参考資料

Application Note

ブート モードの切り替えなしで Sitara MPU の HS-FS シリコンを HS-SE に変換

Zekun Bai, Prashant Shivhare, and Donna Xu

概要

自動車および産業用途では、システムのセキュリティや機能上のプライバシーを保護し、アプリケーション イメージが悪意を持って改ざん、コピー、削除されることを防ぐために、量産製品では一般的な開発用途とは異なり、高セキュリティ チップが一般的に使用されます。Sitara プロセッサは、この問題に対応するために、汎用 (GP) チップと高セキュリティ (HS) チップのタイプを提供しています。Sitara の HS チップには、OTP 識別用 eFuse と複数のハードウェアセキュリティ アクセラレータが組み込まれており、ブート プロセス中のシステム イメージの暗号化、復号、および署名検証を可能にすることで、外部からの悪意ある改ざんからシステムを保護します。

HS デバイスには、その状態を表す 2 種類のサブタイプがあり、高セキュリティ フィールド セキュア化可能 (HS-FS) と高セキュリティ セキュリティ強制 (HS-SE) が存在し、これらは HS デバイスの状態を示します。HS-SE デバイスでは、すべてのセキュリティ機能が有効になります。開発プロセスでは、顧客は TI 提供の Keywriter ツールを使用してチップにキーを書き込み、チップを HS-FS から HS-SE に変換する必要があります。

このプログラミング プロセスには、通常、さまざまなアプローチが含まれます。このアプリケーション ノートでは、一般的な書き込み手法をまとめるとともに、生産ラインでの手動介入を減らし、より効率的に実行できる複数の新しいアプローチを提案しています。これらの新しい方法により、ブート モードを切り替える必要がなくなります。キー プログラミングは、1 つのブート モードで実行できます。

目次

商標

1 はじめに

HS-FS と HS-SE の違いはここにあります。

HS-FS デバイス

- 署名イメージを作成しなくても、HS デバイス上で診断コードを実行できます。
- セキュア ブートなし
- JTAG 開放

HS-SE デバイス

- フルセキュアな HS デバイス
- すべてのセキュリティ ポリシーを適用さ
- セキュア ブートを強制
- JTAG をクローズ
- ファイアウォールが作動
- 利用可能なすべてのセキュリティ機能が有効

開発中は、HS-FS を HS-SE に変換するために、顧客は Keywriter を使用する必要があります。

OTP Keywriter

K3 プラットフォーム向けの OTP ライターは、HS-FS デバイス上で実行し、顧客の eFuse キーを書き込む単一バイナリとして開発されています。

OTP ライターは単一のイメージで構成されており、セキュア部分と非セキュア部分を含みます。

- セキュアでない部分、つまり OTP アプリは R5 上で実行されます
- セキュア部分は、実質的には OTP ドライバであり、DMSC サブシステム上の SYSFW の一部として実行されます

セキュアでない工場内鍵プロビジョニングのサポート:

- Keywriter には、暗号化された TI FEK 秘密鍵
- FEK 公開鍵は、対称鍵 (SMEK と BMEK) を暗号化するために顧客へ提供されます

ユーザーが設定可能なパラメータは、X509 証明書を使用して入力します。この OTP 構成証明書には、以下が含まれています:

- SMPK ハッシュおよび FEK 暗号化 SMEK、オプション、および BCH
- BMPK ハッシュおよび FEK 暗号化 BMEK、オプション、および BCH
- SWREV、KEYREV (アクティブ キーの選択) 、KEYCNT (使用キーの数)
- VPP に使用される GPIO (16FF デバイスの場合はオプション)
- wkup UART 用の UART mux 設定
- TI の対称鍵 (MEK) から導出した鍵で暗号化された、TI FEK 秘密鍵
- 完全なルート キーを持つ署名付き証明書 (クローン保護)

2 ブート モード スイッチ付き HS デバイスのフラッシュ

量産で Keywriter を使用する場合の一般的なブート フローは、メモリ A ブート モードとメモリ B ブート モードです。

以下に例を示します。

1. SD カードをブート メディア (1) 、Keywriter のブートイメージを SD カードに保存します。Norflash をブート メディア (2) として使用し、サービス用ブート イメージとアプリケーションを工場でオフライン書き込みします。
2. 最初の電源投入では、ブート モードを SD カードに設定します。ブート モード設定に基づき、ブート ROM は SD カードから Keywriter を読み込み、顧客キーをチップの OTP 領域に書き込み、チップを HS-FS から HS-SE に変換します。
3. 電源をオフにし、ブート モードを OSPI ブート モードに切り替えます。
4. 再度電源を入れると、ブート ROM は OSPI Norflash からサービス用ブート イメージを読み込みます。ブート イメージは署名され、暗号化されているため、ブート ROM はブート ROM の X509 ヘッダを使って署名を検証して復号し、その後コアを通常どおり起動します。

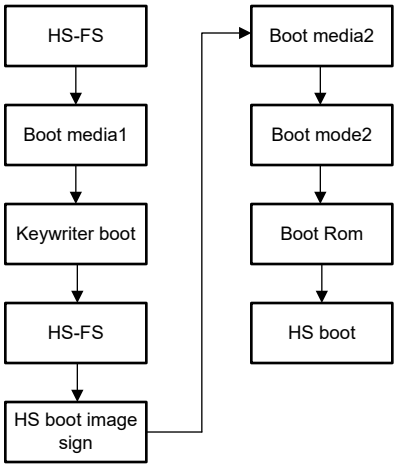

図 2-1. ブート モード スイッチによる標準的なブート フロー

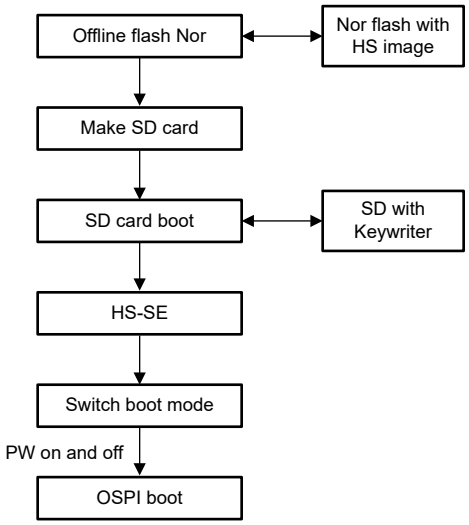

図 2-2. SD カードを一次ブート、OSPI ブートを二次ブートとする構成

この設計は比較的成熟しており、ブート モードを一度の操作で切り替えることで、多くの信頼性やフィールド運用上の問題を回避します。

ただし、両方のメモリ ブート モードに対応することで、BOM コストと設計の複雑さが増します。2 つ目の問題は、ブート モードを切り替えるために追加のジグが必要になることです。この場合、生産ラインが複雑になります。さらに、このブート モードの切り替えにはしばしば手動での介入が必要となり、生産ラインの効率をさらに低下させ、問題発生率を高めます。

3 ブート モード スイッチなしで HS デバイスのフラッシュ

ジグ不要かつ手動書き込みなしで動作し、セキュア ブートに対応できる Keywriter 方式を量産ラインで実装できるなら、セキュア ブートを必須とする顧客にとっては非常に重要な意味を持ちます。TI のプロセッサは、冗長ブート モードと冗長 OSPI ブート オフセット機構をサポートしています。このアプリケーション ノートでは、このメカニズムを活用して、問題に対する複数の設計を提示します。

プライマリ OSPI オフセットおよびバックアップ OSPI オフセット

OSPI プロトコルは、プロトコルのコマンド / アドレス / データ セグメントのビット幅 (1 または 8) とデータレート (シングル データレート (S) またはダブル データレート (D)) に従って記述されます。OSPI ブート モードは 1S-1S-8S モードをサポートしています。送出されるコマンドは 8 ビット、アドレスは 24 ビットです。OSPI モードで発行される読み取りコマンドは 0x8B で、その後にアドレス 0、および 8 サイクルのダミーが続きます。対応する動作周波数は 50MHz です。OSPI ブート モードでは、ROM コードが OSPI モジュールを初期化し、選択されたチップ セレクトに接続された OSPI フラッシュからイメージを読み取ります。フラッシュ メモリのオフセット 0x0 からイメージを正しく読み取れない場合、ROM はオフセット 0x400000 にあるイメージの取得を試みます。これは、ROM でサポートされている唯一の冗長イメージ位置です。ROM コードはまずブート イメージをオンチップ RAM にコピーし、その後そのイメージを実行します。

プライマリ ブート モードおよびバックアップ ブート モード

DMSC は、公開 ROM のブート コントローラです。DMSC は必要な設定を実行し、R5 へのリセットを解除します。

R5 はプライマリのブートモード メディアを確認し、イメージの整合性をチェックします。プライマリのブート モードが失敗した場合、R5 はバックアップ ブート モードに切り替わり、イメージの整合性をチェックします。

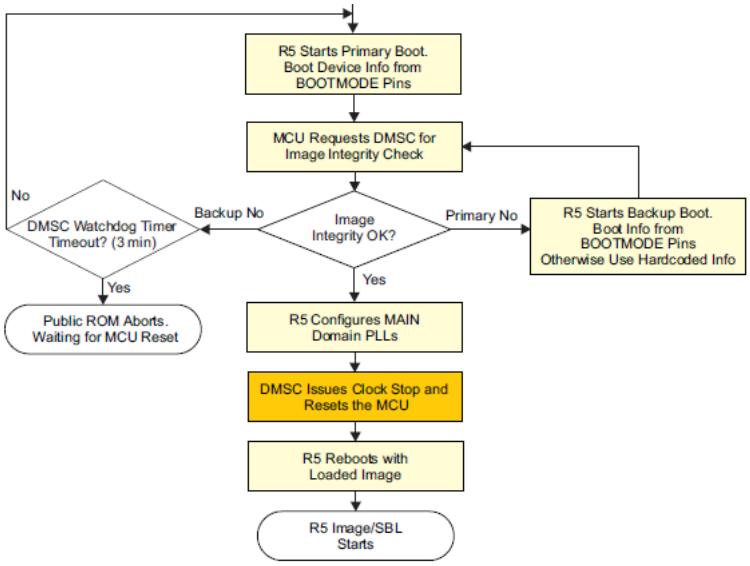

図 3-1. Sitara プライマリ ブートおよびバックアップ ブート フロー

3.1 設計 1：バックアップ ブート メディアからの起動

この設計の基本的な考え方は、DFU 経由で外部 PC に接続することです。PC には、3 つのブートローダーが保存されます。最初のブートローダーは Keywriter で、キーを書き込むために使用されます。2 番目のブートローダーは、SDK が提供する SBL_Uniflash をベースにしています。SBL_Uniflash は、顧客のキーで事前に署名および暗号化されています。ブート後、SBL_Uniflash は通常のアプリケーション ファイルと bootloader3 (顧客のキーで事前に署名・暗号化されたもの) を、フラッシュ メモリの所定のオフセットに書き込みます。電源投入時、Boot ROM はプライマリブート モードである OSPI からアプリケーション ブートローダーを読み込み、SOC を起動します。

要件

ブート モード => プライマリ：OSPI、バックアップ：UART/DFU (DFU が望ましいです)。

フラッシュは完全に空です。さらに重要なのは、フラッシュの 0x0 および 0x400000 のオフセットが消去されるため、ROM がバックアップ ブートへフォールバックする点です。

手順

1. 最初の POR 時：デバイスの状態：HS-FS。ROM は、選択されたバックアップ ブート メディアから OTP Keywriter を起動します。Keywriter ガイドに従ってキーをプログラムします。HS-FS を HS-SE に変換します。
2. 2 番目の POR：デバイスの状態：HS-SE の ROM は、選択されたバックアップブート媒体から Flash Writer を起動します。SDK イメージへのフラッシュ書き込み (SBL/ アプリケーション)
3. 3 番目の POR：デバイスの状態：HS-SE。ROM はプライマリ ブート メディアである OSPI から起動します。

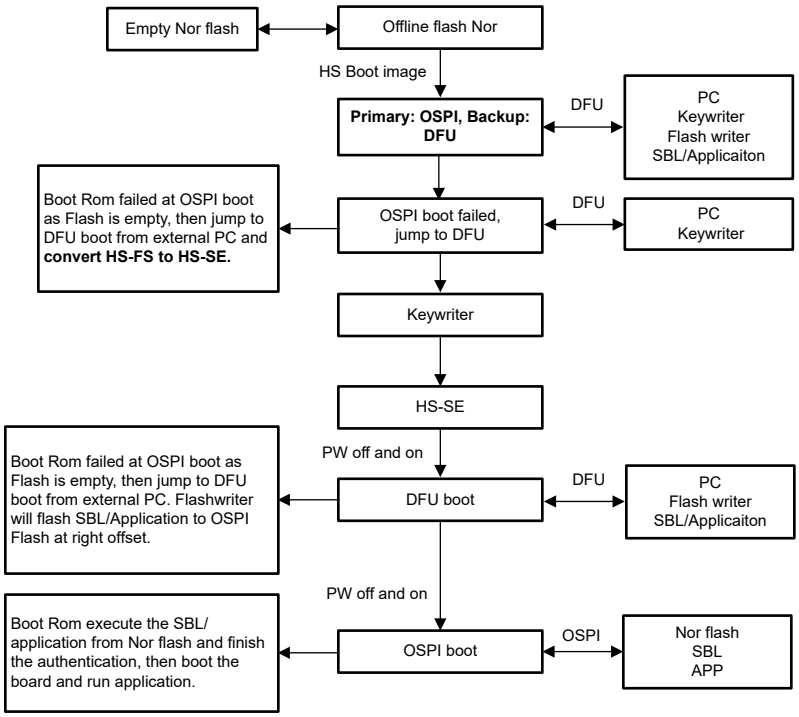

図 3-2. バックアップ ブート メディアから起動し

3.2 設計 2:プライマリ ブート メディアからの起動

この設計の基本的な考え方は、必要なファイルを指定されたオフセット位置にオフラインで Nor フラッシュへ書き込むことです。これらのファイルには 3 つのブートローダーがあります。

- 最初のブートローダーはアドレス 0x0 に配置された Keywriter で、キーを書き込むために使用されます。
- 2 番目のブートローダーは、アドレス 0x400000 にあるフラッシュライターです。このフラッシュ ライターは単一ステージの SBL/SPL ベースで、唯一の役割は、既知の位置 X にある SBL をプライマリのオフセット 0x0 に移動し、必要に応じて冗長オフセット 0x400000 にも移動することです。
- 3 番目のブートローダーには、顧客が提供するブートローダーとアプリケーションが含まれています (事前に署名され、顧客の鍵で暗号化されています)。アドレスは選択した Nor フラッシュ容量に基づいて柔軟に設定できますが、最初の 2 つのブートローダーとアドレスが重ならないことが要件となります。

初回の電源投入時、プログラムはアドレス 0x0 の Keywriter から実行され、チップを HS-FS から HS-SE へ変換します。2 回目の電源投入時、BootROM はアドレス 0x0 の Keywriter の検証に失敗し、アドレス 0x400000 にジャンプして Flash Writer を実行します。このプログラムは、顧客が提供するサービス起動プログラムとアプリケーション (事前に署名され、顧客の鍵で暗号化されたもの) をフラッシュ メモリのアドレス 0x0 に書き込み、以前の Keywriter ファイルを上書きします。この時点で、再度電源を入れると、Boot ROM はアドレス 0x0 から業務用の起動プログラムとアプリケーションを正常に読み込み、キー署名の検証と復号を行い、その後ブートを完了して SOC を起動します。

要件

ブート モード => プライマリ:OSPI、バックアップ:(未使用)

フラッシュは、所定のオフセットに 3 つのファイルをオフライン書き込みして構成されます。

手順

1. 最初の POR 時:デバイスの状態:HS-FS。ROM は、0x0 Nor フラッシュ オフセットから OTP Keywriter をブートします。Keywriter ガイドに従ってキーをプログラムします。HS-FS を HS-SE に変換します。
2. 2 番目の POR:デバイスの状態:HS-SE。ROM は、Nor フラッシュのオフセット 0x400000 から Flash Writer を起動します。SDK イメージ (SBL / アプリケーション) を 0x0、0x8000 にフラッシュします。
3. 3 番目の POR:デバイスの状態:HS-SE。ROM はオフセット 0x0 から顧客のブートローダーとアプリケーションを起動し、ボードを立ち上げます。

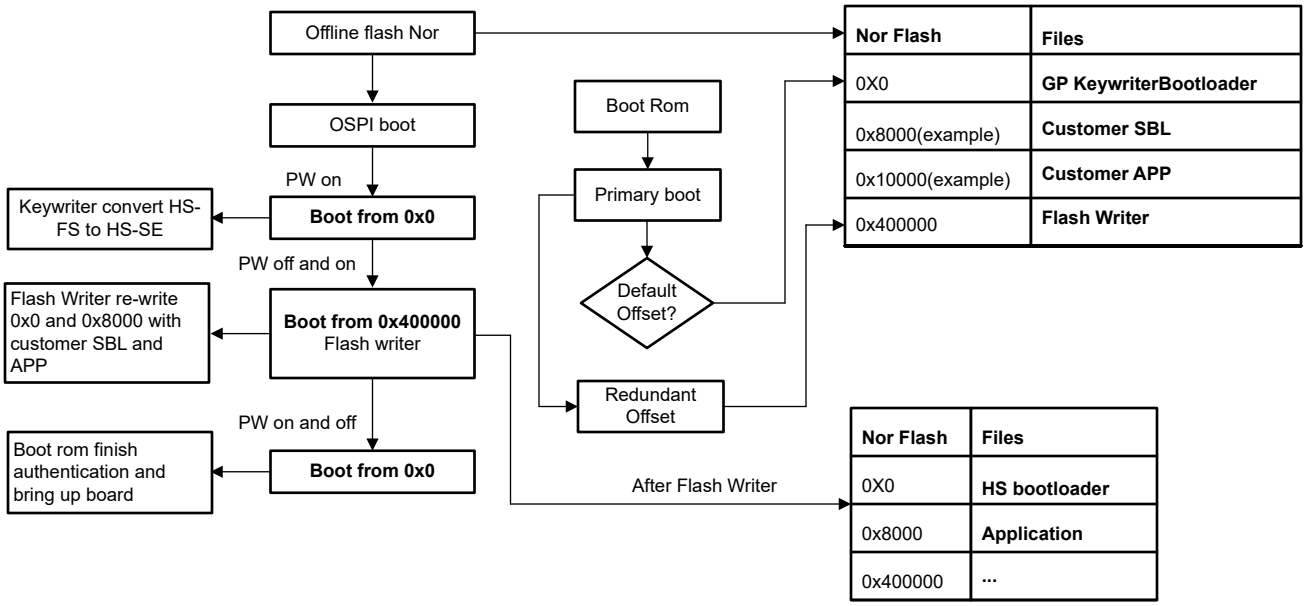

図 3-3. プライマ リブート メディアから起動

4 まとめ

このアプリケーション ノートでは、顧客が一般的に使用しているキー書き込みプロセスについて説明しており、特にブート モードを切り替えてキーを書き込み、その後ビジネス用ファイルを通常どおり起動させる方法を扱っています。

また、このドキュメントでは、2 つの実現可能な新しいアプローチについても説明します。これらのアプローチにより、ブート モードを切り替える必要がなくなります。その代わりに、これらの手法はチップの冗長ブート モードを利用し、複数回の電源サイクルによってキー書き込みとビジネス用ファイルのブートを実現します。このアプローチの主な利点は、生産プロセスの合理化と手動による介入です。さらに、顧客側のハードウェア設計では複数のブート モードをサポートする必要がなくなり、設計がより簡素化され、システムの BOM コストも削減されます。これらは単なる提案であり、お客様はそれらを徹底的にテストする必要があることに注意してください。

重要なお知らせと免責事項

TI は、技術データと信頼性データ (データシートを含みます)、設計リソース (リファレンス デザインを含みます)、アプリケーションや設計に関する各種アドバイス、Web ツール、安全性情報、その他のリソースを、欠陥が存在する可能性のある「現状のまま」提供しており、商品性および特定目的に対する適合性の黙示保証、第三者の知的財産権の非侵害保証を含むいかなる保証も、明示的または黙示的にかかわらず拒否します。

これらのリソースは、TI 製品を使用する設計の経験を積んだ開発者への提供を意図したものです。(1) お客様のアプリケーションに適した TI 製品の選定、(2) お客様のアプリケーションの設計、検証、試験、(3) お客様のアプリケーションに該当する各種規格や、その他のあらゆる安全性、セキュリティ、規制、または他の要件への確実な適合に関する責任を、お客様のみが単独で負うものとします。

上記の各種リソースは、予告なく変更される可能性があります。これらのリソースは、リソースで説明されている TI 製品を使用するアプリケーションの開発の目的でのみ、TI はその使用をお客様に許諾します。これらのリソースに関して、他の目的で複製することや掲載することは禁止されています。TI や第三者の知的財産権のライセンスが付与されている訳ではありません。お客様は、これらのリソースを自身で使用した結果発生するあらゆる申し立て、損害、費用、損失、責任について、TI およびその代理人を完全に補償するものとし、TI は一切の責任を拒否します。

TI の製品は、TI の販売条件 、 TI の総合的な品質ガイドライン 、 ti.com または TI 製品などに関連して提供される他の適用条件に従い提供されます。TI がこれらのリソースを提供することは、適用される TI の保証または他の保証の放棄の拡大や変更を意味するものではありません。 TI がカスタム、またはカスタマー仕様として明示的に指定していない限り、TI の製品は標準的なカタログに掲載される汎用機器です。

お客様がいかなる追加条項または代替条項を提案する場合も、TI はそれらに異議を唱え、拒否します。

最終更新日：2025 年 10 月