IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 15, Issue 5 (Nov. - Dec. 2013), PP 35-38
www.iosrjournals.org

Implementation of AES Algorithm in MicroController Using PIC18F452

Ali E. Taki El_Deen[1] and Ahmed Mohamed Fanni[2]
[1]*IEEE Senior Member, Alexandria University, Egypt*
[2]*Electronics and Communications Dept, Mansoura University,Egypt*

Abstract: *Security has become an increasingly important feature with the growth of electronic communication which calls for more advanced ways to encrypt the raw data[1]AES-128 is going to be implemented as the encryption algorithm as there are 3 types of AES (AES-128,AES-192,AES-256) but why AES? Because AES is famous for its ciphering strength and its strong defense against any attack for example the brute force attack. Using Brute force attack on AES-128 (smallest key length) is unlikely to be practical in the foreseeable future [3]. According to NIST, "Assuming that one could build a machine that could recover a DES key (DES is an old encryption algorithm which no one uses now in modern days) in a second (i.e.,try 2^55 keys per second), then it would take that machine approximately 149 thousand-billion (149 trillion) years to crack a 128-bit AES key". In this paper AES-128 will be implemented on a microcontroller circuit to give it further security and more speed in both the encryption and decryption of the files.*
Keywords: *AES, Cryptography, DES, NIST, Rijndael, Serpent.*

I. Introduction:

Cryptography, the use of codes and ciphers to protect secrets, began thousands of years ago. Until recent decades, it has been the story of what might be called classic cryptography — that is, of methods of encryption that use pen and paper, or perhaps simple mechanical aids. In the early 20th century, the invention of complex mechanical and electromechanical machines, such as the Enigma rotor machine, provided more sophisticated and efficient means of encryption; and the subsequent introduction of electronics and computing has allowed elaborate schemes of still greater complexity, most of which are entirely unsuited to pen and paper[1].

The development of cryptography has been paralleled by the development of cryptanalysis — the "breaking" of codes and ciphers. The discovery and application, early on, of frequency analysis to the reading of encrypted communications has, on occasion, altered the course of history. Thus the Zimmermann Telegram triggered the United States' entry into World War I; and Allied reading of Nazi Germany's ciphers shortened World War II, in some evaluations by as much as two years [4].

Before the modern era, cryptography was concerned solely with message confidentiality (i.e., encryption)—conversion of messages from a comprehensible form into an incomprehensible one and back again at the other end, rendering it unreadable by interceptors or eavesdroppers without secret knowledge (namely the key needed for decryption of that message). Encryption was used to (attempt to) ensure secrecy in communications, such as those of spies, military leaders, and diplomats [7]. In recent decades, the field has expanded beyond confidentiality concerns to include techniques for message integrity checking, sender/receiver identity authentication, digital signatures, interactive proofs and secure computation, among others [5].

Towards the close of the 20th century, the National Institute for Standards and Testing (NIST)acted on the need for a new encryption algorithm capable of protecting top secret information[2].
NIST is part of the Department of Commerce. It is a non-regulatory agency that, promotes "U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology in ways that enhance economic security and improve quality of life.

"Early in the development process, NIST decided to draw on the world's finest cryptographic minds and asked them to submit candidates for the new algorithm because the aging Data Encryption Standard (DES) has many weaknesses. DES has a relatively small 56-bit key which was becoming vulnerable to brute force attacks. In addition, the DES was designed primarily for hardware and is relatively slow when implemented in software. While Triple-DES avoids the problem of a small key size, it is very slow even in hardware; it is unsuitable for limited-resource platforms; and it may be affected by potential security issues connected with the (today comparatively small) block size of 64 bits. In 1997 NIST published a formal call which read in part: It is intended that the AES will specify an unclassified, publicly disclosed encryption algorithm available royalty-free worldwide that is capable of protecting sensitive government information well into the next century.

The purpose of this notice is to solicit candidate algorithms from the public, academic/research communities, manufacturers, voluntary standards organizations, and Federal, state, and local government organizations. Following the close of the submission period, NIST intends to make all submissions publicly available for review and comment.

The entire process spanned five years. Fifteen competing algorithms with colorful names such as Rijndael (the eventual winner), Twofish and Serpent (the runners up) were scrutinized over a three year period [8].

AES is now the industry standard for encryption. The NSA employs it for protecting secret information and industry uses the algorithm for creating commercially available encryption products[3].

File encryption and email encryption are two common applications for AES. File encryption protects the information on your hard disk or thumb drive. With encryption, your data will be secure even if your computer is hacked or your USB drive stolen. Email encryption protects your messages as they journey through the cloud and keeps them from being read by unintended recipients.

II. AES algorithm overview:

AES Algorithm consists of 2 Main Parts:

1- Encryption or Decryption Process:

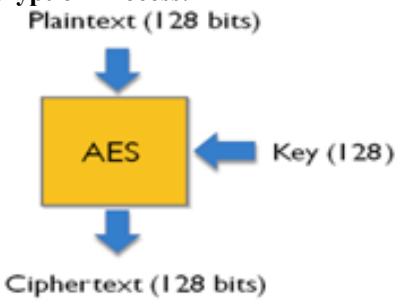

Fig .1 Simple diagram of encryption process

Encryption process contains 4 main parts:

I-Add Round Key:Each byte of the state is combined with the round key using bitwise Xor.

II-Shift Row: transposition step

III-Sub Byte: non-linear substitution step using Sbox

IIII-Mix Column: mixing operation of each column using GF

Encryption process: It starts with AddRoundKey with RoundKey0.

Then go to loop and do SubByte, ShiftRow,MixColumn and AddRoundKey in that order for 9 Rounds each round with different RoundKey(1-9).

Then go to the final round (Round10) and repeat the same previous functions in the loop except MixColumn.

Decryption process: it's the reverse of encryption process in every step which means the decryption 1st round is the 10th round of the encryption and it uses the inverse functions of MixColumn, SubByte and ShiftRow and as u can assume the Keys arrangement are reversed too as it starts with Roundkey10 instead of Roundkey0 as it was in the encryption process [8] [3].

2-Key Generation (Key Expansion):

It involves RotWord, SubByte And Xor bitwise operation to generate enough keys for each round in the Encryption, Decryption process.

Summary of both encryption and decryption process are shown in the Fig. 2.

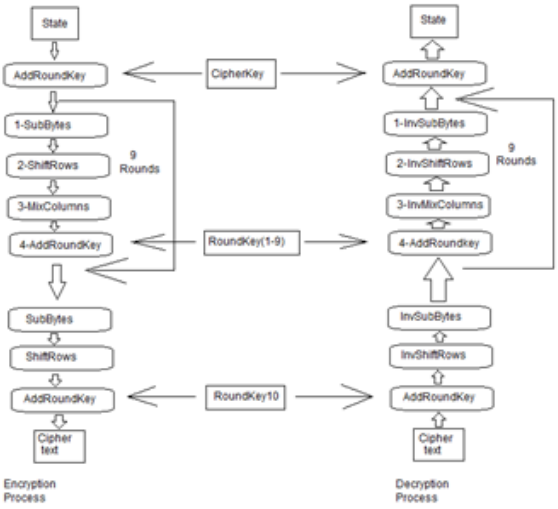

Fig. 2 Encryption and decryption steps

As each round works with different key generated from the key generation process.

III. Implementation of AES Software and Hardware

AES-128 has been implemented on PIC18f452 using MIKROC PRO FOR PIC and interfaced it with Keypad, Alphanumeric LCD 4x20, leds, buttons and Xbee Module for wireless communication as shown in fig. 3.

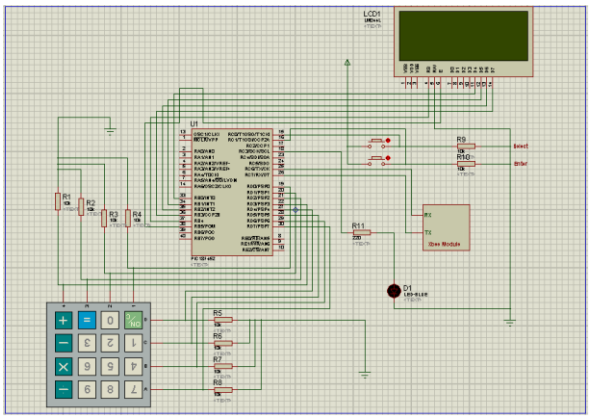

Fig. 3 Schematic Diagram of the device used

PIC18F452 pin diagram [6] is shown in fig. 4.

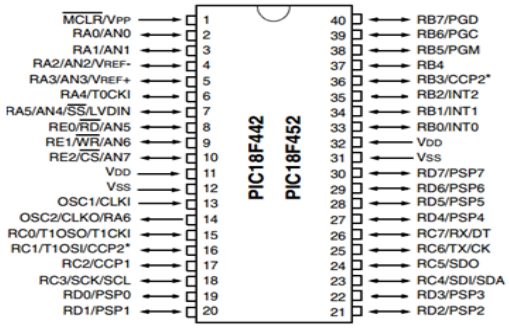

Fig. 4 Pin diagram on PIC18F452

Keypad is used as an input device to enter the Plaintext, LCD is used to display both encrypted and decrypted text, Xbee Module is used to send or receive the encrypted file from or to another exact same device, Leds is used to indicate that there is text being received or the text is done being transmitted. The device can either receive or send the encrypted text to another same device.

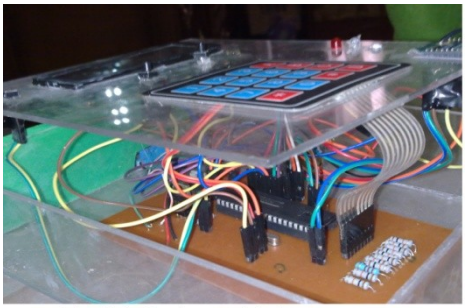

Fig. 5 The Implemented device

Overview on the code

1-SubByte

```
// sub byte
for(Aj=0;Aj<=3;Aj++){
for(Ai=0;Ai<=3;Ai++){
Ax=ACT[Ai][Aj] & 0xf;
Ay=ACT[Ai][Aj]>>4 & 0xf;
ACT[Ai][Aj]=Asbox[Ay][Ax];
ACT[Ai][Aj]=ACT[Ai][Aj];
}
}
```

2-ShiftRow

```
// Shift Row

Act[1][0] = Actt[1][0]; Act[1][0] = Actt[1][1]; Act[1][1] = Actt[1][2]; Act[1][2] = Actt[1][3];
Act[2][2] = Actt[2][0]; Act[2][0] = Actt[2][1]; Act[2][1] = Actt[2][2]; Act[2][2] = Actt[2][3];
Act[3][1] = Actt[3][0]; Act[3][2] = Actt[3][1]; Act[3][3] = Actt[3][2]; Act[3][0] = Actt[3][3];
```

3-MixColumn

```
// Mix Column
for(Aj=0;Aj<=3;Aj++){
ACT[0][Aj]=AS2[(ACT[0][Aj]>>4 & 0xf)][(ACT[0][Aj] & 0xf)]^As3[(ACT[1][Aj]>>4 & 0xf)][(ACT[1][Aj] & 0xf)]^ACT[2][Aj]^ACT[3][Aj];
ACT[1][Aj]=ACT[0][Aj]^AS2[(ACT[1][Aj]>>4 & 0xf)][(ACT[1][Aj] & 0xf)]^As3[(ACT[2][Aj]>>4 & 0xf)][(ACT[2][Aj] & 0xf)]^ACT[3][Aj];
ACT[2][Aj]=ACT[0][Aj]^ACT[1][Aj]^AS2[(ACT[2][Aj]>>4 & 0xf)][(ACT[2][Aj] & 0xf)]^As3[(ACT[3][Aj]>>4 & 0xf)][(ACT[3][Aj] & 0xf)];
ACT[3][Aj]=As3[(ACT[0][Aj]>>4 & 0xf)][(ACT[0][Aj] & 0xf)]^ACT[1][Aj]^ACT[2][Aj]^AS2[(ACT[3][Aj]>>4 & 0xf)][(ACT[3][Aj] & 0xf)];
};
```

4-AddRoundKey

```
// Add Round
for (Ai = Am; Ai < Am+4; Ai++){
for (Aj = Am; Aj < Am+4; Aj++){
ACT[Aj-Am][Ai-Am] = ACTT[Aj-Am][Ai-Am] ^ AKey_g[Aj-Am][Ai];
}
}
```

IV. Advantage of Implementation

You will need to have the device to be able to decrypt the encrypted file because it uses custom Sbox instead of the standard one to ensure more security and the security fuses in the PIC can be enabled so the PIC won't be readable to protect the code hence protect the algorithm so even if the encrypted message got hacked by any means in order to break the encrypted file you will need to have the device first and then guess the key which is practically impossible.

V. Conclusion

In this paper the implementation of AES-128 has given it more encryption power and enhanced its security even more thus make it harder for anyone to hack the ciphered information and decrypted it.
The device components are simple and cheap in price so the cost isn't a problem at all.

REFERENCES

[1] Hans Delfs, Helmut Knebl, “Introduction to Cryptography: Principles and Applications”, Second Edition, ISBN: 9783540492436, 2007.
[2] Richard A. Mollin, “An Introduction to Cryptography”, Second Edition, ISBN: 1584886188 / 9781584886181, 2005.
[3] AviKak, “AES: The Advanced Encryption Standard, Lecture Notes on “Computer and Network Security””, February, 2013.
[4] Christophe RITZENTHALER, “Cryptology course”, 2nd Semester 2006.
[5] ChristofPaar, Jan Pelzl, “Understanding Cryptography”, ISBN: 9783642041006, 2010
[6] Microchip. 2006. PIC18F452 Data Sheet. Printed by Microchip Technology, Inc in the United States of America.
[7] Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone,“Hand Book of Applied Cryptography”, 1997.
[8] AviKak, “AES: The Advanced Encryption Standard, Lecture Notes on “Computer and Network Security””, February, 2013.