A Flexible Design Flow for a Low Power RFID Tag

José C. S. Palma

PPGC - II - UFRGS - Av. Bento Gonçalves, 9500, Porto Alegre, RS – Brazil
jcspalma@inf.ufrgs.br

César Marcon, Fabiano Hessel, Eduardo Bezerra, Guilherme Rohde, Luciano Azevedo, Carlos Reif, Carolina Metzler

PPGCC - FACIN – PUCRS - Av. Ipiranga, 6681, Porto Alegre, RS – Brazil
{cesar.marcon, fabiano.hessel, eduardo.bezerra, grohde }@inf.pucrs.br

Abstract

This paper describes the implementation of a passive RFID tag targeting low power implementation, which works on 915 MHz UHF frequency. The proposed architecture allows customizing the command sets implemented inside its digital block, according to the target application needs, saving area and reducing power consumption. A flexible design flow is proposed for the customization, verification and synthesis of the digital block, targeting low power requirements.

Keywords*: RFID Systems, Tag customization, low power consumption.*

1. Introduction

RFID (Radio Frequency IDentification) technology exists since the Second World War, but only nowadays, it is becoming attractive [1]. Technology advances and components costs reduction motivate it.

RFID systems can be used in applications such as supply chain management, tracking, and security. Several large companies have shown interest in this technology since Wall Mart's announce of start employing RFID.

RFID systems are composed of tags, readers and antennas. Tags exchange data with the reader (or interrogator) through radio frequency (RF) communication [2]. The reader has one or more antennas that send/receive radio waves to/from tags. The number of antennas depends on the communication area required by the application, since the use of more antennas implies larger communication area.

Tags can be classified as active or passive. Active tags have their own power supply, operating in high frequencies and usually are larger than passive tags. Passive tags are low power and need to be smaller, since they do not have internal power supply. They transform the communication RF signals from reader in energy to operate. As a result, passive tags are cheaper than the active ones [3].

This paper describes a passive RFID tag, which has the following features: low power, 915 MHz and communicates by means of UHF. The tag is class 1 generation 2, according to ISO 18000-6B and EPC standards. In addition, the paper also describes its flexible design flow, which allows command set customization according to application requirements. Tag area usage and power consumption can be optimized by selecting a customized and reduced command set. This work focuses only on the implementation, customization and validation of the tag's digital block.

This paper is organized as follows. Section 2 presents some works related to RFID systems. Section 3 presents the architecture of the proposed tag. Section 4 describes the gate-level energy flow used to estimate the tag power consumption. Section 5 discusses the flexible design flow used for tags building. Section 6 describes the entire system validation. Section 7 shows occupation area and power consumption resulted by synthesis step and, finally, Section 8 presents conclusions and ongoing work.

2. Related Work

Choi et al. [4] present a 13.56 MHz RFID reader for home security applications, which allows multi-tag recognition. Leong et al. [5] propose an approach to synchronize multiples RFID readers in order to enable successful dense RFID reader deployment, and minimize the reader collisions.

An important issue on RFID systems is the reading distance, typically limited to less than 1.2 m. Karthaus and Fisher [6] propose a passive UHF transponder with reading distance increased to 4.5 m by using a different voltage generator, a PSK in backward link, and a careful layout and antenna matching.

This work, similarly to [7], [8], [9] and [10], presents the design of a flexible low power RFID tag. The main contribution is the tag flexible design aiming to reduce area and power consumption.

3. The Proposed Architecture

Figure 1 shows the basic diagram of RFID tag architecture, which has three main blocks: External Antenna, Analog Block and Digital Block. The External Antenna interfaces the IC tag to the environment, and it is responsible for capturing data as electromagnetic waves. The tag power supply is implemented by the Analog Block that converts the electromagnetic waves

978-1-4244-1710-0/07/$25.00 ©

into electric energy, which is stored in a bank of capacitors. In addition, this block is responsible for analog and digital signals demodulation/modulation. Finally, the Digital Block performs all tag operations, such as data decoding, commands interpreting, collision arbitration, sent data encoding and memory access. Six modules described next perform these functionalities.

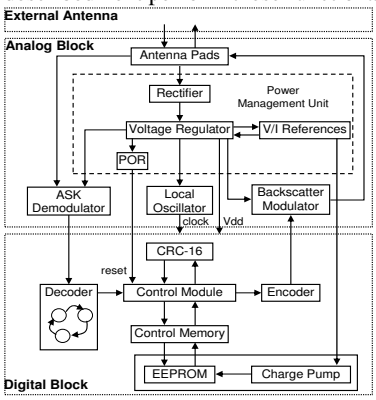

Figure 1 – RFID tag architecture diagram.

The Decoder receives from the ASK Demodulator a 40 Kbps or 160 Kbps Manchester encoded data, which is decoded to a binary format that is sent to the Control Module. Before starting data decoding, it is necessary to synchronize the input signal with a 640 KHz clock provided by the Local Oscillator module.

The Control Module identifies and handles all commands received from the Reader, sending signals and controlling the other digital modules. The CRC-16 module creates a 16-bit cyclic redundancy check of a new packet and checks the integrity of a packet received from the reader. When a CRC error is detected, all data are discarded and the frame is suspended. The Control Memory module is the interface between the Control Module and the EEPROM Memory. It is responsible for enabling the memory access and for returning the instruction results. The tag Digital Block has an internal 512 bits EEPROM, arranged in 8 rows of 4 words. The Encoder module codifies the output frame received from the Control Module, and that must be sent to the Backscatter Modulator module. The coding is done according to FM0 encoding method.

4. Gate-level Energy Flow

Simulations based on a library of gates provide the energy consumption estimation of the tag. The gates, which are described in VHDL, are pre-characterized according to the static and dynamic energy consumption.

Figure 2 presents the flow used to characterize the gate-level energy model and to obtain the energy consumption estimative of the circuit under analysis. The characterization begins with the selection of a set of gates, which are manually described in SPICE and VHDL. After that, the set of gates described in SPICE are simulated at electrical level, considering the adopted technology library (CMOS TSMC 0.35 μm). Each gate is simulated in a SPICE simulator with all input signals combinations and different *fan-outs*, calculating the dynamic energy consumed by the gate due to input transitions.

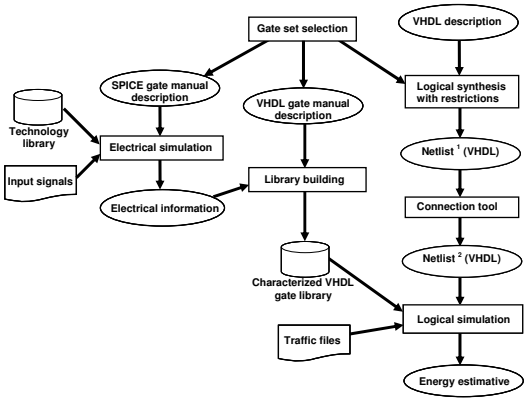

Figure 2 – Flow used to characterize the gate-level energy model and to obtain the energy consumption estimation.

Since leakage current grows dramatically as the feature size of CMOS circuit scaling down and it is the main parcel to static energy consumption [11]. It is also necessary to model the static energy of gates. This is performed by simulating the same gate described in SPICE during periods without transitions. In this case, it is not necessary to perform simulations with different *fan-outs*, since it does not interfere in the static energy consumption. The total static energy consumption is achieved when multiplying the total static energy by the execution time. The sum of static and dynamic parcels computes the total energy consumed by the whole system.

Performing electrical simulations enables to characterize all gate by inserting electrical information to the VHDL description of each gate, creating the characterized VHDL gate library. The next step is to perform the logical synthesis of the circuit under analyzes. It is executed restricting the synthesis tool to use only the set of gates previously characterized on the VHDL gate library. The synthesis tool generates a VHDL netlist, which is analyzed by a connection tool. This tool verifies all connections among gates, calculating the *fan-out* of each one. Such information is important to estimate more accurately the energy consumption of the circuit.

The connection tool generates a second netlist with *fan-out* information, which is simulated in a VHDL simulation tool. Such simulation uses the characterized VHDL gate library and input traffic files to estimate the energy consumption of the circuit under analysis.

5. Flexible Design Flow

The proposed architecture allows the selection of different command sets, according to the target

application needs. This feature makes the RFID tag more flexible, when considering area, energy consumption and read/write range specifications.

Customization is performed over Digital Control module, which implements the set of commands supported by the tag. Since it is the most area and energy consuming module, as shows Table 2, the facility for performing different configurations in the command set will result in a significant impact on the overall tag costs.

Figure 3 shows the command select flow. It starts from the library of commands and an application specification file. Based on these inputs, the customer can select the set of commands to be implemented in the Digital Control module.

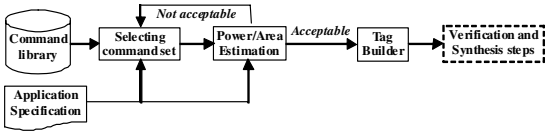

Figure 3 – Command select flow.

The next step is the area usage and energy consumption estimation, which is performed by some iteration. Each one selecting a new instruction set, in order to meet the expected area usage and energy consumption specified in the design requirements. This step corresponds to the right side of the flow presented in Figure 2 – starting from the VHDL description of the customized tag, until achieve its energy consumption estimation.

Table 1 shows area and energy consumption estimation of Digital Block of two different tag configurations. The first one is a complete tag, which implements all supported commands, according to ISO 18000-6B standard. This configuration corresponds to 28 commands, occupies 60060 transistors and consumes an average power of 258 μW. The second configuration is a tag, which implements only two commands: (*i*) *read* – reads one byte of data from the memory – and (*ii*) *write_four_byte* – writes four bytes of data in the memory.

Average power and energy estimative were obtained after the execution of the two commands supported for both tag configurations: *read* and *write_four_byte*. The reduction in the energy consumption was 25,4 % compared to the complete tag.

Table 1: Area and power estimations of Digital Block for two different tag configurations.

Version	Area (transistors)	Average Power (mW)	Energy (J)	%
Complete	60060	2.58e-1	5.2524e-6	100
Reduced	45432	1.924e-1	3.9256e-6	74.6

When all requirements are met, the next step concern the new tag manufacture, followed by validation and synthesis steps, described in the next section.

6. System Validation and Synthesis

This Section describes the system validation through functional and gate-level simulation. Before the validation stage, a tag was configured according to the command select flow, described in the last section. The configuration of a generic tag used, for instance in supply chain applications, has been chosen as a case study. Such tag supports all the commands determined by ISO 18000-6B standard, and some proprietary commands planed to accomplish supply chain purposes.

Figure 4 shows the system validation and synthesis flow, which is performed by functional and gate-level simulation.

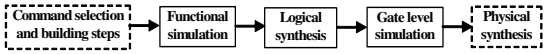

Figure 4 – System validation flow.

Four stages split the validation of the tag VHDL design using Modelsim. In the first three stages, it is performed VHDL functional simulation and, the last one is a gate-level simulation.

In the first stage, the digital modules were individually submitted to white box testing during the development, in order to verify their correct functioning. After this stage, the digital modules were interconnected, creating a new entity named Top Digital Block.

Second stage consists of a white box testing of Top Digital Block, which is performed using waveform analyses. The Top Digital Block has been simulated and its basic functions have been tested. The basic functions include receiving packets, decoding commands, checking CRC, building and sending responses. An auxiliary tool randomly generates frames containing commands, which are read from a text file.

In the third stage, the tests were automated, in order to perform massive verification of the Top Digital Block. First, a C program, named C-Tag, generates all Digital Block outputs according to its inputs. The C-Tag works executing the same stages of a real tag, which are data receiving, data decoding, command executing and response sending. In other words, it reproduces the Digital Block behavior under many conditions, following the ISO 18000-6B standard. This standard defines how this tag should work. Massive verification uses around of ten thousand commands per session, and each command have the same size and data of commands received by a real tag. The next step simulates both C-Tag and Top Digital Block. During C-Tag execution, all outputs and some internal debug signals are stored in a text file. In a similar way, signals from Digital Block are also monitored and stored in a second text file. After that, another external program analyses the data in the two text files and verifies if the behavior of Digital Block matches the C-Tag behavior. If no errors are detected, a new round of tests begins. Otherwise, the detected errors can be rechecked several times, simulating the same system state until the errors

remain. Simulation status is stored in a third text file. In case of error detection, all debug signals are also stored in this file. The simulation stop criterion is a 100% of code coverage and no errors detection. Once simulation stops, the system is ready for logical synthesis. This process creates a Verilog netlist and delay files.

The last stage consists of an automated gate-level simulation with netlist and delay files. The simulation process of the third functional validation stage is performed again, but now with gates and interconnections details. The netlist file is also used to estimate power consumption, while delay files have all signals delays according to connection wires characteristics.

When the simulation reached the stop criteria, the system was sent to the foundry. In a first moment, the foundry will provide a small set of tags for exhaustive testing activities. Afterwards, the final tag design will be used to produce a commercial set of tags.

7. Synthesis Results

Table 2 presents area and power results for Digital Block and its internal modules. The first three columns present, respectively, the total cell area occupied by each module, the total number of cells and the number of sequential cells inside each module. The difference in the number of cells between the Top Digital Block and the sum of all internal modules is due to the insertion of buffers, used for power optimizations in the Top Digital Block. The fourth to seventh columns show the typical and worst case of power consumption of each module in a 3.0 V with 25°C and in a 3.3V with 70°C scenarios.

TABLE 2: AREA OCCUPATION AND POWER CONSUMPTIONS RESULTS FOR DIGITAL BLOCK AND ITS INTERNAL MODULES.

Digital Block	Area occupation			Power consumption (w)			
	Total Cell Area (μm²)	Total Cell	Sequential Cells (FFs)	3.0V and 25°C		3.3V and 70°C	
				Typical	Worst	Typical	Worst
Decoder	34143.2	270	48	7.0454e-03	7.9445e-03	9.1280e-03	1.0269e-02
Control	440203.4	4471	340	3.6410e-02	4.0963e-02	4.7220e-02	5.3013e-02
CRC	7425.6	38	16	1.7745e-03	1.9919e-03	2.3047e-03	2.5814e-03
Memory Control	108945.2	972	125	1.3791e-02	1.5470e-02	1.7896e-02	2.0030e-02
Encoder	9864.4	96	14	1.2958e-03	1.4648e-03	1.6803e-03	1.8925e-03
Test	13722.8	123	2	3.0224e-03	3.4164e-03	3.9190e-03	4.4184e-03
Top Digital	616652.4	5998	545	8.8119e-02	9.9060e-02	1.1264e-01	1.2637e-01

Figure 5 shows the final layout of RFID tag circuit, which uses 0.35μm standard CMOS technology. As previous described in Section 3, this work focusing only in the digital block, which occupies more than 60 % of the entire circuit.

8. Conclusions and Ongoing Work

This paper describes the implementation of a passive RFID tag, which has the following features: low power, 915 MHz and communicates by means of UHF. The main contributions are: (*i*) a new tag architecture, which has been written in VHDL, simulated and synthesized; and (*ii*) a flexible design flow, allowing the commands set customization, aiming efficient power consumption.

Different configurations for the tag's command set have been tried, and a chip design for a typical application was chosen as a case study. The tag has been fully tested and validated at several simulation levels, and the synchronizing algorithm performed according to the expected. The resulting design has been sent to a foundry. Field tests are going to be performed on the actual chip, and a new and revalidated design version will be sent to the foundry for commercial production in large scale.

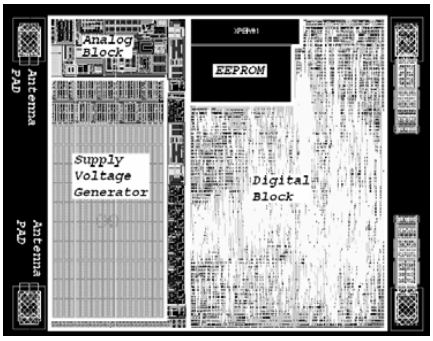

Figure 5 – Layout of the RFID tag.

9. References

[1] Want, R. The Magic of RFID Just How does Those Little Things Work Anyway? ACM Queue, v. 2, n. 7, Oct. 2004.

[2] Mamei, M. et al. *Making Tuple Spaces Physical with RFID Tags*. **Symposium on Applied Computing, ACM Press**, France, 2006.

[3] Weinstein, Ron. *RFID: A Technical Overview and Its Application to the Enterprise.* **IEEE Computer Society**, v. 7, n. 3, pp. 27-33, 2005.

[4] Choi, N-G. et al. *A 13.5 MHz RFID System*. Dec. 2005.

[5] Leong, K. et al. *Synchronization of RFID Readers for Dense RFID Reader Environments*. **International Symposium on Applications and the Internet Workshops**, Jan. 2006.

[6] Karthaus, U. and Fisher, M. *Fully Integrated Passive UHF RFID Transponder IC with 16.7 μW Minimum RF Input Power*. **IEEE Journal on Solid-State Circuits**, v. 38, n. 10, pp. 1602-1608, Oct. 2003.

[7] Leenaerts, D. *Low Power RF IC Design for Wireless Communication*. **International Symposium on Low Power Electronics and Design**. pp. 428-433, Aug. 2003.

[8] Vancherand, F. *New Technologies for Contactless Microsystems*. **ACM International Conference Proceeding Series**, v. 121, pp. 13-17, 2005.

[9] Porret, A-S. et al. *Tradeoffs and Design of an Ultra Low Power UHF Transceiver Integrated in a Standard Digital CMOS Process*. **IEEE Journal of Solid-State Circuits**, v. 35, n. 3, Mar. 2000.

[10] Jones, A. et al. *An automated, FPGA-based reconfigurable, low-power RFID tag*. **Proceedings of the 43rd Annual ACM IEEE Design Automation Conference**, pp. 131-136, 2006.

[11] X. Zhao, K. Wang, X. Cheng, D. Tong. *A Leakage Power Estimation Method for Standard Cell Based Design*. **IEEE Conference on Electron Devices and Solid-State Circuits**, p. 821-824, Dec. 2005.