



摘要

MSP430™ 器件作为 TI 的经典微控制器拥有近 30 年的历史。许多客户在不同的项目上使用了 MSP430 MCU，并且如今仍在使用它们。新一代 MSPM0 产品采用 Arm® Cortex®-M0+ 内核，该内核具有更多功能丰富的外设资源和更小的封装。随着新产品的开发或旧产品的升级，您可能需要使用更新的 MSP 器件。本应用手册介绍了从 MSP430 MCU 到 MSPM0 MCU 的软件迁移。

内容

1 软件移植流程.....	2
2 开发环境.....	3
2.1 集成开发环境 (IDE).....	3
2.2 软件生态系统.....	3
2.3 用于 MSPM0 MCU 的 SysConfig.....	5
2.4 MSP430 和 MSPM0 工程.....	7
2.5 调试器接口.....	8
3 迁移注意事项.....	12
3.1 外设.....	12
3.2 系统时钟.....	12
3.3 运行模式.....	13
3.4 非易失性存储器 (NVM).....	14
3.5 事件和中断处理.....	14
3.6 复位级别.....	15
3.7 GPIO 和引脚多路复用.....	16
3.8 通信接口.....	17
3.9 引导加载程序 (BSL).....	18
3.10 模拟外设.....	20
3.11 计时器.....	23
3.12 硬件设计指南.....	23
4 修订历史记录.....	23

商标

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

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

所有商标均为其各自所有者的财产。

1 软件移植流程

图 1-1 展示了从 MSP430 到 MSPM0 MCU 的软件迁移步骤。以下各节介绍了开发环境、软件和外设之间的差异。

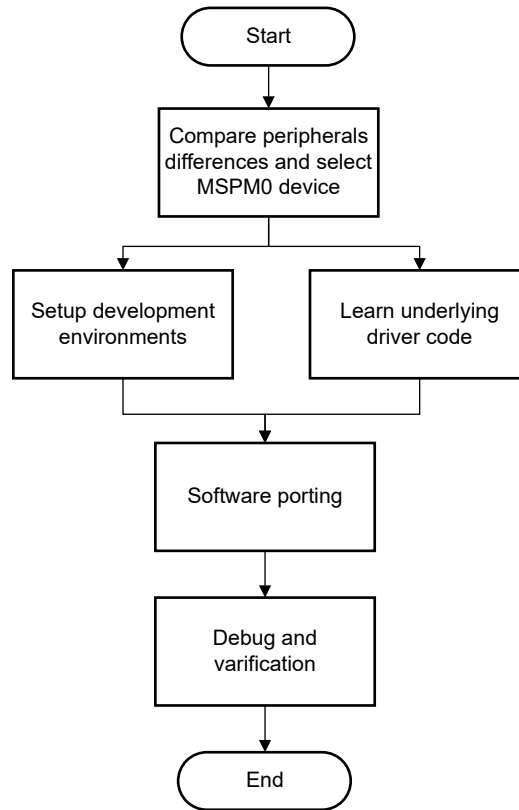


图 1-1. 软件迁移工作流程

2 开发环境

2.1 集成开发环境 (IDE)

Code Composer Studio™ 集成开发环境 (IDE) 简称 CCS，支持 TI 微控制器和嵌入式处理器产品系列中的所有器件。除了 CCS 外，MSP430 和 MSPM0 器件还受其他 IDE 的支持 (请参阅表 2-1)。

表 2-1. IDE 支持

IDE	MSP430	MSPM0
CCS	是	是
IAR	是	是
Keil	否	是

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

2.2 软件生态系统

表 2-2. 软件生态系统比较

特性	MSP430Ware	MSPM0 SDK
寄存器级代码	是	否
驱动程序库	是	是
中间件	是	是
开箱即用代码	是	是
免费 RTOS	否	有

2.2.1 MSP430 软件支持包 : MSP430Ware

MSP430Ware 是一套资源集，可帮助用户高效地创建和构建 MSP430 代码。这些资源支持所有 MSP430 微控制器 (MCU)。除了所有外设的示例代码外，这一完整的设计资源集合还包括各种高度抽象的软件库。MSP430 驱动程序库是一个尤为重要的库，可以帮助软件开发人员利用便捷的 API 来控制错综复杂的低级别硬件外设，从而使生成的代码更易于读取和维护。MSP430Ware 还具有云版本。

2.2.1.1 寄存器级示例代码

16 位 MSP430 MCU 支持高达 25MHz 的内核速度。寄存器级示例代码直接操作寄存器，从而使得工程更快、更小 (示例请参阅图 2-1)。MSP430 MCU 的代码示例涵盖了器件的所有外设功能。

```

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

#include <msp430.h>

int main(void)
{
    volatile unsigned int i;

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

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

```

图 2-1. MSP430 寄存器级示例代码

2.2.1.2 驱动程序库

此外设驱动程序库允许在 API 级别而非器件寄存器级别进行应用开发。这些 API 使开发人员能够专注于应用，而不是所用特定 MSP430 器件的细微差别。如需了解更多信息，请参阅 CCS 中的 *MSP430Ware 外设驱动程序库 API* 和 *MSP430Ware 外设驱动程序库用户指南*，或 [TI Resource Explorer](#)。

2.2.1.3 中间件

对于一些特定的应用场景，中间件示例工程提供了参考代码，可帮助用户更方便地进行设计。例如，IQmath 库有助于开发更快、更复杂的计算，而 USB 开发者软件包有助于开发 USB 应用。有关详细信息，请参阅 [TI Resource Explorer](#) 中每个器件的中间件。图 2-2 展示了典型中间件示例的位置。

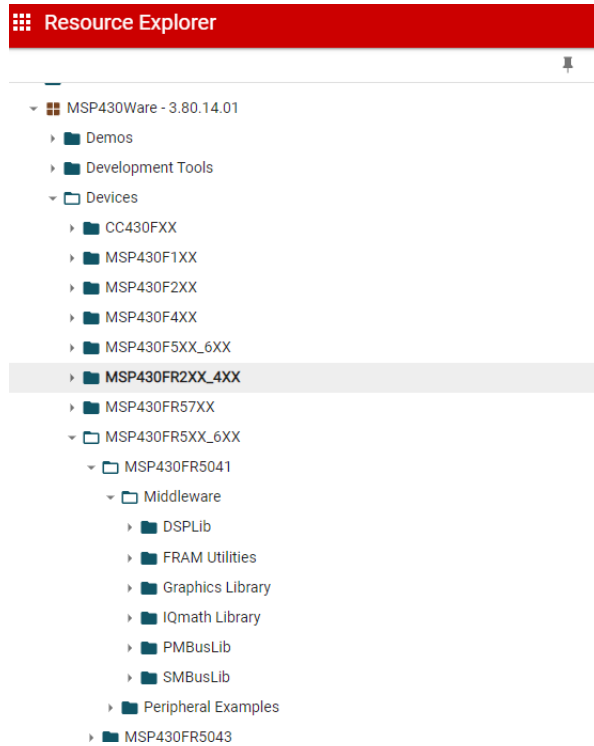


图 2-2. MSP430 中间件

2.2.2 MSPM0 软件支持包 : MSPM0SDK

MSPM0 SDK 提供的组件有助于在德州仪器 (TI) MSPM0+ 微控制器上开发应用。MSPM0 SDK 包含多个软件组件以及关于如何一起使用这些组件的示例。此外，各种示例展示了如何使用各功能区和各个受支持的器件，可用作您自己的项目的起点。图 2-3 展示了典型的 MSPM0 SDK 示例。

示例文件夹分为 RTOS 和非 RTOS 子文件夹 (目前仅支持非 RTOS)。这些文件夹包含每个 LaunchPad™ 开发套件的示例，并根据功能进行整理，例如较低级别的 Driverlib 示例、较高级别的 TI 驱动程序示例以及 GUI Composer、LIN、IQMath 等中间件的示例。有关详细信息，请参阅 *MSPM0 SDK 用户指南*。

一些示例支持 SysConfig，可简化器件配置并加快软件开发。有关更多详细信息，请参阅各个示例。

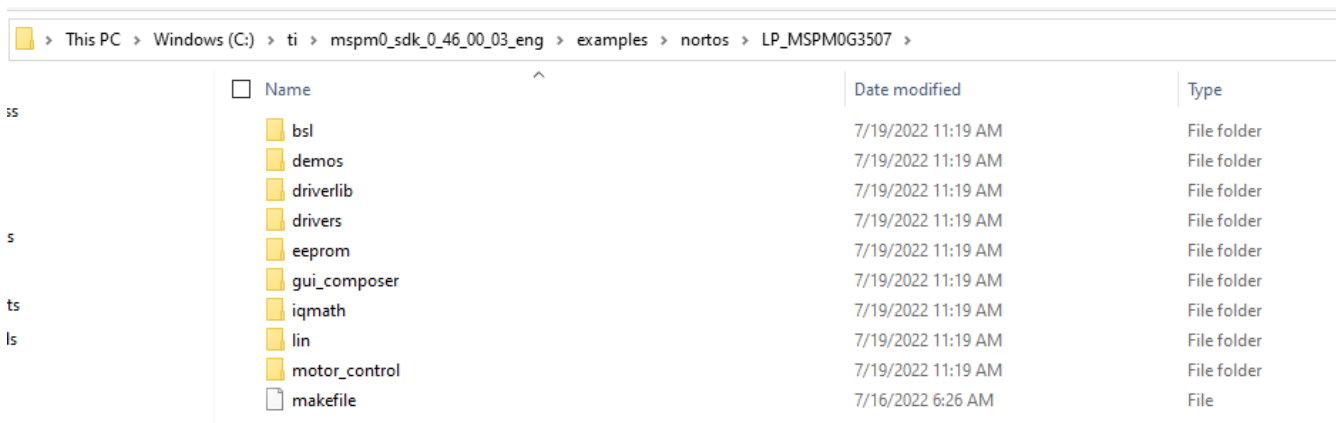


图 2-3. MSPM0 SDK

2.3 用于 MSPM0 MCU 的 SysConfig

SysConfig 是一个直观而全面的图形实用程序集合，用于配置引脚、外设、无线电、子系统和其他组件。

SysConfig 可帮助您直观地管理、发现和解决冲突，以便您有更多时间创建差异化应用程序。该工具输出包括 C

头文件和代码文件，这些文件可与 MSPM0 SDK 示例配合使用，或用于配置定制软件。有关详细信息，请参阅 *MSPM0 SysConfig 指南*。

2.3.1 独立 SysConfig

可以在 PC 上下载并安装独立的 SysConfig (请参阅图 2-4)，以便生成用于配置工程的头文件和代码文件。

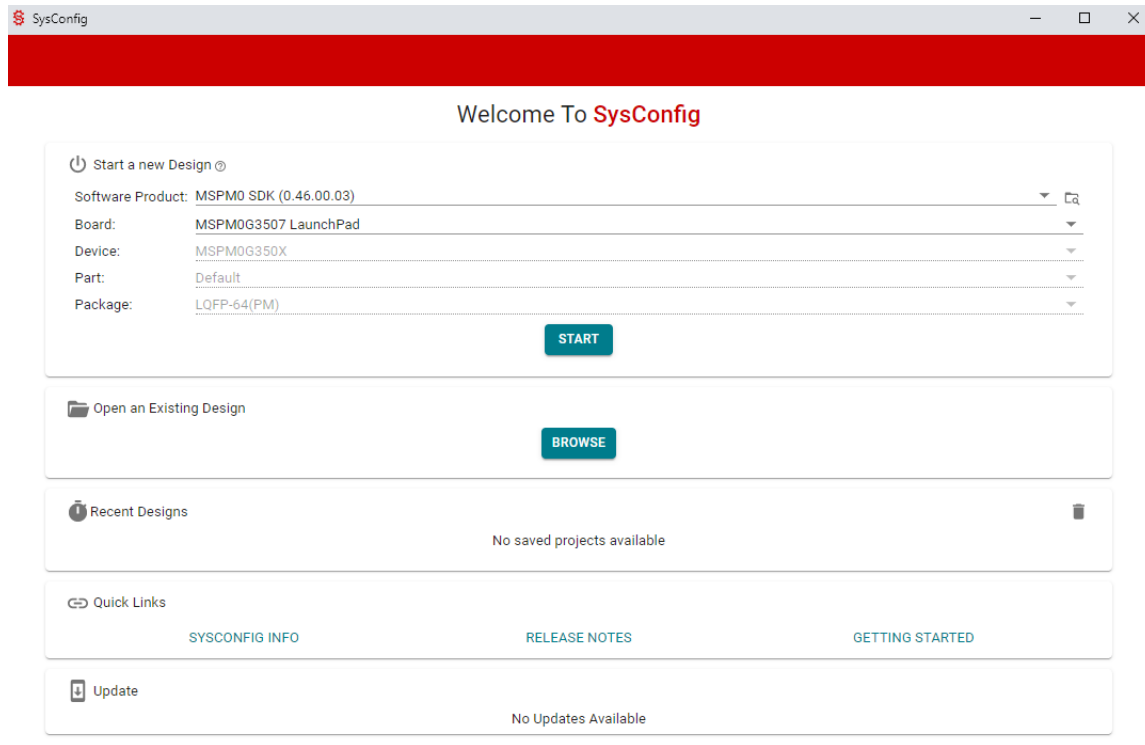


图 2-4. 独立 SysConfig

2.3.2 CCS 集成 SysConfig

MSPM0 SDK 中的许多示例包括扩展名为 *.syscfg* 的 SysConfig 文件。此 SysConfig 工程文件保存系统的初始化设置，例如时钟配置、低功耗模式规则、ADC 配置、I2C、SPI 和 UART 配置以及其他外设配置。这些工程文件可在 CCS 中更改并与工程一起保存。如果要对另一个工程使用相同的配置，则可以轻松地将配置导入到新工程中。

2.3.3 SysConfig 工程示例

在图 2-5 所示的切换输出示例工程中，配置文件包含在工程中。构建工程时，CCS 会根据配置文件生成配置头文件和代码文件。此工程中会配置了 GPIO 和系统时钟。如需进一步开发，请在 CCS 中打开 SysConfig 文件并根据需要修改外设。构建工程时会更新头文件和代码文件。

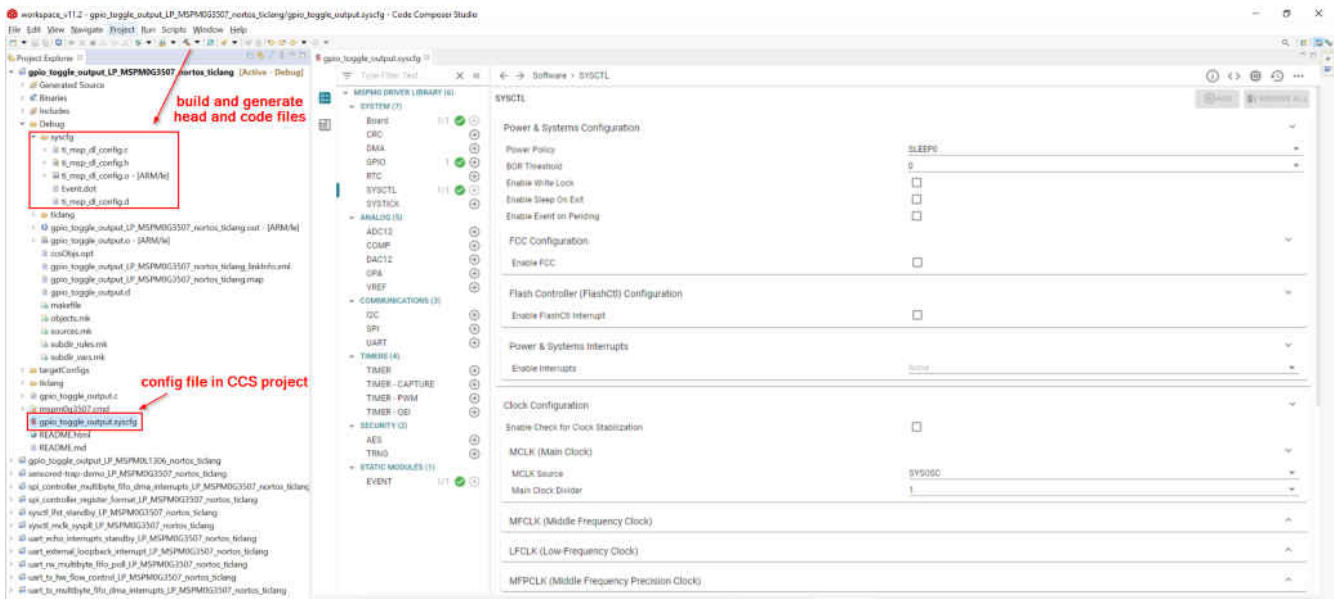


图 2-5. 使用 CCS 的 SysConfig 工程

2.4 MSP430 和 MSPM0 工程

以下是 PWM 输出的示例。在 MSP430 寄存器级代码中，配置寄存器中的每个位都必须设置为时钟控制寄存器、GPIO 寄存器和计时器寄存器的相关值。代码很难读取和修改。

```

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

#include <msp430.h>

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

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

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

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

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

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

```

图 2-6. MSP430 PWM 输出示例

要在 MSPM0 器件中配置相同的功能，请在 SysConfig 工具（用于生成配置文件）中选择所需选项。选择时钟并输入要由该工具计算的 PWM 频率的计数。输入所需的占空比，然后工具会生成比较编号。所有使用的 GPIO 功能均由 SysConfig 配置。使用 SysConfig 有助于设计和初始化您的系统。

SysConfig 输出基于使用 driverlib 的 SDK。与上面的 MSP430 寄存器级示例代码相比，这些 API 可读性更高。

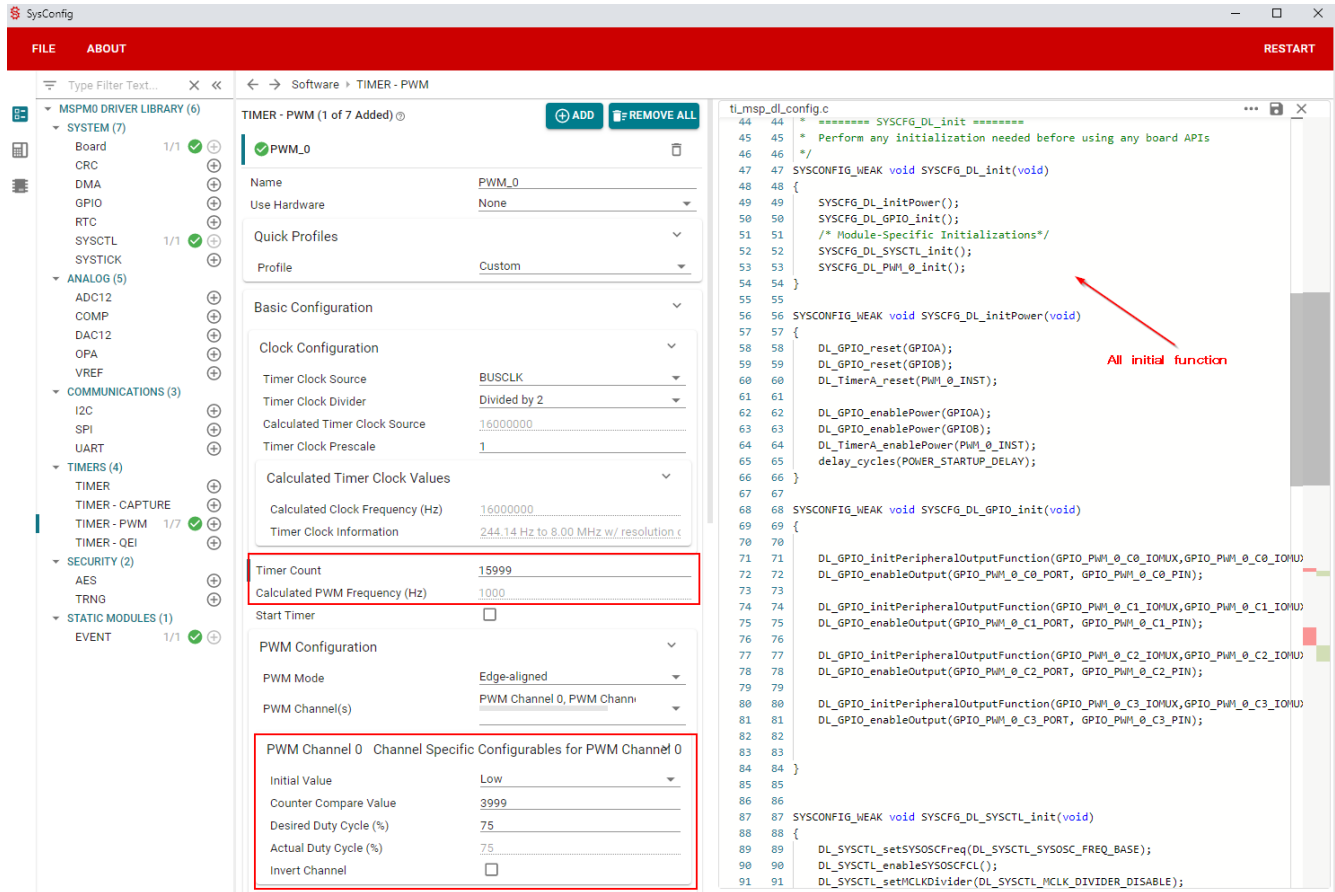


图 2-7. MSPM0 PWM 输出示例

2.5 调试器接口

与 MSP430 器件不同，MSPM0 器件配备了 Arm Cortex-M0+ 内核，因此支持的调试器也不同。表 2-3 列出了这两个系列之间的差异。

表 2-3. 调试器接口

工具类型	MSP430	MSPM0
开发工具	MSP-FET	XDS110/J-Link
生产工具	MSP-GANG	MSP-GANG

2.5.1 MSP430 调试器

MSP-FET（请参阅图 2-8）是一款功能强大的仿真开发工具，可帮助用户在 MSP430 器件上快速开始开发。MSP-FET 支持通过 JTAG 和 SBW 接口进行编程和实时调试。此外，MSP-FET 还可在计算机的 USB 接口和 MSP UART 间提供反向通道 UART 连接。这为 MSP 编程器提供了一种便捷方法，实现了 MSP 和在计算机上运行的终端之间的串行通信。它还支持使用 BSL（引导加载程序）通过 UART 和 I2C 通信协议将程序（通常称为固件）加载到 MSP 目标中。有关详细信息，请参阅 [MSP 调试器用户指南](#)。



图 2-8. MSPFET

2.5.1.1 MSPFET 连接接口

USB 接口将 MSP-FET 连接至计算机，而 14 引脚连接器提供对 MSP 调试仿真端口的访问，该端口使用标准 JTAG 接口或节省引脚的 Spy-Bi-Wire (2 线制 JTAG) 协议。这两种协议具有不同的速度。

表 2-4. MSP430 MCU 的 SBW 和 JTAG 通信速度

接口	慢	中	快
SBW	200 kHz	400kHz	600kHz
JTAG 4 线制 MSP430	1MHz	4 MHz	8MHz

图 2-9 显示了 14 引脚 FET 接口模块连接器与目标器件间的连接，该连接是支持针对 4 线制 JTAG 通信的系统内编程和调试所必需的。图 2-10 给出了 2 线制 JTAG 模式 (Spy-Bi-Wire) 的连接。除了 MSP430G2230 等引脚数少的器件外，大多数 MSP430 器件上都支持 4 线制 JTAG 模式。只有所选器件上支持 2 线制 JTAG 模式。有关哪个器件上可以使用哪种接口的信息，请参阅[适用于 MSP430 MCU 的 Code Composer Studio IDE 用户指南](#)或[适用于 MSP430 MCU 的 IAR Embedded Workbench IDE 用户指南](#)。

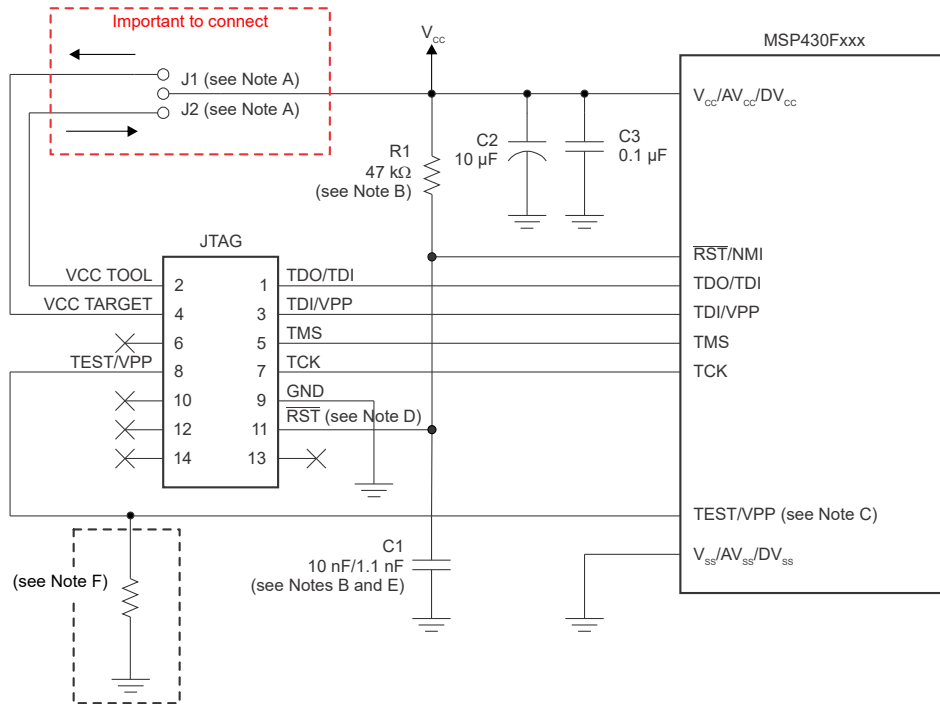


图 2-9. 4 线制 JTAG 通信的信号连接

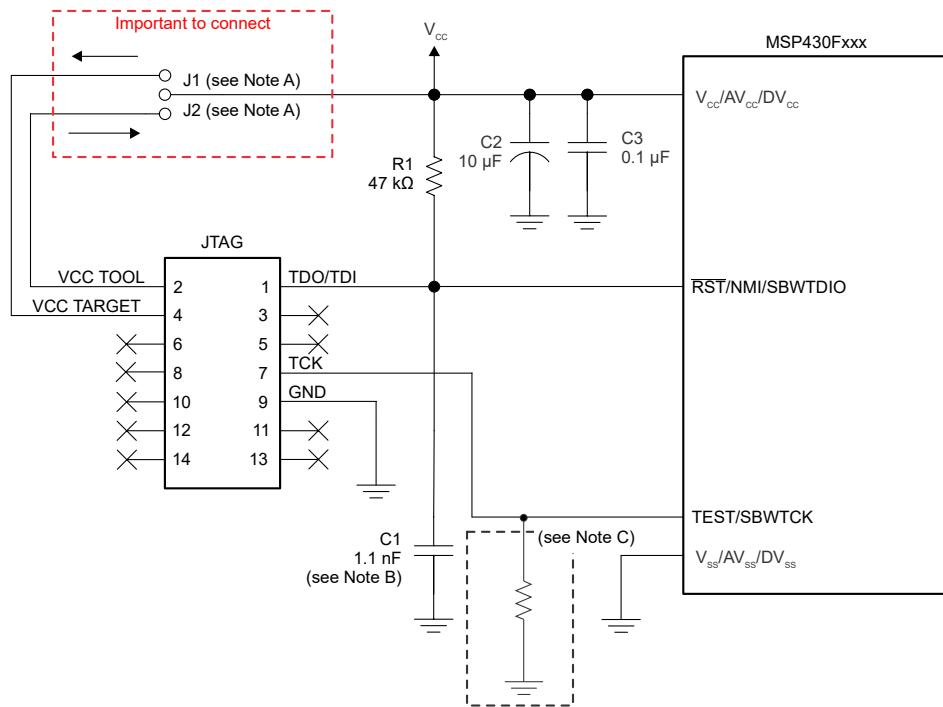


图 2-10. 2 线制 JTAG 通信 (Spy-Bi-Wire) 的信号连接

备注

在一些支持 Spy-Bi-Wire 的 MSP430 器件上，TEST/SBWTCK 对上升的信号沿十分敏感，可能会导致测试逻辑进入一种状态，即无法正确识别输入序列（2 线制或者 4 线制），且 JTAG 访问保持禁用。当 JTAG 连接器被正确连接到目标器件时，SBWTCK 上的单向边沿会出现。

2.5.2 MSPM0 调试器

调试子系统 (DEBUGSS) 将串行线调试(SWD) 两线制物理接口连接到器件内的多个调试功能。MSPM0 器件支持调试处理器执行情况、器件状态和电源状态 (使用 EnergyTrace 技术)。调试器的连接参见图 1-9。

MSPM0 支持用于标准串行线调试的 XDS110 和 J-Link 调试器。

德州仪器 (TI) XDS110 用于 TI 嵌入式处理器。XDS110 通过 TI 20 引脚连接器 (提供适用于 TI 14 引脚、ARM 10 引脚和 ARM 20 引脚连接器的多个适配器) 连接到目标板，通过 USB2.0 高速 (480Mbps) 连接到主机 PC。XDS110 在单个单元中支持更广泛的标准 (IEEE1149.1、IEEE1149.7、SWD)。所有 XDS 调试探针都支持所有具有嵌入式跟踪缓冲器 (ETB) 的 Arm 和 DSP 处理器中的内核和系统跟踪。有关详细信息，请参阅 [XDS110 调试探针](#)。

J-Link 调试探针是优化调试和闪存编程体验的最常见的选择。这些探针得益于创纪录的闪存加载程序、高达 3MiB/s 的 RAM 下载速度以及在 MCU 闪存中设置无限数量断点的功能。J-Link 还支持各种 CPU 和架构，包括 Cortex-M0+。有关详细信息，请访问 [SEGGER J-Link 调试探针页面](#)。

图 2-11 展示了连接到 MSPM0 目标的 XDS110 探针的主要功能区域和接口的简要示意图。

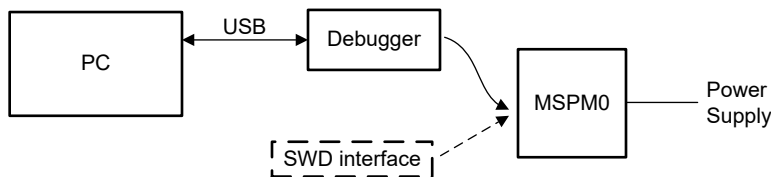


图 2-11. XDS110 探针到 MSPM0 目标

2.5.2.1 MSPM0 调试端口引脚和引脚分配

调试端口使用 SWCLK 和 SWDIO 引脚，它们具有内部下拉和上拉电阻器（请参阅图 2-12）。MSPM0 MCU 系列提供多种具有不同数量可用引脚的封装。相关详细信息，请参阅器件特定数据表。

表 2-5. MSPM0 SWD 接口

器件信号	方向	SWD 功能
SWCLK	输入	来自调试探针的串行线时钟
SWDIO	输入/输出	双向（共享）串行线数据

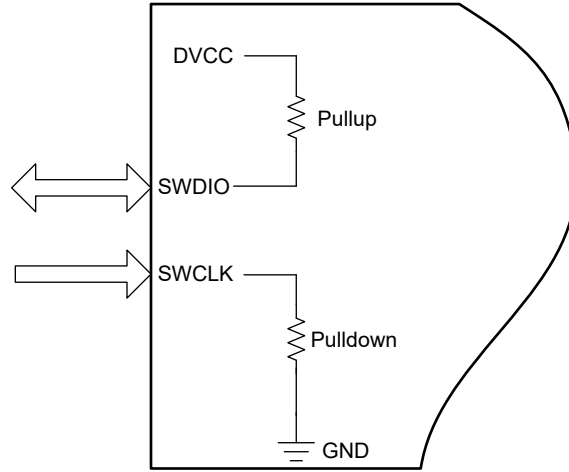


图 2-12. MSPM0 SWD 连接

图 2-13 展示了 MSPM0 SWD 调试端口和标准 10 引脚 JTAG 连接器之间的连接。

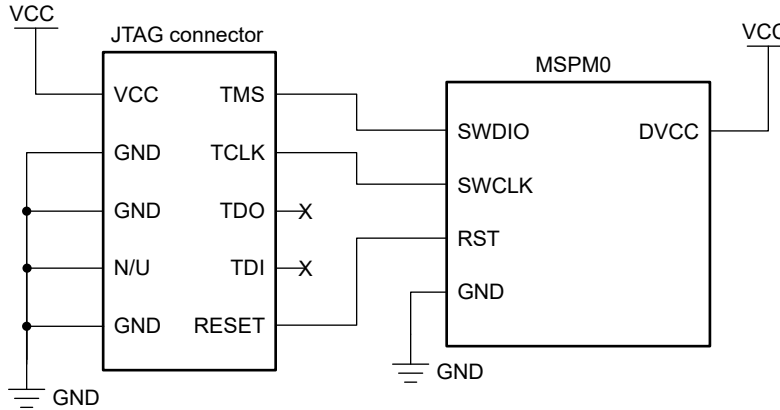


图 2-13. MSPM0 到 10 引脚 JTAG 的连接

3 迁移注意事项

3.1 外设

表 3-1 比较了 MSP 器件系列的外设。要查找每个单独器件上支持的外设，请参阅器件特定数据表。

表 3-1. 外设比较

外设	MSP430	MSPM0L	MSPM0G
CPU	高达 25MHz	高达 32 MHz	高达 80 MHz
SAR ADC	8 位、10 位或 12 位 SAR ADC (取决于器件)	12 位、1Msps SAR ADC	14 位 4Msps SAR ADC 过采样 (4Msps ADC)
Σ - Δ ADC	16 位或 24 位 SD ADC	不适用	不适用
COMP	支持	窗口比较	窗口比较
OPA、GPAMP	GBW 高达 2.8M	烧毁电路 GBW 高达 6M	烧毁电路 GBW 高达 6M
DAC	8 位或 12 位 DAC	8 位 DAC	12 位 DAC
UART	支持 IrDA	支持 LIN、IrDA、ISO7816、RS485	支持 LIN、IrDA、ISO7816、RS485
REF	1.5V、2.0V、2.5V	内部 1.4V 或 2.5V	内部 1.4V 或 2.5V
SPI	高达 8 MHz	高达 16MHz，支持 TI 和 Motorola 模式、命令行模式	高达 32MHz，支持 TI 和 Motorola 模式、命令行模式
I2C	100 kHz	高达 Fm+ 1MHz，支持硬件 SMBus 和 PMBus	高达 Fm+ 1MHz，支持硬件 SMBus 和 PMBus
CAN-FD	不适用	不适用	支持
CRC	CRC16-CCITT	CRC16-CCITT、CRC32-ISO3309	CRC16-CCITT、CRC32-ISO3309
AES	具有 128、192 或 256 位密钥的 AES 128 位数据	具有 128、192 或 256 位密钥的 AES 128 位数据	具有 128、192 或 256 位密钥的 AES 128 位数据
TRNG	不适用	不适用	32 位真随机输出
计时器	通用计时器功能、PWM 或捕获和比较	通用计时器功能、PWM 或捕获和比较	通用计时器功能，具有边沿或中心对齐、互补和死区功能的 PWM
WWDT	1	1	2
RTC	支持	不适用	支持
DEBUG	MSPFET : JTAG (4 线制) 或 SBW	XDS110 或 J-link : SWD	XDS110 或 J-link : SWD
LCD	支持	不适用	不适用
USS	支持	不适用	不适用
CapTIvate 技术	支持	不适用	不适用
USB	支持	不适用	不适用

3.2 系统时钟

3.2.1 振荡器

MSP430 和 MSPM0 器件具有多种类型的时钟源，包括内部和外部时钟源，可实现低系统成本和低功耗。表 3-2 列出了 MSP430 和 MSPM0 器件中的各种不同时钟源。请注意，并非所有器件都具有所有类型的时钟源。相关详细信息，请参阅器件特定数据表。

表 3-2. 振荡器比较

类型	MSP430	MSPM0L	MSPM0G
内部振荡器	DCOCLK : 内部高速振荡器, 频率高达 25MHz	SYSOSC : 4MHz 至 32MHz 的内部振荡器	SYSOSC : 4MHz 至 32MHz 的内部振荡器, 可为 PLL 提供 80MHz 的时钟源
	REFOCLK : 内部 32kHz 振荡器, 可为 FLL 提供时钟源	LFOSC : 内部 32kHz 低频振荡器	LFOSC : 内部 32kHz 低频振荡器
	VLOCKL : 内部 10kHz 超低功耗振荡器	不适用	不适用
外部振荡器	XT1CLK : 高频振荡器或低频振荡器 (32kHz)。取决于器件。	不适用	LFXT : 外部低频振荡器
	XT2CLK : 高频振荡器。取决于器件。	不适用	HFXT : 外部高频振荡器

3.2.1.1 MSPM0 振荡器

MSPM0 器件具有多种类型的时钟源, 包括内部和外部时钟源, 可实现低系统成本和低功耗。表 3-3 列出了 MSPM0 器件中的时钟源。请注意, 并非所有器件都具有所有类型的时钟源。相关详细信息, 请参阅器件特定数据表。

表 3-3. MSPM0 MCU 中的振荡器

类型	时钟源	说明
内部	SYSOSC	系统振荡器 (4MHz 或 32MHz 工厂校准频率、16MHz 或 24MHz 用户校准频率)
	LFOSC	低频振荡器 (典型频率为 32kHz)
	SYSPLL	具有可编程频率的系统 PLL
外部	LFXT	低频、低功耗晶体振荡器 (典型频率为 32kHz)
	HFXT	高频晶体振荡器 (典型频率为 4MHz 至 48MHz)

3.2.2 时钟信号

不同的低功耗模式支持不同的时钟信号, 可降低系统功耗。以下各节介绍了 MSP430 和 MSPM0 MCU 时钟信号的差异。

表 3-4. 时钟信号比较

时钟	MSP430	MSPM0L	MSPM0G
MCLK	源 CPU 和一些数字外设	PD1 总线和外设的主系统时钟	PD1 总线和外设的主系统时钟
CPUCLK		源自 MCLK 的 CPU 时钟	源自 MCLK 的 CPU 时钟
SMCLK	独立于 CPU 工作的外设的子系统主时钟	PD0 外设和 PD0 总线的主系统时钟, 源自 MCLK	PD0 外设和 PD0 总线的主系统时钟, 源自 MCLK
ACLK	辅助时钟 32kHz	32kHz 固定时钟	32kHz 固定时钟
MFCLK		与 MCLK 同步的 4MHz 固定时钟	与 MCLK 同步的 4MHz 固定时钟
MFPCLK		固定 4 MHz	固定 4 MHz

3.3 运行模式

所有 MSP 器件都具有多种低功耗模式。表 3-5 列出了 MSP430 和 MSPM0 MCU 的各种工作模式。

表 3-5. 工作模式比较

MSP430		MSPM0L 和 MSPM0G	
工作模式	说明	工作模式	说明
主动	CPU、时钟和外设工作	运行	CPU、时钟和外设工作
LPM0	处于激活状态且 CPU 时钟关闭	睡眠	处于运行状态且 CPU 时钟关闭

表 3-5. 工作模式比较 (continued)

MSP430		MSPM0L 和 MSPM0G	
工作模式	说明	工作模式	说明
LPM2、LPM3	ACLK 时钟启用，一些外设可选	停止	PD0 启用且 PD1 禁用。可用时钟：MFCLK 或 LFCLK
		待机	PD0 启用且 PD1 禁用。可用时钟：LFCLK
LPM4	所有时钟关闭，一些外设可选	关断	器件关断
LPM3.5	器件通过 RTC 关断		
LPM4.5	器件关断		

3.4 非易失性存储器 (NVM)

3.4.1 MSPM0 存储器保护单元

MSPM0 MCU 支持存储器保护单元，用于根据一组可由编程人员定义的访问权限策略，来检查处理器进行的所有存储器访问。

可通过系统专用外设总线 (PPB) 区域中的存储器映射寄存器配置 MPU。有关 MPU 寄存器配置的更多详细信息，请参阅 [Cortex-M0+ 器件通用用户指南的 MPU 部分](#)。

3.4.2 MSP430 FRAM 和 MSPM0 闪存

FRAM 是一种非易失性存储器，其读取和写入的方式与标准 SRAM 类似。MSP430FRxx 器件上配备 FRAM。

表 3-6. MSP430 FRAM 和 MSPM0 闪存

特性	MSP430 FRAM	MSPM0 闪存
访问	字或字节写入访问	单个闪存字 (64 位) 或多个字
ECC	支持	支持 (MSPM0G 器件)
保护	支持	支持
周期数	10 ¹⁵	100K (下部 32KB) 或 10k (上部 32KB)

FRAM 具有速度限制。CPU 的系统时钟可能超出 FRAM 访问和周期时间要求。针对这些情况，实施了等待状态发生器机制。FRAM 存储器的最大访问速度为 8 MHz。如果 MCLK 的运行速度超过 8MHz 并且需要进行 FRAM 访问，则需要设置等待状态以确保可靠的 FRAM 访问。使用 MCLK ≥ 8 MHz 时，在配置 MCLK 频率之前，应在软件中配置 FRAM 等待状态。

3.4.3 MSP430 闪存和 MSPM0 闪存

表 3-7 比较了 MSP430 闪存器件和 MSPM0 闪存器件的主要特性。详细信息请参阅具体的用户指南。

表 3-7. MSP430 闪存和 MSPM0 闪存

特性	MSP430 闪存	MSPM0 闪存
电压生成	内部	内部
编程模式	字节、字 (2 字节) 和长整型 (4 字节)	单个闪存字 (64 位) 或多个字
擦除	段、存储体、批量	扇区、存储体、批量
ECC	不适用	支持
周期数	100k	100K (下部 32KB) 或 10k (上部 32KB)

3.5 事件和中断处理

在 MSP430 MCU 中，中断优先级是固定的并由图 3-1 中所示连接链中的模块布局来定义。中断分为三种类型：系统复位、不可屏蔽和可屏蔽。

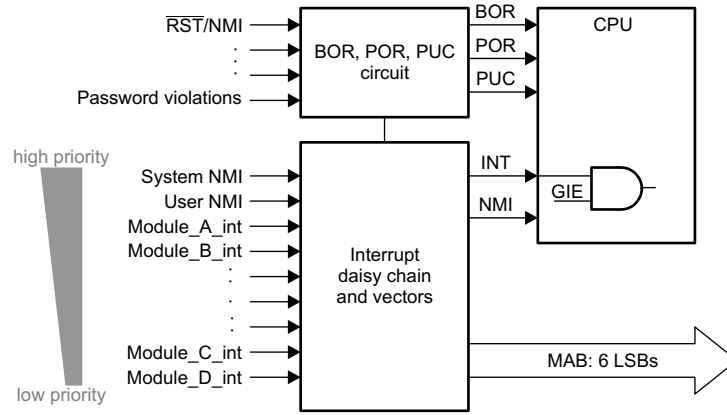


图 3-1. MSP430 事件和中断处理

MSPM0 MCU 具有一个事件管理器，可将数字事件从一个实体传输到另一个实体。事件管理器通过一组定义的事件发布者（发生器）和订阅者（接收器）实现事件传输，这些事件发布者和订阅者通过包含静态路由和可编程路由组合的事件结构进行互连。

事件管理器传输的事件包括：

- 作为中断请求 (IRQ) 传输到 CPU 的外设事件
- 作为 DMA 触发器传输到 DMA 的外设事件
- 传输到另一个外设以直接触发硬件中操作的外设事件

事件管理器通过事件结构将事件发布者连接到事件订阅者。事件结构分为三种类型：静态事件路由、DMA 事件路由和通用事件路由。

图 3-2 展示了事件映射。不同的外设通过不同的事件结构进行路由，可实现不同的事件转换。有关更多详细信息，请参阅器件技术参考手册。

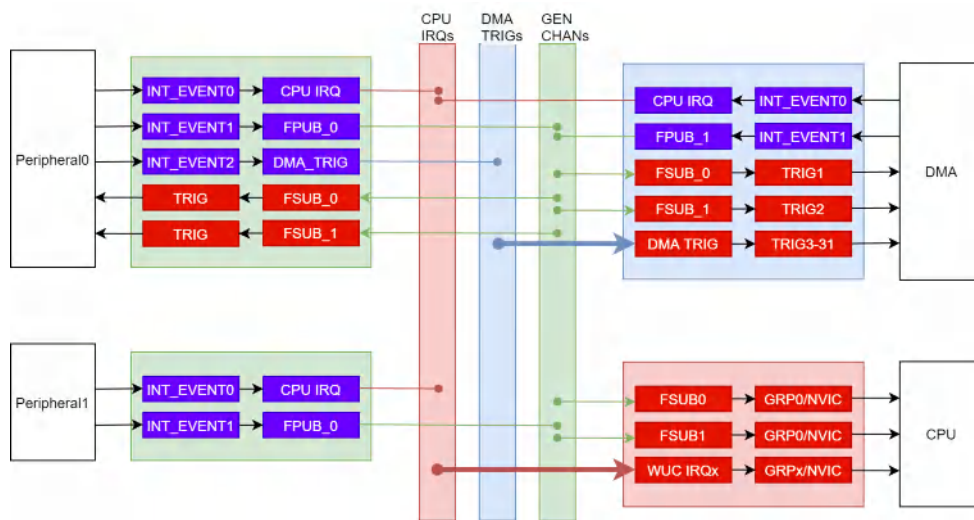


图 3-2. MSPM0 事件和中断处理

3.6 复位级别

MSP430 MCU 支持三个复位级别：POR、BOR 和 POC。每个都由特定事件触发，如器件系列用户指南中所述。这些触发信号之间的关系如下：

- 当 BOR 产生时，总是会生成一个 POR，但一个 POR 不会生成一个 BOR。
- 当 POR 产生时，总是会产生 PUC，但是 PUC 不会生成一个 POR。

MSPM0 MCU 有五个复位级别：

1. 上电复位 (POR)
2. 欠压复位 (BOR)
3. 引导复位 (BOOTRST)
4. 系统复位 (SYSRST)
5. CPU 复位 (CPURST)

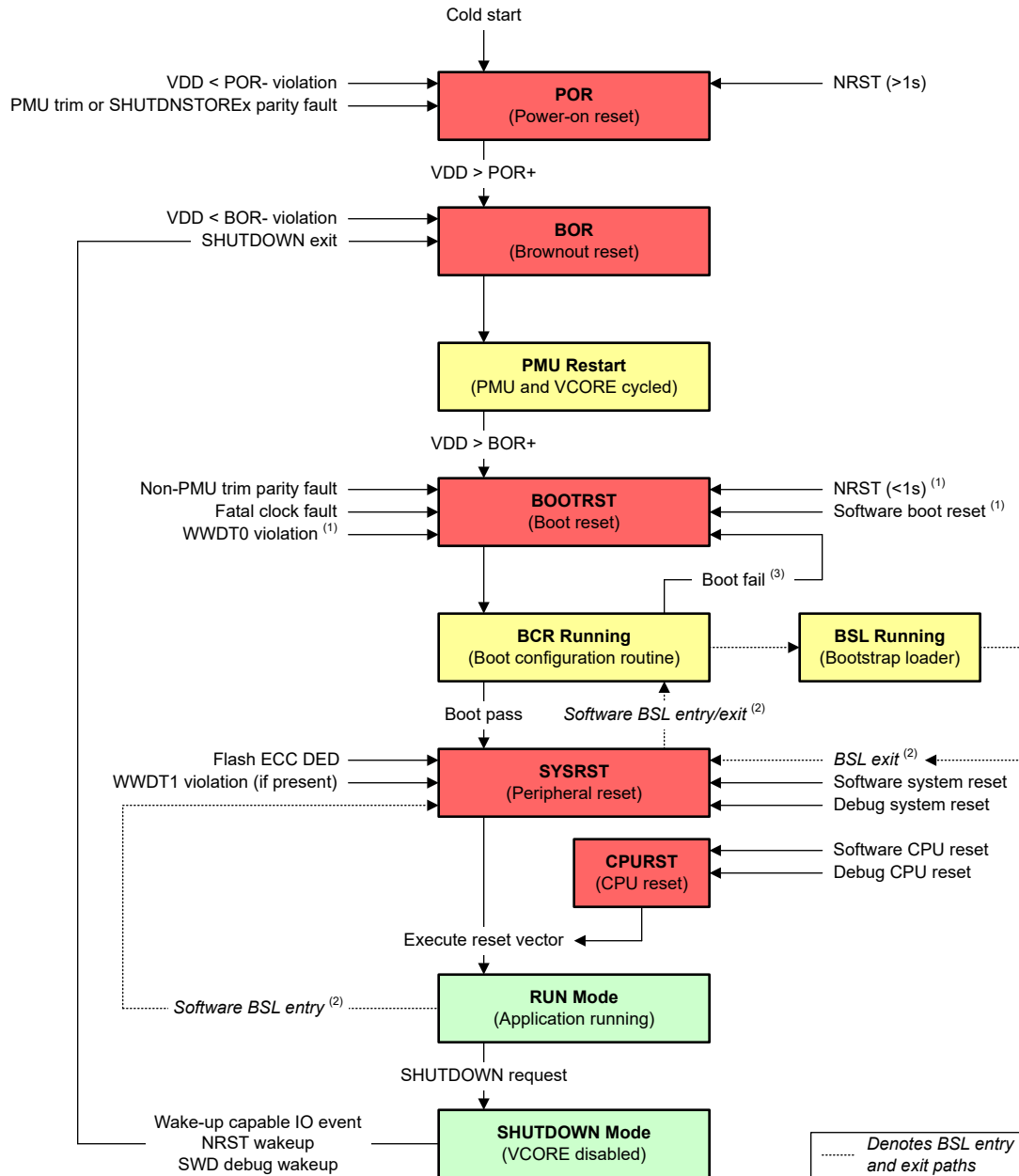


图 3-3. MSPM0 复位

3.7 GPIO 和引脚多路复用

在 MSP430 MCU 中，这些引脚默认设置为数字 I/O，并具有 ADC 或 DAC 输出、CapTIvate I/O 或 eUSCI 等多路复用功能。数字 I/O 特性包括：

- 可独立编程的 I/O
- 输入或输出的任意组合
- 可独立配置的 P1 和 P2 中断。某些器件也许包含额外的端口中断。

- 独立输入和输出数据寄存器
- 单独可配置的上拉或者下拉电阻器

在 MSPM0 器件中，所有引脚都默认设置为模拟功能。

IOMUX 可管理数字 IO 的配置。IOMUX 配置的主要功能包括：

- 选择将哪个外设复用到每个数字 IO 引脚 (例如，GPIO 或 UART 外设)
- 数字输入路径配置
 - 迟滞控制
 - 输入路径启用或禁用
 - 输入逻辑反转控制
- 数字输出路径配置
 - 驱动强度控制
 - 输出连接启用或禁用
 - 输出逻辑反转 (与输入逻辑反转共享控制)
 - 逻辑高电平至高阻态输出转换 (用于开漏型接口)
 - 禁用连接到 IO 的外设时保持“最后状态”
- 唤醒配置 (用于从 SHUTDOWN 模式唤醒)
 - 唤醒比较电平
 - 唤醒启用或禁用
- 上拉或下拉电阻控制

3.8 通信接口

MSPM0 MCU 通信接口的性能优于 MSP430 MCU。MSPM0 MCU 上可以实现更丰富的应用。

3.8.1 SPI

SPI 模块的一个标准化串行接口可使用 SPI 协议在 MSPM0 器件与其他外部器件 (例如传感器、存储器、ADC 或 DAC) 之间传输数据。

SPI 模块具有以下特性：

- 可配置为控制器或外设
- 可编程的时钟位速率以及预分频器；
- 独立的发送 (TX) 和接收 (RX) 先入先出 (FIFO) 缓冲区
- 可编程数据帧大小从 4 位到 16 位 (控制器模式)
- 可编程数据帧大小从 7 位到 16 位 (外设模式)
- 发送和接收 FIFO 中断、超限和超时中断以及 DMA 完成
- 可编程 SPI 模式支持 Motorola SPI、MICROWIRE 或德州仪器 (TI) 格式
- 直接存储器存取控制器接口 (DMA) :
 - 相互独立的发送通道和接收通道
 - 发送完成中断

3.8.2 I2C

I2C 模块的一个标准化串行接口可在 MSP 器件与其他外部 I2C 器件 (例如传感器、存储器或 DAC) 之间传输数据。I2C 外设通过由数据 (SDA) 和时钟 (SCL) 线路组成的两线制串行总线提供双向数据传输。I2C 总线可与诸如串行存储器 (RAM 和 ROM)、网络设备、LCD、音频发生器等外部 I2C 器件相连。在产品开发和制造过程中，I2C 总线还可用于系统测试和诊断用途。该 I2C 外设可以向总线上的其他 I2C 器件发送数据，也可以从这类器件接收数据。

控制器包含 I2C 模块，其特性如下：

- I2C 总线上的器件可指定为控制器或目标器件。
 - 支持作为具有 7 位寻址功能的控制器或目标器件发送和接收数据
- 支持四种 I2C 模式
 - 控制器发送
 - 控制器接收

- 目标发送
- 目标接收
- 支持的传输速度：标准 (100kbps)、快速模式 (400kbps)、超快速模式 (1Mbps)
- 用于接收和发送数据的 FIFO，8 字节
- 故障抑制
- 独立控制器和目标中断生成
- 具有仲裁、时钟同步和多控制器支持的控制器运行
- SMBus 的硬件支持
 - 时钟低电平超时中断
 - 双目标地址能力
 - 快速命令功能
- 为 DMA 提供硬件支持，具有独立的发送通道和接收通道

3.8.3 UART

此接口可用于在 MSPM0 器件和其他器件之间传输数据，支持异步串行通信协议，例如 LIN (本地互连网络)、ISO7816 (智能卡协议)、IrDA (红外数据协会)、硬件流控制 (CTS/RTS) 和多处理器通信。

MSPM0 中 UART 控制器的不同特性：

- 独立的发送和接收 4 深度 FIFO 可减少 CPU 中断服务负载
- 支持 DMA 数据传输
- 支持环回模式运行
- 支持硬件流控制
- 支持 9 位多点配置
- 支持的协议：
 - 本地互连网络 (LIN) 支持
 - DALI
 - IrDA
 - ISO/IEC 7816 智能卡
 - RS485
 - 曼彻斯特编码
 - 空闲线多处理器

3.8.4 CAN FD

与 MSP430 不同，CAN 控制集成在 MSPM0 器件上。控制器局域网 (CAN) 是一种串行通信协议，用于有效地为具有高可靠性的分布式实时控制提供支持。CAN 具有较高的抗电气干扰能力，并且能够检测各种类型的错误。在 CAN 中，许多较短的信息会广播到整个网络，从而在系统的每个节点中提供数据一致性。

MCAN 模块支持经典 CAN 和 CAN FD (具有灵活数据速率的 CAN) 协议。CAN FD 特性可实现更高的吞吐量和增加每个数据帧的有效负载。经典 CAN 和 CAN FD 器件可以在同一网络上共存而不会发生任何冲突，前提是经典 CAN 器件使用部分网络收发器，其中该收发器可以检测和忽略 CAN FD，而不产生总线错误。MCAN 模块符合 ISO 11898-1:2015 标准。

CAN 控制器及其功能的详细信息。请参阅 MSPM0 技术参考手册。

3.9 引导加载程序 (BSL)

表 3-8 比较了 MSP430 和 MSPM0 BSL 实施方法。

表 3-8. BSL 比较

		MSP430									MSP432	MSPM0	
		F20xx、 G2xx0、 G2xx1、 G2xx2、 I20xx	F1xx、 F2xx、 F4xx、 G2xx3	F5xx、F6xx		FR5xx、FR6xx		FR231x 、 FR242x 、 FR243x 、 FR25xx 、 FR263x	FR215x 、 FR235x 、 FR247x 、 FR267x	FR20xx 、 FR21xx 、 FR41xx	P4xx	M0Gxxx M0Lxxx	
非 USB	USB			工厂	加密引导 加载程序								
常规	BSL 存储器类型	无 BSL	ROM	闪存	闪存	ROM	FRAM	ROM	ROM	ROM	闪存	ROM	
	BSL 存储器大小	不适用	1KB	2KB	2KB	2KB	4KB	3KB	3KB	1KB	8KB	5K	
	用户配置								✓		✓	✓	
	UART		✓	✓		✓	✓	✓	✓	✓	✓	✓	
	I2C			✓		✓	✓	✓	✓		✓	✓	
	SPI										✓		
	USB				✓								
	支持基于闪存的接口插件											✓	
调用机制	I/O 上的进入序列	TEST/RST 上的序列	✓	✓		✓	✓	✓	✓	✓			
		PUR 引脚连接至 VUSB			✓								
		已定义 I/O 上的序列					✓				✓	✓	
	空的重置矢量调用 BSL				✓	✓	✓	✓	✓		✓	✓	
	从软件应用程序调用 BSL		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
工具支持	硬件	MSP-BSL “Rocket”	✓	✓		✓	✓	✓	✓	✓	✓		
		MSP-FET		✓		✓	✓	✓	✓	✓	✓	(1)	
		USB 电缆				✓							
		USB 转串行转换器		✓								✓	
		XDS110										✓	
	软件	BSL 脚本编辑器			✓	✓	✓	✓	✓	✓	✓	✓	
		BSLDEMO		✓									
MSPM0 BSL GUI												✓	
安全性	密码保护 (字节)		32	32	32	32		32	32	32	256	32	
	批量擦除错误密码		✓	✓	✓	✓		✓	✓	✓	✓	✓(2)	
	BSL 有效载荷加密						✓				✓		
	通过引导代码更新 IP 受保护的区域										✓		
	认证加密						✓						
	附加安全性						✓						

- (1) 可以使用 USB 转 UART 或 I2C 通道来通过 BSL 与 MSPM0 器件通信。
(2) 三个密码故障会触发包括恢复出厂设置在内的安全操作。

3.10 模拟外设

3.10.1 SAR ADC

表 3-9 比较了 MSP430 和 MSPM0 SAR ADC 的实现。

表 3-9. ADC 比较

特性	MSP430	MSPM0L	MSPM0G
分辨率	10 位或 12 位	10 位或 12 位	10 位或 12 位
采样率	200 ksps	1Msps	12 位 4Msps
FIFO 模式	不支持	FIFO	FIFO
内部 REF	1.5V、2.0V、2.5V	1.4 V/2.5 V	1.4 V/2.5 V
过采样	不支持	支持	支持
窗口比较器	部分器件上支持	支持	支持
同步采样	不支持	不支持	支持

3.10.1.1 同时采样

电流和电压检测等一些应用需要同时在多个模拟信号中进行测量。在这些情况下，需要在单个 MCU 上使用多个 ADC 来执行同步采样。MSPM0xx 平台中具有多个 ADC 外设的任何器件都支持同步采样。

3.10.1.2 窗口比较器

ADC 中有一个可用的窗口比较器单元，可用于检查输入信号是否处于软件设置的预定义阈值范围内。MSPM0L 和 MSPM0G 器件都支持此功能。进入 MEMRES 或 FIFO 的 ADC 结果将根据窗口比较器的阈值进行检查。

根据比较结果可以产生 3 个中断条件：

1. LOWIFG - 转换结果低于下限阈值 (WCLOW)
2. HIGHIFG - 转换结果高于上限阈值 (WCHIGH)
3. INIFG - 转换结果介于或等于下限阈值和上限阈值

3.10.2 COMP

MSPM0 器件具有一个性能优于 MSP430 器件的高速比较器。

表 3-10. COMP 比较

	MSP430	MSPM0L	MSPM0G
多个输入源	是	是	是
用于 COMP 的软件 LPF	是	是	是
基准电压	VREF 和 VDD	VDD	VREF 和 VDD
内部 DAC	6 位	8 位	8 位
传播延迟 (高速)	120ns 至 1 μ s，具体取决于器件	40ns	40ns
窗口比较模式	否	否	是
电压迟滞	可编程	可编程	可编程

3.10.2.1 窗口比较模式

一些 MSPM0G MCU 具有多个比较器。窗口比较器模式的目的是在由上下限阈值定义的指定阈值范围内监控输入信号。

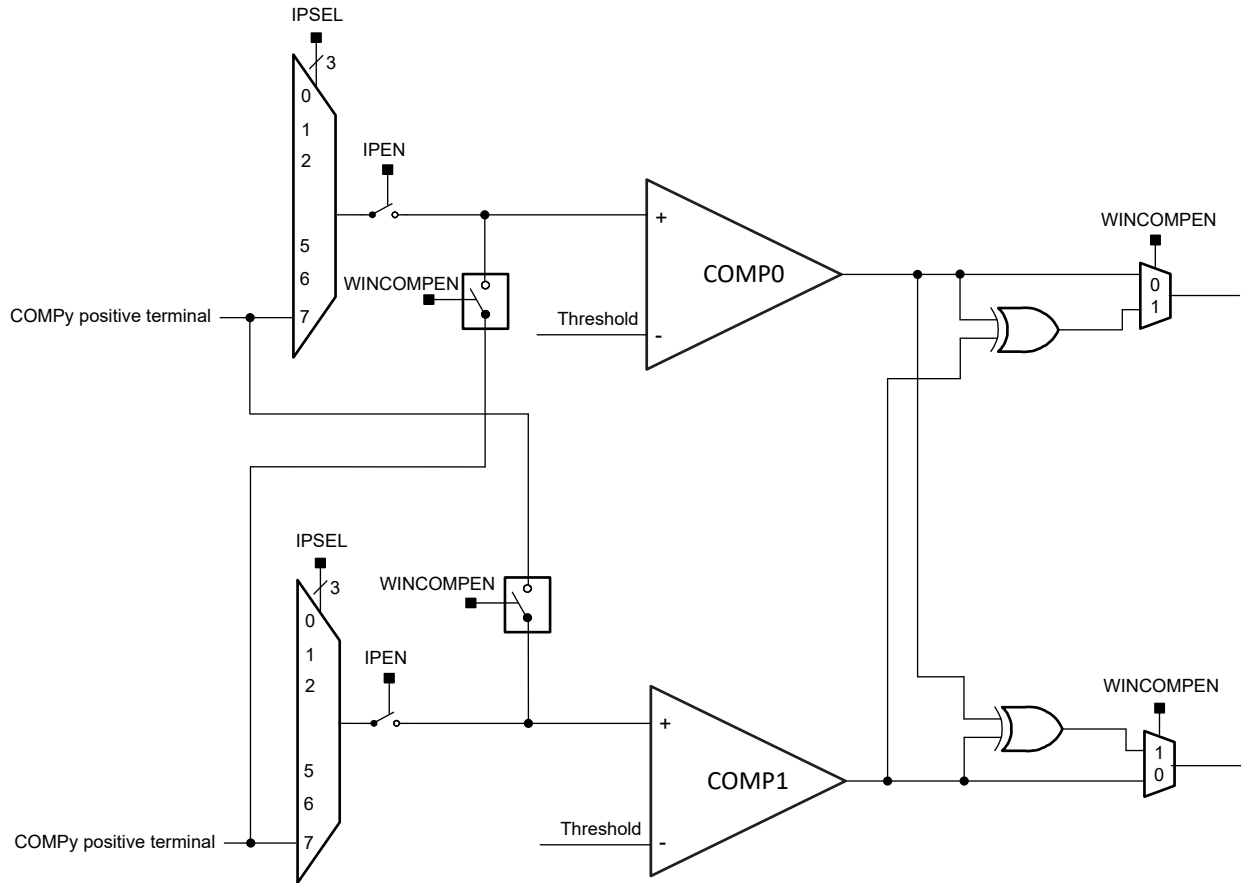


图 3-4. 窗口比较模式方框图

3.10.3 OPA

MSP430 和 MSPM0 MCU 具有丰富的模拟外设。这些外设具有出色的混合信号处理能力，这得益于内部模拟和数字 IP，尤其是 OPA。

表 3-11 比较了 MSP430 和 MSPM0 运算放大器的差异。

表 3-11. 运算放大器比较

	MSP430	MSPM0L、MSPM0G	
	SAC-OA	OPA	GPAMP
GBW	2.8 MHz	6 MHz	350 kHz
PGA	33x	32x	不适用
交互	是	是	是
烧毁检测	否	是	否

图 3-5 展示了典型用例中 MSPM0 OPA 的示例电路。

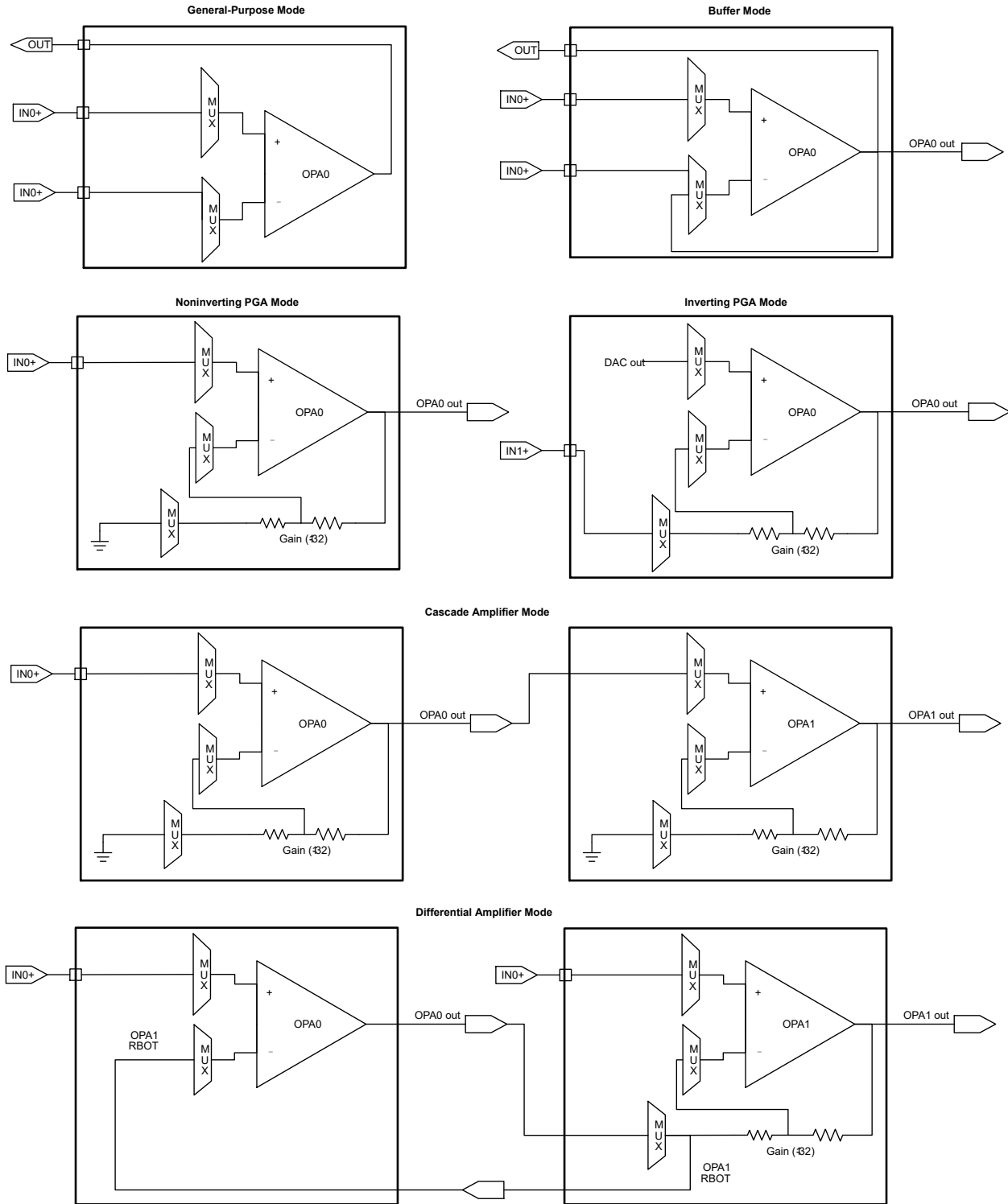


图 3-5. 运算放大器的典型用例

3.11 计时器

表 3-12 将 MSPM0 计时器 (TIMA、TIMH、TIMG) 与 MSP430 计时器 (Timer_A、Timer_B) 进行了比较。

表 3-12. 计时器比较

	MSP430		MSPM0L 和 MSPM0G		
	Timer_A	Timer_B	TIMA	TIMG	TIMH
计数器	16 位	16 位	16 位	16 位	24 位
计数模式	向上计数或向上/向下计数	向上计数或向上/向下计数	向下计数或向上/向下计数	向下计数或向上/向下计数	向下计数或向上/向下计数
比较模式	是	是	是	是	是
PWM 输出	是	是	是	是	是
同步	否	否	是	是	是
交叉触发器	否	否	是	是	是
具有编程死区的互补 PWM	否	否	是	否	否
QEI	否	否	否	是	否

MSPM0 计时器支持互补 PWM、死区和交叉触发器等高级功能，可用于电机控制和其他复杂 PWM 控制。有关更多信息，请参阅 TRM 系列和应用手册。

3.12 硬件设计指南

有关硬件设计注意事项的更多信息，请参阅以下文档：

[MSPM0 L 系列 MCU 硬件开发指南](#)

[MSPM0 G 系列 MCU 硬件开发指南](#)

这些指南介绍了使用 MSPM0 MCU 进行硬件开发的注意事项，包括电源、复位电路、时钟、调试器连接、关键模拟外设、通信接口、GPIO 和电路板布局布线指南的详细硬件设计信息。

4 修订历史记录

注：以前版本的页码可能与当前版本的页码不同

Changes from Revision * (December 2022) to Revision A (March 2023)	Page
• 通篇更新了链接和图像.....	1

重要声明和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2023，德州仪器 (TI) 公司