

A Novel Lattice Architecture for High Speed Discrete MultiTone (DMT) Modulation

Yasser Ismail

Department of Computer Engineering, College of Information Technology, University of Bahrain, Bahrain

e-mail: Yasserali1977@gmail.com

Received 9 Feb. 2013, Revised 23 Mar. 2013, Accepted 15 Apr. 2013

Abstract: The Discrete MultiTone (DMT) modulation system is an attractive method for a high speed data transmission over a non-flat channel with possible colored noise. The IFFT is used in its transmitter in order to divide the available frequency spectrum into large independent narrow band sub-channels. A high speed and an efficient lattice architecture is proposed to implement the IFFT. This will speed up the DMT transmitter, so that, it can be easily used in any real time data transmission application. The proposed IFFT lattice architecture achieves approximately 58%, 39%, and 50% savings in gate count, power consumption, and area compared to the state of the art IFFT architectures that use conventional lattice structure. The proposed IFFT lattice architecture can perform high speed data transmission at 118 MHz. This allows the DMT transmitter to be easily integrated in any real time data transmission system.

Keywords: DMT; lattice structure; CORDIC; high speed data transmission.

I. INTRODUCTION

High speed data transmission is considered as the main concern of the internet access nowadays. Many fast modulation techniques [1-7] have been proposed for fast data transmission. Since it incorporates many Digital Signal Processing (DSP) algorithms such as multi-dimensional tone encoding and frequency domain equalization, Discrete MultiTone (DMT) scheme achieves the highest transmission speed amongst the previous fast techniques [8]. Consequently, DMT has been chosen to be the physical layer of the ADSL standardization committee [9-12]. The theoretical advantages of MultiTone modulation were demonstrated in [13, 14].

DMT is a form of multicarrier modulation that is used to transmit data over communication systems. It divides the available frequency spectrum into large independent narrow band sub-channels. Each sub-channel in the DMT system carries a predetermined amount of data in parallel. The throughput of the whole DMT system is the sum of all data in each parallel sub-channel. The transmitters of the DMT system use the Inverse Fast Fourier Transform

(IFFT) for modulation, while the DMT receivers use the FFT for demodulation process [5-7].

The main bottleneck of using the DMT scheme in high speed data transmission is its high computational complexity due to the large block size of the IFFT/FFT [7]. Lattice structure is one of the best structures that is used for more reductions in the hardware implementation and computations of the IFFT architecture. This is due to the duality property which saves more in hardware complexity [8].

Given that the FFT/IFFT coefficients can be represented in both the Discrete Cosine Transform (DCT) and the Discrete Sine Transform (DST) [15], a time recursive lattice structure is a good choice for generating the FFT/IFFT due to the following reasons:

1. Using a lattice structure can produce dual transforms (e.g., “DCT and Discrete Sine Cosine Transform (DSCT)” or “DST and Discrete Cosine Sine Transform (DCST)”). It means more reductions in the hardware implementation and computations.

2. The resulting architectures for generating (DCT and DSCT) or (DST and DCST) are identical. So, simplicity of the encoder design is achieved.
3. There is no global communications are required in the recursive lattice architecture.
4. The number of multipliers in the lattice structure is a linear function of N, where N is the number of input samples. Consequently, it requires less number of multipliers when N is large compared to most other fast discrete transform algorithms [16].

Multiplications by either sine or cosine are the main bottleneck of using such lattice architecture. One of the main methods to tackle this problem is to use the Coordinate Rotation Digital Computer (CORDIC) [17]. Bit-parallel iterative architecture, bit-serial iterative architecture, and unrolled CORDIC architecture are the main architectures for implementing the CORDIC algorithm [17].

Using the bit-parallel iterative CORDIC architecture is a good choice if we are looking for an efficient area and easy implementation. The main bottleneck of such architecture is that it suffers from slow performance that makes it not suitable for the real time data transmission which is the main target of the proposed work. The slow performance is due to some reasons. The first reason is the absence of the pipelining procedure. Second, it uses word wide data paths that require larger adders. Last, it uses variable shifters that do not map well in hardware because they have high fan-in.

The unrolled CORDIC architecture uses multiple stages of the bit-parallel iterative architecture with parallelism procedure. This results in an improvement in the throughput of the design. This architecture also reduces the hardware complexity compared to the bit-parallel iterative architecture. First, shifters are now fixed which means there is no need for expensive barrel shifters, simple wiring will do the job. Second, elementary angles are now constants for a specific iteration I, consequently no need for expensive ROM. The main bottleneck of such design is that it still uses a parallel fashion that needs word wide data paths that require larger adders.

Using the bit-serial iterative architecture reduces both area and power consumption since only one bit adder/subtractor is needed [17]. Additionally, higher clock rate is obtained in case of using a serial design compared to using a parallel design. Since only one iteration stage is

used in the bit serial design to get the final output, it is not suitable for real-time applications.

In this paper, the modulation speed of the DMT system is increased by reducing the hardware complexity of the IFFT. The recursive lattice structure in [18, 19] is modified in this paper to speed up the modulation process. In this work, a mixed flavor of parallel and serial operations is used to generate a fast CORDIC architecture that could be used in the architecture of the IFFT. The proposed IFFT architecture has lower power consumption and area and has higher speed than the existing IFFT lattice architectures. This gives the proposed IFFT architecture the advantage to be used for DMT-based ADSL system.

The paper is organized as follows. Section II discusses the basic principle of the DMT modulation. Section III discusses the problem formulation. Section IV discusses CORDIC principle. The proposed Bit Serial CORDIC Architecture is discussed in section V. The Lattice-CORDIC Design is illustrated in section VI. Section VII discusses the Overall VLSI architecture of the IFFT. Implementation and discussion is drawn in section VIII. Finally, in section IX conclusions are drawn.

II. BASIC PRINCIPLE OF THE DMT MODULATION

Discrete MultiTone (DMT) modulation is one of the most popular multicarrier modulation that is used to carry copper based Digital Subscriber Lines (DSLs) [20].

The idea of the DMT modulation system is to split the available frequency spectrum into a large number of independent narrow band sub-channels [21]. The throughput of the whole system is the summation of the data transmitted via all the sub-channels.

The transmitter of the DMT modulation consists mainly of M-array quadrature amplitude modulation (M_QAM), and IFFT [22]. The FFT is used in the receiver for the demodulation process. Using an M_QAM, the carrier of each sub-channel is mapped into complex values C_n , where $n=1, 2, \dots, N-1$ is the number of the sub-channels. The $2N$ -points IFFT is used to modulate C_n into different carrier frequencies that are mutually orthogonal. Considering that $C_0=C_N=0$ and assuming that the symmetry property $C_{2N-n} = C_n^*$ is achieved, the resulting multicarrier DMT time-domain sequence δ_k can be expressed as [22]:

$$\delta_k = \frac{1}{\sqrt{2N}} \sum_{n=0}^{2N-1} C_n \exp\left(j2\pi k \frac{n}{2N}\right), \quad k = 0, 1, \dots, 2N - 1 \quad (1)$$

Note that δ_k is a real valued signal of $2N$ sample points.

The demodulation of the DMT sequence is achieved by using $2N$ -points FFT at the receiver side. The result of the FFT will be as follows [22]:

$$\hat{C}_n = \sum_{k=0}^{2N-1} \hat{\delta}_k \exp\left(j2\pi k \frac{n}{2N}\right), n = 0, 1, \dots, 2N - 1 \quad (2)$$

It is worth mentioning that the IFFT and the FFT are characterized by their symmetry properties. Consequently, their operations can be optimized and reduced to approximately half number of computations for the DMT modulation [23].

Since the most exhaustive part of the DMT modulation system is the IFFT, our target in this paper is to design an efficient high speed IFFT. Consequently, increasing the throughput of the whole transmitter. This will help to use the DMT in real time data transmission, efficiently.

III. PROBLEM FORMULATION

The target of the proposed work in this paper is to speed up the DMT modulation process. This is achieved by decreasing the hardware complexity of the IFFT used in the DMT transmitter. As mentioned before in the introduction part, using lattice structure is a good choice to reduce the hardware complexity of the IFFT generator.

The IFFT of a sequence $X(k); k = 0, 1, \dots, N-1$ and $N=256$; can be defined as [15]:

$$x(n) = \frac{1}{N} \left[\sum_{k=0}^{N-1} X_r(k) \cos \frac{2\pi nk}{2N} - \sum_{k=0}^{N-1} X_i(k) \sin \frac{2\pi nk}{2N} \right] \quad (3)$$

$$= \frac{1}{N} [DCT_r(n) - DST_i(n)]$$

Where $X_r(k)$ and $X_i(k)$ denote the real and the imaginary parts of the input sequence $X(k)$ and $n=0,1, \dots, 2N-1$. DCT_r and DST_i are the Discrete Cosine and Sine transforms for the real and the imaginary parts of the input sequence $X(k)$, respectively.

Using the symmetric and anti-symmetric properties of both the DCT_r and the DST_i [15], Equation (3) can be reformulated and implemented using the simplified lattice structure shown in Figure 1. The preprocessing module in Figure 1 can be implemented using a separate DSP for further reduction in the hardware complexity. The output X_n of the pre-processing module is shown in Equation 4 [15]. The main bottleneck of using the lattice architecture in figure 1 is the high hardware implementation complexity. This results from multiplication by cosine and sine functions. This problem will be tackled in this paper as will be seen shortly in the following sections.

$$X_n = [X_r(k) + (-1)^n X_r(N-k)] + [X_i(k) + (-1)^n X_i(N-k)] \quad (4)$$

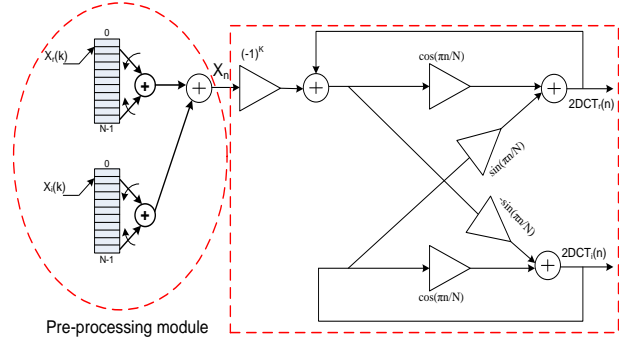


Figure 1. The lattice structure for the IFFT.

IV. COORDINATE ROTATION DIGITAL COMPUTER (CORDIC)

The Coordinate Rotation Digital Computer (CORDIC) is an iterative method for calculating several functions such as trigonometric ones, fixed/floating point, multiply, divide, log, exponent and square root using simple shift and add operations [24]. CORDIC is particularly important due to its simplicity, recursive nature, low hardware complexity and applicability to a wide range of functions.

The basic idea of CORDIC is to decompose the desired rotation angle θ into the weighted sum of a set of predefined elementary rotation angles a_i such that the rotation through each of them can be accomplished with simple shift and add operations [24]. The desired angle can be defined as:

$$\theta = \sum_{i=0}^{n-1} \pm \theta_i = \sum_{i=0}^{n-1} d_i a_i \quad (5)$$

Where d_i represents the direction of the rotation angle in either positive or negative direction.

Consider a vector in Cartesian plane $v_1 = x + jy$. Rotating v_1 by an angle θ is obtained by multiplying v_1 by another vector $v_2 = \cos\theta + jsin\theta$. The product becomes:

$$v_1 v_2 = (x \cos\theta - y \sin\theta) + j(x \sin\theta + y \cos\theta) = X' + jY' \quad (6)$$

Where:

$$X' = x \cos\theta - y \sin\theta = \cos\theta (x - y \tan\theta) \quad (7)$$

$$Y' = x \sin\theta + y \cos\theta = \cos\theta (y + x \tan\theta) \quad (8)$$

If the elementary rotation angles are chosen such that $\tan\theta = \pm 2^{-i}, i \in 1 \rightarrow N_1$, where N_1 is the number of the required iterations to get the value of equation (5),

then the multiplication inside the parenthesis is reduced to a simple shift operation [24]. Equation 7 and 8 can be reformulated as follows:

$$X_{i+1} = K_i[X_i - Y_i d_i 2^i] \quad (9)$$

$$Y_{i+1} = K_i[Y_i + X_i d_i 2^i] \quad (10)$$

Where: $K_i = 1/\sqrt{1+2^{-2i}}$ and $d_i = \pm 1$

Multiplication by K_i is done either at the beginning or at the end of the iteration process. This multiplication is reduced to simple shift and add operations. The value of K_i is 0.6073 as the number of iterations goes to infinity. The angle of a composite rotation may be defined by the direction of each of its constituting elementary rotation angles as follows [24]:

$$Z_{i+1} = Z_i - d_i \tan^{-1} 2^{-i} \quad (11)$$

A complete CORDIC procedure is summarized in Figure 2.

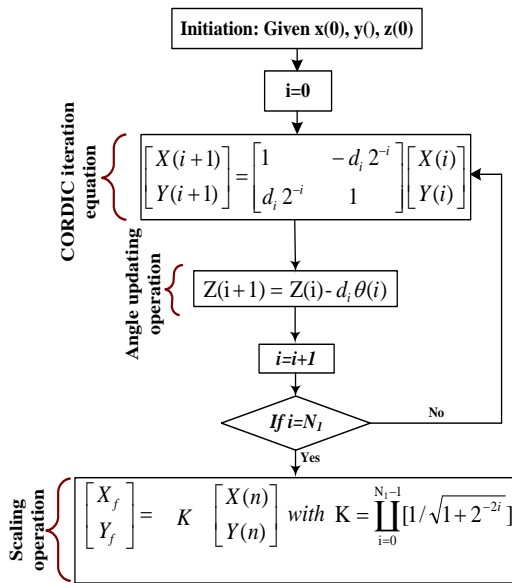


Figure 2. Flow chart of the CORDIC procedure.

V. BIT SERIAL CORDIC ARCHITECTURE

The main idea of using the CORDIC structure is to implement the multiplication operations either by sine or cosine that appear in figure 1. The aim of the proposed CORDIC architecture is to increase the speed with an acceptable degradation in both area and power consumption. This is achieved by serializing both the shifters and the adders as seen in figure 3.

The idea of the proposed CORDIC structure is to have N_1 unrolled iterations. At each iteration i (where $i \in 0 \rightarrow N_1 - 1$), three serial adders/subtractors, three shift registers for X_i , Y_i , and Z_i are used as seen in

figure 3. A control unit is responsible for managing the whole operations. It triggers the iteration stage number $i+1$ to start operation after the iteration stage number i finishes its operations.

The whole operation of the proposed unrolled bit serial online CORDIC processor is summarized as follows. Both X_0 and Y_0 are represented as 22 bits (10 bits for integer part and 12 bits for fractional part). The angle Z_0 is represented using 12 bits (10 bits for integer part and 2 bits for fractional part). Assuming all the registers are reset to zero, the operation starts by enabling the start signal. Once the start signal goes high, the control unit CU_0 initiates X , Y , and Z registers to start loading, serially, with the initial values X_0 , Y_0 and Z_0 . This occurs when the XYZ_load signal goes high. Since 12 bits are only read from Z_0 , a multiplexer is needed at the front of Z_i Shift Register (SR) in the first iteration to continue filling the registers with 10 zeros (chosen from second multiplexer's input). The select control signal is 0 for the first 12 serial clock cycles (ser_clk) and then goes high for the rest of the 10 ser_clk cycles. Once the 22 bits are read, the MSB of Z_0 determines the type of operation for each input (addition or subtraction). The Z_{en} control signal will be high when the last MSB of Z_0 is available. The angle_select control signal will be high after 22 ser_clk cycles to start the addition/subtraction operation. When stage 1 has its first output, the next_stage control signal will be high to trigger stage 2 to start its operations. The inputs to the Z adder/subtractor are the output of the previous Z shift register and another shift register that is loaded with the constant for that specific iteration. The outputs of the adder/subtractor at iteration number i will be loaded in the registers of the next iteration $i+1$. This operation will continue until the last iteration stage. The final output X_n and Y_n should be scaled by multiplying both of them by 0.60727. This multiplication is converted to thirty four shift operations and six addition operations as seen in figure 3. It is worth mentioning that the shift operations in the scaling part are constant and may be done using simple wiring. In addition, only nine stages are used to generate the sine and the cosine. The selection of such number of stages is based on the required acceptable accuracy of the fractional part of the final output.

VI. LATTICE-CORDIC DESIGN

In this section, we will illustrate how the proposed CORDIC architecture can be integrated into the lattice structure of figure 1 to convert the multiplication operations

by sine and cosine into shift and addition/subtraction operations.

The proposed Lattice-CORDIC architecture consists of three main parts as seen in figure 4. The control unit is used to provide with all required control signals. The CORDIC processor is implemented using the proposed serial fashion in section 4. Angle shift register (Angle_SR) is used for storing the desired angles required for the proposed serial CORDIC. The desired angles Z_0 is defined as $\frac{k\pi}{2N}$, where $k = 0, 1, 2, \dots, N$ and N is the number of input values. Table I illustrates an example for the used desired angle in case of $k \in 0 \rightarrow 7$. Note that some of the angles for the proposed CORDIC need to be modified. Since the algorithm converges to a solution if the angles are in a certain range ($[-\pi/2, \pi/2]$), for angles outside this range, a preprocessing step is required to shift the original angle to meet this condition as follows:

$$\begin{aligned} z'_0 &= z_0 \pm 180^\circ \\ x'_0 &= \overline{x_0} \\ y'_0 &= \overline{y_0} \end{aligned} \quad (10)$$

Table I: The angles used by CORDIC design for $k \in 0 \rightarrow 7$.

K	0	1	2	3	4	5	6	7
Angle 1	0	11.25	22.5	33.75	45	56.25	67.5	78.75
Angle 2	0	22.5	45	67.5	90	112.5	135	157.5
Modified Angle 2	0	22.5	45	67.5	90	-67.5	-45	-22.5

After N_1 parallel clock cycles, where N_1 is the number of stages/iterations in the proposed serial CORDIC, the two outputs of the CORDIC processor are ready. The $2DCT_r$ is then feedback to the inputs. DCT_r is serially added to the input X_n and the DST_i is directly input to the other terminal of the CORDIC processor. It is worth mentioning that one parallel clock cycle equals to 23 serial clock cycles.

The main advantage of the proposed serial CORDIC architecture is the simplicity of its hardware due two main reasons. First, shifters are now fixed which means there is no need for expensive barrel shifters, simple wiring will do the job. Second, the elementary angles ($\tan^{-1}2^{-i}, i \in 1 \rightarrow 9$) are now constants for a specific iteration which means there is no need for ROM. Consequently, the proposed Lattice-CORDIC architecture enables a higher clock rate compared to the state of the art lattice architecture [5] and [9]. The main advantage of using such serial architecture is that it uses small registers and adders. This will reduce both the area and the power consumption compared to the architecture in [5] and [9]. Another advantage of the proposed Lattice-CORDIC

architecture is its pipelined fashion which achieves 100% utilization as well as a very high throughput rate.

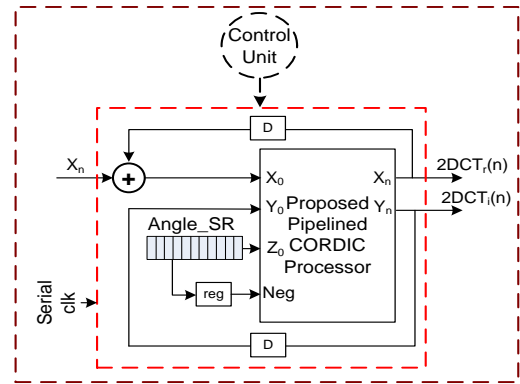


Figure 4. Lattice-CORDIC structures

VII. IFFT ARCHITECTURE

The architecture of the IFFT using the proposed lattice CORDIC consists of three main parts as seen in Figure 5. First, the module array including $N-1$ lattice CORDIC shown in Figure 4. In addition, the module M_0 which is consider as the special case for generating $x(0)$ and $x(N)$. Second, the expanding part which is used in expanding the output of the module array into $2N-1$ sequence. Finally, the shifting part which is used to scale the output by shifting one bit to the left.

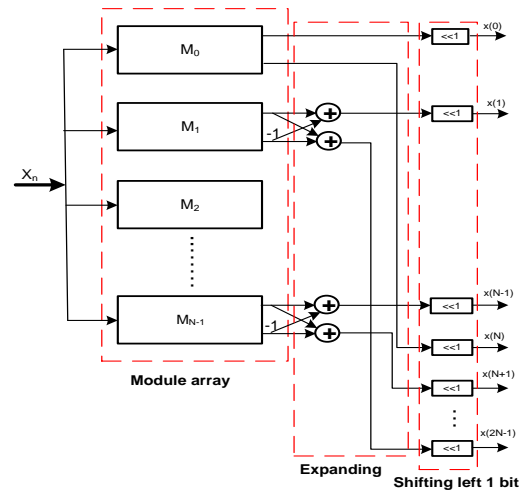


Figure 5. The whole IFFT lattice architecture.

The operation starts when the sequence X_n (coming from the pre-processing DSP) is available. $X_n; n=0, 1, \dots, N-1$; are fed into the module array which takes N_1+1 clock cycles. Then the outputs of the module array are expanded to form $2N-1$ values as seen in Figure 4. The scaling operation is achieved by shifting left by 1 bit. Since all operations are serial, the processing time, area, and power consumption are low compared to the state of the art lattice architectures.

VIII. IMPLEMENTATION AND DISCUSSION

The proposed IFFT architecture in Figure 5 was implemented using functional VHDL. Then this code was verified using the ModelSim tool. The standard cell ASIC design flow approach using OSU (Oklahoma State University) standard cells library was followed for the hardware implementation. The proposed architecture was synthesized using BGX_shell tool in TSMC 0.18 μm technology. The layout was done using Cadence SOC Encounter tool. The value of N is chosen to be 256. The lattice architecture of the proposed IFFT is compared to the Lattice-CORDIC architecture in [18] and the unrolled parallel Lattice-CORDIC Architecture in [17].

The gate count of the proposed IFFT is shown in Table II. This count includes the gates used in lattice-CORDIC module array, the expanding bank, the shifter bank, and the Control Unit. It is noted from the table that using the unrolled parallel Lattice-CORDIC in [17] improves the throughput of the IFFT generator compared to the one in [18]. However, the IFFT generator using the Lattice-CORDIC in [17] degrades the power consumption, area, and the gate count. It means, using the Lattice-CORDIC in [18] is nominated for applications that give the priority to reducing area and power consumption rather than increased encoding speed. Whereas the IFFT generator which use the unrolled parallel CORDIC in [17] is nominated for applications which prioritize high speed processing over reduced area.

Table II: comparison table for the proposed IFFT using serial CORDIC and the state of the art architectures.

	IFFT using CORDIC of [18]	IFFT using parallel CORDIC of [17]	Proposed IFFT
N	256	256	256
Process	0.18 μm	0.18 μm	0.18 μm
Gate count	3.32 M	16.28 M	1.38 M
Avg. Power (mw)	72.32	241.92	23.02
Area (μm^2)	6.25	29.16	2.43
Maximum frequency (MHz)	127.56	129.53	158.86
Throughputs	$\frac{1}{40(N \times N_1) + 25}$	$\frac{1}{N \times N_1 + 3}$	$\frac{1}{N \times N_1 + 3}$

The proposed IFFT achieves same throughput as in case of using the fast Lattice-CORDIC in [17], however, it maintains an efficient area, power consumption, and gate count which are close to the simple design in [18]. This gives the proposed IFFT generator a superior performance if it is used in applications that target high speed and maintaining low power and area consumption. Finally the

chip Layouts of the proposed IFFT generators and using the Lattice-CORDIC in [17] and [18] is shown in Figure 6.

IX. CONCLUSION

The IFFT is the main time consuming part in the DMT transmitter. A high speed lattice CORDIC architecture for implementing the IFFT generator is proposed to speed up DMT transmitter. The proposed architecture achieves approximately 58%, 39%, and 50% savings in gate count, power consumption, and area compared to the conventional state of the art pipelined IFFT generators that uses the lattice CORDIC architectures.

ACKNOWLEDGMENT

The author acknowledges the support of the center for advanced computer studies – Louisiana University - U.S.A for their help with the tools and support to finish this work.

REFERENCES

- [1] B. Daneshrad and H. Samueli, "A 1.6 Mbps digital-QAM system for DSL transmission," IEEE Journal on Selected Areas in Communications, vol. 13, pp. 1600-1610, 1995.
- [2] G. H. Im, D. B. Harman, G. Huang, A. V. Mandzik, M. H. Nguyen, and J. J. Werner, "51.84 Mb/s 16-CAP ATM LAN standard," IEEE Journal on Selected Areas in Communications, vol. 13, pp. 620-632, 1995.
- [3] K. D. Langer, J. Vucic, C. Kottke, L. F. del Rosal, S. Nerreter, and J. Walewski, "Advances and prospects in high-speed information broadcast using phosphorescent white-light LEDs," in 11th International Conference on Transparent Optical Networks, ICTON '09, 2009, pp. 1-6, June 28 2009-July 2 2009.
- [4] C. Milion, T. Duong, N. Genay, E. Grard, V. Rodrigues, B. Charbonnier, J. Le Masson, M. Ouzzif, P. Chanclou, and A. Gharba, "High bit rate transmission for NG-PON by direct modulation of DFB laser using discrete multi-tone," in 35th European Conference on Optical Communication, ECOC '09, 2009, pp. 1-2, 20-24 Sept. 2009.
- [5] Y. Na, M. Yi, and R. Tafazolli, "Multi-tone transmissions over two-user cognitive radio channel with weak interference," in IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC 2008, pp. 1-5, 15-18 Sept. 2008.
- [6] K. Sistanizadeh, P. Chow, and J. Cioffi, "Multi-tone transmission for asymmetric digital subscriber lines (ADSL)," in IEEE International Conference on Communications, ICC 93. Geneva 1993, pp. 756-760, 23-26 May 1993
- [7] S. Xuejie and S. Dasgupta, "Optimumisi-free DMT systems withconcave SNR tobit rate relations: when does othonornmality suffice?," in IEEE 8th Workshop on Signal Processing Advances in Wireless Communications, SPAWC 2007, 2007, pp. 1-4, 17-20 June 2007.

- [8] W. An-Yeu and C. Tsun-Shan, "Cost-efficient parallel lattice VLSI architecture for the IFFT/FFT in DMT transceiver technology," in IEEE International Conference on Acoustics, Speech and Signal Processing, 1998, pp. 3517-3520, 12-15 May 1998.
- [9] P. S. Chow, J. C. Tu, and J. M. Cioffi, "Performance evaluation of a multichannel transceiver system for ADSL and VHDSL services," *Selected Areas in Communications, IEEE Journal on*, vol. 9, pp. 909-919, 1991.
- [10] A. N. Akansu, P. Duhamel, L. Xueming, and M. De Courville, "Orthogonal transmultiplexers in communication: a review," *Signal Processing, IEEE Transactions on*, vol. 46, pp. 979-995, 1998.
- [11] J. W. Lechleider, "High bit rate digital subscriber lines: a review of HDSL progress," *Selected Areas in Communications, IEEE Journal on*, vol. 9, pp. 769-784, 1991.
- [12] G. W. Wornell, "Emerging applications of multirate signal processing and wavelets in digital communications," *Proceedings of the IEEE*, vol. 84, pp. 586-603, 1996.
- [13] I. Kalet, "The multitone channel," *Communications, IEEE Transactions on*, vol. 37, pp. 119-124, 1989.
- [14] P. P. Vaidyanathan and Y.-P. Lin, "Discrete Multitone Modulation With PrincipalComponent Filter Banks," *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: FUNDAMENTAL THEORY AND APPLICATIONS*, vol. 49, No.10, OCTOBER 2002.
- [15] Y. Chi-Li and W. An-Yeu, "An improved time-recursive lattice structure for low-latency IFFT architecture in DMT transmitter," in *The 2001 IEEE International Symposium on Circuits and Systems, ISCAS 2001*, 2001, pp. 250-253 vol. 4, 6-9 May 2001.
- [16] K. J. R. Liu and C. T. Chiu, "Unified parallel lattice structures for time-recursive discrete cosine/sine/Hartley transforms," *Signal Processing, IEEE Transactions on*, vol. 41, pp. 1357-1377, 1993.
- [17] R. Andraka, "A survey of CORDIC algorithms for FPGA based computers," *Proceedings of the ACM/SIGDA sixth international symposium on Field programmable gate arrays*, pp. 191-200, 1998.
- [18] J. Chen and K. J. R. Liu, "A complete pipelined parallel CORDIC architecture for motion estimation," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 45, pp. 653-660, 1998.
- [19] C. T. Chiu and K. J. R. Liu, "Real-time parallel and fully pipelined two-dimensional DCT lattice structures with application to HDTV systems," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 2, pp. 25-37, 1992.
- [20] A. M. Khalid, G. Cossu, R. Corsini, P. Choudhury, and E. Ciaramella, "1-Gb/s Transmission Over a Phosphorescent White LED by Using Rate-Adaptive Discrete Multitone Modulation," *Photonics Journal, IEEE*, vol. 4, pp. 1465-1473, 2012.
- [21] J. Armstrong, "OFDM for Optical Communications," *Lightwave Technology, Journal of*, vol. 27, pp. 189-204, 2009.
- [22] S. C. J. Lee, F. Breyer, S. Randel, R. Gaudino, G. Bosco, A. Bluschke, M. Matthews, P. Rietzsch, R. Steglich, H. van den Boom, and A. Koonen, "Discrete Multitone Modulation for Maximizing Transmission Rate in Step-Index Plastic Optical Fibers," *Lightwave Technology, Journal of*, vol. 27, pp. 1503-1513, 2009.
- [23] H. Sorensen, D. Jones, M. Heideman, and C. Burrus, "Real-valued fast Fourier transform algorithms," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 35, pp. 849-863, 1987.
- [24] Y. H. Hu, "CORDIC-based VLSI architectures for digital signal processing," *IEEE Signal Processing Magazine*, vol. 9, pp. 16-35, 1992.

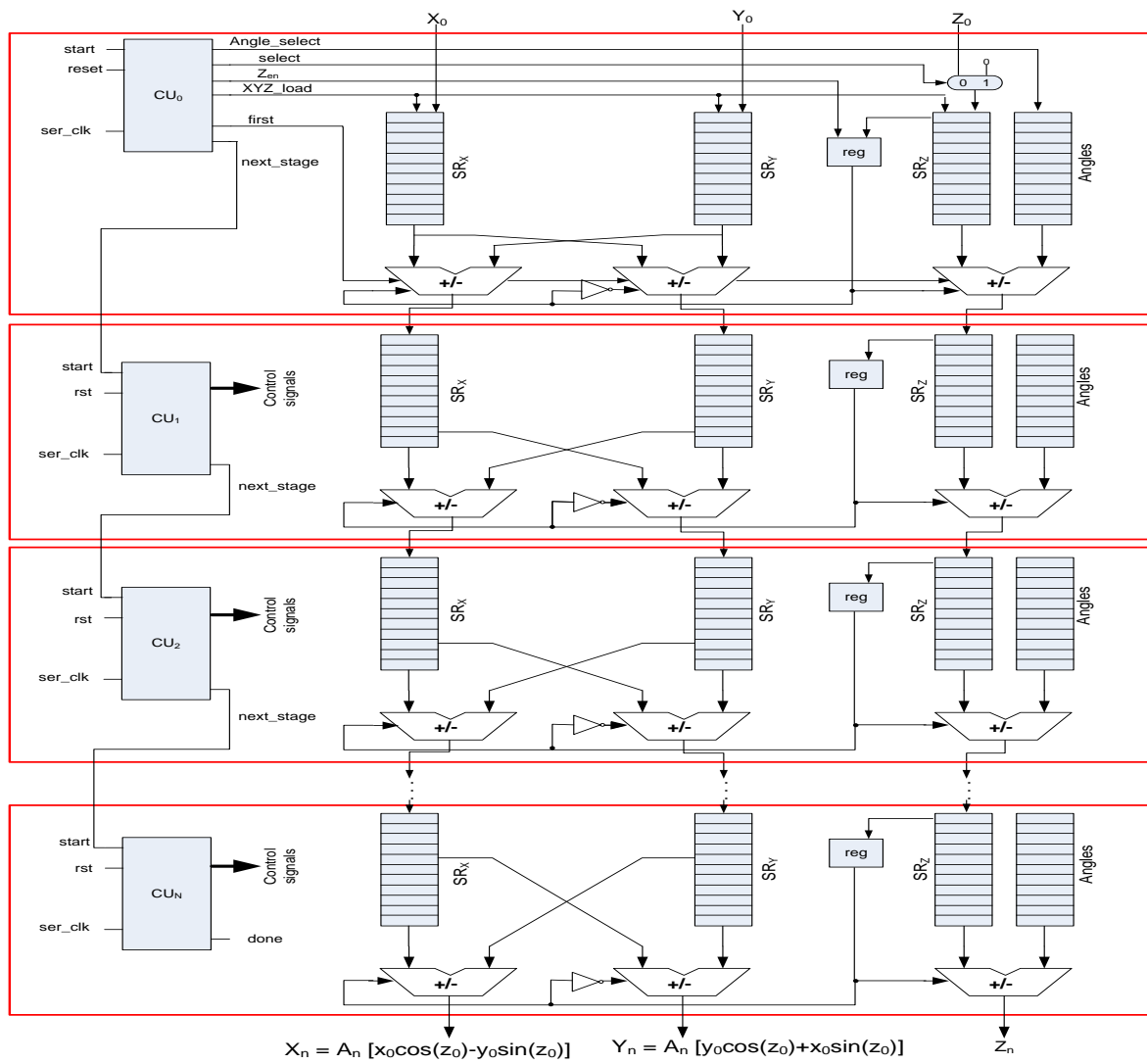


Figure 3. the proposed serial CORDIC architecture.

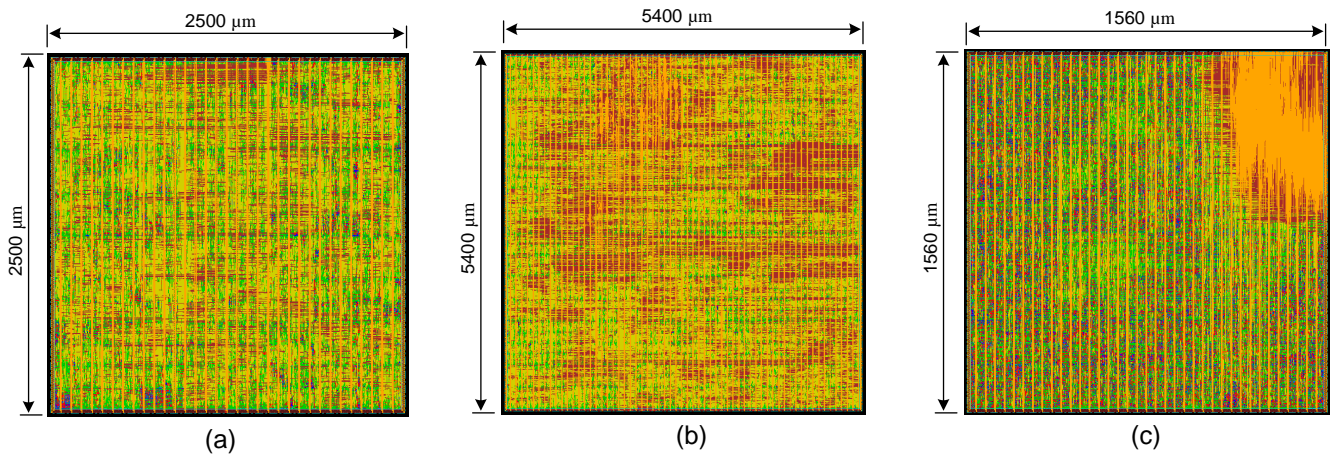


Figure 6. Layout of the IFFT architecture using (a) The CORDIC architecture in [18]. (b) The CORDIC architecture in [17]. (c) The proposed serial CORDIC