

## Application Note

## AM62Ax デバイスの QSPI アドレッシング モードに関する考慮事項



Linjun Meng, Tao Han

Central FAE

## 概要

AM62Ax プロセッサは、起動時に固定の 3 バイト アドレッシング モードを使用するオンチップ ブート ROM を介した QSPI NOR フラッシュからのブートをサポートします。一部の大容量フラッシュ デバイスは、3 バイトと 4 バイトの両方のアドレッシング モードをサポートし、フラッシュへの電源供給が継続している場合は、リセット後もアドレス構成を保持できます。特定の条件下では、この動作によってブート ROM とフラッシュ デバイスの間に互換性の不一致が生じ、証明書認証の失敗やブートの中断につながる可能性があります。

このアプリケーション ノートでは、障害のメカニズムについて説明し、ブート ROM ログ、CCS によるメモリ検査、QSPI バス解析に基づく実用的なデバッグ手法を紹介するとともに、影響を受けるシステムの対策について説明します。この情報は、QSPI NOR フラッシュからブートする AM62Ax ベースの産業用、車載、組み込みシステムを開発するハードウェアおよびソフトウェア エンジニアを対象としています。このドキュメントのガイダンスに従うことで、設計者は起動障害を効率よく切り分け、4 バイト アドレッシング モードをサポートするフラッシュ デバイスを使用する際のブート信頼性を向上できます。

この情報は、QSPI NOR フラッシュからブートする AM62Ax ベースの産業用および組み込みシステムを開発するハードウェアおよびソフトウェア エンジニアを対象としており、起動の信頼性向上と根本原因の早期特定に活用できます。

## 目次

|                              |    |
|------------------------------|----|
| 商標.....                      | 2  |
| 1 はじめに.....                  | 3  |
| 1.1 目的.....                  | 3  |
| 1.2 範囲.....                  | 3  |
| 2 詳細説明.....                  | 4  |
| 2.1 AM62Ax QSPI ブートの概要.....  | 4  |
| 2.2 フラッシュのアドレス指定アーキテクチャ..... | 4  |
| 2.3 ブート互換性に関する考慮事項.....      | 5  |
| 2.4 起動調査の例.....              | 6  |
| 2.5 デバッグ方法.....              | 8  |
| 2.6 対策と設計上の推奨事項.....         | 9  |
| 3 まとめ.....                   | 10 |
| 4 参考資料.....                  | 11 |

## 図の一覧

|                                                |   |
|------------------------------------------------|---|
| 図 2-1. 3 バイト アドレッシング モードでの QSPI 読み取りタイミング..... | 5 |
| 図 2-2. 4 バイト アドレッシング モードでの QSPI 読み取りタイミング..... | 5 |
| 図 2-3. フラッシュのアドレッシング モード保持がブート プロセスに及ぼす影響..... | 6 |
| 図 2-4. ブート ROM の認証失敗ログ.....                    | 7 |
| 図 2-5. ブート ROM がロードしたメモリの内容.....               | 7 |
| 図 2-6. 起動時の QSPI バス キャプチャ.....                 | 8 |

## 表の一覧

|                            |   |
|----------------------------|---|
| 表 2-1. ブート前提条件チェックリスト..... | 4 |
| 表 2-2. リセット動作の比較.....      | 6 |

## 商標

Jacinto™ と Sitara™ はテキサス インスツルメンツの商標です。

すべての商標はそれぞれの所有者に帰属します。

## 1 はじめに

### 1.1 目的

QSPI NOR フラッシュは、実装面積を抑えられ、ピン数が少なく、ブートアーキテクチャがシンプルであるため、AM62Ax ベースのシステムでブートデバイスとして広く使用されています。システム開発時に、設計者はさまざまなリセット条件下で、フラッシュのアドレッシングモード動作とブート ROM の動作がどのように相互に影響するかを理解する必要があります。

一部のフラッシュデバイスは、プロセッサのリセット中でも電源が供給されている場合、構成を保持できます。保持されたフラッシュの状態がブート ROM の想定と異なる場合、ソフトウェアの実行開始前に起動時の互換性問題が発生する可能性があります。このような障害はブート ROM ステージで発生するため、従来のソフトウェアデバッグ手法では根本原因を把握しにくくなります。

このアプリケーションノートでは、これらの考慮事項について説明し、このような問題を診断して軽減するための実用的な手法を紹介します。

### 1.2 範囲

このドキュメントでは、次のものを対象としています。

- QSPI NOR フラッシュからブートする AM62Ax プロセッサ
- 3 バイトと 4 バイトの両方のアドレッシングモードをサポートするフラッシュデバイス
- MCU\_PORz またはウォームリセットメカニズムを実装するシステム
- リセット後も構成状態を保持できるフラッシュデバイス

## 2 詳細説明

### 2.1 AM62Ax QSPI ブートの概要

#### 2.1.1 ブートの前提条件

ブート ROM が QSPI フラッシュに正常にアクセスするには、次の条件を満たす必要があります。

**表 2-1. ブート前提条件チェックリスト**

| 項目     | 要件                      |
|--------|-------------------------|
| 基準クロック | 有効な MCU_OSC0_XI 入力      |
| ブート構成  | QSPI ブート用に構成した BOOTMODE |
| QSPI   | フラッシュ デバイスとの正常な通信       |

起動をデバッグする際は、最初にこれらの条件を確認する必要があります。

#### 2.1.2 ブート ROM の読み取りシーケンス

リセット解除後、AM62Ax ブート ROM は次の処理を実行します。

1. コマンド 0x9F を使用してフラッシュの JEDEC ID を読み取ります。
2. クワッド出力の高速読み出しコマンド 0x6B を発行します。
3. アドレス 0x000000 から tiboot3.bin を読み取ります。
4. 証明書およびイメージ ヘッダーを検証します。
5. 実行を次のブート ステージに移します。

ブート ROM は、次の固定通信パラメータを使用します。

- 読み取りコマンド: 0x6B
- 転送モード: 1S-1S-4S
- アドレス フェーズ: 3 バイト
- ダミー サイクル: 8
- クロック周波数: 50MHz

これらのパラメータはブート ROM の実装にハード コーディングされているため、ソフトウェアでは変更できません。

### 2.2 フラッシュのアドレス指定アーキテクチャ

#### 2.2.1 3 バイト アドレッシング モード

3 バイト アドレッシング モードでは、フラッシュ デバイスは 3 バイト アドレス フェーズを使用し、メモリ空間の先頭 16MB にアクセスできます。

ほとんどの QSPI NOR フラッシュ デバイスは、電源投入後またはハードウェアリセット後に自動的に 3 バイト アドレッシング モードに移行します。

#### 2.2.2 4 バイト アドレッシング モード

大容量フラッシュ デバイスは通常、先頭の 16MB を越えるメモリにアクセスできるように 4 バイト アドレッシング モードをサポートしています。

通常はソフトウェア コマンドでこのモードに移行します。明示的にモードを終了するかリセットするまで、このモードは有効です。

#### 2.2.3 状態保持動作

構成レジスタの動作は、フラッシュ ベンダーおよびデバイス ファミリによって異なります。

一部のデバイスでは、電源が供給されている間、構成状態が保持されます。

独立した RESET# ピンがないフラッシュ デバイスでは、フラッシュの電源が遮断されない場合、MCU\_PORz リセット後もアドレッシング モードが変わらないことがあります。

## 2.3 ブート互換性に関する考慮事項

### 2.3.1 ブート ROM の想定

起動時、AM62Ax ブート ROM は常に、フラッシュ デバイスが 3 バイト アドレッシング モードで動作していると想定します。

ブート ROM はコマンド 0x6B を発行し、その後に固定の 3 バイト アドレス フェーズと 8 ダミー サイクルが続きます。ブート ROM には、フラッシュ デバイスの現在のアドレッシング モードを検出するメカニズムはありません。

### 2.3.2 アドレッシング モード不一致のメカニズム

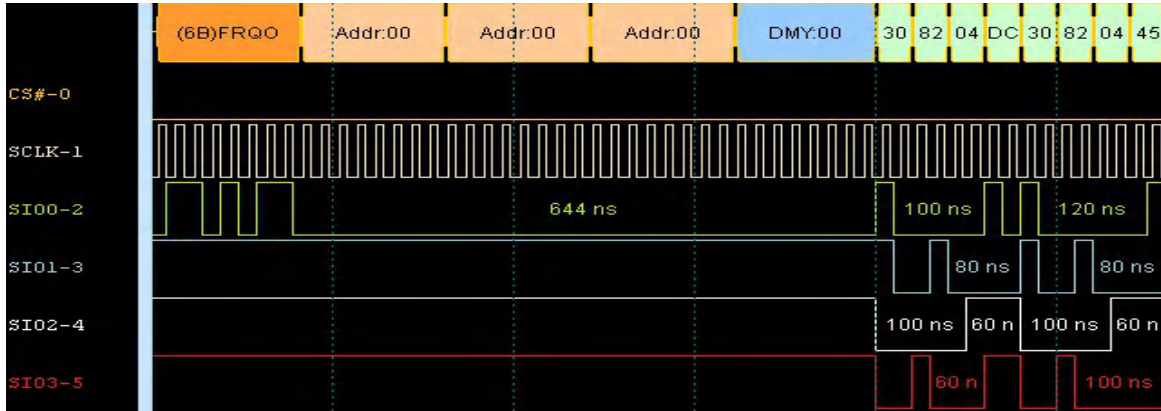


図 2-1. 3 バイト アドレッシング モードでの QSPI 読み取りタイミング

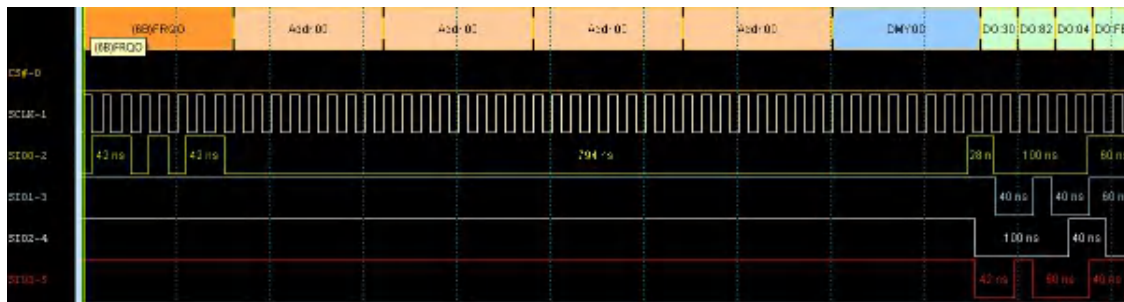


図 2-2. 4 バイト アドレッシング モードでの QSPI 読み取りタイミング

フラッシュが 4 バイト アドレッシング モードのままの場合、ブート ROM が生成したトランザクションは、意図したプロトコルとは異なる方法で解釈されます。

ブート ROM は次のデータを送信します。

コマンド + 3 バイト アドレス + ダミー サイクル + データ

ただし、4 バイト アドレッシング モードに構成されたフラッシュ デバイスは、同じトランザクションを次のように解釈します。

フラッシュは、ブート ROM のダミー期間の最後のバイトを 4 番目のアドレス バイトとして解釈します。

その結果、ブート ROM のダミー期間の一部がアドレス情報として解釈され、フラッシュのデータ出力がブート ROM のサンプリング ポイントに対してずれます。

### 2.3.3 起動への影響

アドレッシング モードの不一致により、ブート ROM がロードするイメージ内のデータ位置がずれます。

調査対象のシステムでは、イメージの内容が 1 バイトずれました。その結果、tiboot3.bin に含まれる証明書ヘッダーが想定されたオフセットからずれ、元のイメージと証明書が有効であるにもかかわらず、証明書認証が失敗しました。

発生する障害のシーケンスを図 2-3 に示します。

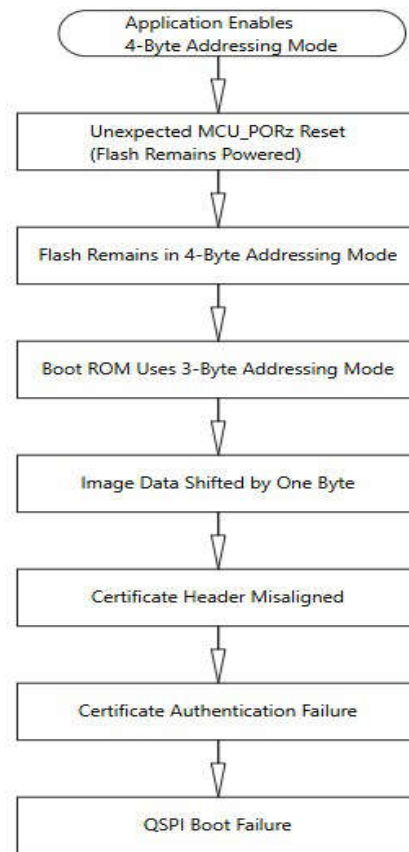


図 2-3. フラッシュのアドレッシング モード保持がブートプロセスに及ぼす影響

## 2.4 起動調査の例

### 2.4.1 外部に現れる障害の兆候

調査対象のプラットフォームでは、表 2-2 に示す動作が確認されました。

表 2-2. リセット動作の比較

| リセット条件        | 結果    |
|---------------|-------|
| コールド パワーオン    | ブート成功 |
| Linux の再起動    | ブート成功 |
| MCU_PORz リセット | 起動失敗  |

UART ブートをバックアップ ブートソースとして構成した場合、QSPI ブートに失敗すると、デバイスは自動的に UART ブートに切り替わりました。

この動作は、ブート ROM ステージで障害が発生したことを外部から確認できる兆候となりました。

### 2.4.2 ブート ROM ログ解析

図 2-4 に、デコードされたブート ROM の障害ログを示します。



```
Warning Log:
Log Max Index Size: 21
Log current entry: 4
Log buffer: 0x43c3e480
0: main.c:1397 ecode: 0x2010018, ts: 445
"Main_derPanic()!!!"
1: main.c:1752 ecode: 0xffffffff, ts: 450
"derOneParse() failed"
2: main.c:1397 ecode: 0x2010018, ts: 679
"Main_derPanic()!!!"
3: main.c:1752 ecode: 0xffffffff, ts: 683
"derOneParse() failed"
```

図 2-4. ブート ROM の認証失敗ログ

TI ROM ブート ログ ダンプ ユーティリティを使用して、ブート ROM の障害ログを取得し、デコードしました。

デコードされたログでは、証明書認証の失敗が報告されました。

このメッセージから、当初はイメージ署名または証明書生成の問題が疑われましたが、その後の検証により、元の tiboot3.bin イメージと証明書が有効であることを確認しました。そのため、認証失敗は根本原因ではなく症状と判断しました。

### 2.4.3 CCS を使用したメモリ検証

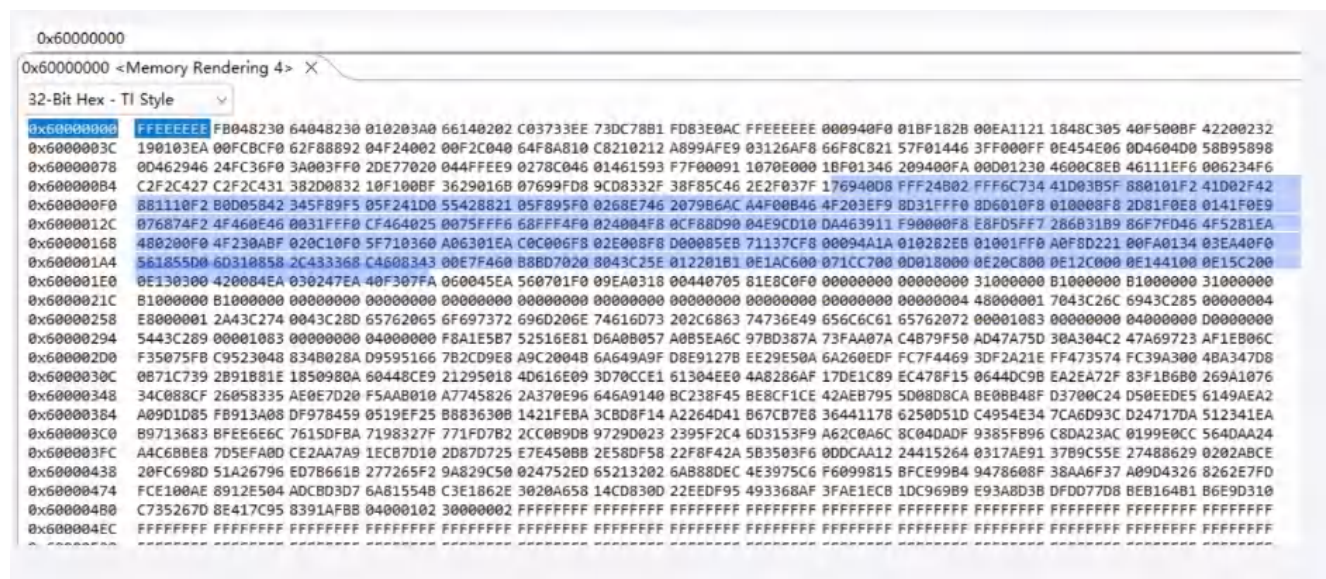


図 2-5. ブート ROM がロードしたメモリの内容

障害が発生したシステムを CCS に接続し、ブート ROM がロードしたメモリの内容を調べました。

想定された証明書識別子が所定のオフセットにありませんでした。メモリの先頭バイトが、元の **tiboot3.bin** イメージの対応するバイトと異なっていました。ブートに成功したケースと比較したところ、証明書ヘッダーの先頭バイトが 1 バイトずれていることが判明しました。

この結果から、証明書の検証が始まる前にイメージが破損していたことがわかりました。

#### 2.4.4 イメージ比較

以下のものの違いを比較しました。

- 元の **tiboot3.bin** イメージ
- ブート ROM がロードしたメモリの内容

比較の結果、イメージ全体で一貫して 1 バイトずれていることが判明しました。

証明書ヘッダーは固定オフセットに配置する必要があるため、このずれが認証失敗の直接の原因となりました。

#### 2.4.5 QSPI バス解析

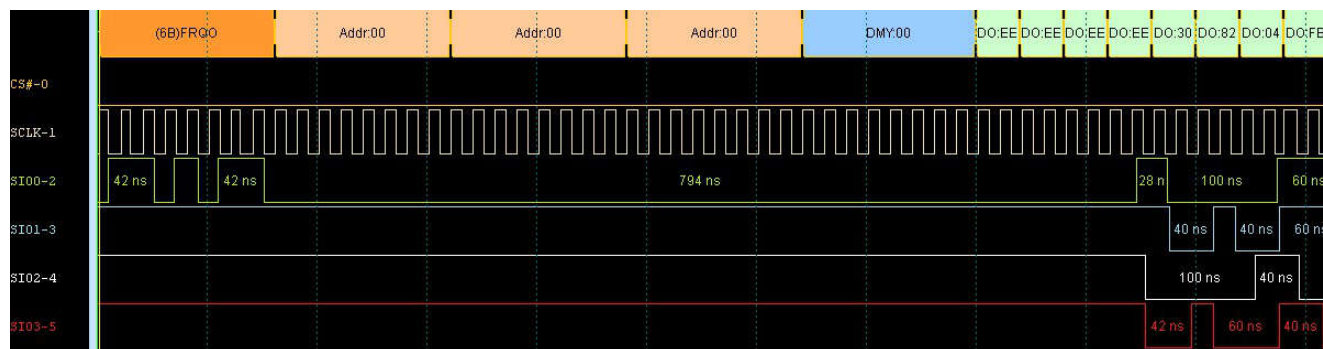


図 2-6. 起動時の QSPI バス キャプチャ

ロジックアナライザのキャプチャから、次のことが確認されました。

- ブート ROM は、3 バイト アドレス フェーズを使用してコマンド **0x6B** を送信しました。
- フラッシュ デバイスは 4 バイト アドレッシング モードのままでした。
- フラッシュのデータ出力は、想定より 1 バイト遅れて開始されました。
- 確認されたずれは、CCS によるメモリ検査で特定したオフセットと一致しました。

この相関関係から、認証失敗の原因が証明書生成エラーやイメージ署名エラーではなく、フラッシュ読み取り時のデータ破損であることを示す直接的な証拠が得られました。

#### 2.4.6 根本原因の確認

以下の結果を総合したもの：

- ブート ROM ログ解析
- CCS によるメモリ検査
- イメージ比較
- QSPI バス解析

起動障害の原因が、ブート ROM とフラッシュ デバイスのアドレッシング モードの不一致であることが確認されました。

ブート ROM は常に 3 バイト アドレス フェーズを使用してフラッシュにアクセスしますが、MCU\_PORz リセット後もフラッシュ デバイスは 4 バイト アドレッシング モードに構成されたままでした。この不一致によりイメージ ストリームに 1 バイトのずれが生じ、最終的に証明書認証が失敗しました。

### 2.5 デバッグ方法

ブート ROM ステージで起動障害が発生した場合は、次の方法を推奨します。

1. ステップ 1: ブート ROM ログの取得



- デバイスがバックアップ ブート モードに移行するかどうかを確認し、適切な TI ユーティリティを使用してブート ROM ログを収集します。
- 2. ステップ 2: 認証エラーの確認
  - 報告された認証失敗の原因が証明書の問題か、破損したイメージ データかを確認します。
- 3. ステップ 3: ブート ROM がロードしたメモリの検査
  - CCS を使用して、ブート ROM がデータを格納したメモリを検査します。
- 4. ステップ 4: 元の tiboot3.bin との比較
  - ブート ROM がロードしたイメージが、書き込まれたイメージと完全に一致することを確認します。
- 5. ステップ 5: QSPI バス タイミングの解析
  - ロジック アナライザを使用して QSPI トランザクションをキャプチャし、プロトコル タイミングを確認します。
- 6. ステップ 6: フラッシュのアドレッシング モード構成の確認
  - リセット後にフラッシュ デバイスが保持しているアドレッシング モードを確認します。

## 2.6 対策と設計上の推奨事項

### 2.6.1 推奨ソリューション: フラッシュ リセットの同期

推奨されるソリューションは、プロセッサがリセットされるたびにフラッシュ デバイスも確実にリセットする方法です。

これにより、ブート ROM の実行開始前にフラッシュが 3 バイト アドレッシング モードに戻り、起動時の堅牢性を最大限に高めることができます。

### 2.6.2 代替ソリューション: ドライブレベルの対策

調査対象のプラットフォームでは、コマンド 0x6C と 4 バイト アドレス フェーズを組み合わせるフラッシュ読み取り動作を標準化するソフトウェア対策を実装しました。

ブート ROM のコマンド 0x6B とは異なり、コマンド 0x6C は、ターゲット アドレスがフラッシュの下位アドレス空間内にある限り、フラッシュ デバイスが現在 3 バイトと 4 バイトのどちらのアドレッシング モードであっても正しく動作します。

この方法により、通常のソフトウェア動作時に現在のフラッシュ アドレッシング モードに依存しなくなります。この変更を実装した後、MCU\_PORz リセット テストを繰り返しても起動障害は発生しませんでした。

### 2.6.3 0x6C 対策の適用範囲

0x6C に基づく対策は、アプリケーションで 16MB のアドレス境界を越える消去またはプログラム動作が不要な場合に有効です。

代表的な例を次に示します。

- ブート イメージの保存
- ファームウェアの保存
- 読み取り専用の構成データの保存

このようなシステムでは、4 バイト アドレス フェーズを使用するコマンド 0x6C にフラッシュ アクセスを統一することで、アドレッシング モードへの依存を解消できます。

### 2.6.4 大容量フラッシュ アプリケーションの設計上の制約

16MB の境界を越えて消去またはプログラム動作を行うアプリケーションでは、上位メモリ領域にアクセスするために、引き続き 4 バイト アドレッシング モードに移行する必要があります。

このような動作の後に予期しないリセットが発生すると、ブート ROM が 3 バイト アドレッシング モードを想定し続ける一方で、フラッシュ デバイスは 4 バイト アドレッシング モードのままになる可能性があります。

独立した RESET# ピンがないフラッシュ デバイスでは、この状態によって、本アプリケーション ノートで説明した起動時の互換性問題が再発生する可能性があります。したがって、次のようになります。

- ハードウェア リセットの同期が引き続き推奨されます。
- デバイスを選定する際は、フラッシュのリセット動作を考慮する必要があります。
- 認定テストには、ウォーム リセットの検証を含める必要があります。

- アドレッシング モードの移行は、ソフトウェアで明示的に管理する必要があります。

### 3 まとめ

このドキュメントで調査した起動問題の原因は、AM62Ax ブート ROM とフラッシュ デバイスの アドレッシング モードの不一致でした。

外部から確認できた症状は証明書認証の失敗でしたが、ブート ROM ログ解析、CCS によるメモリ検査、イメージ比較、QSPI バス検証の結果、実際の根本原因は、4 バイト アドレッシング モードのままのフラッシュ デバイスにブート ROM がアクセスした際に生じた 1 バイトのずれであることが判明しました。

下位 16MB のアドレス空間内だけで動作するシステムでは、コマンド 0x6C に基づくドライバ レベルの対策が有効であることを確認しました。ただし、16MB の境界を越える消去またはプログラム動作が必要なアプリケーションでは、4 バイト アドレッシング モードへの移行と解除を慎重に管理する必要があります。

独立した RESET# ピンがないフラッシュ デバイスでは、フラッシュのリセットをプロセッサのリセットと同期させる方法が引き続き推奨されます。これにより、ブート ROM の実行前にフラッシュのアドレッシング モードを既知の状態にでき、起動時の堅牢性を最大限に高められます。

## 4 参考資料

1. テキサス インスツルメンツ、『[AM62Ax Sitara™ プロセッサ](#)』、データシート。
2. テキサス インスツルメンツ、『[AM62A Sitara™ プロセッサ テクニカル リファレンス マニュアル](#)』、テクニカル リファレンス マニュアル。
3. テキサス インスツルメンツ、『[AM62AX 用プロセッサ SDK Linux ソフトウェア開発者ガイド](#)』。
4. テキサス インスツルメンツ、『[AM62AX 11.01.00.16 向け AM62Ax MCU+ SDK](#)』。
5. Infineon、『[256Mb/512Mb/1Gb SEMPER™ フラッシュ](#)』データシート。
6. Dosilicon、『[DS25M4CB QSPI フラッシュ データシート](#)』。
7. テキサス インスツルメンツ、『[AM62A3:POR リセット後の AM62A3 の異常ブートアップ](#)』、E2E™ テクニカル サポート フォーラム。

## 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みいかなる保証も、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、TI は一切の責任を拒否します。

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](https://www.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。

Copyright © 2026, Texas Instruments Incorporated

最終更新日：2025 年 10 月