AN1806
应用笔记

UM324xF SHA 密钥配置指南

版本：V1.2

广芯微电子（广州）股份有限公司

http://www.unicmicro.com/

版本修订

版本	日期	描述
V1.0	2023.02.02	初始版
V1.1	2023.02.06	在“一次性编程（OTP）区域”章节，新增一条“请注意密钥只可写入一次”的注释说明； 更新软件流程。
V1.2	2023.09.11	更新 sdk 关于代码部分的修改

1 概述

散列函数把消息或数据压缩成摘要，使得数据量变小，将数据的格式固定下来。该函数将数据打乱混合，重新创建一个叫做散列值（或哈希值）的指纹。散列值通常用一个短的随机字母和数字组成的字符串来代表。

2 一次性编程（OTP）区域

在 OTP 区域中，有部分空间用于储存 KEY 和 HASH，这些空间中的数据无法被总线读取，数据通过电路直接输出到加解密模块。

0x04003B80 ~ 0x04003BFF / 0x04003C80 ~ 0x04003CFF / 0x04003D80 ~ 0x04003DFF 可以储存 SHA OTP ICV(需在 OTP 密钥控制寄存器中选择生效密钥位置)。当 0x04003X80 ~ 0x04003XAF 中的数据（X 在寄存器中选择）与 0x04003XC0 ~ 0x04003XEF 中的相同时，数据被用作 SHA OTP ICV，SHA ICV ON 标志置 1。

当数据不相同时，对应的 KEY ON 标志为 0，KEY 输出全为 1。

表 1：一次性编程（OTP）区域地址表

编号	OTP 储存区域地址	大小	OTP 锁定控制地址	备注
0	0x04003B00 ~ 0x04003BFF	256 byte	0x04003FEC	含有 SHA OTP KEY
1	0x04003C00 ~ 0x04003CFF	256 byte	0x04003FF0	含有 SHA OTP KEY
2	0x04003D00 ~ 0x04003DFF	256 byte	0x04003FF4	含有 SHA OTP KEY

注：

- 若将密钥对应的锁定控制地址的第一个字节编程为 8’h00 即可锁定对应的储存空间，使其无法编程，OTP 区域不可擦除，此功能复位生效。
- 在未锁定 OTP 锁定控制地址状态下，请注意密钥一旦写入，其操作位只能由 1 变 0。
- 密钥一旦写入不可读出，读密钥为全 1。
- 请注意密钥只可写入一次。

3 寄存器描述

3.1 OTP 密钥控制寄存器（EFC_KEYCTRL)

地址：0x40000020

复位值：0x0000 0000

位	名称	属性	复位值	描述
31	READ	W	0x0	对此位写 1，可以使 EFC 按照 0~7 位中的设定，重新从 OTP 区域读取 key。
30:12	RSV	-	-	保留
11	SHA_ICV_ON	R	0x0	指示成功从 OTP 区读取 SHA ICV
10	RSV	-	-	保留
9	AES_KEY_2_ON	R	0x0	指示成功从 OTP 区读取 AES KEY 2
8	AES_KEY_1_ON	R	0x0	指示成功从 OTP 区读取 AES KEY 1
7:6	SHA_ICV_location	R/W	0x0	选择读取 SHA ICV 的位置： 0：OTP 区域 0x4003B80 ~ 0x4003BFF 1：OTP 区域 0x4003C80 ~ 0x4003CFF 2/3：OTP 区域 0x4003D80 ~ 0x4003DFF
5:3	RSV	-	-	保留
2	AES_KEY_2_location	R/W	0x0	选择读取 AES KEY 2 的位置： 0：OTP 区域 0x40036C0 ~ 0x40036FF 1：OTP 区域 0x40037C0 ~ 0x40037FF
1	RSV	-	-	保留
0	AES_KEY_1_location	R/W	0x0	选择读取 AES KEY 1 的位置： 0：OTP 区域 0x40034C0 ~ 0x40034FF 1：OTP 区域 0x40035C0 ~ 0x40035FF

3.2 软件流程

写入密钥配置流程【以 SHA-256 为例】

1. 调用 HAL_FLASH_Program 接口写入密钥：

HAL_FLASH_Program (FLASH_TYPEPROGRAM_WORD ,shaaddr,sha_otp_icv0[i]);

2. 调用 HAL_FLASH_SetKeyctrl 接口，配置 OTP 密钥控制寄存器 EFC_KEYCTRL[7:6]：
   HAL_FLASH_SetKeyctrl(otp0,otp0,otp0);
3. 调用__HAL_FLASH_GET_KEYFLAG 接口查看是否成功生效：

```
printf$("FLASH_FLAG_SHAICV:%x\r\n",__HAL_FLASH_GET_KEYFLAG(FLASH_FLAG_SHAICV));
```

若为 1，则表示生效，此时密钥已成功写入，不可更改。