

AES

Généré par Doxygen 1.9.1

1 Liste des bogues	1
2 Index des fichiers	3
2.1 Liste des fichiers	3
3 Documentation des fichiers	5
3.1 Référence du fichier c/aes.c	5
3.1.1 Description détaillée	5
3.1.2 Documentation des fonctions	6
3.1.2.1 aes_decrypt()	6
3.1.2.2 aes_encrypt()	6
3.1.2.3 asciitohex()	7
3.1.2.4 hextoascii()	7
3.1.2.5 keyprocess()	7
3.2 Référence du fichier c/aes.h	8
3.2.1 Description détaillée	8
3.2.2 Documentation des définitions de type	8
3.2.2.1 byte	9
3.2.3 Documentation des fonctions	9
3.2.3.1 aes_decrypt()	9
3.2.3.2 aes_encrypt()	9
3.2.3.3 asciitohex()	10
3.2.3.4 hextoascii()	10
3.2.3.5 keyprocess()	10
3.3 Référence du fichier c/cipher.c	11
3.3.1 Description détaillée	12
3.3.2 Documentation des fonctions	12
3.3.2.1 addRoundKey()	12
3.3.2.2 cipher()	12
3.3.2.3 invCipher()	13
3.3.2.4 invShiftRows()	13
3.3.2.5 keyExpansion()	14
3.3.2.6 mixColumns()	14
3.3.2.7 rcon()	15
3.3.2.8 rotWord()	15
3.3.2.9 shiftOneRow()	15
3.3.2.10 shiftRows()	16
3.3.2.11 subBytes()	16
3.3.2.12 subWord()	17
3.4 Référence du fichier c/cipher.h	17
3.4.1 Description détaillée	18
3.4.2 Documentation des définitions de type	18
3.4.2.1 byte	18

3.4.3 Documentation des fonctions	18
3.4.3.1 addRoundKey()	18
3.4.3.2 cipher()	19
3.4.3.3 invCipher()	19
3.4.3.4 invShiftRows()	19
3.4.3.5 keyExpansion()	20
3.4.3.6 mixColumns()	20
3.4.3.7 rcon()	21
3.4.3.8 rotWord()	21
3.4.3.9 shiftOneRow()	22
3.4.3.10 shiftRows()	22
3.4.3.11 subBytes()	22
3.4.3.12 subWord()	23
3.5 Référence du fichier c/const.h	23
3.5.1 Description détaillée	24
3.5.2 Documentation des définitions de type	24
3.5.2.1 byte	24
3.5.3 Documentation des variables	25
3.5.3.1 a_x_invMixColumns	25
3.5.3.2 a_x_mixColumns	25
3.5.3.3 invSbox	25
3.5.3.4 sbox	26
3.6 Référence du fichier c/tests.c	26
3.6.1 Description détaillée	27
3.6.2 Documentation des fonctions	27
3.6.2.1 main()	28
3.6.2.2 testAddRoundKey()	28
3.6.2.3 testAesEcryptFile()	28
3.6.2.4 testAesEncrypt()	28
3.6.2.5 testAsciiToHex()	28
3.6.2.6 testByteXor()	29
3.6.2.7 testCipher()	29
3.6.2.8 testHexToAscii()	29
3.6.2.9 testInvCipher()	29
3.6.2.10 testInvMixColumns()	29
3.6.2.11 testInvShiftRows()	29
3.6.2.12 testKeyExpansion()	30
3.6.2.13 testMergeArr()	30
3.6.2.14 testMixColumns()	30
3.6.2.15 testMulti()	30
3.6.2.16 testRcon()	30
3.6.2.17 testRotWord()	30

3.6.2.18 testShiftRows()	31
3.6.2.19 testSplitArr()	31
3.6.2.20 testSubBytes()	31
3.6.2.21 testSubWord()	31
3.6.2.22 testSwitchColRows()	31
3.6.3 Documentation des variables	31
3.6.3.1 a_x_invMixColumns_test	32
3.6.3.2 a_x_mixColumns_test	32
3.6.3.3 invSbox_test	32
3.6.3.4 sbox_test	33
3.7 Référence du fichier c/tools.c	33
3.7.1 Description détaillée	34
3.7.2 Documentation des définitions de type	34
3.7.2.1 byte	34
3.7.3 Documentation des fonctions	34
3.7.3.1 byteXor()	34
3.7.3.2 mergeArr()	35
3.7.3.3 multi()	35
3.7.3.4 printByte()	35
3.7.3.5 splitArr()	36
3.7.3.6 switchColRows()	36
3.8 Référence du fichier c/tools.h	37
3.8.1 Description détaillée	37
3.8.2 Documentation des définitions de type	37
3.8.2.1 byte	37
3.8.3 Documentation des fonctions	38
3.8.3.1 byteXor()	38
3.8.3.2 mergeArr()	38
3.8.3.3 multi()	39
3.8.3.4 printByte()	39
3.8.3.5 splitArr()	39
3.8.3.6 switchColRows()	40
Index	41

Chapitre 1

Liste des bogues

Fichier [aes.c](#)

No known bugs.

Fichier [aes.h](#)

No known bugs.

Fichier [cipher.c](#)

No known bugs.

Fichier [cipher.h](#)

No known bugs.

Fichier [const.h](#)

No known bugs.

Fichier [tests.c](#)

No known bugs.

Fichier [tools.c](#)

No known bugs.

Fichier [tools.h](#)

No known bugs.

Chapitre 2

Index des fichiers

2.1 Liste des fichiers

Liste de tous les fichiers avec une brève description :

c/aes.c	AES encryption and decryption protocol	5
c/aes.h	Function prototypes of the aes method	8
c/cipher.c	Cipher method	11
c/cipher.h	Function prototypes of the cipher method	17
c/const.h	Constants	23
c/tests.c	Tests methods	26
c/tools.c	Tools method	33
c/tools.h	Functions prototypes of the tools.c file	37

Chapitre 3

Documentation des fichiers

3.1 Référence du fichier c/aes.c

AES encryption and decryption protocol.

```
#include "cipher.h"
#include "tools.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

Fonctions

- `byte * keyprocess` (char *key, int keysize, int *nr)
Process of keyExpansion.
- char * `hextoascii` (const char *in)
Convert a hexadecimal string to an ascii string.
- char * `asciitohex` (const char *in)
Convert an ascii string to a hexadecimal string.
- int `aes_encrypt` (char *data, int size, char *key, int keysize)
Encrypt data with AES.
- int `aes_decrypt` (char *data, int size, char *key, int keysize)
Decrypt data with AES.

3.1.1 Description détaillée

AES encryption and decryption protocol.

Contient les fonctions de chiffrement et de déchiffrement AES pour des données de taille multiple de 16 octets.

Auteur

Mazzone Rémi (rems-38)
Moussu Guillemot (guillemotmoussu)

Bogue No known bugs.

3.1.2 Documentation des fonctions

3.1.2.1 `aes_decrypt()`

```
int aes_decrypt (
    char * data,
    int size,
    char * key,
    int keysize )
```

Decrypt data with AES.

Paramètres

<i>data</i>	The data to decrypt
<i>size</i>	The size of the data (multiple of 16 bytes)
<i>key</i>	The key to decrypt the data
<i>keysize</i>	The size of the key (16, 24, 32 bytes)

Renvoie

0 if success, 1 if error

Définition à la ligne 80 du fichier aes.c.

3.1.2.2 `aes_encrypt()`

```
int aes_encrypt (
    char * data,
    int size,
    char * key,
    int keysize )
```

Encrypt data with AES.

Paramètres

<i>data</i>	The data to encrypt
<i>size</i>	The size of the data (multiple of 16 bytes)
<i>key</i>	The key to encrypt the data
<i>keysize</i>	The size of the key (16, 24, 32 bytes)

Renvoie

0 if success, 1 if error

Définition à la ligne 55 du fichier aes.c.

3.1.2.3 asciitohex()

```
char* asciitohex (
    const char * in )
```

Convert an ascii string to a hexadecimal string.

Paramètres

<i>in</i>	The ascii string
-----------	------------------

Renvoie

The hexadecimal string

Définition à la ligne 44 du fichier aes.c.

3.1.2.4 hextoascii()

```
char* hextoascii (
    const char * in )
```

Convert a hexadecimal string to an ascii string.

Paramètres

<i>in</i>	The hexadecimal string
-----------	------------------------

Renvoie

The ascii string

Définition à la ligne 32 du fichier aes.c.

3.1.2.5 keyprocess()

```
byte* keyprocess (
    char * key,
    int keysize,
    int * nr )
```

Process of keyExpansion.

Alloue la mémoire pour la clé étendue Calcule les valeurs de Nr et Nk Rempli la clé étendue

Paramètres

<i>key</i>	The initial key (16, 24, 32 bytes)
<i>keysize</i>	The size of the key (16, 24, 32 bytes)
<i>nr</i>	The number of rounds (10, 12, 14) (output variable)

Renvoie

The extended key

Définition à la ligne 22 du fichier aes.c.

3.2 Référence du fichier c/aes.h

Function prototypes of the aes method.

Définitions de type

— typedef unsigned char [byte](#)

Fonctions

- [byte](#) * [keyprocess](#) (char *key, int keysize, int *nr)
Process of keyExpansion.
- char * [hextoascii](#) (const char *in)
Convert a hexadecimal string to an ascii string.
- char * [asciitohex](#) (const char *in)
Convert an ascii string to a hexadecimal string.
- int [aes_encrypt](#) (char *data, int size, char *key, int keysize)
Encrypt data with AES.
- int [aes_decrypt](#) (char *data, int size, char *key, int keysize)
Decrypt data with AES.

3.2.1 Description détaillée

Function prototypes of the aes method.

Contient les prototypes pour le protocole AES

Auteur

Mazzone Rémi (rems-38)

Moussu Guillemot (guillemotmoussu)

[Bogue](#) No known bugs.

3.2.2 Documentation des définitions de type

3.2.2.1 byte

```
typedef unsigned char byte
```

Définition à la ligne 13 du fichier aes.h.

3.2.3 Documentation des fonctions

3.2.3.1 aes_decrypt()

```
int aes_decrypt (
    char * data,
    int size,
    char * key,
    int keysize )
```

Decrypt data with AES.

Paramètres

<i>data</i>	The data to decrypt
<i>size</i>	The size of the data (multiple of 16 bytes)
<i>key</i>	The key to decrypt the data
<i>keysize</i>	The size of the key (16, 24, 32 bytes)

Renvoie

0 if success, 1 if error

Définition à la ligne 80 du fichier aes.c.

3.2.3.2 aes_encrypt()

```
int aes_encrypt (
    char * data,
    int size,
    char * key,
    int keysize )
```

Encrypt data with AES.

Paramètres

<i>data</i>	The data to encrypt
<i>size</i>	The size of the data (multiple of 16 bytes)
<i>key</i>	The key to encrypt the data
<i>keysize</i>	The size of the key (16, 24, 32 bytes)

Renvoie

0 if success, 1 if error

Définition à la ligne 55 du fichier aes.c.

3.2.3.3 asciitohex()

```
char* asciitohex (  
    const char * in )
```

Convert an ascii string to a hexadecimal string.

Paramètres

<i>in</i>	The ascii string
-----------	------------------

Renvoie

The hexadecimal string

Définition à la ligne 44 du fichier aes.c.

3.2.3.4 hextoascii()

```
char* hextoascii (  
    const char * in )
```

Convert a hexadecimal string to an ascii string.

Paramètres

<i>in</i>	The hexadecimal string
-----------	------------------------

Renvoie

The ascii string

Définition à la ligne 32 du fichier aes.c.

3.2.3.5 keyprocess()

```
byte* keyprocess (  
    char * key,
```



```
int keysize,
int * nr )
```

Process of keyExpansion.

Alloue la mémoire pour la clé étendue Calcule les valeurs de Nr et Nk Rempli la clé étendue

Paramètres

<i>key</i>	The initial key (16, 24, 32 bytes)
<i>keysize</i>	The size of the key (16, 24, 32 bytes)
<i>nr</i>	The number of rounds (10, 12, 14) (output variable)

Renvoie

The extended key

Définition à la ligne 22 du fichier aes.c.

3.3 Référence du fichier c/cipher.c

Cipher method.

```
#include "tools.h"
#include "const.h"
#include <string.h>
```

Fonctions

- void **addRoundKey** (byte state[], byte w[], int round)
Add the key to the state (xor operation)
- void **subBytes** (byte state[], const byte box[256])
Substitute the bytes of the state with a box.
- void **shiftOneRow** (byte state[], int row, int direction, int shift)
Shift one row of the state.
- void **shiftRows** (byte state[])
Shift all the rows of the state.
- void **invShiftRows** (byte state[])
Inverse process of shiftRows.
- void **mixColumns** (byte state[], const byte polyMix[16])
Mix the columns of the state.
- void **subWord** (byte state[4])
Equivalent of subBytes for a 4 byte state.
- void **rotWord** (byte state[4])
1 byte righth rotation of a 4 byte state
- void **rcon** (int i, byte out[4])
Create the rcon polynome associated to the round.
- void **keyExpansion** (byte key[], byte w[], int nk, int nr)
Key expansion method.
- void **cipher** (byte in[], byte w[], int nr)
Cipher method.
- void **invCipher** (byte in[], byte w[], int nr)
Inverse cipher method.

3.3.1 Description détaillée

Cipher method.

Contient tous les fonctions nécessaires au chiffrement (et au déchiffrement) d'un bloc de 16 octets avec une clé.

Auteur

Mazzone Rémi (rem38)

Moussu Guillemot (guillemotmoussu)

Bogue No known bugs.

3.3.2 Documentation des fonctions

3.3.2.1 addRoundKey()

```
void addRoundKey (
    byte state[],
    byte w[],
    int round )
```

Add the key to the state (xor operation)

Paramètres

<i>state</i>	The current state (16 bytes)
<i>w</i>	The entire key
<i>round</i>	The current round (relative to Nr)

Renvoie

Void

Définition à la ligne 20 du fichier cipher.c.

3.3.2.2 cipher()

```
void cipher (
    byte in[],
    byte w[],
    int nr )
```

Cipher method.

Paramètres

<i>in</i>	The input block (16 bytes) enlarged over the rounds
<i>w</i>	The expanded key (16*(Nr+1) bytes)
<i>nr</i>	The number of rounds

Renvoie

Void

Définition à la ligne 135 du fichier cipher.c.

3.3.2.3 invCipher()

```
void invCipher (
    byte in[],
    byte w[],
    int nr )
```

Inverse cipher method.

Paramètres

<i>in</i>	The input block (16 bytes) enlarged over the rounds
<i>w</i>	The expanded key (16*(Nr+1) bytes)
<i>nr</i>	The number of rounds

Renvoie

Void

Définition à la ligne 149 du fichier cipher.c.

3.3.2.4 invShiftRows()

```
void invShiftRows (
    byte state[] )
```

Inverse process of shiftRows.

Paramètres

<i>state</i>	The current state (16 bytes)
--------------	------------------------------

Renvoie

Void

Définition à la ligne 59 du fichier cipher.c.

3.3.2.5 keyExpansion()

```
void keyExpansion (
    byte key[],
    byte w[],
    int nk,
    int nr )
```

Key expansion method.

Paramètres

<i>key</i>	The key (16, 24 or 32 bytes)
<i>w</i>	The expanded key generated (16*(Nr+1) bytes)
<i>nk</i>	The number of words in the key (4, 6 or 8 refering to the key size (16, 24 or 32 bytes)))
<i>nr</i>	The number of rounds

Renvoie

Void

Définition à la ligne 110 du fichier cipher.c.

3.3.2.6 mixColumns()

```
void mixColumns (
    byte state[],
    const byte polyMix[16] )
```

Mix the columns of the state.

Paramètres

<i>state</i>	The current state (16 bytes)
<i>polyMix</i>	Either the "a_x-Mix-Columns" or "a_x-Inverse-Mix-Columns" (16 bytes)

Renvoie

Void

Définition à la ligne 69 du fichier cipher.c.

3.3.2.7 rcon()

```
void rcon (
    int i,
    byte out[4] )
```

Create the rcon polynome associated to the round.

Paramètres

<i>i</i>	The current round
<i>out</i>	The word generated (4 bytes)

Renvoie

Void

Définition à la ligne 97 du fichier cipher.c.

3.3.2.8 rotWord()

```
void rotWord (
    byte state[4] )
```

1 byte righth rotation of a 4 byte state

Paramètres

<i>state</i>	The current word (4 bytes)
--------------	----------------------------

Renvoie

Void

Définition à la ligne 88 du fichier cipher.c.

3.3.2.9 shiftOneRow()

```
void shiftOneRow (
    byte state[],
    int row,
    int direction,
    int shift )
```

Shift one row of the state.

Paramètres

<i>state</i>	The current state (16 bytes)
<i>row</i>	The row to shift
<i>direction</i>	The direction of the shift (1 for right, -1 for left)
<i>shift</i>	The number of shifts

Renvoie

Void

Définition à la ligne 32 du fichier cipher.c.

3.3.2.10 shiftRows()

```
void shiftRows (
    byte state[] )
```

Shift all the rows of the state.

Paramètres

<i>state</i>	The current state (16 bytes)
--------------	------------------------------

Renvoie

Void

Définition à la ligne 49 du fichier cipher.c.

3.3.2.11 subBytes()

```
void subBytes (
    byte state[],
    const byte box[256] )
```

Substitute the bytes of the state with a box.

Paramètres

<i>state</i>	The current state (16 bytes)
<i>box</i>	Either the S-Box or the inverse S-Box (256 bytes)

Renvoie

Void

Définition à la ligne 26 du fichier cipher.c.

3.3.2.12 subWord()

```
void subWord (
    byte state[4] )
```

Equivalent of subBytes for a 4 byte state.

Paramètres

<i>state</i>	The current word (4 bytes)
--------------	----------------------------

Renvoie

Void

Définition à la ligne 82 du fichier cipher.c.

3.4 Référence du fichier c/cipher.h

Function prototypes of the cipher method.

Définitions de type

- typedef unsigned char [byte](#)

Fonctions

- void [addRoundKey](#) ([byte](#) state[], [byte](#) w[], int round)
Add the key to the state (xor operation)
- void [subBytes](#) ([byte](#) state[], const [byte](#) box[256])
Substitute the bytes of the state with a box.
- void [shiftOneRow](#) ([byte](#) state[], int row, int direction, int shift)
Shift one row of the state.
- void [shiftRows](#) ([byte](#) state[])
Shift all the rows of the state.
- void [invShiftRows](#) ([byte](#) state[])
Inverse process of shiftRows.
- void [mixColumns](#) ([byte](#) state[], const [byte](#) polyMix[16])
Mix the columns of the state.
- void [subWord](#) ([byte](#) state[4])
Equivalent of subBytes for a 4 byte state.
- void [rotWord](#) ([byte](#) state[4])
1 byte righth rotation of a 4 byte state
- void [rcon](#) (int i, [byte](#) out[4])
Create the rcon polynome associated to the round.
- void [keyExpansion](#) ([byte](#) key[], [byte](#) w[], int nk, int nr)
Key expansion method.
- void [cipher](#) ([byte](#) in[], [byte](#) w[], int nr)
Cipher method.
- void [invCipher](#) ([byte](#) in[], [byte](#) w[], int nr)
Inverse cipher method.

3.4.1 Description détaillée

Function prototypes of the cipher method.

Contient les prototypes pour le cipher

Auteur

Mazzone Rémi (rem-s38)

Moussu Guillemot (guillemotmoussu)

Bogue No known bugs.

3.4.2 Documentation des définitions de type

3.4.2.1 byte

```
typedef unsigned char byte
```

Définition à la ligne 13 du fichier cipher.h.

3.4.3 Documentation des fonctions

3.4.3.1 addRoundKey()

```
void addRoundKey (
    byte state[],
    byte w[],
    int round )
```

Add the key to the state (xor operation)

Paramètres

<i>state</i>	The current state (16 bytes)
<i>w</i>	The entire key
<i>round</i>	The current round (relative to Nr)

Renvoie

Void

Définition à la ligne 20 du fichier cipher.c.

3.4.3.2 cipher()

```
void cipher (
    byte in[],
    byte w[],
    int nr )
```

Cipher method.

Paramètres

<i>in</i>	The input block (16 bytes) enlarged over the rounds
<i>w</i>	The expanded key (16*(Nr+1) bytes)
<i>nr</i>	The number of rounds

Renvoie

Void

Définition à la ligne 135 du fichier cipher.c.

3.4.3.3 invCipher()

```
void invCipher (
    byte in[],
    byte w[],
    int nr )
```

Inverse cipher method.

Paramètres

<i>in</i>	The input block (16 bytes) enlarged over the rounds
<i>w</i>	The expanded key (16*(Nr+1) bytes)
<i>nr</i>	The number of rounds

Renvoie

Void

Définition à la ligne 149 du fichier cipher.c.

3.4.3.4 invShiftRows()

```
void invShiftRows (
    byte state[] )
```

Inverse process of shiftRows.

Paramètres

<i>state</i>	The current state (16 bytes)
--------------	------------------------------

Renvoie

Void

Définition à la ligne 59 du fichier cipher.c.

3.4.3.5 keyExpansion()

```
void keyExpansion (
    byte key[],
    byte w[],
    int nk,
    int nr )
```

Key expansion method.

Paramètres

<i>key</i>	The key (16, 24 or 32 bytes)
<i>w</i>	The expanded key generated (16*(Nr+1) bytes)
<i>nk</i>	The number of words in the key (4, 6 or 8 referring to the key size (16, 24 or 32 bytes)))
<i>nr</i>	The number of rounds

Renvoie

Void

Définition à la ligne 110 du fichier cipher.c.

3.4.3.6 mixColumns()

```
void mixColumns (
    byte state[],
    const byte polyMix[16] )
```

Mix the columns of the state.

Paramètres

<i>state</i>	The current state (16 bytes)
<i>polyMix</i>	Either the "a_x-Mix-Columns" or "a_x-Inverse-Mix-Columns" (16 bytes)

Renvoie

Void

Définition à la ligne 69 du fichier cipher.c.

3.4.3.7 rcon()

```
void rcon (
    int i,
    byte out[4] )
```

Create the rcon polynome associated to the round.

Paramètres

<i>i</i>	The current round
<i>out</i>	The word generated (4 bytes)

Renvoie

Void

Définition à la ligne 97 du fichier cipher.c.

3.4.3.8 rotWord()

```
void rotWord (
    byte state[4] )
```

1 byte righth rotation of a 4 byte state

Paramètres

<i>state</i>	The current word (4 bytes)
--------------	----------------------------

Renvoie

Void

Définition à la ligne 88 du fichier cipher.c.

3.4.3.9 shiftOneRow()

```
void shiftOneRow (
    byte state[],
    int row,
    int direction,
    int shift )
```

Shift one row of the state.

Paramètres

<i>state</i>	The current state (16 bytes)
<i>row</i>	The row to shift
<i>direction</i>	The direction of the shift (1 for right, -1 for left)
<i>shift</i>	The number of shifts

Renvoie

Void

Définition à la ligne 32 du fichier cipher.c.

3.4.3.10 shiftRows()

```
void shiftRows (
    byte state[] )
```

Shift all the rows of the state.

Paramètres

<i>state</i>	The current state (16 bytes)
--------------	------------------------------

Renvoie

Void

Définition à la ligne 49 du fichier cipher.c.

3.4.3.11 subBytes()

```
void subBytes (
    byte state[],
    const byte box[256] )
```

Substitute the bytes of the state with a box.

Paramètres

<i>state</i>	The current state (16 bytes)
<i>box</i>	Either the S-Box or the inverse S-Box (256 bytes)

Renvoie

Void

Définition à la ligne 26 du fichier cipher.c.

3.4.3.12 subWord()

```
void subWord (
    byte state[4] )
```

Equivalent of subBytes for a 4 byte state.

Paramètres

<i>state</i>	The current word (4 bytes)
--------------	----------------------------

Renvoie

Void

Définition à la ligne 82 du fichier cipher.c.

3.5 Référence du fichier c/const.h

Constants.

Définitions de type

— typedef unsigned char `byte`

Variables

— const `byte` `sbox` [256] = {0x63, 0x7c, 0x77, 0x7b, 0xf2, 0x6b, 0x6f, 0xc5, 0x30, 0x01, 0x67, 0x2b, 0xfe, 0xd7, 0xab, 0x76, 0xca, 0x82, 0xc9, 0x7d, 0xfa, 0x59, 0x47, 0xf0, 0xad, 0xd4, 0xa2, 0xaf, 0x9c, 0xa4, 0x72, 0xc0, 0xb7, 0xfd, 0x93, 0x26, 0x36, 0x3f, 0xf7, 0xcc, 0x34, 0xa5, 0xe5, 0xf1, 0x71, 0xd8, 0x31, 0x15, 0x04, 0xc7, 0x23, 0xc3, 0x18, 0x96, 0x05, 0x9a, 0x07, 0x12, 0x80, 0xe2, 0xeb, 0x27, 0xb2, 0x75, 0x09, 0x83, 0x2c, 0x1a, 0x1b, 0x6e, 0x5a, 0xa0, 0x52, 0x3b, 0xd6, 0xb3, 0x29, 0xe3, 0x2f, 0x84, 0x53, 0xd1, 0x00, 0xed, 0x20, 0xfc, 0xb1, 0x5b, 0x6a, 0xcb, 0xbe, 0x39, 0x4a, 0x4c, 0x58, 0xcf, 0xd0, 0xef, 0xaa, 0xfb, 0x43, 0x4d, 0x33, 0x85, 0x45, 0xf9, 0x02, 0x7f, 0x50, 0x3c, 0x9f, 0xa8, 0x51, 0xa3, 0x40, 0x8f, 0x92, 0x9d, 0x38, 0xf5, 0xbc, 0xb6, 0xda, 0x21, 0x10, 0xff, 0xf3, 0xd2, 0xcd, 0x0c, 0x13, 0xec, 0x5f, 0x97, 0x44, 0x17, 0xc4, 0xa7,

```
0x7e, 0x3d, 0x64, 0x5d, 0x19, 0x73, 0x60, 0x81, 0x4f, 0xdc, 0x22, 0x2a, 0x90, 0x88, 0x46, 0xee, 0xb8, 0x14,
0xde, 0x5e, 0x0b, 0xdb, 0xe0, 0x32, 0x3a, 0x0a, 0x49, 0x06, 0x24, 0x5c, 0xc2, 0xd3, 0xac, 0x62, 0x91, 0x95,
0xe4, 0x79, 0xe7, 0xc8, 0x37, 0x6d, 0x8d, 0xd5, 0x4e, 0xa9, 0x6c, 0x56, 0xf4, 0xea, 0x65, 0x7a, 0xae, 0x08,
0xba, 0x78, 0x25, 0x2e, 0x1c, 0xa6, 0xb4, 0xc6, 0xe8, 0xdd, 0x74, 0x1f, 0x4b, 0xbd, 0x8b, 0x8a, 0x70, 0x3e,
0xb5, 0x66, 0x48, 0x03, 0xf6, 0x0e, 0x61, 0x35, 0x57, 0xb9, 0x86, 0xc1, 0x1d, 0x9e, 0xe1, 0xf8, 0x98, 0x11,
0x69, 0xd9, 0x8e, 0x94, 0x9b, 0x1e, 0x87, 0xe9, 0xce, 0x55, 0x28, 0xdf, 0x8c, 0xa1, 0x89, 0x0d, 0xbf, 0xe6,
0x42, 0x68, 0x41, 0x99, 0x2d, 0x0f, 0xb0, 0x54, 0xbb, 0x16}
```

The S-Box.

- const `byte invSbox` [256] = {0x52, 0x09, 0x6a, 0xd5, 0x30, 0x36, 0xa5, 0x38, 0xbf, 0x40, 0xa3, 0x9e, 0x81, 0xf3, 0xd7, 0xfb, 0x7c, 0xe3, 0x39, 0x82, 0x9b, 0x2f, 0xff, 0x87, 0x34, 0x8e, 0x43, 0x44, 0xc4, 0xde, 0xe9, 0xcb, 0x54, 0x7b, 0x94, 0x32, 0xa6, 0xc2, 0x23, 0x3d, 0xee, 0x4c, 0x95, 0x0b, 0x42, 0xfa, 0xc3, 0x4e, 0x08, 0x2e, 0xa1, 0x66, 0x28, 0xd9, 0x24, 0xb2, 0x76, 0x5b, 0xa2, 0x49, 0x6d, 0x8b, 0xd1, 0x25, 0x72, 0xf8, 0xf6, 0x64, 0x86, 0x68, 0x98, 0x16, 0xd4, 0xa4, 0x5c, 0xcc, 0x5d, 0x65, 0xb6, 0x92, 0x6c, 0x70, 0x48, 0x50, 0xfd, 0xed, 0xb9, 0xda, 0x5e, 0x15, 0x46, 0x57, 0xa7, 0x8d, 0x9d, 0x84, 0x90, 0xd8, 0xab, 0x00, 0x8c, 0xbc, 0xd3, 0x0a, 0xf7, 0xe4, 0x58, 0x05, 0xb8, 0xb3, 0x45, 0x06, 0xd0, 0x2c, 0x1e, 0x8f, 0xca, 0x3f, 0x0f, 0x02, 0xc1, 0xaf, 0xbd, 0x03, 0x01, 0x13, 0x8a, 0x6b, 0x3a, 0x91, 0x11, 0x41, 0x4f, 0x67, 0xdc, 0xea, 0x97, 0xf2, 0xcf, 0xce, 0xf0, 0xb4, 0xe6, 0x73, 0x96, 0xac, 0x74, 0x22, 0xe7, 0xad, 0x35, 0x85, 0xe2, 0xf9, 0x37, 0xe8, 0x1c, 0x75, 0xdf, 0x6e, 0x47, 0xf1, 0x1a, 0x71, 0x1d, 0x29, 0xc5, 0x89, 0x6f, 0xb7, 0x62, 0x0e, 0xaa, 0x18, 0xbe, 0x1b, 0xfc, 0x56, 0x3e, 0x4b, 0xc6, 0xd2, 0x79, 0x20, 0x9a, 0xdb, 0xc0, 0xfe, 0x78, 0xcd, 0x5a, 0xf4, 0x1f, 0xdd, 0xa8, 0x33, 0x88, 0x07, 0xc7, 0x31, 0xb1, 0x12, 0x10, 0x59, 0x27, 0x80, 0xec, 0x5f, 0x60, 0x51, 0x7f, 0xa9, 0x19, 0xb5, 0x4a, 0x0d, 0x2d, 0xe5, 0x7a, 0x9f, 0x93, 0xc9, 0x9c, 0xef, 0xa0, 0xe0, 0x3b, 0x4d, 0xae, 0x2a, 0xf5, 0xb0, 0xc8, 0xeb, 0xbb, 0x3c, 0x83, 0x53, 0x99, 0x61, 0x17, 0x2b, 0x04, 0x7e, 0xba, 0x77, 0xd6, 0x26, 0xe1, 0x69, 0x14, 0x63, 0x55, 0x21, 0x0c, 0x7d}

The Inverse S-Box.

- const `byte a_x_mixColumns` [16] = {0x02, 0x03, 0x01, 0x01, 0x01, 0x02, 0x03, 0x01, 0x01, 0x01, 0x02, 0x03, 0x03, 0x01, 0x01, 0x02}

The 16 bytes array used in the MixColumns process ($a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$)

- const `byte a_x_invMixColumns` [16] = {0x0e, 0x0b, 0x0d, 0x09, 0x09, 0x0e, 0x0b, 0x0d, 0x0d, 0x09, 0x0e, 0x0b, 0x0d, 0x0d, 0x09, 0x0e}

The 16 bytes array used in the InverseMixColumns process ($a^{-1}(x) = \{0b\}x^3 + \{0d\}x^2 + \{09\}x + \{0e\}$)

3.5.1 Description détaillée

Constants.

Auteur

Mazzone Rémi (rems-38)

Moussu Guillemot (guillemotmoussu)

Bogue No known bugs.

3.5.2 Documentation des définitions de type

3.5.2.1 `byte`

```
typedef unsigned char byte
```

Définition à la ligne 11 du fichier const.h.

3.5.3 Documentation des variables

3.5.3.1 a_x_invMixColumns

```
const byte a_x_invMixColumns[16] = {0x0e, 0x0b, 0x0d, 0x09, 0x09, 0x0e, 0x0b, 0x0d, 0x0d,
0x09, 0x0e, 0x0b, 0x0b, 0x0d, 0x09, 0x0e}
```

The 16 bytes array used in the InverseMixColumns process ($a^{-1}(x) = \{0b\}x_3 + \{0d\}x_2 + \{09\}x + \{0e\}$)

Définition à la ligne 33 du fichier const.h.

3.5.3.2 a_x_mixColumns

```
const byte a_x_mixColumns[16] = {0x02, 0x03, 0x01, 0x01, 0x01, 0x02, 0x03, 0x01, 0x01, 0x01,
0x02, 0x03, 0x03, 0x01, 0x01, 0x02}
```

The 16 bytes array used in the MixColumns process ($a(x) = \{03\}x_3 + \{01\}x_2 + \{01\}x + \{02\}$)

Définition à la ligne 28 du fichier const.h.

3.5.3.3 invSbox

```
const byte invSbox[256] = {0x52, 0x09, 0x6a, 0xd5, 0x30, 0x36, 0xa5, 0x38, 0xbf, 0x40, 0xa3,
0x9e, 0x81, 0xf3, 0xd7, 0xfb, 0x7c, 0xe3, 0x39, 0x82, 0x9b, 0x2f, 0xff, 0x87, 0x34, 0x8e,
0x43, 0x44, 0xc4, 0xde, 0xe9, 0xcb, 0x54, 0x7b, 0x94, 0x32, 0xa6, 0xc2, 0x23, 0x3d, 0xee,
0x4c, 0x95, 0x0b, 0x42, 0xfa, 0xc3, 0x4e, 0x08, 0x2e, 0xa1, 0x66, 0x28, 0xd9, 0x24, 0xb2,
0x76, 0x5b, 0xa2, 0x49, 0x6d, 0x8b, 0xd1, 0x25, 0x72, 0xf8, 0xf6, 0x64, 0x86, 0x68, 0x98,
0x16, 0xd4, 0xa4, 0x5c, 0xcc, 0x5d, 0x65, 0xb6, 0x92, 0x6c, 0x70, 0x48, 0x50, 0xfd, 0xed,
0xb9, 0xda, 0x5e, 0x15, 0x46, 0x57, 0xa7, 0x8d, 0x9d, 0x84, 0x90, 0xd8, 0xab, 0x00, 0x8c,
0xbc, 0xd3, 0x0a, 0xf7, 0xe4, 0x58, 0x05, 0xb8, 0xb3, 0x45, 0x06, 0xd0, 0x2c, 0x1e, 0x8f,
0xca, 0x3f, 0x0f, 0x02, 0xc1, 0xaf, 0xbd, 0x03, 0x01, 0x13, 0x8a, 0x6b, 0x3a, 0x91, 0x11,
0x41, 0x4f, 0x67, 0xdc, 0xea, 0x97, 0xf2, 0xcf, 0xce, 0xf0, 0xb4, 0xe6, 0x73, 0x96, 0xac,
0x74, 0x22, 0xe7, 0xad, 0x35, 0x85, 0xe2, 0xf9, 0x37, 0xe8, 0x1c, 0x75, 0xdf, 0x6e, 0x47,
0xf1, 0x1a, 0x71, 0x1d, 0x29, 0xc5, 0x89, 0x6f, 0xb7, 0x62, 0x0e, 0xaa, 0x18, 0xbe, 0x1b,
0xfc, 0x56, 0x3e, 0x4b, 0xc6, 0xd2, 0x79, 0x20, 0x9a, 0xdb, 0xc0, 0xfe, 0x78, 0xcd, 0x5a,
0xf4, 0x1f, 0xdd, 0xa8, 0x33, 0x88, 0x07, 0xc7, 0x31, 0xb1, 0x12, 0x10, 0x59, 0x27, 0x80,
0xec, 0x5f, 0x60, 0x51, 0x7f, 0xa9, 0x19, 0xb5, 0x4a, 0x0d, 0x2d, 0xe5, 0x7a, 0x9f, 0x93,
0xc9, 0x9c, 0xef, 0xa0, 0xe0, 0x3b, 0x4d, 0xae, 0x2a, 0xf5, 0xb0, 0xc8, 0xeb, 0xbb, 0x3c,
0x83, 0x53, 0x99, 0x61, 0x17, 0x2b, 0x04, 0x7e, 0xba, 0x77, 0xd6, 0x26, 0xe1, 0x69, 0x14,
0x63, 0x55, 0x21, 0x0c, 0x7d}
```

The Inverse S-Box.

Définition à la ligne 23 du fichier const.h.

3.5.3.4 sbbox

```
const byte sbbox[256] = {0x63, 0x7c, 0x77, 0x7b, 0xf2, 0x6b, 0x6f, 0xc5, 0x30, 0x01, 0x67,
0x2b, 0xfe, 0xd7, 0xab, 0x76, 0xca, 0x82, 0xc9, 0x7d, 0xfa, 0x59, 0x47, 0xf0, 0xad, 0xd4,
0xa2, 0xaf, 0x9c, 0xa4, 0x72, 0xc0, 0xb7, 0xfd, 0x93, 0x26, 0x36, 0x3f, 0xf7, 0xcc, 0x34,
0xa5, 0xe5, 0xf1, 0x71, 0xd8, 0x31, 0x15, 0x04, 0xc7, 0x23, 0xc3, 0x18, 0x96, 0x05, 0x9a,
0x07, 0x12, 0x80, 0xe2, 0xeb, 0x27, 0xb2, 0x75, 0x09, 0x83, 0x2c, 0x1a, 0x1b, 0x6e, 0x5a,
0xa0, 0x52, 0x3b, 0xd6, 0xb3, 0x29, 0xe3, 0x2f, 0x84, 0x53, 0xd1, 0x00, 0xed, 0x20, 0xfc,
0xb1, 0x5b, 0x6a, 0xcb, 0xbe, 0x39, 0x4a, 0x4c, 0x58, 0xcf, 0xd0, 0xef, 0xaa, 0xfb, 0x43,
0x4d, 0x33, 0x85, 0x45, 0xf9, 0x02, 0x7f, 0x50, 0x3c, 0x9f, 0xa8, 0x51, 0xa3, 0x40, 0x8f,
0x92, 0x9d, 0x38, 0xf5, 0xbc, 0xb6, 0xda, 0x21, 0x10, 0xff, 0xf3, 0xd2, 0xcd, 0x0c, 0x13,
0xec, 0x5f, 0x97, 0x44, 0x17, 0xc4, 0xa7, 0x7e, 0x3d, 0x64, 0x5d, 0x19, 0x73, 0x60, 0x81,
0x4f, 0xdc, 0x22, 0x2a, 0x90, 0x88, 0x46, 0xee, 0xb8, 0x14, 0xde, 0x5e, 0x0b, 0xdb, 0xe0,
0x32, 0x3a, 0x0a, 0x49, 0x06, 0x24, 0x5c, 0xc2, 0xd3, 0xac, 0x62, 0x91, 0x95, 0xe4, 0x79,
0xe7, 0xc8, 0x37, 0x6d, 0x8d, 0xd5, 0x4e, 0xa9, 0x6c, 0x56, 0xf4, 0xea, 0x65, 0x7a, 0xae,
0x08, 0xba, 0x78, 0x25, 0x2e, 0x1c, 0xa6, 0xb4, 0xc6, 0xe8, 0xdd, 0x74, 0x1f, 0x4b, 0xbd,
0x8b, 0x8a, 0x70, 0x3e, 0xb5, 0x66, 0x48, 0x03, 0xf6, 0x0e, 0x61, 0x35, 0x57, 0xb9, 0x86,
0xc1, 0x1d, 0x9e, 0xe1, 0xf8, 0x98, 0x11, 0x69, 0xd9, 0x8e, 0x94, 0x9b, 0x1e, 0x87, 0xe9,
0xce, 0x55, 0x28, 0xdf, 0x8c, 0xa1, 0x89, 0x0d, 0xbf, 0xe6, 0x42, 0x68, 0x41, 0x99, 0x2d,
0x0f, 0xb0, 0x54, 0xbb, 0x16}
```

The S-Box.

Définition à la ligne 18 du fichier const.h.

3.6 Référence du fichier c/tests.c

Tests methods.

```
#include "cipher.h"
#include "tools.h"
#include "aes.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

Fonctions

- void [testByteXor](#) (void)
- void [testMulti](#) (void)
- void [testSwitchColRows](#) (void)
- void [testSplitArr](#) (void)
- void [testMergeArr](#) (void)
- void [testAddRoundKey](#) (void)
- void [testSubBytes](#) (void)
- void [testShiftRows](#) (void)
- void [testInvShiftRows](#) (void)
- void [testMixColumns](#) (void)
- void [testInvMixColumns](#) (void)
- void [testSubWord](#) (void)
- void [testRotWord](#) (void)
- void [testRcon](#) (void)
- void [testKeyExpansion](#) (void)
- void [testCipher](#) (void)
- void [testInvCipher](#) (void)
- void [testHexToAscii](#) (void)
- void [testAsciiToHex](#) (void)
- void [testAesEncrypt](#) (void)
- void [testAesEncryptFile](#) (void)
- int [main](#) (void)

Main function.

Variables

- const `byte sbbox_test` [256] = {0x63, 0x7c, 0x77, 0x7b, 0xf2, 0x6b, 0x6f, 0xc5, 0x30, 0x01, 0x67, 0x2b, 0xfe, 0xd7, 0xab, 0x76, 0xca, 0x82, 0xc9, 0x7d, 0xfa, 0x59, 0x47, 0xf0, 0xad, 0xd4, 0xa2, 0xaf, 0x9c, 0xa4, 0x72, 0xc0, 0xb7, 0xfd, 0x93, 0x26, 0x36, 0x3f, 0xf7, 0xcc, 0x34, 0xa5, 0xe5, 0xf1, 0x71, 0xd8, 0x31, 0x15, 0x04, 0xc7, 0x23, 0xc3, 0x18, 0x96, 0x05, 0x9a, 0x07, 0x12, 0x80, 0xe2, 0xeb, 0x27, 0xb2, 0x75, 0x09, 0x83, 0x2c, 0x1a, 0x1b, 0x6e, 0x5a, 0xa0, 0x52, 0x3b, 0xd6, 0xb3, 0x29, 0xe3, 0x2f, 0x84, 0x53, 0xd1, 0x00, 0xed, 0x20, 0xfc, 0xb1, 0x5b, 0x6a, 0xcb, 0xbe, 0x39, 0x4a, 0x4c, 0x58, 0xcf, 0xd0, 0xef, 0xaa, 0xfb, 0x43, 0x4d, 0x33, 0x85, 0x45, 0xf9, 0x02, 0x7f, 0x50, 0x3c, 0x9f, 0xa8, 0x51, 0xa3, 0x40, 0x8f, 0x92, 0x9d, 0x38, 0xf5, 0xbc, 0xb6, 0xda, 0x21, 0x10, 0xff, 0xf3, 0xd2, 0xcd, 0x0c, 0x13, 0xec, 0x5f, 0x97, 0x44, 0x17, 0xc4, 0xa7, 0x7e, 0x3d, 0x64, 0x5d, 0x19, 0x73, 0x60, 0x81, 0x4f, 0xdc, 0x22, 0x2a, 0x90, 0x88, 0x46, 0xee, 0xb8, 0x14, 0xde, 0x5e, 0x0b, 0xdb, 0xe0, 0x32, 0x3a, 0x0a, 0x49, 0x06, 0x24, 0x5c, 0xc2, 0xd3, 0xac, 0x62, 0x91, 0x95, 0xe4, 0x79, 0xe7, 0xc8, 0x37, 0x6d, 0x8d, 0xd5, 0x4e, 0xa9, 0x6c, 0x56, 0xf4, 0xea, 0x65, 0x7a, 0xae, 0x08, 0xba, 0x78, 0x25, 0x2e, 0x1c, 0xa6, 0xb4, 0xc6, 0xe8, 0xdd, 0x74, 0x1f, 0x4b, 0xbd, 0x8b, 0x8a, 0x70, 0x3e, 0xb5, 0x66, 0x48, 0x03, 0xf6, 0x0e, 0x61, 0x35, 0x57, 0xb9, 0x86, 0xc1, 0x1d, 0x9e, 0xe1, 0xf8, 0x98, 0x11, 0x69, 0xd9, 0x8e, 0x94, 0x9b, 0x1e, 0x87, 0xe9, 0xce, 0x55, 0x28, 0xdf, 0x8c, 0xa1, 0x89, 0x0d, 0xbf, 0xe6, 0x42, 0x68, 0x41, 0x99, 0x2d, 0x0f, 0xb0, 0x54, 0xbb, 0x16}
- const `byte invSbox_test` [256] = {0x52, 0x09, 0x6a, 0xd5, 0x30, 0x36, 0xa5, 0x38, 0xbf, 0x40, 0xa3, 0x9e, 0x81, 0xf3, 0xd7, 0xfb, 0x7c, 0xe3, 0x39, 0x82, 0x9b, 0x2f, 0xff, 0x87, 0x34, 0x8e, 0x43, 0x44, 0xc4, 0xde, 0xe9, 0xcb, 0x54, 0x7b, 0x94, 0x32, 0xa6, 0xc2, 0x23, 0x3d, 0xee, 0x4c, 0x95, 0x0b, 0x42, 0xfa, 0xc3, 0x4e, 0x08, 0x2e, 0xa1, 0x66, 0x28, 0xd9, 0x24, 0xb2, 0x76, 0x5b, 0xa2, 0x49, 0x6d, 0x8b, 0xd1, 0x25, 0x72, 0xf8, 0xf6, 0x64, 0x86, 0x68, 0x98, 0x16, 0xd4, 0xa4, 0x5c, 0xcc, 0x5d, 0x65, 0xb6, 0x92, 0x6c, 0x70, 0x48, 0x50, 0xfd, 0xed, 0xb9, 0xda, 0x5e, 0x15, 0x46, 0x57, 0xa7, 0x8d, 0x9d, 0x84, 0x90, 0xd8, 0xab, 0x00, 0x8c, 0xbc, 0xd3, 0x0a, 0xf7, 0xe4, 0x58, 0x05, 0xb8, 0xb3, 0x45, 0x06, 0xd0, 0x2c, 0x1e, 0x8f, 0xca, 0x3f, 0x0f, 0x02, 0xc1, 0xaf, 0xbd, 0x03, 0x01, 0x13, 0x8a, 0x6b, 0x3a, 0x91, 0x11, 0x41, 0x4f, 0x67, 0xdc, 0xea, 0x97, 0xf2, 0xcf, 0xce, 0xf0, 0xb4, 0xe6, 0x73, 0x96, 0xac, 0x74, 0x22, 0xe7, 0xad, 0x35, 0x85, 0xe2, 0xf9, 0x37, 0xe8, 0x1c, 0x75, 0xdf, 0x6e, 0x47, 0xf1, 0x1a, 0x71, 0x1d, 0x29, 0xc5, 0x89, 0x6f, 0xb7, 0x62, 0x0e, 0xaa, 0x18, 0xbe, 0x1b, 0xfc, 0x56, 0x3e, 0x4b, 0xc6, 0xd2, 0x79, 0x20, 0x9a, 0xdb, 0xc0, 0xfe, 0x78, 0xcd, 0x5a, 0xf4, 0x1f, 0xdd, 0xa8, 0x33, 0x88, 0x07, 0xc7, 0x31, 0xb1, 0x12, 0x10, 0x59, 0x27, 0x80, 0xec, 0x5f, 0x60, 0x51, 0x7f, 0xa9, 0x19, 0xb5, 0x4a, 0x0d, 0x2d, 0xe5, 0x7a, 0x9f, 0x93, 0xc9, 0x9c, 0xef, 0xa0, 0xe0, 0x3b, 0x4d, 0xae, 0x2a, 0xf5, 0xb0, 0xc8, 0xeb, 0xbb, 0x3c, 0x83, 0x53, 0x99, 0x61, 0x17, 0x2b, 0x04, 0x7e, 0xba, 0x77, 0xd6, 0x26, 0xe1, 0x69, 0x14, 0x63, 0x55, 0x21, 0x0c, 0x7d}
- const `byte a_x_mixColumns_test` [16] = {0x02, 0x03, 0x01, 0x01, 0x01, 0x02, 0x03, 0x01, 0x01, 0x01, 0x02, 0x03, 0x01, 0x01, 0x01, 0x02}
- const `byte a_x_invMixColumns_test` [16] = {0x0e, 0x0b, 0x0d, 0x09, 0x09, 0x0e, 0x0b, 0x0d, 0x0d, 0x09, 0x0e, 0x0b, 0x0d, 0x0d, 0x09, 0x0e}

3.6.1 Description détaillée

Tests methods.

Réalise un ensemble de tests unitaires sur les fonctions de notre code afin de s'assurer de leur bon fonctionnement.

Auteur

Mazzone Rémi (rems-38)

Moussu Guillemot (guillemotmoussu)

Bogue No known bugs.

3.6.2 Documentation des fonctions

3.6.2.1 main()

```
int main (
    void )
```

Main function.

Appel de toutes les fonctions de test

Renvoie

Void

Définition à la ligne 549 du fichier tests.c.

3.6.2.2 testAddRoundKey()

```
void testAddRoundKey (
    void )
```

Définition à la ligne 119 du fichier tests.c.

3.6.2.3 testAesEcryptFile()

```
void testAesEcryptFile (
    void )
```

Définition à la ligne 508 du fichier tests.c.

3.6.2.4 testAesEncrypt()

```
void testAesEncrypt (
    void )
```

Définition à la ligne 461 du fichier tests.c.

3.6.2.5 testAsciiToHex()

```
void testAsciiToHex (
    void )
```

Définition à la ligne 446 du fichier tests.c.

3.6.2.6 testByteXor()

```
void testByteXor (
    void )
```

Définition à la ligne 31 du fichier tests.c.

3.6.2.7 testCipher()

```
void testCipher (
    void )
```

Définition à la ligne 333 du fichier tests.c.

3.6.2.8 testHexToAscii()

```
void testHexToAscii (
    void )
```

Définition à la ligne 431 du fichier tests.c.

3.6.2.9 testInvCipher()

```
void testInvCipher (
    void )
```

Définition à la ligne 382 du fichier tests.c.

3.6.2.10 testInvMixColumns()

```
void testInvMixColumns (
    void )
```

Définition à la ligne 195 du fichier tests.c.

3.6.2.11 testInvShiftRows()

```
void testInvShiftRows (
    void )
```

Définition à la ligne 165 du fichier tests.c.

3.6.2.12 testKeyExpansion()

```
void testKeyExpansion (
    void )
```

Définition à la ligne 253 du fichier tests.c.

3.6.2.13 testMergeArr()

```
void testMergeArr (
    void )
```

Définition à la ligne 103 du fichier tests.c.

3.6.2.14 testMixColumns()

```
void testMixColumns (
    void )
```

Définition à la ligne 180 du fichier tests.c.

3.6.2.15 testMulti()

```
void testMulti (
    void )
```

Définition à la ligne 47 du fichier tests.c.

3.6.2.16 testRcon()

```
void testRcon (
    void )
```

Définition à la ligne 240 du fichier tests.c.

3.6.2.17 testRotWord()

```
void testRotWord (
    void )
```

Définition à la ligne 225 du fichier tests.c.

3.6.2.18 testShiftRows()

```
void testShiftRows (
    void )
```

Définition à la ligne 150 du fichier tests.c.

3.6.2.19 testSplitArr()

```
void testSplitArr (
    void )
```

Définition à la ligne 76 du fichier tests.c.

3.6.2.20 testSubBytes()

```
void testSubBytes (
    void )
```

Définition à la ligne 135 du fichier tests.c.

3.6.2.21 testSubWord()

```
void testSubWord (
    void )
```

Définition à la ligne 210 du fichier tests.c.

3.6.2.22 testSwitchColRows()

```
void testSwitchColRows (
    void )
```

Définition à la ligne 61 du fichier tests.c.

3.6.3 Documentation des variables

3.6.3.1 a_x_invMixColumns_test

```
const byte a_x_invMixColumns_test[16] = {0x0e, 0x0b, 0x0d, 0x09, 0x09, 0x0e, 0x0b, 0x0d, 0x0d,  
0x09, 0x0e, 0x0b, 0x0b, 0x0d, 0x09, 0x0e}
```

Définition à la ligne 27 du fichier tests.c.

3.6.3.2 a_x_mixColumns_test

```
const byte a_x_mixColumns_test[16] = {0x02, 0x03, 0x01, 0x01, 0x01, 0x02, 0x03, 0x01, 0x01,  
0x01, 0x02, 0x03, 0x03, 0x01, 0x01, 0x02}
```

Définition à la ligne 26 du fichier tests.c.

3.6.3.3 invSbox_test

```
const byte invSbox_test[256] = {0x52, 0x09, 0x6a, 0xd5, 0x30, 0x36, 0xa5, 0x38, 0xbf, 0x40,  
0xa3, 0x9e, 0x81, 0xf3, 0xd7, 0xfb, 0x7c, 0xe3, 0x39, 0x82, 0x9b, 0x2f, 0xff, 0x87, 0x34,  
0x8e, 0x43, 0x44, 0xc4, 0xde, 0xe9, 0xcb, 0x54, 0x7b, 0x94, 0x32, 0xa6, 0xc2, 0x23, 0x3d,  
0xee, 0x4c, 0x95, 0x0b, 0x42, 0xfa, 0xc3, 0x4e, 0x08, 0x2e, 0xa1, 0x66, 0x28, 0xd9, 0x24,  
0xb2, 0x76, 0x5b, 0xa2, 0x49, 0x6d, 0x8b, 0xd1, 0x25, 0x72, 0xf8, 0xf6, 0x64, 0x86, 0x68,  
0x98, 0x16, 0xd4, 0xa4, 0x5c, 0xcc, 0x5d, 0x65, 0xb6, 0x92, 0x6c, 0x70, 0x48, 0x50, 0xfd,  
0xed, 0xb9, 0xda, 0x5e, 0x15, 0x46, 0x57, 0xa7, 0x8d, 0x9d, 0x84, 0x90, 0xd8, 0xab, 0x00,  
0x8c, 0xbc, 0xd3, 0x0a, 0xf7, 0xe4, 0x58, 0x05, 0xb8, 0xb3, 0x45, 0x06, 0xd0, 0x2c, 0x1e,  
0x8f, 0xca, 0x3f, 0x0f, 0x02, 0xc1, 0xaf, 0xbd, 0x03, 0x01, 0x13, 0x8a, 0x6b, 0x3a, 0x91,  
0x11, 0x41, 0x4f, 0x67, 0xdc, 0xea, 0x97, 0xf2, 0xcf, 0xce, 0xf0, 0xb4, 0xe6, 0x73, 0x96,  
0xac, 0x74, 0x22, 0xe7, 0xad, 0x35, 0x85, 0xe2, 0xf9, 0x37, 0xe8, 0x1c, 0x75, 0xdf, 0x6e,  
0x47, 0xf1, 0x1a, 0x71, 0x1d, 0x29, 0xc5, 0x89, 0x6f, 0xb7, 0x62, 0x0e, 0xaa, 0x18, 0xbe,  
0x1b, 0xfc, 0x56, 0x3e, 0x4b, 0xc6, 0xd2, 0x79, 0x20, 0x9a, 0xdb, 0xc0, 0xfe, 0x78, 0xcd,  
0x5a, 0xf4, 0x1f, 0xdd, 0xa8, 0x33, 0x88, 0x07, 0xc7, 0x31, 0xb1, 0x12, 0x10, 0x59, 0x27,  
0x80, 0xec, 0x5f, 0x60, 0x51, 0x7f, 0xa9, 0x19, 0xb5, 0x4a, 0x0d, 0x2d, 0xe5, 0x7a, 0x9f,  
0x93, 0xc9, 0x9c, 0xef, 0xa0, 0xe0, 0x3b, 0x4d, 0xae, 0x2a, 0xf5, 0xb0, 0xc8, 0xeb, 0xbb,  
0x3c, 0x83, 0x53, 0x99, 0x61, 0x17, 0x2b, 0x04, 0x7e, 0xba, 0x77, 0xd6, 0x26, 0xe1, 0x69,  
0x14, 0x63, 0x55, 0x21, 0x0c, 0x7d}
```

Définition à la ligne 25 du fichier tests.c.

3.6.3.4 sbbox_test

```
const byte sbbox_test[256] = {0x63, 0x7c, 0x77, 0x7b, 0xf2, 0x6b, 0x6f, 0xc5, 0x30, 0x01, 0x67,
0x2b, 0xfe, 0xd7, 0xab, 0x76, 0xca, 0x82, 0xc9, 0x7d, 0xfa, 0x59, 0x47, 0xf0, 0xad, 0xd4,
0xa2, 0xaf, 0x9c, 0xa4, 0x72, 0xc0, 0xb7, 0xfd, 0x93, 0x26, 0x36, 0x3f, 0xf7, 0xcc, 0x34,
0xa5, 0xe5, 0xf1, 0x71, 0xd8, 0x31, 0x15, 0x04, 0xc7, 0x23, 0xc3, 0x18, 0x96, 0x05, 0x9a,
0x07, 0x12, 0x80, 0xe2, 0xeb, 0x27, 0xb2, 0x75, 0x09, 0x83, 0x2c, 0x1a, 0x1b, 0x6e, 0x5a,
0xa0, 0x52, 0x3b, 0xd6, 0xb3, 0x29, 0xe3, 0x2f, 0x84, 0x53, 0xd1, 0x00, 0xed, 0x20, 0xfc,
0xb1, 0x5b, 0x6a, 0xcb, 0xbe, 0x39, 0x4a, 0x4c, 0x58, 0xcf, 0xd0, 0xef, 0xaa, 0xfb, 0x43,
0x4d, 0x33, 0x85, 0x45, 0xf9, 0x02, 0x7f, 0x50, 0x3c, 0x9f, 0xa8, 0x51, 0xa3, 0x40, 0x8f,
0x92, 0x9d, 0x38, 0xf5, 0xbc, 0xb6, 0xda, 0x21, 0x10, 0xff, 0xf3, 0xd2, 0xcd, 0x0c, 0x13,
0xec, 0x5f, 0x97, 0x44, 0x17, 0xc4, 0xa7, 0x7e, 0x3d, 0x64, 0x5d, 0x19, 0x73, 0x60, 0x81,
0x4f, 0xdc, 0x22, 0x2a, 0x90, 0x88, 0x46, 0xee, 0xb8, 0x14, 0xde, 0x5e, 0x0b, 0xdb, 0xe0,
0x32, 0x3a, 0x0a, 0x49, 0x06, 0x24, 0x5c, 0xc2, 0xd3, 0xac, 0x62, 0x91, 0x95, 0xe4, 0x79,
0xe7, 0xc8, 0x37, 0x6d, 0x8d, 0xd5, 0x4e, 0xa9, 0x6c, 0x56, 0xf4, 0xea, 0x65, 0x7a, 0xae,
0x08, 0xba, 0x78, 0x25, 0x2e, 0x1c, 0xa6, 0xb4, 0xc6, 0xe8, 0xdd, 0x74, 0x1f, 0x4b, 0xbd,
0x8b, 0x8a, 0x70, 0x3e, 0xb5, 0x66, 0x48, 0x03, 0xf6, 0x0e, 0x61, 0x35, 0x57, 0xb9, 0x86,
0xc1, 0x1d, 0x9e, 0xe1, 0xf8, 0x98, 0x11, 0x69, 0xd9, 0x8e, 0x94, 0x9b, 0x1e, 0x87, 0xe9,
0xce, 0x55, 0x28, 0xdf, 0x8c, 0xa1, 0x89, 0x0d, 0xbf, 0xe6, 0x42, 0x68, 0x41, 0x99, 0x2d,
0x0f, 0xb0, 0x54, 0xbb, 0x16}
```

Définition à la ligne 24 du fichier tests.c.

3.7 Référence du fichier c/tools.c

Tools method.

```
#include <string.h>
#include <stdio.h>
```

Définitions de type

- typedef unsigned char `byte`

Fonctions

- void `byteXor` (`byte` a[], const `byte` b[], int length)
XOR operation between two byte arrays.
- `byte multi` (`byte` a, `byte` b)
Multiplication in $GF(2^8)$ for two bytes.
- void `printByte` (`byte` in[], int length)
Print a byte array.
- void `switchColRows` (`byte` state[])
Switch the columns and the rows of a 4x4 matrix.
- void `splitArr` (const `byte` in[], `byte` out[], int start, int end)
Split an array into another one.
- void `mergeArr` (const `byte` in[], `byte` out[], int start, int end)
Merge an array into another one (use for append an array)

3.7.1 Description détaillée

Tools method.

Contient un ensemble de fonctions utiles pour tout le reste de notre code Ex: affichage, séparation de tableau...

Auteur

Mazzone Rémi (rem-s38)

Moussu Guillemot (guillemotmoussu)

Bogue No known bugs.

3.7.2 Documentation des définitions de type

3.7.2.1 byte

```
typedef unsigned char byte
```

Définition à la ligne 19 du fichier tools.c.

3.7.3 Documentation des fonctions

3.7.3.1 byteXor()

```
void byteXor (
    byte a[],
    const byte b[],
    int length )
```

XOR operation between two byte arrays.

Paramètres

<i>a</i>	First byte array
<i>b</i>	Second byte array
<i>length</i>	Length of the arrays

Renvoie

Void

Définition à la ligne 23 du fichier tools.c.

3.7.3.2 mergeArr()

```
void mergeArr (
    const byte in[],
    byte out[],
    int start,
    int end )
```

Merge an array into another one (use for append an array)

Paramètres

<i>in</i>	The array to merge
<i>out</i>	The array to fill
<i>start</i>	The start index (for the "out" array)
<i>end</i>	The end index (for the "out" array)

Renvoie

Void

Définition à la ligne 64 du fichier tools.c.

3.7.3.3 multi()

```
byte multi (
    byte a,
    byte b )
```

Multiplication in $GF(2^8)$ for two bytes.

Paramètres

<i>a</i>	First byte
<i>b</i>	Second byte

Renvoie

The result of the multiplication

Définition à la ligne 29 du fichier tools.c.

3.7.3.4 printByte()

```
void printByte (
    byte in[],
    int length )
```

Print a byte array.

Paramètres

<i>in</i>	The byte array to print
<i>length</i>	The length of the array

Renvoie

Void

Définition à la ligne 43 du fichier tools.c.

3.7.3.5 splitArr()

```
void splitArr (
    const byte in[],
    byte out[],
    int start,
    int end )
```

Split an array into another one.

Paramètres

<i>in</i>	The array to split
<i>out</i>	The array to fill
<i>start</i>	The start index (for the "in" array)
<i>end</i>	The end index (for the "in" array)

Renvoie

Void

Définition à la ligne 57 du fichier tools.c.

3.7.3.6 switchColRows()

```
void switchColRows (
    byte state[] )
```

Switch the columns and the rows of a 4x4 matrix.

Paramètres

<i>state</i>	The matrix to switch
--------------	----------------------

Renvoie

Void

Définition à la ligne 48 du fichier tools.c.

3.8 Référence du fichier c/tools.h

Functions prototypes of the [tools.c](#) file.

Définitions de type

- typedef unsigned char [byte](#)

Fonctions

- void [byteXor](#) ([byte](#) a[], const [byte](#) b[], int length)
XOR operation between two byte arrays.
- [byte multi](#) ([byte](#) a, [byte](#) b)
Multiplication in $GF(2^8)$ for two bytes.
- void [printByte](#) ([byte](#) in[], int length)
Print a byte array.
- void [switchColRows](#) ([byte](#) state[])
Switch the columns and the rows of a 4x4 matrix.
- void [splitArr](#) (const [byte](#) in[], [byte](#) out[], int start, int end)
Split an array into another one.
- void [mergeArr](#) (const [byte](#) in[], [byte](#) out[], int start, int end)
Merge an array into another one (use for append an array)

3.8.1 Description détaillée

Functions prototypes of the [tools.c](#) file.

Contient les prototypes des fonctions de [tools.c](#)

Auteur

Mazzone Rémi (rems-38)

Moussu Guillemot (guillemotmoussu)

[Bogue](#) No known bugs.

3.8.2 Documentation des définitions de type

3.8.2.1 [byte](#)

```
typedef unsigned char byte
```

Définition à la ligne 13 du fichier tools.h.

3.8.3 Documentation des fonctions

3.8.3.1 byteXor()

```
void byteXor (
    byte a[],
    const byte b[],
    int length )
```

XOR operation between two byte arrays.

Paramètres

<i>a</i>	First byte array
<i>b</i>	Second byte array
<i>length</i>	Length of the arrays

Renvoie

Void

Définition à la ligne 23 du fichier tools.c.

3.8.3.2 mergeArr()

```
void mergeArr (
    const byte in[],
    byte out[],
    int start,
    int end )
```

Merge an array into another one (use for append an array)

Paramètres

<i>in</i>	The array to merge
<i>out</i>	The array to fill
<i>start</i>	The start index (for the "out" array)
<i>end</i>	The end index (for the "out" array)

Renvoie

Void

Définition à la ligne 64 du fichier tools.c.

3.8.3.3 multi()

```
byte multi (
    byte a,
    byte b )
```

Multiplication in $GF(2^8)$ for two bytes.

Paramètres

<i>a</i>	First byte
<i>b</i>	Second byte

Renvoie

The result of the multiplication

Définition à la ligne 29 du fichier tools.c.

3.8.3.4 printByte()

```
void printByte (
    byte in[],
    int length )
```

Print a byte array.

Paramètres

<i>in</i>	The byte array to print
<i>length</i>	The length of the array

Renvoie

Void

Définition à la ligne 43 du fichier tools.c.

3.8.3.5 splitArr()

```
void splitArr (
    const byte in[],
    byte out[],
    int start,
    int end )
```

Split an array into another one.

Paramètres

<i>in</i>	The array to split
<i>out</i>	The array to fill
<i>start</i>	The start index (for the "in" array)
<i>end</i>	The end index (for the "in" array)

Renvoie

Void

Définition à la ligne 57 du fichier tools.c.

3.8.3.6 switchColRows()

```
void switchColRows (
    byte state[] )
```

Switch the columns and the rows of a 4x4 matrix.

Paramètres

<i>state</i>	The matrix to switch
--------------	----------------------

Renvoie

Void

Définition à la ligne 48 du fichier tools.c.

Index

- a_x_invMixColumns
 - const.h, [25](#)
- a_x_invMixColumns_test
 - tests.c, [31](#)
- a_x_mixColumns
 - const.h, [25](#)
- a_x_mixColumns_test
 - tests.c, [32](#)
- addRoundKey
 - cipher.c, [12](#)
 - cipher.h, [18](#)
- aes.c
 - aes_decrypt, [6](#)
 - aes_encrypt, [6](#)
 - asciitohex, [7](#)
 - hextoascii, [7](#)
 - keyprocess, [7](#)
- aes.h
 - aes_decrypt, [9](#)
 - aes_encrypt, [9](#)
 - asciitohex, [10](#)
 - byte, [8](#)
 - hextoascii, [10](#)
 - keyprocess, [10](#)
- aes_decrypt
 - aes.c, [6](#)
 - aes.h, [9](#)
- aes_encrypt
 - aes.c, [6](#)
 - aes.h, [9](#)
- asciitohex
 - aes.c, [7](#)
 - aes.h, [10](#)
- byte
 - aes.h, [8](#)
 - cipher.h, [18](#)
 - const.h, [24](#)
 - tools.c, [34](#)
 - tools.h, [37](#)
- byteXor
 - tools.c, [34](#)
 - tools.h, [38](#)
- c/aes.c, [5](#)
- c/aes.h, [8](#)
- c/cipher.c, [11](#)
- c/cipher.h, [17](#)
- c/const.h, [23](#)
- c/tests.c, [26](#)
- c/tools.c, [33](#)
- c/tools.h, [37](#)
- cipher
 - cipher.c, [12](#)
 - cipher.h, [18](#)
- cipher.c
 - addRoundKey, [12](#)
 - cipher, [12](#)
 - invCipher, [13](#)
 - invShiftRows, [13](#)
 - keyExpansion, [14](#)
 - mixColumns, [14](#)
 - rcon, [14](#)
 - rotWord, [15](#)
 - shiftOneRow, [15](#)
 - shiftRows, [16](#)
 - subBytes, [16](#)
 - subWord, [17](#)
- cipher.h
 - addRoundKey, [18](#)
 - byte, [18](#)
 - cipher, [18](#)
 - invCipher, [19](#)
 - invShiftRows, [19](#)
 - keyExpansion, [20](#)
 - mixColumns, [20](#)
 - rcon, [21](#)
 - rotWord, [21](#)
 - shiftOneRow, [21](#)
 - shiftRows, [22](#)
 - subBytes, [22](#)
 - subWord, [23](#)
- const.h
 - a_x_invMixColumns, [25](#)
 - a_x_mixColumns, [25](#)
 - byte, [24](#)
 - invSbox, [25](#)
 - sbox, [25](#)
- hextoascii
 - aes.c, [7](#)
 - aes.h, [10](#)
- invCipher
 - cipher.c, [13](#)
 - cipher.h, [19](#)
- invSbox
 - const.h, [25](#)
- invSbox_test
 - tests.c, [32](#)

- invShiftRows
 - cipher.c, [13](#)
 - cipher.h, [19](#)
- keyExpansion
 - cipher.c, [14](#)
 - cipher.h, [20](#)
- keyprocess
 - aes.c, [7](#)
 - aes.h, [10](#)
- main
 - tests.c, [27](#)
- mergeArr
 - tools.c, [34](#)
 - tools.h, [38](#)
- mixColumns
 - cipher.c, [14](#)
 - cipher.h, [20](#)
- multi
 - tools.c, [35](#)
 - tools.h, [38](#)
- printByte
 - tools.c, [35](#)
 - tools.h, [39](#)
- rcon
 - cipher.c, [14](#)
 - cipher.h, [21](#)
- rotWord
 - cipher.c, [15](#)
 - cipher.h, [21](#)
- sbox
 - const.h, [25](#)
- sbox_test
 - tests.c, [32](#)
- shiftOneRow
 - cipher.c, [15](#)
 - cipher.h, [21](#)
- shiftRows
 - cipher.c, [16](#)
 - cipher.h, [22](#)
- splitArr
 - tools.c, [36](#)
 - tools.h, [39](#)
- subBytes
 - cipher.c, [16](#)
 - cipher.h, [22](#)
- subWord
 - cipher.c, [17](#)
 - cipher.h, [23](#)
- switchColRows
 - tools.c, [36](#)
 - tools.h, [40](#)
- testAddRoundKey
 - tests.c, [28](#)
- testAesEcryptFile
 - tests.c, [28](#)
- testAesEncrypt
 - tests.c, [28](#)
- testAsciiToHex
 - tests.c, [28](#)
- testByteXor
 - tests.c, [28](#)
- testCipher
 - tests.c, [29](#)
- testHexToAscii
 - tests.c, [29](#)
- testInvCipher
 - tests.c, [29](#)
- testInvMixColumns
 - tests.c, [29](#)
- testInvShiftRows
 - tests.c, [29](#)
- testKeyExpansion
 - tests.c, [29](#)
- testMergeArr
 - tests.c, [30](#)
- testMixColumns
 - tests.c, [30](#)
- testMulti
 - tests.c, [30](#)
- testRcon
 - tests.c, [30](#)
- testRotWord
 - tests.c, [30](#)
- tests.c
 - a_x_invMixColumns_test, [31](#)
 - a_x_mixColumns_test, [32](#)
 - invSbox_test, [32](#)
 - main, [27](#)
 - sbox_test, [32](#)
 - testAddRoundKey, [28](#)
 - testAesEcryptFile, [28](#)
 - testAesEncrypt, [28](#)
 - testAsciiToHex, [28](#)
 - testByteXor, [28](#)
 - testCipher, [29](#)
 - testHexToAscii, [29](#)
 - testInvCipher, [29](#)
 - testInvMixColumns, [29](#)
 - testInvShiftRows, [29](#)
 - testKeyExpansion, [29](#)
 - testMergeArr, [30](#)
 - testMixColumns, [30](#)
 - testMulti, [30](#)
 - testRcon, [30](#)
 - testRotWord, [30](#)
 - testShiftRows, [30](#)
 - testSplitArr, [31](#)
 - testSubBytes, [31](#)
 - testSubWord, [31](#)
 - testSwitchColRows, [31](#)
- testShiftRows
 - tests.c, [30](#)

- testSplitArr
 - tests.c, [31](#)
- testSubBytes
 - tests.c, [31](#)
- testSubWord
 - tests.c, [31](#)
- testSwitchColRows
 - tests.c, [31](#)
- tools.c
 - byte, [34](#)
 - byteXor, [34](#)
 - mergeArr, [34](#)
 - multi, [35](#)
 - printByte, [35](#)
 - splitArr, [36](#)
 - switchColRows, [36](#)
- tools.h
 - byte, [37](#)
 - byteXor, [38](#)
 - mergeArr, [38](#)
 - multi, [38](#)
 - printByte, [39](#)
 - splitArr, [39](#)
 - switchColRows, [40](#)