STSM Project Report: Nele Mentens

ICT COST Action IC1204: Trustworthy Manufacturing and Utilization of Secure Devices

Visitor: Dr. Ir. Nele Mentens, KU Leuven, Leuven, nele.mentens@kuleuven.be

Host institution: Prof. Dr.-Ing. Tim Güneysu, Prof. Dr.-Ing. Christof Paar, RUB, Bochum, Germany

Visiting Period: September 30, 2013 – January 3, 2014.

Planned Research in the STSM Application:

The research of Dr. Ir. Nele Mentens focuses on FPGA security. In particular, two topics were addressed in this STSM:

- Topic 1: dynamic and partial reconfiguration of FPGAs for security purposes,
- Topic 2: design automation and design space exploration for cryptographic hardware.

The planned cooperation in the STSM application focused on these two topics.

Research performed during the STSM:

Topic 1

INTRODUCTION:

An adversary that uses power analysis attacks to retrieve secret information from a hardware device, usually correlates many power traces measured while the device is running with an unaltered secret key and random input data. Dynamic reconfiguration that randomly changes the implementation of the cipher under attack (while the functionality stays the same) makes it harder for the adversary to correlate the traces in order to come to a successful attack. The first work introducing this concept is done by Mentens et al. [1].

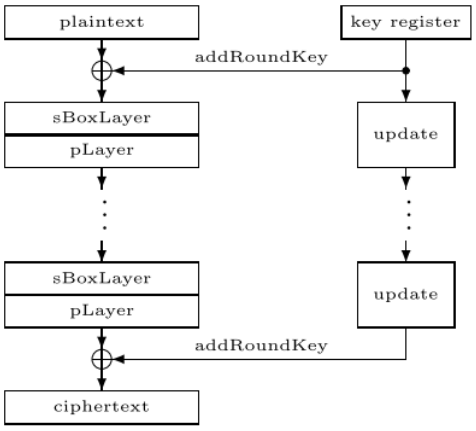
This STSM concentrated on introducing dynamic reconfiguration on many levels of hardware abstraction for several ciphers:

- The cooperation with Prof. Dr.-Ing. Tim Güneysu and Prof. Dr.-Ing. Christof Paar concentrated on reconfiguring the individual LUTs of an FPGA. The block cipher PRESENT was used as an example case [2]. PRESENT was recently included in the new international standard for lightweight cryptographic methods by ISO/IEC [3].
- The cooperation with the ESIT group at RUB (led by Prof. Dr.-Ing. Michael Hübner) addressed the reconfiguration of sub-blocks in the datapath of an elliptic curve point multiplication. The use of elliptic curves for cryptography was first introduced by Miller and Koblitz in [4,5].

CURRENT STATUS

Block cipher PRESENT - reconfiguration of the individual LUTs

Fig. 1 shows the top-level architecture of the round function of PRESENT, taken from [2]. The sBoxLayer consists of sixteen 4-bit to 4-bit S-boxes.



x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S[x]$	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

Figure 1: top-level architecture of the round function of PRESENT (left) and S-box input-output table (right) [2]

Since most side-channel attacks focus on the non-linear part of the cipher, i.e. the S-boxes, the first step in our approach consisted of making one S-box dynamically reconfigurable. For that purpose, a first architecture has been agreed upon. The architecture divides the S-box of PRESENT in different parts with a pipelining register in between. The combinatorial parts in between the pipelining registers can be changed at run-time such that the values that are stored in the pipelining registers are not correlated for consecutive measurements. In addition, the computation is also divided into different shares, which can also be changed at run-time. The architecture of the resulting S-box is shown in Fig. 2.

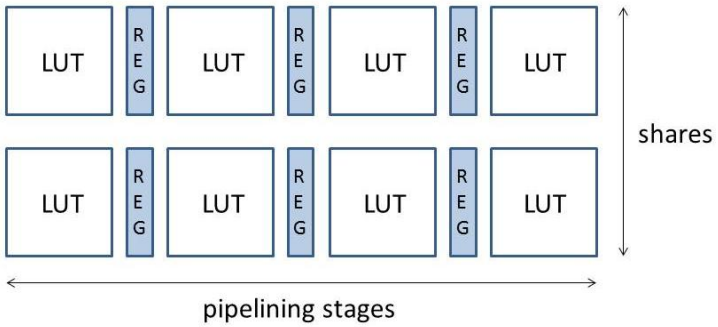


Figure 2: architecture of the dynamically reconfigurable PRESENT S-box, where LUT stands for a configurable look-up table and REG stands for a register with a configurable number of delays

In order to be able to control the dynamic reconfiguration, a control block was added to the implementation. At first instance, we used the ARM core on a ZEDboard, but the plan is to investigate the efficiency of a hardware FSM later. The control block configures the LUTs in between the pipelining registers and also the number of delays that are introduced in each pipelining register. We used Xilinx FPGAs, which made it possible to use LUTs that are configurable at run-time through a shift register (SRL) that is addressable from within the FPGA logic. The architecture also contains a True Random Number Generator (TRNG) of which the output is used to choose between different configurations. The architecture is shown in Fig. 3. At the moment, the researchers are implementing all parts of the architecture. After integration of all parts, side-channel attacks will be performed. The attacks will focus on trying to retrieve the secret key by measuring the power consumption of the S-box. Afterwards, the results of the attacks together with the results in terms of area overhead will be published.

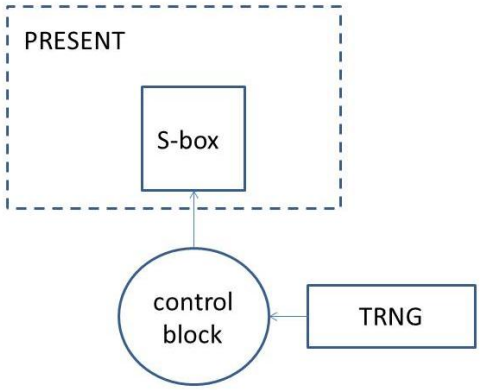


Figure 3: top-level architecture of the dynamically reconfigurable PRESENT implementation

Elliptic curve point multiplication – reconfiguration of the sub-blocks

The most important operation in elliptic curve cryptography is elliptic curve point multiplication. It is used e.g. in Elliptic Curve Diffie-Hellman (ECDH) and Elliptic Curve Digital Signature Algorithm (ECDSA). An elliptic curve point multiplication can be performed by executing consecutive point addition and point doublings. An example of these operations is depicted in Fig. 4 (point addition on the left, point doubling on the right).

Require: $P_1 = (X_1, Y_1, Z_1, aZ_1^4)$,
$P_2 = (X_2, Y_2, Z_2, aZ_2^4)$
Ensure: $P_1 + P_2$
1: $U_1 = X_1 \cdot Z_2^2$
2: $U_2 = X_2 \cdot Z_1^2$
3: $S_1 = Y_1 \cdot Z_2^3$
4: $S_2 = Y_2 \cdot Z_1^3$
5: $H = U_2 - U_1$
6: $r = S_2 - S_1$
7: $X_3 = -H^3 - 2 \cdot U_1 \cdot H^2 + r^2$
8: $Y_3 = -S_1 \cdot H^3 + r \cdot (U_1 \cdot H^2 - X_3)$
9: $Z_3 = Z_1 \cdot Z_2 \cdot H$
10: $aZ_3^4 = a \cdot Z_3^4$
11: Return $P_3 = (X_3, Y_3, Z_3, aZ_3^4)$

Require: $P_1 = (X_1, Y_1, Z_1, aZ_1^4)$
Ensure: $2P_1$
1: $S = 4 \cdot X_1 \cdot Y_1^2$
2: $U = 8 \cdot Y_1^4$
3: $M = 3 \cdot X_1^2 + aZ_1^4$
4: $T = -2 \cdot S + M^2$
5: $X_3 = T$
6: $Y_3 = M \cdot (S - T) - U$
7: $Z_3 = 2 \cdot Y_1 \cdot Z_1$
8: $aZ_3^4 = 2 \cdot U \cdot aZ_1^4$
9: Return $P_3 = (X_3, Y_3, Z_3, aZ_3^4)$

Figure 4: point addition and point doubling algorithms

In the algorithms for point addition and point doubling, modular addition/subtraction and modular multiplication are the two most important operations. Therefore, the two datapath blocks that will be used are a modular adder/subtracter and a modular multiplier. Fig. 5 shows the hierarchy of the elliptic curve operations.

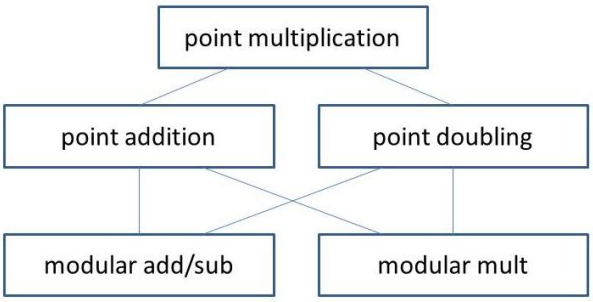


Figure 5: Hierarchy of elliptic curve operations

For point addition and point doubling, there are many algorithmic alternatives (e.g. based on the choice of projective coordinates, for which an overview is given in [6]). Further, taking into account the dependencies between the consecutive modular operations, several orders of execution can be explored. Since some operations can be performed in parallel, different architectures are also possible with a number of modular adders/subtracters and a number of modular multipliers. This results in an architecture with a configurable amount of sub-blocks with accompanying order of execution of the modular operations (see Fig. 6).

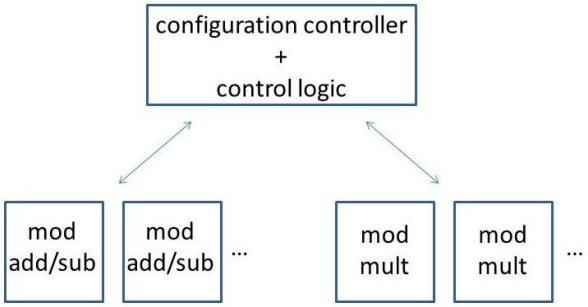


Figure 6: Architecture of the dynamically reconfigurable elliptic curve point multiplication, where the configuration controller decides how many modular adders/subtracters and how many modular multipliers are instantiated.

At the moment, the researchers are implementing the architecture. The resulting architecture should make it more difficult for an attacker to perform side-channel analysis since the architecture dynamically changes in a random way. The resulting side-channel resistance and the overhead in terms of area and speed will be evaluated and published.

FUTURE WORK

Block cipher PRESENT - reconfiguration of the individual LUTs

The next step is to go to a block cipher of which the S-box is more complicated, like AES. This will be much more difficult than the experiment we did on PRESENT, since one PRESENT S-box nicely fits into a single LUT on an FPGA (or more LUTs after introducing dynamic reconfigurability), while one AES S-box does not fit into a single LUT. Therefore, it is not straightforward to extrapolate our approach.

Elliptic curve point multiplication – reconfiguration of the sub-blocks

The next step is to do reconfigurations within one point multiplication. This will, however, only be beneficial when the overhead in reconfiguration time can be drastically reduced. Nevertheless, the novel architecture that implements this strategy can be seen as visionary taking into account that there will be a drastic technological improvement in the reconfiguration delay in the future. Another topic for future work is to unroll the point multiplication loop in order to find more options for the sequence of the operations. These new architectural approaches will naturally be evaluated in terms of side-channel resistance and area and time overhead.

Topic 2

INTRODUCTION

Cryptographic algorithms can be implemented in many different ways, where each hardware architecture has different properties regarding area, operating frequency, throughput and power consumption. In order to automate the design of cryptographic hardware taking into account constraints on these properties, the design space has to be explored after which a suitable architecture can be selected. The cooperation with Dr. Lejla Batina, who was also present during the first half of the STSM, concentrates on exploring the design space of a composite field AES S-box using genetic algorithms. The focus is on exploring all pipelining options and selecting the best possible in terms of throughput. Researchers of the university of Zagreb are also involved in this cooperation.

CURRENT STATUS

One version of a composite field S-box has been chosen. This S-box has been synthesized into a netlist. The strategy on the genetic algorithm to be used is almost finalized. The result will be a modification of the netlist in which pipelining registers have been inserted. An automatic strategy for evaluating the throughput of all versions will be established. This will involve setting up a tool chain in combination with the novel algorithms developed for the exploration. The results will be generated and published.

FUTURE WORK

A search will be done over many options for the composite field arithmetic in order to cover an even larger design space. Further, not only throughput, but also performance, low power,… will be taken into account.

REFERENCES

[1] N. Mentens, B. Gierlichs, I. Verbauwhede. Power and Fault Analysis Resistance in Hardware through Dynamic Reconfiguration. CHES 2008: 346-362.

[2] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. J. B. Robshaw, Y. Seurin, C. Vikkelsoe: PRESENT: An Ultra-Lightweight Block Cipher. CHES 2007: 450-466.

[3] Information technology -- Security techniques -- Lightweight cryptography -- Part 2: Block ciphers. ISO/IEC 29192-2:2012.

[4] V. Miller. Uses of elliptic curves in cryptography. In H. C. Williams, editor, Advances in Cryptology – Proceedings of CRYPTO, number 218 in Lecture Notes in Computer Science, pages 417–426. Springer-Verlag, 1985.

[5] N. Koblitz. Elliptic curve cryptosystem. Math. Comp., 48:203–209, 1987.

[6] H. Cohen, G. Frey, R. Avanzi, C. Doche, T. Lange, K. Nguyen, F. Vercauteren. Handbook of Elliptic and Hyperelliptic Curve Cryptography, CRC Press, 848 pages, 2005.