Cordic Based Design for FFT Processor Architecture

Sabna A M
PG Scholar, Dept .of ECE
Muslim Association College of Engineering
Trivandrum, India

Mrs. Suma J John
Assistant professor, Dept .of ECE
Muslim Association College of Engineering

***Abstract*— In communication field, Fast Fourier Transform is one of the most computationally intensive and power hungry modules. Some of the vital applications of the Fast Fourier Transform includes Signal analysis, Sound filtering, Data compression, Image filtering etc. Design of FFT hardware is a challenging task while balancing design parameters such as speed, power, area, flexibility and scalability. The work in this paper proposes scalable radix2 - 8 point FFT processor architecture. The scalability of FFT processor achieved by using ping pong logic. The scalable FFT processor was designed and implemented using VHDL, simulated using Modelsim. The future work about radix4 FFT processor developed using cordic algorithm technique.**

I. INTRODUCTION

The rapid development of broadband wireless applications is driving new solutions for high throughput, area efficient, and reliable communications in wireless fading environment. Almost every branch of engineering and science uses Fourier methods. The Fast Fourier Transform is one of the rudimentary operations in the field of digital signal and image processing. The Fast Fourier Transform is simply a fast computationally efficient way to calculate the Discrete Fourier Transform. The FFT algorithm was presented by Cooley and Tukey in with an aim to compute Discrete Fourier Transform with significant reduction in number of computations. In fact reduced computations due to FFT algorithm helped to decrease power consumption, area and increase system throughput.

The work in this paper proposes a scalable radix2-8 point FFT processor architecture. Major components of FFT processor are Butterfly unit, data memory, twiddle factor memory, Interconnect, Address generation unit, Control unit. In the processor, two butterfly units are used to compute two outputs per clock cycle. Data memory storing data samples includes two sets of RAM called SetA and SetB. Twiddle factors are stored in a ROM. The link which connects butterfly units and address generation unit with data memory is known as interconnect. Since two memory sets are employed we require two interconnects which are called interconnectA and interconnectB. An address generation unit is required to provide address for data samples and twiddle factors. And a control unit is needed to co-ordinate and synchronize activities of rest of the components. The scalability of FFT processor acheived by using ping pong logic.The ping-pong logic is as follows: In even numbered stages, butterfly inputs are read from SetA and butterfly outputs are stored in SetB. In odd numbered stages, butterfly inputs are read from SetB and butterfly outputs are stored in SetA .The scalable FFT processor was designed and implemented using VHDL,simulated using ModelSim.

The modification work propose a hardware efficient algorithm known as CORDIC for the implementation of radix4 FFT processor. The algorithm is already famous for its simplicity in design ,less hardware utilization and low power consumption. So its use will certainly improve the FFT processor performance.

The paper is structured as follows: the Section II deals with basics of FFT, Section III describes FFT processor architecture in detail, Section IV explains dataflow algorithm for FFT processor architecture, Section V explains results .

II. FAST FOURIER TRANSFORM

The FFT algorithm was first presented by Cooley and Tukey in with an aim to compute Discrete Fourier Trans- form (DFT) with significant reduction in number of computations. In fact, reduced computations due to FFT algorithm helped to decrease power consumption, area and increase system throughput. Direct computation of N-point DFT would require N2 − N complex additions and N2 complex multiplication operations according to equation given by

$$X(K) = \sum_{n=0}^{N-1} x(n)\, e^{-(j2\pi nk}/N) \qquad (1)$$

Where k = 0, 1, 2....N − 1.

However, the using FFT algorithm total number of additions and multiplications reduces to $N*\log_2(N)$ and $N/2*\log_2(N)$ respectively. An N point FFT is given by

$$\mathrm{X(K)}=\sum_{n=0}^{\frac{N}{2}-1} x(2n)e^{-(\frac{j2\pi nk}{N}}/2) + W_N^K \sum_{n=0}^{\frac{N}{2}-1} x(2n+1)e^{-(j2\pi nk/}\mathrm{N/2})$$

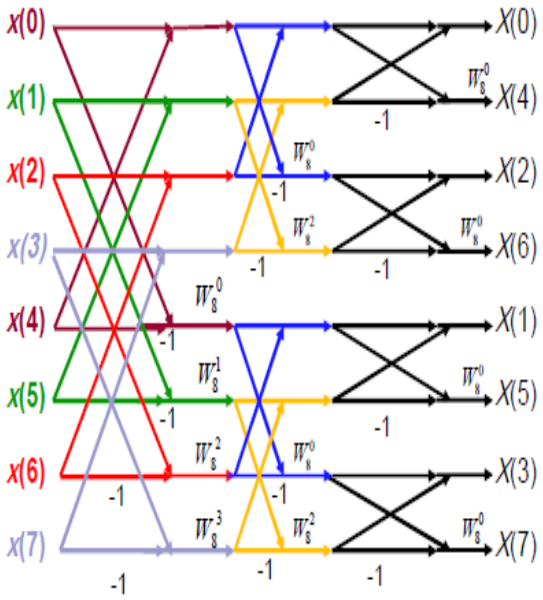

Fig 1:Radix-2 DIF FFT butterfly diagram

III SCALABLE FFT PROCESSOR ARCHITECTURE

The FFT processor is a fixed point processor which supports N-point complex value radix-2 FFT computation. Data path of the processor is 16-bit wide and the architecture is configurable at design time for required maximum FFT size Nmax. Once the processor is configured for Nmax, at runtime it supports any radix-2 FFT size from 16 to Nmax. The memories (data memory and twiddle factor memory) are suitably chosen to support Nmax-point FFT computation. The FFT processor consists of following major components as shown in Fig. 2

- Butterfly unit
- Data memory (RAM)
- Twiddle factor memory (ROM)
- Interconnect
- Address generation unit
- Control unit

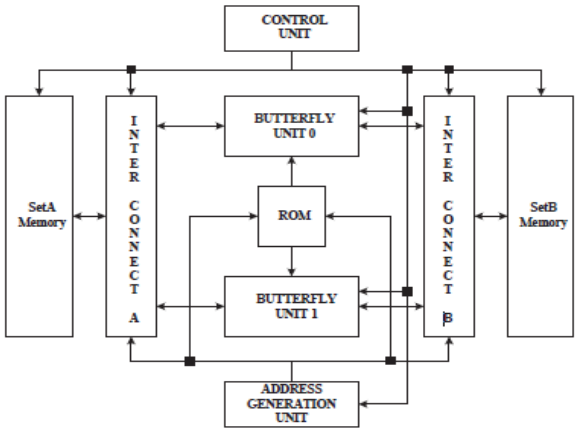

Fig 2: Scalable FFT processor block diagram

In the processor, two butterfly units are used to compute two outputs per clock cycle. Data memory storing data samples includes two sets of RAM called SetA and SetB. Twiddle factors are stored in a ROM. The link which connects butterfly units and address generation unit with data memory is known as interconnect. Since, two memory sets are employed we require two interconnects which are called interconnectA and interconnectB. An address generation unit is required to provide address for data samples and twiddle factors. And a control unit is needed to co-ordinate and synchronize activities of rest of the components.

Overall dataflow through the processor is pipelined and follows ping-pong logic. The ping-pong logic is as follows: In even numbered stages, butterfly inputs are read from SetA and butterfly outputs are stored in SetB. In odd numbered stages, butterfly inputs are read from SetB and butterfly outputs are stored in SetA.

A. *Butterfly unit*

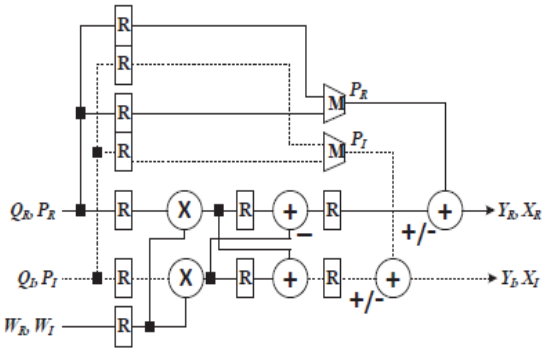

Fig 3: Butterfly unit

The butterfly unit was designed by J.Takala et al. to support radix-2 DIF butterfly operation. We modified and customized it to support Q-14 fixed point computation required for our implementation. Butterfly unit is shown in Fig. 3 and the dotted lines indicate imaginary data. The inputs and outputs of

butterfly unit are 16-bit fixed point values and it has three input and two output ports. Two input ports are shared between QR, QI and PR, PI respectively. The third input port is for twiddle factor which is shared between WR and WI. Two output ports available are shared between YR, YI and XR, XI respectively. The critical path of FFT processor includes a multiplier and an adder which is part of complex multiplier. When butterfly unit is in operation, Q and P are read every alternate clock cycle and same is the case with WR and WI.

B. Data memory (RAM)

The data memory stores data samples required for FFT computation. It is RAM based memory and consists of two memory sets SetA and SetB. Each memory set contains 4 memory banks. SetA includes RAM0, RAM1, RAM2, RAM3 while SetB includes RAM4, RAM5, RAM6, RAM7. Memory banks are clocked dual port memories enabling access within the processor as well as from external interface. Four memory banks were chosen since they offer concurrent access to four data samples at any given time. The read-write operation latency is one clock cycle. Since, two butterfly units operate in parallel concurrent access (read-write) to four data samples is necessary for high throughput. Smallest unit of data stored in memory is a 16-bit word. The real and imaginary parts of complex data are 8-bit values which are packed into a 16-bit value. The lower 8-bits contain imaginary part while upper 8-bits contain real part. Size of a memory bank is configurable at design time and it varies depending on maximum number of FFT points Nmax.

C. Twiddle factor memory (ROM)

The twiddle factor memory is a ROM, its a clocked dual port memory which supplies twiddle factors for butterfly unit0 and for butterfly unit1 .The maximum number of twiddle factors required for an N-point FFT computation is N/2

D. Interconnect

Interconnect is the link between butterfly unit and data memory. An interconnect receives address from address generation unit and routes it to data memory. The data from butterfly units to memory or from memory to butterfly units are routed via interconnect. The data or address routed by interconnect are controlled through control signals issued from control unit. Internal architecture of an interconnect consists of multiplexers and registers. The multiplexers act as switches to connect appropriate inputs to outputs wherein the outputs are registered. Two such interconnects are utilized in the processor architecture and are named interconnectA and interconnectB. The interconnectA connects butterfly unit0 and butterfly unit1 with memory SetA while interconnectB connects butterfly unit0 and butterfly unit1 with memory SetB.

E. Address generation unit

Address generation unit generates addresses in each stage of FFT computation for reading input data samples, twiddle factors and storing output data samples. A novel address generation scheme based on conflict free access of operands was developed for scalable FFT processor architecture. The address generation scheme is capable of supporting address generation for an N-point FFT computation. It uses two basic m-bit counters.

F. Control unit

Control unit is the master of FFT processor. It generates control signals at proper timing to control and coordinate activities of all the other blocks in the processor.

IV DATAFLOW ALGORITHM FOR FFT PROCESSOR

A novel algorithm based on dataflow in the FFT processor architecture. The dataflow algorithm is presented as follows. In even numbered stages, input data samples to butterfly units are read from SetA memory which are routed to butterfly units via interconnectA. The twiddle factors are read from ROM and supplied to butterfly units. After computation the output data samples are routed via interconnectB before they are stored in SetB memory. In odd numbered stages, input data samples to butterfly units are read from SetB memory

are routed to butterfly units via interconnectB. The twiddle factors are read from ROM and supplied to butterfly units. After computation the output data samples are routed via interconnectA before they are stored in SetA memory.

V RESULTS

A scalable radix-2 N-point novel FFT processor architecture based on a reasonable balance between performance, power, area, flexibility and scalability parameters was proposed. The scalable FFT processor was designed, implemented using VHDL, simulated using ModelSim.

.

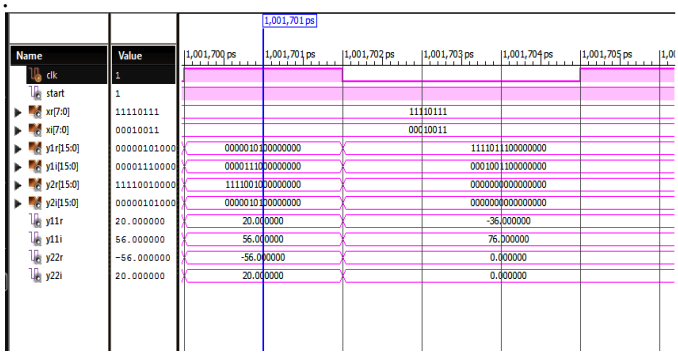

Fig 4:Output of 8 point FFT

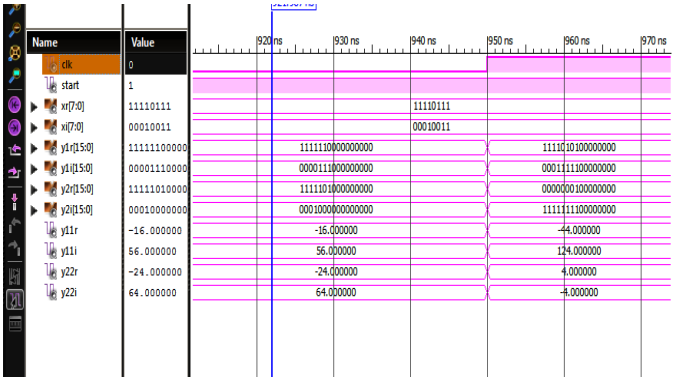

Fig 5 : Output of 8 point FFT

VI CONCLUSION

A scalable radix-2 N-point novel FFT processor architecture based on a reasonable balance between performance, power, area, flexibility and scalability parameters was proposed. The scalable FFT processor was designed, implemented using VHDL, and simulated using ModelSim.
The future work involves radix 4 FFT processor using CORDIC Algorithm.

REFERENCES

[1] Mounir Arioua, Said Belkouch, Mohamed Agdad, Moha Mrabet Hassani,"VHDLimplementation of an optimized 8-point FFT/IFFT processor in pipeline architecture for OFDM systems"Department of Electrical Engineering,ENSAM, Department of Physics,Faculty of Science Semlalia,2010 IEEE

[2] Vasantha Sudheer N, Venu Gopal B"FPGA Implementation of 64 Point FFT Processor"International Journal of Innovative Technology and Exploring Engineering (IJITEE)ISSN: 2278-3075, Volume-1, Issue-4, September 2012

[3] Asmita Haveliya"Design and Simulation of 32-Point FFT Using Radix-2 Algorith forFPGA Implementation"DOI 10.1109 /ACCT.2012.43,2012 IEEE

[4] Gijung Yang, Yunho Jung"Design of Low-Complexity FFT Processor for MIMO-OFDM Based SDR System"IEICE Electronics Express, volum4, no.23 pp.750-754,Dec.2007.

[5] Deepak Revanna,Omer Anjum,Manuele Cucchi,Roberto Airoldi,Jari Nurmi"A Scalable FFT processor Architecture for OFDM Based Communication Ststems,Department of Electronics and communication Engineering,2013IEEE

[6]. Y.-T. Lin, P.-Y. Tsai, and T.-D. Chiueh, "Low-power variable-length fast fourier transform processor," IEEE Proceedings – Computers and Digital Techniques, vol. 152, no. 4, pp. 499 – 506, july 2005.