FPGA implementation of Hilbert Transform via Radix-2^2 Pipelined FFT Processor

Archna Rani[1], Ram Mohan Verma[1], Saurabh Jaiswal[1],
[1] Student, Department of microelectronics,
Indian Institute of Information Technology-Allahabad,
Deoghat, Jhalwa, Allahabad,
UP, India-211012
{archna243900, rammohanverma15, jaiswalsaurabh2009}@gmail.com

***Abstract*— Hilbert transform (HT) is very important algorithm in signal processing. Its application includes constructing analytic signals for various purposes, such as amplitude demodulation, audio production and instantaneous frequency analysis. This paper explains the realization of HT via radix-2^2 single-path delay feedback (SDF) pipelined FFT processor. For this purpose, it requires a direct and an inverse FFT implementation. The processor has been developed using hardware description language Verilog on an Xilinx XC5VLX110T FPGA and simulated up to maximum frequency of 273.299 MHz.**

***Keywords*— Envelope, FFT, FPGAs, Hilbert Transform, Radix 2^2 single-path delay feedback (Radix 2^2 SDF).**

I. INTRODUCTION

Hilbert transform (HT) plays an essential role in constructing analytic signals for a variety of signal and image processing applications. Conventionally, the HT has been used in envelope and instantaneous frequency analysis and as a core part in amplitude demodulators. The applications of HT extend from geophysical, seismic, ultrasonic and radar to biomedical signals and speech recognition systems. The theoretical basis of HT is well established, but the computational procedures are still being developed. Many methods have been proposed, such as digital filtering, the parametric modeling approach,Subband Hilbert Transform[3] and the discrete Cosine transform [1]. In this work we used radix 2^2 SDF FFT-based method for computation of the analytic signal. Radix-2^2 has a simple butterfly structure and higher multiplier utilization [2],[4]. This makes Radix-2^2 single path delay feedback architecture attractive for implementation.

The paper is organised as follows. We discuss the theoretical considerations in section II, in which we brief the properties of HT and the related analytic signal. Then we review the FFT-based method for computation of the analytic signal and finally the Radix-2^2 FFT algorithm is illustrated. The Implementation details are discussed in section III. Synthesis results, consumed resources and its application as an Envelope Detector are explained in Section I. A brief conclusion is presented in Section V.

II. THEORETICAL CONSIDERATIONS

A. Hilbert Transform

Hilbert transform is often used to obtain an analytic signal defined as

$$x_a[n] = x[n] + j\overline{x[n]} \tag{1}$$

where $x[n]$ denotes the Hilbert transform of the discrete time signal $x[n]$, n = 0, 1, ... , N-1 and $j = \sqrt{-1}$ [1].
Sampling the continuous time signal at T intervals gives the discrete time signal x[n] .
The Fourier transform of analytic signal has the following property.

$$X_a(j\omega) = \begin{cases} 2X(j\omega) & 0 \le \omega < \pi \\ 0 & -\pi \le \omega < 0 \end{cases} \tag{2}$$

On the other hand, Fourier transform $\overline{x[n]}$ is of the form

$$\overline{X(j\omega)} = \begin{cases} -jX(j\omega) & 0 \le \omega < \pi \\ jX(j\omega) & -\pi \le \omega < 0 \end{cases} \tag{3}$$

A common use of the Hilbert transform is to regain the amplitude information of a modulated signal[1]. As an example let us consider a complex sinusoidal signal

$$x[n] = a[n]\, e^{-j(\omega n + \varphi)} \tag{4}$$

The time dependent amplitude a[n] may be reconstructed from

$$A[n] = |x_a[n]| = \sqrt{x[n]^2 + \overline{x[n]}^{\,2}} \tag{5}$$

B. Computation of Analytic Signal using FFT

The discrete Fourier transform (DFT) of the signal $x[n]$, n = 0, 1, ... , N-1 is defined as

$$X[k] = \sum_{n=0}^{N-1} x[n]\, W_N^{kn}, \quad k = 0, 1, \ldots, N-1, \tag{6}$$

where $W_N^{kn} = e^{-2\pi jkn/N}$The inverse transform (IDFT) is defined as

$$X[n] = \frac{1}{N} \sum_{k=0}^{N-1} x[k]\, W_N^{-kn}, \quad n = 0, 1, \ldots, N-1 \tag{7}$$

The computation of analytic signal using the FFT- based method is based on the property (2) of the Fourier spectrum of the analytic signal. If $X[k]$ (k=0, 1, ... , N-1) denotes the DFT coefficients of the original signal and $X[k]$ (k=N/2 ....., N-1) represent the values in the negative frequency band $(-\pi < \omega < 0)$,by zeroing those negative frequency coefficients the inverse FFT yields the analytic signal[5] as is shown in Fig-1.

The analytic signal is now computed from

$$x_a[n]=IFFT_N\{\, w[n]\, X[k]\, \} \tag{8}$$

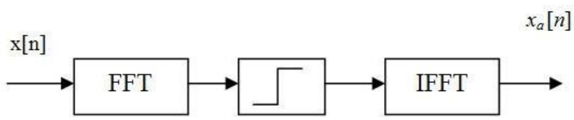

Figure -1 Block diagram for Hilbert Transform.

C. Radix 2^2 FFT Algorithm

Using Common Factor Algorithm (CFA) to decompose the twiddle factor, FFT can be reconstructed as a set of 4 DFTs of length N/4[2]. The decomposition is done by substituting with

$$n = \langle \frac{N}{2}n_1 + \frac{N}{4}n_2 + n_3 \rangle \tag{9}$$

$$k = \langle k_1 + 2k_2 + 4k_3 \rangle \tag{10}$$

In equation (6), which yields

$$X(k_1 + 2k_2 + 4k_3) = \sum_{n_3=0}^{\frac{N}{4}-1}\{H(k_1,k_2,n_3).\ W_N^{n_3(k_1+2k_2)}\}W_{\frac{N}{4}}^{n_3k_3} \tag{11}$$

With

$$H(k_1,k_2,n_3) = [x(n_3)+(-1)^{k_1}.x(n_3+\tfrac{N}{2})] +(-j)^{k_1+2k_2}[x(n_3+\tfrac{N}{4})+(-1)^{k_1}.x(n_3+\tfrac{3N}{4})] \tag{12}$$

Equation (12) represents a Radix-2 butterfly (BFI), while the whole equation represents Radix-2 butterfly with a trivial multiplication by (-j) (BFII). After these two butterflies, twiddle factor multipliers (TFM) are required to calculate the multiplication by the twiddle factor $W^{n_3(k_1+2k_2)}$. The last stage differs according to the size of FFT. If N is a spower of 2, the last stage is BFI and if N is power of 4, the last stage is BFII.

III. IMPLEMENTATION

The word length chosen for our implementation is 12-bits for the data and 12-bits for twiddle factor .The output is also stored in 12-bit register. Both FFT and IFFT are 16 point Decimation in Frequency.

A. FFT

The N-point FFT processor has $i = \log_2 N$ stages. In our design we have 16-point FFT with 4 stages. The inputs are given in normal order and the outputs are in permuted (digit-reversed) order. Each of the four stages with TFM of three stages makes the seven stage pipelined FFT structure. Fig-2 shows the Block Digram for N Point SDF FFT Structure.

The control unit for FFT is a simple counter. A $i = \log_2 N$ counter is used for switching the butterflies between modes. It also used to address ROMs in order to pick the Twiddle Factors. In our implementation a mod-16 counter serves this purpose.

B. IFFT

Similar to FFT, IFFT is also a seven stage pipelined structure. In order to implement IFFT processor, the same design is used with conjugated twiddle factor. Since the output for FFT is in digit-reversed order, the IFFT processor is configured to accept digit-reversed inputs, which produces normal ordered output. This provides a saving of N complex memory words, and a latency saving of N clock cycle.

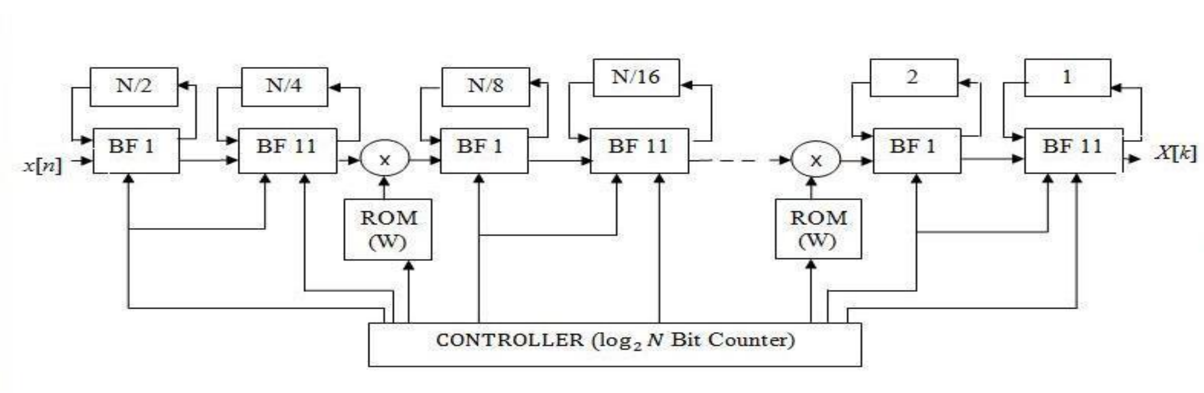

Figure- 2 Block diagram for N Point SDF FFT Structure

F

IV. RESULTS

The HT processor was described with hardware description language Verilog as fixed-point arithmetic and synthesized with Xilinx ISE version 12.3 on XC5VLX110T FPGA. The synthesis tool has allocated the following resources (Table.1).

TABLE 1

	Device Utilization Summary			
	Logic Utilization	**Available**	**Used**	**Utilization**
1	Number of Slice Registers	64000	1060	1%
2	Number of Slice LUTs	64000	1373	2%
3	Number of fully used LUT-FF pairs	1693	740	43%
4	Number of bonded IOBs	680	38	5%
5	Number of BUFG/BUFGCTRLs	32	1	3%
6	Number of DSP48Es	256	6	2%

The HT was further used to compute Envelope of various signals. The block diagram showing the computation process is shown in Fig-3. Fig-4 shows the Envelope(blue) of a modulated sine wave signal(yellow) as seen on an Oscilloscope.

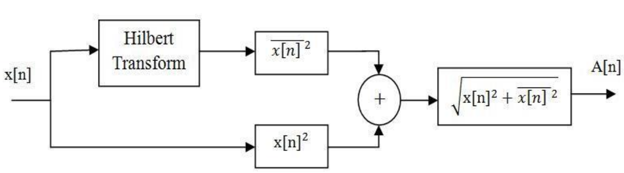

Figure -3 Block diagram for the computation of Envelope using HT module.

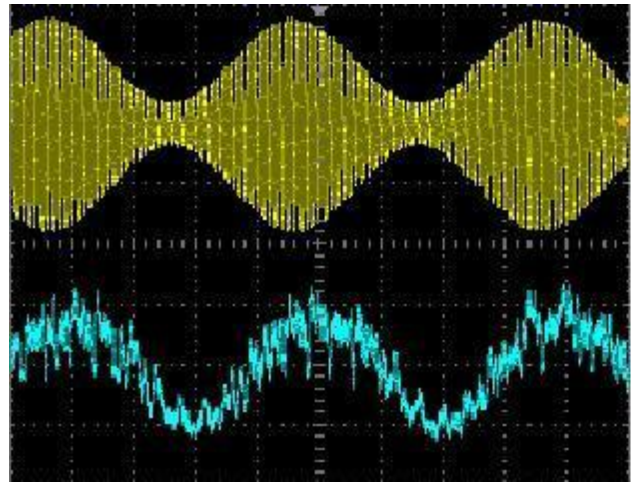

Figure -4 Envelope(blue) of a modulated sine wave signal(yellow) as seen on an Oscilloscope.

C. SQNR

In order to compute the signal-to-quantization noise ratio (SQNR),a sinusoidal modulated signal is given as the input to the pipeline HT. A Matlab script generated double precision floating point HT results, which were used as the true values and compared with the simulated output.Fig-5 shows the plot The mean value for noise was 0.0041 for a 1.0 amplitude signal.

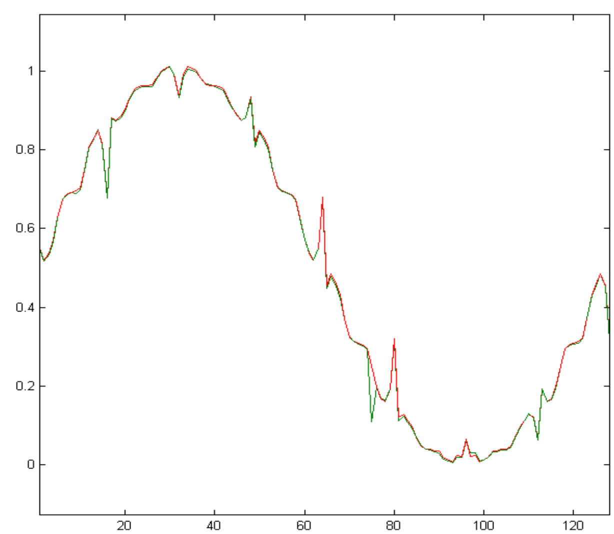

Figure-5 Envelope using MATLAB generated double precision floating point HT results(Green) and the simulated output(Red).

V. CONCLUSION

Implementation of an HT using Radix-2^2 FFT is presented which is further used in Envelope detection and an acceptable SQNR has been achieved.

REFERENCES

[1] H. Olkkonen, P. Pesola, J. T. Olkkonen, “Computation of Hilbert Transform via Discrete Cosine Transform”. *Journal of Signal and Information Processing*, 2010, 1, 18-23.

[2] AHMED SAEED, M. ELBABLY, G. ABDELFADEEL and M. I. ELADAWY.“FPGA implementation of Radix-2^2 Pipelined FFT Processor”. Proceedings of the 8th WSEAS International Conference on Signal Processing.

[3] Abdulnasir Hossen “A NEW FAST APPROXIMATE HILBERT TRANSFORM WITH DIFFERENT APPLICATIONS” IIUM Engineering Journal, Vol. 2, No. 2, 2001

[4] Bin Zhou, Yingning Peng, and David Hwang, “Pipeline FFT Architectures Optimized for FPGAs”. Hindawi Publishing Corporation International Journal of Reconfigurable Computing Volume 2009, Article ID 219140.

[5] Carlos Fritsch, Alberto Iba˜nez, and Montserrat Parrilla.” A Digital Envelope Detection Filter for Real-Time Operation” IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, VOL. 48, NO. 6, DECEMBER 1999.

[6] User Guide, “Virtex-5 FPGA XtremeDSP Design Considerations,” Version 3.3, Xilinx Inc., Jan. 2009.