TA101 TrustFLEX TrustAnchor Data Sheet

TA101-TFLXTLS

Product Page Links

Description

The Microchip Technology Inc. Trust Anchor security device TA101-TFLXTLS is designed for use in industrial, automotive, data center, networking equipment, medical and commercial systems. It supports code authentication (secure boot), Message Authentication Code (MAC) generation, MACsec, trusted firmware updates, and multiple key management protocols, including Transport Layer Security (TLS) and other root-of-trust operations. Typically, it serves as a secure key storage companion device for a host (MCU or MPU) on the same board.

The TA101-TFLXTLS is a pre-configured variant of the TA101 device family. As part of Microchip's generically configured, security-focused product line, this secure element is engineered to address common use cases in connected embedded systems and controlled ecosystem markets, while reducing the complexity and time required to implement security solutions.

This data sheet provides detailed information on the configuration and data elements relevant to the TA101-TFLXTLS, including clearly defined access policies for each data zone slot. Limited command and I/O operating details are included, as these devices are intended for use with the Trust Platform Development Suite (TPDS) tools. Additionally, the application section offers guidance on Microchip's hardware and software tools to support application development, along with links to the location of the tools.

Features

- Advanced Crypto Engine (ACE) for Execution of All Cryptography Commands
- Elliptic Curves Support (ECC)
  - P-256: ECDSA sign, verify, KeyGen and ECDH
  - P-384: ECDSA sign, verify, KeyGen and ECDH
  - P-521: ECDSA sign, verify, KeyGen and ECDH
  - Ed25519: EdDSA sign, verify and KeyGen, Supported modes: Pure, CTX, PH
  - X25519: KeyGen and ECDH
- RSA Support
  - 2048-bit, 4096-bit RSA-OAEP encrypt/decrypt
  - 2048-bit, 4096-bit RSA signature generation and verification
  - 2048-bit, 4096-bit RSA key generation and key derivation
- Symmetric Cryptography and Algorithm Support
  - AES Key Generation (16- or 32-byte keys)
  - AES-ECB encryption/decryption (128-bit, 256-bit); external API support for alternate ciphers (CBC, CCM, Counter mode, etc.)
  - AES-CMAC (128-bit, 256-bit) calculation and validation
  - Authenticated Encryption with Associated Data (AEAD) using AES-GCM (128-bit or 256-bit, single and split modes)

© 2026 Microchip Technology Inc. and its subsidiaries

  - SHA-256, SHA-384, SHA-512 and HMAC-SHA256, HMAC-SHA384 and HMAC-SHA512 digest calculation
- ECDH Key Management Capability with Integrated KDF in Counter Mode Using HMAC-SHA256, HMAC-SHA384 or HMAC-SHA512
- Multiple Key Derivation Functions (KDF) Supported
  - PRF, HKDF, SP800-108 KDF and SHA-256 one-step KDF
  - TLS V1.2: Full session establishment support, including PRF KDF with host SW
  - TLS V1.3: Full-session establishment support, including HKDF with host SW
- Certifications
  - NIST SP800-90 A/B/C Random Number Generator (RNG) (NIST Certified - CAVP DRBG, ESV NRBG)
  - NIST Compliance mode for FIPS 140-3 security enforcment; supports formal NIST certification
  - Vulnerability Assessment Rating: JIL High, AVA_VAN.5
- I/O Options for Security Commands:
  - 1 MHz standard I^2C interface
  - 16 MHz SPI interface modes 0 and 3
- Package Options:
  - 8-lead SOIC
- Voltage Supply Range: 2.7V to 5.5V
- Industrial Grade Temperature Range: -40°C to +125°C Ambient Operating Range

The TA101 device has been assessed by a NIST-accredited laboratory and is compliant with FIPS 140-3 Level 2, with Physical Security Level 3 certification. For the latest FIPS certification status, please contact Microchip or visit the NIST CMVP FIPS certification status website.

Additionally, the TA101 device is engineered to withstand attacks with a "high" attack potential, as determined by the protection level of the private and secret keys. This assessment follows the vulnerability analysis scoring procedure outlined in the JIL-Application-of-Attack-Potential-to-Smartcards-v3.2.1 specification, referred to as “JIL High” throughout this document.

Use Cases

- Validation of firmware integrity via secure boot
- Secure firmware updates during runtime or following an over-the-air-update
- Authentication for WPC 1.3 and 2.x Qi high power transmitters
- Secure network authentication and session establishment using TLS
- Authentication for data center power supply
- Accessory card authentication
- Symmetric and Asymmetric authentication for ecosystem control
- Secure Ethernet link authentication and frame-level encryption using MACSec with symmetric keys

© 2026 Microchip Technology Inc. and its subsidiaries

Table of Contents

Description 1

Features 1

Use Cases 2

1. Pin Configuration 5
   - 1.1. SOIC-8 Pinout with SPI Interface 5
   - 1.2. SOIC-8 Pinout with I^2C Interface 5
2. Device Features 7
   - 2.1. Cryptographic Algorithm Specifications 7
   - 2.2. Secure Boot 8
   - 2.3. Authorization Sessions 8
   - 2.4. Certificate Storage and Processing 9
   - 2.5. Monotonic Counters 9
   - 2.6. Security Features 9
   - 2.7. Power and I/O 10
3. Device Memory 12
   - 3.1. Dedicated Data Memory 12
   - 3.2. Configuration Memory 12
   - 3.3. Shared Data Memory 12
   - 3.4. Volatile Data Memory 15
   - 3.5. Device Internal Handles 15
4. I/O Transactions 17
   - 4.1. I/O Interfaces 17
5. Device Commands 23
   - 5.1. Command Descriptions 23
6. Application Information 34
   - 6.1. External Components 34
   - 6.2. Onboarding Flow 34
   - 6.3. Development Tools 34
7. Electrical Characteristics 37
   - 7.1. Absolute Maximum Ratings 37
   - 7.2. Reliability 37
   - 7.3. DC Characteristics 37
   - 7.4. AC Characteristics 39
8. Package Marking Information 43
9. Package Drawings 44
   - 9.1. 8-Lead SOIC 44
10. Product Identification System 47

© 2026 Microchip Technology Inc. and its subsidiaries

11. Revision History....48

Microchip Information....49
Trademarks....49
Legal Notice....49
Microchip Devices Code Protection Feature....49

Product Page Links....50

© 2026 Microchip Technology Inc. and its subsidiaries

1. Pin Configuration

The TA101-TFLXTLS device is available in two package configurations, depending on the desired I/O interface:

- SPI-only interface in an 8-pin SOIC
- I^2C-only interface in an 8-pin SOIC

Each configuration offers different GPIO options based on the selected interface.

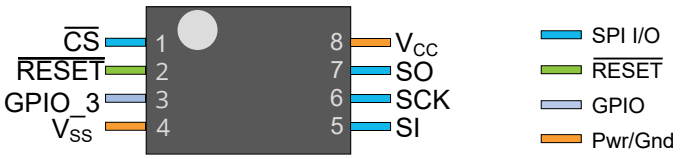
1.1. SOIC-8 Pinout with SPI Interface

The 8-pin SOIC SPI interface consists of the four SPI signals, a Reset signal and GPIO_3.

Table 1-1. 8-Pin SOIC SPI Pin Configuration

Pin Name	Pin Number	Function
$\overline{CS}$	1	Chip Select for SPI
$\overline{RESET}$	2	Reset Input, active low
GPIO_3	3	GPIO_3
V_{SS}	4	Ground
SI	5	SPI Serial Data Input
SCK	6	SPI Clock
SO	7	SPI Serial Data Output
V_{CC}	8	2.7V to 5.5V Power Supply

Figure 1-1. Pinout



1.2. SOIC-8 Pinout with I^2C Interface

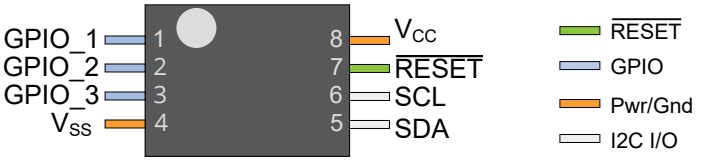
Pull-up resistors are required for proper operation of the I^2C bus, sized according to the board configuration and bus speed per the I^2C specification.

Table 1-2. 8-Pin SOIC I^2C Pin Configuration

Pin Name	Pin Number	Function
GPIO_1	1	GPIO_1
GPIO_2	2	GPIO_2
GPIO_3	3	GPIO_3
V_{SS}	4	Ground
SDA	5	I^2C Data
SCL	6	I^2C Clock
$\overline{RESET}$	7	Reset Input, active low
V_{CC}	8	2.7V to 5.5V Power Supply

© 2026 Microchip Technology Inc. and its subsidiaries

Figure 1-2. Pinout



© 2026 Microchip Technology Inc. and its subsidiaries

2. Device Features

2.1. Cryptographic Algorithm Specifications

The following outlines the reference documents governing the cryptographic algorithms implemented in the TA101-TFLXTLS device.

1. The Random Number Generator (RNG) function integrates both Non-deterministic Random Bit Generator (NRBG) and Deterministic Random Bit Generator (DRBG) components, in accordance with NIST SP800-90C, as specified in:
   a. DRBG: NIST Special Publication 800-90Ar1.
   b. NRBG: NIST Special Publication 800-90B.
   c. RNG: NIST Special Publication 800-90C.
2. Cryptographic digests using SHA-256, SHA-384 and SHA-512:
   a. NIST FIPS Publication 180-4 – Secure Hash Standard (SHS).
3. HMAC calculations utilize key sizes ranging from 16 to 128 bytes, based on SHA-256, SHA-384 or SHA-512 algorithms, as specified in:
   a. NIST FIPS Publication 198-1 – The Keyed-Hash Message Authentication Code (HMAC).
4. Authorization session establishment employs the HMAC-SHA256 using the counter key derivation function (KDF HMAC-Counter), as detailed in:
   a. NIST Special Publication 800-108 – Recommendation for Key Derivation Using Pseudorandom Functions.
   b. NIST Special Publication 800-56Cr2 – Recommendation for Key-Derivation Methods in Key Establishment Schemes (for SHA-256 one-step KDF).
5. Support for TLS 1.2 Key Derivation Function (PRF) is provided in accordance with:
   a. IETF RFC7627 – The Transport Layer Security (TLS) Protocol Version 1.2.
   b. NIST Special Publication 800-135 – Recommendation for Existing Application-Specific Key Derivation Functions.
6. Support for the TLS 1.3 Key Derivation Function (HKDF) is provided as specified in:
   a. IETF RFC5869 – HMAC-Based Extract-and-Expand Key Derivation Function (HKDF).
   b. IETF RFC8446 – The TLS Protocol Version 1.3.
7. Symmetric encryption/decryption using AES-128 and AES-256 implemented according to:
   a. NIST FIPS Publication 197 – Advanced Encryption Standard (AES), ECB mode.
8. The AES-CMAC algorithm is implemented as per:
   a. NIST Special Publication 800-38B – Recommendation for Block Cipher Modes of Operation: *The CMAC Mode for Authentication*.
9. AES encryption/decryption for authorization sessions utilize the GCM AEAD mode, as specified in:
   a. NIST Special Publication 800-38D – Recommendation for Block Cipher Modes of Operation: *Galois/Counter Mode (GCM) and GMAC*.
10. RSA key generation and signature operations (RSASSA-PKCS1-V1_5) follow PKCS#1 procedures, supporting key sizes of 2048, 3072 and 4096. The exponent is fixed at `0x10001`, except for the 3072-bit verification, which optionally supports e = 3 for HDCP protocol compatibility. Reference documents include:
    a. IETF RFC8017 – PKCS #1 RSA Cryptography Specifications Version 2.2.
    b. NIST FIPS Publication 186-5 – Digital Signature Standard (DSS).

© 2026 Microchip Technology Inc. and its subsidiaries

11. RSA key generation and signature operations (RSASSA-PSS) are performed according to PKCS#1 procedures, supporting key sizes of 2048, 3072, and 4096 bits, as specified in:
    a. IETF RFC8017 – PKCS #1 RSA Cryptography Specifications Version 2.2.
    b. NIST FIPS Publication 186-5 – Digital Signature Standard (DSS).
12. RSA encryption and decryption (RSAES-OAEP) with exponent `0x10001` are supported using the specified RSAES_OAEP PKCS#1 V2.2 scheme, with key sizes of 1024, 2048, 3072, and 4096 bits, as specified in:
    a. IETF RFC8017 – PKCS #1 RSA Cryptography Specifications Version 2.2.
13. Elliptic Curve Digital Signature Algorithm (ECDSA) key generation and signature operations using NIST curves P-224, P-256, P-384, and P-521 are performed in accordance with the following specifications. All curve keys can be generated using the internal RNG.
    a. ANSI X9.62-2005, *Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)*: www.ansi.org/.
    b. NIST FIPS Publication 186-5 – Digital Signature Standard (DSS).
14. Elliptic Curve Diffie-Hellman (ECDH) key agreement is supported, with all public and private keys treated as ephemeral and validated accordingly. The Ephemeral Unified C (2e, 0s, ECC CDH) scheme is implemented for P-224, P-256, P-384, and P-521 curves, as specified in:
    a. NIST Special Publication 800-56Ar3 – Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography.
15. Edwards Curve Support:
    a. Ed25519/EdDSA key generation and signature operations (128-bit security strength) are implemented as per IETF RFC8032 – Edwards-Curve Digital Signature Algorithm (EdDSA).
    b. ECDH X25519 key generation and key agreement (128-bit security strength) are supported as per IETF RFC7748 – Elliptic Curves for Security, Sections 5 and 6.
16. Qi Standard Point Expansion is implemented based on the compression protocols specified in:
    a. datatracker.ietf.org/doc/draft-jivsov-ecc-compact/00/ – *Compact representation of an elliptic curve point*.

2.2. Secure Boot

The TA101-TFLXTLS device is engineered to enhance the security features of the host by ensuring that only manufacturer-authorized operating code is executed. Typically, the boot code, which runs from the host's boot ROM, leverages the cryptographic functions of the TA101-TFLXTLS to verify the authorization of the code prior to execution. This verification process occurs each time the host is powered on. The device supports multiple validation approaches, such as Full-Asymmetric and Full-Stored modes.

Host code authentication can be configured in Full mode as follows: During each secure boot cycle, the entire code base within the host is verified. The duration of this process depends on factors such as the memory size, host clock speed, and communication interface bandwidth.

2.3. Authorization Sessions

Authorization sessions in the TA101-TFLXTLS device are designed to manage access to key and data elements, while also establishing an encrypted and integrity-protected communication channel between the TA101-TFLXTLS and an external entity. The host may serve as the endpoint for these sessions or, alternatively, function as a conduit. Authorization sessions are also effective in mitigating various types of attacks, including denial-of-service attempts.

A shared symmetric key is required to initiate an authorization session. The device supports up to two concurrent authorization sessions, provided sufficient memory is allocated. While an authorization session is active, it remains possible to send commands outside of the session.

© 2026 Microchip Technology Inc. and its subsidiaries

2.4. Certificate Storage and Processing

The TA101-TFLXTLS device supports the storage of a limited number of X.509v3 certificates within its internal memory. Certificates can be stored in one of the following formats:

- **Full:** Complete X.509 certificates are stored as unformatted data elements within the device
- **Extracted:** Only essential elements are stored, including the subject public key, usage attributes or restrictions, and an identity digest derived from the source certificate

2.5. Monotonic Counters

The TA101-TFLXTLS device features four monotonic counters, each with a maximum count of approximately 12.8 million. Counter #0 can be configured to increment automatically over time. The remaining three counters are typically incremented with each use of a key—such as for audit log signatures or message authentication codes (MACs)—or can be manually incremented by the user through the `Counter` command.

2.6. Security Features

The TA101-TFLXTLS device incorporates robust protection mechanisms against both active (invasive) and passive (noninvasive) attacks targeting certificates, private keys, and symmetric keys stored within the device. Dedicated hardware and firmware features are implemented to defend against environmental attacks (such as voltage, temperature, and frequency manipulation), emissions attacks, fault attacks, physical tampering, cloning, and other potential threats. Additionally, all internal memory used for private or symmetric keys and other sensitive data is encrypted.

2.6.1. Watchdog Timer

The TA101-TFLXTLS device includes an internal watchdog timer to ensure the command processor operates reliably. If the timer expires, the device automatically resets and transitions to the Active state after the designated wake delay. The command processor resets the timer upon receiving a valid command and while awaiting new commands. Additionally, the timer is reset whenever the device enters Sleep mode.

The minimum duration of the watchdog timer ($t_{WATCHDOG}$) is set to accommodate all commands, ensuring they complete within this interval under all environmental and processing conditions. Exceptions are made for Self_Test and RSA key generation operations, which are managed internally to prevent the watchdog timer from expiring and causing an unintended reset during their execution.

2.6.2. Environmental Tamper Detection

The TA101-TFLXTLS device is equipped with internal sensors that monitor voltage and temperature levels. If these parameters exceed defined limits—either above or below the acceptable range—the device will automatically reset and transition to the Active state following the wake delay.

2.6.3. Command Processor

The command processor includes the Advanced Crypto Engine (ACE), which provides the controls required to configure and provision the TA101-TFLXTLS device. The ACE supports execution of all commands specified in this data sheet. During initial configuration and provisioning, the command processor enables access to the EEPROM configuration memory, allowing users to set desired features and lock specific zones as needed.

In normal operation, the command processor manages the complete configuration required to execute commands and facilitates reading responses. Device status and control registers for the ACE are also accessed through the command processor.

All commands and responses sent to and read from the command processor are available via either the I^2C or SPI interface.

© 2026 Microchip Technology Inc. and its subsidiaries

2.7. Power and I/O

2.7.1. I/O Logic

The I/O logic block serves as the primary interface for connecting the TA101-TFLXTLS device to external microcontrollers and microprocessors. Depending on the package option, either or both the I^2C and SPI interfaces may be available. All device features and cryptographic functions can be accessed through either interface. The choice of interface should be based on system requirements, such as the host I/O configuration and performance needs. Command execution speeds are consistent across interfaces, with the only variation being the time required to transfer command data and responses.

Internal pull-ups on I/O signals can be optionally enabled during initial device configuration. This setting is not user-configurable after the device has been set up and becomes permanent once configuration is complete.

2.7.2. GPIO Circuitry

The TA101-TFLXTLS device features three General Purpose Input/Output (GPIO) signals. The functionality and default states of these signals are determined during the initial configuration phase and cannot be modified once the configuration zone is locked.

GPIO signals can be set as either inputs or outputs, but not as bidirectional. Optional pull-up resistors may be enabled during manufacturing configuration; their status can be verified using the Info command.

The state of each GPIO signal can be read or updated using the Read and Write commands.

2.7.3. Power Management Circuitry

The TA101-TFLXTLS device features power management circuitry that enables operation in either Sleep or Active states. The Power command can be used to place the device into Sleep mode or to reset it. The device automatically wakes upon receiving a command or during initial power-up. The TA101-TFLXTLS operates in two distinct power modes:

Active Mode:

In Active mode, the device is ready to accept commands on any supported bus interface. Upon power-up (when V_{CC} rises above V_{UVD}) or when the $\overline{RESET}$ pin is deasserted, the device enters Active mode by default. During the $t_{PU_POWERON}$ interval, I/O pins are ignored—no I^2C transactions are acknowledged, and SPI CS assertions have no effect.

Sleep Mode:

In Sleep mode, the device's power consumption is significantly reduced, but there is a defined wake-up interval (t_{PU_SLEEP}). When waking from Sleep, all SRAM contents and volatile registers are erased, except for the least significant byte (LSB) of counter #0 in the monotonic counter block. The device automatically enters Sleep mode when the idle timer expires. If the $\overline{RESET}$ pin is asserted, V_{CC} falls below V_{UVD}, or V_{CC} exceeds its maximum, the device halts processing and enters Internal Reset mode, resulting in the loss of all SRAM contents.

Wake Events:

While in Sleep mode, the device will wake upon any I/O transaction. During the wake-up interval (t_{PU_SLEEP}), the host must not send instructions or commands, as they may not be executed.

- For SPI connections, asserting the $\overline{CS}$ pin initiates the wake-up sequence, which must be followed by at least one SPI clock (SCK) cycle. Any data sent during this cycle will be ignored.
- For I^2C connections, the device will only wake if the address matches the value stored in the Config:I2C_Address byte. The device will NACK the address and ignore all subsequent bytes until t_{PU_SLEEP} has elapsed.

© 2026 Microchip Technology Inc. and its subsidiaries

After issuing a Power (Sleep) command, the host must wait for the t_{SLEEP_WAKE} interval before allowing a transition on GPIO_3. Any transitions within this window may be ignored and may not trigger a wake sequence.

© 2026 Microchip Technology Inc. and its subsidiaries

3. Device Memory

The TA101-TFLXTLS device features several distinct memory areas, each serving a specific purpose:

3.1. Dedicated Data Memory

This read-only memory stores the device's unique 8-byte serial number and software revision information. These details are always accessible via the Info command.

3.2. Configuration Memory

The configuration memory is utilized to enable various features and functions tailored to specific application requirements. This memory is primarily pre-configured to accommodate different application scenarios. The following table outlines the pre-configured settings:

Table 3-1. Configuration Memory

Name	Description
Self-Test	All device cryptographic algorithms are set to perform self-tests on demand via the self-test command. If a test fails, the result is automatically cleared when the test is rerun and passes, or upon power-up or $\overline{\text{RESET}}$ pin assertion.
I^2C Address	The device's default 7-bit I^2C address is 0x16, which corresponds to 0x2C for write operations and 0x2D for read operations.
Chip and Compliance Options	The device is configured to reset self-test failures through the self-test command, power cycling, or $\overline{\text{RESET}}$ pin assertion. The compliance option is disabled by default on PROTO units.
GPIO Configuration	GPIO1 is configured as an input. GPIO2 is set to generate a pulse upon command completion. GPIO3 is configured as an output with a default low state and can be updated only during an authenticated session using Auth Key (0x80FC).
Update Options	The device supports updates via patches that are controlled and released by Microchip. Updates are allowed only during an authenticated session with the Auth Key (0x80FC). Patch downgrades are supported, and the loaded image is fully verified at power-up before any commands are executed.
Soft Reboot	The device supports a soft reboot through the Power command.
Secure Boot	Secure Boot is enabled by default in Full Stored mode with a digest size of 32 bytes (SHA256). Updates to the Secure Boot digest are permitted only during an authenticated session using the Auth Key.

3.3. Shared Data Memory

The shared data memory area is designed for storing keys, secrets, certificates and other data. The TA101-TFLXTLS device is pre-configured to allow users to load assets and perform cryptographic operations as needed. The table below summarizes the pre-configured handles and their attributes for use in cryptographic operations:

Table 3-2. Shared Data Memory

Data Element	Handle ID	Description
ECC P256 Private Key	0x8000	Stores an ECC P256 private key. Supports signing of both internally and externally generated digests, as well as key agreement operations. The key is always available for use but cannot be read or written.
ECC P384 Private Key	0x8001	Stores an ECC P384 private key. Supports signing and key agreement operations. The key is always available for use but cannot be read or written.
ECC P521 Private Key	0x8002	Stores an ECC P521 private key. Supports signing and key agreement operations. The key is always available for use but cannot be read or written.
Ed25519 Private Key	0x8003	Stores an Ed25519 private key. Supports signing of digests but does not support key agreement operations. The key is always available for use but cannot be read or written.

© 2026 Microchip Technology Inc. and its subsidiaries

Table 3-2. Shared Data Memory (continued)

Data Element	Handle ID	Description
RSA 2048 Private Key	0x8004	Stores an RSA 2048 private key with SSA_PSS algorithm support. Supports signing of digests. The key is always available for use but cannot be read or written.
RSA 4096 Private Key	0x8005	Stores an RSA 4096 private key with SSA_PSS algorithm support. Supports signing of digests. The key is always available for use but cannot be read or written.
ECC P256 Public Key	0x8006	Stores an ECC P256 public key. Always available for use and can be read at any time. Updates are permitted only during an authenticated session using the Auth Key (0x80FC). The key can be locked to prevent further changes.
ECC P384 Public Key	0x8007	Stores an ECC P384 public key. Always available for use and can be read at any time. Updates are permitted only during an authenticated session using the Auth Key. The key can be locked to prevent further changes.
ECC P521 Public Key	0x8008	Stores an ECC P521 public key. Always available for use and can be read at any time. Updates are permitted only during an authenticated session using the Auth Key. The key can be locked to prevent further changes.
Ed25519 Public Key	0x8019	Stores an Ed25519 public key. Always available for use and can be read at any time. Updates are permitted only during an authenticated session using the Auth Key. The key can be locked to prevent further changes.
RSA 2048 Public Key	0x8010	Stores an RSA 2048 public key. Always available for use and can be read at any time. Updates are permitted only during an authenticated session using the Auth Key. The key can be locked to prevent further changes.
RSA 4096 Public Key	0x8011	Stores an RSA 4096 public key. Always available for use and can be read at any time. Updates are permitted only during an authenticated session using the Auth Key. The key can be locked to prevent further changes.
HMAC256 Symmetric Key	0x8012	Stores a 256-bit HMAC symmetric key. Usable for all symmetric key operations and always available, but cannot be read. Updates are allowed only during an authenticated session with the Auth Key. The key can be locked to prevent further changes.
HMAC512 Symmetric Key	0x8013	Stores a 512-bit HMAC symmetric key. Usable for all symmetric key operations and always available, but cannot be read. Updates are allowed only during an authenticated session with the Auth Key. The key can be locked to prevent further changes.
AES128 Symmetric Key	0x8014	Stores a 128-bit AES symmetric key. Usable for all symmetric key operations and always available, but cannot be read. Updates are allowed only during an authenticated session with the Auth Key. The key can be locked to prevent further changes.
AES256 Symmetric Key	0x8015	Stores a 256-bit AES symmetric key. Usable for all symmetric key operations and always available, but cannot be read. Updates are allowed only during an authenticated session with the Auth Key. The key can be locked to prevent further changes.
User Data1	0x8016	Stores up to 512 bytes of user data. Data can be used, read, and updated at any time without restrictions. The element can be locked to prevent further modifications.
User Data2	0x8017	Stores up to 512 bytes of user data. Data is always available for use, but read and update operations are permitted only during an authenticated session with the Auth Key. The element can be locked to prevent further modifications.
Signer Certificate	0x8018	Stores up to 1500 bytes of certificate data, intended for the application's signer certificate. Can be used, read, and updated at any time without restrictions. The element can be locked to prevent further modifications.

© 2026 Microchip Technology Inc. and its subsidiaries

Table 3-2. Shared Data Memory (continued)

Data Element	Handle ID	Description
Device Certificate	0x8019	Stores up to 1500 bytes of certificate data, intended for the application's device certificate. Can be used, read, and updated at any time without restrictions. The element can be locked to prevent further modifications.
Secure Boot Certificate	0x8020	Stores a Secure Boot certificate containing the public key for Secure Boot. The certificate must include the Secure Boot OID as specified by Microchip and be signed with the corresponding Secure Boot Root Private Key. The certificate can be used and read at any time; updates are permitted only during an authenticated session with the Auth Key.
ECC P384 Secure Boot Root Public Key	0x8021	Stores an ECC P384 public key as the Secure Boot Root. Used to authenticate Secure Boot certificates during public key rotation. Can be used and read at any time; write operations are allowed only before the setup is locked.
ECC P256 Root Public Key	0x8022	Stores an ECC P256 public key as the Root Public Key. Can be used and read at any time; write operations are allowed only before the setup is locked.
ECC P521 Root Public Key	0x8023	Stores an ECC P521 public key as the Root Public Key. Can be used and read at any time; write operations are allowed only before the setup is locked.
ECC P256 Secure Boot Root Public Key	0x8024	Stores an ECC P256 public key as the Secure Boot Root. Used to authenticate Secure Boot certificates during public key rotation. Can be used and read at any time; write operations are allowed only before the setup is locked.
ECC P521 Secure Boot Root Public Key	0x8025	Stores an ECC P521 public key as the Secure Boot Root. Used to authenticate Secure Boot certificates during public key rotation. Can be used and read at any time; write operations are allowed only before the setup is locked.
AES128 GCM Symmetric Key1	0x8026	Stores a 128-bit AES symmetric key. Usable for all symmetric key operations and always available, but cannot be read. Updates are allowed only during an authenticated session with the Auth Key. The key can be locked to prevent further changes.
AES128 GCM Symmetric Key2	0x8027	Stores a 128-bit AES symmetric key. Usable for all symmetric key operations and always available, but cannot be read. Updates are allowed only during an authenticated session with the Auth Key. The key can be locked to prevent further changes.
AES128 GCM Symmetric Key3	0x8028	Stores a 128-bit AES symmetric key. Usable for all symmetric key operations and always available, but cannot be read. Updates are allowed only during an authenticated session with the Auth Key. The key can be locked to prevent further changes.
AES256 GCM Symmetric Key1	0x8029	Stores a 256-bit AES symmetric key. Usable for all symmetric key operations and always available, but cannot be read. Updates are allowed only during an authenticated session with the Auth Key. The key can be locked to prevent further changes.
AES256 GCM Symmetric Key2	0x8030	Stores a 256-bit AES symmetric key. Usable for all symmetric key operations and always available, but cannot be read. Updates are allowed only during an authenticated session with the Auth Key. The key can be locked to prevent further changes.
AES256 GCM Symmetric Key3	0x8031	Stores a 256-bit AES symmetric key. Usable for all symmetric key operations and always available, but cannot be read. Updates are allowed only during an authenticated session with the Auth Key. The key can be locked to prevent further changes.
Auth Key	0x80FC	Stores a 256-bit AES symmetric key for authenticated sessions. The key can be used and written at any time but cannot be read. It can be locked to prevent further modifications.

© 2026 Microchip Technology Inc. and its subsidiaries

Table 3-2. Shared Data Memory (continued)

Data Element	Handle ID	Description
Attestation Key	0x80D0	Stores an ECC P521 private key for attestation. Supports signing of internally generated digests only; does not support key agreement operations. The key is always available for use but cannot be read or written.
Template	0x80D1	Stores template data required for internal sign operations. Configured to use the Attestation Key and InternalSign Random Data during signing. Data is always available for use and can be read at any time, but cannot be updated.
InternalSign Random Data	0x80D2	Stores random data for use during internal sign operations with the Attestation Key. User-provided random data must be written before use. Data can be used, read, and written at any time.

3.4. Volatile Data Memory

The TA101-TFLXTLS device utilizes SRAM memory to store various data elements. All volatile data memory is cleared when the device enters Sleep mode, the $\overline{\text{RESET}}$ pin is asserted or power is removed.

Table 3-3. Volatile Data Memory

Qty.	Name	Description
4	Volatile Registers	Volatile registers that can store keys, state or other information
2	SHA_Context	Internal SHA-256 state used by the SHA command
2	Auth_Session	Nonces, counters and keys associated with authorization sessions

Volatile Registers

The device features four general purpose volatile registers (Vol_Reg[0], Vol_Reg[1], Vol_Reg[2], and Vol_Reg[3]) that are used to store states such as public/private keys, digests, or other data required to complete command sequences. These registers can be written using the Write command, but cannot be read using the Read command. Each register provides sufficient memory to store any public, private, or symmetric key, or any data up to 512 bytes in size, including symmetric keys, ECC keys, and RSA keys up to 4096 bits.

3.5. Device Internal Handles

The following table summarizes the internal handles used by the device, including their storage locations, access permissions, and descriptions:

Table 3-4. Device Internal Handles

Name	Value	Storage	R/W	Description
SHA_Context	4000-4001	SRAM	R/W	SHA context stored in SRAM. The least significant byte (LSB) represents the session ID.
Auth_Session	4100-4101	SRAM	—	Authorization session data stored in SRAM.
GPIO	4300-4302	SRAM	R/W	Read or write the current state of the GPIO pins. The LSB indicates the pin: 0x4300 = GPIO_1, 0x4301 = GPIO_2, 0x4302 = GPIO_3.
Volatile Register	4400-4403	SRAM	W	Volatile register locations within SRAM. These cannot be read using the Read command.
Input Buffer	4800	SRAM	—	Input buffer, used as permitted by specific commands.
Output Buffer	4801	SRAM	—	Output buffer, used as permitted by specific commands.
Configuration	C000	Nonvolatile Memory	R/W	Configuration memory.

© 2026 Microchip Technology Inc. and its subsidiaries

Table 3-4. Device Internal Handles (continued)

Name	Value	Storage	R/W	Description
Counter	D100-D103	Nonvolatile Memory	—	Monotonic counter. Cannot be directly read or written; access is provided via the Counter command.
I/O Protection	80FC	Nonvolatile Memory	—	Authorization key used to authenticate and/or encrypt data on the bus with the connected host.

© 2026 Microchip Technology Inc. and its subsidiaries

4. I/O Transactions

The transaction type is determined by the word address from the I²C port or the instruction code from the SPI port. The encoding of both is the same and is documented in this section. There are three types of I/O transactions:

Table 4-1. I/O Transaction Types

Transaction	Description
Command	Transfers command or response packets between the TA101-TFLXTLS command processor and the host. Each packet consists of a sequence of these transactions and includes explicit length and CRC fields.
Status	Returns the status of the main command processor to the host processor in a 1-byte transaction. These transactions do not include a CRC field.
Control	Provides hardware control for the command processor in a 1-byte transaction. These transactions do not include a CRC field.

Any write transaction that does not contain data will be ignored by the TA101-TFLXTLS device, without setting error bits or interrupting instruction or command flow. Similarly, any invalid transaction-type byte not listed below will be ignored, along with any subsequent data, without setting error bits or interrupting operation.

Table 4-2. Transaction Type Table

Name	Kind	Value	Description
RD_CSR	Status	0011 0000	Reads the Command Processor Status Register
WR_CCR	Control	0010 0000	Writes to the Command Control Register
RD_CMD	Command	0001 0000	Reads the response from the Command Processor response buffer. The first byte is the MSB of the length.
WR_CMD	Command	0000 0000	Writes commands or data to the Command Processor input buffer. The first byte is the MSB of the length.

4.1. I/O Interfaces

The TA101-TFLXTLS device supports both SPI and I²C ports, enabling compatibility with a wide range of host MCUs and MPUs. The availability of these ports depends on the specific part number and package; the configuration is fixed during manufacturing and cannot be modified in the field.

For devices equipped with both SPI and I²C ports, either interface can be used to send instructions or commands. However, only one port may be active at a time—if a transaction is in progress on one port, the other port is unavailable until the transaction completes.

It is not possible to interrupt the execution of a command by the main command processor, except during RSA key generation. Asserting the $\overline{\text{RESET}}$ pin will immediately halt all operations within the device.

4.1.1. I²C Interface

The TA101-TFLXTLS device utilizes the SDA and SCL pins for transferring commands, data and status information. The I²C interface is protocol-compatible with the Microchip AT24C16 Serial EEPROM, operating at 1 MHz, though there are several differences, including a fully programmable device on the TA101-TFLXTLS.

I²C Interface Terminology

Host: The host generates the serial clock and manages data flow on the I²C bus to one or more client devices.

Client: The TA101-TFLXTLS device always functions as a client device on the I²C bus and cannot assume control of the bus.

© 2026 Microchip Technology Inc. and its subsidiaries

Device Address:	The 7-bit I^2C address is used to address the client device. This address is included in the first byte sent for each write or read transaction.
Open-Drain:	The TA101-TFLXTLS device features an open-drain output buffer. When reading data, the buffer actively pulls the bus low; when idle, the bus is passively pulled high by a pull-up resistor.

Remember: The I^2C standard uses the terminology “Master” and “Slave”. The equivalent Microchip terminology used in this document is “Host” and “Client”, respectively.

The SDA pin must be connected to a pull-up resistor, as the TA101-TFLXTLS device uses only an open-drain driver in I^2C mode. The host may use either open-drain or totem pole outputs; if totem pole is used, it must be tri-stated when the TA101-TFLXTLS device is transmitting data. The SCL pin serves as an input and must be driven by an external device. See External Components for the default I/O configuration.

The TA101-TFLXTLS I^2C interface does not support clock stretching or reserved addresses, including the general call function. Consequently, software reset is not supported via I^2C; the $\overline{\text{RESET}}$ pin must be used for this purpose.

4.1.1.1. I^2C Interface Start-up

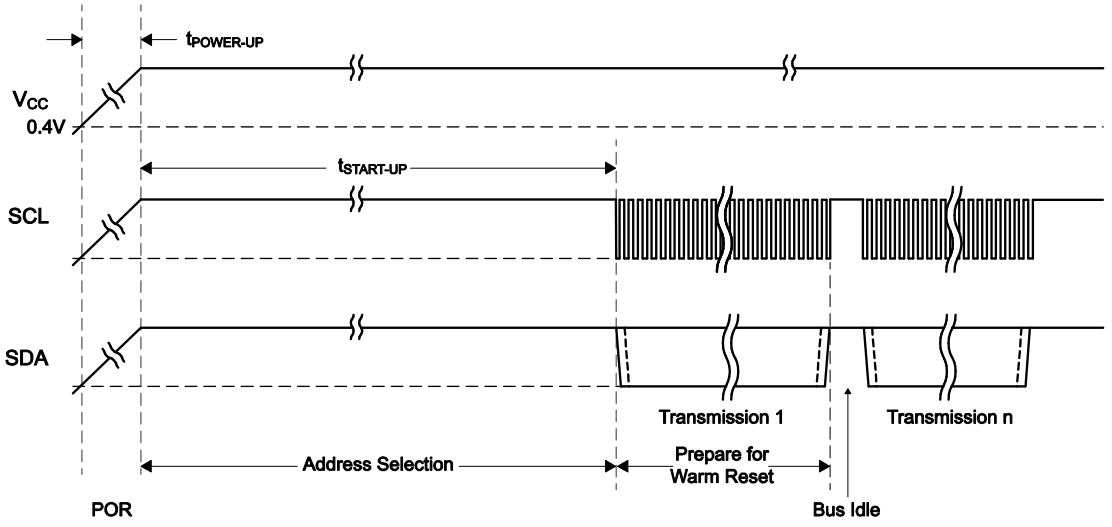
The I^2C communication interface can be started-up either by supplying power to the V_{CC} line or by issuing a wake event after power is supplied (refer to the Wake Event section for details).

Upon wake from sleep, power-up or chip reset, both the I^2C address register and output buffer are cleared. If a device address with the R/$\overline{\text{W}}$ bit set to ‘1’ (Read) is sent to the TA101-TFLXTLS device before a location or action is selected using the R/$\overline{\text{W}}$ bit set to ‘0’ (Write), the device will respond with 0xFF.

After power is applied, the TA101-TFLXTLS device requires a specific amount of time to power up and read configuration data from nonvolatile memory to initialize its internal circuitry. During this initialization period, the device is not ready to accept commands or instructions. The first data transmission should begin no earlier than $T_{PU.POWERUP}$ after the V_{CC} line is asserted. Any signals or transitions on the SDA and SCL lines during this interval will be ignored by the device.

© 2026 Microchip Technology Inc. and its subsidiaries

Figure 4-1. TA101-TFLXTLS I²C Power-On Timing



4.1.1.2. I²C Protocol

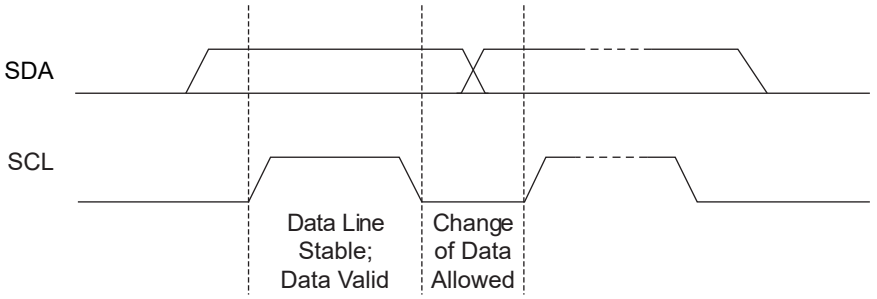
The basic I²C protocol is described in https://www.i2c-bus.org/specification/. Additional protocol details are provided below.

4.1.1.2.1. Bus Conditions

When the device is awake, it honors the following bus conditions:

- **DATA Zero:** If the SDA line is low and stable while SCL transitions from low to high and back to low, a '0' bit is transferred. The SDA line may change state while SCL is low.
- **DATA One:** If the SDA line is high and stable while SCL transitions from low to high and back to low, a '1' bit is transferred. The SDA line may change state while SCL is low.

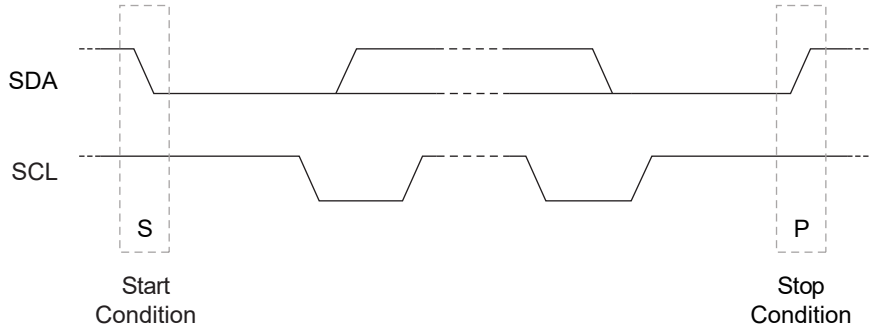
Figure 4-2. Data Bit Transfer on the I²C Interface



- **Start Condition:** A high-to-low transition on SDA with SCL high signals a Start condition, which must precede all commands
- **Stop Condition:** A low-to-high transition on SDA while SCL high signals a Stop condition. Upon receiving this condition, the device ends the current I/O transaction. If sufficient bytes have been received to execute a command, the device transitions to the busy state and begins execution. It is recommended to send a Stop condition at the end of every transaction.

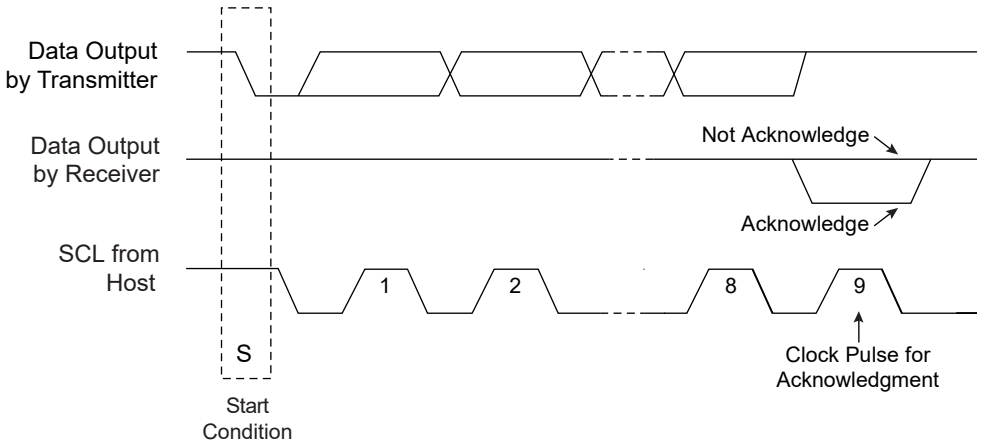
Figure 4-3. Start and Stop Conditions on the I²C Interface

© 2026 Microchip Technology Inc. and its subsidiaries



- **Acknowledge (ACK):** On the ninth clock cycle, after each address or data byte is transferred, the receiver pulls SDA low to acknowledge successful reception
- **Not Acknowledge (NOT ACK):** Alternatively, on the ninth clock cycle, after every address or data byte is transferred, the receiver may leave SDA high to indicate a reception issue or to signal the end of the transaction

Figure 4-4. NOT ACK and ACK Conditions on the I²C Interface



Multiple TA101-TFLXTLS devices can share the same I²C bus if each device is programmed with a unique I²C address in the configuration zone. Since all seven bits of the device address are programmable, the TA101-TFLXTLS device can coexist on the I²C bus with any other I²C device, including Serial EEPROM memory.

4.1.1.2.2. Client Selection

After a Start condition is detected on the I²C bus, the first byte sent to the TA101-TFLXTLS device is the device address. This byte consists of a 7-bit device address and a 1-bit R/$\overline{W}$ field.

As an I²C client, the TA101-TFLXTLS device is selected when the incoming device address matches the "I2C_Address" value stored in its configuration memory. The R/$\overline{W}$ field within the device address (LSb) controls whether the operation is a read or write; this bit is not compared to the value in the Config:I2C_Address.

Table 4-3. TA101-TFLXTLS I²C Device Address and R/$\overline{W}$ Bit

A6	A5	A4	A3	A2	A1	A0	R/$\overline{W}$
I2C_Addr:7	I2C_Addr:6	I2C_Addr:5	I2C_Addr:4	I2C_Addr:3	I2C_Addr:2	I2C_Addr:1	R/$\overline{W}$

By default, devices shipped from Microchip are configured with a 7-bit I²C address of 0x16. Therefore, the first byte for a write transaction 0x2C, and the first byte for a read transaction is 0x2D.

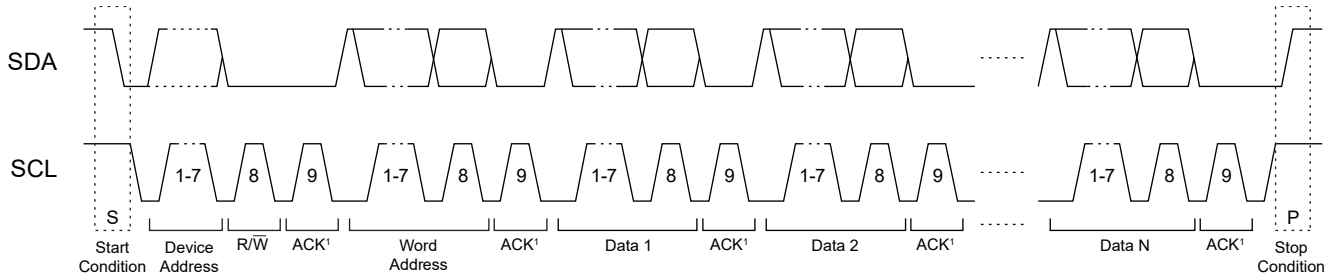
© 2026 Microchip Technology Inc. and its subsidiaries

4.1.1.2.3. I²C Transmission to the TA101-TFLXTLS

Data transmission from the system to the TA101-TFLXTLS device is summarized in the table below. The order of transmission is as follows:

1. Start condition.
2. Device Address byte.
3. Word Address byte.
4. Optional Data bytes (1 through N).
5. Stop condition.

Figure 4-5. Normal I²C Transmission to the TA101-TFLXTLS



Note: During the ACK response, the TA101-TFLXTLS device drives the SDA line low.

According to the I²C standard specification (www.i2c-bus.org/specification/), bits on the I²C bus are labeled in the diagram above such that bit 1 is the first bit transmitted, bit 8 is the last bit transmitted, and bit 9 is the ACK bit received by the transmitter.

This should not be confused with the bit order when a byte integer is transmitted on the bus, where the first bit transmitted is the MSb and the last bit transmitted is the LSb. Throughout this document, bit numbers correspond to their significance: bit 7 is the MSb and bit 0 is the LSb.

The table below labels the bytes of the I/O transaction, with the "I²C Name" column referencing the terminology used in the AT24C16 data sheet.

Table 4-4. I²C Transmission to the TA101-TFLXTLS

Name	I²C Name	Description
Device Address	Device Address	This byte selects the target device on the I²C bus. The TA101-TFLXTLS device is selected if the first 7 bits match the I2C_Address value stored in the configuration zone. The last bit is the I²C R/$\overline{W}$ bit, which must be set to '0' to indicate a write operation (subsequent bytes are sent from the host to the client).
Word Address	Word Address	This byte specifies the destination within the TA101-TFLXTLS device for the following data bytes.
Data	Data1, N	These bytes contain the input data required for the requested operation.

To ensure proper operation, the system must send a Stop condition after the last data byte. Failure to so may result in loss of synchronization with the TA101-TFLXTLS device.

4.1.1.2.4. I²C Reception from TA101-TFLXTLS

Receiving data from the TA101-TFLXTLS device follows the standard I²C protocol. The read instruction does not include a separate address phase; instead, the address is set during the initial write sequence. The process is as follows:

1. Start condition.
2. Device Address byte (R/$\overline{W}$ = 0).
3. Word Address byte.
4. Optional Stop condition.

© 2026 Microchip Technology Inc. and its subsidiaries

5. Start condition.
6. Device Address byte (R/$\overline{W}$ = 1).
7. Data bytes.
8. Stop condition.

Table 4-5. I^2C Reception from the TA101-TFLXTLS

Name	I^2C Name	Direction	Description
Device Address	Device Address	To client	This byte selects the target device on the I^2C bus. The TA101-TFLXTLS device is selected if the first 7 bits of this byte match the I2C_Address value stored in the configuration zone. The last bit is the I^2C R/$\overline{W}$ bit, which must be set to '1' to indicate a read operation.
Data	Data1, N	To host	The data returned from the specified address or register.

4.1.2. SPI Interface

The TA101-TFLXTLS device features a client SPI controller for transferring commands and responses to and from the main command processor. The SPI port is compatible with microcontrollers operating in SPI Mode 0 or Mode 3. Input/output and read/write operations are similar to those used with Microchip SPI serial nonvolatile memories, such as the AT25M01; however, the instruction codes and memory addresses differ between these devices. For information on default I/O configuration, refer to the External Components section.

4.1.2.1. SPI Terminology

Host: The host generates the serial clock.

Client: The TA101-TFLXTLS device always operates as a client on the SPI bus and cannot assume control of the bus.

Chip Select: The TA101-TFLXTLS device is selected when the $\overline{CS}$ pin is low. When not selected, transitions on the Serial Input (SI) pin are ignored, and the Serial Output (SO) pin remains in a high-impedance state.

Instruction Code: After the device is selected ($\overline{CS}$ goes low), the first byte received by the TA101-TFLXTLS is the instruction code, which specifies the operation to be performed.

Remember: The SPI standard uses the terminology "Master" and "Slave". The equivalent Microchip terminology used in this document is "Host" and "Client", respectively.

© 2026 Microchip Technology Inc. and its subsidiaries

5. Device Commands

Commands are sent to the main command processor using a standardized packet format. Once the TA101-TFLXTLS device receives the final byte of the CRC, it will begin executing the command and will disregard any additional bytes sent thereafter.

Table 5-1. Command Packet Formatting

Field Name	Bytes	Description
Length	2	Specifies the total number of bytes in the command, including the "Length" field and the "CRC" field (2 bytes).
Opcode	1	Indicates the command to be executed by the TA101-TFLXTLS.
Param1	1	The first parameter for the command. If used, this parameter will be described in detail in the relevant command documentation and often serves as a mode modifier.
Param2	4	The second parameter for the command. If applicable, this parameter will be described in the relevant command documentation. If not used for a specific command, it must be sent to the TA101-TFLXTLS as 0x00 00 00 00.
Data	0-1024	Provides additional information for the command, with a maximum length of 1024 bytes. If not required for a particular command, this field is either omitted from the input parameter table or specified as 0 bytes.
CRC	2	Contains the CRC value for verification of the Length, Opcode, parameters, and Data fields.

Command response packets from the TA101-TFLXTLS device also follow a standardized format:

Table 5-2. Command Response Packets

Field Name	Bytes	Description
Length	2	Total number of bytes in the response, including the "Length" field and the "CRC" field (2 bytes).
Resp_Code	1	Response code indicating the result of the command; '0x00' signifies success.
Data	0-1024	Additional response data generated by the command.
CRC	2	CRC16 value used to verify the Length, Resp_Code, and Data fields.

The minimum length for a command packet is 10 bytes. If the "Length" field in the command packet is less than 10 bytes, the entire packet will be rejected and the ERR bit in the CSR will be set.

Command packets may be transmitted to the TA101-TFLXTLS device over one or more I/O transactions on the bus. Bytes are accumulated sequentially in the input buffer until the final CRC byte is received. Depending on the selected I/O interface, the host may restart the address counter for the input buffer to resend any portion of the command packet as needed.

All properly formatted commands will generate an output packet in the response buffer. Response packets from the TA101-TFLXTLS device will always be at least 5 bytes in length. Not all I/O transactions include the "Length" and "CRC" fields; only the first transaction contains the "Length" field, and only the final transaction contains the "CRC" field.

5.1. Command Descriptions

The following sections provide a brief overview of all commands supported by the TA101-TFLXTLS device.

5.1.1. AES

The `AES` command computes the AES-ECB or AES-GCM operations, supporting both encryption and decryption. The AES function operates in two phases: the first phase loads a key into the internal AES engine within the command processor, and the second phase executes the encryption

© 2026 Microchip Technology Inc. and its subsidiaries

or decryption operation. The command supports both 128-bit and 256-bit AES encryption and decryption using appropriately sized keys.

Keys may be loaded either as individual keys or as part of a key group. If keys are loaded during an authorization session, they must be used within the same session. When the Chip Options Compliance bit is set, AES-ECB encryption and decryption are not permitted; only AES-GCM operations are supported.

Table 5-3. AES Mode Table

Command	Mode	Description
AES	Key Load	Loads the AES Key and, optionally, the IV
	ECB Modes	Encrypts and decrypts messages
	GCM Modes	Encrypts and decrypts entire message. This mode can only be used if the sum of auth_length and message_length is ≤ 996 bytes.
	GCM Large Message Mode	Encrypts and decrypts large messages using Start, Update and Final phases

5.1.2. Authorize

The `Authorize` command is used to initiate and execute an authorization session. This includes the generation of the nonce used to create a shared session key, the creation of a new authentication session, the execution of the commands nested within the authorization session and the termination of the authorization session.

Authorization sessions continue to persist through a specified number of commands or can be explicitly terminated. Authorization sessions are always terminated when the device goes to sleep or when the power is removed. If an attempt is made to start an authorization session using an authorization session handle that is already valid, an error will be returned.

Table 5-4. Authorize Mode Table

Command	Mode	Description
Authorize	Generate Nonce	Generates a nonce for a remote device
	Setup Session	Initiates a new authentication session
	Nested Command - HMAC	Executes the nested (authorized) command using HMAC
	Nested Command - GCM	Executes a nested (authorized) command using GCM
	Terminate	Terminates the authorization session

Authorization Process

The authorization process consists of four main steps:

1. **Nonces Exchange:** Nonces are exchanged between the host and the TA101-TFLXTLS device.
2. **Session Setup:** The host provides the parameters for the authorization session to the TA101-TFLXTLS device. Both the host and device use HKDF and the exchanged nonces to generate the AES key and initialization vector (IV) for MACs/encryption. The flags parameter can be used to specify whether command encryption is enabled during the session.
3. **Command Execution:**
   Any command (other than Authorize) can be sent to the TA101-TFLXTLS device, accompanied by an authorizing MAC. The entire target command sequence—including length, opcode, and parameters, but excluding the CRC—is encapsulated as the first data parameter of the Authorize command. The second parameter is the MAC, calculated using the session key, IV from step 2, and a sequence value. If the MAC is validated, the command in the first parameter is executed.

   This step can be repeated as needed to perform all required functions within the authorization session. There is a limit on the total number of commands that can be executed within a session, which is enforced after the session is terminated.

© 2026 Microchip Technology Inc. and its subsidiaries

The data parameter for the encapsulated command can be up to 1024 bytes, allowing for encrypted writes of up to 1024 bytes. The total length of the outer Authorize command and the inner embedded command must not exceed 1072 bytes, as detailed below:

- Total Length, Authorize Opcode, Mode, Param: 8 bytes
- Embedded Opcode, Mode, Param: 6 bytes
- Embedded Command Data: 1024 bytes
- Authorization MAC: 32 bytes (or 16 bytes if AES)
- CRC: 2 bytes

There are two versions of this step:

- In the first version, the nested command data parameter and the response data parameter are encrypted. The first six bytes of commands (Opcode, Mode, Param2) and the first byte of the response (Response Code) are not encrypted, but are included in the "authenticate-only" field of GCM for MAC calculation.
- In the alternate version, a MAC is generated over the embedded inputs or outputs, but the data parameters are transmitted in clear text.

4. **Session Termination:** The authorization session can be closed at any time by executing this phase. No authorizing MAC is required to terminate the session. Volatile registers created during the session are linked to that session and are automatically deleted when the session is closed.

5.1.3. Counter

The `Counter` command manages the nonvolatile counters within the TA101-TFLXTLS device. The specific counter to be read or incremented is identified by the handle reference:

1. Read the current value of the counter.
2. Increment the value of the counter.

If the command completes successfully, it returns a status code along with the counter value. If the command fails, only an error code is returned.

Table 5-5. Counter Mode Table

Command	Mode	Description
Counter	Read	Reads the value of the counter
	Increment	Increments the value of the counter

5.1.4. Create

The `Create` command is used to create an element in a volatile register with specific attributes. The size of the element is defined at creation and cannot be changed later.

Additionally, the Create command can be used to generate ephemeral keys in a volatile register. When this option is selected, the attributes provided are stored within the TA101-TFLXTLS device until the ephemeral private key is created and used via ECDH.

Table 5-6. Create Mode Table

Create	Mode	Description
Create	Volatile Register	Creates a volatile register with the specified volatile register ID.
	Ephemeral Volatile Register	Creates an ephemeral element in a volatile register with the specified ID.

© 2026 Microchip Technology Inc. and its subsidiaries

5.1.5. DevUpdate

The `DevUpdate` command updates the nonvolatile operating code within the TA101-TFLXTLS device. For details on available update images for a specific version of the TA101-TFLXTLS, please refer to the Microchip Technical Support Portal.

The update image is divided into blocks, which are sent sequentially to the TA101-TFLXTLS device. The first block is unique and contains information such as the update image version number, compatibility details for supported TA101-TFLXTLS versions, and integrity protection data. Up to 255 blocks can be updated, with each block containing between 1 and 1024 bytes.

All updates must be performed within an authorization session.

Table 5-7. DevUpdate Mode Table

Command	Mode	Description
Dev_Update	First Block	The first block is unique; the initial 26 bytes contain update information, and the remaining bytes may be a part of the firmware update.
	Additional Blocks	Subsequent blocks contain additional firmware data.

5.1.6. ECDH

The `ECDH` command implements the Elliptic Curve Diffie-Hellman (ECDH) key agreement scheme, enabling two parties to generate a shared secret for secure encryption and decryption. The ECDH mode supports the following curves: ECC P256, P384, P521, and X25519.

Table 5-8. ECDH Mode Table

Command	Mode	Description
ECDH	ECDH mode	Outputs either the X parameter only, or both X and Y parameters.

5.1.7. Info

The `Info` command retrieves status or state information from the TA101-TFLXTLS device. It can provide details about the device itself, as well as specific handles or data elements present on the device.

Table 5-9. Info Mode Table

Command	Mode	Description
Info	Revision Mode	Identifies the silicon and firmware revision of the device
	Dedicated Memory	Returns the current contents of the dedicated data memory
	ROM ID	Returns the current ROM ID for comparison with the ROM ID in an update package
	Pull-up Configuration	Returns a byte indicating which internal pull-ups are enabled
	Handle Valid	Provides information on the validity of a device handle
	Handle Info	Returns the attributes and status of a stored data element or volatile register
	Handle Array	Returns a 2-byte count of valid handles, followed by an array of valid handles in ascending order (each 2 bytes)
	Volatile Register Status	Provides information on all volatile registers defined in the device
	Authorization Status	Provides status information for each currently active authorization session
	Chip Status	Provides information on the current configuration, setup lock states, and V_{CC} latch status
	SHA_Status	Returns the status of the two SHA context registers
	Secure Boot Preset Status	Returns the status of secure boot preset and pre-boot preset

© 2026 Microchip Technology Inc. and its subsidiaries

5.1.8. KDF

The `KDF` command implements various Key Derivation Functions (KDF) supported by the TA101-TFLXTLS device:

- **HKDF**: Used in TLS1.3, with mode options for HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512
- **HMAC-Counter**: As specified in SP 800-108, also used for authorization session establishment
- **PRF**: Typically used in TLS 1.2 and earlier
- **SHA**: A one-step key derivation function using a single SHA-256 iteration, as documented in SP800-56Cr2 and as used in other protocols

Each requires different input and output parameters. If the output handle is a shared element within the TA101-TFLXTLS device, it must already exist on the device.

Table 5-10. KDF Mode Table

Command	Mode	Description
KDF	HKDF	KDF based on HMAC-SHA, specified for use with TLS 1.3
	HMAC-Counter	KDF based on HMAC-SHA256/Counter mode as defined in NIST SP 800-108
	PRF	Implements the KDF used in TLS 1.2 and earlier
	SHA-256	Generates a SHA-256 digest from the input key and optional data

5.1.9. KeyGen

The `KeyGen` command is used to generate asymmetric public/private key pairs or symmetric keys using the random number generator, or to compute a public key from an existing private key. The resulting private or secret key can be stored in either shared data memory or a volatile register. The following key types are supported:

- 16-byte or 32-byte AES symmetric key
- ECC P256, P384 and P521 private and public keys
- RSA-2048 and RSA-4096 private and public keys
- Ed25519 and X25519 private and public keys

Note: The TA101-TFLXTLS allows KeyGen only on volatile registers. Private keys on existing private key handles cannot be regenerated.

Table 5-11. KeyGen Mode Table

Command	Mode	Description
KeyGen	Generate Key	Generates a new key using the RNG and stores it in the specified handle. If a private key is generated, the corresponding public key is also output.
	Public Key	Computes the public key from an existing private key at the specified handle.

5.1.10. Lock

The `Lock` command is used to secure a memory area or change the state of the chip. It can lock the configuration memory or specific nonvolatile shared data elements. Once locked, the configuration memory or shared data element cannot be modified. The setup phase lock transitions the device from the setup phase to the production phase, and this process is irreversible.

© 2026 Microchip Technology Inc. and its subsidiaries

Table 5-12. Lock Mode Table

Command	Mode	Description
Lock	Config-NoCRC	Locks the configuration memory without CRC verification
	Config-CRC	Locks the configuration memory if the CRC matches (recommended)
	Data Element Lock	Locks the specified shared data element
	Setup Lock	Setup Phase Lock

5.1.11. MAC

The `MAC` command computes a Message Authentication Code (MAC) using a symmetric key stored within the TA101-TFLXTLS device and a provided input message. Supported algorithms include AES-CMAC and SHA-HMAC.

The mode parameter must be selected according to the desired hash algorithm or AES key size. If the key type is AES, the CMAC algorithm is used. If the key type is HMAC, the HMAC algorithm is used. An error will be returned for unsupported key types. Input messages can be up to 1022 bytes in length.

Table 5-13. MAC Mode Table

Command	Mode	Description
MAC	HMAC SHA-256	Computes a MAC using a SHA-256 hash over a symmetric key and input message
	HMAC SHA-384	Computes a MAC using a SHA-384 hash over a symmetric key and input message
	HMAC SHA-512	Computes a MAC using a SHA-512 hash over a symmetric key and input message
	AES256-CMAC	Computes a CMAC using the AES-256 cipher algorithm
	AES128-CMAC	Computes a CMAC using the AES-128 cipher algorithm

5.1.12. Manage_cert

The `Manage_cert` command is used to parse and verify X.509v3 certificates. Supported signing key algorithms include ECC-P256, ECC-P384, ECC-P521 (ECDSA), RSA-2048, RSA-4096, and Ed25519 (Pure and Ph variants). The subject key may use the same algorithm as the issuing key or a different supported algorithm. Other algorithms, modes, or key lengths are not supported at this time.

Table 5-14. Manage_cert Mode Table

Command	Mode	Description
Manage_cert	Validate Cert	Validates the certificate located in the target (input buffer, shared data, or volatile register).

5.1.13. Power

The `Power` command transitions the TA101-TFLXTLS device to Sleep mode or performs a soft reset, equivalent to asserting the $\overline{\text{RESET}}$ pin low. When Sleep mode is successfully initiated, the GPIO pin states remain unchanged throughout the device's sleep and wake cycle.

Table 5-15. Power Mode Table

Command	Mode	Description
Power	Active	Maintains normal operation mode
	Sleep	Initiates Sleep mode
	Reboot	Performs a soft reset of the device

5.1.14. Random

The `Random` command returns a random number with a security strength of 256 bits, generated by the device's internal FIPS-compliant True Random Number Generator (TRNG). If the noise source

© 2026 Microchip Technology Inc. and its subsidiaries

has not accumulated sufficient entropy before this command is called, the device will wait until adequate entropy is available before returning a result.

The TA101-TFLXTLS RNG generates 32 bytes of random data per iteration. Based on the input parameter, up to 1024 bytes (32 iterations of 32 bytes each) can be returned to the host.

It is recommended that the host provide 16 bytes of external data to be mixed into the random number calculation, in addition to the internal entropy source. This external data can come from any source (e.g., clock, system serial number) and there are no restrictions on its value. If no external data is available, the parameter must still be present and may be set to a fixed value, such as 16 bytes of '0'.

Table 5-16. Random Mode Table

Command	Mode	Description
Random	Random	Generates between 1 and 32 random numbers based on the length parameter. Each random number is 32 bytes in length with a security strength of 256 bits.

5.1.15. Read

The `Read` command retrieves an element from internal memory within the TA101-TFLXTLS device. Access to elements is governed by the attributes associated with each element. Volatile registers cannot be accessed using the `Read` command. Additionally, this command cannot be used to read asymmetric private keys, as there is no method for retrieving private keys from the TA101-TFLXTLS device.

⚠ CAUTION The `Read` command can only be executed after the device has completed a setup lock process. If the `Read` command is issued while the setup is unlocked, the device will return error code 0x8A.

The following types of information can be read, provided the appropriate attributes are set:

- **Configuration Memory**: The entire memory array or specific portions (using offset and length values) can be read
- **Data Elements**: Specific data elements can be read by specifying their handle
- **SHA Context Values**: The full context must be read; partial reads are not permitted
- **GPIO Pin Values**: The state of a GPIO pin is indicated in the least significant bit (LSb) of the byte returned when reading the associated handle

The data are returned in the clear by this command. However, the entire read sequence, including both the input and output, can be encrypted by encapsulating the command within an authorization session.

Table 5-17. Read Mode Table

Command	Mode	Description
Read	Partial Read	Performs a partial read of an element using specified offset and length
	Full Read	Performs a full read of the entire element. Full Reads are limited to handles that contain data smaller than 1024 bytes.

5.1.16. RSA_Enc

The `RSA_Enc` command provides RSA encryption and decryption as specified by the PKCS#1 standard. It supports SHA-256, SHA-384, or SHA-512 as the underlying hash function, and uses MGF1 as the mask generation function.

© 2026 Microchip Technology Inc. and its subsidiaries

Table 5-18. RSA_Enc Mode Table

Command	Mode	Description
RSA_Enc	Encrypt/Decrypt 2048	Encrypts 1 to 190 bytes of plaintext; decrypts 256 bytes of ciphertext
	Encrypt/Decrypt 4096	Encrypts 1 to 446 bytes of plaintext; decrypts 512 bytes of ciphertext

5.1.17. Secure_Boot

The `Secure_Boot` command manages the validation of the host operating code during power-up, reset, and wake events, if configured. The command supports two modes:

- **Full-Asymmetric Mode**: Validates the signature on every boot
- **Full-Stored Mode**: Compares the digest on every boot

The mode of operation is selected in the nonvolatile memory configuration. For the TA101-TFLXTLS, the size of the pre-boot digest and digest can be specified in the configuration. For Partial Secure Boot, the hash algorithm can also be configured.

The Secure_Boot command supports ECC-P256, ECC-P384, ECC-P521 keys, Ed25519 ph mode (without context), and 2048-bit RSA keys for signature verification. RSA4096 and Ed25519 Pure Ctx modes are not supported for code digest signature validation.

Digest calculation during the boot phases can be performed either externally or within the TA101-TFLXTLS device, regardless of the Secure Boot mode configured.

Table 5-19. Secure_Boot Mode Table

Command	Mode	Description
Secure_Boot	Full-Boot	Run-time boot phase for Full-Asymmetric mode; validates the certificate against the digest
	Full Stored-Update	Validates the new code signature and stores the new code digest for future use
	Full Stored-Boot	Run-time boot phase for Full-Stored mode; validates the input digest against the stored digest
	Lock	Prevents further Secure_Boot execution until the next power or reset cycle

5.1.18. Self-Test

The `Self_Test` command is used to verify the integrity and functionality of the internal cryptographic execution blocks within the main command processor. Tests can be selected individually or in combination, based on the map parameter provided. If any requested test fails, the TA101-TFLXTLS device will enter Self-Test Failure mode. To clear a failed test, the test must be executed again.

This command forces the execution of the specified tests, regardless of whether they were automatically performed during previous power-up or wake cycles. The total execution time for the Self-Test command is the sum of the delays for each requested operation. If all tests are selected, the overall execution delay may be significant.

The command output provides a map indicating which tests have failed, mirroring the input map.

Table 5-20. Self-Test Command Modes

Command	Mode	Description
Self-Test	Run Selected	Executes the self-tests specified in the Map Input Parameter
	Run Failure Clear Tests	Runs the tests in the Self-Test Failure-Clear configuration map

© 2026 Microchip Technology Inc. and its subsidiaries

5.1.18.1. Self-Test Map

Table 5-21. Self-Test Mode Map

Name	Byte #	Bit #	Description
Reserved	0	7:0	Must be 0
Reserved	1	0	Must be 0
ROM_CRC	1	1	Integrity test of all ROMs and loaded patch code in EEPROM using CRC (Use_HMAC = 0)
ROM_HMAC	1	1	Integrity test of all ROMs and loaded patch code in EEPROM using HMAC (Use_HMAC = 1)
Reserved	1	4:2	Must be 0
EdDSA-Ed25519	1	5	ECC (EdDSA) sign and verify using Ed25519
ECDH - X25519	1	6	ECDH operation using X25519
HMAC SHA512	1	7	Valid for SHA384, SHA512, SHA384-HMAC, and SHA512-HMAC
NRBG	2	0	RNG health test of noise source (1000 cycles)
DRBG	2	1	RNG digital test using CTR_DRBG AES-256 mode
Reserved	2	2	Must be 0
ECDSA	2	3	ECC (ECDSA) sign and verify; valid for all Pxxx curves
ECDH	2	4	ECDH using P256 curve; valid for all Pxxx curves
RSA	2	5	RSA sign and verify; valid for all RSA modulus sizes
Reserved	2	6	Must be 0
KEY	2	7	Stored key integrity test using the CRC algorithm
Reserved	3	1:0	Must be 0
CMAC	3	2	Valid for AES-128 CMAC and AES-256 CMAC in the main command processor
GCM	3	3	Valid for AES-128 and AES-256 GCM functions in the main command processor
KDF	3	4	KDF – SP800-108 based on SHA256/HMAC Counter mode in the main command processor
SHA	3	5	Valid for SHA256 and SHA256-HMAC in the main command processor
PRF	3	6	PRF – Based on SHA256-HMAC
HKDF	3	7	HKDF – Based on SHA256-HMAC and SHA512-HMAC

5.1.19. Sequence

The `Sequence` command enables data transfer between the host and the TA101-TFLXTLS device during a sequence calculation. If executed outside of a sequence, the command will always return a success status, regardless of the input provided.

Table 5-22. Sequence Mode Table

Command	Mode	Description
Sequence	Share_Key	Generates a shared key between two entities. The remote entity may be another TA101-TFLXTLS device or a different cryptographic system.

5.1.20. SHA

The `SHA` command calculates a SHA-256, SHA-384, or SHA-512 digest within the main command processor. Digest calculation typically involves a series of operations. If the message is less than 1,024 bytes, the digest can be computed in a single execution. For larger messages, the process includes an initialization phase, one or more update phases, and a finalize phase.

© 2026 Microchip Technology Inc. and its subsidiaries

The resulting digest can be stored in the SHA context for use in subsequent commands. The digest is always returned in the output buffer, but the host may choose to ignore it if only context storage is required.

Messages of any length are supported, but only 1,024 bytes may be processed per command iteration. Running the initialization phase invalidates any previous state before setting up the new context. Contexts can be read from the device using the `Read` command and written back using the `Write` command.

Note: The SHA command does not support HMAC calculation. Refer to the MAC command if HMAC is required.

Table 5-23. SHA Mode Table

Command	Mode	Description
SHA	Initialize	Initializes the SHA context; no input data required.
	Update	Updates the SHA context with input data.
	Finalize	Finalizes the SHA calculation; no input data required. The digest is placed in the output buffer and/or written to the context handle.
	Entire Message	Processes the entire SHA message in the input stream. The digest is placed in the output buffer and/or written to the context handle.

5.1.21. Sign

The `Sign` command generates a digital signature for a message using a private key stored within the TA101-TFLXTLS device. The Key_Type of the stored private key must match the Key_Type specified in the Input mode parameter; otherwise, an error will be returned. The Alg_Mode specified in the Input mode parameter is always used for the signature calculation, regardless of the algorithm associated with the key in the device.

The message to be signed can be sourced from the input buffer, a SHA context, or an internal memory location. The expected message length depends on the specific key used.

Table 5-24. Sign Mode Table

Command	Mode	Description
Sign	Internal Sign	Signs an internal message
	External Sign	Signs an internal message

5.1.22. Verify

The `Verify` command is used to validate a signature against a message and a public key. Both the message and the public key can be stored within theTA101-TFLXTLS device or provided as input parameters. If a public key is used from a stored certificate, the certificate must first be written to the device using the `Manage_Cert` command.

If the public key is stored as a public key element within the device, the Key_Type must match the Key_Type specified in the Input mode parameter; otherwise, an error will be returned. For stored data elements or when the public key is provided in the input stream, the Key_Type in the Input mode parameter is always used. The Alg_Mode in the Input mode parameter is always used for the verification calculation, regardless of the algorithm associated with the key in the device.

The message to be verified can be sourced from a SHA context or an internal memory location. For ECC-P256, ECC-P384, and ECC-P521, point expansion of the public key can also be performed during verification.

Table 5-25. Verify Mode Table

Command	Mode	Description
Verify	Verify	Performs signature verification with the key

© 2026 Microchip Technology Inc. and its subsidiaries

5.1.23. Write

The `Write` command writes an element to the memory of the TA101-TFLXTLS device. The target location can be either nonvolatile or volatile data memory, as specified by the handle parameter. The source can be the input data stream or another handle within the device, in which case the command functions as an internal transfer.

The following types of information can be written, provided the appropriate attributes are set and the device status allows the operation:

- **Data Elements**: Specific data elements can be written by specifying the handle and the data to be written. The element must not be locked.
- **SHA Context Values**: The entire SHA context must be written, with a required length of 106 bytes. Partial context writes are not permitted.
- **GPIO Pin Values**: For GPIO signals configured as outputs, the value can be written by specifying the GPIO handle. The write operation uses a single-byte value, with the least significant bit (LSb) indicating the output value.

Input data is transmitted in clear text to the TA101-TFLXTLS device. However, data can be encrypted by encapsulating the command within an authorization session, allowing secret data or keys to be securely written to the device.

Table 5-26. Write Mode Table

Command	Mode	Description
Write	Partial Write	Performs a partial write of an element using an offset and length
	Full Write	Performs a complete write of an element

© 2026 Microchip Technology Inc. and its subsidiaries

6. Application Information

6.1. External Components

The TA101-TFLXTLS device requires external bypass capacitors to be connected to the V_{CC} pin. It is recommended to use both 0.1 µF and 1 µF bypass capacitors in parallel, placed as close as possible to the V_{CC} pin. The 0.1 µF capacitor should be positioned closest to the pin to effectively handle high-frequency current transients.

For proper I^2C communication, pull-up resistors are always required on the system bus for the SDA and SCL pins. The appropriate resistor value depends on the total bus capacitance and the desired communication speed.

Pull-up resistors are not required for SPI bus operation, as the drivers use CMOS totem pole outputs. However, it is recommended that the system designer ensures that signals are not left floating, as this could result in unintended current consumption.

6.2. Onboarding Flow

1. The user begins by ordering the TA101x-TFLXTLSS-PROTO and using the necessary development tools (as referenced in Development Tools). The TA101x-TFLXTLSS-PROTO is intended for prototyping purposes only and is not suitable for production.
2. After finalizing use cases with the Trust Platform Design Suite (TPDS), the user works with the TPDS device configurator. The configurator enables the user to bring their own cryptographic assets to the device and leverage Microchip's secure provisioning service. An encryption utility within the TPDS configurator packages the required digital assets and submits to Microchip in the required format.
3. The user submits a support ticket requesting provisioning services and uploads the encrypted files. For additional guidance, please refer to the tutorial.
4. This request initiates the secret exchange process. The user is guided through the process of authorizing Microchip to provision the required cryptographic assets, keys and certificates in their TA101-TFLXTLS device on their behalf.
5. Upon completion, Microchip's factories, equipped with Hardware Secure Modules (HSMs), securely provision the user's cryptographic assets into the silicon and ship verification units.
6. The user purchases the verification units through Microchip's official purchasing channels, validates them, and provides digital confirmation via microchipdirect.com.
7. Once validation is confirmed, Microchip will proceed with mass production of the silicon, provisioned with the agreed-upon credentials.

6.3. Development Tools

The TA101-TFLXTLS is supported by a range of hardware and software tools, as well as backend services, to facilitate rapid application development. Initial development can begin with the Trust Platform Design Suite tools, which offer a user-friendly graphical interface to implement your use case and generate the necessary C code for your application.

For applications that require functionality beyond what the predefined Trust Platform Design Suite tools offer, development can be extended using CryptoAuthLib or its Python® version. CryptoAuthLib also serves as the foundation for the code generated by the Trust Platform Design Suite.

Comprehensive application verification can be performed using hardware tools in conjunction with TA101-TFLXTLS device samples. Since the device's access policies are pre-configured, development efforts can focus on system-level code implementation.

Once development is complete, TA101-TFLXTLS devices can be ordered directly through Microchip Direct.

© 2026 Microchip Technology Inc. and its subsidiaries

6.3.1. Trust Platform Design Suite (TPDS)

To streamline the implementation process, Microchip offers the web-based Trust Platform Design Suite (TPDS), which guides developers from concept to production through an intuitive workflow. TPDS enables proto provisioning of devices and provides example code to support application development within the configuration and access policy constraints of the TA101-TFLXTLS.

Development can begin with TA101-TFLXTLSx-PROTO (where x = SPI or I^2C) loose devices, allowing you to prototype and implement your selected use cases using TPDS. Information on compatible development kits and add-on sockets is provided in the following sections.

For additional details, please refer to Microchip's Trust Platform information page.

TA101-TFLXTLS Configurator

The TA101-TFLXTLS Configurator is available for download through myMicrochip as part of the TA101 TrustFLEX TPDS extension, upon registration. Installation instructions are provided in the package for integration with the Trust Platform Design Suite. The Configurator streamlines the generation and loading of user assets, which can be assigned to various data elements and sections. Once assets are uploaded, the connected TA101-TFLXTLS-PROTO device can be proto provisioned for different use cases.

The Configurator also allows modification of settings such as the I^2C address, Secure Boot modes, and lockable options, as permitted. This functionality enables the creation of provisioning packages for TA101-TFLXTLS production devices with user-defined configurations and assets. Provisioning packages are required to utilize the Microchip secure key provisioning service.

6.3.2. CryptoAuthLib

CryptoAuthLib is a software library designed to support Microchip's family of CryptoAuthentication™ devices. Microchip recommends using this library when developing with the TA101-TFLXTLS. CryptoAuthLib provides the necessary API calls to execute the commands described in this data sheet.

The library is designed for seamless integration with many of Microchip's microcontrollers and can be easily extended to support other microcontrollers through a Hardware Abstraction Layer (HAL), including those from other vendors.

For access to CryptoAuthLib, please refer to the TA101-TFLXTLS product page. Users are required to request the TrustFLEX Trust Anchor CryptoAuthLib (TrustFLEX TACAL) through the Secure Document Extranet (SDE) process. No non-disclosure agreement (NDA) is required to obtain the TrustFLEX TACAL.

6.3.3. Hardware Tools

A variety of hardware tools are available to support development with the TA101-TFLXTLS device. For information on additional tools not listed here, please refer to the Microchip website. Specific hardware tools are also referenced in relevant case examples throughout the documentation.

6.3.3.1. EV89U05A CryptoAuth Pro Trust Platform

The Microchip CryptoAuth Pro Trust Platform (EV89U05A) is an enhanced version of the CryptoAuth Trust Platform Board and is part of the CryptoAuthentication™ evaluation kit portfolio. This kit features a more powerful Arm® Cortex®-M4F microcontroller (PIC32CX SG41), four on-board CryptoAuthentication devices, two mikroBUS™ socket footprints, and an on-board 10/100 Mbit Ethernet PHY. Additionally, the board includes the Microchip Technology PKoB4 debugger, which is compatible with MPLAB® X IDE.

The EV89U05A CryptoAuth Pro Trust Platform is designed to work seamlessly with Microchip's Trust Platform Development Suite of tools, providing a comprehensive environment for secure application development.

© 2026 Microchip Technology Inc. and its subsidiaries

6.3.3.2. Add-On Socket Boards

Socket boards are used to facilitate application development with TA101-TFLXTLS-PROTO devices and to test verification units during the secure key provisioning process. These boards are designed to work in conjunction with microcontroller boards that support the mikroBUS™ extension header. Using a socket board allows developers to experiment with various device configurations to determine the optimal setup for a specific application.

The socket boards support both 3.3V and 5.0V supply operation and are compatible with either I^2C or SPI versions of the device, with appropriate jumper settings. Depending on the I/O mode, the device will support between one and three GPIOs.

Available socket board: **AC164167** – TA 8-PIN SOIC

© 2026 Microchip Technology Inc. and its subsidiaries

7. Electrical Characteristics

7.1. Absolute Maximum Ratings

Ambient Temperature under Bias[1]	-40°C to +125°C
Storage Temperature (without Bias)	-65°C to +150°C
Maximum Supply Voltage	6.0V
DC Voltage on Any Pin[2]	-0.5V to V_{CC} + 0.5V
ESD Ratings	
— Human Body Model (HBM) ESD[3]	≥ ±4 kV
— Charged Device Model (CDM) ESD[4]	≥ ±750V

Notes:

1. Recent Partial Networking Transceivers from Microchip and others use a spec throughout the document called the Virtual Junction Temperature, measured in accordance with IEC60747-1. An alternate definition is $T_{VJ} = T_A + P \times R_{th(j-a)}$, where P is the power and $R_{th(j-a)}$ is the thermal resistance from virtual junction to ambient. T_{VJ} would be higher than +125°C (maximum).
2. V_{CC} is the supply voltage where the device is driven and must be within the specified operating voltage range.
3. Specified by: JEDEC® Standard JS-001-2017.
4. Specified by: JEDEC® Standard JS-002-2014.

Note: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

7.2. Reliability

The TA101-TFLXTLS is fabricated with Microchip's high reliability CMOS EEPROM manufacturing technology.

Table 7-1. EEPROM Reliability

Parameters	Min.	Typ.	Max.	Units
Write Endurance at +25°C (Each Byte)	100k	—	—	Write Cycles
Data Retention at +55°C	>40	—	—	Years
Read Endurance	Unlimited			Read Cycles

7.3. DC Characteristics

Table 7-2. DC Characteristics – All Interfaces

Applicable over the recommended operating range from T_A = -40°C to +125°C, V_{CC} = +2.7V to +5.5V.

Parameters	Test Conditions	Symbol	Min.	Typ.	Max.	Units	Type[1]
Supply Voltage on Pin V_{CC}	—	V_{CC}	2.7	—	5.5	V	A
Supply Current on Pin V_{CC}	Active mode[4]	I_{IO_Active}	—	25	40	mA	A
	Idle mode[2] (T_A = +85°C)	I_{IO_Idle}	—	—	10	mA	B
	Sleep mode	I_{IO_Sleep}	—	7	15	uA	B
V_{CC} Rise Rate	—	V_{RISE}	—	—	0.1	V/µs	C
Undervoltage Detection Rising	V_{CC} rising	V_{UVDR}		2.58		V	B
Undervoltage Detection Falling	V_{CC} falling	V_{UVDF}		2.4		V	B

© 2026 Microchip Technology Inc. and its subsidiaries

Table 7-2. DC Characteristics – All Interfaces (continued)

Parameters	Test Conditions	Symbol	Min.	Typ.	Max.	Units	Type(1)
Power-on Reset Voltage Rising	V_{CC} rising[3]	V_{PORR}	1.8		2.3	V	B
Power-on Reset Voltage Falling	V_{CC} falling[3]	V_{PORF}	1.75		2.1	V	B
High-Level Input Voltage	—	V_{IH}	0.7 x V_{CC}	—	V_{CC} + 0.3	V	A
Low-Level Input Voltage	—	V_{IL}	-0.3	—	0.3 x V_{CC}	V	A
Theta JA	—	Θ_{JA}	—	73	—	°C/W	8-Pin SOIC
				41.3			24-pad VQFN

Notes:

1. Type means: A = 100% tested, B = characterized, C = design parameter.
2. Idle means that power is applied, the device is NOT in Sleep mode and no commands nor instructions are running.
3. The state of the V_{CC} latches will be retained so long as V_{CC} remains above the V_{POR} level.
4. Active current is measured with all GPIO pins either driven to ground or configured as inputs. Active current also excludes any DC load on the I/O pins.

Table 7-3. DC Characteristics – SPI Interface, $\overline{\text{RESET}}$ and GPIO Pins

Applicable over the recommended operating range from T_A = -40°C to +125°C, V_{CC} = +2.7V to +5.5V.

Parameters	Test Conditions	Symbol	Min.	Typ.	Max.	Units	Type(1)
Input Current(2)	0.1V_{CC} < Vi < 0.9V_{CC}	I_L	-2	—	+2	µA	A
Programmable Pull-Up(3)	—	R_{PU}	24k	40k	62k	Ω	A
High-Level Output Voltage	I_{OH} = -4 mA	V_{OH}	V_{CC} – 0.4	—	—	V	A
Low-Level Output Voltage	I_{OL} = 4 mA	V_{OL}	—	—	0.4	V	A

Notes:

1. Type means: A = 100% tested.
2. This specification is only valid when the internal pull-ups are disabled. Otherwise, the input current is determined by the internal pull-up resistance value R_{PU}.
3. See External Components for default device I/O configuration.

Table 7-4. DC Characteristics of SDA and SCL Pins for I^2C Interface

Applicable over the recommended operating range from T_A = -40°C to +125°C, V_{CC} = +2.7V to +5.5V.

Parameters	Test Conditions	Symbol	Min.	Typ	Max.	Units	Type(1)
Input Current(2)	0.1V_{CC} < Vi < 0.9V_{CC}	Ii	-10	—	+10	µA	A
Low-Level Output Voltage	I_{OL} = 20 mA V_{CC} > 3.6V to 5.5V	V_{OL}	0	—	0.4	V	B
	I_{OL} = 14 mA V_{CC} = 2.7V to 3.6V	V_{OL}	0	—	0.4	V	B
Programmable Pull-Up(3)	—	R_{PU}	2.3k	3.0k	4.5k	Ω	A

Notes:

1. Type means: A = 100% tested, B = characterized on samples.
2. The input current specification is only valid when the internal pull-ups are disabled. Otherwise, the input current is determined by the internal pull-up resistance value R_{PU}.
3. See External Components for default device I/O configuration.

© 2026 Microchip Technology Inc. and its subsidiaries

7.4. AC Characteristics

7.4.1. All Interfaces

Table 7-5. AC Timing Characteristics – All Interfaces
Applicable over the recommended operating range from T_A = -40°C to +125°C, V_{CC} = +2.7V to +5.5V.

Parameters	Symbol	Min.	Typ.	Max.	Units	Type(4)
Wake-up Time from Sleep State. V_{CC} > 2.7V	$t_{PU.SLEEP}$(1)	—	3	6.0	ms	A
Power-up Time from V_{CC} < 2.7V	$t_{PU.POWERON}$(1)	—	4	8.0	ms	A
Idle Tmer	t_{IDLE}(2)	0.85	1	1.15	s	B
Rate at which the Nonvolatile Portion of Monotonic Counter Increments	$t_{MONOTONIC}$	42	51	60	s	B
Noise Suppression on $\overline{RESET}$ Input Pin	t_{RESET_NOISE}(3)	0	—	0.150	µs	A
Minimum Allowed Reset Pulse	t_{RESET_MIN}(3)	1.0	—	—	µs	A
GPIO_3 Transition Ignored, Measured Starting with the Last Bit of Power (Sleep)	t_{SLEEP_WAKE}	—	—	250	µs	A
Low-Pulse Width for GPIO_3 High to Wake TA101-TFLXTLS	$t_{WAKE_GPIO_LOW}$	40	—	—	µs	A
Watchdog Time-out Value	$t_{WATCHDOG}$	900	1000	1100	ms	B

Notes:

1. Various situations can cause the power-up delays to exceed these parameters as follows:
   - If the power-on or the wake self-test functions are enabled in the configuration area, the execution of those self-test operations will increase the delay
   - If an internal failure occurs to cause a boot event, then, there may be an additional delay during the boot to write the internal failure log in the nonvolatile memory within the chip
   - If a device update is started but does not complete due to a power interruption, on the next power-up, some clean-up may be required and may take additional time
   - If the 1-minute timer is enabled and is being updated in the nonvolatile memory concurrent with the wake event, the device will accept an `Input` command after t_{PU_SLEEP}/$t_{PU_POWERON}$, but will not start the execution of that command until the nonvolatile update is complete
2. The idle timer specifications here assume that the idle timer is enabled and configured for 1 second. It is recommended that these times be multiplied by the delay time value set in the idle timer configuration field if that is not 1.
3. All noise pulses ≤ t_{RESET_NOISE} are assured to be suppressed. All pulse widths ≥ t_{RESET_MIN} are assured to pass to the device. Pulses in between these values may or may not be suppressed.
4. Type Means: A = 100% Tested, B = Characterized.

7.4.2. I²C Interface Timing

Table 7-6. AC Characteristics of I²C Interface
Applicable over the recommended operating range from T_A = -40°C to +125°C, V_{CC} = +2.7V to +5.5V.

Parameters	Symbol	Fast-Mode Plus		Units
		Min.	Max.	
SCL Clock Frequency	f_{SCL}	—	1000	kHz
SCL High Time	t_{HIGH}	260	—	ns
SCL Low Time	t_{LOW}	500	—	ns
Start Setup Time	$t_{SU.STA}$	260	—	ns
Start Hold Time	$t_{HD.STA}$	260	—	ns
Stop Setup Time	$t_{SU.STO}$	260	—	ns

© 2026 Microchip Technology Inc. and its subsidiaries

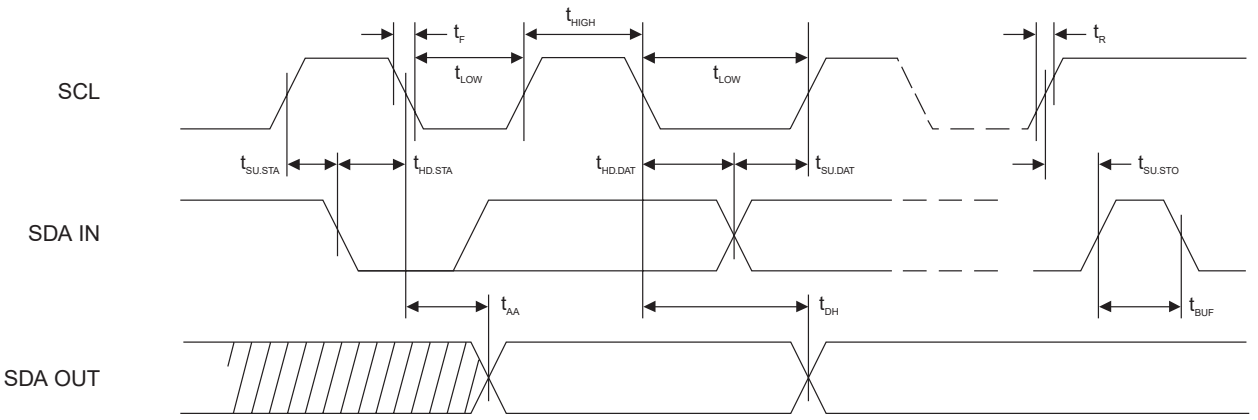
Table 7-6. AC Characteristics of I^2C Interface (continued)

Parameters	Symbol	Fast-Mode Plus		Units
		Min.	Max.	
Data in Setup Time	$t_{SU.DAT}$	50	—	ns
Data in Hold Time	$t_{HD.DAT}$	0	—	ns
Input Rise Time(1, 3)	t_R	—	120	ns
Input Fall Time(1, 3)	t_F	20 x (V_{DD}/5.5V)(5)	120	ns
Clock Low to Data Out Valid	t_{AA}	—	450	ns
Time bus must be free before a new transmission can start (1)	t_{BUF}	500	—	ns
Pulse width of spikes that must be suppressed by the input filter(4)	t_{SP}	—	50	ns

Notes:

1. Values are based on characterization and are not tested.
2. AC measurement conditions: input pulse voltages: 0.3 x V_{CC} to 0.7 x V_{CC}, input rise and fall times: ≤ 50 ns.
3. System designers must ensure that all AC parametrics are met. Rise fall times shown are for the Fast Mode Plus (1 MHz) of operation. For slower clock speeds, the rise and fall times may be increased but must still meet the industry standard I^2C specification UM10204.
4. Input filters on the SDA and SCL pins will suppress noise spikes of less than 50 ns.
5. Backwards compatibility is necessary for the Fast mode (400 kHz) specifications.

Figure 7-1. I^2C Synchronous Data Timing



7.4.3. SPI Interface Timing

Table 7-7. AC Characteristics of SPI Interface

Applicable over the recommended operating range from T_A = -40°C to +125°C, V_{CC} = +2.7V to +5.5V.

Parameters	Symbol	Min.	Max.	Units
SCK Clock Frequency	f_{SCK}	—	16	MHz
SCK High Time	t_{WH}	20	—	ns
SCK Low Time	t_{WL}	25	—	ns
$\overline{CS}$ High Time	t_{CS}	100	—	ns
$\overline{CS}$ Setup Time	t_{CSS}	100	—	ns
$\overline{CS}$ Hold Time	t_{CSH}	100	—	ns
Data in Setup Time	t_{SU}	5	—	ns
Data in Hold Time	t_H	5	—	ns

© 2026 Microchip Technology Inc. and its subsidiaries

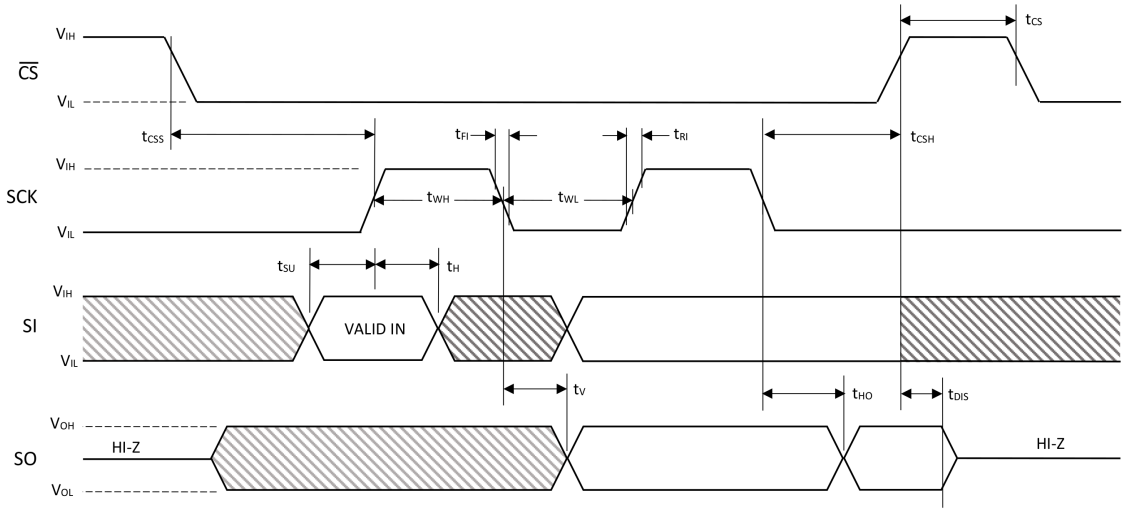
Table 7-7. AC Characteristics of SPI Interface (continued)

Parameters	Symbol	Min.	Max.	Units
Input Rise Time(1, 2)	t_{RI}	—	2	µs
Input Fall Time(1, 2)	t_{FI}	—	2	µs
Output Valid	t_V	—	25	ns
Output Hold Time	t_{HO}	0	—	ns
Output Disable Time	t_{DIS}	—	25	ns

Notes:

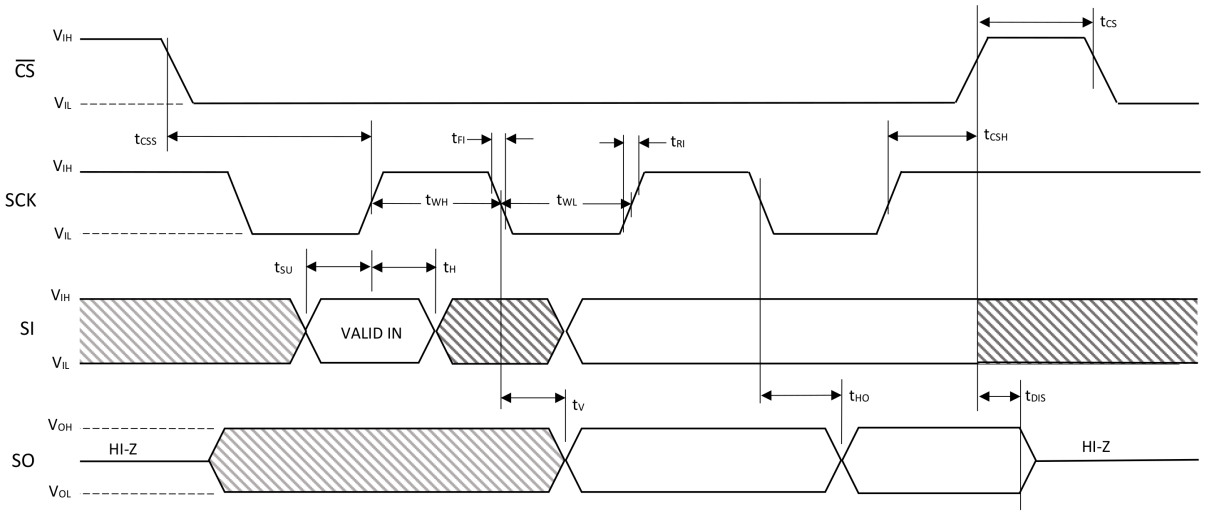
1. Values are based on characterization and are not production tested.
2. System designers must ensure that all AC parametrics are met, which will typically require rise and fall times faster than these values for most clock rates. Ramp rates slower than this may result in improper operation.

Figure 7-2. SPI Mode 0 Synchronous Data Timing



© 2026 Microchip Technology Inc. and its subsidiaries

Figure 7-3. SPI Mode 3 Synchronous Data Timing

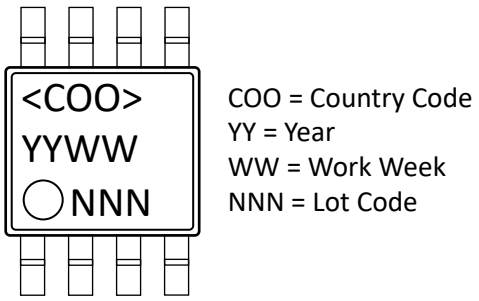


© 2026 Microchip Technology Inc. and its subsidiaries

8. Package Marking Information

As part of Microchip's overall security features, the part marking on all crypto devices is intentionally vague. The marking on the top of the package does not reveal the specific device type or the manufacturer. The alphanumeric code present on the package contains manufacturing information, which may vary by assembly lot. Therefore, it is recommended that the package marking not be used as a means of device identification during inspection procedures.

Figure 8-1. 8-Pin SOIC Package Marking



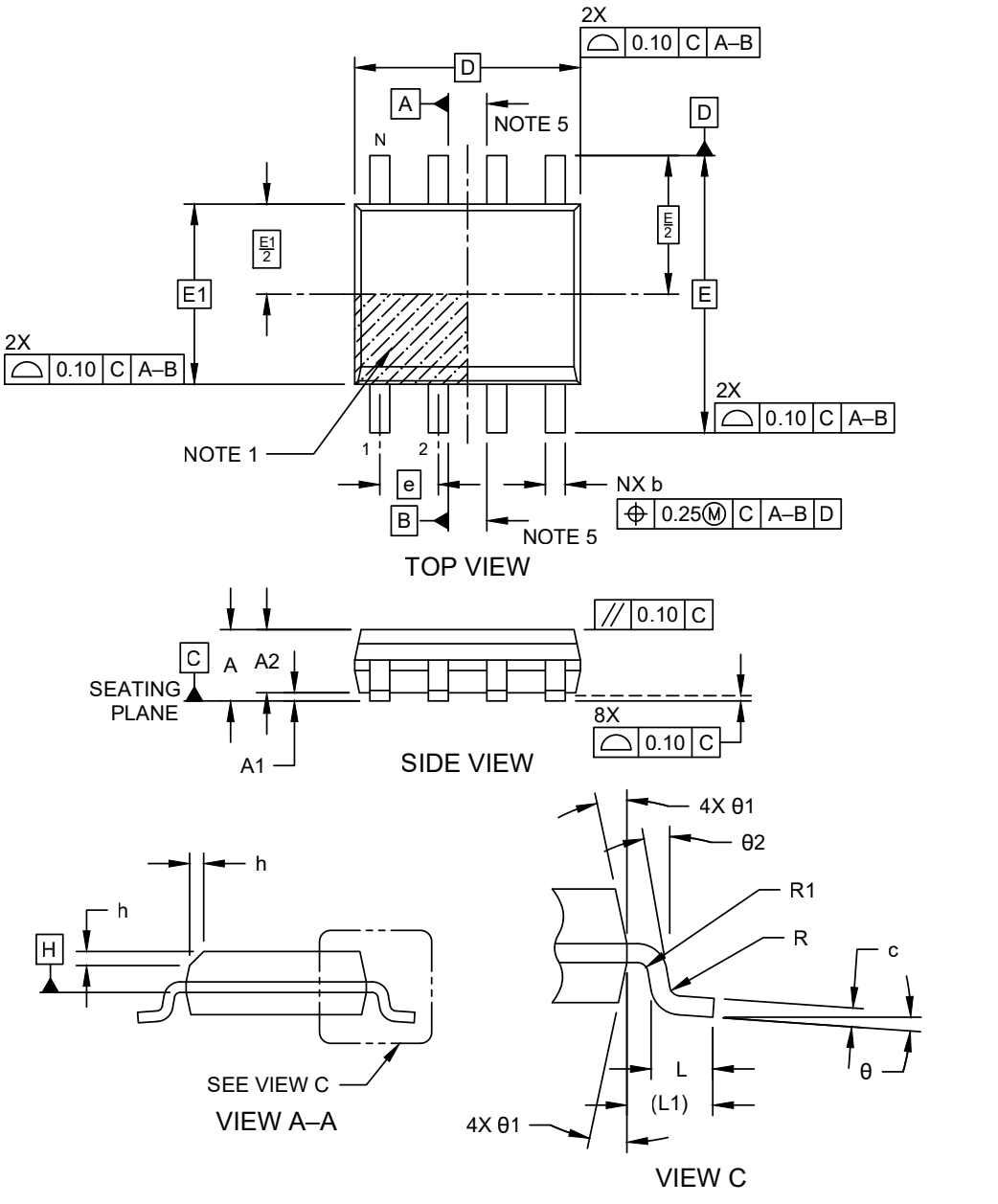
© 2026 Microchip Technology Inc. and its subsidiaries

9. Package Drawings

9.1. 8-Lead SOIC

8-Lead Plastic Small Outline (OA) - Narrow, 3.90 mm (.150 In.) Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging

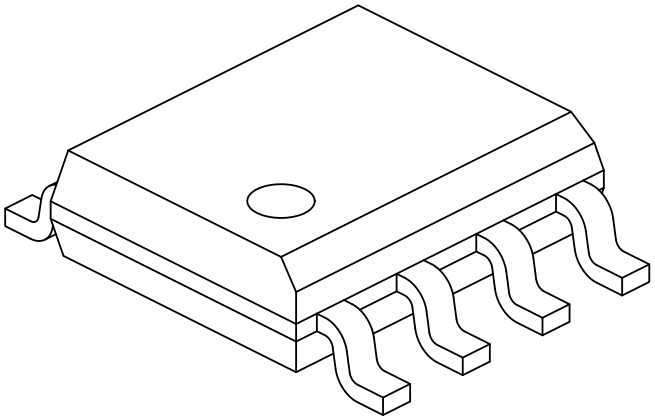


Microchip Technology Drawing No. C04-00057-OA Rev L Sheet 1 of 2

© 2026 Microchip Technology Inc. and its subsidiaries

8-Lead Plastic Small Outline (OA) - Narrow, 3.90 mm (.150 In.) Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		8	
Pitch	e		1.27 BSC	
Overall Height	A	–	–	1.75
Molded Package Thickness	A2	1.25	–	-
Standoff §	A1	0.10	–	0.25
Overall Width	E		6.00 BSC	
Molded Package Width	E1		3.90 BSC	
Overall Length	D		4.90 BSC	
Chamfer (Optional)	h	0.25	–	0.50
Foot Length	L	0.40	–	1.27
Footprint	L1		1.04 REF	
Lead Thickness	c	0.17	–	0.25
Lead Width	b	0.31	–	0.51
Lead Bend Radius	R	0.07	–	–
Lead Bend Radius	R1	0.07	–	–
Foot Angle	θ	0°	–	8°
Mold Draft Angle	θ1	5°	–	15°
Lead Angle	θ2	0°	–	–

Notes:

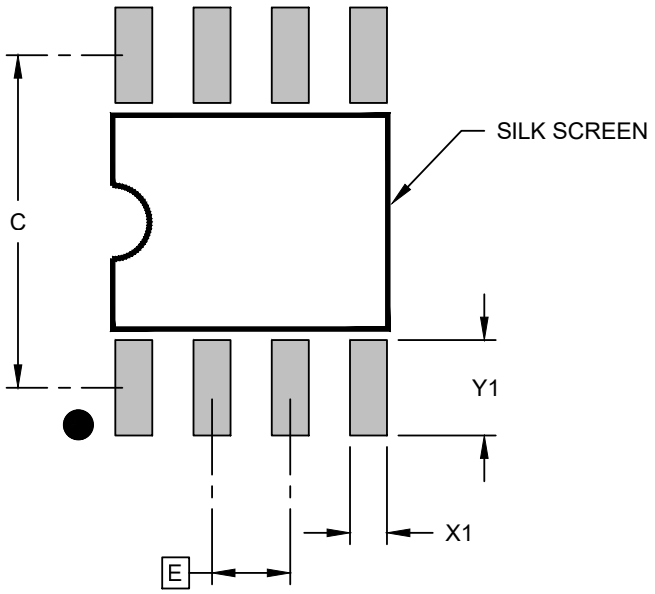
1. The Pin 1 visual index feature may vary, but it must be located within the hatched area.
2. § Significant Characteristic
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.
5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing No. C04-00057-OA Rev L Sheet 2 of 2

© 2026 Microchip Technology Inc. and its subsidiaries

8-Lead Plastic Small Outline (OA) - Narrow, 3.90 mm (.150 In.) Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		1.27 BSC	
Contact Pad Spacing	C		5.40	
Contact Pad Width (X8)	X1			0.60
Contact Pad Length (X8)	Y1			1.55

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-02057-OA Rev L

© 2026 Microchip Technology Inc. and its subsidiaries

10. Product Identification System

To order or obtain information, for example, on pricing or delivery, contact Microchip: https://www.microchip.com/en-us/about/contact-us.

PART NO	I/O Type	-	Trust Type	Trust Option	Package Option	-	Shipping Format	-	Product Identifier
xxxxx	y	-	ttt	vvv	ppp	-	s	-	VAO

Device:	TA101	
I/O Type	Blank	8-PIN SOIC SPI Interface Only
	T	8-PIN SOIC I^2C Interface Only
Trust Type	ttt	TFLX
Trust Option	vvv	TLS
Package Option	S	8-Pin SOIC
Shipping Options	T	Tape and Reel(1)
	B	Bulk Units
Automotive Identifier	VAO	Generic Automotive Product; only for automotive-qualified ICs. Otherwise blank.

Examples:

Customer Ordering Code	Interface	Package	Delivery	MOQ	Description
TA101T-TFLXTLSS	I^2C	SOIC-8	Tape and Reel	3300	TA101-TFLXTLS, I^2C, SOIC-8, T&R 3300 MOQ
TA101T-TFLXTLSS-HV	I^2C	SOIC-8	Tape and Reel	9900	TA101-TFLXTLS, I^2C, SOIC-8, T&R 9900 MOQ For non-automotive, high-volume applications, budgetary quote P/N
TA101-TFLXTLS-HV	SPI	SOIC-8	Tape and Reel	9900	TA101-TFLXTLS, SPI, SOIC-8, T&R 9900 MOQ For non-automotive, high-volume applications, budgetary quote P/N
TA101-TFLXTLSS	SPI	SOIC-8	Tape and Reel	3300	TA101-TFLXTLS, SPI, SOIC-8, T&R 3300 MOQ
TA101T-TFLXTLSS-HV-VAO	I^2C	SOIC-8	Tape and Reel	9900	TA101-TFLXTLS, I^2C, SOIC-8, T&R 9900 MOQ For automotive applications, budgetary quote P/N
TA101-TFLXTLS-HV-VAO	SPI	SOIC-8	Tape and Reel	9900	TA101-TFLXTLS, SPI, SOIC-8, T&R 9900 MOQ For automotive applications, budgetary quote P/N
TA101T-TFLXTLSS-PROTO	I^2C	SOIC-8	Bulk	10	TA101-TFLXTLS, I^2C, SOIC-8, Bulk 10 MOQ
TA101-TFLXTLSS-PROTO	SPI	SOIC-8	Bulk	10	TA101-TFLXTLS, SPI, SOIC-8, Bulk 10 MOQ

Note:

- Tape and Reel identifier appears only in the catalog part number description. This identifier is used for ordering purposes and is not printed on the device package. Check with your Microchip Sales Office for package availability with the Tape and Reel option.

© 2026 Microchip Technology Inc. and its subsidiaries

11. Revision History

Revision C (July 2026)

Information added regarding additional handles, Secure Boot and MACSec changes.

Revision B (March 2026)

Removed Confidential and NDA notices from cover page and document footer.

Revision A (March 2026)

Initial release of this document.

Table 11-1. Silicon Revision Information
Revision information can be read by using the `Info` Command

Device	Hardware Revision	Device Rom Revision	ACE ROM Revision
TA101-TFLXTLS	0x81, 0x02	0x00, 0x01	0x01

© 2026 Microchip Technology Inc. and its subsidiaries

Microchip Information

Trademarks

The “Microchip” name and logo, the “M” logo, and other names, logos, and brands are registered and unregistered trademarks of Microchip Technology Incorporated or its affiliates and/or subsidiaries in the United States and/or other countries (“Microchip Trademarks”). Information regarding Microchip Trademarks can be found at https://www.microchip.com/en-us/about/legal-information/microchip-trademarks.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP “AS IS”. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP’S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip products are strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.

© 2026 Microchip Technology Inc. and its subsidiaries

Product Page Links

TA101-TFLXTLS

© 2026 Microchip Technology Inc. and its subsidiaries