Accelerate time to market with MSPM0 MCU ecosystem

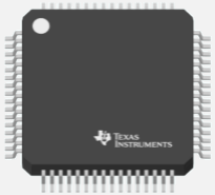
Amruta Deole

J.D. Crutchfield

Welcome to the MSPM0 MCU webinar!

In today's session we will dive into the MSPM0 hardware and software ecosystem.

Today's agenda



- MSPM0 MCU ecosystem
- Hardware overview
- Software overview
  - Code development
  - Migration tools
  - MSPM0 SDK middleware
- Q&A

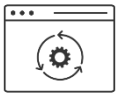
Portfolio and hardware

MSPM0 MCU ecosystem

Code source:

MSPM0-SDK

Code generation:



SysConfig

ZeroCode

IDEs:

CCSv12 (Eclipse)

CCS

ARMKEIL
Microcontroller Tools

IAR SYSTEMS

PC programming tool:

UniFlash

Debugger:

XDS110

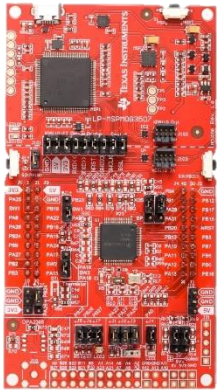
J-LINK

Production Programmer:

C-GANG

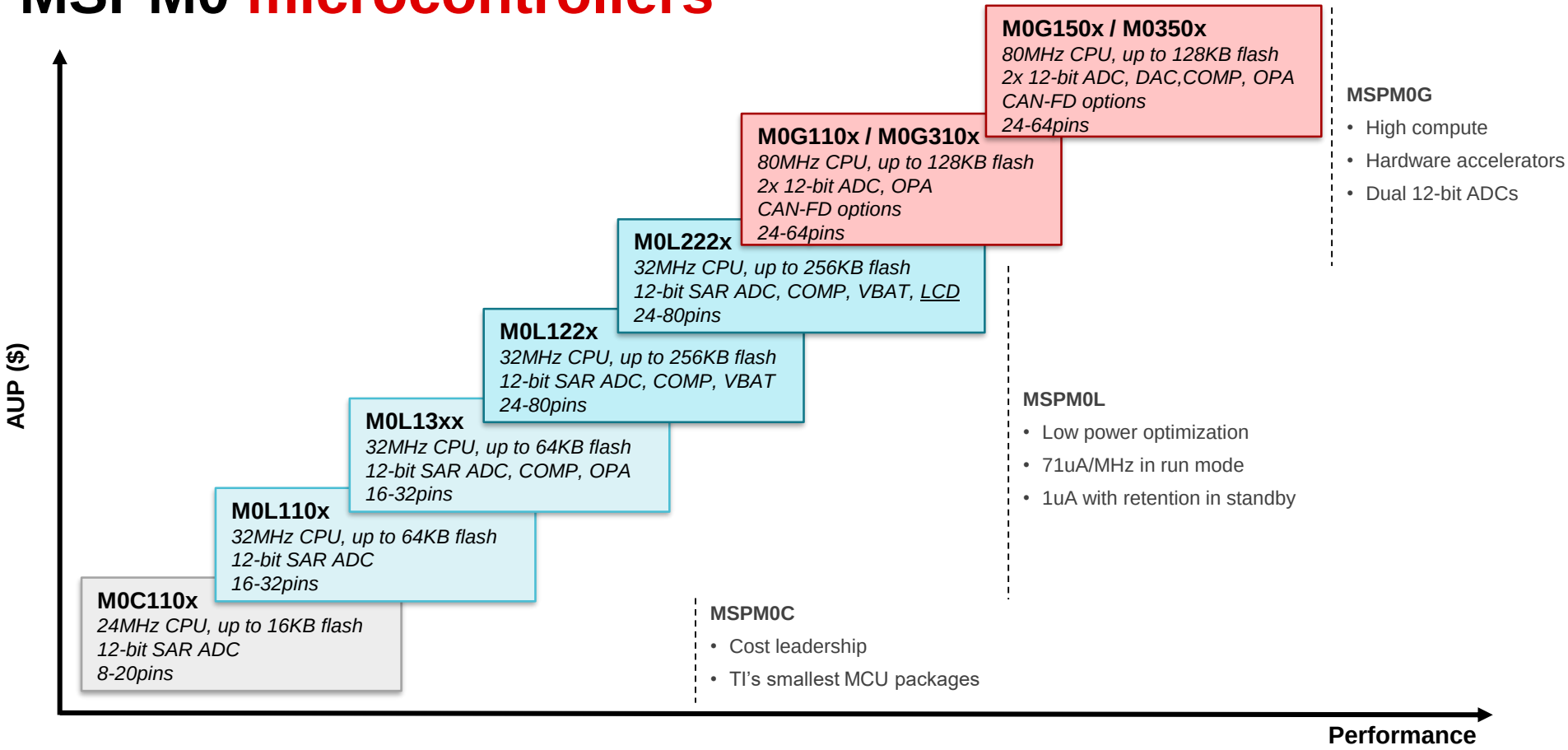
MSPM0 MCU:

Development board:



LaunchPad

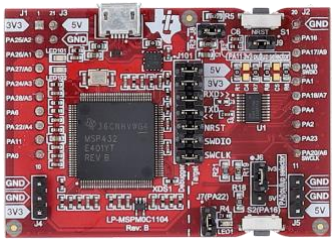
MSPM0 microcontrollers



MSPM0 MCU evaluation boards

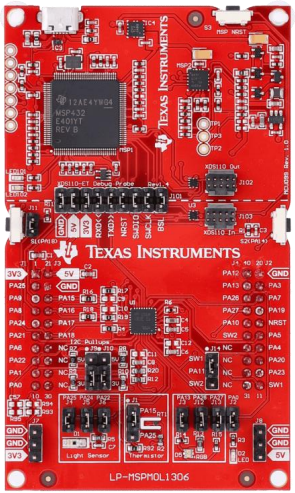
MSPM0C LaunchPad

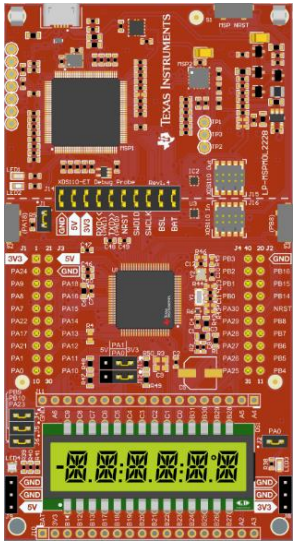
- LP-MSPM0C1104



MSPM0L LaunchPads

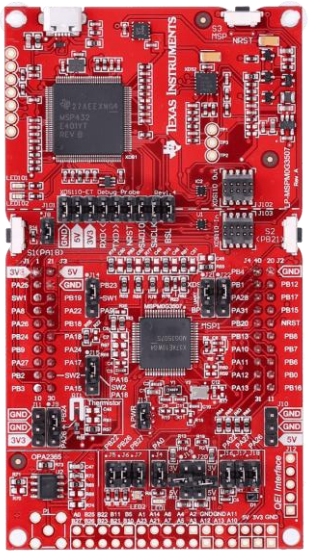
- LP-MSPM0L1306
- LP-MSPM0L2228





MSPM0G LaunchPad

- LP-MSPM0G3507



MSPM0 LaunchPad ecosystem

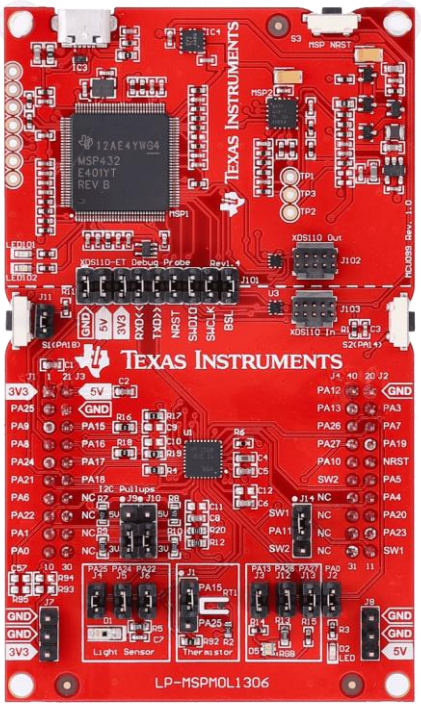
An easy to use prototyping platform

Integrated debugger

- Built in XDS110 supports real time debugging with breakpoints
- Back-channel UART sends data back to PC console
- Removable jumpers to program and debug other hardware

BoosterPack header

- Access to device pins through 40pin BoosterPack header
- Plug in a BoosterPack EVM to quickly connect to sensors, LCDs, and other devices



LP-MSPM0L1306 pictured

EnergyTrace

- Collect accurate power usage measurements
- Read EnergyTrace power state data from MSPM0 device

Onboard analog

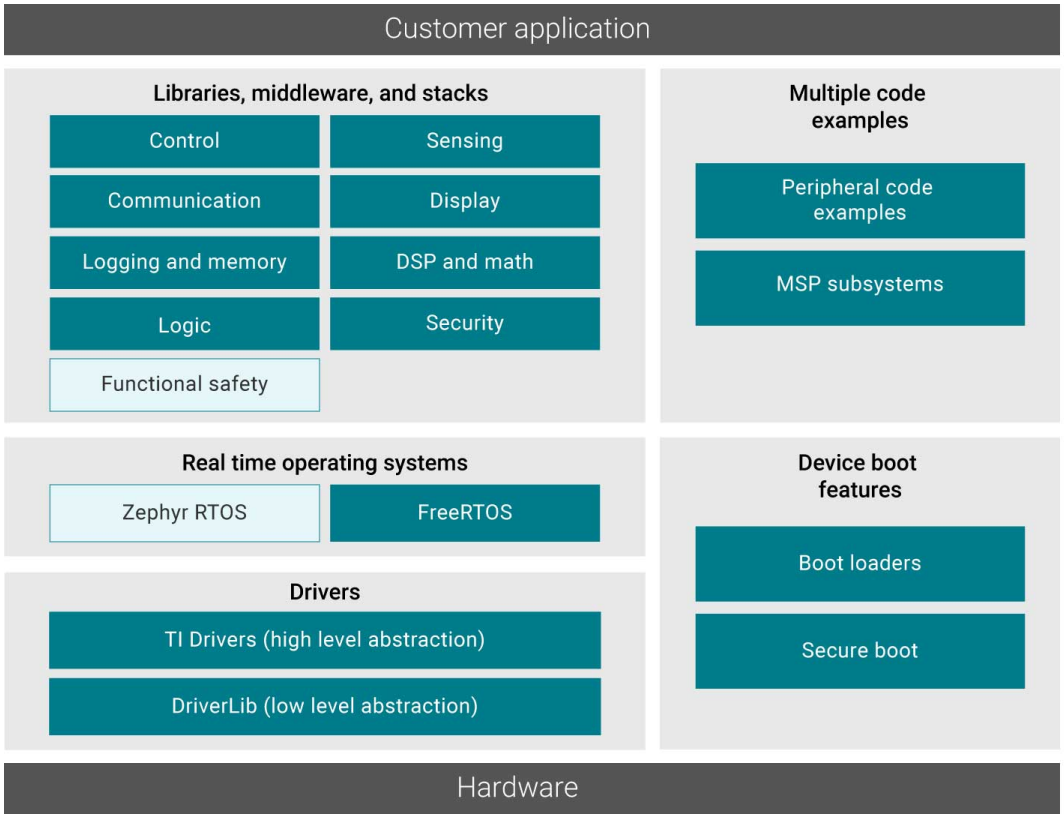
- Use onboard thermistor, light sensor and RGB LED to prototyping analog functionality directly on the LaunchPad

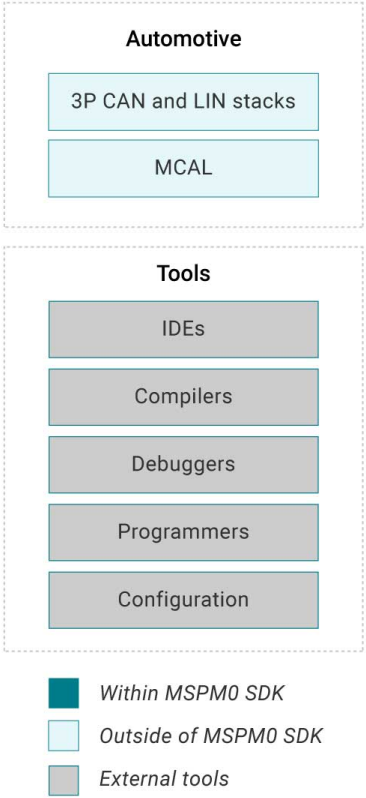
Order today!

- Available on ti.com

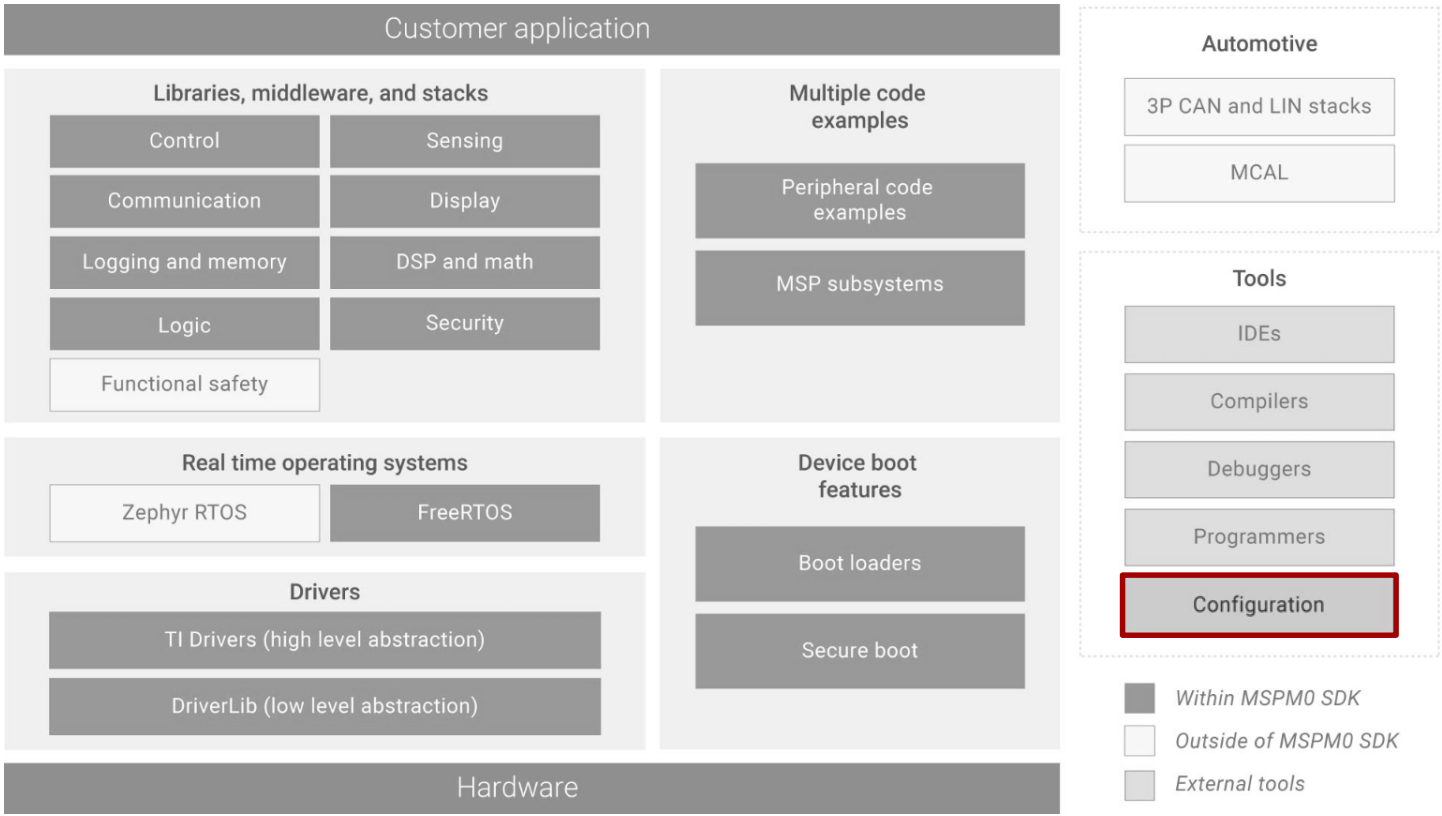
Software overview

MSPM0 SDK | All the software you need in one place



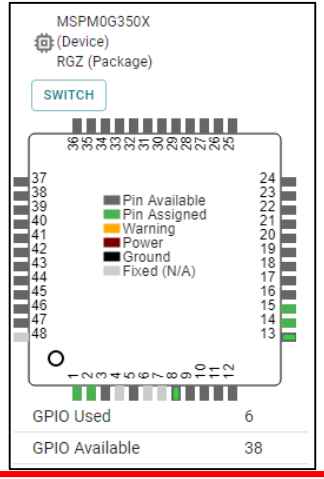


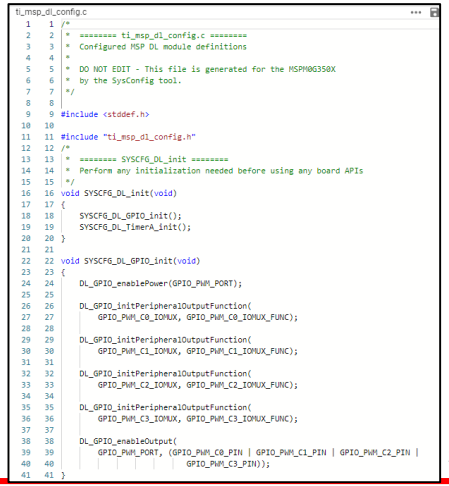
MSPM0 code development tools



SysConfig | intuitive GUI-based configuration

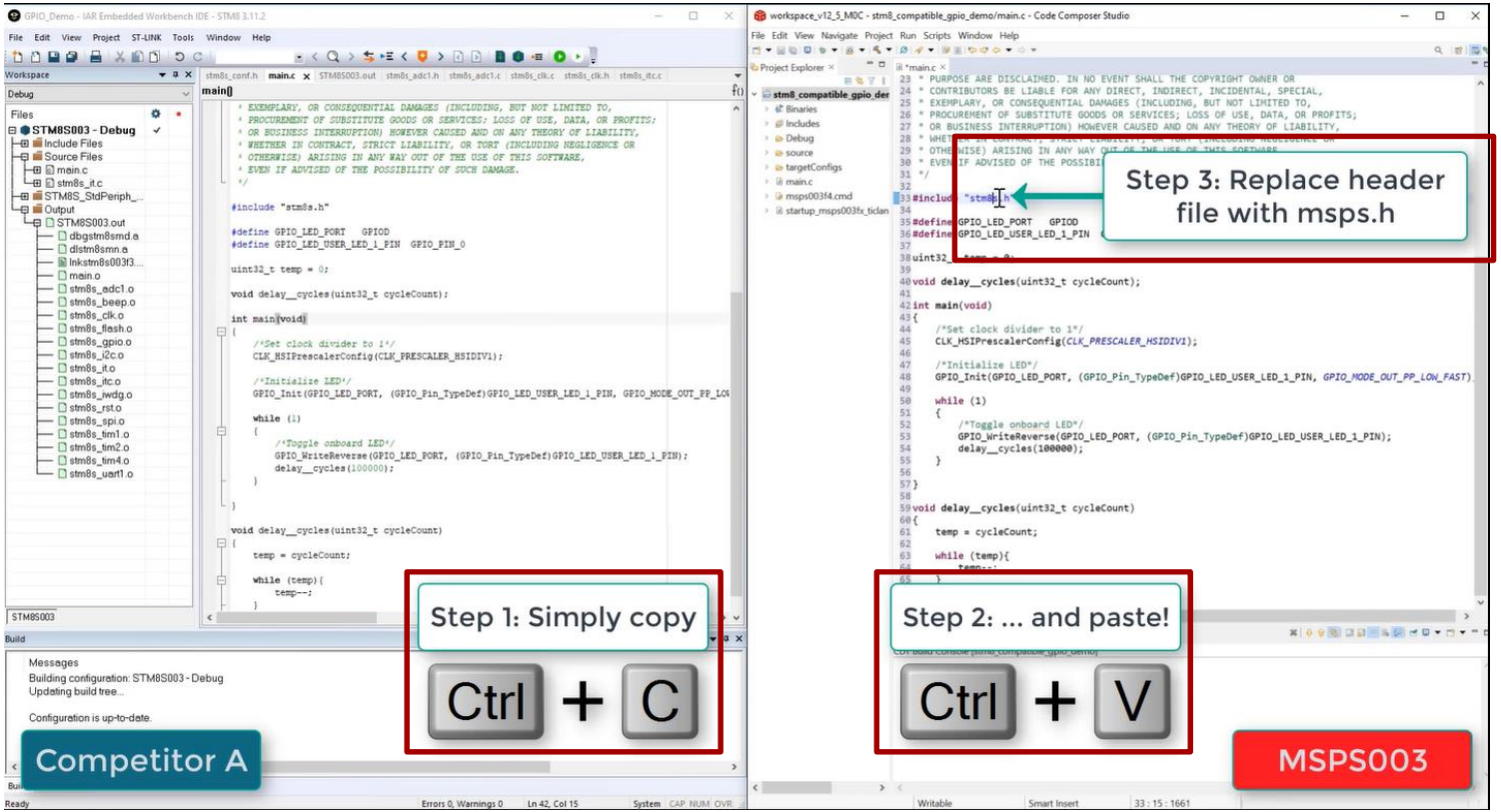
- Graphical utilities to configure pins, peripherals, subsystems, and other components
- Generates C code files for SDK examples or custom software
- Real-time conflict management ensures creating a valid configuration
- Real-time code preview
- Board and device-level views to show pin and peripheral configurations
- Planned library support for:
  - DriverLib
  - TI Drivers
- Tool support:
  - Standalone Desktop and Cloud application
  - Integrated in CCS, IAR and Keil IDEs



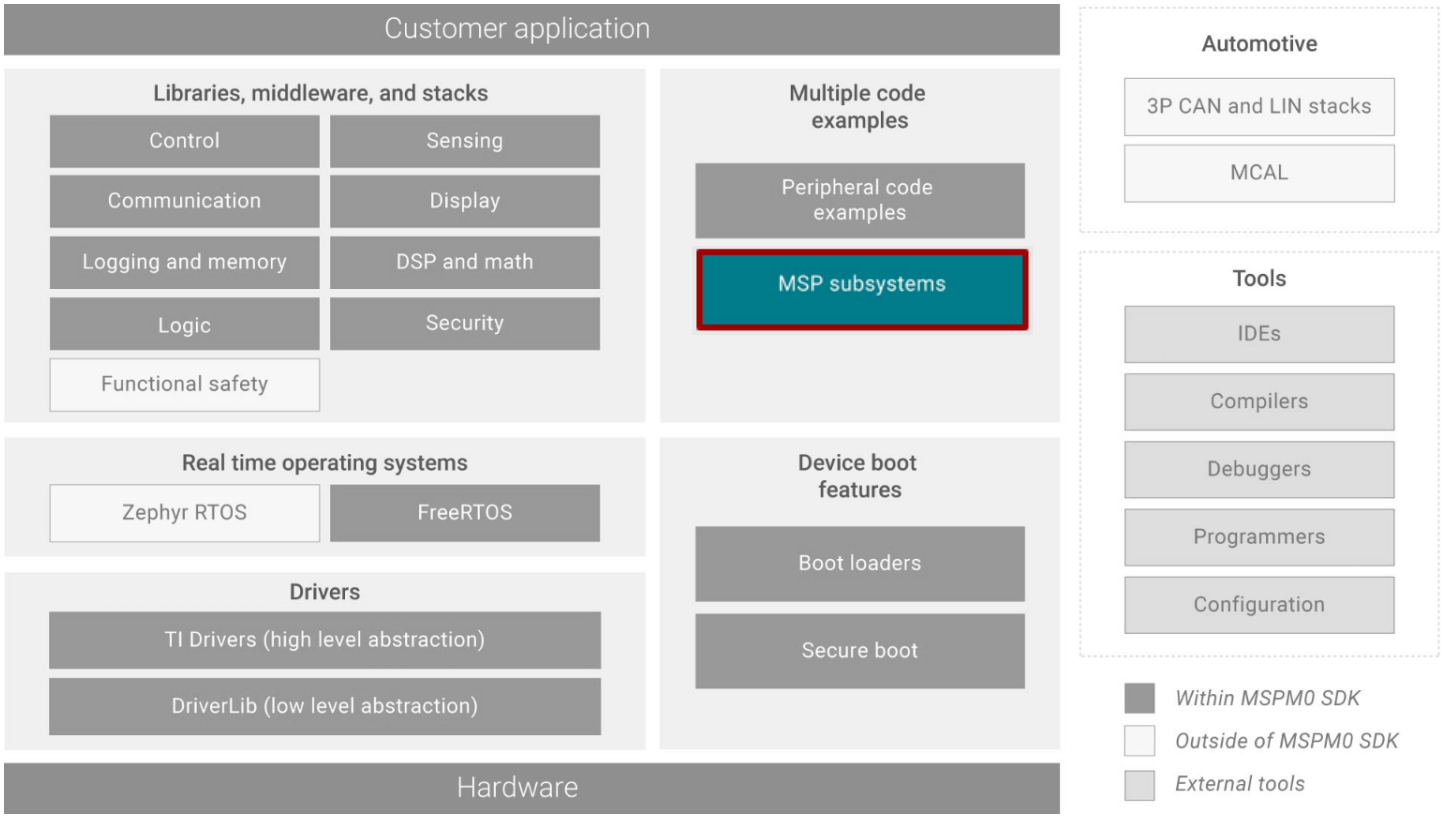


SysConfig Demo Video

Simplified code migration from competing devices



MSP MCU subsystems



What are subsystems?

Building blocks for common MCU design challenges

Functional Block Diagrams + Application Code Examples + Design Steps + Software Flow Chart + Design Considerations + Future Enhancements

Subsystem Design

Dynamic Programmable Gain Amplifier

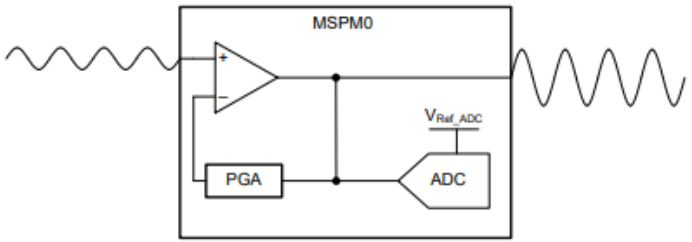


Figure 1-1. Subsystem Functional Block Diagram

```
#include "ti_msp_dl_config.h"

#define HIGHMARGIN 3890 // 4095*0.75 = 75% of max ADC value
#define LOWMARGIN  1638 // 4095*0.25 = 25% of max ADC value
#define MAXGAIN  DL_OPA_GAIN_N7_P8 // Maximum GAIN level of OPA wanted
#define MINGAIN  DL_OPA_GAIN_N1_P2 // Minimum GAIN level of OPA wanted.
//For non-inverting PGA mode this is an OPA GAIN of 2x. See advisory in TRM for MIN GAIN.

int main(void)
{
    SYSCFG_DL_init();

    //Set ADC Window Compare thresholds
    DL_ADC12_disableConversions(ADC12_0_INST);
    DL_ADC12_configWinCompHighThld(ADC12_0_INST, HIGHMARGIN);
    DL_ADC12_configWinCompLowThld(ADC12_0_INST, LOWMARGIN);
```

Faster development time | Save on software resources

MSPM0 MCU subsystem availability

General functionality

Subsystem	M0G	M0L	M0C
Digital mux	✓	✓	✓
Task scheduler	✓	✓	✓
Diode matrix	✓	✓	✓
Power sequencer	✓	✓	✓
LED driver with PWM	✓	✓	✓
FIR (digital filters)	✓		
IIR (digital filters)	✓		
RTC scheduler	✓	✓	✓
Interfacing with 5V	✓	✓	✓

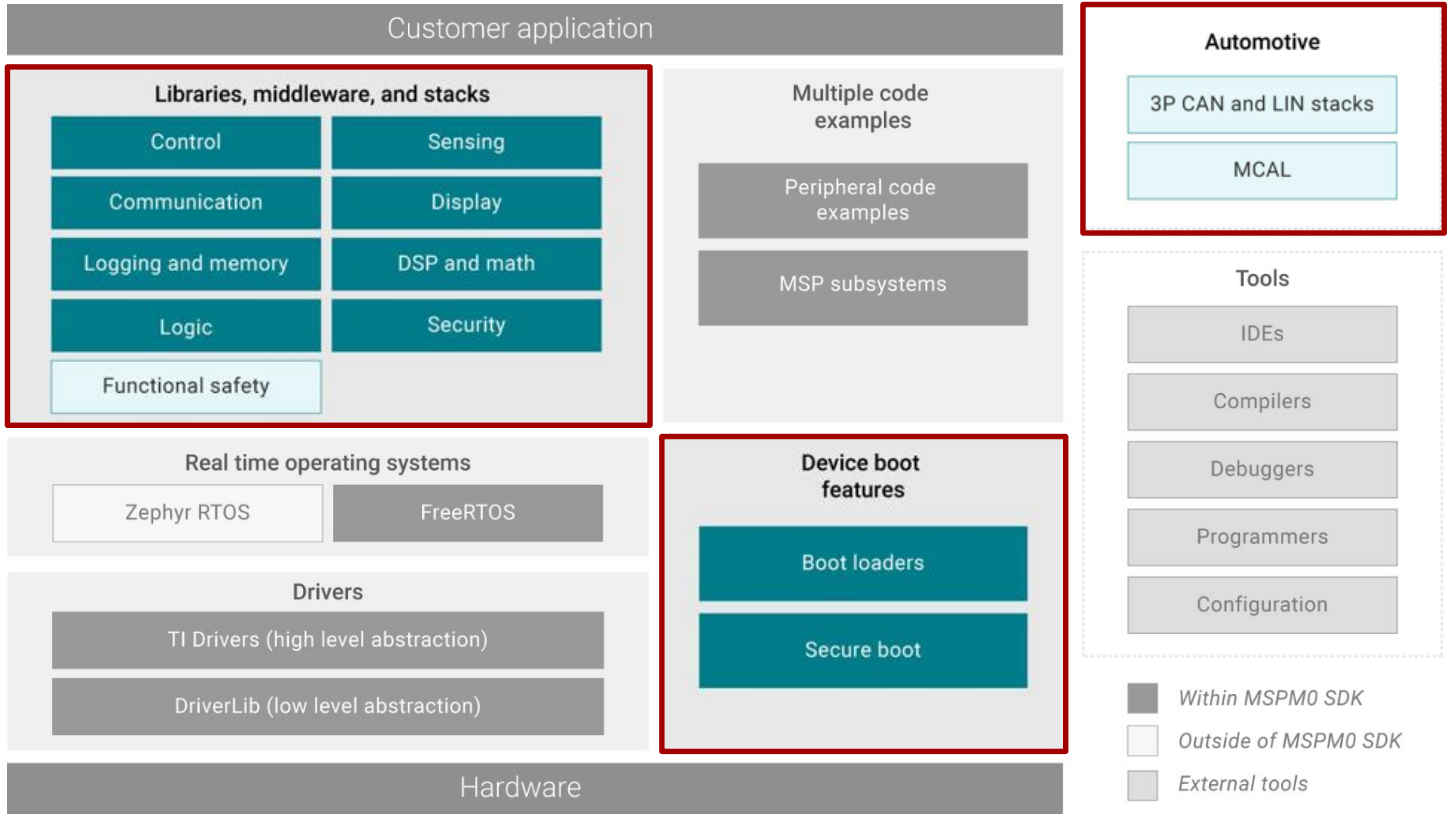
Communication bridges

Subsystem	M0G	M0L	M0C
UART to I2C	✓	✓	✓
CAN to UART	✓		
CAN to SPI	✓		
CAN to I2C	✓		
I2C to UART	✓	✓	✓
UART to SPI	✓	✓	✓

Analog functions

Subsystem	M0G	M0L	M0C
ADC to PWM	✓	✓	✓
PWM DAC	✓	✓	✓
ADC with DMA ping pong	✓	✓	
ADC with SPI interface	✓	✓	✓
ADC with I2C interface	✓	✓	✓
ADC with UART	✓	✓	✓
Data sensor aggregator	✓	✓	✓
PGA	✓	✓	
Transimpedance amplifier	✓	✓	

MSPM0 middleware



MSPM0 SDK middleware

Control

- Brushed motor control library
- Hall sensored trap motor control library
- Sensorless FOC library
- Stepper motor control library

Display

- LCD Demo
- GUI composer library

Security

- Boot image manager examples

Functional safety

- MSPM0 diagnostic library for FuSa

Communication

- LIN library
- CAN stack
- SENT library
- SMBus library
- PMBus library
- DALI
- One-wire demo

DSP and math

- CMSIS DSP code examples
- IQmath library

Logic

- Power sequencing

Sensing

- PIR motion detection demo
- SASI smoke detector demo
- Dualray smoke detector demo
- Battery gauge code examples
- Battery monitor code examples
- Blood pressure monitor demo
- Pulse oximeter demo
- Energy metrology library

Logging and memory

- EEPROM emulation
- Custom bootloader code examples

MSPM0 motor control resources

Software

Run software examples in MSPM0-SDK for motor control from TI Resource Explorer

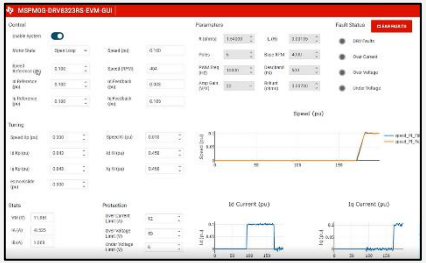
- MSPM0-SDK
- TI Resource Explorer

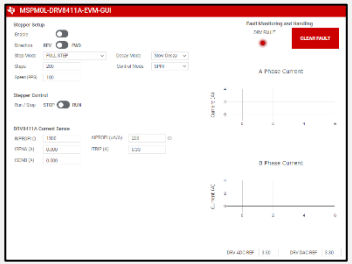
↓

- CCS
- CCS Cloud
- IAR
- Keil

GUI

Spin motors with easy-to-use GUIs from the TI Gallery



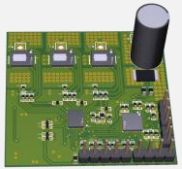


Hardware

Use available hardware solutions on ti.com to run simple code examples with LaunchPad or onboard MSPM0

DRV8x EVM or BoosterPack

TIDA-010250 for High Voltage 3-phase Field-oriented Control

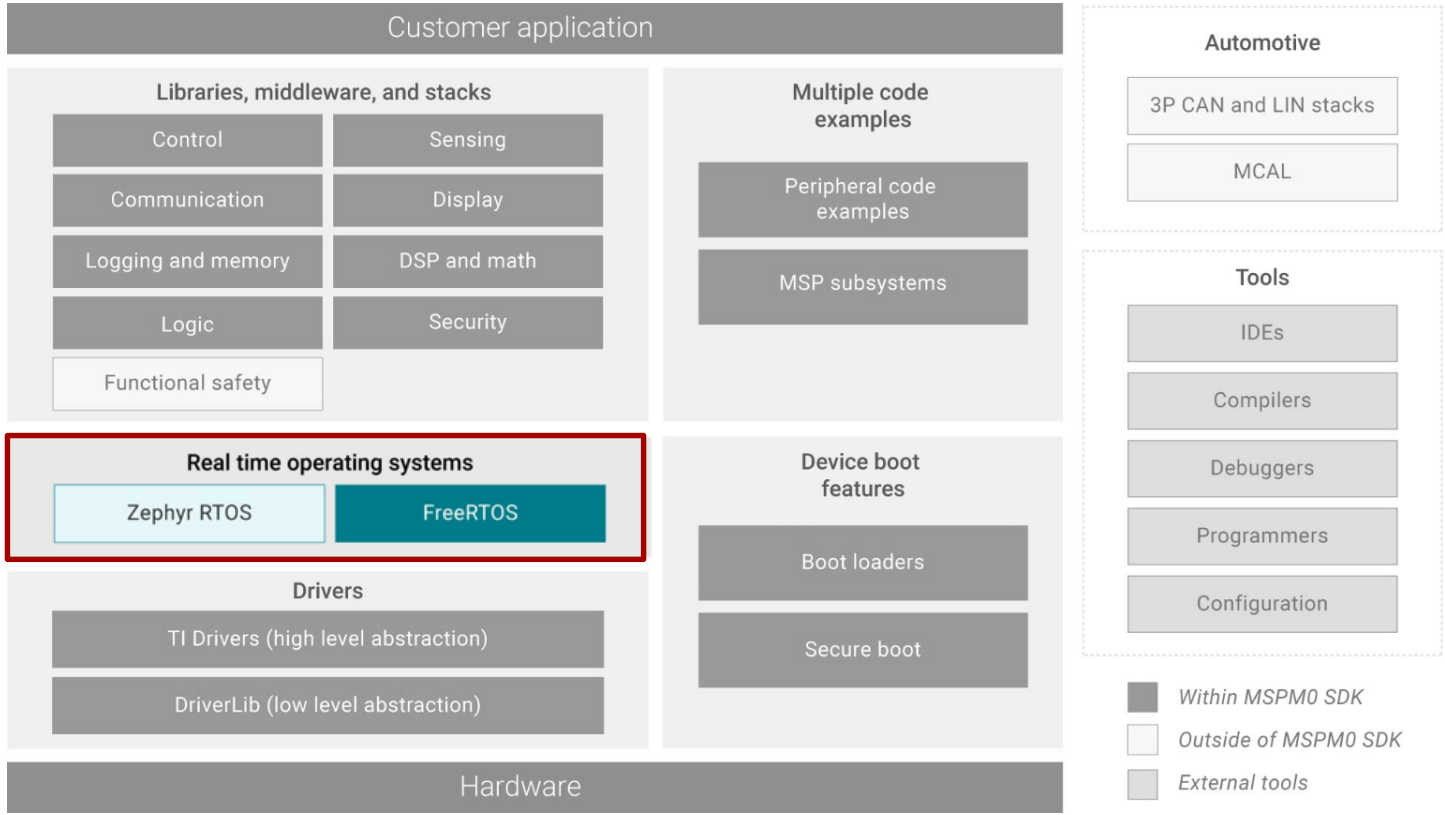


TIDA-010251 for Low-voltage motor control

User documentation

User documentation and guides on TI Resource Explorer

Real-time operating systems



Operating systems

Zephyr RTOS

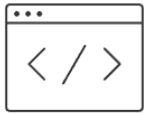
- Robust and secure kernel which supports multiple scheduling algorithms
- Backed by the Linux Foundation (long-term sustainability and vendor-neutral governance)
- Current MSPM0 support for a wide variety of peripherals, and more under development
- Available at Zephyr GitHub

FreeRTOS

- Market leading RTOS for MCUs
- Minimalist function set, basic task handling, and memory management
- All TI Driver RTOS examples are supported with FreeRTOS

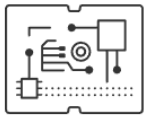
Additional resources

MSPM0 Software Resources:



- MSPM0 Software Overview
- MSPM0-SDK — MSPM0 Software Development Kit (SDK)
- TI Resource Explorer – View SDK and examples online
- Subsystems – Common MCU Designs

MSPM0 Hardware Design Guides:



- MSPM0 G-series Hardware Dev Guide
- MSPM0 L-series Hardware Dev Guide
- MSPM0 C-series Hardware Dev Guide

Direct Support:

- MSPM0 E2E Forum

MSPM0 training and development resources:

- MSPM0 MCUs Quick Reference Guide
- MSPM0 Academy — Self-pace training module
- TI Precision Labs – MCUs - Training video series

Migration tool:

- Simplified code migration tool

Migration guides:

- Migration Guide From STM32 to Arm-Based MSPM0
- STM8 to MSPM0C Migration Guide
- Microchip to MSPM0 Migration Guide
- Migration Guide From Renesas RL78 to Arm-Based MSPM0

Q&A

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