proceedings

MDPI

Proceedings

An FPGA Platform Proposal for Real-Time Acoustic Event Detection: Optimum Platform Implementation for Audio Recognition with Time Restrictions [†]

Marcos Hervás [1,*,‡] and Rosa Ma Alsina-Pagès [1,*,‡]

[1] GTM—Grup de Recerca en Tecnologies Mèdia, La Salle—Universitat Ramon Llull. C/Quatre Camins, 30, 08022 Barcelona, Spain

[*] Correspondence: mhervas@salleurl.edu (M.H.); ralsina@salleurl.edu (R.M.A.-P.); Tel.: +34-932-902-445 (M.H.); +34-932-902-455 (R.M.A.-P.)

[†] Presented at the 3rd International Electronic Conference on Sensors and Applications, 15–30 November 2016; Available online: https://sciforum.net/conference/ecsa-3.

[‡] These authors contributed equally to this work.

Published: 14 November 2016

Abstract: Nowadays, monitoring of people and events is a common matter in the street, in the industry or at home, and acoustic event detection is commonly used. This increases the knowledge of what is happening in the soundscape, and this information encourages any monitoring system to take decisions depending on the measured events. Our research in this field includes, on one hand, smart city applications, which aim is to develop a low cost sensor network for real time noise mapping in the cities, and on the other hand, ambient assisted living applications through audio event recognition at home. This requires acoustic signal processing for event recognition, which is a challenging problem applying feature extraction techniques and machine learning methods. Furthermore, when the techniques come closer to implementation, a complete study of the most suitable platform is needed, taking into account computational complexity of the algorithms and commercial platforms price. In this work, the comparative study of several platforms serving to implement this sensing application is detailed. An FPGA platform is chosen as the optimum proposal considering the application requirements and taking into account time restrictions of the signal processing algorithms. Furthermore, we describe the first approach to the real-time implementation of the feature extraction algorithm on the chosen platform.

Keywords: acoustic sensor; signal processing; machine learning; real-time; FPGA; VHDL

1. Introduction

Monitoring all kind of human activities has never been as common as today, when it is usual to have many sensors spread along the city, some factories or even our homes. Many surveillance systems operate nowadays, taking advantage of all the data that the current technology enables us to obtain and process [1].

The Grup de Recerca en Tecnologies Mèdia (GTM) has conducted several research projects in the study of acoustic signal processing and event recognition for different applications. In indoor applications, an approach to ambient assisted living purposes was conducted in [2] to help the diagnosis of the first stages of people with dementia. For outdoor applications, especially oriented to smartcity environments, we have detected types of vehicles [3], classified events in soundscapes [4] and even specifically in surveillance applications [5].

Nowadays, GTM is involved in two projects related to audio event detection. DYNAMAP is a project funded by the European Commission (LIFE ENV/IT/001254) which aim is to develop a low cost sensor network for real time noise mapping in the cities [6]. GTM develops in the project an Anomalous Event Detection Algorithm in order to avoid the noise computation of any other event but the traffic noise to calculate the noise maps of the city [7]. The second project, named HomeSound (2014-SGR-0590), consists of programming a low-cost GPU platform [8] for the audio event detection of fifteen in-home common sounds (e.g., water, walking, glass breaking, dog barking, etc.). The GPU platform is capable of computing the feature extraction and the machine learning methods to classify the environmental sounds real-time, and send the results to the cloud to be registered via Ethernet, or even to activate any kind of alarm. The real-time implementation of the conducted projects in GTM led us to the study of the best hardware platform in terms of efficiency and cost to implement these algorithms.

This paper is structured as follows. Section II explains several characteristics from the existing hardware platforms, while Section III details the hardware proposal. Finally, the conclusions of this first approach to the real—time low cost Field Programmable Gate Array (FPGA) proposal are enumerated.

2. Hardware Platforms Comparison

In this section, a brief comparison of several hardware platforms is performed. In this sense, leader microcontroller manufacturers are: (i) Renesas Technology; (ii) Freescale Semiconductor; (iii) ST Microelectronics; (iv) Microchip Technology; (v) NXP Semiconductors; (vi) Texas Instruments and (vii) Infineon Technologies [9], who provide general purpose 32-bit microcontrollers (MCU) for Internet of Things (IoT) and metering for low cost and low power embedded applications.

These families of MCU are based on ARM Cortex-M or proprietary architectures and they are able to work up to 240 MHz. Cortex-M0 and M0+ are used for low cost, low area and designed for higher performance. Cortex-M4 and Cortex-M7 include floating point and DSP capabilities [10].

However, these microcontrollers are not recommended in intensive real-time applications; despite Cortex-M4 and Cortex-M7 include floating point and DSP capabilities, due to the high cost of implementation of the signal processing algorithms. In a typical case of environmental audio event recognition case, the sampling frequency may be 48 kHz and an overlap of 50% between frames of at least 30 ms is desirable. Therefore, the cycle count and the time execution of the required algorithms, such as windowing, FFT, 48 FIR filters, DCT, etc. turns to be around 15 ms. Moreover, the system has to manage the TCP/IP stack and the acquisition process. This time estimation has been extracted from Table 1, where the execution cost in terms of cycle count and time for both a FIR filter of 32 samples and different FFT size are shown using the STM32F10x Digital Signal Processing (DSP) library from STMicrocontroller for the ARM cortex M architecture.

For this reason, a higher performance device should be used, such as application processors, true DSP or application processors with GPU coprocessor. Currently, there are some low cost open-source hardware platforms based on ARM cortex-A such as BeagleBoard, Raspberry Pi, CubieBoard, PhidgetSBC and UDOO [11], where Cortex-A is the architecture of the application processor provided by ARM. The cost of these platforms is from 35$ to 150$.

3. Hardware Proposal and Basic Algorithm Implementation

In this paper we propose to choose a low-cost alternative platform based on programmable logic able to exploit algorithm parallelization for real time applications.

The main difference between MCU and FPGA is their architectures and their programmability paradigm. Whereas a program in a MCU is executed as a sequential series of instructions, a FPGA contains an array of discrete logic resources that can be fully configurable to implement any algorithms which can fit it. The implementation in FPGA of any algorithm may take benefit from the ability to parallelize any part of the implementation. Initially, FPGAs were very expensive devices used generally for prototyping but currently the price is similar to an application microprocessor. The power consumption of the FPGA is higher than any MCU or DSP, because every part of the MCU or DSP has been designed and optimized to execute a deterministic function. However, FPGAs can be fully programmable to do any task the user is able to program.

Table 1. Execution time of the FFT and FIR algorithm for different number of points and different system frequency for a CORTEX-M3 [12].

FFT (ASM)	24 MHz Cycle Count	24 MHz Time (μs)	48 MHz Cycle Count	48 MHz Time (μs)	72 MHz Cycle Count	72 MHz Time (μs)
FFT-64	3847	160	4025	84	4764	66
FFT-256	21,039	876	22,176	462	26,065	362
FFT-1024	100,180	4174	102,057	2126	127,318	1768
FIR-32	3516	146.5	3525	73.4	3727	5176

3.1. Platform Description

Initially, a FPGA comprised a matrix of configurable logic blocks (CLBs) connected through programmable interconnects. Nowadays, these devices may or may not have additional hardware that increase the performance of these elements.

- Memory Control Blocks (MCB) to manage auxiliary DDR memories.
- Digital Clock Managers (DCMs) able to modify some aspects of the clock signals such as (i) multiply or divide the input frequency; (ii) condition a clock; (iii) phase shift; (iv) eliminate clock skew and (v) mirror, forward, or rebuffer a clock Signal.
- Block RAMs to implement two independent 18 kbits RAM or one 36 kbits in Xilinx series 7 FPGA.
- A DSP block which include a pre-adder, multiplication and accumulator able to implement different functions of digital signal processing.

The platform presented is the low cost Basys-3 developed by Digilentinc [13], and it comprises an Artix-7 Xilinx FPGA (see Figure 1). The logic resources available are shown in Table 2.

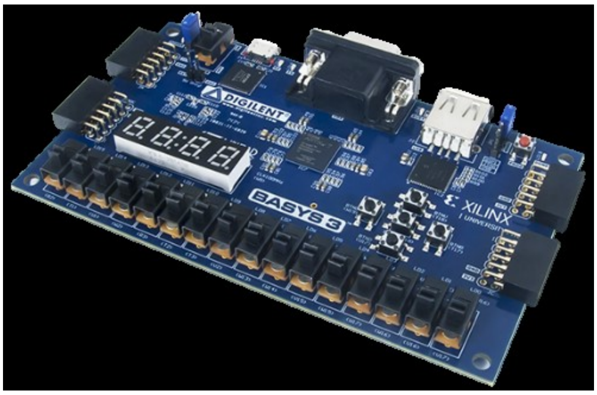

Figure 1. Basys-3 platform developped by Digilentinc based on an Artix-7 [13].

Table 2. Logic Resources of the BASYS-3 platform.

Basys-3	Slices	Logic Cells	Bloc RAM	DSPs	Price
XC7A35T-1CPG236C	33,280	33,280	1800 kbit	90	150 $

3.2. Algorithm Description

The implementation of the firmware in a FPGA follows two criteria: (i) performance optimization which pretends to increase the maximum usable frequency in the design and (ii) area optimization which is based in reducing the number of logic resources required.

The algorithm presented in this paper is a proof of concept to evaluate the use of these kind of platforms in real time digital signal processing for acoustic event recognition problems. The block diagram for the feature extraction algorithm is shown in Figure 2. The implementations presented are: (i) windowing; (ii) FFT; (iii) 48 GTCC [5] filter banks and (iv) square root, which are parts of the features extraction, and they all meet a good trade-off between area and speed optimization. We assume that the audio frames are of 30 ms length, which corresponds with 1440 samples at 48 ksps. For this reason, the length of the programmed FFT is of 2048 samples.

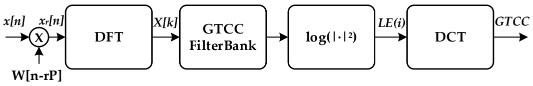

Figure 2. Block diagram of the feature extraction signal processing.

A Hamming windowing has been implemented with a series of 2048 registers, where the last samples are stored. When a predefined number of new data is introduced, the windowing process starts. A reduction area optimization has been carried out, saving the coefficients in 1 Block Ram, working as a Read-Only Memory (ROM) and multiplying the value of the memory for every address with the datum stored in the flip-flop with the same index. The output of every multiplication, which is done at 100MHz, correspond to the data input of the FFT. In Figure 3, a block diagram of the implementation carried out in VHDL is shown.

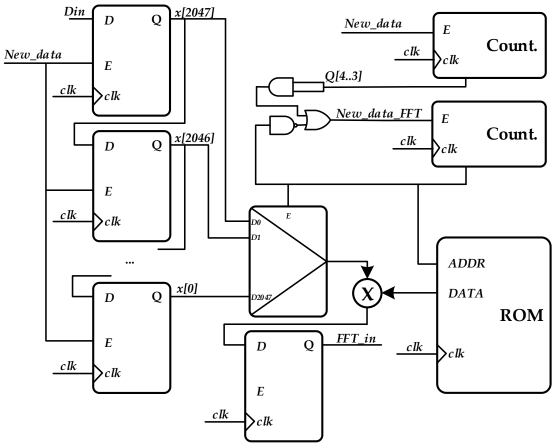

Figure 3. Implementation of windowing proposed to insert data to the FFT block.

The FFT implementation has been done with a Xilinx IPCore. The algorithm selected is radix2 with 2048 transform length, it requires 26,659 transform cycles, and the resources used are summarized in Table 3. The transform cycles is much lower than the ARM implementation presented in Table 1, with a high number of data inputs. The algorithm selected was carried out to minimize the logic resources needed.

Finally, the implementation of the module of the FFT with the multiplication, ADD and SQRT blocks, and the parallelization of the 48 filter banks developed in series is shown in Figure 4.

Table 3. Resources used by the presented implementations

Basys-3	LUT	FF	BRAM	DSP
FFT	709	1385	4	4
48 Filter Banks	0	0	48	0
Square Root	783	0	0	0
Total	7949	24,800	11	25

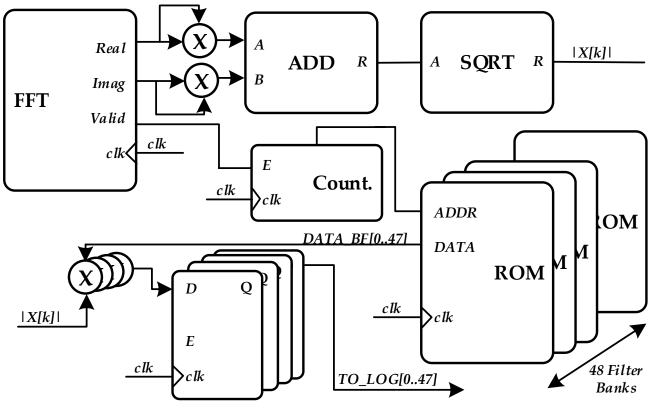

Figure 4. Basys-3 platform developped by Digilentinc based on an Artix-7.

4. Conclusions

We conclude that the Basys-3 FPGA platform is a good trade-off between cost and features for the audio detection algorithm implementation. It satisfies the restrictions for the real-time performance in typical conditions for that application. After the proof of principle of the test of the feature extraction presented in this paper, we plan to develop several machine-learning algorithms in VHDL to work in the FPGA, and evaluate the cost of the entire algorithm performing in the proposed platform. The results presented in this work encourage us to the use of this programmable logic platform able to exploit parallelization for real-time algorithms. This results encourage us to implement an embedded microcontroller in the FPGA, Microblaze, to control the system remotely through Ethernet and to compute easily non-intensive parts of the algorithm, due to the quantity of free resources available.

Acknowledgments: This research has been partially funded by the European Commission under project LIFE DYNAMAP ENV/IT/001254 and the Secretaria d'Universitats i Recerca del Departament d'Economia i Coneixement (Generalitat de Catalunya) under grant ref. 2014-SGR-0590.

Author Contributions: Marcos Hervás has performed the simulations and has written part of the paper. Rosa Ma Alsina-Pagès works for DYNAMAP project and conceived the tests, and wrote the other part of the paper.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

FPGA	Field Programmable Gate Array
VHDL	VHSIC Hardware Description Language
GTM	Grup de Recerca en Tecnologies Mèdia
ARM	Advanced RISC Machines
MCU	Microcontrollers
DSP	Digital Signal Processor
CLB	Configurable Logic Blocks
MCB	Memory Control Blocks
DCM	Digital Clock Managers
GTCC	Gammatones Coefficients
FFT	Fast Fourier Transform
ROM	Read Only Memory
RAM	Random Access Memory

References

1. Crocco, M.; Cristiani, M.; Trucco, A.; Murino, V. Audio Surveillance: A Systematic Review. *J. ACM Comput. Surveys* **2016**, *48*, 52.
2. Guyot, P.; Valero, X.; Pinquier, J.; Alías, F. Two-step detection of water sound events for the diagnostic and monitoring of dementia. In Proceedings of the 2013 IEEE International Conference on Multimedia and Expo (ICME), San Jose, CA, USA, 15–19 July 2013.
3. Valero, X.; Alías, F. Hierarchical Classification of Environmental Noise Sources by Considering the Acoustic Signature of Vehicle Pass-bys. *Arch. Acoust.* **2012**, *37*, 423–434.
4. Valero, X.; Oldoni, D.; Alías, F.; Botteldooren, D. Support Vector Machines and Self-Organizing Maps for the recognition of sound events in urban soundscapes. In Proceedings of the Inter-Noise 2012, New York, NY, USA, 19–22 August 2012.
5. Valero, X.; Alías, F. Gammatone Wavelet Features for Sound Classification in Surveillance Applications. In Proceedings of the 20th European Signal Processing Conference (EUSIPCO 2012), Bucharest, Romania, 27–31 August 2012.
6. Sevillano, X.; Socoró, J.C.; Alías, F.; Belucci, P.; Peruzzi, L.; Radaelli, S.; Coppi, P.; Nencini, L.; Cerniglia, A.; Bisceglie, A.; et al. DYNAMAP—Development of low cost sensors networks for real time noise mapping. *Noise Mapp.* **2016**, *3*, 172–189.
7. Socoró, J.C.; Ribera, G.; Sevillano, X.; Alías, F. Development of an Anomalous Noise Event Detection Algorithm for dynamic road traffic noise mapping. In Proceedings of the 22nd International Congress on Sound and Vibration (ICSV22), Florence, Italy, 12–16 July 2015.
8. JETSON TK1. NVIDIA. Available online: http://www.nvidia.com/object/jetson-tk1-embedded-dev-kit.html (accessed on 15 May 2016).
9. Teng, A.; Blanco, A.; Van, G.; Reilly, N. Market Share Analysis: Microcontrollers, Worldwide, 2014. Gartner Inc. Available online: https://www.gartner.com/doc/3048717/market-share-analysis-microcontrollers-worldwide accessed on 26 May 2015).
10. Smallest and Lowest Power Cortex Processors—Optimized for Discrete Processing and Microcontrollers. ARM Ltd. Available online: http://www.arm.com/products/processors/cortex-m/index.php (accessed on 15 May 2016).
11. Maksimović, M.; Vujović, V.; Davidović, N.; Milošević, V.; Perišić, B. Raspberry Pi as Internet of Things Hardware: Performances and Constraints. In Proceedings of the 1st International Conference on Electrical, Electronic and Computing Engineering (IcETRAN 2014), Vrnjačka Banja, Serbia, 2–5 June 2014.
12. UM0585. STMicroelectronics. Available online: http://users.ece.utexas.edu/~valvano/EE345M/UM0585.pdf#page6 (accessed on 12 June 2010).
13. Fergenson, D. Basys 3 Artix-7 FPGA Trainer Board: Recommended for Introductory Users. Digilent Inc. Available online: http://store.digilentinc.com/basys-3-artix-7-fpga-trainer-board-recommended-for-introductory-users/ (accessed on 6 January 2016).