Experimental Comparison of Crypto-Processor Architectures for Elliptic and Hyper-Elliptic Curve Cryptography

Gabriel Gallin[2,1], Arnaud Tisserand[2,1] and Nicolas Veyrat-Charvillon[3,1]
[1]IRISA, [2]CNRS – [3]Univ. Rennes 1 – INRIA. 6 rue Kerampont, 22305 Lannion, France.

Public key cryptography is required in many applications such as key exchange, digital signature and some specific encryption schemes. *Elliptic curve cryptography* (ECC) has become the standard public-key crypto-system in many countries, where it has superseeded RSA thanks to its lower cost and much better performance. For instance, ECC uses a mere 224-bit key to replace a 2048-bit RSA key, for the same theoretical level of security. Recent research has pointed out *hyper-elliptic curve cryptography* (HECC) as a possible way to further improve public-key crypto-systems. HECC can provide the same theoretical level of security as ECC, with keys and finite field elements that are only half the size, albeit curve operations require more computation steps. In order to provide a fair comparison of ECC and HECC, one needs to implement, optimize and analyze them within the same experimental setup.

Our research group has been studying and implementing arithmetic operators, hardware accelerators and counter-measures for ECC for a long time, and has more recently taken an interest in HECC. Our goal for the near future is to evaluate the possible trade-offs between computation time, implementation costs (silicon area and energy consumption) and resilience to physical attacks (side-channel analysis in particular) for HECC.

In this work, we present a customizable crypto-processor dedicated to ECC or HECC (left part of Figure 1). The processor has been fully implemented and validated on FPGA. We are also developing dedicated programming tools (assembler, small compiler and a library of ECC/HECC cryptographic primitives). We present implementation results (on Xilinx Spartan 6 LX75) and comparisons for various configurations of the crypto-processor. Using an exploration of the number of parallel arithmetic units and of their inner sizes/performances, we are able to propose a large number of speed–area trade-offs so as to best answer the specific constraints of each application (right part of Figure 1).

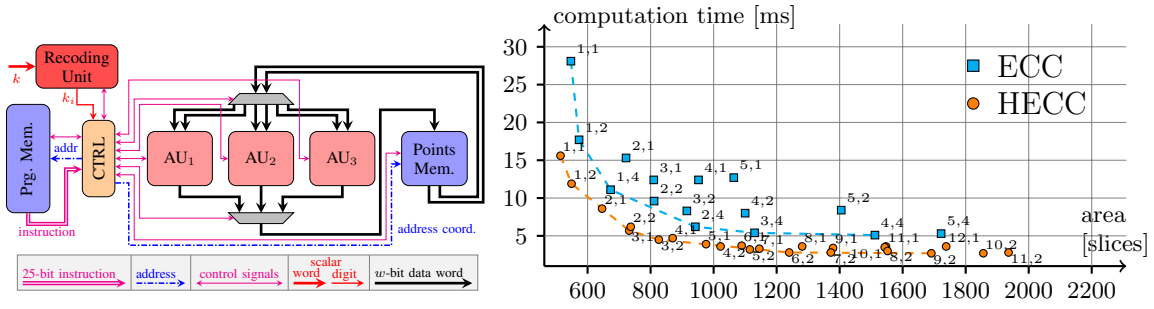

Figure 1: (H)ECC crypto-processor architecture (left) and time-area trade-offs for various configurations of its arithmetic units in the computation a scalar multiplication ECC-256 or HECC-128 bits (right).

Our experimental results tend to confirm that HECC is more efficient than ECC for the same theoretical security level, with an approximate 40 % speed improvement for equivalent silicon areas. We plan to distribute some configurations of our (H)ECC crypto-processor and the programming tools in open source in the future.

Thanks

This work is partly funded by the PAVOIS (ANR-12-BS02-002-01, `http://pavois.irisa.fr/`) and HAH (Labex CominLab and Lebesgue, `http://h-a-h.inria.fr/`) projects.