Application Note

MSPM0 MCUs Quick Reference Guide

Table of Contents

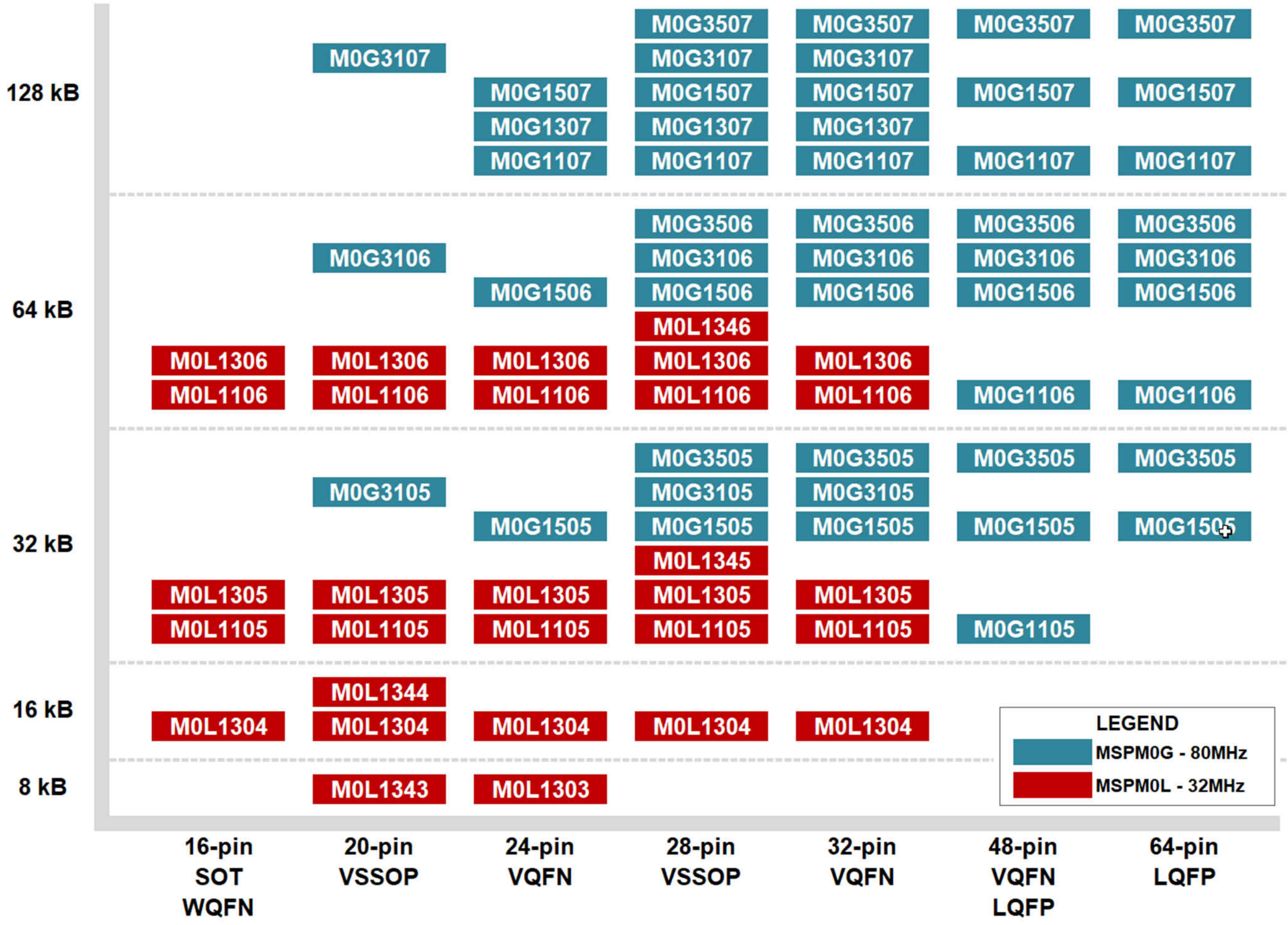
Trademarks

TI E2E™, LaunchPad™, Code Composer Studio™, and EnergyTrace™ are trademarks of Texas Instruments.
Arm® and Cortex® are registered trademarks of Arm Limited.
All trademarks are the property of their respective owners.

1 MSPM0 Overview

1.1 Portfolio

TI's scalable portfolio of MSPM0 MCUs is based on the enhanced Arm® Cortex®-M0+ 32-bit processor operating at speeds up to 80 MHz. The Cortex-M0+ processor is the most energy-efficient Arm processor available for embedded applications. These cost-optimized MCUs offer pin-to-pin compatibility across a wide range of memory and package sizes as shown in Figure 1-1. Software development can be re-used across the entire MSPM0 portfolio. An overview of the MSPM0 MCUs and key features is shown in Table 1-1. Refer to the device-specific data sheet to learn more about additional device-specific features.



	16-pin SOT WQFN	20-pin VSSOP	24-pin VQFN	28-pin VSSOP	32-pin VQFN	48-pin VQFN LQFP	64-pin LQFP
128 kB		M0G3107	M0G1507 M0G1307 M0G1107	M0G3507 M0G3107 M0G1507 M0G1307 M0G1107	M0G3507 M0G3107 M0G1507 M0G1307 M0G1107	M0G3507 M0G1507 M0G1107	M0G3507 M0G1507 M0G1107
64 kB	M0L1306 M0L1106	M0G3106 M0L1306 M0L1106	M0G1506 M0L1306 M0L1106	M0G3506 M0G3106 M0G1506 M0L1346 M0L1306 M0L1106	M0G3506 M0G3106 M0G1506 M0L1306 M0L1106	M0G3506 M0G3106 M0G1506 M0G1106	M0G3506 M0G3106 M0G1506 M0G1106
32 kB	M0L1305 M0L1105	M0G3105 M0L1305 M0L1105	M0G1505 M0L1305 M0L1105	M0G3505 M0G3105 M0G1505 M0L1345 M0L1305 M0L1105	M0G3505 M0G3105 M0G1505 M0L1305 M0L1105	M0G3505 M0G1505 M0G1105	M0G3505 M0G1505
16 kB	M0L1304	M0L1344 M0L1304	M0L1304	M0L1304	M0L1304		
8 kB		M0L1343	M0L1303				

LEGEND
🟦 MSPM0G - 80MHz
🟥 MSPM0L - 32MHz

Figure 1-1. MSPM0 Portfolio

Table 1-1. Key Features

Feature	MSPM0Gx Series	MSPM0Lx Series
Core / Frequency	CM0+ / 80 MHz	CM0+ / 32 MHz
Supply Voltage	1.62 V to 3.6 V	1.62 V to 3.6 V
Temperature	-40°C to 125°C	-40°C to 125°C
Memory	128KB to 32KB	64KB to 8KB
RAM	Up to 32KB	Up to 4KB
GPIO (max)	60	28
Analog	2x 4-Msps 12-bit ADC	1x 1-Msps 12-bit ADC
	3x high-speed comparators	1x high-speed comparator
	2x op amps	2x op amps
	1x 12-bit DAC	

Table 1-1. Key Features (continued)

Feature	MSPM0Gx Series	MSPM0Lx Series
Communication (max)	2x SPI	1x SPI
	2x I2C Fast+	2x I2C Fast+
	4x UART (LIN)	2x UART (LIN)
	1x CAN-FD	
Timers	7	4
Advance Timers[HJ2]	Yes (2x)	No
Hardware Accelerator	Optional	N/A
Security	CRC, TRNG, AES256	CRC
Low power	Active: 85 µA/MHz	Active: 85 µA/MHz
	Standby (RTC): 1.5 µA	Standby: 1.5 µA

1.2 Ecosystem

MSPM0 MCUs are supported by an extensive hardware and software ecosystem. The ecosystem includes easy to use development tools, affordable evaluation boards, and a wide range of embedded software kits, drivers, and examples. Additionally, interactive trainings included in MSP Academy provide a guided learning path and online support is offered directly through the TI E2E™ support forums.

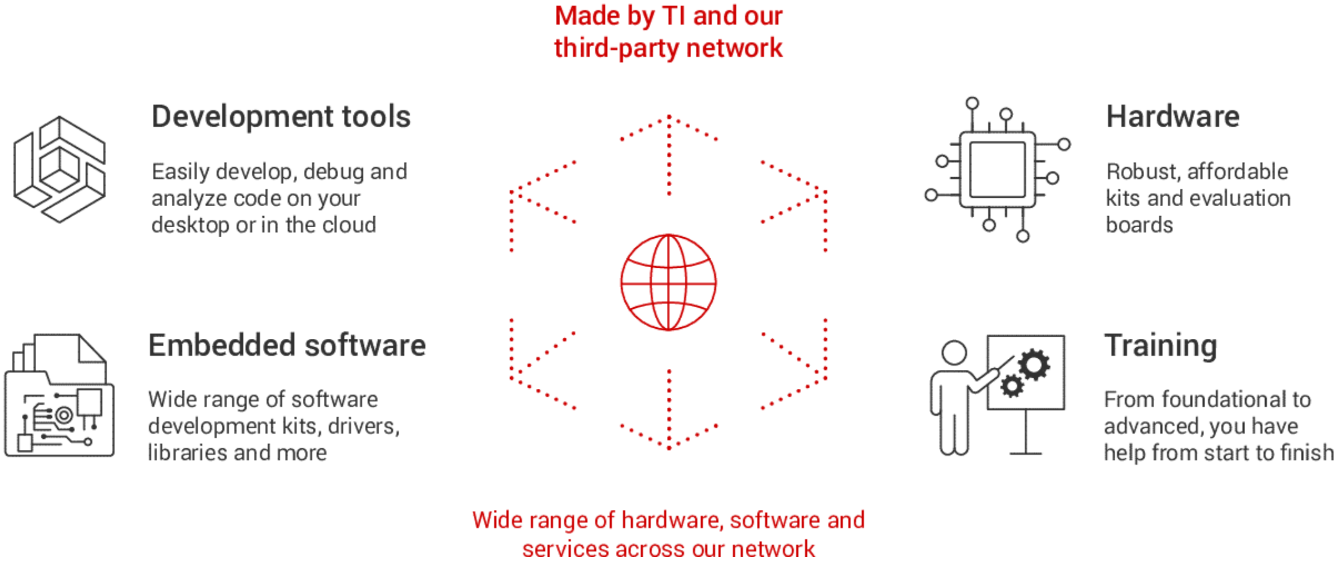

Made by TI and our third-party network
Development tools
Easily develop, debug and analyze code on your desktop or in the cloud
Hardware
Robust, affordable kits and evaluation boards
Embedded software
Wide range of software development kits, drivers, libraries and more
Training
From foundational to advanced, you have help from start to finish
Wide range of hardware, software and services across our network

Figure 1-2. MSPM0 Ecosystem

To start developing with MSPM0 MCUs, purchase a MSPM0 LaunchPad™ development kit. Download Code Composer Studio and the MSPM0 SDK to access example code and demos. Consider completing the trainings and labs in MSPM0 Academy. These labs demonstrate key features of the MSPM0 peripherals and include several interactive exercises.

2 MSPM0 Design Resources

2.1 Training Resources

2.1.1 MSP Academy

MSPM0 Academy delivers easy-to-use training modules that span a wide range of topics and launchpads in the MSPM0 SDK. These interactive trainings are a great starting point for all developers to learn about the MSPM0. Trainings walk through the different peripherals as well as system level topics. The Academies are available in both English and Mandarin.

- English MSPM0 Academy
- Chinese MSPM0 Academy

2.1.2 Precision Labs

Precision Labs - Microcontrollers (TIPL) is the most comprehensive online classroom for analog signal chain and embedded processing designers. From foundational knowledge to advanced concepts, our logical, sequenced and comprehensive teaching approach is both intuitive and practical. The training series, which includes videos and downloadable reference materials, will deepen the technical expertise of experienced engineers and accelerate the development of those early in their career. Join our industry experts as they guide you through our extensive library of tutorials, as well as hands-on experiments for selected topics, and become an expert yourself!

2.2 Development Tools

2.2.1 Integrated Development Environments (IDEs)

There are three main IDEs that support MSPM0.

IDE	Description
Code Composer Studio: Theia	TI's new Theia-based IDE compatible with Visual Studio Code extensions
Code Composer Studio	TI's eclipse-based IDE that provides a suite of tools to develop and debug embedded application.
IAR Embedded Workbench for Arm	Third-party IDE from IAR Systems.
Keil µVision	Third-party IDE from Arm Keil

Features	CCS	IAR	Keil
Edit code	✓	✓	✓
Compile and debug code	✓	✓	✓
SWD programming	✓	✓	✓
Windows OS	✓	✓	✓
Mac OS	✓		
Linux OS	✓		

Code Composer Studio (CCS) is TI's preferred IDE. CCS is free and integrates TI Resource Explorer, which is the easiest way to access code examples and trainings provided by the MSPM0 SDK.

Here are additional resources for working with CCS:

- CCS interactive academy training
- CCS v12.1.0 Getting Started Guide
- Video: Getting Started with Code Composer Studio v9.3

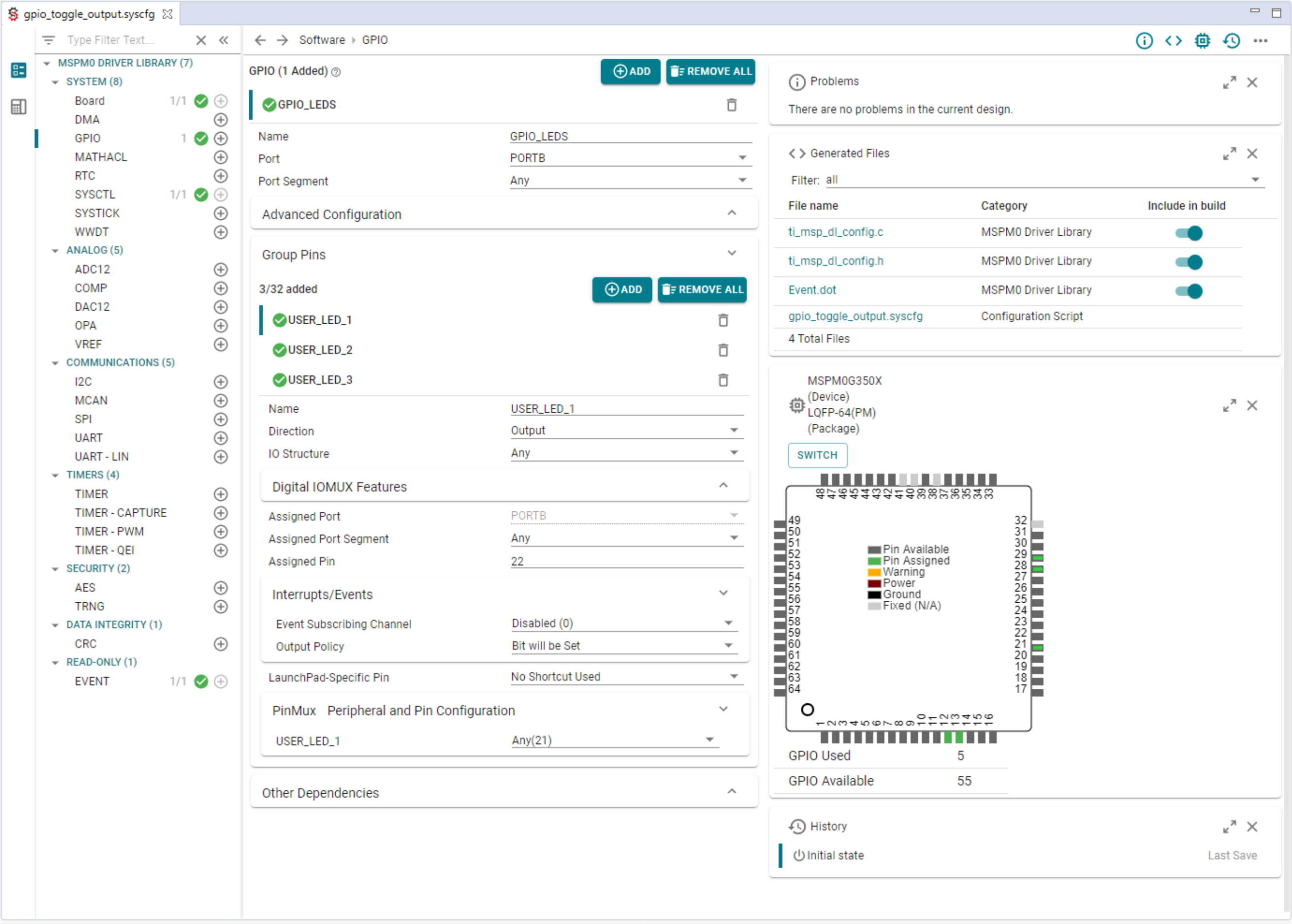
- CCS Integrated tools:
  - SysConfig: System configuration and software generation tool
  - TI Resource Explorer: Contains MSPM0 software resources for evaluation and development.
  - EnergyTrace: Performs an energy-based code analysis that measures and displays the energy profile for MSPM0 MCUs in any application.

– ULP (Ultra-Low Power) Advisor: Provides suggestions for ultra-low power improvements in your code.

2.2.2 SysConfig Code Generation Tools

System Configuration (SysConfig) GUI is a powerful interactive and intuitive graphical tool for enabling, configuring, and generating initialization code for the MSPM0 MCU. It helps you manage and resolve resource conflicts visually as well as provides a pin mux utility. SysConfig can be used as a standalone program with IAR, Keil, and other IDEs but it is integrated as a plugin in CCSv12.0+ and CCS Theia.

- SysConfig guide for MSPM0



gpio_toggle_output.syscfg

Type Filter Text...

- MSPM0 DRIVER LIBRARY (7)
  - SYSTEM (8)
    - Board 1/1 ✓
    - DMA
    - GPIO 1 ✓
    - MATHACL
    - RTC
    - SYSCTL 1/1 ✓
    - SYSTICK
    - WWDT
  - ANALOG (5)
    - ADC12
    - COMP
    - DAC12
    - OPA
    - VREF
  - COMMUNICATIONS (5)
    - I2C
    - MCAN
    - SPI
    - UART
    - UART - LIN
  - TIMERS (4)
    - TIMER
    - TIMER - CAPTURE
    - TIMER - PWM
    - TIMER - QEI
  - SECURITY (2)
    - AES
    - TRNG
  - DATA INTEGRITY (1)
    - CRC
  - READ-ONLY (1)
    - EVENT 1/1 ✓

Software › GPIO

GPIO (1 Added) ADD REMOVE ALL

✓ GPIO_LEDS

Name	GPIO_LEDS
Port	PORTB
Port Segment	Any

Advanced Configuration

Group Pins

3/32 added ADD REMOVE ALL

✓ USER_LED_1
✓ USER_LED_2
✓ USER_LED_3

Name	USER_LED_1
Direction	Output
IO Structure	Any

Digital IOMUX Features

Assigned Port	PORTB
Assigned Port Segment	Any
Assigned Pin	22

Interrupts/Events

Event Subscribing Channel	Disabled (0)
Output Policy	Bit will be Set
LaunchPad-Specific Pin	No Shortcut Used

PinMux Peripheral and Pin Configuration

USER_LED_1	Any(21)

Other Dependencies

Problems

There are no problems in the current design.

Generated Files

Filter: all

File name	Category	Include in build
ti_msp_dl_config.c	MSPM0 Driver Library	●
ti_msp_dl_config.h	MSPM0 Driver Library	●
Event.dot	MSPM0 Driver Library	●
gpio_toggle_output.syscfg	Configuration Script	
4 Total Files		

MSPM0G350X (Device)
LQFP-64(PM) (Package)
SWITCH

GPIO Used	5
GPIO Available	55

History

Initial state Last Save

Figure 2-1. MSPM0 SysConfig Example

2.2.3 Analog Config Tool

Analog Config is a GUI based tool that uses SysConfig and visual block diagrams to rapidly develop a MSPM0 project in a zero-code environment. With just a few clicks, users can visualize their analog signal chain, make modifications to it, and view real-time data for evaluation. Projects can be exported to CCS for further customization and evaluation, or users can flash the device directly from Analog Configurator.

- MSPM0 Analog Configurator
- Analog Configurator GUI

2.2.4 Compilers

TI Arm Clang is derived from the open-source clang compiler and is TI's preferred compiler for MSPM0. It supports TI Arm-based platforms, specifically those featuring TI Arm Cortex-M and Cortex-R series devices.

- User's guide: TI Arm Clang Compiler Tools User's Guide
- Related software development user's guides:

- ARM Assembly Language Tools User's Guide
- ARM Optimizing C/C++ Compiler User's Guide

Third-party Compilers:

- GCC ARM: GCC open source compiler
- Keil ARM Clang: ARM compiler version 6
- IAR EWARM: ARM Cortex-M edition

2.2.5 Debugging and Programming Tools

Table 2-1 compares the features of different MSPM0 debugger and programmer tools recommended by TI, and Table 2-2 compares those interfaces.

Table 2-1. Debugger and Programmer Tools

Feature	XDS110	MSP-GANG
Type	Debugger	Programmer
2-wire SWD	Yes	Yes
BSL mode	Yes	Yes
MTB Trace	Yes	No
Supported by CCS, IAR and Keil	Yes	No
EnergyTrace technology (ET)	Yes	No
Number of simultaneous programming Targets	1	8
Stand alone or script based programming	No	Yes
Stage	Development	Production

Table 2-2. Debugger and Programmer Features

Feature	SWD	ROM Bootloader (BSL)	Main Memory Bootloader Plug-in
Debug	Yes	No	No
Program	Yes	Yes	Yes
Required pins	2	2	Customizable
Protocol	2-wire	UART or I2C	Customizable
Advantages	Fewer pins and traces than JTAG	Simplest programming interface	Supports customized BSL

2.2.5.1 Debuggers/Programmers

Emulation development tools that support all MSPM0. Tools covered in this section include both debuggers and production programmers with not debugging capabilities.

2.2.5.1.1 XDS110

The TI XDS110 is the preferred debugger tool for the MSPM0 MCUs. The XDS110 enables microtrace buffer (MTB) on MSPM0 devices that support it. The XDS110 connects to the target board using a TI 20-pin connector (with multiple adapters for TI 14-pin and, Arm 10-pin and Arm 20-pin) and to the host PC through USB 2.0 High Speed (480 Mbps). The XDS110 also features two additional connections: the auxiliary 14-pin port connector that enables EnergyTrace technology, a full duplex UART port and four general-purpose I/Os, and the expansion 30-pin connector to connect the XDS110 EnergyTrace HDR add-on. MSPM0 LaunchPad kits also include embedded XDS-110 circuitry on board and can be used as programmers if needed.

- XDS110 User's Guide

Figure 2-2. XDS110 Debug Probe

2.2.5.1.2 MSP-GANG – TI Production Programmer

The MSP-GANG is a production programmer. It can program up to eight of the same MSP devices at one time, but it cannot perform real time debugging of code. It can be operated with or without being connected to a PC. The MSP Gang Programmer is not a gang programmer in the traditional sense, in that there are not eight sockets to program target devices. Instead, the MSP Gang Programmer connects to target devices that are mounted in the final circuit or system.

Figure 2-3. MSP-GANG Programmer

2.2.5.1.3 Segger J-Link and Other Third-Party Arm Debuggers

MSPM0 MCUs are not limited to the previously listed TI tools and can take advantage of the broader Arm ecosystem. MSPM0 MCUs use an Arm Cortex-M0+ core and standard SWD programming interface so third-party Arm compatible tools are also options. The available tools include IAR i-jet, Keil ULINK, P&E Micro Cyclone, Lauterbach uTrace, and more.

SEGGER J-Link debug probes are possibly the most widely used line of debug probes available today and they bring this experience to their support of the MSPM0 MCU portfolio. With up to 3MB/s download speed to RAM and record breaking flashloaders, as well as the ability to set an unlimited number of breakpoints in the MCU flash memory, the J-Link debug probes optimize the debugging and flash programming experience.

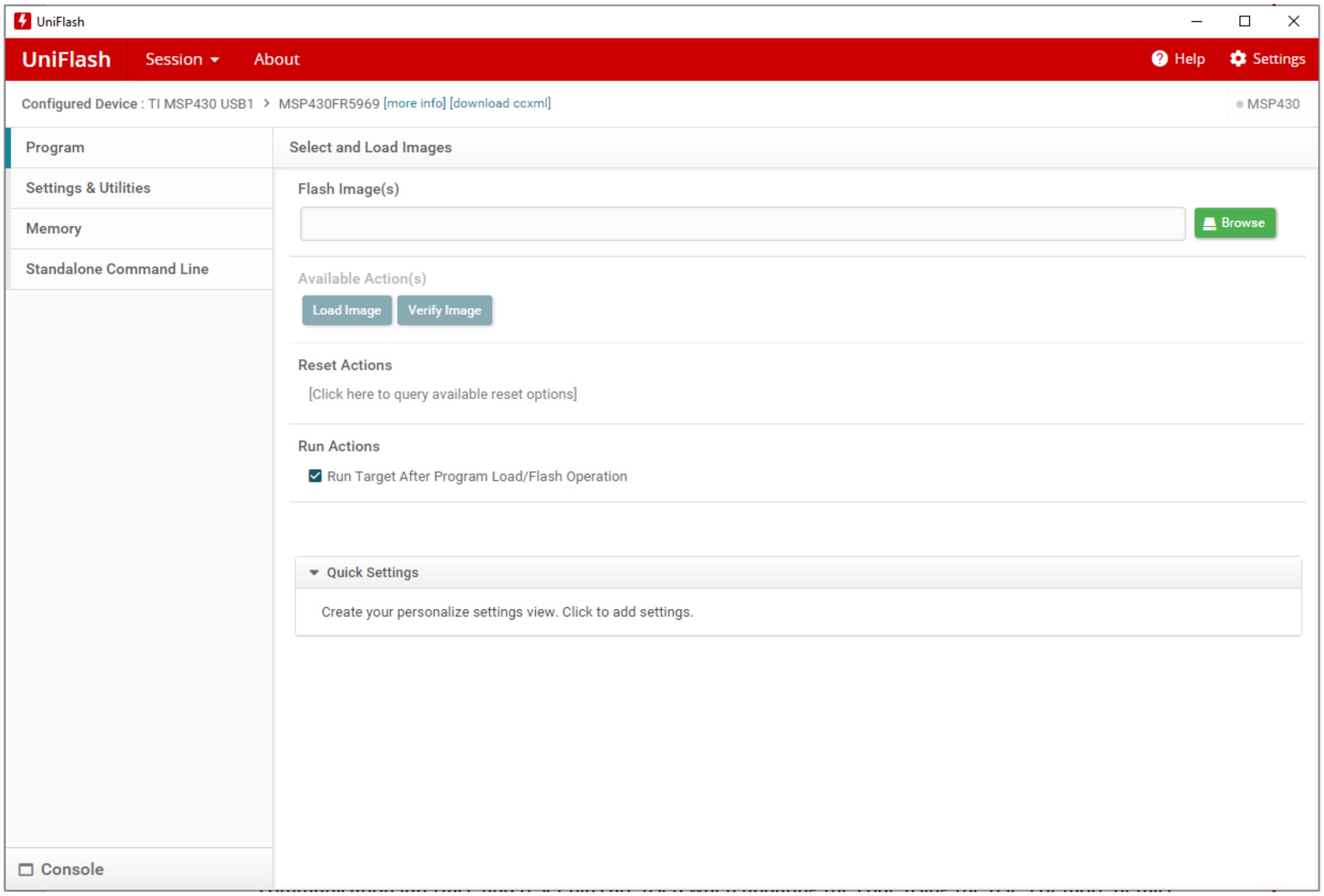
- Using Segger Programmers With MSPM0 MCUs
- J-Link/J-Trace User's Guide

2.2.5.2 Software Programming Tools

2.2.5.2.1 Uniflash

UniFlash is a standalone tool used to program on-chip flash memory on TI MCUs. Uniflash has a GUI, command line, and scripting interface. CCS Uniflash is available free of charge.

- UniFlash Guide for MSPM0
- UniFlash Quick Start Guide



Configured Device : TI MSP430 USB1 > MSP430FR5969 [more info] [download ccxml]

[Click here to query available reset options]

☑ Run Target After Program Load/Flash Operation

Create your personalize settings view. Click to add settings.

Figure 2-4. UniFlash

2.2.5.2.2 Bootstrap Loader (BSL)

The BSL is a program that is stored in the MSPM0 flash or ROM at the factory. The BSL is used for programming, erasing, and reading code stored in main memory, information memory, or RAM. The BSL communication interface and reset pins are used when updating the code using the BSL. For more details about the BSL and its communication interface, see the bootstrap loader (BSL) section in the device-specific data sheet.

- MSPM0 BSL application note
- MSPM0 BSL example in MSPM0 SDK

2.3 Embedded Software Resources

2.3.1 MSPM0 Software Development Kit (SDK)

MSPM0 SDK is a collection of software resources such as code examples and driver libraries that helps you create and build MSPM0 code efficiently and effectively. The SDK also contains detailed and comprehensive documentation to accelerate development. The MSPM0 SDK is integrated into TI Resource Explorer in CCS or online in the cloud version.

- MSPM0 SDK Documentation:
  - SDK User's guide
- Contents:
  - Peripheral drivers to enable development on MSPM0 MCUs
  - Middleware libraries to provide ready-to-use software and examples for a wide variety of applications and use-cases
  - FreeRTOS M0+ kernel to enable RTOS applications
  - Hundreds of examples from the most basic to highly integrated demos to accelerate application development
  - SysConfig Metadata to enable use of SysConfig with MSPM0 MCUs
  - Documentation and code examples showcasing the features of the device and software

2.3.1.1 Code Examples

The MSPM0 SDK is packaged with a wide selection of code examples to enable engineers to quickly develop applications. The examples folder is divided into RTOS and non-RTOS. These folders contain examples for each LaunchPad kit and are organized based on function with lower-level Driverlib examples, higher-level TI Drivers examples, and examples for middleware such as GUI Composer, LIN, IQMath, and others. Most examples support SysConfig to simplify device configuration and accelerate software development.

- MSPM0 SDK code examples:
  - MSPM0Gxx code examples
  - MSPM0Lxx code examples

2.3.1.2 Driver Library (DriverLib)

DriverLib layer consist of low-level drivers providing support for all the device features with the highest optimization for performance and low memory footprint. DriverLib contains software APIs that abstract away the details of the device hardware registers. The SDK includes dozens of examples showing how to use DriverLib APIs. DriverLib supports CCS, IAR, and Keil IDEs.

- MSP Driver Library
- DriverLib API Guide

2.3.2 Real-Time Operating System (RTOS)

MSPM0 can support a variety of open source RTOS systems but the MSPM0 SDK includes support for FreeRTOS. FreeRTOS is a market-leading RTOS. Distributed freely under the MIT open source license, and includes a kernel and a growing set of libraries for use across many different applications.

For other open source RTOS systems, visit the OSRTOS web page.

2.3.3 Subsystem Examples

MSPM0 subsystems are design resources that help to solve a common MCU design challenges. Each subsystem contains a software package and documentation that describes the design choices made in the subsystem. While MSPM0 code examples aim to show how a particular feature works, MSPM0 subsystems show how different peripherals can be used together to perform a task. The accompanying documentation explains the design intent and inner workings to help engineers make modifications to meet their system requirements. The goal of MSPM0 subsystems is to help engineers speed up their prototyping and design phase by providing high quality references for typical microcontroller use cases.

- MSPM0 Subsystem application briefs can be found in Section 3.1
- MSPM0Gxx subsystem software examples

- MSPM0Lxx subsystem software examples

2.4 Hardware Tools and EVMs

2.4.1 Development Boards

MSPM0 has a simple ecosystem of development boards, shown in Table 2-3. The LaunchPad kit can be used across several stages of your design.

Table 2-3. Development Boards

Feature	LaunchPad Kit	BoosterPack Module
Available on TI.com	Yes	Yes
Onboard debugger	Yes	No
Pinout	Basic	Basic
Advantages	Inexpensive, easy-to-use	Plug-in module compatible with most LaunchPad kits
Stages	Evaluating, developing	Rapid prototyping with digital or analog sensors

The TI LaunchPad kits are low-cost development boards developed by TI that include an onboard debugger. Compared to other development boards, LaunchPad kits support a diverse ecosystem of plug-in modules, called BoosterPack modules. Different LaunchPad kits and BoosterPack modules can be connected together to create a larger system. Typically, a LaunchPad kit is the first choice for evaluating or developing MSP-based applications.

- MSPM0Gxx LaunchPad kit
- MSPM0Lxx LaunchPad kit

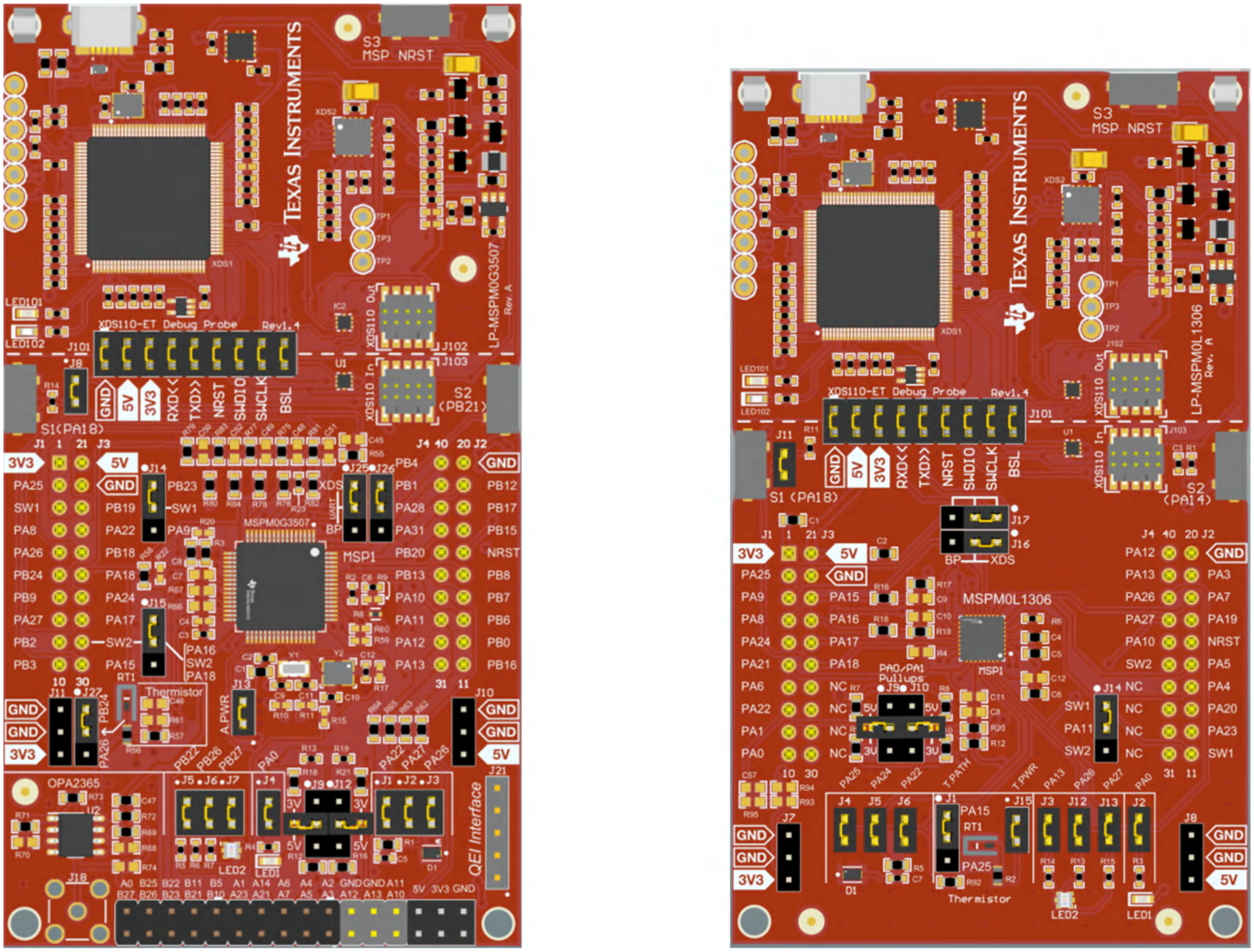


Figure 2-5. MSPM0Gxx and MSPM0Lxx LaunchPad Kits

3 MSPM0 Application Resources

3.1 Application Notes

The following application notes describe MSPM0 MCUs and peripherals and can be referenced while developing your design.

- Migration Guides
  - STM32 to MSPM0 Migration Guide
  - MSP430 to MSPM0 Migration Guide
- MSPM0 Subsystems
  - PWM LED Driver
  - Programmable gain amplifier
  - Transimpedance amplifier
  - 5-V Interface
  - ADC with DMA in ping pong mode
  - Thermistor temperature sensor
- Hardware
  - Troubleshooting Guidelines for MSP Devices
  - MSP 32-kHz Crystal Oscillators
  - MSPM0Gx MCUs Hardware Development Guide
  - MSPM0Lx MCUs Hardware Development Guide
- Analog
  - Making system design easy with MSPM0 precision analog
  - A Glossary of Analog-to-Digital Specifications and Performance Characteristics
  - General Oversampling of MSP ADCs for Higher Resolution
  - High-Speed, Analog-to-Digital Converter Basics
  - Implementing a Thermocouple Interface With ADC12_A
- Communication
  - Understanding the I2C Bus
- Power
  - Low-power optimization guide for MSPM0 G-series MCUs
  - Low-power optimization guide for MSPM0 L-Series MCUs
- ESD
  - Electrostatic Discharge (ESD)
  - MSP System-Level ESD Considerations

3.2 Application-Specific Resources

This section contains MSPM0 resources targeting specific applications that can also be referenced while developing your design.

Sector	Application
Power Delivery	• Battery managements
Building Automation	• Smoke detectors
Factory Automation	• Field sensors
Medical	• Thermometers • Pulse-ox
Motor Control	• Power and garden tools • Motor control: Trap • Motor control: FOC • Motor control: H-bridge

Sector	Application
Personal Electronics	• Gimbals • TWS chargers

4 Acronyms and Definitions

Acronym	Definition
ADC	Analog-to-digital converter
AES	Advanced encryption standard
Arm	Arm Limited
BSL	Bootstrap loader
CCS	Code Composer Studio™ IDE
CPU	Central processing unit
DSP	Digital signal processing
ECDSA	Elliptic curve digital signature algorithm
ESD	Electrostatic discharge
ET	EnergyTrace™ technology
EVM	Evaluation module
FAQ	Frequently asked questions
GUI	Graphical user interface
IDE	Integrated development environment
JTAG	Joint Test Action Group
KB	Kilobytes
MCU	Microcontroller
MSP	Mixed-signal processor
NVM	Nonvolatile memory
OPA	Operational amplifier
OS	Operating system
PC	Personal computer
RAM	Random access memory
ROM	Read-only memory
RTOS	Real-time operating system
SDK	Software development kit
SMT	Surface mount
SWD	Serial wire debug
TIA	Transimpedance amplifier
TS	Target socket
TRM	Technical reference manual
TRNG	True random number generator

5 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

DATE	REVISION	NOTES
February 2023	A	Initial Public Release

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265