M Srinivas, M Praveena / International Journal of Engineering Research and Applications (IJERA)
ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 3, May-Jun 2012, pp.3183-3187

Design of a Modified Gabor Filter by Using Verilog HDL

M Srinivas*, M Praveena*
*(Department of ECE, VNR VJIET, Hyderabad, India
* (Department of ECE, VNR VJIET, Hyderabad, India

ABSTRACT

This paper presents the improvisation of Modified Gabor Filter design using Verilog HDL. This work details important enhancement made to the modified Digital Gabor filter to minimize the sizing problem and the coding style that synthesizable. The intention is to study, analyze, simplify and improvise the design synthesis efficiency and accuracy while maintaining the same functionality. The main characteristic of the proposed approach was to replace the parallel multiplication-accumulation unit (MAC) to a compact programming approach where the convolution matrix takes place. This significant change helps to reduce the sizing problem without jeopardizing the functionality of the modified Digital Gabor Filter. The result provides area efficiency architecture for the effective design.

Keywords –Digital filter, digital design, fingerprint, Modified Gabor filter, MAC, verilog HDL, Xilinx.

1. INTRODUCTION

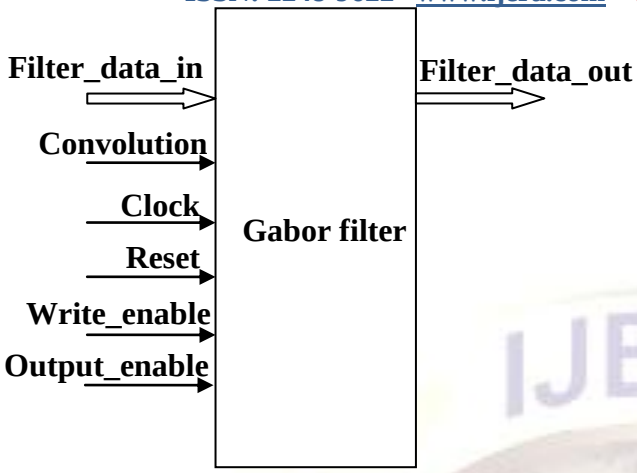
Gabor filter is a linear filter which is used for edge detection.Gabor filter is used in finger print enhancement which is considered as one of the complex process in fingerprint verification. Gabor filter has kernel coefficients and memory where convolution takes place. So Gabor filter plays an important role in fingerprintverification process. Designing a modified Gabor filter will help enhancing the quality of fingerprint image. In Fingerprint recognition process gabor filter captures information about local orientation and frequency of a finger print image. Here gabor filter is tuned to specific frequency and direction to obtain information about local frequency and orientation The convolution process takes place at the multiplication-accumulation unit (MAC) of the digital filter design. The parallel design of the MAC is to speed up the convolution process. Group of series data is transferred simultaneouslyhere and that is allowed with parallel design. Filter size was compromised by parallel MAC. Serial MAC allows group of series data bits to transfer one series of data at a time. The speed factor is compromised here but the area consumption is reduced. The speed in serial design can be improved if the filter is operated at high frequency.

This modified Gabor Filter was designed by transforming the design into Verilog using Xilinx 10.1. The target device is Spartan 3A family. The figure shown below is the block diagram of modified Gabor filter. Table 3.5 shows the resource utilization table and the proposed model has drastically reduced the slices etc.Basically there are 3 major parts in the filter: CLU, ALU and MEMORY. The 'convolution' signal indicates the operation of the filter. If
the signal is high that is 1 then the convolution process takes place between memory and kernel coefficients. If it is low then data is loaded into the memory. The data enters the filter pixel by pixel. The 'PIXEL_X' and 'PIXEL_Y' signal gives the address of memory location.

ALU is the main part of the filter that is doing the convolution process. There are 3 parts in ALU: ROM, DECODER and MAC. Here memory and CLU are taken as single block; there by reducing the area. The MAC unit is divided into 2 parts: multiplier and adder. The multiplier has got 9 parallel multipliers. So the multiplication will be done in the same time as to speed up the convolution process. The adder consists of 8 adders which are connected in sequence. The job of the adder is to sum up all the 9 multiplier outputs. The design is significantly large since 9 parallel multipliers and 8 adders are used.

2.BLOCK DIAGRAM

Figure 1 shows the block diagram of a modified gabor filter. There are 6 input pins and one output pin on the block diagram. Filter_data_in stands for 16bit input data. If convolution signal is low that is zero, filter_data_in is loaded into the memory.Kernel(gabor)coefficients are declared as parameter. If convolution signal is high that is 1 convolution process is performed between memory and kernel coefficients. If write_enable signal is 1 then the convolution output is loaded into the register .when output_enable signal is 1the data in the register is moved into filter_data_out pin.



F1gure 1: Block Diagram Of Gabor Filter

3. ALGORITHM FOR MODIFIED GABOR FILTER

STEP1: All the kernel (GABOR) coefficients are declared as parameters.p0, p2, p3……. p8

STEP2:When Convolution Signal is 1'b0 then Filter data input loaded into memory .i.e.

Memory [8] <=Filter_data_in [0],
Memory [7] <=Filter_data_in [1],
Memory [6] <=Filter_data_in [2],
.
.
.
Memory [0] <=Filter_data_in [8].

It takes 9 clock cycles to finish step 2.

STEP3: When convolution signal is 1'b1 then the convolution process takes place between memory and coefficients.

It takes 1 complete clock cycle to finish step 3.

STEP4: When Write signal is 1'b1 then the convolution output is loaded into the register.

It takes 1 complete clock cycle to finish step 4.

STEP 5: When output_enable is 1'b1 then the data in the register is sent to Filter_data_out. When the convolution signal is 1'b0 again the process goes to STEP 2.

It takes 1 complete clock cycle to finish step 5.

4. FLOW CHART

Figure 2 shows the flow chart of verilog design description for compact verilog programming approachfor the modified gabor filter .

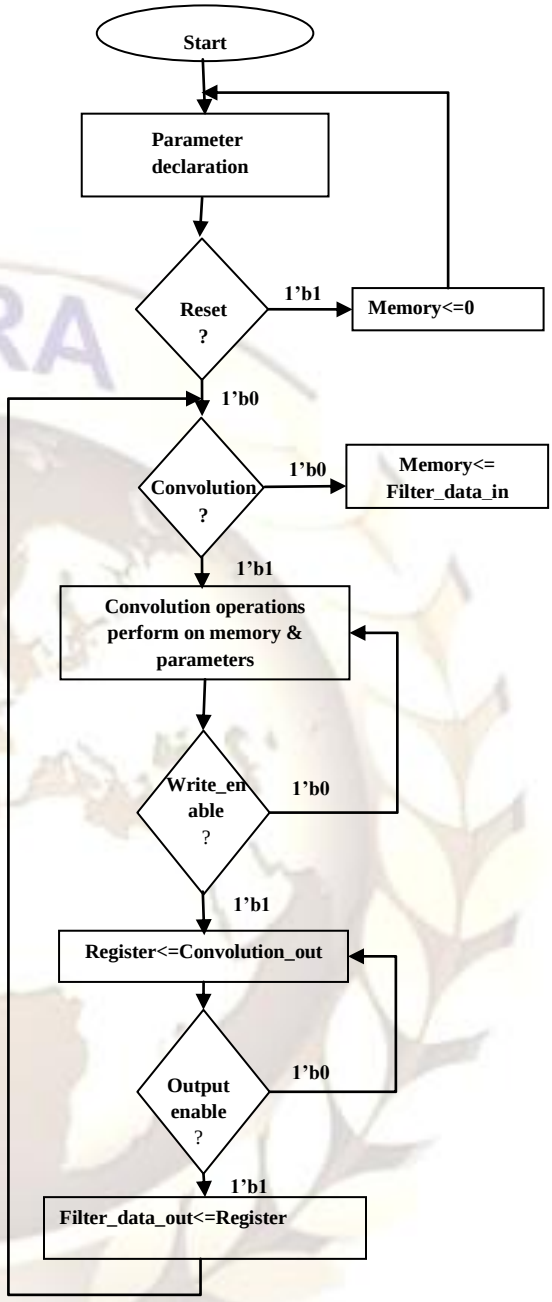


Figure 2: Flow chart for Gabor Filter implementation

5. RESULTS AND DISCUSSIONS

5.1 OUTPUT WAVEFORM

Figure 3 shows the output of modified gabor filter. When convolution signal is 1the convolution output and filter_in is shown in the rounded circle on the output wave window .

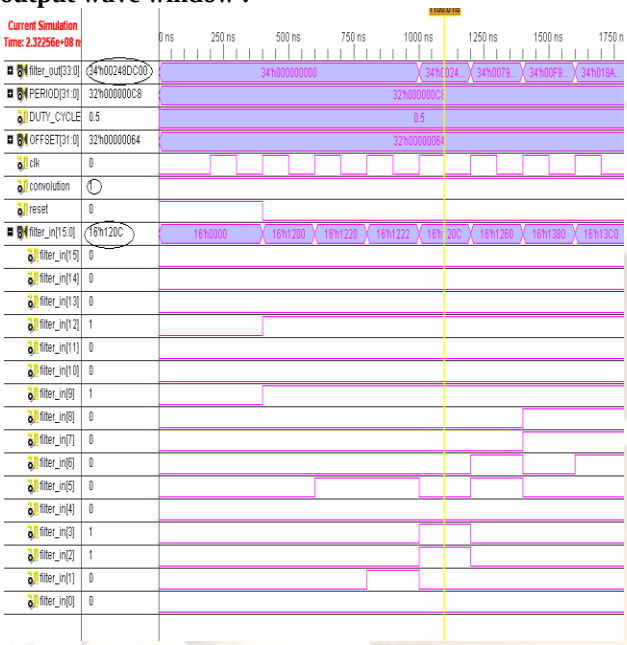


Figure 3: Output for Gabor Filter Top module

5.2 RTL SCHEMATIC

Figure 4 shows the RTL schematic for the modified gabor filter

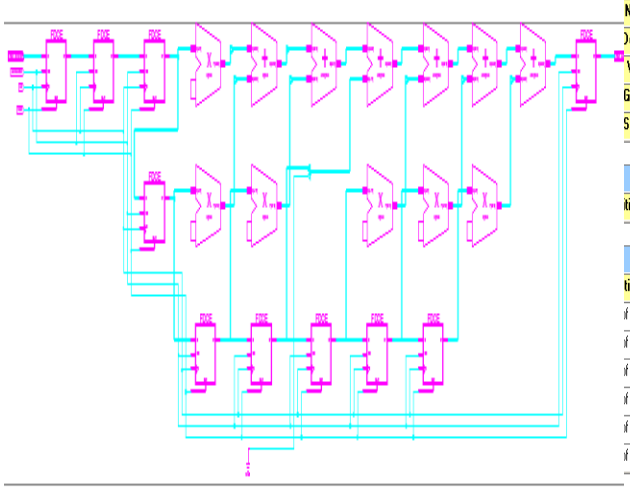


Figure4: RTL Schematic for Gabor Filter Top module

5.3 TECHNOLOGY SCHEMATIC

Figure 4 shows the Technology schematic for the modified gabor filter.

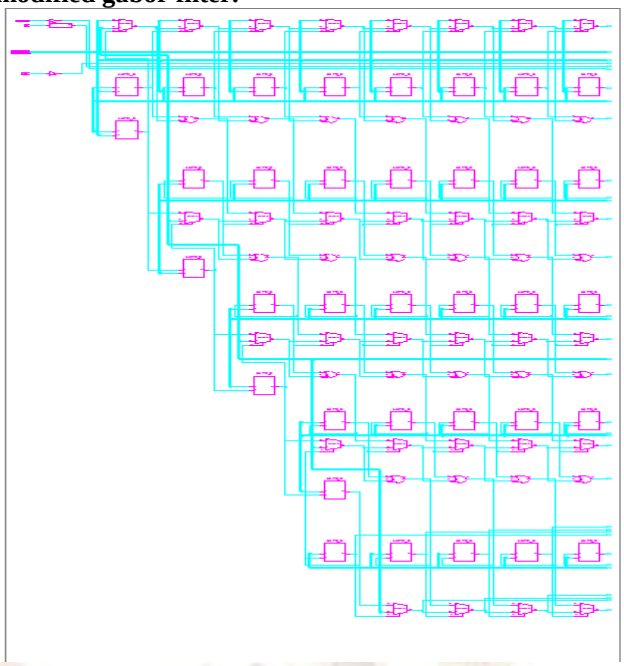


Figure 5: RTL Technology Schematic for Gabor Filter Top module

5.4 SYNTHESIS REPORT

Figure 6 shows the synthesis report for the modified gabor filter

fpgadesign Project Status			
File:	fpgadesign.ise	Current State:	Synthesized
Name:	gabor_filter	• Errors:	No Errors
Device:	xc3s400-5pq208	• Warnings:	12 Warnings
Version:	ISE 10.1 - Foundation Simulator	• Routing Results:	
Goal:	Balanced	• Timing Constraints:	
Strategy:	Xilinx Default (unlocked)	• Final Timing Score:	

fpgadesign Partition Summary
tion information was found.

Device Utilization Summary (estimated values)			
tilization	Used	Available	Utilization
f Slices	177	3584	4%
f Slice Flip Flops	178	7168	2%
f 4 input LUTs	188	7168	2%
f bonded IOBs	53	141	37%
f MULT18X18s	6	16	37%
f GCLKs	1	8	12%

Detailed Reports					
Name	Status	Generated	Errors	Warnings	Infos
Report	Current	Wed May 23 12:40:46 2012	0	12 Warnings	6 Infos

Figure 6: Synthesis Report for Gabor Filter Top module

5.5 RESOURCE UTILIZATION TABLE

Table 1 shows the resource utilization comparisons between reference1,reference2 and proposed model.

S. No	Device parameter	Reference 1	Reference 2	Proposed model
1	Number of Slices	818%	84%	4%
2	Number of Slice Flip Flops	615%	50%	2%
3	Number of 4 input LUTs	552%	71%	2%
4	Number of bonded IOBs	29%	42%	37%
5	Number of MULT18X18s	1200%	70%	37%
6	Number of GCLKs	8%	37%	12%
7	Number of warnings	80%	26%	12%
8	Number of infos	34%	10%	6%

Table 1:Resource Utilization Table

5.6 COMPARISON GRAPH (REFERENCE 2 VS PROPOSED MODEL)

Figure 7 shows the resource comparison graph between reference2 and proposed model. The red line shows the reference2 and blue line shows the proposed model.

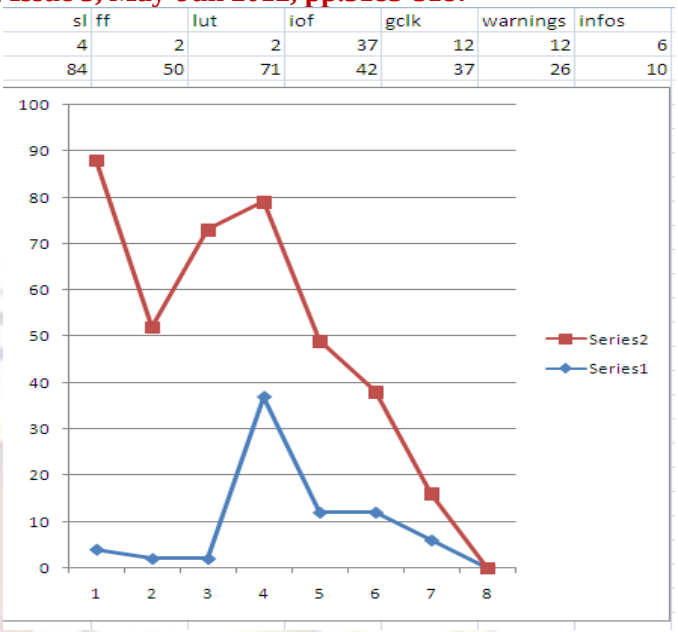


Figure 7: Comparison graph between reference 2 vs proposed model

5.7 BAR CHART

Figure 8 shows the resource comparison bar chart between reference2 and proposed model. The red bar shows the reference2 and blue bar shows the proposed model.

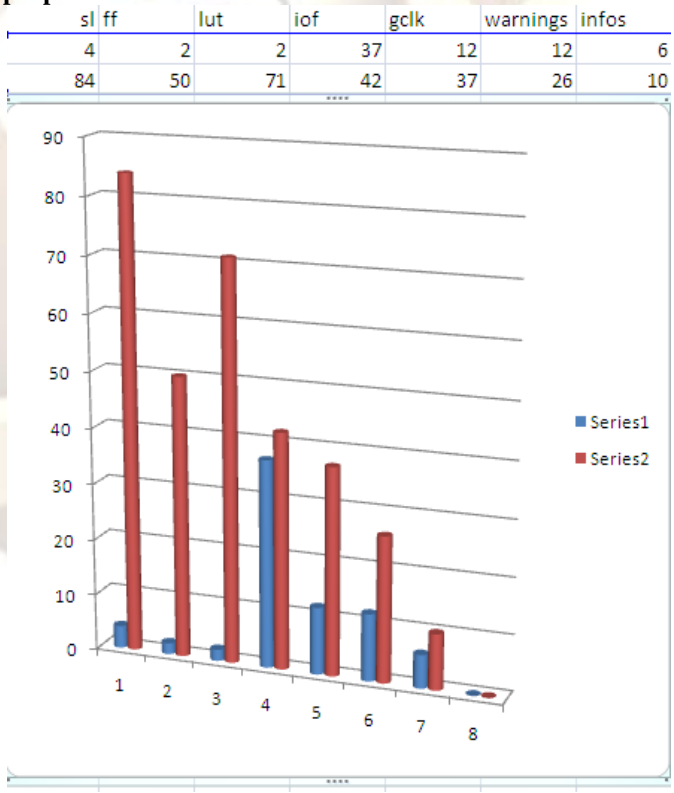


Figure 8: Comparison Bar Chart between reference 2 vs proposed model

6. CONCLUSION

The design enhancement proposed for Modified Gabor Filter has successfully reached. The area of the design has been significantly reduced while the function of the filter is perfectly maintained. The numbers of slices used from previous design (Reference 2) reduce from 1625 slices to 177 slices. This significant change is due to the compact verilog HDL coding used in the Modified Gabor Filter. The enhancement made in the multiplication-accumulation unit has been proven effectively reliable and functional. By adjusting the memory and the controller unit, the functionality of a complete and correct digital Gabor Filter is obtained. By minimizing the area, the speed of the design is relatively slower. It took 12 complete cycles to finish the convolution.

7. ACKNOWLEDGEMENT

I am extremely pleased to present myself this dissertation after getting innumerable movements and hard time with a never-ending coordination. I sincerely acknowledge offering my words of thanks to our beloved and respected Mrs. M Praveena Assistant Professor, Mr.Dr.P.Srihari Head of the deportment and Dr.Y.Padma Sai VLSI coordinator who is a constant source of inspiration and encouragement. I sincerely acknowledge offering my words of thanks to my beloved and respected parents for their constant support and encouragement. Sincere thanks to my friend Bekkam Satheesh who helped me during the paper preparation.

8. REFERENCES

[1] Razak, A.H.A. Taharim, R.H. "Implementing GaborFilter for Fingerprint Recognition using verilog HDL,"IEEE explorer , March 2009.

[2] dros,M.F.MMohamed,S.ARazak,A.H.AZoolfakar,A.S Al-Junid,S.A.M,improvisation of gabor filter design using veriloghdl,ieee explorer,2010

[3] P. H. W. L. Ocean Y. H. Cheung, Eric K.C. Tsang,BertamE.SHi, "Implementing Of Gabor-type Filters on Field Programmable Gate Arrays," 2005.

[4] A. P. Arrigo Benedetti, NelloScarabottolo, "Image Convolution o FPGAs:the implementation of a multi- FPGA structure," 1998.

[5] K. S. Vasily G. Moshnyaga, Keikichi Tamaru, "A Memory based architecture for real-time convolution with variable kernels," 1998.

[6] Clifford E. Cummings, "Verilog-2001 Behavioral and Synthesis Enhancement," Dec 2001.

[7] HimanshuBhatnagar, "Advanced ASIC Chip Syhthesis," Kluwer Academic Publisher, 1999, pp 202-203.

[8] Don Mills,Clifford E. Cummings, "RTL Coding Styles That Yield Simulation and Synthesis Mismatches," Oct 2000.

[9] Rajesh Bawankule" Verilog Code Writing Guidelines," Sept 2002.

[10] Michael D. Ciletti, "Modeling, Synthesis, and Rapid Prototyping With the Verilog HDL," Prentice Hall, Dec 1999.

M Srinivas received the B.Tech degree in electronics and communication engineering from CMRcollege of Engineering & Technology, affiliated to Jawaharlal Nehru Technological University Hyderabad, AP, India, in 2010, he is pursuing the M.Tech in VLSI System Design at VNR VignanaJyothi Institute of Engineering & Technology, Bachpally, Hyderabad, India. His research interests include Digital Design (FPGA).

Ms.M.Praveena has obtained her M.Tech. in VLSI System Design from VNR VignanaJyothi Institute of Engineering & Technology, JNT University, Hyderabad, India; Awarded in 2007. B.Tech. in ECE from Gudlavalleru Engineering College, Gudlavalleru, JNT University, India, Awarded in 2003.She is working as Asst Professor in ECE Dept at VNR VignanaJyothi Institute of Engineering & Technology, Bachpally, and Hyderabad, India. Her research interests in Switching Activity in Field Programmable Gate Array,Vlsi implementation of image processing applications