Application Note

SPC1068 Security 使用指南

Revision 2 – July 2019

目录

表格列表

图片列表

1 引言

为了降低芯片内部程序被窃取的风险，很多芯片都带有片内程序保护措施。SPC1068 在设计之初就考虑到了程序保护的重要性，为用户提供了非常强大而可靠的程序保护措施。这些措施包括 Debug 锁定、代码加密和随机码保护。

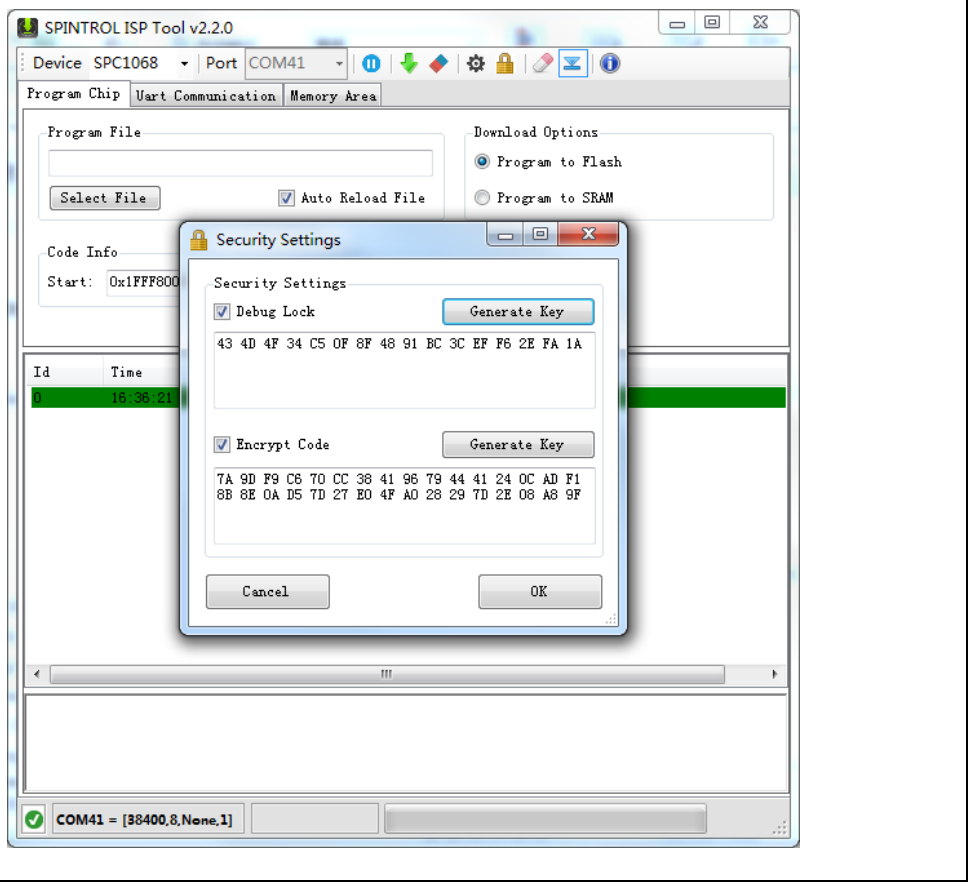
2 Security 功能

2.1 Debug 锁定

Debug 接口在烧写和调试程序方面是十分必要的。这也意味着可以通过 Debug 接口读取或者改写芯片内部存储器的内容。因此，在产品量产时，Debug 功能必须被禁用。用户可以通过 Spintrol 提供的 ISP 工具来实现 Debug 锁定的功能，如图 2-1 所示。用户只需勾选 Debug Lock 选项，那么 ISP 工具在下载程序时会将该信息传递给 Boot Loader。当 SPC1068 再次上电后，Boot Loader 就会锁定 Debug 接口。这样就可以避免通过 Debug 接口去获取芯片内部的程序。

在使能 Debug 锁定功能的同时，还需要设置长度为 16 字节的密码。ISP 工具在下载用户程序时会将该密码传递给 Boot Loader。接着，Boot Loader 会将该密码加密后存储于芯片内部。Debug 接口锁定后，如果用户再次向 SPC1068 中烧写程序，需要先向芯片传递密码，Boot Loader 会接收并验证该密码。如果密码不正确，Boot Loader 会拒绝新程序的下载，同时擦除 Flash 中的内容。这就避免了未授权的下载尝试，防止芯片被破解。

图 2-1. ISP 工具 Security 设置界面



2.2 代码加密

代码加密是将原始的 HEX 数据进行加密，然后再写到 Flash 介质中。这种方法可以避免通过反汇编技术破解代码的尝试。

SPC1068 在设计时考虑到了这种需求，具备对用户程序进行加密的功能。在图 2-1 所示的 ISP 工具中，用户只需要勾选 Encrypt Code 选项，即可使能程序加密的功能。于此同时，用户还需要设定加密引擎需要的 32 字节 Key，用户可以手工输入设定，也可以单击 Generate Key 按钮随机生成。ISP 工具在下载程序时先将 32 字节的 Key 传递给 Boot Loader，然后 Boot Loader 使用这些 Key 将 ISP 传递过来的程序加密后存储于芯片内部的 Flash。Boot Loader 在程序下载完成后，会将 32 字节的 Key 加密后保存在芯片内部。这样即使 SPC1068 内部数据泄露，加密用的 Key 和用户程序都不会有被破解的风险。当芯片再次启动后，Boot Loader 会利用保存在芯片内部的 Key 将 Flash 中的已经加密的程序解密，然后再执行程序。如果解密失败，Boot Loader 会擦除 Flash 中的内容。

2.3 随机码保护

有条件的芯片破解人员也许会具备使用特定设备将芯片打开的能力，然后将 Flash 中的数据读取出来，然后再将这些数据写到未编程的 SPC1068 芯片中，实现对用户产品的复制。对于这种情形，SPC1068 也具备相应的保护机制。

SPC1068 片内包含 OTP Flash，一旦被 Lock 住，其中的内容就无法改写，只能够读取。SPC1068 出厂时 OTP Flash 中会被写入 8 字节的随机码。客户的应用程序在运行时可以从芯片的 OTP Flash 中读取并校验该随机数。如果该随机数与应用程序预设的值不一致，就终止应用程序的运行。由于所有出厂的 SPC1068 芯片的 OTP Flash 都已经锁定，而且每个芯片的随机码也不相同，即使芯片破解人员获得了 Flash 中的程序数据，然后将这些程序数据写到其他 SPC1068 芯片中，程序也不会正常工作。这样，客户的产品就不能够被批量复制。

3 修订记录

表 3-1. 文档修订记录

日期	版本	修改内容
2017-09-14	1	初始版本
2019-07-12	2	1. 修改章节 2.1。