ISSN 2278-3091

Volume 9, No.5, September - October 2020

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse293952020.pdf

https://doi.org/10.30534/ijatcse/2020/293952020

Study and Analysis of BTED Error Correction Codes for Cryptography Applications

Ramesha M [1], Jeevan K M [2], Dankan Gowda V [3], Bharathi Gururaj [4], Sridhara S B [5]

[1] Assistant Professor, GITAM School of Technology, GITAM University, Bengaluru, India,
rameshmalur037@gmail.com

[2] Assistant Professor, GITAM School of Technology, GITAM University, Bengaluru, India,
jeevanjeevan77@gmail.com

[3] Assistant Professor, B.M.S. Institute of Technology and Management, Bengaluru, India, dankan.v@bmsit.in

[4] Assistant Professor, ACS College of Engineering, Bengaluru, India, bharathigururaj@gmail.com

[5] Professor, Vijaya Vittala Institute of Technology, Bengaluru, India, sridharasb1947@gmail.com

ABSTRACT

In recent days securing of transmitted data is an essential part of any communication system. There are several approaches with respect secure data like the utilization of cryptographic algorithms and other techniques. In this paper, Bit Transition Encoder and Decoder (BTED) error correction codes are analyzed for securing user data with the utilization of error correction codes (ECC) operations, such as point addition, point addition, point doubling, and point negative operation. Bit Transition Encoder and Decoder (BTED) error correction codes better than conventional error correction codes with several aspects such as area overhead in hardware, decoding time, and efficiency. The proposed techniques can be effectively utilized for memory applications. The error-correcting capability of these codes is also included for efficient implementation. This paper analyses the BTED Error Correction Codes for Cryptography Applications and compared the error correction capability of various error correction codes.

Key words: Cryptography, Error Correction codes, Data Security, Encoding, Decoding

1. INTRODUCTION

Encoding and decoding of data play a vital role in any wireless communication system. There are several encoding mechanisms are utilizing for the encoding of the data. The encoding process imposes redundant bits to secure the data in several formats. The imposing of redundant bits creates ambiguity for the decryption of data and also the possibility of error in the received data. By using a suitable error correction algorithm the ambiguity of recovering data can be minimized with the help of error correction codes operations. The commonly used error correction operations are point addition, point doubling, and point negative operation. In point addition operation, the bits adding a point along an elliptic curve to itself repeatedly to obtain the desired results. Certain cryptographic algorithms are implemented based on the point addition mechanism. Point doubling is the operations specified for points of the elliptic curve to obtain accurate results from the encoded data. It should be the same as if we wanted to sum not two distinct but rather two equal points. Point negative operation is performed on the bits to accommodate a particular interval for the processing bit [1]. Figure 1 shows a generalized block diagram of encryption and decryption. As an instance, the input to encryption block is considered as plain text and key. The key is generated by the use of the data set. The data set consists of binary digits that pertain to the input but modified through by performing point addition, point addition, point doubling, and point negative operation. The received encrypted data is decrypted in the decrypted block. The decryption block consists of cipher text transferred data and key that is generated from the error correction operations. The once the algorithm extracts the plain text it will be compared with the original data for authentication purposes.

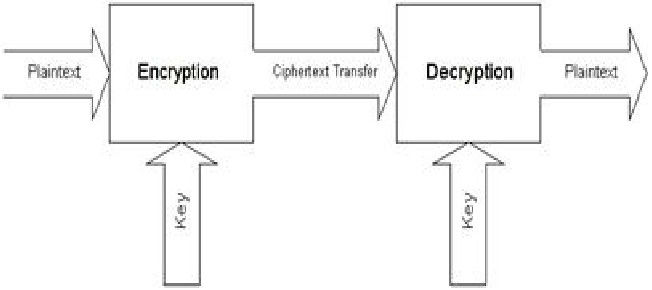


Figure1: Block diagram of encryption and Decryption

The overall organization of the paper as follows. In the second section, the ECC operations are explained with mathematical expressions and their corresponding curves. In the third section, Scalar multiplications of points are discussed. In the fourth section, the BTED algorithm explained, and finally, the results and conclusion of the paper is described.

2. ECC POINT OPERATIONS

In this section, the computations of curve points for encryption and decryption operations are analyzed in detail and also how to compute the points for a negative number is discussed.

A.Point Addition:

In point addition operation, the bits adding a point along an elliptic curve to itself repeatedly to obtain the desired results [2]. Given 2 points on an elliptic curve, $J(x_1, y_1)$ and $K(x_2, y_2)$, then the addition of these points results in $L(x_3, y_3)$, which lies on the curve as depicted in Figure 2.

$$\lambda = [(y_2-y_1)/(x_2-x_1)] \pmod{p} \quad (1)$$

$$x_3= [\lambda-x_1-x_2] \pmod{p} \quad (2)$$

$$y_3= [\lambda(x_1-x_3)-y_2] \pmod{p} \quad (3)$$

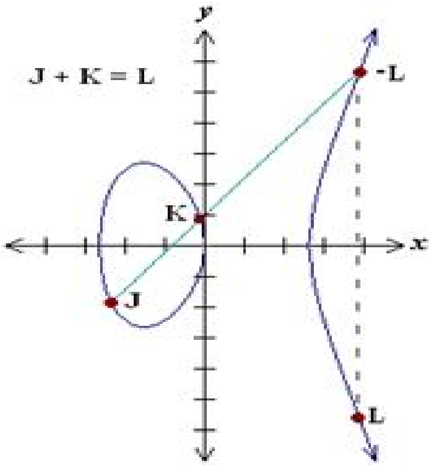


Figure.2: Point addition operation on elliptic curve.

B.Point Doubling

Point doubling is the operations specified for points of the elliptic curve to obtain accurate results from the encoded data. Given a point $J(x_1, y_1)$ on an elliptic curve, point doubling i.e., $J(x_1, y_1) + J(x_1, y_1)$ yields $L(x_3, y_3)$, which lies on that curve as outlined in Figure.3.

$$\lambda = [(3x_1^2 + a) / (2y_1)] \pmod{p} \quad (4)$$

$$x_3=[\lambda-2x_1](\text{mod } p) \quad (5)$$

$$y_3= [\lambda(x_1-x_3)-y_2] \pmod{p} \quad (6)$$

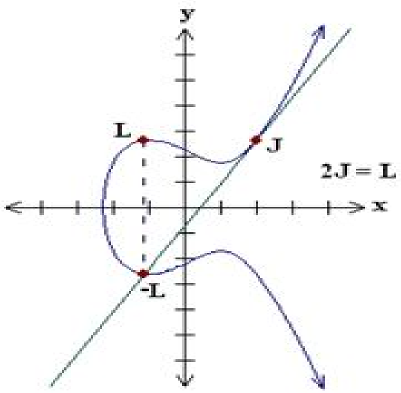


Figure.3: Point doubling operation on elliptic curve

C.Point Negative

Point negative operation is performed on the bits to accommodate a particular interval for the processing bit. Given a point $J(x_1, y_1)$ on an elliptic curve, to find $-J(x_1, y_1)$ is given by $J(x_1, p- y_1)$, as illustrated in Figure.4.

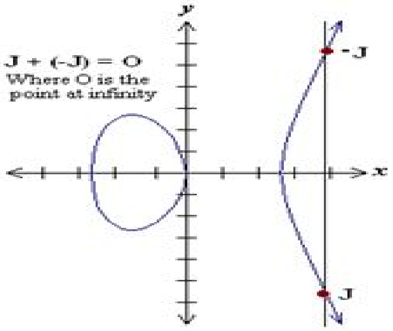


Figure.4: Point Negative operation on elliptic curve

3. SCALAR MULTIPLICATION OF POINTS

Given a point P(x1, y1) on an elliptic curve, to calculate Q(x2, y2) = k P(x1, y1), where k is any number, it requires frequent point additions and point doublings. The calculated points are stored in read-only memory for write and read operation. The volume of each memory location is 32 bits, and the total number of memory locations is 256 [3]. All 256 points are in the form of 16x16 matrixes, and each point size is 32 bits. The input information is replaced with the S-box value deployed on x and y coordinate value, as depicted in the below instance. Assume the input data is 4512, the primary digit in x-coordinate i.e., 4 and the last number in y-coordinate i.e., 2. The value 4512 is substituted by the fourth column and second-row values from S-box. The outcome of the S-box value is encrypted; employing BTED and encrypted data is broadcasted through wireless communication [4]. The received information at the receiver is decrypted by utilizing the syndrome calculator and error detector circuit. The S-box value is plotted as points on an elliptic curve. Depending on the input values, the S-Box values are substituted for subsequent encryption and decoding using BTED [5]. The figure 5 shows the hierarchical demonstration of elliptic-curve cryptography (ECC).

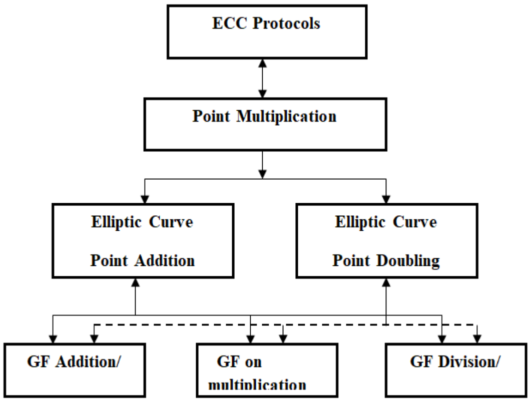


Figure.5: The hierarchical demonstration of elliptic-curve cryptography (ECC)

4. BTED ENCRYPTION AND DECRYPTION

The decoding is necessary to identify and correct errors in the word obtained from encryption output. Assume that original data bits are B3, B2, B1, and B0 and the redundant bits are C0 and C1. The bits C0 and C1 are obtained via the binary formula (XOR operation).

$C0 = B0$ xor $B2$ = 1 xor0 =1

$C1 = B1$ xor $B3$ = 0 xor1 = 1

Then assume now that Multiple Bit Upsets (MCUs) occur in bits $B3$, $B2$, and $B0$, the received redundant bits $C0$ and $C1$ are computed.

$$C_0^{'} = B_0^{'} xor B_2^{'} = 0\ xor\ 1 = 1$$

$$C_1^{'} = B_1^{'} xor B_3^{'} = 0\ xor\ 0 = 0$$

In order to detect these errors, the syndrome bits $S0$ and $S1$ are obtained as follows.

$$S_0 = C_0^{'} xor C_0 = 1 xor\ 1 = 0$$

$$S_1 = C_1^{'} xor C_1 = 0\ xor\ 1 = 1$$

These results suggest that the error bits B2 and B0 are wrongly observed as the initial bits thus, these 2 errors bits not corrected. This instance describes that for a direct paired task, the number of even bit errors cannot be recognized. In the beginning, from data bits 'D' the obtained repetitive bits H4, H3, H2, H1, H0, and V01-V31 are formed. Using equation 13 and 14, the level condition bits H4, H3, H2, H1, H0 and the vertical error bits S3-S0 are assumed as follows:

$$\Delta H_4 H_3 H_2 H_1 H_0^1 = H_4 H_3 H_2 H_1 H_0^1 - H_4 H_3 H_2 H_1 H_0 \qquad (7)$$

$$S_3 - S_0 = V_0^1 \oplus V_0 \qquad (8)$$

Similar computations are executed for the rest of the vertical error bits. Here "–" represents the decimal integer subtraction. When S3 – S0 are equated to zero, then the stored on code word has only original data bits in the symbols, and there are no errors. When $\Delta H_4 H_3 H_2 H_1 H_0^1$ and S3 – S0are nonzero, then there is an error. Induced errors are identified in symbol 0. These errors are corrected by using the below equation (9).

$$D_{0correct} = D_0 \oplus S_0 \qquad (9)$$

Earlier it was determined by considering error location; a parallel calculation can be used to differentiate a limited number of errors. In any instance, when these decimal computations are employed to distinguish errors, these errors can be determined in such a way that unraveling error is kept aside. The positioning technique of decimal error detection using the proposed structure is specified in the above equations and also shown in decrypted data. Initially, the extra horizontal bits H4H3H2H1H0 are obtained from the primary data bits as follows.

$$H_4 H_3 H_2 H_1 H_0 = D_3 D_2 D_1 D_0 + D_{11} D_{10} D_9 D_8 = 1010 + 1100 = 10110$$

$$H_4 H_3 H_2 H_1 H_0^{'} - D_3 D_2 D_1 D_0 + D_{11} D_{10} D_9 D_{10}$$
$$= 0111 + 1111 = 10110$$

Then, the flat syndrome bits $H4H3H2H1H0$ can be obtained using decimal integer subtraction as follows.

$$\Delta H_4 H_3 H_2 H_1 H_0 = H_4 H_3 H_2 H_1 H_0^{'} - H_4 H_3 H_2 H_1 H_0$$
$$= 10110 - 10010 = 00100$$

The decimal approximation of H4H3H2H1H0 is not equal to zero, and the errors are known and represented as symbol 0 or symbol 2. Thus, these flopped bits are placed in a precise area to utilize the vertical disorder bits S3 − S0 or S11 − S8. Thus, in decimal computation, the proposed system has higher flexibility for ensuring memory against M. Consequently, this is feasible for single and decimal slips and also for various types of errors per line may be studied for the proposed strategy irrespective of errors. In decimal one of the basics for these bits, H4H3H2H1H0 is zero. The 7-bit slips occur in symbol 0 and 2 simultaneously; the disentangling error can be declined in the sequence 1, 2, and 3, and are essential properties of the BTED algorithm. In this algorithm, all single-error and multiple errors are resolved in two sequential images. Phenomena of sequence 4 and 5 conferred are redressed, and various errors per column are identified by the even error bits.

Thus exhibit memory recollections from substantial Multiple Bit Upsets (MCUs). Then again, impacts of type 4 and 5, is important to identify this decoding slip based on these essential variables are achieved [6]. Data bits in symbols with decimal integer 0 and 2 are equated 2m – 1 and therefore the error computation in symbol 0 and 2 equal to 2m – 1 and the error computation in symbol 0 and 2. E.g., when m = 4, then the probability of the cryptography errors is shown in the below equation.

$$X_{\Delta H=0} = 4x \left(\frac{1}{2^4}\right)^2 xX_{BWMC} \approx 0.001 \qquad (10)$$

Similarly, If m=8, then the probability of the cryptography errors is obtained in using the below equation.

$$X_{\Delta H=0} = 4x \left(\frac{1}{2^8}\right)^2 xX_{BWMC} \approx 0.0000011 \qquad (11)$$

The plot of eight errors in data is shown in Figure.6. From this figure, it is assumed that memory as a regime that contains few errors, and in the middle of these errors, there are no more than 3 bits. Thus it is not a major problem. In a comparative method the errors in symbol1 and symbol 2 are differentiated and rectified [7].

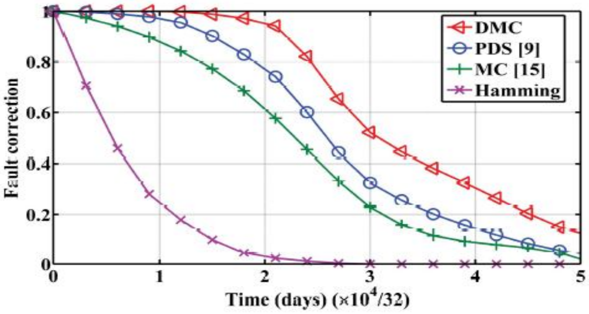


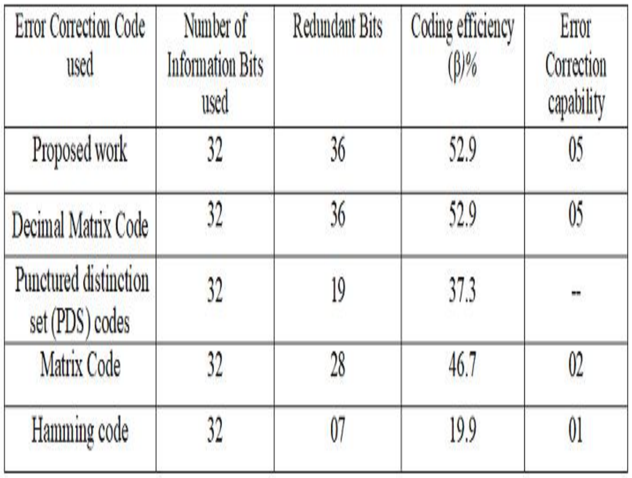
Figure.6: Time Difference security code plot for eight errors in data

5. RESULTS AND DISCUSSION

The total power of the Passive tag design is about 5mW, which incorporates a Digital baseband processor, memory, and peripheral devices. The area is smaller because the BTED method is included in the design. In program design, target specifications like input and output are considered. After software design, the Verilog HDL code is compiled and synthesized to verify the design syntax. Once the compilation and synthesis are completed, the Verilog HDL code is simulated to study the performance. The simulation was executed using Simulink software. The entire design is copied on to the FPGA Artix-7 FPGA Kit to examine its practicality once the simulation is completed.

There are several other parameters are taken to considerations for comparison purposes namely, total power, delay, and device utilization summary are indicated in Table 1 for different ECC's. This proposed research work has satisfactory performance in terms of power and speed compared to other ECC's.

Table 1: Error Correction capability BTED Technique



Error Correction Code used	Number of Information Bits used	Redundant Bits	Coding efficiency (β)%	Error Correction capability
Proposed work	32	36	52.9	05
Decimal Matrix Code	32	36	52.9	05
Punctured distinction set (PDS) codes	32	19	37.3	--
Matrix Code	32	28	46.7	02
Hamming code	32	07	19.9	01

6. CONCLUSION

Error rectification codes are employed to boost memory assurance and make the memory error-free. Various ECC's are employed to differentiate the event of an error and also corrected and the distinguished ones. On the other hand, the error detection ability and, therefore, the overheads differ given the codes used. The projected BTED system accomplishes recognition and correcting the errors effectively using the bitwise matrix code algorithm technique. Comparing with earlier works, the improved BTED can correct errors up to 5 bits in a symbol. The projected system demonstrates that it has a higher protection level compared to the large multiple cell upsets in the memory cells.

REFERENCES

1. S.Raza, **Secure communication for the Internet of Things - a comparison of link-layer security and IP sec for 6LoWPAN**, *Security and Communication Networks*, vol. 7, no. 12, pp. 2654–2668.
2. O. Srinivasa Rao, **Efficient Mapping Methods for Elliptic Curve Cryptosystems**, *International Journal of Engineering Science and Technology*, Vol. 2(8), 2010, ISSN: 0975-5462, 3651-3656.
3. Sanghyeon Baeg, Pedro Reviriego, Juan Antonio Maestro, Shijie and Richard Wong (2011), **Analysis of a Multiple Cell Upset Failure Model for Memories**. *ACM Transactions on Design Automation of Electronic Systems*, Vol. 16, No. 4, Article 45, 1084-4309.
4. B, Sridhara & M, Ramesha & Patil, Veeresh. **Adaptive Scheduling Design for Time Slotted Channel Hopping Enabled Mobile Adhoc Network.** *International Journal of Advanced Computer Science and Applications*.11. 10.14569/IJACSA.2020.0110333.
5. M. Ramesha, S. B. Sridhara, A. B. Anne Gowda, N. Anughna, and B. Gururaj, **Design and development of low power BTED cryptography algorithm on FPGA,** *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 4, pp. 4359–4362, 2020, doi: 10.30534/ijatcse/2020/27942020.
6. V. D. Gowda, S. B. Sridhara, K. B. Naveen, M. Ramesha, and G. N. Pai, **Internet of things: Internet revolution, impact, technology road map and features**, *Advances in Mathematics: Scientific Journal*, vol. 9, no. 7, pp. 4405–4414, 2020, doi: 10.37418/amsj.9.7.11.
7. Ramesha M, **FPGA Implementation of Low Power High Speed BTED Algorithm for 8 Bit Error Correction in Cryptography System**, *International Journal of Emerging Trends in Engineering Research*, vol. 8, no. 7, pp. 3893–3897, 2020, doi: 10.30534/ijeter/2020/158872020.
8. V. Dankan Gowda, A. C. Ramachandra, M. N. Thippeswamy, C. Pandurangappa, and P. Ramesh Naidu, **Modelling and performance evaluation of**

anti-lock braking system, *J. Eng. Sci. Technol.*, vol. 14, no. 5, pp. 3028–3045, 2019.

9. Dankan. V. Kishore, D. V. Gowda, Shivashankar, and S. Mehta, **MANET topology for disaster management using wireless sensor network**, in *International Conference on Communication and Signal Processing, ICCSP 2016*, 2016, pp. 0736–0740, doi: 10.1109/ICCSP.2016.7754242.