

两层阵列二的补码复数并行算法 及其光学实现*

李国强 刘立人 邵 岚

(中国科学院上海光学精密机械研究所, 上海 201800)

摘 要 提出了直接的二的补码阵列算法。基于混合二的补码表达方式, 加法、减法和阵列乘法都能全并行地实现。直接算法克服了传统算法中存在的困难, 即原理上对操作数的编码位数无限制, 更有效地利用了空间带宽积, 卷积后的序列直接加权求和即为两数的乘积。复数运算由包含四个区域的两层阵列完成, 并进行了光学实验。

关键词 并行光计算, 数字编码, 阵列乘法, 复数运算。

1 引 言

数字编码是光计算最基本的问题, 它与运算精度、算法及体系结构密切相关。指导思想是尽可能使算法无进位, 以便于光学并行实现。迄今为止, 人们已深入研究过二进制、余数^[1]、原码^[2]、二的补码^[3]、MSD^[4]等编码方法。其中二进制只能对正数编码, 处理负数时需附加偏置量^[5]或采用时间^[6]、空间或频率^[7]复用技术, 从而需要对中间结果进行矫正、存储或组合; 后几种能表示正负数, 但原码的乘法存在符号判断, 加法不能直接进行。余数和 MSD 算法基于查找表或符号替代, 光学硬件不易实现, 不能用于快速矩阵运算。二的补码加法无进位, 乘法通过模拟卷积实现, 能使算法完全并行, 但传统的二的补码乘法有几点不足: 1) 乘数和被乘数必须用表示它们的乘积所需的位数来编码, 因而要事先知道这个位数。事实上, 这种限制已经使相乘运算本身失去了意义; 2) 上述编码后的乘积只保留后一半, 而前一半要舍掉, 这浪费了光学系统的空间带宽积; 3) 中间结果对模 2 进行时序操作, 所得余数才是最终结果。因此, 该算法亦不能有效地进行矩阵运算。Taylor 等人^[8]曾提出了一种改进方案, 解决了前两个问题, 可是后处理量大, 且需作很多符号判断和无规则的校正。本文正试图彻底克服以上几点困难。文中首先引入了直接的二的补码阵列并行算法^[9]。在混合二的补码数制表达下, 加法无进位, 减法通过加法完成。对任意两个 N 比特的数相乘, 只需要 $(N+2)$ 个通道, 在均一化的阵列结构实现乘积过程中, 操作数同时正交输入, 在交叉点形成符号全为正的乘积项, 对各对角线方向乘积项分别求和便直接得到混合进制表示的乘积结果。一切操作都是全并行的。该算法可推广到任意基的补码编码。

* 国家自然科学基金资助项目。

收稿日期: 1994年3月28日; 收到修改稿日期: 1994年5月20日

在实际应用中,复数运算更具有有一般性,然而目前数字编码还主要用于实数。根据二的补码阵列算法,本文提出了一种两层阵列复数乘法方案。一个复数单元包含 4 个子区域,每个子区域完成一对实数相乘,其中两个区域对应实部,另两区域对应虚部,将它们的对应项相加,对实部再加上附加项即为最终结果。阵列算法中仅含求反、与、求和等操作,非常适合于光学实现。文中给出了全并行的自由空间光学实验系统和实验结果。

2 直接的二的补码阵列并行乘法

在二的补码编码中,任一模拟量 x 可用一个 N 比特的序列 $x_{N-1}x_{N-2}\cdots x_1x_0$ 表示。其中 x_{N-1} 为符号位,正数为 0,负数为 1。表达式为

$$x = -x_{N-1}2^{N-1} + \sum_{i=0}^{N-2} x_i 2^i, \quad (1)$$

x 的相反数 $-x$ 等于

$$-x = -\bar{x}_{N-1}2^{N-1} + \sum_{i=0}^{N-2} \bar{x}_i 2^{i+1} + 1, \quad \bar{x}_i = 1 - x_i, \quad (i = 0, 1, \dots, N-1). \quad (2)$$

设两操作数分别表示成 $u_{N-1}u_{N-2}\cdots u_1u_0$ 和 $v_{N-1}v_{N-2}\cdots v_1v_0$ 。下面将讨论直接的二的补码并行算法,包括加法、减法和阵列乘法。

1) 加法

与电子学二进制算术不同,混合的 2 的补码表达法使得 u 、 v 两数的相加可以无进位地分别将它们对应位相加,结果 s 为一个新的 N 位的序列 $s_{N-1}s_{N-2}\cdots s_1s_0$, 并满足下列关系:

$$s = u + v = -s_{N-1}2^{N-1} + \sum_{i=0}^{N-2} s_i 2^i, \quad s_i = u_i + v_i, \quad (i = 0, 1, \dots, N-1). \quad (3)$$

2) 减法

2)式表明, u 和 v 之差 d 可由下列和式求得:

$$d = u - v = -d_{N-1}2^{N-1} + \sum_{i=0}^{N-2} d_i 2^i + 1, \quad d_i = u_i + \bar{v}_i, \quad (i = 0, 1, \dots, N-1). \quad (4)$$

3) 阵列乘法

根据以上概念,可以导出阵列并行算法。数 u 和 v 的积为

$$\begin{aligned} p &= u \times v \\ &= (-u_{N-1}2^{N-1} + \sum_{i=0}^{N-2} u_i 2^i)(-v_{N-1}2^{N-1} + \sum_{i=0}^{N-2} v_i 2^i) \\ &= (u_{N-1}v_{N-1}2^{2N-2} + \sum_{i=0}^{N-2} \sum_{k=0}^{N-2} u_i v_k 2^{i+k}) - (\sum_{i=0}^{N-2} u_{N-1}v_i 2^{i+N-1} + \sum_{i=0}^{N-2} v_{N-1}u_i 2^{i+N-1}). \end{aligned} \quad (5)$$

注意到上式中后半部分的乘积项是负的。利用(2)式可将它们转化为正项。由于

$$\sum_{i=0}^{N-2} v_i 2^i = 0 \cdot 2^{N-1} + \sum_{i=0}^{N-2} v_i 2^i, \quad (6)$$

$$\therefore -\sum_{i=0}^{N-2} v_i 2^i = -1 \cdot 2^{N-1} + \sum_{i=0}^{N-2} \bar{v}_i 2^i + 1. \quad (7)$$

将(7)式代入(5)式,得到

$$-u_{N-1}2^{N-1} \sum_{i=0}^{N-2} v_i 2^i = (-1 \cdot 2^{N-1} + \sum_{i=0}^{N-2} \bar{v}_i 2^i + 1) u_{N-1} 2^{N-1}. \quad (8)$$

类似地,

$$-v_{N-1}2^{N-1} \sum_{i=0}^{N-2} u_i 2^i = (-1 \cdot 2^{N-1} + \sum_{i=0}^{N-2} \bar{u}_i 2^i + 1) v_{N-1} 2^{N-1}. \quad (9)$$

因此,(5)式可以写成

$$p = (u_{N-1}v_{N-1} - u_{N-1} - v_{N-1}) 2^{2N-2} + \sum_{i=0}^{N-2} \sum_{k=0}^{N-2-i} u_i v_k 2^{i+k} + \sum_{i=0}^{N-2} (u_{N-1} \bar{v}_i + v_{N-1} \bar{u}_i) 2^{i+N-1} + (u_{N-1} + v_{N-1}) 2^{N-1}. \tag{10}$$

显然，系数 $z = u_{N-1}v_{N-1} - u_{N-1} - v_{N-1}$ 需作进一步考虑。其真值见表 1。为将 z 中的减法转变成光学上易执行的操作，可作如下恒等变换：

Table 1. Truth table for $z = u_{N-1}v_{N-1}$

| u_{N-1} | v_{N-1} | z | $z + 2$ |
|-----------|-----------|-----|---------|
| 0 | 0 | 0 | 2 |
| 1 | 0 | -1 | 1 |
| 0 | 1 | -1 | 1 |
| 1 | 1 | -1 | 1 |

$$\begin{aligned} & (u_{N-1}v_{N-1} - u_{N-1} - v_{N-1}) 2^{2N-2} \\ &= (z + 2) 2^{2N-2} - 1 \cdot 2^{2N-1} \\ &= (1 + \bar{u}_{N-1}\bar{v}_{N-1}) 2^{2N-2} - 1 \cdot 2^{2N-1}. \end{aligned} \tag{11}$$

这样(5)式最终等效于下列均一化的全加形式：

$$p = 1 \cdot (-2^{2N-1}) + (1 + \bar{u}_{N-1}\bar{v}_{N-1}) 2^{2N-2} + \sum_{i=0}^{N-2} \sum_{k=0}^{N-2-i} u_i v_k 2^{i+k} + \sum_{i=0}^{N-2} (u_{N-1} \bar{v}_i + v_{N-1} \bar{u}_i) 2^{i+N-1} + (u_{N-1} + v_{N-1}) 2^{N-1}. \tag{12}$$

利用阵列结构可以实现上述卷积。图 1 示出了包含 u_j 和 v_j 及其反的乘法列阵。阵列中，两数的各位数字正交展开，所有的乘积项都通过各交叉点上对应数字的操作并行地产生，同时每一条对角线方向上的乘积项权重相等，因而将它们无进位地分别相加，就得到一个 $2N$ 位的混合补码进制序列表示的乘积结果。与二的补码一样，前 $2N - 1$ 位的权重为正，符号位权重为负。加权求和的结果直接为 u 和 v 的乘积。换句话说，不需要中间的数制转换。与传统的二的补码算法相比，这是一个很大的优点。而且这里的直接算法只需要 $N + 2$ 个通道，每个乘积项都直接贡献于结果，这大大节省了空间带宽积。此外，若(12)式中的最后一项由后处理完成，则使用 N 个通道也是可行的。再者，运算前无需知道表达乘积所需的位数。总之，改进的二的补码列阵并行算法解决了原有算法存在的困难。

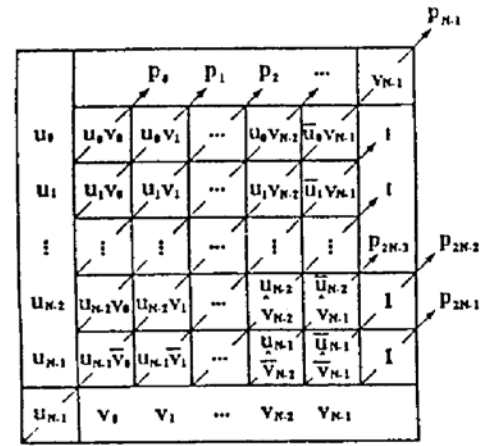


Fig. 1 Parallel array multiplication scheme

3 两层阵列复数乘法

阵列乘法可用于复数运算。设两操作数为 $a = a_1 + ja_2$ 和 $b = b_1 + jb_2$ ，其中 j 为虚数单位， a_1, a_2, b_1 及 b_2 均为实数，且表示成 N 比特的二的补码形式，即 $(a_1^{N-1}, \dots, a_1^1, a_1^0), (a_2^{N-1}, \dots, a_2^1, a_2^0), (b_1^{N-1}, \dots, b_1^1, b_1^0), (b_2^{N-1}, \dots, b_2^1, b_2^0)$ 。这里的位序由上标记。 a 和 b 的乘积为

$$a \times b = (a_1 + ja_2)(b_1 + jb_2) = (a_1b_1 - a_2b_2) + j(a_1b_2 + a_2b_1) \tag{13}$$

其中乘积 a_1b_1, a_1b_2 及 a_2b_1 可直接根据阵列算法得到。为避免实部的减法，将 $a_1b_1 - a_2b_2$ 改写成 $a_1b_1 + (-a_2b_2)$ ，其中

$$\begin{aligned} (-a_2) \times b_2 &= (-b_2^{N-1} 2^{N-1} + \sum_{i=0}^{N-2} b_2^i 2^i) (-\bar{a}_2^{N-1} 2^{N-1} + \sum_{i=0}^{N-2} \bar{a}_2^i 2^i + 1) \\ &= (-b_2^{N-1} 2^{N-1} + \sum_{i=0}^{N-2} b_2^i 2^i) (-\bar{a}_2^{N-1} 2^{N-1} + \sum_{i=0}^{N-2} \bar{a}_2^i 2^i) \\ &\quad + (-b_2^{N-1} 2^{N-1} + \sum_{i=0}^{N-2} b_2^i 2^i). \end{aligned} \tag{14}$$

与(12)式的推导类似，(14)式可演化为下列多项式：

$$\begin{aligned}
 (-a_2) \times b_2 = & 1 \cdot (-2^{2N-1}) + (1 + a_2^{N-1} b_2^{N-1}) 2^{2N-2} + \sum_{i=0}^{N-2} \sum_{k=0}^{N-2} \bar{a}_2^i b_2^k 2^{i+k} \\
 & + \sum_{i=0}^{N-2} (\bar{a}_2^{N-1} \bar{b}_2^i + b_2^{N-1} a_2^i) 2^{i+N-1} + (\bar{a}_2^{N-1} 2^{N-1} + \sum_{i=0}^{N-2} b_2^i 2^i). \quad (15)
 \end{aligned}$$

这可用图 2 所示的阵列产生,除了乘积项 $b_2^i 2^i (i = 0, 1, \dots, N-2)$ 留作后处理外。与其它方案相比,这里的后处理既简单又规则。

(13)式中的四项乘积能够用一个两层阵列并行实现(见图 3)。子阵列 $a_1 b_1, a_1 b_2, a_2 b_1$ 形式相同,而子阵列 $(-a_2 b_2)$ 稍有差别,与 $a_2 b_1$ 相比,这一区域的 a_2 作了一次反。这可以很简单地

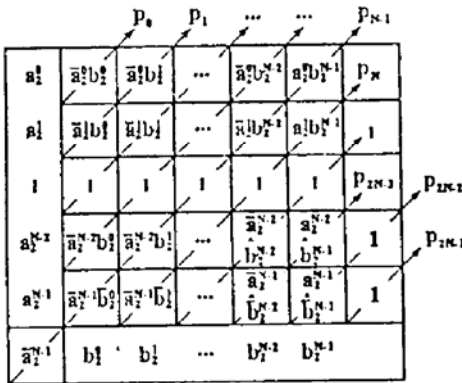


Fig. 2 Arrangement for $(-a_2 b_2)$ subarray

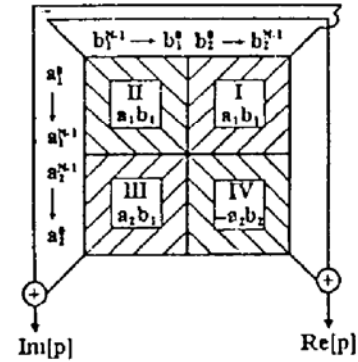


Fig. 3 A two-stage array complex multiplier unit

通过偏振光学调制。实部和虚部的各对应项在阵列外相加,从而得到

$$\begin{aligned}
 \text{Re}[p] = & 2(-2^{2N-1}) + (2 + \bar{a}_1^{N-1} \bar{b}_1^{N-1} + a_2^{N-1} \bar{b}_2^{N-1}) 2^{2N-2} + \sum_{i=0}^{N-2} \sum_{k=0}^{N-2} (a_1^i b_1^k \\
 & + \bar{a}_2^i b_2^k) 2^{i+k} + \sum_{i=0}^{N-2} (a_1^{N-1} \bar{b}_1^i + b_1^{N-1} \bar{a}_1^i + \bar{a}_2^{N-1} b_2^i + b_2^{N-1} a_1^i) 2^{i+N-1} \\
 & + (a_1^{N-1} + b_1^{N-1} + \bar{a}_2^{N-1}) 2^{N-1} + \sum_{i=0}^{N-2} b_2^i 2^i, \quad (16)
 \end{aligned}$$

$$\begin{aligned}
 \text{Im}[p] = & 2(-2^{2N-1}) + (2 + \bar{a}_1^{N-1} \bar{b}_2^{N-1} + a_2^{N-1} \bar{b}_1^{N-1}) 2^{2N-2} + \sum_{i=0}^{N-2} \sum_{k=0}^{N-2} (a_1^i b_2^k \\
 & + a_2^i b_1^k) 2^{i+k} + \sum_{i=0}^{N-2} (a_1^{N-1} \bar{b}_2^i + b_2^{N-1} \bar{a}_1^i + a_2^{N-1} \bar{b}_1^i + b_1^{N-1} \bar{a}_2^i) 2^{i+N-1} \\
 & + (a_1^{N-1} + a_2^{N-1} + b_1^{N-1} + b_2^{N-1}) 2^{N-1}, \quad (17)
 \end{aligned}$$

基于两层阵列复数相乘单元,采用三层阵列结构可实现复数矩阵运算。

4 实 验

对两层阵列的复数运算进行了自由空间光学模拟实验。不失一般性,选取 $N = 9$ 。相应地,二的补码表达范围为 $[-256, 255]$ 。任取 $a = 255 + j(-200), b = 103 + j207$, 它们的数字编码为 $a = 011111111 + j100111000, b = 001100111 + j011001111$ 。根据(12)式和图 1,子阵列 $a_1 b_1$ 示于图 4(a)中,子阵列 $a_1 b_2$ 和 $a_2 b_1$ 也同样得到。而子阵列 $(-a_2 b_2)$ 则依照(15)式。图 4(b)是 $a_1 b_1$ 和 $a_2 b_2$ 及 $a_1 b_2$ 和 $a_2 b_1$ 分别相加后的阵列,左面对应实部 $\text{Re}[p]$,右面对应虚部 $\text{Im}[p]$ 。对角线方向的并行相加产生下列两个混合二的补码序列:

序列 1: 240133236897556642, 序列 2: 23112345677765321。

序列 2 的加权求和即为虚部 $j32185$; 序列 1 的值为 67548, 贡献于实部,再加上校正项 $b_2^i 2^i (i = 0, 1, \dots, 7)$, 得到 67665。因此,乘积结果为 $67665 + j32185$ 。

实现两层阵列复数乘法的光学系统示于图 5 中,包含三个部分:

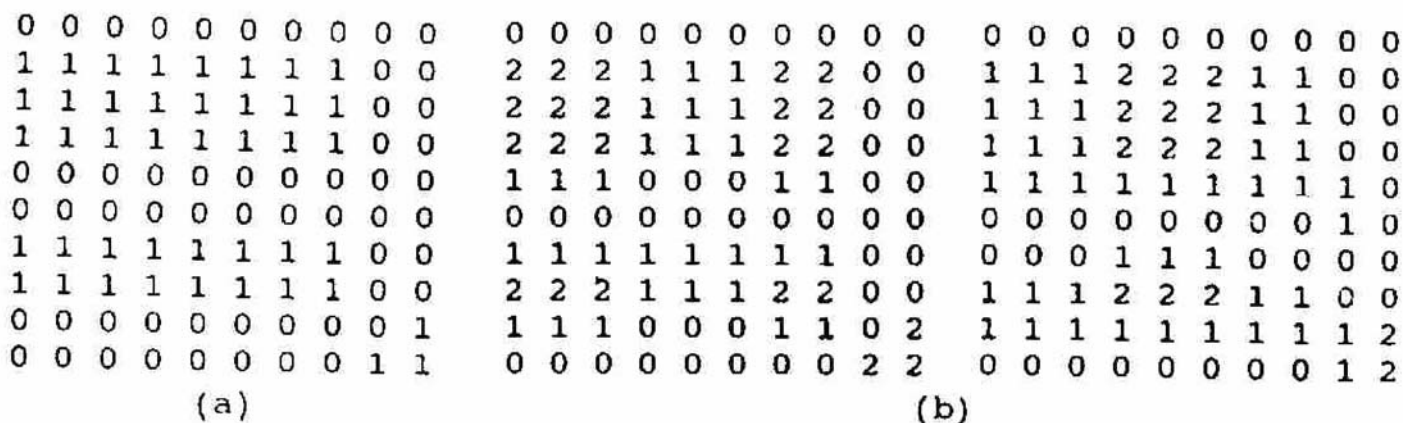


Fig. 4 Numerical simulations. (a) a, b_1 subarray; (b) multiple-valued product array for p , with the left half — $Re [p]$, the right half — $Im [p]$

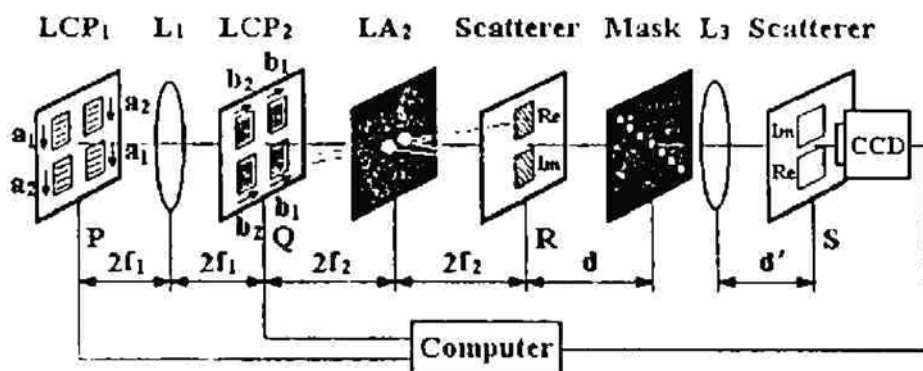


Fig. 5 Optical configuration for the two-stage array complex multiplier

第一部分利用透镜成像形成四个乘积项子阵列。复数 a 和 b 分别从两个液晶屏(LCP)输入, 每个液晶屏上都有四个子阵列, 每个子阵列又分别代表一个实数。液晶屏用均匀光源照明。在 LCP_1 上, 数字垂直输入, 水平展开; LCP_2 上, 数字水平输入, 垂直展开。箭头指向数字位序升高的方向。 LCP_1 上的图案成像到 LCP_2 上, 并互相匹配。由于两个液晶屏的出射光的偏振方向相同, LCP_2 后的出射图案便是两者对应点相乘结果。于是就得到了含有四个子阵列的两层阵列。

第二部分是两个透镜构成的多重成像系统^[10], 将相应的两组子阵列分别相加, 从而合并成两个多值化的子阵列, 分别对应乘积的实部和虚部。

第三部分利用非相干光相关器^[11]同时将这两个子阵列对角线上的元素分别相加, 得到混和二补码表达的实部和虚部。在相关面上, 像点沿对角线方向伸展, 需用掩模板遮去不需要的点。像点的强度表示数值的大小。系统与一微机相连, 完成预处理和后处理。

实验系统的有关参数如下: $f_1 = 240 \text{ mm}$, $f_2 = 90 \text{ mm}$, $d = 110 \text{ mm}$, $d' = 58 \text{ mm}$ 。阵列中每个像素的大小约 1 mm , 子阵列的水平间距为 100 mm , 透镜列阵上两透镜的间距为 50 mm 。图 6 是相关面上的强度分布、上、下两部分的中央水平方向最长的一行像点分别对应乘积的虚部和实部。

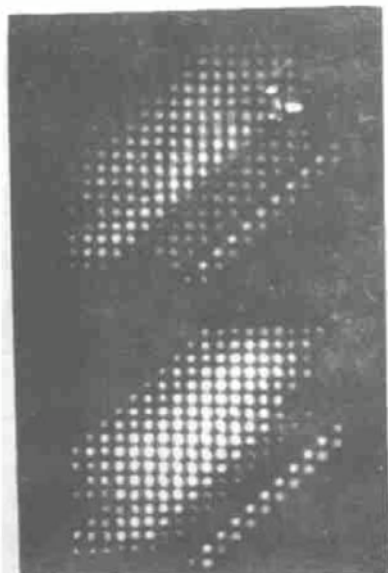


Fig. 6 Experimental photograph

结束语 本文提出了直接的二的补码并行阵列算法，克服了原有算法中的困难。利用四个实数子阵列构成的两层阵列实现了高精度的复数运算。三层阵列可实现复数矩阵运算。实验上，用全并行的自由空间光学系统执行了两层阵列的复数乘法。对一个给定的数，适当增大基底将减少通道数，本文提出的补码阵列算法适用于其它高阶基。此外，光学结构可集成模块化。

参 考 文 献

- [1] A. Huang, Y. Tsunoda, J. Goodman *et al.*, Optical computation using residue arithmetic. *Appl. Opt.*, 1979, 18(2): 149~162
- [2] D. Casasent, B. Taylor, Banded-matrix high-performance algorithm and architecture. *Appl. Opt.*, 1985, 24(10): 1476~1480
- [3] R. Bocker, S. Clayton, K. Bromley, Electrooptical matrix multiplication using the twos complement arithmetic for improved accuracy. *Appl. Opt.*, 1983, 22(13): 2019~2021
- [4] B. Drake, R. Bocker, M. Lasher *et al.*, Photonic computing using the modified signed-digit number representation. *Opt. Engng.*, 1986, 25(1): 38~43
- [5] D. Casasent, C. Perlee, Bipolar biasing in high accuracy optical linear algebra processors. *Appl. Opt.*, 1986, 25(7): 1033~1035
- [6] W. Collins, R. Athale, P. Stlwell, Improved accuracy for an optical iterative processor. *Proc. SPIE*, 1983, 352: 59~66
- [7] D. Casasent, J. Jackson, Space and frequency-multiplexed optical algebra processor; fabrication and initial tests. *Appl. Opt.*, 1986, 25(14): 2258~2263
- [8] B. Taylor, D. Casasent, Twos-complement data processing for improved encoded matrix-vector processors. *Appl. Opt.*, 1986, 25(6): 956~961
- [9] C. Baugh, B. Wooley, A two's complement parallel array multiplication algorithm. *IEEE Trans. Computers*, 1973, C-22(12): 1045~1047
- [10] L. Liu, X. Liu, Cascadable binary pattern logic processor using multiple-imaging. *Opt. Commun.*, 1991, 82(5, 6): 446~452
- [11] L. Liu, Optoelectronic implementation of mathematical morphology. *Opt. Lett.*, 1989, 14(10): 482~484

Two-Stage Twos Complement Array Complex Multiplier Algorithm and Optical Implementation

Li Guoqiang Liu Liren Shao Lan

(Shanghai Institute of Optics and Fine Mechanics, Academia Sinica, Shanghai 201800)

(Received 28 March 1994; revised 20 May 1994)

Abstract A direct twos complement parallel array multiplication algorithm is introduced for digital optical computing without carries. In contrast to the conventional 2's complement algorithm, the proposed version has no restriction on the bit number of the operands and uses space bandwidth product more efficiently. On this basis, complex operation is available with a two-stage array architecture. Correspondingly, an optical experiment is demonstrated.

Key words parallel optical computing, digital encoding, array multiplication, complex operation.