

多值对称函数基于二值全加器的电路实现¹

陈借雄 沈继忠

(杭州大学电子工程系 杭州 310028)

摘要 本文讨论了多值对称函数的定义和性质,指出了多值对称函数可以按函数值 j 分解,而与 j 相应的子函数 L_j 必为对称函数,且可表示成蜕化多值基本对称函数乘积项之和的形式。在此基础上提出了多值对称函数基于二值全加器的逻辑综合。

关键词 多值逻辑, 对称函数, 逻辑设计

中图分类号 TN791

1 引言

对称函数是逻辑函数中一类有用的特殊函数。对称函数由于其独特的优点在国内引起了学术界的重视^[1-3]。关于二值对称函数的性质以及基本对称函数的逻辑综合可以在许多文献中找到^[4,5]。对于多值对称函数的研究也取得了进展,然而迄今为止关于多值逻辑的研究还很不充分,有待于进一步深入。本文在作者以前关于二值对称函数研究的基础上,将二值对称函数某些结果推广应用于多值对称函数,从而提出了多值对称函数的基于二值全加器的电路实现。

2 多值对称函数的定义和性质

定义1 设 $f(x_1, \dots, x_n)$ 为 n 变量的多值逻辑函数,如果交换任意一对变量时函数值均保持不变,即有 $f(x_1, \dots, x_i, \dots, x_j, \dots, x_n) = f(x_1, \dots, x_j, \dots, x_i, \dots, x_n)$, 式中 $f, x_i \in \{0, 1, \dots, r-1\}$, 则称函数 $f(x_1, \dots, x_n)$ 为多值对称函数。

定义2 设 $f(x_1, \dots, x_n)$ 为多值对称函数,当且仅当 n 变量中取值为 $0, 1, \dots, r-1$ 的变量数分别为 a_0, a_1, \dots, a_{r-1} 时函数值为 $r-1$ 。否则函数值为 0 , 则称函数 $f(x_1, \dots, x_n)$ 为多值基本对称函数。

定义3 设 $f(x_1, \dots, x_n)$ 为多值对称函数,当函数和输入变量均仅取 $0, r-1$ 二个值时,则称函数 $f(x_1, \dots, x_n)$ 为蜕化多值对称函数。

定义4 设 $f(x_1, \dots, x_n)$ 为多值基本对称函数,当该函数和输入变量均仅取 $0, r-1$ 二个值时,则称函数 $f(x_1, \dots, x_n)$ 为蜕化多值基本对称函数。

定义5 设有非空集合 $A = \{0, r-1\}$, 在 A 上定义了取小 (\wedge)、取大 (\vee) 和阈运算 (${}^i x^i$) 等三种运算, 则称 $\langle \{0, r-1\}, \wedge, \vee, {}^i x^i \rangle$ 为蜕化多值代数系统。

¹ 1994-07-29 收到, 1995-03-08 定稿
浙江省自然科学基金资助课题

容易证明, 蜕化多值代数系统 $\langle \{0, r-1\}, \wedge, \vee, 'x' \rangle$ 与布尔代数系统 $\langle \{0, 1\}, \bullet, +, \rightarrow \rangle$ 是同构的, 因此布尔代数中有关对称函数的公式、定理可推广应用于蜕化多值代数系统。现列举若干有关性质如下:

(1) 任意 n 变量的蜕化多值对称函数 $f(x_1, \dots, x_n)$ 可由蜕化多值基本对称函数之取大运算实现, 即有 $f(x_1, \dots, x_n) = \sum_{i=0}^n A_i \vee S_i$, 式中 $A_i \in \{0, r-1\}$, \sum 表示取大运算, S_i 为蜕化多值基本对称函数, 当 n 变量中 i 个变量为 $r-1$, 其余变量为 0 时, $S_i = r-1$; 否则 $S_i = 0$ 。

(2) $S_{a_1, \dots, a_p}(x_1, \dots, x_n) \wedge S_{b_1, \dots, b_q}(x_1, \dots, x_n) = S_{c_1, \dots, c_k}(x_1, \dots, x_n)$, 式中 $S_{a_1, a_2, \dots, a_p}(x_1, \dots, x_n)$ 表示该函数在 n 输入中有 a_1, a_2, \dots, a_p 个变量为 $r-1$ 时函数值为 $r-1$; 否则函数值为 0 。 $\{c_1, \dots, c_k\} = \{a_1, \dots, a_p\} \cap \{b_1, \dots, b_q\}$, 即集合 C 为集合 A 和集合 B 的交集。

(3) $S_{a_1, \dots, a_p}(x_1, \dots, x_n) \vee S_{b_1, \dots, b_q}(x_1, \dots, x_n) = S_{c_1, \dots, c_k}(x_1, \dots, x_n)$, 式中 $\{c_1, \dots, c_k\} = \{a_1, \dots, a_p\} \cup \{b_1, \dots, b_q\}$, 即集合 C 为集合 A 和集合 B 的并集。

(4) $\overline{S_{a_1, \dots, a_p}(x_1, \dots, x_n)} = S_{c_1, \dots, c_k}(x_1, \dots, x_n)$, 式中 $\{c_1, \dots, c_k\} = \{0, 1, \dots, n\} - \{a_1, \dots, a_p\}$, 即集合 C 为集合 A 的补集。

3 多值对称函数的综合

文献 [2] 指出, 任意多值对称函数均可表示为 $f(x_1, \dots, x_n) = \sum_{j=0}^{r-1} j \wedge L_j(x_1, \dots, x_n)$, 式中 L_j 为 $f(x_1, \dots, x_n)$ 的 j 次逻辑分量, 并可表示为 $L_j(x_1, \dots, x_n) = S_{\{a\}}^j \vee S_{\{b\}}^j \vee \dots \vee S_{\{p\}}^j$, 式中 $S_{\{a\}}^j, \dots, S_{\{p\}}^j$ 为蜕化多值对称函数, 且 $S_{\{a\}}^j = S_{a_1}^j({}^1x_1^1, \dots, {}^1x_n^1) \wedge S_{a_2}^j({}^2x_1^2, \dots, {}^2x_n^2) \dots \wedge S_{a_{r-1}}^j({}^{r-1}x_1^{r-1}, \dots, {}^{r-1}x_n^{r-1}) \dots S_{\{p\}}^j = S_{p_1}^j({}^1x_1^1, \dots, {}^1x_n^1) \wedge S_{p_2}^j({}^2x_1^2, \dots, {}^2x_n^2) \dots \wedge S_{p_{r-1}}^j({}^{r-1}x_1^{r-1}, \dots, {}^{r-1}x_n^{r-1})$, $S_{a_i}^j$ 表示当且仅当输入 n 变量中有 a_i 个变量取值为 i , 即 ${}^i x_1^i, \dots, {}^i x_n^i$ 中有 a_i 个取值为 $r-1$ 时函数 $S_{a_i}^j$ 为 $r-1$; 否则函数取值为 0 。

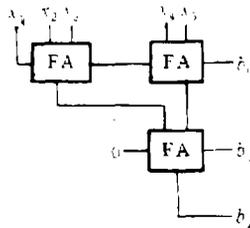


图 1 五变量全加网络

为了计算, ${}^i x_1^i, \dots, {}^i x_n^i$ 等 n 个阈运算中取值为 $r-1$ 的数目 (相应于 x_1, \dots, x_n 等 n 个输入变量中取值为 i 的变量数), 让我们先来分析一下二进制全加器的逻辑功能。在传统二值逻辑中, 根据二进制数全加规则可以将一位二进制数全加器的逻辑功能表示成 $A + B + C = C'S = b_1b_0$, 式中 “+” 为二进制加法运算, A, B 为本位二进制输入, C 为进位输入,

S 为本位和输出 (求和后所得二进制的低位), C' 为进位输出 (求和后所得二进制数的高位), $A, B, C, C', S \in \{0, 1\}$ 。文献 [4] 指出, 由一位二进制全加器组成的全加网络可用于计算输入 n 变量中取值为 1 的变量数, 而全加网络可用代数分解予以设计。例如, 对于五变量全加网可分解如下:

$$\begin{aligned} x_1 + x_2 + x_3 + x_4 + x_5 &= (x_1 + x_2 + x_3) + x_4 + x_5 \\ &= C'_1 S_1 + x_4 + x_5 = C'_1 0 + (S_1 + x_4 + x_5) = C'_1 0 + C'_2 S_2 \\ &= (C'_1 + C'_2) S_2 = C'_3 S_3 S_2. \end{aligned}$$

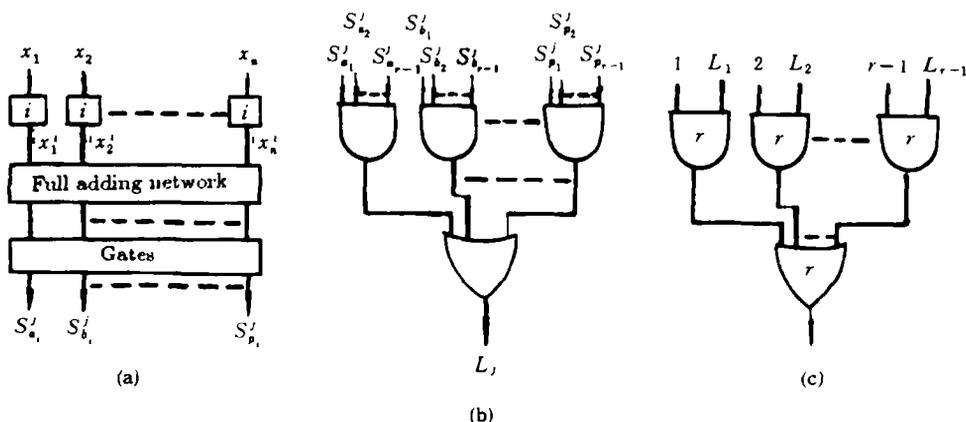


图 2 多值对称函数实现框图

与上式相应的电路实现如图 1 所示。注意到布尔代数系统中与运算 $A \cdot B$ 可看成是对变量 A, B 的取小运算, 即有 $A \cdot B = A \wedge B$. 类似地有, $A + B = A \vee B, \bar{A} = {}^0A^0$. 上述诸式中 $A, B, A \cdot B, A + B, \bar{A} \in \{0, 1\}$. 不难发现, 只要取输入、输出高电平相应于逻辑值 $r-1$, 则二值“与”、“或”、“非”门可以分别实现蜕化多值逻辑中的 $\wedge, \vee, {}^0x^0$ 运算。例如, 对于 TTL 电路而言, 我们可取电源电压值相应于逻辑值 $r-1$, 并在其输出端与电源之间跨接电阻 R 就可保证输出高电平相应于逻辑值 $r-1$ 。这样一来, 只要门电路的输入仅取逻辑值 $0, r-1$, 经改造后的 TTL “与”、“或”、“非”门就可分别实现蜕化多值系统中的 \wedge, \vee 及 ${}^0x^0$ 运算。类似地, 布尔代数系统中的二进制全加网络也可用于计算蜕化多值代数系统中输入变量中取值为 $r-1$ 的变量数。

在上述讨论的基础上, 我们可以提出任意多值对称函数基于全加器的电路实现框图, 如图 2(a)~2(c) 所示。在图 2(a) 中, n 个输入量先经过幅度鉴别, 这样 n 变量中取值为 i 的变量数归结为, ${}^i x_1^i, \dots, {}^i x_n^i$ 中取值为 $r-1$ 的数目。后者可以用二值全加器组成的全加网络予以计算, 在全加网络的输出可以得到它的二进制表示 $b_k \dots b_0$ 。然后将 $b_k \dots b_0$ 作为变量, 根据蜕化多值对称函数的脚标用二值门电路得到蜕化多值基本对称函数 $S'_{a_i}, S'_{b_i}, \dots, S'_{p_i}$ 。图 2(b) 给出了用二值“与”门和“或”门实现 L_j 的电路。图 2(c) 示出了用 r 值取小门和 r 值取大门实现所需的任意多值对称函数 $f(x_1, \dots, x_n)$ 。由上述诸图可见, 除了在输入部分使用了多值阈门和在输出部分使用多值门电路外, 其余均采用二值电路。

例 1 试实现三值对称函数 $f(x_1, \dots, x_5) = [1 \wedge (s_{212}^1 \vee s_{113}^1)] \vee [2 \wedge S_{023}^2]$. 由 (6) 式可得 $S_{212}^1 = S_1^1({}^1x_1^1, \dots, {}^1x_5^1) \wedge S_2^1({}^2x_1^2, \dots, {}^2x_5^2), S_{113}^1 = S_1^1({}^1x_1^1, \dots, {}^1x_5^1) \wedge S_3^1({}^2x_1^2, \dots, {}^2x_5^2), S_{023}^2 = S_2^2({}^1x_1^1, \dots, {}^1x_5^1) \wedge S_3^2({}^2x_1^2, \dots, {}^2x_5^2)$.

故有

$$\begin{aligned} f(x_1, \dots, x_5) &= 1 \wedge \{ [S_1^1({}^1x_1^1, \dots, {}^1x_5^1) \wedge S_2^1({}^2x_1^2, \dots, {}^2x_5^2)] \vee [S_1^1({}^1x_1^1, \dots, {}^1x_5^1) \\ &\quad \wedge S_3^1({}^2x_1^2, \dots, {}^2x_5^2)] \} \vee [2 \wedge S_2^2({}^1x_1^1, \dots, {}^1x_5^1) \wedge S_3^2({}^2x_1^2, \dots, {}^2x_5^2)] \\ &= 1 \wedge \{ S_1^1({}^1x_1^1, \dots, {}^1x_5^1) \wedge [S_2^1({}^2x_1^2, \dots, {}^2x_5^2) \vee S_3^1({}^2x_1^2, \dots, {}^2x_5^2)] \} \\ &\quad \vee [S_2^2({}^1x_1^1, \dots, {}^1x_5^1) \wedge S_3^2({}^2x_1^2, \dots, {}^2x_5^2)]. \end{aligned}$$

由于蜕化多值代数系统中，取小、取大运算可分别用二值“与”门和“或”门实现，因此上式可改写成

$$f(x_1, \dots, x_5) = 1 \wedge \{S_1^1(^1x_1, \dots, ^1x_5) \cdot [S_2^1(^2x_1, \dots, ^2x_5) + S_3^1(^2x_1, \dots, ^2x_5)]\} \\ \vee [S_2^2(^1x_1, \dots, ^1x_5) \cdot S_3^2(^2x_1, \dots, ^2x_5)].$$

根据上式中 S 的脚标以及五变量的情况下不可能出现六个和六个以上的 $^1x_i, ^2x_i$ 同时为 $r-1$ ，可以得到它们的 K 图表示，如图 3 所示。由图 3 可得

$$S_1^1 = \bar{b}_2^1 \cdot \bar{b}_1^1 \cdot b_0^1, \quad S_2^1 + S_3^1 = b_2^1, \quad S_2^2 = b_1^1 \cdot \bar{b}_0^1, \quad S_3^2 = b_1^1 \cdot b_0^1.$$

故有

$$f = (1 \wedge \bar{b}_2^1 \bar{b}_1^1 b_0^1 b_2^2) \vee b_1^1 \bar{b}_0^1 b_1^2 b_0^2$$

与上式相应的电路如图 4 所示。

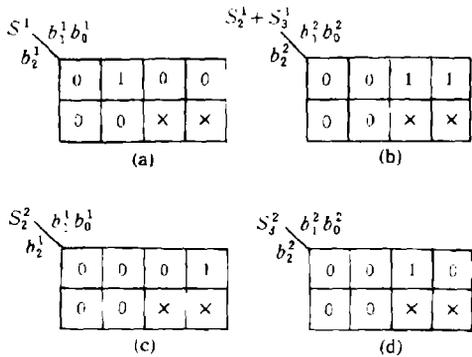


图 3 $S_1^1, S_2^1 + S_3^1, S_2^2, S_3^2$ 的 K 图表示

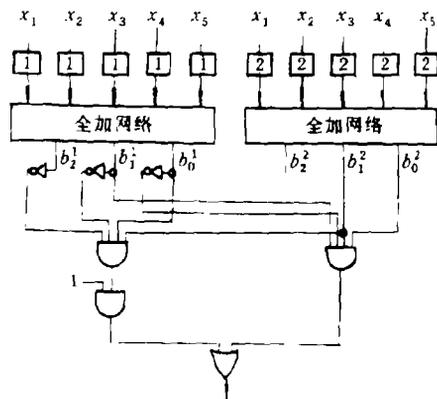


图 4 例 1 函数的电路实现

4 结 论

由于蜕化多值代数系统与布尔代数系统是同构的，因此布尔代数中的有关性质、定理可以推广用于蜕化多值代数系统。文中指出蜕化多值代数系统中的取小、取大、阈运算和全加网络可分别地用“与”门、“或”门、反相器和全加器予以实现。在此基础上，我们根据多值对称函数的代数表示提出了多值对称函数基于全加器的电路实现。最后应该指出，由于一位全加器还是一个二变量通用逻辑门^[4]，因此图 2(a)，2(b) 中的二值电路均可用一位全加器实现。

参 考 文 献

[1] Chen X(陈偕雄). The Radio and Electronic Engineer, 1983, 53(2): 67-74.
 [2] Tapia Ma. Int. J. Electronics, 1989, 67(5): 703-715.
 [3] Butler J T, Schueller K A. Worst case number of terms in symmetric multivalued functions, IEEE proc. 21th ISMVL, Victoria: 1991, 94-101.
 [4] 陈偕雄. 科技通报. 1990, 6(1) : 1-5.

[5] 赵小杰, 陈借雄. 杭州大学学报, 1990, 17(4): 401-408.

THE LOGIC SYNTHESIS OF MULTIVALUED SYMMETRIC FUNCTION BASED ON BINARY FULL-ADDER

Chen Xiexiong Shen Jizhong

(*Hangzhou University, Hangzhou 310028*)

Abstract This paper discusses the definition and properties of multivalued symmetric functions, and points out that a multivalued symmetric function can be decomposed according to the value of the function j . The subfunction L_j corresponding to j will certainly be a symmetric function, and it may be expressed as the sum-of-products form of degenerated multivalued fundamental symmetric functions. Based on this consideration, the logic synthesis circuit realization for the multivalued symmetric functions based upon full-adders is proposed.

Key words Multivalued logic, Symmetric functions, Logic design

陈借雄: 男, 1941年生, 教授, 中国电子学会会士, 从事数字电子学专业.

沈继忠: 男, 1965年生, 副教授, 中国电子学会高级会员, 从事数字电子学专业.