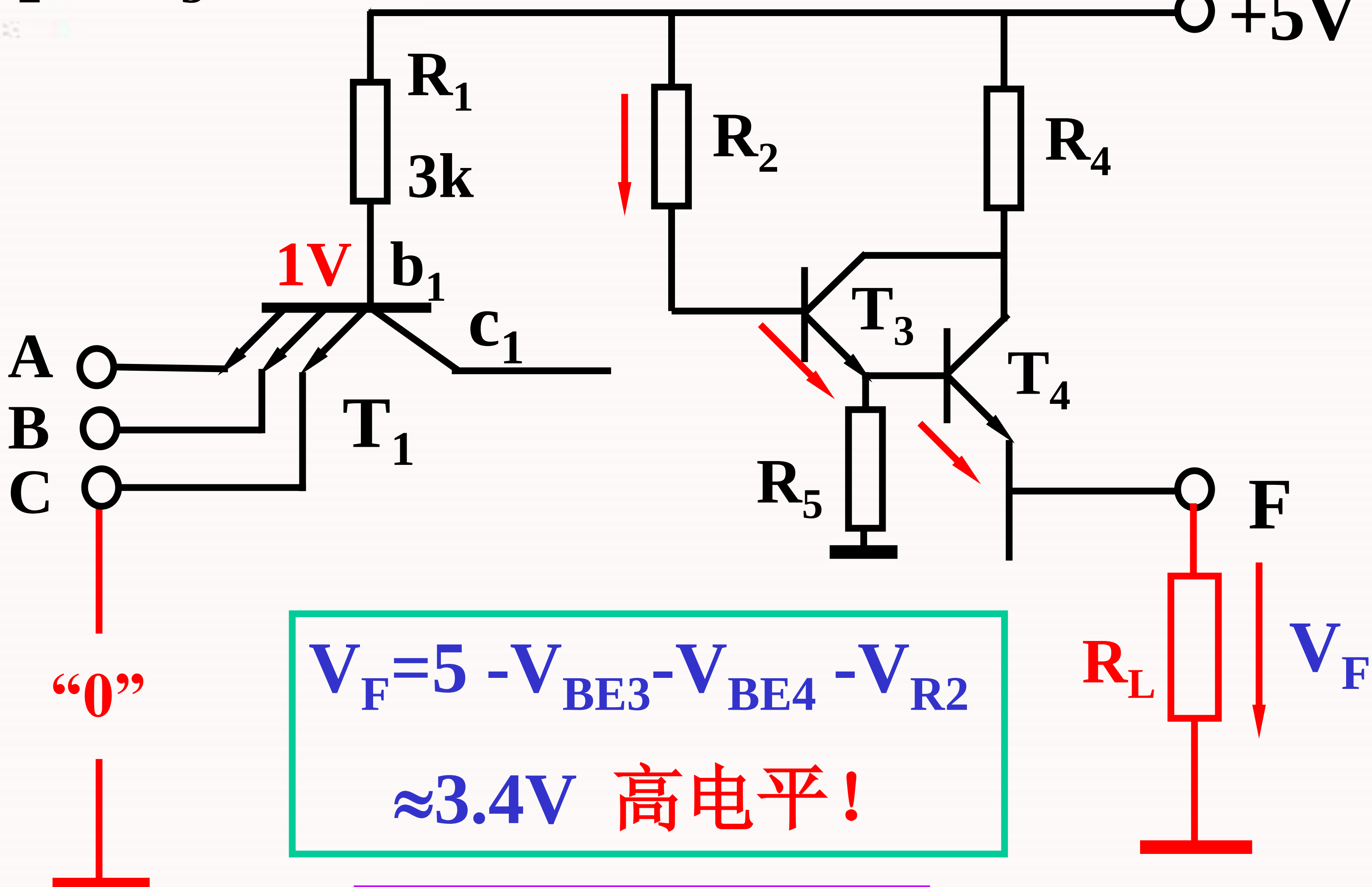
$b_1 = A \cdot B \cdot C$

二、 工作原理

1、任一输入为低电平（0.3V）时



T_2，T_5：截止

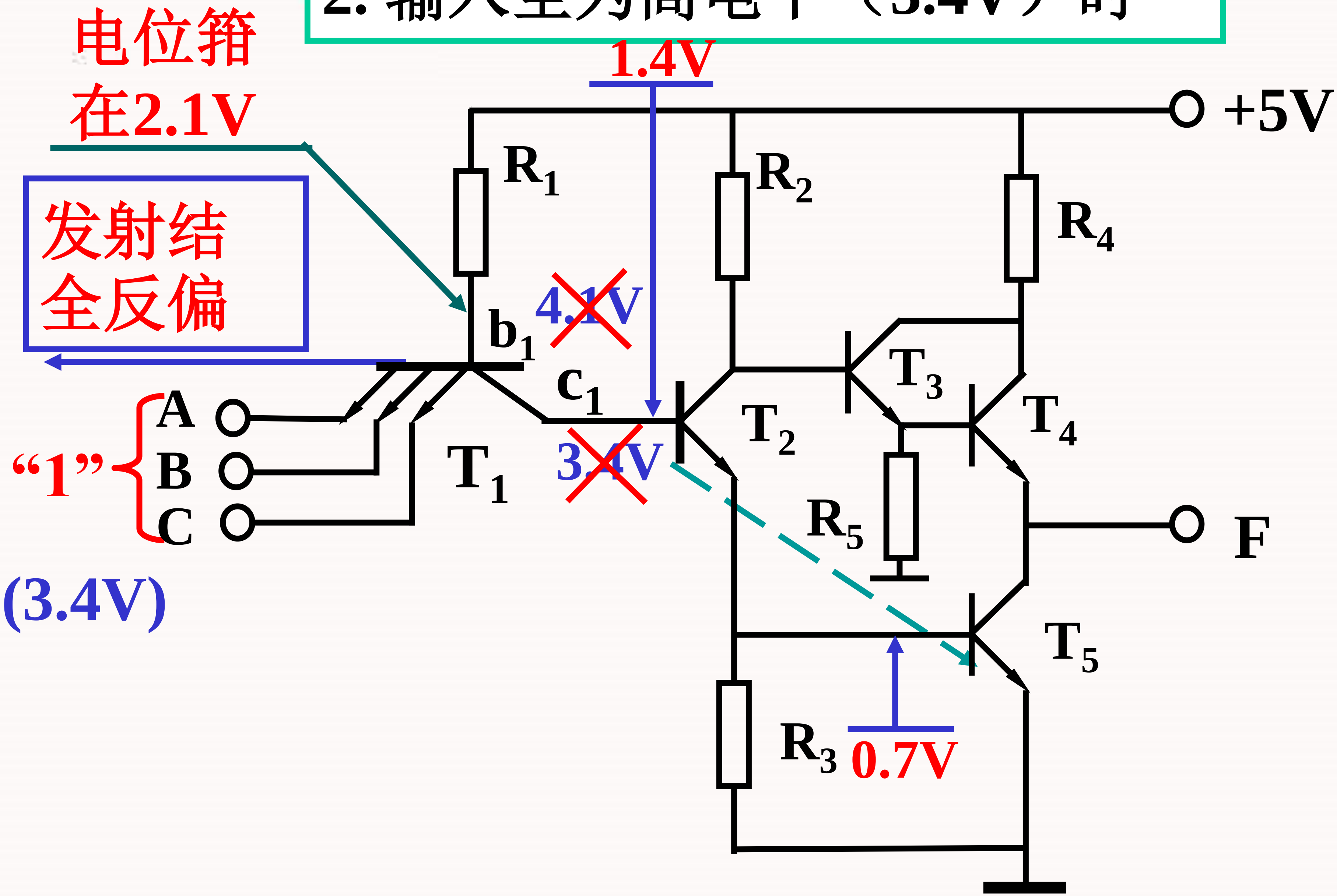


$V_F = 5 - V_{BE3} - V_{BE4} - V_{R2}$

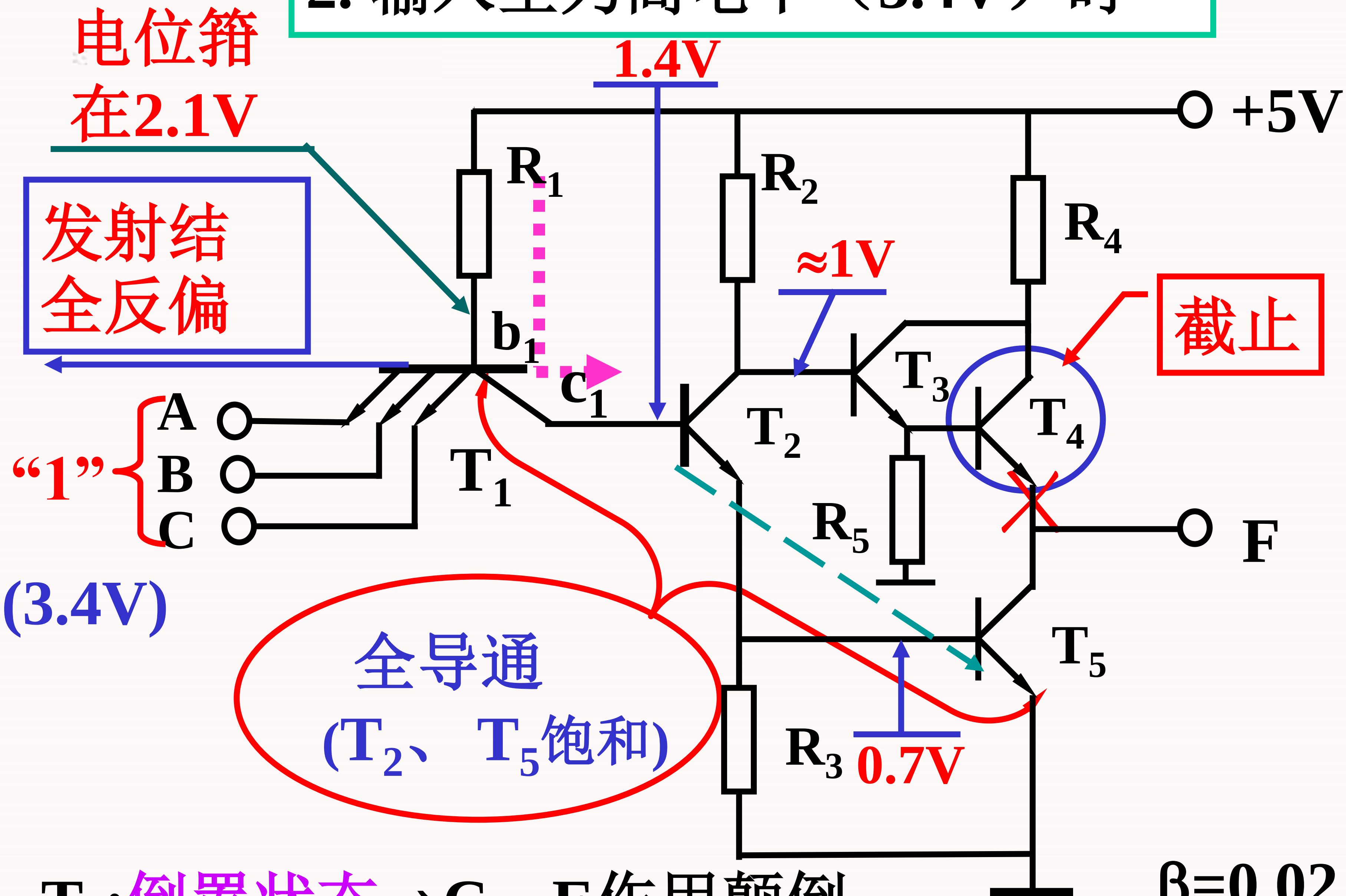
$\approx 3.4V$ 高电平！

逻辑关系:有0出1

2. 输入全为高电平（3.4V）时

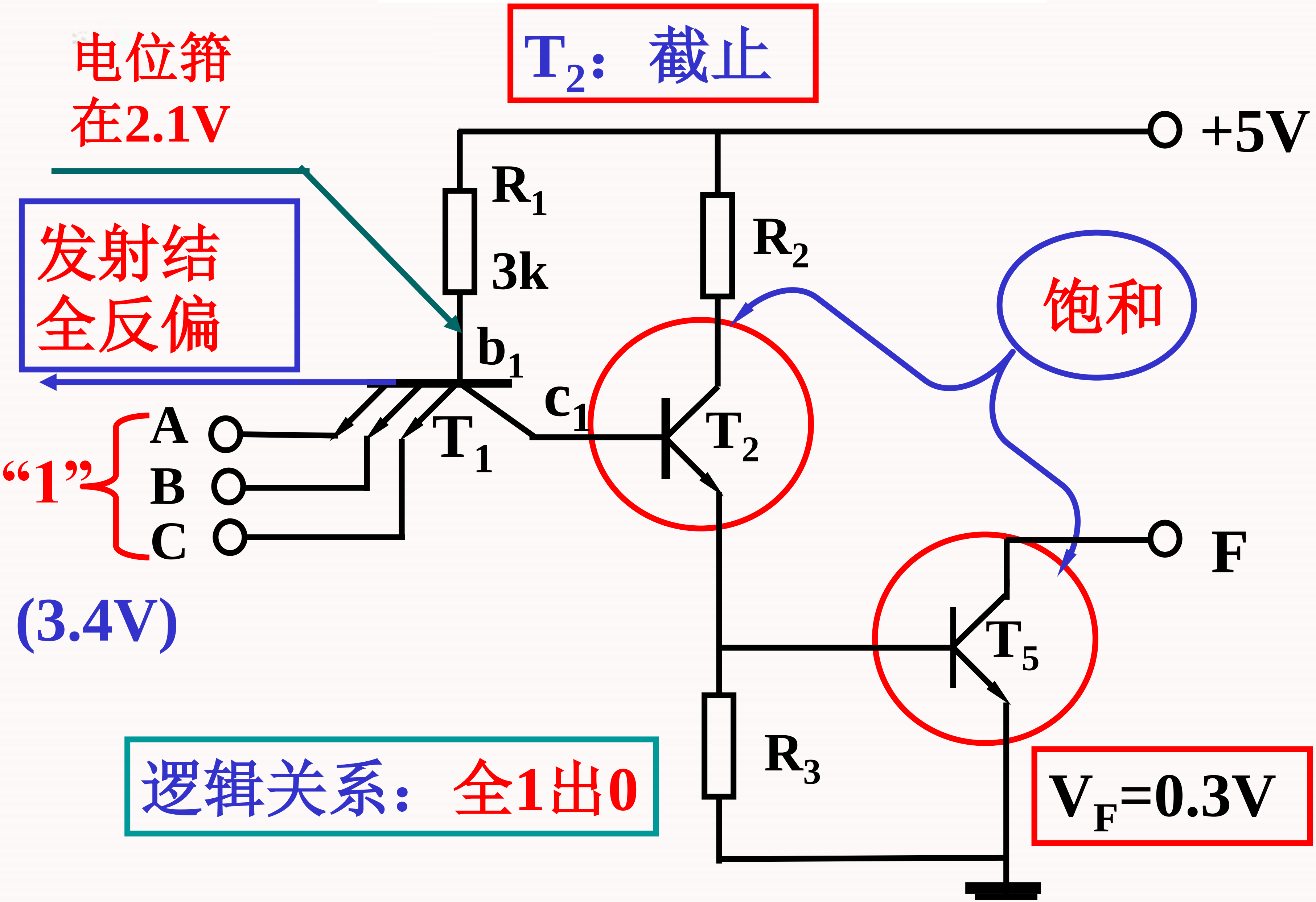


2. 输入全为高电平（3.4V）时



T_1:倒置状态⇒C、E作用颠倒

β=0.02


T2：截止
电位箝
在2.1V
发射结
全反偏
+5V
R1
3k
R2
b1
c1
T1
T2
饱和
A
B
C
"1"
(3.4V)
F
T5
R3
VF=0.3V
逻辑关系：全1出0

TTL与非门

输入有0：T_2、T_5截止,T_3、T_4导通； $V_0 = V_{0H}$。

输入全1： T_4 截止,T_2、T_5饱和导通; $V_0 = V_{0L}$。

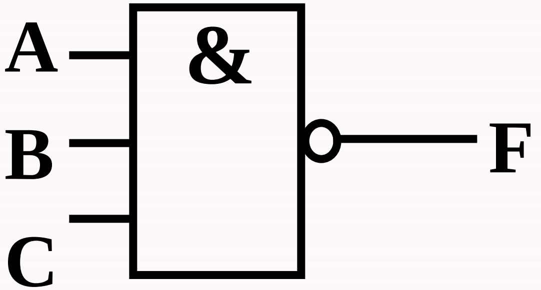
逻辑关系：

$$F = \overline{ABC}$$

有0出1

全1出0

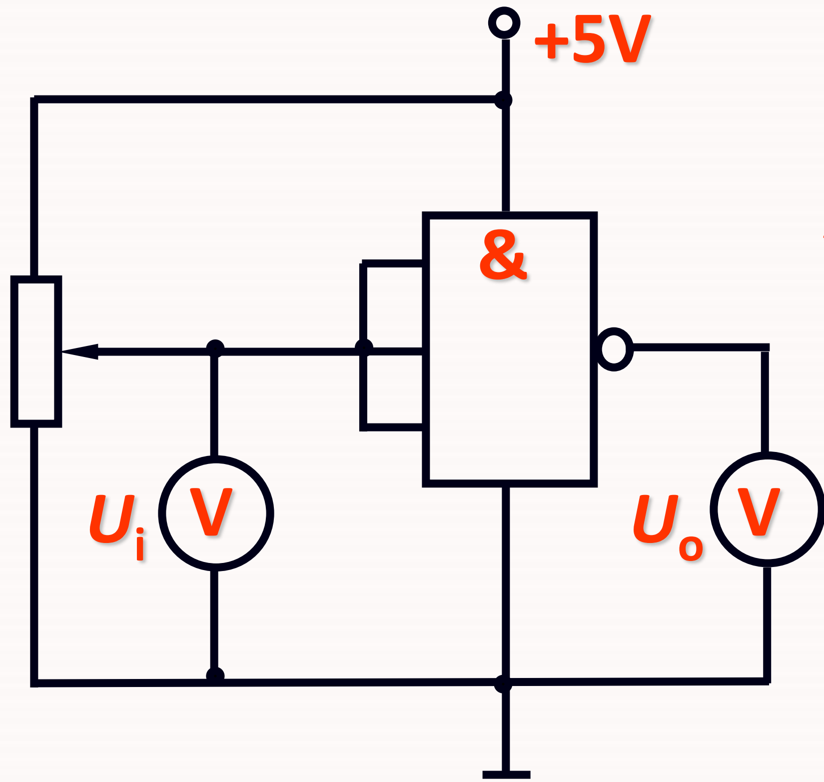
与非门



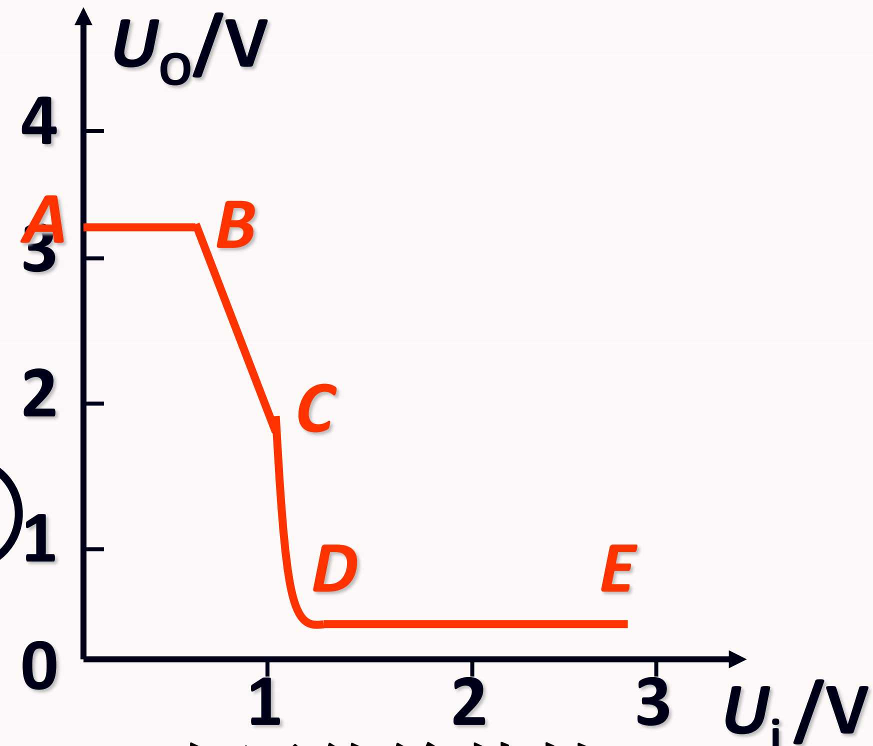
- TTL“与非”门特性及参数

  (1) 电压传输特性：

  输出电压 U_o与输入电压 U_i的关系。



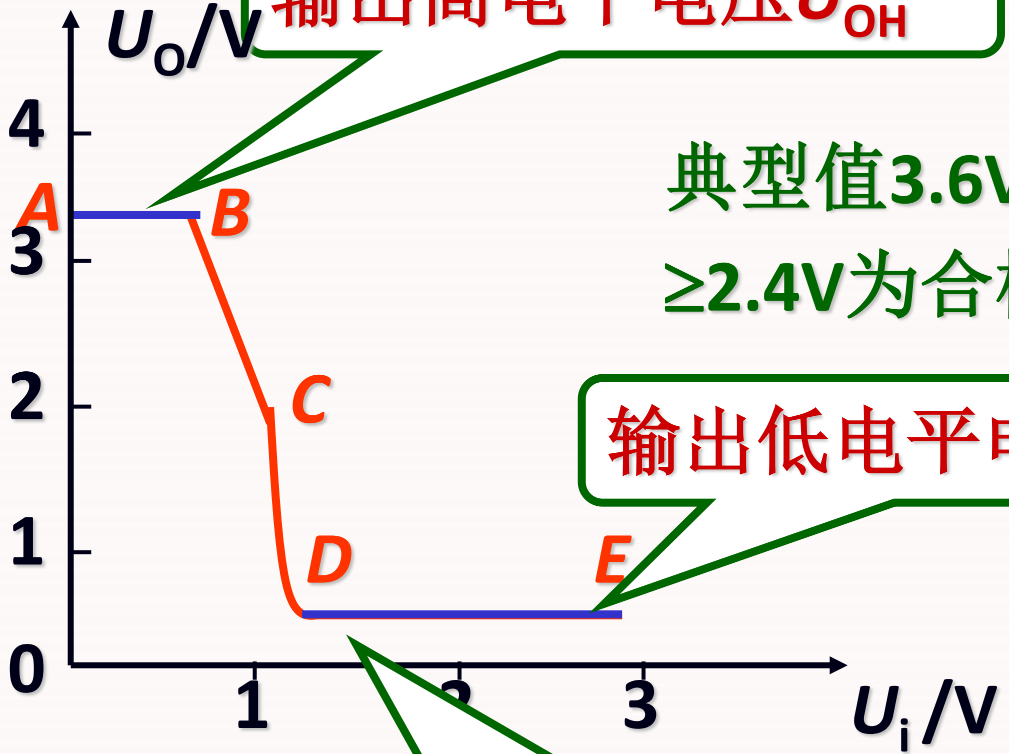
测试电路



电压传输特性

- **(2)TTL“与非”门的参数**

输出高电平电压U_{OH}和输出低电平电压U_{OL}



输出高电平电压U_{OH}

典型值3.6V，

≥2.4V为合格

输出低电平电压U_{OL}

典型值0.3V，

≤0.4V为合格

阈值电压Vth=1.4V

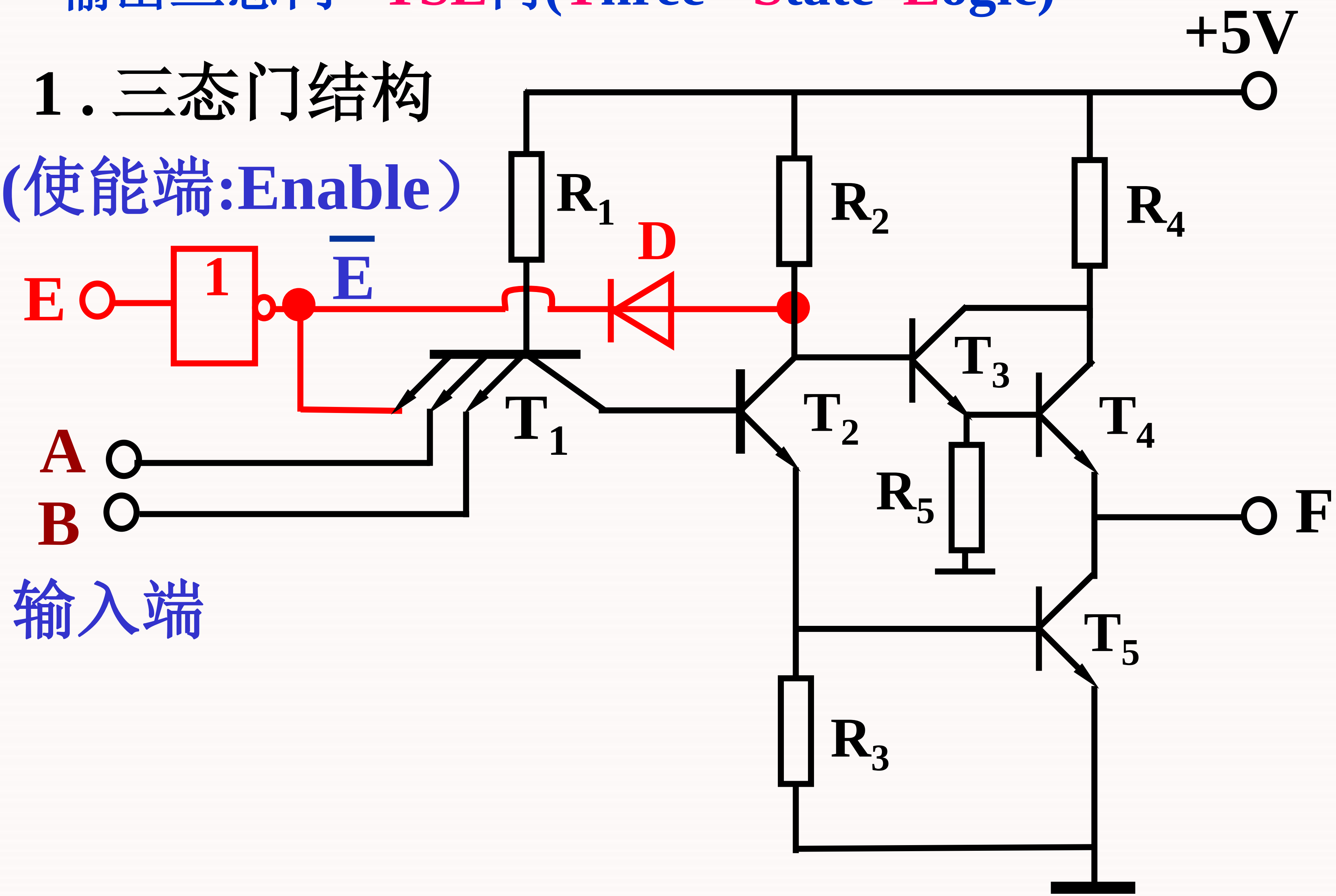
扇出系数N_O

指一个“与非”门能带同类门的最大数目，它表示带负载的能力。对于TTL“与非”门 $N_O \geq 8$。

输出三态门 –TSL门(Three - State Logic)

1. 三态门结构

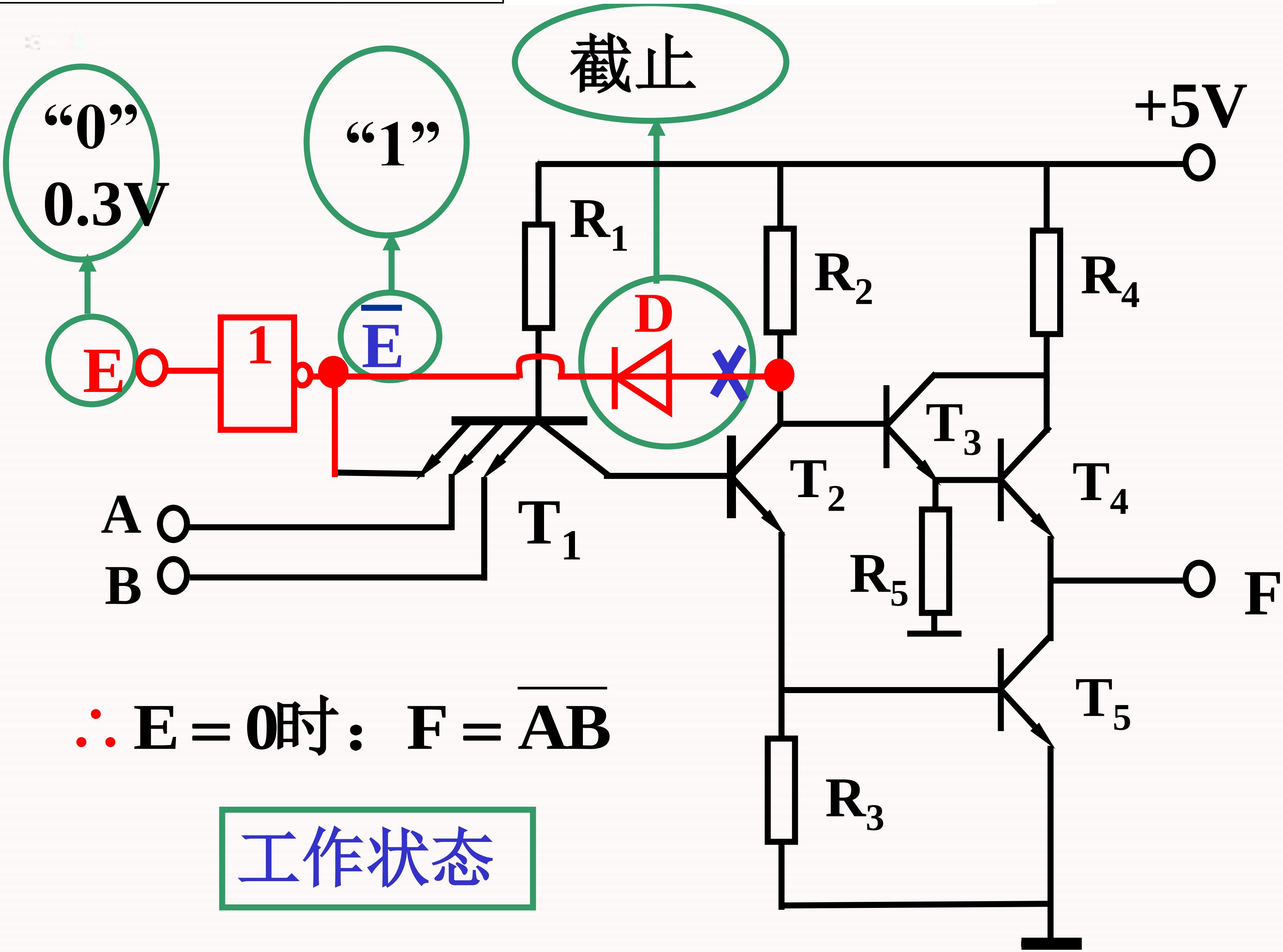
(使能端:Enable)



输入端

2．工作原理

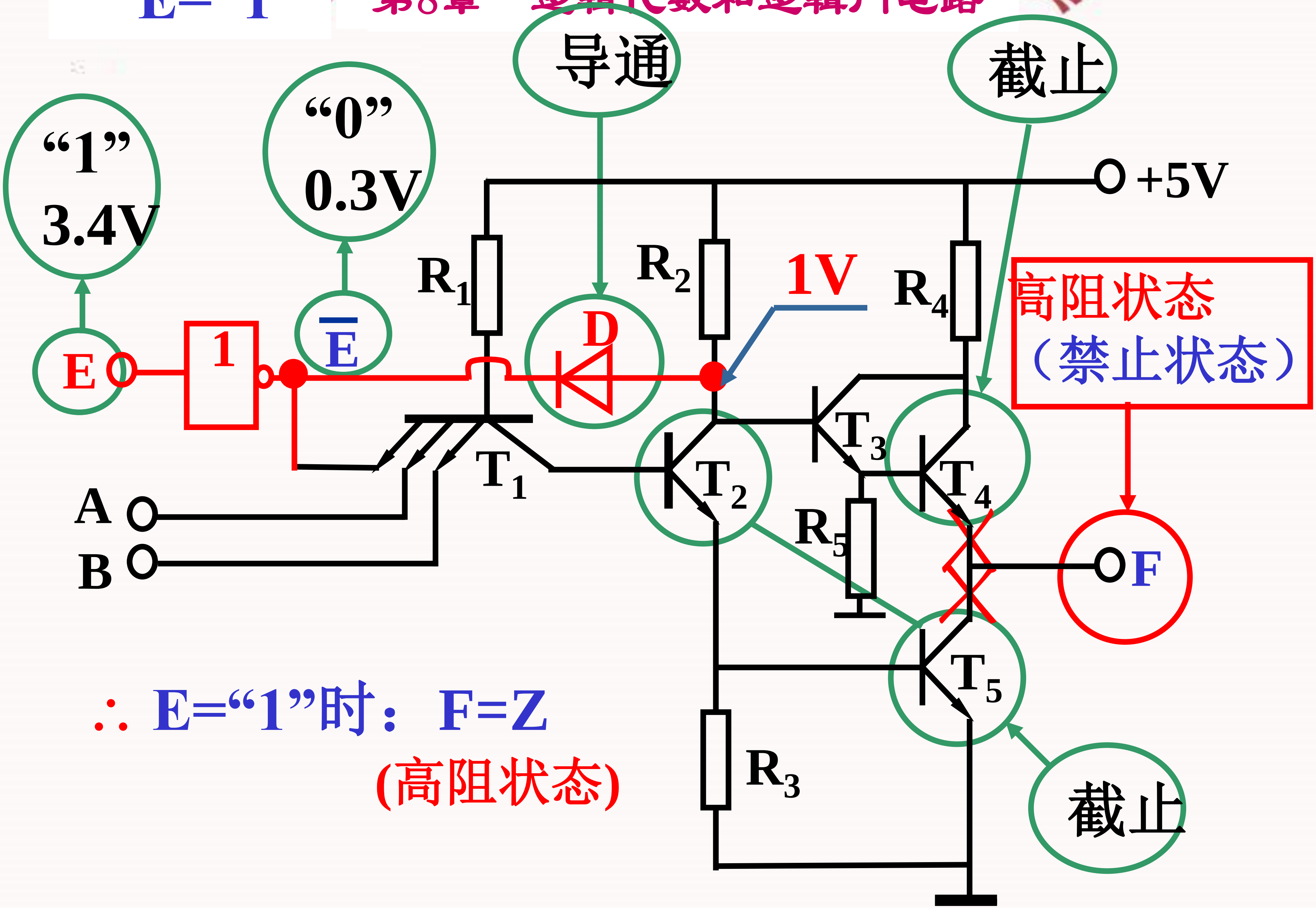
E=“0”



$\therefore E=0$时：$F=\overline{AB}$

工作状态

E=“1”



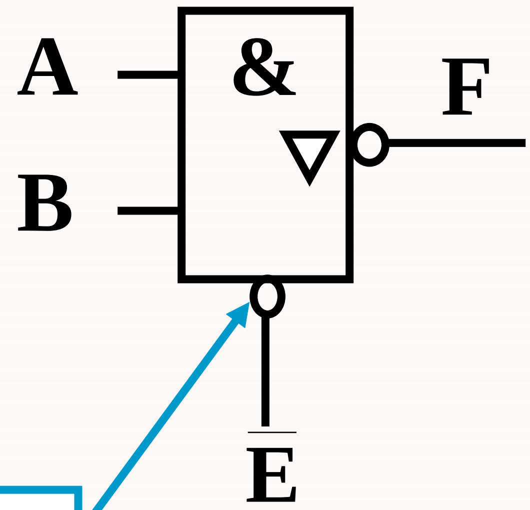
∴ E=“1”时：F=Z

(高阻状态)

功能表：

E	输出
0	$F=\overline{ABC}$ （工作状态）
1	高阻状态 （禁止状态）

符号：



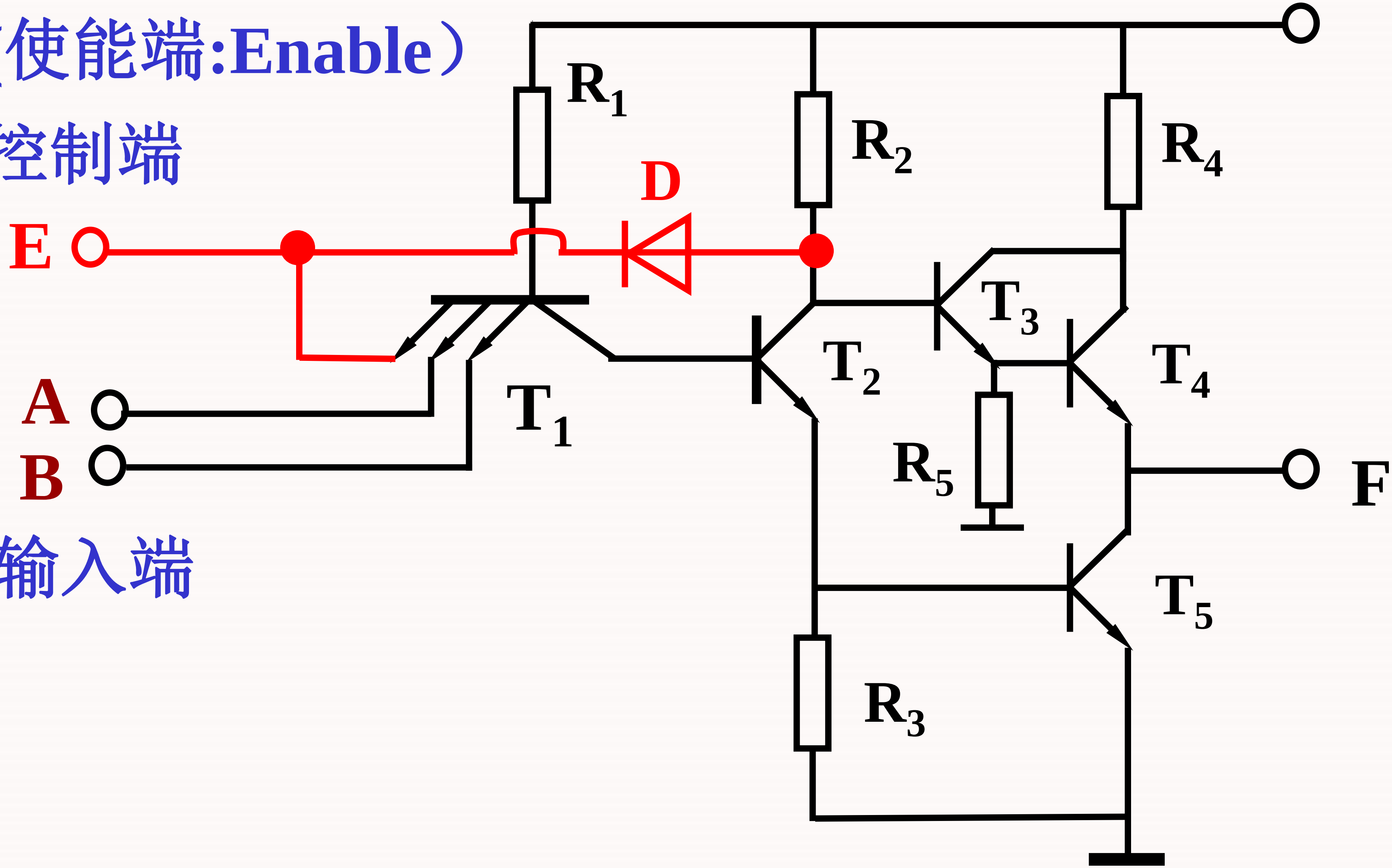
接低电平时为工作状态

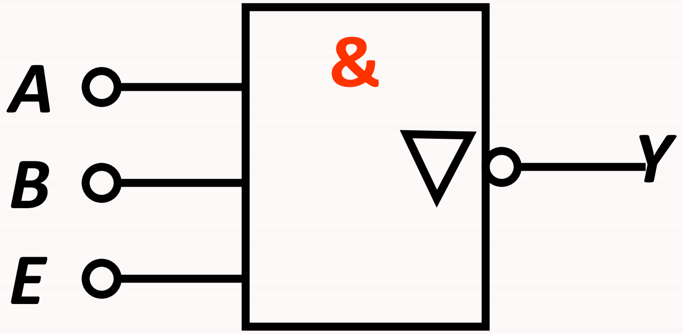
若去掉使能端的非门:

(使能端:Enable)

控制端

输入端





逻辑符号

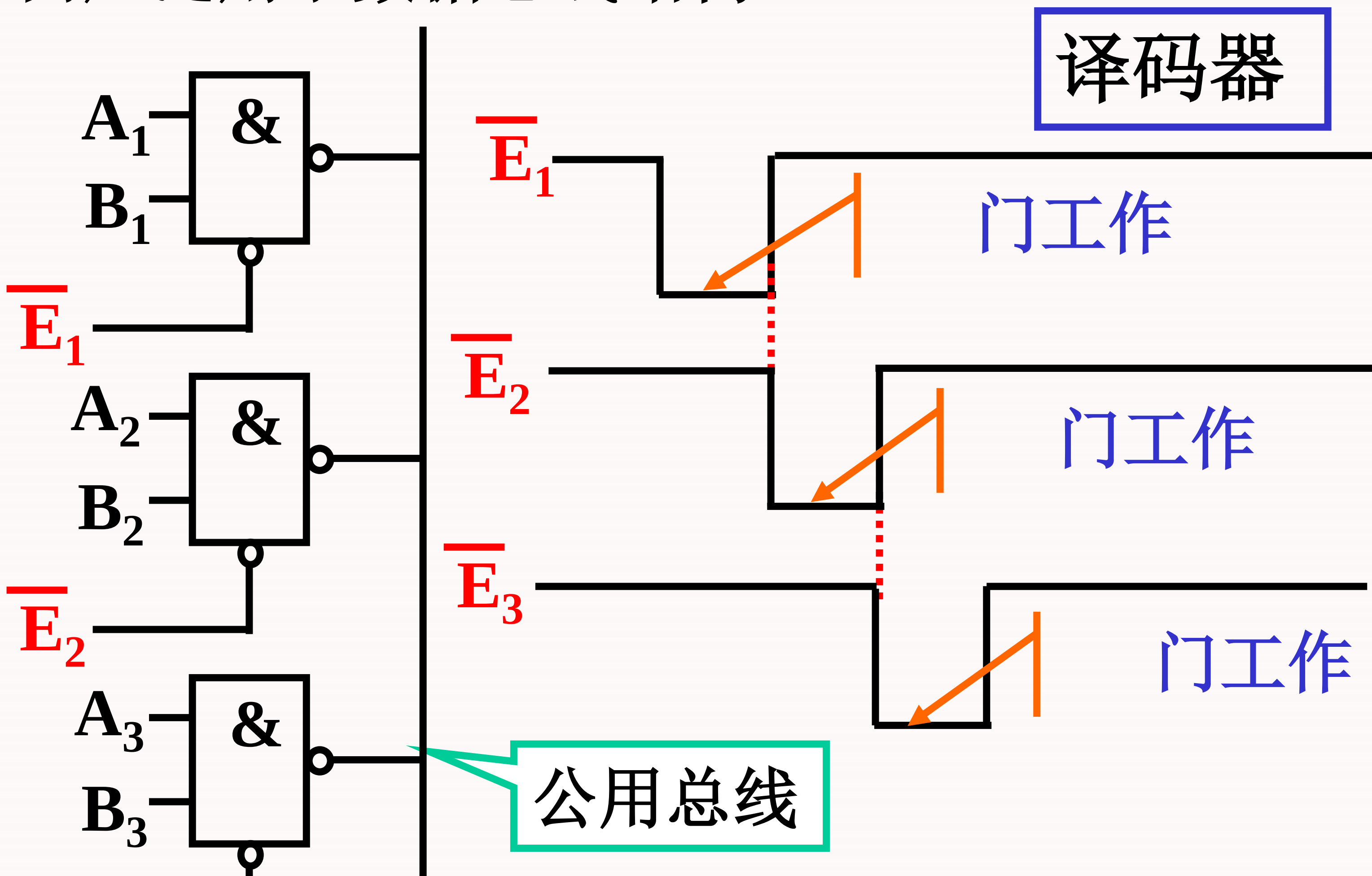
功能表

$E=1$	$Y=\overline{AB}$
$E=0$	输出高阻

3．用途

（1） 三态门广泛用于数据总线结构

任何时刻只能有一个控制端有效，即只有一个门处于数据传输，其它门处于禁止状态

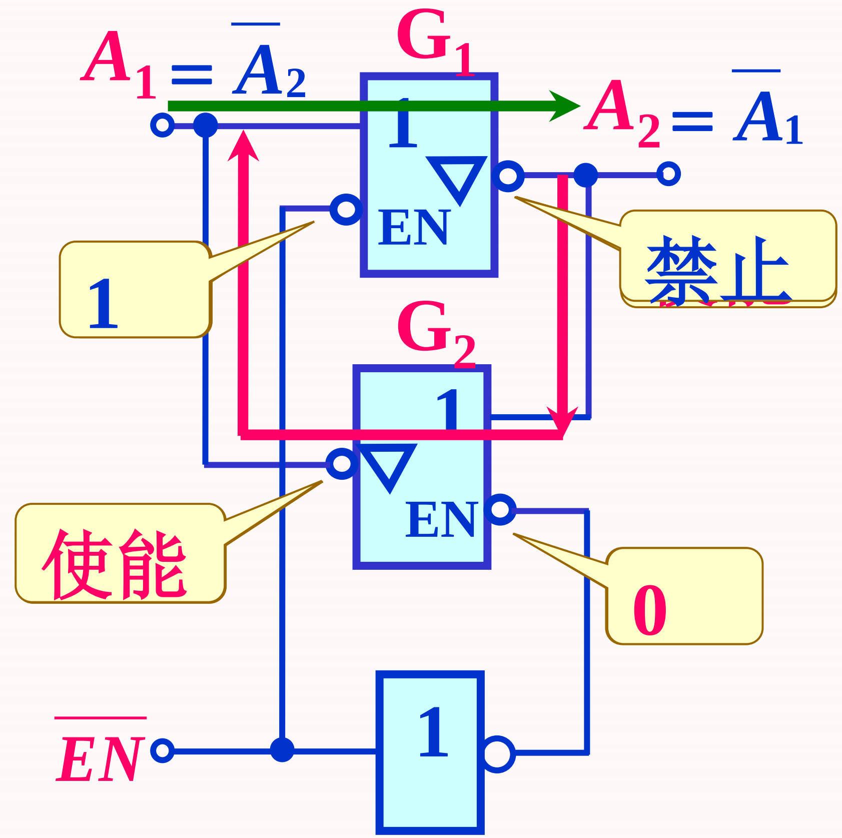


用公用总线**分时**传送不同数据

(2) 用于信号双向传输

$\overline{EN}=0$ 时

$\overline{EN}=1$ 时



计算机PCI总线的地址/数据复用总线，DMA控制器数据总线等纯双向的信号采用双向端口模式

小结:集成门电路使用中应注意的几个问题

分类	TTL	CMOS
工作电源	$V_{CC} = 5$ V	$V_{DD} = 3 \sim 18$ V
输出电平	$U_{OL} = 0.3$ V　$U_{OH} = 3.6$ V	$U_{OL} \approx 0$ V　$U_{OH} \approx V_{DD}$
阈值电压	$U_{TH} = 1.4$ V	$U_{TH} = 0.5\ V_{DD}$
输入端串接电阻R_i	当 $R_i > R_{on}$（2.5 kΩ） 输入由 0 → 1	在一定范围内，R_i的改变不会影响输入电平
输入端悬空	即 $R_i = \infty$ 输入为 “1”	不允许
多余输入端的处理	1. 与门、与非门接电源；或门、或非门接地。 2. 与其它输入端并联。	

第8章　小　结

一、数制和码制

1. 数制： 计数方法或计数体制（由基数和位权组成）

种 类	基 数	位 权	应 用	备 注
十进制	0 ~ 9	10^i	日常	
二进制	0，1	2^i	数字电路	$2 = 2^1$
八进制	0 ~ 7	8^i	计算机程序	$8 = 2^3$
十六进制	0 ~ 9，A ~F	16^i	计算机程序	$16 = 2^4$

各种数制之间的相互转换，特别是十进制→二进制的转换，要求熟练掌握。

2. 码制： 常用的 BCD 码有 8421 码、2421 码、5421 码、余 3 码等，其中以 8421 码使用最广泛。

二、常用逻辑关系及运算

1. 三种基本逻辑运算：与 、或、非

2. 四种复合逻辑运算：与非 、或非、与或非、异或

真值表　　函数式　　逻辑符号

三、逻辑代数的公式和定理

是推演、变换和化简逻辑函数的依据，有些与普通代数相同，有些则完全不同，要认真加以区别。这些定理中，摩根定理最为常用。

四、逻辑函数的化简法

化简的目的是为了获得最简逻辑函数式，从而使逻辑电路简单、成本低、可靠性高。化简的方法主要有公式化简法和图形化简法两种。

1. 公式化简法： 可化简任何复杂的逻辑函数，但要求能熟练和灵活运用逻辑代数的各种公式和定理，并要求具有一定的运算技巧和经验。

2. 图形化简法： 简单、直观，不易出错，有一定的步骤和方法可循。但是，当函数的变量个数多于六个时，就失去了优点，没有实用价值。

约束项：（无关项） 可以取 0，也可以取 1，它的取值对逻辑函数值没有影响，应充分利用这一特点化简逻辑函数，以得到更为满意的化简结果。

五、逻辑函数常用的表示方法：

真值表、卡诺图、函数式、逻辑图和波形图。

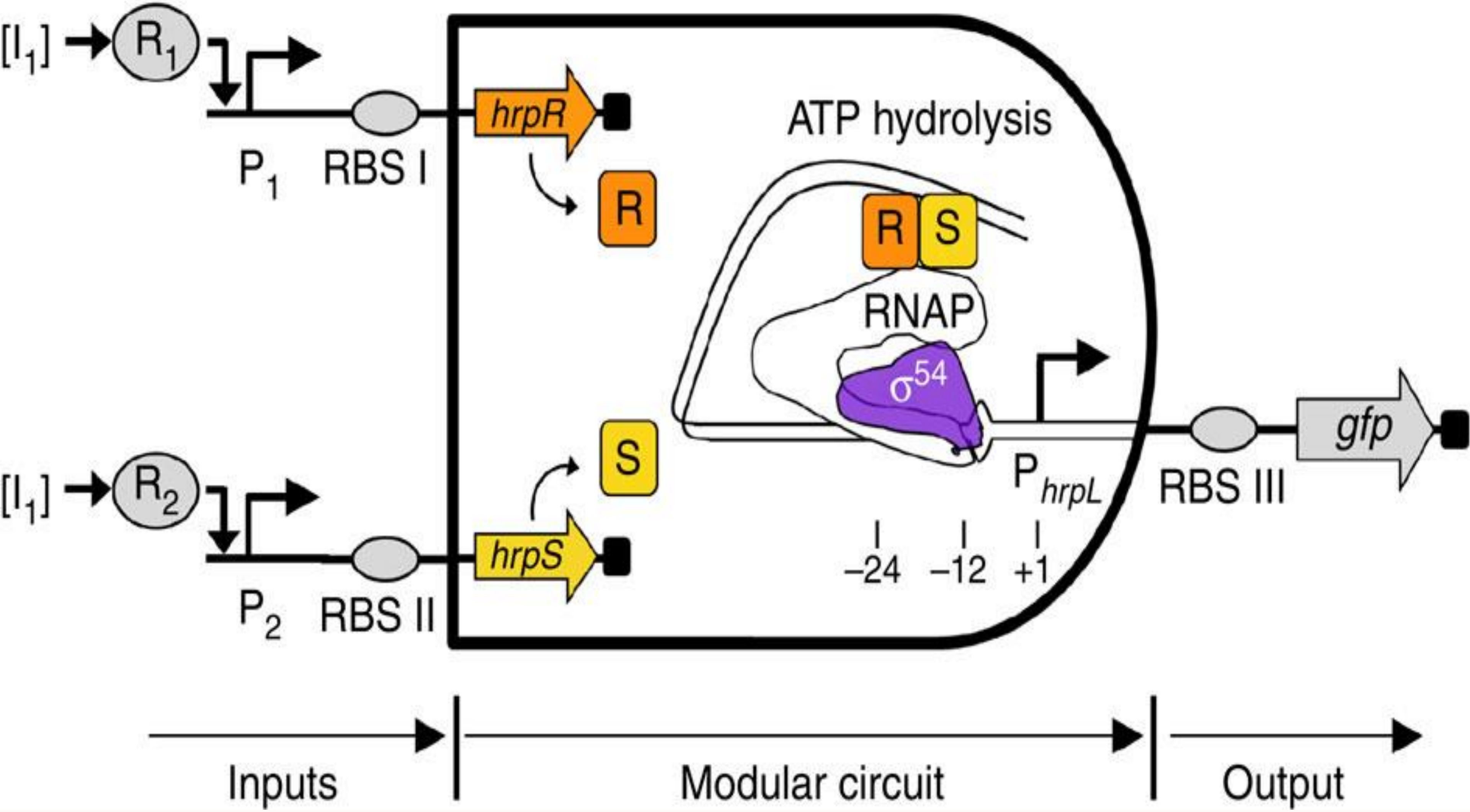
它们各有特点，但本质相同，可以相互转换。尤其是由真值表 → 逻辑图 和 逻辑图 → 真值表， 在逻辑电路的分析和设计中经常用到，必须熟练掌握。

用DNA和细菌组装的可模块化逻辑门

http://www.nature.com/ncomms/journal/v2/n10/full/ncomms1516.html

下载 http://www.nature.com/ncomms/journal/v2/n10/pdf/ncomms1516.pdf

- 英国科学家使用细菌和基因手段，通过对一种无害的大肠杆菌进行基因改造，制造出了用于制造计算设备的基础元件——逻辑门。这种可模块化的新型“生物逻辑门”标志着朝最终制造出生物计算机迈进了一大步。
- 科学家使用已经被修改过的DNA来对肠道内常见的大肠杆菌进行重新编程，让其在受到化学物质的刺激时，能用蛋白质等作为输入信息和输出信息，完成逻辑运算，从而具有与当前计算机所用电路逻辑门类似的信息处理能力。他们研制出了一类“AND（与）门”，也制造出了一类“NOT（非）门”，并将两者结合在一起制造出了更复杂的“与非门”。



[I1]
R1
P1
RBS I
[I1]
R2
P2
RBS II
hrpR
R
S
hrpS
ATP hydrolysis
R
S
RNAP
σ54
PhrpL
–24
–12
+1
RBS III
gfp
Inputs
Modular circuit
Output