Crypt(BM)_64A - A New Cipher oriented to Wireless Sensor Networks

Bac Do Thi[1] and Minh Nguyen Hieu[2]
[1]University of Information and Communication Technology, Thai Nguyen, Viet Nam
[2]Le Qui Don Technical University, Ha Noi, Viet Nam
dtbac@ictu.edu.vn; hieuminhmta@ymail.com

Abstract – **This article proposed a new cipher – Crypt(BM)_64A. This cipher was developed on the basis of Controlled Substitution Permutation Networks (CSPNs) oriented to applications in Wireless Sensor Networks (WSNs). Crypt(BM)_64A was proved to be secure according to NESSIE test and differential characteristic in order to prevent differential cryptanalysis. The proposed cipher was also implemented in FPGA for the technological parameter comparison with some commonly used ciphers.**

Keywords: Controlled Substitution Permutation Networks, Wireless Sensor Networks, encryption, differential cryptanalysis

I. INTRODUCTION

In the current time, Wireless Sensor Networks (WSNs) are applied in a many fields such as industrial monitoring and control, home and civil electricity automation, military sensing, medical and health monitoring, environmental sensing and intelligent agriculture, ... WSNs' important characteristics are: physically small in size, consecutive operating with high concentration, limited capacity in physical connection and hierarchical control, multi-applied in designing and using, reliable operation [14]. There are also factors influencing WSNs, i.e. low power consumption, cost savings, applicability, type of network, security, data throughput, delay and mobility [12,13]. Therefore, WSNs' security and reliability needs to be studied to make it suitable to their characteristics and factors mentioned above. The popular ciphers applied in WSNs are DES, RIPEMD, PMAC, AES, …[14,15,16]. These ciphers however are designed in accordance with multi-target but not the application in WSNs only. Thus, there will be certain constrains when using those ciphers in WSNs.

The use of data-dependent transformations has been an area of increasing interest for the designers of ciphers. Data-Dependent Permutations (DDP) has attracted much attention the last few years in cryptography [9,10,11]. Recently a class of the advanced DDP-like Operations (DDOs) has been proposed [3,11] to increase the efficiency of the hardware implementation of the DDO-based ciphers. In particular, data-dependent (DD) operations (DDOs) provide a fast and simple cryptologic primitive when implemented in hardware.

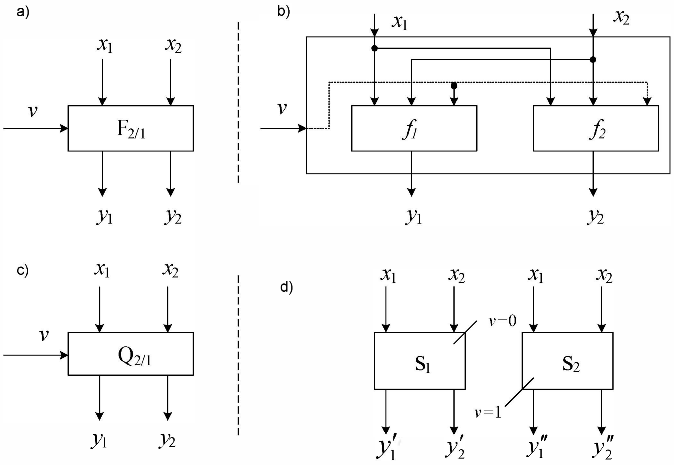
This article proposed Crypt(BM)_64A, a new cipher oriented to application in WSNs. This cipher was proved to be suitable to WSNs.

The article was organized as follow. The second section presents the design of controlled operations in terms of CSPNs (CSPNs - Controlled Substitution Permutation Networks). The thirds section describes the architecture of Crypt(BM)_64A. The fourth and fifth provide the security estimation of Crypt(BM)_64A according to NESSIE standards and the efficiency estimation of this cipher when implemented in FPGA by making a comparison to the commonly used ciphers.

II. STRUCTURE OF CSPNS

A. *Structure of Controlled Elements (CEs) $F_{2/1}$*

CEs with $F_{2/1}$ type is described in Figure 1 by the following ways [3,11]:



$$y_1 = x_2 v \oplus x_1 \oplus x_2$$
$$y_2 = x_1 v \oplus x_2$$

Figure 1. Structure of Controlled Elements $F_{2/1}$; a. General structure; b. Representation of $F_{2/1}$ of Boolean Function (BF); c. Specific structure of $Q_{2/1}$ with correspondent BF; d. Representation of $F_{2/1}$ as two substitutions.

In Figure 1a, $F_{2/1}$ is represented as a black box with 2 input bits (x_1, x_2); a controlling bit v and 2 output bits (y_1, y_2). CEs $F_{2/1}$ is responsible for transforming input values into output values and this transformation is dependent of value of the controlling bit v.

In Figure 1b, $F_{2/1}$ is represented as a pair of BF in three variables $y_1 = f_1(x_1, x_2, v)$; $y_2 = f_2(x_1, x_2, v)$. Figure 1c describe a pair of specific BF. In Figure 1d, $F_{2/1}$ is represented as two substitu-

978-1-4673-4352-7/12/$31.00 ©2012 IEEE

tion (S_1, S_2), where S_1 is correspondent with the case $v = 0$ and S_2 is correspondent with the case $v = 1$.

According to [3, 17], there is a great deal of CEs variants. Figure 1c is a specific CE of $Q_{2/1}$ type. The selection of CEs suitable to design efficient Cryptographic DDO is based on the following criteria:

1) Each of two outputs of the CEs should be a nonlinear BF having maximum possible NL for balanced BFs in three variables.
2) Each modification of the CEs should be a bijective transformation $(x_1, x_2) \rightarrow (y_1, y_2)$.
3) The linear combination of two outputs of the CEs, i.e., $f = y_1 \oplus y_2$, should have maximum possible NL for balanced BFs in three variables.
4) Each modification of the CEs should be an involution

There exist 24 possible variants can be used in designing CEs $F_{2/1}$ satisfying the 4 above criteria as indicated in [3].

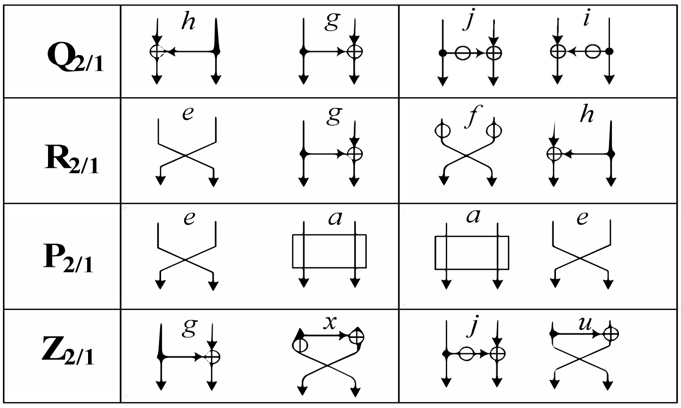


Figure 2. Some pairs of CEs $F_{2/1}$.

The combination of 2 in 24 above cases will create a great amount of possible variants of CEs $F_{2/1}$ (see Figure 2). They are divided into 4 subclasses: $\{Q_{2/1}\}$, $\{R_{2/1}\}$, $\{Z_{2/1}\}$, $\{P_{2/1}\}$[3]. The most interesting subclass of CEs namely $Q_{2/1}$ with the feature $Q_{2/1} = Q_{2/1}^{-1}$. In Crypt(BM)_64A design, the $\{Q_{2/1}\}$ was used as it satisfies the criteria each specific type of CE also include its inverse element i.e. $Q_{2/1} = Q_{2/1}^{-1}$. $Q_{2/1}$ can be one of following pairs: (h, g), (j, i), (u, t), (x, r), … In figure 1c is described as in the case of (h, g), i.e., if $v = 0$ then the relationship between input and output bits is described as h, and if $v = 1$ then the relationship between input and output bits is described as g (see Figure 2).

B. Structure of $Q_{n/m}$ and $Q_{n/m}^{-1}$

Basing on CE $Q_{2/1}$ the design of CE $Q_{n/m}$ and $Q_{n/m}^{-1}$ was proposed (see Figure 3). This design was for using cryptographic algorithms and implementation in FPGAs oriented to applications in WSNs. Hence it is efficient and cost savings.

In Figure 3a, the structure of $Q_{n/m}$ including s layers named from layer 1 to layer s. Among the layers $\pi_1, \pi_2, \ldots, \pi_{s-1}$ was used in order to creating the necessary chaotic in encryption. Each layer used $n/2$ CEs which are parallel connected. The $Q_{2/1}$ was used as CE. In Figure 3b the structure of $Q_{n/m}^{-1}$ is the same as the controlling vectors and the fixed permutation is reversed with the inverse order. The structure of $Q_{n/m}$ and $Q_{n/m}^{-1}$ are reversible.

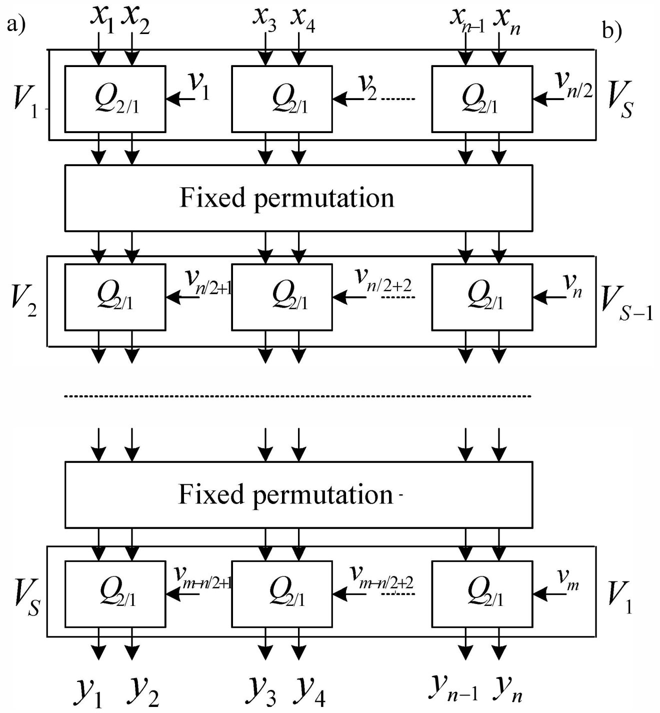


Figure 3. Structure of $Q_{n/m}$ (a) và $Q^{-1}_{n/m}$(b).

With the above design basis, we proposed the structure of CEs $Q_{4/4}$, $Q^{-1}_{4/4}$ and $Q_{16/32}$, $Q^{-1}_{16/32}$ which were to be used in the construction of our cipher. The structures of $Q_{4/4}$, $Q^{-1}_{4/4}$ were described in Figure 4 and in Figure 5 presented the structures of $Q_{16/32}$, $Q^{-1}_{16/32}$. In Figure 4, $Q_{4/4}$ was designed with 4 CEs $Q_{2/1}$ divided to 2 layers, each layer consisted of 2 elements parallel connected. Between the 2 layers is fixed permutation π defined as π = (1)(2,3)(3,2)(4) (where (1) means the first bit is not permuted; (2,3) means the second bit was permuted for the third bit). The controlling vector of the first layer is $V_1 = (v_1^{(1)}, v_2^{(1)})$ and in the second layer is $V_2 = (v_3^{(2)}, v_4^{(2)})$. The only difference between $Q_{4/4}$ and $Q^{-1}_{4/4}$ is that the controlling vector of the first layer (or 2) of $Q^{-1}_{4/4}$ is the controlling vector of the second layer (or 1) of $Q_{4/4}$. In Figure 5, $Q_{16/32}$ is designed with 8 CEs divided to 2 layers, each layer consisted of 4 elements parallel connected. The controlling vectors of the first layer are V_1, V_2 and of the second layer are V_3, V_4. Between the 2 layers are fixed permutation π defined as:

$$\pi = (1)(2,5)(3,9)(4,13)(5,2)(6)(7,10)(8,14)(9,3)$$
$$(10,7)(11)(12,15)(13,4)(14,8)(15,12)(16).$$

As $Q_{16/32}$ and $Q^{-1}_{16/32}$ are mutual inverse so they differ only the order of controlling vector, i.e. the controlling vector of the first layer (or 2) of $Q^{-1}_{16/32}$ is the controlling vector of the second layer (or 1) of $Q_{16/32}$.

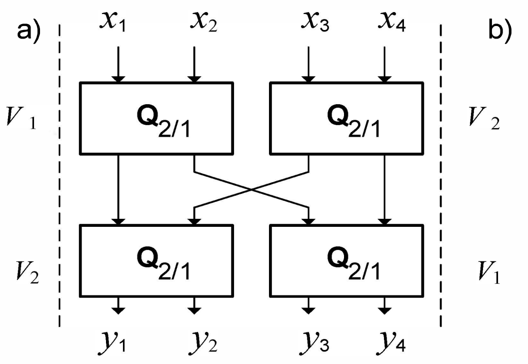


Figure 4. The structure of $Q_{4/4}$ and $Q^{-1}_{4/4}$

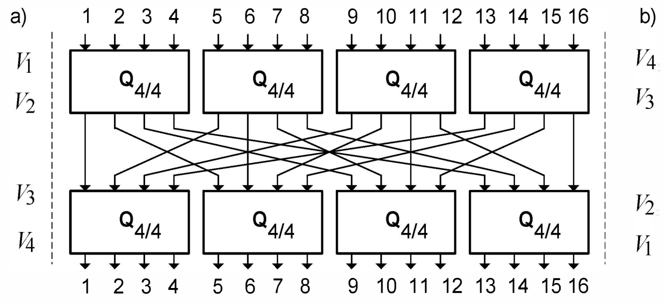


Figure 5. The structure of $Q_{16/32}$ and $Q^{-1}_{16/32}$.

III. Construction of Crypt(BM)_64A

A. Objectives

The objectives of designing this cipher are the following:

1) Application in WSNs.
2) Implemented in hardware with FPGA devices for cost reducing.
3) The structure in support of performing both encryption and decryption for cost savings.
4) Flexible mechanism, high speed in requirement of frequent change of key. Being supportive of 128 bit, 192 bit, and 256 bit keys.
5) Equally secure in comparison with other commonly used ciphers.
6) Easily developed when there are changes in elements to provide more space than the other ciphers.
7) Consecutive processing the 2 branche.

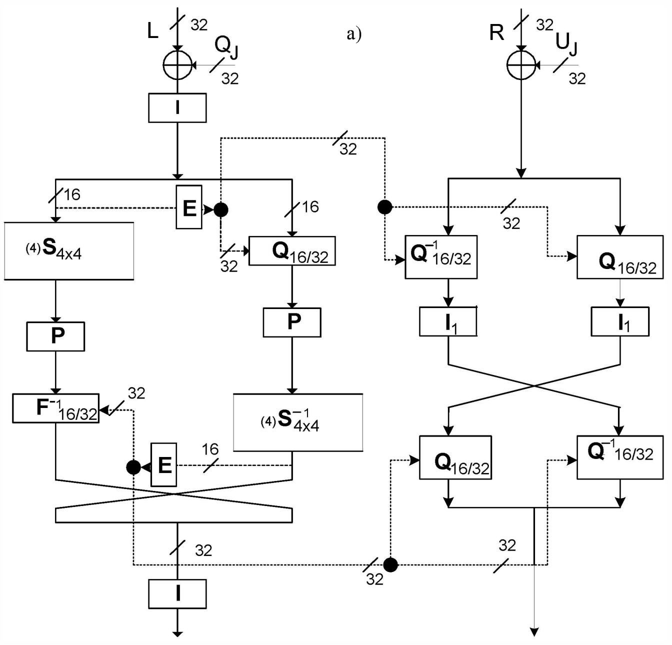


Figure 6. Cipher Crypt(BM)_64A: Procedure $\text{Crypt}^{(e)}$.

B. The Design Scheme

In accordance with the above objectives, the cipher was particularly described with one encryption round in Figure 6. The cipher was performed with ten basic encryption rounds and one final transformation. Among the rounds, there were permutations inversing the left and right branches.

Ciphering procedure of Crypt(BM)_64 is described as follows. $C = \text{T}^{(e=0)}(M, K)$ and $M = \text{T}^{(e=1)}(C, K)$, where M is the plaintext, C is the ciphertext ($M, C \in \{0,1\}^{64}$), T is the transformation function, and $e \in \{0,1\}$ is a parameter defining encryption ($e = 0$) or decryption ($e = 1$) mode. First data block is divided into two 32-bit subblocks L and R and then using the procedure $\text{Crypt}^{(e)}$ ten encryption rounds are performed. The last round is followed by final transformation (FT).

The steps in performing the ciphers:

1) For $i = 1$ to 9 do:
   $\{(L, R) \leftarrow \text{Crypt}^{(e)}(L, R, Q_j, U_j); (L, R) \leftarrow (R, L)\}$.
2) Perform transformation:
   $\{(L, R) \leftarrow \text{Crypt}^{(e)}(L, R, Q_{10}, U_{10})\}$.
3) Perform final transformation:
   $\{(L, R) \leftarrow (L \oplus Q_{11}, R \oplus U_{11}); (L, R) \leftarrow (L, R)\}$.

Figure 6 describes one basic encryption round, where I, I_1, P are presented as below:

I: (1)(2,9)(3,17)(4,25)(5)(6,13)(7,21)(8,29)(10)(11,18)(12,26)
(14)(15,22)(16,30)(19)(20,27)(23)(24,31)(28)(32)

I_1: (1,9)(2,13)(3,10)(4,14)(5,11)(6,15)(7,12)(8,16)

P: (1)(2,5)(3,9)(4,13)(6)(7,10)(8,14)(11)(12,15)(16)

The controlling vectors of $Q_{16/32}$ and $Q^{-1}_{16/32}$ were create by the expansion of box E described by the formula $\text{E}(X) = (X, X^{<<<8})$.

The cipher used the substitution 4x4 as in Serpent [7] in order to enhance the reliability of Crypt(BM)_64A. The substitution is presented in Table I.

TABLE I. REPRESENTATION OF BOX S_{4X4}.

	0	1	2	3	4	5	6	7
S_0/S_0^{-1}	14/14	4/3	13/4	1/8	2/1	15/12	11/10	8/15
S_1/S_1^{-1}	3/9	13/10	4/5	7/0	15/2	2/15	8/12	14/3
S_2/S_2^{-1}	10/1	0/8	9/14	14/5	6/13	3/7	15/4	5/11
S_3/S_3^{-1}	1/12	4/0	11/15	13/5	12/1	3/13	7/10	14/6
	8	9	10	11	12	13	14	15
S_0/S_0^{-1}	3/7	10/13	6/9	12/6	5/11	9/2	0/0	7/5
S_1/S_1^{-1}	12/6	0/13	1/11	10/14	6/8	9/1	11/7	5/4
S_2/S_2^{-1}	1/15	13/2	12/0	7/12	11/10	4/9	2/3	8/6
S_3/S_3^{-1}	10/11	15/14	6/8	8/2	0/4	5/3	9/7	2/9

C. *Key scheduling*

Subkeys $K_i \in \{0,1\}^{32}$ of the 128, 192 and 256-bit secret key K = (K_1, K_2, ..., K_i) are used directly in procedure Crypt as round keys Q_j /U_j (encryption) or Q'_j/U'_j (decryption) specified in Table II, III and IV. Thus, no preprocessing the secret key is used. More over, in each round transformation we use only one 32-bit subkey combined with both the left and the right data subblocks. This makes the hardware implementation to be cheaper.

TABLE II. THE KEY SCHEDULING IN CRYPT(BM)_64 WITH 128 BITS KEY (J = 11 CORRESPONDS TO FINAL TRANSFORMATION)

No. rounds j	1	2	3	4	5	6
Enc Q_j/U_j	K_1/K_3	K_2/K_3	K_3/K_2	K_4/K_1	K_4/K_2	K_1/K_3
Dec Q'_j/U'_j	K_1/K_3	K_4/K_1	K_2/K_4	K_1/K_2	K_4/K_3	K_1/K_3
No. rounds j	7	8	9	10	11	
Enc Q_j/U_j	K_4/K_3	K_1/K_2	K_2/K_4	K_4/K_1	K_1/K_3	
Dec Q'_j/U'_j	K_4/K_2	K_4/K_1	K_3/K_2	K_2/K_3	K_1/K_3	

TABLE III. THE KEY SCHEDULING IN CRYPT(BM)_64 WITH 192 BITS KEY (J = 11 CORRESPONDS TO FINAL TRANSFORMATION)

No. rounds j	1	2	3	4	5	6
Enc Q_j/U_j	K_1/K_3	K_2/K_4	K_5/K_6	K_2/K_1	K_4/K_6	K_5/K_3
Dec Q'_j/U'_j	K_1/K_3	K_6/K_5	K_3/K_2	K_1/K_3	K_2/K_3	K_5/K_3
No. rounds j	7	8	9	10	11	
Enc Q_j/U_j	K_2/K_3	K_1/K_3	K_3/K_2	K_6/K_5	K_1/K_3	
Dec Q'_j/U'_j	K_4/K_6	K_2/K_1	K_5/K_6	K_2/K_4	K_1/K_3	

TABLE IV. THE KEY SCHEDULING IN CRYPT(BM)_64 WITH 256 BITS KEY (J = 11 CORRESPONDS TO FINAL TRANSFORMATION)

No. rounds j	1	2	3	4	5	6
Enc Q_j/U_j	K_1/K_3	K_2/K_4	K_5/K_8	K_7/K_6	K_4/K_2	K_2/K_8
Dec Q'_j/U'_j	K_1/K_3	K_7/K_4	K_3/K_7	K_1/K_3	K_6/K_5	K_2/K_8
No. rounds j	7	8	9	10	11	
Enc Q_j/U_j	K_6/K_5	K_1/K_3	K_3/K_7	K_7/K_4	K_1/K_3	
Dec Q'_j/U'_j	K_4/K_2	K_7/K_6	K_5/K_8	K_2/K_4	K_1/K_3	

IV. SECURITY ESTIMATION OF CRYPT(BM)_64A

To estimate the security of the proposed cipher, there are two ways of estimation:

1) NESSIE Test
2) Differential cryptanalysis

A. *NESSIE Test*

According to NESSIE announcement [8], the security of ciphers is estimated basing on 4 followings criteria:

1) The average number of output bits changed when 1 input bit is changed (denoted: d_1).
2) The degree of complete change (denoted: d_c).
3) The degree of avalanche effect (denoted: d_a).
4) The degree of suitability to strict avalanche effect standard (denoted: d_{sa}).

TABLE V. TEST RESULT ACCORDING TO NESSIE STANDARDS

Number of rounds	$\#K = 100$ $\#X = 100$			
	(1) = d_1	(2) = d_c	(3) = d_a	(4) = d_{sa}
1	15.736512	0.613281	0.491766	0.486935
2	31.202336	1.000000	0.975073	0.972059
3	31.985408	1.000000	0.998791	0.991877
4	32.007180	1.000000	0.999117	0.991930
5	31.986433	1.000000	0.999024	0.991982
6	31.997164	1.000000	0.998959	0.991927
7	32.009150	1.000000	0.998917	0.992038
8	31.995650	1.000000	0.999079	0.991945
9	32.002222	1.000000	0.998868	0.992044
10	31.999683	1.000000	0.999011	0.992078

Basing on the above criteria, the security or the transformations in the ciphers will be the best when the following conditions happen consecutively: $d_c = 1$, $d_a \approx 1$, $d_{sa} \approx 1$ và $d_1 \approx ½ n$.

On that basis, security estimation of Crypt(BM)_64A was carried out with the number of tests were 10.000 (in which Data: 100; Key: 100) and the results have shown that with only 3 rounds are sufficent to satisfy the test criteria, i.e. $d_c = 1$, $d_a \approx 1$, $d_{sa} \approx 1$ và $d_1 \approx 32$.

B. *Differential Cryptanalysis*

According to some materials as [3,9], differential characteristic is one of morden method in security estimation of ciphers against differential analysis. Differential characteristic of expanded controlled operations depends on the construction, distribution of controlling vectors and the difference of the CEs used in the structure.

According to the analysis of differential characteristics, it is concluded that with of smaller weight differential trail, the higher possibility of the existence of differential trail. Let Δ_i be the differential trail, where i is the weight of differential trail and let $P = F_{n/m}(\Delta_i \to \Delta_j)$ be the possibility of the existence of Δ_j from the transformation of $F_{n/m}$.

TABLE VI. PROBABILITIES OF DIFFERENTIAL CHARACTERISTICS

i	0	1	2	0	1	2	1	2	1	2	0	1	2
j	0	0	0	1	1	1	1	1	2	2	2	2	2
k	1	1	1	1	1	1	0	0	0	0	1	1	1
P	1/4	1/2	1/4	1/4	1/2	1/4	1/2	1/2	1	0	1/4	1/2	1/4

With the scheme of Crypt(BM)_64A presented in the section 3 with the use of CE $Q_{2/1}$, we will analyze the existence of differential trail Δ_1 after 2 encryption rounds. First we need to analyze the existence of differential trail Δ_1 in each $Q_{2/1}$ element, i.e. through the different trails we have the possibility of existence as indicated in Table V. As we can see, the differential trail passes through the left branch (see Figure 7), this trail will pass through the expansion E(16→32) then we receive the differential trail with weight = 2 and it is easy to find the possibility $P_3 = Q_{16/32}(\Delta_0 \to \Delta_0)$. $P_1 = 1/2$ is the possibility of the differential trail passes through the left of the left branch. The analysis of the table S gives the possibility $P_2 = S(\Delta_1 \to \Delta_2) = 1/2$ and the differential trail with weight equal 2 will pass through the transformation of $Q_{16/32}$ and we receive $P_4 = Q_{16/32}(\Delta_2 \to \Delta_1)$. The differential trail $\Delta_1$in the first round will pass the second round. Hence, we need to calculate the possibility $P_5 = Q_{16/32}(\Delta_1 \to \Delta_1)$ only.

Only after 2 encryption rounds are sufficient to satisfy the value less than 2^{-26} $(P = P_1 \times P_2 \times P_3^3 \times P_4 \times P_5^2 \approx 2^{-26})$.

Due to the experimental software, the results have shown that only after 6 encryption rounds the value is less than $2^{-80,094}$. Thus, only 6 rounds are enough to prevent the differential cryptanalysis. However, in order for the security 10 rounds are selected to prevent other types of attacks.

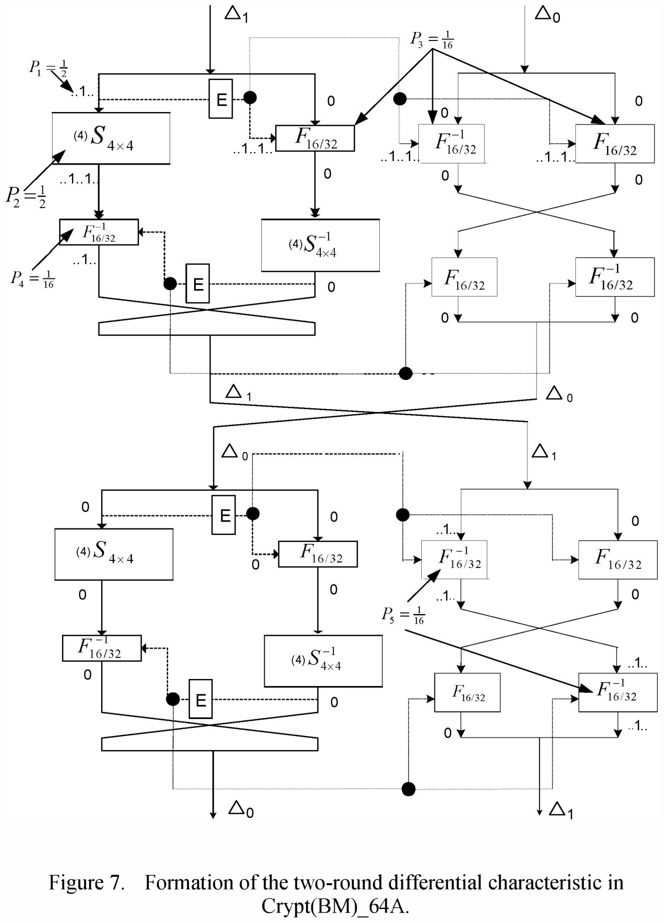


Figure 7. Formation of the two-round differential characteristic in Crypt(BM)_64A.

V. Implementation of Crypt(BM)_64A in FPGA

Implementation and estimation of Crypt(BM)-64A and were carried out to make a comparision with other commonly used ciphers. Criteria for the comparison included [10,3]: area of computation according to the number of slice; the frequence caculated according to MHz unit; throughput caculated according to the number of bits processed with the same period of time; throughput/area. Crypt(BM)_64A was implemented in Sapartan 3E device with XC3S1200E-FT256.

The estimation was also carried out according to 2 strategies: Iterative Looping (IL) and PiPelining (PP). The IL architecture performed 1 cycle and the algorithm was iterated *n* times by returning the results to the previous round. With this design, the area is smaller but more clock cycles are used and the encryption process is relatively slow. With PP architecture, the rounds are reconstructed. This design offers higher speed but requiring bigger area, invulnerably hierarchical structure providing better performance enhancing security for the design.

The following texts are the results collected after the implementation of Crypt(BM)_64A in FPGA with the Spartan 3E family with XC3S1200E-FT256 descent basing on architecture IL (Table VII) and PP (Table VIII).

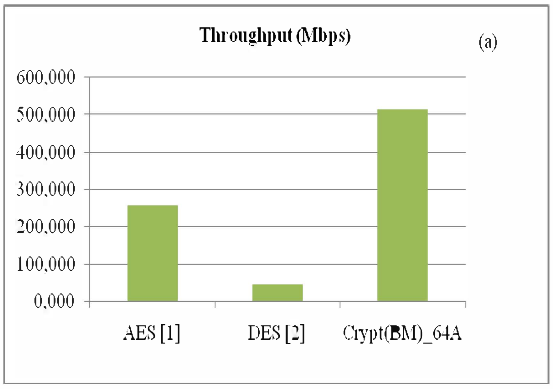
TABLE VII. Comparison AES, DES with Crypt(BM)_64A (IL)

	Area (slice)	Frequency (MHz)	Throughput (Mbps)	Through-put/Area
AES [1]	2744	20,192	258,458	0,094
DES [2]	2888	11,534	46,136	0,016
Crypt(BM)_64A	383	79,927	511,533	1,336

TABLE VIII. Comparison AES, DES with Crypt(BM)_64A (PP)

	Area (slice)	Frequency (MHz)	Throughput (Mbps)	Through-put/Area
AES [1]	4272	22,410	2868,480	0,671
DES [2]	2964	25,189	1612,091	0,544
Crypt(BM)_64A	1988	80,206	5133,184	2,582

According to the statistics above and the chart in Figure 8, the cipher we constructed is more efficient in many aspects, particularly: throughput and throughput/area. The throughput of Crypt(BM)_64A is approximately 2 folds compared to AES and 11 folds compared to DES (Figure 8a). With regard to the implementation cost of throughput/area, Crypt(BM)_64A is more than 24 times higher compared to AES and more than 83 times higher compared to DES (Figure 8b).



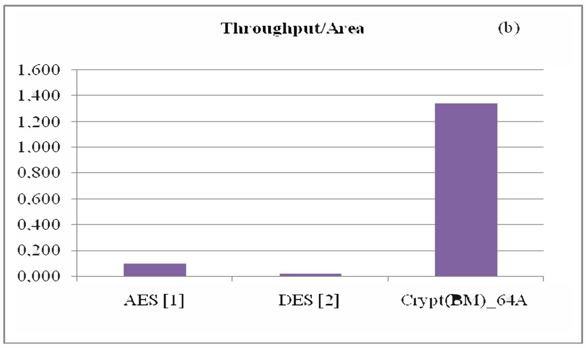


Figure 8. Efficiency comparison of Crypt(BM)_64A with AES, DES basing on architecture IL in FPGA.

With the architecture PP, the proposed cipher has prominent characteristics in area and throughput/area (Figure 9). Its throughput is approximately 2 times higher than that of AES and 3 times higher as that of DES (Figure 9a). Regarding the estimation of implementation cost of throughput/area, Crypt(BM)_64A is superior to AES and DES (Figure 9b).

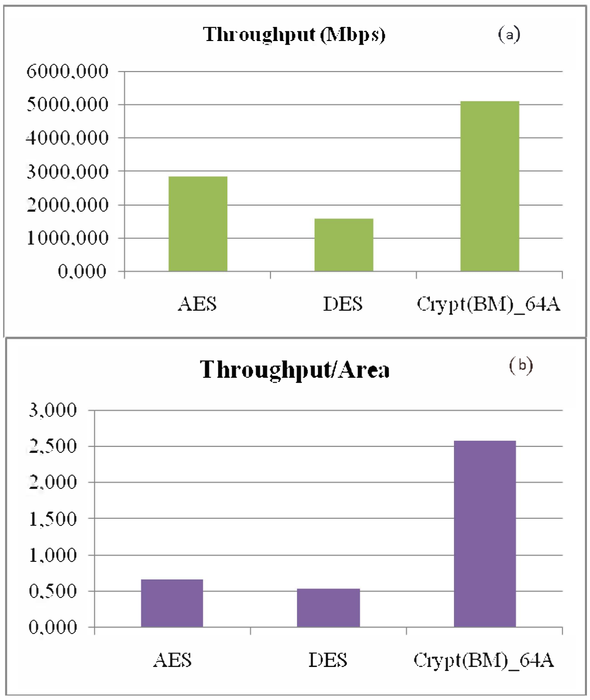


Figure 9. Efficiency comparison of Crypt(BM)_64A with AES, DES basing on architecture PP in FPGA

VI. Conclusion

With the objective of application in WSNs-oriented, Crypt(BM)_64A satisfies all of the criteria for security according of NESSIE standards. It also offers simple computation, small amount of power consumption, high speed computation and implementation in FPGA-oriented.

By analyzing the difference, it was indicated that Crypt_64A has an interesting differential trail and is superior to other existing ciphers which have the same developing orientation.

This proved the security of Crypt(BM)_64A to be suitable to the proposed design objectives. Crypt(BM)_64A provides more options for the cryptographical solutions in the growth of technologies and services.

References

[1] A Satoh and K. Takano. A Scalable Dual-Field Elliptic Curve Cryptographic Processor. IEEE Transactions on Computers, 52(4):449-460, April 2003.

[2] Francisco Rodriguez-Henriquez N.A. Saqib A. Diaz-Perez Cetin Kaya K09. Cryptographic Algorithms on Reconfigurable Hardware, Springer, Printed in the United States of America.

[3] Nikolay A. Moldovyan, Alexander A. Moldovyan, Data-Driven Block ciphers for fast telecommunication systems, Auerbach Publications Taylor & Francis Group, New York, pp.72-80, 2008.

[4] Moldovyan, A.A., Moldovyan, N.A., and Sklavos, N. 2006. Controlled elements for designing ciphers suitable to efficient VLSI implementation. Telecommunication Systems 32, pp 149‑63.

[5] Moldovyan, N.A., Eremeev, M.A., Sklavos, N., and Koufopavlou, O. 2004. New class of the FPGA efficient cryptographic primitives. Proceedings of the ISCAS 2004. Vancouver, Canada. Vol. II, pp. 553‑56.

[6] Moldovyan, N.A., Moldovyan, A.A., Eremeev, M.A., and Summerville, D.H. 2004. Wireless networks security and cipher design based on data-dependent operations: Classification of the FPGA suitable controlled elements. Proceedings of the CCCT- 2004. Vol. VII, Austin, TX. pp. 123‑28.

[7] R. Anderson, E. Biham, and L. Knudsen, “Serpent: a proposal for the advanced encryption standard,” in 1st Advanced Encryption Standard Candidate Conference Proceedings, Venture, California, Aug. 20-22, 1998.

[8] B. Preneel et al., Comments by the NESSIE project on the AES finalists, May 24, 2000 (http://www.nist.gov/aes).

[9] Nguyen Hieu Minh, Nguyen Thien Luan, and Luu Hong Dung, KT-64: A New Block Cipher Suitable to Efficient FPGA Implementation, IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.1, January 2010.

[10] Moldovyan, N.A., Sklavos, N., Moldovyan, A.A., and Koufopavlou, O. 2005. Chess-64, A block cipher based on data-dependent operations: Design variants and hardware implementation efficiency. Asian Journal of Information Technology 4: 320‑28.

[11] Moldovyan, N.A., Moldovyan, A.A., Eremeev, M.A., and N. Sklavos. 2006. New class of cryptographic primitives and cipher design for network security. International Journal of Network Security 2: 114‑25.

[12] Muaz Niazi, Amir Hussain,Sensing Emergence in Complex Systems, IEEE Sensors Journal (In-press, 2011).

[13] Muaz Niazi, Amir Hussain, Agent based Tools for Modeling and Simulation of Self-Organization in Peer-to-Peer, Ad-Hoc and other Complex Networks, Feature Issue, IEEE Communications Magazine, Vol.47 No.3, March 2009, pp 163–173.

[14] Hu F, Ziobro J, Tillett J, Sharma N. K, “Secure wireless sensor networks: problems and solutions,” J. Syst., Cybern. Inf., 11(9):419-439, 2004.

[15] Saraogi M, “Security in Wireless Sensor Networks,” Project Paper at Computer and Network Security, Sections 494/4594/9. University of Tennesse, 2006.

[16] Mauw S, Vessem I, Bos B, “Forward secure communication in wireless sensor networks,” LNCS, 3934:32-42, 2006.

[17] Nguyen Hieu Minh, Do Thi Bac, Ho Ngoc Duy, “New SDDO-Based Block Cipher for Wireless Sensor Network Security”, IJCSNS - International Journal of Computer Science and Network Security, VOL.10 No.3, March 2010.