SIMULATION OF FIR FILTER BASED ON CORDIC ALGORITHM

Shalini Rai and Rajeev Srivastava

Department of Electronics & Communication,
University of Allahabad, Allahabad (UP)

ABSTRACT

Coordinate Rotation Digital Computer (CORDIC) discovered by Jack E Volder. It is a shift-add operation and iterative algorithm. CORDIC algorithm has wide area for several applications like digital signal processing, biomedical processing, image processing, radar signal processing, 8087 math coprocessor, the HP-35 calculator, Discrete Fourier, Discrete Hartley and Chirp-Z transforms, filtering, robotics, real time navigational system and also in communication systems. In this paper, we discussed about the CORDIC algorithm and CORDIC algorithm based finite impulse response low pass & high pass filter. We have generated the M-code for the CORDIC Algorithm and CORDIC Algorithm based FIR filter with the help of MATLAB 2010a.We also discussed about the frequency response characteristics of FIR filter.

KEYWORDS

CORDIC Algorithm, FIR Filter, MATLAB

1. INTRODUCTION

Filtering process is the process for refining of the signals regarding the applications. We categorized the filter in main two parts analog filters and digital filters [7, 8]. Digital filters are more advantageous than analog filters. Digital filters are used in DSP applications [3]. There are basically four types of filter structure like low pass filter, high pass filter band pass, band reject filter. The field programmable gate array (FPGA) is bench for the implementation of FIR and IIR digital filters in the VLSI area. In the last five decades the CORDIC algorithm [1, 2] is the very popular algorithm to producing the fast VLSI implementations. It is a hardware efficient algorithm i.e. optimizing the area, speed, power and hardware cost. It is 2-D rotational vector algorithm. The extended form of algorithm is a unified algorithm [5, 6] for the computations of rotation in hyperbolic, linear, circular coordinates systems. Basically it is used for the computations of several trigonometric functions, hyperbolic function and logarithmic functions of real and complex numbers. As time passes there are several advancements in CORDIC algorithm for reduction of the number of iterations, like the angle-recording (AR), modified vector rotation, mixed scaling rotation (MSR) and scaling free CORDIC algorithms have been proposed for reduction of no of iteration, improving the system performance and speed up the system. In this paper we have discussed about the CORDIC Algorithm, FIR filter in section II, III respectively. In section IV we have to deal with designing of FIR filter based on CORDIC algorithm and its frequency response characteristics with the help of MATLAB 2010a.In section V we have discussed about the Results and conclusion.

DOI : 10.5121/vlsic.2017.8402

2. THEORY OF CORDIC ALGORITHM

Coordinate Rotation Digital Computer (CORDIC) was implemented in 1959 by J.E.Volder. It is used in two different modes one is rotation mode and vectoring mode. Overall the algorithm can be realized as an iterative sequences of additions or subtractions and shift operations by using the two modes, which are rotated by a fixed rotation angle (μ- rotations).The CORDIC algorithm is in the general rotation transform[3,4]-

$$x' = x\cos\phi - y\sin\phi \quad (1)$$
$$y' = x\sin\phi + y\cos\phi \quad (2)$$

The above equations can readjusted as

$$x' = [x - y\tan\phi]\cos\phi \quad (3)$$
$$y' = [y + x\tan\phi]\cos\phi \quad (4)$$

These rotation of angles constrained so $\tan(\phi) = \pm 2^{-i}$. This will reduces tangent multiplication by simple shift operation. i is the no iteration. The angle ø decomposes into elementary rotations in sequence manner.

$$\Phi = \Sigma\alpha i \quad (5)$$

So iterative equations of Cordic Algorithm are

$$xi+1 = [xi - yi\tan\alpha i]\cos\alpha i \quad (6)$$
$$yi+1 = [yi + xi\tan\alpha i]\cos\alpha i \quad (7)$$

For the trigonometric identities

$$\cos(\alpha i) = 1/(1+\tan^2\alpha i)^{1/2} \quad (8)$$

Replace the term $1/(1+\tan^2\alpha i)^{1/2} = ki$

or $ki = 1/(1+2^{-2i})^{1/2}$ (9)

ki denotes as constant multiplication factor .The gain is defined as the inverse of the constant multiplication factor.

$$Ai = 1/ki \quad (10)$$

The system gain $An = \Pi[1+2^{-2i}] \approx 1.647$

So the above equation no.(6) and (7) becomes

$$xi+1 = ki.[xi - yi.di.2^{-i}] \quad (11)$$
$$yi+1 = ki.[yi + xi.di.2^{-i}] \quad (12)$$

di is the decision function depends the rotational mode.
First is the rotation mode and second is the vectoring mode

a) For the rotation mode

$$di= -1 \text{ if } \quad zi < 0 \qquad (13)$$
$$di = +1 \text{ if } \text{ else}$$

after n^{th} iteration it produces the following results

$$xn= an[x0cosz0- y0sinz0] \qquad (14)$$
$$yn= An[x0sinz0 + y0cosz0] \qquad (15)$$
$$zn =0 \qquad (16)$$

b) For vectoring mode

$$di = +1 \text{ if } y<0 \qquad (17)$$
$$= -1 \quad \text{else}$$

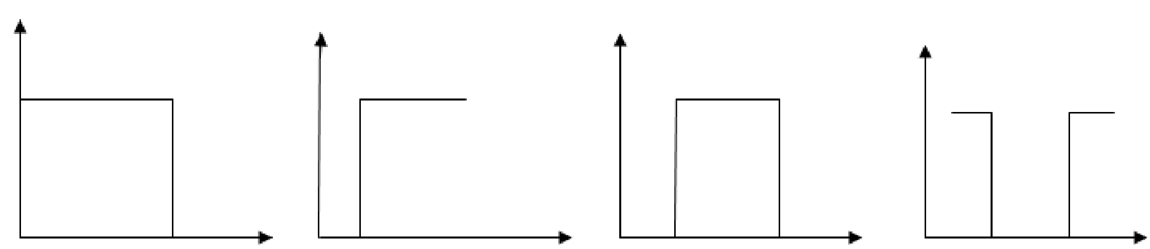
After n iteration it produces the following results

$$xn = An\ (x0^2+y0^2)^{1/2} \qquad (18)$$
$$zn= \tan^{-1}(y0/x0)+z0 \qquad (19)$$
$$yn=0 \qquad (20)$$

3. FINITE IMPLUSE REPONSE FILTER

In many applications of signal processing we want to change the relative amplitudes and frequency contents of a signal. This process is known as filtering .The ideal filters have a frequency response that is real and non – negative, i.e. has a zero phase characteristics. A linear phase characteristics introduces a time shift and this causes no distortion in the shape of the signal in the pass-band. Since the Fourier transfer of a stable impulse response is continuous function of ω, cannot get a stable filter.

An ideal frequency selective filter passes complex exponential signal. For a given set of frequencies and completely rejects the others. In figure 1 shows frequency response for ideal low pass filter (LPF), ideal high pass filter (HPF), ideal band pass filter (BPF) and ideal backstop filter (BSF).



(a) Low Pass Filter (b)High Pass Filter (c) Band Pass Filter (d) Band Reject Filter

Figure1: Ideal Filter Frequency Response

Finite Impulse Response: The term digital filter arises because these filters operate on discrete-time signals. Finite Impulse Response is the filter in which output response is equal to the weighted finite sum of past, present and perhaps future values of the filter input, i.e.

$$y[n] = \sum_{k=-M1}^{M2} b_k x[n-k] \qquad (21)$$

where both M1 and M2 are finite. An FIR filter is based on a feed-forward filter. Feed forward means that there is no feedback of past or future to form the present output, just input related terms. The causal FIR filters has difference equation of the form

$$y[n] = \sum_{k=0}^{M} b_k x[n-k] \qquad (22)$$

The time domain impulse response of a filter corresponding to a given (desired) frequency response may be calculated from the inverse Fourier transform of the desired frequency response:

$$H_d(n) = ½*\Pi \int_{-\Pi}^{\Pi} H_d(\omega)\, e^{jn\omega}\, d\omega$$

The samples $h_d(n)$ from the above are time domain values, as indicated by the index n. These are the time domain samples that would have the frequency response $H_d(\omega)$.The conceptual leap is that we use these numbers as weighting coefficients in a difference equation to form filter itself.

For M=3 the FIR Filter

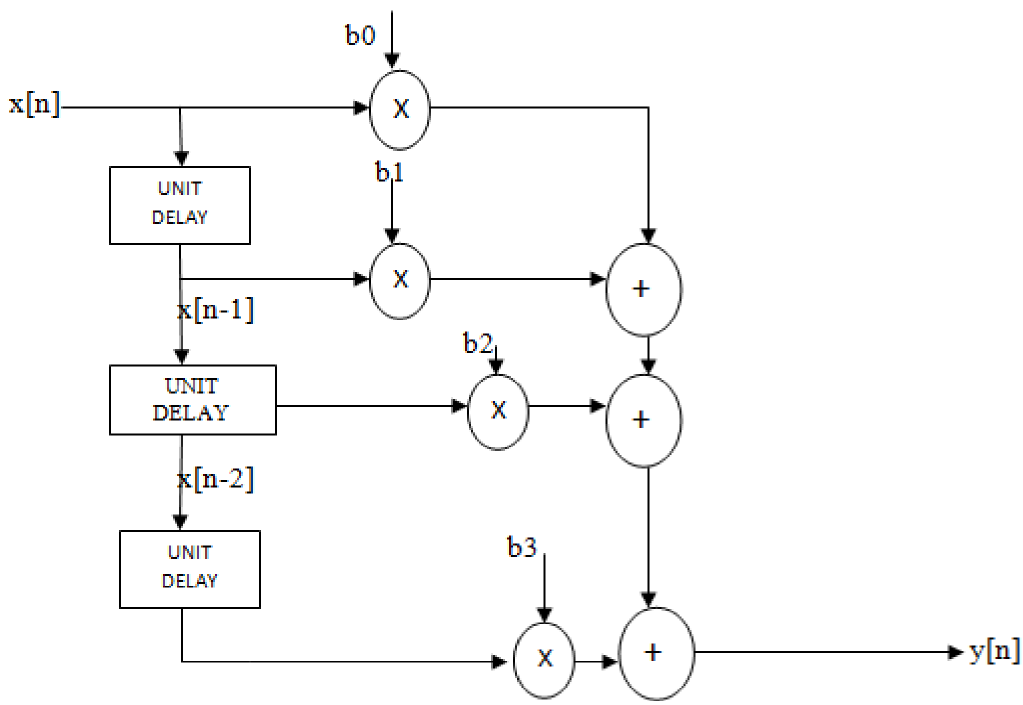


Figure2: Fir filter for M=3

The impulse response $h_d(n)$ thus computed will be infinite in extent. In a practical filter, the order must be limited. This is obtained by truncating the impulse response. Assuming N is odd, the calculation of $h_d(n)$ over the range

$$-(N-1)/2 \leq n \leq (N-1)/2$$

4. CORDIC BASED HIGH PASS FIR FILTER

In this paper we design the CORDIC algorithm based High pass FIR filter with help of MATLAB2010a and by its simulink tool. We have chosen the arbitrary frequency response equation of a High pass FIR filter [10].

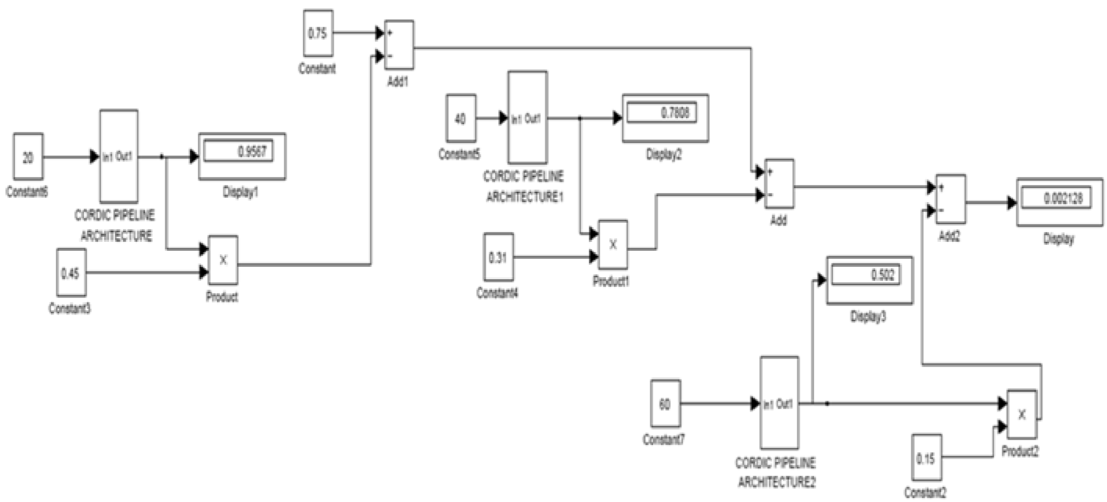


Figure 3: Simulation of high pass filter based on CORDIC algorithm

Angle (w in degree)	Output Magnitude Value
10	-0.1291

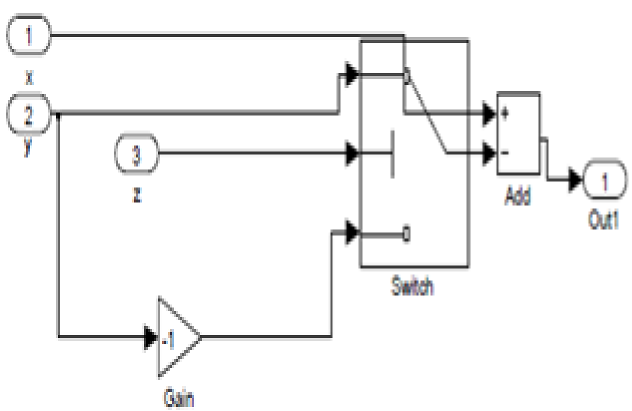


Figure 4: Adder System

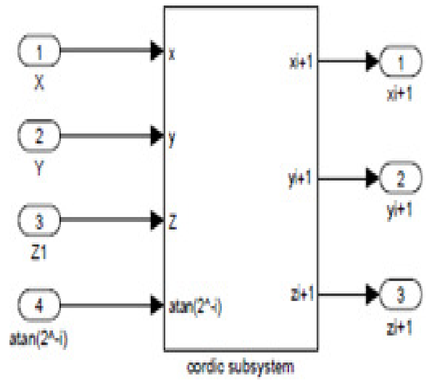
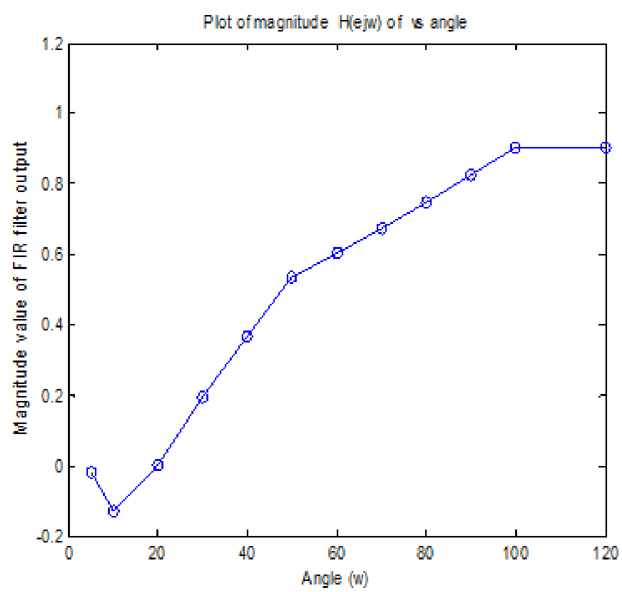


Figure 5: CORDIC Subsystem

5. RESULTS AND CONCLUSION

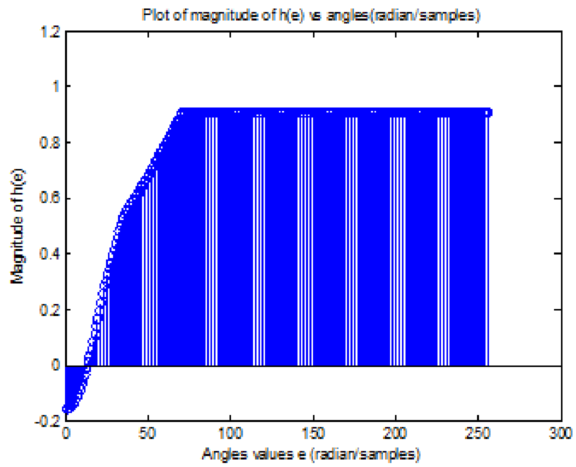
PLOT OF MAGNITUDE VALUE VS ANGLE (BY SIMULINK OUTPUT)

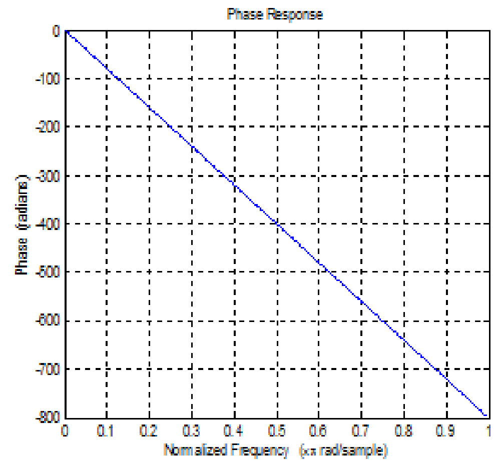
Angle (w in degree)	Output Magnitude Value
10	-0.1291
20	0.002128
30	0.1961
40	0.3673
50	0.5355
60	0.6026
70	0.6738
80	0.7458
90	0.827
100	0.9052
120	0.9052

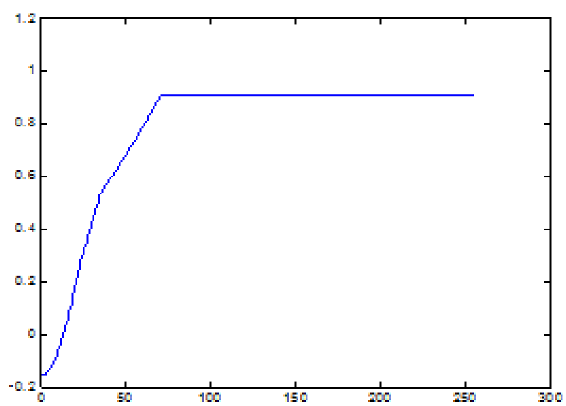


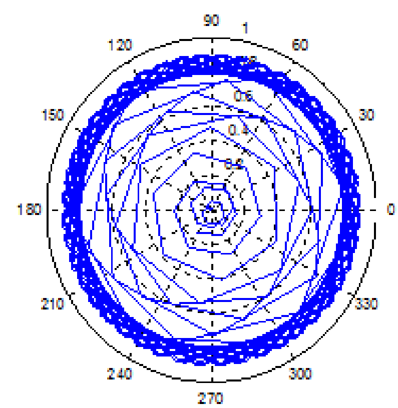
MATLAB 2010a M-CODE OUTPUT

a) Filter length N=256 and for CORDIC algorithm no of iteration i=16





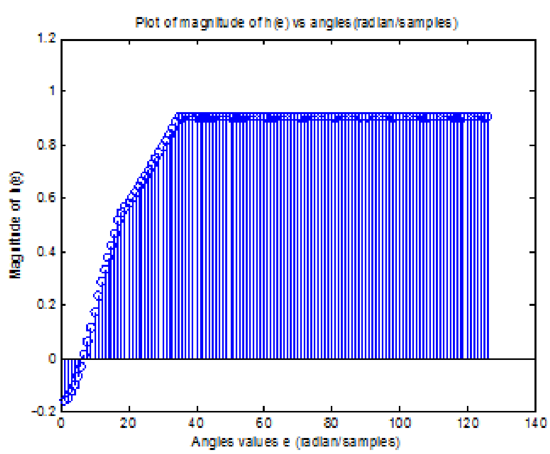


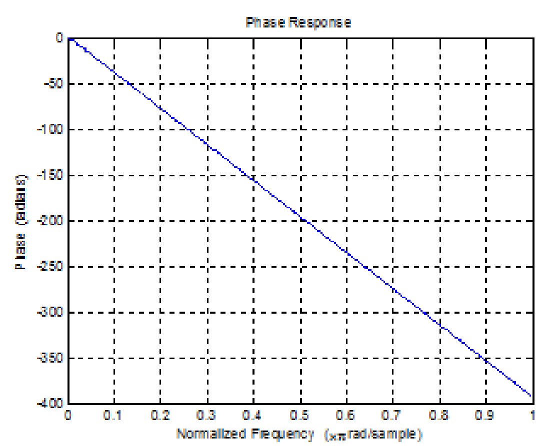


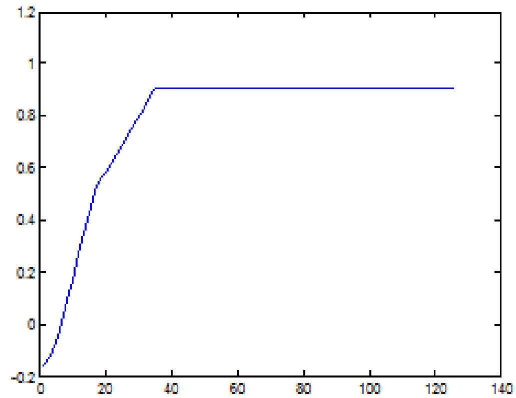
Plot of h(e)

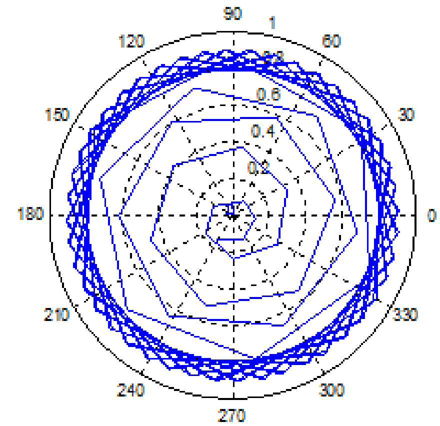
Polar Plot of h(e)

b) Filter length N=126 and for CORDIC algorithm no of iteration i=16



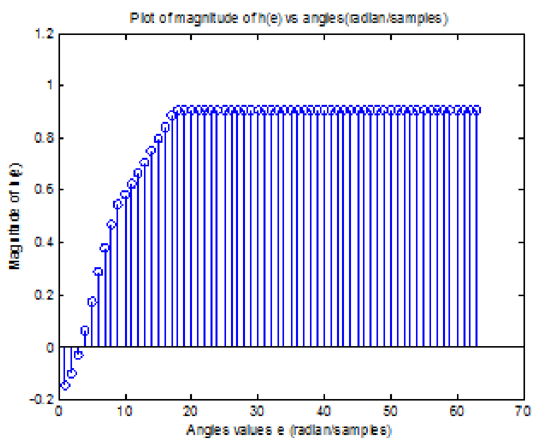


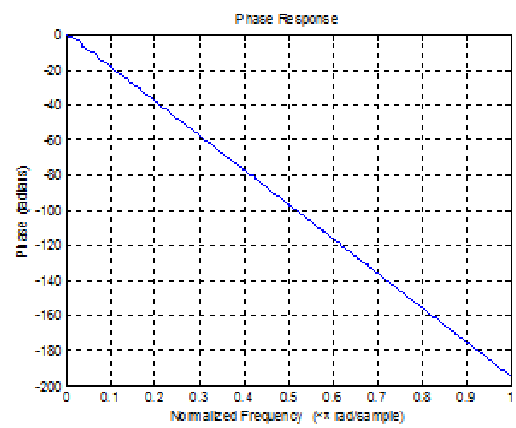


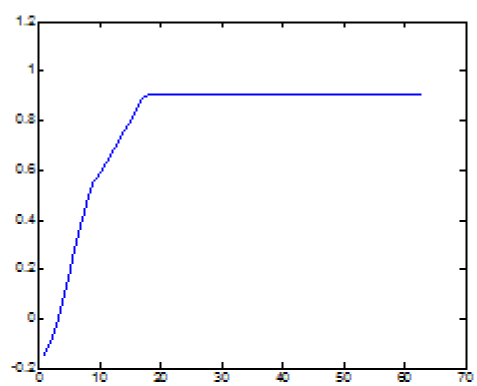


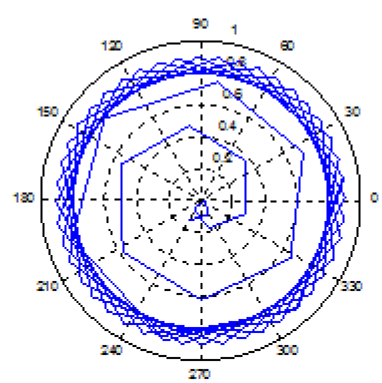
Plot of h(e) Polar Plot of h(e)

c) Filter length N=63 and for CORDIC algorithm no. of iteration i =16



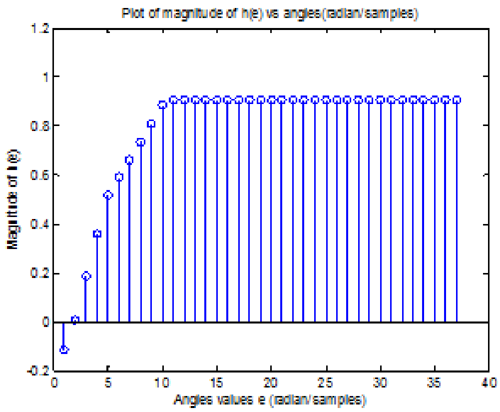


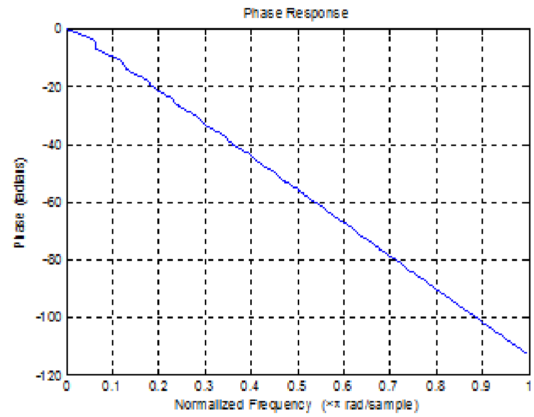


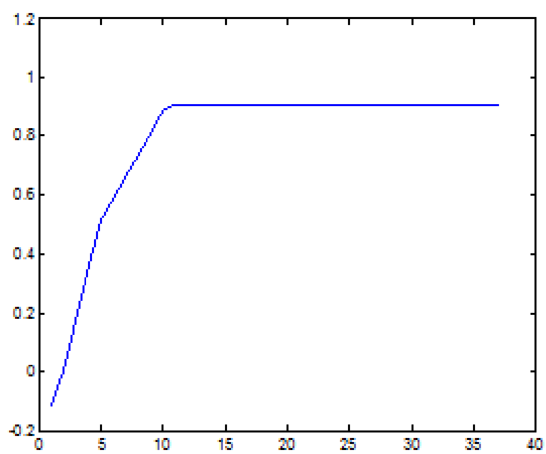


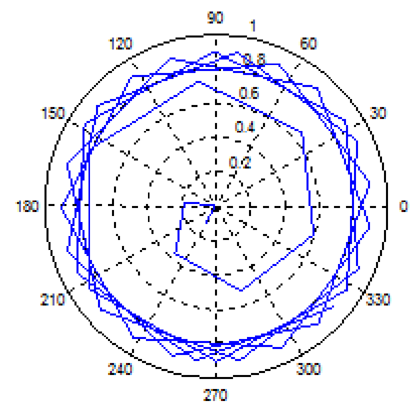
Plot of h(e) Polar Plot of h(e)

d) Filter length N=37 and for CORDIC algorithm no. of iteration i=16





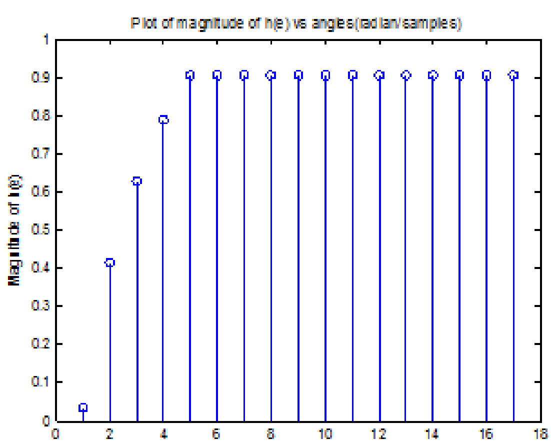


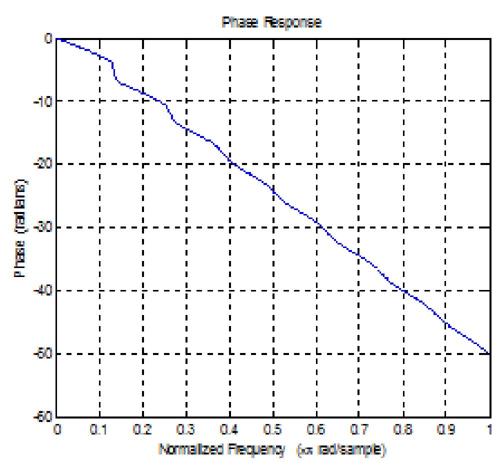


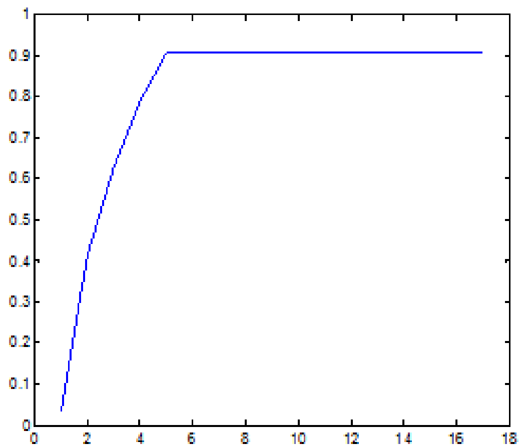
Plot of h(e)

Polar Plot of h(e)

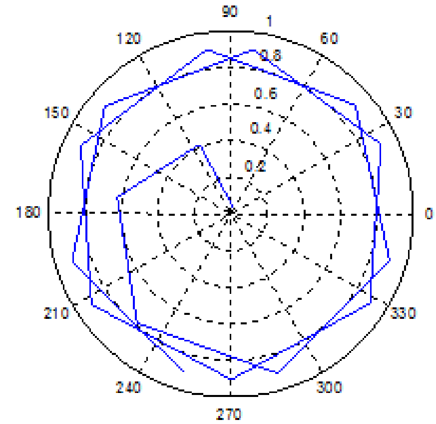
(e) Filter length N=17 and for CORDIC algorithm no. of iteration i=16





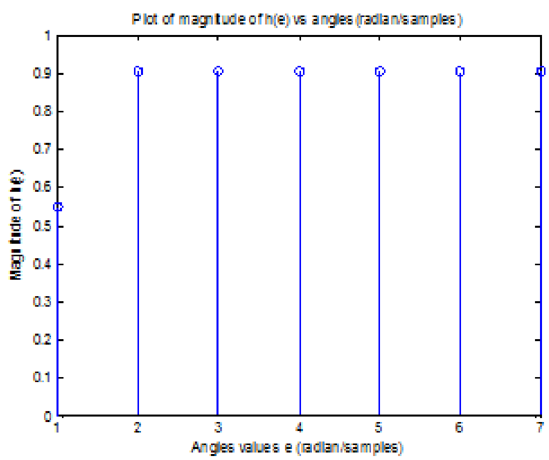


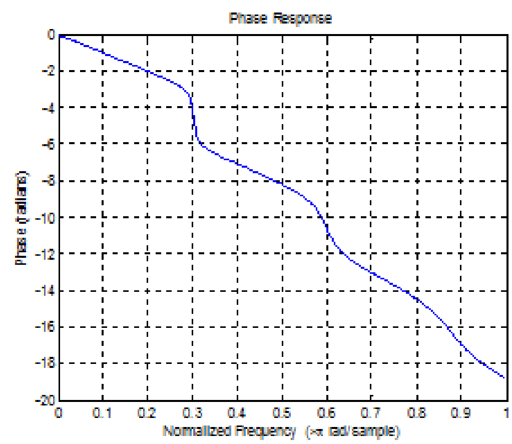
Plot of h(e)

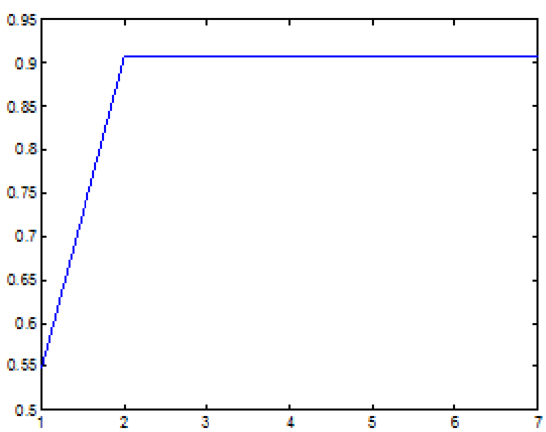


Polar Plot of h(e)

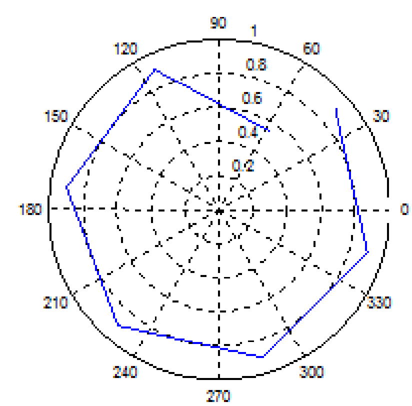
(f) Filter length N=7 and for CORDIC algorithm no. of iteration i=16







Plot of h(e)



polar plot of h(e)

We observed from the above results that the FIR filter has linear phase response. In magnitude response (from simulink) the magnitude of the output is varies from negative value to positive value and after 120° value the magnitude value is constant.

In magnitude response (from MATLAB) of the filter, the magnitude value of the output is varies from negative value to the positive value it becomes constant after at angle 75 radian/samples for filter length N=256,after 35 radian/samples for the filter length N=126,after 20 radian/samples for filter length N=63, after 10 radian/samples for filter length N=37,after 5 radian/samples for the filter length N=17 and after 2 radian/samples for the filter length N=7.It means the cutoff point at the angle(radian/samples) e=2 for the filter length N=7,e=5 for the filter length N=17, e=10 for the filter length N= 37, e=20 for the filter length N=63 , e=35 for the filter length N=126 and e=75 for the filter length N=256 respectively, those are very close ideal characteristics of the high pass filter. There are some distortion is observed in the phase response characteristics of the filter when we take the filter length N= 17and 7.When we are decreased the filter length the cutoff point also decreased but some distortion are occurred in phase response characteristics. Magnitude plot of h versus samples value and the polar plot of the h are denser as filter length increases.

Name ▲	Value	Min	Max
N	7	7	7
cos	[0.6233,-0.1716,-0.171...	-0.1716	0.6233
cos1	[-0.1716,-0.1716,-0.17...	-0.1716	-0.1716
cos2	[-0.1716,-0.1716,-0.17...	-0.1716	-0.1716
d	<1x16double>	1	1
e	7	7	7
h	[0.5484,0.9061,0.9061,...	0.5484	0.9061
i	16	16	16
sin	[0.7816,0.9849,0.9849,...	0.7816	0.9849
sin1	[0.9849,0.9849,0.9849,...	0.9849	0.9849
sin2	[0.9849,0.9849,0.9849,...	0.9849	0.9849
w	[2.6928,5.3856,8.0784,...	2.6928	18.8496
x	<1x17double>	-0.1716	0.6071
y	<1x17double>	0	0.9960
z	<1x17double>	17.1063	18.8496

Name ▲	Value	Min	Max
N	17	17	17
cos	<1x17double>	-0.1716	0.9322
cos1	<1x17double>	-0.1716	0.7388
cos2	<1x17double>	-0.1716	0.4456
d	<1x16double>	1	1
e	17	17	17
h	<1x17double>	0.0346	0.9061
i	16	16	16
sin	<1x17double>	0.3612	0.9955
sin1	<1x17double>	0.6735	0.9955
sin2	<1x17double>	0.8949	0.9849
w	<1x17double>	1.1088	18.8496
x	<1x17double>	-0.1716	0.6071
y	<1x17double>	0	0.9960
z	<1x17double>	17.1063	18.8496

Name ▲	Value	Min	Max
N	37	37	37
cos	<1x37double>	-0.1716	0.9854
cos1	<1x37double>	-0.1716	0.9426
cos2	<1x37double>	-0.1716	0.8728
d	<1x16double>	1	1
e	37	37	37
h	<1x37double>	-0.1166	0.9061
i	16	16	16
sin	<1x37double>	0.1689	0.9988
sin1	<1x37double>	0.3331	0.9917
sin2	<1x37double>	0.4876	0.9988
w	<1x37double>	0.5094	18.8496
x	<1x17double>	-0.1716	0.6071
y	<1x17double>	0	0.9960
z	<1x17double>	17.1063	18.8496

Name ▲	Value	Min	Max
N	63	63	63
cos	<1x63double>	-0.1716	0.9948
cos1	<1x63double>	-0.1716	0.9799
cos2	<1x63double>	-0.1716	0.9553
d	<1x16double>	1	1
e	63	63	63
h	<1x63double>	-0.1447	0.9061
i	16	16	16
sin	<1x63double>	0.0996	0.9994
sin1	<1x63double>	0.1981	0.9994
sin2	<1x63double>	0.2947	0.9970
w	<1x63double>	0.2992	18.8496
x	<1x17double>	-0.1716	0.6071
y	<1x17double>	0	0.9960
z	<1x17double>	17.1063	18.8496

Name	Value	Min	Max
N	126	126	126
cos	<1x126 double>	-0.1716	0.9985
cos1	<1x126 double>	-0.1716	0.9948
cos2	<1x126 double>	-0.1716	0.9886
d	<1x16 double>	1	1
e	126	126	126
h	<1x126 double>	-0.1560	0.9061
i	16	16	16
sin	<1x126 double>	0.0499	0.9994
sin1	<1x126 double>	0.0996	0.9994
sin2	<1x126 double>	0.1490	0.9970
w	<1x126 double>	0.1496	18.8496
x	<1x17 double>	-0.1716	0.6071
y	<1x17 double>	0	0.9960
z	<1x17 double>	17.1063	18.8496

Name	Value	Min	Max
N	256	256	256
cos	<1x256 double>	-0.1716	0.9994
cos1	<1x256 double>	-0.1716	0.9985
cos2	<1x256 double>	-0.1716	0.9970
d	<1x16 double>	1	1
e	256	256	256
h	<1x256 double>	-0.1589	0.9061
i	16	16	16
sin	<1x256 double>	0.0246	0.9997
sin1	<1x256 double>	0.0491	0.9997
sin2	<1x256 double>	0.0735	0.9994
w	<1x256 double>	0.0736	18.8496
x	<1x17 double>	-0.1716	0.6071
y	<1x17 double>	0	0.9960
z	<1x17 double>	17.1063	18.8496

REFERENCES

[1] J.E.Volder, "The CORDIC trigonometric computing technique," IRE Transactions on Electronic Computers, vol.8, no.3 pp.330-334, 1959.

[2] J.S.Walther, "A unified algorithm for elementary functions", in proceedings of AFIPS Spring Joint Computer Conference, pp.379-385, May 1971.

[3] Y.H.HU, "CORDIC based VLSI architecture for digital signal processing," IEEE signal processing Magazine, vol.9, no.3, pp16-35, 1992.

[4] Ray Andraka, Andraka consulting group, "A Survey of CORDIC ALGORITHMS FOR FPGA based computers" ,in Proceedings of the 6th ACM/SIGDA International Symposium on Field Programmable Gate Arrays (FPGA"98),PP.191-200,Feburary 1998.

[5] J.E.Volder, "The birth of CORDIC", Journal of VLSI Signal Processing, Vol.25, no.2, PP.101- 105, 2000.

[6] J.S.Walther, "The story of Unified CORDIC", Journal of VLSI signal processing, vol.25, no.2, pp - 107-112, 2000.

[7] Pramod.K.Meher, Javier VALLS, Tso Bing Juang, K.Sridharan, Koushik Maharatna, "50 years of CORDIC Algorithms, Architectures and Applications", IEEE Transactions on Circuits and System -1: Regulars Papers, vol.56, no.9. September 2009.

[8] B.Lakshmi and A.S.Dhar, "CORDIC Architectures: A Survey", Hindawai Publishing Corporation VLSI Design Volume 2010, Article ID 794891, 19 Pages.

[9] Richa Upadhyay, Dr.Nisha Sarwade, Shrugal Varde, "Simulink Design of Pipelined CORDIC for Generation of Sine and Cosine Values", International Journal of Computing Engineering ResearchVol3 Issue.3, March 2013.

[10] Nutan Das , Swarnaprabha Jena, Siba Kumar Panda, " FPGA implementation of Angle Generator for CORDIC Based High pass Filter Design", IOSR Journal of Electronics and Communication Engineering, pp 01-11,2016