FPGA Prototyping and Validation of an EPC Gen 2 RFID Tag

Lauro Puricelli[#1], Robert Torrel[#2], Herbert Peralta[#3]

[#]*CENTRO NACIONAL DE TECNOLOGIA ELETRÔNICA AVANÇADA - CEITEC S.A.*
Av. João de Oliveira Remião, 777, Porto Alegre – RS - Brazil

[1]lauro.puricelli@ceitec-sa.com

[2]robert.torrel@ceitec-sa.com

[3]herbert.peralta@ceitec-sa.com

***Abstract*— This paper presents the methods used to validate the digital hardware implementation of an Electronic Product Code Class 1 Generation 2 (EPC Gen 2) RFID IC (ASIC) using an FPGA prototyping board. The prototype was functionally verified against a commercial EPC Gen 2 compliant reader. FPGA synthesis results are presented. The experimental results ensured that the hardware implementation was functionally correct.**

Keywords*— *RFID, EPC Gen-2, FPGA, Prototyping Validation

I. INTRODUCTION

RFID (Radio Frequency Identification) technology has been adopted in many applications such as product tracking, animal identification and security devices. The basic unit for an RFID system is called Tag, which consists of a microchip attached to an antenna. The Tag's information can be accessed without any physical contact, using radio frequency (RF) waves sent by the RFID reader.

Considering the UHF (Ultra High-Frequency) class of RFID Tags, the EPC Class 1 Generation-2 UHF Air Interface Protocol, also known as EPC Gen 2 [1], is widely used by the industry.

A complex system like EPC Gen-2 could demand a lot of time for verification, considering the mixed-signal nature of the IC. The simulation of each part of the chip (digital and analog) rely on models that may neglect some aspects of the system. Also, the validation of the IC with an actual RFID reader is only possible after fabrication.

In order to improve this process, one possible strategy that can be adopted is to prototype the digital design into an FPGA. This method can guarantee that errors and bugs not yet discovered could be found before the fabrication of the chip in a silicon-based platform, i.e. the FPGA.

This work presents the prototyping of the digital section of a EPC Gen-2 based RFID IC using an FPGA. A prototype board containing an analog front-end is used.

This work is divided as follows: section II presents a brief description about RFID system; section III describes the strategy used to prototype a digital section of the RFID IC into a FPGA as well as the synthesis results; section IV describes the test setup and the results; finally, section V presents the conclusions related to the developed work.

II. RFID SYSTEM

RFID is a popular automatic identification technology. This systems consist of three main components: an antenna, a reader and a tag (composed of an RFID IC and an antenna). There are three different ranges of operation frequency: LF (Low Frequency – 125/134.2kHz), HF (High Frequency – 13.56 MHz) and UHF (Ultra High Frequency – 865 ~ 930 MHz)[2].

The project of an RFID IC comprises mixed-signal circuits (RF, analog and digital) for modulation and demodulation of an electromagnetic wave and its conversion to digital signals that are processed according to the protocol chosen.

As stated before, for the UHF RFID, one used pattern in the industry is EPC Gen 2. This standard defines a communication protocol that includes error detection scheme and an anti-collision algorithm based on the use of pseudo-random numbers.

EPC Gen 2 protocol uses FM0 and Miller encoding. The interrogator communicates with one or more Tags by modulating an RF carrier Using DSB-ASK, SSB-ASK, or PR-ASK with PIE (Pulse Interval Encoding) [1].

A tag communicates with an Interrogator using backscatter modulation, in which the Tag switches the reflection coefficient of its antenna between two states according to the data being sent. The tag can be seen as a finite state machine**,** whose response depends on the current state and the command received [3].

Tag-to-interrogator link frequencies vary from 40kHz to 640kHz and is the determined by query commands sent by the reader.

III. FPGA PROTOTYPING

RFID ICs are mixed-signal ICs composed of a digital section and an analog section. Considering only the digital section of the IC, the development flow comprises architectural definition, RTL coding, logic synthesis and physical implementation (placement, CTS, routing). Throughout the development flow the design is verified using a verification environment and a set of test-cases. This verification environment emulates the stimuli that are applied into the chip

(digital section). Signals from the Analog-to-Digital interface are modelled as well as the commands from the RFID system "reader".

Due to the fact that the verification environment uses models to emulate the application, some unexpected aspects of the system may not be accounted for. This may result in problems during chip testing after fabrication.

In order to accelerate hardware validation and to increase the reliability of the functionality, an FPGA prototype was used. This strategy is largely used in the industry [4, 5] and it allows the designer to find and fix some errors that could not be detected before the tape-out phase, saving time and avoiding rework (e.g. any wrong interpretation of the standard may be verified against a commercial reader prior to tape-out).

Figure 1 shows the difference between the ASIC flow and the FPGA prototyping flow. The FPGA prototype flow uses RTL code as an input, synthesises this code into a gate netlist and runs Place&Route (using the proper set of constraints). As an output a bitstream file is generated and used to configure the FPGA.

The main difference between them is in tape-out phase. When the wafer comes from foundry, it pass through the process of thinning, dicing, packaging and then, each individual chip is assembled to be tested its functionalities. This steps can takes months depending on the kind of process made.

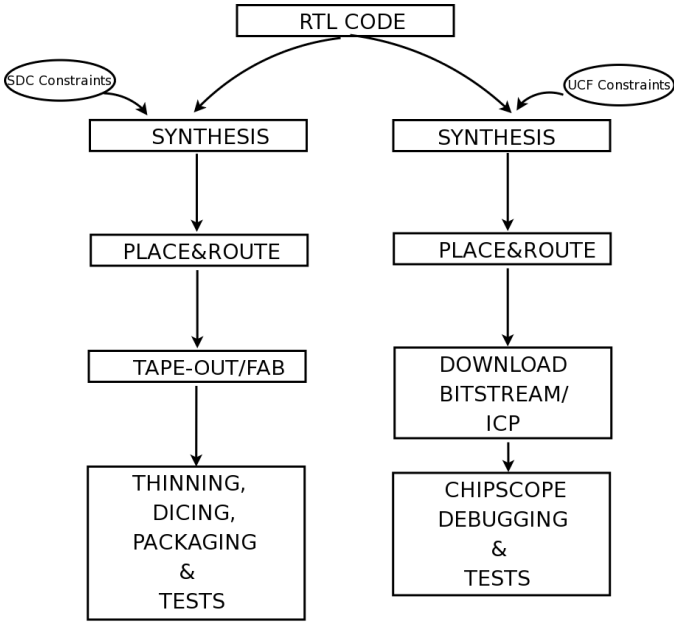

Figure 1. ASIC and FPGA Flow

Note that, this flow allows a fast chip implementation of the digital circuit. By combining this FPGA implementation and an analog front-end, using a prototype board, it is possible to validate the functionality of the chip using a commercial RFID reader. Additionally , the change of some RTL block and then the simulation of all ASIC can take more than a week, programming the FPGA with the modifications needed in a digital block takes half hour.

The main challenge to develop a prototype of a RFID IC is related to the fact that it is a mixed-signal chip. Therefore, the prototype board needs to combine both the digital circuit (FPGA) with the analog circuit. For this project, it was used a board with Xilinx Spartan-3 FPGA and an analog front-end.

In order to use the same RTL developed for the ASIC into the FPGA, some changes were necessary. First of all, it was created an extra digital block to replace an analog circuit that was missing from the analog front-end. This was necessary because the analog front-end from the prototype board was not fully compatible with our ASIC specifications.

Also, the memory IP was replaced for the FPGA block RAM components. The memory contents must be initialised with proper values according to the test being performed.

A wrapper was necessary to interconnect the digital circuit to the FPGA interface pins (input/output pins). Some of the FPGA input pins were used to send to and receive from the chip stimulus according to the protocol. Another input pin was used to reset the tag's state machine. Some FPGA output pins were connected to the leds in the prototype board used to show the state of some important registers, error signals, tag's operation mode and others according to the test's requirements.

Furthermore, the constraints used in the ASIC flow were adapted to FPGA constraints (e.g. clock definition). Also some specific FPGA constraints were defined (e.g. buffers and I/O pins).

Scripts were used to automate as much as possible the flow. ISE 14.1 tool was used to run FPGA synthesis, Place&Route and bitstream file generation**.** Impact tool, was used to upload de bitstream in FPGA using a JTAG interface. Moreover, some tools provided by Xilinx Company like Chipscope was used for debugging purposes.

FPGA synthesis results are presented in Table 1.

TABLE I
SYNTHESIS RESULTS

Logic Utilization	Used	Available	Utilization
Total number Slice Registers	1686	15360	10%
Number used as Flip-Flop	1683		
Number used as Latches	3		
Number of 4 input LUTs	4965	15360	32%
Number of occupied Slices	3074	7680	40%
Number of Slices containing only related logic	3074	3074	100%
Number of Slices containing only unrelated logic	0	3074	0%
Total Number of 4 input LUTs	5278	15360	34%
Number used as logic	4965		
Number used as a route-thru	313		
Number of Bonded IOBs	13	173	7%
Number of RAMB16s	3	24	12%
Number of BUFGMUXs	3	8	37%
Average Fanout of Non-Clock Nets	4.16		

IV. Tests and comparisons

The test setup included a computer, a RFID reader and the prototype board (with both the digital circuit and an analog front-end). This setup is shown in Figure 2.

By using an Ethernet interface to communicate with the RFID reader, all protocol settings were loaded using the computer. A set of scripts were used to automate test generation in the reader (e.g. reading and writing in memory banks, singulate a specific tag, and send different set of EPC commands to the Tag).

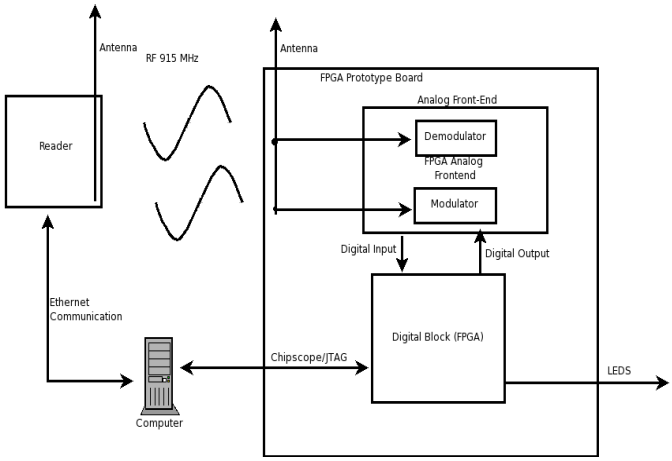

Figure 2. Schematic of workstation

Between each test, the tag (FPGA) was restarted using the reset button to simulate the expected behaviour of the chip. As proof that the tests were successful, the developed scripts had a report saying if the results were ok or not, according to the specification of each kind of test. If the results were not as the expected, some actions were taken such as the review of RTL, the repetition of the test or even assure that the reader was configured with a compatible and correct configuration.

The tests made with the FPGA were done to verify its conformance with the protocol. For this, besides the tests described above, we used several techniques as using the Chipscope program to validate the digital registers in real time debugging, use the FPGA's leds, input and output pins to stimulate some digital blocks, send commands manually from the reader to verify the correct execution of the test (e.g. when writing in a memory bank), read this bank to see if the word was written correctly and compare some results with another commercial or test tags. Figure 3a shows the wave sent by the reader in FM0 Encoding. Figure 3b shows the communication between a tag and a reader. It can be seen that for each command sent by the reader (bigger signal) there's a corresponding response by the tag (smaller signal).

Some comparisons of performance (e.g. numbers of reads per second) between the FPGA prototype and commercial tags were made and the results were comparable with commercial tags.

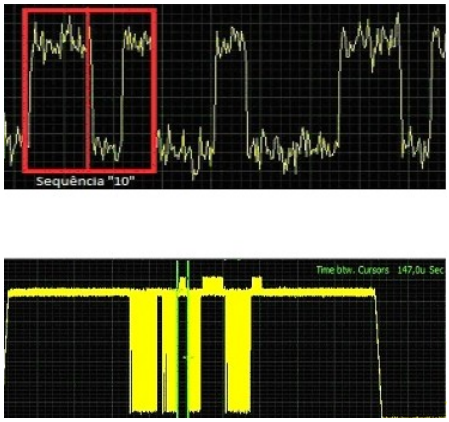

Figure 3.
Figure (3a) Data in FM0 encoding
Figure (3b) Reader-to-tag communication

V. Conclusions

An RFID IC prototype was successfully developed using an FPGA for prototyping the digital section of the chip. The prototype was functionally verified against a commercial EPC Gen 2 compliant reader with an extensive range of tests that covered almost all functionalities of the chip. The approach described proved to be very useful for jobs that requires rapid prototyping techniques and a fast behaviour simulation of the final RFID IC. The results of tests performed were very important because it proved to the team the correctness of work being developed.

References

[1] EPCGlobal. The global language of business. http://www.gs1.org/epcglobal, 2008.

[2] A. K. Jones, A Field Programmable RFID Tag and Associated Design Flow, Field-programmable Custom Computing Machines, 2006. FCCM '06.

[3] V. R. Pamula and J. Bernhard, Model Based Development of the Digital Part of a RFID Transponder with Xilinx System Generator for a FPGA Platform, 2012 Fourth International EURASIP Workshop on RFID Technology

[4] C. Angerer, B. Knerr, M. Holzer, A. Adalan, and M. Rupp. Flexible Simulation and Prototyping for RFID Designs. International EURASIP Workshop on RFID Technology, 2007.

[5] Huansheng Ning, Yu Cong, Z.-Q Xu, T. Hong, J.-C. Zhao and Yan Zhang. Performance Evaluation of RFID Anti-Collision Algorithm with FPGA Implementation. 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW), 2007.