Teaching FPGA Security (extended version)

Lilian Bossuet
Laboratoire Hubert Curien, UMR CNRS 5516
University of Lyon
Saint-Etienne, France

***Abstract*— Teaching FPGA security to electrical engineering students is new at graduate level. It requires a wide field of knowledge and a lot of time. This paper describes a compact course on FPGA security that is available to electrical engineering master's students at the Saint-Etienne Institute of Telecom, University of Lyon, France. It is intended for instructors who wish to design a new course on this topic. The paper reviews the motivation for the course, the pedagogical issues involved, the curriculum, the lab materials and tools used, and the results. Details are provided on two original lab sessions, in particular, a compact lab that requires students to perform differential power analysis of FPGA implementation of the AES symmetric cipher. The paper gives numerous relevant references to allow the reader to prepare a similar curriculum.**

***Index Terms*— FPGA security, data security, cryptography, education, digital system design, embedded system security**

I. INTRODUCTION

In recent years, there has been a shift and a blurring of boundaries between the areas of security-critical embedded systems (e.g. aviation, military, banking, and energy) and areas of "quality of service" embedded systems (e.g. the smart home, consumer electronics, commerce, multimedia, e-health, and the smart grid). Reliability requirements and strong security that were previously limited to the above mentioned areas will gradually extend to all areas of embedded deployment. As a result, FPGA security has become a hot topic of FPGA community. The emergence of FPGA security issues even for consumer products requires a large number of well-prepared engineers in the appropriate scientific and technical fields.

FPGA security concerns different fields: data security (confidentiality, integrity, authentication and non-repudiation), system security (bitstream security, user authentication, guaranteed quality of services, the war against malicious hardware) and the protection of intellectual property (IP protection, the war against counterfeiting, illegal copying and theft). In all cases, security is not simply a new application for FPGA but a new design constraint, like low power, high speed and low area. As a consequence, electrical engineering students need methods and tools to design tamper resistant circuits and systems. Teaching FPGA security implies understanding the potential vulnerabilities and attacks and developing design techniques and countermeasures to combat such attacks.

The importance of FPGA security is evidenced by the annual expansion of the market for secure embedded systems (software and hardware solutions) and the increased attention this topic now receives at top level technical conferences (e.g. DATE, DAC), at reconfigurable architectures dedicated-conferences and workshops (e.g. ICFPT, FPL, SPL, FPGA, FCCM, ARC, RAW, ERSA, ReCoSoC, ReConFig). Many new workshops are dedicated sessions to FPGA security (e.g. HOST, COSADE, CryptArchi, TRUDEVICE, and CHES). This trend is also apparent in the publication of new books [1]-[2] during the last three years, as well as new edition of recent book [3] dedicated to FPGA and reconfigurable architectures security.

Students who focus on VLSI design and hardware architecture (which is the case of most electrical engineering majors), will benefit from instruction in FPGA security to enable them to design secure (reconfigurable) systems. For students to qualify in this field, special courses on FPGA security should be available. Unfortunately, most courses that teach security focus on software security and target either students in computer science or information systems [4] who already have the pre-required knowledge.

This paper describes a method that has been used for teaching FPGA security at graduate level at the Bordeaux Institute of Technology since 2006, and, since 2010, at the Saint-Etienne Institute of Telecom, University of Lyon. The paper is organized as follows: in section II, we describe the pedagogical issues addressed by the course. The course itself is described in section III. In section IV, we give a brief description of the lab sessions and micro-project subjects. In section V, we present data concerning the strengths and weaknesses of the course content and in section VI, we present a number of conclusions.

II. PEDAGOGICAL ISSUES

Most of the pedagogical issues described in this section were first presented at the CryptArchi workshop in 2007 [5]. The talk on teaching FPGA security was the first on the topic to be presented at a dedicated workshop and provided an opportunity for the FPGA security community to tackle pedagogical issues.

The first problem facing course designers is the rapidity of developments in the field of FPGA and hardware security. Attacks are continuously evolving along with countermeasures and it is consequently difficult to establish a curriculum, which, in any case, will have to evolve over time.

The second problem is the wide range of knowledge required to design a secure embedded system. Fig. 1 shows the design levels of a typical embedded system (from physical to software layers) [6]. Many different threats can be operated at each level (the list in Fig. 1 is not exhaustive). Many

countermeasures are also possible at each design level, making it difficult for students to master all the levels at which an attack can occur.

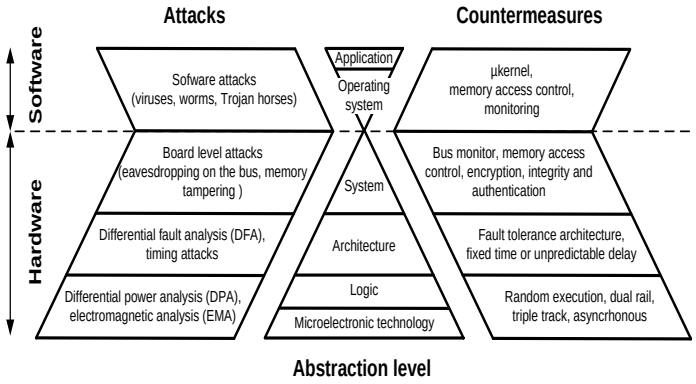

Fig. 1. Embedded system security design pyramid: from the physical to the software level.

Unfortunately, without a good knowledge of the pyramid in Fig. 1, there is a risk of overlooking potential security weakness at one or another level, or even of creating a new security weakness while trying to secure a system.

Like other design requirements (e.g. power consumption, area, speed and cost) the FPGA security course cannot take too many hours, and consequently has to be effective despite the limited time budget.

To understand how to secure an electronic system, the students first need to understand the attacks. The best way to do this has been shown to be in lab sessions. Unfortunately, there are two major issues here. In the case of active attacks, especially fault injection attacks, expensive and complex means of attack (e.g. laser injection, electromagnetic injection) are required. Such means of attacks are difficult to use by students during lab sessions. In the case of passive attacks, side channel attacks (e.g. power consumption analysis, electromagnetic radiation analysis) the attacks are not expensive, but measurements and analysis can be very time consuming (i.e., take one to several days), which does not fit the time slots generally allocated for lab sessions (usually a few hours). This makes it extremely difficult to operate physical attacks during a lab session.

Finally, assuming security is a new constraint that needs to be taken into account during the early stages of the development of an embedded system, security has to be taken into account at the same time as all the usual constraints (power consumption, size, etc.). Design choices can only be made by finding a compromise between certain performance aspects and the level of security required by the application. Security always comes at a cost, so a precise evaluation is required before choosing the best compromise between security and performance. Even so, it is impossible to give students a security "cookbook", as each new system involves particular security issues.

In the following section, we present the course used at Saint-Etienne Institute of Telecom that was designed to solve these pedagogical issues.

III. Overview of FPGA security Course

A. Main prerequisites

The FPGA security of electronic systems is a very complex problem, as can be seen in Fig. 1. As a considerable amount of knowledge is a prerequisite for this course, in our case, it is only available to fifth-year graduate students in electrical engineering. Students need to have previously acquired the necessary knowledge in information theory, digital signal processing, instrumentation and measurement, digital system design, computer science, analog system design and more. Below is a list the main foundation knowledge that is required in these fields:

- Information theory: probabilities, Shannon entropy, relative entropy, conditional entropy, mutual information, and practical use of the Matlab tool.
- Digital signal processing: spectral analysis, error correcting codes.
- Instrumentation and measurement: use of spectral analyzers, oscilloscopes, measurement of physical information such as power consumption and electromagnetic emanation.
- Digital system design: VLSI design, VHDL (or Verilog), FPGA design (SRAM and FLASH based), co-design, reconfigurable architecture design.
- Computer science: computer architecture, microprocessor architecture and programming, embedded system design.
- Analog system design: analog filter design, feedback loop system, CMOS characterization, analysis of transistor physical noise.
- Optional: an introduction to fuzzy logic could help understand certain security approaches, beginner knowledge in laser techniques.

B. Course design

Since FPGA security is a new field, there is no 'generally accepted' list of topics to cover. This section describes the security course available to graduate students at Saint-Etienne Institute of Telecom. It comprises fourteen 90-minute lectures, one 9-hour lab, one 3-hour lab and an optional 30-hour mini-project. Without the mini-project, the course represents a workload of about 70 hours with around 50% of self study. Table I lists the lectures and lab topics covered by the FPGA security course.

The following section provides more detailed information about the fourteen 90-minutes lectures.

Student learning outcomes are evaluated both by practical assessment in the labs (in the form of technical lab reports written by the students) and by a theoretical written exam comprising a multiple-choice quiz, an oral exam (focused on FPGA implementation of public key encryption algorithm) and a final written exam at the end of the series of lectures.

TABLE I
LECTURES, LABS TOPICS AND DURATION OF PROPOSED FPGA SECURITY COURSE SEQUENCE

Type	Topics summary	Duration
Lecture #1	Introduction to security issues of digital data and embedded systems.	90 minutes
Lecture #2-3	Description of cryptographic services (confidentiality, integrity, authentication and non-repudiation). Presentation of modern symmetric encryption algorithm (e.g. AES) and hash functions (e.g. SHA).	2*90 minutes
Lecture #4-6	Public key cryptography, FPGA implementation of RSA.	3*90 minutes
Lab #1	AES hardware implementation targeting Altera FPGA.	6*90 minutes
Lecture #7-9	True random number generators (TRNG), jitter study, TRNG evaluation and testing, statistic testing, encryption key generation, FPGA implementation.	3*90 minutes
Lecture #10-11	Physical unclonable function (PUF), PUF FPGA implementation and characterization, war against counterfeiting, active and passive IC metering scheme, FPGA IP watermarking.	2*90 minutes
Lecture #12	Physical attack, side channel attacks (SCA), fault injection attacks (FIA).	90 minutes
Lab #2	Differential power analysis (DPA) targeting FPGA implementation of AES.	2*90 minutes
Lecture #13	Physical attack countermeasure, at logical level, at architectural level, at algorithmic level and system level.	90 minutes
Lecture #14	FPGA bitstream security, reconfigurable architecture for security.	90 minutes

C. Course design

According to table I, fourteen lectures are covered in the FPGA security course sequence with incremental approach. The content of each of lecture is described below.

Lecture #1 Introduction. Lecture #1 introduces and familiarizes students with the hardware security issues. After a historical overview of the data security (history of cryptography and computer security), the course focuses on the problems of embedded system security and hardware security. Charismatic examples are presented [7]: virus targeting mobile phones [8, 9], car security [10, 11], medical appliances security [12], the *Stuxnet* virus that targeted SIEMENS motor controller [13, 14] and security of embedded systems [6, 15]. The general idea is shown how the problem of security is issued since the beginning of computer science to the rise of embedded systems and the Internet of things. Moreover, recent integrated circuit (IC) security issues such as hardware Trojan [16] and IC counterfeiting [17] are presented in order to prepared young engineers to design trusted IC [18].

Lectures #2 and #3 Cryptographic services. Lectures #2 and #3 present the cryptographic services such as data confidentiality, data integrity, data authentication and non-repudation. Symmetric encryption algorithms and hash functions are presented. A focus on advance encryption standard (AES) is necessary to used it during as a key example for hardware attacks [19-21]. For AES presentation we use animation provided by E. Zabala [21] as lecture material. Hardware implementations of AES are presented with sequential and parallel architectures. Performance comparison of AES hardware with FPGA and ASIC is also discussed [22].

Lectures #4, #5 and #6 Public Key Cryptography. Two lectures are necessary to present public key cryptography. RSA and elliptic curve cryptography (ECC) encryption algorithms are studied with FPGA implementation. Most of the time is spent on Montgomery hardware implementation and performance comparison between software implementation and FPGA implementation [23, 24].

Lectures #7, #8 and #9 TRNG. TRNGs are key element of the security of cryptographic systems. Indeed, random numbers are often used in key generation processes, authentication protocols, zeroknowledge protocols, padding, in many digital signature and encryption schemes, and even in some side channel attack countermeasures [1, 25]. These two lectures describe random number generation mechanisms and physical entropy extraction. Moreover, recent work on TRNG attack by using electromagnetic channel is also presented [26].

Lectures #10 and #11 PUF. The continuous increase in the number of publications on silicon physical unclonable functions (PUFs) highlights their scientific and practical interest. These two lectures present the few basic PUF principles: one can use the race of delays between two symmetrical delay lines (arbiter PUF [27]), frequency mismatch in multiple ring-oscillators (RO-PUF [28]), metastability of a couple of cross-coupled elements (SRAM-PUF [29] and butterfly PUF [30]), and a mixture of a chain of configurable delay lines and a ring oscillator (Loop-PUF [31]). Moreover, the lecture #11 focuses on FPGA implementation of PUF which highlight the issue of symmetrical routing of some structure [32]. Performance comparison of PUF structures with FPGA implementation is also discussed [33].

Lecture #12 SCA and FIA. Lecture #12 focuses on hardware attacks which are mainly focused on finding a secret key used by embedded cryptographic functions (e.g. decipher) for protecting data. These lectures present first side channel attacks (SCA) [34]. They use so-called side channels such as timing of computations, power consumption, device electro-magnetic emanations, sound, temperature and optical radiations, etc. for getting additional information on processed confidential data. Side channel attacks are passive attacks analyzing the behavior of the circuit during its operation by measuring its dynamic characteristics. Secondly, the lectures present fault injection attacks (FIA) [35], which belong to the group of active attacks. FIAs can complete SCAs. Indeed, it is possible to use timing and power supply channel to inject faults by using clock glitches or power supply spikes. Optical channel can be used to inject single-bit error even in fine grain modern devices. Fault injection aims to modify the behavior of the circuit or to propagate errors along the data path. In both cases,

the circuit is disturbed in order to make it deliver information that couldn't be obtained under normal operation. To end these lecture some results on SCA targeting FPGA (SRAM [36] and FLASH [37] technologies) are discussed.

Lectures #13 Countermeasures. Following the presentation of attacks, lecture #13 gives the student a view of hardware countermeasures for SCA [38] and FIA [39].

Lectures #14 FPGA security. This last lecture highlights the FPGA security issue. Indeed, configurable systems (based on FPGA) are particularly vulnerable to configuration cloning, reverse engineering, replay attack (after configuration update) and fault injection targeting configuration memory [1, 2]. The security of FPGA configuration remains an open topic. FPGA vendors propose bitstream encryption schemes for SRAM [40] and FLASH technology [41]. Such schemes are suitable for numerous industrial applications. However, security provided by current FPGA devices is not sufficient for many critical applications. Recent works highlight security flaws existing in commercial products [42, 43]. It is therefore necessary to search for solutions that will ensure security of reconfigurable systems. Since a decade, research works proposed interesting FPGA bitstream protection by using: partial and dynamic reconfiguration [44], message authentication code [45], public key encryption [46, 47], PUF [48], protection against replay attack [49] and use of trusted platform module (TPM) [50, 51]. These works are presented and discussed during this last lecture.

The following section provides more detailed information about the two lab sessions and some optional student mini-project subjects.

IV. Brief description of labs and projects

A. Lab#1: AES FPGA implantation

During this lab session, students have to design a hardware full 128-bit AES symmetric block cipher with key expansion (using the 'electronic code book' mode of operation). They are invited to target Altera Cyclone II FPGA. For this purpose, the instructor distributes standard specifications for the AES Rijndael algorithm [19]. Students have to perform a hardware description of AES cipher with VHDL [20]. Logical synthesis is performed using the Altera Quartus-II tool. Functional and timing simulations are performed using the Mentor Graphic ModelSim. For this simulation, the instructor provides a VHDL test bench file that describes input stimuli with a selected secret key and then compares simulated hardware cipher output with pre-computed AES output.

The AES algorithm is based on the four following 128-bit transformations that perform sequentially during 10 rounds [19] (note that 128-bit input and output transformation data are represented by a four-by-four matrix of bytes):

AddRoundKey: A transformation in which a 128-bit round key is added to 128-bit input using XOR operation.

SubBytes: A transformation that processes the 128-bit input using a non-linear byte substitution table (S-box) that operates on each of the input bytes independently (16 times).

ShiftRows: A transformation that processes the 128-bit input by cyclically shifting the last three rows of the input by different offset values.

MixColumns: A transformation that takes all the data in all the columns of the 128-bit input matrix and mixes them to produce new columns.

AddRoundKey and *ShiftRows* have not proved to be problematic for students during the VHDL description. To help students describe *SubBytes*, the instructor provides a dual RAM initialization file (such as a .hex format with a Quartus-II tool). This file contains the full S-box table. For the implementation view, only eight dual-port 256-byte RAMs are sufficient to implement *SubBytes*.

Understanding *MixColumns* requires assistance from the instructor and an interactive presentation of modular computing. Nevertheless, implementing *MixColumns* needs only a common logical operator such as XOR, byte left shift (to perform multiplication by 2) and NOT gates. *MixColumns* is design to perform logical operation on 32-bit AES column. Four *MixColumns* component are used to realize a full 128-bit *MixColumns*. Proposed architecture for *MixColumns* follows the Fig. 2 description. The modular multiplication *2time* is given in GF(2^8) [19, 20]. Function of most significant bit of the operand, *2time* is realized by simple byte left shift or a byte left shift and some bits inversion [19, 20].

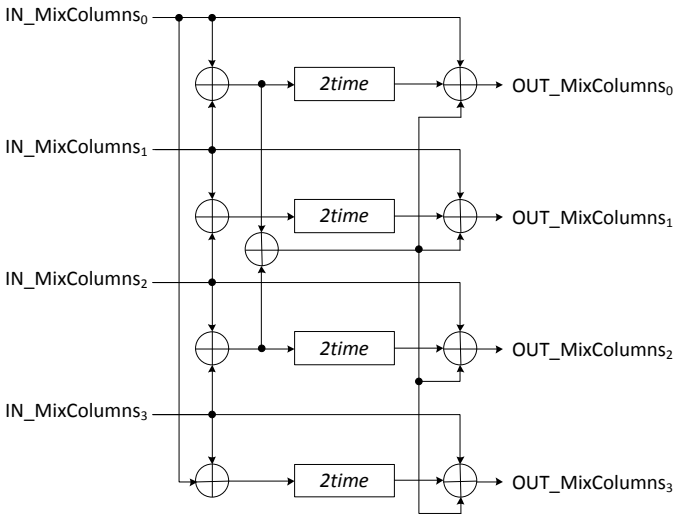

Fig. 2. Proposed architecture for 32-bit *MixColumns*.

Implementing key expansion hardware is as complex as implementing cipher hardware. Actually, key expansion uses some of the cipher transformations such as *SubBytes* and *ShiftRows*.

This lab could be shorter depending on the amount of help provided by the instructor. What is vital is that the students understand the AES algorithm by the end of the lab session. This understanding is crucial for the rest of the course especially for understanding physical attacks. In the rest of the course, only the AES algorithm is used to illustrate the attacks.

B. Lab#2: Differential power analysis (DPA) targeting FPGA implementation of AES

Side channel attacks analyze the behavior of the circuit while it is operating. The analysis can be based on its power consumption. If it is possible to link the measured power consumption and the secret key used for an encryption algorithm, for example, then the secret key can be deduced by analyzing the measured values. This is the case for the use of a symmetric cryptography algorithm such as AES embedded in a hardware circuit. The power consumption of CMOS logic gates used in this case characteristically depends on the transistor commutations and therefore on the internal signals. This property is used very effectively in the differential power analysis (DPA) attack, which was developed in 1999 [52, 53]. This attack allows the 128-bit secret key used in the AES algorithm to be discovered using minimal equipment (an oscilloscope and a computer), even if the key itself is mathematically unbreakable with computational methods.

The aim of this lab session is to enable students to improve their understanding by performing practical experiments with the DPA. However, the full DPA process is very time consuming. It has two main steps. The first consists in measuring the circuit power consumption. This circuit has to embed an AES cipher. The number of power consumption measurements can be huge, i.e., about ten thousand traces (with several hundred measurement points). As a consequence, it is impossible to carry out measurements during the lab session. For this lab session, as shown in Fig. 3, the instructor provides one thousand power consumption traces (we use Microsemi Flash-Based technology FUSION FPGA as device target [37]). Each trace is measured during a 128-bit plain text cipher, and 512 measurements with 8-bit values per trace are used as the sample. As can be seen in Fig. 3, the instructor provides a 1000-by-512 matrix of 8-bit power consumption values (P matrix). Note that the number of power consumption traces is reduced compared to the first DPA proposed [52] by using correlation analysis [54].

To attack the AES cipher, DPA uses the correlation between power consumption and secret cipher key during the two first computation steps (*AddRoundKey* and *SubBytes*). These two transformations act byte by byte. As a result, it is possible to perform the DPA to discover the secret key byte by byte. For this lab session, it is sufficient to limit the DPA in the secret key to the first byte found. According to this limitation, the instructor provides only the first byte of the thousand randomly chosen plain texts. As shown in Fig. 3, the instructor provides a 1000-by-one matrix of 8-bit plain text (T Matrix).

In this lab session, students have to model the two first AES steps (this model is called 'prediction model' in DPA) with the Matlab tool. Subsequently, they can try all 256 possible one-byte sub-keys and compute the correlation between the output of the prediction model (the H matrix) and the provided measurement of power consumption (the P matrix).

As already mentioned, understanding attacks is a crucial step in the design of efficient countermeasures. In lecture #13, which follows this lab session, several DPA countermeasures are presented which act at algorithmic, architectural, logical or physical levels (see Table I).

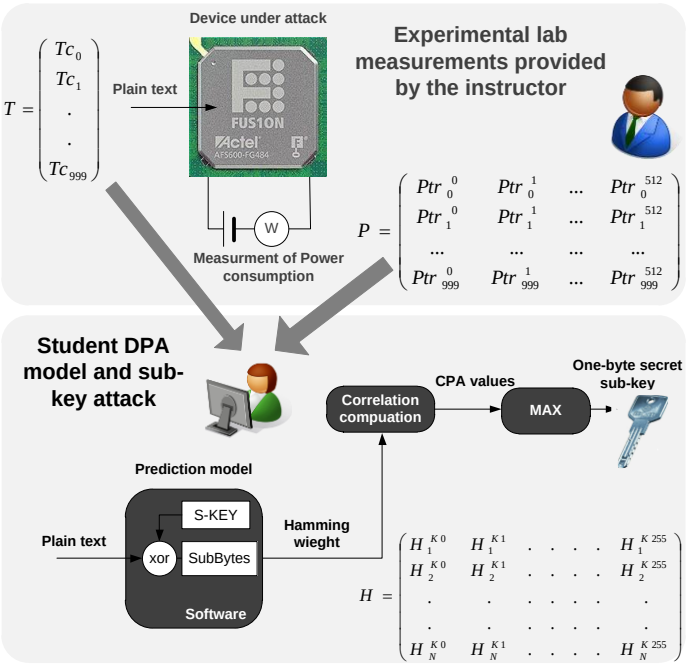

Fig. 3. The DPA lab description.

Figure 4 shows the correlation curve found by students using the Matlab program. The 'Max. correlation value' was found for the secret sub-key (43 in decimal value for this example). This curve can be post-analyzed by students to understand how the DPA works in relation to AES computation (see lab#1).

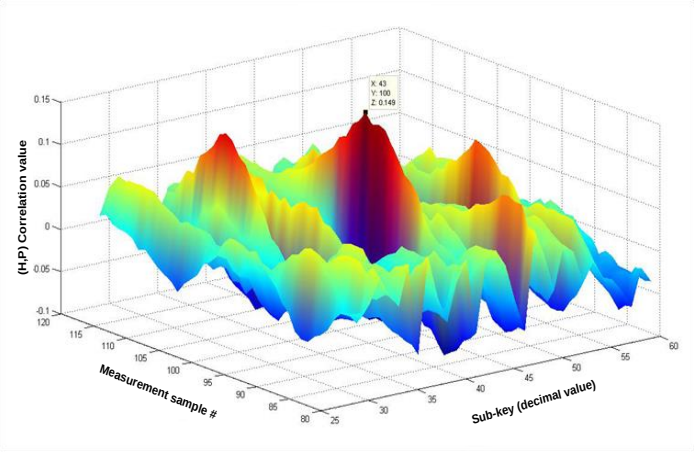

Fig. 4. Correlation results from Matlab student program with a limited range of sub-keys tested.

C. Examples of mini-projects

In section III-B of this article, we discussed an optional 30-hour mini-project. In the course at St Etienne, this project is optional in the sense that students can choose a subject that

concerns only one part of their electrical engineering curriculum. They do not always choose to focus on FPGA security. If they do, several topics are covered, including (with reference design):

- Management of SRAM FPGA bitstream security with dynamic reconfiguration [44].
- FPGA implantation of hardware symmetric or public key (part of) cipher [3].
- FPGA implantation of hardware hash function [3].
- Study and test of some DPA countermeasures (with FPGA implementation) [40].
- FPGA implantation and test of silicon PUF [2].
- FPGA implantation and test of TRNG [1].

The last two examples require the help of the instructor to perform the FPGA placement constraint in order to design functional PUF/TRNG [32].

These projects provide the students with their first work experience with a design problematic. Subsequently, they have to complete a six-month internship in a company or in an academic laboratory where they have to implement a design project. Some students (two to four per year or more, depending on the subjects offered) choose FPGA/hardware security as their internship topic.

In the following section, we examine the learning objectives of the proposed curriculum, and explore how to go about achieving them.

V. Assessment of Student FPGA Security Learning Outcomes

FPGA security is a new topic in education, and it is hard to evaluate student learning outcomes as, so far, there are no references with which to compare them. All the same, the curriculum we describe here has proved to be efficient for acquiring knowledge about FPGA security and answers some of the pedagogical issues listed in section II. As can be seen in Table II, several learning objectives concerning FPGA security were achieved. Despite the vast amount of knowledge required before beginning to learn about FPGA security, the course we describe respects the allocated time budget.

As can also be seen in Table II, students apparently did not acquire sufficient knowledge about side channel attack countermeasures. There are several possible explanations for this. Countermeasures call on a very wide range of skills (e.g. data random masking, power consumption noise generator, fault-tolerant architecture, dynamic architecture, power supply regulator and filter, dual-rail logic, tamper resistant device). This area is evolving rapidly and attacks follow the same trend. Education can provide a static picture at a given moment in time, but unfortunately it cannot ensure students will understand future evolution of attacks/countermeasures.

As can be seen in Table II, the FPGA security course enables students to understand and retain information on FPGA security concepts. Acquisition of this basic knowledge is checked in two written exams: one multiple-choice quiz and one final examination. In addition, the course provides students with opportunities for practical application and critical thinking in the lab sessions. Their ability to apply theoretical knowledge is checked in the written technical lab report. If students choose to work on FPGA security in their mini-project, the course helps develop creative thinking, which is checked by the

TABLE II
SUCCESSFULLY COMPLETED MAIN LEARNING OBJECTIVES

Objectives	Achieved	Tools / assessment of learner outcomes	Results (*, **)
Make students aware of FPGA security issues.	Fully	Interactive lecture and description of practical security threats / Multiple choice quiz	Average score in multiple choice quiz: *16.7/20, **15.4/20
Provide students with knowledge on the main algorithms used in the field of cryptography.	Fully	Interactive lecture and AES lab / Multiple choice quiz + oral exam (for public key cryptography only)	
Prepare student for the hardware design of ciphers.	Fully	AES hardware implementation lab / Written technical report.	Average grade in AES lab technical report: *15.0 /20
Understanding side channel attacks within a limited time budget.	Fully	Interactive lecture and DPA Lab with experimental measurements performed ahead of time by the instructor / Written technical report.	Average grade in DPA lab technical report: *14.8/20
Provide students with knowledge to understand the main countermeasures to side channel attacks.	Partially	Post DPA lab lecture / final written exam	Average score in final written exam: *13.8/20
Motivate students to work in the field of FPGA/hardware security	Partially	A lot of practical examples and the mini-project if selected by students / motivating students to choose the topic for their internship	Number of students concerned: *2 out of a total of 20 (10%)

* Results are for the academic year 2012-2013 at the Saint-Etienne Institute of Telecom. The number of master students who took this course was 20.

** Results of multiple choice quiz are for the academic year 2012-2013 at the Bordeaux Institute of Technology, the number of master students who took this course was 43.

written and oral presentation of the results of their mini-project.

FPGA security is not the main course of study at Saint-Etienne Institute of Telecom, this is the third year this course has been piloted at this institute, consequently, up to now, only a few students have chosen the topic (during their engineering degree or for their PhD.). Nevertheless, student satisfaction was very high, a post-course survey on course content had very encouraging results and student feedback was very positive.

VI. CONCLUSION

Along with size, energy, and power consumption, security has become an integral part of the design space of reconfigurable systems, whether embedded or not. Introducing this graduate level course in FPGA security was motivated by the increasing importance of this topic for consumer product design. However, teaching FPGA security is not a simple didactic project, as it requires a huge range of knowledge and theoretically, a lot of time. In this paper, we describe the components of a course dedicated to FPGA security and show that it is possible to reach reasonable learning objectives with a limited time budget. To achieve these objectives, this paper describes two lab sessions: the first a lab focusing FPGA implementation of AES symmetric cipher, the second a more original lab focusing on AES DPA attacks targeting FPGA implementation.

ACKNOWLEDGMENT

The author would like to thank several colleagues at the University of Lyon, Viktor Fischer, Florent Bernard, Lubos Gaspar, Robert Fouquet, at the University of South Britanny, Guy Gogniat, and at the Bordeaux Institute of Technology, Dominique Dallet and Bertrand Le Gal for their support.

The work presented in this paper was realized in the frame of the SALWARE project number ANR-13-JS03-0003 supported by the French "Agence Nationale de la Recherche".

REFERENCES

[1] B. Badrignans, J-L. Danger, V. Fischer, G. Gogniat, L. Torres (eds.), "Security trends for FPGAs," New York, Springer, 2011.

[2] M. Tehranipoor, C. Wang, "Introduction to FPGA security and trust," New York, Springer, 2011.

[3] F. Rodriguez-Henriguez, N.A. Sagib, A. Diaz Perez, C. Kaya Koç, "Cyrptographic algorithms on reconfigurable hardware," New York, Springer-Verlag, 2nd ed., 2010.

[4] "Computer science curriculum 2008: An interim revision of CS 2011," ACM, IEEE Computer Society, 2008.

[5] L. Bossuet, G. Gogniat, "How to teach FPGA security," International Workshops on Cryptographic Architectures Embedded in Reconfigurable Devices (CryptArchi'07), 2007. On-line available : http://labh-curien.univ-st-etienne.fr/~bossuet/publication2_fichiers/CA07.pdf

[6] L. Bossuet, G. Gogniat. "Hardware security in embedded systems," In Communicating embedded systems for networks, 1st ed., F. Krief, Ed, Wiley-ISTE, 2010, ch. 5.

[7] Mocana Copr., "Attacks on Mobile and Embedded Systems – Five Important Trends" White paper, 2011.

[8] D. Dangon, "Mobile phones as computing devices: the viruses are coming!" Pervasive Computing, IEEE, vol. 3, issue 4, p. 11-15, 2004.

[9] P. Wang, M. C. Gonzáles, C. A. Hidalgo, A.L. Barabási, "Understranding the Spreading Patterns of Mobile Phone Viruses," Science, vol. 324, May 2009, p. 1071-1075, 2009.

[10] S. Checkoway, D. McCoy, b. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, T. Kohno, "Comprehensive Experimental Analyses of Automotive Attack Surfaces," In Proceedings of the 20th USENIX Conference on Security (SEC'11), p. 6-6, 2011.

[11] M. Wolf, A. Weimerskirch, T. Wollinger, "State of the Art: Embedding Security in Vehicles," EURASIP Journal of Embedded Systems, vol. 2007, 16 pages, 2007.

[12] D. Halperin, T. S. Heydt-Benjamin, B. Ransford, S. C. Clark, B. Defend, W. Morgan, "Pacemarkers and implantable Cardiac Defibrillators: Software radio Attacks and Zero-Power Defenses", In the Proceedings of the 2008 Symposium on Security and Privacy (SP'08), IEEE, p. 129-142, 2008.

[13] R. Langner, "Dissecting a Cyberwarfare Weapon," Security & Privacy, IEEE, vol. 9, no. 3, p. 49-51, 2011.

[14] T.M. Chen, S. Abu-Nimeh, "Lessons from Stuxnet," Computer, IEEE, vol. 44, no. 4, 2011.

[15] S. Ravi, A. Raghunathan, P. Kocher, S. Hattangady, "Security in Embedded Systems: Design Challenges" ACM Transaction on Embedded Computing Systems, vol. 3, no. 3, p. 461-491, 2004.

[16] R. Karri, J. Rajendran, K. Rosenfeld, M. Tehranipoor, "Trustwothy Hardware: Identifying and Classifying Hardware Trojans," Computer, vol. 43, no. 10, p. 39-46, 2010.

[17] C. Gorman, "Counterfeit Chips on the Rise," IEEE Spectrum, June 2012.

[18] L. Bossuet, D. Hely, "SALWARE: Salutary Hardware to design Trusted IC", Workshop on Trustworthy Manufacturing and Utilization of Secure Devices (TRUDEVICE'13), 2013.

[19] FIPS, "Specification for the advanced encryption standard (AES)," Federal information, processing standard publication 197, NIST, 2001.

[20] J. Daemen, V. Rijmen, "The design of Rijndael," Berlin, Springer, 2002.

[21] E. Zabala, "Rijndael Cipher – 128-bit version (data block and key) Encryption" Flash Animation, University ORT, Montevideo, Uruguay, 2003. http://www.formaestudio.com/rijndaelinspector/

[22] T. Wollinger, C. Paar, "How Secure are FPGAs in Cryptographic Applications," In Proceedings of 13th International Conference on Field-Programmable Logic and Applications (PFL'03), p. 707-711, 2003.

[23] C. McIvor, M. McLoone, J. V. McCanny, "FPGA Montgomery Multiplier Architectures – A Comparison," In Proceedings of the 12th Annual Symposium on Field-Programmable Custom Computing Machines (FCCM'04), IEEE, p. 279-282, 2004.

[24] G. Perin, D. G. Mestiqua; J. B. Martins, "Montgomery Modular Multiplication on Reconfigurable Hardware : Systolic versus Multiplexed Implementation," International Journal of Reconfigurable Computing, Hindawi Publishing Corp., vol 2011, 10 pages, 2011.

[25] V. Fischer, "A Closer Look at Security in random Number Generator," in Proceedings of Constructive Side-Channel Anaysis and Secure Design (COSADE'12), Springer, Lecture Notes in Computer Science, vol. 7275, p. 167-182, 2012.

[26] P. Bayon, L. Bossuet, A. Aubert, V. Fischer, F.Poucheret, B. Robisson and P. Maurine, "Contactless Electromagnetic Active Attack on Ring Oscillator Based True Random Number Generator", in Proceedings of Constructive Side-Channel

Anaysis and Secure Design (COSADE'12), Springer, Lecture Notes in Computer Science, vol. 7275, p. 151-166, 2012.

[27] S. Devadas, E. Suh, S. Paral, R. Sowell, T.Ziola, and V. Khandelwal, "Design and Implementation of PUF-Based "Unclonable" RFID ICs for Anti-Counterfeiting and Security Applications," In Proceedings of International Conference on RFID (RFID'08), IEEE, p.58-64, 2008.

[28] A. Maiti, J. Casarona, L. McHale and P. Schaumont, "A large scale characterization of RO-PUF," in Proceedings of International Symposium on Hardware-Oriented Security and Trust (HOST'10), IEEE, p.94-99, 2010.

[29] J. Guajardo, S.S. Kumar, G.J. Schrijen and P. Tulys, "FPGA Intrinsic PUFs and Their Use for IP Protection", in Proceedings of International Conference on Cryptographic Hardware an Embedded Systems (CHES'10), Springer, Lecture Note in Computer Science, vol. 4727, p. 63-80, 2010.

[30] S.S. Kumar, J. Guajardo, R. Maes, G.-J. Schrijen and P. Tulys, "Extended Abstract: The Butterfly PUF Protecting IP on every FPGA," in Proceedings of International Symposium on Hardware-Oriented Security and Trust (HOST'08), IEEE, p. 67-70, 2008.

[31] Z. Cherif, J.L. Danger and L. Bossuet, "An easy-to-design PUF based on a single oscillator: the loop PUF", in Proceedings of International Conference on Digital System Design (DSD'12), Euromicro, pp. 1-7, 2012.

[32] S. Morozov, A. Maiti, P. Schaumont, "An Analysis of Delay Based PUF Implementation on FPGA, " in Proceedings of the 6th International Conference on Reconfigurable Computing: Architectures, Tools and Applications (ARC'10), Springer, Lecture Note in Computer Science, vol. 5992, p. 382-387, 2010.

[33] S. Katzenbeisser, Ü. Kocabaş, V. Rožić, A.R. Sadeghi, I. Verbauwhede, C. Wachsmann. "PUFs: Myth, Fact or Busted? A Security Evaluation of Physically Unclonalble Functions Cast in Silicon" in Proceedings of International Conference on Cryptographic Hardware an Embedded Systems (CHES'12), Spinger, Lecture Note in Computer Sciences, vol. 7428, p. 283-301, 2012.

[34] P. Kocher, J. Jaffe, B. Jun, P. Rohatgi, "Introduction to differential power analysis," Journal of Cryptographic Engineering, Springer, vol. 1, p. 5-27, 2011.

[35] H. Bar-El, H. Choukri, D. Naccache, M. Tunstall, C. Whelan, "The Sorcerer's Apprentice Guide to Fault Attacks," Proceedings of the IEEE, vol. 94, no. 2, p. 370-382, 2006.

[36] S. Berna Örs, E. Oswald, B. Preneel, "Power-Analysis Attacks on an FPGA – First Experimental Results," in Proceedings of International Conference on Cryptographic Hardware an Embedded Systems (CHES'03), Spinger, Lecture Note in Computer Sciences, vol. 2779, p. 35-50, 2003.

[37] N. Kamoun, L. Bossuet, A. Gazel. "Experimental implementation of DPA attacks on AES design with flash-based FPGA technology". In the Sixth IEEE International Multi-Conference on Systems Signals and Devices (SSD'09), p. 1-4, 2009.

[38] T. Popp, E. Oswald, S. Mangard, "Power Analysis Attacks and Countermeasures," IEEE Design & test of Computers, vol. 24, no. 6, p. 535-543, 2007

[39] P. Maistri, "Countermeasures against fault attacks: The good, the bad, and the ugly," in Proceedings of 17th International On-Line Testing Symposium (IOLTS'11), IEEE, p. 134-137, 2011.

[40] S. McNeil, "Solving Today's Design Security Concerns," White Paper, Xilinx, WP365, 2012.

[41] Microsemi, "Design Security Solutions," web page, http://www.actel.com/products/solutions/security/designsecurity.aspx

[42] A. Moradi, A. Barenghi, T. Kasper, C. Paar, "On the Vulnerability of FPGA Bitsream Encryption against Power Analysis Attacks: Extracting Keys from Xilinx Virtex-II FPGAs," in Proceedings of the 18th Conference on Computer and Communication Security (CCS'11), ACM, p. 111-124, 2011.

[43] S. Skorobogatov, C. Woods, "Breakthrough Silicon Scanning Discovers Backdoor in Military Chip," in Proceedings of International Conference on Cryptographic Hardware an Embedded Systems (CHES'10), Springer, Lecture Note in Computer Science, vol. 7428, p. 23-40, 2012.

[44] L. Bossuet, G. Gogniat, W. Burleson, "Dynamically Configurable Security for SRAM FPGA Bitstreams," in International Journal of Embedded Systems, Inderscience Publishers, vol. 2, no. 1/2, p. 73-85, 2006.

[45] S. Drimer, M. G. Kuhn, "A Protocol for Secure Remote Updates of FPGA Configurations", in Proceedings of the 5th International Workshop on Reconfigurale Computing: Architectures, Tools and Applications (ARC'09), Springer, Lecture Note on Computer Science, vol. 5453, p. 50-61, 2009.

[46] J. Castillo, P Huerta, J.L. Martínez, "Secure IP downloading for SRAM FPGAs," Microprocessors and Microsystems, Elsevier, vol. 31, p. 77-86, 2007.

[47] S. Malipatlolla, S. A. Huss, "A Novel Method for Secure Intellectual Property Deployment in Embedded Systems," in Proceedings of the 7th Southern Conference on Programmable Logic (SPL'11), p. 203-208, 2011.

[48] Y. Hori, Y. Katashita, A. Satoh, "Tackling the Security Issues of FPGA Partial Reconfiguration with Physical Unclonable Functions," in Proceedings of Engineering of Reconfigurable Systems and Algorithms (ERSA'12), p. 79-90, 2012.

[49] F. Devic, L. Torres, B. Badrignans, "Secure Protocol Implementation for Remote Bitstream Update Preventing Replay Attacks on FPGAs" in Proceedings of International Conference on Field Programmable Logic and Applications (PFL'10), p. 179-182, 2010.

[50] T. Eisenbarth, T. Güneysu, C; Paar, A.R. Sadeghi, D. Schellekens, M. Wolf, "Reconfigurable Trusted Computing in Hardware," in Proceedings of the Workshop on Scalable Trusted Computing (STC'07), ACM, p. 15-20, 2007.

[51] B. Glas, A. Klimm, O. Sander, K. Muller-Glaser, J. Beker, "A System Architecture for Reconfigurable Trusted Platforms" in Proceedings of the International Conference on Design, Automation and Test in Europe (DATE'08), pp. 541-544, 2008.

[52] P. Kocher, J. Jaffe, B. Jun, "Differential Power Analysis", in Wiener M. (Ed.), Proceedings of the 19th Annual International Cryptology Conference (CRYPTO'99), Springer, Lecture Note on Computer Science, vol. 1666, p. 388-397, 1999.

[53] S. Mangard, E. Oswald, T. Popp, "Power analysis attacks: revealing the secrets of smart cards," New York, Springer-Verlag, 2nd ed., 2011

[54] E. Brier, C. Clavier, F. Olivier, "Correlation power analysis with a leakage model", in Proceedings of International Conference of Cryptographic Hardware and Embedded Systems (CHES'04), Springer, Lecture Notes in Computer Science, vol. 3156, p. 135-152, 2004.