INTEGER FFT WITH OPTIMIZED COEFFICIENT SETS

Wei-Hsin Chang and Truong Nguyen

Department of Electrical and Computer Engineering, UCSD
La Jolla, California 92093-0407

ABSTRACT

In this paper, the principle of finding the optimized coefficient set of integer fast Fourier transform (IntFFT) is introduced. IntFFT has been regarded as an approximation of original FFT since it utilizes lifting scheme (LS) and decomposes the complex multiplication of twiddle factor into three lifting steps. Based on the observation of the quantization loss model of lifting operations, we can select an optimized coefficient set and achieve better signal-to-quantization-noise ratio (SQNR). A mixed-radix 128-point FFT is used to compare the SQNR performance between IntFFT and other FFT implementations. A fixed-point simulation environment with the presence of additive white Gaussian noise (AWGN) channel is also constructed for comparison purposes.

Index Terms— Fast Fourier Transform, Integer FFT, Quantization Loss Analysis

I. INTRODUCTION

The fast Fourier transform (FFT) plays a significant role in orthogonal frequency division multiplexing (OFDM) systems such as wireless LAN (802.11 a/g), multi-band OFDM (MB-OFDM), digital audio broadcasting (DAB) and digital video broadcasting (DVB-T, DVB-H). Considering the actual hardware implementation, the accuracy of FFT/IFFT module is an important design factor of system performance. In practice, fixed-point arithmetic is used to implement FFT in hardware because it is not possible to keep infinite resolution of coefficients and operations. All coefficients and input signals have to be presented with finite number of bits in binary format depending on the tradeoff between the hardware cost (memory usage) and the accuracy of output signals. Besides the direct implementation of the complex multiplier, the numerical decomposition [1], coordinate digital computer (CORDIC) algorithm and IntFFT [2] are also utilized for the design of FFT architecture and have different advantages. For example, the numerical decomposition consists of only three multiplications and three additions with one additional coefficient look-up table. CORDIC algorithm rewrites the conventional complex multiplication with a series of angle rotations. It is particularly useful when no multiplier unit is available. During the past decade, lifting factorization has been developed as a useful tool for the construction of wavelets [3]. In [2], the idea of lifting scheme is further applied to compute the non-trivial complex multiplication where the original operation is expressed as three lifting steps. Therefore, it is interesting to analyze the performance comparison between these four implementations of complex multiplier in terms of SQNR.

This work is supported by Texas Instruments Inc.

The rest of this paper is organized as follows. In section II, different implementations of complex multiplier are reviewed. In section III, the quantization loss of LS-based complex multiplier and numerical decomposition are derived. The selection criteria of optimized lifting coefficient set and its advantages are also presented. In section IV, the fixed-point simulation of all complex multipliers is illustrated. Finally we conclude this work in section V.

II. PRELIMINARY WORK

Given an input sequence $x(n)$, the N-point discrete Fourier transform (DFT) is defined as

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot W_N^{nk}, \quad k = 0, 1, ..., N-1. \tag{1}$$

where n is the time index, k is the frequency index, and the twiddle factor W_N^{nk} is defined as

$$W_N^{nk} = exp(\frac{-2j\pi nk}{N}) = cos(\frac{2\pi nk}{N}) - j \cdot sin(\frac{2\pi nk}{N}). \tag{2}$$

In order to reduce the arithmetic complexity as well as hardware cost, the main idea of all FFT algorithms is to extract as many twiddle factors as possible. Several FFT algorithms have been proposed including radix-2 [4], radix-4 [5], radix-2^2 [6], and split radix [7] FFT. Regardless of which FFT algorithm is adopted, the complex multiplier can be realized by different approaches. Numerical decomposition rewrites the complex multiplication using three real multiplications and three additions. Let $t = c + js$ be a complex number with magnitude one (i.e. $|t| = 1$), and $x = x_r + jx_i$. Then the real part and the imaginary part can be written as Eq. (3) and Eq. (4) respectively. Both $c + s$ and $c - s$ are assumed to be pre-calculated.

$$cx_r - sx_i = x_r(c - s) + s(x_r - x_i) \tag{3}$$
$$cx_i + sx_r = x_i(c + s) + s(x_r - x_i). \tag{4}$$

The CORDIC algorithm is a hardware friendly method to calculate trigonometric functions. Its basic idea is to calculate the angle rotation as a series of smaller incremental rotations. Therefore, the desired rotation angle, Θ, is decomposed into a set of pre-defined elementary angles as:

$$\Theta = \sum_{n=0}^{N_c-1} \mu_n a_n + \epsilon, \tag{5}$$

where N_c is the number of elementary angles, $\mu_i \in \{1, -1\}$ is used to indicate the direction of angle rotations, a_i is the ith elementary angle, and ϵ is the difference between the real angle and the approximated angle. The CORDIC algorithm has been extended and utilized for FFT implementations to perform complex

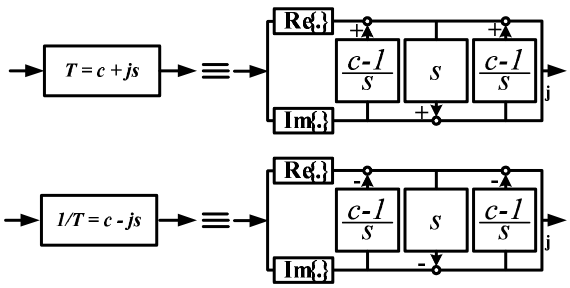

Fig. 1. Represent original twiddle factors as three lifting steps.

multiplications with only additions and shifts. By calculating the product of all cosine values in advance, each iteration only involves one addition and one subtraction. In other words, the complex multiplication is represented as

$$y = [x_r \quad x_i] \prod_n^{Nc} \begin{bmatrix} \cos a_n & \sin a_n \\ -\sin a_n & \cos a_n \end{bmatrix}$$
$$= \underbrace{(\prod_n^{N_c} \cos a_n)}_{pre-calculated} [x_r \quad x_i] \prod_n^{N_c} \begin{bmatrix} 1 & \tan a_n \\ -\tan a_n & 1 \end{bmatrix}. \quad (6)$$

From the viewpoint of hardware design, the advantage of CORDIC FFT is that only additions and shifts are required. It also leads to smaller hardware cost for ASIC designs because no general multiplier is needed. However, it inevitably degrades the output accuracy since it is an approximation of the original complex multiplication due to the presence of angle difference, ϵ.

IntFFT [2] is an integer-to-integer mapping transform in which the wavelet transform concept of lifting scheme is utilized to FFT algorithms. If we represent the complex multiplication of twiddle factors in a matrix form as:

$$tx = (cx_r - sx_i) + j(cx_i + sx_r)$$
$$= \begin{bmatrix} 1 & j \end{bmatrix} \begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} x_r \\ x_i \end{bmatrix} = \begin{bmatrix} 1 & j \end{bmatrix} R \begin{bmatrix} x_r \\ x_i \end{bmatrix} \quad (7)$$

Furthermore, the R matrix shown in Eq. (7) can be decomposed into three lifting steps [3].

$$R = \begin{bmatrix} c & -s \\ s & c \end{bmatrix} = \begin{bmatrix} 1 & \frac{c-1}{s} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ s & 1 \end{bmatrix} \begin{bmatrix} 1 & \frac{c-1}{s} \\ 0 & 1 \end{bmatrix} \quad (8)$$

In this equation, s and c represent $\sin\theta$ and $\cos\theta$ respectively, where θ is the angle of twiddle factor as defined in Eq. (2). As a result, the original complex multiplications of multiplying twiddle factors can be represented in lifting scheme form as shown in Fig. 1. Notice that when the angle of the twiddle factor, θ_n, is small, the value of lifting coefficients in the above decomposition may become very large. Different decompositions of lifting scheme are derived to avoid this problem [2]. As listed in table I, four possible different types of lifting decompositions can be used interchangeably.

Table I. Four equivalent decompositions of lifting scheme.

	Type (a)	Type (b)	Type (c)	Type (d)
P	$P_a = \frac{c-s}{s}$	$P_b = \frac{c+1}{s}$	$P_c = \frac{s-1}{c}$	$P_d = \frac{s+1}{c}$
Q	$Q_a = s$	$Q_b = -s$	$Q_c = c$	$Q_d = -c$

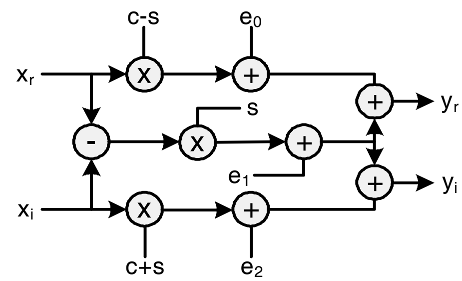

Fig. 2. The quantization loss model of the numerical decomposition.

III. OPTIMIZED SET OF LIFTING COEFFICIENTS

III-A. Quantization Loss Model

The additive noise model of quantization loss is widely adopted to measure the effect of the fixed length operations in digital signal processing systems [5]. The quantized product can be expressed as the sum of unquantized product and an uniformly-distributed additive quantization noise, e. For the numerical decomposition as expressed in Eq. (3) and Eq. (4), the equivalent quantization loss model is depicted in Fig. 2. Without loss of generality, we assume there is no loss from the pre-calculated coefficients, $c+s$ and $c-s$. All the quantization noise sources, e_i, are statistically independent and have the same variance δ^2. Define $Y = Y_r + jY_i$ as the original output and $Q(Y)$ as the quantized output, the mean square error (MSE) of numerical decomposition is expressed as in Eq. (9). Similarly, the MSE of direct implementation can also be simply derived as $4\sigma^2$.

$$E_N[|Q(Y) - Y|^2] = E[|(e_0 + e_1) + j(e_2 + e_1)|^2]$$
$$= E[e_0^2] + 2E[e_1^2] + E[e_2^2] = 4\sigma^2 \quad (9)$$

The equivalent quantization loss model of LS-based complex multiplier is illustrated in Fig. 3. The lifting coefficients $P[n]$ and $Q[n]$ are generated from corresponding twiddle factors, where both $P[n]$ and $Q[n]$ are real-valued discrete functions, and n is the index of twiddle factors. Again, we assume all the noise sources are independent and uniformly distributed with variance σ^2. The overall variance of the quantization loss is a function of $P[n]$ and can be expressed as Eq. (10) [8]. As we can see, the MSE of LS-based complex multiplier depends on the selected $P[n]$ coefficients. Recall from section II that four equivalent decompositions of lifting scheme can be used alternatively. It is possible to reduce the quantization loss by choosing proper $P[n]$ coefficients.

III-B. Periodicity of Lifting Coefficients

Since lifting coefficients are the functions of sinusoids, we use the periodicity property of sinusoids to present lifting coefficients

$$E_{LS}[|Q(Y)-Y|^2] = E[[(1+p[n]q[n])e_1 + p[n]e_2 + e_3]^2 + (q[n]e_1 + e_2)^2] = [(\frac{\cos(\omega n)-1}{\sin(\omega n)})^2 + 3]\cdot\sigma^2 = (p[n]^2+3)\sigma^2 \qquad (10)$$

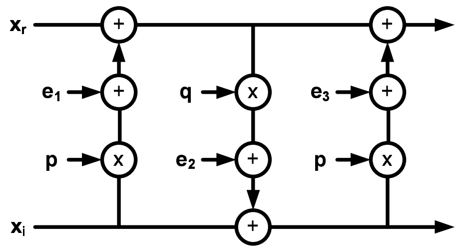

Fig. 3. The equivalent quantization loss model of LS-based complex multiplier.

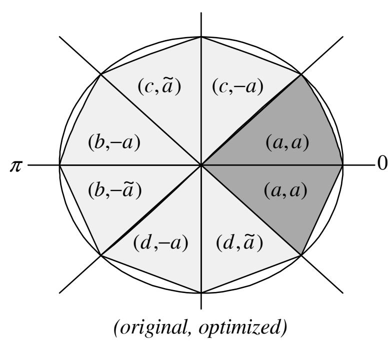

Fig. 4. The optimized coefficient set. $-a$ and $\tilde{a}$ indicate sign reverse and index reverse respectively.

with a smaller set. For example, if $\frac{N}{8} < n < \frac{N}{4}$, $P_d[n]$ and $Q_d[n]$ can be represented by $P_a[n]$ and $Q_a[n]$ as follows:

$$P_d[\frac{N}{4}-n] = \frac{\sin(-\frac{2\pi}{N}(\frac{N}{4}-n))+1}{\cos(-\frac{2\pi}{N}(\frac{N}{4}-n))} = P_a[n] \qquad (11)$$

$$Q_d[\frac{N}{4}-n] = -\cos(-\frac{2\pi}{N}(\frac{N}{4}-n)) = Q_a[n], \qquad (12)$$

Therefore, the polar plane can be partitioned into 8 sub-regions. Fig. 4 shows the mapping diagram of each decomposition in different sub-regions. All the other decompositions can be represented by Type (a). For instance, the non-trivial twiddle factors used in the first multiplier stage of 128-point radix-4 DIF FFT is mapped into Type (a) as shown in table II. Combining the idea of coefficient selection from section III.A, the criteria of choosing lifting coefficient must fulfill two conditions:

1) The dynamic range of selected coefficients should be smaller to keep lower MSE.
2) The selected coefficient set must be able to cover all other sub-regions.

Although the selection of coefficient set is not unique, it is convenient to choose the first sixteen pairs of Pa and Qa as the optimized coefficient set. The theoretical MSE performance comparison of different schemes is shown in Fig. 5. It is obvious that the LS-based complex multiplier with optimized coefficient set has the smallest MSE compared to the other methods.

Table II. The mapping of non-trivial multiplications into the optimized set.

Original Index	Original Type	Mapped Index	Mapped Type
1	(a)	1	(a)
2	(a)	2	(a)
17	(d)	15	(a)
18	(d)	14	(a)
49	(b)	15	(a)
50	(b)	14	(a)
97	(c)	1	(a)
98	(c)	2	(a)
...	...	...	...

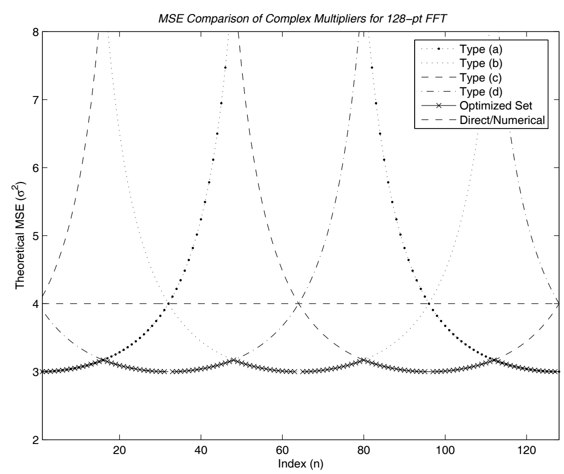

Fig. 5. MSE comparison chart of different complex multiplier implementations.

IV. SIMULATION RESULTS

IV-A. Quantitive Simulation

The theoretical performance evaluation of each FFT implementation has been given in previous works. James [9] derived the fixed-point MSE analysis of quantization loss for mixed-radix FFT algorithms with conventional complex multipliers. Perlow and Denk [10] proposed an error propagation model to estimate the performance of finite wordlength FFT architecture. Park and Cho [11] also used a propagation model to derive the error analysis for CORDIC implementations. The comparison of a vector rotation example has been addressed in table V of [12], which shows that the direct implementation outperforms CORDIC algorithm. In this paper, all four kinds of FFT implementations are simulated with fixed-point operations. A mixed-radix 128-point IFFT/FFT with different decomposition schemes is implemented to quantitively compare their SQNR performance. FxpFFT, NumFFT and CORFFT stand

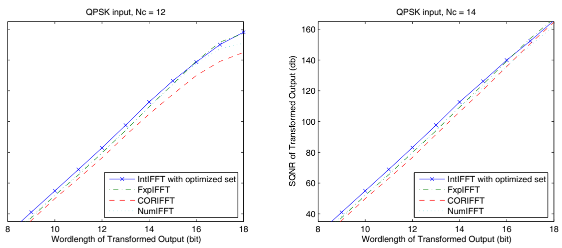

Fig. 6. SQNR comparisons of IFFT output with QPSK signal input.

Fig. 7. Fixed-point simulation with AWGN channel.

for direct implementation, numerical decomposition and CORDIC algorithm respectively. The experiment setup is as follows. First, fixed-point IFFT is performed with random QPSK signals. 1,000 trials are made for each IFFT. The wordlength of coefficients and the number of iterations used in the CORDIC algorithm is set to 12 and 14. The wordlength of internal variables is swept from 8 bits to 18 bits. As shown in Fig. 6, SQNR of IntFFT with optimized coefficient set has better SQNR performance than others.

IV-B. AWGN Environment

The performance comparison is also experimented with AWGN channel. As shown in Fig. 7, floating-point IFFT with QPSK signals is performed as golden reference. The transformed output is transmitted over an AWGN channel and then fed into different fixed-point FFT implementations. As seen from the simulation results of Fig. 8, IntFFT can yield comparative performance even when noisy channel is present.

V. CONCLUSIONS

In this work, the IntFFT with optimized coefficient set is proposed and quantitively compared to other popular FFT implementations. The advantage of using the selected coefficient set are twofold: First, it utilizes the periodicity of lifting coefficients so as to reduce the size of coefficient ROM since only $\frac{N}{8}$ coefficients are enough to represent all twiddle factors. Second, it also leads to better accuracy because the MSE with optimized coefficient set is lower. Based on the quantitive simulation results of Fig. 6, IntFFT with optimized coefficient set has better SQNR performance compared to other implementations. Besides, the simulation results show that the SQNR of CORDIC-based FFT is lower than the others when the number of iterations is set to the same as the wordlength of coefficients.

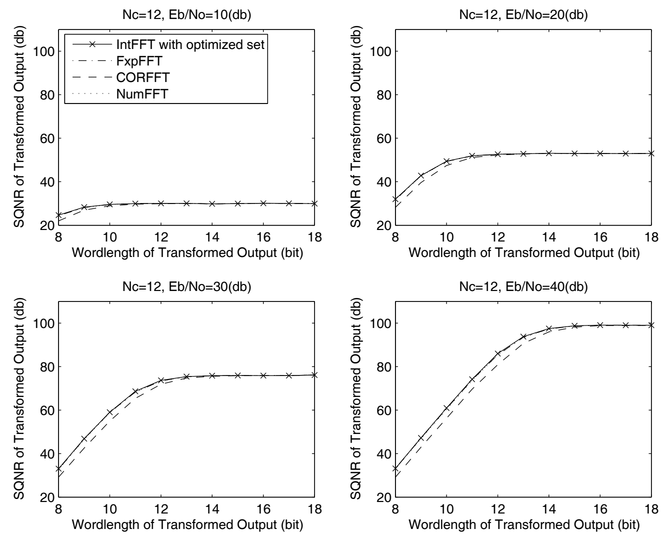

Fig. 8. SQNR comparisons with AWGN channel, Nc = 12.

VI. REFERENCES

[1] K. K. Parhi, *VLSI Digital Signal Processing Systems - Design and Implementation*. Wiley Inter-Science, 1999.

[2] S. Oraintara, Y. Chen, and T. Nguyen, "Integer fast fourier transform," *Signal Processing, IEEE Transactions on*, vol. 50, no. 3, pp. 607–618, 2002.

[3] I. Daubenchies and W. Sweldens, "Factorizing wavelet transforms into lifting steps," Bell Labs., Lucent Technol., Red Bank, NJ, Tech. Rep., 1996.

[4] J. W. Cooley and J. Turkey, "An algorithm for the machine calculation of complex fourier series," *Math. Comput.*, vol. 19, pp. 297–301, apr 1965.

[5] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-Time Signal Processing*, 2nd ed. Pretice Hall, 1998.

[6] S. He and M. Torkelson, "Designing pipeline FFT processor for OFDM (de)modulation," in *Signals, Systems, and Electronics, 1998. ISSSE 98. 1998 URSI International Symposium on*, 1998, pp. 257–262.

[7] H. Sorensen, M. Heideman, and C. Burrus, "On computing the split-radix FFT," *Acoustics, Speech, and Signal Processing, IEEE Transactions on*, vol. 34, no. 1, pp. 152–156, 1986.

[8] W.-H. Chang and T. Nguyen, "Architecture and performance analysis of lossless fft in ofdm systems," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 3, 2006, pp. III–1024–III–1027.

[9] D. James, "Quantization errors in the fast fourier transform," *Acoustics, Speech, and Signal Processing, IEEE Transactions on*, vol. 23, no. 3, pp. 277–283, 1975.

[10] R. Perlow and T. Denk, "Finite wordlength design for VLSI FFT processors," in *Signals, Systems and Computers, 2001. Conference Record of the Thirty-Fifth Asilomar Conference on*, vol. 2, 2001, pp. 1227–1231 vol.2.

[11] S. Y. Park and N. I. Cho, "Fixed-point error analysis of CORDIC processor based on the variance propagation formula," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 51, no. 3, pp. 573–584, 2004.

[12] C.-H. Lin and A.-Y. Wu, "Mixed-scaling-rotation CORDIC (MSR-CORDIC) algorithm and architecture for high-performance vector rotational DSP applications," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 52, no. 11, pp. 2385–2396, 2005.