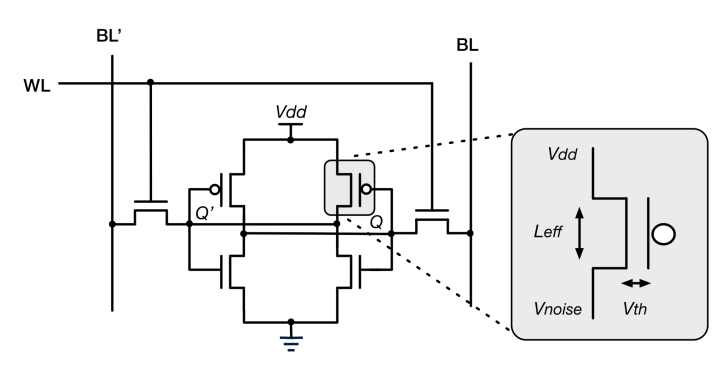
## SRAM PUF设计原理

(1).可行性分析

在本论文中，使用SRAM(Static Random Access Memory，静态随机存取存储器)来实现PUF功能，必然要首先考虑SRAM是否适合作为实现PUF的载体以及SRAM是否能够产生足够的熵来作为密钥生成源，所以在接下来介绍了SRAM的原理，然后介绍了衡量SRAM PUF性能的几个重要参数，并通过实际测量几块SRAM上电初值来进行实例参考。

1.SRAM PUF原理介绍

SRAM(Static Random Access Memory，静态随机存取存储器)是现代电子设备中必不可少的组成部分，SRAM是由一位位的存储单元规则排列而成，其每一位存储单元是由图2-1所示的6个CMOS晶体管组成，包括两个交叉耦合的反相器和两个读写控制开关。

图2-1 SRAM存储单元结构图（需要重新画图）

SRAM未上电时，由于未存储电荷，所以QQ’=“00”；在上电的瞬间，反相器发生反转，所以QQ’=“01”或“10”。在理想情况下，QQ’=“01”或者“10”是随机的，即两者的概率是相等的。但是SRAM存储单元的上电初值受到两类因素的影响。第一种是晶体管本身的因素。在SRAM的制造过程中，由于工艺的离散性，比如掺杂浓度的不同、晶体管长宽比不同等因素，每位存储单元并不是完全对称的，其阈值电压是不一样的，从而使得上电初值偏向于“0”或者偏向于“1”。第二种是外界环境因素。外界温度、电源电压的波动甚至封装的影响都会造成上电初值的变化。理论上来说，SRAM对称结构越匹配，对外界噪声就越敏感。图2-2(a)和(b)表示了存储单元的上电倾向和噪声影响。其中，(a)图为“1”偏存储单元，说明了存储单元结构的不对称性的强度大大超过了噪声对节点状态的影响。图(b)为无偏存储单元，噪声的影响超过了制造存储单元结构的不对称性。

图2-2 (a) 1偏存储单元 (b)无偏存储单元

为了验证SRAM 作为PUF的可行性，对实际SRAM进行多次上电初值读取，读取大小为256Byte，统计其中“0”和“1”的个数，得到图2-3所示的直方图。

图2-3 SRAM上电分布图.

在SRAM芯片中，将SRAM存储单元的地址作为PUF的激励，将地址对应的上电初值作为PUF的响应，则该SRAM就能实现PUF的功能。当SRAM未初始化时，可作为一个熵源。当SRAM的存储单元无偏时，其上电初值具有很高的随机性。但是，SRAM中有些存储单元是“0”偏或“1”偏的，所以，SRAM的上电初值还需要进行进一步的处理之后，才能产生高质量的真随机数种子。

2.片内汉明距离

汉明距离是以理查德·卫斯里·汉明的名字命名的。在信息论中，两个等长字符串之间的汉明距离是两个字符串对应位置的不同字符的个数。例如：1011101 与 1001001 之间的汉明距离是 2。对于同一个 PUF来说，外界噪声和环境因素的影响使得相同的激励会产生不同的响应。片内汉明距离是指对同一个PUF实体连续输入两次相同的激励，其对应的响应之间的汉明距离，它反映了同一个PUF的稳定性。在理想情况下，我们希望片内汉明距离是“0”，即SRAM在相同的地址下上电初值不会变化。但在实际情况下，片内汉明距离是永远存在的，我们希望它越小越好。通过对同一片SRAM 256Byte大小的相同地址的上电初值进行实测，得到了五万组数据，以第一组数据为参考，计算剩下数据和第一组数据的汉明距离，绘制图像如图2-4所示。

图 2-4 片内汉明距离

3.片间汉明距离

片间汉明距离指的是不同的PUF实体，输入相同的激励得到的响应之间的汉明距离，它反映了不同PUF之间的相关性。由于PUF具有唯一性，理论上两块不同的PUF是独立不相关的，所以汉明距离与总长度之间比值的理想值为50%。通过选取五片不同的存储器，对256Byte大小的上电初值进行测量，以第一片SRAM作为参考，得到片间汉明距离与总长度的比值，绘制图像如图2-5所示。

图 2-5 片间汉明距离

(2).基于ECC的SRAM PUF整体设计架构

将SRAM PUF作为密钥产生器使用，会面临两个问题：一是SRAM的上电初始值并不全是均匀分布的，即可能存在“0”偏或“1”偏的情况；二是由于噪声等外界因素的影响，两次SRAM之间的上电初值存在一定的汉明距离。为了解决这两个问题，引入模糊提取器【1】Fuzzy extractors: How to generate strong keys from biometrics and other noise的概念。

1. 模糊提取器

模糊提取器是Yevgeniy Dodis等人在2008年提出来的概念。它能够以容错的方式从它的输入w中提取中随机均匀分布的密钥K。所谓容错的方式是指当它的输入发生微小的变化变为w’时，密钥K能够被精确地再生出来。为了帮助从w’中再生出K，模糊提取器需要输出一个辅助数据P，P是可以安全公开的，而且不会影响到密钥K的随机性。模糊提取器的抽象模型如图2-6所示。

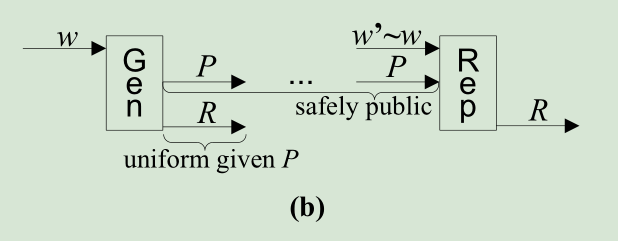


图2-6 模糊提取器的抽象模型

模糊提取器的工作过程分为两个阶段：注册阶段(Generation phase)和密钥重构阶段(Regeneration phase)。在注册阶段，如图2-7(a)所示，主要完成两件事情，一是密钥K的生成，二是辅助数据P的生成。由于SRAM有些存储单元的上电初值并不是随机均匀分布的，所以输入数据w要经过压缩函数进一步压缩转化，最终输出全熵的密钥K。辅助数据P的存在是为了在再生阶段帮助恢复密钥K，P是由输入数据X和经过编码后的具有纠错功能的纠错码C异或生成的。在密钥重构阶段，如图2-7(b)所示，辅助数据与再次读取的相同地址的具有噪声的SRAM上电值w’进行异或，得到具有噪声的纠错码C’，然后经过纠错算法的译码得到正确的纠错码C，再次和辅助数据P进行异或操作，恢复出原始的上电初值w，再经过压缩函数得到相同的密钥K。通过对注册阶段和密钥重构阶段的分析，可知要实现模糊提取器的功能，必须具备检错与纠错模块(Error Check and Correct module)和压缩模块这两个基本的模块。在下文中将对ECC模块和压缩模块所采用的算法作出具体的选择。

图2-7 (a)注册阶段 (b)密钥重构阶段

1. 压缩算法的选择

压缩算法的存在是为了将非均匀随机分布的SRAM上电初始值进行处理，得到映射分布均匀和差分分布均匀的具有全熵的真随机密钥。

* 1. 散列函数简介

单向散列函数(也称伪随机函数(Pseudo Random Function, PRF))是密码学和信息安全领域的一个基本算法，它可以通过一个称为压缩(compression)的处理过程将任意长度的输入映射一个成固定长度的输出，该输出称为散列值，我们也常称输入为消息，输出为摘要。这种形式的压缩并不是那种典型的数据压缩，而是一种不可逆的映射。在密码学意义上，一个安全的哈希函数应具备两种基本的性质：它必须是单向的并且必须是碰撞约束的。

单向性是指给定一个散列函数的输出，推导出函数的输入信息在多项式时间内是不可能完成的，这意味着散列函数是非线性的。碰撞约束是指给定一个散列函数的输出，找出另外一个能产生相同输出的输入(称为碰撞)是困难的。碰撞约束分为两类，分别是预映射碰撞约束和第二预映射碰撞约束。预映射碰撞约束(pre-image collision resistance)(如图2-8所示)是指给定一个输出Y，找到另一个输入X’，使得X’的散列值等于Y是困难的。第二预映射碰撞约束(second pre-image collision resistance)(如图2-9所示)是指找到两个消息M1(给定的)和M2(随机选取的)，使它们的散列值完全相等是困难的。

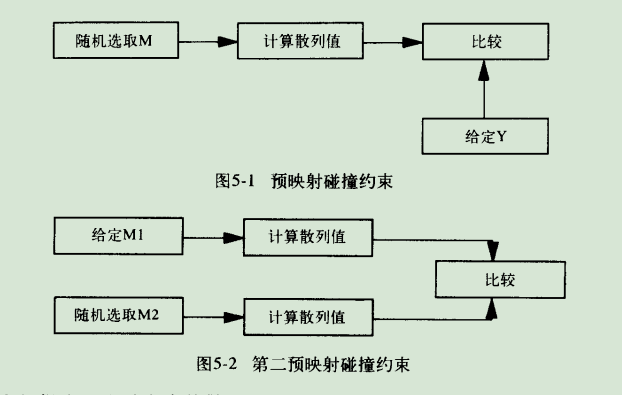


图2-8 预映射碰撞约束

图2-9 第二预映射碰撞约束

安全的哈希算法除了这两项最基本的性质外，还应具有映射分布均匀性和差分分布均匀性。散列结果中，值为“0”的bit和值为“1”的bit总数应大致相等；输入中1bit的改变，将引起结果中一半以上的bit的改变，这又叫做“雪崩效应”(avalanche effect)；要使散列结果出现1bit的变化，则输入中至少有一半以上的bit必须发生变化。所以其实质是使输入中的每一个bit的信息，尽量均匀的反应到输出的每一位bit中，而输出中的每一个bit，都是输入中尽可能多bit的信息一起作用的结果。

目前最典型的哈希算法有MD4，MD5，SHA1，SHA2。其中MD5，SHA1，SHA2均是以MD4为基础设计的。目前已经证明MD4和MD5是不安全的，因为它们不是抗碰撞的。其中，MD5散列算法在2008年国际密码学大会上宣布被山东大学的王小云教授成功破解【2Crypto 2004. Security & Privacy】。美国国家标准技术研究所（National Institute of Standard and Technology, NIST）发表声明SHA1没有被破解，并且没有足够的理由怀疑它会很快被攻破，而仅仅在发布声明的一周之后，王小云教授就宣布了破译SHA1的消息【3Finding collisions in the full SHA-1】。SHA1和SHA2属于安全哈希算法(Secure Hash Algorithm, SHA)家族。目前SHA2系列算法是安全的。SHA2系列算法包括：SHA-224、SHA-256、SHA-384和SHA-512四种算法，这四种哈希算法的计算过程是一样的，不同之处在于操作数长度、初始向量值、常数值和最后产生信息摘要的长度。在本设计中，采用SHA256算法，生成256bit的密钥数据。

* 1. SHA256算法原理

SHA256算法要求输入报文的最大长度不超过bit，输入按照512bit分组进行处理，产生256bit的报文摘要。整个过程可以分为两个阶段：预处理阶段和计算阶段。在预处理阶段中，进行消息的填充、附加长度值和初始化哈希值。在计算阶段，利用初始哈希值、常量和输入消息进行压缩运算，进行64轮的迭代，每一轮迭代生成的哈希值用于下一轮的计算中，最后一轮得到的哈希值就是最终的“信息摘要”。

* + 1. 预处理阶段

1. 附加填充比特

在输入消息(M)进入哈希计算前，需要按照规则填充成一定的长度，确保填充后的消息为512的整数倍，同时要求填充后的数据中拥有输入消息M原有数据位数的信息。填充规则为：假设消息M长度为L，则在L末尾增加一个比特位，并赋予该比特质为“1”，同时增加K个值为“0”的比特位，使其满足L+1+K=448mod512。然后再在其后面添加64bits的二进制数据块，其值等于原消息长度L。这样，填充后的消息总长度为512的整数倍。

如图2-10所示，假设输入消息为“abc”，其ASCII码值如图中所示共24bits，按照填充规则，在“abc”后添加一位比特，其值为“1”，同时添加423个值为“0”的比特位，使其满足24+1+423=448mod512，然后在最后添加64bit的比特位，其值为24，即输入消息“abc”的长度。

图2-10 附加填充比特结果

1. 分割填充的信息块

对于SHA256而言，输入是按照512bit的长度进行处理的。所以要对填充后的消息按照512bit的长度进行分组，分为N个512bit的消息块：，，······ 。对每个512bit的消息块，再分割成16个32bit字(0≤t≤15,0≤i≤N)用于哈希计算。例如可以将分割为：、、、······。

1. 初始化哈希值

SHA256的哈希初值是8个32位字，这些初值是对自然数中前8个质数（2,3,5,7,11,13,17,19）的平方根的小数部分取前32bit而来。使用一个256bits的缓存{abcdefgh}来存放散列函数的中间及最终结果，表示如下：

a=0x6A09E667

b=0xBB67AE85

c=0x3C6EF372

d=0xA54FF53A

e=0x510E527F

f=0x9B05688C

g=0x1F83D9AB

h=0x5BE0CD19

* + 1. 计算阶段

1. SHA256中的哈希常量

在SHA256的函数运算中，有64个32bits常量值{，……}用于每一轮迭代中。和8个哈希初值类似，这些常量是对自然数中前64个质数(2,3,5,7,11,13,17,19,23,29,31,37,41,43,47,53,59,61,67,71,73,79,83,89,97…)的立方根的小数部分取前32bit而来，具体数值如表1所示：

表1 SHA256哈希常量表

1. 哈希迭代计算

在迭代计算中处理512-bit(16个字)输入消息，计算过程中使用了六种基本逻辑函数，由64步迭代运算组成。每步以256-bit缓存值abcdefgh为输入，然后更新缓存内容。每步使用一个32-bit常数值和一个32-bit的消息。由于512-bit的输入消息只能分割成16个32-bit的所以每次读入第i个512-bit的消息块后，需要进行扩展，扩展规则为：

(2-1)

其中为第i个512-bit信息块中第t个32-bit分割块。

六种基本函数如下：

(2-2)

其中：

|  |  |
| --- | --- |
| 逻辑运算 | 含义 |
| ＾ | 按位“与” |
|  | 按位“补” |
|  | 按位“异或” |
|  | 循环右移n个bit |
|  | 右移n个bit |

具体计算步骤：

(a).若加密的是第一个消息块，则使用哈希初值对内部的8个32-bit数据块（a,b,c,d,e,f,g,h）进行迭代运算，对哈希值初始化，否则使用上一个消息块循环迭代计算后得到的哈希值对其进行初始化。

(b).初始化完成后，对a~h进行64次迭代运算：

For t = 0 to 63

{

h = g

g = f

f = e

e = d +

d = c

c = b

b = a

a =

}

(c).完成64次循环后，将得到的a~h与(a)中得到的初始哈希值做模的加法运算，最终得到本次运算的哈希值。

（此处有公式）

(d).重复步骤(a)(b)(c)共N次，第N次所得到的哈希值即为最终的消息摘要。

1. ECC算法的选择

从上文分析可知，两次SRAM上电初值存在一定的汉明距离，因此精确重建原始密钥，需要纠错算法来纠正发生错误的编码。Shannon的信道编码定理【a mathematical theory of communication】我们，只要信息的传输速率低于信道容量，就必然存在一种编码方法，使得信息出现差错的概率随码长的增加趋于任意小。目前主要的纠错码从构造方法上可分为分组码和卷积码。在分组码中，Haming在1950年提出了一种以奇偶校验为基础的第一个分组码，汉明码。