

University of Pristina  
Faculty of Electric and Computer Engineering  
Department of Computer Engineering

Efficient Implementation of the AES

**Mentor: Students:**

**Prof.Asoc.Dr.Blerim Rexha Armend Ukëhaxhaj**

**Asistent: Fjollë Krasniqi**

**M.Sc. Petrit Rama Miranda Vranovci**

**Contents**

[Abstract 2](#_Toc529190846)

[Introduction 2](#_Toc529190847)

[AES Encryption 3](#_Toc529190848)

[Confusion layer 3](#_Toc529190849)

[AddRoundKey 3](#_Toc529190850)

[Substitution layer 4](#_Toc529190851)

[SubBytes 4](#_Toc529190852)

[Mathematical description of the S-Box 5](#_Toc529190853)

[Diffusion Layer 5](#_Toc529190854)

[ShiftRows 5](#_Toc529190855)

[MixColumn 6](#_Toc529190856)

[Decryption in AES 6](#_Toc529190857)

# 

# Abstract

In today’s world most of the communication is done using electronic media. Data Security plays a vital role in such communication. Hence, there is a need to protect data from malicious attacks. Cryptography is the science of secret codes, enabling the confidentiality of communication through an insecure channel. It protects against unauthorized parties by preventing unauthorized alteration of use. Generally speaking, it uses a cryptographic system to transform a plaintext into a cipher text, using most of the time a key. Advanced Encryption Standard (AES), also known as Rijndael, is an encryption standard used for securing information. AES is a block cipher algorithm that has been analyzed extensively and is now used widely. AES is a symmetric block cipher that is intended to replace DES as the approved standard for a wide range of applications. The block cipher Rijnddael was designed by Dr. Joan Daemen and Dr. Vincent Rijmen and the name of the algorithm is a combination of the names of its two creators. Rijndael is very secure and has no known weakness.

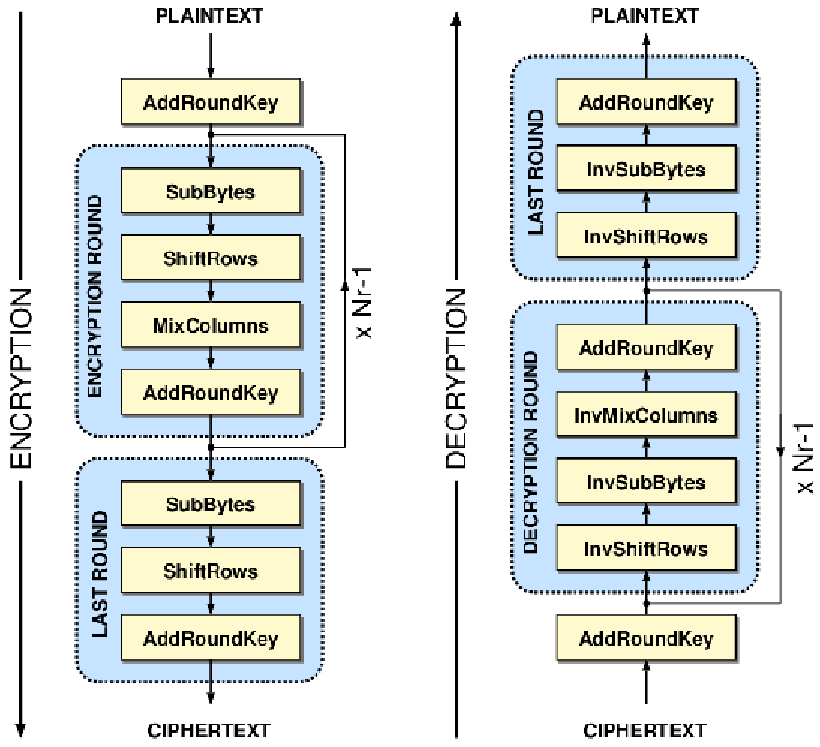
# Introduction

The Advanced Encryption Standard (AES) was specified in 2001 by the National Institute of Standards and Technology.

The purpose was to provide a standard algorithm for encryption, strong enough to keep U.S. government documents secure for at least the next 20 years. The earlier Data Encryption Standard (DES) had been rendered insecure by advances in computing power, and was effectively replaced by triple-DES. Now AES will largely replace triple-DES for government use, and will likely become widely adopted for a variety of encryption needs, such as secure transactions via the Internet.

# AES Encryption

The encryption phase of AES can be broken into three phases: the initial round, the main rounds, and the final round. All of the phases use the same sub-operations in different combinations as follows:

* ****Initial Round**

1. *AddRoundKey*

* **Main Rounds**

1. *SubBytes*
2. *ShiftRows*
3. *MixColumns*
4. *AddRoundKey*

* **Final Round**

1. *SubBytes*
2. *ShiftRows*
3. *AddRoundKey*

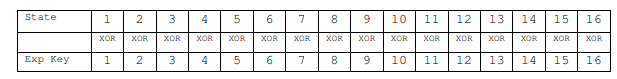
The main rounds of AES are repeated a set number of times for each variant of AES. AES-128 uses 9 iterations of the main round, AES-192 uses 11, and AES-256 uses 13.

## Confusion layer

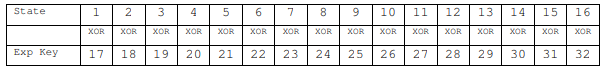
### AddRoundKey

The AddRoundKey operation is the only phase of AES encryption that directly operates on the AES round key. In this operation, the input to the round is exclusive-ored with the round key. Each of the 16 bytes of the state is XORed against each of the 16 bytes of a portion of the expanded key for the current round. The Expanded Key bytes are never reused. So once the first 16 bytes are XORed against the first 16 bytes of the expanded key then the expanded key bytes 1-16 are never used again. The next time the Add Round Key function is called bytes 17-32 are XORed against the state.

The first time Add Round Key gets executed:



The second time Add Round Key is executed:

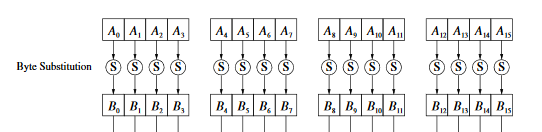


## Substitution layer

### SubBytes

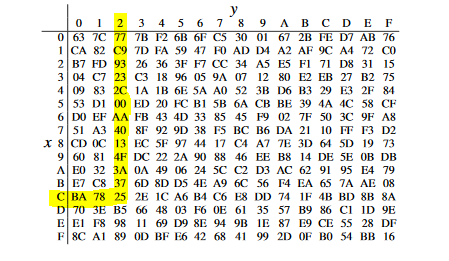
The first layer in each round is the Byte Substitution layer. The Byte Substitution layer can be viewed as a row of 16 parallel S-Boxes, each with 8 input and output bits. All 16 S-Boxes are identical, unlike DES where eight different S-Boxes are used. In the layer, each state byte Ai is substituted, by another byte Bi:

S(Ai)=Bi



The S-Box is the only nonlinear element of AES, i.e., it holds that ByteSub (A)+ByteSub(B) ≠ByteSub(A+B) for two states A and B.

The S-Box substitution is a bijective mapping, each of the 28=256 possible input elements is one-to-one mapped to one output element. This allows us to uniquely reverse the S-Box, which is needed for decryption. In software implementations the S-Box is usually realized as a 256-by-8 bit lookup table with fixed entries



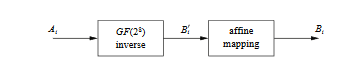
S((C2)hex)=(25)hex

On a bit level —the only thing that is ultimate of interest in encryption is the manipulation of bits — this substitution can be described as:

S(1100 0010)=(0010 0101)

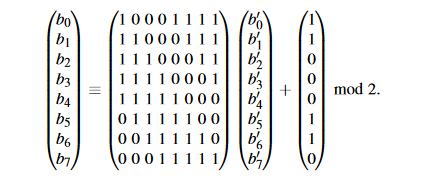
### Mathematical description of the S-Box

The AES S-Boxes have a strong algebraic structure. An AES S-Box can be viewed as a two-step mathematical transformation. Bi=S(Ai)



For each input element Ai, the inverse is computed: B′i=Ai−1, where both Ai and B′i are considered elements in the field GF(28) with the fixed irreducible polynomial P(x)=x8+x4+x3+x+1

The inverse of the zero element does not exist. However, for AES it is defined that the zero element Ai=0 is mapped to itself.In the second part of the substitution, each byte B′I is multiplied by a constant bit- matrix followed by the addition of a constant 8-bit vector. The operation is described by:

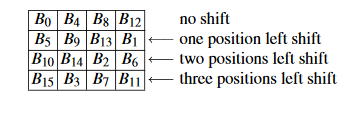
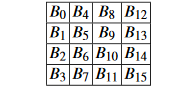


The advantage of using inversion in GF(28) as the core function of the Byte Substitution layer is that it provides a high degree of nonlinearity, which in turn provides optimum protection against some of the strongest known analytical attacks. The affine step “destroys” the algebraic structure of the Galois field, which in turn is needed to prevent attacks that would exploit the finite field inversion.

## Diffusion Layer

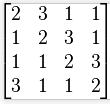
### ShiftRows

The ShiftRows transformation cyclically shifts the second row of the state matrix by three bytes to the right, the third row by two bytes to the right and the fourth row by one byte to the right. The first row is not changed by the ShiftRows transformation. The purpose of the ShiftRows transformation is to increase the diffusion properties of AES. If the input of the ShiftRows sublayer is given as a state matrix, the process goes in this way:

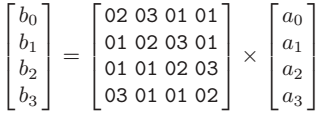


### MixColumn

In the MixColumns step, the four bytes of each column of the state are combined using an invertible linear transformation. The MixColumns function takes four bytes as input and outputs four bytes, where each input byte affects all four output bytes. Together with ShiftRows, MixColumns provides diffusion in the cipher.  
During this operation, each column is multiplied by the known matrix that for the 128 bit key it looks like this:



The multiplication operation is defined as: multiplication by 1 means no change, multiplication by 2 means shifting to the left, and multiplication by 3 means shifting to the left and then performing xor with the initial unshifted value. After shifting, a conditional xor with 0x1B should be performed if the shifted value is larger than 0xFF.



# Decryption in AES

To decrypt an AES-encrypted ciphertext, it is necessary to undo each stage of the encryption operation in the reverse order in which they were applied. The three stage of decryption are as follows:

* **Inverse Final Round**

1. *AddRoundKey*
2. *ShiftRows*
3. *SubBytes*

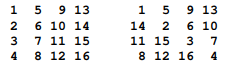
* **Inverse Main Round**

1. *AddRoundKey*
2. *MixColumns*
3. *ShiftRows*
4. *SubBytes*

* **Inverse Initial Round**

1. *AddRoundKey*

Of the four operations in AES encryption, only the AddRoundKey operation is its own inverse (since it is an exclusive-or). To undo AddRoundKey, it is only necessary to expand the entire AES key schedule (identically to encryption) and then use the appropriate key in the exclusive-or. The other revers operation is ShiftRows which is similar to encryption, only rotations or shifting bytes in rows are made to the right and not to the left.



The other decryption operation is SubBytes, whose logic is the same as during encryption. While the latest decryption operation is MixColumns (which is similar to the MixColumns method of encryption), only that the matrix for multiplication differs.

