FPGA Implementation Of DFT By Systolic Arrays

Ramanjaneyulu A, Asst. Prof in Dept. of ECE, VITAE, Deshmuki-508284
Pavan Kumar K, Adhoc Lecturer in Dept. of ECE, JNTU, Hyderabad-500085

Abstract

Field Programmable Gate Arrays (FPGAs) provide an efficient platform for physical hardware realization of digital signal processing (DSP) circuits. Discrete Fourier Transform (DFT) is a most frequently used block in many communication systems. In this work a recently developed high performance systolic DFT structure is realized in FPGA hardware. Virtex II device (XC2V8000) is used for physical implementation of this structure, which results in utilization of 26% slices. The ISE simulation results of circuit behavior is compared with the MATLAB based simulation outputs for verification of the DFT performance.

1. Introduction

Field Programmable Gate Arrays (FPGAs) with their concurrent processing capability are suitable for the implementation of many signal and image processing algorithms for diversified applications. The baseband processing in almost all modern communication system is being carried out in FPGAs. In communication systems, digital signal processing and image processing applications the discrete Fourier transform (DFT) is widely used functional block [1]. In order to convert time domain into frequency domain DFT is used in transmitter side and IDFT is used in receiver side of communication systems. For fast computation and parallel implementation of DFT, several algorithms have been proposed in literature [2], [3]. To achieve high computing speed and low computation complexity cyclic convolution and circular correlation are used. To maintain regularity in processing elements and to implement with less time systolic arrays are used [5].The N length DFT is converted into (N-1/2) point circular convolution. The systolic structures have same processing elements with control tag bits, based on which the addition or Subtraction operation is decided. In [8] a reduced complexity algorithm has been reported for computation of DFT in which a systolic architecture is also presented. In this work the above systolic array for DFT is implemented and the performance is verified.

This realisation in FPGA paves the way of integration of DFT with other baseband processing in FPGA for communication systems.

In section 2, the details of algorithm to be implemented in Very High Speed Integrated Circuit Hardware Description Language (HDL) language is presented. Section 3, provides the FPGA implementation of DFT. In section 4, the simulation and synthesis results are discussed. The findings of this study are concluded in section 5.

2. DFT algorithm implemented

The objective of this work is to realize the DFT algorithm systolic array proposed in [8] in FPGA and observe the performance. The major governing equations of DFT algorithm are as follows.

The DFT of a sequence y (n) for n=0, 1, 2, 3. . . N-1 is given as,

$$X(k) = A(k) - jB(k), \qquad \text{for } 0 \le k \le \text{N-1} \qquad (1a)$$

$$A(k) = \sum_{n=0}^{N-1} y(n)\cos[4\alpha_N kn] \qquad (2a)$$

$$B(k) = \sum_{n=0}^{N-1} y(n)\sin[4\alpha_N kn] \qquad (2b)$$

for $\alpha_N = \Pi / 2N$ .When N is even, we can reduce into sum of {N/2+1} terms as

$$A(k) = \sum_{n=0}^{N/2} a(n)\cos[4\alpha_N kn] \qquad (2c)$$

$$B(k) = \sum_{n=0}^{N-1} b(n)\sin[4\alpha_N kn], \qquad \text{for } 0 \le k \le \text{N-1} \qquad (2d)$$

Where

$$a(n) = y(n) + y(N-n) \qquad (3a)$$
$$b(n) = y(n) - y(N-n), \quad \text{for } 1 \le \text{n} \le \text{(N/2)-1} \qquad (3b)$$

$$A(k) = A_1(k) + A_2(k) \qquad (4a)$$
$$B(k) = B_1(k) + B_2(k), \quad \text{for } 0 \le k \le N-1 \qquad (4b)$$

When (N/2) is even then N=4M, where M is an integer

$$A_1(k) = \sum_{n=0}^{M} a_1(n)\cos[2\alpha_M kn] \qquad (5a)$$
$$A_2(k) = \sum_{n=0}^{M-1} a_2(n)\cos[\alpha_M k(2n+1)] \qquad (5b)$$

$$B_1(k) = \sum_{n=0}^{M-1} b_1(n)\sin[2\alpha_M kn] \qquad (5c)$$
$$B_2(k) = \sum_{n=0}^{M-1} b_2(n)\sin[\alpha_M k(2n+1)] \qquad (5d)$$

for $1 \le \text{k} \le \text{M-1}$

The equations (5b), (5d) are discrete cosine and discrete sine transforms. The above equations are analysed by the methods followed in [6-7].

$$A_1(0) = \sum_{n=0}^{M} a_1(n) \qquad (5e)$$
$$A_2(0) = \sum_{n=0}^{M} a_2(n) \qquad (5f)$$
$$B_1(0) = B_2(0)\text{=0} \qquad (5g)$$

$$a_1(n) = a(2n)$$
$$\text{a}_2\text{(n)=a(2n+1)}$$
$$\text{b}_1\text{(n)=b(2n)}$$
$$\text{b}_2\text{(n)=b(2n+1), for } 1 \le \text{n} \le \text{(N/4)-1} \qquad (6a)$$

$$a_1(0) = y(0)$$
$$\text{a}_2\text{(0)=y(1)+y(N-1)}$$
$$\text{b}_1\text{(0)=0}$$
$$\text{b}_2\text{(0)=y(1)-y(N-1).} \qquad (6b)$$

The N components of A (k) and B (k) can be computed into {(N/4) +1} components

$$A(N/2+k) = A_1(k) - A_2(k), \quad \text{for } 0 \le \text{k} \le \text{N/4} \qquad (7a)$$
$$A(N/2-k) = A((N/2)+k), \quad \text{for } 1 \le \text{k} \le \text{N/4} \qquad (7b)$$
$$A(N-k) = A(k), \quad \text{for } 1 \le \text{k} \le \text{N/4-1} \qquad (7c)$$

$$B(N/2+k) = B_1(k) - B_2(k), \quad \text{for } 0 \le \text{k} \le \text{N/4} \qquad (7d)$$
$$B(N/2-k) = -B((N/2)+k), \quad \text{for } 1 \le \text{k} \le \text{N/4-1} \qquad (7e)$$

$$B(N-k) = -B(k), \quad \text{for } 1 \le \text{k} \le \text{N/4-1} \qquad (7f)$$

2.1. Conversion into Circular form

$$A_1(2k) = a_1(0) + a_1(M) + S(k) \qquad (8a)$$
$$A_1(2k-1) = a_1(0) - a_1(M) + D(k) \qquad (8b)$$

for $1 \le \text{k} \le \text{(M-1)/2}$

$$S(\zeta) = \sum_{n=1}^{(M-1)/2} \{sign_S(\zeta,n)\}s(n)\cos[\pi\varphi_M(\zeta,n)] \qquad (9a)$$
$$D(\xi) = \sum_{n=1}^{(M-1)/2} \{sign_D(\xi,n)\}d(n)\cos[\pi\theta_M(\xi,n)] \qquad (9b)$$

$$s(n) = a_1(n) + a_1(M-n) \qquad (9c)$$
$$d(n) = a_1(n) - a_1(M-n) \qquad (9d)$$

$$sign_S(\zeta,n) = (-1)^{\text{round}(2\zeta\text{n/M})} \qquad (9e)$$
$$sign_D(\xi,n) = (-1)^{\text{round}((2\xi-1)\text{n/M})} \qquad (9f)$$

$$\varphi_M(\zeta,n) = \begin{cases} [2\zeta n)_M / M, & \text{if } (2\zeta\text{n})_M \le M/2 \\ M - [((2\zeta n)_M]/M, & \text{if } (2\zeta\text{n})_M > M/2 \end{cases} \qquad (9g)$$

$$\theta_M(\xi,n) = \begin{cases} [((2\xi-1)n)_M / M, & \text{if } ((2\xi-1)\text{n})_M \le M/2 \\ M - [((2\xi-1)n)_M]/M, & \text{if } (2\xi\text{-1)n})_M > M/2 \end{cases}$$
$$\text{for } 1 \le \xi,\ \zeta \le \text{(M-1)/2.} \qquad (9f)$$

When M is prime, the sequence S (ζ) and D (ξ) for 1≤ζ, ξ≤ (M-1)/2 in (9a) and (9b), can be converted into two (M-1)/2 point circular convolution.

2.2. Twiddle Factor Multiplication

Cooley Turkey algorithm is used for complex twiddle factor multiplication. If the complex twiddle factor is $e^{j\pi/9}$=C + S. Then the computed terms taken as C, C+S, C-S. For n bit complex multiplication the twiddle factor value is multiplied by 2^{n-1}. The value is finally divided by 2^{n-1}.

2.3. Converting into Circular Convolution

The equations A_1 (k), B_1 (k), A_2(k) and B_2(k) converted into circular form. The conversion of A_1 (k) into required circular convolution form for M=7 is shown for (9a) and (9b) . Similarly B_1 (k) is converted into circular convolution form. The equations (5b) and (5d) are converted into circular convolution form by the methods followed in [6-7].

$$\begin{bmatrix} S(1) \\ S(2) \\ S(3) \end{bmatrix} = \begin{bmatrix} \cos(2\beta) & -\cos(3\beta) & -\cos(\beta) \\ -\cos(3\beta) & -\cos(\beta) & \cos(2\beta) \\ -\cos(\beta) & \cos(2\beta) & -\cos(3\beta) \end{bmatrix} \begin{bmatrix} s(1) \\ s(2) \\ s(3) \end{bmatrix}$$

$$\begin{bmatrix} D(1) \\ D(2) \\ D(3) \end{bmatrix} = \begin{bmatrix} \cos(\beta) & \cos(2\beta) & \cos(3\beta) \\ \cos(3\beta) & -\cos(\beta) & -\cos(2\beta) \\ -\cos(2\beta) & -\cos(3\beta) & \cos(\beta) \end{bmatrix} \begin{bmatrix} d(1) \\ d(2) \\ d(3) \end{bmatrix}$$

where $\beta = \pi/7$ and

$s(1) = a_1(1) + a_1(6)$

$s(2) = a_1(2) + a_1(5)$

$s(3) = a_1(3) + a_1(4)$

$d(1) = a_1(1) - a_1(6)$

$d(2) = a_1(2) - a_1(5)$

$d(3) = a_1(3) - a_1(6)$

2.4. Systolic Architecture

Every processing element in systolic structure is identical to each other. Based on tag bit value the addition or subtraction is decided. In systolic arrays the pipelining of the data is observed.

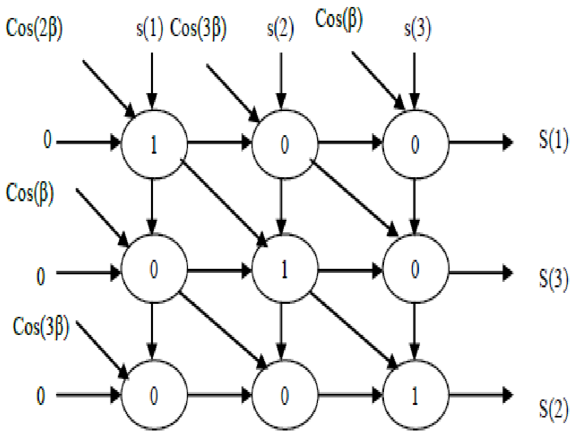

Fig. 1. Systolic Structure for S(k) [8]

The dependence graph consists of 9 processing elements, arranged in 3 rows and columns. If tag =1 then addition operation is performed otherwise subtraction is performed. Each processing element has two multipliers and two adder/subtractor. One multiplier and one adder/subtractor are used to calculate real values and the other for imaginary.

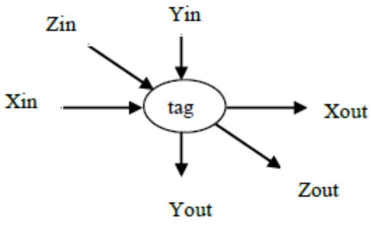

Fig.2. Function of each processing element

If tag = 1 Xout = Xin + Yin. Zin

Else Xout = Xin - Yin. Zin

Yout = Yin, Zout = Zin

3. FPGA Implementation

A field-programmable gate array (FPGA) is a programmable integrated circuit designed to be configured by the customer or designer after manufacturing. The logic realization in FPGA is performed by describing the functionalities through

VHDL. In this work DFT is basically divided into 8 sub modules. According to the equations all the modules are designed and are coded in VHDL language. For enabling proper testing in the simulation environment the structure as shown in fig.4 is implemented in FPGA. The data which is coming serially is converted into parallel data. The parallel data is given to respective DFT block and the DFT calculations are performed. The computed parallel DFT parallel data are converted to serial by the parallel to serial converter. Based on the tag bit in systolic structure the addition or the subtraction is used in each processing element. Ripple carry adders are used for less area usage. The targeted device is Virtex II xc2v8000. The design, synthesis and simulation is carried out by ISE simulator.

4. Results and Discussions

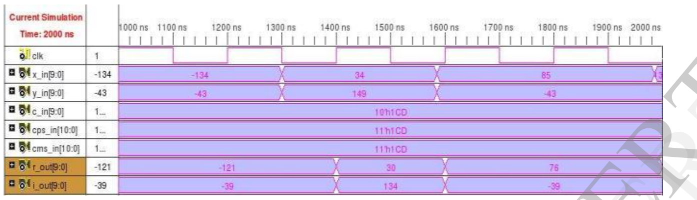

Fig. 3. Complex multiplication result

If (X +j Y) (C +j S) = R +j I then in the multiplication result R and I are expressed as R=Y(C - S) + (X - Y) C and I=X(C + S) – (X - Y) C. For complex multiplication the values of C, C+S ,C-S are calculated and stored in memory. The behavioural simulation of twiddle factor multiplication is reported in fig.5.

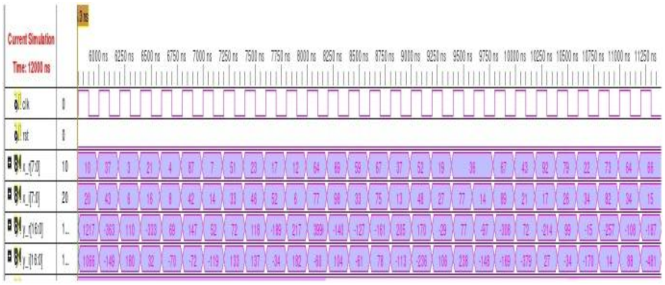

Fig. 4. Functional Simulation of 28 point DFT

The input real and imaginary values are taken as x_r and x_i. For every positive edge of clock the input is taken serially. After 28 clocks the input values are taken DFT block and processing is done. The output is serially taken as y_r and y_i for real and imaginary values respectively. In fig 6 the behavioural simulation results are depicted.

4.1 Table: Synthesis report of 28 point DFT

	Resources Used	% utilization
No. of Slice Flip Flops	2508	2
No. of Slices	12168	26
No. of 18*18 Multipliers	168	100
No. of 4 input LUT’s	23181	24
No. of GCLK’s	1	6

5. Conclusion

This work presents FPGA implementation of an efficient systolic DFT structure. To minimize the area ripple carry adders and ripple carry subtractions are used in the design. 28-point DFT is considered for physical implementation in the Virtex II FPGA. The functional simulation results of ISE simulator and MATLAB environment are in well agreement with each other. The systolic DFT processor is of high throughput and low latency owing to its systolic structure.

References

[1] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*. Upper Saddle River, NJ: Prentice-Hall, 1996.

[2] P. Duhamel and M. Vetterli, "Fast Fourier transforms: A tutorial review and a state of the art," *Signal Process.*, vol. 19, pp. 259–299, 1990.

[3] R.N. Bracewell, "*The Fourier Transform and its Applications*, "McGraw-Hill, Third Edition, 2000.

[4] *Fundamentals of Digital Image Processing*. Englewood Cliffs,NJ: Prentice-Hall, 1989.

[5] H. T. Kung, "Why systolic architectures",*Comput. Mag.*, vol. 15, pp.37–45.

[6] C. Cheng and K. K. Parhi, "A Novel Systolic Array Structure for DCT," *IEEE Trans. Circuits and Systems II: Express Briefs*, Accepted for future publication, 2005.

[7] D. F. Chiper, M. N. S. Swamy, M. O. Ahmad, and T. Stouraitis, "A systolic array architecture for the discrete sine transform," *IEEE Trans. Signal Processing*, vol. 50, pp. 2347-2354, 2002.

[8] P.K.Meher,"Efficient Systolic Implementation of DFT Using aLow-Complexity Convolution-Like Formulation", *IEEE Transactions on Circuits and Systems—ii: Express Briefs, vol. 53, no. 8, August 2006.*

[9] P. K. Meher, "A New Convolutional Formulation of the DFT and Efficient Systolic Implementation," Submitted to *IEEE International Region 10 Conference* TENCON'05, Melbourne, Australia-Nov2005.