数字逻辑基础

主 讲：何宾

Email：hebin@mail.buct.edu.cn

2014.06

逻辑代数理论
--背景

逻辑代数(switching algebra)，称为交换代数，也叫做开关代数。

- 起源于英国数学家乔治·布尔(George Boole)于1849年创立的布尔代数，是数字电路设计理论中的数字逻辑科目的重要组成部分。
- 逻辑变量之间的因果关系以及依据这些关系进行的布尔逻辑的推理，可用代数运算表示出来，这种代数称为逻辑代数。
- 逻辑代数是一个由逻辑变量集、常量 0 和 1 及“与”、“或”、“非”三种运算所构成的代数系统。

逻辑代数理论
--逻辑代数中运算关系

参与逻辑运算的变量叫逻辑变量，用字母A，B……表示。每个变量的取值非0即1。

- 0、1不表示数的大小，而是代表两种不同的逻辑状态。
- 如果有若干个逻辑变量（如A,B,C,D）按与、或、非三种基本运算组合在一起，得到一个表达式L。
- 对逻辑变量的任意一组取值（如0000、0001、0010）来说，L有唯一的值与之对应，则称L为逻辑函数。
- 由逻辑变量A、B、C、D所表示的逻辑函数记为：

$$L=f(A,B,C,D)$$

逻辑代数理论
--逻辑代数中运算关系

两个主要的二元运算的符号定义为 (逻辑与)和 (逻辑或)，把单一的一元运算的符号定义为 (逻辑非)。我们还使用值 0 (逻辑假)和 1 (逻辑真)。逻辑代数有下列性质:

- 结合律

$$a \vee (b \vee c) = (a \vee b) \vee c \qquad a \wedge (b \wedge c) = (a \wedge b) \wedge c$$

- 交换律

$$a \vee b = b \vee a \qquad a \wedge b = b \wedge a$$

- 吸收律

$$a \vee (a \wedge b) = a \qquad a \wedge (a \vee b) = a$$

逻辑代数理论
--逻辑代数中运算关系

- **分配律**

$$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$$

$$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$$

- **互补律**

$$a \vee \neg a = 1 \qquad a \wedge \neg a = 0$$

- **幂等律**

$$a \vee a = a \qquad a \wedge a = a$$

逻辑代数理论
--逻辑代数中运算关系

- **有界律**

$$a \vee 0 = a \qquad a \wedge 1 = a$$

$$a \vee 1 = 1 \qquad a \wedge 0 = 0$$

- **德.摩根定律**

$$\neg(a \vee b) = \neg a \wedge \neg b$$
$$\neg(a \wedge b) = \neg a \vee \neg b$$

- **对合律**

$$\neg\neg a = a$$

逻辑代数理论
--代数的基本规则

代入规则

任何一个含有变量 X 的等式，如果将所有出现 X 的位置，都用一个逻辑函数 F进行替换，此等式仍然成立。例如，等式：

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

将所有出现A的地方，用函数E+F代替，则等式仍然成立：

$$B\cdot [(E+F)+C]=B\cdot (E+F)+B\cdot C=B\cdot E+B\cdot F+B\cdot C$$

逻辑代数理论
--代数的基本规则

对偶规则

设 F 是一个逻辑函数式，如果将 F 中的所有的·变成 +，+ 变成 ·，0 变成 1，1 变成 0，而变量保持不变。那么就的得到了一个逻辑函数式 F'，这个 F' 就称为 F 的对偶式。如果两个逻辑函数 F 和 G 相等，则它们各自的对偶式 F' 和 G' 也相等。例如：

$$F=A+B$$

对偶式 F'为：

$$F'= A \cdot B$$

逻辑代数理论
--代数的基本规则

吸收律：

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

成立。

则：

它们的对偶式：

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

也是成立的。

逻辑代数理论
--代数的基本规则

反演规则

当已知一个逻辑函数 F，要求时，只要把 F 中的所有·变成 +，+ 变成 ·，0 变成 1，1 变成 0，原变量变成反变量，反变量变成原变量，即得 $\overline{F}$。例如：

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

反演式为:

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

逻辑代数理论
--逻辑函数表达式

所有的逻辑函数表达式，不管逻辑关系多复杂，一定可以使用“与或”表达式或者“或与”表达式进行表示。

- **在逻辑函数表达式中，逻辑与关系用符号‘·’表示，逻辑或关系用符号‘+’表示。所以，与或表达式也称之为积之和（Sum of Product，SOP）表达式，或与表达式也称之为和之积（Product of Sum，POS）表达式。**

逻辑代数理论
--逻辑函数表达式

“与或”表达式

是指由若干“与项”进行“或”运算构成的表达式。每个“与项”可以是单个变量的原变量或者反变量，也可以由多个原变量或者反变量相“与”组成。例如：

AB、$\overline{A}BC$、$\overline{C}$均为“与项”。

“与项”又被称为“乘积项”。将这3个“与项”相“或”便可构成一个3变量函数的“与或”表达式，表示为：

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

逻辑代数理论
--逻辑函数表达式

真值表

A	B	C	Y	
0	0	0	0	
0	0	1	1	$\bar{A}\cdot\bar{B}\cdot C$
0	1	0	0	
0	1	1	1	$\bar{A}\cdot B\cdot C$
1	0	0	0	
1	0	1	1	$A\cdot\bar{B}\cdot C$
1	1	0	0	
1	1	1	0	

思考**:** 如何理解这张表的含义?

逻辑代数理论
--逻辑函数表达式

“或与”表达式

是指由若干“或项”进行“与”运算构成的表达式。每个“或项”可以是单个变量的原变量或者反变量，也可以由多个原变量或者反变量相“或”组成。例如：

A+B、(B+C)、$\mathrm{A}+\bar{B}+\mathrm{C}$、D

均为“或项”。将这4个“或项”相“与”便可构成一个4变量函数的“或与”表达式，表示为：

$$\mathrm{F}(\mathrm{A},\mathrm{B},\mathrm{C},D)=(\mathrm{A}+B)\cdot(\mathrm{B}+C)\cdot(\mathrm{A}+\bar{B}+C)\cdot\mathrm{D}$$

逻辑代数理论
--逻辑函数表达式

真值表

A	B	C	Y	
0	0	0	0	$A+B+C$
0	0	1	1	
0	1	0	0	$A+\overline{B}+C$
0	1	1	1	
1	0	0	0	$\overline{A}+B+C$
1	0	1	1	
1	1	0	0	$\overline{A}+\overline{B}+C$
1	1	1	0	$\overline{A}+\overline{B}+\overline{C}$

思考: 如何理解这张表的含义？

逻辑代数理论
--逻辑函数表达式

最小项和最大项

在三变量SOP表达式中，每个乘积项包含所有三个输入变量。同样，在POS表达式中，每个和项包含所有三个输入变量。

- 包含三个输入变量的乘积项称为最小项；
- 包含三个输入变量的和项称为最大项。

如果将一个给定行上的输入1或者0当作一个二进制数，则最大项或者最小项的数字可以分配到真值表中的每一行。

逻辑代数理论
--逻辑函数表达式

最小项和最大项表示

A	B	C	#	最小项	最大项	F
0	0	0	0	$\bar{A}\cdot\bar{B}\cdot\bar{C}$	$A+B+C$	0
0	0	1	1	$\bar{A}\cdot\bar{B}\cdot C$	$A+B+\bar{C}$	1
0	1	0	2	$\bar{A}\cdot B\cdot\bar{C}$	$A+\bar{B}\cdot\bar{C}$	0
0	1	1	3	$\bar{A}\cdot B\cdot C$	$A+\bar{B}+\bar{C}$	1
1	0	0	4	$A\cdot\bar{B}\cdot\bar{C}$	$\bar{A}+B+C$	0
1	0	1	5	$A\cdot\bar{B}\cdot C$	$\bar{A}+B+\bar{C}$	1
1	1	0	6	$A\cdot B\cdot\bar{C}$	$\bar{A}+\bar{B}+C$	0
1	1	1	7	$A\cdot B\cdot C$	$\bar{A}+\bar{B}+\bar{C}$	0

逻辑代数理论
--逻辑函数表达式

对最小项和最大项编码，这样将SOP和POS等式用简化方式表示。

- SOP等式使用符号$\sum$，表示乘积项求和；
- POS等式使用$\prod$，表示和项求积。
- 真值表内输出为1的一行定义了一个最小项，输出为0的一行定义了最大项。下面给出使用最小项和最大项简单表示方法。

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

$$F = \prod M(0,2,4,6,7)$$