# Novel Cryptographic Algorithm for 4G / LTE-A

Eman Ashraf Mohammed
Dept. of comm. and elect.
Faculty of engineering , Mansoura university, Egypt

Nihal F. F. Areed
Dept. of comm. and elect.
Faculty of engineering , Mansoura university,Egypt

Ali Takieldeen
IEEE Senior Member,
Delta University,Egypt

Mohammed abd-elazeem
Dept. of comm. and elect.
Faculty of engineering , Mansoura university,Egypt

## ABSTRACT
LTE (Long Term Evolution) is the fourth generation of mobile communication systems, Novel cryptographic algorithm based on enhancement at RC6 algorithm presented here. Using two parallel RC6 at connect them together with some functions in order to use 256 bit input block instead of 128 bit input block as the usual RC6 . This proposed method makes the encryption decryption process faster, stronger, and more secure against attacks. The algorithm applied to many types of data files, shown in MATLAB how it works. The results are compared with EEA2 (EPS Encryption Algorithm 2) used in LTE as EPS stands for Evolved Packet System.

## Keywords
LTE; encryption; AES; RC6

## 1. INTRODUCTION
4G LTE/LTE-A networks has three sets of cryptographic algorithms [1]:

- First set is EEA1/EIA1, which is based on, SNOW 3G algorithm.
- Second set is EEA2/EIA2, which is based on AES algorithm at counter mode [2] figure 1.
- Third set is EEA3/EIA3, which is based on ZUC algorithm.

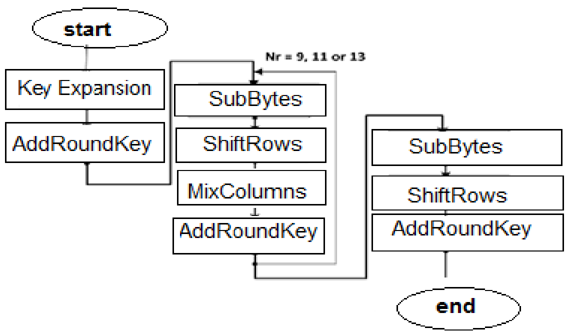



**Fig. 1 AES Algorithm**

## 2. NOVEL ALGORITHM
The proposed version of RC6 is a block cipher with 256-bit block size, 128-bit key size and 20 rounds. RC6 separate 128 bit input to 4-words (W) W=32bit [3], while proposed one works with 8-word input and output, Figure 4.

### A. Key Expansion

The key Expansion of proposed algorithm is exactly same as the RC6 where w=32, r=20 and b=16, figure 2 [4].

**Input**

b Byte key that is preloaded into c word array L[0,1,......c-1]

r denotes the no of rounds.

**Output**
(2r+ 4) W- Bit round keys S [0, 1, ... , 2r+3].

The proposed algorithm is used at counter mode as shown in Figure 3. A counter stream of 256 bit is used as input to the algorithm as shown in Fig 7 with the encryption key (K), producing a key stream of 256 bit, Xoring this key stream with the plain text after divide it to 265 bit blocks in order two produce the cipher text.

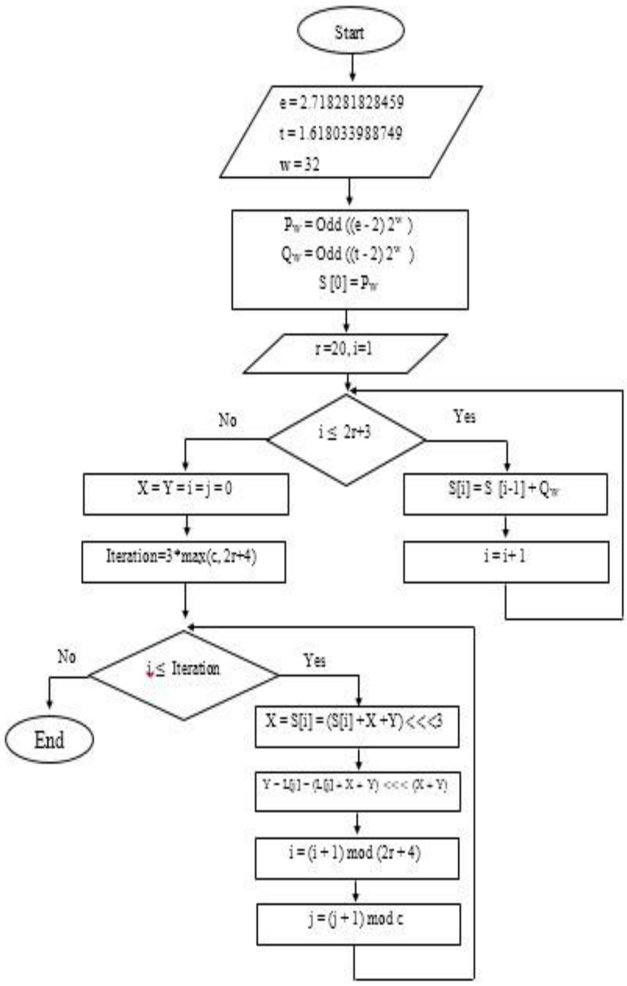



**Fig. 2 Flow Chart of Key Expansion**

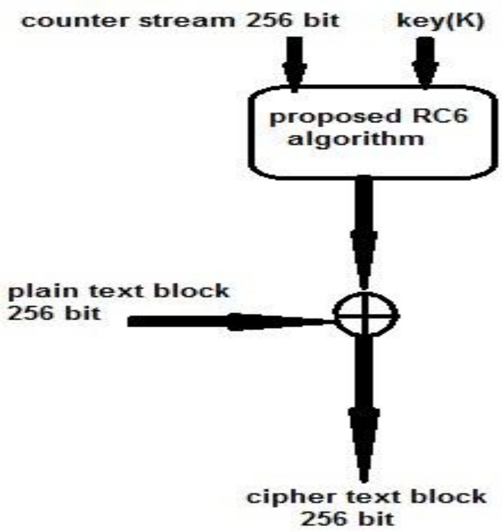



**Fig. 3 Proposed Algorithm at Counter Mode**

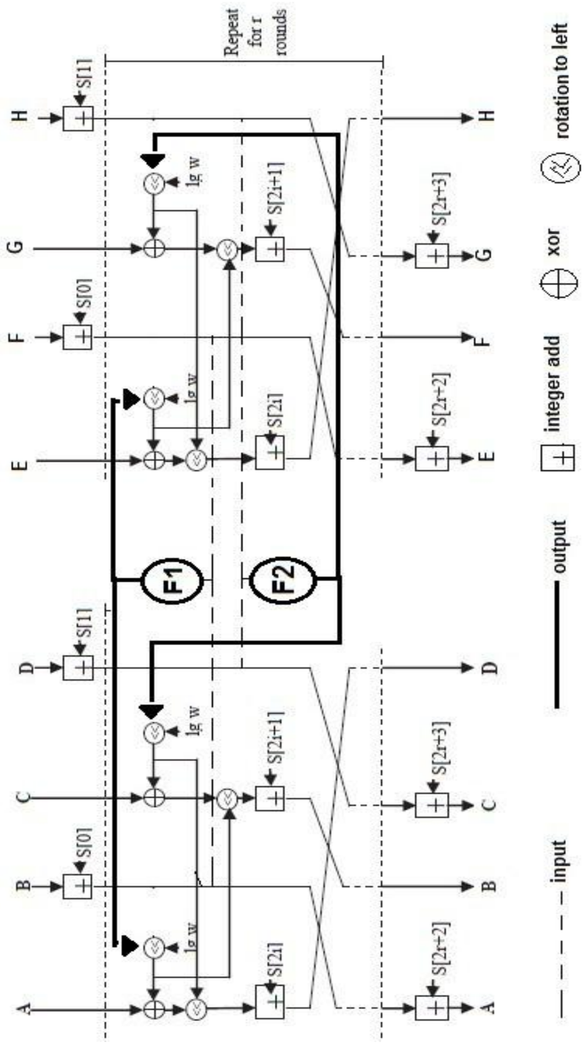



**Fig. 4 Proposed System**

Where: F1=$B^2+F^2$-BF-7 , F2= $D^2+H^2$-DH-7

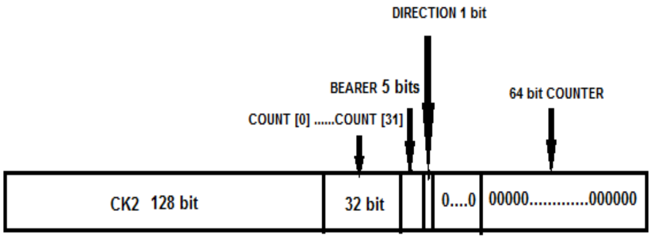



**Fig. 5 Counter Stream**

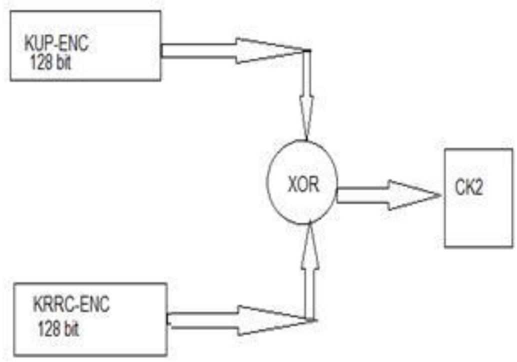



**Fig. 6 Generation of CK2**

- COUNT-C: Frame dependent input used to synchronize the sender and the receiver.
- BEARER: Service bearer identity.
- LENGTH: Number of bits to be encrypted decrypted.
- KRRCenc: key is used to encrypt RRC signaling traffic.
- KUPenc: key is used to encrypt UP traffic. [5][6].

**Table1: Parametric Comparison**

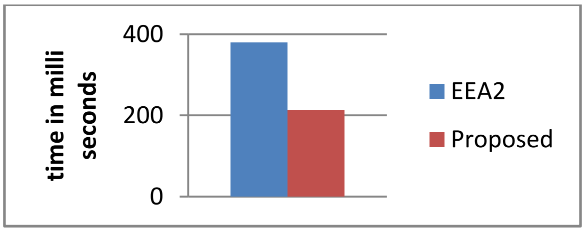
| Parameters | Novel | EEA2 |
|---|---|---|
| **Key Length** | 128 bit | 128 bit |
| **No. of rounds** | 20 | 10 |
| **No of rounds in key** | 2r+4 | r+1 |
| **Block size in words** | 8W | 4W |
| **W (word size in bits)** | 32 | 32 |
| **Block size in bits** | 256 | 128 |
| **Used function** | F1=$B^2+F^2$-BF-7<br>F2= $D^2+H^2$-DH-7 | Sub byte<br>Shift row<br>Mix column<br>Add round key |
| **Used operations** | +,-, ⊗ ,‹‹‹ , ›››, * | ⊗ ,‹‹‹ , ››› |

**Table 2: The Total Execution Time for Different Files**

| FILE NAME | FILE SIZE (KB) | Total Execution Time in Milliseconds | |
|---|---|---|---|
| | | *EEA2* | *Proposed System* |
| A.txt | **48** | 42 | 19 |
| B.txt | **58.61** | 60 | 36 |
| C.txt | **99** | 94 | 55 |
| D.png | **246.51** | 117 | 73 |
| E.png | **320.62** | 169 | 105 |
| F.jpg | **693** | 215 | 119 |
| G.jpg | **898.12** | 263 | 158 |
| H.mp3 | **962** | 214 | 122 |
| I.mp3 | **5344.27** | 1243 | 692 |
| J.mp4 | **7309.33** | 1372 | 753 |
| **Throughput(MB/S)** | | 4.285 | 7.712 |
| **Average execution time** | | 378.9 | 213.2 |

Algorithm has internal parallelism, so, implementations of it should show decrease of execution time and increased Throughput as shown in table 2.

**Chart 1: Average Execution Time for Each Algorithm**




The throughput can be calculated by the following equation [7]:

$$\text{Throughput of encryption} = \frac{len\ (bytes)}{Execution\ \ time\ \ (seconds)} \quad \text{Eq. 2}$$

## 3. SECURITY ANALYSIS

***Key Space Analysis***

There is two tests to show the key space analysis against

Brute-force attacks infeasible summarized as follows [8]:

**a) *Exhaustive Key Search***

For good encryption algorithm, the key space should be large enough to make brute-force attack infeasible [8], where k is the key size in bits.

$$\frac{2^{\wedge}128}{3000*(10^{\wedge}6)*60*60*24*365}$$

$$\approx 3.59676*(10^{\wedge}21)\text{years}$$

This is a very long time, and makes the algorithm resistant to this type of attack.

**b) Key Sensitivity Test**

When there is a slight different in the encryption key the cipher data cannot be decrypted. So, the system is resistant to brute-force attacks to some extent [8].

1. An original image is encrypted by using the secret key K={'0a"f1"8b"d6"d9"b0"8b"08"32"4e"77"6b"d8"d1" 81"77'};
2. K'={'1a"f1"8b"d6"d9"b0"8b"08"32"4e"77"6b"d8"d1" 81"77'}(change is made in the most significant digit in the secret key)
3. K'={'0a"f1"8b"d6"d9"b0"8b"08"32"4e"77"6b"d8"d1" 81"74'}(change is made in the least significant digit in the secret key) and the resultant image is referred to as encrypted image C1,C2,C3 respectively. The results are compared using Eq.1as follows:

$$\text{The Correlation Coefficient} = \frac{cov\ (x,y)}{\sigma x\ \ \sigma y}$$

$$= \frac{\sum_{i=1}^{N}(xi - E(x))(yi - E(y))}{\sqrt{(\sum_{i=1}^{N} xi - E(x))^{\wedge}2}\sqrt{(\sum_{i=1}^{N} yi - E(y))^{\wedge}2}} \quad \text{Eq. 2}$$

$$\text{Where,} \quad E(x) = \frac{1}{N}\sum_{1}^{N} xi \quad \text{Eq. 3}$$

Where x and y are the values of corresponding pixels in the two encrypted images to be compared.

**Table 3: Correlation Results**

| Correlation between | Correlation coefficient |
|---|---|
| Image C1 ,Image C2 | 0.0129 |
| Image C1 ,Image C3 | 0.0156 |
| Image C2 ,Image C3 | 0.0123 |

From the results, there is no correlation between C1, C2, C3 although there is a small change in the encryption key.

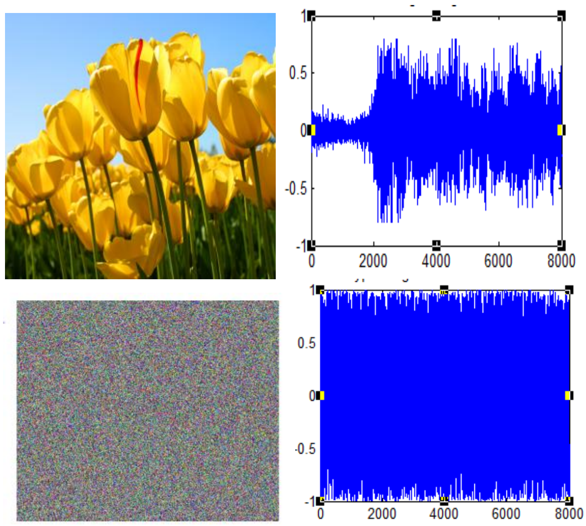



**Fig.7 Original image, original audio, encrypted image and encrypted audio respectively**

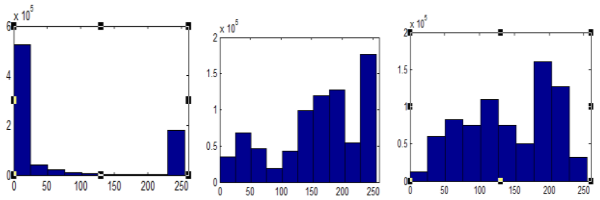


**Fig.8 Histogram for original image: Red, Green, Blue**

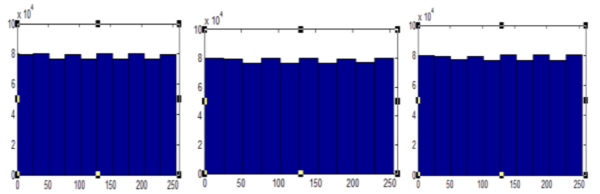


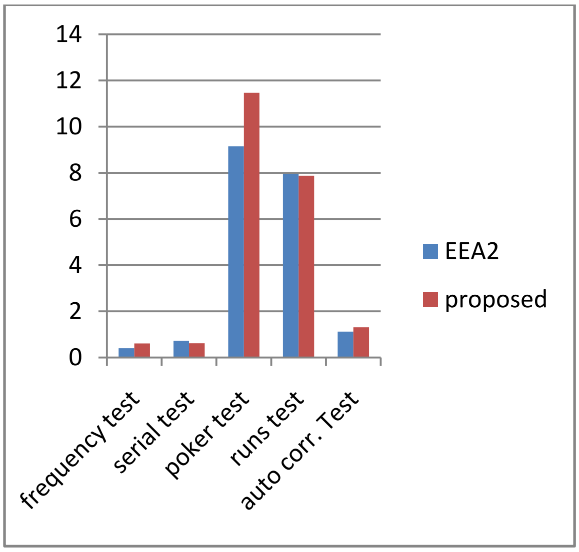
**Fig.9 Histogram for proposed image: Red, Green, Blue**

## 4. STATISTICAL ANALYSIS.

### a) NIST tests (15 test)

Frequency test (monobit test), Serial test (two-bit test), Poker test, Runs test, Autocorrelation test. The threshold values for five tests are 3.8415, 5.9915, 14.0671, 9.4877, and 1.96, respectively[9], so from results it is obvious that the binary sequence is truly random

**Chart 2: The Five Statistical Tests**




### b) Testing histograms

For an audio file handle.wav, it is obtained the next results.

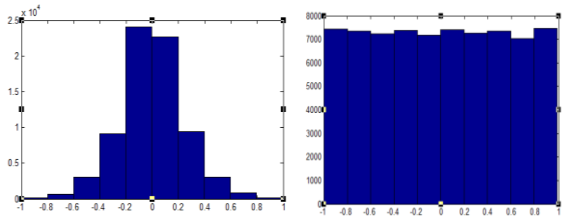


**Fig.10 Original audio histogram and histogram of proposed system encrypted signal respectivly.**

### c) The Histogram Uniformity

This is shown by a test on the histograms of the encrypted data as shown in figures 7, 8, 9, 10.It is clear that the histograms of the cipher data are fairly uniform and significantly different from the respective histograms of the plain data so it is strong against statistical attacks on the proposed algorithm.

## 5. DIFFERENTIAL ANALYSIS

### a) The Avalanche Effect Measuring Factor

There must be significant change in the output of any cryptographic algorithm when the input (plaintext or key) changed slightly. The change of about 50% makes the algorithm truly random [10]. Repeating the previous process for several combinations of Plaintext-key (10). Averaging the results over all different plaintext-key combinations. This differential attack would become very inefficient and practically useless [11].

**Table 4: Avalanche Effect**

| Algorithm | Avalanche effect | |
|---|---|---|
| | 1 bit change in plaintext | 1 bit change in cipher key |
| **EEA2** | 95% | 51% |
| **Proposed** | 98% | 53% |

## 6. BIT ERROR RATE CALCULATION

The bit error rate curve indicates the quality of the encryption system. To get the bit error rate we used QPSK modulation and sent the data over AWGN channel to add some noise to the encrypted data as shown in figure 11. The result is obtained in figure 12. It is obvious that our system reduces BER [12].

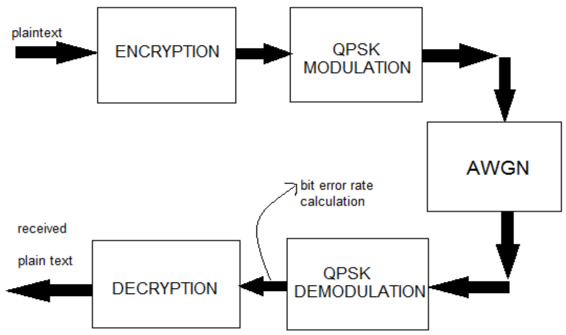



**Fig.11 Implementation over AWGN channel**

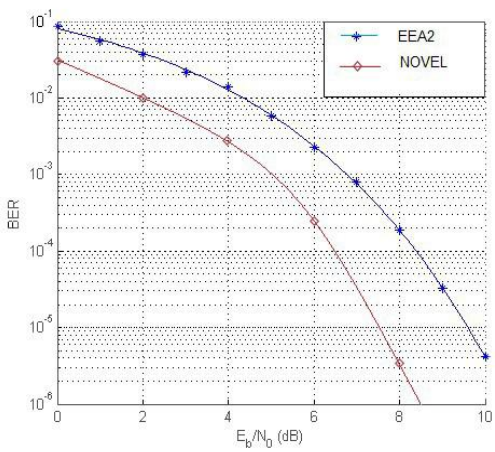



**Fig.12 BER results**

# 7. CONCLUSIONS

This study allows proposed algorithm for the encryption in the LTE using 256 bit input block cipher algorithm based on rc6. Shown in MATLAB how the novel algorithm works in comparison with the EEA2 algorithm. the conclusions that can be drawn from our paper that among EEA2 and proposed, It is found that our system is better owing to increased throughput ,increased avalanche effect, increased efficiency and decreased encryption time. Future work will be done by performing more tests and comparing the results to those of other proposed solutions, or we can apply it using quantum key distribution.

# 8. REFERENCES

[1] Ghizlane ORHANOU, Said EL HAJJI, Youssef BENTALEB and Jalal LAASSIRI, "EPS Confidentiality and Integrity mechanisms Algorithmic Approach", IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 4, No 4, July 2010.

[2] V.Kaul, P. Choudhari, and S. Narayankhedkar, "Security enhancement for data transmission in 4G networks," proc. of 5th International Conference Confluence of the Next Generation Information Technology Summit (Confluence), -, pp. 373-378, 2014.

[3] Milind Mathur, Ayaush Kesarwani. "Comparison between DES, 3DES, RC2, RC6 , BLOWFISH and AES" , proc. Of National conference on New Horizons in it,2013.

[4] Kirti Aggarawat , "Comparison of RC6 and modified RC6 ", International conference of Advances in Computer Engineering and applications, 2015.

[5] D. Forsberg, G. Horn, W.-D. Moeller, and V. Niemi, LTE security: John Wiley & Sons, 2012.

[6] H. Zarrinkoub, Understanding LTE with MATLAB: from mathematical modeling to simulation and prototyping: John Wiley & Sons, 2014.

[7] Walied W. Souror , Ali E. Taki el-deen, Rasheed Mokhtar el-awady Ahmed , Adel Zaghlul Mahmoud1 "An Implementation of High Security and High Throughput Triple Blowfish Cryptography Algorithm" International Journal of Research and Reviews in Signal Acquisition and Processing (IJRRSAP) Vol. 2, No. 1, March 2012.

[8] K. Aggarwal, "Comparison of RC6, modified RC6 & enhancement of RC6," proc. of International Conference on Advances Computer Engineering and Applications (ICACEA), pp. 444-449, 2015.

[9] Ali E. Taki El_Deen, Mohy E. Abo-Elsoud, Salma M. Saif," Text and Biomedical Images Disguising using Advanced Encryption Standard", International Journal of Engineering Research & Technology (IJERT)Vol. 2 Issue 12, December – 2013.

[10] Eman Ashraf Mohammed, Nihal Fayez Areed, Ali Takieldeen, Rasheed M. El-Awady," Hybrid Cryptographic Algorithm for LTE DataConfidentiality", International Journal of Engineering Research & Technology (IJERT),Vol. 5 Issue 12, December-2016.

[11] A. K. Mandal and A. Tiwari, "Analysis of Avalanche Effect in Plaintext of DES using Binary Codes," International Journal of Emerging Trends and Technology in Computer Science (IJETTCS), vol. 1, pp. 166-177, 2012.

[12] S. R. Basavarasu, "Voice and Image Encryption, and, Performance Analysis of Counter Mode Advanced Encryption Standard for WiMAX," University of Toledo, 2013.

# 9. AUTHOR PROFILE

**Nihal Fayez Areed** assoc prof. at communication and electronics dept. received the PhD degree of communication engineering. Her current research interests are in Electromagnetic fields Antennas and wave propagation, Photonic.

**Ali Taki El-Deen** (IEEE senior member) received the PhD degree in Electronics and Communications Engineering in "Encryption and Data Security in Digital Communication Systems".

**Eman Ashraf Mohammed** received BSc in Electronics and Communications, Master student.