J. of Mult.-Valued Logic & Soft Computing, Vol. 11, pp. 185-211
Reprints available directly from the publisher
Photocopying permitted by license only

Properties of Logic Functions in Spectral Domain of Sign Hadamard-Haar Transform

BOGDAN J. FALKOWSKI AND SHIXING YAN

School of Electrical and Electronic Engineering, Nanyang Technological University, Block S1, 50 Nanyang Avenue, Singapore 639798

(Recommended for publication by Radomir S. Stankovic)

A non-linear transform, called 'Sign Hadamard-Haar transform' is introduced. The transform is unique and converts ternary vectors into ternary spectral domain. Recursive definition and fast transform have been developed. Formulae to represent Sign Hadamard-Haar spectra for basic logic functions and their variables for two different codings are shown. New transform is extremely effective in terms of computational costs when compared with known ternary transforms. Advantage of new transform over known Sign Walsh transform in ternary communication system is also discussed.

Keywords: Spectral techniques, Discrete transforms, Nonlinear transforms, Quantized transforms, Ternary Reed-Muller transform, Ternary digital transformation, Ternary communication system, Logic functions, Cyclic and dyadic addition.

1. INTRODUCTION

In many applications of computer engineering and science, where logic functions need to be analyzed or synthesized, it is useful to transform such functions to the corresponding spectral domain that provides various new insights into solving some important problems. The most popular

*Corresponding Author: E-mail: efalkowski@ntu.edu.sg, Tel: +65 67904521, Fax: +65 67912687

transforms used in the design of logic networks are Walsh, Haar, Chrestenson, Reed-Muller and arithmetic transforms [1, 6, 16-20, 24-26, 28, 29, 31-36]. The renewed interest in applications of spectral methods in logic synthesis is caused by their excellent testability design and the development of efficient methods of calculating different spectra from reduced representations of logic functions in the form of arrays of cubes or decision diagrams [6, 16, 17, 19, 20, 28, 29, 31-34]. A lot of research work was also done on reduction of decision diagrams using various techniques combining frequently logic and its Walsh spectral domain such as shifting using spectral linear transformations with variable reordering techniques [22, 31] and even more general spectral autocorrelation approach [21, 31]. These recent developments have an enormous influence on the practical application of spectral methods in binary and multiple-valued logic design. Most practical logic functions can now be represented and transformed using representations that do not grow exponentially.

The most popular ternary transform is Reed-Muller transform over GF(3) [15]. It operates on ternary logic functions and provides 3^n ternary spectra for a given ternary function. Another family of invertible nonlinear transforms, which uniquely map ternary logic functions into ternary transform space, are sign transforms. The first transform under the name of 'sign transform' was based on Walsh functions [1, 17-20, 31, 33-35, 37] and is known as Sign Walsh transform [2, 5]. An efficient method to calculate sign Walsh spectra from disjoint cube representation of logic functions was shown in [7]. Another sign transform based on Haar functions and called Sign Haar transform has been developed [9]. The common feature of sign transforms is that in all the definitions the sign function is used as the quantizer. It has been found that sign transforms play a similar role in the design of ternary output digital systems [2, 5, 34] as the Reed-Muller transform does for the design of ESOPs [1, 28, 29, 33-35]. The fundamental advantage of sign transforms as well as ternary Reed-Muller transform over other transforms in different algebra used in ternary logic design is that the memory required to store functional and spectral data is exactly the same since both of them operate on ternary values. It is in stark contrast to traditional (Walsh, Haar, Chrestenson, arithmetic) spectra of logic functions where signs together with magnitudes have to be stored in the spectral domain.

In this paper, a new transform called Sign Hadamard-Haar transform is developed. This novel transform is based on the combination of Hadamard and Haar butterfly diagrams together with operation of the quantizer. The basis functions of new transform are based on rationalized Hadamard-Haar functions [27] and therefore this new quantized transform is called 'Sign Hadamard-Haar transform'. The new sign transform is unique, non-linear and invertible and has all the advantages of known Sign Haar [9] and Sign Walsh transforms [2, 5, 34]. It should be noticed that Sign Walsh [2, 5, 34] and Sign Haar [8] transforms have been used in logic function decomposition and testing of logical circuits so similar applications can be expected for the new transforms as well.

In this article, we present detailed properties of new Sign Hadamard-Haar transform. The basic definitions for this transform have been given here. Besides applications in ternary logic design, a new Hadamard-Haar transform can be used when there is a need for a unique coding of ternary vectors into the special domain of the same dimensions. In this article, we are showing one such application in the form of ternary communication system and compare computational advantages of our new transform over known Sign Walsh transform. Another area in which our new transform could be used is development of cryptographic functions that need to be immune to input transformations [30]. Much of this work involves the study of properties of Boolean functions. They should not be linear or affine, nor even close to linear or affine. There should be a balance of zeros and ones, and no correlation between different combinations of bits. The output bits should behave independently when any single input bit is complemented. Additionally, in the last years differential power analysis of cryptosystems has attracted much attention. It tries to recover secret keys by monitoring power signals cryptographic devices. Since in the usual doubling and addition scheme it can easily be detected whether an addition and a doubling or a doubling alone takes place, the binary representation can be guessed. But if +1 and -1 are used, a potential attacker can gain less information [4]. The recent idea to represent balanced binary representations which yield good results with substitutions which yield representations of smaller weights was based on elliptic curves also using digits 0, +1, and -1 [3, 4, 23]. Hence one of the possible applications of our new transform would be security coding in cryptographic systems with Sign Hadamard-γ Haar-χ transform which means application of Hadamard and Haar butterfly diagrams in fast Sign Hadamard-Haar transformation γ and χ times, respectively and for $\gamma = 1$ and $\chi = n - 1$ it is simply Sign Hadamard-Haar transform of size n.

Different quantized transforms have been known to be extremely effective both in terms of memory requirements and processing time and hence very advantageous in the design of ternary logic vectors. There exists many methods to calculate well known ternary Reed-Muller transform such as direct matrix calculation, fast transform calculation, Gray code ordered fast transform calculation, column and row method calculation that are given here for comparison purpose with our new transform [7, 11, 13]. However even the two fastest methods based on row and column to calculate ternary Reed-Muller transform require increasing number of ternary addition and multiplication over GF(3), what is not necessary for new transform as we apply sign transform at each level of butterfly diagram and the sign operation is simpler than the GF(3) operations. An important property of Sign Hadamard-Haar transform is that during its calculation it does not require any ternary multiplication and addition that is necessary for the ternary Reed-Muller transform over GF(3) even though the computer memory required to store functional and spectral data is exactly the same for both transforms since both of them operate on ternary values. It is in high contrast to rationalized Hadamard-Haar spectrum [27] where signs together with magnitudes have to be stored in spectral domain that increases significantly requirements for storage space in spectral domain versus functional when only ternary data are considered.

The structure of this article is as follows. For comparison purpose, Section 2 covers basic definitions of ternary Reed-Muller and Sign Walsh transform as well as introduces novel Sign Hadamard-Haar transform. Section 3 discusses basic properties of logic functions in Sign Hadamard-Haar spectra, while Section 4 discusses calculation of the new transform through matrix multiplication. Section 5 shows computational advantages of our new transform over Sign Walsh and ternary Reed-Muller transform in terms of calculations required to compute corresponding transforms. Application of Sign Hadamard-Haar transform in ternary communication system and its advantage over Sign Walsh transform is discussed in Section 6 while Section 7 concludes the paper.

2. DEFINITIONS OF TERNARY REED-MULLER, SIGN WALSH AND SIGN HADAMARD-HAAR TRANSFORMS

Definition 1 *The ternary Reed-Muller transform is defined by the following equations [15]:*

For $n=1$, where n is the number of variables,

$$TRM_1 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 1 \end{bmatrix}, \quad TRM_1^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 1 \\ 2 & 2 & 2 \end{bmatrix},$$

and for arbitrary n,

$$TRM_n = \overset{n-1}{\otimes} TRM_1 = \overset{n-1}{\otimes} \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 1 \end{bmatrix}, \tag{1}$$

$$TRM_n^{-1} = \overset{n-1}{\otimes} TRM_1^{-1} = \overset{n-1}{\otimes} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 1 \\ 2 & 2 & 2 \end{bmatrix} \tag{2}$$

where "$\overset{n-1}{\otimes}$" represents Kronecker product applied n-1 times to either the matrix TRM_1 or TRM_1^{-1} with additions and multiplications over GF(3). Table 1 shows the addition and multiplication operations over GF(3).

TABLE 1
Addition and multiplication rules over GF(3).

+	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

×	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

In contrary to Reed-Muller transform over GF(3), two types of coding are used for logic functions when different spectra of such functions are calculated. The truth vector for *S*-coding is represented in the following way: the *true* minterms (minterms for which Boolean function has logical values 1) are denoted by -1, *false* minterms (minterms for which Boolean function has arbitrary logical values 1) by +1, and *do not care* (*DC*) minterms (minterms for which Boolean function can have arbitrary logical values 0 or 1) by 0. Hence binary vectors formed of only {+1,-1} represent logical values of completely specified Boolean functions, and formed of {+1, 0, -1} the values of incompletely specified Boolean functions. In the continuation, to shorten the notation, functional and spectral data will be represented by either {+,-} or {+, 0, -}. The data in functional domain can be arbitrary binary/ternary vectors or *S*-coded completely (binary) or incompletely (ternary) specified Boolean functions. The following symbols will be used, let R_1 = {+,-}, R_2 = {+, 0, -}, R_l^n means *n*-space Cartesian product of a set R_l (l = 1, 2). In the *R*-coding the truth vector is represented by its original values: 0 for *false* minterms and 1 for *true* minterms. The *DC* minterms are represented by 0.5.

Definition 2 *An n-variable S-coded completely specified Boolean function is the mapping* $m_1 : R_1^n \rightarrow R_1$.

Definition 3 *An n-variable S-coded incompletely specified Boolean function is the mapping* $m_2 : R_1^n \rightarrow R_2$.

Definition 4 *An invertible Sign Hadamard-Haar transform hh and its inverse transform* hh^{-1} *are the mappings hh:* $R_2^N \rightarrow R_{2(hh)}^N$ *and* hh^{-1}*:* $R_{2(hh)}^N \rightarrow R_2^N$*, where* $N = 2^n$.

In the above equations, symbol $R_{2(hh)}^N$ represents a set with the elements from R_2^N permuted by the mapping *hh* of all the elements of the set R_2^N. When only completely specified Boolean functions are considered, the symbol R_2^N is replaced with R_1^N and $R_{2(hh)}^N$ with $R_{1(hh)}^N$ where the latter represents a proper subset of set $R_{2(hh)}^N$ generated by the *hh* mapping of all the elements of set R_1^N. In order to obtain Sign Hadamard-Haar spectrum *hh* (an element of set $R_{2(hh)}^N$), and its inverse (a corresponding element of the original data set R_2^N), the results of each fast forward or inverse Hadamard-Haar butterfly block are quantized first. In the above equations, the cardinality of the original data set R_2^N and its transformed spectrum $R_{2(hh)}^N$ is equal to 3^N. When some permutation is performed on the elements of set R_2^N the same permutation happens to the elements in $R_{2(hh)}^N$ spectrum of the original set.

The following symbols are used: Let $\vec{x}_n = \{x_n, x_{n-1}, \cdots, x_p, \cdots, x_2, x_1\}$, and $\vec{\omega}_n = \{\omega_n, \omega_{n-1}, \cdots, \omega_p, \cdots \omega_2, \omega_1\}$ be *n*-tuples over GF(2). The symbol x_p stands for a data variable, α_p represents a Sign Walsh transform variable, and ω_p a Sign Hadamard-Haar transform variable, *p* is an integer and $1 \le p \le n$. Let

$\vec{F}=[F_0,F_1,...,F_j,...,F_{N-2},F_{N-1}]$ be a ternary vector. For example, it can be the *S*-coded truth vector of f: $(0, 1)^n \rightarrow (-1, 0, 1)$ where the value of $F_j\,(1 \le j < N)$ is given by $F(\vec{x}_n)$ when $\sum_{p=1}^{n} x_p 2^{p-1}=j$.

Let $\overrightarrow{W}_F=[w_0,w_1...,w_j,...,w_{N-2},w_{N-1}]$ and $\overrightarrow{HH}_F=[hh_0,hh_1...,hh_j,...,hh_{N-2},hh_{N-1}]$ be the vector corresponding to Sign Walsh spectrum of $\vec{F}$ and Sign Hadamard-Haar spectrum of $\vec{F}$, accordingly. The value of w_j $(0 \le j < N)$ is given by $\overrightarrow{W}_F(\alpha)$ when $\sum_{p=1}^{n} \alpha_p 2^{p-1}=j$. The value of hh_j $(0 \le j < N)$ is given by $\overrightarrow{HH}_F(\omega)$ when $\sum_{p=1}^{n} \omega_p 2^{p-1}=j$. Let $\vec{O}_i$ represent the vector of i zeros, $1 \le i < n$. Let the symbol $\oplus_c$ represent cyclic addition, the symbol $\oplus_d$ represent dyadic addition, and the symbol $\wedge$ represent bit-by-bit logical AND.

When the above operations are applied to two vectors $\vec{A}_l$ and $\vec{B}_k$, $1 \le l < k$, l and k are two different integer numbers, they result in the vector $\vec{C}_k$ of the length k. Only l elements of $\vec{B}_k$ and all elements of $\vec{A}_l$ are manipulated on, remaining (k-1) elements of the resulting vector $\vec{C}_k$ are not affected by the applied operation and are simply the same as the elements of the vector $\vec{B}_k$ between positions k and $l + 1$.

Definition 5 *An invertible forward sign Walsh transform w is defined as [2, 5]:*

$$w(\overrightarrow{\alpha_n})=\text{sign}\left\{\sum_{x_n=0}^{1}\text{sign}\left[\sum_{x_{n-1}=0}^{1}\text{sign}\left(\ldots\text{sign}\sum_{x_1=0}^{1} f(\overrightarrow{x_n})(-1)^{\sum_{p=1}^{n}\alpha_p x_p}\right)\right]\right\} \tag{3}$$

The inverse sign Walsh transform is:

$$f(\overrightarrow{x_n})=\text{sign}\left\{\sum_{\alpha_1=0}^{1}\text{sign}\left[\sum_{\alpha_2=0}^{1}\text{sign}\left(\ldots\text{sign}\sum_{\alpha_n=0}^{1} w(\overrightarrow{\alpha_n})(-1)^{\sum_{p=1}^{n}\alpha_p x_p}\right)\right]\right\} \tag{4}$$

In eqns. 3-4, $1 \le p \le n$.

Definition 6 *An invertible forward Sign Hadamard-Haar transform hh is:*

$$hh(\vec{O}_n \oplus_d \omega_1 \oplus_d \omega_n 2^{n-1}) = \text{sign}\sum_{x_n=0}^{1}[\ (-1)^{x_n\omega_n}[\text{sign}\sum_{x_{n-1}=0}^{1}[\text{sign}\sum_{x_{n-2}=0}^{1}[\cdots\text{sign}\sum_{x_1=0}^{1}\{(-1)^{x_{n-1}\omega_1} f(\vec{x}_n)\}\cdots]]]] \tag{5}$$

and

$$hh(\vec{O}_n \oplus_d \vec{\omega}_i \oplus_d 2^i \oplus_d \omega_n 2^{n-1}) = \text{sign}\sum_{x_n=0}^{1}[(-1)^{x_n\omega_n}[\text{sign}\sum_{x_{n-i-1}=0}^{1}[\text{sign}\sum_{x_{n-i-2}=0}^{1}[\cdots\text{sign}\sum_{x_1=0}^{1}(-1)^{x_{n-2}} f\{(\vec{O}_{n-1} \oplus_d \vec{\omega}_i)$$
$$\oplus_c (n-i-1)\} \oplus_d \vec{x}_{n-i-1}\}]] \tag{6}$$

where $\vec{\omega}_i = \{\omega_i, \omega_{i-1}, \cdots, \omega_2, \omega_1\}$, $1 \le i < n-1$.

The inverse Sign Hadamard-Haar transform is

$$(\vec{x}_n) = \text{sign}\{(-1)^{x_1}\{\text{sign}\{hh\{[(\vec{O}_1 \wedge \vec{x}_{n-1}) \oplus_c 1] \oplus_d 2^{n-2}\} + (-1)^{x_n} hh\{[(\vec{O}_1 \wedge \vec{x}_{n-1}) \oplus_c 1] \oplus_d 2^{n-2} \oplus_d 2^{n-1}\}\}\} +$$
$$\text{sign}\{(-1)^{x_2}\{\text{sign}\{hh\{[(\vec{O}_2 \wedge \vec{x}_{n-1}) \oplus_c 2] \oplus_d 2^{n-3}\} + (-1)^{x_n} hh\{[(\vec{O}_2 \wedge \vec{x}_{n-1}) \oplus_c 2]$$
$$\oplus_d 2^{n-3} \oplus_d 2^{n-1}\}\}\} + \cdots + \text{sign}\{(-1)^{x_i}\{\text{sign}\{hh\{[(\vec{O}_i \wedge \vec{x}_{n-1}) \oplus_c i] \oplus_d 2^{n-i-1}\} +$$
$$(-1)^{x_n} hh\{[(\vec{O}_i \wedge \vec{x}_{n-1}) \oplus_c i] \oplus_d 2^{n-i-1} \oplus_d 2^{n-1}\}\}\} + \cdots + \text{sign}\{(-1)^{x_{n-2}}\{\text{sign}\{$$
$$hh\{[(\vec{O}_{n-2} \wedge \vec{x}_{n-1}) \oplus_c (n-2)] \oplus_d 2\} + (-1)^{x_n} hh\{[(\vec{O}_{n-2} \wedge \vec{x}_{n-1}) \oplus_c (n-2)] \oplus_d 2 \oplus_d 2^{n-1}\}\}\} +$$
$$\text{sign}\{\sum_{\omega_1=0}^{1}(-1)^{x_{n-1}\omega_1}\{\text{sign}[h(\vec{O}_{n-1} \oplus_d \omega_1) + (-1)^{x_n} h(\vec{O}_{n-1} \oplus_d \omega_1 \oplus_d 2^{n-1})]\}\}\cdots\}\}\} \tag{7}$$

where $1 \le i < n-1$, $\vec{x}_{n-1} = \{x_{n-1}, x_{n-2}, \cdots, x_i, \cdots, x_2, x_1)$.

In eqn. 7,

$$\vec{O}_i \wedge \vec{x}_{n-1} = \vec{O}_i \wedge \{x_{n-1}, x_{n-2}, \cdots, x_{i+1}, x_i, x_{i-1}, \cdots, x_2, x_1\} = \{x_{n-1}, x_{n-2}, \cdots x_{i+1}, 0, 0, \cdots, 0, 0\}.$$

In eqns. 3-7,

$$\text{sign}\, z = \begin{cases} -1 & z < 0 \\ 0 & z = 0 \\ +1 & z > 0 \end{cases}. \tag{8}$$

Proof of uniqueness *Let us prove uniqueness of Sign Hadamard-Haar transform for a single variable and the general case for n variables follows by induction. The output function* $f(x_i) \in \{-1,0,+1\}$ where $x_1 \in \{0,1\}$. *From eqn. 5, forward Sign Hadamard-Haar transform,*

$$hh(\vec{\omega}_1) = hh(\omega_1) = \text{sign}[\sum^{1}(-1)^{\omega_1 x_1} f(\vec{x_1})] = \text{sign}[\sum^{1}(-1)^{\omega_1 x_1} f(x_1)] = \text{sign}[f(0) + (-1)^{\omega_1} f(1)] \cdot \quad (9)$$

From eqn. 7, inverse Sign Hadamard-Haar transform,

$$f(\vec{x}_1) = f(x_1) = \text{sign}\,[\sum_{\omega_1=0}^{1}(-1)^{x_1\omega_1} hh(\vec{O_1} \oplus_d \omega_1)] = \text{sign}\,[\sum_{\omega_1=0}^{1}(-1)^{x_1\omega_1} hh(\omega_1)]$$

$$= \text{sign}\,[hh(0) + (-1)^{x_1} hh(1)]\,. \quad (10)$$

For the case of $x_1 = 0$, the right-hand side of eqn. 10 yields

$$\text{sign}[hh(0) + (-1)^{x_1} hh(1)] = \text{sign}[\text{sign}[f(0) + f(1)] + \text{sign}[f(0) - f(1)]] = f(0)\,.$$

The case $x_1 = 1$ is proved similarly.

Example 1 *For n = 3 the definitions of forward Sign Hadamard-Haar transform hh become:*

$$hh(\vec{O}_3 \oplus_d \omega_1 \oplus_d \omega_3 2^2) = hh(\omega_3, 0, \omega_1) = \text{sign}\sum_{x_3=0}^{1}[\ (-1)^{x_3\omega_3}[\text{sign}\sum_{x_2=0}^{1}[\ \text{sign}\sum_{x_1=0}^{1}\{(-1)^{x_2\omega_1} f(\vec{x}_3)\}]]]$$

and for $1 \le i < n\text{-}1$

$$hh(\vec{O}_3 \oplus_d \vec{\omega}_i \oplus_d 2^i \oplus_d \omega_3 2^2) = hh(\omega_3, 1, \omega_1) = \text{sign}\sum_{x_3=0}^{1}[(-1)^{x_3\omega_3}[\text{sign}\sum_{x_1=0}^{1}(-1)^{x_1} f\{(\vec{O}_2 \oplus_d \vec{\omega}_1) \oplus_c 1\} \oplus_d \vec{x}_1\}]]$$

$$= \text{sign}\sum_{x_3=0}^{1}[(-1)^{x_3\omega_3}[\text{sign}\sum_{x_1=0}^{1}(-1)^{x_1} f(x_3, \omega_1, x_1)]]\,.$$

In a similar manner definitions for inverse transform from eqn. 6 when n = 3 can be derived.

Example 2 *Consider the logic function given by* $f_0(\vec{x}_3) = (1,0,1,1,0,1,1,0)$. *The elements of its corresponding Sign Hadamard-Haar spectrum* $hh_0(\vec{\omega}_3)$ *could be calculated from* $f_0(\vec{x}_3)$ *as follows:*

$$hh_0(0,0,0)=\text{sign}\sum_{x_3=0}^{1}[(-1)^{x_3\omega_3}[\text{sign}\sum_{x_2=0}^{1}[\text{sign}\sum_{x_1=0}^{1}\{(-1)^{x_2\omega_1} f_0(\vec{x}_3)\}]]]=\text{sign}\sum_{x_3=0}^{1}[\text{sign}\sum_{x_2=0}^{1}[\text{sign}\sum_{x_1=0}^{1} f_0(\vec{x}_3)]]$$

$$= \text{sign}[\text{sign}[\ \text{sign}[\ f_0(0,0,0)+f_0(0,0,1)]+\text{sign}[\ f_0(0,1,0)+f_0(0,1,1)]]+$$

$$\text{sign}[\text{sign}[\ f_0(1,0,0)+f_0(1,0,1)]+\text{sign}[\ f_0(1,1,0)+f_0(1,1,1)]]]$$

$$= \text{sign}\,[\text{sign}\,[\text{sign}[\,1+0\,]+\text{sign}[\,1+1\,]]+\text{sign}\,[\text{sign}[\,0+1\,]+\text{sign}[\,1+0\,]]]\ =1\,.$$

$$hh_0(1,1,1)=\text{sign}\sum_{x_3=0}^{1}[(-1)^{x_3\omega_3}[\text{sign}\sum_{x_1=0}^{1}(-1)^{x_1} f_0(x_3,\omega_1,x_1)]=\text{sign}\sum_{x_3=0}^{1}[(-1)^{x_3}[\text{sign}\sum_{x_1=0}^{1}(-1)^{x_1} f_0(x_3,1,x_1)]$$

$$= \text{sign}[\text{sign}[\ f_0(0,1,0)-f_0(0,1,1)]-\text{sign}[\ f_0(1,1,0)-f_0(1,1,1)]]$$

$$= \text{sign}[\text{sign}[\,1-1\,]-\text{sign}[\,1-0\,]]=-1\,.$$

In the similar manner the other elements of $hh_0(\vec{\omega}_3)$ *can be calculated and the final Sign Hadamard-Haar spectrum is* $\overrightarrow{HH}_0(\vec{\omega}_3)=(1,0,0,1,0,0,1,-1)$.

The elements of $f_0(\vec{x}_3)$ *can be calculated from the spectrum* $hh_0(\vec{\omega}_3)$ *too. Using eqn. 6, when n = 3, the first and the last minterms in the truth vector are obtained as follows:*

$$f_0(0,0,0)=\text{sign}\{(-1)^{x_1}\{\text{sign}\{hh_0\{[(\vec{O}_1\wedge\vec{x}_2)\oplus_c 1]\oplus_d 2\}+(-1)^{x_3}hh_0\{[(\vec{O}_1\wedge\vec{x}_2)\oplus_c 1]\oplus_d 2\ \oplus_d 2^2\}\}\}+$$

$$\text{sign}\{\sum_{\omega_1=0}^{1}(-1)^{x_2\omega_1}\{\text{sign}[hh_0(\vec{O}_2\oplus_d\omega_1)+(-1)^{x_3}hh_0(\vec{O}_2\oplus_d\omega_1\oplus_d 2^2)]\}\}\}$$

$$= \text{sign}\{\text{sign}[\ hh_0(0,1,0)+hh_0(1,1,0)]+\text{sign}\{\text{sign}[\ hh_0(0,0,0)+hh_0(1,0,0)]$$

$$+\,\text{sign}[\,hh_0(0,0,1)+hh_0(1,0,1)]\}\}$$

$$= \text{sign}\{\text{sign}[\ 0+0]+\text{sign}\{\text{sign}[\ 1+0]+\text{sign}[\,0+1]\}\}=1,$$

$$f_0(1,1,1)=\text{sign}\{(-1)^{x_1}\{\text{sign}\{hh_0\{[(\vec{O}_1\wedge\vec{x}_2)\oplus_c 1]\oplus_d 2\}+(-1)^{x_3}hh_0\{[(\vec{O}_1\wedge\vec{x}_2)\oplus_c 1]\oplus_d 2\ \oplus_d 2^2\}\}\}+$$

$$\text{sign}\{\sum_{\omega_1=0}^{1}(-1)^{x_2\omega_1}\{\text{sign}[hh_0(\vec{O}_2\oplus_d\omega_1)+(-1)^{x_3}hh_0(\vec{O}_2\oplus_d\omega_1\oplus_d 2^2)]\}\}\}$$

$$= \text{sign}\{\text{-sign}\,[hh_0(0,1,1)-hh_0(1,1,1)]+\text{sign}\{\text{sign}[\,hh_0(0,0,0)-hh_0(1,0,0)]$$

$$-\,\text{sign}[hh_0(0,0,1)-hh_0(1,0,1)]\}\}$$

$$= \text{sign}\{\text{-sign}\,[\,1-(-1)]+\text{sign}\{\text{sign}[1-0]+\text{sign}[0-1]\}\}=-1.$$

The other elements of $f_0(\vec{x}_3)$ *could be calculated in a similar way.*

3. PROPERTIES OF SIGN HADAMARD-HAAR SPECTRA OF LOGIC FUNCTIONS AND VARIABLES

Sign Hadamard-Haar spectra for common logic functions and the major properties of the transform are presented. There is no direct relationship between Sign Hadamard-Haar spectra calculated for *S* and *R* coded logic functions except some special cases for selected logic functions described by Properties 4, 5, 7, 8 in this Section what differs from other transforms used in logic design (i.e., Walsh, Haar, arithmetic). Therefore, basic properties for logic operators have to be derived separately for both codings.

In the following presentation of the properties, let logic function *f* and its corresponding Sign Hadamard-Haar transform *hh* be defined as in the previous section. Let *p* and *q* be ternary variables, where $p,q \in \{-1,0,1\}$. In order to illustrate better investigated properties let us show a sign domain map.

Definition 7 *A sign domain map is a graphical two-dimensional representation of Sign Hadamard-Haar spectrum and is an equivalent of a Karnaugh map in logic domain where spectral variables listed in Gray code order are used to indicate all the cells of the map and sign spectral coefficients' values are entered into the cells.*

Property 1 *The number of cells in sign domain map of the spectrum of an n-variable logic function is exactly the same as the number of the minterms (cells on Karnaugh map) of such a function.*

Property 2 *For arbitrary ternary variables p and q:*

$$\mathrm{sign}[\mathrm{sign}(p+q)+\mathrm{sign}(p-q)] = p \tag{11}$$

and

$$\mathrm{sign}[\mathrm{sign}(p+q)-\mathrm{sign}(p-q)] = q\,. \tag{12}$$

Property 3 *Let function* $f(\vec{x}_n)$ *be a constant, such that its ternary vector* $\vec{F}$ *has all the coefficients equal and* F_j $(1 \le j \le 2^n)$; $F_j \in \{-,0,+\}$. *Then,*

$$f(\vec{x}_n) = 0 \Leftrightarrow hh(\vec{\omega}_n) = 0,\;\; x_i, \omega_j \in \{0,1\} \quad \text{and } 1 \le i \le n\,, \tag{13}$$

$$f(\vec{x}_n) = \pm 1 \Leftrightarrow hh(\vec{\omega}_n) = \pm \prod_{j=1}^{n} (1-\omega_j)\,\cdot \tag{14}$$

Example 3 *For n = 3,* $f_1(\overrightarrow{x_3}) = (+,+,+,+,+,+,+,+) \Leftrightarrow \overrightarrow{HH_1}(\overrightarrow{\omega_3}) = (+,0,0,0,0,0,0,0)$. *Karnaugh map of the function* $f_1(\overrightarrow{x_3})$ *and sign domain map of the spectrum are shown in Figure 1(a).*

Property 4 *When S-coded n-variable function is dependent on a single logic variable in affirmation,* $f(\vec{x}_n) = f(x_n,...,x_1) = x_j, j \in \{1,...,n\}, x_j \in \{+1,-1\}$, *then its Sign Hadamard-Haar transform is*

$$hh(\vec{\omega}_n) = \begin{cases} +1 \times (\omega_{n-j} \wedge \{\bigwedge_{k=n-j+1}^{n} \overline{\omega_k}\}) & \text{when } j < n \\ +1 \times (\omega_n \wedge \{\bigwedge_{k=1}^{n-1} \overline{\omega_k}\}) & \text{when } j = n \end{cases} \tag{15}$$

where $\omega_k \in \{0,1\}$ *and the logic AND operations in bracket ( ) will yield value 1 or 0. The symbol* $\overline{\omega_k}$ *represents the logical inversion of the transform variable* ω_k.

Example 4 *For n = 3, when* $f_2(\overrightarrow{x_3}) = x_3$, *Sign Hadamard-Haar transform is*

$$hh_2(\overrightarrow{\omega_3}) = +1 \times (\omega_3 \wedge \{\bigwedge_{k=1}^{2} \overline{\omega_k}\}) = +1(\omega_3 \wedge \overline{\omega_1} \wedge \overline{\omega_2}) = \omega_3 \overline{\omega_2}\, \overline{\omega_1}.$$

Hence $f_2(\overrightarrow{x_3}) = (+,+,+,+,-,-,-,-) \Leftrightarrow \overrightarrow{HH_2}(\overrightarrow{\omega_3}) = (0,0,0,0,+,0,0,0)$.

The function $f_2(\overrightarrow{x_3})$ *and its corresponding spectrum are shown in Figure 1(b).*

Example 5 *For n = 3, when* $f_3(\overrightarrow{x_3}) = x_2$, *Sign Hadamard-Haar transform is*

$$hh_3(\overrightarrow{\omega_3}) = +1 \times (\omega_{3-2} \wedge \{\bigwedge_{k=3-2+1}^{3} \overline{\omega_k}\}) = +1(\omega_1 \wedge \overline{\omega_2} \wedge \overline{\omega_3}) = \overline{\omega_3}\, \overline{\omega_2} \omega_1.$$

Hence $f_3(\overrightarrow{x_3}) = (+,+,-,-,+,+,-,-) \Leftrightarrow \overrightarrow{HH_3}(\overrightarrow{\omega_3}) = (0,+,0,0,0,0,0,0)$.

The function $f_3(\overrightarrow{x_3})$ *and its corresponding spectrum are shown in Figure 1(c).*

Property 5 *When S-coded n-variable function is dependent on a single logic variable in negation,* $f(\vec{x}_n) = f(x_n,...,x_1) = \overline{x}_j, j \in \{1,...,n\}, x_j \in \{+1,-1\}$, *then its Sign Hadamard-Haar transform is*

$$hh(\vec{\omega}_n) = \begin{cases} -1 \times (\omega_{n-j} \wedge \{\bigwedge_{k=n-j+1}^{n} \overline{\omega_k}\}) & \text{when } j < n \\ -1 \times (\omega_n \wedge \{\bigwedge_{k=1}^{n-1} \overline{\omega_k}\}) & \text{when } j = n \end{cases}. \tag{16}$$

Property 6 *For S-coded n-variable function $f(\vec{x}_n)$ whose Sign Hadamard-Haar spectrum is $hh(\vec{\omega}_n)$, the spectrum of the negated function is derived simply by inverting all the signs of the original spectra. Hence when*

$$f(\vec{x}_n) \Leftrightarrow hh(\vec{\omega}_n) \quad \text{then} \quad \overline{f(\vec{x}_n)} \Leftrightarrow -hh(\vec{\omega}_n). \tag{17}$$

From eqn. 17 the negated spectrum is obtained by inverting the signs of coefficients.

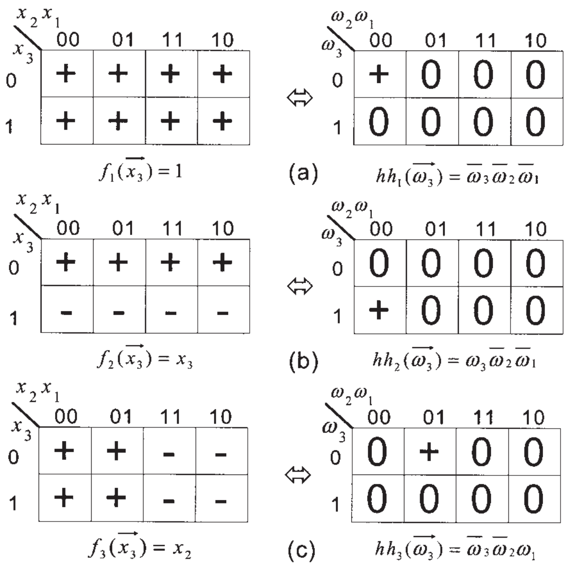

FIGURE 1
Karnaugh and sign domain maps for S-coded functions, f_1, f_2 and f_3.

Property 7 *When R-coded n-variable function is dependent on a single logic variable in affirmation, $f(\vec{x}_n) = f(x_n,\ldots,x_1) = x_j, j \in \{1,\ldots,n\}, x_j \in \{0,1\}$, then its Sign Hadamard-Haar transform is*

$$hh(\vec{\omega}_n) = \begin{cases} \bigwedge_{k=1}^{n} \overline{\omega_k} - (\omega_{n-j} \wedge \{\bigwedge_{k=n-j+1}^{n} \overline{\omega_k}\}) & \text{when } j < n \\ \bigwedge_{k=1}^{n} \overline{\omega_k} - (\omega_n \wedge \{\bigwedge_{k=1}^{n-1} \overline{\omega_k}\}) & \text{when } j = n \end{cases}. \tag{18}$$

Example 6 *For n = 4, when* $f_4(\overrightarrow{x_4}) = x_2$, *then by eqn. 18 Sign Hadamard-Haar transform is*

$$hh_4(\overrightarrow{\omega_4}) = \overset{4}{\underset{k=1}{\wedge}}\overline{\omega_k} - (\omega_{4-2} \wedge \{\overset{4}{\underset{k=4-2+1}{\wedge}}\overline{\omega_k}\}) = (\overline{\omega_1} \wedge \overline{\omega_2} \wedge \overline{\omega_3} \wedge \overline{\omega_4}) - (\omega_2 \wedge \overline{\omega_3} \wedge \overline{\omega_4})$$

$$= \overline{\omega_4}\,\overline{\omega_3}\,\overline{\omega_2}\,\overline{\omega_1} - \overline{\omega_4}\,\overline{\omega_3}\omega_2 .$$

Hence,

$$f_4(\overrightarrow{x_4}) = (0,0,+,+,0,0,+,+,0,0,+,+,0,0,+,+) \Leftrightarrow \overrightarrow{HH}_4(\overrightarrow{\omega_4}) = (+,0,-,-,0,0,0,0,0,0,0,0,0,0,0,0).$$

Property 8 *When R-coded n-variable function is dependent on a single logic variable in negation,* $f(\vec{x}_n) = f(x_n,..., x_1) = \overline{x}_j, j \in \{1,..., n\}, x_j \in \{0,1\}$, *then its Sign Hadamard-Haar transform is*

$$hh(\vec{\omega}_n) = \begin{cases} \overset{n}{\underset{k=1}{\wedge}}\overline{\omega_k} + (\omega_{n-j} \wedge \{\overset{n}{\underset{k=n-j+1}{\wedge}}\overline{\omega_k}\}) & \text{when } j < n \\ \overset{n}{\underset{k=1}{\wedge}}\overline{\omega_k} + (\omega_n \wedge \{\overset{n-1}{\underset{k=1}{\wedge}}\overline{\omega_k}\}) & \text{when } j = n \end{cases} . \tag{19}$$

4. CALCULATION OF SIGN HADAMARD-HAAR TRANSFORM BY MATRIX APPROACH

Sign Hadamard-Haar transform may be evaluated using a matrix approach. It is also possible to evaluate a single spectral coefficient of Sign Hadamard-Haar transform without need to calculate the whole spectrum.

Definition 8 *Let* T_n *and* T_n^{-1} *be* $2^n \times 2^n$ *forward and inverse Sign Hadamard-Haar transform matrices. Then*

$$T_n = [\vec{t_0}, \vec{t_1}, \cdots, \vec{t_i}, \cdots, \overrightarrow{t_{2^n-1}}]^T,\ T_n^{-1} = [\overrightarrow{t_0^{-1}}, \overrightarrow{t_1^{-1}}, \cdots, \overrightarrow{t_i^{-1}}, \cdots, \overrightarrow{t_{2^n-1}^{-1}}]^T, \text{ where } 0 \le i \le 2^n - 1. \tag{20}$$

Definition 9 *Let* $\vec{A} = [\vec{a_0}, \vec{a_1}, \cdots, \vec{a_i}, \cdots, \overrightarrow{a_{2^n-1}}]$ *be a 1 × 2^n row vector and let* $\vec{B} = [\vec{b_0}, \vec{b_1}, \cdots, \vec{b_i}, \cdots, \overrightarrow{b_{2^n-1}}]^T$ *be a 2^n × 1 column vector. The vector product A*B is a 1 × 2^n row vector with the elements derived from component-wise multiplication:*

$$\vec{A} * \vec{B} = [\overrightarrow{a_0 b_0}, \overrightarrow{a_1 b_1}, \cdots, \overrightarrow{a_i b_i}, \cdots, \overrightarrow{a_{2^n-1} b_{2^n-1}}]. \tag{21}$$

Definition 10 *Let $\vec{A}$ be a 1 × 2^n row vector where entries are $[\vec{a_i}]$. The sign modulo function $\lceil \vec{A} \rceil$ is a scalar where the elements in $\vec{A}$ are summed in pairs in the form of a binary tree summation, and the tree is evaluated from the bottom up.*

Property 9 *Let $\vec{F} = [F_0, F_1, ..., F_j, ..., F_{N-2}, F_{N-1}]$, $\overrightarrow{HH}_F = [hh_0, hh_1 ..., hh_j, ..., hh_{N-2}, hh_{N-1}]$ define the ternary vector and its Sign Hadamard-Haar spectra coefficients.*

Then, $hh_i = \lceil \vec{t_i} * \vec{F} \rceil$ and $F_i = \lceil \vec{s_i} * \overrightarrow{HH_F} \rceil$.

Example 7 *Let n = 3, and the ternary vector $\vec{F} = [-,-,+,-,+,-,-,+]$. From Definition 10*

$$T_n = [\,(+ + + + + + + +),(+ + - - + + - -),(+ - 0\,0 + - 0\,0),(0\,0 + - 0\,0 + -),$$

$$(+ + + + - - - -),(+ + - - - - + +),(+ - 0\,0 - + 0\,0),(\,0\,0 + - 0\,0 - +)]^T.$$

By Property 9 and Definition 10

$$hh_0 = \lceil (+ + + + + + + +) * (- - + - + - - +)^T \rceil = \lceil (- - + - + - - +) \rceil = \lceil (- 0\,0\,0) \rceil = \lceil (- 0) \rceil = -,$$

$$hh_1 = \lceil (+ + - - + + - -) * (- - + - + - - +)^T \rceil = \lceil (- - - + + - + -) \rceil = \lceil (- 0\,0\,0) \rceil = \lceil (- 0) \rceil = -,$$

$$hh_2 = \lceil (+ - 0\,0 + - 0\,0) * (- - + - + - - +)^T \rceil = \lceil (- + 0\,0 + + 0\,0) \rceil = \lceil (0\,0 + 0) \rceil = \lceil (0 +) \rceil = +,$$

$$hh_3 = \lceil (0\,0 + - 0\,0 + -) * (- - + - + - - +)^T \rceil = \lceil (0\,0 + + 0\,0 - -) \rceil = \lceil (0 + 0 -) \rceil = \lceil (+ -) \rceil = 0,$$

$$hh_4 = \lceil (+ + + + - - - -) * (- - + - + - - +)^T \rceil = \lceil (- - + - - - + + -) \rceil = \lceil (- 0\,0\,0) \rceil = \lceil (- 0) \rceil = -.$$

In the similar manner the other elements of $\overrightarrow{HH_F}$ are calculated and the final Sign Hadamard-Haar spectrum is $\overrightarrow{HH_F} = [-,-,+,0,-,-,-,+]$.

5. COMPARISON OF COMPUTATIONS OF SIGN HADAMARD-HAAR, SIGN WALSH AND TERNARY REED-MULLER TRANSFORMS

In this section, the computational advantages of new Sign Hadamard-Haar transform over known Sign Walsh transform and ternary Reed-Muller transform over GF(3) are discussed, but all these transforms have the same computer memory requirements for storage of data. The computational costs mean the number of additions, subtractions and multiplications required for the generation of forward and inverse transforms.

Fast flow diagrams for calculation of forward and inverse Sign Hadamard-Haar transforms *hh* are shown for N = 8 in Figure 2(a) and (b) where (•) represents the sign function, the solid lines and dotted lines represent addition and subtraction, respectively. The total number of operations (additions and subtractions) required to perform forward fast Sign Hadamard-Haar transform *hh* for a single element of set $\{+1, 0, -1\}^N$ and inverse Sign Hadamard-Haar transform hh^{-1} for a single element of set $\{+1, 0, -1\}^N_{(hh)}$ and for transform matrix of order $N = 2^n$ is equal to $3 \times 2^{n+1} - 8$.

The computational cost of Sign Walsh transform is $2n2^n$ [2, 5]. Fast flow diagrams for calculation of forward and inverse Sign Walsh transform *hh* are shown for N = 8 in Figure 3(a) and (b). It is obvious from Figure 2 and Figure 3 that the computational costs of Sign Hadamard-Haar transform are less than the computational costs of Sign Walsh transform. Table 3 shows that the computational costs of Sign Walsh transform increase considerably when compared with Sign Hadamard-Haar transform for higher n.

There are many methods to calculate ternary Reed-Muller transform such as direct matrix calculation, fast transform calculation based on Kronecker product, Gray code ordered fast transform calculation, column and row method calculation [8, 13, 15]. The number of ternary additions and multiplications required to calculate a particular polarity coefficient vector using Green's fast method are $n3^n$ and $4n3^{n-1}$, respectively [15]. While the column polarity matrix method and row polarity matrix method are more advantageous than Green methods in the number of ternary additions and multiplications that need to be calculated, Table 3 and Table 4 show the numbers of ternary additions and multiplications required to calculate a particular polarity coefficient vector using these methods, respectively. It should be noticed that even the most efficient methods to calculate ternary Reed-Muller form require calculation of the whole polarity matrix in order to find the best expansion as well as multiplications over GF(3) and their number of additions operations is much higher than the corresponding number of only operations (additions or subtractions) for both sign transforms who do not have any polarity. Table 2 and Table 3 show that the computational costs of operations to calculate Sign Hadamard-Haar transform are less than the numbers of ternary additions required for ternary

Reed-Muller transform using even the best methods while the multiplications are not required in the calculation of Sign Hadamard-Haar transform. Moreover, the sign operation used in the new transform is much simpler than the GF(3) operations used in the ternary Reed-Muller transform.

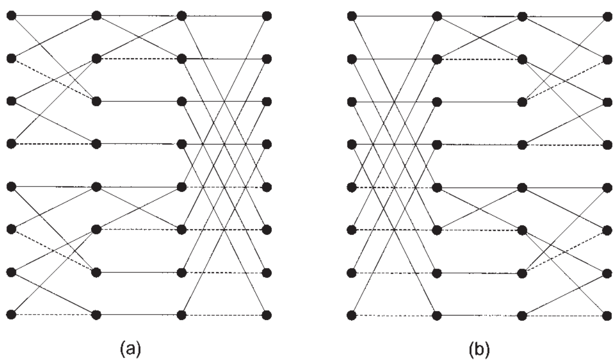

FIGURE 2
Butterfly diagram for (a) forward and (b) inverse Sign Hadamard-Haar transform, $n = 3$.

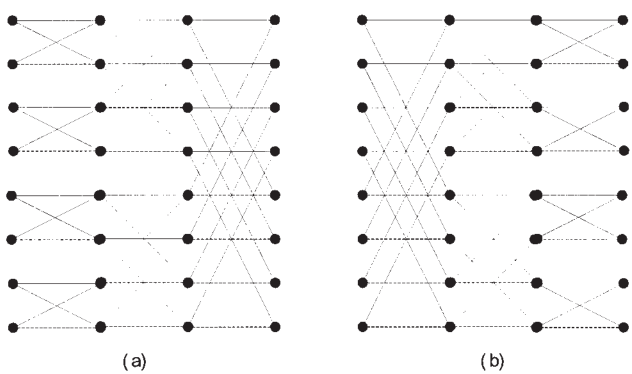

FIGURE 3
Butterfly diagram for (a) forward and (b) inverse Sign Walsh transform, $n = 3$.

TABLE 2
Comparison of computational costs.

n	Sign Hadamard-Haar transform	Sign Walsh transform
	$3\times2^{n+1}-8$	$2n2^{n}$
1	4	4
2	16	16
3	40	48
4	88	128
5	184	320
6	376	768
7	760	1,792
8	1,528	4,086
9	3,064	9,216
10	6,136	20,480
11	12,280	45,056
12	24,568	98,304
13	49,144	212,992
14	98,296	458,752
15	196,600	983,040

TABLE 3
Comparison of computational costs involved in deriving a whole polarity ternary Reed-Muller matrix (additions).

n	Direct	Fast	Gray code Order	Row polarity matrix	Column polarity matrix
1	6	6	6	7	5
2	108	90	72	58	50
3	1350	972	702	397	395
4	14,580	9,396	6,480	2,644	2,900
5	146,286	86,994	58,806	17,779	20,705

TABLE 4
Comparison of computational costs involved in deriving a whole polarity ternary Reed-Muller matrix (multiplications).

n	Direct	Fast	Gray code order	Row polarity matrix	Column polarity matrix
1	2	2	2	4	1
2	38	30	24	24	10
3	494	324	234	108	79
4	5,486	3,132	2,160	432	580
5	56,126	28,998	19,602	1,620	4,141

6. APPLICATION OF SIGN HADAMARD-HAAR IN TERNARY COMMUNICATION SYSTEM

In the previous Sections it has been shown that Sign Hadamard-Haar transform similarly to well know Sign Walsh transform exhibit non-linear properties. Though non-linear, these transforms are unique and invertible. With intrinsic coding property, these transforms reveal possible application in secured communication systems [12]. In this Section, the application of Sign Hadamard–Haar as the sequence for ternary communication system will be considered. For comparison the well known Sign Walsh transform will also be used in the same application. In such a system, the incoming binary/ternary data is first encoded by performing Sign Hadamard-Haar transform on it. The digital modulation technique responsible for carrying information in Sign Hadamard-Haar spectra is Ternary Amplitude Frequency Shift Keying (TAFSK) [12, 14]. In this signaling, a ternary +1 is transmitted by a Radio Frequency (RF) pulse of carrier cosω_1t, a ternary -1 is transmitted by an RF pulse of carrier cosω_2t, and a 0 corresponds to no RF pulse. The technique combines Binary Amplitude Shift Keying and Binary Frequency Shift Keying for the ternary case. The Power Spectral Density (PSD) of the resultant signaling is given by

$$S(\omega)=\frac{1}{2}[A_1(\omega+\omega_1)+A_1(\omega-\omega_1)+A_1(\omega+\omega_2)+A_1(\omega-\omega_2)] \tag{22}$$

where

$$A_1(\omega) = \frac{2}{9} T_0 \mathrm{sinc}^2\left(\frac{\omega T_0}{2\pi}\right)\left[1 + \frac{T_0}{\pi}\sum_{m=-\infty}^{\infty} \delta\left(\omega - \frac{2\pi m}{T_0}\right)\right]$$

Proof *Let Sign Hadamard-Haar transform of binary/ternary data streams be represented by*

$$A(t) = \sum_{k=-\infty}^{\infty} a_k p(t-kT_0) = A_1(t) + A_2(t)$$

where p(t) represents a full rectangular pulse which repeats every T_0 seconds, and it is assumed that a_k is equally likely to be +1, 0 or -1, i.e. $P(a_k = 1) = P(a_k = -1) = P(a_k = 0) = \frac{1}{3}$. *Furthermore*

$$A_1(t) = \sum_{k=-\infty}^{\infty} a_k^{\langle 1\rangle} p(t-kT_0) \text{ and } A_2(t) = \sum_{k=-\infty}^{\infty} a_k^{\langle -1\rangle} p(t-kT_0)$$

with $P\left(a_k^{\langle 1\rangle} = 1\right) = P\left(a_k^{\langle -1\rangle} = -1\right) = \frac{1}{3}$ and $P\left(a_k^{\langle 1\rangle} = 0\right) = P\left(a_k^{\langle -1\rangle} = 0\right) = \frac{2}{3}$. *The PSD of ON-OFF signaling [14] is*

$$A_0(\omega) = \frac{|P(\omega)|^2}{T_0}\left[\sum_{m=-\infty}^{\infty} R_m e^{-jm\omega T_0}\right]$$

where R_m is the coefficient of the time-autocorrelation function of the signaling and $j = \sqrt{-1}$. *Therefore the PSD of $A_1(t)$ is $A_1(\omega) = A_0(\omega)$, with*

$$R_0 = \lim_{N_T \to \infty} \frac{1}{N_T} \sum_{k=1}^{N_T} \left(a_k^{\langle 1\rangle}\right)^2 = \frac{1}{3}$$

and

$$R_m = \lim_{N_T \to \infty} \frac{1}{N_T} \sum_{k=1}^{N_T} a_k^{\langle 1\rangle} a_{k=m}^{\langle 1\rangle} = \frac{1}{9}, \text{ if } m \neq 0.$$

Since

$$P(\omega) = T_0 \mathrm{sinc}\left(\frac{\omega T_0}{2\pi}\right)$$

and using

$$\sum_{m=-\infty}^{\infty} \mathrm{e}^{-jm\omega T_0} = \frac{T_0}{2\pi} \sum_{m=-\infty}^{\infty} \delta\left(\omega - \frac{2\pi m}{T_0}\right)$$

then

$$A_1(\omega) = \frac{2}{9} T_0 \mathrm{sinc}^2\left(\frac{\omega T_0}{2\pi}\right)\left[1 + \frac{T_0}{\pi}\sum_{m=-\infty}^{\infty}\delta\left(\omega - \frac{2\pi n}{T_0}\right)\right].$$

Since $\left(a_k^{\langle 1\rangle}\right)^2 = \left(a_k^{\langle -1\rangle}\right)^2$ *and* $a_k^{\langle 1\rangle} a_{k+m}^{\langle 1\rangle} = a_k^{\langle -1\rangle} a_{k+m}^{\langle -1\rangle}$, *therefore* $A_2(\omega) = A_1(\omega)$. *Using the frequency shifting property and since*

$$s(t) = A_1(t)\cos\omega_1 t + A_2(t)\cos\omega_2 t$$

the proof of eqn. 22 is complete.

For $\omega_2 > \omega_1$, if $\omega_2 - \omega_1 = 2\omega_0$ then the transmission bandwidth of TAFSK signaling is $4f_0$ (where $f_0 = 1/T$ is the clock frequency).

A new recursive transform can be developed based on Sign Hadamard-Haar transform. As an example, Sign Hadamard-Haar transform can be applied twice onto a ternary truth column vector $\vec{F}$ such that a new transform space is developed. The overall transform is named as Sign Hadamard-Haar-2 transform. In general, there together 3^N different Sign Hadamard-Haar transform spaces, denoted as Sign Hadamard-Haar-q transforms with $1 \leq q \leq 3^N$, where when $q = 1$, the transform yields the original Sign Hadamard-Haar transform. For comparison purpose, Sign Walsh-q transform can be defined similarly.

Figure 4 shows the block diagram of a TASFK transmitter. The continuous streams of binary/ternary data are converted to parallel words of length N by means of a serial-parallel converter. Sign Hadamard-Haar-q transform is applied to each word before converting back to the format of serial data. The output signal V of the parallel-serial converter controls the output frequency of the voltage-controlled oscillator, and both outputs are fed together into the mixer. The output of the mixer is TAFSK signaling. The output of the oscillator is mathematically given by

$$VCO = V_0 \cos[2\pi(f_c + Vf_m)t] \tag{23}$$

where $V \in \{-1, 0, 1\}$ and V_0 is an arbitrary amplitude. If $f_m = f_0$, then the resultant transmission bandwidth will be $4f_0$, and $f_c + f_0 = f_2$, $f_c - f_0 = f_1$.

Figure 5 shows a block diagram of a TAFSK receiver. The incoming noisy RF signal is bandpass filtered centered at frequency f_c. The bandpass filters centered at f_2 and f_1 are matched to the two RF pulses corresponding to ternary logic of -1 and +1, accordingly. The outputs of the two matched filters are detected by two envelope detectors. The envelope detector is sampled at $t = T_0$ to make the ternary decision of -1 or 0 and 1 or 0 by negative and positive threshold devices, respectively. The output of summer is ternary, which is fed to a serial-parallel-serial converter, an inverse Sign Hadamard-Haar-q transform block and a parallel-serial converter to extract the original message.

The proposed non-coherent system is the simplest implementation of a ternary communication system. Other possibilities include the complicated M-ARY communication systems [12]. The addition of a Sign Hadamard-Haar transform provides security in the digital communication system. The level of security is easily adjustable by controlling q, which corresponds to Sign Hadamard-Haar-q transform applied q times. If q is varied for each word transformed in a manner transparent to a friendly receiver, the level of security in the communication system will be further enhanced. It is obvious that Sign Hadamard-Haar transform provides security to information data, however another possibility to increase the security of the digital communication system is the use of Sign Hadamard-γ-Haar-χ transform described in the introduction. Though the latter transform is more computationally expensive, it can be also in the form of Sign Hadamardγ-Haar-χ–q, transform and it provides better security properties by its design and is suitable for cryptographic systems.

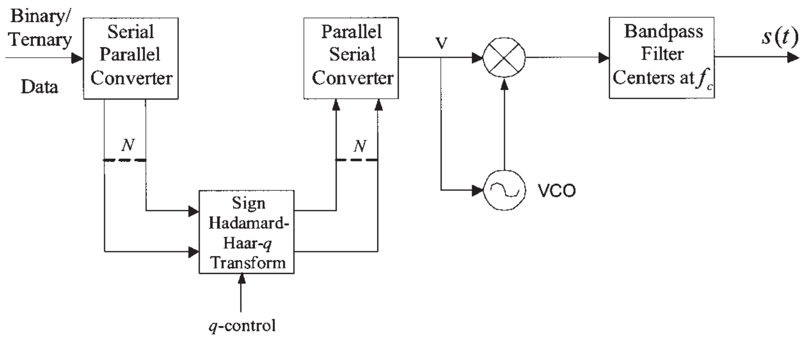

FIGURE 4
Block diagram of TAFSK transmitter.

The application of Sign Hadamard-Haar transform in a ternary communication system has been considered. Such application may be used by other quantized transforms as well. When the well known Sign Walsh transform is applied to the ternary communication system, Sign Hadamard-Haar-q transform will be replaced by Sign Walsh-q transform in the Figure 4 and Figure 5. Table 5 shows some results about computational costs of two systems where Sign Hadamard-Haar-q transform and Sign Walsh-q transform are implemented respectively. From Table 5, it is obvious that the system based on Sign Hadamard-Haar-q transform is much more efficient than the one based Sign Walsh-q transform due to much smaller number of computations especially for higher q and n.

TABLE 5
Comparison of computational costs in ternary communication system.

q	n	Sign Hadamard-Haar- q transform	Sign Walsh-q transform
		$(3\times 2^{n+1}-8)\times q$	$2qn2^n$
3	1	12	12
	2	48	48
	3	120	144
5	4	440	640
	5	920	1,600
	6	1,880	3,840
8	7	6,080	14,336
	8	12,224	32,768
	9	24,512	73,728
10	10	61,360	204,800
	11	122,800	450,560
	12	245,680	983,040
12	13	589,728	2,555,904
	14	1,179,552	5,505,024
	15	2,359,200	11,796,480

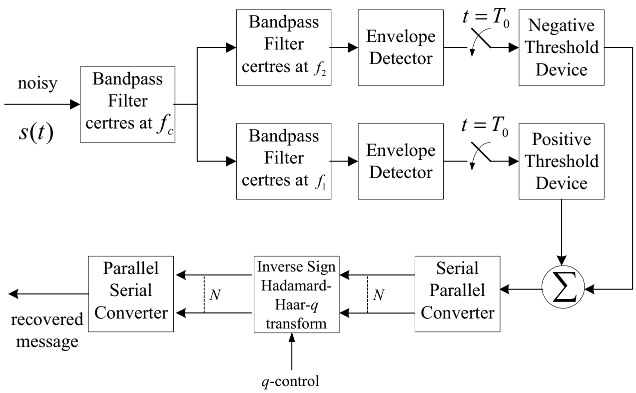

FIGURE 5
Block diagram of TAFSK receiver.

7. CONCLUSION

A novel non-linear transform called 'Sign Hadamard-Haar transform' has been introduced. It has been shown that the transform exhibits a non-linear property. Essentially, it transforms ternary data into ternary spectrum. Though non-linear, the transform is unique, and hence invertible. Recursive definition of the transform has been given. The procedures of obtaining the transform by recursive definition is increasingly cumbersome with the increase of the transform order $N = 2^n$. Fortunately, a fast butterfly signal-flow graph has been derived to facilitate the calculation of the transform in a very efficient way. Sign Hadamard-Haar transform can also be calculated using matrix approach that allows to calculate only some selected spectral coefficients. Such an approach is useful in many applications of spectral techniques such as testing or decomposition [17-20, 31, 33, 34]. Formulae to calculate Sign Hadamard-Haar spectra of logic functions and variables in two types of coding have also been developed.

Besides being extremely fast when compared with other ternary transforms, this transform exhibits another major advantage in the computer memory requirements necessary to store the calculated ternary spectrum. It requires exactly the same storage as the size of the original ternary data since only three different symbols have to be coded in the memory that is also property of ternary Reed-Muller transform and other quantized transforms. When the fast flow diagram is directly implemented in software there is no need to keep the original data and each consecutive butterfly requires geometrically smaller number of operations on the transformed data. Each time only 2^n ternary data has to be stored in the memory.

Sign Hadamard-Haar transform is just another representative of a family of quantized transforms based on the usage of sign function as the quantizer. In fact, with its unique and isomorphic properties, new sign transform can be derived using Sign Haar, Sign Walsh and Sign Hadamard-Haar transform as the three main bases so that a new transform space of similar ternary values is developed. As an example, one may develop a new recursive sign transform with just Sign Hadamard-γ-Haar-χ transform as the basis where γ and χ show the number of butterflies in Hadamard and Haar part of the transform. By simply applying Sign Hadamard-γ-Haar-χ transform twice onto a ternary truth column vector $\vec{F}$ such that a new transform space is developed, and name the overall transform as Sign Hadamard-γ-Haar-χ-2 transform. In general, if n is the number of variables of the ternary function, there are altogether 3^n different Sign Hadamard-γ-Haar-χ transform spaces, denoted as Sign Hadamard-γ-Haar-χ transform-q transforms with $1 \le q \le 3^n$ which is well suited to security coding in cryptographic systems and ternary communication systems, where when $q = 1$, $\gamma = 1$, and $\chi = n - 1$, the transform yields the original Sign Hadamard-Haar transform of size n.

There exist many generalizations of discrete transforms based on Walsh, Haar and Chrestenson functions [1, 17-19, 24-28, 31, 33-35, 37]. In the simplest generalization, the base functions are different linear combinations of Haar and Walsh functions. It is always possible to derive basic Walsh and Haar transform from their generalization. Hence it would be advantageous to develop the theory of generalized sign transforms where the known transforms, Sign Haar and Sign Walsh transforms together with Sign Hadamard-Haar transform would be just special cases of some generalized sign transform and some initial research in this aspect has been done in [10]. Development of a theory for generalized sign transforms together with investigations of basic properties of such transforms are topics of current research investigations of the authors. Considering a generalized sign transform as a Fourier-like or Chrestenson-like transform on a suitably defined algebraic structure offers a direct way for a generalization of the theory of such a transform from ternary logic functions to higher-level multiple-valued logic functions.

8. REFERENCES

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

[2] Besslich, Ph.W., and Trachtenberg, E.A., (Nov. 1996), 'Three-valued quasi-linear transformation for logic synthesis', *IEE Proceedings - Computers and Digital Techniques*, **143**, (6), pp. 391-400.

[3] Blake, I., Seroussi, G., Smart, N., (1999), *Elliptic Curves in Cryptography*, London Mathematical Society Lecture Note Series, vol. 265, Cambridge: Cambridge University Press.

[4] Coron, J.S., 'Resistance against differential power analysis for elliptic curve cryptosystems', *Workshop on Cryptographic Hardware and Embedded Systems*, (1999), published in Lecture Notes in Computer Science, vol. 1717, Berlin: Springer-Verlag, pp. 292-302.

[5] DiToro, N., (March 1991), *The Sign Transform: Its Application to Fault Detection in Digital Circuits and Its Efficient Implementation*, Master Thesis, Dept. of Electrical and Computer Engineering, Philadelphia: Drexel University.

[6] Falkowski, B.J., and Chang, C.H., (Nov. 1997), 'Forward and inverse transformations between Haar spectra and ordered binary decision diagrams for Boolean functions', *IEEE Transactions on Computers,* **46**, (11), pp. 1272-1279.

[7] Falkowski, B.J., and Kannurao, S., (Nov. 2002), 'Generation of sign Walsh spectra from disjoint cubes of Boolean functions', *Computers and Electrical Engineering*, **28**, (6), pp. 631-641.

[8] Falkowski, B.J., Lozano, C.C., and Rahardja, S., 'Calculation of ternary fixed-polarity Reed-Muller transform using column polarity matrix', *Proc. 16th European Conference on Circuit Theory and Design*, Krakow, Poland, Sept. 2003, vol. I, pp. 357-360.

[9] Falkowski, B.J., and Rahardja, S., 'Sign Haar transform', *Proc. IEEE 27th Int. Symposium on Circuits and Systems*, London, United Kingdom, May 1994, vol. 2, pp. 161-164.

[10] Falkowski, B.J., and Rahardja, S., (Mar. 1998), 'Recursive relations and fast flow diagram for non-linear quantized transform', *IEE Proceedings - Computers and Digital Techniques*, **145**, (2), pp. 97-107.

[11] Falkowski, B.J., and Yan, S., 'Sign Hadamard-Haar transform', *Proc. 16th European Conference on Circuit Theory and Design*, Krakow, Poland, Sept. 2003, vol. II, pp. 257-260.

[12] Fan, P., and Darnell, M., (1996), *Sequence Design for Communications Applications*, Taunton, Somerset, England: Research Studies Press Ltd. and New York: John Wiley & Sons Inc.

[13] Fei, B., Hong, Q., Wu, H., Perkowski, M.A. and Zhuang, N., (Oct. 1993), 'Efficient computation for ternary Reed-Muller expansions under fixed polarities', *International Journal of Electronics*, **75**, (4), pp. 685-688.

[14] Filippov, L.I., and Zinoviev, A.L., (1966), *Methods of Analytic Expression of Radio Communication Signals*, Moscow: Vysshaia Shkola, in Russian.

[15] Green, D.H., (Nov. 1989), 'Ternary Reed-Muller switching functions with fixed and mixed polarities', *International Journal of Electronics*, **67**, (5), pp. 761-775.

[16] Hansen, J.P., and Sekine, M., 'Decision diagrams based techniques for the Haar wavelet transform', *Proc. 1st Int. Conference on Information, Communications and Signal Processing*, Singapore, Sept. 1997, vol. 1, pp. 59-63.

[17] Hurst, S.L., (1992), *Custom VLSI Microelectronics,* New York: Prentice Hall.

[18] Karpovsky, M.G., (1976), *Finite Orthogonal Series in the Design of Digital Devices,* New York: John Wiley.

[19] Karpovsky, M.G., (1985), *Spectral Techniques and Fault Detection,* Orlando: Academic Press.

[20] Karpovsky, M.G., Stankovic, R.S., and Astola, J.T., 'Spectral techniques for design and testing of computer hardware', *Proc. 1st Int. Workshop on Spectral Techniques and Logic Design for Future Digital Systems*, Tampere, Finland, June 2000, Published as TICSP Series #10, Dec. 2000, pp. 9-42.

[21] Karpovsky, M.G., Stankovic, R.S., and Astola, J.T., (May 2003), 'Reduction of sizes of decision diagrams by autocorrelation functions', *IEEE Transactions on Computers*, **52**, (5), pp. 592-606.

[22] Meinel, C., Somenzi, F., Theobald, T., (May 2000), 'Linear sifting of decision diagrams and its application in synthesis', *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **19**, (5), pp. 521-533.

[23] Morain, F, and Olivos, J., (1990), 'Speeding up the computations on an elliptic curve using addition-subtraction chains', *Information Theory and Applications*, **24**, pp. 531-543.

[24] Moraga, C., 'Complex spectral logic', *Proc. 8th IEEE Int. Symposium on Multiple- Valued Logic*, Rosemont, Illinois, May 1978, pp. 149-156.

[25] Moraga, C., 'Systolic systems and multiple-valued logic', *Proc. 14th IEEE Int. Symposium on Multiple- Valued Logic*, Winnipeg, Canada, May 1984, pp. 98-108.

[26] Moraga, C., (1985), 'On some applications of the Chrestenson functions in logic design and data processing', *Mathematics and Computers in Simulation*, **27**, pp. 431-439.

[27] Rao, K.R., Jalali, A., and Yip, P., 'Rationalized Hadamard-Haar transform', *Proc. 11th Annual Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, California, Nov. 1977, pp. 194-203.

[28] Sasao, T., and Fujita, M., eds., (1996), *Representation of Discrete Functions*, Boston: Kluwer Academic.

[29] Sasao, T., (1999), *Switching Theory for Logic Synthesis*, Boston: Kluwer Academic.

[30] Schneier, B., (1996), *Applied Cryptography, Protocols, Algorithms, and Source Code in C*, New York: John Wiley.

[31] Stankovic, R.S. and Astola, J.T., (2003), *Spectral Interpretation of Decision Diagrams,* New York: Springer-Verlag.

[32] Stankovic, M., Jankovic, D., and Stankovic, R.S., (1996), 'Efficient algorithm for Haar spectrum calculation', *Scientific Review*, **21**, (2), pp. 171-182.

[33] Stankovic, R.S., Stankovic, M., and Jankovic, D., (1998), *Spectral Transforms in Switching Theory, Definitions and Calculations,* Belgrade: Science Publisher.

[34] Stankovic, R.S., Stojic, M.R., and Stankovic, M.S., eds., (1996), *Recent Development in Abstract Harmonic Analysis with Applications in Signal Processing,* Belgrade: Science Publisher.

[35] Stojic, M.R., Stankovic, M.S., and Stankovic, R.S., (1993), *Discrete Transforms and Their Application*, Belgrade: Science Publisher, in Serbian.

[36] Thornton, M.A., 'Modified Haar transform calculation using digital circuit output probabilities', *Proc. 1st Int. Conference on Information, Communications and Signal Processing*, Singapore, Sept. 1997, vol. 1, pp. 52-58.

[37] Yaroslavsky, L.P., (1985), *Digital Picture Processing – An Introduction*, Berlin: Springer-Verlag.