3.6 数据选择器应用之：实现逻辑函数

对于n个地址输入的MUX，其表达式为：

$$Y=\sum_{i=0}^{2^n-1} m_i D_i$$

其中mi是由地址变量An-1、…、A1、A0组成的地址最小项。

任何一个具有l个输入变量的逻辑函数都可以用最小项之和来表示：

$$F=\sum_{i=0}^{2^l-1} m_i (1,or 0)$$

这里的mi是由函数的输入变量A、B、C、…组成的最小项。

1) $l \le n$的情况

(l为函数的输入变量数，n为选用的MUX的地址输入端数)

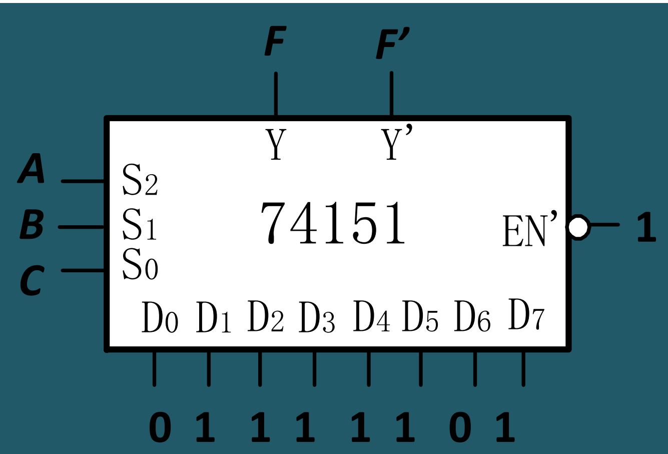
当$l=n$时，只要将函数的输入变量A、B、C、…依次接到MUX的地址输入端，根据函数F所需要的最小项，确定MUX中D_i的值(0或1)即可；

当$l<n$时，将MUX的高位地址输入端不用(接0或1)，其余同上。

$$Y = \sum_{i=0}^{2^n-1} m_i D_i \qquad F = \sum_{i=0}^{2^l-1} m_i (1, or 0)$$

例1. 用8选1 实现下面的组合逻辑函数.

$$F=\sum m(1,2,3,4,5,7)$$



从*F*可以看出，该逻辑函数包含3 个变量 (设为A、B、C)，一个 8选1 数据选择器有 3 个地址输入端。

$$Y=\sum_{i=0}^{7} m_i D_i=(A_2A_1A_0)(D_0D_1D_2D_3D_4D_5D_6D_7)^T$$

$$=m_0\cdot D_0+m_1\cdot D_1+m_2\cdot D_2+m_3\cdot D_3+m_4\cdot D_4+m_5\cdot D_5+m_6\cdot D_6+m_7\cdot D_7$$

$$F=\sum m(1,2,3,4,5,7)=m_1+m_2+m_3+m_4+m_5+m_7$$

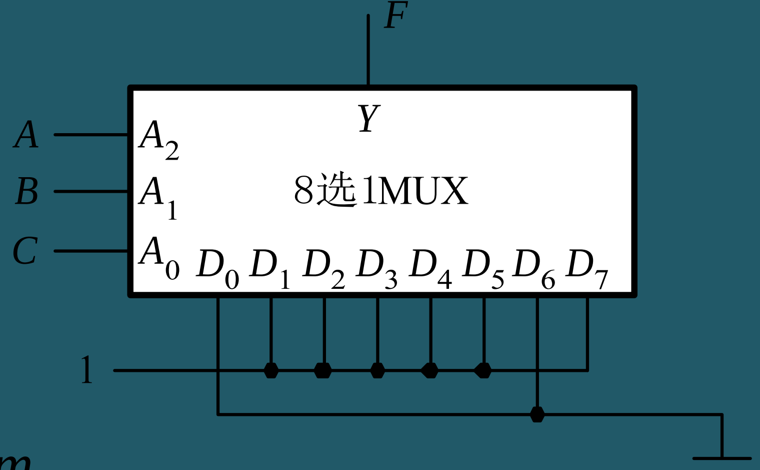
$$=m_0\cdot 0+m_1\cdot 1+m_2\cdot 1+m_3\cdot 1+m_4\cdot 1+m_5\cdot 1+m_6\cdot 0+m_7\cdot 1$$

例2. 试用8选1MUX实现逻辑函数：

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

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

F



$$F = \sum m(1,2,3,4,5,7) = m_1 + m_2 + m_3 + m_4 + m_5 + m_7$$

$$= m_0 \cdot 0 + m_1 \cdot 1 + m_2 \cdot 1 + m_3 \cdot 1 + m_4 \cdot 1 + m_5 \cdot 1 + m_6 \cdot 0 + m_7 \cdot 1$$

需要注意:因为函数F中各最小项的标号是按A、B、C的权为4、2、1写出的，因此A、B、C必须依次加到A_2、A_1、A_0端。

数据选择器应用之：实现逻辑函数

当l>n的情况

当逻辑函数的变量数l大于MUX的地址输入端数n时，不能采用前面所述的简单方法。

如果从l个输入变量中选择n个直接作为MUX的地址输入，那么，多余的$(l-n)$个变量就要反映到MUX的数据输入D_i端，即D_i是多余输入变量的函数，简称余函数。

因此设计的关键是如何求出函数$D_i=f(l-n)$。

例. 用4选1实现逻辑函数： $F=\sum m(1,2,3,4,5,7)$

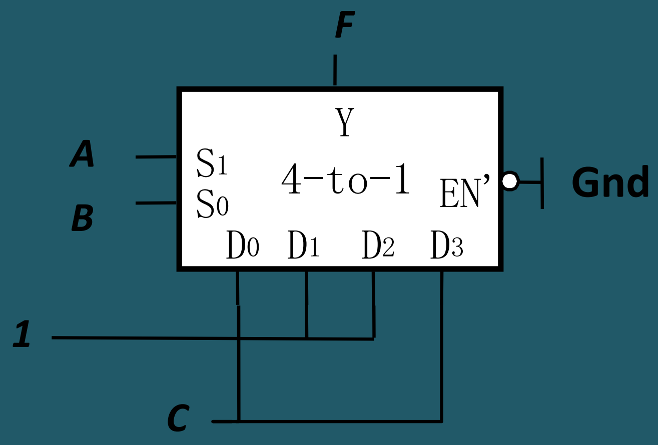
变量数 *l* > 数据选择器地址输入端数*n*

$$Y=\sum_{i=0}^{3} m_i D_i=(A_1A_0)(D_0D_1D_2D_3)^T=m_0\cdot D_0+m_1\cdot D_1+m_2\cdot D_2+m_3\cdot D_3$$

$$F=\sum m(1,2,3,4,5,7)$$
$$=\bar{A}\bar{B}C+\bar{A}B\bar{C}+\bar{A}BC+A\bar{B}\bar{C}+A\bar{B}C+ABC$$

必须选择2个变量作为数据选择器的地址端

$$F=\bar{A}\bar{B}C+\bar{A}B\bar{C}+\bar{A}BC+A\bar{B}\bar{C}+A\bar{B}C+ABC$$
$$=m_0C+m_1\bar{C}+m_1C+m_2\bar{C}+m_2C+m_3C$$
$$=m_0C+m_1\cdot 1+m_2\cdot 1+m_3C$$



降维K图法求余子式D_i

n变量的逻辑函数，可以用n维(即n变量)K图表示，也可以用(n-1)、(n-2)、…维K图表示，这种(n-1)、(n-2)、…维K图称为降维K图。

降维K图法求余子式D_i的求解步骤：

① 画出函数F的K图

② 选择地址输入

③ 在F的K图上确定余函数D_i的范围(子K图)

④ 求余函数D_i

⑤ 画出逻辑图

例. 试用8选1MUX实现逻辑函数：$F(A,B,C,D)=\sum m(0,4,5,7,12,13,14)$

① 画出F的四变量K图

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

② 选择地址变量,确定余函数Di

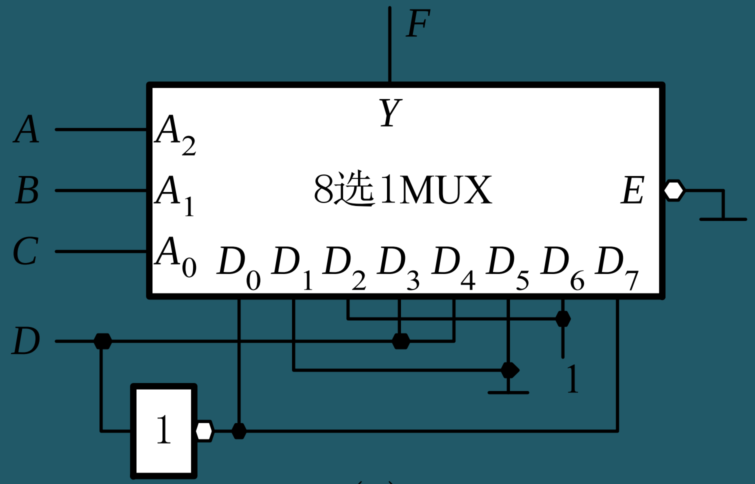
CD \ AB	00	01	11	10
00	1	1	1	0
	D_0	D_2	D_6	D_4
01	0	1	1	1
11	0	1	0	0
	D_1	D_3	D_7	D_5
10	0	0	1	0

选择地址（$A_2A_1A_0$=ABC）

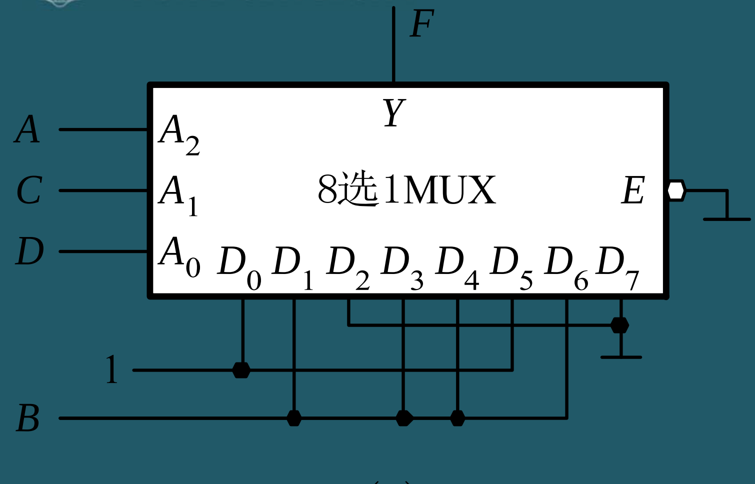
CD \ AB	00 01	11 10
00	1 D_0 1	1 D_4 0
01	0 D_1 1	1 D_5 1
11	0 D_3 1	0 D_7 0
10	0 D_2 0	1 D_6 0

选择地址（$A_2A_1A_0$=ACD）

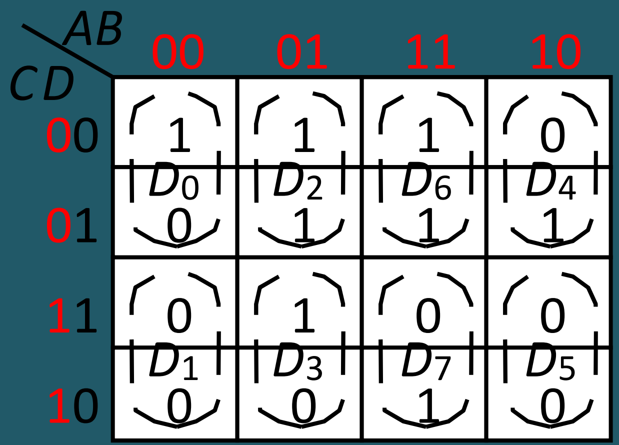
分别确定D_0、D_1、…、D_7。



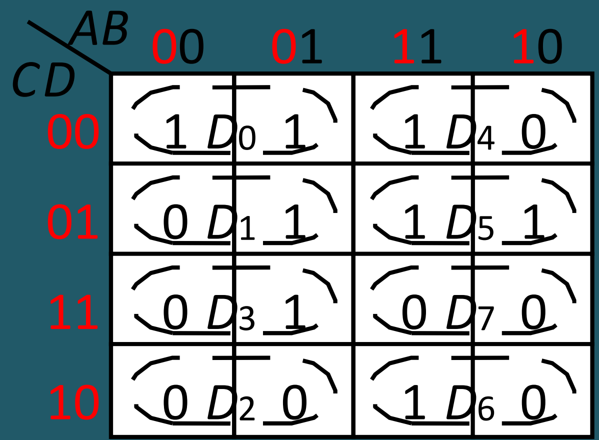
(a)



(b)



选择地址（$A_2A_1A_0$=ABC）



选择地址（$A_2A_1A_0$=ACD）