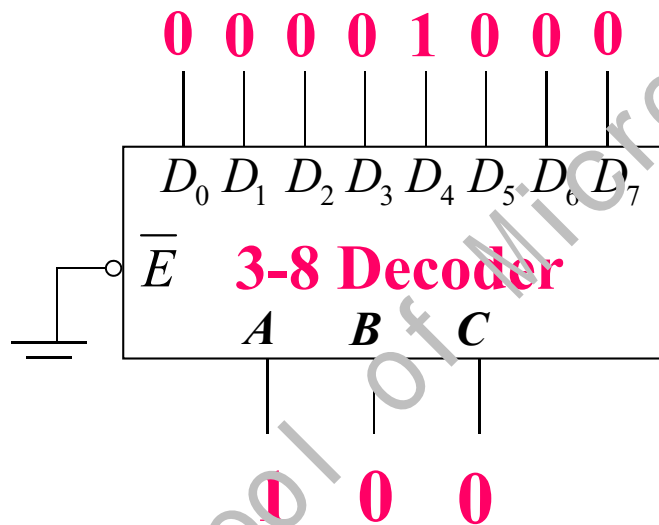


2. 3线-8线译码器

高电平有效 3-8 译码器

符号



简化逻辑符号

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

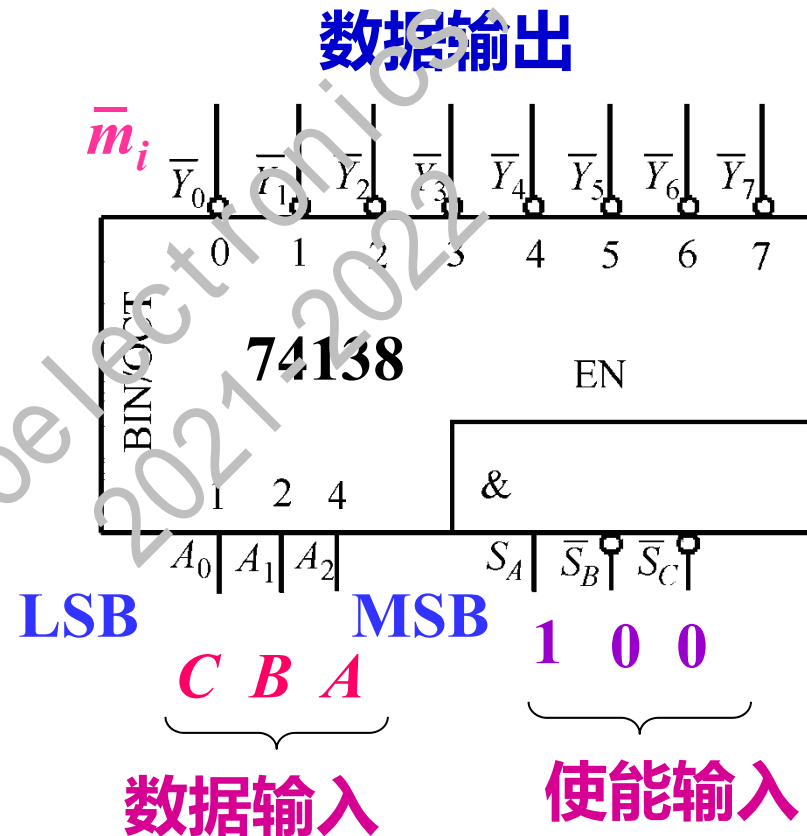
低电平有效 3-8 译码器： IC 74138

3 数据输入

8 输出

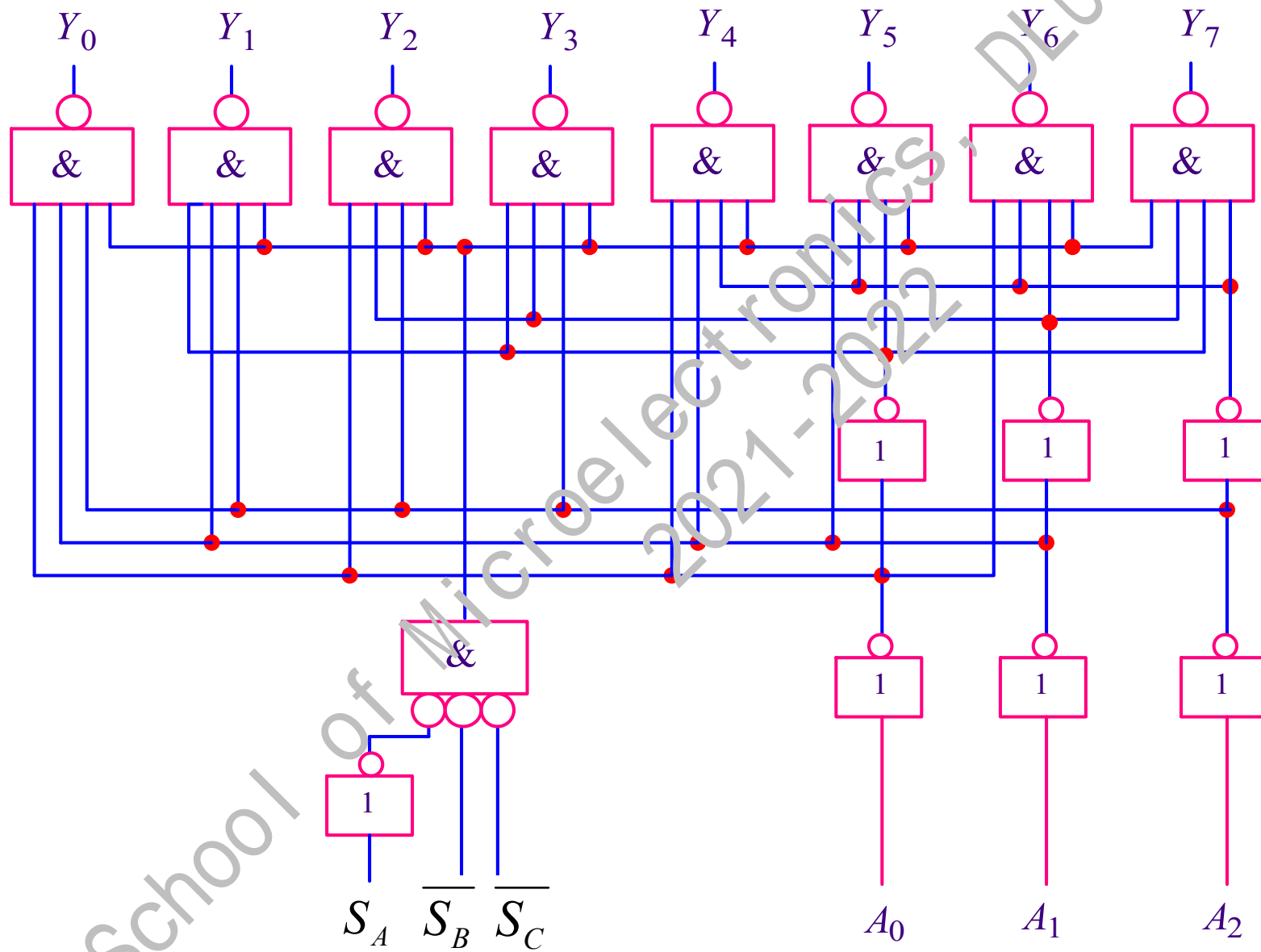
3 使能输入

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

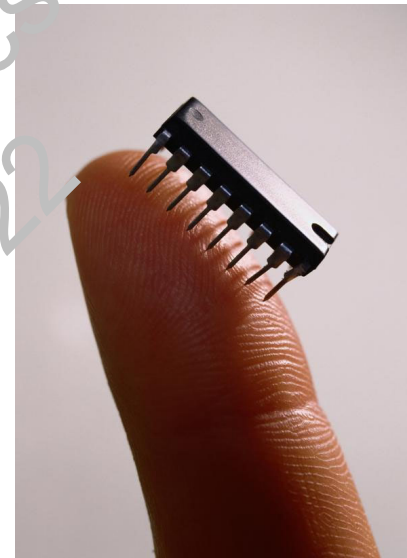
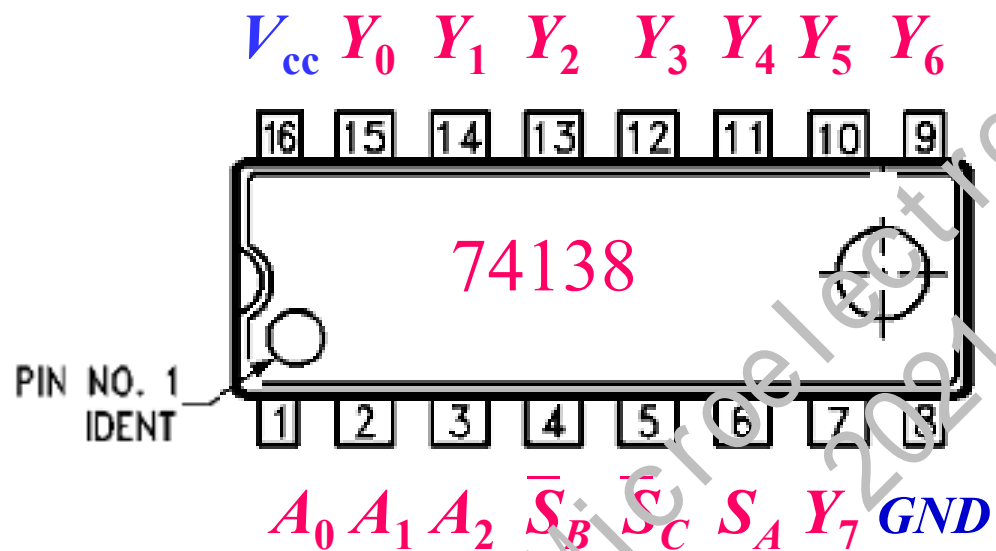


国标符号

74138: MSI (Medium Scale Integration)



管脚图



查手册

管脚图
功能表

3. 二进制译码器的应用——实现逻辑函数

例1: 用译码器和逻辑门实现下列一组函数

$$F_1(A, B, C) = \overline{A}\overline{B}C + B\overline{C} + \overline{A} \cdot \overline{C}$$

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

变成标准形式

F_1		AB			
		00	01	11	10
C	0	1	1	1	
	1				1

F_2		AB			
		00	01	11	10
C	0		0		
	1		0	0	

$$F_1(A, B, C) = \sum (0, 2, 5, 6) = \prod (1, 3, 4, 7)$$

$$F_2(A, B, C) = \sum (0, 1, 4, 5, 6) = \prod (2, 3, 7)$$

方法 1: 译码器 + 或门

标准与或式

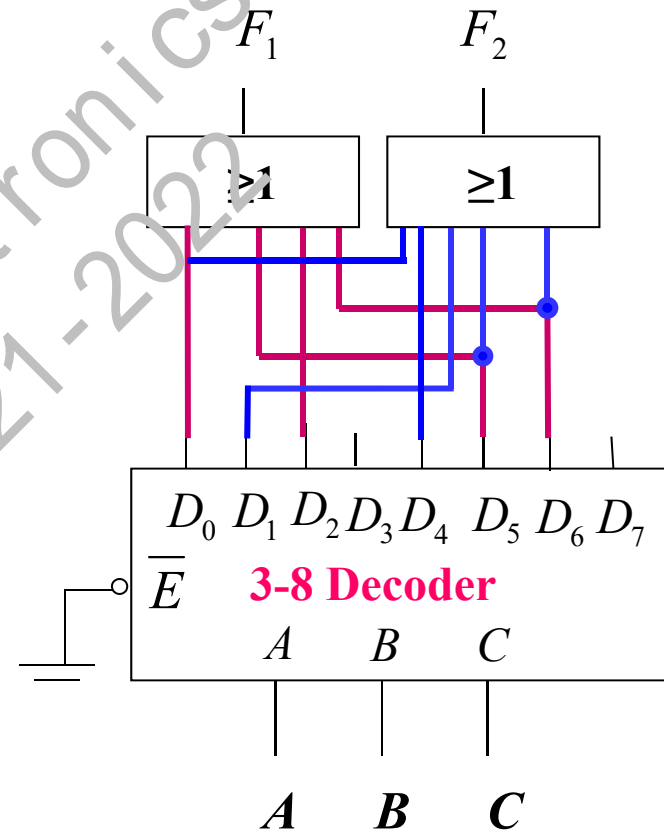
$$F_1(A, B, C) = \sum (0, 2, 5, 6)$$

$$F_2(A, B, C) = \sum (0, 1, 4, 5, 6)$$

输出: 最小项

高电平有效译码器

或门 → 最小项编号



方法2: 译码器 + 与非门

与或式 \rightarrow 与非门

$$F_1(A, B, C) = m_0 + m_2 + m_5 + m_6$$

$$= \overline{\overline{m_0 + m_2 + m_5 + m_6}}$$

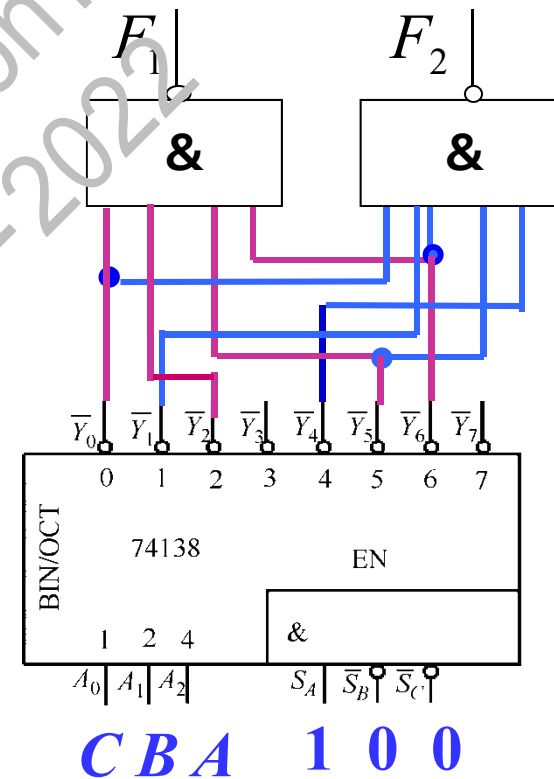
$$= \overline{m_0 \cdot m_2 \cdot m_5 \cdot m_6}$$

低电平有效译码器 (74138)

与非门 \rightarrow 最小项编号

$$F_1(A, B, C) = \sum (0, 2, 5, 6)$$

$$F_2(A, B, C) = \sum (0, 1, 4, 5, 6)$$



方法 3: 译码器 + 与门

标准或与式

$$F_1(A, B, C) = \sum (0, 2, 5, 6)$$

$$F_2(A, B, C) = \sum (0, 1, 4, 5, 6)$$

低电平有效译码器

$$F_1(A, B, C) = \Pi (1, 3, 4, 7)$$

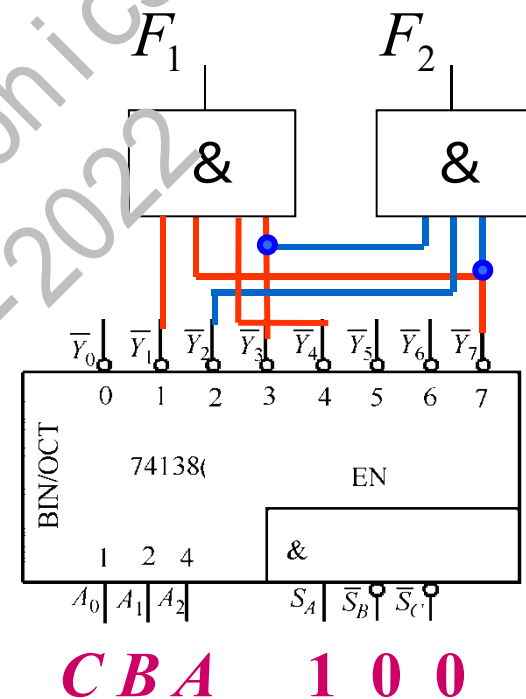
$$= M_1 \cdot M_3 \cdot M_4 \cdot M_7$$

$$= \bar{m}_1 \cdot \bar{m}_3 \cdot \bar{m}_4 \cdot \bar{m}_7$$

$$F_2(A, B, C) = \Pi (2, 3, 7)$$

$$= M_2 \cdot M_3 \cdot M_7$$

$$= \bar{m}_2 \cdot \bar{m}_3 \cdot \bar{m}_7$$



与门 → 最大项编号

结论:

用一个译码器实现一组函数

高电平有效译码器 + 或门 (最小项编号)

低电平有效译码器 + 与门 (与非门)

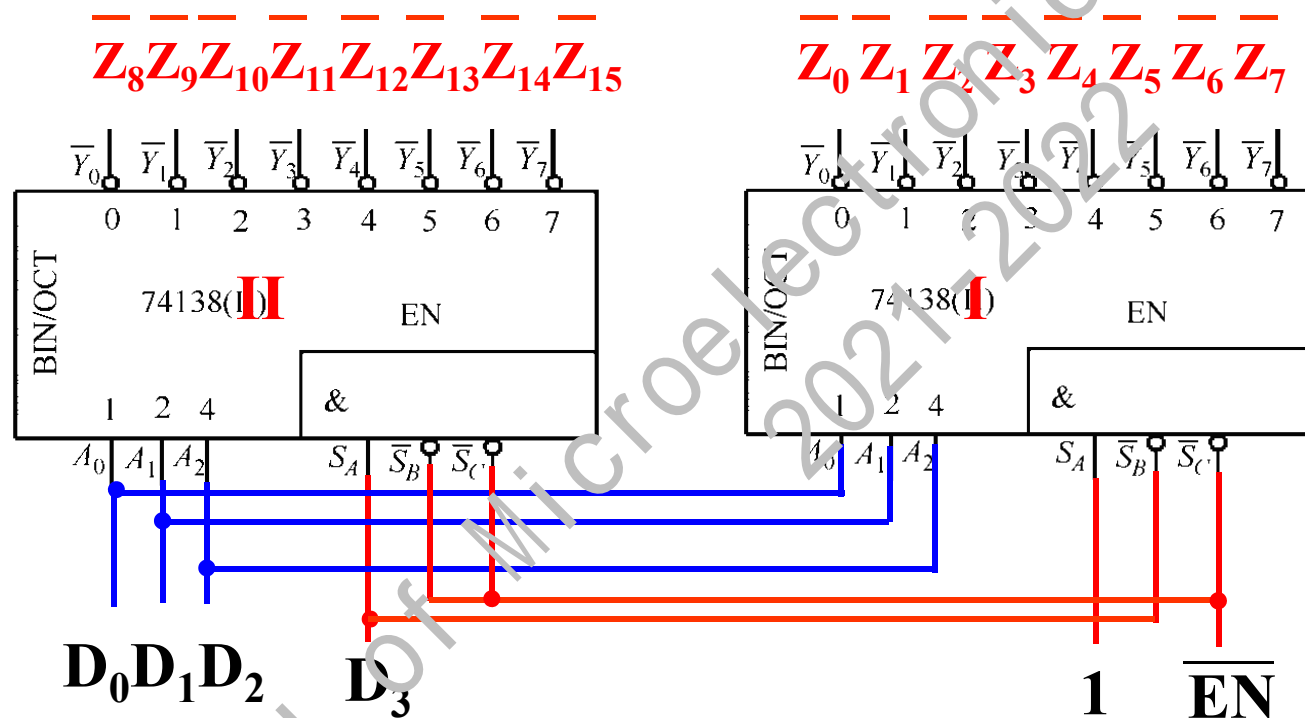
最大项编号

最小项编号

4. 二进制译码器的应用——用3-8线译码器扩展成4-16线译码器

用使能端扩展

数据输入： $D_3D_2D_1D_0$



$D_3D_2D_1D_0$

0 0 0 0

.....

0 1 1 1

1 0 0 0

.....

1 1 1 1

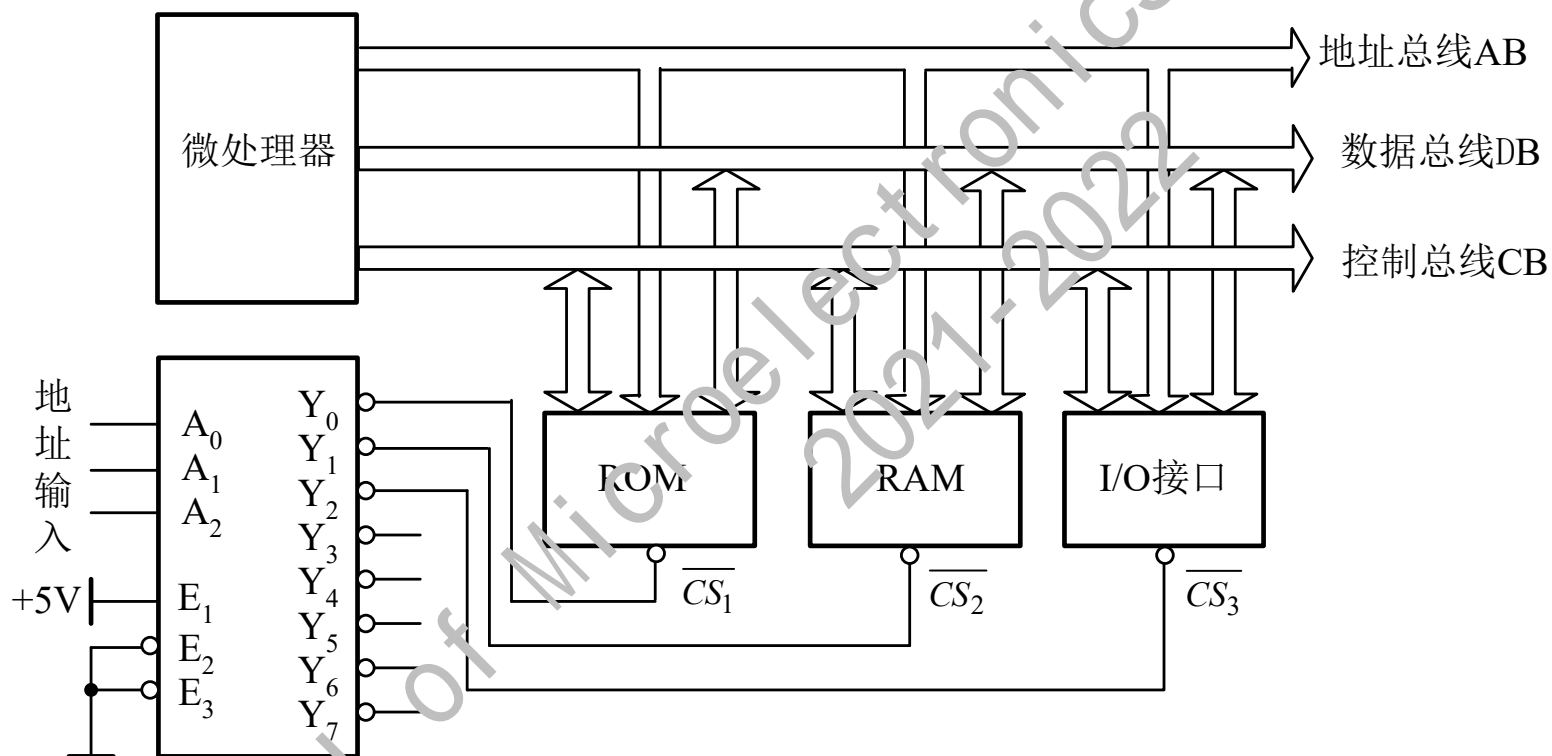
4-16线译码器 $S_A(I)=1$

D_3 : $S_A(I)$ 接 $S_B(I)$, 作为4-16线译码器MSB

$S_A(II)$ 、 $S_C(II)$ 、 $S_C(I)$ 作为4-16线译码器使能端

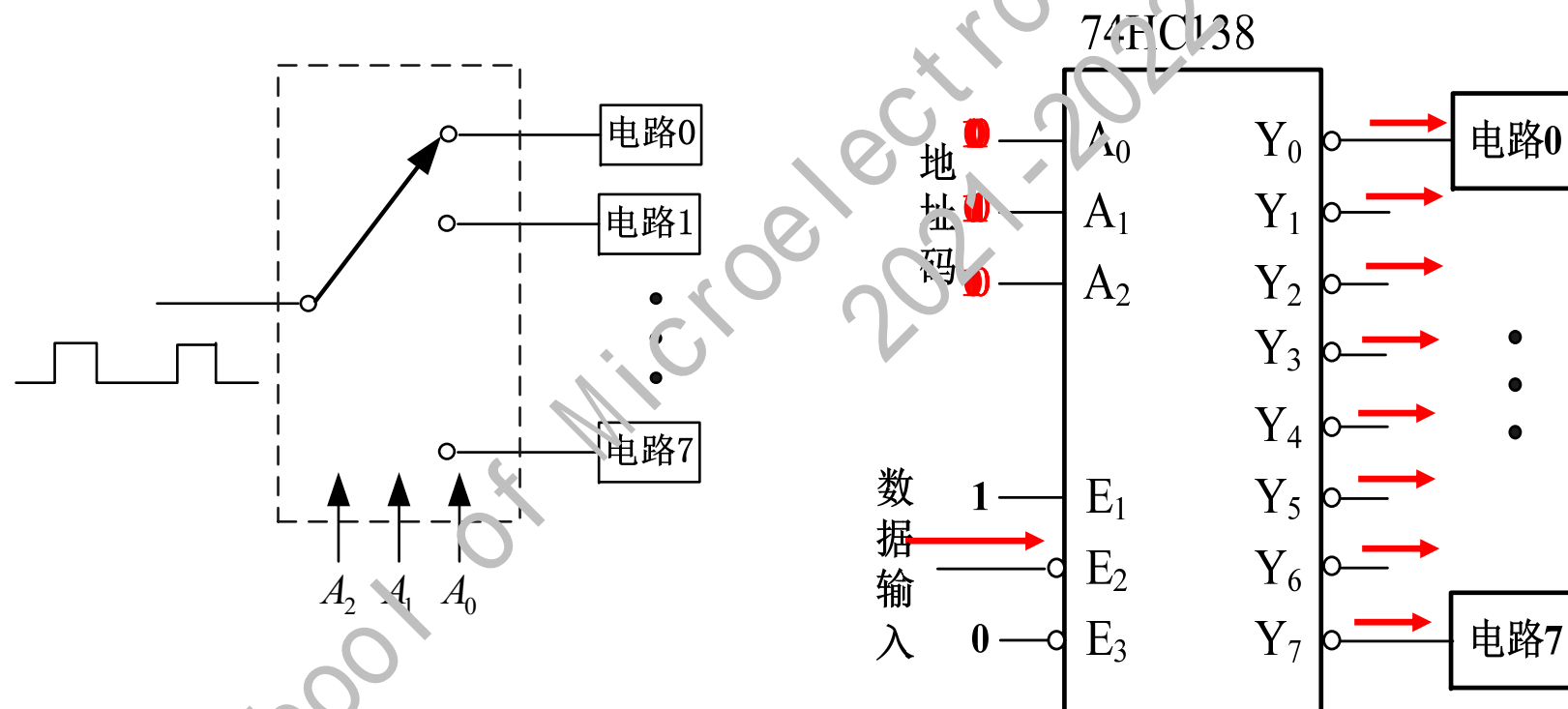
总使能端

5. 二进制译码器的应用——用74138构成地址译码器



6. 二进制译码器的应用——用74138构成数据分配器

由总线来的数字信号输送到不同的下级电路中去



思考：为什么数据从 $\overline{E_2}$ 输入？

§ 4.4.2 BCD码转十进制译码器

BCD-to-Decimal Decoders

功能: 将 **BCD** 码转换成十进制码

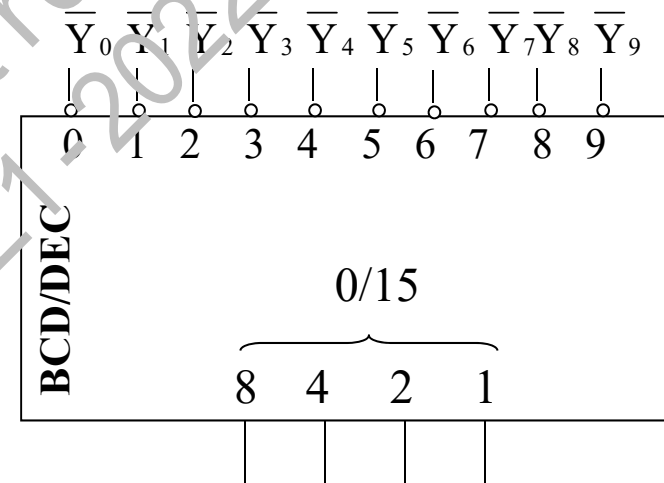
4—10线译码器 IC **7442**

注意:

输出: 低电平有效

输入: 有效输入 0000-1001

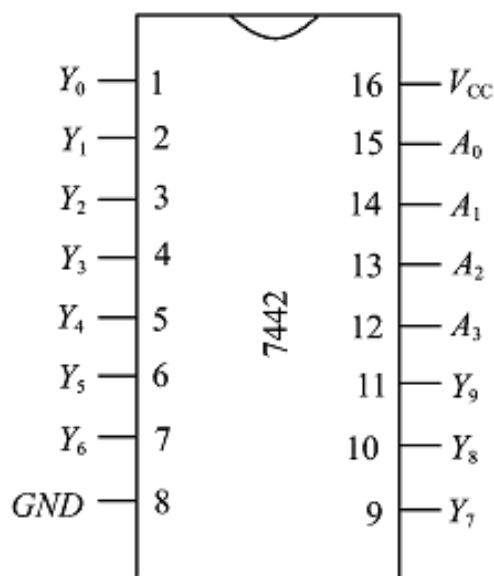
无效输入 1010-1111



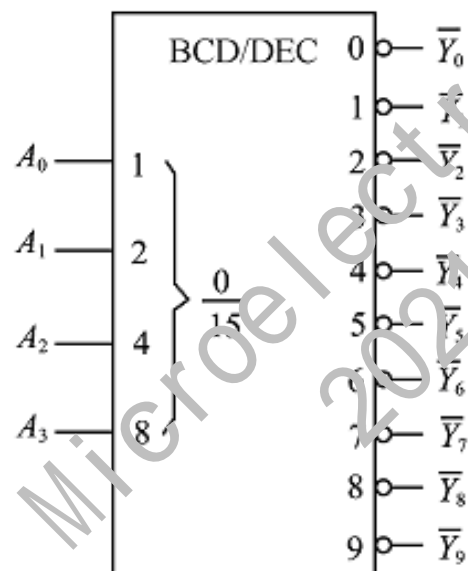
输入 BCD 码

输入数码是几, 第几号输出就是唯一的低电平0

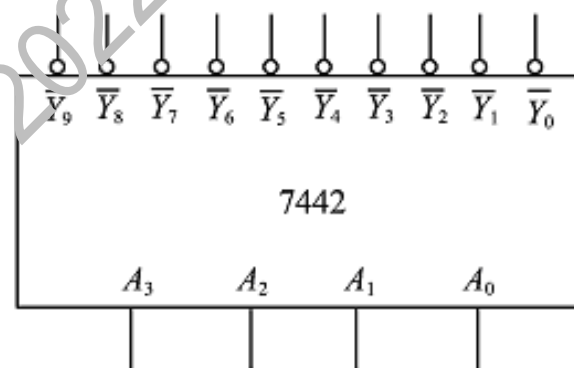
4线10线译码器7442管脚图和符号图



(b)管脚图



(c)国际标准符号



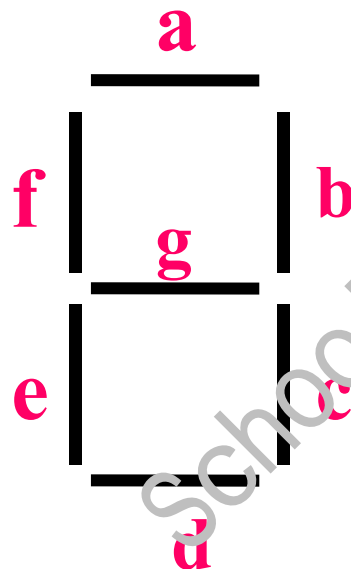
(b)惯用符号

§ 4.4.3 显示译码器 (/驱动器)

Display Decoder (/Driver)



1. 7段数码管



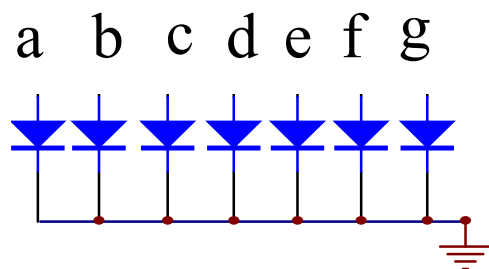
7 段数码管显示器是常见的显示器

数码管由7段发光管构成

LED: Light Emitting Diode 半导体发光二极管

LCD: Liquid Crystal Display 液晶显示器

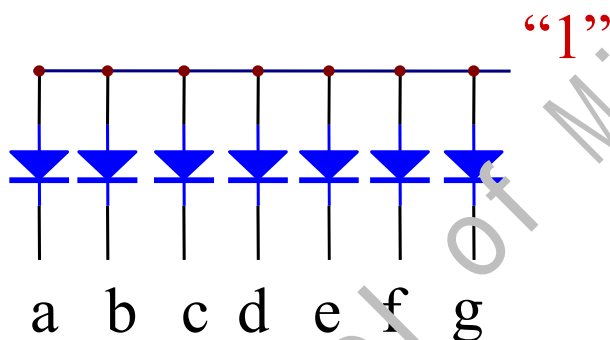
连接方式不同分成**共阴极**和**共阳极**两种



BS201A

共阴极

二极管 → 逻辑**高** → 亮



BS201B

共阳极

二极管 → 逻辑**低** → 亮

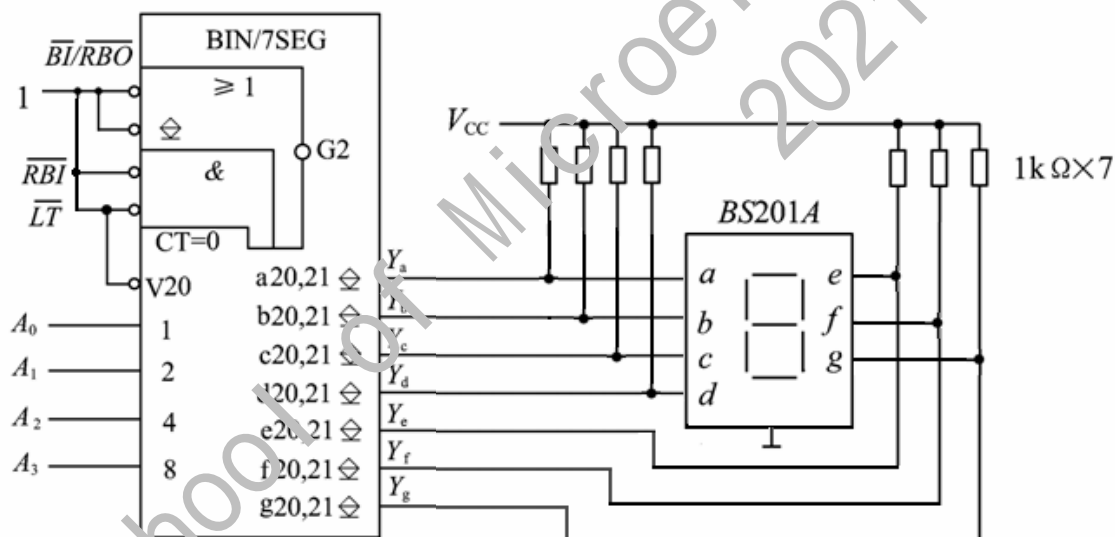
2. 显示译码器

要显示0 - 9 十个数字，需要用译码器来驱动

显示译码器 / 驱动器 7448

输入 4 线 4 位二进制数 / 8421 BCD 码

输出 7 线 \longrightarrow 驱动 7 段数码管



输出高有效，
驱动共阴极管

不一定只有一个输出端高（或低）有效

7448的逻辑功能

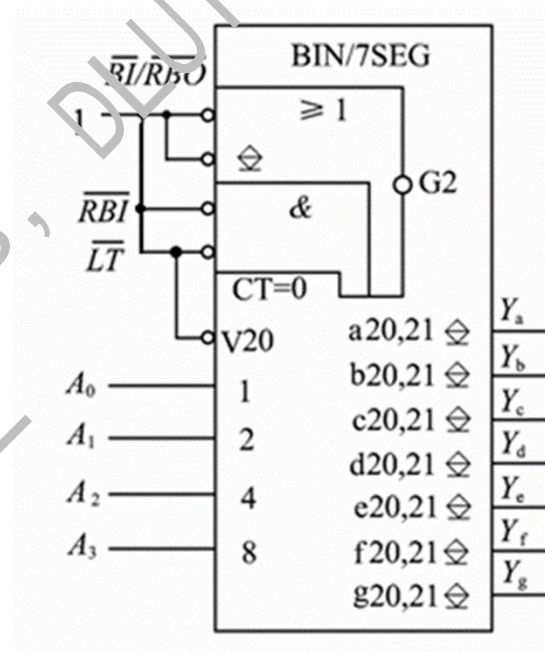
(1) **正常译码显示**。当 $LT=1$ ， $BI/RBO=1$ 时，对输入为十进制数1~15的二进制码（0001~1111）进行译码，产生对应的七段显示码

(2) **灭零**。当 $LT=1$ ，而输入为0的二进制码0000时，只有当 $RBI=1$ 时，才产生0的七段显示码；如果此时 $RBI=0$ ，则译码器的 $a \sim g$ 输出全0，使显示器全灭；所以 RBI 称为**灭零输入端**

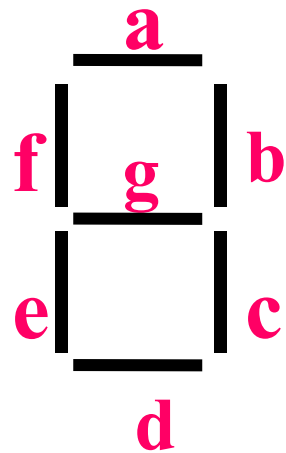
(3) **试灯**。当 $LT=0$ 时，无论输入怎样， $a \sim g$ 输出全1，数码管七段全亮。由此可以检测显示器七个发光段的好坏。 LT 称为**试灯输入端**。

(4) **特殊控制端 BI/RBO** 。 BI/RBO 可以作输入端，也可以作输出端。

- 作输入使用时，如果 $BI=0$ 时，不管其他输入端为何值， $a \sim g$ 均输出0，显示器全灭。因此 BI 称为**灭灯输入端**
- 作输出端使用时，受控于 RBI 。当 $RBI=0$ ，输入为0的二进制码0000时， $RBO=0$ ，用以指示该片正处于灭零状态。所以， RBO 又称为**灭零输出端**



显示译码器内部电路设计



A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	0	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	0	0	1	1

Display

0
1
2
3
4
5
6
7
8
9

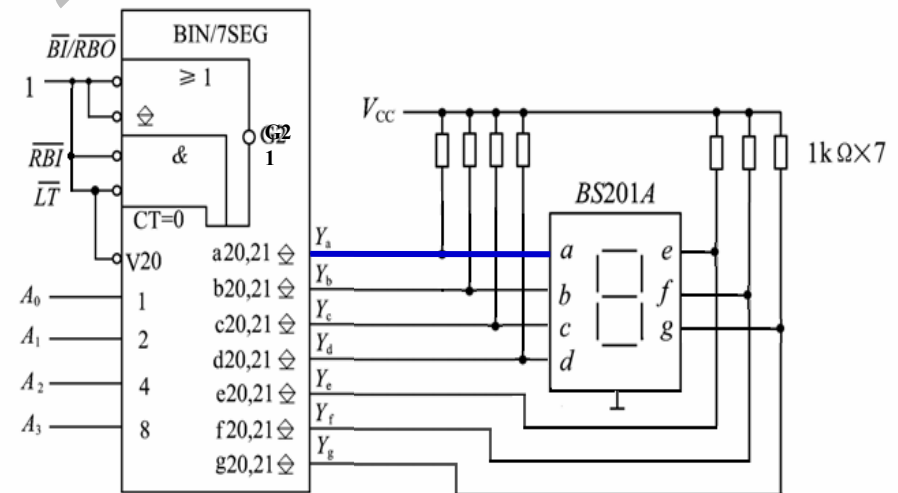
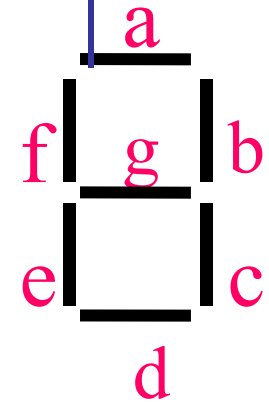
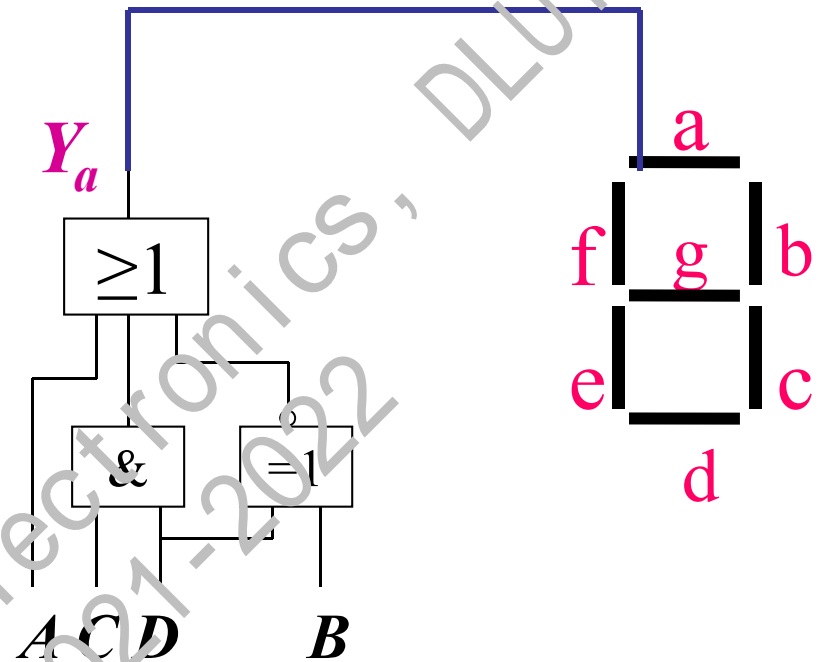
分别做7个卡诺图

Y_a

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

$$Y_a = A + \bar{B} \cdot \bar{D} + BD + CD$$

$$= A + CD + \overline{B \oplus D}$$



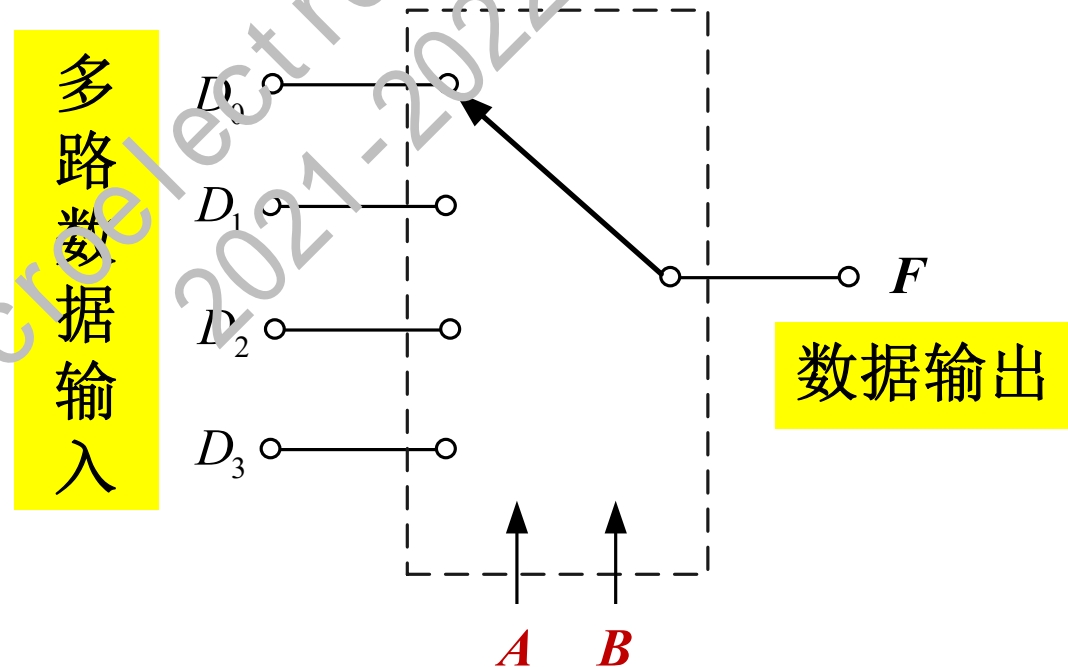
§ 4.5 多路（数据）选择器 MUX

Multiplexers (Data Selectors)

功能：在多路输入数据中选择一路进行输出

1. 4 线-1线 MUX

4个数据 D_0, D_1, D_2, D_3 中选一个输出，
选择开关由地址 $A B$ 控制



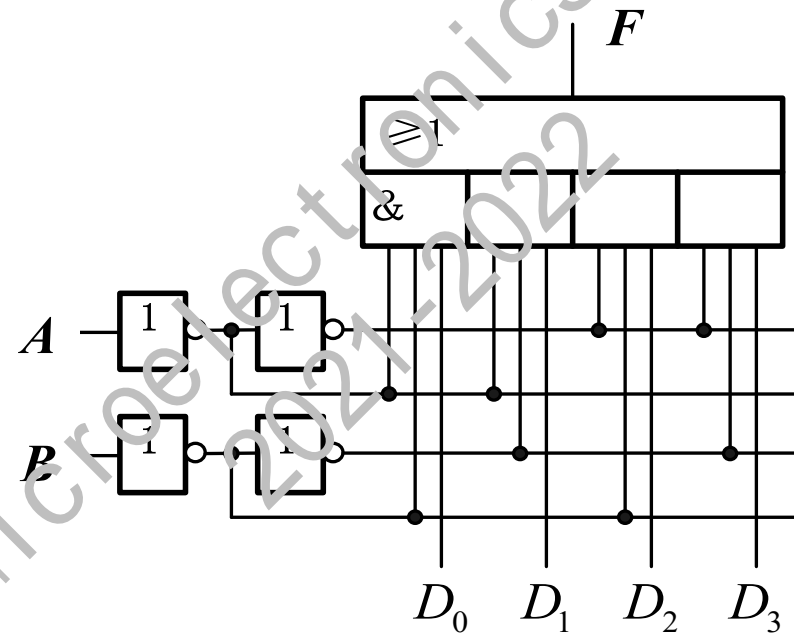
$A B$: 控制输入 (地址输入)

n 位地址线可以控制 2^n 个数据输入

真值表 (AB 和 D_i ($i=0\dots3$) 当作输入, F 为输出)

A	B	D_i	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

$$F = \overline{A}\overline{B}D_0 + \overline{A}BD_1 + A\overline{B}D_2 + ABD_3$$



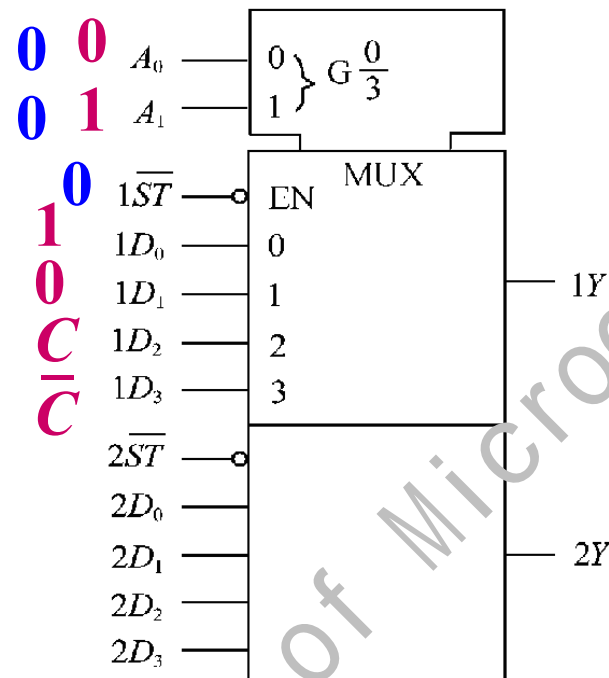
控制码是几, 就把第几号数据送到唯一的输出端

AB 任取一值时, 最多只有一个与门输出1(D), 其他为0, 取或之后为 F

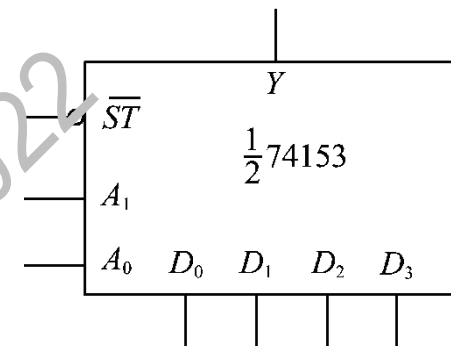
Decoder + Data lines + OR gate

MSI 4 - 1 MUX 74153 (一芯片上有 2 个 4 - 1 MUX)

符号



国际标准符号

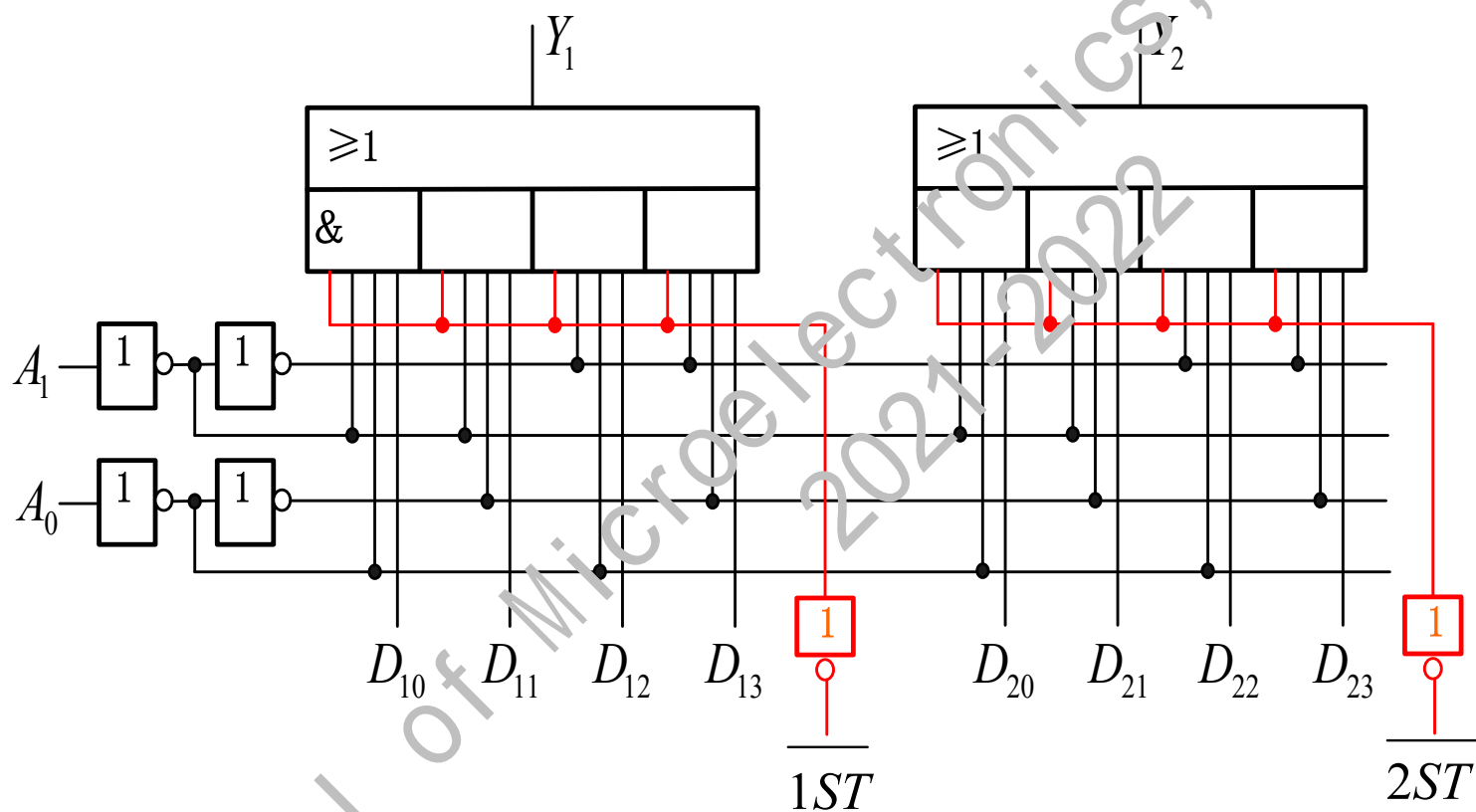


惯用符号 (简化符号)

\overline{ST} **Select Transform** 选通端, 低电平有效

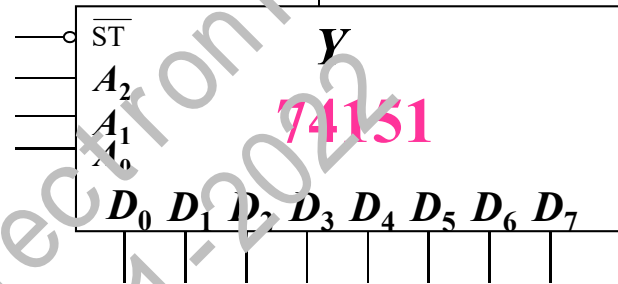
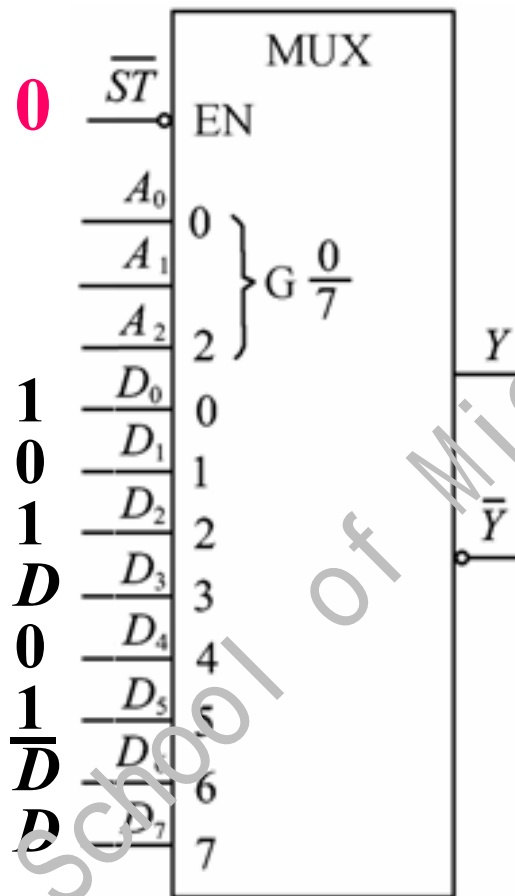
$A_1 A_0$: **地址线** (控制输入)

双4选1数据选择器——74153



2. 8线-1线 MUX 74151 (MSI)

3 位地址线: $A_2 A_1 A_0$; 8 条数据线: $D_0 - D_7$



地址输入

A_2	A_1	A_0	F
1	0	0	0
0	1	1	D
0	0	0	1
1	1	0	\overline{D}

3. MUX实现逻辑函数

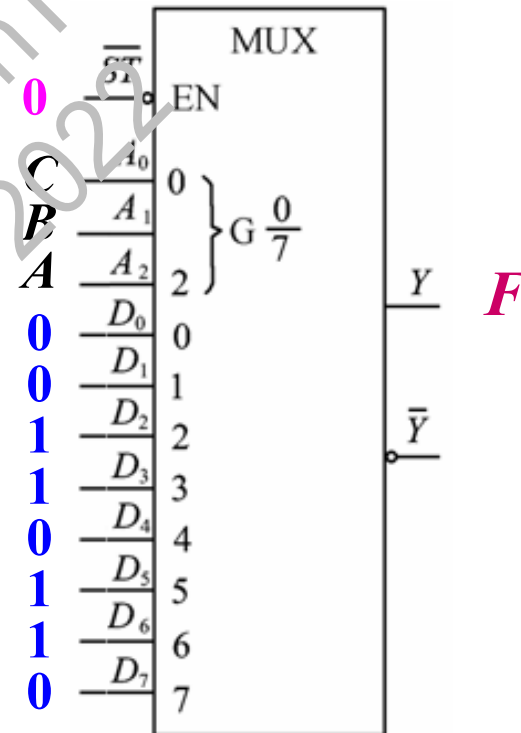
例 1: 用MUX 实现函数

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

解: 3 变量

选择 74151 (8-1 MUX)

F AB		C			
		00	01	11	10
C	0		1	1	
	1		1		1



一个 MUX 只能实现一个逻辑函数

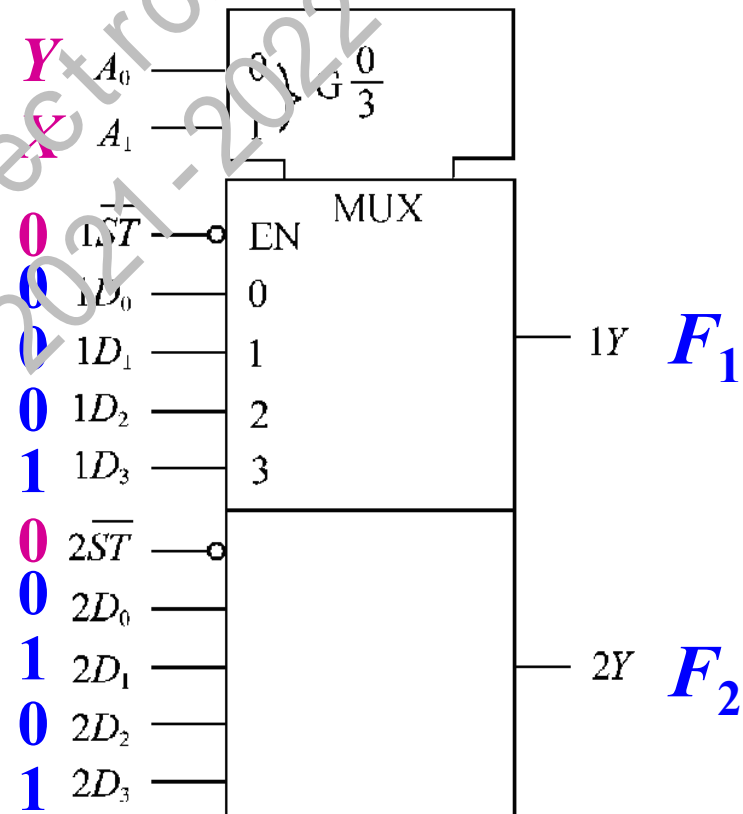
例 2: 用双4选1 MUX 74153 实现下列函数

$$F_1(X, Y) = X(\bar{X} + Y) = XY = m_3$$

$$F_2(X, Y) = \prod(0, 2)$$

解:

标准形式



例 3: 用一片 74151 实现下列函数

$$F(A, B, C, D) = ABCD + \overline{A}\overline{B}C\overline{D} + \overline{A}BCD + \overline{A}\overline{B}\overline{C} + ABC\overline{D}$$

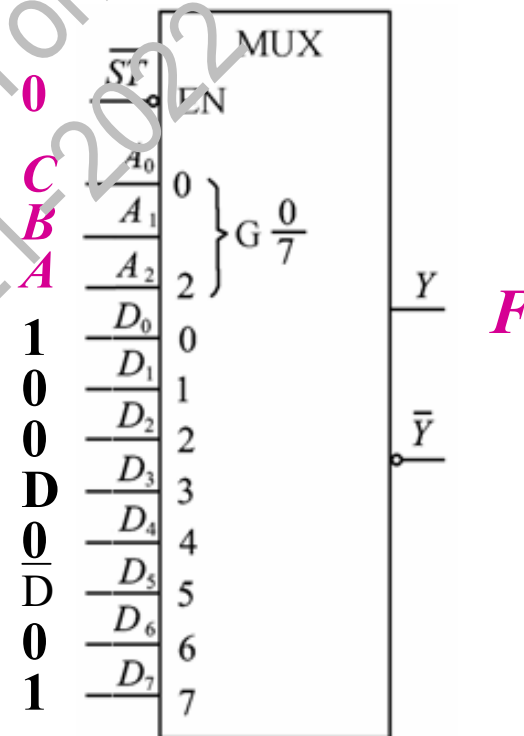
解:

74151 3变量

$D \rightarrow VEM$

F	AB	C			
		00	01	11	10
0		1	0	0	0
1		0	D	D + \overline{D}	\overline{D}

1



例 4. 用一片4-1 MUX实现

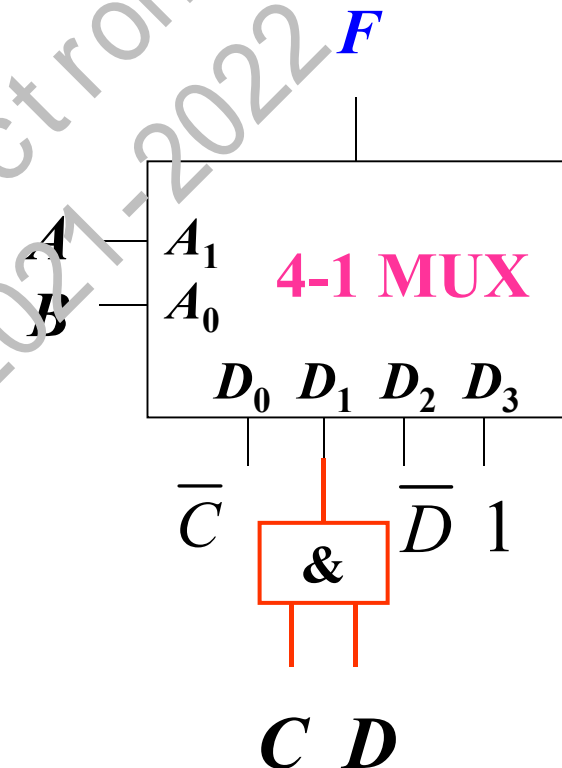
$$F(A,B,C,D) = \overline{A}BCD + AB + \overline{A} \cdot \overline{B} \cdot \overline{C} + \overline{A}\overline{B} \cdot \overline{D}$$

解:

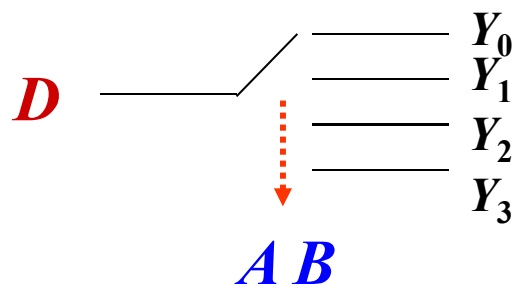
4-1 MUX 2 变量

2 变量 C 、 $D \rightarrow$ VEM

		A	
		0	1
B	0	\overline{C}	\overline{D}
	1	CD	1



数据分配器 Demultiplexers (DEMUX) (Data Distributors)



A	B	Y_0	Y_1	Y_2	Y_3
0	0	D	0	0	0
0	1	0	D	0	0
1	0	0	0	D	0
1	1	0	0	0	D

控制数码是几，就把输入数据送到第几路输出端

1-4 DEMUX

1-8 DEMUX

1-16 DEMUX

Decoder + Data

常用译码器实现数据分配

例：用74138实现1-8DEMUX

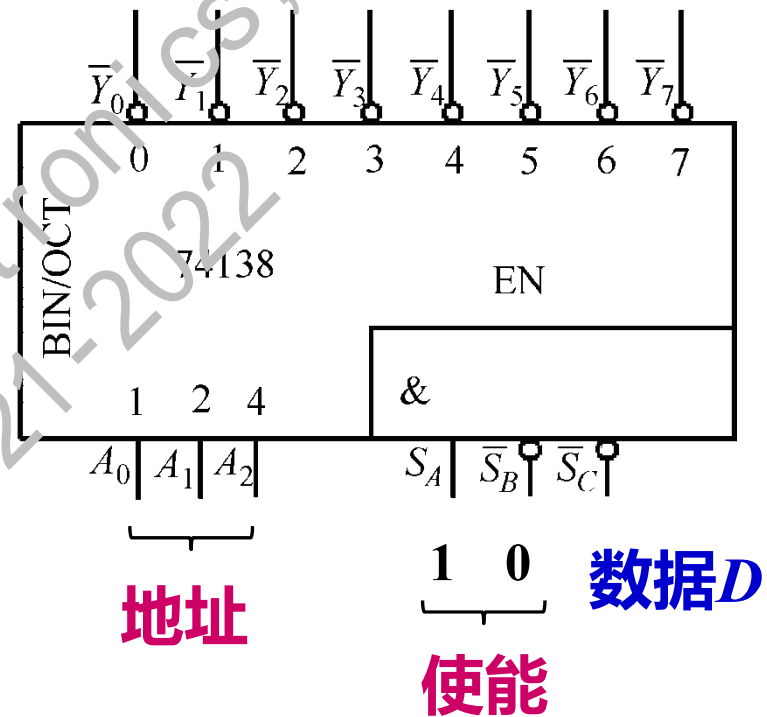
一个使能端 $\overline{S_c}$ 作为数据输入

$A_2 A_1 A_0$ 作为地址输入

如 $A_2 A_1 A_0=110$, $\overline{Y_6}$ 为输出端

$\overline{S_c} = D = 0$, 译码器工作, $\overline{Y_6} = 0$

$\overline{S_c} = D = 1$, 译码器被锁住, $\overline{Y_6} = 1$

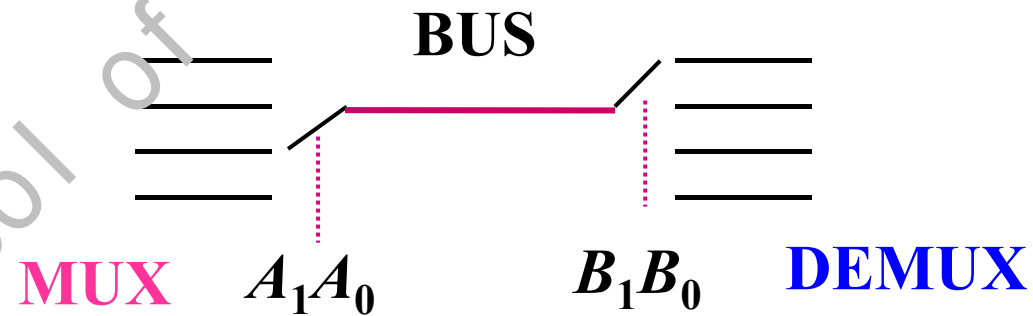


MUX 应用

1) 实现逻辑函数

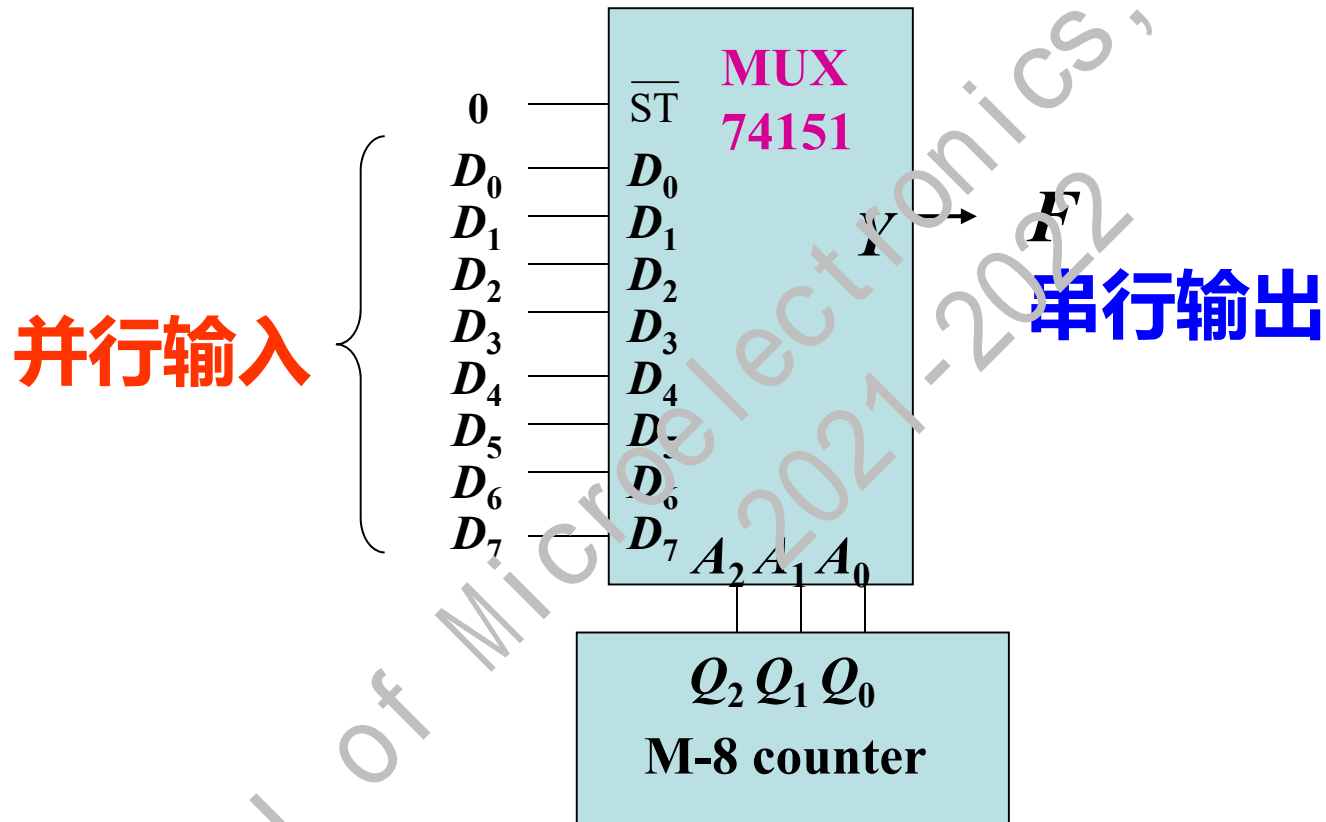
2) 多路数字开关

路由选择，将 MUX 和 DEMUX 结合使用，实现时分多路数据通信。



3) 数据并行/串行转换

MUX和计数器



计数器从000 ~ 111循环，使MUX依次选择 $D_0 \sim D_7$ 输出。

智能压力传感器框图

