

基于忆阻器的逻辑门和二位吠陀乘法器设计

刘海峰¹, 方 芳², 王 伟¹, 王 彬¹

(1 合肥工业大学 计算机与信息学院, 安徽 合肥 230009;

2 合肥工业大学 管理学院, 安徽 合肥 230009)

摘 要: 忆阻器是一种新型的无源元件, 它拥有纳米级的尺寸、非线性以及记忆功能等特点, 在模型分析、基础电路设计、电子元件设计、集成电路设计、神经突触等方面得到了运用. 该文设计了一种混合 CMOS 忆阻器逻辑门电路, 使用基于忆阻器的与门和或门、基于 CMOS 工艺的反相器, 混合实现了基于忆阻器的异或门电路, 进而组合已实现的逻辑门电路完成二位吠陀乘法器的设计. 实验结果表明, 设计的二位吠陀乘法器, 相较于已有的二位吠陀乘法器, 在完成相同功能时, 功耗约为原来的 80% 和面积开销为原来的 76%.

关键词: 忆阻器; 无源元件; CMOS; 逻辑门电路; 吠陀乘法器

中图分类号: TP34

文献标识码: A

文章编号: 1000-7180(2018)10-0046-07

Design of the Logic Gate and Two-bit Vedic Multipliers Based on Memristors

LIU hai-feng¹, FANG Fang², WANG Wei¹, WANG Bin¹

(1 School of Computer and Information, Hefei University of Technology, Hefei 230009, China;

2 School of Management, Hefei University of of Technology, Hefei 230009, China)

Abstract: Memristor is a new type of passive components, it has the characteristics of nanometer size, non-linearity and memory function. And it has been applied in model analysis, basic circuit design, electronic component design, integrated circuit design, synaptic and other aspects. In this paper, a hybrid CMOS memristor logic gate is designed, which uses an AND gate and OR gate based on memristor and an inverter based on CMOS process to realize an exclusive OR gate circuit based on memristor. Then, the two-bit Vedic multipliers are implemented using the memristor logic gate designed in this paper. The experimental results show that the two-bit Vedic multipliers designed in this paper compared with the existing two-bit Vedic multipliers, consumes approximately 80% of the original power and 76% of the area overhead when performing the same function.

Key words: memristor; passive components; CMOS; logic gate circuit; vedic multiplier

1 引言

1971 年华裔科学家蔡少堂教授首次提出忆阻器的概念, 推断在电阻、电容、电感之外, 他还有一种元件来表示电荷和磁通量的关系. 他根据这种新元件的特性称之为记忆电阻器, 简称忆阻器^[1]. 2008 年, 惠普 (HP; Hewlett-Packard) 实验室研究团队成

功做出纳米忆阻器件, 证实了蔡少堂教授的推断^[2], 并且进一步研究发现, 忆阻器的纳米级尺寸有助于摩尔定律的延续, 非线性、非易失性对计算机存储和生物神经突触的研究有重要的意义^[3].

忆阻器的记忆特性可以用来制造新非易失性随机存储器^[4], 与如今的动态随机存储器相比, 可以大大加快计算机的开机时间. 忆阻器的另外一个特性

收稿日期: 2018-01-20; **修回日期:** 2018-02-01

基金项目: 国家自然科学基金重点项目 (61432004); 国家自然科学基金 (61474035, 61204046, 61306049); 安徽省科技攻关项目 (1206c0805039); 安徽省自然科学基金 (1508085QF129); 教育部新教师基金 (20130111120030)

不仅可以表示逻辑“0”和“1”的状态,还能表示“0”和“1”之间的状态,这对计算机的开发有着巨大的意义. 本文就是研究忆阻器可表示逻辑“0”和“1”来实现逻辑门电路. 由于目前的逻辑门电路普遍由COMS制造的,而忆阻器的特性必定会让其在未来电路上发挥着重要作用.

许多研究工作已经利用忆阻器实现与门和或门^[5],但是不能仅用忆阻器去实现非门,对于实现异或门也如此. 然而异或门在逻辑电路中有着很重要的作用,比如加法器、乘法器等. 本文设计了一种混合 CMOS 忆阻器逻辑门电路,搭建了新的二位二进制吠陀乘法器.

2 问题分析

2.1 忆阻器工作原理

目前,忆阻器的模型普遍是以 HP 实验室的模型为基础,下面就以该模型对忆阻器进行介绍.

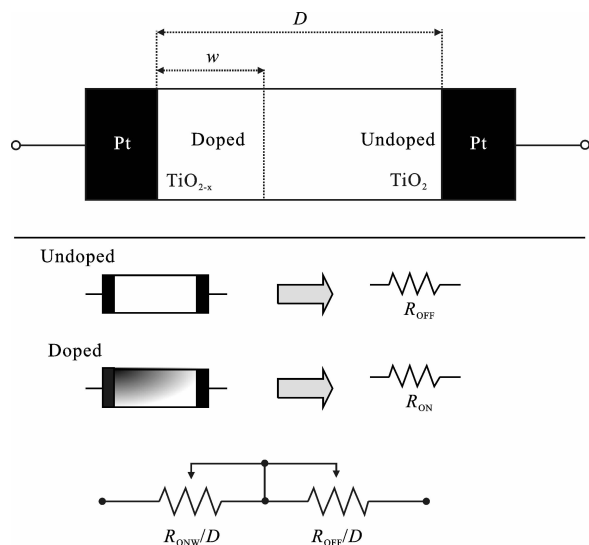


图1 HP 实验室忆阻器物理模型

HP 的忆阻器原型器件是二端为金属电极中间为氧化物的器件,在两端铂电极之间置入氧化物掺杂薄膜,其中的薄膜可分为两个区域:一个区域仅有二氧化钛(TiO_2)不含有掺杂物,导电性较弱,因此电阻较高,属于未掺杂区,另一个区域加入了正 2 价氧空穴形成的 TiO_{2-x} 掺杂物,导电性较强,因此电阻较低,属于掺杂区. 当有电流通过忆阻器件时,因电流方向的不同,掺杂区扩张或缩小,即掺杂区和非掺杂区的长度发生变化,但是因为两个区域的电导率不同,使得忆阻器的电阻值发生变化^[6]. 如图 1 所示. 设薄膜长度为 D ,掺杂区的长度为 w ,当掺杂区占据整个薄膜时,也就是 $w = D$,此时电阻很小为

R_{ON} ,而非掺杂区占据整个薄膜时,也就是 $w = 0$,此时电阻很大为 R_{OFF} . 其中 $R_{\text{ON}} \ll R_{\text{OFF}}$. 忆阻器的阻值表示为:

$$R_M(w) = R_{\text{ON}} \frac{w}{D} + R_{\text{OFF}} \left(1 - \frac{w}{D}\right) \quad (1)$$

假设在忆阻器外加偏置电压 $v(t)$,由欧姆定律可得:

$$v(t) = \left(R_{\text{ON}} \frac{w(t)}{D} + R_{\text{OFF}} \left(1 - \frac{w(t)}{D}\right)\right) i(t) \quad (2)$$

$$\frac{dw(t)}{dt} = \mu_V \frac{R_{\text{ON}}}{D} i(t) \quad (3)$$

式中, $i(t)$ 为在 t 时间流过忆阻器的电流; $w(t)$ 为在 t 时间掺杂区的长度; μ_V 是忆阻器中离子平均迁移速度. 对式(3)求积分:

$$w(t) = \mu_V \frac{R_{\text{ON}}}{D} q(t), w(t) \in [0, D] \quad (4)$$

$q(t)$ 为 t 时间内流经忆阻器的总电荷量. 由于 $R_{\text{ON}} \ll R_{\text{OFF}}$,忆阻器阻值 $M(q)$ 可得:

$$M(q) = R_{\text{OFF}} \left(1 - \frac{\mu_V R_{\text{ON}}}{D^2} q(t)\right) \quad (5)$$

以上可知,当掺杂区的长度 $w(t) = 0$ 时, $M(q) = R_{\text{OFF}}$,由于阻值很高,为高阻态,可认为阻值为无限大;掺杂区的长度 $w(t) = D$ 时, $M(q) = R_{\text{ON}}$,但是阻值很低,为低阻态,可认为阻值为 0. 这是忆阻器表示逻辑状态“0”和“1”的理论基础^[7-9].

图 2 所示为忆阻器的装置符号,其中黑色区域为掺杂区,施加正向电压,图中的黑色区域会扩大,电阻减小;施加反向电压,白色区域会扩大,电阻增大^[10-13].

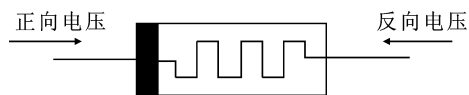


图2 忆阻器装置符号

2.2 基本的逻辑门

传统 MOS 管制造的逻辑门,其实就是利用 MOS 管的导通阻值几乎为零,截断阻值几乎为无穷大的特点实现基本的逻辑门的. 上一节可知,忆阻器的两种阻态低阻态和高阻态正好对应 MOS 管的导通截止两种状态. 当忆阻器为低阻态时,由于忆阻器电阻值极小,可认为是 MOS 管的导通状态;相反忆阻器为高阻态时,由于忆阻器电阻值极大,可认为是 MOS 管的截止状态. 根据这个特性,许多研究者仅用忆阻器设计与门和或门,如图 3 所示.

由图 3(a) 所示,对 A, B 两端施加不同的电压

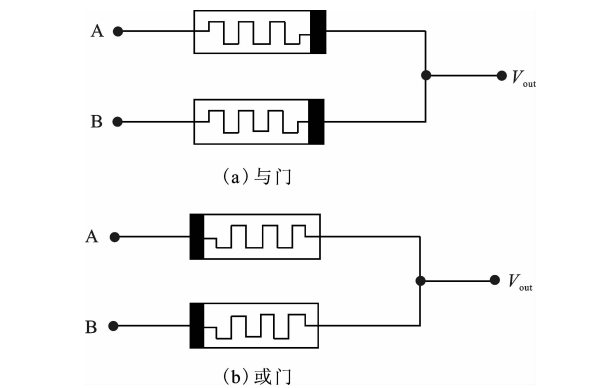


图 3 与门和或门

V_L, V_H , 得出 V_{out} 端的电压, 可分为四种情况. 其中设 V_L 表示逻辑“0”, V_H 表示逻辑“1”, r_1, r_2 表示忆阻器 M_1, M_2 的阻值.

- (1) A 端为 V_L , B 端也为 V_L , 无论 r_1, r_2 的阻值是多少, V_{out} 电压都是低电压, 即 $V_{out} = V_L$, 表示逻辑“0”;
- (2) A 端为 V_L , B 端为 V_H , 此时 M_1 施加的是正向电压, M_2 为反向电压, 在电压充分作用后, $r_1 = R_{ON}, r_2 = R_{OFF}$, 此时 $V_{out} \approx V_L$, 表示逻辑“0”;
- (3) A 端为 V_H , B 端为 V_L , 此时 M_1 施加的是反向电压, M_2 为正向电压, 在电压充分作用后, $r_1 = R_{OFF}, r_2 = R_{ON}$, 此时 $V_{out} \approx V_L$, 表示逻辑“0”;
- (4) A 端为 V_H , B 端也为 V_H , 无论 r_1, r_2 的阻值是多少, V_{out} 电压都是高电压, 即 $V_{out} = V_H$, 表示逻辑“1”.

以上分析, 可得逻辑与门的真值表, 如表 1 所示.

表 1 忆阻器实现逻辑与门真值表		
输入		输出
A	B	V_{out}
0	0	0
0	1	0
1	0	0
1	1	1

由图 3(b) 所示, 将忆阻器两端对调相接, 即是逻辑或门:

- (1) A 端为 V_L , B 端也为 V_L , 无论 r_1, r_2 的阻值是多少, V_{out} 压都是低电压, 即 $V_{out} = V_L$, 表示逻辑“0”;
- (2) A 端为 V_L , B 端为 V_H , 此时 M_1 施加的是反向电压, M_2 为正向电压, 在电压充分作用后, $r_1 = R_{OFF}, r_2 = R_{ON}$, 此时 $V_{out} \approx V_H$, 表示逻辑“1”;

- (3) A 端为 V_H , B 端为 V_L , 此时 M_1 施加的是正向电压, M_2 为反向电压, 在电压充分作用后, $r_1 = R_{ON}, r_2 = R_{OFF}$, 此时 $V_{out} \approx V_H$, 表示逻辑“1”;
- (4) A 端为 V_H , B 端也为 V_H , 无论 r_1, r_2 的阻值是多少, V_{out} 电压都是高电压, 即 $V_{out} = V_H$, 表示逻辑“1”.

以上分析, 可得逻辑与门的真值表, 如表 2 所示.

表 2 忆阻器实现逻辑或门真值表		
输入		输出
A	B	V_{out}
0	0	0
0	1	1
1	0	1
1	1	1

2.3 逻辑异或门的实现

从上一节可以看出, 忆阻器实现的逻辑门是通过外加偏置电压, 使得忆阻器本身状态改变, 达到改变 V_{out} 端的电压状态. 忆阻器的这种特性要出现电压差才会改变自身的状态, 而逻辑非门是一个输入端, 一个输出端, 这种情况使得仅用忆阻器很难实现逻辑非门. 非门的实现对于异或门的实现有很大的意义. 为了便于实现异或门, 这里引入了 CMOS 反相器. CMOS 反相器的工作的特性和异或运算的特性很相似. 异或运算的真值表反应了其运算特点如表 3 所示^[14-16].

表 3 异或门真值表		
输入		输出
A	B	V_{out}
0	0	0
0	1	1
1	0	1
1	1	0

异或运算是当两个输入端不同时, 输出端才会输出逻辑“1”. 这个特点是异或门实现的核心. 根据 CMOS 反相器的工作特性, 将与门和或门与 CMOS 反相器组合起来, 实现异或门的功能.

3 方法与设计

3.1 异或门的设计实现

从前面的内容可知, 异或门在两个输入端的值不一样的情况下, 输出端的值是逻辑“1”; 输入端的值一样的情况下, 输出端的值是逻辑“0”. 这相当于

一种选择器,在输入端值不一样的时候,选择输出逻辑“1”,一样的时候选择输出逻辑“0”.在这里要介绍一下本文使用的 CMOS 反向器,如图 4 所示.

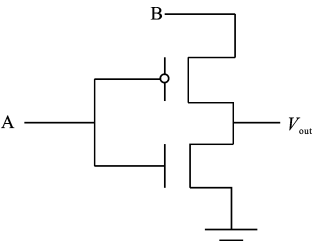


图 4 CMOS 反向器

- (1)A 端为低电压,B 端为低电压,反向器截止, V_{out} 端电压为低电压;
- (2)A 端为低电压,B 端为高电压,反向器导通, V_{out} 端电压为高电压;
- (3)A 端为高电压,B 端为高电压,反向器截止, V_{out} 端电压为低电压.

下面给出输入端不同情况下,与门和或门的输出端情况.

- (1)输入端为“0”,“0”,与门输出端是“0”,或门输出端是“0”;
- (2)输入端为“0”,“1”或是“1”,“0”,与门输出端都是“0”,或门输出端都是“1”;
- (3)输入端为“1”,“0”,与门输出端是“1”,或门输出端是“1”.

三种情况正好与 CMOS 反相器三种情况的相对应:与门对应 CMOS 反相器的 A 端,或门对应的 B 端.同时 CMOS 反向器的三种情况又可以归纳成两种情况,即 A 端、B 端相同, V_{out} 端电压为低电压; A 端、B 端不同, V_{out} 端电压为高电压.这两种情况正好对应异或运算的情况. 根据分析的情况,将与门、或门、CMOS 反向器组合搭建实现异或门,如图 5 所示.

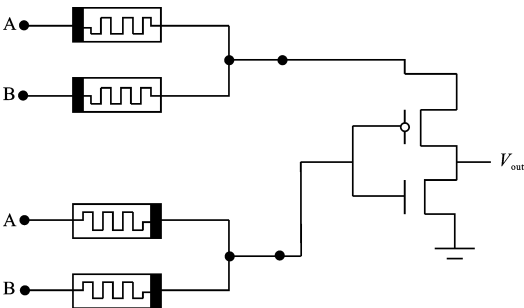


图 5 基于忆阻器实现的异或门

基于忆阻器的异或门就实现了,通过上述分析,

该异或门是符合逻辑异或运算.

3.2 二位二进制吠陀乘法器的设计实现

吠陀数学是一个完整的数学系统,本文使用的是吠陀乘法中 Urdhva Tiryakbhyam 算法. 这里用十进制三位数乘以十进制三位数来介绍算法过程,如图 6 所示.

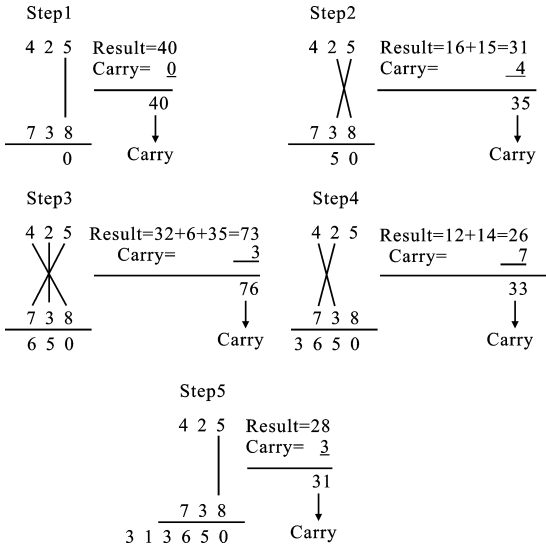


图 6 Urdhva Tiryakbhyam 算法步骤图

步骤一:最低位相乘,结果 $R = 40$,来自低位计算的进位 $C_0 = 0$,令 $T = C_0 + R = 40$,向高位的进位 C_1 为 T 的最高位,即 $C_1 = 4$, T 的最低位 0 留下作为最后结果的最低位;

步骤二:被乘数与乘数的最低位和次低位交叉相乘并把结果相加, $R = 31$,这里的 C_0 为步骤一的 C_1 ,即 $C_0 = 4$, $T = C_0 + R = 35$,此时向高位进位 $C_1 = 3$, T 的最低位作为最后结果的十位数;

步骤三:被乘数的最高位和乘数最低位相乘,乘数的最高位和被乘数最低位相乘,被乘数和乘数中间位相乘,计算结果再相加, $R = 73$,同时这里 $C_0 = 3$,则 $T = C_0 + R = 76$,此时向高位进位 $C_1 = 7$, T 的最低位作为最后结果的百位数;

步骤四:被乘数的最高位和乘数次低位相乘,乘数的最高位和被乘数次低位相乘,计算结果相加, $R = 26$,这里 $C_0 = 7$,则 $T = C_0 + R = 33$,此时向后进位 $C_1 = 3$, T 的最低为作为最后结果的千位数;

步骤五:被乘数的最高位和乘数最高位相乘,结果 $R = 28$,这里 $C_0 = 3$,则 $T = C_0 + R = 31$,其实此时就可以将 31 作为最后结果的最高两位填入,因此最后结果就是 313 650.

该算法是将复杂的三位数乘法换成个位数的乘法和加法,使计算简便. 这种算法对于计算的速度提

升有很大意义,目前许多研究者已经开始将吠陀乘法运用到计算机中,进一步加快计算机的计算速度.经研究表明,Urdhva Tiryakbhyam 算法同样适用于二进制数的运算,本文就是根据这点来设计二进制二位乘法器.

二进制数运算是有其特点的,由于二进制数只有“0”和“1”,这样使得计算更加简便,使得该算法在计算机的应用上显得更加有意义.以下是本文设计的乘法器的核心算法:

设被乘数为 A_1A_0 ,其中 A_1 代表被乘数最高位, A_0 代表被乘数最低位,乘数为 B_1B_0 ,其中 B_1 为乘数最高位, B_0 为乘数最低位. S_0, S_1, S_2, S_3 表示最后结果的最低位到最高位的各个位置的数.

(1) A_0 与 B_0 相乘,因为 A_0 与 B_0 都是二进制数,而二进制数相乘的特点是只有在 A_0 与 B_0 都为“1”时,其结果才为“1”,符合逻辑与运算真值表的结果,所以将 A_0 与 B_0 作为基于忆阻器的与门的两个输入端,输出端为 S_0 . 由于是一位数相乘而且没有加法,这里并没有产生进位,所以没有进位项.

(2) 根据算法是 A_1 与 B_0, A_0 与 B_1 分别相乘,用两个与门来完成这两个乘法. 设 A_1 和 B_0 的与门输出端为 P_1, A_0 和 B_1 的与门输出端为 P_2 ;

(3) 接着应当把 P_1 与 P_2 结果相加,此时,考虑两个二进制数加法的特点:

- ①“0”和“0”相加,结果为“0”;
- ②“0”和“1”相加,结果为“1”;
- ③“1”和“1”相加,结果为“0”,进位为“1”,但是此时不考虑进位.

在不考虑进位的情况下,这种特点正好符合逻辑运算中异或运算,所以将 P_1 和 P_2 作为已设计好的异或门的两个输入端,即在步骤(2)的基础上将两个与门输出端分别连接到异或门的输入端上,而异或门的输出端就作为 S_1 ;

(4) 第(3)步是暂时不考虑进位的,这一步将考虑进位. 二进制数相加只有在都是“1”的情况下才会出现进位,由于异或门内部的与门输出端的变化和进位项的变化是一样的,该输出端可以作为进位,因此令异或门内部的与门输出端作为进位 C ;

(5) A_1 与 B_1 相乘,用与门实现,结果输出端为 P_3 ;

(6) 将进位 C 和 P_3 作为异或门的两个输入端,即将第(4)步和第(5)步形成的逻辑电路再用一个异或门连接起来,而这个异或门的输出端结果作为 S_2 ,对于进位处理和第(4)一样,将这一步的异或门内部的与门输出端作为 S_3 . 最后结果就为 $S_3S_2S_1S_0$.

按照上面算法设计组合逻辑电路的乘法器如图 7 所示.

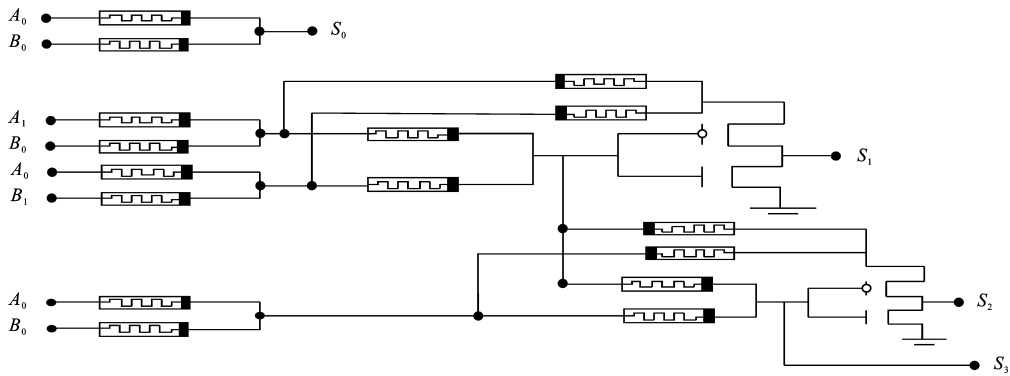
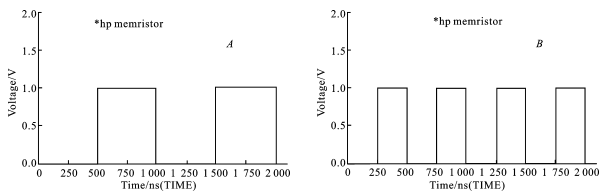


图 7 二位二进制吠陀乘法器

4 实验结果与分析

本文通过 SPICE 仿真软件对所设计的逻辑门电路和组合逻辑电路进行仿真,采用的是文献[4]所提出的忆阻器 SPICE 仿真模型. 实验根据文献[5]先完成了对与门或门的仿真. 此处取 $D = 1 \text{ nm}, \mu_V = 10^{-10} \text{ cm}^2 \text{ S}^{-1} \text{ V}^{-1}, p = 1, M_1, M_2: R_{\text{OFF}} = 16 \text{ k}\Omega, R_{\text{ON}} = 0.1 \text{ k}\Omega$. 仿真波形图如图 8 所示,图中

A, B 为输入信号, AND 为与门输出信号, OR 为或门输出信号.



(a) 输入端 A 的波形

(b) 输入端 B 的波形

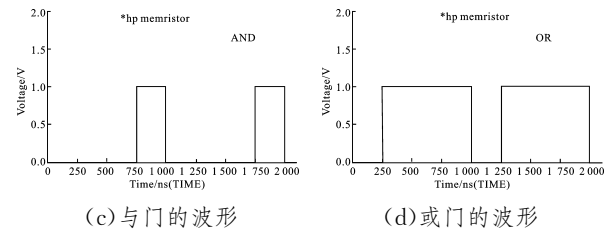


图 8 与门和或门仿真波形图

从图 8 可以看出,两种逻辑门的输出信号是和逻辑与、或门的真值表结果一样. 对本文第三部分设计的异或门进行仿真,图 9 是所设计异或门的仿真波形图. 其中输入端信号还是图 8 中 A,B 的输入信号,输出信号是图 9 所示的,其结果也是符合要求的.

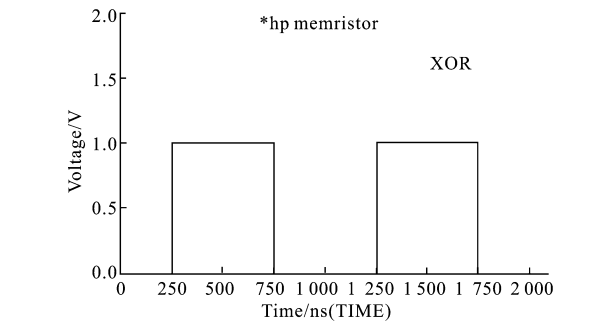


图 9 异或门仿真波形

从仿真结果来看,所设计的异或门是符合要求的,利用该异或门和已经设计过的与门、或门组合实现的二位二进制吠陀乘法器的仿真结果如下. 其中便于和文献[17]相比,这里将按照文献[17]仿真来设置周期长度和周期数.

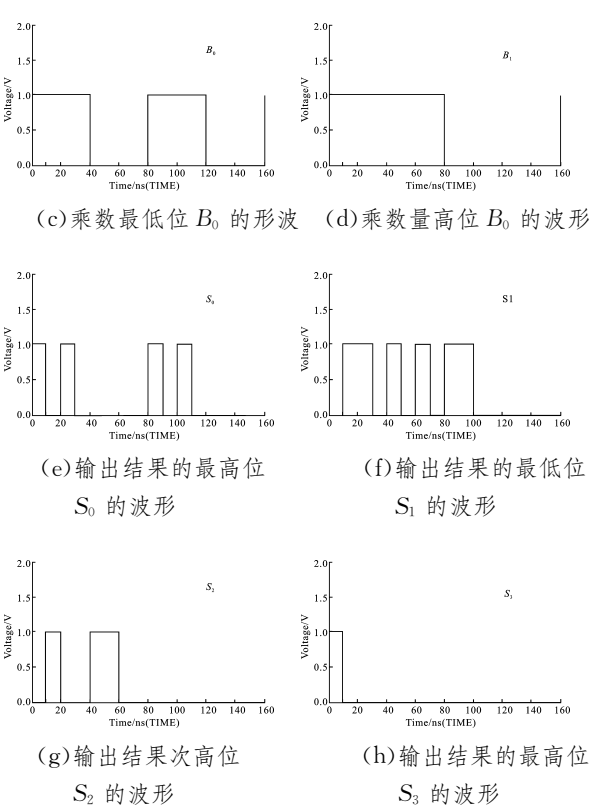
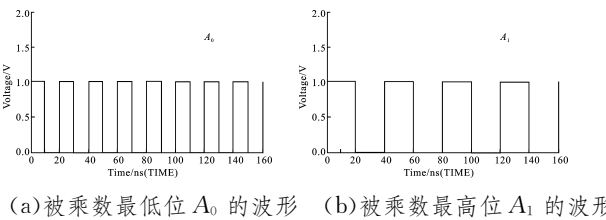


图 10 二位二进制吠陀乘法器仿真波形

其中 A_0,A_1 为被乘数的最低位和最高位, B_0,B_1 为乘数的最低位和最高位,而 S_0,S_1,S_2,S_3 依次为最后结果的最低位到最高位的各个位的数. 从仿真波形来看, S_0,S_1,S_2,S_3 波形变化是符合二进制乘法的.

本文的 2 位吠陀乘法器与文献[17]的乘法器相比较,本文的乘法器在功耗上,面积开销都有所减少,比较结果如表 4 所示.

从表中可以看出本文方法的面积开销比文献[17]小约为 76%,比纯 CMOS 电路也小;而其功耗比文献[17]方法的功耗小,约为文献[17]的 80%. 由于忆阻器本身不产生泄露功耗(Leakage power),

表 4 本文方案与文献[17]的对比

Multiplier	设计方法	面积/ μm^2	功耗/W	泄露功耗(Leakage power)/W
2bit—multiplier	CMOS	169.67	0.247 n	0.115 n
	文献[17]	50.05	1.09 μ	0.173 n
	本文	38.25	0.89 μ	0.095 n

表中的泄露功耗均是由乘法器中使用的 CMOS 反相器产生. 实验结果表明,本文方法在 3 种方法中实现了最小的泄露功耗.

5 结束语

本文设计了一种混合 CMOS 忆阻器逻辑门电

路,使用基于忆阻器的与门和或门、基于 CMOS 工艺的反相器,结合异或运算的特点,混合实现了基于忆阻器的低开销异或门电路. 并在此基础上,根据二进制数运算的特点,对原始电路进行优化,搭建新的 2 位二进制吠陀乘法器,使得其在功能上是符合乘法器运算的同时,减少了该乘法器的功耗和面积开

销. 实验结果表明,与其他混合 CMOS 忆阻器逻辑门电路相比,提出的电路有较低的面积要求和较低的功耗. 与纯 CMOS 电路相比面积开销较低,但是功耗却变得更高,这是因为本文所用的忆阻器 Spice 模型并不是低功耗的,随着低功耗忆阻器的研究,如文献[7],降低功耗已经是一个可以解决的问题,同时根据已有的研究表明,实际忆阻器的功耗和面积都是极小的. 该乘法器的实现,对将忆阻器运用到计算机领域,具有一定的积极作用.

参考文献:

- [1] Chua L O. Memristor-the missing circuit element[J]. IEEE Trans Circuit Theory, 1971, 18 (5): 507 - 519.
- [2] Strukoud B. The missing memristor found[J]. Nature, 2008, 453(7191): 80-83.
- [3] 蔡鹏坤. 第四种无源电子元件忆阻器的研究及应用进展[J]. 电子元件与材料, 2010, 29(4): 78-82.
- [4] Shen W C, Yuan H T, Chih Y D, et al. Memristor logic operation gate with share contact RRAM Cell[J]. IEEE Electron Device Letters, 2011, 32 (12): 1650-1652.
- [5] Kvatinisky S, Kolodny A, Weiser U C, et al. Memristor-based IMPLY logic design procedure[C]//IEEE International Conference on Computer Design, Amherst, MA, USA, 2011, 47 (10): 142-147.
- [6] Biolek Z, Biolek D, Biolkova V. SPICE model of memristor with nonlinear dopant drift[J]. Radio-engineering, 2009, 18(2): 210-214.
- [7] Goux L, Fantini A, Kar G, et al. Ultralow sub-500nA operating current high-performance Ti/Al₂O₃/HfO₂/Hf/TiN bipolar RRAM achieved through understanding-based stack-engineering[J]. Ecs Solid State Letters, 2012, 1 (4): 63-65.
- [8] Cho K, Lee S J, Eshraghian K. Memristor-CMOS logic and digital computational component[J]. Microelectronics Journal, 2015, 46(3): 214-220.
- [9] Chen L, He Z, Wang X, et al. Several logic gates extended from MAGIC-memristor-aided logic[J]. International Symposium on Neural Networks. Springer, Cham, 2017: 170-179.
- [10] Rose G S, Rajendran J, Manem H, et al. Leveraging memristive systems in the construction of digital logic circuits[J]. the IEEE, 2012, 100(6): 2033-2049.
- [11] Jo S H, Chang T, Ebong I, et al. Nanoscale memristor device as synapse in neuromorphic systems[J]. Nano Letters, 2010, 10(4): 1297-1301.
- [12] Del M H M. CMOS analog circuit design via geometric programming[J]. Ieice Trans Fundamentals A, 2004, 87(2): 3266-3271.
- [13] James A P, Francis L R V J, Kumar D S. Resistive threshold logic[J]. IEEE Transactions on Very Large Scale Integration Systems, 2013, 22(1): 190-195.
- [14] Xia Q, Robinett W, Cumbie M W, et al. Memristor-CMOS hybrid integrated circuits for reconfigurable logic. [J]. Nano Letters, 2009, 9(10): 36-40.
- [15] Eshraghian K, Cho K R, Kavehei O, et al. Memristor MOS content addressable memory (MCAM): hybrid architecture for future high performance search engines [J]. IEEE Transactions on Very Large Scale Integration Systems, 2010, 19(8): 1407-1417.
- [16] Pushpangadan R, Sukumaran V, Innocent R, et al. High speed vedic multiplier for digital signal processors[J]. Iete Journal of Research, 2009, 55 (6): 282-286.
- [17] James A P, Kumar D S, Ajayan A. Threshold logic computing: memristive-CMOS circuits for fast fourier transform and vedic multiplication[J]. IEEE Transactions on Very Large Scale Integration Systems, 2015, 23(11): 2690-2694.

作者简介:

刘海峰 男, (1992-), 硕士研究生. 研究方向为忆阻器的应用与研究. E-mail: 1097442285@qq.com.

方芳 男, (1979-), 博士, 副教授. 研究方向为决策支持、调度管理技术、多模态数据融合.

王伟 男, (1979-), 博士, 副教授, 硕士生导师. 研究方向为超大规模集成电路设计与测试、SoC 测试与可靠性设计技术、类脑计算、智能硬件.

王彬 男, (1991-), 硕士研究生. 研究方向为忆阻器的应用与研究.