FPGA Design of a Camera Control System for Road Traffic Monitoring

George K. Adam, Georgia Garani, Dimitrios Ventzas
Computer Science and Engineering
Technological Educational Institute of Thessaly,
Larissa, Greece

***Abstract*—Video surveillance and monitoring more than a decade now remains a big challenge. Today is becoming more and more vital due to rapid development of hardware equipments as well as the software methods that support intelligent monitoring and analytics. This paper proposes the architecture of a FPGA design of a camera controller for surveillance that could be applied for road traffic monitoring and image data management purposes. The proposed system could deal with real-time monitoring of persistent and transient objects within the specific environment of a motorway or crossroad, taking input from a camera. Functional models were synthesized in VHDL. The design and simulations analysis were carried out within Quartus II Altera software package while further analysis was also performed with ModelSim. The functional performance and operation of the proposed design was evaluated based on such simulations and analysis.**

Keywords—monitoring; camera controller; FPGA design; simulations

I. INTRODUCTION

Video surveillance systems that use analog CCTV cameras and interfaces as the basis are not easily expandable, and have low video resolution with little or no signal processing. However, next generation of video surveillance systems with digital LAN cameras, that support complex image processing, and video-over-IP routing are more intelligent and allow further scalability and flexibility [6], [7], [8]. In this direction, programmable modules, with or without embedded DSP blocks, with memory blocks, several interfaces, and other units, fulfill the new contemporary system requirements [5], [12].

Today several systems are designed on the same chip, based on a Field Programmable Gate Array (FPGA), for intelligent and fast data processing and miniaturization of the system. Particularly, in modern surveillance systems, FPGA-based intelligent controllers play an important role in improving the performance of the camera control system [2], [3], [4], [16], [19].

The purpose of this paper is to present first results of an ongoing research project on the implementation of a smart camera control system for traffic surveillance based on a FPGA. The problem of providing automated vision surveillance on an efficient way that last years has been intensively examined [10], [11], [13]. Today there is a rapid development of such systems towards intelligent control units [1], [9], [15] and several algorithms have been proposed towards this direction [14], [17].

In this research, the system proposed captures a video stream, computes traffic information and performs on chip video analytics and finally transfers the video stream and the traffic information to a host PC in order to be generated further alarm signals and events to be triggered according to various traffic situations. The objective of the paper is to describe the design of an intelligent surveillance system within an academic environment for traffic monitoring purposes. The proposed architecture incorporates a camera controller that obtains traffic images and performs analytics. It performs contour identification, axis specification, axis shift, percent zoom, smooth, etc., as well as facilities for mathematical and statistical analysis of the numerical data (pedestrians, cars, etc.), such as: densities, deviations, aggregates, etc.

In section II is described the design methodology followed and the goals aimed to be achieved. In section III the system architecture is detailed and section IV discusses the simulations carried towards the evaluation of system's functionality. Finally, section V presents a brief discussion and conclusions on the proposed system.

II. DESIGN METHODOLOGY AND GOALS

A. Design Approach

The overall design of the IP surveillance camera controller is to be implemented as a complete system on a chip using a FPGA device. An FPGA-based architecture is quite flexible and customizable, to such extend that easily allows a system architecture to be enhanced and modified according to various system specifications and requirements. Thus FPGA architectures are easily reconfigurable.

In addition, in our case study of remote traffic monitoring would be possible to add custom video processing software in real time (video analytics) and system upgrades (i.e. a new FPGA programming file uploaded to the embedded processor) remotely over the Ethernet.

Such design requirements are satisfied by most of the FPGAs providers, like ALTERA, XILINX, etc. Adopted is an open design philosophy so that the actual design to be targeted to any ALTERA FPGA (e.g. Cyclone III or IV) or even to another vendor's FPGA [18], [20].

In our system design, Altera's Quartus II development software is adopted that provides a complete, multiplatform design environment for all phases of FPGA design.

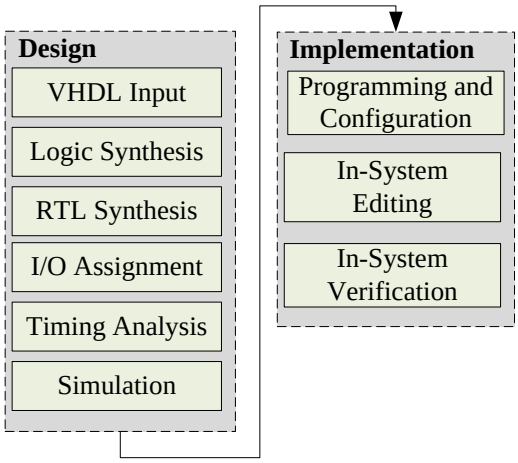

Fig. 1. FPGA design and implementation flow.

Our design and implementation flow (Figure 1) based on Quartus II development software involves the following stages:

- VHDL input: the actual design entry where the desired system is specified by using a hardware description language, VHDL.
- Logic synthesis: the described circuit is synthesized into logic gates (gate level).
- RTL synthesis: builds the actual device configuration netlist (transistor – register transfer level).
- I/O assignment: assigns the logic elements defined in the netlist in the actual routes and pins of the FPGA chip.
- Timing analysis: produces an indicative performance of the circuit, based on propagation delays analysis.
- Simulation: mainly functional simulation is performed.
- Programming and Configuration: design compilation and generation of device programming and configuration files.
- In-system editing: the actual designed circuit is implemented in a physical FPGA chip by programming.
- In-system verification: overall verification and circuit performance analysis.

B. *Overall project goal*

This work is part of a research project that aims towards the implementation of a smart camera control system for traffic surveillance. In video traffic surveillance, car detection and tracking constitute the minimum level building block necessary. Thus, detecting changes in a camera video-stream is the basis for an intelligent analysis.

The proposed surveillance camera control system (see Figure 2) overall contains most of the essential modules for video capture, video analytics, and event generator of control signals and alarms. A host PC could be connected to perform the overall configuration of the proposed system as well as additional traffic data management, using a specific software module developed in-house for this purpose (on Visual Basic), called TDAP tool (Traffic Data Analysis and Processing tool). This tool consists of database structural schemes capable to manage and store image and video traffic data and provides functions for basic image data analysis on various traffic situations.

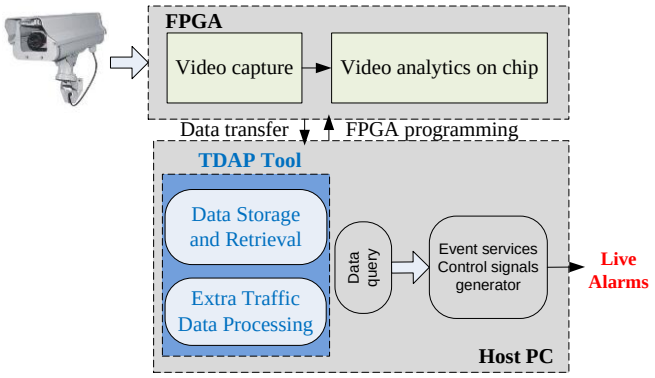

Fig. 2. General layout of the proposed surveillance camera control system.

III. The FPGA System Architecture

A. *Preliminary investigation and experimentation*

Initially, in order to determine the appropriate devices and equipment as well as to check and test various conditions and environmental parameters, a host pc with a simple camera was used as the camera microcontroller, as shown in Figure 3.

Sample experiments were carried out using a personal computer equipped with a simple Connectix Quickam camera (4-bit gray scale movies) with frame rates up to 24fps, powered from computer keyboard port (draws less than 350 milliwatts of current). PC image processing software provided by the vendor was used to capture the images for further processing. Image data received from the camera was 4 bits per pixel (up to 320 x 240 pixel capture). A JPEG standard for video data compression was chosen as well as a BMP without compression for the image format. The microcomputer-based control system employs an Intel Pentium 4 CPU board (2.4GHz, 2GB address space) under Windows XP operating system.

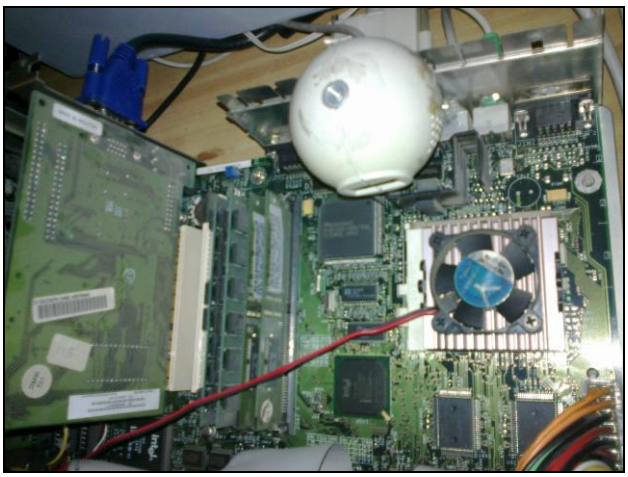

Fig. 3. Top view of the computer-based simple camera test-bed system.

Although such a system was not suitable for real-time traffic management applications, such as detection of vehicles and pedestrians, or red light violations and potential accidents, however was useful as a testbed for concluding on various image processing functions as well as on the proposal of an efficient hardware implementation.

DSP could have also been used (as shown in Figure 4), since image and video processing algorithms designed specifically for DSPs have certainly higher performance than that of the microprocessor on the PC. However, in our case of intelligent camera surveillance controller the decision was that FPGAs were the only choice for the design due to project specifications.

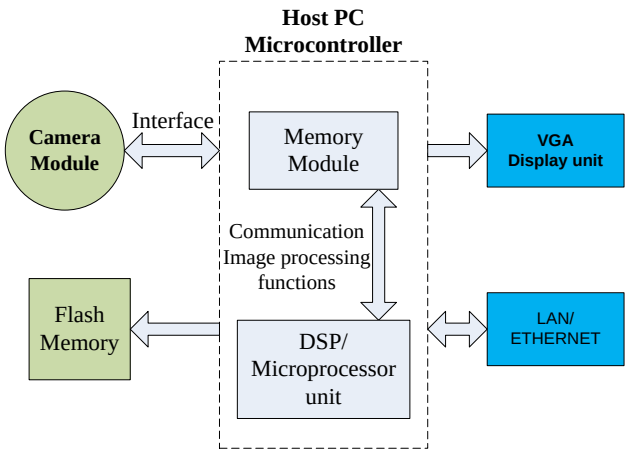

Fig. 4. Block diagram of surveillance camera controller based on DSP/CPU.

Another interesting solution which is under investigation is the combination of an FPGA plus an external DSP processor. In such a case, the design could follow a FPGA plus DSP development path. Towards this direction, vendors like ALTERA and MathWorks have joined forces and produced an interesting development environment that allows DSP development combining the benefits of FPGAs. In particular, allows the designer to perform the algorithmic development in the MATLAB software and system-level design in the Simulink software, and then port the design to hardware description language (HDL) files for use in the Quartus II FPGA development software

B. FPGA-based architecture

FPGAs can be developed to implement parallel design methodology, which is not easily applicable in dedicated DSP designs. A FPGA is a device that consists of an array of basic logic cells that can be configured after fabrication using a certain programming technology. FPGAs offer all of the features needed to implement complex designs as embedded systems.

This work proposes a camera surveillance controller designed and simulated on an ALTERA FPGA. ALTERA is one of the vendors that offer a wide range of choices to developers for use in FPGA-based designs, with embedded and configurable cores.

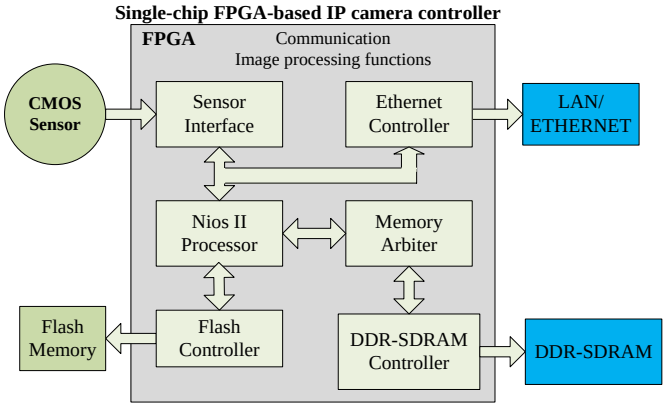

Fig. 5. Top-level block diagram of the IP surveillance camera controller reference design.

Figure 5 shows a simplified block diagram of an FPGA-based surveillance camera control system.

The basic layout consists of input, output, processing, and storage blocks such as, a network IP camera (Complementary Metal-Oxide Semiconductor (CMOS) sensor), an Ethernet interface controller that provides network access, a memory arbiter that controls data exchange between processor and the DDR-SDRAM block, a flash memory controller, other specialized hardware units and ports for communication and programming (through a JTAG) and finally, the main Nios II soft-core processor which is interfaced with the remaining devices. This is a programmable processor implemented on the FPGA for controlling the various processes implemented and performed on the board platform. The Nios II soft-core processor used in this implementation automates connecting soft-hardware components to create a complete camera control system. This is done through a FPGA programming file sent to the Nios II embedded processor, which writes the file to flash memory.

Such a design specifies that all the modules are implemented on a FPGA. However, designing systems with embedded processors requires both hardware and software elements. So, the actual implementation consists of custom hardware and specific software that runs on the developed

hardware. A host PC does the FPGA programming and is used also for debugging and additional image processing functions. A VGA display is used to view the streamed video output of the IP surveillance camera.

For specific image data analysis and management purposes was developed an in-house traffic data analysis and processing tool (TDAP tool). This tool was used to perform simple image processing techniques over a database of image and video traffic data. Figure 6 provides a simplified view of its structural environment. This application provides basic traffic data management functions within the database, such as: retrieve, update, filters, reports, queries, etc. Further facilities are provided for processing road map data (raster or matrix images, etc.), such as: contour identification, axis specification, axis shift, percent zoom, smooth, etc., as well as facilities for mathematical and statistical analysis of the numerical data (pedestrians, cars, etc.), such as: densities, deviations, aggregates, etc.

Some of the important processing facilities are as follows:

- Image processing, such as spatial representation, color processing, bit-level processing, median filter processing, inverse, edge detection, contrast manipulation, sharpening, etc.
- Spatial and temporal data management
- Statistical processing, such as arithmetic averages, population distributions, percentage deviations, geometrical image transfer and transformation, standard deviation, aggregations, classification, regression, etc.

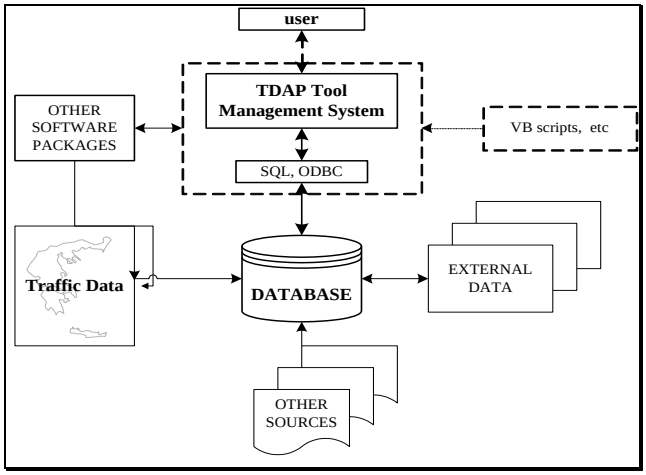

Fig. 6. Structural schema of TDAP tool.

Overall, the proposed control system could deal with real-time monitoring of persistent and transient objects (cars, pedestrians, etc.) within the specific environment of a highway, motorway or crossroad, taking input from a camera placed on the road side. The FPGA receives its input signals from the camera sensor through an interface that converts it into RGB format and stores it in the SDRAM. The type of image compression plays an important role on hardware system requirements, such as memory units, data rate, external storage units, etc. This is because data must be compressed to a manageable data rate that can be routed over an interface or the network. So, the surveillance system proposed will use the H.264 bandwidth-limited network, since a color transmission, e.g. at 30 fps at 640 x 480 pixels minimum requires a data rate of 26 Mbytes/sec.

Beyond the proposed hardware the software as well inside the FPGA performs specific video processing and provides video output and event signals based on traffic monitoring situations. The external SDRAM is being used as a buffer to store the incoming video data (image frames captured) that is to be sent to the host PC. An SDRAM controller performs the addressing and signaling tasks to read and write from/to the external SDRAM unit.

C. Further project work

A prototype of the proposed camera surveillance control system could be developed based on an ALTERA development board that features a Cyclone III or IV FPGA chip, using the Quartus II Web Edition design tools to specify the settings for the Nios II processor, to add peripherals and select bus connections, I/O memory mappings, IRQ assignments, etc.

Such a platform could be installed either on a vehicle or on road traffic infrastructure equipment (e.g. traffic lights or signs). The system could be easily accessed by using remote PC to which it communicates and which perform the configuration procedures and additional image and video processing capabilities.

IV. SYSTEM SIMULATIONS

Although most of the simulations and analysis was carried out within Quartus II software package some further analysis was also performed with ModelSim software package. The functional performance and operation of the proposed design was evaluated based on such simulations and analysis. Design simulations were consisted of functional models synthesized in VHDL and were executed using ModelSim software package. The VHDL code produced was partitioned into several process statements according to the described architecture. The architecture of the simulated controller with all important signals is visible in the VHDL simulator. The whole length of the simulation model code used in system's functionality verification is not presented due to its substantial length. Overall the proposed design seems to have an optimal functionality.

The actual performance of the proposed system design of the control system remains to be tested under real-time conditions and should be compared to the specifications defined during the initial design stages. In this way, during real-time test procedures, several system internal data would be collected and analyzed for verifying system's efficiency and accuracy. All the tests procedures would be carried out based on real-time traffic monitoring cases.

V. Discussion and Conclusions

An accurate design of a camera controller for surveillance is essential for road traffic monitoring and image data management purposes. This paper proposes a FPGA-based method of designing an essential camera controller. Simulation models were used in functional evaluation of the controller. The controller seems to be capable of achieving real image data and realize efficient flow control as captured through traffic monitoring. The validity of the proposed system design remains to be evaluated through actual experiments on traffic road monitoring.

Acknowledgment

This research has been co-financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: ARCHIMEDES III. Investing in knowledge society through the European Social Fund.

References

[1] K. Srinivas, Implementation of FPGA in the design of Embedded Systems, Thesis, Department of Electronics and Communication Engineering, National Institute of Technology, Rourkela, May, 78 pages, 2007.

[2] Eric Monmasson, and Marcian N. Cirstea, FPGA Design Methodology for Industrial Control Systems—A Review, IEEE Transactions on Industrial Electronics, Vol. 54, No. 4, August 2007.

[3] T. N. Chang, B. Cheng, and P. Sriwilaijaroen, "Motion control firmware for high-speed robotic systems," IEEE Trans. Ind. Electron., vol. 53, no. 5, pp. 1713–1722, Oct. 2006.

[4] M. Gabrick, R. Nicholson, F. Winters, B. Young, and J. Patton, "FPGA considerations for automotive applications," in Proc. SAE Conf., CD-ROM, 2006.

[5] R. Dubey, P. Agarwal, and M. K. Vasantha, "Programmable logic devices for motion control—A review," IEEE Trans. Ind. Electron., vol. 54, no. 1, pp. 559–566, Feb. 2007.

[6] In Su K, Hong Seok Choi, Yi Kwang Moo, Choi Jin Young, and Kong Seong G. Intelligent visual surveillance — a survey. International Journal of Control, Automation, and Systems, 8:926–939, 2010.

[7] Valera, M. and Velastin, S.A. Intelligent distributed surveillance systems: a review. IEE Proc. . Vis. Image Signal Process., Vol. 152, No. 2, pp. 192.204, April 2005.

[8] Tomi D. Raty. Survey on contemporary remote surveillance systems for public safety. IEEE Transactions on Systems, Man, and Cybernetics, 493–515, September 2010.

[9] Peter L. Venetianer and Hongli Deng. Performance evaluation of an intelligent video surveillance system – a case study. Computer Vision and Image Understanding, 114(11):1292–1302, 2010.

[10] Dee, H., Velastin, S. A. "How close are we to solving the problem of automated visual surveillance? A review of real-world surveillance, scientific progress and evaluative mechanisms". Machine Vision and Applications, 19 (5-6), pp. 329-343, Sept. 2008.

[11] Foresti G., Micheloni C., Snidaro L., Remagnino P., Ellis T., "Active video-based surveillance system: the low-level image and video processing techniques needed for implementation," IEEE Signal Processing Mag., vol. 22, no. 2, pp. 25–37, March 2005.

[12] G. L. Foresti, P. M¨ah¨onen, and C. S. Regazzoni, editors. Multimedia video-based surveillance systems. Kluwer Academic Publishers, Boston, 2000.

[13] C. J. Costello, C. P. Diehl, A. Banerjee, and H. Fisher. Scheduling an active camera to observe people. In Proc. 2nd ACM International Workshop on Video Surveillance and Sensor Networks, pages 39–45, ACM Press, New York, NY, 2004.

[14] R. Collins, A. Lipton, H. Fujiyoshi, and T. Kanade. Algorithms for cooperative multisensor surveillance. Proceedings of the IEEE, 89(10):1456–1477, October 2001.

[15] Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, S. Pankanti, S., "Smart video surveillance: exploring the concept of multiscale spatiotemporal tracking," IEEE Signal Processing Mag., vol. 22, no. 2, pp. 38–51, March 2005.

[16] V. B. Jagdale, R. J. Vaidya, High Definition Surveillance System Using Motion Detection Method based on FPGA DE-II 70 Board, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-2, December 2012.

[17] Mohamed F. Abdelkader, Rama Chellappa, Qinfen Zheng, ―Integrated Motion Detection and Tracking for Visual Surveillance‖ Proceedings of the Fourth IEEE International Conference on Computer Vision Systems (ICVS 2006) 0-7695-2506-7/06, 2006.

[18] Altera DE2-70 Development & Education Board User Manual Version 1.08, Terasic Technologies, 2009.

[19] Haoming Zeng, FPGA based smart NIR camera, Thesis, Mid Sweden University, The Department of Information Technology and Media (ITM), 2012.

[20] ALTERA, Five Ways to Build Flexibility into Industrial Applications with FPGAs, Altera Corporation, www.altera.com, 2011.