International Journal of Research
Available at https://pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 21
October 2018

A Normal I/O Order Radix-2 Fft Architecture to Process Two Data Streams

K.Chandra Obul Reddy & D.C.Vinod Rathna Kumar
[1]PG Student, Dept of ECE (VLSI), GCET, Kadapa, AP, India.
[2]Associate Professor, Dept of ECE, GCET, Kadapa, A.P, India.

Abstract- *Nowadays, many applications require simultaneous computation of multiple independent fast Fourier transform (FFT) operations with their outputs in natural order. Therefore, this brief presents a novel pipelined FFT processor for the FFT computation of two independent data streams. The proposed architecture is based on the multipath delay commutator FFT architecture. It has an N/2-pointdecimation in time FFT and N/2-point decimation in frequency FFT to process the odd and even samples of two data streams separately. The bit reversal operation is performed by the shift registers in the FFT architecture by interleaving the data. Therefore, the proposed architecture requires a lower number of registers and has high throughput.*

***Index Terms*—Bit reversal, bit reversed order, fast Fourier transform (FFT), multipath delay commutator (MDC) FFT, normal order.**

I. INTRODUCTION

Fast fourier transform (FFT) is one of the most commonly used operations in the wireless communication applications, such as orthogonal frequency division multiple (OFDM) accesses, ultrawide-band, digital video broadcast terrestrial, and signal processing application as well. A family of pipelined FFT architectures is discussed in which single-path delay feedback (SDF) and multipath delay commutator (MDC) are very popular. There are applications, such as image processing, array signal processing, multiple-input–multiple-output OFDM, and so on, in which more than one data stream need to be processed. Therefore, simultaneous multiple FFT operations are required and a dedicated bit reversal circuit is also required generating the outputs in natural order.There are FFT architectures, which can handle multiple independent data streams. However, all the data streams are processed by a single FFT processor. In four independent data streams are processed one by one. Similarly, eight data streams are processed at two domains. Thus, the outputs of multiple data streams are not available in parallel. In order to simultaneously process the data streams, more than one FFT processors need to be employed. In one to four data streams are processed using multiple data paths for wireless local area network application. Data of different data streams are interleaved to process them simultaneously. Nevertheless, the architectures do not have any specific bit reversal circuit.

Certain circuits are proposed to reorder the bit reversed FFT output into normal order. The bit reversal circuits are proposed for different radices. Similar structure is proposed for variable length FFT, whose register complexity is N. These circuits are suitable for bit reversing the data from the pipelined FFT architecture. However, only the bit reversal structures are proposed. In later works the bit reversal circuit is integrated to FFT architecture; as a result, the total register requirement of the design is reduced from $5N/2$ to $2N$. Two-, four-, and eight-parallel pipelined radix-2^k decimation in frequency (DIF) feed forward FFT architectures are proposed and they need extra N registers to generate the output in natural order. Moreover, these two-, four-, and eight-parallel FFT architectures can start its operation only when $x(n+N/2)$, $x(n+3N/4)$, and $x(n+7N/8)$samples arrive, respectively. Therefore, hardware is underutilized and additional registers are required to store the first $N/2$, $3N/4$, and $7N/8$ samples.

In modified MDC FFT architectures with a new data scheduling method and a rearranging structure are proposed in which the drawbacks of are eliminated. The architectures operated at the frequency of incoming sample rate but the architecture operates half the clock frequency to generate the same throughput. Thus, the throughput of the architecture in is doubled, if all the architectures are operated at the same speed. Similarly, combined single path delay commutator-SDF FFT architecture with I/O in natural order is proposed in which the bit reversal is carried out only with $N/2$ registers. However, its throughput is low and the required number of register is high. In low complexity FFT architectures are proposed but these architectures can process only real-valued signals (signals only with real part). Moreover, they generate two outputs per clock cycle and these outputs are not in natural order. Thus, most of the recent architectures require bit reversal structures to generate the outputs in natural order.

II. PREVIOUS WORK

International Journal of Research
Available at https://pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 21
October 2018

The throughput of the architecture is doubled, if all the architectures are operated at the same speed. Similarly, combined single path delay commutator-SDF FFT architecture with I/O in natural order is proposed in which the bit reversal is carried out only with$N/2$ registers. However, its throughput is low and the required number of register is high. Low complexity FFT architectures are proposed but these architectures can process only real-valued signals (signals only with real part). Moreover, they generate two outputs per clock cycle and these outputs are not in natural order. Thus, most of the recent architectures require bit reversal structures to generate the outputs in natural order.

III. PROPOSED PIPELINED FFT ARCHITECTURE

The idea of computing an N -point FFT using two $N/2$-point FFT operations with additional one stage of butterfly operations is shown in Fig. 1, which is not the exact architecture but provides the methodology.

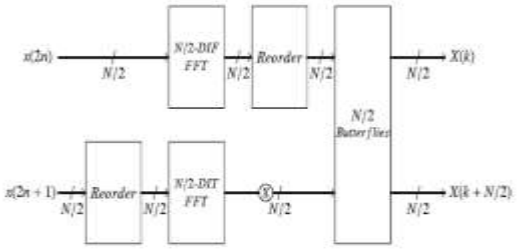

Fig:1. Idea of the proposed method.

The reordering blocks in Fig. 1 are merely present to state that the $N/2$ odd samples ($x(2n+1)$) are reordered before the $N/2$-point DIT FFT operation and $N/2$ even samples ($x(2n)$) are reordered after the $N/2$-point DIF FFT operation. In order to compute the N -point DIT FFT from the outputs of two $N/2$-point FFTs, additional one stage of butterfly operations are performed on the results of the two FFTs. Thus, the outputs generated by additional butterfly stage are in natural order.

For the purpose of simplicity, the proposed 16-point FFT architecture in Fig. 2 is explained. It has two eight-point MDC FFT architectures to process two data streams. The delay commutator units present at the left side of SW_1 dissociate the odd and even samples. The shift registers in the delay commutator units, which receive inputs, are used to bit reverse the odd input samples. These shift registers are called reordering shift registers (RSRs). The RSR in the last stage store outputs from the eight-point DIF FFT and bit reverses them. The BF2 carries out two-parallel butterfly operations between the bit reversed data in the RSR in the last stage and outputs from the eight-point DIT FFT. Thus, the upper and lower BF2 in the last stage generate the FFT outputs of the first and the second data streams in normal order. The two data paths from SW_2 are combined together, so the word length of the data path in last stage is twice and so thick lines are used for representing the data paths and registers.

A. Operation of the Proposed Architecture

The FFT architecture in Fig. 2 is divided into six levels ($L1$, $L2$, $L3$, $M1$, $M2$, and $M3$). The RSR registers in the levels L_1 and M_1 reorder the odd input data and the RSR registers in the levels L_3 and M_3 reorder the partially processed even data. The eight-point DIF and DIT FFT operations are performed in the levels $L_2$and M_2, respectively. The data from $L_1$and M_1 can be forwarded to L_2 and M_2, respectively, or vice versa with the help of SW_1. Similarly, the data from L_2 and M_2 can be forwarded to L_3 and M_3, respectively, or vice versa with the help of SW_2

The two input streams to the FFT processor are represented as X_1 and X_2. The odd and even samples of two input streams are disassociated by the delay commutator units in L_1 and M_1 (X_1 is disassociated into $\{E_1(i,j),O_1(i,j)\}$, respectively, and X_2 is disassociated into $\{E_2(i,j),O_2(i,j)\}$). In these representations, I define the nature of the data and j defines the number of thedata set whose FFT has to be computed. The even set of input data $[x(0),x(2),x(4)\ldots]$ is defined as $E(1,j)$ and the odd set of input data $[x(1),x(3),x(5)\ldots]$ is defined as $O(1,j)$. $E(2,j)/O(2,j)$ is the set of scheduled or ordered even/odd data, which are ready to be fed to eight-point DIF/DIT FFT. The outputs of eight-point DIF/DIT FFT are defined as $E(3,j)/O(3,j)$, which are fed to the third level for 16-point FFT computation. 1) The first eight samples of $X1$ are loaded into the registers in $L1$. After eight clock cycles, the switch (SW1) is set in the normal mode and the first eight samples of $X2$ are loaded into the registers ($4D$) in $M1$. Simultaneously, $E1(1, 1)$ (even samples of $X1$) is forwarded from $L1$ to $L2$ as $E1(2, 1)$ to perform the eight-point FFT operation. The odd samples of $X1$ and $X2$ are bit reversed by the RSR in $L1$ and $L2$, respectively.2) After eight clock cycles, the positions of the switches SW1 and SW2 are set in the swap mode and the normal mode, respectively. The odd samples ($O1(1, 1)$) of $X1$ are forwarded from $L1$ to $M2$ as $O1(2, 1)$ and the even samples ($E2(1, 1)$) of $X2$ is forwarded from $M1$ to $L2$ as $E2(2, 1)$. Simultaneously, $E1(2, 1)$ is forwarded from $L2$ to $L3$ as $E1(3, 1)$ and reordering is performed.3) After eight clock cycles, SW1 and SW2 are set in the normal mode and the

International Journal of Research
Available at https://pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 21
October 2018

swap mode, respectively. The odd samples of $X2(O2(1, 1))$ are forwarded from $M1$ to $M2$ as $O2(2, 1)$ and $O1(2, 1)$ is forwarded from $M2$ as $O1(3, 1)$ to $L3$ where the butterfly operations with $E1(3, 1)$ corresponding to the last stage (of the data stream $X1$) are performed. In the meantime, $E2(2, 1)$ from $L2$ is forwarded to $M3$ as $E2(3, 1)$and reordering is performed in the RSR.4) After eight clock cycles, the switch (SW_2) is set to normal position to allow the partially processed odd samples ($O_2(3, 1)$) from M_2 to M_3 and perform the butterfly operations of the last stage (of the data stream X_2).

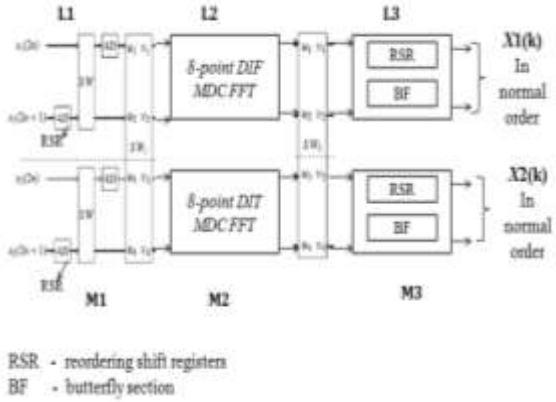

Fig:2. Proposed 16-point radix-2 FFT architecture with outputs in natural order.

B. ADVANTAGES AND APPLICATIONS

Advantages:

- Improvement in performance
- High throughput
- Less area

Applications:

- Wireless communication
- Signal processing applications

IV. SIMULATION RESULTS

These Waveforms are obtained from the xilinx Simulator by simulating Verilog HDL codes of respective Architectures

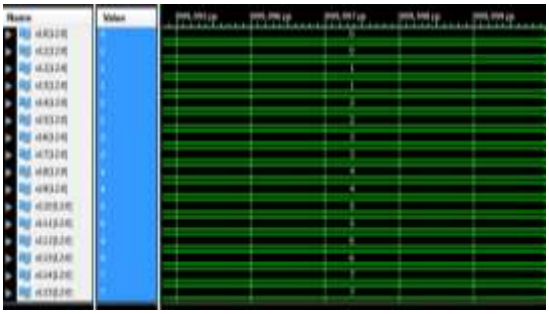

Fig:3. first stream inputs

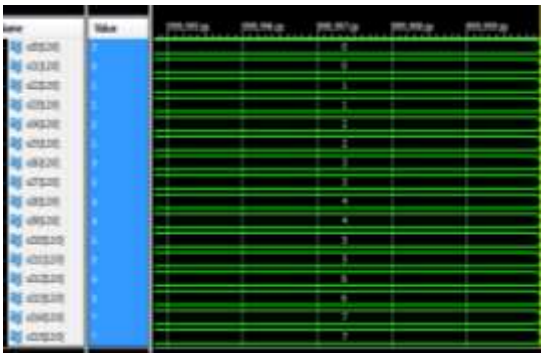

Fig:4. second stream inputs

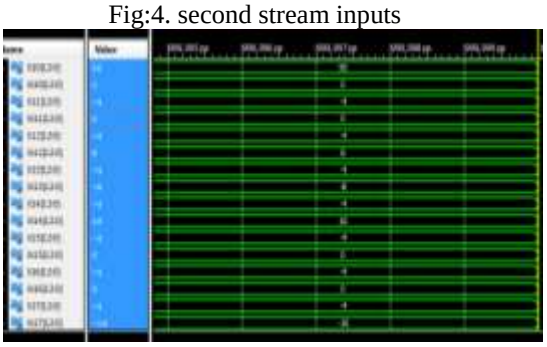

Fig:5 first stream outputs

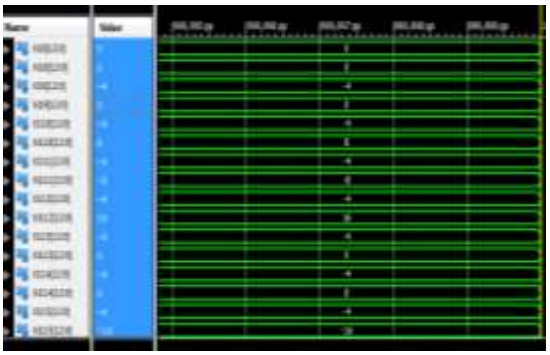

Fig6. first stream outputs

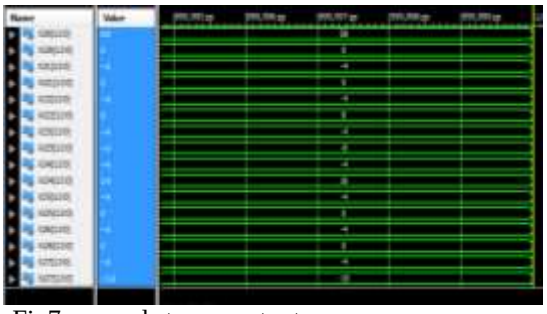

Fig7. second stream outputs

International Journal of Research
Available at https://pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 21
October 2018

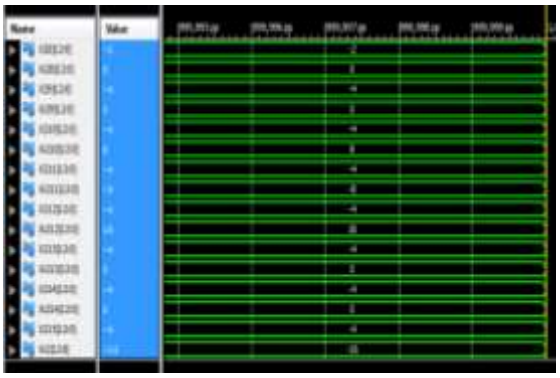

Fig8. second stream outputs

Performance evaluation

	FFTS	AREA	H/W UTILIZATION
EXISTING	TWO 16 POINT	MORE	<100%
PROPOSED	TWO 8 POINT	LESS	100%

Total latency= 16.18 ns

V. CONCLUSION

This brief has presented a novel FFT processor whose outputs are generated in the natural order. The proposed processor can process two independent data streams simultaneously, and makes it suitable for many high-speed real-time applications. Moreover, the proposed architecture provides throughput higher than the prior architectures. These attributes make the proposed FFT processor superior in sense of hardware complexity and performance.

VI. REFFERENCES

[1] S. He and M. Torkelson, "A new approach to pipeline FFT processor," in *Proc. 10th Int. Parallel Process. Symp.*, 1996, pp. 766–770.

[2] Y. Chen, Y.-W.Lin, Y.-C.Tsao, and C.-Y. Lee, "A 2.4-Gsample/s DVFS FFT processor for MIMO OFDM communication systems," *IEEE J. Solid-State Circuits*, vol. 43, no. 5, pp. 1260–1273,May 2008.

[3] S.-N. Tang, C.-H.Liao, and T.-Y. Chang, "An area- and energy-efficient multimode FFT processor for WPAN/WLAN/WMAN systems," *IEEE J. Solid-State Circuits*, vol. 47, no. 6, pp. 1419–1435,Jul. 2012.

[4] P. P. Boopal, M. Garrido, and O. Gustafsson, "A reconfigurable FFT architecture for variable-length and multi-streaming OFDM standards," in *Proc. IEEE ISCAS*, May 2013, pp. 2066–2070.

[5] K.-J. Yang, S.-H. Tsai, and G. C. H. Chuang, "MDC FFT/IFFT processor with variable length for MIMO-OFDM systems," *IEEE Trans.Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 4, pp. 720–731,Apr. 2013.

[6] 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.

[7] M. Ayinala, M. Brown, and K. K. Parhi, "Pipelined parallel FFT architectures via folding transformation," *IEEE Trans. VeryLarge Scale Integr. (VLSI) Syst.*, vol. 20, no. 6, pp. 1068–1081,Jun. 2012.

[8] M. Garrido, J. Grajal, and O. Gustafsson, "Optimum circuits for bit reversal," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 58, no. 10, pp. 657–661, Oct. 2011.

[9] S.-G. Chen, S.-J. Huang, M. Garrido, and S.-J. Jou, "Continuous-flow parallel bit-reversal circuit for MDF and MDC FFT architectures," *IEEETrans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 10, pp. 2869–2877,Oct. 2014.

[10] M. Garrido, J. Grajal, M. A. Sanchez, and O. Gustafsson, "Pipelined radix-2^k feedforward FFT architectures," *IEEE Trans. Very Large ScaleIntegr. (VLSI) Syst.*, vol. 21, no. 1, pp. 23–32, Jan. 2013.

[11] Z. Wang, X. Liu, B. He, and F. Yu, "A combined SDC-SDF architecture for normal I/O pipelined radix-2 FFT," *IEEE Trans. Very Large ScaleIntegr. (VLSI) Syst.*, vol. 23, no. 5, pp. 973–977, May 2015.

[12] A. X. Glittas and G. Lakshminarayanan, "Pipelined FFT architectures for real-time signal processing and wireless communication applications," in *Proc. 18th Int. Symp. VLSI Design Test*, Jul. 2014, pp. 1–2.