**Design And Implementation Of FPGA Based Radix-4 FFT Processor Using CORDIC**

Pankaj H. Behare  
Researcher  
J.D college of engineering & Management  
Nagpur  
E-mail: pankajbehare48@gmail.com

Nilesh A. Mohota  
Assistant Professor  
J.D college of engineering & Management  
Nagpur  
E-mail: nileshmohota@gmail.com

**Abstract** - The Fast Fourier Transform (FFT) is an efficient algorithm for computing the Discrete Fourier Transform (DFT) and needs less number of computations than that of direct evaluation of DFT. It has several applications in signal processing. Because of the complexity of the processing algorithm of FFT, recently various FFT algorithms have been proposed to meet real-time processing requirements and to reduce hardware complexity over the last decades. The FFT processor is a critical block in all multi-carrier systems used primarily in the mobile environment. The portability requirement of these systems is mainly responsible for the need of low power FFT Architectures. We propose to implement an FFT processor, which will be synthesized on Field Programmable Gate Array (FPGA). The purpose of this project is to obtain an area efficient description of an FFT processor by using CORDIC algorithm.

**Key words** - FFT, DFT, DIF, CORDIC, RADIX-4, FPGA

**Introduction:**

Audio and communications signal processing are well developed lines massively used now a days in many application lines and products. Since digital communications are quite active fields, the arithmetic complexity of the Discrete Fourier Transform (DFT) algorithm becomes a significant factor with impact in global computational costs. Cooley and Tukey [1] developed the well-known radix-2 Fast Fourier Transform (FFT) algorithm to reduce the computational load of the DFT. The Discrete Fourier Transform (DFT) X(k) of N points is given by

\[
X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk} \quad \{ 0 \leq k < N \}, \quad W_N^{nk} = \exp(-j2\pi nk/N) \]

\[ ........(1) \]

Where the \( X(k) \) and \( x(n) \) are frequency-domain sequences and time-domain sequence. Instead of the direct implementation of the equation (1), the FFT algorithm factorizes a large point DFT recursively into many small point DFT in order to reduce the overall operations. There are two well-known types of decompositions called Decimation in Time (DIT) and Decimation in Frequency (DIF) FFT. The only difference between these two algorithms is that, DIT starts with bit reverse order input and generates normal order output. Nevertheless DIF starts with normal order input and generates bit reverse order output. Throughout this paper DIF algorithm is used [7]. The conventional method of Fast Fourier Transform FFT calculation involves \( N^2 \) complex multiplications and \( N(N-1) \) complex additions. The radix-2 Cooley-Tukey algorithm performs the same computation involving \( (N/2) \log_2 N \) complex multiplications and \( (N) \log_2 N \) complex additions. But it is more efficient computationally to employ a radix-4 FFT algorithm other than radix -2 logarithms. The radix-4 decimation-in-frequency DFT is given by the equation below

\[
X(4k) = \sum_{n=0}^{N/4} x(n) W_N^{n4k} + x(n + N/4) + x(n + N/2) + x(n + N) W_N^{n4k} \]

\[
X(4k+1) = \sum_{n=0}^{N/4} x(n) W_N^{n4k} - x(n + N/4) - x(n + N/2) + x(n + N) W_N^{n4k} \]

\[
X(4k+2) = \sum_{n=0}^{N/4} x(n) W_N^{n4k} - x(n + N/4) + x(n + N/2) - x(n + N) W_N^{n4k} \]

\[
X(4k+3) = \sum_{n=0}^{N/4} x(n) W_N^{n4k} + x(n + N/4) - x(n + N/2) - x(n + N) W_N^{n4k} \]

\[ ........(2) \]

The input to each \( N/4 \)-pointDFT is a linear combination of four signal samples scaled by a twiddle factor. This procedure is repeated \( v \) times, where \( v = \log_4 N \). The complete butterfly
operation for Radix-4 DIF is shown in figure 1(a) and in a more compact form in figure 1(b) shown below.

![Fig.1 (a) complete butterfly operation of radix-4](image)
![Fig.1 (b) compact form.](image)

**Proposed system:**
Design of FFT Processor using CORDIC algorithm:
The design flow of FFT Processor using CORDIC is shown in figure below. The selector block is nothing but a memory path buffer which computes respective memory of input samples. When Active signal is asserted and there are some input data, the address generator block assigns a memory position for each input sample. Now when Dual port Ram gets write Address signal from address generator block, it saves both memory path along with respective input samples. The 4 point FFT block has butterfly unit within it. The overall structure of processor is CORDIC based FFT [3] processor model is shown in Figure below. The entire model is made of the address generation unit, the control unit, the dual port RAM unit, the 4-point butterfly unit and the CORDIC twiddle factor generation unit [6]. This model is characterized by setting the parameter, sampling points and the accuracy to meet the actual needs.

![Fig.2 Simplified Block Diagram.](image)

To perform these operations concurrently, a dual -port RAM has been employed. The control unit involves the timing control of the data storage, reading and writing to make the corresponding data and rotating factor coefficient flow into butterfly and CORDIC computing unit in sequence in FFT operation. Data and address of the twiddle factor” can be easily generated by the counter. The address generation logic is very simple and does not limit the throughput of the system. When a start signal is asserted, at the same time, both to 4 Point FFT and Rotation factor generator block, the FFT block sends a

![Fig.3 Block diagram of Radix-4 FFT Processor using CORDIC.](image)
round block, remapping of memory path and twiddle factors are held and fed back to FFT block. Now when address generator block sends read address signal to DRAM [7], it sends stored input data samples along with memory path in FFT block. Finally this twiddle factors are applied to the output of the butterflies, and a bit reverse scramble is done.

**Progressive work:**
Ex. Simulation model of 8 pt fft by taking twiddle factor 1. An 8 point FFT structure is coded in Verilog. It is compiled and simulated. Xilinx ISE Design Suite is used for compilation and simulation.

Test input: \{1, 2, 4, 2, 1, 3, 5, 6\}

Output : \{24, -6, -10, 4, -2, 4, -4, -2\}

**RTL Schematic:**

![RTL Schematic](image)

**Simulation:**

![Simulation](image)
FINAL WORK:

RTL Schematic.

Fig. 7: Full RTL view of Fft processor

Fig. 8: RTL schematic of stages of 16 point FFT.

Fig. 9: RTL Schematic of each stage 16 point FFT Processor.

Simulation of 16 point FFT

Fig. 10: Simulations of FFT 16 point.
Area Requirement:

<table>
<thead>
<tr>
<th>Project Title</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module Name</td>
<td>Status</td>
</tr>
<tr>
<td>Target Device</td>
<td>Status</td>
</tr>
<tr>
<td>Product Version</td>
<td>Status</td>
</tr>
<tr>
<td>Design Goal</td>
<td>Status</td>
</tr>
<tr>
<td>Design Strategies</td>
<td>Status</td>
</tr>
<tr>
<td>Environment</td>
<td>Status</td>
</tr>
</tbody>
</table>

Conclusions & Future Work

As FFT takes values in time domain and generates equivalent sample values in frequency domain, 16 values of are taken as an input to the design which is in time domain and, using Decimation in Time method for FFT computation output is generated in terms of frequency samples. Due to in-place computation memory requirements reduced and using CORDIC for complex twiddle factor generation and multiplication the speed of the computation gets improved with less complexity another factor which helps to improve speed is radix-4 algorithm which computes the output in half the stages required for radix-2 algorithm.

Future work:

As there will be always some improvement that can be done, same goes to this project. The future scope should include the implementation of Floating point CORDIC FFT processor with more number of points, implementation of radix 4 architecture, CORDIC based implementation and simulation of Discrete Cosine
Transform, Discrete Hartley Transform and Singular Value Decomposition. The above proposed architectures can be used in various signal processing, image processing and communication engineering application.

Applications:

1. In analysis and implementation of digital communication systems.
2. Television terrestrial broadcasting systems such as the xDSL (de)modulator.
3. Phase correlation system.
4. Mobile receiver.
5. Fault characterization and classification.
6. Radar.
7. Medical Imaging
   1. X-ray computed tomography (CT).

Advantages

[1] Radix-4 algorithms have an advantage over radix-2 algorithms because a single radix-4 butterfly does the work of four radix-2 butterflies which will reduce the steps of computation and end up with more speed in processing the data.

[2] The radix-4 FFT requires only 75% as many complex multiplies as the radix-2 FFTs, although it uses the same number of complex additions. These additional savings make it a widely-used FFT algorithm.

References:

[14]. Meng Qian, Applications of CORDIC Algorithm to Neural Networks in VLSI Design ,IMACS 2006,Beijing, China
[16]. Bu-chin wang Digital signal processing Techniques and Applications in Radar Image processing.


