ARM RECOGNITION: ENCRYPTION BY USING AES ALGORITHM

Shraddha Satish Kashid[1], Nagnath Hulle[2]

[1]*Student, Electronics & Telecommunication, G.H.R.C.O.E & M, Ahmadnagar, Maharashtra, India*
[2]*Professor, Electronics & Telecommunication, G.H.R.I.E.T, Pune, Maharashtra, India*

Abstract

To provide the security of the Military confidential data we use encryption algorithm which take over reward of superior encryption algorithm. The proposed implementation using encryption algorithm was implemented on ARM 7 to encrypt and decrypt the confidential data on data storage devices such as SD card or Pen drive. The main objective of proposed implementation is to provide protection for storage devices. The ARM and encryption algorithm protect the data accessibility, reliability and privacy successfully. Since (AES) Advanced Encryption Standard algorithm is widely used in an embedded system or fixed organization. These AES algorithms are used for proper designs in defense for security.

Keywords: Plain text, Cipher text, Data security, AES, Embedded System.ARM, storage device.

----------***----------

1. INTRODUCTION

An approach of cryptography is the best way on our processor based communiqué system instead of conservative systems. Since such type of systems are not viable to contribute a universal mainframe. It is an alternative system to low priced and transportable systems to make sure for communication protection. By the varieties of power driven or motorized an embedded systems be able to ASIC, Digital Signal Processing (DSP) or Microcontroller. There are many trendy embedded work station so as to have a lion's split of the promotes. But the ARM7 is the widely used because of their less power utilization, has less price in market, under sized body shape [1].

A variable input in addition to building block dimension there is a symmetric block cipher called Rijindael. There are two Belgian cryptographers Vincent Rijmen and Joan Daemen. Since, Rijindael made up through two Belgian cryptographers Vincent Rijmen and Joan Daemen. In the year 2001, National Institute of Standards and Technology (NIST) won the Advanced Encryption Standard collection, which is known as AES. According to NIST the assortment of AES be completed on the platform as security, performance, efficiency, flexibility, and put into practice aptitude. In November2001, the elected algorithm, viz., Advanced Encryption Standard (AES), have replace DES and published as FIPS197 [2].
AES is demanding in implementation of Hardware and Software. Implementation of Hardware is ordinary into elevated rate function [3]. For embedded system application most required or highly popular implementation is Hardware implementation. The performance of software implementation is not so faithfully fast, it is so sluggish and taking long processor time, that becomes the system more leisurely.

For AES implementation purpose in market there are various hardware platforms are available which is based on mostly processors or controllers among 8-bit terms to16-bit terms. By using a large bit processor or controller for the implementation of AES becomes a system more exclusive with high ending operations.[4] whereas implantation by using 8-bit processors/ controllers are very sophisticated and used for minimum cost and for low end applications. By using KEIL as a compiler tool on ARM processor LPC2148 is we proposed in this paper. The processor which we are using in this implementation is also capable of time measurement, means the time necessary for implementation of key opening out for various cycles of AES is deliberated. The time required for Encryption and Decryption is also calculated by ARM processor. Embedded C is using for coding of AES.

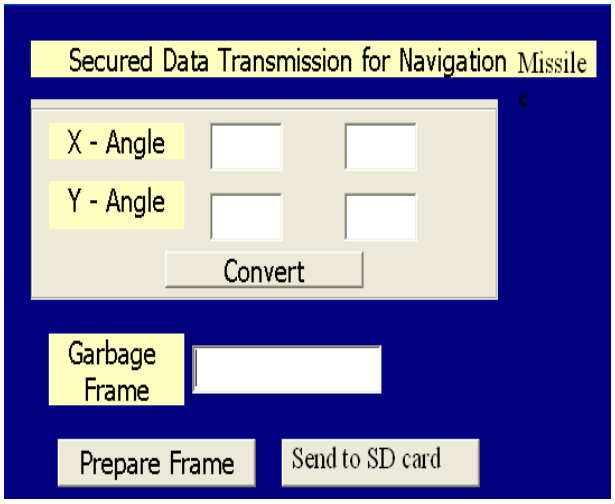

Fig.1 General View of Secured Data Navigation

2. RELATED WORK

The algorithm is flexible in supporting any combination of data and key size of 128, 192, and 256 bits. However, AES merely allows a 128 bit data length that can be divided into four basic operation blocks. These blocks operate on array of bytes and organized as a 4×4 matrix that is called the state. As AES has been invented according to investigation by Rijndael. In the literature many researchers used different approaches have been used for the implementation on the basis of different technical acceptations which may be like AES strength execution, AES for efficiency or effectiveness execution, AES by hardware and software implementation and all that. This section referred to ideal standard identifications on the target for evaluation achievement stages of AES optimization.

Ashruf.et.al in [7] have implemented AES in Molen hardware like Field Programmable Gate Array (FPGA) and General Purpose Processors (GPP) merging together to form Molen processor. The processor by using Molen hardware implementation architecture gives result as fast as poor FPGA. FPGA has highest velocity and more flexibility. The implementation of Molen hardware is only depends on its size rather than its cost.

The researcher T.Ravichandra Babu [9] implemented AES algorithm onto ARM processor platform. According to this implementation, shift Rows and Substitute byte (sub byte) operations is implemented on software. Whereas, on hardware Add Round Key operation and Mix Column operations performed. Therefore, Hardware and Software execution of AES have completed.

K.Atasu et al [11] include memory and speed efficiency optimized AES implementation. They open for us the Mix Column concept by focusing speed optimization on linear integration component. They proposed a new combine approach in this paper. It uses standard approach for the encryption [2] and the transposed approach [8] for decryption. That gives an excellent performance than the uncontaminated approach of standard and transposed.

3. AES ALGORITHM

All Rijndael was deliberate to contain the subsequent characteristics

- Resistance alongside every recognized attack.
- Speed and code density on a extensive series of platform.
- Devise effortlessness.

AES is a symmetric building block secret message by means of block measurement lengthwise of 128 bits. It allows three different key lengths 128,192 and 256 bits.

In encryption process; for processing of 128 bit keys required 10rounds, 192 bit keys required 12 rounds and 256 bit keys required 14 rounds. AES is a in a circle based algorithm. For encryption and decryption, every series have four functions apart from previous surrounding (Last round necessary three functions).

The encryption algorithms have four encircling functions Sub Byte, Shift Rows, Mix Column (misplaced in last round) and Add Round Key.

The decryption furthermore have the identical numeral of rounds through overturn conversion, sort of in circles purpose is diverse i.e. InvShiftRow, InvSubByte, AddRoundKey and InvMixColumn (absent in last round) [3][4]. TABLE I shows number of AES parameters for the accepted three AES versions.

Table 1: AES Parameters

PARAMETERS	AES-256	AES-192	AES-128
Number of rounds	14	12	10
Plaintext box size(Bits)	128	128	128
Key size(Bits)	256	192	128

According to Rijndaels research they found very big criteria using very low memory aspects in a constrained space location, across a wide range computing surrounding hardware and software. Rijndael is a superior victor during incorporation hardware and software do not having any feed backing or non- feed backing surroundings.[6] against energy and timing. Initially we transferred the input bit data in byte sequence at sender side. Then in next pace built a array of the two dimensional bytes. Which is also known as it's State. This array having four bytes rows individually. All Cipher and Inverse Cipher operations also mean the International operations are performed in a state array of the AES algorithm. Essentially, AES algorithm leaning four byte transformation at sender side (for encryption of data) and opposite transformation (inverse) at receiver side (for decryption of data) respectively

a) (S-box): Byte replacement using substitution box table.
b) (Row transformation): Shifting rows of the state array using dissimilar offsets.
c) (Mixing columns): Mixing all the data inside every column of the state array.
d) (Add round key): Adding a round key to the state.

The fig.1 shows AES Rijndael Encryption and Decryption structure, where the input to the encryption and decryption algorithm is 128 bit block. The key provided is expanded into an array of forty - four 32 bit words, w[i].

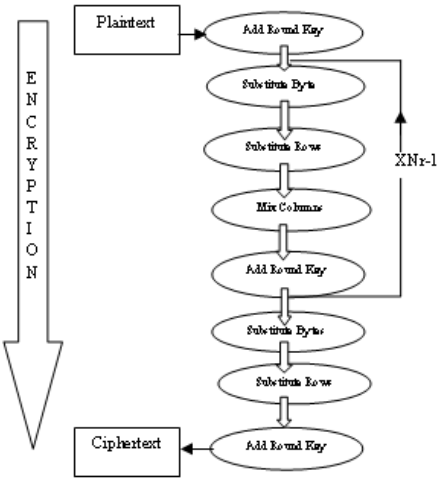

Fig 2: Encryption Side

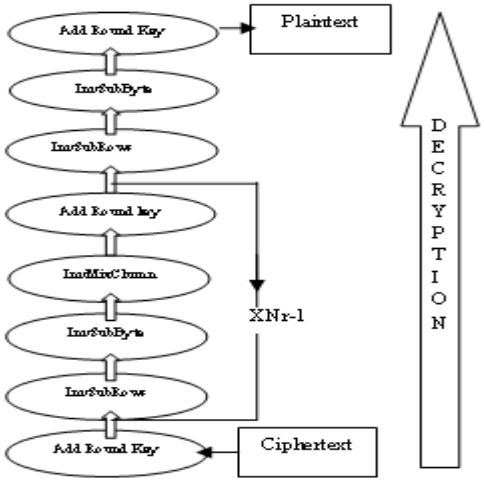

Fig 3: Decryption Side

4. PROPOSED MODEL SYSTEM

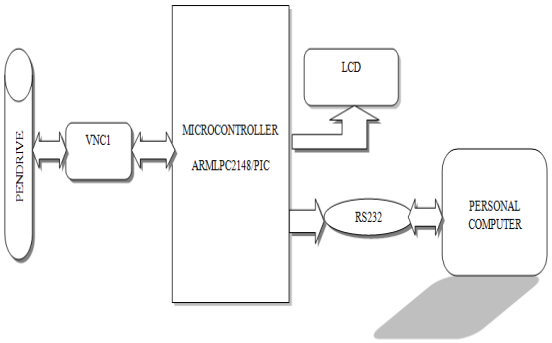

Fig 4: Encoder System

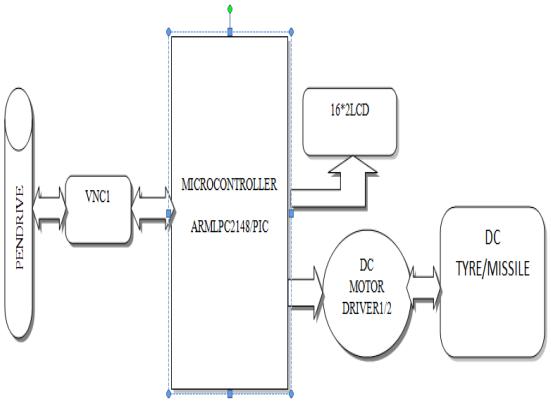

Fig 5: Decoder System

As the name of the project suggests we are making an encryption and decryption system for military application.

The project consists of mainly 2 parts:

4.1 Encoder Unit

This section consists of a master pc terminal on which we are developing our own vb s/w. The vb s/w is used to encode the frame and send the encoded frame to the master Microcontroller. The master Microcontroller then stores this encoded frame to the SD card in a text format.

4.2 Interfacing of ARM LPC 2148 Board and Computer

The RS232 port of computer is connected to RS232 Port0 of ARM LPC2148 boards. The AES encryption and decryption code is dumped onto ARM through RS232 port by using a flash burner called Philips flash utility V2.2.3.

4.3 Decoder Unit

In this section the user has to insert the SD card into the SD card slot connected to the Microcontroller. The Microcontroller then reads the encode frames from SD card and decodes the frame by applying the AES algorithm. Then finally the encoded frame is displayed on LCD. The decoder Microcontroller decodes the no of steps the motor is supposed to move after the decoding is over the decoder unit Microcontroller turns the motor based missile model giving us the exact position of missile.

This system basically consists of SD card interface, LCD display and the dc motor for missile navigation.

4.4 Interfacing of ARM LPC 2148 Board and Pen Drive.

The RS232 port0 of ARM LPC 2148 is connected to RS232 port of Pen drive. The SD card is controlled by ARM through AT commands. The important AT commands are as follows:

- AT- It used to check communication between Pen drive and ARM
- ATE0-Command Echo
- AT+CMGF-This command is used to set the SMS mode. Either text or PDU mode can be selected by assigning 1 or 0 in the command
- AT+CMGS-This command sends a short message from the modem to the network
- AT+CMGR-Read message
- AT+CMGD-This command deletes a message from the location from SIM storage.

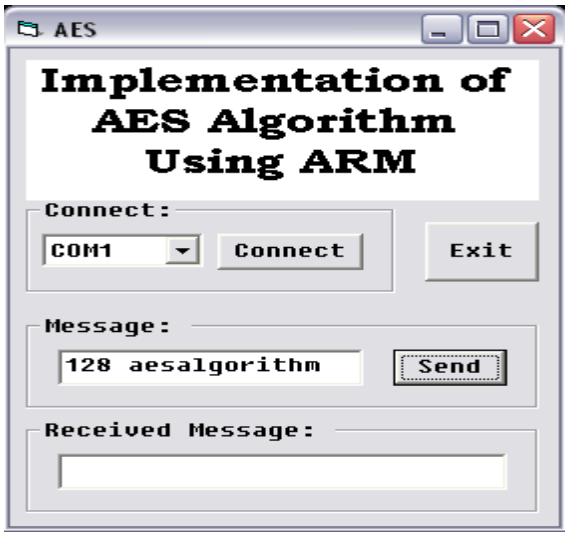

Fig.6: Plain Text during Encryption

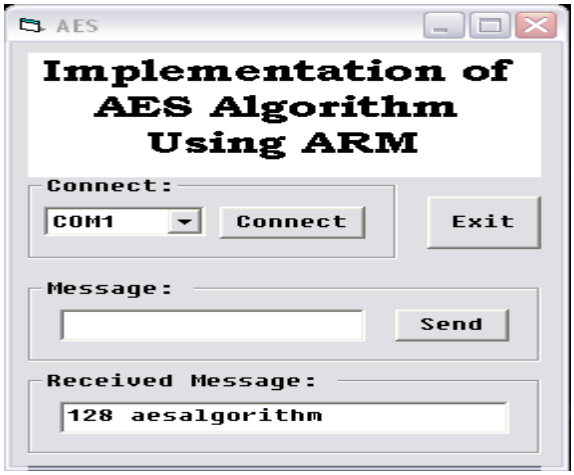

Fig.7: Plaintext during Decryption

Here we write the GUI code in visual basic. Which relatively contain two windows namely sending window and receiving window respectively? Above fig.5 shows the data of any kind e.g. '128 AES algorithm' which is send during encryption. After sending this data by the sender at the receiving end this encrypted data is converted through plain text by AES decryption. At sender side the encrypted data LCD shows the cipher text message while at receiver side the LCD shows the plaintext message at the ending.

5. CONCLUSIONS

Implementation of ARM Recognition Encryption By Using AES Algorithm on embedded platform is presented in this paper. According to the previous research work the researcher was used the the separate memory for hardware and software. For the implementation of shift rows and substitute byte on software and add round keys and mixed column operations on hardware. But in this proposed work we implemented both on hardware. That increases the speed and effective area and decreases time for implementation. That makes the system more reliable and reducing its cost.

ACKNOWLEDGEMENTS

I would like to thank all mighty for the successful completion and moreover the teaching staff of the college for their persistence in keeping me on schedule and quality. I am thankful to my seminar guide Prof. Nagnath Hulle, for his active involvement and guidance throughout the seminar work. I would like to thank many other individuals from department, including our respected M.E coordinator Prof.V.Bhope who contributed greatly to this seminar work and provided us all the proper facilities. I would also like to thank our respected principal sir for providing us good infrastructure and all amenities. Sincere thanks to the management and the lab attendants for their full cooperation throughout the seminar work. Last but not the least my friends and my family for their continuous support and encouragement.

If I forget to give thanks as well as if I forget to mention reference name of anybody in reference list I apologies for that.

REFERENCES

[1]. N. Sloss, D. Symes, and C. Wright, ARM System Developer's Guide, Designing and Optimizing System Software, Morgan Kaufmann, 2004.
[2]. Journal of research of the NIST, volume 106, November 3, May- June 2001
[3]. NIST, Advanced Encryption Standard (AES), (FIP PUB 197), November 26, 2001.
[4]. J. Daemen and V. Rijmen, AES Proposal: Rijndael (Version 2). NIST AES

[5]. UM10120 LPC2131/2/4/6/8 *User manual* File Format: PDF/Adobe AcrobatNumerous editorial updated throughout the *user manual*. 02. 2006.09.18. Updated edition of the *User Manual* covering both *LPC213x* and *LPC213x*/01 devices. For *www.nxp.com/documents/user manual/UM10120.pdf*

[6]. M. McLoone, J. McCanny, "High Performance Single-Chip FPGA Rijndael Algorithm Implementations," Proceedings Cryptographic Hardware and Embedded Systems Workshop, CHES, Paris, May 2001

[7]. R. Ashruf et al, Reconfigurable Implementation for the AES Algorithm, Delft University of Technology, Netherlands, 2005.

[8]. G. Bertoni, L. Breveglieri, P. Fragneto, M. Macchetti and S. Marchesin, "Efficient Software Implementation of AES on 32-bit Platforms," CHES 2002, LNCS 2523, pp. 159–171, 2003.

[9]. T.Ravichandra Babu, K.V.V.S.Murthy, G.Sunil , "AES Algorithm Implementation using ARM Processor",2nd International Conference and workshop on Emerging Trends in Technology (ICWET) 2011.

[10]. B. Gladmans, A specification for Rijndael, the AES Algorithm. Available at http://fp.gladman.plus.com, May 2002.

[11]. K. Atasu et al, Efficient AES Implementation for ARM Based Platforms, ACM, 2004 Philips LPC2131, LPC2132, LPC2134, LPC2136, LPC2138 Data Sheet.