DOI: https://doi.org/10.24113/ijoscience.v3i5.53

High Speed Area Efficient VLSI Architecture for DCT and DHT Algorithm

Sonam Nema
M. Tech Scholar
Department of Electronics and Communication Engineering
SAM College of Engineering & Technology
Bhopal, M.P., India
nema.sonam@gmail.com

Adesh Gour
Professor
Department of Electronics and Communication Engineering
SAM College of Engineering & Technology
Bhopal, M.P., India
adesh_gour11@rediffmail.com

Abstract - **Low-power layout is one of the most vital challenges to maximize battery life in portable devices and to save the energy during simulation operation. Image and video compressor is widely used in Discrete Cosine Transform (DCT). Many types of techniques are used in design discrete cosine transform (DCT). Multiplier and adder are two main components in design to DCT, Loeffer (1989) have developed a new architecture DCT, it consists of 11 multiplications and 29 additions. By now a day we required low chip area and fast speed algorithm, but the multiplier consumed large area compared to adder. We are designed to multiplier less CORDIC (Coordinate Rotation Digital Computer) algorithm based on DCT. CORDIC is a main component of shift and add for rotation vector and plan which is usually used for calculation of trigonometric functions. CORDIC algorithm is efficient area and delay compared to existing algorithms. All design are implementation Xilinx 14.1i and verified the result.**

Keywords:-Discrete Cosine Transform, Discrete Fourier Transform, Coordinate Rotation Digital Computer

I. INTRODUCTION

For a long term the field of virtual signal Processing has been dominated by Microprocessors. That is especially due to the fact they provide designers with the blessings of unmarried cycle multiply-gather education in addition to unique addressing modes. Even though these processors are reasonably-priced and bendy they are exceptionally slow on the subject of performing sure disturbing signal processing duties e.g. photograph Compression, virtual verbal exchange and Video Processing. Of past due, rapid advancements were made inside the field of VLSI and IC layout. As a result unique purpose processors with custom-architectures have come up. Higher speeds can be performed by way of those customized hardware answers at competitive prices. to feature to this, numerous easy and hardware-efficient algorithms exist which map nicely onto those chips and can be used to decorate velocity and flexibility at the same time as appearing the favored signal processing obligations [1][2][3]. One such easy and hardware-efficient algorithm is CORDIC, an acronym for Coordinate Rotation virtual computer, proposed with the aid of Jack E Volder [4. CORDIC uses only Shift-and Add mathematics with table look-as much as implement special capabilities. With the aid of making moderate modifications to the preliminary situations and the LUT values, it is able to be used to successfully enforce Trigonometric, Hyperbolic, Exponential features, and Coordinate changes and so on. Since it makes use of simplest shift-upload arithmetic, VLSI implementation of such a set of rules is easily workable. DCT set of rules has numerous programs and is extensively used for photograph compression. Imposing DCT the use of CORDIC algorithm reduces the range of computations at some stage in processing, increases the accuracy of reconstruction of the picture, and decreases the chip vicinity of implementation of a processor constructed for this reason. This reduces the overall energy consumption. FPGA provides the hardware environment wherein dedicated processors can be tested for their capability. They perform numerous high-speed operations that cannot be realized via a simple microprocessor. The number one benefit that FPGA gives is On-web site 2programmability. Therefore, it bureaucracy the suitable platform to put into effect and check the capability of a committed processor designed using CORDIC algorithm [5].

The advances in IC generation have outstanding pastimes in developing special reason, parallel processor arrays such systolic arrays had been extensively used. The basic arithmetic computation of those parallel arrays has often been implemented with a MAC, due to the fact those operations arise in DSP programs. The reduction in hardware cost also motivated the development of sophisticated signal processing algorithms which want the assessment of capabilities together with trigonometric and logarithmic capabilities, which cannot be evaluated effectively with MAC based totally arithmetic gadgets. So whilst sign processing algorithms contain those fundamental functions, it's far sure to observe significant overall performance failure. So an arithmetic computing set of rules called CORDIC (Coordinate Rotation virtual pc) has acquired tremendous interest, because it gives an iterative system to correctly calculate each of these standard capabilities. Specially, all the evaluation duties in CORDIC are formulated as a rotation of vectors in various Coordinate systems. by means of varying a few parameters, the equal CORDIC processor is capable of iteratively calculates those standard features the use of the identical hardware inside the identical quantity of implementation of pipelined VLSI array processors.

II. Literature Review

CORDIC or Coordinate Rotation virtual laptop is easy and hardware-green algorithm for the implementation of diverse simple, especially trigonometric, functions. Instead of using Calculus based totally methods such as polynomial or rational useful approximation, it makes use of easy shift, upload, subtract and table look-up operations to reap this goal. The CORDIC algorithm become first proposed through Jack E Volder in 1959. It's also applied in both Rotation mode and Vectoring mode. In either mode, the algorithm is rotation of a perspective vector by way of a exact perspective but in variable guidelines. This constant rotation in variable direction is carried out thru an iterative series of addition/subtraction followed through bit-shift operation. The final result is acquired via as it should be scaling the end result obtained after successive iterations. Attributable to its simplicity the CORDIC set of rules can be without difficulty implemented on a VLSI gadget. Hardware requirement and cost of CORDIC processor is less as most effective shift registers, adders and look-up table (ROM) are required range of gates required in hardware implementation, inclusive of on an FPGA, is minimal as hardware complexity is greatly reduced as compared to different processors together with DSP multipliers it's miles tremendously easy in layout No multiplication and only addition, subtraction and bit-moving operation ensures simple VLSI implementation. Postpone worried for the duration of processing is comparable to that in the implementation of a department or rectangular-rooting operation. Both if there is a lack of a hardware multiplier (e.g. uC, uP) or there is a need to optimize the variety of logic gates (e.g. FPGA) CORDIC is the preferred choice.

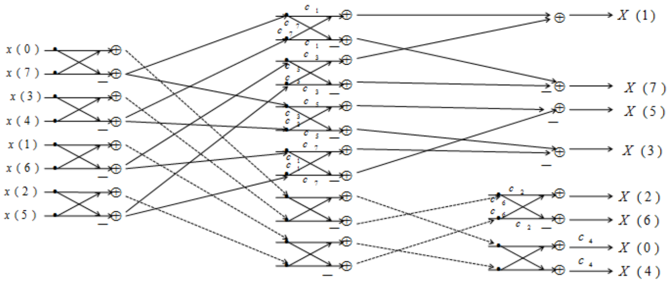


Figure 1: 8-point Discrete Cosine Transform

III. Discrete Cosine Transform

Discrete Cosine Transformation (DCT) is the most widely used transformation algorithm. DCT, first proposed by way of Ahmed [9] et al, 1974, has got greater importance in current years, in particular in the fields of photograph Compression and Video Compression. This chapter makes a speciality of green hardware implementation of DCT by way of reducing the variety of computations, enhancing the accuracy of reconstruction of the unique information, and lowering chip place. due to which the electricity consumption additionally decreases. DCT also improves velocity, compared to different trendy picture compression algorithms like JPEG.

DCT output

$$F(0)=0.5(f(0)+f(1)+f(2)+f(3)+f(4)+f(5)+f(6)+f(7))\cos\frac{\pi}{4}$$

$$F(1)=0.5[\{(f(0)-f(7)\}\cos\frac{\pi}{16}+\{f(1)-f(6)\}\cos\frac{3\pi}{16}+\{f(2)-f(5)\}\cos\frac{5\pi}{16}+\{f(3)+f(4)\}\cos\frac{7\pi}{16}]$$

$$F(2)=0.5[\{(f(0)-f(3)-f(4)+f(7)\}\cos\frac{2\pi}{16}+\{f(1)-f(2)-f(5)+f(6)\}\cos\frac{6\pi}{16}]$$

$$F(3)=0.5[\{(f(0)-f(7)\}\cos\frac{3\pi}{16}+\{f(6)-f(1)\}\cos\frac{7\pi}{16}+\{f(5)-f(2)\}\cos\frac{\pi}{16}+\{f(4)+f(3)\}\cos\frac{5\pi}{16}]$$

$$F(4)=0.5[(f(0)+f(3)+f(4)+f(7)-f(1)-f(2)-f(5)-f(6))\cos\frac{\pi}{4}]$$

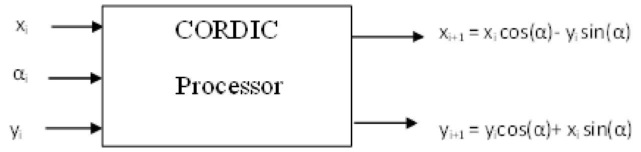
$$F(5)=0.5[\{(f(0)-f(7)\}\cos\frac{5\pi}{16}+\{f(6)-f(1)\}\cos\frac{\pi}{16}+\{f(2)-f(5)\}\cos\frac{7\pi}{16}+\{f(3)+f(4)\}\cos\frac{3\pi}{16}]$$

$$F(6)=0.5[\{(f(0)-f(3)-f(4)+f(7)\}\cos\frac{6\pi}{16}-\{f(1)-f(2)-f(5)+f(6)\}\cos\frac{2\pi}{16}]$$

$$F(7) = 0.5[\{(f(0) - f(7)\}\cos\frac{7\pi}{16} + \{f(6) - f(1)\}\cos\frac{5\pi}{16} + \{f(2) - f(5)\}\cos\frac{3\pi}{16} + \{f(4) + f(3)\}\cos\frac{\pi}{16}]$$

IV. CORDIC ALGORITHM

The simple form of CORDIC is based on observation that if a unit length vector with at (x,y)=(1,0) is rotated by an angle α degrees, its new end point will be at $(x, y) = (\sin\alpha, \cos\alpha)$ thus coordinates can be computed by finding the coordinates of new end point of the vector after rotation by an angle α. Rotation of any (x , y) vector:


x_i
α_i
y_i
CORDIC
Processor
$x_{i+1} = x_i\cos(\alpha) - y_i\sin(\alpha)$
$y_{i+1} = y_i\cos(\alpha) + x_i\sin(\alpha)$

Basic equation of CORDIC algorithm

$$x_{i+1} = x_i\cos(\alpha) - y_i\sin(\alpha) \qquad (1)$$

$$y_{i+1} = y_i\cos(\alpha) + x_i\sin(\alpha) \qquad (2)$$

Rearrange equations

$$x_{i+1} = \cos(\alpha)\,[x - y\tan\alpha] \qquad (3)$$

$$y_{i+1} = \cos(\alpha)[y + x\tan\alpha] \qquad (4)$$

$$\tan\alpha = \frac{\sin\alpha}{\cos\alpha}$$

Rotations and pseudo rotation:

Consider the vector $OE^{(i)}$ in figure having one end point at O and other $E^{(i)}$ with _ Co-ordinates $(x_i,\ y_i)$. If $OE^{(i)}$ is rotated the origin by angle α^i as shown in figure thaw end point $OE^{(I+1)}$ will have coordinates $(x_{i+1},\ y_{i+1})$ satisfying.

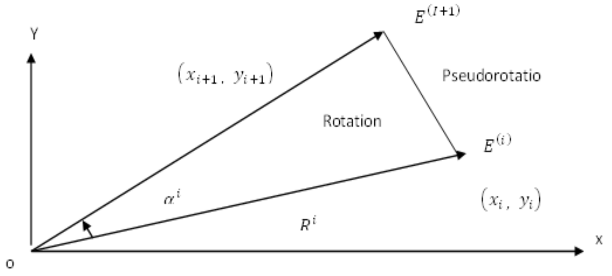


Figure 2: A pseudo rotation steps in CORDIC

In order to clearly illustrate the features and advantages of the algorithm, the VLSI architecture for a DHT of length N=8 is presented in figure 2.2. it can be seen that the architecture is highly parallel and has a modular and regular structure being formed of only a few sharing block: addition, sub tractor and multiplier. Each multiplier is shared by two inputs with a constant.

First two stages do not include any multiplication. Remaining terms are multiplied by the first coefficient. In the next stage again two new coefficients are introduced which is multiplied by the lower half of the third stage. In each stage multiplying of coefficients stage precedes its summing stage. After coefficient multiplication it is preceded by its summing stage to form the common terms used in the final stage. Last stage includes only summing of terms. Finally we get the transformed data sequence in order and do not need any permutation.

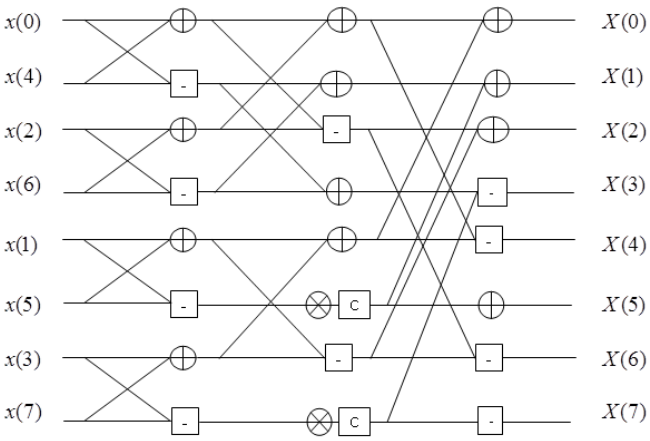


Figure 3: 8-point Discrete Hartley Transform

- Mathematical calculation for N=8

$$X(0)=[(x(0)+x(4))+(x(2)+x(6))]+[(x(1)+x(5)) \\ +(x(3)+x(7))]$$

$$X(2)=[(x(0)+x(4))-(x(2)+x(6))]+[(x(1)+x(5)) \\ -(x(3)+x(7))]$$

$$X(4)=[(x(0)+x(4))+(x(2)+x(6))]-[(x(1)+x(5)) \\ +(x(3)+x(7))]$$

$$X(6)=[(x(0)+x(4))-(x(2)+x(6))]-[(x(1)+x(5)) \\ -(x(3)+x(7))]$$

$$X(1)=[(x(0)-x(4))+(x(2)-x(6))]+c(x(1)-x(5))$$

$$X(3)=[(x(0)-x(4))-(x(2)-x(6))]+c(x(3)-x(7))$$

$$X(5)=[(x(0)-x(4))+(x(2)-x(6))]-c(x(1)-x(5))$$

$$X(3)=[(x(0)-x(4))-(x(2)-x(6))]-c(x(3)-x(7))$$

$$with \quad c=\sqrt{2}$$

Where c is the multiplier

V. Simulation Result

All of the designing and experiment concerning set of rules that we've noted in this paper is being advanced on Xilinx 14.1i updated version. Xilinx 9.2i has couple of the hanging capabilities including low memory requirement, rapid debugging, and low price. The present day release of ISETM (incorporated software environment) layout tool provides the low reminiscence requirement approximate 27 percentage low. ISE 14.1i that offers advanced gear like smart bring together technology with higher utilization in their computing hardware gives faster timing closure and higher fine of results for a better time to designing answer.

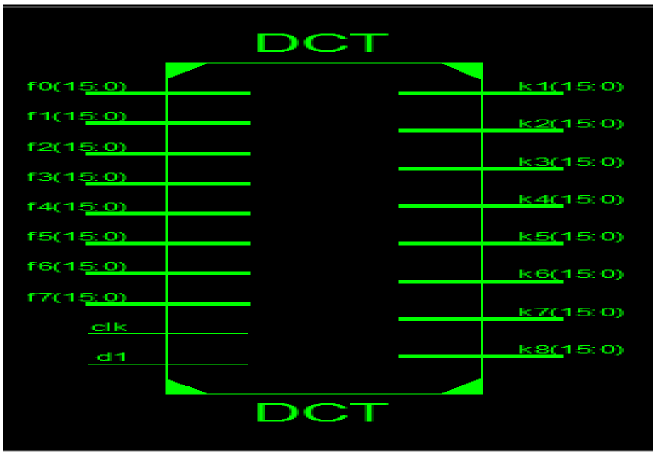
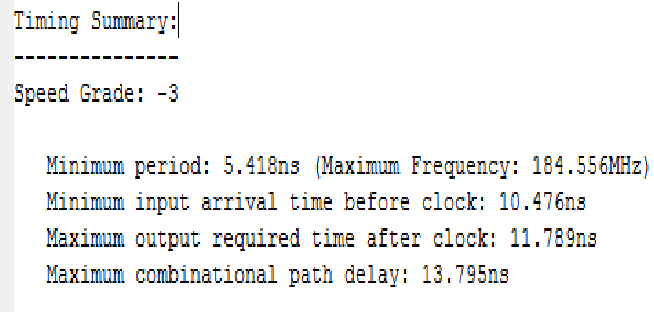


Figure 4: Resister transfer Level (RTL) View of 8-point DCT

Table 1: Device Utilization

	Mamatha I et al.	Proposed
Number of Register	1102	342
Numbe of Slice LUTs	2541	1303
LUT-FF Pairs	958	236
Number of DSP 48Es	72	-
Number of IOBs	1588	258
Maximum Frequency Operation	224.9MHz	184.556 MHz



```
Timing Summary:
---------------
Speed Grade: -3

   Minimum period: 5.418ns (Maximum Frequency: 184.556MHz)
   Minimum input arrival time before clock: 10.476ns
   Maximum output required time after clock: 11.789ns
   Maximum combinational path delay: 13.795ns
```

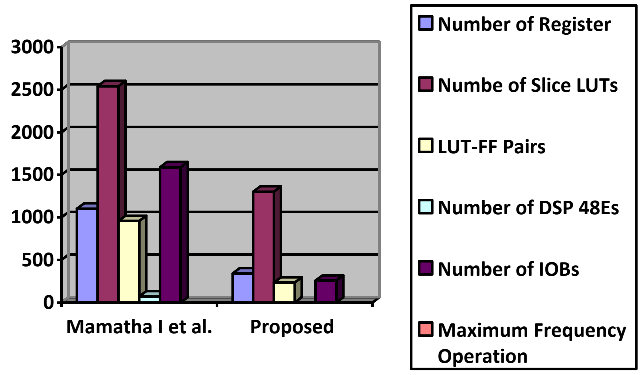
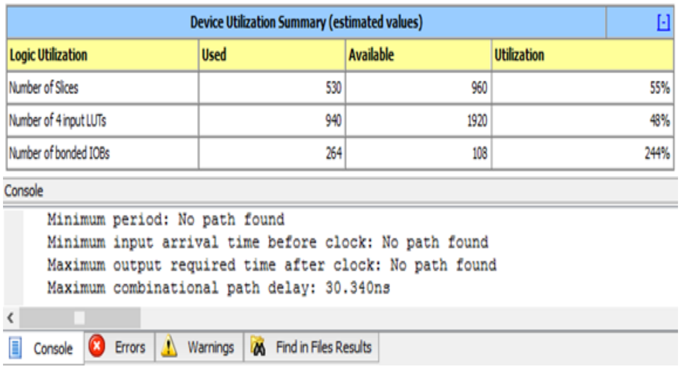


Figure 5: Bar Graph of the 8-point DCT



Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	530	960	55%
Number of 4 input LUTs	940	1920	48%
Number of bonded IOBs	264	108	244%

Console

```
Minimum period: No path found
Minimum input arrival time before clock: No path found
Maximum output required time after clock: No path found
Maximum combinational path delay: 30.340ns
```

Figure 6: Synthesis Report of Proposed Discrete Hartley Transform

Table II: Simulation Result for Proposed Structure with Different Device Family

Parameter	Doru et al. [1]	Modified DHT	Proposed DHT
No. of Slices	698	650	530
No. of 4 input LUTs	1284	1117	940
No. of bounded IOBs	264	264	264
Maximum combinational path delay(in ns)	39.931 ns	39.531 ns	30.34 ns

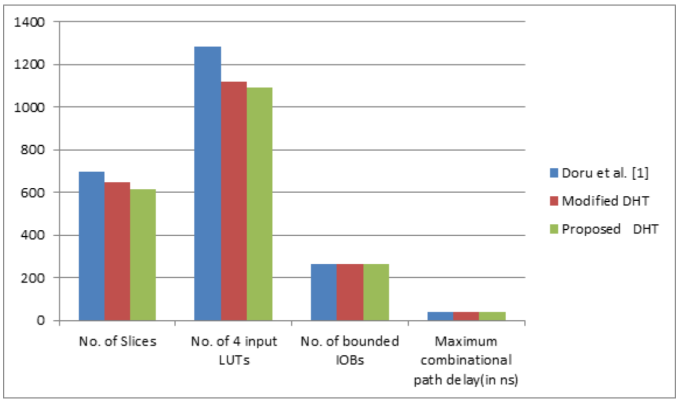


Figure 7: Bar Graph of the different DHT

VI. Conclusion

We found that CORDIC based DCT algorithm is the best algorithm in the existing algorithm. So we are implementation to CORDIC based DCT algorithm in this paper. The performance evaluation of the various sub modules are carried out using Xilinx 14.1 ISE Simulator and it was found that the circuits designed using DCT logic showed a reduced delay and power. As a future work more arithmetic and logical function can be used. The implementation DHT algorithm using Vedic multiplier and Kogge Stone adder utilizing value of design parameters such as LUTs, number of slice and MCPD

REFERENCES

[1] Doru Florin Chiper, Senior Member, IEEE, “A Novel VLSI DHT Algorithm for a Highly Modular and Parallel Architecture”, IEEE Transactions on Circuits and Systems—Ii: Express Briefs, Vol. 60, NO. 5, May 2013.

[2] Mamatha I, Nikhita Raj J, Shikha Tripathi, Sudarshan TSB, “Systolic Architecture Implementation of 1D DFT and 1D DCT”, 978-1-4799-1823-2/15/$31.00 ©2015 IEEE.

[3] Liyi Xiao Member, IEEE and Hai Huang, “Novel CORDIC Based Unified Architecture for DCT and IDCT”, 2012 International Conference on Optoelectronics and Microelectronics (ICOM) 978-1-4673-2639-1/12/$31.00 ©2012 IEEE.

[4] Shymna Nizar N.S,Abhila and R Krishna, “An Efficient Folded Pipelined Architecture For Fast Fourier Transform Using Cordie Algorithm”, 2014 IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT) IEEE.

[5] E. Jebamalar Leavline, S.Megala2 and D.Asir Antony Gnana Singh, “CORDIC Iterations Based Architecture for Low Power and High Quality DCT”, 2014 International Conference on Recent Trends in Information Technology 978-1-4799-4989-2/14/$31.00 © 2014 IEEE.

[6] Satyasen Panda, “Performance Analysis and Design of a Discreet Cosine Transform processor Using CORDIC algorithm”, 2008-2010.

[7] Keshab K. Parhi, “VLSI Digital Signal Processing Systems, design and implementation”, Wiley.

[8] Deepika Ghai, “COMPAATIVE ANALYSIS OF VARIOUS CORDIC TECHNIQUES”, June, 2011.

[9] Yuan-Ho Chen *et al*, “A High Performance Video Transform Engine by Using Space- Time Scheduling Strategy”, IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 20, NO. 4, APRIL 2012.

[10] Xue Liu, Feng Yu, Ze-ke Wang, " A Pipelined Architecture for Normal I/O Order FFT", Journal of Zhejiang University-SCIENCE C (Comput & Electron) , vol.12, no.1, 2011 , pp::76-82.

[11] Weihua Zheng, Kenli Li, Keqin Li, "A Fast Algorithm Based on SRFFT for Length N=qx2m DFTs", IEEE Trans. Circuits and Systems-II: Express Briefs, vol. 61, no.2, 2014, pp: 110-114.

[12] Weihua Zheng, Kenli Li, and Keqin Li," A Fast Algorithm Based on SRFFT for Length $N = q \times 2m$ DFTs , *IEEE Trans. Circuits Syst.II, Exp. Briefs*, vol. 61, no. 2, pp.110-114, Feb. 2014