XAPP1084 (v1.2) 2012 年 8 月 10 日

Virtex-6 および 7 シリーズ FPGA での不正操作防止デザインの開発

著者 : Ed Peterson

はじめに

このアプリケーション ノートでは、FPGA をベースとするシステム内の知的財産 (IP) や重要な情報を保護するアンチ タンパー (不正操作防止/AT) 機能について説明し、それらの実例を紹介します。この保護機能 (不正操作防止機能) は、FPGA がビットストリームでコンフィギュレーションされる間およびその前後に有効にならなければなりません。重要なデータには、FPGA ロジックの機能を設定するコンフィギュレーション データ、ビットストリーム内に含まれるクリティカルなデータ/パラメーター (ブロック RAM の初期値、フリップフロップの初期状態など)、そしてコンフィギュレーション後の通常動作時に FPGA に対して動的に書き込みまたは FPGA から動的に読み出す際に使用する外部メモリ データなどがあります。

このアプリケーション ノートでは、Virtex®-6 ファミリ、7 シリーズ (Artix™、Kintex™、Virtex) FPGA、および Zynq™-7000 ファミリのエクステンシブル プロセッシング プラットフォーム (EPP) で利用できる AT 機能について説明し、これらの機能が必要な理由、各機能の使用例、およびインプリメンテーションの詳細を示します。また、さらなる不正操作防止効果を実現するためのさまざまな方法についても説明します。

このアプリケーション ノートを参考にすることで、FPGA デザインで最高レベルの保護機能を実現する AT の効果的な設計手法を習得できます。これらの設計手法は幅広い分野を対象とし、商用デザインのクローニング (模造) やオーバービルディング (過剰生産) への対策、または軍用のクリティカル テクノロジ (CT) をリバース エンジニアリングから保護する目的などに使用できます。

この資料は、ザイリンクス FPGA のアーキテクチャ [参照 1] および設計に関して専門的な知識があり、また ISE® を使用した設計フロー[参照 2]の経験が豊富なユーザーを対象としています。『デザインの安全性の確保』[参照 3] では、FPGA のさまざまな安全保障上の脅威とソリューションについて説明しています。

概要

ザイリンクスは、長年、FPGA AT ソリューションのプロバイダーとして業界を牽引してきました。Virtex-II は、ビットストリームの暗号化機能を備えた最初の FPGA です。Virtex-6 や 7 シリーズ FPGA では、ビットストリームの暗号化や認証のほかにさまざまな AT 機能をシリコンに統合してソリューションを拡充してきました。また、ザイリンクスは、コンフィギュレーション後に有効な信頼性の高い不正操作防止機能として、Security Monitor というソフト IP コアも提供しています [参照 4]。ただし、Security Monitor は、特定条件下での使用制限があります。詳細は、ザイリンクス FAE へお問い合わせください。

不正操作防止に対して敵よりも一歩リードするということは、潜在的な脆弱性や攻撃を把握して攻撃に対抗するために新しい緩和技術 (対策) を開発し続けることです。ザイリンクスは、AT 機能を重視するユーザーが FPGA テクノロジを安心して利用できるように、複数世代にわたって改良を続けています。

システム設計者は、ザイリンクス FPGA のさまざまな AT 機能の特徴を理解することで、どれほどの AT 機能を FPGA デザインに組み込むべきかを判断できるようになります。AT は、個々のシリコン AT 機能を有効にして使用する方法と、これらの AT 機能を組み合わせて複数使用する方法があります (通常は開発者が FPGA 内で組み合わせて使用し、特定ガイダンスに従う)。

© Copyright 2011-2012 Xilinx, Inc. Xilinx, the Xilinx logo, Artix, ISE, Kintex, Spartan, Virtex, Zynq, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. All other trademarks are the property of their respective owners.

デザインに組み込む AT 機能を判断するには、主に次の 3 つの要素を考慮します。

- 価値：IP の知覚的価値、および不正操作が行われた場合に経済的または国家安全保障に与えるダメージ。特定の AT 機能は非常にコストがかかり、保護するテクノロジの価値に対するコストを慎重に検討する必要があります。
- アドバーサリー (敵)：システムへの接近手段、および攻撃にあたっての専門レベル/資質。たとえば、システムへのアクセスは「銃、扉、警備員」で防ぐことはできるか、一般市場で簡単に入手できるか、敵は個人レベルのハッカーまたは組織的な大規模グループか、など。このように、敵の攻撃レベルは極端なものから中間レベルに至るまでさまざまです。
- 設計段階：FPGA デザインに AT 機能をインプリメントするタイミングは、システム開発のどの段階か。ザイリンクスでは、スケジュールとコストを最小限に抑えるために、できるだけ早い段階 (システムで CT が定義された後など) で FPGA AT 機能をインプリメントすることを推奨しています。設計後期に AT 機能を挿入する場合、通常よりも多くのコストと時間が必要になります。

その他には、一部の AT 機能を有効にすると、FPGA ロジック リソースを余分に消費する可能性があることに留意してください。通常、全体的なリソース消費による影響は最小限に抑えられますが、これらの機能のインプリメント方法や FPGA デバイスのサイズによって異なります (例：大規模デバイスほど影響が少ない)。

ザイリンクスでは、FPGA にビルトインされている AT 機能を受動的セキュリティと能動的セキュリティに分類しています。受動的セキュリティ機能は FPGA 内に構築されているため、ユーザーが FPGA ロジック デザインに対して何もする必要がありません。また、このセキュリティ機能は、一時的な特性を持ち、FPGA の通常動作サイクルの異なるタイミングで効果を発揮します。

- コンフィギュレーション前 (例：バッテリー電源によりバックアップされている RAM (BRAM) 内の復号化キーによる保護)
- コンフィギュレーション中 (例：ビットストリームの認証や復号化によるユーザー デザインの保護)
- コンフィギュレーション後 (例：リードバック機能の無効化によるユーザー データの保護)

能動的セキュリティ機能の場合、FPGA デザインでユーザー デザインを操作する必要があります。これらの機能は、ユーザー ビットストリームで FPGA がコンフィギュレーションされ、ユーザー デザインがアクティブになった後にのみ有効になります。例としては、KEYCLEARB をアサートしてバッテリー駆動型の AES キーを消去したり、PROGRAM_B インターセプト機能を利用する方法があります。

最小限の措置として、システム設計者は適切な受動的セキュリティ機能 (例：ビットストリームの暗号化および認証) をデザインに含める必要があります。これらの機能は、ユーザー デザインの機能性に影響を与えることはありませんが、ロジック上の負荷 (例：キー管理)、システム上の負荷 (例：キー ストレージ用に BRAM を使用する場合には、バッテリーが必要)、およびコンフィギュレーション時間の延長 (暗号化されたコンフィギュレーションは 8 ビット データ幅に制限されている) などが生じる可能性があります。それ以外では、これらの機能を無償で利用して、ある程度の不正操作防止策を講じることができます。すでに配備済みのシステムや開発後期のデザインには、これらの AT 機能を利用することを推奨します (まだ導入していない場合)。

さらに、この資料で紹介する不正操作防止機能およびガイドラインは、主に 3 つのカテゴリに分類されます。

1. 防止 (例：ビットストリームの暗号化/認証)
2. 検出 (例：電圧および温度の監視)
3. 応答 (例：BBRAM に格納されたビットストリーム復号化キーの消去)

表 1 に、Virtex-6 および 7 シリーズ FPGA にビルトインされている AT 機能とそのカテゴリを示します。

表 1 : AT 機能の詳細と分類

Virtex-6 および 7 シリーズ FPGA シリコンの AT 機能	種類	カテゴリ
揮発性の AES-256 BBRAM キー ストレージ	受動的[1]	防止
不揮発性の AES-256 eFUSE キー ストレージ	受動的[1]	防止
256 ビット AES ビットストリームの復号化	受動的[2]	防止
HMAC SHA-256 ビットストリームの認証	受動的[2]	防止
耐性強化されたリードバック無効化回路	受動的[3]	防止
優れたキー ロード有限ステート マシン (FSM) 回路	受動的[3]	防止
JTAG の無効化	能動的	防止
JTAG の監視	能動的	検出
内部コンフィギュレーション メモリの完全性	能動的	検出
オンチップ温度および電圧の監視/警告 (SYSMON/XADC)	能動的	検出
PROG インターセプト (PREQ/PACK)	能動的	検出
固有識別子 (Device DNA およびユーザー eFUSE)	能動的	検出
内部コンフィギュレーション メモリの消去 (IPROG)	能動的	応答
内部 AES-256 BBRAM キーの消去 (KEYCLEARB)	能動的	応答
グローバル トライステート (GTS)	能動的	応答
グローバル セット/リセット (GSR)	能動的	応答

注記 :

1. コンフィギュレーション前
2. コンフィギュレーション中
3. コンフィギュレーション後

後続セクションでは、上記の機能の概要とその必要性を説明し、例を用いて適切な使用方法 (単独使用、またはほかのビルトイン機能やユーザー ロジックと組み合わせて使用) を詳しく解説しています。さらに、FPGA デザインやシステム全体の不正操作防止レベルを引き上げるために有効な手段およびテクニックについても具体的に説明しています。

FPGA レベルで有効化される AT 機能は、包括的なシステム レベル AT ソリューションの一環として考えるべきです。この資料で説明する機能やテクニックは、FPGA にとって非常に優れた傘のような機能を果たしますが、システム全体を考慮した複数層アプローチを展開し、その一つとしてこの AT 機能を利用すると最も効果的です。

受動的 AT シリコン機能

ビットストリームの暗号化/復号化

暗号化した[1]ビットストリームを外部フラッシュに格納し、(FPGA の復号化エンジン内で) FPGA コンフィギュレーション時に動的に復号化することで、高レベルの機密性を確保できます。これにより、同じ秘密キーを共有する人のみビットストリーム内の情報へアクセスできるようになります。ビットストリームの暗号化は、システムの電源がオフの間およびコンフィギュレーション中の機密性を確保することができ、ブロック RAM やフリップフロップの初期化データなどの FPGA デザインのコンテンツを保護します。ザイリンクスでは、外部メモリに格納されているビットストリームは常に暗号化された状態で保持することを推奨しています。

このセキュリティ機能を利用するには、まず最初に暗号化するためにユーザーが指定したキーの値を使用して、BitGen ソフトウェア [参照 2] でコンフィギュレーション ビットストリームを暗号化する必要があります。AES キーが提供されていない場合は、Bitgen が自動的にキーを生成します。ただし、Bitgen

1. ザイリンクスは、CBC (暗号文ブロック連鎖) モードで 256 ビット キーの AES を使用して暗号化します。

で生成されるキーは単なる疑似ランダムであり、実際のランダム キーほどセキュリティ レベルが高くないということに留意してください。このキーは、iMPACT ソフトウェア [参照 2] を使用して、JTAG 経由で FPGA へロードされます。キーは、FPGA 内の揮発性 BBRAM または不揮発性 eFUSE のいずれかのストレージへロードできます。いずれのメモリへキーを格納すべきかを判断するために、システム設計者は BBRAM (表 2) と eFUSE (表 3) の長所と短所をそれぞれ理解しておく必要があります。

表 2 : BBRAM ストレージへ格納する場合の長所および短所

長所	短所
• 揮発性、プログラム可能 • 受動的および能動的にキーを消去可能 (例 : 証拠を削除できる) • 不正操作防止[(1)]	• 外部バッテリーが必要 • 多くのバッテリー ベンダーは高温または長期利用における動作仕様を定義していません (一部のベンダーはこれらの問題に対応するため、ベータボルタ式バッテリーの提供を開始)。

注記 :

1. JTAG 経由で BBRAM の読み出し / 書き込みが行われようとした場合、アクセスが有効になる (例 :「key access mode」になる) 前に BBRAM から FPGA のコンフィギュレーション データがすべて消去されます。

表 3 : eFUSE ストレージへ格納する場合の長所および短所

長所	短所

• 外部バッテリーが不要
• スプーフィング (なりすまし) が不可
  • eFUSE キーで暗号化されたビットストリームのみ FPGA へロードすることができます。`cfg_aes_only`[(1)(2)] eFUSE ビットもセットされている場合には、その他すべてのビットが拒否されます。
	• 恒久的 : キーの消去が不可
	• BBRAM ソリューションよりもセキュリティ レベルが低い (例 : 証拠が残る)
	  • データを安全にスクランブルする唯一の方法は、承認されたアルゴリズムや秘密キーを使用して暗号化することであるため、ザイリンクスでは eFUSE キーにスクランブルをかけません。

注記 :

`cfg_aes_only` オプション (eFUSE 制御レジスタ FUSE_CNTL 内にある) を使用する場合には、重要なポイントが 2 つあります。

1. ザイリンクスの間接プログラム コア ビットストリームは暗号化されていないため、ビットストリームの間接フラッシュ プログラム手法 [参照 5] を使用する場合、このオプションはボード上のフラッシュに暗号化ビットストリームがロードされた後に有効になる必要があります。それ以外の場合、FPGA は eFUSE キーを使用して間接プログラム コアを復号化しようとして、コンフィギュレーションがエラーとなります。
2. スプーフ保護以外にも、`cfg_aes_only` オプションを有効にして、eFUSE キーで暗号化されたビットストリームを攻撃者が復元できないように保護することが重要です。この場合に予想される攻撃の状況として、ユーザー ビットストリームが eFUSE に格納されているキーで暗号化され、攻撃者のデザインが BBRAM キーで暗号化され、`cfg_aes_only` オプションが設定されていないことが考えられます。攻撃者は ICAP (内部コンフィギュレーション アクセス ポート) 経由のパーシャル リコンフィギュレーションを使用してユーザー ビットストリームを読み出す可能性があります。ユーザー ビットストリームが BBRAM キーで暗号化され、攻撃者のデザインが eFUSE キーで暗号化されている場合は、この攻撃が不可能です。

ビットストリームの暗号化およびキー ストレージの詳細は、Virtex-6 および 7 シリーズ FPGA のコンフィギュレーション ユーザー ガイド [参照 6]、[参照 7] を参照してください。

ビットストリームの認証

Virtex-6 または 7 シリーズ FPGA でユーザーがビットストリームの暗号化/復号化を有効にした場合、認証機能が自動的にオンになります (つまり、両者は相前後して有効/無効となる)。認証方式は、SHA-256 ハッシュ アルゴリズム (FIPS180-2) [参照 10] を使用する HMAC (ハッシュ ベース メッセージ認証コード) [参照 9] を採用します。

このタイプの認証によって、暗号文改ざんを防止することが可能です。代表的な暗号文改ざんには AES-CBC 攻撃があります (図 1 参照)。

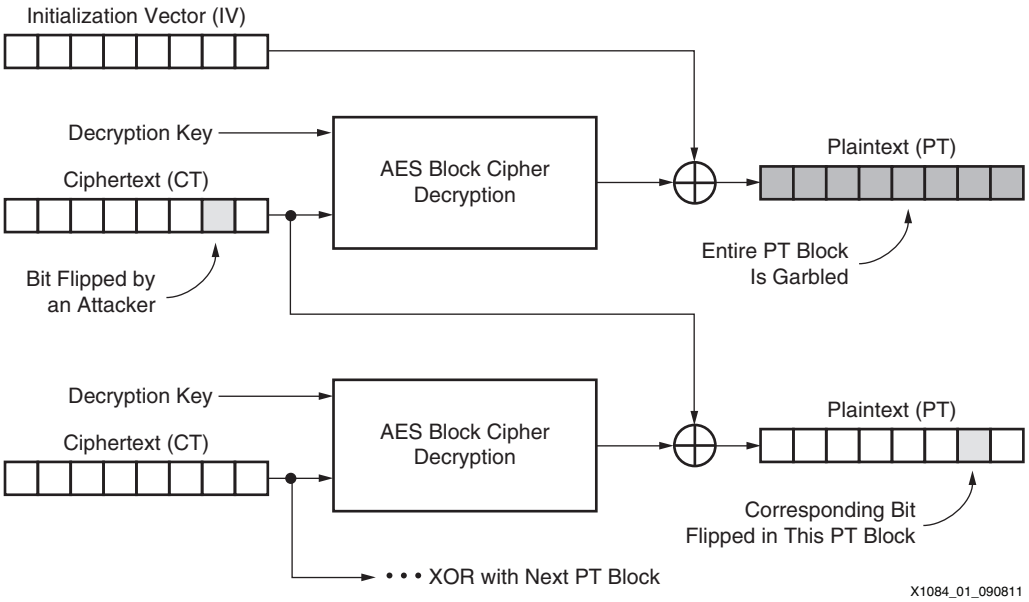

図 1 : AES-CBC ビット反転攻撃

CBC は、AES のエラーを伝播しない (non-error propagating) モードであり、Virtex-6 や 7 シリーズ FPGA で使用されるモードです (注記参照)。伝播途中にビットが反転された場合、後続のすべてのビットへ影響を与えることはありません。1 ビットが反転すると、その暗号文ブロック全体と次のブロックの関連ビットへ影響を与えます (ノイジー チャネル上を伝送する場合に最適なプロパティ)。ただし、この場合、攻撃者はビットストリーム内に小さな確定的な変更を加えることができます。この攻撃により、その他の 128 ビットがスクランブルされて、キー自体の実用性が制限されます。ビットストリーム全体の CRC32 インテグリティ チェックでも、攻撃者による変更を検出できます (ビットストリームでこのチェック機能が無効の場合を除く)。

> **注記**：ザイリンクスが FPGA デバイスを設計していた当時は、NIST 認定のその他の AES モード (AES Galois/Counter Mode (GCM) など) がなかったため、この攻撃を考慮して別の暗号化アルゴリズムを使用する選択肢がありませんでした。このタイプの攻撃には、Virtex-6 および 7 シリーズ FPGA が備える認証機能で対応できます。

この認証方法ではキーを使用するため、BitGen ソフトウェアに HMAC キーを提供する必要があります。(HMAC キーは、AES 暗号化キーと同じ NKY キー ファイル内に含まれます。HMAC キーが提供されていない場合は、Bitgen が自動的にキーを生成します。) ただし、このキーは、AES キーのように JTAG 経由で直接 FPGA デバイスへロードされません。暗号化されたビットストリーム内に含まれて保護されています。オンチップ 暗号化プロセス中に、この HMAC キーがビットストリームから抽出されて認証アルゴリズムで使用されます (つまり、HMAC キー用のオンチップ キー ストレージは不要)。ビットストリーム認証の詳細は、Virtex-6 および 7 シリーズ FPGA のコンフィギュレーション ユーザー ガイド [参照 6]、[参照 7] を参照してください。

図 2 に、受動的セキュリティ対策 (ビットストリームの暗号化および認証) が有効の場合に暗号化されないビットストリーム (平文)、暗号化されるビットストリーム (暗号文)、認証されるビットストリームに該当する部分をそれぞれ示しています。

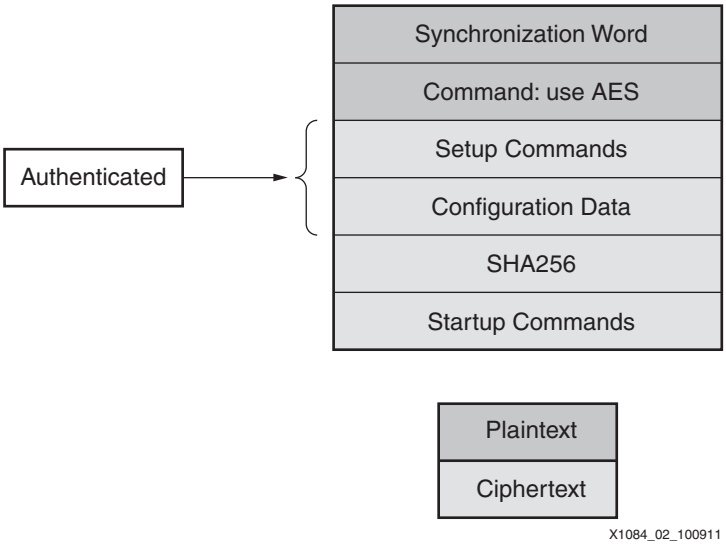

図 2 : 安全なビットストリーム形式

認証プロセスをパスしたデバイスは通常動作を開始します (スタートアップ コマンドを発行)。ビットストリームのロードが完了した後に High になるはずの DONE 出力信号、Low 駆動する INIT_B 信号、bootsts レジスタで設定される HMAC_ERROR_0 ビット (bit location = [7] HMAC_ERROR_0) が現れない場合は認証が失敗したことになります。このレジスタは、iMPACT ソフトウェアから Read Device Status コマンドを発行して JTAG 経由で読み出すことが可能です。認証失敗は、ビットストリームが不正操作されている可能性があることを示しますが、ビットストリームのロードに使用したチャネルがノイズの影響を受け、コンフィギュレーション プロセス中にビットが反転した可能性も考えられます。

図 3 に、ソフトウェアで認証コードが作成され、その後ハードウェアで検証されてデバイスの動作を許可するまでのフローを示します。

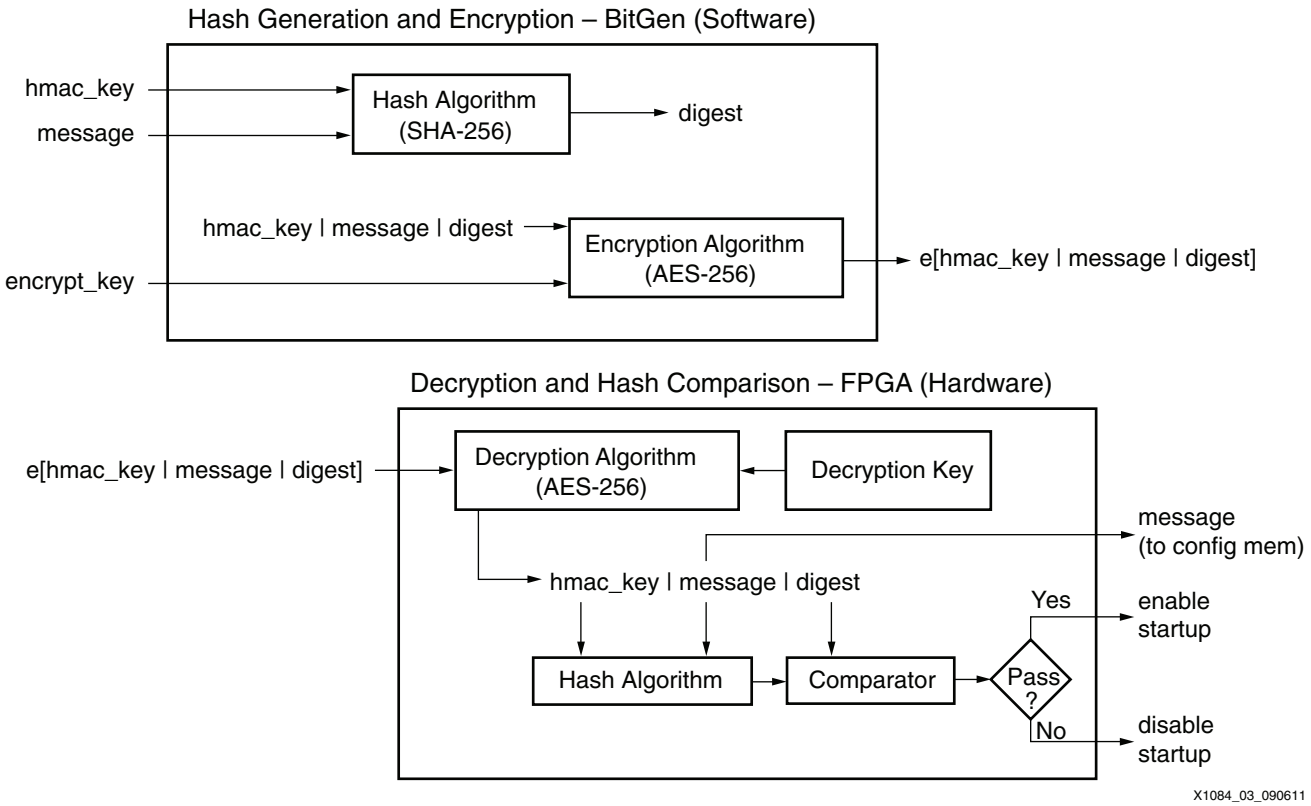

図 3 : ハッシュ化されたメッセージの認証動作

強化型リードバック無効化回路

暗号化されたビットストリームが FPGA へロードされている場合、内部コンフィギュレーション メモリのリードバックは、いかなる外部インターフェイス (JTAG など) 経由でも実行できません。外部リードバックは、強化されたトリプル リダンダント ロジックによってすべてブロック (無効化) されます。暗号化されたビットストリームがロードされた後にコンフィギュレーション メモリをリードバックできる唯一の方法は ICAP です。ICAP は FPGA ロジック内のユーザー デザインへ直接アクセスすることによってのみ利用できるため、ロード プロセスでビットストリームが認証された後は信頼されたチャネルとして許可されます。ユーザー デザインが ICAP をインスタンシエートしていない場合は、ICAP を使用できません。

> **注記 :** ビットストリーム内の特定制御ビットを使用する BitGen セキュリティ オプションでは、リードバック機能を有効/無効にできます。このビットは、コンフィギュレーション中に変更可能なため、暗号化された/認証されたビットストリームを使用していないデバイスでは攻撃者がリードバック機能の無効設定を簡単に変更できます。強化型リードバック無効化回路はそのような弱点がなく、暗号化/認証されたビットストリームを使用している場合のセキュリティ オプションよりも常に優先されます。

優れた FSM 回路

キー ロード用の FSM 回路は、ステート間で大きなハミング距離を使用することで、通常のステート マシン動作を迂回して安全でない状態に FPGA を置く可能性があるグリッチ攻撃から守ることができます。

能動的 AT シリコン機能

「概要」で言及したとおり、能動的 AT 機能を使用する場合は、ユーザーがデザインに手を加える必要があります。たとえば、不正操作に反応して KEYCLEARB 入力を駆動するように、デザインに STARTUP_VIRTEX6 をインスタンシエートする必要があります。表 4 に、これらの能動的セキュリティの機能、使用例、およびインプリメント方法を説明しています。各機能の詳細は、表 4 の後のセクションで説明しています。

表 4：能動的セキュリティ機能の使用例

機能	使用例	方法
JTAG の無効化、JTAG の動作監視	認証されていない JTAG アクセスを検出して阻止します。	FPGA ロジックに特別な方法で BSCAN プリミティブをインスタンシエートして、監視/応答機能を追加します。
内部コンフィギュレーション メモリの完全性	コンフィギュレーション メモリの完全性をバックグランドでチェックします (干渉なしのランタイム チェック)。	POST_CRC コンフィギュレーション ユーザー制約を有効にして、FRAME_ECC プリミティブをインスタンシエートします。コンフィギュレーション メモリの完全性をチェック/応答する FPGA ロジックを構築します。
オンチップ温度および電圧の監視/警告 (SYSMON)	デバイスが通常の環境範囲内で動作するように監視します。	システム モニター プリミティブをインスタンシエートして、環境状態をチェック/応答する FPGA ロジックを構築します。
PROG インターセプト (PREQ/PACK)	デバイス コンフィギュレーションに遅延を与えて、リセット不可能なデータ エレメントを消去します。	STARTUP プリミティブをインスタンシエートして、PROG_REQ を受信した後の PROG_ACK アサートの正しい条件を決定する FPGA ロジックを構築します。 この機能は、暗号化されたビットストリームに対してのみ使用できます。
固有識別子 (Device DNA/ ユーザー eFUSE)	固有識別子が認識されない場合には、デザインの動作を不可にします (または制限付きで許可します)。	固有識別子を読み出し/処理し、それらが有効であるかを判断できる FPGA ロジックを構築します。
内部コンフィギュレーション メモリの消去	不正操作に反応してコンフィギュレーション メモリを消去します。	STARTUP プリミティブをインスタンシエートして、PROG_REQ 受信後の PROG_ACK アサートの正しい条件を決定する FPGA ロジックを構築します。
内部 AES-256 キーの消去 (KEYCLEARB)	不正操作に反応してバッテリー バックアップのキーを消去します。	STARTUP プリミティブをインスタンシエートして、KEYCLEARB アサートの正しい条件を決定する FPGA ロジックを構築します。 この機能は、暗号化されたビットストリームに対してのみ使用できます。
グローバル トライステート (GTS)	不正操作に反応して出力を無効にします。	STARTUP プリミティブをインスタンシエートして、GTS アサートの正しい条件を決定する FPGA ロジックを構築します。
グローバル セット/リセット (GSR)	不正操作に反応して、ユーザー フリップフロップを初期状態に戻します。	STARTUP プリミティブをインスタンシエートして、GSR アサートの正しい条件を決定する FPGA ロジックを構築します。

JTAG の無効化 (防止)

通常、攻撃者はシステムへ侵入しようとするとき、まず JTAG ポートを狙います。そこで JTAG チェーンを切断するためにビルトインの JTAG プリミティブ (BSCAN) を特別な方法でインスタンシエートして、未認証アクセスを監視します。Virtex-6 FPGA の場合、プリミティブ名は BSCAN_VIRTEX6 となります。7 シリーズ FPGA の場合のプリミティブ名は BSCANE2 となります。

FPGA (および FPGA デバイスと同じチェーン内にあるその他の上位デバイスまたは下位デバイスのいずれか) の JTAG チェーンを切断するには、次の 2 つの方法があります。

1 つ目は、ユーザー デザインに BSCAN プリミティブをインスタンシエートして DISABLE_JTAG 属性を TRUE に設定する方法です。Virtex-6 FPGA デザインの VHDL インスタンシエーションの例を示します。

```
BSCAN_VIRTEX6_U0 :BSCAN_VIRTEX6
  generic map (
    DISABLE_JTAG => TRUE,
    JTAG_CHAIN => 1 -- can be 1, 2, 3, or 4 depending on the
                    -- desired physical location of the primitive
  )
  port map (
    CAPTURE => open,
    DRCK    => open,
    RESET   => open,
    RUNTEST => open,
    SEL     => open,
    SHIFT   => open,
    TCK     => tck_signal,
    TDI     => tdi_signal,
    TMS     => tms_signal,
    UPDATE  => open,
    TDO     => '1'
  );
```

注記：TCK、TDI、および TMS 信号は JTAG 動作の監視に使用できるため、上記ではこれらの信号のみ接続されています (詳細は 9 ページの「JTAG の監視 (検出)」を参照)。

2 つ目は、BitGen 実行中に -g DISABLE_JTAG オプションを追加する方法です。いずれの方法を使用しても JTAG を切断できます。1 つ目の方法ではユーザー デザインによって確実にチェーンが切断されますが、2 つ目の方法ではツール フロー内でチェーンを切断する判断を遅らせることができます。この判断を下すタイミングは、システム設計者が決定します。

注記：BitGen オプションが設定されていて、攻撃者がこのオプションをオフにしようとした場合、このビット反転攻撃が認証ステップで検出され、デバイスはスタートアップを開始しません。

FPGA ロジック デザインにソフト ChipScope™ Pro ロジック アナライザー コア[参照 8] をインスタンシエートした場合、このコアは、ホストへ送り返す通信パスとして JTAG ポートを使用するため、JTAG チェーンを切断すると ChipScope Pro ロジック アナライザー コアを使用できなくなります (JTAG ポートを介して通信するその他のプロセッサ デバッガーをすべて含む)。FPGA をデバッグする間は JTAG チェーンをそのままの状態にし、ChipScope Pro アナライザーの機能 (またはプロセッサ デバッガー) が不要になる開発段階の後期に切断できます。

JTAG チェーンを切断するオプションが有効に設定されている間は、デバイスの JTAG コンフィギュレーションを使用できません。ユーザーは、シリアル、シリアル パラレル インターフェイス (SPI)、バイト パラレル インターフェイス (BPI)、SelectMAP など、別のコンフィギュレーション インターフェイスを使用する必要があります [参照 6][参照 7]。FPGA デバイスのコンフィギュレーションが実際に遅延されている限り、JTAG バウンダリ スキャン ボード レベル テストは通常に動作します。

JTAG の監視 (検出)

デバイス内から JTAG アクティビティを検出するには、BSCAN プリミティブの JTAG TCK、TDI、または TMS 信号の組み合わせを監視します。外部 JTAG コマンドは、これらの信号をトグルする必要があるため、これらの信号ラインの動作検出回路がトグルを検出します。たとえば、TDI 信号の立ち上がりエッジを監視する場合は、図 4 の回路を使用できます。

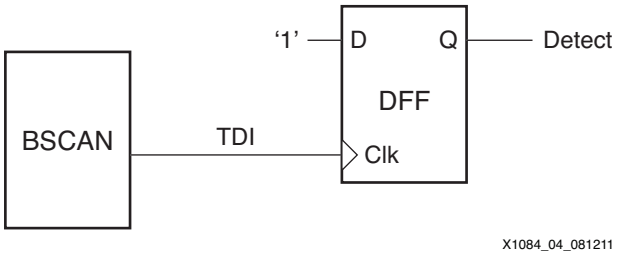

図 4 : JTAG アクティビティ検出回路の例

TDI ラインで立ち上がりエッジがラッチされると、DFE の Detect 出力は、デバイスが再コンフィギュレーションされるまで (つまり、PROGRAM_B 入力がアサートされるまで) 1 を保持します。この方法は、TCK、TDI、TMS 信号ラインの立ち上がり/立ち下がりエッジを監視する場合にも適用できます。JTAG 動作検出回路のいずれかの DFF 出力が 1 にセットされた場合には、それらに応じて不正操作ペナルティを適用できます。

内部コンフィギュレーション メモリの完全性 (検出)

内部コンフィギュレーション メモリ セル (暗号化されたビットストリームでコンフィギュレーションされている) の一部が破損していると、FPGA が予期しない不正な動作を実行する可能性があります。データ破損は、意図的な不正操作または意図的でないシングル イベント アップセット (SEU) が原因となります。リードバック CRC (巡回冗長検査) 機能を使用することによって、ユーザー デザインのバックグランドでコンフィギュレーション データが継続的にリードバックされて、反転したビットを検出できます。この機能は、FRAME_ECC プリミティブと併用して SEU 訂正などの高度な動作を行うことができます。

インプリメンテーションの詳細は、Virtex-6 および 7 シリーズ FPGA のコンフィギュレーション ユーザー ガイド ([参照 6]、[参照 7]) の第 9 章「リードバック CRC」を参照してください。その他に、ザイリンクスは Virtex-6 デバイス用に Soft Error Mitigation (SEM) コア [参照 11] も提供しています。このコアは、単一 IP コアに検出、訂正、分類機能を統合しています (エラー挿入を含むため、システムのあらゆる側面を評価可能)。

最小限の手段として、FPGA コンフィギュレーション メモリ内の反転 (破損) ビットを検出するために、直接 POST_CRC チェック (コンフィギュレーション メモリ上の相対的な CRC チェック) を実行して、システム レベルで対応することが可能です。ただし、この最小限の手段では、ユーザーがビット エラーをピンポイントで特定できず、エラーを修正できません。

温度および電圧の監視/警告 (検出)

攻撃者は、FPGA の通常動作時の電圧や温度を変更してデバイスに予想外の動作をさせることで、その中からデータを抽出したり、特定のセキュリティ機能を迂回するように仕掛ける可能性があります。たとえば、『FIPS 140-2 - Security Requirements For Cryptographic Modules (暗号モジュールに関するセキュリティ要件)』[参照 12] では、「特に、暗号化モジュールは、指定された通常動作範囲外の動作温度や電圧の変動を監視し、適切に対応する必要がある」と規定されています。

この規格要件を満たすには、オンチップ ハード IP ブロック (Virtex-6 FPGA システム モニター (SYSMON) [参照 13] または 7 シリーズ FPGA XADC (ザイリンクス アナログ デジタル変換器) [参照 14] を使用できます。これらは両方とも FPGA 内部にあるマルチチャネル ADC であり、オンチップ電源電圧やオンチップ ダイ温度、および FPGA へ供給する外部アナログ電圧を監視することが可能です。また、これらはユーザー デザインに簡単にインスタンシエートできます。このようなオンチップの監視機能は攻撃者が簡単に手を加えることができないため、オフチップよりも安全です。

オンチップのパラメーターを監視する SYSMON や XADC には、上限/下限の警告機能を直接プログラムできます。さらに、FPGA ロジックを使用して外部電圧入力のアラーム リミットを作成することも可能です (例 : 外部アナログ電圧タンパー ループ、電圧センサーの出力)。ユーザー デザイン/システムで

アラーム信号のステータスを使用し、それらがアクティブになると適切に対応できます (例：適切な不正操作ペナルティを講じる)。アナログ入力の帯域幅は制限されています (最大入力周波数の詳細は、『Virtex-6 FPGA システムモニター ユーザーガイド』[参照 13] および『7 シリーズ FPGA デュアル 12 ビット 1MSPS アナログ/デジタルコンバーター ユーザー ガイド』[参照 14] を参照)。

温度や電圧の変化を素早く検出しなければならない場合は、オフチップ ソリューションが必要になります。検出に必要な帯域幅の指定はシステム設計者の判断に委ねられています。『The Sorcerer's Apprentice Guide to Fault Attacks』[参照 15] には、チップを狙うさまざまな攻撃について説明しています。それらの一つに電源電圧の変動があります。

固有識別子 (検出)

固有識別子 (UI) には、Device DNA とユーザー eFUSE の 2 種類あります。これらは、アンチクローンセキュリティ対策 (つまり、ユーザーのビットストリームを盗んで自分のデバイスをプログラムする行為への対策) として使用したり、また UI の値に基づいて特定機能を有効化/無効化 (アップグレード/ダウングレード) するために使用できます。

Device DNA は、57 ビットのデバイス固有のシリアル番号で構成されており、生産過程で FPGA 上の OTP (ワンタイム プログラマブル) ヒューズにザイリンクスが設定します (FPGA ロジックによるこの値へ読み出しアクセスは、DNA_PORT プリミティブ経由で実行、または JTAG を介して外部から読み出しを実行)。ユーザー eFUSE には、ユーザーが読み出し/書き込み可能な 32 ビットの OTP エリアがあり、JTAG 経由でユーザーが設定します (FPGA ロジックによるこの値への読み出しアクセスは、EFUSE_USR プリミティブ経由で実行します)。これらの UI は個別に使用できますが、セキュリティレベルに応じて組合わせて使用することも可能です。

注記：Device DNA やユーザー eFUSE を使用することで固有の ID を持つことになりますが、暗号化のように高い機密性や認証機能 (例 : AES-256/HMAC) が備わるわけではありません。クローン対策には、AES-256 暗号化の使用が最も効果的です。設計者は、その上にこれらの UI を利用することで、AT 対策に対してもう一重のセキュリティを追加できます。

これらの UI は、1 つのデバイスへビットストリームを関連付けるために使用できます (これは Device DNA の場合。ユーザー eFUSE の場合は複数デバイスへ関連付ける)。ユーザーが FPGA デザイン内に UI の比較機能を構築し、比較結果に基づいて FPGA の動作が制御されます。たとえば、UI 比較でエラーが生じた場合に動作開始を拒否したり、制限付きで動作許可することができます。UI の使用例を次に示します。

1. セットアップ : JTAG 経由で FPGA から UI 値を読み出し、UI 値からチェックサムを生成して FPGA へアクセスできるフラッシュ デバイスへ格納 (一方通行機能を使用 - キー タイプが最も安全) します (図 5 参照)。

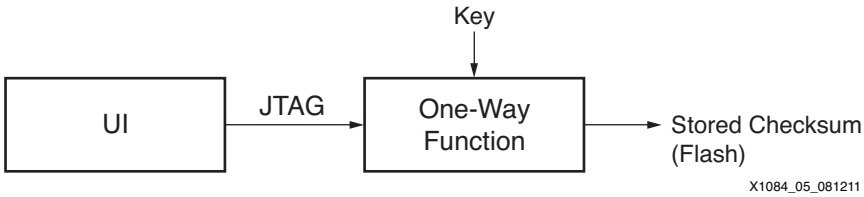

図 5 : 秘密のキーを使用して DNA 値を暗号化

図 5 に示す一方通行用のキーは、暗号化されたビットストリーム内に格納できます。ビットストリームの暗号化を行わずにこの方法でキーの機密性を確保するにはビットストリームを複雑化する必要があります。

2. FPGA をコンフィギュレーションします。
3. 比較 : FPGA ユーザー デザインが DNA_PORT プリミティブまたは EFUSE_USR プリミティブのいずれか一方または両方から UI 値を読み出し、同じアルゴリズムを使用してチェックサムを計算

します。その後、ユーザー デザインは計算されたチェックサムとフラッシュから読み出したチェックサムを比較します。チェックサムが一致した場合はデザインの通常動作が許可されます。

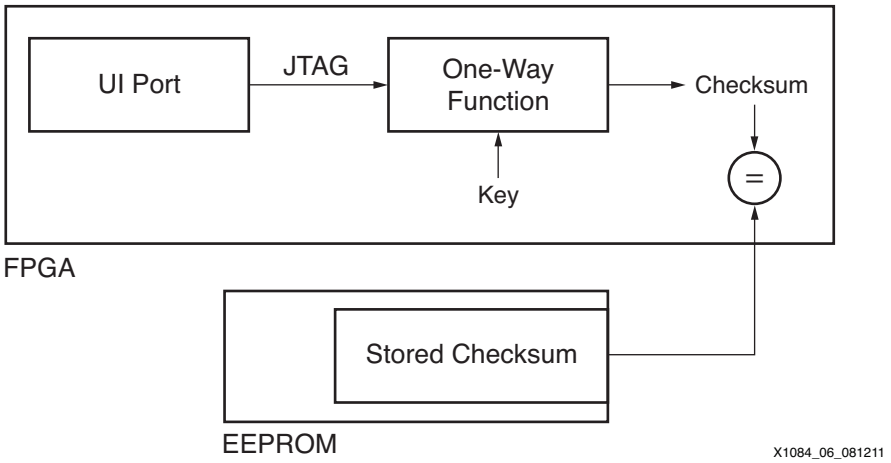

図 6 : チェックサムの比較

これらの UI の詳細は、Virtex-6 FPGA および 7 シリーズ FPGAのコンフィギュレーション ユーザー ガイド ([参照 6]、[参照 7]) を参照してください。その他に、『Spartan-3 ジェネレーション FPGA を使用したセキュリティ ソリューション』[参照 16] では Spartan FPGA デバイスの Device DNA について説明しています。

クリティカルなユーザー クロック/信号の監視 (検出)

ビットストリームの暗号化を使用する場合、ユーザーは CFGMCLK (コンフィギュレーションの内部オシレーター クロック出力) という連続したクロック ソースを活用できます。このクロックは、STARTUP ブロック プリミティブの出力として配置されています (「PROG インターセプト (防止/検出)」の STARTUP ブロックの VHDL インスタンシエーション例を参照)。このクロックは常に駆動しているため、ユーザー クロック (またはその他のクリティカルなユーザー信号) のベースとして使用し、ファンクションを監視できます。CFGMCLK は、公称値 50MHz から大幅に変動する可能性がありますが (Virtex-6 の場合は ±55%、7 シリーズおよび Zynq デバイスの場合は ±50%)、それでもファンクションを監視する上では非常に有効で、クリティカルなユーザー クロックや信号は有効な動作を続け、下限と上限の周波数範囲内で確実にトグルします (CFGMCLK の変動を考慮)。クリティカルなユーザー クロックや信号がこの範囲から外れた場合は、ユーザー デザインの故障または不正操作のいずれかが生じていることを示し、適切な対策を講じることができます。

PROG インターセプト (防止/検出)

コンフィギュレーション時にすべてのメモリ エレメントが消去されるわけではありません。たとえば、マイクロプロセッサのキャッシュや FIFO を使用するギガビット スピード シリアル I/O (GTX および GTH トランシーバー) は、外部 PROGRAM_B ピンがアサートされた後でもそのままの状態を保持して使用される場合があります (PROGRAM_B がアサートされると、FPGA はリセットされて再びビットストリームでコンフィギュレーションされる) 。攻撃者は、PROGRAM_B ピンをアサートして、ユーザー ビットストリームを独自のビットストリーム (FPGA がコンフィギュレーションされた後に保持されたメモリ内容をすべて消去するビットストリーム) に置き換えることができます。PROG インターセプト機能 (STARTUP ブロックにある PROG リクエスト/アクノレッジ信号ペア - PREQ/PACK) を使用することで、これらのメモリ エレメントが先にクリアされるまで (あるいは、PROGRAM_B がアサートされる前に実行すべきハウスキーピング タスクが実行されるまで) FPGA の再コンフィギュレーションを永久的に遅延できます。

もう一つの使用例として PROG インターセプト機能は、動作中に PROGRAM_B ピンのアサートが行われるべきではない配備後のシステムに有効です。この信号が確認されると不正操作が行われたことを

示します。おそらく攻撃者が最初にする行為の一つは、PROGRAM_B をアサートして FPGA のスタートアップ動作を観察することです。ユーザー デザインで PREQ のアサートを検出すると、ペナルティ信号がアサートされます。その後、PACK をアサートすることで PROGRAM_B がアサート可能になります。

暗号化されたビットストリームがロードされて PROGRAM_B ピンが外部でアサートされる (または内部 IPROG コマンドが ICAP に送信された) 場合は、STARTUP ブロックの PREQ 出力がアサートされます。コンフィギュレーションは、ユーザーが PACK 入力を High (立ち上がりエッジを採用) に駆動するまで、またはデバイスの電源が一度オフになって再度オンになるまで延期されます。

一般的なセキュリティ設定で、PREQ/PACK 信号が接続されている STARTUP ブロックの VHDL インスタンシエーション例を次に示します。

```
STARTUP_U0 :STARTUP_VIRTEX6
  generic map (
    PROG_USR => TRUE ) -- turn on PROGRAM_B intercept security feature
  port map (
    CFGCLK    => open,
    CFGMCLK   => cfgmclk_signal,
    DINSPI    => open,
    EOS       => open,
    PREQ      => preq_signal, -- PROGRAM request to FPGA logic output
    TCKSPI    => open,
    CLK       => '0',
    GSR       => gsr_signal,
    GTS       => gts_signal,
    KEYCLEARB => keyclearb_signal,
    PACK      => pack_signal, -- PROGRAM acknowledge input (rising edge)
    USRCCLKO  => '0',
    USRCCLKTS => '0',
    USRDONEO  => '0',
    USRDONETS => '0'
);
```

キーの消去 (応答/ペナルティ)

FPGA で最も重要な部分は、ビットストリームの暗号化に使用する AES キーです。攻撃者がキーを利用できるようになると、元のビットストリームの内容を簡単に解読されてしまいます。システム設計者は、ユーザー デザインを KEYCLEARB 入力 (12 ページの「PROG インターセプト (防止/検出)」の STARTUP ブロック プリミティブにある) へ接続することで、内部 (外部) 不正操作に反応するペナルティ信号としてこの信号をアサートできます。BBRAM ブロックの 256 ビット キーと暗号解読ブロックの 1920 ビット拡張キーの両方が消去されると、攻撃者はオフチップの不揮発性メモリに格納されている暗号化ビットストリームを使用できなくなります。この KEYCLEARB 信号は、不揮発性 eFUSE キーには影響を与えません。

KEYCLEARB 信号のアサート判断 (および不正操作に反応するその他すべての応答) は、必ずしも FPGA 内部から生じるとは限りません。システム内の別の部分の不正操作 (例 : システム レベルまたはモジュール レベルのタンパー バウンダリの侵害) に起因する可能性もあります。

キーが消去されると、同じキーで再プログラムされるまで、あるいは IPROG ([参照 6]、[参照 7]) を使用して別のビットストリーム (暗号化されていない基本機能のみを含むビットストリーム) で再コンフィギュレーションされるまで FPGA デバイスは動作できません。KEYCLEARB 入力は、適切な条件の下でのみアサートされるように保証されなければなりません。通常は、このような状態になった装置は、取り外して中央拠点 (製造施設など) へ移動し、キーをロードして再び有効にする必要があります。

IPROG (応答/ペナルティ)

IPROG は、ICAP インターフェイスを介して送られる内部コマンドで、FPGA コンフィギュレーション メモリ、すべてのフリップフロップ、および拡張メモリを消去しますが、キー自体は消去しません。外

部 PROGRAM_B ピンのアサートと同じです。このコマンドは、コンフィギュレーション メモリ (コンフィギュレーション データ、ブロック RAM、およびフリップフロップのステート) を効率的に消去するため、KEYCLEARB 信号と組み合わせて使用して不正操作に対応できます。KEYCLEARB は、IROG コマンドが発行される前にアサートされます (KEYCLEARB がアサートされると直ぐに IPROG コマンドを ICAP へ送信可能)。

これらの 2 つのペナルティが適用された後は既存ビットストリームが FPGA で解読されないため、FPGA は動作できない状態になります。デバイスを (暗号化ビットストリームで) コンフィギュレーションできない場合は、不正操作が生じたことを表します。この場合、ユーザー デザインには暗号化されていないビットストリームをロードする選択肢があるため、KEYCLEARB/IPROG が実行された後でも CT を露出することなく、基本機能 (少ない機能) を取り込むことができます。

IPROG コマンドをコンフィギュレーション エンジンへ送信するには、ユーザー デザインに ICAP プリミティブをインスタンシエートして、適切なコマンド シーケンスを記述する必要があります。詳細は、Virtex-6 FPGA または 7 シリーズ FPGA コンフィギュレーション ユーザー ガイド [参照 6]、[参照 7] の「IPROG リコンフィギュレーション」セクションを参照してください。

グローバル トライステート (応答/ペナルティ)

システム デザインによっては、外部の FPGA ピン (例 : 暗号モジュール) からクリティカルな (Red) 情報を読み出すことができます。不正操作に反応して STARTUP ブロック (12 ページの「PROG インターセプト (防止/検出)」の「グローバル リセット (応答/ペナルティ)」セクションにあるサンプル コードを参照) のグローバル トライステート (GST) 入力がアサートされると、すべての FPGA 出力がハイインピーダンス状態になり、FPGA 外部へのデータ フローが停止します。これは、Red データ フローをできるだけ早く停止するために、IPROG や KEYCLEARB よりも前に講じる即効プロセスです。

グローバル リセット (応答/ペナルティ)

大切なデータや重要度の高いパラメーターは、ユーザー キーなどの FPGA ロジック レジスタ内 (AES BBRAM ではない) に格納できます。不正操作に反応して、STARTUP ブロック (12 ページの「PROG インターセプト (防止/検出)」参照) のグローバル リセット (GSR) 入力がアサートされると、すべての FPGA レジスタ (例 : フリップフロップ) がデフォルト状態に戻ります。これは、FPGA 内にあるすべての重要なデータをできるだけ早く消去するために、IPROG や KEYCLEARB よりも前に講じる即効プロセスです。GSR は、SRL (シフト レジスタ ルックアップ テーブル) やブロック RAM のコンテンツを消去しません。これらを消去する場合は、ユーザー デザインまたは IPROG コマンドを使用する必要があります。

不正操作防止の ガイドライン

このセクションでは、前述したビルトインのシリコン AT 機能と共に、Virtex-6 FPGA や 7 シリーズ FPGA を使用して耐タンパー性の (不正操作されにくい) デザインを作成するためのガイドラインおよび技術的ヒントを提供します。

必要に応じて CT のみロードする (防止)

ユーザー デザインをノンクリティカル テクノロジ ブロックとクリティカル テクノロジ (CT) ブロックに分割できる場合は、ユーザー デザインのノンクリティカル部分のみを常に保有し、必要なときに FPGA のパーシャル リコンフィギュレーション (PR) 機能 [参照 17] を使用して CT をロードできます。その後、CT の役割が終わると、(PR 領域のブラック ボックス バージョンをロードして) 消去できるようになります。CT 用の部分的なビットストリームは、AES キーで暗号化できますが、システム設計者が選択するアルゴリズムによって FPGA ロジック内で暗号化することも可能です。不正操作が検出されると、CT の PR 領域とキー (通常、ブロック RAM または FPGA ロジック レジスタ内に格納されている) は消去されます。

例として、15 ページの図 7 および 15 ページの図 8 に、FPGA、CPU、外部メモリ デバイス (FPGA コンフィギュレーション、PR、CPU コード、および CPU ブート コード用) で構成される一般的なシステムを示します。図 7 の PR 領域 (「User CT Logic Region」と表示) は空の状態です。

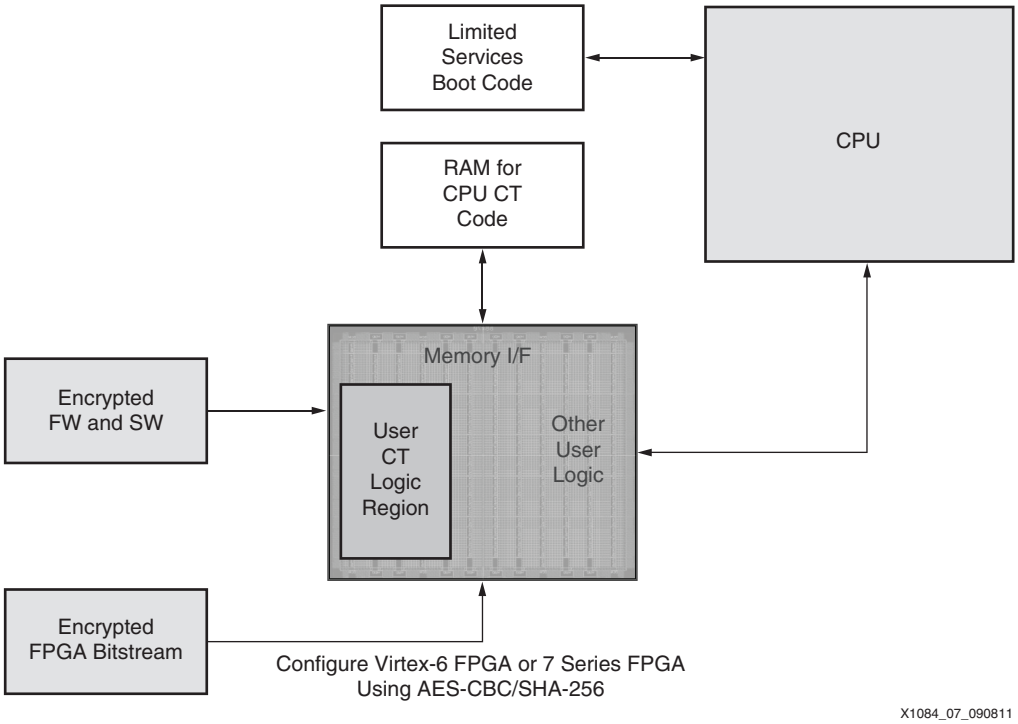

図 7：使用モデル：システムの CT を保護

図 8 の場合、FPGA のこの領域には、PR によって CT ロジック (User CT Logic) が動的にロードされます。CT の役割が完了すると (または、不正操作が生じた場合)、PR モジュールのブラックボックスバージョンをロードすることで PR 領域は図 7 の状態に戻ります。

注記：不正操作が検出されると、重要度の高い CPU コード メモリも消去されるため、暗号化されたデータ、消去された PR 領域、あるいは重要度の低い外部メモリ コンテンツのみ残ります。

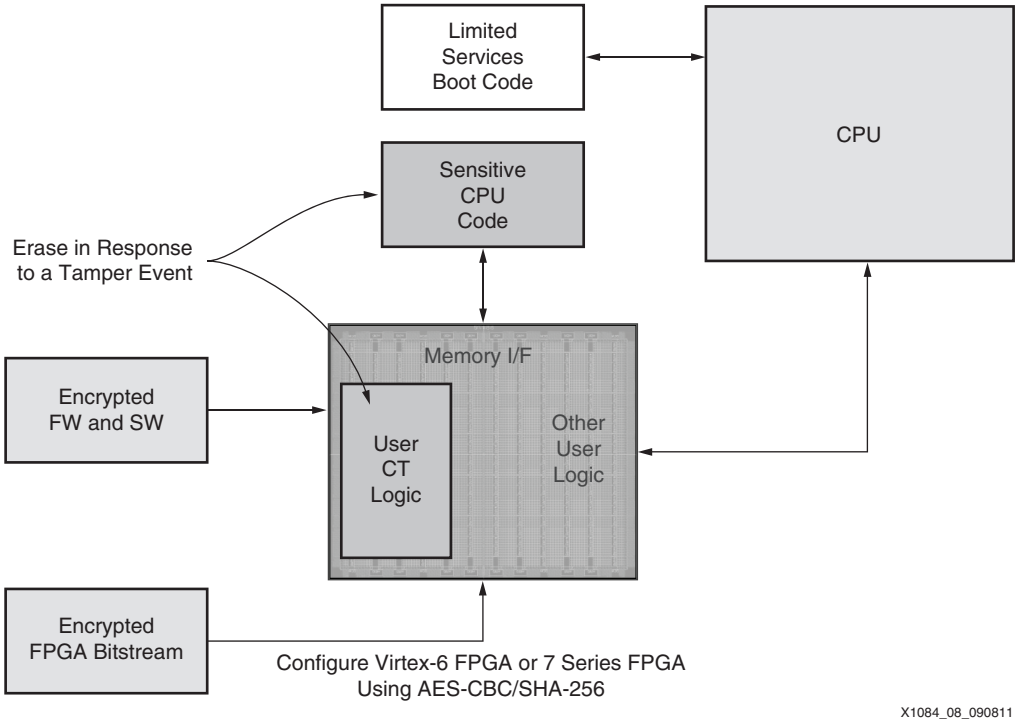

図 8：使用モデル：システムの CT を保護 - 不正行為に対応

これらの例では、PR の実行に ICAP が使用されます。ICAP は信頼性の高いチャネルであるため、PR ビットストリーム (暗号化/非暗号化) を許可します (cfg_aes_only eFUSE ビットが設定された eFUSE キーが使用されている場合でも許可)。ザイリンクスでは、常に、暗号化された PR ビットストリームを使用することを推奨します (暗号解読には、FPGA のコンフィギュレーション エンジンにある専用暗号解読ロジック、または FPGA ロジックにユーザーが構築した任意の暗号解読機能を使用)。安全な PR に関するその他の情報は、『PRC/EPRC : パーシャル リコンフィギュレーションのデータ インテグリティおよび セキュリティ コントローラー』[参照 18] を参照してください。

外部シャントによるキーの消去

BBRAM キーを消去するもう一つの方法として、外部シャントを使用して V_{BATT} ラインをグランド接続する方法があります。KEYCLEARB などのアクティブ機能は FPGA に電源が投入されてコンフィギュレーションされた後にのみ使用可能なため、FPGA の主電源 (V_{CCINT} および V_{CCAUX}) が供給されない場合はこの方法を用いることができます。たとえば、FPGA の主電源がオンになる前にシステムレベルの不正操作が検出された場合、FPGA の外部バッテリー電源ライン V_{BATT} ピンをオープンにし、トランジスタ分岐 (シャント) などで V_{BATT} をグランドに接続します。この場合、V_{BATT} ピンをグランド接続する前にバッテリー接続がオープンになるように回路を設計する必要があります。もう一つのオプションは、抵抗を介してバッテリーを V_{BATT} へ接続する方法です (V_{BATT} の最大入力電流は 150nA)。適切な抵抗値を選択することで、バッテリーから過剰な電流フローを生じさせることなく V_{BATT} ピンをグランドに短絡できます。

FPGA に電源が投入されない場合 (V_{CCINT} や V_{CCAUX} などがない場合、ただし V_{BATT} はこれらに含まれない)、ユーザーが V_{BATT} ピンをすぐにグランドに短絡して AES キー (BBROM 内に格納) が消去されるまでのワーストケースの最大時間は 50ms です (–55°C)。また、V_{BATT} のみ電源を入れた場合、BBRAM キーが保持されるストレージの温度範囲は –65°C ～ 150°C です。

予防策としてあらかじめキーを消去

その他のキー消去オプションには、予防策としてあらかじめ消去する方法があります。

ビットストリームをロードして暗号解読した後、システムを配備する前に BBRAM キーを意図的に消去できます。この方法は、配備された後に電源を再度オン/オフする必要がないシステム (発射後のミサイルなど) にのみ有効です 。これは、eFUSE キーに使用することも可能で、eFUSE キーの読み出し/書き込み機能が無効になっていない場合に限り、配備する前に JTAG を介してすべて 1 に書き換えることで完了します。

反復パターンなどセキュリティ レベルの低いキーを使用しない

ユーザー キーには、すべて 0、すべて 1、反復パターンなどを使用すべきではありません。可能な限りキーの再利用 (AES や HMAC と同じキーを使用するなど) は避けてください。キーの利用者を厳しく制限する必要があります (つまり、キー情報を知る必要がある人のみアクセスできるようにする)。ランダムな値を使用してキーを作成することが理想的です (セキュリティ レベルの低いキー値を避ける。たとえば、すべて 0 のランダム キーは単純過ぎる)。BitGen ソフトウェアで AES や HMAC キーを生成できますが、この場合、日付や時間に基づいた疑似ランダム プロセスを使用します。最も安全なキーを作成するには、完全に不規則で複雑な値を使用してください。

暗号化システムにおいて、キー管理は非常に重要なエレメントです (おそらく、最も複雑なエレメントとなる)。キーに関するその他の役立つ情報として、NIST キー管理のガイドライン[参照 19] を参照してください。

フィールド (現場) でキーを変更

最も明確な方法は、iMPACT ソフトウェアを使用して FPGA にキーをロードし、FPGA がこの新しいキーで暗号化されたビットストリームに対応する方法です (この場合、JTAG プログラミング ケーブルでアクセスできるポータブル コンピューターが必要)。もう一つの方法は、エンベデッド マイクロコントローラー [参照 20] またはエンベデッド JTAG ACE Player [参照 21] を使用するザイリンクスのイン

システム プログラミング機能を使用する方法です。ユーザーが iMPACT を使用して生成した -keyonly オプションに基づいて、隣接するプロセッサから JTAG シーケンスを開始し、これによって BBRAM のキーがアップデートされます。これは Red キーをロードする (キーは暗号化されていない) ため、Red キー情報を FPGA の JTAG ポートへ転送する間の未承認アクセスから守るために適切なセーフガードを適用しなければならないことに留意してください。

不正操作ステータスをシステムへ送信

不正操作が行われた場合、ペナルティ信号のアサートのほかに、ユーザー デザインは不正操作のステータス情報をシステムへ送ることができます (または不揮発性メモリへ書き込むことが可能)。そして、今後の分析材料としてこれらの情報をシステムへ格納します。FPGA は揮発性であるため、IPROG コマンド (不正操作ペナルティ) が与えられる前に別の場所へデータを送信するように設計する必要があります。

FPGA プローブ ポイントへのアクセス制限

攻撃されにくい FPGA を構築するには、レイヤード (重構造) アプローチが最も効果的です。CT を含むすべてのデバイス周囲に強力な不正操作バウンダリ (不正操作検出スイッチなど) を使用できます (たとえば、不正操作検出スイッチがアクティベートされると、FPGA の V_{BATT} ラインがグランドに短絡される)。プリント回路基板には FPGA 信号用に埋め込み型のビアと配線を使用し、電源配線は埋め込まれたレイヤー内で配置され (アクセスが困難になる)、適切なデカップリングを行います (可能であれば Buried Capacitance テクノロジを使用)。JTAG バウンダリ スキャン技術は、ボードレベルのファクトリ テストで信頼性が確保されているため、量産ボードからテスト ポイントを削除してください。

差分電力解析 (DPA)

半導体デバイスの DPA 攻撃は、長年に渡り検出されている攻撃の種類です [参照 22]。これらの攻撃は、プロセッサ、マイクロコントローラー、ペイメントカード (スマート カード)、FPGA などさまざまなデバイスをターゲットにできます。専門誌では、(そのような攻撃を可能にする目的で特別に設計されたボードを使用して)、FPGA へ DPA 攻撃を実行してビットストリームの暗号解読キーを抽出することに成功した例が発表されています。

このような攻撃は、技術的に実行可能である上に専門的な知識や装置がなくても、また物理的にデバイスへ接近しなくても実行できる厄介な問題です。デザインのセキュリティを懸念するシステム設計者は、アーキテクチャやデザイン ファイル、物理的装置へのアクセス、適切と思われるその他の対策を守るために包括的な (多重) セキュリティ戦略を講じる必要があります。ザイリンクスは、今後もこの分野の研究と開発を続けていきます。詳細は FAE へご連絡ください。

まとめ

このアプリケーション ノートでは、FPGA 設計者が現在利用できる Virtex-6 および 7 シリーズ FPGA の AT 機能について説明し、それらを効果的に使用する実際の例を紹介しました。ここで紹介したこれらの AT 機能を活用してガイドライン (設計サイクル初期) に従うことで、不正操作防止機能を備えた FPGA デザインが実現します。

1 つの AT 機能や技術だけでは常に 100% の効果を実現したりシステム全体のすべての AT ニーズを満たすことは不可能ですが、できるだけ攻撃者の作業を複雑化 (および高価に) させたり、何重にもセキュリティの壁を作ることで (最良とまではいかなくても) ほぼ確実に満足いく結果を得ることができるでしょう。

新しい統合回路 (FPGA など) の開発および検証に使用されるツールや技術は、日々進化して改善されています。同時に、攻撃者が使用するツールも進化と改善を繰り返しているため、FPGA 設計者は利用できる AT 機能や技術を常に把握しておく必要があります。また、ザイリンクスはこれらの開発の最先端をリードし、現在および将来に向けてカスタマー IP を守るために機能強化および新機能の開発に尽力しています。

参考資料

このアプリケーション ノートの参考資料は次のとおりです。

1. 入門
   http://japan.xilinx.com/company/gettingstarted/index.htm
2. ISE Design Suite
   http://japan.xilinx.com/support/documentation/dt_ise.htm
3. WP365 :『デザインの安全性の確保』
4. Security Monitor IP コアの製品パンフレット
   http://www.xilinx.com/publications/prod_mktg/CS1140_AD_SecMonIP_ProdBrf_Update_June_2012.pdf
5. SPI または BPI フラッシュ メモリの間接的プログラムの概要
   http://japan.xilinx.com/support/documentation/sw_manuals/xilinx11/pim_c_introduction_indirect_programming.htm
6. UG360 :『Virtex-6 FPGA コンフィギュレーション ユーザー ガイド』
7. UG470 :『7 シリーズ FPGA コンフィギュレーション ユーザー ガイド』
8. ChipScope Pro およびシリアル I/O ツールキット
   http://japan.xilinx.com/tools/cspro.htm
9. 『The Keyed-Hash Message Authentication Code (HMAC)』(FIPS PUB 198a)
   http://csrc.nist.gov/publications/fips/fips198/fips-198a.pdf
10. 『Secure Hash Standard』(FIPS PUB 180-2)
    http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf
11. Soft Error Mitigation (SEM) コア
    http://japan.xilinx.com/products/intellectual-property/SEM.htm
12. 『Security Requirements for Cryptographic Modules』(FIPS PUB 140-2)
    http://www.nist.gov/itl/upload/fips1402.pdf
13. UG370 :『Virtex-6 FPGA システムモニター ユーザーガイド』
14. UG480 :『7 シリーズ FPGA デュアル 12 ビット 1MSPS アナログ / デジタルコンバーター ユーザー ガイド』
15. 『The Sorcerer's Apprentice Guide to Fault Attacks』(Hagai Bar-El、Hamid Choukri、David Naccache、Michael Tunstall、Claire Whelan)
    http://eprint.iacr.org/2004/100.pdf
16. WP266 :『Spartan-3 ジェネレーション FPGA を使用したセキュリティ ソリューション』
17. パーシャル リコンフィギュレーション
    http://japan.xilinx.com/tools/partial-reconfiguration.htm
18. XAPP887 :『PRC/EPRC : パーシャル リコンフィギュレーションのデータ インテグリティおよびセキュリティ コントローラー』
19. NIST 社のサイト「Key Management Guideline」
    http://csrc.nist.gov/groups/ST/toolkit/key_management.html
20. XAPP058 :『組み込み型マイクロコントローラを使用するザイリンクスのインシステム プログラミング機能』
21. XAPP424 :『エンベデッド JTAG ACE Player』
22. Google 検索結果「differential+power+analysis」
    http://scholar.google.com/scholar?q=Differential+power+analysis&hl=en&as_sdt=0&as_vis=1&oi=scholart

改訂履歴

次の表に、この文書の改訂履歴を示します。

日付	バージョン	内容
2011 年 9 月 21 日	1.0	初版
2011 年 12 月 1 日	1.1	「概要」に Security Monitor IP コアの利用についての説明を追加。「ビットストリームの暗号化/復号化」に BitGen で生成されるキーについての説明を追加。「ビットストリームの認証」の注記に認証プロセスについての説明を追加。図 2 の「Authenticated」ボックスの背景色を白に変更。「強化型リードバック無効化回路」の BitGen セキュリティ オプションに関する注記を変更。表 4 の「方法」欄の「PROG インターセプト (PREQ/PACK)」と「内部 AES-256 キーの消去 (KEYCLEARB)」の内容を変更。「JTAG の無効化 (防止)」のVHDL ソース コードのフォーマットを変更。「グローバル トライステート (応答/ペナルティ)」セクションの追加。「外部シャントによるキーの消去」の最初の段落を変更。「フィールド (現場) でキーを変更」の最後の段落に Red キーのロードに関する説明を追加。「差分電力解析 (DPA)」の最後の文を変更。
2012 年 8 月 10 日	1.2	「参考資料」に「Security Monitor IP の製品パンフレット」および「SPI または BPI フラッシュ メモリの間接的プログラムの概要」へのリンクを追加。「概要」の「価値」と「敵」の説明を変更。表 2 および表 3 に注記を追加。「クリティカルなユーザー クロック/信号の監視 (検出)」を追加。表 2 と表 3 に注記を追加。「クリティカルなユーザー クロック/信号の監視 (検出)」を追加。「IPROG (応答/ペナルティ)」の内容を変更。

Notice of Disclaimer

The information disclosed to you hereunder (the “Materials”) is provided solely for the selection and use of Xilinx products.To the maximum extent permitted by applicable law:(1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same.Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications.You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent.Certain products are subject to the terms and conditions of the Limited Warranties which can be viewed at http://www.xilinx.com/warranty.htm; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx.Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in Critical Applications:http://www.xilinx.com/warranty.htm#critapps.

本資料は英語版 (v1.2) を翻訳したもので、内容に相違が生じる場合には原文を優先します。
資料によっては英語版の更新に対応していないものがあります。
日本語版は参考用としてご使用の上、最新情報につきましては、必ず最新英語版をご参照ください。

この資料に関するフィードバックおよびリンクなどの問題につきましては、jpn_trans_feedback@xilinx.com までお知らせください。いただきましたご意見を参考に早急に対応させていただきます。なお、このメールアドレスへのお問い合わせは受け付けておりません。あらかじめご了承ください。