ISSN 2347 - 3983
Volume 7, No. 11 November 2019
International Journal of Emerging Trends in Engineering Research
Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter157112019.pdf
https://doi.org/10.30534/ijeter/2019/157112019

FPGA Implementation of Katan Block Cipher for Security in Wireless Sensor Networks

Radhika Rani Chintala[1], Lakshmi Sri Ram Janjanam[2], Sai Kousik G[3], Sai Pawan S[4]
[1] Faculty, Department of Computer Science Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India
[234] Student, Department of Computer Science Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India
radhikarani_cse@kluniversity.in; lakshmisriram8999@gmail.com

ABSTRACT

In this paper we are going to design an efficient and enhanced low weight cipher algorithm for multipurpose applications. A number of new factors like limited computational power, RAM size, ROM size, register width, different operating environment and etc., constrained WSN (Wireless Sensor Networks) to use traditional security measures. These constraints on Wireless sensor networks enabled devices results in the emergence of a new field, Lightweight Cryptography. Recently a number of software and hardware implementation of lightweight ciphers are designed for Wireless sensor networks applications. To attain the declared objective, we choose a single cipher only. Which is specifically the KATAN light - weight block cipher algorithm and it needed to be modeled, implement and optimize on specific XILINX ISE platform.

Key words: KATAN, Cipher, Cryptography, Light-weight, Wireless Sensor Networks (WSN), Field-programmable gate array (FPGA), encryption.

1. INTRODUCTION

Now-a-day's information technologies are widely reached into people's daily life's activity, and it became main trend in present society. An average human life cannot be imagined without various type of gadgets with them. A lot of households use devices with an embedded OS (operating system) (besides usual personal computers), which can be connected to the Internet and can it be even be united into a wireless network.
Everywhere people are surrounded by the variety of a terminals, readers, sensors etc. Such a expansion of the smart technologies crucially raising the data security problems. However, now this is impossible to suggest a crypto-graphic primitive that can be implements in all types of target devices. We can tell that AES [1] is a really strong algorithms with good performances. It is absolutely advisable to be use AES in the high-end device, in a large variety of the embedded systems (or) in some low-end devices (with several constraints). But there is a problem because it's not possible to use a common type of cryptographic algorithm in a specific devices with extreme constrained of resources.
Examples for such type of devices include:

1.1. RFIDs.
1.2. Low-end smart cards (including wireless).
1.3. Wireless sensors etc.

The underlying principles and approaches to the design of algorithm intend for usage in devices with extremely lesser resources are the slightly different from the designing criteria of a commonly used crypto - graphic algorithms. This specific field is covered by a branch of modern cryptography, those are lightweight cryptography.
Different-Different criteria is presented in light-weight crypto-graphy, so there are some similar options for light-weight algorithms that are limited requirements of resources for expected devices, in which these are includes the following:

1.4. RAM (Random-Access-Memory).
1.5. Computing capacity of microprocessor/controller.
1.6. ROM (Read-Only Memory) etc.

In this paper we proposed a review of a set of light-weight block ciphers. Also, we have tried to analyse and generalize the main approaches to the design of light-weight algorithms (KATAN), more over main factors like power, area, time. Also, the constraints of their uses and the trends of light-weight cryptography.

2. BACKGROUND WORK

In the past two decades, Wireless Sensor Networks (WSN) has emerged as a distinguishable approach and it is able to

drag the attention of so many researchers. Security and Privacy is the mostly concerned thing in wireless sensor networks. Light-weight cryptography is the main aspect of security which has ran to many results being published in the research literature. The following section presents some research in the field related to WSN. Paulo S.L.M. Barreto and Vincent Rijmen (2000) presented a lightweight SPN block cipher KHAZAD that favours component reuse following WTS [2] (Wide Trail Strategy). In wide trail strategy round the transformations have 2 multiplicative inverse steps; first, local non-linear transformation (any number of outputted bits are depended upon a limited number of inputted bits) and second, a linear mixed transform for achieving high rate of diffuse. KHAZAD have a 64-bits long block and its key size is 128-bits. Its S-box is randomly generated, and decryption differs from encryption only in the key schedule [3]. Kazumaro Aoki et al. ({2000}) presented Camelia, is the one-twenty-eight (128) bit of Feistel block cipher having 128/192/256-bit keys with 18/24/24 round for Camelia-128/192/256 respectively. It uses four different 8X8 S-box in the non-linear layer, designed to minimize hardware size with additional input/output key whitening. The F-function uses SPN structure and is inserted after every 6 rounds [4]. Phillip Rogaway et al. (2001) described a parallelizable block cipher mode of operation, named as OCB (Offset-Codeback-Mode) for approved Encryption that provides privacy and authentication. Offset codeback mode is an advancement over IAPM that performs encryption on arbitrary length bit string $M \in \{zero, one\} *$ using mod M /n + 2 block cipher mentions, whereas 'n' is the cipher-block length. In it offset calculation and session setups are economical and a single cryptographic key is proposed. There is no extended-precision addition and proposed block cipher calls are most favourable [1]. William C. Barker and Elaine Barker (2004) specified Triple Data Encryption Algorithm (TDEA) Feistel block-cipher by implementing DEA cryptographic engine. It has a block size and key there both sizes is 64-bit (56 bits randomly generates by the algorithm as key-bits and remaining 8-bits used for error detection). Operations performed in DEA engine are initial permutation, complex key dependent computation, and inverse initial permutation. DEA engine runs in two directions – they are forward direction and reverse direction. It is in the serial order in which the key-bits are used that varies in these two directions. In TDEA forward and inverse operation is defined as a compound operation of DEA forward and inverse transformations. TDEA key consists of a key bundle, KEY with three keys. To be secure the number of blocks processed with on bundle key should be less than 232 [5]. Katsuyuki Takashima (2005) designed mCrypton (miniature of Crypton), a lightweight SPN enhancement over Crypton. It is having a 64-bit cipher-block size, 64 or 96 or 128-bits key-size and number of rounds are 12. Round transformation has four steps: substitution (pick over using 4-bit of S-boxes), bit several possible ways (Column-wise), transposition (Column-to-Row), and key addition. mCrypton processes 8-byte data block into 4X4 nibble array representation as in Crypton. Key scheduling consists two stages; the first stage is a key generation for round using S-boxes and the second stage is key variable update through round constant and rotations (word-wise and bit-wise). S-boxes used in key scheduling are same as that in round transformation. Decryption differs by encryption with a different key schedule [6]. There exists a MITM attack on mCrypton [7]. Francois-Xavier et al. (2006) designed a Feistel lightweight block cipher, SEA for software implementations on an 8-bit processor. SEA n, b operates on a number of blocks and key sizes. Operations performed in SEA are: bitwise Exclusive OR; S-box in parallel; word rotation and inverse word rotation; bit rotation and; addition mod. In key scheduling, during half rounds, the master key is encrypted while during the other half rounds master key is decrypted [8].

3. CRYPTOGRAPHY

Cryptography is a method of protecting data and communications through the using of codes so that only those for whom the data is intended can read and process it. The pre-fix "crypt" means "hidden" or "vault" and the suffix "graphy" stands for "writing."

In terms of computer science, crypto-graphy means to secure the data from attackers and communicating techniques obtains based upon the mathematical concept and it is a set of rule-base on calculations called algorithms to transfer messages in a way that are hard to decipher for attackers.

These algorithms are used to generate a crypto-graphic key and signatures of users in digital way (digital signature) and protection of data privacy through verification process, browsing the useful data in internet, secured communications and credit/master cards billings and sending or receiving emails.

4. KATAN ALGORITHM

KATAN-family of encryption algorithms KATAN, in KATAN we are having three types of encryption techniques -1. KATAN 32, 2. KATAN 48, 3. KATAN 64. The number in the name of algorithm designates the size of the block of ciphered data in bits. All algorithms use 80-bit encryption key. In this we are using KATAN64. Any of algorithms of KATAN loads the ciphered text block of data into the two shift registers, forming internal state of algorithm (i.e. the certain intermediate values depending on the block of data and the key of en-ciphering; it is also processed by algorithm in the course of en-ciphering). Encipher is made up of 254 rounds in each of which the nonlinear function used by forming feedback registers. A key feature of this algorithm is that it can be implement using 802 GE in case KATAN32, which allow its usage in systems with some limited computing resources. The KATAN algorithm, is in a fact that it is a family of 6 hardware's-based light-weight block ciphers. Each algorithms knowingly need to share the some

basics of same type of encipher algorithm, but differentiation is there at key schedule process and block size. For example, the KATAN block cipher supports the 32bit, 48bit and 64bit block size. Still, while the KATAN 32, KATAN 48, KATAN 64 ciphers all are having same size of 80-bit key in the both encryption and decryption processes.

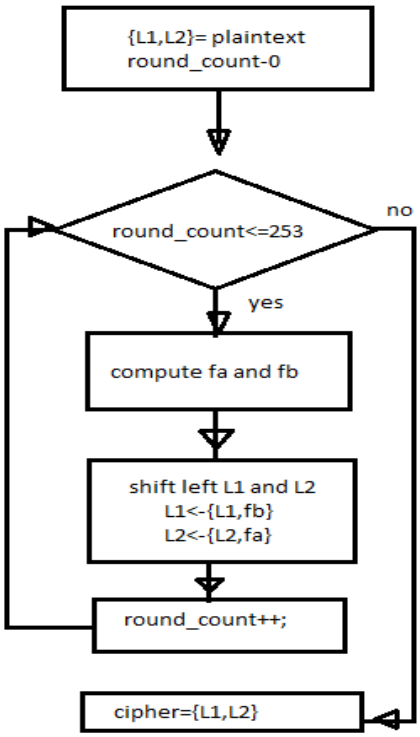

Figure 1: KATAN flow chart

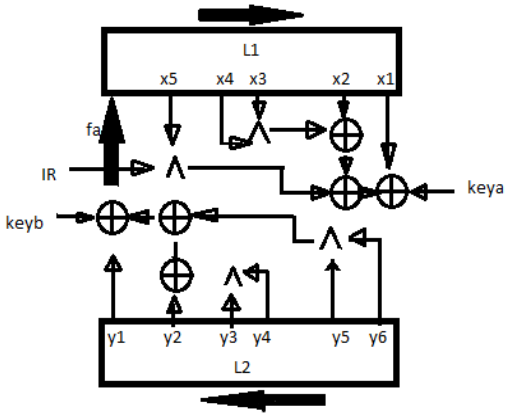

Figure 2: KATAN cipher round functions diagram

The KATAN cipher algorithm, shown in Figure1, is having a two fifty four rounds of execution process. Initially, the plain-text (just a normal text without any encryption) loads into two individual registers those are the L_1 register and L_2 register. The key (eighty-bit of master key) also given as a inputs to the KATAN cipher.

Thereafter, in each and every round, few bits from register L_1 and register L_2 are processed by two functions those functions are f_a function and f_b function, and then loaded to the minimal important bits of the register L_1 and register L_2 later we have to do left shifting, as elaborated in Figure2.

The functions are computed as shown in Equation. (1) below:

$$f_a(L_1) = L_1[x_1] \text{ XOR } L_1[x_2] \text{ XOR } (L_1[x_3] \wedge L_1[x4]) \text{ XOR } (L_1[x_5] \wedge IR) \text{ XOR } key_a \quad (1)$$

$$f_b(L_2) = L_2[y_1] \text{ XOR } L_2[y_2] \text{ XOR } (L_2[y_3] \wedge L_2[y_4]) \text{ XOR } (L_2[y_5] \wedge L_2[y_6]) \text{ XOR } key_b$$

4.1. IR is an irregular update which is calculated through LFSR. In a particular order that the irregular update is the output of the mostly-significant bits to a Linear feedback shift register (LFSR), this is what implements the polynomial equation as: $x_8 + x_7 + x_5 + x_3 + 1$.

4.2. key_a and key_b are 2 sub-key bits. For i^{th} round, key_a is key_{2i} and key_b is key_{2i+1}.

4.3. key_j is the j^{th} bit of key which is generates as:

$$Key_j = \begin{cases} Key_j, & \text{for } j=0\ldots79 \\ Key_{j-80} \text{ XOR } Key_{j-50} \text{ XOR } Key_{j-13}, & \text{Otherwise.} \end{cases}$$

The two functions fa function and f_b function are applied for one time, two times or the third time for KATAN (32, 48, 64 -bit) respectively.

5. LFSR (LINEAR-FEEDBACK-SHIFT- REGISTER)

In processing, a LFSR (linear-feedback-shift-register) is a shift register for that register the previous state output of a linear function is inputted to it.

A shift register is the sequential series of the bit cell, each of which is a flip-flop:

- It has a input signal.
- It will store a single bit of state, either a0 (or) a1.
- It change that state bit to the input signal, upon receiving the clock signal.
- It has an output signal that is the state of the shift register — if the shift register contains a 1, then it outputs a 1; if it contains a 0, then it outputs a 0.

The output of one cell is connect to the input of the next cell, so the bits in the shift register pass on from left side direction to right side direction, (or) from the right side direction to left side direction, depend on how shift register is made up.

The bits shift from right to left, and the cells are numbered from 0 to N−1N−1 (here we have a 5-bit shift-register, so N=5N=5). The state bits, collectively denoted SS, are individually denoted S_0S_0 to SN−1SN−1 from right to left.
If we just have a shift register, then it's not very interesting:
$S_0[k]=u[k-1]$ $S_0[k]=u[k-1]$ (cell #0 gets its input from the input signal uu)
$S_j[k]=S_{j-1}[k-1]$ $S_j[k]=Sj\text{-}_1[k-1]$ for $j>0j>0$ (each of the other cells gets its input from the previous cell)
$y[k]=S_{N-1}[k]y[k]=S_{N-1}[k]$
This effectively produces $S_j[k]=u[k-j-1]$ $S_j[k]=u[k-j-1]$ and y[k]=u[k−N] y[k]=u[k−N] because the input is delayed by j+1j+1 and NN timesteps, respectively.

6. FIELD PROGRAMMABLE GATE ARRAYS (FPGA)

The Field Programmable Gate Arrays (FPGAs) is a kind of conducting device (semiconducting) and it works on a matrix which is able to configure of logic block (CLB) and connected with a interconnects, those interconnects are programmable. Field programmable gate array can be re-programmable in any case for a desired application (or) functionality requirements after the successful manufacturing (or) development of field programmable gate array. This feature distinguishes FPGA from Application Specific Integrated Circuit (ASIC), those which are successfully costumed manufacture for a particular tasks. Although the one-time programmable (OTP) field programmable gate array are available, and the SRAM are also re-programmable as the design evolved.

6.1. FPGA Applications

Due to their programming nature, FPGA are a ideal fit for many different-different markets. As the industry leaders, Xilinx provided comprehensive solution consist of FPGA devices, advance software, and configurable, ready-to-use Internet Protocol cores for markets and the applications such as:

A. Aerospace & Defence.
B. ASIC Prototyping.
C. Audio.
D. Automotive.
E. Broadcast & Pro AV.
F. Consumer Electronics.
G. Data Centre.
H. High Performance Computing and Data Storage.
I. Industrial.
J. Medical.
K. Video & Image Processing.
L. Wired Communications.
M.Wireless Communications**.**

Table 1: Constants

NOTATIONS	DESCRIPTION
P0	Clock power
P1	Leakage power
P	Total power
A	Total Area
M0	Minimum time
M1	Minimum inputted arrived time before the clock
M2	Maximum output that required some amount of time after the clock
C	Clock time
T	Total time

7. IMPLEMENTAION PROCESS

To implement the KATAN64 lightweight block cipher on FPGA for security in wireless sensor networks we have to go through a process.
First, we need to create a Verilog code for KATAN64 that code should consist of three modules:
i. LFSR module.
ii. KEY generation module.
iii. Final module.

These modules will generate the cipher text for a given plane text. Here we need to run the XILINX ISE software then load the three modules into it now simulate the LFSR Verilog code in that we must provide inputs of clock and feedback then run the code. Now simulate the key generation Verilog code then give inputs for clock and key. Later run the code.
After completion of the above steps now we go to the final code part, now simulate the final code and give inputs for clock, input plane text of 64bit and key of 80bit.

So, we can run the entire Verilog final code of KATAN64 then we can generate the cipher text.
If the output is correct, then the simulation panel will turn into green colour otherwise red colour. Now we have to see the power factor, area factor and time factor to check these factors we have to go to summary option in that we can observe the total power consumption.

For time factor go to the implementation in that go for design option inside that option select analyse time now we can see time consumption. For area factor go to the implementation in that go for design option inside that option select analyse area now we can see area consumption. By following the above process for implementation of KATAN64 lightweight block cipher on FPGA for security in wireless sensor

networks (WSN), we are getting power, area and time factors. We can give more security than other block ciphers for wireless sensor networks.

8. PUBLISHED RESULTS OF KATAN64

Table2: Results

CONSTANTS	FPGA
P0	0.001W
P1	0.080W
P	0.081W
A	1073
M0	2.552ns
M1	3.098ns
M2	3.634ns
C	2.552ns
T	11.836ns

All the notation meanings are mentioned in table1.These results from table2 are produced by **KATAN64** light weight block cipher, which is implementation of KATAN64 of FPGA on wireless sensor networks.

9. CONCLUSION

we provided a deep discussion of lightweight block ciphers for a low type of factors (resource) wireless sensor networks. In that low type of resources wireless sensor networks devices, energy is the one of the important resources, it has priority among other resources. We focused to improve the light-weight block ciphers performances, maintaining of energy to better performance in battery life even in excess consumptions of power and to provides high privacy (SECURE) for the wireless sensor networks devices by using light-weight block cipher-KATAN64, more over the time factor which is input arrival time and output execution time are maintained. For area factor also we maintained resources space utilization. Now we can use light-weight block ciphers (KATAN64) for wireless sensor networks (WSN) confidently.

REFERENCES

[1] Daemen, J. and Rijmen, V., 1999. AES proposal: Rijndael.

[2] Madakam, S., Ramaswamy, R. and Tripathi, S., 2015. Internet of Things (IoT): A literature review. Journal of Computer and Communications, 3(05), p.164. {a}. https://doi.org/10.4236/jcc.2015.35021

[3] Hafsa Tahir, A.K. and Junaid, M., 2016. Internet of Things (IoT): An Overview of Applications and Security Issues Regarding Implementation. {b}

[4] Xu, Q., Ren, P., Song, H. and Du, Q., 2016. Security enhancement for IoT communications exposed to eavesdroppers with uncertain locations. IEEE Access, 4, pp.2840-2853. {c} https://doi.org/10.1109/ACCESS.2016.2575863

[5] Kaur, A., 2016. Internet of Things (IoT): Security and Privacy concerns. International Journal of Engineering Sciences & Research Technology. (pp. 161-165). DOI: 10.5281/zenodo.51013

[6] Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J., Seurin, Y. and Vikkelsoe, C., 2007, September. PRESENT: An ultralightweight block cipher. In CHES (Vol. 4727, pp. 450- 466). https://doi.org/10.1007/978-3-540-74735-2_31

[7] Zhang, W., Bao, Z., Lin, D., Rijmen, V., Yang, B. and Verbauwhede, I., 2015. RECTANGLE: a bit-slice lightweight block cipher suitable for multiple platforms. Science China Information Sciences, 58(12), pp.1-15.

[8] Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B. and Wingers, L., 2013. The SIMON and SPECK Families of Lightweight Block Ciphers. Cryptol ogy ePrint Archive, Report 2013/404.

[9] Suzaki, T., Minematsu, K., Morioka, S. and Kobayashi, E., 2011, November. Twine: A lightweight, versatile block cipher. In ECRYPT Workshop on Lightweight Cryptography (Vol. 2011).

[10] Engels, D.W., Saarinen, M.J.O., Schweitzer, P. and Smith, E.M., 2011. The Hummingbird-2 Lightweight Authenticated Encryption Algorithm. RFIDSec, 11, pp.19-31. https://doi.org/10.1007/978-3-642-25286-0_2

[11] Shibutani, K., Isobe, T., Hiwatari, H., Mitsuda, A., Akishita, T. and Shirai, T., 2011, September. Piccolo: An ultra-lightweight block cipher. In CHES (Vol. 6917, pp. 342-357). https://doi.org/10.1007/978-3-642-23951-9_23

[12] Bansod, G., Pisharoty, N. and Patil, A., 2016. PICO: An Ultra Lightweight and Low Power Encryption Design for Ubiquitous Computing. Defence Science Journal, 66(3). https://doi.org/10.14429/dsj.66.9276

[13] AlDabbagh, S.S.M., Shaikhli, A., Taha, I.F. and Alahmad, M.A., 2014, September. Hisec: A new lightweight block cipher algorithm. In Proceedings of the 7th International Conference on Security of Information and Networks (p. 151). ACM.

[14] Baysal, A. and Şahin, S., 2015, September. Roadrunner: A small and fast bit slice block cipher for low cost 8-bit processors. In International Workshop on Lightweight Cryptography for Security and Privacy (pp. 58-76). Springer, Cham. https://doi.org/10.1007/978-3-319-29078-2_4

[15] MumthazPookuzhy Ali, Geethu T George, 2017. "Optimised Design of Light Weight Block Cipher Lilliput with Extended GeneralisedFeistal Network (EGFN)." International Journal of Innovative Research in Science, Engineering and Technology, Vol. 6, Issue 4. Website: www.ijirset.com.

[16] Usman, M., Ahmed, I., Aslam, M.I., Khan, S. and Shah, U.A., 2017. SIT: A Lightweight Encryption Algorithm for Secure Internet of Things. arXiv preprint arXiv:1704.08688.

https://doi.org/10.14569/IJACSA.2017.080151
[17] Shirai, T., Shibutani, K., Akishita, T., Moriai, S. and Iwata, T., 2007, March. The 128-bit block cipher CLEFIA. In FSE (Vol. 4593, pp. 181-195).
[18] Needham, R.M. and Wheeler, D.J., 1997. Tea extensions. Report, Cambridge University, Cambridge, UK (October 1997).
[19] Guo J., Peyrin T., Poschmann A., and Matt Robshaw M., Preneel B. and Takagi T., 2011. The LED Block Cipher. CHES 2011, In International Association for Cryptologic Research, LNCS 6917 (pp. 326–341).
https://doi.org/10.1007/978-3-642-23951-9_22
[20] Lim, C.H. and Korkishko, T., 2005, August. mCryptona lightweight block cipher for security of low-cost RFID tags and sensors. In WISA (Vol. 3786, pp. 243- 258).
https://doi.org/10.1007/11604938_19
[21] Mohammed, A.A. and Ibadi, A.O., 2017. A Proposed Non Feistel Block Cipher Algorithm.
[22] Daemen, J. and Rijmen, V., 2001, December. The wide trail design strategy. In IMA International Conference on Cryptography and Coding (pp. 222-238). Springer, Berlin, Heidelberg.
https://doi.org/10.1007/3-540-45325-3_20
[23] Ch. Radhika rani, lakku sai jagan, Ch. Lakshmi Harika, v.v. Durga ravali Amara., 2018, light weight encryption algorithms for wireless body area networks. In International Journal of Engineering and Technology (vol. 7(2.20), pp. 64-66).
https://doi.org/10.14419/ijet.v7i2.20.11754
[24] Ch. Radhika rani, Narasinga Rao M R, Somu Venkateswarlu., 2018, Review on the Security Issues in Human Sensor Networks for Healthcare Applications, In International Journal of Engineering and Technology (vol. 7(2.32), pp. 269-274).
https://doi.org/10.14419/ijet.v7i2.32.15582
[25] Aakash Dutta, K. Naveen Kumar, N. Sai, Ch. Radhika rani, 2018, An Efficient Light Weight Cryptography Algorithm Scheme for WSN Devices Using Chaotic Map And GE. In International Journal of Pure and Applied Mathematics (vol. 118, No. 20, pp. 861-875).
[26] Ch. Radhika rani, Sadineni Srujana, Nalluri Ajith Kumar., 2019, An Analysis of Light Weight Block Ciphers in Wireless Body Area Networks, In International Journal of Innovative Technology and Exploring Engineering (ISSN. 2278-3075, vol. 8, Issue-7C2).
[27] Ch. Radhika rani, Narasinga Rao M R, Somu Venkateswarlu., 2019, Performance Metrics and Energy Evolution of a Light Weight Block Cipher in Human Sensor Networks, In International Journal of Advanced Trends in Computer Science and Engineering (vol. 8, No. 4, ISSN. 2278-3091).