DATA SECURITY APPLICABLE FOR IP USING ALGORITHMS AND ITS IMPLEMENTATION ON FPGA

Mrs. Divyashree M[1], Nikhil P N[2], Prashanth B[3], Sudeep G[4], Vardhman Ramu Manakapure[5]

[2,3,4,5] Students, Telecommunication Department, MVJ College of Engineering, Bangalore, Karnataka, India.
[1] Asst. Professor, Telecommunication Department, MVJ College of Engineering, Bangalore, Karnataka, India.

---***---

Abstract - The quick advancement of computerized information trade has constrained the data security to be of much essential in information stockpiling and transmission. As large amount of data is transmitted over the network, it is preliminary to secure all types of data before sending them. Data can be broken using algebraic cryptanalysis. This provides a serious threat so by combination of AES, MD5 and CRC can be considered to be unbreakable and thus it can be used in many systems. A new technique is developed in which the combination of AES, MD5 and CRC algorithms using FPGA that can be used as a standard device in the secured communication system. These algorithms are implemented in the FPGA with the help of VHDL or Verilog. This results in very low frequency requirement to perform this operation with consideration of high speed by reducing the gate counts with low power consumption of whole circuit, multiple key size support and low cost compared to earlier methods. The information to encryption side is in the form of plain text and the same will appear in the decryption side and its real time input/output also achieved effectively. The hardware design is targeted on Xilinx Spartan 3AN device and it supports lower versions as well. To improve the quality of data security systems, in this research the integration of AES 128 bit algorithm by using MD5 hashing is proposed. The use of MD5 aims to increase the key strength of the encryption and decryption process of document files. Encryption and decryption takes lower time by using AES and MD5 combination is faster than using AES only. This technique can also be applicable for securing IP as well.

We are planning this venture for the high security of the information by encryption and unscrambling utilizing different calculations as said.

KEY WORDS: IPSec, AES, MD5, XILINX, FPGA etc.

1. INTRODUCTION

Data security[1] play vital role in secure the message while sending data in internet, since there are lot of hackers, and malware as threats data needed to be secured. Many methods are used to protect the transfer of data, including encryption and from the ground up engineering. The current focus is on prevention as much as on real time protection against well-known and new threats. The network attacks can be classified as passive attacks or active attacks. The former learns and makes use of the information from the system without the awareness of the system administrator. Data security is a branch of computer security specifically related to the Internet, often involving browser security but also network security on a more general level, as it applies to other applications or operating systems as a whole. Its objective is to establish rules and measures to use against attacks over the Internet.

The primary capacity of the work introduced in this work is following.

1. **Advanced Encryption Standard** (AES) [2] is a plan which utilizes both substitution and change which is so quick in both equipment and programming. It has a settled square of 128 bits and key size of 128 bit, 192 bits and 256bits.
2. **Message Digest (MD5)** [3] is an arrangement which uses both substitution and change which is so brisk in both hardware and programming. It has a settled square of 128 bits and key size of 128 bit,192 bits and 256bits.
3. **IP checksum** is a value that is computing frame data packet to check its integrity what's more, its a mistake identification code which is ordinarily utilized as a part of computerized system and capacity gadgets and to recognize coincidental changes to crude information. It is known as checksum because the value or the message can be expanded with adding information.

1.1 OBJECTIVE

To configuration propelled encryption standard calculation with a reasonable and less perplexing framework. To give the solid security usage in a firewall that can be executed to all IoT applications utilizing FPGA can be straight forward to end clients and can give security to singular clients. High level of encryption by utilizing three different calculation for greater security Internet convention is more impervious to send the information if all the movement for the outside must utilize IP and firewall is the best way to entrance. Web security is a branch of PC security particularly identified with the Internet, regularly including security yet additionally arrange security on a more broad level, as it applies to different applications or working frameworks overall. Its goal is to set up guidelines and measures to use against assaults over the Internet. The Internet speaks to a shaky channel for trading data prompting a high danger of interruption or misrepresentation, for example, phishing, online infections, Trojans, worms and that's only the tip of the iceberg.

Numerous techniques are utilized to ensure the exchange of information, including encryption and starting from the earliest stage designing. The present spotlight is on counteractive action as much as on continuous assurance against understood and new dangers.

2. METHODOLOGY

This framework comprise of following Algorithms:

- Advanced Encryption Standard (AES).
- Message Digest 5 (MD5).
- IP check sum.

This system works on

- Xilinx Spartan 3 and Spartan 3N.
- FPGA board.

BLOCK DIAGRAM

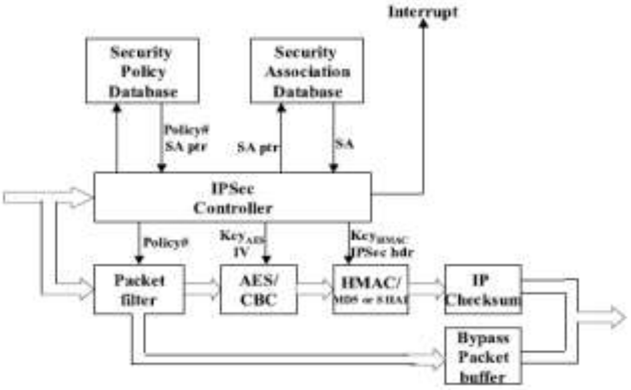


Figure1. Block Diagram.

4. Advanced Encryption Standard [2]

- **Advanced Encryption Standard** (**AES**) likewise known by its unique name Rijndael is a detail for the encryption of electronic information.

- AES works on a 4 × 4 section significant request network of bytes, named the state, albeit a few adaptations of Rijndael have a bigger piece measure and have extra segments in the state. Most AES computations are done in a specific limited field.

For example, if there are 16 bytes b0, b1, b2, b3……b15 this bytes are represented as 4X4 matrix.

$$\begin{bmatrix} b_0 & b_4 & b_8 & b_{12} \\ b_1 & b_5 & b_9 & b_{13} \\ b_2 & b_6 & b_{10} & b_{14} \\ b_3 & b_7 & b_{11} & b_{15} \end{bmatrix}$$

The number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the cipher text.

The number of cycles of repetition is as follows:

- 10 cycles of repetition for 128-bit keys.
- 12 cycles of repetition for 192-bit keys.
- 14 cycles of repetition for 256-bit keys.

Basically there are various steps for encryption of the data.

4.1 The Sub Bytes step

In sub byte advance, in every byte ai,j in a state framework is supplanted with a subbyte S(ai,j) utilizing a 8 bit substitution box (S-box).It gives non linearity in figure. Non linearity implies which gives multiplicative reverse GF(2^8).To abstain from assaulting on straightforward mathematical properties, the S-box is developed by consolidating the opposite capacity. It additionally keep away from picked settled point ie, S(ai,j) =/ai,j.In this the information is substituted as S(ai,j)=S(aj,i).

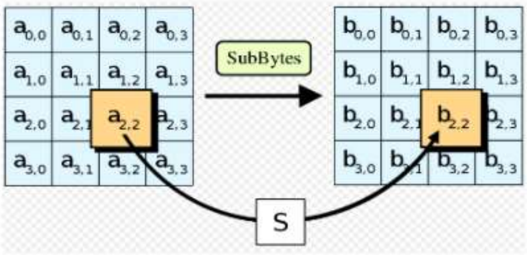

$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
SubBytes
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,1}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$
S

Fig2. Ithis each byte is replaced is S(ai,j)=S(aj,i).

4.2 The Shift row step

The Shift Rows step operates on the rows of the state. It will cyclically shifts each byte in each row. The first row will not change. Each byte of the second row is shifted one to the left. Similarly, the third and fourth rows are shifted by offsets of two and three respectively. The shifting pattern is the same. Row n is shifted left circular by n-1 byte.

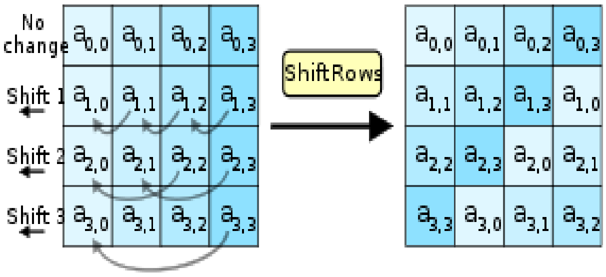

No change
Shift 1
Shift 2
Shift 3
$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
ShiftRows
$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,1}$ $a_{1,2}$ $a_{1,3}$ $a_{1,0}$
$a_{2,2}$ $a_{2,3}$ $a_{2,0}$ $a_{2,1}$
$a_{3,3}$ $a_{3,0}$ $a_{3,1}$ $a_{3,2}$

Figure 3: In this figure each byte is shifted cyclically.

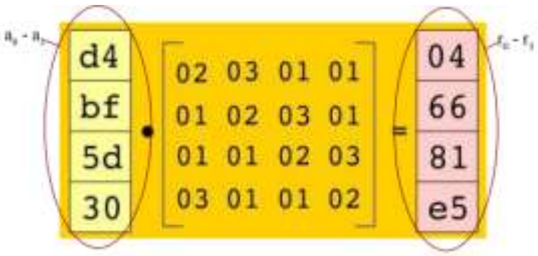
4.3 The Mix Columns step

If there should be an occurrence of this progression, the four byte of every section of the state are consolidated utilizing invertible straight change. In this stage four bytes are input bytes and four bytes are yield bytes in which every one of the four bytes influences the yield bytes.

In the event of this progression, the four byte of every segment of the state are consolidated utilizing invertible straight change. In this stage four bytes are input bytes and four bytes are yield bytes in which every one of the four bytes influences the yield bytes.

During this operation, each column is transformed using a fixed matrix.

Let us take this below example



In this example, our a0 – a3 is equals to d4 – 30 and r0 – r3 is equals to 04 – e5. One thing to note in this is that it still follows the matrix multiplication rules. Currently the matrix size looks like,

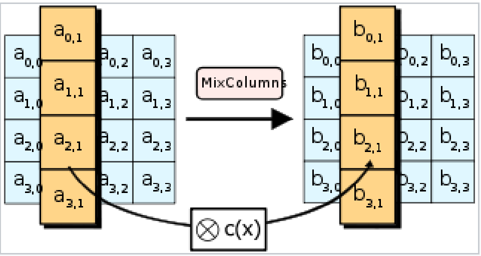
[4 x 1] . [4 x 4] ≠ [4 x 1]

But we need the formula to be

[4 x 4] . [4 x 1] = [4 x 1]

We get,

$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix}$$



Each column of the state is multiplied with the fixed polynomial c(x).

Mix column step is calculated by using formula.

$$\begin{bmatrix} r_0 \\ r_1 \\ r_2 \\ r_3 \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

where r0, r1, r2 and r3 are the results after the transformation. a0 – a3 can be obtain from the matrix after the data undergoes substitution process in the S-Boxes.

4.4 The add round key

In the Add Round Key advance, the sub key is joined with the state. For each cycle, a sub key is gotten from the fundamental key each sub key is an indistinguishable size from the state. The sub key is included by joining every byte of the state with the relating byte of the sub key utilizing bitwise XOR.

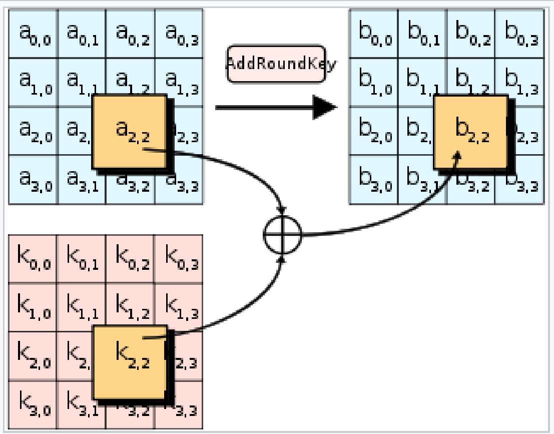

$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
AddRoundKey
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$
$k_{0,0}$ $k_{0,1}$ $k_{0,2}$ $k_{0,3}$
$k_{1,0}$ $k_{1,1}$ $k_{1,2}$ $k_{1,3}$
$k_{2,0}$ $k_{2,2}$ $k_{2,3}$
$k_{3,0}$ $k_{3,1}$ $k_{3,2}$ $k_{3,3}$

Figure 4.4: Each byte of the state is combined with the byte of the round sub key using XOR operation.

5. Message Digest (MD5) [3]

In cryptography, MD5 (Message-Digest calculation 5) is a generally utilized cryptographic hash work with a 128-piece hash esteem. As an Internet standard MD5 has been utilized in a wide assortment of security applications, and is additionally ordinarily used to check the honesty of files.MD5 (Message process calculation) takes as info a message of self-assertive length and creates as yield a 128bits message process of the input. The MD5 calculation is proposed for computerized signature application, where a vast document must be packed in a safe way before being encoded with a private key under an open key cryptosystem, for example, RSA. Takes as info a message of discretionary length and creates as yield a 128 piece "unique mark" or "message process" of the information. It is guessed that it is computationally infeasible to deliver two messages having a similar message process. Proposed where a substantial document must be "packed" in a safe way before being encoded with a private key under an open key cryptosystem, for example, PGP(Pretty Good Privacy).PGP is a prominent program used to scramble or decode the confirm messages with advanced mark and scrambled put away records.

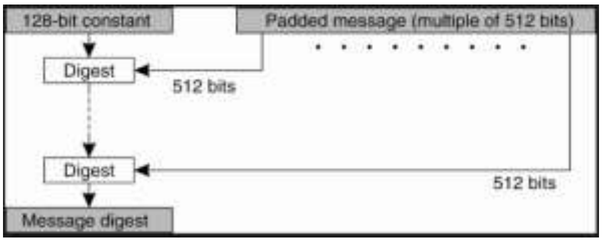


Figure 6: MD5 algorithm structure

5.1 MD5 Implementation steps

Give the information a chance to be b-bit message as info, and that we wish to discover its message process. Here b is a subjective non negative whole number, b might be zero, and it might be discretionarily vast. Give us a chance to consider

m_0, m_1,m_2,… ... m_{b-1}

The five steps are performed to compute the message digest of the message.

Step 1.Append Padding Bits

The message is "padded" (extended) so that its length (in bits) is congruent to 448, modulo 512. Padding is always performed, even if the length of the message is already congruent to 448, modulo512.A single "1" bit is appended to the message, and then "0" bits are appended so that the length in bits of the padded message becomes congruent to 448, modulo 512.In all, at least one bit and at most 512 bits are appended.

Step 2. Append Length

A 64-bit portrayal of b is attached to the consequence of the past step1. In this occasion if b is more noteworthy than 2^64, at that point just the low-arrange 64 bits of b are used. The coming about message (subsequent to cushioning with bits and with b) has a length that is a correct various of 512 bits. The information message will have a length that is a correct different of 16 words. (32bits).

Step 3. Initialize MD Buffer

A four-word cushion (A,B,C,D) is utilized to process the message process.

Here each of A, B, C, D is a 32-bit enroll. These registers are instated to the accompanying qualities in hexadecimal, low-arrange bytes first.

word A **:** 01 23 45 67
word B **:** 89 ab cd ef
word C **:** fe dc ba 98
word D **:** 76 54 32 10

Step 4. Process Message in 16-Word Blocks

Four functions will be defined such that each function takes an input of three 32bits words and produces a 32 bit word output.

F (X,Y,Z) = XY U not(X) Z
G (X,Y,Z) = XZ U Y not(Z)
H (X,Y,Z) = X xor Y xor Z
I (X,Y,Z) = Y xor (X Unot(Z))

Step 5. Output

The message digest produced as output is A, B, C, D. That is, we begin with the low-order byte of A, and end with the high-order byte of D.

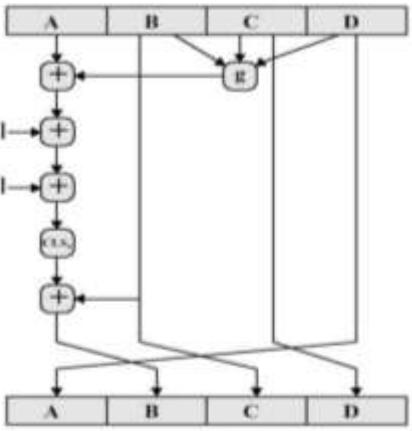


Figure 7 . Process of 512 bit message block

6. Cyclic redundancy check (CRC) [4]

A cyclic excess check (CRC) is a blunder identifying code generally utilized as a part of advanced systems and capacity gadgets to distinguish incidental changes to crude information. CRCs are supposed on the grounds that the check (information confirmation) esteem is an excess (it grows the message without including information).A CRC-empowered gadget figures a short, settled length twofold succession, known as the check esteem or CRC, for each piece of information to be sent or put away and shaping a codeword.

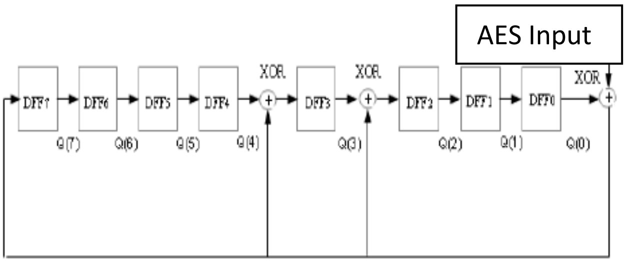
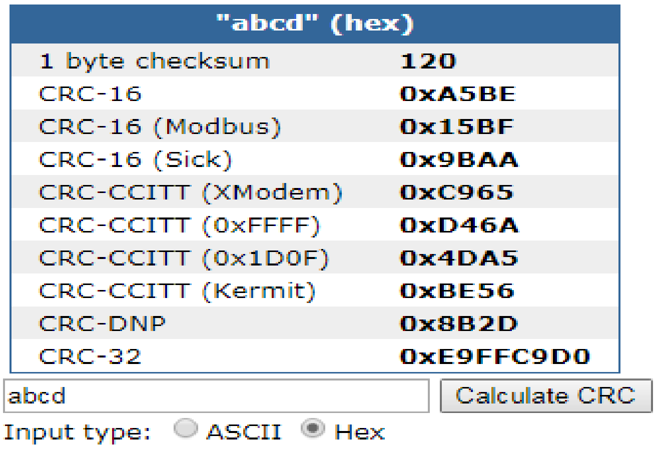


Fig. cyclic redundant code

6.1 CRC Online Calculator [5]

Whenever digital data is stored or interfaced, data corruption might occur. Since the beginning of computer science, people have been thinking of ways to deal with this type of problem. For serial data they came up with the solution to attach a parity bit to each sent byte. This simple detection mechanism works if an odd number of bits in a byte changes, but an even number of false bits in one byte will not be detected by the parity check. To overcome this problem people have searched for mathematical sound mechanisms to detect multiple false bits. The **CRC** calculation or Cyclic Redundancy Check was the result of this. Nowadays CRC calculations are used in all types of communications. All packets sent over a network connection are checked with a CRC. Also each data block on your hard disk has a CRC value attached to it. Modern computer world cannot do without this CRC calculation. So let's see why they are so widely used. The answer is simple they are powerful, detect many types of errors and are extremely fast to calculate especially when dedicated hardware chips are used. This calculator is used for generating CRC values



"abcd" (hex)	
1 byte checksum	120
CRC-16	0xA5BE
CRC-16 (Modbus)	0x15BF
CRC-16 (Sick)	0x9BAA
CRC-CCITT (XModem)	0xC965
CRC-CCITT (0xFFFF)	0xD46A
CRC-CCITT (0x1D0F)	0x4DA5
CRC-CCITT (Kermit)	0xBE56
CRC-DNP	0x8B2D
CRC-32	0xE9FFC9D0

abcd

Input type: ○ ASCII ● Hex

Fig 6.1: Online CRC calculator

APPLICATONS

- Data starting point confirmation distinguishing who sent the information.

•Confidentiality (encryption)- that the information has not been perused by anybody and not course by anybody.

•Connectionless honesty guaranteeing that the information has not been changed.

•Reply insurance detection of bundles got more than once to help ensure against dissent of administration assault.

•Extensible Network Services Platform.

Advantages:

•It is greater adaptability of web convention.

•It is so simpler to keep up and more secure also.

•Security at the system layer level which is totally imperceptible in its task.

•IPSec is perfect for checking and securing a wide range of web movement, inbound and in addition outbound.

•No application reliance since whole security framework is executed at the system level, in this way there is no applications issues.

Disadvantages:

•CPU overhead.

•Compatibility Issues.

•Broken calculations.

7. CONCLUSION

Secure control and configuration of network devices are needed to protect reconfigurable systems from attack. Modern network devices use FPGA technology to provide high performance and programmability. Power PC processor cores on the latest Xilinx FPGAs enhance the reconfigurable logic with embedded processing capability. The key plays a vital role in communication between sender and receiver. The hardware implementation performs the data encryption and decryption of data path with minimum software interference. Through the implementation of the AES and CRC using FPGA Spartan3AN can achieve complete data secure. These algorithms can be applied for data security implementation to a network device to provide a secure communication.

7.2 Data Encryption and Decryption Output Using AES and CRC Result

The combination of AES and CRC results can be obtained. We are using done1 signal, when it is high then plain text gets AES encrypted and CRC valve is obtained, for done2 is high AES decrypted value which is similar to plain text value and CRC value is obtained. If the key is changed then the decrypted value will not be same as the encrypted results.

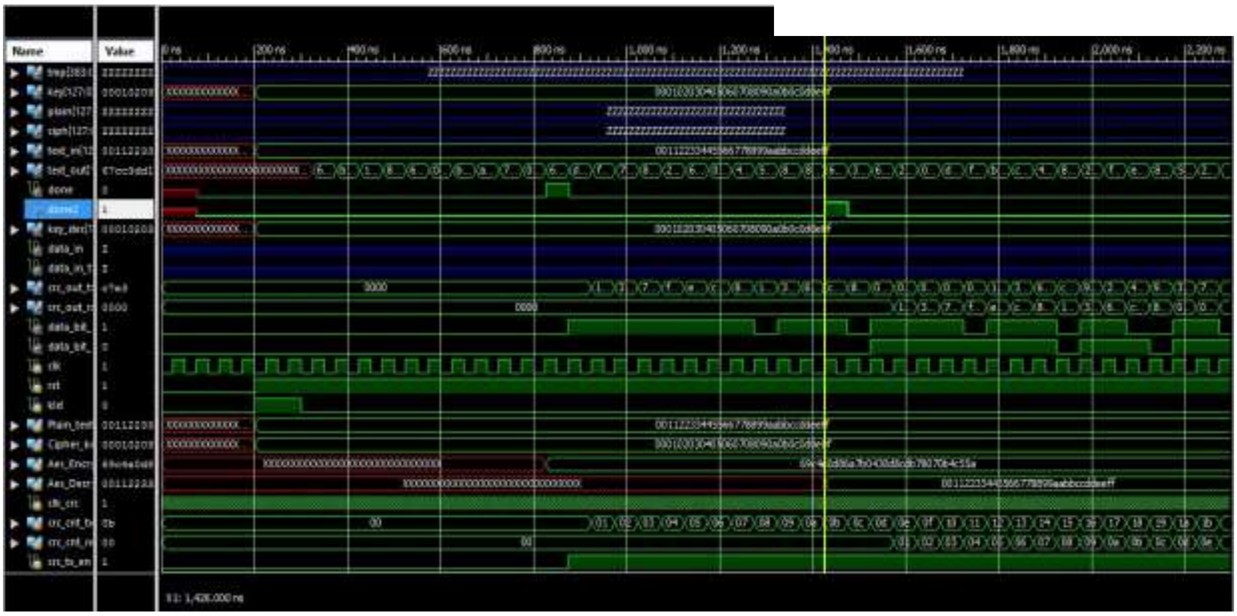


Fig . 7.2 Data Encryption and Decryption Output Using AES and CRC Result

REFERENCES

[1] HuiSuoa ,Jiafu Wan, CaifengZoua , JianqiLiua, “Data Security in the Internet of Things: A Review”, IEEE International Conference on Computer Science and Electronics Engineering, 2012, DOI: DOI 10.1109/ICCSEE.2012.373.

[2] An Implementation of AES Algorithm Based on FPGA,Wei Wang School of Electronics Engineering and Automation, Tianjin Polytechnic University, 300387, China

[3] M. Stevens, Fast collision attack on MD5. EPrint-2006-104, pp. 1–13 (2006),
http: //eprint.iacr.org/2006/104.pdf

[4] Philip Koopman, TridibChakravarty , “Cyclic Redundancy Code (CRC) Polynomial Selection For Embedded Networks,” The International Conference on Dependable Systems and Networks, DSN-2004

[5]https://www.lammertbies.nl/comm/info/crc-calculation.html