International Journal for Innovative
Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

COPY RIGHT

IJIEMR Transactions, online available on 23rd February 2018. Link :
http://www.ijiemr.org/downloads.php?vol=Volume-7&issue=ISSUE-2

Title: Fixed And Normalized Architecture Based On Multi Level Radices For Delay Commutator And Feedback Techniques.

Volume 07, Issue 02, Page No: 605 - 610

Paper Authors

***ANIVIREDDY POSIYA, M SURESH KUMAR.**

* Dept of ECE, Kakinada Institute of Engineering & Technology For Women.

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

International Journal for Innovative Engineering and Management Research

A Peer Revieved Open Access International Journal

www.ijiemr.org

FIXED AND NORMALIZED ARCHITECTURE BASED ON MULTI LEVEL RADICES FOR DELAY COMMUTATOR AND FEEDBACK TECHNIQUES

***ANIVIREDDY POSIYA, **M SURESH KUMAR.**

*PG Scholar,Vlsd, Dept.of ECE, Kakinada Institute of Engineering & Technology, Korangi, A.P.

**Assistant Professor, Dept.of ECE,Kakinada Institute of Engineering & Technology, Korangi, A.P.

ABSTARCT:

We present an efficient combined single-path delay commutator-feedback radix-2 pipelined fast Fourier transform architecture, which includes log2 N − 1 SDC stages, and 1 SDF stage. The SDC processing engine is proposed to achieve 100% hardware resource utilization by sharing the common arithmetic resource in the time-multiplexed approach, including both adders and multipliers. Thus, the required number of complex multipliers is reduced to log4 N − 0.5, compared with log2 N − 1 for the other radix-2 SDC/SDF architectures. In addition, the proposed architecture requires roughly minimum number of complex adders log2 N + 1 and complex delay memory 2N + 1.5 log2 N − 1.5.

INTRODUCTION:

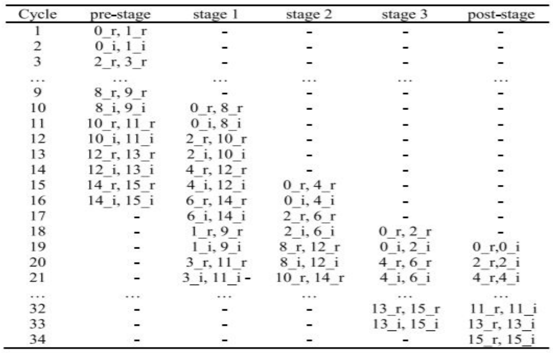
Fast Fourier transform (FFT) has played a significant role in digital signal processing field, especially in the advanced communication systems, such as orthogonal frequency division multiplexing (OFDM) [1] and asymmetric digital subscriber line [2]. All these systems require that the FFT computation must be high throughput and low latency. Therefore, designing a high-performance FFT circuit is an efficient solution to the abovementioned problems. In particular, the pipelined FFT architectures have mainly been adopted to address the difficulties due to their attractive properties, such as small chip area, high throughput, and low power consumption. To the best to our knowledge, two types of pipelined FFT architectures can be found in this brief: delay feedback (DF) and delay commutator (DC). Further, according to the number of input data stream paths, they can be classified into multiple-path (M) or single-path (S) architectures. The two classifications form four kinds of pipelined FFT architectures [e.g., single-path DC (SDC)]. Multiple-path (M) architectures [3]–[9], are often adopted when the throughput requirement is beyond the theoretical limitation that the single-path architecture can offer at a given clock frequency. However, they require concurrent read (write) operations for the multipath input (output) data. Therefore, single-path (S) architectures could be appropriate in some cases when the system cannot ensure concurrent operations. However, the arithmetic utilization is relatively low, compared with 100% utilization of the existing MDF/MDC architectures [4]. In this brief, we focus on the SDC radix-2 pipelined FFT architecture, which can also achieve 100% multiplier utilization by reordering the inner data sequence. For single-input data stream, the conventional radix-2 SDF FFT architecture [10] requires 2 log2 N complex adders and log2 N − 1complex multipliers, where N is the FFT size. Both Chang [11] and Liu et al. [12] present the novel SDC architectures to reduce 50% complex adders by reordering inner data sequences. However, the utilization of the corresponding complex multipliers still remains 50% for the both architectures. We therefore study whether the complex multiplier unit can be modified to achieve the 100% utilization. In the radix-2 FFT architectures, there is a common observation that one half data (sum part of butterfly operation) do not involve complex multiplication (W0 N) at all, while the other half (difference part) indeed involves complex multiplication (Wk N). Hence, it has the opportunity to achieve the objective that reduces the arithmetic resource of the conventional complex multipliers by a factor of 2, leading to 100% utilization. It is ideal for two consecutive complex input data to contain a complex number, which needs to execute complex multiplication. If so, we can minimize the reordering memory requirement while achieving the above objective that reduces 50% the arithmetic resource of complex multipliers. Fortunately, the improved SDC architecture can produce the sum and the corresponding difference results of a butterfly operation in consecutive two cycles. The sum part is directly passed to the next stage, while the difference part needs to execute complex multiplication before passing to the next stage. Therefore, the SDC architecture is ideal for our efficient pipelined radix-2 FFT architecture. However, the SDF architecture does not meet the above constraint well since the sums of the all butterflies in the stage are produced first, followed by the corresponding differences. In this brief, we present an efficient combined SDC-SDF radix-2 pipelined FFT architecture, which includes log2 N − 1 SDC stages, 1 SDF stage, and 1 bit reverser. The SDC processing engine (SDC PE) in each SDC stage achieves the 100% hardware resource utilizations of both adders and multipliers. We include the SDF stage to reorder the data sequence, and then the delay memory of the bit reverser is reduced to N/2. The proposed architecture can produce the same normal output order as [26].

REVIEW OF PIPELINED FFT IMPLEMENTATIONS

Assuming that the input data enters the FFT circuit serially in a continuous flow, the radix-2 MDC and SDF architectures can be directly deduced according to the DFG in Fig. 1. The radix-2 MDC architecture is the most direct implementation approach of pipelined FFT, but its hardware utilization is only 50%. Comparedwith, the radix-2 SDF design reduces the required memory size. However, the utilizations of adders and multipliers are still 50%. Figure1 DFG of DIF radix-2 FFT (N=16) Besides the basic radix-2 architectures, various high-radix pipelined FFT architectures have also been proposed to address the arithmetic resource utilization problem. They are radix-4 MDC radix-4 SDC, radix-4 SDF], radix-2 2 SDF, radix-2 3 SDF, radix-2 4 SDF, radix2 5 SDF, radix-r k SDC/SDF, and radix-2 k feedforward. Compared with the radix-2 architectures, the high radix architecture can only process the FFT, whose size is a power of its high radix, not just 2. In order to extend the application scope of the FFT architectures, the new dynamic data scaling architectures for pipelined FFTs have been proposed to implement both 1-D and 2-D applications. The MDC-based FFT architecture has been proposed for the MIMO-OFDM systems with variable length. Employing folding transformation and register minimization techniques, the novel parallel pipelined architecture for complex and real valued FFT has been proposed to significantly reduce power consumption. For single-input data stream, we propose an efficient combined SDC-SDF radix-2 pipelined FFT architecture, and the proposed SDC PE structure can reduce 50% complex multipliers.

PROPOSED ARCHITECTURE

The proposed FFT architecture consists of 1 prestige, log2N −1 SDC stages, 1 post-stage, 1 SDF stage, and 1 bit reverser, shown in Fig. 5.2 (a). The pre-stage shuffles the complex input data to a new sequence that consists of real part followed by the corresponding imaginary part, shown in Table I.



Cycle	pre-stage	stage 1	stage 2	stage 3	post-stage
1	0_r, 1_r	-	-	-	-
2	0_i, 1_i	-	-	-	-
3	2_r, 3_r	-	-	-	-
...	...	...	...	...	...
9	8_r, 9_r	-	-	-	-
10	8_i, 9_i	0_r, 8_r	-	-	-
11	10_r, 11_r	0_i, 8_i	-	-	-
12	10_i, 11_i	2_r, 10_r	-	-	-
13	12_r, 13_r	2_i, 10_i	-	-	-
14	12_i, 13_i	4_r, 12_r	-	-	-
15	14_r, 15_r	4_i, 12_i	0_r, 4_r	-	-
16	14_i, 15_i	6_r, 14_r	0_i, 4_i	-	-
17	-	6_i, 14_i	2_r, 6_r	-	-
18	-	1_r, 9_r	2_i, 6_i	0_r, 2_r	-
19	-	1_i, 9_i	8_r, 12_r	0_i, 2_i	0_r,0_i
20	-	3_r, 11_r	8_i, 12_i	4_r, 6_r	2_r,2_i
21	-	3_i, 11_i -	10_r, 14_r	4_i, 6_i	4_r,4_i
...	...	...	...	...	...
32	-	-	-	13_r, 15_r	11_r, 11_i
33	-	-	-	13_i, 15_i	13_r, 13_i
34	-	-	-	-	15_r, 15_i

Table 1: Data output order of the proposed pipelined architecture for 16 point FFT

The corresponding post-stage shuffles back the new sequence to the complex format. The SDC stage t (t =1,2,...,log 2N −1) contains an SDC PE, which can achieve100% arithmetic resource utilization of both complex adders and complex multipliers. The last stage, SDF stage, is identical to the radix-2 SDF, containing a complex adder and a complex subtracted. By using the modified addressing method, the data with an even index are written into memory in normal order, and they are then retrieved from memory in bit-reversed order while the ones with an odd index are written in bit reversed order. Final, the even data are retrieved in normal order. Thus, the bit reverser requires only N/2 data buffer. Table 1 illustrates the inner data sequence of 16- point FFT computation. The complex input data at cycle m are (m−r, m−i), where m−r and m−i (m=0,1,...,15) represent the real and imaginary parts, respectively. We only include the pre-stage, SDC stage 1, 2, 3, and post-stage, since the SDF stage has the same sequence as the post stage except the 8-cycle delay, and the bit reverser, 8-cycle delay over the SDF stage, produces normal output sequence. The SDC PE, shown in Figure 2 (b), consists of a data commutator, a real add/sub unit, and an optimum complex multiplier unit. In order to minimize the arithmetic resource of the SDC PE, the most significant factor is to maximize the arithmetic resource utilization via reordering the data sequences of the above three units.

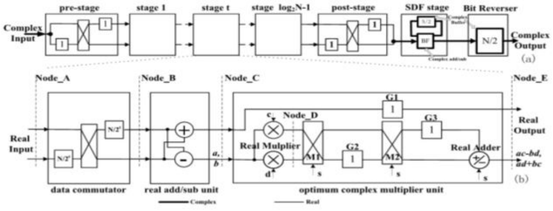
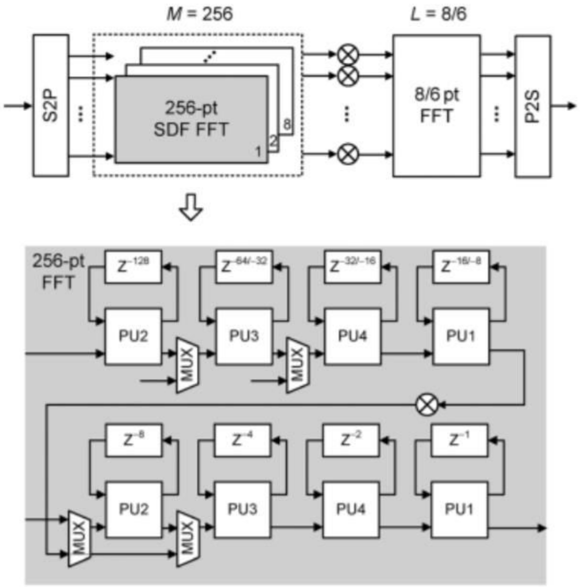


Figure 2: Representing the Reconfigurable Block Diagram for FFT Structures

A reconfigurable FFT processor using SDF architecture is designed and implemented which is can b used as FFT core in 3GPP LTE standard. The proposed architecture satisfies the requirements of 3GPP LTE wireless standard with reduced area and power. Starting from FFT decomposition and architecture parallelism, the maximum-size 2048-point FFT is decomposed into M=256 and L=8 to achieve minimum power area product The 8-path 256-point SDF FFT architecture is shown in Fig.3. It can support 16 to 256 points by reconfiguring the data-path inter-connection between the PUs. To support 1536 points, a 6-point FFT module is constructed by sharing hardware resources with the 8- point FFT.



Proposed architecture is reconfigurable 128-2048 point FFT architecture as shown in fig 3. The method called architecture parallelism is used to achieve this. In architecture parallelism an N point FFT is divided into M and L point FFT (N=M*L). For P way parallel architecture, first stage comprises of P M- point FFTs and second stage with P L- point FFTs. When P=L, the single input SDF FFTs can be combined into a single Linput parallel FFT. The first stage pipelined 256 point FFT is reconfigurable to support 16-256 points. The second stage parallel FFT support 8 or 6 points. The overall FFT meets the 3GPP-LTE standard specification (128, 256, 512, 1024, 1536, 2048 points).

The reordering process is performed as follows.

1) In the first cycle, when 8−rcomes, the signals(s=1) selects“through”; that is, the up (down) input of the multiplexer (M1 orM2) connects to the up (down) output. Then, the G2 (or G3) would bed×8−r (or c×8−r) in the second cycle.

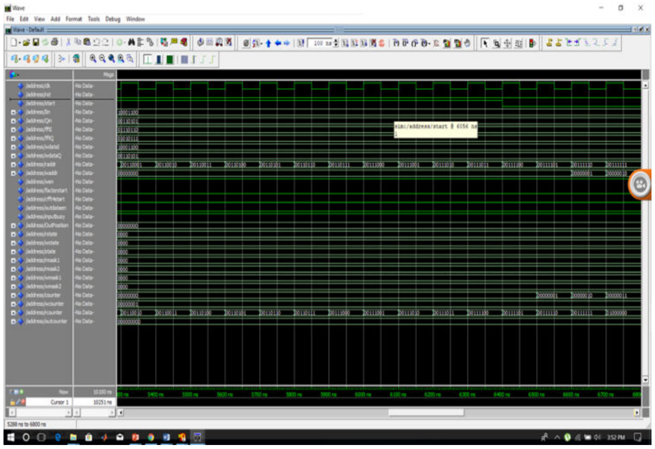
2) In the second cycle, when 8−i comes, the signal s (s =0) selects “swap”; that is, the up (down) input of the multiplexer (M1orM2) connects to the down (up) output. Then, the G2 (or G3) would be c×8−i (ord×8−r) in the third cycle. The swill make the Real Adder perform subtraction operation and then c ×8−r−d×8−i(8−r*) would appear at the Node−E.

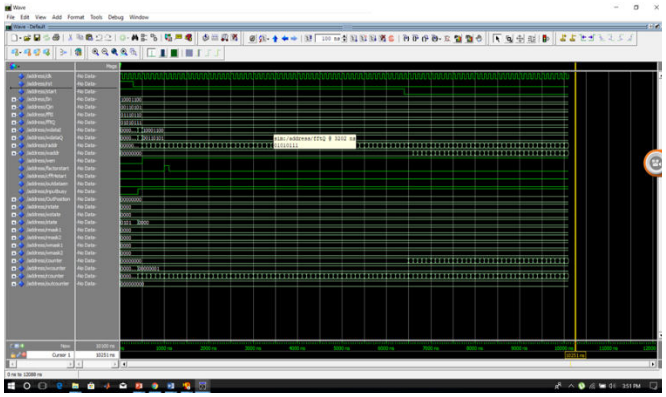
3) In the third cycle, the signal s (s =1) selects “through” for M1and M2,and chooses addition operation for Real Adder. Then,d×8−r+c×8−i (8−i*) would appear at the Node−E.

Consequently, the complex result data couple (0−r*,8−r*) and (0−i*,8−i*) would come out at New−Label (Node−E) with one clock delay in consecutive two cycles. The above mechanism can be iterated by applying to the other couples in the stage 1, e.g., (2−r, 10−r) and (2−i, 10−i), and so on. If we carry the above process toward the log2N −1stages to completion, we can complete the majority part of the radix-2 FFT computation. The SDC PE can reduce 50% the arithmetic resource of complex multipliers in the time-multiplexing approach, at the expense of 1.5 complex delay memory overhead for each SDC PE.

SIMUALTION RESULTS:

The various block in the proposed architecture is coded using Verilog program. Few block simulation is shown below. The most important block of pre stage simulation result is shown in figure 4 &5.





CONCLUSION

A combined SDC-SDF pipelined FFT architecture which produces the output data in the normal order. The proposed SDC PE mainly reduces 50% complex multipliers, compared with the other radix-2 FFT designs. Therefore, the proposed FFT architecture is very attractive for the singlepath pipelined radix-2 FFT processors with the input and output sequences in normal order.

REFERENCES

[1] L. J. Cimini, “Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing,” IEEE Trans. Commun., vol. 33, no. 7, pp. 665–675, Jul. 1985. [2] J. M. Cioffi, The Communications Handbook. Boca Raton, FL, USA: CRC Press, 1997.

[3] Y. W. Lin, H. Y. Liu, and C. Y. Lee, “A 1-GS/s FFT/IFFT processor for UWB applications,” IEEE J. Solid-State Circuits, vol. 40, no. 8, pp. 1726– 1735, Aug. 2005.

[4] C. Cheng and K. K. Parhi, “High throughput VLSI architecture for FFT computation,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 54, no. 10, pp. 339–344, Oct. 2007.

[5] S. N. Tang, J. W. Tsai, and T. Y. Chang, “A 2.4-GS/s FFT processor for OFDM-based WPAN applications,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 57, no. 6, pp. 451–455, Jun. 2010.

[6] Y. Jung, H. Yoon, and J. Kim, “New efficient FFT algorithm and pipeline implementation results for OFDM/DMT

applications," IEEE Trans. Consum. Electron., vol. 49, no. 1, pp. 14–20, Feb. 2003.

[7] M. Shin and H. Lee, "A high-speed, four-parallel radix-24 FFT processor for UWB applications," in Proc. IEEE ISCAS, May 2008, pp. 960–963.

[8] J. H. McClellan and R. J. Purdy, "Applications of digital signal processing to radar," in Applications of Digital Signal Processing. Englewood Cliffs, NJ, USA: Prentice-Hall, 1978, ch. 5.

[9] L. R. Rabiner and B. Gold, Theory and Application of Digital Signal Processing. Englewood Cliffs, NJ, USA: Prentice-Hall, 1975, pp. 604–609.

[10] E. H. Wold and A. M. Despain, "Pipeline and parallel-pipeline FFT processors for VLSI implementation," IEEE Trans. Comput., vol. C-33, no. 5, pp. 414–426, May 1984. [11] Y. N. Chang, "An efficient VLSI architecture for normal I/O order pipeline FFT design," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 55, no. 12, pp. 1234–1238, Dec. 2008

[12] X. Liu, F. Yu, and Z. K. Wang, "A pipelined architecture for normal I/O order FFT," J. Zhejiang Univ. Sci.C, vol. 12, no. 1, pp. 76–82, Jan. 2011.