International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 07 | July 2021 www.irjet.net p-ISSN: 2395-0072

VLSI Design of DNA Key Generating Algorithm for Symmetric Cryptographic Systems

Rahul Kulkarni[1], Manjunath H S[2], Darshan T[3], Sudha H[4]

[1,2,3]*B.E., in ECE, BIT, Bengaluru*
[4]*Associate Professor, Dept. of ECE, BIT, Bengaluru*

Abstract - *This work represents the design of a Key Generating Algorithm based on DNA Cryptography focused on providing an extra layer of security for protecting the key. An illegal proficient exercise, i.e., data theft during communications, is still going on and attempts are constantly being made in this area to break the encrypted data before reaching it to the authorized destination by some Cryptanalyst. On the other end, there is a lot of research going on to render the data safe through encrypting them during communication and a complex key generation process for decrypting encrypted data. Many latest attacks are made by learning the patterns of the key. So, to improve the security and to optimize the design, this approach generates DNA string as the key that provides enhanced security. The notion of using DNA computing in cryptography has been recognized as a potential breakthrough that might meet a new need for unbreakable algorithms. DNA Key Generating Algorithm has been developed for more secure data concealing and symmetric key generation utilizing genetic databases. The proposed algorithm is implemented using Verilog coding, simulated using Vivado simulator and synthesized using Vivado 2017.4.*

Key Words: **DNA Cryptography, AES Algorithm, DNA Digital Coding, DNA Key Generating Algorithm.**

1.INTRODUCTION

Cryptography is a field of science concerned with encoding of data in order to conceal communications. It is the area in which message/data can be enciphered and sent across the network and it can be deciphered to its original form. It plays a vital role in the infrastructure of communication security. Cryptography is a way of using codes to safeguard information and communications so that only those who are supposed to read and process it may do so. Here, the encryption takes place at the sender end and after getting the ciphered message, it is deciphered at the receiver end with the same key. [1]
Before the source and destination may communicate, both of these entities must agree on specific protocols for transferring information, which is similar to a handshake procedure. There is a need to adopt more secure and reliable Key generating and Encryption algorithm.

1.1 DNA Cryptography

It is one of the most secure and robust approach. DNA cryptography is the technique of concealing data using DNA sequences. DNA cryptography is the most recent advancement in cryptographic techniques, in which the natural process of DNA creation is utilized to encrypt data and later decode it. DNA cryptography is a subject in which several studies are ongoing, and it is still expected to produce improved solutions to modern-day difficulties and issues. PCR, DNA synthesis, and DNA Digital Coding are examples of DNA Cryptography technologies that have previously been adopted. [2]
Here we have adopted DNA Digital Coding technique where encoding and decoding is done with the binary values 0's and 1's. DNA Coding is based on biological structure of DNA which is composed of four basic nucleotide bases: Adenine - A, Cytosine - C, Guanine - G and Thymine - T.

1.2 AES Algorithm

The Advanced Encryption Standard Algorithm is the most widely used symmetric encryption algorithm. Rather than being a Feistel cypher, it is an iterative cypher. It is built on the basis of a 'substitution–permutation network'. It consists of a sequence of connected processes, which require substituting particular outputs for inputs (replacements) and others involving shuffling bits around (permutations). Remarkably, AES uses bytes for its calculations. As a result, AES interprets a plaintext blocks of 128 bits as 16 bytes. For matrix processing, these 16 bytes are organized into four columns and four rows. The number of rounds in AES is flexible and is determined by the key length. For 128-bit keys, AES employs 10 rounds. Each of these rounds use a unique 128-bit round key derived from the original AES key. [3] Each round of the AES algorithm (excluding round 10) has the following steps:

- Add round key
- Substitute bytes
- Shift rows
- Mix columns

The schematic of AES structure is as shown below:

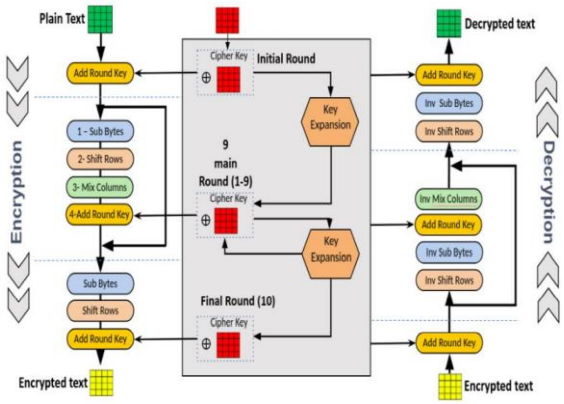

Fig -1: AES Structure

2. DNA Digital Coding

It is one of the evolving technologies in encrypting the information using the DNA strands. DNA Strands are lengthy polymers with a million nucleotide connections. 1 of 4 nitrogen bases, a 5-carbon sugar, and a phosphate gathering make up these nucleotides. This methodology produces information in randomized DNA Nucleotide form which is further converted to binary form and not easily predictable by the cryptanalyst. [4]

Table -1: DNA Coding Table

Binary Value	DNA Digital Coding
00	A
01	T
10	G
11	C

3. PROPOSED WORK

The proposed system utilizes DNA Digital Coding for key generation and maps digital data into DNA nucleotide bases. The AES algorithm has been used for encryption and decryption process. The key required for encryption and decryption is generated by the Key Generating Algorithm which is designed and explained in subsequent sections.
We are using DNA Digital coding to generate and randomize a key to enhance the security in an efficient way. This proposed system provides an extra stage of data security and improves the data storage capability.

The Block Diagram of the Proposed Design is as follows.

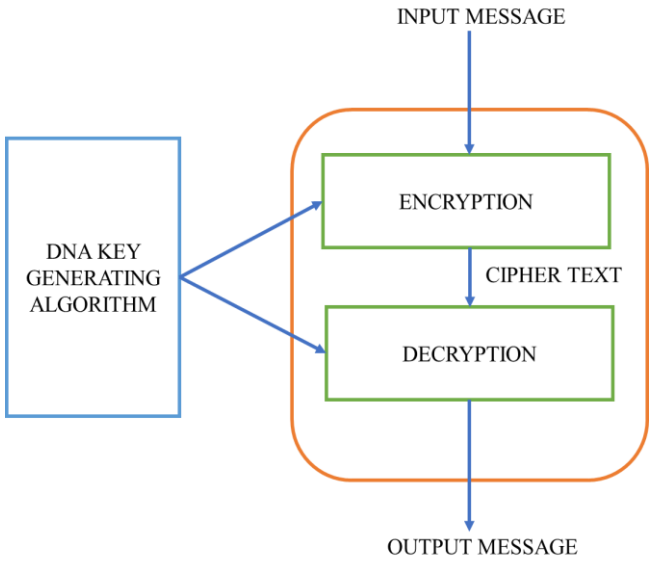

Fig -2: DNA Key Generation based Cryptosystem

4. DNA KEY GENERATION

The step by step coding is explained below:

- Step 1:
  The key in binary form is then grouped into pairs and using the DNA digital coding format from Table 1 the necessary DNA nucleotide is obtained. This is called bit encoding.
  Ex: Input Main Key, m_key = AESalgorithm@123 in binary form is –
  01000001010001010100110110000101101100011001110110111011100100110100101110100011010000110110101000000001100010011000100011 00 11.
  This binary form is grouped in pairs using Table 1 as:
  main_dna_key=TAATTATTTTACTGATTGCATGTCTGCCTCAGTGGTTCTATGGATGCTTAAAACATACAGACAC

- Step 2:
  The next step is annealing. Here, the obtained nucleotide in Step 2 is substituted with its complementary bases shown below.
  Complimentary Rule:
  - A → T
  - T → A
  - G → C
  - C → G

  Ex: DNA nucleotide obtained in Step 1 is substituted as:
  comp_dna_key=ATTAATAAAATGACTAACGTACAGACGGAGTCACCAAGATACCTACGAATTTTGTATGTCTGTG

- Step 3:
  Then the obtained nucleotide in Step 2 is concatenated with original nucleotide in Step 1 to obtain a new nucleotide.
  Ex:
  conc_dna=TAATTATTTTACTGATTGCATGTCTGCCTCAGTGGTTCTATGGATGCTTAAAACATACAGACACATTAATAAAATGACTAACGTACAGACGGAGTCACCAAGATACCTACGAATTTTGTATGTCTGTG

- Step 4:
  In this step, each T base is replaced(mapped) to U base in the new nucleotide obtained in Step 3.RNA is formed. This process is called transcription.
  Replace T → U
  Ex:
  rna_out=UAAUUAUUUUACUGAUUGCAUGUCUGCCUCAGUGGUUCUAUGGAUGCUUAAAACAUACAGACACAUUAAUAAAAAUGACUAACGUACAGACGGAGUCACCAAGAUACCUACGAAUUUUGUAUGUCUGUG

- Step 5:
  In RNA, we have namely three stop codons namely UAA, UGA, and UAG. We remove these stop codons (replace it with blank space) to get several keys of variable lengths.
  Ex:
  ___UUAUUUUAC___UUGCAUGUCUGCCUCAGUGGUUCUAUGGAUGCU___AACAUACAGACACAU___AAUGAC___CGUACAGACGGAGUCACCAAGAUACCUACGAAUUUUGUAUGUCUGUG

- Step 6:
  After removing the stop codons, now we convert the sequence back to DNA form by replacing U with T.
  Replace U → T
  Ex:
  dna_f_key=___TTATTTTAC___TTGCATGTCTGCCTCAGTGGTTCTATGGATGCT___AACATACAGACACAT___AATGAC___CGTACAGACGGAGTCACCAAGATACCTACGAATTTTGTATGTCTG

- Step 7:
  The several keys of variable lengths are separated and stored in registers as shown in the example
  Ex: r1= ___
  r2= TTATTTTAC
  r3= TTGCATGTCTGCCTCAGTGGTTCTATGGATGCT
  r4= AACATACAGACACAT
  r5 = ___
  r6 = AATGAC
  r7=CGTACAGACGGAGTCACCAAGATACCTACGAATTTTGTATGTCTGTG

- Step 8:
  The longest key among the registers is chosen as the Generated DNA Key.
  Ex:
  T=CGTACAGACGGAGTCACCAAGATACCTACGAATTTTGTATGTCTG

- Step 9:
  128-bits of Generated DNA Key (16 characters) is chosen and given as the key for Encryption & Decryption modules of AES-128.
  Ex: key=GAATTTTGTATGTCTG

- Step 10:
  Using the key obtained in step 9 and the Plaintext given initially to, the encrypted text (cipher text) is obtained by AES-128 Encryption.
  Ex:
  Plaintext= BIT_college@1979
  encdata = % h ! x{

- Step 11:
  With encrypted data (cipher text) and the Key obtained in step 9, The Decrypted data(Plaintext) is obtained back.
  Ex: decdata= BIT_college@1979

5. SIMULATION RESULTS

The system is designed in Verilog and simulated using Vivado Simulator and various simulation outputs are shown below.

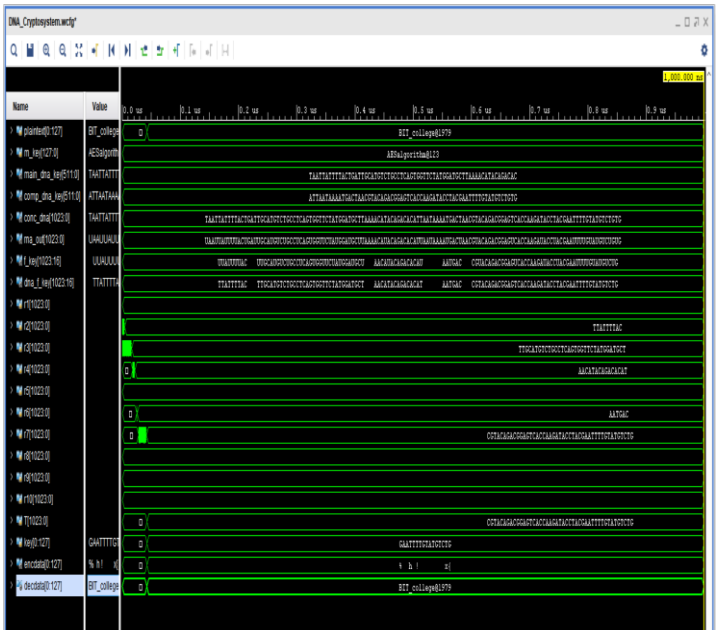

Fig -3: Cryptosystem with DNA Key Generation and AES

6. SYNTHESIS RESULTS

The design is synthesized by Vivado 2017.4 with ZedBoard Zynq Evaluation and Development Kit (xc7z020clg484-1) FPGA as the target device.

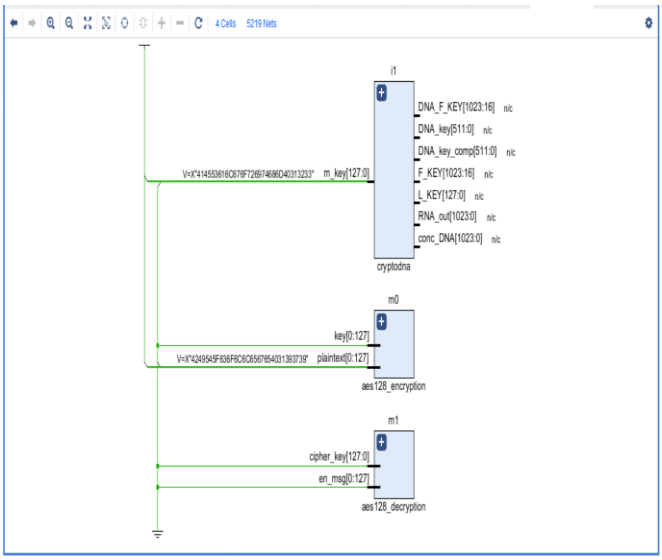

Fig -4: RTL Schematic with Top Modules

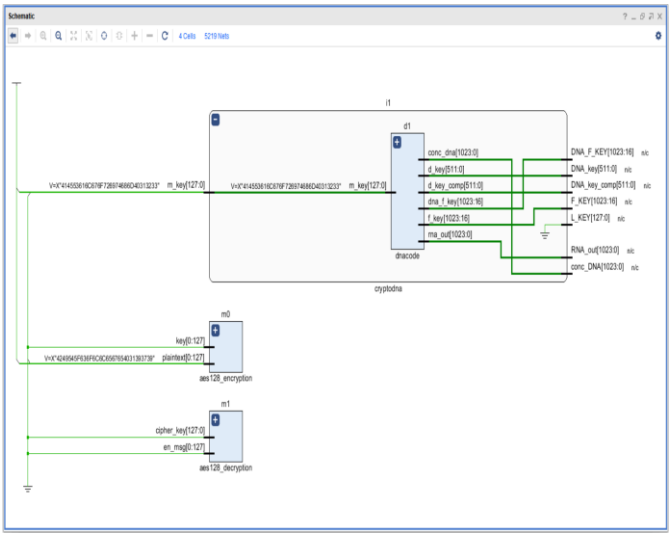

Fig -5: RTL Schematic with elaborated modules

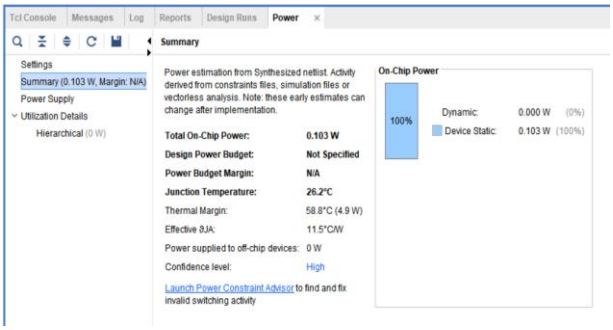

Fig -6: Power Report

7. CONCLUSIONS

The DNA Key Generating Algorithm can be used as a strong algorithm in improving the security of the Key. Its breaking time and key generation are engineered in such a way that it appears like decrypting the ciphered data would take an eternity. The technique is highly robust against different attacks since a random key is produced at the sender every time and is used to decrypt the cipher text at the receiver. The modern cryptosystems can implement the designed algorithm in future. The proposed approach can be further extended to accept other forms of data like MP3, MP4, .txt files and other multimedia files. The current work will also aid in the implementation and use of DNA-based cryptography and steganography techniques.

ACKNOWLEDGEMENT

The finest gesture one can make towards another is gratitude. Any achievement is incomplete without acknowledging the individuals who helped make it possible. We would like to take this occasion to convey our heartfelt gratitude and best wishes to Sudha H, Associate Professor, Department of ECE, BIT, Bengaluru our guide and mentor, for her consistent support and encouragement. We owe her a huge debt of gratitude for her prompt contributions, thorough reviews, and helpful ideas.

REFERENCES

[1] Bahubali Akiwate, Latha Parthiban, “A Dynamic DNA for Key-based Cryptography”, 2018.

[2] Anupam Das, Shikhar Kumar Sarma, Shrutimala Deka, “Data Security with DNA Cryptography”, 2019.

[3] Thockchom Birjit Singha, Roy Paily Palathinkal, Shaik Rafi Ahamed, “Implementation of AES Using Composite Field Arithmetic for IoT Applications”, 2020.

[4] Y. Bhavani, Sai Srikar Puppala, B.Jaya Krishna, Srija Madarapu, “Modified AES using Dynamic S-Box and DNA Cryptography”, 2019

[5] William Stallings, “Cryptography and Network Security: Principles and Practice”.