

## Realization of Linear Back-Projection Algorithm for Capacitance Tomography Using FPGA

B. Almashary<sup>1</sup>, S. M. Qasim<sup>1</sup>, S. Alshebeili<sup>1</sup>, W. A. Al-Masry<sup>2</sup>

<sup>1</sup> Electrical Engineering Department, College of Engineering, King Saud University, P.O. Box 800, Riyadh 11421, Saudi Arabia

<sup>2</sup> Chemical Engineering Department, College of Engineering, King Saud University, P.O. Box 800, Riyadh 11421, Saudi Arabia, [walmasry@ksu.edu.sa](mailto:walmasry@ksu.edu.sa)

### ABSTRACT

*This paper presents the evaluation of Field Programmable Gate Arrays (FPGA) architecture for realization of the Linear Back Projection (LBP) algorithm for image reconstruction in electrical capacitance tomography (ECT). The top-down approach has been adopted for the design of the hardware of LBP algorithm. We used Virtex FPGA from Xilinx and achieved an improvement with factor of 18 in speed compared to other designs which have used single digital-signal-processor (DSP), and a factor of 4.6 improvement compared to four parallel DSPs. Therefore, adopting FPGA as a hardware platform to implement LBP algorithm achieves superior performance in terms of speed and design compactness compared to DSP implementation. The details of the architecture and the methodology to code LBP algorithm using (Very High Speed Hardware Description Language (VHDL) are presented and the power consumption is estimated.*

**Keywords:** capacitance tomography, FPGA, linear back-projection, VHDL.

### 1 INTRODUCTION

Electrical Capacitance Tomography (ECT) is used to obtain information about spatial distribution of a mixture of dielectric material inside a vessel, by measuring the electrical capacitances between sets of electrodes placed around its periphery and converting them into an image showing the distribution of permittivity and hence the material distribution over a cross section, using a suitable algorithm.

Image reconstruction algorithms for ECT can be categorized into two groups, non-iterative algorithms, and iterative algorithms. Among the non-iterative algorithms, the (Linear Back-Projection (LBP)) algorithm is the simplest and fastest. Although it has some limitations in terms of accuracy and spatial resolution, it is well suited for fast dynamic processes like multiphase flow (Garacia 2003), and widely used for on-line image reconstruction (Yang 2003). Running on a modern PC-type computer, reconstruction rates of a few tens of frames per second can be easily achieved. Nevertheless, for some applications (like flow imaging) hundreds of frames per second may be required.

Recently, a high-performance parallel digital-signal-processor architecture has been proposed for image reconstruction using LBP algorithm (Garacia 2003) on 1, 2, 3, and 4 processors. It was found that the execution speed of the LBP algorithm has increased in almost a linear way with the number of processors. For one DSP, 7.365 ms was needed to complete one image reconstruction; this is equivalent to 135 frames per second (Garacia 2003). It was found that more improvement can be achieved using more processors. Of course, the more processors you use, the more hardware you need. Therefore the need for more improvement in the speed with less hardware presents a challenging researching point.

In this paper, we propose using another type of hardware for image reconstruction using LBP algorithm. FPGAs architecture is employed in our proposed system to achieve faster and more compact realization of the system. The FPGAs have the advantage of being re-configurable directly by the software. This adds more advantage in terms of flexibility in designing the hardware and updating its structure. The main difference between DSP and FPGAs comes from the simple fact that DSP is a pre-designed hardware (processor) optimized to implement specific mathematical operations. However, FPGA is a platform that can be used to design a specific hardware optimized to implement a specific algorithm. The direct mapping between the hardware and the algorithm (as in FPGAs) is the key difference, and it has a significant impact on the performance of the algorithm.

FPGAs from Xilinx (Xilinx 1996), such as Virtex series provide two-dimensional array of logic blocks where each block contains several flip-flops and look-up-tables capable of implementing many logic functions. In particular, more complex functions such as matrix multiplication can be mapped very successfully to FPGAs. In essence, the LBP algorithm is matrix-multiplication and as such requires several multiplications to be performed. The exact number of multipliers is dependent on the number of electrodes and frames. In DSP, the overall performance will be limited by the number of multiplications that could be done in parallel. In practice, a DSP will require several clock cycles to perform all the necessary multiplications and additions. The FPGA on the other hand can implement as many multipliers as are necessary at the full input data rate.

Our work introduces the details of the architecture used and the methodology to code LBP algorithm using VHDL. Different software tools are used for simulation and implementation on Virtex FPGA from Xilinx. The performance is evaluated in terms of speed and compared with DSP-based performance.

## 2 LBP ALGORITHM AND ECT

The basic idea of ECT is to use N-electrode sensor to measure the capacitance and get  $N(N-1)/2$  independent capacitance values. From this data and using a suitable reconstruction algorithm, an image is obtained, showing the mixture components within the cross-section defined by the electrode ring (Garacia 2003). Usually, a set of electrodes eight or 12 is used, and the sensing electronics provides excitation signals and convert the capacitances into voltage signal, which are conditioned and then digitized for data acquisition. The computer controls the system hardware and implements the image reconstruction algorithm to show the permittivity distribution. The LBP has been implemented to reconstruct images for ECT using 12-electrode sensor and a transputer-based multiprocessor system (Xie 1992).

The relationship between capacitance and permittivity distribution can be approximated and written in a linear normalized form as:

$$B = S \cdot X \quad (1)$$

Where B is the normalized capacitance matrix, S is the transducer sensitivity matrix (normalized capacitance with respect to normalized permittivity), and X is the pixel gray level matrix (the normalized permittivity).

The task of image reconstruction for ECT is to determine the permittivity distribution from the measured capacitance. In the discrete form, it is to find the unknown X from the known B, while S is treated as a constant matrix for simplicity (Yang 2003).

The basic idea of LBP algorithm is to calculate the gray level using the transpose of the sensitivity matrix instead of its inverse. The order of the above matrices depends on the number of electrodes used and the number of pixels. In our case, our ECT measurements are based on 8 electrodes and 32X32 pixels. This makes B a (28x1) vector, X a (1024x1) vector, and S a (28x1024) matrix. Therefore, to solve for X, the transpose of S (we will call it A), needs to be multiplied by B (i.e.  $A \cdot B = X$ ).

## 3 FPGA DESIGN FLOW

This section discusses the FPGA design flow required to realize the initial concept in FPGA. Design entry is done using VHDL, a textual language for describing digital hardware. Although VHDL is mainly used for simulation, it can also be used for synthesis. Synthesis is the process of taking an input form like VHDL and generating a netlist. A design must be verified for correctness before it is synthesized. A design is ready for synthesis once it simulates correctly. In order to find the actual delay of the circuit, we have to do the timing simulation (Back-annotation). Once the design is placed and routed, a binary file can be generated that contains all the vital information needed for programming the FPGA with the design. These steps are depicted in Figure 1. Different design tools have been used at each stage. FPGA Advantage from Mentor Graphics has been used for the design entry. Modelsim simulator from Model Technology has been used to simulate the code. Leonardo Spectrum

synthesizer from Exemplar Logic has been used for synthesis. ISE Alliance from Xilinx has been used placing and routing for of the design to the FPGA.

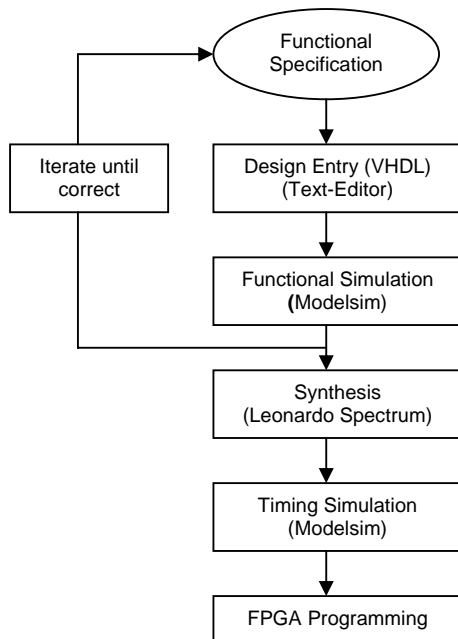


Figure 1: FPGA Design Flow

## 4 METHODOLOGY AND IMPLEMENTATION

This section describes the methodology used in implementing Linear Back-Projection. The LBP Algorithm is based on matrix multiplication and as such requires several multiplications to be performed. The exact number of multipliers is dependent on the number of electrodes and frames. In DSP, the overall performance is limited by the number of multiplications that could be done in parallel. In practice, a DSP will require several clock cycles to perform all the necessary multiplications and additions. FPGA on the other hand can implement as many multipliers at the full input data rate.

The design involves the computation of  $A \cdot B = X$ , where  $A[1024:28]$  is a matrix,  $B[28:1]$  and  $X[1024:1]$  are vectors, and  $X$  is what we are trying to compute. Load the inputs  $A$  ( $a_1$  to  $a_{28}$ ) and  $B$  ( $b_1$  to  $b_{28}$ ) into registers  $A$  and  $B$  respectively. The column of matrix  $B$  is fixed whereas the row of matrix  $A$  changes 1024 times. The inputs to the circuit are fed in bit serial fashion. If reset pin is high, all the contents of  $A$  and  $B$  are reset to zero and we get a zero at the serial output. At the rising edge of the clock, each of these elements is multiplied, accumulated and then added. This process continues till we reach the last element of both  $A$  ( $a_{28}$ ) and  $B$  ( $b_{28}$ ). After a delay of 400 ns, the first element of vector  $X$  is available at the serial output and this output is stored in on-chip Memory at location 1. This operation is repeated 1024 times to cover all the rows of matrix  $A$ . Hence, after a delay of 0.41 ms we get the vector  $X$  with 1024 elements filled in the memory at locations 1 to 1024 respectively. The synthesis tool infers a dual port RAM for the memory. Figure 2 summarizes the basic principle that has been adopted to implement the algorithm. The complete code for this algorithm is written using VHDL.

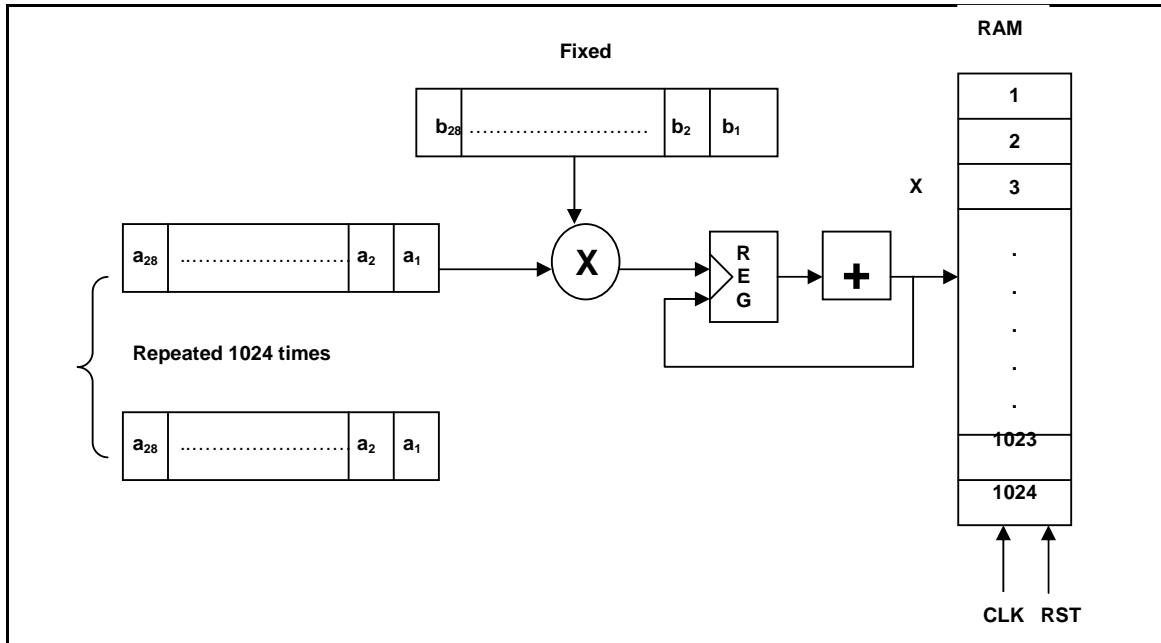


Figure 2: Basic block diagram

## 5 RESULTS AND ANALYSIS

The LBP algorithm was described using VHDL code, compiled and simulated using the Modelsim Simulator. A sample of functional simulation results is shown in Figure 3. The design was further synthesized into Xilinx Virtex FPGA ((XV600 FG676) using the Leonardo Spectrum synthesis tool optimized for delay. The actual delay of the circuit is obtained by doing the back-annotation using the Xilinx Alliance place and route tool. The results of the synthesis process are summarized in Table 1. We have also used the Xilinx XPower tool to estimate the power consumption of our design, and Table 1 includes this result as well.

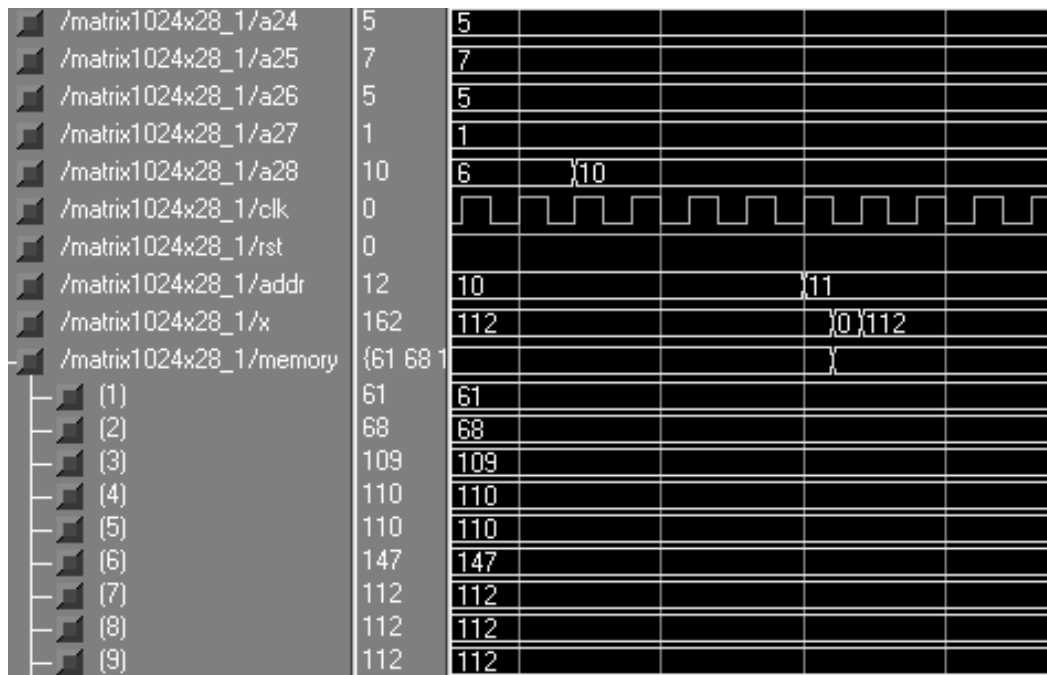


Figure 3: Functional simulation results

Number of CLB Slices	:	1,345 out of 6,912	19%
Total Number 4 input LUTs	:	2,585 out of 13,824	18%
Number used as LUTs	:	731	
Number used for Dual Port RAMs	:	1,792 (Two LUTs used per Dual Port RAM)	
Number of bonded IOBs	:	387 out of 444	87%
Number of GCLKs	:	1 out of 4	25%
Number of GCLKIOBs	:	1 out of 4	25%
Total equivalent gate count for design: 132,241			
Additional JTAG gate count for IOBs : 18,624			
Peak Memory Usage: 95 MB			
Maximum Frequency (Before Place and Route) : 75.878 MHz			
Maximum Frequency (After Place and Route) : 41.906 MHz			
Total estimated power consumption:		152 mW	

**Table 1. Synthesis results of Xilinx XV600 FG676 (Speed grade:-4) Virtex FPGA**

From the timing simulation results, it is found that FPGA based system takes 0.41 ms to reconstruct one image; this is equivalent to an image rate of about 2439 frames per second. Table 2 summarizes the speed-up achieved for LBP Algorithm with FPGA as compared to DSP system, where the speed-up ( $S_p$ ) has been calculated using the formula given below

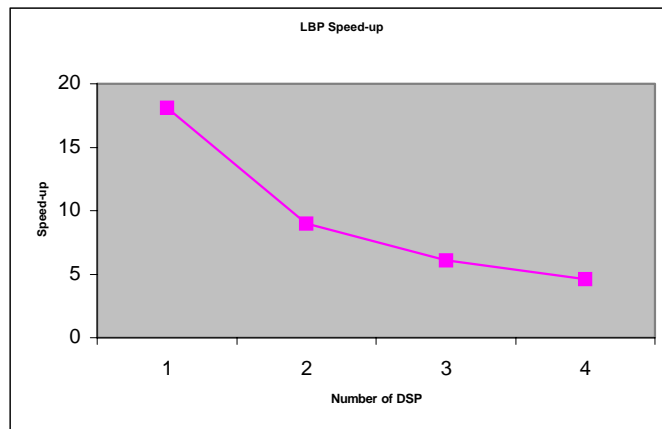
$$S_p = T[\text{FPGA}] / T[\text{DSP}] \quad (2)$$

where, T [FPGA] is the total time taken by FPGA based system to complete a single image reconstruction and T [DSP] is the total execution time taken by DSP based system to reconstruct a single image.

No. of DSP	Image frame/sec T[DSP]	Speed-up ( $S_p$ )
1	135	18.06
2	270	9.03
3	400	6.1
4	530	4.6

**Table 2. Speed-up comparison of FPGA (vs) DSP (For T [FPGA] =2439 frames/sec)**

Figure 4 shows the plot of speed-up versus the number of DSP Processors used for the implementation of the LBP Algorithm. It shows that FPGA based system running LBP Algorithm is 4.6 times faster as compared to four DSP based system (Garacia, 2003) and 18 times faster as compared to single DSP based system (Garacia, 2003).



**Figure 4: LBP speed-up FPGA vs. DSP**

## 6 CONCLUSIONS

This paper presents the evaluation of FPGA architecture for realization of the LBP algorithm used in electrical capacitance tomography. The top-down approach has been adopted for the design of the LBP algorithm. FPGA provides enormous improvement in the image-reconstruction speed as compared to DSP based systems. The FPGA with LBP Algorithm is found to be running at a clock frequency of 41.906 MHz and consumes 152 mW of power. We used Virtex FPGA from Xilinx and obtained an improvement of factor of 18 in speed, compared to other designs which have used single DSP, and a factor of 4.6 improvement compared to four parallel DSPs. This confirms that adopting FPGA as a hardware platform to implement LBP algorithm is better in terms of speed and design compactness.

## 7 REFERENCES

GARACIA-NOCETTI, D., GAMIO, J., AGUILAR, L., (2003), *Parallel Realization of the Linear Back-projection Algorithm for Capacitance Tomography using TMS320C6701 DSP*, 3<sup>rd</sup> World Congress on Industrial Tomography, Banff, Canada, pp. 648-653.

YANG, W., PENG L., (2003), Image Reconstruction Algorithms for Electrical Capacitance Tomography, *Measurement Science Technology*, vol. 14, pp. R1-R13.

XIE C., *et al.*, (1992), *Electrical Capacitance Tomography for Flow Imaging: System Model for Development of Image Reconstruction Algorithms and Design of Primary Sensors*, IEE Proceedings G vol. 139 (1), pp. 89-98.

XILINX, (1996), *The programmable Logic Data Book*, Xilinx Inc.