Product Overview

F29-TIFS-SDK Product Brief

Software Product Overview

The Hardware Security Manager in F29 devices contains several component blocks designed to achieve system security objectives. These include various memories, the Security Manager, cryptographic accelerator engines, peripheral modules, and the secure mailbox. The host C29 subsystem interfaces with the HSM subsystem to perform the cryptographic operations required for code authentication, secure boot, secure firmware upgrades and encrypted run-time communications.

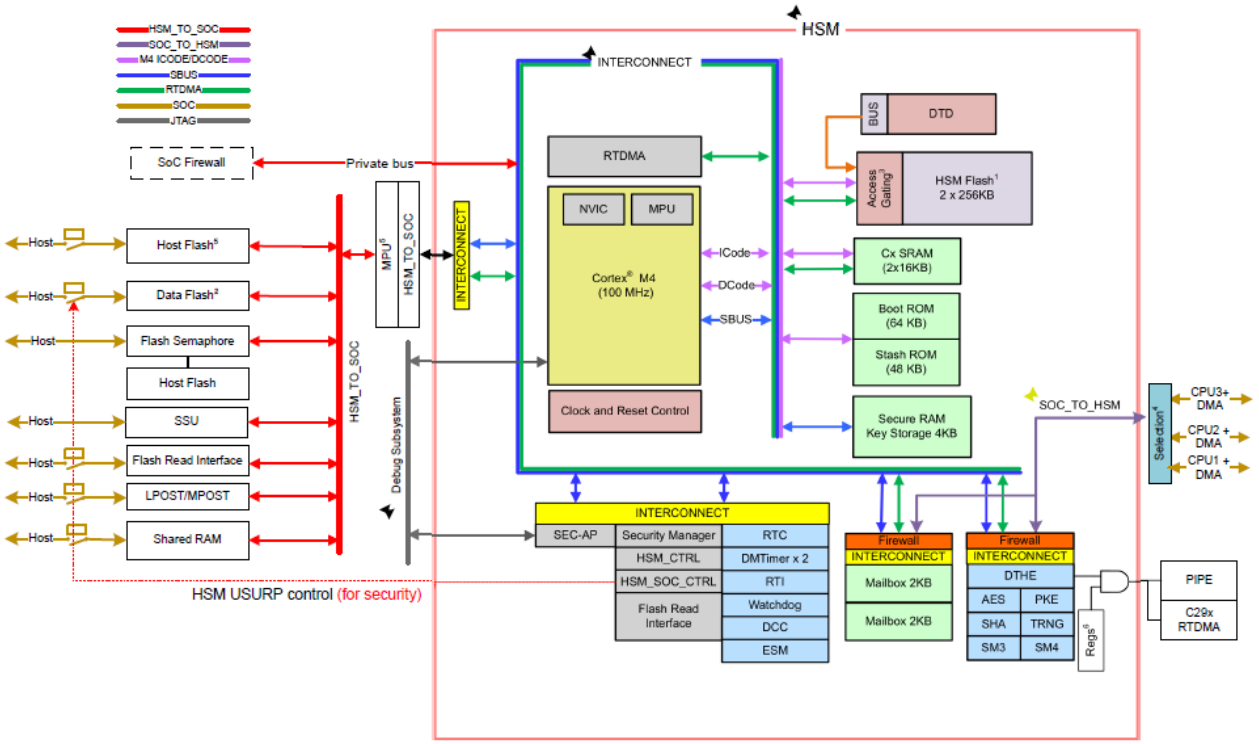

Figure 1. HSM Block Diagram

Security Goals of F29 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 and 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 issues can be avoided.
  - This is also required to prove security to certification entities and customers.

Device Lifecycle and Provisioning Flow

The flow is also explained in *Enabling Cybersecurity for High Performance Real-Time Control Systems*.

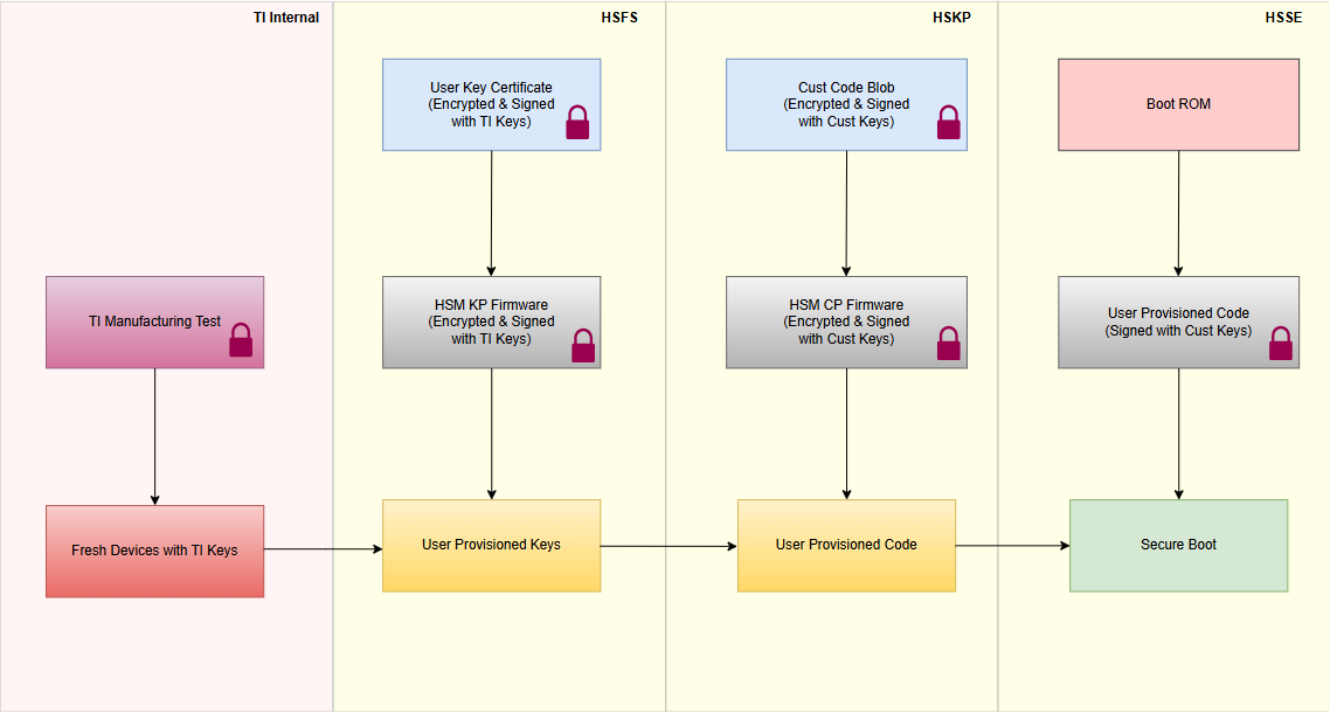

Figure 2. F29x Device Provisioning Flow

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 flash kernel example and host programmer tool. Confidentiality of the user keys is maintained throughout the process that allows users 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 HSKP (Key Provisioned) state.

Stage 3:

Now as the device is in HSKP state, the application code can be programmed into the device. The code provisioning is then encrypted and signed with user keys. This is also securely transmitted using the flash kernel and the host programmer. The user code which is suppose to be provisioned into the HSM or C29 flash is encrypted and signed using the user keys that are available in the device secure storage. Once all the required images for example HSM Code, C29 Code, and Secure Config (SecCfg) are provisioned, the device is converted to HSSE state. HSSE (Security Enforced) life-cyle of device maintains that the code is always a secure boot.

TI Security software components

TI delivers 2 primary software components as part of TIFS-SDK:

- One-Time Programmable (OTP) Key Provisioning Package
- TI-Foundational Security for MCU devices Add-on Package

TI's F29x 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-KP (temporary lifecycle with key provisioned). These provisioning flows are end-to-end secured and can be utilized for non-secure factory floor provisioning.

List of Features Supported by Key Provisioning Flow

This is the support available in 1.01.00 release of F29x-TIFS-SDK.

- Signed Key writer firmware for HSM which accepts x.509 customer keys certificate with all Flash OTP fields configured.
- Supports programming keys at one-pass customer key certificates.
- Supports RSA-4K, ECDSA secp256R1, secp384R1, secp521r1, brainpool512r1 based Key Provisioning.
- Supports UART modes for key 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, SWREV-SSU
  - KEY REV

TI Foundational Software for MCU Devices

What is TIFS-MCU ?

TIFS stands for Texas Instruments Foundational Security for F29x SoCs. TIFS 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 F29-SDK offering for F29x devices like F29H85x. TIFS-MCU enables a bare metal 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.

What is Code Provisioning Firmware in TIFS-MCU ?

Code Provisioning firmware is TI delivered software (including source) which enables secure provisioning of software into the internal flash of the device. This allows users to program HSM as well as C29 applications securely even in non-secure environment.

Table 1. List of Features Supported by Code Provisioning Flow

Features of Code Provisioning Flow	Image Integrity	Bank Mode
HSM Run Time Firmware Provisioning	• RSA-4K with SHA512 • ECDSA secp256R1 with SHA512 • ECDSA secp384R1 with SHA512 • ECDSA secp521R1 with SHA512 • ECDSA brainpool512R1 with SHA512	All Bank Modes
C29 CPU1 Provisioning	• RSA-4K with SHA512 • ECDSA secp256R1 with SHA512 • ECDSA secp384R1 with SHA512 • ECDSA secp521R1 with SHA512 • ECDSA brainpool512R1 with SHA512	All Bank Modes
Secure Config Provisioning	• RSA-4K with SHA512 • ECDSA secp256R1 with SHA512 • ECDSA secp384R1 with SHA512 • ECDSA secp521R1 with SHA512 • ECDSA brainpool512R1 with SHA512	All Bank Modes

Table 2. List of Software Deliverables for Secure Code Provisioning Flow

List of Software Components	Software Type	OPN	Delivery Location	Source Available in 1.01.00
UART Flash Kernel	Example	F29H85x-SDK	ti.com	Yes
Host Programmer	Tool for: • Windows • Linux • MacOS	F29H85x-SDK	ti.com	Yes
OTP Key Writer Certificate Generation	Python tool	F29H85x-TIFS-SDK	Secure Resources	Yes
HSM KP firmware	Encrypted Firmware	F29H85x-TIFS-SDK	Secure Resources	No
HSM CP firmware	Example	F29H85x-TIFS-SDK	Secure Resources	Yes
Code signing tool	Python tool	F29H85x-TIFS-SDK	Secure Resources	Yes

What is available in the TIFS-MCU SDK ?

TIFS-MCU SDK offers out-of-box services and example HSM firmware showcasing use cases which execute in HSM subsystem.

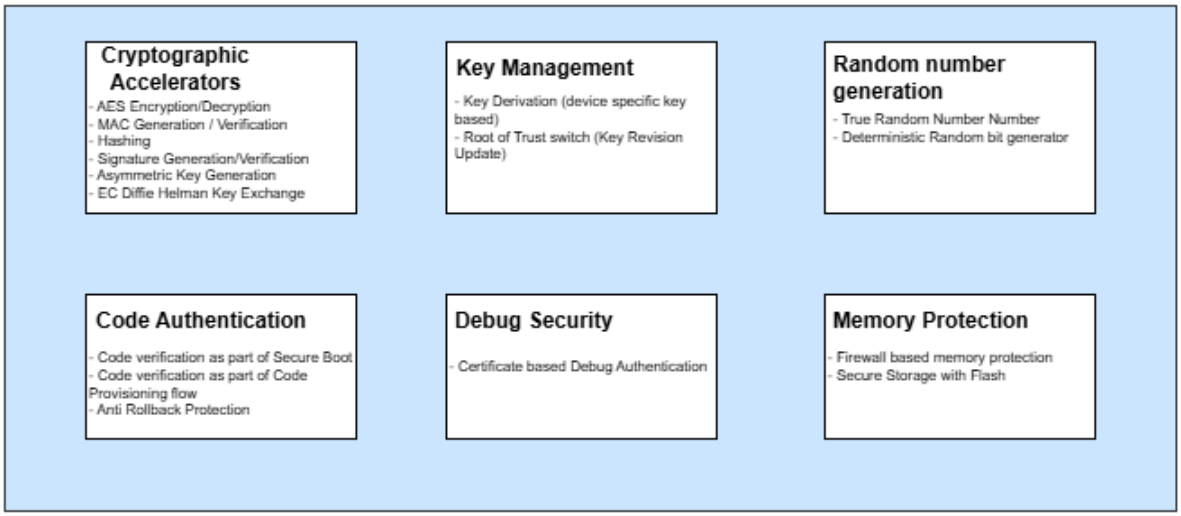

Figure 3. Native Services Provided by TIFS-MCU

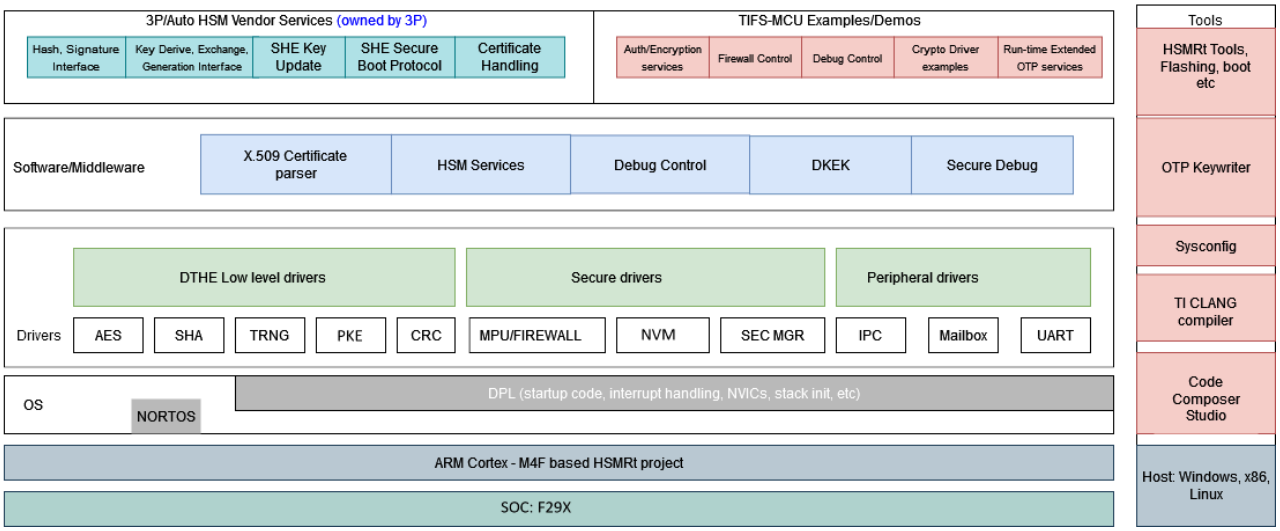

Figure 4. Software Architecture Block Diagram of TIFS-MCU

Table 3. 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.
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, HSM FRI • Security Manager • Firewall
TIFS-MCU Middle-ware	TIFS-MCU middle-ware that are supported in TIFS-MCU package *List of Middle-ware:* • HSM Server • HSM Memory Log • ASN1 Parser and Certificate Parser • Key Derivation • Crypto Interface
TIFS-MCU Services	TIFS-MCU middle-ware 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 Extended OTP Services • HSM Anti Rollback Services • HSM Root of Trust Switching Services • HSM Secure Firmware Update Services
TIFS-MCU Firmware	Out of Box Example implementation of TIFS-MCU firmware with all the mentioned services enabled
Examples and Demos	
Examples and Demos	*List of HSM Examples:* • Combined Services Demo showcasing all the HSM services • Boot Manager demonstrating Firmware Update in Flash Boot Mode • Encryption/Decryption Cryptographic Examples • Hashing Cryptographic Examples • Asymmetric Cryptographic 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
TI C29-CGT Toolchain	CLANG based C29 compiler from TI for C29 CPU
SysConfig	System configuration tool, used to configure peripherals, pinmux, clocks and generate system initialization code

Table 3. TIFS-MCU Software Components (continued)

TIFS-MCU Software Components	Description
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-KP to establish customer root-of-trust.
TIFS-MCU tools	Tools and scripts to leverage the services provided via TIFS-MCU.

Table 4. HSM Services Supported

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 Extended OTP Service	TIFS-MCU provides services to program General purpose OTP regions which are an array of otp flash bits that can be defined for user usage models.	Yes
HSM Anti Rollback Service	TIFS-MCU provides anti-rollback services that help prevent booting older software images. The device has OTP fields to hold software revision for SBL, HSMRt, SECCFG and Application Images.	Yes
HSM Root of Trust Switch Service	TIFS-MCU provides Secure RoT Switching to switch to backup keys from the secondary keys. There are two Root of Trust Keys that are present in the SoC: Secondary (SMPK/SMEK) and Backup (BMPK/BMEK). If the secondary key is compromised, the attacker can take the control of the entire SoC.	Yes
HSM Firmware Update Service	TIFS-MCU provides secure firmware update service which consists of certificate authentication flow, verification against the root of trust keys and maintain the image integrity of the image throughout the flow.	Yes

Table 5. Crypto HW Accelerators and Modes Supported

Crypto Core	Support Available in SW Driver	Examples Available	Specification
AESEncryption and Decryption	• 128,192 and 256 bits Keys • ECB, CBC, CCM, CTR, CFB • One-Shot + Streaming Mode • CPU Polling Mode	Yes	
AESMAC Generation and Verification	• 128,192 and 256 bits Keys • CCM, CBC-MAC, CMAC • One-Shot + Streaming Mode • CPU Polling Mode	Yes	

Table 5. Crypto HW Accelerators and Modes Supported (continued)

Crypto Core	Support Available in SW Driver	Examples Available	Specification
SHAHasing Algorithm	• SHA256, SHA512 • HMAC SHA-256, HMAC SHA-512 • One-Shot + Streaming Mode • CPU Polling Mode	Yes	
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	
ECDSA Signing and Verification	• SECP256, SECP384, SECP521 • BRAINPOOL-P512 • CPU Polling Mode	Yes	
EDDSA Signing and Verification	• ED25519 • CPU Polling Mode	Yes	
ECDH Diffie Helman Key Exchange	• SECP256, SECP384, SECP521 • BRAINPOOL-P512 • CPU Polling Mode	Yes	

List of Valid Devices

- F29H85x

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