AIML-103

Machine Learning Techniques for Predictive Maintenance

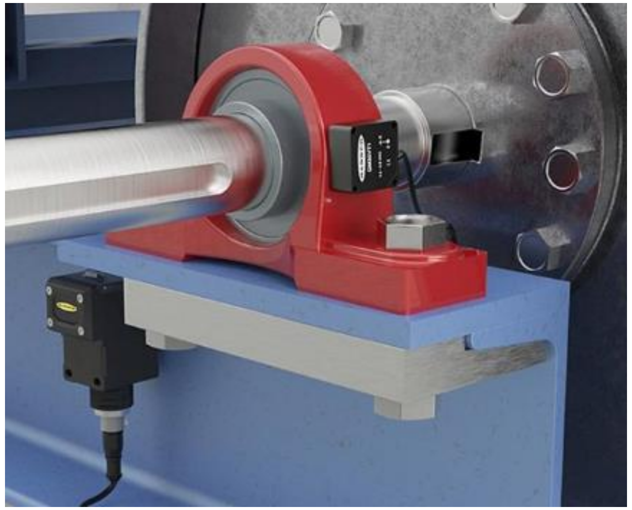
Tristan Cool | August 2023

Agenda

Predictive Maintenance

vs Condition Monitoring

Predictive Maintenance — Applications



Industries

- Chemical and Petroleum refinement
- Water filtration
- Pulp & Paper processing
- Power Plants
- Manufacturing & Warehousing
- Rail, Shipping & Logistics
- Construction & Farming vehicles
- HVAC & Refrigeration

Applications

- Motor & electrical drives
- Factory machinery/tool vibration
- Valves and pressure sensors and pumps
- Noise detection from bearings
- Heat measurement of lubricant/fluids

Preventative VS **Predictive** VS Reactive Maintenance

PREVENTATIVE

Unspecific and **unintelligent**

No information or insight gathered

Time-consuming

Causes **routine down-time**

PREDICTIVE

Intelligent and **insightful** for all machines

Automated and efficient

Adaptable and scalable

No down-time

Maximized R.O.I

REACTIVE

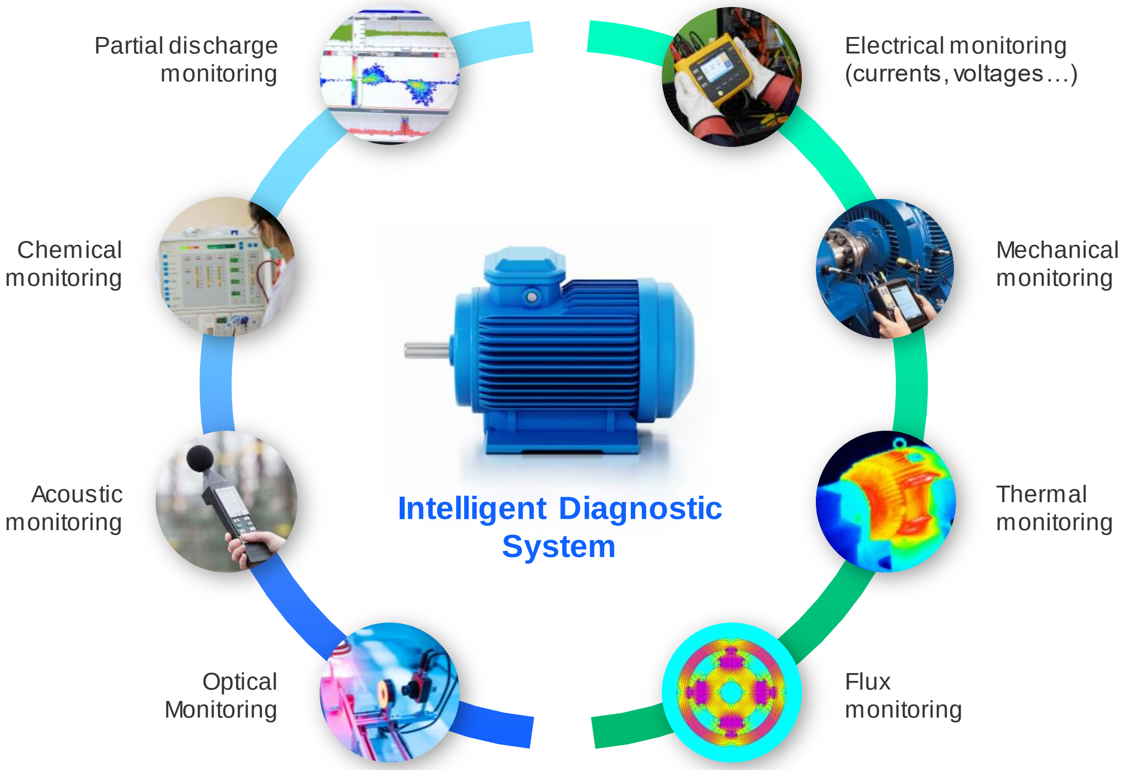
Machine **failure specific**

Occurs **after failure**

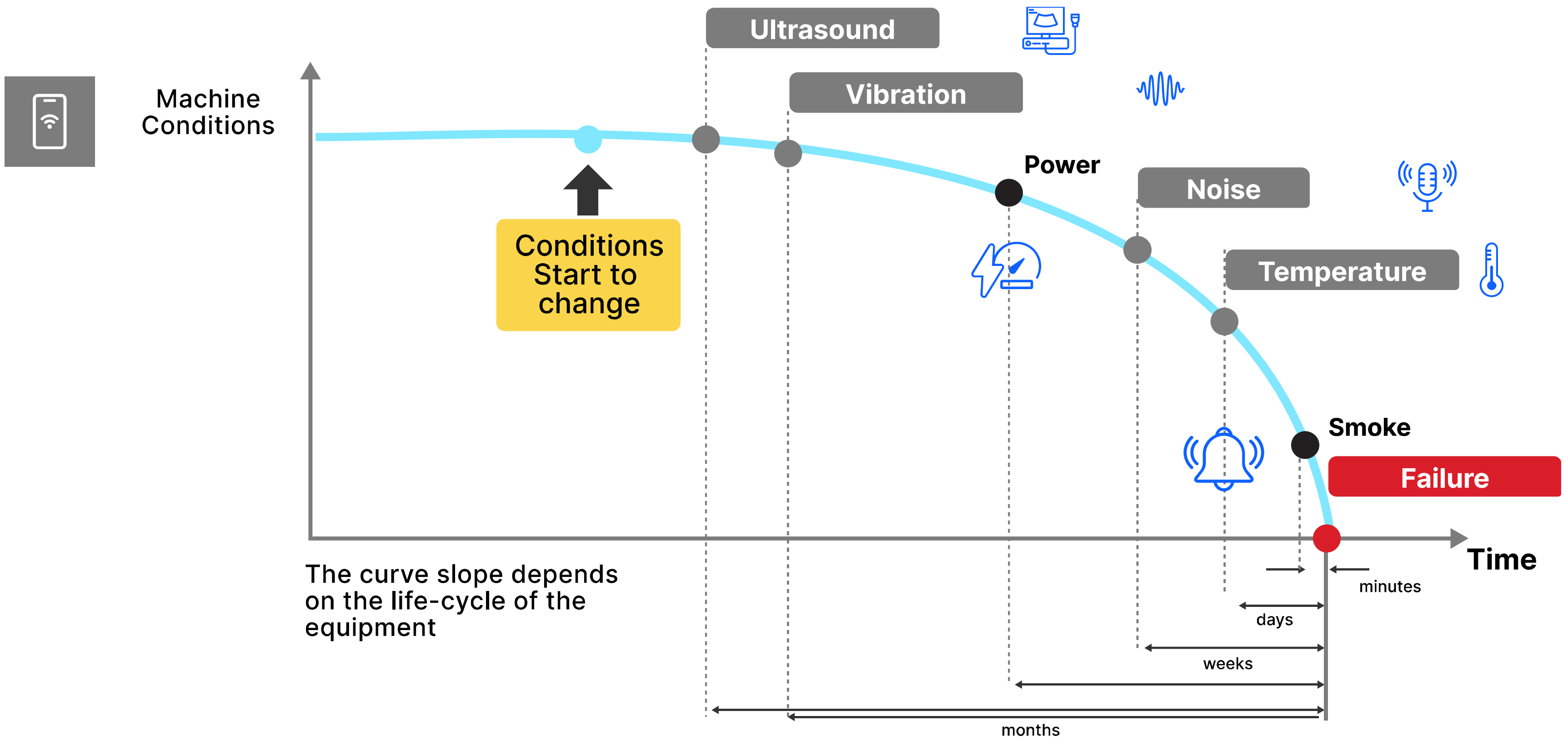
Requires **expensive human expertise** and intervention

Causes **extended down-time**

Predictive Maintenance – Methodology

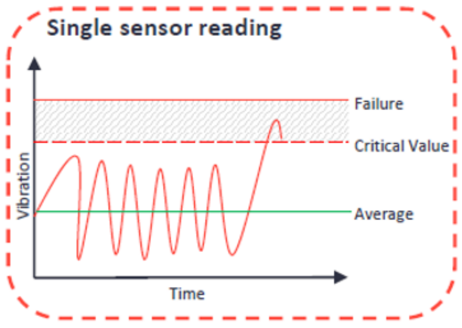


Predictive Maintenance – Life Cycle



A machine's maintenance profile can truly be *predicted*

Predictive Maintenance VS Condition Monitoring

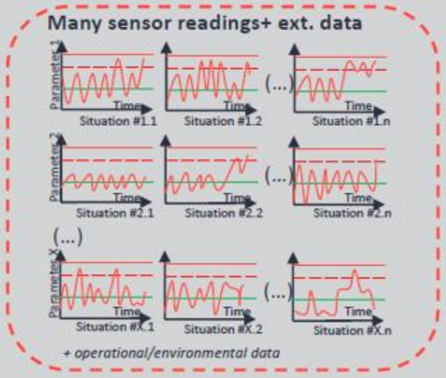


THRESHOLD ANALYSIS

Critical values defined based on machine specifications.

Each sensor is **monitored individually**.

Warnings occur only if critical **values are surpassed**.

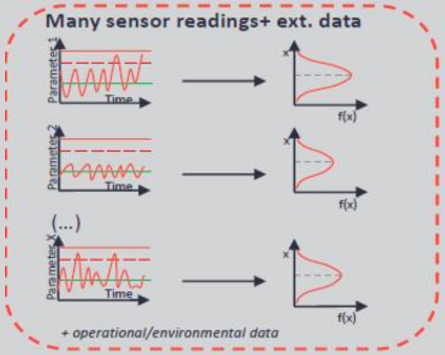


STATIC ANALYSIS

Combining **multiple timelines** and all measurable parameters.

Static rules for indirect failure prediction.

Health/risk score for equipment status.



DYNAMIC ANALYSIS

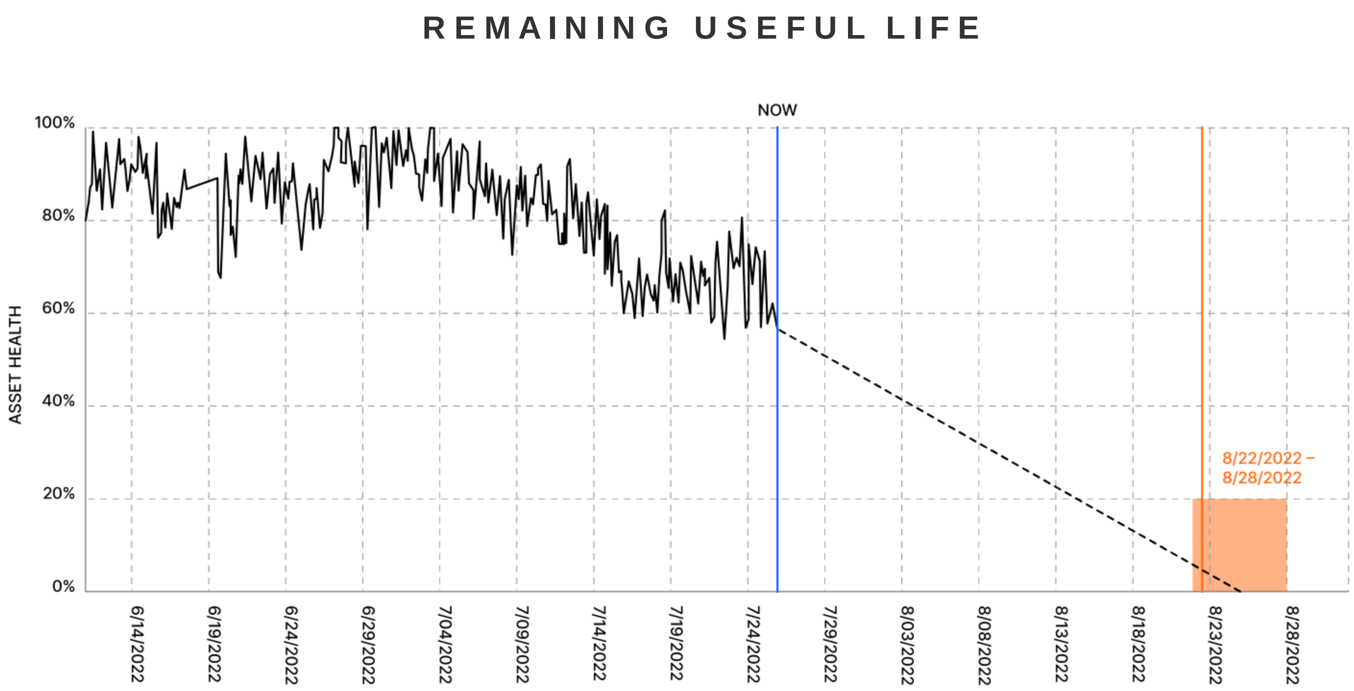
Dynamic models enable the **prediction of failure** likelihood.

Algorithms are **model-trained** based on **anomaly detection**.

Advanced processing required.

Source: IoT Analytics

Predictive Maintenance Insights – R.U.L



- Anomalies and any major or minor incidents are recorded over time.
- Performance is plotted against a timeline to determine an overall asset health score
- Data is extrapolated following this algorithm to determine a trendline to an exact date (or number of days) when an asset is expected to fail

IoT & AI/ML Solutions

Silicon Labs

IoT Benefits for Predictive Maintenance

- **Industrial-IoT Coverage**
  - Long range signal strength for industrial environments and interference
- **Industrial-IoT Reliability**
  - Large networks with low latency
  - High payloads for Cloud or Edge computations
- **Integration ROI**
  - AI/ML insights by modeling and training time series data
  - Scalable networks (multiple sensor types and gateways)
  - IoT for more than just a condition-monitoring data-pipe
  - Integration with technician’s tools and legacy systems
- **OEM & Retro-fitted use-cases**
  - High processing requirements
  - Low power requirements

IoT Technologies for Predictive Maintenance

BG22 | MG24 - MG27 | FG25 - FG28 | SIW917

Silicon Labs solutions cover a variety of IoT protocols suited for different range, power and topography.

- **Wi-Fi 6 + BLE combo**
  - Long range and dense networks
  - AP connection or Cloud backhaul gateways
- **Sub-GHz**
  - For end-nodes in crowded 2.4G environments
  - For end-nodes with metal interference
  - with BLE and Wi-SUN
- **Proprietary, 15.4, BLE Mesh and Wirepas Mesh**
  - For very large networks with multiple hops and low latency

AI/ML on Silicon Labs' Wireless SoCs

EFR32 Series 2 and Wi-Fi SoCs

Higher Performance Platform
- ARM Cortex M33 (78 MHz)
- Improved radio performance
- Lower power (MCU active, TX/RX)

Improved Security
- Secure Vault - Mid
- Secure Vault - High (select OPNs)

Acceleration - MVP
- AI/ML acceleration
- Faster AoA/AoD calculation
- Math library (matrix and vector ops)

AI Software
- TensorFlow Lite for Microcontrollers with accelerated kernels in GSDK
- 3rd Party end-to-end tools

All Series 2 SoCs support ML

78MHz CortexM33
AI/ML accelerator
1.5MB / 256kB
2.4 GHz radio
20 dBm TX Power
Secure Vault
Low power

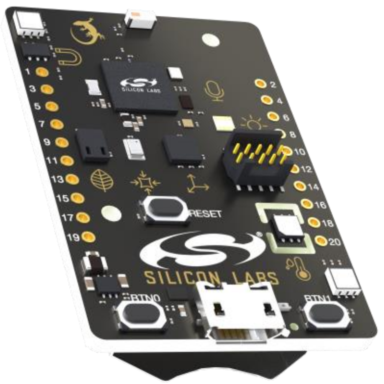
WiFi 6

180MHz CortexM4
160 MHz NWP
AI/ML accelerator
Up to 8MB / 672kB
2.4 GHz radio
21 dBm TX Power
PSA L2 Security
Low power

xG24-DK2601B Developer kit

Broad Range of Sensors
- 9-axis Inertial Sensor
- 2 Digital Microphones
- PIR sensor
- Pressure Sensor
- Relative Humidity and Temperature Sensor
- UV and Ambient Light Sensor
- Hall-effect Sensor

Ready to demonstrate ML
- Sample applications in GSDK
- Examples on GitHub
- Examples and tutorials in MLTK
- Many sample applications and demos from partners
- Plug&Play Sensor extensions with Sparkfun Qwiic



sparkfun ELECTRONICS

TensorFlow Lite

Common Machine Learning software and tools on our Wireless SoC portfolio
Use cases are dependent on RAM and wireless stack

xG24: 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
- Dedicated **Math library** to accelerate matrix and vector linear algebra 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

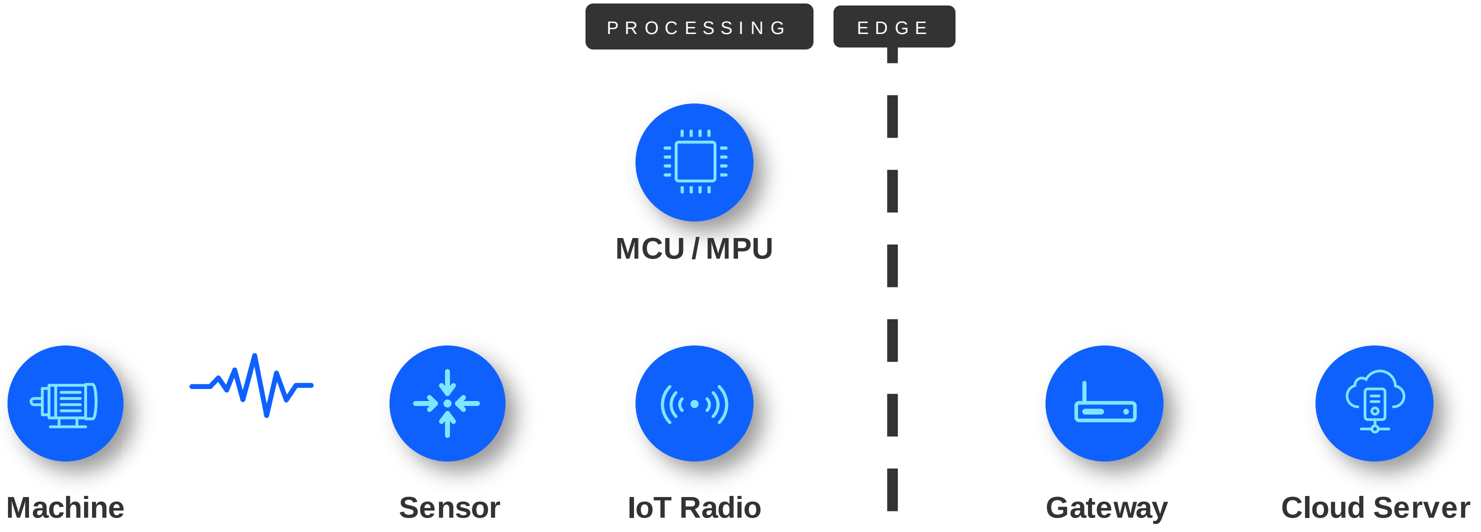
Dedicated Security Core

- 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) -125°C
- 6x6 QFN48 (28/32 GPIO) -125°C
- 7x7 SiP Module (+10 dBm)
- 12.9x15.0 PCB Module (+10 dBm)

Predictive Maintenance Processing on the Edge



Leverage more processing at the Edge

Offload main processor

Accelerate time-series vector computations (4x, 8x)

Why AI/ML at the Edge?

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 and Power 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
- Process vs. transmit tradeoff in power cons.

Offline Mode Operation

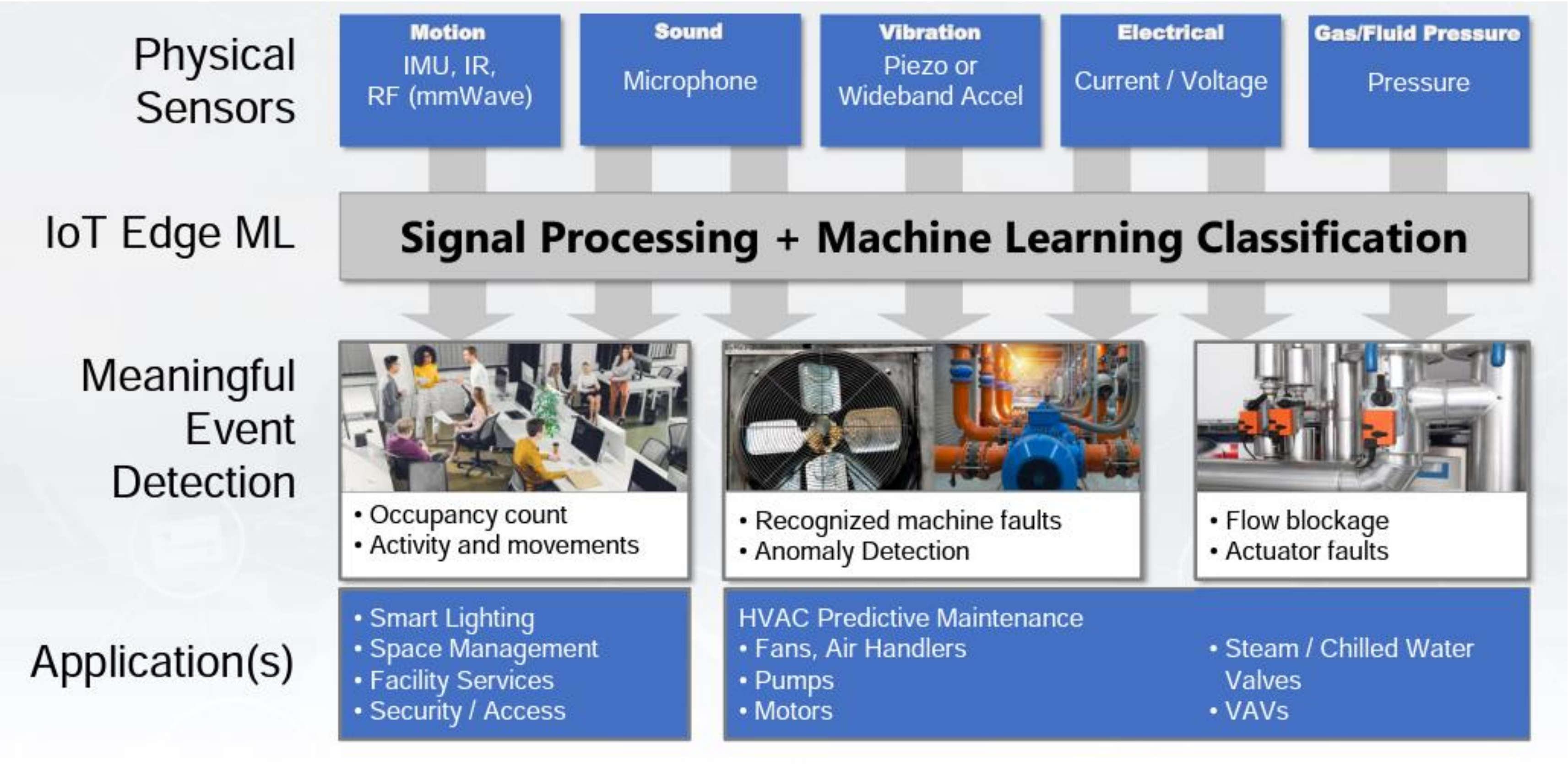
- Local system keeps operating standalone in case of any network issue
- Connectivity is occasional or blocked by admin

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

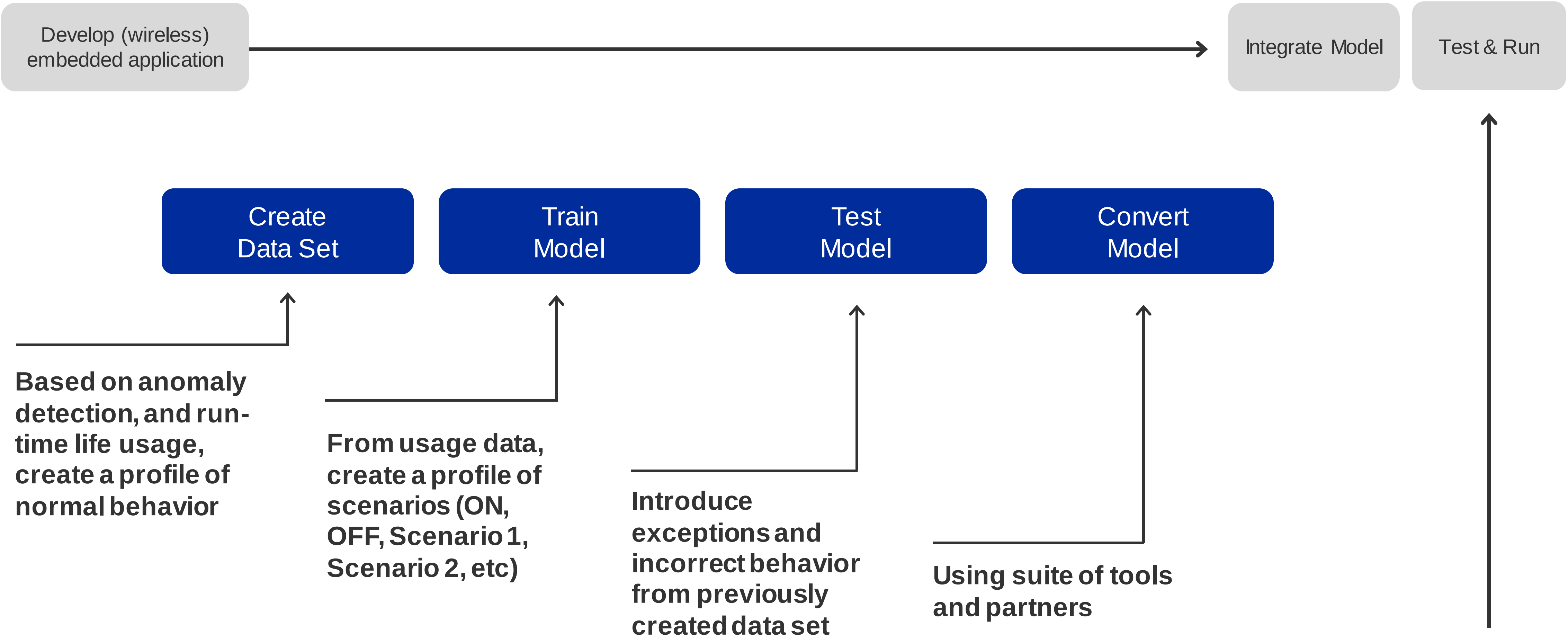
Data processing is more efficient with AI/ML at the Tiny Edge – various new use cases enabled

Use Cases for AI/ML at the Edge in Predictive Maintenance



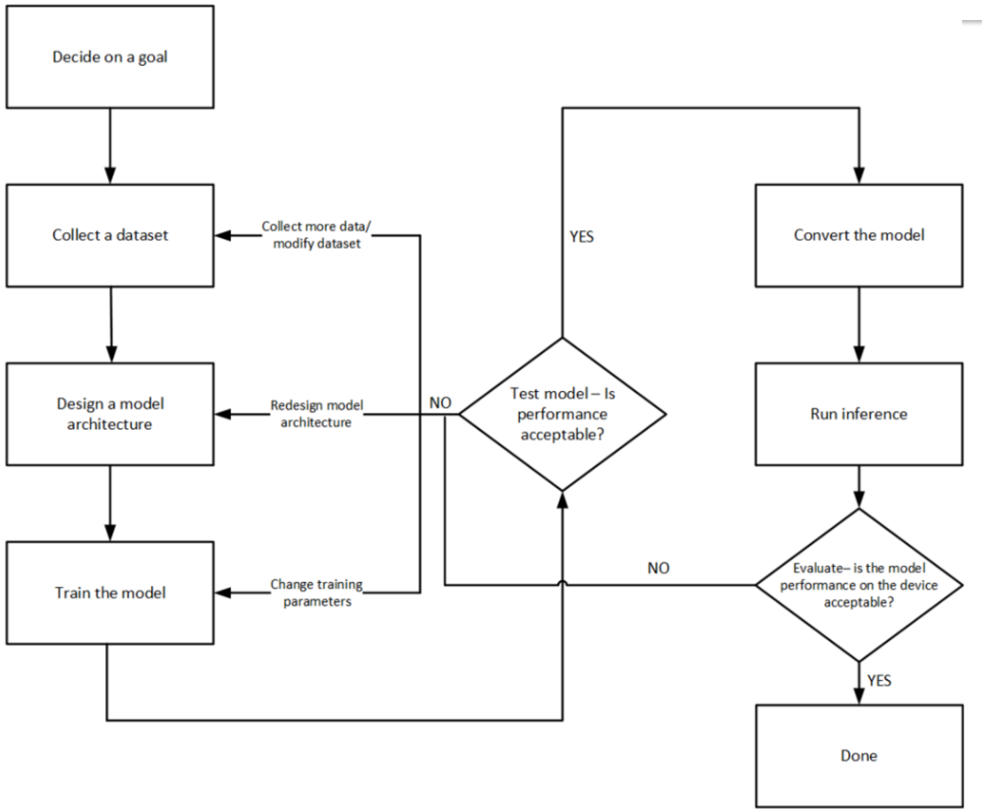
Source: SensiML – WorksWith 2021

Machine Learning Development – Model Training



Machine Learning Development – Model Training

- **Goal**
  - What are you trying to achieve?
- **Collect a dataset**
  - Construct a dataset that you will use to train the model (anomalies)
  - 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
  - Iterate and refine



Silabs Tech Talk

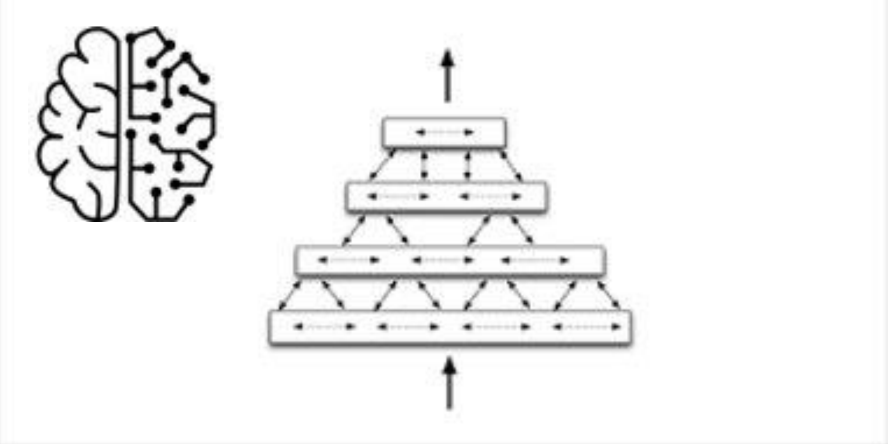
Predictive Maintenance Model Training

Watch Now

Machine Learning Development – Hierarchical Temporal Memory (HTM)

What is Hierarchical Temporal Memory?

- HTM simulates the structure and **biological functionality of the neocortex (memory-based)** and is particularly suitable for **sequence learning and prediction**
- HTM manipulates sensory data represented as **simple ideas in the lower level** and the idea **gets more abstract in the higher level**



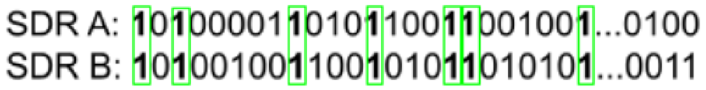
When to use Hierarchical Temporal Memory?

- When using **multiple input sources** that are vastly different but contribute to a singular output.
- When data is **input at high-speeds**, but is **not temporal** (not time specific, can be re-ordered)
- When manipulating **unlabeled** and **small-batch data**
- When the model needs to **learn continuously unsupervised**
- When the model needs to **make predictions** based on previous learnings.
- Does not require accelerated computations and can adapt to **high levels of noise** and **sub-sampling**

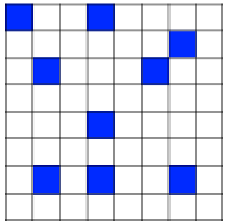
Machine Learning Development – Hierarchical Temporal Memory (HTM)

How does it work?

- Inputs from various input sources are **semantically encoded** as a sparse array (of 0s and 1's) called a **sparse distributed representation (SDR)**

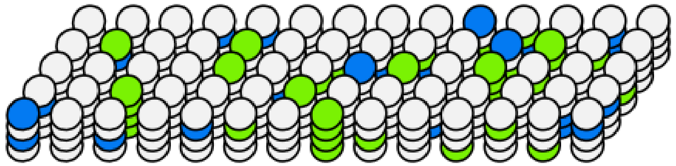


- This encoded array goes through a processing called **spatial pooling**



- Spatial pooling (SP) is the process of converting the **encoded SDR into a sparse array** to normalize/standardize the input data from various sources into a **sparse output vector** or **mini-columns** of definitive size and **fixed sparsity**

- The **temporal memory algorithm consists of two phases**.
  - 1st phase is to **evaluate the SP output against predictions** and choose a set of active cells.
  - 2nd phase is to **form a prediction** by putting cells into a predictive state

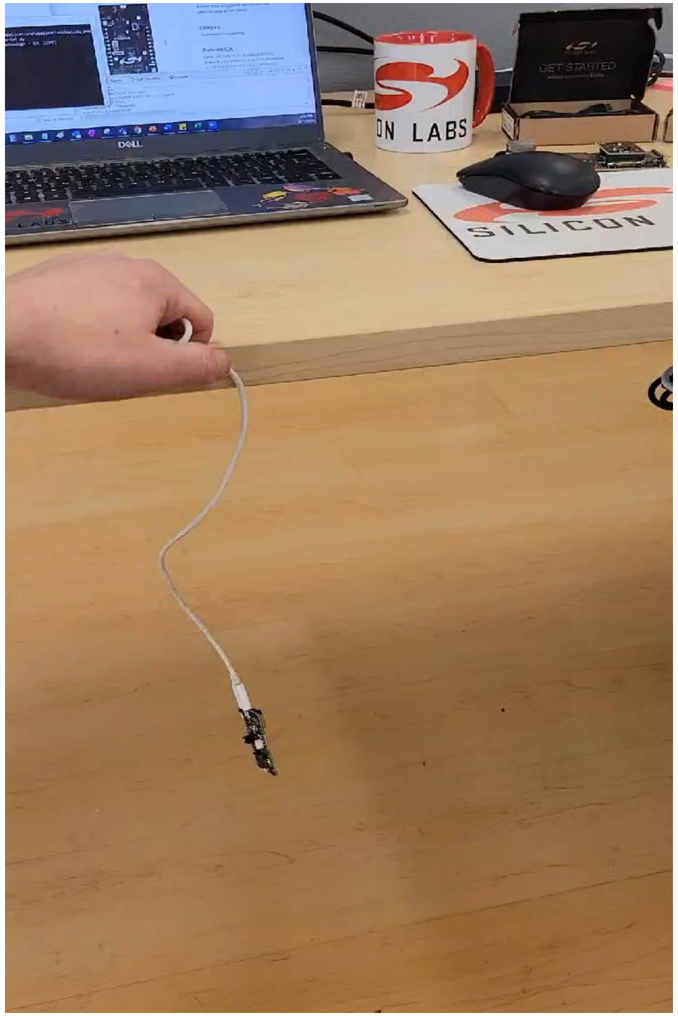


- The system's ability to **learn and remember** can be set by adjusting the **permanence** value which controls how likely a cell's state is changed.

AI & ML - Anomaly Detection

DEMO – Predictive Maintenance Techniques

Anomaly Detection – HTM DEMO



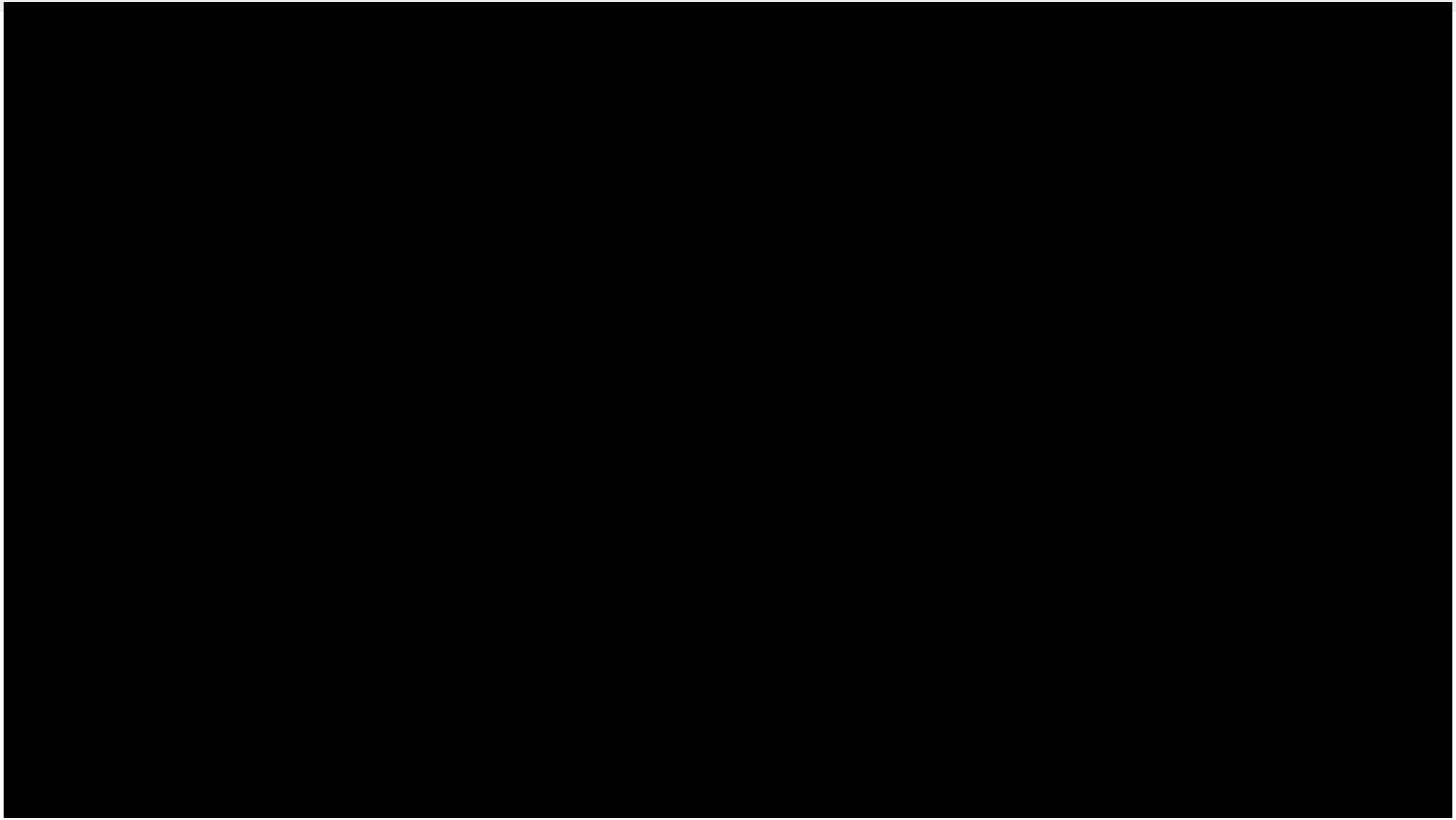
HARDWARE:
[EFR32xG24 Dev Kit – DK2601B]

SOFTWARE EXAMPLE:
[github.com/SiliconLabs/machine_learning_applications]

PROCEDURE:

1. Import project and dependencies
2. Build and flash to device
3. Open serial display python script
4. DEMO:
   1. Move the board in a *non-random* way - pendulum off a table
   2. Introduce an anomaly – brute vibration to table
   3. Graph anomaly score over time - average of past 10 scores, between 0 and 1 (0: no anomaly ; 1: high anomaly) using Hierachical Temporal Memory (HTM) encoding

Anomaly Detection – HTM DEMO



Anomaly Detection – BONUS DEMO – Electric Fan

Electric hand-held fan – 3 anomalies introduced purposely

Additional Resources

Q&A

Machine Learning Development – Software and Tool Support

ML EXPERT	ML EXPLORER	ML SOLUTIONS

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

Anomaly Detection — Micro.ai
micro.ai

- 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

System Integrators

KLIKA·TECH GLOBAL IOT SOLUTIONS

AIZIP

AITAD

Bellintegrator

Cortex M (& MVP)

*Machine Learning Toolkit is public but pre-alpha release

Additional Resources

- **WorksWith 2023**
  - AIML-101 : *Ensuring First Time Success of ML Applications*
  - AIML-102 : *Machine Health and Condition Monitoring using Edge Impulse*
  - IOT-103 : *IoT Trends: Embedded ML in Edge Devices*

- **DEMOs:**
  - Silabs Tech Talk – Pred.Maint. Model Training
  - WorksWith23 – Pred. Maint Anomaly Detection github
  - SensiML – Pump & Fan Anomaly Detection
  - Run-Time and Lifetime examples: *coming in 23Q4*

- **Silicon Labs resources:**
  - Predictive Maintenance - landing page
  - Machine Learning – landing page and docs
  - Machine Learning – User Guide
  - Machine Learning – ML Tool Kit github

- **Partner resources:**
  - Edge Impulse
  - SensiML
  - MicroAI
  - Capgemini

CONCLUSION

- Predictive Maintenance is an insightful expansion to conventional Condition Monitoring.
- The IoT and advanced Edge computing are useful tools in creating and scaling a useful predictive maintenance network.
- Consult Silicon Labs and our valued partners for advice on Machine Learning methodologies for your Predictive Maintenance use-case.