ISSN 2395-1621

FPGA Implementation of Frequency Up-Down Converter Using CORDIC Algorithm

#1Ms. Nikam R.H., #2Ms. Mane S.N, #3Ms. Raut P.V., #4Ms. Choudhari M.D.

[1]rnikam512@gmail.com
[2]swati.mane44@gmail.com
[3]rautpriya77@gmail.com
[4]meghachoudhari99@gmail.com

#1assistant professor with Department Of EnTC, PES's College Of Engineering Phaltan, Shivaji University, Kolhapur, Maharashtra, India
#234 Graduate student with Department Of EnTC,PES's College Of Engineering Phaltan, Shivaji University, Kolhapur, Maharashtra, India

ABSTRACT

CORDIC algorithm is widely used for Image processing, Robotics, Fiber optics, VLSI implementation of DSP Applications,etc. This paper shows how to use CORDIC to implement Direct Digital Synthesizers (DDS), Up-Down converters of in phase and quadrature phase signals. Also this paper shows how to use CORDIC in Software Defined Radio(SDR).The CORDIC algorithm used in Rotational mode for frequency up-down converter.In Rotational mode there is a rotatation of a vector by an angle, and convert from polar to Cartesian coordinates.SDR is a collection of hardware and software technologies where some or all of the radios operating functions are implemented through modifiable software or firmware operating on programmable processing technologies. The focus of this paper is to analysis and simulation of up-down conversion using Direct Digital Synthesizer having CORDIC algorithm at the place of ROM Lookup Table.

Keywords— CO-ordinate Rotation Digital Computer (CORDIC),Direct Digital Synthesizer (DDS), Digital Up Converter (DUC), Digital Down Converter (DDC), Field programmable gate-array (FPGA), Look Up Table (LUT), Software Defined Radio (SDR)

ARTICLE INFO

Article History

Received :2nd March 2016

Received in revised form :

3rd March 2016

Accepted :6th March, 2016

Published online :

7th March 2016

I. INTRODUCTION

The Coordinate Rotation Digital Computer (CORDIC) was introduced by Jack Volder in 1959.[1]. It is a versatile algorithm widely used for digital signal processing applications. It iteratively performs the rotation of a two-dimensional vector using only add and shift operations therefore this algorithm is easy to implement. CORDIC has been used for hardware implementations in VLSI applications. Some algorithms such as linear transforms, digital filtering, and matrix based DSP computing algorithms offering efficient implementation using CORDIC. It was shown that as compared to the conventional multiply-and-add hardware method CORDIC is very much easy ,so it can be a good alternative for conventional method. CORDIC can also applied in implement of communication subsystems used in a digital radio: direct digital synthesizers; amplitude modulation (AM), phase modulation (PM), and frequency modulation (FM) analog modulators; amplitude shift keying (ASK), phase shift keying (PSK), and frequency shift keying (FSK) modulators, up-/down converters of inphase and quadrature signals, full mixers for complex signals, and phase detection for synchronizers etc.

II.CORDIC FUNDAMENTALS

CORDIC is presented only as a computational resource with three inputs ($X0$, $Y0$, and $Z0$) and three outputs (XN, YN, and ZN) for understanding of how to use the CORDIC algorithm in the implementation of digital intermediate frequency (IF) communications systems. This allows performing the following operations[3]as shown in Figure 1.

• Rotation Mode (RM): In Rotation mode the co-ordinates of a vector (I, Q) are changed from polar to Cartesian co-ordinates. Also vector is rotate by an angle q when it is operating in RM; after that output vector is multiplied *K*. *K is a constant value.*

• Vectoring Mode (VM): When it is operating in vectoring mode (VM) Cartesian-to-polar conversion is obtained; here is also answer is multiplied by constant value K..

Figure 1. shows a generic scheme of RM CORDIC which shows how to implement different digital communication tasks . The figure 1. is composed of an RM CORDIC where signals I is connected to *X0* and Q is connected to *Y0* inputs, and the phase term q connected to *Z0* input is $q = (\Sigma[fc + fm]) + Øm) \times \pi$, where q is composed of the accumulation, at a sample period of *Ts*, of two frequency terms, *fc* and *fm*, and a phase term, *Øm*. The signed modulo-1 (limited to the interval [–1,1]), addition technique is used in this computations and the frequency and phase terms *fc*, *fm* and ϕ*m* are normalized to constant value 1. The *Z0* input of CORDIC block needs a phase input in the interval [–π, π], so to extend the interval of the normalized term q , multiplication by π is required for the interval required by CORDIC [3].

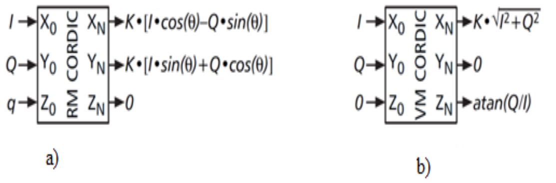

Figure 1 a) Rotation mode b) vectoring mode CORDIC.

Instead of perfect rotation, CORDIC uses a pseudo-rotation of a two-dimensional vector, where the original vector is rotated by an angle q, and its magnitude is enlarged by a constant factor *K*.

Iteratively computation of angle q is performed by pseudo-rotation with the following iterations in CORDIC algorithm:

Xi+1 = [*Xi*– *di*× 2–*i*× *Yi*]

Yi+1=[*Yi*+*di*×2–*i*×*Xi*] ………… (1)

Zi+1 = *Zi*– *di*× a*i*

CORDIC performs several micro-rotations by the angles ai=±*atan* (2–*i*) ,instead of directly performing a rotation by the angle q. This means that the rotation angle q is broken down into a set of predefined angles a*i*, so after a number of iterations the angle q is approximated by Σ (*di*× ai), where *di* belongs to the set {–1, 1}.Eq.(1) shows the two operating modes of CORDIC, i.e. the rotation mode (RM) and the vectoring mode (VM), which depend on how the directions of the micro rotations (*di*) are chosen:

di= *sign* (*Zi*) and *di*= –*sign* (*Yi*) for RM and VM respectively.

III. CORDIC ALGORITHM USED IN FREQUENCY CONVERSION SYSTEM

A. DIRECT DIGITAL SYNTHESIS

Waveforms in digital domain are obtained directly through the DDS. Mostly in communications systems the output waveforms are the sine and cosine waves. A DDS is combination of a phase accumulator and a phase-to amplitude converter [3], as shown in Fig. 3 a. In a conventional DDS based on lookup tables (LUTs) the phase accumulator is an integer *N*-bit accumulator (an unsigned modulo-2N accumulator), whose output directly addresses the LUT where the amplitude values of sine or cosine waves are stored. The maximum value of the accumulator (2N–1) represents the phase 2 π of the sine or cosine wave. The accumulator generates a ramp signal when it is incremented by a fixed value, due to its unsigned modulo-2N property; hence, a periodic waveform is obtained at the output of the phase-to-amplitude converter (Fig. 3 b).

The RM of CORDIC algorithm can operate as a quadrature phase-to-amplitude converter that directly generates sine and cosine waveforms [4]. The main advantage of using CORDIC-based DDS with respect to LUTbased methods is that it can achieve both high phase resolution and high precision with lower hardware cost [5]. A difference between LUT-based method and CORDIC based DDS is that accumulator generates an integer value that addresses an LUT in the LUT-based method, while it generates an angle in CORDIC-based DDS. Thus, in the last case a ramp signal in the interval [–π,π,] must be obtained by the accumulator,as shown in Fig. 3c. The N-bit adder implementation is easily done using accumulator. A two's complement fractional numeric format (only one integer bit) is considered; hence, output is a ramp signal in the interval [–1,1], and a multiplier by π is used to get the desired range.

To generate sine and cosine waveforms of a digital frequency *fc* with the scheme based on CORDIC of Fig. 2, the parameters *fm*, *Øm*, and *Q* must be zero and I = 1/*K*. By giving fixed value to *fc* we can control oscillation frequency. In such a case CORDIC generates directly the cosine and sine waveforms (*si*(*n*) = *cos*(*fc*. π *n*) and *sq*(*n*) = *sin*(*fc* . π .*n*)) through *XN* and *YN* outputs, respectively. The maximum synthesized frequency is obtained by taking the value *fc* = 1 (which is equivalent to analog frequency of *fs*/2); and the minimum frequency is achieved with *fc* = 2–(*N*–1), where *N* is the word-length of the accumulator.

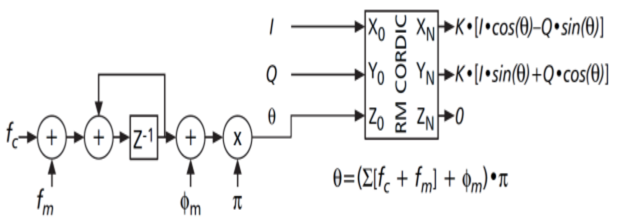

Figure 2 Generic scheme to use CORDIC in Rotation Mode.

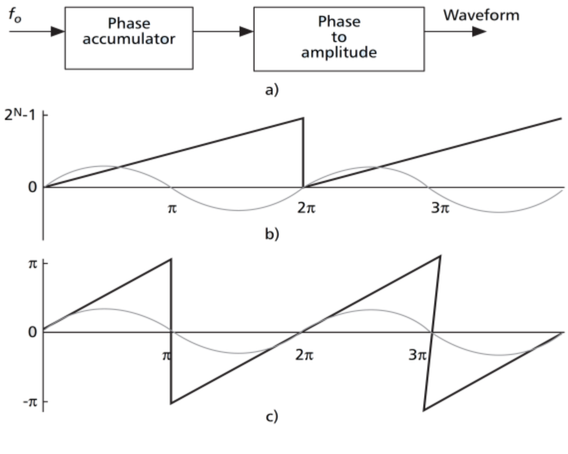

Figure 3 a) DDS block diagram; b) waveforms of the LUT based method; c) waveforms of the CORDIC-based method

B.UP-DOWN CONVERSION:DIGITAL MIXERS IN QUADRATURE MODULATION

Digital up-down conversion to/from an IF is a typical solution in SDR systems. In QAM a bits tream is grouped in symbols, and symbols are divided into their in-phase and quadrature components that are pulse shaped and interpolated up to the mixer rate. In an SDR receiver, the sampled received signal is mixed, and the oversampled in-phase and quadrature branches are decimated and filtered by the matched filter. The upconversion mixer, shown in Fig. 4a, can be implemented with the CORDIC RM scheme of Fig. 2. Parameters *fm* and f*m* must be zero, and the baseband in-phase and quadrature signals *si*(*n*) and *sq*(*n*) are connected to *X*0 and *Y*0 from the output *XN*, a scaled-by-*K* version of the mixed signal (*s*(*n*) = *K* × [*si*(*n*) × *cos(fc* × π× *n*)– *sq*(*n*) × *sin*(*fc* × π × *n*)]) is obtained.

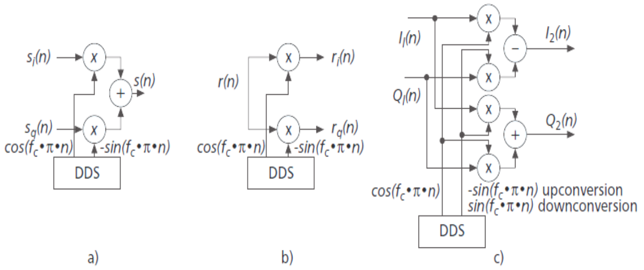

Figure 4 a) Half mixer upconverter; b) Half mixer downconverter; c) Complex mixer up-downconverter

In the receiver side, after sampling, the IF signal is mixed. This operation, depicted in Fig.4b, can again be implemented using a CORDIC RM: the received signal *r*(*n*) must be connected to *Y*0, and *X*0 must be zeroed. The scaled-by-*K* version of the in-phase and quadrature oversampled received signals (*ri*(*n*) = *K* × *r*(*n*) × *cos(fc* × π × *n*) and *rq*(*n*) = –K × *r*(*n*) × *sin*(*fc* × π × *n*)) are obtained from *YN* and *XN*, respectively.

Digital downconversion based on the Hilbert transform, and carrier and frequency synchronization are situations where complex mixing (Fig. 4c) of quadrature signals is required. This mixing can be seen as a multiplication of a complex signal (*I* + *jQ*) by a complex exponential e^{jw}: if this frequency is positive (like in up-conversion) the in-phase signal *I*1 is connected to *X*0, and the quadrature signal *Q*1 to *Y*0, and their respective in-phase and quadrature outputs (*I*2 = *K* × [*I*1 × *cos(fc* × π × *n*) – *Q*1 × *sin* (*fc* × π × *n*)] and *Q*2 = *K* × [*I*1 × *sin(fc* × π × *n*) + *Q*1 × *cos* (*fc* × π × *n*)]) are obtained from *XN* and *YN*, respectively. When the frequency is negative (as in downconversion) the in-phase signal *I*1 is connected into *Y*0 and the quadrature *Q*1 into *X*0, being the in-phase and quadrature output signals (*I*2 = *K* × [*Q*1 × *sin(fc* × π × *n*) + *I*1 × *cos(fc* × π × *n*)] and *Q*2 = *K*([*Q*1 × *cos(fc* × π × *n*) – *I*1 × *sin(fc* × π × *n*)]) obtained from *YN* and *XN*, respectively.

The main advantage of using CORDIC as a digital mixer with respect to the conventional scheme of Fig. 4 (LUT-based DDS and multipliers) is that the multipliers and the ROM to store the sine and cosine waveforms are avoided. The extreme case is given by the complex mixer in which four multipliers and the LUT-based DDS is replaced by a single RM CORDIC.

IV.RESULTS & DISCUSSION

A MATLAB SIMULATION

To obtain the resultant waveforms we need the matlab simulation. Here are the results of system which are shown in fig 4a, fig 4b and fig.4c respectively.

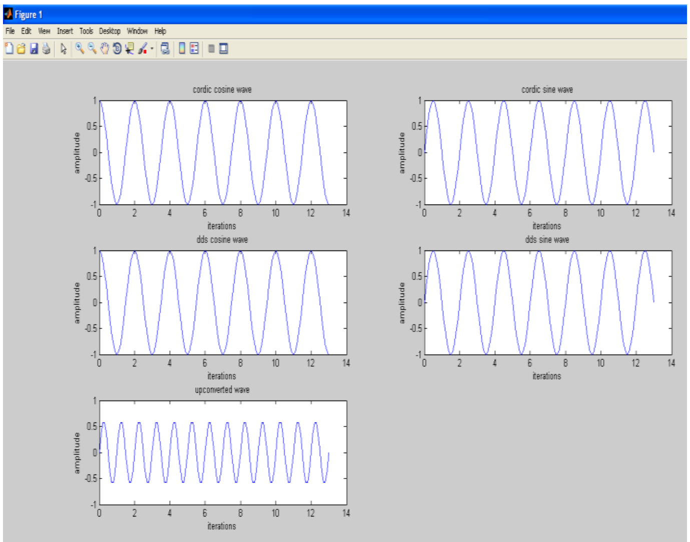

Fig5a.Half mixer upconverter output in matlab

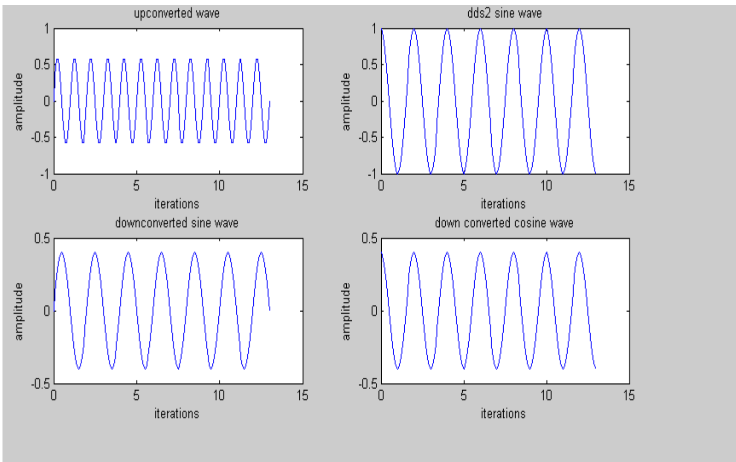

Fig5b.Half mixer downconverter output in matlab

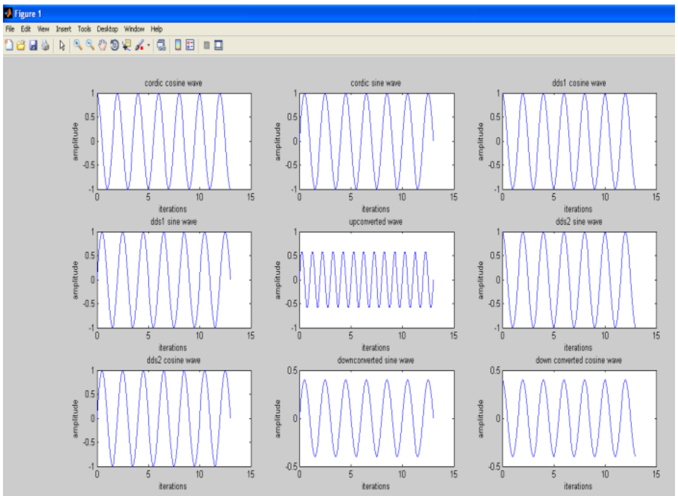

Fig5c.Complex mixer up-downconverter output in matlab

V.CONCLUSION

This paper presents the use of CORDIC algorithm for Direct Digital Synthesizer. CORDIC algorithm is best technique for phase to sine amplitude conversion. The CORDIC design used here is based on Pipeline data path Architecture. With the pipeline architecture, the design is able to calculate continuous input, has high throughput, and doesn't need ROM or registers to save constant angle iteration of CORDIC. CORDIC algorithm provides fast and area efficient computations of sine and cosine functions without using ROM LUTs. This paper is focused on the Direct Digital Synthesizer using CORDIC approach, to increase the speed with minimum area requirement in FPGA. Also this paper reveals that how to use the CORDIC algorithm to implement different blocks found in communication systems like up-down converter.

REFRENCES

[1] J. E. Volder, "The CORDIC trigonometric computing technique," IRE Trans. Electronic Computing, volume EC-8, pp330 – 334, 1959.

[2] Pramod K. Meher, Javier Valls,Tso-Bing Juang, K. Sridharanand KoushikMaharatna "50 Years of CORDIC: Algorithms, Architectures, and Applications" IEEE Transactions On Circuits And Systems—I: Regular Papers, Vol. 56, No. 9, pp 1893-1907,September 2009

[3] J. Valls, T. Sansaloni, A. Perez-Pascual, V. Torres, and V. Almenar, "The use of CORDIC in software defined radios: A tutorial," IEEE Commun. Mag., vol. 44, no. 9, 2006.

[4] J.H.Reed,"Modern Engineering Approach to Software Defined Radio. "

[5]Y.H. Hu, "Pipelined CORDIC architecture for the implementation of rotational based algorithm," in Proceedings of the International Symposium on VLSI Technology, Systems and Applications, p. 259, May 1985.

[6]R.Andraka, "A survey of CORDIC algorithms for FPGA based computers, Proceedings of ACM/SIGDA sixth International Symposium on field Programmable Gate Arrays", 1998,pp.191-200

[7] J. E. Meggitt, "Pseudo division and pseudo multiplication processes," IBM Journal, vol. 6, no. 2, pp. 210–226, 1962.

[8] C.H.Lin and A.Y. Wu, "Algorithm and Architecture for High-Performance Vector Rotational DSP Applications," Regular IEEE Transactions: Circuits and Systems I, Volume 52, pp 2385- 2398, November 2005.

[9] J.S. Walther, "A unified algorithm for elementary functions," Proc. Spring. Joint Comp. Conf., vol. 38, pp. 379-385, 1971.

[10] M.D. Erecegovac and T. Lang, Digital Arithmetic, Elsevier, Amsterdam, the Netherlands, 2004.

[11] L. Cordesses, "Direct digital synthesis: A tool for periodic wave generation (part 1)," IEEE Signal Process. Mag., vol. 21, no. 4, 2004.

[12] J. Vankka "Digital Synthesizers and Transmitters for Software Radio." Dordrecht, Netherlands: Springer, 2005.

[13] J. Vankka, "Methods of Mapping from Phase to Sine Amplitude in Direct Digital Synthesis," IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, vol. 44, pp. 526-534, March 1997.

[14] J. Vankka, "A Direct Digital Synthesizer with a Tunable Error Feedback Structure," IEEE Transactions on Communications, vol. 45, pp. 416-420, April 1997.

[15] J. Vankka, "Digital Modulator for Continuous Modulations with Slow Frequency Hopping," IEEE Transactions on Vehicular Technology, vol. 46, pp. 933-940, Nov. 1997.

[16] J. Vankka, M. Waltari, M. Kosunen, and K. Halonen, "Direct Digital Syntesizer with on- Chip D/A-converter," IEEE Journal of Solid-State Circuits, Vol. 33, No. 2, pp. 218-227, Feb.1998.

[17] M. Kosunen, J. Vankka, M. Waltari, L. Sumanen, K. Koli, and K. Halonen, "A CMOS Quadrature Baseband FrequencySynthesizer/Modulator",Analog Integrated Circuits and Signal Processing, Vol. 18, No. 1, pp. 55-67, Jan. 1999.

[18] Jouko Vankka, "Direct Digital Synthesizers: Theory, Design and Applications." , November 2000.

[19] J. Vankka, "Methods of Mapping from Phase to Sine Amplitude in Direct Digital Synthesis", pp. 86–94, in (edited by) V. F. Kroupa, "Direct Digital Frequency Synthesizers", 1999, IEEE Press.

