Fastest Linearly Independent Arithmetic Transforms over GF(3)

Bogdan J. Falkowski and Cheng Fu
School of Electrical and Electronic Engineering
Block S1, Nanyang Technological University
50 Nanyang Avenue, Singapore 639798

ABSTRACT

In this paper, the family of fastest ternary Linearly Independent Arithmetic transforms, which possesses forward and inverse butterfly diagrams with lowest computational complexity have been identified. This family is recursively defined and has consistent formulas relating forward and inverse transform matrices. Computational costs of the calculation for presented transforms are also discussed.

1. INTRODUCTION

There are different strategies in the analysis and design of nonlinear filtering methods [1]. One of them is to combine spectral techniques based on Walsh and other transforms with theory of logic functions using known concepts like Chow parameters, Boolean derivatives to obtain new tools [3]. Arithmetic transform [1, 5, 8] that is an example of more general class of Linearly Independent Arithmetic (LIA) transforms [7] is frequently used to represent probabilistic and stochastic properties of logic functions. Two fastest binary Linearly Independent Arithmetic (LIA) transforms having most efficient computational complexity among arithmetic transforms, have been identified recently [4]. Before these transforms have been introduced, Arithmetic transform was known as the most computationally efficient transform in standard algebra. While LIA transforms are useful for binary case, due to increased interests in multiple-valued transforms and their various applications [5], these transforms can be modified to such a case as well. The simplest type of such a generalization is ternary case where the transforms are based on Galois Field (3). In this article we propose new classes of fastest ternary Linearly Independent Arithmetic transforms over GF(3) and corresponding ternary polynomial expansions that are generated through recursive matrices. Relations, properties, butterfly diagrams and computational costs of such transforms are also discussed. Similarly to the binary case these new expansions can have applications to describe probabilistic and stochastic behavior of ternary logic functions as well as in statistical analysis of nonlinear filters providing more flexibility than the current methods based on binary case.

2. BASIC DEFINITIONS OF TERNARY LIA TRANSFORMS

Definition 1: Let M_n be an $N \times N$ $(N = 3^n)$ matrix with rows corresponding to minterms and columns corresponding to some switching ternary functions of *n* variables. If the sets of rows are linearly independent with respect to *Ternary Galois Field,* then M_n has only one inverse in GF(3) and is said to be *ternary linearly independent*. If M_n is linearly independent in GF(3), then M_n is a non-singular square matrix with respect to standard arithmetic and has a unique inverse M_n^{-1}.

Definition 2: The ternary LIA expansion for any *n*-variable ternary function $f(\overrightarrow{x_n})$ is given by

$$f(\overrightarrow{x_n}) = \sum_{j=0}^{3^n-1} A_j g_j \,, \tag{1}$$

where g_j is any set of *n*-variable ternary logic functions such that the matrix $M_n = [\overrightarrow{g_0}, \overrightarrow{g_1}, ... \overrightarrow{g_{3^n-1}}]$, $\overrightarrow{g_j}$ represents the truth vector of the ternary functions, $0 \le j \le 3^n - 1$, A_j is the respective coefficient for the particular transform matrix M_n with arithmetic inverse M_n^{-1}, in $\overrightarrow{x_n}$ the Most Significant Bit corresponds to x_n, and the symbol $\sum$ is the standard arithmetic addition.

Definition 3: Let $\vec{F} = [F_0, F_1, ..., F_{3^n-1}]^T$ represent a column vector defining the truth vector of a ternary logic function $f(\overrightarrow{x_n})$ in a natural ternary ordering, and M_n is a ternary LIA transform defined by Definitions 1 and 2, then

$$\vec{F} = M_n \vec{A} \,, \tag{2}$$

and

$$\vec{A} = M_n^{-1} \vec{F} \,, \tag{3}$$

where $\vec{A} = [A_0, A_1, ... A_{3^n-1}]^T$ is the spectral coefficient column vector for the particular ternary LIA transform matrix M_n with arithmetic inverse M_n^{-1}.

Definition 4: Let M_n be a $3^n \times 3^n$ square matrix as specified in Definition 1. Then M_n can be recursively defined by

$$M_n = \begin{bmatrix} M_{n-1}^{(1)} & O_{n-1} & M_{n-1}^{(2)} \\ O_{n-1} & M_{n-1}^{(5)} & O_{n-1} \\ M_{n-1}^{(3)} & O_{n-1} & M_{n-1}^{(4)} \end{bmatrix}, \tag{4}$$

where each submatrix $M_{n-1}^{(j)}$, $j = \{1,2,3,4,5\}$, has a dimension of $3^{n-1} \times 3^{n-1}$, and O_{n-1} is a $3^{n-1} \times 3^{n-1}$ submatrix with all its elements equal to 0.

3. FASTEST TERNARY LIA LOGIC TRANSFORMATION

In this section, two basic classes of ternary LIA logic transformations with fastest and most efficient computational complexity have been found.

The basic forward transformation matrices are constructed from the recursive submatrices O_{n-1}, M_{n-1}, X_{n-1} and $Y_{n-1}^{I(II)}$ where O_{n-1} is a $3^{n-1} \times 3^{n-1}$ submatrix with all it's elements 0, $X_{n-1} = I_{n-1}$ or J_{n-1}, I_{n-1} is a $3^{n-1} \times 3^{n-1}$ identity submatrix and J_{n-1} is a $3^{n-1} \times 3^{n-1}$ reverse-identity submatrix, $Y_{n-1}^{I(II)}$ is a $3^{n-1} \times 3^{n-1}$ submatrix with all its elements 0 except one element located at one corner of each submatrix depending on the location of $Y_{n-1}^{I(II)}$, and the value of the non-zero element is either 1 or 2 denoted by the superscript of Y being I or II, respectively.

Then the first class transforms under this category is defined by

$$M_n = \begin{bmatrix} M_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & X_{n-1} & O_{n-1} \\ Y_{n-1}^{I(II)} & O_{n-1} & M_{n-1} \end{bmatrix} \text{or} \qquad (5)$$

$$M_n = \begin{bmatrix} M_{n-1} & O_{n-1} & Y_{n-1}^{I(II)} \\ O_{n-1} & X_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & M_{n-1} \end{bmatrix} . \qquad (6)$$

The inverse transformation matrix of the first class of ternary LIA transform is given by

$$M_n^{-1} = \begin{bmatrix} M_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & X_{n-1} & O_{n-1} \\ Z_{n-1}^{I(II)} & O_{n-1} & M_{n-1} \end{bmatrix} \text{or} \qquad (7)$$

$$M_n^{-1} = \begin{bmatrix} M_{n-1} & O_{n-1} & Z_{n-1}^{I(II)} \\ O_{n-1} & X_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & M_{n-1} \end{bmatrix} \qquad (8)$$

where $X_{n-1} = I_{n-1}$ or J_{n-1}, and this submatrix is the same in both forward and inverse matrices. $Z_{n-1}^{I(II)}$ is a $3^{n-1} \times 3^{n-1}$ submatrix with all its elements 0 except one element located at one corner of each submatrix depending on the location of $Z_{n-1}^{I(II)}$, and the value of the non-zero element is either -1 or -2 denoted by the superscript of Z being I or II, respectively.

In what follows the properties for the transform from Equations (5) and (7) will be analyzed when $X_{n-1} = I_{n-1}$, $Y_{n-1} = Y_{n-1}^{I}$ and $Z_{n-1} = Z_{n-1}^{I}$. Similar properties can also be derived from Equations (6) and (8) by changing X_{n-1} and Y_{n-1} or Z_{n-1}.

$$A_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}, \quad A_n = \begin{bmatrix} A_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & A_{n-1} \end{bmatrix},$$

$$B_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$B_n = \begin{bmatrix} B_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & B_{n-1} \end{bmatrix}, C_n = \begin{bmatrix} I_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ Z_{n-1} & O_{n-1} & I_{n-1} \end{bmatrix},$$

for n=3, $M_n^{-1} = C_n \bullet B_n \bullet A_n$. $M_n^{=1}$ can be defined in a similar manner for higher n and for forward transform.

Figures 1 and 2 show the forward and inverse butterfly diagrams of the LIA transformation matrices based on Equations (5) and (7) for n=3.

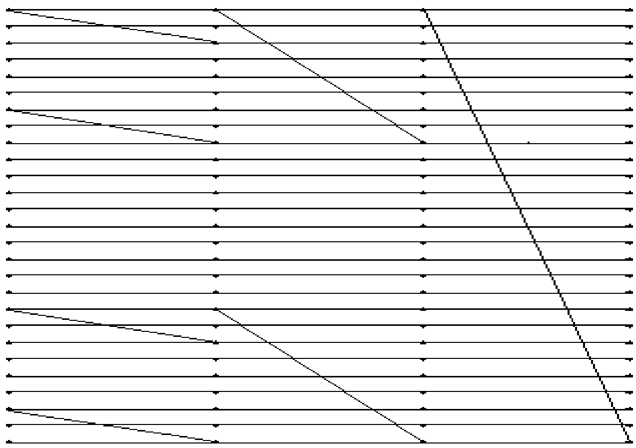

Figure 1. Forward butterfly diagram.

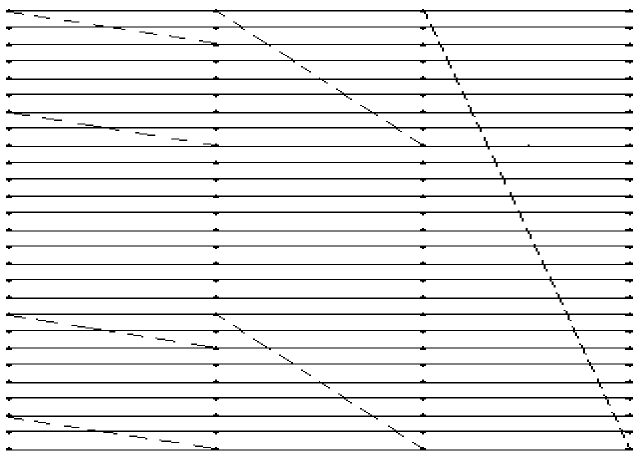

Figure 2. Inverse butterfly diagram.

The solid and broken lines in Figures 1 and 2 represent addition and subtraction respectively.

The computational complexity of a transform depends on the way of the construction of the higher matrices from the basic submatrices. With the introduction of Y_{n-1} and Z_{n-1} in the recursive construction, the introduced matrices are the fastest and most efficient ternary LIA transformation matrices. The number of arithmetic additions/subtractions required to compute the transform for any *n*-variable ternary logic function is $2^n - 1$. The list of all ternary LIA transform matrices that are faster than the known Arithmetic transform for ternary case belonging to the first Class Z1 is shown in the upper part of Table 1.

An example of the forward transformation matrix of the second class of fastest ternary LIA transforms is

$$M_n = \begin{bmatrix} O_{n-1} & O_{n-1} & M_{n-1} \\ O_{n-1} & X_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & Y_{n-1}^{I(II)} \end{bmatrix} \text{ or} \qquad (9)$$

$$M_n = \begin{bmatrix} Y_{n-1}^{I(II)} & O_{n-1} & M_{n-1} \\ O_{n-1} & X_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & O_{n-1} \end{bmatrix} \qquad (10)$$

Equations (9) and (10) can be obtained from Equations (5) and (6), respectively, by grouping the recursive equations in the submatrices vertically and interchanging them in the submatrices horizontally. Similarly, the inverse transformation matrices for Equations (9) and (10) can be derived from Equations (7) and (8), respectively, by grouping the recursive equations in the submatrices horizontally and interchanging them in the submatrices vertically, as shown below:

$$M_n^{-1} = \begin{bmatrix} Z_{n-1}^{I(II)} & O_{n-1} & M_{n-1} \\ O_{n-1} & X_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & O_{n-1} \end{bmatrix} \text{ or} \qquad (11)$$

$$M_n^{-1} = \begin{bmatrix} O_{n-1} & O_{n-1} & M_{n-1} \\ O_{n-1} & X_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & Z_{n-1}^{I(II)} \end{bmatrix} \qquad (12)$$

where the meaning of the submatrices X_{n-1}, Y_{n-1} and Z_{n-1} is the same as explained before.

Example: Let M_n be defined by Equation (9), where $X_{n-1} = J_{n-1}$ and $Y_{n-1} = Y_{n-1}^{I}$. Then M_n^{-1} is defined by Equation (11), where $X_{n-1} = J_{n-1}$, $Z_{n-1} = Z_{n-1}^{I}$ and the corresponding forward and inverse butterfly diagrams are shown in Figure 3 for *n*=2. The first part of the butterfly diagram is a vertical-flipping part. It can be presented as J_n. Then for *n*=2, $M_n = B_n^* \bullet A_n^* \bullet J_n$, where the B_n^* and A_n^* are the same as earlier defined matrices B_n and A_n with the only difference that all the elements -1 become 1. These ternary LIA transforms also belong to the fastest ternary LIA transforms called Class Z2 and have the same lowest computation cost as Class Z1. The list of all ternary LIA transform matrices belonging to the second fastest Class Z2 is shown in the lower part of Table 1. The price of hardware overhead for Class Z2 consists only of the circuitry for permutation of input data. For some ternary logic functions, the Class Z2 can have simpler ternary polynomial expansion based on Definition 2 than the corresponding ones to the Class Z1. Hence it is worthwhile to calculate all the fastest transforms shown in Table 1 for a given ternary function in order to find the spectral coefficient vector with biggest number of zeros that simplifies the final polynomial expansion for such a function.

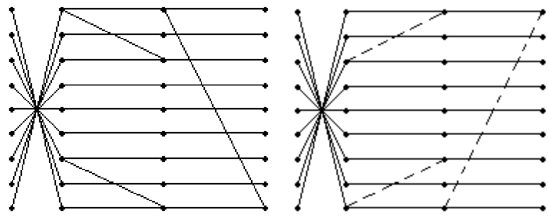

Figure 3. Forward and inverse butterfly diagrams.

4. CONCLUSION

The suitability of a ternary LIA transform in a given application depends not only on the choice of its basis function but also on the existence of efficient ways of its calculation as well as the complexity of its final polynomial expansion. Two classes of transforms discussed in this paper have the least computational complexity among all the ternary LIA transforms. Recursive equations defining the ternary LIA transforms and their corresponding butterfly diagrams are also shown. Similarly to known polynomial expansions based on binary and multiple-valued logic [1, 3, 5, 8], the new expansions can have applications in spectral representations of ternary logic functions, and calculation of their probabilistic and stochastic behavior. They can also be the bases of new ternary word decision diagrams in a manner similar to the ones developed for Linearly Independent Decision Diagrams in [6]. The unified approach to butterfly creation presented here allowed the identification of fastest class for ternary LIA transforms and it may have applications also to other types of transforms [1-3, 5, 8].

5. REFERENCES

[1] S. Agaian, J. Astola, and K. Egiazarian, *Binary Polynomial Transforms and Nonlinear Digital Filters*. New York: Marcel Dekker, 1995.

[2] A. Drygajlo, "Butterfly orthogonal structure for fast transforms, filter banks and wavelets", *Proc. IEEE Int. Conf. Acoustic, Speech, Signal Processing*, San Francisco, California, Vol. 5, pp. 81-84, March 1992.

[3] K. Egiazarian, P. Kuosmanen, and J. Astola, "Boolean derivatives, weighted Chow parameters and selection of stack filters", *IEEE Trans. on Signal Processing*, Vol. 44, No. 7, pp. 1634-1641, July 1996.

[4] B.J. Falkowski and S. Rahardja, "Boolean verification with fastest LIA transforms", *Proc.* 35th *IEEE International Symposium on Circuits and Systems*, Scottsdale, Arizona, Vol. 5, pp. 321-324, May 2002.

[5] G. A. Kukharev, V. P. Shmerko, and E.N. Zaitseva, *Multiple-Valued Data Processing Algorithms and Systolic Processors*. Minsk: Science and Engineering, 1990.

[6] M.A. Perkowski, B.J. Falkowski, M. Chrzanowska-Jeske, and R. Drechsler, "Efficient algorithms for creation of Linearly-Independent Decision Diagrams and their mapping to regular layouts", *VLSI Design*, Vol. 14, No. 1, pp. 35-52, Feb. 2002.

[7] S. Rahardja and B.J. Falkowski, "Fast linearly independent arithmetic expansions", *IEEE Trans. on Computers*, Vol. 48, No. 9, pp. 991-999, Sept. 1999.

[8] R.S. Stankovic, M. Stankovic, and D. Jankovic, *Spectral Transforms in Switching Theory, Definitions and Calculations*. Belgrade: IP Nauka, 1998.

Table 1. List of All Fastest Ternary LIA Transformation Matrices

Class	No	Forward $M_n \Leftrightarrow M_n^{-1}$ Inverse	No	Forward $M_n \Leftrightarrow M_n^{-1}$ Inverse
Z1	1	$\begin{bmatrix} M_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ Y_{n-1}^{I} & O_{n-1} & M_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} M_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ Z_{n-1}^{I} & O_{n-1} & M_{n-1} \end{bmatrix}$	3	$\begin{bmatrix} M_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ Y_{n-1}^{II} & O_{n-1} & M_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} M_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ Z_{n-1}^{II} & O_{n-1} & M_{n-1} \end{bmatrix}$
		$\begin{bmatrix} M_{n-1} & O_{n-1} & Y_{n-1}^{I} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & M_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} M_{n-1} & O_{n-1} & Z_{n-1}^{I} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & M_{n-1} \end{bmatrix}$		$\begin{bmatrix} M_{n-1} & O_{n-1} & Y_{n-1}^{II} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & M_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} M_{n-1} & O_{n-1} & Z_{n-1}^{II} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & M_{n-1} \end{bmatrix}$
	2	$\begin{bmatrix} M_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ Y_{n-1}^{I} & O_{n-1} & M_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} M_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ Z_{n-1}^{I} & O_{n-1} & M_{n-1} \end{bmatrix}$	4	$\begin{bmatrix} M_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ Y_{n-1}^{II} & O_{n-1} & M_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} M_{n-1} & O_{n-1} & O_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ Z_{n-1}^{II} & O_{n-1} & M_{n-1} \end{bmatrix}$
		$\begin{bmatrix} M_{n-1} & O_{n-1} & Y_{n-1}^{I} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & M_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} M_{n-1} & O_{n-1} & Z_{n-1}^{I} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & M_{n-1} \end{bmatrix}$		$\begin{bmatrix} M_{n-1} & O_{n-1} & Y_{n-1}^{II} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & M_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} M_{n-1} & O_{n-1} & Z_{n-1}^{II} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ O_{n-1} & O_{n-1} & M_{n-1} \end{bmatrix}$
Z2	1	$\begin{bmatrix} Y_{n-1}^{I} & O_{n-1} & M_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & O_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} O_{n-1} & O_{n-1} & M_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & Z_{n-1}^{I} \end{bmatrix}$	3	$\begin{bmatrix} Y_{n-1}^{II} & O_{n-1} & M_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & O_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} O_{n-1} & O_{n-1} & M_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & Z_{n-1}^{II} \end{bmatrix}$
		$\begin{bmatrix} O_{n-1} & O_{n-1} & M_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & Y_{n-1}^{I} \end{bmatrix} \Leftrightarrow \begin{bmatrix} Z_{n-1}^{I} & O_{n-1} & M_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & O_{n-1} \end{bmatrix}$		$\begin{bmatrix} O_{n-1} & O_{n-1} & M_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & Y_{n-1}^{II} \end{bmatrix} \Leftrightarrow \begin{bmatrix} Z_{n-1}^{II} & O_{n-1} & M_{n-1} \\ O_{n-1} & J_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & O_{n-1} \end{bmatrix}$
	2	$\begin{bmatrix} Y_{n-1}^{I} & O_{n-1} & M_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & O_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} O_{n-1} & O_{n-1} & M_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & Z_{n-1}^{I} \end{bmatrix}$	4	$\begin{bmatrix} Y_{n-1}^{II} & O_{n-1} & M_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & O_{n-1} \end{bmatrix} \Leftrightarrow \begin{bmatrix} O_{n-1} & O_{n-1} & M_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & Z_{n-1}^{II} \end{bmatrix}$
		$\begin{bmatrix} O_{n-1} & O_{n-1} & M_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & Y_{n-1}^{I} \end{bmatrix} \Leftrightarrow \begin{bmatrix} Z_{n-1}^{I} & O_{n-1} & M_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & O_{n-1} \end{bmatrix}$		$\begin{bmatrix} O_{n-1} & O_{n-1} & M_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & Y_{n-1}^{II} \end{bmatrix} \Leftrightarrow \begin{bmatrix} Z_{n-1}^{II} & O_{n-1} & M_{n-1} \\ O_{n-1} & I_{n-1} & O_{n-1} \\ M_{n-1} & O_{n-1} & O_{n-1} \end{bmatrix}$