

利用安全存储器实现 FPGA 设计安全解决方案

引言

由于很容易捕获到配置比特流，并进行复制，因此，FPGA 设计很难防范设计窃取。和窃取知识产权 (IP) 相比，几乎不可能从比特流中提取出 IP，但是却能从 FPGA 中克隆整个设计。为了保护配置比特流，有的 FPGA 现在能够对比特流进行加密。然而，对于不具备嵌入式比特流加密手段来加密配置比特流的 FPGA 而言，需要在生产过程中增加步骤对 FPGA 中的密钥进行编程，因此提高了成本。对于大批量应用，使用安全辅助芯片的性价比会更高一些。

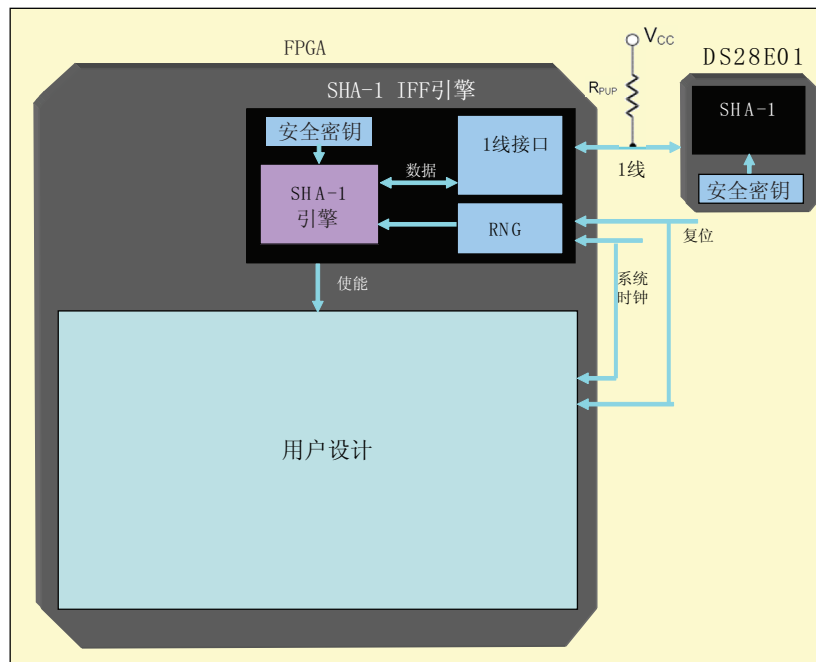
本文档提供解决方案来帮助保护 FPGA 设计不被克隆。利用“识别、朋友或者敌人 (IFF)”设计安全方法，在 FPGA 中和安全存储器中的哈希计算结果匹配之前，这一方案禁用 FPGA 中的设计，因此，即使捕获到了配置数据比特流，设计也是安全的。在这一解决方案中，安全存储器是 FPGA 的安全辅助芯片。

设计实现

在 IFF 概念中，采用了安全辅助器件来计算哈希算法。Dallas 半导体公司的安全存储器 DS28E01 结合了 1024 位 EEPROM 和符合 ISO/IEC 10118-3 安全哈希算法 (SHA-1) 的“挑战-响应”认证安全方法。DS28E01 是 1 线接口器件，因此，这一解决方案只需要一个 FPGA I/O 引脚。安全存储器需要采用上拉电阻和 1 线 I/O 引脚连接（关于 DS28E01 电气规范，请联系 Dallas 半导体公司）。

图 1 所示为采用了 IFF 概念的设计安全参考设计的顶层结构图。安全存储器 SHA-1 引擎根据存储在安全存储器中的密钥来计算哈希算法，该密钥是 FPGA 产生的随机数，在安全存储器中具有唯一的 ID。

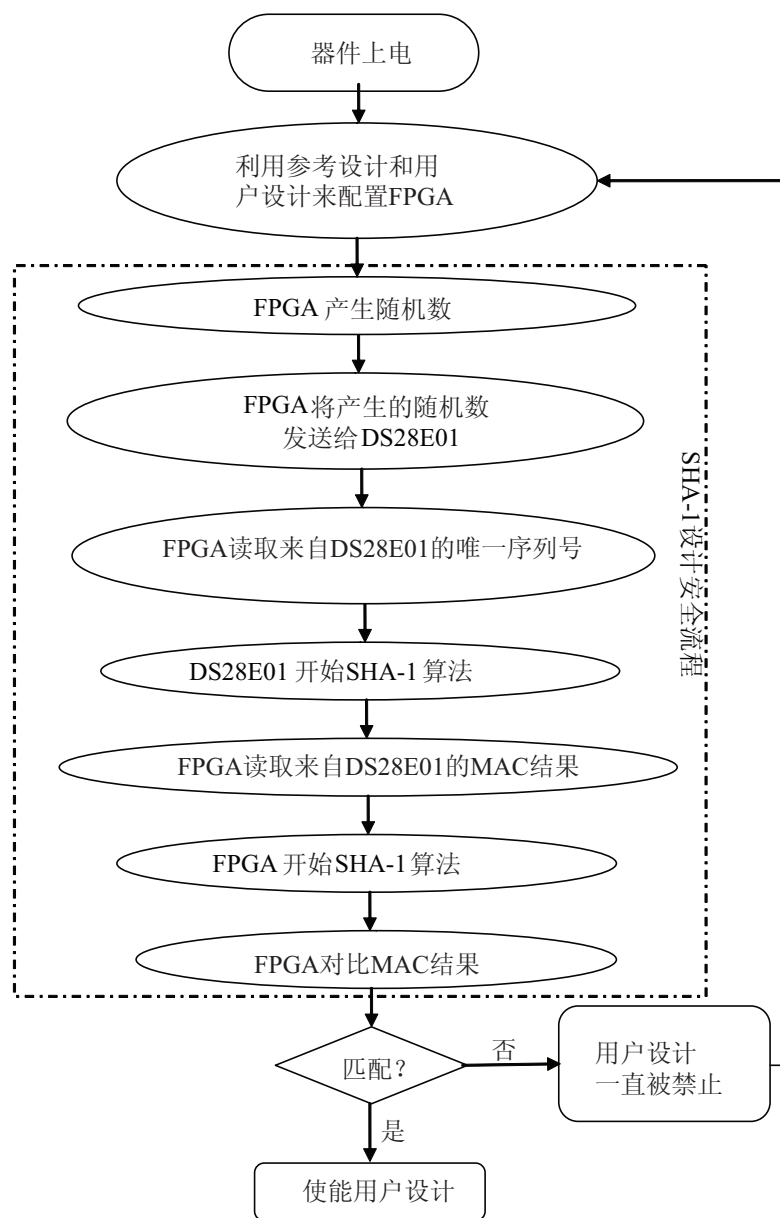
图 1. 采用了 IFF 概念的设计安全解决方案



在使用 FPGA 时，SHA-1 IFF 模块中有安全存储器中的匹配密钥，能够根据和安全存储器中 SHA-1 引擎相同的输入来计算 SHA-1 算法。配置完 FPGA 后，不会启用用户设计。只有当安全存储器和 FPGA 中的哈希计算结果相匹配时，SHA-1 IFF 模块才会使能用户设计。

系统一旦上电，以嵌有 SHA-1 IFF 参考设计的用户设计配置完 FPGA 后，FPGA 产生一个随机数，把它发送给安全存储器。FPGA 读取来自 DS28E01 的 160 位消息认证码 (MAC) 计算结果，将其和 FPGA SHA-1 IFF 引擎 MAC 结果对比。如果 MAC 结果匹配，SHA-1 IFF 模块使能用户设计，如果不匹配，则禁用它。图 2 所示为采用了 IFF 概念的设计安全流程。

图 2. 采用了 IFF 概念的设计安全流程



这一参考设计为用户利用 FPGA 来设置 DS28E01 器件提供了其他方法。FPGA 配置完成后，它支持 FPGA 向安全存储器发送密钥，在生产过程中比较安全的地方进行设置。


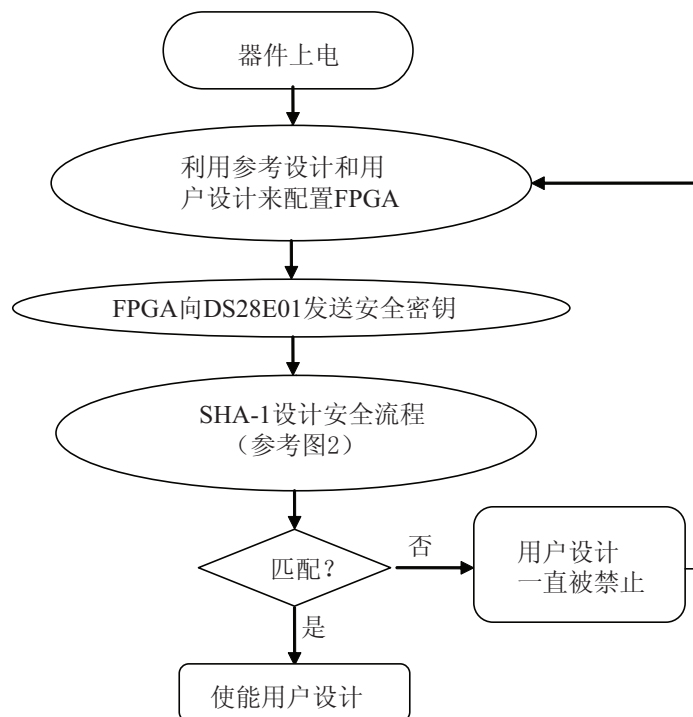
 只在安全存储器第一次编程时采用这一方法。关于在大批量编程中怎样对 DS28E01 器件编程的方法，请联系 Dallas 半导体公司。

图 3 所示为通过 FPGA 对 DS28E01 编程的设计安全流程。

图 3. 通过 FPGA 对 DS28E01 编程的设计安全流程



一旦使能用户设计后，关断 SHA-1 IFF 引擎模块以降低功耗。用户可以连接外部逻辑或者状态机来复位 SHA-1 IFF 引擎，再次启动工作。一旦使能信号变为高电平，SHA-1 IFF 引擎模块不断计算并检查 SHA-1 算法。

设计模块组成

这一解决方案的参考设计含有三个主要模块：

- **SHA-1 引擎**：这一模块计算 SHA-1 算法，进行安全认证。它接收安全存储器通过 1 线接口传送来的设计，将其和 MAC 结果进行对比。只有当哈希计算结果和安全存储器中 SHA-1 引擎的哈希计算结果匹配时，才使能用户设计。
- **随机数发生器 (RNG)**：当复位信号置位 SHA-1 引擎模块时，RNG 为该模块产生一个随机数。SHA-1 IFF 参考设计使用了一个 8 位 RNG 块。SHA-1 引擎模块处理这一 8 位随机数，转换成 40 位随机数据，进行哈希计算。
- **1 线接口**：这一模块支持 FPGA 中参考设计和安全存储器之间的数据传送。

用户设计模块

SHA-1 IFF 引擎系统时钟频率典型的 F_{MAX} 是 100 MHz，或者更低。用户必须向参考设计输入 SHA-1 IFF 引擎频率，从而保证数据在 FPGA 和安全存储器之间正确地发送和接收。用户可以为 SHA-1 IFF 引擎和用户设计提供不同的时钟。

解决方案的安全性

上电时，当 FPGA 中的配置数据比特流在 FPGA 和外部存储器之间传送时，可以捕获到它。利用捕获到的配置数据比特流，配置另一 FPGA 器件，就可以复制这一 FPGA 设计。这一方案可以确保克隆器件无法工作，从而保护了用户设计。没有正确的密钥和哈希算法计算结果，会一直禁用 FPGA 中的用户设计。

为了将设计克隆到另一 FPGA 设计中，必须克隆密钥和安全存储器唯一的 ID。这很难实现，因为不能读出 DS28E01 密钥，也无法从 MAC 结果中反向篡改 SHA-1 算法来确定密钥。

结论

即使捕获了配置数据比特流，这一 FPGA 设计安全 IFF 解决方案也能保护 Altera® FPGA 设计不被克隆。在 FPGA 中和安全存储器中的哈希计算结果匹配之前，一直禁止用户设计。这一设计安全解决方案保护了 FPGA 设计人员的 IP。

详细信息

- Dallas 半导体公司 /Maxim 集成产品：
www.maxim-ic.com



101 Innovation Drive
San Jose, CA 95134
www.altera.com

版权 © 2007 Altera 公司。保留所有版权。Altera、可编程解决方案公司、程式化 Altera 标识、专用器件名称和所有其他专有商标或者服务标记，除非特别声明，均为 Altera 公司在美国和其他国家的商标和服务标记。所有其他产品或者服务名称的所有权属于其各自持有人。Altera 产品受美国和其他国家多种专利、未决应用、模板著作权和版权的保护。Altera 保证当前规范下的半导体产品性能与 Altera 标准质保一致，但是保留对产品和服务在没有事先通知时的升级变更权利。除非与 Altera 公司的书面条款完全一致，否则 Altera 不承担由此处所述信息、产品或者服务导致的责任。Altera 建议客户在决定购买产品或者服务，以及确信任何公开信息之前，阅读 Altera 最新版的器件规范说明。