MULTILAYER HADAMARD DECOMPOSITION OF DISCRETE HARTLEY TRANSFORMS

H.M. DE OLIVEIRA, R.J. DE SOBRAL CINTRA, R.M. CAMPELLO DE SOUZA

CODEC-Grupo de Pesquisa em Comunicações, Departamento de Eletrônica e Sistemas
CTG-UFPE C.P.7800, 50711-970, Recife
E-mail: hmo@npd.ufpe.br, ricardo@npd.ufpe.br, rjsc@proplan.ufpe.br

ABSTRACT

Discrete transforms such as the Discrete Fourier Transform (DFT) or the Discrete Hartley Transform (DHT) furnish an indispensable tool in Signal Processing. The successful application of transform techniques relies on the existence of the so-called fast transforms. In this paper some fast algorithms are derived which meet the lower bound on the multiplicative complexity of a DFT/DHT. The approach is based on a decomposition of the DHT into layers of Hadamard-Walsh transforms. In particular, schemes named Turbo Fourier Transforms for short block lengths such as N=4, 8, 12 and 24 are presented.

1. INTRODUCTION

Discrete transforms defined over finite or infinite fields have been playing a relevant role in Engineering. A striking example is the Discrete Fourier Transform (DFT), which has found applications in several areas, especially in Electrical Engineering. Another relevant example concerns the Discrete Hartley Transform (DHT) [1], the discrete version of the integral transform introduced by R.V.L. Hartley in [2]. Besides its numerical side appropriateness, the DHT has proven over the years to be a powerful tool [3-5]. A decisive factor for applications of the DFT has been the existence of the so-called fast transforms (FT) for computing it [6]. Fast Hartley transforms also exist and are deeply connected to the DHT applications [7,8]. Recent promising applications of discrete transforms concern the use of finite field Hartley transforms [9] to design digital multiplex systems, efficient multiple access systems [10] and multilevel spread spectrum sequences [11].

Discrete transforms presenting a low multiplicative complexity have been an object of interest for a long time, including Arithmetic Fourier Transforms (AFT) [12]. Very efficient algorithms such as the Prime Factor Algorithm (PFA) or Winograd Fourier Transform Algorithm (WFTA) have also been used [13,14]. The minimal multiplicative complexity, μ, of the one-dimensional DFT for all possible sequence lengths, N, can be computed by converting the DFT into a set of multi-dimensional cyclic convolutions. A lower bound on the multiplicative complexity of a DFT is given in [15] (Theorem 5.4, p.98). For some short blocklengths, the values of μ(DFT(N)) are given in Table 1 (some local minima of μ).

The discrete Hartley transform of a signal v_i, i=0,1,2,...,N-1 is defined as

$$V_k \stackrel{\Delta}{=} \sum_{i=0}^{N-1} v_i cas\left(\frac{2\pi ki}{N}\right), \text{ k=0,1,2,...,N-1,} \qquad (1)$$

where cas(x)=cos(x)+sin(x) is the "cosine and sine" Hartley symmetric kernel.

In this paper, some FTs are presented, which meet the minimal multiplicative complexity. There is a simple relationship between the DHT and the DFT of a given real discrete signal f_i, i=0,1,...,N-1: If $f_i \leftrightarrow F_k$ is a DFT pair and $f_i \leftrightarrow H_k$ is the corresponding DHT pair, then [3] $\forall$k

$$H_k = \Re e F_k - \Im m F_k$$

and

$$F_k = \frac{1}{2}\left[\left(H_k + H_{N-k}\right) - j\left(H_k - H_{N-k}\right)\right].$$

Table 1. Minimal multiplicative complexity for computing a DFT of length N.

N	μ(DFT(N))
4	0
8	2
12	4
24	12

Therefore, an FFT algorithm for the DHT is also an FFT for the DFT and vice versa (see Corollary 6.9 [15]). Besides being a real transform, the DHT is also *involutionary*, i.e., the kernel of the inverse transform is exactly the same as the one of the direct transform (self-inverse transform). Since the DHT is a more symmetrical version of a discrete transform, this symmetry is exploited so as to derive a (turbo) FT that requires the minimal number of real floating point multiplications. The idea behind our approach is to carry out a DHT decomposition based on classical transforms by Jaques Hadamard [16].

2. COMPUTING A 4-BLOCKLENGTH DHT

Let **v** $\leftrightarrow$ **V** be a discrete Hartley transform pair of blocklength N=4. The matrix formulation of (1) corresponds to **V**=[T].**v**, i.e.,

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_0 \\ v_1 \\ v_2 \\ v_3 \end{bmatrix}.$$

It is therefore equivalent to a 4-point H/W transform so it has a zero-multiplicative complexity. Figure 2 shows a possible implementation of the 4-DHT in terms of 2-H/W transforms (Figure 1). The complexity for the 4-DHT is given by: A=8 additions and M=0 multiplications.

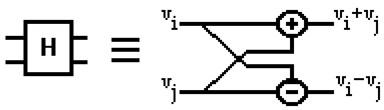

Figure 1. A free-multiplication diagram for a 2-H/W transform.

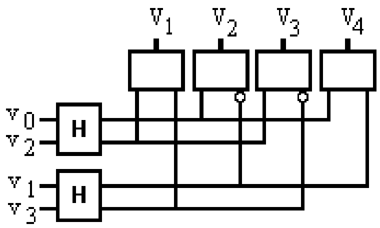

Figure 2. A free-multiplication diagram for a 4-DHT based on Hadamard-Walsh transform. The small circles on the summation boxes indicate a subtraction instead of an addition. "H" blocks denote a Hadamard transform.

3. COMPUTING AN 8-BLOCKLENGTH DHT

Let $\mathbf{v} \leftrightarrow \mathbf{V}$ be an 8-length Hartley Transform pair. The corresponding matricial formulation is $\mathbf{V}=[T]\cdot\mathbf{v}$ where [T] is a "cas" 8×8 matrix. Let $S_i(0)=v_i$, $\forall i=0,1,...,7$ (input data). The 0-order "pre-additions" are, respectively, $\{S_0(0)=v_0$, $S_1(0)=v_1$, $S_2(0)=v_2$, $S_3(0)=v_3$, $S_4(0)=v_4$, $S_5(0)=v_5$, $S_6(0)=v_6$, $S_7(0)=v_7\}$.

The transform matrix for the 8-DHT is therefore:

$$\begin{bmatrix} V_1\\ V_2\\ V_3\\ V_4\\ V_5\\ V_6\\ V_7\\ V_8 \end{bmatrix} = \begin{bmatrix} 1 & 1.4142 & 1 & 0 & -1 & -1.4142 & -1 & 0\\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1\\ 1 & 0 & -1 & 1.4142 & -1 & 0 & 1 & -1.4142\\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1\\ 1 & -1.4142 & 1 & 0 & -1 & 1.4142 & -1 & 0\\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1\\ 1 & 0 & -1 & -1.4142 & -1 & 0 & 1 & 1.4142\\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_0\\ v_1\\ v_2\\ v_3\\ v_4\\ v_5\\ v_6\\ v_7 \end{bmatrix}.$$

We start by remarking initially that

$$cas\left(\frac{2\pi k(i+N/2)}{N}\right) = cas\left(\frac{2\pi ki}{N}+\pi k\right) = (-1)^k cas\left(\frac{2\pi ki}{N}\right),$$

which follows from the addition of arcs formula: $cas(\alpha-\beta)=\cos\beta.cas\alpha-\sin\beta.cas'\alpha$, where cas'(.) is the complementary cas function, $cas'(\alpha)=\cos\alpha-\sin\alpha$ [3]. Clearly, moduli of components on the 2nd column are identical to the corresponding elements at the 6th column; the same for the 3th column and 7th column. We can thus consider new variables (v_1+v_5) and (v_1-v_5) instead of v_1 and v_5; (v_2+v_6) and (v_2-v_6) instead of v_2 and v_6, and so on.

•*1st-order pre-additions* (layer #1)
$\{S_0(1)=(v_0+v_4),\ S_1(1)=(v_0-v_4)\}$, $\{S_2(1)=(v_2+v_6),\ S_3(1)=(v_2-v_6)\}$, $\{S_4(1)=(v_1+v_5),\ S_5(1)=(v_1-v_5),\ S_6(1)=(v_3+v_7),\ S_7(1)=(v_3-v_7)\}$.

The first-order pre-additions as defined above always yield at least a half of vanishing elements in the new transform matrix. Although such an implementation requires only two multiplications, we may go further and combine other columns.

•*2nd-order pre-additions* (layer #2)
$\{S_0(2)=(v_0+v_4),\ S_1(2)=(v_0-v_4)\}$, $\{S_2(2)=(v_2+v_6),\ S_3(2)=(v_2-v_6)\}$, $\{S_4(2)=(v_1+v_5)+(v_3+v_7),\ S_5(2)=(v_1+v_5)-(v_3+v_7),$
$S_6(2)=(v_1-v_5)+(v_3-v_7),\ S_7(2)=(v_1-v_5)-(v_3-v_7)\}$.
Thus,

$$\begin{bmatrix} V_1\\ V_2\\ V_3\\ V_4\\ V_5\\ V_6\\ V_7\\ V_8 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & .707 & .707\\ 1 & 0 & -1 & 0 & 0 & 1 & 0 & 0\\ 0 & 1 & 0 & -1 & 0 & 0 & .707 & -.707\\ 1 & 0 & 1 & 0 & -1 & 0 & 0 & 0\\ 0 & 1 & 0 & 1 & 0 & 0 & -.707 & -.707\\ 1 & 0 & -1 & 0 & 0 & -1 & 0 & 0\\ 0 & 1 & 0 & -1 & 0 & 0 & -.707 & .707\\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} S_0(2)\\ S_1(2)\\ S_2(2)\\ S_3(2)\\ S_4(2)\\ S_5(2)\\ S_6(2)\\ S_7(2) \end{bmatrix}.$$

Clearly, some pre-additions terms involve a Walsh/Hadamard transform. A scheme for the implementation of an 8-DHT is shown in Figure 3. Only two multiplications by 0.707... are required.

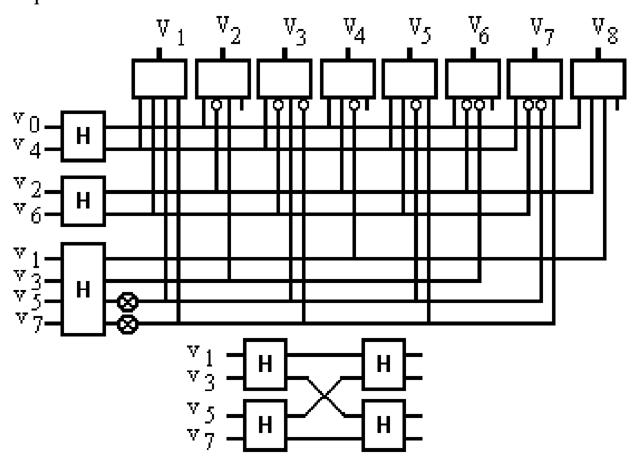

Figure 3. An 8-DHT/DFT diagram. (a) A fast transform with 2 multiplications (b) Detail of the 4-H/W transform.

The complexity for computing an 8-DHT is therefore: A=22 additions and M=2 multiplications.

4. COMPUTING A 12-BLOCKLENGTH DHT

Let $\mathbf{v} \leftrightarrow \mathbf{V}$ be a 12-DHT pair. The corresponding matrix formulation is now $\mathbf{V}=[T]\cdot\mathbf{v}$ where [T] is a 12×12 "cas" transform matrix and $\mathbf{V}=(V_1,V_2,...,V_{12})^T$. As usual, 0-order pre-additions (data) are defined as $S_i(0)=v_i$, $\forall i=0,1,...,N-1$.

Denoting by [T(0)] the transforming matrix, the Hartley spectrum can be computed according to $\mathbf{V}=[T(0)]\cdot\mathbf{S}(0)$ where $\mathbf{S}(0)=(S_0(0),S_1(0),...,S_{11}(0))^T$. Applying the same reasoning of Section 2, we define:

•*1st-order pre-additions* (layer #1)

$\{S_0(1)=v_0+v_6,\ S_1(1)=v_0-v_6\},\ \{S_2(1)=v_3+v_9,\ S_3(1)=v_3-v_9\}$

$\{S_4(1)=v_1+v_7,\ S_5(1)=v_1-v_7,\ S_6(1)=v_2+v_8,\ S_7(1)=v_2-v_8,$

$S_8(1)=v_4+v_{10},\ S_9(1)=v_4-v_{10},\ S_{10}(1)=v_5+v_{11},\ S_{11}(1)=v_5-v_{11}\}.$

The resulting transform is:

$$\begin{bmatrix} V_1\\ V_2\\ V_3\\ V_4\\ V_5\\ V_6\\ V_7\\ V_8\\ V_9\\ V_{10}\\ V_{11}\\ V_{12} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1.366 & 0 & 1.366 & 0 & .366 & 0 & -.366\\ 1 & 0 & -1 & 0 & 1.366 & 0 & .366 & 0 & -1.366 & 0 & .366 & 0\\ 0 & 1 & 0 & -1 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & 1\\ 1 & 0 & 1 & 0 & .366 & 0 & -1.366 & 0 & .366 & 0 & -1.366 & 0\\ 0 & 1 & 0 & 1 & 0 & -.366 & 0 & -.366 & 0 & -1.366 & 0 & 1.366\\ 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 1 & 0 & -1 & 0\\ 0 & 1 & 0 & -1 & 0 & -1.366 & 0 & 1.366 & 0 & .366 & 0 & .366\\ 1 & 0 & 1 & 0 & -1.366 & 0 & .366 & 0 & -1.366 & 0 & .366 & 0\\ 0 & 1 & 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & -1\\ 1 & 0 & -1 & 0 & -.366 & 0 & -1.366 & 0 & .366 & 0 & 1.366 & 0\\ 0 & 1 & 0 & -1 & 0 & .366 & 0 & -.366 & 0 & -1.366 & 0 & -1.366\\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} v_0+v_6\\ v_0+v_6\\ v_3+v_9\\ v_3-v_9\\ v_1+v_7\\ v_1-v_7\\ v_2+v_8\\ v_2-v_8\\ v_4+v_{10}\\ v_4-v_{10}\\ v_5+v_{11}\\ v_5-v_{11} \end{bmatrix}.$$

Therefore, this equation can be written as $\mathbf{V}=[T(1)].\mathbf{S}(1)$ where $\mathbf{S}(1)=(S_0(1),S_1(1),...,S_{11}(1))^T$. Observing the remaining symmetries, we go further and define:

•*2nd-order pre-additions* (layer #2)

$\{S_0(2)=v_0+v_6,\ S_1(2)=v_0-v_6\},\ \{S_2(2)=v_3+v_9,\ S_3(2)=v_3-v_9\},$

$\{S_4(2)=(v_1+v_7)+(v_4+v_{10}),\ S_5(2)=(v_1+v_7)-(v_4+v_{10}),$

$S_6(1)=(v_1-v_7)+(v_2-v_8),\ S_7(2)=(v_1-v_7)-(v_2-v_8),$

$S_8(2)=(v_2+v_8)+(v_5+v_{11}),\ S_9(2)=(v_2+v_8)-(v_5+v_{11}),$

$S_{10}(2)=(v_4-v_{10})+(v_5-v_{11}),\ S_{11}(2)=(v_4-v_{10})-(v_5-v_{11})\}.$

We have then

$$\begin{bmatrix} V_1\\ V_2\\ V_3\\ V_4\\ V_5\\ V_6\\ V_7\\ V_8\\ V_9\\ V_{10}\\ V_{11}\\ V_{12} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1.366 & 0 & 0 & 0 & 0 & .366\\ 1 & 0 & -1 & 0 & 0 & 1.366 & 0 & 0 & 0 & .366 & 0 & 0\\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0\\ 1 & 0 & 1 & 0 & .366 & 0 & 0 & 0 & -1.366 & 0 & 0 & 0\\ 0 & 1 & 0 & 1 & 0 & 0 & -.366 & 0 & 0 & 0 & 0 & -1.366\\ 1 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0\\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & -1.366 & 0 & 0 & .366 & 0\\ 1 & 0 & 1 & 0 & -1.366 & 0 & 0 & 0 & .366 & 0 & 0 & 0\\ 0 & 1 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1\\ 1 & 0 & -1 & 0 & 0 & -.366 & 0 & 0 & 0 & -1.366 & 0 & 0\\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & .366 & 0 & 0 & -1.366 & 0\\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} v_0+v_6\\ v_0+v_6\\ v_3+v_9\\ v_3-v_9\\ (v_1+v_7)+(v_4+v_{10})\\ (v_1+v_7)-(v_4+v_{10})\\ (v_1-v_7)+(v_2-v_8)\\ (v_1-v_7)-(v_2-v_8)\\ (v_2+v_8)+(v_5+v_{11})\\ (v_2+v_8)-(v_5+v_{11})\\ (v_4-v_{10})+(v_5-v_{11})\\ (v_4-v_{10})-(v_5-v_{11}) \end{bmatrix}.$$

The spectrum can be computed in terms of the 2^{nd} layer pre-additions as $\mathbf{V}=[T(2)].\mathbf{S}(2)$ where [T(2)] is the 12×12 matrix above and $\mathbf{S}(2)=(S_0(2),S_1(2),...,S_{11}(2))^T$. There is no pair of non-combined identical columns left (signs of elements not considered). However, the integer part of the elements greater than unity into the [T(2)] matrix can be handled separately.

Spectral component calculation (special ADD to balance the matrix):

V1	→	$[(v_1-v_7)+(v_2-v_8)]=S_6(2)$
V2	→	$[(v_1+v_7)-(v_4+v_{10})]=S_5(2)$
V3	→	0
V4	→	$-[(v_2+v_8)+(v_5+v_{11})]=-S_8(2)$
V5	→	$-[(v_4-v_{10})-(v_5-v_{11})]=-S_{11}(2)$
V6	→	0
V7	→	$-[(v_1-v_7)-(v_2-v_8)]=-S_7(2)$
V8	→	$-[(v_1+v_7)+(v_4+v_{10})]=-S_4(2)$
V9	→	0
V10	→	$-[(v_2+v_8)-(v_5+v_{11})]=-S_9(2)$
V11	→	$-[(v_4-v_{10})+(v_5-v_{11})]=-S_{10}(2)$
V12	→	0

The procedure of combining pair of columns can be iterated yielding the following new pre-addition sets:

•*3rd-order pre-additions* (layer #3)

$\{S_0(3)=v_0+v_6,\ S_1(3)=v_0-v_6\},\{S_2(3)=v_3+v_9,\ S_3(3)=v_3-v_9\}$

$S_4(3)=\{(v_1+v_7)+(v_4+v_{10})\}+\{(v_2+v_8)+(v_5+v_{11})\}$

$S_5(3)=\{(v_1+v_7)+(v_4+v_{10})\}-\{(v_2+v_8)+(v_5+v_{11})\}$

$S_6(3)=\{(v_1+v_7)-(v_4+v_{10})\}+\{(v_2+v_8)-(v_5+v_{11})\}$

$S_7(3)=\{(v_1+v_7)-(v_4+v_{10})\}-\{(v_2+v_8)-(v_5+v_{11})\}$

$S_8(3)=\{(v_1-v_7)-(v_2-v_8)\}+\{(v_4-v_{10})+(v_5-v_{11})\}$

$S_9(3)=\{(v_1-v_7)-(v_2-v_8)\}-\{(v_4-v_{10})+(v_5-v_{11})\}$

$S_{10}(3)=\{(v_1-v_7)+(v_2-v_8)\}+\{(v_4-v_{10})-(v_5-v_{11})\}$

$S_{11}(3)=\{(v_1-v_7)+(v_2-v_8)\}-\{(v_4-v_{10})-(v_5-v_{11})\}$

The final relationship between the Hartley spectrum and the pre-additions can be established:

$$\begin{bmatrix} V_1\\ V_2\\ V_3\\ V_4\\ V_5\\ V_6\\ V_7\\ V_8\\ V_9\\ V_{10}\\ V_{11}\\ V_{12} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & .366 & 0\\ 1 & 0 & -1 & 0 & 0 & 0 & .366 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0\\ 1 & 0 & 1 & 0 & 0 & .366 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 1 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -.366 & 0\\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -.366 & 0 & 0\\ 1 & 0 & 1 & 0 & 0 & -.366 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1\\ 1 & 0 & -1 & 0 & 0 & 0 & -.366 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & .366 & 0 & 0\\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} S_0(3)\\ S_1(3)\\ S_2(3)\\ S_3(3)\\ S_4(3)\\ S_5(3)\\ S_6(3)\\ S_7(3)\\ S_8(3)\\ S_9(3)\\ S_{10}(3)\\ S_{11}(3) \end{bmatrix} + \begin{bmatrix} S_6(2)\\ S_5(2)\\ 0\\ -S_8(2)\\ -S_{11}(2)\\ 0\\ -S_7(2)\\ -S_4(2)\\ 0\\ -S_9(2)\\ -S_{10}(2)\\ 0 \end{bmatrix}$$

The only four real floating-point multiplications required are $0.366\times\{S_5(3),\ S_6(3),\ S_9(3),\ S_{10}(3)\}$. A corresponding block diagram is sketched in Figure 4 below.

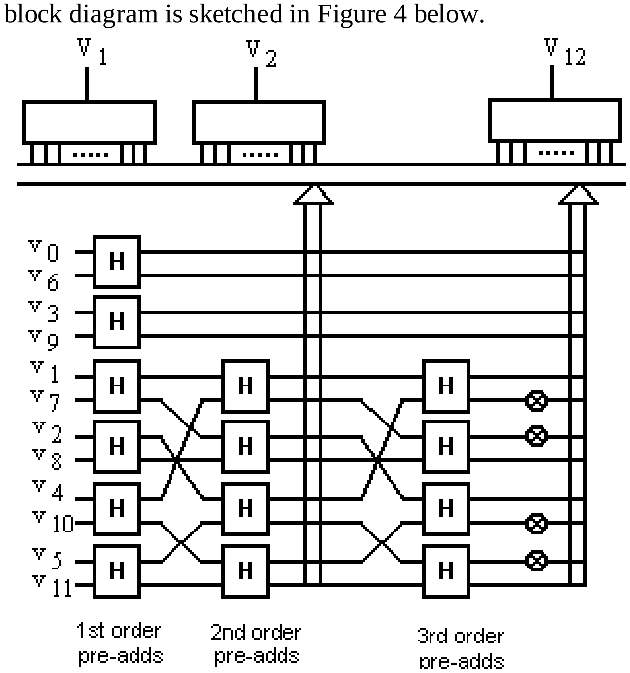

Figure 4. A 12-DHT/DFT fast transform diagram.

The complexity of such an implementation of a 12-DHT is given by: A=52 additions and M=4 multiplications.

5. COMPUTING A 24-BLOCKLENGTH DHT

Let $\mathbf{v}\leftrightarrow\mathbf{V}$ be a 24-DHT pair. Let us denote by $\mathbf{V}=[T]\cdot\mathbf{v}$ the matrix formulation of the DTH, where [T] represents the 24×24 cas-transform matrix and $\mathbf{V}$ is the discrete Hartley output spectrum. Following the same steps as before, the 0-order pre-additions are defined as $S_i(0)=v_i$, i = 0, …, 23. We have then Equation A (see Appendix).

Going further, the 1st-order pre-additions (layer #1) will be:

$S_0(1)=v_0+v_{12}$, $S_1(1)=v_0-v_{12}$, $S_2(1)=v_1+v_{13}$, $S_3(1)=v_1-v_{13}$, $S_4(1)=v_2+v_{14}$, $S_5(1)=v_2-v_{14}$, $S_6(1)=v_3+v_{15}$, $S_7(1)=v_3-v_{15}$, $S_8(1)=v_4+v_{16}$, $S_9(1)=v_4-v_{16}$, $S_{10}(1)=v_5+v_{17}$, $S_{11}(1)=v_5-v_{17}$, $S_{12}(1)=v_6+v_{18}$, $S_{13}(1)=v_6-v_{18}$, $S_{14}(1)=v_7+v_{19}$, $S_{15}(1)=v_7-v_{19}$, $S_{16}(1)=v_8+v_{20}$, $S_{17}(1)=v_8-v_{20}$, $S_{18}(1)=v_9+v_{21}$, $S_{19}(1)=v_9-v_{21}$, $S_{20}(1)=v_{10}+v_{22}$, $S_{21}(1)=v_{10}-v_{22}$, $S_{22}(1)=v_{11}+v_{23}$, $S_{23}(1)=v_{11}-v_{23}$.

A new set of pre-addition can be considered. Let the 2nd-order pre-additions be:

$S_0(2)=S_0(1)$, $S_1(2)=S_1(1)$, $S_2(2)=S_{12}(1)$, $S_3(2)=S_{13}(1)$, $S_4(2)=S_2(1)+S_{14}(1)$, $S_5(2)=S_2(1)-S_{14}(1)$, $S_6(2)=S_3(1)+S_{11}(1)$, $S_7(2)=S_3(1)-S_{11}(1)$, $S_8(2)=S_4(1)+S_{16}(1)$, $S_9(2)=S_4(1)-S_{16}(1)$, $S_{10}(2)=S_5(1)+S_9(1)$, $S_{11}(2)=S_5(1)-S_9(1)$, $S_{12}(2)=S_8(1)+S_{20}(1)$, $S_{13}(2)=S_8(1)-S_{20}(1)$, $S_{14}(2)=S_{10}(1)+S_{22}(1)$, $S_{15}(2)=S_{10}(1)-S_{22}(1)$, $S_{16}(2)=S_{15}(1)+S_{23}(1)$, $S_{17}(2)=S_{15}(1)-S_{23}(1)$, $S_{18}(2)=S_{17}(1)+S_{21}(1)$, $S_{19}(2)=S_{17}(1)-S_{21}(1)$, $S_{20}(2)=S_6(1)+S_{18}(1)$, $S_{21}(2)=S_6(1)-S_{18}(1)$, $S_{22}(2)=S_7(1)+S_{19}(1)$, $S_{23}(2)=S_7(1)-S_{19}(1)$.

Again, we have a few cases where the pair do not match perfectly. Applying the same strategy adopted in the 12-blocklength case, we put apart some matrix components in order to "balance" the matrix. The 3rd-order pre-additions follows:

$S_0(3)=S_0(2)$, $S_1(3)=S_1(2)$, $S_2(3)=S_2(2)$, $S_3(3)=S_3(2)$, $S_4(3)=S_{20}(2)$, $S_5(3)=S_{21}(2)$, $S_6(3)=S_4(2)+S_{12}(2)$, $S_7(3)=S_4(2)-S_{12}(2)$, $S_8(3)=S_5(2)+S_9(2)$, $S_9(3)=S_5(2)-S_9(2)$, $S_{10}(3)=S_8(2)+S_{14}(2)$, $S_{11}(3)=S_8(2)-S_{14}(2)$, $S_{12}(3)=S_{13}(2)+S_{15}(2)$, $S_{13}(3)=S_{13}(2)-S_{15}(2)$, $S_{14}(3)=S_{22}(2)+S_{23}(2)$, $S_{15}(3)=S_{22}(2)-S_{23}(2)$, $S_{16}(3)=S_{10}(2)+S_{19}(2)$, $S_{17}(3)=S_{10}(2)-S_{19}(2)$, $S_{18}(3)=S_{11}(2)+S_{18}(2)$, $S_{19}(3)=S_{11}(2)-S_{18}(2)$, $S_{20}(3)=S_6(2)$, $S_{21}(3)=S_7(2)$, $S_{22}(3)=S_{16}(2)$, $S_{23}(3)=S_{17}(2)$.

The "special add" vector required in this step is written in Equation B (see Appendix).

The simplification is not finished yet. The procedure of combining matched columns must be called once more. Making the following definitions, we get the final 4th-order pre-addition, remarking that — as in the last iteration — another "special add" vector must be putted aside, yielding:

$S_0(4)=S_0(3)$, $S_1(4)=S_1(3)$, $S_2(4)=S_2(3)$, $S_3(4)=S_3(3)$, $S_4(4)=S_4(3)$, $S_5(4)=S_5(3)$, $S_6(4)=S_{17}(3)$, $S_7(4)=S_{18}(3)$, $S_8(4)=S_6(3)+S_{10}(3)$, $S_9(4)=S_8(3)+S_{13}(3)$, $S_{10}(4)=S_8(3)-S_{13}(3)$, $S_{11}(4)=S_6(3)-S_{10}(3)$, $S_{12}(4)=S_9(3)+S_{12}(3)$, $S_{13}(4)=S_7(3)+S_{11}(3)$, $S_{14}(4)=S_7(3)-S_{11}(3)$, $S_{15}(4)=S_9(3)-S_{12}(3)$, $S_{16}(4)=S_{14}(3)+S_{23}(3)$, $S_{17}(4)=S_{14}(3)-S_{23}(3)$, $S_{18}(4)=S_{15}(3)+S_{21}(3)$, $S_{19}(4)=S_{15}(3)-S_{21}(3)$, $S_{20}(4)=S_{16}(3)$, $S_{21}(4)=S_{19}(3)$, $S_{22}(4)=S_{20}(3)$, $S_{23}(4)=S_{22}(3)$.

Deriving the DHT in terms of the fourth pre-addition layer, we obtain Equation C (see Appendix).

As we get only twelve floating-point multiplication, the theoretic lower bound on the number of multiplications required to compute a DFT is again achieved. The corresponding block diagram is depicted below.

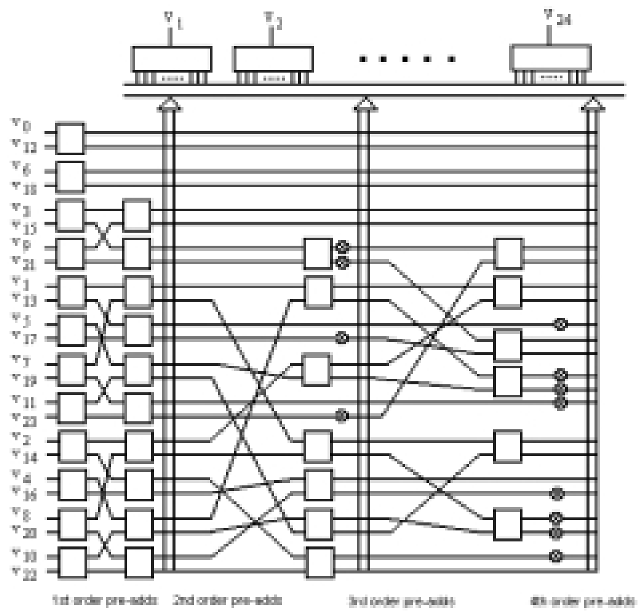

Figure 5. Diagram of a 24-DHT/DFT with 12 multiplications for circuit implementation.

The complexity of such an implementation of a 24-DHT is given by: A=138 additions and M=12 multiplications.

6. CONCLUSIONS

Some Fast Transforms are derived which achieve the lower bound on the multiplicative complexity of a DFT/DHT. In particular, new schemes named Turbo Hartley Transforms (THT) for short block lengths are presented. They are based on a multilayer decomposition of the DHT using Hadamard-Walsh transforms. Each HWT implements pre-additions. These schemes are attractive and easy to implement using a Digital Signal Processor (DSP) or low-cost high-speed dedicated Integrated Circuits.

7. APPENDIX

This appendix shows some matrices obtained in the 24-DHT algorithm derivation. Equation A corresponds to the 24×24 cas-transform matrix; Equation B shows an intermediate matrix with three layers; and Equation C presents the final matix formulation, which achieves minimal multiplication complexity, i.e., only 12 real floating point multiplications.

$$
\begin{bmatrix} V_0 \\ V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \\ V_7 \\ V_8 \\ V_9 \\ V_{10} \\ V_{11} \\ V_{12} \\ V_{13} \\ V_{14} \\ V_{15} \\ V_{16} \\ V_{17} \\ V_{18} \\ V_{19} \\ V_{20} \\ V_{21} \\ V_{22} \\ V_{23} \end{bmatrix} =
\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1.224 & 1.366 & 1.414 & 1.366 & 1.224 & 1 & .707 & .366 & 0 & -.366 & -.707 & -1 & -1.224 & -1.366 & -1.44 & -1.366 & -1.224 & -1 & -.707 & -.366 & 0 & .366 & .707 \\
1 & 1.366 & 1.366 & 1 & .366 & -.366 & -1 & -1.366 & -1.366 & -1 & -.366 & .366 & 1 & 1.366 & 1.366 & 1 & .366 & -.366 & -1 & -1.366 & -1.366 & -1 & -.366 & .366 \\
1 & 1.414 & 1 & 0 & -1 & -1.414 & -1 & 0 & 1 & 1.414 & 1 & 0 & -1 & -1.414 & -1 & 0 & 1 & 1.414 & 1 & 0 & -1 & -1.414 & -1 & 0 \\
1 & 1.366 & .366 & -1 & -1.366 & -.366 & 1 & 1.366 & .366 & -1 & -1.366 & -.366 & 1 & 1.366 & .366 & -1 & -1.366 & -.366 & 1 & 1.366 & .366 & -1 & -1.366 & -.366 \\
1 & 1.224 & -.366 & -1.414 & -.366 & 1.224 & 1 & -.707 & -1.366 & 0 & 1.366 & .707 & -1 & -1.224 & .366 & 1.414 & .366 & -1.224 & -1 & .707 & 1.366 & 0 & -1.366 & -.707 \\
1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\
1 & .707 & -1.366 & 0 & 1.366 & -.707 & -1 & 1.224 & .366 & -1.414 & .366 & 1.224 & -1 & -.707 & 1.366 & 0 & -1.366 & .707 & 1 & -1.224 & -.366 & 1.414 & -.366 & -1.224 \\
1 & .366 & -1.366 & 1 & .366 & -1.366 & 1 & .366 & -1.366 & 1 & .366 & -1.366 & 1 & .366 & -1.366 & 1 & .366 & -1.366 & 1 & .366 & -1.366 & 1 & .366 & -1.366 \\
1 & 0 & -1 & 1.414 & -1 & 0 & 1 & -1.414 & 1 & 0 & -1 & -1.414 & -1 & 0 & 1 & -1.414 & 1 & 0 & -1 & 1.414 & -1 & 0 & 1 & -1.414 \\
1 & -.366 & -.366 & 1 & -1.366 & 1.366 & -1 & .366 & .366 & -1 & 1.366 & -1.366 & 1 & -.366 & -.366 & 1 & -1.366 & 1.366 & -1 & .366 & .366 & -1 & 1.366 & -1.366 \\
1 & -.707 & .366 & 0 & -.366 & .707 & -1 & 1.224 & -1.366 & 1.414 & -1.366 & 1.224 & -1 & .707 & -.366 & 0 & .366 & -.707 & 1 & -1.224 & 1.366 & -1.414 & 1.366 & -1.224 \\
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\
1 & -1.224 & 1.366 & -1.414 & 1.366 & -1.224 & 1 & -.707 & .366 & 0 & -.366 & .707 & -1 & 1.224 & -1.366 & 1.414 & -1.366 & 1.224 & -1 & .707 & -.366 & 0 & .366 & -.707 \\
1 & -1.366 & 1.366 & -1 & .366 & .366 & -1 & 1.366 & -1.366 & 1 & -.366 & -.366 & 1 & -1.366 & 1.366 & -1 & .366 & .366 & -1 & 1.366 & -1.366 & 1 & -.366 & -.366 \\
1 & -1.414 & 1 & 0 & -0 & 1.414 & -1 & 0 & 1 & -1.414 & 1 & 0 & -1 & 1.414 & -1 & 0 & 1 & -1.414 & 1 & 0 & -1 & 1.414 & -1 & 0 \\
1 & -1.366 & .366 & 1 & -1.366 & .366 & 1 & -1.366 & .366 & 1 & -1.366 & .366 & 1 & -1.366 & .366 & 1 & -1.366 & .366 & 1 & -1.366 & .366 & 1 & -1.366 & .366 \\
1 & -1.224 & -.366 & 1.414 & -.366 & -1.224 & 1 & .707 & -1.366 & 0 & 1.366 & -.707 & -1 & 1.224 & .366 & -1.414 & .366 & 1.224 & -1 & .707 & 1.366 & 0 & -1.366 & .707 \\
1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\
1 & -.707 & -1.366 & 0 & 1.366 & .707 & -1 & -1.224 & .366 & 1.414 & .366 & -1.224 & -1 & .707 & 1.366 & 0 & -1.366 & -.707 & 1 & 1.224 & -.366 & -1.414 & .366 & 1.224 \\
1 & -.366 & -1.366 & -1 & .366 & 1.366 & 1 & -.366 & -1.366 & -1 & .366 & 1.366 & 1 & -.366 & -1.366 & -1 & .366 & 1.366 & 1 & -.366 & -1.366 & -1 & .366 & 1.366 \\
1 & 0 & -1 & -1.414 & -1 & 0 & 1 & 1.414 & 1 & 0 & -1 & -1.414 & -1 & 0 & 1 & 1.414 & 1 & 0 & -1 & -1.414 & -1 & 0 & 1 & 1.414 \\
1 & .366 & -.366 & -1 & -1.366 & -1.366 & -1 & -.366 & .366 & 1 & 1.366 & 1.366 & 1 & .366 & -.366 & -1 & -1.366 & -1.366 & -1 & -.366 & .366 & 1 & 1.366 & 1.366 \\
1 & .707 & .366 & 0 & -.366 & -.707 & -1 & -1.224 & -1.366 & -1.414 & -1.366 & -1.224 & -1 & -.707 & -.366 & 0 & .366 & .707 & 1 & 1.224 & 1.366 & 1.414 & 1.366 & 1.224
\end{bmatrix} \cdot
\begin{bmatrix} v_0 \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \\ v_9 \\ v_{10} \\ v_{11} \\ v_{12} \\ v_{13} \\ v_{14} \\ v_{15} \\ v_{16} \\ v_{17} \\ v_{18} \\ v_{19} \\ v_{20} \\ v_{21} \\ v_{22} \\ v_{23} \end{bmatrix}
$$

Equation A: The transform matrix for a 24-blocklength DHT.

$$
\begin{bmatrix} V_0 \\ V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \\ V_7 \\ V_8 \\ V_9 \\ V_{10} \\ V_{11} \\ V_{12} \\ V_{13} \\ V_{14} \\ V_{15} \\ V_{16} \\ V_{17} \\ V_{18} \\ V_{19} \\ V_{20} \\ V_{21} \\ V_{22} \\ V_{23} \end{bmatrix} =
\begin{bmatrix}
1 && 1 && 1 && 1 &&&& 1 &&&&& &&&&& &&& \\
& 1 && 1 &&&&& &&&&& & .707 && .366 &&&& 1.224 &&& .707 \\
1 && -1 &&& 1 &&& 1.366 &&&&& .366 &&&&& &&&&& \\
& 1 && -1 &&&&& &&&&& && .707 &&& 1 &&& 1.414 && \\
1 && 1 && -1 && 1.366 &&&&& .366 &&&&& &&&&& && \\
& 1 && 1 &&&&& &&&&& & -.707 && -.366 &&&& 1.224 &&& -.707 \\
1 && -1 &&& -1 &&&& 1 &&& 1 &&&&& &&&&& & \\
& 1 && -1 &&&&& &&&&& && -.707 &&&& -.366 && .707 & 1.224 & \\
1 && 1 && 1 && .366 &&&& -1.366 &&&&& &&&&& &&& \\
& 1 && 1 &&&&& &&&&& & .707 &&& -1 &&&&&& -1.414 \\
1 && -1 &&& 1 &&& -.366 &&&&& -1.366 &&&&& &&&&& \\
& 1 && -1 &&&&& &&&&& && .707 &&&& .366 && -.707 & 1.224 & \\
1 && 1 && -1 &&& -1 &&&& 1 &&&&& &&&&& && \\
& 1 && 1 &&&&& &&&&& & -.707 && .366 &&&& -1.224 &&& -.707 \\
1 && -1 &&& -1 &&&& -1.366 &&& .366 &&&&& &&&&& & \\
& 1 && -1 &&&&& &&&&& && -.707 &&& 1 &&& -1.414 && \\
1 && 1 && 1 && -1.366 &&&& .366 &&&&& &&&&& &&& \\
& 1 && 1 &&&&& &&&&& & .707 && -.366 &&&& -1.224 &&& .707 \\
1 && -1 &&& 1 &&& -1 &&&&& 1 &&&&& &&&&& \\
& 1 && -1 &&&&& &&&&& && .707 &&&& -.366 && -.707 & -1.224 & \\
1 && 1 && -1 &&& -.366 &&&& -1.366 &&&&& &&&&& && \\
& 1 && 1 &&&&& &&&&& & -.707 &&& -1 &&&&&& 1.414 \\
1 && -1 &&& -1 &&&& .366 &&& -1.366 &&&&& &&&&& & \\
& 1 && -1 &&&&& &&&&& && -.707 &&&& .366 && .707 & -1.224 &
\end{bmatrix} \cdot
\begin{bmatrix} S_0(2) \\ S_1(2) \\ S_2(2) \\ S_3(2) \\ S_{20}(2) \\ S_{21}(2) \\ S_4(2)+S_{12}(2) \\ S_4(2)-S_{12}(2) \\ S_5(2)+S_9(2) \\ S_5(2)-S_9(2) \\ S_8(2)+S_{14}(2) \\ S_8(2)-S_{14}(2) \\ S_{13}(2)+S_{15}(2) \\ S_{13}(2)-S_{15}(2) \\ S_{22}(2)+S_{23}(2) \\ S_{22}(2)-S_{23}(2) \\ S_{10}(2)+S_{19}(2) \\ S_{10}(2)-S_{19}(2) \\ S_{11}(2)+S_{18}(2) \\ S_{11}(2)-S_{18}(2) \\ S_6(2) \\ S_7(2) \\ S_{16}(2) \\ S_{17}(2) \end{bmatrix} +
\begin{bmatrix} 0 \\ S_{10}(2) \\ 0 \\ 0 \\ 0 \\ -S_9(2) \\ 0 \\ -S_{11}(2) \\ 0 \\ 0 \\ 0 \\ -S_{18}(2) \\ 0 \\ S_{10}(2) \\ 0 \\ 0 \\ 0 \\ -S_{19}(2) \\ 0 \\ -S_{11}(2) \\ 0 \\ 0 \\ 0 \\ -S_{18}(2) \end{bmatrix}
$$

Equation B: 24-blocklength DHT with three Hadamard layers.

$$
\begin{bmatrix} V_0 \\ V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \\ V_7 \\ V_8 \\ V_9 \\ V_{10} \\ V_{11} \\ V_{12} \\ V_{13} \\ V_{14} \\ V_{15} \\ V_{16} \\ V_{17} \\ V_{18} \\ V_{19} \\ V_{20} \\ V_{21} \\ V_{22} \\ V_{23} \end{bmatrix} =
\begin{bmatrix}
1 && 1 && 1 &&&& 1 &&&&& &&&&& &&&&& \\
& 1 && 1 &&&&& &&&&& &&& .707 &&&& .366 && 1.224 & \\
1 && -1 &&& 1 &&&&& &&&& .366 &&&&& &&&& \\
& 1 && -1 &&&& 1 &&&&& &&&&& & .707 &&&&& \\
1 && 1 && -1 &&&&& &&&& .366 &&&&& &&&&& \\
& 1 && 1 &&&&& &&&&& &&& -.707 &&&& -.366 && 1.224 & \\
1 && -1 &&& -1 &&&&&& 1 &&&&& &&&&& && \\
& 1 && -1 &&&&& &&&&& &&&&& & -.707 && -.366 && 1.224 \\
1 && 1 && 1 &&&&& &&& .366 &&&&& &&&&& & \\
& 1 && 1 &&& -1 &&&&& &&&&& & .707 &&&&& & \\
1 && -1 &&& 1 &&&&& &&&& -.366 &&&&& &&&& \\
& 1 && -1 &&&&& &&&&& &&&&& & .707 && .366 && 1.224 \\
1 && 1 && -1 &&&&& -1 &&&&& &&&&& &&&& \\
& 1 && 1 &&&&& &&&&& &&& -.707 &&&& .366 && -1.224 & \\
1 && -1 &&& -1 &&&&& &&&&& -.366 &&&&& &&& \\
& 1 && -1 &&&&& &&&&& &&&&& -.707 &&&&& \\
1 && 1 && 1 &&&&& &&& -.366 &&&&& &&&&& & \\
& 1 && 1 &&&&& &&&&& &&& .707 &&&& -.366 && -1.224 & \\
1 && -1 &&& 1 &&&&& -1 &&&&& &&&&& &&& \\
& 1 && -1 &&&&& &&&&& &&&&& & .707 && -.366 && -1.224 \\
1 && 1 && -1 &&&&& &&&& -.366 &&&&& &&&&& \\
& 1 && 1 &&& -1 &&&&& &&&&& & -.707 &&&&& & \\
1 && -1 &&& -1 &&&&& &&&&& .366 &&&&& &&& \\
& 1 && -1 &&&&& &&&&& &&&&& & -.707 && .366 && -1.224
\end{bmatrix} \cdot
\begin{bmatrix} S_0(4) \\ S_1(4) \\ S_2(4) \\ S_3(4) \\ S_4(4) \\ S_5(4) \\ S_6(4) \\ S_7(4) \\ S_8(4) \\ S_9(4) \\ S_{10}(4) \\ S_{11}(4) \\ S_{12}(4) \\ S_{13}(4) \\ S_{14}(4) \\ S_{15}(4) \\ S_{16}(4) \\ S_{17}(4) \\ S_{18}(4) \\ S_{19}(4) \\ S_{20}(4) \\ S_{21}(4) \\ S_{22}(4) \\ S_{23}(4) \end{bmatrix} +
\begin{bmatrix} 0 \\ 0 \\ S_8(3) \\ .707 \cdot S_{21}(3) \\ S_7(3) \\ 0 \\ 0 \\ 0 \\ -S_{10}(3) \\ -.707 \cdot S_{23}(3) \\ -S_{13}(3) \\ 0 \\ 0 \\ 0 \\ -S_9(3) \\ -.707 \cdot S_{21}(3) \\ -S_6(3) \\ 0 \\ 0 \\ 0 \\ -S_{11}(3) \\ .707 \cdot S_{23}(3) \\ -S_{12}(3) \\ 0 \end{bmatrix} +
\begin{bmatrix} 0 \\ S_{10}(2) \\ 0 \\ 0 \\ 0 \\ -S_9(2) \\ 0 \\ -S_{11}(2) \\ 0 \\ 0 \\ 0 \\ -S_{18}(2) \\ 0 \\ S_{10}(2) \\ 0 \\ 0 \\ 0 \\ -S_{19}(2) \\ 0 \\ -S_{11}(2) \\ 0 \\ 0 \\ 0 \\ -S_{18}(2) \end{bmatrix}
$$

Equation C: 24-blocklength DHT with four Hadamard layers and minimal multiplicative complexity.

8. REFERENCES

[1] R.N. Bracewell, The Discrete Hartley Transform, *J. Opt. Soc. Amer*., vol.**73**, pp.1832-1835,Dec., 1983.

[2] R.V.L. Hartley, A More Symmetrical Fourier Analysis Applied to Transmission Problems, *Proc. IRE*, vol.**30**, pp.144-150, Mar., 1942.

[3] R.N. Bracewell, *The Hartley Transform,* Oxford University Press, 1986.

[4] K.J. Olejniczak, G.T. Heydt Eds., *Proc. of the IEEE*, Section on the Hartley Transform, vol.**82**, n.3, Mar., pp.372-447, 1994.

[5] J.-L. Wu and Shiu, Discrete Hartley Transform in Error Control Coding, *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-**39**, pp.2356-2359, Oct., 1991.

[6] R.E. Blahut, *Fast Algorithms for Digital Signal Processing,* Addison-Wesley, 1985.

[7] G. Bi and Y.Q. Chen, Fast DHT Algorithms for Length $N=q*2^m$, *IEEE Trans. on Signal Processing*, vol. **47**, n.3, Mar., pp.900-903, 1999.

[8] M. Popovic and D. Stevié, A New Look at the Comparison of the Fast Hartley and Fourier Transforms, *IEEE Trans. on Signal Processing*, vol.**42**, n.8, Aug., pp.2178-2182, 1994.

[9] R.M. Campello de Souza, H.M. de Oliveira, A.N. Kauffman and A.J.A. Paschoal, Trigonometry in Finite Fields and a New Hartley Transform, *Proceedings of the 1998 IEEE Intern. Symp. on Info. Theory*, p.293, Cambridge, MA, Aug., 1998.

[10] H.M. de Oliveira, R.M. Campello de Souza and A.N. Kauffman, Efficient Multiplex for Band-limited Channels: Galois Division Multiple Access, *Proceedings of the 1999 Workshop on Coding and Cryptography*, WCC-99, pp.235-241, Paris, Jan., 1999.

[11] H.M. de Oliveira and R.M. Campello de Souza, Orthogonal Multilevel Spreading Sequence Design, *5th Intern. Symp. on Communications Theory and Application*, ISCTA, Ambleside, UK, 11-16 July, 1999.

[12] I.S. Reed *et al*., Fourier Analysis and Signal Processing by use of the Moebius Inversion Formula, *IEEE Trans. Acoustics, Speach and Signal Processing*, ASSP-**38**, pp.459-470, Mar., 1990.

[13] S. Winograd, On Computing the Discrete Fourier Transform, *Math. Comp*., vol.**32**, pp.175-199, 1978.

[14] D. Yang, Prime Factor Fast Hartley Transform, *Elect. Letters*, Jan.,vol.**26**, n.2, pp. 119-121,1990.

[15] M.T. Heideman, *Multiplicative Complexity, Convolution, and the DFT,* Springer-Verlag, 1988.

[16] J. Hadamard, Resolution d'une Question Relative aux Determinants, *Bull. Sci. Math.*, Ser. 2, vol. **17**, Part I, pp. 240-246.