计算机科学2004Vol. 31№. 12

逻辑综合中的Cube运算

刘丹非

（云南师范大学计算机科学系　昆明650092）

摘　要　Cube运算是EDA中进行逻辑综合(Logic Synthesis)的重要方法之一，Cube运算的实质仍然是卡诺言图化简。卡诺图是一个二维的平面图，Cube运算是建立在多维的空间坐标体系上的计算模型。本文从真值表和卡诺图出发，研究Cube运算的基本方法，从而得出多维变量的数学模型和计算方法。

关键词　卡诺图，Cube运算，逻辑综合

The Cube Calculation in the Logic Synthesis

LIU Dan-Fei

(Department of Computer Science, Yunnan Normal University, Kunming 650092)

Abstract In the EDA, the Cube calculation is regarded as an important method which proceeds logic's Logic Synthesis. In substance, the Cube calculation is still Karnaugh. The Karnaugh is a plane chart for two dimensions. The Cube calculation is a computing model based on multi-dimensions coordinate system. From the truth table and Karnaugh, this paper studies the method of Cube calculation in the Logic Synthesis.

Keywords Cube calculation, Logic synthesis

在数字逻辑电路中，当变量的输入、输出关系确定后，可以用公式法，卡诺图法等进行逻辑化简，但是当变量数多于5后，上述方法也变得越来越难，同时随着EDA(Electronic Desing Automation)的发展，多变量的计算和化简方法，尤其是计算机方法的算法模型显得日益重要。Cube运算就是EDA中进行逻辑综合(Logic Synthesis)的重要方法之一。Cube运算是经典逻辑运算的多维演绎，本文从真值表和卡诺图出发，研究Cube运算的基本方法，从而得出多维变量的数学模型和计算方法。

1　逻辑函数的Cube表示法与K维立方体

若逻辑函数$y=f(x_1,x_2,x_3,\cdots,x_n)$。令每一输入变量对应于正交体系的一个坐标轴，则可得函数y的空间图形。

若$n=3$则$y=f(x_1,x_2,x_3)=\sum_m(0,1,2,3,4,5,6,7)$，其空间图形为三维坐标体系的Cube。其卡诺图为：

x1 \ x2x3	00	01	11	10
0	m_0	m_1	m_3	m_2
1	m_4	m_5	m_7	m_6

图1　n=3的卡诺图

其卡诺图与三维坐标轴之间的关系是卡诺图中的一个小方格对应立方体中的一个顶点，取值为0最小项在立方体的顶点用空心圈表示，取值为1最小项在立方体的顶点用实心圈表示，取值为任意项在立方体的顶点用星号表示。

在$y=f(x_1,x_2,x_3,\cdots,x_n)$中，若定义“$X$”为输入变量在对应的乘积项中不出现的变量或相当于卡诺图中相邻最小项可以消去的变量“X”，用“X”的个数表示并定义0个“X”，称为0维立方体，1个“X”，称为1维立方体，2个“X”，称为2维立方体，……K个“X”，称为K维立方体，并分别表示为：C_j^0 C_j^1 C_j^2 $C_j^3\cdots C_j^k$，其中j为最小项中“1”的个数，K为维数。且$0\leqslant K\leqslant n$

显然　$y=\sum_{j=0} C_j^0$为最小项之和

其中$C_j^k=b_{n-1}b_{n-2}\cdots b_j\cdots b_2b_1b_0$　($b\subset 0,1,x$)

例如：$n=3$，有$C_1^0=001$或者010或100；

$C_1^1=0X1$或者$X01$或$X10$……；

$C_1^2=XX1$或$X1X$；

$C_2^1=X11$或$1X1$或$11X$。

由逻辑函数的一致性原理：$AB+A\overline{B}=A=AX$，

则有：$C_i^0+C_j^0=C_i^1$。对应卡诺图即两个相邻最小项可以消去一个变量，在Cube运算中即两个“0”维立方体合为一个“1”维立方体。

例如$n=4$，有$C_2^2=X1X1$和$C_1^2=X0X1$则$C_2^2+C_1^2=C_1^3=XXX1$。

2　Quine-McCluskey算法

由于用卡诺图化简时，是靠观察的方法确定能够合并的最小项，而用计算机化简时必须找到一种能用计算机能接受的方法，Quine-McCluskey算法提出了这种算法的两个步骤：即：(1)决定函数的素隐含项；(2)找出函数的最小覆盖。以下用立方体的概念说明实现这两个步骤的方法。

2.1　决定素隐含项

素隐含项是指：一个素隐含项与函数的任何其他素隐含项不可能相距单位距离，或者说一个素隐含项不可能与另一个素隐含项结合起来消去一个变量。

例如图1中(0,0,1)与(1,1,1)是素隐含项，因为它们的$d=1.4142$，而(0,0,1)与(0,0,0)不是素隐含项，因为它们的$d=1$。

所谓决定素隐含项就是先确定函数中的“0”维立方体和“1”维立方体，然后再确定“2”维立方体、“3”维立方体……，若用S^k表示K维立方体的集合，则$S_j^k\subset S^k$其中S_j^k是S^k的子集，j为“1”的个数)在S^k中找出或检验K维立方体与相邻子集的距离d，若$d=1$，则此k维立方体可形成一新的$k+1$维的立方体C^{k+1}。

例如：$f(a,b,c,d)=\sum_{m}(0,1,2,5,7,8,10,13,15)$

ab \ cd	00	01	11	10
00	1	1		1
01		1	1	
11		1	1	
10	1			1

图2　$n=4$卡诺图

按上述规则：

(1)S^0中有：$S_0^0=m_0=(0000)$；$S_1^0=(m_1,m_2,m_8)=(0001,0010,1000)$；$S_2^0=(m_5,m_{10})=(0101,1010)$；$S_3^0=(m_7,m_{13})=(0111,1110)$；$S_4^0=(m_{15})=(1111)$

(2)求 S^1。由 S_0^0,S_1^0可得：

$S_0^1=(m_0,m_1)(m_0,m_2)(m_0,m_8)=(000X,00X0,X000)$

由 S_1^0,S_2^0可得：

$S_1^1=(m_1,m_5)(m_{10},m_2)(m_{10},m_8)$

$=(0X01,X010,10X0)$

$S_2^1=(m_5,m_7)(m_5,m_{13})$

$=(01X1,00X0,X101)$

$S_3^1=(m_7,m_{15})(m_{13},m_{15})=(X111,11X1)$

(3)求 S^2。由 S_0^1,S_1^1可得：

$S_0^2=(X000,X010)(00X0,10X0)$

$=(X0X0,X0X0)$

$S_1^2=(X111,X101)(11X1,01X1)$

$=(X1X1,X1X1)$

(4)求 S^3不存在。

2.2　找出其中不能再合并的项

即：$000X$，$0X01$，$X0X0$，$X1X1$是素隐含项，从卡诺言图化简可得到$0X01$是冗余项，在 Q-M 算法里还得用最小覆盖去掉冗余项。

2.3　最小覆盖

一个函数的覆盖由包含此函数的所有“0”维立方体的素含项组成，当覆盖不包含另一个覆盖时，此覆盖为最小的。为了确定最小覆盖在 Q-M 算法中需要进行：识别 EPI(素隐含项)，更新集合和除去冗作项。

为简化说明此方法，依照卡诺图中在相邻最小项合并乘积项圈画的原则：在圈的最小项中至少可以找到一项未被圈过一次的最小项，这样可以将函数所包含的最小项和已确定素隐含项组成最小项——素隐含项表，在表中将素隐含项所包括的最小项的位置上画“○”，然后把纵列上只有一个“○”换成“◎”，则“◎”所在行对应的素隐含项不是冗余项。在余下素隐含项中若除去此行，余下的素隐含项存在某纵列只有一个“○”则除去的素隐含项是冗余项。

按照此方法，上述例子中，显然$0X01$或者$000X$是冗余项。

由图3最小覆盖的素隐含项仅为$000X$，$X0X0$，$X1X1$或者$0X01$，$X0X0$，$X1X1$，化简结果为：$y=f(a,b,c,d)=\overline{ABC}+\overline{B}\,\overline{D}+BD$，与卡诺图化简的结果完全相同。

m	m_0	m_1	m_2	m_5	m_7	m_8	m_{10}	m_{13}	m_{15}
000x	○	○							
0x01		○		○					
x0x0	○		◎			○	○		
x1x1				○	○			◎	○

图3　最小覆盖化简

小结　在决定函数的素隐含项时，找出所有最小项，采用逐项逐位比较的办法即可找出素隐含项，这是计算机程序设计善长之处。

在找出函数的最小覆盖时，用矩阵的办法并不难找出冗余项。

Cube 运算的实质仍然是卡诺图化简，但卡诺图是一个二维的平面图，Cube 运算将其扩展为多维的空间坐标体系；卡诺图化简时简单直观，但变量不能太多，Cube 运算繁锁抽象，适合计算机编程实现。可见 Cube 运算是 EDA 中对电路综合的行之有效的化简方法。

参考文献

1　李亚民．计算机组成与系统结构．北京：清华大学出版社，2000

2　孟宪元，李广军．可编程 ASIC 设计与应用．成都：电子科技大学出版社，2000

3　江国强．现代数字逻辑电路．北京：电子工业出版社，2002

4　薛宏熙，边计年．数字系统设计自动化．北京：清华大学出版社，2002

(上接第206页)

I/O 端口和系统中断控制器的输入(IRQ)。获取这些参数的方法由硬件的接口方式决定。PCI 总线作为一种即插即用的总线结构，在 BOOTROM 和操作系统的支持下，能够自动为设备分配合适的硬件接口参数。

硬件的行为和特性是由内部的寄存器控制的。基于 PCI 总线的系统采用内存映射来访问寄存器。

对于采用中断方式的硬件设备，在接口函数中必须实现中断服务程序。中断程序的编写必须遵循一条规则：不能有运行时间过长的代码；不能独占共享资源以避免死锁；程序结束后尽可能地快速返回。

上述步骤完成后，即可启动硬件设备。

结束语　考虑到作为下一代存储技术的体全息存储的存储容量大，并行速度高，又兼顾刚开始价格较高，我们将其定位于高速网络数据存储服务器应用，所以面向千兆以太网来设计这个的体全息存储数据通道。

我们在利用网络处理器 IXP1200的传统网络处理的同时，基于嵌入式实时系统 VxWorks 充分挖掘了其 StrongARM 核的强大潜能，使得体全息存储与千兆以太网融为一体，开辟了一条通向新型体全息存储体的高速数据通道。经过测试，我们所设计的体全息高速数据通道的有效数据传输速率达100MB/s，完全满足当初的设计要求。

参考文献

1　Psaltis D, Burr GW. Holographic Data Storage. Computer, 1998, 31(2): 52～60

2　Kevin C, William W, Lisa D. Commercialization of Holographic Storage at InPhase Technologies. IEEE Invited Paper, 2002

3　Wu Fei, XIE ChangSheng, Hu DiQing, Wu Ming. The Design and Research of High-Speed Channel for Volume Holographic Data Storage. APOC 04-157, 2003

4　Ju C, Dai F, Hong J. Electron. Lett., 1996, 32(15): 1400

5　Intel Corp. IXP1200 Evaluation System User's Manual, 2001