The use of Smart Card ICs has become more widespread, having expanded from historical banking and telecommunication applications to electronic passports, electronic IDs, anti-counterfeiting devices, smartgrid applications, and more. The security requirements for most of these applications are crucial and evolving. In addition, more and more sophisticated attacks are being developed every day. As a result, design of Smart Card ICs is a growing challenge.

This paper summarizes the present-day security challenges for Smart Card ICs and describes how a special technology, called Physical Unclonable Functions (PUF), delivers comprehensive protection in today’s applications. PUF technology provides a secure method for storing a key, withstanding today’s attacks, and even protecting against future potential attacks.

Note: For the purposes of this document, the term "Smart Card ICs" refers to microcontrollers based on smart-card secured technologies in traditional smart-card applications and in the secure elements of NFC-enabled devices, authentication tokens, and other high-security modules.

Types of attacks
The Smart Card industry typically places attacks in one of three categories:

- Side channel attacks (non-invasive attacks) – such as using information out of the power profile or the electromagnetic emanation
- Fault attacks (semi-invasive attacks) – such as disturbing the IC by applying laser light or a voltage glitch
- Reverse engineering (invasive attacks) – reverse engineering parts of the IC, possibly combined with probing signals
There has been important progress in all of these attack categories during the last few years. Smart Cards have to use sophisticated countermeasures to withstand these new attacks. In some markets, such as electronic passports, Smart Card ICs have to withstand attacks in the field for the ten years they are valid.

Reverse engineering attacks are back in focus, especially after recent attacks on some widely used Smart Card ICs. These attacks often have a much higher impact than side channel and fault attacks because there are essentially no ways compensate for them with additional software countermeasures. Fault and side channel attacks are often carried out on a limited basis, on specific modules or portions of the device, and can be addressed with additional software countermeasures.

There are several countermeasures available today that hinder reverse engineering and prevent attacks, and new technology nodes and more sophisticated techniques continue to improve security. But, in the end, given unlimited effort, there is an attack path for every chip. The reality is that there is no such thing as guaranteed protection.

Typical reverse engineering attacks on Smart Card ICs include the following:

- Reverse engineering of a functional block
- Reverse engineering of parts of the IC as preparation for a subsequent probing attack
- Extracting memory content

The standard countermeasures taken against reverse engineering attacks include the following:

- Memory encryption
- Encryption of data
- Scrambled logic (especially no hard macros)
- No logic relevant to security in top metal layers

**The problem of storing a key**

The standard countermeasures used against reverse engineering today are unlikely to be able to protect against future challenges. The memories of a Smart Card device (ROM, EEPROM, Flash, RAM) are usually protected by memory encryption. These memories contain security-relevant assets which need to be protected. The Flash or ROM also contains the Smart Card Operating System (OS) code, which is critical intellectual property (IP) and essential to protect.

There is, of course, some basic physical protection for the different memories. But attackers have been able to extract some (even if only a little) data from these kinds of memories. It is expected that these attack techniques will improve significantly in coming years. At that point, the protection of the assets is down to the memory encryption being used. A fundamental issue which still remains – even with a theoretically unbreakable memory encryption – is the protection of the key. Storing the key in one of these memory areas is not an option if this area can be read out.

An alternative approach is needed. One such alternative is Physical Unclonable Function (PUF) technology.

**Physical Unclonable Function (PUF) **

Physical Unclonable Functions (PUFs) are defined as functions based on physical characteristics which are unique for each chip, difficult to predict, easy to evaluate and reliable. These functions should also be individual and practically impossible to duplicate. PUFs can serve as a root of trust and can provide a key which cannot be easily reverse engineered.
In principle, any physical device characteristic that fluctuates can be turned into a PUF. Two prominent examples of PUFs in Smart Card applications (and other secure applications) are arbiter PUFs and SRAM-based PUFs. Arbiter PUFs rely on race signal conditions, and are not the focus of this paper. SRAM-based PUFs work with the SmartCard IC’s internal memory, and are described in more detail below.

**PUF and environment monitoring**

In addition to SRAM-based and arbiter PUFs, there are other PUF technologies that can be used to help monitor the surrounding environment of the Smart Card IC, including the card body.

The IC itself can check whether the environment is intact. During production or personalization, the IC measures its PUF environment and stores this unique measurement. From then on, the IC can repeat the measurement (preferably during startup), and check if the environment has changed, which would indicate an alteration in the card body. This protects against many kinds of invasive attacks.

**PUF technology in NXP’s next-generation Smart Card ICs**

SRAM PUFs rely on physical characteristics of SRAM technology. These types of PUFs are currently being integrated into NXP’s next-generation Smart Card ICs. After powering the Smart Card IC (and as a result the embedded SRAM), the cells are initialized with a pattern randomly made of zero and one logical values. This startup behavior – each specific bit in SRAM getting zero or one as an initial value - is different for every individual chip. But, looking at an individual Smart Card IC, this random initialization of SRAM content is very similar from one startup to another (for a single device). Small deviations in processing inside a SRAM cell lead to variants of electrical characteristics for each transistor. The SRAM cell design is symmetrical but the deviations lead to a small asymmetry resulting in a preferred state (0 or 1) during startup. The SRAM content after startup can serve as a unique fingerprint of the Smart Card IC. As the behavior is not completely the same for every startup, and some of the SRAM cells show different initialization after startup, error correction is necessary. Typically, codes for error correction (such as Reed-Solomon codes) are used to derive a unique device fingerprint. The derived fingerprint can then be used as a key, to protect a cryptographic key, or to protect a memory. The physical behavior over the device lifetime, as well as the error probabilities, make it essential to evaluate the reliability in a typical Smart Card environment together with a suitable post processing function (i.e. error correction).

**SRAM PUF implementation**

The SRAM PUF generates a device-individual fingerprint using the startup behavior of SRAM cells. The generated SRAM PUF fingerprint can be used for various use cases. It can be used directly as a key or indirectly to protect sensitive data (e.g. application keys).

The SRAM PUF typically consists of the following components:

1. SRAM area used as PUF source
2. Measurement circuit and error correction used to derive an individual IC fingerprint (hardware)
3. PUF IP (Activation Code Constructor and Key Extractor), which adds functionality to protect keys or memories (hardware)
The PUF IP adds functionality to protect application keys with the IC individual fingerprint. It acts as an internal key vault and therefore solves the problem of storing a key. The OS has to provide the application keys to the PUF IP. The PUF IP then uses the IC fingerprint to protect these values. The PUF IP needs to store an activation code, which can be public related to the respective application key, and stored in non-volatile (NV) memory such as EEPROM or Flash. The key is essentially split into two parts -- the SRAM PUF fingerprint and the activation code. The attacker must know both values to reconstruct the application key.

PUF usage is typically divided into two phases, Enrollment and Reconstruction.

**Enrollment Phase**

**Reconstruction Phase**

The figure shows the PUF and Error Correction post-processing that produces the PUF data, which is the IC’s individual fingerprint.

The Enrollment Phase occurs just once, when a new key is generated or being stored. The key is put into the Activation Code Constructor, which produces the activation code to be stored in NV memory.

In the Reconstruction Phase, the activation code is used in the Key Extractor to reconstruct the key. The actual key is not stored in NV memory. The key cannot be derived with the activation code alone; the code and the PUF data must both be available to reconstruct the key.

**PUF reliability and limitations**

To evaluate the reliability of SRAM PUFs, researchers have analyzed the physical characteristics of SRAM startup behavior under various operation conditions. Temperature and aging have been given special attention. The results have shown that SRAM PUFs are suitable for use in Smart Card ICs with a lifetime of more than 10 years.
It’s also important to understand the limitation of a countermeasure based on PUF technology. PUF is primarily used to protect critical data, such as keys or complete memories, from offline attacks. Offline attacks are attacks mounted when the device is powered off. There might be further attack paths on the critical data when it is being used (the device is “online”). For instance, PUF can’t provide additional protection when a key is being used for a cryptographic operation. This still needs to be covered by other measures.

A PUF implementation also has to be carefully designed and security tested such that the PUF implementation itself is not opening other security attack paths, such as some weaknesses with respect to side channel or fault attacks.

**SRAM PUF use cases**

**Key protection**
As described above, SRAM PUFs can protect application keys or other critical user assets in a way that is more secure than just storing the key in NV memory. The following table compares the two methods: The traditional method, where the application key is stored in NV memory, and the PUF-based one, where PUF is added to protect the application key. The table provides an overview of the different attack categories together with an assessment whether the attacks are applicable, now and in the future, without and with PUF technology.

### Attacks on Smart Card ICs without and with PUF technology (Key protection)

<table>
<thead>
<tr>
<th>Type of attack</th>
<th>Traditional technology, application key stored in NV</th>
<th>Application key stored with PUF technology</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Today</td>
<td>Tomorrow</td>
</tr>
<tr>
<td>Copy NV memory contents from one device to another (offline)</td>
<td>Difficult to impossible</td>
<td>May be possible</td>
</tr>
<tr>
<td>Extract application key (offline)</td>
<td>Difficult to impossible</td>
<td>May be possible</td>
</tr>
<tr>
<td>Extract application key with fault attack (by disturbing I/O function)</td>
<td>Difficult (normally secured by software and hardware firewalls)</td>
<td>May be possible</td>
</tr>
</tbody>
</table>

Reading out Non-Volatile (NV) memories (i.e. EEPROM or Flash) is mostly impossible, but reading out ROM contents may be easier. Thus far, only a few bytes have been successfully read out of EEPROM or Flash. But even if this is not feasible today, attacks are likely to improve significantly in future. If at some point it is possible to read out larger parts of the memories, or even the complete memory, the protection for the application key is gone.

PUF technology also protects against fault attacks in this use case. With standard approaches (where the application key is stored in NV), there might be ways to manipulate an I/O function such that the application key is compromised. A key protected with PUF technology cannot be extracted with this attack. Only the (non-sensitive) activation code can be extracted.

**Memory protection, countermeasure against cloning**
Another important use case for PUF technology within Smart Card ICs is the encryption of memories. Nowadays, the encryption keys for ROM, EEPROM and Flash memories have to be stored in the memory areas as well. If – at some point in the future – it becomes possible to read out complete memory contents, there is no secure place to store the encryption key. In this case not even the strongest memory encryptions will prevent successful attacks.
Attacks on Smart Card ICs with and without PUF technology (Cloning)

<table>
<thead>
<tr>
<th>Attack</th>
<th>Standard memory encryption</th>
<th>Memory encryption using PUF technology</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Today</td>
<td>Tomorrow</td>
</tr>
<tr>
<td>Copy NV memory from one device to another (offline)</td>
<td>Difficult to impossible</td>
<td>May be possible</td>
</tr>
<tr>
<td>Extract memory contents (offline)</td>
<td>Difficult to impossible</td>
<td>May be possible</td>
</tr>
</tbody>
</table>

PUFs can play a crucial role in future resistance to reverse engineering attacks on memory. Offline attacks on memory content (such as applications keys or critical IP) become impossible when using PUF technology for memory encryption.

External memory protection, countermeasure against cloning

In some applications, a Smart Card might also be connected to an external (unprotected) memory. Standard external memories (such as Flash memories in USB sticks) are easy to read. PUF technology can be used to protect these kinds of insecure external memories. PUF technology can serve as a root of trust, implementing the source for the memory encryption.

Attacks on external memories with and without PUF technology

<table>
<thead>
<tr>
<th>Type of attack</th>
<th>External memories (unprotected)</th>
<th>External memories protected by PUF technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extract (sensitive) data (offline)</td>
<td>Easy</td>
<td>Impossible</td>
</tr>
<tr>
<td>Clone / copy data from one device to another (by reverse engineering memory contents, offline)</td>
<td>Easy</td>
<td>Impossible</td>
</tr>
</tbody>
</table>

Fingerprinting

Another use case of PUF technology within the Smart Card IC area is fingerprinting. The PUF is used to provide every device with an individual fingerprint, which is characterized and stored in a database during the production phase. At a later stage, every device can be identified in the field using this PUF fingerprint information.

PUFs in different applications

Although Smart Card ICs are used in a variety of different application scenarios, the applications have several things in common. The Smart Card IC usually protects sensitive user data (such as private keys for electronic signatures), and the OS source code is often seen as a critical asset that needs to be protected. If attackers gain access to the source code, it's easier to identify areas of vulnerability, and source code is typically a crucial piece of intellectual property for the OS provider. This next section looks at specific requirements for today's most popular applications for Smart Card ICs.

eGovernment

Electronic passports (ePassports) and electronic ID cards (eIDs) are the leading Smart Card applications in eGovernment. Passports equipped with Smart Card ICs were introduced in 2005. The Smart Card ICs typically store personal information that is also printed on the document -- such as name and photo -- along with optional data such as fingerprints. ePassports also usually include a mechanism for authentication, called chip or active authentication, that prevents the document from being copied and provides proof that the document is valid. Authentication relies on a private key hidden in the Smart Card IC's secure memory. PUF technology can protect the key and prevent reverse engineering.
Using PUF technology, the private key used in a passport (which prevents the cloning of the document) is more strongly protected against (memory) reverse engineering attacks than with standard methods.

With eID applications, the situation is similar. The document typically uses an authentication method based on a private key stored in NV memory. PUF technology can be used for added protection against future attacks. The eID may also use an electronic signature, based on a private key.

Some eID cards use a private authentication key, called a group key, that is shared among a set of cards. The idea is that a group key does a better job of masking the identity of each individual in the group, making it harder for thieves to track a given user. The security requirements for this kind of key are especially high, because anyone successfully extracting the key ID has access to more information. PUF technology can protect authentication keys or the memory that contains the key.

Payment
In the payment market, MasterCard, VISA and other large players are defining the standards and the usage of Smart Card ICs. The cryptographic protocols used within Smart Card ICs are defined by the EMVCo standard. This standard comes with a set of protocols that serve different application scenarios and provide different security levels.

There are two main EMVCo protocol branches: Static Data Authentication (SDA), which may include an Online Authentication procedure, and Dynamic Data Authentication (DDA). Both protocols rely on a private key which is buried in the Smart Card IC’s memory. Keeping the key private prevents the payment card from being copied.

Mobile devices
Smart Card ICs are now in widespread use in mobile phones and tablets. A common use case is the electronic wallet, where the Smart Card IC acts as a secure element to emulate a bank card in a mobile phone. The Smart Card IC, equipped with data similar to that of a bank card, works with a contactless interface to let the phone serve as a contactless payment card. In this case the threats and attacks are the same as for a standard payment card.

In future, PUF technology might be used to protect external memories. The secret key used for encrypting external memories (partly or just some application keys) could be provided by PUF technology.
**Authentication**

In the consumer segment, which includes mobile phones and tablets, manufacturers need to protect against the copying of accessories. In the case of printers, for example, copying or cloning printer cartridges is a serious threat. Some manufacturers are integrating Smart Card ICs into their accessories to prevent cloning and unauthorized access. Because this is a new trend, formal standards for cryptographic protocols have yet to be developed, but most are based on a symmetric or asymmetric cryptography. The security relies on a private key buried in the Smart Card IC. PUF technology can, again, be used to increase security of the key.

<table>
<thead>
<tr>
<th>Type of attack</th>
<th>Private key for device authentication stored in NV memory</th>
<th>Private key for device authentication stored with PUF technology</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Today</td>
<td>Tomorrow</td>
</tr>
<tr>
<td>Extracting Private key material (offline)</td>
<td>Hard to impossible</td>
<td>Maybe possible</td>
</tr>
<tr>
<td>Cloning of a device (by reverse engineering the memory content)</td>
<td>Hard to impossible</td>
<td>Maybe possible</td>
</tr>
</tbody>
</table>

**Conclusion**

Reverse engineering has emerged as one of the most dangerous kinds of attacks for Smart Card ICs. Unlike other attacks on Smart Card ICs, including side channel or fault attacks, reverse engineering attacks are difficult to protect against using additional software countermeasures.

Partial reads of NV memory are already possible, and full reads are likely to be possible sometime in the future. This means that even the strongest memory encryption methods won’t be able to protect stored assets over the longer term.

SRAM PUFs, with their ability to protect against offline attacks, are an especially useful tool in the fight against reverse engineering. By generating a unique IC fingerprint, SRAM PUFs make it much harder to attack memories and other sensitive data when the device is powered off. With SRAM PUFs, it’s possible to implement a secure mechanism for key reconstruction without storing the key. In this way, PUF technology effectively addresses the issues of reverse engineering and NV memory readouts.

SRAM PUFs will be a key element of the “Integral Security Concept” used in NXP’s Smart Card ICs. NXP is also investigating on future PUFs to be used for other security countermeasures. One idea is a PUF which enables the Smart Card IC to check if the surrounding environment is still intact. The chip measures the environment (the surrounding card body) in the production and personalization stages and rechecks the measurement regularly when the IC is in use (preferably at every startup). Such PUF technology could serve as a highly effective protection against many types of reverse engineering and fault attacks.

**About NXP**

Building on trusted security, a complete product portfolio and the best contactless performance, NXP is the leader in the overall ID market and in key market segments such as transport ticketing, eGovernment, access, infrastructure, RFID/Authentication, payments and NFC. NXP provides the entire ID market with end-to-end solutions, enabling customers to create trusted solutions for a smarter life.