参考資料

Arm ベースのアプリケーション プロセッサのセキュリティ

Amrit Mundra
Security Architect and Systems Engineer

はじめに

コンピュータ セキュリティといえば、かつては PC 上の迷惑なウイルスを意味していましたが、その後、その重要性がますます高まっていきました。企業や政府のシステムがハッキングされたことで、個人情報や財務情報が詐欺、窃盗、横領の危険にさらされることになったのです。今、組み込みシステムのセキュリティ、正確には、組み込みシステムのセキュリティの低さが、非常に重要なデータへの脅威となっています。

今日、世界はデータで動いており、ビットやバイトのひとつひとつが潜在的な攻撃対象であると考えるべきです。同時に、ソフトウェア システムもハードウェア システムも、より複雑化し、接続され、相互に依存するようになっています。それに複雑さには脆弱性が伴います。何十億、何兆行ものコードと、相互に関連するハードウェア モジュール、サブシステム、パーティションが極小のシリコン スライスに詰め込まれているのですから、ハッカーにはたまらないでしょう。

当然のことながら、ハッカーは立ち止まっているわけではありません。衛星通信システム、ワイヤレス ベースステーション、家庭や企業のレーザー プリンタ、スマート グリッド システム、除細動器などの医療機器、その他多くのシステムにおいて、組み込みシステムの脆弱性に関する報告は後を絶ちません。マルチコアの組み込みシステム オン チップ (SoC) におけるセキュリティの必要性は、年々高まるばかりです。心臓機器、スマートフォン、車載対応制御ユニットなどの組み込みデバイスは、コントロール センターを保護するために、組み込み SoC を含む複数のコンポーネントに依存しています。

最初に、組み込みシステム内で複数のコアを持つ Arm® ベースのアプリケーション プロセッサのセキュリティを確保するために必要な要素を説明します。次に、これらのプロセッサのセキュリティの基本層であるセキュア ブートについて、より詳細に説明します。セキュア ブートによって、システムは「電源を入れた瞬間」から保護されるのです。セキュア ブートなしでは、「電源を入れた瞬間」から実際に使用可能になるまでの間、システムにセキュリティの隙間が生じます。脅威の性質は常に変化しているため、セキュリティは絶えず変化しています。

システムのセキュリティ面の目標は、データを盗んだり、システムを乗っ取ったりして本来の目的とは異なる使い方をしようとするハッカーからシステムを守ることですが、これは、関連する機能安全のコンセプトとは異なります。安全性は、システムがさまざまな状況に適切に対応し、必要であれば影響を最小限に抑えて適切に機能することに重点を置いています。これらのコンセプトを組み合わせることで、実際に何かが壊れ、悪意のある行為者が存在する現実世界でも、システムは意図したとおりに動作することができます。

リスク管理

セキュリティの脅威は常に存在しており、モノのインターネット (IoT) の急速な普及に伴い、目立たない低コストのエンドノード デバイスからでさえも、どこからでもこれらの脅威は発生する可能性があります。セキュリティの基本的な問題は、システムが攻撃されるかどうかではなく、むしろ攻撃されるタイミングにあります。このことから、セキュリティは保護と同様にリスク管理であるということがわかります。

システムが攻撃の対象になる可能性があることを考えると、システム設計者はどのようにしてセキュリティ侵害のリスクを極限まで抑えることができるのでしょうか。

保護対象

価値のあるものはすべて攻撃の対象になり得ます。もちろん、ハッカーの考え方や意図によっては、あらゆるものが価値あるものとして認識される可能性もあります。最も単純なレベルでは、システムに侵入するスリルだけでも、ハッカー コミュニティの大半にとって意味があるのです。ただ、ほとんどのハッカーは、無邪気にスリルを求めているわけではありません。多くのハッカーは、電子ウォレットからお金を引き出したり、クレジットカード番号や銀行口座番号などの金融情報を盗んで不正利用したりするのを躊躇しません。知的財産は、売却や競争上の優位性を得るために盗まれる可能性があります。一方で、国家機密が悪用され、交通システム、給水設備、エネルギー配給ネットワーク、原子力発電所、国の公共インフラなどを混乱させ、損害を与え、破壊するために利用される可能性もあります。

もちろん、こうした価値あるものはすべて保護されなければなりませんが、その前に、セキュリティ システム自体が安全である必要があります。組み込みシステムの場合、システム内のセキュリティ要素と、それが保護するものは保護される必要があります。最も基本的なレベルでは、ソフトウェア、ユーザー、

および接続リンクの検証に使用される暗号化キーと ID が保護の対象となります。また、ネットワーク内の各システムやノードで実行されているソフトウェアの整合性の確保も重要です。そのためには、ネットワーク上やインターネット上であまり注目されることのないノードであっても、起動時および実行時のソフトウェアを可視化して制御する必要があります。

セキュリティのコスト

セキュリティには、あらゆるものと同じようにコストがかかります。システム デベロッパにとってのセキュリティのコストには、セキュリティ対策を設計してシステムに統合するコストと、そのセキュリティ対策がシステムのパフォーマンスに与える影響に対処するコストが含まれます。セキュリティの脅威が常に変化していること、また、IoT のような取り組みを通じて、組み込みシステムが持続的に普及してきていることを考慮すると、新しいシステムの設計には、セキュリティのコストをその利点に照らして測定する指標の開発を含めることが必要です。組み込みデバイスは、乗っ取られ、より価値あるリソースが存在する可能性のある他のシステムへの攻撃の起点として使用可能です。たとえば、プリンタ / コピー機をハッキングしても、ハッカーにとっては大した価値はないかもしれませんが、プリンタが印刷したりコピーしたりするすべての文書がキャプチャされ、ハッカーに送信された場合、被害は計り知れません。

組み込みシステムをベースとした製品の多くは大量生産されるため、セキュリティのコストに関して、組み込みシステムには優位性があります。その結果、これらの製品用に開発されたセキュリティ サブシステムのコストは、大量生産の間に償却でき、これによってセキュリティの単価を下げることができるのです。さらに、新しい設計のために開発された汎用性、拡張性、移植性の高いセキュリティ アーキテクチャは、多くの場合、関連性の高いシステムに転用したり、他の製品のニーズに合わせてアーキテクチャを若干変更したりできます。

アーキテクチャに関する検討事項

多くのセキュリティ サブシステムは、階層構造で設計され、コンパートメント化をうまく利用しています。セキュリティ対策を階層的に展開すると、システムのセキュリティに累積的な効果がもたらされます。これは、何らかのアクションが実行される前に、各階層がそれより下の階層や上の階層のセキュリティを認証する必要があるからです。コンパートメント化は、システム上で動作するソフトウェアのランタイム セキュリティを確保する上で重要であり、設計者は、保護対象のリソースやプロセスの相対的な価値に応じてセキュリティ対策をカスタマイズできます。

組み込みセキュリティは、ハードウェアから始まります。ソフトウェアとハードウェアのセキュリティ機能を組み合わせることで、どちらかのソリューションが単独で機能するよりもセキュアな保護層が実現できます。さらに、ベンダが提供するツールを使用すると、セキュリティ サブシステムの開発を効率化し、結果として得られるアーキテクチャがデベロッパの要件を確実に満たすようにできるのです。たとえば、ハードウェア ベースのセキュリティ アクセラレータを使用すると、セキュリティ サブシステムのパフォーマンス コストを抑えることができます。

当然ながら、セキュリティ アーキテクチャの強度は、それが構築される基盤に依存するものです。基本層には、セキュア ブート プロセス、ハードウェア ベースのデバイス ID / キー、暗号化アクセラレーション機能という 3 つの要素が不可欠となっています。

セキュリティ ピラミッド

セキュリティ ピラミッド (図 1 を参照) は、マルチコア SoC 組み込みプロセッサ向けの包括的なセキュリティ サブシステムの各層と構成部分を示しています。

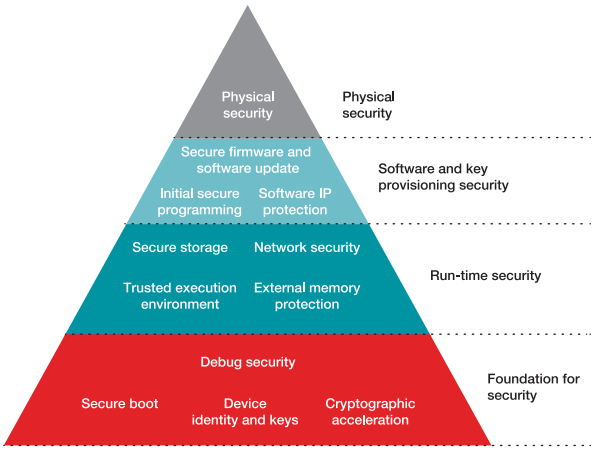

図 1. セキュリティ ピラミッド

セキュア ブート

セキュア ブート プロセスは、組み込みシステムの信頼の基点を確立するものです。ブートが外部フラッシュ メモリから起動される場合でも、セキュア ブート プロセスは、組み込み暗号化キーなどを含む多くのメカニズムを通じて、ブート ファームウェアの整合性を検証します。セキュア ブート層は、マルウェアによってシステムが乗っ取られたり、システム内 IP がクローン化されたり、好ましくないアプリケーションが意図せず実行されたりすることや、その他のセキュリティ リスクからシステムを保護します。

また、セキュア ブートは、内部メモリを保護するために IP を暗号化して安全にコピーすることで、追加の保護層を設けるのに役立ちます。暗号化機能を持つことで、指向性探索攻撃ができなくなるため、コードベースのセキュリティも強化されます。

要は、セキュア ブートは組み込みシステムのセキュリティの基盤を確立するのに大いに役立つということです。

暗号化アクセラレーション機能

さまざまな公開キーや秘密キーの生成、検証、認証に関わる暗号化処理は、組み込みシステムのパフォーマンスやスループットに対して大きな負担となります。マルチコア アプリケーション プロセッサの中には、ハードウェア ベースのアクセラレータやコプロセッサを搭載しているものがあり、コーディング / デコーディングの各処理を大幅に高速化できます。ソフトウェアベースのアクセラレーション機能も利用できますが、ソフトウェアであるため、ハードウェアベースの暗号化アクセラレーション機能ほど本来の安全性は高くありません。

一般的な暗号化要素	
乱数生成器 (RNG)	暗号化アルゴリズムとハッシュ関数で使用されます。ハードウェアで生成された乱数は、ソフトウェアで生成された RNG よりもセキュアです。
暗号化アルゴリズム	
トリプルデータ暗号規格 (3DES)	3DES は、暗号化されたデータの保護を強化し、DES アルゴリズムのいくつかの脆弱性を克服するために、DES 暗号化を 3 回実行します。
公開キー暗号化アルゴリズム (PKA)	公開キー / 秘密キーを使用した RSA または ECC 非対称暗号化を使用した高速化された PKA で、セキュア ブートで使用される認証に役立ちます。
高度暗号化標準 (AES)	AES は、現在広く使用されている最も高度な暗号化アルゴリズムの 1 つです。
ハッシュ関数 (署名、認証など)	
メッセージ ダイジェスト アルゴリズム (MD5)	このハッシュ関数は広く使われていますが、アプリケーションによっては脆弱性があります。
セキュア ハッシュ アルゴリズム 2 (SHA2)	大きなハッシュを処理するため、SHA1 よりもセキュアです。

表 1. *一般的な暗号化関数の例*

デバイス ID およびキー

ローカルエリア ネットワーク (LAN)、ワイドエリア ネットワーク (WAN)、またはインターネットを介した通信を信頼して利用するためには、デバイス同士が共有可能な固有の ID を持つ必要があります。そこで通信を行うデバイスは、対話に参加している他のデバイスの真正性や信頼性を判断することができます。

アプリケーション プロセッサには、多くの場合、何らかの固有の識別 (ID) コードが付けられています。また、ID コードの他にも、たとえばクラウド サービス経由でアクセス可能な、対応する公開キーを持つ署名または証明書キーによってデバイスを識別することもできます。

図 2. 盗難防止に役立つデバイス ID

デバッグ セキュリティ

システム開発中、ファームウェアやソフトウェアをデバッグしたり、ハードウェアの潜在的な問題をトラブルシューティングしたりするために、設計者は組み込みマルチコア アプリケーション プロセッサにアクセスする必要があります。ほとんどの場合、JTAG ポートがこのアクセスに使用されます。動作環境では、デバッグ ポートを何らかのヒューズで密閉するか、認証された暗号化キーを使用してのみアクセスできるようにする必要があります。そうしなければ、デバッグ ポートは、ハッカーにシステムへの簡単な侵入経路を提供することになります (図 3 を参照)。

図 3. MSP430™ MCU デバック ポート

信頼できる実行環境

ランタイム セキュリティ層は、ブートアップ プロセスの後、システムのオペレーティング システム (OS) が実行されている間、システムを保護する役割を果たすいくつかの異なる機能で構成されています。ランタイム セキュリティの重要な要素として、いつ侵入が発生したのか、いつ侵入が試みられたのかを判断するために、システムのあらゆる側面を監視することが挙げられます。

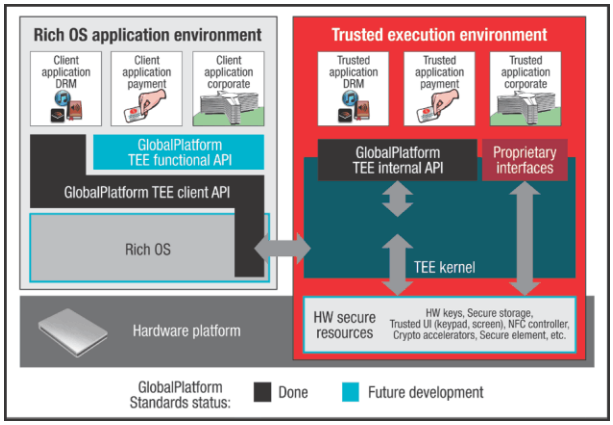

図 4. 信頼できる実行環境 (Trusted Execution Environment、TEE)

信頼できる実行環境のセキュリティでは、システムはセキュアなアプリケーションとセキュアでないアプリケーションを同時にホストし、データの漏洩がないようにパーティションを維持することができます。アプリケーションおよび関連するコード / データベースが他のアプリケーションから完全にサンドボックス化されている場合、機密性の高いアプリケーションを実行することが重要です。

信頼できる実行環境は、マルチコア システム内でセキュアなパーティションを提供します。基本的に、このパーティションでは認証されたセキュアなファームウェア、ソフトウェア、およびアプリケーションのみが実行され、認証されたデータが保存できます。

信頼できる実行環境をマルチコア / マルチプロセッシング システムの他の部分から隔離することで、システムを経由する可能性のある不信なコード、アプリケーション、データがミッション クリティカルなソフトウェア、データ、その他の IP を侵害することを防止できます。

外部メモリの保護

設計者がシステムに別のアプリケーションやサブシステムを追加しなければならない場合、通常、メイン プロセッサに外付けされ、メモリ バスで接続されたメモリを追加する必要があります。設計者は、外部メモリに保存されたデータを改ざんや置き換えから保護し、信頼できるデータまたはアプリケーション コードのみが外部メモリに保存されるようにしなければなりません。外部メモリのコンテンツを保護するためには、さまざまな方法を採用できます。たとえば、データをプロセッサの統合メモリにロードせずに外部メモリから直接実行するセキュアな

インプレース実行や、メイン プロセッサ上でアプリケーションを実行しながら機密性を維持する動的な復号化などがあります。

図 5. セキュア メモリ

ネットワークのセキュリティ

ハッカーは、無線や有線のネットワーク通信を傍受することに非常に長けています。実際、一部の通信プロトコルには、セキュリティ上の弱点があり、すでに悪用されていることがわかっています。安全性の高い通信プロトコルのみを導入する場合、通信ストリームの暗号化と復号化、および送信者や受信者の真正性の検証には、多くの処理サイクルが必要になります。設計者は、通信スループットとセキュリティのバランスの問題に対処しなければならないことがありますが、組み込みプロセッサの中には、標準的な通信プロトコルと組み合わせて使用される暗号化アルゴリズム用のハードウェア ベースのアクセラレータを統合することで、この難題を回避しているものもあります。

図 6. セキュア ストレージ

セキュア ストレージ

暗号化キーやセキュリティ データは、システム メモリ内で好ましくないアクセスを受けにくい場所に保存する必要があります。暗号化されたキー データ、マスター キーでしかロック解除できない改ざん防止機能、非揮発性メモリと暗号化エンジン間に設けられた秘密キー バスなど、セキュア ストレージを提供するためには多くの機能を使用できます。

初期のセキュア プログラミング

設計、キーのプロビジョニング、製造がそれぞれ別々であり、場合によっては海を隔てて行われることもある今日のグローバリゼーションの時代では、キーのようなセキュリティ資産を安全に保管することが課題となります。さらに複雑なのは、ビジネス モデルとして、完全に信頼されていない製造体制を持つ ODM が関与している場合があることです。

初期のセキュア プログラミングのようなセキュリティ イネーブラは、信頼できない施設でプログラムされた初期ファームウェアやキーの機密性、整合性、および真正性を強化するため、またはアプリケーションの最初の起動時に、お客様が評価して使用を選択できる方法論を提供するものです。

ファームウェアとソフトウェアのセキュア更新

システムを更新することは、セキュリティ フレームワークにとって不可欠な部分です。これによって、お客様はリモートでソフトウェアにパッチを適用したり、ソフトウェアを更新したりして、システム内の特定された脆弱性に対処することができます。しかしながら、更新中の最大の課題として、スパイ行為、なりすまし、再生攻撃を抑止することがあります。

セキュリティ フレームワークは、認証、暗号化、整合性チェックなどの追加のキーとメカニズムを提供して、更新の真正性を保証するために導入できます。

TI のソフトウェア知的財産 (IP) の保護

お客様は、市場においてエンド ユーザーに対する重要な価値提案となり得る知的財産 (IP) を作成するために多額の投資を行っています。そのため、お客様の IP を保護できるように、暗号化されたブート、隔離された処理を実行する機能、ファイア ウォールのようなメカニズムを提供するセキュリティ フレームワークが不可欠になります。

物理的なセキュリティ

高度なハッキング組織もそれほど高度でないハッキング組織も、埋め込まれた資産にアクセスするために、システムからチップを取り出したり、チップ パッケージからシリコン ダイを取り出したりすることがあります。

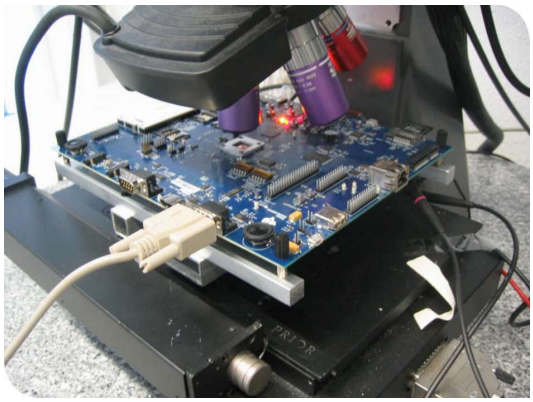

図 7. 物理的な攻撃を受けているシステム

デバイスやダイを取り出すと、ハッカーはレーザーを照射したり、指定された電力制限を超えてデバイスに電力を供給したり、さまざまな手段を講じます。デバイスが刺激にどのように反応するかを観察することが目的であり、この反応はハッカーがデバイスにアクセスするために悪用できる脆弱性を示唆する可能性があるためです。

アプリケーション プロセッサの中には、SoC のデジタル セクションとアナログ セクションへの物理的な侵入を阻止するためのハードウェアとソフトウェアの機能が統合されているものがありますマルチコア アプリケーション プロセッサに統合された改ざん防止モジュールには、電力モニタや温度モニタ、リセット機能、周波数モニタ、プログラム可能な改ざん保護機能を搭載できます。

エンクロージャの保護

エンクロージャ保護機能とは、システムを格納するエンクロージャを保護する物理的な手段で、ロック機構から電子スイッチ、分離式ワイヤ トリップ機構などがあります (**図 8** を参照)。

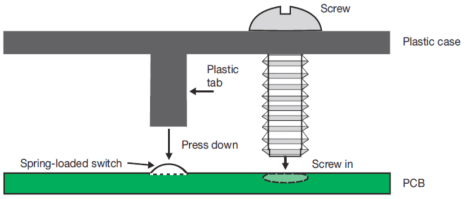

図 8. エンクロージャの保護

組み込みセキュリティの出発点

組み込みマルチコア アプリケーション プロセッサのセキュリティの基本は、ハードウェアから始まります。ハードウェアがセキュアでなければ、いくらセキュリティ ソフトウェアを使ってもセキュアな状態は確保できません。セキュリティ機能がハードウェアに組み込まれていることを前提とすると、セキュリティ サブシステムを構築するために最初に注目すべきは、電源投入後に実行される最初のソフトウェア、つまりブート コードです。ブート プロセスが認証できなければ、システム上で実行される他のソフトウェアも認証することができません。したがって、ブート プロセスのセキュリティ確保が、システムのすべてのセキュリティを左右する重要な支点になるのです。

セキュア ブート プロセスは信頼の基点を確立するもので、これがすべてのセキュリティ サブシステムの目標なのです。セキュア ブート プロセスを通じて信頼の基点を確立することで、システムの整合性を確保し、ハッカーによるシステムのいかなる部分の乗っ取りからも保護することができます。また、システム内のお客様のソフトウェアを保護し、システムまたはその一部がコピーされないように、クローン対策バリアとしても機能します。

通常、セキュア ブート プロセスには、システム内のどこかにある非揮発性のワンタイム プログラマブル メモリに公開暗号化キーをプログラムすることが含まれます。この公開キーは、実行が始まる前に暗号化されたブート コードの正当性を認証するために、ブート コードに関連する秘密キー / 公開キーとマッチングされる必要があります。ブート ファームウェアは、組み込みプロセッサの RAM にロードすることもできますし、セキュリティを強化するために、組み込みプロセッサの外部メモリからセキュアにインプレース実行することもできます。ファームウェア イメージの中には、さまざまなコンポーネントやモジュールで構成されているものがあり、各モジュールを復号化

して実行する前に認証を要求することで、ブート セキュリティを強化しています。

TI アプリケーション プロセッサのセキュリティ イネーブラ

Arm ベースのアプリケーション プロセッサでは、デベロッパが資産 (データ、コード、ID、キー) を保護するためのセキュリティ対策を実装できるよう、幅広いセキュリティ イネーブラを提供しています。

セキュリティ イネーブラ	AM335x	AM437x	AM438x	AM570x/ AM574x	AM64x/AM65x	AM62/ AM62L/ AM62P	AM68x/AM69x	DRA821/ DRA829/ TDA4VM
暗号化アクセラレーション機能	✓	✓	✓	✓	✓	✓	✓	✓
デバイスの ID / キー	✓	✓	✓	✓	✓	✓	✓	✓
セキュア ブート	✓	✓	✓	✓	✓	✓	✓	✓
デバッグ セキュリティ	✓	✓	✓	✓	✓	✓	✓	✓
外部メモリの保護			✓		✓	✓	✓	✓
信頼できる実行環境 (Trusted Execution Environment、TEE)		✓	✓	✓	✓	✓	✓	✓
ネットワークのセキュリティ					✓	✓	✓	✓
セキュア ストレージ		✓	✓	✓	✓	✓	✓	✓
ソフトウェアの IP 保護	✓	✓	✓	✓	✓	✓	✓	✓
初期のセキュア プログラミング	✓	✓	✓	✓	✓	✓	✓	✓
ファームウェアのセキュア更新	✓	✓	✓	✓	✓	✓	✓	✓
物理的なセキュリティ			✓					
アクセス権の申請	TI 担当者に連絡する	**詳細**	**詳細**	**詳細**	**詳細**	**詳細**	**詳細**	**詳細**

表 2. *TI アプリケーション プロセッサのセキュリティ イネーブラ*

まとめ

組み込みプロセッサのセキュリティは、多方面に渡る複雑な課題です。IoT の台頭と組み込みシステムの普及により、ハッカーはこれまで以上に多くの攻撃対象を持てるようになりました。

もちろん、基本的なセキュリティ機能はハードウェア内にすでに存在していなければなりませんが、組み込みマルチコア SoC のセキュリティ サブシステムの構築は、基本層であるセキュア ブートから開始する必要があります。セキュア ブート プロセスに基づく信頼の基点がなければ、他のセキュリティ対策など何の役にも立たないのです。この信頼の基点が確立されると、デバッグ セキュリティ、ランタイム セキュリティ、ネットワーク セキュリティなど、システム セキュリティの他の側面がしっかりとした土台を持つことになります。そうでなければ、すべてのセキュリティ対策は砂上の楼閣と化すことになります。

参考資料

1. テキサス・インスツルメンツ：*e-Book（電子書籍）：『セキュリティを考慮しながらアプリケーションを構築』*(SWPB021)
2. テキサス・インスツルメンツ：『**ハードウェア アクセラレーション形式の暗号化による、セキュリティの確保とチップ パフォーマンスの向上**』
3. テキサス・インスツルメンツ：『**組み込み Sitara ™プロセッサのセキュア ブート**』
4. テキサス・インスツルメンツ：『**Sitara AM438x プロセッサ、改ざん防止機能**』

重要なお知らせと免責事項

テキサス・インスツルメンツは、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、 テキサス・インスツルメンツ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 テキサス・インスツルメンツ製品の選定、(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 テキサス・インスツルメンツ製品を使用するアプリケーションの開発の目的でのみ、 テキサス・インスツルメンツ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。 テキサス・インスツルメンツ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、 テキサス・インスツルメンツおよびその代理人を完全に補償するものとし、 テキサス・インスツルメンツは一切の責任を拒否します。

テキサス・インスツルメンツの製品は、 テキサス・インスツルメンツの販売条件、または ti.com やかかる テキサス・インスツルメンツ製品の関連資料などのいずれかを通じて提供する適用可能な条項の下で提供されています。 テキサス・インスツルメンツがこれらのリソースを提供することは、適用される テキサス・インスツルメンツ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。

お客様がいかなる追加条項または代替条項を提案した場合でも、 テキサス・インスツルメンツはそれらに異議を唱え、拒否します。

郵送先住所：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265