Low Power Reconfigurable Hilbert Transformer Design with Row Bypassing Multiplier on FPGA

Meenakshi aggarwal, Richa Barsainya, Tarun Kumar Rawat
Division of Electronics and Communication Engineering
Netaji Subhas Institute of Technology
New Delhi-110078
Email: mishaagg@gmail.com, richa.barsainya@gmail.com, tarundsp@gmail.com

***Abstract*—Reconfigurability and low power have always been the main concern for the efficient filter implementation. This paper introduces two new low power and high speed reconfigurable Hilbert transformer designs. These designs are based on the carry save adder (CSA) and ripple carry adder (RCA) based row bypassing multipliers. The primary power reduction is procured by turning off adders when the multiplier operands are zero. In addition, the proposed Hilbert transformers are implemented with parallel architecture of multipliers to shorten the delay time. The proposed designs can be dynamically reconfigured with arbitrary coefficients that are only limited by their length and word size. These Hilbert transformers have been implemented and tested on Vertex-IV field programmable gate array (FPGA) board. The effectiveness of the proposed design method is presented with an example. The performance of both the designs is evaluated in terms of area (number of slices), speed, i.e., maximum frequency and power consumption. The results depict that the CSA row bypassing multiplier based Hilbert transformer achieves 17% increase in speed and 13% area reduction in comparison with RCA row bypassing multiplier based Hilbert transformer. While the power dissipation of the later transformer is 65% less than the former one.**

Keywords: FPGA, Hilbert transformer, IIR digital filters, row bypassing, carry save adders, ripple carry adders

I. Introduction

Rapid technological change has forced manufacturing to face a new economic objective: reconfigurability, i.e., the capability of reconfigurable computing of a system, so that its behavior can be changed by reconfiguration. Power consumption is another important issue in the design of today and future digital systems. For a long battery lifetime in portable devices such as medical equipments, low power consumption is required. Recently, with the onset of software defined radio (SDR) technology, digital signal processing and biomedical engineering, research is now focused on low power reconfigurable realizations of digital filters. Recently, many reconfigurable finite impulse response (FIR) digital filters have been proposed in literature [1]-[3]. It is known that the FIR filter consumes more computational power compared to an infinite impulse response (IIR) filter with similar sharpness or selectivity, particularly when low frequency cutoffs are needed. Also, the IIR filter realization has an advantage over FIR filter realization in term of number of coefficients and statistical performance [4], [5]. In our paper, we have utilized one of the known IIR filter digital structure, i.e.,direct form II for the realization of Hilbert transformer.

In signal processing, Hilbert transformer is considered as an important tool and find applications in different areas like digital communication where it is used for single side-band modulation and edge detection of digital images [6], [7], [8]. Previously FIR and IIR digital based Hilbert transformers have been developed using several methods namely the Remez exchange algorithm [9], eigen filter method [10], and weighted least square method [11]. Various methodologies for implementing the Hilbert transformer were also investigated which comprises of switched-capacitor implementation [12], neural network [13], and multiplier-less triangular array realization [14]. However, it is observed that these approaches are suitable for the fixed coefficient applications. The FPGA implementation of fast fourier transform (FFT) based Hilbert transformation is presented in [15] [16]. Although, these filters are not reconfigurable in nature. Whereas in current scenario, reconfigurability is demanded. This problem is dealt in this paper. In order to design a low power reconfigurable digital filter, one should focus on multipliers to make them efficient as these are the most area and power consuming elements. Hence, by reducing the power consumption in multipliers a large power can be saved. In a logic circuit, the power dissipation can be distinguished as static and dynamic power dissipation. The static power consumption is proportional to the number of transistors used. While the dynamic power dissipation depends upon charging and discharging of load capacitance [17]. The average dynamic power dissipation of a CMOS gate is

$$P_{avg} = \frac{1}{2} C f V_{dd}^2 N \tag{1}$$

where, C is the load capacitance, V_{dd} is the power supply voltage, f is the clock frequency, and N is the number of switching activity in a clock cycle. Thus, by reducing the switching activity of a given logic circuit, the power consumption can be reduced without altering its function.

In this work, two new reconfigurable Hilbert transformers based on low power, row bypassing multipliers are proposed. Hilbert transformers are implemented using two multipliers proposed in [18] and [19], respectively. The multiplier design [18] is using carry save adders and a final ripple carry adder for its implementation. While in [19] the multiplier is based on only ripple carry adders. The filter coefficients are directly saved into look-up-table (LUT). These coefficients are ac-

978-1-4673-9197-9/16/$31.00 ©2016 IEEE

cessed by multipliers. The proposed designs are implemented on Vertex-IV field programmable gate array (FPGA) board. The performance is compared in terms of speed, i.e., maximum frequency, area (number of slices) and power consumption. The proposed Hilbert transformer can be implemented and utilized in those applications where filter coefficients need to change such as in communication systems.

The paper is structured as follows. The basic concepts of Hilbert transformer is explained in section II. In Section III, the methodology of bypassing multipliers is explained. The FPGA implementation of Hilbert transformer is described in Section IV. In Section V, design example and comparative results are presented. Conclusion is summarized in Section VI.

II. Hilbert transformer

It is known that there exists an explicit relation between the discrete Hilbert transformer and complex half-band filter. Complex half-band filter satisfies frequency domain constraints of the Hilbert transformer [5]. The frequency response of the ideal Hilbert transformer is characterized as

$$H_{HT}(e^{j\omega}) = \begin{cases} j, & -\pi < \omega < 0 \\ -j, & 0 < \omega < \pi \end{cases} \quad (2)$$

The complex half-band filter can be easily procured by adding a shift of $\frac{\pi}{2}$ radians in the real half-band filter's frequency response [5]. The real half-band filter $G(z)$ can be written as

$$G(z) = \frac{1}{2}[A_1(z^{-2}) + z^{-1}A_2(z^{-2})] \quad (3)$$

where, $A_1(z)$ and $A_2(z)$ represents are stable allpass filters. After applying frequency transformation, complex half band filter can be procured from half-band filter [20] using

$$H(z) = jG(-jz) \quad (4)$$

The resultant complex half-band transfer function is expressed as

$$H(z) = \frac{1}{2}[A_1(-z^{-2}) + jz^{-1}A_2(-z^{-2})] \quad (5)$$

where $A_1(-z^{-2})$ and $A_2(-z^{-2})$ represents real and stable allpass filter. It is to note that if $G(z)$ is a half-band lowpass filter with its passband on the right half of the unit circle, then by applying the frequency transformation, $H(z)$ becomes the complex half-band filter with its passband on the upper half of the unit circle [21]. The complex half-band realization using allpass filter is shown in Fig. 1.

The allpass filter block in the complex half-band filter realization is replaced with the direct form II structure of allpass filter to obtain the canonic structure to Hilbert transformer. In canonic realization, multiplier coefficients of the structure are mainly the filter coefficients [5]. The required number of coefficients for the realization of the Nth order IIR filter are $2N + 1$ using canonic structures.

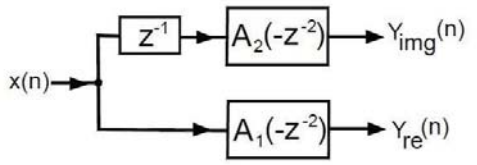

Fig. 1. Complex half-band filter realization using Allpass filter

III. Low power multiplier with row bypassing

The power consumption of a multiplier can be reduced by turning off the components through multiplexers when multiplier operands are zero. The average of zero input of the operand in the multiplier is found as 73.8% in conventional DSP applications [19]. Therefore, to reduce power consumption and increase in speed, row bypassing techniques are used in multipliers. These bypassing multipliers can be designed using carry save adders and ripple carry adders described as follows.

A. Row bypassing multiplier based on CSA

The purpose of row bypassing multiplier [18] is to disable adders in the j_{th} row if the bit y_j is zero in the multiplier, i.e., all the bits in the $x_i y_j$, $0 \leq i \leq n-1$, are zero, where n is the operands wordlength. Hence, the power dissipation can be reduced. The conventional full adder cell needs to be modified in order to disable the adder of a specific row as shown in Fig. 2.

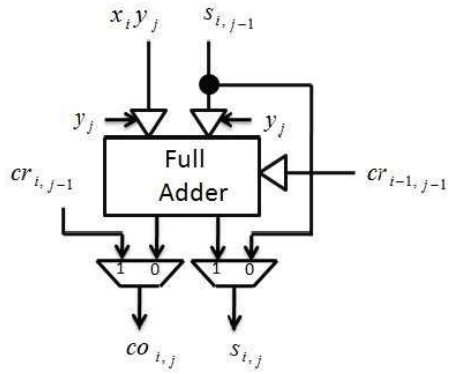

Fig. 2. Modified full adder cell

Where x_i, y_j and cr are the three inputs which are applied with tri-state buffers and $s_{i,j}$ and $co_{i,j}$ are the outputs. At the outputs two multiplexers are used to perform bypassing technique. The applied tri-state buffer take a decision to disable the adder when the value of multiplier bit y_j is zero. The multiplexers are used to select correct outputs. In Hilbert transformer the input vectors and coefficients may be positive or negative, therefore, we need to use signed multiplier. Therefore, a 8×8 signed Braun multiplier is designed using the modified adder cell shown in Fig. 2.

B. RCA based row bypassing multiplier

The RCA based row bypassing multiplier is presented in [19]. The basic adder cell has two inputs and one output. Hence, only two tri-state buffers are required to disable the adder operation if the bit y_j is zero in the multiplier. Also, only one multiplexer compared to CSA based multiplier is required to select correct output. The ripple carry adders are rather slow compared to carry save adders due to the longer critical path. Therefore, speed is enhanced by a parallel architecture of an 8×8 signed multiplier which is designed using two 8×4 multiplier blocks. The block diagram of 8×8 multiplier is depicted in Fig. 3. An 8×8 signed multiplier [19] is used to implement Hilbert transformer in this paper.

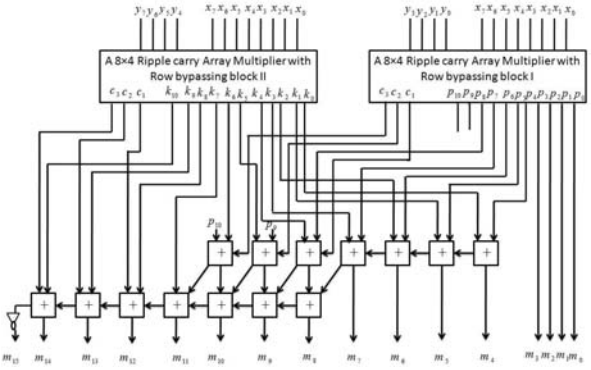

Fig. 3. Block diagram of 8×8 signed row bypassing multiplier using carry save adders

IV. FPGA IMPLEMENTATION

The Hilbert transformer structure depicted in Fig. 1 has been implemented in two ways: using CSA row bypassing based multiplier and using RCA row bypassing based multiplier explained in the previous section. The coefficients are directly saved into LUT to make the operation faster. One more advantage of storing the coefficients into LUT is that the same coefficients need to save only once and can be fetched as many times they are required. Hence, the less number of memory locations may be required to store coefficients. The proposed designs can be dynamically reconfigured with arbitrary coefficients that are only limited by their length and word size. The structures of Hilbert transformer are coded in Verilog HDL language and synthesis, translate, map, place and route of the designs are done using Xilinx ISE 13.4 targeting Vertex IV (xc4vsx25-10ff668) FPGA device [22], [23]. These structures are tested and simulated on ISim simulator by applying different input test vectors. The Power dissipation of these filters is estimated by XPower tool. A design example of Hilbert transformer is considered in the following section.

V. DESIGN EXAMPLE

In this section the implementation of Hilbert transformer is shown by an example. The direct form II structure of a Hilbert transformer is implemented on Xilinx Vertex IV FPGA. The FPGA is programmed using a combination of Xilinx core generation and Verilog (HDL) code. The input signals are the parallel inputs, global clock and a reset.

The specifications of real half band filter are as follows: stopband edge frequency, $\omega_s = 0.6\pi$ and stopband ripple, $\delta_s = 0.016$. For these specifications, the transfer function of the real-half band filter is given by

$$H(z) = \frac{1}{2}\left[\left(\frac{0.236471021 + z^{-2}}{1 + 0.236471021z^{-2}}\right) + z^{-1}\left(\frac{0.7145421497 + z^{-2}}{1 + 0.7145421497z^{-2}}\right)\right] \quad (6)$$

By applying frequency transformation, the resultant complex half-band filter transfer function is obtained as

$$H(z) = \frac{1}{2}\left[\left(\frac{0.236471021 - z^{-2}}{1 - 0.236471021z^{-2}}\right) + jz^{-1}\left(\frac{0.7145421497 - z^{-2}}{1 - 0.7145421497z^{-2}}\right)\right] \quad (7)$$

where

$$A_1(-z^{-2}) = \frac{0.236471021 - z^{-2}}{1 - 0.236471021z^{-2}}$$

and

$$A_2(-z^{-2}) = \frac{0.7145421497 - z^{-2}}{1 - 0.7145421497z^{-2}}$$

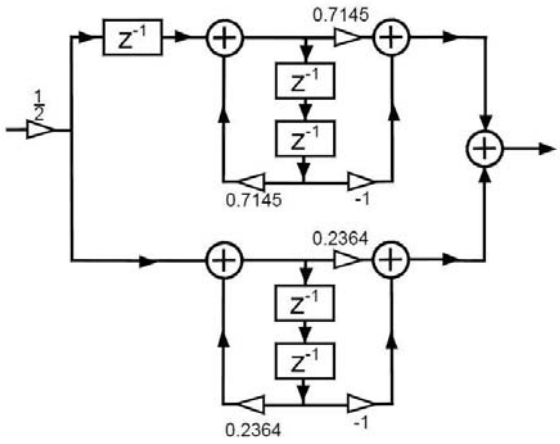

Fig. 4. Direct form II realization of Hilbert Transformer

Direct form II realization of Hilbert transform for given example is shown in Fig. 4 and its magnitude response is depicted in Fig. 5. The performance of Hilbert transformer presented in the example is implemented with the two bypassing multipliers and compared in Table I in terms of delay (minimum period), speed (maximum frequency) and dynamic power dissipation. The hardware utilization of the two approaches is summarized in Table II. The comparative results show that the power dissipation of CSA row bypassing based Hilbert transformer is 65% more than RCA row bypassing based Hilbert transformer. The reason is more switching activity involved in former transformer as the number of multiplexers are almost double to the later. Whereas, the minimum period is decreased by 14.5% and maximum frequency is improved by 17% in CSA rowbypassing based transformer. Therefore, there is a tradeoff between speed and power dissipation. Similarly, from

TABLE I. DELAY, SPEED AND THE TOTAL POWER DISSIPATION FOR HILBERT TRANSFORMER

Type	Minimum Period	Maximum Frequency (MHz)	Dynamic Power Dissipation (mW)
CSA row bypassing based	7.792	128.334	23.13
RCA row bypassing based	9.115	109.708	13.98

TABLE II. HARDWARE UTILIZATION SUMMARY OF HILBERT TRANSFORMER

Type	4 input LUTs			Slices			Slice flip-flops		
	used	Available	Utilization	used	Available	Utilization	used	Available	Utilization
CSA row bypassing based	246	20480	1%	144	10240	1%	40	20480	1%
RCA row bypassing based	292	20480	1%	166	10240	1%	39	20480	1%

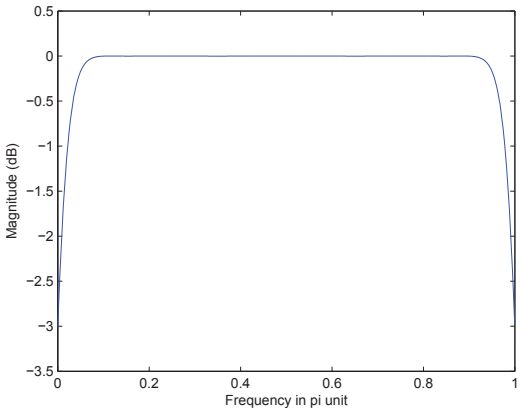

Fig. 5. Passband magnitude response of Hilbert Transformer

Table II, it is observed that area (slices) required for CSA bypassing multiplier based transformer is 13% less compared to the second transformer design. The choice of the design can be made according to the application. For comparison of the speed and area of the RCA and CSA rowbypassing based multipliers, many random input samples are applied in testbench and corresponding results are observed. The performance of the digital filters depends upon the value of filter coefficients and input samples, as the number of nonzero bits in coefficients increases, the hardware and power dissipation may vary. Hence, in this work, the power, speed and area are considered as an average of the synthesis results in all the tables. For other examples, the comparison results of the speed, area and power dissipation may vary according to the number of nonzero coefficient bits. Hence, the power, speed and area values of the synthesis results are dependent on the nonzero bits in filter coefficients and value of input samples and hence we have considered an average of the synthesis results in all the tables in this paper.

VI. CONCLUSION

The efficient implementation of low power and high speed Hilbert transformer is proposed. The primary power reductions are obtained by turning off adders when the multiplier operands are zero. Therefore, CSA and RCA based row bypassing multipliers are used to design a Hilbert transformer. In addition, the ripple carry adders are rather slow compared to carry save adders due to the longer critical path. Therefore, speed of RCA rowbypassing multiplier based transformer is enhanced by a parallel architecture of multiplier to shorten the delay time. The proposed designs can be dynamically reconfigured with arbitrary coefficients that are only limited by their length and word size. These transformers are implemented and tested on Xilinx Vertex-IV xc4vsx25-10ff668 FPGA device family. The performance analysis of the two implementation is made for speed, area and power dissipation. For comparison of the speed and area of the RCA and CSA rowbypassing based multipliers many random input samples are applied using testbench for the filter coefficients given in the example. The results indicate that CSA row bypassing multiplier based Hilbert transformer is better in terms of speed and area while RCA row bypassing multiplier based Hilbert transformer is better in terms of power dissipation. Among these the designer can choose the optimum Hilbert transformer structure for a specific application.

REFERENCES

[1] K. H. Chen and T. D. Chiueh, "A low-power digit-based reconfigurable FIR filter," IEEE Trans. Circuits Syst. II, vol. 53, no. 8, pp. 617–621, Aug. 2006.

[2] M. Kumm, K. Mller and P. Zipf, "Dynamically Reconfigurable FIR Filter Architectures with Fast Reconfiguration," International Workshop on Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC), pp. 1–8, July 2013. DOI 10.1109/ReCoSoC.2013.6581517.

[3] R. Mahesh and A. P. Vinod, "New reconfigurable architectures for implementing FIR filters with low complexity," IEEE Trans. Comput. Aided Design Integr. Circuits Syst., vol. 29, no. 2, pp. 275–288, Feb. 2010.

[4] R. Ansari, "IIR discrete-time Hilbert transformers," IEEE Trans., vol. I, ASSP-35, no. 8, pp. 1116–1119, Aug. 1987.

[5] S. K. Mitra, "Digital signal processing," Mc Graw Hill, 4th ed., 2013.

[6] S. L. Hahn, "Hilbert Transforms in Signal Processing," Norwood, MA: Artech House, 1996.

[7] P. Duraiswamy, J. Bauwelinck and J. Vandewege, "Efficient implementation of 90 phase shifter in FPGA," EURASIP Journal on Advances in Signal Processing, no. 1, pp. 1–5, 2011.

[8] K. Kohlmann, "Comer detection in natural images based on the 2-D Hilbert transformer," Journal of Signal Processing, vol. 48, pp. 225–234, 1996.

[9] J. H. McClellan, T. W. Parks and L. R. Rabiner, "A computer program for designing optimum FIR linear phase digital filters," IEEE Trans. Audio Electro acoust., vol. AE-21, pp. 506–526, Dec. 1973.

[10] S. C. Pei and J. J. Shyu, "Design of FIR Hilbert transformers and differentiators by eigenfilter," IEEE Trans. Circuits Syst., vol. 35, pp. 1457–1461, Nov. 1988.

[11] S. Sunder and V. Ramachandran, "Design of equiripple nonrecursive digital differentiators and Hilbert transformers using weighted least-squares technique," IEEE Trans. Signal Processing, vol. 42, no. 9, pp. 2504–2509, Sept. 1994.

[12] K. P. Pun, J. E. Franca and C. A. Leme, "Polyphase SC IIR Hilbert transformer," Electron. Lett., vol. 35, pp. 689–690, Apr. 1999.

[13] A. Hiroi, K. Endo, H. Kamata and Y. Ishida, "Design and implementation of an ideal Hilbert transformer using neural networks," In Proc. IEEE Pacific Rim Conf. Communication, Computer and Signal Processing, vol. I, pp. 292–295, 1993.

[14] S. Samadi, Y. Igarashi and H. Iwakura, "Design and multiplierless realization of maximally flat FIR digital Hilbert transformers," IEEE Trans. Signal Processing, vol. 47, pp. 1946–1953, July 1999.

[15] A. Rani, R.M. Verma and Saurabh Jaiswal, "FPGA implementation of Hilbert transform via radix-2-2 pipelined FFT processor," Computing, Communications and Networking Technologies (ICCCNT), 2013 Fourth International Conference on. IEEE, 2013.

[16] A. Amalin Prince, Prakhar K. Verma, C. Jayakumar and Daniel Raju, "Efficient architecture for real time implementation of Hilbert Transform in FPGA," Electrical, Computer and Communication Technologies (ICECCT), 2015 IEEE International Conference on. IEEE, 2015.

[17] K.K. Parhi, VLSI Digital Signal Processing Systems: Design and Implementation, John Wiley, 1999.

[18] J.Ohban, V.G.Moshnyaga and K.Inoue, "Multiplier energy reduction through bypassing of partial products," In proceedings of the IEEE Asia-Pacific Conference on Circuits and Systems, vol. 2, pp.13-17, 2002.

[19] Ko-Chi Kuo n, Chi-WenChou, "Low power and high speed multiplier design with row bypassing and parallel architecture," Microelectronics Journal, vol. 41, pp. 639-650, 2010.

[20] S. A. Samad, A. Hussain and D. Isa, "Wave digital filters with minimum multiplier for discrete Hilbert transformer realization," International Journal of Signal processing, vol. 86, no. 12, pp. 3761–3768, 2006.

[21] P. A. Regalia, S. K. Mitra and P. P. Vaidyanathan, "The digital all-pass filter: A versatile signal processing building block," Proc. of IEEE, vol. 76, pp. 19–37, Jan. 1988.

[22] Roger Woods, John McAllister, Gaye Lightbody and Ying Yi, "FPGA-based Implementation of Signal Processing Systems," Wiley Publication, 1st ed., 2008.

[23] www.xilinx.com/support/documentation, ISE In-Depth Tutorial, UG695 (v14.1) April 24, 2012.