FPGA implementation of Angle Generator for CORDIC Based High pass FIR Filter Design

Nutan Das¹, Swarnaprabha Jena², Siba Kumar Panda²

¹ M Tech Scholar, VLSI Design, Centurion University of Technology & Management, BBSR, Odisha
² Assistant Professor, Department of ECE, Centurion University of Technology & Management, BBSR, Odisha

Abstract: The design of high speed VLSI architecture introduced a large number of algorithms for real-time Digital Signal Processing (DSP) and in the VLSI Technology, the realizations of this DSP algorithms have been directed in many of the computations in Signal Processing and wireless communication applications such as trigonometric and complex functions. This paper proposed a hardware efficient CORDIC based parallel architecture for the calculation of cosine and sine functions and High Pass FIR (Finite Impulse Response) Filter design using Very High Speed Integrated Circuit Hardware Description Language (VHDL). The generation of cosine sine functions of CORDIC algorithm have synthesized, simulated and tested it on Xilinx FPGA SPARTAN 3E kit. The code have synthesized using Xilinx ISim 14.4 simulator software. This paper also described the angle to 15bit binary conversion, analysis of all output values and the power utilization summary. Finally, magnitude plot of HighPass FIR Filter is plotted and analysed.

Keywords: CORDIC, FPGA, FIR FILTER, SIGNAL PROCESSING, VLSI SIGNAL PROCESSING, VHDL

I. Introduction

In advanced VLSI technology have stimulated a great interest in developing special purpose processor architecture arrays to facilitate real-time signal processing. The reductions in the hardware cost, less area, high speed and less power consumption are motivated the development of various architectures to design filters in Digital Signal Processing. Generally, DSP algorithms use the calculation of elementary functions which require high computational power.

Now, Field-Programmable Gate Array (FPGA) is introduced hardware technology for DSP systems offer the capability to develop the most suitable architecture for the computational, memory and power requirements of the DSP applications. For FPGA implementation the DSP system produced highly parallel, pipelined and hybrid architecture. These Filters are used to demonstrate the mapping, introduced retiming and the low power optimizations are demonstrated by using a FFT-based applications and development.

In the CORDIC (Coordinate Rotation Digital Computer), J.E. Volder [1], Rotation is a 2-D (dimensional) vector and a target angle is divided into desired rotation angle into the weighted sum of a set of predefined elementary rotation angles in which through each rotation can be realized with shift and add/sub operations. Also, the CORDIC algorithm is again extended [2] to proposed a unified algorithm for computation of rotation in and hyperbolic coordinate systems, circular, linear, embedding coordinate system. These functions are implemented for many applications like digital signal processing [3].

CORDIC used mainly simple shift and add operations[4], that performed several computational tasks are calculation of trigonometric functions (cosine, sine), hyperbolic functions (tanh, hcos) and logarithmic functions, real and complex functions, and simple arithmetic functions are multiplications, division, square-root functions. Now CORDIC Algorithms are used in the area of signal and image processing and communication systems, robotics and 3-D (Dimensional) graphics, Fast Fourier Transforms (FFT), Filters, Computer Graphics[5] and Robotics [6], CORDIC offers high computational applications: computer graphics, in which a combination of scaling and rotations are required for real time world. For fast computation programming CORDIC is again used in the field of Robotics.

II. Background And Literature Review

Amritakar Mandal et al. [7] described the design of pipelined architecture for the computation of Sine and Cosine values and it is based on application of a specific CORDIC processor. The design of CORDIC is mainly based on the circular rotation mode gives a high system throughput by reducing latency in each individual pipelined stage.

Lakshmi.B et al. [8] explained a low latency field programmable gate array implementation of an unfolded architecture for the implementation of rotational CORDIC algorithm. Here the computational hardware was highly suitable for the implementation of customized hardware in portable devices and in which a large parallelism and low clock rate were utilized for low power consumption.
Siba Ku panda et al. [9, 10] explained an innovative idea of designing multiplier circuit for various VLSI signal processing applications.

Ray Andraka [11] has summarized that by using CORDIC Algorithm some functions like sine, cosine, tangent, circular, linear, hyperbolic functions, Cartesian to polar transformations and inverse functions was implemented. For FPGA Implementation next time this paper also described as bit serial, iterative and online CORDIC architecture design.

Velmurugan [12] explained a proposed low power analog and mixed mode implementation of particle filter for target tracking. Based on CORDIC Algorithm[13] the particle filter algorithm was used to exploits addition, multiplication, Gaussian and Arctan calculations.

X. Hu et al. [14] has described a proposed unified CORDIC algorithm based on the parallel and pipelined CORDIC processor. J. Sudhaa etal.[15] has proposed the potential application of the proposed 2D Gaussian function includes image enhancement, smoothing, edge detection, filtering etc and to gain real time performance, the architectures developed exploits high degrees of pipelining and parallel processing.

N. Takagi, etal [16] described a proposed Double rotation and correcting rotation methods to implement constant scale factor CORDIC in which result was 50% increasing in number of iterations. For that this increase in latency is reduced by proposed branching algorithm.

J. Duprat et al. [17] described additional CORDIC module to perform rotations in both directions, if the direction cannot be determined using intermediate results. The main demerits of branching method is the necessity of performing two conventional CORDIC iterations in parallel, which consumes more silicon area than the conventional methods. Also, this method gives a faster implementation than [16].

M. Chakraborty, etal.[18] explained a class of pipelined CORDIC architectures for the LMS-based transversal adaptive filter. Introducing an alternate formulation of the LMS algorithm, obtained by expressing the mean square error as a convex function of a set of angle variables and monotonically related to the filter tap weights. Also proposed architectures employed a micro-level pipelining and are adjustable to strike tradeoffs between throughput efficiency vis-a-vis hardware complexity.

Shanmuga Kumar M. etal. [19] summarized a proposed the high precision and hardware efficient CORDIC structure for handheld scientific calculator. Tsao Y C, etal.[20] presented the realization of area efficient architectures using Distributed Arithmetic (DA) for implementation of Finite Impulse Response (FIR) filter. Also described the performance of the bit-serial and bit parallel DA along with pipelining architecture and different quantized versions are analyzed for FIR filter Design.

Shrikant Patel etal [21] presented the implementation of Modified Distributed Arithmetic, and introduce it into the FIR filters design, and then presents a 31-order FIR low-pass filter using Modified Distributed Arithmetic and save considerable MAC blocks to decrease the circuit scale.

M.Chakrapani etal [22] explained the mode of operation of CORDIC algorithm and Control CORDIC Angle correction and Quadrant correction. Latency can be further reduced by another CORDIC algorithm techniques. By angle selection scheme the Overshoot problem in original CORDIC has been overcome and complexity of hardware also increases for the increasing of number of bits.

The contribution of this paper is based on a new and modified architecture in which it needs less area for implementation and less delay output. The need of hardware efficient algorithms in the advanced technologies For the fast signal processing requirements. So, the current trend is back toward hardware efficient algorithms, many of those applications require elementary calculations like sin and cosine function. So, in this paper i am presenting a angle generator using CORDIC algorithm and its performance reports. The parallel architecture takes the angle as input and gives both sine and cosine for the given input in predetermined number of micro rotations. These micro rotations are decided by the accuracy demanded by the application. Here our objective is design to plot the magnitude curve using the frequency response equation of a FIR Filter using CORDIC algorithm in VHDL.

Section I gives brief introduction, section II defines the Background and Literature Review. The Basic CORDIC algorithm and architecture are briefly studied in section III. The FPGA Implementation, Simulation and Synthesis results are discussed in section IV and finally conclusion is explained in section V.

### III. Basic Cordic Algorithm And Architecture

Coordinate Rotation Digital Computer (CORDIC), firstly implemented in1959 by J.E.Volder [1], is an ancient principles of two-dimensional geometry. CORDIC is a typical iterative algorithm, is used to accelerate many digital functions and applications in signal processing. This algorithm may be realized as an iterative form of add, sub and shift operations. The two basic modes in which the different functions are computing and these are a rotation mode and the vectoring mode. The algorithm can be realized as an iterative sequence of additions or subtractions and shift operations by using the two modes, which are rotated by a fixed rotation angle (micro rotations).

In Rotation mode is shown in Fig-2, we can rotate (1,0) by an angle $\varphi$ degrees to get (x, y).
FPGA implementation of Angle Generator for CORDIC Based High pass FIR Filter Design

\[ x = x \cdot \cos(\phi) \]
\[ y = y \cdot \sin(\phi) \]  

(1)

\[ x' = x \cdot \cos(\phi) - y \cdot \sin(\phi) \]
\[ y' = y \cdot \cos(\phi) + x \cdot \sin(\phi) \]  

(2)

Rearranged as the equations are

\[ x' = \cos(\phi) \cdot [x - y \cdot \tan(\phi)] \]
\[ y' = \cos(\phi) \cdot [y_i + x \cdot \tan(\phi)] \]  

(3)

Rewrite in terms of \( a_i \): (0 \( \leq \) \( i \) \( \leq \) \( n \))

\[ x_{i+1} = \cos(\phi_i) \cdot [x_i - y_i \cdot d_i \cdot \tan(\phi_i)] \]
\[ y_{i+1} = \cos(\phi_i) \cdot [y_i + x_i \cdot d_i \cdot \tan(\phi_i)] \]  

(4)

Figure- 1 (a) and (b) Vector mode by the angle \( \phi \)

We can restrict the rotation angle \( \tan(\alpha_i) \) to \( \tan(\phi_i) = \pm 2^{-i} \), then the multiply operation by tangent part is reduced to shift operations. The new iterative rotation can be defined as

\[ x_{i+1} = K_i \cdot [x_i - y_i \cdot d_i \cdot 2^{-i}] \]
\[ y_{i+1} = K_i \cdot [y_i + x_i \cdot d_i \cdot 2^{-i}] \]  

Where, \( i = 0,1,2,3,4,\ldots n \).

(5)

Figure- 2 (a) and (b) Rotation mode by the angle \( \Theta \)
FPGA implementation of Angle Generator for CORDIC Based High pass FIR Filter Design

Removing the scaling factor $K_i$ from the iterative part yields a simple shift-add algorithm for rotation CORDIC. After several iterations, the product of $K_i$ factors will converge to a constant coefficient 0.607252. The exact gain depends on the number of iterations:

$$A_n = \prod_{i=0}^{n} \sqrt{1 + 2^{-2i}} \approx 1.646762 = \frac{1}{K_i}$$

(6)

Then the angle accumulator is added to the Equation [7], below

$$\begin{align*}
\mathbf{w}_{i+1} &= z_i - d_i \cdot \tan^{-1} \left( \frac{1}{2} \cdot z_i \right) < 0 \\
\mathbf{d}_i &= \begin{cases} 
+1, & z_i \geq 0 \\
-1, & \text{otherwise}
\end{cases}
\end{align*}$$

The angle accumulator is initialized with the input rotation angle and the rotation direction at each iteration $i$ is decided by the magnitude of the residual angle in the angle accumulator. If the residual angle is positive, then

$$d_i = \begin{cases} 
+1, & z_i \geq 0 \\
-1, & \text{otherwise}
\end{cases}$$

A CORDIC has three inputs $x_0, y_0, z_0$ and depending on the inputs to the Rotation CORDIC and various results can be produced at the outputs are $X_n, Y_n, Z_n$ shown in Fig:2. After several iterations it will produce the results according to a CORDIC Block is shown in Fig-3.

$$\begin{align*}
x_n &= A_n \left( x_0 - y_0 \tan z_0 \right) \\
z_n &= 0 \\
y_n &= A_n \left( y_0 + x_0 \tan z_0 \right)
\end{align*}$$

(8)

Figure-3 A CORDIC Block

For convergence of $\phi$ to 0 choose $d_i = \text{sgn } Z_i$

If we can start with $X_0=1/k$ and $y=0$, at the end of the process, we find $X_n=\cos Z_0$ and $Y_n=\sin Z_0$ and the domain of convergence is lies between -99.7 < $Z_0$ < 99.7.

A. Using CORDIC in Vectoring mode is shown in Fig-2, we can write the expression as below:

$$\begin{align*}
x_{i+1} &= x_i - y_i \cdot d_i \cdot 2^{-i} \\
y_{i+1} &= y_i + x_i \cdot d_i \cdot 2^{-i} \\
z_{i+1} &= z_i - d_i \cdot \tan^{-1} 2^{-i}
\end{align*}$$

(9)

And we can solve as

$$\begin{align*}
x_n &= A_n \sqrt{x^2 + y^2} \\
y_n &= 0 \\
z_n &= z_0 + \tan^{-1} \left( \frac{y_0}{x_0} \right)
\end{align*}$$

(10)

For convergence of $Y_n$ to 0, choose

$$\begin{align*}
d_i &= -\text{sgn}(x_i, y_i) \\
z_n &= \tan^{-1} y_0
\end{align*}$$

If we start with $x_0=1$ and $z_0=0$, we find
FPGA implementation of Angle Generator for CORDIC Based High pass FIR Filter Design

\[ d_i = \begin{cases} +1, & y_i < 0 \\ -1, & \text{otherwise} \end{cases} \]

3.1 Required Angle
At first pre-computation of \( \tan \alpha = (2^{-1}) \) or \( \alpha = \tan^{-1}(2^{-1}) \) value then we obtained binary value of each required angle. Here we use 10 iterations, so each iteration we use the \( \alpha \) value from the TABLE-1 as shown below.

<table>
<thead>
<tr>
<th>ITERATION(i)</th>
<th>REQUIRED ANGLE ( \alpha ) ( = \tan^{-1}(2^{-1}) )</th>
<th>( \tan \alpha )</th>
<th>OBTAINED BINARY VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>45</td>
<td>0.7854</td>
<td>001000000000000000000000</td>
</tr>
<tr>
<td>1</td>
<td>26.565</td>
<td>0.4636</td>
<td>000100110110111111111111</td>
</tr>
<tr>
<td>2</td>
<td>14.036</td>
<td>0.2450</td>
<td>000010011001001001001001</td>
</tr>
<tr>
<td>3</td>
<td>7.125</td>
<td>0.1244</td>
<td>000001010000000000000000</td>
</tr>
<tr>
<td>4</td>
<td>3.5763</td>
<td>0.0624</td>
<td>000000101000000000000000</td>
</tr>
<tr>
<td>5</td>
<td>1.7899</td>
<td>0.0312</td>
<td>000000010000000000000000</td>
</tr>
<tr>
<td>6</td>
<td>0.8951</td>
<td>0.0156</td>
<td>000000001010000000000000</td>
</tr>
<tr>
<td>7</td>
<td>0.4476</td>
<td>0.0078</td>
<td>000000000101000000000000</td>
</tr>
<tr>
<td>8</td>
<td>0.2238</td>
<td>0.0039</td>
<td>000000000010100000000000</td>
</tr>
<tr>
<td>9</td>
<td>0.1119</td>
<td>0.0019</td>
<td>000000000001010000000000</td>
</tr>
</tbody>
</table>

3.2 Working Cordic Architecture
The working CORDIC parallel Architecture figure is given below. This architecture consists of Add block, add/sub block, comparator and shift blocks as shown in Fig-4. We have assumed that, if the input angular value<=45 degree, then inputs (x, y) are (1,0) or else (0,1). Then compare this angle with a reference angle (30 degree) and right bit-shifted operation is occurred depending upon the iteration step(i), x(i) and y(i) and then added or subtracted depending upon the value of comparator circuit(z0) with x(i) and y(i) respectively to generate x(i+1) and y(i+1). Again we repeat the 10 iterations to getting the desire output values.

![Figure-4 Working Parallel CORDIC Architecture](image)
3.3 Cordic Based High Pass FIR Filter

The Block diagram of CORDIC based high pass FIR filter is shown in FIG-5. Here the main blocks are Bit-shift, Add/Sub and CORDIC and the given inputs to the HP FIR filter design is in a 15-bit binary format. The angular inputs w, 2w, 3w are provided through the CORDIC blocks and generate the Cosine angle and finally using the Add/Sub blocks produce the desire output. Here we have chosen an arbitrary frequency response equation of an High Pass FIR filter is given in equation (11).

\[ H(e^{jw}) = 0.75 - 0.45 \cos w - 0.31 \cos 2w - 0.15 \cos 3w \] (11)

CORDIC based High Pass FIR Filter

IV. FPGA Implementation And Simulation Result

In this paper, the FPGA implementation is done for generating the sine and cosine values in binary format of given angle (30 degree) and taking the input (1,0) of X and Y value in 2bit form using CORDIC algorithm as shown in Fig-6 and RTL schematic is also shown in Fig-7. Both 15 bit CORDIC architecture and CORDIC based FIR Filter module have implemented by using Xilinx ISE Design 14.4 as shown in Fig-8 and Fig-9, described in VHDL language and verified and synthesized the functionality using Xilinx ISim simulator as shown in Fig-10 and Fig-11. The top-level RTL schematic figures are shown below.
FPGA implementation of Angle Generator for CORDIC Based High pass FIR Filter Design

Figure-7 RTL Schematic of Parallel CORDIC Top level 2bit Architecture

Figure-8 Top level RTL CORDIC Architecture

Figure-9 Top level RTL CORDIC FIR Filter
FPGA implementation of Angle Generator for CORDIC Based High pass FIR Filter Design

4.1 Output Simulation Result
The output simulation result for parallel CORDIC Architecture (CORDIC_NEW) is shown in Fig-12 and also output simulation result for FIR Filter(30 degree) and (25 degree) are shown in Fig-13 and Fig-14.
FPGA implementation of Angle Generator for CORDIC Based High pass FIR Filter Design

Figure-12 Simulation Result For Parallel CORDIC Architecture (CORDIC_NEW)

Figure-13 Simulation Result For (30 Degree) FIR Filter

Figure-14 Simulation Result For (25 Degree) FIR Filter

4.2 Magnitude plot of High Pass FIR Filter

The Magnitude plot of high pass filter is shown in Fig-15, described the accuracy between the educational value and output calculated values of generated angles (10 degree to 85 degree) from CORDIC based HP FIR Filter Design circuit using VHDL language. The plot clearly reflects the accuracy of the output which is generated from the CORDIC based FIR Filter frequency response equation (11).
4.3 Power Summery
The Power Summery of Parallel CORDIC Algorithm is given in Fig-16. The total Power utilization is 0.031 W.

<table>
<thead>
<tr>
<th>Optimization</th>
<th>None</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>0.030</td>
</tr>
<tr>
<td>Quiescent (W)</td>
<td>0.002</td>
</tr>
<tr>
<td>Dynamic (W)</td>
<td></td>
</tr>
<tr>
<td>Total (W)</td>
<td>0.031</td>
</tr>
</tbody>
</table>

Figure-16 Power Summery of Parallel CORDIC Algorithm

V. Conclusion
We have successfully simulated and implemented the CORDIC algorithm in FPGA for calculating the Cosine and Sine of an angle in 15 bit binary format on Xilinx Spartan3E (XC3S100Ecp132-5) device using ISE Design Suite 14.4,1Sim Simulator of VHDL language. Also we have plotted the magnitude curve of a High Pass FIR Filter and this error less curve was generated for output accuracy between equational calculation output values and generated output of CORDIC_FIL values. In future CORDIC algorithm can be further extended for clock-pipellined architecture in VERILOG language and to calculate more complex and higher order problems and digital filter design in DSP and VLSI domain with more accuracy, high speed and low cost. The number of Input Output Blocks utilization in both cases (CORDIC_NEW, CORDIC_FIR) are 90% and 68% respectfully and the total power utilization is 0.030.

References
FPGA implementation of Angle Generator for CORDIC Based High pass FIR Filter Design


