ISSN XXXX XXXX © 2017 IJESC

Research Article Volume 7 Issue No.8

Design and Implementation of Parallel Bit Reversal on FFT by using Verilog HDL

Ravi L.S[1], Chithra .C .P [2], Farzana Parveen B.C[3]
Assistant Professor[1, 3], Student[2]
Department of ECE
RIT, Hassan, Karnataka, Indi[a1, 2]
SJMIT, Chitradurga, Karnataka, India[3]

Abstract:
Bit-reversal is an essential part of the fast Fourier transform. However, compared to the amount of works on FFT architectures, much fewer works are dedicated to bit-reversal circuits until recent years. In this brief, the minimum latency and memory required for calculating the bit-reversal of continuous-flow parallel data are formulated. The proposed circuit is simple and efficient for reordering the output samples of parallel pipelined FFT processors. The proposed approach can be Implemented using Verilog HDL and Simulated by Modelsim 6.4 c. Finally it's synthesized by Xilinx tool.

Keywords: FFT, Commutators, S-box, memory, Buffer group, radix 2.

I.INTRODUCTION

The fast Fourier transform is one of the most important and fundamental algorithms in the digital signal processing area. It is a fast computational algorithm to implement the discrete Fourier transform and it has been widely used in various applications such as spectrum analysis, wireless communications and imaging systems. Many variants of the FFT algorithm have been developed, such as radix-2 and radix-4 FFT. Since arithmetic operations significantly contribute to overall system power consumption, SRFFT is a good candidate for the implementation of a low-power FFT processor. The FFT computation requires an indexing scheme at each stage to address input/output data and coefficient multipliers properly. When the fast Fourier transforms is executed using an in-place method, the input or output data must be accessed in a digit-reversed order. The reversal procedure rearranges the address sequence used to access data by reordering the bits of a binary representation of the address according to the radix of the FFT. The bit reordering procedure can be realized in firmware, software or hardware.

II.EXISTING SYSTEM

Several bit-reversal circuits for parallel pipelined FFT architectures have also been presented. A bit-reversal circuit for 8-parallel data is proposed, but the complexity of the circuit is high. For the P-parallel N-point FFT architecture, a total memory of N x N=P is enough to carry out the reordering task. The reordering circuit is designed for a specific FFT and uses slightly more than N memory words. The first circuit calculates the bit-reversal of parallel data using a total memory of N. However, none of the existing works have derived the minimum latency and memory required for calculating the parallel bit-reversal.

III. PROPOSED SYSTEM

FFT algorithms are basically called Fast Fourier Transform algorithms and it is important in the area of DSP which is used to calculate DFT efficiently. DSP processors have special architectural provisions to implement FFT algorithms efficiently. FFT is extensively utilized in the areas of DSP such as filtering, spectral analysis, etc. FFT plays a significant task in present digital communication systems for instance DVB and OFDM systems. For the sequence x(n), N point DFT can be expressed by equation

$$X[K] = \sum_{n=0}^{N-1} x(n) W_N^{nk} \qquad k = 0,1,\ldots N-1$$

Where,
X (k) = frequency-domain sequences
x (n) = time-domain sequence
W_N^{nk} =Twiddle factor

$$W_N^{nk} = e^{-j\frac{2\pi}{N}kn}$$
$$= \cos\left(\frac{2\pi}{N}kn\right) - j \sin\left(\frac{2\pi}{N}kn\right)$$

The essential computational component of FFT is called butterfly. Every butterfly unit needs complex multiplications and also complex additions. For the computation of FFT, decomposition of the N-point into repetitive micro operations is done. This is known as butterfly operation. During hardware 0 implementation of FFT, if a individual butterfly structures is implemented on the chip, this butterfly unit will perform all the computation recursively. The DFT calculation demands a complex implementation. The 0size0 butterfly is 'r', the resulting FFT is known as 'radix-r' FFT. In the Fig 2.2 when the inputs area and b the output is given by equation.

$$A = a + W'_N b$$

$$B = a - W'_N b$$

The direct estimation of DFT is found to be especially very complicated and because of this reason FFT algorithm is used. To minimize the number of computation FFT algorithm uses a divide and conquer approach recursively.

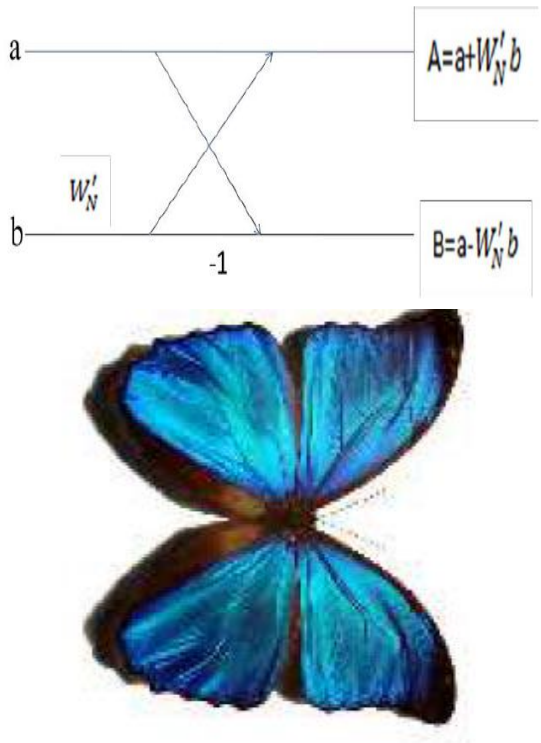

Figure.1. FFT Butterfly diagram

Selecting FFT radix is the first step of the algorithmic level. It is mainly dependent on factors such as speed, power and area designed for different number of transistors. Each of the algorithms is identical in the structural arrangement but differing merely in the core computation of butterfly. Radix-2 is an FFT algorithm within which divide and conquer method is used. The simplest FFT algorithm is Radix-2 algorithm its butterfly structure is shown in the Fig 2.3. The radix-2 algorithm implemented either in time or frequency domains are considered to be one of the simplest algorithms. The value of 'N' be should be chosen so that N= .The resulting DFT obtained after successive decomposition will be of size N=2. Therefore this type of algorithms is known as radix-2 Algorithm, or the radix of this algorithm is 2. The proposed circuit is composed of a sub-bit reversal module, input and output commutator, and a group of P buffer banks. The bit-reversal permutation is accomplished by cascading two sub-permutations, sub-bit-reversal and segments reordering, which are realized by the sub-bit reversal module and buffer group, respectively. These P x P segments are reordered by the P buffer banks, each of which consists of P - 1 buffer cells connected in series. The input and output commutator are also adopted to access the P buffer banks efficiently.

Sub-Bit Reversal Module

The sub-bit-reversal module consists of P bit-reversal circuits, each of which calculates N/P^2-point bit-reversal. The circuit is adopted to implement these N/P^2-point bit-reversal circuits, except that the P circuits share the same multiplexer generator. The N/P^2-point bit reversal circuit is composed by cascading [m/2] basic circuits.

Input and Output Commutators

After been processed by the sub-bit-reversal module, every N/P^2 samples in each path are organized into a segment. As one buffer bank can only accept one input and generate one output at a time, the input and output commutator are adopted. The P-Path input segments are switched to proper buffer banks by the input commutator, and the P buffer banks' outputs are switched to proper output paths by the output commutator.

Buffer Group

The buffer group consists of P buffer banks and one S-generator. Each buffer bank reorders P segments according to the S-generator. The buffer bank is composed of P -1 buffer cells. The buffer length of each buffer cell is N/P^2, leading to a total memory of $N (P-1)/P^2$. The buffer cell bypasses the input to the output when the signal S is greater than the cell number n. Otherwise, it outputs the segment which is stored in the buffer and stores the input segment. As a result, the buffer bank outputs the segment stored in the buffer cell whose number is equal to signal S. The buffer cells whose number is less than the signal S remain the same, and the buffer cells whose number n is greater than the signal S shifts segment to the buffer cell (n - 1).

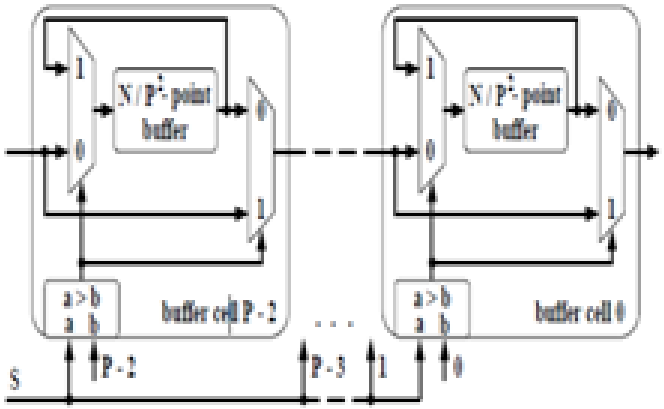

Figure.2. Buffer Bank

S-GENERATOR

As a result, the buffer bank outputs the segment stored in the buffer cell whose number is equal to signal S. The buffer cells whose number is less than the signal S remain the same, and the buffer cells whose number n is greater than the signal S shifts segment to the buffer cell (n - 1). The overall implementation is as shown in Figure 3

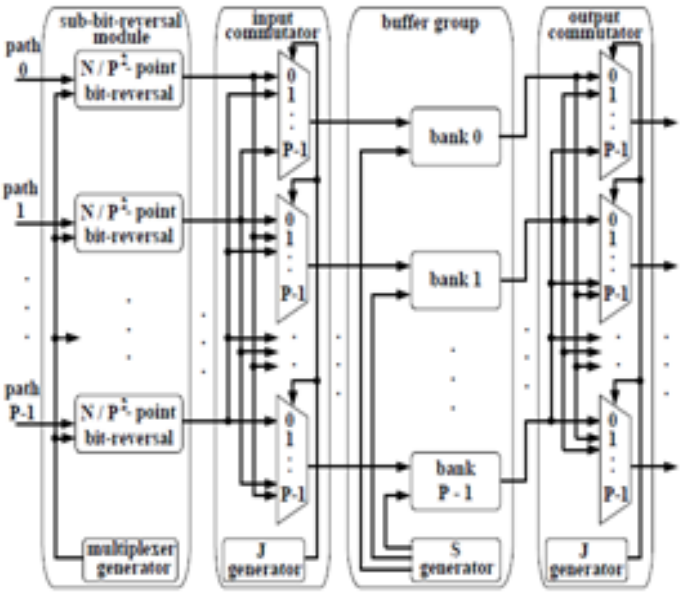

Figure.3.Efficient circuit for P-parallel bit-reversal

IV. RESULTS

PARALLEL BIT-REVERSAL

The overall output waveform of the parallel bit-reversal architecture is shown in Figure 2. Clk and Rst are the control signals. In0 – In7 are the inputs that are given and Out0 - Out7 are the main outputs.

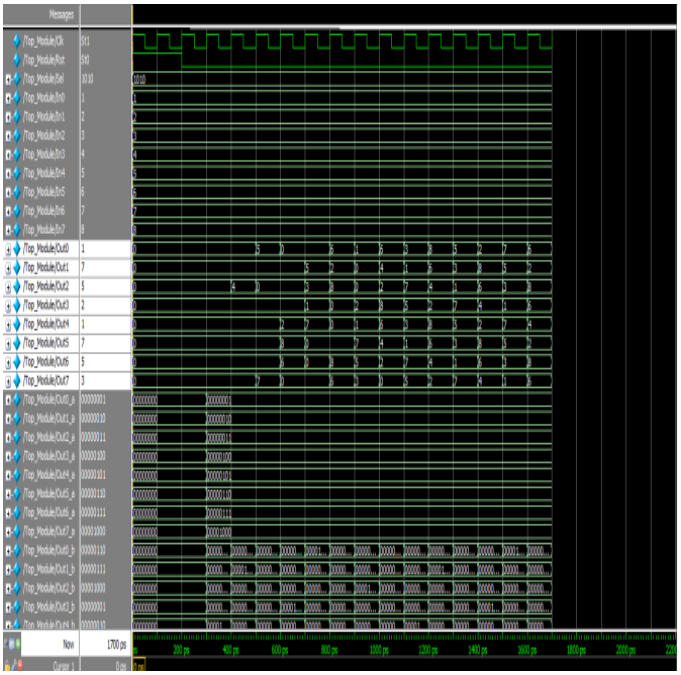

Figure.4. Output snapshot of parallel bit-reversal circuit

The RTL schematic is generated after the optimization and technology targeting phase of the synthesis process. The schematic shows a representation of the design in terms of logic elements optimized to the target Xilinx device or "technology"; for example, in terms of of LUTs, carry logic, I/O buffers, and other technology-specific components. Viewing this schematic allows you to see a technology-level representation of your HDL optimized for a specific Xilinx architecture, which might help you discover design issues early in the design process. The Figure 5 shows the technology schematic of parallel bit-reversal circuit.

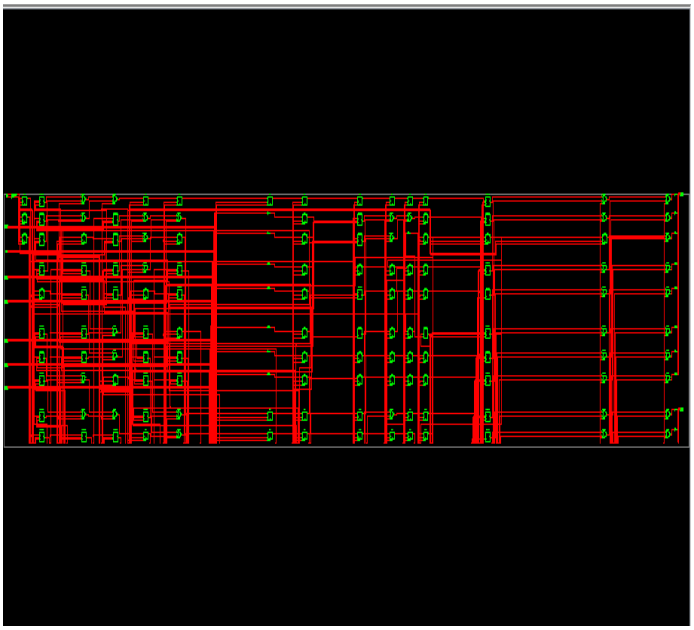

Figure.5. RTL schematic of parallel bit-reversal circuit

BUFFER GROUP

Buffer group is the combination of buffer cells. The buffer group consists of 8 inputs, 8 outputs, 4 bit selection line i.e. Sel and the output of buffer group are shown in Figure 6.

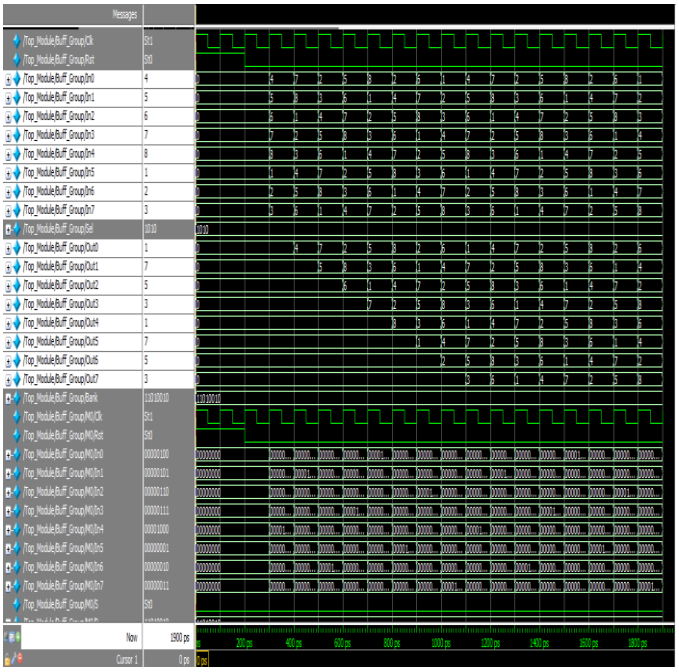

Figure .6. Output snapshot of Buffer group

The buffer group uses S-generator to generate the data and it is then connected to the buffer bank. The RTL schematic of buffer group is shown in Figure 7.

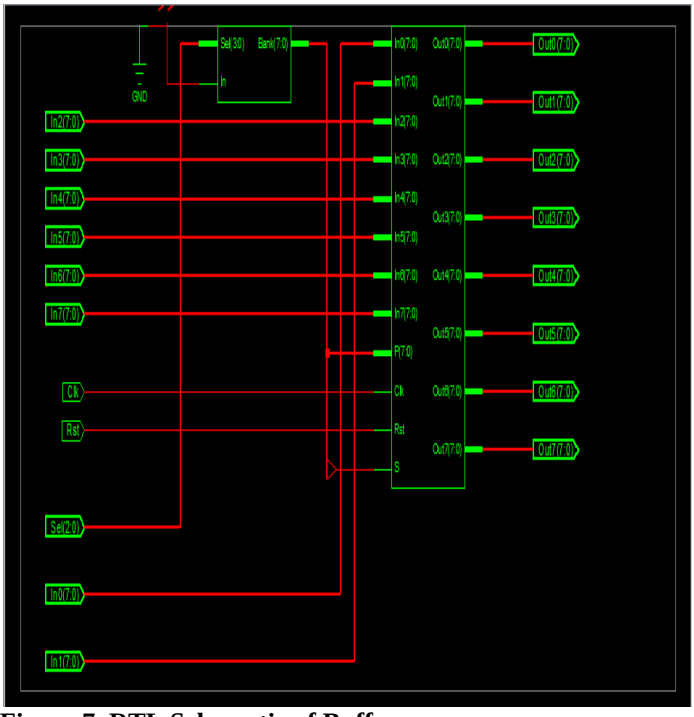

Figure.7. RTL Schematic of Buffer group

SUB-BIT REVERSAL MODULE

The sub-bit reversal module consists of P bit-reversal circuits, each of which calculates N/P^2- point bit reversal. The data which is given should be transferred to the next block which is done in Initial block. The waveform is shown in Figure 8.

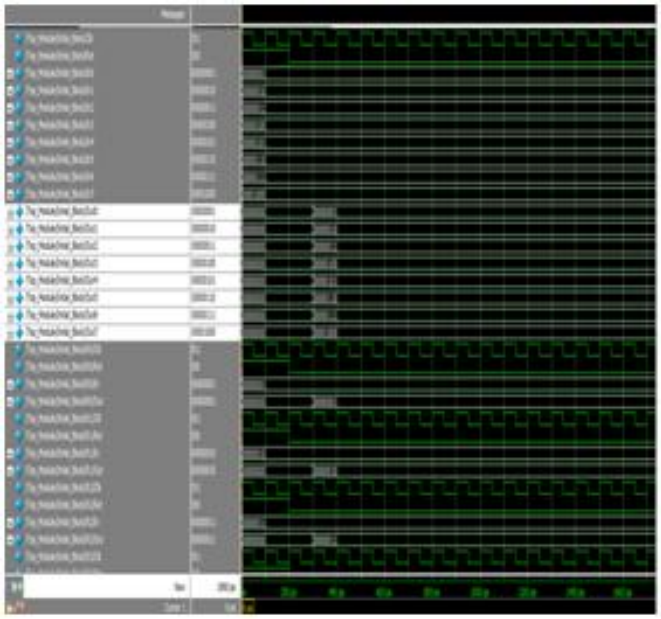

Figure. 8. Output snapshot of point bit-reversal

Sub-bit reversal is the initial block which consists of 8 buffer units. The RTL schematic of the sub-bit reversal module is shown in the Figure 9.

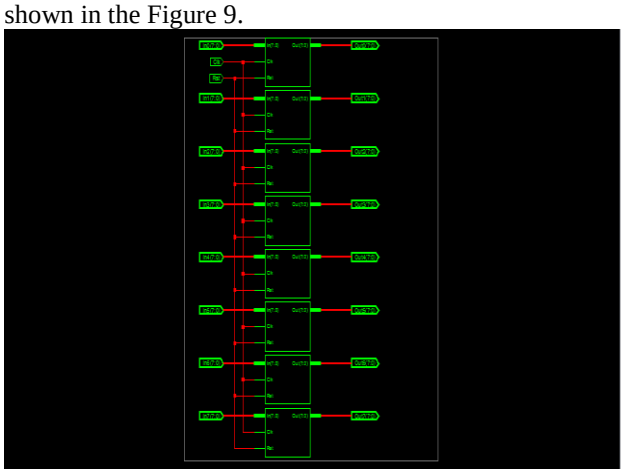

Figure.9.RTL schematic of Sub-bit reversal module

S-GENERATOR

S-generator is the selection line generator for the buffer group. The S-generator has input, selection line and buffer bank as shown in the Figure 10.

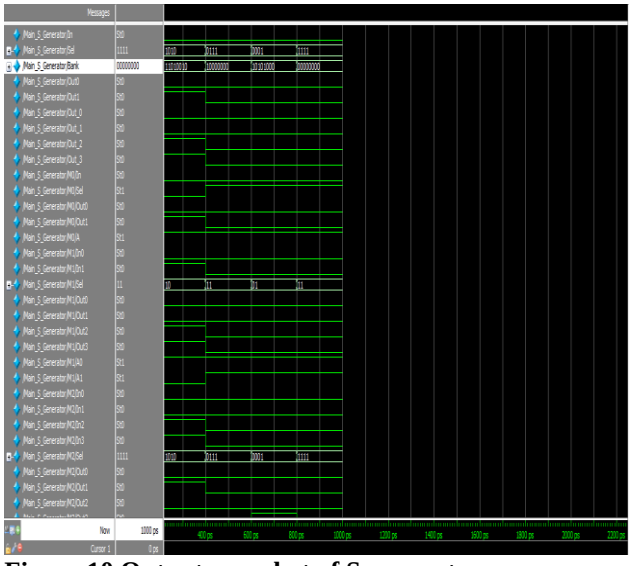

Figure.10.Output snapshot of S-generator

V. ADVANTAGES AND APPLICATIONS

ADVANTAGES

- It uses the minimum memory that is required for calculating the parallel bit-reversal.
- The circuit achieves the minimum latency.

APPLICATIONS

- **Spectrum analysis:**

The analysis of electrical signals, otherwise known as signal analysis, is a fundamental challenge for virtually all electronic design engineers and scientists. While it provides valuable insight into a signal's properties, signal analysis is only as good as the instrument with which it is performed. To fully understand the performance of a system, a signal must also be analyzed in the frequency domain. This is exactly what the spectrum analyzer does.

- **Wireless communications:**

Orthogonal Frequency Division Multiplexing is a popular scheme for high data rate wireless transmission. OFDM may combine with antenna array at the transmitter and receiver to improve the system capacity on frequency selective and time variant channel, resulting in a Multiple Input Multiple Output configuration.

- **Imaging systems:**

The Fourier Transform is an important image processing tool which is used to decompose an image into its sine and cosine components.

V. CONCLUSION

This brief formulates the minimum latency and memory required for calculating the bit-reversal of continuous-flow parallel data. An efficient circuit for parallel bit-reversal is also proposed. The circuit not only achieves the lowest latency but also uses the minimum memory. The proposed architecture is simple and generic. Moreover, the circuit supports continuous flow data and is very suitable for reordering the samples of parallel pipelined FFT processors.

A. Future Work

Along with the minimum latency and memory required for calculating the bit-reversal of continuous-flow parallel data, this design can be further enhanced to reduce the area.

VI. REFERENCES

[1]. Y.-N. Chang, "An efficient VLSI architecture for normal I/O order pipeline FFT design," Circuits and Systems II: Express Briefs, IEEE Trans., vol. 55, no. 12, pp.1234–1238, 2008.

[2]. F. Kristensen, P. Nilsson, and A. Olsson, "Flexible baseband transmitter for OFDM, "in Proceedings of the IASTED International Conference on Circuits, Signals, and Systems (Cancun, Mexico), 2003.

[3]. W. Gao, S. Kwong, and H. Sang, "Low-cost memory data scheduling method for reconfigurable FFT bit-reversal circuits," Electronics Letters, vol. 51, no. 3, pp. 217– 219, 2015.

[4] T. S. Chakraborty and S. Chakrabarti, “On output reorder buffer design of bit reversed pipelined continuous data FFT architecture,” in Proc. IEEE Asia Pac. Conf. Circuits Syst. (APCCAS), 2008, pp. 1132–1135.

[5]. C. Yu, “Flexible and low-complexity bit-reversal scheme for serial-data FFT processors,” Electronics Letters, vol. 51, no. 4, pp. 328–330, 2015.

[6]. M. Garrido, J. Grajal, and O. Gustafsson, “Optimum circuits for bit reversal,” Circuits and Systems II: Express Briefs, IEEE Trans., vol. 58 no. 10, pp. 657–661, 2011.

[7]. S. Yoshizawa, A. Orikasa, and Y. Miyanaga, “An area and power efficient pipeline FFT processor for 8x8 MIMO-OFDM systems,” in Proc. IEEE int. Symp. CircuitsbSyst., 2011, pp. 2705–2708.

[8]. M. Garrido, J. Grajal, M. Sanchez, and O. Gustafsson, “Pipelined radix- 2k feed forward FFT architectures,” Very Large Scale Integration (VLSI) Systems, IEEE Trans., vol. 21, no. 1, pp. 23–32, Jan 2013.

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

[10]. S.-G. Chen, S.-J.Huang, M. Garrido, and S.-J.Jou, “Continuous-flow parallel bit-reversal circuit for MDF and MDC FFT architectures,” Circuits and Systems I: Regular Papers, IEEE Trans., vol. 61, no. 10,pp. 2869–2877, Oct 2014.