Discrete Hartley Transform using Recursive Algorithm

Vivek Singh, Dhwani Kaushal & Priyanka Jain

Delhi Technological University, Delhi, India

ABSTRACT: This paper presents a recursive algorithm for the computation of discrete Hartley transform for N = 2.m where m is an even integer. Necessary formulas for the computation have been derived along with the realization of recursive structure. DHT is an important transform used to convert data in the time domain to the frequency domain using real values only. With the help of a recursive algorithm, we can compute DHT by using a minimum number of adders and multipliers.

Keywords—recursive structure, Infinite response filter structure (IIR), discrete Hartley transform

1 INTRODUCTION

Discrete Hartley Transform (DHT) is a powerful mathematical tool used for processing discrete data, and its unique capabilities make it suitable for a variety of applications. In contrast to the Discrete Fourier Transform (DFT), DHT has no complex values and involves only real computations. In the Fast Fourier Transform (FFT) method, one complex multiplication is equivalent to four real multiplications; while the computation of the DHT does not involve anything like this [1]. Also, the real and imaginary parts of DFT can be obtained from the even and odd parts of DHT, directly.

The Hartley transform can also be used to compute discrete Fourier, discrete cosine, discrete sine, and discrete Hilbert transforms [12]. Discrete Hartley Transform combined with the Walsh-Hadamard transform is also used to compute the Fast Walsh-Hadamard-Hartley transform [9]. Hartley Transform can be used for many applications, like for Visible Light Communication Systems using adaptively biased OFDM [13]. Progressive image secret sharing can be accomplished through the utilization of the DHT, providing different advantages compared to other existing methodologies [10]. In applications with the requirement of a power spectrum, straight the DHT can be used, without the need for DFT [7].

The direct computation of the discrete Hartley transform (DHT) incurs a heavy computational overhead, demanding N^2 computations i.e., multiplications and additions. Researchers have developed many fast algorithms for the computation of DHT, one of them is the fast Hartley Transform developed by Bracewell [2]. Hartley transform is better than Fourier since it is two to four times faster than Fourier transform [6]. The FHT algorithm developed by Bracewell causes a significant reduction in the computational complexity of the DHT, making it more practical for real-time applications, but hardware complexity is very high, making its implementation difficult on hardware architecture. Specialized hardware architectures can be designed for computing DHT efficiently, combining the fast algorithms with specialized hardware structures has made DHT a viable alternative to the DFT for various applications that involve signal processing and the computation complexity will be $N^2/2$ for all values of k.

In this paper, a new recursive algorithm for DHT has been presented with its recursive structure. The paper is organized as section 2 discusses the algorithm for DHT, in section 3, the realization of the algorithm through IIR filter structure is presented, Performance comparison of the suggested structure with the existing work in the literature is done in section 4, and section 5 deals with the conclusion.

2 ALGORITHM FOR RECURSIVE FORMULA FOR DHT

N point Discrete Hartley transform is express as [8]:

$$\mathrm{X}[k] = \sum_{n=0}^{N-1} x[n]\, cas\left(\frac{2\pi nk}{N}\right) \tag{1}$$

where k = 0, 1, 2 ... N-1 and $\mathrm{cas}\theta = \sin\theta + \cos\theta$

$$H[k] = \sum_{n=0}^{N/2-1} x[n] cas\left(\frac{2\pi kn}{N}\right) + \sum_{n=N/2}^{N-1} x[n] cas\left(\frac{2\pi kn}{N}\right) \tag{2}$$

where k = 0, 1, 2 ... N-1

Put $n' = n + \frac{N}{2}$ in the second term of the equation (2)

$$H[k] = \sum_{n=0}^{N/2-1} x[n] cas\left(\frac{2\pi kn}{N}\right) + \sum_{n=0}^{N/2-1} x\left[n+\frac{N}{2}\right] cas\left(\frac{2\pi k\left(n+\frac{N}{2}\right)}{N}\right) \tag{3}$$

where k = 0, 1, 2 ... N-1

Solving the right term of equation (3) and expending the cas term, we obtain,

$$\sum_{n=0}^{N/2-1} x\left[n+\frac{N}{2}\right]\left(\cos\left(\frac{2\pi Kn}{N}+\pi K\right) + \sin\left(\pi k+\frac{2\pi kn}{N}\right)\right) \tag{4}$$

where k = 0, 1, 2 ... N-1

Using the following identities in equation (4):

$$\sin(A+B) = \sin A\cos B + \cos A\sin B;\ \cos(A+B) = \cos A\cos B - \sin A\sin B$$

$$\sum_{n=0}^{N/2-1} x\left[n+\frac{N}{2}\right]\begin{pmatrix} cos\left(\frac{2\pi kn}{N}\right) cos(\pi k) - sin\left(\frac{2\pi kn}{N}\right) sin(\pi k) \\ +sin\left(\frac{2\pi kn}{N}\right) cos(\pi k) + cos\left(\frac{2\pi kn}{N}\right) sin(\pi k) \end{pmatrix} \tag{5}$$

where k = 0, 1, 2 ... N-1

In equation (5), $\sin(\pi k) = 0$; $\cos(\pi k) = \left(-1^k\right)$; where k = 0, 1, 2 ... N-1

Equation (5) reduces to

$$\sum_{n=0}^{N/2-1} x\left[n+\frac{N}{2}\right]\left((-1)^k\left(cas\left(\frac{2\pi kn}{N}\right)\right)\right) \tag{6}$$

where k = 0, 1, 2 ... N-1

Putting the equation (6) back in equation (3)

$$H[k] = \sum_{n=0}^{N/2-1} x[n] cas\left(\frac{2\pi kn}{N}\right) + \sum_{n=0}^{N/2-1} x\left[n+\frac{N}{2}\right](-1)^k\, cas\left(\frac{2\pi kn}{N}\right) \tag{7}$$

where k = 0, 1, 2 ... N-1

Therefore, equation (7) becomes

$$H[k] = \sum_{n=0}^{N/2-1} \left[x(n) + (-1)^k x\left(n+\frac{N}{2}\right)\right]\left(cas\left(\frac{2\pi kn}{N}\right)\right) \tag{8}$$

where k = 0, 1, 2 ... N-1

Assume $W_k[n] = x(n) + (-1)^k x\left(n+\frac{N}{2}\right)$

$$X[k] = \sum_{n=0}^{N/2-1} W_k[n] cas\left(\frac{2\pi nk}{N}\right) \tag{9}$$

where k = 0, 1, 2 ... N-1

Put $n = \frac{N}{2} - 1 - n$

$$X[k] = \sum_{n=0}^{N/2-1} W_k\left[\frac{N}{2} - 1 - n\right] cas\left(\pi k - \frac{2\pi k}{N}(n+1)\right) \tag{10}$$

where k = 0, 1, 2 ... N-1

Using trigonometric identities used above in:

$$\sum_{n=0}^{N/2-1} W_k\left[\frac{N}{2} - 1 - n\right] x \begin{pmatrix} cos(\pi k)\, cos\frac{2\pi k(n+1)}{N} + sin(\pi k)\, sin\frac{2\pi k}{N}(n+1) \\ + sin(\pi k)\, cos\frac{2\pi k}{N}(n+1) - cos(\pi k)\, sin\frac{2\pi k}{N}(n+1) \end{pmatrix} \tag{11}$$

where k = 0, 1, 2 ... N-1

In equation (11), $\sin(\pi k) = 0$, $\cos(\pi k) = \left(-1^k\right)$ and $cms\theta = sin\,\theta - cos\,\theta$

$$X[k] = (-1)^k \sum_{n=0}^{N/2-1} W_k\left[\frac{N}{2} - 1 - n\right] cms\left(\frac{2\pi k}{N}(n+1)\right) \tag{12}$$

where k = 0, 1, 2 ... N-1

$$X[k] = (-1)^k . G_i[k] \tag{13}$$

where k = 0, 1, 2 ... N-1

Let $i = \mathrm{N}/2 - 1$ and $\theta_k = (2\pi\mathrm{k}/\mathrm{N})$

$$G_i[k] = \sum_{n=0}^{i} W_k[i-n]\, cms\big((n+1)\theta_k\big) \tag{14}$$

where k = 0, 1, 2 ... N-1

$$G_i[k] = \sum_{n=0}^{i} W_k[i-n]\left[\cos\big((n+1)\theta_k\big) - \sin\big((n+1)\theta_k\big)\right] \tag{15}$$

where k = 0, 1, 2 ... N-1

Using trigonometric identities used above, in equation (15), we get

$$G_i[k] = \sum_{n=0}^{i} W_k[i-n]\begin{bmatrix} \big(2\cos(n\,\theta_k)\cos\theta_k - \cos\big((n-1)\theta_k\big)\big) \\ -\big(2\sin(n\,\theta_k)\cos\theta_k - \sin\big((n-1)\theta_k\big)\big) \end{bmatrix} \tag{16}$$

where k = 0, 1, 2 ... N-1

$$G_i[k] = 2\cos\theta_k \sum_{n=0}^{i} W_k[i-n]\, cms(n\,\theta_k) - \sum_{n=0}^{i} W_k[i-n]\, cms\big((n-1)\theta_k\big) \tag{17}$$

where k = 0, 1, 2 ... N-1

$$G_i[k] = \begin{bmatrix} 2\cos\theta_k \left[W_k[i] + \sum_{n=0}^{i-1} W_k[i-1-n]\, cms\big((n+1)\theta_k\big)\right] \\ -\begin{bmatrix} W_k[i]\, cms(-\theta_k) + W_k[i-1]\,.(1) \\ + \sum_{n=0}^{i-2} W_k[i-2-n]\, cms\big((n+1)\theta_k\big) \end{bmatrix} \end{bmatrix} \tag{18}$$

where k = 0, 1, 2 ... N-1

From equation (14), we can write

$$G_{i-1}[k] = \sum_{n=0}^{i-1} W_k[i-1-n]\, cms\big((n+1)\theta_k\big) \tag{19}$$

where k = 0, 1, 2 ... N-1

$$G_{i-2}[k] = \sum_{n=0}^{i-2} W_k[i-2-n]\, cms\big((n+1)\theta_k\big) \tag{20}$$

where k = 0, 1, 2 ... N-1

Using equation. (19) and equation (20) in equation (21), we obtain a recursive from as:

$$G_i[k] = 2\cos\theta_k \left[W_k[i] + G_{i-1}[k]\right] - \begin{bmatrix} W_k[i]\, cms(-\theta_k) \\ +W_k[i-1]\,.(1) + G_{i-2}[k] \end{bmatrix} \tag{21}$$

where k = 0, 1, 2 ... N-1

$$G_i[k] = W_k[i]\cos(\theta_k) + 2\, G_{i-1}[k]\cos\theta_k - W_k[i]\sin(\theta_k) - W_k[i-1] - G_{i-2}[k] \tag{22}$$

where k = 0, 1, 2 ... N-1

3 REALIZATION OF HARDWARE

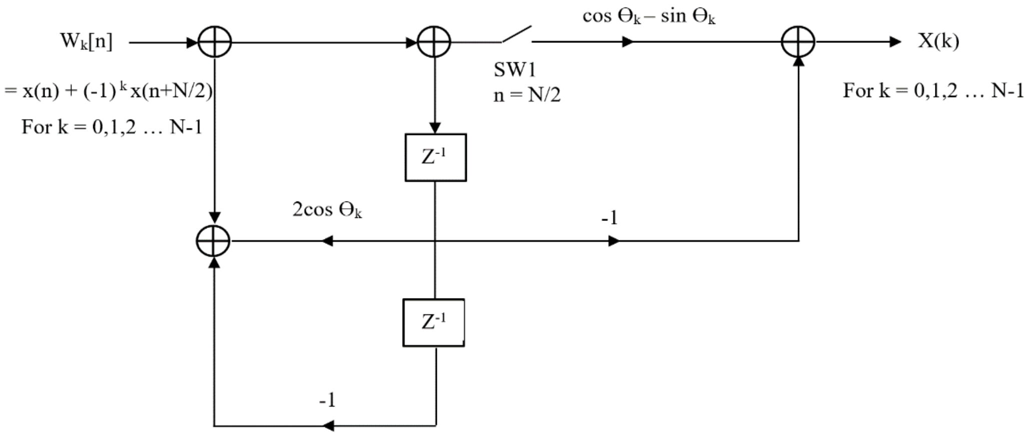

Figure 1. Recursive Structure for the computation of DHT

It can be inferred from the recursive structure that the hardware requirement for the realization of discrete Hartley transform is only 3 adders,2 delay elements, and 2 multipliers. Another advantage over here is that this same structure can be used to compute DHT of length N, where N = 2.m where m is an even integer.

4 PERFORMANCE

The proposed algorithm for the computation of DHT has low hardware complexity, because of which a lesser number of multipliers and adders are used in the proposed structure when compared to the Ref. [3][4][5][8][11]. The comparison is depicted in table 1 and table 2.

Table 1. Comparison of algorithm proposed for computation of DHT in terms of number of adders

Length(N)	Proposed	[3]	[4]	[5]	[8]	[11]
4	3	--	--	--	--	7
6	3	--	--	--	--	15
8	3	16	16	19	38	--
16	3	33	67	35	78	--
32	3	64	205	67	158	--
64	3	128	553	131	318	--

Table 2. Comparison of algorithm proposed for computation of DHT in terms of number of multipliers

Length(N)	Proposed	[3]	[4]	[5]	[8]	[11]
4	2	--	--	--	--	6
6	2	--	--	--	--	12
8	2	32	2	18	32	--
16	2	64	12	34	64	--
32	2	128	40	66	128	--
64	2	256	112	130	256	--

5 CONCLUSION

A recursive algorithm and its realization through a recursive filter structure are proposed in the work. The hardware complexity of the structure is less as compared to the other existing structures for DHT, so the suggested algorithm and its structure could be used for any real-life applications of digital processing.

6 REFERENCES

[1] Bracewell, ‘Discrete Hartley transform,’ J. Opt. Soc. Am. 73, 1832-1835 (1983)

[2] Bracewell,. ‘The fast Hartley transform.’ Proceedings of the IEEE 72: 1010-1018. (1984)

[3] Chang, L.W., and Lee, S.W.: ‘Systolic arrays for the discrete Hartley transform’, IEEE Trans., ASP-39, (ll), pp. 2411-2418, , (Nov. 1992)

[4] Chiper, ‘A Novel VLSI DHT Algorithm for a Highly Modular and Parallel Architecture’, in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 60, no. 5, pp. 282-286, (May 2013)

[5] Fang and Lee, ‘Efficient CORDIC-based systolic architectures for the discrete Hartley transform,’ IEE Proc. – Comput. Digit. Tech., vol. 142, pp. 201-207,(May 1995)

[6] Hong, Huang, Xiao, Li, Sun, and Zhang. 2023. ‘Programmable In-memory Computing Circuit of Fast Hartley Transform.’, ACM Trans. Des. Autom. Electron. Syst. 28, 6, Article 100(October 2023)

[7] Jain, Kumar and Bala Jain, ‘Discrete sine transform and its inverse—realization through recursive algorithms,’ Int. J. Circ. Theor. Appl.; 36:441–449, (2008)

[8] Liu, Ray, and Chiu, C.-T.: ‘Unified parallel lattice structures for time-recursive discrete wsine/sine/Hartley transforms’, IEEE Trans., ASP-41, 9 pp. 1357-1377, (March 1993)

[9] Mardan & Hamood. ‘New fast Walsh–Hadamard–Hartley transform algorithm.’ International Journal of Electrical and Computer Engineering (IJECE).13.1533.10.11591/ijece.v13i2 .pp1533-1540, (2023)

[10] M Holla, Suma, ‘Progressive Hartley image secret sharing for high-quality image recovery’, Cogent Engineering, 10:2, 2262805, (2023)

[11] Murty, ‘Novel Recursive Algorithm For Realization Of One-Dimensional Discrete Hartley Transform’, International Journal of Research and Reviews in Applied Sciences, 10, (2012)

[12] Nascimento, ‘Hartley Transform Signal Compression and Fast Power Quality Measurements for Smart Grid Application,’ in IEEE Transactions on Power Delivery, vol. 38, no. 6, pp. 4134-4144, (Dec. 2023)

[13] Wu, Xie, Shi, Zhang, Yao, Liu, ‘An Adaptively Biased OFDM Based on Hartley Transform for Visible Light Communication Systems’, IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Article ID 2023EAL2059, Advance online publication (September 20, 2023)