Scientific Journal Impact Factor (SJIF) : 1.711
International Journal of Modern Trends in Engineering and Research
ISSN (ONLINE) : 2349 - 9745 ISSN (PRINT) : 2393 - 8161

A NOVEL VLSI ARCHITECTURE FOR ENHANCED SECURITY OF DATA USING 3DES

Ms. Sruthi Thampi P.S.
Dept.of Electronics And Communication Mohandas College of Engineering And Technology
Thiruvananthapuram, India

Abstract—With the fast change in technologies today, more and more multimedia data are generated and transmitted, leaving our data vulnerable to be edited, modified and dupli- cated. The classical way of transmitting redundant data over a bandwidth constrained insecure channel is to first compress it and then encrypt. The novelty in this paper is reversing the order of compression and encryption, without compromising either the encryption efficiency or the information secrecy. Here, a combined encryption and compression scheme is used, in which triple data encryption standard (3DES) with block ciphering will be used to achieve an enhanced security while making it suitable for compression using Run Length Encoding (RLE). By compressing the encrypted data, we can send more data through the channel, whereby we can achieve efficient channel utilization without compromising on the security of the encryption scheme. For combating bit errors, an improved Hamming code with selective bit placement algorithm will be used, which can efficiently detect and correct errors. The overall system can achieve improved detection accuracy, better channel utilization and provide extremely secure and authentic communication. The entire system is functionally simulated and verified using Xilinx ISE Design Suite 14.5 and Modelsim SE 6.5b.

Keywords-Triple data encryption standard (3DES); Encrypted Data; Run Length Encoding (RLE); Hamming code

I. INTRODUCTION

The electronic and information revolutions have brought a plethora of sophistications to the today's world. Computer, one of the versatile inventions of human, always has more to offer for the benefit of the planet. Information, the most sought after commodity of electronic epoch, proves itself as an icon of power. Particularly if the information is confidential and is of critical utility, the power it wields be- comes immense. In order to prevent misuse of this enormous power by unauthorized people, security systems have to be implemented to guard the powerbase.

Security of the data conventionally is relied on the encryption techniques. Simply, with the uprising number of established and successful attacks like cryptanalysis or worst case brute force attacks on encryption based systems, this is high time some improved security system has to be built up. In traditional schemes, the data is compressed, encrypted and then encoded for transmission; which are prone to bit errors while transmission and also poses a security risk as the compressed and encrypted data are susceptible to statistical analysis based decryption mechanisms. Due to these rea- sons, the existing systems are not very reliable and usually encrypted data is transmitted without any compression, but this contributes to poor channel utilization.

This paper focus on compression of encrypted data where the encryption procedure utilizes block cipher such as the Triple Data Encryption Standard (3DES) and compression using Run Length Coding (RLE). Loosely speaking, block ciphers operate on inputs of fixed length and serve as an im- portant building

blocks that can be used to construct secure encryption schemes. By compressing the encrypted data, more data can be send through the channel whereby it can achieve efficient channel utilization without compromising on the security of the encryption scheme. For combating bit errors, hamming codes are used which can efficiently detect and correct bit errors. The overall system can achieve improved detection accuracy, better channel utilization and provide highly secure and reliable communication.

In Section II, we concentrate on the problems that we seek to solve and summarizes the existing work on the subject. After a short review of the existing problems, on Section III, we discuss about encryption scheme using DES as well as 3DES algorithm and also explain how it works. Section IV deals with RLE scheme, which is used for compression. Furthermore Section V covers the need of security and shows how the error detection and correction systems helps to maintain security for a transmitted message. Section VI presents some simulation results and discussions. Finally conclusions and future work discussions are given in Section VII.

II. OVERTURES

In the last decade, we have seen an unprecedented explosion of textual information through the use of the Internet, digital library and information retrieval system. As the internet user's growth rate increases day by day, the need for secure transmission of data as well as efficient channel utilization also increases. Both encryption and compression are two important factors that enable to improve the communication system,but which should be preceding the other was a question of great importance.Traditionally, a compressed data is encrypted and is then encoded for transmission,as shown in fig. 1, which is more prone to bit errors and it also poses a high security risk as the compressed and encrypted data are susceptible to statistical analysis based decryption mechanisms. Due to these reasons,the existing schemes are not very reliable and usually encrypted data is sent without any compression but this leads to poor channel utilization.

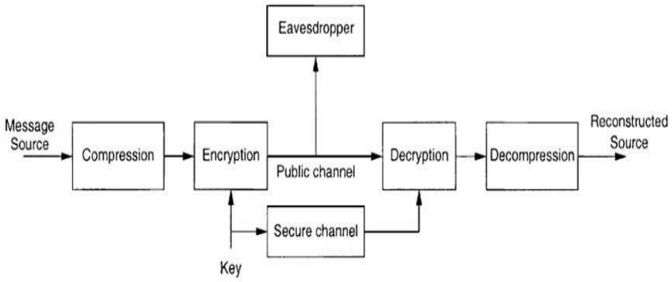

Figure 1. Traditional System

Through this piece of work,a new communication scheme have been proposed which can provide enhanced security without compromising on channel utilization and has capability of correcting bit errors as shown in fig 2. Here, a combined encryption and compression scheme is used, in which triple data encryption standard (3DES) with block ciphering will be used to achieve an enhanced security while making it suitable for compression using Run Length Encoding (RLE). By compressing the encrypted data, we can send more data through the channel, whereby we can achieve efficient channel utilization without compromising on the security of the encryption scheme. For combating bit errors, an improved Hamming code with selective bit placement algorithm will be used, which can efficiently detect and correct errors. The receiver receives a corrupted form of transmitted word which is corrected by the error correction scheme and then decompressed and decrypted. So, the error correction plays an important role in the overall efficiency of the proposed scheme. The overall system can achieve improved detection

accuracy, better channel utilization and provide highly secure and reliable communication. The proposed scheme can be used in all bandwidth limited channels which demands high security.

III. ENCRYPTION ALGORITHM

Security attacks against network are increasing significantly with time. Our communication media should also be secure and confidential. Cryptosystem is a system or product that provides encryption and decryption. Cryptosystem uses an encryption algorithms which determines how simple or complex the encryption process will be. In encryption, key is a piece of information which states the particular conversion of plaintext to ciphertext, or vice versa during decryption. Depending on the algorithm, and length of the key, the strength of encryption can be measured.

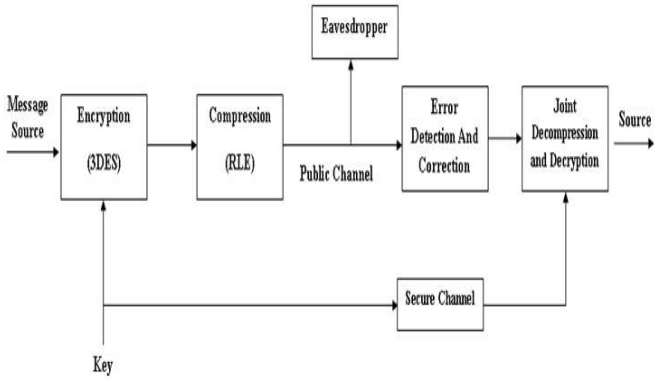

Figure 2. Proposed System

There are two encryption/decryption key types: In some of encryption technologies when two end points need to communicate with one another via encryption, they must use the same algorithm, and in the most of the time the same key, and in other encryption technologies, they must use different but related keys for encryption and decryption purposes. Cryptography algorithms are either symmetric algorithms, which use symmetric keys (also called secret keys), or asymmetric algorithms, which use asymmetric keys (also called public and private keys). Symmetric encryption system uses only private keys, which can be anything from a numerical symbol to a string of random letters. As a solution for the not completely safe Symmetric Encryption, there is the Asymmetric Encryption system that uses a pair of keys for added security: a private and a public key.

A. 3DES Algorithm

In cryptography, 3DES (Triple DES) is the common name for the Triple Data Encryption Algorithm (TDEA or Triple DEA) symmetric-key block cipher, which applies the Data Encryption Standard (DES) cipher algorithm three times to each data block. The main purpose behind the development of Triple DES was to address the obvious flaws in DES without making an effort to which each map 6 input bits to 4 output bits, producing a 32-bit output, which is then permuted by permutation P. This is designed in such a way that, after expansion, each S-box's output bits are spread across 6 different S boxes in the next round. 3DES simply extends the key size of DES by applying the algorithm three times in succession with three keys.

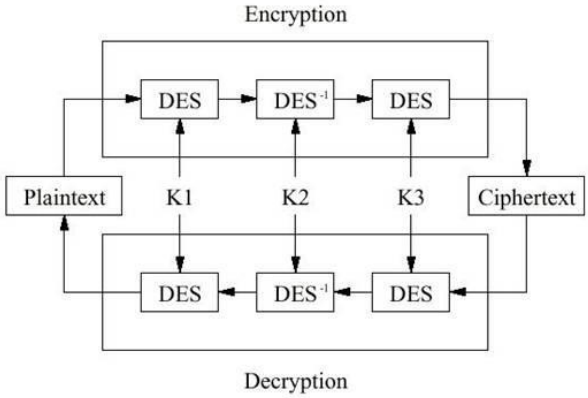

Figure 3. Working of Triple DES Algorithm

The Data Encryption Standard (DES) is a block cipher that uses shared secret encryption. It was first adopted in the year 1977 by the National Bureau of Standards as Federal Information Processing Standard 46 (FIPS PUB
46). DES encrypts data in 64-bit blocks using a 56-bit key. The algorithm was initially controversial with classified design elements, a relatively short key length, and suspicions about a National Security Agency (NSA) backdoor. DES consequently came under intense academic scrutiny which motivated the modern understanding of block ciphers and their cryptanalysis.

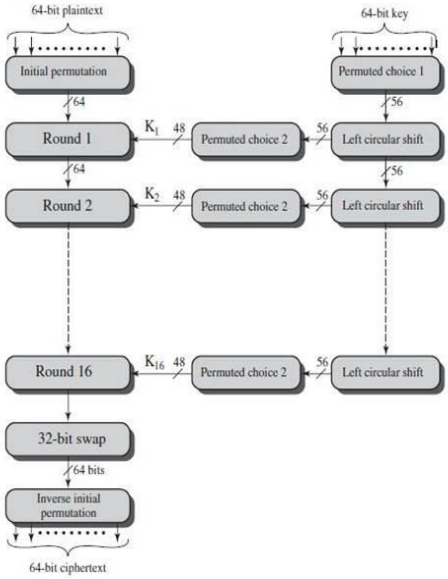

Figure 4. General depiction of DES algorithm

The algorithm's overall structure is shown in Fig.4: there are 16 identical stages of processing, termed rounds. There is also an initial and final permutation, termed IP and FP, which are inverses (IP"undoes" the action of FP, and vice versa). IP and FP have no cryptographic significance, but were included in order to facilitate loading blocks in and out of mid-1970s 8-bit based hardware. Before the main rounds, the block is divided into two 32-bit halves and processed alternately; this criss-crossing is known as the Feistel scheme. The Feistel structure ensures that decryption and encryption are very similar processes the only difference is that the sub keys are applied in the reverse order when decrypting. The rest of the algorithm is identical. This greatly simplifies implementation, particularly in hardware, as there is no

need for separate encryption and decryption algorithms.. The F-function scrambles half a block together with some of the key. The output from the F-function is then combined with the other half of the block, and the halves are swapped before the next round. After the final round, the halves are swapped; this is a feature of the Feistel structure which makes encryption and decryption similar processes.

The F-function (feistel function), operates on half a block (32 bits) at a time and consists of four stages:

- Expansion - the 32-bit half-block is expanded to 48 bits using the expansion permutation, denoted E in the diagram, by duplicating half of the bits. The output consistsof eight 6-bit (8 * 6 = 48 bits) pieces, each containing a copy of 4 corresponding input bits, plus a copy of the immediately adjacent bit from each of the input pieces to either side.
- Key mixing - the result is combined with a subkey using an XOR operation. 16 48-bit subkeys one for each round are derived from the main key using the key schedule.
- Substitution - after mixing in the subkey, the block is divided into eight 6-bit pieces before processing by the Sboxes, or substitution boxes. Each of the eight S-boxes replaces its six input bits with four output bits according to a non-linear transformation, provided in the form of a lookup table. The S-boxes provide the core of the security of DES;without them, the cipher would be linear, and trivially breakable.
- Permutation - finally, the 32 outputs from the S-boxes are rearranged according to a fixed permutation, the P- box. This is designed so that, after permutation, each S-box's output bits are spread across 4 different S boxes in the next round.

For a single F-function, the left and right halves of each 64-bit intermediate value are treated as separate 32-bit quantities, labeled L (left) and R (right). As in any classic Feistel cipher, the overall processing at each round can be summarized in the following formulas:

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$$

Right hand side of fig 4. illustrates the key schedule for encryption the algorithm which generates the subkeys. Initially, 56 bits of the key are selected from the initial 64 by Permuted Choice 1(PC-1) the remaining eight bits are either discarded or used as parity check bits. The 56 bits are then divided into two 28-bit halves; each half is thereafter treated separately. In successive rounds, both halves are rotated left by one or two bits (specified for each round), and then 48 subkey bits are selected by Permuted Choice 2(PC-2) 24 bits from the left half, and 24 from the right. The rotations mean that a different set of bits is used in each subkey; each bit is used in approximately 14 out of the 16 subkeys.The key schedule for decryption is similar the subkeys are in reverse order compared to encryption. Apart from that change, the process is the same as for encryption. The same 28 bits are passed to all rotation boxes. The 3DES uses encrypt-decrypt-encrypt (EDE) sequence with three keys instead of three times encryption (EEE) with three different keys to preserve compatibility with DES; a hardware circuit that implemented 3DES (with EDE) could also be used to do DES as well (by, say, making all three subkeys the same). A TDEA key consists of three keys for the cryptographic engine (Key1, Key2 and Key3); the three keys are also referred to as a key bundle (KEY). The standards define three keying options:

- Keying option 1: All three keys are independent.

Keying option 1 is the strongest, with 3 56 = 168 independent key bits.

- Keying option 2: K1 and K2 are independent, and K3

= K1. Keying option 2 provides less security, with 2

56 = 112 key bits. This option is stronger than simply DES encrypting twice, e.g. with K1 and K2, because it protects against meet-in-the-middle attacks.

- Keying option 3: All three keys are identical, i.e. K1

= K2 = K3. Keying option 3 is equivalent to DES, with only 56 key bits. This option provides backward compatibility with DES, because the first and second DES operations cancel out.

IV. COMPRESSION

Most representations of information contain large amounts of redundancy. Data compression is a way to reduce storage cost by eliminating redundancies that happen in most files, which is formerly known as source coding;coding done at the source of the data before it is stored or transmitted. It involves the development of a compact representation of information. When data compression is used in a data transmission application, speed is the primary goal. Speed of transmission depends upon the number of bits sent, the time required for the encoder to generate the coded message and the time required for the decoder to recover the original ensemble. In a data storage application, the degree of compression is the primary concern. Thus it is subjected to a space - time complexity trade off.

A. Run Length Coding

Run Length Encoding (RLE) is the simplest of the data compression algorithms. It is a method that allows data compression for information in which symbols are repeated constantly. The method is based on the fact that the repeated symbol can be substituted by a number indicating how many times the symbol is repeated and the symbol itself. It replaces runs of two or more of the same character with a number which represents the length of the run, followed by the original character. Single characters are coded as runs of

1. The major task of this algorithm is to identify the runs of the source file, and to record the symbol and the length of each run. The Run Length Encoding algorithm uses those runs to compress the original source file while keeping all the non-runs without using for the compression process.

It can also be done in such a way that the data bits are covered up by zero bits for transmission. The logic used here is that no voltage is required for the transmission of zeroes, thus reducing the overall power consumption. The algorithm used for both the compressor and decompressor units are as follows:

- Compression

1) Group the data each with four bits.
2) Check whether all the bits are same in a group or not
3) If a group consist of equal bits then send zeros, otherwise send ones
4) Mark the positions of all ones and zeros before transmission

- Decompression

1) Group the data each with four bits
2) Note the positions of ones and zeros
3) Put original values on the position of ones, and place zeros/ones according to the transmitted positions.

V. ERROR DETECTION AND CORRECTION

Transmission of information seems relatively simple. But, in reality, it is a bit trickier. Environmental interference and physical defects in the communication medium can cause random bit errors during data transmission. Error coding is a method of detecting and correcting these errors to ensure information is transferred intact from its source to its destination. Error coding is used for fault tolerant computing in computer memory, magnetic and optical data storage media, satellite and deep space communications,

network communications, cellular telephone networks, and almost any other form of digital data communication.

A. Hamming Codes

Hamming codes are a family of linear error-correcting codes that generalize the Hamming(7,4)-code. Hamming codes can detect up to two-bit errors or correct one-bit errors without detection of uncorrected errors. By contrast, the simple parity code cannot correct errors, and can detect only an odd number of bits in error. Hamming codes are perfect codes, that is, they achieve the highest possible rate for codes with their block length and minimum distance.The algorithm used is as follows:

1) Number the bits starting from 1: bit 1, 2, 3, 4, 5, etc
2) Write the bit numbers in binary: 1, 10, 11, 100, 101, etc
3) All bit positions that are powers of two (have only one 1 bit in the binary form of their position) are parity bits: 1, 2, 4, 8, etc. (1, 10, 100, 1000)
4) All other bit positions, with two or more 1 bits in the binary form of their position, are data bits
5) Each data bit is included in a unique set of 2 or more parity bits, as determined by the binary form of its bit position.

- Parity bit 1 covers all bit positions which have the least significant bit set: bit 1 (the parity bit itself), 3, 5, 7, 9, etc
- Parity bit 2 covers all bit positions which have the second least significant bit set: bit 2 (the parity bit itself), 3, 6, 7, 10, 11, etc.
- Parity bit 4 covers all bit positions which have the third least significant bit set: bits 4-7, 12-15, 20-23, etc. and so on.

VI. RESULTS AND DISCUSSIONS

The simulation output waveform views have been given.The design is simulated by using the simulation tool Xilinx ISE Design Suite 14.5,and waveforms obtained using Modelsim SE 6.5b . Fig 5 shows the encryption of a plain text using 3des algorithm. The message is encrypted using a key and further send for compression. The plain text, which is the input and the cipher text, which is the encrypted output along with the input keys are as shown below:

Plain Text :5CD9F6A0D10E22CO Key1:87EDA98B92642C20
Key2:1234567890BCDEFA Key3:2345678901CDEFAB
Cipher text: 0FFFEEAA12FBC00F

Fig.6 shows the compression of the encrypted data.The cipher text obtained as a result of encryption is subjected to RLE where the input data is classified into a group of four from LSB bits. The postion where all ones and all zeroes are counted and registered using signals eq0 and eq1.These signals make the decompressing process possi- ble.The detected positions enables the user to decompress the data without data losses. a group with equal number of bits are compressed and is replaced by zeroes, which reduce the overall transmission power.

Input Text: 0FFFEEAA12FBC00F Compressed Data: 0000EEAA120BC000

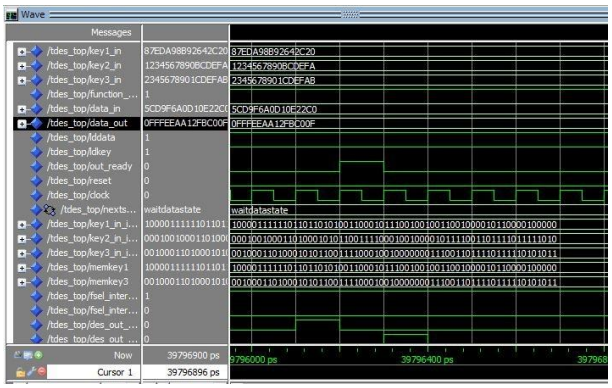

Figure 5. Output waveform of encypted data

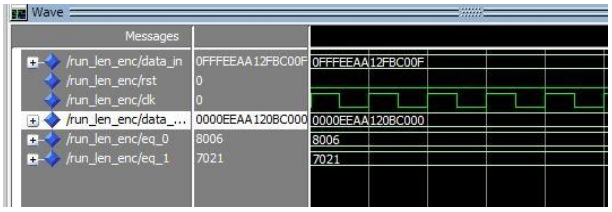

Figure 6. Output waveform of compressed data

Fig 7 shows the performance of an error correction and detection system, when there is no error, which is less probable to occur practically.
Data In: 0000EEAA120BC000
Data Out: 0000EEAA120BC000
Fig 8 shows the performance of EDAC system with single bit error. Here the bit in the 67th position has altered and thus error has occurred.The EDAC system detects the error and its position and corrects the error and is given out.
Bit Reversal :67th bit
Data In : 0000EEAA120BC000
Data Out: 0000EEAA120BC000

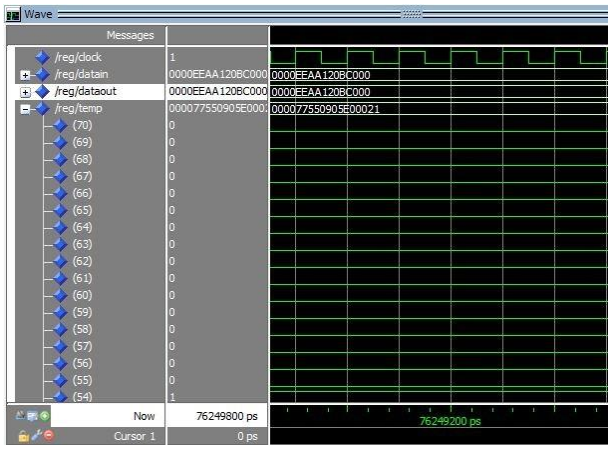

Figure 7. CaseI :Output waveform of EDAC system with no error

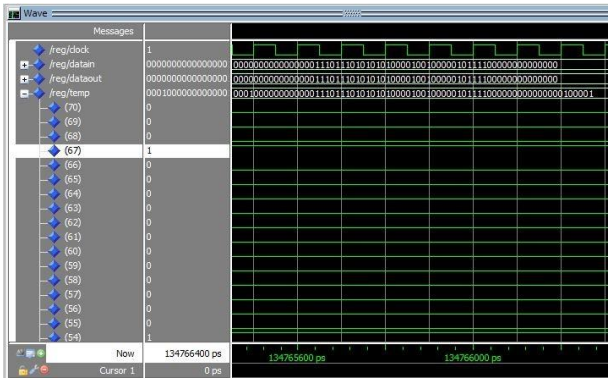

Figure 8. CaseII :Output waveform of EDAC system with single bit error

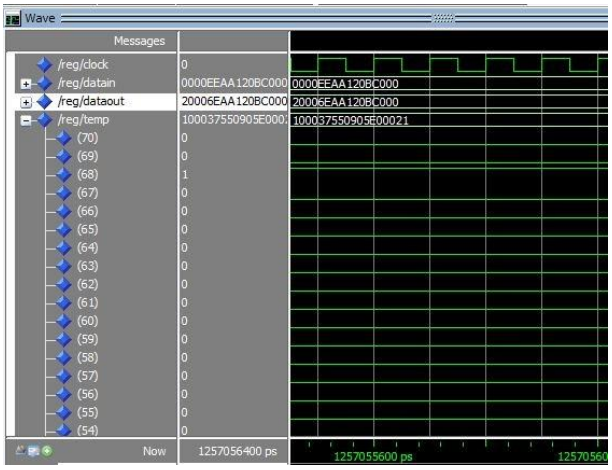

Figure 9. CaseIII :Output waveform of EDAC system with double bit error

Fig.9 shows the performance of EDAC system with double bit error. Here error occurred in 68th and 54th bit by altering its values.The EDAC system detects the error,but the bit reversal in the bit position no:69 and 67 couldn't not be corrected.

Bit Reversal : 68th and 54th bit

Data In : 0000EEAA120BC000

Data Out: 20006EAA120BC000

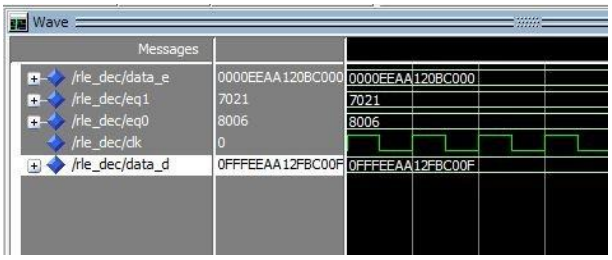

Figure 10. Output waveform of decompressed data

Fig 10 shows the decompresssion of corrected data.The positions of the zeroes,ones and compressed data are the input to this section,the ones are again replaced by zeroes and get back the cipher text.

Compressed Input : 0000EEAA120BC000

Decompressed Output: 0FFFEEAA12FBC00F

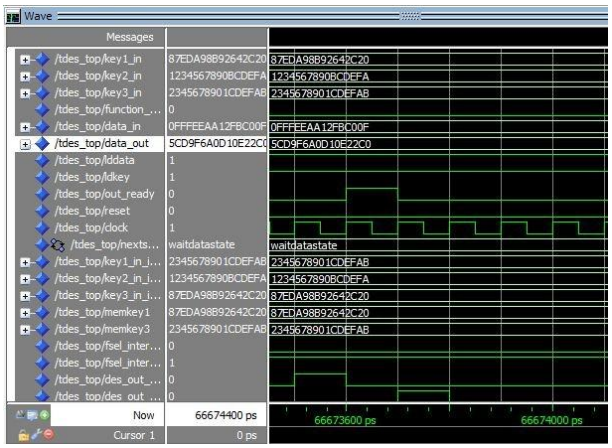

Figure 11. Output waveform of decrypted data

Fig 11 shows the decryption of the cipher text.The output of the decompressor is nothing but the cipher text,this cipher text is decrypted using the same key used for encryption to get back our original message or plain text.The overall working shows how the proposed system work as a whole even if an error encounters.

Cipher Text: 0FFFEEAA12FBC00F Key1:87EDA98B92642C20
Key2:1234567890BCDEFA Key3:2345678901CDEFAB
Plain Text :5CD9F6A0D10E22CO

VII. CONCLUSION

Our modern world teems in communication. In the modern world, security, channel utilization, and error cor- rection are of greater importance when bandwidth limited channels are considered, which demands high security. Se- curity is a prevalant concern in information and data systems of all types. Historically, military and national security issues drove the need for secure communication. One means of providing security is through encryption. In this study a combined encryption and compression scheme is used along with error detection and correction capability to provide a communication system with a better channel utilization and enhanced security. The system is encrypted using triple data encryption scheme and is compressed by run ength coding. For combating bit errors, an improved hamming coding is used. The overall system achieves an improved detection accuracy, better channel utilization and provide extremely secure and authentic communication. The entire system is functionally simulated and verified using Xilinx ISE Design Suite 14.5 and Modelsim SE 6.5b.

REFERENCES

[1] Poornima.P.V, Amrutha.V ´ Security Enhanced Communica- tion Scheme with Error Correction Capability and Efficient Channel Utilization´ , (IJCSIT) International Journal of Com- puter Science and Information Technologies, Vol. 5 (2) , 2014, 2177-2185

[2] Chethan Kumar K V, S Sujatha,´ VLSI Implementation of DES TDES Algorithm with Cipher Block Concept´ , International Journal of Emerging Science and Engineering (IJESE) ISSN:

[3] S.Poongodi M.E,Dr.B.Kalavath,M.Shanmugapriya, ´ Secure
Transformation of Data in Encrypted Image Using Reversible Data hiding Technique ´ , International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 2, Issue 4, July 2013

[4] Anmol Jyot Maan,´ Analysis and Comparison of Algorithms for Lossless Data Compression ´ , International Journal of Information and Computation Technology. ISSN 0974-2239 Volume 3, Number 3 (2013), pp. 139-146

[5] P. Aatheeswaran , Dr.R.Suresh Babu, ´ FPGA can be imple- mented by using Advanced Encryption Standard Algorithm ´ ,International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering Vol. 2, Issue 1, January 2013

[6] Demijan Klinc, Carmit Hazay, Ashish Jagmohan, Hugo Krawczyk, and Tal Rabin,´ On Compression of Data Encrypted With Block Ciphers´ , IEEE Transactions On Information Theory, VOL. 58, NO. 11, NOV 2012
[7] Aqib Al Azad ,´ Efficient VLSI Implementation of DES and Triple DES Algorithm with Cipher Block Chaining concept using Verilog and FPGA´ , International Journal of Computer Applications (0975 8887) Volume 44 No16, April 2012
[8] M.Pitchaiah, Philemon Daniel, Praveen,´ Implementation of Advanced Encryption Standard Algorithm ´ , International Journal of Scientific Engineering Research Volume 3, Issue
3, March -2012
[9] Muthumanickam, Dr. A. Nagappan, T.Sheela, ´ Analysis Of High Performance Vlsi For Telecommunication Data ´ , IRACST Engineering Science and Technology: An Interna- tional Journal (ESTIJ), ISSN: 2250-3498, Vol.2, No.1, 2012
[10] Prof. Atul S. Joshi,Dr. Prashant R. Deshmukh, Prof. Aditi Joshi, ´ Joint Binary Code Compression And Encryption´ [IJE- SAT] International Journal Of Engineering Science Advanced Technology Volume-2, Issue-6, 1643 1647, 2012
[11] G.Aparna, G. Lokeshwari, Dr. Vijaybabu Gorumuchu, ´ En- cryption for Secured ECG Distribution using DES for Medical Applications ´ , International Journal of Wisdom Based Com- puting, Vol. 1 (3), December 2011
[12] T.SubhamastanR ao , M.Soujanya, ´ Simultaneous Data Com- pression and Encryption ´ , (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 2 (5) ,
2011, 2369-2374
[13] Ajit Singh1 and Rimple Gilhotra,´ Data Security Using Pri- vate Key Encryption System Based On Arithmetic Coding ´, International Journal of Network Security Its Applications (IJNSA), Vol.3, No.3, May 2011
[14] William Stalings,´ Cryptography and and network security-A practical Approach´ , Pearson,2011
[15] Behrouz A. Forouzan, Debdeep Mukhopadhyay, ´ Cryptogra- phy and Network Security´ , Tata mcgraw hill education private limited, 2010
[16] Rengarajan Amirtharajan, Vivek Ganesan, R Jithamanyu and John Bosco Balaguru Rayappan, ´ An Invisible Communication for Secret Sharing against Transmission Error ´ , Universal Journal of Computer Science and Engineering Technology 1 (2), 117-121, Nov. 2010
[17] Hamdan.O.Alanazi, B.B.Zaidan, A.A.Zaidan, Hamid A.Jalab, M.Shabbir and Y. Computing, Volume 2, Issue 3, March 2010 , ISSN 2151-9617,pp 152-157
[18] Mark Johnson, Prakash Ishwar, Vinod Prabhakaran, ´ On Compressing Encrypted Data´ , IEEE Transactions On Signal Processing, VOL. 52, NO. 10, OCT 2004
[19] M. Johnson, D. Wagner, and K. Ramchandran, ´ On com- pressing encrypted data without the encryption key,´ presented at the Theory Crypto. Conf., Cambridge, MA, Feb. 2004
[20] Gael Rouvroy,Francois-Xavier Standaert,Jean-Jacques Quisquater, Jean-Didier Legat,´ Efficient Uses of FPGAs for Implementations of DES and Its Experimental Linear Cryptanalysis ´ , IEEE Transactions On Computers, Vol. 52, No. 4, April 2003, pp no. 1 to 10 on the TMS320C6000 ´
,SPRA702 - November 2000
[21] R. Stephen Preissig,´ Data Encryption Standard (DES) Im- plementation on the TMS320C6000 ´ ,Texas Instruments, SPRA702 - November 2000
[22] Amit Dhir, ´ Data Encryption using DES/Triple-DES Func- tionality in Spartan-II FPGAs´ ,Xilinx, WP115 (v1.0) March
9, 2000
[23] Abel-Ghaffer, K.A.S., and Weber G.H. , ´ Analysis of coding scheme for modulation and error control´ , IEEE Transactions on Information Theory, vol 41.,no.6, pp 1955-1968,Nov 1995

