FPGA Implementation of Image Watermarking Process using Xilinx SystemGenerator

Chandani Sharma, Garima Gupta, Ankit Sharma

Associate Professor, Department of CSE, MMICTBM(MCA), Maharishi Markandeshwar (Deemed to be) University, Mullana, Ambala Haryana
Assistant Professor, Department of ECE, AKTU, Lucknow
Associate Professor, CSE Department, Chandigarh University, Chandigarh.

Abstract. Watermarking is a tool of confidentiality that protects valuable information hidden in the set of digital media by modifying the particular graphic contents. Special purpose hardware designed systems can be mapped in watermarking through FPGA. In this paper, the process of watermarking is performed on image using Xilinx system generator. Watermarking is performed in DWT domain using MATLAB after scaling of watermark data in Simulink environment. The co-hardware simulation has been performed for developed model using VIRTEX 6, ML605 BOARD to see the capability of watermarking system. The Simulink and system Generator through FPGA model extracts the Watermark appropriately and thereafter device utility for embedding system is determined.

Keywords:Image Watermarking, DWT, Scaling, System Generator, Co-hardware Simulation.

1 INTRODUCTION

The recent advances in communication and digital systems have brought extensible innovation opportunities in designing prototype systems and exploring its applications. Digitized data accelerates easy way users edit data and replicate it in case visual effects loss is to be maintained. This leads to protection of content and make it authorized access protecting intellectual property rights. To solve issues of authentication and copyright protection the process of digital watermarking is introduced.

A lot of research is already carried out in Hardware implementation of the Digital Watermarking system using FPGA. For video authentication, real time implementation of invisible watermarking is done using FPGA[9]. Watermarking is explored using Simulink

block in MATLAB and then the algorithm is converted into Hardware Description Language (HDL) using Xilinx System Generator tool [10].

The algorithm is prototyped in Virtex-6 (vsx315tff1156- 2) FPGA. Real time implementation of Digital Watermarking for Image and Video [4] using DCT domain. The embedding for invisible, semi fragile watermark infortion into compressed video streams using DCT is also worked upon [1]. Xilinx System generator and use of HAAR DWT for watermarking[3] and ML507 Evaluation Platform which is based on the Virtex-5 FPGA using Xilinx System Generator tool [2]. The co-hardware simulation for video watermarking and implementation of design using a Spartan3 device (3S200PQ208) then a Virtex II Pro (xc2vp7- 6ff672) [4]. FPGA and ASIC implementation of color image watermarking. The use of pseudo noise code enhances the security of watermarking scheme [5]. Xilinx Spartan 3E FPGA kit with XC3S500E device is used for performing the task. Xilinx ISE 14.7 project navigator along with XST synthesis tool and ISim simulator are used for interfacing, RTL synthesis and simulation respectively. With the development of technology, efficient and rapid prototyping systems have emerged in current scenario. These systems require a development environment targeting the hardware design platform. Creating specialized hardware would greatly reduce the time consumed by these processes. Simulink in MATLAB is an environment for modeling and simulation.

This paper presents a work where watermarking is first modeled and simulated on MATLAB-Simulink environment on XSG tool and further co-simulation is done on ML605 board. This is represented through methodology given below in Fig 1.

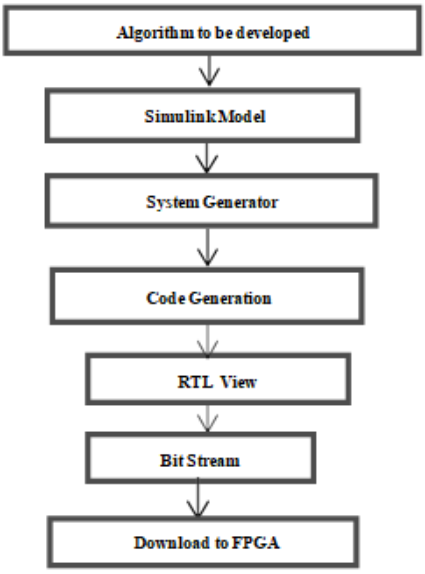


Fig 1: Design Methodology with Xilinx System

System Generator is a design tool based on MATLAB-Simulink. Xilinx is a family member of the System Generator tool. The tool provides high-level generalizations that are

automatically amassed into an FPGA at the push of a button. The Xilinx Integrated Software Environment (ISE) is a powerful design environment that is working in the background while implementing System Generator blocks. When XSG is configured with MATLAB, it activates Xilinx block set library to appear in MATLAB-Simulink environment.

With the help of Simulink and Xilinx block set, model is designed and successfully run; a net list is automatically generated for the given model. Then schematic representation of our synthesized source file can be displayed with the help of RTL view. This schematic shows a representation of the pre-optimized design in terms of standard symbols such as multipliers, counters, adders, AND gates, and OR gates that are independent of the targeted Xilinx device. This schematic also calculates the actual resource utilization for the designed model. Then mapping of synthesize design to physical resource of the target device is done to create implemented design. In this paper, hardware implementation of image watermarking is presented on ML605 Board.

PRELIMINARY MODEL GENERATION

Low Pass Filter and High Pass Filter

For 2-D DWT implementation, the image taken is first converted into four sub bands- LL, LH, HL and HH. This frequency wise decomposition separates the image into high frequency and low frequency components. Frequency components can be separated by using high pass and low pass filters.

Fig. 2 shows the 2-D decomposition of input image. Here, g0 [n] is low pass filter and h0 [n] is high pass filter representation. After down sampling four components (LL, LH, HL and HH) are obtained as shown in Fig. 2.

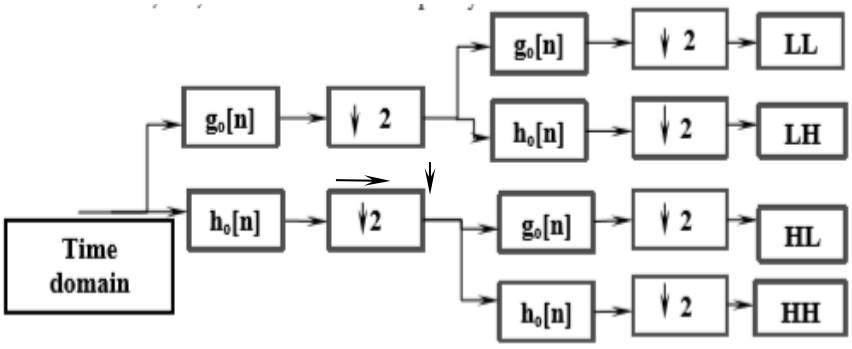


Fig 2: 2-D Decomposition Routine

Xilinx FIR Compiler 5.0 of XSG has been used to design LPF and HPF which are further used to convert the image into four sub- bands LL, LH, HL, HH as shown in Fig 2. The FDA Toolbox is used to define the filter order and coefficients for FIR compiler as shown in Fig 3.

Having proper parameter settings of FDA Toolbox and Xilinx FIR Compiler 5.0 block, filter coefficients are automatically added from FDA Toolbox in both LPF and HPF filters designed with FIR Compiler as shown in Fig. 4.

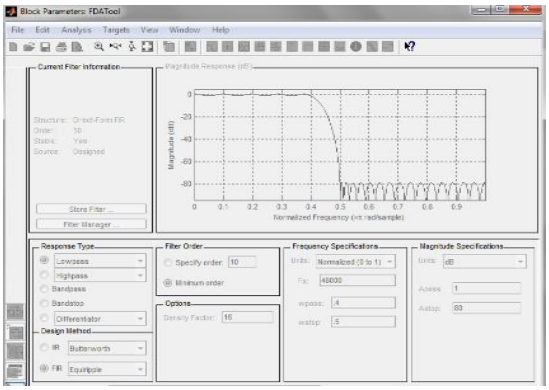


Fig 3: FDA Tool for Filter Designing

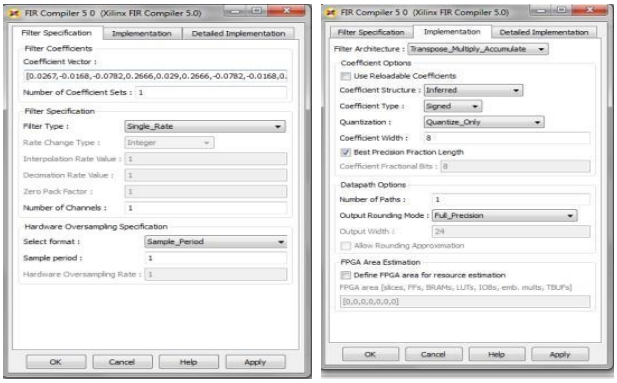


Fig 4: LPF and HPF using Xilinx

Coefficients for LPF and HPF are shown in Fig. 5. This figure shows four tables of filter coefficients- two for embedding and two for extraction. The output of the filters is displayed using spectrum scope.

Index Number	Coefficient
(=0	0.029
(=+1,-1	0.2666
(=+2,-2	-0.0782
(=+3,-3	-0.0168
(=+4,4-	0.0267

(a)

Index Number	Coefficient
(=0	1.115
(=+1,-1	0.591
(=+2,-2	-0.0575
(=+3,-3	-0.0912
(=+4,4-	0

(b)

Index Number	Coefficient
(=0	1.1150
(=+1,-1	-0.592
(=+2,-2	-0.0575
(=+3,-3	0.0912
(=+4,4-	0

(c)

Index Number	Coefficient
(=0	0.6029
(=+1,-1	-0.2668
(=+2,-2	-0.0782
(=+3,-3	0.0168
(=+4,4-	0.0267

(d)

Fig 5: Filter coefficients
(a) LPF Coefficients for Embedding (b) LPF Coefficients for Extraction

(c) HPF Coefficients for Embedding (d) HPF Coefficients for Extraction

WATERMARK EMBEDDING

The DWT based image watermark embedding method is used in this experiment. Both input and watermark images are either black and white image or color image. Fig. 6 shows the block diagram for watermark embedding and Fig. 7 shows the XSG implementation for watermark embedding.

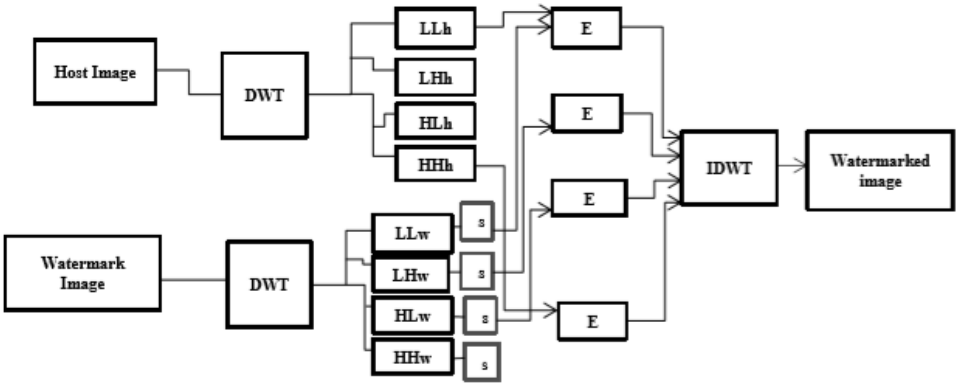


Fig 6: Block Diagram for Watermark Embedding with Host (Pepper Image) and Watermark (Leena Image)

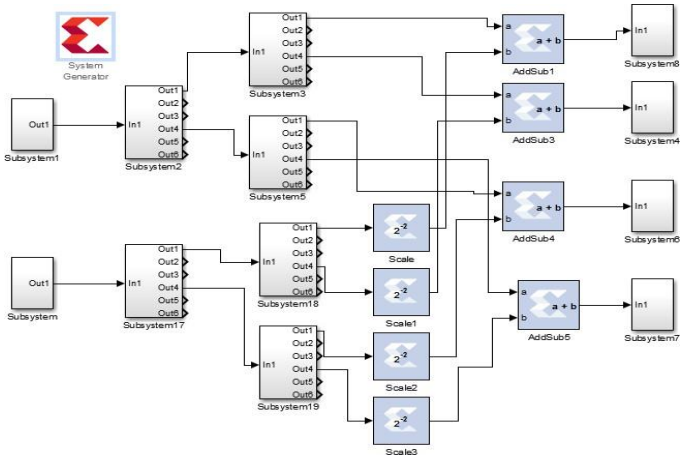


Fig 7: Embedding Model using XSG

After this model is simulated in MATLAB- Simulink environment, net list is generated and then configured for hardware co-simulation. Steps for watermark embedding are as follows-

Step 1: Both host and watermark images are read in MATLAB-Simulink environment and apply pre-processing on it. Here Pepper image is considered as host and Leena as the watermark image. Host is color image and watermark is gray scale image.

Step 2: Further pre-processing like red component selection (from host image), resizing, 2D to 1D conversion, frame conversion and buffering are done before passing the images to the XSG model as input.

Step 3: In XSG model, 2-D DWT of both host and watermark image is first calculated. Four components after DWT of host image are LLh, LHh, HLh and HHh. Four components after DWT of watermark image are LLw, LHw, HLw and HHw.
Step 4: Each component of watermark image (LLw, LHw, HLw and HHw) is scaled separately with a scaling factor before embedding with host image.
Step 5: Scaled components of watermark image are embedded with components of the host image according to the Eqn.1, Eqn.2, Eqn. 3 and Eqn.4 as given below:

$$LL = LLh + \alpha * LLw \quad (1)$$

$$LH = LHh + \alpha * LHw \quad (2)$$

$$HL = HLh + \alpha * HLw \quad (3)$$

$$HH = HHh + \alpha * HHw \quad (4)$$

Step 6: By applying inverse DWT to the new values of LL, LH, HL and HH the watermarked image is formed.

Fig. 8 shows Co-hardware simulation of embedding system. Co-hardware simulation of embedding system is generated and inputs-outputs are connected to the generated model.

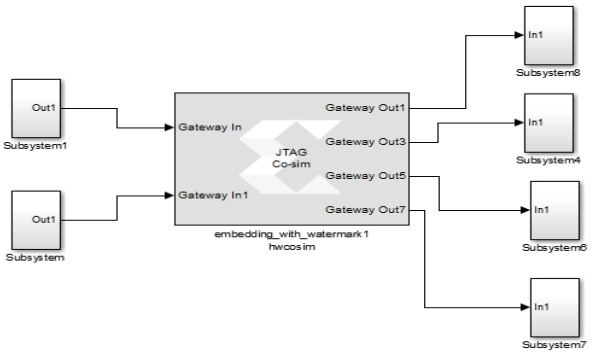


Fig 8: Co-hardware Simulation of Watermark Embedding System

WATERMARK EXTRACTION

Watermark extraction gives back the embedded watermark. Hence the embedding steps in the reverse direction need to apply to get back embedded watermark.

Step 1: Watermarked image is taken as an input of the extraction system. Apply pre-processing operations like- resizing, 2D to 1D conversion as at the time of embedding.

Step 2: DWT is applied on the watermarked image to get LL, LH, HL and HH sub-bands.
Step 3: As, $Watermarked\ image = host\ image + \alpha * watermark\ image$

Therefore,
$Watermark\ image = (Watermarked\ image - host\ image)/\ \alpha$

Where α=scaling factor. Hence, extraction is done using this method.

Step 4: The extracted watermark is compared with the original watermark for its similarity. As similar will be the extracted watermark close to the original watermark, as better will be the results.

Fig 9 shows block diagram of extraction system. Input watermarked image is called from MATLAB workspace.

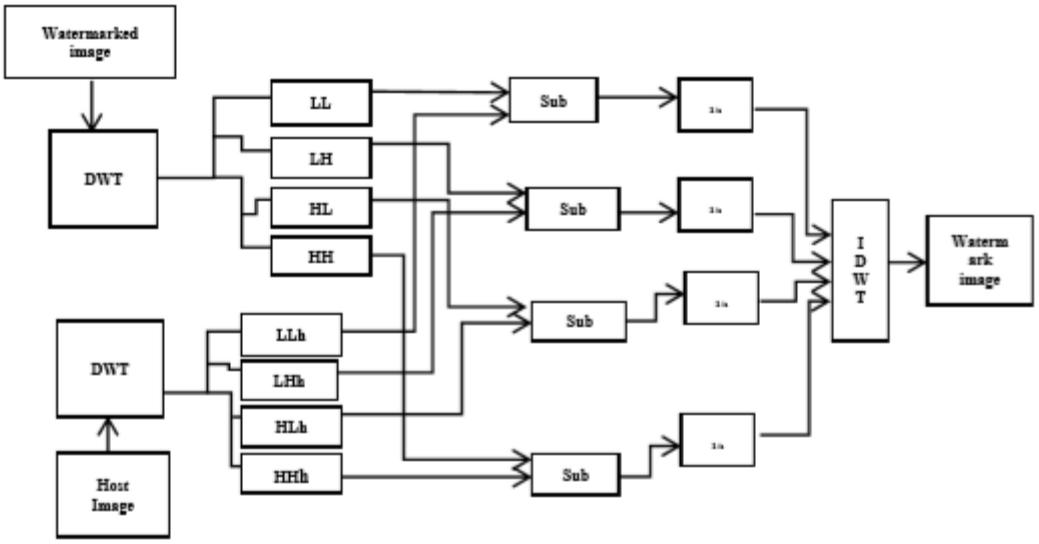


Figure 9: Block Diagram for Extracting Watermark

Host image is Peppers image that is connected as another input of extraction system. After applying DWT to both the watermark and host image, host image sub-bands and watermarked image sub-bands are obtained. Host sub-bands are subtracted from watermarked sub-bands and then inverse scaling is applied. At the output, four sub-bands of watermark are obtained and when IDWT is applied to these sub-bands, watermark image is obtained. Figure 10 shows the implementation of extraction system using XSG followed by 11 shows Co-hardware simulation model of extraction system.

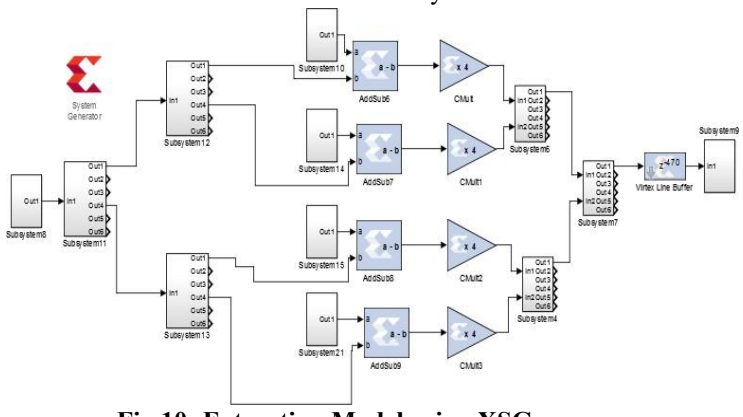


Fig 10: Extraction Model using XSG

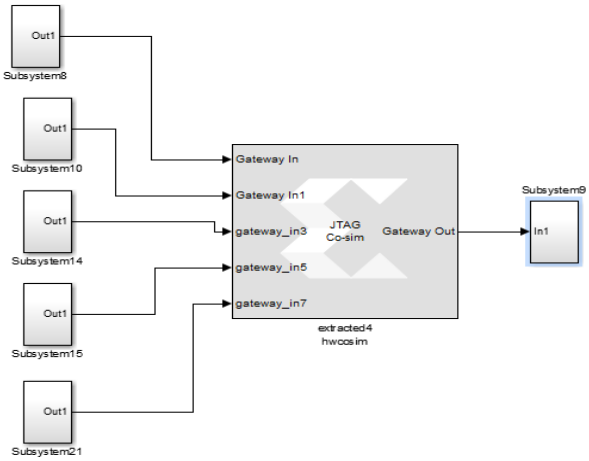


Fig 11: Co-hardware Simulation of Watermark Extraction System

RESULTS AND DISCUSSION

Models for watermark embedding and extraction as shown in Fig 7 and Fig 10 are implemented. In this section, the results of these models for only simulation and hardware co-simulations are presented, compared and discussed. Higher the similarity between watermarked image and host image, as better will be watermarking method. In this implementation, host image is the image of Peppers and watermark image is the image of Leena. This is shown in Fig 12.

Fig 12: Host Image (Peppers) and Watermark Image (Leena)

Hardware Co-Simulation results

Embedding and extraction system is simulated through XSG and then interfacing of XSG with ML605 kit provides results after co-simulation.

Watermark Embedding

Results for Simulation only through Simulink and Hardware Co-simulations on ML605 board with target device Virtex 6 FPGA are shown in Table 1 Here two images - peppers image as host and Leena image as watermark image is used as shown in Figure 12. Four different sub-bands of watermarked image LL, LH, HL and HH, as watermarking is done through DWT are shown in Table 1. Here, Scaling Factor is taken as 0.2. It is also observed

from Table 1 that results of Simulation through Simulink and Hardware Co- simulations are same.

Table 1: Co-hardware simulation results of four sub-bands at the time of embedding (α=0.2)

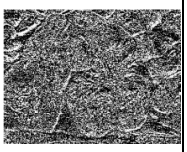
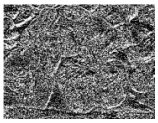
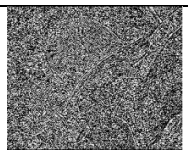
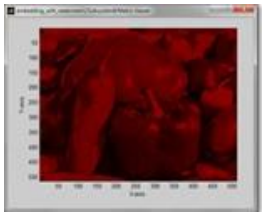
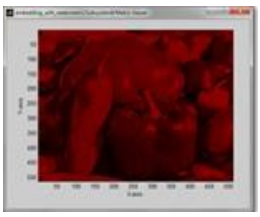
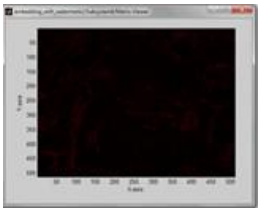
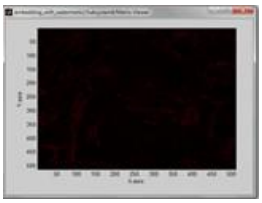
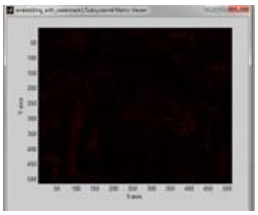
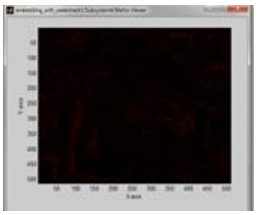
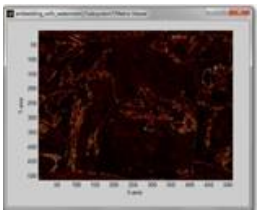
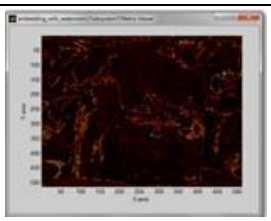
S. No.	Sub-bands	Host Sub-bands	Watermark Sub-bands	Watermarked Sub- bands
1.	LL			
2.	LH			
3.	HL			
4.	HH			

Table 2: Visual comparison between simulink results and hardware Co-simulation results

S. No.	**Name of Sub-band**	**Image through Simulink**	**Image through Hardware Co-Simulation**
1.	**LL**		

2.	**LH**		
3.	**HL**		
4.	**HH**		

Watermark Extraction

With the help of four sub-bands of watermarked image and host image, watermark image is again recovered in the extraction stage. All the four parts of extracted watermark are combined back to give an extracted watermark. Extracted watermark is same as embedded one. The watermark extracted and obtained through matrix viewer is shown in Figure 13.

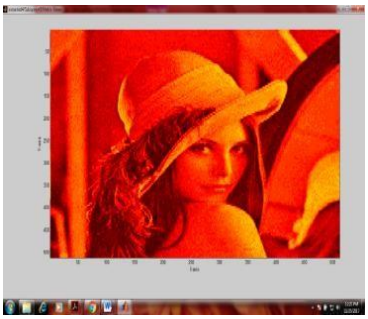


Figure 13: Extracted Watermark

Design Summary and Device Utility for Embedding Process

Figure 14 shows actual device utilization for embedding process. As RTL view represents the design using Macro blocks. Device utility show how much gates, registers, Look up Tables (LUTs), Flip flops, RAMs has been utilized for the purpose of designing.

Table 3 shows the comparison of this scheme with the scheme of Korrapati, Nelakudit, Mandhala and Kim (2015) in terms of device utilization. Authors implemented the system using virtex 6 LX240Tboard and XSG.

Table 3 shows the effectiveness of implemented scheme for slice register and flip-flop utilization as compared to the other scheme. Here utilization is 0% for register and flip flop whereas utilization is 1% [4]. However bonded utilization is 21% in the implemented system and it is 11%.

embedding_with_watermark1_cw Project Status (09/28/2018 - 14:27:52)			
Project File:	embedding_with_watermark1_cw.xise	**Parser Errors:**	No Errors
Module Name:	embedding_with_watermark1_cw	**Implementation State:**	Synthesized
Target Device:	xc6vlx240t-1ff1156	**• Errors:**	No Errors
Product Version:	ISE 14.6	**• Warnings:**	168 Warnings (168 new)
Design Goal:	Balanced	**• Routing Results:**	
Design Strategy:	Xilinx Default (unlocked)	**• Timing Constraints:**	
Environment:	System Settings	**• Final Timing Score:**	

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	1	301440	0%
Number of fully used LUT-FF pairs	0	1	0%
Number of bonded IOBs	129	600	21%
Number of BUFG/BUFGCTRLs	1	32	3%

Figure 14: Device utility for Embedding Process

Table 3: Comparison of device utilization

S. No.	Name of Components	Values obained from Virtex 6 ML605	Values obtained from Virtex 6 LX240
1.	Number of Slice registers	1	16
2.	Number of FF used	0	156
3.	Number of bonded IOBs	129	50

CONCLUSION

In this paper a method for hardware implementation of watermarking is shown. Results obtained after hardware simulation are satisfactory. The device utility is justified with values obtained after comparing different devices from Vertex 6 ML605 and Vertex 6LX240 based on number of slice registers, FF used and bonded IOB'S. When compared with existing methods, the implemented algo turns out to be beneficial and more better. Further, this method can be improved by introducingmore robustness in it.

References

1. Moreo, A. T., Lorente, P., Valles, F., Muro, J. and Andrés, C., "Experiences on developing computer vision hardware algorithms using Xilinx System Generator", *Microprocessors and Microsystems*, Vol. 9, Issue 8-9, pp. 411-419. 2005.

2. Roy, S., Li, X., Shoshan, Y., Fish, A. and Yadid-Pecht, O., "Hardware implementation of digital watermarking system for video authentication". *IEEE Transactions on circuits and systems for video technology,,* Vol. 23, Issue 2, pp. 289-301. 2013.
3. Karthigaikumar, P., Anumol and Baskaran, K., "FPGA implementation of high speed low area dwt based invisible image watermarking algorithm", *(Procedia Engineering) Proceedings of International Conference on Communication Technology and System Design*, Vol. 30, pp. 266-273. 2011.
4. Korrapati, R., Nelakudit, U., Mandhala, V. and Kim, T.- h., "FPGA Implementation of Watermarking Scheme Using XSG". *International Journal of Security and Its Applications,* Vol. 9, Issue 1, pp. 89-96. 2015.
5. Joshi, A., Mishra, V. and Patrikar, R., 2012 Real Time Implementation of Digital Watermarking Algorithm for Image and Video Application. IntechOpen. DOI: 10.5772/37182.
6. Bhaisare, A., Karode, A. and Suralkar, S., "Implementation of Real Time Digital Watermarking system for video authentication using FPGA", *IEEE International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC)* 2016, Jalgaon, India.
7. Hajjai, M., Ben Abdellali, A., Farhani, N., Gafsi , M. and Mtibaa, A., "Real time implementation of numerical watermarking system using xilinx system generator", *16th International Conference on Science and Techniques of Automatic Control and Computer Engineering (STA)*, Monastir, Tunisia, Dec. 21-23, 2015, pp. 404-409.
8. Saidani, T., Dhaha, D., Elhamzi, W., Atri, M. and Tourki, R., 2009, "Hardware co-simulation for video processing using xilinx system generator", *Proceedings of the World Congress on Engineering (WCE)*, London, U. K., July 1-3, 2009.
9. Ling, H., Wang, L. and Zou F. Jan. 2011, "Real-time video watermarking scheme resistant to geometric distortions". *Journal of Electronics Imaging*, Vol. 20, Issue 1, https://doi.org/10.1117/1.3565193.
10. Chandani Sharma; "Face Recognition and Detection using Image Processing" International Journal of Recent Research in Mathematics, Computer Science and Information Technology, ISSN 2350-1022, Vol. 1, Issue 1, pp. 23-27, April-June 2014.
11. Chandani Sharma, Anamika Jain; "Tuning Of Gains In Basis Weight Using Fuzzy Controllers" International Journal of Recent Research in Interdisciplinary Sciences, ISSN 2350-1049, Vol. 1, Issue 1, pp. 29-34, April-June 2014.
12. Chandani Sharma, Garima Gupta, "Novel Face Detection Technique" National Conference on Electronics and Communication Systems (NCECS-2013) on 5th and 6th April, 2013 at Indraprastha Engineering College, Ghaziabad (UP), India.
13. Chandani Sharma, Versha Varshney, "Morphological Shared Neural Nets For Face Detection" 2nd National Conference on Global Innovation in Technology & Management (GITM-2013) on 28th April, 2013 at Dev Bhoomi Group of Institutions, Saharanpur (UP), India.
14. Chandani Sharma, Jai Singh, "Human Face Visualization Using Gabor Filters and Neural Nets" and "Potential To Convert Non Conventional Energy Resources Into Electric Power In Remote Areas" 1st National Conference on Current Challenges & Opportunities in Technology & Management (CCOTM-2012) on 17th March, 2012 at Dev Bhoomi Group of Institutions, Saharanpur (UP), India.
15. Chandani Sharma, Madan Pal Singh, Pooja Sharma, "Audio Data Security Algorithm Using Watermarking Technique" and " Dragon Curve Fractal Antenna for UHF / VHF

Applications – YFFABB" 1st National Conference on Computer Electronics Technology (CET-2012) on 28th & 29th March, 2012 at Shobhit Institute of Engg. & Tech, Gangoh, Saharanpur (UP), India.