# Public-Key Cryptography for RFID Tags and Applications

**Chapter** · January 2009

DOI: 10.1007/978-0-387-76481-8_13

Lejla Batina
Radboud University
**177** PUBLICATIONS **4,377** CITATIONS
SEE PROFILE

Jorge Guajardo
Bosch
**90** PUBLICATIONS **4,273** CITATIONS
SEE PROFILE

Pim Tuyls
**127** PUBLICATIONS **6,827** CITATIONS
SEE PROFILE

# Public-Key Cryptography for RFID-Tags

L. Batina[1], J. Guajardo[2], T. Kerins[2], N. Mentens[1], P. Tuyls[2], and I. Verbauwhede[1]

[1] Katholieke Universiteit Leuven, ESAT/COSIC, Belgium
{Lejla.Batina,Nele.Mentens,Ingrid.Verbauwhede}@esat.kuleuven.be
[2] Philips Research Laboratories, Eindhoven, The Netherlands
{Jorge.Guajardo,Tim.Kerins,Pim.Tuyls}@philips.com

**Abstract.** RFID-tags are a new generation of bar-codes with added functionality. They are becoming very popular tools for identification of products in various applications like *e.g.* supply-chain management. An emerging application is the use of RFID-tags for anti-counterfeiting by embedding them into a product. However, there is a risk related to naively using those tags for several applications. In particular, if no appropriate cryptographic measures are taken, the privacy of a user carrying tagged items can be severely damaged. In order to enable these applications and at the same time minimize the risks, public-key cryptography (PKC) offers attractive solutions. Whether a public-key cryptosystem can be implemented on an RFID tag or not remains an open problem. In this paper, we focus on the problem of anti-counterfeiting measures that can be provided by RFID-tags. More precisely, we investigate which PKC-based identification protocols are useful for this application. We discuss the feasibility of identification protocols based on Elliptic Curve Cryptography (ECC) and show that it is feasible on RFID tags.

**Key Words:** RFID, authentication, Elliptic Curve Cryptography (ECC)

## 1 Introduction

In recent years, the growth of counterfeit goods has experienced a rather steep increase. Such increase translates into a large source of losses for manufacturers. The following numbers will provide an idea of the extent and criticality of the problem: (i) it has been estimated that the world market for counterfeit goods was worth between 350 and 385 billion USD in 2001 and it was expected to surpass the 500 billion USD per year mark by 2004 [15, 35], (ii) in the copyright industry, almost 50% of all motion picture videos, more than 40% of all business software, and a third of all music recordings are pirated copies, (iii) about 10% of clothing, fashion and sports wear are fake and the online sales of luxury goods reaches 25 billion USD annually, (iv) in the automotive industry 5% to 10% of all spare parts are counterfeits, and (v) between 5% and 8% of the 500 billion USD in medicines sold worldwide are counterfeit as estimated by the World Health Organization [14, 35], in developing countries the percentage of counterfeit drugs account for up to 60% of all drugs [16, 20]. Notice that the above numbers only point to the economical consequences of counterfeit products. However, in the particular case of the pharma industry, counterfeit products have a direct (negative) impact on the health and life of thousands of people worldwide. It is clear that new technologies need to be put in place to thwart the counterfeiting threat. RFID has been identified as one of these technologies as shown for example by legislation introduced in the US mandating use of RFID technology as anti-counterfeiting technology for at-risk pharmaceuticals for all medicines in the supply chain by the end of 2010 [20] (see also [28]).

However, the use of RFID as an anti-counterfeiting technology is at present rather primitive. The whole security relies on the premise that an RFID tag is harder to copy than a bar code. Although, this is certainly true, it will only be a matter of time until counterfeiters can clone simple RFID tags. Thus, sound technological solutions for the counterfeiting problem need to be developed. By sound, we mean solutions based on cryptography, fundamental physical properties of materials that make them unclonable or a combination of both. Notice that the anti-counterfeiting problem can also be rephrased as an authentication problem. In other words, how can a reader tell that a certain RFID tag is really the one that it intended to talk to? In this setting, RFID-tags contain some secret reference information that is used to check their authenticity. In order to avoid counterfeiting, RFID-tags have to be unclonable. First, this implies that it should be hard to make a physical clone. Secondly, this also means that retrieving the secret reference information by attacking the protocols that are carried out between the reader and a tag (proving its authenticity) should be unfeasible. Protection against physical unclonability is provided by using physical countermeasures such as Physical Unclonable Functions [36] and protection against active or passive attacks on the protocols is provided by cryptographic techniques such as digital signatures and secure identification protocols. The required cryptographic primitives range from symmetric and asymmetric algorithms to hash functions and random number generators.

In short, RFID-based identification is an example of an emerging technology which requires authentication as a cryptographic service [9]. This property can be achieved by symmetric as well as asymmetric primitives. Previously known work considered only symmetric-key algorithms e.g. AES [8]. The suitability of Public-Key (PK) algorithms for RFID is an open research problem as limitations in costs, area and power are quite severe. Recently, a few papers [36, 38] discussed feasibility of ECC based PKC on RFID-tags. Here, we extend that line of work and discuss implementations aspects of even stronger PK-based protocols in more detail. In particular, the contributions of this paper are:

1. We present protocols for the case in which readers are on-line (and hence share a secret with the tags). It is shown that the protocols for this case are very cheap and can easily be implemented on a high-end RFID-tag. We emphasize that by todays standards, the tags that we consider would correspond to a mid to high range tag. Although, it is anticipated that in the near future price pressure will continue to limit the number of gates in the ultra low cost tags, it can also be envisioned that eventually this number of gates will be available on all tags. We also emphasize that the cost of a security solution is directly dependent on the thing(s) that are being safeguarded. Thus, just as there are applications for which our solutions would be too expensive, we believe that there are also RFID applications for which such cost might be acceptable.
2. We extend the research for off-line verification (where the readers do not share any secret with the tags) [36]. The protocols investigated in [36] were only secure against passive attacks. Here, we investigate the efficiency of protocols (Okamoto-identification protocol) that are also secure against active and concurrent attacks. It is shown that only a small price for much additional security has to be paid.
3. We present ECC-based implementation of the above mentioned protocols.

The remainder of the paper is organized as follows. Sections 2 and 3 provide an overview of related work and present the general setting of RFID-tags for anti-counterfeiting. Section 4 describes secure authentication protocols for RFID tags for the on-line case. In Sect. 5 the PUF-Certificate-Identity based approach for the off-line case is presented. Hardware implementations of the protocols for off-line verification are described in Sect. 6. Finally, our results are presented in Sect. 8.

## 2 Related Work

A first set of related papers to ours are [19] and [18]. Both deal with the cloning problem of RFID-tags and hence with the problem of using RFID-tags for anti-counterfeiting purposes. The focus of these papers is on efficient protocols for authenticating these tags. In these papers, one focuses on authentication of RFID-tags in the on-line situation; *i.e.* when the reader shares a secret with the RFID-tag that is being authenticated. Clearly, for applications with large deployments of RFID tags, this is not a reasonable assumption. Moreover, they do not take physical cloning into account. Recently, an attack on the protocol of [19] was presented in [12]. In [36], RFID-tags that withstand general cloning attacks (including physical ones) are introduced. Based on an Integrated PUF (I-PUF) [10, 21, 31] a PUF-Certificate-Identity Based identification scheme was introduced. This scheme allows for off-line authentication. In [36] the implementation of the Schnorr Identification scheme was investigated for this purpose. This protocol is only secure against passive attacks but it is very efficient.

There has not been many attempts at hardware implementations of PKC on RFID tags or other low-power application platforms *e.g.* sensor nodes. Gaubatz et al. [11] showed that RSA is not a feasible solution while NtruEncrypt can be implemented in not more than 3000 gates. More recent work of Wolkerstorfer [38] is the first to claim possible to have low-power and compact implementation of ECC that meets the constraints imposed by the EPC standard. However, our solution is smaller as the off-line authentication in our case does not require full ECDSA signature generation to be executed on the RFID tag. This allowed for further area optimizations.

## 3 Assumptions

We consider RFID-tags embedded in a product or its package for detection and prevention of product counterfeiting. The tag is manufactured and embedded into the product by a legitimate authority which is assumed to be trusted. We consider an active attacker that knows the position of the tag in the product or its package, so she can remove the tag from the package to investigate it. We also assume that the attacker can (passively) eavesdrop on the channel between a reader and the tag, or can install a fake reader that communicates with the tag (active attack). Finally, we assume that the attacker can physically attack the tag; *i.e.*

she can try to read out its memory. The goal of the attacker is to produce a fake RFID-tag containing reference information such that it can only be distinguished from a real tag with small probability . Clearly, by embedding such a fake tag into a fake product, the fake product is identified as an authentic one.

## 4 Authentication

We distinguish between on-line and off-line authentication. Off-line authentication is the most attractive one from a practical point of view but also the most challenging one, as costs grow much more in this case.

### 4.1 On-line Authentication

We assume that every reader is connected with a reference database through an authenticated channel. The reference database contains for each tag ID a list of Challenge-Response Pairs (CRPs) of its corresponding PUF. We assume that there is a *large* number of challenge response pairs available for the PUF. More precisely, we assume that the PUF that has so many challenge-response pairs such that an attack (performed during a limited amount of time) based on exhaustively measuring challenge-response pairs only has a negligible probability of success [37].

During the **enrollment phase** the PUF is challenged by a Certification Authority (CA) with $n$ independent challenges [37], say $c_1, \ldots, c_n$ and the corresponding responses $x_1, \ldots, x_n \in \{0,1\}^k$ are measured. The data $(c_i, x_i), i = 1, \ldots, n$ are securely stored in the database (and unknown to an attacker). No additional information is stored in the (ROM) memory of the RFID-tag.

During the **authentication phase**, the following protocol is performed between the tag and the reader.

1. The reader asks the tag for its identification number, ID.
2. The reader gets from the database a random challenge response pair say $(c_i, x_i)$ for this ID.
3. The reader sends the challenge $c_i$ over a public channel to the tag.
4. The tag challenges its PUF according to the challenge $c_i$, measures $y_i$ and sends $y_i$ over the public channel to the reader.
5. The reader verifies whether $d_H(x_i, y_i) \leq \delta$, where $\delta$ is some predetermined threshold. If this condition is satisfied, the reader considers the tag to be authentic, in the other case it is decided that this is a counterfeit tag.
6. The database removes the pair $(c_i, x_i)$ from the database.

**Security:** It is clear that in order to have a secure system for RFID-tags with some reasonable life time, a large number of CRPs is needed (e.g. $\sim 10^9$ [37]). Since the various CRP pairs are independent, a passive attacker has a probability of guessing a response $z_i$ with $d_H(z_i, r_i) \leq \delta$ to a challenge $c_i$ equal to $\sum_{i=0}^{\delta} \binom{k}{i}/2^k \approx 2^{(h(\alpha)-1)k}$ when $\delta = \alpha k$ and $h$ denotes the binary entropy function. Note that an active attacker will probe the PUF of the tag with a fake reader that sends well chosen challenges $c'_1, \ldots, c'_m$ to the tag. When the responses $y_1, \ldots, y_m$ are returned, he records those and uses them to make a model of the PUF and to guess the responses to other remaining (unused) challenge-response pairs. It was shown in [37] that the number of responses that can be obtained in a limited amount of time is small compared to the total number of challenges, *i.e.* $m \ll n$ (Typically $m = 100$ and $n = 10^9$). Hence, the probability that $c_j \in \{c'_1, c'_2, \ldots, c'_m\}$ for some $j \in_R \{1, \ldots, n\}$ is $\mathcal{O}(\frac{m}{n})$. This implies that the verifier has to keep its database with CRP-pairs secret. Finally, we note that after some time the database might run out of CRPs. In [10] protocols have been developed to update a CRP database with new CRPs.

**Complexity:** We note that from a computational point of view, this protocol is very inexpensive for the RFID-tag. It only has to measure responses to challenges. This requires only 1000 gates as was explained in [36]. Note that no cryptographic operations have to be performed. In another variant of this protocol keys are derived from the responses of the PUF using the helper data scheme.

## 5 Off-line Authentication

In [36] a construction for off-line authentication was given. It was called PUF-Certificate-Identity based Identification. For the sake of completeness we describe it briefly here but refer to [36] for the details. The construction of the PUF-Certificate-Identity-based Identification scheme (PUF-Cert-IBI) extends the one of Certificate-based IBI in [2]. Given a tag with identity $I$, a PUF, a standard identification scheme $\mathcal{SI} = (K_g, P, V)$ (where $K_g$ denotes the key generation algorithm, and $P, V$ denote the interactive protocols run by the prover and verifier respectively) and a secure signature scheme $\mathcal{SS} = (\mathrm{SK}_g, \mathrm{Sign}, V_f)$ (with $\mathrm{SK}_g$ denoting the key generation algorithm, Sign denoting the signing algorithm and $V_f$ the verification algorithm run by a verifier) an Identity-Based Identification scheme $(\mathrm{MK}_g, \mathrm{UK}_g, \hat{P}, \hat{V})$ is constructed as follows.

During **enrollment** the issuer uses $\mathrm{SK}_g$ as the master-key generation algorithm $\mathrm{MK}_g$ for the secure signature scheme. The algorithm $\mathrm{UK}_g$ creates for each tag a public-secret key pair $(pk, sk)$ using the algorithm $K_g$ for the SI-scheme. The issuer runs a protocol with the tag to determine the challenge $c$ for the PUF and helper data $w$ such that the PUF response $x(c)$ maps onto the secret key $sk$. The helper data $w$ are written into the ROM (EEPROM) memory of the tag. Finally, the issuer creates the following certificate that is also stored in the ROM of the tag $\mathrm{Cert} \leftarrow (pk, \mathrm{Sign}(msk, pk||I))$.

During **authentication** the algorithms $\hat{P}$ and $\hat{V}$ are run as follows. The tag (in the role of the prover) sends the certificate Cert to the reader. If Cert is valid, the tag and the reader run the SI-protocol. If the tag passes this protocol too, the reader decides that the tag is authentic and otherwise not. Note that in order to run this last step, the tag has to challenge its PUF and use the helper data to obtain the secret key $sk$ from the measured response $y(c)$.

The security of the scheme depends on three factors: (i) the security of the PUF as a secure storage of the secret key, (ii) the security of the identification scheme used, and (iii) the security of the signature scheme used. It was shown that if the PUF is unclonable and a good Fuzzy Extractor is used for key extraction, the PUF provides a secure way of storing secret keys. The security of the scheme against impersonation attacks depends on the security of the identification scheme used against those attacks. Therefore, it is of crucial importance to understand which trade-off is being made between efficiency and security.

### 5.1 Okamoto's Identification Protocol based on ECDLP

In [36], Schnorr's identification protocol [33] is used as the SI in the Cert-IBI. Furthermore, it is shown that the elliptic curve version of Schnorr's identification protocol can be efficiently implemented. Schnorr's protocol is, however, only resistant against passive attacks under the discrete logarithm assumption. Another protocol that is also resistant against active and concurrent attack under the discrete logarithm assumption is Okamoto's identification protocol [30] (see also [2]). We investigate therefore the efficiency of the implementation of this protocol here in detail. More precisely, the steps of the protocol are:

- *Commitment by a Prover-Tag:* The tag picks $r_i \in_R \{0, \ldots, n-1\}$, $i = 1, 2$ and sends $X = r_1 P_1 + r_2 P_2$ to the reader.
- *Challenge from a Verifier-Reader:* The reader picks a number $e \in [1, 2^t]$ and sends it to the tag.

1. **Common Input:** The set of system parameters in this case consists of: ($q$, $FR$, $a$, $b$, $P_1$, $P_2$, $n$, $h$). Here, $q$ specifies the finite field, $FR$ is a field representation, $a$, $b$, define an elliptic curve, $P_i$ is a point on the curve of order $n$ and $h$ is the cofactor. In the case of tag authentication, these parameters are assumed to be fixed.
2. **Prover-Tag Input:** The prover's secret $(s_1, s_2)$ such that $Z = -s_1P_1 - s_2P_2$.
3. **Protocol:** The protocol involves the exchange of the following messages:

| Prover $P$ | | Verifier $V$ |
|---|---|---|
| $r_1, r_2 \in_R \mathbb{Z}_n$ | | |
| $X \leftarrow r_1P_1 + r_2P_2$ | $\xrightarrow{X}$ | |
| | $\xleftarrow{e}$ | $e \in_R Z_{2^t}$ |
| $y_i = r_i + es_i \bmod n$,<br>$i = 1, 2$ | $\xrightarrow{y_1, y_2}$ | |
| | | If $y_1P_1 + y_2P_2 + eZ = X$<br>then accept else reject |

**Fig. 1.** Okamoto's identification protocol.

– *Response from a Tag:* The tag computes $y_i = r_i + es_i$, $i = 1, 2$ and sends them to the reader.
– The verifier checks that $yP + eZ$ equals $X$.
Check: $y_1P_1 + y_2P_2 + eZ = (r_1 + es_1)P_1 + (r_2 + es_2)P + e(-s_1P_1 - s_2P_2)Z = r_1P_1 + r_2P_2 = X$

*Remark 1.* We are considering Okamoto's identification protocol as it provides security against active adversaries and it is based on the hardness of the DL problem. Other protocols found in the literature include Beth's identification protocol [4] and the XDL-IBI scheme in [2]. Beth's protocol only requires one point multiplication but it remains an open problem to prove its security against active adversaries. The XDL-IBI scheme also requires only one point multiplication but is only secure against passive adversaries and concurrent attacks (under a modified assumption). Thus, it seems that by analyzing both Schnorr's and Okamoto's we cover the efficiency of all *available* ID protocols based on the hardness of the DL problem.

## 6 ECC Implementation of Authentication Protocols

In this section, we describe our hardware implementation for both authentication protocols i.e. the one of Schnorr and the scheme of Okamoto based on elliptic curve cryptography. We also compare results for both cases.

### 6.1 Elliptic Curve Cryptography

The main operation in any ECC-based primitive is the scalar multiplication.

**Point Multiplication** The point scalar multiplication is achieved by repeated point addition and doubling. We can use the basic double-and-add algorithm [24] in both cases. In the case of Schnorr's identification protocol, we can also use the Montgomery ladder method [26] and benefit from the Lopez-Dahab projective coordinates [23].

**Point Addition and Doubling** The point addition/doubling depend on the type of projective coordinate used. Table 1 summarizes the number of operations required for known projective coordinates in terms of multiplications, squarings, and additions. The number of operations are assuming general operands, i.e., no particular values for $Z$ or the curve coefficients $a, b$ are considered. In the case, of Jacobian coordinates, the number of multiplications and squarings can be reduced to 14 and 4, respectively for about half of all elliptic curves.

**Table 1.** Operation Counts for point addition and doubling

| Coordinate System | Addition | | | Doubling | | |
|---|---|---|---|---|---|---|
| | Mult. | Sqr. | Add. | Mult. | Sqr. | Add. |
| Jacobian projective $(X/Z^2, Y/Z^3)$ [32, 6] | 15 | 5 | 7 | 5 | 5 | 4 |
| Lopez-Dahab $(X/Z, Y/Z)$ [23] | 4 | 1 | 2 | 2 | 5 | 1 |
| Modified Lopez-Dahab $(X/Z, Y/Z)$ [36] | 6 | 1 | 2 | 3 | 5 | 1 |

**Field Operations** Fields of characteristic two in polynomial basis were chosen for this investigation as arithmetic can be implemented efficiently and relatively cheaply in hardware over these fields. Although this is well understood, few previous attempts have been made to develop truly low area implementations of this arithmetic for ECC. Addition of two elements $c = a + b \in \mathbb{F}_{2^n}$ is performed via an $n$–bitwise logical XOR operation. The standard way to compute the product $c = a \cdot b \in \mathbb{F}_{2^n} \cong$

$\mathbb{F}_2[x]/f(x)$, and $a = \sum_{i=0}^{n-1} a_i x^i$, $b = \sum_{j=0}^{n-1} b_j x^j$, $f = x^n + \sum_{i=0}^{s} f_i x^i$, $s < n$, is the one that uses convolution [5]

$$c = \sum_{j=0}^{n-1} \sum_{i=0}^{n-1} a_i b_j x^{i+j} \bmod f = a \sum_{j=0}^{n-1} b_j x^j \bmod f \tag{1}$$

This represents the most compact solution, where the $b_j a x^j$ partial products from (1) are computed iteratively and reduction modulo $f$ of the degree $n$ partial product polynomial is performed on each of the $n$ iterations. The digit serial multiplication algorithm [34] may be considered as a generalization of this. Rather than processing the binary coefficients $b_j$ of $b \in \mathbb{F}_{2^n}$ serially, a number of them are processed in parallel. Here there is scope to trade-off an increase in gate count for increased performance. This is an important consideration in low frequency implementations over relatively small (composite) fields as discussed here.

Here $b = \sum_{j=0}^{n-1} b_j x^j$, rather than being considered as $n$ coefficients of $\mathbb{F}_2$ is considered as being composed of $d = \lceil \frac{n}{D} \rceil$ *words*, each word containing $D$ elements of $\mathbb{F}_2$. Now $b = \sum_{k=0}^{d-1} \tilde{b}_k x^{kD}$, each $\tilde{b}_k = \sum_{l=0}^{D-1} b_{l+kD} x^l$, and

$$c = \sum_{k=0}^{d-1} (a\tilde{b_k}) x^{kD} \bmod f \tag{2}$$

can be calculated in $d$ iterations. Notice that the $\tilde{b}_k a$ partial products are calculated recursively. A variant of the Song-Parhi method is illustrated as Algorithm 1. When $D = 1$ then $d = n$ and $\tilde{b}_k = b_j \in \mathbb{F}_2$ and this method reverts to Horner multiplication. Squaring $c = a^2 \in \mathbb{F}_{2^n}$ is a special case of multiplication [7]. It is well known that $a^2 = \sum_{i=0}^{n-1} a_i x^{2i}$ which can then be reduced modulo $f$ to a field element in $\mathbb{F}_{2^n}$.

**Algorithm 1** Digit serial multiplication in $\mathbb{F}_{2^n}$

```
Require: $a = \sum_{i=0}^{n-1} a_i x^i$, $b = \sum_{k=0}^{d-1} \tilde{b_k} x^{kD}$ where $\tilde{b_k} = \sum_{j=0}^{D-1} b_{l_k} x^l$ and $f \in \mathbb{F}_2[x]$
Ensure: $c = a \cdot b \bmod f(x)$
c ← 0
for $k$ from 1 to $d - 1$ do
    $c \leftarrow x^D (c + \tilde{b}_{d-1} a) \bmod f$
    $b \leftarrow x^D b$                                {Only a $D$-bit left shift}
end for
$c \leftarrow (c + \tilde{b}_{d-1} a) \bmod f$
Return $c$
```

## 7 ECC processor

Our Elliptic Curve Processor (ECP) for RFID is shown in Fig. 2. The operational blocks are as follows: a Control Unit(CU), an Arithmetic Unit (ALU), and Memory (RAM and ROM). The ECC parameters and the constants are stored in ROM. On the other hand, RAM contains all input/output and intermediate variables and it therefore communicates with both, the ROM and the ALU.

The Control Unit controls scalar multiplication and point operations. In addition, the controller commands the ALU which performs field multiplication, addition and squaring. When the START signal is set, the bits of $k = \sum_{i=0}^{n-1} k_i 2^i$, $k_i = \{0, 1\}$, $n = \lceil \log_2 k \rceil$, are evaluated from MSB to LSB. When all bits have been evaluated, an internal counter gives an END signal. The result of the last point calculation is written to the output register and the VALID output is set. The CU consists of a number of simple state machines and a counter and its area cost is small.

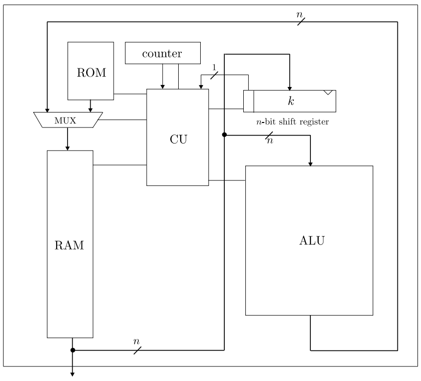

**Fig. 2.** ECP Architecture.

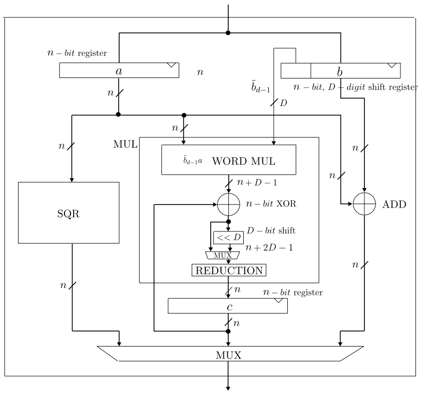

**Fig. 3.** ALU Architecture.

## 8 Results

In this section, we provide results for the latency and the area complexities of both Schnorr's and Okamoto's protocols. As we are interested in implementations of identification protocols (e.g. Schnorr, Okamoto) the operation required is one point multiplication in the case of Schnorr's protocol or multiple-point multiplication in the case of Okamoto's scheme.

### 8.1 Implementation of the Okamoto's scheme

In [36], the *feasibility* of the ECC version of Schnorr's identification protocol in an RFID system was investigated and area and latency estimates were provided. Here, we provide detailed numbers and we also investigate the feasibility of the Okamoto's scheme as it provides security against active adversaries which Schnorr's scheme does not.

For the case when point multiplication is implemented by means of Montgomery's ladder and the point operations are implemented as in [23, 36], it can be shown that the number of cycles required for one point multiplication is $(n-1)(9\lceil\frac{(n-1)}{D}\rceil+56)$ and $(n-1)(9\lceil\frac{(n-1)}{D}\rceil+57)$ with and without a dedicated squarer circuit, respectively.

As can be seen from Okamoto's scheme, the required computation on a tag is of a form $kP+lQ$ *i.e.* so-called multiple-point multiplication. For the purpose of speeding-up this computation one uses Shamir's trick [13]. The scalars $k$ and $l$ are stored in a 2-row matrix in which each row contains binary representation of one of the scalars. All values of the form $iP+jQ$, $0 \leq i, j < 2^w$ are precalculated and stored where $w$ is given width of the window. The algorithm to perform this so-called simultaneous point multiplication is computing at each of $\lceil\frac{t}{w}\rceil$ steps $w$ doublings and 1 addition from the list of the precalculated values of the form $iP+jQ$. As a width of the window $w$ is a variable that allows some trade-off, we chose the smallest window *i.e.* $w = 1$. In this way, the memory requirements are minimized as only 3 points have to be stored: $P, Q, P+Q$. The exact computation is given in Algorithm 2 [13]. The expected running time of the algorithm for $w = 1$ is $\frac{3}{4}t$ point additions and $(t-1)$ point doublings.

**Algorithm 2** Simultaneous point multiplication

```
Require: $k = (k_{t-1}, \ldots, k_0)_2$, $l = (l_{t-1}, \ldots, l_0)_2$, $P, Q$ points on the curve
Ensure: $R = kP + lQ$
Compute $P + Q$
$R \leftarrow \infty$
for $i$ from $t-1$ downto 0 do
    $R \leftarrow 2R$
    $R \leftarrow R + (k_iP + l_iQ)$
end for
Return(R)
```

We have implemented the Schnorr scheme in VHDL and obtained area and timing values for a 0.25$\mu$m CMOS library. We have used these values to estimate the performance of the binary method of multiplication (i.e. using Jacobian coordinates and the binary method for point multiplication) and of Okamoto's identification protocol using Shamir's Trick. Table 2 summarizes the results. We notice that the amount of logic required to support Okamoto's protocol is not significantly larger than that corresponding to the implementation of Schnorr's. However, the required RAM to implement Okamoto's identification protocol is more than twice the required RAM required for Schnorr's. In practice these means an increase in area anywhere from 20 to 50% depending on the chosen RAM implementation (i.e. whether a RAM cell is implemented as a register requiring at least 6 equivalent gates worth of area or as dedicated embedded RAM requiring somewhere between 1.5 and 2 equivalent gates [27, 17]). In terms of latency, Okamoto's identification protocol is almost twice as slow as Schnorr's over elliptic curves due to the fact that the coordinate representation introduced in [23] is only applicable to the Montgomery Ladder method of exponentiation. In addition, simultaneous double exponentiation is naturally about 25% slower than the regular binary method for exponentiation. With respect to the most compact solution, as required due to low gate-count and low-power requirements, implementing curve-based protocols with shorter bit-lengths appears to be an attractive option. For example, in the case of ECC one could use 130-bit long parameters. This solution would still maintain a suitable level of security [22], especially for low-cost RFIDs, and the gate complexity would scale-down accordingly resulting in more attractive solutions from the area and performance points of view. We conclude by noticing that the performance of the simultaneous point multiplication (as well as the binary method) can be easily improved by using Non-Adjacent Form representation for the multiplier. Such methods in the binary case would for example reduce the number of multiplications from a half on average to a third, providing significant performance improvements (see for example [25]).

**Table 2.** Implementation results @ 175 $kHz$ and assuming a dedicated squarer circuit.

| Implementation | | ALU | RAM Mont. | RAM | RAM | Perf. Mont. | Perf. | Perf. | Area wo RAM |
|---|---|---|---|---|---|---|---|---|---|
| Digit size | Field Type | [gates] | Ladder | Binary | Okamoto | Ladder | Binary | Okamoto] | [gates] |
| | | | [bits] | [bits] | [bits] | [$s$] | [$s$] | [$s$] | |
| D=1 | $\mathbb{F}_{2^{131}}$ | 6306 | 917 | 1965 | 2096 | 0.91 | 1.23 | 1.59 | 8582 |
| | $\mathbb{F}_{2^{139}}$ | 6690 | 973 | 2085 | 2224 | 1.02 | 1.38 | 1.79 | 9044 |
| | $\mathbb{F}_{2^{163}}$ | 7846 | 1141 | 2445 | 2608 | 1.38 | 1.89 | 2.44 | 10122 |
| D=2 | $\mathbb{F}_{2^{131}}$ | 6962 | 917 | 1965 | 2096 | 0.48 | 0.65 | 0.83 | 8603 |
| | $\mathbb{F}_{2^{139}}$ | 7379 | 973 | 2085 | 2224 | 0.53 | 0.72 | 0.93 | 9734 |
| | $\mathbb{F}_{2^{163}}$ | 8663 | 1141 | 2445 | 2608 | 0.71 | 0.98 | 1.27 | 10933 |

### 8.2 Generation of Randomness

Often a source of randomness is needed on the tag; this can be derived from thermal noise, shot noise, jitter, etc. Here we will derive that randomness from the PUF. In general this can be done by applying a random challenge to the PUF e.g. in a range out of its specification. The random challenge can be generated by the reader. For a construction of a random number generator based on a PUF, we refer to O'Donnel *et al.* [29].

## 9 Concluding Remarks

In this paper we discussed the feasibility of public key based secure identification protocols for RFID-tags. As an example we investigated the implementation of Okamoto's identification protocol in detail. It was shown that it is just slightly more expensive than Schnorr's identification protocol. Finally, we notice that the performance of Okamoto's protocol can be further improved using the techniques presented in [1] and recently improved in [3]. Such improvements will be considered in future work.

## Acknowledgments

Thanks to the anonymous referees for helpful comments and for bringing to our attention reference [1].

## References

1. Toru Akishita. Fast Simultaneous Scalar Multiplication on Elliptic Curve with Montgomery Form. In S. Vaudenay and A. M. Youssef, editors, *Selected Areas in Cryptography — SAC 2001*, volume 2259 of *LNCS*, pages 255–267. Springer, 2001.
2. Mihir Bellare, Chanathip Namprempre, and Gregory Neven. Security proofs for identity-based identification and signature schemes. In C. Cachin and J. Camenisch, editors, *Advances in Cryptology — Eurocrypt 2004*, volume 3027 of *LNCS*, pages 268–286. Springer-Verlag, 2004.
3. D. J. Bernstein. Differential addition chains. Technical Report Document ID: 9620b81ea01f66b2a782be234dade959, February 19th, 2006. Available at http://cr.yp.to/papers.html.
4. T. Beth. Efficient Zero-Knowledge Identification Scheme for Smart Cards. In C. G. Günther, editor, *Advances in Cryptology — EUROCRYPT'88*, pages 77–84, 1988.
5. T. Beth and D. Gollmann. Algorithm engineering for public key algorithm. *IEEE Journal on Selected Areas in Communications*, 7(4):458–465, May 1989.
6. D.V. Chudnovsky and G.V. Chudnovsky. Sequences of numbers generated by addition in formal groups and new primality and factorization tests. *Advances in Applied Mathematics*, 7(4):385–434, 1986.

7. E. D. Mastrovito. *VLSI Architectures for Computation in Galois Fields.* PhD thesis, Dept. Electrical Engineering, Linköping University, Linköping, Sweeden, 1991.
8. M. Feldhofer, S. Dominikus, and J. Wolkerstorfer. Strong Authentication for RFID Systems using the AES Algorithm. In M. Joye and J. J. Quisquater, editors, *Proceedings of 6th International Workshop on Cryptographic Hardware in Embedded Systems (CHES)*, volume 3156 of *LNCS*, pages 357–370. Springer-Verlag, 2004.
9. International Organization for Standardization. ISO/IEC 18000-3. Information Technology AIDC Techniques - RFID for Item Management, March 2003.
10. B. Gassend, D. E. Clarke, M. van Dijk, and S. Devadas. Silicon physical random functions. In Vijayalakshmi Atluri, editor, *ACM Conference on Computer and Communications Security — CCS 2002*, pages 148–160. ACM, November 18-22, 2002.
11. G. Gaubatz, J.-P. Kaps, and B. Sunar. Public Key Cryptography in Sensor Networks - Revisited. In *1st European Workshop on Security in Ad-Hoc and Sensor Networks (ESAS 2004)*, Heidelberg, Germany, August 2004.
12. H. Gilbert, M. Robshaw, and H. Sibert. An Active Attack Against HB+ — A Provably Secure Lightweight Authentication Protocol. IACR ePrintArchive 2005/237, 2005.
13. D. Hankerson, A. Menezes, and S. Vanstone. *Guide to Elliptic Curve Cryptography.* Springer-Verlag, 2004.
14. Business Action to Stop Counterfeiting and Piracy  Fact Sheet. Technical report, ICC.
15. ICC Policy Statement: The fight against piracy and counterfeiting of intellectual property. Submitted to the 35th World Congress, Marrakech, Document no 450/986, ICC, June 1st, 2004.
16. Intellectual Property: Source of innovation, creativity, growth and progress. Technical report, ICC, August 2005.
17. K. Itoh. Low-Voltage Embedded RAMs in the Nanometer Era. In *IEEE International Conference on Integrated Circuits and Technology — ICICT 2005*, pages 235–242. IEEE Computer Society, 2005.
18. A. Juels. Strengthening EPC Tags Against Cloning. In M. Jakobsson and R. Poovendran, editors, *ACM Workshop on Wireless Security — WiSe 2005* , pages 67–76. ACM Press, 2005.
19. A. Juels and S. A. Weis. Authenticating pervasive devices with human protocols. In V. Shoup, editor, *Advances in Cryptology: Proceedings of CRYPTO 2005*, volume 3621 of *LNCS*, pages 293–308. Springer-Verlag, 2005.
20. R. Koh, E. W. Schuster, I. Chackrabarti, and A. Bellman. Securing the Pharmaceutical Supply Chain. White Paper MIT-AUTOID-WH-021, Auto-Id Center MIT, Cambridge, Ma 02139-4307, USA, September 1st, 2003. Available at http://www.mitdatacenter.org/MIT-AUTOID-WH021.pdf.
21. J. Lee, D. Lim, B. Gassend, G. E. Suh, M. van Dijk, and S. Devadas. A Technique to Build a Secret Key in Integrated Circuits for Identification and Authentication Applications. In *VLSI Circuits Symposium*, pages 176–179. IEEE Computer Society, June 17-19, 2004.
22. A. Lenstra and E. Verheul. Selecting cryptographic key sizes. In H. Imai and Y. Zheng, editors, *Workshop on Practice and Theory in Public Key Cryptography — PKC 2000*, volume 1751 of *LNCS*, pages 446–465. Springer-Verlag, 2000.
23. J. López and R. Dahab. Fast multiplication on elliptic curves over GF($2^m$). In Ç. K. Koç and C. Paar, editors, *Proceedings of 1st International Workshop on*

*Cryptographic Hardware and Embedded Systems (CHES)*, volume 1717 of *LNCS*, pages 316–327. Springer-Verlag, 1999.

24. A. Menezes, P. van Oorschot, and S. Vanstone. *Handbook of Applied Cryptography.* CRC Press, 1997.
25. Bodo Möller. Algorithms for Multi-exponentiation. In S. Vaudenay and A. M. Youssef, editors, *Selected Areas in Cryptography — SAC 2001*, volume 2259 of *LNCS*, pages 165–180. Springer, 2001.
26. P. Montgomery. Speeding the Pollard and Elliptic Curve Methods of Factorization. *Mathematics of Computation*, Vol. 48:243–264, 1987.
27. Y. Nakagome, M. Horiguchi, T. Kawahara, and K. Itoh. Review and future prospects of low-voltage RAM circuits. *IBM Journal of Research and Development*, 47(5/6):525–552, 2003.
28. M. C. O'Connor. Pfizer Using RFID to Fight Fake Viagra. *RFID Journal*, January 6th, 2006.
29. C.W. O'Donnel, G.E. Suh, and S. Devadas. PUF-Based Random Number Generation. Technical Report 481, MIT CSAIL, November 2004. Available at http://www.csg.csail.mit.edu/pubs/publications.html.
30. T. Okamoto. Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes. In E. F. Brickell, editor, *Advances in Cryptology — CRYPTO'92*, volume 740 of *LNCS*, pages 31–53. Springer, 1992.
31. B. Skoric P. Tuyls. Secret key generation from classical physics. *Philips Research Book Series*, September 2005.
32. *IEEE P1363-2000: IEEE Standard Specifications for Public Key Cryptography*, 2000. Available at http://standards.ieee.org/catalog/olis/busarch.html.
33. C.-P. Schnorr. Efficient Identification and Signatures for Smart Cards. In Gilles Brassard, editor, *Advances in Cryptology — CRYPTO '89*, volume LNCS 435, pages 239–252. Springer, 1989.
34. L. Song and K.K. Parhi. Low Energy Digit-Serial/Parallell Finite Field Multipliers. *Kluwer Journal of VLSI Signal Processing Systems*, 19(2):149–166, 1998.
35. T. Staake, F. Thiesse, and E. Fleisch. Extending the EPC Network – The Potential of RFID in Anti-Counterfeiting. In A. Omicini H. Haddad, L. M. Liebrock and R. L. Wainwright, editors, *ACM Symposium on Applied Computing — SAC 2005*, pages 1607–1612. ACM Press, March 13-17 2005.
36. P. Tuyls and L. Batina. RFID-tags for Anti-Counterfeiting. In D. Pointcheval, editor, *Topics in Cryptology - CT-RSA 2006*, volume 3860 of *LNCS*, pages 115–131. Springer Verlag, February 13-17 2006.
37. P. Tuyls, B. Skoric, S. Stallinga, A.H.M. Akkermans, and W. Ophey. Information theoretical security analysis of physical unclonable functions. In A.S. Patrick and M. Yung, editors, *Proceedings of 9th Financial Cryptography and Data Security Conference*, volume 3570 of *LNCS*, pages 141–155. Springer-Verlag, 2005.
38. J. Wolkerstorfer. Scaling ECC Hardware to a Minimum. In ECRYPT workshop - Cryptographic Advances in Secure Hardware - CRASH 2005, September 6-7 2005. invited talk.