·CTCIS 2016 会议推荐论文· DOI:10.15961/j.jsuese.2017.01.025

基于秘密共享的 AES 的 S 盒实现与优化

钟卫东[1,2],孟庆全[2*],张帅伟[2],汪晶晶[1]
(1.网络与信息安全 武警部队重点实验室,陕西 西安 710086;2.武警工程大学 电子技术系,陕西 西安 710086)

摘 要:针对构建新的密码结构抵抗 DPA 攻击尤其是 glitch 攻击的问题,通过将输入变换到复合域 $GF(((2^2)^2)^2)$ 求逆,再变换回有限域 $GF(2^8)$ 输出的方法构造了一个低消耗的 AES 的 S 盒;并基于秘密共享的思想分仿射变换、求逆变换、逆仿射变换 3 步对 S 盒进行分组,得到一个新的 S 盒。新的 S 盒在求逆变换中采用 4×4 的正确项分组法,相比于 Nikova 提出的经典方案,减少了实现所占用的空间,降低了消耗。通过分析验证,本文方案具有较为优良的消耗特性,且对 1 阶 DPA 攻击及 glitch 攻击具有与 Nikova 方案同等级的抵抗能力。

关键词:秘密共享;AES;1 阶 DPA 攻击;glitch 攻击;S 盒
中图分类号:TP309 **文献标志码**:A **文章编号**:2096-3246(2017)01-0191-06

Implementation and Optimization of S-box on AES Based on Secret Sharing

ZHONG Weidong[1,2], *MENG Qingquan*[2*], *ZHANG Shuaiwei*[2], *WANG Jingjing*[1]
(1. Key Lab. of Network and Info. Security of the Chinese Armed Police, Xi'an 710086, China;
2. Dept. of Electronic Technol., Eng. College of the Chinese Armed Police Force, Xi'an 710086, China)

Abstract: DPA is widely used in the present as a new type of password attack technology, especially the most widely used glitch attack, which can break a large number of existing password program. In this paper, to resist DPA attacks especially glitch attacks, a new password structure of a low-consumption S-box based on AES was constructed by converting inputs into the composite field for seeking inverse, and converting them back to finite field for outputs. Then based on the idea of secret sharing, a new one was obtained by grouping S-box through the steps of affine transformation, inversion transformation and inverse-affine transformation. Compared to the classical scheme proposed by Nikova Svetla, the occupied space and consumption were reduced by using the method of correction terms. The analysis and experiments showed that the proposed scheme has better consumption characteristics, and the same level of resistance for first order DPA attack and glitch attack compared with the scheme of Nikova Svetla.

Key words: secret sharing; AES; first order DPA attack; glitch attack; S-box

AES 作为新一代国际加密标准,因其具有的高效性和较高的安全性而得到了广泛的关注和研究[1-2]。关于 AES 在硬件上实现的大量实验和优化,让其具有了相当高的效率,使得 AES 在硬件上得到了广泛的应用,但同时也使 AES 遭到了诸多的攻击。1999 年,Kocher 等[3]提出了 DPA 的概念,对密码学界造成了极大地冲击,有别于传统的对加密方案分析进行的攻击,DPA 攻击的对象是加密方案的硬件实现,通过测量获取进程中的差分信息而实现攻击。Daemen 等[4-5]通过添加掩码对 DPA 攻击进行防御,然而随着 glitch 攻击的引入,这些方案失去了效果。2006 年,Nikova 等[6]提出了基于秘密共享、门限方案抵抗 DPA 攻击的思想。2011 年,Moradi 等[7]在 EUROCRYPT2011 上提出了基于秘密共享、门限方案和多方计算思想,构造抗 DPA 攻击及故障攻击加密方案的理论方法。2013,Bilgin 等[8]

收稿日期:2016-09-18
基金项目:国家自然科学基金资助项目(61272492;61103230)
作者简介:钟卫东(1970—),男,副教授. 研究方向:密码学;信息安全.
* 通信联系人 E-mail:pyw000107@yeah.net

又发表论文实现了针对较小S盒的硬件实现并对S盒分组做了深入的研究。2014年,Bilgin等[9]又实现了一个AES的S盒的经典分组方案。

本文针对AES中单个S盒进行研究,利用元素在有限域$GF(2^8)$与复合域$GF(((2^2)^2)^2)$的转换实现了一个低消耗的S盒;随后基于秘密共享、门限方案和多方计算思想,构造出新的S盒,并采用的分解法进行分组,通过计算得到满足特性的分组。通过分析,本文方案具有较Nikova经典方案优良的功耗特性和更小的实现面积,且可以同等级地抵抗1阶DPA攻击及glitch攻击。

1 预备知识

1.1 DPA

差分功耗分析(differential power analysis, DPA)不同于针对密码方案进行攻击的传统密码攻击技术,它通过大量测量密码设备在加密过程中对数据进行处理时,产生的功耗、电磁辐射等信息,利用差分分析的方法,针对数据不同时信息的变化来确定数据是0或是1,反复进行猜测出算法所使用的密钥[3]。DPA不需要知道密码设备的任何详细信息,只需已经公开的密码算法即可。

DPA攻击通常包括以下4个步骤:

1)选取密码设备在运行进程时产生的一个中间值D,构建一个用于进行差分运算的函数$D(m,k)$,其中,k是密钥的一个部分,m为运行过程中可测量非常量数据。

2)通过对密码设备输入大量不同数据,获取运行进程中产生的m_i及轨迹t_i。

3)猜测子密钥k_i,根据猜测的k_i与已知的m_i计算得到中间值D,按照选定模型将轨迹t_i分类,计算平均功耗轨迹$\bar{t}$与分类的轨迹进行差分运算,公式如下:

$$\Delta_D(j) = \frac{\sum_{i=1}^{n} D(m_i,k_i)t_i(j)}{\sum_{i=1}^{n} D(m_i,k_i)} - \frac{\sum_{i=1}^{n}(1-D(m_i,k_i))t_i(j)}{\sum_{i=1}^{n}(1-D(m_i,k_i))}。$$

若得到的结果出现尖峰则猜测正确,否则重复此步骤。

4)通过上述步骤猜测得到全部子密钥,计算恢复密码设备的密钥。

1.2 Shamir门限秘密共享方案

Shamir[10]与Blakley于1979年分别提出了秘密共的概念,并给出了(k,n)门限秘密共享方案。(k,n)门限秘密共享方案是把一个秘密分成若干份并分给n个人,只有其中k个以上的人合作才能恢复秘密。Shamir门限秘密共享方案通过构建一个$k-1$次多项式,使用其中的常数项来保管秘密,将秘密分成n份分给参与者。k个以上的参与者可以使用拉格朗日插值公式恢复秘密,少于k个参与者则得不到关于秘密的任何信息。

1.3 门限秘密共享方案抗侧信道攻击和glitch攻击的应用

1.3.1 符号及用语

定义$\oplus$为在域$GF(2^m)$的异或运算,$\sum$为实数的加法运算;$\bar{x}$表示一组变量$(x_1,x_2,\cdots,x_n)$,$\bar{x_i}$表示除去x_i的一组变量$(x_1,x_2,\cdots,x_{i-1},x_{i+1},\cdots,x_n)$,$\Pr(t(\bar{x}) = T)$表示$t$取值$T$的概率。

定义变量x被分成n份,如果下式成立:

$$x = \bigoplus_{x=1}^{n} x_i。$$

使用(n,n)秘密共享方案,为了唯一地确定x,需要全部n份秘密。对于一个优秀的(n,n)秘密共享方案,直至获得$n-1$份秘密仍无法获得关于x的任何信息。在这里引入(k,t,n)斜坡方案,方案中t个诚实的参与者可以恢复秘密,但超过k个恶意的参与者也可以从中获取秘密相关的信息。文中将使用$(1,n,n)$斜坡方案和秘密共享方案,且条件概率$\Pr(\bar{x} \mid x)$均匀,即:

$$\forall \bar{X}:\Pr(\bar{x} = \bar{X}) = c\Pr(x = \bigoplus_{i=1}^{n} x_i)。$$

其中,c是一个标准常量,以确保概率和为1。

1.3.2 基本原理

对于一个域$GF(2^m)$上的线性变换$z = L(x)$,最安全的实现方法是使独立地对n份分别处理。定义$z_i = L(x_i)$,$0 \le i < n$,可以得到:

$$z = \bigoplus_{i=1}^{n} z_i = \bigoplus_{i=1}^{n} L(x_i) = L(\bigoplus_{i=1}^{n} x_i) = L(x)。$$

对于拥有更多输入的线性变换$z = LL(x,y,\cdots)$,可以用同样的方法进行处理。

这样实现的线性变换具有一个典型的特征,就是任意的输出份z_i,只取决于一份输入变量$(x_i,y_i,\cdots)$。这样的线性变换不会泄露信息,可以用于抵抗侧信道攻击[6]。

为了使电路安全实现,希望非线性变换具有与

线性变换相同的性质。直观地看,如果输出 z_i 的取值不取决于输入$(x_i,y_i,\cdots)$,则 z_i 不会与$(x_i,y_i,\cdots)$ 相关。通过额外的约束条件,也可以确保与 z 不相关。

定义 $z = N(x,y,\cdots)$ 为域 $GF(2^m)$ 上的非线性变换,$f_1,f_2,\cdots,f_n$ 为一组满足以下性质的函数:

性质 1(不完整性)　每个函数至少与输入任意变量 $x,y,\cdots$ 的一份不相关

$$z_1 = f_1(\bar{x}_1,\bar{y}_1,\cdots),$$
$$z_2 = f_2(\bar{x}_2,\bar{y}_2,\cdots),$$
$$\cdots$$
$$z_n = f_n(\bar{x}_n,\bar{y}_n,\cdots)。$$

性质 2(正确性)　n 份输出的总和等于正确得到输出:

$$z = \bigoplus_{i=1}^{n} z_i = \bigoplus_{i=1}^{n} f_i(\cdots) = N(x)。$$

定理 1　对于满足性质 1、2 的具有 s 个变量的非线性变换,最小的分组数 n 满足:

$$n \geq s + 1。$$

由此可以得出,要实现一个非线性变换至少需要分成 3 组。对定理 1 的证明可以推广到一般的单项式,例如 x^2y^2 可以作为一个具有 4 个变量的乘法实现。因为并不一定所有的变量都不相关,可能存在其他更少分组的解决方法,可以得到以下推论:

推论 1　域 $GF(2^m)$ 上的具有 u 个变量的非线性变换 N,最大的分组数为 $1 + 2^{mu}$。

将前文提到的条件概率进行扩展,得到:

条件 1

$$\Pr(\bar{x} = \bar{X},\bar{y} = \bar{Y},\cdots) = c\Pr(x = \bigoplus_i X_i, y = \bigoplus_i Y_i,\cdots)。$$

这表明 $\bar{x}$、$\bar{y}$ 的联合分布出现偏差将导致 x、y 的联合概率产生偏差。由此可以证明:

定理 2　在电路实现中,一组函数满足性质 1、2,且输入满足条件 1,则所有的中间值与输入 x、y、输出 z 不相关;同时任意单个函数 f_i 与 x、y、z 不相关。

性质 1 可以直观看出,条件 1 则可以全部列举出来进行验证,性质 2 由于

$$z_1 \oplus z_2 \oplus z_3 =$$
$$(x_1 \oplus x_2 \oplus x_3)(y_1 \oplus y_2 \oplus y_3) = xy = z。$$

也得以证明。

性质 3(均匀性)　对于所有满足条件 1 的输入 $x,y,\cdots$ 和输入份 $x_i,y_i,\cdots$,若条件概率 $\Pr(\bar{z} = \bar{Z} \mid z = \bigoplus_i Z_i)$ 是一个常量,则称 $z = N(x,y,\cdots)$ 是均匀的。

1.3.3　抵抗侧信道攻击及故障攻击的证明

定理 3　如果输入份 $x_i,y_i,\cdots$ 的分布满足性质 1、2 及条件 1,则电路实现的平均功耗与 x、y、z 不相关,甚至在一些输入中存在故障。

证明:由定理 2 可以得到任意单个函数 f_i 与 x、y、z 不相关,在电路实现中即可得到任意单个电路的平均功耗与 x、y、z 不相关;因为整个电路的平均功耗等于各个电路的平均功耗之和,所以整个电路的平均功耗之和也与 x、y、z 不相关;由定理 2 可知,并没有在研究特性是对是否存在故障做出假设,即证明对于存在故障的情况通用。

1.4　基于秘密共享对 S 盒分组的研究

要实现非线性变换,分组数量至少为 3,又由于过多的分组将导致电路实现时消耗过大,这里选取分组数量 $d = 3,4$ 进行研究。

定义 1　如果存在一组可逆的仿射或线性置换 $A(x)$ 和 $B(x)$,使得 2 个 S 盒 $S_1(x)$ 和 $S_2(x)$ 满足 $S_1 = B \circ S_2 \circ \boldsymbol{A}$,则称 2 个 S 盒仿射或线性等价。

任何可逆仿射置换 $A(x)$ 可以被表示成 $\boldsymbol{A} \cdot x + a$,其中,$a$ 是一个 n bit 常量,$\boldsymbol{A}$ 是一个在 $GF(2)$ 上的 $n \times n$ 的可逆矩阵。可以轻易得出共有 $2^n \times \prod_{i=0}^{n-1}(2^n - 2^i)$ 个不同的可逆仿射置换。

用仿射等价来定义等价类[8]:

当 $n = 3$ 时,通过计算共有 4 个等价类,其中 3 个等价类包含二次函数,1 个包含仿射函数。

当 $n = 4$ 时,通过计算共有 302 个等价类,其中 6 个等价类包含二次函数,1 个包含仿射函数,剩下的 295 个包含三次函数。

定理 4　如果可以对一个类中的代表进行分组,则对于同一个类中的任何置换都可以进行分组。

对于等价类的分组,首先引入直接分组法:构造一个满足正确性的分组,这点十分容易做到;为了满足不完整性,使一次项仅 x_i 在 f_{i-1} 中包含,二次项 x_ix_{i+1} 仅在 f_{i-1} 中包含,三次项 x_i 仅在 f_{i-1} 中包含。

然而通过直接分组法得到的分组可能不满足均匀性,根据计算[8],在 $n = 3$、4 时有大量的分组存在这一问题。为了解决这一问题,再次引入正确项法:

在直接分组法的基础上,在至少 2 个分组中加入不破坏不完整性的成对的项,使分组满足均匀性。同时由于项是成对加入的,并不会破坏正确性。

2　基于秘密共享的 S 盒的实现

2.1　S 盒的实现

通常 AES 的 S 盒的功能可以分成 2 个部分进行实现,分别是有限域 $GF(2^8)$ 上的逆变换和有限

域 $GF(2)$ 上的仿射变换,其中有限域 $GF(2^8)$ 上的逆变换是唯一的非线性部分。有限域 $GF(2)$ 上的仿射变换由于是线性变换,这里只以模块表示。有限域 $GF(2^8)$ 上的逆变换,分为 3 步:

1) 将元素输入由乘法器和计数器组成的模块使之变换到复合域 $GF(((2^2)^2)^2)$;

2) 将元素输入反相器进行求逆;

3) 将结果输入到2个乘法器组成的模块变换回有限域 $GF(2^8)$,实现求逆操作。

整个 S 盒的具体实现方法如图 1 所示。在实现的电路中共用到 1 个 $GF(2^2)$ 反相器,1 个 $GF(2^2)$ 平方计数器,3 个 $GF(2^2)$ 乘法器和 2 个 4 bit 异或门。

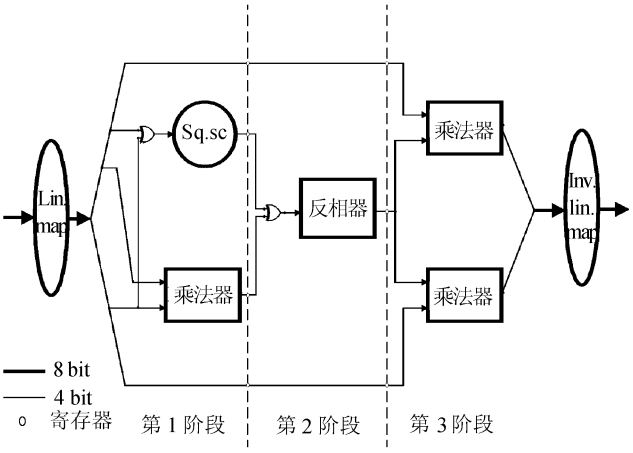

图 1 低消耗 S 盒的实现

Fig. 1 Implementation of the S-box with low consumption

2.2 共享 S 盒的实现

2.2.1 共享 S 盒实现的经典方案

在这里,首先对 Nikova 提出的经典方案做以介绍,该方案共分为 3 个阶段[1]。

在第 1 阶段中,基于秘密共享将输入的 8 bit 信息分成 4 组 8 bit 的信息 a、b、c、d,对 4 组信息进行线性变换,以实现 $GF(2)$ 上的仿射变换。将每组信息分成 2 组 4 bit 的信息,得到 8 组 4 bit 的信息(a_1, b_1, c_1, d_1, a_2, b_2, c_2, d_2),分为 S_1、S_2 两组写入寄存器中。将 S_1、S_2 输入 8 × 3 的 $GF(2^4)$ 乘法器,得到 3 组 4 bit 的信息;同时将 S_1、S_2 进行 2 组共 6 次异或操作,得到 2 组 4 bit 的信息,为了满足分组性质,与 2 对 4 bit 掩码信息进行异或操作。

在第 2 阶段中,将得到的 5 组 4 bit 信息输入 5 × 5 的反相器中,进行求逆变换,得到 5 组 4 bit 的输出信息。

在第 3 阶段中,引入 3 对 4 bit 掩码信息与 5 组 4 bit 的信息进行异或操作,得到满足分组性质的 4 组 4 bit 的信息 S_3。将 S_1、S_3 输入 8 × 3 的 $GF(2^4)$ 乘法器,得到 3 组 4 bit 的信息 m_1;将 S_2、S_3 输入 8 × 3 的 $GF(2^4)$ 乘法器,得到 3 组 4 bit 的信息 m_2。将 m_1、m_2 合成得到 3 组 8 bit 的信息,进行逆仿射变换,得到的 3 组 8 bit 的信息基于秘密共享相加即可正确输出。

2.2.2 共享 S 盒的具体实现

在选择分享份数 n 时,n 的取值应大于 2;可以看出当 $n = 3$ 时,满足秘密共享方案 3 个性质的分类方法十分有限,存在一定遭受穷举攻击的潜在风险[2]。据此,在尽可能减小值,以减小实现所需消耗和面积的情况下,选择 $n = 4$。参考 2.2.1 节对 S 盒的实现,本文基于秘密共享的 S 盒实现方案分为 3 个阶段(图 2)。

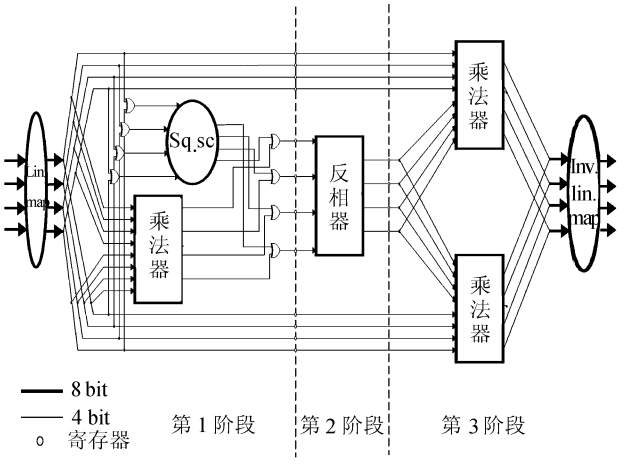

图 2 共享 S 盒的实现

Fig. 2 Implementation of the sharing S-box

在第 1 阶段中,$GF(2)$ 上的仿射变换的实现部分,将输入的 8 bit 信息基于秘密共享分成 4 组 8 bit 的信息 a、b、c、d,对 4 组信息进行线性变换,以实现 $GF(2)$ 上的仿射变换。将每组信息分成 2 组 4 bit 的信息,得到 8 组 4 bit 的信息(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8),分为 S_1、S_2 两组写入寄存器中。

在域变换的部分,将 S_1、S_2 输入 8 × 4 的 $GF(2^2)$ 乘法器,得到 4 组 4 bit 的信息(f_1, f_2, f_3, f_4);同时将 S_1、S_2 进行 4 次异或操作,得到 4 组 4 bit 的信息(g_1, g_2, g_3, g_4)。将(f_1, f_2, f_3, f_4) 与(g_1, g_2, g_3, g_4) 进行异或操作得到 4 组 4 bit 的信息(h_1, h_2, h_3, h_4) 乘法器电路中,采取直接分组法得到具体的方程式如下:

$$(f_1, f_2, f_3, f_4) = (x_1, x_2, x_3, x_4) \times (x_5, x_6, x_7, x_8),$$

$$f_1 = x_2x_5 \oplus x_3x_5 \oplus x_4x_5 \oplus x_2x_6 \oplus x_3x_6 \oplus x_4x_6 \oplus x_2x_7 \oplus x_3x_7 \oplus x_4x_7 \oplus x_2x_8 \oplus x_3x_8 \oplus x_4x_8,$$

$$f_2 = x_1x_5 \oplus x_3x_5 \oplus x_4x_5 \oplus x_1x_6 \oplus x_3x_6 \oplus x_4x_6 \oplus x_1x_7 \oplus x_3x_7 \oplus x_4x_7 \oplus x_1x_8 \oplus x_3x_8 \oplus x_4x_8,$$

$$f_3 = x_1x_5 \oplus x_2x_5 \oplus x_4x_5 \oplus x_1x_6 \oplus x_2x_6 \oplus x_4x_6 \oplus x_1x_7 \oplus x_2x_7 \oplus x_4x_7 \oplus x_1x_8 \oplus x_2x_8 \oplus x_4x_8,$$

$$f_4 = x_1x_5 \oplus x_2x_5 \oplus x_3x_5 \oplus x_1x_6 \oplus x_2x_6 \oplus x_3x_6 \oplus x_1x_7 \oplus x_2x_7 \oplus x_3x_7 \oplus x_1x_8 \oplus x_2x_8 \oplus x_3x_8。$$

与 Nikova 的方案相比,在这里减少了异或操作的数量,并且由于下一阶段的求逆操作引入分解的分组方法,并不需要通过添加掩码信息来满足分组性质,这两方面能够降低消耗和减小面积。

在第 2 阶段中,将得到的 4 组 4 bit 信息(h_1,h_2,h_3,h_4)输入 4 × 4 的 $GF(2^2)$ 反相器中,进行求逆变换,得到 4 组 4 bit 的输出信息(m_1,m_2,m_3,m_4)。

在反相器电路中,无法直接分类得到满足性质的分类,使用正确值法会因为添加掩码信息而产生大量的消耗,同样扩大分组数也会指数级增大所需等价门数。因此在这里引入分解法[8]:将一个困难的置换分解为已经可以分解的置换,如 $\mathrm{Inv}(x) = F(G(H(x)))$。通过这种方法,得到反相器的具体方程式如下:

$$(m_1,m_2,m_3,m_4) = \mathrm{Inv}(h_1,h_2,h_3,h_4),$$

$$m_1 = h_2 + h_3 \oplus h_1h_3 \oplus h_2h_3 \oplus h_2h_3h_4,$$

$$m_2 = h_4 \oplus h_1h_3 \oplus h_2h_3 \oplus h_2h_4 \oplus h_1h_3h_4,$$

$$m_3 = h_1 \oplus h_2 \oplus h_1h_3 \oplus h_1h_4 \oplus h_1h_2h_4,$$

$$m_4 = h_2 \oplus h_1h_3 \oplus h_1h_4 \oplus h_2h_4 \oplus h_1h_2h_3。$$

与 Nikova 的方案相比,我们的方案通过使用分解法这一新方法,避免了掩码信息的添加,进一步降低消耗和减小面积。

在第 3 阶段中,将 (m_1,m_2,m_3,m_4) 与 (f_1,f_2,f_3,f_4) 输入 8 × 4 的 $GF(2^2)$ 乘法器得到 4 bit 信息 (p_1,p_2,p_3,p_4),将 (m_1,m_2,m_3,m_4) 与 (g_1,g_2,g_3,g_4) 输入 8 × 4 的 $GF(2^2)$ 乘法器得到 4 bit 信息(q_1,q_2,q_3,q_4);将(p_1,p_2,p_3,p_4) 与(q_1,q_2,q_3,q_4) 进行异或,再进行逆仿射变换,得到的输出即为正确的输出。

3　共享方案的优化

3.1　针对设计电路图的优化

通过测量 S 盒中不同阶段的功耗值,DPA 攻击仍可以获得一些信息[11-12]。因此,在不同阶段分别引入随机掩码,进一步使中间值随机化,以此来杜绝这些信息的泄露。

在第 1 阶段中,将 4 组 4 bit 的信息(g_1,g_2,g_3,g_4) 与成对的随机掩码进行异或运算,使输出结果随机化,且保证结果的正确。

在第 2 阶段中,将通过反相器输出的 4 组 4 bit 的信息(m_1,m_2,m_3,m_4) 与 3 对随机掩码进行异或,其中 3 组与单组掩码异或,剩下一组通过 4 × 1 异或门与对应的 3 组掩码进行异或,具体实现如图 3 所示。

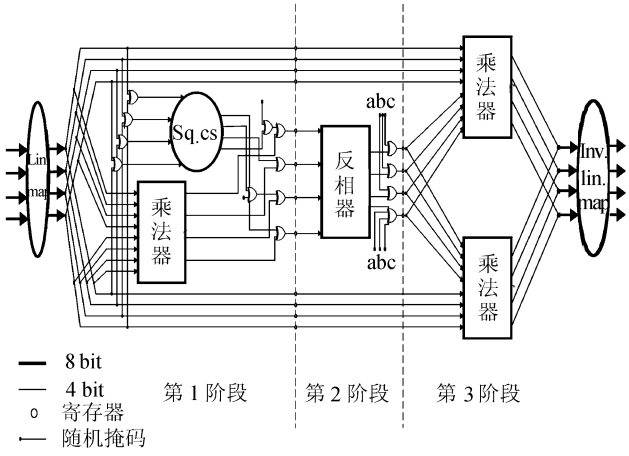

图 3　共享 S 盒的优化

Fig. 1　Optimization of the sharing S-box

3.2　针对分组方法的优化

可以直观看出,用于乘法器的分组方法较为复杂,消耗了大量的资源[13-17]。通过逐步推演,得到了一组更优的分法,可以有效地减少所需等价门数。具体的方程式如下所示:

$$(f_1,f_2,f_3,f_4) = (x_1,x_2,x_3,x_4) \times (x_5,x_6,x_7,x_8),$$

$$f_1 = x_1x_5 \oplus x_3x_5 \oplus x_4x_5 \oplus x_2x_6 \oplus x_3x_6 \oplus x_1x_7 \oplus x_2x_7 \oplus x_3x_7 \oplus x_4x_7 \oplus x_1x_8 \oplus x_3x_8,$$

$$f_2 = x_2x_5 \oplus x_3x_5 \oplus x_1x_6 \oplus x_2x_6 \oplus x_4x_6 \oplus x_1x_7 \oplus x_2x_8 \oplus x_4x_8,$$

$$f_3 = x_1x_5 \oplus x_2x_5 \oplus x_3x_5 \oplus x_4x_5 \oplus x_1x_6 \oplus x_3x_6 \oplus x_1x_7 \oplus x_2x_7 \oplus x_3x_7 \oplus x_1x_8 \oplus x_4x_8,$$

$$f_4 = x_1x_5 \oplus x_3x_5 \oplus x_2x_6 \oplus x_4x_6 \oplus x_1x_7 \oplus x_4x_7 \oplus x_2x_8 \oplus x_3x_8 \oplus x_4x_8。$$

4　结　论

在对基于秘密共享对抵抗侧信道攻击的应用及 S 盒的分组进行研究的前提下,构建了一个低消耗的 AES 的 S 盒,并对 S 盒基于秘密共享进行分组,构造了一个新的 S 盒。与 Nikova 的经典方案进行比较,本文方案具有等价的整体结构,对 1 阶 DPA 攻击及 glitch 攻击具有同等级的抵抗能力;同时基于分解法的采用及对具体实现的方程式的优化和改进,本文方案具有较之更为良好的功耗特性和更小的实现面积。但即使进行优化,与传统不抗 1 阶 DPA 攻击及 glitch 攻击的 S 盒相比,新的 S 盒在功耗方面仍有所增加,因此在使用时应有一定的取舍。

下一步,将继续研究更加优化的方案,并构造可以抵抗高阶 DPA 攻击的 S 盒。

参考文献:

[1] Bilgin B, Gierlichs B, Nikova S, et al. A more efficient AES threshold implementation[C]//Progress in Cryptology—FRICACRYPT 2014. Switzerland: Springer, 2014: 267 – 284.

[2] Daemen J, Rijmen V. The design of rijndael: AES—The advanced encryption standard[M]. Berlin: Springer-Verlag, 2002.

[3] Kocher P, Jaffe J, Jun B. Differential power analysis[C]// International Cryptology Conference on Advances in Cryptology. Berlin: Springer-Verlag, 1999: 388 – 397.

[4] Bilgin B, Bogdanov A, Kneževic M, et al. Fides: Lightweight authenticated cipher with side-channel resistance for constrained hardware [C]//Cryptographic Hardware and Embedded Systems—CHES 2013. Berlin: Springer, 2013: 142 – 158.

[5] Bilgin B, Daemen J, Nikov V, et al. Efficient and first-order DPA resistant implementations of keccak[C]//Smart Card Research and Advanced Applications. Berlin: Springer, 2013: 187 – 199.

[6] Nikova S, Rechberger C, Rijmen V. Threshold implementations against side-channel attacks and glitches[C]//International Conference on Information and Communications Security. Berlin: Springer-Verlag, 2006: 529 – 545.

[7] Moradi A, Poschmann A, Ling S, et al. Pushing the limits: A very compact and a threshold implementation of AES [C]//Advances in Cryptology—EUROCRYPT 2011. Berlin: Springer, 2011: 69 – 88.

[8] Bilgin B, Nikov V, Nikova S, et al. Threshold Implementations of all 3 × 3 and 4 × 4 S-boxes[C]//Cryptographic Hardware and Embedded Systems—CHES 2012. Berlin: Springer, 2012: 76 – 91.

[9] Bilgin B, Nikova S, Nikov V, et al. Threshold implementations of small S-boxes[J]. Cryptography & Communications, 2014, 7(1): 3 – 33.

[10] Shamir A. How to share a secret[J]. Communications of the Acm, 1979, 22(11): 612 – 613.

[11] Kutzner S, Nguyen P H, Poschmann A, et al. On 3-Share Threshold Implementations for 4-Bit S-boxes[C]//International Conference on Constructive Side-Channel Analysis and Secure Design. Berlin: Spring-Verlag, 2013: 99 – 113.

[12] Goubin L, Patarin J. DES and differential power analysis (The Duplication Method)[C]//International Workshop on Cryptographic Hardware & Embedded Systems. Berlin: Springer-Verlag, 2000: 158 – 172.

[13] Nikova S, Rijmen V, Schläffer M. Secure hardware implementation of nonlinear functions in the presence of glitches[J]. Journal of Cryptology, 2011, 24(2): 292 – 321.

[14] Brier E, Clavier C, Olivier F. Correlation power analysis with a leakage model[M]//Cryptographic Hardware and Embedded Systems—CHES 2004. Berlin: Springer, 2004: 8004 – 8010.

[15] Moradi A. Statistical tools flavor side-channel collision attacks[C]//International Conference on Theory and Applications of Cryptographic Techniques. Berlin: Springer-Verlag, 2012: 428 – 445.

[16] Batina L, Gierlichs B, Prouff E, et al. Mutual information analysis: A comprehensive study[J]. Journal of Cryptology, 2011, 24(2): 269 – 291.

[17] Mangard S, Pramstaller N, Oswald E. Successfully attacking masked AES hardware implementations[C]//International Conference on Cryptographic Hardware and Embedded Systems. Berlin: Springer-Verlag, 2005: 157 – 171.

(编辑 杨 蓓)

引用格式: Zhong Weidong, Meng Qingquan, Zhang Shuaiwei, et al. Implementation and optimization of S-box on AES based on secret sharing[J]. Advanced Engineering Sciences, 2017, 49(1): 191 – 196. [钟卫东, 孟庆全, 张帅伟, 等. 基于秘密共享的 AES 的 S 盒实现与优化[J]. 工程科学与技术, 2017, 49(1): 191 – 196.]