

Implementation of FIR Filter and FFT Systems on a STRATIX-III FPGA Processor

Prashanth B.U.V
 Instrumentation Engineer
 Department of Science & Technology (DST)
 Promotion of University Research & Scientific Excellence (PURSE)
 Sri Venkateshwara University, Tirupati-517501,
 Andhra Pradesh, India

ABSTRACT

In this paper, the implementation of DSP modules like FIR Filter and FFT based systems are designed and implemented. The design is based on high performance FPGA “Cyclone II” and implementation is done after functional and timing simulation. The simulation tool used is ModelSim. The tool for synthesis and implementation is Quartus II. The experimental results shows the functional and timing analysis for all the DSP modules carried out using high performance synthesis software from Altera.

Keywords: *Simulation, FPGA, Code Conversion*

1. INTRODUCTION

This paper specifies the design analysis and simulation aspects of an N tap FIR filter, direct form FIR filter and FFT based computational systems. An N tap FIR filter is described by the following equation $y(n) = \sum x(n) * h(n-k)$, $0 \leq k \leq N-1$ the Coefficients are generated using FIR Compiler tool. Direct form FIR structure is implemented in pipelined and iterative form Filter structure is implemented in a modular form. FFT reduces the computation time required to compute DFT and improves performance by a factor of 100 over the direct evaluation of DFT. To compute all values of N requires N^2 complex multiplications and $N*(N-1)$ complex additions. FFT reduces the number of complex multiplications and additions to $N/2(\log_2 N - 1)$ multiplications and $N(\log_2 N)$ additions. In DIT algorithm the input is in bit reversed order and output is in natural order. The main advantages of these implementations are, it is very easy to interface with the special purpose DSP processor and also it is easy for cascading. The FIR filter structure is implemented in a modular form.

2. BLOCK SCHEMATIC

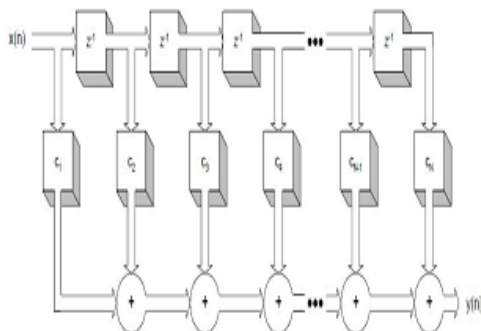


Figure 1 Block diagram of FIR filter-cascaded.

In this section the FIR filter is implemented on a FPGA as a fully parallel implementation and is called as a pipelined FIR

filter, which has the ability to perform 60 operations in a single clock cycle. The block diagram shown below is pipelined FIR filter.

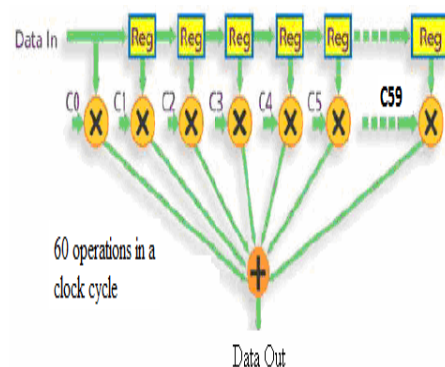


Figure 2 Block diagram of FIR filter-pipelined

3. FLOW DIAGRAM

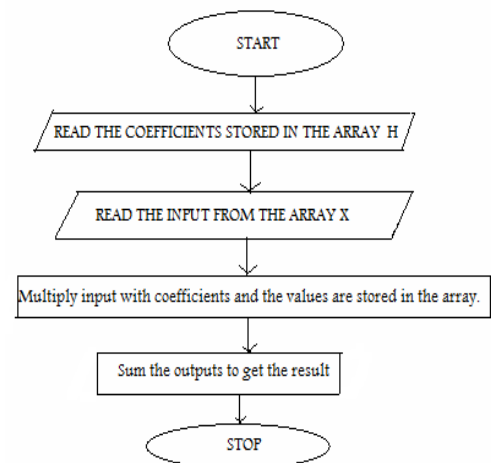


Figure 3 Flow chart FIR filter-pipelined

According to the above flow diagram a software routine is developed and accordingly the simulation is carried out using Quartus II software. The target device selected is Stratix III. The functional and timing analyses were carried out. The maximum frequency obtained is 247.65MHz.

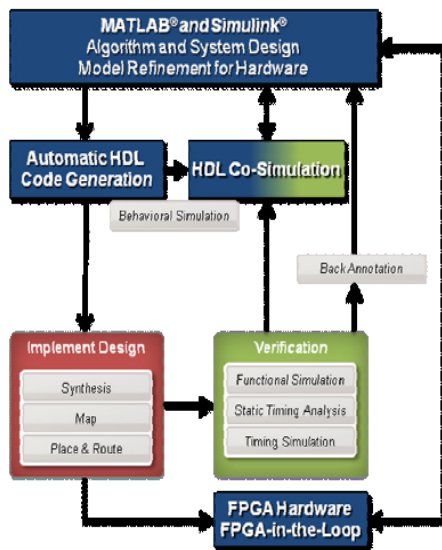


Figure 9. Mapping diagram of MATLAB to FPGA.

In this section the computational simulation of FFT implemented in QUARTUS Stratix-II VHDL environment is as shown in below figure

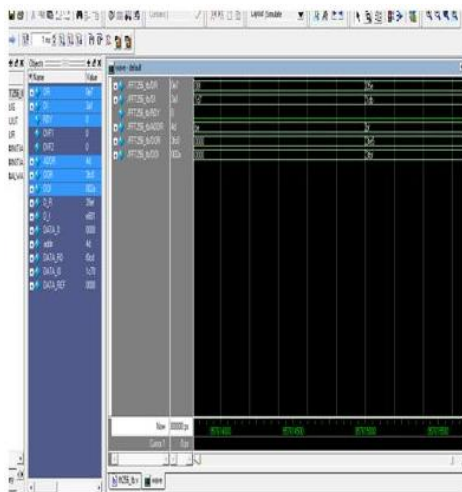


Figure10. Simulation of FFT Computational analysis.

Further the two simulation results of FFT are obtained in MATLAB are as shown below.

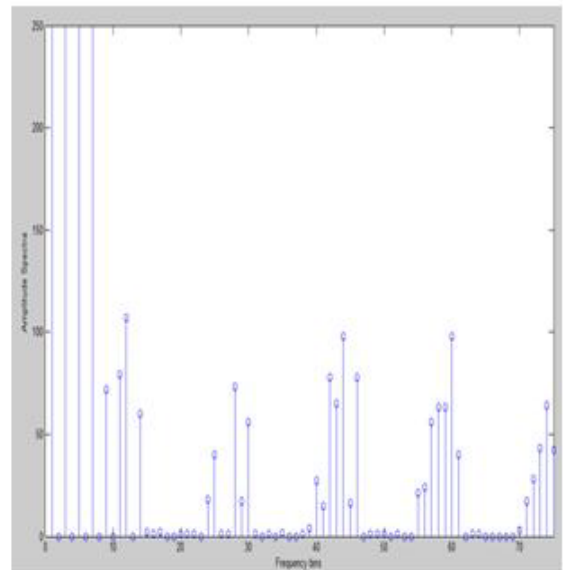


Figure11. Simulation of FFT in MATLAB for mapping on to the FPGA.

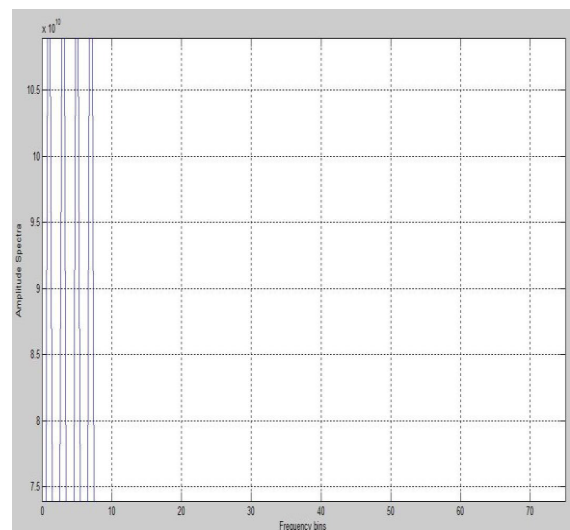


Figure12. FFT Simulations in MATLAB.

8. CONCLUSIONS

In this paper the implemented DSP modules are FIR filter and FFT based computational systems. All these DSP modules are designed from the block diagram approach to the synthesis and simulation aspects. The functional timing analysis results and the synthesis results are measured in precise and accurate manner. Finally the simulation waveforms are obtained in the FPGA simulation tools and the simulation waveforms are verified with the hardware design aspects, and matching results are obtained. The filter structure is implemented in a modular form. All the simulations are carried out in the Quartus-II software. The target device selected is Stratix-III. The functional and timing analysis for all the modules are carried out and accurate measurements are obtained.

9. REFERENCES

- [1] Steven Smith, (2003), *Digital Signal Processing-A Practical guide for Engineers and Scientists*, 3rd Edition, Elsevier Science, USA
- [2] John G. Proakis, Dimitris K Manolakis, (2003), *Digital Signal Processing Principles, Algorithms, Applications*, 4th edition, Prentice Hall, USA
- [3] "A New Common Sub expression Elimination Algorithm for Realizing Low-Complexity Higher Order Digital Filters" IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 29, No. 5, pp 844 - 848, May 2010.
- [4] Vinay K. Ingle, John G. Proakis, (2009), *Digital Signal Processing Using MATLAB, 3e*, Cengage Learning.
- [5] Lawrence R. Rabiner, Ronald W. Schafer, (2011), *Theory and Applications of Digital Speech Processing*, 1e, Prentice Hall.