A. Mandal and R. Mishra / IU-JEEE Vol. 13(1), (2013), 1581-1587

Efficient Use of Application Specific CORDIC for Digital Demodulation in I/Q Receiver

AMRITAKAR MANDAL[1], RAJESH MISHRA [2]

[1,2]School of Infromation and Communication Technology, Gautam Buddha University, India
amritkar2k@gmail.com, rmishra@gbu.ac.in

Abstract: *Digital modalities of sine and cosine waves are gaining enormous attention in the field of vector rotated Digital Signal Processing (DSP) applications. COordinate Rotation DIgital Computer (CORDIC) algorithm has become very important and widely researched topic due to its simplicity to cater almost perfect digital sine and cosine waveforms during modulation and demodulation processes in various digital designs. In DSP applications, the quantization errors generated in CORDIC may propagate through subsequent modules ending up with reduced SNR of the system as a whole. In this paper, we have presented the design of a pipelined CORDIC architecture for detection of amplitude- phase variations in a demodulator of FMCW radar. The angle approximation and rounding off error of CORDIC have been intensively studied for the determination of design parameters. An expression for overall quantization error is derived. The design of application specific CORDIC processor in the circular rotation mode gives a high system throughput due to its pipelined architecture by reducing latency in each individual pipelined stage. Saving area on FPGA is essential to the design of pipelined CORDIC and that can be achieved through the optimization in the number of micro rotations. Hardware synthesized result using Cadence design tools are presented.*
Keywords: *Digital Signal Processing (DSP), CORDIC, Pipelined Architecture, FMCW Radar, Quantization error*

1. Introduction

The COordinate Rotation Digital Computer (CORDIC) algorithm [1] is a hardware efficient iterative algorithm which allows a simple shift and adds operation to calculate hyperbolic, exponential, and logarithmic and trigonometric functions like sine, cosine, magnitude and phase with great precision for Digital Signal Processing applications [2]. The sine and cosine terms can be calculated using polynomial approximation or interpolation method using table look-up, but it has a huge drawback in implementation where large number of gates and ROM memory is required. The proposed architecture improves the performance of previous approaches with simpler design with reduced area and power consumption. The CORDIC architecture has been proposed is free from the internal ROM memory and sign-bit register (SBR) [3] which is usually used to store the control-bits and direction of information for the number of shifts corresponding micro-rotations and directions of micro-rotations respectively.

The proposed design can be utilized where high speed, low latency and a high throughput at the output stage is necessary. The design can find a great applicability in a real time application like digital demodulation in Software Defined Radio (SDR) [4] technology based FMCW radar. In this paper, CORDIC based digital phase-locked loop based demodulator has been designed and simulated output in MATLAB has been presented. In general, radar system uses coherent oscillator as a reference frequency for detection of Doppler phase shift. The amount of phase shift depends on the radial speed of the moving object. The problem with the analog demodulation is the non-linearity of Voltage Controlled Oscillator (VCO) which suffers to maintain spectral purity over the desired frequency range [5]. CORDIC can be a good choice for accurate and efficient digital demodulation.

For optimal design of system using the CORDIC processor, the analysis of various error sources is necessary . In DSP systems, signals are required to be quantized and represented in fixed wordlength [6]. In general, larger the dynamic range of the signals, more severe is the round-off noise. Therefore, exact computation of wordlength is necessary for designing an architecture for CORDIC. If wordlength is larger, the computational speed of CORDIC reduces significantly [7,8]. On the other hand, if we implement with smaller wordlength, the design will suffer from danger of overflow . In this paper, both the problems of overflow and quantization noise have been addressed adequately in section 4 for the design process.

The paper is structured as follows. Section 2, CORDIC algorithm for DSP application has been discussed. In section 3, a thorough description of design of pipelined architecture along with numerical error analysis and optimization is described. In section 4, basics of demodulation has been discussed. In section 5, design and

Received on: 15.10.2012
Accepted on: 17.04.2013

application of the proposed CORDIC for digital demodulation has been explained. At the last, in section 6, hardware synthesis result has been presented.

2. CORDIC Algorithm for DSP Application

The theory of CORDIC computation is to decompose the desired rotation angle into the weighted sum of a set of predefined elementary rotation angles through each of them can be accomplished with simple shift-add operation for a desired rotational angle θ ,it can be represented for M iterations of an input vector $(x,y)^T$ setting initial conditions $x_0 = x$, $y_0 = y$ and $z_0 = \theta$ as $z_f = \theta - \sum_{i=0}^{M-1} \delta_i \alpha_i$ (1)

If $z_f = 0$ holds, then $\theta = \sum_{i=0}^{M-1} \delta_i \alpha_i$ (2)

i.e. the total accumulated rotation angle is equal to θ. $\delta_i, 0 \le i \le M-1$, denote a sequence of ±1s that determine the direction of each elementary rotation.

Almost every signal used in DSP module is complex in nature. So all these signal will follow the Euler's theorem resulting one sided spectrum with direction of rotation (positive or negative frequency) and with known real (cosine) and imaginary (sine) components.

$$\cos \omega_c t + j \sin \omega_c t = e^{j\omega_c t} \quad (3)$$

Let a signal vector $\hat{v}$ with angle θ is passed through CORDIC processor. The outcome from CORDIC can be shown as follows.

$$\hat{v} = v e^{j\theta} = v.\exp(j(\sum_{i=0}^{M-1} \sigma_i.\alpha_i)) = v.(\prod_{i=0}^{M-1} e^{j\delta_i\alpha_i})$$

$$e^{j\delta_i\alpha_i} = \cos(\delta_i.\alpha_i) + j\sin(\delta_i.\alpha_i)$$
$$= \cos(\delta_i.\alpha_i).(1 + j\tan(\delta_i.\alpha_i))$$
$$= \cos(\delta_i.\alpha_i).(1 + j\delta_i.2^{-i})$$
$$= \cos(\alpha_i).(1 + j\delta_i.2^{-i}) \quad (4)$$

Therefore,

$$\hat{v} = v.(\prod_{i=0}^{M-1} \cos(\alpha_i).(1 + j\delta_i.2^{-i}))$$
$$= v.(\prod_{i=0}^{M-1} \cos(\alpha_i)).(\prod_{i=0}^{M-1} (1 + j\delta_i.2^{-i}))$$
$$\hat{v} = v.K_i.(\prod_{i=0}^{M-1} (1 + j\delta_i.2^{-i})) \quad (5)$$

where $K_i = \cos\left(\arctan 2^{-i}\right) = \sqrt{(1+2^{-2i})}$ is known as gain factor for each iteration.

In iterative terms, the signal can be represented with known numbers of iteration, the equation can be given by :

$$v_{i+1} = v_i.K_i.(1 + j\delta_i.2^{-i}) \quad (6)$$

The complex signal can be represented as:

$$x_{i+1} + jy_{i+1} = K_i.(x_i + jy_i).(1 + j\delta_i.2^{-i})$$
$$= K_i[(x_i - y_i.\delta_i.2^{-i}) + j(y_i + x_i.\delta_i.2^{-i})] \quad (7)$$

The simplified iterative CORDIC algorithm can be shown as follows.

$$x_{i+1} = K_i(x_i - y_i\delta_i 2^{-i})$$
$$y_{i+1} = K_i(y_i + x_i\delta_i 2^{-i}) \quad (8)$$

Table 1. Pre-Computed Angles for Pipelined CORDIC

i	$2^{-i} = \tan\alpha_i$	$\alpha_i = \arctan(2^{-i})$	α_i in radian
0	1	45°	0.7854
1	0.5	26.565°	0.4636
2	0.25	14.063°	0.2450
3	0.125	7.125°	0.1244
4	0.0625	3.576°	0.0624
5	0.03125	1.7876°	0.0312
6	0.015625	0.8938°	0.0156
7	0.0078125	0.4469°	0.0078
..	…..	……..	…….

3. Pipelined CORDIC Architecture

In this CORDIC architecture, a number of identical rotational modules have been incorporated and each module is responsible for one elementary rotation. Because of identical CORDIC iterations, it is convenient to map them into pipelined architecture [9]. The purpose of pipelined implementation is to device a minimum critical path. Therefore, this kind of architecture provides better throughput and lesser latency compared to other designs. It is associated with a number of stages of CORDIC Units where each of the pipelined stages consists of a basic CORDIC engine. The CORDIC engines are cascaded through intermediate latches as shown in **Figure 1**. The shift operations are hardwired using permanent oblique bus connections to perform multiplications by 2^{-i} reducing a large silicon area as required by barrel shifters.The critical-path of the pipelined CORDIC is the time required by the Add/Substract operations in each of the stages. Every stage contributes critical path delay amounts to $T_{Path}=T_{Add} + T_{MUX} + T_{2C,}$where T_{Add}, T_{MUX} and T_{2C} are the time required for addition , 2:1 Multiplexing and 2's Complement operations, respectively. The pre-computed angles (**Table 1**) of i-th iteration angle α_i required at each CORDIC engine can be stored at a ROM memory location, are known. Therefore, the need of multiplexing and sign detection is avoided to reduce critical path. The latency of computation is thus depends primarily on the adder used. Since no sign detection is needed to force $z_f = 0$, the carry save adders are well suited in this architecture. The use of these adders reduces the stage delay significantly. The delay can be adjusted by using proper bit-length in the shift register.

With the pipelining architecture, the propagation delay of the multiplier is the total delay of a single adder. So ultimately the throughput of the architecture is increased to

a many fold as the throughput is given by: "1/delay due to a single adder". It implies that speed up factor becomes more than M and latency of the design is M times of the delay of a single adder. It is obvious that if we increase the number of iterations then the latency of the design also will increase significantly. If an iterative implementation of the CORDIC were used, the processor would take several clock cycles to give output for a given input. But in the pipelined architecture, it converts iterations into pipeline phases. Therefore, an output is obtained at every clock cycle after pipeline stage propagation. Each pipeline stage takes exactly one clock cycle to pass one output (Simulated output shown in **Figure 2**).

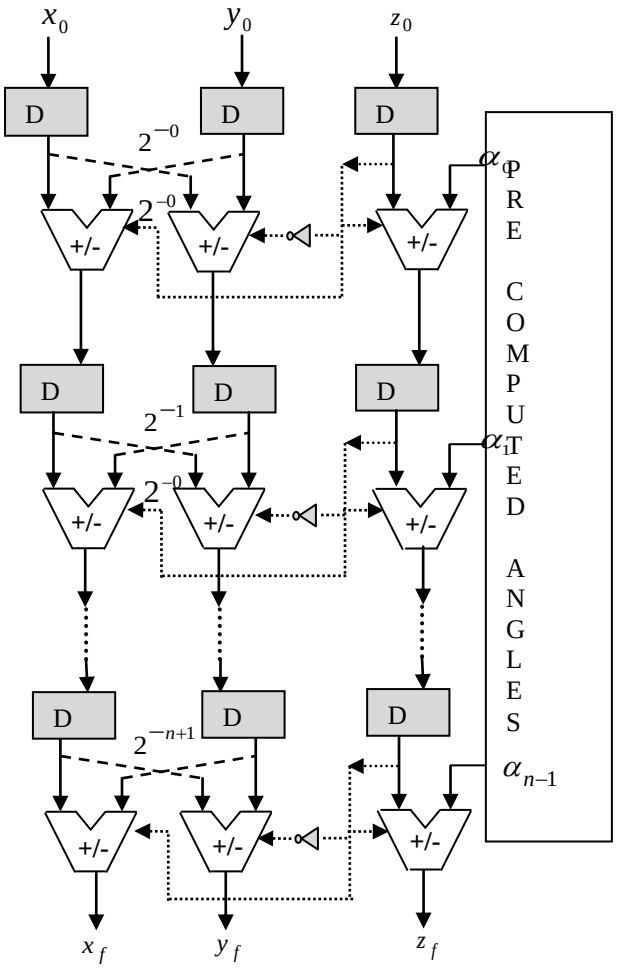


Figure 1. Pipelined CORDIC Architecture

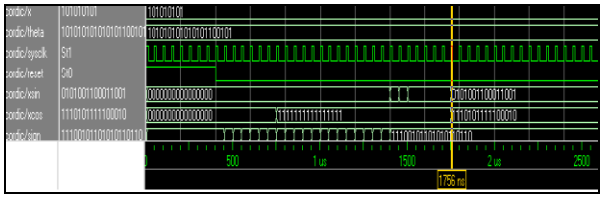


Figure 2. Simulation Result of CORDIC

The most recurrent problems for a CORDIC implementation are overflow. Since the first tangent value is $2^0 = 1$, then rotation range will be $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$. The difference in binary representation between these two angles is one bit. Overflow arises when a rotational angle crosses a positive right angle to a negative one. To avoid overflow, an overflow control is added. It checks for the sign of the operands involve in addition or subtraction and the result of the operation. If overflow is produced, the result keeps its last sign without affecting the final result. In the overflow control, the sign of z_i determines whether addition or subtraction is to be performed.

3.1. Numerical Analysis and Error Optimization

Theoretically, CORDIC realization has infinite number of iterations and that leads to accurate result. But practically CORDIC realization uses finite number of iterations resulting in approximation error. This kind of error arises due to approximations in angle as well as finite word length [7]. To get total approximation error, the error due to the angle approximation process will be derived and followed by the error due to the truncation of word length will be derived. The total error is taken as the summation of the two.

In the angle approximation process the angle θ is approximated as the algebraic sum of predefined elementary angles.

$$\theta = \sum_{i=0}^{M-1} \sigma_i \alpha_i + \delta \tag{9}$$

δ = Angle yet to be rotated after completion of the CORDIC iterations.

Due to convergence relationship of the CORDIC algorithm

$$\delta_{\max} = \tan^{-1} 2^{-(M-1)} \tag{10}$$

Let v^* be the ideal result obtained by the rotation of the vector $v(0) = [x(0) \quad y(0)]^t$ by an angle of θ. Let $\tilde{v}(M)$ be the output of the CORDIC block after scaling operation and assuming that there is infinite precision in the CORDIC operation module. Then, $v^* = \begin{bmatrix} \cos\delta & -\sin\delta \\ \sin\delta & \cos\delta \end{bmatrix} \tilde{v}(M)$ (11)

Let $D = \begin{bmatrix} \cos\delta & -\sin\delta \\ \sin\delta & \cos\delta \end{bmatrix}$

The error in the output due to the process of angle approximation is $v^* - \tilde{v}(M)$.

$$= v^* - \tilde{v}(M) = [D - I] * \tilde{v}(M) \tag{12}$$

It can be easily shown that

$$\| D - I \| = \sqrt{(\cos\delta - 1)^2 + (\sin\delta)^2}$$

$$= 2 * \sin|\delta/2|$$

$$\| D - I \| = 2 * \sin|\delta/2| \le 2 * |\delta/2| = |\delta| \tag{13}$$

From equation 9,

$$\| D - I \| \le |\delta| \le \tan^{-1}(2^{-(M-1)}) < 1/2^{M-1} \tag{14}$$

we can get a consolidated truncation error due to finite wordlength using scale factor K and number of finite iterations (M).

$$K*\sqrt{2}*2^{-b}(1+\sum_{j=0}^{M-1}(\prod_{i=j}^{M-1}\sqrt{(1+2^{-2i})}) \quad (15)$$

The scaling operation also introduces some error which amounts to maximum of 2^{-b}. So the final expression for the total quantization error is addition of all previously mentioned errors.

$$|\text{Total error}| \leq \frac{1}{2^{M-1}}*|v^*|+K*\sqrt{2}*2^{-b}(1+\sum_{j=0}^{M-1}\prod_{i=j}^{M-1}\sqrt{(1+2^{-2i})})+2^{-b} \quad (16)$$

Let the output of the CORDIC block has 12 bits in its fractional part. Therefore, the upper limit of the total quantization error can be taken as 2^{-12}.

$$|\text{Total error}| \leq \frac{1}{2^{M-1}}*|v^*|+K*\sqrt{2}*2^{-b}(1+\sum_{j=0}^{M-1}\prod_{i=j}^{M-1}\sqrt{(1+2^{-2i})})+2^{-b} \leq 2^{-12} \quad (17)$$

The above inequality is simulated in MATLAB to find out fractional bits of the internal word length of the CORDIC.

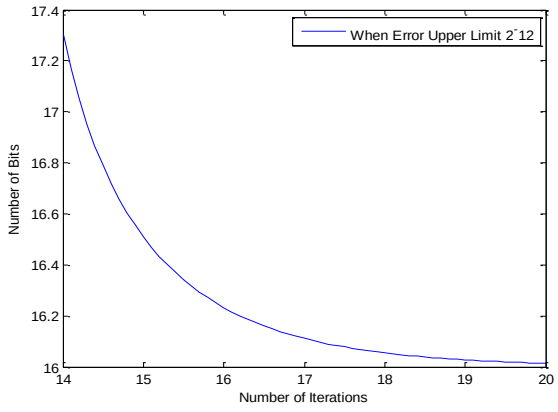


Figure 3. The required Bits Vs. Iterations for CORDIC Internal Design

From the **Figure 3**, it is shown that the optimum value for number of bits would be 17 for the latency to be kept as minimum as possible.

4. Basics of I/Q Channel Demodulation

As per the Euler's theorem, vector sum of cosine component is completely real whereas the spectrum of sine component is totally imaginary. If the cosine and sine components are combined, the resultant spectrum becomes one sided with direction of rotation (positive or negative frequency) and with known real (cosine) and imaginary (sine) components [11, 12].

$$\cos\omega_c t + j\sin\omega_c t = e^{j\omega_c t} \quad (18)$$

The process of recovering both real and imaginary signal component is known as *I/Q* demodulation. *I* stand for in-phase channel which processes cosine (real) components. *Q* stands for in-quadrature channel which processes sine (imaginary) component. The input of *I/Q* channel is Intermediate Frequency (IF) as shown in **Figure 4**. If the carrier frequency of IF is f_c with a time varying amplitude $a(t)$ and time varying phase $\phi(t)$, then input signal $s(t)$ will be: $S(t) = a(t)\cos[(2\pi f_c t + \phi(t)]$. (19)

In *I* channel, the IF signal is multiplied by reference carrier frequency produced by crystal oscillator at zero phase reference. The output of the *I* channel mixer is, *I(t)*, given by :

$$I(t) = a(t)\cos[(2\pi f_c t + \phi(t)].\cos(2\pi f_c t) = a(t)\cos[\phi(t)] + a(t)\cos[4\pi f_c t + \phi(t)] \quad (20)$$

The first term is the average value (DC) of the product and represents cosine of the signal phase and amplitude. The second term with high frequency component is suppressed by Low Pass Filter (LPF). So the output of the *I* channel is

$$I(t) = a(t)\cos[\phi(t)]. \quad (21)$$

Similarly, the *Q* channel output can be derived. The LPF output at *Q* channel is:

$$Q(t) = a(t)\sin[\phi(t)]. \quad (22)$$

Thus, *I* and *Q* channel together provide the amplitude and phase modulation.

5. Complex DPLL Demodulator for I/Q Channel Receiver

The down converted and filtered baseband signal has two components: real and imaginary parts. Therefore, the baseband signal can hold both the amplitude and phase of the sinusoidal signal at the same time. It does not hold any image frequency. So only loop filter is sufficient for the digital demodulation using Digital PLL [10]. Using the vector rotation operator $[x, y]^T \angle\theta$, the complex first-order DPLL demodulator equations for a given input signal can be stated as:

a). The real part of the output in phase comparator equation : $\varepsilon_n = \Re\{v(n)\angle -\theta_n\}$

b). The loop filter equation: $c_n = 2\pi K_l \varepsilon_n$,where K_l is the loop filter coefficient. The loop filter coefficient K_l depends on the sampling frequency and number of iterations of CORDIC algorithm. For M number of iterations, the loop filter coefficient K_l can be given by:

$$K_{CORDIC} = \frac{K_l}{\prod_{i=0}^{M-1}\sqrt{1+2^{-2i}}} \quad (23)$$

c). The phase accumulator equation:

$$\theta_{n+1} = (\theta_n + 2\pi K_l \varepsilon_n + \omega_c) \bmod 2\pi, \quad (24)$$

where $\omega_c = 2\pi f_c$ is the center frequency.

The CORDIC based DPLL (as shown in **Figure 5**) tries to adjust the continuous phase rotation in such a way that the complex component of the rotated vector will always be zero.

The simulation study has been carried out in MATLAB simulator. The down converted IF (30 MHz) is taken as input signal for the demodulator. The coherent oscillator frequency of 30 MHz has been taken as reference signal for demodulation purpose. The sampling frequency as well as clock frequency of 150 MHz has been used.

XILINX ISE 10.1 and simulated on ModelSim. The area utilization of proposed design is implemented on above said FPGA kit in terms of Sequential, Inverter and Logic circuits. The area consumed by Sequential, Inverter and Logic are 89%, 0.2% and 10.8% respectively of available resources as shown in **Table 3.**

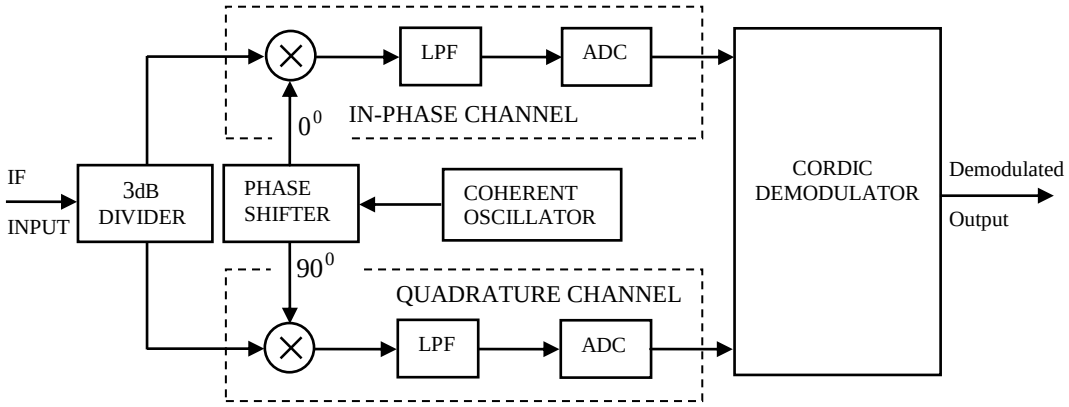


Figure 4. CORDIC Based DPLL in I/Q Channel

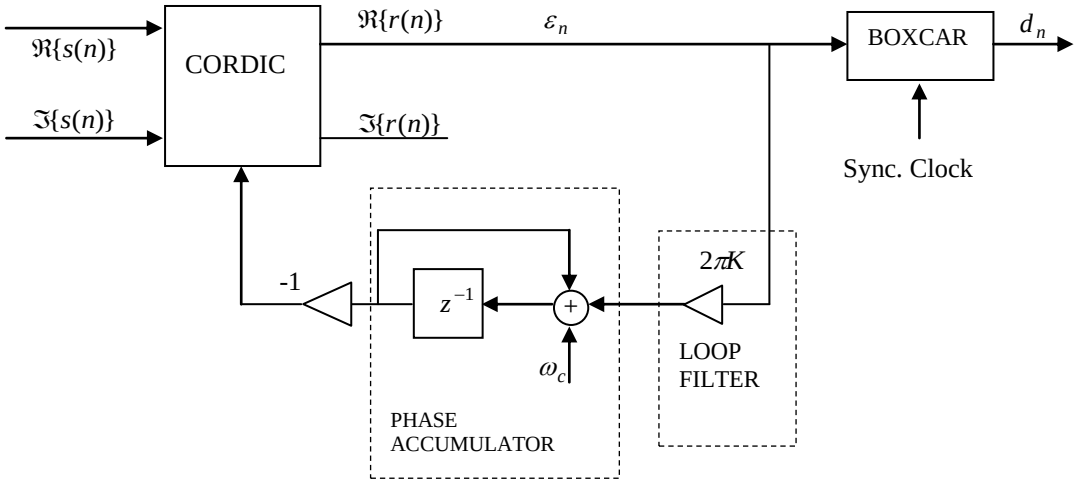


Figure 5. Complex DPLL using CORDIC

Initial phase shift of 30^0 is introduced for the simulation. Simulated radar detected signal with reference to coherent reference signal and the final phase-amplitude variation at the demodulator output have been shown in **Figure 6**, **Figure 7** and **Figure 8** respectively.

6. Hardware Synthesis

The main part of VLSI design is optimization of design in terms of speed, power, resource utilization and delay etc.

6.1 Area Analysis

The proposed architecture design was synthesized on Spartan-3 based xc3s50pq208-5 FPGA device using

6.2 Power Analysis

In the digital IC design, the main sources of power consumption are due to static (P_{st}) and dynamic (P_{dyn}).

The total power consumption by a circuit can be written as:

$$P_T = P_{st} + P_{dyn} \quad (25)$$

Static power is generated from reverse-biased diode leakage current and sub-threshold conduction. Both the factors are technology dependent. Whereas dynamic power, consist of short circuit (Psh) and switching (P_{sw}) power, depends on circuit design.

$$P_{dyn} = P_{sh} + P_{sw} \quad (26)$$

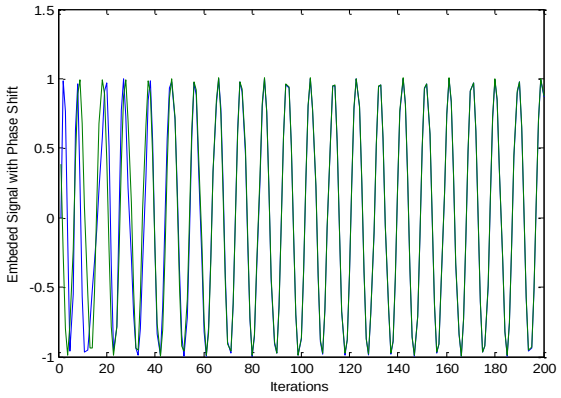
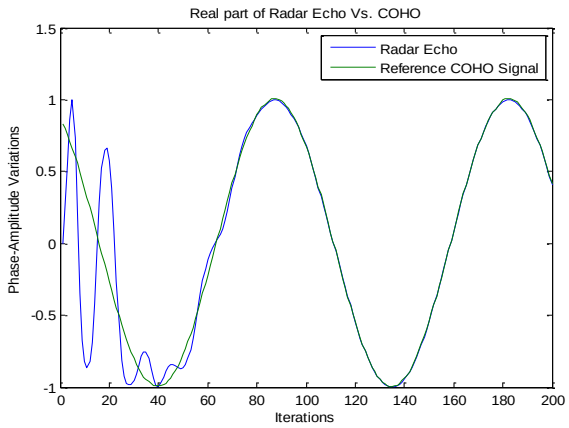
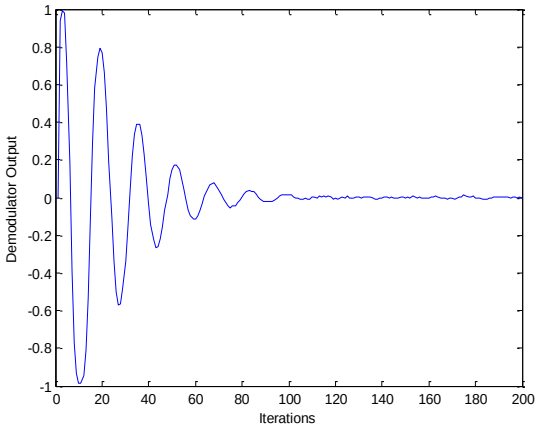


Figure 6. The Echo Signal and Reference COHO as a Input



to Radar Demodulator

Figure 7. Expanded view of the Echo Signal and Reference



COHO as a Input to Radar Demodulator

Figure 8. The Demodulated Phase-Amplitude Output

The switching power can be calculated as:

$$P_{sw} = 0.5 * k * C_L * V_{\sup}^2 * f \quad (27)$$

where $V_{\sup}$ is the supply voltage and k is the switching activity that determines the number of transition ($0 \to 1$ and $1 \to 0$) per data cycle and f is the main clock frequency. Since speed or throughput of the device depends on the clock frequency, it should not be reduced. So to get reduced dynamic power, capacitive load (C_L) and k can be reduced by using less number of logic, smaller devices, fewer and shorter wires.

The result of power measurement for proposed architecture also has been measured using Cadence custom IC design tool and shown in **Table 2**.

Table 2. Power Measurement

Cells	Leakage Power (nW)	Dynamic Power (µW)	Total Power (µW)
95	112.132	112.64	112.752

Table 3. Area Utilization

Types	Instances	Area	Area %
Sequentials	59	3941.784	89.0
Inverter	1	6.653	0.2
Logic	35	479.002	10.8
Total	95	4427.438	100.0

7. Conclusions

The paper presents the demodulation technique in a high speed Radar receiver using an application specific CORDIC processor based complex Digital PLL. With using reduced number of micro-rotation and adequate optimized convergence property of CORDIC design, implementation of this kind of demodulator becomes easier. The architecture given in this paper enhances throughput and minimizes latency that facilitates its use in real time signal processing like digital demodulation in radar receiver. Numbers of micro-rotations have been adjusted so as to achieve better loop performance and speed of operation while minimizing quantization error. The inherent issue of CORDIC i.e. overflow is quite appropriately resolved using proposed design. The property of good convergence of CORDIC is efficiently used in this application.

6. References

[1] J. E. Volder, "The CORDIC Trigonometric Computing Technique", *IRE Transactions on Electronic Computing*, vol. EC-8, pp. 330-334, Sept, 1959.

[2] Y. H. Hu, "CORDIC-Based VLSI Architectures for Digital Signal Processing", *IEEE Signal Processing Magazine*, vol. 9, no. 3, pp. 16-35, 1992.

[3] S. Aggarwal, P. K. Meher, K. Khare, "Area-Time Efficient Scaling-Free CORDIC Using Generalized Micro-Rotation Selection", *IEEE Transactions on VLSI System*, vol. 20, no. 8, pp. 1542-1546, Aug, 2012.

[4] S. Ugazio, "Design of real time adaptive DPLLs for generic and variable Doppler frequency", in *International Conference on Localization and GNNS (ICL-GNSS)*", pp. 169-174, Tampere, Jun, 2011.

[5] B. K. Kang, H. J. Kwon, B. K. Mheen, H. –J. Yoo and Y. H. Kim, "Nonlinearity Compensation Circuit for voltage-controlled oscillator operating in linear frequency sweep

mode", *IEEE Microwave and Guided Wave Letters*, vol. 10, no. 12, pp. 537-539, Dec 2000.
[6] K. Kota, J. R. Cavallaro, "Numerical accuracy and hardware trade-offs for CORDIC arithmetic for special purpose processors", *IEEE Transactions on Comuters*, vol. 42, no. 7, pp. 769-779, Jul 93.
[7] Y. H. Hu, "The Quantization Effects of the CORDIC Algorithm", *IEEE Transactions on signal processing*, vol. 40, no. 4, pp. 834-844, Apr 1992.
[8] T. –Y. SUNG, H. –C. HSIN, "Fixed-Point Error Analysis of CORDIC Arithmetic for Special-Purpose Signal Processors", *IEICE Trans. Fundamentals*, vol. E90-A, no. 9, Sept 2007.
[9] A. Banerjee, A. S. Dhar, "Pipelined VLSI Architecture using CORDIC for Transform Domain Equalizer", *journal of signal Processing systems*, vol. 70, pp. 39-48, 2013.
[10] J. Vuori, "Implementation of a Digital Phase-Locked Loop Using CORDIC Algorithm", *IEEE International Symposium on Circuits and Systems*, vol. 4, pp. 164-167, Atlanta, 1996.
[11] W. Xu, C. Gu, C. Li, M. Sarrafzadeh, "Robust Doppler radar demodulation via compressed sensing", *Electronics Letters*, vol. 48, no. 22, pp. 1428-1430, Oct, 2012.
[12] B. R. Mahafza, A. Elsherbeni, "MATLAB Simulations for Radar Systems Design", CRC, Florida, USA, 2003.

Amritakar Mandal received his Bachelor of Engineering Degree in Electronics and Communication Engineering from The Institution of Engineers (India) in 2006 . He completed M.Tech with specialization in VLSI from Shobhit University, India in 2009. He is currently working toward the Ph.D Degree in Electronics and Communication Engineering in the Department of Information and Communication Technology , Gautam Buddha University, India. He has vast experience in Electronic Warfare and Surveillance Radar systems. His research interests include Radar Signal Processing and High Speed VLSI Design for Communication Systems.

Rajesh Mishra is currently working as Assistant Professor in School of Information and Communication Technology, Gautam Buddha University Greater Noida, Delhi NCR (India). He received his BE (Electronics Eng.), M. Tech and Ph. D. degree (Reliability Eng.) from Reliability Engineering Centre, IIT Kharagpur (India) in year 2000, 2004, and 2009 respectively. He has research interest in the area of reliability engineering, layout design for capacitated networks, and network optimization. He has published papers in several international journals such as IJPE, IEEE, RESS, QTQM etc.