LPC55S36

SESIP Security Target

Rev. 1.0 — 12 October 2023

Document information

Information	Content
Keywords	SESIP, PSA, Security Target, LPC55S36
Abstract	Security target for evaluation of the LPC55S36 developed and provided by NXP Semiconductors, according to SESIP Assurance Level 3 (SESIP3) based on SESIP methodology, version 1.1, and PSA Certified Level 3

Revision History

Rev.	Date	Description
1.0	12 October 2023	• First released version

1 Introduction

This Security Target describes the LPC55S36 platform and the exact security properties of the platform that are evaluated against GlobalPlatform Technology Security Evaluation Standard for IoT Platforms (SESIP), version 1.1, SESIP Assurance Level 3 (SESIP3) [1].

1.1 ST Reference

LPC55S36, SESIP Security Target, Revision 1.0, NXP Semiconductors, 12 October 2023.

1.2 SESIP Profile Reference and Conformance Claims

Table 1. SESIP Profile for Secure MCUs and MPUs Conformance Claims

Reference	Value
SP Name	GlobalPlatform Technology SESIP Profile for Secure MCUs and MPUs [2]
SP Version	Version 1.0
Assurance Claim	SESIP Assurance Level 3 (SESIP3)
Package Claim	Base SP, Package Security Services, Package Software Isolation, Package Hardware Protection

Table 2. SESIP Profile for PSA Certified Level 3 Conformance Claims

Reference	Value
SP Name	SESIP Profile for PSA Certified Level 3 [3]
SP Version	V1.0 REL 01
Assurance Claim	SESIP Assurance Level 3 (SESIP3)
Optional and Additional SFRs	See Section 4.3

1.3 Platform Reference

LPC55S36

Table 3. Platform Reference

Reference	Value	
Platform Name and Version	See Table 5	
Platform Identification	Chip name and version,	LPC55S36, 1B
	PSA-RoT name and version	LPC55S36 SDK, 2.14.0 RFP3 RC2 with Attestation Demonstration
Platform Type	Microcontroller platform for IoT applications	
Trusted Subsystem Identification	EdgeLock System S50 (ELS S50), version 2.13.0	

1.4 Included Guidance Documents

The following documents are included with the platform:

Table 4. Guidance Documents

Document	Reference
Reference Manual	LPC55S3x Reference Manual [4]
Datasheet	LPC55S3x Product Data Sheet [5]
SESIP Security Target	LPC55S36, SESIP Security Target, Revision 1.0, NXP Semiconductors, 12 October 2023.
API Reference Manual	User Manual of Crypto Library Normal Secure (CLNS) [6]
Application Note	AN13443 Secure boot on LPC55S3x [7]
Application Note	AN13529 Debug authentication on LPC55S3x [8]
Application Note	Attestation on LPC55S3x [9]
Tool User Guidance	User Guide for MCUXpresso Config Tools [10]
Tool User Guidance	blhost User's Guide [11]
Application Note	AN6259, Common Trust Provisioning Conceptional Overview [13]
Software Development Kit	LPC55S36 SDK, 2.14.0 RFP3 RC2 with Attestation Demonstration

1.5 Platform Overview and Description

1.5.1 Platform Security Features

LPC55S36 employs a security subsystem, EdgeLock System S50 (ELS S50, legacy name CSS or CSSv2), which together with its driver provides the following security features:

- AES 128/192/256 with ECB, CBC, CTR and GCM mode
- ECDSA and ECDHE with p-256
- SHA2 256/384/512
- HMAC, CMAC
- TLS key derivation and key store
- TRNG and DRBG
- Key wrapping and management
- Dedicated DMA controller
- Attestation Service

On top of ELS S50, LPC55S36 provides the following security features at SoC level:

- Arm TrustZone enabled
- Secure boot, update and debug authentication
- Physical Unclonable Function (PUF) that can generate, store, and reconstruct key sizes from 64 to 4096 bits, and be directly fed to ELS S50
- 128 bit unique device serial number for identification (UUID)
- Secure GPIO
- Intrusion and Tamper detection and response sub-system

- Device Identifier Composition Engine (DICE)

Particular for Secure Boot and Update LPC55S36 supports:

- Secure boot using ECDSA P-256/P-384 signed images
- Uses custom certificate format to validate image public keys
- Up to four revocable Root of Trust (RoT) or Certificate Authority keys, Root of Trust establishment by storing the SHA-2 hash digest of the hashes of up to four RoT public keys in protected flash region (PFR)
- Anti-rollback feature for firmware update and revocable image signing keys/certificates.
- PFR authentication using OTP-eFuse and CMAC computed using DUK (Device Unique Key)
- Image authentication APIs and authentication of XIP images
- Booting of SB3.1 signed & AES encrypted images over serial interfaces (UART, I2C, SPI-slave, USB-HID)
- SB commands to program flash, OTP-eFuse, PFR, PUF provisioning, QSPI flash programming, write to RAM and execute RAM (after image authentication). SB commands in recovery boot supports commands including flash/PFR/OTP programming
- SB3 firmware update APIs
- Boot ROM supports Device Identifier Composition Engine (DICE) Specification (version Family 2.0, Level 00 Revision 69) specified by Trusted Computing Group

For more product features beyond security, refer to Chapter 2 of [4].

1.5.2 Platform Type

Processor with internal hardware isolation with Arm TrustZone technology, secure memory, and a secure subsystem.

1.5.3 Platform Physical Scope

The physical scope is the LPC55S36 microcontroller silicon chip as shown in Figure 1.

The hardware components and interfaces are listed in Chapter 2 of [4].

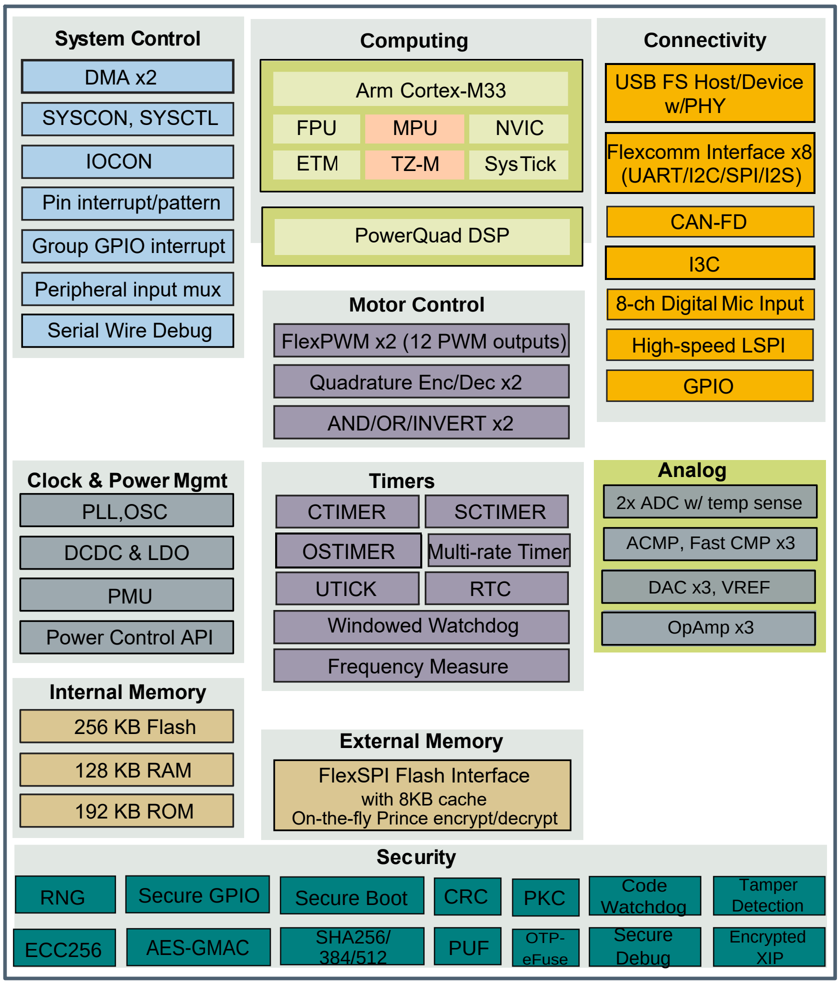

Figure 1. LPC55S36 Block Diagram

1.5.4 Platform Logical Scope

The logical scope includes the ROM firmware, and the optional flash loadable updatable platform root of trust (RoT) as illustrated in Figure 2 and listed in Table 5. Any additional firmware, OS or application software stored on the platform is not in scope of this evaluation.

Table 5. Platform Deliverables

Type	Name	Release	Form of delivery
IC Hardware	LPC55S36	1B	Silicon Chip
ROM Firmware	LPC55S36 ROM	K3.1.1	Onchip Firmware
ROM Firmware Patch	LPC55S36 ROM Patch	T1.1.2	Onchip Firmware
Security Enclave	EdgeLock System S50 (ELS S50)	2.13.0[1]	Onchip Hardware Subsystem

Table 5. Platform Deliverables*...continued*

Type	Name	Release	Form of delivery
Updatable Platform RoT	LPC55S36 SDK	2.14.0 RFP3 RC2 with Attestation Demonstration	Software Package
Crypto Library	Crypto Library Normal Secure for LPC55S36 SDK	1.5.0	Included in Software Package

[1] Version information as defined in Chapter 37.5.19 of [4].

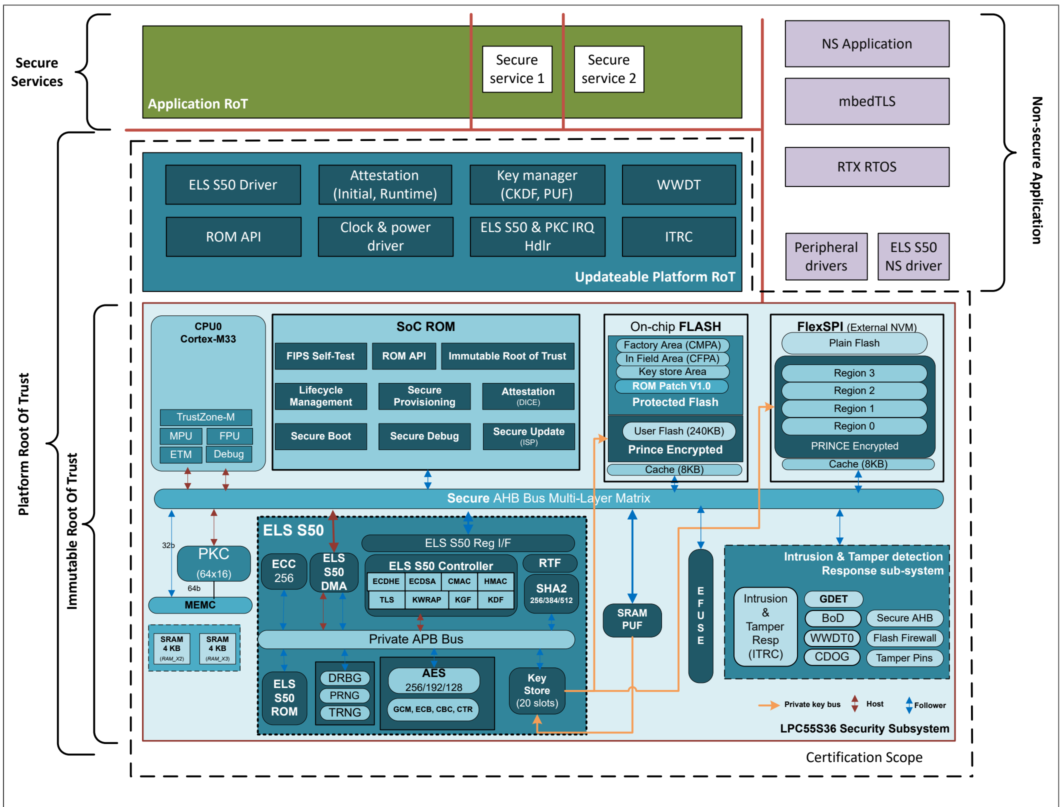

Figure 2. LPC55S36 Logical Architecture and Certification Scope

1.5.5 Required Non-Platform Hardware/Software/Firmware

No additional non-platform hardware, software or firmware is required for the correct functioning of the security claims described in this document except for Section 3.2.5.2. For security claim of Section 3.2.5.2, compatible external non-volatile memory shall be deployed via FlexSPI. See Chapter 19 of [4] for more information.

1.5.6 Life Cycle

This device supports a security life cycle state model. The current life cycle state determines the device functionality, debug and test port availability, and asset accessibility. The life cycle state is controlled by the LC_STATE fuse value, and state values are selected so that additional fuse bits are burned to advance the state. Because fuses control the life cycle state, moving to a more advanced state is an irreversible and permanent process. The life cycle can only be advanced and can't return to a previous state.

The Boot ROM is responsible for checking the life cycle state. Based on the life cycle state the ROM will determine what boot flow is used, including if control will be passed to application code or not. The ROM also handles the opening of test and debug ports based on the life cycle state. If the part is in the Bricked state or any invalid life cycle state, then the ROM will lock the part.

See more in Chapter 33 of [4].

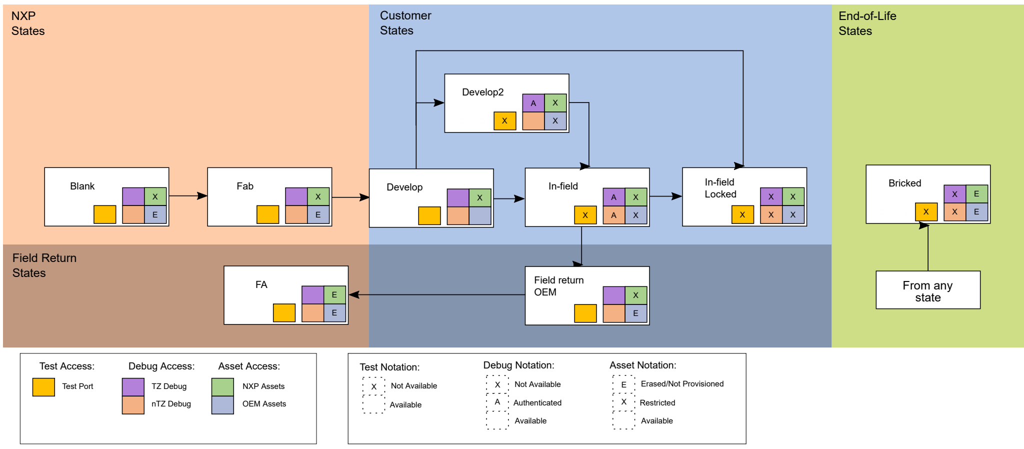

Figure 3. LPC55S36 Life Cycles

1.5.7 Configurations

The MCU/MPU ensures the execution of platform trusted code, particularly the functions related to secure boot, updatability, and code isolation.

The security features discussed above are complemented by security services intended to be used by the higher software layers to implement a full-fledged Root of Trust and operating system

1.5.8 Use Case

[any user]

The product may be physically accessed by an unknown or untrusted user, in an environment where access to the product cannot be sufficiently controlled or even in a more hostile environment.

[any code]

It cannot be excluded that the product will execute code that is unknown to the product developer.

2 Security Objectives for the Operational Environment

2.1 Platform Objectives for the Operational Environment

For the platform to fulfill its security requirements, the operational environment (technical or procedural) must fulfill the following objectives:

Table 6. Platform Objectives for the Operational Environment

Title	Description	Reference
Platform Verification	The operating system or application code are expected to verify the correct version of all platform components it depends on, as described in Section 3.2.1.1 of this document.	Section 3.2.1.1
Secure Boot	The operating system or application code are expected to make use of the Secure Boot feature as described in Chapter 30 of [4].	[7] and Chapter 30 of [4]
Secure Debug	The integrating environment is expected to configure the debug functionality as described in Chapter 63.3.8 of [4] and [8] to meet the extra physical attacker resistance.	[8] and Table 666 in Chapter 63.3.8 of [4]
Key Management	Cryptographic keys and certificates outside of the Platform are subject to secure key management procedures.	This document
Trusted Users	Actors in charge of platform management, for instance for signature of firmware update, are trusted.	This document
SW Integration	The operating system or application code are expected to ensure the correct version of the crypto library and SDK drivers are integrated and configured.	This document
Secure Update and Key Revoke	The operating system or application code are expected to update an image with proper remedy solution and version increased and/or revoke key in case of security incidence occurrence of the image and/or the key.	Chapters 27, 29 and 30 of [4]
Lifecycle Management	The operating system or application code are expected to provide lifecycle states and secure mechanism of lifecycle state transition according to the use case, and the operational environment is expected to configure the platform accordingly for lifecycle state transitions. In general, the operating system or application code are expected to configure the platform to In-field or in-field locked state.	Chapter 34 of [4]
Software Isolation	LPC55S36 provides majorly two different isolation mechanisms: S50 vs rest of SoC, and TrustZone Secure vs Non-secure. The operating system or application code are expected to configure and utilize at least one mechanism for isolation between platform and application.	Chapters 36 and 38 of [4]

Table 6. Platform Objectives for the Operational Environment...*continued*

Title	Description	Reference
Physical Attacker Resistance Configurations	If local physical attack is applicable for the use cases, the following configurations shall apply: • The operating system or application code are expected to configure the `main_clk` to one of the internal clock sources; • The operating system or application code are expected to keep the ITRC output configured to `CHIP_RESET`; • The operating system or application code are expected to configure the security sensor settings as the `helloworld` example in SDK at application initialization; • The operating system or application code are expected not to be executed from external flash with XIP mode. • The operating system or application code are expected not to use printf or any external console output functions in secure partition. • ARM CM33 CPU is not hardened against physical attacks, e.g., voltage glitching or EMFI. It is therefore recommended to harden secure application against such attacks using software-based countermeasures and leverage code watchdog offered by LPC55S36.	This document, [10], Chapters 8.1.6, 35.4.2, 48 of [4], and LPC55 S36 SDK, 2.14.0 RFP3 RC2 with Attestation Demonstration

3 Security Requirements and Implementation

3.1 Security Assurance Requirements

The claimed assurance requirements package is: **SESIP Assurance Level 3 (SESIP3)** as defined in Chapter 4 of GlobalPlatform Technology Security Evaluation Standard for IoT Platforms (SESIP), version 1.1 [1].

3.1.1 Flaw Reporting Procedures (ALC_FLR.2)

In accordance with the requirement for flaw reporting procedures (ALC_FLR.2), the developer has defined the following procedure:

NXP has defined a Product Security Incident Response Process (PSIRP), implemented by a dedicated team (PSIRT). This process provides a publicly available interface (https://nxp.com/psirt), and includes four major steps:

- **Reporting**. The process begins when the PSIRT becomes aware of a potential security vulnerability in an NXP product. The reporter receives an acknowledgment and updates throughout the handling process.
- **Evaluation**. The PSIRT confirms the potential vulnerability, assesses the risk, determines the impact and assigns a processing priority. If the vulnerability is confirmed, the priority determines how the issue is handled throughout the remaining steps in the process.
- **Solution**. Working with PSIRT, the product team develops a solution that mitigates the reported security vulnerability. Solutions will take different forms based on the vulnerability. Because of the nature of NXP products – mostly silicon products where the firmware is in ROM -, very often the solution can only be provided in a next version of the chips and the short-term solution will consist of recommending security measures to be applied in systems using the NXP product.
- **Communication**. As said above, because of the nature of the NXP products, the solution to systems using the affected products often needs to be found in additional countermeasures in those systems. The communication on the vulnerability and solutions will in most cases be done directly towards the affected customers. For previously unknown or unreported issues, NXP will acknowledge the reporter of the issues (unless the reporter requests otherwise).

The platform's Secure Boot feature is able to verify the authenticity of customer code during the initial boot and outside of the boot sequence, providing an appropriate mechanism for supporting the update of this code. The update mechanism is also supported in the loadable firmware of TF-M. See Section 3.2.2.1 for further information.

3.2 Security Functional Requirements

In the following Security Functional Requirements, the term **platform** covers the **LPC55S36 physical and logical scope**, and the term **application** refer to any additional firmware, OS or application software which is out of evaluation scope. It represents a part of the final connected device.

LPC55S36 fulfils the following security functional requirements:

3.2.1 Identification and Attestation of Platforms and Applications

3.2.1.1 Verification of Platform Identity

The platform provides a unique identification of the platform, including all its parts and their versions.

Conformance rationale:

SoC Hardware identifier and revision number can be identified by using `GetProperty` command in ISP mode as specified in Section 27.5.2 and 27.5.15 of [4] with tag `10h` to read `DIEID`. The return value shall match to the value in Section 8.5.1.161 and the version shall match the value indicated in Section 1.5.4 (for version 1B, the field `REV_ID` of `DIEID` shall be `1`).

The ROM version can be read by `GetProperty` command with tag `01h`, and the return value shall be the same as Section 1.5.4.

The ROM patch revision can be read by `GetProperty` command with tag `18h`, and the return value shall be the same as Section 1.5.4.

The SDK as Updatable Platform RoT Firmware is delivered in logical format as a software library. One can identified the version in readme file and verify the commit hash as defined in Section 1.5.4.

3.2.1.2 Verification of Platform Instance Identity

The platform provides a unique identification of that specific instantiation of the platform, including all its parts and their versions.

Conformance rationale:

The platform stores a 128-bit IETF RFC4122 compliant non-sequential Universally Unique Identifier (UUID). It can be read from the flash PFR region at register location `0x3FC70` onwards.

One way to read out UUID is to use `GetProperty` command in ISP mode with tag `12h` as specified in Section 27.5.2 and 27.5.15 of [4].

Furthermore, LPC55S36 supports Device Identifier Composition Engine (DICE) Specification (version Family 2.0, Level 00 Revision 69) specified by Trusted Computing Group, which provides another way to uniquely identify a product instance.

NXP also provide trust provisioning service, where a certificate is injected during NXP manufacturing which can be used to verify the platform instance identity and genuineness. See more in [13]

3.2.1.3 Attestation of Platform Genuineness

The platform provides an attestation of the “Verification of Platform Identity” and “Verification of Platform Instance Identity”, in a way that cannot be cloned or changed without detection.

Conformance rationale:

Secure Attestation is a set of mechanisms used to provide evidence to a remote party on the device’s genuine identity, its software and firmware versions, as well as its integrity and lifecycle state. Device Identity Composition Engine (DICE), as defined by Trusted Computing Group, uses Immutable RoT during boot time to create a unique Device Identity which takes into account Unique Device Secret (UDS), hardware state of the device and its firmware. Runtime Fingerprint (RTF) is the NXP-proprietary attestation mechanism, which measures the device's state during boot-time and run-time as well. See more in [9].

Trust provisioning is a process used for creation of initial Device Identity keys. Its major objective is to provide a cryptographic proof of the device's origin and to offer a set of tools to OEM for secure provisioning of their own assets. In a nutshell, a device-unique private-public key pair is created on every device, the public portion of which is collected and signed by NXP. That signed public key is installed back onto every device in a form of device-unique certificate, which serves the actual proof of the device's origin. See more in [13].

3.2.1.4 Attestation of Platform State

The platform provides an attestation of the state of the platform, such that it can be determined that the platform is in a known state.

Conformance rationale:

See Section 3.2.1.3.

3.2.1.5 Secure Initialization of Platform

The platform ensures its authenticity and integrity during the platform initialization. If the platform authenticity or integrity cannot be ensured, the platform will go to *reset state*.

Conformance rationale:

Secure ROM boot loader LPC55S36 provides secure boot operation.

Secure boot prevents unauthorized code from being executed on a given product. It achieves this level of security by always leaving the device's ROM in an executing mode when coming out of a reset. This allows the ROM to examine the first user executable image resident in internal flash memory to determine the authenticity of that code. If the code is authentic, then control is transferred to it. This establishes a chain of trusted code from the ROM to the user boot code. This chain can be further extended, through the verification of digital signatures associated with additional code layers.

Cipher-based Message Authentication Code (CMAC, 128 or 256 bit key) and Elliptic Curve Digital Signature Algorithm (ECDSA, P-256 with SHA256 or P-384 with SHA384) are used in this architecture to verify authenticity of the boot code. The boot code is always signed with ECDSA private keys. The corresponding ECDSA public keys used for signature verification (Root of Trust Keys) are contained in certificate block that is contained in the signed image. Support is provided for up to four Root of Trust keys.

During first boot of the application image, the ECDSA algorithm is always used to verify authenticity of the image. After that, either CMAC or complete ECDSA verification can be performed according to configuration in SEC_BOOT_EN field in Customer Manufacturing/Factory Programmable Area (CMPA). See more in Chapter 29 of [4].

The boot image can further be stored in the prince encrypted region for confidentiality protection. See more in Section 3.2.5.1 and Section 3.2.5.2. The Updatable Platform RoT Firmware include TF-M ported is protected by the abovementioned features.

The operating system or application code have the option to further enable built-in self-tests in Secure Boot ROM to ease the certification of NIST CMVP. See more in Table 276 of Chapter 29.4 of [4].

ROM supports the dual image boot, that means, two boot images can be placed, either in internal flash or in external flash; ROM decides to boot which image based on the image version, boot the one with the newer image version first, if fail, boot the older one. See more in Chapter 26.3.1 of [4].

LPC55S36 also supports Secure boot by SB3.1 file from FlexSPI interface, which can be used for external flash boot, external boot recovery as well as ease for OEM manufacture.

3.2.2 Product Lifecycle: Factory Reset / Install / Update / Decommission

3.2.2.1 Secure Update of Platform

The platform can be updated to a newer version in the field such that the integrity, authenticity and confidentiality of the platform is maintained.

Conformance rationale:

Secure ROM boot loader LPC55S36 provides secure firmware update operation.

Secure Update is the process used to securely update the firmware image in the field. The firmware image is encrypted using AES-128 or AES-256 and signed using ECDSA P-256 or ECDSA P-384, following the SB3.1 firmware image format. Secure Update guarantees authenticity and confidentiality of the new image. It also ensures that the new image is up-to-date, preventing the rollback to an older image. Running firmware is in charge of receiving and verifying the new firmware image. The follow-up Secure Boot verifies the new firmware image again, making sure the Immutable RoT is still in charge of ensuring authenticity of the latest firmware.

The anti rollback is achieved by 32-bit monotonic counter for secure firmware version, 32-bit monotonic counter for image key revocation, and 4 revokable RoT. The new FW version value must be equal to or greater than the counter to be acceptable else rollback will be detected. See more in Chapter 29.4.3 of [4].

Also the dual image boot and secure boot from FlexSPI interface provide recovery capability for the device. See more in Section 3.2.1.5.

Furthermore, trust provisioned private key, together with other pre-installed key material, is then used for authentication and secure connection to the device, enabling secure provisioning of OEM assets even in the manufacturing environment OEM may not fully trust. See more in [13].

3.2.2.2 Field Return of Platform

The platform can be returned to the vendor without user data.

Conformance rationale:

LPC55S36 provides secure Field Return feature.

In the Field Return OEM state, every boot ROM verifies that the customer key store (`0x3E400` - `0x3E5FF`) is blank. If not, erases before opening debug access. PUF and ELS S50 modules are put in FA mode which will re-key all application usage keys including memory encryption Prince keys. This mechanism protects leakage of any residue data left during life-cycle state transition.

It can further move to FA life cycle state if the device is being returned to NXP for testing and failure analysis, and further sensitive information is erased.

See more in Chapters 33.2.5 and 33.2.6 and 62.3.10 of [4].

3.2.2.3 Decommission of Platform

The platform can be decommissioned.

Conformance rationale:

The End-of-Life security life cycle state, or Bricked State, can be used by customers or NXP to remove a chip permanently from regular use and erase/block access to secrets inside the chip. See more in Chapter 33.2.7 in [4].

3.2.3 Extra Attacker Resistance

3.2.3.1 Physical Attack Resistance

The platform detects or prevents attacks by an attacker with physical access before the attacker compromises any of the functional requirements, ensuring that the functional requirements are not compromised.

Conformance rationale:

LPC55S36 is equipped with Intrusion and Tamper Response Controller (ITRC). ITRC provides mechanism to configure the response action for an intrusion event detected by an on chip security sensors. Intrusion Response is the action a device performs in order to prevent misuse of the device or disclosure of critical assets (cryptographic keys, personal data) that are generated or stored within the device. The response mechanism is typically triggered by either a signal from an on-chip sensor designed to detect that the device is in a threat condition or by an explicit command provided by the software. See more in Chapter 35 of [4].

Also, the software components including ROM leverage the code watchdog. For code watchdog, see more in Chapter 48 of [4]. The crypto coprocessor and the library are secure hardened against potential physical attacks.

Furthermore, this device has one instance of the independent real time clock, RTC. This block is a low power module that provides time keeping and calendaring functions and additionally provides protection against tampering (external or internal tamper events), protection against spurious memory/register updates and battery operation. See Chapter 11 of [4]. This function provides another layer of protection yet needs further HW support at board level, hence, not in the evaluation scope.

3.2.3.2 Software Attacker Resistance: Isolation of Platform (between SPE and NSPE)

The platform provides isolation between the application and itself, such that an attacker able to run code as an application on the platform cannot compromise any other claimed security functional requirements.

Conformance rationale:

There are multiple isolation features presented in the platform.

The ELS S50 module is a security subsystem supporting a wide range of cryptographic algorithms and providing strong key isolation from the rest of the system. When embedded in an SoC, ELS S50 serves as the main building block of the SoC's immutable Root of Trust. It is used as part of the trust anchor during secure boot, secure debug access, life-cycle management, and trust provisioning.

ELS S50 has its own controller and exclusive system resources with enforced access control, hence it is isolated from the rest of platform. See more in Chapter 36 of [4].

PRINCE-based memory encryption also ensures Secure Isolation between multiple IP vendors. Initial Vector (IV) is derived by secure-privilege and a different value is used for every independent memory region, ensuring the isolation between each other. See more in Chapter 34 of [4].

Furthermore, LPC55S36 provides Protected Flash Region (PFR) and ROM API to flash firewall setup and access control. See more Chapter 26.5, 28.2.2, 29.2.4 of [4].

ARM TrustZone enables Secure Isolation during run-time by providing four distinct levels of privilege: secure-privilege, secure-user, non-secure-privilege, non-secure-user. Every peripheral is equipped with Peripheral Protection Checker (PPC) that can be programmed to control access to that peripheral, following the ARM TrustZone philosophy. Every memory is equipped with Memory Protection Checker (MPC) that can also be programmed in the same way as the PPC. Secure AHB Controller is in charge of programming all PPC and MPC blocks and only the highest level of privilege, which is secure-privilege, is allowed to do that. See more in Chapter 38 of [4].

3.2.3.3 Software Attacker Resistance: Isolation of Platform (between PSA-RoT and Application Root of Trust Services

The platform provides isolation between the application and itself, such that an attacker able to run code as an application on the platform cannot compromise any other claimed security functional requirements.

Conformance rationale:

The isolation between PSA-RoT and Application Root of Trust Services is included in Section 3.2.3.2.

Also see more in Section 3.2.4.3 about key isolation.

3.2.4 Cryptographic Functionality

3.2.4.1 Cryptographic Operation

The platform provides the application with *operations in Table 7* functionality with *algorithms in Table 7* as specified in *specifications in Table 7* for key lengths *described in Table 7* and modes *described in Table 7*.

Table 7. Cryptographic Operations

Operation	Algorithm	Specification	Key Lengths	Modes
Encryption and decryption	AES	NIST FIPS 197	128, 192, 256 [1][2]	ECB, CBC, CTR
Authenticated Encryption, Authenticated Decryption	AES	NIST SP.800-38d	128, 192, 256 [1][2]	GCM
Hashing	SHA2	NIST FIPS 180-4	224, 256, 384, 512	-
MAC generation and verification	HMAC	RFC2104	Up to 512 [1][3]	SHA-256
MAC generation and verification	CMAC	RFC4493	128, 256 [1][2]	AES

Table 7. Cryptographic Operations...*continued*

Operation	Algorithm	Specification	Key Lengths	Modes
Signature generation and verification	EdDSA	NIST FIPS 186-5	255	Ed25519
Signature generation and verification	ECDSA	NIST FIPS 186-5	192, 224, 256, 384, 521	secpXXXr1, XXX = key length
			192, 224, 256	secpYYYk1, YYY = key length
			160[4], 192, 224, 256, 320, 384, 512	brainpoolPZZZr1, ZZZ = key length
Signature generation and verification	RSA	PKCS v1.15 and RSA PSS	2048, 3072, 4096	-

[1] All key lengths supported by key stored in memory outside of ELS S50.
[2] ELS S50 keystore available for 128- and 256-bit keys.
[3] ELS S50 keystore available for 256-bit keys.
[4] Refer to [12] for considerations on algorithm and key lengths.

Conformance rationale:

The crypto coprocessors are located in ELS S50. Crypto Library for ELS S50 has been developed leveraging these coprocessors. ELS S50 provides the symmetric, hashing, and more functions. See more in [6].

On top of ELS S50 security coprocessors, LPC55S36 also deploys Public-Key Crypto Coprocessor (PKC, Chapter 37 of [4]). Crypto Library has been developed leveraging it for public key algorithms.

3.2.4.2 Cryptographic Key Generation

The platform provides the application with a way to generate cryptographic keys for use in *algorithms in Table 8* as specified in *specifications in Table 8* for key lengths *described in Table 8*

Table 8. Cryptographic Key Generation

ID	Algorithm	Specification	Key Lengths
AES	AES	NIST SP800-133	128, 192, 256
ECC	ECC	ANSI X9.62	160, 192, 224, 255, 256, 320, 384, 512, 521
RSA	RSA	PKCS#1	2048, 3072, 4096
HKDF	HKDF	RFC5869	128, 256
CKDF	CKDF	NIST 800-108	128, 256
TLS KDF	TLS Master Key Derivation	TLS1.2	-
TLS KDF	TLS Session Key Derivation	TLS1.2	-

Table 8. Cryptographic Key Generation*...continued*

ID	Algorithm	Specification	Key Lengths
ECDH	ECDH	NIST SP 800-56A	255 (Curve25519), 448 (Curve448)
			192, 224, 256, 384, 521 (secpXXXr1, XXX = key length)
			192, 224, 256 (secpYYYk1, YYY = key length)
			160[1], 192, 224, 256, 320, 384, 512 (brainpoolPZZZr1, ZZZ = key length)

[1] Refer to [12] for considerations on algorithm and key lengths.

Conformance rationale:

The crypto library also provides key generation service leveraging the coprocessors.

3.2.4.3 Cryptographic KeyStore

The platform provides the application with a way to store *cryptographic keys* such that not even the application can compromise the *authenticity, integrity, confidentiality* of this data. This data can be used for the cryptographic operations *encryption, decryption, signature generation, MAC generation and verification, key derivation, shared secret generation*.

Conformance rationale:

ELS S50 provides key store function. The input key is one of the following:

1. The device unique key (DUK), a master key which is transferred from PUF via a dedicated hardware interface.
2. An encrypted (wrapped) key in system memory. KeyIn unwraps these keys before writing them to keystore. ELS S50 key wrapping uses the algorithm defined in the RFC3394 standard.

See more in Chapter 36.5.2.7 and 36.5.2.8 of [4].

One can further use Physically Unclonable Function (PUF) for keystore. See more in Chapter 40 of [4].

3.2.4.4 Cryptographic Random Number Generation

The platform provides the application with a way based on *physical noise* to generate random numbers to as specified in *NIST.SP.800-90B*.

The platform provides the application with a way based on *DRBG* to generate random numbers to as specified in *NIST.SP.800-90A CTR-DRBG with AES-128*.

Conformance rationale:

ELS S50 has physical true random number generator and internal DRBG module as defined in NIST SP 800-90A. With dedicated firmware driver, the RNG can archive NIST SP 800-90B compliance as well.

Furthermore:

- TRNG is capable to pass AIS 31 statistical tests T0-T8

See more in Chapters 36.5.6 and 36.5.7 of [4].

3.2.5 Compliance Functionality

3.2.5.1 Secure Encrypted Storage

The platform ensures that all data stored by the application, except for *data not stored in the configured address area*, is encrypted as specified in *PRINCE [14]* with a platform instance unique key of key length *128 bits*.

Conformance rationale:

This device offers support for real-time encryption and decryption for on-chip flash using the PRINCE encryption algorithm. See more in Chapters 26.3.1.1 and 34 of [4].

3.2.5.2 Secure External Storage

The platform ensures that all data stored outside the direct control of the platform, except for *data not stored in the configured address area,* is protected such that the *confidentiality and binding to platform instance* is ensured.

Conformance rationale:

External flash storage can also be encrypted by PRINCE algorithm using IPED engine to achieve confidentiality. The key is stored in ELS S50 and derived from PUF which also provides binding to platform instance. See more in Chapters 19.3.9 and 26.3.1.2 of [4].

3.2.5.3 Secure Debugging

The platform only provides *Arm's Serial Wire Debug (SWD) interface* authenticated as specified in *Chapter 63 of [4]* with debug functionality.

The platform ensures that all data stored by the application, with the exception of *subdomain(s) debug access enabled*, is made unavailable.

Conformance rationale:

The fundamental principles of debugging, which require access to the system state and system information, conflict with the principles of security, which require the restriction of access to assets. Thus, many products disable debug access completely before deploying the product. This causes challenges for product design teams to do proper Return Material Analysis (RMA).

To address these challenges, the chip offers a debug authentication protocol as a mechanism to authenticate the debugger (an external entity) has the credentials approved by the product manufacturer before granting debug access to the device.

The debug authentication is a challenge-response scheme and assures that only the debugger in possession of the required debug credentials can successfully authenticate over the debug interface and access restricted parts of the device. Furthermore, the debug subsystem is sub-divided into multiple debug domains to allow finer access control.

See more in Chapter 62.3.8 of [4] and [8].

3.2.5.4 Residual Information Purging

The platform ensures that *key store areas*, with the exception of *none*, is erased using the method specified in *Chapter 36.5.2.9 of [4]* before the memory is (re)used by the platform or application again and before an attacker can access it.

The platform ensures that *user flash area, CFPA, CMPA and key store areas*, with the exception of *none*, is erased using the method specified in *Section 33.2.5.3.2 and Chapter 62.3.8.1.4 of [4]* before the memory is (re)used by the platform or application again and before an attacker can access it.

Conformance rationale:

ELS S50 provide KDELETE command which removes the key and zeroise the register. See more in Chapter 36.5.2.9 of [4]. At driver side, API `mcuxClKey_flush()` is provided.

Another instance for residual information purging is to enter the FA Mode (`SET_FA_MODE`), or Bulk Erase Flash if they are enabled in `DCFG_CC_SOCU` credential constraints which requires debug authentication. See more in Section 3.2.2.2 of this document as well as Chapters 62.3.8.1.4, 62.3.8.1.5 and 33.1.1 of [4].

3.2.5.5 Reliable Index

The platform implements a strictly increasing function.

Conformance rationale:

Anti roll back mechanism is employed as described in Section 3.2.2.1.

Also CFPA page provides 8 customer defined monotonic counters, see Chapter 29.4.3 of [4].

4 Mapping and Sufficiency Rationales

4.1 SESIP3 Sufficiency

Assurance Class	Assurance Family	Covered By	Rationale
ASE: Security target evaluation	ASE_INT.1 ST Introduction	Section 1	The ST reference is in Section 1.1, the TOE reference in Section 1.3, the TOE overview and description in Section 1.5.
	ASE_OBJ.1 Security requirements for the operational environment	Section 2	The objectives for the operational environment in Section 2 refer to the guidance documents.
	ASE_REQ.3 Listed security requirements	Section 3	All SFRs in this ST are taken from [1]. SFR "Identification of Platform Type" is included. SFR "Secure Update of Platform" is mentioned but refers to ALC_FLR.2.
	ASE_TSS.1 TOE Summary Specification	Section 3	All SFRs are listed per definition, and for each SFR the implementation and verification is defined in the SFR.
ADV: Development	ADV_FSP.4 Complete functional specifications	Material provided to evaluator.	The evaluator will determine whether the provided evidence is suitable to meet the requirement.
	ADV_IMP.3 Complete mapping of the implementation representation of the TSF to the SFRs	Material provided to evaluator.	The evaluator will determine whether the provided evidence is suitable to meet the requirement.
AGD: Guidance documents	AGD_OPE.1 Operational user guidance	Section 1.4	The evaluator will determine whether the provided evidence is suitable to meet the requirement.
	AGD_PRE.1 Preparative procedures	Section 1.4	The evaluator will determine whether the provided evidence is suitable to meet the requirement.
ALC: Life-cycle support	ALC_CMC.1 Labelling of the TOE	Material provided to evaluator.	The evaluator will determine whether the provided evidence is suitable to meet the requirement.
	ALC_CMS.1 TOE CM Coverage	Material provided to evaluator.	The evaluator will determine whether the provided evidence is suitable to meet the requirement.

Assurance Class	Assurance Family	Covered By	Rationale
	ALC_FLR.2 Flaw reporting procedures	Section 3.1.1	The flaw reporting and remediation procedure is described.
ATE: Test	ATE_IND.1 Independent testing: conformance	Material provided to evaluator.	The evaluator will determine whether the provided evidence is suitable to meet the requirement.
AVA: Vulnerability assessment	AVA_VAN.3 Focused vulnerability analysis	N.A. A vulnerability analysis is performed by the evaluator to ascertain the presence of potential vulnerabilities.	The evaluator performs penetration testing, to confirm that the potential vulnerabilities cannot be exploited in the operational environment for the TOE. Penetration testing is performed by the evaluator assuming an attack potential of Enhanced-Basic.

4.2 Conformance Mapping for SESIP Profile for Secure MCUs and MPUs

This section provides rationales of conformance claimed in Section 1.2

Table 9. SESIP Profile for Secure MCUs and MPUs Sufficiency

Package Claimed	Security Functional Requirements	Covered By
Base	Verification of Platform Identity	Section 3.2.1.1
	Secure Initialization of Platform	Section 3.2.1.5
	Secure Updated of Platform	Section 3.2.2.1
	Residual Information Purging	Section 3.2.5.4
	Secure Debugging	Section 3.2.5.3
Security Services	Cryptographic Operation	Section 3.2.4.1
	Cryptographic Key Generation	Section 3.2.4.2
	Cryptographic KeyStore	Section 3.2.4.3
	Cryptographic Random Number Generation	Section 3.2.4.4
Software Isolation	Software Attacker Resistance: Isolation of Platform	Section 3.2.3.2, Section 3.2.3.3
Hardware Protections	Physical Attacker Resistance	Section 3.2.3.1
Additional Security Functional Requirements (Optional)	Verification of Platform Instance Identity Attestation of Platform Genuineness Attestation of Platform State Decommission of Platform Field Return of Platform Secure Encrypted Storage Secure External Storage Reliable Index	Section 3.2.1.2 Section 3.2.1.3 Section 3.2.1.4 Section 3.2.2.3 Section 3.2.2.2 Section 3.2.5.1 Section 3.2.5.2 Section 3.2.5.5

4.3 Conformance Mapping for SESIP Profile for PSA Certified Level 3

This section provides rationales of conformance claimed in Section 1.2

Table 10. SESIP Profile for PSA Certified Level 3 Sufficiency

Package Claimed	Security Functional Requirements	Covered By
Base	Verification of Platform Identity	Section 3.2.1.1
	Verification of Platform Instance Identity	Section 3.2.1.2
	Attestation of Platform Genuineness	Section 3.2.1.3
	Secure Initialization of Platform	Section 3.2.1.5
	Attestation of Platform State	Section 3.2.1.4
	Secure Updated of Platform	Section 3.2.2.1
	Physical Attacker Resistance	Section 3.2.3.1
	Software Attacker Resistance: Isolation of Platform (between SPE and NSPE)	Section 3.2.3.2
	Software Attacker Resistance: Isolation of Platform (between PSA-RoT and Application Root of Trust Services)	Section 3.2.3.3
	Cryptographic Operation	Section 3.2.4.1
	Cryptographic Key Generation	Section 3.2.4.2
	Cryptographic KeyStore	Section 3.2.4.3
	Cryptographic Random Number Generation	Section 3.2.4.4
Optional SFR	Secure Debugging	Section 3.2.5.3
	Secure Encrypted Storage (internal storage)	Section 3.2.5.1

5 Bibliography

5 . 1 Evaluation Documents

[1] GlobalPlatform Technology Security Evaluation Standard for IoT Platforms (SESIP), version 1.1, GP_FST_070.

[2] GlobalPlatform Technology SESIP Profile for Secure MCUs and MPUs, Version 1.0, GPT_SPE_150.

[3] SESIP Profile for PSA Certified Level 3, V1.0 REL 01, PSA JSA, Oct 2022.

5 . 2 Developer Documents

[4] LPC55S3x Reference Manual, Rev. 2, 11/2022, NXP Semiconductors

[5] LPC55S3x Product Data Sheet, Rev. 2.0, October 2022, NXP Semiconductors

[6] User Manual of Crypto Library Normal Secure (CLNS), CLNS SDK 1.3.0, NXP Semiconductors

[7] AN13443 Secure boot on LPC55S3x, Rev. 0, November 2021, NXP Semiconductors

[8] AN13529 Debug authentication on LPC55S3x, Rev. 0, Feb 2022, NXP Semiconductors

[9] Attestation on LPC55S3x, Draft 0, NXP Semiconductors, Nov 2022

[10] User Guide for MCUXpresso Config Tools, Rev. 4, NXP Semiconductors, Sep 2022

[11] blhost User's Guide, Rev. 8, NXP Semiconductors, Nov 2020

[12] AN13023, Selecting and using cryptographic algorithms and protocols, Rev 1.0, NXP Semiconductors, November 2021.

[13] AN6259, Common Trust Provisioning Conceptional Overview, Rev 1.1, NXP Semiconductors, March 2021.

5 . 3 Standards

[14] J. Borghoff, et al, PRINCE - A Low-latency Block Cipher for Pervasive Computing Applications, Cryptology ePrint Archive, Report 2012/529.

6 Legal information

6.1 Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

6.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

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.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Suitability for use in automotive applications — This NXP product has been qualified for use in automotive applications. If this product is used by customer in the development of, or for incorporation into, products or services (a) used in safety critical applications or (b) in which failure could lead to death, personal injury, or severe physical or environmental damage (such products and services hereinafter referred to as "Critical Applications"), then customer makes the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, safety, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. As such, customer assumes all risk related to use of any products in Critical Applications and NXP and its suppliers shall not be liable for any such use by customer. Accordingly, customer will indemnify and hold NXP harmless from any claims, liabilities, damages and associated costs and expenses (including attorneys' fees) that NXP may incur related to customer's incorporation of any product in a Critical Application.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. 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.

6.3 Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Tables

Figures

Contents

Date of release: 12 October 2023