·安全技术· 文章编号：1000-3428(2018)11-0148-06 文献标志码：A 中图分类号：TP309

基于秘密共享的 SM4 算法 S 盒实现方案

李新超[a,b]，钟卫东[a,b]，刘明明[a,b]，李 栋[a,b]

（武警工程大学 a. 网络与信息安全武警部队重点实验室；b. 密码工程学院，西安 710086）

摘 要：针对现有的 SM4 算法掩码方案不能完全抵抗差分功耗分析（DPA）攻击的现状，提出一种基于秘密共享抵抗 DPA 攻击的方案。通过将输入变换到复合域中求逆，结合秘密共享、门限方案构造一个新型 S 盒。S 盒利用秘密共享函数代替仿射变换，在乘法器分组中采用虚拟值法，并在反相器中引入分解法，使得实现方案具有较少的运算次数和较低的空间占比。安全性分析与实验结果表明，该方案所构造的 S 盒可有效抵御高阶 DPA 攻击及 glitch 攻击，且具有较低的功率消耗。

关键词：SM4 算法；差分功耗分析；S 盒；秘密共享；虚拟值；复合域

中文引用格式：李新超，钟卫东，刘明明，等. 基于秘密共享的 SM4 算法 S 盒实现方案[J]. 计算机工程，2018，44(11)：148-153.

英文引用格式：LI Xinchao，ZHONG Weidong，LIU Mingming，et al. S-box implementation scheme of SM4 algorithm based on secret sharing[J]. Computer Engineering，2018，44(11)：148-153.

S-box Implementation Scheme of SM4 Algorithm Based on Secret Sharing

LI Xinchao[a,b]，ZHONG Weidong[a,b]，LIU Mingming[a,b]，LI Dong[a,b]

（a. Key Laboratory for Network and Information Security of Chinese Armed Police Force；b. School of Cryptographic Engineering，Engineering University of the Chinese Armed Police Force，Xi'an 710086，China）

【Abstract】Aiming at the current situation that the existing SM4 algorithm mask scheme cannot completely resist the Differential Power Analysis（DPA） attack，a scheme based on secret sharing resistance DPA attack is proposed. This method constructs a new S-box by transforming the input into the composite domain and combining with the secret sharing as well as the threshold scheme. The new S-box uses secret sharing function instead of affine transformation，adopts the virtual value method in the multiplier grouping，and introduces the decomposition method in the inverter，which makes the scheme have less computation times and lower space proportion. Security analysis and experimental results show that the S-box constructed by this scheme can effectively resist high-order DPA attacks and glitch attacks，and has low power consumption.

【Key words】SM4 algorithm；Differential Power Analysis（DPA）；S-box；secret sharing；virtual value；composite field

DOI：10.19678/j.issn.1000-3428.0051707

0 概述

SM4 是我国自行设计的分组密码标准，2012 年由国家密码管理局发布，采用非平衡 Feistel 结构，加密算法与密钥扩展算法均采用 32 轮非线性迭代结构，具有较高的安全性[1]。自 1999 年 Kocher 利用差分功耗分析（Differential Power Analysis，DPA）攻击成功破解 DES 算法之后，各种已知的密码算法包括 SM4 都面临着巨大的威胁，如何抵御 DPA 攻击成为密码学研究的一大热点。

2007 年，文献[2]通过对 SM4 算法进行分析，成功得到了 S 盒的代数表达式，为 SM4 算法的进一步研究奠定了基础。2008 年，文献[3]对 SM4 算法进行了 DPA 攻击，证明了 SM4 算法面对 DPA 攻击的脆弱性，同年又提出了一种能够抵御 DPA 攻击的安全掩码方案及其 VLSI 实现，进行对掩码防御 DPA 攻击的新尝试[4]。2014 年，文献[5]提出了在复合域求逆中对 S 盒进行掩码来抵御 DPA 的方法，并进行了实验验证。该方法加快 S 盒中求逆的运算速度，降低了消耗，并通过添加掩码，增强了抵抗 DPA 攻击的能力，但效果仍不明显。由于 SM4 算法脱胎于 AES 算法，两者具有相似的 S 盒结构，对 AES 的 DPA 攻击同样给 SM4 算法造成了威胁。为了抵御 DPA 攻击，2006 年，文献[6]提出了基于秘密共享抵

基金项目：国家自然科学基金（U1636114）；国家社会科学基金（16btj033）。

作者简介：李新超（1992—），男，硕士研究生，主研方向为密码学、信息安全；钟卫东，教授；刘明明、李 栋，硕士研究生。

收稿日期：2018-05-31 **修回日期：**2018-07-03 **E-mail：**papchao316@163.com

抗 DPA 攻击的思想,开创了抵御 DPA 攻击的新思路,并为共享 S 盒的实现提供了理论依据。2011 年,文献[7]提出了高效 S 盒的硬件实现,并将秘密共享思想应用于 AES 的 S 盒,提高了 AES 抵抗一阶 DPA 攻击的水平。2014 年,文献[8]基于秘密共享思想及复合域求逆方法对共享 S 盒的分组情况进行了研究,并提出了针对 AES 算法的共享实现方案。

针对 SM4 算法如何有效防御 DPA 攻击问题,本文根据 SM4 算法 S 盒的特点,在对 Bilgin 的共享 S 盒分组实现方案进行分析研究的基础上,结合文献[5]的工作,构造了一个适用于 SM4 算法的低功耗共享 S 盒。

1　相关知识

1.1　S 盒

S 盒作为 SM4 算法唯一的非线性部分,是评估 SM4 算法安全性的关键。2007 年,文献[2]通过对 SM4 算法 S 盒的研究,成功获得了 S 盒的代数表达式,并给出了相关参数的具体值。通过实验验证发现,该表达式及相关参数值能够满足所有的 S 盒取值。

代数表达式为:

$$S(x) = I(x \cdot \boldsymbol{A}_1 + \boldsymbol{C}_1) \cdot \boldsymbol{A}_2 + \boldsymbol{C}_2 \tag{1}$$

不可约多项式为:

$$f(x) = x^8 + x^7 + x^6 + x^5 + x^4 + x^2 + 1 \tag{2}$$

循环矩阵为:

$$\boldsymbol{A}_1 = \boldsymbol{A}_2 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \tag{3}$$

行向量为:

$$\boldsymbol{C}_1 = \boldsymbol{C}_2 = (1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1) \tag{4}$$

由于 AES 算法 S 盒与 SM4 的 S 盒具有相似的求逆部分,因此可在 SM4 的 S 盒中引入复合域方法求逆,提高运算速度,降低功耗。SM4 算法 S 盒复合域求逆过程如图 1 所示。

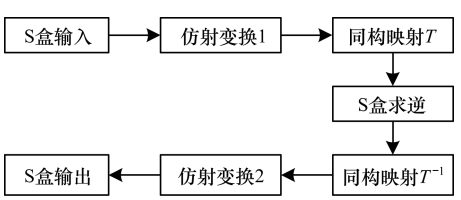

图 1　复合域 S 盒实现框架

1.2　差分功耗分析

差分功耗分析是一种典型的侧信道攻击,也是最流行的能量分析攻击,其基本原理是利用了密码设备的瞬时能量消耗与设备处理的数据和执行的操作之间的关联性[9]。通过使用特殊的电子测量仪和数学统计技术,可以检测和分析 CMOS 芯片在执行不同的指令时产生的功率消耗,从而获得芯片中与密钥有关的重要信息。DPA 攻击的目标是密码设备对大量不同数据分组进行加解密操作时记录的能量轨迹,使用这些轨迹来分析固定时刻设备的能量消耗,并将能量消耗视作被处理数据的函数,从而恢复出密码设备中的密钥,因此攻击者不必了解要攻击设备的具体信息[10]。DPA 攻击通常包括以下 4 个步骤:

步骤 1　选择密码设备执行算法的某个中间值 D,构造差分函数 $D(m,k)$,其中,m 为已知的非常量数据,k 为待猜测的子密钥。

步骤 2　对密码设备输入大量不同数据,测量能量消耗轨迹 m_i 及 t_i。

步骤 3　猜测子密钥 k,计算假设中间值,按照选定模型将中间值映射为能量消耗值,将轨迹 t_i 分类,与实际能量迹比对,计算平均功耗轨迹 t,并与分类的轨迹进行差分运算,公式如下:

$$\Delta_D(j) = \frac{\sum_{i=1}^{n} D(m_i,k_i) \times t_i(j)}{\sum_{i=1}^{n} D(m_i,k_i)} - \frac{\sum_{i=1}^{n} (1 - D(m_i,k_i)) \times t_i(j)}{\sum_{i=1}^{n} (1 - D(m_i,k_i))} \tag{5}$$

分析差分结果,若出现明显尖峰则猜测正确,否则重复步骤 3。

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。

1.3　秘密共享抵抗功耗攻击原理

文献[11]提出秘密共享的概念,并给出了 (k,n) 门限秘密共享方案。其基本思想是把一个完整的秘密分成若干份分给 n 个参与者,只有参与者达到 k 个或 k 个以上才可以重构这个秘密,k 称为方案的门限值。下面介绍秘密共享如何抵御 DPA 攻击[6]。

定义符号⊕为域 $GF(2^m)$ 上的异或运算,$\sum$ 为实数加法运算,$\bar{\boldsymbol{x}}$ 表示矢量 $(x_1,x_2,\cdots,x_n)$,$\overline{\boldsymbol{x}_i}$ 表示 $\bar{\boldsymbol{x}}$ 中缺 $\boldsymbol{x}_i$ 的矢量 $(x_1,x_2,\cdots,x_{i-1},x_{i+1},\cdots,x_n)$,$\Pr(t(\bar{\boldsymbol{x}}) = T)$ 表示 t 值取 T 时的概率。受秘密共享和门限方案系统启发,把 x 分成 n 份,下式成立:

$$x = \bigoplus_{i=1}^{n} x_i \tag{6}$$

使用 (n,n) 秘密共享方案,为了唯一地确定 x,需要全部的 n 份秘密。在一个完美的 (n,n) 秘密共享方案中,即使掌握 $n-1$ 份秘密,仍然得不到关于 x 的任何信息。在一个 (k,t,n) 斜坡方案中,至少需要 t 个诚实方才能恢复秘密,然而 k 个恶意方也可恢复秘密。这里使用 $(1,n,n)$ 斜坡方案,条件概率分布 $\Pr(\bar{\boldsymbol{X}} \mid \boldsymbol{X})$ 均匀,因此,得到条件 1:

$$\forall \bar{\boldsymbol{X}}:\Pr(\bar{\boldsymbol{x}}=\bar{\boldsymbol{X}})=c\Pr(x=\bigoplus_{i=1}^{n}x_i) \tag{7}$$

其中,c 是常量,确保 $\sum\bar{\boldsymbol{X}}\Pr(\bar{\boldsymbol{x}}=\bar{\boldsymbol{X}})=1$。

将 $\Pr(\bar{\boldsymbol{X}}\mid\boldsymbol{X})$ 进行扩展,得到条件 2:

$$\Pr(\bar{\boldsymbol{x}}=\bar{\boldsymbol{X}},\bar{\boldsymbol{y}}=\bar{\boldsymbol{Y}},\cdots)=c\Pr(x=\bigoplus_{i}X_i,y=\bigoplus_{i}Y_i,\cdots) \tag{8}$$

定义 $z=L(x)$ 为域 $GF(2^m)$ 上的线性变换,将 x 和 z 分成 n 份单独处理,即 $z_i=L(x_i),0\leqslant i<n$。

$$z=\bigoplus_{i=1}^{n}z_i=\bigoplus_{i=1}^{n}L(x_i)=L(\bigoplus_{i=1}^{n}x_i)=L(x) \tag{9}$$

该线性变换面对侧信道攻击时不会泄露任何信息,称为安全电路,其特征是任意输出的一份 z_i 只与 x_i 有关,同理,多个输入时有 $z=LL(x,y,\cdots)$,z_i 只与 $(x_i,y_i\cdots)$ 有关。考虑到线性变换与非线性变换具有相似的特征,可构造非线性变换的安全电路。若对变换 $z=LL(x,y,\cdots)$ 来说,z_i 不完全取决于 $(x_i,y_i\cdots)$ 的值,则 z_i 与 $x,y,\cdots$ 不相关,z_i 的计算也不会泄露关于 $x,y,\cdots$ 的信息,通过附加条件,可确保 z_i 与 z 不相关。

定义 $z=N(x,y,\cdots)$ 为域 $GF(2^m)$ 上的非线性变换,函数 $f_1,f_2,\cdots,f_n$ 需满足以下 3 个性质:

性质 1(不完整性) 每个函数至少独立于输入变量 $x,y,\cdots$ 的一份:

$$z_1=f_1(x_2,x_3,\cdots,x_n,y_2,y_3,\cdots,y_n,\cdots) \tag{10}$$

$$z_2=f_2(x_1,x_3,\cdots,x_n,y_1,y_3,\cdots,y_n,\cdots) \tag{11}$$

$$\vdots$$

$$z_n=f_n(x_1,x_3,\cdots,x_{n-1},y_1,y_3,\cdots,y_{n-1},\cdots) \tag{12}$$

性质 2(正确性) n 份输出之和等于正确的输出:

$$z=\bigoplus_{i=1}^{n}z_i=\bigoplus_{i=1}^{n}f_i(\cdots)=N(x) \tag{13}$$

性质 3(均匀性) 若所有输入 $x,y,\cdots$ 和输入分组 $x_i,y_i,\cdots$ 满足条件 2,且条件概率:

$$\Pr(\bar{z}=\bar{Z}\mid z=\bigoplus_{i}Z_i) \tag{14}$$

是一个常量,则称 $z=N(x,y,\cdots)$ 满足均匀性。

由于前一阶段的输出是下一阶段的输入,性质 3 确保了数据在处理过程中能保持同步。

定理 1 满足性质 1 和性质 2 的非线性变换,若有 s 个变量,其最小分组数 n 满足:

$$n\geqslant 1+s \tag{15}$$

这表明要实现一个非线性变换至少需要分成 3 组[12]。由于并非所有变量都相互独立,还可能存在其他需要更少分组的解决方法,据此推论:

推论 1 域 $GF(2^m)$ 上的非线性变换,若具有 u 个变量,则最大分组数 n 满足:

$$n=1+2^{mu} \tag{16}$$

条件 2 表明,$\bar{\boldsymbol{x}}$、$\bar{\boldsymbol{y}}$ 的联合分布出现任何偏差都会导致 x、y 的联合分布产生偏差,据此可证明:

定理 2 在一个电路实现中,若一组函数满足性质 1 和性质 2,且输入满足条件 2,则所有的中间值结果独立于输入 $x,y,\cdots$ 和输出 z。同时,任一单个函数 f_i 的功耗和其他特性独立于 $x,y,\cdots$ 和 z。

定理 2 表明,完全可以构造出一组函数,满足任一单个函数 f_i 的功耗与输入 $x,y,\cdots$ 和输出 z 无关,此时功耗与数据之间不存在依赖性,DPA 攻击不再发挥作用[12]。

2 共享 S 盒实现

在秘密共享 S 盒中,需要将仿射变换的输入输出分成 3 组进行运算[13],由于 SM4 算法 S 盒 2 次仿射变换具有相同的循环矩阵和行向量,且仿射变换本身具有线性关系,因此用秘密共享函数代替 2 次仿射变换,完成对输入输出数据的分组[14]。

2.1 秘密共享函数

利用 1.1 节给出的循环矩阵和行向量数值,定义秘密共享函数 $u=L(v,s,r)$,v 为输入,u 为输出,u、v 均为 8 bit 数据,令 $u=(a,b,c,d,e,f,g,h)$,$v=(j,k,l,m,n,o,p,q)$,则有以下公式成立:

$$a=j+k+m+p+q+1 \tag{17}$$

$$b=j+k+l+n+q+1 \tag{18}$$

$$c=j+k+l+m+o \tag{19}$$

$$d=k+l+m+n+p \tag{20}$$

$$e=l+m+n+o+q+1 \tag{21}$$

$$f=j+m+n+o+p \tag{22}$$

$$g=k+n+o+p+q+1 \tag{23}$$

$$h=j+l+o+p+q+1 \tag{24}$$

将共享函数分为 3 组,有 $u_1=L_1(v_2,s_2,r)$。

$$a_1=j_2+k_2+m_2+p_2+q_2+s_2+r \tag{25}$$

$$b_1=j_2+k_2+l_2+n_2+q_2+s_2+r \tag{26}$$

$$c_1=j_2+k_2+l_2+m_2+o_2+r \tag{27}$$

$$d_1=k_2+l_2+m_2+n_2+p_2+r \tag{28}$$

$$e_1=l_2+m_2+n_2+o_2+q_2+s_2+r \tag{29}$$

$$f_1=j_2+m_2+n_2+o_2+p_2+r \tag{30}$$

$$g_1=k_2+n_2+o_2+p_2+q_2+s_2+r \tag{31}$$

$$h_1=j_2+l_2+o_2+p_2+q_2+s_2+r \tag{32}$$

同理,有 $u_2=L_2(v_3,s_3,r)$。

$$a_2=j_3+k_3+m_3+p_3+q_3+s_3+r \tag{33}$$

$$b_2=j_3+k_3+l_3+n_3+q_3+s_3+r \tag{34}$$

$$c_2=j_3+k_3+l_3+m_3+o_3+r \tag{35}$$

$$d_2=k_3+l_3+m_3+n_3+p_3+r \tag{36}$$

$$e_2=l_3+m_3+n_3+o_3+q_3+s_3+r \tag{37}$$

$$f_2=j_3+m_3+n_3+o_3+p_3+r \tag{38}$$

$$g_2=k_3+n_3+o_3+p_3+q_3+s_3+r \tag{39}$$

$$h_2=j_3+l_3+o_3+p_3+q_3+s_3+r \tag{40}$$

同理,有 $u_3=L_3(v_1,s_1,r)$。

$$a_3=j_1+k_1+m_1+p_1+q_1+s_1+r \tag{41}$$

$$b_3=j_1+k_1+l_1+n_1+q_1+s_1+r \tag{42}$$

$$c_3 = j_1 + k_1 + l_1 + m_1 + o_1 + r \tag{43}$$
$$d_3 = k_1 + l_1 + m_1 + n_1 + p_1 + r \tag{44}$$
$$e_3 = l_1 + m_1 + n_1 + o_1 + q_1 + s_1 + r \tag{45}$$
$$f_3 = j_1 + m_1 + n_1 + o_1 + p_1 + r \tag{46}$$
$$g_3 = k_1 + n_1 + o_1 + p_1 + q_1 + s_1 + r \tag{47}$$
$$h_3 = j_1 + l_1 + o_1 + p_1 + q_1 + s_1 + r \tag{48}$$

其中,s 是常量,满足 $s = s_1 + s_2 + s_3$,共享函数实现流程如图 2 所示。

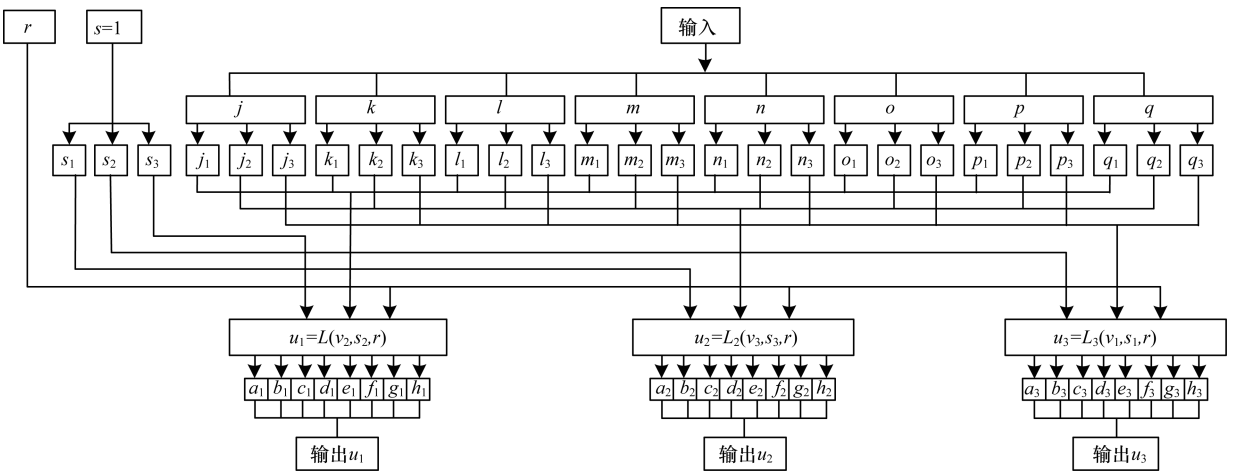

图 2　共享函数实现流程

2.2　秘密共享 S 盒实现

由 1.3 节可知,满足秘密共享 S 盒 3 个分组性质的最小分组数为 3,因此,选用 3 个分组来构造 SM4 算法共享 S 盒时可以减少运算量,降低功率消耗和面积,同时保证其抵抗高阶 DPA 攻击及 glitch 攻击的能力。本文基于秘密共享的 SM4 算法S 盒具体实现方案分为 3 个阶段,如图 3 所示。

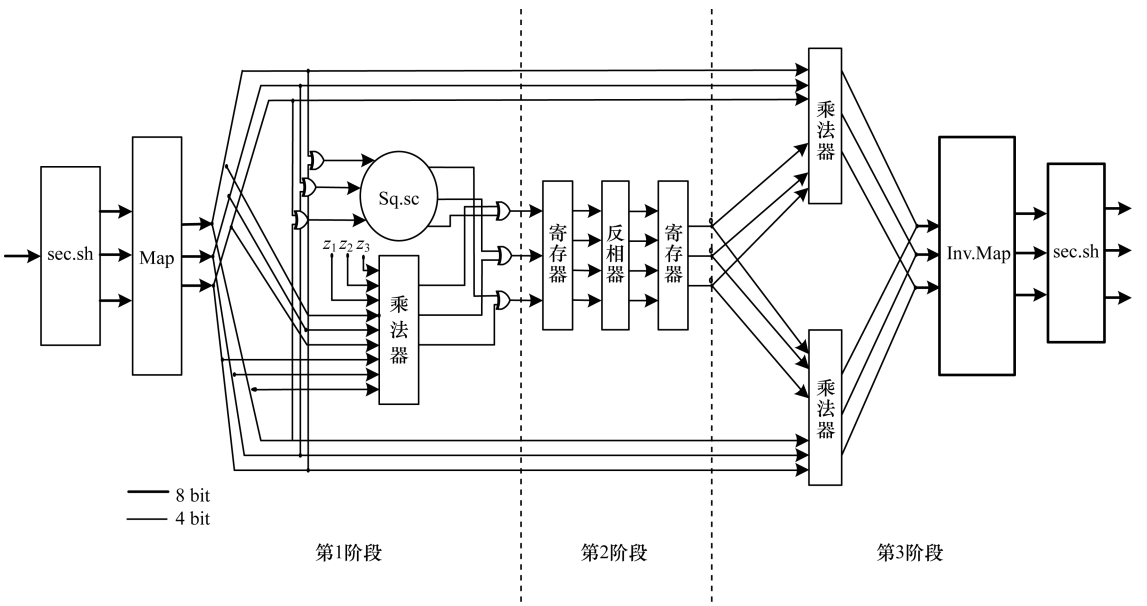

图 3　秘密共享 S 盒具体实现

第 1 阶段,通过秘密共享函数将进入 S 盒的 8 bit信息分成 3 组 8 bit 信息,进入 Map 部分进行第同构映射变换,得到 3 组 8 bit 信息 a_1、a_2、a_3,再将每组 8 bit 信息分成 2 组 4 bit 信息,共得到 6 组 4 bit 信息 b_1、c_1、b_2、c_2、b_3、c_3。将 6 组 4 bit 信息分成 $S_1(b_1,b_2,b_3)$ 和 $S_2(c_1,c_2,c_3)$ 2 组,S_1、S_2 进行3 次异或运算后输入平方计数器,得到 3 组 4 bit 信息 d_1、d_2、d_3,同时将 S_1、S_2 输入 $GF(2^4)$ 乘法器,得到3 组 4 bit信息 e_1、e_2、e_3。d_1、d_2、d_3 和 e_1、e_2、e_3 进行 3 次异或运算后得到 f_1、f_2、f_3,将 f_1、f_2、f_3 输入寄存器进行分组。

在乘法器中,由于 2 个输入每个分成 3 组相乘时,不满足均匀性的性质[12],因此,这里采用虚拟值的方法[15],在进入乘法器时添加一个 4 bit 虚拟值 z,并将 z 分成 3 组 4 bit 信息 z_1、z_2、z_3,与 b_1、c_1、b_2、c_2、b_3、c_3 同时进入 $GF(2^4)$ 乘法器,得到 e_1、e_2、e_3,运算公式如下:

$$e_1 = b_2c_2 \oplus b_2c_3 \oplus b_3c_2 \oplus b_2z_2 \oplus b_3z_3 \oplus c_2z_2 \oplus c_3z_3 \tag{49}$$

$$e_2 = b_3c_3 \oplus b_1c_3 \oplus b_3c_1 \oplus b_3z_3 \oplus b_1z_1 \oplus c_3z_3 \oplus c_1z_1 \tag{50}$$

$$e_3 = b_1c_1 \oplus b_1c_2 \oplus b_2c_1 \oplus b_1z_1 \oplus b_2z_2 \oplus c_1z_1 \oplus c_2z_2 \tag{51}$$

第 2 阶段,把 f_1、f_2、f_3 输入寄存器重新分组,得到 4 组 4 bit 信息 f_1^*、f_2^*、f_3^*、f_4^*,而后输入反相器,采用分解法求逆[16],得到 4 组 4 bit 信息 g_1^*、g_2^*、g_3^*、g_4^*。分解法可有效减少运算次数,降低功耗,具体方程式如下:

$$(g_1^*, g_2^*, g_3^*, g_4^*) = \mathrm{Inv}(f_1^*, f_2^*, f_3^*, f_4^*) \tag{52}$$

$$g_1^* = f_2^* \oplus f_3^* \oplus f_1^* f_3^* \oplus f_2^* f_3^* \oplus f_2^* f_3^* f_4^* \tag{53}$$

$$g_2^* = f_4^* \oplus f_1^* f_3^* \oplus f_2^* f_3^* \oplus f_2^* f_4^* \oplus f_1^* f_3^* f_4^* \tag{54}$$

$$g_3^* = f_1^* \oplus f_2^* \oplus f_1^* f_3^* \oplus f_1^* f_4^* \oplus f_1^* f_2^* f_4^* \tag{55}$$

$$g_4^* = f_2^* \oplus f_1^* f_3^* \oplus f_1^* f_4^* \oplus f_2^* f_4^* \oplus f_1^* f_2^* f_3^* \tag{56}$$

将 g_1^*、g_2^*、g_3^*、g_4^* 输入寄存器重新得到 3 组 4 bit 信息 g_1、g_2、g_3,输出进入第 3 阶段。

第 3 阶段,将 b_1、b_2、b_3 和 g_1、g_2、g_3 输入 $GF(2^4)$乘法器,采用虚拟值方法得到 3 组 4 bit 信息 h_1、h_2、h_3,同时将 c_1、c_2、c_3 和 g_1、g_2、g_3 输入 $GF(2^4)$乘法器得到 3 组 4 bit 信息 k_1、k_2、k_3。将得到的 h_1、h_2、h_3 与 g_1、g_2、g_3 输入 Inv. AT. lin. lmap 部分,进行第 2 次仿射变换及相应的线性变换得到 3 组 8 bit 信息 m_1、m_2、m_3,将 m_1、m_2、m_3 相加即得 S 盒最终输出结果。

3 安全性分析及实验验证

3.1 安全性分析

3.1.1 抗高阶功耗攻击分析

高阶 DPA 是利用密码设备内部的几个中间值的联合泄露信息进行的攻击[17]。本文采用基于秘密共享方法的设计实现了 SM4 的抗功耗攻击 S 盒方案,整个实现过程满足秘密共享对于分组的要求。在第一阶段,输入被分成了 3 组进行运算,其中间值 a_1、a_2、a_3 与 f_1、f_2、f_3 符合秘密共享分组要求,即满足正确性、不完整性和均匀性。因此,有 f_1、f_2、f_3 互相独立且与 a_1、a_2、a_3 线性无关。在第 2 阶段,引入分解法,中间值 f_1、f_2、f_3 求逆得到 g_1、g_2、g_3,因此有 g_1、g_2、g_3 互相独立且与 f_1、f_2、f_3 线性无关。在第 3 阶段,由 g_1、g_2、g_3 得到 m_1、m_2、m_3,满足 m_1、m_2、m_3 均互相独立且与 g_1、g_2、g_3 线性无关。因此,当攻击者进行 HO-DPA 攻击时,仅能得到 3 组独立无关的中间值,无法通过对这 3 组中间值的统计分析来获取关于密钥的任何信息,从而达到抵御高阶功耗攻击的目的。因此,本文所设计的 SM4 加密方案能够有效抵御 HO-DPA 攻击。

3.1.2 抗 glitch 攻击分析

glitch 攻击是利用电路输入信号到达时间的不同,进而引起输出产生临时状态的特点来获取信息[17]。本文采用基于秘密共享的 SM4 抗功耗攻击 S 盒方案,由于 S 盒输入数据被分成 3 个分组进行计算,因此发生在一个分组上的 glitch 与发生在另一个分组上的 glitch 并不同步,对攻击者而言,很难通过一组分组上的 glitch 攻击来获取整个电路的信息。因此,本文的秘密共享 S 盒方案能够抵御 glitch 攻击。

3.2 实验结果

基于 TSMC 60 nm 工艺,在相同的条件下,仿真实现 SM4 算法的复合域掩码方案,同时实现本文的秘密共享 S 盒方案。为了便于比较,复合域掩码方案与本文掩码方案均在 TSMC 60 nm 工艺下,约束频率是 50 MHz,性能指标如表 1 所示。

表 1 算法电路实现比较

方案	工艺/nm	频率/MHz	实现面积/mm^2
文献[5]方案	60	50	0.21
本文方案	60	50	0.20

本文 S 盒方案采用秘密共享方法实现,将 S 盒输入分成 3 组,并保证分组运算后的中间值结果互相独立,即分组运算后的中间值功耗与分组前的中间值之间不相关,保证了本文 S 盒具有抵御高阶 DPA 攻击及 glitch 攻击的安全性。现有方案的 S 盒比较如表 2 所示。

表 2 现有方案的 S 盒比较

特征	文献[5]方案	文献[18]方案	本文方案
实现方法	查找表	PPRM	复合域
一阶 DPA 安全	是	是	是
高阶 DPA 安全	否	否	是
Glitch 攻击安全	否	否	是

4　结束语

SM4 作为我国自行设计的分组密码标准,采用 32 轮非线性迭代结构,具有很高的安全性,但仍然面临 DPA 攻击的巨大风险。现有的防护 DPA 攻击的掩码方案虽然很多,但多数局限性较强,且防护能力较弱。本文通过对秘密共享抵抗 DPA 攻击的原理分析及文献[8]提出的 AES 算法 S 盒共享实现方案的研究,构造了一个适用于 SM4 算法实现的新型共享 S 盒,新的 S 盒通过利用秘密共享函数代替仿射变换,在乘法器分组中采用虚拟值法,并在反相器中引入分解法,使得本文实现方案具有较少的运算次数和较低的空间占比。安全性分析和实验结果表明,该方案对高阶 DPA 攻击乃至 glitch 攻击具有较强的抵抗能力。

参考文献

[1] 吕述望,苏波展,王　鹏,等. SM4 分组密码算法综述[J]. 信息安全研究,2016(11):995-1007.

[2] LIU F,JI W,HU L,et al. Analysis of the SM4 block cipher[C]//Proceedings of ACISP'07. Townsville, Australia:[s. n.],2007:158-170.

[3] BAI X,GUO L,LI T. Differential power analysis attack on SM4 block cipher[C]//Proceedings of ICCSC'08. Washington D. C.,USA:IEEE Press,2008:613-617.

[4] BAI X, XU Y, GUO L. Securing SM4 cipher against differential power analysis and its VLSI implementation[C]//Proceedings of the 11th IEEE International Conference on Communications Systems. Washington D. C., USA:IEEE Press,2008:167-172.

[5] LIANG H,WU L,ZHANG X,et al. Design of a masked S-box for SM4 based on composite field[C]//Proceedings of the 20th International Conference on Computational Intelligence and Security. Washington D. C., USA: IEEE Press,2014:387-391.

[6] NIKOVA S,RECHBERGER C,RIJMEN V. Threshold implementations against side-channel attacks and glitches[C]//Proceedings of International Conference on Information and Communications Security. Berlin, Germany:Springer,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]//Proceedings of Advances in Cryptology-EUROCRYPT'11. Berlin, Germany: Springer,2011:69-88.

[8] BILDIN B,GIERLICHS B,NIKOVA S,et al. A more efficient AES threshold implementation [C]// Proceedings of International Conference on Cryptology. Berlin,Germany:Springer,2014:267-284.

[9] KOCHER P C, JAFFE J, JUN B. Differential power analysis[C]//Proceedings of International Cryptology Conference on Advances in Cryptology. Berlin, Germany:Springer,1999:388-397.

[10] MANGARD S, OSWALD E, POPP T. Power analysis attacks:revealing the secrets of smart cards[M]. Berlin, Germany:Springer,2010.

[11] SLINKO A. Secret sharing [M]. Berlin, Germany: Springer,2015.

[12] NIKOVA S, RIJMEN V, SCHLAFFER M. Secure hardware implementation of nonlinear functions in the presence of glitches[J]. Journal of Cryptology, 2011, 24(2):292-321.

[13] 冷建伟,李　鹏. 基于自适应特征分布更新的压缩跟踪算法[J]. 计算机工程,2018,44(2):264-270.

[14] WANG Y,YUAN Z,LI Z,et al. Secret sharing based countermeasure for AES S-box[C]//Proceedings of IEEE International Symposium on Integrated Circuits. Washington D. C.,USA:IEEE Press,2011:504-507.

[15] BILGIN B,NIKOVA S,NIKOVA V,et al. Threshold implementations of small S-boxes[J]. Cryptography and Communications,2015,7(1):3-33.

[16] 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(1):191-196.

[17] 袁　征. 功耗攻击防御技术在分组密码中的应用研究[D]. 长沙:湖南大学,2012.

[18] 牛砚波,蒋安平. 一种低功耗抗差分功耗分析攻击的 SM4 算法实现[J]. 微电子学与计算机,2014,31(9):28-32.

编辑　索书志

(上接第 147 页)

[13] 陈雪娇,王　攀,刘世栋. 网络应用流类别不平衡环境下的 SSL 加密应用流识别关键技术[J]. 电信科学,2015,31(12):83-89.

[14] CHAWLA N V, BOWYER K W, HALL L O, et al. SMOTE: synthetic minority over-sampling technique[J]. Journal of Artificial Intelligence Research, 2002, 16(1): 321-357.

[15] DRAPER-GIL G, LASHKARI A, MAMUN M, et al. Characterization of encrypted and VPN traffic using time-related features [C]//Proceedings of the 2nd International Conference on Information Systems Security and Privacy. Setúbal, Portugal: Science and Technology Publications,2016,407-414.

[16] GitHub,Inc. Keras:deep learning for humans [EB/OL]. [2018-06-06]. https://github.com/fchollet/keras.

[17] ABADI M,AGARWAL A,BARHAM P,et al. TensorFlow: large-scale machine learning on heterogeneous systems [EB/OL]. [2018-06-06]. http://cn.arxiv.org/pdf/1603.04467v1.

[18] KIM H, CLAFFY K, FOMENKOV M, et al. Internet traffic classification demystified: myths, caveats, and the best practices [C]//Proceedings of the 2008 ACM CoNEXT Conference. New York, USA: ACM Press, 2008:1-12.

编辑　刘盛龄