SIAM J. COMPUT.
Vol. 20, No. 4, pp. 686-694, August 1991

THE FAST m-TRANSFORM: A FAST COMPUTATION OF CROSS-CORRELATIONS WITH BINARY m-SEQUENCES*

ERICH E. SUTTER†

Abstract. An algorithm is presented for the fast computation of the m-transform, a Hadamard transform intimately related to cross-correlation of analog signals with binary m-sequences. It is shown that m-transforms are in the same Hadamard equivalence class as Walsh–Hadamard transforms and can, thus, be computed by means of the Fast Walsh Transform (FWT) algorithm, preceded and followed by a permutation. The FWT is performed in place in the original data array, while the permutations are executed during loading and reading of this array. Real-time generation of the array addresses for loading and reading adds little to execution time of the FWT. The implementation described here lends itself particularly well to applications in linear and nonlinear systems analysis.

Key words. fast cross-correlation, Hadamard transforms, m-sequences, nonlinear systems analysis, Walsh–Hadamard transforms

AMS(MOS) subject classifications. 65F30, 65C25, 05B20, 42C10

Introduction. The first theoretical work on binary m-sequences was published by Zierler in 1959 [1]. During the following decades their properties were extensively studied [2]. Researchers soon found applications in the fields of systems analysis and identification. The ease and speed with which these pseudorandom sequences could be generated made them very attractive in situations where random white processes are called for. In one of the first such applications, Briggs et al. [3] used them for a linear correlation analysis of process dynamics. Subsequently, numerous applications in nonlinear systems analysis were explored [4], [5], [6]. It was discovered, however, that the randomness properties of m-sequences, as exhibited in higher order auto-correlation functions, are not adequate for emulation of truly random sequences [6], [7], [8]. Detailed studies of their auto-correlation properties [9] ultimately discredited binary m-sequences as test inputs for stochastic white noise of nonlinear systems. The recent introduction of a deterministic technique [10], however, renewed interest in the application of binary m-sequences to systems analysis problems. In this new approach, the derivation of the binary kernels of all orders is reduced to a single cross-correlation of the binary m-sequence test input and the corresponding output. It is necessary, however, that the test extend over a long, complete m-sequence cycle, and that the entire cross-correlation cycle be computed. Because of the often very large size of the arrays, selection of the right algorithm can be very important. Traditionally, such cases called for application of the convolution theorem, requiring execution of three Fast Fourier Transforms (FFTs). In this case, however, where one of the arrays is a binary sequence of a specific class, a much faster computational technique is possible. As shown below, the computation can be reduced to a single Fast Walsh Transform (FWT).

1. Background.

1.1. Hadamard bases and Walsh–Hadamard transforms. A Hadamard matrix is an orthogonal $m \times m$ matrix whose elements are binary $(+1, -1)$. The linear transform mediated by the Hadamard matrix is called a Hadamard transform. Orthogonality requires that the dimensionality be even. The rows or columns of the matrix are orthogonal binary vectors in an m-dimensional vector space.

$$\mathbf{H}\mathbf{H}^T = \mathbf{H}^T\mathbf{H} = m \cdot \mathbf{I}. \tag{1}$$

* Received by the editors August 21, 1989; accepted for publication (in revised form) October 30, 1990.
† Smith-Kettlewell Eye Research Institute, 2232 Webster St., San Francisco, California 94115.

Clearly, multiplication with -1 and permutations of rows and columns cannot affect this property. Any two Hadamard matrices $\mathbf{H}_1$ and $\mathbf{H}_2$ are said to be equivalent if

$$\mathbf{H}_2 = \mathbf{P}_r^T \mathbf{H}_1 \mathbf{P}_c, \tag{2}$$

where $\mathbf{P}_c$ and $\mathbf{P}_r$ are permutation matrices for columns and rows, respectively.

The existence of different equivalence classes has been demonstrated by Hall [11] for the special cases $m = 16$ and $m = 20$.

Of special interest here are Hadamard matrices of order 2^n. For each $m = 2^n$ there exists at least one equivalence class that contains the different representations of the Walsh transform matrix. The Walsh matrices can be defined in various ways leading to different orderings of the Walsh vectors (see, e.g., [12]). The representation considered here is called natural, or Hadamard, ordering. It is achieved by means of a pair of binary registers of length n. These registers, C and R, contain the binary representation of the row number r and column number c, respectively. Let r_i and c_i be the digits of the binary registers C and R, respectively. The Walsh matrix is then given by

$$W(c, r) = (-1)^{q(c,r)} \quad \text{where} \quad q(c, r) = \sum_{i=0}^{n-1} r_i c_i. \tag{3}$$

Each matrix element is thus defined as the parity of the bitwise logic AND between a register r, containing its row number, and a register c, containing the column number.

Example. $n = 3$.

Matrix element $W_{5,6}$:

$$\left.\begin{array}{ll} \textit{row} & r = 5\colon \{r_i\} \to (101) \\ \textit{column} & c = 6\colon \{c_i\} \to (110) \end{array}\right\} \quad \text{AND} \to (001) \to \textit{parity} \to W(5, 6) = -1. \tag{4}$$

The entire matrix is

(5)

c	0	1	2	3	4	5	6	7		
r										
0	+	+	+	+	+	+	+	+	=	$\vec{w}_0$
1	+	−	+	−	+	−	+	−	=	$\vec{w}_1$
2	+	+	−	−	+	+	−	−	=	$\vec{w}_2$
3	+	−	−	+	+	−	−	+	=	$\vec{w}_3$
4	+	+	+	+	−	−	−	−	=	$\vec{w}_4$
5	+	−	+	−	−	+	−	+	=	$\vec{w}_5$
6	+	+	−	−	−	−	+	+	=	$\vec{w}_6$
7	+	−	−	+	−	+	+	−	=	$\vec{w}_7$

All other Walsh–Hadamard matrix representations of the same order are obtained by permutation of the row and column numbers, and are, thus, in the same equivalence class according to equivalence relation (2):

$$W'(c, r) = W(p_2(c), p_1(r)), \tag{6}$$

where p_1 and p_2 are permutation operators.

The Walsh–Hadamard transform, in its natural ordering as defined by (3), can be computed by a simple Fast Walsh Transform (FWT) algorithm [13]. Similar fast algorithms have been developed for various other orderings [12]. According to (2),

any other transform of this equivalence class can be computed by means of the same algorithm preceded and followed by a permutation matrix. Techniques of transformation from one ordering to another have also been developed [14].

1.2. Binary m-sequences. Binary m-sequences, also called maximal length shift register sequences, can be generated by means of digital shift registers with feedback, as shown in Fig. 1 [2]. The content of a certain set of register stages is summed modulo 2 and fed back into the input.

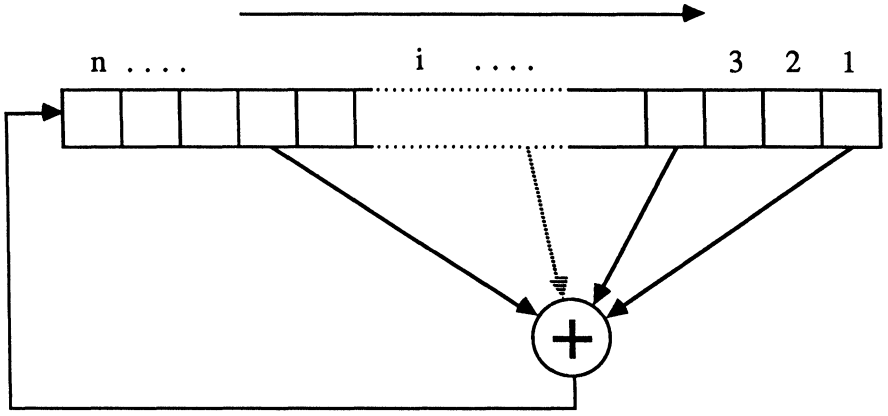


FIG. 1

With properly chosen feedback taps, the register cycles through all possible configurations, except for the all-zero configuration, which is a cycle in itself. For the larger of the two cycles, the binary sequence of 0's and 1's generated by the output of the register is called a maximal length shift register sequence or binary m-sequence. It follows immediately that:

(1) m-sequences have a period of 2^n-1, where n is the number of stages in the generating register.

(2) The number of 1's exceeds the number of 0's by exactly one, i.e.,

$$\sum_{i=0}^{2^n-1} a_i = 2^{n-1}. \tag{7}$$

These sequences have been extensively studied [1], [2], [6], [9].

Let $A_1=\{a_1, a_2, a_3, \cdots\}$ be a binary m-sequence with period 2^n-1 and $A_i=\{a_i, a_{i+1}, a_{i+2}, \cdots\}$ be the sequence in all its cyclical shifts. Let $A_0=\{0, 0, 0, \cdots\}$.

$$\begin{array}{lcccccccc} A_0 & = & 0 & 0 & 0 & \cdot & \cdot & 0 & \\ A_1 & = & a_1 & a_2 & a_3 & \cdot & \cdot & a_N & \\ A_2 & = & a_2 & a_3 & a_4 & \cdot & \cdot & a_1 & \\ A_3 & = & a_3 & a_4 & a_5 & \cdot & \cdot & a_2 & \text{where } N=2^n-1. \\ \cdot & & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \\ \cdot & & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \\ A_N & = & a_{N-1} & \cdot & \cdot & \cdot & \cdot & a_{N-1} & \end{array} \tag{8}$$

The sequences $A_0, A_1, A_2, \cdots, A_p$ form an Abelian group with respect to the operation of elementwise addition modulo 2. Specifically

$$A_i+A_0=A_i, \qquad A_i+A_i=A_0 \quad \text{for any } i \quad \text{and} \quad A_i+A_j=A_{k(i,j)} \quad \text{for } i\neq j\neq 0. \tag{9}$$

The proof follows directly from the recurrence relation defined by Fig. 1 (see [2, p. 44]).

2. Binary m-transform.

DEFINITION. Let $\{M_i\}$ be the set of sequences obtained from the sequences $\{A_i\}$ by replacing all the 0's by 1's and the 1's by -1's, and adding a zeroth element of 1 to each A_i.

$$\mathbf{M} = \begin{bmatrix} M_0 \\ M_1 \\ M_2 \\ M_3 \\ \cdot \\ \cdot \\ M_N \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & m_1 & m_2 & \cdot & \cdot & m_N \\ 1 & m_2 & m_3 & \cdot & \cdot & m_1 \\ 1 & m_3 & m_4 & \cdot & \cdot & m_2 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & m_N & m_1 & \cdot & \cdot & m_{N-1} \end{bmatrix} \qquad N = 2^n - 1. \tag{10}$$

The transform defined by matrix (10) will be called m-transform.

Through the substitution $0 \to 1, 1 \to -1$, the operation of addition modulo 2 becomes multiplication:

$$\begin{array}{c|cc} & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 1 & 0 \end{array} \qquad \begin{array}{c} 0 \to 1 \\ 1 \to -1 \end{array} \qquad \begin{array}{c|cc} & 1 & -1 \\ \hline 1 & 1 & -1 \\ -1 & -1 & 1 \end{array} . \tag{11}$$

With the completion of the rows with a zeroth element of 1, the zero cycle is included in each row. With it, the matrix becomes symmetrical. From (7) and (9) it follows that the rows M_r form an orthogonal basis

$$\vec{M}_r \cdot \vec{M}_s = \begin{cases} 2^n & \text{for } i = k \\ 0 & \text{for } i \neq k, \end{cases} \tag{12}$$

and that $\mathbf{M}$ is a symmetric orthogonal matrix

$$\mathbf{M}^T\mathbf{M} = \mathbf{M}\mathbf{M} = 2^n \cdot \mathbf{I} \quad \text{where } \mathbf{I} \text{ is the identity.} \tag{13}$$

With the above substitution, the rows M_r now form an Abelian group with respect to elementwise multiplication.

As a binary orthogonal matrix, $\mathbf{M}$ is a Hadamard matrix.

Note that the cross-correlation of a data array of $2^n - 1$ real numbers with a binary m-sequence (elements $+1$ and -1) is the sequence of elements 1 to $2^n - 1$ of the m-transform if the data array is supplemented with a zeroth element of 0.

THEOREM. *All Walsh and* m*-transform matrices of dimension* 2^n *are in the same equivalence class of Hadamard matrices.*

Proof. Each row M_r of the matrix $\mathbf{M}$ can be obtained as the parity of a particular collection of taps t_r on its n stages during a single cycle through the configurations of the generating register. This can be seen as follows. For the first n rows, t_r is just a single tap on the rth stage. For row M_{n+1}, a single tap on stage $n+1$ would be needed. According to Fig. 1, this tap is equivalent to the configuration t_{n+1} of the feedback taps (see Fig. 2).

For $M_{n+2}, M_{n+3}, \cdots$, the feedback taps have to be shifted left one stage each time. Whenever a tap is shifted off the left end of the register, it is replaced by the feedback tap configuration. If, in this process, a new tap coincides with an already existing tap, this tap position contributes even parity and can, thus, be dropped.

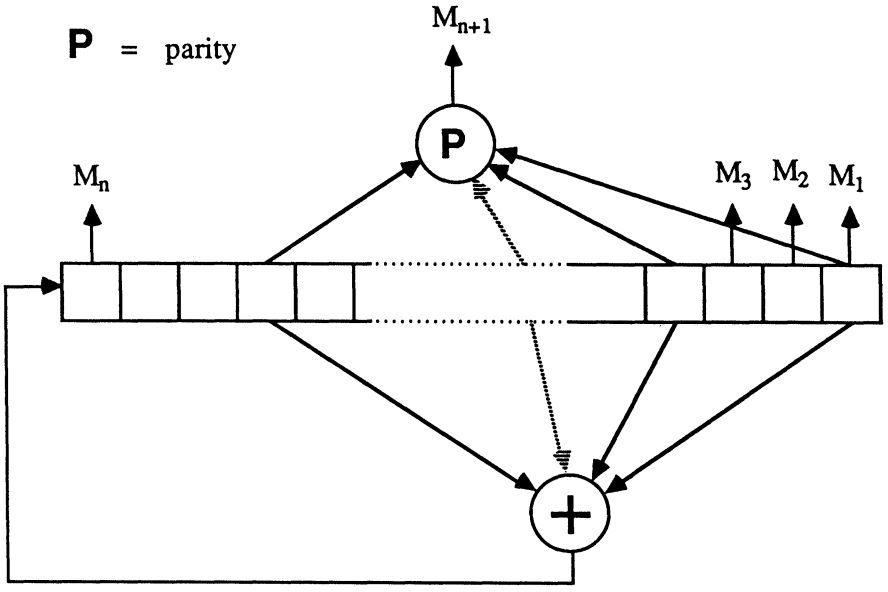


FIG. 2

The generation of the tap configurations t_r derived above can be implemented in a separate shift register of the same length n. It will be called tap register T. The 1's in this register signify the position of taps. The tap register is initialized with a right-justified 1 for the first row and shifted left for consecutive rows. The output of the register is added (modulo 2) to the stages where the generating register has its feedback taps (see top of Fig. 3).

The tap configurations sequentially generated by this method will produce consecutive rows of the matrix **M**. Similarly, for a fixed configuration of the generating register, the set of all tap configurations yields a column. With each shift of the generating register, a new column is generated.

So, to get to the matrix element $M_{c,r}$, the tap and generating registers are advanced by $(c-1)$ and $(r-1)$ steps, respectively. $M_{c,r}$ is then the parity of the bitwise logic AND (tap operation) of the two registers. For the generation of the zeroth row (zeroth column), the column (row) register is initialized with all 0's.

Note that a row generated by a collection t_r of taps is simply the bitwise product of all the rows generated by the individual taps in the collection. It follows that the Abelian group of rows is generated by rows M_1 through M_n in the same way as the Walsh basis is generated by the Rademacher functions [15].

This derivation of the m-transform matrix serves as another definition of the m-transform in terms of the generating and tap registers.

$$M(r, c) = (-1)^{q(r,c)} \quad \text{where } q(r, c) = \sum_{i=0}^{n-1} t_i(r) g_i(c), \tag{14}$$

where $g_i(c)$ and $t_i(r)$ are cth and rth bit configurations generated by the registers G and T, respectively, and $g_i(0) = t_i(0) = 0$. For the generation of the nontrivial cycle, the registers G and T can be initialized with any binary number not equal to 0, depending on the chosen starting point of the m-sequence. In applications to deterministic nonlinear analysis, the register T takes on a special function that determines the initialization [10].

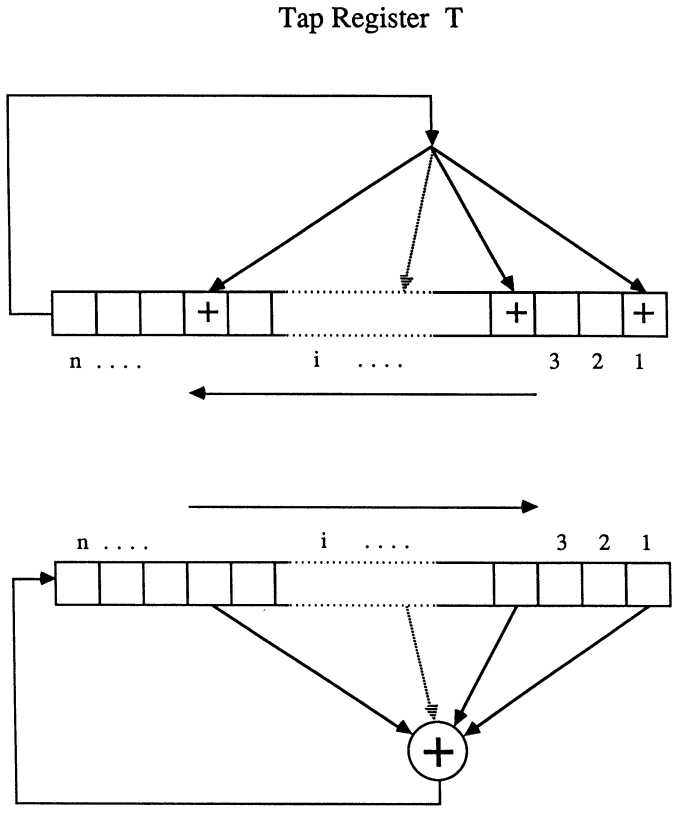


FIG. 3

This derivation of the matrix **M** matches the definition of the natural Walsh transform matrix **W** (3), except in the sequence in which the $2^n - 1$ register configurations are being generated. It, thus, follows that

$$M(r, c) = W(t(r), g(c)), \tag{15}$$

i.e., m-transforms and Walsh transforms belong to the same equivalence class of Hadamard transforms.

2.1. Fast computation of m-transforms. The equivalence between Walsh and m-transforms makes it possible to compute m-transforms by means of the Fast Walsh Transform (FWT) algorithm using natural (Hadamard) ordering. The permutations $r \to g(r)$ and $c \to t(c)$ preceding and following the FWT do not add significantly to the computation times. This section discusses efficient execution of these permutations.

In matrix notation, (15) can be written as

$$\mathbf{M} = \mathbf{P}_{r\to t}^T \mathbf{W} \mathbf{P}_{g\to c}, \tag{16}$$

where $\mathbf{P}_{c\to g}$ is the permutation matrix $c \to g(c)$ defined by the generating register, and $\mathbf{P}_{r\to t}$ is the permutation matrix $r \to t(r)$ defined by the tap register.

From the symmetry of the matrices **M** and **W**, it follows that

$$\mathbf{M}^T = \mathbf{P}_{c\to g}^T \mathbf{W}^T \mathbf{P}_{r\to t} = \mathbf{P}_{c\to g}^T \mathbf{W} \mathbf{P}_{r\to t} = \mathbf{M}, \tag{17}$$

i.e., the roles of the tap and generating registers can be interchanged. The sequence of operations chosen here has important advantages in applications to nonlinear systems analysis [10].

Fig. 4 illustrates the relationship between the registers and matrices for the case $n=3$.

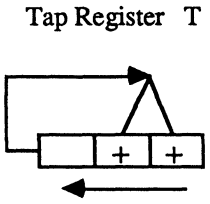
The permutation $\mathbf{P}_{c\to g}$ is equivalent to loading data point number c at the cth binary array address generated by register G. The permutation $\mathbf{P}^T_{r\to t}$, after execution of FWT, is equivalent to reading point number r of the m-transform from the rth binary array address generated by the tap register T.

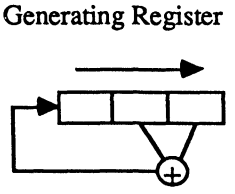
It is, of course, possible to compute the two address arrays for a particular m-transform ahead of time. However, the generation of the addresses is considerably faster than loading from a conventional storage medium, particularly if the instruction set of processors contains the register operation of bitwise exclusive OR (EXOR). Consecutive configurations of the register T (addresses for retrieval of the m-transform) can be generated at high speed using the following simple operations. (1) Shift register T left by one. (2) If bit $n+1$ of register T is set, then $T \equiv T$ EXOR C, where the register C contains 1's bit position $(n+1)$, as well as the position of feedback taps and 0's everywhere else.

Since each bit of register T cycles through the m-sequence, the same code can be used to generate consecutive addresses for loading of the data points. The output of T is simply shifted from the left through the n least significant bit positions of another register G.

3. Discussion and conclusions. Among the Hadamard sets, the m-sequence bases are unique in that they share the following two important properties. First, all m-sequence basis vectors are related to one another by cyclical shifts of the elements 1 through 2^n-1. Second, the basis vectors form an Abelian group with respect to elementwise multiplication. These properties make them extremely valuable as test inputs for the analysis of nonlinear systems. They make it possible to reduce the data analysis to a single cross-correlation between the system response and the m-sequence input [10]. Since these two arrays can be very large, efficient computation of the cross-correlation cycle is of great importance. The technique presented here reduces the computation to a single Fast Walsh Transform that is performed in-place. The reduction in computation time, compared to the traditional method employing FFTs and the convolution theorem, is considerable. Three FFTs and an array multiplication are replaced by a single Fast Walsh Transform (FWT) preceded and followed by simple and highly efficient routines for loading and unloading of the data array. The loading and unloading routines require little or no overhead, depending on the application. The FWT algorithm is basically an abbreviated FFT, requiring no sine table and no multiplications. An exact quantitative measure of the speed advantage of the FWT over the FFT cannot be given, since it depends on the available hardware. In most cases, it can easily be implemented in integer, rather than floating point format without loss of accuracy. In a test on a Macintosh II computer using the 68081 math co-processor, the FWT was faster than a real FFT by a factor of six. Both transforms used in the comparison were based on the Cooley–Tukey algorithm. Since the computation of the cross-correlation cycle requires only one FWT, one can expect an overall speed advantage of a factor between 15 and 30. On systems without hardware multiplier, the savings are significantly larger.

Consider also that the Fast m-transform makes use of the fact that the m-sequence is completely determined by the length of the generating register and the configuration





r	$t_1(r)$	t
0	000	0
1	001	1
2	010	2
3	100	4
4	011	3
5	110	6
6	111	7
7	101	5

zero cycle

m-sequence cycle

c	$g_1(c)$	g
0	000	0
1	001	1
2	100	4
3	010	2
4	101	5
5	110	6
6	111	7
7	011	3

$$\begin{bmatrix} 1&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&1&0&0&0&0\\ 0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&1\\ 0&0&0&0&0&1&0&0 \end{bmatrix} * \begin{bmatrix} +&+&+&+&+&+&+&+\\ +&-&+&-&+&-&+&-\\ +&+&-&-&+&+&-&-\\ +&-&-&+&+&-&-&+\\ +&+&+&+&-&-&-&-\\ +&-&+&-&-&+&-&+\\ +&+&-&-&-&-&+&+\\ +&-&-&+&-&+&+&- \end{bmatrix} * \begin{bmatrix} 1&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0\\ 0&0&0&1&0&0&0&0\\ 0&0&0&0&0&0&0&1\\ 0&0&1&0&0&0&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&1&0 \end{bmatrix}$$

$$[P_{r\to t}]^T \qquad [WT] \qquad [P_{c\to g}]$$

$$= \begin{bmatrix} +&+&+&+&+&+&+&+\\ +&-&+&+&-&+&-&-\\ +&+&+&-&+&-&-&-\\ +&+&-&+&-&-&-&+\\ +&-&+&-&-&-&+&+\\ +&+&-&-&-&+&+&-\\ +&-&-&-&+&+&-&+\\ +&-&-&+&+&-&+&- \end{bmatrix}$$

$$[MT]$$

FIG. 4

of the feedback taps. No memory allocation is necessary for storage of the m-sequence. This greatly facilitates implementation of the cross-correlation of large arrays on microcomputers.

REFERENCES

[1] N. ZIERLER, *Linear recurring sequences*, J. Soc. Indust. Appl. Math., 7 (1959), pp. 31-49.
[2] S. W. GOLOMB, *Shift Register Sequences*, Aegean Park Press, Laguna Hills, CA, 1982.
[3] P. A. N. BRIGGS, P. H. HAMMOND, M. T. G. HUGHES, AND G. O. PLUMB, *Correlation analysis of process dynamics using pseudo-random binary test perturbations*, Proc. Inst. Mech. Engrg., 179 (1964-65), pp. 37-50.
[4] E. P. GYFTOPOULOS AND R. J. HOOPER, *Signals for transfer-function measurements in nonlinear systems*, in Noise and Nuclear Systems, USAEC Symposium Series 4, TID-7679, United States Atomic Energy Commission, 1964, pp. 335-345.
[5] R. J. HOOPER AND E. P. GYFTOPOULOS, *On the measurement of characteristic kernels of a class of nonlinear systems*, in Neutron Noise, Waves and Pulse Propagation, USAEC Conference Report 660206, United States Atomic Energy Commission, 1967, pp. 343-356.
[6] H. R. SIMPSON, *Statistical properties of a class of pseudorandom sequences*, Proc. IEE-E, 103 (1966), pp. 2075-2080.
[7] N. REAM, *Nonlinear identification using inverse-repeat* m-*sequences*, Proc. IEE-E, 117 (1966), pp. 213-218.
[8] C. SWERUP, *On the choice of noise for the analysis of the peripheral auditory system*, Biol. Cybernet., 29 (1978), pp. 97-104.
[9] H. A. BARKER AND T. PRADISTHAYON, *High-order autocorrelation functions of pseudorandom signals based on* m-*sequences*, Proc. IEE-E, 117 (1970), pp. 1857-1863.
[10] E. E. SUTTER, *A practical non-stochastic approach to nonlinear time-domain analysis*, in Advanced Methods of Physiological Systems Modelling, Vol. 1, Biomedical Simulations Resource, Department of Biomedical Engineering, University of Southern California, Los Angeles, CA, 1987.
[11] M. HALL, JR., *Hadamard matrices of order* 16, Lett. Propuls. Lab. Res. M., Vol. 36-10, Jet Propulsion Laboratory, Pasadena, CA, 1961, pp. 21-26.
[12] D. F. ELLIOTT AND K. R. RAO, *Fast Transforms: Algorithms, Analysis, Applications*, Academic Press, New York, 1982, p. 301.
[13] K. W. HENDERSON, *Comment on Computation of the fast Walsh-Fourier transform*, IEEE Trans. Comput., 19 (1970), pp. 50-51.
[14] B. J. FINO AND V. R. ALGAZI, *Unified matrix treatment of the fast Walsh-Hadamard transform*, IEEE Trans. Comput., 25 (1976), pp. 1142-1146.
[15] H. RADEMACHER, *Einige Sätze von allgemeinen Orthogonalfunktionen*, Math. Ann., 87 (1922), pp. 122-138.