Preface

The concept of reconfigurable computing was devised much earlier than the first devices capable of bringing into light the theoretical capabilities of such new computing paradigm. Since the mid 80’s, when the first Field Programmable Gate Array was launched to the market, reconfigurable technology business has had a steady growth to date.

Since its appearance in the electronic market in 1985 FPGAs have been gaining more and more space in the electronic design world. Its new concept of reconfigurable hardware has seduced designers to adopt this technology in a world of fast and demanding electronic consumer needs changes. The main advantage of reconfigurable hardware is that it allows designing and re-designing an electronic component according to user needs after the fabrication process.

Also, because of their regular fabric, FPGAs have benefited from the constant improvement in the fabrication technology under Moore’s Law. Starting with a few thousands of gates, FPGAs now can bring to designer hundreds of thousands of logic cells, DSPs slices, blockRAMs of internal memory, high speed serial transceivers, user configurable analog/digital converters, etc., increasing enormously their field of application. Besides, FPGA vendors provide users with a plethora of already optimized components, designed to face any kind of system development and restriction. Depending on the vendors and models, FPGAs offer different primitives and hardware components such as embedded processors, DSP blocks, memory, A/D and D/A converters, etc., that facilitate electronic design, contributing to reduce complexity of design and time-to-market.

Latest technologies have led to the integration of complete reprogrammable Systems on Chip, composed by multicore processors (such as ARM Cortex M1, CortexM3, Microblaze, among others) and reconfigurable logic in an all-in-one solution. Most important FPGA vendors, such as Xilinx, Altera, Microsemi (Xilinx, 2014; Altera, 2015; Microsemi, 2014), have adopted this heterogeneous technology. Reconfigurable logic can be managed from the software side tailoring the logic to cope with the acceleration needs of the software algorithms.
Since last years FPGA become a key device in many modern instrumentation systems FPGA-based user-configurable instrumentation has been incorporated in their product portfolios by several vendors, such as National Instruments (http://www.ni.com/fpga-hardware/), Agilent, Sundance (http://www.sundancedsp.com/datasheets/VideoGuru-Flyer.pdf). Agilent (Agilent, 2011-2012) provides instruments such as logic analyser with support for FPGA design debugging. Xilinx has developed logic cores for embedded instrumentation to monitor internal signals of a design with user-configurable number of probe ports, and including many advanced features available in modern logic analysers (Xilinx, 2014b). Besides, companies have developed designs flows and tools for FPGA-based design using the hardware in the loop (HIL) approach such as Labview (National Instruments, 2014), Matlab (Mathworks, 2016). Using HIL it is possible to simulate a plant using a model or a virtual representation together with an implemented version of a FPGA-based controller.

However, there exist applications with specific needs that are beyond the capabilities of standard instrumentation. Currently available equipment can’t offer dedicated and customizable characteristic according to specific problem to be faced. Therefore, the development of special and case-specific instrumentation opens a very attractive set of solutions to provide a wide range of methods and tools for signal processing and measurement. Scientific applications requiring data analysis, data intensive acquisition and processing can take benefits from the parallel structure offered by FPGAs. Their inherent parallelism and reconfigurability can be managed and handled to create and develop tailored instrumentation to face specific and non-standards problems. Aspects such as custom triggering, specific control algorithms, high speed and parallel data acquisition, response times within a clock cycle, computation of large data sets, or extremely fast data transfer, and high-speed signal processing, among others, are suitable of being implemented using FPGAs. Another interesting aspect to consider is the opportunities provided by bit-level manipulation to create specific and user-defined numerical representations and custom data reduction to improve testing performance.

The available instrumentation currently existing in the market hasn’t enough flexibility to cope with so specific requisites. In most of the cases the configuration capabilities are generic and hard to be adapted. Many of these new applications require a processing of the sampled information using complex algorithms that go beyond just a specific sequence of triggering, e.g. parameterizable noise generator with Gaussian distribution. For that reason, FPGA-based instrumentation can be found nowadays in several disciplines, cryogenic instrumentation platform based on FPGA (Conway Lamb et al., 2016), nuclear instrumentation (Bobin, Bouchard, Pierre, & Thian, 2012), universal embedded digital instrumentation (Ferry, 2013), just to name a few of them.
One of the new possibilities to explore in the space of solution for user-configurable instrumentation is partial dynamic reconfiguration of FPGAs. Partial dynamic reconfiguration gives the flexibility to change or modify part of a design whereas the rest is running without to stop the whole system. This interesting characteristic can be exploited to develop adaptable instrumentation that can follow the evolution of the system to be measured or treated.

As can be seen, the design of adaptable platforms for FPGA-based user-configurable instrumentation has an enormous added value to be exploited, that can be summarized as follow: the possibility to create and develop special instruments not already available in the market; the possibility to offer cheaper alternatives, where the same generic equipment may be reused in different situations. This is of a special interest in the case of developing countries, where the budget can be quite limited, or simply because the acquisition of this kind of equipment is subject to a long bureaucratic process. And the possibility to have not only a standalone instrument, but a subsystem that may be embedded in the design. That may be the case, for example, of the embedded analyzer commercial tools such as ChipScope (Xilinx) or Signal Tap (Altera), or the one proposed in Chapter 6.

The purpose of the book is to illustrate how the use of reconfigurable devices such as FPGAs can provide several solutions to the different challenges which have been identified before. Thanks to their compile-time reconfiguration capabilities, FPGA-based designs can be customized at design time to provide solutions to a specific need. Here several solutions are possible depending on the approach followed. Sometimes design are reused as black box subsystems, while in some other cases reuse comes in the form of a library of components plus a set of integration tools and/or methods. In any case, reuse addresses both the problems of the definition of customized instruments, not available by any other means, and the cost of the specific equipment created to face with a specific magnitude or problem to measure or observe.

We will consider an instrument in a broad sense. Sometimes it will be a full design with one concrete purpose, such as a virtual platform, while in other cases it will be a set of hardware modules that could be reused for a certain type of applications: genetic operators, hardware random number generators. Or even it will be a methodology as it is the case of embedded debugging.

Some of these solutions are not new at all, but the community have been working on then for the last decades. However the current technology with a higher capacity of integration, and heterogeneous processing elements integrated in the same chip, makes them especially suitable for dedicated instrumentation, fostering the use of FPGAs for reducing and improving the design time and facilitating the creation of hybrid hardware/software instrumentation.
The book is composed of 10 chapters which have been structured in four sections. The first three are devoted to three different concrete topics, while the fourth one includes a number of applications belonging to different fields.

Section 1 includes two chapters related to nuclear physics applications. This is a field specially well suited for the use of FPGAs for many reasons. In some cases it is required the processing of a huge amount of information coming from the observation of physical phenomena, such as particle collision detection. The processing of that type of information can take profit of the fine-grain parallelism level exhibited in the FPGA reconfigurable fabric. In some other cases it may not be a problem of information volume, but simply it may be required a really low latency response, beyond that provided by a microprocessor-based system. One third situation is that were only a hardware solution can be applied, mainly because safety or security reasons, as it is the case for certain really critical subsystems in a nuclear facility.

That last case is exactly the one described in Chapter 1, where motivated by the need of upgrading old nuclear power reactors of the Russian type, an FPGA-based controller is presented to replace a part of the electronic circuitry. More concretely, it analyses the replacement of the control rods used to regulate the core activity of the reactor, or even stop it completely in a few seconds. Here the benefits of using FPGAs are more related to their flexibility, the improved testability, and the reduction in the drift which may occur in analog-based systems, than on its computational power. The work explores two different design approaches, one based on a finite-state machine, and a pure combinational implementation, and it discusses the benefits and drawbacks of each one, taking always safety into account as a major concern in these kind of installations.

Chapter 2 is devoted to the analysis of algorithms for gamma and x-ray digital spectroscopy, which have many applications in the fields of research, industrial, medical and security. One of the problems regarding this kind of spectroscopy is the implementation of correction methods to isolate overlapping peaks and summing effects obtained during the measurement of the radiation of radionuclides. In this sense, the work proposes recovery and correction algorithms for such problems, which are well suited for their FPGA implementation.

Chaos exists in many highly complex nonlinear systems, and how it has been reported to provide a successful application in AD converters, nonlinear oscillators, or random number generators, among others. Therefore, Section 2 relates to the implementation of solutions that can be applied to applications modelled after the chaos theory, and the use of chaotic maps. One big challenge in the design of such kind of systems is the numerical representation used, and the balance between resources consumed and precision at the time of being implemented in hardware. Another problem is the design of hardware random number generators, one of the
possible applications of chaotic maps, such as the ones used to model Gaussian noise in communication systems.

A review of previous research on FPGA-based implementations of chaotic maps (one and two-dimensional ones) is presented in Chapter 3. Most of these works use a fixed-point numerical representation, and try to reduce the effect of degradation increasing the number of bits used for representation. Alternatively, the chapter proposes an alternative model which can be both implemented using fixed-point or floating-point arithmetic. The results are then validated using the previous review of the state of the art as the reference framework.

Chapter 4 discusses the implementation of a random number generator, using a Gaussian distribution, and based on a chaotic map. Gaussian random numbers are widely used in the field of signal communication. In this case, the generator is applied to the modelling of Additive White Gaussian Noise channels. Those kinds of channels are widely used in the evaluation of communication systems, since AWGN is a standard in that field. The main contribution of the work is the proposal of a hardware implementation using the Gaussian distribution, while most related works make use of a uniform and therefore less realistic one.

The two chapters in Section 3 cover the topic of virtual instrumentation under two different points of view. One is based in the concept of a versatile hardware platform that provides a set of different possible hardware interfaces and communication protocols that can be configured to conform customized equipment. Such hardware platform is complemented with the software counterpart which adds the extra signal processing, application glue logic and graphical user interface. A completely different alternative is the instrumentation of the end user application, embedding customized monitors as part of the running system, using partial dynamic reconfiguration. Such instrumentation can be very valuable during the debug and test of the application providing extra observability capabilities to the designer, while it will be removed for the final design.

Chapter 5 introduces the concept of Reconfigurable Virtual Instrumentation in which reconfigurable hardware can be turned into different instruments on demand, through a software procedure following the software defined reconfigurability approach. Although in this paper a scientist and academic perspective was adopted, conclusion drawn along the chapter can be adopted or extended for applications in industrial sectors.

Chapter 6 describes the Internal System Monitor using dynamic reconfiguration capabilities of FPGAs. This chapter presents a dynamically reconfigurable monitoring system designed to verify implemented designs in FPGAs, which can be deployed in a reconfigurable area, and that can be removed after completion of the design verification process. The system provides the mechanism to create and
adapt cores into a static design, to facilitate a customized post-deployment monitoring and verification process.

Section 4 groups a set of different applications, ranging from image processing for image retrieval to the design of a general purpose reusable wireless sensor network node. The main benefits that the use of FPGA provides in all these cases can be summarized into the following:

- The resulting infrastructure can be easily extended or modified to customize the design depending on the concrete requirements of the final user, as it is the case of the wireless sensor network node.
- The FPGA implements the core of computation of a problem that requires a very low latency and with a fine-grain parallelism, which matches the inherent architecture of the FPGA fabric. Typical problems that match that description are the ones related to digital signal processing and signal encoding/decoding techniques.
- The hardware implementation provides a more efficient implementation with respect to the ratio between computational power and power consumption, for example for embedding complex algorithm processing such as the motion path planning described in Chapter 10.

In Chapter 7 you will find a complete implementation of a Content Based Image Retrieval (CIBR) system on a hybrid FPGA. CIBR is typically used for similarity searches on very large collections of complex objects, where the time required for the indexing and retrieval of the images is critical. That may be the case, for example, for equipment used in bioscience, or real-time content delivery, such as surveillance video cameras or biometric systems. The CIBR system is based in the use of a metric space where a visual descriptor characterizing an image query is computed and checked against a pivot-based indexed database. The hybrid nature of the SoC used favours the reuse of the solution, where the FPGA fabric implements the digital processing task, while the search task is performed in software, thus making it quite simple to explore different algorithmic alternatives.

Another application related to digital signal processing is the detection and validation of digital signals immersed in noise, or interfered by other signal sources. This is a frequent operation regardless of the target application, and the typical solution is the use of coding techniques combined the correlation function to improve the performance of signal detection. Such techniques should provide a good signal-to-noise ratio as well as a set of uncorrelated codes, which is not easily achieved. The solution proposed in Chapter 8 focuses on the optimization of such coding techniques based on the use of complementary sequences, as a way to provide both uncorre-
lated and low SNR codes. Communication systems, radars, ultrasonic sensors and indoor positioning are some of the applications that can take benefit of these codes.

Wireless sensor networks are being used in a large number of fields: agriculture, health, defence, robotics, etc. Chapter 9 proposes the design of a reconfigurable smart sensor node, based on the IEEE 1451.0 communications standard. The modular architecture of the node makes it possible to customize it with the addition or removal of sensor functionality, depending on the concrete use of the sensor, while maintaining the rest of the infrastructure. Additionally, the node can be extended through the modification or addition of the internal protocols used for the interface between the transducers and the sensors network.

Finally, in Chapter 10 a general purpose set of hardware genetic operators are used as the basis for the implementation of a robot motion path planning algorithm. General hardware modules are provided for the selection, crossover and mutating array operations, as well as a 16-bit random number generator. This is complemented with a population and fitness evaluation modules which have been specifically developed for the specific application. There resulting design could be therefore adapted to new contexts by simply rewriting the software modules. It is also worth considering the approach for autonomous embedded systems where the computational power of the microprocessors is quite limited.

Julio Daniel Dondo Gazzano  
University of Castilla – La Mancha, Spain

Fernando Rincon Calle  
University of Castilla – La Mancha, Spain

Maria Liz Crespo  
International Centre for Theoretical Physics, Italy

Andres Cicuttin  
International Centre for Theoretical Physics, Italy

REFERENCES


