

第 4 章 组合逻辑电路

Combinational Logic Circuit

§4.1 组合电路分析

Combinational Logic Circuit Analysis

§4.2 组合逻辑电路设计

Combinational Logic Circuit Design

§4.3 编码器 Encoders

§4.4 译码器 Decoders

§4.5 多路（数据）选择器 MUX

Multiplexers (Data Selectors)

§4.6 比较器 Comparators

§4.7 加法器 Adders

§4.8 组合逻辑电路的竞争冒险

Race-Hazard of Combinational Logic

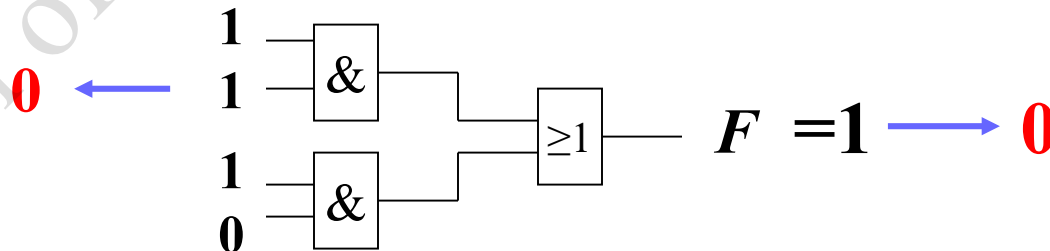
第 4 章 组合逻辑电路

Combinational Logic Circuit

逻辑电路 { 组合逻辑电路
时序逻辑电路

组合逻辑电路特点:

任何时刻输出仅取决于该时刻输入，与过去的输入无关
由门电路构成
无反馈线 (No Memory)



§4.1 组合电路分析

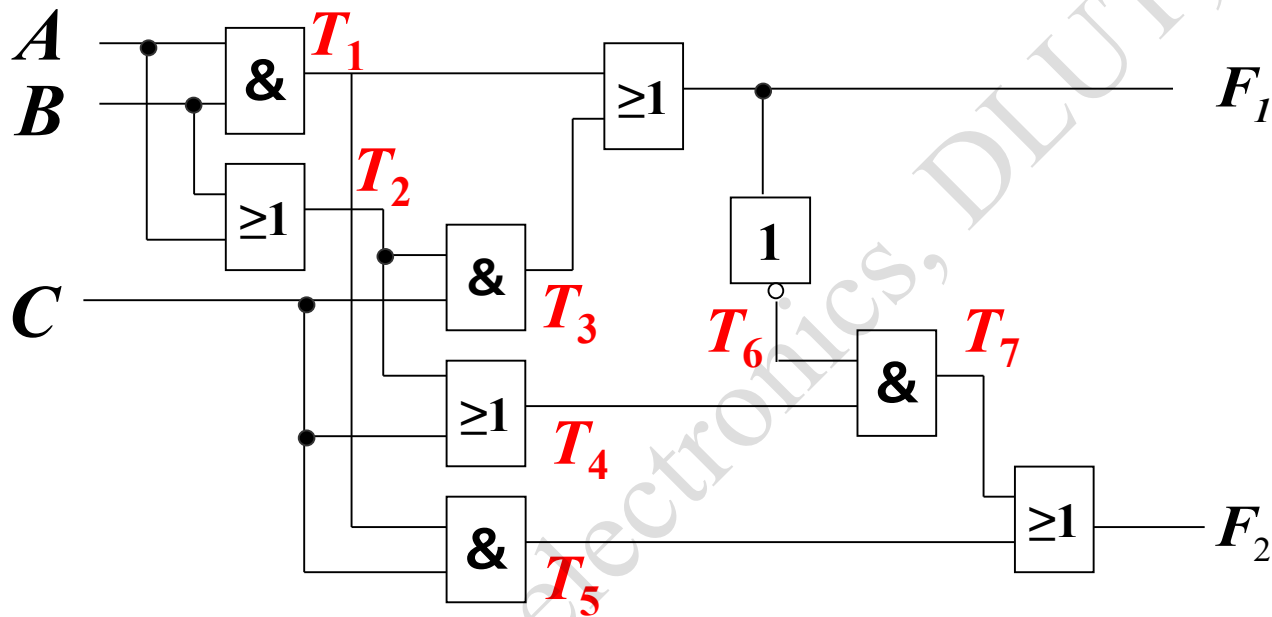
Combinational Logic Circuit Analysis

分析：已知电路，求输出 (F)，分析电路功能

步骤：

- ① 从输入端到输出端，**逐级**写出各逻辑门的输出
- ② 化简逻辑函数
- ③ 列出真值表
- ④ 分析电路功能

例：分析下图电路



解：1. 写出各门输出变量 T_i

2. 化简 T_i

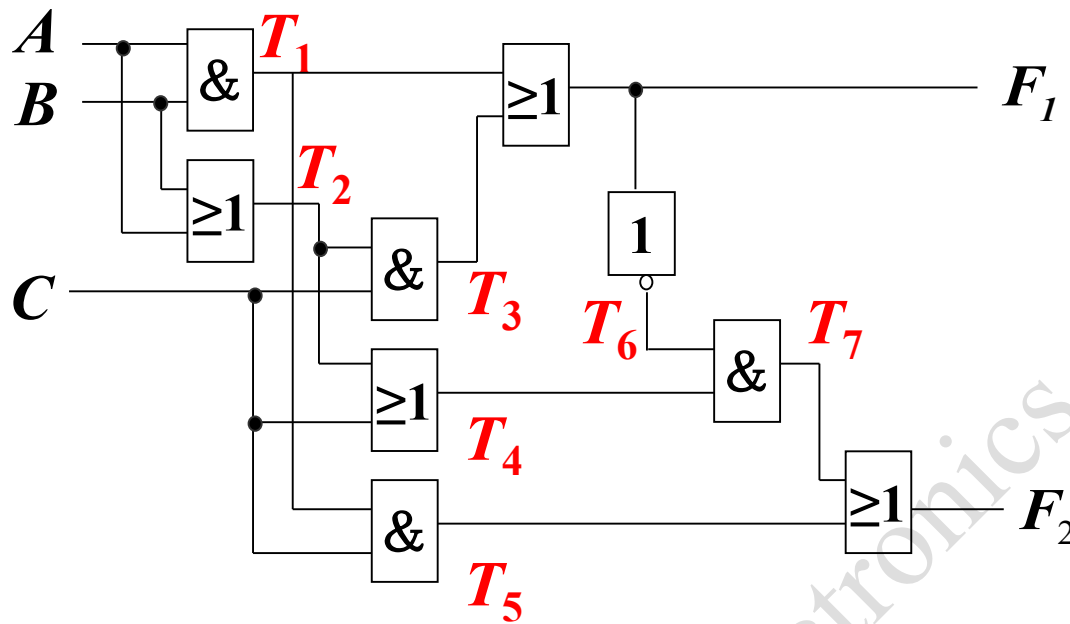
$$T_1 = AB$$

$$T_2 = A+B$$

$$T_3 = (A+B)C$$

$$T_4 = A+B+C$$

$$T_5 = ABC$$



$$T_1 = AB$$

$$T_2 = A+B$$

$$T_3 = (A+B)C$$

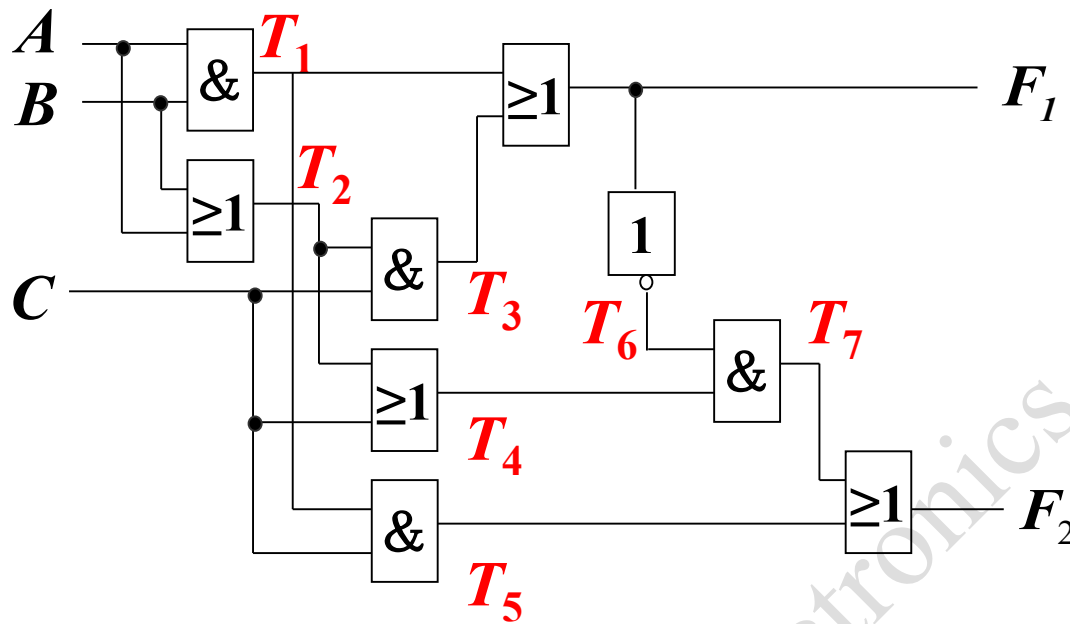
$$T_4 = A+B+C$$

$$T_5 = ABC$$

$$F_1 = T_1 + T_3 = AB + (A+B)C = AB + AC + BC$$

$$T_6 = \overline{F_1}$$

$$\begin{aligned} T_7 &= T_6 \cdot T_4 = \overline{(AB + AC + BC)}(A + B + C) \\ &= \overline{ABC} + \overline{AB\overline{C}} + \overline{A\overline{B}C} \end{aligned}$$



$$T_5 = ABC$$

$$T_7 = \overline{A}\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\overline{C}$$

$$F_1 = T_1 + T_3 = AB + (A + B)C = AB + AC + BC$$

$$F_2 = T_7 + T_5 = \overline{A}\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\overline{C} + ABC$$

3. 列出真值表

$$F_1 = AB + BC + AC$$
$$= \sum(3, 5, 6, 7)$$

$$F_2 = \bar{A} \cdot \bar{B}C + \bar{A}B\bar{C} + A\bar{B} \cdot \bar{C} + ABC$$
$$= \sum(1, 2, 4, 7)$$

4. 分析

$$F_1 = AB + BC + AC$$

$$F_2 = \bar{A} \cdot \bar{B}C + \bar{A}B\bar{C} + A\bar{B} \cdot \bar{C} + ABC$$
$$= \bar{A}(\bar{B}C + B\bar{C}) + A(\bar{B} \cdot \bar{C} + BC)$$
$$= \bar{A}(B \oplus C) + A(\overline{B \oplus C})$$
$$= A \oplus B \oplus C$$

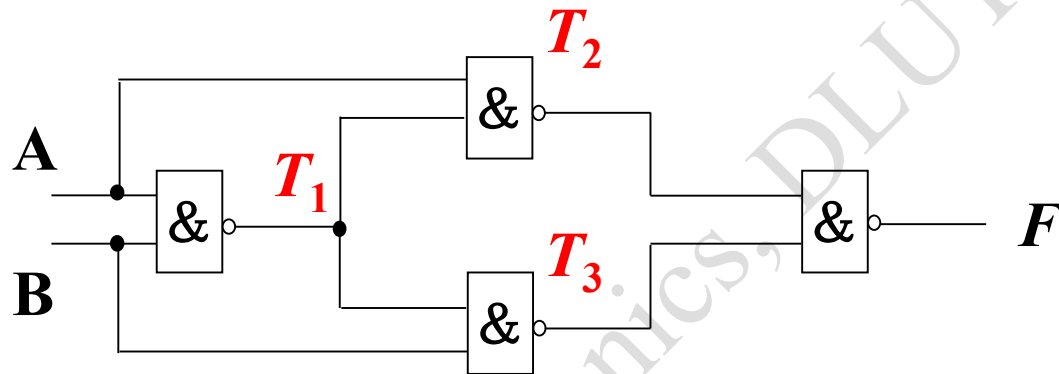
真值表

A	B	C	F ₁	F ₂
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

三变量表决电路

异或

例：分析下列电路功能



$$T_1 = \overline{AB}$$

$$T_2 = \overline{A} \cdot \overline{T_1} = \overline{A} \cdot \overline{\overline{AB}}$$

$$T_3 = \overline{B} \cdot \overline{T_1} = \overline{B} \cdot \overline{\overline{AB}}$$

$$F = \overline{T_2 \cdot T_3} = \overline{\overline{\overline{A} \cdot \overline{\overline{AB}} \cdot \overline{B} \cdot \overline{\overline{AB}}}}$$

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

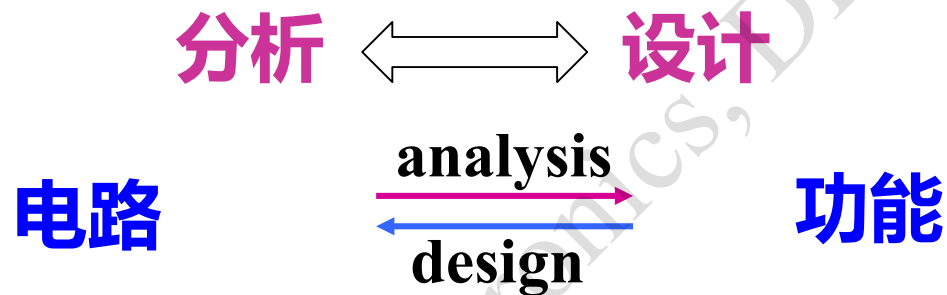
$$= \overline{A(\overline{A} + \overline{B}) + B(\overline{A} + \overline{B})}$$

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

$$= A \oplus B$$

§4.2 组合逻辑电路设计

Combinational Logic Circuit Design



设计的
主要步骤

- 确定输入、输出及它们的关系
- 列出真值表
- 得出函数的最简形式
- 画出电路图

例 1: 设计一个三人表决电路

三人选举组长, 1 和 0 分别表示同意和不同意; 获得2票或以上票数当选 (logic 1), 否则落选 (logic 0)。

三位选民
 A, B, C $\left\{ \begin{array}{l} 1 \text{ 同意} \\ 0 \text{ 不同意} \end{array} \right.$

结果: F $\left\{ \begin{array}{l} 1 \text{ 当选} \\ 0 \text{ 落选} \end{array} \right.$

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

<i>A</i>	<i>B</i>	<i>C</i>	<i>F</i>
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

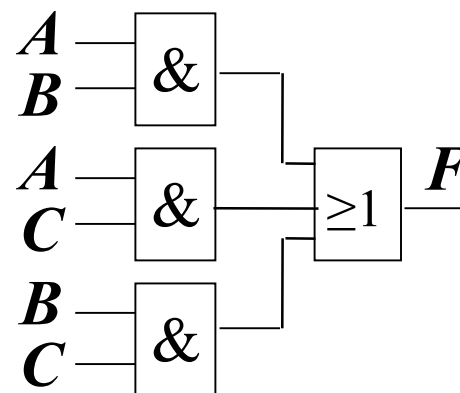
化简函数

		<i>AB</i>			
		00	01	11	10
<i>C</i>	0	0	0	1	0
	1	0	1	1	1

$$F = AB + AC + BC$$

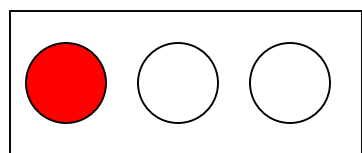
电路

关键: 分析

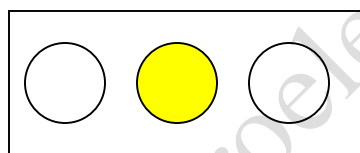


例 2：设计一个交通灯错误状态报警电路：红黄绿三色交通灯，一盏灯亮为正确，其它情况全为错误，需要发出报警信号。

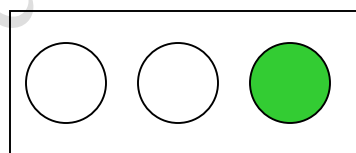
工作状态：有一盏灯亮，并只有一盏灯亮



R Y G

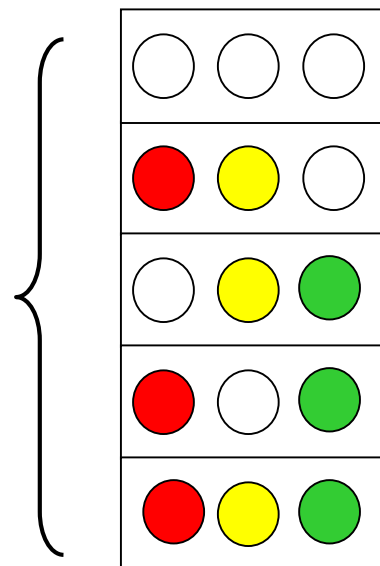


R Y G



R Y G

错误状态：其它



解:

1. 分析

输入 $\left\{ \begin{array}{l} 1 \text{ 亮} \\ 0 \text{ 灭} \end{array} \right.$
 $R \ Y \ G$

输出
错误状态 $\left\{ \begin{array}{l} 1 \text{ 错误} \\ 0 \text{ 没有错误} \end{array} \right.$
 F

2. 真值表:

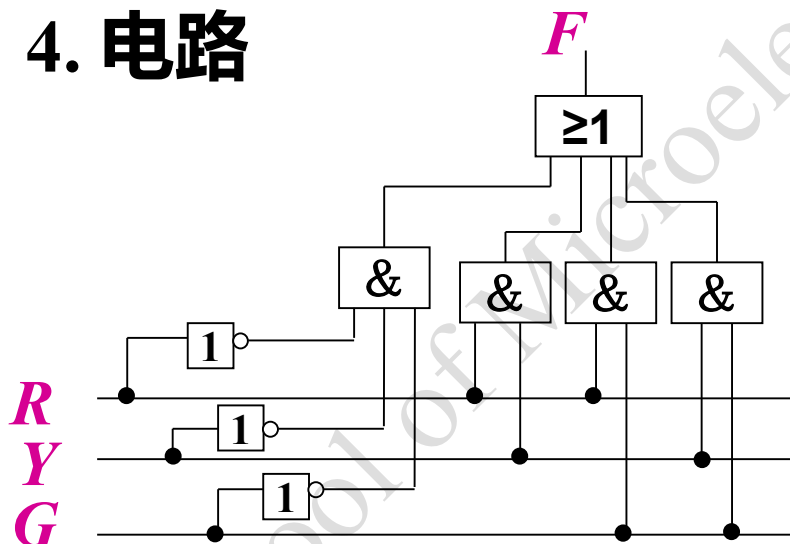
R	Y	G	F
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

R	Y	G	F
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

3. 化简函数

		F			
		RY			
G		00	01	11	10
	0	1	0	1	0
	1	0	1	1	1

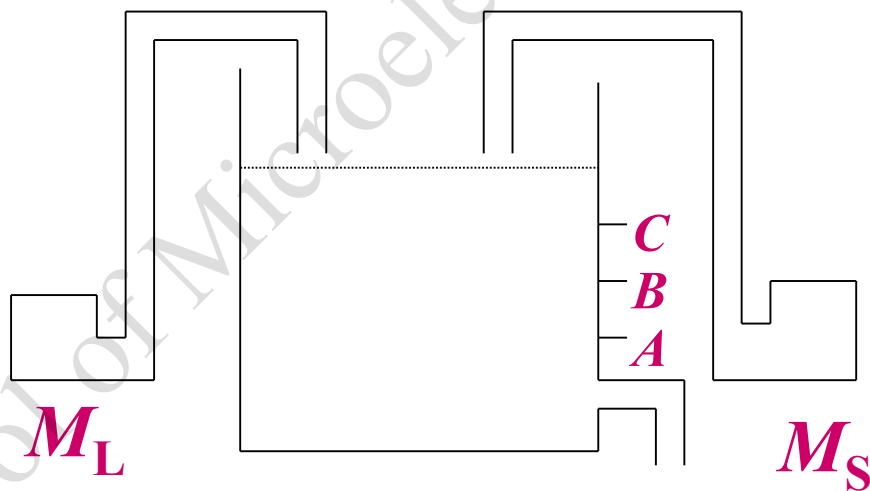
4. 电路



$$F = \overline{R} \cdot \overline{Y} \cdot \overline{G} + RY + RG + YG$$

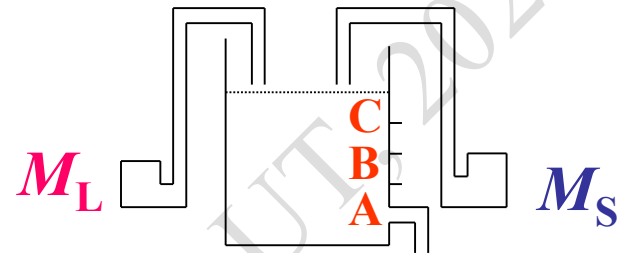
例 3:

一大一小两个水泵 (M_L, M_S) 向水箱泵水; 当水箱中水位低于 C 点时, 小水泵 M_S 单独泵水; 当水位低于 B 点时, 大水泵 M_L 单独泵水; 当水位低于 A 点时, 两个泵同时泵水; 写出两个水泵工作的逻辑函数。



输入
 A, B, C $\begin{cases} =1 & \text{低于相应水位} \\ =0 & \text{不低于} \end{cases}$

输出
 M_S, M_L $\begin{cases} =1 & \text{工作} \\ =0 & \text{不工作} \end{cases}$



A	B	C	$M_L M_S$	
0	0	0	0	0
0	0	1	0	1
0	1	0	Φ	Φ
0	1	1	1	0
1	0	0	Φ	Φ
1	0	1	Φ	Φ
1	1	0	Φ	Φ
1	1	1	1	1

← 低于B 而
不低于C
不可能出现

M_L AB

C	00	01	11	10
0	0	Φ	Φ	Φ
1	0	1	1	Φ

$$M_L = B$$

M_S AB

C	00	01	11	10
0	0	Φ	Φ	Φ
1	1	0	1	Φ

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

例 4:

三位评委裁判举重比赛，一名主裁判，两名副裁判。认为成功举起杠铃时按下按钮 (logic 1)，否则为 logic 0；结果由红、绿灯表示：灯亮和灭分别为逻辑1和 0。红灯和绿灯都亮，表示“**完全举起**”；只有红灯亮表示“**需要研究录像决定**”；其他情况为没有举起。

1. 三位裁判都按键，红、绿灯都亮；
2. 一位主裁判和一位副裁判按键，红、绿灯都亮；
3. 一位主裁判**或**两位副裁判按键，只有红灯亮；
4. 其他情况灯都不亮。

用与非门设计一个满足上述要求的控制电路。

输入

A 主裁
B } 副裁
C }
1 按下按钮
0 不按

输出

R, G
1 亮
0 暗

真值表

A	B	C	R	G
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	0
1	0	0	1	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

化简

R *AB*

<i>C</i>	00	01	11	10
0	0	0	1	1
1	0	1	1	1

$$R = A + BC = \overline{\overline{A} \cdot \overline{BC}}$$

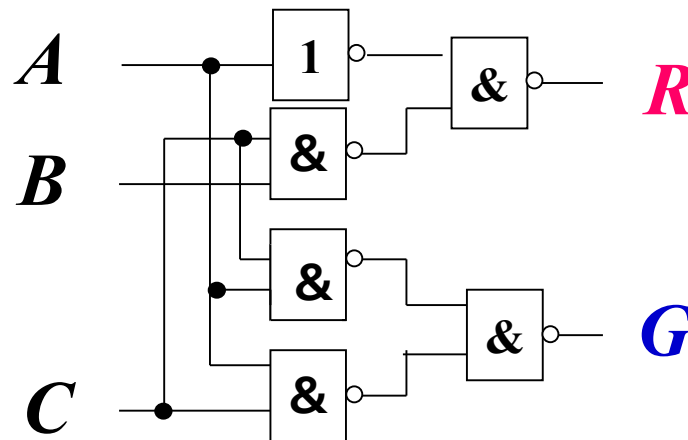
G *AB*

<i>C</i>	00	01	11	10
0	0	0	1	0
1	0	0	1	1

$$G = AB + AC = \overline{\overline{AB} \cdot \overline{AC}}$$

电路

NAND gates



§ 4.3 编码器 Encoders

编码器：将信号或数据编制、转换为可用于通讯、传输和储存的形式的设备

功能：将输入信号转换为二进制代码

编码器分类

- 输出代码种类

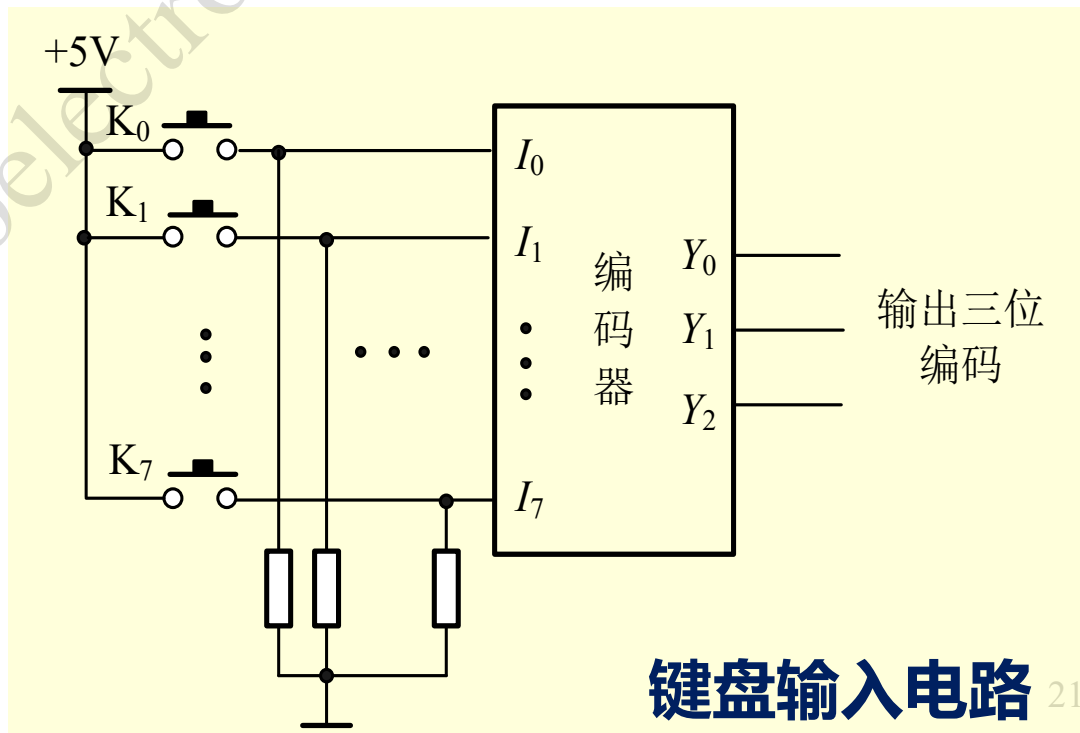
- 二进制编码器

- 二—十进制编码器

- 优先权

- 普通编码器

- 优先编码器



一般而言, N 个不同的信号, 至少需要 n 位二进制数编码。 N 和 n 之间满足下列关系:

$$2^n \geq N$$

例 设计一个4线—2线编码器 (用或门)

解:

(1) 确定输入、输出变量: 由题意知输入为 I_0 、 I_1 、 I_2 、 I_3 四个信息, 输出为 Y_0 、 Y_1 。

(2) 真值表与化简:

I_0	I_1	I_2	I_3	Y_1	Y_0
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1

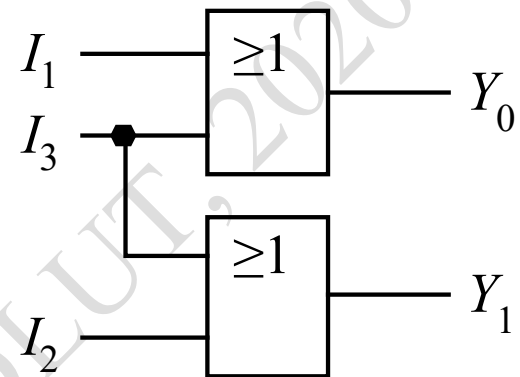
Y_1 $I_2 I_3$ $Y_1 = I_2 + I_3$

$I_0 I_1$	00	01	11	10
00	×	1	×	1
01	0	×	×	×
11	×	×	×	×
10	0	×	×	×

Y_0 $I_2 I_3$ $Y_0 = I_1 + I_3$

$I_0 I_1$	00	01	11	10
00	×	1	×	0
01	1	×	×	×
11	×	×	×	×
10	0	×	×	×

(3) 画编码器电路如图所示



8线 - 3线二进制普通编码器

真值表

输入:

8线, $I_0 \sim I_7$, 八种状态

输出:

三位二进制数

F_2 、 F_1 、 F_0

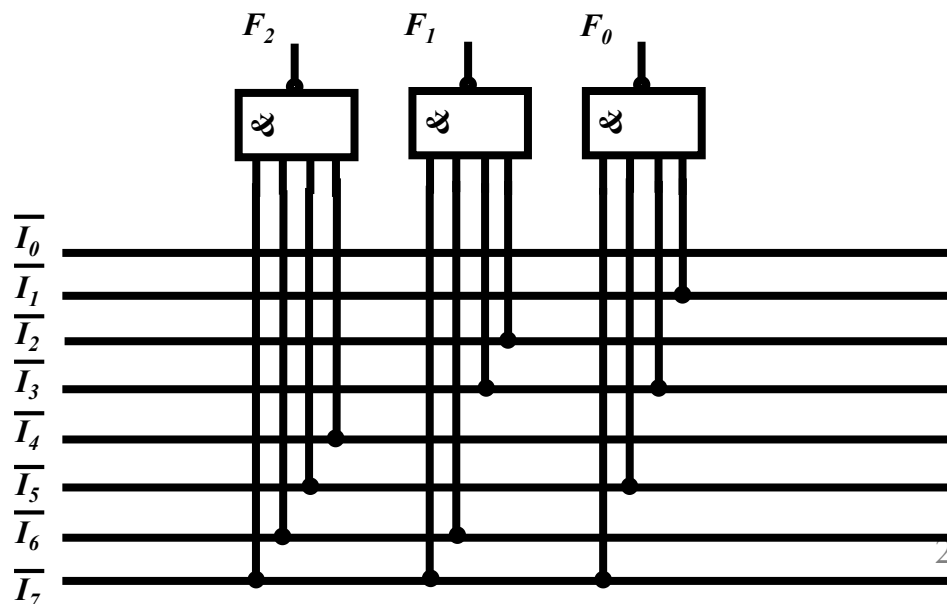
I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7	F_2	F_1	F_0
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

输出表达式

$$F_2 = I_4 + I_5 + I_6 + I_7 = \overline{I_4} \cdot \overline{I_5} \cdot \overline{I_6} \cdot \overline{I_7}$$

$$F_1 = I_2 + I_3 + I_6 + I_7 = \overline{I_2} \cdot \overline{I_3} \cdot \overline{I_6} \cdot \overline{I_7}$$

$$F_0 = I_1 + I_3 + I_5 + I_7 = \overline{I_1} \cdot \overline{I_3} \cdot \overline{I_5} \cdot \overline{I_7}$$



§ 4.4 译码器 Decoders

将输入的**二进制代码转换成对应的输出信号**（检测输入端的**二进制代码**）**或另一种形式的代码**，译码器通常是一个多输入多输出的组合逻辑电路。

n -bit 二进制输入 $\xrightarrow{\text{转换}}$ $\leq 2^n$ 输出

译码器

二进制译码器

码制变换译码器

显示译码器

§ 4.4.1 二进制译码器 Binary Decoders

将二进制代码“翻译”成——对应的输出高、低电平信号。

Inputs: n 位二进制代码

Outputs: 2^n 个输入的各种组合

用 n 个二进制输入端控制 2^n 个输出端

1. 2线 - 4线译码器 2-to-4 Decoder

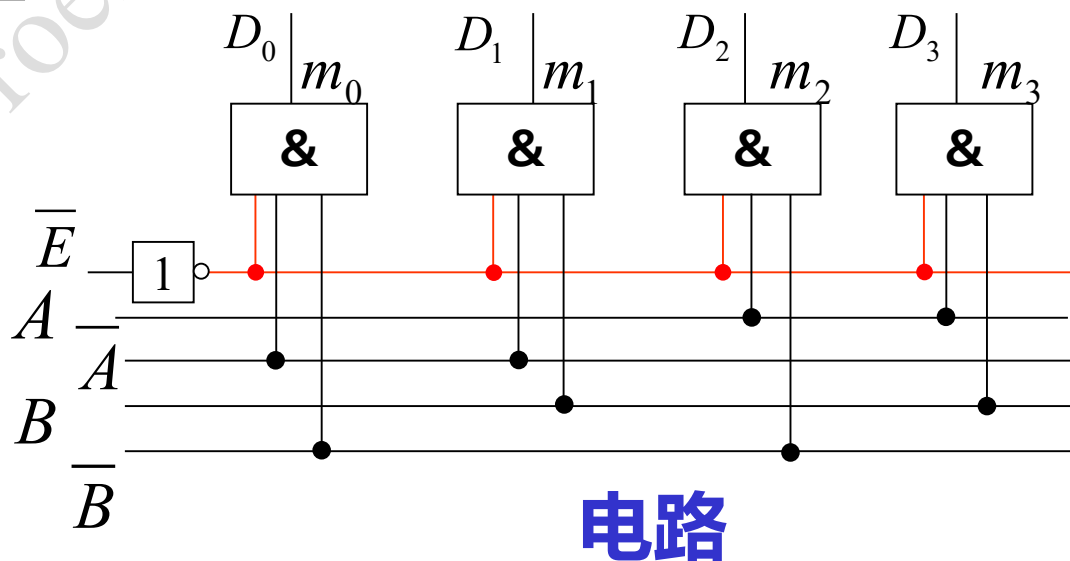
1) 输出高电平有效译码器 Active-High

Input		Output			
A	B	D_0	D_1	D_2	D_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

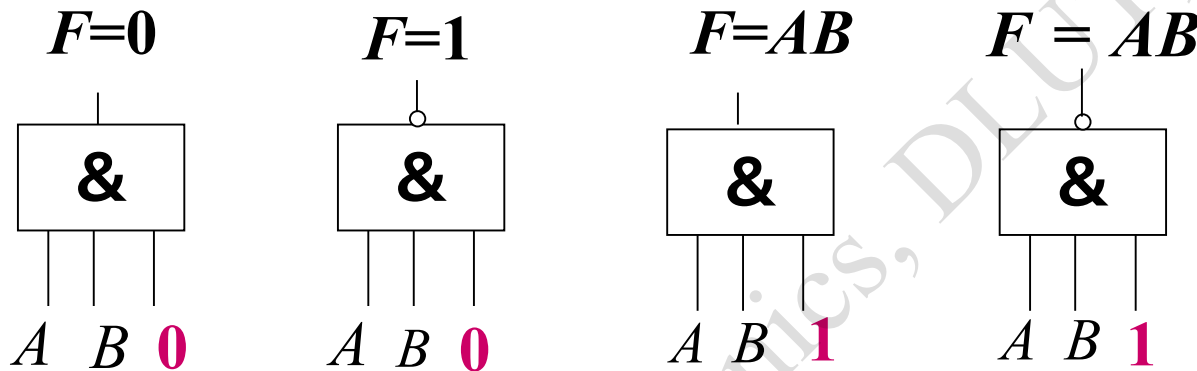
输入数码是二进制数几，
第几号输出就是唯一的高电
平，其余输出皆为低电平。

\overline{E} : enable

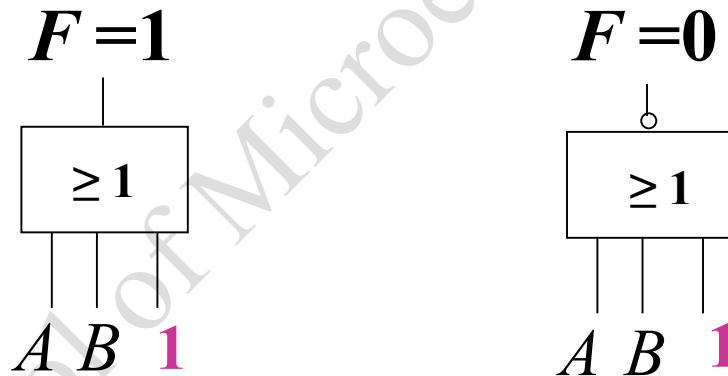
$\overline{E} = 0$, 译码器工作
 $\overline{E} = 1$, 译码器被锁住



当与门和与非门输入 0，逻辑门被锁住

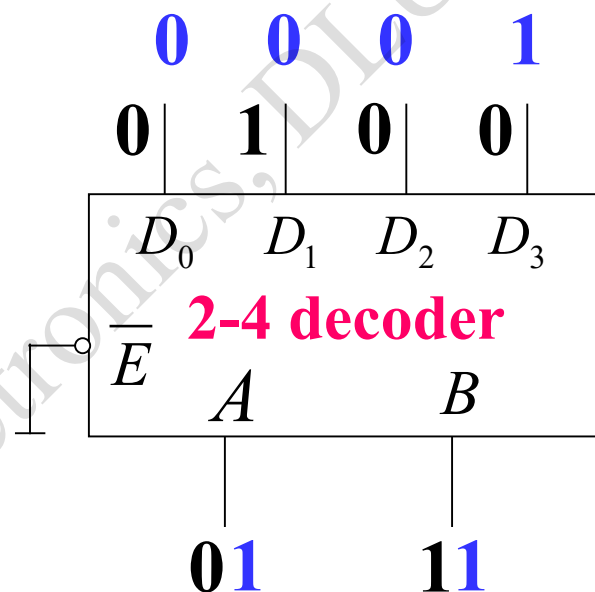


当或门和或非门输入 1，逻辑门被锁住



2 - 4 高电平有效译码器符号

A, B: 地址线



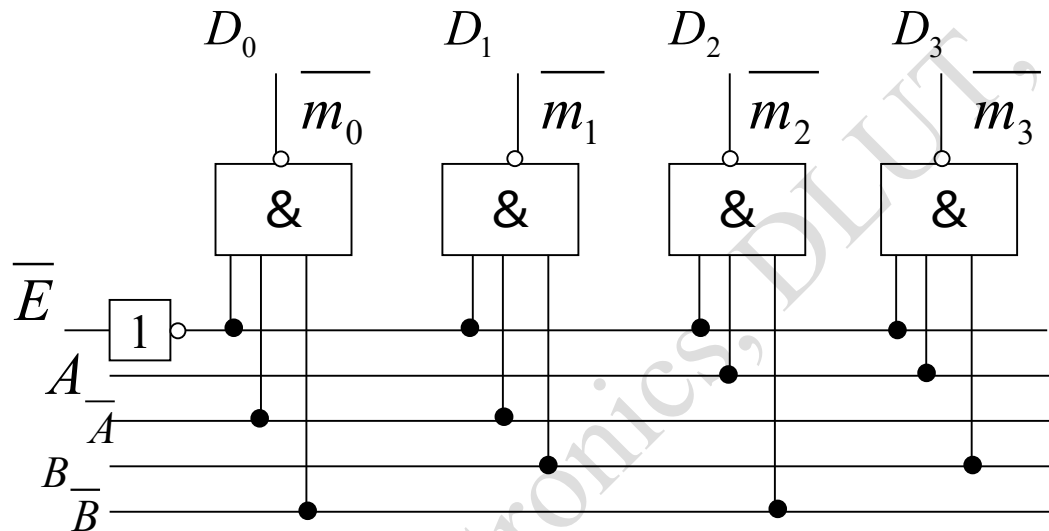
**注意：译码器的输出是标准形式
(最小项，最大项)**

2) 2-4 线低电平有效译码器 Active-Low

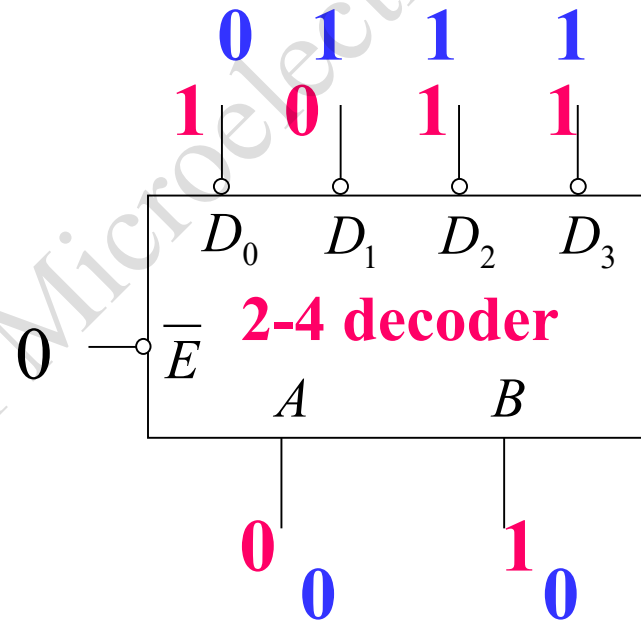
Input		Output			
A	B	D ₀	D ₁	D ₂	D ₃
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

输入数码是几，第几号输出就是唯一的低电平0，其余输出均是高电平1

电路



符号

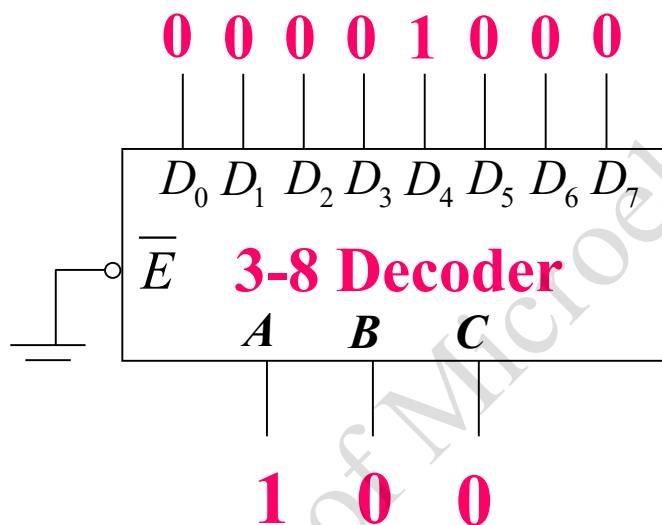


If $\bar{E} = 1$
 $D_0 D_1 D_2 D_3 = ?$

2. 3线-8线译码器

高电平有效 3-8 译码器

符号



译码器：输入数码是二进制数几，第几号输出就是唯一的有效电平，其余输出皆为无效电平。

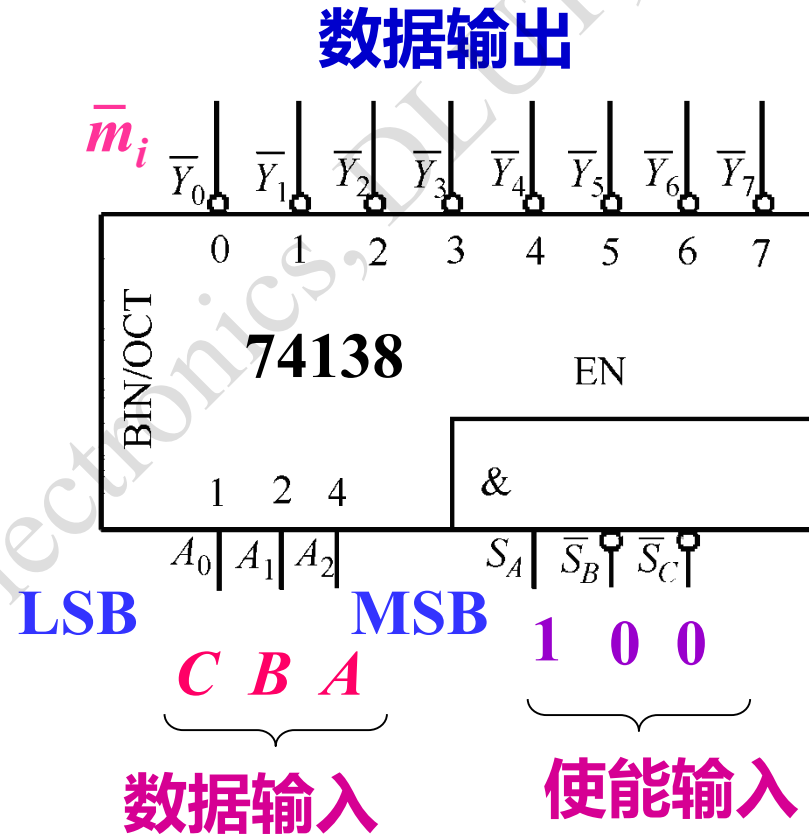
低电平有效 3-8 译码器： IC 74138

3 数据输入

8 输出

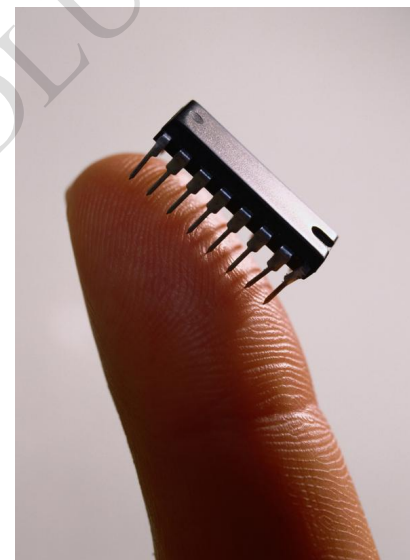
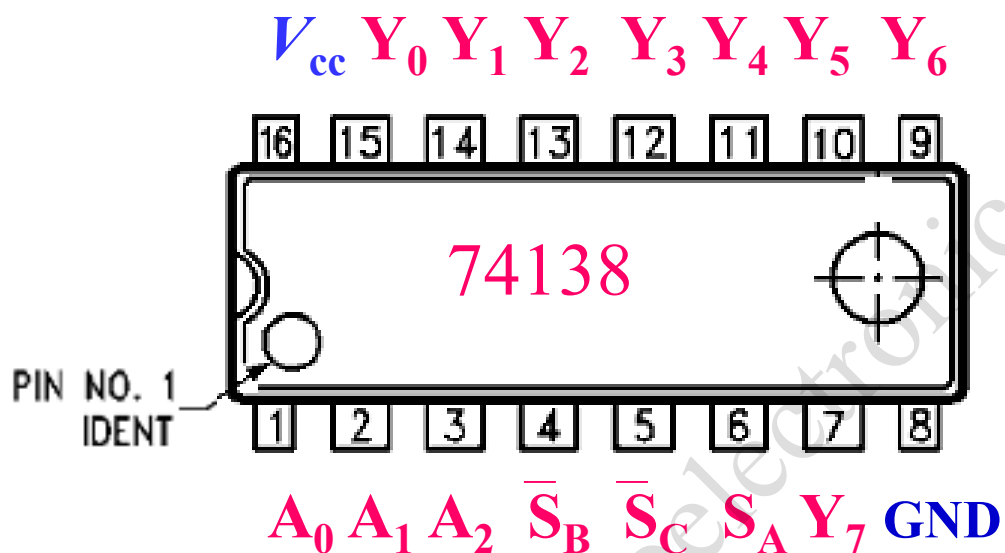
3 使能输入

$\left\{ \begin{array}{l} S_A \\ \bar{S}_B \\ \bar{S}_C \end{array} \right\}$ Active-high
Active-low



74138: MSI (medium scale integration)

管脚图



查手册 { 管脚图
功能表