

FPGA implementation of a de-noising using Haar level 5 wavelet transform

Euclides L. Chuma, Luís G. P. Meloni, Yuzo Iano and Leonardo L. Bravo Roger

Abstract— This paper proposes an implementation in Field Programmable Gate Array (FPGA) of a de-noising using Haar wavelet transform. For this a signal with noise was applied at a Haar level 5 Discrete Wavelet Transform (DWT) through an threshold and then through an Inverse Wavelet transform (IDWT). The design procedure has been designed using the Electronic Design Automation (EDA) tools for system design on FPGA. Simulation, synthesis and implementation on the FPGA target technology has been realized.

Keywords— Haar, wavelet, DWT, de-noising, FPGA.

I. INTRODUCTION

Currently the discrete wavelet transform (DWT) is one of the most popular of the time-frequency-transformations because have temporal resolution and so it captures information of frequency and location in time unlike other transforms as Fourier that not bring the information of the temporal location.

In the wavelet transform the signals are analyzed at multiple scales by changing the width of the analysis window, and produces their scale-space representation. The wavelet transform deals with the limitations of uncertainty principle and there a trade-off between the length of the sliding window (i.e. the spatial resolution) and the distance between adjacent spectral lines (i.e. the frequency resolution). So the wavelet transform use longer windows (i.e. better frequency resolution) for low frequency components of the signal and shorter windows (i.e. better spatial resolution) for high frequency components of short duration.

Recently the Field Programmable Gate Array (FPGA) has gained ground in many applications of microelectronic because have a relatively high capacity and low cost and also the FPGA has a design flexibility and adaptability of software implementations. Therefore the FPGA is a good choice to implementation platform for wavelet transform.

The FPGA implementation of Haar discrete wavelet transform already been used in others works in applications with real-time and high-performance requirements like de-noising in images [1], face detection algorithm [2] and electrocardiogram (ECG) signal filtering [3].

To development the de-noising was used a coding algorithm composed by three parts: DWT, threshold and IDWT as show the Figure 1.

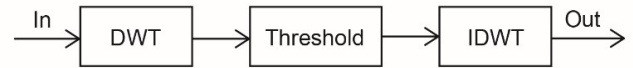


Fig. 1. Parts of the algorithm of the de-noising.

In this paper the Haar wavelets is initially presented. Then is introduced the discrete wavelet transform (DWT) the Inverse discrete wavelet transform (IDWT) and threshold method. Finally the results are presented followed by a conclusion.

II. HAAR WAVELET

The Haar wavelet is the simplest type of wavelet and was the first DWT invented by Hungarian mathematician Alfréd Haar. The Haar transform decomposes a discrete signal into two subsignals of half length. One subsignal is running the average or trend and the other subsignal is running the difference or fluctuation. So the function $\psi(x) \in W_j$ is known by Haar wavelet function and given by [4]:

$$\psi(x) = \begin{cases} 1, & \text{if } 0 \leq x \leq \frac{1}{2} \\ -1, & \text{if } \frac{1}{2} \leq x \leq 1 \\ 0, & \text{for others cases} \end{cases} \quad (1)$$

The Figure 2 show the Haar wavelet function:

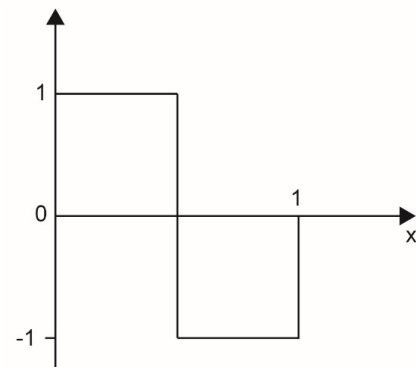


Fig. 2. Haar wavelet function $\psi(x)$.

The Haar wavelet transform into lifting steps has a number of advantages [5]:

- It is simple.
- It is fast.
- It is memory efficient, since it can be calculated without temporary array.
- It is exactly reversible without the edge effects that are a problem with other wavelet transforms.

But the Haar wavelet also has limitations because the Haar wavelet performs an average and difference on a pair of values. So the Haar window have only two elements wide. If a big change occurs from an even value to an odd value, the change will not be reflected in the high frequency coefficients. Thus the Haar wavelet is not appropriate for some types signs such as audio signals [6].

III. DISCRETE WAVELET TRANSFORM

The discrete wavelet transform (DWT) is very popular type of the wavelet transform to signal processing and image processing and compression, inclusive it is part of the JPEG2000 standard [7].

The DWT is computed using the algorithm as show in the Figure 3. In each decomposition stage the DWT produces wavelet coefficients corresponding respectively to the upper half and the lower half of the spectrum of the input signal. This algorithm was development by Mallat [8][9] and it is called as Fast Orthogonal Wavelet Transform.

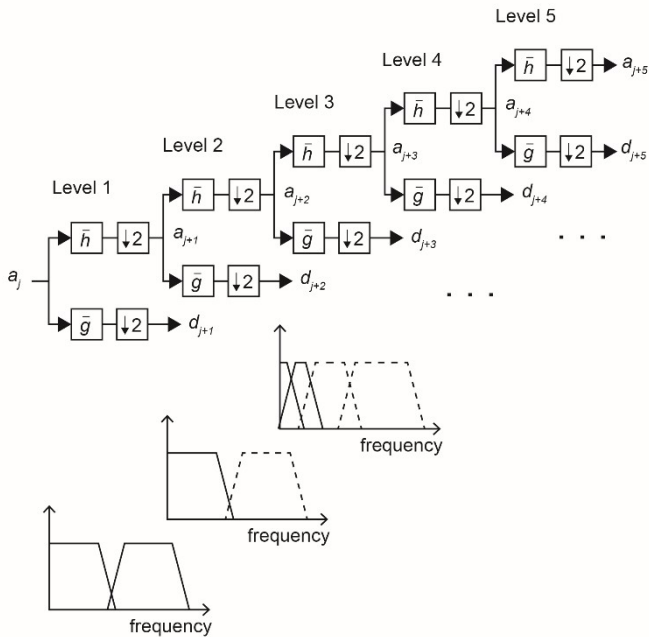


Fig. 3. The algorithm for computing the 1-dimensional DWT with 5 levels.

In the Mallat [8][9] the decomposition is given by:

$$a_{j+1}[p] = \sum_{n=-\infty}^{+\infty} h[n-2p]a_j[n] = a_j * \bar{h}[2p] \quad (2)$$

$$d_{j+1}[p] = \sum_{n=-\infty}^{+\infty} g[n-2p]a_j[n] = a_j * \bar{g}[2p] \quad (3)$$

And the reconstruction is given by:

$$\begin{aligned} a_j[p] &= \sum_{n=-\infty}^{+\infty} h[p-2n]a_{j+1}[n] + \sum_{n=-\infty}^{+\infty} g[p-2n]d_{j+1}[n] = \\ &= \tilde{a}_{j+1} * h[p] + \tilde{d}_{j+1} * g[p] \end{aligned} \quad (4)$$

In the Figure 4 show the Inverse Discrete Wavelet Transform that is the inverse process of the DWT.

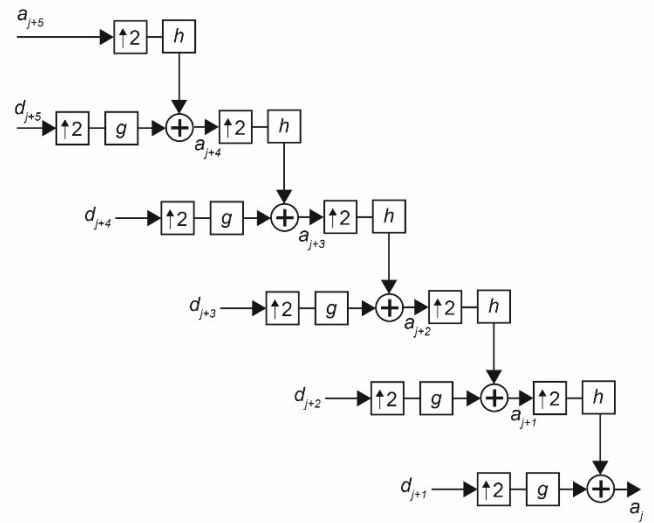


Fig. 4. The algorithm for computing the IDWT with 5 levels.

IV. THRESHOLD

The threshold must present a compromise between removing noise while keep the signal shape. In the implementation used in this paper was used a simple threshold where the values were determined empirically by observing the maximum values of the wavelet coefficients.

V. DESIGN OF HAAR WAVELET TRANSFORM USING FPGA

The programming of the algorithm in behavioral mode was made in the VHDL (VHSIC Hardware Description Language) language that is used to development of digital circuits in FPGAs, CPLDs and ASICs. The code was organized in building blocks and the first block is the FSM (Finite State Machine) that is responsible by control flow and the clock rate. Other building block is used to make the DWT process that basically consist of compute the coefficients and down-sampling. And another block used is responsible by IDWT process that compute the inverse and make the up-sampling. The threshold is simple and is among the DWT and IDWT blocks. In the Meyer-Baese [10] there examples of codes in VHDL to work with wavelets.

The design was implemented using Altera Quartus 15 development environment [11]. As a target chip FPGA

Cyclone V model 5CGXFC9E7F35C8 from Altera [11] was chosen.

The next step was to simulate using ModelSim-Altera. For this was necessary to program a testbench in VHDL to establish the parameters and input data. The input signal with noise was generated in Simulink as can be seen in Figure 5. The input signal with noise was exported from Simulink and coded in VHDL for use in the testbench.

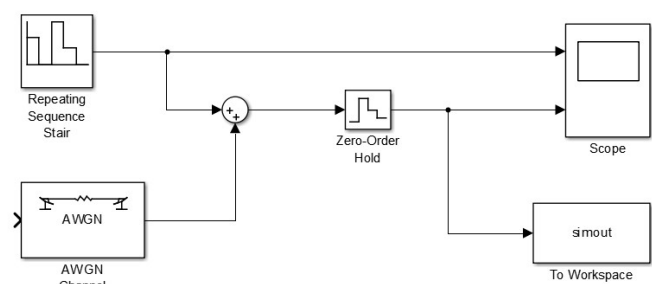


Fig. 5. Simulink diagram for generate the data input to simulation.

VI. RESULTS

The simulation results to the test bench in the ModelSim-Altera can be seen in the Figure 6 and the comparison between the input signal with noise and output signal generated by the simulation can be seen in the Figure 7. Both figures show that there was an overall reduction of noise and especially the noise of the low signal was virtually removed.

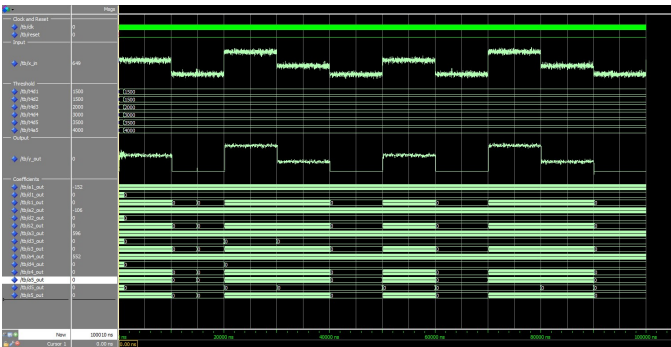


Fig. 6. Simulation result in ModelSim-Altera.

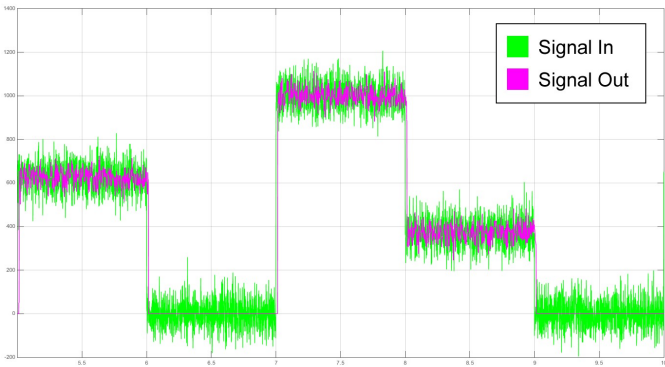


Fig. 7. Comparison between the input signal and the output signal with de-noising.

In Figure 8 is possible to see that the reduction of noise starts 32 samples after that the thresholds was defined, what is expected in a Haar level 5 wavelet transform.

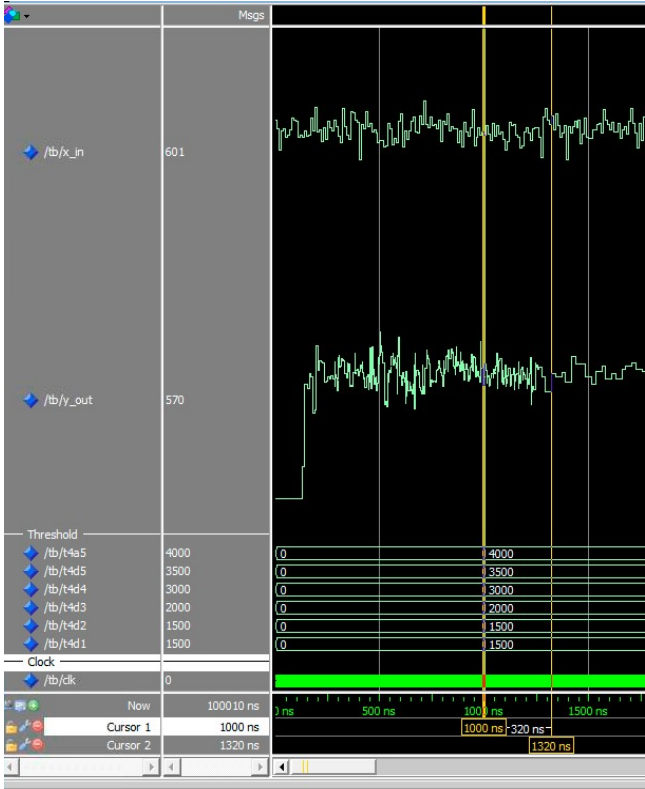


Fig. 8. Thresholds defined and the start of noise reduction.

Even simulation meeting the requirements is no guarantee that the implementation in VHDL will work properly in a real FPGA. So to ensure the correct operation of the de-noising in FPGA is necessary to make the synthesis process.

The results of the synthesis for this FPGA are shown in Figure 9 and in the Table I.

| Flow Summary | |
|---------------------------------|--|
| Flow Status | Successful - Mon Jul 03 22:06:31 2017 |
| Quartus II 64-Bit Version | 15.0.0 Build 145 04/22/2015 SJ Web Edition |
| Revision Name | dwt |
| Top-level Entity Name | dwt |
| Family | Cyclone V |
| Device | 5CGXFC9E7F35C8 |
| Timing Models | Final |
| Logic utilization (in ALMs) | 586 / 113,560 (< 1 %) |
| Total registers | 1156 |
| Total pins | 393 / 616 (64 %) |
| Total virtual pins | 0 |
| Total block memory bits | 3,536 / 12,492,800 (< 1 %) |
| Total DSP Blocks | 0 / 342 (0 %) |
| Total HSSI RX PCs | 0 / 12 (0 %) |
| Total HSSI PMA RX Deserializers | 0 / 12 (0 %) |
| Total HSSI TX PCs | 0 / 12 (0 %) |
| Total HSSI PMA TX Serializers | 0 / 12 (0 %) |
| Total PLLs | 0 / 20 (0 %) |
| Total DLLs | 0 / 4 (0 %) |

Fig. 9. Summary of synthesis in the Altera Quartus.

TABLE I. SYNTHESIS RESULTS.

| Parameters | Resource utilization |
|---|---------------------------------|
| Logic utilization (in ALMs) | 586 / 113,560 (< 1 %) |
| Total block memory bits | 3,536 / 12,492,800 (< 1 %) |
| Maximum Frequency (TimeQuest Fast 1100mv 85C Model) | 295.6 MHz |

VII. CONCLUSIONS

This paper proposed an implementation of the Haar wavelet transform with 5 levels using FPGA technology. The design was tested and the results are presented. Based on the results obtained by the simulation of the implementation it is possible say that the FPGA gives a fast and reliable platform to make the wavelet transform and inverse wavelet transform. Also was observed that the threshold is important and others algorithms must be tested.

ACKNOWLEDGMENT

The authors would like to thanks CAPES, CNPq, and Fapesp; also the other teachers and staff of DECOM, FEEC, and UNICAMP.

REFERENCES

- [1] RAMESHA, K. P., RAMANAREDDY, K. V., DR. YELLAMPALLI, S., *FPGA Implementation of Image De-noising using Haar Wavelet Transform*, International Journal of Engineering Research & Technology (IJERT), vol. 5, issue 5, 2016.
- [2] GAO, C., LU, S.-L.L., SUH, T., LIM, H. *Field programmable gate array-based Haar classifier for accelerating face detection algorithm*, IET Image Processing, vol. 4, issue 3, pp. 184–194, 2010.
- [3] VIJENDRA, V., MEGHANA, K. *ECG signal Filtering using DWT haar wavelets Coefficient techniques*, International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS), 2016.
- [4] HAAR, A. *Zur theorie der orthogonalen funktionensysteme*. Math. Annal., 69, pp. 331–371, 1910.
- [5] DAUBECHIES, Ingrid, SWELDENS, Wim. *Factoring wavelet transforms into lifting steps*. Sep. 1996
- [6] WALKER, James S. *A Primer on Wavelets and Scientific Applications*. CRC Press LLC. 1999
- [7] JPEG. *JPEG2000*. Joint Photographic Experts Group. Retrieved 2009-11-01. <https://jpeg.org/jpeg2000/index.html>
- [8] MALLAT, S. *An efficient image representation for multiscale analysis*. Proc. Machine Vision Conference, February 1987.
- [9] MALLAT, S. *A theory for multiresolution signal decomposition: The wavelet representation*. IEEE Trans. Patt. Anal. Mach. Intell., 11(7): 674–693, 1989.
- [10] MEYER-BAESE, Uwe. *Digital Signal Processing with Field Programmable Gate Arrays*. ed. 4, Springer, 2014
- [11] ALTERA. *Cyclone V Device Handbook*. 2015