

## User's Guide

ルネサス **RL78** から **ARM®** ベースの **MSPM0** への移行ガイド

## 概要

このアプリケーション ノートは、ルネサス **RL78** プラットフォームから テキサス・インスツルメンツの **MSPM0 MCU** エコシステムへの移行を支援します。このドキュメントでは、**MSPM0** の開発およびツール エコシステム、コア アーキテクチャ、ペリフェラルに関する考慮事項、およびソフトウェア開発キットについて説明します。この目的は、2 つのファミリの違いを強調し、**RL78** の開発環境に関する既存の知識を活用して、**MSPM0** シリーズの **MCU** で迅速に開発を開始することです。

## 目次

|                                                          |           |
|----------------------------------------------------------|-----------|
| <b>1 MSPM0 製品ラインアップの概要</b>                               | <b>4</b>  |
| 1.1 はじめに                                                 | 4         |
| 1.2 ルネサス <b>RL78 MCU</b> と <b>MSPM0 MCU</b> の製品ラインアップの比較 | 4         |
| <b>2 エコシステムと移行</b>                                       | <b>6</b>  |
| 2.1 エコシステムの比較                                            | 6         |
| 2.2 移行プロセス                                               | 13        |
| 2.3 例                                                    | 28        |
| <b>3 コア アーキテクチャの比較</b>                                   | <b>35</b> |
| 3.1 CPU                                                  | 35        |
| 3.2 組み込みメモリの比較                                           | 35        |
| 3.3 電源投入とリセットの概要と比較                                      | 38        |
| 3.4 クロックの概要と比較                                           | 40        |
| 3.5 <b>MSPM0</b> の動作モードの概要と比較                            | 41        |
| 3.6 割り込みとイベントの比較                                         | 43        |
| 3.7 デバッグとプログラミングの比較                                      | 48        |
| <b>4 デジタル ペリフェラルの比較</b>                                  | <b>49</b> |
| 4.1 汎用 I/O (GPIO、IOMUX)                                  | 49        |
| 4.2 UART (Universal Asynchronous Receiver-Transmitter)   | 51        |
| 4.3 シリアル・ペリフェラル・インターフェイス (SPI)                           | 51        |
| 4.4 I2C (Inter-Integrated Circuit)                       | 52        |
| 4.5 タイマ (TIMGx、TIMAx)                                    | 53        |
| 4.6 ウィンドウ付きウォッチドッグ タイマ (WWDT)                            | 54        |
| 4.7 リアルタイム クロック (RTC)                                    | 54        |
| <b>5 アナログ ペリフェラルの比較</b>                                  | <b>55</b> |
| 5.1 A/D コンバータ (ADC)                                      | 55        |
| 5.2 コンパレータ (COMP)                                        | 56        |
| 5.3 D/A コンバータ (DAC)                                      | 57        |
| 5.4 オペアンプ (OPA)                                          | 58        |
| 5.5 基準電圧 (VREF)                                          | 58        |
| <b>6 まとめ</b>                                             | <b>59</b> |
| <b>7 参考資料</b>                                            | <b>59</b> |
| <b>8 改訂履歴</b>                                            | <b>59</b> |

## 図の一覧

|                               |    |
|-------------------------------|----|
| 図 2-1. <b>MSPM0</b> エコシステムの概要 | 6  |
| 図 2-2. <b>MSPM0 SDK</b> の構造   | 7  |
| 図 2-3. <b>RL78</b> ソフトウェア開発環境 | 7  |
| 図 2-4. <b>MSPM0 SysConfig</b> | 9  |
| 図 2-5. ペリフェラルリストの比較           | 10 |

|                                                                 |    |
|-----------------------------------------------------------------|----|
| ☒ 2-6. 割り込み設定の比較.....                                           | 10 |
| ☒ 2-7. MSPM0 のデバッグ.....                                         | 11 |
| ☒ 2-8. MSPM0G3507 Launchpad の概要.....                            | 12 |
| ☒ 2-9. Arm Cortex 10 ピンの定義.....                                 | 12 |
| ☒ 2-10. MSPM0G3507 Launchpad の特徴と機能.....                        | 13 |
| ☒ 2-11. MSPM0 の移行フローチャート.....                                   | 13 |
| ☒ 2-12. MSPM0C、MSPM0L、MSPM0G シリーズの製品ラインアップ.....                 | 14 |
| ☒ 2-13. MSPM0 製品選択ツール.....                                      | 15 |
| ☒ 2-14. MSPM0 の重要ドキュメント一覧.....                                  | 15 |
| ☒ 2-15. MSPM0 関連技術ドキュメント一覧.....                                 | 15 |
| ☒ 2-16. 注文と品質情報の表示画面.....                                       | 16 |
| ☒ 2-17. CCS のインストール.....                                        | 16 |
| ☒ 2-18. CCS のインストール - MSPM0 サポートの選択.....                        | 17 |
| ☒ 2-19. CCS のインストール - J-Link ダウンロードの選択.....                     | 17 |
| ☒ 2-20. CCS Launch Workspace.....                               | 18 |
| ☒ 2-21. CCS で新しいプロジェクトを作成する.....                                | 18 |
| ☒ 2-22. よく使う機能.....                                             | 19 |
| ☒ 2-23. よく使うデバッグ機能.....                                         | 19 |
| ☒ 2-24. よく使うプロジェクト設定.....                                       | 19 |
| ☒ 2-25. MSPM0 SDK のダウンロード.....                                  | 20 |
| ☒ 2-26. MSPM0 SDK のインストール.....                                  | 20 |
| ☒ 2-27. MSPM0 SDK フォルダ.....                                     | 20 |
| ☒ 2-28. ドキュメントの概要.....                                          | 21 |
| ☒ 2-29. CCS プロジェクトのインポート.....                                   | 22 |
| ☒ 2-30. SDK からプログラムを選択します.....                                  | 22 |
| ☒ 2-31. 重複したプロジェクトの削除.....                                      | 23 |
| ☒ 2-32. プロジェクトと README.md.....                                  | 23 |
| ☒ 2-33. CCS プロジェクトの概要.....                                      | 24 |
| ☒ 2-34. Smart Configuration と SysconfigSysConfig の MCU ビュー..... | 24 |
| ☒ 2-35. 関連ファイルの追加.....                                          | 25 |
| ☒ 2-36. オプションセットを含める.....                                       | 25 |
| ☒ 2-37. ビルド成功.....                                              | 25 |
| ☒ 2-38. Ultra Librarian ツールへの入口.....                            | 26 |
| ☒ 2-39. MSPM0 最小システム.....                                       | 26 |
| ☒ 2-40. MSPM0 最小システムの注意事項.....                                  | 26 |
| ☒ 2-41. プログラム ファイルの作成.....                                      | 27 |
| ☒ 2-42. プログラムソフトウェアおよびツール.....                                  | 27 |
| ☒ 2-43. コード サンプル ファイル.....                                      | 28 |
| ☒ 2-44. SysConfig 内の PWM 構成.....                                | 29 |
| ☒ 2-45. 各項目の詳細情報の取得.....                                        | 29 |
| ☒ 2-46. ピン構成.....                                               | 30 |
| ☒ 2-47. SysConfig ファイルの更新.....                                  | 30 |
| ☒ 2-48. ハードウェア設定.....                                           | 31 |
| ☒ 2-49. ブレークポイントソリューションの追加.....                                 | 31 |
| ☒ 2-50. Ultra Librarian ツールのダウンロード.....                         | 32 |
| ☒ 2-51. Altium Designer スクリプトを実行.....                           | 33 |
| ☒ 2-52. PCB ライブラリと回路図ファイル.....                                  | 33 |
| ☒ 2-53. ライブラリのインポート.....                                        | 34 |
| ☒ 3-1. MSP リセット機能.....                                          | 39 |
| ☒ 3-2. RL78 の内部マスカブル割り込み階層.....                                 | 44 |
| ☒ 3-3. MSPM0 のペリフェラル割り込み階層.....                                 | 45 |
| ☒ 3-4. 汎用イベント ルート.....                                          | 45 |
| ☒ 3-5. イベント管理の登録関係.....                                         | 46 |
| ☒ 3-6. MSPM0 イベントおよび割り込み処理.....                                 | 46 |
| ☒ 3-7. RL78 ELC (イベントリンクコントローラ).....                            | 47 |
| ☒ 3-8. RL78 イベントおよび割り込み処理.....                                  | 47 |

## 表の一覧

|                                                             |    |
|-------------------------------------------------------------|----|
| 表 1-1. テキサス・インスツルメンツ MSPM0Gx/Lx/Cx と ルネサス RL78 シリーズの比較..... | 4  |
| 表 2-1. エコシステムの比較.....                                       | 6  |
| 表 2-2. ソフトウェア エコシステム.....                                   | 8  |
| 表 2-3. CCS と e <sup>2</sup> Studio の比較.....                 | 8  |
| 表 2-4. MSPM0 でサポートされている IDE の概要.....                        | 9  |
| 表 2-5. MSPM0 デバッガの比較.....                                   | 11 |
| 表 2-6. MSPM0 のカバレッジ例.....                                   | 21 |
| 表 2-7. 空のプロジェクトの説明.....                                     | 21 |
| 表 3-1. CPU 機能セットの比較.....                                    | 35 |
| 表 3-2. フラッシュ機能の比較.....                                      | 35 |
| 表 3-3. フラッシュ・メモリ領域の比較.....                                  | 36 |
| 表 3-4. SRAM 機能の比較.....                                      | 37 |
| 表 3-5. 電源投入の概要と比較.....                                      | 38 |
| 表 3-6. 発振器の比較.....                                          | 40 |
| 表 3-7. MSPM0 マイコンの発振器.....                                  | 40 |
| 表 3-8. クロック信号の比較.....                                       | 41 |
| 表 3-9. ペリフェラル クロック ソース.....                                 | 41 |
| 表 3-10. RL78 デバイスと MSPM0 デバイスの動作モードの比較.....                 | 42 |
| 表 3-11. 割り込みの比較.....                                        | 44 |
| 表 3-12. イベント管理の比較.....                                      | 48 |
| 表 3-13. プログラミングモードの比較.....                                  | 48 |
| 表 4-1. GPIO 機能の比較.....                                      | 49 |
| 表 4-2. UART 機能の比較.....                                      | 51 |
| 表 4-3. SPI 機能の比較.....                                       | 52 |
| 表 4-4. I2C 機能の比較.....                                       | 52 |
| 表 4-5. タイマの命名.....                                          | 53 |
| 表 4-6. タイマ機能の比較.....                                        | 53 |
| 表 4-7. タイマ モジュールの代替品.....                                   | 53 |
| 表 4-8. タイマの使用事例の比較.....                                     | 53 |
| 表 4-9. WWDTC の命名法.....                                      | 54 |
| 表 4-10. WDT 機能の比較.....                                      | 54 |
| 表 4-11. RTC 機能の比較.....                                      | 54 |
| 表 5-1. 機能セットの比較.....                                        | 55 |
| 表 5-2. 変換モード.....                                           | 55 |
| 表 5-3. COMP 機能セットの比較.....                                   | 56 |
| 表 5-4. DAC 機能セットの比較.....                                    | 57 |
| 表 5-5. OPA 機能セットの比較.....                                    | 58 |
| 表 5-6. VREF 機能セットの比較.....                                   | 58 |

## 商標

LaunchPad™, EnergyTrace™, and ブースターパック™ are trademarks of Texas Instruments.

ARM® and Cortex® are registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

すべての商標は、それぞれの所有者に帰属します。

# 1 MSPM0 製品ラインアップの概要

## 1.1 はじめに

MSPM0 マイクロコントローラ (MCU) 製品は、強化された ARM® Cortex®-M0+ 32 ビット コア プラットフォームをベースとする MSP の高集積超低消費電力 32 ビット MCU ファミリの一部です。コスト最適化されたこれらの MCU は、高性能アナログ ペリフェラルの統合、拡張温度範囲のサポート、および小型フットプリント パッケージを実現します。テキサス・インスツルメンツの MSPM0 ファミリの低消費電力 MCU は、アナログとデジタルの統合度が異なるデバイスで構成されており、エンジニアはプロジェクトの要件に合致する MCU を見つけることができます。MSPM0 MCU ファミリーは、Arm Cortex-M0+ プラットフォームと超低消費電力のシステム アーキテクチャを組み合わせたもので、システム設計者は性能向上と消費電力低減を同時に実現できます。

MSPM0 MCU は、ルネサス RL78 に代わる競争力のある選択肢を提供します。このアプリケーション ノートは、デバイスの機能とエコシステムを比較することで、RL78 MCU から MSPM0 MCU への移行を支援します。

## 1.2 ルネサス RL78 MCU と MSPM0 MCU の製品ラインアップの比較

表 1-1. テキサス・インスツルメンツ MSPM0Gx/Lx/Cx と ルネサス RL78 シリーズの比較

|              |        | ルネサス<br>RL78 G シ<br>リーズ                | ルネサス<br>RL78 L シ<br>リーズ         | ルネサス<br>RL78 I シ<br>リーズ                          | ルネサス<br>RL78 F シ<br>リーズ                           | TI MSPM0<br>Gx シリーズ       | TI MSPM0<br>Lx シリーズ          | TI MSPM0<br>Cx シリーズ                              | TI MSPM0<br>Hx シリーズ                              |
|--------------|--------|----------------------------------------|---------------------------------|--------------------------------------------------|---------------------------------------------------|---------------------------|------------------------------|--------------------------------------------------|--------------------------------------------------|
| コア           |        | RL78 CPU コア                            |                                 |                                                  |                                                   | ARM Cortex-M0+            |                              |                                                  |                                                  |
| 周波数          |        | 16/20/24/32<br>MHz                     | 24MHz                           | 24 ~<br>32MHz                                    | 24/32/40V                                         | 80MHz                     | 32MHz                        | 24MHz、<br>32MHz <sup>(1)</sup>                   | 32MHz                                            |
| 電源電圧         |        | 1.6/1.8/2.7<br>~ 5.5 V,<br>1.6 ~ 3.6 V | 1.6-5.5 V<br>1.8 ~<br>3.6/5.5 V | 1.7/1.9/2.4/2<br>.7 ~ 5.5 V,<br>1.6 ~ 3.6 V      | 2.7-5.5 V, 1.8<br>~ 5.5 V                         | 1.62~3.6V                 | 1.62~3.6V                    | 1.62~3.6V                                        | 4.5~5.5 V <sup>(2)</sup>                         |
| 温度           |        | -40 ~<br>125°C,<br>-25 ~ 75°C          | -40 ~ 85°C,<br>-40 ~<br>125°C   | -40 ~ 85°C,<br>-40 ~<br>105°C,<br>-40 ~<br>125°C | -40 ~<br>105°C, -40<br>~ 125°C,<br>-40 ~<br>150°C | -40~125 °C                | -40~125 °C                   | -40~125°C                                        | -40~125°C                                        |
| メモリ          |        | 768KB~<br>1KB                          | 256KB~<br>8KB                   | 512KB~<br>8KB                                    | 512KB~<br>8KB                                     | 128KB~<br>32KB            | 64KB~8KB                     | 16KB~8KB                                         | 64KB~<br>32KB                                    |
| RAM          |        | 最大 144 KB                              | 最大 32 KB                        | 最大 32 KB                                         | 最大 4 KB                                           | 最大 32 KB                  | 最大 4 KB                      | 最大 8 KB                                          | 最大 8 KB                                          |
| GPIO (最大)    |        | 130                                    | 79                              | 76                                               | 130                                               | 60                        | 28                           | 45                                               | 45                                               |
| アナログ         | ADC    | 最大 12 ビット<br>x 28 チャン<br>ネル            | 最大 12 ビット<br>x 14 チャン<br>ネル     | 最大 12 ビット<br>x 17 チャン<br>ネル                      | 最大 12 ビット<br>x 25 チャン<br>ネル                       | 2 個の 4<br>Msps 12 ビッ<br>ト | 1 個の<br>1Msps 12 ビ<br>ット ADC | あり<br>(C1103/4 は<br>サポートなし)                      | あり                                               |
|              | DAC    | 最大 10 ビット<br>x 2 チャン<br>ネル             | 最大 12 ビット<br>x 2 チャン<br>ネル      | 12 ビット x 1<br>チャンネル<br>(RL/I1E)                  | 最大 8 ビット<br>x 1 チャン<br>ネル                         | 12 ビット                    | 8 ビット                        | 1x 1.5Msps<br>12 ビット<br>ADC (最大<br>27 チャン<br>ネル) | 1x 1.5Msps<br>12 ビット<br>ADC (最大<br>27 チャン<br>ネル) |
|              | コンパレータ | 最大 2 チャ<br>ネル                          | 最大 2 チャ<br>ネル                   | 最大 2 チャ<br>ネル                                    | 最大 1 チャ<br>ネル                                     | 3x 高速                     | 1x 高速                        | 8 ビット<br>(C1103/4 は<br>サポートなし)                   | なし                                               |
| 通信 (最大<br>数) | UART   | 4                                      | 4                               | 4                                                | 5                                                 | 4                         | 2                            | 1 x 高速<br>(C1103/4 は<br>サポートなし)                  | なし                                               |
|              | I2C    | 10                                     | 5                               | 4                                                | 5                                                 | 2 (高速)                    | 2 (高速)                       | 2x 4 Mbit/s                                      | 3x 4 Mbit/s                                      |
|              | SPI    | 0                                      | 4                               | 1                                                | 4                                                 | 2                         | 1                            | 1、最大 1<br>Mbit/s                                 | 2、最大 1<br>Mbit/s                                 |
|              | CAN    | 0                                      | 0                               | 0                                                | 2                                                 | 1 x (CAN-<br>FD)          | 0                            | 12 Mbit/s                                        | 16 Mbit/s                                        |
|              | LIN    | 1 (UART サポート)                          |                                 |                                                  | 3 (UART サ<br>ポート)                                 | 1 (UART サポート)             |                              |                                                  |                                                  |

表 1-1. テキサス・インスツルメンツ MSPM0Gx/Lx/Cx と ルネサス RL78 シリーズの比較 (続き)

|                      | ルネサス<br>RL78 G シリ<br>ーズ                                                             | ルネサス<br>RL78 L シリ<br>ーズ                                      | ルネサス<br>RL78 I シリ<br>ーズ                                                   | ルネサス<br>RL78 F シリ<br>ーズ      | TI MSPM0<br>Gx シリーズ                                                                  | TI MSPM0<br>Lx シリーズ                 | TI MSPM0<br>Cx シリーズ                                                                                                         | TI MSPM0<br>Hx シリーズ                  |
|----------------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------|---------------------------------------------------------------------------|------------------------------|--------------------------------------------------------------------------------------|-------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|--------------------------------------|
| その他の主なペリフェラル/<br>特長  | 内部昇圧<br>LCD<br>USB (RL/<br>G1A)<br>1 チャンネル<br>PGA<br>Bluetooth<br>(RL/G1D)<br>1%発振器 | 内部昇圧<br>LCD<br>USB (RL78/<br>L1C)<br>3 チャンネル<br>AMP          | LCD<br>1 チャンネル<br>PGA<br>3 チャンネル<br>AMP<br>USB<br>VBAT、シグ<br>マ デルタ<br>AFE | MATHACL、<br>ASIL-B、<br>150°C | 2 x op amps<br>CAN-FD、<br>USB、<br>Fast4Msps、<br>Sim-Sam<br>ADC、算術<br>演算アクセラ<br>レーション | 2 x op amps<br>LCD<br>(L2228)       | 最小サイズの<br>QFN パッケ<br>ージ (2 x 2)<br>および BGA<br>パッケージ<br>(0.861 x<br>1.6)、<br>0.5/0.65mm<br>ピッチ パッケ<br>ージ、業界と<br>のピン互換<br>DMA | 5V 電源<br>DMA ウィンド<br>ウウォッチドッ<br>グタイマ |
| タイマ番号                | 1/2/4/5                                                                             | 1/2/3                                                        | 1/2/5                                                                     | 1/2                          | 4                                                                                    | 7                                   | 5 (C1103 お<br>よび C1104<br>は 3 つのタイ<br>マをサポート)                                                                               | 4                                    |
| ピン数                  | 16~128 ピ<br>ン                                                                       | 32~100 ピ<br>ン                                                | 20~100 ピ<br>ン                                                             | 20~144 ピ<br>ン                | 20~100 ピ<br>ン                                                                        | 16~80 ピン                            | 8~48 ピン                                                                                                                     | 20~48 ピン                             |
| セキュリティ               | CRC、<br>RNG、<br>AES ライブラ<br>リ、<br>SHA ハッシ<br>ュ関数ライブ<br>ラリ、<br>RSA ライブラ<br>リ         | CRC、AES<br>GCM                                               | CRC                                                                       | CRC                          | CRC、<br>TRNG、<br>AES256                                                              | CRC                                 | アクティブ:<br>100µA/MHz<br>スタンバイ:<br>5µA                                                                                        | 該当なし                                 |
| 低消費電力 <sup>(3)</sup> | アクティブ:<br>Low から<br>37.5µA/MHz<br>まで停止:<br>Low から<br>0.2µA                          | アクティブ:<br>Low から<br>66µA/MHz<br>まで停止:<br>Low から<br>0.23µA まで | アクティブ:<br>Low から<br>96µA/MHz<br>まで停止:<br>Low から<br>0.23µA まで              | (言及してい<br>ません)               | アクティブ:<br>85µA/MHz<br>スタンバイ:<br>1.5µA                                                | アクティブ:<br>71µA/MHz<br>スタンバイ:<br>1µA | 最小サイズの<br>QFN パッケ<br>ージ (2 x 2)<br>および BGA<br>パッケージ<br>(0.861 x<br>1.6)、<br>0.5/0.65mm<br>ピッチ パッケ<br>ージ、業界と<br>のピン互換<br>DMA | 5V 電源<br>DMA ウィンド<br>ウウォッチドッ<br>グタイマ |

- (1) MSPM0C1103 および MSPM0C1104:24MHz、MSPM0C1105 および MSPM0C1106:32MHz  
(2) H シリーズの最初のデバイスの対応電圧は 4.5~5.5V で、今後より幅広い電源電圧にも対応する予定です。  
(3) RL78 のストップモードは、MSPM0 シャットダウンモードと同様です (CPU、クロック、ペリフェラルはシャットダウン)。

## 2 エコシステムと移行

MSPM0 MCU は、ハードウェアおよびソフトウェアの大規模なエコシステムによってサポートされており、リファレンス デザインやコード サンプルを利用して設計をすぐに開始できます。MSPM0 MCU は、オンライン リソース、MSP Academy を使用したトレーニング、[TI E2E™ サポート フォーラム](#)によるオンライン サポートによってもサポートされています。

### 2.1 エコシステムの比較

表 2-1. エコシステムの比較

| 機能                | RL78 デバイス                                              | MSPM0 デバイス                                                                                                              |
|-------------------|--------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|
| コード ソース           | ミドルウェア<br>ルネサス フラッシュドライバとセルフプログラミングライブラリ<br>ドライバ<br>OS | MSPM0-SDK (DriverLib、ミドルウェア、RTOS、サンプル コード)                                                                              |
| IDE               | E <sup>2</sup> スタジオ<br>CS +<br>CC-RL<br>IAR            | CCS<br>IAR<br>Keil                                                                                                      |
| ソフトウェアの構成         | スマートコンフィギュレータ                                          | SysConfig                                                                                                               |
| フラッシュ プログラミング ツール | ルネサス フラッシュ プログラマ                                       | UniFlash                                                                                                                |
| プログラマ             | PG-FP6                                                 | MSP-GANGC-GANG                                                                                                          |
| デバッグ              | E2 エミュレータ<br>E2 エミュレータ Lite<br>EZ-CUBE                 | XDS110<br>J-LINK                                                                                                        |
| ハードウェア            | 高速なプロトタイピング ボード<br>ターゲット ボード<br>スタータ キット               | LP-MSPM0G3507 LaunchPad、LP-MSPM0L1306 LaunchPad、LP-MSPM0C1104 LaunchPad、LP-MSPM0C1106 LaunchPad、LP-MSPM0H3216 LaunchPad |

MSPM0 エコシステムの概要を、[図 2-1](#) に示します。

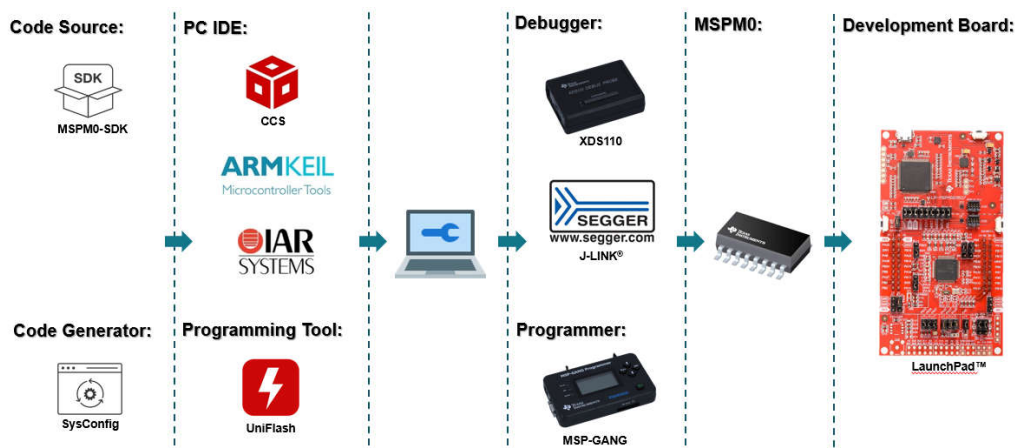


図 2-1. MSPM0 エコシステムの概要

#### 2.1.1 MSPM0 ソフトウェア開発キット (MSPM0 SDK)

MSPM0 SDK には豊富なサンプル コードが含まれており、エンジニアがテキサス・インスツルメンツの MSPM0+ マイコンでアプリケーション開発を行いやすくなっています。各機能分野とすべてのサポート デバイスの使用法を解説しており、設計を開始する際の出発点になります。[図 2-2](#) に、MSPM0 SDK の構造を示します。

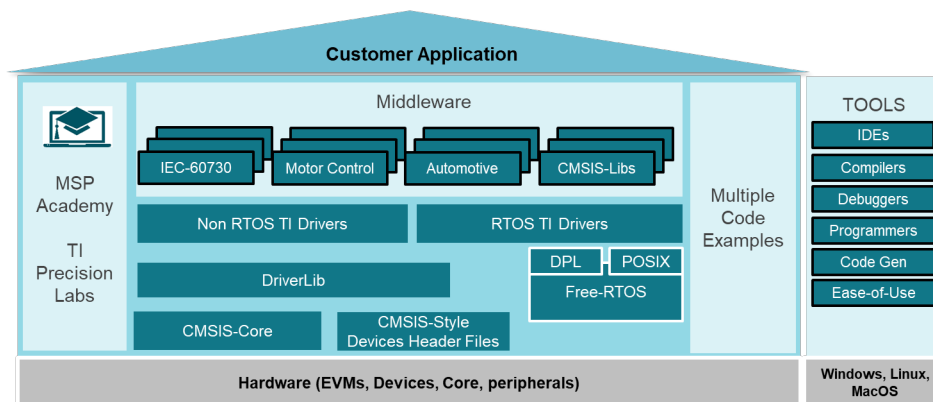


図 2-2. MSPM0 SDK の構造

MSPM0 SDK は、[MSPM0-SDK サポートソフトウェア | TI.com](#) からダウンロードできます。MSPM0 SDK には以下の 4 つのフォルダが含まれています:

**例:** 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™ 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。

**Docs:** ユーザーガイドや API ガイドなど、関連するすべての資料が付属しています。

**出典:** すべてのドライバとミドルウェアを対象とするソースコードとライブラリ。

**Tools:** MSPM0 アプリケーションの開発およびテストに役立つツールのセット。

ルネサス RL78 は、DSP や USB ドライバなどの大規模なミドル/大規模なサンプルコードをサポートしていますが、コード開発プログラム用のパッケージはなく、RL78 にはサンプルコードがないため、ユーザーは新しいプロジェクトを作成し、デバッグなどの設定を一から行う必要があります。一方、MSPM0 SDK はすべてのソースコードをミドルウェアとドライバライブラリと統合し、開発を容易にしています。このサンプルコードは、顧客が迅速に開発を開始し、マイコン パリフェラルの詳細を確認するのに役立ちます。

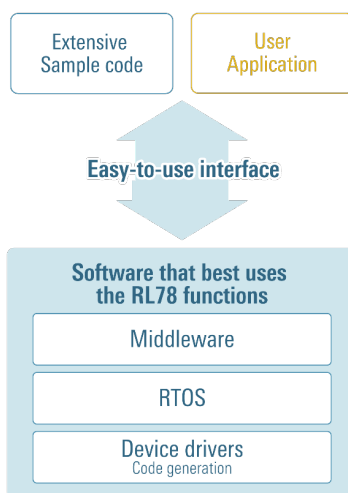


図 2-3. RL78 ソフトウェア開発環境

**表 2-2. ソフトウェア エコシステム**

| 機能          | RL78 ソフトウェア | MSPM0 SDK |
|-------------|-------------|-----------|
| レジスタレベルのコード | いいえ         | いいえ       |
| ドライバライブラリ   | あり          | あり        |
| ミドルウェア      | あり          | あり        |
| セルフプログラミング  | あり          | いいえ       |
| すぐに使用できるコード | いいえ         | あり        |
| 無償の RTOS    | あり          | あり        |

ほとんどの MSPM0 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。

その他の参考ドキュメントを以下に示します：

- [MSPM0 SDK ユーザー ガイド](#)
- [MSPM0 ツール ガイド](#)
- [Driverlib API ガイド](#)

### 2.1.2 MSPM0 でサポートされている IDE

統合開発環境 (IDE) は、プログラマがソフトウェア・コードを効率的に開発するのに役立つソフトウェア・アプリケーションであり、通常はエディタ、コンパイラ、デバッガなどが含まれています。

RL78 の典型的な IDE は e<sup>2</sup>studio であり、サンプルコードをダウンロードでき、使いやすい Eclipse コードエディタがあります。TI では、TI のマイコン (MCU) と組み込みプロセッサのポートフォリオをサポートする Code Composer Studio IDE (CCS) を強くお勧めします。CCS は Eclipse ベースの IDE でもあるため、ユーザーは開発を容易に開始できます。特に、CCS は、最適化された C/C++ コンパイラ、ソースコード エディタ、プロジェクトのビルド環境、デバッガ、プロファイラ、その他の多くの機能を含む、組み込みアプリケーションの開発とデバッグに使用する一連のツールで構成されています。また、CCS は完全に無料で使用できます。

**表 2-3. CCS と e<sup>2</sup> Studio の比較**

| 複数の IDE              | CCS                                                        | e <sup>2</sup> スタジオ                         |
|----------------------|------------------------------------------------------------|---------------------------------------------|
| 使用許諾                 | 未使用                                                        | 未使用                                         |
| コンパイラ                | TI の Arm CLANG, GCC                                        | CC-RL, LLVM                                 |
| IDE に統合した消費電流        | EnergyTrace                                                | ルネサス QE                                     |
| ペリフェラルの API 機能に関する支援 | 非対応                                                        | 対応                                          |
| 表示言語                 | English (英語)                                               | 英語<br>日本語<br>中国語                            |
| ファイルを変換します           | 16 進ファイル<br>バイナリファイル<br>Motorola S レコードファイル<br>Ti_txt ファイル | 16 進ファイル<br>バイナリファイル<br>Motorola S レコードファイル |
| コード GUI を生成します       | SysConfig                                                  | スマート構成                                      |

CCS は、統合型の TI Resource Explorer に、MSPM0 デバイス構成と SysConfig からの自動コード生成、MSPM0 サンプルコードとアカデミートレーニングを統合しています。さらに、CCS は、一体型の開発ツールを提供します。

CCS に加えて、MSPM0 デバイスは以下の表 表 2-4 に示す業界標準 IDE でもサポートされています。

- [Code Composer Studio](#)
- [IAR](#)
- [Keil](#)

表 2-4. MSPM0 でサポートされている IDE の概要

| 複数の IDE     | CCS (Eclipse)    | IAR                   | Keil                   |
|-------------|------------------|-----------------------|------------------------|
| 使用許諾        | 未使用              | 有料                    | 有料                     |
| コンパイラ       | TI Arm Clang GCC | IAR C/C++コンパイラ™、Arm 用 | ARM コンパイラ バージョン 6      |
| ディスクサイズ     | 0.88G(ccs20.1.1) | 6.33G (アーム 8.50.4)    | 2.5G (µVision V5.37.0) |
| XDS110      | 対応               | 対応                    | 対応                     |
| J-Link      | 対応               | 対応                    | 対応                     |
| EnergyTrace | 対応               | いいえ                   | いいえ                    |
| MISRA-C     | いいえ              | 対応                    | いいえ                    |
| セキュリティ      | いいえ              | 対応                    | いいえ                    |
| ULINKplus   | いいえ              | いいえ                   | 対応                     |
| 機能安全        | いいえ              | 対応                    | 対応                     |

CCS の使用といくつかの機能を [セクション 2.2.2.2](#) に示します。その他の参考資料は次のとおりです：

- [CCS クイック スタート ガイド](#)
- [CCS](#)
- [CCS トレーニング ビデオ](#)
- [CCS ユーザーズ ガイド](#)
- [IAR | クイック スタート ガイド](#)
- [IAR トレーニング ビデオ](#)
- [IAR ユーザーズ ガイド](#)
- [Keil クイック スタート ガイド](#)
- [Keil トレーニング ビデオ](#)
- [Keil スタート ガイド](#)

### 2.1.3 SysConfig

Smart Configuration (スマートコンフィグレーション) と同様に、**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。**SysConfig** は **CCS** に統合されていますが、**Keil** および **IAR** でも使用できます。**SysConfig** は、[SysConfig IDE](#)、[構成](#)、[コンパイラ](#)、[デバッグ](#) | [TI.com](#) からダウンロードできます。

それから、**SysConfig** は、IDE なしで実行できます。スタンドアロン バージョンは、コード生成やデバイスの機能評価には使用できますが、サンプルを実行することはできません。

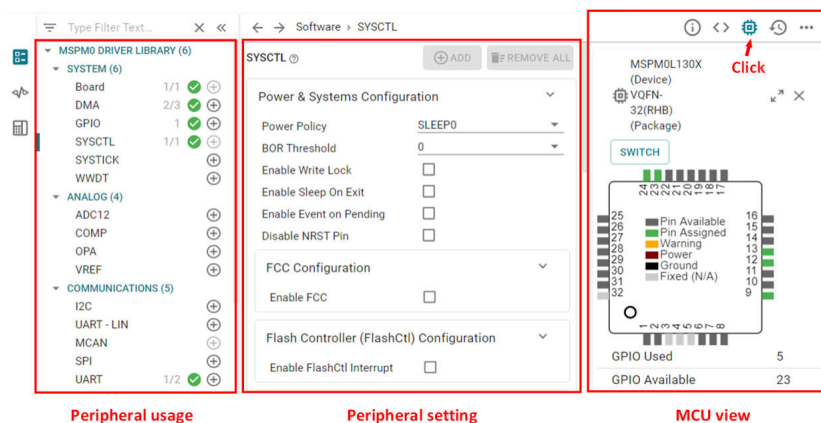


図 2-4. MSPM0 SysConfig

詳細については、『[MSPM0 SysConfig ガイド](#)』を参照してください。

スマートコンフィグレーションと比較して、SysConfig には次の利点があります。

- **SysConfig** 内のペリフェラルの分類はより明確で、全体的なインターフェイスは高い読みやすさを備えており、[図 2-5](#) に示すように人間とコンピュータのインタラクションインターフェイスは優れています。
- **SysConfig** には、ハードウェア回路図表示、各ペリフェラルの詳細な説明と、GUI インターフェイスのすべての構成に関する詳細な機能説明があります。
- スマートコンフィグレーションの割り込みとピン設定は別のモジュールになっていますが、散乱しており、開発にはつながりません。テキサス・インスツルメンツの **SysConfig** は、特定のペリフェラル モジュールに直接構成できるため、開発を簡単かつ明確にすることができます。割り込み設定の比較を [図 2-6](#) に示します。**SysConfig** は、ADC モジュールのイベント構成など、複数のペリフェラルリンク構成を実現できます。
- ユーザーがペリフェラルの特定の機能を構成するとき、**SysConfig** はコードの変更を **REAL-TIME** で表示でき、スマート構成ではすぐに確認できません。

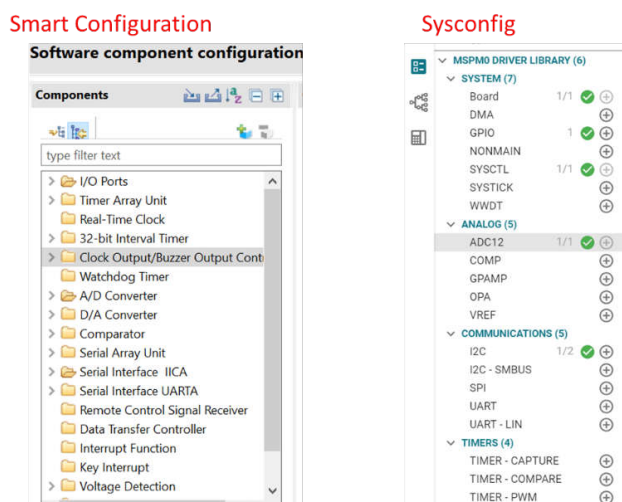


図 2-5. ペリフェラルリストの比較

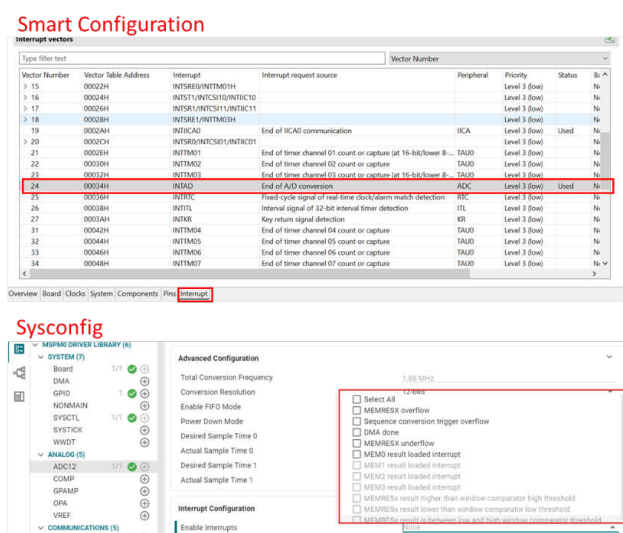


図 2-6. 割り込み設定の比較

## 2.1.4 デバッグ ツール

RL78 の場合は、TOOL0 ピンを使用して、デバッガ (E2 エミュレータや E2 エミュレータ Lite など) と RL78 または F23 の間をインターフェイスし、専用のシングルライン UART で操作します。RL78 の一般的なデバッグツールは E2 エミュレータであり、消費電流の測定、監視ポイント、外部トリガ入出力の設定をサポートしています。

MSPM0 の場合、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。デバッガの接続の詳細については、[図 2-7](#) を参照してください。

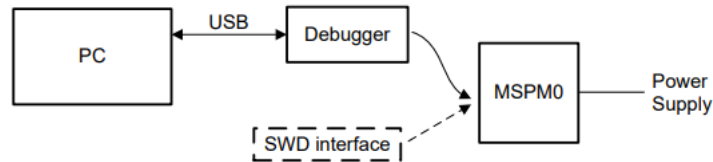


図 2-7. MSPM0 のデバッグ

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 デバッガ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 Debug Probe](#)」(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。記録的なブレークダウンを実現するフラッシュ ロード、最大 3MiB/s の RAM ダウンロード速度、MCU のフラッシュ メモリ内に無制限のブレークポイントを設定する機能を活用できます。また、J-Link は CortexM0+ 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、[J-Link デバッグ プローブのページ](#)を参照してください。

ここで、MSPM0 をサポートする XDS110 と J-LINK デバッガの機能の違いの概要を示します。

表 2-5. MSPM0 デバッガの比較

| 特長                           | XDS110      | XDS110 OB <sup>(1)</sup> | J-Link                 |
|------------------------------|-------------|--------------------------|------------------------|
| cJTAG (SBW)                  | ✓           | ✓                        | ✓                      |
| BSL <sup>(2)</sup> ツール       | ✓           | ✓                        |                        |
| バックチャネル UART                 | ✓           | ✓                        | 2.5G (μVision V5.37.0) |
| 電源                           | 1.8 ~ 3.6 V | 3.3/5V                   | 5V                     |
| IDE <sup>(3)</sup> : CCS     | ✓           | ✓                        | ✓                      |
| IDE: サード パーティ <sup>(4)</sup> | IAR, Keil   | IAR, Keil                | IAR, Keil              |

- (1) XDS110 OB は、オンボードの XDS110 を意味します。
- (2) BSL はブートストラップ ロードを意味します。
- (3) IDE は、統合開発環境のことです。
- (4) サード パーティには IAR と Keil が含まれています。

## 2.1.5 LaunchPad™

スタータキット、FPB、ターゲットボードを備えた RL78 とは異なり、LaunchPad 開発キットは唯一の MSPM0 用評価基板です。

LaunchPad キットは使いやすい評価基板で、MSPM0 の開発を開始するために必要なものがすべて含まれています。これには、EnergyTrace™ テクノロジーを使用したプログラミング、デバッグ、消費電力測定用のオンボード デバッグ プローブが含まれています。MSPM0 LaunchPad には、オンボード ボタン、LED、温度センサなどの回路も搭載されています。

さまざまなブースタパック プラグイン モジュールをサポートする 40 ピンのブースターパック™ プラグイン モジュール ヘッダーにより、迅速で簡単なプロトタイプ製作が可能になります。ワイヤレス接続、グラフィカル ディスプレイ、環境センシングなどの機能も迅速に追加できます。

- [LP-MSPM0G3507 LaunchPad 開発キット](#)
- [LP-MSPM0L1306 LaunchPad 開発キット](#)
- [LP-MSPM0C1104 LaunchPad 開発キット](#)

図 2-8 に LaunchPad (MCU と XDS110 デバッガを含む) の概要を示します。ジャンパーを取り外せば、J-Link など他のデバッガを使用して MCU のデバッグを行うこともできます。

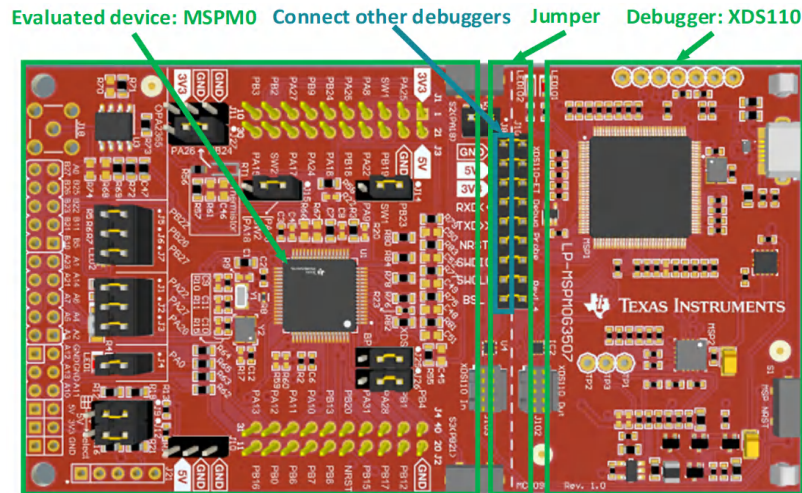


図 2-8. MSPM0G3507 Launchpad の概要

ジャンパ絶縁端子台には、電源 (GND、5V、3.3V)、UART (RXD、TXD)、リセットピン、ARM デバッグチャネル (SWDIO、SWCLK)、BSL が含まれています。

ジャンパキャップに加えて、LaunchPad の右側にある標準的な Arm Cortex 10 ピンコネクタ (図 2-9 を参照) を使用して書き込みを行うこともできます。Cortex デバッグ コネクタは、JTAG デバッグ、シリアル ワイヤ デバッグ、シリアル ワイヤ ビューア (シリアル ワイヤ デバッグ モードを使用している場合は SWO 接続経由) 機能をサポートしています。

|           |   |    |              |
|-----------|---|----|--------------|
|           | 1 | 2  |              |
| VCC       | ○ | ○  | SWDIO / TMS  |
| GND       | ○ | ○  | SWDCLK / TCK |
| GND       | ○ | ○  | SWO / TDO    |
| KEY       | ○ | ○  | NC / TDI     |
| GNDDetect | ○ | ○  | nRESET       |
|           | 9 | 10 |              |

図 2-9. Arm Cortex 10 ピンの定義

図 2-10 に、MSPM0G3507 LaunchPad の機能の一部を示します。

LaunchPad の下部には BoosterPack 用のコネクタがあり、専用の機能モジュールを直接差し込んで、すばやくプロトタイプを構築できます。また、DuPont ワイヤを単独で引き出して素早く使用することも可能です。LaunchPad には両側にユーザー定義ボタン、温度センサー、光センサー、単色 LED、その下に RGB LED が搭載されています。

- [LP-MSPM0G3507 LaunchPad 開発キット](#)
- [LP-MSPM0L1306 LaunchPad 開発キット](#)

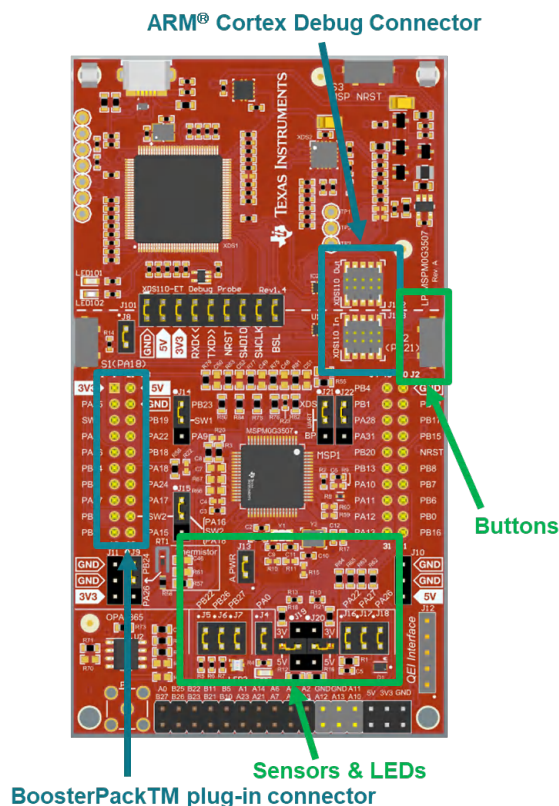


図 2-10. MSPM0G3507 Launchpad の特徴と機能

## 2.2 移行プロセス

MSPM0 へのスムーズな移行のために、図 2-11 に詳細な手順をフロー形式で示します。各ステップについて詳しく説明されており、具体例も続くセクションで紹介されています。

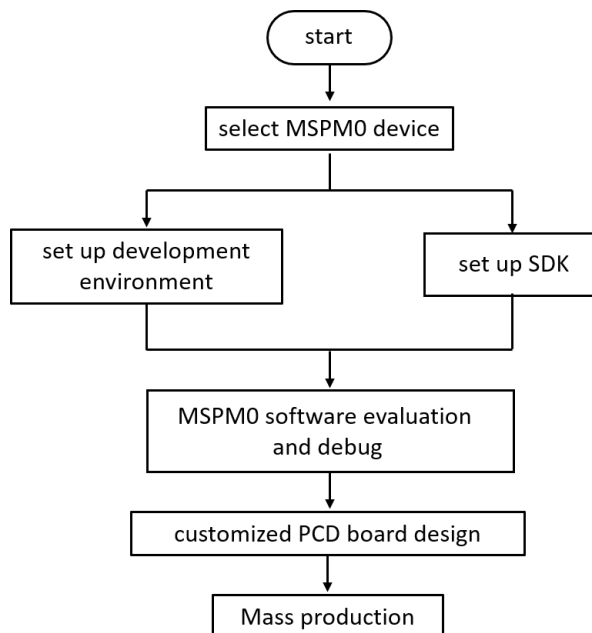


図 2-11. MSPM0 の移行フローチャート

### 2.2.1 ステップ1: 適切な MSPM0 MCU を選択する

移行の最初のステップは、アプリケーションに適した MSPM0 デバイスを選択することです。図 2-12 に MSPM0 C、L、G シリーズファミリのポートフォリオを示します。これらは、TI の公式 Web サイトで参照できます。どちらの製品ラインアップも、メモリ容量やパッケージに基づいてデバイスを分類しているため、目的に合ったツールを選びやすくなっています。

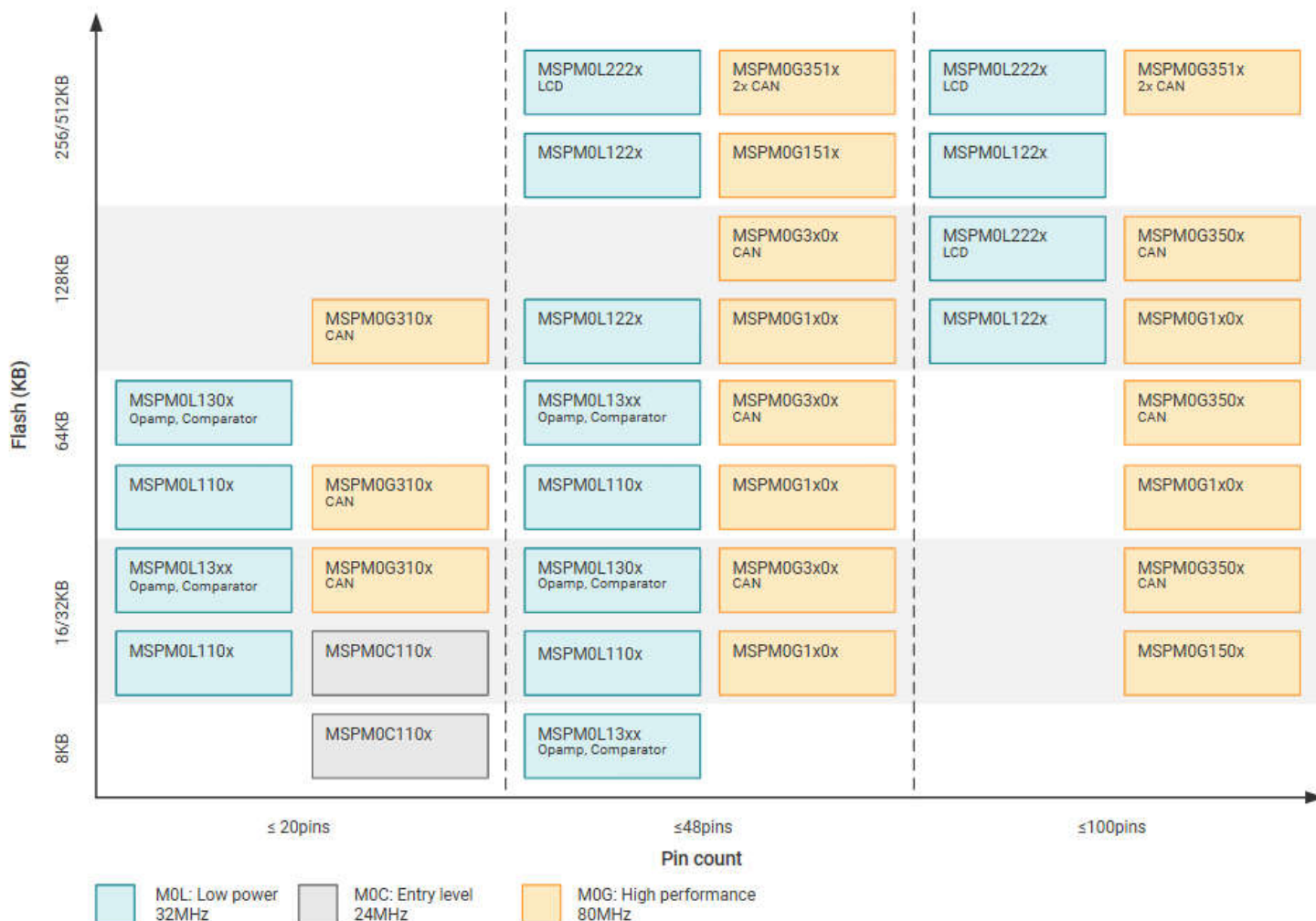


図 2-12. MSPM0C、MSPM0L、MSPM0G シリーズの製品ラインアップ

特定のデバイスを絞り込むには、製品選定ツールが重要な役割を果たします。このリンクでは、左のフィルタを使用して、マイコンの周辺機器の要件に基づいて初期のスクリーニングを実行できます。たとえば、UART の番号に一致する MCU を絞り込む場合、フィルタ ツールで条件を選択すると、右側に該当する MCU デバイスが表示されます (図 2-13 を参照)。また、左上の検索ボックスからデバイス名を入力することで、該当デバイスの詳細情報ページに直接アクセスすることもできます。

Hide filters Columns Reset table 23 of 23 total products Email Download Excel Log in to view inventory

| Product number                                                                                                | Images | Price/Quantity (USD) | Frequency (MHz) | Flash memory (kByte) | RAM (kByte) | Number of GPIOs | UART | Number of I2Cs | SPI |
|---------------------------------------------------------------------------------------------------------------|--------|----------------------|-----------------|----------------------|-------------|-----------------|------|----------------|-----|
| <input type="checkbox"/> <b>MSPM0L1345 – NEW</b><br>Data sheet: PDF   HTML<br><a href="#">View alternates</a> |        | US\$0.484   1ku      | 32              | 32                   | 4           | 22              | 2    | 2              | 1   |
| <input type="checkbox"/> <b>MSPM0L1346 – NEW</b><br>Data sheet: PDF   HTML<br><a href="#">View alternates</a> |        | US\$0.544   1ku      | 32              | 64                   | 4           | 22              | 2    | 2              | 1   |
| <input type="checkbox"/> <b>MSPM0L1343 – NEW</b><br>Data sheet: PDF   HTML<br><a href="#">View alternates</a> |        | US\$0.412   1ku      | 32              | 8                    | 2           | 15              | 2    | 2              | 1   |
| <input type="checkbox"/> <b>MSPM0L1344 – NEW</b><br>Data sheet: PDF   HTML<br><a href="#">View alternates</a> |        | US\$0.422   1ku      | 32              | 16                   | 2           | 15              | 2    | 2              | 1   |

図 2-13. MSPM0 製品選択ツール

デバイスページでは、データシート、テクニカルリファレンス マニュアル (TRM)、正誤表などの主要ドキュメントの検索、ダウンロードを容易に行うことができます (図 2-14 を参照)。デバイス固有のデータシートでは、対象の MSPM0 の仕様や機能に関する情報が紹介されています。デバイス固有の TRM では、MSPM0 シリーズの使用法や特徴が紹介されています。デバイス固有の正誤表では、MSPM0 関連のシリーズまたはバージョンの電子誤植の説明が紹介されています。

**NEW**  
**MSPM0L1345** ACTIVE

32-MHz Arm® Cortex®-M0+ MCU with 32-KB flash, 4-KB SRAM, 12-bit ADC, comparator, TIA

DATA SHEET [MSPM0L130x Mixed-Signal Microcontrollers datasheet \(Rev. C\)](#) PDF | HTML **datasheet**

USER GUIDES [MSPM0 L-Series 32-MHz Microcontrollers Technical Reference Manual \(Rev. C\)](#) **User Guides**

ERRATA [MSPM0L110x, MSPM0L13xx Microcontrollers Errata \(Rev. A\)](#) **errata**

図 2-14. MSPM0 の重要ドキュメント一覧

Web サイトには MSPM0 に関連する技術ドキュメントが掲載されており、なかでも代表的なものがアプリケーション ノートです。

## Technical documentation

★ = Top documentation for this product selected by TI

| Type             | Title                                                                      | Date ↓↑     |
|------------------|----------------------------------------------------------------------------|-------------|
| All              | Filter title by keyword                                                    |             |
| ★ Data sheet     | MSPM0L130x Mixed-Signal Microcontrollers datasheet (Rev. C) PDF   HTML     | 27 Jun 2023 |
| ★ Errata         | MSPM0L110x, MSPM0L13xx Microcontrollers Errata (Rev. A) PDF   HTML         | 28 Apr 2023 |
| ★ User guide     | MSPM0 L-Series 32-MHz Microcontrollers Technical Reference Manual (Rev. C) | 05 May 2023 |
| Application note | MSPM0 Bootloader (BSL) Host Implementation (Rev. A) PDF   HTML             | 06 Jul 2023 |
| Application note | EEPROM Emulation Type A Solution PDF   HTML                                | 18 Apr 2023 |
| Application note | STM32에서 Arm 기반 MSPM0으로의 마이그레이션 가이드 (Rev. A) PDF   HTML                     | 12 Apr 2023 |
| Application note | 從 STM32 到 Arm 架構的 MSPM0 移轉指南 (Rev. A) PDF   HTML                           | 12 Apr 2023 |
| Application note | EEPROM Emulation Type B Design PDF   HTML                                  | 11 Apr 2023 |

図 2-15. MSPM0 関連技術ドキュメント一覧

選択後、「注文と品質」セクションで価格やその他の情報を確認できます (図 2-16 を参照)。

MSPM0L1306 PREVIEW Data sheet Order now

Product details | Technical documentation | Design & development | **Ordering & quality** | Support & training

**Ordering & quality**

| Part number ↓↑              | Buy                                                                                  | TI.com inventory ↓↑ | Qty   Price (USD) ↓↑ | Package qty   Carrier ↓↑ |
|-----------------------------|--------------------------------------------------------------------------------------|---------------------|----------------------|--------------------------|
| XMSM0L1306SDGS20R<br>ACTIVE | <input type="text" value="Enter quantity"/><br><span>Add to cart</span><br>Limit: 5  | 93                  | 1ku   <span>▼</span> | 1   LARGE T&R            |
| XMSM0L1306SDGS28R<br>ACTIVE | <input type="text" value="Enter quantity"/><br><span>Add to cart</span><br>Limit: 10 | 170                 | 1ku   <span>▼</span> | 5,000   LARGE T&R        |

図 2-16. 注文と品質情報の表示画面

## 2.2.2 ステップ2.IDE の設定と CCS の簡単な説明

### 2.2.2.1 IDE の設定

TI の CCS が選択された IDE です。

1. ダウンロードリンクをクリックしてインストールを開始し、[次へ]を押し続けます。

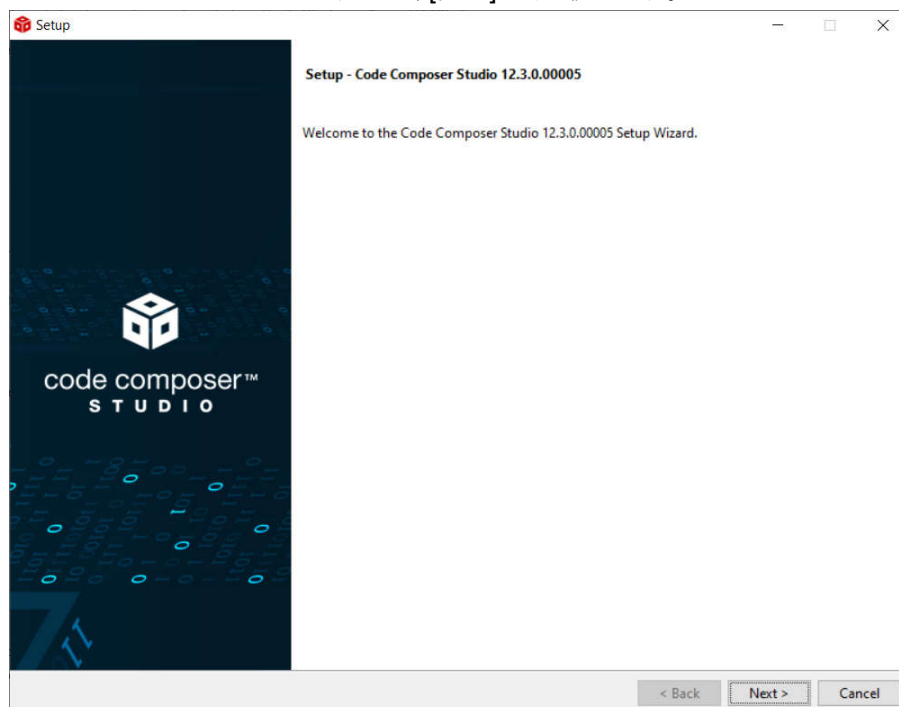


図 2-17. CCS のインストール

## 2. MSPM0 サポートコンポーネントを選択します。

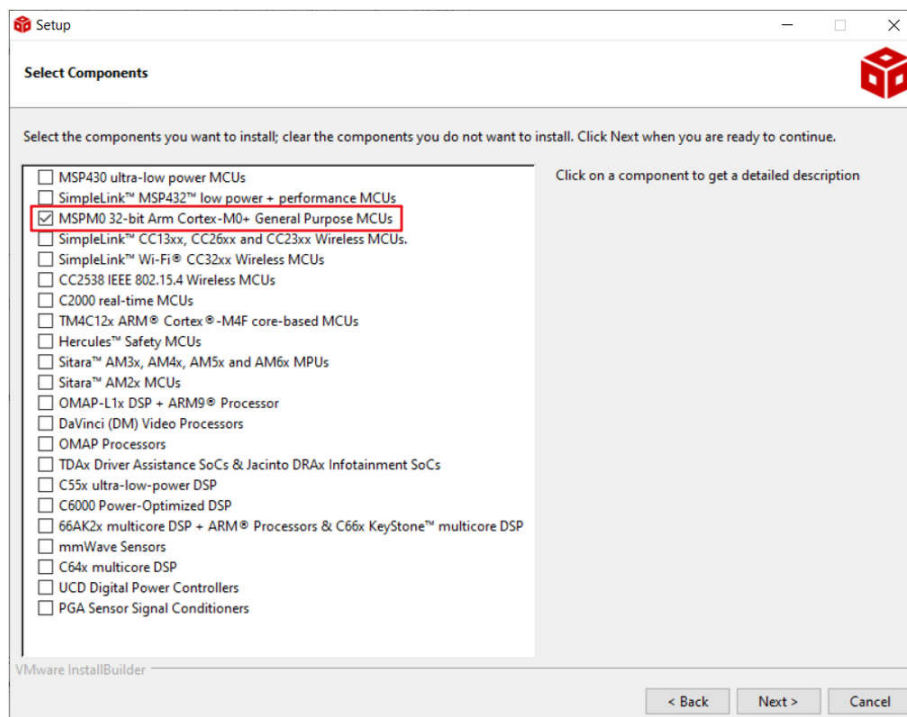


図 2-18. CCS のインストール - MSPM0 サポートの選択

## 3. 必要に応じて、J-link を選択します。

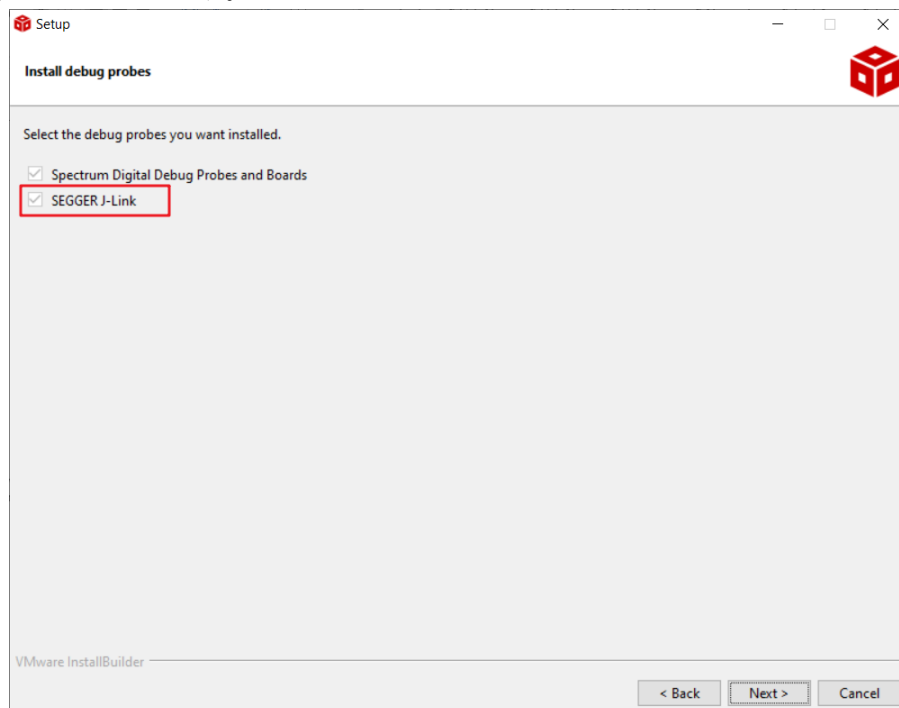


図 2-19. CCS のインストール - J-Link ダウンロードの選択

## 4. CCS のダウンロードを完了します。

### 2.2.2.2 CCS の簡単な説明

新しいワークスペースを起動します。ワークスペースとは、インポートしたプロジェクトをコピーするアドレスです。CCS のステップは、e<sup>2</sup>studio のステップと同じです。

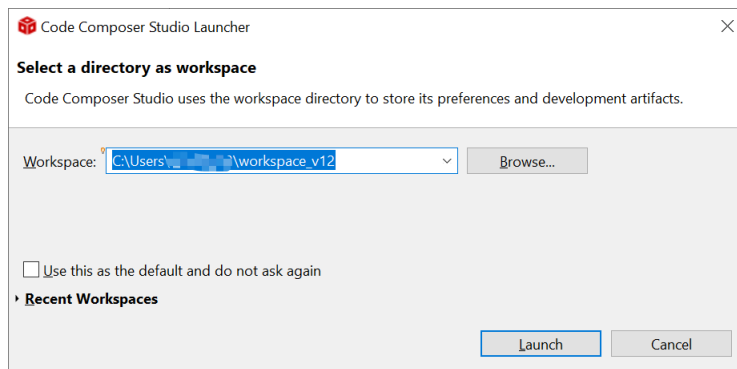


図 2-20. CCS Launch Workspace

新しいプロジェクトを作成する場合は、[ファイル] → [新規作成] → [CCS プロジェクト] に移動します。ここで実施すべき重要な作業が 2 つあります。1 つ目は MSPM0 デバイスの選択、もう 1 つは接続の選択です (図 2-21 を参照)。選択した後、プロジェクト名を追加して [完了] ボタンを押すと、プログラムの作成が可能になります。

これは、デバイス、ツール・チェーン、プログラム名を選択する必要がある e<sup>2</sup>studio で新しいプロジェクトを作成する場合と似ています。

TI では、CCS の使用方法を紹介する [セクション 2.2.4](#) から MSPM0 SDK のサンプルを見つけることを推奨します。

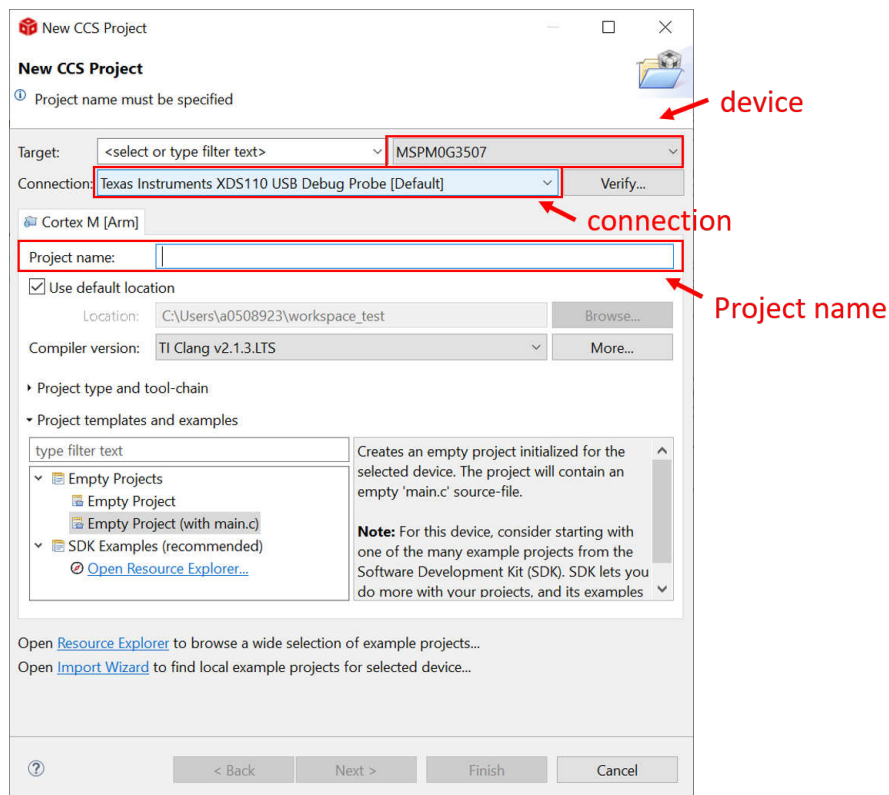


図 2-21. CCS で新しいプロジェクトを作成する

図 2-22 では、CCS 機能について簡単な説明をしています。

## ショートカットキーの機能:

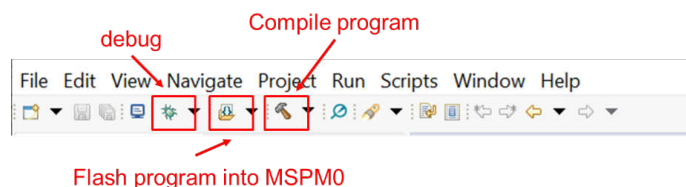


図 2-22. よく使う機能

## デバッグ機能:

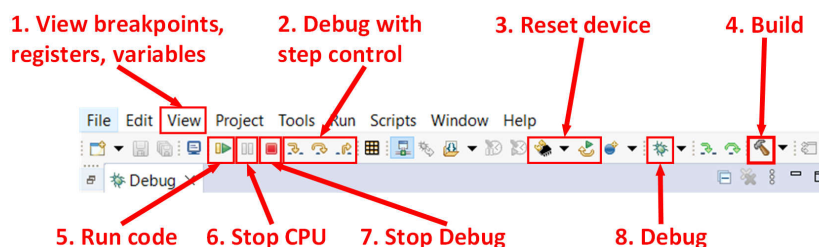


図 2-23. よく使うデバッグ機能

## プロジェクトプロパティでよく使う設定:

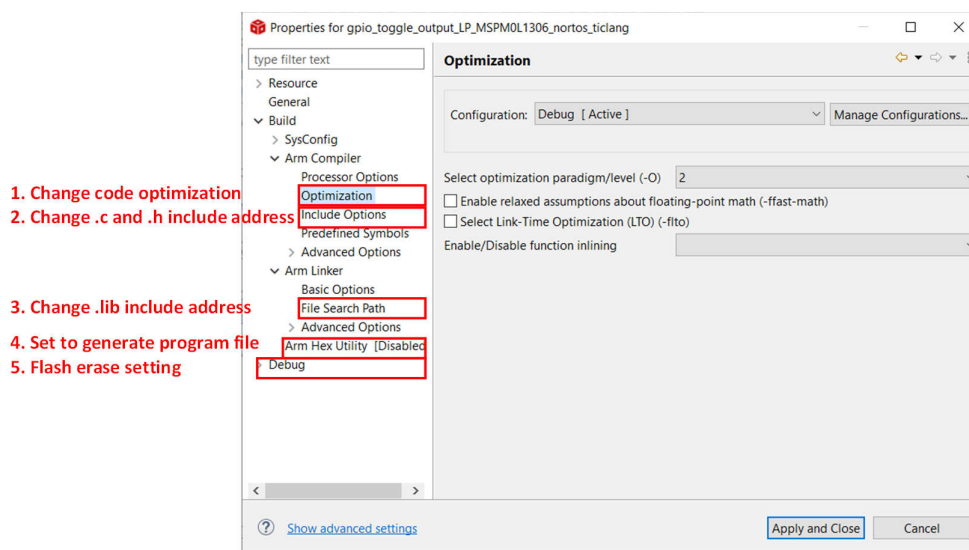


図 2-24. よく使うプロジェクト設定

詳細については、『[MSPM0 MCU 用 Code Composer Studio IDE バージョン 12.4+- MSPM0 MCU 用 Code Composer Studio IDE 1.0](#)』ドキュメントを参照してください。

### 2.2.3 ステップ3:MSPM0 SDK の設定とMSPM0 SDK の簡単な説明

RL78 については、ソフトウェアパッケージはなく、IDE *Developer Assistance* からサンプルコードを検索したり、Web サイトから特定のサンプルコードをダウンロードしたりすることのみが許可されています。TI は、開発を支援するソフトウェア開発キットを提供しています。

### 2.2.3.1 MSPM0 SDK の設定

1. リンクをクリックして、**MSPM0 SDK** をダウンロードします。



図 2-25. MSPM0 SDK のダウンロード

2. [次へ] を押して SDK をインストールします。

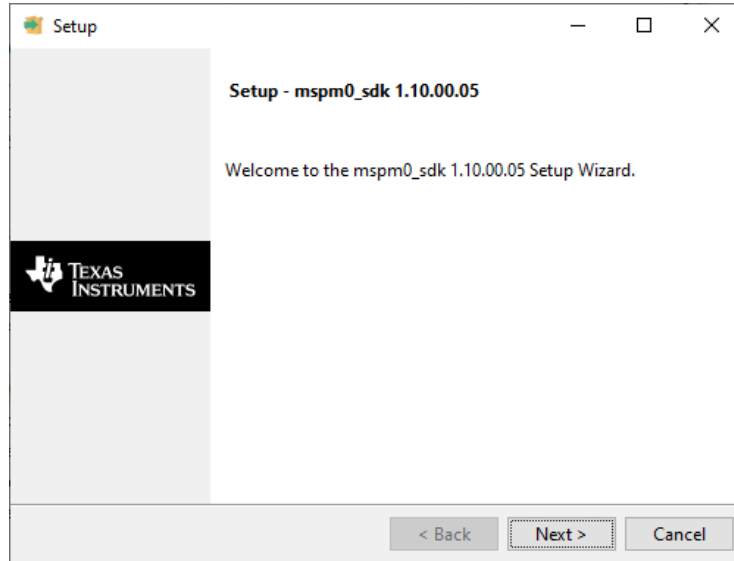


図 2-26. MSPM0 SDK のインストール

3. MSPM0 SDK のダウンロードを完了します。

### 2.2.3.2 SDK の簡単な説明

コンテンツのダウンロードが完了すると、ファイルの内容が SDK フォルダに表示されます。ファイルの場所は、デフォルトでは `c:\ti\mspm0_sdk_xxx` です (図 2-27 を参照)。この場所に含まれている、主に使用されるフォルダは、**sample**、**docs** フォルダです。

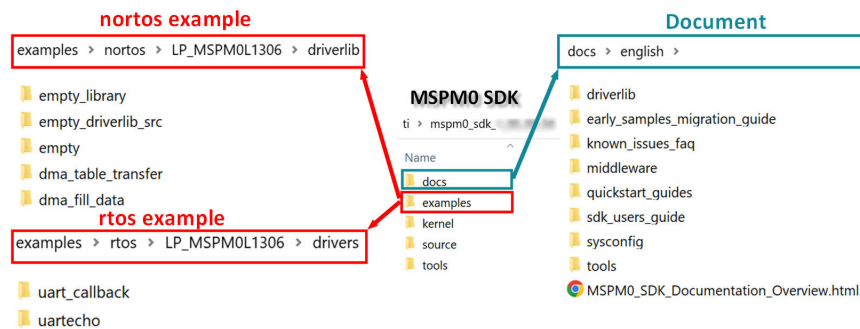


図 2-27. MSPM0 SDK フォルダ

表 2-6 に、サンプル カバレッジの概要を示します。

**表 2-6. MSPM0 のカバレッジ例**

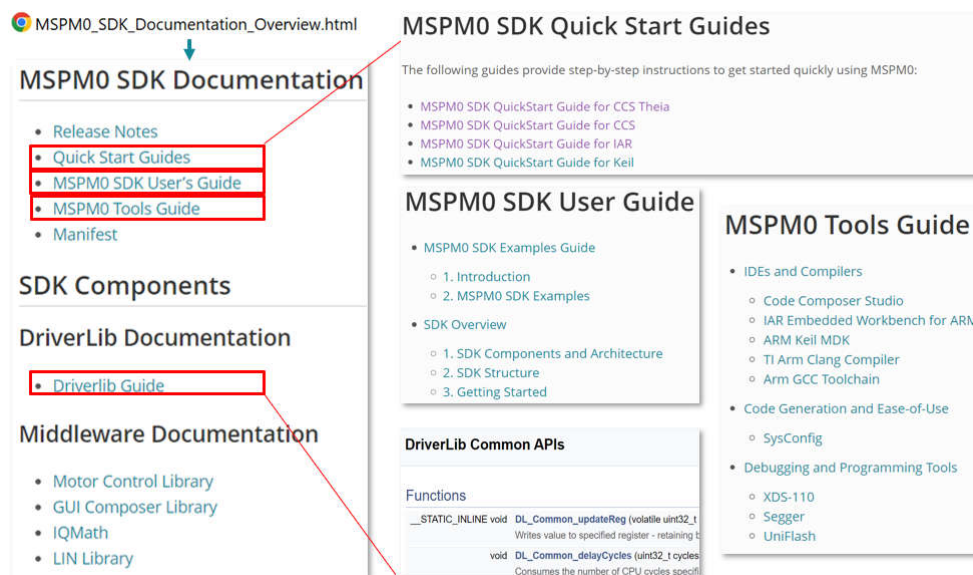
| SDK でサポート | プラットフォーム                 |         |                |               |
|-----------|--------------------------|---------|----------------|---------------|
| IDE       | CCS                      |         | Keil           | IAR           |
| コンパイラ     | TI Arm-Clang             | GUN Arm | Arm/Keil コンパイラ | IAR Arm コンパイラ |
| RTOS      | FreeRTOS                 |         |                |               |
| サンプル コード  | Driverlib、TI ドライバ (ドライバ) |         |                |               |

nortos の例にある 3 つの空のプロジェクトが、ユーザーがプロジェクトをビルドするために使用できます。表 2-7 違い。

**表 2-7. 空のプロジェクトの説明**

| 例                        | SysConfig を使用します | プロジェクトにライブラリファイルを含めます |
|--------------------------|------------------|-----------------------|
| 空                        | あり               | いいえ                   |
| empty_library            | いいえ              | あり                    |
| empty_driverlib_src (推奨) | あり               | あり                    |

Docs フォルダの構造と重要なドキュメントを 図 2-28 に表示します。



**図 2-28. ドキュメントの概要**

## 2.2.4 ステップ 4: ソフトウェア評価

以下に、サンプルを CCS にインポートする手順を示します。

1. メニューから「Project」(プロジェクト)を選択し、「Import CCS Projects」(CCS プロジェクトをインポート)をクリックします。

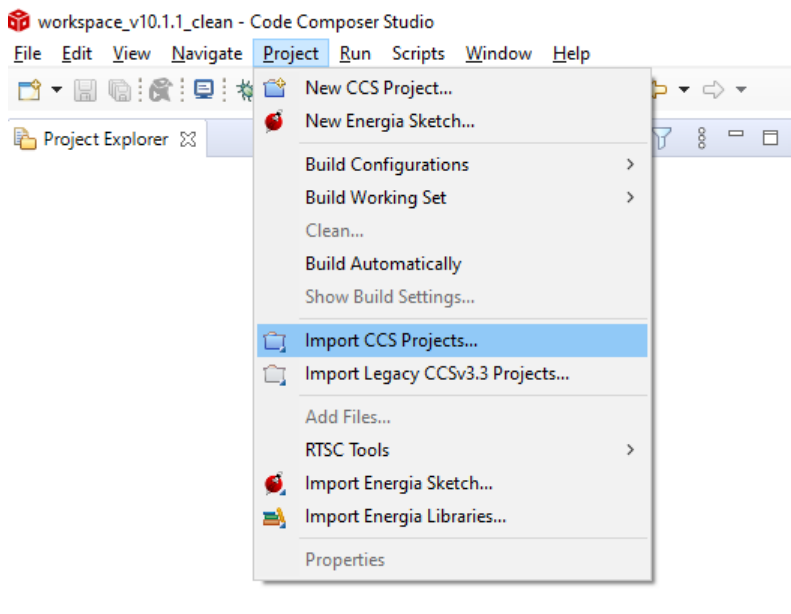


図 2-29. CCS プロジェクトのインポート

2. SDK からプログラムを選択します。たとえば、MSPM0L1306 を考えてみましょう。

\\mspm0\_sdk\_1\_10\_00\_05\\examples\\nortos\\LP\_MSPM0L1306\\driverlib

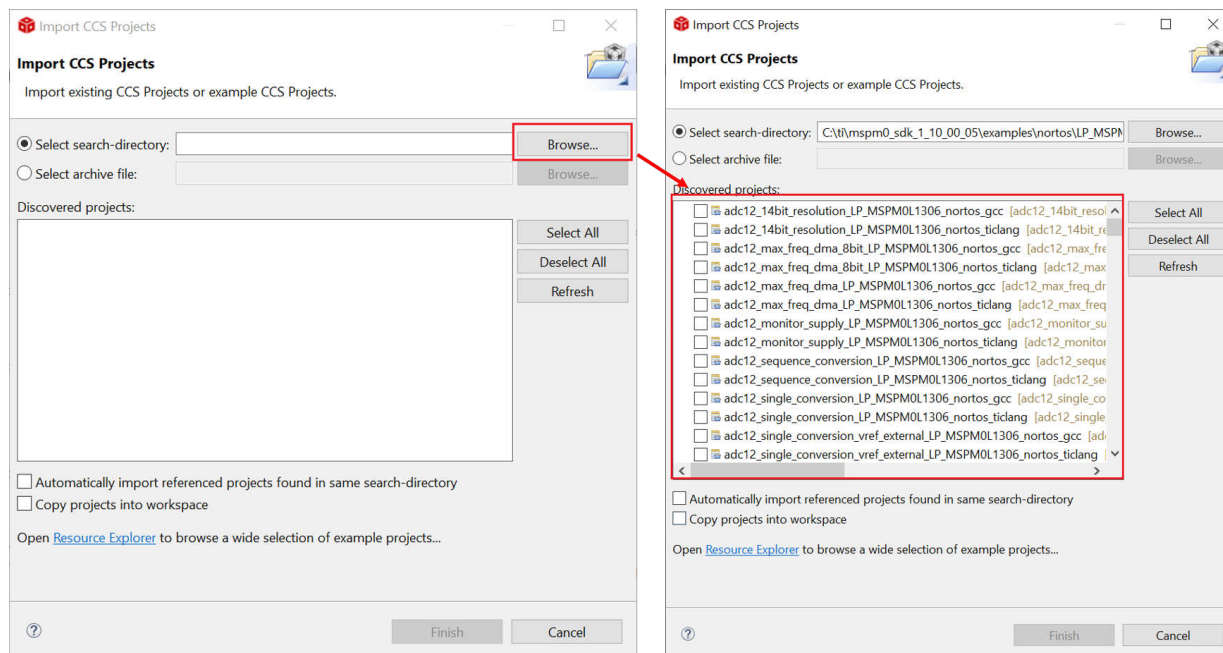


図 2-30. SDK からプログラムを選択します

ファイルをインポートできない場合、ワークスペースの下にある同じ名前のプロジェクトを削除します。

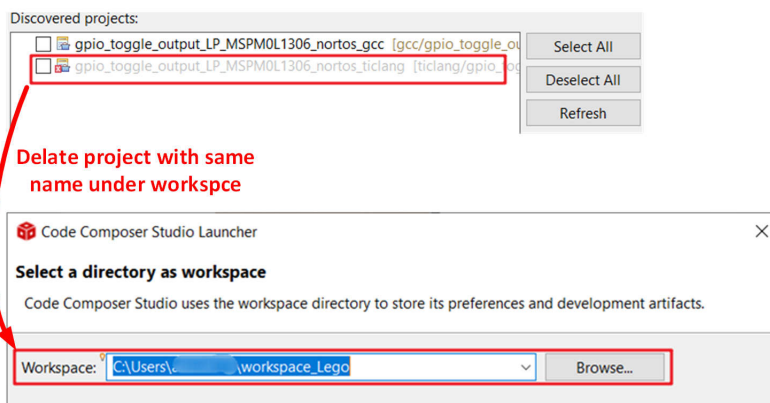


図 2-31. 重複したプロジェクトの削除

- インポート後、左側にプロジェクトが表示され、自動的に README.md が開きます。TI では、最初に README.MD ファイルを読むことを推奨します。このファイルには、この例の目的ハードウェア構成が記載されています。

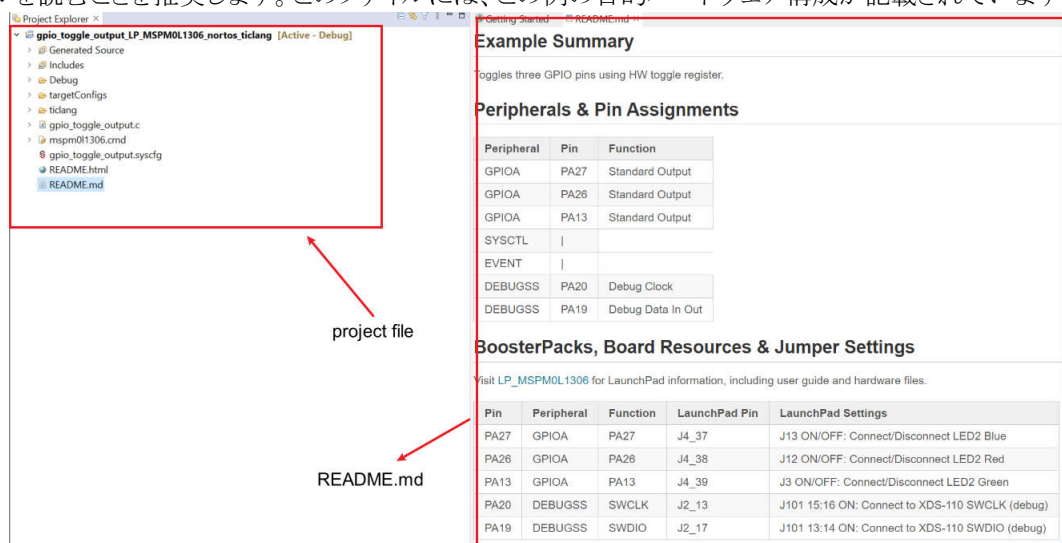


図 2-32. プロジェクトと README.md

4. 図 2-33 に、プロジェクトで重要なファイルを示します。

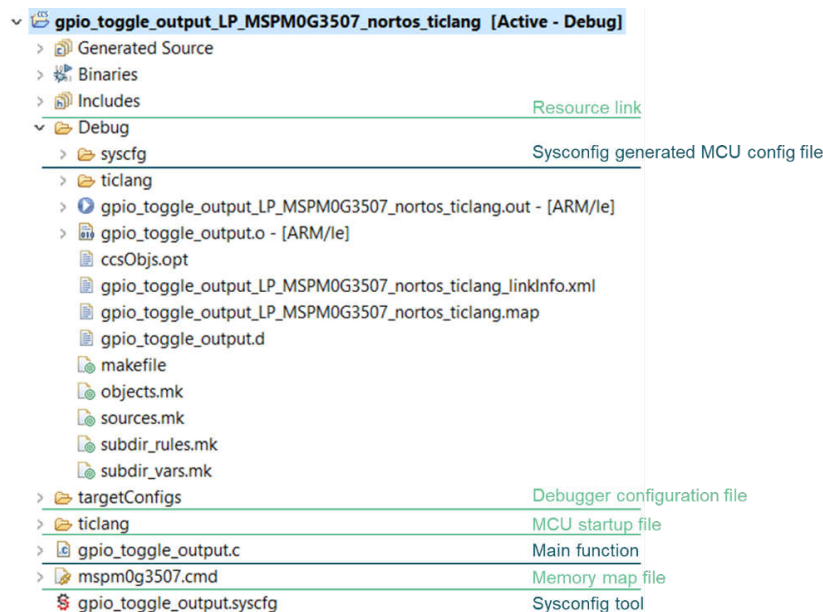
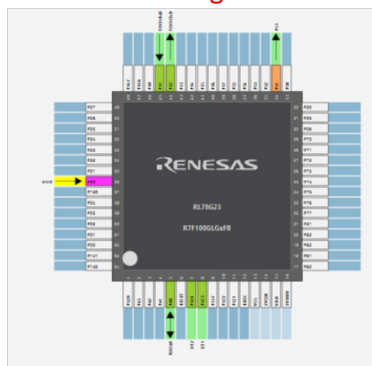


図 2-33. CCS プロジェクトの概要

5. RL78 開発と同様に、.scfg ファイルをダブルクリックしてスマート構成インターフェイスにアクセスします。.syscfg ファイルをダブルクリックすると SysConfig が開き、必要な周辺機能をグラフィカルなインターフェースで設定できます。TI では、SysConfig の MCU ビューの使用を推奨しています。これを使用すると e2studio の MCU または MPU パッケージと同様に、ソフトウェアエンジニアと連携して最初にピン機能を決定することができます。

RL78 MCU Package



MSPM0 MCU view

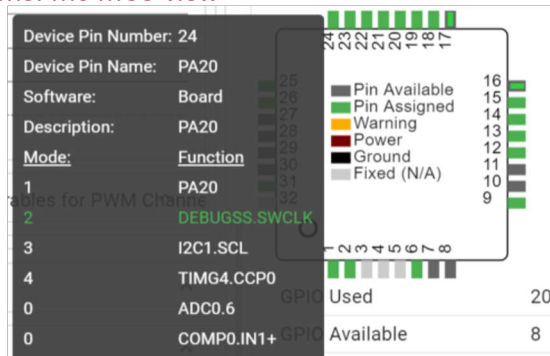


図 2-34. Smart Configuration と SysconfigSysConfig の MCU ビュー

6. コードや SysConfig のサンプルをもとに、ユーザーはプロジェクトを改良したり、[Ti.com](http://ti.com) で公開されているデバイス固有の TRM (テクニカルリファレンスマニュアル) またはアプリケーションノートを参照して、プロジェクトを編集することができます。
7. サードパーティのライブラリを追加する場合、以下の手順に従ってください。まず、関連ファイルをプロジェクトに追加します (図 2-35 を参照)。

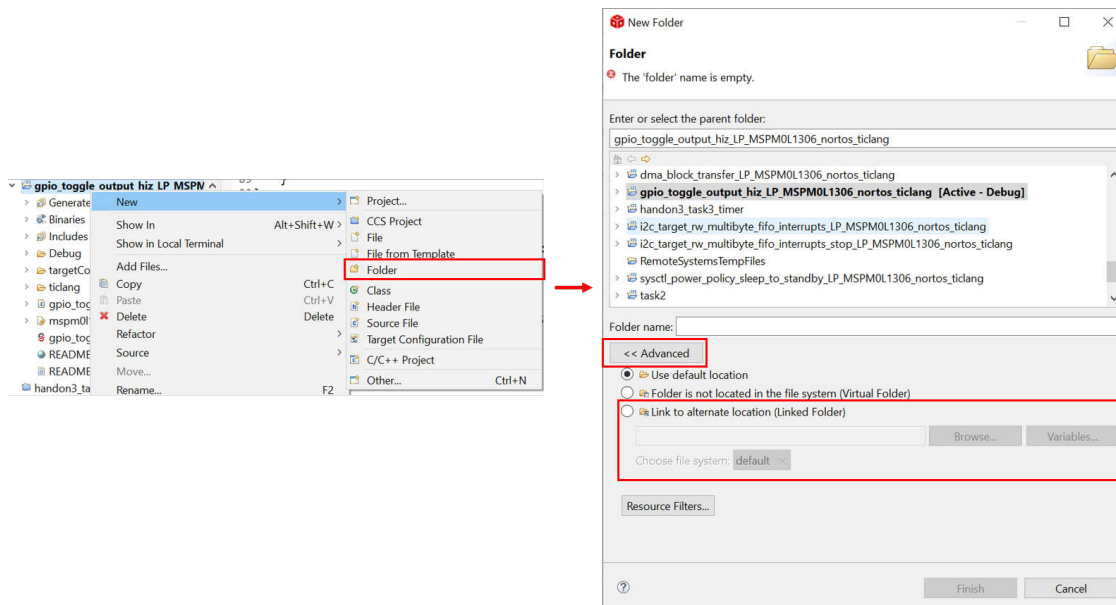


図 2-35. 関連ファイルの追加

次に、コンパイラにヘッダファイルを追加するよう指示するための設定が必要です。

Step1. Left-click on the project

Step2.

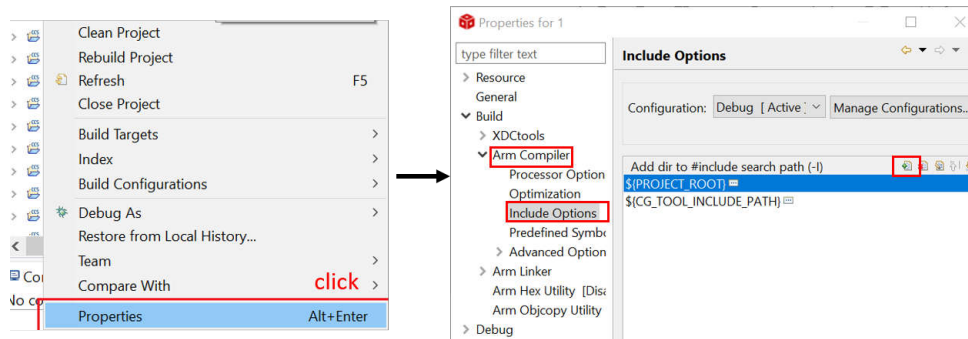


図 2-36. オプションセットを含める

8. ソフトウェアの評価を完了したら、メインツールバーのビルドアイコンをクリックします (図 2-37 を参照)。「Build Finished」(ビルド完了) の表示には、正常に完了したことが示されます。

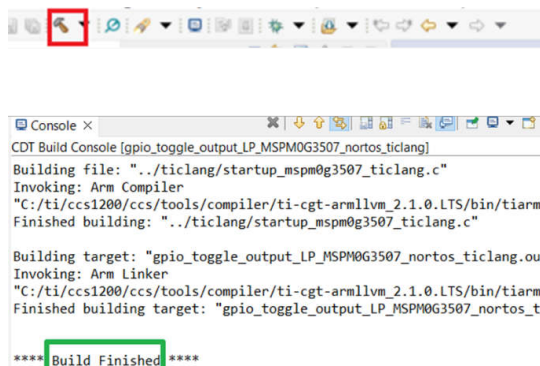


図 2-37. ビルド成功

## 2.2.5 ステップ5.PCB ボードの設計

1. 設計パッケージを入手するには、[Ti.com](https://www.ti.com) にアクセスし、目的のデバイスページに移動します。[MSPM0L1304](#) を例にします。
2. 「Design and development」(設計と開発) → CAD/CAE シンボルをクリックします。必要に応じて異なるパッケージモデルを選択してダウンロードします。

MSPM0L1304 ACTIVE step 1

Product details | Technical documentation | Design & development | Ordering & quality | Support & training

Design & development

For additional terms or required resources, click any title below to view the detail page where available.

step 2

All | Hardware development | Software development | Design tools & simulation | CAD/CAE symbols

step 3

| Package          | Pins | Download                     |
|------------------|------|------------------------------|
| SOT-23-THN (DYY) | 16   | <a href="#">View options</a> |
| VQFN (RGE)       | 24   | <a href="#">View options</a> |
| VQFN (RHB)       | 32   | <a href="#">View options</a> |
| VSSOP (DGS)      | 28   | <a href="#">View options</a> |
| VSSOP (DGS)      | 20   | <a href="#">View options</a> |

図 2-38. Ultra Librarian ツールへの入口

3. 自作の基板に組み込むための MSPM0 ハードウェア設計。図 2-39 に、最小システム設計例を示します。

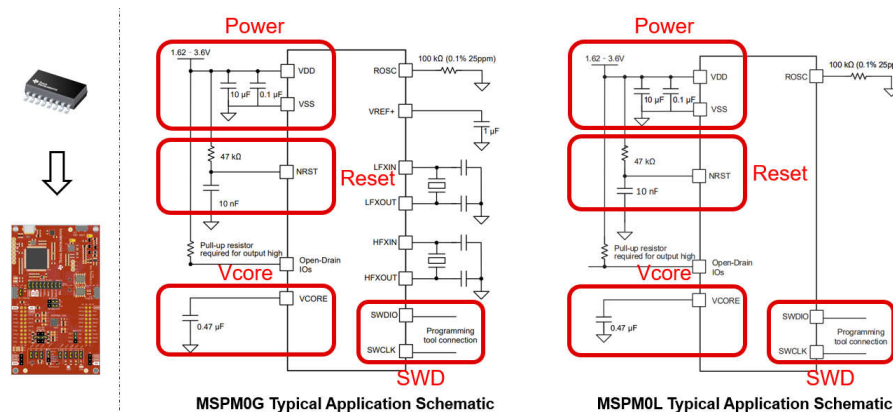


図 2-39. MSPM0 最小システム

最小システムにする場合、次の点に注意する必要があります：

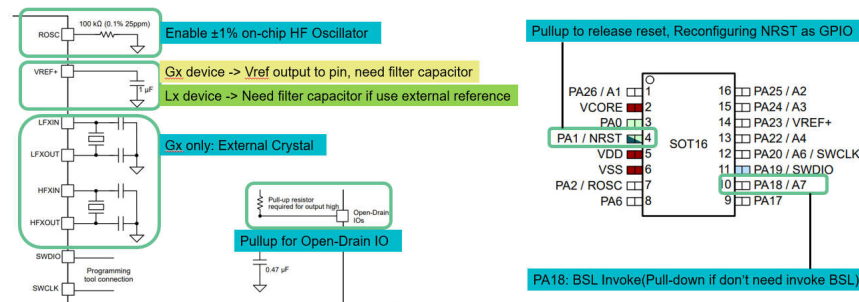


図 2-40. MSPM0 最小システムの注意事項

ハードウェア開発の詳細については、次の資料を参照してください:

- 『[MSPM0 G 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](#)』

## 2.2.6 ステップ6.量産

- CCS を使用して本番運用ファイル (.bin/.txt/...) を生成します。

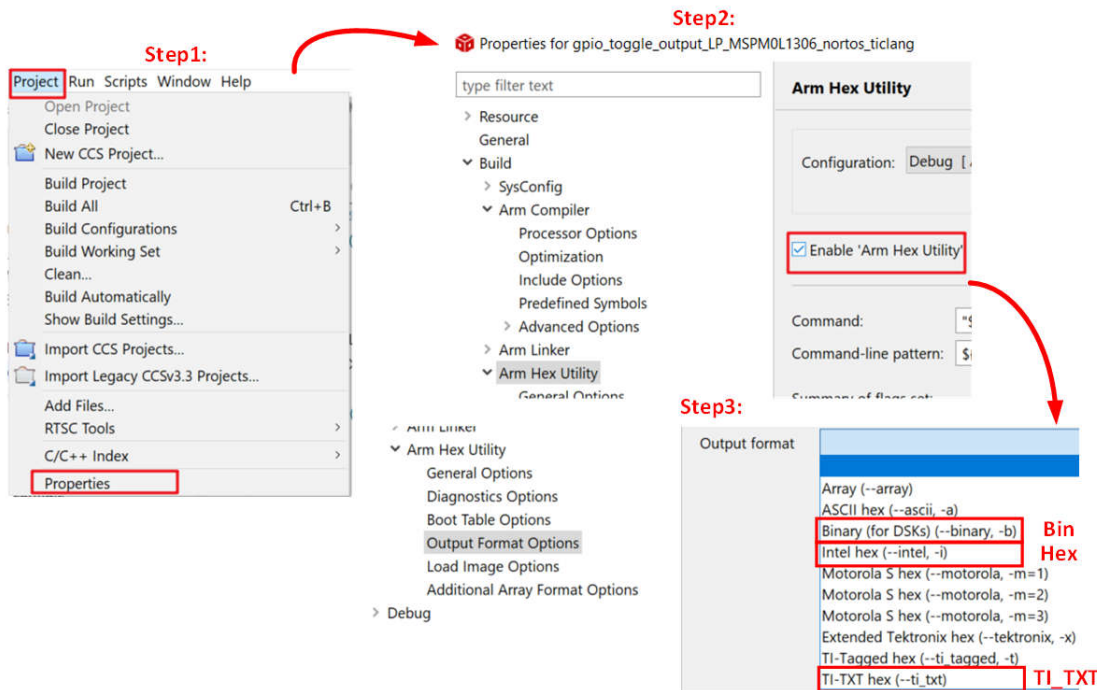


図 2-41. プログラム ファイルの作成

- MSP デバイスをプログラムするプログラマ/デバッガを選択します。

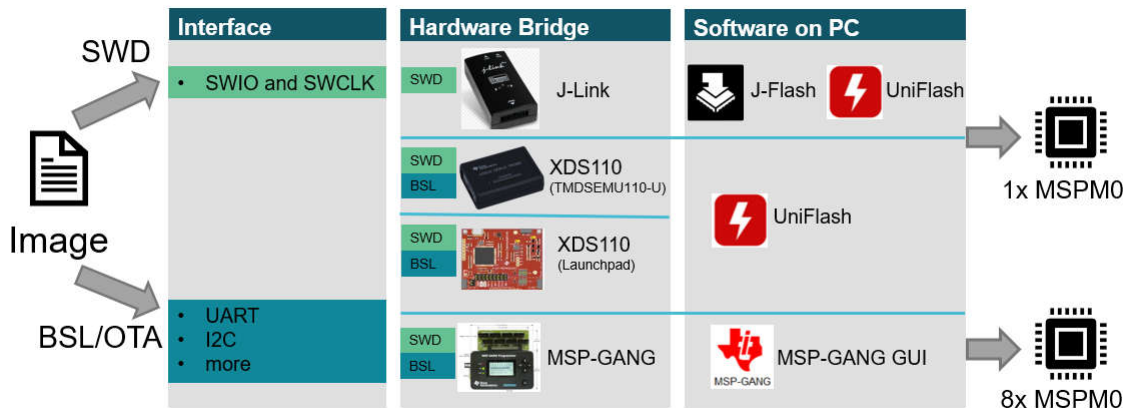


図 2-42. プログラムソフトウェアおよびツール

MSP-GANG および J-LINK の使用方法については、以下を参照してください: [MSPM0 デザイン フロー ガイド](#)。  
デバッグの詳細については、以下を参照してください: [デバッグおよびプログラミング ツール ガイド](#)。

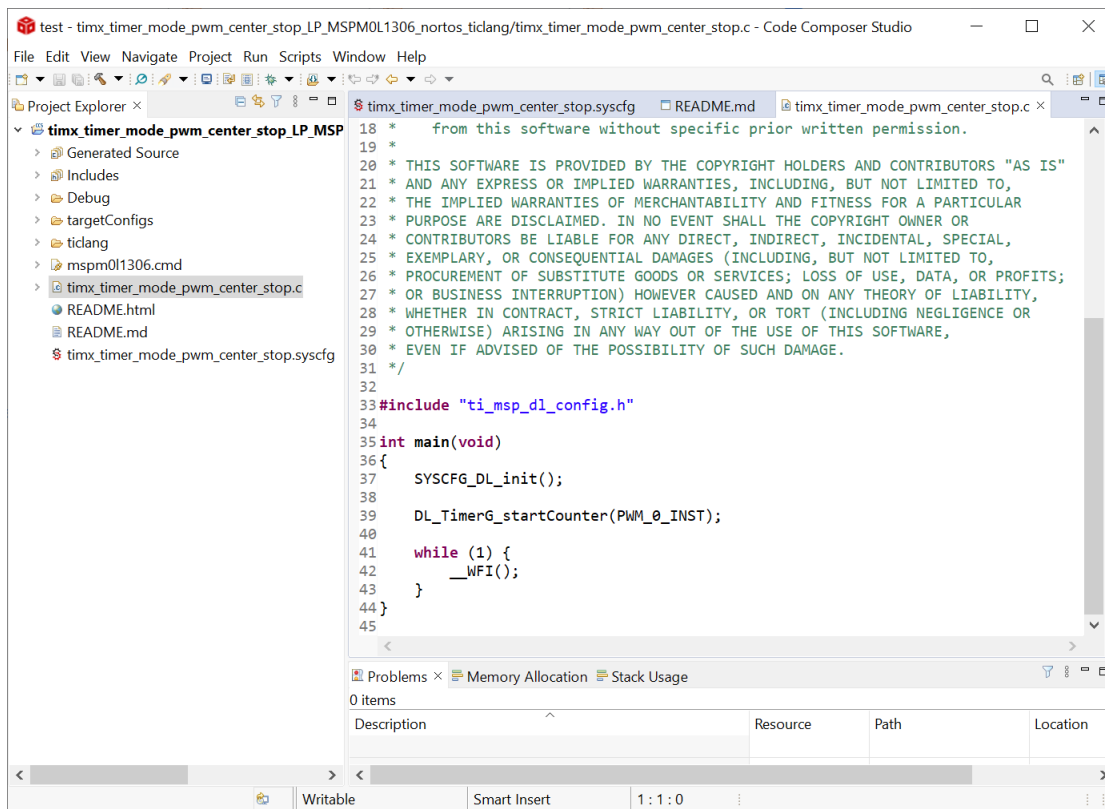
## 2.3 例

MSPM0 の設計フローを以下に示します。この例では、PWM を使用して LED を駆動することを目的としています。

1. 適切な MSPM0 MCU を選択し、ハードウェアを選択して、EVM を注文します。LaunchPad MSPM0L1306 を使用します。
2. CCS と SDK を設定します。(セクション 2.2 を参照してください。)
3. コードをインポートします。

環境の準備ができた時点で、CCS にコードをインポートできます。この例では、タイマを使用して PWM を制御します。最初に RL78 と MSPM0 のタイマーモジュールの違いを理解し、SDK 内で類似する例を選びます。

SDK 内で最も近い例は、`timx_timer_mode_pwm_center_stop` です。類似の例が見つかったら、CCS を開き、「Project > Import CCS Project...」の順に選択してサンプルコードをインポートし、MSPM0 SDK のサンプルフォルダに移動します。



```

18 *   from this software without specific prior written permission.
19 *
20 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
21 * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
22 * THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
23 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
24 * CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
25 * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
26 * PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS;
27 * OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
28 * WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
29 * OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
30 * EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
31 */
32
33#include "ti_msp_dl_config.h"
34
35int main(void)
36{
37    SYSCFG_DL_init();
38
39    DL_TimerG_startCounter(PWM_0_INST);
40
41    while (1) {
42        __WFI();
43    }
44}
45

```

図 2-43. コード サンプル ファイル

#### 4. プロジェクトを編集します。

**SysConfig** の設定内容を確認するには、**.syscfg** ファイルを開きます。PWM を生成するために **TIMER-PWM** セクションを選択します (図 2-44 を参照)。PWM のクロック設定 (自己周波数やデューティサイクル) を確認します。この例では、PWM の周波数が **2.7Hz**、デューティサイクルが **75%** です。目的のデューティサイクル (**50%** など) を入力することでデューティサイクルを簡単に変更でき、カウンタ比較値が自動的に変更されます。

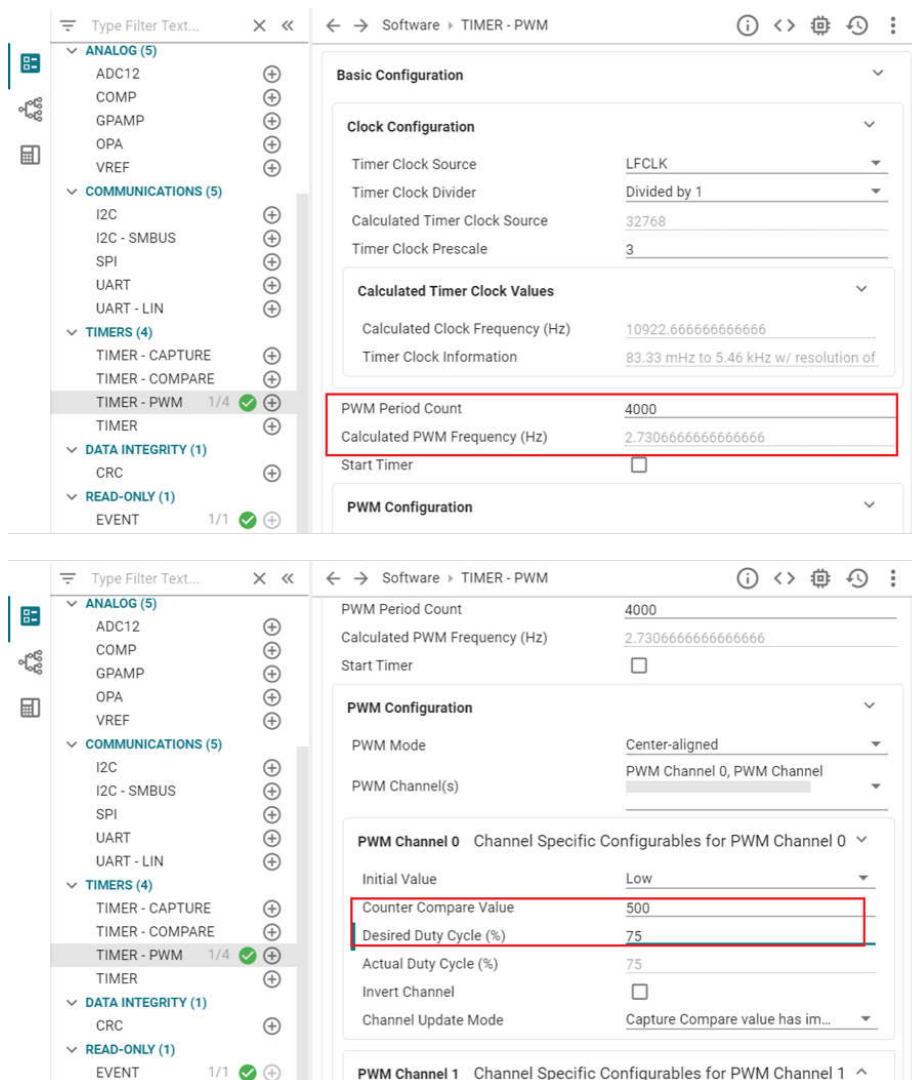


図 2-44. SysConfig 内の PWM 構成

各機能モジュールの詳細については、各項目の横にある「?」をクリックしてください。

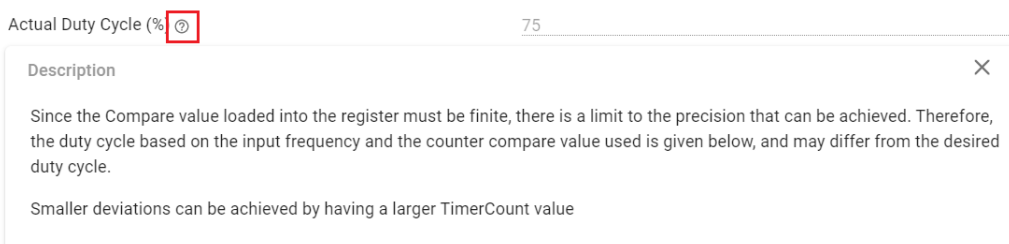


図 2-45. 各項目の詳細情報の取得

また、右上にあるチップ アイコンをクリックし、TIMER-POWER モジュールの他の機能と使用中のピンを確認します。強調表示されている PWM のピンを確認します。

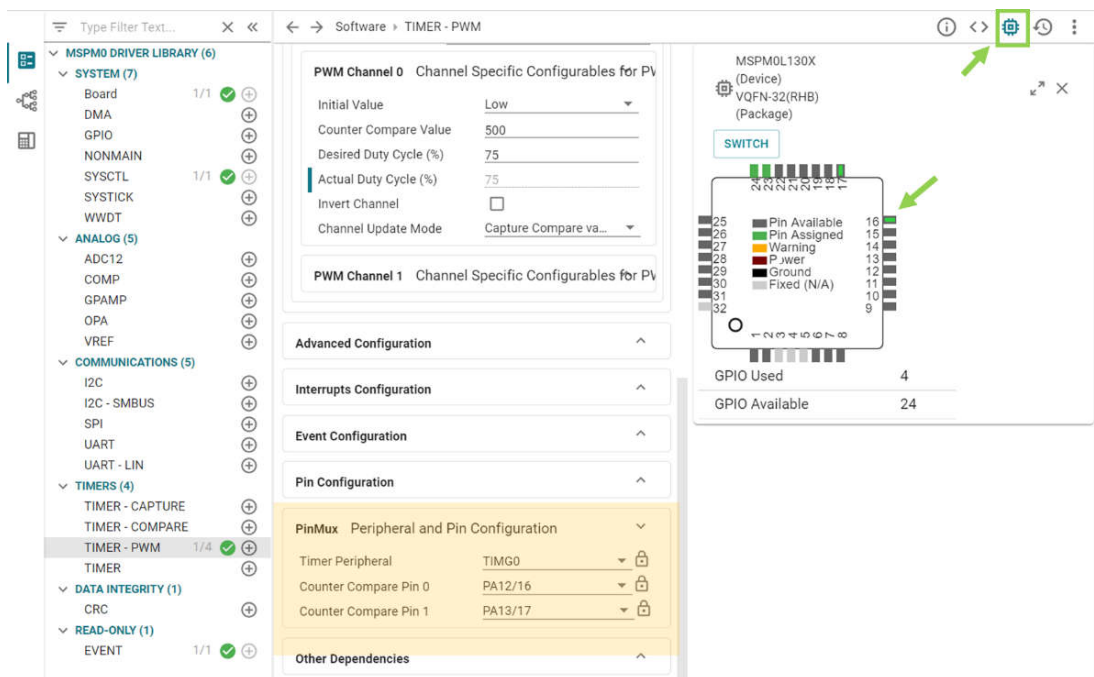


図 2-46. ピン構成

プロジェクトを保存して再ビルドすると、SysConfig によって、図 2-47 内のファイルが更新されます。この時点で、サンプル ハードウェア構成が変更され、移植対象の元のソフトウェアのすべての機能が一致するようになりました。

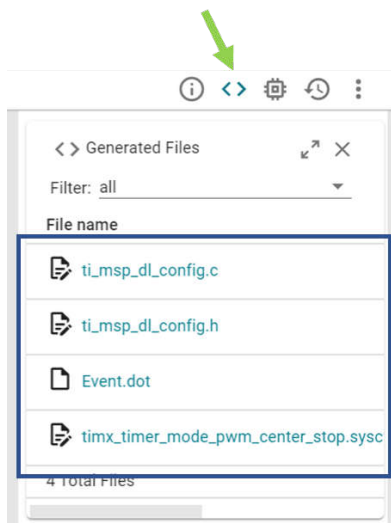


図 2-47. SysConfig ファイルの更新

あとは、アプリケーションレベルのソフトウェアを確認するだけです。この例では、SDK コードと同様に PWM 波形を生成するため、.c ファイルを変更する必要はありません。

5. 次はハードウェアの設定です。

ローンチパッドをコンピュータに接続します。ピン構成に従って、DuPont ケーブルを使用して PA12 を LED ピンに接続します。

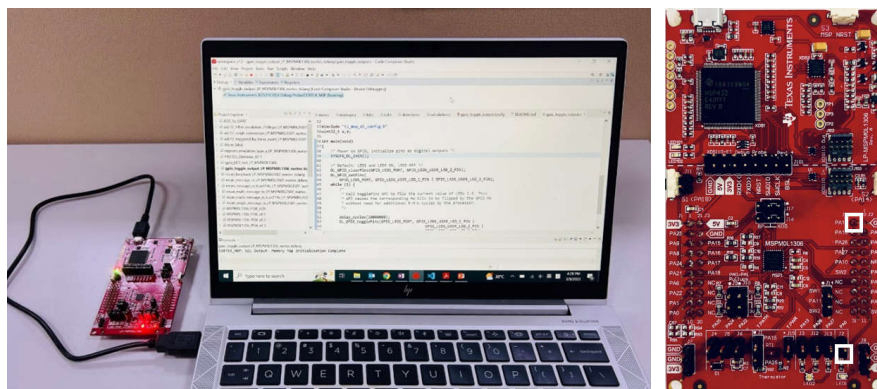


図 2-48. ハードウェア設定

6. デバッグと検証。

デバッグアイコンをクリックして、デバッグを開始します。行番号の前のスペースをダブルクリックするか、1 行のコード `__BKPT()` を追加することで、ブレークポイントを設定できます。

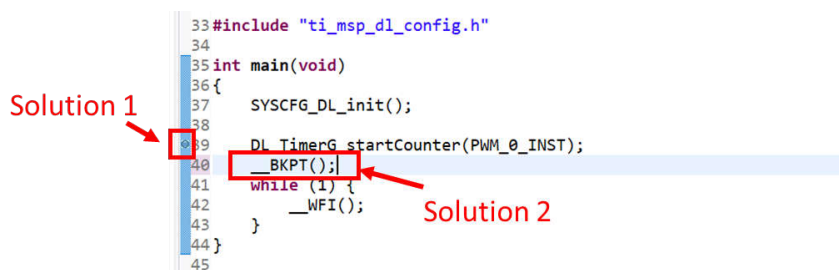


図 2-49. ブレークポイントソリューションの追加

デバッグ機能 (詳細は[セクション 2.2.2.2](#)を参照) を使用して、手順が正しく実行できるかどうかを検証します。デバッグ中にコードをステップ実行すると LED の点滅を確認できます。

7. PCB ライブラリを生成し、Altium Design にインポートします。

具体的な手順を[図 2-50](#)に示します。MSPM0 デバイスページの下にある Ultra Librarian ツールへの入り口に移動します (詳細は[セクション 2.2.5](#)を参照)。表示オプションをクリックします。希望する CAD 形式とピン配列を選択し、Altium 設計ライブラリファイル入手します。

## step1

MSPM0L1306 ✔ ACTIVE

Product details | Technical documentation | **Design & development** | Ordering & quality | Support & training

### Design & development

For additional terms or required resources, click any title below to view the detail page where available.

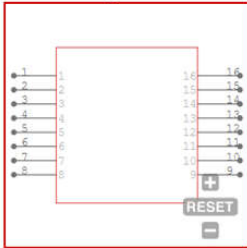
All | Hardware development | Software development | Design tools & simulation | **CAD/CAE symbols**

| Package          | Pins | Download            |
|------------------|------|---------------------|
| SOT-23-THN (DYY) | 16   | <b>View options</b> |
| VQFN (RGE)       | 24   | View options        |
| VQFN (RHB)       | 32   | View options        |
| VSSOP (DGS)      | 20   | View options        |
| VSSOP (DGS)      | 28   | View options        |

## step2

Ultra Librarian | Texas Instruments - XMSM0L1306SDYYR | English

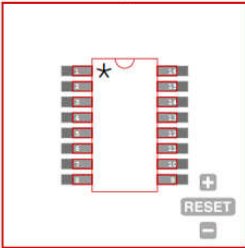
**Symbol**



Normal View ▼

SOT\_06SDYYR 1 ▼

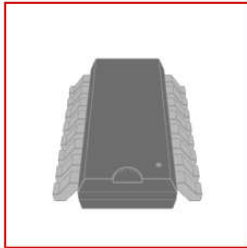
**Footprint**



Basic View ▼

SOT\_06SDYYR\_TEX ▼

**3D Model**



**Choose CAD Formats & Download**

## step3

Ultra Librarian | Texas Instruments - XMSM0L1306SDYYR | English

Choose CAD Format(s) Return to Previews

**3D CAD Model**

Altium ▼

☒ Altium Designer 

☐ PCAD v14 

☐ PCAD v15 

Autodesk ▶

Cadence ▶

DesignSpark ▶

KiCAD ▶

Mentor ▶

Pulsonix ▶

Quadcept ▶

TARGET 3001I ▶

Zuken ▶

Symbol Pin Ordering ? Sequential ▼      Footprint Units ? English (mil) ▼

☐ I have read and agree to the Ultra Librarian Terms And Conditions

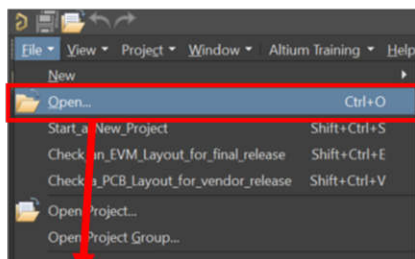
☐ 进行人机身份验证 

**Submit**

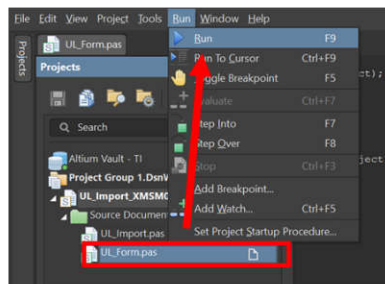
図 2-50 Ultra Librarian ツールのダウンロード

ライブラリをダウンロードしたら、Altium Designer のスクリプトを実行し、PCB ライブラリと回路図ライブラリを生成します (図 2-51 を参照)。

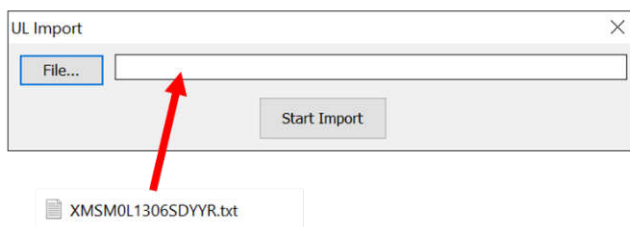
**Step 1. Open .PjScr file**



**Step 2. run UL\_Form.pas**

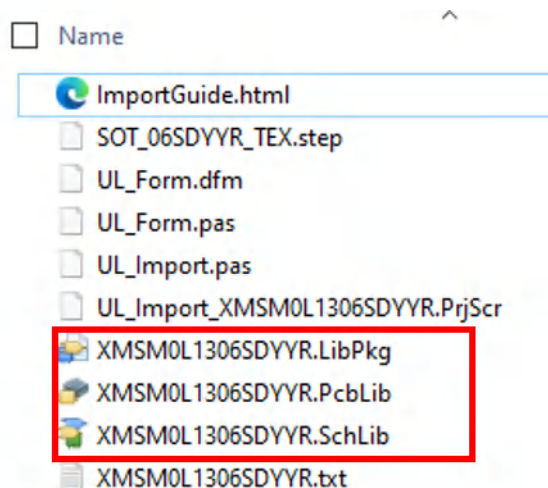


**Step 3. import file**



**図 2-51. Altium Designer スクリプトを実行**

手順を完了すると、同じソースフォルダ内に以下の新しいファイルが生成されます。



**図 2-52. PCB ライブラリと回路図ファイル**

最後に、AD ライブラリにこれらをインポートします (図 2-53 を参照)。これをもとに、回路図と PCB を設計できます。

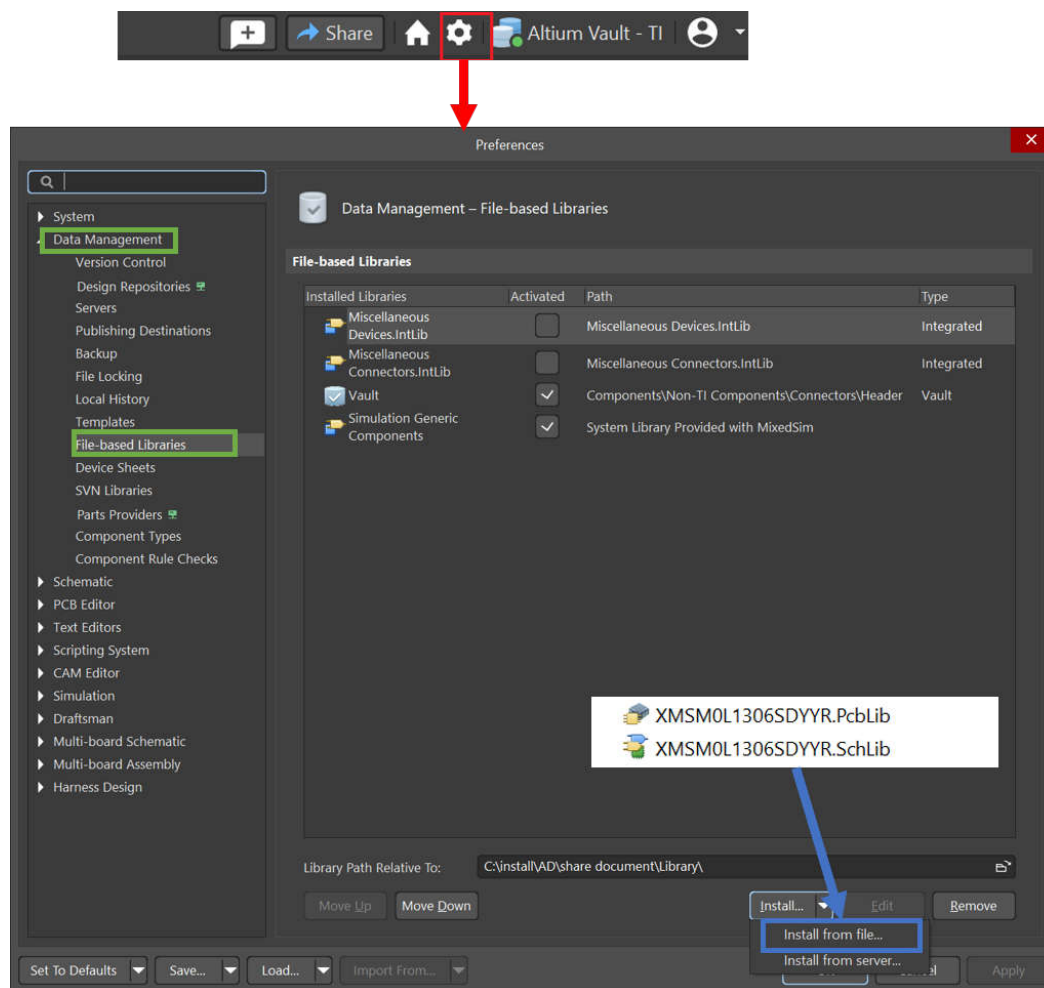


図 2-53. ライブラリのインポート

8. MSPM0 での設計。
9. 量産。

## 3 コア アーキテクチャの比較

### 3.1 CPU

MSPM0 ファミリーは、ARM Cortex M0+ CPU コアアーキテクチャをベースにしています。RL78 ファミリーは、RL78 CPU コアアーキテクチャをベースにしています。表 3-1 に、MSPM0 ファミリーの CPU と RL78 の CPU の主な特長を比較して紹介しています。

表 3-1. CPU 機能セットの比較

| 特長           | RL78                  | MSPM0G              | MSPM0L              | MSPM0C                                                                     | MSPM0H              |
|--------------|-----------------------|---------------------|---------------------|----------------------------------------------------------------------------|---------------------|
| アーキテクチャ      | プライベート RL78 コア        | Arm Cortex M0+      | Arm Cortex M0+      | Arm Cortex M0+                                                             | Arm Cortex M0+      |
| 命令セット        | CISC                  | RISC                | RISC                | RISC                                                                       | RISC                |
| パイプライン       | 3 ステージ                | 2 ステージ              | 2 ステージ              | 2 ステージ                                                                     | 2 ステージ              |
| 動作周波数 (最大)   | 40MHz                 | 80MHz               | 32MHz               | MSPM0C1103 および<br>MSPM0C1104: 24MHz<br>MSPM0C1105 および<br>MSPM0C1106: 32MHz | 32MHz               |
| DMA          | あり                    | あり                  | あり                  | あり                                                                         | あり                  |
| Coremark/MHz | 1.5952 <sup>(1)</sup> | 2.39 <sup>(2)</sup> | 2.39 <sup>(2)</sup> | 2.39 <sup>(2)</sup>                                                        | 2.39 <sup>(2)</sup> |

- (1) RL78G23 のスコアは、32MHz の動作周波数とルネサス CC-RL V1.12 コンパイラで得られます。  
(2) スコアは、ARM 公式 Web サイトの *Arm Cortex-m0+ プロセッサ* のデータシートから取得しています。

### 3.2 組み込みメモリの比較

#### 3.2.1 フラッシュの特長

MSPM0 および RL78 ファミリーの MCU には、実行可能なプログラム コードとアプリケーション データの保存に使用される不揮発性フラッシュ メモリが搭載されています。表 3-2 に、フラッシュ機能の比較を示します。

表 3-2. フラッシュ機能の比較

| 特長          | RL78                                                                                                    |                                                                                                                          | MSPM0                                                                                                                  |
|-------------|---------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
| フラッシュ メモリ   | プログラム フラッシュ                                                                                             | RL78Gxx の範囲: 1KB ~ 768KB<br>RL78Lxx の範囲: 8KB ~ 256KB<br>RL78Ixx, RL78Hxx の範囲: 8KB ~ 512KB<br>RL78Fxx の範囲は 8KB ~ 512KB です | MSPM0Gxx の範囲: 32KB ~ 128KB<br>MSPM0Lxx の範囲: 8 KB ~ 64 KB、<br>MSPM0Cxx は 8 Kb または 64 KB、<br>MSPM0Hxx は 32KB または 64KB です |
|             | データ フラッシュ                                                                                               | RL78Gxx の範囲は 0 ~ 8KB<br>RL78Lxx の範囲は 2KB ~ 8KB<br>RL78Ixx, RL78Hxx の範囲は 0KB ~ 4KB<br>RL78Fxx の範囲は 4KB ~ 16KB です          |                                                                                                                        |
| 単一フラッシュ サイズ | プログラム フラッシュ                                                                                             | 32 ビット                                                                                                                   | 64 ビット                                                                                                                 |
|             | データ フラッシュ                                                                                               | 32 ビットまたは 8 ビット                                                                                                          |                                                                                                                        |
| メモリ構成       | ブロック サイズ (512 B または 1KB)<br>バンク サイズ (可変)<br>ほとんどのデバイスは 2 バンク<br>11C デバイス (512KB) は 3 バンク <sup>(1)</sup> |                                                                                                                          | セクタ サイズ (1KB)<br>バンク サイズ (可変)<br>デバイス最大 256KB - 1 バンク<br>256KB - 2 バンクを超えるデバイス                                         |
| アクセス権       | 8 ビットまたは 16 ビット                                                                                         |                                                                                                                          | シングル フラッシュ ワード (64 ビット) または複数ワード                                                                                       |
| プログラム モード   | プログラム フラッシュ                                                                                             | シングル フラッシュ ワード (32 ビット)                                                                                                  | シングル フラッシュ ワード (64 ビット) または複数ワード                                                                                       |
|             | データ フラッシュ                                                                                               | シングル フラッシュ ワード (32 または 8 ビット)                                                                                            |                                                                                                                        |

表 3-2. フラッシュ機能の比較 (続き)

| 特長        | RL78               | MSPM0                            |
|-----------|--------------------|----------------------------------|
| 消去        | ブロック消去             | セクタ消去<br>バンク消去 (最大 256KB)        |
| エラー コード訂正 | RL78F23, F24 をサポート | 対応                               |
| 書き込み保護    | あり                 | あり、静的と動的                         |
| 読み取り保護    | あり                 | あり                               |
| サイクル      | 1000k (TYP)        | 100K (下位 32KB) または 10k (32KB 超過) |

- (1) ほとんどの RL78 デバイスには 2 つのバンク (1 つのコード バンクと 1 つのデータ バンク)があり、コード フラッシュ メモリが 512KB である一部の RL78I1C デバイスには 3 つのバンク (2 つのコード バンクと 1 つのデータ バンク)があります。

前の表に示したフラッシュ メモリ機能に加えて、MSPM0 フラッシュ メモリには以下の機能もあります：

- 電源電圧範囲全体にわたって、インサーキット プログラムと消去がサポートされています。
- 内部プログラミング電圧の生成

### 3.2.2 フラッシュの構成

フラッシュ メモリは 1 つ以上のバンクに編成され、各バンクのメモリはさらに 1 つ以上の論理メモリ領域にマップされ、アプリケーションで使用できるようにシステム アドレス空間が割り当てられます。

#### 3.2.2.1 フラッシュ メモリ領域

表 3-3 に、RL78 デバイスと MSPM0 デバイスのフラッシュ・メモリ領域を示します。

表 3-3. フラッシュ・メモリ領域の比較

| RL78 (コード フラッシュ)       |                              | MSPM0              |                        |
|------------------------|------------------------------|--------------------|------------------------|
| プログラム領域                |                              | MAIN               | アプリケーション コードおよびデータ     |
| ブートクラスタ <sup>(1)</sup> | プログラム領域が狭い                   | NONMAIN            | BCR による構成              |
|                        | オンチップ デバッグ セキュリティ ID 設定      |                    | BSL による構成              |
|                        | フラッシュ シリアル プログラミング セキュリティ ID |                    |                        |
|                        | オプション バイト                    | FACTORY            | デバイス ID など             |
|                        | CALLT テーブル                   |                    |                        |
|                        | ベクタ テーブル                     | データ <sup>(2)</sup> | データまたは EEPROM エミュレーション |

- (1) 2 つのブートクラスタ (クラスタ 0 とクラスタ 1) を持つ一部の RL78 デバイスは、ブートスワップ機能を実装できます。
- (2) 1 つのバンクを持つ MSPM0 デバイスは、FACTORY、NONMAIN、および MAIN 領域を BANK0 (存在する唯一のバンク) に実装しており、データ領域は利用できません。複数のバンクを持つ MSPM0 デバイスは、FACTORY、NONMAIN、および MAIN 領域も BANK0 に実装していますが、MAIN 領域または DATA 領域を実装できる追加のバンク (BANK1 から BANK4) が含まれています。

#### 3.2.2.2 MSPM0 の NONMAIN メモリ

NONMAIN フラッシュ メモリには、FLASHSWP0 や FLASHSWP1 (静的書き込み保護ポリシー) など、デバイスをブートするために BCR および BSL によって使用されるコンフィギュレーション レジスタが含まれています。この領域はその他の目的のためには使用されません。BCR と BSL にはどちらも構成ポリシーがあり、NONMAIN フラッシュ領域にプログラムされた値を変更することで、デフォルト値 (開発および評価時の標準値) をそのままにするか、特定の目的 (量産プログラミング時の標準値) に変更することができます。

#### 3.2.2.3 RL78 のフラッシュメモリ レジスタ

スペシャル ファンクション レジスタ領域 (SFR および 2 番目の SFR) のフラッシュメモリ レジスタは、フラッシュメモリのプログラミングを制御するために使用されます。たとえば、FLPMC レジスタは、コードおよびデータ フラッシュ メモリのプログラミングの有効化/無効化を制御します。また、データ・フラッシュへのアクセスを有効化または無効化するために DFLCTL レジスタが使用されます。

### 3.2.3 内蔵 SRAM

MSPM0 および RL78 ファミリの MCU には、アプリケーション データの保存に使用される SRAM が搭載されています。

**表 3-4. SRAM 機能の比較**

| 特長                             | RL78                                                                                                                          | MSPM0                                                                                    |
|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------|
| <b>SRAM メモリ <sup>(1)</sup></b> | RL78Gxx の範囲は 0.1KB ~ 48KB<br>RL78Lxx の範囲は 1KB ~ 16KB<br>RL78Ixx, RL78Hxx の範囲は 0.7KB ~ 32KB<br>RL78Fxx の範囲は、0.5KB ~ 32KB の範囲です | MSPM0Gxx 16KB ~ 32KB<br>MSPM0Lxx 2KB ~ 4KB<br>MSPM0Cxx: 1 KB またはまたは 8KB<br>MSPM0Hxx: 8KB |
| <b>パリティ チェック</b>               | 対応                                                                                                                            | MSPM0Gxx: 対応<br>、MSPM0Lxx: 対応 <sup>(2)</sup><br>MSPM0Cxx: 非対応<br>MSPM0Hxx: 非対応           |
| <b>ECC</b>                     | 対応 (RL78F13, F14, F15, F23, F24)                                                                                              | MSPM0Gxx: 対応<br>、MSPM0L: 対応 <sup>(3)</sup> MSPM0Cxx: 非対応<br>MSPM0Hxx: 非対応                |
| <b>書き込み保護 (RAM ガード)</b>        | あり                                                                                                                            | あり                                                                                       |

(1) RL78 の SRAM の特定の領域は汎用レジスタに使用されます。

(2) L シリーズをサポートしているのは、MSPM0Lx22x と MSPM0L111x のみです。

(3) L シリーズの MSPM0Lx22x のみがサポートしています。

MSPM0 MCU には、低消費電力の高性能 SRAM が搭載されており、デバイスでサポートされている CPU 周波数範囲全体にわたってゼロ ウェイト ステートに対応します。SRAM は、コードに加えて、呼び出しスタック、ヒープ、グローバル データなどの情報を格納するために使用できます。SRAM の内容は、実行、スリープ、停止、スタンバイ動作モードでは完全に保持されますが、シャットダウン モードでは失われます。書き込み保護メカニズムが搭載されているため、アプリケーションは 1KB の分解能で下位 32KB の SRAM を動的に書き込み保護できます。32KB 未満の SRAM を搭載したデバイスでは、SRAM 全体に対して書き込み保護が提供されます。書き込み保護は、CPU または DMA によってコードが意図せず上書きされることに対してある程度の保護を提供するため、実行可能コードを SRAM に配置する場合に役立ちます。SRAM にコードを配置すると、ゼロ ウェイト状態動作と低消費電力を実現することで、重要なループの性能を向上できます。

### 3.3 電源投入とリセットの概要と比較

RL78 デバイスと MSPM0 デバイスの両方に最小動作電圧があり、デバイスまたはデバイスの一部をリセット状態に保持することでデバイスを正しく起動させるためのモジュールが搭載されています。表 3-5 に、2 つのファミリ間でどのように行われ、どのモジュールが、ファミリ間で電源投入プロセスとリセットを制御するかを比較したものを示します。

表 3-5. 電源投入の概要と比較

| RL78                                |                                                                                                                                                    | MSPM0                                            |                                                                                                                                                    |
|-------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>POR (パワーオンリセット回路)</b><br>(1)     | 立ち上がり検出: $V_{DD} > V_{POR}$ 、POR リセット信号をリリースし<br>立ち下がり検出: $V_{DD} < V_{PDR}$ 、POR リセット信号が生成されます                                                    | <b>パワーオンリセット (POR)</b>                           | 立ち上がり検出: $V_{DD} > POR+$ 、POR 状態がリリースされ、バンドギャップリファレンスと BOR が開始し<br>立ち下がり検出: $V_{DD} < POR-$ 。デバイスは POR 状態に保持                                       |
| <b>LVD (電圧検出器) -リセットモード</b>         | 立ち上がり検出: $V_{DD} > V_{LVD}$ 、LVD リセット信号をリリースし<br>立ち下がり検出: $V_{DD} < V_{LVD}$ 、LVD リセット信号が生成されます                                                    | <b>ブラウンアウトリセット (BOR) - 0 レベル</b> (2)             | 立ち上がり検出: $V_{DD} > BOR0+$ 、デバイスはブートプロセスを続行し、PMU は<br>立ち下がり検出を開始します。 $V_{DD} < BOR0-$ 、デバイスは BOR 状態に保持。                                             |
| <b>LVD (電圧検出器) - 割り込みおよびリセットモード</b> | 立ち上がり検出: $V_{DD} > V_{LVDH}$ 、LVD リセット信号をリリース<br>立ち下がり検出:<br>1) $V_{DD} < V_{LVDH}$ : 割り込み要求信号が生成される<br>2) $V_{DD} < V_{LVDL}$ : LVD リセット信号が生成されます | <b>ブラウンアウトリセット (BOR) - 1 ~ 3 レベル</b> (2)         | 立ち下がり検出<br>1) $V_{DD} < BORx-$ ( $x = 1, 2, 3$ ) で、割り込み要求が生成され、BOR 回路が自動的に BOR スレッショルドレベルを BOR0.2 に切り替えます<br>$V_{DD} < BOR0-$ 、デバイスは BOR 状態に保持されます |
| <b>LVD (電圧検出器) - 割り込みモード</b>        | 立ち上がり検出: $V_{DD} > V_{LVD}$ 、LVD リセット信号は、LVD リセットが解放された後にリリースされます。 $V_{DD} > V_{LVD}$ または $V_{DD} < V_{LVD}$ の時に、割り込み要求信号が生成されます                   | 該当なし                                             | 該当なし                                                                                                                                               |
| <b>RTC POR (RTC パワーオンリセット)</b>      | RTCPOR 回路の電源電圧と検出電圧の比較による RTC および XT1 オシレータのリセット                                                                                                   | RTC および関連クロックは、BOOTRST、BOR、または POR によってリセットされます。 |                                                                                                                                                    |

- (1) 一部の RL78 デバイスには、電源検出の検出レベルをオプションバイトを使用して選択できる SPOR (Selectable Power-On Reset Circuit) が搭載されています。
- (2) 四つの選択可能な BOR スレッショルドレベル (BOR0 ~ BOR3) があります。スタートアップ中は、デバイスが常に指定された最小  $V_{DD}$  で起動するように、BOR スレッショルドは常に BOR0 (最小値) です。ブート後、ソフトウェアはオプションで BOR 回路を再構成し、別の (より高い) スレッショルド・レベルを使用できます。

RL78 の各種電圧スレッショルドの関係は次のとおりです。  $V_{PDR} < V_{POR} < \text{動作電圧の下限} < V_{LVDL} < V_{LVDH}$ 。 MSPM0 の各種電圧スレッショルドの関係は次のとおりです。  $POR- < POR+ < BOR0- < BOR0+$ 、そして BOR0+ は規定された  $V_{DD}$  最小値であり、内部回路の正常な動作を可能にします。

図 3-1 は MSPM0 リセット関数を表示しています。 MSPM0 のデバイスには 5 つのリセットレベルがあります。パワーオンリセット (POR)、ブラウンアウトリセット (BOR)、ブートリセット (BOOTRST)、システムリセット (SYSRST)、CPU リセット (CPURST) です。

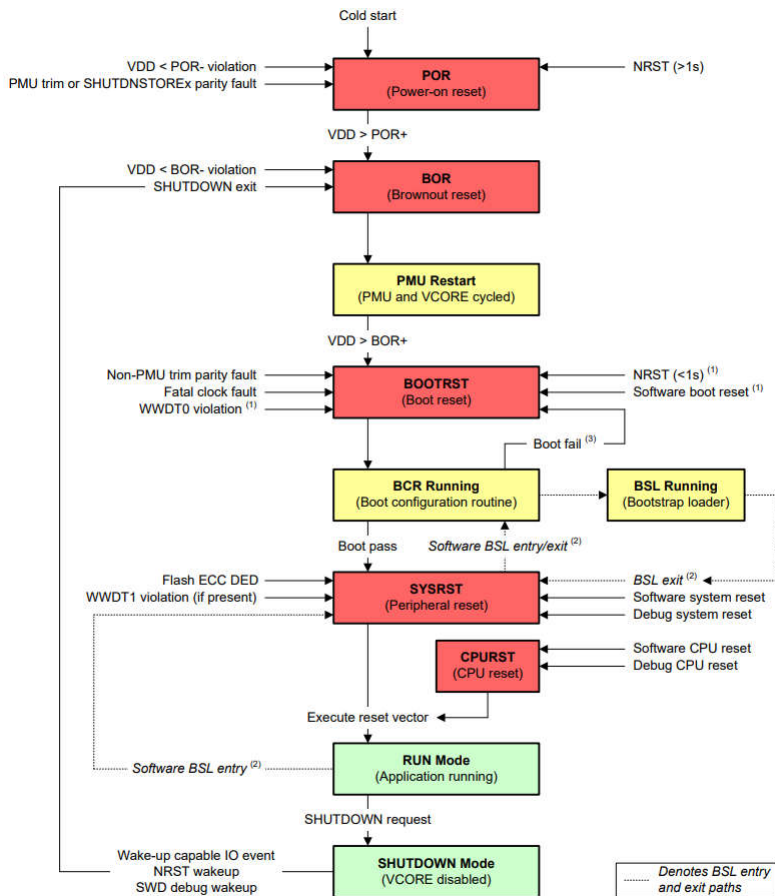


図 3-1. MSP リセット機能

## 3.4 クロックの概要と比較

### 3.4.1 発振器

RL78 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 3-6 に、RL78 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 3-6. 発振器の比較

| タイプ   | RL78                                                      | MSPM0L                       | MSPM0G                                                  | MSPM0C                          | MSPM0H               |
|-------|-----------------------------------------------------------|------------------------------|---------------------------------------------------------|---------------------------------|----------------------|
| 内部発振器 | $f_{HOCO}/f_{IH}$ : 40MHz までの内部高速発振器で、80MHz の PLL に供給できます | 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 | SYSOSC: 内部発振器 (24MHz および 32MHz) | SYSOSC: 内部 32MHz 発振器 |
|       | $f_{IM}$ : 4MHz までの内部中速度発振器                               |                              |                                                         |                                 |                      |
|       | $f_{IL}$ : 内部低速発振器 (15kHz または 32.768kHz)                  | 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: 内部 32kHz 発振器             | LFOSC: 内部 32kHz 発振器  |
|       | $f_{WDT}$ : WDT の内部発振器 (15kHz)                            | 該当なし                         | 該当なし                                                    | 該当なし                            | 該当なし                 |
| 外部発振器 | $f_X$ : 20MHz までの高周波発振器で、80MHz の PLL に供給できます              | 4MHz ~ 32MHz <sup>(1)</sup>  | HFXT: 4MHz から 48MHz への外部高周波発振器。80MHz 用 PLL を供給できます      | 4MHz ~ 32MHz <sup>(1)</sup>     | 該当なし                 |
|       | $f_{XT}$ : 低周波発振器 (32.768kHz または 38.4kHz)                 | 32kHz <sup>(1)</sup>         | LFXT: 32kHz 低周波数外部発振器                                   | 32kHz <sup>(1)</sup>            | 32kHz                |

(1) C および L シリーズの外部発振器に対応しているのは、MSPM0Lx22x および MSPM0C1105 および MSPM0C1106 のみです。

#### 3.4.1.1 MSPM0 発振器

MSPM0 デバイスには、低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 3-7 に、MSPM0 デバイスのクロックソースを表示します。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 3-7. 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 (出荷時に調整された周波数は 4MHz または 32MHz、ユーザー・トリムされた周波数は 16MHz または 24MHz) |
|     | LFSOSC                | 低周波発振器 (代表的な周波数 32kHz)                                                 |
|     | SYSPLL <sup>(1)</sup> | 周波数がプログラム可能なシステム PLL                                                   |
| 外部  | LFXT                  | 低周波の低消費電力水晶発振器 (代表的な周波数 32kHz)                                         |
|     | HFXT                  | 高周波水晶発振器 (代表的な周波数 4 ~ 48MHz)                                           |

(1) SYSPLL 発振器を備えたのは、MSPM0Gx デバイスのみです。

### 3.4.2 クロック信号の比較

異なるクロック信号は分周して他のクロックを供給し、多数のペリフェラルに分配することができます。

表 3-8. クロック信号の比較

| クロックの説明                  |     | RL78 クロック                                                           | MSPM0 クロック                                       |
|--------------------------|-----|---------------------------------------------------------------------|--------------------------------------------------|
| 外部デジタル クロック入力            | 高周波 | EXCLK ( $f_{EX}$ )                                                  | HFCLK_IN                                         |
|                          | 低周波 | EXCLK ( $f_{EXS}$ )                                                 | LFCLK_IN                                         |
| 高周波数外部クロック               |     | $f_{MX}$                                                            | HFCLK                                            |
| 低周波数外部クロック               |     | $f_{SUB}^{(1)}$                                                     | LFCLK_IN と LFXT の選択                              |
| PLL 回路出力クロック             |     | $f_{PLL}$                                                           | SYSPLLCLK0、SYSPLLCLK1、SYSPLLCLK2x <sup>(2)</sup> |
| メイン システム クロック            |     | $f_{MAIN}^{(3)}$                                                    | MCLK、ULPCLK <sup>(4)</sup> (BUSCLK)              |
| CPU/ペリフェラル用の高周波クロック      |     | $f_{MAIN}$ または $f_{MP}/n^{(5)}$                                     | HSCLK <sup>(6)</sup> と SYSOSC の選択                |
| CPU/ペリフェラル用の低周波数クロック     |     | $f_{SL}$                                                            | LFCLK (32kHz 固定)                                 |
| ソース CPU                  |     | $f_{CLK}$                                                           | CPUCLK                                           |
| ほとんどのペリフェラルハードウェア用のクロック  |     | $f_{CLK}$                                                           | MCLK、ULPCLK                                      |
| 高速ペリフェラル用の使用可能なクロック      |     | $f_{MX}$ 、 $f_{IH}$ 、 $f_{MAIN}$ 、 $f_{PLL}$ 、 $f_{MP}$ 、 $f_{CLK}$ | MCLK                                             |
| 低速低消費電力ペリフェラル用の使用可能なクロック |     | $f_{SUB}$ 、 $f_{IL}$ 、 $f_{SL}$ 、 $f_{CLK}$                         | ULPCLK                                           |
| 固定周波数クロック                |     | 該当なし                                                                | MFCLK: 4Mhz (MCLK に同期)                           |
|                          |     |                                                                     | MFPCCLK: 4MHz                                    |

- (1)  $f_{SUB}$  は、低周波数の外部発振器 ( $f_{XT}$ ) または低周波数の外部デジタルクロック入力 (EXCLK) から供給できるサブシステムクロックです。
- (2) SYSPLLCLK2x は PLL の出力速度の 2 倍で、分周できます。
- (3) RL78 のメイン システム クロックは、 $f_{MX}$  または  $f_{HOCO}/f_{IH}$  から供給されます。
- (4) MSPM0 のメイン システム クロックは、LFCLK、HSCLK、または SYSOSC から供給されます。MCLK は PD1 のメイン システム クロックであり、ULPCLK は MCLK から派生した PD0 のメイン システム クロックです。
- (5)  $f_{MP}$  はメイン・システムまたは PLL 選択のクロックであり、 $n$  は 1、2、4、.....と選択できます。ほとんどの RL78 デバイスは  $f_{main}$  を CPU/ペリフェラルの高周波クロックとして選択します。一部の RL78 デバイスでは、CPU / ペリフェラルの高周波クロックとして  $f_{MP}/n$  を選択します。
- (6) HSCLK には、SYSPLL または HFCLK が供給されます。

表 3-9. ペリフェラル クロック ソース

| ペリフェラル               | RL78                                                                                               | MSPM0               |
|----------------------|----------------------------------------------------------------------------------------------------|---------------------|
| リアルタイム クロック (RTC)    | $f_{IH}$ 、 $f_{IL}$ 、 $f_{MX}$ 、 $f_{SUB}$ 、 $f_{CLK}$                                             | LFCLK (LFOSC、LFXT)  |
| UART                 | $f_{CLK}$                                                                                          | BUSCLK、MFCLK、LFCLK  |
| SPI/CSI (簡略化された SPI) | $f_{CLK}$                                                                                          | BUSCLK、MFCLK、LFCLK  |
| I2C                  | $f_{CLK}$                                                                                          | BUSCLK、MFCLK        |
| CAN                  | $f_{MX}$ 、 $f_{MP}$ 、 $f_{CLK}$                                                                    | PLLCLK1、HFCLK       |
| ADC                  | $f_{CLK}$                                                                                          | ULPCLK、HFCLK、SYSOSC |
| タイマ                  | $f_{HOCO}$ 、 $f_{IL}$ 、 $f_{MX}$ 、 $f_{SL}$ 、 $f_{PLL}$ 、 $f_{MP}$ 、 $f_{CLK}$ 、 $f_{TMKB2}^{(1)}$ | BUSCLK、MFCLK、LFCLK  |
| コンパレータ               | $f_{PLL}$ 、 $f_{CLK}$                                                                              | ULPCLK              |

- (1)  $f_{TMKB2}$  クロックは、RL78 MCU の 16 ビット・タイマを供給するためにのみ使用されます。

### 3.5 MSPM0 の動作モードの概要と比較

MSPM0 MCU には 5 つのメイン動作モード (電力モード) があり、アプリケーションの要件に基づいてデバイスの消費電力を最適化できます。消費電力を低減するためのモードは次のとおりです。RUN、SLEEP、STOP、STANDY、SHUTDOWN。CPU は RUN モードではコードをアクティブに実行しています。ペリフェラル割り込みイベントにより、デバイスを SLEEP、STOP、または STANDBY モードから RUN モードにウェークアップできます。SHUTDOWN モードでは、内部コアレギュレータが完全にディセーブルされ、消費電力が最小化されます。また、NRST、SWD、または特定の IO でのロジックレベルの一致によってのみウェークアップが可能です。RUN、SLEEP、STOP、STANDBY の各モードには、複数の構成可能なポリシー オプション (例: RUN.x) も含まれており、性能と消費電力のバランスを確保できます。

性能と消費電力のバランスをさらに高めるために、MSPM0 デバイスには次の 2 つの電力ドメインが実装されています。PD1 (CPU、メモリ、高性能ペリフェラル用) と PD0 (低速、低消費電力ペリフェラル用)。PD1 は、RUN モードと SLEEP モードで常に電源が供給されますが、他のすべてのモードではディセーブルになります。PD0 は、RUN、SLEEP、STOP、STANDBY の各モードで常に電源が供給されます。SHUTDOWN モードでは、PD1 と PD0 の両方がディセーブルになります。

### 3.5.1 動作モードの比較

表 3-10 に、RL78 デバイスと MSPM0 デバイスの簡単な比較を示します。

表 3-10. RL78 デバイスと MSPM0 デバイスの動作モードの比較

| RL78                    |                                                                                                                                                                                                              |                                                                                   | MSPM0   |   |                                                                                                                                                        |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|---------|---|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| 動作モード                   | 説明                                                                                                                                                                                                           |                                                                                   | 動作モード   |   | 説明                                                                                                                                                     |
| メインラン                   | CPU はメインのシステム・クロックで動作します <sup>(1)</sup>                                                                                                                                                                      | CPU、クロック、ペリフェラルが動作                                                                | RUN     | 0 | MCLK と CPUCLK は、高速クロック ソース (SYSOSC、HFCLK、または SYSPLL) で動作します                                                                                            |
|                         | CPU はサブシステム・クロックで動作します                                                                                                                                                                                       | CPU、クロック、ペリフェラルが動作                                                                |         | 1 | MCLK と CPUCLK は LFCLK から実行します (32kHz 時)。                                                                                                               |
|                         |                                                                                                                                                                                                              |                                                                                   |         | 2 |                                                                                                                                                        |
| HALT                    | CPU はメインのシステム・クロックで動作します <sup>(1)</sup>                                                                                                                                                                      | CPU 動作が停止します。メインのシステムクロックの動作が継続されます。サブシステムクロックのステータスは保持されます。ほとんどのペリフェラル機能が動作可能です。 | SLEEP   | 0 | CPU 動作が停止します。SYSOSC はイネーブル状態を維持し、その他の高速発振器はオプションです。低速発振器はイネーブル状態を維持します。MCLK は高速クロック ソース (SYSOSC) で動作します。                                               |
|                         | 該当なし                                                                                                                                                                                                         | 該当なし                                                                              |         | 1 | CPU 動作が停止します。SYSOSC はイネーブルを維持し、その他の高速発振器はディセーブルになります。低速発振器はイネーブル状態を維持します。MCLK は LFCLK から実行されます。                                                        |
|                         | CPU はサブシステム・クロックで動作します                                                                                                                                                                                       | CPU 動作が停止します。メインシステムクロックの動作が停止します。サブシステムクロックの動作を継続します。ほとんどのペリフェラル機能が動作可能です。       |         | 2 | CPU 動作が停止します。高速発振器はディセーブル状態になります。低速発振器はイネーブル状態を維持します。MCLK は LFCLK から実行されます。                                                                            |
| スヌーズ <sup>(2) (3)</sup> | CPU 動作は停止します。f <sub>HOCO</sub> /f <sub>IH</sub> 動作が開始し、f <sub>X</sub> 、f <sub>EX</sub> 、および f <sub>PLL</sub> 動作が停止します。STOP モードで使用されている間のサブシステムクロックのステータスは継続されます。ADC、UART、CSI などのペリフェラル機能は、CPU 動作を介さずに動作できます。 |                                                                                   | ストップ    | 0 | CPU 動作が停止します。SYSOSC のステータスが保持されます。その他の高速発振器は無効になります。低速発振器はイネーブル状態を維持します。ULPCLK は最大 4MHz に制限されます。PD0 はイネーブル、PD1 はディセーブルになります。ADC などのアナログ ペリフェラルは動作可能です。 |
|                         | 該当なし                                                                                                                                                                                                         |                                                                                   |         | 1 | SYSOSC と ULPCLK が 4MHz のギアに切り替えられます (STOP0 と同様)。                                                                                                       |
|                         |                                                                                                                                                                                                              |                                                                                   |         | 2 | CPU 動作が停止します。高速発振器はディセーブル状態になります。ULPCLK は 32kHz で動作します。PD0 はイネーブル、PD1 はディセーブルになります。ADC の使用はサポートされていません。                                                |
| 該当なし                    | 該当なし                                                                                                                                                                                                         |                                                                                   | STANDBY | 0 | CPU 動作が停止します。高速発振器はディセーブル状態になります。すべての PD0 ペリフェラルが ULPCLK と LFCLK を受信します。ADC はサポートされていません。                                                              |
|                         |                                                                                                                                                                                                              |                                                                                   |         | 1 | STANDBY0 と同様に、TIMG0/1 のみが ULPCLK または LFCLK を受信します。                                                                                                     |
| STOP <sup>(3)</sup>     | CPU 動作が停止します。メインシステムクロックの動作が停止します。STOP モードが設定される前のサブシステムクロックのステータスは保持されます。システム全体が停止します。                                                                                                                      |                                                                                   | シャットダウン |   | 利用可能なクロックがなくなり、デバイスはシャットダウンされます。                                                                                                                       |

- (1) CPU は f<sub>IH</sub>/f<sub>HOCO</sub>、f<sub>X</sub>、f<sub>EX</sub>、または f<sub>PLL</sub> で動作できます。
- (2) スヌーズ・モードは、CPU またはペリフェラル・ハードウェア・クロック (f<sub>CLK</sub>) として高速オンチップ発振器が選択されている場合にのみ指定できます。
- (3) スヌーズ・モードは、CSI、UART、A/D コンバータなどに対してのみ規定できます。CSI または UART データ受信の場合、タイマ・トリガ信号による A/D 変換要求など、MCU は STOP モードからスヌーズ・モードにギアシフトされます。次に、CSI または UART データは CPU を動作させずに受信され、A/D 変換が実行されます。

### 3.5.2 低消費電力モードでの MSPM0 機能

MSPM0 のペリフェラル モードは、低消費電力動作モードにおいては、利用可能な機能または動作速度を制限できます。具体的な詳細については、MSPM0 デバイス固有のデータシートに掲載されている「動作モードでサポートされる機能」の表を参照してください。いくつかの例を以下に示します。

- 『MSPM0G350x ミックスド シグナル マイクロコントローラ データシート』
- MSPM0L134x、MSPM0L130x ミックスド シグナル マイクロコントローラ データシート
- MSPM0C110x、MSPS003 ミックスド シグナル マイクロコントローラ データシート

MSPM0 デバイスの追加機能は、一部のペリフェラルが非同期高速クロック要求を実行できることです。これにより、MSPM0 デバイスを、ペリフェラルがアクティブでない低消費電力モードに移行しつつ、ペリフェラルをトリガまたはアクティブにすることもできます。非同期高速クロック要求が発生した場合、MSPM0 デバイスは内部発振器を高速にすばやく立ち上げたり、一時的に高い動作モードに移行して差し迫った動作を処理したりすることができます。これにより、最小消費電力モードでのスリープ中に、タイマ、コンパレータ、GPIO、および RTC からの CPU の高速ウェークアップ、SPI、UART、I2C の受信、または DMA 転送と ADC 変換のトリガを行うことができます。非同期クロック要求の実装とペリフェラルのサポートおよび目的の具体的な詳細については、MSPM0 のデバイス固有の TRM の該当する章を参照してください。

- 『MSPM0 G シリーズ 80MHz マイクロコントローラ テクニカル リファレンス マニュアル』
- 『MSPM0 L シリーズ 32MHz マイクロコントローラ テクニカル リファレンス マニュアル』
- 『MSPM0 C シリーズ 24MHz マイクロコントローラ テクニカル リファレンス マニュアル』

### 3.5.3 低消費電力モードへの移行

MSPM0 デバイスは、イベントの待機、**\_WFE()**、または割り込みの待機、**\_WFI()**、の命令を実行するときに、低消費電力モードに移行します。低消費電力モードは、現在の電力ポリシー設定によって決定されます。デバイスの電力ポリシーは、ドライバライブラリ関数によって設定されます。次の関数呼び出しは、電力ポリシーをスタンバイ 0 に設定します。

**DL\_SYSCTL\_setPowerPolicySTANDBY0 ();**

**STANDBY0** は、任意の動作モードに置き換えることができます。電源ポリシーを管理する **driverlib** API の全リストについては、『MSPM0 SDK API ガイド』の該当するセクションを参照してください。また、さまざまな動作モードの開始方法を示す以下のサンプルコードも参照してください。すべての MSPM0 デバイスで、同様のサンプルを利用できます。

### 3.5.4 低消費電力モードのサンプルコード

SDK のインストール先に移動し、低消費電力モードのサンプルコードを「examples > nortos > LP name > driverlib」から検索します。

## 3.6 割り込みとイベントの比較

### 3.6.1 割り込みと例外

MSPM0 と RL78 は両方とも、デバイスで利用可能なペリフェラルに応じて、割り込みベクトルと例外ベクトルを登録し、マップします。各デバイス ファミリの割り込みベクトルの概要と比較を表 3-11 に示します。割り込みまたは例外の優先度の値が低いほど、優先度の高い値を持つ割り込みよりも優先度が高くなります。プロセッサが割り込みを処理中の場合、より高いプログラム可能な優先度の割り込みのみがその処理をプリエンプトすることができます。

表 3-11. 割り込みの比較

| 特長      | RL78                                               | MSPM0x                                                                                                                                                                                      |
|---------|----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 割り込みタイプ | マスク可能: デバイスによって決定され、内部割り込みと外部割り込みに分類されます。          | ペリフェラル割り込み: MSPM0G の NVIC は最大 32 のペリフェラル割り込みベクタをサポートし、MSPM0L の NVIC は最大 19 個のペリフェラル割り込みベクタをサポート。MSPM0C の NVIC は最大 23 個のペリフェラル割り込みベクタをサポート。MSPM0H は最大 22 個のペリフェラル割り込みベクタをサポート <sup>(1)</sup> |
|         | リセット: デバイスによって決定されます                               | Reset, NMI, Hard Fault, SVCcall, PendSV, SysTick                                                                                                                                            |
| 優先レベル   | デフォルトの優先度レベル: デバイスによって決定されます <sup>(2)</sup>        | デフォルトの優先度レベル: NVIC 番号 <sup>(3)</sup>                                                                                                                                                        |
|         | マスク可能割り込みには、次の 4 つのプログラム可能な優先度レベルがあります: 0、1、2、3    | システム例外 (リセット、NMI、ハードフォルト) にはそれぞれ固定の優先レベル (-3、-2、-1) があります<br>ペリフェラル割り込みには、以下の 4 つのプログラム可能な優先レベルがあります: 0、64、128、192                                                                          |
| 優先度セット  | PR0xy および PR1xy レジスタ: マスク可能割り込み優先度レベルを設定するために使用します | NVIC の IPRx レジスタ: ペリフェラル割り込み優先度レベルを設定するために使用します                                                                                                                                             |
| 割り込みマスク | MKxy レジスタ: 対応するマスク可能割り込みをイネーブル/ディセーブルにするために使用します   | ペリフェラル側の IMASK レジスタ: イベントに伝搬する割り込み条件を設定するために使用します <sup>(4)</sup>                                                                                                                            |
|         |                                                    | NVIC の ISER および ICER レジスタ: ペリフェラル割り込みをイネーブルまたはディセーブルにするために使用します                                                                                                                             |

- (1) NVIC に加え、MSPM0 デバイスには INT\_GROUP0 や INT\_GROUP1 などの割り込みグループ モジュールが MSPM0 デバイ스에搭載されている場合があり、32 以上のペリフェラル割り込みを NVIC へ接続することが可能です。
- (2) デフォルトの優先度は、複数のマスク可能な割り込みに同じプログラム可能な優先度がある場合の相対的な割り込み優先度を示します。
- (3) NVIC 番号は、複数の NVIC 割り込みに同じプログラム可能な優先度がある場合の相対的な割り込み優先度を示します。
- (4) セクション 3.6.2 に、MSPM0 のイベントハンドラに関連する管理レジスタを示します。

### 3.6.1.1 RL78 の割り込み管理

RL78 デバイスは、PR0xy および PR1xy レジスタを使用して各割り込み状態の優先度を設定し、MKxy レジスタを使用して割り込み状態を有効または無効にします。たとえば、図 3-2 に、RL78 の内部マスク可能割り込み階層を示します。各割り込み要求がアクノリッジされる前に、IE = 1 を設定して割り込み要求のアクノリッジをイネーブルにするために、EI 命令を常に発行する必要があります。

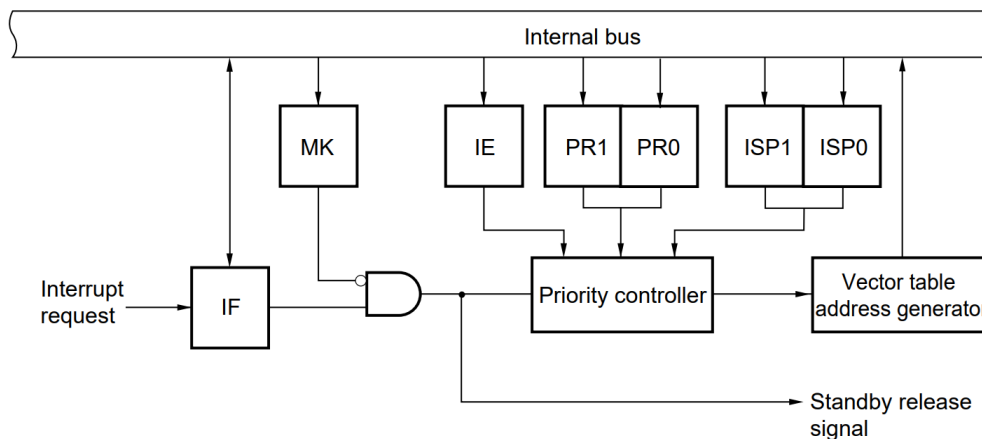


図 3-2. RL78 の内部マスク可能割り込み階層

### 3.6.1.2 MSPM0 の割り込み管理

MSPM0 デバイスでは、RL78 デバイスと異なり、NVIC の IPRx レジスタで各ペリフェラル割り込みソースの優先レベルを設定し、NVIC の ISER および ICER レジスタでペリフェラル割り込みソースをマスク/マスク解除します。各ペリフェラル割り込みには、各種の割り込み条件があります。たとえば、ペリフェラル割り込みソースとして、UARTx は送信割り込みや受信割り込みなど複数の割り込み条件を持っています。これらの割り込み条件は、ペリフェラル側の 6 つの標準レジスタで管理されています。

図 3-3 に、ペリフェラル割り込みの階層を示します。

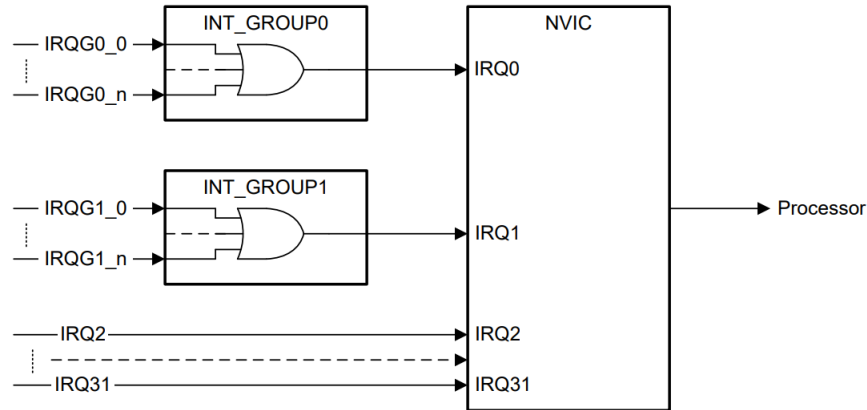


図 3-3. MSPM0 のペリフェラル割り込み階層

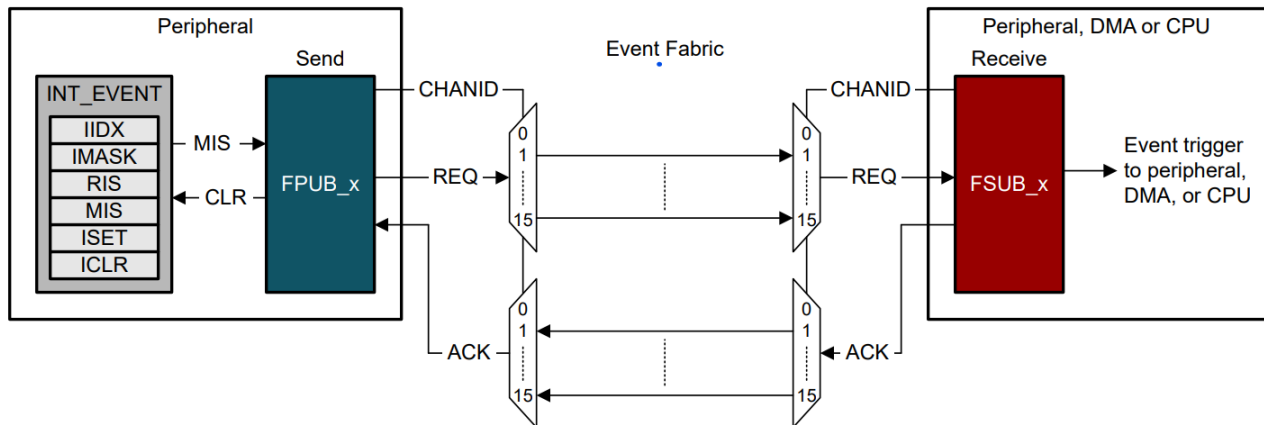
### 3.6.2 MSPM0 のイベントハンドラ

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 マネージャー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。また、イベント マネージャはパワー マネージメントおよびクロック ユニット (PMCU) とのハンドシェイクを実行し、トリガされたイベント アクションを実行するために必要なクロックと電力ドメインが存在することを確認することもできます。

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、次の 3 種類があります: CPU 割り込み (固定イベントルート)、DMA ルート、汎用ルート。例として、図 3-4 に汎用ルートを示します。



イベント マネージャのレジスタ セットには、次の 6 つの標準的なレジスタがあります: RIS、IMASK、MIS、ISET、ICLR、IIDX。イベントレジスタは相互接続されています (図 3-5 を参照)。マスクが解除されると、保留中の割り込みが RIS レジスタと MIS レジスタの両方に表示され、イベントが生成されます。CPU 割り込みイベントルートによる CPU 割り込みの場合、IIDX レジスタの読み出しにより、RIS レジスタと MIS レジスタの優先度が最も高い保留中の割り込みがクリアされ、その割り込みのインデックスがアプリケーションソフトウェアに返されます。

図 3-4. 汎用イベント ルート

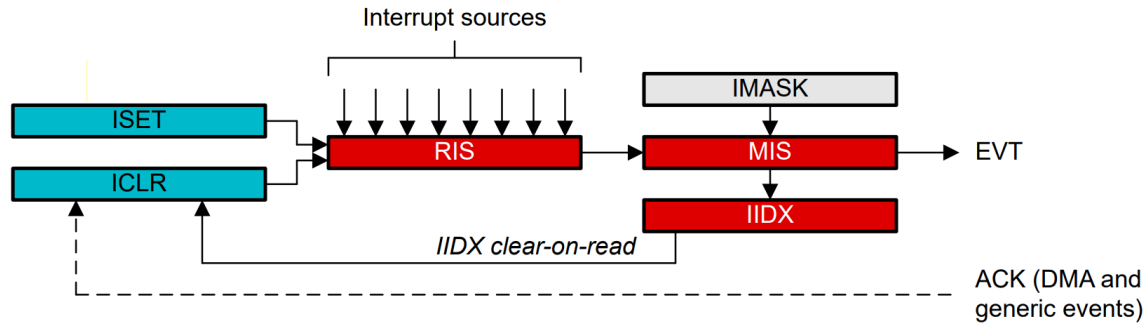


図 3-5. イベント管理の登録関係

図 3-6 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。MSPM0 で詳しいイベント ハンドラの使用方法については、[MSPM0G テクニカル リファレンス マニュアル](#)、または [MSPM0L テクニカル リファレンス マニュアル](#)、または [MSPM0C テクニカル リファレンス マニュアル](#) のイベントセクションを参照してください。

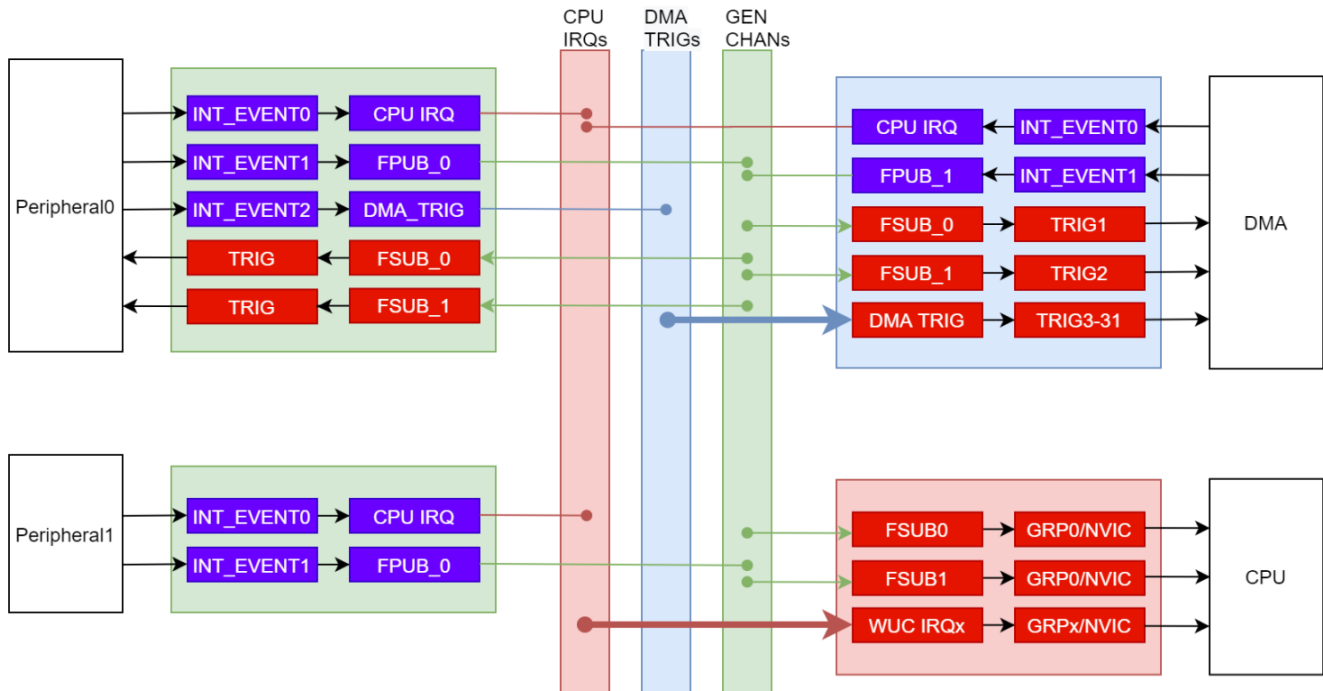


図 3-6. MSPM0 イベントおよび割り込み処理

### 3.6.3 RL78 Event Link Controller (ELC)

一部の RL78 マイコンには、各ペリフェラル機能からのイベント出力を相互接続 (リンク) するイベント・リンク・コントローラ (ELC) が備えられています。RL78 マイコンは、イベントをリンクさせることで、CPU を経由せずにペリフェラル機能間での動作を直接調整することができます。

図 3-7 に、ELC のブロック図を示します。ELSELRn レジスタ (一部のデバイスでは ELISELn および ELOSELn レジスタを使用) は、受信後のイベント受信ペリフェラル機能 (リンク宛先ペリフェラル機能) の動作に各イベント信号をリンクします。さまざまな ELSLRn レジスタはさまざまなイベントジェネレータを表し、ELSELRn レジスタに設定された値によって、リンク送信先ペリフェラル機能の動作が決定されます。

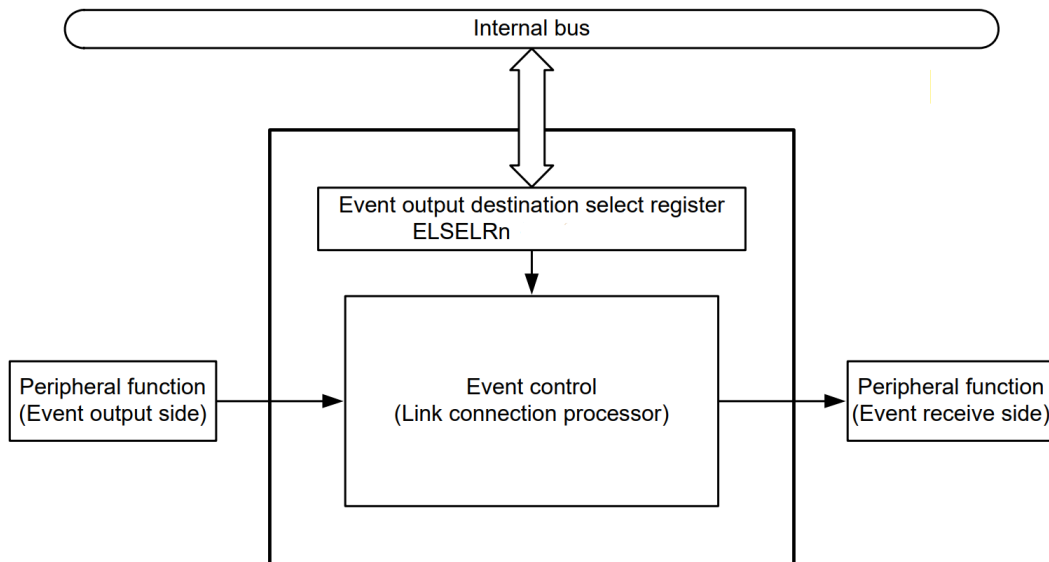


図 3-7. RL78 ELC (イベントリンクコントローラ)

図 3-8 に、割り込み処理と ELC の関係を示します。割り込み制御回路への割り込み要求として、ペリフェラル機能によって生成されたイベント信号を使用する経路は、この信号を ELC イベントとして使用する経路とは独立しています。したがって、割り込み制御に関係なく、各イベント信号をイベント受信ペリフェラル機能を動作させるためのイベント信号として使用できます。

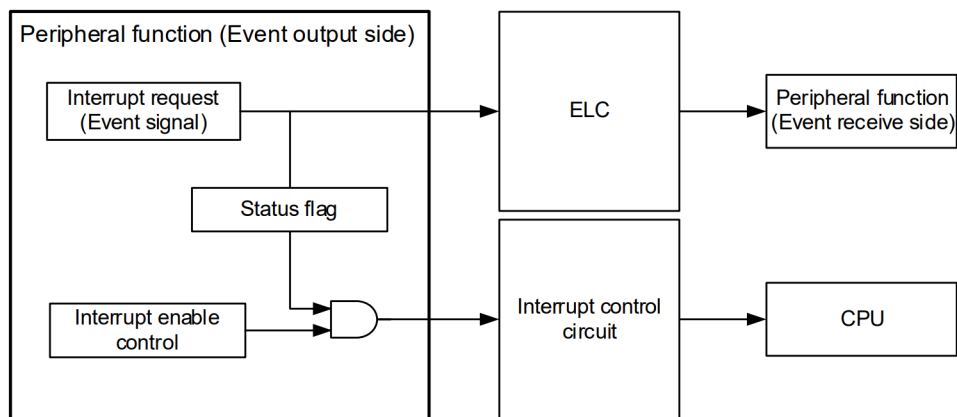


図 3-8. RL78 イベントおよび割り込み処理

### 3.6.4 イベント管理の比較

RL78 と MSPM0 では、イベント管理の構造や機能が異なります。セクション 3.6.2 と セクション 3.6.3 の比較を 表 3-12 に示します。

表 3-12. イベント管理の比較

| 特長     |                 | RL78                         | MSPM0                           |
|--------|-----------------|------------------------------|---------------------------------|
| パブリッシャ |                 | ペリフェラル                       | ペリフェラル                          |
| サブスライバ |                 | CPU、DMA トリガ、ペリフェラル           | CPU、DMA トリガ、ペリフェラル              |
| 管理スタイル | ペリフェラル → CPU    | 割り込み制御回路                     | イベント マネージャ                      |
|        | ペリフェラル → DMA    | DMA コントローラ <sup>(1)</sup>    |                                 |
|        | ペリフェラル → ペリフェラル | イベントリンクコントローラ <sup>(1)</sup> |                                 |
| ルートタイプ |                 | ポイント ツー ポイント (1:1)           | ポイント ツー ポイント                    |
|        |                 |                              | ポイントツーツー (スプリッタ) <sup>(2)</sup> |

- (1) すべての RL78 デバイスに DMA コントローラとイベントリンクコントローラの両方が搭載されているわけではなく、一部の RL78 デバイスには A と B のいずれも搭載されていません
- (2) 汎用ルートチャンネルは、選択されているチャンネルに応じて、1 つのサブスライバ (1:1) または 2 つのサブスライバ (1:2 スプリットルート) に設定できます。

## 3.7 デバッグとプログラミングの比較

### 3.7.1 デバッグの比較

Arm SWD 2 線式 JTAG ポートは、MSPM0 デバイスのメインのデバッグおよびプログラミング インターフェイスです。このインターフェイスは通常、アプリケーション開発時および量産プログラミング時に使用されます。

MSPM0 デバイスとは異なり、RL78 デバイスには SWD 2 線式 JTAG ポートがありません。RL78 デバイスは、デバッグとプログラミングに専用のフラッシュ メモリ プログラマ (PG-FP5/6、E1/E2/E2 Lite / E20 オンチップ デバッグ エミュレータ) を使用します。そして、プログラマと RL78 マイクロコントローラ間の通信は、RL78 マイクロコントローラの専用のシングルライン UART を介した TOOL0 ピンを使用したシリアル通信によって確立されます。

### 3.7.2 プログラミングモードの比較

#### 3.7.2.1 MSPM0 のブートストラップ ローダ (BSL) のプログラミング

ブートストラップ ローダ (BSL) プログラミング インターフェイスは、Arm SWD に対する代替プログラミング インターフェイスです。このインターフェイスはプログラミング機能のみを提供し、通常は標準の組込み通信インターフェイスを通して使用します。これにより、システム内または外部ポート内の他の組み込みデバイスへの既存の接続を経由して、ファームウェアを更新できます。プログラミングの更新はこのインターフェイスの主な目的ですが、BSL プログラミングインターフェイスは初期の量産プログラミングにも利用できます。

#### 3.7.2.2 RL78 のシリアル・プログラミング (外部デバイスを使用)

RL78 のシリアル・プログラミング・モードを使用すると、RL78 マイコンと、UART に接続した外部デバイス (マイコンまたは ASIC) を使用してファームウェアを更新できます。外部デバイスを使用して RL78 マイクロコントローラにデータを書き込み、または RL78 からデータを削除する処理はオンボードで実行されます。オフボードの書き込みはできません。表 3-13 に MSPM0 と RL78 の各デバイス ファミリのさまざまなプログラミング モードの比較を示します。

表 3-13. プログラミングモードの比較

| プログラミング機能 | RL78                                  | MSPM0                    |
|-----------|---------------------------------------|--------------------------|
| タイプ       | 外部デバイスを使用するシリアル プログラミング               | ブートストラップ ローダ             |
| セキュリティ    | メモリのセキュリティとアクセス制限オプション <sup>(1)</sup> | セキュア ブート オプション: CRC 保護機能 |
| カスタム化可能   | いいえ                                   | あり、ピン起動とプラグイン機能を構成可能     |
| メソッドを起動する | リセットリリース後、TOOL0 ピンは ハイ                | BOOTRST、SW エントリで 1 ピン ハイ |

表 3-13. プログラミングモードの比較 (続き)

| プログラミング機能         | RL78                             | MSPM0                                        |
|-------------------|----------------------------------|----------------------------------------------|
| コマンド              | 消去、書き込み、情報の取得、セキュリティなどです。<br>(1) | 接続、ロック解除、消去、書き込み、メモリの読み戻し、工場出荷時のリセット、情報取得など。 |
| パスワードによる保護        | いいえ                              | あり                                           |
| サポートされているインターフェイス | 専用 UART                          | UART、I2C、SPI (カスタム プラグインが必要)、CAN (プラグイン予定)   |

(1) ブロック消去、書き込みコマンド、および再書き込みブートクラスタは、デフォルト設定で有効になっています。ブロック消去の無効化、書き込みの無効化、再書き込みの無効化。ブートクラスタは、Security Set コマンドを使用して実行できます。

さらに、RL78 デバイスは、ユーザー・プログラム経由でフラッシュ・メモリを書き換えるために使用できる自己プログラミング・モードをサポートしています。このモードでは、ユーザーアプリケーションは RL78 マイクロコントローラの自己プログラミング ライブラリを使用してフラッシュメモリを書き換えることができます。

## 4 デジタル ペリフェラルの比較

### 4.1 汎用 I/O (GPIO、IOMUX)

MSPM0 GPIO 機能は、RL78G が提供するほぼすべての機能を網羅しています。RL78 では、ピン機能やポート機能という用語を使用します。この用語は、デバイスのピン管理や割り込みの生成など、ピンに関わるあらゆる機能を包括的に指します。MSPM0 GPIO および IOMUX の機能について以下に説明します：

- MSPM0 GPIO とは、IO の読み取りと書き込み、割り込みの生成などを実行できるハードウェアのことです。
- MSPM0 IOMUX とは、さまざまな内部デジタル ペリフェラルをピンに接続するために使用するハードウェアのことです。IOMUX は、GPIO など、多くの異なるデジタル ペリフェラルにサービスを提供しますが、これらに限定されるものではありません。

MSPM0 GPIO と IOMUX を組み合わせることで、RL78 ポート機能およびピン機能と同じ機能を実現できます。さらに、MSPM0 は、DMA 接続、制御可能な入力フィルタリング、イベント機能など、RL78 デバイスでは利用できない機能を提供します。

表 4-1. GPIO 機能の比較

| 機能                                                 | RL78                                                               | MSPM0                                                                        |
|----------------------------------------------------|--------------------------------------------------------------------|------------------------------------------------------------------------------|
| 出力モード                                              | プルアップ<br>N チャネル付きのオープンドレイン                                         | プルアップまたはプルダウン付きのプッシュプル<br>オープンドレイン、プルダウン<br>Hi-Z 付き                          |
| 入力モード                                              | プルアップ<br>入力スレッシュホールドレベル<br>CMOS または TTL 入力バッファ<br>アナログ             | フローティング<br>プルアップまたはプルダウン<br>アナログ                                             |
| GPIO 速度の選択                                         | いいえ                                                                | MSPM0 は、すべての IO ピンに標準 IO (SDIO) を提供する。<br>MSPM0 高速 IO (HSIO) は、選択されたピンで利用可能。 |
| High-drive GPIO                                    | ポートとチップタイプによっては、最大値は<br>VDD = 5V で約 56mA、VDD = 3.3V で<br>13.3mA です | VDD = 3.3V で約 20mA、High Drive IO (HDIO) と呼ばれる。                               |
| アトミック ビットのセットとリセット                                 | あり                                                                 | あり                                                                           |
| 代替機能                                               | PIOR レジスタを使用                                                       | IOMUX を使用                                                                    |
| 高速トグル                                              | いいえ                                                                | MSPM0 はクロック サイクルごとにピンを切り替えることが可能                                             |
| ウェークアップ                                            | いいえ                                                                | GPIO ピンの状態変更                                                                 |
| GPIO は DMA によって制御される                               | いいえ                                                                | MSPM0 でのみ利用可能                                                                |
| ユーザー制御の入力フィルタリングにより、1、3、8<br>ULPCLK 周期未満のグリッチを除去する | いいえ                                                                | MSPM0 でのみ利用可能                                                                |

表 4-1. GPIO 機能の比較 (続き)

| 機能                | RL78 | MSPM0         |
|-------------------|------|---------------|
| ユーザー制御可能な入力ヒステリシス | いいえ  | MSPM0 でのみ利用可能 |

GPIO サンプルコード: GPIO サンプル コードの詳細については、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

## 4.2 UART (Universal Asynchronous Receiver-Transmitter)

RL78 と MSPM0 はどちらも、非同期 (クロックレス) 通信を実行するためのペリフェラルを備えています。RL78 の UART では、UART として使用されるシリアルアレイユニット (SAU) 周辺機器、SAU の高度な機能、または LIN-UART、DALI/UART などのスタンドアロン UART 周辺機器が標準的な機能となっています。MSPM0 では、標準機能には **main** という名前が付けられ、高度な機能には **extend** という名前が付けられています。また、MSPM0 は、IrDA ハードウェアサポート、スマートカードモード、ハードウェアフロー制御など、RL78 デバイスでは利用できない機能を提供します。

表 4-2. UART 機能の比較

| 機能                      | RL78                      | MSPM0                    |
|-------------------------|---------------------------|--------------------------|
| データドリフト                 | MSB または LSE               | 同等                       |
| DMA を使用した連続通信           | あり                        | あり                       |
| データ フェーズ                | あり (逆出力)                  | いいえ                      |
| スヌーズ モード時のレセプション        | あり                        | あり、すべての低消費電力モードでアクティブです。 |
| 1 線式半二重通信               | あり                        | あり <sup>(1)</sup>        |
| データ長                    | 5、7、8、9、16 <sup>(2)</sup> | 5、6、7、8                  |
| LIN ハードウェアのサポート         | あり                        | あり                       |
| DALI ハードウェアのサポート        | あり                        | あり                       |
| IrDA ハードウェア サポート        | いいえ                       | あり                       |
| マンチェスター コードのハードウェア サポート | いいえ                       | あり                       |
| スマート カード モード (ISO7816)  | いいえ                       | あり                       |
| 低消費電力モードからのウェークアップ      | あり (LIN)                  | あり                       |
| 自動ボーレート検出               | あり (LIN)                  | いいえ                      |
| 外部ドライバ イネーブル            | いいえ                       | あり                       |
| ハードウェア フロー制御            | いいえ                       | あり                       |
| マルチプロセッサ                | いいえ                       | あり                       |
| Tx/Rx FIFO の深度          | いいえ                       | 4                        |

(1) 伝送と受信の間でペリフェラルを再構成する必要があります

(2) UARTax ではデータ長 5 ビット、UARTx では 9 ビットが使用可能で、16 ビットは一部のデバイスでのみ使用できます。

### UART サンプル コード

UART サンプル コードの詳細については、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

## 4.3 シリアル・ペリフェラル・インターフェイス (SPI)

MSPM0 と RL78 はどちらも、シリアル ペリフェラル インターフェイス (SPI) をサポートしています。RL78 SPI 機能は、CSI (クロック・シリアル・インターフェイス) として使用される SAU (シリアル・アレイ・ユニット) のペリフェラルにあります。それ以外に、MSPM0 は、コントローラとペリフェラルを使用して、SPI の通信当事者を代表します。全体として、MSPM0 および RL78 SPI のサポート内容はおおむね同等ですが、違いについて表 4-3 に示します。

### 注

RL78 では、デバイスごとに異なる SPI サポートレベルを提供します。これらは SPI および簡略化された SPI と呼ばれます。

**表 4-3. SPI 機能の比較**

| 機能                                             | RL78                                                                   | MSPM0                            |
|------------------------------------------------|------------------------------------------------------------------------|----------------------------------|
| 操作ワイヤ                                          | SCK, SI, SO                                                            | SCLK, PICO, POCI, CSx            |
| コントローラまたはペリフェラルの動作                             | あり                                                                     | あり                               |
| データ ビット幅 (コントローラ モード)                          | 7～16 ビット <sup>(1)</sup>                                                | 4～16 ビット                         |
| データ ビット幅 (ペリフェラル モード)                          |                                                                        | 7～16 ビット                         |
| 最大速度                                           | 16MHz (CSI00 のみ、コントローラ)<br>8MHz (ELSE、コントローラ)<br>5.33MHz (ELSE、ペリフェラル) | 最高 32MHz                         |
| シンプレックス通信 (単方向データ ライン)                         | あり                                                                     | あり                               |
| ハードウェア チップ セレクト マネージメント                        | いいえ                                                                    | あり (4 つのペリフェラル)                  |
| I/O クロックの位相制御                                  | あり                                                                     | あり                               |
| MSB ファーストまたは LSB ファーストのシフトによるトランスファー・ディレクション設定 | あり                                                                     | あり                               |
| SPI フォーマットのサポート                                | Motorola                                                               | Motorola、テキサス・インスツルメンツ、MICROWIRE |
| ハードウェア CRC                                     | いいえ                                                                    | なし、MSPM0 は SPI パリティ モードを備えています   |
| TX FIFO の深さ                                    | いいえ                                                                    | 4                                |
| RX FIFO の深さ                                    | いいえ                                                                    | 4                                |

(1) 対応するデータビット長は、デバイスと動作モードによって異なります。

SPI サンプルコード: SPI サンプル コードの詳細については、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

#### 4.4 I2C (Inter-Integrated Circuit)

MSPM0 と RL78 はどちらも I2C ペリフェラルをサポートしています。RL78 の I2C では、シリアルアレイユニット (SAU) ペリフェラルで基本機能が提供されています。単純化された I2C として使用され、高度な機能は IICA ペリフェラルで提供されています。MSPM0 の場合、I2C では基本機能と高度な機能の両方が I2C ペリフェラルに搭載されています。MSPM0 は、コントローラとターゲットを使用して、通信の両側を代表します。全体として、MSPM0 および RL78 I2C のサポート内容はおおむね同等ですが、重要な違いについて [表 4-4](#) に示します。

**表 4-4. I2C 機能の比較**

| 機能                           | RL78      | MSPM0          |
|------------------------------|-----------|----------------|
| コントローラ モードとターゲット モード         | あり (IICA) | あり             |
| マルチ コントローラの機能                | あり (IICA) | あり             |
| 最大データ転送レート                   | 1MHz      | 同等 (高速モード プラス) |
| アドレッシング モード                  | 7 ビット     | 7 ビット          |
| アドレス番号 (ターゲット モード)           | 1 つのアドレス  | 2 つのアドレス       |
| イベント管理                       | いいえ       | あり             |
| クロック ストレッチ                   | あり        | あり             |
| ウェークアップ機能 (低消費電力モード)         | あり        | あり             |
| ソフトウェア・リセット                  | あり        | あり             |
| FIFO / バッファ                  | 1 バイト     | TX: 8 バイト      |
|                              |           | RX: 8 バイト      |
| DMA                          | いいえ       | あり             |
| プログラム可能なアナログおよびデジタル ノイズ フィルタ | いいえ       | あり             |

I2C サンプルコード: I2C サンプル コードの詳細については、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

## 4.5 タイマ (TIMGx、TIMAx)

RL78 と MSPM0 はどちらも、さまざまなタイマを搭載しています。MSPM0 は低消費電力監視から高度なモーター制御までのユースケースをサポートする、さまざまな機能を備えたタイマを提供しています。

表 4-5. タイマの命名

| RL78    |                    | MSPM0 |          |
|---------|--------------------|-------|----------|
| タイマ名    | 略称                 | タイマ名  | 略称       |
|         |                    | 高度な制御 | TIMA0    |
| 汎用      | タイマ RJ、RD、RG、RX    | 汎用    | TIMG0-11 |
| 時間間隔の生成 | インターバルタイマ          | 高解像度  | TIMG12   |
| 基本      | タイマ アレイ ユニット (TAU) |       |          |

表 4-6. タイマ機能の比較

| 機能             | RL78 タイマ       | MSPM0         |
|----------------|----------------|---------------|
| 分解能            | 8、12、16、32 ビット | 16 ビット、32 ビット |
| PWM            | あり             | あり            |
| キャプチャ          | あり             | あり            |
| 比較             | あり             | あり            |
| ワンショット         | あり             | あり            |
| アップ ダウン カウント機能 | あり             | あり            |
| 電力モード          | あり             | あり            |
| QEI のサポート      | いいえ            | あり            |
| プログラマブル プリスケアラ | あり             | あり            |
| シャドウ レジスタ モード  | いいえ            | あり            |
| イベント / 割り込み    | あり             | あり            |
| 自動リロード機能       | あり             | あり            |
| フォルト処理         | いいえ            | あり            |

表 4-7. タイマ モジュールの代替品

| RL78 タイマ        | MSPM0 同等品     | 推論                 |
|-----------------|---------------|--------------------|
| TAU             | 任意            | 汎用、16 ビット分解能       |
| 32 ビットインターバルタイマ | TIMG12        | 32 ビット分解能          |
| タイマ RJ、RD、RG、RX | TIMA、TIMG0-11 | 高度な目的、PWM、キャプチャ、比較 |

表 4-8. タイマの使用事例の比較

| 機能     | RL78 タイマ     | MSPM0 タイマ                             |
|--------|--------------|---------------------------------------|
| PWM    | タイマ RJ、RD、RG | すべてのタイマには、エッジ アラインまたはセンタアラインのオプションがある |
| キャプチャ  | タイマ RD、RG、RX | すべてのタイマ                               |
| 比較     | タイマ RD       | すべてのタイマ                               |
| ワンショット | すべてのタイマ      | すべてのタイマ                               |
| プリスケアラ | 4 ビット プリスケアラ | 8 ビット プリスケアラ                          |
| 同期     | TAU          | すべてのタイマにこの機能がある                       |

### タイマ サンプル コード

タイマ サンプル コードの詳細については、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

## 4.6 ウィンドウ付きウォッチドッグ タイマ (WWDT)

RL78 と MSPM0 はどちらもウィンドウ ウォッチドッグ タイマを備えています。ウィンドウ ウォッチドッグ タイマ (WWDT) は、指定された時間内にアプリケーションがチェックインに失敗した場合、システム リセットを開始します。

表 4-9. WWDT の命名法

| RL78              | MSPM0                     |
|-------------------|---------------------------|
| ウォッチドッグ タイマ (WDT) | ウィンドウ付きウォッチドッグ タイマ (WWDT) |

表 4-10. WDT 機能の比較

| 機能             | RL78   | MSPM0G |
|----------------|--------|--------|
| ウィンドウ モード      | あり     | あり     |
| インターバル タイマ モード | いいえ    | あり     |
| LFCLK 電源       | あり     | あり     |
| 割り込み           | あり     | あり     |
| カウンタの分解能       | 17 ビット | 25 ビット |
| クロック デバイダ      | いいえ    | あり     |

### WWDT サンプル コード

WWDT サンプル コードの詳細については、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

## 4.7 リアルタイム クロック (RTC)

RL78 および MSPM0<sup>1</sup> は、どちらもリアルタイム クロック (RTC) を提供しています。リアルタイム クロック (RTC) モジュールは、選択可能な 2 進数または 2 進化 10 進数形式での、秒、分、時間、曜日、日、年のカウンタを使用することで、アプリケーションのタイムトラッキング機能を提供します。

RTC は、MSPM0G シリーズ、MSPM0x22x、MSPM0C1105/6、MSPM0H シリーズに対応しています。

表 4-11. RTC 機能の比較

| 機能              | RL78 | MSPM0 |
|-----------------|------|-------|
| 電力モード           | あり   | あり    |
| 2 進数符号化形式       | いいえ  | あり    |
| 何年も経った          | 99 年 | 199 年 |
| うるう年補正          | あり   | あり    |
| カスタマイズ可能なアラームの数 | 1    | 2     |
| 内部および外部の水晶振動子   | あり   | あり    |
| オフセット キャリブレーション | あり   | あり    |
| 割り込み            | あり   | あり    |

### RTC サンプル コード

RTC サンプル コードの詳細については、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

<sup>1</sup> RTC をサポートしているのは MSPM0G デバイスのみです。

## 5 アナログ ペリフェラルの比較

### 5.1 A/D コンバータ (ADC)

RL78 と MSPM0 はどちらも、アナログ信号を同等のデジタル信号に変換するための ADC ペリフェラルを搭載しています。どちらのデバイス ファミリーも、12 ビット ADC を搭載しています。RL78 I シリーズには 24 ビットのデルタ・シグマ ADC が搭載されていますが、ここでは扱いません。MSP430F676x を検討できます。表 5-1 と 表 5-2 に、ADC のさまざまな機能とモードの比較を示します。

ADC 外部リファレンスは、MSPM0C1103 および MSPM0C1104 のデバイスではサポートされていません。

表 5-1. 機能セットの比較

| 機能                      | RL78                                                                                   | MSPM0                                                                |
|-------------------------|----------------------------------------------------------------------------------------|----------------------------------------------------------------------|
| 分解能 (ビット)               | 12、10、8                                                                                | 12、10、8                                                              |
| 変換レート (MSPS)            | 0.888                                                                                  | 4                                                                    |
| オーバーサンプリング (ビット)        | 該当なし                                                                                   | 14                                                                   |
| ハードウェア オーバーサンプリング       | 16x                                                                                    | 128x                                                                 |
| FIFO                    | いいえ                                                                                    | あり                                                                   |
| ADC 基準電圧 (V)            | 内部: 1.48、VDD                                                                           | 内部: 1.4、2.5、VDD                                                      |
|                         | 外部:<br>$2.4 \leq V_{REFP} \leq V_{DD} \leq 5.5V$<br>$V_{REFM} = V_{SS}$ または $V_{REFM}$ | 外部:<br>$1.4 \leq V_{REF} \leq V_{DD}$                                |
| 動作時電力モード                | 動作、スリープ                                                                                | 動作、スリープ、停止、スタンバイ <sup>(1)</sup>                                      |
| 自動パワーダウン                | あり                                                                                     | あり                                                                   |
| 外部入力チャネル <sup>(2)</sup> | 最大 31                                                                                  | MSPM0G: 最大 16 個の MSPM0L: 最大 16 個<br>の MSPM0C: 最大 27 個の MSPM0H: 最大 27 |
| 内部入力チャネル                | 温度センサ、内部基準電圧、タッチ センサ キャパシタンス                                                           | 温度センサ、電源監視、アナログ シグナル チェーン                                            |
| DMA サポート                | あり (DTC/DMA)                                                                           | あり (DMA)                                                             |
| ADC ウィンドウ コンパレータ ユニット   | あり (ADxL)                                                                              | あり                                                                   |
| 同時サンプリング                | いいえ                                                                                    | あり                                                                   |
| ADC の数 <sup>(3)</sup>   | 最大 1                                                                                   | MSPM0G: 最大 2 つの MSPM0L/C/H: 最大 1                                     |

(1) ADC はスタンバイ モードでトリガできるため、動作モードが変化します。

(2) 外部入力チャネルの数は、デバイスごとに異なります。

(3) ADC の数はデバイスごとに異なります。

表 5-2. 変換モード

| RL78 <sup>(1)</sup>                   | MSPM0               | 備考                                          |
|---------------------------------------|---------------------|---------------------------------------------|
| モードのうち、ワンショット変換モード/シングルスキャンモードから選択します | シングル チャネル シングル変換    | ADC は、1 つのチャネルを 1 回サンプリングして変換します            |
| スキャンモードのうち、ワンショット変換モード/シングルスキャンモード    | チャネル変換のシーケンス        | ADC は一連のチャネルをサンプリングし、1 回変換します。              |
| モードのうち、シーケンシャル変換モード、連続スキャンモードから選択します  | シングル チャネル変換を繰り返します  | 1 つのチャネルを連続的にサンプリングし、1 つのチャネルを変換することを繰り返します |
| スキャン・モード、シーケンシャル変換モード、連続スキャン・モード      | チャネル変換のシーケンスを繰り返します | 一連のチャネルをサンプリングして変換した後、同じシーケンスを繰り返します        |

(1) RL78 ADC 変換モードの名称はデバイスごとに異なります。ここでは、「/」分割を使用する 2 種類の名称を使用しています。

### ADC サンプル コード

ADC サンプル コードの詳細については、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

## 5.2 コンパレータ (COMP)

RL78 および MSPM0 ファミリの部品はどちらも、一部のデバイスでオプションのペリフェラルとしてコンパレータを内蔵しています。RL78 では、コンパレータは **CMP**、**COMP**、コンパレータ **x** と表記され、MSPM0 では **COMPx** と表記されます。G1F ファミリでは、これらに 0~1 の番号が付けられ、MSPM0 ファミリでは 0~2 の番号が付けられています。RL78 ファミリでは、G1F シリーズは BLDC モータに使用され、高度な機能コンパレータを備えています。他のシリーズは主に基本機能コンパレータを備えています。コンパレータ モジュールは、さまざまな内部および外部ソースを入力として受け取ることができ、電源モードの切り替えや PWM 信号の制御などに使用できます。MSPM0 および RL78 コンパレータ モジュールの機能別の比較については [表 5-3](#) を参照してください。

MSPM0H シリーズ、ならびに MSPM0C 1103、MSPM0C 1104 のデバイスは COMP をサポートしていません。

**表 5-3. COMP 機能セットの比較**

| 機能                     | RL78                  | MSPM0G                                                             | MSPM0L                                                     | MSPM0C                                                     |
|------------------------|-----------------------|--------------------------------------------------------------------|------------------------------------------------------------|------------------------------------------------------------|
| 使用可能なコンパレータ            | 最大 2                  | 最大 3                                                               | 最大 1                                                       | 最大 1                                                       |
| 出力のルーティング              | 外部<br>イベントリンカ コントローラ  | 多重化された I/O ピン<br>割り込みまたはイベント インターフェイス                              | 多重化された I/O ピン<br>割り込みまたはイベント インターフェイス                      | 多重化された I/O ピン<br>割り込みまたはイベント インターフェイス                      |
| 正入力                    | 外部 4 アナログ・ピン入力        | 多重化された I/O ピン                                                      | 多重化された I/O ピン                                              | 多重化された I/O ピン                                              |
|                        |                       | DAC12 出力 <sup>(1)</sup>                                            | DAC8 出力                                                    | DAC8 出力                                                    |
|                        |                       | DAC8 出力                                                            |                                                            |                                                            |
|                        | コンパレータ 0 の PGA 出力     | 内部 $V_{REF}$ : 1.4V および 2.5V                                       | /                                                          | /                                                          |
| 負入力                    | 外部アナログ ピン入力           | 多重化された I/O ピン                                                      | 多重化された I/O ピン                                              | 多重化された I/O ピン                                              |
|                        |                       | 内部温度センサ                                                            | 内部温度センサ                                                    | 内部温度センサ                                                    |
|                        | 内部 $V_{REF}$ : 1.45V  | 内部 $V_{REF}$ : 1.4V および 2.5V                                       | DAC8 出力                                                    | DAC8 出力                                                    |
|                        | 8 ビット DAC<br>コンパレータ 1 | DAC8 出力                                                            | OPA0 <sup>(3)</sup> 出力                                     | OPA0 <sup>(3)</sup> 出力                                     |
|                        |                       | OPA0 出力 <sup>(3)</sup>                                             |                                                            |                                                            |
| プログラマブル ヒステリシス         | なし、10mV、20mV、30mV     | なし、10mV、20mV、30mV<br>DAC8 を使用するその他の値は、 $0V \sim V_{REF}/V_{DD}$ です | なし、10mV、20mV、30mV<br>DAC8 を使用するその他の値は、 $0V \sim V_{DD}$ です | なし、10mV、20mV、30mV<br>DAC8 を使用するその他の値は、 $0V \sim V_{DD}$ です |
| レジスタ ロック               | いいえ                   | あり、COMP レジスタの一部 (書き込みにはキーが必要)                                      | あり、COMP レジスタの一部 (書き込みにはキーが必要)                              | あり、COMP レジスタの一部 (書き込みにはキーが必要)                              |
| ウィンドウ コンパレータの構成        | TAU0 のタイマ ウィンドウ       | あり                                                                 | なし (シングル COMP)                                             | なし (シングル COMP)                                             |
| 入力短絡モード                | いいえ                   | あり                                                                 | あり                                                         | あり                                                         |
| 動作モード                  | 実行                    | 高速、低消費電力                                                           | 高速、低消費電力                                                   | 高速、低消費電力                                                   |
| PWM の高速シャットダウン         | あり                    | あり (TIMA フォルト ハンドラを使用)                                             | いいえ                                                        | いいえ                                                        |
| 出力フィルタリング              | 除去デジタル・フィルタ (3 サイクル)  | ブランキング フィルタ                                                        | ブランキング フィルタ                                                | ブランキング フィルタ                                                |
|                        |                       | 調整可能なアナログ フィルタ                                                     | 調整可能なアナログ フィルタ                                             | 調整可能なアナログ フィルタ                                             |
| 出力極性の制御                | あり                    | あり                                                                 | あり                                                         | あり                                                         |
| 割り込み                   | 立ち上がりエッジ              | 立ち上がりエッジ                                                           | 立ち上がりエッジ                                                   | 立ち上がりエッジ                                                   |
|                        | 立ち下がりエッジ              | 立ち下がりエッジ                                                           | 立ち下がりエッジ                                                   | 立ち下がりエッジ                                                   |
|                        | 両方のエッジ                | 出力準備完了                                                             | 出力準備完了                                                     | 出力準備完了                                                     |
| 入力交換モード <sup>(4)</sup> | いいえ                   | あり                                                                 | あり                                                         | あり                                                         |

(1) DAC12 ペリフェラルを搭載したデバイスのみ

(2) OPA1 ペリフェラルを搭載したデバイスのみ

(3) OPA0 ペリフェラルを搭載したデバイスのみ

- (4) 入力切り替えモードを有効にすると、コンパレータの正入力端子と負入力端子の信号が入れ替わります。さらに、コンパレータからの出力信号も反転されます。

COMP サンプルコード:COMP サンプル コードの詳細については、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

### 5.3 D/A コンバータ (DAC)

RL78 および MSPM0 ファミリの部品はどちらも 12 ビットまたは 8 ビットの DAC ペリフェラルを搭載しており、さまざまなアプリケーションで D/A 変換を実行できます。RL78 の資料では、12 ビット DAC は 12 ビット D/A コンバータと呼ばれ、8 ビット DAC は DAC、D/A コンバータと呼ばれています。12 ビット D/A コンバータは、RL78 I1E および L1A ファミリのデバイスでのみ利用できます。MSPM0 では、12 ビット DAC ペリフェラルを DAC12 と呼びます。これにより、DAC12 は、特定の MSPM0 デバイスに搭載されている各コンパレータ ペリフェラルで利用できる 8 ビット DAC と差別化します。追加の 8 ビット DAC については、このドキュメントのコンパレータのセクションで説明します。この DAC12 ペリフェラルは、MSPM0G ファミリのデバイスでのみ利用できます。

RL78 および MSPM0G の 12 ビット DAC ペリフェラルの機能を表 5-4 にまとめます。

MSPM0L と C DAC は COMP に内蔵されています。MSPM0H シリーズと MSPM0C1103、MSPM0C1104 にはコンパレータがないため、DAC はサポートされていません。

表 5-4. DAC 機能セットの比較

| 機能            | RL78                  | MSPM0G                                                                                           | MSPM0L/C                     |
|---------------|-----------------------|--------------------------------------------------------------------------------------------------|------------------------------|
| 分解能           | 12 ビット                | 12 ビット (11ENOB)                                                                                  | 8 ビット                        |
| 出力レート         | 33kSPS                | 1MSPS                                                                                            | 1MSPS                        |
| 出力チャンネル数      | 2 <sup>(1)</sup>      | 1 <sup>(2)</sup>                                                                                 | 1                            |
| データ形式         | 12 ビットの右揃え、12 ビットの左揃え | 8 ビットの右揃え、12 ビットの右揃え、2 の補数、またはストレート バイナリ                                                         | 8 ビットの右揃え                    |
| DMA 連携        | あり (DTC)              | あり                                                                                               | あり                           |
| 出力のルーティング     | 外部ピン                  | 外部ピン<br>内部ペリフェラル接続: OPA IN+、COMP IN+、ADC0                                                        | 内部ペリフェラル接続: OPA IN+、COMP IN+ |
| 内部リファレンス電圧    | はい、1.48V              | あり、2.5V または 1.4V                                                                                 | あり、2.5V または 1.4V             |
| 外部基準電圧        | あり                    | あり                                                                                               | いいえ                          |
| FIFO          | いいえ                   | あり                                                                                               | いいえ                          |
| 出力バッファ        | いいえ                   | あり                                                                                               | いいえ                          |
| 出力オフセットの設定が可能 | いいえ                   | あり                                                                                               | いいえ                          |
| 自己校正モード       | いいえ                   | あり                                                                                               | いいえ                          |
| トリガ ソース       | イベントリンク               | 内部専用のサンプル タイム ジェネレータ、DMA 割り込み / イベント、FIFO スレッシュホールド割り込み / イベント、2 つのハードウェアトリガ (イベント ファブリックから利用可能) | イベント                         |

- (1) 12 ビットの分解能を持つ L1A デバイスでのみ使用可能です。  
(2) デュアル DAC チャンネルは、将来の MSPM0G デバイス用に計画されています。

DAC12 サンプルコード:DAC12 サンプル コードの詳細については、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

## 5.4 オペアンプ (OPA)

RL78 ファミリのデバイスは、異なるタイプのアンプを備えた一連のデバイスを提供します。RL78 には 3 つの OPA と 1 つの PGA を備えた PGA とアンプ ユニットがあります。MSPM0 OPA モジュールは非常に柔軟性が高く、センシングまたは制御アプリケーションでは、個別に、または組み合わせて、多くのディスクリート アンプを置き換えることができます。MSPM0 OPA モジュールと RL78 アンプ ユニットの違いを [表 5-5](#) に示します。

OPA は MSPM0L1306 と MSPM0G3507 のみがサポートしています。MSPM0L1345 には TIA があります。

**表 5-5. OPA 機能セットの比較**

| 機能           | RL78                     | MSPM0                     |
|--------------|--------------------------|---------------------------|
| GBW (低電力モード) | 1.5kHz (PGA)、60kHz (OPA) | 1MHz (低消費電力モード)           |
| GBW (標準モード)  | 67kHz (PGA)、1.7MHz (OPA) | 6MHz (標準モード)              |
| アンプの構成またはタイプ | 汎用アンプ                    | 汎用モード                     |
|              | バッファ モード                 | 同等                        |
|              | PGA ペリフェラル               | PGA モード (反転または非反転)        |
|              | 差動アンプモード                 | 同等                        |
|              | いいえ                      | カスケード アンプ モード             |
| 入力 / 出力の配線   | 外部ピンの配線                  | 外部ピンの配線                   |
|              | DAC および ADC への内部接続       | ADC および COMP モジュールへの内部接続  |
| フォルト検出       | いいえ                      | バーンアウト電流源 (BCS)           |
| チョッパ安定化      | いいえ                      | 標準 (チョッピング周波数を選択可能)       |
|              |                          | ADC 支援チョップ                |
|              |                          | ディセーブル                    |
| 基準電圧         | 内部バイアス電圧 1V              | 内部 VREF (MSPM0G デバイスのみ)   |
|              | DAC12 (H1D)              | DAC12 (MSPM0G デバイスのみ)     |
|              | DAC8 (H1D)               | DAC8 (COMP モジュール搭載デバイスのみ) |

OPA サンプルコード: OPA サンプル コードの詳細については、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

## 5.5 基準電圧 (VREF)

RL78 と MSPM0 はどちらも内部基準電圧を備えており、内部ペリフェラルに基準電圧を供給したり、また、MSPM0 のみが内部基準電圧を外部ペリフェラルに出力したりできます。

**表 5-6. VREF 機能セットの比較**

| 機能          | RL78                          | MSPM0G                             | MSPM0L、MSPM0C、MSPM0H                              |
|-------------|-------------------------------|------------------------------------|---------------------------------------------------|
| 内部基準電圧 (V)  | 1.45、SBIAS                    | 1.4、2.5                            | 1.4、2.5                                           |
| 外部基準電圧 (V)  | $2.4 \leq V_{REFP} \leq 5.5V$ | 外部: $1.4 \leq V_{REF} \leq V_{DD}$ | 外部: $1.4 \leq V_{REF} \leq V_{DD}$ <sup>(1)</sup> |
| 内部基準電圧の出力   | あり                            | あり                                 | あり                                                |
| ADC への内部接続  | あり                            | あり                                 | あり                                                |
| DAC への内部接続  | あり                            | あり                                 | いいえ                                               |
| COMP への内部接続 | あり                            | あり                                 | いいえ                                               |
| OPA への内部接続  | あり                            | あり                                 | いいえ                                               |

(1) MSPM0C1103 と MSPM0C1104 は、外部基準電圧に対応していません。

MSPM0 VREF の場合、パワー ビット、PWREN Bit0 (ENABLE) をイネーブルにします。

VREF サンプルコード: VREF を使用したサンプル コードについては、『[MSPM0 SDK サンプル ガイド](#)』を参照してください。

## 6 まとめ

この記事は、RL78 デバイスから MSPM0 へのより良い移行を実現するのに役立ちます。そのために、エコシステム、CPU、アナログ、デジタル ペリフェラルという 2 つの間の違いを比較してみます。また、このアプリケーション ノートでは実例を交えて、MSPM0 をすぐに始めて開発の進行をスピード アップできるように、詳しく説明しています。

## 7 参考資料

- テキサス・インスツルメンツ、『[MSPM0 SDK ユーザー ガイド](#)』
- テキサス・インスツルメンツ、『[MSPM0 ツール ガイド](#)』
- テキサス・インスツルメンツ、『[Driverlib API ガイド](#)』
- テキサス・インスツルメンツ、『[Code Composer Studio \(CCS\)](#)』
- テキサス・インスツルメンツ、『[MSPM0 L シリーズ 32MHz マイコン技術参照マニュアル](#)』、技術参照マニュアル
- テキサス・インスツルメンツ、『[MSPM0 C シリーズ 24MHz マイコン](#)』、技術参照マニュアル
- IAR、[IAR ウェブサイト](#)
- Keil、[Keil ウェブサイト](#)
- テキサス・インスツルメンツ、『[CCS クイック スタート ガイド](#)』
- テキサス・インスツルメンツ、『[CCS トレーニング動画](#)』
- テキサス・インスツルメンツ、『[CCS ユーザーズ ガイド](#)』
- テキサス・インスツルメンツ、『[IAR クイック スタート ガイド](#)』
- IAR、[IAR トレーニング動画](#)
- IAR、[IAR ユーザーズ ガイド](#)
- Keil、[Keil クイック スタート ガイド](#)
- Keil、[Keil トレーニング動画](#)
- Keil、[Keil を始める](#)
- テキサス・インスツルメンツ、『[SYSCONFIG IDE](#)』
- テキサス・インスツルメンツ、『[MSPM0 SysConfig ガイド](#)』
- テキサス・インスツルメンツ、『[XDS110 デバッグプローブ](#)』
- テキサス・インスツルメンツ、『[J-Link デバッグプローブのページ](#)』
- テキサス・インスツルメンツ、『[LP-MSPM0G3507 LaunchPad 開発キット](#)』
- テキサス・インスツルメンツ、『[LP-MSPM0L1306 LaunchPad 開発キット](#)』
- テキサス・インスツルメンツ、『[LP-MSPM0C1104 LaunchPad 開発キット](#)』

## 8 改訂履歴

資料番号末尾の英字は改訂を表しています。その改訂履歴は英語版に準じています。

| Changes from Revision * (November 2023) to Revision A (May 2025) | Page |
|------------------------------------------------------------------|------|
| • ドキュメント全体を通して MSPM0H シリーズの仕様を更新.....                            | 4    |
| • ドキュメント全体を通して RL78 デバイスとの比較対象を MSPM0H シリーズに更新.....              | 4    |
| • IDE のバージョンや MSP フラッシュツールなど、新しいエコシステムを更新.....                   | 6    |

## 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

## 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