

Design of A Pipelined Parallel FFT Architecture Using Radix 8 Algorithm

Prof.S.I.Parihar¹, Prof. G.P.Halde², Prof. S.S.Wasnik³
^{1,2,3}Asst.Professor,P.I.E.T.,Nagpur

Abstract-A formal procedure for designing FFT (Fast Fourier Transform) architectures using radix 8 algorithms is proposed. Novel parallel-pipelined architectures for the computation of delay and power are found out. FFT is used to produce frequency analysis of discrete non periodic signals. FFT is simply a fast way to calculate the discrete Fourier transform. This paper proposes the design of no. of points FFT processing blocks. The work of this project is focused on the design and implementation of FFT. The direct mathematical derivation method is used for this design. In this project the coding is done in VHDL and logic simulation is done using Xilinx ISE Design suite 9.1.

Keywords-XOR, NS-3

I. INTRODUCTION

FFT plays an important role in the analysis, design and implementation of the discrete time signal processing algorithms and systems it is used to convert the samples in time domain to frequency domain. The wide usage of DFT'S in digital signal processing applications is the motivation to implement FFT'S. Almost every branch of engineering and science uses Fourier methods.

The words 'frequency', 'spectrum', 'phase' and 'period', are important parts of an engineer's vocabulary.

Fast Fourier Transform (FFT) is widely used in the field of digital signal processing (DSP) such as filtering, spectral analysis, etc., to compute the discrete Fourier transform (DFT). FFT plays a critical role in modern digital communications such as digital video broadcasting and orthogonal frequency division multiplexing (OFDM) systems. Much research has been carried out on designing pipelined architectures for computation of FFT of complex valued signals (CFFT). Various algorithms have been developed to reduce the computational complexity, of which Cooley-Tukey radix-2 FFT is very popular.

Algorithms including radix-16, radix-32 has been developed based on the basic radix-8 FFT approach. The architectures based on these algorithms are some of the standard FFT architectures in the literature. Radix-2 multi-path delay commutator (R2MDC) is one of the most classical

approaches for pipelined implementation of radix-2 FFT. Efficient usage of the storage buffer in R2MDC leads to the Radix-2 Single-path delay feedback (R2SDF) architecture with reduced memory. R4SDF and R4MDC have been proposed as radix-4 versions of R2SDF and R4MDC, respectively. Radix-4 single-path delay commutator (R4SDC) is proposed using a modified radix-4 algorithm to reduce the complexity of the R4M DC architecture.

The RFFT plays an important role in different fields such as communication systems, biomedical applications, sensors and radar signal processing. A dedicated RFFT processor architecture will reduce the hardware complexity in discrete multi-tone (DMT) based technologies like very high bit-rate digital subscriber line (VDSL) and asymmetric digital subscriber line (ADSL).

In order to meet the high performance and real-time requirements of modern applications, hardware designers have always tried to implement efficient architectures for the computation of the FFT. In this context, pipelined hardware architectures are widely used, because they provide high throughputs and low latencies suitable for real time, as well as a reasonably low area and power consumption. There are two main types of pipelined architectures: feedback (FB) and feed forward (FF). On the one hand, feedback architectures are characterized by their feedback loops, i.e., some outputs of the butterflies are fed back to the memories at the same stage. Feedback architectures can be divided into single-path delay feedback (SDF), which processes a continuous flow of one sample per clock cycle, and multi-path delay feedback (MDF) or parallel feedback, which process several samples in parallel. On the other hand, feed forward architectures, also known as multi-path delay commutator (MDC), do not have feedback loops and each stage passes the processed data to the next stage. These architectures can also process several samples in parallel.

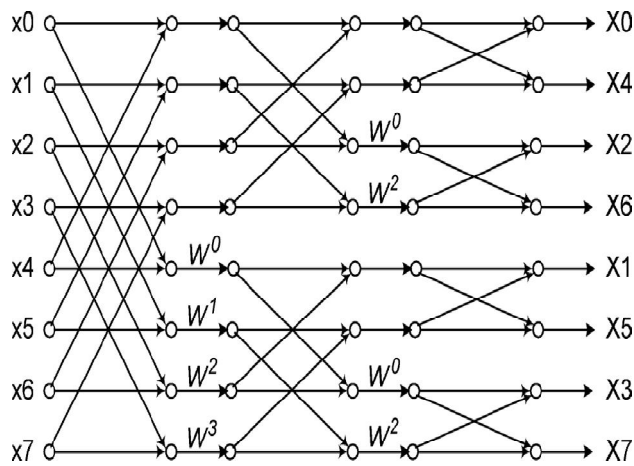


Fig. point FFT architecture

In current real-time applications, the FFT has to be calculated at very high throughput rates, even in the range of Giga samples per second. These high-performance requirements appear in applications such as orthogonal frequency division multiplexing (OFDM), and ultra wide band (UWB). In this context two main challenges can be distinguished. The first one is to calculate the FFT of multiple independent data sequences. In this case, all the FFT processors can share the rotation memory in order to reduce the hardware. Designs that manage a variable number of sequences can also be obtained. The second challenge is to calculate the FFT when several samples of the same sequence are received in parallel. This must be done when the required throughput is higher than the clock frequency of the device. In this case it is necessary to resort to FFT architectures that can manage several samples in parallel.

II. METHODOLOGY

- Study of different algorithm.
 1. Direct Use of the CFFT
 2. Doubling Algorithm
 3. Packing Algorithm
- Selecting particular algorithm for RFFT.
- Design of parallel pipelined DIF FFT and DIT FFT architectures.
- Synthesis and simulation in XILINX 9.1 ISE
- Area, power, delay results for proposed architecture

First radix 8 algorithm is designed. Then using radix 8 algorithm 16 point FFT architecture will be designed. Using radix 8 algorithm one can design up to n point FFT architecture. But in this paper, one can try up to 64 point FFT architecture. After that pipelined mode operations will be performed.

III. LITERATURE REVIEW

Manohar Ayinala, Student Member, IEEE, Michael Brown, and Keshab K. Parhi, Fellow, IEEE proposed an architecture titled “**Pipelined Parallel FFT Architectures via Folding Transformation**” saying that a novel approach to develop parallel pipelined architectures for the fast Fourier transform (FFT). A formal procedure for designing FFT architectures using folding transformation and register minimization techniques is proposed. Novel parallel-pipelined architectures for the computation of complex and real valued fast Fourier transform are derived. For complex valued Fourier transform (CFFT), the proposed architecture takes advantage of utilized hardware in the serial architecture to derive -parallel architectures without increasing the hardware complexity by a factor of L.

Mario Garrido, Member, IEEE, J. Grajal, M. A. Sánchez, and Oscar Gustafsson, Senior Member, IEEE proposed that “**Pipelined Radix-2(n) Feed forward FFT Architectures**” says that the appearance of radix- was a milestone in the design of pipelined FFT hardware architectures. Later, radix- was extended to radix-. However, radix- was only proposed for single-path delay feedback (SDF) architectures, but not for feed forward ones, also called multi-path delay commutator (MDC). This paper presents the radix- feed forward (MDC) FFT architectures. In feed forward architectures radix- can be used for any number of parallel samples which is a power of two. Furthermore, both decimation in frequency (DIF) and decimation in time (DIT) decompositions can be used.

Manohar Ayinala, Member, IEEE, and Keshab K. Parhi, Fellow, IEEE “**FFT Architectures for Real-Valued Signals Based on Radix-2(2) and Radix-2(3) Algorithms**” This paper presents a novel approach to develop pipelined fast Fourier transform (FFT) architectures for real-valued signals. The proposed methodology is based on modifying the flow graph of the FFT algorithm such that it has both real and complex data paths. The imaginary parts of the computations replace the redundant operations in the modified flow graph. New butterfly structures are designed to handle the hybrid data paths. The proposed hybrid data path leads to a general approach which can be extended to all radix- based FFT algorithms. Further, architectures with arbitrary level of parallelism can be derived using the folding methodology. Novel 2-parallel and 4-parallel architectures are presented for radix and radix- algorithms

IV. CONCLUSION

This paper has presented a novel approach to derive the FFT architectures for a given algorithm. The proposed approach can be used to derive partly parallel architectures of any arbitrary parallelism level. Using this approach parallel architectures have been proposed for the computation of complex FFT based on radix-8 algorithms.

REFERENCES

- [1] J. W. Cooley and J. Tukey, "An algorithm for machine calculation of complex Fourier series," *Math.Comput.*, vol. 19, pp. 297–301, Apr.1965.
- [2] K. K. Parhi, C. Y. Wang, and A. P. Brown, "Synthesis of control circuits in folded pipelined DSP architectures," *IEEE J. Solid-State Circuits*, vol. 27, no. 1, pp. 29–43, Jan. 1992.
- [3] K. K. Parhi, "Systematic synthesis of DSP data format converters using lifetime analysis and forward-backward register allocation," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 39, no. 7, pp. 423–440, Jul. 1992.
- [4] K. K. Parhi, "Calculation of minimum number of registers in arbitrary life time chart," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 41, no. 6, pp. 434–436, Jun. 1995.
- [5] K. K. Parhi, *VLSI Digital Signal Processing Systems: Design and Implementation*. Hoboken, NJ: Wiley, 1999.
- [6] C. Wang and K. K. Parhi, "High-level DSP synthesis using concurrent transformations, scheduling, and allocation," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 14, no. 3, pp. 274–295, Mar 1995.
- [7] T. C. Denk and K. K. Parhi, "Exhaustive scheduling and retiming of digital signal processing systems," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process*, vol. 45, no. 7, pp. 821–838, Jul. 1998.