

| 方法 项目 | | Triathlon | | Simon |
|-------------------|-----------------|----------------------------|-----------------------|----------------------------|
| | | 128 bytes CBC | 128 bits CTR | 8 bytes(With Key Schedule) |
| 总计 | | 320 bytes | 24 bytes | 192 bytes |
| RAM | 明文 | 128 bytes | 16 bytes | 8 bytes |
| | 密文 | — | — | 8 bytes |
| | 轮密钥 keys | 176 bytes | — | 176 bytes |
| | 向量 vector | 8 bytes | — | — |
| | 计数器 counter | — | 8 bytes | — |
| | 保存密文 tempCipher | 8 bytes | — | — |
| 总计 | | 24 bytes | 184 bytes | 24 bytes |
| Flash (data) | Simon 常量 Z | 8 bytes | — | 8 bytes |
| | 初始密钥 masterKey | 16 bytes | — | 16 bytes |
| | 随机数 nonce | — | 8 bytes | — |
| | 轮密钥 keys | — | 176 bytes | — |
| 三部分总计 | | 542 bytes/Flash(code)/Time | 380 bytes/4243 cycles | 276 bytes/5371 cycles |
| 密钥扩展 | | 158/3482 | — | 158/3482 |
| 加密总计 | | 176/30467 | 380/4243 | 118/1885 |
| Flash (code) Time | 加载向量 vector | 16/16 | — | — |
| | 异或向量 vector | 16/128 | — | — |
| | 加载计数器 counter | — | 24/20 | — |
| | 加载随机数 nonce | — | 32/48 | — |
| | 复制计数器 counter | — | 8/4 | — |
| | 异或随机数 nonce | — | 16/16 | — |
| | 重新加载 counter | — | 8/4 | — |
| | 计数器 counter 自增 | — | 16/8 | — |
| | 加密计数器 counter | — | 148/4048 | — |
| | 明文异或 counter | — | 32/16 | — |
| | 加载明文 | 16/256 | 32/32 | 16/16 |
| | 明文加密 | 74/29568 | — | 74/1848 |
| | 更新 vector | 8/64 | — | — |
| | 密文写回 | 16/256 | 32/32 | 16/16 |
| | 循环控制及地址加载 | 30/214 | 32/17 | 12/6 |
| 解密总计 | | 208/30979 | | |
| 解密 | 加载向量 vector | 16/16 | | |
| | 加载密文 | 16/256 | — | — |
| | 保存密文 | 16/256 | | |
| | 解密 | 74/29568 | | |

批注 [a1]: 原文中是 176 bytes，作者在文章中指出了不计算明文和密文的占用的 RAM

批注 [a2]: 两个 block 算在一起
Flash: $148 = 2 * 2 * (34+3)$
一条指令 2 个 bytes;
Time: $4048 = 2 * 44 * (42 + 4)$
2: 加密数据的 block 数量;
44: 轮数;
42: 一轮加密的 cycles;
4: 每轮结束后的循环控制, brne 指令算作 2 个 cycles;

批注 [a3]:
 $29568 = 16 * 44 * (38 + 4)$
16: 加密数据的 block 数量;
44: 轮数;
38: 一轮加密的 cycles;
4: 每轮结束后的循环控制, brne 指令算作 2 个 cycles;

批注 [a4]:
 $1848 = 44 * (38+4)$
44 轮, 每轮 38 个 cycles 加上 4 个 cycles 的循环控制 (brne 算作 2 个 cycles)

批注 [a5]:
4 条获取地址指令, 2 条初始化寄存器指令;

批注 [a6]:
 $29568 = 16 * 44 * (38 + 4)$
16: 解密数据的 block 数量;
44: 轮数;
38: 一轮解密的 cycles;
4: 每轮结束后的循环控制, brne 指令算作 2 个 cycles;

| | |
|-------------|------------------------|
| 异或向量 vector | 16/128 |
| 更新 vector | 16/256 |
| 写回明文 | 16/256 |
| 循环控制 | 38/278 |

注：a) Flash(code)/Time 部分的每一项包括两部分：代码消耗的 Flash 和对应代码的运行时间；

b)最终 Flash 的消耗是 Flash(data)与 Flash(code)之和，表中是分成两部分给出的，没有合在一起；

[1] Simon_Sce1 加密循环控制及地址加载：2 条指令加载 vector 地址（执行一次），4 条指令加载加密数据地址、初始化 block 计数器（执行一次），3 条指令加载轮密钥地址、初始化轮数计数器（执行 16 次），1 条指令设置寄存器值为 0（执行 16 次），5 条指令控制 block 的循环（执行 16 次，其中一条指令 adiw 周期是 2，一条 brne 算作 2，一条 jmp 的为 3，因此一次 9 个 cycles）；

$$\text{Flash: } 30 = (2 + 4 + 3 + 1 + 5) * 2$$

$$\text{Time: } 214 = (2 + 4 + 3 * 16 + 1 * 16 + 9 * 16)$$

[2] Simon_Sce1 解密循环控制及地址加载：2 条指令加载 vector 地址（执行一次），4 条指令加载加密数据地址、初始化 block 计数器（执行一次），3 条指令加载轮密钥地址、初始化轮数计数器（执行 16 次），2 条指令加载密钥末尾地址（执行 16 次），1 条指令设置寄存器值为 0（执行 16 次），2 条指令加载临时密钥地址（执行 16 次），5 条指令控制 block 的循环（执行 16 次，其中一条指令 adiw 周期是 2，一条 brne 算作 2，一条 jmp 的为 3，因此一次 9 个 cycles）；

$$38 = (2 + 4 + 3 + 2 + 1 + 2 + 5) * 2$$

$$278 = (2 + 4 + 3 * 16 + 2 * 16 + 1 * 16 + 2 * 16 + 9 * 16)$$

[3] Simon_Sce2 加密循环控制及地址加载：1 条寄存器初始化指令（执行 1 次），2 条指令加载计数器地址（执行 1 次），2 条指令加载 nonce 地址（执行 2 次），3 条指令加载轮密钥地址、初始化轮数计数器（执行 2 次），2 条指令加载明文地址（执行 1 次），1 条 adiw 指令（执行 1 次），

$$32 = (1 + 2 + 2 * 2 + 3 * 2 + 2 + 1) * 2$$

$$17 = (1 + 2 + 2 * 2 + 3 * 2 + 2 + 1 * 2)$$