Design of a Security-Enhanced UHF RFID Chip

Design of a Security-Enhanced UHF RFID Chip

Master's Thesis

at

Institute for Applied Information Processing and Communications
Graz University of Technology

Integrated Systems Laboratory
Swiss Federal Institute of Technology

submitted by

Johann Ertl

johann.ertl@student.tugraz.at

March 2013

Advisors: ETH Zürich: Dr. Norbert Felber, Dr. Luca Henzen
TU Graz: Dr. Martin Feldhofer, Dr. Thomas Plos
Assessor: Prof. Dr. Karl-Christian Posch

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

TU Graz
Graz University of Technology

Abstract

In the past few years, radio-frequency identification (RFID) has become omnipresent in many everyday applications. Contactless ticketing, access systems, payment systems, electronic passport, near-field communication (NFC), or electronic immobiliser are only a few applications using RFID technology. A huge market for automatic identification is supply-chain management. Advances in integrated-circuit (IC) technology make RFID labels cheap enough to replace the current barcode system in many use-cases. The Electronic Product Code (EPC) C1G2 standard, developed for supply-chain applications, is a high-performance UHF RFID standard that allows operating ranges up to $10\,\mathrm{m}$ and an inventory speed up to 500 tags per second. A main drawback of the standard are its weak security properties.

Possible access of unauthorised readers allows data manipulation and it is easy to clone tags by copying the EPC value. Since the EPC value is a unique identifier privacy issues arise. Strong authentication features can prevent forgery and data manipulation but implementing cryptographic algorithms on passive low-cost tags is challenging due to the fierce constraints regarding maximum chip-area usage and power consumption.

This work implements the digital part of an EPC C1G2-compliant tag with security enhancements based on a low power and low-area Advanced Encryption Standard (AES) implementation. It suggests a mutual authentication procedure between reader and tag using a standard challenge-response protocol. Secure challenge generation is achieved by using the lightweight stream cipher Grain. Randomisation techniques during execution increase resilience against side-channel analysis attacks. The design has been produced as a prototype chip in a $130\,\mathrm{nm}$ standard-cell process. In order to evaluate the randomisation countermeasures two versions were fabricated on one die. The resulting design has a complexity of $12\,000\,\mathrm{GE}$ which fits on less than $1/10\,\mathrm{mm}^2$ die area excluding the bonding pads. Simulation shows an average power consumption during one full authentication round of less than $5\,\mu\mathrm{W}$. Hence, both values area and power consumption meet the constraints of low-cost passive RFID tags.

Strong cryptography is possible on low-cost passive RFID tags. Improved security properties of large-scale RFID systems not only make them more reliable but will also increase the acceptance of the consumer.

Keywords: Radio-Frequency Identification (RFID), Electronic Product Code (EPC), Ultra-High Frequency (UHF), Advanced Encryption Standard (AES), Grain, Application-Specific Integrated Circuit (ASIC), Mutual Authentication, Side-Channel Analysis, Low-Power Design

Kurzfassung

In den letzten Jahren hat Radiofrequenzidentifikation (RFID) in vielen alltäglichen Anwendungen Einzug gefunden. Kontaktlose Tickets, Zutritts- und Zahlungssysteme, elektronischer Reisepass, Nahfeldkommunikation (NFC) oder elektronische Wegfahrsperre sind nur einige Anwendungen von RFID Technologie. Ein riesiger Markt für automatische Identifikation ist das Versorgungskettenmanagement. Fortschritte bei integrierten Schaltungen (ICs) machen RFID-Etiketten billig genug, um das aktuelle Barcode-System in vielen Anwendungsbereichen zu ersetzen. Der Elektronische Produktcode (EPC) C1G2 Standard, entwickelt für genau diese Anwendungen, ist ein UHF-RFID-Standard, der eine Betriebsreichweite von bis zu $10\,\mathrm{m}$ erlaubt und bis zu 500 Transponder pro Sekunde auslesen kann. Eine große Schwäche des Standards sind seine geringen Sicherheitseigenschaften.

Der Zugriff unberechtigter Lesegeräte erlaubt Datenmanipulation. Weiters ist es leicht Transponder durch Kopieren des EPC-Wertes zu klonen. Auf Grund der Eindeutigkeit des EPC-Wertes ergeben sich potentielle Verletzungen der Privatsphäre. Starke Authentifizierungsmechanismen können Fälschungen und Datenmanipulation zwar verhindern, allerdings ist das Implementieren von kryptographischen Algorithmen auf Transpondern eine große Herausforderung, da Einschränkungen hinsichtlich maximaler Chipfläche und Stromverbrauch vorliegen.

Diese Arbeit implementiert den digitalen Teil eines EPC C1G2-konformen RFID Transponders mit erhöhter Sicherheitsfunktionalität, basierend auf einer leistungs- und flächenoptimierten AES Implementierung. Es wird eine gegenseitige Authentifizierung zwischen Lesegerät und Transponder, basierend auf einem standardisierten Aufforderungs-Antwort Protokoll, verwendet. Kryptografische Sicherheit der Aufforderungsgenerierung wird durch den Stromchiffre Grain gewährleistet. Randomisierung während der Ausführung erhöht die Widerstandsfähigkeit gegen Seitenkanalattacken. Das Design wurde als Prototyp in einem $130\,\mathrm{nm}$ Standardzellen-Prozess produziert. Um die Randomisierungstechniken zu evaluieren wurden zwei Varianten des Designs hergestellt. Der Schaltkreis hat eine Komplexität von $12\,000\,\mathrm{GE}$, was ohne Verbingungsanschlüsse einer Chipfläche von weniger als $1/10\,\mathrm{mm}^2$ entspricht. Simulationen zeigen eine durchschnittliche Leistungsaufnahme während einer vollständigen Authentifizierung von weniger als $5\,\mu\mathrm{W}$. Sowohl Chipfläche als auch Leistungsaufnahme erfüllen die Einschränkungen für preiswerte, passive RFID Transponder.

Starke Kryptographie auf preiswerten/passiven RFID Transpondern ist möglich. Es verbessert die Sicherheitseigenschaften von großen RFID Systemen, macht sie zuverlässiger, und erhöht auch die Akzeptanz der Endverbraucher.

Stichwörter: Radiofrequenzidentifikation (RFID), Dezimeterwellen (UHF), Elektronischer Produktcode (EPC), Advanced Encryption Standard (AES), Grain, Anwendungsspezifischer Integrierter Schaltkreis (ASIC), Gegenseitige Authentifizierung, Seitenkanalanalyse, geringe Leistungsaufnahme

Deutsche Fassung:
Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008
Genehmigung des Senates am 1.12.2008

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am …… ……..
(Unterschrift)

Englische Fassung:

STATUTORY DECLARATION

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

…… ……..
date (signature)

Contents

List of Figures

List of Tables

Preface

Radio-frequency identification (RFID) is a broad term for various everyday applications. For example, ticketing, payment and access systems, electronic passport, NFC, or car immobiliser use RFID technology. Probably the biggest market is going to be supply-chain management. Automatic identification of products and goods throughout a globalised economy using cheap RFID labels can increase the efficiency of logistic management and retail business. The EPCglobal organisation drives the development of universal standards for this RFID use case. The Electronic Product Code (EPC) is a unique identifier for possibly all objects traded globally. In combination with the EPC Class-1 Generation-2 (C1G2) RFID air-interface standard and decreasing prices of integrated circuits (ICs), it is expected to replace the current barcode system in many applications.

The EPC C1G2 standard is a high performance, passive UHF RFID communication standard. It allows an operating range of several meters and can read a large number of tags per second for fast and convenient tracking of labelled goods. Large quantities of RFID tags allow low unit costs of under $0.05\,\$$ per label. A main weakness of the standard are its limited security features. Sensitive data during transmission is only protected in the reader-to-tag link. The absence of authentication mechanisms allows unauthorised readers to manipulate tag data. It is also possible to copy EPC values and user data to an empty tag and clone the supposedly worldwide unique tag. More advanced security features can prevent sabotage of large RFID systems, impede data manipulation, and identify product forgery. Furthermore, privacy issues posed by the uniqueness of the EPC value can be defused with more advanced security features on the tag.

The hardware resources on an RFID tag are limited. The passive power supply over the RF field limits the maximum power consumption. The maximum chip area for the circuitry of the tag is mainly limited by economic factors. The price of an RFID label depends heavily on the die size of the IC. In order to compete with the cheap barcode systems, every cent difference in unit production costs of the electronic labels decides if the benefits of the RFID system outweigh the additional costs. Therefore, most suggestions for additional security features of the EPC C1G2 standard propose lightweight protocols based on the hardware resources already present on a tag. However, the weak cryptographic properties of the standard-compliant pseudorandom number generator (PRNG) or the linear cyclic redundancy check (CRC) results in complicated authentication protocols whose security is difficult to evaluate. A lot of suggestions have already been broken.

This work implements a security-enhanced EPC C1G2 standard compliant digital controller using strong symmetric cryptography. An interleaved protocol design allows the usage of a low power and low-area Advanced Encryption Standard (AES) implementation. Together with cryptographically secure pseudo-random numbers, mutual authentication is possible through custom user commands. The final design is synthesised and produced in a $130\,\mathrm{nm}$ standard-cell ASIC process. Simulation results show an area usage of $12\,000\,\mathrm{GE}$ and an average power consumption of $4.7\,\mu\mathrm{W}$. Both results meet the fierce constraints of low-cost passive UHF RFID systems.

This thesis is structured as follows: **Chapter 1** provides an overview of RFID in general. It describes the main components, used frequencies and power-supply technologies, and example applications. **Chapter 2** discusses specifically the EPC C1G2 standard. After giving some background infor-

mation and design criteria, it explains the communication flow between reader and tag. This includes encoding in both directions, command structures and header information, tag states, and the tag's internal design and memory structure. **Chapter 3** provides background information about security and privacy issues in RFID systems in general and specifically in the current version of the EPC standard. After a summary of different suggestions for security enhancements of the current standard, the approach of this work is presented: A symmetric challenge-response authentication protocol based on AES and Grain as PRNG. All parts including communication flow, custom command structure, AES, and Grain are described.

After specification of the security enhancements, **Chapter 4** illustrates the architecture of the baseband system. Different design approaches are evaluated and block diagrams of all parts are described. Different aspects of the process from a defined system architecture to a tape-out-ready implementation are presented in **Chapter 5**. This includes for example, verification and test setup, low-power methods, synthesis and back-end design, and simulation results. It also discusses vulnerabilities of the AES implementation through side-channel analysis and possible countermeasures. Finally, the results are compared to related work on EPC standard baseband systems without or with additional security circuitry.

Chapter 6 provides a final summary of the work including strengths and weaknesses of the implementation choices, experiences during the development process and possible improvements. The work finishes with a short outlook on additional research necessary in order to bring strong security features to EPC RFID systems.

A datasheet of the produced test samples of the prototype chip are provided in **Appendix A**. This includes a brief summary of features, a description of the operation modes, usage information, memory maps of the RAMs, and pin-configuration/description.

Chapter 1

Radio-Frequency Identification

In the past few years radio-frequency identification (RFID) has been on its way to replace more and more automatic identification (auto-ID) procedures, such as barcode systems, ticketing, passports, and smart-card applications. The main advantages over optical or contact-based systems are usually higher effectiveness, security and more convenience. The idea to identify objects using radio technology over short to medium distances is more than 50 years old, but only recent progress in integrated circuit (IC) production has made RFID technology cheap enough for widespread usage. The global market increased from less than 1 billion $US before 2000 to 5.5 billion $US in 2009 and will probably exceed 10 billion $US by the year 2014 [Liard and Carlaw, 2009]. An RFID system typically consists of two main parts as shown in Figure 1.1, which are an RFID tag and RFID reader. The tag or transponder is attached to the object of interest and exchanges information with the reader or interrogator over the air using radio technology. Both parts have an antenna attached to exchange data and optionally supply the tag with power and a clock signal. In most RFID systems the reader device connects to a back-end system to receive information from, or forward tag data to a system application. The following sections will provide a brief overview over RFID tags and readers, frequencies used, reading ranges, and common applications.

1.1 RFID Tag

The tags or transponders are the main component in an RFID system. They store the information that the auto-ID system needs for identifying the object or person, to update a logistic system, allow or deny entrance, or process a payment. It consists of a microchip attached to an antenna. The power supply defines a first categorisation of RFID transponders. Tags can operate active, semi-passive and passive [Finkenzeller, 2010].

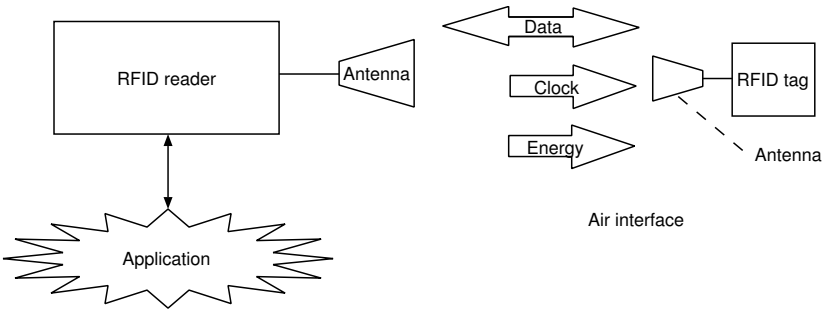


Figure 1.1: Overview of an RFID system.

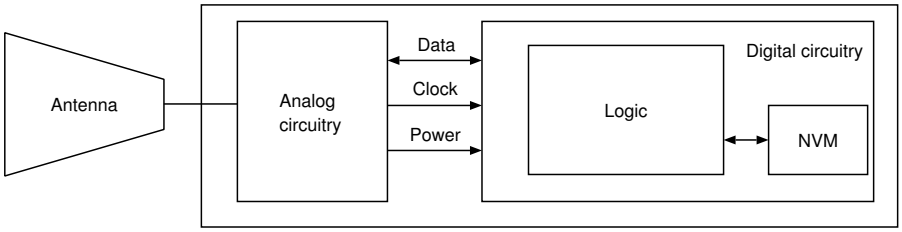


Figure 1.2: Architecture of a passive RFID tag.

- **Active tags:** The tag has its own power supply, usually a battery. The main advantage of active tags is their ability to send information actively without the presence of a strong reading-device RF field. Independent power supply usually enables higher transmission distances than passive systems. The lower power constraints to the chip allow integrating more complex tasks, such as cryptographic operations or environment sensors. The disadvantages of a battery-based power supply of each tag are high production costs, bigger size of the transponders, and higher maintenance expenses. RFID systems usually need very high unit numbers of tags and therefore costs per tag are crucial for widespread usage. Batteries have a limited lifetime and so the effort to detect and replace defect items is comparable high. Nowadays, active tags are typically used in systems where they provide additional functionality like monitoring and processing sensor data, or provide higher transmission ranges.

- **Semi-passive tags:** Similar to active tags the power supply for the chip is provided by a battery attached to the tag, but the communication between the reader and the tag works like for a passive tag. The semi-passive tag does not send actively but waits until it is close enough to the reader and transmits data via load modulation or backscatter (see section 1.3). Advantages and disadvantages are similar to active tags but with the same limitations in reading range like passive tags.

- **Passive tags:** These tags need no inbuilt power supply. The energy for the analog interface and also the digital processing unit comes from the field emitted by the reader device. This concept allows to build very compact, robust, and cheap transponders which is crucial for applications, such as logistics, where the unit numbers are very high. Only within a certain distance to a reading device the chip on the tag receives enough power supply and can process or transmit information. In order to produce high unit numbers at reasonable costs these transponders have strong constraints regarding chip size. Since the power supply over the air is limited the tags have to fulfil severe power-consumption constraints. Due to higher integration and decreasing power consumption of modern IC processes these tags are most widespread in RFID systems.

Figure 1.2 shows an overview of a passive RFID tag. The requirements vary greatly depending on the application, but basically an RFID tag consists of an antenna or coupling element, an analog circuitry, and a digital circuitry including a logic unit as well as nonvolatile memory (NVM). The size and shape of the antenna depends on the operating frequency. Systems working in the near field use a coil as coupling device and far field systems optimised dipole antennas. The analog circuitry supplies the digital part with stable power retrieved from the reader field. It has to provide a clock signal and demodulates data sent from the reader. For data transmission to the reader this circuit uses load modulation by changing the impedance of the coil, or backscatter by changing the reflection coefficient of the dipole antenna. The complexity of the digital part and the size of the memory depends heavily on the application. We distinguish low-end, mid-range, and high-end systems [Finkenzeller, 2010].

- **Low-end systems** like electronic article surveillance (EAS) systems or read-only transponders have only 1 bit up to a few bytes permanently encoded data and when they enter the RF field of a reader they start to broadcast their serial number.

- **Mid-range systems** provide a few byte up to 100 kB memory. The logic part of the digital circuitry implements decoding and handling of multiple reader commands, anti-collision procedures and read/write handling of the included NVM. The digital design of this work falls in this category and additionally implements cryptographic commands for mutual authentication.
- **High-end** RFID tags implement a microprocessor with sufficient memory to store a smart-card operating system and application data. These systems provide the same functionality as contact-based smart cards, such as bank, ID, payment, or ticketing cards.

1.2 RFID Reader

Every RFID application that processes the information from the transponders needs a reading device, which handles the power supply of passive tags and the communication to one or more tags in reading distance. RFID readers have much lower power and cost constraints than tags because they have their own power supply and their unit numbers are much lower than those of tags in most use cases. Besides providing a strong-enough RF field, with one or more antennas, readers handle all communication features like establishing a connection, performing anti-collision and authentication procedures, and forward the information to the application [Finkenzeller, 2010]. Readers typically implement several protocols and can handle various types of tags in order to increase compatibility between different RFID systems. Most readers are locally fixed devices at places relevant for the application, so power consumption and size are secondary design criteria. Only recent developments like handheld readers and near-field communication pushes the development for higher integration and results in single-chip reader ICs.

1.3 Frequencies, Coupling, and Reading Range

For contactless data transmission RFID systems use radio waves with a wide frequency range ($135\,\mathrm{kHz} - 2.45\,\mathrm{GHz}$) and operates in so called unlicensed industrial, scientific, and medical (ISM) spectrum space. Table 1.1 shows the main characteristics depending on the frequency band. The four main frequency bands are: low frequency (LF), high frequency (HF), ultra-high frequency (UHF), and microwave. A main difference between LF and HF system to UHF and microwave systems is the way how the coupling between the reader and the tag is realised:

- LF and HF systems use **near-field** coupling, where the wavelength is greater than the size of the tag antenna and the reading distance. The reader generates a magnetic field which induces an electric current in the tag's antenna. For the communication to the reader the tag changes the impedance of its antenna coil and the resulting change of current drawn can be detected by the reader (load modulation). The maximum reading range of these systems is limited from a few cm up to at most $1.5\,\mathrm{m}$. Fluids do not shield the communication. Power constraints to passive tags are lower and they have to deal with less interference from other readers or EM sources.
- UHF and microwave systems operate in the **far field** where the wavelength is smaller than the average reading distance. Data transmission from tag to reader is done by changing the reflection coefficient of the tag antenna and the difference in reflection of the continuous EM waves is detected by the reader (backscatter). UHF systems allow higher data rates and read ranges but these systems have to cope with lower power supply of the tag, higher interference from other systems at similar frequencies and are more sensitive to reflection/disturbance from obstacles near the reader.

Operating ranges of RFID systems depend on various factors. Active tags do not need a reader RF field because of their independent power supply and usually enable higher read ranges. Reading ranges

Frequency band	LF	HF	UHF	Microwave
Typical RFID frequencies	$125 - 134\,\mathrm{kHz}$	$13.56\,\mathrm{MHz}$	$433\,\mathrm{MHz}$ $865 - 956\,\mathrm{MHz}$	$2.45\,\mathrm{GHz}$
Approximate range	up to $0.5\,\mathrm{m}$	up to $1.5\,\mathrm{m}$	up to $7\,\mathrm{m}$	up to $10\,\mathrm{m}$
Data-transfer rate	Low	High	Medium	Medium
Coupling	Near field	Near field	Far field	Far field
Penetrates: water metal	Yes No	Yes No	No No	No No
Common usage	Animal ID, car immobiliser	Smart labels, contact-less smart cards	Logistics	Moving vehicle toll
Protocols and standards	ISO 11784/5 14223 ISO18000-2 HiTag	MIFARE ISO14443 TAG-IT TIRIS	ISO18000-6A,B,C EPC™class 0 and 1 AAR S918 Ucode	ISO18000-4 Intellitag μ-chip

Table 1.1: RFID operating frequencies and characteristics based on [Matt et al., 2006].

from passive tags depend mainly on their power consumption and frequency range. Ultra-high frequency systems with far-field coupling enable greater operating ranges than inductive coupled systems that work in the near field only.

1.4 Example Applications

The main criteria for a widespread usage of an RFID application is the simple economic principle of identifying objects and persons. The cost for identification must be much smaller than the object is worth, or the economic gain from automating the identification process. So, successful RFID systems involve either expensive goods, provide additional features to other auto-ID systems, or realise very cheap tag technologies [Dobkin, 2008]. The following sections will provide several examples of successful RFID applications.

1.4.1 Contactless Smart Cards

Almost everybody has smart cards in the wallet that have powerful ICs integrated and work contact based. The microprocessor and sufficient memory allow to run software on the card and realise many different applications including high-security tasks. Bank, customer, telephone and ID cards for example make use of this technology. The main disadvantage regarding durability and handling of these cards is their contact-based interface. Proximity-coupling RFID solutions like ISO 14443 provide full smart card functionality but contactless communication allows more robust cards, more compact readers, and easier handling. The breakthrough happened in the mid 1990s when lower power consumption of the silicon chips allowed transponders to use $13.56\,\mathrm{MHz}$ instead of $135\,\mathrm{kHz}$ operating frequency, which needs less windings and allows standard credit-card format implementations. First widespread usage of contactless smart cards in Germany was the customer loyalty card by Lufthansa AG initially issued 1995 [Finkenzeller, 2010]. Some other examples of applications for contactless smart cards:

- **Public Transport:** Contactless smart cards including payment function ability are an ideal way to improve ticketing for public-transport systems. The costs for paper-ticket printers are very high. Ticket sales through the driver cause security risks through distraction, long waiting time for the

passengers, and need additional staff for ticket control preventing fare dodgers. Electronic reading-devices in public transport are often exposed to dirt, humidity, and potential vandalism. Therefore, compact contactless devices are more durable than other systems. Seoul, the capital of South Korea started to install a full-coverage contactless-ticketing system in its public-transport system including buses and underground railway in 1997. London successfully uses RFID technology for ticketing and access control to the underground. With broader coverage of NFC-enabled mobile devices public transport will become an even more important application for contactless smart-card systems.

- **Contactless Payment Systems:** Magnetic stripe on bank and credit cards have been the main technique for fast identification in payment processes for a long time, but they provide no security against forgery and are not very durable. Contactless smart-card implementations provide high security features combined with high customer convenience and fast payment processes. All big credit-card companies have brought contactless payment systems to market. MasterCard® issued Paypass in 2003, ExpressPay by American Express® in 2005, and Visa® Contactless 2006 are all credit-card products integrating wireless technology.

- **Access Control:** In many big institutions where the access of a lot of people has to be authorised, electronic control systems replaced the classical key because of their superiority regarding security and flexibility. Again, handling contact or optical-based key cards is much more inconvenient than comparable RFID solutions.

1.4.2 Animal Identification

One of the first widespread RFID applications was animal identification applied for internal usage, such as automatic feeding, or productivity measuring, as well as for external usage like quality control, inter-company tracking, and epidemic control. Transponders work in the LF range where EM waves also penetrate water and therefore can also be implanted under the skin [Finkenzeller, 2010].

1.4.3 Ski Tickets

Especially in Austria one of the first RFID applications used and noticed by a lot of people in everyday life were contactless ski-ticketing systems. The rough environmental circumstances like cold temperature and high humidity make contactless access control at the ski lift superior to any other auto-ID system. Also the handling of tickets is very inconvenient for the customers if they have to take it out of their pockets every time when they are wearing gloves. Customers purchase RFID-enabled tickets and the reading range is sufficient that the entrance system at the lift can detect a customer with a valid ticket in his anorak pocket when he approaches the entrance gate. Ski ticketing is a good example for early adoption of RFID technology even at a time when transponders were still quite expensive because in this use case it is superior to cheaper systems and provides enough economic gain.

1.4.4 Near-Field Communication (NFC)

NFC is a technology that allows to add a flexible RFID interface to electronic devices like mobile phones. An NFC device can emulate a passive contactless smart card as well as an active reader device. The development started in 2002 by NXP and Sony with the idea to combine several RFID applications into one device. NFC-enabled smartphones for example can be used for applications like ticketing, entrance control, payment systems, and data exchange between two phones, or reading passive tags embedded in smart posters [Finkenzeller, 2010].

1.4.5 Electronic Immobilisation

In order to prevent unauthorised commissioning of a car, automobile companies started to integrate LF RFID transponders into the ignition key. The lock in the car also contains an RFID reader. Reader and transponder share a secret key. If a wrong ignition key is used the engine does not start. After the introduction of these immobiliser systems to all new cars in 1995 the theft rate declined by a factor of 40 [Finkenzeller, 2010].

1.4.6 Electronic Passport

By 2006 all EU countries and several other countries introduced the electronic passport (ePass). The passport cover contains a high-end RFID tag which provides not only sufficient memory to store person-related data including biometric information but also implements high cryptographic functions. Besides faster person control at airports the main idea of the ePass is to prevent forgery. The reading distance is about $10\,\mathrm{cm}$ and only authorised readers have access to the stored information. Data integrity and authenticity is secured by a digital signature [Finkenzeller, 2010].

1.4.7 Supply-Chain Management

In today's complex and globalised economy tracking goods during production, transportation, and distribution until sale to the final customer is a very challenging task. Often many different companies from various countries all over the world are involved. Automation of this task reduces a lot of unnecessary manpower, decreases error rates and stocktaking costs, and is base for a more efficient logistics with lower storage costs. As stated at the beginning of this section RFID technologies are only employed if the costs for tagging is much smaller or negligible compared to the value of the item itself.

In the 1970s when RFID tagging was very expensive the American rail industry started to track their railcars with RFID technology. Another early example of very expensive goods, respectively goods collections where high tagging costs did not matter, was tracking of shipping containers. Early systems used active transponders to identify and track containers worldwide. In the early 1990s the development of much cheaper passive tags enabled tracking of pallets, boxes, or single items. Tag-IT from Texas Instruments and U-code developed by NXP were widespread used in tracking packages, and single items within one organisation. Big manufacturers tracked their expensive items during the production line, huge distribution centres started using RFID technology, and retailers tried to optimise their logistics. Still fairly high costs of the tags and different incompatible standards and implementations did not allow a general application throughout the supply chain involving different companies or institutions [Dobkin, 2008].

With the standardisation of the Electronic Product Code (EPC) and EPC Class 0 and Class 1 tags, the vision of uniformed labelling of all goods in worldwide supply chain came a big step closer. Wal-Mart forced suppliers to label every case and pallet with EPC labels by 2006 and retailers like Tesco, Metro, and Target followed. With further decrease of costs for EPC RFID tags tracking of single goods and a complete replacement of the barcode system might be possible in the near future. This scenario, often referred as 'the Internet of things', will likely be the biggest market for RFID systems in the future. Chapter 2 will provide more information about the history and development of the EPC and the compatible tag standard.

Chapter 2

EPC Class-1 Generation-2 Standard

The Electronic Product Code (EPC) Class 1 Generation 2 (C1G2) or ISO/IEC 18000-6C standard is currently the most used standard for passive RFID systems operating in the $860\,\mathrm{MHz} - 960\,\mathrm{MHz}$ frequency range. It was released 2004 by EPCglobal with the goal to eliminate the shortcomings from generation-one implementations and to provide enough flexibility in one specification in order to meet the requirements for most applications relying on the EPC. Besides the physical and logical requirements for successful communication between reader and tag it also defines the tag's memory structure and internal states. The following sections in this chapter provide a brief description of the standard's history, a short explanation of the EPC itself, and discuss requirements for a successful EPC standard. Moreover, this chapter summarises the main points from the standard specification [EPCglobal, 2008]: reader-to-tag communication, tag-to-reader communication, commands and tag states, internal memory structure, anti-collision procedure, and memory-access commands.

2.1 History

After several proprietary and incompatible RFID systems for item tracking to support supply-chain management and logistics in the 1990s, MIT researcher David Brock came up with the idea to uniquely label every produced object. Together with colleagues they developed the Electronic Product Code (EPC) which enables to assign a standardised and unique code to every manufactured product. In order to store and process the EPC codes automatically an RFID solution seemed an obvious choice. Optical systems like the barcode are much cheaper because of the simple printed labels but provide very limited amount of data storage and functions, are very inflexible, and usually the reading of the information cannot be fully automated. In order to push the research and development of suitable RFID solutions for this task the Auto-ID Center was founded in 1999.

During the next 3 years many private corporations and research institutions joined the Auto-ID Center and explored ways to realise an ubiquitous RFID system for item tracking. In order to utilise the EPC information a standardised infrastructure was being considered. A service similar to the Domain Name Service (DNS) should provide location information of an item, whose EPC information is known. A markup language defines the way to describe the properties of an object connected to an EPC. Potential labelling of every produced item results in huge tag-unit numbers and therefore the main precondition for successful introduction of an RFID solution is the cost for the electronic labels. EPC tags should be as simple as possible and avoid complex anti-collision protocols, additional memory beyond the EPC, or error correction. As operating frequency a UHF system in the $900\,\mathrm{MHz}$ range turned out to be the most suitable regarding costs, read range, and capability. These activities resulted in the first-generation air-interface standards for Class 0 and Class-1 tags in 2001.

2003 the non-profit organisation EPCglobal Inc. was founded to further promote supply-chain RFID

standards. The first-generation standards used the same RF bands but differed not only in features, but were also incompatible in modulation and encoding. Therefore, within one year of development a second generation air-interface standard was released. Main design criteria were to define one standard that covers most applications without introducing incompatibilities and also considers existing systems [Dobkin, 2008]. The EPC Class-1 Generation-2 standard is now the most used RFID communication protocol in supply-chain management applications.

2.2 Electronic Product Code

Defined in the EPCglobal Tag Data Standard 1.6 [EPCglobal, 2011], the EPC is a unique identifier for any physical object. The design criteria besides compatibility to existing identifiers and standards, used with the current barcode system, were flexibility for future demands and focus on usage of RFID technology as data carrier. On application level it has the form of a Uniform Resource Identifier (URI) called Pure Identity EPC URI, but because memory on RFID tags is costly there also exists an efficient binary encoding for the storage on the labels. The binary encoding starts with a fixed 8-bit header which defines the overall length of the EPC. The current standard defines 14 different encoding schemes with a minimum length of 96 bits.

2.3 Requirements for an EPC RFID Standard

As previously mentioned an item-tracking RFID system has to provide **several meters reading range** and should use **cheap** and therefore **passive tag technology**. Only far-field implementations come into consideration and systems in the $900\,\mathrm{MHz}$ range have to deal less with interferences as systems in the $2.4\,\mathrm{GHz}$ range, because heavily used wireless communication networks like bluetooth and WiFi also operate in this range. The exact regulations of the unlicensed industrial, scientific, and medical (ISM) frequencies vary depending on the country respectively region (USA $500\,\mathrm{kHz}$ channels in the $902 - 928\,\mathrm{MHz}$ range, Europe $200\,\mathrm{kHz}$ channels in the $865 - 868\,\mathrm{MHz}$ range). Even though RFID systems are fairly short-range technologies and usually are only used indoor, the EPC C1G2 standard has to deal with effects like diffraction and reflection. This results in an unreliable and non-continuous connection between reader and tag and also changes the signal strength depending on the current obstacles present in the RF field. Antennas used in the unlicensed spectrum are limited to $6 - 10\,\mathrm{dBi}$ gain which is another obstacle for a stable link. The main source of interferences are other readers because the tag-to-reader signal is about $50 - 60\,\mathrm{dB}$ below the reader-to-tag signal [Dobkin, 2008].

The EPC C1G2 standard tries to reduce this problem by defining a **dense interrogator mode** which defines the minimum attenuation of the reader signal in neighbouring channels. Modulation for the reader-to-tag communication is limited to amplitude-shift keying because other modulations are too complex for passive tags. A high average reader-power transmission is guaranteed by choosing an enconding with short amplitude-low times. The tag supports **two types of frequency-shift keying encodings** (FM0 and Miller) and **flexible data rates** which allows the reader placing the spectrum of the tag backscatter signal to a low interference channel.

The reader is always the master in the communication and sets all downlink and uplink parameters. Every reader command is therefore prepended by a sync sequence, which sets all parameters for the following communication.

In order to cope with the unstable link in a passive UHF system most commands append a CRC-5 or CRC-16 checksum to provide data integrity. **Persistent tag flags** allow distinction between already read tags and new tags during an inventory round, even if tags are not continuously supplied with sufficient power. Those flags also enable simultaneous interaction with multiple readers.

In many supply-chain management applications often a large number of tags enter the reading range

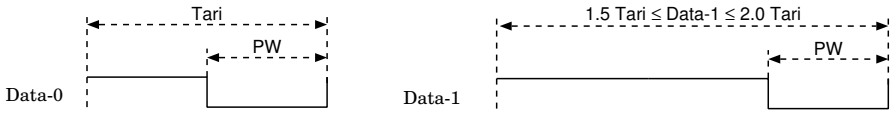


Figure 2.1: Representation of Data-0 and Data-1 using PIE.

of a reading device (for example a pallet with hundreds of single items moves through a checkpoint). The EPC C1G2 standard uses a **slotted Aloha protocol** for tag singulation. In the first generation of the standard, a part of the EPC value was the basis for an anti-collision algorithm but as unprogrammed tags or tags with identical EPC are possible the generation-2 standard uses pseudo-random numbers on the tag to allow fast inventory in all cases.

A **defined tag memory structure**, including read/write/lock procedures, increases the compatibility between different implementations.

2.4 Reader-to-Tag Modulation and Encoding

For communication to one or more tags in the field the reader modulates an RF carrier using amplitude-shift keying (ASK) modulation. Other modulations like frequency or phase-modulated signals would be too costly for a passive low-cost tag to demodulate. The reader can choose between single sideband, double sideband, or phase-reversal ASK. Since the tag also receives the power from the RF field the choice of the encoding must consider the average power transmission of the modulated signal. Pulse interval encoding (PIE) as shown in Figure 2.1 uses the bandwidth inefficiently, but the periods with low RF field are minimised. The tag can therefore also extract enough power from the field during reader-to-tag data transmissions. Tari ($6.25 - 25\,\mu$s) is the reference time interval representing Data-0 with a pulse width (PW) between 0.265 to 0.525 Tari (minimal $2\,\mu$s). Data-1 is represented by a time interval of $1.5 - 2.0$ Tari and the same PW.

For every inventory round the reader defines the down and uplink data rate encoded in a synchronisation frame prepended to all reader commands. The first command of every inventory round (Query, see Section 2.7) uses a preamble as shown in Figure 2.2 the other commands use a frame sync which does not contain an T⇒R calibration value because the uplink parameters do not change during a session. Both synchronisation frames start with an initial delimiter ($12.5\,\mu\text{s} \pm 5\,\%$), followed by a Data-0 symbol, and a reader-to-tag synchronisation symbol (RTcal) with a length of $2.5 - 3.0$ Tari. The tag uses $\text{RTcal}/2$ as a pivot value to interpret further reader data symbols (Data-0 $<$ $\text{RTcal}/2$ $<$ Data-1). In order to set the backscatter link frequency a tag-to-reader synchronisation symbol (TRcal) is transmitted in case of a Query command. The back-link frequency (BLF) can be determined by $\text{BLF} = \text{DR}/\text{TRcal}$ with $\text{DR} = 8$ or $64/3$ as specified in the header of the Query command.

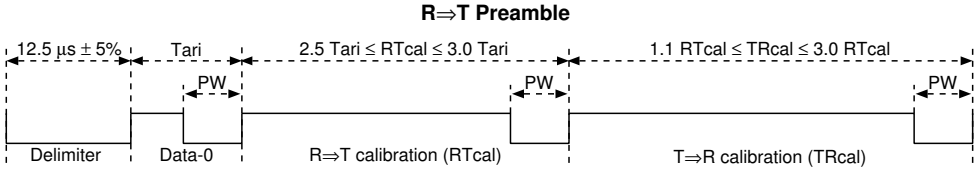


Figure 2.2: Synchronisation frame at the beginning of a reader command (Preamble).

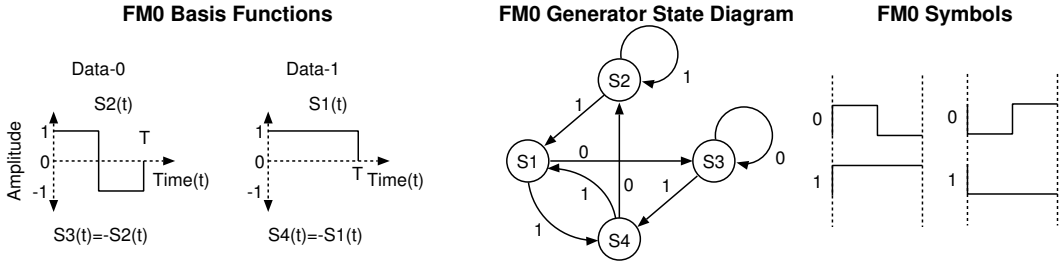


Figure 2.3: FM0 basic function, generator state diagram, and symbols.

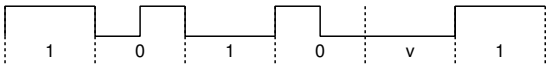


Figure 2.4: FM0 preamble.

2.5 Tag-to-Reader Modulation and Encoding

A tag sends data to the reader using backscatter modulation. It switches its reflection coefficient depending on the data between two states and can choose between ASK or PSK as modulation format. The reader selects the BLF with the TRcal time period and the encoding (FM0 or Miller). Depending on the BLF and encoding the data rate is between $40\,\mathrm{kbps} - 640\,\mathrm{kbps}$. Independent of the modulation, the reader cannot detect the amplitude or phase state of the backscatter signal accurately because increased signal power of the backscatter signal can lead to a decreased reader signal. Therefore, both encoding schemes in the EPC C1G2 standard are frequency-shift keying based since the reader can only reliably detect if a transition occurred or not [Dobkin, 2008].

Figure 2.3 shows the FM0 basis function, the generator state diagram, and symbols. At every symbol boundary the phase changes. The Data-0 symbol has an additional mid-symbol phase inversion. Every tag-to-reader frame starts with a preamble, shown in Figure 2.4. In noisy environments the reader can demand an additional pilot tone before the preamble, which consists of 12 Data-0 symbols. Every frame ends with a dummy Data-1 symbol.

The reader may demand Miller encoding for the tag-to-reader communication that uses $2 - 8$ sub-carrier cycles per bit. This increases interference rejection but at the cost of lower data rates. Hence, a reader device can make environment-depending noise to data-rate trade-offs in dens-interrogator environments [Dobkin, 2008]. Figure 2.5 shows the Miller basis function and generator diagram. It inverts its phase between two Data-0 symbols and makes a phase inversion in the middle of a Data-1 symbol. In the Query command the reader sets 2, 4 or 8 sub-carrier cycles per bit (M). Therefore the resulting data rate is between $5\,\mathrm{kbps} - 320\,\mathrm{kbps}$. Figure 2.6 shows the Miller preamble that starts every tag-to-reader frame. It starts with an unmodulated sub carrier for a period of $4\,\mathrm{M/BLF}$ followed by a 010111 sequence. Optionally, the reader can demand a longer unmodulated sub carrier sequence ($16\,\mathrm{M/BLF}$). Every frame ends with a Data-1 bit.

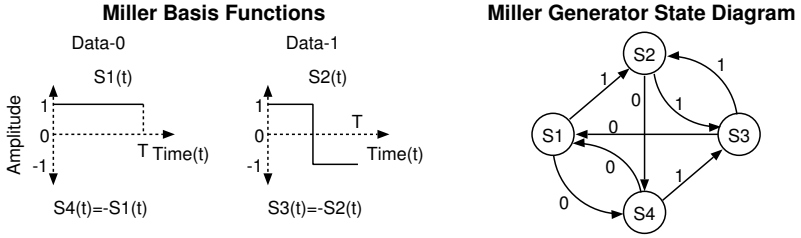


Figure 2.5: Miller basic function and generator state diagram.

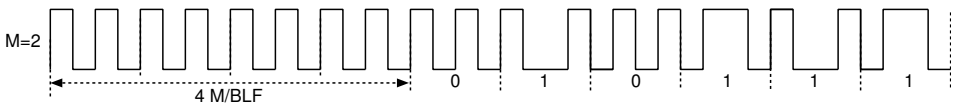


Figure 2.6: Miller preamble with two BLF cycles per symbol ($\mathrm{M} = 2$).

2.6 Tag Memory Structure

In order to increase compatibility between different implementations the standard also specifies the tag-internal memory structure and how to address the memory. The non-volatile memory is separated into four logical blocks (called banks):

- **Bank 00: Reserved memory**. Contains the password for the kill command and a possible access password. The kill password is stored at $00_\mathrm{h} - 1\mathrm{F}_\mathrm{h}$ and the access password at the address space $20_\mathrm{h} - 3\mathrm{F}_\mathrm{h}$. This is also the memory space to store private keys for strong authentication.
- **Bank 01: EPC memory**. Contains an EPC value as briefly described in Section 2.2. At the beginning of the memory block ($00_\mathrm{h} - 1\mathrm{F}_\mathrm{h}$) the tag stores the CRC-16 of the EPC memory, followed by the Protocol Control (PC) value ($20_\mathrm{h} - 3\mathrm{F}_\mathrm{h}$) which describes the format of the EPC stored in this tag. The PC value also informs about an optional XPC value at the end of the memory block which provides more information about additional tag functionality like recommissioning or security features.
- **Bank 10: TID memory**. This blocks starts with an 8-bit ISO/IEC 15963 allocation class identifier and stores information for the reader about possible custom commands and additional features implemented by the tag.
- **Bank 11: User memory**. Provides space for data of custom features. This bank is optional and during recommissioning a reader can instruct a tag to disable this bank if existent.

The logical address for all banks starts at zero. Memory-access commands have a memory-bank parameter to select one of the 4 blocks and an address parameter. Addresses are formatted as an extensible bit vector (EBV). An address field consists of one or more 8-bit blocks where the first bit of each block determines if another block follows. The value of the EBV is represented like a usual binary number with all blocks combined, ignoring the first bit of each block.

2.7 Tag Commands and States

Like all passive RFID standards the communication works as a reader-talks-first master-slave protocol. The EPC C1G2 standard defines 11 mandatory and some optional reader commands and 7 tag states. The reader sends commands to potentially present tags in the field and depending on their current state, matching or non-matching flags, and matching or non-matching command-selection bits the tag responds with a specified reply, and/or changes its state, or ignores the command. Table 2.1 lists all mandatory reader commands, their binary code at the beginning of the frame, bit length, and how they are protected against transmission errors.

A reader can use the **Select** command at the beginning of each inventory round in order to select a subset of tags. A target parameter modifies the five tag flags (S0-S3 session flags, SL selection flag) depending on a mask-bit sequence. The command specifies a memory bank, an address pointer, and an up to 256-bit long mask sequence. The tag compares the mask with its memory content in the specified sections and sets its flags according to a 3-bit action field in the Select command. A reader can issue a sequence of Select commands in order to perform Boolean operations of multiple mask sequences.

Command	Code	Length [bits]	Protection
QueryRep	00	4	Unique length
ACK	01	18	Unique length
Query	1000	22	Unique length, CRC-5
QueryAdjust	1001	9	Unique length
Select	1010	>44	CRC-16
NAK	11000000	8	Unique length
ReqRN	11000001	40	CRC-16
Read	11000010	>57	CRC-16
Write	11000011	>58	CRC-16
Kill	11000100	59	CRC-16
Lock	11000101	60	CRC-16

Table 2.1: Mandatory EPC C1G2 reader commands.

Although in practise simple Select commands are usually more efficient because a tag does not acknowledge the command and sequences of Select commands increase the chance that tags in the field do not receive all of them correctly.

The inventory commands **Query**, **QueryRep**, **QueryAdjust**, **ACK**, and **NAK** perform the media access control, which is based on a slotted Aloha anti-collision protocol. Every inventory round starts with a Query command as shown in Table 2.2. After the 4-bit command code the reader sets the BLF multiplier and encoding for the tag-to-reader communication for this inventory round. Sel, Session, and Target value define the current session, select a group of tags for this round, and manipulate the inventory flags to enable inventory from multiple readers. The 4-bit Q sets the number of slots for this round (2^Q) and a CRC-5 checksum is appended in order to enhance integrity. QueryRep is a short 4-bit command that marks the beginning of the next slot and QueryAdjust increases or decreases the number of available slots. If the reader receives a tag answer without a collision it sends an ACK command to acknowledge an inventory round which is closed by the tag backscattering its EPC. The not-acknowledge command (NAK) tells the tag to participate in another inventory round if the EPC value was invalid.

After a successful inventory round the reader can either start a new inventory session to identify other tags in the field or send a **Req_RN** command to request a new handle and put the tag into an access state. In the access state, the reader can send Read, Write, Lock, or Kill commands. The **Read** and **Write** commands have a similar structure. After an 8-bit header the command specifies the memory bank and address encoded as an EBV. The Read command specifies the number of words to read and the Write command one 16-bit word of data that needs to be written. Both commands end with the 16-bit handle and a CRC-16 checksum. A **Lock** command enables the reader to block access to certain memory regions or banks, for example the address space of saved passwords. The **Kill** command permanently disables a tag.

Depending on the reader commands the tag changes between 7 states. Ready, Arbitrate, Reply, and Acknowledged are states from power up to a successful inventory round. The Open and the Secured state are memory-access states either without or with using an access password. Deactivated tags are in

	Cmd	DR	M	TRext	Sel	Ses	Tar	Q	CRC-5
Length [bit]	4	1	2	1	2	2	1	4	5
Description	1000	Uplink parameters			Selection parameters			# bits slot-counter	Checksum

Table 2.2: Structure of the Query command.

a Killed state. Below is a short description of the states:

- **Ready:** After the tag enters an RF field and has sufficient power supply it goes to the Ready state. In this initial holding state the tag waits for a Query command in order to start an inventory round.
- **Arbitrate:** If a Query command matches session bits and flags and the tag slot counter is > 0 the tag waits in the Arbitrate state for slot decreasing or changing commands until the slot counter equals zero. This is a holding state for tags taking part in an inventory round.
- **Reply:** The tag sends an answer to the reader's inventory commands and waits in the Reply state for an ACK command. With a successful ACK command the tag backscatters its EPC and changes to the Acknowledged state.
- **Acknowledged:** The tag has now completed a successful inventory round. Depending on the reader command it can change its state to memory-access states, repeat the backscattering of its EPC, or go back to Arbitrate or Ready state. If the tag does not receive a valid command within a specified time it goes back to Arbitrate state.
- **Open:** After a Req_RN command with a matching random number is received, the tag sends a new random-number handle and enters the Open state. In this state it can receive and perform memory access operations. A valid Kill command permanently sets the tag state to Killed.
- **Secured:** This state is similar to the Open state, but the reader must transmit a valid access password. A tag in this state can perform all access commands including Lock.
- **Killed:** From Open or Secured state the tag permanently goes to this state if the reader sends a Kill command with a valid kill password. After an acknowledge response a tag in the Killed state does not respond to any reader command afterwards.

2.8 Tag Selection, Inventory, and Access

This section gives examples how tag selection, the inventory process, and memory-access operations look like in practise. For successful selection of one or a subset of tags in the field the reader can modify five different flags. In order to deal with possible power losses in the UHF field, four of the flags are persistent for $500\,\mathrm{ms}$ up to a few seconds without active power supply of the tag. This allows for example multiple readers to work in the same area and to alternately access tags, without losing the information, which tags have already been read. These persistent flags allow smooth inventory even if the power supply of some tags in the field is lost for a short period of time. The Select command can manipulate the flags and set the conditions for inventory. The Query command addresses the session flags when starting a new inventory round. With the inventory commands the reader performs the anti-collision procedure. Figure 2.7 shows an example how such an inventory round can look like. After an optional

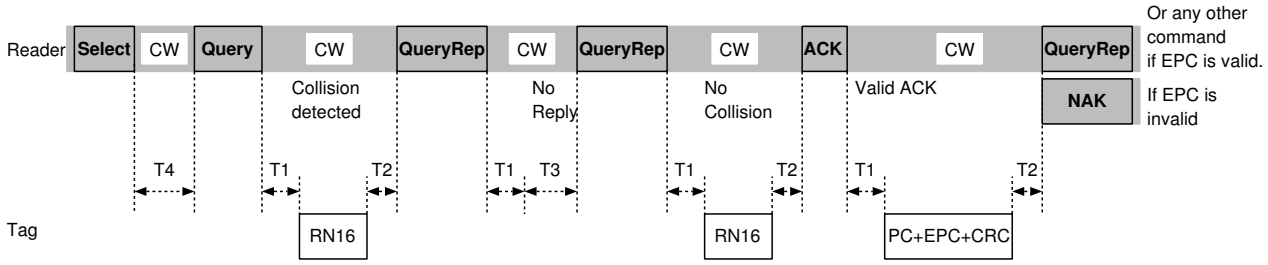


Figure 2.7: Example of an inventory sequence.

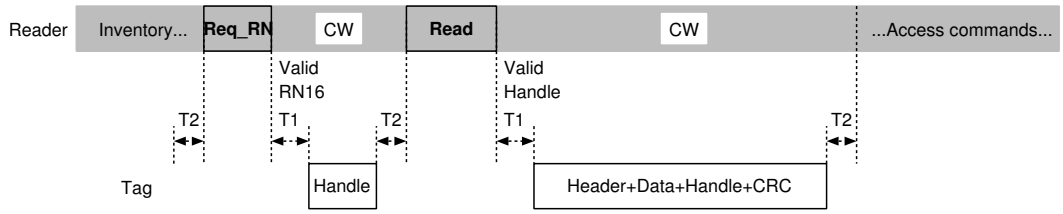


Figure 2.8: Example of a read procedure after successful inventory.

Select command the reader starts the inventory with a Query command. All tags matching the selection parameters randomly initialise their slot counter and reply with a 16-bit random number. It is used in the following inventory commands to address a specific tag in the field. In this example the reader detects a collision in the first slot and therefore it continuous the sequence with QueryRep commands and all tags decrement their slot counter. In the third slot only one tag has a slot-counter value equal to zero and sends a RN16 reply. The reader acknowledges a successful reply. If the handle value of the ACK command is matching the tag responds with the EPC memory content. A NAK response from the reader indicates an invalid EPC, a QueryRep or QueryAdjust continues the inventory round for the other tags and means that the reader received a valid EPC. A Req_RN command tells the inventoried tag to wait for memory-access commands. The four response-time parameters T_1, T_2, T_3, and T_4 in Figure 2.7 and 2.8 are defined as follows:

$$\begin{aligned} MAX(RTcal, 10\,T_{pri}{}^{1}) * (1 - |^{2}FT|) - 2\,\mu s) \;&\leq T_1 \leq\; MAX(RTcal, 10\,Tpri) * (1 + |FT|) + 2\,\mu s) \\ 3.0\,T_{pri} \;&\leq T_2 \leq\; 20.0\,T_{pri} \\ 0.0 \;&\leq T_3 \\ 2.0\,RTcal \;&\leq T_4 \end{aligned}$$

After a successful inventory of one or more tags the reader can send a Req_RN command if it intends to perform further access procedures. A tag responds with a new RN16 number appended with a CRC-16. This handle value is used by the reader in order to access the tag in future commands. Figure 2.8 shows an example read command starting after a successful inventory round.

In order to access protected memory space or to perform a Kill command the reader has to send the correct 32-bit kill or access password if they are set in the tag memory.

[1] Period of a tag-to-reader sub-carrier cycle.
[2] Frequency tolerance, $4\% - 22\%$ depending on BLF and DR.

Chapter 3

Security Enhancement of the EPC C1G2 Standard

With RFID applications becoming omnipresent in everyday life more and more questions regarding security and privacy arise. Since chip area and power-consumption constraints of passive RFID tags are fierce, current standards and implementations often provide only limited security features or base on proprietary developments. A well known example how a proprietary RFID system was compromised after a short period of time is the digital signature transponder (DST) manufactured by Texas Instruments. Used in millions of car immobiliser keys and the Exxon SpeedPass™ electronic payment system, it features tag authentication based on a proprietary 40-bit symmetric cipher. Bono et al. [2005] were able to reverse engineer the cipher and to recover the key for a given challenge-response pair within hours using 16 FPGA boards. In that way it is possible to make a clone containing the same key and to start the car or go on a shopping tour.

Besides security issues, privacy concerns get even more attention in mainstream media. When Metro introduced its future store using RFID labels in 2003, there were discussions about privacy invasion through tracking and monitoring of customers. Initiatives like StopRFID raised attention to privacy threats of widespread usage of RFID in logistics and product labelling [StopRFID, 2005]. End-customer acceptance of these systems is unlikely if these concerns cannot be rebutted. In low-level RFID applications security was not a big consideration in the beginning but research concerning this topic has increased substantially in the past few years.

This chapter explores security and privacy issues of RFID applications in general. First, it examines the security measures in the current version of the EPC C1G2 standard and their shortcomings. After an outline of related work on security improvements, the challenge-response authentication scheme based on a strong symmetric block cipher that is implemented in this work is presented. Finally, we provide a short overview of the used cipher, namely the Advanced Encryption Standard (AES) and discuss the importance of good (pseudo-) random number generation.

3.1 RFID Security and Privacy

Since RFID is a very broad term for various systems and applications security issues have to be analysed depending on the different use cases [Garfinkel et al., 2005]. Factors like data storage and calculation capabilities of the tag, operating ranges, system distribution, or number of tags have to be considered. Security threats usually arise from misbehaving or manipulated tags in a system. Unauthorised readers pose a thread for the privacy of people carrying objects with tags attached [Juels, 2006].

RFID systems rely on correct and authentic information that the readers in the system collect. Wrong or manipulated data very quickly eliminates the advantages of RFID technology. Some examples of

security problems typically found in practise are:

- **Cloning:** If any reader has access to the whole tag memory it is very easy to duplicate tags. Since the EPC value is only a bit string stored in one of the tag-memory blocks, an attacker only needs to read the EPC value of the tag to be cloned and write it into a programmable tag. In theory an RFID label should provide unique identity of an object and is also intended as anti-counterfeit measurement. But without authentication mechanisms tag cloning is simple and it is possible to attach a fake RFID label to a counterfeit good [Juels, 2006].
- **Data manipulation and sabotage:** An attacker can manipulate the tag data within the supply chain of a company. If the collected RFID data is inconsistent with the real world, the company would have to correct the information manually or perform an expensive physical inventory without an auto-ID system. Data manipulation can also disable anti-theft systems or fool automated cashier systems based on RFID.
- **Denial-of-service (DOS):** This threat usually cannot be completely dissolved. Like every RF communication a jamming transmitter can disable communication between reader and tags within reading range. Also malicious blocker tags can prevent successful inventory of tags. They impersonate multiple fake tags in the reading fields and disable the anti-collision algorithm by spamming every available slot [Garfinkel et al., 2005].

Privacy issues arise from unauthorised readers that collect data from tags and try to combine the information with data collected from other places or database information connected to the unique identifier of the tag. A first consideration when discussing privacy issues is the reading range of an RFID system. A contactless smartcard with a reading range of several centimetres is much more difficult to read without notice of the user than a UHF EPC tag with an operating range of up to $10\,\mathrm{m}$. It is important to note that there are several "reading" ranges. The **nominal read range** is the shortest and denotes the maximum operating range specified by the standard or product specification under normal conditions. With improved readers and more sensitive and powerful antennas, the operating range can be significantly increased (**rough scanning range**). Eavesdropping a communication can be done in greater distances than the nominal read range since the tag already receives enough power from the first reader and a second reader can listen to the communication. The **tag-to-reader eavesdropping range** can be larger than the rough scanning range, but it is much smaller than the **reader-to-tag eavesdropping range** because of much higher power transmission from the reader. In UHF systems reader signals can be read several hundred meters away [Juels, 2006].

Media coverage concentrates on customer privacy threats like hidden **tracking** or **inventoring**. Tracking is possible because a tag responds to any reader request with its unique EPC value and therefore a person can be traced with multiple readers when for example wearing cloths with RFID labels. The privacy threat increases when the ID can be combined with additional personal information like identity information, shopping preferences or credit worthiness. Hidden inventoring exploits the fact that an EPC value contains free accessible information about the product attached to. Therefore, it is possible with a single inventory of tags in reading range to gain useful knowledge about persons without their knowledge. It could be useful for an adversary to know what kind of medications are in a person's pocket or what literature is in the backpack [Juels, 2006].

Information leakage of tags is not only a problem on the customer site of the supply chain. RFID-enabled supply-chain management can pose additional business espionage threats. Reading tag information within a company's production and distribution chain can reveal important confidential information [Garfinkel et al., 2005].

3.2 Security Aspects of the EPC C1G2 Standard

The second generation of the EPC standard provides high performance with about $200 - 500$ tagreads per second inventory speed under practical conditions. It allows high flexibility for one or more readers to adjust for different environments and use cases. The costs for this high performance and flexibility are five times more gate equivalents (GE) required on the tag IC compared to the first-generation standard. The main weakness of the standard is its missing or weak security and privacy protection [Dobkin, 2008].

The security protection in the EPC standard is built on optional 32-bit access and kill passwords in combination with the Access, Lock and Kill commands. If the access password in the reserved memory bank is set to zero any reader can access the memory and change the lock status. Once a tag is programmed with a password, the reserved memory bank is locked against read or write access. Changing access rights for the other memory banks requires the reader to provide the correct access password. With the Lock command it is possible to restrict the write access of the other three memory banks. It is also possible to permanently lock a bank and disable write access in general [EPCglobal, 2008].

After receiving a Kill command with the correct 32-bit password, the tag goes into the Killed state and does not respond to any reader commands in the future. The idea behind this concept is to protect consumer privacy. Once the item leaves the supply chain at the point-of-sale device, the tag is permanently disabled and poses no longer a privacy threat to the customer. A disadvantage of this procedure is that the RFID tag can no longer provide benefits for the customer. Also small shops selling RFID-enabled products might not have the infrastructure to kill all tags leaving the store [Juels, 2006].

The EPC C1G2 standard does not encrypt sensitive data but considers the fact that the reader-to-tag signal is much stronger than the tag-to-reader signal. Before the reader sends sensitive data, like passwords or data in memory write commands, it requests a random number from the tag. The reader then blinds the data before transmitting it by xor-ing it with the random number [EPCglobal, 2008].

3.2.1 Possible Attacks on the EPC C1G2 Standard

Considering the weak security features, there are several possible attack scenarios and information-leakage problems in the current version of the standard. First, the EPC value is transmitted to every reader during an inventory round without any authentication of the reading device. This enables an attacker to simply read the EPC value and copy it to an unprogrammed tag. Tag cloning is easy and therefore provides no security against counterfeit of products. The unique EPC value that is freely accessible to all readers also poses privacy threats as explained in Section 3.1.

Sensitive data transmitted by the reader is masked with random numbers from the tag but these values are transmitted by the tag in plain text. If an attacker can eavesdrop the tag-to-reader communication, it is possible to recover the access or kill passwords sent by the reader. In many short-range RFID systems it is very difficult for an attacker to listen to the communication without being noticed. But in UHF systems, even though the backscatter signal strength is very low, the communication can be eavesdropped up to $10\,\mathrm{m}$ using a standard antenna and an equaliser. Dobkin [2008] estimates that with more sophisticated and sensitive equipment the reading range can be extended significantly. This also allows eavesdropping through obstacles like walls. Therefore, blinding of access passwords in the strong reader-to-tag link is not enough to prevent hidden eavesdroping in practise.

3.2.2 Related Work on Security Enhancements

Many different approaches for security enhancements of the EPC C1G2 standard have been proposed during the last years. This section lists the most relevant approaches and developments in this research area. The suggestions can be loosely categorised by the additional hardware needed to be implemented on the tag in order to perform the improved protocol. Many researchers believe that in an EPC C1G2

environment the implementation of cryptographic algorithms on the tag is too costly and only use the already present 16-bit PRNG and CRC-16. Chien [2007] calls these approaches "**lightweight**" protocols. Even though these proposals are compatible with the standard's recommended hardware functionality, most proposals are not compatible with current systems because of different communication flows. The second category uses strong conventional cryptographic functions like symmetric ciphers or hash functions ("**full-fledged**"). These protocols usually have a simpler structure and higher security since they are based on standardised and well known algorithms. The main disadvantage of this approach is the need for additional circuitry and higher power consumption of the tag IC.

Lightweight Proposals without Additional Tag Circuitry

There are a variety of suggestions to introduce security and/or privacy-preserving measures to the current standard without major changes of the tag hardware. Some aim at providing practical security measures and argue that there is no need for bullet-prove authentication schemes in a lot of low-cost EPC applications. Others propose sophisticated authentication protocols with key-update mechanisms and multiple secrets and messages during authentication using the EPC standard's PRNG and CRC.

Juels [2005] suggests to use the kill password also for a simple tag authentication. It prevents simple cloning of tags but is vulnerable to tag-to-reader eavesdropping similar to the reader authentication used for memory access and Kill/Lock commands already defined in the EPC standard.

Nguyen Duc et al. [2006] propose an authentication protocol for also addressing privacy issues. It uses the already integrated 16-bit PRNG and the 16-bit CRC. Tag and back-end system share an access password as defined in the EPC standard and a session key K_i. The back-end system also stores all EPC values present in the system. After every inventory round the session key K_i is updated using the PRNG function. During the inventory round reader and tag exchange random nonces and the tag replies with[1] $\mathrm{M}_1 = \mathrm{CRC}(1 \,||\, \mathrm{EPC} \oplus r \oplus r') \oplus K_i$ instead of the plain EPC value. The back-end system searches through all entries in order to verify if a corresponding pair exists. For reader authentication the tag verifies if a reader can provide $\mathrm{M}_2 = \mathrm{CRC}(1 \,||\, \mathrm{EPC} \,||\, \mathrm{PIN} \,||\, r) \oplus K_i$ with the correct PIN (access password) and session key. Possible attacks on this approach arise from the weak cryptographic properties of the CRC as a hash function [Yeh et al., 2010] and unknown specification or implementations of the currently used PRNG functions conforming the standard specification [Melia-Segui et al., 2011]. The need for a synchronous update of the key K_i on the tag as well as in the back-end system leads to possible DOS attacks. If an end-session command is intercepted and only one part updates the session key K_i there is no further communication possible [Chien and Chen, 2007].

Several papers published weaknesses in its predecessors and proposed improvements. Chien and Chen [2007] intended to prevent desynchronisation attacks by storing a tuple of session keys in the back-end system. Still Yeh et al. [2010] demonstrate a DOS attack and propose an updated protocol. The publication from Habibi and Gardeshi [2011] presents a practical attack on the Yeh et al. proposal and suggests a revised protocol.

Burmester and de Medeiros [2008] present TRAP-3 (trivial RFID authentication protocol) which uses a similar lightweight approach but uses a 32-bit PRF and 48-bit passwords to improve security. Again a desynchronisation attack was found by Yeh and Lo [2009] who propose an improved TRAP-3 version.

Full-fledged Proposals Using Strong Cryptography

When considering implementing strong cryptographic primitives on the tag, the power and area constraints limit the choices of available standardised algorithms. Public-key cryptography, often used for authentication procedures, is hardly feasible on low-cost passive RFID tags. Therefore, only symmetric

[1] r = Tag nonce, r' = Reader nonce, K_i = Session key

cryptography like hash functions or block ciphers come into consideration [Feldhofer and Rechberger, 2006]. Many proposals use one-way hash functions. Weis et al. [2003] suggest a randomised hash-lock scheme where the tag does not respond with its plain text ID (in this case the EPC value) but with[2] R, $h(ID||R)$. The back-end system verifies the tag with exhaustive search through all possible ID values in the system and responds with the corresponding ID. This solves privacy issues but has security weaknesses because the ID is sent in plain text back to the tag. Dimitriou [2005] suggests a hash-chain based authentication protocol where the secret value is updated after every successful authentication round. The work from Cho et al. [2011] presents recent work on hash-based authentication protocols for RFID and tries to eliminate weaknesses from earlier works. None of the listed hash-based proposals suggests a specific hash function or estimates the additional hardware costs.

Feldhofer and Rechberger [2006] compare the hardware costs for possible hash functions to standard block-cipher implementations and come to the conclusion that for the same security level a block cipher is preferable to a hash function.

Actual realisations of security-enhanced EPC C1G2 tag designs are rare. There are four prototyped suggestions which use low-area and low-power optimised block ciphers. A secured tag baseband using the Tiny Encryption Algorithm (TEA) was realised by Zhang et al. [2008]. TEA is a block cipher with 128-bit key and processes 64-bit data blocks and has a lower area and power footprint than AES. Shen et al. [2010] use the block-cipher International Data Encryption Algorithm (IDEA) which also uses 128-bit keys and 64-bit data blocks and has low power consumption and short latency time.

Man et al. [2007b] implemented an AES-enhanced baseband system meeting area and power constraints of RFID systems. It encrypts the communication between the reader and the tag, but the paper misses detailed information about the authentication procedure. Also, no information is provided if the tag meets the response time requirements of the standard since it encrypts the messages on the fly during communication. A very power efficient tag IC implementation including an AES core is shown by Ricci et al. [2008]. The way how the authentication mechanism or encryption is integrated into the standard's communication flow is not shown in the paper.

Suna and Lee [2011] elaborate the implementation of an AES enhanced EPC tag with focus on the currently ongoing standardisation of ISO/IEC 29167, a standard for future security services for ISO/IEC 18000 RFID devices. It discusses possible tag memory structures and protocol details, but does not show an actual implementation of the design.

3.3 Authentication Using Standardised Symmetric Cryptography

The last section showed that lightweight approaches are currently more popular for enhancing the current EPC C1G2 standard. Avoiding extra hardware costs on the tag by using the CRC-16 as a hash function and the already integrated 16-bit PRNG leads to complicated protocols with incompatible communication flow to the existing standard, possibly unreliable key-update schemes, and difficult analysis of the security strength.

This work implements a challenge-response protocol for mutual authentication whose security relies on standardised symmetric cryptographic primitives. Authentication is used to verify the claimed identity of one communication partner to the other partner. Symmetric means that both partners use the same key in contrast to asymmetric or public-key cryptography with a different public and private key. Symmetric systems have the advantage of much lower computational costs but have to solve the problem of key management and distribution. Feldhofer et al. [2004] present a way to use a standardised symmetric algorithm like AES in an RFID environment. In the following we give an overview of a challenge-response authentication protocol as standardised in ISO/IEC 9798-2. Then we explain how this idea can be integrated in the EPC C1G2 standard using an interleaved protocol approach as proposed in Feldhofer

[2] R = Random number generated by the tag, h = One-way hash function.

et al. [2004] and present the custom-command structure in order to guarantee compatibility with the current standard communication procedure. After an analysis of the security enhancement we provide a short explanation of the AES standard and discuss the importance of the challenge generation.

3.3.1 Challenge-Response Authentication Protocol

ISO/IEC 9798-2 specifies entity authentication based on symmetric-cryptography algorithms [ISO/IEC, 2008]. Two parties share a secret key, implement a symmetric cryptographic primitive, and are able to generate (pseudo-) random numbers. Figure 3.1 shows an authentication process of entity R (reader) against T (tag). T sends a random number C_T to R, which encrypts the number using the shared secret K and sends it back to T. T can now verify if R owns the secret key. An authentication of T against

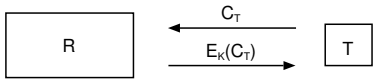


Figure 3.1: Basic authentication process of R against T.

R is the same in reverse order. For mutual authentication both entities exchange a random number and send the encrypted numbers as shown in Figure 3.2. The order of the numbers in the encrypted

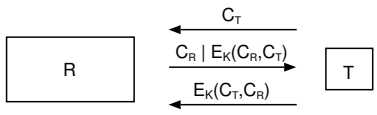


Figure 3.2: Basic mutual authentication process of R and T.

messages is reversed to prevent two identical encrypted messages which would allow a reply attack. Besides implementation details to prevent replay or reflection attacks every system using symmetric authentication mechanisms has to solve the problem of secure key management and distribution.

3.3.2 Integration into the EPC C1G2 Standard

The EPC C1G2 standard is a high-performance protocol. Therefore, maximum response times of the tag are short. Using the highest BLF the tag has only about $20\,\mu$s to prepare the response. Even if the AES engine is clocked with the highest frequency available on a passive UHF tag (about $3.5\,$MHz, see Section 4) it has only 70 clock cycles to complete the encryption. This is unfeasible for a low-power and low-area AES implementation.

Feldhofer et al. [2004] suggested an interleaved protocol as a solution for using AES in an RFID environment. The reader does not wait for the tag to complete the encryption and sending the answer

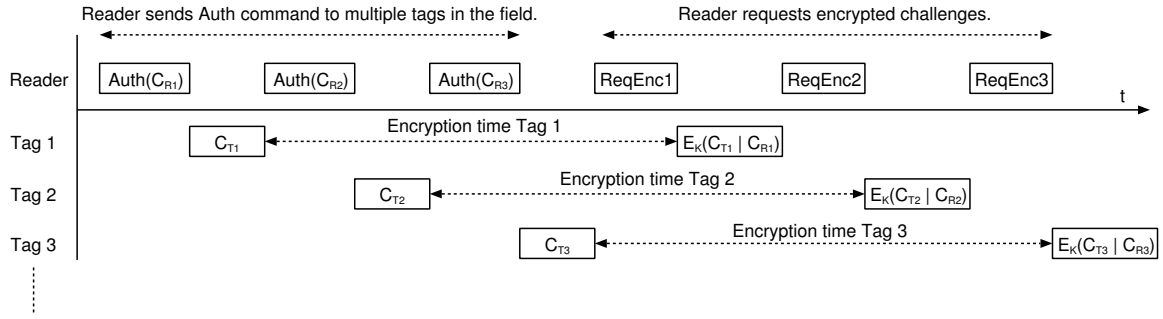


Figure 3.3: Interleaved authentication protocol with three tags involved.

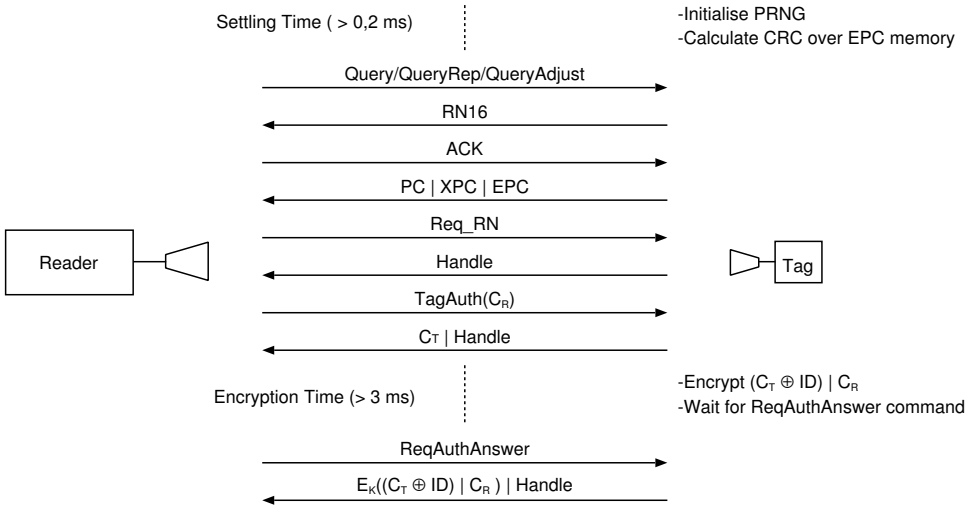


Figure 3.4: A full tag-authentication round after tag startup.

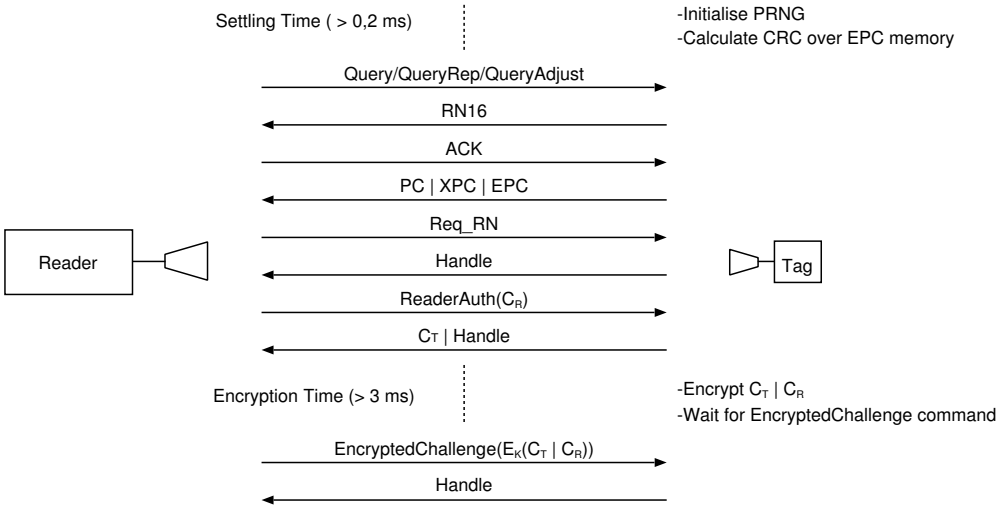


Figure 3.5: A full reader-authentication round after tag startup.

but sends a separate command to request the response. This leads to communication overhead and performance loss if the reader communicates only with one tag, but has the big advantage that the reader can address other tags present in the field meanwhile. Figure 3.3 shows how the reader can efficiently authenticate multiple tags in the field. The reader sends an Auth command to the first tag and during the waiting time the reader can send an authentication command to other tags. After the first tag has finished the encryption process the reader sends ReqEnc commands to collect the response from the tags.

Plos [2007] already showed a successful integration of such an interleaved authentication protocol into the EPC Gen2 standard using a semi-passive microcontroller-based tag prototype.

Figure 3.4 shows the communication flow for a full tag-authentication procedure. The inventory sequence is identical to the standard specification. After a successful anti-collision procedure the tag responds with its PC, XPC, and EPC. The request for a Handle brings the tag into the Open state where it waits for authentication commands. For the structure of the custom authentication commands see Section 3.3.3. The TagAuth reader command is answered with a short acknowledge command containing the tag challenge $\mathrm{C_T}$ and the tag loads the 64-bit reader challenge xor the last 64 bits of the EPC value ($\mathrm{C_R} \oplus \mathrm{ID}$) and the 64-bit tag challenge ($\mathrm{C_T}$) into the AES engine and encrypts it using the secret shared key (K). After the encryption is done (minimum $\sim 3\,\mathrm{ms}$) the reader requests the encrypted answer with

the RequAuthAnswer command.

The communication flow for reader authentication in Figure 3.5 has a similar structure. The authentication round starts in the same way, only using a different header for the ReaderAuth command. After the encryption time the reader sends the encrypted challenge to the tag. The tag compares the encrypted reader challenge with its own calculation and if they match it responds with the Handle value. The reader is now authenticated and can send memory access commands.

3.3.3 Custom Commands

For integration into the current standard the authentication process uses custom reader commands and tag answers according to the EPC-standard definitions. Every custom reader command starts with a 2-byte header with a fixed first byte value of 0xE0 and ends with the Handle value of the addressed tag and the CRC-16 checksum. The tag answer to custom commands starts with a 1-bit header set to 0 indicating a successful processing of the reader command. The reply frame ends with the current tag handle and a CRC-16 checksum. Table 3.1 shows the structure of the initial reader command to start a tag authentication and the corresponding reply including the tag challenge. The initial reader command for

TagAuth	Command code	Length	C_R	Handle	CRC-16
Length [bit]	16	16	Variable (64)	16	16
Description	11100000 10000001	Length of C_R in bit	Reader challenge	Handle	

Tag reply to TagAuth	Header	C_T	RN	CRC-16
Length [bit]	1	Variable (64)	16	16
Description	0	Tag challenge	Handle	

Table 3.1: Initial TagAuth reader command and corresponding tag reply.

the reader authentication (ReaderAuth) has the same structure except a different header value (0xE082). The tag replies the same way to both initial authentication reader commands.

Since it is a reader-talks-first protocol the tag has to wait for a request from the reader to transmit the encrypted challenge after it has performed the AES encryption. Table 3.2 shows the structure of the reader command to request the tag response.

ReqAuthAnswer	Command code	Handle	CRC-16
Length [bit]	16	16	16
Description	1110000 10000100		

Tag reply to ReqAuthAnswer	Header	$E_K((C_T \oplus ID)\|C_R)$	Handle	CRC-16
Length [bit]	1	Variable (128)	16	16
Description	0	Encrypted challenge		

Table 3.2: ReqAuthAnswer command and corresponding tag reply.

In order to perform reader authentication the reader again waits until the tag has finished the encryption process and sends the encrypted challenge to the tag. During encryption the tag does not respond to reader commands at all. If the tag is ready to compare its own calculation with the information in the EncryptedChallenge command and the encrypted challenge is correct it replies with a standard acknowledge frame (Table 3.3). In case of a wrong EncryptedChallenge message the tag responds with an error message and waits for a new authentication round.

EncryptedChallenge	Command Code	Length	$E_K(C_R\|C_T)$	Handle	CRC-16
Length [bit]	16	16	Variable (128)	16	16
Description	11100000 10000100		Encrypted challenge	Handle	

Tag Answer to EncryptedChallenge	Header	Handle	CRC-16
Length [bit]	1	16	16
Description	0	Handle	

Table 3.3: Encrypted challenge command and tag reply if authentication was successful.

3.3.4 Analysis of the Suggested Security Enhancements

One might now ask which issues regarding security and privacy are solved with integrating strong symmetric cryptography into the tag. The tag authentication provides cloning protection and therefore can be used as anti-forgery countermeasure. It is possible to create an EPC label with the same ID but every legitimate reader can request a tag authentication and verify if the tag is authentic. Reader authentication prevents security issues related to unauthorised memory access commands and kill or recommissioning procedures. When only authorised readers can manipulate tag data, sabotage and manipulation of logistic information is not possible.

Authentication does not solve data leakage of sensitive information exchanged. Direct encryption of data is not practicable because of the short response time available for the tag during communication. Suna and Lee [2011] suggest a way to secure sensitive data using the additional hardware. Instead of blinding (xor) the information with a previously transmitted random number as specified in the standard the encryption engine calculates session keys derived from the challenges (C_R, C_T) and the shared key K which can be used for blinding during memory-access commands.

The two communication flows in Figure 3.4 and 3.5 show that also in the proposed enhanced protocol the EPC value is transmitted in plain and before any authentication. This raises the same privacy concerns as in the standard protocol. It is possible to prevent exposure of the EPC value as shown in Figure 3.6 by using tag authentication, but it would come with high costs. When using a standard 96-bit EPC value the tag only sends the first 32 bit during the inventory round. The last 64 bits containing the privacy relevant object class and serial number are only transmitted during a tag authentication procedure. Since the reader does not know the EPC value after the inventory round, it has to request the back-end system to search through all possible keys in order to verify the encrypted challenge. Besides this computational overhead in the back-end system, every reader has to perform a tag authentication round even when it only wants to perform an inventory of all tags to get their EPC values. This would decrease performance dramatically and probably is not applicable in a large logistic application. Still the idea is also part of the ISO/IEC 29167-10 draft and it is possible to make trade-offs between privacy protection and performance loss depending on the application.

Besides improved security, compatibility to the existing standard and systems is an important aspect. The proposed enhancements use the same inventory procedure and are fully compliant with the existing EPC standard. The protocol control (PC) and extended protocol control (XPC) words can be used to inform the reader about the presence of the security features. The ISO/IEC 29167 is an official standard in progress, concerning security services for the EPC C1G2 standard using AES which defines the structure of the custom commands. Memory access and kill commands are the same as defined in the EPC C1G2 standard but the reader has to perform an authentication in order to bring the tag into the secured state. Even though the proposed authentication mechanism in this work uses the same principles and basic structure, there are differences in the exact command definition because the prototyping of this work's secure-tag baseband was finished before the first publication of the ISO/IEC 29167-10 working draft. However, the main challenge implementing strong cryptography into the tag without exceeding the chip-area and power constraints for low-cost passive UHF tags is the same in both cases.

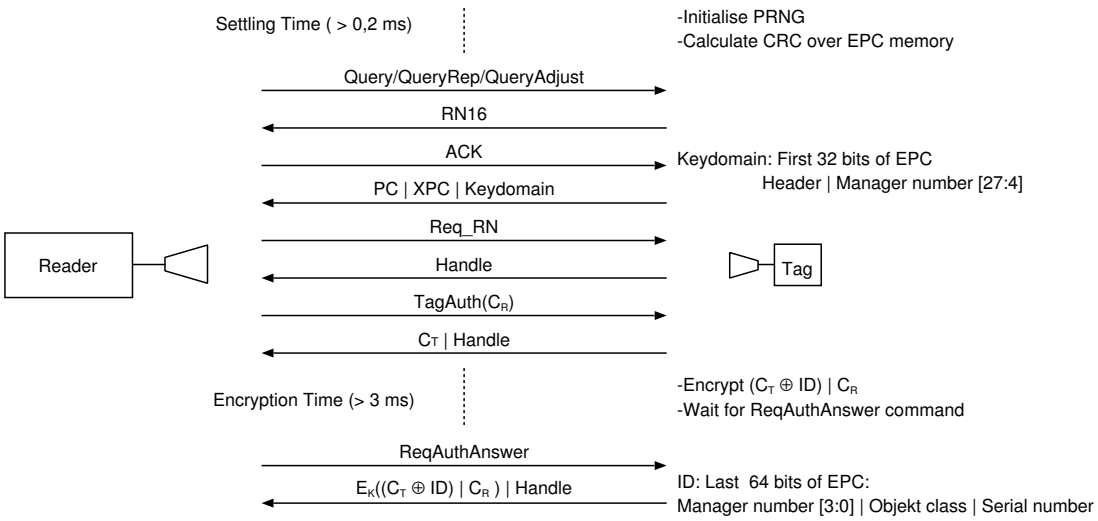


Figure 3.6: Possible communication flow to protect privacy and prevent tracking.

3.4 Advanced Encryption Standard

Several severe weaknesses found in proprietary cryptographic solutions showed the importance of using well documented and extensively analysed standardised algorithms. The Advanced Encryption Standard (AES) describes the currently most used symmetric block cipher. After a five-year standardisation period, the National Institute of Standards and Technology (NIST) presented the AES 2001 as a successor of the Data Encryption Standard (DES). Fifteen design proposals were evaluated and the final choice was Rijndael developed by Joan Daemen and Vincent Rijmen. Besides resistance against modern cryptanalysis and the clear description, it can be efficiently implemented in both software and hardware. It processes 128-bit data blocks and uses 128, 192 or 256-bit keys. The following short summary will only refer to the 128-bit key variant because it is the most used in constrained systems including this work. The difference in the three variants is only in key expansion and number of rounds.

```
1  Cipher(byte input[4*4], byte output[4*4], word round_key[4*11])
2  begin
3    byte state[4,4]
4    state = input
5    AddRoundKey(state, round_key[0, 3])
6
7    for round = 1 step 1 to 9
8      SubBytes(state)
9      ShiftRows(state)
10     MixColumns(state)
11     AddRoundKey(state, round_key[4*round, 4*round+3])
12   end for
13
14   SubBytes(state)
15   ShiftRows(state)
16   AddRoundKey(state, round_key[40, 43])
17   output = state
18 end
```

Listing 3.1: Pseudo-code of AES 128-bit encryption with 16-byte state and expanded 44-byte keyword as arguments [National Institute of Standards and Technology (NIST), 2001]

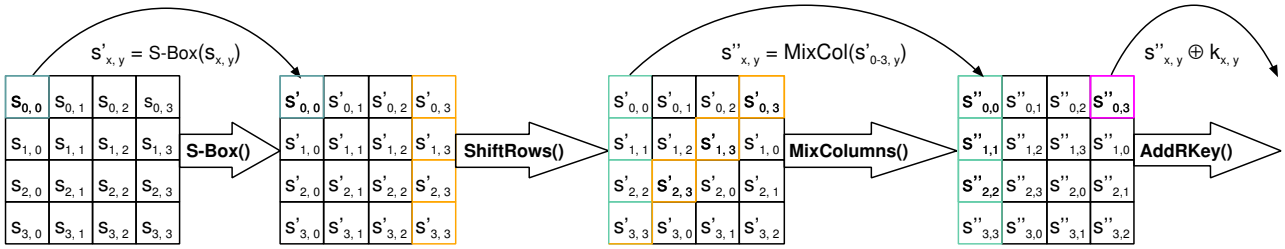


Figure 3.7: The four operations within one AES round.

Listing 3.1 shows pseudo-code of the encryption process. Internally the algorithm performs the operation on a two-dimensional array of 4x4 bytes. The key is expanded to 11 round keys. After an initial AddRoundKey, 10 rounds with the four functions SubBytes, ShiftRows, Mixcolumns and AddRoundKey are performed.

Figure 3.7 shows the four operations and how they are applied to the state. SubByte is an invertible byte-wise substitution. ShiftRows is a shift operation on the second, third, and forth row. MixColumns can be written as a multiplication of a 4x4 matrix with each column. The last round skips this operation. AddRoundKey is an xor of the state at the end of each round with the round key [National Institute of Standards and Technology (NIST), 2001].

3.5 Random-Number Generation using Grain

Every challenge-response protocol needs random numbers generated by both entities. If the challenges do not change or are predictable for an attacker, the protocol becomes susceptible to replay attacks. The attacker can use a previously eavesdropped $\mathrm{E_K(C)}$ message if the challenge C does not change, or request new challenges until one occurs that was already used in a previous authentication session between to legitimate entities.

Generating 'good' random numbers in a digital circuitry is a difficult task. There are suggestions for true random-number generators, using for example inverter loops, but these designs usually have several drawbacks. The quality of the generated numbers is difficult to verify and these circuits are often influenced by changing temperatures and voltages. Especially low-cost passive RFID devices have to be resistant against changing environmental conditions. Therefore, RFID tags in general use pseudo-random number generators (PRNG) to generate random numbers needed for anti-collision procedures or challenges for authentication protocols.

A PRNG produces a fixed sequence of bits depending on the initialisation. The quality of the PRNG is measured by the length of the bit stream until an observer can distinguish the generated sequence from a truly random sequence. The EPC Gen2 standard defines a 16-bit PRNG which is used for generating the tag handles and initialising the slot counter during the inventory round. It also provides the bit masks to blind sensitive information in the reader-to-tag link. The standard leaves the implementation unspecified but defines the following quality criteria:

- Independent on data stored on tag, field strength, and data rates.
- The probability for a generated 16-bit random number to match a specific number j has to conform $0.8/2^{16} < P(RN16 = j) < 1.25/2^{16}$.
- The probability of 2 tags out of 10 000 generating the same sequence has to be below 0.1%.
- The probability to predict a 16-bit random number should be below 0.02%.

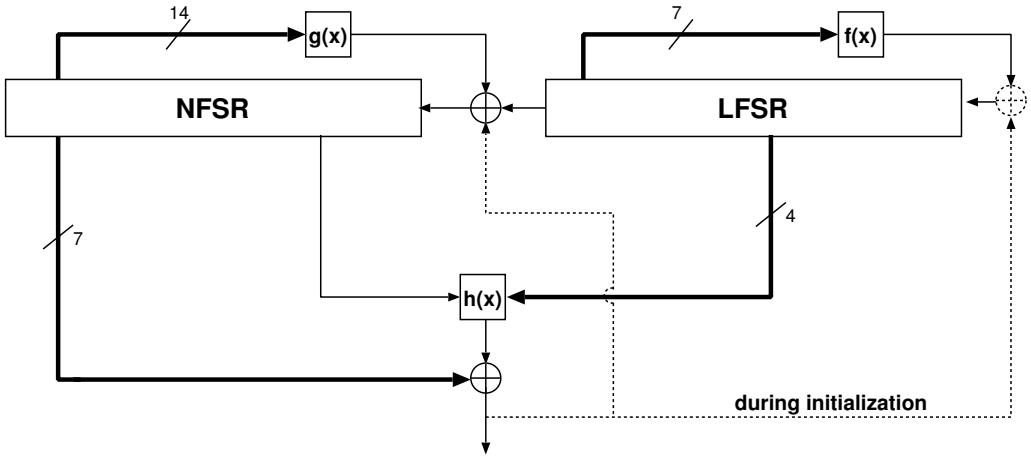


Figure 3.8: Overview of the Grain cipher.

These criteria are sufficient for the anti-collision protocol but are not enough to provide cryptographic security. Information about the implementations in current commercial solutions are rare. Melia-Segui et al. [2011] show a successful attack on PRNG proposals for the EPC standard. The attack on the PRNG of the Mifare system [Garcia et al., 2008], which also provides security functions, shows the possible risks of weak custom PRNG implementations.

The working draft of ISO/IEC 29167-10 also stresses the importance of high-quality random-number sequences and refers to standardised test suits such as NIST SP800-22 but leaves the implementation open to the manufacturer.

This work uses the lightweight stream cipher **Grain** in order to provide secure pseudo-random numbers for the authentication process. Like most custom PRNG implementations it uses feedback shift registers which are easy to implement in hardware. The advantage to use a published stream cipher for this task is that it is constantly analysed using modern cryptanalysis. A weakness during initialisation has been found in the first version and has been eliminated in an updated specification. This constant review process makes published solutions usually more secure than closed proprietary PRNG implementations. The disadvantages of using Grain for pseudo-random number generation is the fairly long initialisation phase of 160 clock cycles and higher area usage than custom PRNG implementations.

Grain

In the years 2004 to 2008 the project eSTREAM tried to find a portfolio of new stream cyphers. Hell et al. [2006] proposed the cipher Grain which was designed for low hardware complexity and power consumption. It is a synchronous cipher that uses an 80-bit key, a 64-bit IV vector, and produces in the default implementation 1 bit per cycle.

Figure 3.8 shows an overview of the Grain cipher. It consists of two feedback shift registers, one with nonlinear feedback (NFSR) and one with linear feedback (LFSR). The polynomials of the feedback registers are defined as follows:

$$f(x) = 1 + x^{18} + x^{29} + x^{42} + x^{57} + x^{67} + x^{80}$$

$$\begin{aligned} g(x) \quad = \quad & 1 + x^{18} + x^{20} + x^{28} + x^{35} + x^{43} + x^{52} + x^{59} + x^{66} + x^{71} + x^{80} + \\ & x^{17}x^{20} + x^{43}x^{47} + x^{65}x^{71} + x^{20}x^{28}x^{35} + x^{47}x^{52}x^{59} + x^{17}x^{35}x^{52}x^{71} + \\ & x^{20}x^{28}x^{43}x^{47} + x^{17}x^{20}x^{59}x^{65} + x^{17}x^{20}x^{28}x^{35}x^{43} + x^{42}x^{52}x^{59}x^{65}x^{71} + \\ & x^{28}x^{35}x^{43}x^{47}x^{52}x^{59} \end{aligned}$$

The output function $h(x)$ uses 4 bits form the LFSR (tag positions: $x_0 = s_{i+3}$, $x_1 = s_{i+25}$, $x_2 = s_{i+46}$, $x_3 = s_{i+64}$) and 1 bit from the NFSR (tag position: $x_4 = b_{i+63}$).

$$h(x) = x_1 + x_4 + x_0x_3 + x_2x_3 + x_3x_4 + x_0x_1x_2 + x_0x_2x_2 + x_0x_2x_4 + x_1x_2x_4 + x_2x_3x_4$$

The output of $h(x)$ xor 7 additional bits from NFSR results in the keystream. At the beginning of a keystream generation the cipher must be initialised with the key and the initial vector (IV). The registers of the NFSR are filled with the 80-bit key and the LFSR is initialised with the 64-bit IV. The remaining 16 bits are set to '1'. During the initialisation period of 160 cycles the output is xor-ed to the inputs of both feedback registers.

3.6 Summary

The EPC C1G2 standard is a high-performance protocol but it only provides weak security features. There are basically two ways to increase the security properties of the standard. Many proposals suggest modified communication protocols without adding additional hardware functionality to the tag. These lightweight approaches come at almost no additional costs but result in complicated protocol designs whose security is often difficult to evaluate. On the other hand, authentication mechanisms based on well-known cryptographic algorithms result in straightforward protocol designs and it is easier to evaluate the security gain of the overall system. The main task when adding strong cryptography to an EPC standard compliant system is to meet the fierce chip area and power-consumption constraints of low-cost passive RFID systems.

This work suggests a mutual challenge-response authentication protocol based on AES. Using a standardised cryptographic algorithm and protocol guarantees well tested and evaluated security properties of the final system. Standardised in 2001, the AES is currently on of the most used block ciphers. The design of the authentication protocol is based on ISO/IEC 9798-2 and cryptographically secure challenges are generated by the stream cipher Grain. In spite of the simple structure of the challenge-response authentication, implementation details and message structures have to consider replay and reflection attacks.

The integration into the current EPC C1G2 standard is done by defining custom user commands. The inventory round remains unchanged which guarantees compatibility of security-enhanced tags with current systems. Using an interleaved protocol design reduces the performance loss of the RFID system due to the long encryption time of the AES module. The next chapters elaborate the practical challenges to bring strong cryptography to passive RFID tags. Besides the general design flow of VLSI circuits, there is a special focus on low-power/area optimisation.

Chapter 4

Tag Architecture

A typical EPC RFID tag consists of an IC attached to a UHF antenna as shown in Figure 4.1. The IC combines an analog front-end, a digital controller, and nonvolatile memory (NVM). The analog front-end performs the demodulation of the received data frames (Demod signal), the backscatter modulation of the data transmitted to the reader (Mod signal), and extracts power from the RF field to supply the IC (Vcc, Gnd). It also generates clock and reset signals for the digital part of the circuit. The digital controller performs the de-/encoding of the data frames and implements the protocol handling. This includes CRC-5 and CRC-16 modules, slot counter, PRNG, and memory-access handling of the NVM. The EPC standard memory banks are stored in the NVM, containing the EPC value, access/kill passwords, and additional user data.

This work implements the digital control part of a security-enhanced EPC tag with an AES and a Grain core. It realises the main commands and tag states of the official standard specification as discussed in Chapter 2 extended with the custom user commands from Chapter 3 used for the authentication protocol. After discussing the constraints regarding area usage and power consumption, this chapter provides an overview of the architecture. Further, it describes the different components and the communication flow between them and external interfaces. Finally, the design of the cryptographic modules is discussed with focus on the specific constraints for passive RFID tags.

4.1 Area and Power Constraints

The area constraint for an EPC tag derives mainly from economic factors. EPC tags are intended for applications with very high label unit numbers and every cent difference in per item costs changes the overall costs of the RFID system significantly. Sarma [2001] sets the price goal for wide-spread usage of passive low-level RFID tags to $5\,\mathrm{cent}$ or lower. Depending on the process technology, this results in about $0.25\,\mathrm{mm}^2$ available die size for the IC. Using a $130\,\mathrm{nm}$ CMOS process technology results in about $50\,000$ gate equivalents (GE) available for the whole IC. When considering that the analog

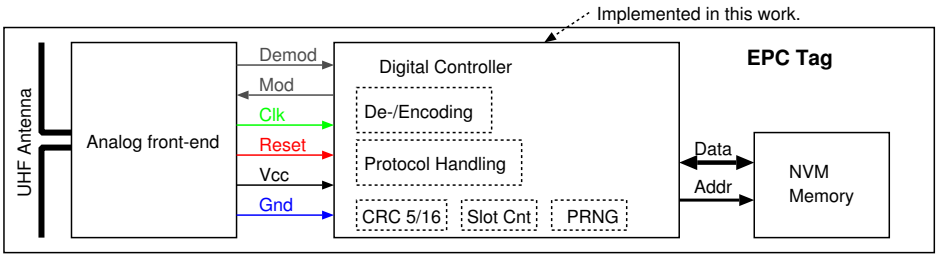


Figure 4.1: Overview of a passive UHF EPC tag.

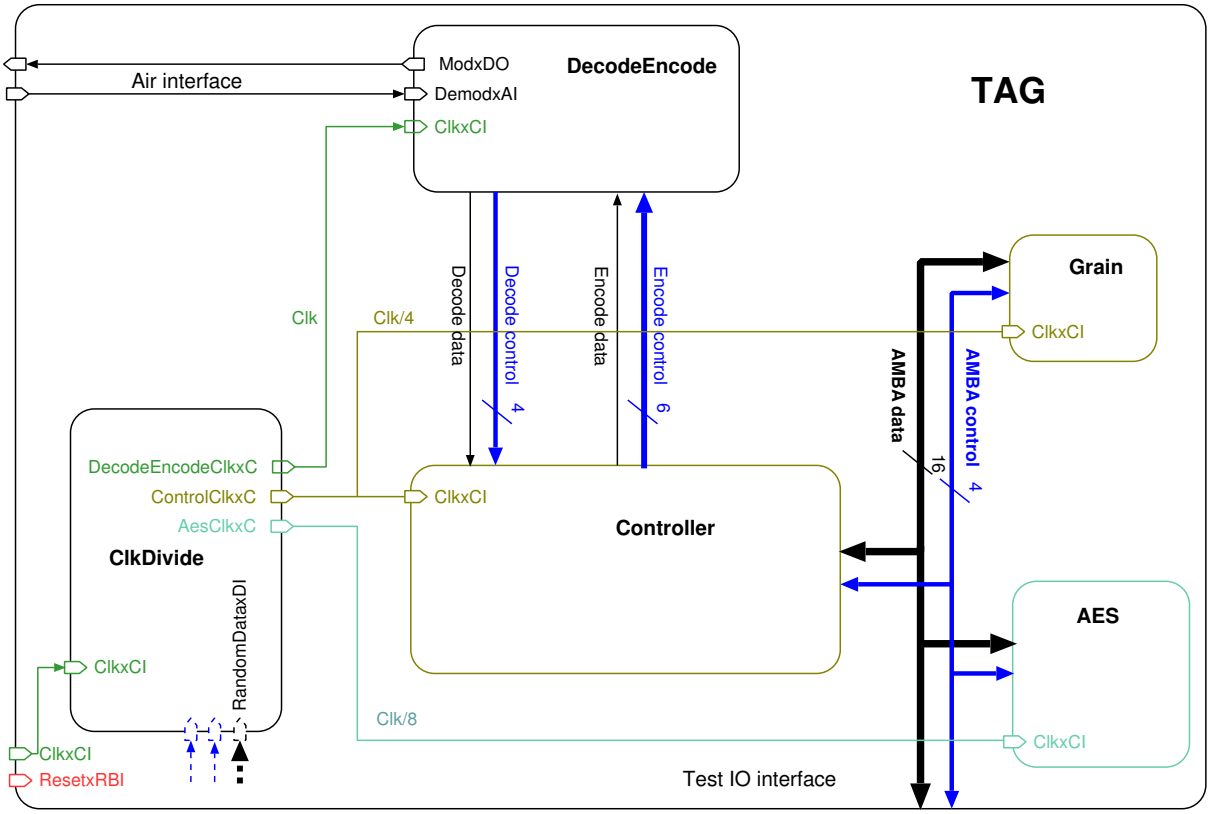


Figure 4.2: Overview of the secure tag digital controller.

circuitry and the NVM need a big part of the chip area, there are only about $10\,000 - 15\,000\,\mathrm{GE}$ left for the digital controller including cryptographic units. These are only rough estimations and depend on many factors like technology, price limits for one tag, and economic benefits from introducing an RFID system. In literature the upper limit used for cryptographic circuits on passive EPC tags is estimated between $2\,000 - 5\,000\,\mathrm{GE}$ [Guajardo et al., 2009].

The passive power supply over the RF field limits the maximum power consumption of the tag. It mainly depends on the distance between tag and reader and is the range of $50\,\mu\mathrm{W}$ at $3\,\mathrm{m}$ to $10\,\mu\mathrm{W}$ at $5\,\mathrm{m}$ [Zhang et al., 2008]. Besides average power consumption, the tag has to avoid large power-consumption spikes which could overstress the capacity of the power-retrieving circuitry. The higher the power consumption of the tag, the lower is the possible performance in terms of operating range and maximum data rates of the RFID system. Therefore, the maximum average power-consumption limit for the design in this work is set to $10\,\mu\mathrm{W}$. From an architectural point of view, the main approach to reduce the power consumption is to use a minimum clock frequency for every task. The lower bound for the system clock is derived from the sampling accuracy needed during de-/encoding. Reader frames start with synchronise symbols that determine the down and uplink data rates. In order to sample input frames correctly and to achieve the demanded tolerance of the BLF, the synchronisation symbols have to be sampled with at least $3.5\,\mathrm{MHz}$ [Man et al., 2007b]. The main controller and the cryptographic units use lower frequencies as explained in the following sections. For low-power methods on implementation level see Section 5.2.

4.2 Overview of the Tag Design

Figure 4.2 shows an overview of the tag design. For the communication from/to the reader, it provides signals for the RFID air interface. The DemodxAI signal is the input signal from the antenna demodu-

lated by the analog circuitry. For the backscatter modulation the controller provides the ModxDO signal. In a completely integrated tag, the clock signal and the power supply would also be provided by the analog front-end. The target ASIC process does not support NVM for the EPC memory and the keys. In order to make the initialisation of the memories and the setup for test runs more convenient, the design implements a test interface which allows direct access to all the memories in the design. It is also possible to control and use the AES and Grain modules independently from the RFID interface. Internally, the design uses an 8-bit variant of the AMAB APB bus [ARM, 1997] for the communication between the RFID protocol controller and the cryptographic units. For a detailed description of the read and write operations over the test interface see the datasheet in Appendix A.

The main parts of the design are DecodeEncode, Controller, AES, Grain, and ClkDivide:

- **DecodeEncode:** This module implements the decoding and encoding of the data transmitted over the air interface.
- **Controller:** Main control unit that is used for the protocol handling. It implements the command handling and the tag states of EPC C1G2 standard and controls the cryptographic units.
- **AES:** An encryption-only 128-bit implementation of the Advanced Encryption Standard. The work uses an adapted intellectual property (IP) model presented by Feldhofer et al. [2005].
- **Grain:** Low-power implementation of the stream cipher Grain used as a PRNG. It generates secure pseudo-random numbers for the anti-collision mechanism and the challenge-response protocol.
- **ClkDivide:** For reducing the power consumption, this module provides different clock frequencies for the components of the design.

4.3 DecodeEncode

The DecodeEncode module performs the detection and decoding of incoming frames, calculates the demanded response time, and encodes the output bitstream. In order to support all data rates with the demanded accuracy, the input signal has to be measured with at least $300\,\mathrm{ns}$. This sets the lower bound of the clock frequency for this module to $3.5\,\mathrm{MHz}$ [Man et al., 2007b]. Figure 4.3 shows an overview of the module. The 10-bit time counter and the 3 registers measure and store the synchronisation symbols at the beginning of reader commands. The control logic for this module uses an FSM with 53 states and an 8-bit counter. Separating it into two state machines for decoding and encoding would not result in a lower complexity because of the overhead for the control signals of components used during both processes. The FSM selects reference values and compares them with the counter value during decoding and encoding. It also handles the communication with the Controller of the tag.

4.3.1 Decoding Reader Frames

When the Controller is ready to process incoming reader commands it enables the DecodeEncode unit to wait for an incoming reader frame. The asynchronous demodulated input signal is synchronised using 2 registers and a signal change triggers the FSM during decoding. The FSM selects the reference values to check the delimiter and following synchronisation symbols Tari, RTcal and TRcal for validity. For decoding the bitstream, the pulse length is compared with $\mathrm{RTcal}/2$ to distinguish between Data-0 and Data-1 symbols. If an EOF symbol is detected, the FSM initialises the response time and waits for the Controller to provide the bitstream of the tag-answer frame.

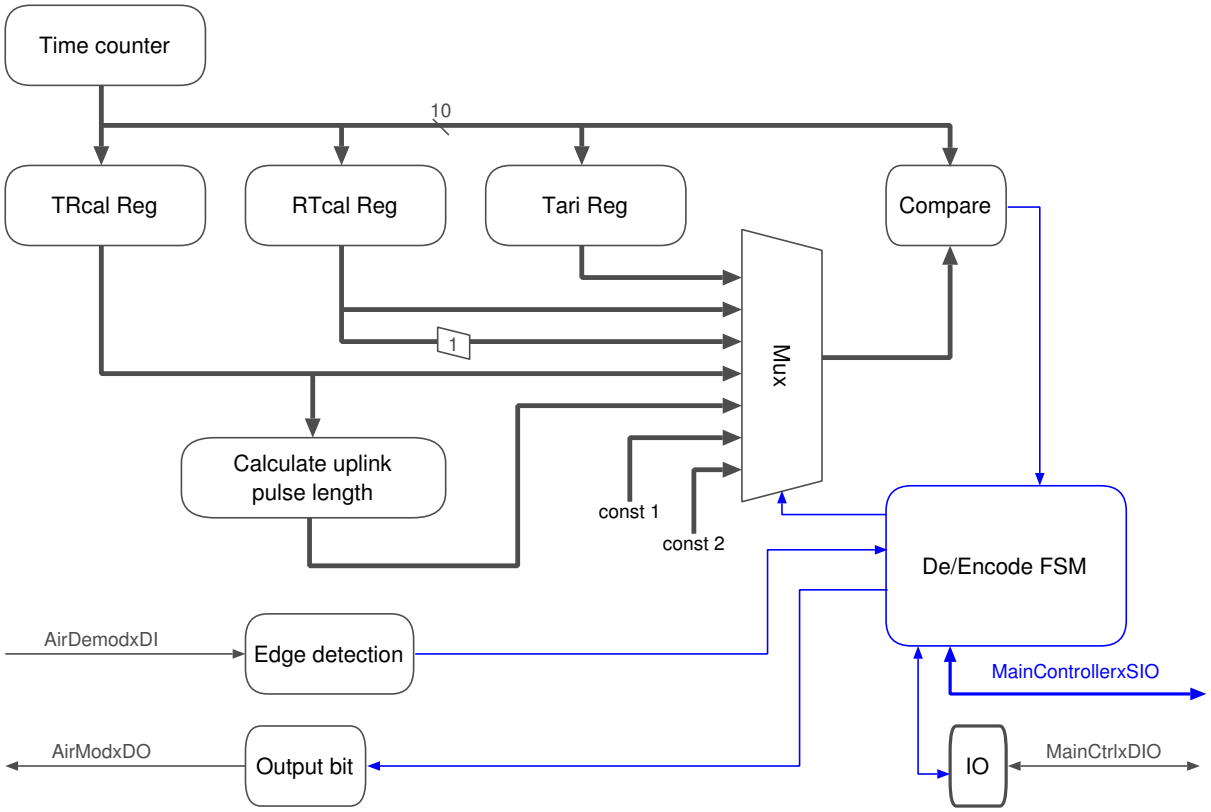


Figure 4.3: Overview of the DecodeEncode unit.

4.3.2 Encoding Tag Answer Frames

The DecodeEncode module contains a so-called calculation unit for performing basic operations to determine the uplink pulse length and the response time. The reference time for the uplink pulse length (FM0 or Miller encoding) is the result of either $\mathrm{TRcal}/8$ or $3 * \mathrm{TRcal}/64$. The maximum of RTcal and 10 pulse lengths of the tag sub carrier defines the response time for a tag reply. Both sub-carrier pulse length and response time are calculated using shift operations and one 12-bit adder.

After the response time, the DecodeEncode unit requests the first bit for transmission from the Controller. The header of the Query command sets the encoding and in case of Miller encoding, the number of sub-carrier cycles per symbol. The time counter compared to the pulse-length reference value determines the changing edges of the AirModxDO signal. The frame ends with an EOF symbol according to the standard definition.

4.3.3 Communication between DecodeEncode and Controller

Since the Controller unit runs at a lower clock frequency than the DecodeEncode unit, these two units use a partial handshake protocol [Kaeslin, 2008]. The synchronisation values sent by the reader determine the data rates and therefore the DecodeEncode unit forces the communication flow. Figure 4.4 shows the communication flow in both directions when the clock frequency of the Controller unit is $1/4$ of the system clock. During decoding the DecodeEncode unit sets a Ready signal when a valid data bit is ready. The ready signal stays high for at least 4 clock cycles and therefore at least one rising edge of the Controller clock is within this time period. The Controller guarantees to read the data bit within the ready-signal high period.

When encoding data for backscatter transmission, the Controller waits for a request signal and applies valid data together with a Ready signal until the request signal changes to low.

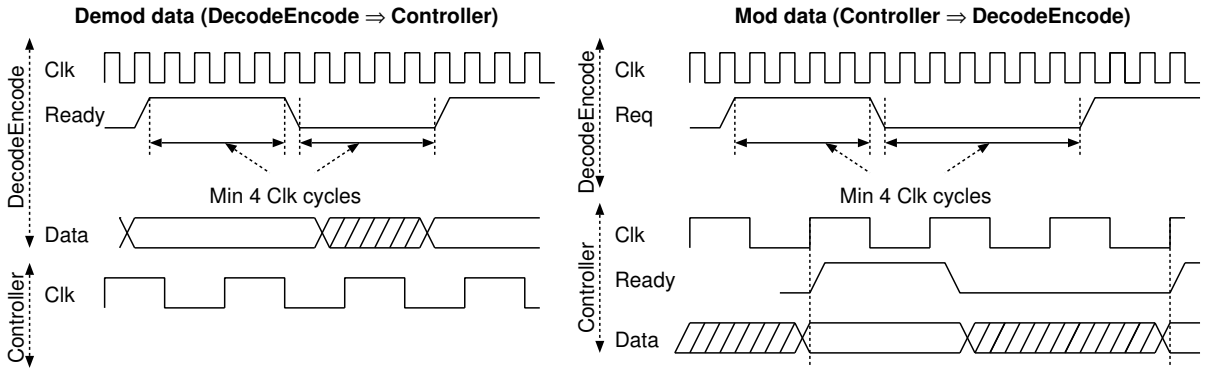


Figure 4.4: Dataflow between the DecodeEncode unit and the Controller.

4.4 Controller

This is the main control unit of the design. When implementing an RFID protocol the first design decision is the choice between a CPU architecture, which executes a protocol control program from a read-only memory (ROM) or a fixed hardwired finite-state machine (FSM) based approach. A CPU architecture provides more flexibility to adapt the design to changes in the standard and to implement further custom commands, but at the cost of higher complexity and therefore higher power consumption. In order to meet the area and power constraints, an FSM-based design is used.

A second general design decision is the word width for exchanging data with the EPC memory and the cryptographic units. The DecodeEncode unit provides and requests a bit stream which would not be practical for the internal communication. The standard organises the EPC memory in 16-bit words and there are several 16-bit data values like the handle value and also the CRC-16 results. On the other hand, the EPC address format uses an 8-bit based EBV and the IP module of the AES provides an 8-bit interface. Also the final Grain implementation uses an 8-bit architecture. Therefore, the design uses an 8-bit datapath for data exchange between the modules.

On a commercial RFID tag, the EPC memory banks are implemented as NVM which was not available on the ASIC process used for this prototype. Therefore, standard registers emulate the behaviour of a real-world tag. They are initialised to a valid EPC value at power up and can also be accessed over the test interface.

Figure 4.5 shows an overview of the main control unit. It implements the control logic with the modules Control-FSM, CMD-Decode, AESControl and GrainControl. Additionally, it contains an EPC-standard slot counter, two CRC modules, and an EPC-memory register file. Finally, there are small additional modules like an address counter, intermediate registers, and xor/compare functions.

4.4.1 Control Logic

Since the protocol is quite extensive and considering the additional authentication mechanisms, the design implements the control logic for this unit in four separate FSMs. The CMD-Decode unit processes the header bits of all incoming frames. First it parses the first 2 up to 16 bits to determine the command code. Furthermore, it processes header bits concerning the demanded encoding settings and control bits for the anti-collision routine. The slot counter, CRC-5 module, and status bits for the encoding process are managed by this unit. It consists of an FSM with 71 states and an additional 1-byte status register.

The AES and GrainControl modules handle the control signals for the cryptographic units to decrease the complexity of the main control FSM.

Control-FSM is the main control unit. It implements the tag states (Section 2.7) and the command handling. It also coordinates the enable signals for all other units in the design to reduce activity in

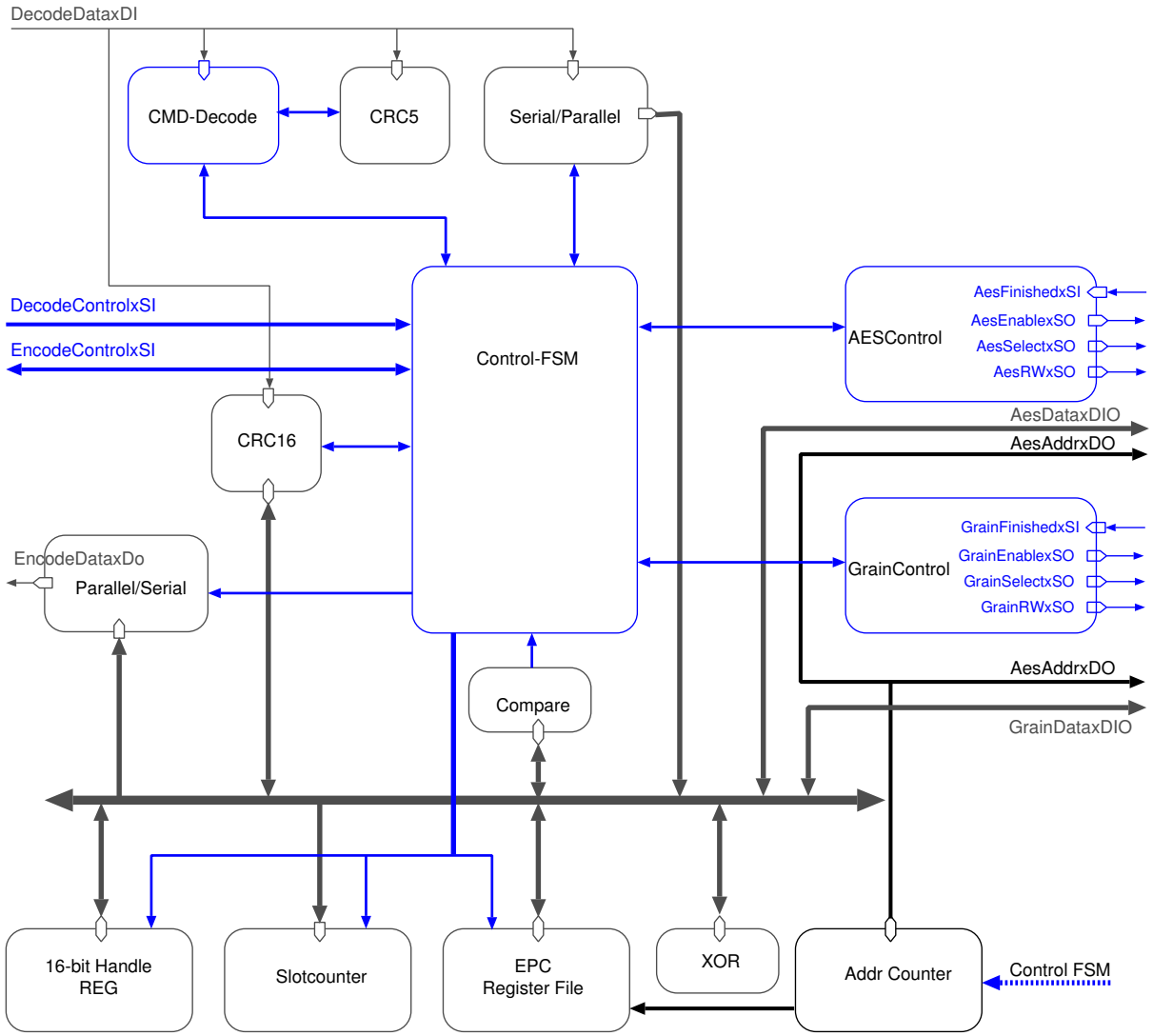


Figure 4.5: Overview of the main control unit.

modules which are idling. The Control-FSM is realised as a nested state machine with two state registers, FSM_reg1 and FSM_reg2. This improves the maintainability of the long source code and has good power characteristics because only one state register is clocked during Idle mode. The main state signal (FMS_reg1) implements 45 states, representing the current tag states and the processed command. The sub-state signal (FSM_reg2) with 75 states sets all control signals during command processing.

4.4.2 Cyclic-Redundancy Check

The standard defines two cyclic redundancy checks (CRC) for transmission-error detection. A CRC is a value, appended to a message, based on the reminder of a polynomial division. It is easy to implement in hardware and the quality of error detection is dependent on the number of bits used for the checksum. The standard uses a custom 5-bit CRC for the Query command only. Therefore, the CRC-5 module implements verification only.

An ISO/IEC 13239 CRC-16 is used for error detection in several reader frames and also tag answers. The CRC-16 module can verify the checksum of an incoming frame and also calculates the checksum for a tag answer. Moreover, at every startup of the tag the CRC-16 of the EPC memory is recalculated as defined in the EPC Gen2 standard.

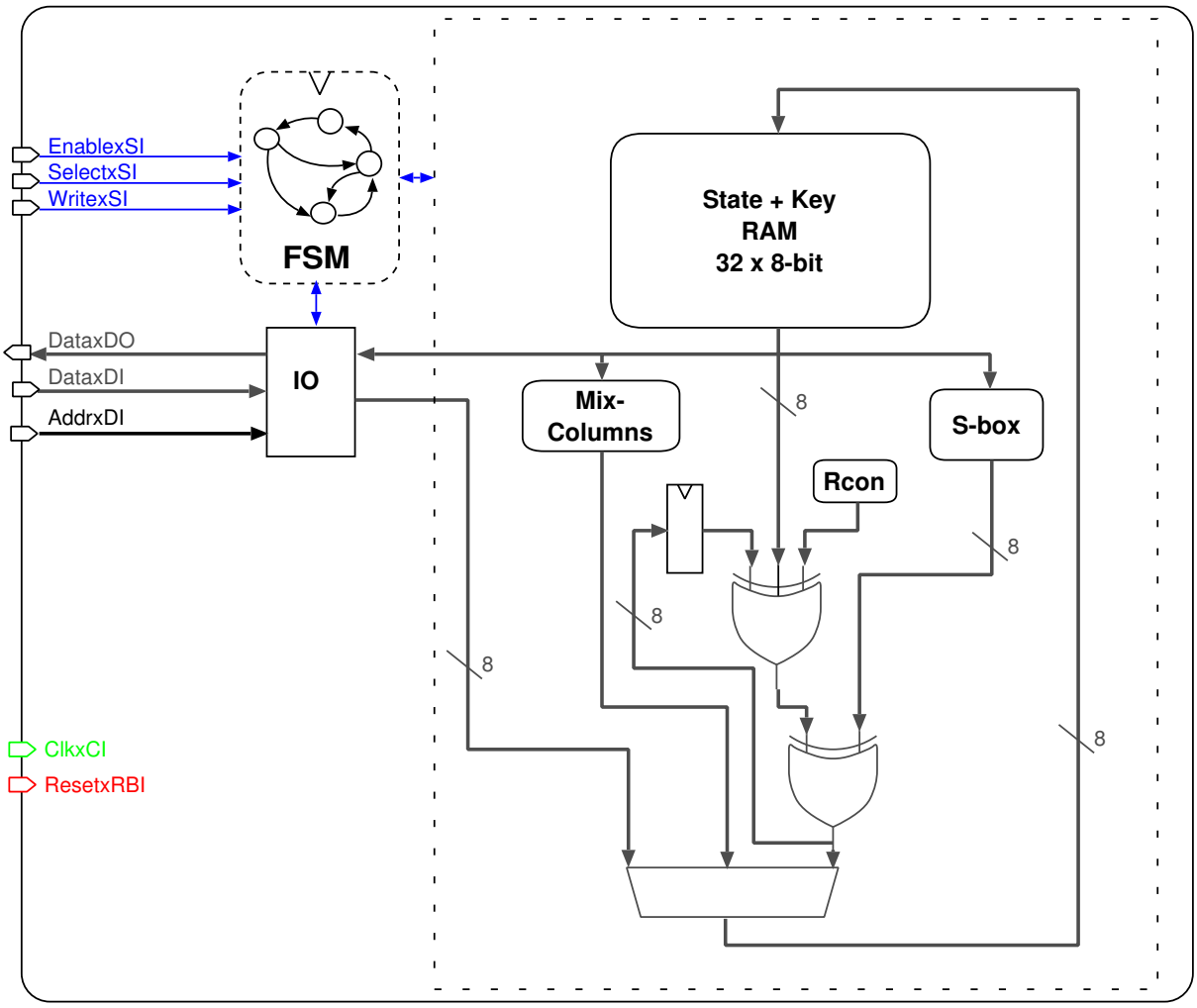


Figure 4.6: Architecture of the AES unit.

4.5 AES

This tag design uses an IP module presented by Feldhofer et al. [2005], which was adapted for this use case. Developed with focus on the constraints for passive RFID-tag implementations, it introduces an 8-bit architecture and supports encryption, decryption, and cipher block chaining (CBC) mode with an area usage of 3 400 GE.

In this work an encryption-only version of the AES module is used. Figure 4.6 shows an overview of the architecture. The biggest part is the 32x8-bit random-access memory (RAM). The module uses a flip-flop based approach instead of a dedicated RAM-macro file because higher area efficiency is only valid for larger RAMs. The question of memory structure was reevaluated for the used ASIC process. The UMC 130 nm process supports custom RAM macro cells with individual size and port bit width. Still, the area overhead for placing a small 32x8-bit memory using a macro cell turns out to be bigger than the area savings of macro RAM cells in comparison to flip-flops. A way to decrease the area of the flip-flop RAM is the usage of clock-signal-high sensitive latches instead of edge-triggered flip-flops. Latches can be used instead of flip flops without changing the control circuit of the design, but an additional 8-bit register at the input port of the RAM needs to be added and has to be clocked every write circle. Synthesis shows an area reduction of 18% for the RAM and 12% for the AES core. The power-consumption estimation predicts an equal or even slightly worse behaviour of the latch design because of the additional register at the input port.

The datapath and control logic is unmodified and was provided as RTL VHDL model by IAIK. It is a small AES implementation with a very low power consumption. The datapath implements the basic AES operations SubBytes, MixColumns, AddRoundKey, and KeyScheduling. The S-Box implementation is the biggest part of the datapath and is realised as combinational logic in contrast to a straightforward ROM lookup implementation. This results in a lower power consumption and additional sleep logic eliminates switching activity during idle. The MixColumns implementation uses only one instead of four multipliers and processes one column in 28 clock cycles [Feldhofer et al., 2005].

A full encryption of one 128-bit block takes 1024 clock cycles which equals $\sim 3\,\mathrm{ms}$ or $\sim 40\,\mathrm{kbit/s}$ at the chosen $440\,\mathrm{kHz}$ clock frequency. Vulnerabilities to side-channel attacks are discussed in the next chapter in Section 5.3.

4.6 Grain

A straightforward hardware implementation of the Grain cipher as described in Section 3.5 is simple. Still, there are several different possible implementations for lowering the power consumption per output bit. Therefore, the design criteria for the architecture is to find an optimal trade-off between the average power consumption, chip area, and number of clock cycles per random bit. Additionally, the overhead for the interface to the rest of the design has to be considered. The different architectures are compared using the same 8-bit AMBA interface and a fixed throughput. The clock frequencies are chosen to generate $640\,\mathrm{kbit/s}$ of pseudo-random data. This is the maximum data rate supported by the standard and therefore no buffers are needed during the challenge exchange process.

In a standard Grain implementation, there is one instance of the feedback and output functions. All 160 shift registers are clocked every cycle which results in one pseudo-random bit. For increasing the throughput of the cipher, the design allows up to 16 parallel instances of the feedback functions. In an RFID environment, throughput is not the main goal but this option can be used to find a suitable trade-off between the average power consumption and the area usage. In order to decrease the number of active registers at the rising clock edge, the registers are separated into groups using gating cells and clocked consecutively. The number of active flip-flops and gating cells n is:

$$n(b) = \frac{160}{b} + b$$

with b clock gates [Feldhofer, 2007]. Minimising this equation results in $\mathrm{b} = \sqrt{160} = 12.6$ flip-flops per clock gate. In practice only 8 and 16-bit solutions are interesting because otherwise there would be too much overhead for a standard interface.

Figure 4.7 shows the fundamental structure of the five different evaluated architectures: a.) Serial, b.) Radix-8, c.) Radix-16, d.) Radix-16-Mux, and e.) Radix-8-Mux. The serial implementation is the standard version of a FSR. There is one instance of the feedback function and all registers are clocked every cycle, resulting in one output bit. The Radix-8 version uses 8 instances of the feedback function and the shift register consists of 10 8-bit registers. The registers are not all clocked simultaneously but in a way that the design produces 8 bits every 8 clock cycles. This results in only ~ 20 active flip-flops per clock cycle compared to the 160 in the serial implementation using the same clock speed and producing the same output data rate. Radix-16 and Radix-16-Mux separate the FSR into blocks of 16 bits. With a clocking strategy that again generates effectively 1-bit-per-cycle, it is possible to use latches instead of edge triggered registers because only one block is clocked at a time and the input values of an active register do not change during the active clock cycle. Implementing 16 instances of the feedback function results in a high area overhead and therefore the Radix-16-Mux implementation is a compromise between the Radix-8 and Radix-16 suggestions, by multiplexing the 16-bit blocks into 8 instances of the feedback function. Katti et al. [2006] propose a low-power architecture for FSRs where instead of shifting the input data, the feedback is selected and written back to the right place without shifting, as indicated by

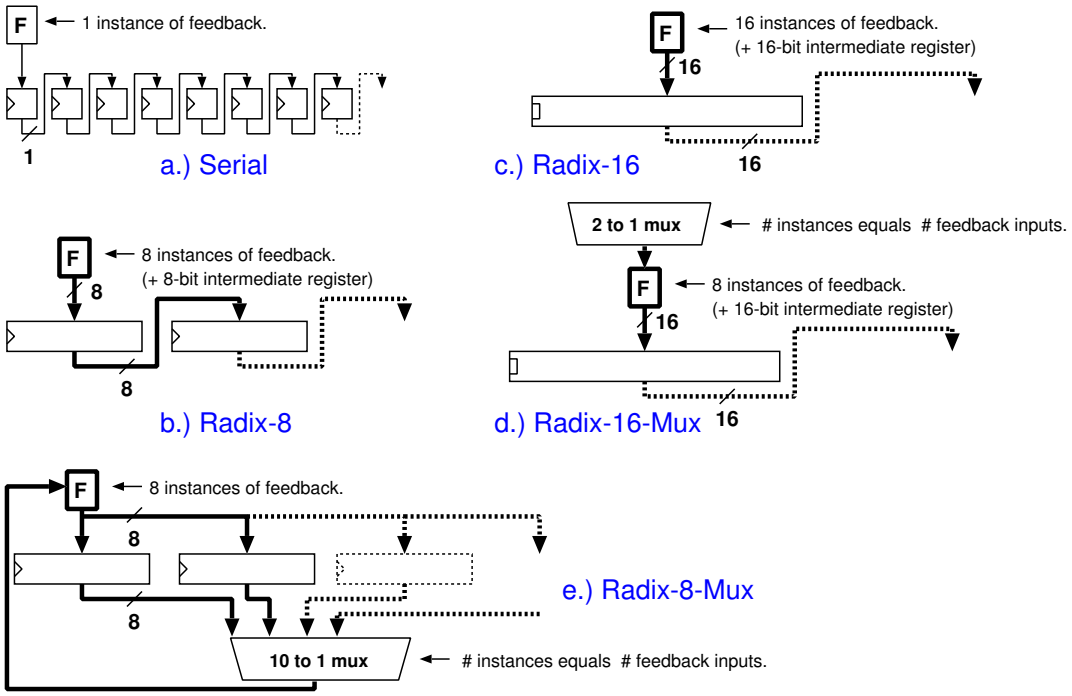


Figure 4.7: Principle structure of the five different FSR architectures evaluated.

the Radix-8-Mux schematic. Again 8 instances of the feedback functions are used because of practical reasons concerning the interface. With this architecture only 16 registers are active in one cycle and it produces 8-bits-per-cycle.

Figure 4.8 shows the simulation results for all five structures. The area is measured in gate equivalents (GE) and the power consumption is simulated for an output data rate of $640\,\mathrm{kbit/s}$. All designs provide an 8-bit interface. The serial implementation results in just above $1\,000\,\mathrm{GE}$ but a high power consumption of $3.5\,\mu\mathrm{W}$. Increasing the number of instances of the feedback function reduces the power consumption significantly due to lower activity of the 160 registers but at the cost of higher area usage. The Radix-16-Mux idea performs worse in power consumption and area usage than the Radix-8 version and can be dismissed. The idea with multiplexing the feedback instead of shifting through the register in the Radix-8-Mux suggestion indeed results in the lowest power consumption. Unfortunately it turns out that because of the high number of inputs for the feedback, in comparison to FSR with more simple

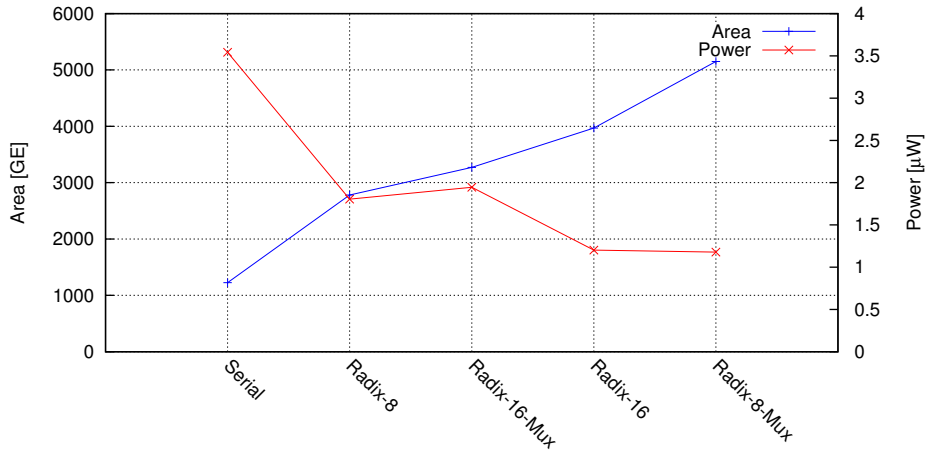


Figure 4.8: Area usage and power consumption of the five different FSR structures.

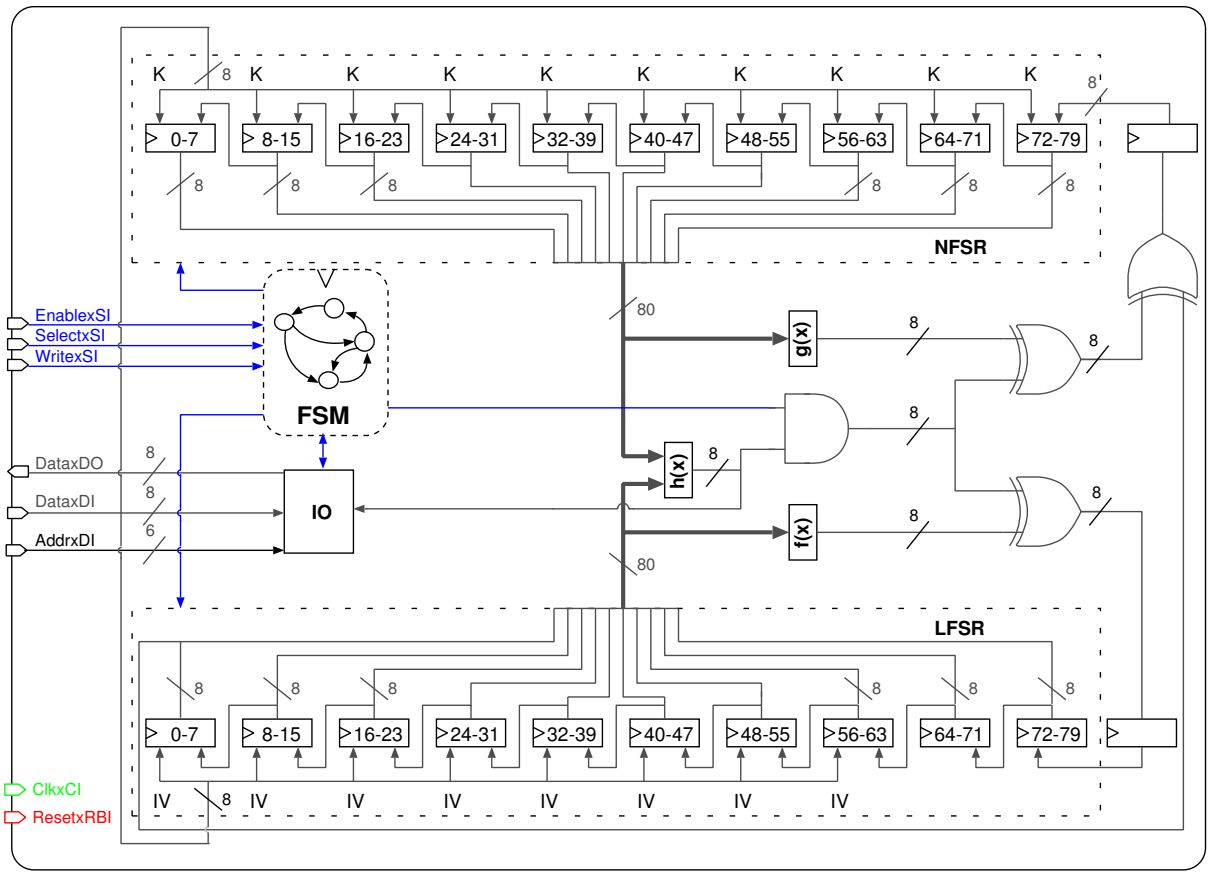


Figure 4.9: Detailed architecture of the Grain unit (Radix-8 version).

feedback functions, the overhead for the input selection results in an unacceptable high area usage.

Considering the area constraint for the whole tag design, a gate count of more that 3 000 GE for the PRNG unit is unacceptable. The Radix-8 version shows a decent result, when comparing the area-power products and has a low overhead for the interface. Figure 4.9 shows a detailed overview of the final architecture. Both FSRs consist of 10 8-bit registers and 8 instances of each feedback function. The FSM handles the AMBA control signals and data input/output handling. The key and the IV value can be written directly over the AMBA interface and after the initialisation that requires 160 clock cycles, the architecture produces one pseudo-random byte every 8 clock cycles.

4.7 ClockDivide

This small component provides the different clock frequencies for the design. The DecodeEncode unit runs at the system clock frequency provided by the analog front-end. For the main control unit and the Grain unit, ClockDivide generates a 4 times slower clock. The AES unit runs at $1/8$ of the system clock frequency. Additionally, it is possible to write one byte of data into this unit in order to randomise the timing of the AES clock. Implementation details and explanation of the clock randomisation are provided in the next chapter.

Chapter 5

Implementation

The final architecture of the secure tag controller is implemented in a $130\,\mathrm{nm}$ standard-cell ASIC process. The IIS at ETHZ provides the infrastructure, know-how, and $1\,\mathrm{mm}^2$ die area for student projects. This chapter discusses various aspects of the process from the architecture to the tape-out-ready chip layout. First, it describes the verification process and the test setup throughout the implementation process. Then it discusses methods to decrease the power consumption of the final circuit. Section 5.3 provides a brief explanation of possible attacks on the security using side-channel analysis and describes the implemented countermeasures used in this design.

Testing of the final ASIC after production has to be considered during the design process as Section 5.4 points out. The verified RTL model of the design goes through multiple steps during back-end design to get the final fab-ready chip layout. Section 5.5 discusses different aspects of this highly automated process. Finally, the simulation results regarding area usage and power consumption are presented and compared with related work.

5.1 Functional and Protocol Verification

Designing and producing a microchip is a time consuming and costly endeavour. Therefore, detailed verification of correct functionality is necessary during the whole development process. First, the specifications have to conform to the requirements of the project. During the design phase, the model is consecutively verified if it behaves as expected. All steps from the high level VHDL model, the RTL description, to the synthesised netlist have to be simulated and tested. Finally, testing after production is necessary to find fabrication faults [Kaeslin, 2008].

The functional specification for this prototype is the EPC C1G2 standard extended with the custom commands presented in Section 3.3.3. Additionally, the official reference software implementation of the two ciphers AES and Grain are used to verify the correct functionality of the cryptographic units. Besides standard-compliant behaviour, there are two main constraints for the resulting circuit. First, the power consumption should be below $10\,\mu\mathrm{W}$. Second, the overall chip area should not exceed $15\,\mathrm{kGE}$.

The Microelectronics Design Center provides a testbench approach for verification of the design during development. A testbench, also modelled in VHDL, simulates the environment of the circuit in a real-world application. It provides all input signals, including the clock and reset signals. All output signals are sampled every clock cycle. A software model, described in the next section, generates valid data for all input signals and expected output values for every clock cycle. This data is generated before a simulation run and the testbench fetches and applies one set of input signals after every rising clock edge. The output values of the model under test (MUT) are sampled shortly before the next rising clock edge and compared with the expected values (cycle-true testing). Figure 5.1 shows an overview of the whole test setup.

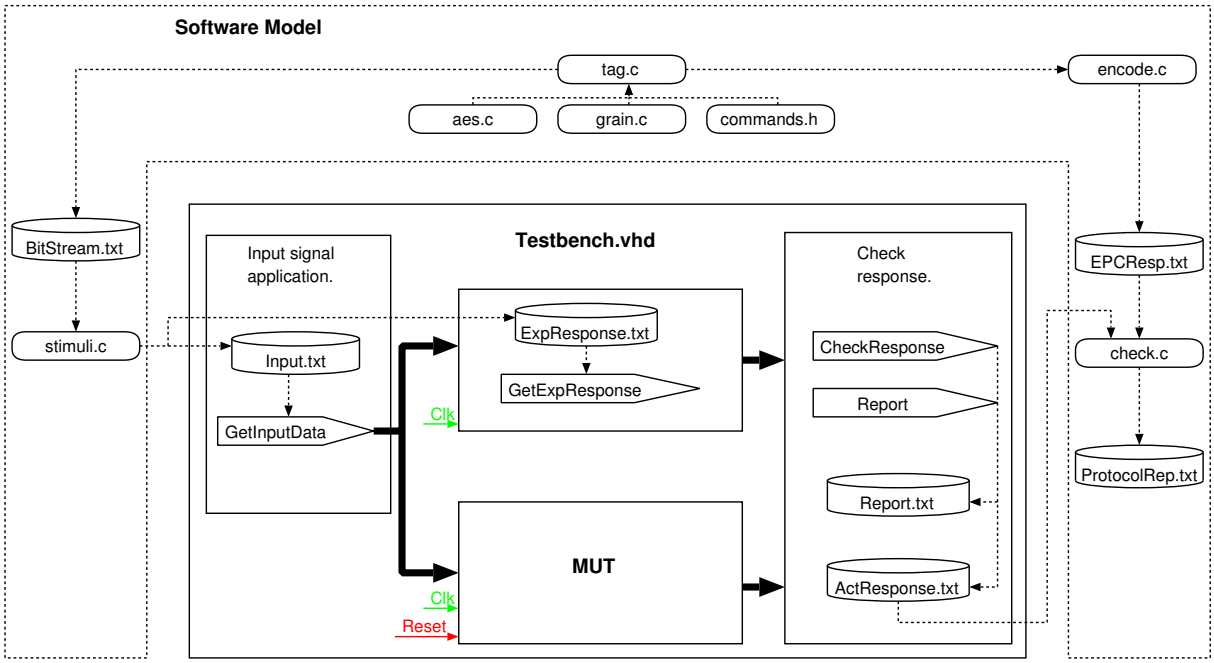


Figure 5.1: Overview of the testbench including the software model.

5.1.1 Software Model

For valid input and expected output data during simulation, a software model generates an EPC-standard compliant communication between reader and tag. The official reference implementations of the AES and Grain cipher provide valid test data during the authentication process. The software model generates a test file, containing bit streams of sample data-exchange flows including valid header information. This approach of generating fixed sample communication flows is a compromise between development effort for the software model and flexibility during testing. The main disadvantage of this approach, instead of writing independent models for a standard-compliant reader and tag, is the difficulty to automate testing of all possible real-world use cases. For example, multiple tags entering the field, communication errors in different parts of the message, or all possible combinations of valid and invalid command sequences. Still, the fixed sequence of reader and tag commands contains several different anti-collision scenarios and authentication command sequences, without making the overall simulation time impractical in everyday usage.

Encoding the bitstream of the generated reader commands and corresponding tag answers results in valid signal data for all input and output pins. The generated test files contain data sets for every clock cycle, and the testbench uses this information to apply data to the input pins and compare the output-pin values to the expected values from the software model. Furthermore, the tag responses are automatically checked if they conform to the EPC-standard constraints, like response time or tolerance limits for the BLF. Again, the static test approach limits the possible test cases. Although tested for various frequencies and data rates in both directions and combinations of all possible encodings, real-word scenarios like changing frequencies during communication or all kind of interferences are difficult to simulate using a static software model.

5.1.2 Rapid Prototyping

Simulation of the design and comparison to software generated reference values is a powerful tool to find errors during the design process. Still, errors in the software model or false assumptions about the specification would produce a malfunctioning chip. Software models can also hardly imitate all possible

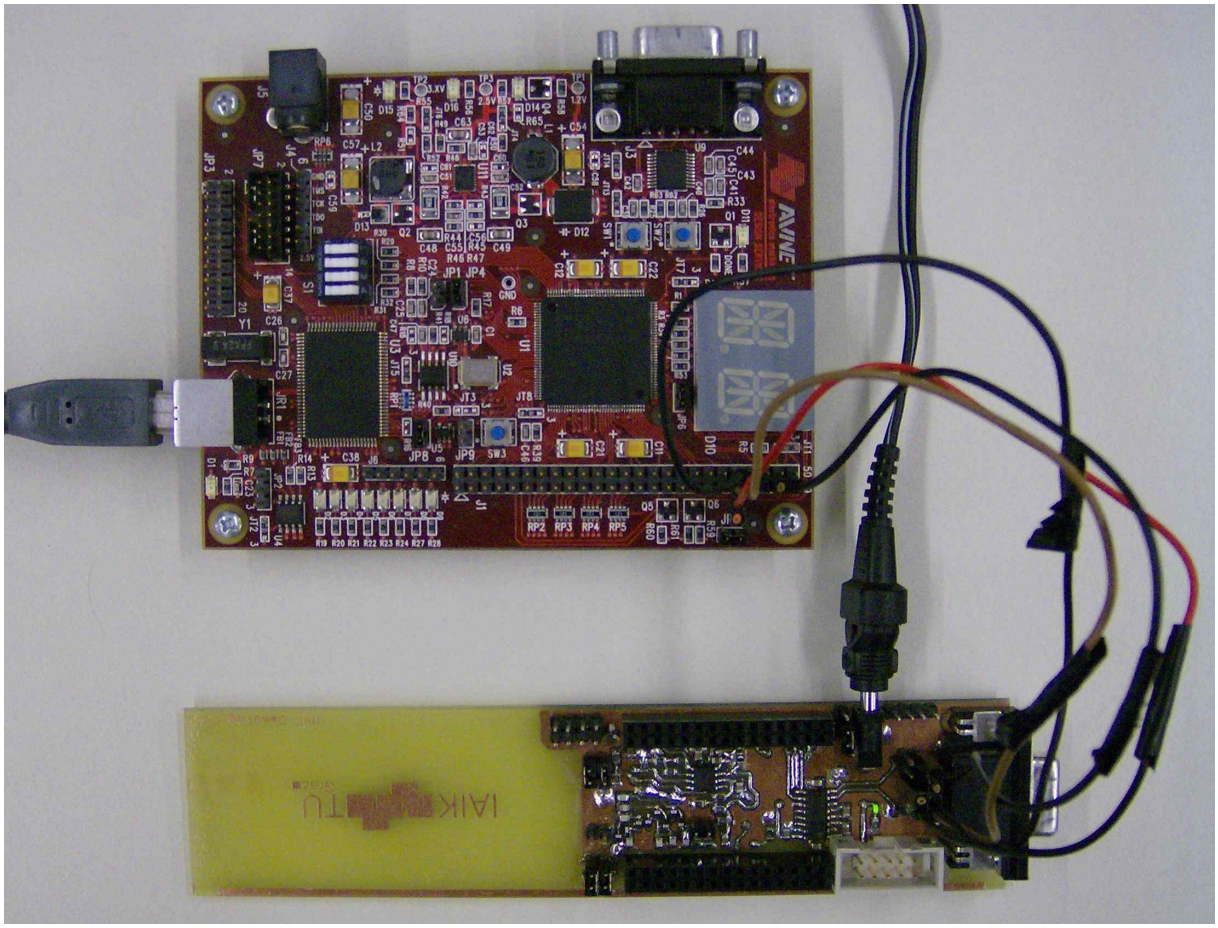


Figure 5.2: IAIK UHF DemoTag connected to an Avnet FPGA evaluation board.

real-world scenarios.

Using an FPGA, it is possible to emulate the behaviour of the circuit and test its functionality in real-world applications, before producing a costly ASIC. Testing was done at IAIK using an Avnet evaluation board with a Xilinx XC3S100 FPGA connected to the analog interface of the IAIK UHF DemoTag. Figure 5.2 shows a picture of the evaluation board connected to the analog front-end of the DemoTag. Successful inventory sequences with a commercial UHF EPC-compliant reader from CAEN verified correct implementation of the de-/encoding process and correct response-time behaviour of the design.

5.2 Low-Power Design Methods

As already stated in the previous chapters, a low power consumption of the tag is crucial for the overall performance of the RFID system. Bringing the power consumption down from the general constraint elaborated in Chapter 4.1, enables higher operating ranges and a more stable communication link in noisy environments. Beside the average power consumption, it is important to avoid power consumption spikes which could overload the power-retrieving circuit and cause resets.

The power consumption of CMOS circuits is composed of **static** and **dynamic** energy dissipation. Static power consumption is independent of the state and activity of the circuit and results from the leakage currents of the transistors. Power estimations have shown that the static power consumption is below 1% of the overall consumption using the low-leakage standard cell library and is therefore ignored during the design process. The dynamic power-consumption results from charging and discharging capacitive

loads, driving of resistive loads, and crossover currents [Kaeslin, 2008]. The main factors for dynamic power consumption can be written as the following relation:

$$P_{dyn} \propto f_{clk} * \alpha * C_{load} * U_{dd}^2$$

Since the dynamic power dissipation occurs at state changes of the CMOS circuit, it is proportional to the effective clock frequency f_{clk} and the switching activity α. C_{load} is the switching capacity of the circuit which is dependent on the size of the circuit and driving strength of the standard cells. Finally, the supply voltage U_{dd} plays a big part and is mostly dependent on the used technology. The UMC $130\,\mathrm{nm}$ CMOS process uses $1.2\,\mathrm{V}$ as a standard supply voltage. The small size of the circuit and very low clock frequency should allow to decrease the operating voltage. Unfortunately, the synthesis and simulation setup did not support simulation and verification with values below the standard supply voltage. Therefore, only practical tests can show a possible decrease in supply voltage in order to lower the power consumption. Comparable silicon implementations of RFID circuits like for example Yongzhen et al. [2009], Ricci et al. [2008], or Zhang et al. [2008] with a similar technology report successful operation at lower than standard supply voltage.

Clock Gating

Reducing the logical clock frequency is an effective way to decrease the power dissipation of the circuit. As already mentioned in Section 4.1, the minimum clock frequency for the digital part is defined by the minimum sampling accuracy of the synchronisation frames. This results in a $3.5\,\mathrm{MHz}$ clock frequency for the DecodeEncode unit. The other parts of the design run at lower clock frequencies. The main controller unit has to be able to process incoming and provide outgoing bitstreams at the highest possible data rate defined in the standard. This constraint results in a minimum clock frequency of $1/4$ of the system clock. The same applies to the PRNG unit (Grain) which generates one random byte within 8 clock cycles. The interleaved protocol design of the authentication process loosens the time constraint of the AES core for one encryption calculation. The drawback of a long time interval between starting an authentication and requesting the encrypted answer is a performance loss of the overall system. Feldhofer et al. [2005] suggest a $100\,\mathrm{kHz}$ clock for the proposed AES core on passive RFID tags which results in about $10\,\mathrm{ms}$ encryption time. The first power simulations showed that due to the advances in VLSI process technology a clock frequency of $500\,\mathrm{kHz}$ is possible without exceeding the power consumption during de-/encoding. Since the other components are mostly inactive during the encryption process, the clock frequency for the AES unit is set to $1/8$ of the system clock.

Using different clock frequencies within one design poses a lot of obstacles during implementation. Synthesis tools cannot handle different clock domains and automatically verify possible timing violations. Fortunately, the low system clock frequency and small circuit size allow a synthesis of the whole design with the timing constraints of the system clock frequency without increasing the size of the circuit or driving strength of the standard cells, which would increase the power consumption. This allows to generate lower logical clocks, using the standard-library clock-gating cells. Clock gating means to disable the clock signal for parts of the circuits which are inactive. Figure 5.3 shows a clock-gating cell and an example wave form. The gated clock signal results from an AND conjunction with an enable signal. For avoiding glitches on the clock signal, a latch captures the enable signal before the critical rising clock edge. The ClkDivide unit uses standard gating cells from the library to generate signals with one clock pulse every 4 and every 8 clock cycles. The clock gating on register level is handled automatically by the synthesis tool using gating cells on every register modelled with enable signals. Different logical clock frequencies and extensive clock gating on register level result in a 75% lower power consumption in comparison to the straightforward implementation.

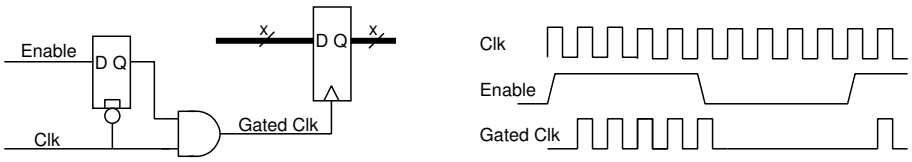


Figure 5.3: Clock gating using a latch to avoid glitches and corresponding wave forms.

5.3 Side-Channel Analysis Countermeasures

Using strong cryptographic algorithms results not necessarily in a secure authentication protocol. Standardised algorithms like the AES are heavily tested against all known state-of-the-art attack methods on algorithmic and mathematical level. Differential or linear cryptanalysis tries to exploit statistical correlation between the plain-text block and secret key and the corresponding encrypted block. These methods consider the computation as a black box and until now no methods have been found to speed up the secret-key search significantly over brute-force try-and-error approach.

However, in practice an AES implementation is not a black box that leaks no information about the internal state during an encryption. When executed as a software implementation, the execution time can correlate with the processed data, including the secret key. Hardware failures or faults caused on purpose by manipulation of the supply voltage or clock speed can possibly reveal internal information. Additionally, cryptographic algorithms are implemented and executed on electronic devices where an attacker can easily measure any physical parameter during execution like electromagnetic emissions or power consumption [Kocher et al., 1999].

In passive RFID systems, the antenna is the only interface to the outside which limits the possibilities for an attacker to acquire side-channel information. Still, measurements in the RF field allow assumptions about the power consumption of the tag during data processing. Therefore, power-analysis methods are applicable. Kocher et al. [1999] first described advanced methods to determine the secret key using accurate power-consumption measurements. As already explored in Section 5.2, the power consumption of a CMOS circuit depends on the processed data. The main power dissipation is caused by signal changes and so correlates with the processed data at a given point in time. A straightforward way to retrieve side-channel information is to interpret one or a few power traces directly (simple power analysis). This requires a detailed knowledge about the internal structure of the algorithm implementation in order to make good guesses about the secret key. Detailed information about the internal architecture is often not available and this approach is susceptible to noise from other parts of the chip. A more sophisticated attempt is differential power analysis (DPA). By using a large number of power traces of the same time interval and applying statistical methods, it is possible to correlate these measurements with values of the internal state even without having a detailed knowledge of the exact architecture and in presence of noise.

The feasibility of DPA attacks on EPC C1G2 tags was already shown by Plos [2008]. With a standard antenna that measures the changes in the RF field, it is possible to determine data processed by a commercial UHF tag. Even though an advanced authentication protocol with standardised cryptography increases the security properties of the RFID system, the threat remains the same. In case of an AES-based symmetric authentication, the security relies on the secret 128-bit key. Considering the structure of the AES algorithm a realistic attack scenario would be to target 8-bit blocks of the secret information used in the first round of the encryption. Feldhofer and Popp [2008] state that it only takes $60 - 1\,000$ power measurements to determine most of the secret-key bytes of the unprotected AES IP core used in this design. Inbuilt in the tag circuit the power measurements are more difficult because of the indirect measurements in the RF field and additional noise of the DecodeEncode and Controller units. Still, it remains a very realistic attack scenario on the system.

5.3.1 Possible Countermeasures against Power Analysis Attacks

Power analysis attacks exploit the fact that the power consumption of a CMOS circuit correlates with the internal data currently processed. In case of the AES implementation, the internal data is linked to the secret key. There are mainly two ways to cut this connection, **masking** and **hiding**. [Feldhofer and Popp, 2008]

Masking randomises the internal data by using internally generated random numbers to mask the sensitive data during execution. Even though the power characteristic of the circuit is still depended on the data, the measured traces differ in each execution because the masking values change every time.

Hiding on the other hand tries to disconnect the power consumption from the internal values. One possibility is to implement logic gates which have the same power characteristics for all possible input values. Using custom cell libraries, it is possible to synthesise the circuit without major changes of the implementation itself and it does not effect the throughput. The main disadvantage is that the additional chip-area overhead for this approach is significant. The other possibility to hide side-channel leakage is to move the point-in-time of the execution randomly. Depending on the algorithm, it is often possible to change the execution order of individual instructions (shuffling). A further approach is to insert dummy cycles that process random data in between the actual encryption. Hiding in the time domain requires little additional circuitry but usually lowers the throughput of the implementation.

Feldhofer and Popp [2008] implemented a DPA-resistant version of the AES core by using hiding through randomisation, a secure logic style, and masking. It is important to note that DPA attacks cannot be fully prevented because no countermeasure cuts the connection to the processed sensitive internal data completely. The security gain through countermeasures is usually specified as an estimation of how many additional power measurements have to be made in order to determine the internal value. In practise this results in a trade-off between gain in DPA resistance and additional area overhead and lower throughput.

Considering the area constraint of $< 15\,000\,\mathrm{GE}$ for the whole design, it is not possible to use all proposed countermeasures for this implementation. Applying hiding techniques using a secure logic style requires about 7 times more GE and masking would require a second RAM where the random values are stored. This leaves hiding techniques in the time domain since throughput is not a major factor in passive RFID applications and the interleaved protocol design allows low application performance loss even if the encryption time is increased.

5.3.2 Implemented Randomisation Countermeasures

Considering the internal 8-bit structure of AES, a DPA attacker would target one byte of sensitive data processed at a certain point in time of the execution. For obtaining enough side-channel information, up to a few thousand power measurements of the same execution are needed. If the sensitive byte is processed randomly at different points in time, the same correlation results require more measurements.

The design of Feldhofer and Popp [2008] that is the basis for our IP block implements two methods of randomisation countermeasures. Shuffling changes the execution order of operations of the algorithm and dummy cycles change the execution point in time of sensitive information every round. Random data is generated before every encryption round by the Grain unit and written to the AES memory. Figure 5.4 shows how the execution order can be altered in this 8-bit architecture. For one operation, 4 bits of random data alter the starting address of the RAM. This requires almost no overhead except the storage for the random bytes. It can be accomplished by initialising the address counter for row and column selection during execution with the random value instead of starting always at address $0\mathrm{x}00$ of the state memory.

Dummy cycles execute between $0 - 16$ operations before and after the actual calculation. Instead of using the state values, the operations are performed on a random 4-byte dummy state. Figure 5.5 shows the difference between the standard execution time line and the version with dummy operations in

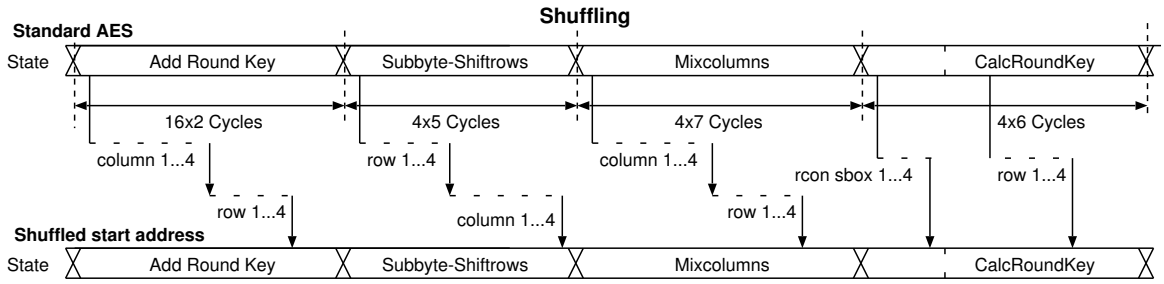


Figure 5.4: Randomisation during AES encryption by altering the start address (top: standard AES, bottom: execution with randomised start address).

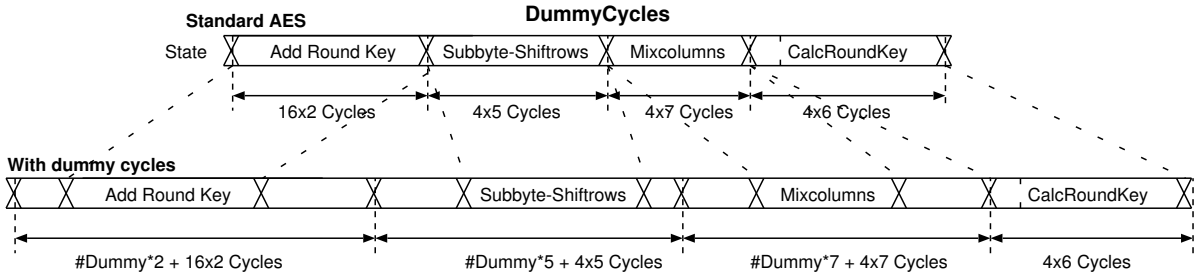


Figure 5.5: Randomisation during AES encryption adding a randomised number of dummy cycles before and after an operation (top: standard AES, bottom: with dummy cycles).

between. Again the additional hardware costs are low except for the additional 4 bytes of dummy state in the RAM. In contrast to randomisation of the execution order, inserting dummy cycles lowers the throughput of the circuit. Using 16 dummy operations in four rounds results in about twice the number of clock cycles for a full encryption. Still, the resulting $6\,\mathrm{ms}$ encryption time at $1/8$ of the tag's system clock meets the constraint set beforehand.

Beyond the randomisation implemented in the IP model, the architecture of the RFID controller design allows another simple randomisation countermeasure at no cost. The clock frequency of the AES unit is set to $1/8$ of the system clock by the ClockDivide unit using clock gating. Instead of one clock cycle exactly every 8 system clock cycles, the ClockDivide unit fetches random data from the Grain unit and generates one impulse randomly in the 8 clock-cycle period. The resulting clock frequency for the AES unit remains the same, but the time interval between two cycles varies. Figure 5.6 illustrates the differences between the standard AES clock waveform and the randomised version.

5.3.3 Analysis of the DPA Countermeasures

Now it is time to analyse the security gain of the randomisation techniques in comparison to the implementation overhead. Since only randomisation in the time domain is implemented, the additional resources for the implementation with DPA countermeasures are very low. The increase of chip area is less than $10\,\%$ which results mainly from the additional 9 bytes of RAM storage in the AES unit and some changes in the control logic. The average power consumption for one authentication round re-

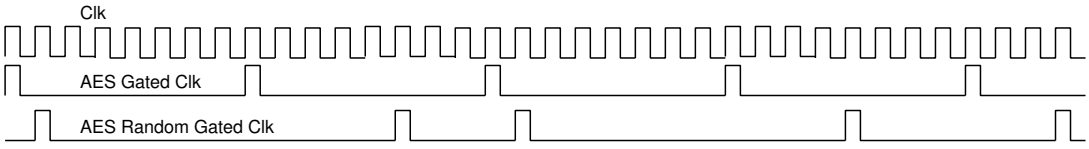


Figure 5.6: Example waveform of the random clock gating for the AES unit.

mains almost unchanged and does not affect the maximum reading distance of the passive RFID tag. The main performance loss is the doubling in encryption time but as already stated, the interleaved protocol decreases the performance loss in authenticating numerous tags per second.

An important question is, how many additional measurements have to be done in comparison to the straightforward implementation without countermeasures. In general, successful DPA attacks depend on various factors. Besides circuit-specific characteristics like architecture, technology, and noise, the setup plays a large role. More accurate measurement, preprocessing of the power traces, and better power-consumption models will speed up the process. Therefore, countermeasures are evaluated as a factor of additional measurements in comparison to the standard implementation and not in absolute terms.

Considering the implemented shuffling, there are 16 possibilities for one specific byte of sensitive information to be processed. Consequently, only one in 16 power traces has a specific byte at the same position. In a standard DPA setup, $16^2 = 256$ times more power traces are needed to receive the same results. The usage of advanced attack techniques like windowing, reduces this factor to 16. The same applies to the insertion of dummy operations. For example the first byte of the key schedule is added to the state either in the first 2 clock cycles of the AddRoundKey operation or after up to 15 dummy calculations executed with random data values. Combining both randomisation countermeasures results in a 1 in 76 chance for the same byte of data to be executed at the same time in two separate encryption runs [Feldhofer and Popp, 2008]. The difference to $16 * 16 = 256$ possibilities result from the fact that several combinations of dummy operations and shuffling result in the same power trace for the relevant byte.

The security gain from the random clock gating is lower than the theoretical 8 possibilities a clock pulse can occur. It is possible to distinguish between the active clock period and the other seven clock cycles where the AES core is inactive. Hence, preprocessing the power traces can possibly eliminate the inactive cycles and align power traces with different clock-gating patterns. The security gain depends on the DPA setup and poses only practical obstacles to an attacker.

In conclusion, the constraints to chip area and power consumption only allow modest DPA countermeasures at low additional hardware costs for this class of passive RFID tags. Since the AES unit can only be accessed over the slow RFID interface, any side-channel attack is slowed down. Practical experiments will show if the combination of low throughput and the implemented randomisation is enough to make DPA attacks on the design impracticable.

5.4 Design for Test

Every produced microchip has to be verified individually before shipment. There are several reasons why post-production testing has to be considered during the design process of an IC. Increasing complexity packed on smaller chip area, limited number of pins, and limited time for testing during production make test and verification of ICs a difficult task. Testing has also become an increasing part of the overall production costs because of the extensive test equipment and the production delay caused by long tests of high-complexity circuits.

There are general guidelines for IC designs to improve testability. Synchronous designs are easier to test and therefore all clock gates and the ClockDivide unit can be disabled during test. A global reset signal allows to set all registers in the design to a defined start state. Redundant circuitry is often not testable and has to be avoided.

Simplified fault models decrease the complexity of testing. Common practise is to assume only one fault at a time and to reduce possible errors to stuck-at-0 and stuck-at-1 faults. This models shortcuts of circuit nodes to either ground or power. In order to achieve a high fault coverage, a good test sequence should force signal changes in every node of the circuit and detect false behaviour if a stuck-at fault is present. With circuits like for example large state machines and counters, it is often difficult to reach

every state within a reasonable time interval. Scan chains try to reduce this problem by connecting poorly controllable registers in the design to a shift register. The costs for this test strategy are an increased routing overhead and a multiplexer at every register input signal to switch between test and normal mode [Felber, 2008].

An RFID chip has a very low overall complexity which makes testing less challenging. On the other hand, the requirement for very low production costs and the large unit numbers requires a very fast and efficient test strategy. This results in a partial scan-path strategy. All RAM registers of the design are easily accessible through the 8-bit AMBA test interface and therefore excluded from the scan chain. The other registers are connected to one 374-bit scan chain.

The automated test-vector generation results in a fault coverage of $98.43\,\%$ using 566 scan-chain patterns and $5\,909$ general patterns. Description of the input-pin settings during test can be found in the datasheet (Appendix A).

5.5 Synthesis and Backend Design

Synthesis is the process to generate a standard-cell netlist of the circuit from the hardware description at RTL. Back-end design transforms the standard cell netlist to a fabricable layout. This includes positioning of the pad-frame, placing and routing of the cells on the chip, power distribution, and clock-tree insertion. Additionally, multiple verification steps, like design rule check, layout-versus-schematic check, post-layout timing verification, and power grid and signal integrity analysis are performed [Kaeslin, 2008].

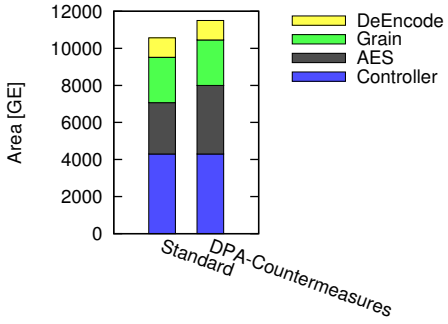
Every student project had $1\,\mathrm{mm}^2$ die area including the pad frame using a $130\,\mathrm{nm}$ CMOS process from UMC. This design uses the low-leakage standard-cell library for all parts of the circuit.

The Synopsys Design Compiler takes an RTL description of a digital circuit and a standard-cell library and generates a gate-level netlist. The designer's task is to set appropriate constraints in order to achieve correct behaviour of the resulting circuit and also optimise area and power consumption. The provided script-based setup allows incremental improvements of the synthesis flow. A main part of the synthesis process is usually to find a compromise between the maximum clock speed and the complexity, thus area usage of the resulting circuit. In case of an RFID controller, the requirement for the maximum clock frequency is way below the possible synthesis propagation delay. Therefore, the only goal during synthesis is to reach minimum area usage and low power consumption. It turns out that any clock-frequency constraint below $55\,\mathrm{MHz}$ does not decrease the complexity of the resulting circuit. Hence, the maximum propagation delay is set to $20\,\mathrm{ns}$ for the whole design. The best results concerning area usage and power consumption are achieved by setting area and power constraints to a minimum, allow optimisation throughout the design, and usage of the compile_ultra command. Listing 5.5 shows the relevant part of the script in the used design flow.

```
1 ...
2 set_max_area 0.0
3 set_max_dynamic_power 0.0
4 set ungroup_keep_original_design
5 compile_ultra -gate_clock -scan -area_high_effort_script
6 ...
```

The rest of the script handles constraints for the clock, reset, and test signals. It also applies clock gating at register level automatically and inserts the scan chain excluding all RAM registers. The optimised setting results in 15% lower area usage and power consumption of the design in comparison to the first synthesis results.

After synthesis and testing of the design, the netlist of standard cells is placed and routed on the available die area, and connected to the pad frame. SoC Encounter automates most of these steps and a



Component	Standard	Perc.	DPA	Perc.
DecodeEncode	1 050 GE	10 %	1 050 GE	9 %
Grain	2 450 GE	23 %	2 450 GE	21 %
AES	2 770 GE	26 %	3 700 GE	32 %
Controller	4 300 GE	41 %	4 300 GE	38%
Sum	10 570 GE	100 %	11 500 GE	100 %

Figure 5.7 & Table 5.1: Area results of the chip and its components.

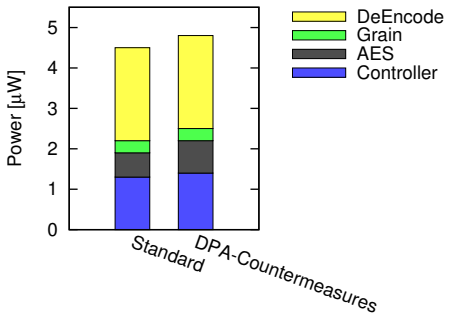
sample back-end design flow is again provided by the Design Center. The die area is fixed for all projects to $1\,\mathrm{mm}^2$ and first trials of place and route of the design showed a very low utilisation of the die area of about $10\,\%$. This allowed to place two different designs. One design uses the straightforward AES core and the second one implements the randomisation techniques explained in Section 5.3.2. Two pad frames, with a reduced pin count of 30 instead of 48, are placed symmetrically on the die. The pins are connected mirror-inverted which allows the usage of the same bonding diagram and results in the same pin layout for both designs by simply rotating the die $180°$ before packaging. The back-end design flow itself turned out to be a straightforward process using the provided script templates. The low power consumption simplifies the power distribution over the chip. A small design size and the low target clock frequency pose fewer obstacles for the clock tree insertion. Before tape-out, several tests are performed to eliminate errors during the back-end design flow. The layout/design rule check (DRC) verifies if all geometrical shapes and combinations of them conform the process specifications. The layout-versus-schematic (LVS) test extracts a netlist from the finalised design and compares its functionality with the tested schematic before the backend process. Furthermore, delay and timing verification eliminates possible setup or hold violations and a power-grid analysis finds critical supply-voltage drops during execution. The datasheet in Appendix A shows an illustration of the die and provides a pinout and port description.

5.6 Results

After the successful implementation, it is time to review the results with attention to the initial constraints and also compare the final design with related implementations.

Figure 5.7 shows an overview of the final complexity of the circuit in GE. The biggest part of the circuit is the main control unit with 4 300 GE. The EPC memory is excluded because a real RFID-tag implementation would use non-volatile memory. The register-based RAM is only used for test purpose in this prototype since other memory options are not available in the used multi-project ASIC process. The encryption-only AES core without DPA countermeasures results in 2 700 GE. The implementation of the randomisation techniques increases the size of AES core by about 1 000 GE. This is the main difference in size of the implementation with DPA countermeasures. Otherwise the two variations only differ in minor changes of the control logic and a different ClockDivide unit. The area/power trade-off implementation of the PRNG using the Grain cipher has a size of 2 400 GE. Finally, the smallest component with 1 050 GE is the DecodeEncode part. Since it is the most active part of the circuit, it holds and uses only the minimal data and performs only decoding and encoding of the bit stream. Furthermore, a few 100 GE are needed to implement the multiplexer and control logic for the AMBA test interface and the ClockDivide unit, omitted in the diagram.

The overall area usage is about 12 000 GE and meets the requirement of keeping the gate count below 15 000 GE. One GE in the used 130 nm process results in about $5\,\mu\mathrm{m}^2$ chip area. Excluding the bonding pads, the design fits on less then $1/10\,\mathrm{mm}^2$ leaving enough space for the analog circuitry, the NVM



Component	Standard	Perc.	DPA	Perc.
DecodeEncode	2.3 μW	51 %	2.3 μW	48 %
Grain	0.3 μW	6 %	0.3 μW	6 %
AES	0.6 μW	14 %	0.8 μW	17 %
Controller	1.3 μW	29 %	1.4 μW	29 %
Sum	4.5 μW	100 %	4.8 μW	100 %

Figure 5.8 & Table 5.2: Average power consumption of the components during one authentication round.

block, and two bonding pads for the antenna connection to keep the overall tag die size below $1/4\,\mathrm{mm}^2$.

In this prototype implementation the two different designs are placed on a fixed area of $1\,\mathrm{mm}^2$, where the major chip area is covered by the bonding pads for the test interface. Appendix A.3 shows a schematic of the final tape-out ready design.

In order to estimate the power consumption the Synopsis power-analysis tool is used. Since the power dissipation is heavily depended on the switching activity of the circuit, the tool uses information from a Modelsim simulation to determine the activity of the circuit. Then it estimates the power consumption on average during one simulation run. The following simulation parameters are used for the power simulation:

Clock period	280 ns
Tari	25 μs
RTcal	58 μs
TRcal	181 μs
Encoding	FM0
Command sequence	One full tag authentication:
	Query — ACK — REQ_RN — TAG_AUTH — REQ_TAG_AUTH
Simulation interval	$\sim$ 28 ms

Figure 5.8 shows the average power consumption of the design during this simulation period, as stated by the Synopsis power-estimation tool. Even though the smallest component of the circuit, the DecodeEncode unit consumes the most power. This is due to the high clock frequency and the high activity during one full authentication round. The main controller uses about 1.3 μW and the cryptographic units are below 1 μW. The additional power consumption of the version with DPA countermeasures is very low because randomisation in time domain only increases the overall time for one authentication round and has little effect on the average power consumption.

One has to consider that averaging the power consumption over a full authentication round including the initial query commands is somewhat misleading. The cryptographic units have a higher power consumption during activity but the communication between reader and tag consumes most of the time. Besides the average power consumption, the digital circuit must not cause spikes in the power consumption which could result a short disconnect and harm the overall performance of the RFID system. A detailed simulation showing the power consumption over time was not available, but it is possible to look at the different components individually and evaluate possible cases of high power consumption. The AES unit at the clock frequency used in the design consumes about 2.2 μW during encryption. The Grain unit generates the random numbers during transmission of data and therefore the power consumption is depended on the uplink data rate. In the worst case it generates 640 kbit/s of random bits per second and consumes up to 1.8 μW. Grain and the AES core never run simultaneously.

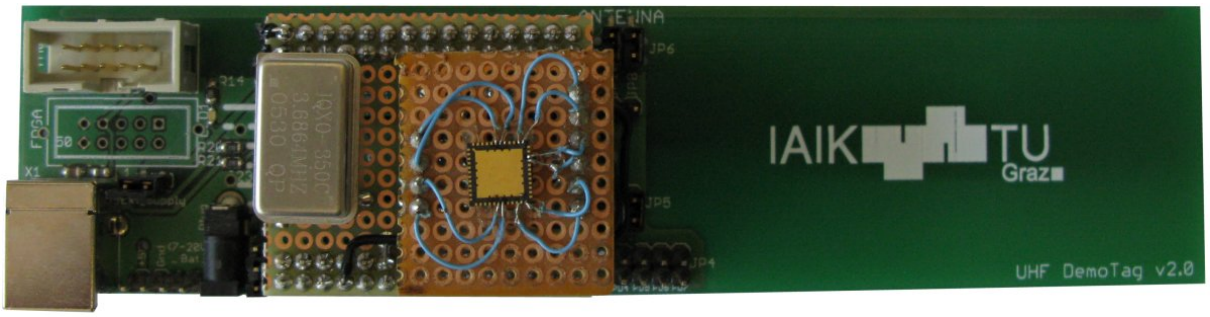


Figure 5.9: Prototype chip mounted to an IAIK UHF DemoTag.

The DecodeEncode unit consumes the most power when encoding uplink data at a high data rate with a high number of subcarrier cycles per symbol. During encoding it uses $3.5\,\mu\mathrm{W}$ with the same data-rate settings, which are at the lower end of the standard specification. Hence, the highest power consumption can be expected during uplink transmission when the tag generates and sends the pseudo-random data. This is the reason for the decision to spend additional $1\,500\,\mathrm{GE}$ on the low-power version of the Grain unit, even though the average power consumption of this unit over the full authentication round seems negligible at first glance.

The $5\,\mu\mathrm{W}$ average power consumption is below the constraint set at the beginning of the project. Still, since the data rates used for the power estimation are on the lower end, operation at higher speed will increase the power consumption and lower the maximum operating range. Additionally, the effort to read and write the EPC flash memory would increase power usage in a real-world tag implementation.

Lowering of the supply voltage in order to decrease the overall power consumption could not be tested in the simulation setup. The operating clock frequency of the circuit is only 10% of the synthesis constraint. This huge margin suggests that it should be possible to use less than the nominal $1.2\,\mathrm{V}$ supply voltage without causing timing violations in the circuitry. Since the power consumption is proportional to $\mathrm{U}_{\mathrm{dd}}^2$, lowering the supply voltage to $1\,\mathrm{V}$ would decrease the power consumption by about 30%. Reducing the voltage to $0.9\,\mathrm{V}$ would even save $\sim 45\%$ of power consumption.

In order to sum up the result section, Figure 5.10 shows a die photograph of the produced prototype chip. The overall die area is $1\,\mathrm{mm}^2$ with both versions (straightforward implementation and with DPA countermeasures) placed on it. In the shown package the version with countermeasures is connected to the IO pins. For testing, the chip is mounted to the analog interface of an IAIK UHF DemoTag as shown in Figure 5.9.

5.7 Comparison with Related Work

As shown above, the design meets the general constraints for passive RFID tags. In the following, these results are compared to related work. It is difficult to directly compare different implementations, since all differ in features, process technology, and operating voltage. Table 5.3 shows an overview of EPC G2 standard implementations with or without additional security enhancements. The area is either listed in GE or in die area without pad frame, depending on the notation in the published paper. Possible analog front-end or a NVM are not included in the area comparison. The power consumption depends heavily on the data rates. Most papers do not specify the conditions for the measurement or simulation and therefore similar parameters as in this work are assumed. Only the supply voltage is decreased significantly in many designs in comparison to the standard process-technology supply voltage. Next to the target process technology, the table lists implemented security enhancements of the EPC C1G2 standard and the underlying cryptographic primitive. The last column provides some brief additional information about the designs.

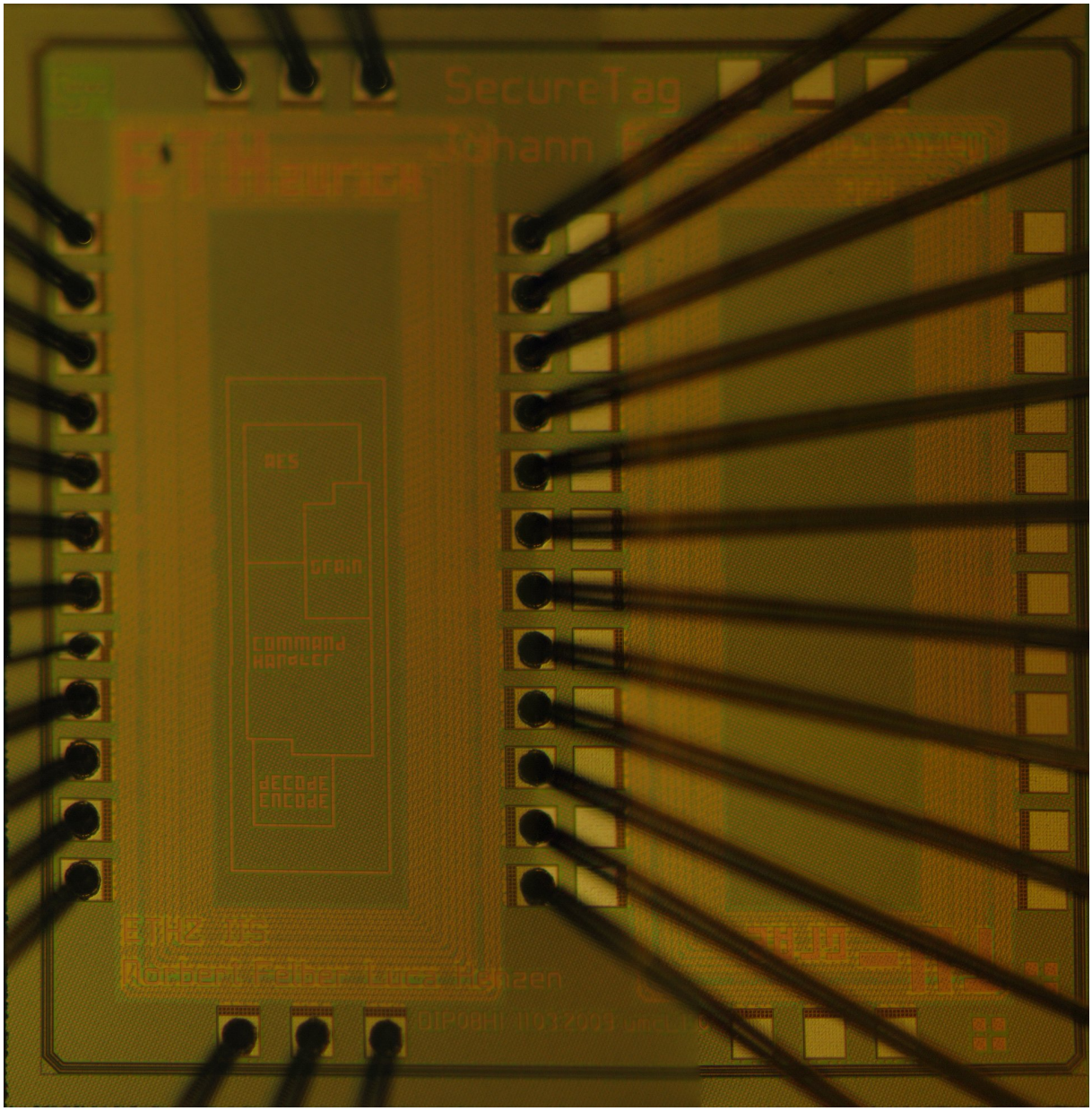


Figure 5.10: Die photograph of the manufactured prototype chip.

	Area	Power@Voltage	Process-techn.	Security enhance.	Comment
Roostaie et al. 2008	0.3 mm^2 (digi. core)	6.4 μW @ 1 V (digital core)	180 nm	none	EPC standard impl. included analog front-end included NVM
Man et al. 2007a	0.23 mm^2	3.44 μW @ 1.8 V	180 nm	none	EPC compliant no NVM/analog front-end
Man et al. 2007b	0.44 mm^2	4.7 μW @ 1.8 V	180 nm	AES	not entirely EPC compliant on-the-fly encryption
Wang et al. 2007	7 800 GE	– (FPGA only)	FPGA	Hash	not entirely EPC compliant custom hash function
Xiao et al. 2011	17 000 GE	30.67 μW @ 1.2 V	130 nm	Humm-ingbird	not entirely EPC compliant included NVM
Zhang et al. 2008	9 450 GE	2.1 μW @ 1 V	180 nm	TEA	not entirely EPC compliant included NVM
Yongzhen et al. 2009	0.25 mm^2	6.9 μW @ 1 V	180 nm	AES	not entirely EPC compliant
Ricci et al. 2008	0.17 mm^2	1.5 μW @ 0.6 V	180 nm	AES	not entirely EPC compliant
Bernardi et al. 2007	1.4 mm^2	1 500 μW @ 1 V	90 nm	CRY	not entirely EPC compliant asymmetric cryptography
This work 2009	12 000 GE	4.7 μW @ 1.2 V	130 nm	AES	EPC compliant secure PRNG

Table 5.3: Comparison of different UHF EPC baseband-processor implementations.

Roostaie et al. present an EPC standard compliant tag implementation including the analog interface and a flash NVM. The area and power results for the digital part of the circuit serve as a reference for implementations with security enhancements. Man et al. present a basic standard-compliant baseband system and also an AES-enhanced version. The area and power results are promising but the paper lacks information about the protocol used in the AES-enhanced version. It suggests that the communication flow between the reader and the tag is AES encrypted on the fly. Hence, the implementation is not compatible with standard compliant infrastructures because of a different inventory sequence and direct encryption would require a very fast encryption to meet the response time constraints of the standard.

Using a custom lightweight hash function implemented with only ~ 400 GE, the design of Wang et al. shows the lowest area usage of all designs but no power results are available. Hummingbird is a symmetric cipher which uses a 256-bit key but only processes 16-bit data blocks. This allows fast encryption and Xiao et al. use this property to meet the strict response-time constraints without using an interleaved protocol. The short encryption time of only 25 clock cycles seems to come at high costs. 17 000 GE and 30 μW is too high for passive UHF tag implementations.

Zhang et al. show a promising trade-off between security and hardware resources. TEA is a lightweight symmetric cipher implemented with only 2 350 GE and shorter encryption time than AES.

There are two further AES-enhanced baseband systems. Ricci et al. present an ultra-low power consumption of 1.5 μW at a non-standard supply voltage of 0.6 V which is only barely above the transistor threshold of 0.5 V. In order to meet the short response times, the tag decrypts random numbers received from a valid reader and stores them in NVM. During communication, sensitive data is then blinded by using these random numbers. Yongzhen et al. use AES in a similar way to achieve the response time constraint. The cryptographic primitive generates and caches one-time session keys before a communication with sensitive data. During communication the tag has to perform only an xor operation before

replying.

Most works with security enhancement try to conceal the EPC number from unauthorised readers already during the inventory round. This aims at easing possible privacy threats as discussed in Section 3. On the other hand, changes in the inventory round cause incompatibilities with the standard specification and significant additional workload in the back-end of large RFID systems.

The only work trying to use asymmetric cryptography in an EPC RFID environment is presented by Bernardi et al. The resulting large chip size in a $90\,\mathrm{nm}$ process and the huge power consumption would require an active power supply of the tag and is only suitable for higher-class tags.

None of the designs specify their PRNG implementation and only state to produce challenges according to the low EPC standard requirements. As already discussed in Section 3.5, poor randomisation of the challenge generation can undermine the security of the authentication protocol. In real-world application this results in a trade-off between additional area usage by using a stream cipher like Grain as PRNG, and possible security vulnerabilities, caused by low-quality challenge generation. A suggestion for true random-number generation is presented by Chen et al. [2009] resulting in $0.05\,\mathrm{mm}^2$ area and $1\,\mu\mathrm{W}$ power consumption generating data at $40\,\mathrm{kbit/s}$ ($180\,\mathrm{nm}, 0.8\,\mathrm{V}$).

Chapter 6

Concluding Remarks and Outlook

The weak security properties of the EPC C1G2 standard have been shown in multiple publications. There are many suggestions for lightweight security enhancements of the standard using only a weak PRNG and basic operations like xor and shift. This requires no additional circuitry on the tag but provides only limited security. It is often difficult to analyse the custom authentication mechanisms and many suggestions have been broken soon after publication. Implementing strong cryptographic algorithms on a passive UHF tag is often considered impossible due to the fierce constraints regarding area and power consumption. This work shows a successful integration of strong symmetric cryptography into passive EPC C1G2 tags. A well-known challenge-response protocol, based on an AES encryption engine, allows mutual authentication between the RFID reader and the tag. In comparison to related work on security enhancements of the EPC standard, it also considers the importance of cryptographically secure pseudo-random numbers. Using a stream cipher like Grain as PRNG prevents security vulnerabilities of the authentication process. A second threat to theoretically secure authentication protocols are side-channel attacks. Implementing randomisation techniques in the time domain improves power-analysis attack resilience with almost no additional circuitry or increase in power consumption.

The presented authentication mechanism is compliant with existing systems. Custom commands provide the additional security features after a standard inventory sequence. The interleaved protocol design allows the usage of a relatively slow but very efficient AES implementation without large performance loss in multi-tag use cases.

The proposed design of a security enhanced EPC G1C2 digital controller has been fabricated using a $130\,\mathrm{nm}$ ASIC standard-cell process. The design including an AES encryption core and a secure PRNG results in $12\,000\,\mathrm{GE}$. Excluding the pad frame, this results in less than $0.1\,\mathrm{mm}^2$ die area for the digital circuitry.

The average power consumption during one authentication round is $4.5\,\mu\mathrm{W}$ at the default $1.2\,\mathrm{V}$ power supply, which makes it suitable for passive RFID applications with several meters in operating range. A big part of the power consumption is due to the decoding and encoding of the communication. This part is the biggest resource for possible further decrease in power consumption. Variable clock frequencies depending on the data rates or improved sequence-generator designs could be an improvement over the implemented solution in this design. Most EPC-tag baseband processor implementations also separate the decoding and encoding units. The implementation of the cryptographic algorithms and the main controller perform very well regarding power consumption.

General conclusions for implementing an EPC-standard baseband processor after the design and implementation of this prototype are: For a low-class passive RFID protocol, the EPC standard is very demanding. The chosen test setup using static command sequences turned out to be quite inflexible to test all possible use cases properly. It was also difficult to evaluate different approaches and designs of the controller part, since the power consumption heavily depends on the communication flow including data rates and encodings. Even though the control logic of the main-controller unit is separated into multiple

state machines, the resulting code turned out to be quite bulky and difficult to modify and maintain. A more flexible approach than fixed-wired FSMs for the tag state and command handling could be worth the additional area usage. Especially since a real-world controller has to handle additional tasks like controlling the semi-persistent flags, the extensive Select command, and accessing the NVM, which is greatly simplified in this prototype work using a register file.

Bringing strong cryptography to low-cost passive EPC tags is a very important step to enable future large-scale RFID systems which are resistant against data manipulation, sabotage, and forgery. Cryptographic algorithms on the tag also allow protocols to address privacy issues. Still, further questions have to be answered for implementing a security-enhanced system. First of all, how secure is the low-power implementation of the AES algorithm against side-channel analysis attacks and are the implemented randomisation techniques sufficient to make an attack on the implementation unpractical? Is the performance of the suggested protocol good enough for real-world applications? The interleaved protocol design reduces the performance drop in tag-reads per second and the power-consumption overhead is comparable low. Still, the EPC has been designed for high-performance inventory and large operating ranges. On system level it is important to consider the compatibility with existing setups and infrastructures. The currently developed standard ISO/IEC 29167-10 will improve the interoperability between systems with different security features and mechanisms. Using symmetric cryptography for authentication mechanisms makes the key handling and distribution more difficult. Especially in large systems with a high number of readers and tags this issue poses a challenging task.

There is a lot of research going on regarding these topics and solving the security and privacy issues will be an important factor for the effectiveness and also acceptance of future wide-spread EPC RFID systems in everyday life.

Appendix A

Datasheet

This datasheet provides usage information for the prototype chip. It includes a brief list of features, explanation of the three operation modes (RFID, DMA, and Test), a memory map, and a pinout diagram including a detailed port description.

A.1 Features

- Secure UHF RFID digital controller ASIC
- Technology: UMC 130 nm LL
- EPC C1G2 standard compliant
- Custom commands for secure authentication based on symmetric cryptography
- Low power/area AES core
- Low power/area Grain implementation
- DPA countermeasures using randomisation
- Test interface for direct memory access (DMA)
- Chip area: 12 000 GE
- Average power consumption during one authentication round: 5 μW

A.2 Usage

The standard usage of the chip is as a digital controller for an EPC-compliant tag. Connecting it to an analog interface, like for example the IAIK UHF DemoTag, allows real-world tests of the implemented security features using a standard UHF reader capable of sending custom commands. Additionally, there is a test interface for direct access to the RAM memories of the design and independent test and execution of the cryptographic units. This section explains the usage and the communication flow in the different operation modes and provides a memory map of the RAM registers.

A.2.1 Operation Modes

There are three operation modes. The standard RFID mode, a DMA test mode, and an initial post-production test mode.

Command	Header Code
QueryRep	00
ACK	01
Query	1000
QueryAdjust	1001
Select	1010
NAK	11000000
ReqRN	11000001
TagAuthentication	11100000 10000001
ReaderAuthentication	11100000 10000010
ReqAuthAnswer	11100000 10000011
EncryptedChallenge	11100000 10000100
InitAES	11100000 10000101
InitGrain	11100000 10000110

Table A.1: Supported EPC standard and custom commands.

RFID Mode

If the ModexSI (pad_Mode_1, pad_Mode_0) is set to '00', then the chip works as a digital controller for an RFID tag using the EPC standard. The input clock has to be 3.5 MHz in order to achieve the compulsory accuracy regarding answer time and up-link data rate. Table A.1 lists the supported commands including their header command codes. More information about the EPC commands and usage can be found in Section 2. The full structure and explanation of the implemented custom commands for authentication are shown in Section 3.3.3.

Direct Memory-Access Mode

Using the 8-bit test interface, it is possible to access the memory blocks of the design and control words for the cryptographic units separately. In any DMA mode the ClockDivide unit is deactivated and all blocks run with the externally applied clock frequency. For all three modes (EPC memory access, Grain, and AES) the basic procedure to read and write data words is the same:

- Apply a 6-bit address to the DataxDI(5 downto 0) pins.
- Toggle the WriteAddrEnablexSI pin one time to store the address into an intermediate register.
- When writing data, set WriteEnablexSI to '1', apply the 8-bit input data to the DataxDI pins and change the EnablexSI signal to '1'. With the next rising clock edge the data is written to the memory.
- When reading data change the EnablexSI signal to '1' and from the next rising clock edge onwards, the DataxDO pins are set to the requested register byte.

Test Mode

The TestEnablexSI signal enables the test mode where all clock-gating cells are disabled and the access to the scan chain is enabled. During test the following pins are reconnected to test signals. The DataxDO[0] signal is connected to the output of the scan chain ScanOutxTO and DataxDI[1] to the start signal of the scan chain ScanInxTI. The ScanEnablexTI signal is wired to the DataxDI[0] pin.

[R]ead/[W]rite	Memory-word description	DMA address
R/W	CRC-16 [7:0]	0x00
R/W	CRC-16 [15:8]	0x01
R/W	PC [7:0]	0x02
R/W	PC [15:8]	0x03
R/W	EPC [7:0]	0x04
⋮	⋮	⋮
R/W	EPC [95:88]	0x0F

Table A.2: EPC memory map (ModexSI = '11').

[R]ead/[W]rite	Memory-word description	DMA address
R/W	Control/Stream Byte	0x00
R/W	Key [7:0]	0x10
⋮	⋮	⋮
R/W	Key [79:70]	0x09
R/W	IV [7:0]	0x20
⋮	⋮	⋮
R/W	IV [63:56]	0x27

Table A.3: Grain memory map (ModexSI = '10').

This mode allows fast post-production tests and is not used during operation afterwards. In practice, the test mode is often disabled before packaging in order to avoid security leaks through the test signals.

A.2.2 Memory Maps and Control Words

Table A.2 to A.4 show the memory maps for the three RAM blocks of the design.

The EPC mode (ModexSI = '11') allows to change the default value of the EPC memory. At startup/reset the EPC memory is initialised as follows:

Addr	x02	x03	x04	x05	x06	x07	x08	x09	x0A	x0B	x0C	x0D	x0E	x0F
Val	x30	x00	x11	x11	x22	x22	x33	x33	x44	x44	x55	x55	x66	x66

The first row shows the address and the second row a valid PC value followed by a random EPC value. The first two registers, containing the CRC-16, are reset to 0x0000 and a valid CRC is calculated every startup (in RFID mode) as described by the EPC standard.

The Grain mode (ModexSI = '10') enables independent usage of the Grain unit with the test interface. After initialisation with the 80-bit key and the 64-bit IV value, it is possible to initialise the stream cipher and read random byte values at address 0x00. Writing the control word 0x00 to address 0x00 starts the initialisation which takes 160 clock cycles. Afterwards, the Grain unit stores 1 byte per 8 clock cycles in a register at address 0x00, which can be read over the AMBA interface.

The AES core has similar behavior but allows more control and status options. After initialisation with the 128-bit key and the 128-bit data, the control word 0x01 starts the encryption. Reading the status byte at address 0x01 provides information about the state of the encryption unit. Status1[7:4] stores the current round number during encryption. Status1[0] indicates a finished encryption. The encrypted data can be read at address 0x10 - 0x1F. The implementation with DPA countermeasures needs additional randomisation data during initialisation. 5 bytes of random data (0x04 - 0x08) control the randomisation during encryption and 4 bytes at address 0x0C - 0x0F are used as state values during the dummy cycles.

[R]ead/[W]rite	Memory-word description	DMA address
W	Control	0x00
R	Status1	0x01
R	Status2	0x02
R	Ram_In	0x03
R/W	Shuff1	0x04
R/W	Shuff2	0x05
R/W	Dummy1	0x06
R/W	Shuff3_Dummy2	0x07
W	Shuff4 (Clk_Gating)	0x08
R/W		0x09
R/W		0x0A
R/W		0x0B
R/W	State_Dummy0	0x0C
R/W	State_Dummy1	0x0D
R/W	State_Dummy2	0x0E
R/W	State_Dummy3	0x0F
R/W	Data [7:0]	0x10
⋮	⋮	⋮
R/W	Data [127:120]	0x1F
R/W	Key [7:0]	0x20
⋮	⋮	⋮
R/W	Key [127:120]	0x2F

Table A.4: AES memory map (ModexSI = '01').

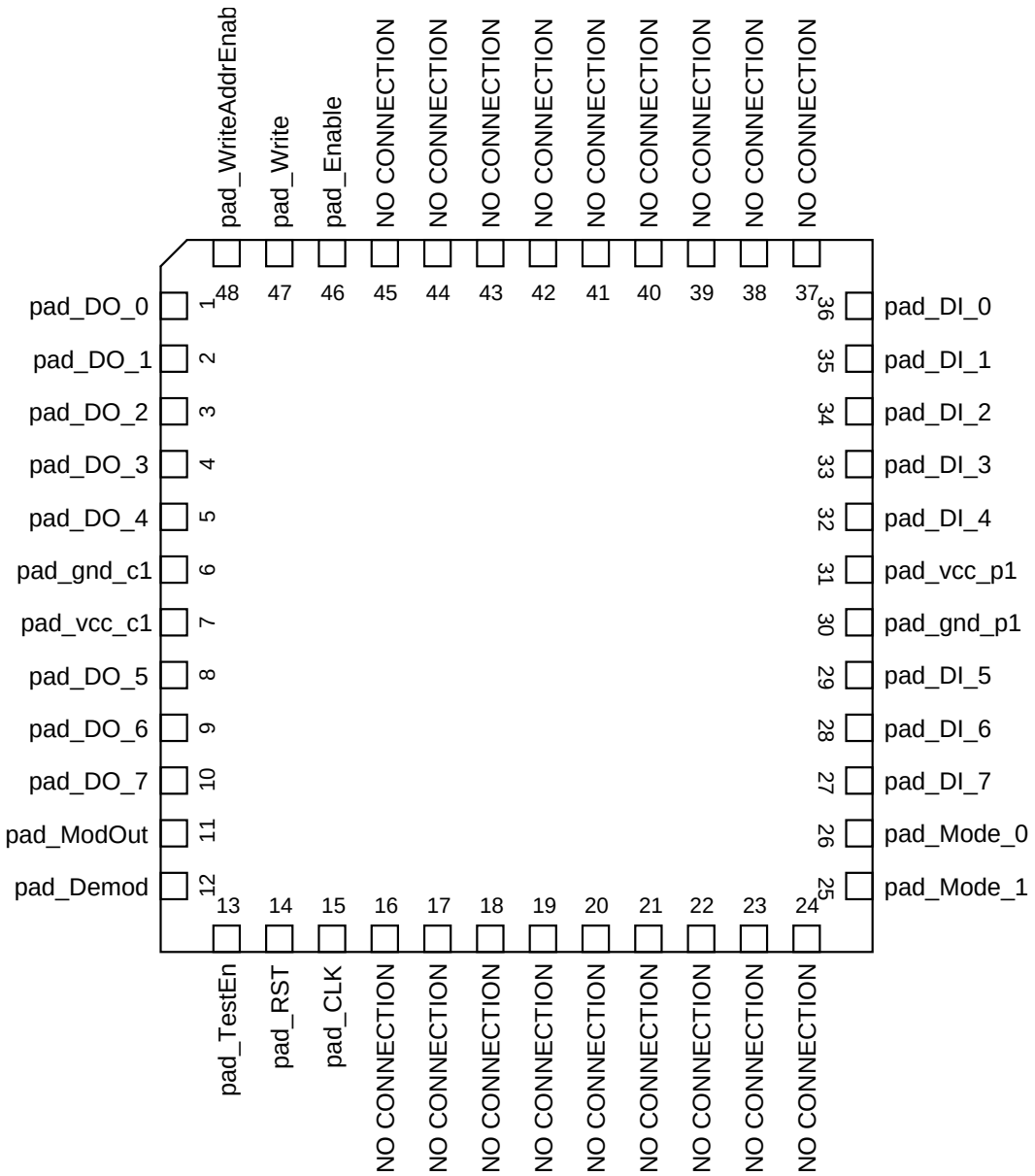


Figure A.1: Pinout.

A.3 Pinout and Port Description

Figure A.1 shows the connections of the packaged chip. Since there are two versions of the implementation with smaller pad frames on one die, only 26 pins besides power supply are connected.

Table A.5 lists all pin names with their corresponding internal signal names using the same notation as in Section 4 and a short functional description.

Pad Name	Signal Name	Type	Description
pad_gnd_c1/p1		Power	Ground Pins.
pad_vcc_c1/p1		Power	VCC Pins.
pad_CLK	ClkxCI	Clock in	Rising-edge sensitive clock signal.
pad_RST	ResetxRBI	Reset in	Asynchronous active-low reset signal.
RFID interface			
pad_Demod	AirDemodxAI	Data in	Demodulated input signal from the analog interface.
pad_Mod	AirModxDO	Data out	Modulated output signal for the analog interface.
AMBA interface			
pad_DO_[7:0]	DataxDO[7:0]	Data out	Data-output signal.
pad_DI_[7:0]	DataxDI[7:0]	Data in	Data-input signal.
pad_WriteAddrEnable	WriteAddrxSI	Signal in	Enables to write an address over DataxDI[5:0] pins.
pad_Write	WritexSI	Signal in	Write data signal.
pad_Enable	EnablexSI	Signal in	Enable signal for read and write access.
pad_Mode[1:0]	ModexSI[1:0]	Signal in	Sets operation Mode. ‘00’ = RFID Mode ‘01’ = AES Mode ‘10’ = Grain Mode ‘11’ = EPC Mode
Test (only if pad_TestEn = ‘1’)			
pad_TestEn	TestEnxTI	Test in	Enables test mode.
pad_DataOut[0]	ScanOutxTO	Test out	Output signal of the scan chain.
pad_DataIn[0]	ScanEnablexTI	Test in	Enables the scan chain.
pad_DataIn[1]	ScanInxTI	Test in	Input signal of the scan chain.

Table A.5: List of pins with functional description.

Figure A.2: Overview of the die, showing the two pad frames with the separate implementation versions.

Appendix B

Acronyms

AES Advanced Encryption Standard

AMBA Advanced Microcontroller Bus Architecture

ASIC Application-Specific Integrated Circuit

ASK Amplitude-Shift Keying

auto-ID Automatic Identification

BLF Backscatter Link Frequency

CMOS Complementary Metal-Oxide-Semiconductor

CRC Cyclic Redundancy Check

DMA Direct Memory Access

DNS Domain Name Service

DOS Denial of Service

DPA Differential Power Analysis

DRC Design Rule Check

DR Divide Ratio

DSB-ASK Double-Sideband Amplitude-Shift Keying

DST Digital Signature Transponder

EAS Electronic Article Surveillance

EBV Extensible Bit Vector

EM Electromagnetic

EOF End of Frame

EPC Electronic Product Code

ETHZ Swiss Federal Institute of Technology Zürich

EU European Union

FPGA Field-Programmable Gate Array

FSM Finite-State Machine

FSR Feedback Shift Register

FT Frequency Tolerance

GE Gate Equivalents

HF High Frequency

IAIK Institute for Applied Information Processing and Communications

IC Integrated Circuit

IDEA International Data Encryption Algorithm

ID Identification

IIS Integrated Systems Laboratory

IP Intellectual Property

ISM Industrial, Scientific and Medical

ITF Interrogator Talks First

LF Low Frequency

LVS Layout versus Schematic

MIT Massachusetts Institute of Technology

MUT Model under Test

NFC Near-Field Communication

NIST National Institute of Standards and Technology

NVM Non-Volatile Memory

PC Protocol Control

PIE Pulse-Interval Encoding

PR-ASK Phase-Reversal Amplitude-Shift Keying

PRF Pseudo-Random Function

PRNG Pseudo-Random Number Generator

PSK Phase-Shift Keying

PW Pulse Width

RAM Random-Access Memory

RFID Radio-Frequency Identification

RF Radio Frequency

ROM Read-only Memory

RTL Register Transfer Level

SSB-ASK Single-Sideband Amplitude-Shift Keying

TEA Tiny Encryption Algorithm

UHF Ultra-High Frequency

URI Uniform Resource Identifier

VHDL Very High Speed Integrated Circuit Hardware Description Language

VLSI Very Large Scale Integration

Appendix C

Symbols

RTcal Interrogator-to-Tag calibration symbol

R⇒T Interrogator-to-Tag

TRcal Tag-to-Interrogator calibration symbol

T⇒R Tag-to-Interrogator

Tari Reference time value for data-0 symbol (Type A Reference Interval)

$\propto$ Proportional

$\oplus$ XOR operator

Bibliography

ARM [1997]. *AMBA Advanced Microcontroller Bus Architecture Specification*. 'Advanced RISC Machines Ltd (ARM)'. (Cited on page 32.)

Bernardi, Paolo, Filippo Gandino, Bartolomeo Montrucchio, Maurizio Rebaudengo, and Erwing Ricardo Sanchez [2007]. *Design of an UHF RFID transponder for secure authentication*. In *GLSVLSI '07: Proceedings of the 17th ACM Great Lakes symposium on VLSI*, pages 387–392. ACM, New York, NY, USA. ISBN 978-1-59593-605-9. doi:http://doi.acm.org/10.1145/1228784.1228876. (Cited on pages 53 and 54.)

Bono, Steve, Matthew Green, Adam Stubblefield, Ari Juels, Avi Rubin, and Michael Szydlo [2005]. *Security Analysis of a Cryptographically-Enabled RFID Device*. In *14th USENIX Security Symposium – USENIX'05*, pages 1–16. USENIX, Baltimore, Maryland, USA. (Cited on page 17.)

Burmester, Mike and Breno de Medeiros [2008]. *The Security of EPC Gen2 Compliant RFID Protocols*. In Bellovin, Steven M., Rosario Gennaro, Angelos D. Keromytis, and Moti Yung (Editors), *Proceedings of the 6th International Conference on Applied Cryptography and Network Security – ACNS 2008*, *Lecture Notes in Computer Science*, volume 5037, pages 490–506. Springer, New York City, New York, USA. (Cited on page 20.)

Chen, Wei, Wenyi Che, Zhongyu Bi, Jing Wang, Na Yan, Xi Tan, Junyu Wang, Hao Min, and Jie Tan [2009]. *A 1.04 uW Truly Random Number Generator for Gen2 RFID tag*. In *Solid-State Circuits Conference, 2009. A-SSCC 2009. IEEE Asian*, pages 117 –120. doi:10.1109/ASSCC.2009.5357193. (Cited on page 54.)

Chien, Hung-Yu [2007]. *SASI: A New Ultralightweight RFID Authentication Protocol Providing Strong Authentication and Strong Integrity*. *IEEE Transactions on Dependable and Secure Computing*, 4(4), pages 337–340. (Cited on page 20.)

Chien, Hung-Yu and Che-Hao Chen [2007]. *Mutual Authentication Protocol for RFID Conforming to EPC Class 1 Generation 2 standards*. *Computer Standars & Interfaces, Elsevier*, 29(2), pages 254–259. (Cited on page 20.)

Cho, Jung-Sik, Sang-Soo Yeo, and Sung Kwon Kim [2011]. *Securing against brute-force attack: A hash-based RFID mutual authentication protocol using a secret value*. *Computer Communications*, 34(3), pages 391 – 397. ISSN 0140-3664. doi:10.1016/j.comcom.2010.02.029. `http://www.sciencedirect.com/science/article/pii/S0140366410001040`. (Cited on page 21.)

Dimitriou, Tassos [2005]. *A Lightweight RFID Protocol to protect against Traceability and Cloning attacks*. In *Conference on Security and Privacy for Emerging Areas in Communication Networks – SecureComm 2005*, pages 59–66. IEEE, IEEE Computer Society, Athens, Greece. (Cited on page 21.)

Dobkin, Daniel [2008]. *The RF in RFID Passiv UHF RFID in Practice*. Elsevier Inc. (Cited on pages 6, 8, 10, 12 and 19.)

EPCglobal, Inc. [2008]. *EPC™ Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communication at 860Mhz-960MHz Version 1.2.0*. (Cited on pages 9 and 19.)

EPCglobal, Inc. [2011]. *EPCglobal Tag Data Standards Version 1.6*. (Cited on page 10.)

Felber, Norbert [2008]. *Desig for Testability*. Lecture notes. (Cited on page 48.)

Feldhofer, M., J. Wolkerstorfer, and V. Rijmen [2005]. *AES implementation on a grain of sand*. *IEE Proceedings Information Security*, 152(1), pages 13–20. ISSN 1747-0722. (Cited on pages 32, 36, 37 and 43.)

Feldhofer, Martin [2007]. *Comparison of Low-Power Implementations of Trivium and Grain*. In *The State of the Art of Stream Ciphers*, pages 236 – 246. (Cited on page 37.)

Feldhofer, Martin, Sandra Dominikus, and Johannes Wolkerstorfer [2004]. *Strong Authentication for RFID Systems using the AES Algorithm*. In Joye, Marc and Jean-Jacques Quisquater (Editors), *Workshop on Cryptographic Hardware and Embedded Systems – CHES 2004*, *Lecture Notes in Computer Science*, volume 3156, pages 357–370. IACR, Springer, Boston, Massachusetts, USA. (Cited on pages 21 and 22.)

Feldhofer, Martin and Thomas Popp [2008]. *Power Analysis Resistant AES Implementation for Passive RFID Tags*. In *Austrochip 2008*, pages 1 – 6. (Cited on pages 44, 45 and 47.)

Feldhofer, Martin and Christian Rechberger [2006]. *A Case Against Currently Used Hash Functions*. In *in RFID Protocols, Workshop on RFID Security (RFIDSEC)*, page 2006. (Cited on page 21.)

Finkenzeller, Klaus [2010]. *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards Radio Frequency Identification and Near-Field Communication*. 2 Edition. John Wiley & Sons, Ltd,. (Cited on pages 3, 4, 5, 6, 7 and 8.)

Garcia, Flavio, Gerhard de Koning Gans, Ruben Muijrers, Peter van Rossum, Roel Verdult, Ronny Schreur, and Bart Jacobs [2008]. *Dismantling MIFARE Classic*. In Jajodia, Sushil and Javier Lopez (Editors), *Computer Security - ESORICS 2008*, *Lecture Notes in Computer Science*, volume 5283, pages 97–114. Springer Berlin / Heidelberg. ISBN 978-3-540-88312-8. http://dx.doi.org/10.1007/978-3-540-88313-5_7. (Cited on page 28.)

Garfinkel, Simson, Ari Juels, and Ravi Pappu [2005]. *RFID Privacy: An Overview of Problems and Proposed Solutions*. *IEEE Security and Privacy*, 3(3), pages 34–43. (Cited on pages 17 and 18.)

Guajardo, Jorge, Pim Tuyls, Neil Bird, Claudine Conrado, Stefan Maubach, Geert-Jan Schrijen, Boris Skoric, Anton M. H. Tombeur, and Peter Thueringer [2009]. *RFID Security: Cryptography and Physics Perspectives*. In Kitsos, Paris and Yan Zhang (Editors), *RFID Security*, pages 103–130. Springer US. ISBN 978-0-387-76481-8. http://dx.doi.org/10.1007/978-0-387-76481-8_5. (Cited on page 31.)

Habibi, Mohammad Hassan and Mahmud Gardeshi [2011]. *Cryptanalysis and improvement on a new RFID mutual authentication protocol compatible with EPC standard*. In *International Conference on Information Security and Cryptology – ICISC 2011*, pages 49–54. Springer, Mashhad, Iran. (Cited on page 20.)

Hell, Martin, Thomas Johansson, and Willi Maier [2006]. *Grain - A Stream Cipher for Constrained Environments*. In *eSTREAM, ECRYPT Stream Cipher Project*. http://www.ecrypt.eu.org/stream/index.html. (Cited on page 28.)

ISO/IEC [2008]. *ISO/IEC 9798-2 Information technology - Security techniques - Entity authentication - Part 2: Mechanisms using symmetric encipherment algorithms*. (Cited on page 22.)

Juels, A. [2006]. *RFID security and privacy: a research survey*. *Selected Areas in Communications, IEEE Journal on*, 24(2), pages 381–394. ISSN 0733-8716. doi:10.1109/JSAC.2005.861395. (Cited on pages 17, 18 and 19.)

Juels, Ari [2005]. *Strengthening EPC Tags Against Cloning*. Manuscript. (Cited on page 20.)

Kaeslin, Hubert [2008]. *Digital Integrated Circuit Design: From VLSI Architectures to CMOS Fabrication*. 1 Edition. Cambridge Univ Pr. ISBN 0521882672. http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20%&path=ASIN/0521882672. (Cited on pages 33, 40, 43 and 48.)

Katti, R.S, Xiaoyu Ruan, and H. Khattri [2006]. *Multiple-output low-power linear feedback shift register design*. 53(7), pages 1487–1495. ISSN 1057-7122. doi:10.1109/TCSI.2006.877889. (Cited on page 37.)

Kocher, Paul, Joshua Ja E, and Benjamin Jun [1999]. *Differential Power Analysis*. In *Advances in Cryptology crypto99*, pages 388–397. Springer-Verlag. (Cited on page 44.)

Liard, M and S Carlaw [2009]. *RFID Annual Market Overview KEy Market Developments, Trends, and Growth Segments*. Technical Report, ABIresearch. (Cited on page 3.)

Man, A.S.W., E.S. Zhang, H.T. Chan, V.K.N. Lau, C.Y. Tsui, and H.C. Luong [2007a]. *Design and Implementation of a Low-power Baseband-system for RFID Tag*. In *Circuits and Systems*, pages 1585–1588. doi:10.1109/ISCAS.2007.378716. (Cited on page 53.)

Man, A.S.W., E.S. Zhang, V.K.N. Lau, C.Y. Tsui, and H.C. Luong [2007b]. *Low Power VLSI Design for a RFID Passive Tag baseband System Enhanced with an AES Cryptography Engine*. In *RFID Eurasia*, pages 1–6. doi:10.1109/RFIDEURASIA.2007.4368097. (Cited on pages 21, 31, 32 and 53.)

Matt, Ward, Rob van Kranenbury, and Gaynor Backhouse [2006]. *RFID: Frequency, standards, adoption and innovation*. *JISC Technology and Standards Watch,*. (Cited on page 6.)

Melia-Segui, Joan, Joaquin Garcia-Alfaro, and Jordi Herrera-Joancomarti [2011]. *A Practical Implementation Attack on Weak Pseudorandom Number Generator Designs for EPC Gen2 Tags*. *Wireless Personal Communications*, 59(1), pages 27–42. (Cited on pages 20 and 28.)

National Institute of Standards and Technology (NIST) [2001]. *FIPS-197: Advanced Encryption Standard, November 2001*. http://csrc.nist.gov/publications/fips/fips197. (Cited on pages 26 and 27.)

Nguyen Duc, Dang, Jaemin Park, Hyunrok Lee, and Kwangjo Kim [2006]. *Enhancing Security of EPCglobal Gen-2 RFID Tag against Traceability and Cloning*. In *Symposium on Cryptography and Information Security*. Hiroshima, Japan. (Cited on page 20.)

Plos, Thomas [2007]. *Implementation of a Security-Enhanced Semi-Passive UHF RFID Tag*. Master's Thesis, IAIK TUGraz. (Cited on page 23.)

Plos, Thomas [2008]. *Susceptibility of UHF RFID Tags to Electromagnetic Analysis*. In Malkin, Tal (Editor), *Topics in Cryptology - CT-RSA 2008*, *Lecture Notes in Computer Science*, volume 4964, pages 288–300. Springer Berlin Heidelberg. ISBN 978-3-540-79262-8. doi:10.1007/978-3-540-79263-5_18. http://dx.doi.org/10.1007/978-3-540-79263-5_18. (Cited on page 44.)

Ricci, Andrea, Matteo Grisanti, Ilaria De Munari, and Paolo Ciampolini [2008]. *Design of an Ultra Low-Power RFID Baseband Processor Featuring an AES Cryptography Engine*. In *Digital System Design Architectures, Methods and Tools, 2008. DSD '08. 11th EUROMICRO Conference on*, pages 831 –838. doi:10.1109/DSD.2008.129. (Cited on pages 21, 43 and 53.)

Roostaie, V., V. Najafi, S. Mohammadi, and A. Fotowat-Ahmady [2008]. *A low power baseband processor for a dual mode UHF EPC Gen 2 RFID tag*. In *Design and Technology of Integrated Systems in Nanoscale Era*, pages 1–5. doi:10.1109/DTIS.2008.4540230. (Cited on pages 51 and 53.)

Sarma, Sanjay [2001]. *Towards the 5cent Tag*. Technical Report, MIT Auto-ID Center. (Cited on page 30.)

Shen, Xiang, Dan Liu, Yuqing Yang, and Junyu Wang [2010]. *A low-cost UHF RFID tag baseband with an IDEA cryptography engine*. In *Internet of Things (IOT), 2010*, pages 1 –5. doi:10.1109/IOT.2010.5678440. (Cited on page 21.)

StopRFID [2005]. *Die StopRFID-Seiten des FoeBuD e.V.* `http://www.foebud.org/rfid`. Accessed: 01.04.2012. (Cited on page 17.)

Suna, Choi and Sangyeon Lee [2011]. *Security Enhanced Authentication Protocol for UHF Passive RFID System. IARIA*, page 307 to 311. (Cited on pages 21 and 25.)

Wang, Jianping, Huiyun Li, and Fengqi Yu [2007]. *Design of Secure and Low-Cost RFID Tag Baseband*. In *Wireless Communications, Networking and Mobile Computing*, pages 2066–2069. doi:10.1109/WICOM.2007.516. (Cited on page 53.)

Weis, Stephen A., Sanjay E. Sarma, Ronald L. Rivest, and Daniel W. Engels [2003]. *Security and privacy aspects of low-cost radio frequency identification systems*. In *International Conference on Security in Pervasive Computing*, pages 201–212. Springer-Verlag. (Cited on page 21.)

Xiao, Mengqin, Xiang Shen, Junyu Wang, and J. Crop [2011]. *Design of a UHF RFID tag baseband with the hummingbird cryptographic engine*. In *ASIC (ASICON), 2011 IEEE 9th International Conference on*, pages 800 –803. ISSN 2162-7541. doi:10.1109/ASICON.2011.6157326. (Cited on page 53.)

Yeh, Kuo-Hui and N.W. Lo [2009]. *Improvement of an EPC Gen2 Compliant RFID Authentication Protocol*. In *Information Assurance and Security, 2009. IAS '09. Fifth International Conference on*, volume 1, pages 532 –535. doi:10.1109/IAS.2009.341. (Cited on page 20.)

Yeh, Tzu-Chang, Yan-Jun Wang, Tsai-Chi Kuo, and Sheng-Shih Wang [2010]. *Securing RFID systems conforming to EPC Class 1 Generation 2 standard. Expert Systems with Applications*, 37(12), pages 7678 – 7683. ISSN 0957-4174. doi:10.1016/j.eswa.2010.04.074. `http://www.sciencedirect.com/science/article/pii/S095741741000374X`. (Cited on page 20.)

Yongzhen, Qi, Wang Xin'an, Feng Xiaoxing, and Gu Weqing [2009]. *Design and implementation of a security-enhanced baseband system for UHF RFID tag*. In *ASIC, 2009. ASICON '09. IEEE 8th International Conference on*, pages 999 –1002. doi:10.1109/ASICON.2009.5351524. (Cited on pages 43 and 53.)

Zhang, Qi, Yunlong Li, and Nanjian Wu [2008]. *A novel low-power digital baseband circuit for UHF RFID tag with sensors*. In *Proc. 9th International Conference on Solid-State and Integrated-Circuit Technology ICSICT 2008*, pages 2128–2131. doi:10.1109/ICSICT.2008.4735016. (Cited on pages 21, 31, 43 and 53.)