Designing an Optical Multiplier by Using a Modified Signed-Digit Number System

Sabah S. Alsheraidah
Basrah Technical Institute, Basrah
(E-mail: sabahhd@yahoo.com)

ISSN –1817 –2695
((Received 15/1/2008, Accepted 13/4/2009))

Abstract

In this paper, a parallel optical 2D data array multiplier for MSD (modified signed-digit) number system is proposed and designed. The design is performed by using DDP (digit-decomposition-plan) technique and DSS (duplication-shifting-superimposing) multiplication algorithm. The MSD multiplier is based on logical formulas which are newly derived according to the fundamental parallel multiplication algorithm. An optical implementation with classical optical elements is suggested for this multiplier. A simulated demonstration example is performed to validate the proposed design.

KEYWORDS: OPTICAL COMPUTER, ARITHMETIC, MODIFIED-SIGNED-DIGIT, MULTIPLIER

1. Introduction

Some optical computing techniques have been established by using many parallel algorithms in order to perform the arithmetic operations. At the beginning of the eighties, researchers tended to use a new technology of data processing in various purposes such as image processing, data computing, control systems, etc. The proposed technology used light in large parts of the processing systems. Hence, it is called *" Optical Processing Systems"*. The use of light in the optical processing systems led to many advantages. The important advantage is the very high speed with the parallelism. The parallelism of optical processing systems can be achieved by processing the data in one-dimension (1D) and multi-dimension (2D, 3D… etc.) [1]. The optical computing techniques have been established by using many parallel algorithms in order to perform the arithmetic operations. Some of these parallel algorithms used the redundant binary numbers [2]. Some other used a multi-leveled number system based on residue arithmetic algorithm [3]. The common technique used in optical parallel arithmetic operations was the *"Signed-Digit number systems"* (SD) [4]. Optical coding schemes had been investigated for coding the information and realizing the parallel algorithms such as symbolic substitution (SS) [5], optical shadow casting (OSC) [6], optical correlation [7] and digit-decomposition plane representation (DDP) [8]. Another parallel MSD multiplier based on SS technique had been discussed in [9]. In this research, parallel optical 2D array multiplier for the MSD number system is suggested. The process is based on DDP representation technique and DSS multiplication algorithm. Also, the proposed parallel MSD multiplier is implemented optically by using simple optical tools. Finally, a simulation example is discussed in order to verify these design principles.

2. MSD Number System

A modified signed decimal number D can be represented in terms of an n-bit, and radix-r signed-digit number as:

$$D = \sum_{i=0}^{n-1} x_i r^i \qquad (1)$$

In MSD number system, r=2 and the digit x_i is a member of the set $\{\bar{1},0,1\}$ where $\bar{1}$ represents -1. Since a number may have more than one representation in the signed-digit number, it is also known as a redundant number system. For illustration, consider the following decimal number 19 and its equivalent MSD representation:

$$(19)_{10} = (10011)_{MSD} = (1010\,\bar{1})_{MSD} = (101\,\bar{1}1)_{MSD} = (11\,\bar{1}\,\bar{1}1)_{MSD}$$

This redundancy helps us to achieve parallel carry-free addition and borrow-free subtraction as well as other complex arithmetic operations.

3. Digit-Decomposition-Plane Representation (DDP)

DDP representation has proposed by *Hongxin Huang* and *etal* [8]. It is an extension for bit-plane representation method. DDP representation can be applied to code large 2D data arrays of MSD numbers. It can be explained by applying it to 2D MSD numbers array as follows:

(i) If (i,j)-th digit in the 2D MSD numbers array equals to 1, so the corresponding (i,j)-th pixel of the DDP-1 plane will be transparent (white) and the corresponding (i,j)-th pixels of DDP-0 and DDP-$\bar{1}$ planes will be opaque (dark).

(ii) If the digit equals to 0, then the corresponding pixel of DDP-0 plane will be transparent and the corresponding pixels of DDP-1 and DDP-$\bar{1}$ planes will be opaque.

(iii) Otherwise, if the digit equals to $\bar{1}$, the corresponding pixel of DDP-$\bar{1}$ plane will be transparent and the corresponding pixels of DDP-1 and DDP-0 planes will be opaque.

Figure (1) shows an example for DDP coding method for MSD numbers array (3 DDP planes).

$$\begin{bmatrix} 3 \\ -4 \\ 6 \\ -3 \end{bmatrix}_{10} = \begin{bmatrix} 010\,\bar{1} \\ \bar{1}100 \\ 0110 \\ 00\bar{1}\,\bar{1} \end{bmatrix}_{MSD}$$

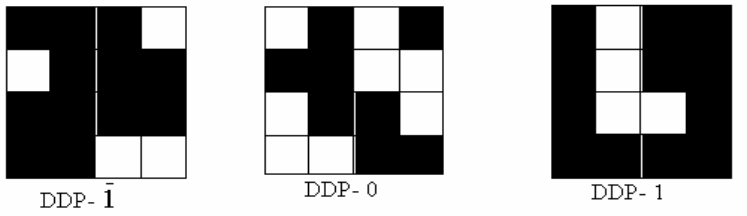


Fig. 1: DDP Representations of the MSD Data Arrays

There are two advantages of DDP coding method. The first one is that any DDP plane is the complement of the superimposing plane of the other planes. The superimposing plane is obtained by accumulating the bright and dark pixels in one plane. The second one is that if all DDP planes are superimposed, the result will be a totally transparent plane.

4. Parallel Multiplication Algorithm for MSD Number Systems

In the classical multiplication algorithm, when two n-digit numbers (X called a multiplicand and Y called a multiplier) are multiplied, 2n-digit number will be produced as maximum result. Each i-th digit of the multiplier y_i is multiplied by the multiplicand X to produce i-th partial product PP_i ($pp_{n-1},\ldots,pp_2, pp_1, pp_0$). The n PP_i s are shifted each i position to the left, and added together to calculate the final product Z which is 2n-digit length at maximum. In general, the multiplication process for

any number system passes through two phases, first one is partial products generation and second one is partial products accumulation to find the final result. Equation (2) explains the PP_i s generation and accumulation:

$$PP_i = X \quad y_i \quad r^i$$

$$Z = \sum_{i=0}^{n-1} PP_i' \qquad (2)$$

where
X : multiplicand number,
y_i : i-th digit of the multiplier number Y,
PP_i : i-th partial product,
PP_i' : i-th partial product shifted i position to the left,
Z : final product, and
r : radix of number system =2.

The n partial products PP_i s is generated in parallel and can be accumulated using a tree MSD adder. The MSD number involves only three digit weights $(1,0,\bar{1})$. So, the partial product generated based on equation (2) will be get in one step (no carry). Table (1) divides all possible combinations of 1-digit MSD multiplication (x_i, y_i) into three groups $(G_1 - G_3)$ depending on PP_i digit output's type.

Table 1: The PP_i Generation of MSD Multiplication

groups	(multiplicand , multiplier)	partial product
G_i	(x_i, y_i)	pp_i
G_1	$(1,1),(\bar{1},\bar{1})$	1
G_2	$(1,0),(0,1),(\bar{1},0),(0,\bar{1}),(0,0)$	0
G_3	$(1,\bar{1}),(\bar{1},1)$	$\bar{1}$

The two M×N×n MSD data arrays A (multiplicand) and B (multiplier) are coded as DDP planes. These DDP planes form the logical formulas of the MSD multipliers in the n channels of the DSS algorithm. Each channel generates one partial product array PP_k, and all channels are operated in parallel. So, the n PP_k s is generated simultaneously. Figure (2) illustrates the total system operation of the MSD parallel multipliers with DDP representation and DSS algorithm. The final result array Z is represented as M×N×(2n+β) DDP planes. The symbol β represents a numerical value {0, 1, 2…}, and it is increased depending on the number of the DSS channels.

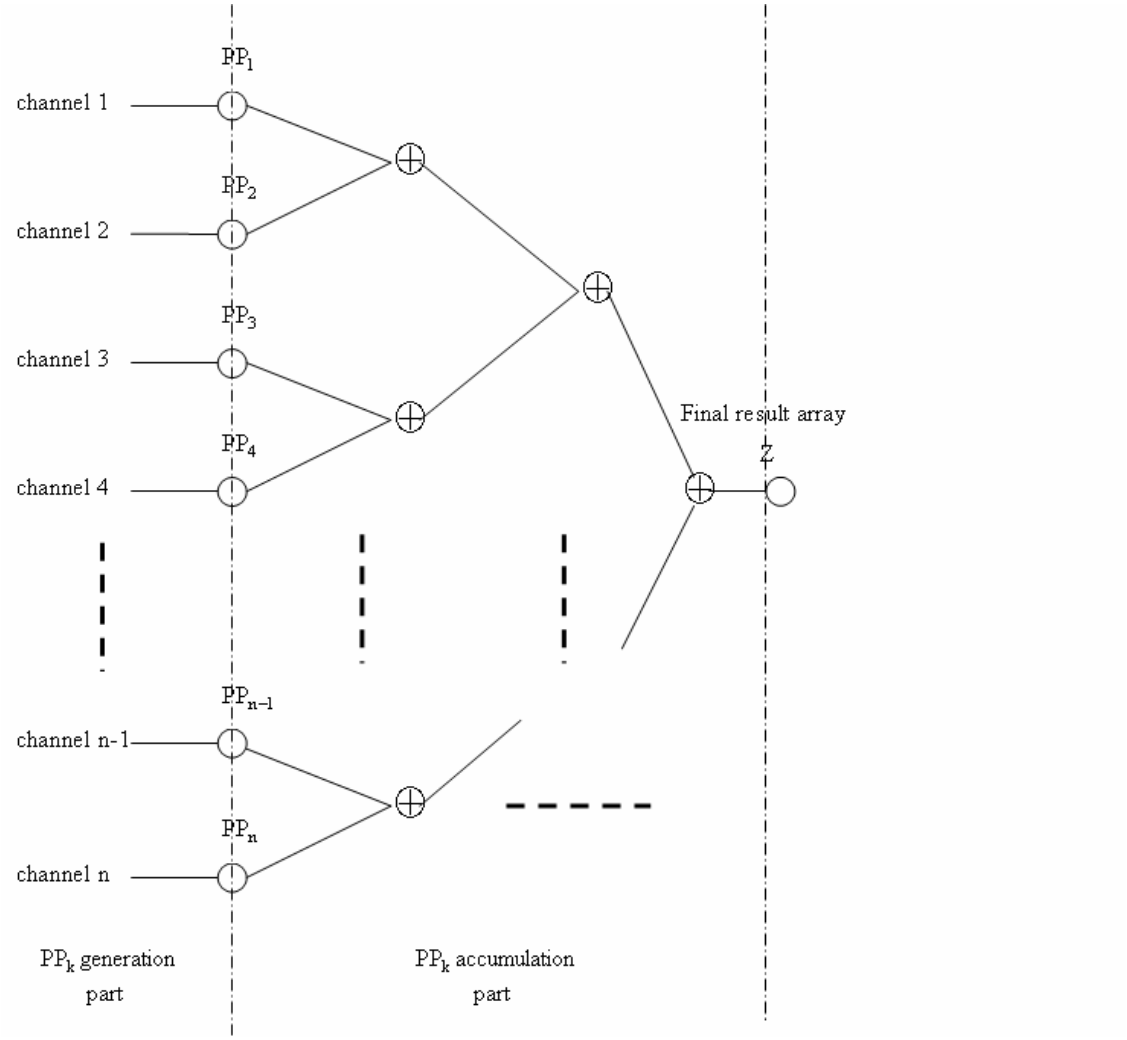


Fig. 2: The Operation System of the MSD Multiplier

Since the MSD multiplication numbers have no carry, the partial products are generated in one step. The first computational rule of Table (1) (G_1) can be expressed by the following conditional statement:

$$IF\ (x_i, y_i) = (1,1)\ \mathrm{OR}\ (\bar{1},\bar{1})\ \ \mathrm{THEN}\ \ pp_i = 1 \quad \textbf{(3)}$$

Rule (3) can be rewritten immediately with DDP planes form as:

$$PP1 = A1 * B1 + A\bar{1} * B\bar{1} \quad \textbf{(4)}$$

where PP1 represents the DDP-1 of the generated partial product array PP, the symbols *, + represent AND and OR operation, respectively.

In same manner, $PP\bar{1}$ and PP0, which represent DDP-$\bar{1}$ and DDP-0 of the partial product array PP, can be generated using the following rules:

$$PP\bar{1} = A1 * B\bar{1} + A\bar{1} * B1 \quad \textbf{(5)}$$

$$PP0 = A0 + \overline{A0} * B0 \quad \textbf{(6)}$$

Now, the three rules (4), (5), and (6) are used in n channels of the DSS algorithm and operated in parallel to generate $3\times n$ $M\times N\times 2n$ DDP planes of n partial products arrays PP_ks. These n PP_ks are summed using the parallel two-step MSD adder with DDP representation, as a tree adder. So, it is used to obtain the final result MSD array Z in three $M\times N\times(2n+\beta)$ DDP planes $Z1, Z0,$ and $Z\bar{1}$.

5. Optical Implementation

The optical MSD adder can be implemented practically using simple optical tools. These are as follows:

BS **BC** : They denote a beam splitter and beam combiner.

: It denotes a mirror.

: It denotes SLM (Spatial Light Modulators), or LDA (Light Detector Array), or Double each CMP (Complement Plane).

: The direction of the light signal emitted by an optical source.

The logical formulas of the single-step PP_ks generation of the parallel MSD array multiplier can be implemented in optical scheme as shown in Fig.(3) with replacing rule (4) by rule (7) below:

$$\bar{PP1} = \overline{PP1 + PP0}$$

This optical system will be used in the n channels of the DSS algorithm to obtain n partial products arrays PP_ks, each one as three DDP planes $PP1$, PP0, and $\bar{PP1}$. Accumulation part will be built to be a tree adder for MSD arrays using the optical parallel two-step MSD array adder n-1 times with $O(\log_2 n)$ stages as shown in Fig.(2). The final results will be three $M \times N \times (2n+\beta)$ DDP planes represent the final result Z array.

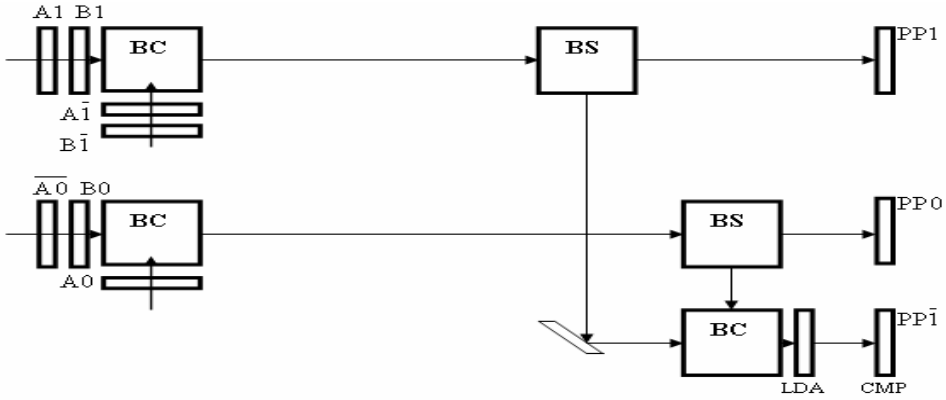


Fig. 3: Optical Implementation of the PP_k Generation Part of the MSD Array Multiplier

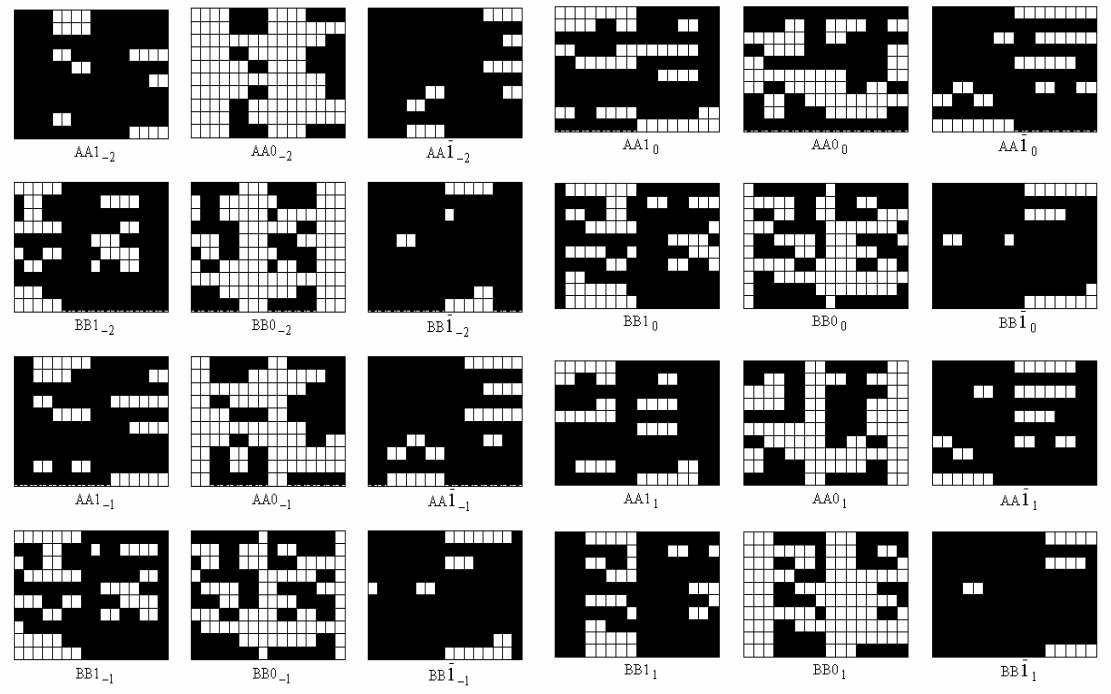
6. Simulation Results

Two $10\times2\times4$ MSD arrays A and B are multiplied by the suggested MSD array multiplier as shown below:

$$A=\begin{bmatrix}15 & -15\\ 13 & 2\\ -1 & -7\\ 9 & 14\\ 7 & -14\\ 0 & 6\\ -4 & -5\\ -10 & 0\\ 11 & 1\\ -15 & 15\end{bmatrix}_{10}=\begin{bmatrix}1111 & \bar{1}\bar{1}\bar{1}\bar{1}\\ 1101 & 0010\\ 000\bar{1} & 0\bar{1}\bar{1}\bar{1}\\ 1001 & 1110\\ 0111 & \bar{1}\bar{1}\bar{1}0\\ 0000 & 0110\\ 0\bar{1}00 & 0\bar{1}0\bar{1}\\ \bar{1}0\bar{1}0 & 0000\\ 1011 & 0001\\ \bar{1}\bar{1}\bar{1}\bar{1} & 1111\end{bmatrix}_{MSD}\qquad B=\begin{bmatrix}15 & -15\\ 2 & 11\\ 10 & -12\\ 7 & 1\\ -9 & 6\\ 13 & 3\\ 2 & 5\\ 8 & 0\\ 14 & -1\\ 15 & -15\end{bmatrix}_{10}=\begin{bmatrix}1111 & \bar{1}\bar{1}\bar{1}\bar{1}\\ 0010 & 1011\\ 1010 & \bar{1}\bar{1}00\\ 0111 & 0001\\ \bar{1}00\bar{1} & 0110\\ 1101 & 0011\\ 0010 & 0101\\ 1000 & 0000\\ 1110 & 000\bar{1}\\ 1111 & \bar{1}\bar{1}\bar{1}\bar{1}\end{bmatrix}_{MSD}$$

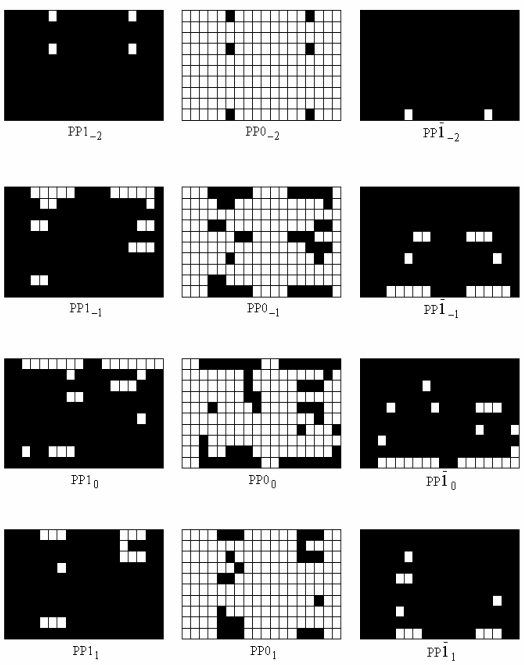
Figure (4-a) shows the $10\times2\times8$ DDP planes of the doubled and shifted versions of the input MSD arrays A and B which will be denoted by AA and BB, respectively. The subscript that belongs to the set {-2, -1, 0, 1} represents the DSS channel number.

Figure (4-b) indicates the DDP planes of the four PP arrays, which are generated by four DSS channels. It is noted that, one zero is padded to the MSB position of each MSD number in the PP_k s' DDP planes to be $10\times2\times9$ pixels resolution, where β=1 in this example. These four PP arrays will be entered to the tree adder to be added in parallel. Figure (4-c) presents the final result Z array.

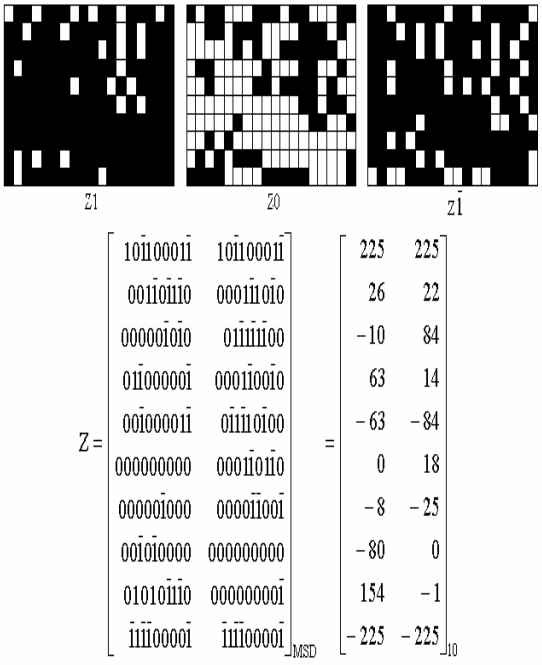


(a)

Fig. (4): Simulation Results of the Parallel Optical MSD Array Multiplier.
(a): The Four Doubled and Shifted Versions of the DDP Planes.
(b): DDP Planes of the Four Generated Partial Product Arrays PPs.
(c): DDP Planes of the Final Results MSD Array Z.



(b)



(c)

7. Conclusions

An optical design for parallel arrays multiplier of the MSD number system is proposed. The presented MSD multiplier combines the DDP representation, parallel MSD multiplication algorithm, and the DSS multiplication algorithm. The MSD multiplier consists of two parts, the partial products generation and the partial products accumulation. The classical optical devices are used to implement the multiplier. A simulation example is discussed to test the performance of the newly design. It is found that the proposed MSD multiplier can be used in optical applications.

References

1- H. Huang, M. Ithon, and T. Yatagai, "Parallel Modified Signed-Digit Computing for Large-Scale Two-Dimensional Data Array," *Opt. Rev.*, Vol.1, No.1, pp.20-23, 1994.

2- S. Zhang and M. A. Karim, "Optical Arithmetic Processing Using Improved Redundant Binary Algorithm," *Opt.Eng.*, Vol.38, No.3, pp.415-421, March 1999.

3- C. D. Capps, R. A. Falk, and T. L. Houk, "Optical Arithmetic Logic Unit Based on Residue Arithmetic and Symbolic Substitution," *Appl. Opt.*, Vol.27, pp.1682-1687, 1988.

4- A. K. Cherri, M. S. Alam, and A. A. S. Awwal, "Optoelectronic Symbolic Substitution Based Conical Modified Signed-Digit Arithmetic," *Opt. Laser Techno.*, Vol.29, pp.151-157, 1997.

5- A. K. Cherri, " Signed-Digit Arithmetic for Optical Computing: Digit Groping and Pixel Assignment for Spatial Coding," *Opt. Eng.*, Vol.38, No.3, pp.422-431, 1999.

6- M. A. Al-Ebbady, "*Parallel Optical Array Arithmetic Using SD Number Systems Based on DDP Representation*," M. Sc. Thesis in Elect. Eng. Dept., Eng. College, University of Basrah, 2004.

7- F. Ahmmed, A. S. Awwal, and G. J. Power, "Synthetic Correlation Modified Signed-Digit Processing," *Opt. Eng.*, Vol.38, No.3, pp.449-455, 1999.

8- H. Huang, M. Itoh, and T. Yatagai, "Optical Scalable Parallel Modified Signed-Digit Algorithms for Large Scale Array Addition and Multiplication Using Digit-Decomposition-Plane Representation," *Opt. Eng.*, Vol.38, No.3, pp.432-440, 1999.

9- A. K. Cherri and M. S. Alam, "Algorithms for Optoelectronic Implementation of Modified Signed-Digit Division, Square Root, Logarithmic, and Exponential Functions," *Appl. Opt.*, Vol.40, pp.1236-1243, 2001.

تصميم ضارب ضوئي باستخدام النظام العددي الرقمي الإشارة المعدل

صباح سليم الشريدة
المعهد التقني في البصرة

المستخلص

قي هذا البحث، تم اقتراح وتصميم ضارب ضوئي متوازي لسلسلة بيانات ثنائية الأبعاد لنظام رقمي أشارة– رقم – معدل تم اقتراحه وتصميمه. التصميم نفذ باستخدام تقنيات مستويات فصل الأرقام وخوارزميات الضرب (المضاعفة–الإزاحة–التجميع). الضارب أعتمد على المعادلات المنطقية التي اشتقت حديثاً من قبل الباحث على أساس خوارزمية الضرب المتوازي. وتم اقتراح البناء الضوئي بالأدوات الضوئية التقليدية لهذا الضارب. تم محاكاة مثال استعراضي لغرض تحقيق التصميم الضوئي المقترح.