Diyala Journal of Engineering Sciences

ISSN 1999-8716
Printed in Iraq

Vol. 09, No. 04, pp. 83-92, December 2016

LOW POWER AND HIGH SPEED DFT ARCHITECTURE

Ahmed K. Jameil
Assistant lecturer, College of Engineering, University of Diyala
ahmedkhudaer@yahoo.com
(Received: 9/12/2015; Accepted: 17/2/2016)

ABSTRACT: - Discrete Fourier Transform (DFT) plays essential role in many signal processing applications. In this paper, novel hardware architecture is presented for DFT. It is based on various transform sizes (N) in aim to provide low power and high speed for modern multimedia applications. The proposed architecture uses two Multipliers and adders to perform the computation of DFT for different sizes of input data. Full analysis of architecture is discussed thoroughly for various transform sizes (N). This analysis consists of power consumption, hardware cost and speed parameters discussion. In addition, the implementation of this architecture in Field Programmable Gate Arrays (FPGA) is explained. Less than 1.62 mW dynamic power consumption for N=128 at 100 MHz operating frequency is achieved by proposed architecture. Finally, the comparison with state of art architectures results reveals that the proposed architecture outperforms other architectures in terms of speed and hardware cost.
Keywords: *Verilog-HDL, DFT, Synchronous Dataflow Graph (SDFG).*

INTRODUCTION

Fourier transforms represent an important tool in the different applications of digital signal processing (DSP). Such applications include image processing, acoustics, optics, signal, speech, medical electronics and telecommunications systems [1, 7]. The spectrum analysis, the linear filtering of signals and correlation analysis are important applications of the DFT. The spectrum analysis application is based on the close relationship between the Fourier transform of a signal and the DFT of the same sampled signal .The second application (linear filtering) is due to the fact that the DFT has the cyclic convolution property, that is, it can transform the convolution operation into a number of point wise product.[6]

The direct computation of DFT requires on the order of N^2 operations where N is the transform size or the filter length. [5] Thus, it is difficult and time-consuming to be implemented in hardware. However, the DFT has an advantage over the FFT because it requires lower number of iteration cycles which reduces the chip area and power consumption of its hardware architecture. [4]

Most of researches to date for implementation and benchmarking of the DFT algorithms have been performed using general purpose processors, Digital Signal Processors(DSPs) and dedicated DFT processor ICs . However, as Field Programmable Gate Arrays (FPGAs) have grown in capacity, improved in performance, and decreased in cost. They become a useable solution for performing computationally intensive tasks, with the ability to tackle applications for custom chips and programmable DSP devices. [2, 3, 10] Many papers are introduced on fast hardware implementations of DFT. As VLSI technology is advanced greatly, new criteria of designing and evaluating of the fast algorithms in the architecture realization. They are designed and optimized for specific tasks such as power consumption, hardware cost optimization or speed. [8, 9]

The DFT requires low number of iteration cycles and thus lower hardware usage and power consumption when it compared with the FFT. [4] So, this paper introduces a low power and high speed architecture for the DFT algorithm based on this property. The proposed architecture is implemented using Quartus 13.1 design suite from Altera. [13] The Efficient DFT architecture is developed using hardware description language Verilog-HDL code generation based on the Synchronous Dataflow Graph (SDFG) specifications.
The important features of the proposed architecture are:

- Low power consumption.
- Easy to be implemented for any required DFT transform size.
- Low initial latency.
- Low hardware cost.
- Low complexity.

The rest of this paper is organized as follows: in section-II, the Discrete Fourier Transform (DFT) algorithm is introduced; section-III introduces the proposed architectures of the DFT. The implementation results are listed and discussed in section-IV, comparison and discussion results are discussed in section -V. Finally, the conclusions are listed in section-VI.

I. THE DISCRETE FOURIER TRANSFORM ALGORITHM (DFT)

The DFT is considered as an important transform that used to analyze signals to its fundamental and harmonics frequencies. The DFT can be computed using: [12]

$$X[k] = \sum_{n}^{N-1} x[n] \times \exp\left(-j \frac{2\pi kn}{N}\right) \quad \textbf{(1)}$$

where N is the size of the transform, x the input signal and X is the DFT output signal .The regular algorithm for DFT computation is shown in Figure (1). The Figure describes the computation procedure of N sample input signal x[n], where, n and k are the indices in time and frequency, respectively. X[k] and Y[k] are the real and imaginary parts of the DFT signal.

II. PROPOSED DFT ARCHITECTURE

The DFT is considered as low iteration cycles algorithm when it compared with the fast Fourier transform algorithm [4] Thus, in this section an efficient DFT Architecture is introduced. The Signal flow graph, Synchronous Dataflow graph, Data path unit of the DFT and the top level of the proposed architecture will be explained.

A. Signal Flow Graph

As shown in Figure (1), The DFT required high computation complexity. Therefore, it is important to present an efficient architecture to tackle the computation challenges of the DFT. The signal Flow Graph (SFG) of the DFT algorithm is shown in Figure (2). In this Figure, the input signal, x (n), is multiplied by sin (g) and cos (g) trigonometric functions in parallel. The addition operations are performed as described by the two nodes X (k) and Y(k) in Figure (2). RealX and ImagY represent the real and imaginary parts of the DFT output signals, respectively.

B. Synchronous Dataflow Graph

Synchronous Dataflow Graph (SDFG) consists of operations and tasks that run on a predefined number of samples to produce fixed number of output values. The DFT SDFG is shown in Figure (3). There are N^2 and $N(N-1)$ nodes for multiplication and addition operations, respectively. The feedback paths from output to the multipliers require a delay, and the delays are denoted by black dots on the SDFG. These delays are implemented using memory registers. To compute the DFT of the input signal the operations in Figure (3) are repeated N-times.

C. Datapath Unit of the DFT

The actual computation operations of any DSP algorithm are represented by a dedicated Datapath unit. Figure (4) shows the Datapath unit for 8N bits dedicated processor to compute the DFT for N-sample input data each with 8-bit size.

The DFT processor comprises of register memory, two counters and two Arithmetic Logic Units (ALU). An N 8-bit register size is required to compute the DFT for N-sample input data and the two counters are used as controllers. The two ALU are used for real and imaginary parts data accumulating.

D. Toplevel of the DFT Architecture

In Figure (5) represents the Datapath unit (DU) and the CU is the control unit. The DU is described section II-C and the CU is used to generate a control vector (CV) which is implemented as a finite state machine. The size of the input and output data ports is 8 and 32-bit, respectively. The output data is distributed as 16-bit for each Real and Imaginary parts. The input data is fed sample by sample in a rate of one-sample/clock cycle.

III. IMPLEMENTATION PERFORMANCE EVALUATION

The proposed architecture is implemented using EP3C5F256C6 Cyclone III Altera FPGA platform. The synthesis results are computed using Quartus 13.1. [13] Five transform sizes are selected to show the performance of the proposed architecture. The selected transform sizes are N=8, 16, 32, 64 and 128. In additional, any other transform size can be implemented and evaluated easily.

A. Hardware Utilization Rate

The hardware usage of the proposed DFT architecture is shown in TABLE I. It can be noticed from the table, that 15-registers and 26-LUT are required to perform the DFT computation operations of 8 samples. The required hardware resources increased slightly according to the size of the transform. It is obvious from TABLE I that four registers and five LUT are added for each transform size, although the overall hardware usage can be considered as very low utilization rate. The aforementioned results declare clearly that the proposed architecture consumes about less than 1% hardware components from the available platform resources.

B. Power Consumption

The power consumption of the proposed architecture is evaluated using five transform sizes as shown in TABLE II to TABLE IV for N=8, 32 and 128 transform sizes. A full comparison of the dynamic power consumption is shown in Figure (6). The dynamic power consumption is computed using N=8, 16, 32, 64 and 128-sample transform size for various operating frequencies.

It can be noticed from TABLE II - TABLE IV and Figure (6) that the dynamic power consumption of the proposed architecture is less than 3.18 mW. However, the total power consumption is about 87 mW. The high total power consumption can be explained because it depends on static and dynamic power consumption. The dynamic power consumption depends on the proposed architecture while the static power illustrates the power consumed by the specific FPGA platform. Thus, to reduce the total power consumption any other FPGA platform with low static can be used instead.

C. Timing Report

TABLE V illustrates the timing report of the proposed architecture using EP2S15F67C5 Altera FPGA platform based on N-sample DFT. From the table it is obvious that up to 264 MHz operating frequency with 0.03 µs initial latency can be attained. Even at high transform size (N=128) an up to 234 MHz operating frequency can be achieved. It is also clear that the computation time range is from 0.24 to 69 µs for transform size from N=8 to 128, respectively. The low computation time represents an important feature that make the proposed architecture suitable for various DSP systems. For most desirable systems the transform size is N=8. It is consistently in evaluation that the N=8 is by far the most efficient algorithm in terms of computation time with high speed. Further, the initial latency represents

an important parameter for the comparison of speed performance in DSP applications. In particular, low latency rate is usually desirable in most high-performance DSP applications. So, all DFT processor performance shown in TABLE V have latency range from 0.03 to 0.55 µs for N=8 to N=128, respectively.

V. COMPARISON AND DISCUSSION

In this section, the performance of the proposed DFT architecture is compared with state of art architectures in terms of hardware cost and processing speed as shown in TABLE VI and TABLE VII. It can be noticed from TABLE VI that the proposed architecture has the advantage of low hardware usage whenever compared with other architectures, only 2 multipliers and 2 adders are required to perform the computation of the DFT in the proposed architecture. The reduction of hardware resources is achieved by minimizing the required number of adders. This minimization is the best among proposed and state of art architectures. Thus, the low hardware usage of the proposed architecture turns out to be an appropriate solution for VLSI implementation whereas the hardware cost is a crucial factor.

TABLE VII presents a comparison between the performance of the proposed architecture with other architectures [6, 10, 11]. The maximum operating frequency and the required number of logic cells are listed in TABLE VII. It is obvious that an up to 264 MHz can be achieved which is much higher than the operating frequency of other architectures. Furthermore, the required number of logic cells is much less than required number in other architectures [6, 10, 11]. Hence, the proposed architecture provides efficient DFT architecture in terms of performance which make it suitable candidate for the low hardware usage and high frequency requirement applications.

VI. CONCLUSIONS

In this paper, low power and high speed architecture for DFT is introduced. The DFT architecture is tested using various transform sizes N=8, 16, 32, 64 and 128, so it is suitable for any transform size. The whole DFT computation process is accomplished using two adders, two multipliers and a memory buffer. The performance analysis of the proposed architecture reveals that a speed of up to 264 MHz for N=8 can be achieved with 0.03 and 0.24 µs initial latency and computation time, respectively. Further, the dynamic power consumption is less than 3.18 mW for N=128 while it reduced to 0.8 mW for N=8. The performance of the proposed DFT architecture outperforms other architectures in terms of hardware usage and speed. The low hardware usage can be considered as an important feature for limited area applications.

REFERENCES

1. U. Meyer-Baese, (2014), “Digital Signal Processing with Field Programmable Gate Arrays”, 4th Edition. Springer-Verlag Berlin Heidelberg, pp. 347-349.
2. P. A. Milder, F. Franchetti, J. C. Hoe, and M. Puschel, (2008), “Formal datapath representation and manipulation for implementing DSP transforms”, in Proc. Design Automation Conference, pp. 385–390.
3. Proakis, J. G., E. A. B. da silva and S.L. Netto, (2010), “Digital Signal Processing, system analysis and design”, 2nd Edition, pp.143-154.
4. Y. Juan, H. Huang, S. Lai, (2015),”A Distortion Cancelation Technique with the Recursive DFT Method for Successive Approximation Analog-to-Digital Converters”, IEEE Transactions on Circuits and Systems II, Vol. 63, issue 2, pp. 146-150.
5. S. C. Lai ,Y.S. Lee, S. F. Lei,(2013),“Low-Power and Optimized VLSI Implementation of Compact Recursive Discrete Fourier Transform (RDFT) Processor for the Computations of DFT and Inverse Modified Cosine Transform (IMDCT) in a Digital Radio Mondiale (DRM) and DRM+ Receiver”, jlpea,Vol.3,pp.99-113.

6. A. Mankar, N. Prasad, S. Meher, (2013),"FPGA Implementation of Discrete Fourier Transform Core Using NEDA", IEEE International Conference on Communication Systems and Network Technologies,pp.711-715.
7. M. V. Prasad, K. C. Ray and A.S. Dhar, (2010), "FPGA Implementation of Discrete Fractional Fourier Transform", International Conference on Signal Processing and Communications, IEEE, IISc, Bangalore, India,17-21 July,pp.1-5.
8. A. Mandal and R. Mishra, (2012)," Reconfigurable Design of Pipelined CORDIC Processor for Digital Sine-Cosine", Journal of Signal Processing Theory and Applications, Vol. 1, pp.27-35.
9. Lai, S.C., Juang, W.H., Lin C., Luo, C.H., Lei, S.F., (2011), "High-throughput, power-efficient, coefficient-free and reconfigurable green design for recursive DFT in a portable DRM receiver". Int. J. Electr. Eng., pp. 137–145.
10. A. Shaik, U. Yedukondalu, (2013), "A Lower Power, Low Complexity and Memory Free Implementation of Novel Recursive DFT and IDFT Algorithms", IJERA, Vol. 3, Issue 4, pp.662-666.
11. M. Rawski, M. Vojtynski, T. Wojciechowski, and P. Majkowski, (2007), "Distributed Arithmetic Based Implementation of Fourier Transform Targeted at FPGA Architectures", Proc. Intl. Conf. Mixed Design, pp. 152 – 156.
12. S. A. Khan, (2011),"DIGITAL DESIGN OF SIGNAL PROCESSING SYSTEMS", 1st Edition, John Wiley & Sons, Ltd, pp.295-300.
13. Altera, (2013), "Floating-Point Megafunctions User Guide", ver. 13.
14. S. Y. Lee and C. Chen, (2006), "VLSI Implementation of Programmable FFT Architectures for OFDM Communication System", The International Wireless Communications and Mobile Computing Conference, Canada, July 3-6, pp. 893-898.

TABLE I: Utilisation rate, Maximum speed, initial latency and computation time of the proposed architecture using various DFT sizes

	Available	**Transform Length**				
	Resources	**N=8**	**N=16**	**N=32**	**N=64**	**N=128**
Dedicated logic register	5136	15	19	23	27	31
Number used as logic	5136	26	26	36	42	47
Number of IOs	183	59	65	67	71	75
Combinational LUTs	5136	26	31	36	42	47
Average fan out	--	1.32	1.4	1.5	1.57	1.63

TABLE II: Power Consumption Details of the Proposed Architecture When N=8

Clock Frequency (MHz)	**I/O Power (mW)**	**Dynamic (mW)**	**Static (mW)**	**Total (mW)**
50	17.1	0.74	46.12	64.03
100	19.9	1.46	46.13	67.51
150	22.75	2.17	46.13	71.05
200	25.53	2.9	46.13	74.57

TABLE III: Power Consumption Details of the Proposed Architecture When N=32

Clock Frequency (MHz)	I/O Power (mW)	Dynamic (mW)	Static (mW)	Total (mW)
50	19.19	0.88	46.13	66.19
100	23.4	1.56	46.13	71.1
150	27.64	2.31	46.14	76.09
200	31.81	3.07	46.14	81.01

TABLE IV: Power Consumption Details of the Proposed Architecture When N=128

Clock Frequency (MHz)	I/O Power (mW)	Dynamic (mW)	Static (mW)	Total (mW)
50	21.37	0.8	46.13	68.3
100	26.85	1.62	46.14	74.6
150	32.26	2.39	46.14	80.8
200	37.79	3.18	46.15	87.12

TABLE V: Implementation Timing Report of the proposed architecture using EP2S15F672C5 Altera FPGA Platform

	Transform Length				
	N=8	N=16	N=32	N=64	N=128
Max Speed (MHz)	264.69	260.42	246.79	232.99	234.63
Initial Latency (µs)	0.03	0.06	0.13	0.27	0.55
Computation Time (µs)	0.24	0.98	4.15	17.58	69.83

TABLE VI: Hardware costs for various DFT architectures

	Adder/ subtractor	Multiplier
Computations of DFT[5]	4	2
RDFT[10]	12	2
DFT-NEDA[6]	31	0
The proposed architecture	2	2

TABLE VII: Comparison between the proposed architecture and other architectures

	Max Operating Frequency (MHz)	Resource Usage (Logic Cells)
FFT-LC[11]	43.51	4723
FFT-DA[11]	74.36	7222
RDFT[10]	69.31	780
DFT-NEDA[6]	82.03	428
The proposed architecture	264.69	126

$N = Transform\ \text{size}$
$x[n] = Input\ \text{samples}$
$for(k = 0; k < N; k++)\{$
$X[k] = 0;$
$Y[k] = 0;$
$for(n = 0; n < N; n++)\{$
$X[k] = X[k] + x[n] \times \cos\left(\frac{2\pi kn}{N}\right)$
$Y[k] = Y[k] + x[n] \times \sin\left(\frac{2\pi kn}{N}\right)$
$\}\}$

Figure (1): The DFT computation algorithm

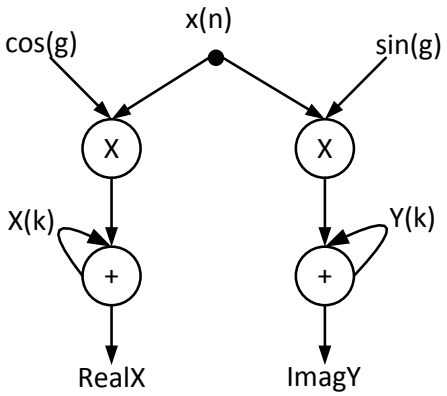

Figure (2): Signal Flow Graph

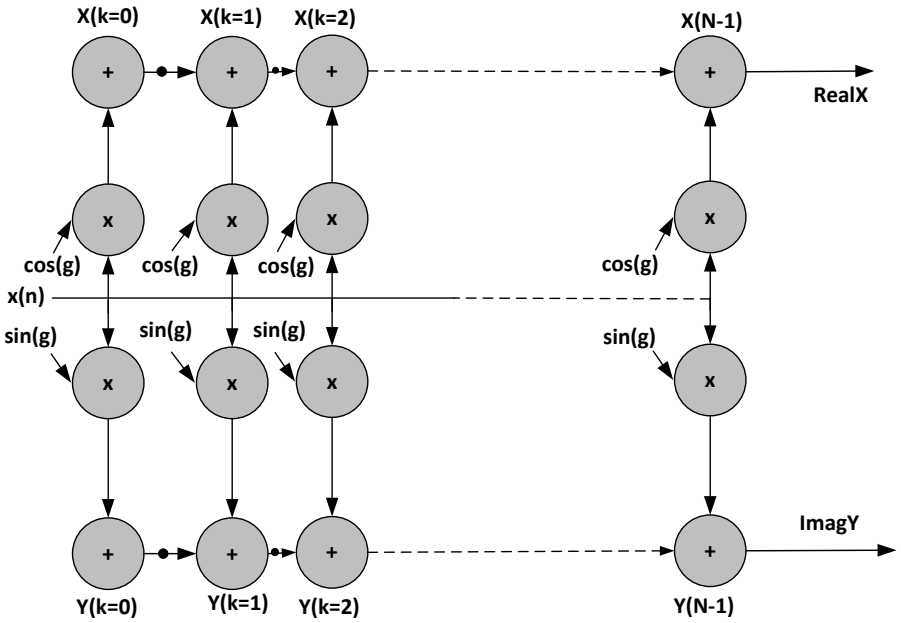

Figure (3): Synchronous Dataflow Graph (SDFG)

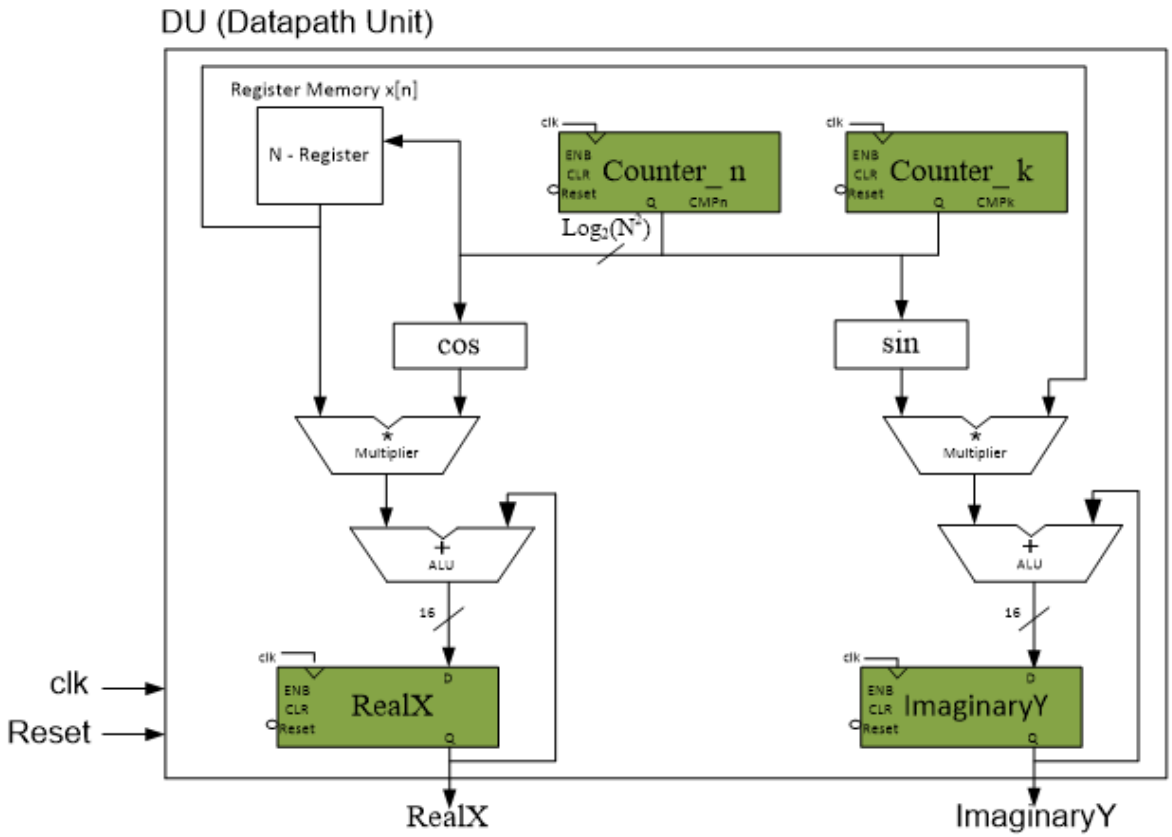

Figure (4): The Datapath Unit of the DFT.

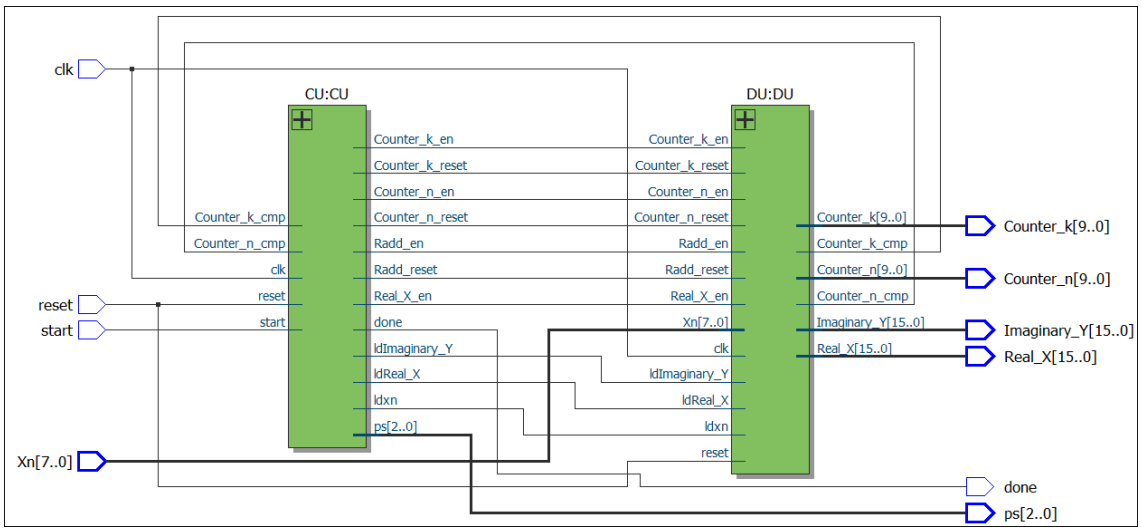

Figure (5): The Toplevel of DFT architecture for N=32

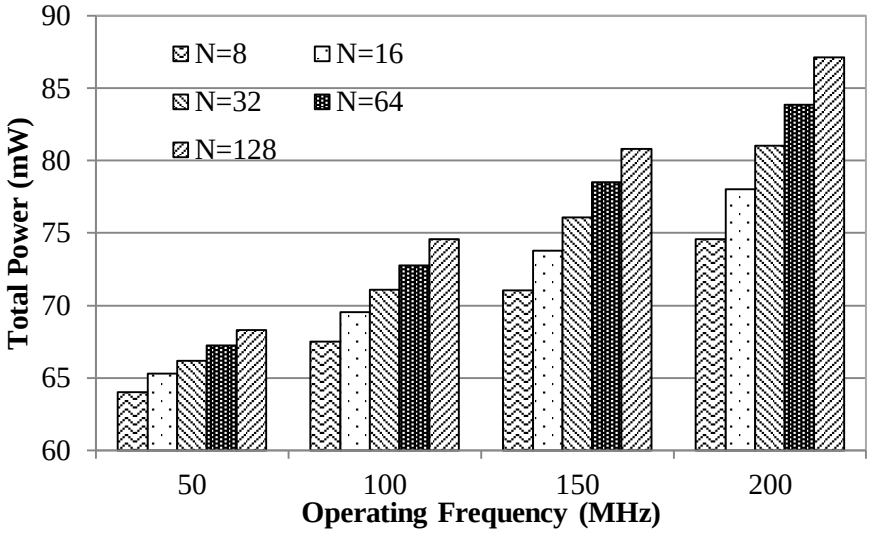

Figure (6): Dynamic power consumption of the proposed architecture using N=8, 16, 32, 64 and 128 for various operating frequencies

معمارية ذات طاقة منخفضة وسرعة عالية لتحويلات فورير المتقطعة

م.م أحمد خضير جميل
كلية الهندسة – جامعة ديالى – بعقوبة – ديالى – العراق
ahmedkhudaer@yahoo.com

الخلاصة :

في هذا البحث تم تصميم معمارية لتحويلات فورير المتقطعة لمختلف احجام عينة الادخال لتطبيقات منخفضة الطاقة ذات السرعة العالية. تم تنفيذ المعمارية المقترحة على اجهزة Altera ضمن البوابات المنطقية القابلة للبرمجة (FPGA). المعمارية المقترحة تستخدم 2 مضاعف منطقي (Two Multipliers) و 2 جامع (Two Adders) لحساب تحويلات فورير المتقطعة لاي عدد من العينات. تحليلات كاملة للمعمارية المقترحة باستخدام احجام مختلفة للعينة تم حسابها وتناولها في هذا البحث. هذه التحليلات تتضمن القدرة المستهلكة والمكونات المادية المستخدمة في المعمارية وسرعة التنفيذ. اظهرت النتائج ان القدرة المستهلكة اقل من1.62 ملي واط لاشارة بحجم 128 عينة بتردد تشغيل مقدارة 100 ميكا هيرتز. واخيرآ تمت مقارنة نتائج المعمارية المقترحة مع معماريات اخرى وأظهرت النتائج ان المعمارية المقترحة افضل من حيث السرعة و تكلفة الاجهزة .