AN12278

LPC55S00 物联网安全解决方案

版本 1 — 2020 年 5 月 应用笔记

目录

1 简介

安全性应该是所有连接设备制造商的关注点。根据应用和目标市场的不同，政府对安全性有相应的要求。

LPC55S00 是一系列基于 Arm® Cortex®-M33 的 MCU，适用于嵌入式应用。它利用新 ARMv8M 架构，具有丰富外设集。LPC55S6x 引入了更高级别的性能和高级安全功能，包括 TrustZone-M 和多个协处理器扩展。该系列增加了 LPC55S2x 和 LPC55S1x。LPC55Sxx 系列为最终产品在其整个生命周期内免受意外威胁提供保护。本应用笔记提供了每个 LPC55Sxx MCU 安全系统的差异和进步。

LPC55Sxx 采用安全设计，并由驱动安全片上系统 (SoC) 的安全软件提供支持。本应用笔记描述了 LPC55Sxx MCU 的 SoC 内置安全功能，因为它们与最先进的安全功能如何在设备的生命周期内实现安全目标有关，讨论了 LPC55Sxx 的安全子系统和 SoC 特定细节，并提供了安全启动的概述。可以在 LPC55Sxx 用户手册和其他应用笔记中更深入地了解这些功能，在本文档中尽可能引用。有关大多数参考资料，请参阅 www.nxp.com 上的 LPC5500 文档页面。本应用笔记的资源部分列出了一组丰富的支持信息，并提供了术语表来定义和解释本应用笔记中使用的众多首字母缩略词。

1.1 安全愿景

在设计安全系统时，从安全模型开始很重要。 安全模型由策略、威胁形势和下述方法构建而成。 应用安全模型为理解和设计设备的安全目标提供了一个框架。集成到嵌入式控制器中的技术使这些方法或安全策略如何执行以实现产品安全目标成为可能。下面介绍了一个基本的安全模型，并提供了策略、威胁格局和方法策略的示例。确定应保护的数据的现有规则（例如，固件、密钥、用户和应用程序数据密码、个人信息和网络凭据的管理）。

威胁模型是终端设备面临并提供防护的攻击和攻击者的定义。 它考虑了对设备的访问和攻击的成本。例如，专业攻击者使用现成的工具来获取访问权限并插入恶意软件。这些方法是执行设备策略的手段。这涉及应用安全技术来实现产品目标（例如，禁用调试访问以限制处理器上机密数据的可用性）。

1.2 物联网安全的运用

所有物联网设备都需要保护。如果这些设备被攻击者滥用，这些我们称之为“事物”的节点可能会导致互联网服务的大规模中断或给事物的用户带来伤害。

作为设计师，您的任务是设计一个安全的事物并通过威胁分析找到所有漏洞。 攻击者只需要找到一个。你怎么能成功？ 如果没有成功，从攻击中恢复的计划是什么？ 你有没有办法安全地更新你的设备的固件来对抗攻击？ 这不仅必须是可信的东西，而且还必须与整个应用程序网络（称为云或后端）保持可信连接。

随着越来越多的设备被部署，我们希望信任水平随着物联网端点的增长而扩展。安全不能成为一个可选项目，它必须从设计概念的一开始就进行架构，无论您正在设计什么。验证一切是物联网安全环境设计的口头禅。您的事物必须与网关、计算和存储以及应用程序和分析层一起进行身份验证。在本应用说明中，我们将重点放在事物本身的安全性以及它与后端层通信时的安全性。我们讨论了物联网设备的高级安全目标以及设备整个生命周期所需的安全保护。

1.3 使用 LPC55Sxx 的物联网设备的生命周期

交付到物联网设备世界的事物遵循标准生命周期。在下一小节中，我们确定了生命周期阶段和创建安全可信设备的建议。 每个生命阶段的任务提供了需求的上下文。在每个阶段，任务都集中在 LPC55Sxx MCU 附加值上。有关如何使用安全功能的详细信息，请参阅 MCU 用户指南。

图 1. 生命周期

- 采购

  建立值得信赖的供应链，这样您就可以确保所有硬件和软件组件的设计完整性。为此，您需要使用 MCU 和包含可信软件和硬件的工具，例如 LPC55Sxx。安全子系统允许隔离安全软件 IP。物理不可克隆功能（PUF）提供信任根（ROT）密钥管理和唯一身份。

- 开发

  在一般开发过程中，所有调试端口都启用。ROM 允许在部件配置和安全启动和程序完成后进行调试访问。设备未被提供密钥，因此在开发过程中只需要使用图像的 CRC。

  在安全开发期间，测试 SRAM PUF 生成的 ROT 密钥。软件可以在 Flash 中加密，安全可信环境可以得到全面测试。 客户可以使用公钥来建立和验证通信。通过基于 ROM 的调试身份验证保护产品免受非法调试。

- 生产

  可以在此周期内完成密钥生成和配置。由于每次 MCU 上电都会生成私钥和公钥，因此无需在制造阶段创建密钥。 此时，唯一身份被记录下来，并允许在设备部署（或目标使用）期间与后端安全通信。

- 部署

  定义调试配置的灵活性允许模块制造商实施分层保护方法。该部件可由 1 级客户（安全代码开发人员）出售给仅开发非安全代码的 2 级客户。在这种情况下，第 1 层客户发布部件以始终允许非安全调试，从而允许第 2 层客户进一步密封部件。

  调试端口根据应用程序配置关闭。 调试可以在调试身份验证后启用、永久禁用或为非安全调试接口启用。

  如果启用安全启动，只允许签名的镜像。

  客户密钥随着固件更新公钥和 Prince 密钥而更新，同时维护 RoT 密钥。

  可以使用引导加载程序或安全固件擦除和重新编程安全固件。

  应用程序代码固件通过暴露 API 的机制进行编程。单独的安全区域（独立的 Prince 密钥）可用于存储固件。

- 使用

  物联网设备被添加到物联网基础设施并由后端进行身份验证。数据完整性在内部进行检查，然后安全地传送到后端系统。

- 维护

  可以使用安全的镜像空中升级（OTA）更新受信任事物的固件。API 将验证接收到的图像并允许和安装固件更新。 如果固件更新被撤销，ROM 会提供从 SPI NOR 闪存启动作为备份。

- 客户退回（FA 模式）

  密钥和固件被破坏。设计人员使用调试验证机制来设置客户程序区中的 FA_MODE 字段，或者客户将拆焊的部件运送到 NXP。所有密钥都被擦除，PUF 索引被阻止，调试端口被打开。

- 停用

  当将事物从服务中移除时，事物的唯一身份可以从可信事物列表中移除，并且固件可以被更新或擦除以使事物失效。

1.4 上层安全目标

今天的物联网设备必须采用先进的安全技术设计，以防止未经授权的访问、恶意控制和数据盗窃。设备相互连接和云连接需要可扩展且易于使用的安全解决方案。密钥用于建立安全连接并保护物联网设备传输的数据。物联网系统开发人员必须确定所需的保护级别，以确保产品符合适用的安全标准和法规，以及产品用户期望和公司风险管理政策，以防止本地和远程攻击。

对于当今的嵌入式设计，在事物的生命周期中需要考虑六个基本安全目标。LPC55Sxx 系列 MCU 提供安全硬件功能，结合软件，可实现上述每个目标。

以下是上层安全目标的系统级要求。

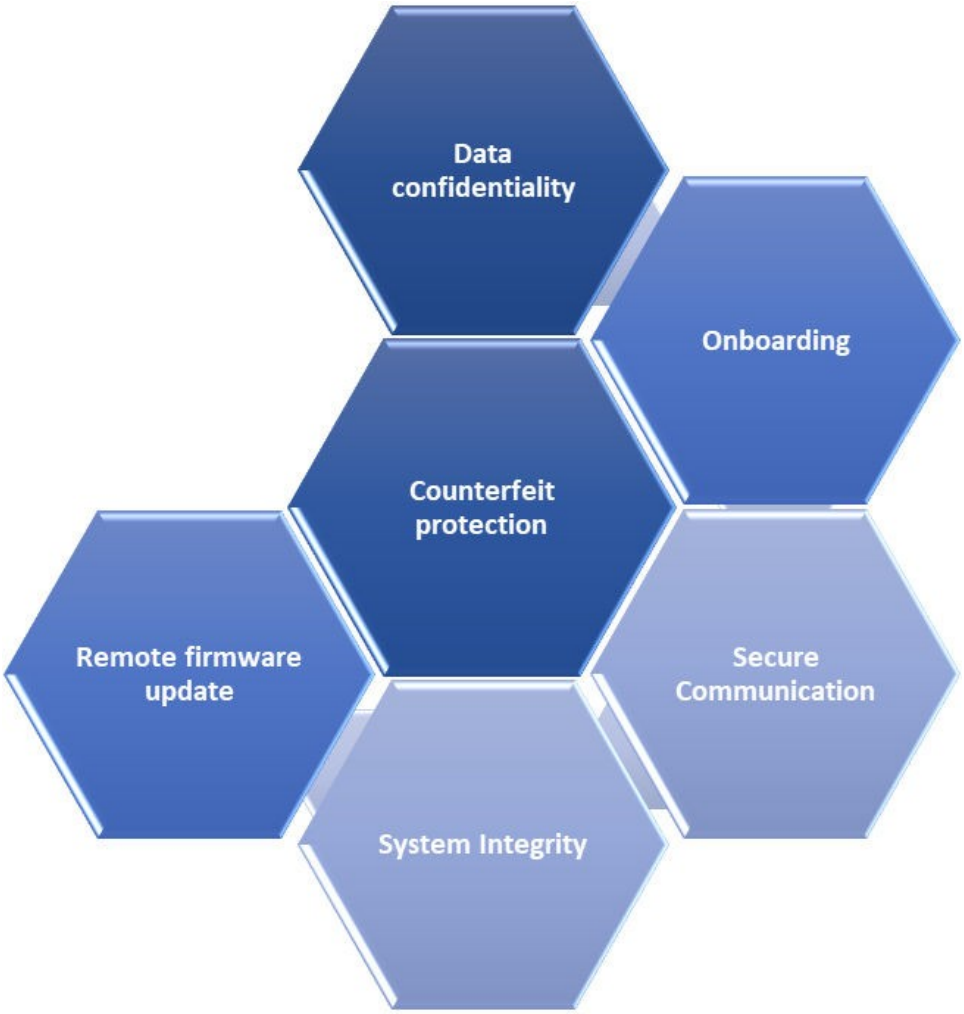

图 2. 上层安全目标

- 防伪

  这是关于确保设备是真实的。它首先建立一个独特的身份并支持攻击者难以复制的身份验证。如果进行物理（或逻辑）检查，则 RoT 密钥不可访问且无法读取。

- 部署上线

  这是关于您如何将 IoT 设备接入您的网络并将其投入使用。设备必须强制执行特权级别，将非安全固件与安全固件隔离开来。您的设备必须使用硬件保护私钥，同时保护终端设备和后端系统之间的共享凭据。

- 系统完整性

  每次开机时，设备都必须通过安全启动来保护自己。MCU 必须强制信任 MCU 提供的功能。MCU 上的固件必须加密敏感的软件功能以防止逆向工程。设备必须支持回退镜像以维持最低功能。

- 安全通信

  这是关于使用非对称加密来建立会话密钥，然后切换到对称加密进行批量通信以保护连接和静态数据。设备应维护在闪存中加密的审计日志。

- 确保数据机密性和完整性

  这是关于控制数据流并保护静态数据的机密性和完整性，并依赖于安全通信。这是关于在 SoC 内安全地收集和存储数据。

- 发现漏洞时更新固件/软件

  这是关于让您的物联网设备完成它的设计任务并确保关键服务的可用性。您的设备应监控和报告未经授权的访问，限制敏感功能和服务（无线网络连接）并安全地支持固件更新。在产品生命周期结束时，必须将其停用以删除所有敏感数据并防止黑客将其带到另一个网络。

2 LPC55Sxx 产品概述

2.1 LPC55Sxx MCU 平台和通用模块

LPC55S69 具有一个带有 TrustZone、MPU、FPU 和 SIMD 的 150-MHz Cortex-M33 内核和另一个 150-MHz Cortex-M33。LPC55S66 仅实现了一个 150-MHz 内核。核心 0 上有两个协处理器，一个称为 PowerQuad 的 DSP 加速器，以及一个称为 CASPER 的加密引擎。内核平台有一个多层总线矩阵，允许从两个内核同时执行以及其他主设备对外围设备和存储器的并行访问。片上存储器包括高达 640 KB 的闪存、高达 320 KB 的 RAM 和 128 KB 的 ROM。

定时器包括 5 个 32 位定时器、SCTimer/PWM、多速率定时器、窗口看门狗定时器、实时时钟 (RTC) 和微定时器。每个内核都有自己的 systick 计时器。LPC55S6x 安全子系统包括 TrustZone、HW SRAM PUF、调试认证、实时 PRINCE 解密、TRNG、安全启动、DICE、SHA-2、AES-256 和 PFR。

LPC55S2x 是 LPC55S69 内核和外设的子集，具有高达 512 KB 的闪存和 256 KB 的 RAM。LPC55S2x 安全子系统包括 HW SRAM PUF、调试认证、实时 PRINCE 解密、TRNG、安全启动、DICE、SHA-2、AES-256 和 PFR。当部件被锁定时，部件中的所有代码都是安全的。LPC55S1x 是一款成本降低的 MCU，具有一个 150-MHz Cortex-M33 内核，带有 TrustZone、MPU、FPU，协处理器接口上带有 CASPER 加密引擎。它还具有 256 KB 闪存、96 KB RAM 和其他外设，例如 CAN-FD。 LPC55S1x 安全子系统包括 TrustZone、HW SRAM PUF、调试认证、实时 PRINCE 解密、TRNG、安全启动、DICE、SHA-2、AES-256、代码看门狗和 PFR。

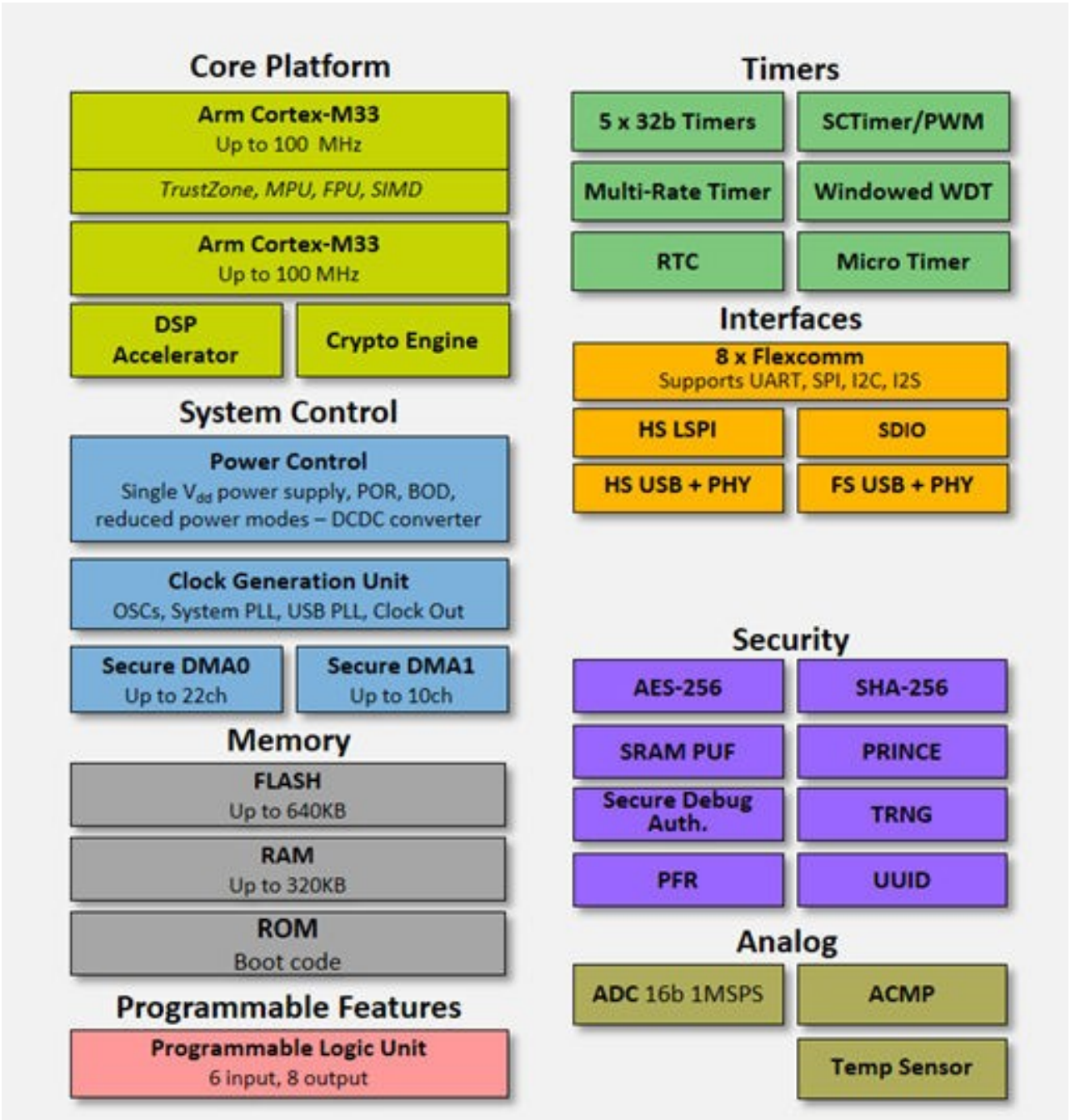

图 3. LPC55S6x MCU 平台和通用模块

通讯接口包括一个带片上 HS PHY 的高速 USB，一个可以不需外接晶振运行的全速 USB，同时支持 WIFI 和 SD 卡的 SDIO 接口，1 个高达 50 MHz 的高速 SPI 时钟速率和 8 个 Flexcomms，支持多达 8 个 SPI、I2C、UART 或 I2S。模拟系统包括一个以 1 MSPS 采样的 16 通道 16 位 ADC、一个模拟比较器和一个温度传感器。其他模块包括一个可编程逻辑单元、一个降压 DC-DC 转换器，在 -40 至 105 °C 的温度范围内工作电压为 1.8V 至 3.6 V。

2.2 LPC55Sxx 安全模块

2.2.1 LPC55Sxx 安全

LPC55S6x 和 LPC55S1x 是具有增强安全功能的 SoC，并且在带有 TrustZone 的 SoC 中具有隔离性。LPC55S6x 和 LPC55S1x 具有称为可信执行环境（TEE）的安全和非安全资源域控制器。通过正确的软件配置，TEE 可以在 SoC 内创建安全硬件隔离。还包括一个支持安全启动、身份验证签名和加密的 ROM，以及一个需要身份验证才能解锁调试功能的安全调试接口。

2.2.2 LPC55Sxx 平台安全架构

恩智浦设计的 LPC55Sxx 具有全面的安全基础，支持基于 Arm TrustZone M 的可信执行，结合启动和运行时硬件架构功能。恩智浦提供支持软件隔离的硬件，包括安全属性单元（SAU）、安全总线、DMA 和安全 GPIO。LPC55S6x 和 LPC55S1x 安全将 TrustZone 与安全硬件相结合。整个系统中的软件、CPU、互连、内存和外围设备都具有安全性。有可信赖的外围设备，可在整个设备生命周期内增强安全性。

这种硬件隔离创建了一个可信、安全的执行环境，将安全和非安全资源的访问分开，并创建了一个对存储在内存中的数据访问受限的执行环境。

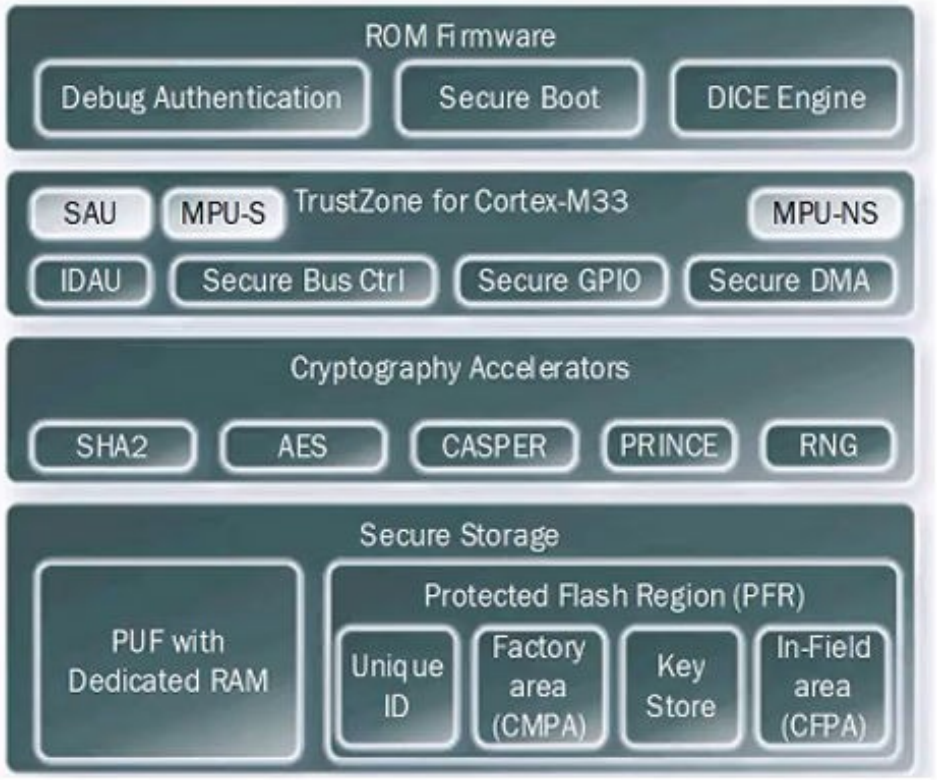

图 4. LPC55S6x 和 LPC55S1x 安全子系统

2.2.3 LPC55Sxx 安全功能

图 4 详细说明了对 LPC55Sxx 上的安全服务有贡献的所有模块。在底部，是与芯片特定秘密（UUID、PUF 和 PFR）的存储和生成相关的不同硬件功能。该芯片提供了使用恩智浦编程的唯一 ID 和尖端 PUF 技术生成密钥的选项。ROM 使用此信息，连同 AES 的加密加速器、哈希散列，将安全服务应用于芯片的安全启动。ROM 支持从内部闪存启动加载，并通过 I/O 串行接口上的 ISP 机制支持工厂车间编程或作为备份从外部 SPI 闪存引导加载。

在调整安全目标和生命周期之前，先简要介绍 LPC55Sxx 产品的 LPC55Sxx SoC 硬件安全功能。在 表 1 的列标题中，6x(0A)是指第一版硅片，6x(1B) 是指第二版（或量产）硅片。

表 1. LPC55Sxx 系列安全汇总表

			LPC55S			
			6x(0A)	6x(1B)	2x	1x
ROM 支持的功能	安全启动	签名镜像	x	x	x	x
		不可更改的启动固件	x	x	x	x
		PRINCE 加密	x	x	x	x
		预配置 Trustzone	x	x	x	x
	DICE Engine	CDI = HMAC(UDS, RTF)		x	x	x
		RTF = image || NMPA || CMPA	x	x	x	x
		RTF = image || NMPA || CMPA || EPOCH		x	x	x
		RTF = SHA256(image) || NMPA || CMPA||EPOCH				x
	SYSCON 寄存器 PSA 启动状态	BOOT_SEED				x
		HMAC = HMAC (BOOT_SEED, SHA-256(image) || NMPA || CMPA || EPOCH)				x
	调试认证	支持恩智浦调试认证协议版本 1.0（RSA-2048）和 1.1（RSA-4096）	x	x	x	x
	恢复启动	外部 SPI 闪存恢复启动镜像		x	x	x
		FLASH_REMAP（双重镜像）				x
TrustZone for Cortex-M33	Arm 的安全实现	安全归因单元（SAU）	x	x		x
		内存保护单元（MPU）安全	x	x		x
		内存保护单元（MPU）非安全	x	x	x	x
	NXP 安全子系统实现	特定属性单元（使用 IDAU 接口）	x	x		x
		安全总线控制器	x	x		x
		安全 GPIO 控制器	x	x		x
		安全 DMA 控制器	x	x		x
		内存保护检查器（MPC）	x	x		x

下页继续...

表 1. LPC55Sxx 系列安全汇总表（续上页）

			LPC55S			
			6x(0A)	6x(1B)	2x	1x
		外设保护检查器（PPC）	x	x		x
安全子系统	密码学加速器	HashCrypt 引擎：对称 AES-256 和哈希 SHA2	x	x	x	x
		AES-ICB（CTR 形式）抗 SCA	x	x	x	
		AES-ECB，CBC，CTR				x
		SHA 摘要上下文				x
	PRINCE	即时 Flash 加密/解密引擎	x	x	x	x
		三个区域（次区域使能）	x	x	x	x
	CASPER	带有用于 RSA 和 ECC 的 RAM 的加密加速器和信令处理引擎	x	x	x	x
	随机数生成器（RNG）	RNG				x
		TRNG —— 尚未认证，已完成一些检查，将通过大多数测试套件 DieHard、NIST SP800-22、FIPS 140-1	x	x	x	x
		RNG 健康状态，熵积累，初始熵				x
	Code Watchdog	检测意外指令序列的执行				x
安全存储	物理不可克隆功能（PUF）	用于密钥生成和身份识别的 SRAM PUF	x	x	x	x
		设备唯一根密钥（256 位强度）	x	x	x	x
		可以存储 64 位到 4096 位的密钥	x	x	x	x
	受保护的 flash 区域（PFR）	符合 RFC4122 的 128 位 UUID	x	x	x	x
		PUF 密钥存储	x	x	x	x
		激活码	x	x	x	x
		Prince 区域密钥码	x	x	x	x
		固件更新加密密钥	x	x	x	x
		唯一设备机密（UDS）	x	x	x	x

下页继续...

表 1. LPC55Sxx 系列安全汇总表（续上页）

			LPC55S			
			6x(0A)	6x(1B)	2x	1x
	客户制造可编程区域（CMPA）	启动设置	x	x	x	x
		RoT 密钥表哈希	x	x	x	x
		调试配置	x	x	x	x
		PRINCE 配置	x	x	x	x
		Flash 重映射				x
		客户数据（224B）	x	x	x	x
	客户现场可编程区域（CFPA）	单调计数器	x	x	x	x
		Prince IV 代码	x	x	x	x
		客户数据（224B）	x	x	x	x

有关如何使用这些模块的完整说明，请参阅 LPC5500 系列 Cortex-M33 MCU 的相关文档。

图 4 详细说明了对 LPC55Sxx 上的安全服务有贡献的所有块。该图分为四个子组，然后分为组成这些子组的硬件块。下面是对这些块的描述。

2.3 ROM 固件

内部 ROM 存储器存储不可更改的引导代码，用于执行凭证处理、解密、身份验证、安全程序流和安全状态执行。任何复位后，CM33 从 ROM 开始其代码执行。每次部件上电或每次复位时都会执行引导加载程序代码。ROM 允许各种引导选项和 API。根据客户制造可编程区（CMPA）、系统编程（ISP）引脚和镜像头类型定义的值，引导加载程序决定是从内部闪存引导还是进入 ISP 模式。

2.3.1 安全认证启动的保护

LPC55Sxx 允许启动私钥签名图像。安全启动 ROM 支持三种类型的安全保护模式，为使用 MCU 的终端产品提供可扩展的安全和制造选项。ROM 支持从加密的 PRINCE 区域启动。加密镜像可以对知识产权进行强有力的保护。ROM 支持公钥和镜像撤销——即一种不允许应用新更新的方法，除非它们是特定版本。这是回滚保护的基础。ROM 支持使用安全镜像预先配置 TrustZone-M 设置，最大限度地减少在 TrustZone 配置期间中断用户固件流程的攻击的可能性。如果所有其他安全闪存镜像均未通过身份验证，则 ROM 支持从 SPI 闪存安全启动。在 ISP 模式下，外部 SPI 闪存的内容被复制到 RAM、验证并执行复制的代码（如果受保护）。

2.3.2 LPC55Sxx ROM 固件功能

ROM 支持安全启动、CRC 镜像完整性检查和调试身份验证。安全启动会在启动过程中检查代码凭据的完整性。 系统完整性得以保持，因为每次启动都会检查真实性。 对于固件更新，基于 ROM 的身份验证支持 SPI 闪存恢复和回滚保护。

2.3.2.1 LPC55Sxx 安全启动

安全启动可防止执行未经授权的代码。MCU ROM 总是在每次上电和复位时执行。ROM 检查驻留在内部闪存中的第一个用户可执行映像，以确定该代码的真实性。如果代码是真实的，则控制权转移给它。这建立了从 ROM 到用户引导代码的可信代码链。可以使用 Boot ROM API 进一步扩展此链。

此 SoC 中用于验证启动代码真实性的方法是验证代码上的 RSA 签名。代码使用 RSA 私钥签名。用于签名验证的相应 RSA 公钥包含在签名映像中包含的 X.509 证书中。

Boot ROM 提供了一个选项，用于通过镜像头配置上述设置，并在控制传递给应用程序代码之前将其锁定。由于 ROM 是不可变代码，高度安全的应用程序可以使用此选项。

ROM 提供了一种从外部 1 位 SPI 闪存设备执行恢复启动的方法，以防止更新使 MCU 变砖并使应用程序无法使用。您可以通过多种方式使用此功能。

1. 恢复媒体模型，其中外部 SPI 闪存用于存储 SB2.1 格式的工厂映像。当主闪存上的映像损坏时，ROM 将尝试通过使用外部闪存上的镜像启动设备来恢复设备。
2. 暂存媒体模型。在 OTA 期间，可以将 SB2.1 文件格式的固件更新包下载到外部 SPI 闪存中。下载并验证映像后，正在运行的固件可以使用 ROM API 调用 SPI 恢复启动。 重新启动时，ROM 将执行新加载的 SB2.1 文件以更新主闪存。

如果禁用安全启动，则恢复启动功能会从 SPI 闪存启动纯文本映像，将镜像复制到 SRAM，然后跳转执行代码。如果启用了安全启动，则 SB2.1 映像将从外部 SPI 闪存加载到内部闪存中，验证公钥的散列。如果身份验证通过，则 ROM 跳转以执行代码。

有关用于所有安全引导操作的步骤的详细说明，请参阅 *LPC55Sxx 安全引导*（文档 AN12283）。

2.3.2.2 LPC55Sxx 调试认证

LPC55Sxx 提供调试身份验证协议作为验证调试器并授予其访问设备权限的工具。LPC55Sxx 上的调试身份验证方案是基于证书的质询-响应（challenge-response）方案，可确保拥有所需调试凭据的调试器只能通过调试接口成功进行身份验证并访问设备的受限部分。该协议为设备及其调试接口提供了一种机制来验证调试器（或用户）的身份和凭据。在调试验证过程成功完成之前，调试器无法访问设备的安全部分。下面的列表是调试身份验证流程的通用示例。

- 供应商生成 RoT 密钥对并在发货前使用 RoT 公钥哈希的 SHA256 哈希对设备进行编程。
- 现场技术人员生成自己的密钥对并将公钥提供给供应商进行授权。
- 供应商认证现场技术人员的公钥。在调试凭证证书中，供应商分配访问权限。
- 遇到被锁定产品问题的最终客户将其交给现场技术人员。
- 现场技术人员使用他的凭据对设备进行身份验证并解锁产品以进行调试。

2.3.2.3 设备标识符组合引擎（DICE）架构

LPC55Sxx MCU boot ROM 是一个可信平台模块，其中包含支持设备标识符组合引擎 (DICE) 硬件要求的固件。DICE 包含的规范由 Trusted Computing Group 提供。请参阅 Trusted Platform Architecture Hardware Requirements for a Device Identifier Composition Engine。该引擎创建一个身份标识，该值源自唯一设备机密 (UDS) 和运行时指纹 (RTF)，这是芯片的一种压缩密码学表示。此派生值是复合设备标识符 (CDI)。您可以使用复合设备标识符来证明启动代码的可信度。

2.4 ARMv8M Cortex-M33 的安全架构和 TrustZone

TrustZone 是 Cortex M33 中可用的一项技术。TrustZone 提供了实现分离和访问控制的方法，以隔离受信任的软件和资源，从而减少关键组件的受攻击面。可信固件可以保护可信操作，是存储和运行关键安全服务的理想选择。该代码保护受信任的硬件以增强和强化受信任的软件。这包括用于加密加速器、随机数生成器和安全存储的硬件辅助模块。最佳实践要求此代码是小的、经过良好审查的代码，并提供安全服务。

LPC55S66x 和 LPC55S1x 已将内核 0 实现为 Cortex-M33，并启用了完整的 TEE 和 TrustZone 支持。LPC55S69 具有第二个 Cortex-M33（核心 1），它不支持使用 TZ 实现安全环境。LPC55S2x MCU 的用户没有可用的 TZ，必须考虑整个 MCU 是安全的还是非安全的。

隔离只是基础。安全是关于分层保护，添加更多硬件和软件以创建更多层。

2.4.1 安全和非安全内存归属

内存可以是安全（S）、非安全（NS）或非安全可调用（NSC）。 这些由“安全归因单元”（SAU）和“实现定义归因单元”（IDAU）定义。安全数据只能通过 S 代码读取。安全代码只能由处于 S 模式的 CPU 执行。NS 数据可以被安全状态和非安全状态 CPU 访问。NS 代码不能被 S 代码执行。NSC 是一个特殊区域，允许 NS 代码跳转并执行安全网关（SG）操作码。这是 NS 代码调用 S 函数的唯一方式。如果在 NSC 区域执行 SG，CPU 处于 NS 状态，则 CPU 转移到 S 状态。到 S 函数的跳转是在安全模式下完成的。CPU 状态可以是安全特权、安全非特权、特权（处理程序）或非特权（线程）。

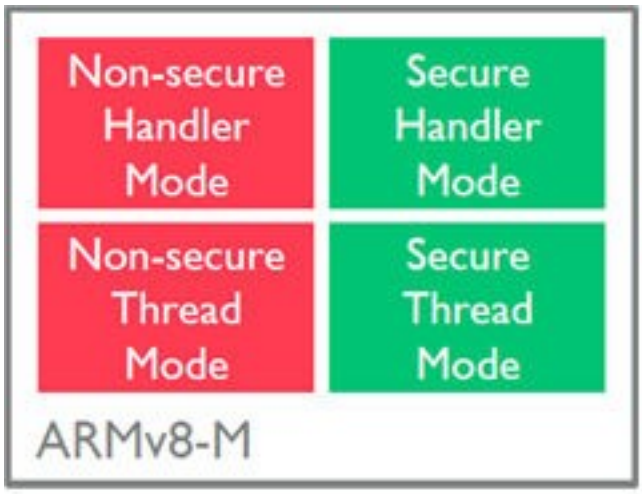

图 5. 内存特性

2.4.2 ARMv8M 附加 CPU 状态

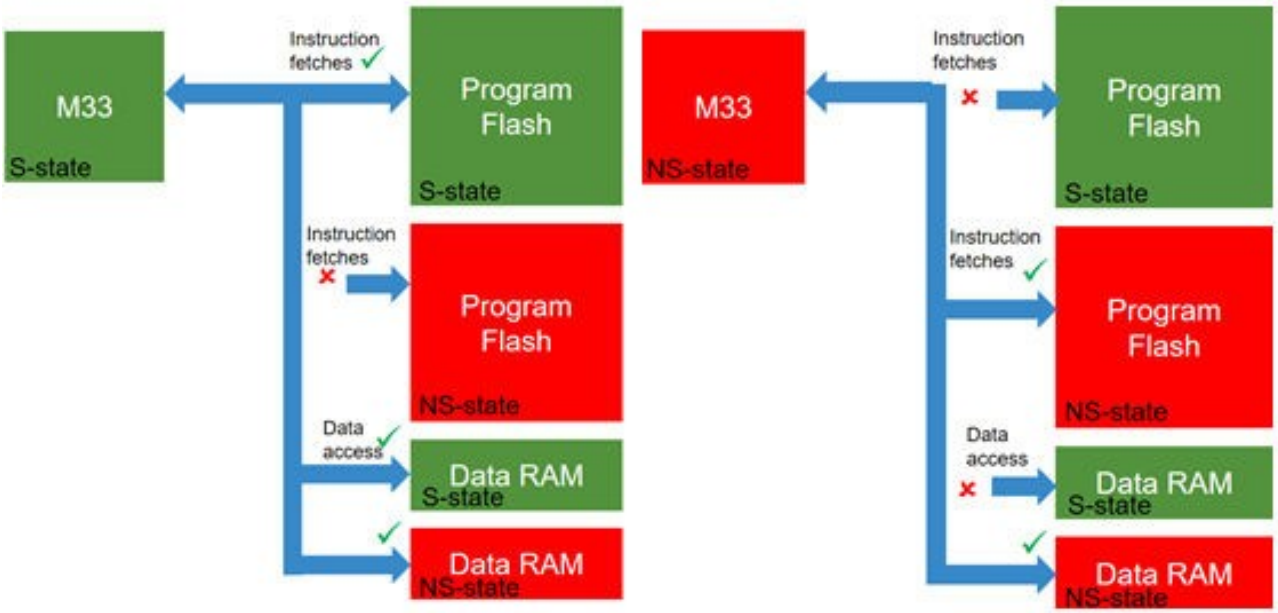

图 6. 安全和非安全代码

安全和非安全代码在单个 CPU 上运行，以实现高效的嵌入式实现。

处于非安全状态的 CPU 只能从非安全程序存储器执行。处于非安全状态的 CPU 只能访问 NS 内存中的数据。

对于安全、可信的代码，有一个新的安全堆栈指针和堆栈限制检查。S 和 NS 区域有单独的内存保护单元（MPU），每个状态都有专用的 SysTick 计时器。安全端可以配置中断的目标域。

2.4.3 安全 MPU、NVIC、SYSTICK 和安全堆栈指针

为了支持安全状态 Cortex-M33 架构，core0 扩展到包括安全 MPU、安全 NVIC、安全 SYSTICK 和带有堆栈限制检查器的安全堆栈指针。每个 CPU 都有一组自己的这些模块，以更好地隔离安全和非安全的环境。Core1 没有可信执行环境，因此被视为微控制器中的非安全资源。

2.4.4 安全总线控制器

LPC55Sxx 使用安全总线控制器模块矩阵来管理 MCU 中的数据流。外设保护检查器（PPC）、内存保护检查器（MPC）、主设备安全包装器（MSW）以及安全锁定和错误日志、安全中断屏蔽、管理程序中断和 GPIO 屏蔽的组合。

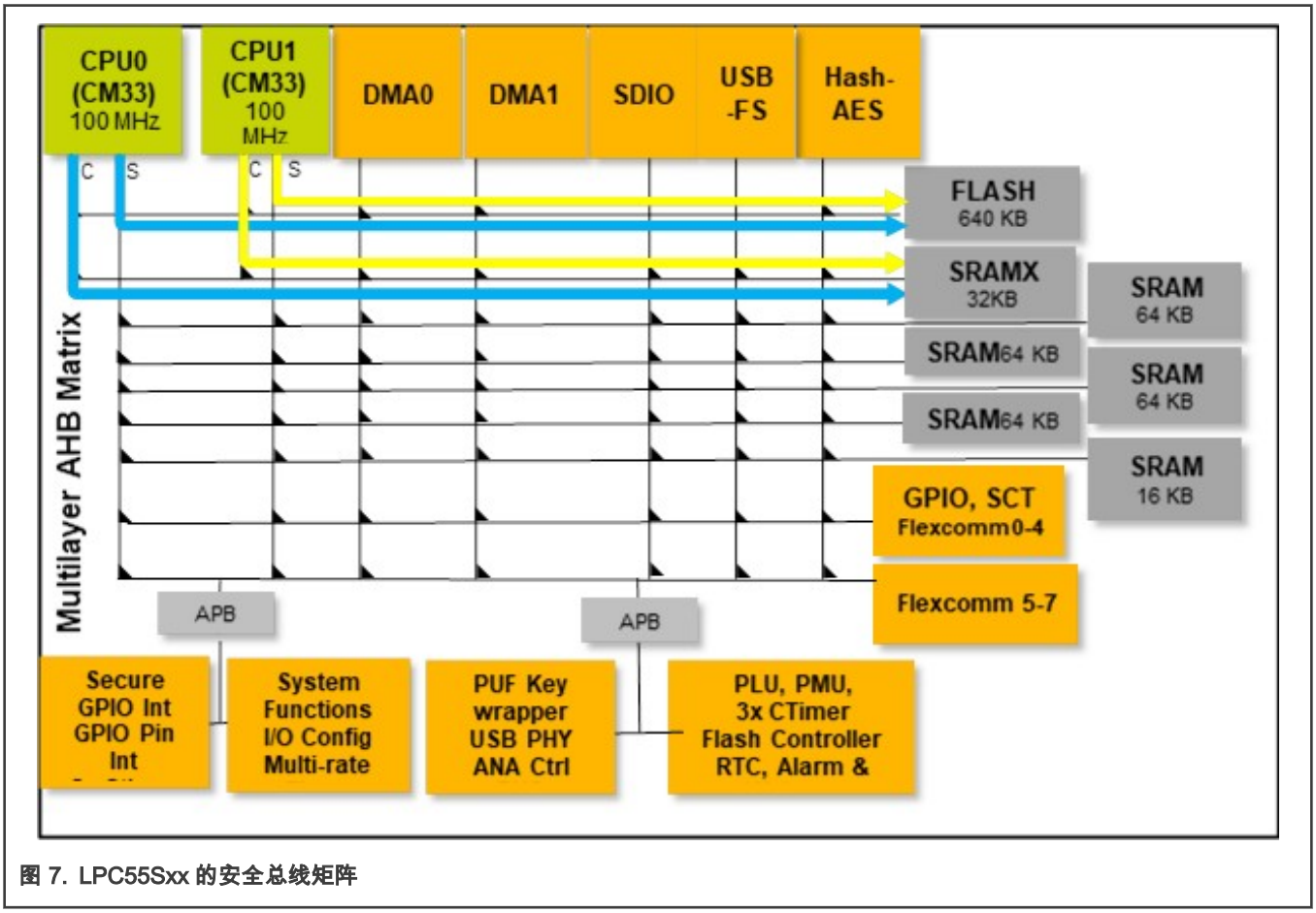

图 7. LPC55Sxx 的安全总线矩阵

M33 内核或 DMA 引擎等总线主控之间的总线矩阵由主设备安全包装器（MSW）包裹，并具有用于篡改检测的安全边带信号。每个总线从端口都有一个 PPC。MPC 用于存储器和总线桥。

2.4.5 地址定义的安全性

LPC55Sxx 的独特之处在于针对 core0 的 ARM TrustZone 的恩智浦 IDAU（实现特定设备属性单元）实施涉及使用地址位 28 将地址空间划分为潜在的安全和非安全区域。地址位 28 未在内存访问硬件中解码，因此每个物理位置出现在两个位置。其他硬件确定允许对任何地址进行哪些类型的访问（包括非安全可调用（NSC））。

表 2. TrustZone 和系统通用映射

开始地址	结束地址	TrustZone, CORE0	CPU 总线	CM-33 用法（两个 CPU）
0x0000 0000	0x1FFF FFFF	Non-Secure	代码	Flash Memory, Boot ROM, SRAM X.
0x2000 0000	0x2FFF FFFF	Non-Secure	数据	SRAM A, SRAM B, SRAM C, SRAM D, SRAM E.
0x3000 0000	0x3FFF FFFF	Secure	数据	同上
0x4000 0000	0x4FFF FFFF	Non-Secure	数据	AHB and APB 外设

下页继续...

表 2. TrustZone 和系统通用映射（续上页）

开始地址	结束地址	TrustZone, CORE0	CPU 总线	CM-33 用法（两个 CPU）
0x5000 0000	0x5FFF FFFF	Secure	数据	同上

2.4.6 归属单元

用于 ARMv8-M 实现的 TrustZone 由安全归因单元（SAU）和实现定义的归因单元（IDAU）组成。设备属性单元（DAU）通过 IDAU 接口连接到 CPU0，如 图 8 所示。SAU 和 IDAU 的组合将特定的安全属性（S、NS 或 NSC）分配给来自 CPU0 的特定地址。CPU0 的访问取决于其安全状态以及由 IDAU 和 SAU 设置的最终安全属性，然后由安全 AHB 控制器与特定检查器进行比较，该检查器标记了内存和外设的各种访问策略，如 图 9 所示。所有地址要么是安全的或不安全。ARMV8M 内部的安全归因单元（SAU）与 MPU 协同工作。LPC55S6x 和 LPC55S1x 支持 8 个 SAU 区域。

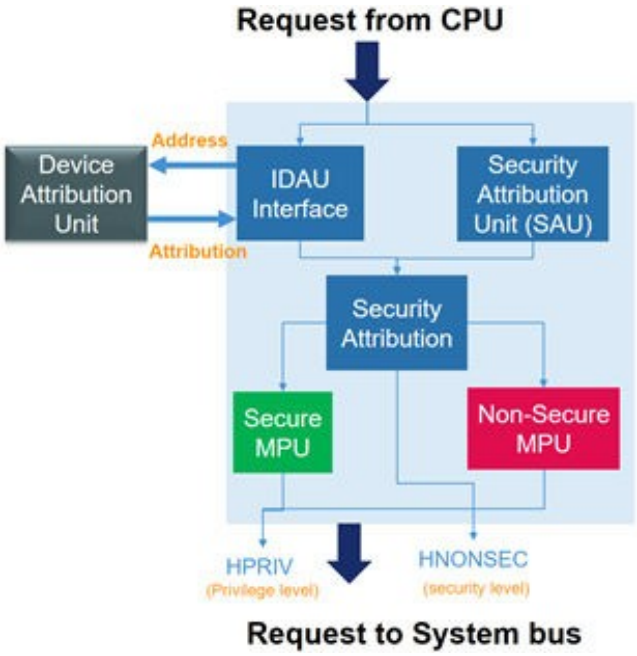

图 8. 安全归属单元

恩智浦集成了一个特定于实现的设备属性单元（IDAU），允许安全操作系统与应用程序完全分离。

2.4.7 LPC55Sxx IDAU 和安全归属单元操作

IDAU 是一个简单的设计，使用地址位 28 来允许在两个位置的存储器混叠。如果地址位 28 = 0，则内存是非安全的。如果地址位 28 = 1，则内存是安全的。

SAU 允许 8 个内存区域，并允许用户覆盖 IDAU 的固定映射，以定义非安全区域。默认情况下，所有内存都设置为安全。

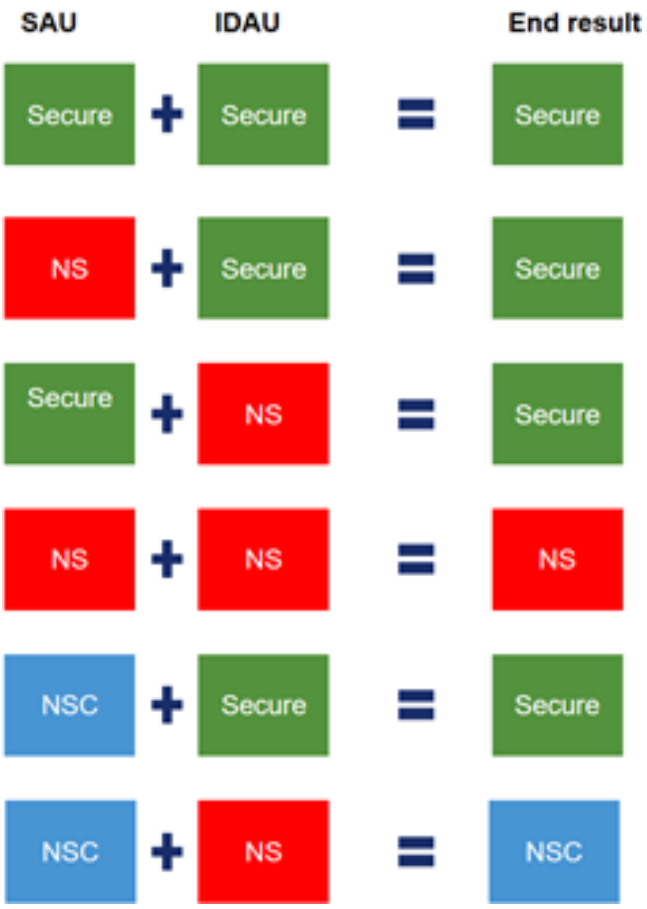

图 9. 安全归属逻辑

2.4.8 内存保护检查器（MPC）

MPC 与所有存储设备、片上闪存和 SRAM 一起使用。所有规则都在安全控制寄存器组中设置。用户必须具有最高级别的“安全特权”才能设置规则。如果处于默认状态，则忽略权限级别。默认情况下，只检查安全级别。

2.4.9 主设备安全包装（MSW）

MSW 封装了三种类型的总线主控：具有安全扩展的 TrustZone 的 Cortex M33，DMA、SDHC、USB-FS 和 Hash-Crypt 等简单主控以及 micro-CM33 和 EZH 等智能主控。

2.4.10 安全锁定

安全总线控制器允许锁定以下配置：

- 所有 PPC 和 MPC 检查器设置
- 所有主设备安全级别（MSW）设置
- SAU 设置
- 安全的 MPU 设置
- 安全向量偏移地址（S_VTOR）
- 非安全 MPU 设置
- 非安全向量偏移地址（NS_VTOR）

- Core1-CM33 MPU 设置
- Core1-CM33 矢量偏移地址（VTOR）

Boot ROM 为用户提供了一个选项，可以通过镜像头指定上述设置，并在控制传递给应用程序代码之前将其锁定。由于 ROM 是不可变代码，高度安全的应用程序可以使用此选项。

2.4.11 安全错误日志

当内存保护检查器（MPC）检测到安全违规时，如果数据或指令访问发生违规，则会引发安全违规中断。异常细节可以记录在安全 AHB 控制器模块和安全控制寄存器中的可读标志。可以记录以下内容：

- 每个 AHB 从端口/层的安全违规地址。
- 状态位指示发生了哪个 AHB 从端口/层违规。
- 主设备安全级别、权限级别和访问类型（代码/数据）。

Core0 将切换到安全模式来处理异常。

2.4.12 管理程序（Hypervisor）中断

LPC 硬件实现了一项功能，借此对安全 AHB 控制器模块进行非安全访问将引发中断。对安全 AHB 控制器模块的访问仅限于安全特权级别。AHB 控制器将引发安全违规中断，该中断可配置为安全，从而允许非安全代码引发安全中断。 这可用作对管理程序的调用。

ARMv8-M 监管者调用是备份的，因此可以存在于安全模式中，并且存在一个单独的监管者处理程序用于非安全。使用 SVC（Supervisor Call 操作码）不允许非安全代码调用管理程序（Hypervisor），因为管理程序（Hypervisor）的安全属性是安全特权的，因此非安全代码不能进入它。

2.5 LPC55Sxx 上的密码学加速器

为了应对非对称密码学的挑战，恩智浦开发了 CASPER，这是一种具有 RAM 共享功能的密码加速器和信号处理引擎。 它是某些非对称加密算法的硬件加速器引擎，例如椭圆曲线加密（ECC）。CASPER 位于 Cortex-M33 协处理器总线上。

其他加密加速器包括 Hash-Crypt 引擎（AES 和 SHA）、PRINCE 实时闪存加密/解密引擎和 TRNG。

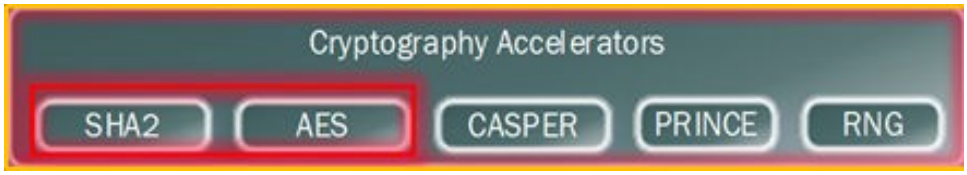

图 10. AES 和 SHA 密码学加速器

2.5.1 SHA & AES 引擎

SHA 模块可以支持 SHA1 和 SHA2-256 算法。AES 模块支持使用 AES-ECB、AES-CBC 和 AES-CTR 模式的对称加密。这些模块可以与来自总线的直接写入或 DMA 写入一起使用。

2.5.1.1 AES 引擎

LPC55Sxx 器件提供片上硬件 AES 加密和解密引擎，用于数据加密或解密、数据完整性和来源证明。AES 引擎可以使用 PUF 中的密钥或软件提供的密钥对数据进行加密或解密。AES 引擎在电子码本（ECB）模式、密码块链（CBC）模式或计数器（CTR）模式下支持 128 位、192 位或 256 位密钥。AES 引擎在 ICB（索引代码簿）模式下支持 128 位密钥，可提供额外的对侧信道攻击的保护。

2.5.1.2 ICB-AES

而作为分组密码的 AES 是一个电子密码本，有一种特殊的混合密码模式可用，称为 ICB。ICB 是 CTR 密码模式的一种形式，但其目的是抗侧信道分析。它使用了一种方法，该方法是针对各种侧信道攻击的方法，例如 SPA/DPA/DPX（功率分析）和发射分析。由于具有 SCA 抗性，ICB 比正常的 AES ECB 和 CTR 模式慢。此模式可用于敏感信息的超安全片上存储。

2.5.1.3 SHA

LPC55Sxx 器件提供片上哈希支持，以执行 SHA-1 和 SHA-2 使用 256 位摘要（SHA-256）。哈希散列是一种将任意大的消息或代码镜像减少到称为摘要的相对较小的固定大小“唯一”数字的方法。SHA-1 哈希产生一个 160 位的摘要（五个字（word）），SHA-256 哈希产生一个 256 位的摘要（八个字（word））。

哈希散列用于四个主要目的：

- 数字签名模型的核心，包括证书，例如用于安全更新。
- 支持质询/响应或在与基于散列的消息身份验证代码（HMAC）一起使用时验证消息。
- 在安全启动模型中，验证代码完整性。
- 验证外部存储器没有受到损害。

2.5.2 TRNG

TRNG 模块是一个硬件加速器模块，可生成 256 位熵。该模块的目的是生成高质量、密码学安全的随机数据。随机数生成器用于数据屏蔽、加密、建模和模拟应用程序，这些应用程序使用必须以随机方式生成的密钥。LPC55S6x 嵌入了一个硬件 IP（结合适当的软件和随机模型的编写），可用于生成高质量的真随机数。

LPC55S6x 嵌入了一个可以使用的硬件 IP（结合适当的软件和随机模型的编写） 生成高质量的真随机数（FIPS140-2、AIS31、P2/PTG.3）。

使用不太复杂的软件，RNG 功能的目标应降低到 AIS31、P2/PTG.2 甚至 AIS31、P1/PTG.1。 不同之处在于熵源的激活和监控方式。

2.5.3 基于哈希的消息认证码（HMAC）

HMAC 可以在 LPC55Sxx 上使用预共享密钥和使用 SHA256 引擎的散列实现。它是一种验证消息（加密与否）的方式，也可用于质询或响应。双方必须有一个预先共享的密钥，它只是一个共享的秘密值，而不是一个加密密钥。

HMAC 比签名快得多，但仅适用于预共享密钥，不得泄露或丢失（与公钥不同）。可以使用 Diffie-Hellman 等信任模型动态共享 HMAC 密钥，也可以使用由两个设备共享的板载唯一密钥。

2.5.4 PRINCE 实时加解密

LPC55Sxx 器件支持使用 PRINCE 加密算法对片上闪存进行实时加密和解密。与 AES 相比，PRINCE 速度更快，因为它可以在不增加额外延迟的情况下进行解密和加密。PRINCE 在读取或写入数据时运行，无需先将数据存储在 RAM 中，然后再加密或解密到另一个空间。它对 64 位的块大小和 128 位的密钥进行操作。

此功能对于资产保护非常有用，例如保护应用程序代码、保护存储的密钥以及启用安全闪存更新。

2.5.5 CASPER - 加密加速器

CASPER Crypto 协处理器用于为某些非对称加密算法（例如椭圆曲线加密（ECC））所需的各种功能启用硬件加速。

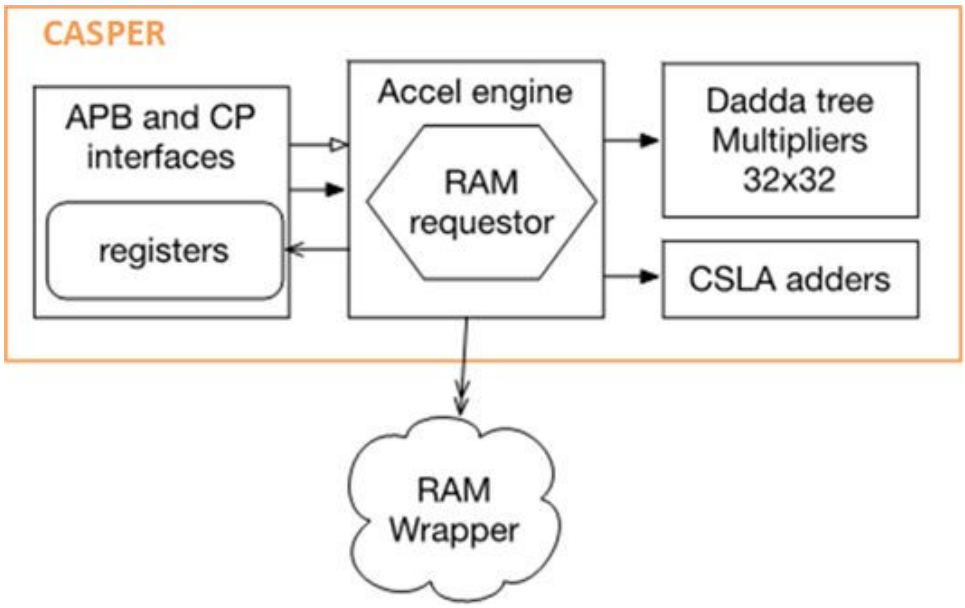

图 11. CASPER

- 到 ARMv8-M Cortex-M33 的 Co-proc 接口
  - 64b 数据总线
  - 减少 CPU 负载：一次写入启动加速器和启动序列的可能性
- 快速共享 RAM 访问
  - 2×32b RAM 分配给 CASPER RAM 接口，允许 64 位并行访问
  - RAM 与系统内存共享
- 一组加法器和寄存器以允许 MAC（乘法和累加）
- 两个 32×32 乘法器
- 侧信道保护
  - 使用随机掩码

NXP 开发了一个软件加密库，可将标准大数运算库函数映射到 CASPER 硬件功能。

2.6 LPC55Sxx 安全存储 PUF、PFR 和 UUID

LPC55Sxx 安全存储功能可用于实现防伪安全目标。与芯片特定秘密的存储和生成相关的硬件功能包含在以下模块中：

- 通用唯一标识符（UUID）
- 物理不可克隆功能（PUF）
- 客户制造商可编程区（CMPA）

该芯片提供了使用恩智浦编程的唯一 ID、PFR 或 PUF 来生成密钥的选项。ROM 使用此信息以及加密加速器将安全服务应用于芯片的安全启动、调试和 PRINCE 配置。

2.6.1 受保护的 Flash 区域

受保护的闪存区域（PFR）由客户可编程区域和工厂编程区域组成。客户制造可编程区（CMPA）用于存储引导配置、RoT 密钥表哈希、调试身份验证配置和 PRINCE 配置。PRF 区域可以使用 ROM API 使用 BLHost 命令进行编程。BLHOST 命令由主机设备（如 PC 或主机 MCU）执行，以在 ISP 模式下与 ROM API 交互。有关命令的详细信息，请参阅用户手册中的 Boot Rom 章节。图 12 是 CMPA 中存储的状态和控制位的摘录。

客户现场可编程区（CFPA）包含版本控制计数器、RoT (RKTH) 密钥撤销和 PRINCE 区域 IV 代码。CFPA Ping 和 Pong 页面用于选择版本号最高的 CFPA 页面。有 17 个可能的撤销 ID。恩智浦制造程序区（NMPA）是用户可以找到 UUID（通用唯一标识 ID）的地方，可用于唯一标识设备。

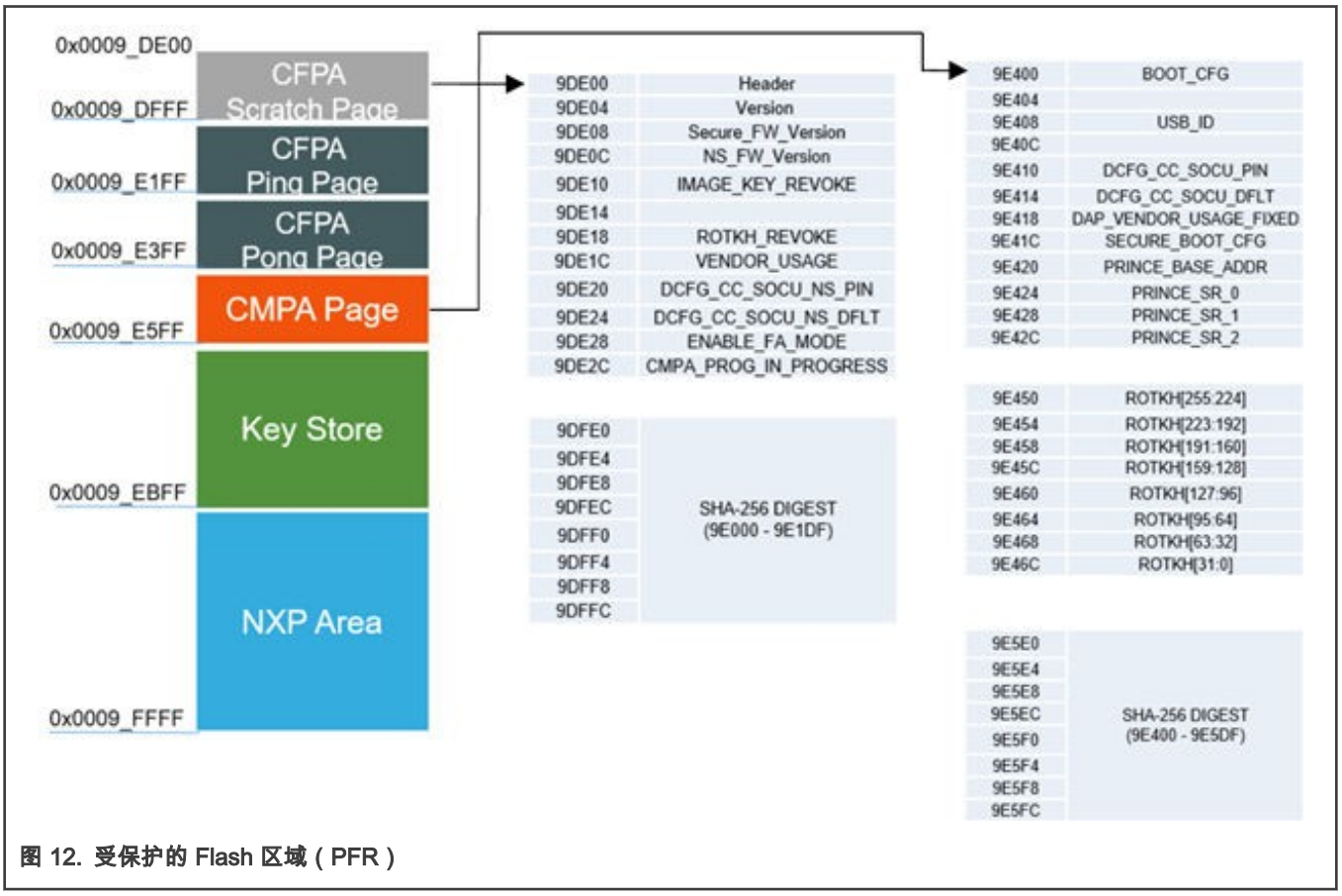

图 12. 受保护的 Flash 区域（PFR）

2.6.2 UUID 唯一标识

LPC55Sxx 存储 128 位 IETF RFC4122 兼容的非顺序密码学分布式 ID，每个设备具有 128 位 UUID。LPC55Sxx 的硬件为唯一身份提供支持，以防止可能将不受信任的数据注入网络的假冒设备或克隆的可能性。

2.6.3 PUF 特性和用法

该系列微控制器包括一个基于 SRAM 的物理不可克隆功能（PUF）控制器，可安全生成唯一的设备指纹和设备唯一的加密密钥。SRAM PUF 机制紧密集成到 LPC55S00 系列中，使来自 PUF 的密钥能够直接被设备的内部高级加密标准（AES）-256 加密引擎使用。与其他密钥注入或存储方式相比，独特且不可克隆的密钥提供了显着的安全优势。PUF 密钥将安全性的基础建立在设备唯一的不可克隆密钥上，从而减少了“一次攻击，到处重复”（ break once repeat everywhere ）攻击的威胁。

SRAM PUF 硬件构建了一个根密钥，用作密钥加密密钥（KEK）以保护其他用户密钥。PUF 使用源自未初始化 SRAM 的设备数字指纹和称为激活码（AC）的纠错数据生成设备唯一的 256 位 KEK。AC 是在“注册”过程中生成的 1192 字节长的数据块，该过程发生在设备的配置/个性化过程中。每次执行 PUF Enroll 时，都会生成不同的 AC，从而产生不同的数字指纹。AC 必须存储在 PFR 中，以允许后续启动以重现 PUF 密钥。然后可以将加密的密钥（密钥代码）存储在 MCU 中。

密钥可以是对称密钥或非对称公钥/私钥对。例如，应用软件可以使用密钥生成接口来创建椭圆曲线公钥对以供 TLS 协议使用。这使得可以用于远程验证设备的每个设备密钥的强大存储成为可能。

系统可能具有需要配置到设备中的其他秘密。这些秘密可以使用 PUF 生成的密钥来本地加密和存储此信息。这种“包装”的信息可以安全地存储在内部闪存中，因为它是加密的。

LPC55Sxx PUF 硬件可将密钥路由到 AES 和 PRINCE 加密引擎，以支持设备启动期间内部闪存的解密和身份验证。 在每次启动时，ROM 使用密钥来验证和解密固件。

索引为 0 的 PUF 密钥永远不会离开设备。索引零密钥仅通过内部路径路由到 AES 和 PRINCE 模块。

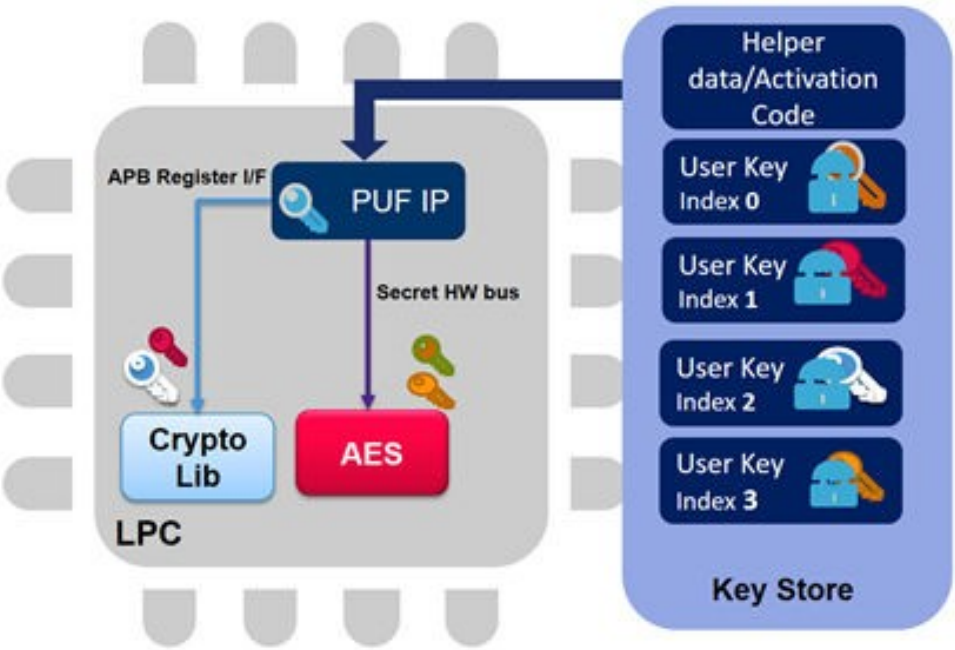

图 13. PUF 密钥储存

2.6.4 PUF Key 的提供

通过使用 BLHOST 命令，可以对产品进行个性化。*LPC55Sxx Usage of the PUF and Hash Crypt to AES Coding*（文档 AN12324）描述了如何使用根密钥安全地生成、存储和检索用户密钥，并提供了配置过程的分部指南。

2.7 安全外围设备

2.7.1 GPIO 屏蔽

在 LPC55Sxx 上，所有数字引脚状态均可通过 GPIO 控制器寄存器读取。信息可能会通过连接到配置为安全外设的外设的引脚泄漏。例如，连接到安全元件的 I2C 引脚。对此类泄漏敏感的安全软件应使用安全 AHB 控制器模块中的 SEC_GPIO_MASK0/1/2/3 寄存器屏蔽相应的引脚。SDK 中的示例代码演示了如何使用安全 AHB 控制器来保护 GPIO 引脚。

2.7.2 安全中断屏蔽

LPC55S69 有两个 CPU。 Core0 有 TrustZone，core1 没有 TrustZone。 在 LPC55S69 上，两个 CPU 都可以访问所有中断。为了保障安全应用，LPC55S6x 上实现了中断屏蔽功能。通过对安全 AHB 控制器模块中的屏蔽寄存器进行编程，可以屏蔽对 core1 的任何中断。Core0 具有内部可编程性，可将任何中断配置为安全中断，使其仅对 NVIC_S 可见，并对 NVIC_NS 屏蔽。

2.7.3 安全外设

还有另外两个被认为是安全的外设，即安全 DMA 和安全 GPIO。有两个 DMA 控制器，一个可以配置为安全的，另一个可以配置为非安全的。建议使用 10 通道的 DMA 控制器作为安全 DMA。所有端口 0 的 32 个引脚都具有安全 GPIO 作为可选引脚复用功能。功能与标准 GPIO 控制器相同。

3 安全模块的用途

LPC55Sxx 系列 MCU 具有抵御本地和远程攻击类型的安全功能。在设计 IoT 设备期间，您将面临关于使用什么功能来实现目标的问题。下表提供了安全目标的摘要，以及使用 LPC55Sxx MCU 上的哪个模块、用于典型安全算法的模块以及每个模块旨在保护的内容。

3.1 将安全目标与安全块对应

表 3 将安全目标与 LPC55Sxx 上可用的安全模块一一对应。

表 3. 用于解决安全目标的 LPC55S00 安全模块

安全目标	LPC55Sxx 上的安全模块
防伪	Unique ID Chipunique Root Key (PUF) PRINCE– Encrypted flash
为设备提供安全信任配置	制造期间–使用加密闪存映像和生成 PUF key
安全通信	使用硬件加速器和通信的安全可信软件建立和维护认证通信的协议。硬件包括： • CASPER - 通过 AES 的非对称加密加速器 PUF 公钥 • AES 256 Engine • SHA Hashing Engine (SHA1 & SHA2) TRNG
数据保密	保持经过身份验证的通信从和传输加密数据到物联网设备。 PUF (key store), AES-256 Engine, CASPER
固件升级	提供 OTA 更新的软件 API，通过安全二进制 (SB) 文件处理由 ROM API 完成安全启动。PRINCE 加密和解密以保护安全固件。
系统完整性	安全启动、PRINCE、针对 ARMV8M 的 TrustZone 运行时保护、Pole/anti-pole 检查
部署上线	使用 PUF 密钥、客户配置区域中的 RoT 公钥哈希灵活配置以实现信任。

3.2 将安全算法与加速器块相对应

表 4 将安全算法与可用的加速器块相对应。

表 4. 具有算法和可用 MCU 加速器的密码学功能

功能	算法	加速器
对称密码学	AES (ECB, CBC, CTR)	HashCrypt 128, 192 & 256 bit keys
对称密码学	PRINCE (CTR)	PRINCE
非对称密码学	RSA, ECC	CASPER
Hash	SHA1, SHA2-256	HashCrypt
MAC	HMAC, CMAC	HashCrypt + SW
签名	HASH (SHA256) + Asymmetric crypto (RSA, ECDSA)	HashCrypt + CASPER

必须开发软件来驱动硬件。LPC55Sxx SDK 可帮助您开始保护您的产品。在 表 4 中，我们将不同的安全模块与软件使用这些模块的方式一一对应。请参阅适用于 LPC55Sxx 的 MCUXpresso SDK，了解驱动程序、演示、RTOS 以及安全模块的软件/硬件实现示例。

3.3 LPC55Sxx 模块保护什么

表 5. 这些模块用于保护什么

LPC55Sxx 上的安全模块	保护域	示例
SHA 引擎（SHA1, SHA2）， AES 256 引擎 TRNG，CASPER	通信交互	密码学、签名验证
SRAM PUF，AES-256 引擎，CASPER	数据	机密，密钥，个人信息
PRINCE（加密的 flash）	固件	知识产权盗窃、逆向工程
UUID（RFC4122），芯片唯一性的根密钥（PUF）	运营完整性	维持服务和收入
Pole/anti-pole 检查，SRAM PUF，安全启动	Anti-tamper	物理攻击，密钥不是存储的而是生成的。
Secure boot ROM，PFR（对于 RoTKH）	ROT	Secure boot ROM, PFR（对于 RoTKH）

4 LPC55Sxx 功能满足的安全目标

在本节中，我们深入探讨如何使用 LPC55Sxx 的安全功能来满足每个已确定目标的需求。在每个小节中，我们讨论在生命周期中很重要的六个安全目标，然后描述可用于支持该目标的 LPC55Sxx 安全功能。详细信息在第一次引用安全功能的位置。在随后的安全目标子部分中，可能会提及前面部分中涵盖的功能。他们是：

1. 防伪保护。
2. 部署上线。
3. 系统完整性。
4. 安全通信。
5. 确保数据的机密性和完整性。
6. 发现漏洞时更新固件/软件。

4.1 防伪保护

LPC55Sxx 安全存储功能可用于实现防伪保护的目标。SoC 具有唯一 ID、芯片唯一根密钥、安全启动和实时闪存加密/解密。

在产品生命周期的所有阶段，从制造到部署到使用、维护到停用，都需要这种保护。

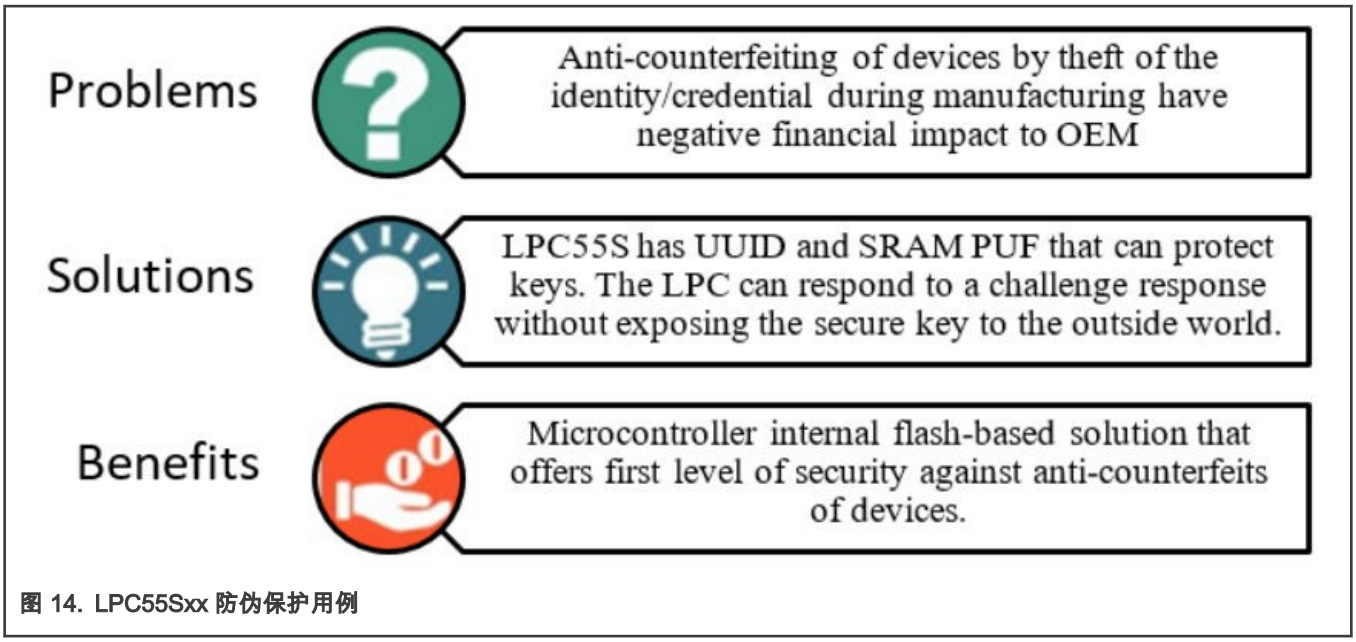

图 14. LPC55Sxx 防伪保护用例

表 6. LPC55Sxx 防伪保护所需的操作

#	操作	执行者
1	基于 PUF 创建根公私钥对	• OEM. • 出厂后上电时
2	不存储私钥，且私钥不离开设备	—
3	公钥被签名并作为证书存储	• OEM. • 出厂后上电时
4	响应外部源的质询（Challenge）。	任何可以与 LPC 云供应商和/或 OEM 的通信 IC/设备。

4.1.1 建立攻击者难以复制的唯一身份和认证

RoT 的基础是 LPC55Sxx SRAM PUF。基于不可克隆的硬件硅指纹，每个芯片都有自己独特的身份。

4.2 部署上线

在目标环境中使用 Thing 有一个正常的过程，这意味着将新设备带入受信任的环境。例如，如果您需要在办公室安装 1000 个灯泡，则需要将新灯泡安装到照明控制系统中。由 LPC55Sxx 控制的灯泡将具有唯一的身份以及加密的私钥和公钥。您可以委托灯泡仅在安全环境中工作。

开发完成后，产品中需要这种保护，从制造到部署到使用到维护再到停用。

4.2.1 数字签名

数字签名将公钥/私钥（例如 RSA 或 ECC）与 SHA 散列相结合。签名模型的优点是公钥和签名都可以公开。因此，签名可以与镜像一起存储在非安全位置，然后使用公钥的副本进行验证。

4.2.2 强制执行权限级别

LPC55Sxx 允许使用 PUF 密钥进行灵活配置。它在客户配置区域具有基于 PUF 的 PKI、RoT 公钥哈希以实现信任。

4.2.3 保护终端设备和后端系统之间的共享凭证

LPC55Sxx 的启动安全性和非对称加密保护终端设备和后端之间的共享凭证，以便将设备身份扩展到与设备相关联的基于后端的服务。

4.2.4 用硬件保护对称密钥和私钥

PUF 密钥是生成的，且不可读取，因为它们没有存储在保险丝（fuse）或非易失性存储空间中。

通过调试接口的 LPC55Sxx 访问可以设置为验证调试或完全锁定调试。在运行存储在片上闪存中的嵌入式代码之前，ROM 启动可以安全地检查客户配置空间和镜像完整性。

4.3 系统完整性

在产品生命周期的每个阶段，从采购到开发、制造、部署、使用、维护到停用，都需要这种保护。

4.3.1 保护物理和逻辑免受入侵

LPC55Sxx 包括使用不同寄存器中的两个互补位用于所有安全选择的基本防篡改。SRAM PUF 在上电时生成密钥。当设备断电或可视性检查时，这些密钥不存在。PUF 与 AES 和 PRINCE 引擎之间有一条安全总线用于凭证注入机制。

4.3.2 使用 MCU 提供的功能增强信任

LPC55S6x 和 LPC55S1x TrustZone 和安全硬件可确保系统的完整性、对其操作的信任，并在设备的整个生命周期内得到维护。

4.3.3 加密敏感软件功能，防止逆向工程

闪存中的图像可以使用 PRINCE 进行加密。在执行过程中，代码会被即时解密。

4.4 安全通信

在产品生命周期的各个阶段，从部署到使用到维护到停用，都需要这种保护。Thing 将用非对称密码算法建立通信，然后一旦建立了安全通信，就使用更快的对称加密交换数据或更新。

4.4.1 维护加密的审计日志

客户 IP 应记录所有通信尝试，无论成功与否。

4.5 数据保密性和完整性

在产品生命周期的各个阶段，从部署到使用到维护到停用，都需要这种保护。这个安全目标是通过实现安全通信目标来实现的。这样，由安全策略标识的数据在由设备处理时保持机密。此外，可以保持对安全外围设备的本地控制。

4.6 安全远程固件更新

每当需要更改设备功能时，无论是解决安全问题还是增强产品功能，都必须进行安全的远程固件更新。安全启动是此过程的第一步。

LPC55Sxx 允许启动已签名的镜像。安全启动 ROM 支持三种类型的安全保护模式，为使用 LPC55Sxx 的终端产品提供可扩展的安全和制造选项。ROM 支持从加密的 Prince 区域启动。加密镜像可以保护知识产权。ROM 支持公钥和镜像撤销——即不允许应用新更新的方法，除非它们是特定版本。这是回滚保护的基础。ROM 支持 TrustZone-M 设置的预配置。此外，如前所述，DICE 以及冗余镜像支持（后备镜像）由 ROM 启用。

LPC55S6xx 安全启动：

- 使用 SHA256 摘要的 RSASSA-PKCS1-v1_5 签名作为加密签名验证。
- 支持 RSA-2048 位公钥（2048 位模数，32 位指数）。
- 支持 RSA-4096 位公钥（4096 位模数，32 位指数）。
- 使用 x509 证书格式来验证镜像公钥。

- 支持多达四个可撤销的信任根 (RoT) 或证书颁发机构密钥，通过在受保护的闪存区域 (PFR) 中存储四个 RoT 公钥哈希的 SHA-256 哈希摘要来建立信任根。
- 支持使用镜像密钥撤销的防回滚功能，并使用 x509 证书中的序列号字段支持多达 16 个镜像密钥证书撤销。
- 如果闪存镜像不安全，支持镜像回退选项。ISP 模式下的镜像恢复使用外部 SPI NOR 闪存来引导 MCU。

LPC55Sxx 提供两个阶段的闪存编程。第一阶段是开发和部署。在此阶段，JTAG 或 SWD 端口用于闪存擦除和编程。但是，一旦部署了设备，就应该禁用这种闪存编程模式。此后，程序闪存的唯一替代方案是通过安全启动 ROM 无线传输。

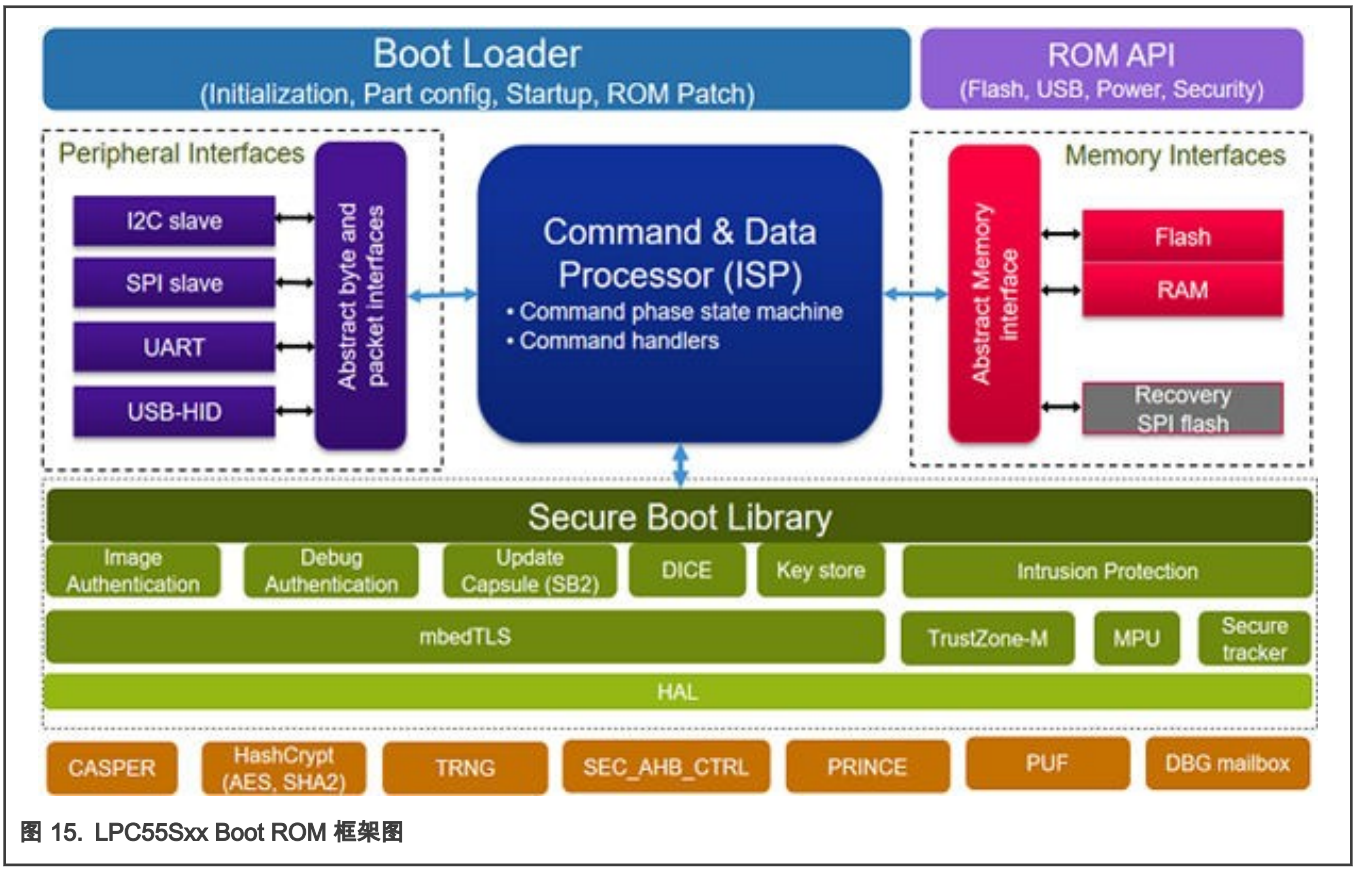

图 15. LPC55Sxx Boot ROM 框架图

镜像验证是一个两步过程。第一步是验证嵌入在镜像中的证书。这包含在第二步中用于验证整个镜像签名（包括证书）的镜像公钥，以允许客户添加额外的 PKI 结构。ROM 使用 SHA-256 和 RSA 签名来验证并从嵌入在图像中的 x509 证书中提取镜像公钥以进行镜像身份验证。RSA 密钥大小由 SECURE_BOT_CFG 控制字中的“RSA4K”字段确定。默认值为 2048 位（公钥模数大小）。如果设置了标志 RSA4K，则强制使用 4096 位密钥。

在 x509 证书中，有一个字段（序列号）必须与来自 PFR 中的值进行比较。如果序列号（可以信任，因为它经过签名）与 PFR 不一致，镜像验证过程将失败。如果序列号相同，则固件可以允许固件更新。序列号可以是更高版本（仅高 1 个版本），但不能是早期版本。它可以向前跳，但不能后退。

5 总结

5.1 建立可信供应链

恩智浦将为您提供一套全面的工具，带您完成生命周期的每个阶段，包括 MCUXpresso IDE、配置和时钟工具以及 Elf2SB GUI。恩智浦提供了一套强大的软件驱动程序、示例项目、应用笔记，和培训以帮助您。 SDK 中有演示项目，用于演示 LPC55Sxx 中每个安全硬件的驱动程序以及与 AWS 服务器的安全通信演示。恩智浦的合作伙伴可以进行渗透测试、认证加密和 TRNG 硬件，可以提供云端设备和用户管理、批量编程和云端数据管理。

5.2 去创造

使用 LPC55Sxx、软件、工具和方法，您可以创建支持设备生命周期内安全交互的事物。随着 LPC55Sxx 等 MCU 的出现，可以实现物联网的安全要求。LPC55Sx MCU 系列利用 Arm Cortex-M33 技术，结合了显着的产品架构增强和比前几代更高的集成度；使用加速器和高级安全功能提供功耗改进，包括基于 PUF 的 ROT 和配置、加密图像的实时执行以及使用 Arm TrustZone-M（TZ-M）的资产保护。如本应用笔记中详述的，安全目标是通过集成到 LPC55Sxx 中的功能实现的。恩智浦提供这款 SoC，让您能够创建在产品的整个生命周期内都值得信赖的低成本、低功耗 IOT 设备。

6 资料

以下是进一步研究 LPC55Sxx MCU 的基本安全功能和实现所需的资源：

1. LPC5500 MCU Series
2. LPC55S6x MCU Family Fact Sheet
3. *LPC55Sxx Secure Boot*（文档 AN12283）
4. Intrinsic ID White Papers on IoT Security and PUF
5. PC55S69-EVK LPCXPRESSO55S69 DEVELOPMENT BOARD
6. 代码签名工具

   elf2USB GUI - 工具文档在 LPC55S69 SDK 下，路径为*.../LPCXpresso55S69/middleware/mcu-boot/doc*。
7. LPC5500 MCU Series: Securing the edge with worlds first ARM Cortex M33 MCU

   https://www.nxp.com/video/lpc5500-mcu-series-securing-the-edge-with-worlds-first-arm-cortex-m33-based-mcus:NXP-LPC5500-VIDEO
8. IoT Solutions -Arm Special Edition by Lawrence C. Miller
9. *LPC55Sxx Usage of the PUF and Hash Crypt to AES Coding*（文档 AN12324）

   本应用笔记描述了如何使用根密钥安全地生成、存储和检索用户密钥。
10. Advance your IoT Security Leveraging Hardware Protected Keys on Microcontrollers

    https://www.nxp.com/design/training/advance-your-iot-security-part-1-protected-keys-on-broad-market-mcus:TIP-ADVANCE-YOUR-IOT-SECURITY
11. ARM+NXP Webinars on LPC5500

    https://pages.arm.com/webinar-recording-achieving-secure-execution-environments-typ.html?aliId=eyJpIjoiOVVaTjEwY2tuRXNSQ3czUSIsInQiOiJIeVwveVQrbjNkUFRtandKK1NkYktKQT09In0%3D
12. *LPC55xx Secure GPIO and usage*（文档 AN12326）
13. Element14 Securing IoT Sensors with NXP LPC5500 MCUs and Trustzone Block
14. DICE - Trusted Computing Group Specification

    Microsoft 开发了一套 DICE 代码。如要了解更多细节，请查看 DICE: Device Identifier Composition Engine。

7 修订记录

表 7 总结了自初始版本以来对本文档所做的更改。

表 7. 修订记录

版本号	日期	说明
0	2019 年 2 月	初始版本

下页继续...

表 7. 修订记录（续上页）

版本号	日期	说明
1	2020 年 5 月	增加有关 LPC55S2x 和 LPC55S1x MCU 的安全信息。添加新的安全功能表。增加额外的资源。

8 词汇表

AC — Activation Code，PUF 的激活码。

AES — 高级加密标准 - AES 引擎支持 128 位、192 位或 256 位密钥的加解密操作。

API — Application Processor Interface

CASPER — 带有 RAM 协处理器引擎的加密加速器和信号处理引擎

CRC — 循环冗余校验

CRYPT — 密码学

CBC — Cypher Block Chaining

CDI — 复合设备标识符 - 从唯一设备密钥和用于识别系统上运行的用于生成此数据的软件的第一个可变代码的身份派生的身份标识值。

CFPA — 客户现场可编程区域

CMPA — 客户制造/工厂可编程区域

CTR — 计数器加密方式

DICE — 设备标识符组合引擎（DICE） — DICE 是不可变的并创建 CDI

DMA — 直接内存访问，可以将一个 DMA 配置为安全

DSP — 数字信号处理器

ECB — 电子密码本

ECC — 椭圆曲线密码学

FIPS — 联邦信息处理标准，出版物 197

HASH — 用于验证文件的镜像摘要的算法

HMAC — HMAC（K，x） — 使用密钥 K 将 HMAC-SHA256 算法应用于 8 位字节字符串 x。

ICB — I Code book

IoT — 物联网

IP — 知识产权

IV — 初始化向量 — 与 AES-CBC 模式一起使用

NMPA — 恩智浦制造可编程区域

NS — 非安全

NVIC — 嵌套向量中断控制器。 每个 CPU 有一个 NVIC。

M2M — 机器对机器，通常是一种通信方式

MCU — 微控制器单元

MITM — 中间人。尝试从 IoT 获取凭据的攻击

MPC — 内存保护检查器 — ROM 和每个 RAM bank 都有相关的 MPC。Flash 也有 MPC。

Onboarding — 为客户端提供凭据以访问网络资源并分配适当权限的过程。

OTA — Over The Air, 指从远程网络更新固件

PFR — 受保护的闪存区域

PKI — 私钥接口

PRINCE — 使用“Prince”算法的片上闪存自动加密/解密模块

PUF — 物理不可克隆的功能

ROM — 只读存储器

RSA — 公钥密码学加密算法

RTF — 运行时指纹

RoT — 信任根

RoTKH — 信任密钥散列的根

S — 安全

SAU — 安全属性单元—LPC55Sxx 上的 NXP 安全模块

SHA — 一种摘要算法

SoC — 指的是微控制器或微处理器

TEE — 可信执行环境

Thing — 连接物联网中其他设备的设备，例如家庭自动化设备、健身可穿戴设备、智能仪表、楼宇门禁、传感器

TPM — 可信平台模块

TRNG — 真随机数生成器

TZ-M — 基于 ARM V8M 规范的 Cortex M 的 TrustZone

UDS — 设备唯一机密。 UDS 只有制造商和 DICE 知道，并用于 DICE 创建 CDI。

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

Limited warranty and liability — Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

Right to make changes - NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Security — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer’s applications and products. Customer’s responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer’s applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 2020 年 5 月
Document identifier: AN12278

arm