

Open Source, Arduino, and the Path to Production-Ready IoT

October 22, 2026

2026
tech talks
WEBINAR SERIES



OPEN
SOURCE

Agenda

- 01** Why open development and communities matter
- 02** Silicon Labs support for open development and communities
- 03** Arduino introduction, boards, protocols
- 04** Getting started + demo
- 05** Community in action
- 06** Your questions / resources

How to move from concept to product?

Developers need breadth in how they discover, evaluate, and develop



Open development speeds up the journey



Discover and evaluate faster

Working boards, code and examples - inspect real implementations



Reduce integration friction

Start from maintained examples, not a blank project



Debug in the open

Known issues, visible fixes - real-world use surfaces edge cases



Reuse, learn and innovate

Build on the ecosystem's work - a boost for smaller teams too

Production still needs



Performance testing, optimizations



Security vulnerability management



Long-term maintenance



Documentation and support

Silicon Labs supports the whole journey - from first example to production

Developer ecosystem mapping

Developers now combine vendor tools, open projects and community expertise - and switch between them as needs change

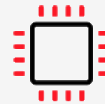
Silicon Labs connects the path between vendor supplied and open ecosystems



**Developer
communities**



**Open source
projects**



**Third-party
boards**



**Vendor SDKs
and tools**

Zephyr

Open source RTOS

Linux Foundation

**Zephyr Project Platinum
member**

Matter

Open application
standard

*Connectivity Standards
Alliance.*

**Promoter Member /
Board of Directors**

Thread

Mesh network standard,
OpenThread by Google

Thread Group / Google

**Sponsor Member / Board
of Directors**

Arduino

Open source developer
hardware and software
platform

**Silicon Labs Arduino
Nano Matter board and
GitHub repo**

GitHub

Software management
and collaboration
platform

**Silicon Labs Software
Repo**

Silicon Labs Open Source Community

Silicon Labs is making it easier for developers to build on working application examples, collaborate through GitHub, and contribute improvements to examples and documentation that can be incorporated into future SDK releases.

1 BUILD

Start from Silicon Labs applications

2 EXTEND

Raise issues, suggest fixes, open pull requests



Build. Use. Pressure test. Share. Contribute. Review. Improve.

PHASE 1



Reviewed and tested: accepted contributions go through Silicon Labs engineering review and testing before they can become part of a future SDK release.

Phase 1 is Bluetooth LE - more wireless technologies to follow over time.

From prototype to production: what developers get, and where

Open resources and community expertise connect to supported SDKs, tools and support as a product matures

STAGE	WHAT AND WHERE	WHAT DEVELOPERS GET
01 Build in the open	Arduino Core for Silicon Labs - Bluetooth LE, Matter and Zigbee libraries and examples, open on GitHub, Edge AI ML SDK foundation Bluetooth LE examples and docs on GitHub - issues and pull requests, now open to contributions Zephyr upstream - board, SoC, HAL and driver enablement Matter and OpenThread - open, interoperable implementations	Start from working code - a wireless prototype in minutes - learn from other developers and contribute back
02 Ship and maintain	Simplicity SDK and Simplicity SDK for Zephyr - tagged releases with documented quality levels Simplicity Studio 6 + VS Code - configurators, analysis and debug tools Simplicity AI SDK – connects developers and their AI coding assistants to the Simplicity Ecosystem Silicon Labs Certifications - certified Matter and Thread solutions Silicon Labs Community and technical support	Security, power, certifications and long-term maintenance for a production product

Easy to start, with a path to scale: the same Silicon Labs hardware from the first sketch to the product

Open source is in Arduino's DNA



OPEN SINCE DAY ONE

Open hardware

designs and schematics anyone can study, modify and reuse

Open software

IDE, cores and libraries developed in the open on GitHub



A GLOBAL COMMUNITY

34M+ users

from students and makers to engineers at companies using Arduino PRO

Shared knowledge

forum, libraries and projects built by the community



ZEPHYR OS

Arduino Core for Zephyr

1.0 since Sep 2026, replacing Mbed OS

Maker to pro

the RTOS industry ships; Arduino is a Zephyr Silver member



BETTER WITH SILICON LABS

We bring

ease of use and millions of developers

Silicon Labs brings

wireless silicon, production-grade stacks, pro tooling and the Arduino Core for Silicon Labs



ARDUINO NANO MATTER

Prototype to production, one board

Built for **Matter over Thread**, with Bluetooth LE and now Zigbee. Start in the Arduino IDE, grow into Zephyr or Simplicity Studio, with no hardware change.

NEW · ZIGBEE

First official Arduino board with Zigbee

New Arduino Zigbee library with 14 ready-to-run examples, plus Zigbee coordinator firmware

- Bluetooth LE
- Thread
- Matter
- Zigbee



MODULE	Silicon Labs MGM240S
CPU	Cortex-M33, DSP + FPU, 78 MHz
MEMORY	1536 kB Flash · 256 kB RAM
CONNECTIVITY	Thread · BLE · Matter · Zigbee
INTERFACES	I2C · SPI · UART · PWM · ADC · DAC
ON BOARD	RGB LED · buttons · USB-C debug
SIZE · POWER	18 × 45 mm · 3.3 V I/O · 5 V in
CERTIFIED	Matter · Bluetooth · FCC · IC · CE



AVAILABLE

Bluetooth LE

Core libraries and examples



AVAILABLE

Matter over Thread

Matter library and examples



NEW

Zigbee

Zigbee library, 14 examples

Plus ready-made firmware: OpenThread RCP and Zigbee NCP images turn a Nano Matter into a Thread radio or a Zigbee coordinator

Arduino APIs, libraries and examples are open source; the radio stacks ship as precompiled Silicon Labs SDK libraries. Silicon Labs Arduino Core 5.0.0.



One board, four software stacks

Start in the Arduino IDE. When the product needs more control, move to **native Zephyr** or **Simplicity Studio** - both built for professional development, with no hardware change.

WHY GO PRO

What the pro paths add

Multi-threading and task scheduling, advanced Bluetooth LE stack features, power management, security - plus configurators and analysis tools

START



ARDUINO IDE

Arduino Core for Silicon Labs

Bluetooth LE, Matter over Thread and Zigbee libraries and examples - open on GitHub

Also on Nano Matter: Arduino Core for Zephyr 1.0 - Bluetooth LE and peripherals; no Thread, Matter or Zigbee yet



GO PRO · PICK BY PRODUCT NEEDS



PRO PATH 1

Native upstream Zephyr

Upstream board since Zephyr 4.2, contributed by Silicon Labs; west, Kconfig, devicetree, RTOS threads



PRO PATH 2

Simplicity Studio 6 + VS Code

Full Simplicity SDK: Bluetooth LE, Matter, Thread, Zigbee; Silicon Labs OpenOCD extension to flash and debug

Path to product: the Nano Matter itself, or your own board with the same module



Getting started: eight boards, one Arduino Core

From board setup to a first wireless application in a familiar four-step flow

DEMO BOARD



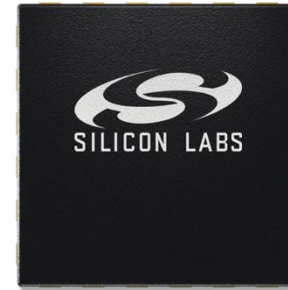
**Arduino Nano Matter
MGM240S**



**Seeed Studio XIAO
MG24**



**SparkFun Thing+
Matter MGM240P**



**Silicon Labs xG24,
xG27, BGM220 kits**

PARTNER BOARDS

Arduino Nano Matter
SparkFun Thing Plus Matter
Seeed Studio XIAO MG24 (Sense)
Ezurio Lyra 24P 20 dBm Dev Kit

SILICON LABS KITS

xG24 Explorer Kit
xG24 Dev Kit
xG27 Dev Kit
BGM220 Explorer Kit



1 Install

Arduino IDE 2.x and the Silicon Labs platform from Boards Manager



2 Select

Board and programmer in the Tools menu



3 Open

Blink first, then a wireless library example



4 Build and run

Flash, check serial output, change one thing, rebuild

Demo: wireless in the Arduino IDE, in minutes



Arduino Nano Matter

Silicon Labs MGM240S module

STEP 1



Arduino IDE + Bluetooth LE

- ✓ Board + stack: Bluetooth LE (Silicon Labs)
- ✓ Upload the ble_blinky example
- ✓ Connect over Bluetooth LE, then toggle the LED
- ✓ One small change, upload again

STEP 2



Professional next step

- ✓ Same board, native Zephyr
- ✓ Or Simplicity Studio 6 + VS Code
- ✓ No hardware change

Wireless is built-in: the Arduino Core runs Bluetooth LE, Matter over Thread, and Zigbee.

Community in action

Silicon Labs enablement lives in the open - in the Arduino core, upstream Zephyr and the Bluetooth LE examples on GitHub



Find

Start from a maintained example - e.g. the Bluetooth LE examples on GitHub



Reproduce

Run the smallest case on the exact board



Ask and triage

Share logs, versions and a minimal reproducer



Share

Document the fix, or open a pull request - the Bluetooth LE examples now accept them



Arduino Core: explore and prototype

Silicon Labs maintains board support, wireless libraries and examples on GitHub - issues, discussions and pull requests



NEW

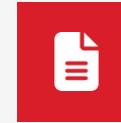
Bluetooth LE: open for contribution

Example apps (Zlib) and the Bluetooth LE docs on GitHub - issues and pull requests, reviewed and tested before they ship



Zephyr upstream: build products

Silicon Labs contributes SoC, HAL, driver and board enablement for professional development



IN PROGRESS

Matter in Zephyr: open by design

Zephyr module proposal (#104127), upstream Matter pull requests merged Sept 2026, a Zephyr lighting example in the public Matter SDK

Recap

Explore. Build. Contribute.

Arduino lowers the barrier to wireless exploration.

Build and extend through open source collaboration and contributions.

The same Silicon Labs hardware continues into multiple development environments.

Protocol maturity differs by path - check each release.

Where to go next

ARDUINO

github.com/SiliconLabsSoftware/arduino

github.com/arduino/ArduinoCore-zephyr

forum.arduino.cc

ZEPHYR

docs.zephyrproject.org

github.com/SiliconLabsSoftware/zephyr-silabs

docs.silabs.com/zephyr/latest

SILICON LABS

docs.silabs.com

community.silabs.com

silabs.com/software-and-tools/github

[Blog: From prototype to production](#)