# Security and Performance Evaluation of Lightweight Cryptographic Algorithms in RFID

[*] MOJTABA ALIZADEH, [*] MAZLEENA SALLEH, [+] MAZDAK ZAMANI, [+] JAFAR SHAYAN,
[+] SASAN KARAMIZADEH
[*] Faculty of Computer and Information Systems, [+] Advanced Informatics School
Universiti Teknologi Malaysia
[*] 81310 Skudai, [+] 54100 Kuala Lumpur
Malaysia
amojtaba2@live.utm.my, mazleena@utm.my, mazdak@utm.my, sjafar3@live.utm.my,
ksasan2@live.utm.my

*Abstract:* - This paper analyzes the security and performance of lightweight encryption algorithms, which are used in RFID applications. Four of the most popular algorithms which are TEA, HIGHT, KATAN, and KLEIN are implemented on AVR Atmel ATtiny45 microcontroller to evaluate their memory efficiency and energy consumption in performance analysis. In addition, degree of confusion and diffusion are evaluated in security analysis.

*Key-Words:* - Radio-frequency identification (RFID); Cryptography; Network security; performance evaluation; data encryption algorithm

## 1 Introduction

One of the important elements of the electromagnetic spectrum is radio that covers all formats of radiation. The electromagnetic spectrum includes other parts such as gamma rays, cosmic-ray photons, x-rays, and visible light [1].

There are three general bands that are utilized in RFID systems, the first band that it called Low Frequency (LF) at 125 kHz to 134 kHz, the second band is High Frequency (HF) at 13.56 MHz, and the third is Ultra HF at 860 to 930 MHZ [2]. Depending on the different conditions, various types of RFID can be used. RFID can identify objects that contain small tags in different environments without any physical contact. There are three main components in a typical RFID system: readers, back-end servers, and tags [3]. The task of the Reader or Transceiver is providing the needed energy for tag and also trigging the communication signals to the tag to do specific actions [4].

To transmit information between reader and tag, a reader antenna transmits a radio signal, after that these signals can be received by the tag [5].

After receiving the reader's signal, tag answers with a replying radio signal. The signal that transmitted by tag can be read by a reader's receiver. The tag may be performing some encryption functions depending on its computing power. Different kinds of tag are implemented in RFID system; some of them are read-only, and the rests are able to be read or written [6-9].

There are a lot of malicious attacks against RFID systems, which are categorized to active and passive attacks [10]. By improving RFID technology day by day, the threats are changing too [11].

Based on the multitude of employing RFID applications and by considering that RFID tags may contain sensitive private information like biomedical data or health, the importance of security in RFID has risen [12]. To reduce the effect of security and privacy problems, it is desirable to implement different encryption algorithms by considering the nature of RFID tags; these smart devices have enormously constrained resources in terms of computational capabilities, memory, and power supply. These limitations make designing a secure tag more difficult because in order to improve a security, strong encryption algorithm should be implemented but there is no longer enough processing capacity in RFID tags [13].

Modern encryption algorithms that were designed for normal computer is not appropriate for RFID tags because in order to implement these kinds of encryption, sufficient computational capacity, enough memory, and power should be provided [14-16].

## 2 Overviews of Cryptographic Algorithms

In this section, the lightweight encryption algorithms are investigated.

## 2.1 KLEIN

The structure of KLEIN is typical Permutation Network (SPN) same as many ciphers such as PRESENT [17] as AES [18]. The number of round ($N_R$) is 16 rounds for KLEIN 80, which is analyzed in this research [19]. All the operations can be optimized during the round transformation.

It is necessary for all block ciphers to use key schedule to convert master key to series of subkeys. This cipher is constructed based on message authentication and hash function. However, the complexity of key schedule must be proper because of the security issues the key schedule should be very fast.

The Key Schedule algorithm of KLEIN-64 is described in Figure 1. The subkey of KLEIN can be created when each round is transforming to save the memory and increase performance of the algorithm. To resist weak key attacks, the structure of key schedule of KLEIN provides enough complexities because of the Feistel-like structure. This hypothesis was proven by researchers, which were found on the PRESENT block cipher recently [20, 21] An incremental round counter is used to simplify the cipher as described in Figure 1 [19].

## 2.2 KATAN

There are three types of the KATAN ciphers includes KATAN32, KATAN48 and KATAN64. The size of key is 80-bit for all the ciphers in these kinds of algorithms [22].

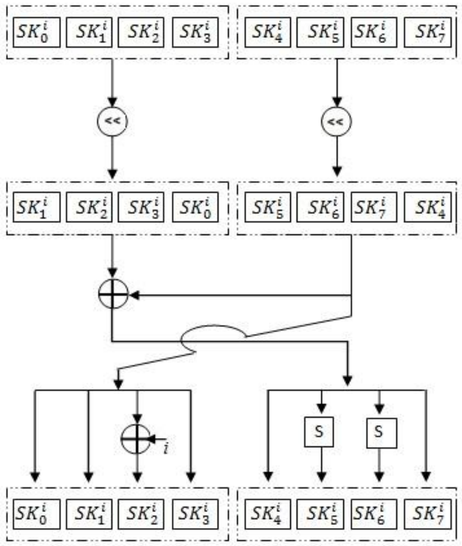

Figure 1. Key Schedule algorithm of KLEIN [19]

In this study, KATAN48 is analyzed. The KATAN32 is the smallest cipher among all ciphers in this family because the size of blocks is 32 bits in this algorithm.

There are some differences in characteristics of the algorithm between the KATAN cipher families. These differences include the size of the blocks of plaintext and ciphertext, the bits positions which the nonlinear functions enter [17].

The structure of KATAN32 is described in Figure 2. In this structure, the counter which calculates the number of rounds is designed. The process is that LFSR as a round counter is started to the all 1's state, and it uses the feedback polynomial $x_8 + x_7 + x_5 + x_3 + 1$ for clocking. The encryption process starts in the next step, and it ends after 254 additional clocks [22].

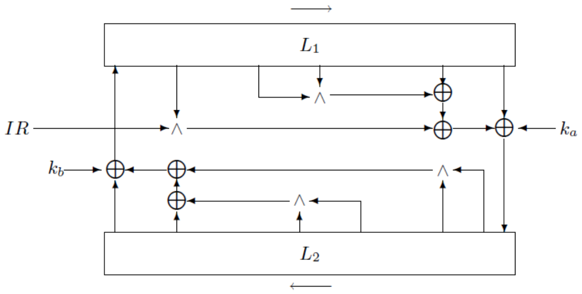

Figure 2. The KATAN/KTANTAN ciphers structure [22]

## 2.3 HIGHT

HIGHT is a block cipher encryption algorithm that is suitable for constraint-resource devices, especially for RFID systems. HIGHT has 128-key length, and it uses 64-bit block length. The numbers of rounds are 32 in this algorithm, and the structure of HIGHT is Feistel network. The left bit wise rotation, addition mod 28, and XOR are three different operations of HIGHT [23].

Encryption process of HIGHT is described in Figure 3, where SK and WK is the subkeys and whitening keys respectively. The plaintext is described by P= $P_7\|\ldots P_1\|P_0$ and the ciphertext is C = $C_7\|\ldots C_1\|C_0$, which contain 8 bytes. The 128-bit master key consists of 16 bytes and is shown by MK = $MK_{15}\|\ldots\|$ $MK_0$ [23].

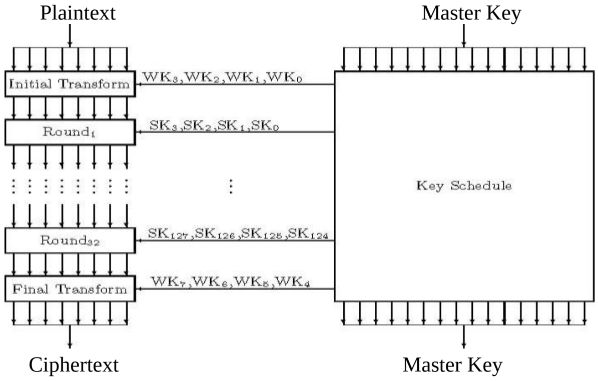

Figure 3. The Encryption process of HIGHT [23]

In this study, only the encryption process of HIGHT is described because the process of encryption is similar to decryption process in this cipher.

### 2.4 TEA

The Tiny Encryption Algorithm is a kind of lightweight encryption algorithm that was developed by Roger Needham and David Wheeler at the Cambridge University. The structure of TEA is a Feistel cipher which uses various operations such as ADD, SHIFT, and XOR. The Shannon's twin properties of diffusion and confusion are provided in the TEA which is necessary for a secure block cipher without the need for P-boxes and S-boxes respectively. The size of data block is 64-bit in this algorithm, and a 128-bit key is used in this algorithm [24].

The routine structure of TEA is shown as in Figure 4. To provide nonlinearity, the routine relies on the alternate use of ADD and XOR. All bits of the key and data can be mixed frequently by a dual shift [25]. Delta is a key schedule constant and *K* is a key and it is derived from the golden number is used where

$$\text{Delta} = \left(\sqrt{5}\ - 1\right) 2^{31} \qquad (1)$$

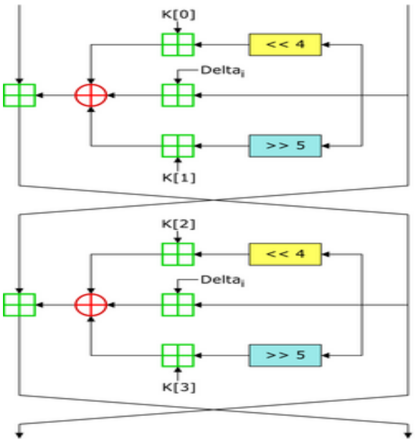

Figure 4. Two Feistel rounds (one cycle) of TEA [25]

## 3 Implementing the Ciphers on AVR Microcontroller

In this section, the process of implementing the ciphers on AVR Atmel microcontroller is described. To analyze the performance based on memory efficiency and energy consumption, the ciphers should be implemented. The security metrics were the degree of diffusion and confusion. The Assembly language is selected among different programming languages such as C or Java. One of the most important characteristics of the assembly language is shortness and easiness. The instructions of assembly language are translated one by one to executed machine instructions.

Assembly language is chosen due to the fact that there is no need to have extra loops and unnecessary features, and it is helpful to use this language in RFID systems because there are no longer enough resources in RFID tags. Another reason is that with this language programs are shorter, easier to be built and debug [26]. The AVR Studio 5.1 is used as the assembler to implement ciphers on AVR ATtiny45 Microcontroller.

## 4 Performance Evaluation

The results of implementing the ciphers on AVR microcontroller are described. There are two different kinds of performance analysis in this study, memory efficiency, and energy consumption.

### 4.1 Memory efficiency

The performance of focused lightweight algorithms is analyzed in this part. The memory usage of different lightweight algorithms is compared in Figure 5.

The figure shows the percentage of memory using for each cipher. The size of SRAM and In-System Programmable Flash for Atmel ATtiny45 micro controller is 256 and 4k bytes respectively. The percentages are calculated based on the size of this SRAM and In-System Programmable Flash of this microcontroller.

According to the results of this study, the KLEIN uses longer Flash memory space than the rest because the size of assembly code of this algorithm is more than others; however, the percentage of SRAM usage for this cipher in not as much as other ciphers. The KATAN uses a smaller amount of the SRAM memory in comparison to other ciphers, and also the percentage of the FLASH memory usage for this cipher is less than other three ciphers.

According to this figure, KATAN is the more appropriate cipher in case of using memory.

The Data Memory Usage for TEA and HIGHT algorithm is equivalent. The reason of this similarity is that the size of key and the size of a block cipher are the same for both algorithms. The size of blocks for these algorithms is 64-bit, and the size of key is 128-bit. The same result can be achieved for another two algorithms, KATAN and KLEIN. These two algorithms are using 64-bit blocks and 80-bit key. The key and block size for focused algorithms are different, given this KATAN and KLEIN use fewer amounts of Data Memory that other two ciphers.

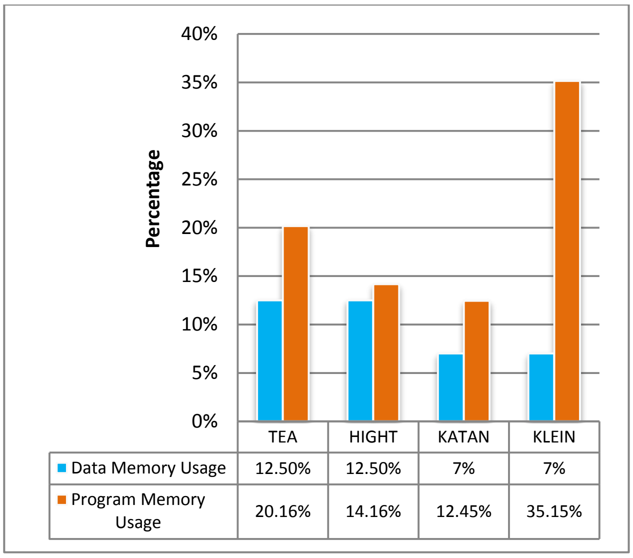

Figure 5. The comparison of Data and Memory usage of ciphers

The Program Memory Usage depends on the size of Assembly code for each algorithm. As it is explained in the previous parts, the size of code for KLEIN cipher is more than the rest ciphers. This size of programming code causes that KLEIN algorithm uses more memory in comparison to other algorithms. The KATAN uses less memory in comparison to others because of the size of codes for this cipher.

### 4.2 Energy Consumption

To measure energy consumption, it is assumed that the energy per CPU cycle is fixed and can be calculated as follows [27].

$$E = I \times VCC \times \tau \times N \qquad (2)$$

Where, I is the average current in amperes which is consumed for T seconds and VCC is the supply voltage of the system. τ is the clock period and N is the number of clock cycle.

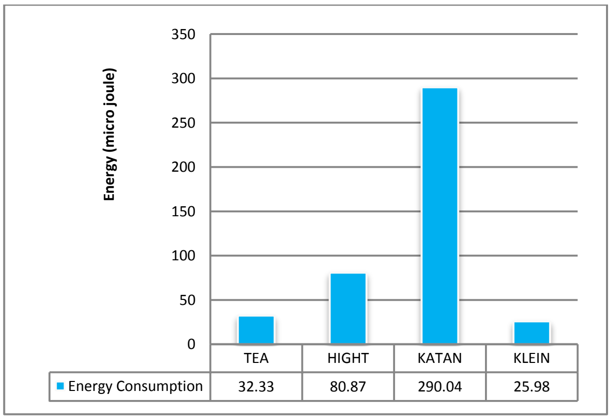

Figure 6. Energy consumption comparison of focused ciphers

Figure 6 shows the power consumption of ciphers, which are analyzed in this study.

Energy consumption calculated in micro joule ($10^{-6}$J) in this survey. The energy consumption includes the key scheduling and encryption. Low computational complexity is so important for battery-powered devices such as RFID tags because processing time effects on power consumption directly.

According to Figure 6, KATAN consumes more energy than the rest and the difference between power consumption of KATAN and others is obviously big. KLEIN is the best algorithm in comparison to other ciphers in aspect of energy consumption. However, it has the largest size of codes.

In the previous parts of this study, the size of programming code for each cipher was analyzed. According to results of code analysis, KLEIN algorithm has the biggest size of code among investigated ciphers, but it consumes less energy than other ciphers by considering the results of energy consumption analysis.

By Analyzing the energy consumption of these algorithms, it can be concluded that the number of CPU clock cycles is more influential that the size of codes. In other words, the performance of code depends on several specifications of a cipher such as the kind of instructions, kind of operation, kind of structure, number of loops, and number of rounds, which are more significant than size of code. For example, the numbers of CPU clock cycle for an algorithm are different when we execute it with a different number of rounds.

Another effective metric is a kind of instruction. The number of clock cycle for each instruction is different and to reduce the power consumption of an algorithm, it is effective to use suitable instruction to write a code.

Kind of operation used in an algorithm is also important because some operations need a lot of clock cycles for execution. The structure of an algorithm is another effective metrics and energy consumption for Feistel structure is different from SPN network. The number of repeated loops can effect on the power consumption directly.

## 4 Security Evaluation

There are two kinds of security test, including diffusion and confusion test to benchmark security. The results of these tests are described in this part.

### 4.1 Degree of Diffusion

One of the most important metric that are used to benchmark the security of an algorithm is the diffusion. To get better results, a random value of plaintext is created and from this plaintext, several other derived plaintexts. In the first step, one bit of the previous 20 plaintexts was changed, and same as earlier steps encrypted. The results of both steps were XORed to make a 20 x 64 matrix. To benchmark diffusion, the numbers of 1's were added then the percentage of these 1's was calculated. The results of this analysis are described in Table 1.

| | TEA | HIGHT | KATAN | KLEIN |
|---|---|---|---|---|
| Degree of diffusion | 51.1 % | 49.7 % | 51 % | 48.6 % |

TABLE 1. THE ANALYSIS OF DIFFUSION TEST

### 4.1 Degree of Confusion

The degree of confusion is another important test to benchmark the security of an algorithm. The size of plain text was 64 bits. The effect of changing a key was tested in this step. To get results, a random value of key is created and from this key, several other derived keys. In the next step, with one bit difference in the key; all the previous plaintexts were encrypted again, and the results were XORed together. The number of 1s added then the percentage of these 1's was calculated. The results of this analysis are described in Table 2.

| | TEA | HIGHT | KATAN | KLEIN |
|---|---|---|---|---|
| Degree of Confusion | 49.14% | 49.21% | 48.90% | 50.31% |

TABLE 2. THE ANALYSIS OF CONFUSION TEST

As the result of security benchmarking of focused algorithm, the structure of cipher is an effective parameter to determine the security level of an algorithm. Among these ciphers, KLEIN has the lowest degree of diffusion and the highest degree of confusion. It can be concluded that these diffusion and confusion degrees of KLEIN is related to the structure of this algorithm because unlike other ciphers in this study that their structure is Feistel, the structure of KLEIN is SPN.

## 5 Conclusion

This paper analyzes the performance of lightweight encryption algorithms, which are used in RFID applications. Four of the most popular algorithms which are TEA, HIGHT, KATAN, and KLEIN are implemented on AVR Atmel ATtiny45 microcontroller to evaluate their memory efficiency and energy consumption in performance analysis part and also degree of confusion and diffusion in security analysis part. The performance criteria were memory efficiency and energy consumption. KATAN is the more appropriate cipher in case of using memory, but it consumes more energy than the rest and the difference between power consumption of KATAN and others is obviously big. KLEIN is the best algorithm in comparison to other ciphers in aspect of energy consumption. The security criteria were degree of confusion and diffusion. KLEIN has the lowest degree of diffusion and the highest degree of confusion. It can be concluded that these diffusion and confusion degrees of KLEIN is related to the structure of this algorithm because unlike other ciphers in this study that their structure is Feistel, the structure of KLEIN is SPN.

*Acknowledgment*

This work is a part of a research that has been done under support from Universiti Teknologi Malaysia.

*References:*

[1] P. Sanghera and F. Thornton, *How to Cheat at Deploying and Securing RFID*: Syngress, 2007.

[2] F. Thornton, B. Haines, A. M. Das, H. Bhargava, A. Campbell, and J. Kleinschmidt, *RFID Security*: Syngress, 2005.

[3] I. Erguler and E. Anarim, "Security flaws in a recent RFID delegation protocol," pp. 1-13, 2011.

[4] P. Kitsos and Y. Zhang, *RFID security: techniques, protocols and system-on-chip design*: Springer, 2008.

[5] S. Nikbakhsh, A. B. A. Manaf, M. Zamani, and M. Janbeglou, "A Novel Approach for Rogue Access Point Detection on the Client-Side," in *Advanced Information Networking and Applications Workshops (WAINA), 2012 26th International Conference on*, 2012, pp. 684-687.

[6] P. H. Cole, *Networked RFID systems and lightweight cryptography : raising barriers to product counterfeiting*. Berlin [u.a.: Springer, 2008.

[7] M. Zamani, A. Bt Abdul Manaf, S. M. Abdullah, and S. Shojae Chaeikar, "Correlation between PSNR and Bit per Sample Rate in Audio Steganography 2

PSNR of Different Bit per Sample," 2012, pp. 163-168.

[8] M. Zamani, A. Bt Abdul Manaf, and S. M.Abdullah, "Efficient Embedding for Audio Steganography 2 Listening Test to Determine the," pp. 195-199, 2012.

[9] M. Zamani, A. Bt Abdul Manaf , and S. M. Abdullah, "Correlation between PSNR and Size Ratio in Audio Steganography," 2012, pp. 82-87.

[10] M. Janbeglou, M. Zamani, and S. Ibrahim, "Redirecting network traffic toward a fake DNS server on a LAN," in *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*, 2010, pp. 429-433.

[11] A. Mitrokotsa, M. Rieback, and A. Tanenbaum, "Classifying RFID attacks and defenses," *Information Systems Frontiers,* vol. 12, pp. 491-505, 2010.

[12] M. Gharooni, M. Zamani, M. Mansourizadeh, and S. Abdullah, "A confidential RFID model to prevent unauthorized access," in *Application of Information and Communication Technologies (AICT), 2011 5th International Conference on*, 2011, pp. 1-5.

[13] D. Engels, X. Fan, G. Gong, H. Hu, and E. M. Smith, "Hummingbird: ultra-lightweight cryptography for resource-constrained devices," presented at the Proceedings of the 14th international conference on Financial cryptograpy and data security, Tenerife, Canary Islands, Spain, 2010.

[14] A. Poschmann, M. Robshaw, F. Vater, and C. Paar, "Lightweight Cryptography and RFID: Tackling the Hidden Overheads Information, Security and Cryptology – ICISC 2009." vol. 5984, D. Lee and S. Hong, Eds., ed: Springer Berlin / Heidelberg, 2010, pp. 129-145.

[15] S. Shojae Chaeikar, S. Abd Razak, S. Honarbakhsh, H. Rouhani Zeidanloo, M. Zamani, and F. Jaryani, "Interpretative Key Management (IKM), A Novel Framework," *2010 Second International Conference on Computer Research and Development,* pp. 265-269, 2010.

[16] S. Shojae Chaeikar, A. Bt Abdul Manaf, and M. Zamani, "Comparative Analysis of Master-Key and Interpretative Key Management ( IKM ) Frameworks," 2009.

[17] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. J. B. Robshaw, Y. Seurin, and C. Vikkelsoe, "PRESENT: An ultra-lightweight block cipher," in *9th International Workshop on Cryptographic Hardware and Embedded Systems, CHES 2007, September 10, 2007 - September 13, 2007*, Vienna, Austria, 2007, pp. 450-466.

[18] M. Feldhofer, J. Wolkerstorfer, and V. Rijmen, "AES implementation on a grain of sand," *Information Security, IEE Proceedings,* vol. 152, pp. 13-20, 2005.

[19] Z. Gong, S. Nikova, and Y. W. Law, "KLEIN: A new family of lightweight block ciphers," in *7th International Workshop on RFID Security and Privacy, RFIDSec 2011, June 26, 2011 - June 28, 2011*, Amherst, MA, United states, 2012, pp. 1-18.

[20] J. Cho, "Linear Cryptanalysis of Reduced-Round PRESENT ". vol. 5985, J. Pieprzyk, Ed., ed: Springer Berlin / Heidelberg, 2010, pp. 302-317.

[21] K. Ohkuma, "Weak Keys of Reduced-Round PRESENT for Linear Cryptanalysis Selected Areas in Cryptography." vol. 5867, M. Jacobson, *et al.*, Eds., ed: Springer Berlin / Heidelberg, 2009, pp. 249-265.

[22] C. De Cannière, O. Dunkelman, and M. Knežević, "KATAN and KTANTAN — A Family of Small and Efficient Hardware-Oriented Block Ciphers Cryptographic Hardware and Embedded Systems - CHES 2009." vol. 5747, C. Clavier and K. Gaj, Eds., ed: Springer Berlin / Heidelberg, 2009, pp. 272-288.

[23] D. Hong, J. Sung, S. Hong, J. Lim, S. Lee, B.-S. Koo, C. Lee, D. Chang, J. Lee, K. Jeong, H. Kim, J. Kim, and S. Chee, "HIGHT: A new block cipher suitable for low-resource device," in *8th International Workshop on Cryptographic Hardware and Embedded Systems, CHES 2006, October 10, 2006 - October 13, 2006*, Yokohama, Japan, 2006, pp. 46-59.

[24] S. J. Shepherd, "The Tiny Encryption Algorithm," *Cryptologia,* vol. 31, pp. 233-245, 2007.

[25] D. Williams, "The Tiny Encryption Algorithm ( TEA )," *Network Security,* pp. 1-14, 2008.

[26] S. Prüfungsarbeit, "Performance Analysis of Contemporary Light-weight Cryptographic Algorithms on a Smart Card Microcontroller Erklärung," 2007.

[27] D. Salama, H. A. Kader, and M. Hadhoud, "Studying the Effects of Most Common Encryption Algorithms," vol. 2, pp. 1-10, 2011.