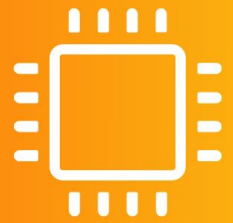
Welcome

EFR and EFM: An optimized platform for AI/ML at the (Tiny) Edge

Andrew Halstead

tech talks



WIRELESS COMPUTE

Agenda

01 Introduction

02 What is AI/ML at the Edge? And why has it become important?

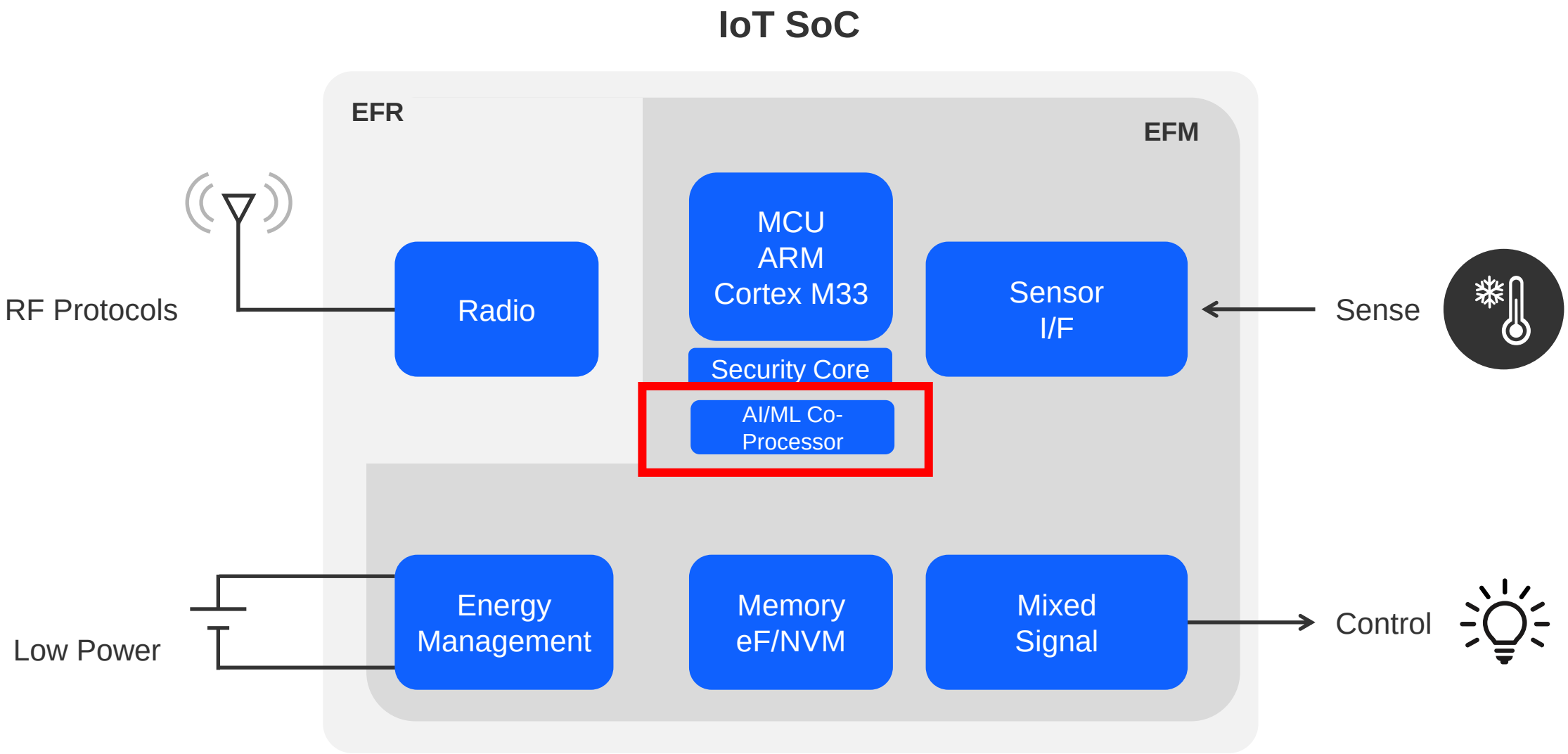
03 What is Machine Learning in an Embedded Context?

04 How Silicon Labs is enabling AI/ML through Hardware?

05 How Silicon Labs is enabling AI/ML through Software?

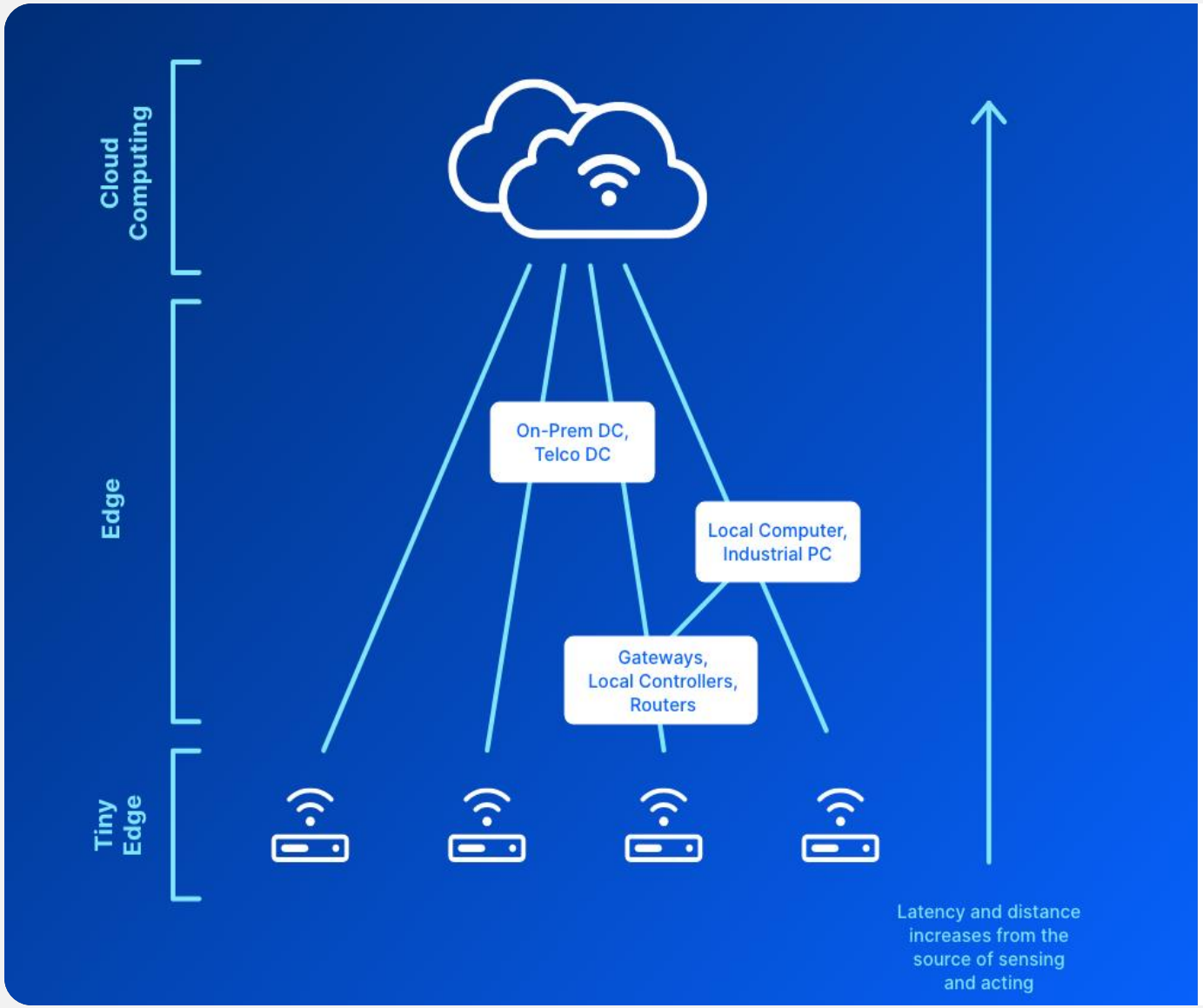
Introduction

100% IoT Focused Company



What is AI/ML at the Edge? And Why has it become Important?

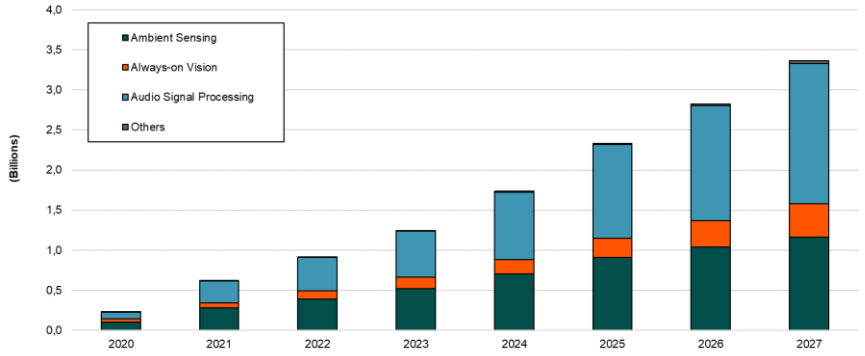
Artificial Intelligence(AI) and Machine Learning(ML) at the Tiny Edge



Key Benefits

- Low Latency
- Privacy, IP Protection, Security
- Bandwidth Constraints
- Offline Mode Operation
- Cost Reduction

>3B Devices sold with TinyML in 2027



Source: ABI Research, Artificial Intelligence and Machine Learning, 2 QTR 2022

SILICON LABS

Why Machine Learning on Microcontrollers?

Low Latency Required

- Mission or safety-critical applications require real-time reactions
- Large data to process - typically at vision use cases - no time to upload to anywhere to process

Privacy and IP Protection, Security

- Data never leaves the sensing device, only inference result/metadata is transferred
- Less sensitive data to transmit, less chance to be hacked
- Protecting IP

Bandwidth Constraints

- Long range, low power, and slow networks can't transfer all TimeSeries data to process somewhere else
- Overloading of mesh network is an issue
- Large data to chunk e.g. hi-res images

Offline Mode Operation

- Local system keeps operating standalone in case of any network issue
- Connectivity is occasional or blocked by admin

Cost Reduction

- Network and infrastructure costs
- Data ingestion costs
- Data storage costs
- Cloud services
- Ops, maintenance
- Compact edge with ML solutions integrated to wireless SoC
- Cheaper devices

Power constraints

- Ultra-low power applications
- Always-on systems
- Healthy tradeoff in transmit to higher level compute vs. locally process

Data processing is more efficient with Machine Learning at the sensor level

What is Machine Learning in an Embedded Context?

The Terms…

- **Artificial Intelligence**
  - Can be broadly defined as the effort to automate intellectual tasks normally performed by humans. Most notably however, it does not necessary involve learning.
  - The sub-sets of AI are ML, DL, and Generative AI.
- **Machine Learning**
  - Is a subset of artificial intelligence (AI) that focuses on the development of algorithms and statistical models without explicit programming instructions
  - Allow computers to learnin from data, could be considered a sophisticated pattern matching technique.
  - The fields of supervised/unsupervised learning apply here.
- **Deep Learning**
  - A complex subset of machine leaning that describes a set of algorithms that have a logical structure closely resembling the neurons in a human brain.
  - It is called ‘Deep’ because of the successive transformation on data i.e. processing layers.
- **Generative AI**
  - Refers to a class of artificial intelligence systems that are capable of generating new content that is similar to examples in the dataset they were trained on.
  - Use techniques like deep learning and neural networks to learn patterns and relationships in data and then generate new content based on those patterns

Where is the drivable lane?

- **A rules-based approach: intractable problem**

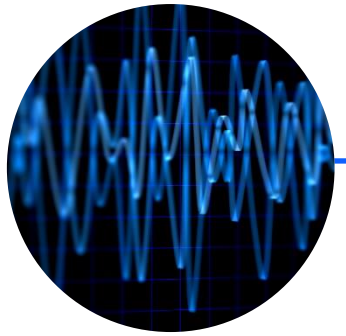





Event Detection using Machine Learning

Sensors
- Acceleration, Temperature, Current/Voltage
- Time-series data on ADC or GPIO

ML methods based on Time-series Data
- Data anomaly detection
- Data pattern matching



Microphones

Analog or Digital
- Audio mic array with beamforming
- Audio mic input with Audio Front End, DSP

ML methods based on Audio
- Audio pattern matching (ex. glass break)

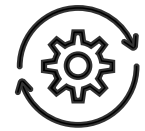
ML methods based on Voice
- Wake word/command word detection

Camera

Low resolution imaging
- Image capture (including fingerprint reader)

ML methods based on Vision
- Fingerprint reading
- Always-on vision – object detection
- Image classification and detection

Event Detection



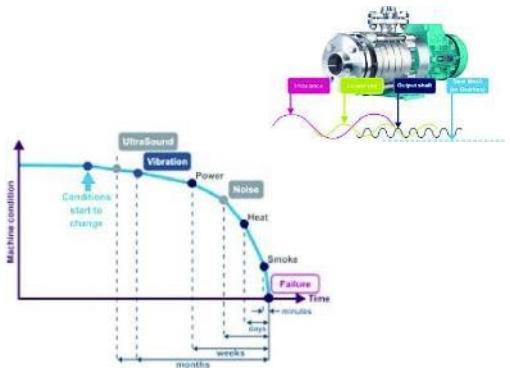
Act

Communicate

Machine Learning Application Examples on a CortexM4, M33

Wireless SoC's typical/recommended Resource needs with ML applications in Order of Magnitudes

RAM: 64kB
Ops/s: 5M-40M



SENSOR

Signal Processing (time series, low-rate data)

- Predictive/Preventative Maintenance
- Anomaly detection (e.g. air quality, abnormal usage, leak detection)
- Condition based monitoring – machine health, Cold chain monitoring, Battery monitoring
- Bio-signal analysis -healthcare and medical (e.g., pulse detection, EKG)
- Accelerometer use-cases e.g., fall detection, pedometer, step counting
- Agricultural use-cases (e.g. cow health)

RAM: 128kB
Ops/s: 40M-100M

AUDIO

Audio Pattern Matching

- Security applications e.g., Glass break, scream, shot detection
- Cough detection
- Machine malfunction detection
- Breath monitoring

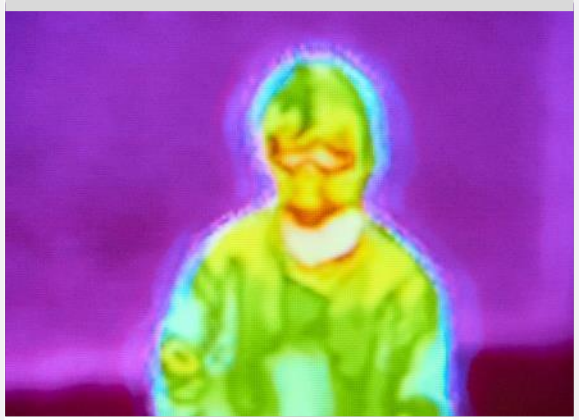
RAM: 256kB
Ops/s: 50M-500M

VOICE

Voice Commands

- 10 words command set for smart appliance
- Wake-word detection (Always-On voice)
- Smart device voice control
- Voice assistant

RAM: 256kB
Ops/s: 200M-1.5G w /hardware accelerator



VISION

Low-resolution vision

- Wake-up on object detection (always-on)
- Presence detection
- People counting, people-flow counting
- Movement detection
- Smart city monitoring (e.g. Parking spot)
- Fingerprint matching

How Silicon Labs is enabling AI/ML through Hardware.

Silicon Labs Machine Learning Solution Benefits

- Industry's widest portfolio of wireless solutions combined with ML for Tiny Edge devices
  - Bluetooth, 802.15.4/ZigBee/Thread, Matter, Z-Wave, Prop, Wi-Sun, Sidewalk, WiFi
- Integrated ML hardware accelerator (xG24, xG28) provides up to 8X faster ML inferencing with 1/6th of energy
  - Reduces BOM, footprint and design complexity while minimizing latency
- ML development tools and solutions for explorers to experts for faster application development
  - TensorFlow Lite Micro supported in GSDK
  - Partnerships with Edge Impulse, SensiML and MicroAI accelerate embedded ML development
  - Silicon Labs' ML Tool Kit on GitHub provides complete control & flexibility for the expert developers
- Wide range of use cases including low data rate sensors, audio/voice and low-res images

End-to-End Machine Learning Solution for Wireless IoT Edge Devices

BG24 and MG24: Optimized for Battery Powered IoT Mesh Devices

SOCS AND MODULES

SOC DEVICE SPECIFICATIONS

High Performance Radio
- Up to +19.5 dBm TX
- -97.6 dBm RX @ BLE 1 Mbps
- -105.4 dBm RX @ 802.15.4

Efficient ARM® Cortex®-M33
- 78 MHz (FPU and DSP)
- Up to 1536kB of Flash
- Up to 256kB of RAM

Matrix Vector Processor
- AI/ML Accelerator

Low Power
- 5.0 mA TX @ 0 dBm
- 19.1 mA TX @ +10 dBm
- 4.4 mA RX (BLE 1 Mbps)
- 5.1 mA RX (802.15.4)
- 33.4 µA/MHz
- 1.3 µA EM2 with 16 kB RAM

Security
- Secure Vault Mid/High
- ARM ® TrustZone ®

SOC DEVICE SPECIFICATIONS

Low-power Peripherals
- EUSART, USART, I2C
- 20-bit ADC, 12-bit VDAC, ACMP
- Temperature sensor +/- 1.5°C
- 32kHz, 500ppm PLFRCO

World Class Software
- Matter[1]
- Thread[1]
- Zigbee[1]
- Bluetooth (1M/2M/LR)
- Bluetooth mesh
- Dynamic multiprotocol[1]
- Proprietary

Wide Operating Range
- 1.71 to 3.8 volts
- +125ºC operating temperature

Multiple Package Options
- 5x5 QFN40 (26 GPIO)
- 6x6 QFN48 (28/32 GPIO)

DIFFERENTIATED FEATURES

Integrated Power Amplifier
- +19.5 dBm output power

AI/ML accelerator
- Accelerates inferencing while reducing power consumption

Secure Vault High
- Protects data and device from local and remote attacks

20-bit ADC
- 16-bit ENOB for advance sensing

PLFRCO
- Eliminates need for 32 KHz crystal

Why FG28?

- **Dual-Band (Sub-GHz + 2.4 GHz) Support with Series 2 Performance**
  - Increased processor performance over FG1x devices including AI/ML hardware accelerator
- **Multiprotocol Support**
  - Support for static and dynamic multiprotocol use cases for select Sub-GHz and Sub-GHz + Bluetooth scenarios
- **Broader Ecosystem Support for Low- power Devices**
  - Full support for Wi-SUN LFN low-power nodes
  - Support for both Bluetooth LE and FSK PHYs for Amazon Sidewalk
- **Up to 49 GPIOs for Better System Integration**
  - Eliminate system complexity by incorporating more into FG28 (QFN68)
- **Migration Path from Earlier FG Devices**
  - Footprint compatible path from FG12 (QFN68) and FG23 (QFN48)

xG28: Single or Dual Band SoC for the Next Generation of IoT

Single or Dual Band

More GPIOs

DEVICE SPECIFICATIONS

High Performance Dual Band Radio

- Up to +20 dBm Sub-GHz Output Power
- -125.8 dBm Rx Sensitivity @ 915 MHz 4.8 kbps O-QPSK
- Up to +10 dBm 2.4 GHz Output Power
- -94.2 dBm Rx Sensitivity @ BLE 1 Mbps

Efficient ARM® Cortex®-M33

- Up to 78 MHz
- Up to 1024kB Flash, 256kB RAM

Low Power

- 82.8 mA TX Current (915 MHz, +20 dBm)
- 26.2 mA Tx Current (915 MHz, +14 dBm)
- 4.6 mA RX (915 MHz 4.8 kbps O-QPSK)
- 22.5 mA TX Current (2.4 GHz +10 dBm)
- 5.2 mA RX (BLE 1 Mbps)
- Active Current: 33 µA/MHz @39 MHz
- 1.3 µA EM2 (16 kB Retained) / 2.8 µA EM2 (256 kB Retained)

Protocol support

- Wi-SUN
- Amazon Sidewalk
- CONNECT
- Wireless M-BUS
- Proprietary
- Bluetooth LE

Package Options

- 6x6 QFN48 (31 GPIO)
- 8x8 QFN68 (49 GPIO)

DIFFERENTIATED FEATURES

Single and Dual Band Support

- Supports Sub-GHz and Sub-GHz + Bluetooth LE

Large memory footprint

- Support larger stacks or applications in a single chip

AI/ML accelerator

- Faster inferencing with lower power

Secure Vault™ Mid and High options

- Flexible platform for evolving security needs

+20 dBm output power

- Eliminates the need for an external power amplifier

16-bit ADC

- Up to 14-bit ENOB for better analog resolution

Preamble Sense

- Ultra low power receive mode

Antenna Diversity

- 6-8 dBm better link budget (Sub-GHz only)

Segment LCD

- 4x48 segment LCD

High GPIO count

- Support up to 49 GPIO

BG24 and MG24: Optimized for Battery Powered IoT Mesh Devices

Sensing at the Edge

AI/ML Hardware Accelerator Key Features

- Optimized Matrix processor to accelerate ML inferencing with a lot of processing power **offloading the CPU**
- Real and complex data
- **up to 8x faster** inferencing over Cortex-M
- Up to **6x lower power** for inferencing
- **Math library** to accelerate matrix ops

Low-Power SoCs and Modules Optimized for Battery Powered IoT Mesh Devices

High Performance Radio
-Up to +19.5 dBm TX
-97.6 dBm RX @ BLE 1 Mbps
-105.7 dBm RX @ BLE 125 kbps
-104.5 dBm RX @ 15.4
Improved Wi-Fi Coexistence
RX Antenna Diversity

Low Power
5.0 mA TX @ 0 dBm
19.1 mA TX @ +10 dBm
4.4 mA RX (BLE 1 Mbps)
5.1 mA RX (15.4)
33.4 µA/MHz
1.3 µA EM2 with 16 kB RAM

World Class Software
Simplicity Studio 5
Matter[1]
Thread[1]
Zigbee[1]
Bluetooth (1M/2M/LR)
Bluetooth mesh
Dynamic multiprotocol[1]
Proprietary

ARM® Cortex®-M33
78 MHz (FPU and DSP)
Trustzone®
Up to 1536kB of Flash
Up to 256kB of RAM

Dedicated Security Core
Secure Vault™ - Mid
Secure Vault™ - High

Low-power Peripherals
EUSART, USART, I2C
20-bit ADC, 12-bit VDAC, ACMP
Temperature sensor +/- 1.5°C
32kHz, 500ppm PLFRCO

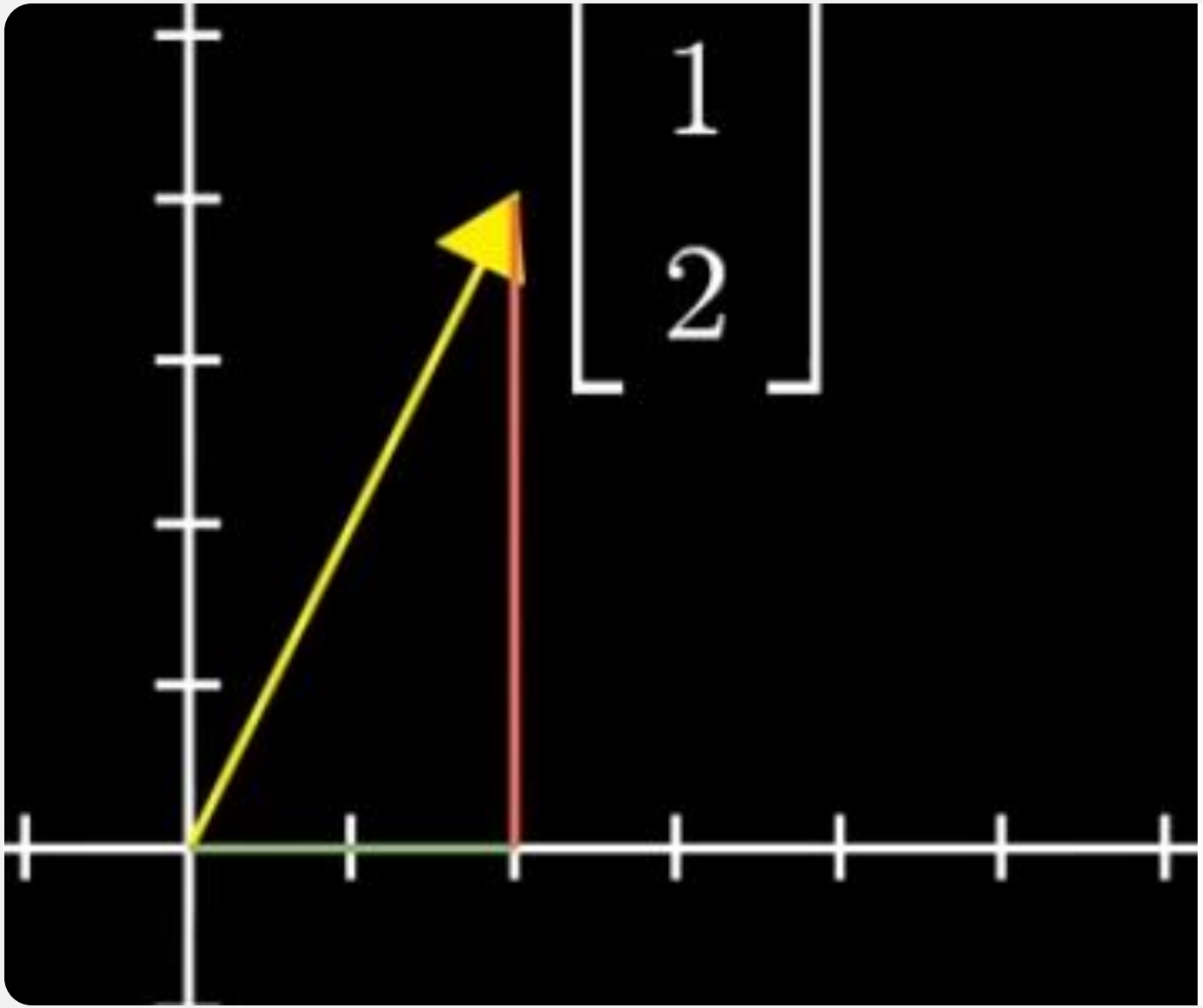
AI/ML
AI/ML Hardware Accelerator

SoCs and Modules
5x5 QFN40 (26 GPIO) -125ºC
6x6 QFN48 (28/32 GPIO) -125ºC
7x7 SiP Module (+10 dBm)
12.9x15.0 PCB Module (+10 dBm)

[1]Requires MG24

Introducing the MVP

Some Terms…



- **MVP – stands for Matrix-Vector Processor**
- **What is a ‘Vector’?**

  In a ML context, it is and ordered collection of elements in a row or column matrix.
- **What is a ‘Matrix’?**
  - In linear algebra they are used to represent linear transformations. The elements within

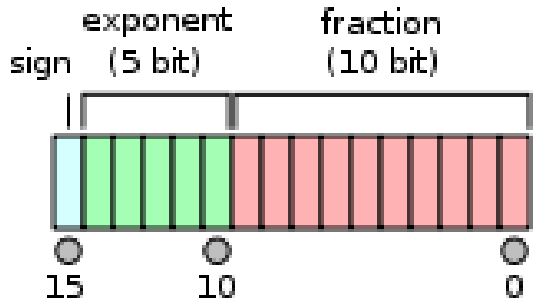
  a matrix are the coefficients of the transform.
- **Why are Vector and Matrices relevant in Machine Learning?**
  - Allows for the representation of datasets.
- **What is Matrix-Vector multiplication (dot-product)**
  - M x N matrix multiplied by a N x 1 vector = M x 1 vector.
  - Applies a transformation on a vector to give a new vector.
- **Why are these relevant in Machine Learning?**
  - We want to make transformations on data.
  - In the training of a model, our goal is to use data to find the optimal parameters of a function

  to minimise a cost function. These parameters are represented in Matrices.
  - In inference, we have established the coefficients of these functions and apply

  them to input data which is represented as vectors/matrices.

What is the MVP? When is it useful?

- **Fundamentally, the MVP performs floating point operations very efficiently in hardware.**
  - Why not just use the M33 core?
- **It's native processing numeric format is IEEE 745-2008 half-precision (16-bit) floating point numbers.**
  - Why are floating point numbers relevant in ML?
- **How is this format different from integer format?**
  - FP16 numbers are represented in scientific notation, in base 2.
- **What types of application benefit from the MVP?**
  - Any applications that use mathematical operations that involve floating point.
  - Another application outside of ML that benefits is AoA.



Technical Details of the MVP

Resource	Cortex-M33	MVPv1	Description
Data Buses	1x32-bit read/write bus	2x32-bit read buses 1x32-bit write bus	MVPv1 has an advantage moving data---but only an advantage when moving 2 or 3 streams of data simultaneously
I/O data type	8, 16, and 32-bit integers 16/32-bit floating point	8-bit integers 16-bit floating point	MVPv1's limited native type support limits the data that it can read/write to memory and may require conversions by M33 to deal with other data types efficiently
Computation types supported by hardware	8, 16, 32-bit integers 32-bit floating point	16-bit floating point	MVPv1 only supports 16-bit floating point in its ALU, thereby limiting precision of intermediate values and output
Instruction Location	Sequenced instructions	8 macro instruction records	The M33 has advantages to being able to program up highly complex sequences; MVPv1 limits itself to for-loop constructs, but in doing so can perform calculations with nearly ideal bus utilization and speed, and removes the performance overhead of maintaining loop indices
Data Organization Restrictions	Data can be organized in highly complex ways, where software defines the access pattern through instructions	Data is defined in a flexible matrix/array/tensor format requiring a regular pattern of storage that can be specified in terms of independent per-dimension strides	With MVPv1's restricted access pattern, its control logic can update array indexes in parallel with operations, and removes the performance overhead of maintaining array indices
Single Instruction, Multiple Data (SIMD)	4 bytes	2x16-bit floating point numbers (or 2 converted int8 integers)	When doing byte operations, the M33 has an advantage. MVP has native complex MAC support, increasing its effective operations per cycle for those operations.

The MVP Math library

- **Accelerate and do more efficiently linear algebra operations with internal MVP subsystem**
- **Math APIs (alternative to CMSIS_DSP) available in GSDK Alpha, GA release in 23Q2**

VECTOR OPERATIONS

- Vector Add
- Vector Absolute Value
- Vector Clip
- Vector Dot Product
- Vector Multiply
- Vector Negate
- Vector Offset
- Vector Scale
- Vector Sub
- Complex Vector Conjugate
- Complex Vector Dot Product
- Complex Vector Magnitude
- Complex Vector Magnitude Squared
- Complex Vector Multiply
- Complex Vector Multiply Real
- Vector Copy
- Vector Fill

MATRIX OPERATIONS

- Matrix Initialize
- Matrix Multiply
- Matrix Scale
- Matrix Sub
- Matrix Transpose
- Matrix Multiply Vector
- Matrix Add
- Complex Matrix Multiply
- Complex Matrix Transpose

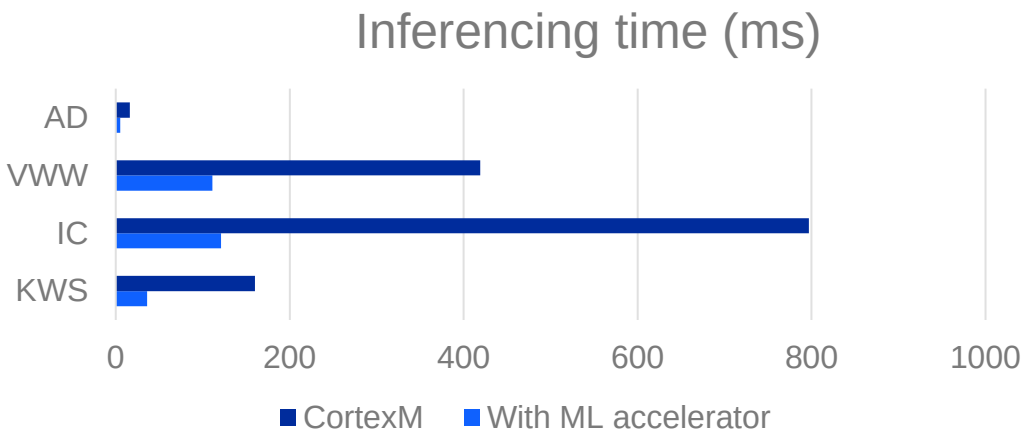
Matrix dims.		CMSIS f32 cpu-cycles	CMSIS f16 cpu-cycles	MVP cpu-cycles	instr	stalls
2x2	2x2	226	304	403	8	0
4x2	2x4	602	913	424	32	0
6x2	2x6	1210	1921	464	72	0
8x2	2x8	2050	3321	516	128	0
10x2	2x10	3122	5113	592	200	0
12x2	2x12	4426	7297	676	288	0
14x2	2x14	5962	9873	784	392	0
16x2	2x16	7730	12841	904	512	0
18x2	2x18	9730	16201	1036	648	0
20x2	2x20	11962	19953	1192	800	0
20x4	4x20	17962	27956	1593	1200	1
20x6	6x20	23742	39956	2193	1600	201
20x8	8x20	27562	47556	2793	2000	400
20x10	10x20	33162	59556	3393	2400	601
20x12	12x20	37162	67156	3993	2800	801
20x14	14x20	42762	79156	4593	3200	1000
20x16	16x20	46762	86756	5193	3600	1201
20x18	18x20	52362	98756	5793	4000	1401
20x20	20x20	56362	106356	6393	4400	1600

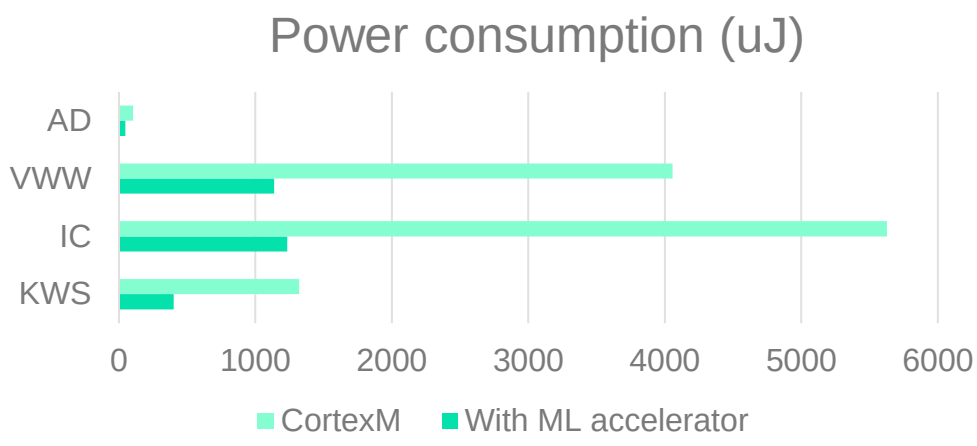
- ✓ **Faster and more efficient execution of many algorithms with large data for example filtering algorithms**
- ✓ **Saving CPU cycles, saving power, resulting longer battery life**
- ✓ **Option to win sockets against faster CPUs**

The Benefits…

- Dedicated **ML computing subsystem** next to the CPU: Matrix Vector Processor (MVP)
- Optimized MVP to accelerate ML inferencing with a lot of processing power **offloading the CPU**
- **Up to 8x faster** inferencing over Cortex-M (see below perf. benchmark)
- Up to **6x lower power** for inferencing (see below perf. benchmark)
- Dedicated OPNs for MVP accelerated parts → EFR32MG24B[2]… or [3]

Performance data with ML hardware accelerator vs. pure SW on CortexM*



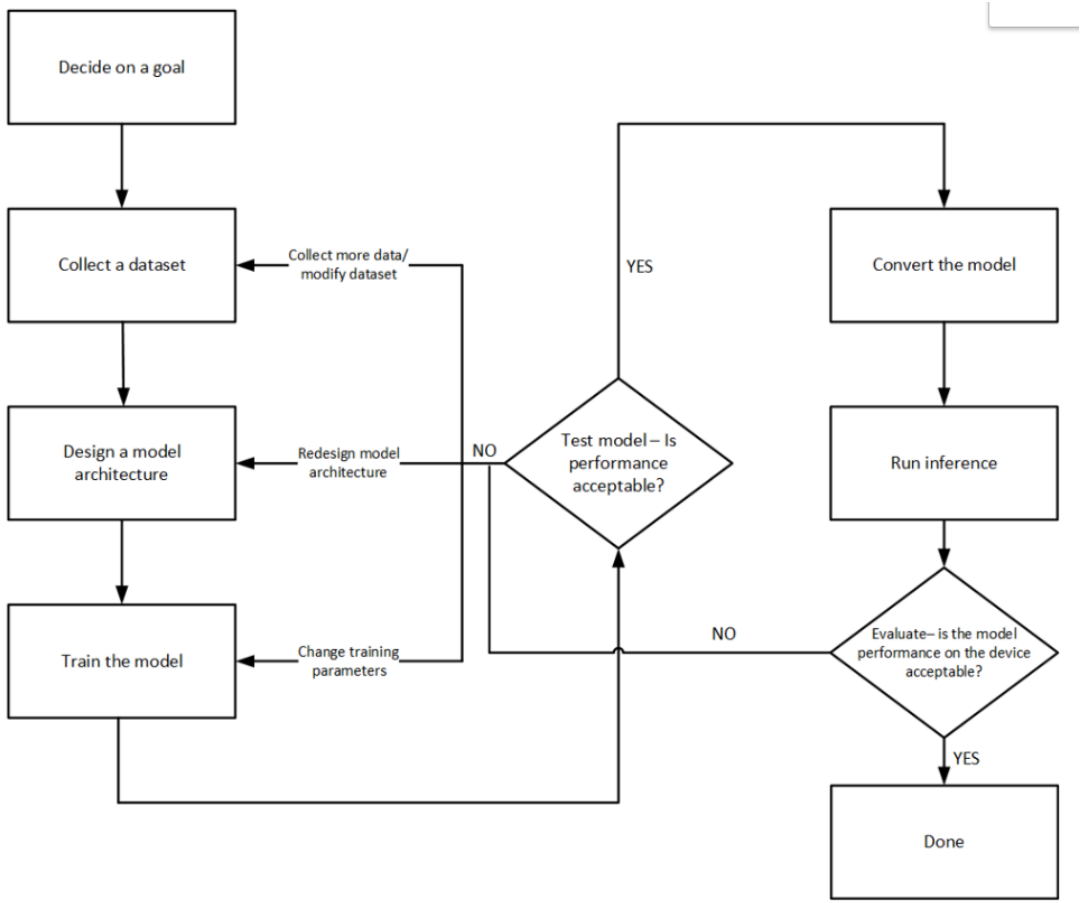


*Standardized performance benchmark validated by independent benchmarking body **MLCommons.org**. Published in MLPerf Tiny v1.0. Results are for inferencing only (not for the complete application). You can refer to MLCommons as validated results-

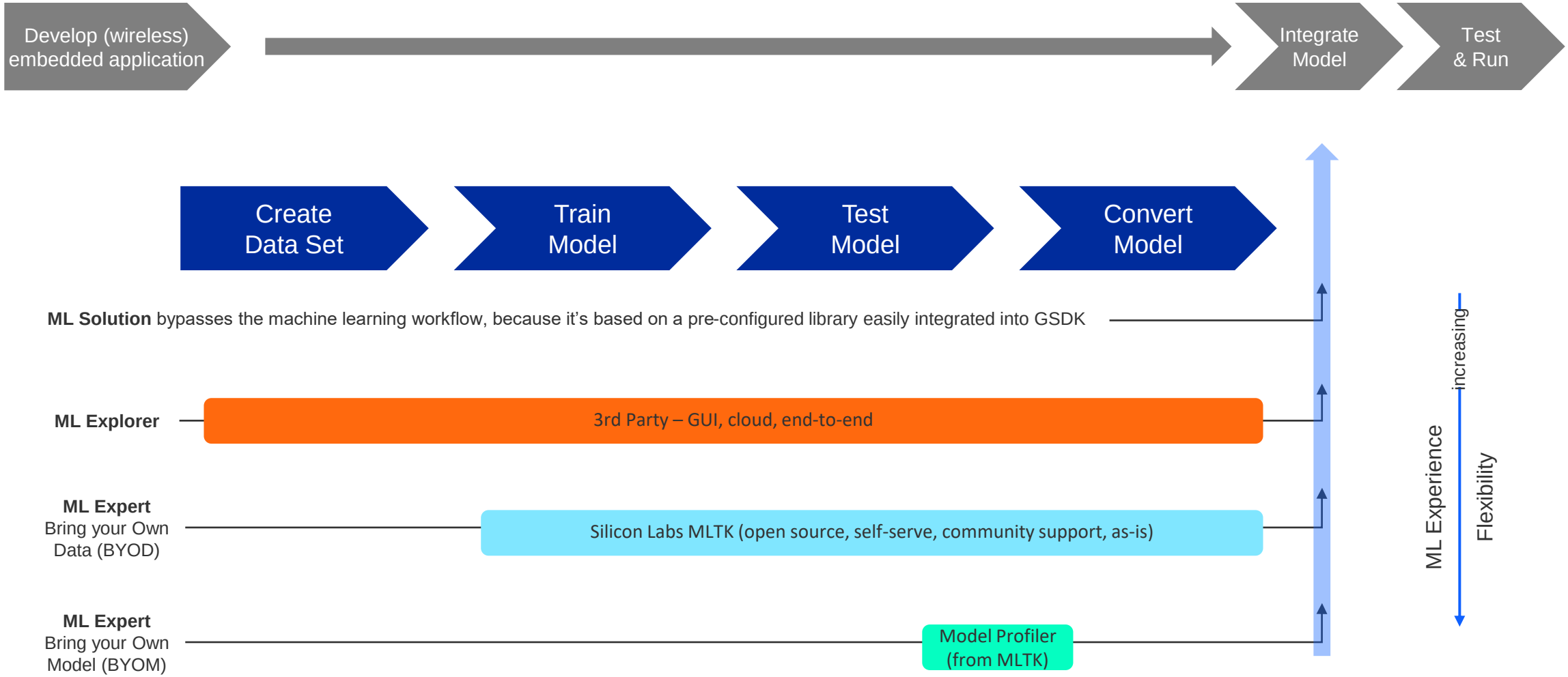
How Silicon Labs is enabling AI/ML through software?

Machine Learning Development Steps

- **Goal**
  - What are you trying to achieve?
- **Collect a dataset**
  - Construct a dataset that you will use to train the model, some will be kept aside for testing the model.
- **Design Model architecture**
  - It is not the raw data that is inputted into the model, it is the pre-processed data.
  - Therefore, we must choose a pre-processing block that is relevant for the type of data we are dealing with.
- **Train the Model**
  - About 80% of the dataset should be used at this stage.
  - the desired output is good predictions on generalized inputs.
  - Need to avoid underfitting and overfitting.
- **Test the Model**
  - check the performance of the model



Embedded Development with Machine Learning (supervised)



Software and Tool Support by customer skills: know your customer's skills

ML Expert

Python scripts and tutorials

SILICON LABS
Machine Learning Toolkit*

siliconlabs.github.io/mltk

TensorFlow

TFLite Flatbuffer

TFLite-micro Interpreter

CMSIS-NN Kernels | Silicon Labs HW-based Kernels

Cortex M | MVP (NPU)

ML Explorer

GUI Developer Tools

EDGE IMPULSE

edgeimpulse.com

SensiML™

sensiml.com

TFLite-micro Interpreter

CMSIS-NN Kernels | Silicon Labs HW-based Kernels

Cortex M | MVP (NPU)

ML Solutions

Solution Libraries

Wake Word / Voice Command — sensory

sensory.com

Anomaly Detection — Micro.ai

micro.ai

System Integrators

KLIKA·TECH GLOBAL IOT SOLUTIONS

AIZIP

Capgemini

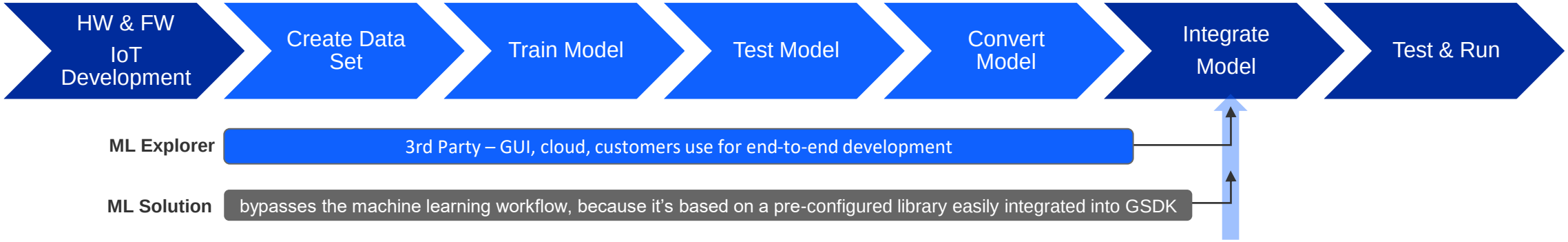
AITAD

Talent·Technology·Solutions Bellintegrator

Cortex M (& MVP)

*Machine Learning Toolkit is public but pre-alpha release

ML Partnership Overview



Full Development Services

	Edge Impulse	SensiML	MicroAI*	Sensory	AIZIP	NeutonAI	Klika-Tech	AITAD
Low-Rate Sensors	✓	✓	✓		✓	✓	✓	✓
Audio Pattern Matching	✓	✓			✓	✓	✓	✓
Voice Commands	✓	✓		✓	✓	✓	✓	✓
Low-Resolution Vision	✓				✓			✓
Regions	AMER, EMEA, APAC	AMER, EMEA, APAC	AMER, APAC	AMER, EMEA, APAC	AMER, EMEA, APAC	AMER, EMEA, APAC	AMER, EMEA, APAC	EMEA

*MicroAI has a platform for users to develop their own application, but it's solely for anomaly detection or system health score

ML Explorer

ML Solution

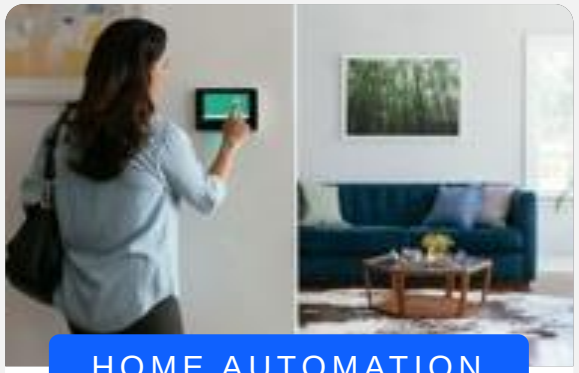
Full Dev Service

Example Usecases

Smart Home

SMART CONTROL

- Voice control of lights, blinds, appliances, HVAC, etc.
- Wake-word detection (e.g. ‘Alexa’) for always-on systems
- Command words for home control (~10 words set)

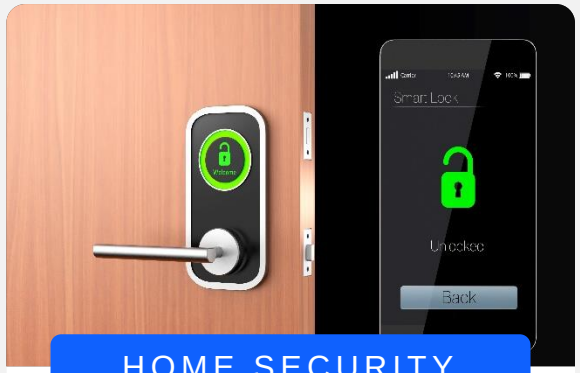


HOME AUTOMATION

- Occupancy detection (e.g. human vs. pet)
- People counting sensor
- Voice controlled sensors (e.g. smoke alarm ‘hush’)
- Air quality detector
- Pet detection (smart pet-door)

HOME SAFETY

- Baby-care and monitoring sensors with audio and vision
- Smoke detectors
- Gas sensor
- Air quality detector – anomaly detection
- Water leakage detector

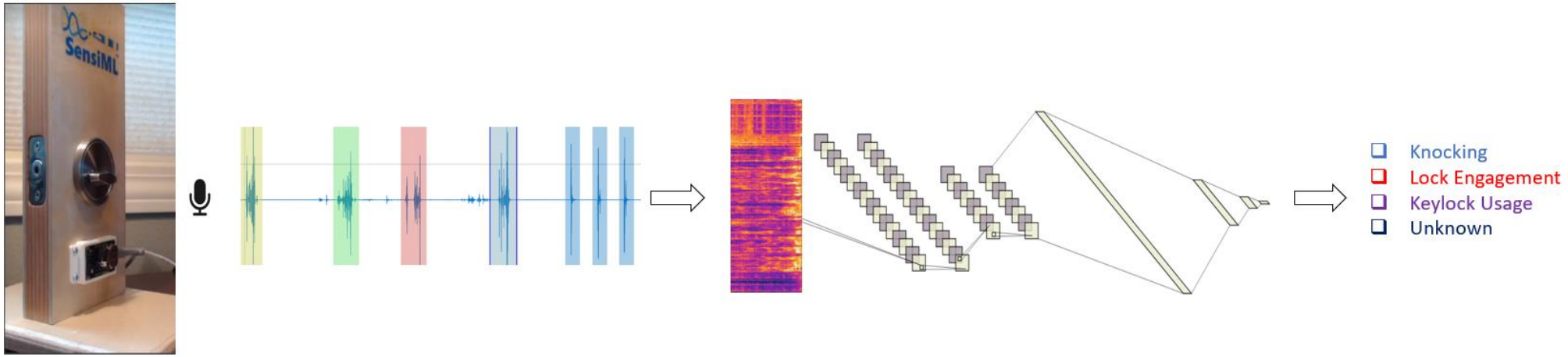


HOME SECURITY

- Smart lock – activity detection (audio, IMU)
- Smart lock – Fingerprint
- Glass break detector (IMU, Audio)
- Security sensors like scream, ’help’, shot detection
- Easy BT direction finding (inside/outside, up/down) – smart lock, garage door

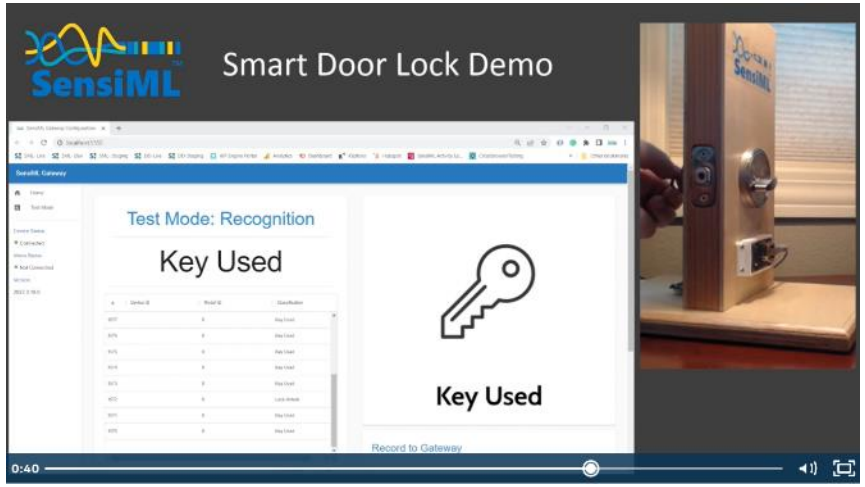
Enabled voice control, New sensors with audio pattern matching, Better detection accuracy, **avoid false alarms**

Usecase 1: Smart Door Lock

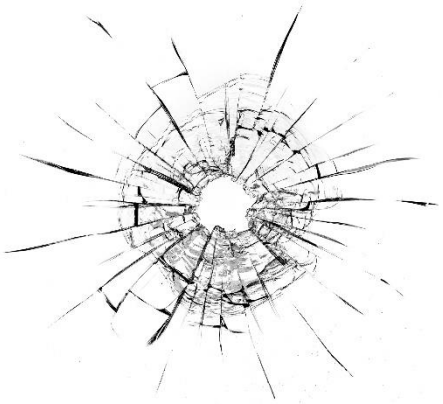


SensiML's smart lock demo and tutorial:

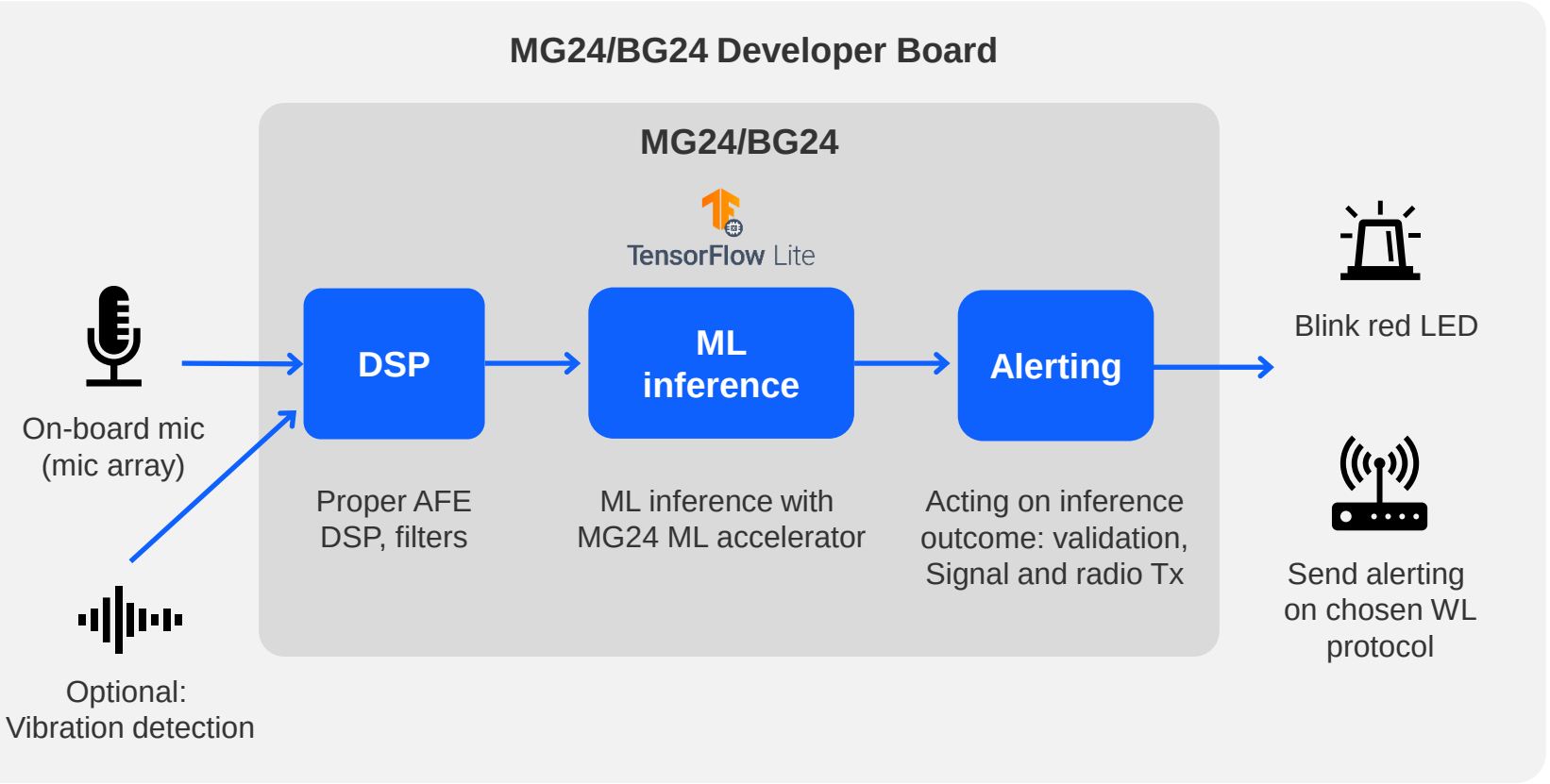
https://sensiml.com/blog/creating-an-acoustic-smarthome-sensor-with-silabs-xg24-platform/



Usecase 2: Glass Break Sensing



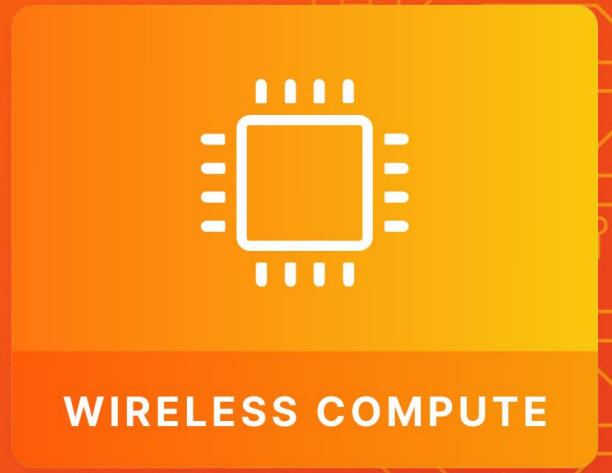
Loudspeaker with different glass break-in audio samples



- Battery operated Edge device
- MG24 runs complete application including ML
- Audio sensing with DSP
- Optional vibration sensing

- HW Accelerated ML processing while CPU offloaded
- Wireless connectivity using BLE, ZigBee or Thread
- ML sensing method outperforms standard sensing especially in avoiding false positive detections

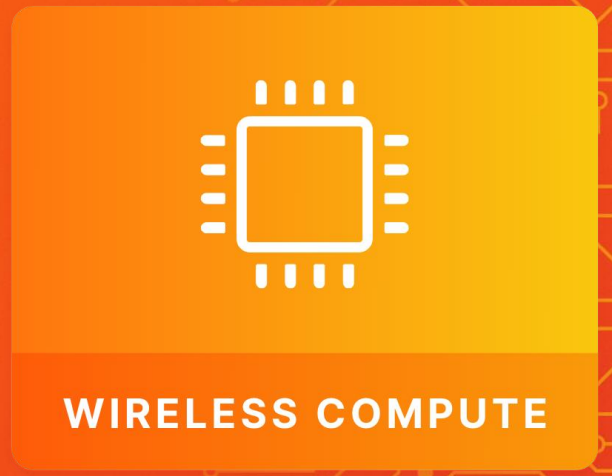
Q&A



Thank You

Watch ON DEMAND

tech talks



2024 tech talks BY SILICON LABS

APAC Tech Talks: Wireless Technology Training

Register Now