

Application Note

MSPM0 および MSPM33 デバイス: バンクスワップによる IO 状態の保持

Erik Vaughn

概要

マイコンは通常、GPIO を使用して、PCB 上の他の IC へ複数の信号を駆動します。マイコンのライフサイクルを維持するには、これらの GPIO の電圧レベルが重要です。ほとんどのマイコンでは、ファームウェア更新を実行するためにリセットが必要です。これは、ファームウェアの更新と、特定の GPIO 電圧の維持の両方を必要とするアプリケーションで問題となります。ファームウェアを更新して GPIO 電圧を維持するために、MSPM0 および MSPM33 デバイスには、ブートプロセスに必要なリセットを実行する前に IO 状態をラッチする方法が提供されています。これらのラッチを設定するには、デバイスを低消費電力モード (シャットダウン) に移行する必要がありますが、これにより、デバイスをウェークアップするときのさらなる問題が発生します。このドキュメントでは、これらのラッチをトリガする方法、シャットダウン後にマイコンを自動的にウェークアップする方法、この機能を実現するソフトウェア アプリケーションを構築する方法、ブートプロセスの使用事例について説明します。

目次

1 概要	2
1.1 MSPM0 マイコンおよび MSPM33 マイコンのファームウェア更新.....	3
1.2 MSPM0 と MSPM33 の電力モード.....	4
2 IO ラッチの仕組み	5
2.1 IO 状態がラッチされる方法.....	5
2.2 ブートプロセス中の IO 状態の維持.....	6
3 シャットダウン IO 機能に関するソフトウェアの設定	7
3.1 シャットダウンからウェークアップするソフトウェアを構成する.....	7
3.2 ブートローダーを併用した状態保持 IO の活用.....	8
3.3 シャットダウン メモリ.....	9
4 まとめ	9
5 参考資料	9

商標

すべての商標は、それぞれの所有者に帰属します。

1 概要

マイコンが LDO をイネーブルにしているアプリケーションや、アナログ信号のバイアス印加、または個別の IC のオンを行うアプリケーションなど、マイコンのライフサイクル中に GPIO の状態を維持する必要があるアプリケーションでは、マイコンへの電力供給が重要である間にこれらの GPIO の電圧状態を維持できます。通常、GPIO 出力電圧と IOMUX 構成に関するこの情報は、マイコンのメモリマップレジスタ (MMR) 内に維持されます。マイコンがリセットすると、これらの MMR がリセットされ、これらの外部電圧信号が失われます。マイコンが GPIO をアクティブに駆動している間にファームウェア更新を実行できる可能性がある使用事例では、マイコンは、システムリセットを発行する必要があります。このシステムリセットにより、マイコンは MMR 情報を失い、GPIO 電圧が消失します。これにより問題が発生しますが、回路にハードウェアを追加することで解決できます。ただし、その場合はシステム全体のコストが増加してしまいます。この追加コストを防止するため、MSPM0 および MSPM33 の各製品には、IOCell 構造内にラッチが搭載されており、更新することにより IO 状態を維持できます。

IOMUX 内のこれらのラッチは、MSPM0 および MSPM33 の各デバイスのすべてのピンで利用できます。IOMUX 内のこれらのラッチは、内部プルアップ / プルダウン抵抗と GPIO 出力電圧を維持します。これらのラッチをトリガするには、マイコンはシャットダウン状態に移行してウェイクアップする必要があります。このプロセスにより、マイコンも強制的にプライマリまたはセカンダリ ブートローダーを実行します。アプリケーション イメージを更新する前にこのステップを追加することにより、GPIO 電圧を必要な期間適切に維持できます。

Old Method: GPIO state lost

New method: GPIO State's maintained

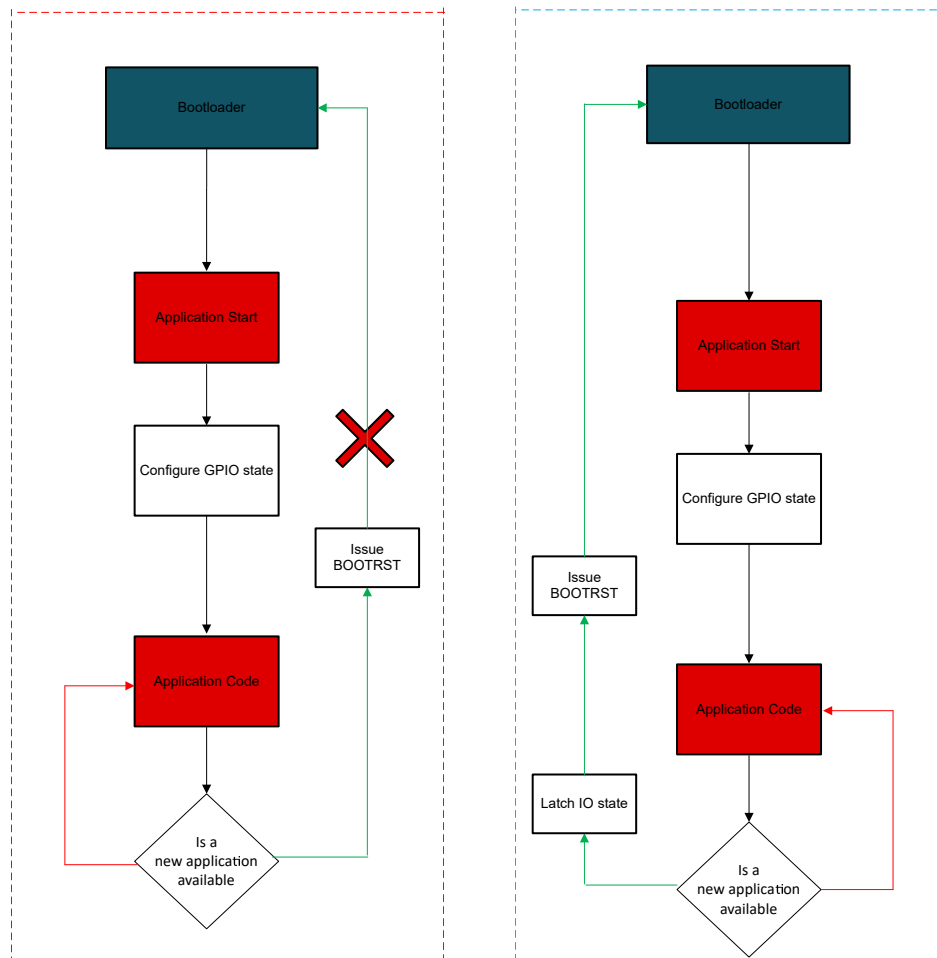


図 1-1. 標準ブートプロセスと維持される GPIO 状態のブートプロセスの比較

MSPM0 および **MSPM33** の各種デバイスには、ファームウェア更新プロセスをサポートするさまざまな方法があります。通常、これは**プライマリ (ROM) ブートローダー**または**BSL** (とも呼ばれる) または**セカンダリ ブートローダー** (フラッシュ ブートローダーとも呼ばれる) を使用して実現されます。これらのブートローダーは、いくつかの通信ペリフェラルを利用して外部デバイスから更新を読み取り、コード メモリに新しいアプリケーション イメージを書き込みます。これらのブートローダーの中には、**カスタム セキュア コード (CSC)** などのセキュリティ機能を提供するものがあります。**CSC** は、新しいアプリケーション イメージが適切な署名を持っているかを検証し、新しいファームウェアがセキュア素子から送信されたものであることを保証します。これまでに説明したすべてのブートローダーは、同様のフローに従います。このフローでは、**ROM ブートローダー**または**セカンダリ ブートローダー**のどちらかを使用してブートローダーを正しく実行するために、マイコンがリセットを実行する必要があります。

MSPM0 および MSPM33 には、[図 1-2](#) に示すような各種のリセット オプションがあります。赤いボックスは、マイコンによってトリガされる可能性があるさまざまな種類のリセットを示しています。BCR と BSL も [図 1-2](#) グラフィックで強調表示されています。BSL はブートストラップ ロダー、BCR はブート構成ルーチンです。これらが組み合わさることで ROM ブートローダーを構成します。この ROM ブートローダーには、新しいファームウェア更新をトリガする機能があります。その場合、BCR ボックスと BSL ボックスの上にリセットが発生したときに、ブラウンアウトリセット (BOR)、パワーオンリセット (POR)、ブートリセット (BOOTRST) など、必要なファームウェアの更新がトリガされます。

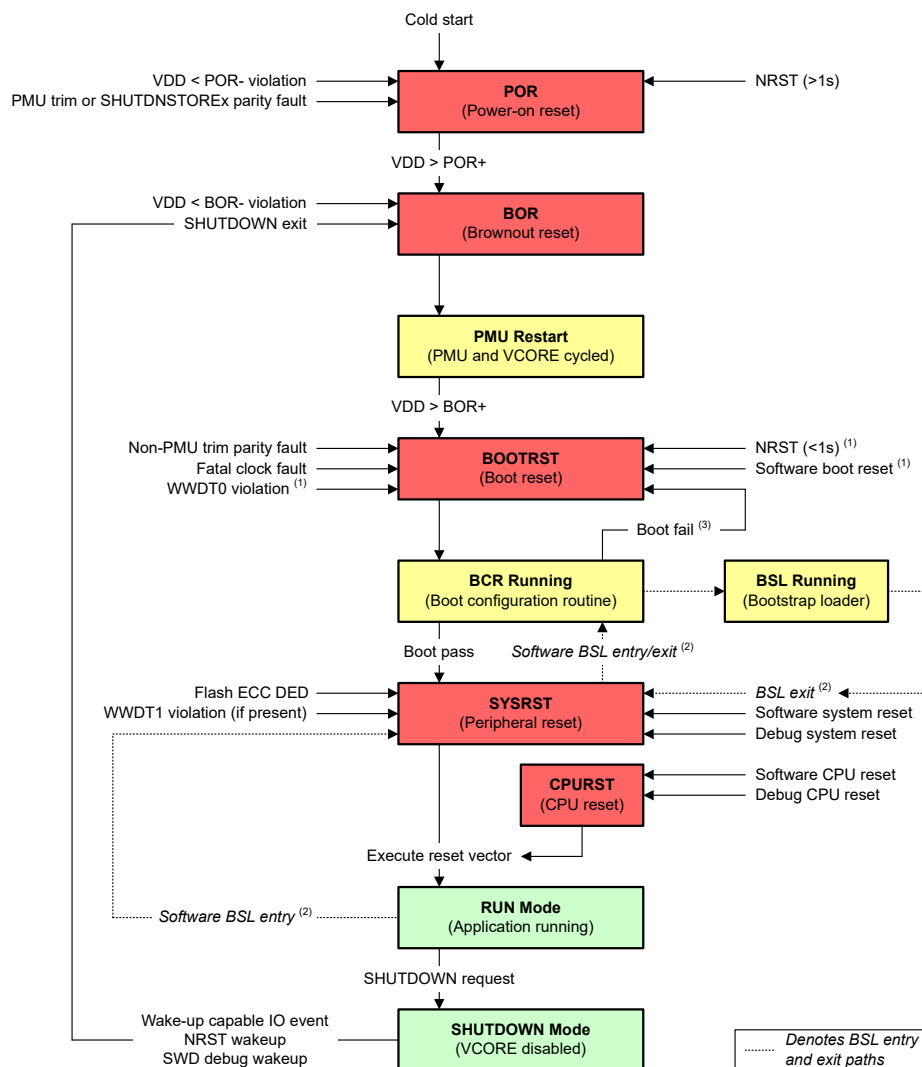


図 1-2. デバイスのリセット フロー

1.2 MSPM0 と MSPM33 の電力モード

MSPM0 デバイスおよび MSPM33 デバイスには、RUN、SLEEP、STOP、STANDBY、SHUTDOWN の 5 つの異なる電力モードがあります。これらの GPIO 状態をトリガするには、デバイスは低消費電力モード (シャットダウン) 状態に移行する必要があります。他のモードと比較した場合のシャットダウン電力モードの特徴は、すべてのペリフェラルとマイコンがディープスリープモードに移行し、構成されたペリフェラルのみがマイコンをウェークアップできる点にあります。さまざまな動作モードの概略リファレンスとして、[図 1-3](#) を参照してください。

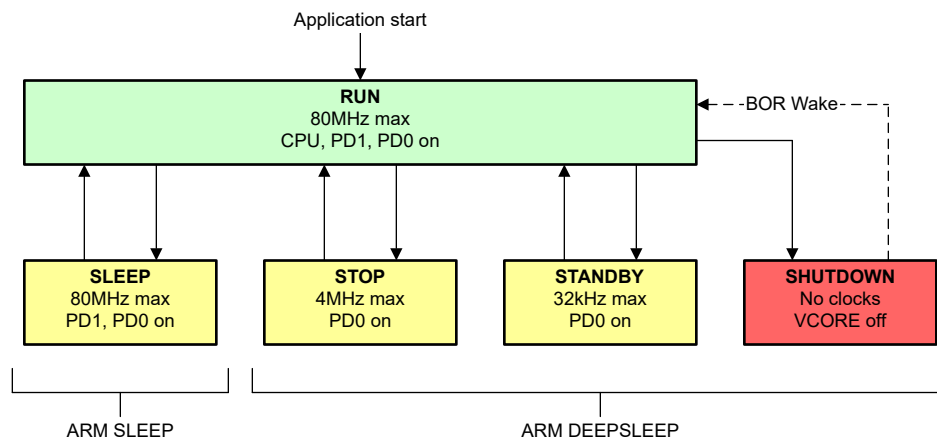


図 1-3. MSPM0G の動作モード

[図 1-3](#) に示すように、デバイスのシャットダウンモードを終了すると、BOR がトリガされます。これにより、マイコンはリセットをトリガし、アプリケーションコードを再度開始します。これは、コードの開始時にトリガする必要があるファームウェア更新を実行することが目標である場合に便利です。ただし、BOR がトリガされる前に、デバイスはシャットダウンからウェークアップする必要があります。マイコンをシャットダウンからウェークアップするには、ウェークアップ対応 IO、デバッグ接続、または NRST のいずれかを使用します。このプロセスを自動化するには、NRST を発行したり、デバッグ接続を作成したりするよりも、ウェークアップ対応 IO が最も簡単な方法です。このドキュメントでは、これについて [セクション 2](#) で詳しく説明します。

2 IO ラッチの仕組み

セキュア ブート プロセスでラッチを使用する方法を適切に理解するには、最初にこれらのラッチがトリガされる方法のプロセス、IO 状態が維持される状態、ウェークアップ手順を自動化する方法について理解する必要があります。このセクションでは、GPIO 電圧レベルをラッチできるハードウェア、シャットダウンに移行してこれらのラッチをトリガする方法、シャットダウンに移行した後にデバイスを自動的にウェークアップする方法について説明します。

2.1 IO 状態がラッチされる方法

IOMUX ラッチのトリガ

IO マルチプレクサの内部には、GPIO のロジックレベルを維持する特定のラッチがあります。下の図の **SHUTDOWN** ラッチを参照してください。マイコンがシャットダウンに移行すると、これらのラッチは自動的に設定されます。ラッチが設定された後、SHDINIOREL レジスタの **RELEASE** ビットが設定されるまで、GPIO の状態は維持されます。IO 状態のラッチとラッチ解除に必要な手順を以下に示します。

1. SHUTDOWN 信号がアサートされると、出力パスの D-Q ラッチが DOUT、DRV (駆動強度)、PIPU (プルアップ イネーブル)、および PIPD (プルダウン イネーブル) をキャプチャします。
2. ラッチは、GPIO MMR とは無関係に状態を維持します
3. 解放信号 (SHDINIOREL を書き込むことにより駆動) は、ラッチをリセットし、IO 制御を通常の MMR 駆動パスに戻します。

この状態は、BOOTRST、BCR、BSL、SYSRST、CPURST によって維持されます。ソフトウェアが SHDINIOREL でラッチを解放するまで、**ブートプロセス中の IO 状態の維持** に示す階層ではすべて低いリセットが行われます。詳細については、テクニカル リファレンス マニュアルの「PMCU」の章を参照してください。

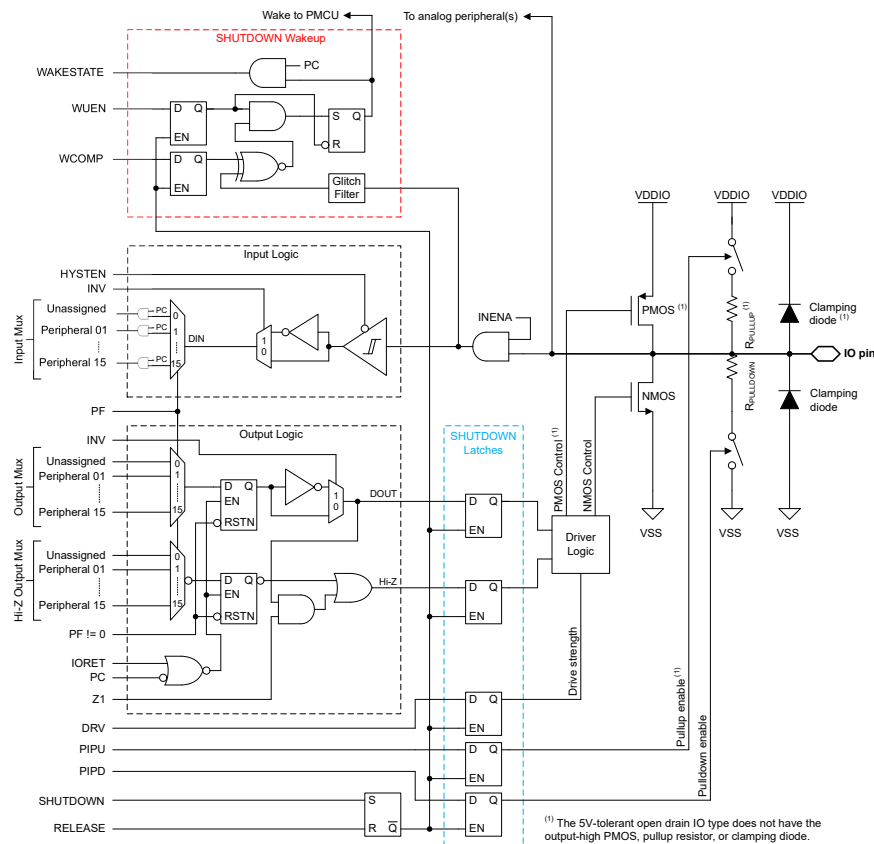


図 2-1. IOMUX 構造

シャットダウンからのウェークアップ

IO 状態を適切にラッチするには、マイコンはシャットダウンに移行するだけです。問題なくマイコンが直ちにウェークアップできることを確認するために、内蔵プルアップ抵抗を使用して、ウェークアップ機能を備えた個別の IO を構成します。このウェーク機能のロジックレベルが High の場合、デバイスがシャットダウンに移行した後、マイコンは直ちにウェークアップします。

MSPM0 または MSPM33 デバイスにシャットダウン対応 IO があることを確認するには、デバイスのデータシートを参照してください。以下の表に、MSPM0 または MSPM33 デバイスでサポートされる汎用 IO タイプと、各 IO タイプの機能を示します。GPIO がウェークアップ機能付き SDIO、ODIO、または HDIO である場合、シャットダウンからのウェークアップをサポートしています。

表 2-1. IO タイプ別のデジタル IO 機能

バッファのタイプ	反転制御	駆動能力制御	ヒステリシス制御	プルアップ抵抗	プルダウン抵抗	ウェークアップ ロジック
SDIO (標準駆動)	Y			Y	Y	
WAKE ⁽¹⁾ 付き SDIO (標準駆動)	Y			Y	Y	Y
HDIO (高駆動)	Y	Y		Y	Y	Y
HSIO (高速)	Y	Y		Y	Y	
ODIO (5V 対応のオープンドレイン)	Y		Y		Y	Y

2.2 ブートプロセス中の IO 状態の維持

ブートリセット後は GPIO レジスタの情報が失われることにより IO 状態も喪失するため、アプリケーションがそれらを適切に再設定できるようになるまでは、前述の IOMUX 用内部ラッチを使用して IO 状態を保持できます。これらのラッチをトリガするには、マイコンがシャットダウンに移行する必要があります。マイコンがシャットダウンからウェークアップした後、デバイスは BCR、BSL、または構成されているセカンダリ ブートローダーを実行します。その後、メイン アプリケーションが IO 状態を適切に再構成し、IOMUX ラッチを解放できるまで、マイコンはこのプロセスによって IO 状態を適切に維持します。

IO 状態をラッチするようアプリケーションを構成するには、以下の手順に従います。

1. GPIO を希望の状態に構成します。
2. シャットダウンからのウェークアップ機能を備えた単一の GPIO を、内部プルアップ抵抗を有効にし、High 検知でウェークアップするように設定します。
3. デバイスをシャットダウン状態にします。
4. ウェークアップ後、SHDINIOREL に書き込んで IO ラッチを解放してから、MMR を再構成します。

上記の手順を使用すると、デバイスはシャットダウンから直ちにウェークアップし、ブートシーケンスが再起動されます。シャットダウンからウェークアップすると、BOR がトリガされます。これにより、デバイスは強制的にブートリセットプロセスを実行します。BOR の後に起きることについては、以下の図を参照してください。

3 シャットダウン IO 機能に関するソフトウェアの設定

IO ラッチの仕組みが理解できたので、この機能を活用して、GPIO 状態を保持したままデバイスをシャットダウン状態へ移行させるための一般的なソフトウェア設計の枠組みを構築できます。このセクションでは、IOMUX ラッチを中心にアプリケーションを構築し、GPIO 状態を維持しながらマイコンがブートローダーを適切に通過できるようにする方法について説明します。

3.1 シャットダウンからウェークアップするソフトウェアを構成する

IO 状態を維持しながら、SYSRST または他のタイプのリセットを使用してマイコンをリセットすることが目標である場合、ユーザーはウェーク機能を備えた GPIO に内部プルアップ抵抗を構成して、High のときに GPIO をウェークアップするように設定できます。この手順を実行すると、デバイスをシャットダウン状態に移行できます。マイコンは、他の入力が必要とせずにウェークアップします。

IO 状態をラッチするようマイコンを構成するには、次の 2 つの主なステップがあります。

1. シャットダウン状態からウェークアップする機能がある IO を、内部プルアップ抵抗を使用して構成し、高電圧が受信されたときにウェークアップします。
2. マイコンをシャットダウン状態にします

最初の手順は、以下のコード サンプルを使用してウェーク IO を構成することです。

```
DL_GPIO_initDigitalInputFeatures(Wake_IO_PIN_0_IOMUX,  
    DL_GPIO_INVERSION_DISABLE, DL_GPIO_RESISTOR_PULL_UP,  
    DL_GPIO_HYSTERESIS_DISABLE, DL_GPIO_WAKEUP_ON_1);
```

2 つ目のポイントとして、以下の API を使用できます

```
DL_SYSCCTL_setPowerPolicySHUTDOWN();  
__WFI();
```

上述のラインにより、本デバイスがシャットダウン モードになります。ウェークアップ機能付き GPIO が High 検知でウェークアップする設定になっており、さらにプルアップ抵抗が接続されているため、マイコンはシャットダウン移行直後に即座にウェークアップしてしまいます。

IO がラッチされた後、GPIO 機能を適切に更新するには、ユーザーは IO 状態を解放する必要があります。これを行うには、以下の API を使用して、SHDINIOREL レジスタの RELEASE ビットを直接設定できます。

```
/* Release shutdown IO latches - call this after MMRs have been re-initialized */  
DL_SYSCCTL_releaseShutdownIO();
```

3.2 ブートローダーを併用した状態保持 IO の活用

この機能は、ブートプロセス中やバンクスワップ中に IO 状態が失われる可能性がある、マイコンのブートローダーに役立ちます。特定の IO が信号のバイアスに使用される場合や、両方のアプリケーションで既知のデジタル状態にロックされる場合は、前に示したソフトウェアを使用して IO 状態を設定できます。ブートローダーを使用するアプリケーションを開発する際は、GPIO の状態をラッチするプログラムを構成する方法の一般的な考え方として 図 3-1 を使用できます。この例では、アプリケーション A が利用可能な新しいファームウェア イメージを検出し、バンクスワップを実行する必要があります。シャットダウンに入る前に、アプリケーションは出力 GPIO を希望するホールド状態に構成します。シャットダウンからウェークアップすると (BOR をトリガすると)、実行するアプリケーションがブートローダーによって決定されます。各アプリケーションは、メイン アプリケーション ループを実行する前に、GPIO 状態を再構成し、SHDINIOREL 経由でシャットダウン IO ラッチを解放することにより開始します。

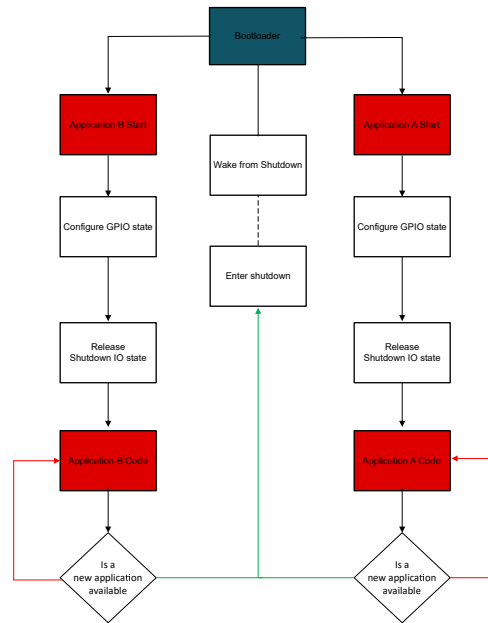


図 3-1. IO ラッチ機能を利用するためのブートフロー

マイコンがシャットダウンからウェークアップしたかどうかの判断

アプリケーション イメージが異なる電力モードを使用しており、マイコンがそのアプリケーション イメージを再起動する可能性がある複数の異なる方法が存在する場合。マイコンがシャットダウンからウェークアップして SHDINIOREL レジスタの RELEASE ビットを設定する必要があるのかを判断するには、RSTCAUSE レジスタを読み取ります。例として以下のコードを使用します。

```

if (DL_SYSCTL_getResetCause() == DL_SYSCTL_RESET_CAUSE_SHUTDOWN) {
    /* Device woke from shutdown - IO latches are active */
    /* Re-initialize peripherals, then release latches */
    DL_SYSCTL_releaseShutdownIO();
}

```

マイコンが、アプリケーション イメージが開始する前に SYSRST を発行した CSC または他のブートローダーを使用する場合、RSTCAUSE レジスタを対応するソフトウェアトリガの SYSRST ID 値と比較できます。詳細については、テクニカル リファレンス マニュアルの「PMCU」の章を参照してください。

3.3 シャットダウン メモリ

さらに、このマイコンを使用して、SYSCTL ペリフェラル内に最大 4 バイトのデータを保存できます。このデータは、SHUTDNSTORE0 - SHUTDNSTORE3 レジスタ内に保存できます。シャットダウンに入る前にデータが保存されている限り、SHUDNSTORE レジスタは、GPIO の状態が維持される方法と同様に維持されます。

シャットダウン メモリ ソフトウェア

このシャットダウン ストレージ バイトを保存するには、次の API を使用します。

```
/* Before shutdown: store application state */
DL_SYSCTL_setShutdownStorageByte(DL_SYSCTL_SHUTDOWN_STORAGE_BYTE_0, myStateValue);
```

シャットダウンからウェークアップした後、以下の API を使用して、対応するレジスタを直接読み取れます

```
/* After wakeup: retrieve stored state */
uint8_t myStateValue = DL_SYSCTL_getShutdownStorageByte(DL_SYSCTL_SHUTDOWN_STORAGE_BYTE_0);
```

シャットダウン メモリの使用事例

バンクスワップやブートローダーを使った使用事例では、バンクスワップの間に MMR と SRAM が消失しました。そのような使用事例では、ユーザーは代わりにこのメモリに直接書き込めます。これは、前のアプリケーションの実行時にしか計算されないデータを、デバイスが実行中に計算した場合に役立ちます。この例の 1 つは、I2C バス上のデバイスが通信したときにインクリメントされるカウンタです。

4 まとめ

マイコンのライフサイクル全体にわたって GPIO 電圧を維持する必要があるアプリケーションでは、MSPM0 および MSPM33 の各種マイコンは、IO セルのシャットダウン IO 機能を利用して電圧を適切にラッチできます。デバイスを低消費電力モード (シャットダウン) にすると、ラッチを設定できます。マイコンは GPIO をプルアップ抵抗として事前構成して、直ちにマイコンをウェークアップできます。このことを念頭に置いてソフトウェアを構成すると、前述のコードを利用して、デバイスをシャットダウンし、ブートプロセスをトリガして、IO 状態をラッチするソフトウェアを構築できます。この機能を使用することにより、ファームウェアのアップデート実行時に IO 電圧が消失しないよう、ソフトウェア プロジェクトを容易に設定できます。

5 参考資料

- テキサス インストルメンツ、『MSPM33 C3 シリーズ 160MHz マイコン技術参照マニュアル』、技術参照マニュアル。

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](#) 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