《密码学》课程设计实验报告

实验序号：03　　实验项目名称：分组密码 AES

学　　号		姓　　名		专业、班	18 信安 3-4 班
实验地点	网安基地新珈楼 C203	指导教师	王张宜	时间	2020.11.24

一、　实验目的及要求

教学目的：

(1) 掌握分组密码的基本概念；

(2) 掌握 AES 密码算法；

(3) 了解 AES 密码的安全性；

(4) 掌握分组密码常用工作模式及其特点；

(5) 熟悉分组密码的应用。

实验要求：

(1) 熟悉 AES 算法的基本结构；

(2) 掌握 AES 算法的基本运算；

(3) 掌握 AES 算法的实现与优化方法；

(4) 熟悉 AES 算法的安全性。

二、实验设备（环境）及要求

Windows 操作系统，高级语言开发环境

三、实验内容与步骤

1. AES 算法的基本结构

输入：128 位明文，128/192/256 位密钥

子过程：

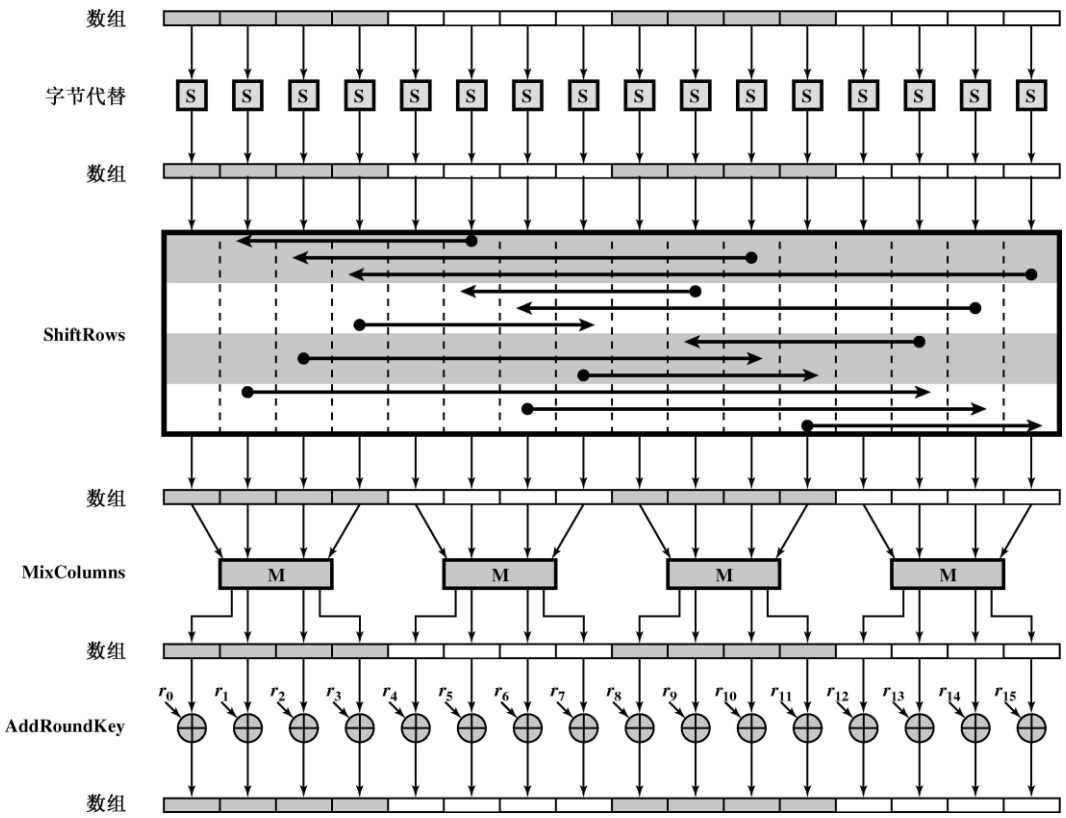
（1）S 盒变换（教材 p86 及 p92 表 3-10）

（2）行移位（教材 p87 表 3-9）

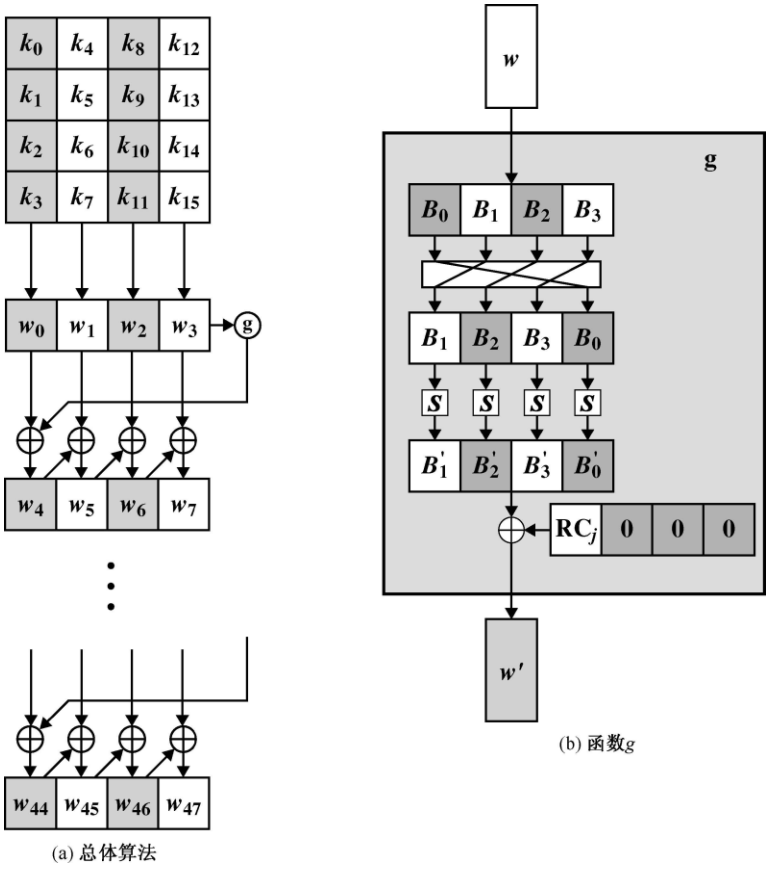
（3）列混合（教材 p87 及 p93 式 3-32）

（4）轮密钥加（教材 p87）

输出：128 位密文。



图：AES 轮函数结构



图：AES 轮密钥产生

2．AES 算法的基本运算（重点）

方法：手工计算或者通过编程代码实现下列运算：

（1）$GF(2^8)$ 上的加法（教材 p83 定义 3-2）

（为了描述方便，用花括号表示 16 进制，下同）

例：{BC} ⊕ {6A} = {D6}（下图中的 $A_{3,3} \oplus K_{3,3}=B_{3,3}$）

计算或编程方法：按位异或（提示——C、Java 等语言中的^运算符）

（2）$GF(2^8)$ 上的多项式加法（教材 p83 定义 3-7）

例：$a(x)=\{BC\}x^3+\{42\}\ x^2+\{9F\}\ x+\{4C\}$

$K(x)=\{6A\}x^3+\{00\}\ x^2+\{5C\}\ x+\{57\}$

$a(x) \oplus K(x)=\{D6\}x^3+\{42\}\ x^2+\{C3\}\ x+\{1B\}$

计算或编程方法：按位异或（提示——C、Java 等语言中的^运算符）

47	40	A3	4C
37	D4	70	9F
94	E4	3A	42
ED	A5	A6	BC

⊕

AC	19	28	57
77	FA	D1	5C
66	DC	29	00
F3	21	41	6A

=

EB	59	8B	1B
40	2E	A1	C3
F2	38	13	42
1E	84	E7	D6

$A_{0,0}$	$A_{0,1}$	$A_{0,2}$	$A_{0,3}$
$A_{1,0}$	$A_{1,1}$	$A_{1,2}$	$A_{1,3}$
$A_{2,0}$	$A_{2,1}$	$A_{2,2}$	$A_{2,3}$
$A_{3,0}$	$A_{3,1}$	$A_{3,2}$	$A_{3,3}$

⊕

$K_{0,0}$	$K_{0,1}$	$K_{0,2}$	$K_{0,3}$
$K_{1,0}$	$K_{1,1}$	$K_{1,2}$	$K_{1,3}$
$K_{2,0}$	$K_{2,1}$	$K_{2,2}$	$K_{2,3}$
$K_{3,0}$	$K_{3,1}$	$K_{3,2}$	$K_{3,3}$

=

$B_{0,0}$	$B_{0,1}$	$B_{0,2}$	$B_{0,3}$
$B_{1,0}$	$B_{1,1}$	$B_{1,2}$	$B_{1,3}$
$B_{2,0}$	$B_{2,1}$	$B_{2,2}$	$B_{2,3}$
$B_{3,0}$	$B_{3,1}$	$B_{3,2}$	$B_{3,3}$

$A_{3,3} \oplus K_{3,3} = B_{3,3} \pmod 2$

对于 AES 中的轮密钥加运算，即可以表示为对应“字节”的加法，每格相加，即定义 3-2；也可以表示为对应 32 位“字”的加法，每列相加，即定义 3-7；甚至可以表示为整个 128 位“状态”的按位异或。

思考：在不同 CPU 架构下，哪种表示方法的执行速度最快？

（3）$GF(2^8)$ 上的乘法（教材 p83 定义 3-8）

（a）借助 xtime 运算快速实现

原理：复习有限域的性质——分配率

对于｛02｝•｛？ ？ ｝（教材 p83 定义 3-5）定义为倍乘函数 xtime，可以用移位运算和条件异或运算来快速实现。由于 $GF(2^8)$ 中的所有元素都可以表示为 02 的不同幂次的和，因此所有的乘法运算都能够通过重复调用倍乘函数 xtime（定义 3-5）和加法（定义 3-2）快速实现。

例：y •｛15｝= y •｛01⊕04⊕10｝= y •｛01⊕02^2⊕02^4｝

= y⊕xtime(xtime(y))⊕xtime(xtime(xtime(xtime(y))))

= y⊕xtime(xtime(y⊕xtime(xtime(y))))

思考：该算法的效率分析？（最好情况、最坏情况）

改进：将 xtime（y）的所有 256 种取值预计算，并造表。

（b）借助生成元快速实现

GF(2^8)的全体非零元素对于乘法构成循环群。设 a 为生成元，则循环群

$\boldsymbol{G}=\{a^0,a^1,\ldots,a^{254}\}$。

G 中的乘法运算

$a^p\ a^q = a^{(p+q) \bmod 255}$ ，

于是可以把 **GF**(2^8)上的乘法简化为整数的加法运算。注意，零元素 00 与任何元素相乘都得 00。

例：｛57｝•｛83｝= ｛C1｝

计算或编程方法：

步骤 1：（准备阶段）造表

预计算两个 256 字节的表：生成元为 03 的指数表（附表 5）和生成元为 03 的对数表（附表 6）

步骤 2：查对数表

$\mathrm{Log}_{\{03\}}\{57\} = 98$ （注：指数表和对数表是 16 进制表述，高位-行号，低位-列号）

$\mathrm{Log}_{\{03\}}\{83\} = 80$

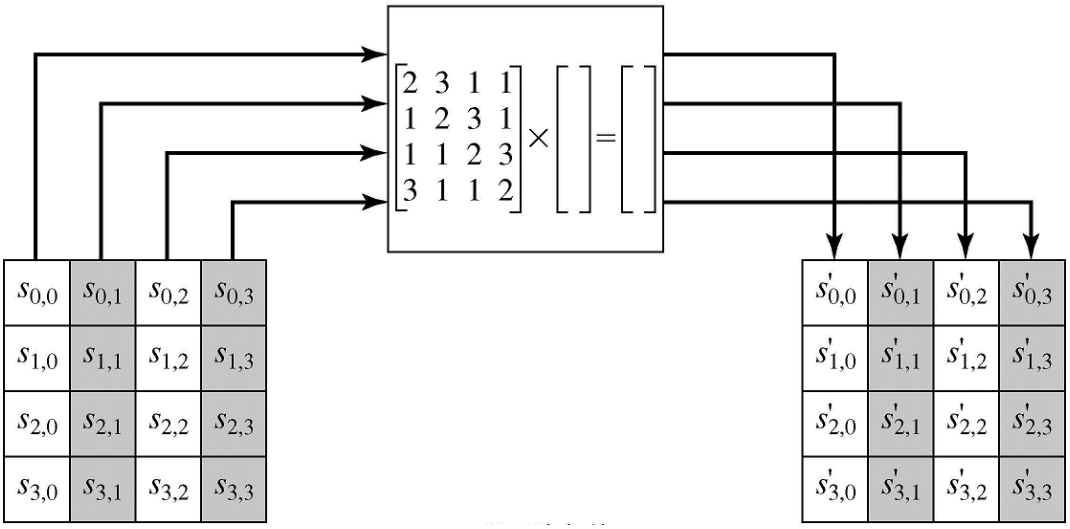
步骤 3：｛57｝•｛83｝= ｛03｝^{98} •｛03｝^{80}= ｛03｝$^{98+80 \bmod 255}$= ｛03｝^{178}

步骤 4：查指数表｛03｝^{178}= ｛03｝$^{\{B2\}}$ =193= ｛C1｝

思考：该算法的效率分析？（时间复杂度、空间复杂度）

（4）$GF(2^8)$ 上的多项式乘法（教材 p83 定义 3-8、p93 优化方案）

（a）AES 中的列混合运算的实现



其中的运算按列（32 位字）实现，当然也可表述为下面的 4×4 的字节矩阵相乘：

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

大家手工计算时，按列进行表述较为简单：

$$\begin{aligned} s'_{0,j} &= (2 \bullet s_{0,j}) \oplus (3 \bullet s_{1,j}) \oplus s_{2,j} \oplus s_{3,j} \\ s'_{1,j} &= s_{0,j} \oplus (2 \bullet s_{1,j}) \oplus (3 \bullet s_{2,j}) \oplus s_{3,j} \\ s'_{2,j} &= s_{0,j} \oplus s_{1,j} \oplus (2 \bullet s_{2,j}) \oplus (3 \bullet s_{3,j}) \\ s'_{3,j} &= (3 \bullet s_{0,j}) \oplus s_{1,j} \oplus s_{2,j} \oplus (2 \bullet s_{3,j}) \end{aligned}$$

例如下面的列混合计算：

87	F2	4D	97
6E	4C	90	EC
46	E7	4A	C3
A6	8C	D8	95

⟶

47	40	A3	4C
37	D4	70	9F
94	E4	3A	42
ED	A5	A6	BC

其中的第一列运算步骤为：

$$\begin{aligned} (\{02\} \bullet \{87\}) \oplus (\{03\} \bullet \{6E\}) \oplus \{46\} \oplus \{A6\} &= \{47\} \\ \{87\} \oplus (\{02\} \bullet \{6E\}) \oplus (\{03\} \bullet \{46\}) \oplus \{A6\} &= \{37\} \\ \{87\} \oplus \{6E\} \oplus (\{02\} \bullet \{46\}) \oplus (\{03\} \bullet \{A6\}) &= \{94\} \\ (\{03\} \bullet \{87\}) \oplus \{6E\} \oplus \{46\} \oplus (\{02\} \bullet \{A6\}) &= \{ED\} \end{aligned}$$

在 GF(2^8)中，加法就是按位 XOR 操作，乘法是根据在上述方程所示的规则执行的。注意将某值乘上 x(即{02})其结果就是将该值向左移一位， 如果该值的最高位为 1，那么在移位后还要异或(0001 1011)。（参考 xtime 的快速实现方法）

对第一个方程，我们有{02}•{87} = (0000 1110) ⊕ (0001 1011)=(0001 0101)；{03}•{6E} = {6E}⊕ ({02}•{6E}) = (0110 1110) ⊕(1101 1100) =(1011 0010)。 于是：

{02}•{87}	=	0001	0101
{03}•{6E}	=	1011	0010
{46}	=	0100	0110
{A6}	=	1010	0110
		0100	0111 = {47}

其它的方程也可以通过类似的方式得以验证。

（b）列混合运算的优化方案 1

加密过程：$c(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$

解密过程：$d(x) = \{0B\}x^3 + \{0D\}x^2 + \{09\}x + \{0E\}$

仅牵涉到与固定系数 02，03，01，01 以及 0E,0B,0D,09,所以在需要提高速度而存储空间较大的应用中可以预先计算所有 256×6 个乘法（01 不用计算），这样需要 **1.5K 字节**空间，但可省去大量乘法运算，这样可使 MixColumn 运算和 InvMixColumn 运算的乘法速度更快。

（c）列混合运算的优化方案 2

定义四个新表，T_0 到 T_3：

$$T_0 = \begin{pmatrix} S[a]\,02 \\ S[a] \\ S[a] \\ S[a]\,03 \end{pmatrix} \quad T_1 = \begin{pmatrix} S[a]\,03 \\ S[a]\,02 \\ S[a] \\ S[a] \end{pmatrix}$$

$$T_2 = \begin{pmatrix} S[a] \\ S[a]\,03 \\ S[a]\,02 \\ S[a] \end{pmatrix} \quad T_3 = \begin{pmatrix} S[a] \\ S[a] \\ S[a]\,03 \\ S[a]\,02 \end{pmatrix} \qquad (3\text{-}40)$$

T_0 到 T_3 中的每一个都是一个 256 个 4 字节元素的表，它们共占 **4KB** 的存储空间。

利用 T_0 到 T_3，可通过查表实现圈变换，于是式（3-39）变为：

$$e_j = T_0[a_{0,j}] \oplus T_1[a_{1,j+c1}] \oplus T_2[a_{2,j+c2}] \oplus T_3[a_{3,j+c3}] \oplus k_j \qquad (3\text{-}41)$$

这样，加密算法圈变换中的每一列变换，可通过式（3-56）作 4 次查表和 4 次异或运算得到。注意，在最后一圈中，没有 MixColumn 变换。这说明我们不能按式（3-41）来计算，而只

能按式（3-34）、（3-36）和（3-37）来计算。

（d）在单片机、手机、PDA 等资源受限环境下的实现

在 8 位 CPU 上，行移位、轮密钥加、S 盒变换都是对字节（8 位）操作，容易实现。但对于 32 位字的列混合操作，实现过程（CPU 位宽、存储受限）如下：

输入：4 个字节 a[0]、a[1]、a[2]、a[3]；

输出：4 个字节 a[0]、a[1]、a[2]、a[3]；

加密过程：t= a[0]⊕a[1]⊕a[2]⊕a[3] ；

u= a[0] ；

v= a[0]⊕a[1]；v=xtime(v)；a[0]=a[0] ⊕v⊕t；

v= a[1]⊕a[2]；v=xtime(v)；a[1]=a[1] ⊕v⊕t；

v= a[2]⊕a[3]；v=xtime(v)；a[2]=a[2] ⊕v⊕t；

v= a[3]⊕u；v=xtime(v)；a[3]=a[3] ⊕v⊕t；

思考：1、该算法的效率分析？（时间复杂度、空间复杂度）

2、该算法的正确性证明？

3、AES 的安全性

（1）AES 的 S 盒的实现

最简单、高效的实现方案：造表（教材 p92 表 3-10）

思考：该算法的效率分析？（时间复杂度、空间复杂度）

（2）编程研究 AES 的 S 盒的以下特性：

①明文输入改变 1 位，密文输出平均改变多少位？

②S 盒输入改变 1 位，S 盒输出平均改变多少位？

③L 输入改变 1 位，L 输出平均改变多少位？

④对于一个输入，连续施加 S 盒变换，变换多少次时出现输出等于输入？

4.扩展思考（教材习题）

(1) 比较 AES 和 DES，说明它们各有什么特点？
(2) AES 的解密算法与加密算法有什么不同？
(3) 在 GF（2^8）中，01 的逆元素是什么？
(4) 在 AES 中，对于字节“ 00”和“ 01”计算 S 盒的输出。
(5) 证明：模 x^4+1，c(x)与 d(x)互逆。
(6) 证明：$x^i \bmod (x^4+1)=x^{i \bmod 4}$ 。
(7) 利用 AES 的对数表或反对数表计算 ByteSub(25)。
(8) 求出 AES 的 S 盒的逆矩阵。
(9) 设 S 是状态，W 是圈密钥：
① 证明：InvShiftRow(InvByteSub(S))= InvByteSub(InvShiftRow(S))。
② 证明：InvMixColunm(S⊕W)= InvMixColunm(S) ⊕InvMixColunm(W)。
③ 说明上述结论对 AES 解密算法的设计有何作用。
(10) 了解 AES 采用的 SP（代替-置换）结构的特点。

5. 扩展练习（附加题）

题目 1：S 盒的安全性测试：对于 AES S 盒，计算其差分分布表和非线性度；
注: 差分分布表的定义是对于一对任意输入 x1 和 x2,满足 Δx=x1+ x2,输出等于 Δy=y1+ y2 的统计计数中的最大次数
例如对于 Δx=0, 则 Δy=0 出现 256 次, 其余 Δy=1, 2, 3…, 255 出现 0 次, 则 Δx=0 的差分次数是 256;

题目 2：S 盒的设计：产生新的 S 盒使其达到题 1 中的性质最优；
例：AES 的 S 盒是计算输入 X 的逆，然后做仿射变换得出输出 $Y=AX^{-1}+B= AX^{254}+B$。
尝试 $Y=AX^C+B$ 的形式,
(1) C 要求汉明重量为 7(例如 AES 中 254=11111110),
(2) 新盒可以改变仿射变换使用的（满秩）矩阵 A 或向量 B
给出结果,并计算其差分分布表和非线性度.

四、实验结果与数据处理

五、分析与讨论

六、教师评语

签名：

日期：

成绩

AES　S 盒题库

姓名	S 盒输入 (10 进制)	S 盒输入 (16 进制)	S 盒输出 (10 进制)	S 盒输出 (16 进制)
2017301510027 李兆恒	106			
2016301500207 肖亦晓	125			
2016301500241 杨燏锂	178			
2017301500191 吴润泽	245			
2017301500330 陈 磊	156			
2018302060248 张培铭	229			
2018302070001 沈思源	81			
2018302070131 谌金垚	202			
2018302080167 王智超	80			
2018302110381 李江旭	105			
2018302180056 鲁震豪	72			
2018302180057 王晨旭	45			
2018302180059 梅润元	95			
2018302180060 吴逸豪	9			
2018302180062 翁 斌	236			
2018302180063 袁浩天	117			
2018302180064 张展鹏	34			
2018302180068 陈亚楠	67			
2018302180069 郭点点	87			
2018302180070 程昊天	109			
2018302180073 郭梦卓	151			
2018302180077 郑津哲	2			
2018302180079 王怡静	157			
2018302180083 王丹君	252			
2018302180084 倪迩畅	104			
2018302180085 张思远	183			
2018302180086 严诚逸	137			
2018302180087 李宁馨	193			
2018302180088 周渴一	188			
2018302180091 盛威龙	135			
2018302180097 易子嘉	73			
2018302180098 陈映江	78			
2018302180099 郭瑞华	139			
2018302180151 邱振芳	27			
2018302180154 许 可	53			

2018302180160 唐炜钦	87			
2018302180163 李金峰	184			
2018302180166 梅荣新	159			
2018302180168 徐搏鸿	169			
2018302180169 叶嘉昕	155			
2018302180174 汪 毓	229			
2018302180176 蔡文颂	134			
2018302180179 陈思涵	128			
2018302180181 潘昱霏	58			
2018302180183 曾一帆	92			
2018302180184 林玉龙	236			
	11			
	250			
	38			
	19			
	138			
	204			
	52			
例 1	4	04	242	f2
例 2	206	ce	139	8b
例 3	18	12	201	c9