Product Overview

AM26xx Family TIFS-SDK Product Brief

1 Security Goals of AM26x Devices

- Modules and platform protection -
  - Protect modules (hardware and software) and defend platform from takeover and unauthorized modifications.
  - Protect critical assets and resources from hardware and software attacks
- Limit the attack surface for critical assets -
  - Isolate critical assets in protected space with heavily restricted access. Focus on protection against class-based attacks.
  - Assume rest of system is compromised to protect critical assets.
- Sand-box security -
  - Security operates in isolated environment.
  - Application modules/tasks are securely isolated from each other, even on the same CPU.
- Layered security -
  - Multi-tier approach, such that compromises do not spread and break the entire system security.
  - Each tier operates in isolation with other tiers.
- Traceability, accountability and isolation for security development -
  - Security must be developed in isolated environment so that unexpected potential leaks are avoided.
  - This is also required to prove security to certification entities and customers.

2 Software Components Delivered by TI

TI delivers two primary software components as part of TIFS-SDK:

- OTP Key Provisioning Package
- TIFS-MCU (TI's Foundational Security) Add-on Package

3 Device Lifecycle and Provisioning Flow

The flow is also explained in MCU+ Academy.

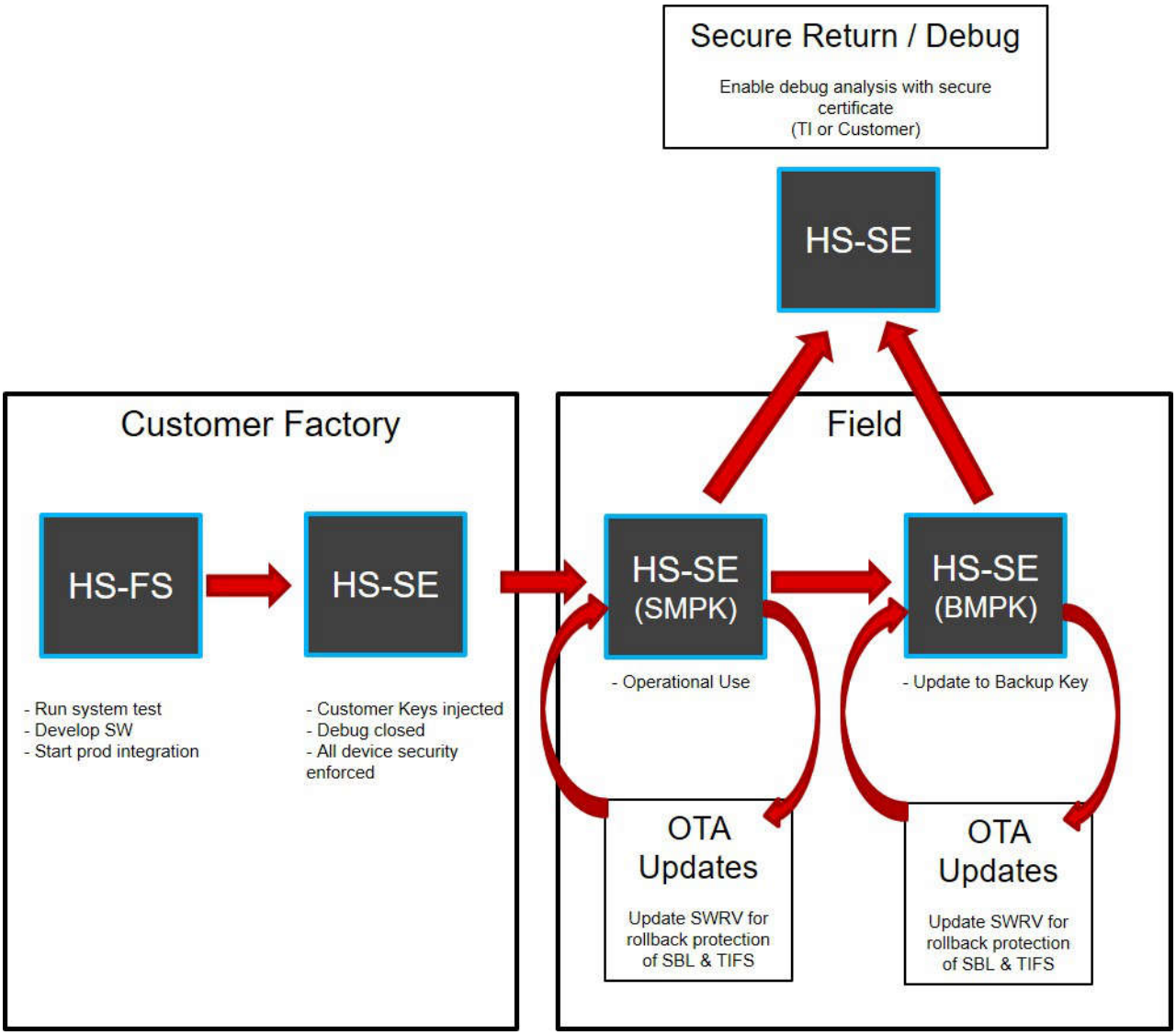

Figure 1. Secure Lifecycle of AM26x Device Family

Stage 1:

User is delivered the device in HSFS (Field Secure) state. The device contains TI keys provisioned. In this state, the HSM core only executes a code which is encrypted and signed with TI keys.

Stage 2:

TI supports provisioning of user keys by using a key provisioning package (in a trusted environment), with user keys encrypted and signed using the TI keys. This key certificate is then securely transmitted using the SBL Key Writer example executing on the device available as part of OTP Key Provisioning Package. There are multiple ways of using SBL Key Writer, for example UART Boot Mode, Flash Boot Mode (QSPI or OSPI boot mode) and more. Confidentiality of the user keys is maintained throughout the process that allows user to replicate the process even in a non-secure environment. Once user encryption keys are provisioned, the life-cycle of the device is changed to HSSE state.

Stage 3:

Now as the device is in HSSE state for example, device with keys provisioned, secure boot enforced and debug interfaces locked, the application code can be programmed into the external flash (using the device, if required). The uniflash available as part of MCU+ SDK can be used to program the bootloader as well as application images into the external flash. Note, as the device is transitioned to HSSE, the provisioning process by uniflash images are also securely booted on the device. As the device supports encrypted secure boot, the data can be made encrypted over communication medium as well as external flash storage to maintain confidentiality.

The user code for example, HSM Run Time firmware and the SBL are encrypted and signed using the user keys that is available in the device secure storage. Once all the required images such as HSM Run Time software, SBL software and Application Images are provisioned the code is always a secure boot.

4 TI's AM26xx OTP Key Writer Package

TI provides a OTP Key Writer package which enables the transitioning of the Secure device life cycle from HS-FS (development variant without security enforcement) to HS-SE (production variant with security enforced). These provisioning flows are end-to-end secured and can be utilized for non-secure factory floor provisioning.

4.1 List of Features Supported by OTP Key Writer Flow

- Signed Key writer firmware for HSM which accepts x.509 customer keys certificate with all eFuse fields configured.
- Supports programming keys at one-pass customer key certificates.
- Supports following boot modes - UART, JTAG, USB (only for AM261x) and Flash (QSPI/OSPI) modes for boot for key programming.
- Supports RSA as well as ECDSA based OTP programming.
- Supports OpenSSL v3.0.2 and above.
- Encryption keys (SMEK and BMEK) are made optional. Public keys (SMPK and BMPK) are mandatory fields.
- Option of using Python script for generating x.509 certificate
- Following are the keys programmable:
  - MSV
  - SMPK, SMEK
  - BMPK, BMEK
  - EXT OTP
  - KEY COUNT
  - SWREV-HSM, SWREV-APP, SWREV-SBL
  - KEY REV

5 TI Foundational Software for MCU devices

What is TIFS-MCU ?

TIFS stands for Texas Instruments Foundational Security for AM26xx SoCs. It provides device root of trust and foundational security services. The HSM or hardware security module consists of a secure core based secure subsystem.

TIFS-MCU serves as an add-on package on top of MCU+ SDK offering for AM26xx devices like AM263x/ AM263Px/AM261x. TIFS-MCU enables a baremetal security stack on secure CPU that can be leveraged by the user too.

1. Develop device root of trust and provide foundational security services
2. Integrate with 3P Auto-HSM stacks

TIFS-MCU is not a replacement for AUTOSAR-HSM stack. TIFS-MCU enables foundational security SW with all the building blocks required for root-of-trust within the device and utilizes various services. TIFS-MCU can be easily integrated by AUTOSAR-HSM stack vendors to develop HSM stacks that adhere to SHE/EVITA standards.

Table 1. List of Features Supported by HSSE Based Secure Boot (Supported by ROM)

Features of Secure Boot	Algorithm Supported (AM263x/ AM263Px)	Algorithm Supported (AM261x)	Support available in 10.02.00
HSM Run Time Firmware Boot	• Certificate verification • RSA-4K • Decryption Support • AES-CBC-256	• Certificate verification • RSA-4K • ECDSA (secp256r1) • ECDSA (secp384r1) • ECDSA (secp521r1) • ECDSA (brainpool512r1) • Decryption Support • AES-CBC-256	Yes
SBL Boot	• Certificate verification • RSA-4K • Decryption Support • AES-CBC-256	• Certificate verification • RSA-4K • ECDSA (secp256r1) • ECDSA (secp384r1) • ECDSA (secp521r1) • ECDSA (brainpool512r1) • Decryption Support • AES-CBC-256	Yes

Table 2. List of Features Supported by HSSE Based Secure Boot (Support by TIFS-MCU)

Features of Secure Boot	Algorithm Supported (AM263x)	Algorithm Supported (AM263Px)	Algorithm Supported (AM261x)	Support Available in 10.02.00
RAM based Multi Core Application Boot through Root Keys	• Certificate verification • RSA-4K • Decryption Support • AES-CBC-256	• Certificate verification • RSA-4K • Decryption Support • AES-CBC-256	• Certificate verification • RSA-4K • ECDSA (secp256r1) • ECDSA (secp384r1) • ECDSA (secp521r1) • ECDSA (brainpool512r1) • Decryption Support • AES-CBC-256	Yes
XiP based Multi Core Application Boot through Root Keys	• XiP not supported on AM263x	• MAC Support via Root Keys • AES-GCM-128 • Decryption Support via Root Keys • AES-CTR-128	• MAC Support via Root Keys • AES-GCM-128 • Decryption Support via Root Keys • AES-CTR-128	Yes
RAM based Multi Core Application Boot through Auxilary Keys	• Certificate verification (with different SHA options) • RSA-4K • ECDSA (secp256r1) • ECDSA (secp384r1) • ECDSA (secp521r1) • ECDSA (brainpool512r1) • Decryption Support • AES-CBC-256	• Certificate verification (with different SHA options) • RSA-4K • ECDSA (secp256r1) • ECDSA (secp384r1) • ECDSA (secp521r1) • ECDSA (brainpool512r1) • Decryption Support • AES-CBC-256	• Certificate verification (with different SHA options) • RSA-4K • ECDSA (secp256r1) • ECDSA (secp384r1) • ECDSA (secp521r1) • ECDSA (brainpool512r1) • Decryption Support • AES-CBC-256	Yes

Table 2. List of Features Supported by HSSE Based Secure Boot (Support by TIFS-MCU) (continued)

Features of Secure Boot	Algorithm Supported (AM263x)	Algorithm Supported (AM263Px)	Algorithm Supported (AM261x)	Support Available in 10.02.00
XiP based Multi Core Application Boot through Auxilary Keys	• XiP not supported on AM263x	• MAC Support via Auxilary Keys • AES-GCM-128 • Decryption Support via Auxilary Keys • AES-CTR-128	• MAC Support via Auxilary Keys • AES-GCM-128 • Decryption Support via Auxilary Keys • AES-CTR-128	Yes

For more details on secure boot time on AM26x devices are available in the list of links.

Table 3. List of Software Deliverables for Secure Programming flow

List of Software Components	Software Type	OPN	Delivery Location	Source Available in 10.02.00
SBL Keywriter	Example	AM263X_RESTRICTED_SECURITY	Secure Resources	Yes
		AM263PX_RESTRICTED_SECURITY		
		AM261x-TIFS-SDK		
Uart Bootloader	Tool for - • Windows • Linux • MacOS	MCU_PLUS_SDK	ti.com	Yes
Uart Uniflash	Tool for - • Windows • Linux • MacOS	MCU_PLUS_SDK	ti.com	Yes
OTP Key Writer Certificate Generation	Python tool	AM263X_RESTRICTED_SECURITY	Secure Resources	Yes
		AM263PX_RESTRICTED_SECURITY		
		AM261x-TIFS-SDK		
OTP KW HSM firmware	Encrypted and signed with TI Keys	AM263X_RESTRICTED_SECURITY	Secure Resources	No
		AM263PX_RESTRICTED_SECURITY		
		AM261x-TIFS-SDK		
SBL and HSM signing tool	Python tool	MCU_PLUS_SDK	Secure Resources	Yes
App signing tool	Python tool	MCU_PLUS_SDK	Secure Resources	Yes

Native services provided by TIFS-MCU

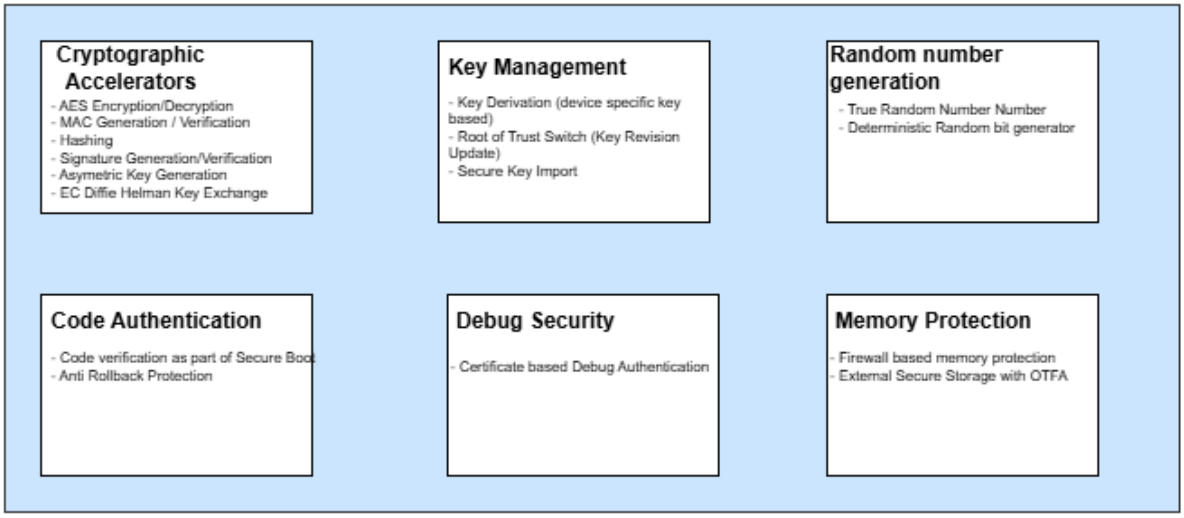

Figure 2. Top Security Features of TIFS-SDK of AM26x Devices

Software block diagram of TIFS-MCU

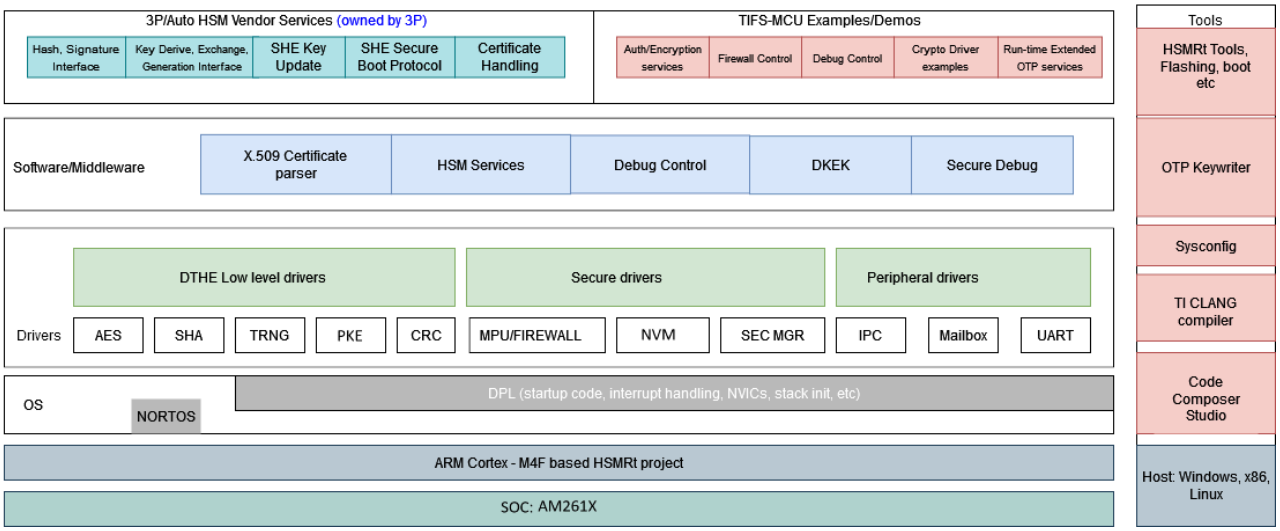

Figure 3. AM261x SW Block Diagram

Table 4. TIFS-MCU Software Components

TIFS-MCU Software Components	Description
OS Kernel	
No RTOS	Contains modules which implement no-RTOS execution environment consisting of timers, ISR, main thread. Allows software on top to run in bare metal mode. *Note - HSM Server is only supported in NORTOS.*
Driver Porting Layer (DPL)	APIs used by drivers to abstract the OS environment. Example, Semaphore, HW interrupts, mutex, clock.

Table 4. TIFS-MCU Software Components (continued)

TIFS-MCU Software Components	Description
Security Device Drivers and Modules	
TIFS-MCU Peripheral Drivers	Device Drivers library and APIs for HSM. *List of SOC Peripheral Driver:* • HSM MBOX and Secure IPC • Crypto Drivers • HSM Flash • Security Manager • Firewall
TIFS-MCU Middle-ware	TIFS-MCU middleware that are supported in TIFS-MCU package *List of Middleware:* • HSM Server • HSM Memory Log • ASN1 Parser and Certificate Parser • Key Derivation • Crypto Interface
TIFS-MCU Services	TIFS-MCU middleware that are supported in TIFS-MCU package *List of HSM Services:* • HSM Get Version Service • HSM Get UID Service • HSM Run Time Debug Authentication Service • HSM Derived KEK Service • HSM Random Number Generate Service • HSM Runtime Firewall Service • HSM Extended OTP Service • HSM Anti Rollback Service • HSM Root of Trust Switch Service • HSM Proc Auth Boot Service (single/streaming) • HSM Key Import Service • HSM OTFA Service
TIFS-MCU Firmware	Out of Box Example implementation of TIFS-MCU firmware with all the mentioned services enabled

Table 4. TIFS-MCU Software Components (continued)

TIFS-MCU Software Components	Description
Examples and Demos	
Examples and Demos	*List of HSM Examples:* • HSM Get Version Example • Debug Authentication Example • Extended OTP Examples • Run-time Firewall Example • Firewall Interrupt Service Example • Anti Rollback Example • Derived KEK Example • RNG Example • Encryption/Decryption Cryptograhic Examples • Hashing Cryptograhic Examples • Asymmetric Cryptograhic Examples
Tools (used on host machine)	
Code Composer Studio (CCS)	IDE used to build projects, debug programs
TI CLANG Compiler Toolchain	CLANG based ARM compiler from TI for ARM M4F, R5F
SysConfig	System configuration tool, used to configure peripherals, pinmux, clocks and generate system initialization code
SDK Tools and Utilities	Additional tools and utilities, like flashing tools, booting tools, CCS loading scripts used with the SDK development flow
OTP Keywriter	OTP Keywriter is used to fuse customer keys into the device and convert HS-FS to HS-SE to establish customer root-of-trust.
TIFS-MCU tools	Tools and scripts to leverage the services provided via

Table 5. HSM Services Supported in 10.02 Release

Services	Description	Examples Available
HSM Get Version Service	HSM GetVersion service is to get the current TIFS-MCU Firmware version	Yes
HSM Get UID Service	When TIFS-MCU Firmware receives a request to GetUID from HSM Server, the UID is copied from secure memory to the output memory location requested by the user.	Yes
HSM Run Time Debug Authentication Service	To unlock the debug port during the run-time, you need an X509 certificate signed with private keys. This service is used to provide the signed certificate to TIFS-MCU Firmware for processing.	Yes
HSM Derived KEK Service	TIFS-MCU provides this service to get a derived KEK based on some input constants. • This key is unique for every unit device and is kept secret. • This key cannot be fetched from hardware in any manner.	Yes
HSM Random Number Generate Service	TIFS-MCU provides this service to get a random number from the given input constants.	Yes
HSM Runtime Firewall Service	TIFS-MCU provides this service to program the system firewalls controlled by HSM only for protection, isolation etc.	Yes
HSM Extended OTP Service	TIFS-MCU provides this service to program general purpose or user defined OTP row programming.	Yes
HSM Anti Rollback Service	TIFS-MCU provides this service to program SW revisions in the eFuses to prevent Anti-Rollback to previous software in the system.	Yes
HSM Root of Trust Switch Service	TIFS-MCU provides this service to change the root of trust switch from the primary key to backup key.	Yes
HSM Proc Auth Boot Service	TIFS-MCU provides the Proc Auth Boot service to authenticate and decrypt the application images signed with root or auxillary keys.	Yes (*part of SBL in MCU+ SDK*)
HSM Key Import Service	TIFS-MCU provides the Key import service to import the Auxillary keys into the system.	Yes (*part of SBL in MCU+ SDK*)
HSM OTFA Service	TIFS-MCU provides the OTFA service to configure the OTFA regions based on root as well as auxiliary keys.	Yes (*part of SBL in MCU+ SDK*)

Table 6. Crypto HW Accelerators and Modes Supported

Crypto Core	Support Available in SW Driver	Examples Available	Specification
AES Encryption and Decryption	• 128,192 and 256 bits Keys • ECB, CBC, CCM, CTR, CFB • One-Shot + Streaming Mode • CPU Polling Mode • EDMA Mode (Polling)	Yes	
AES MAC Generation and Verification	• 128,192 and 256 bits Keys • CCM, CBC-MAC, CMAC • One-Shot + Streaming Mode • CPU Polling Mode • EDMA Mode (Polling)	Yes	
SHA Hasing Algorithm	• SHA256, SHA512 • HMAC SHA-256, HMAC SHA-512 • One-Shot + Streaming Mode • CPU Polling Mode • EDMA Mode (Polling)	Yes	

Table 6. Crypto HW Accelerators and Modes Supported (continued)

Crypto Core	Support Available in SW Driver	Examples Available	Specification
RSA Encryption and Decryption Signing and Verification	• RSA 2048, 3072, 4096 bit • RSA PKCS1_5, PSS2_1 • CPU Polling Mode	RSA PKCS1_5 with 4K only	
RSA Key Generation Service	• RSA 2048, 3072, 4096 bit • CPU Polling Mode	Example with 4096 bit key only *(only for AM261x)*	
ECDSA Signing and Verification	• SECP256, SECP384, SECP521 • BRAINPOOL-P512 • CPU Polling Mode	Yes	
ECDSA Key Generation Service	• SECP256, SECP384, SECP521 • BRAINPOOL-P512 • CPU Polling Mode	Yes	
EDDSA Signing and Verification	• ED25519 • CPU Polling Mode	Yes	
ECDH	• SECP256, SECP384, SECP521 • BRAINPOOL-P512 • CPU Polling Mode	Yes *(only for AM261x)*	

5.1 List of NIST Standards and References

- *Boot time calculator for AM263x, AM263Px, AM261x*
- *FIPS Pub. 197: "Announcing the ADVANCED ENCRYPTION STANDARD (AES)"*
- *[NIST-SP800-38A] "Recommendation for Block Cipher Modes of Operation: Methods and Techniques"*
- *[GCM] The Galois/Counter Mode of Operation (GCM), May 31, 2005*
- *[CCM] "Counter with CBC-MAC (CCM) - AES Mode of Operation"*
- *FIPS Pub. 180-4: Secure Hash Standard, NIST*
- *FIPS Pub. 198-1: The Keyed-Hash Message Authentication Code (HMAC), July, 2008*

6 List of Valid Devices

- AM2631
- AM2631-Q1
- AM2632
- AM2632-Q1
- AM2634
- AM2634-Q1
- AM263P2
- AM263P4
- AM263P4-Q1
- AM2612

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265