

Application Note

MSP430™ MCU から MSPM0 MCU への移行ガイド



Diego Abad Sajamin

概要

MSP430™ デバイスは、テキサス・インスツルメンツの従来型マイコンとして、30 年近く使用されてきました。多くのお客様がさまざまなプロジェクトで MSP430 MCU を使用しており、現在も引き続き使用しています。新世代の MSPM0 製品は、より豊富な機能を備えた周辺リソースとより小型のパッケージを備えた Arm® Cortex®-M0+ コアを採用しています。新製品の開発や旧製品のアップグレードにより、お客様は新しい MSP デバイスを使用できるようになります。このアプリケーションノートでは、MSP430 MCU から MSPM0 MCU へのソフトウェアの移行について説明します。

目次

1 ソフトウェア ポーティング フロー.....	2
2 開発環境.....	3
2.1 統合開発環境 (IDE).....	3
2.2 ソフトウェア エコシステム.....	3
2.3 MSPM0 MCU 用の SysConfig.....	6
2.4 MSP430 および MSPM0 プロジェクト.....	7
2.5 デバッガ インターフェイス.....	8
3 移行にあたっての考慮事項.....	13
3.1 周辺機器.....	13
3.2 システム クロック.....	13
3.3 動作モード.....	14
3.4 不揮発性メモリ (NVM).....	15
3.5 イベントと割り込みハンドラ.....	16
3.6 リセットレベル.....	17
3.7 GPIO とピンのマルチプレクシング.....	18
3.8 通信インターフェイス.....	19
3.9 BSL.....	20
3.10 アナログ ペリフェラル.....	22
3.11 タイマ.....	25
3.12 ハードウェア設計ガイド.....	25
3.13 改訂履歴.....	26
4 改訂履歴.....	27

商標

MSP430™, Code Composer Studio™, and LaunchPad™ are trademarks of Texas Instruments.

Arm® and Cortex® are registered trademarks of Arm Limited.

すべての商標は、それぞれの所有者に帰属します。

1 ソフトウェア ポーティング フロー

図 1-1 に、MSP430 から MSPM0 MCU へのソフトウェア移行の手順を示します。以下のセクションでは、開発環境、ソフトウェア、周辺装置の相違点について説明します。

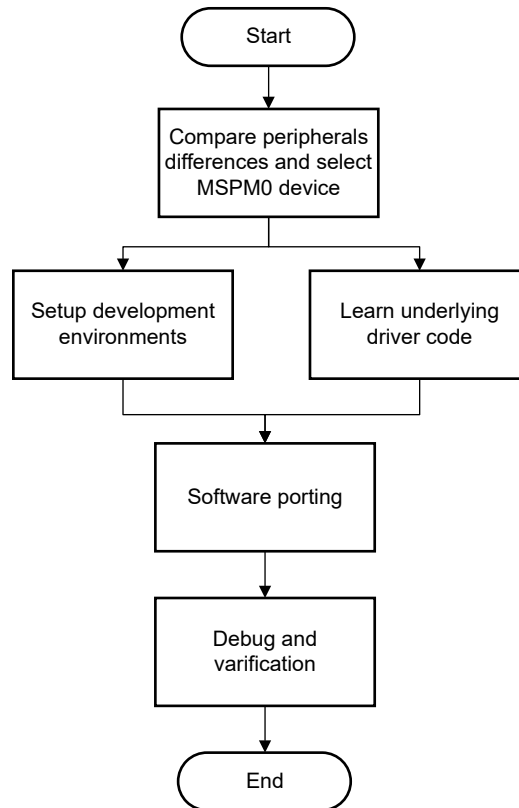


図 1-1. ソフトウェア移行ワークフロー

2 開発環境

2.1 統合開発環境 (IDE)

Code Composer Studio™ 統合開発環境 (IDE) (CCS) は、TI のマイコンと組み込みプロセッサ製品ラインアップに属するすべてのデバイスをサポートします。CCS に加えて、MSP430 および MSPM0 デバイスは、他の IDE でサポートされています (表 2-1 を参照)。

表 2-1. IDE サポート

IDE	MSP430	MSPM0
CCS	あり	あり
IAR	あり	あり
Keil	なし	あり

- CCS: <https://www.ti.com/tool/CCSTUDIO>
- IAR: <https://www.iar.com/>
- Keil: <https://www.keil.com/>

2.2 ソフトウェア エコシステム

表 2-2. ソフトウェア エコシステムの比較

機能	MSP430Ware	MSPM0 SDK
レジスタレベルのコード	あり	なし
ドライバライブラリ	あり	あり
ミドルウェア	あり	あり
すぐに使用できるコード	あり	あり
無償の RTOS	なし	あり

2.2.1 MSP430 ソフトウェア サポート パッケージ: MSP430Ware

MSP430Ware は、MSP430 コードの効果的な作成 構築に役立つリソース集です。これらのリソースは、すべての MSP430 マイコン (MCU) をサポートしています。この完全な設計リソース集には、すべての周辺機器のサンプルコードに加えて、高度に抽象化されたソフトウェア ライブラリの幅広い選択肢が含まれています。特に MSP430 ドライバライブラリは重要なライブラリであり、これによりソフトウェア開発者は便利な API を活用して下位の複雑なハードウェア パリフェラルを制御でき、最終的なコードは読みやすく、保守しやすいものになります。また、MSP430Ware にはクラウド バージョンもあります。

2.2.1.1 レジスタレベルのサンプルコード

16 ビット MSP430 MCU は、最大 25MHz のコア速度をサポートしています。レジスタレベルのサンプルコードは、レジスタを直接操作して、プロジェクトを高速化し、サイズを縮小します (例については 図 2-1 を参照してください)。MSP430 MCU のコード例は、デバイスのすべての周辺機能をカバーしています。

```

*****
*
*           MSP430 CODE EXAMPLE DISCLAIMER
*
* MSP430 code examples are self-contained low-level programs that typically
* demonstrate a single peripheral function or device feature in a highly
* concise manner. For this the code may rely on the device's power-on default
* register values and settings such as the clock configuration and care must
* be taken when combining code from several examples to avoid potential side
* effects. Also see www.ti.com/grace for a GUI- and www.ti.com/msp430ware
* for an API functional library-approach to peripheral configuration.
*
* --/COPYRIGHT--*/
//*****
// MSP430F665x Demo - Software Toggle P1.0
//
// Description: Toggle P1.0 by xor'ing P1.0 inside of a software loop.
// ACLK = 32.768kHz, MCLK = SMCLK = default DCO~1MHz
//
//           MSP430F665x
//           -----
//           /|\
//           | |
//           --| RST
//           | |
//           | |
//           | |
//           P1.0 |-->LED
//
// P. Thanigai
// Texas Instruments Inc.
// May 2012
// Built with IAR Embedded Workbench Version: 5.40 & CCS V5.2
//*****

#include <msp430.h>

int main(void)
{
    volatile unsigned int i;

    WDTCTL = WDTPW | WDTHOLD;           // Stop WDT
    P1DIR |= BIT0;                       // P1.0 set as output

    while(1)                             // continuous loop
    {
        P1OUT ^= BIT0;                   // XOR P1.0
        for(i=20000;i>0;i--);           // Delay
    }
}

```

図 2-1. MSP430 レジスタレベルのサンプルコード

2.2.1.2 ドライバ ライブラリ

この周辺装置ドライバ ライブラリを使用すると、デバイス レジスタ レベルではなく API レベルでアプリケーションを開発できます。API を使用すると、開発者は使用中の特定の MSP430 デバイスの微妙な差異ではなく、アプリケーションに集中できます。詳細については、CCS または [TI Resource Explorer](#) の *MSP430Ware 周辺装置ドライバライブラリ API* および *MSP430Ware 周辺装置ドライバライブラリ ユーザーズ ガイド* を参照してください。

2.2.1.3 ミドルウェア

特定のアプリケーション シナリオでは、ミドルウェアのサンプル プロジェクトでレファレンス コードが提供され、ユーザーがより便利に設計できるようになります。たとえば、IQmath ライブラリはより高速で複雑な計算の開発に役立ち、USB 開発者パッケージは USB アプリケーションの開発に役立ちます。詳細については、[TI Resource Explorer](#) で各デバイスのミドルウェアを参照してください。図 2-2 は、一般的なミドルウェア事例の置き場所を示しています。

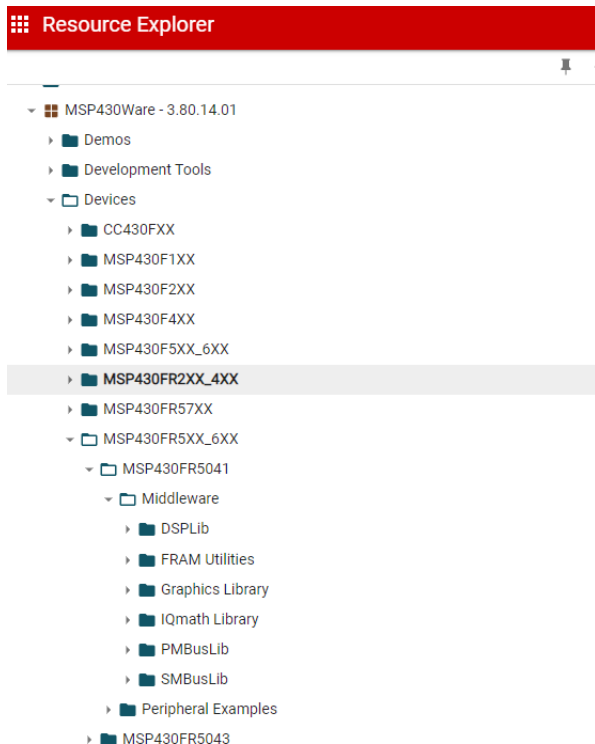


図 2-2. MSP430 ミドルウェア

2.2.2 MSPM0 ソフトウェア サポート パッケージ: MSPM0SDK

MSPM0 SDK は、Texas Instruments MSPM0+ マイクロコントローラ上でアプリケーションを開発するのに役立つコンポーネントを提供します。MSPM0 SDK は、複数のソフトウェア コンポーネントと、これらのコンポーネントを一緒に使用方法の例で構成されています。さらに、各機能分野とサポートされている各デバイスを独自のプロジェクトの開始点として使用法を解説しています。図 2-3 は典型的な MSPM0 SDK の例を示します。

サンプル フォルダは RTOS と非 RTOS のサブフォルダに分割されています (現在、非 RTOS のみがサポートされています)。これらのフォルダには各 LaunchPad™ 開発キットのサンプルが含まれており、下位レベルの Driverlib サンプル、上位レベルの TI ドライバ サンプル、GUI Composer、LIN、IQMath などのミドルウェアのサンプルの機能に基づいて編成されています。詳細については、MSPM0 SDK ユーザー ガイドを参照してください。

一部のサンプルは SysConfig をサポートしており、デバイスの構成を簡素化し、ソフトウェアを迅速に開発できるようにします。詳細については、それぞれの例を参照してください。

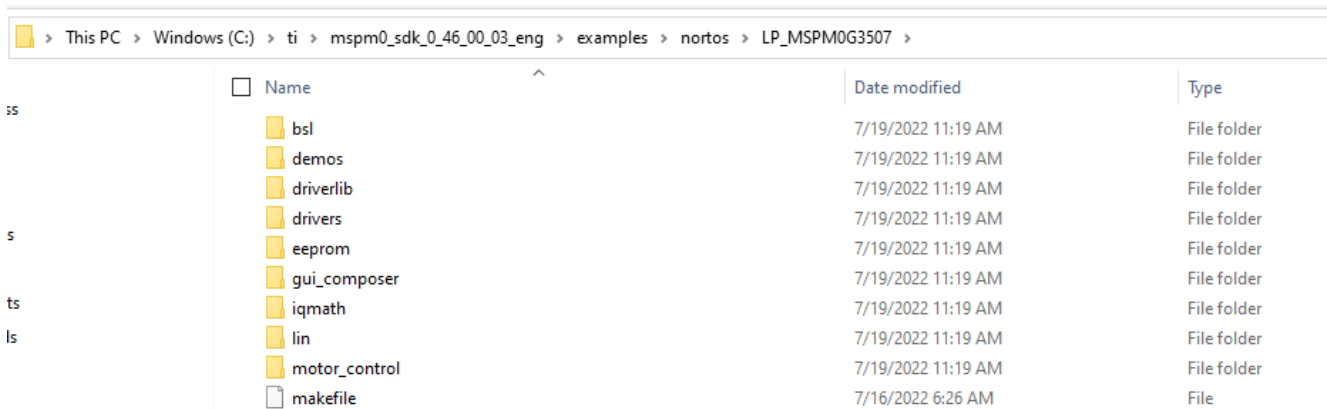


図 2-3. MSPM0 SDK

2.3 MSPM0 MCU 用の SysConfig

SysConfig は、ピン、ペリフェラル、無線、サブシステム、他の機能を構成するために使用できる、直観的で包括的なグラフィカル ユーティリティのコレクションです。SysConfig を使用すると、コンフリクトの管理、表面化、解決をビジュアルな方法で実行できるので、より多くの時間をアプリケーションの差異化に割り当てることができます。このツールの出力には C ヘッダとコード ファイルが含まれており、MSPM0 SDK サンプルと組み合わせて使用することも、カスタムソフトウェアの構成に使用することもできます。詳細については、*MSPM0 SysConfig ガイド*を参照してください。

2.3.1 スタンドアロン SysConfig

スタンドアロン SysConfig (図 2-4 を参照) を PC にダウンロードしてインストールすると、プロジェクトを構成するためのヘッダ ファイルとコード ファイルを生成できます。

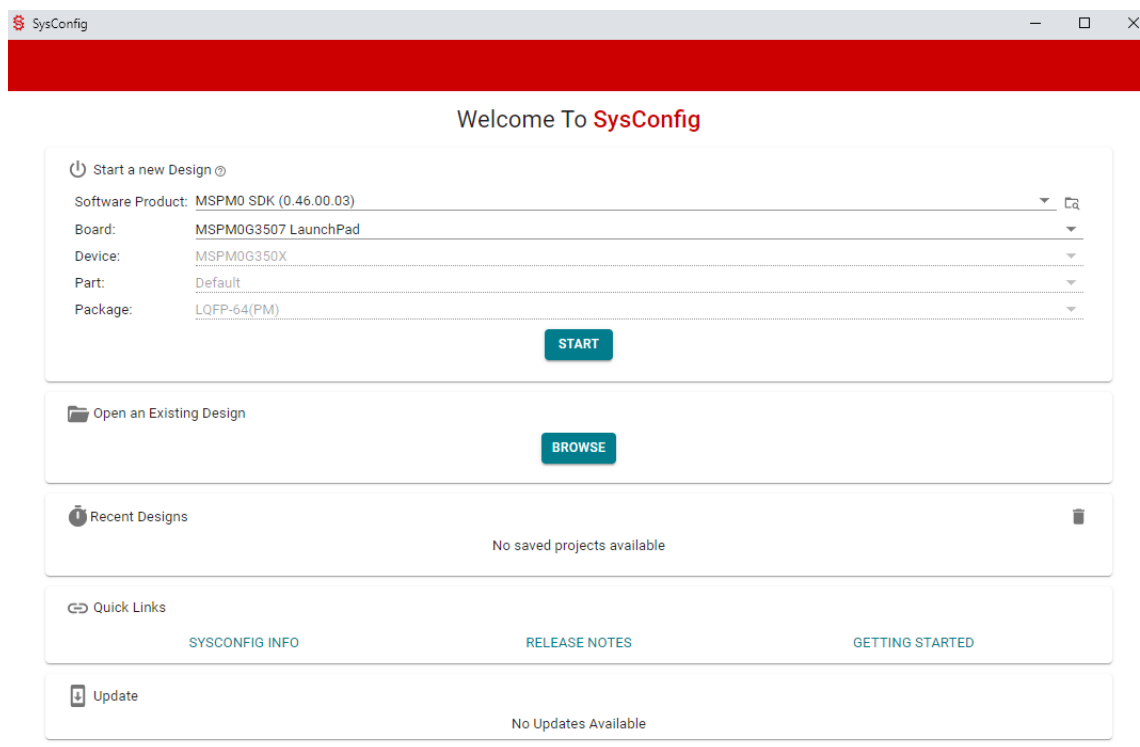


図 2-4. スタンドアロン SysConfig

2.3.2 CCS 統合 SysConfig

MSPM0 SDK の多くの例には、拡張子が `.syscfg` の SysConfig ファイルが含まれています。この SysConfig プロジェクトファイルは、クロック構成、低電力モード ルール、ADC 構成、I2C、SPI、UART 構成、その他の周辺装置構成などのシステムの初期化を保存します。これらのプロジェクト ファイルは CCS で変更でき、プロジェクトとともに保存できます。ユーザーが別のプロジェクトで同じ構成を使用したい場合は、その構成を新しいプロジェクトに簡単にインポートできます。

2.3.3 SysConfig プロジェクトの例

図 2-5 に示すトグル出力のサンプル プロジェクトでは、構成ファイルがプロジェクトに含まれています。プロジェクトがビルドされると、CCS は構成ファイルに基づいて構成ヘッダとコード ファイルを生成します。このプロジェクトでは、GPIO とシステム クロックを構成します。さらに開発を行うには、CCS で `sysconfig` ファイルを開き、必要に応じて周辺装置を変更します。プロジェクトがビルドされると、ヘッダ ファイルとコード ファイルが更新されます。

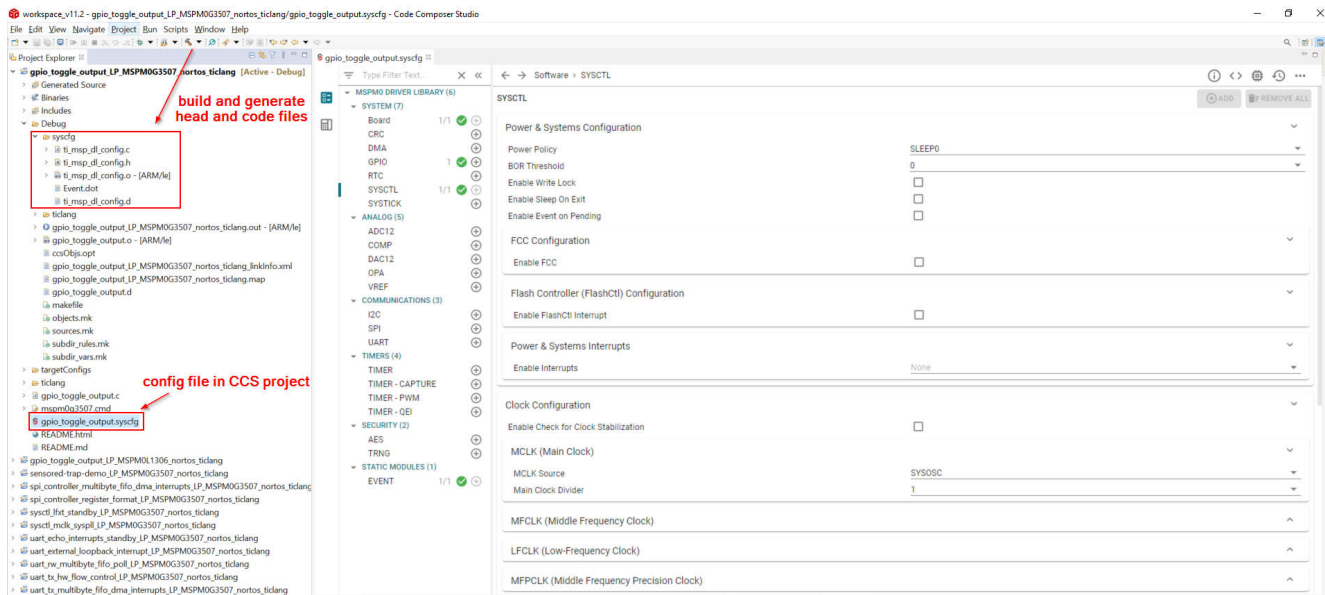


図 2-5. CCS を使用した SysConfig プロジェクト

2.4 MSP430 および MSPM0 プロジェクト

以下に、PWM 出力の例を示します。MSP430 レジスタ レベル コードでは、構成レジスタの各ビットを、クロック制御レジスタ、GPIO レジスタ、およびタイマ レジスタの関連する値に設定する必要があります。コードは読みにくく、変更も困難です。

```
//
//      MSP430FR5994
//
//      |-----|
//      | \   |
//      |  \  |
//      |   \ |
//      |    \|
//      |-----RST
//
//      |
//      |      P1.2/TA1.1|--> CCR1 - 75% PWM
//      |      P1.3/TA1.2|--> CCR2 - 25% PWM
//
//
//      William Goh
//      Texas Instruments Inc.
//      October 2015
//      Built with IAR Embedded Workbench V6.30 & Code Composer Studio V6.1
//*****

#include <msp430.h>

int main(void)
{
    WDCTL = WDTWP | WDTOLD;           // Stop WDT

    // Configure GPIO
    P1DIR |= BIT2 | BIT3;             // P1.2 and P1.3 output
    P1SEL0 |= BIT2 | BIT3;            // P1.2 and P1.3 options select
    P1SEL1 &= ~(BIT2 | BIT3);

    // Disable the GPIO power-on default high-impedance mode to activate
    // previously configured port settings
    PM5CTL0 &= ~LOCKLPM5;

    CSCTL0_H = CSKEY_H;               // Unlock CS registers
    CSCTL1 = DCOFSEL_6;               // Set DCO to 8MHz
    CSCTL2 = SELA__VLOCLK | SELS__DCOCLK | SELM__DCOCLK; // Set ACLK = VLO; SMCLK = DCO/8
    CSCTL3 = DIVA__8 | DIVS__8 | DIVM__8; // Set all dividers
    CSCTL0_H = 0;

    TA1CCR0 = 1000-1;                 // PWM Period
    TA1CCTL1 = OUTMOD_7;              // CCR1 reset/set
    TA1CCR1 = 750;                    // CCR1 PWM duty cycle
    TA1CCTL2 = OUTMOD_7;              // CCR2 reset/set
    TA1CCR2 = 250;                    // CCR2 PWM duty cycle
    TA1CTL = TASSEL__SMCLK | MC__UP | TACLK; // SMCLK, up mode, clear TAR

    __bis_SR_register(LPM0_bits);     // Enter LPM0
    __no_operation();                 // For debugger
}
```

図 2-6. MSP430 PWM 出力の例

MSPM0 デバイスで同じ機能を構成するには、構成ファイルを生成する SysConfig ツールでオプションを選択します。クロックを選択し、ツールで計算する PWM 周波数のカウント数を入力します。デューティサイクルを入力すると、ツールによって比較数値が生成されます。使用されるすべての GPIO 機能は SysConfig によって構成されます。SysConfig を使用すると、システムの設計と初期設定が容易になります。

SysConfig 出力は、driverlib を使用する SDK に基づいています。上記の MSP430 レジスタレベルのサンプルコードと比較すると、API はより読みやすくなっています。

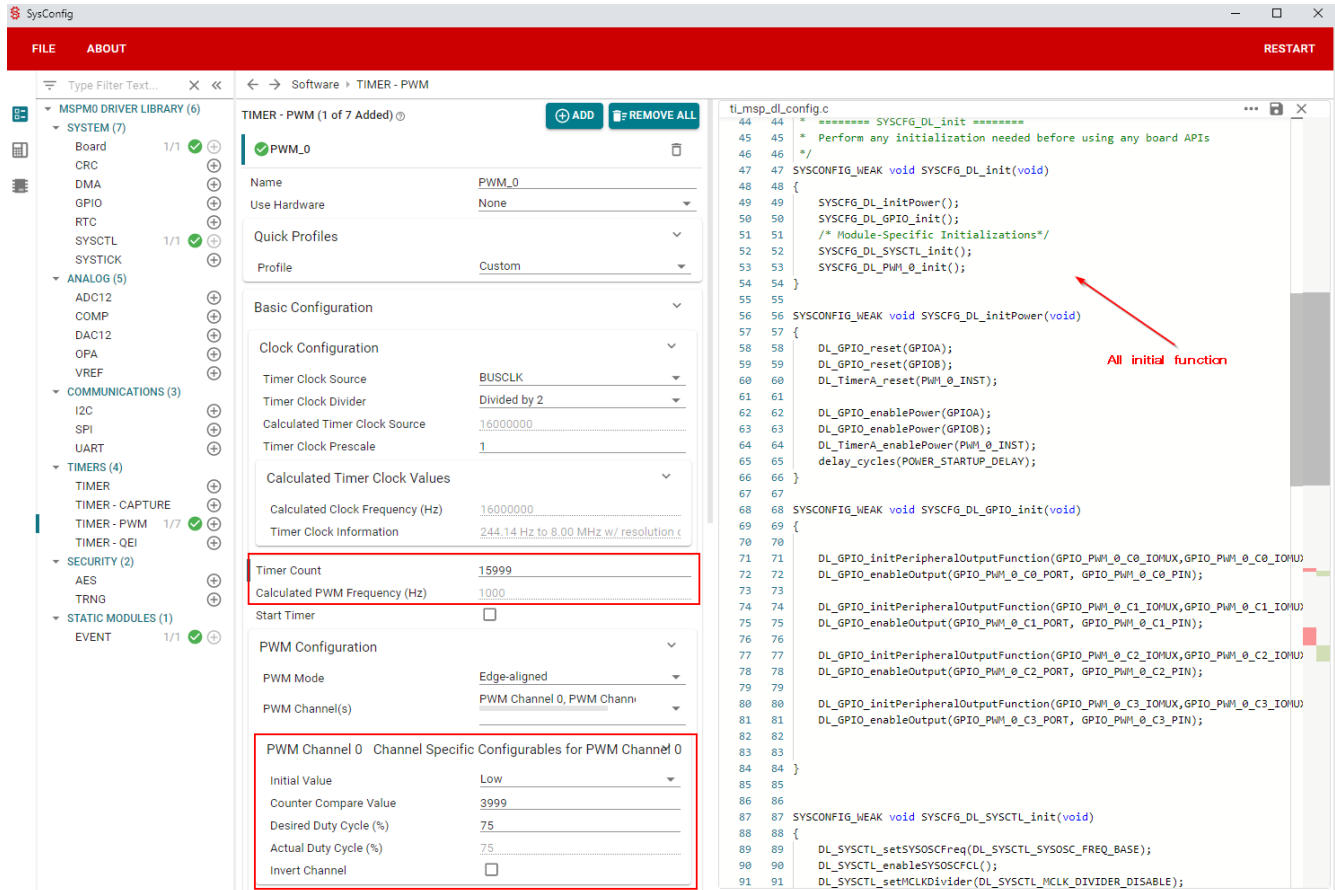


図 2-7. MSPM0 PWM 出力の例

2.5 デバッグ インターフェイス

MSP430 デバイスとは異なり、MSPM0 デバイスには Arm Cortex-M0+ コアが搭載されているため、サポートされるデバッグも異なります。表 2-3 に、これら 2 つのファミリの違いを示します。

表 2-3. デバッグ インターフェイス

ツール タイプ	MSP430	MSPM0
開発ツール	MSP-FET	XDS110/J-Link
量産向けツール	MSP-GANG	MSP-GANG / C-GANG

2.5.1 MSP430 デバッグ

MSP-FET (図 2-8 を参照) は、ユーザーが MSP430 デバイスの開発を迅速に開始するのに役立つ強力なエミュレーション開発ツールです。MSP-FET は、JTAG および SBW インターフェイスの両方でプログラミングとリアルタイム デバッグをサポートします。さらに、MSP-FET は、コンピュータの USB インターフェイスと MSP UART 間のバックチャネル UART 接続も提供します。これにより、MSP プログラムは、MSP とコンピュータ上で実行されている端末との間でシリアル通信を行う便利な方法を利用できるようになります。MSP-FET は、UART および I2C 通信プロトコルを介して BSL (ブートストラ

ップローダ)を使用してプログラム(ファームウェアと呼ばれることが多い)をMSPターゲットにロードする機能もサポートしています。詳しくは、[MSP デバッガ ユーザーガイド](#)を参照してください。



図 2-8. MSPFET

2.5.1.1 MSPFET 接続インターフェイス

USB インターフェイスから MSP-FET をコンピュータに接続し、14 ピン コネクタは標準 JTAG インターフェイスまたはピンを節約する Spy-Bi-Wire (2 線式 JTAG) プロトコルのいずれかを使用する MSP デバッグ エミュレーション ポートへのアクセスできます。これら 2 つのプロトコルは速度が異なります。

表 2-4. MSP430 MCU の SBW と JTAG 通信の速度

インターフェイス	低速	中	ファースト
SBW	200kHz	400kHz	600kHz
JTAG 4 線式 MSP430	1MHz	4MHz	8MHz

図 2-9 は、4 線式 JTAG 通信のシステム内プログラミングとデバッグをサポートするために必要な、14 ピン FET インターフェイス モジュール コネクタとターゲット デバイス間の接続を示しています。図 2-10 に、2 線式 JTAG モード (Spy-Bi-Wire) の接続を示します。4 線式 JTAG モードは、ピン数の少ないデバイス (MSP430G2230 など) を除くほとんどの MSP430 デバイスでサポートされています。2 線式 JTAG モードは、選択したデバイスでのみサポートされています。どのデバイスでどのインターフェイス方式が使用できるかについては、[MSP430 MCU 向け Code Composer Studio IDE ユーザーズガイド](#)または [MSP430 MCU 向け IAR Embedded Workbench IDE ユーザーズガイド](#)を参照してください。

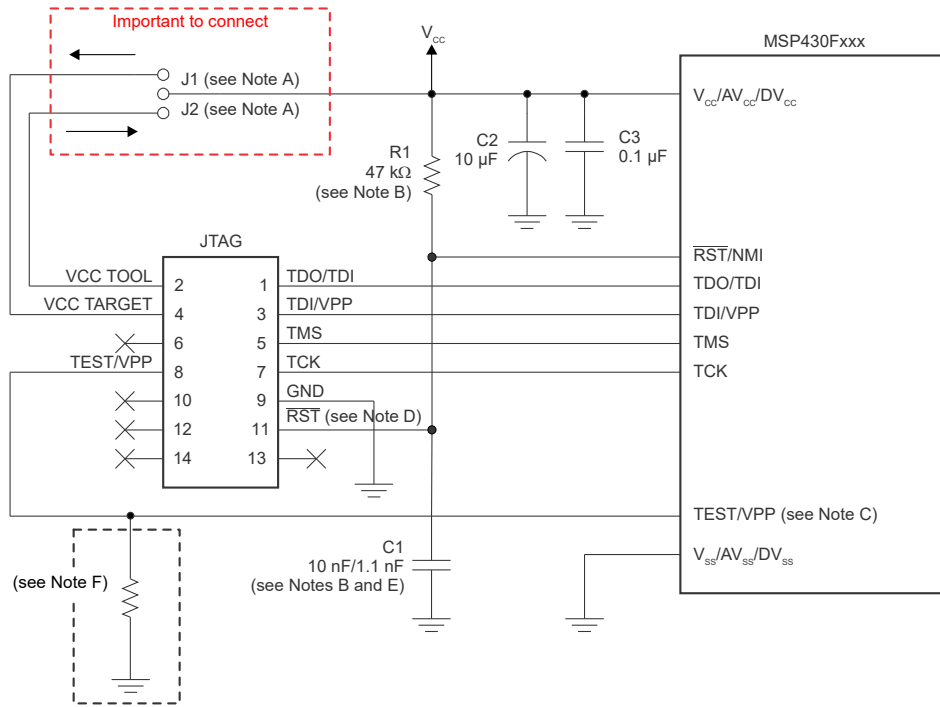


図 2-9. 4 線式 JTAG 通信の信号接続

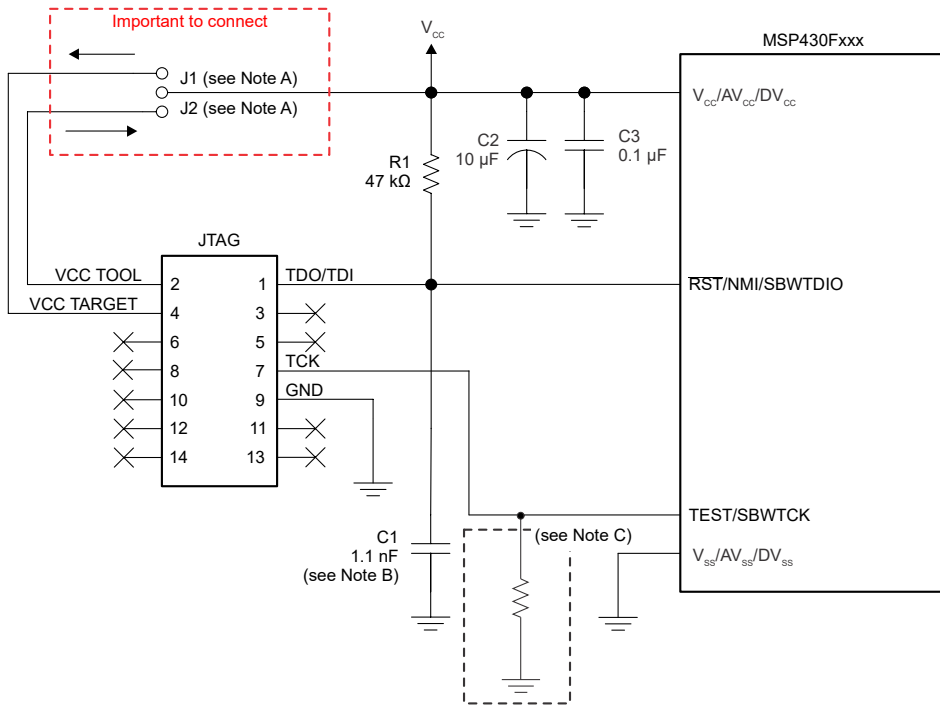


図 2-10. 2 線式 JTAG 通信 (Spy-Bi-Wire) の信号接続

注

一部の Spy-Bi-Wire 対応 MSP430 デバイスでは、TEST/SBWTK は立ち上がり信号エッジに非常に敏感であり、テスト ロジックがエントリ シーケンス (2 線式または 4 線式) が正しく認識されず、JTAG アクセスが無効のままになる状態になる可能性があります。JTAG コネクタがターゲット デバイスに接続されている場合、SBWTK に意図しないエッジが発生する可能性があります。

2.5.2 MSPM0 デバッグ

デバッグ サブシステム (DEBUGSS) は、シリアルワイヤ デバッグ (SWD) の 2 線式物理インターフェイスを、デバイス内の複数のデバッグ機能に接続します。MSPM0 デバイスは、プロセッサの実行、デバイスの状態、電力状態 (EnergyTrace テクノロジーを使用) のデバッグをサポートしています。デバッグの接続図 1-9 を参照してください。

MSPM0 は、標準的なシリアルワイヤ デバッグ用の XDS110 および J-Link デバッグをサポートしています。

テキサス・インスツルメンツの XDS110 は、TI の組み込みプロセッサ用です。XDS110 は、TI 20 ピン コネクタ (TI 14 ピンおよび Arm 10 ピンおよび 20 ピンコネクタ用の変換アダプタも利用可能) を使用してターゲット ボードに接続し、USB 2.0 ハイスピード (480Mbps) でホスト PC に接続します。XDS110 は、単一のユニットで幅広い規格 (IEEE1149.1、IEEE1149.7、SWD) をサポートしています。すべての XDS デバッグ プローブは、ETB (Embedded Trace Buffer、組み込みトレースバッファ) 搭載のすべての Arm と DSP プロセッサに対し、コアトレースとシステムトレースをサポートしています。詳細については、[XDS110 デバッグ プローブ](#) を参照してください。

J-Link デバッグ プローブは、デバッグとフラッシュプログラミングの経験を最適化するための最も一般的な選択肢です。これらのプローブは、最高記録を達成したフラッシュローダ、最大 3MiB/秒の RAM ダウンロード速度、および MCU のフラッシュメモリに無制限の数のブレークポイントを設定できる機能を活用できます。また、J-Link は Cortex-M0+ を含む幅広い CPU とアーキテクチャもサポートしています。詳細については、[SEGGER J-Link デバッグ プローブのページ](#) を参照してください。

図 2-11 に、XDS110 プローブから MSPM0 ターゲットへの主要な機能領域とインターフェイスの概略図を示します。

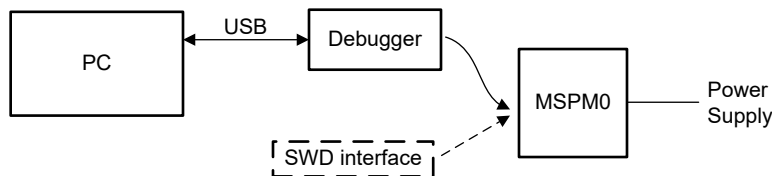


図 2-11. XDS 110 プローブから MSPM0 ターゲットへ

2.5.2.1 MSPM0 デバッグ ポートピンとピン配置

デバッグ ポートは、内部プルダウンおよびプルアップ抵抗を備えた SWCLK ピンと SWDIO ピンを使用します (図 2-12 を参照)。MSPM0 MCU ファミリーは、さまざまなピン数のさまざまなパッケージで提供されています。詳細は、デバイス固有のデータシートを参照してください。

表 2-5. MSPM0 SWD インターフェイス

デバイス信号	方向	SWD の機能
SWCLK	入力	デバッグ プローブからのシリアルワイヤクロック
SWDIO	入力 / 出力	双方向 (共有) シリアルワイヤデータ

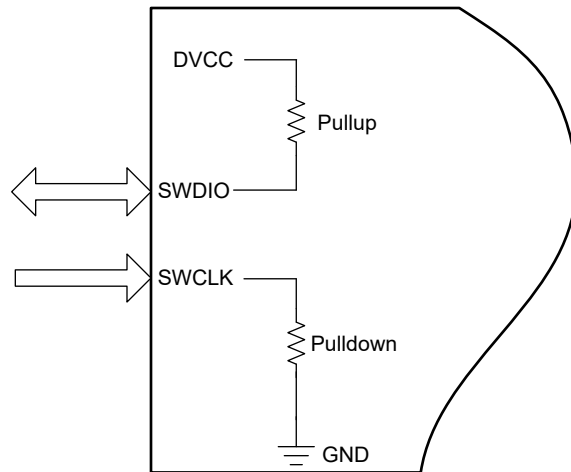


図 2-12. MSPM0 SWD 接続

図 2-13 は、MSPM0 SWD デバッグ ポートと標準の 10 ピン JTAG コネクタ間の接続を示しています。

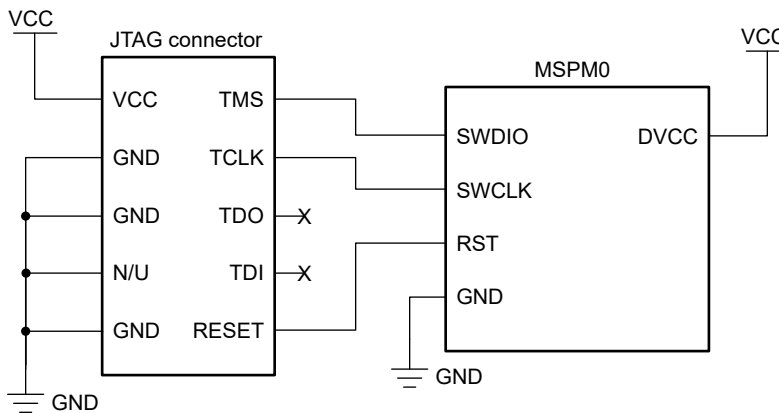


図 2-13. MSPM0 10 ピン JTAG への接続

3 移行にあたっての考慮事項

3.1 周辺機器

表 3-1 は、MSP デバイス ファミリの周辺機器を比較しています。各デバイスでサポートされている周辺機器については、デバイス固有のデータシートを参照してください。

表 3-1. 周辺機器の比較

ペリフェラル	MSP430	MSPM0C	MSPM0L	MSPM0G
CPU	最高 25MHz	最大 24/32MHz	最高 32MHz	最高 80MHz
SAR ADC	8、10、または 12 ビットの SAR ADC (デバイスに依存する)	12 ビット、1.5/1.68MSPS SAR ADC	12 ビット、1.68MSPS SAR ADC	14 ビット、4MSPS SAR ADC オーバーサンプリング (4MSPS ADC)
シグマ デルタ ADC	16 ビットまたは 24 ビット SD ADC	該当なし	該当なし	該当なし
COMP	対応	N/A / ウィンドウ比較	ウィンドウ比較	ウィンドウ比較
OPA、GPAMP	最大 2.8M の GBW	該当なし	最大 6M のパナアウト回路 GBW	最大 6M のパナアウト回路 GBW
DAC	8 ビットまたは 12 ビット DAC	8 ビット DAC	8 ビット DAC	12 ビット DAC
UART	IrDA をサポート	LIN、IrDA、ISO7816、RS485 をサポート	LIN、IrDA、ISO7816、RS485 をサポート	LIN、IrDA、ISO7816、RS485 をサポート
REF	1.5V、2V、2.5V	内部 1.4V または 2.5V	内部 1.4V または 2.5V	内部 1.4V または 2.5V
SPI	最高 8MHz	最大 12/16MHz、TI および Motorola モード、コマンドライン モードをサポート	最大 16MHz、TI および Motorola モード、コマンドライン モードをサポート	最大 32MHz、TI および Motorola モード、コマンドライン モードをサポート
I2C	100kHz	最大 Fm+ 1Mhz、ハードウェア SMBus および PMBus をサポート	最大 Fm+ 1Mhz、ハードウェア SMBus および PMBus をサポート	最大 Fm+ 1Mhz、ハードウェア SMBus および PMBus をサポート
CAN-FD	該当なし	該当なし	該当なし	対応
CRC	CRC16-CCITT	CRC-16-CCITT	CRC16-CCITT、CRC32-ISO3309	CRC16-CCITT、CRC32-ISO3309
AES	128 ビット、192 ビット、または 256 ビットのキーを持つ AES 128 ビット データ	該当なし	128 ビット、192 ビット、または 256 ビットのキーを持つ AES 128 ビット データ	128 ビット、192 ビット、または 256 ビットのキーを持つ AES 128 ビット データ
TRNG	該当なし	該当なし	該当なし	32 ビットの真のランダム出力
TIMER	一般的なタイマ機能、PWM、キャプチャと比較	一般的なタイマ機能、PWM、キャプチャと比較	一般的なタイマ機能、PWM、キャプチャと比較	一般的なタイマ機能、エッジまたはセンターアライン、補完、デッドバンド機能を備えた PWM
WWDT	1	1	1	2
RTC	対応	対応	対応	対応
DEBUG	MSPFET: JTAG (4 線式) または SBW	XDS110 または J-link: SWD	XDS110 または J-link: SWD	XDS110 または J-link: SWD
LCD	対応	該当なし	対応	該当なし
USS	対応	該当なし	該当なし	該当なし
CapTIvate 技術	対応	該当なし	該当なし	該当なし
USB	対応	該当なし	該当なし	該当なし

3.2 システム クロック

3.2.1 発振器

MSP430 および MSPM0 デバイスには、低システムコストと低消費電力のために、内部および外部の多くのタイプのクロックソースが用意されています。表 3-2 に、MSP430 および MSPM0 デバイスの各種クロックソースを示します。すべて

のデバイスにすべてタイプのクロックソースがあるわけではないことに注意してください。詳細は、デバイス固有のデータシートを参照してください。

表 3-2. 発振器の比較

タイプ	MSP430	MSPM0C	MSPM0L	MSPM0G
内部発振器	DCOCLK: 25MHz までの内部高速度発振器	SYSOSC: 内部発振器 (4MHz ~ 24/32MHz)	SYSOSC: 内部発振器 (4MHz ~ 32MHz)	SYSOSC: 内部発振器 (4MHz ~ 32MHz)。この発振器は、80MHz の PLL に供給できます
	REFOCLK: FLL を供給できる内部 32kHz 発振器	LFOSC: 32kHz の低周波数内部発振器	LFOSC: 32kHz の低周波数内部発振器	LFOSC: 32kHz の低周波数内部発振器
	VLOCKL: 内部 10kHz 超低消費電力発振器	該当なし	該当なし	該当なし
外部発振器	XT1CLK: 高周波発振器または低周波発振器 (32kHz)。デバイスによって異なります。	LFXT: 低周波数外部発振器	LFXT: 低周波数外部発振器	LFXT: 低周波数外部発振器
	XT2CLK: 高周波発振器デバイスによって異なります。	HFXT: 高周波数外部発振器	HFXT: 高周波数外部発振器	HFXT: 高周波数外部発振器

3.2.1.1 MSPM0 発振器

MSPM0 デバイスには、低システムコストと低消費電力のために、内部および外部の多くのタイプのクロックソースが用意されています。表 3-3 に、MSPM0 デバイスのクロックソースを示します。すべてのデバイスにすべてタイプのクロックソースがあるわけではないことに注意してください。詳細は、デバイス固有のデータシートを参照してください。

表 3-3. MSPM0 マイコンの発振器

タイプ	クロック・ソース	説明
内部	SYSOSC	システム発振器 (工場調整周波数 4 または 32MHz、ユーザー調整周波数 16 または 24MHz)
	LFOSC	低周波数発振器 (代表的な周波数 32kHz)
	SYSPLL	プログラム可能な周波数を備えたシステム PLL
外部	LFXT	低周波ローパワー水晶発振器 (代表的な周波数 32kHz)
	HFXT	高周波水晶発振器 (標準周波数 4 ~ 48MHz)

3.2.2 クロック信号

システムの電力消費を削減するために、さまざまな低電力モードでさまざまなクロック信号がサポートされています。次のセクションでは、MSP430 および MSPM0 MCU のクロック信号の違いについて説明します。

表 3-4. クロック信号の比較

クロック	MSP430	MSPM0C	MSPM0L	MSPM0G
MCLK	ソース CPU と一部のデジタル周辺機器	PD1 バスおよび周辺機器のメインシステムクロック	PD1 バスおよび周辺機器のメインシステムクロック	PD1 バスおよび周辺機器のメインシステムクロック
CPUCLK	該当なし	MCLK から生成された CPU クロック	MCLK から生成された CPU クロック	MCLK から生成された CPU クロック
SMCLK /ULPCLK	CPU から独立して動作する周辺機器用のサブシステムメインクロック	PD0 周辺機器と PD0 バスのメインシステムクロック (MCLK から生成)	PD0 周辺機器と PD0 バスのメインシステムクロック (MCLK から生成)	PD0 周辺機器と PD0 バスのメインシステムクロック (MCLK から生成)
ACLK	補助クロック 32kHz	固定 32kHz クロック	固定 32kHz クロック	固定 32kHz クロック
MFCLK	該当なし	MCLK に同期した固定 4MHz	MCLK に同期した固定 4MHz	MCLK に同期した固定 4MHz
MFPCCLK	該当なし	固定 4MHz	固定 4MHz	固定 4MHz

3.3 動作モード

すべての MSP デバイスには複数の低電力モードがあります。表 3-5 は、MSP430 および MSPM0 MCU の動作モードをリストします。

表 3-5. 動作モードの比較

MSP430		MSPM0C、MSPM0L および MSPM0G	
動作モード	説明	動作モード	説明
アクティブ	CPU、クロック、ペリフェラルが動作	RUN	CPU、クロック、ペリフェラルが動作
LPM0	CPU クロック ダウンでアクティブ	SLEEP	CPU クロック ダウンで動作
LPM2、LPM3	ACLK クロックが有効、一部の周辺機器はオプション	ストップ	PD0 有効および PD1 無効。利用可能クロック: MFCLK または LFCLK
		STANDBY	PD0 有効および PD1 無効利用可能クロック: LFCLK
LPM4	すべてのクロックはオフ、一部の周辺機器はオプション	シャットダウン	デバイスのシャットダウン
LPM3.5	RTC でデバイスをシャットダウン		
LPM4.5	デバイスのシャットダウン		

3.4 不揮発性メモリ (NVM)

3.4.1 MSPM0 メモリ保護ユニット

MSPM0 MCU は、プログラマによって定義されたアクセス許可ポリシー設定に基づいて、プロセッサによって行われたすべてのメモリ アクセスをチェックするメモリ保護ユニットをサポートしています。

MPU は、システム プライベート ペリフェラル バス (PPB) 領域のメモリ マップ レジスタを通じて構成されます。MPU レジスタ構成の詳細については、[Cortex-M0+ デバイスの汎用ユーザー ガイドの MPU セクション](#)を参照してください。

3.4.2 MSP430 FRAM および MSPM0 フラッシュ

FRAM は標準的な SRAM と同様に読み取り/書き込みできる不揮発性メモリです。FRAM は MSP430FRxx デバイスで使用できます。

表 3-6. MSP430 FRAM および MSPM0 フラッシュ

機能	MSP430 FRAM	MSPM0 フラッシュ
アクセス権	ワードまたはバイト書き込みアクセス	単一のフラッシュワード (64 ビット) または複数のワード
ECC	対応	対応 (MSPM0G/L デバイス)
保護	対応	対応
サイクル	10 ¹⁵	100K (下位 32KB) または 10k (32KB 超過)

FRAM には速度制限があります。CPU のシステム クロックは、FRAM アクセスおよびサイクル タイムの要件を超える場合があります。このようなシナリオでは、待機状態ジェネレータ メカニズムが実行されます。FRAM メモリの最大アクセス速度は 8MHz です。MCLK が 8MHz より高速に動作していて、FRAM アクセスが必要な場合は、信頼性の高い FRAM アクセスのためには待機状態になっている必要があります。MCLK ≥ 8MHz を使用する場合は、MCLK 周波数を設定する前に、ソフトウェアで FRAM 待機状態を設定します。

3.4.3 MSP430 フラッシュおよび MSPM0 フラッシュ

表 3-7 に、MSP430 フラッシュ デバイスと MSPM0 フラッシュ デバイスの主な機能を示します。詳細については、該当するユーザーガイドを参照してください。

表 3-7. MSP430 フラッシュおよび MSPM0 フラッシュ

機能	MSP430 フラッシュ	MSPM0 フラッシュ
電圧生成	内部	内部
プログラム モード	バイト、ワード (2 バイト)、ロング (4 バイト)	単一のフラッシュワード (64 ビット) または複数のワード
消去	セグメント、バンク、マス	セクタ、バンク、マス
ECC	該当なし	対応 (MSPM0G/L デバイス)
サイクル	100k	100K (下位 32KB) または 10k (32KB 超過)

3.5 イベントと割り込みハンドラ

MSP430 MCU では、割り込み優先順位は固定されており、[図 3-1](#) に示すように接続チェーン内のモジュールの配置によって定義されます。割り込みには、システムリセット、(非) マスク可能、マスク可能の 3 種類があります。

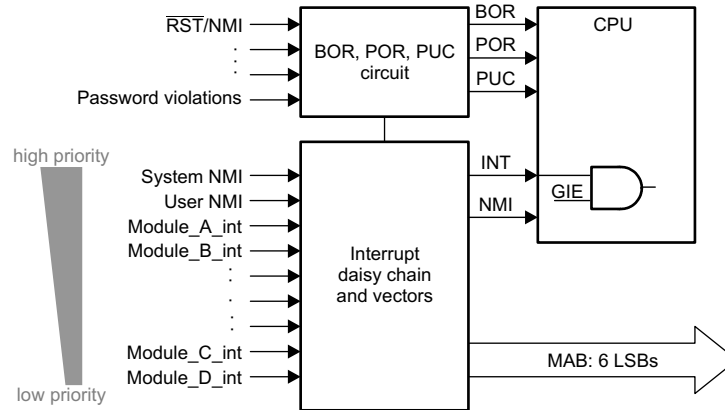


図 3-1. MSP430 イベントおよび割り込み処理

MSPM0 マイコンには、ある機能から別の機能へデジタルイベントを送るイベント マネージャが搭載されています。イベント マネージャは、静的なルートとプログラマブルなルートの組み合わせを含むイベント ファブリックによって相互接続された一連の定義済みイベント パブリッシャ (ジェネレータ) およびサブスクライバ (レシーバ) によるイベント転送を実装しています。

イベント マネージャによって転送されるイベントには、以下が含まれます。

- 割り込み要求 (IRQ) として CPU に転送されるペリフェラル イベント
- DMA トリガとして DMA に転送されるペリフェラル イベント
- ハードウェアでの動作を直接トリガするため、別のペリフェラルに転送されるペリフェラル イベント

イベント マネージャは、イベントファブリックを介してイベント パブリッシャをイベント サブスクライバに接続します。イベント ファブリックには、静的イベント ルート、DMA イベント ルート、汎用イベント ルートの 3 種類があります。

[図 3-2](#) にイベント マップを示します。さまざまなイベント遷移を実現するために、さまざまなペリフェラルがさまざまなイベントファブリックを介してルーティングされます。詳細については、に記載しているデバイスのテクニカル リファレンス マニュアルを参照してください。

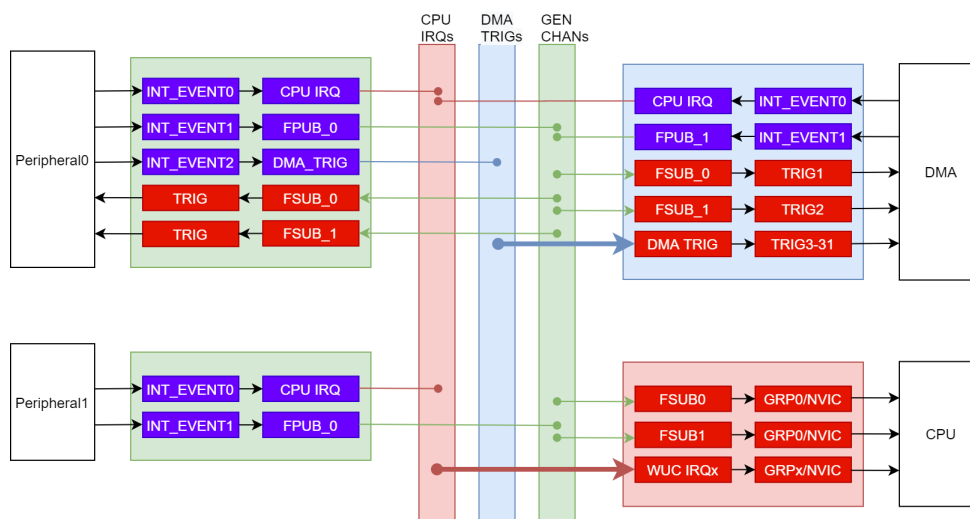


図 3-2. MSPM0 イベントおよび割り込み処理

3.6 リセットレベル

MSP430 MCU は 3 つのリセットレベルをサポートしています。POR、BOR、PUC。各リセットは、デバイスファミリのユーザーガイドに記載されている特定のイベントによってトリガされます。これらのトリガ信号の関係は次のとおりです。

- BOR が生成されるときには必ず POR も生成されますが、POR によって BOR が生成されるわけではありません。
- POR が生成されるときには必ず PUC も生成されますが、PUC によって POR が生成されるわけではありません。

MSPM0 MCU には 5 つのリセットレベルがあります。

1. パワーオンリセット (POR)
2. 電圧低下リセット (BOR)
3. ブートリセット (BOOTRST)
4. システムリセット (SYSRST)
5. CPUリセット (CPURST)

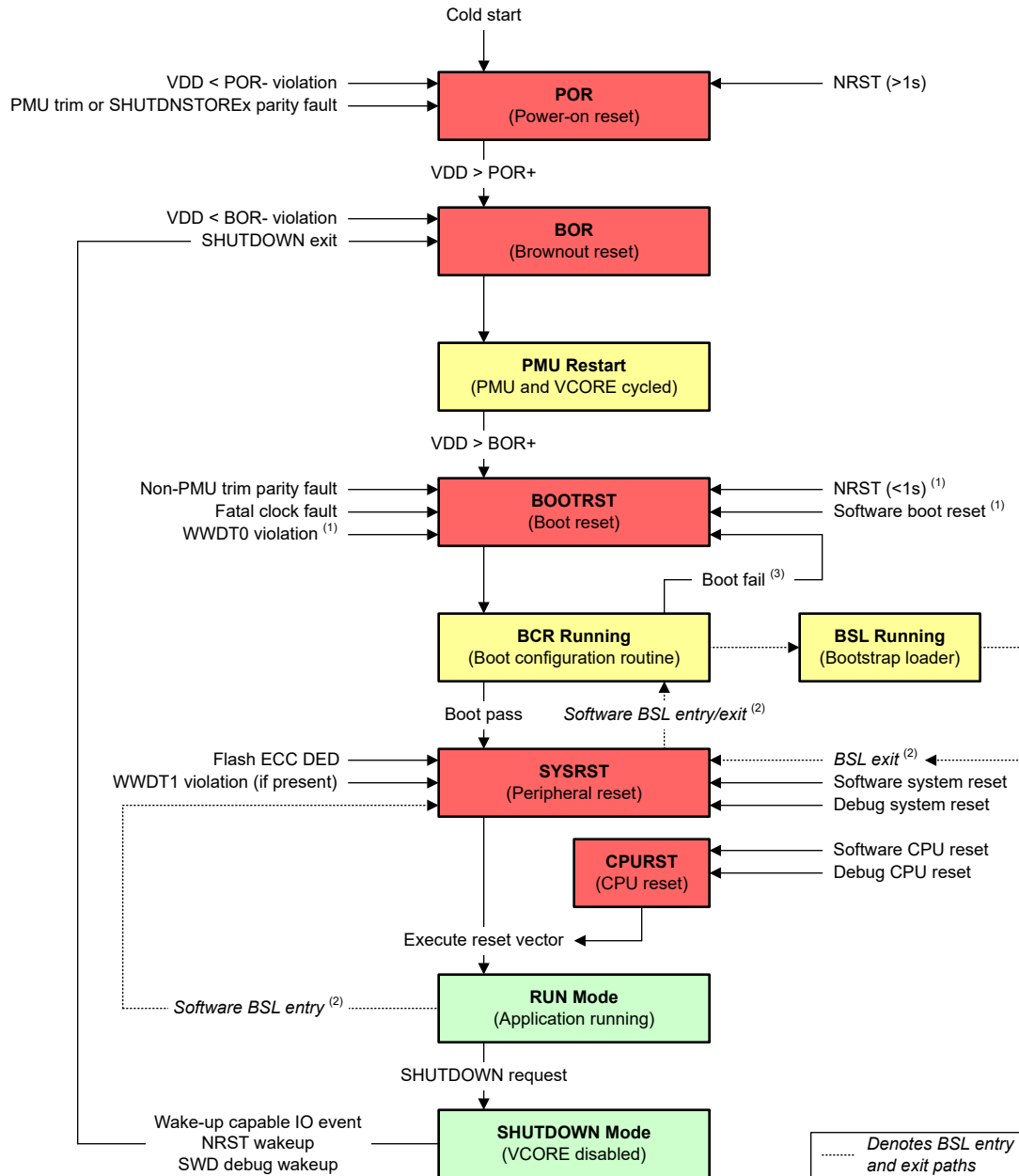


図 3-3. MSPM0 リセット

3.7 GPIO とピンのマルチプレクシング

MSP430 マイコンでは、ADC 出力または DAC 出力、CapTIvate I/O、eUSCI などのマルチプレクサ機能によって、ピンがデフォルトでデジタル I/O に設定されています。デジタル I/O 機能には次のものが含まれます。

- 個別にプログラミング可能な I/O
- 任意の入力または出力の組み合わせ

- P1 および P2 割り込みを個別に構成可能。一部のデバイスには、追加のポート割り込みが含まれています
- 独立した入力/出力データレジスタ
- プルアップ抵抗またはプルダウン抵抗を個別に構成可能

MSPM0 デバイスでは、デフォルトですべてのピンがアナログ機能に設定されています。

IOMUX は、デジタル IO の構成を管理します。IOMUX によって構成される主な機能は次のとおりです。

- 各デジタル IO ピンにマルチプレクサされる周辺機器 (GPIO または UART など) の選択
- デジタル入力パスの構成
 - ヒステリシス制御
 - 入力パスの有効化または無効化
 - 入力ロジック反転制御
- デジタル出力パス構成
 - 駆動能力制御
 - 出力接続の有効化または無効化
 - 出力ロジックの反転 (入力ロジックの反転と共有される制御)
 - ロジック High からハイ インピーダンスへの出力変換 (オープンドレイン形式インターフェイス用)
 - IO に接続されている周辺機器が無効の場合、「最後の状態」の保持
- ウェークアップ構成 (シャットダウン モードからのウェークアップ用)
 - ウェークアップ比較レベル
 - ウェークアップの有効化または無効化
- プルアップ抵抗またはプルダウン抵抗の制御

3.8 通信インターフェイス

MSPM0 MCU の通信インターフェイス性能は、MSP430 MCU の性能よりも優れています。MSPM0 MCU には、より豊富なアプリケーションが搭載することができます。

3.8.1 SPI

SPI モジュールは、SPI プロトコル (センサ、メモリ、ADC、DAC など) を使用して MSPM0 デバイスとその他の外部デバイス間でデータを転送するための標準化されたシリアル インターフェイスを提供します。

SPI モジュールには次の機能があります。

- コントローラまたはペリフェラルとして構成可能
- プログラマブルなクロック ビットレートとプリスケアラ
- 送信 (TX) と受信 (RX) の個別の先入先出バッファ (FIFO)
- 4 ビットから 16 ビットまでのプログラム可能なデータ フレーム サイズ (コントローラ モード)
- データ フレーム サイズを 7 ビットから 16 ビットまでプログラム可能 (ペリフェラル モード)
- 送信および受信 FIFO の割り込み、オーバーランおよびタイムアウト割り込み、DMA 完了
- プログラム可能な SPI モードは、Motorola SPI、MICROWIRE、またはテキサス インストルメンツ形式をサポート
- ダイレクト メモリ アクセス コントローラ インターフェイス (DMA) :
 - 送信用と受信でチャンネルを分離
 - 転送完了割り込み

3.8.2 I2C

I2C モジュールは、MSP デバイスと他の外部 I2C デバイス (センサ、メモリ、DAC など) 間でデータを転送するための標準化されたシリアル インターフェイスとなります。I2C 周辺機器は、データ (SDA) ラインとクロック (SCL) ラインで構成される 2 線式シリアル バスを介して双方向のデータ転送を実現します。I2C バスは、シリアル メモリ (RAM および ROM)、ネットワーク デバイス、LCD、トーン ジェネレータなどの外部 I2C デバイスとのインターフェイスです。I2C バスはまた、製品の開発時や製造工程において、システムのテストや診断を行う目的で使用することも可能です。この I2C 周辺機器は、バス上の他の I2C デバイスとの間で送信と受信を行うことができます。

コントローラには、次の機能を備えた I2C モジュールが含まれています。

- I2C バス上のデバイスは、コントローラまたはターゲットのいずれかとして指定できます。

- 7 ビットアドレス指定によるコントローラまたはターゲットとしてデータの送信と受信の両方をサポート
- 4 つの I2C モードをサポート
 - コントローラ送信
 - コントローラ受信
 - ターゲット送信
 - ターゲット受信
- サポートされている送信速度: 標準 (100kbps)、高速 (400kbps)、高速 + (1Mbps)
- 受信および送信データ用の FIFO、8 バイト
- グリッチ抑制
- 独立したコントローラとターゲットの割り込み生成
- アービトレーション(調停) 機能、クロック同期、マルチコントローラ サポートを備えたコントローラ操作
- SMBus 用ハードウェア サポート
 - クロック Low タイムアウト割り込み
 - デュアル ターゲット アドレス機能
 - クイック コマンド機能
- 送信と受信に別々のチャンネルを備えた DMA のハードウェア サポート

3.8.3 UART

このインターフェイスは、LIN (ローカル相互接続ネットワーク)、ISO7816 (スマート カード プロトコル)、IrDA (赤外線データ アソシエーション)、ハードウェア フロー制御 (CTS/RTS)、およびマルチプロセッサ通信などのシリアル非同期通信プロトコルを使用して、MSPM0 デバイスと別のデバイス間でデータを転送するために使用します。

UART コントローラは、MSPM0 にさまざまな機能を持たせています。

- 送信と受信を分離した 4 深度の FIFO により CPU の割り込みサービス負荷を軽減
- DMA データ転送をサポート
- ループバック モード動作をサポート
- ハードウェア フロー制御をサポート
- 9 ビットのマルチドロップ構成をサポート
- サポートされているプロトコル:
 - ローカル相互接続ネットワーク (LIN) のサポート
 - DALI
 - IrDA
 - ISO/IEC 7816 スマート カード
 - RS485
 - マンチェスター符号化
 - アイドル ライン マルチプロセッサ

3.8.4 CAN FD

MSP430 とは異なり、CAN 制御は MSPM0 デバイスに統合されています。コントローラ エリア ネットワーク (CAN) は、高い信頼性で分散リアルタイム制御を効率的にサポートするシリアル通信プロトコルです。CAN は電氣的干渉に対する高い耐性を持ち、さまざまな種類のエラーを検出できます。CAN では、多くのショート メッセージがネットワーク全体にブロードキャストされるため、システムのすべてのノードでデータの整合性が確保されます。

MCAN モジュールは、従来型の CAN と CAN FD (CAN とフレキシブル データレート) の両方のプロトコルをサポートしています。CAN FD 機能により、データフレームあたりのスループットが向上し、ペイロードが増加します。従来型の CAN デバイスと CAN FD デバイスは、バス エラーを生成せずに CAN FD を検出および無視できる部分的ネットワークトランシーバが従来型の CAN デバイスで使用されている場合、競合なしで同じネットワーク上に共存できます。MCAN モジュールは、ISO 11898-1:2015 に準拠しています。

CAN コントローラと機能の詳細については、MSPM0 テクニカル リファレンス マニュアルを参照してください。

3.9 BSL

表 3-8 は、MSP430 と MSPM0 BSL 実装を比較したものです。

表 3-8. BSL の比較

		MSP430									MSPM0	
		F20xx、G2xx0、G2xx1、G2xx2、I20xx	F1xx、F2xx、F4xx、G2xx3	F5xx、F6xx		FR5xx、FR6xx		FR231x、FR242、FR243x、FR25xx、FR263x	FR215x、FR235x、FR247x、FR267x	FR20xx、FR21xx、FR41xx	M0Cxxx x	M0Gxxx x M0Lxxxx
				非 USB	USB	ファクトリ	暗号化ブートローダ					
総則	BSL メモリタイプ	BSL なし	ROM	フラッシュ	フラッシュ	ROM	FRAM	ROM	ROM	ROM	BSL なし	ROM
	BSL メモリのサイズ	該当なし	1KB	2KB	2KB	2KB	4KB	3KB	3KB	1KB	該当なし	5K
	ユーザー設定								✓			✓
	UART		✓	✓		✓	✓	✓	✓	✓		✓
	I2C			✓		✓	✓	✓	✓			✓
	SPI											
	USB				✓							
	フラッシュ ベースのインターフェイス プラグインをサポート											✓
メカニズムを起動	I/O の入力シーケンス	TEST/RST のシーケンス	✓	✓		✓	✓	✓	✓	✓		
		PUR ピンを VUSB に接続			✓							
		定義された I/O のシーケンス					✓					✓
	リセットベクタが空の場合、BSL を呼び出します				✓		✓	✓	✓			✓
	ソフトウェア アプリケーションから BSL を呼び出します		✓	✓	✓	✓	✓	✓	✓	✓		✓
ツールのサポート	ハードウェア	MSP-BSL 「ロケット」	✓	✓		✓	✓	✓	✓	✓		
		MSP-FET		✓		✓	✓	✓	✓	✓		(1)
		USB ケーブル			✓							
		USB-to-Serial コンバータ	✓									✓
		XDS110										✓
	ソフトウェア	BSL スクリプト		✓	✓	✓	✓	✓	✓	✓		
		MSPM0 BSL GUI		✓								✓
セキュリティ	パスワード保護 (バイト)		32	32	32	32		32	32	32		32
	間違ったパスワードで一括消去		✓	✓	✓	✓		✓	✓	✓		✓(2)
	BSL ペイロード暗号化						✓					
	ブートコードによる IP 保護領域の更新											
	認証済み暗号化						✓					
セキュリティの強化						✓						

- (1) USB から UART または I2C チャンネルを使用して、BSL によって MSPM0 デバイスと通信できます。
(2) パスワードの入力を 3 回失敗すると、工場出荷時設定へのリセットを含むセキュリティアクションがトリガされます。

3.10 アナログ ペリフェラル

3.10.1 SAR ADC

表 3-9 は、MSP430 と MSPM0 の各 SAR ADC の実装を比較したものです。

表 3-9. ADC の比較

機能	MSP430	MSPM0C	MSPM0L	MSPM0G
分解能	10 ビットまたは 12 ビット	10 ビットまたは 12 ビット	10 ビットまたは 12b ビット	10 ビットまたは 12 ビット
サンプルレート	200ksps	1.68Msps	1.68Msps	12 ビット 4Msps
FIFO モード	非対応	FIFO	FIFO	FIFO
内蔵 REF	1.5V、2V、2.5V	1.4V、2.5V	1.4V、2.5V	1.4V、2.5V
オーバーサンプリング	非対応	対応	対応	対応
ウィンドウ コンパレータ	一部のデバイスでサポート	対応	対応	対応
同時サンプリング	非対応	非対応	非対応	対応

3.10.1.1 同時サンプリング

電流や電圧の検知などの一部のアプリケーションでは、複数のアナログ信号を同時に測定する必要があります。このような状況では、単一の MCU 上で複数の ADC を使用して同時サンプリングを実行することが必要になります。MSPM0xx プラットフォーム内の複数の ADC 周辺装置を備えたデバイスはすべて、同時サンプリングをサポートしています。

3.10.1.2 ウィンドウ コンパレータ

ADC には 1 つのウィンドウ コンパレータ ユニットがあり、これを使用して、入力信号がソフトウェアによって設定された定義済みのスレッシュホールド内にあるかどうかを確認できます。一部の MSPM0C、MSPM0L、および MSPM0G デバイスはこの機能をサポートしています。MEMRES または FIFO に入力される ADC の結果は、ウィンドウ コンパレータのスレッシュホールドと照合されます。

この比較に基づいて、ウィンドウ コンパレータは次の 3 つの割り込み条件を生成します。

1. LOWIFG – 変換結果が下限スレッシュホールド (WCLOW) を下回っている
2. HIGHIFG – 変換結果が上限スレッシュホールド (WCHIGH) を超えている
3. INIFG - 変換結果が Low スレッシュホールドと High スレッシュホールドの間か、それに等しい

3.10.2 COMP

MSPM0 デバイスには、MSP430 デバイスよりも優れた性能を備えた高速コンパレータが搭載されています。

表 3-10. COMP の比較

	MSP430	MSPM0C	MSPM0L	MSPM0G
複数の入力ソース	あり	あり	あり	あり
COMP 用ソフトウェア LPF	あり	あり	あり	あり
基準電圧	VREF と VDD	VDD	VDD	VREF と VDD
内蔵 DAC	6 ビット	8 ビット	8 ビット	8 ビット
伝搬遅延 (高速)	120ns ~ 1µs (デバイスによって異なる)	40ns	40ns	40ns
ウィンドウ比較モード	なし	あり	あり	あり
電圧ヒステリシス	プログラム可能	プログラム可能	プログラム可能	プログラム可能

3.10.2.1 ウィンドウ比較モード

一部の MSPM0G MCU には複数のコンパレータが搭載されています。ウィンドウ コンパレータ モードの目的は、下限スレッシュホールドと上限スレッシュホールドによって定義された指定されたスレッシュホールド範囲内で入力信号を監視することです。

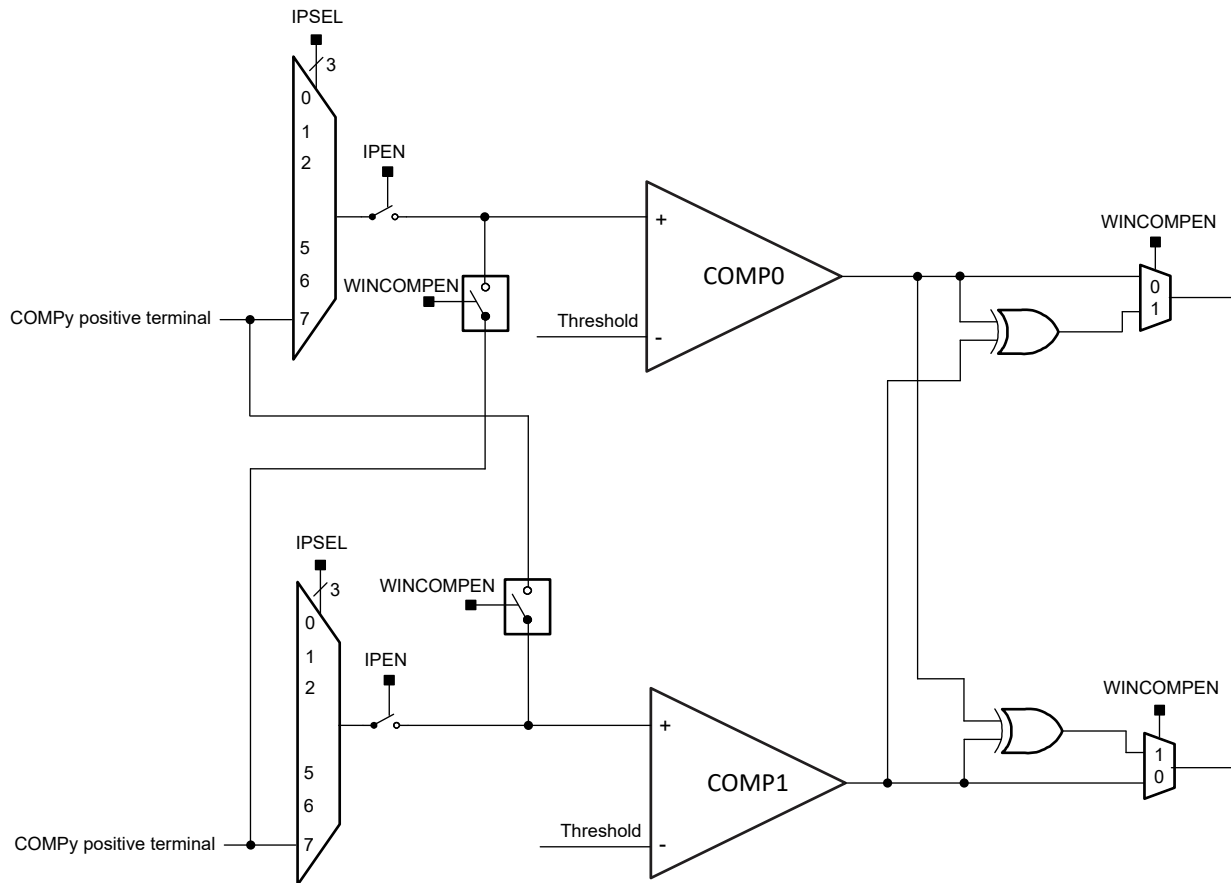


図 3-4. ウィンドウ比較モードのブロック図

3.10.3 OPA

MSP430 および MSPM0 MCU には、アナログ周辺機器が豊富に用意されています。これらの周辺機器は、特に OPA の内部アナログおよびデジタル IP の恩恵を受ける混合信号を処理する優れた能力を備えています。

表 3-11 は、MSP430 と MSPM0 のオペアンプの違いを比較したものです。

表 3-11. オペアンプの比較

	MSP430	MSPM0	MSPM0L、MSPM0G	
	SAC-OA	該当なし	OPA	GPAMP
GBW	2.8MHz	該当なし	6MHz	350kHz
PGA	33x	該当なし	32x	該当なし
相互作用	あり	なし	あり	あり
バーンアウト検出	なし	なし	あり	なし

図 3-5 に、代表的な使用事例での MSPM0 OPA のサンプル回路を示します。

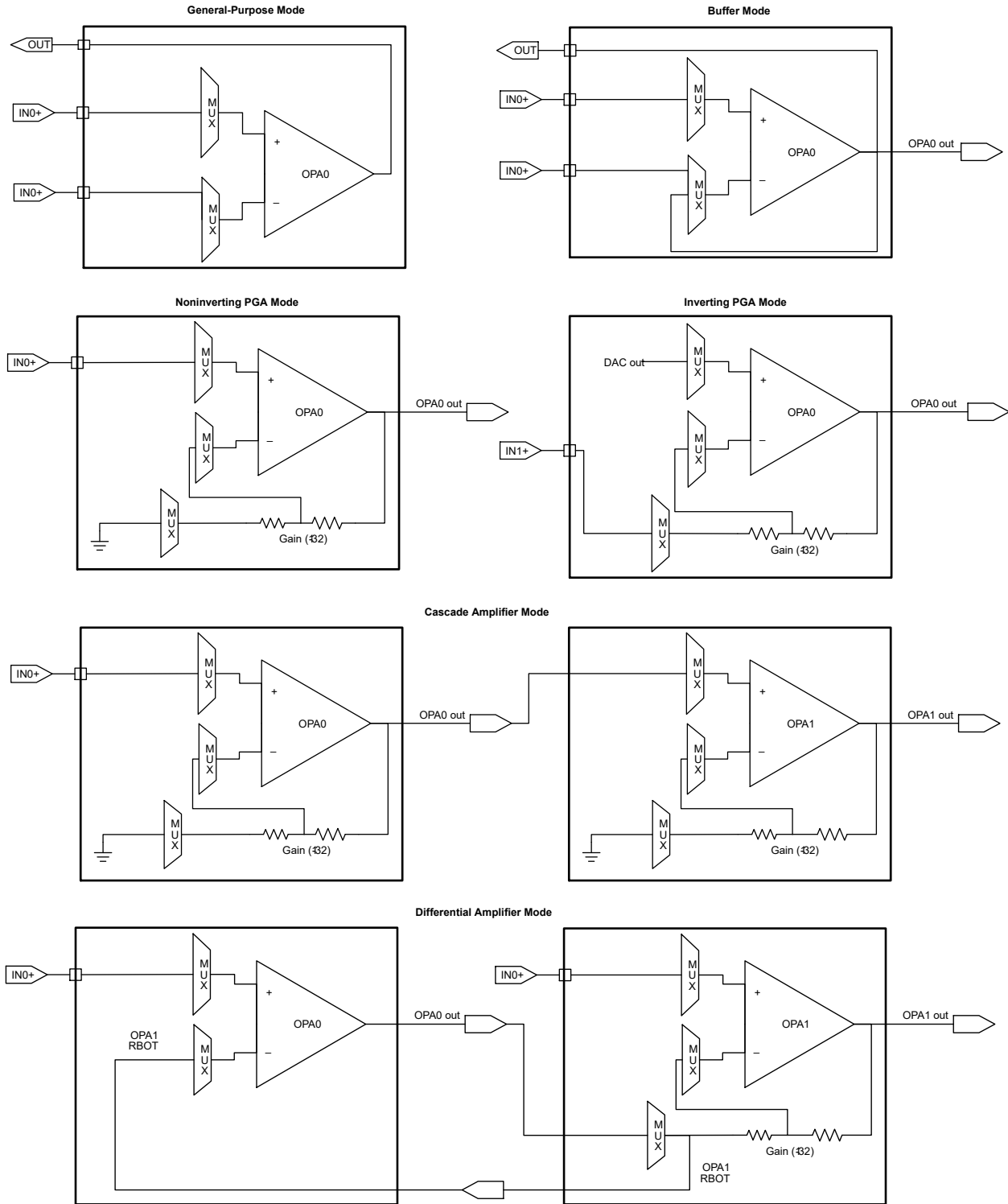


図 3-5. オペアンプの代表的な使用事例

3.11 タイマ

表 3-12 は、MSPM0 タイマ (TIMA、TIMH、TIMG) と MSP430 タイマ (Timer_A、Timer_B) を比較しています。

表 3-12. タイマの比較

	MSP430		MSPM0C、MSPM0L および MSPM0G		
	Timer_A	Timer_B	TIMA	TIMG	TIMH
カウンタ	16 ビット	16 ビット	16 ビット	16 ビット	24 ビット
カウント モード	アップまたはアップダウン	アップまたはアップダウン	ダウンまたはアップダウン	ダウンまたはアップダウン	ダウンまたはアップダウン
比較モード	あり	あり	あり	あり	あり
PWM 出力	あり	あり	あり	あり	あり
同期	なし	なし	あり	あり	あり
クロストリガ	なし	なし	あり	あり	あり
プログラミング デッドバンド 付き相補型 PWM	なし	なし	あり	なし	なし
QEI	なし	なし	なし	あり	なし

MSPM0 タイマは、モーター制御やその他の複雑な PWM 制御に使用できる相補型 PWM、位相シフト、デッドバンド、クロストリガなどの高度な機能をサポートしています。詳細については、ファミリの TRM およびアプリケーション ノートを参照してください。

3.12 ハードウェア設計ガイド

ハードウェア設計の考慮事項の詳細については、次のドキュメントを参照してください。

[MSPM0 C シリーズ MCU ハードウェア開発ガイド](#)

[『MSPM0 L シリーズ MCU ハードウェア開発ガイド』](#)

[『MSPM0 G シリーズ MCU ハードウェア開発ガイド』](#)

これらのガイドは、電源、リセット回路、クロック、デバッグ接続、主要なアナログ周辺機器、通信インターフェイス、GPIO、基板レイアウト ガイダンスなどの詳細なハードウェア設計情報を含む、MSPM0 MCU を使用したハードウェア開発の検討事項を解説しています。

3.13 改訂履歴

Changes from Revision A (March 2023) to Revision B (May 2025)	Page
• すべての比較表に MSPM0C ファミリを追加.....	1
• 表 2-3 の量産向けツールセクションに「C-Gang」を追加しました.....	8
• 最新のデバイスリリースを考慮して、MSPM0L と MSPM0G の比較表の仕様を更新.....	13
• 機能サポートを追加.....	25
• MSPM0 C シリーズ MCU ハードウェア開発ガイド を追加.....	25

4 改訂履歴

Changes from Revision * (December 2022) to Revision A (March 2023)

Page

-
- 全体を通してリンクと画像を更新.....1
-

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](https://www.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
Copyright © 2025, Texas Instruments Incorporated