Digital Signatures Based on Elliptic Curves in RFIDs

Christoph Ruland and Tobias Lohmann

Institute for Data Communications Systems, University of Siegen, D-57076 Siegen, Germany

Summary

Radio Frequency Identification (RFID) systems can be found in wide spread applications – from simple theft prevention over multi-bit transponders up to complex applications involving contactless smartcards. This paper shows that the security gap between low-cost RFID Tags and contactless smartcards can be filled. It is examined how much power a passive tag can gain from a magnetic field and which amount of energy is needed by elliptic curve (EC) computations. The values are merged in a diagram giving the minimum timings possible to calculate and verify elliptic curve cryptography (ECC)-signatures.

Key words:

Radio Frequency Identification (RFID), Elliptic Curves, Signatures ECDSA, ECGDSA, ECMR, ECNR

1. Introduction

Modern automatic identification (Auto-ID) systems have a long technological history and multiple roots. The most widely recognized Auto-ID system is the bar code system developed during the early 1970’s [1] but the technology which is more related to the actual one is even older. During the 2nd World War, allied planes were equipped with devices that allowed a friend or foe recognition [2]. A civil variant is able to detect friends and foes inside a shop: the electronic article surveillance (EAS) system. More sophisticated systems also found their way in public life and people are using ID technology for entering a ski-lift or to disable the immobilizer of their car. In the last couple of years there has been done lot of work to map all those ″root-technologies″ to one inheritor: Radio Frequency Identification (RFID). Some of them just had to be renamed to the term RFID, others had to be reinvented like the EPC tag (Electronic Product Code) to replace EAN bar codes (Electronic Article Number) [3]. The major task in this sector is to downsize the costs of a tag, so that it is lower than the monetary benefit that the RFID-System is able to gain. This still seems to be hard because the ink which is needed for bar codes is nearly free.

Another fact is that there are rising concerns about the technology that provides information and can be read wirelessly and without any notice of its owner. People are afraid (or aware) that they can loose their privacy [4]. A lot of suggestions have been made to maintain privacy by adding extra functionality to the RFID tags but they all add more circuitry and higher costs. One basic method is to introduce a kill-command that disables a tag [5] – but the question is: who will be authorized to issue such a command? It is clear that this function has to be protected by a key or password. It must be secured. Applying even simple means against unauthorized tag access introduce the problem of key management. It is necessary to find a trade-off between the relative gain in security and the costs that come with them. When we talk about costs in this paper we do not only mean increasing chip sizes and increasing monetary costs, in the scope of this paper we especially address the increasing power consumption. 90% to 95% of the RFID devices are passive [7] which implies that they have to be powered by inductive coupling. Chapter 2 will show that increasing power consumption leads to a lowered maximum read range.

Developers of smartcards already had to face and solve most of the questions and problems that occur when adding security functions in embedded systems in the last decade. Smartcards have become very powerful and are able to process various symmetric cryptographic protocols such as 3DES, AES and strong asymmetric computations by RSA and on Elliptic Curves (ECC) [6]. They are designed to fulfill high demanding security requirements and are evaluated up to Common Criteria EAL5. Most RFID tags also need electronic circuitry inside. Therefore a tag can be seen as the same embedded system with wireless interface. It was just a logic step to add the wireless RF interface to existing smartcard controllers. The result is a very secure RFID tag with state of the art cryptography. But the resulting device will also be only able to operate close to a reader and the monetary cost for a smartcard is 20 times higher than for a simple tag.

This research was driven by the fact that the authors could not find products offering standardized asymmetric cryptography and the full functionality according to ISO15693 ″Identification cards – Contactless integrated circuit(s) cards - Vicinity cards″ that operate at distances up to a meter.

Manuscript received January 5, 2007.
Manuscript revised January 25, 2007

2. Transmittable Power

Passive RFID tags gain their energy form the alternating magnetic field that is radiated by the antenna of the reader. This chapter will present results for the maximum power that can be used by the logic of an RFID tag.
Inductive coupling is only possible in the so called ″near field″, whose dimension is mainly conditioned by the used frequency [9][10]. A good approximation of the maximum distance is determined by equation (1).

$$d = \frac{\lambda}{2\pi}. \qquad (1)$$

The interested reader can find more details in [2][15].

RFID Systems according to ISO14443 or ISO15693 operate at a frequency of 13.56MHz [8] which leads to a maximum operational radius of 3.5 m.
The maximum strength of the magnetic field an RFID-reader is allowed to emit is limited to 7.5 A/m. This value marks an upper constraint under which all RFID-Systems have to operate.
The power-relation between the reader and the tag can basically be seen as a transformer with a big gap between primary and secondary side. This implies that the well known electronic equations can be used.
The way in which the magnetic field behaves in order by the distance of its origin is highly dependent on the size of its emitting antenna. If the current and the number of windings is kept constant, small antennas produce a high initial field strength, that starts to decline very closely. A large antenna has a relative small initial field, but it will stay constant for a longer distance.

The optimum diameter of a reader's antenna is found at $\sqrt{2}$ -times of the designated reading range. It is then possible to adjust the current and the windings of the reader's coil to match the upper strength of 7.5 A/m.
In order to achieve realistic values, the diameter of the supplying antenna was set to 1 m and the antenna of the RFID-tag was chosen to have a radius of 2.5 cm in order to fit inside a sticker or card. The inductive coupled system was simulated with MATLAB and the relation between an ohmic load and the induced voltage was shown. Vice versa, it was possible to derive the maximum load (minimum ohmic resistance) R_{min} that can be applied, when a fixed voltage has to be preserved. In the following context, the behavior of three different CMOS-technologies with supply-voltages V_L of 3.3 V, 2.5 V and 1.8 V will be examined. This leads us to the following three curves presented in figure 1. They show the corresponding maximum power $P_{max} = V_L/R_{min}$ in dependence on the distance between the RFID-reader and the RFID-tag. The curves therefore define the upper bound of the power that can be consumed if the tag operates at a given distance. The complete derivation of the curves is found in [15].

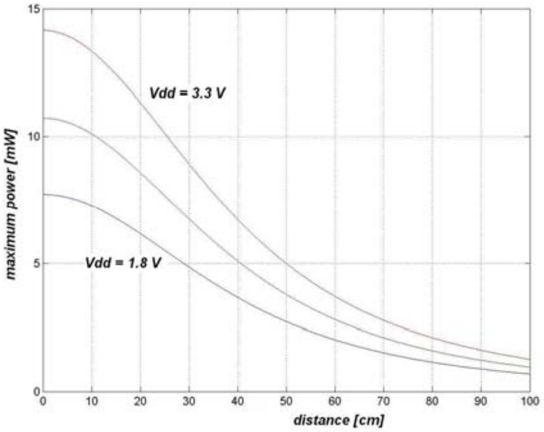

Fig. 1 Available Tag Power in dependence of range

3. Energetic consumption of digital signature schemes

The circuitry of most RFID tags is based on CMOS (complementary metal oxide semiconductor) technology. CMOS technology has the great advantage that it is possible to design electronic circuits with only relevant power consumption when the transistors change their operational state.

In order to estimate the energy needed for calculating a signature, the digital signature schemes ECDSA, ECGDSA, ECMR and ECNR [20][21] are traced back to their underlying operations in the finite field and the integrated circuits needed for executing those operations:

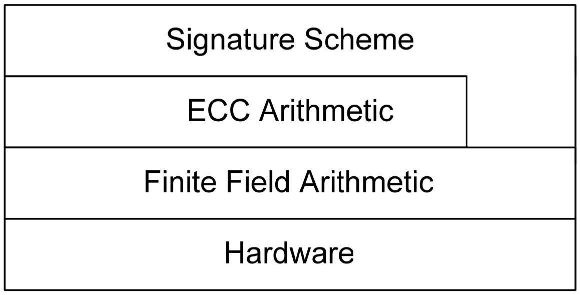

Fig. 2 Hierarchical composition of arithmetic execution layers

3.1 Arithmetic in the finite field

The "layer" of the finite field arithmetic will be executed on a dedicated hardware. It is designed according to the operand length of the field elements and it is supposed that the size of the field stays fixed during the life-cycle of the RFID tag. This work is focused on realizations that are based on Galois fields defined by primes or/and extension fields of characteristic 2. The finite field operations supported by this layer are listed in the following figure:

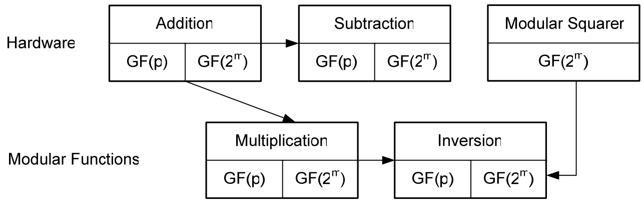

Fig. 3 Diversification of finite field arithmetic

The functionality can be divided in hardware-based and function-based operations. Addition, subtraction and – just in case of GF(2^m) – also squaring. The other operations are performed as an algorithm-controlled sequence of the mentioned hardware functions. The related dependencies are shown by the solid arrows in figure 3.

Addition

In GF(p), the underlying adder has to support integer operations with carry propagation e.g. a carry-ripple-, carry save or a von Neumann-adder, which was chosen because it offers the best trade-off between area and latency. If the elements are represented in their binary complement, the hardware doesn't have to distinguish between an addition and a subtraction. Since the maximum result is 2p-2 and the resulting element has to be < p, it might be necessary to reduce the result by the modulus.

In case of GF(2^m), there is even no logical difference between an addition and subtraction and the result is again an element of GF(2^m). The 2nd advantage is that the operation is performed by a simple XOR of the binary coefficients. The energy focused comparison of both arithmetic units shows that an addition in GF(2^m) is about eleven times cheaper then in GF(p).

Modular squaring

Modular squaring in GF(2^m) can be done by a specialized squaring unit, unique for every generating polynomial of a finite field. The square of any element is built by interleaving zeros in its binary representation:

$$a(x)^2 = \sum_{i=0}^{m-1} a_i x^{2i} = a_{m-1}x^{2m-2} + a_{m-2}x^{2m-4} + \ldots + a_2 x^4 + a_1 x^2 + a_0 \qquad (2)$$

Since the size of the resulting element is at most 2m-2 bits, it can be reduced in an inexpensive way due to the fact that the hamming weight of the used reduction polynomials is low (three or five). Secondly, no reduction is required for half of the higher order bits because they are always zero. The squarer can therefore be implemented as a hard wired XOR circuit as shown in the following example for GF(2^4) with *$f(x) = x^4 + x + 1$:*

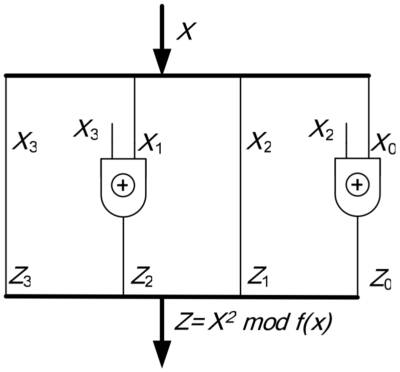

Fig. 4 Modular Squarer for GF(2^4)

The complexity of the modular squarer is only related on the size of the finite field and the hamming-weight of the generating polynominal. Table 1 shows the resulting number of XOR gates and their related energy consumption, needed to build a modular square in the examined fields GF(2^{113}), GF(2^{163}) and GF(2^{193}):

CMOS-Technology	GF(2^{113}) 56 XOR Energy	GF(2^{163}) 246 XOR Energy	GF(2^{193}) 96 XOR Energy
0.35 µm	42,47 pWs	186,58 pWs	72,81 pWs
0.25 µm	30,60 pWs	134,41 pWs	52,45 pWs
0.18 µm	8,57 pWs	37,66 pWs	14,69 pWs

Table 1: Hardware and energetic complexity of GF(2^m) squares

Modular Multiplication

The authors analyzed the two different schemes, known as Montgomery-multiplication (MM) and interleaved-modular-multiplication (IMM). Both algorithms are iterative multipliers that reduce the intermediate results in each round and thus keep them smaller than 3p-3.

The IMM is a binary iterative MSB-first multiplier that doubles the result Z=X·Y mod M in each calculation step and additionally adds the value of Y < p if the actual bit of the factor X is set to "1". The reduction is done with at most 2 subtractions per iteration.

The Montgomery-multiplier performs the same operation Z=X·Y mod M starting with the LSB of X. It does not compute Z=X·Y mod M directly, but X·Y·R^{-1} mod M where R^{-1} is a special fixed element of the finite field. Usually, R is chosen to be $2^{\lceil \mathrm{ld}(p) \rceil}$. Calculations are therefore not done in the finite field itself, but in a mirrored Montgomery-domain$^{(R)}$. The transformation of X to $X^{(R)}$ is performed by one MM of X·R^2. The multiplication is done as follows: If the actual bit of $X^{(R)}$ is set to "1", $Y^{(R)}$ is added to $Z^{(R)}$. If the result is odd, the algorithm additionally adds M to $Z^{(R)}$. Since all M are even and $X^{(R)}$+M mod M = $X^{(R)}$ mod M, $Z^{(R)}$ will be also be even and the modular result is not altered. $Z^{(R)}$ is now dividable by 2, which is done by a very simple and energy-efficient right shift. When the iterative multiplication is completed, the final result will always be smaller than *2p* and can be corrected with one subtraction. The result is still a representation in the Montgomery-domain and has to be transformed with another MM of $Z^{(R)}\cdot 1$. The advantage of calculating modular multiplications in the montgomery-domain relies on the fact, that a reduction done by shifting the operand is a cheaper operation than performing a subtraction. But this fact only counts for calculations in GF(p), since the energy-consumption and delay of the underlying adder is high. In GF(2^m), subtractions are cheap enough and there is no advantage using the montgomery multiplication. Furthermore, the transformation and retransformation can be saved.

Modular Inversion

Finding the inverse of an element a ∈ GF(q) ($a\cdot a^{-1}$ mod q = 1) is the most expensive operation inside the finite field arithmetic. The most popular methods are the extended Euclidian algorithm and the inversion by Fermat's little theorem. Another, and very effective, method for GF(2^m) is the scheme by Itoh and Tsuji [22]. It is based on Fermat's theorem but drastically reduces the number of multiplications which are needed for calculating the inverse from (m-1) to $\lfloor ld(m-1) \rfloor$+Hw(m-1)-1. Hw denotes the hamming weight of the scalar in its binary representation. The number of needed squaring does not change significantly (m to m-1), but they are nearly free when utilizing the hardware squarer. Using the Itoh-Tsuji scheme for inversions in GF(2^m) will save up to 90-95 % of time and energy.

3.2 EC Arithmetic

Elliptic curves (EC) can be defined over prime or extension fields. Based upon the results of the latter subchapter, one can see that binary extension fields are the most suitable choice for hardware implementation. The elliptic curve over GF(2^m) in its affine representation exists of the set of solutions (points) that satisfy the following cubic equation (3):

$$E: \quad y^2 + xy = x^3 + ax^2 + b . \tag{3}$$

The shown equation for elliptic curves and all other formulas, needed for the point-arithmetic (addition and doubling) can be adapted to other coordinates like into general projective, Jacobian-projective or Lopez-Dahab-projective representation. All ofthem offer the advantage that it is possible to avoid the computation of field-inverses under most circumstances. The points of the elliptic curve and a special point in infinity define an abelian group that allows cyclic (finite-field) EC-point based computations. Elliptic curve cryptography (ECC) is based on the finite set of EC-Points and the fact that it is easy to perform a scalar multiplication R=k•P, defined by the addition chain R = P+P+P+…+P, but hard to obtain the scalar k when only the Points P and R are present. This is known as the discrete logarithm problem for elliptic curves (ECDLP).

There are different methods to calculate a scalar multiplication. The simplest variant is the "double and add" algorithm that performs a point doubling in each step of the calculation and additionally a point addition if the corresponding coefficient of the binary representation of k is "1". The drawbacks of this method are that this algorithm needs to calculate a field inverse in each iteration step and that an attacker may obtain knowledge about the secret k when analyzing the runtime-behavior of the algorithm, as calculating 2P+P takes longer than calculating 2P. This attack is known as the simple power analysis (SPA).

A more sophisticated method is the scalar Montgomery Multiplication proposed by Lopez and Dahab [18]. It uses mixed coordinates and is able to calculate the scalar multiplication by only using the y-coordinate. It is therefore possible to save most of the power consuming inversions. Additionally, point doubles and additions are performed independently on the scalar factor k. This makes the algorithm resistant against power and timing attacks. Table 3 summarizes the computational costs of the three scalar multiplications, where the hamming weight of k is supposed to be m/2 where m is the length of the factor in its binary representation.

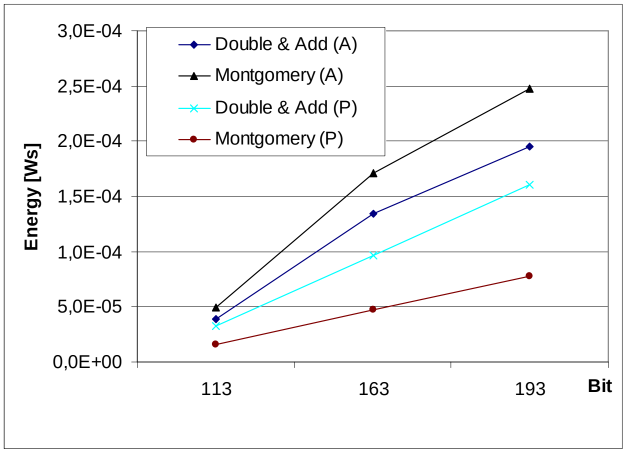

Fig. 5 Energetic comparison of projective scalar point multiplications

All digital ECC signature schemes have in common that they use a secret scalar *d* (private key) for signing a message token (e.g. a Hash) while the point d•P or d^{-1}•P (public key) is used for verification. If the message was altered or the public key doesn't correspond to the signing key, the verification will fail. There are two classes of signature schemes: signatures with appendix (e.g. ECDSA, ECGDSA) and signatures giving message recovery (e.g. ECMR, ECNR). They offer the possibility to transmit a small message within the signature. If a message is longer than the capacity to recover the message, the rest of the message is treated like it is done by signature schemes with appendix. The mentioned representatives of each class were analyzed with regard to their energy consumption. All of them have in common that a signature generation involves one scalar point multiplication of an EC point while the verification step takes two scalar multiplications. Simply spoken, the schemes only differ in the way in which the scalars are computed and processed. To visualize this observation, the generation of a signature with appendix (eg. ECDSA) is compared to a scheme with message recovery (e.g. ECNR):

```
$Input:$   $domain\ parameters\ of\ the\ elliptic\ curve$
           $private\ key\ d$
           $hashfunction\ h \quad message\ m$
$Output: signature\ (r, s)$
1. $k = rand(1, n-1]$
2. $R = k \bullet P \quad r = R_x \bmod n$
3. $s = k^{-1}(h(m) + d \cdot r) \bmod n$
4. $if\ r \vee s = 0 \quad goto\ 1$
5. $return\ (r, s)$
```

Alg. 1 ECDSA (signing)

The generation of a ECNR signature starts similar to ECDSA but in step 2, R_x - the "whitness" (r)- is modified by the recoverable message. The 2nd part of the ECNR signature (s) also computed in a different way.

```
$Input:$   $domain\ parameters\ of\ the\ elliptic\ curve$
           $private\ key\ d$
           $message\ with\ redundancy\ M$
$Output: signature\ (r, s)$
1. $k = rand(1, n-1)$
2. $R = k \bullet P \quad \Pi = R_x$
3. $r = (M + \Pi) \bmod n$
4. $s = (k - dr) \bmod n$
5. $return\ (r, s)$
```

Alg. 2 ECNR (signing)

One can see that ECDSA needs invert *k* in order to calculate *s*, while ECNR doesn't. The energetic impact of this small difference is shown in the following figure 6.

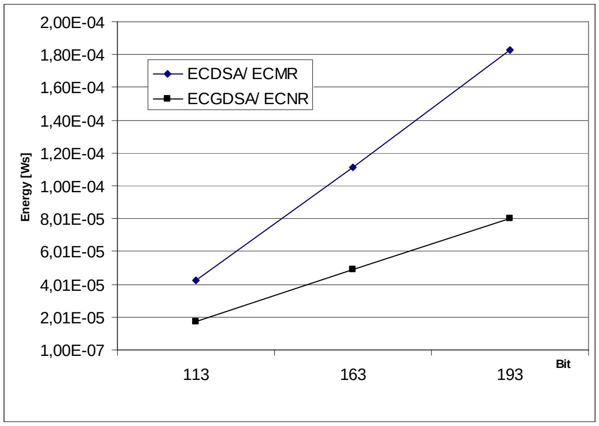

Fig. 6 Energetic comparison of signature schemes (signing)

On one hand, Figure 6 shows the energetic influence when the signature scheme involves the computation a field inverse in GF(p). Inversion free methods like ECGDSA or ECNR are so able to sign a message with 40% less energy. On the other hand, it shows that the influence of choosing a signature scheme with or without message recovery is negligible.
For verification, ECDSA, ECMR and ECGDSA have to compute field inverses modulo the order of the base point and so they do not differ in their energetic behavior. ECNR is the only scheme that offers an inversion free verification.

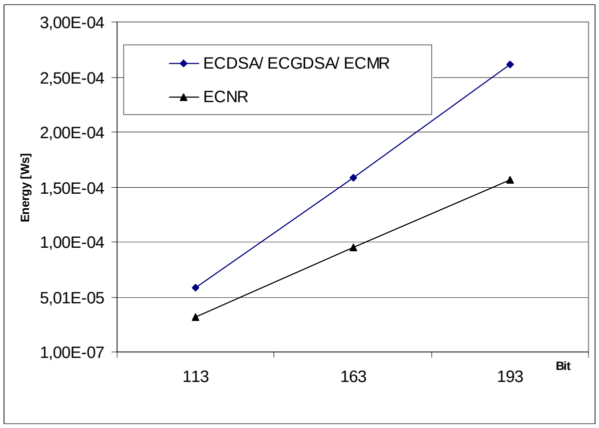

Fig. 7 Energetic comparison of signature schemes (verification)

Since it also offers the possibilities of message recovery ECNR is supposed to be the most recommended scheme.

4. Analysis

In order to determine if und under which conditions the cryptographic algorithms can be implemented in RFID tags, this paper takes an approach that implies that the limiting factor is the straitened power transfer between an RFID reader and the tag. The information which was obtained in the latter chapters can be used to define boundary conditions under which EC based cryptography is possible. The minimum calculation time is derived by dividing the energy of a signature process by the power that is available at a certain distance.

$$T = \frac{W_{signing\,/\,verification}}{P(d)} . \qquad (4)$$

The energy needed for signing and verifying a signature of length n can be normalized by $1/\lceil ld(n) \rceil^3$ and the following diagram provides the results that are independent of the bit length.

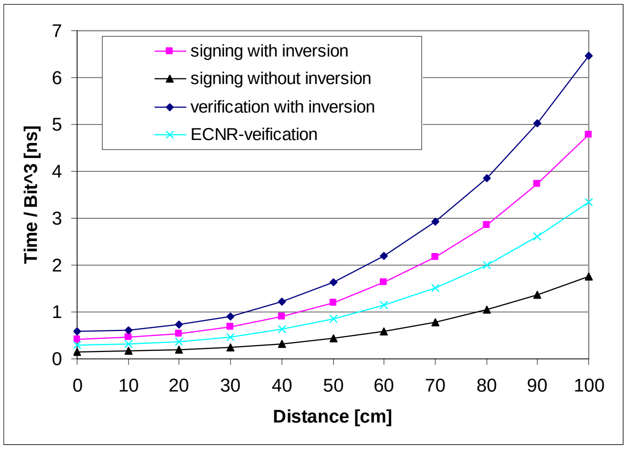

Fig. 8 Normalized minimum execution times in dependence of the range for 0.35 µm CMOS

5. Conclusions

The amount of power available for a tag at a certain distance was given in chapter two. By reducing the standardized signature schemes to finite field arithmetic and basic logic functions, it was possible to derive the energy needed by the different functional steps in chapter 3. Within this step, the energy consumption could be minimized by choosing the (energetically) best algorithms available. The results of chapters 2 and 3 were merged into minimum possible timings achievable for different signature schemes and steps. It was shown that ECNR-signatures are the best option for an RFID system because it offers the lowest need of energy and additionally provides the possibility of message recovery.

The authors showed that strong asymmetric cryptography is even possible with a relative coarse semiconductor process of 0.35 µm. Nevertheless, RFID Tags also have to contain other circuitries that handle radio access (anti-collision) and other functions. Those were not taken into account. Furthermore, the logic that has to control the cryptographic unit also will need space and energy – so does the memory that will be needed by the algorithms.
Since the scalar multiplication (next to the GF(p) inversion) is the most time and energy consuming operation in EC-based digital signature algorithms, it should be possible to expand the total tag running time (to clock down the logic) to a level where the functionality is guaranteed. Furthermore, the semiconductor technology is also still under rapid development and the authors predict that the capabilities of RFID tags will increase in the same way. If the market for RFID providing public key cryptography is big enough it should be possible to fill the mentioned security gap between AutoID tags and Smartcards.

Acknowledgment

The work presented in chapter 2 was sponsored by Siemens Corporate Technology, Munich.

References

[1] S. E. Sarma, S. A. Weis, D. W. Engels. RFID Systems and Security and Privacy Implications. Cryptographic Hardware and Embedded Systems – CHES, August 2002.
[2] K. Fong. RFID Security, http://www.cs.siu.edu/~kfong/research/RFID.ppt
[3] MIT Auto-ID Center. http://www.autoidcenter.org
[4] CASPIAN. http://www.nocards.org
[5] Auto-id Center. Draft protocol specification for a 900 MHz class 0 Radio Frequency Identification Tag, 23 Feb 2003.
[6] Infineon technologies. SLE 66CLX641P Short Product Information, April 2004.
[7] K. Finkenzeller. RFID-Handbuch, Hanser Verlag 2002.
[8] ISO/IEC 14443. Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 2: Radio frequency power and signal interface, July 2001.
[9] G. Lehner, G. Elektomagnetische Feldtheorie für Ingenieure und Physiker, Springer Verlag, 1990
[10] W.R. Smythe. Static and Dynamic Electricity, McGraw-Hill Book Company, 1968
[11] MOSIS, www.mosis.org
[12] ASICSws, www.asics.ws
[13] J. Krasner. Using Elliptic Curve Cryptography (ECC) for Enhanced Embedded Security, November 2004.
[14] R. J. Baker, H. W. Li, D. E. Boyce. CMOS Circuit Design, Layout, And Simulation. IEEE Press 1998.
[15] T. Lohmann, M. Schneider, Ch. Ruland. Analysis of power constraints of cryptographic algorithms in mid-cost RFID Tags, In Smart Card Research and Advanced Applications – CARDIS 2006, volume 3928 of Lecture Notes of Computer Science, pages 278 – 288. Springer Verlag 2006.

[16] P. L. Montgomery, Modular multiplication without trial division, In Mathematics of Computation, volume 44, pages 519 – 521.

[17] G. R. Blakley, A computer algorithm for the product AB modulo M, IEEE Transactions on Computers, volume 43, pages 290 – 292, 1983

[18] J. Lopez, R. Dahab. Fast Multiplication on elliptic curves over GF(2m) without precomputation. In Cryptographic Hardware and Embedded Systems – CHES'99, volume 1717 of Lecture Notes of Computer Science, pages 316 – 327. Springer Verlag 1999.

[19] Certicom. SEC 2:Recommend Elliptic Curve Domain Parameters. Standards for Efficient Cryptography, 2000

[20] ISO/IEC FDIS 15946-2. Information technology – Security techniques – Cryptographic techniques based on elliptic curves. Digital signatures

[21] ISO/IEC FDIS 15946-4. Information technology – Security techniques – Cryptographic techniques based on elliptic curves. Digital signatures giving message recovery

[22] T. Itoh, S. Tsuji. A fast algorithm for computing multiplicative inverses in GF(2m) using normal bases. In Information and Computation, volume 78, pages 171-177, 1998.

Christoph Ruland, Professor, Dr., born 1949 in Hamburg, Germany, studied mathematics, physics and computer science at the University of Bonn. He received a diploma in mathematics as well as doctor degree. Applied Sciences in Aachen in 1982, and a full professor with the University of Siegen in 1992. He is the Director of the Institute for Data Communications Systems of the University of Siegen.

His main research area is the integration of security into communication systems on all layers. He has written books and many publications about information security in networks and is an active member in the ISO "Security Techniques" committee for 15 years.Professor Ruland founded the "Company for Cryptographic Communication Security and Communication Technology" (KryptoKom) in 1988.

Tobias Lohmann, born 1977 in Siegen, Germany studied electronic engineering at the University of Siegen where he received his Dipl.-Ing. degree in 2002. Since then he is working as a research assistant in the Dept. of Electrical Engineering, the institute for Data Communication Systems, University of Siegen. His research interests includes RFID systems, wireless networks, cryptographic protocols and wave propagation.