Technical White Paper

解析嵌入式系統的後量子密碼學

Walter Schnoor

摘要

什麼是 PQC？

後量子密碼學 (PQC) 是一類加密基元，旨在抵禦同時擁有傳統計算能力和加密相關量子計算能力的攻擊者。

目錄

1 簡介 2
2 嵌入式系統的安全目標 2
3 加密基元 2
4 CRQC 的潛在影響 2
5 未來的新演算法與標準 3
6 CNSA 合規性 3
7 對稱式加密和雜湊函數 3
8 PQC 數位簽章：ML-DSA 3
9 是否需要硬體加速？ 4
10 是否需要預先雜湊？ 6
11 NIST 選擇的其他 PQC 數位簽章方案 6
12 有狀態數位簽章方案 7
13 何為金鑰協議？ 7
14 讓嵌入式系統未來無憂 8
15 參考資料 9

註冊商標

Arm® and Cortex® are registered trademarks of Arm Limited.
所有商标均为其各自所有者的财产。

English Document: STDA060
Copyright © 2026 Texas Instruments Incorporated

1 簡介

儘管後量子密碼學聽起來可能很神祕，但讓您的嵌入式系統防範未來具備量子能力的攻擊者，這並不一定很神祕。這是因為雖然 PQC 改變了用於實現安全性的底層演算法，但不會改變您的高階安全目標。換句話說，「要做什麼」沒變，變的是「如何做」。本白皮書說明 PQC 如何契合嵌入式系統中的常見安全目標、與 PQC 相關的業界標準，以及對使用微控制器的資源受限嵌入式系統的影響，包括對延遲、記憶體和通訊時間的意義。

2 嵌入式系統的安全目標

為了解 PQC 對嵌入式系統的影響，最好從安全目標本身開始。許多嵌入式系統共享一套常見的安全目標，包括：

- 安全啟動，可確保在處理器上執行的軟體或韌體為正版，換句話說，系統製造商已授權該軟體或韌體。
- 安全更新，為保護智慧財產權，將新的軟體和韌體更新保密。
- 證明，使裝置或系統能夠證明其知悉某項秘密，從而向雲端伺服器或系統中的另一個模組證明自身真實且可信任。
- 安全通訊，可維護嵌入式系統中裝置之間或嵌入式系統與遠端伺服器之間訊息的機密性、完整性和真實性。

在加密系統中部署加密基元可以實現這些安全目標。例如，電子控制單元中的微控制器可透過納入不可變更的信任根韌體來實作安全啟動，該韌體使用數位簽章演算法在允許應用軟體於 CPU 上執行之前對其進行驗證，從而防止在 CPU 上執行未經授權的程式碼。如果底層基元在未來被入侵，則安全啟動等安全目標可能連帶被入侵。

3 加密基元

加密基元是建構加密系統所需的加密學核心建構區塊。加密基元的一個範例是對稱區塊密碼，其使用單一金鑰將明文加密為密文，並將密文解密回明文。另一個範例是數位簽章演算法，其可以使用兩個金鑰驗證訊息的真實性：一個是私密金鑰，用於將訊息簽章為真實；另一個是公開金鑰，用於驗證訊息是否由匹配的私密金鑰簽章。這些基元的安全性源自所謂的困難性假設。

當今最廣泛使用的數位簽章演算法之一是橢圓曲線數位簽章演算法 (ECDSA)，其安全性基於橢圓曲線離散對數問題 (ECDLP) 的困難性。目前，尚無已知演算法可以有效地使用當今的電腦來解析 ECDLP。因此，ECDLP 被認為具有足夠的「困難性」，可賴以提供安全性。然而，量子電腦則是另一回事。

4 CRQC 的潛在影響

傳統運算是指當今的電腦，其以一系列離散指令執行既定操作。在某些情況下可以並行處理這些指令，但運算仍受限於循序操作或多組循序操作並行。

另一方面，量子電腦可以利用可同時為「0」和「1」的資料位元，使其能以當今電腦無法做到的方式同時搜尋解決方案空間 [1]。

密碼學相關量子電腦 (CRQC) 理論上能透過使用已知演算法解析 ECDLP。如果這種量子電腦存在，將打破使當今流行的加密基元 (如 ECDSA) 安全的困難性假設。攻擊者可能會破解該加密基元，從而破解該加密基元支援的安全目標。攻擊 ECDSA 的主要目標是復原用於簽章新訊息以證明其為真實的私密金鑰，從而使攻擊者隨後能使用該私密金鑰簽署詐欺訊息，產生有效簽章。這需要在給定公開金鑰 P 和公開產生器點 G (其中，橢圓曲線上 $P = kG$) 的情況下找出私密金鑰值 k。對於傳統電腦，在給定公開值 P 與 G 的情況下復原 k 很困難 (已知方法以指數時間執行) 。使用 CRQC，則可執行 Shor 演算法：這是一種量子演算法，可將攻擊從指數時間降至多項式時間，從而使執行變得實用。在撰寫本文時，尚不存在規模大且穩定、足以執行 Shor 演算法等可破解 ECDSA 熱門實作 (例如使用國家標準與技術研究所 [NIST] P-256 曲線的 ECDSA) 之演算法的量子電腦，但將來可能會出現。

雖然沒人知道何時會出現足以破解當今傳統加密基元的強大量子電腦 [1]，但許多行業都開始保護其產品免受未來威脅，原因如下：

- 構建受信任的新加密技術需要時間，將其整合至產品中也需要時間。
- 許多嵌入式系統的現場使用壽命已延長 10 年或更久。現在因應 CRQC 可能問世，可降低未來失去安全性的風險。
- 許多市場正在推動監管，尤其是企業和資料中心市場。

因此，我們看到更多嵌入式系統設計人員轉為採用 PQC 加密基元，以達成安全目標。

English Document: STDA060
Copyright © 2026 Texas Instruments Incorporated

5 未來的新演算法與標準

大多數嵌入式系統的安全目標與用於實現這些目標的加密基元（傳統或量子）無關。採用 PQC 的動機並非改變嵌入式系統的安全目標；而是改變用於實現安全目標的底層算術，從而使困難性問題能夠抵禦具備 CRQC 能力的攻擊者。

6 CNSA 合規性

在美國，用於國家安全系統的產品必須符合國家安全局 (NSA) 商業國家安全演算法 (CNSA) 套件要求。2022 年，NSA 將 CNSA 套件從 1.0 版更新為 2.0 版；CNSA 2.0 的主要目的之一是納入 PQC。新部署的 CNSA 套件必須在 2027 年 1 月 1 日前符合 2.0 版本，而已部署的系統必須在 2030 年 12 月 31 日前符合或替換為符合 CNSA 2.0 標準的系統。

表 6-1 列出了 CNSA 1.0 核准的加密基元與 CNSA 2.0 核准的所需加密基元之對比。

表 6-1. CNSA 1.0 與 CNSA 2.0 核准的加密基元對比

應用用途	加密基元	CNSA 1.0 核准	CNSA 2.0 核准 [2]
一般用途密碼學	對稱式加密	聯邦資訊處理標準 (FIPS) 出版物 (PUB) 197 (僅限進階加密標準 [AES]-256)	
	安全雜湊	FIPS PUB 180-4 (僅限安全雜湊演算法 [SHA]-384)	FIPS PUB 180-4 (SHA-384 或 SHA-512)
	數位簽章	FIPS PUB 186-4 (ECDSA P-384 或 Rivest、Shamir 和 Adleman 3,072 位元模數)	FIPS PUB 204 (僅限基於模組晶格的數位簽章標準 [ML-DSA]-87)
	金鑰協議	橢圓曲線 Diffie-Hellman P-384 或有限體 Diffie-Hellman (3,072 位元模數)	FIPS PUB 203 (僅限基於模組晶格的金鑰封裝機制 [ML-KEM]-1,024)
僅限韌體或軟體簽章	數位簽章	–	NIST 特別出版物 (SP) 800-208 (Leighton-Micali 簽章 [LMS] 或擴展 Merkle 簽章方案 [XMSS] [1])
僅限硬體完整性檢查	安全雜湊	–	FIPS PUB 202 (SHA-3-384 或 SHA-3-512)

7 對稱式加密和雜湊函數

CNSA 2.0 並未顯著改變對稱式加密基元和安全雜湊函數的最低要求。例如，CNSA 2.0 保留了與 CNSA 1.0 中使用的對稱式加密 (AES-256) 相同的演算法和安全級別。這是因為 CRQC 目前對 AES-256 並未構成重大威脅。儘管存在可以降低 AES 安全性的量子演算法，但 CNSA 1.0 和 2.0 (256 位元) 中的金鑰大小選擇足夠高，使得該基元仍保持安全。以 Grover 演算法為例。Grover 演算法是一種量子演算法，可將 AES 的暴力攻擊時間從 2k 降至 2k/2，其中 k 為金鑰大小（單位為位元）。在最佳情況下，使用 Grover 演算法時，AES-256 的安全強度將從 2^{256} 降低至 2^{128}，仍被認為安全。相較之下，使用 AES-128 的設計，其安全強度可能從 2^{128} 降低至 2^{64}，可能易受攻擊。因此，在目前使用 AES-128 的應用中，建議改用 AES-256，從而抵消量子電腦的優勢。

CNSA 2.0 中一些最重要的變更與數位簽章和金鑰協議有關。

8 PQC 數位簽章：ML-DSA

對於通用數位簽章（包括韌體和軟體更新簽章），CNSA 2.0 要求使用 ML-DSA。ML-DSA 的安全性依賴模組容錯學習困難性問題，並被認為可以抵禦同時具有傳統運算和量子運算能力的攻擊者。ML-DSA 的設計基於提交給 NIST 的 Crystals-Dilithium 簽章方案，NIST 在經過三輪分析與評估後，選擇將其標準化為 FIPS 204 [4]。

在使用 ECDSA 進行數位簽章的嵌入式應用（包括安全啟動、證明和其他簽章使用案例）中，ML-DSA 提供了具有類似程式設計介面的後量子安全替代方案。ML-DSA 還具有可擴展性；它支援三種不同的安全級別 (ML-DSA-44、ML-DSA-65 和 ML-DSA-87)，每種級別在金鑰與簽章大小、運算延遲、運算靜態隨機存取記憶體 (SRAM) 空間及安全性之間有所取捨。CNSA 2.0 強制要求採用最高安全級別：ML-DSA-87。

對嵌入式系統開發人員而言，ML-DSA 與 ECDSA 的顯著差異如下：

Copyright © 2026 Texas Instruments Incorporated

- ML-DSA 的金鑰與簽章大小遠大於 ECDSA。ML-DSA-87 的簽章大小 (4,627 位元組) 比 ECDSA P-384 (96 位元組) 大 48 倍以上，在執行證明功能或安裝新軟體和韌體更新時，需要進行更多快閃記憶體和 SRAM 配置，並需要更長的線上傳輸時間。
- 在比較基於 CPU 的實施時，ML-DSA 驗證往往比 ECDSA 驗證更快，從而縮短驗證時間。例如，ML-DSA-87 簽章驗證操作可在 NIST P-384 曲線上 ECDSA 的十分之一時間內執行，從而顯著縮短微控制器的啟動時間。
- 與 ECDSA 相比，ML-DSA 實作通常需要更多 SRAM 來執行必要的運算，並在工作記憶體和執行延遲之間有不同的取捨。

表 8-1 比較了每個 ML-DSA 安全級別的傳統運算簽章 (ECDSA P-384) 與後量子運算簽章 (ML-DSA)。

表 8-1. ECDSA 參數大小和性能與 ML-DSA 的比較

參數	ECDSA	ML-DSA-44	ML-DSA-65	ML-DSA-87
安全等級	P-384	類別 2	類別 3	類別 5
金鑰 (位元組)	48	2,570	4,032	4,896
公鑰 (位元組)	96	1,312	1,952	2,592
簽章 (位元組)	96	2,420	3,309	4,627
簽章線上傳輸時間 (秒) (1)	0.009	0.22	0.30	0.42
金鑰產生時間 (秒) (2)	1.12	0.084	0.14	0.23
簽章時間 (秒) (2)	1.23	≅0.4	≅0.8	≅1.6
驗證時間 (秒) (2)	**2.26**	**0.10**	**0.16**	**0.26**
驗證所需的 SRAM (千位元)(2)	2.2	21.9	29.1	37.6

(1) 以每位元組 9 位元、100kHz 速率傳輸簽章的時間
(2) 測試環境為一款德州儀器 (TI) 80MHz Arm® Cortex®-M0+ 微控制器，執行 TI MSP 密碼編譯函式庫

9 是否需要硬體加速？

對於建構使用橢圓曲線密碼學之嵌入式系統的開發人員，若應用對延遲敏感，則可以使用具有加速功能的微控制器進行橢圓曲線運算。但這可能會導致所選微控制器比應用實際所需更為昂貴。

例如，在挑戰回應應用中，為挑戰回應機制計算 ECDSA P-384 簽章，在 80MHz Arm Cortex-M0+ 處理器上可能需要超過 2 秒，遠長於以 100Kbps (典型速率，假設使用標準 I^2C 或 115200 序列通用非同步接收器與傳送器等通訊方式) 在線上傳輸所產生簽章所需的 6ms。

當轉向採用 PQC 時，較大的數位簽章和較短的計算時間會顯著改變這種動態。例如，傳輸簽章所需的線上時間增加了 48 倍，從大約 9 ms (對於 ECDSA P-384) 增加至約 420 ms (對於 ML-DSA-87)。簽章驗證運算時間從約 2,200ms (對於 ECDSA P-384) 減少至約 260 ms，這幾乎是線上傳輸簽章所需時間的一半。因此，諸如證明等應用往往更受通訊時間而非運算時間限制，使得基於 CPU 的實作更具經濟效益。圖 9-1 比較了 Arm Cortex-M0+ 和 Arm Cortex-M33 目標 MCU 上各安全級別的驗證性能。

Copyright © 2026 Texas Instruments Incorporated

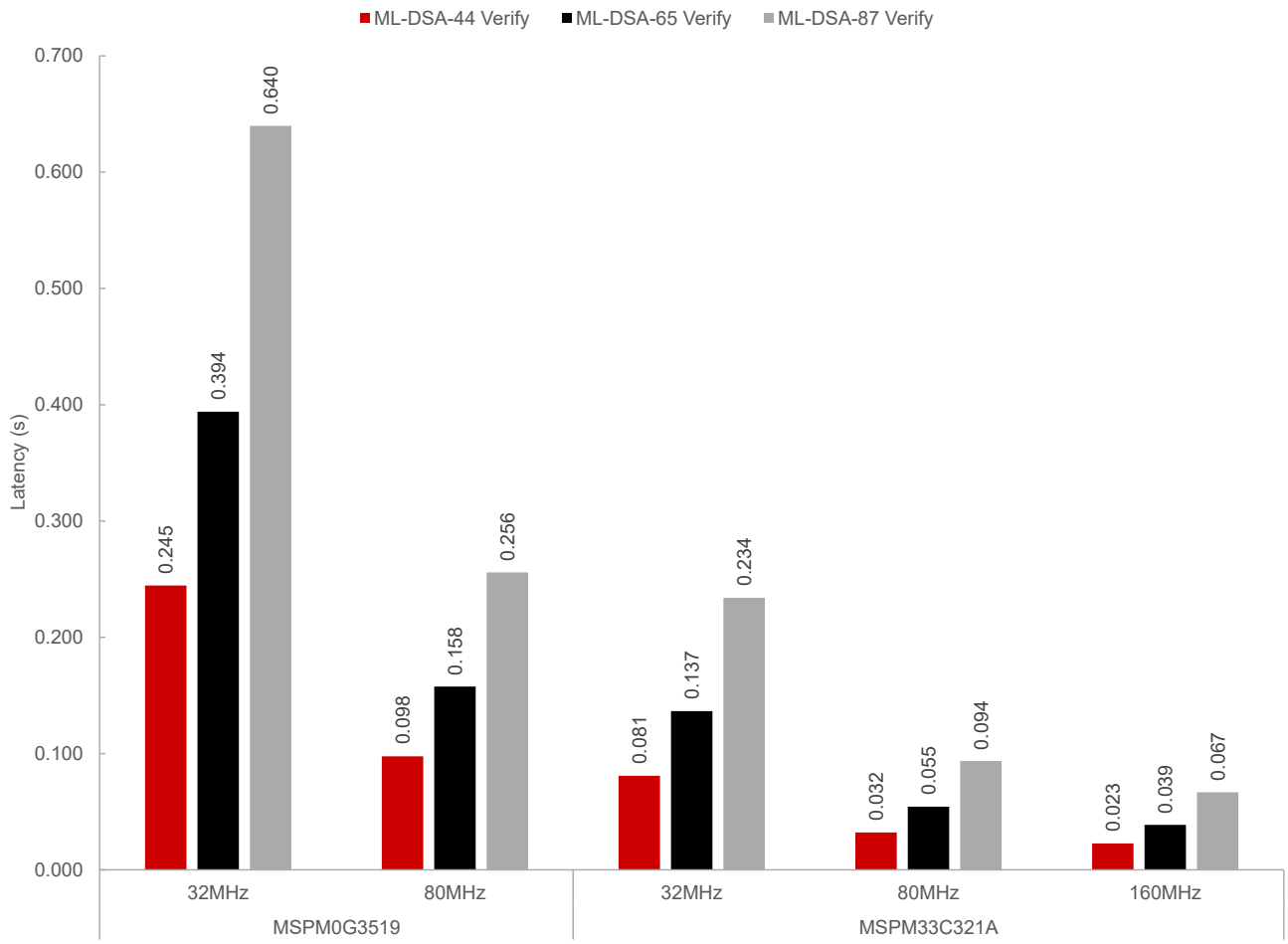


图 9-1. TI MSPM0、MSPM33 微控制器的 ML-DSA 驗證性能

Copyright © 2026 Texas Instruments Incorporated

10 是否需要預先雜湊？

在安全啟動實作中，常見做法是先使用 SHA-256 或 SHA-384 等安全雜湊函數，對應用軟體與韌體映像進行雜湊，以建立固定大小的訊息代表值，然後透過 ECDSA 進行簽章和驗證。ML-DSA 消除了此「預先雜湊」步驟。事實上，ML-DSA 的設計者傾向於該方案的使用者將數位簽章的整個訊息內容直接傳入簽章演算法本身。

在 FIPS PUB 204 中，此過程稱為「純粹」模式。FIPS PUB 204 還定義了稱為 HashML-DSA 的預先雜湊模式，其包含有關計算中用於產生簽章（從而將雜湊函數綁定至簽章）之雜湊函數的相關資訊。

除了 FIPS 204 之外，還可以採用第三種選項：先對訊息進行預先雜湊，但在純粹模式下使用 ML-DSA 簽章該雜湊值，如同將訊息直接傳入 ML-DSA 一樣，而非使用 FIPS 204 的預先雜湊 HashML-DSA 模式。此做法省去了雜湊函數對簽章的綁定，但它與當今 ECDSA 與 SHA 的使用方式更具可比性。

在產生和驗證數位簽章時，ML-DSA 在內部使用 Keccak 函數。Keccak 是 SHA-3 函數和 SHA 與 Keccak (Shake) 可擴展輸出函數內部使用的核心函數。ML-DSA 使用 Keccak 的方式有很多種，包括訊息的安全摘要和驗證過程中公開金鑰的確定性擴展。

這一點很重要，因為在資源受限的系統上，納入 ML-DSA 意味著同時納入 Keccak，於相同程式碼佔用空間內提供安全雜湊函數。因此，使用純粹 ML-DSA 運算模式意味著您無需放棄額外的非揮發性程式碼記憶體來儲存 SHA-384 或 SHA-512 等額外雜湊函數。此外，Keccak 還可用於將 SHA-3 作為通用雜湊函數，用於應用中的其他目的。

目前，符合 CNSA 2.0 的系統可以使用經核准的安全雜湊函數 (SHA-384 或 SHA-512) 對訊息進行預先雜湊，但不能使用 FIPS PUB 204 所規定的 ML-DSA 預先雜湊模式 (HashML-DSA)；相反，FIPS PUB 204 純粹模式為預設模式。

11 NIST 選擇的其他 PQC 數位簽章方案

除了 ML-DSA 之外，NIST 還選擇了兩種後量子運算數位簽章方案進行標準化：

- 基於 Sphincs+ 簽章方案的無狀態雜湊數位簽章標準 (SLH-DSA)，標準化為 FIPS PUB 205。
- 基於 N 次截斷多項式環單位晶格的快速傅立葉轉換數位簽章演算法 (FN-DSA) (或稱 Falcon)，目前正在標準化為 FIPS PUB 206。

SLH-DSA 是一種雜湊式簽章方案。與 ML-DSA 一樣，它是無狀態的。與 ML-DSA 不同，較高安全等級的簽章具有非常大的訊號大小（數十千位元），這限制了 SLH-DSA 在嵌入式應用（特別是基於微控制器的應用）中的適用性。目前，CNSA 2.0 不允許基於 SLH-DSA 的簽章，即使 NIST 已將 SLH-DSA 標準化。

FN-DSA 承諾提供比 ML-DSA 和 SLH-DSA 更小的簽章，但代價是複雜度。FN-DSA 依賴於浮點運算，眾所周知，浮點運算在不同處理器架構中很難以恆定時間實作。此外，並非所有嵌入式架構都包含浮點單元，因此喪失性能優勢。目前，CNSA 2.0 不允許使用 FN-DSA [3]，NIST 尚未將 FN-DSA 標準化。

由於這些原因，業界大致正朝向以 ML-DSA 作為事實上的數位簽章方案，因為它在性能、大小和安全性方面取得了良好的平衡。

English Document: STDA060
Copyright © 2026 Texas Instruments Incorporated

12 有狀態數位簽章方案

ML-DSA 和所討論的方案均為無狀態方案。在對給定私密金鑰進行簽章時，無需保存狀態，這是嵌入式應用追求的一種品質特性，尤其是在資源受限環境中正常運作期間頻繁進行挑戰-回應簽章的證明使用案例。

在標準化無狀態方案之前，多種後量子安全的有狀態簽章方案為特定使用案例（例如軟體與韌體映像簽章 [安全啟動]）提供選項，包括 NIST SP 800-208 中標準化的 LMS 和 XMSS。

這些方案提供的較小公開金鑰 (小於 ML-DSA) 對實作安全啟動的嵌入式系統有利，因為在驗證新韌體時，需要較少的嵌入式非揮發性記憶體（通常為快閃記憶體）。簽章操作確實需要追蹤給定私密簽章金鑰的狀態，以確保私密金鑰的保密性。對於韌體和軟體更新而言，這或許可以接受，因為可以在不頻繁更新韌體和軟體映像的受控 IT 環境中維護簽章硬體安全模組的狀態。然而，在嵌入式環境中，這往往不可接受。

因此，LMS 和 XMSS 在符合 CNSA 2.0 的系統中允許使用，但僅限於韌體和軟體數位簽章；不能用於證明等其他使用案例。相較之下，ML-DSA 可用於所有數位簽章使用案例，包括安全啟動和證明。

13 何為金鑰協議？

金鑰協議是透過不安全的通路在兩個實體之間建立共享密鑰的必要步驟。與數位簽章一樣，依賴於有限體 Diffie-Hellman (FFDH) 或橢圓曲線 Diffie-Hellman (ECDH) 等當今熱門基元的金鑰協議演算法，同樣易受使用 Shor 演算法的量子攻擊。為解決這一問題，NIST 在 FIPS 203 中標準化了 ML-KEM（模組晶格金鑰封裝機制）。ML-KEM 是一種後量子安全金鑰封裝機制 (KEM)，設計用於當今使用 FFDH 或 ECDH 的情境。

與從握手雙方隨機性衍生共享密鑰的 Diffie-Hellman 式設計不同，KEM 具有金鑰產生步驟、封裝步驟和解封裝步驟。在用戶端-伺服器模型中，使用 ML-KEM 達成金鑰協議的流程如下：

1. 首先由伺服器產生金鑰對。封裝（或「公開」）金鑰分發給用戶端。解封裝（或「私密」）金鑰由伺服器保留。
2. 當用戶端想與伺服器建立安全通道並議定對稱金鑰時，其基於隨機種子執行 ML-KEM 封裝操作，從中獲取密文和共享密鑰。然後，將密文傳送給伺服器。
3. 伺服器接收密文並執行解封裝操作，得出與用戶端相同的共享密鑰。
4. 伺服器和用戶端現在可使用共享密鑰進行通訊。

ML-KEM 在軟體中可能表現出色，在 160MHz Cortex-M33 MCU 上封裝時間低至 7ms。图 13-1 比較了 Arm Cortex-M0+ 和 Arm Cortex-M33 目標 MCU 上各安全級別的封裝性能。

English Document: STDA060
Copyright © 2026 Texas Instruments Incorporated

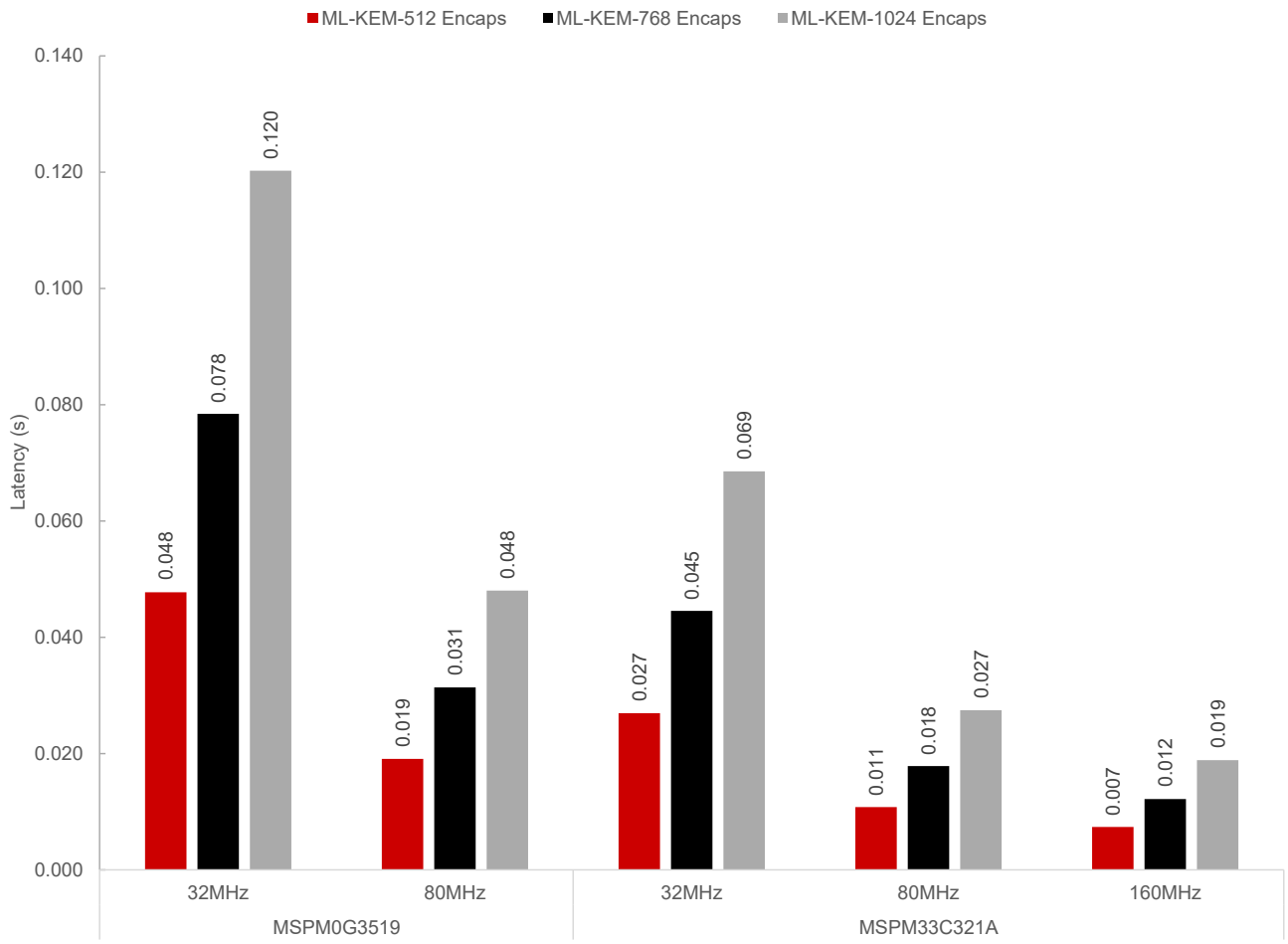


图 13-1. TI MSPM0、MSPM33 微控制器的 ML-KEM 封裝性能

ML-KEM 除了設計為量子安全之外，還具有其他必要的安全特性。值得注意的是，它支援 IND-CCA2（在所選的密文攻擊下無法區分）。這表示即使攻擊者可在用戶端至伺服器的傳輸過程中（封裝與解封裝之間）操縱密文，攻擊者也不會獲得優勢或額外知識，因為解封裝的輸出始終與隨機輸出無法區分。此外，ML-KEM 的輸出可以直接用作對稱式加密金鑰，從而簡化嵌入式應用，並可能消除額外金鑰衍生步驟的需求。

14 讓嵌入式系統未來無憂

設計人員需要考量未來攻擊者可能擁有一台具有足夠能力的量子電腦來攻擊傳統加密基元的可能性，這些基元用於提供基本安全目標，包括安全啟動、證明及安全通訊。透過 NIST 和 NSA，美國已建立一套框架，用於選出即使面對具備此類能力的攻擊者，仍被認為安全的加密基元。

嵌入式系統的挑戰包括非揮發性記憶體和工作記憶體容量較小、通訊介面較慢，以及儲存狀態的能力有限。TI 在超低功耗和低成本微控制器上提供對後量子安全加密基元的支援，以實現 CNSA 2.0 合規性，並在性能與記憶體之間取得平衡，包括：

- 數位簽章：完全支援預先雜湊和純粹形式的 ML-DSA-44、ML-DSA-65 和 ML-DSA-87。
- 金鑰封裝：完全支援 ML-KEM-512、ML-KEM-768 和 ML-KEM-1024。
- 對稱式加密：支援 AES-256 和相關運算模式，包括帶密碼區塊連結訊息驗證碼的計數器與 Galois 計數器模式。
- 安全雜湊函數：完全支援 SHA-2 和 SHA-3 及 SHAKE 套件。

English Document: STDA060
Copyright © 2026 Texas Instruments Incorporated

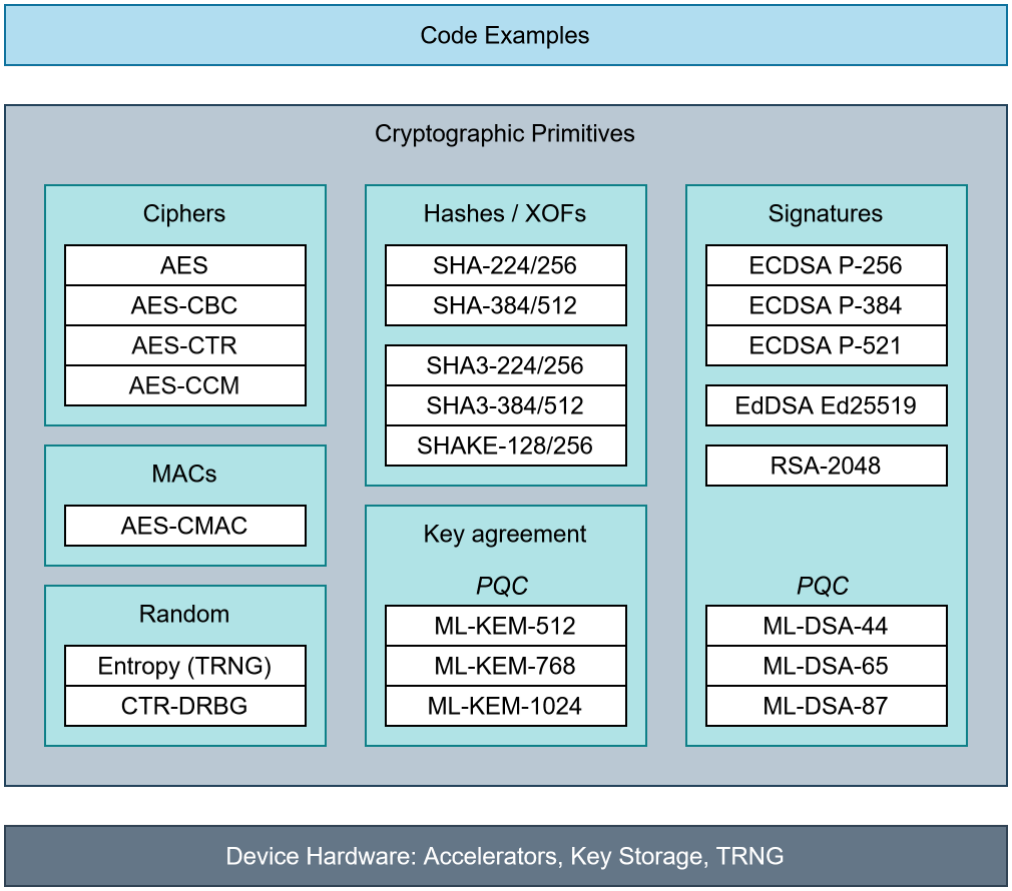


图 14-1. MSP-CRYPTO-LIB 功能

在基於 MSPM0 Arm Cortex-M0+ 與 MSPM33 Arm Cortex-M33 的微控制器上使用 TI MSP 密碼學函式庫，可在不到 260ms (Cortex-M0+) 或 70ms (Cortex-M33) 的時間內完成符合 CNSA 2.0 的 ML-DSA-87 數位簽章驗證，適用於安全啟動使用案例，即使在資源與成本受限的環境中也能提供快速啟動時間，同時為實現量子運算的未來做好準備。

15 參考資料

1. NIST。2024 年 8 月 13 日。「什麼是後量子密碼學？」於 2026 年 2 月 3 日存取。
2. NSA。2022 年 9 月。「宣布商業國家安全演算法套件 2.0。」於 2026 年 2 月 3 日存取。
3. NSA。2024 年 12 月。「商業國家安全演算法套件 2.0 與量子運算 FAQ。」於 2026 年 2 月 3 日存取。
4. NIST FIPS 204。2024 年 8 月 13 日。「基於模組晶格的數位簽章標準。」於 2026 年 6 月 6 日存取。

Copyright © 2026 Texas Instruments Incorporated

重要聲明與免責聲明

TI 以「現狀」及所含一切錯誤提供技術與可靠數據 (包含產品規格書)、設計資源 (包含參考設計)、應用或其他設計建議、網頁工具、安全資訊和其他資源，且不承擔所有明示或默示保證，包括但不限於適銷性或用於特定用途之適用性的任何默示保證，或不侵害第三方智慧財產的任何默示保證。

所述資源可供專業開發人員應用 TI 產品進行設計使用。您應自行負責 (1) 選擇適合您應用的 TI 產品，(2) 設計、驗證與測試您的應用，與 (3) 確保應用符合適用標準，以及任何其他安全、安保、法規或其他要求。

這些資源得進行修改且無需通知。TI 對您使用所述資源的授權僅限於開發資源所涉及 TI 產品的相關應用。除此之外不得複製或展示所述資源，也不提供其它 TI 或任何第三方的智慧財產權授權許可。如因使用所述資源而產生任何索賠、賠償、成本、損失及債務等，TI 對此概不負責，並且您須賠償由此對 TI 及其代表造成的損害。

TI 的產品均受 TI 的銷售條款 、TI 的通用品質指南 或 ti.com 上其他適用條款，或連同這類 TI 產品提供之適用條款所約束。TI 提供此等資源並不會擴大或以其他方式改變 TI 對於 TI 產品的適用保證或保證免責聲明。 除非 TI 明確將某產品指定為自訂或客戶指定型號，否則 TI 產品均為標準、類比、通用裝置。

TI 反對並拒絕您可能提出的任何附加或不同條款。

Copyright © 2026, Texas Instruments Incorporated

上次更新 10/2025

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), 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 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, TI’s General Quality Guidelines, 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. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

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

Copyright © 2026, Texas Instruments Incorporated

Last updated 10/2025