Application Brief

Understanding Security Features for C2000 Real-Time Control MCUs

Device/Family Description

The TI C2000™ microcontrollers (MCUs) are designed for real-time control applications in both industrial and automotive spaces. The TI C2000 microcontrollers are built with real-time control in mind. All C2000 MCUs are based on the 32-bit C28x MCU core processor with speeds from 40 MHz to 200 MHz. With tightly coupled analog peripherals such as analog-to-digital converters (ADCs), comparators and fast response digital stimuli like PWMs, there is a compelling reason to use the C2000 MCUs in a real-time control application.

TI Embedded Security Portfolio

Table 1. Security Enablers

C2000 MCU Series	Security Enablers	Detailed Security Features
TMS320F28003x TMS320F28002x TMS320F2838x TMS320F28004x TMS320F2837xD TMS320F2837xS TMS320F2807x TMS320F2806x TMS320F2805x TMS320F2803x TMS320F2802x TMS320F2833x/23x F28M3x	Device identify	Unique Identification (UID) Number: Ability for user to enable mechanisms for device identification in communications, seed for data integrity algorithms, initialization vector for authentication and encryption or decryption, or to protect against code cloning
	Software IP protection	Code Security Module (CSM): Ability for user to block unauthorized access or programming of firmware stored in on-chip memories
	Debug security	Emulation Code Security Logic (ECSL) via CSM: Ability for user to enable full debug access to memory via a password

Table 2. Latest Security Enablers

C2000 MCU Series	Security Enablers	Detailed Security Features
TMS320F28003x TMS320F2838x	Additional Debug Security	JTAGLOCK: Ability to block emulation of the device; unlockable via password
	Cryptographic acceleration	Hardware Advanced Encryption Standard (AES 128/192/2566 bit) engine to boost performance
	Secure Boot	Option to enable AES128 Cipher-based Message Authentication Code (CMAC) to to pre-authenticate first 16KB of flash prior to transferring code execution.

Security Problem Targeted: Typical Threats/ Security Measures

As in any microcontroller, a good portion of the R&D investment at the user level goes into the firmware development. As such, intellectual property (IP) housed in a product's firmware can provide key competitive advantages for a business in the marketplace and is at a high risk of theft. It is straightforward enough to do a visual component teardown of a system for purposes of copying an end product, but protection of the firmware running on the MCU prevents full duplication of the working system.

Another scenario that is increasingly common is co-development of the firmware. Many times certain system firmware is outside the main engineering team, perhaps even outside the company. In these situations, one party sometimes wants to keep their firmware private, while still allowing the second to develop and test a portion of the program on the same system. Such scenarios are typically not covered by traditional run-time software protections and require protection while the MCU is in a debug state.

This is especially common in automotive applications where there may be multiple companies involved in producing and debugging firmware in a highly connected system. These types of threats can be addressed by security enablers on most C2000 devices.

Security Implementation

When a new device is shipped from TI, it arrives in a completely unlocked state. After security protocols are enabled by you, a locked zone will only be accessible by code that also exists in the same zone. Dedicated, unlocked memory exists so that data can be transferred between zones if needed. In addition to this fundamental implementation, there are other options or layers that can be selectively enabled:

1. Selection of memory blocks to be protected:

   In many cases, not all the memory, either volatile or nonvolatile, will need to be locked. This is true for certain pieces of firmware shared across different sub-systems or that contain non-proprietary IP.
2. Zone ownership (DCSM only):

   In addition to protecting various blocks of memory, there are two zones in each DCSM implementation. Once the memories are allocated for protection, the next step is deciding which of these zones will have control over the selected memories. However, if there is no need for code protection between developers on the same device, a single-zone configuration can be used.
3. Execute-only protection (DCSM only):

   If a region will be used only for execution, rather than internal data storage, the programmer can enable “execute-only protection” to block any read access (even from the same region/zone) for added security.
4. CPU protection (DCSM and F2837x/07x only):

   Debug access to the core processing unit (CPU) registers is also blocked if the DCSM detects code executing from any locked region.
5. Emulation Code Security Logic (ECSL):

   Even using the above measures, it may still be desirable to restrict an emulation connection if the MCU is executing from a locked region. This may be temporarily disabled during a debug session using a password.
6. Unique Identification (UID):

   By using a UID number provided on each device techniques can be implemented to further allow software to only run on known devices. For more information, see *C2000™ Unique Device Number*.
7. JTAGLOCK:

   The JTAG (emulator) interface can be disabled and protected with a user chosen password. This helps ensure only authorized individuals can view/ debug the application.
8. AES acceleration:

   One of the world’s most common encryption algorithms is known for its speed and simplicity. Even given that, a software implementation of AES in a real time microcontroller is comparatively slow to the demands of a real time control system. The hardware accelerator vastly improves the cycle time of processing cryptographic messages while freeing up the CPU bandwidth in the process. Several different modes and bit sizes are available.
9. Secure Boot:

   As another layer of firmware protection, secure boot can optionally be enabled to run at boot before turning execution over to the user flash code. Along with the programming protection built into the security logic, this helps ensure the code that runs on the device is authentic. The algorithm used is an AES128 CMAC algorithm. Tools are available to embed the required MAC value into the final code image. For more information, see Secure Boot on C2000 Device.

Additional Resources

While security risks can take many forms across end applications, IP/ firmware protection is a threat common to most systems. The C2000 MCU family can enable our customers to address these concerns through flexible features for multi-development environments. For additional information on the C2000 MCU security techniques, see the C2000 MCU Funcational Safety page. For specific device information refer to the technical reference manual which can be found beneath the data sheet and Errata in any C2000 MCU product folder.

Note

Security is hard, TI makes it easier.

For more information about TI's Embedded Security Solutions, visit https://www.ti.com/technologies/security/overview.html.

Trademarks

C2000™ is a trademark of Texas Instruments.
All trademarks are the property of their respective owners.

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