FPGA Implementation of a CORDIC-based Radix-8 FFT Processor for Real-Time Harmonic Analyzer

Venkata Subbarao Gutta
M.tech/VLSI design
SRM University
Kattankuluthur-603203

S. Malarvizhi
Professor
SRM University
Kattankuluthur-603203

ABSTRACT

In this paper, the design of a CORDIC algorithm based radix-8 FFT processor is presented for real-time harmonic analyzer. The actual need of FFT is the obtaining a levels of harmonics in power signal.   Generally the radix-8 FFT processor is a twiddle factor based butterfly computation. At each stage of the butterfly the twiddle factor is multiplied with the input sequence and each stage required a RAM to store the twiddle factor angles. The choice of the CORDIC algorithm for realizing butterfly operation for FFT which eliminates the need for storing twiddle factors and angles saves a lot of hardware compared to its counterparts employing other techniques. The importance of radix -8 FFT compare with radix-2 and radix-4 it's taking less calculation resources. The total CORDIC based radix-8 FFT is implemented in a field programmable gate array (FPGA) that is characteristic of high efficiency, low cost, convenient implementation and short development cycle, and its performance is found to be satisfactory. . A dual-port memory structure and the corresponding addressing scheme are used to realize the in place data access. The address generation unit required for fetching data from and writing results into the dual-port memory in proper sequence is also incorporated within the chip which houses the controller as well. The full design is implementing using XILINX Vertex4 series.

Keywords
CORDIC; Radix-8; FFT; FPGA

1 INTRODUCTION

With more and more non-linear devices used in the power grid, the contents of harmonics also increase, while the power quality has been required more strictly. Harmonics in power systems result in increased heating in the equipment and conductors, misfiring in variable speed drives, and torque pulsations in motors. High levels of power system harmonics can create voltage distortion and power quality problems.

In order for the safe operation of power grids and power electronics technology to meet the needs of development, we must take effective measures to curb harmonics, and the harmonics test is first step of harmonic analysis.

At present, various approaches are available for harmonic detection. The method based on the instantaneous reactive power (IRP) theory, in which the harmonic signals are extracted from voltage and current measurements, has been widely utilized for harmonic compensation.

 But now it appears to need more researches suitable for digital control substituting for conventional analog control. As to the methods of wavelet transform [4] and neural networks only when each harmonic voltage and current amplitude and

phase angle are worked out, can the harmonics content be found. Since the wavelet transform and neural network is so complex that they have not yet been widely used in practical projects.

FFT based harmonic estimation is popular, but it needed huge data processing, therefore either DSP,MCU or FPGA has to be used. Considering the real time measurement of FPGA is preferable. As it has the characteristic of high parallel, throughput and accuracy in the detection of the power harmonics is obtained.

In this paper, a new pipelined, reduced memory CORDIC based architecture is presented for a radix-8 FFT. A dual-port memory structure and the corresponding addressing scheme are used to realize in-place data accesses.

The proposed memory-reduced CORDIC algorithm eliminates the need for storing twiddle factors and angles, resulting in significant area savings with no negative impact on performance. As a case study, the CORDIC-based Radix-8 FFT processor have been implemented in FPGA. The rest of this work is arranged as follows.

In Section 2, Radix-8 FFT and CORDIC algorithm fundamentals are briefly described. Then, the hardware architecture of the proposed CORDIC-based FFT is presented in Section 3. The experimental results are provided in section 4 and a look at future researches will be stated in the conclusion.

2. FFT AND CORDIC ALGORITHM
2.1 Radix-8 FFTAlgorithm
 The N-point discrete Fourier transform [5] is defined by

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{4nk}, \quad k=0, 1, \ldots, N\text{-}1, \quad W_N^{nk} = e^{-j\frac{2\pi}{N}nk} \quad (1)$$

The radix-8 decimation-in-time FFT algorithm is described briefly in the following equations. The N-point input sequence is split into eight subsequences, X(8m), X(8m+1), X(8m+2), X(8m+3), X(8m+4), X(8m+5), X(8m+6), X(8m+7), where k=0,1, … , N/4-1. Thus DFT of the radix-8 decimation-in-time can be expressed as:

$$X(k) = \sum_{m=0}^{\frac{N}{8}-1} x(8m) W_N^{8mk} + \sum_{m=0}^{\frac{N}{8}-1} x(8m+1) W_N^{(8m+1)k}$$

$$+ \sum_{m=0}^{\frac{N}{8}-1} x(8m+2) W_N^{(8m+2)k} + \sum_{m=0}^{\frac{N}{8}-1} x(8m+3) W_N^{(8m+3)k}$$

$$+ \sum_{m=0}^{\frac{N}{8}-1} x(8m+4) W_N^{(8m+4)k} + \sum_{m=0}^{\frac{N}{8}-1} x(8m+5) W_N^{(8m+5)k}$$

$$+ \sum_{m=0}^{\frac{N}{8}-1} x(8m+6) W_N^{(8m+6)k} + \sum_{m=0}^{\frac{N}{8}-1} x(8m+1) W_N^{(8m+1)k}$$
(2)

Let $W_N^{8mk} = W_{N/4}^{mk}$, $W_N^k = W^p$, A $= \sum_{m=0}^{\frac{N}{8}-1} x(8m)W_{N/8}^{mk}$,

B $= \sum_{m=0}^{\frac{N}{8}-1} x(8m+1)W_{N/8}^{mk}$, C $= \sum_{m=0}^{\frac{N}{8}-1} x(8m+2)W_{N/8}^{mk}$,

D $= \sum_{m=0}^{\frac{N}{8}-1} x(8m+3)W_{N/8}^{mk}$, E $= \sum_{m=0}^{\frac{N}{8}-1} x(8m+4)W_{N/8}^{mk}$,

F $= \sum_{m=0}^{\frac{N}{8}-1} x(8m+5)W_{N/8}^{mk}$, G $= \sum_{m=0}^{\frac{N}{8}-1} x(8m+6)W_{N/8}^{mk}$,

H $= \sum_{m=0}^{\frac{N}{8}-1} x(8m+7)W_{N/8}^{mk}$, then Eq. (2) can be written as

X (k) =A + BW^p+CW^{2p}+DW^{3p}+EW^{4p}+FW^{5p}+GW^{6p}+HW^{7p}

X (K+N/8) =A+BW^P-JCW^{2P}-JDW^{3P}-EW^{4P}-FW^{5P}+JGW^{6P}

+JHW^{7P}

X (K+2N/8) =A-JBW^P-CW^{2P}+JDW^{3P}+EW^{4P}-JFW^{5P}-GW^{6P}

+JHW^{7P}

X (K+3N/8) =A-JBW^P+JCW^{2P}+DW^{3P}-EW^{4P}+JFW^{5P}-JGW^{6P}

-HW^{7P}

X (K+4N/8) =A-BW^P+CW^{2P}-DW^{3P}+EW^{4P}-FW^{5P}+GW^{6P}

-HW^{7P}

X (K+5N/8) =A-BW^P-JCW^{2P}+JDW^{3P}-EW^{4P}+FW^{5P}+JGW^{6P}

-JHW^{7P}

X (K+6N/8) =A+JBW^P-CW^{2P}-JDW^{3P}+EW^{4P}+JFW^{5P}-GW^{6P}

-JHW^{7P}

X (K+7N/8) =A+JBW^P+JCW^{2P}-DW^{3P}-EW^{4P}-JFW^{5P}-JGW^{6P}

+HW^{7P}

....

(3),

Eq. (3) can be expressed as Eq. (4) in a matrix form according to the characteristic of W_N^{mk} .

$$
\begin{bmatrix} A' \\ B' \\ C' \\ D' \\ E' \\ F' \\ G' \\ H' \end{bmatrix} =
\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & -J & -J & -1 & -1 & J & J \\
1 & -J & -1 & J & 1 & -J & -1 & J \\
1 & -J & J & 1 & -1 & J & -J & -1 \\
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\
1 & -1 & -J & J & -1 & 1 & J & -J \\
1 & J & -1 & -J & 1 & J & -1 & -J \\
1 & J & J & -1 & -1 & -J & -J & 1
\end{bmatrix}
\begin{bmatrix} A \\ BW^P \\ CW^{2P} \\ DW^{3P} \\ EW^{4P} \\ FW^{5P} \\ GW^{6P} \\ HW^{7P} \end{bmatrix}
$$

(4)

Where

A'=X (k), B'= X (K+N/8),C'=X(K+2N/8),D'=X(K+3N/8),

E'=X (K+4N/8),F'=X(K+5N/8),G'=X(K+6N/8),H'=X(K+7N/8)

Thus, a radix-8 butterfly computing unit can be viewed as the input sequences first multiplied by a rotating factor, then to beside matrix, which makes the total number of multiplications for the radix-8 much less than for the radix-2 and radix-4.

2.2 CORDIC Algorithm

CORDIC algorithm was proposed by J.E. Volder[2]. It is an iterative algorithm to calculate the rotation of a vector by using only additions and shifts Fig.1 shows rotation of a vector V_i.
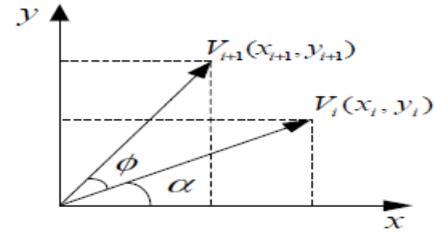

Fig 1: Schematic diagram of the transformation of $V_i(x_i, y_i)$ to $V_{i+1}(x_{i+1}, y_{i+1})$

$x_{i+1} = r\cos(\alpha + \emptyset) = r(\cos\alpha\cos\emptyset - \sin\alpha\sin\emptyset)$

$= x_i\cos\emptyset - y_i\sin\emptyset$

(5)

$y_{i+1} = r\sin(\alpha + \emptyset) = r(\sin\alpha\cos\emptyset + \cos\alpha\sin\emptyset)$

$= y\cos\emptyset + x_i\sin\emptyset$

(6)

Let each rotation angle $\emptyset$ be equal to arctan 2^{-i} then:

$x_{i+1} = \cos\emptyset(x_i - y_i . 2^{-i})$

(7)

$y_{i+1} = \cos\emptyset(y_i + x_i . 2^{-i})$

(8)

Since $\emptyset = \arctan 2^{-i}$, $\cos\emptyset$ can be simplified to a constant with fixed number of iteration:

$x_{i+1} = k_i(x_i - y_i . d_i 2^{-i})$

(9)

$y_{i+1} = k_i(y_i + x_i . d_i 2^{-i})$

(10)

Where $k_i = \cos(arc\tan(2^{-i}))$ and $d_i = \pm 1$. Product of k_i 's can be represented by the k factor which can be used as a single constant multiplication either at the beginning or end of the iterations. Then, (9) and (10) can be simplified as:

$x_{i+1} = (x_i - y_i . d_i . 2^{-i})$

(11)

$y_{i+1} = (y_i + x_i . d_i . 2^{-i})$

(12)

The direction of each rotation is defined by d_i and the sequence of all d_i's determines the final vector. d_i is given by:

$d_i = \begin{cases} -1 & if\ z_i < 0 \\ +1 & if\ z_i < 0 \end{cases}$

(13)

Where z_i is called as angle accumulator and given by

$$z_{i+1} = z_i - d_i . arc \tan 2^{-i}$$

(14)

All operation described by Eqs.(11)-(14) Can be implemented by only additions and shifts. Therefore, CORDIC algorithm does not require dedicated multipliers.

As shown in Eq. (1), the key operation of the FFT is x(n). W_N^{nk} , (where $W_N^{nk} = e^{-j\frac{2\pi}{N}nk}$ is just the so called twiddle factor). This is equivalent to "rotate x (n) by angle $-\frac{2\pi}{N}$nk" operation which can be realized easily by the CORDIC algorithm. Without normal multiplication, CORDIC based butterfly can be very fast.

3. PROPOSED CORDIC BASED FFT

3.1 CORDIC Based FFT Architecture
The overall structure of processor[1] is CORDIC based FFT processor model is shown in Figure 2. The entire model is made of the address generation unit, the control unit, the dual port RAM unit, the 8-point butterfly unit and the CORDIC twiddle factor generation unit. This model is characterized by setting the parameter, sampling points and the accuracy to meet the actual needs.
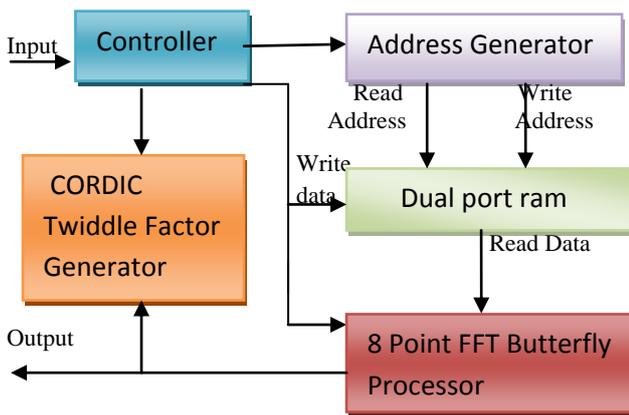

Fig 2: Proposed CORDIC based FFT

The FFT processor presented here is based on radix-8 DIT algorithm in which the in-place computation is utilized to achieve an efficient use of the memories. 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.

3.2 CORDIC Unit
The CORDIC processor performs the vector rotation to compute a set of trigonometric functions. The principle idea of the CORDIC algorithm is to decompose a rotation into a sequence of micro-rotations. The pipelined CORDIC arithmetic unit can be obtained by decomposing the CORDIC algorithm in to sequence of operational stages. The corresponding stages are expressed by the following iterative Eqs.(11)-(14)

The CORDIC method can be employed in two different modes, known as the rotation mode and the vector mode. In this paper, one uses x and y to represent the real part and imaginary part input and output vectors. In the rotation mode, the co-ordinate components of a vector and an angle of rotation are given and the co-ordinate components of the original vector are computed after rotation by a given angle in the vector mode, the co-ordinate components of a vector are given and the magnitude and angular arguments of the original vector are computed.
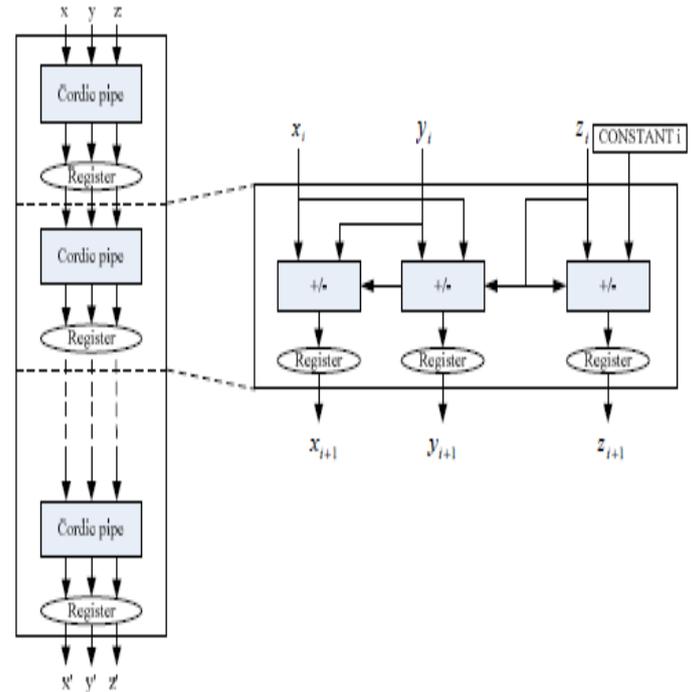

Fig 3: CORDIC unit architecture

All the iterations of CORDIC algorithm are performed in parallel by using a pipelined structure ensures the highest possible throughput because a CORDIC transformation can be performed in each clock cycle.

3.3. FFT Butterfly Processor
Based on Eq.(4), eight subsequences of FFT X(8m),X(8m+1),X(8m+2),X(8m+3),X(8m+4),X(8m+5),X(8m+6),X(8m+7) presented in the section 2.1 can be then schematized in Figure (4). The butterfly computation is the basic operator of an FFT processor. The operations of eight complex additions and seven complex multiplications need to be performed in each sequence of radix-8 butterfly processor. Complex multiplication by twiddle factor can be replaced by the pipelined CORDIC algorithm. The complex multiplication is then transformed into CORDIC iterations in which only the adder and the shifter are used. In general, since the multiplication is much more expensive in term of computer operations than addition, the CORDIC methods generally offer also considerable savings. In additions the hardware consumptions can be reduced significantly by using the CORDIC unit[6].

The output of each 8-point FFT is a linear combination of eight signal sampling subsequences which are rotated and scaled by a k factor. This procedure will repeat log_8^N times
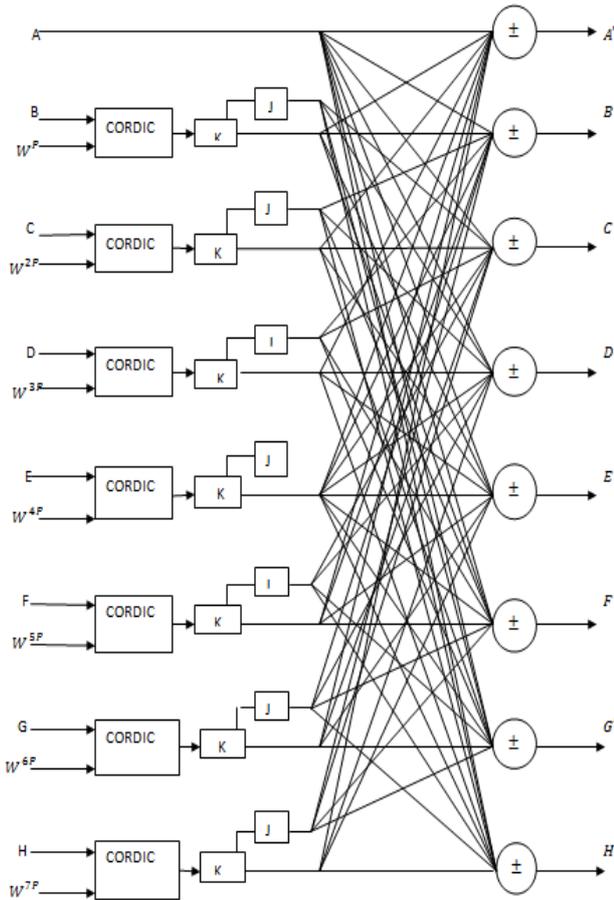
Fig 4: CORDIC 8-point Butterfly
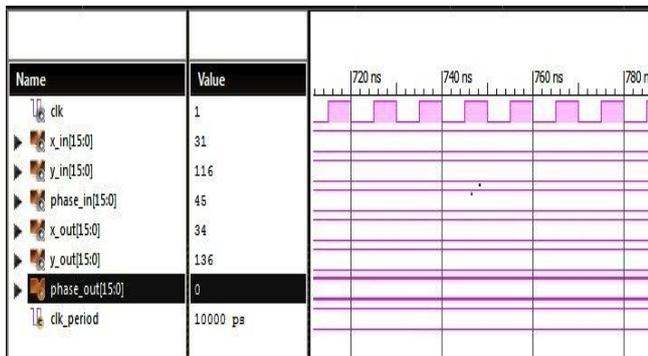
4. CORDIC SIMULATION RESULTS

Fig 5: CORDIC Simulation results

5. CONCLUSION

In this paper, the design of CORDIC algorithm based radix-8 FFT processor for spectrum analyzer using FPGA with intended application in power harmonics signal processing is presented. The choice CORDIC algorithm for realizing the butterfly operation for the FFT. This kind of architecture can solve the contradiction between the real time and accuracy in the detection of power harmonics well.

6. REFERENCES

[1] Ren-Xi Gong, Jiong-Quan Wei and Dan Sun " FPGA Implementation of CORDIC –based Radix-4 FFT Processor for Real-Time Harmonic Analyzer", 2011 Seventh International Conference on Natural Computation. 2011, pp 1832-1835.

[2] J.Volder, "The CORDIC trigonometric computing technique", IEEE Transactions on Electronic Computers, vol.EC-8, no. 8, pp 330-334, September 1959.

[3] Hong.Zhi. Wang, Louet. Y, Palicot. J, Alaus, "Memory efficient FFT architecture using R-LFSR based CORDIC common operator", Cognitive Information Processing (CIP), 2010 2nd International Workshop. 2010, pp. 162-167

[4] LIU. Min, WANG. Ke. Ying "A high accurate harmonic analysis method based on FFT and neural networks in power system", Electric Power College, South China University of Technology, Guangzhou, 510640, china

[5] Li. Cheng. Shi, Chu. Jain. Peng, Li. Xin. Bing, "A high speed real time and Fixed-point FPGA Realization of a CORDIC based FFT Processor", Microelectronics & Computer, April 21th, 2004, pp. 88-91

[6] T. Widhe, J. Melander, L. Wanhammer, "Design of Radix-8 Butterfly PEs for VLSI"