FPGA Design and Implementation of High Secure Channel Coding Based AES

Mostafa Ahmed Mohamed Sayed(✉), Liu Rongke, and Zhao Ling

School of Electronic and Information Engineering,
Beihang University, Beijing, China
mostafa_adawy@ymail.com, rongke_liu@buaa.edu.cn

Abstract. However applying encryption in physical layer reveals high levels of security, it can increase the system complexity and it can affect the communication reliability. This paper shows how to overcome these problems, where it doesn't only show the design of combined Low Density Parity Check (LDPC) code and Customized Stream Advanced Encryption Standard (CSAES) to increase the security level, but it also introduces a practical implementation for it. The proposed algorithm is designed in order to optimally exploit the hardware resources, and FPGA parallelism to achieve high throughput and to save hardware size. The design method shows how channel coding can be exploited to increase the security and resist attacks without affecting the communication reliability. The proposed algorithm is implemented on (Cyclone-IV4CE115) to achieve variable throughputs. It achieves 604 Mbps and 10^{-6} BER at SNR = 3.25 dB, while it can achieve 2 Gbps for SNR greater than 6 dB. NIST tests are applied to check the ciphered output randomness, and also the resistance of the algorithm against some attacks is discussed.

Keywords: LDPC · AES · McEliece · Combined encryption-channel

1 Introduction and Overview

The methodology of integrating the channel coding algorithm and encryption algorithm is often based on using public generator matrix that gives no information about its parity check matrix such as public key encryption. In 1978, the first channel coding based cryptocodes algorithm was introduced by McEliece Public key (MP) cryptography [1]. It begins with choosing the irreducible t-degree polynomial (Goppa Code) for the parity check matrix H; calculate G from Parity matrix H, then choosing $S_{k\times k}$ scrambling matrix and $P_{n\times n}$ permutation matrix. The next step is to publicly calculate the public key from (1), while keeping the ingredient of the public key secret so only who have the secret key can recover the information from the codeword. In encryption, the sender can encrypt input data $\mathcal{U}$ by multiplying it by the public generator matrix G', then adding random intentional errors to obtain encrypted word C as shown in (2) [1].

This work was supported by the National Natural Science Foundation of China (61401010).

Q. Chen et al. (Eds.): ChinaCom 2016, Part II, LNICST 210, pp. 355–366, 2018.
DOI: 10.1007/978-3-319-66628-0_34

$$G' = S \cdot G \cdot P \tag{1}$$

$$C = \mathcal{U} \cdot G' + e_{\text{int}} \tag{2}$$

The weight of e must be chosen less than the error correction capability of the decoder. Because only the authenticated receiver has the secret key, it can remove the intentional errors and recover the information as shown in the next equation.

$$\widehat{\mathcal{X}} = C \cdot P^{-1} = \mathcal{U} \cdot S \cdot G + e_{\text{int}} \cdot P^{-1} \tag{3}$$

Despite the robustness of MP algorithm, its key length and complexity directed the researchers to try other more lower complexity solutions such as replacing Goppa codes by LDPC codes [2]. However LDPC codes reduce the complexity, it makes the algorithm vulnerable to many attacks that exploit the sparse nature of the LDPC code. Different methods are proposed to solve this problem, such as using of more denser matrices, or QC-LDPC based random differences family code construction [3], using irregular QC-LDPC [4, 5], or selection of better intentional error vectors methods [6]. Other methods of MP based LDPC codes apply the addition of intentional error in modulation function to increase the security [7]. Many researches exploit the MP structure and the AES structure to combine between them. These methods based on exploiting the nature of LDPC code as high diffusion codes to replace the multiplication MixColumns operation of AES and to reduce the number of AES rounds. Because such systems contain both public key encryption algorithm, and also private key encryption, it is called hybrid systems. Some researches show that only six rounds [8] or seven rounds [9] or nine rounds [10] from AES is enough for such combined system.

The evaluation of security level for a certain cryptosystem is defined by the number of operations that are required to break it, which called work factor as described in [4], or cryptanalysis complexity as defined in [11]. This factor represents the capability of the algorithm to resist attacks. For example AES system actually has four parameters in each round. These parameters are the field irreducible polynomial, the affine transformation for Sbox, the offsets for ShiftRows, and the polynomials for MixColumns. Rijndael was designed to have resistance against the majority of known attacks based on its linear and nonlinear function represented in diffusing layer and substitution layer (Sbox) respectively [12]. The Sbox nonlinearity is measured by its differential probability, and output correlation. AES Sbox Differential Probability (DP) is $\delta = 2^{-6}$, and Sbox maximum correlation $\kappa = 2^{-3}$ [13]. As described in [14] the resistance of AES against differential cryptanalysis depends on both the non-linear building blocks and the linear mixing maps interconnecting them (i.e. MixColumns and Shiftrows). Super boxes include both linear and non-linear components. The differential property of the linear mixing map is called the differential branch number which can be calculated from (4). Branch number $\beta(\Phi)$ is used to determine the bound of Expected Differential Probability (EDP) of the Super box as described in (5) based on differential property δ of its nonlinear component (Sbox).

$$\beta(\Phi) \min_{a \neq 0.} (\omega_d(a) + \omega_o(\phi(a))) \tag{4}$$

$$\max_{a \neq 0} \mathrm{EDP} \leq \left(\delta^{\beta(\Phi)}\right) \tag{5}$$

Where ω_d the weight of the input difference and $\omega_o(\phi(a))$ is the weight of the output of the linear mixing mapping. The maximum branch number $\beta(\Phi)$ obtained from EDP/DP as explained in [11, 14] for four rounds of AES is 2^{25}. So the maximum differential probability equals δ^{25}, and the maximum linear probability equals κ^{25}.

To decide the algorithm strength, the maximum differential cryptanalysis must be lower than 2^{-127}, and maximum linear probability must be lower than 2^{-64} to achieve complexity higher than $O(2^{128})$. Customized Sbox has to be tested to check its strength as shown in [15]. The nonlinearity test indicates the minimum hamming distance between its output 2^n binary string and n variable affine transformation. Strict Avalanche Criteria (SAC) test and Propagation Characteristic (PC) test reflect the relationship between the input changes of the Sbox to its output changes, to pass this test, half of the output must change randomly.

The proposed algorithm LDPC-CSAES introduces a practical algorithm that achieves a better level of security compared with previous work. It shows how the integration can increase the system capability to resist attacks, and achieves high throughput, while keeping the error correction performance without effect. It is based on stream AES to encrypt the data before and after LDPC.

Stream ciphers are used in order to not affect the error correction rate, but it requires synchronization, for this reason, a novel idea for synchronization that increases the security and resists attacks is introduced.

The security improvement in the proposed algorithm is based on the following:

- Double AES size, AES parameters customization, and shared shuffling function [16].
- CTR mode of operation combined with LDPC, where CTR mode has better resistance against attacks as described in [17].
- Adding extra data and permutation [18], Parallel processing.
- Using of Sync Word (SW) to prevent modifications as will be discussed later.

The next section discusses the security, the complexity, the reliability degradation problems that are targeted to be solved by LDPC_CSAES. Section 3 explains the proposed algorithm, while Sect. 4 explains its FPGA implementation. Section 5 discusses the results, related work comparison, and LDPC-CSAES resistance to attacks.

2 Problem Formulation

The tradeoff between error correction capability of the algorithm, system complexity, and security level, especially in the presence of side channel attacks, is a serious problem. The problems that face the McEliece like algorithm designers embody in choosing random matrix and method of generating error vector, where adding fixed

errors can be removed very easily, while adding variable errors is restricted by the capability of error correction of the decoder, which cannot exceed the minimum hamming distance between codes according to the decoding principles [6], and it resulted in performance degradation. So the tradeoff between error correction rate and security represent a real problem in this situation. The MP secret ingredient can be easily extracted by Power analysis (PA) attacks, the problem that requires adding more complications to the system [19].

Using of joint AES–LDPC, increases the security level of MP like algorithm, but it increases the complexity also, and it is based on ECB mode of operation that resulted in reducing the error capability of the system where one bit error means a frame error because of AES avalanche criteria. The second problem resulted from using ECB mode its vulnerability to attacks [17]. Actually joint AES–LDPC is based on reducing some rounds of conventional AES in order to reduce the complexity, so it is always have low security level compared with conventional systems that contain separate AES and separate LDPC. Increasing the security level of joint AES guided the researchers to increase the no of rounds again from 6 rounds in 2008 [8], to 7 rounds in 2013 [9], to 9 rounds in 2014 [10], ends with 10 rounds again as conventional AES in 2015 [20]. The JASALC method described in [11] based on replacing the MixColumns operation by QC-LDPC parity matrix, and interlace of other AES round's functions with QC-LDPC parity matrix. It is claimed that this method has low complexity, while decoding operation based on Sbox of soft information input, and dual step decoding actually increase the complexity.

3 The Proposed Algorithm and Related Works

Unlike other algorithms, the proposed algorithm is based on CTR mode of operation instead of ECB mode, where CTR mode has better resistance against PA attacks compared with ECB [17]. The problem of CTR mode embodies in its need for synchronization. The LDPC-CSAES introduces a solution for synchronization, where it separately encode the synchronization word with a random constructed LDPC matrix, and repeat it, then concatenate it with the encrypted codeword and send it after permutation operation, this method achieves high error correction rate where it based on repeated codes that is decoded separately and decoded together as will be shown in next subsection. It also increases the security, where the addition of random data to the codeword is recommended to increase the security as described in [18]. The Sync Word (SW) consists of system ID, Counter Value (CV), and Random Vector (RV), where RV is also used to mask the ID and CV, so adding it to encrypted codeword doesn't affect the total frame randomness.

The input data block to the algorithm consists of two frames, each frame input data length is 256 bit which is double size of conventional AES as recommended in [21] to resist side channel attacks. The algorithm begins with input initialization which is executed by XOR of IV and SW. Every round of AES-256 contains customized Sbox, conventional MixColumns for 128 bit input, customized permutation represented in 256 bit permutation, and key 256 bit XOR, as shown in Fig. 1. After 3 rounds of AES-256 the AES output is xored with LDPC input. During LDPC coding operation,

the AES-256 is working in parallel to prepare a cipher stream (randomly permuted frame from the AES-256 output after 6 rounds and 9 rounds) to be xored with the 512 bit output of LDPC to obtain the first encrypted codeword W1. The second frame treated like the first frame except for the AES-256 input for the second frame is the output of the AES-256 from the previous frame as represented in Fig. 1 to obtain W2. Repeated SW, W1, and W2, are concatenated together and permuted to obtain the encrypted block. The structure of AES-256 is the same for decryption-decoding process because of using stream cipher. The decryption-decoding process begins with inverse permutation for the received data and then SW decoder is used to decide whether to initialize the operation or not. According to the received data, i.e. if the received data is too noisy or modified the operation is stopped, this method proposes a novel idea to resist fault attacks, and modification attacks, and in the same time not affect the error correction rate in case of noise according to its error correction performance.

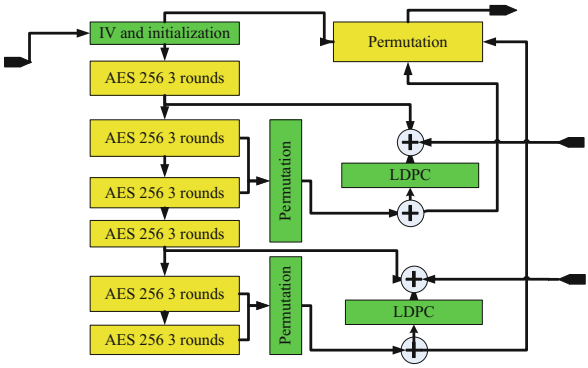

Fig. 1. Data flow for one block of two frames 256 bit length

The LDPC decoding algorithm for the proposed method is consisting of two decoders the first is Scaled Min-Sum (SMS) Algorithm and the second is Weighted Bit Filliping (WBF) algorithm that is used as a post processing decoder to enhance the performance and at higher SNR used as the main decoder to obtain high throughputs. The WBF [22] begins by calculating the hard decision for the received codeword, then calculate syndrome from (6) where Y is the received codeword and H is the parity matrix. If S is null or if the maximum iteration reached, then stop decoding and output X as the decoded data. If S is not null so it uses (7) and (8) to calculate the weight of the error for every variable node and then flip the maximum error e_i bits

$$S = Y \cdot H^T \tag{6}$$

$$w_{ij} = \prod_{i \in 1:n} s_j . H_{ij}, \text{where } j \in 1 : m \tag{7}$$

$$e_i = \sum_{j \in 1:m} w_{ij} \tag{8}$$

SMS algorithm is a massage passing algorithm that is iteratively calculates the effect of variable/check nodes that share certain check/variable nodes through tanner graph to correct error bits, so the algorithm is based on two main functions which are check node update as shown in (9) and variable node update showed in (10). The algorithm stops decoding if reaches to the right code by the help of syndrome calculator or when reaching to the maximum iteration.

$$E_{i,j}^{new} = \alpha \prod_{i'} \text{sign}\left(L_{i,j}\right) \times \min_{i'} |L_{i,j}| \tag{9}$$

$$L_{i,j}^{new} = u_i^{ch} + \sum_{j'} \left(L_{i,j}\right) \tag{10}$$

Sbox customization is done by customizing each internal operation, where the AES substitution process is the inverse of the number in GF(2^8) modulo irreducible polynomial followed by affine transformation and adding constant. The customization for Sbox depends on using a polynomial from the 30 available polynomials, using different constant and different affine matrix. The count of different affine transformation is 19482323021283328000 in the AES system [23, 24]. The generated Sboxs are tested to check its avalanche criteria, strict avalanche criteria, bit independency criteria and correlation as described in [15]. The ShiftRows or permutation is achieved by bit shuffling (bit permutation) to resist Square attacks that based on byte oriented behaviors to extract the key.

4 LDPC-CSAES FPGA Implementation

In this section the hardware implementation and optimization that is achieved in order to reduce the hardware size without affecting the functionality and performance are introduced.

The top-level of decoder-decryption module is shown in Fig. 2. The green modules in Fig. 2 represent the SW/LDPC decoder; the yellow modules represent the AES-256 modules, while the purple modules represent other control modules such as inverse permutation. LDPC-CSAES contains one Round of AES-256 and one LDPC decoder. The AES round has the same structure of conventional AES's round, but the Sbox of the proposed algorithm is customized and a 256 bit permutation is used instead of byte permutation used in conventional AES. The implementation of Sbox is based on Lookup Tables (LUTs), while the MixColumns is implemented by logic functions. The extracted SW is decoded by SMS algorithm through the Variable Node Update (VNU), and Check Node Update (CNU), where CNU, and VNU is generic modules that can process any regular LDPC (column weight = 3, raw weight = 6). So the first hardware reduction is achieved by resources reuse where SW and The other two codewords are

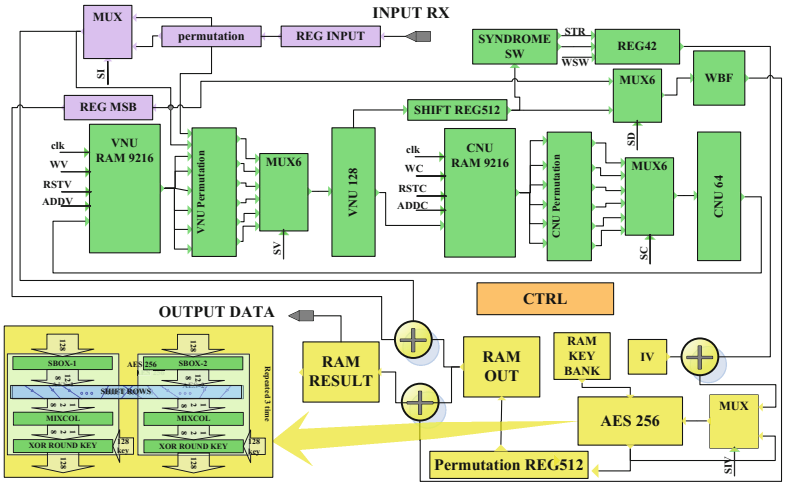

Fig. 2. LDPC-CSAES block diagram

decoded by the same module. The problem is the existence of different mapping according to the Parity matrix H, and this problem is solved by using multiplexers to choose between different mappings according to H.

Figure 3 is simple block diagram of SW decoder to describe the SW decoding operation. It consists of three parallel decoders work in parallel on the repeated SW codeword in the first stage then add the output of the three decoders together and decode it again in the second decoding stage. After decoding if the syndrome gives null it stops decoding and output the SW and initiates the system to start. If the syndrome is not null it saves power and put the system in idle state. The syndrome of WBF decoder is used while SMS iterations execution to allow the decoder to stop decoding early to reduce the computational complexity and reduces hardware size through exploiting WBF Syndrome. If SMS algorithm stops without right decoding the WBF is working on these data as a second decoder until reaching to the maximum number of iteration or reaching to the right code. The SMS one iteration costs 8 clocks 4 clocks for the

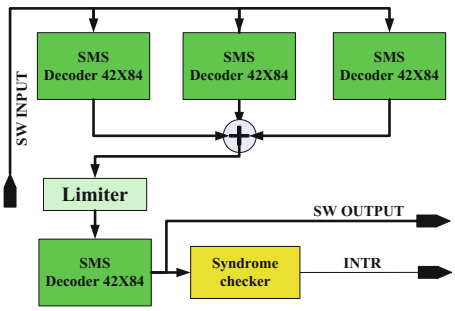

Fig. 3. Sync-word decoder

variable nodes update and 4 clocks for check node update, and SW iteration Costs 2 clocks. The maximum number of iterations required for decoding SW is 5 for the first stage and 5 for the second, while the maximum number of iterations required for SMS and WBF is 10 iterations for each, but actually using of SMS before WBF speeds the convergence rate of WBF and reduces the SMS's average number of iterations. Also using WBF reduces the average required number of iterations for the SMS through early stop decoding feature.

To reduce the required hardware for CNU module and avoid critical paths, a 3-MIN method described in [25] and a new scaling method are used. The new scaling method based on subtracting the value of the most 2 bits from the value itself. This method suits 6 bits quantization; it reduces the hardware size without affecting the performance. The variable node update module size is also reduced as shown in Fig. 4, where it uses only 5 adders instead of 6 adders. The WBF algorithm is implemented by simple logic gats, and it costs only one clock to execute iteration. The resources utilization is represented in Table 1. According to timing diagram in Fig. 5 the proposed algorithm can achieve 603 Mbps throughput and 10^{-6} BER at SNR 3.25 dB, using clock frequency 250 MHz, while it can achieve 2 Gbps and 10^{-6} BER, at SNR > 6 based on using WBF as the main decoder.

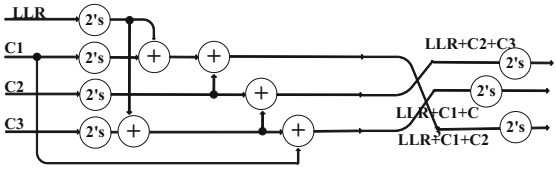

Fig. 4. Variable nodes update circuit diagram

Table 1. Resources utilization for 6 bits quantization for the proposed method

Algorithm		Logic functions	Logic registers	Memory elements
LDPC	CNU	18624	–	13824
	VNU	25216	–	13824
	WBF	4020	512	–
	CTRL	864	810	–
AES		938	1024	69888
Input interface		3840	7424	512
Total (LDPC-CSAES)		53 k (46%)	9 k(8%)	95 k(2%)

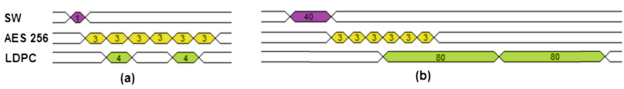

Fig. 5. LDPC_SAES timing diagram, (a) encoding, (b) decoding

5 Comparative Results, Testing, and Resistance to Attacks

5.1 Comparative Results, Testing

There are two points for comparison which are the error correction rate, and the security level. The security level can be represented in the maximum deferential probability, maximum correlation, and resistance to attacks. Although the LDPC used is (512, 256) which is short code length it achieves high error correction rates reaches to 10^{-6} at 3.25 dB, and that is because of using the new method of SW decoding that achieves very high block error correction rate as shown in Fig. 6, stream ciphering, and two decoders (SMS and WBF). The proposed method achieves 0.75 dB coding gain compared to JSALC code length 256 [11], and more than 2.5 dB coding gain compared with other joint AES [8, 20]. The customized Sboxs maximum correlation is 2^{-3} and maximum differential probability is 2^{-6}, where they are examined after generation as discussed in Sect. 4, and only the Sboxs that path the test are selected. The proposed method uses MixColumns from conventional AES so the branch number for 4 rounds of the proposed method equals to the conventional AES which equal 2^{25} [11, 14]. So the maximum differential cryptanalysis is (2^{-150}) which is lower than (2^{-127}) and have complexity greater than $O(2^{128})$. The same is the maximum linear probability is (2^{-75}) which is lower than (2^{-64}) and its complexity is greater than $O(2^{128})$. From another point of view, the bit propagation for the 3 rounds of AES is 4^3 and LDPC propagation is 4^2 [11, 14]. For the proposed algorithm we have two frames, every frame is xored before LDPC and after LDPC, where LDPC code rate is 0.5 so we need 256 cipher bit to XOR the LDPC input and 512 bit to XOR the LDPC output for every frame so 6 cipher outputs are required for one block contains 2 frames. Every 3 rounds of the proposed AES, a cipher output with length 256 can be obtained. So the numbers of rounds for the first frame is 3, 6 and 9 rounds of AES, while the second AES input is the previous AES output, so the numbers of rounds for the second frame is 12, 15 and 18 rounds of AES. So the Total Propagation for the first frame is bounded by $4^3 * 4^2 * 4^6 \leq \text{TP} \leq 4^3 * 4^2 * 4^9 = 4^{11} \leq \text{TP} \leq 4^{14}$, and for the second frame, the TP bounds is $4^{29} \leq \text{TP} \leq 4^{32}$, if we consider the least bounds which are 4^{11} for the first frame and 4^{29} for the second frame, the proposed algorithm still achieves higher

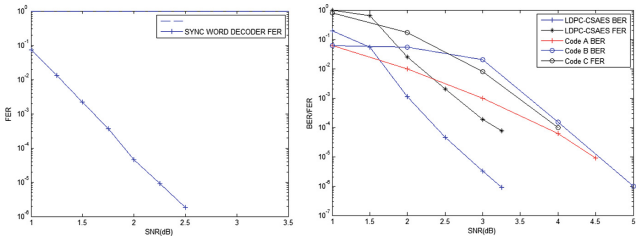

Fig. 6. (a) LDPC-CSAES SW Block error rate against SNR for AWGN channel, BPSK, and (5, 1) Quantization (b) Performance comparison between LDPC-CSAES error correction rate, A: BER 6 round AES-LDPC [8], B: BER 10 round AES-LDPC [20], C FER JSALC code length 256 [11].

propagation than other related works. For more accurate evaluation for the complexity of the proposed algorithm, the NIST tests are used to examine it.

5.2 LDPC-CSAES Resistance to Attacks

If the attacker tries to use brute force to attack the proposed algorithm so, he has to try invisible number of trials (2^{256} key * 2^{256} I_V * 256! internal permutation * 512! intermediate permutation * 1276! final permutation 30 polynomials-Sbox * 20922789888000 affine transformations * ($C_6^{256} * C_3^{512}$) random LDPC matrix). Which is the highest value compared to others secure channel coding algorithms. The proposed method is also immune against other attacks like differential and linear attacks, where as discussed before it has complexity greater than $O(2^{128})$. Bit shuffling and LDPC make the square attacks which based on byte orientation useless [8]. In side channel attacks the attacker always has a hypothetical model of the encryption algorithm, especially for the last and first round. Unlike other related works the proposed algorithm based on CTR mode integrated with LDPC, and uses customized parameters, so that the attacker has no hypothetical model for it and the attacker cannot attack the last round because of LDPC [17].

The parallel processing of LDPC and AES, beside the double structure make the algorithm very immune for side channel attacks, including fault attacks, especially in the presence of the SW decoding that controls the algorithm and stop it, if modification exist in the received codeword

6 Conclusion

The proposed method introduces a solution that gathers between high level of security, high error correction capability, high throughput, and low complexity for a practical secure channel coding based on the integration between AES and LDPC. In this paper a LDPC is exploited to increasing the security level and resist attacks without any degradation of its performance. The proposed method introduces a solution for stream cipher synchronization that is exploited to make the algorithm immune for modification attacks. It also resists side channel attacks. The throughput achieved at 3.25 dB is 604 Mbps with low hardware size.

References

1. McEliece, R.J.: A public-key cryptosystem based on algebraic coding theory. DSN progress report (1978)
2. Baldi, M., Chiaraluce, F.: LDPC codes in the McEliece cryptosystem. In: IEEE International Symposium on Information Theory (2007)
3. Baldi, M., Chiaraluce, F.: Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC Codes. In: ISIT 2007, pp. 2591–2595 (2007)
4. Shooshtari, M.K., Ahmadian, M.: Improving the security of McEliece-like public key cryptosystem based on LDPC codes. In: ICACT 2009, pp. 1050–1053 (2009)

5. Xu, C., Chang, Y.: Encryption scheme of physical layer based on irregular LDPC codes. In: Proceedings of AIAI 2010 (2010)
6. Stuart, C.M., Deepthi, P.P.: Hardware efficient scheme for generating error vector to enhance the performance of secure channel code. In: IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES), pp. 1–5 (2015)
7. Ayeh, E., Varanasi, M., Adamo, O.: Joint Encryption error correction and modulation (JEEM) scheme. In: 2012 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR), pp. 1–7 (2012)
8. Xiao, Y., Su, Q.: Design of LDPC-based error correcting cipher. In: IET 2nd International Conference on Wireless, Mobile and Multimedia Networks (ICWMMN) Proceedings, pp. 470–474 (2008)
9. Gupta, C.P., Gautam, S.: Joint AES encryption and LDPC coding. Int. J. Sci. Eng. Res. (IJSER) **4**(7), 603–606 (2013)
10. Lin, K., Lin, W., Deng, Z., Li, N.: A joint encryption and error correction method used in satellite communications. China Commun. J. **11**(3), 70–79 (2014)
11. Abu-Surra, S., Taori, R., Pisek, E.: Enhanced cryptcoding: joint security and advanced dual-step quasi-cyclic LDPC coding. In: IEEE Global Communications Conference (GLOBECOM), pp. 1–7 (2015)
12. Federal Information: Announcing the Advanced Encryption Standard (AES), 26 November 2001
13. Lamberger, M., Pramstaller, N., Rijmen, V., Vercauteren, F., Daemen, J.: Computational aspects of the expected differential probability of 4-round AES and AES-like ciphers. IEEE Computing 2009, pp. 85–104 (2009)
14. Rijmen, V., Daemen, J.: New criteria for linear maps in AES-like ciphers. IEEE Cryptogr. Commun. **1**(1), 47–69 (2008)
15. Mazumdar, B., Mukhopadhyay, D.: Design for security of block cipher S-Boxes to resist differential power attacks. In: International Conference on VLSI Design, pp. 113–118 (2012)
16. Scripcariu, L.: A study of methods used to improve encryption algorithms robustness. In: 2015 International Symposium on Signals, Circuits and Systems (ISSCS), pp. 1–4 (2015)
17. Ragel, R., Ambrose, J.A., Ignjatovic, A., Parameswaran, S., Jayasinghe, D.: Advanced modes in AES: are they safe from power analysis based side channel attacks? In: 2014 IEEE 32nd International Conference on Computer Design (ICCD), pp. 173–180 (2014)
18. Esmaeili, M., Gulliver, T.A.: A secure code based cryptosystem via random insertions, deletions, and errors. IEEE Commun. Lett. **20**(5), 870–873 (2016)
19. Richmond, T., Drutarovsk, M., Petrvalsky, M.: Countermeasure against the SPA attack on an embedded McEliece cryptosystem. In: 2015 25th International Conference Radioelektronika (RADIOELEKTRONIKA), pp. 462–466 (2015)
20. Viswanath, K., Pearlsy, P.V.: Cryptocoding system based on AES and concatenated coding scheme involving BCH and QC-LDPC. In: 2015 International Conference on Applied and Theoretical Computing and Communication Technology (ICATCCT), vol. 15, pp. 189–194 (2015)
21. Khan, A.K., Mahanta, H.J.: Side channel attacks and their mitigation techniques. In: 2014 First International Conference on Automation, Control, Energy and Systems (ACES), pp. 1–4 (2014)
22. Lin, S., Fossorier, M.P.C., Kou, Y.: Low-density parity check codes based on finite geometries: a rediscovery and more. IEEE Trans. Inform. Theory, 2711–2736 (2001)
23. Jing, M.-H., Chen, J.-H., Chen, Z.-H.: Diversified mixcolumn transformation of AES. In: 2007 6th International Conference on Information, Communications & Signal Processing, pp. 1–3 (2007)

24. Chen, Z.-H., Chen, J.-H., Chen, Y.-H., Jing, M.-H.: Reconfigurable system for high-speed and diversified AES using FPGA. Microprocess. Microsyst. **31**(2), 94–102 (2007)
25. Aziz, S.M., Pham, D.M.: An automated design methodology for FPGA-based multi-Gbps LDPC decoders. In: 2012 15th International Conference on Computer and Information Technology (ICCIT), pp. 495–499 (2012)