Implementation of Real-Time System for Medical Image Processing using Verilog Hardware Description Language

IULIANA CHIUCHIȘAN, MARIUS CRISTIAN CERLINCĂ
Electrical Engineering and Computer Sciences Department
“Ștefan cel Mare” University of Suceava
13 Universității Street, 720229 Suceava
ROMANIA
iulia@eed.usv.ro, mariusc@eed.usv.ro

Abstract: - Image processing algorithms implemented in hardware have emerged as the most viable solution for improving the performance of image processing systems and the introduction of reconfigurable devices and hardware description languages has further accelerated the design of image processing in hardware. To implement the image processing algorithms and to process the large amount of data captured from sources such as medical instruments, intelligent high speed real-time systems have become imperative. This paper is providing an innovative hardware implementation of a real-time configurable system for medical image processing implementation and simulation followed by immediate synthesis using FPGA. The authors present a pipelined architecture of a real-time system for DIP and focuses on medical images enhancement using this system. The hardware modeling is done with the hardware description language Verilog.

Key-Words: - digital image processing (DIP), hardware description language (HDL), Verilog, reconfigurable architecture (FPGA), real-time configurable system for DIP

1 Introduction
Digital image processing is an ever expanding and dynamic area with variety of applications in different fields, reaching out into everyday life such as medicine, space exploration, surveillance, authentication, automated industry inspection and many more areas. Applications such as these involve different processes like image enhancement. Generally even specialized image processing programs running on PCs cannot adequately process large amounts of high-resolution streaming data, since PC processors are made to be for general use and hence are unable to efficiently implement many current sophisticated DIP algorithms [1]. Also, to process the large amount of 3D data from medical instruments, intelligent high speed real-time systems have become imperative, which may process data before passing it to the human analyst [1]. Many image processing applications require that several operations be performed on each pixel in the image resulting in a even large number of operations per second.

One alternative is to use a field programmable gate array (FPGA). Continual growth in the size and functionality of FPGAs over recent years has resulted in an increasing interest in their use as implementation platforms for image processing applications, particularly real time video processing. Application specific hardware implementation offers a greater speed than a software implementation and with advances in the VLSI technology hardware implementation has become an attractive alternative. Implementing complex computation tasks on hardware and by exploiting parallelism and pipelining in algorithms yield significant reduction in execution times [2]. Implementing image processing algorithms on reconfigurable hardware minimizes the time-to-market cost, enables rapid prototyping of complex algorithms and simplifies debugging and verification [2].

Sparsh Mittal, Saket Gupta and S. Dasgupta present in their paper [1] the advantages and limitation of FPGAs. The salient features of FPGAs that make them superior in speed, over conventional general-purpose hardware is their greater I/O bandwidth to local memory, pipelining, parallelism and availability of optimizing compiler [1]. Therefore, FPGAs are an ideal choice for implementation of real-time image processing algorithms. This reconfigurable devices have traditionally been configured using a Hardware Design Language and the two principal languages used are Verilog HDL and Very High Speed Integrated Circuits (VHDL), which allows designers to design at various levels of abstraction [3].
The Hardware Description Languages larger availability allows the designers to not only logically describe circuit functionality but to simulate and evaluate the processing performances using appropriate development and test environments. While the simulation is generating the logical results a natural step consist in extending the use of the hardware simulators into the field of digital signal processing [3].

2 Hardware Implementation of Real-Time System for Image Processing

Given the importance of real-time image processing in medical instruments and the significance of their implementations on hardware to achieve better performance, this paper discusses medical image enhancement techniques with their implementation and results using a hardware description language, Verilog.

Verilog is a hardware description language (HDL) used to model electronic systems. The language supports the design, testing and implementation of analog, digital, and mixed-signal circuits at various levels of abstraction. The language differs from a conventional programming language in that the execution of statements is not strictly linear. A subset of statements in the language is synthesizable. If the modules in a design contain only synthesizable statements, software can be used to transform or synthesize the design into a netlist that describes the basic components and connections to be implemented in hardware. The netlist may then be transformed into a form describing the standard cells of an integrated circuit (e.g. an ASIC) or a bitstream for a programmable logic device (e.g. a FPGA) [2].

The real time image processing requirements demand a system having characteristics like high performance, flexibility, easy upgradability, low development cost. This work proposes a novel hardware architecture of a real-time configurable system for DIP using Verilog HDL and reconfigurable architecture. The implementation of the proposed system was described at Register Transfer Level (RTL).

The images are processed by switching and processing them by a series of filters to improve image. The configurable system described and tested has constituted from four filters for improving digital images and using a single control input the system achieves to inter-change them on-the-fly (is not necessary to switch off the reconfigurable equipment). These filters are connected in a pipeline structure, with the possibility of amending the order of filters and the parameters of each filter features (Fig.1).

![Fig.1 Real-time configurable filter system DIP structure (RTL Schematic)](image)

The principal objective of this basic algorithms is to process an image so that the result is more suitable than the original image for a specific application [3].

The scope of the proposed system is for medical equipment because the speed of diagnosis of disease is sometimes vital, but can be used in any other area in which the speed of image processing in real time is vital. For this purpose, configurable filter system proposed for image processing in real time provides a topical solution, being complementary, alongside the other classical processes for real-time applications.

3 Experimental Results

The real-time configurable system for image processing discussed above was modeled in Verilog HDL using the Xilinx ISE Design Suite environment (Fig. 2). The design was implemented on Xilinx Virtex-6 FPGA ML605 based hardware.

![Fig.2 Simulation results from ISIM Simulator](image)
Another important point is that during Verilog implementation, the logic and memory resource usage was maximized to achieve better timing performance. The frequency and processing time of operation of the hardware for the real-time configurable system on a 400x400 size grayscale MRI image (magnetic resonance imaging) is shown in Table 1. The proposed architecture needs a number of four clocks to process one pixel.

Table 1: Experimental results with frequency and processing time

<table>
<thead>
<tr>
<th>Xilinx Virtex-6 FPGA ML605</th>
<th>Frequency [MHz]</th>
<th>Time [ns]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>219.443</td>
<td>18.2280</td>
</tr>
</tbody>
</table>

Also, an evaluation of logic resources required achieved for hardware implementation of the real-time configurable system for DIP, using a platform Xilinx Virtex-6 FPGA ML605, is performed and tabulated in Table 2.

Table 2: Experimental results with area optimization

<table>
<thead>
<tr>
<th>Device Utilization Summary</th>
<th>Used</th>
<th>Available</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Slice Registers</td>
<td>32</td>
<td>301,440</td>
<td>1%</td>
</tr>
<tr>
<td>Number of Slice LUTs</td>
<td>116</td>
<td>150,720</td>
<td>1%</td>
</tr>
<tr>
<td>Number of fully used LUT-FF pairs</td>
<td>30</td>
<td>117</td>
<td>25%</td>
</tr>
<tr>
<td>Number of bonded IOBs</td>
<td>92</td>
<td>600</td>
<td>15%</td>
</tr>
<tr>
<td>Number of BUFG/BUFGCTRLs</td>
<td>1</td>
<td>32</td>
<td>3%</td>
</tr>
</tbody>
</table>

The experimental results are shown in this section. We use a 400x400 size grayscale MRI image (MRI.bmp) as an input image, shown in each left panel.

In the Figure 3 and Figure 4, right panels, are illustrated the processed images, obtained by interchange filters in the system using a single command input (order). The parameters applied to filters in the system are constant and were set to the following values:
- Brightness filter: value=30, sign=0;
- Contrast Filter: value=150, valueToAdd=5, valueToSubtract=5;
- Thresholding Filter: value=90.

Figure 3 illustrates Verilog results for filters order: 1.Brightness Filter, 2.Gray Filter, 3.Contrast Filter, 4.Thresholding Filter (order = 8b’00011011).


In the Figure 5 and Figure 6, right panels, are illustrated the processed images, as an effect of changing the parameters applied to filters in the system. The filter order remains unchanged: 1.Gray Filter, 2.Brightness Filter, 3.Thresholding Filter, 4.Contrast Filter (order = 8b’01001110).

Figure 5 illustrates Verilog results for filters order: 1.Gray Filter, 2.Brightness Filter, 3.Thresholding Filter, 4.Contrast Filter (order = 8b’01001110).
In the Figure 5, the parameters were set to the following values:
- Brightness filter: $value=40$, $sign=0$;
- Contrast Filter: $value=100$, $valueToAdd=2$, $valueToSubstract=2$;
- Thresholding Filter: $value=80$.

In the Figure 6, the parameters were set to the following values:
- Brightness filter: $value=10$, $sign=1$;
- Contrast Filter: $value=180$, $valueToAdd=10$, $valueToSubstract=10$;
- Thresholding Filter: $value=80$.

![Fig.6 Results of the real-time configurable system for DIP simulated and synthesized in Verilog](image)

The class of image processing algorithms considered in our work is limited to basic algorithms and further work on complex image processing algorithms will be performed. This application offers solutions for problems that you may encounter in the real time processing of images, particularly in the case of medical images or in any area where image processing is required in real time and the processing speed is vital.

4 Conclusion
The greater future potential of the proposed system lies in fact that it is not necessary to use an additional processor dedicated to image processing, which could slow down the flow: image acquisition $\rightarrow$ preprocessing the image $\rightarrow$ image use in order to establish a decision which can be human-type (in the case of a diagnosis based on medical images) or non-human (in the case of an industrial process).

The solution proposed in this paper is carried out a partial shift from imaging systems based on Digital Signal Processors (DSP) to image processing systems based on reconfigurable hardware (FPGA).

References: