

Think inside the box

Smart and sustainable products
with edge machine learning.



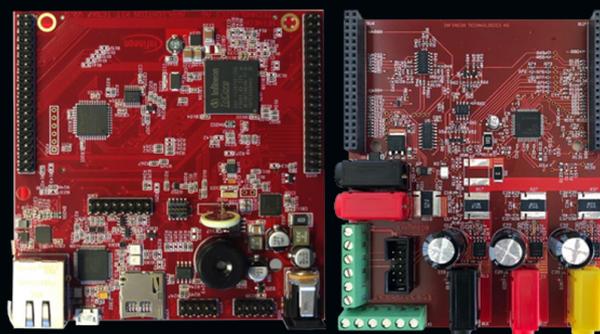
Ekkono x Infineon

Every installation is unique, and conditions change over time. Personalization, individualization and adaptation are at the core today, and it all starts with data – data generated by your products. With the right tools, that data can turn into insights that are leveraged for new service offerings: smart maintenance, context-aware adaptive control, automated configuration, over-the-air upgrades, and other big IoT opportunities that are there for the taking.

Do these challenges and strategic initiatives apply to you or your products? The solution spells Edge Machine Learning. By running machine learning onboard your product, based on real-time and granular sensor data, it can self-learn and predict normal behavior. This presents a world of opportunities for custom maintenance and performance optimization. Now you can shift your service offering from reactive to proactive by constantly learning more while in operations, based on where, what and how a specific unit is used.

The word 'Ekkono' means cognition, and that is what we do. Our software expedites the implementation of smart and predictive features on your products. The uniqueness is that we do machine learning – not just inference, but on-device incremental model training on streaming data. This means that your product can train a custom model that incorporates local conditions, and that conti-

nues to learn more and more over time. You can detect deviations, predict remaining useful life of a wear part, identify the need for maintenance, simulate optimal settings, or replace a physical sensor with a virtual twin. All of this can be accomplished while simultaneously reducing dependency on connectivity and improving integrity by not sending sensitive raw data to the cloud.



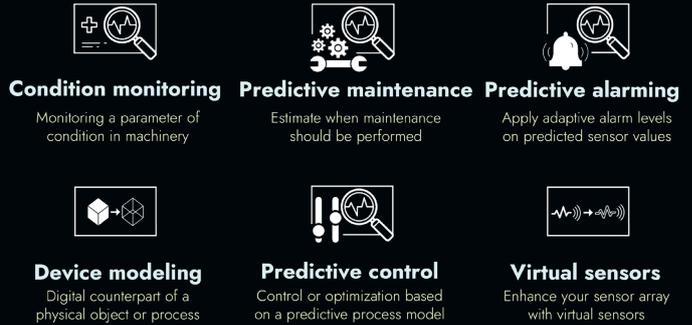
The best part is that you can do this even on your small and constrained devices. The tiny footprint of Ekkono's software enables even sensor-size of units to become smart. In collaboration with Infineon® we demonstrate how our software can run on an AURIX™ microcontroller to empower even a car component like a transmission or steering system, or an industrial equipment like a pump or an electric motor, to enjoy these privileges.

For more information, visit www.ekkono.ai and www.infineon.com.



Use Cases

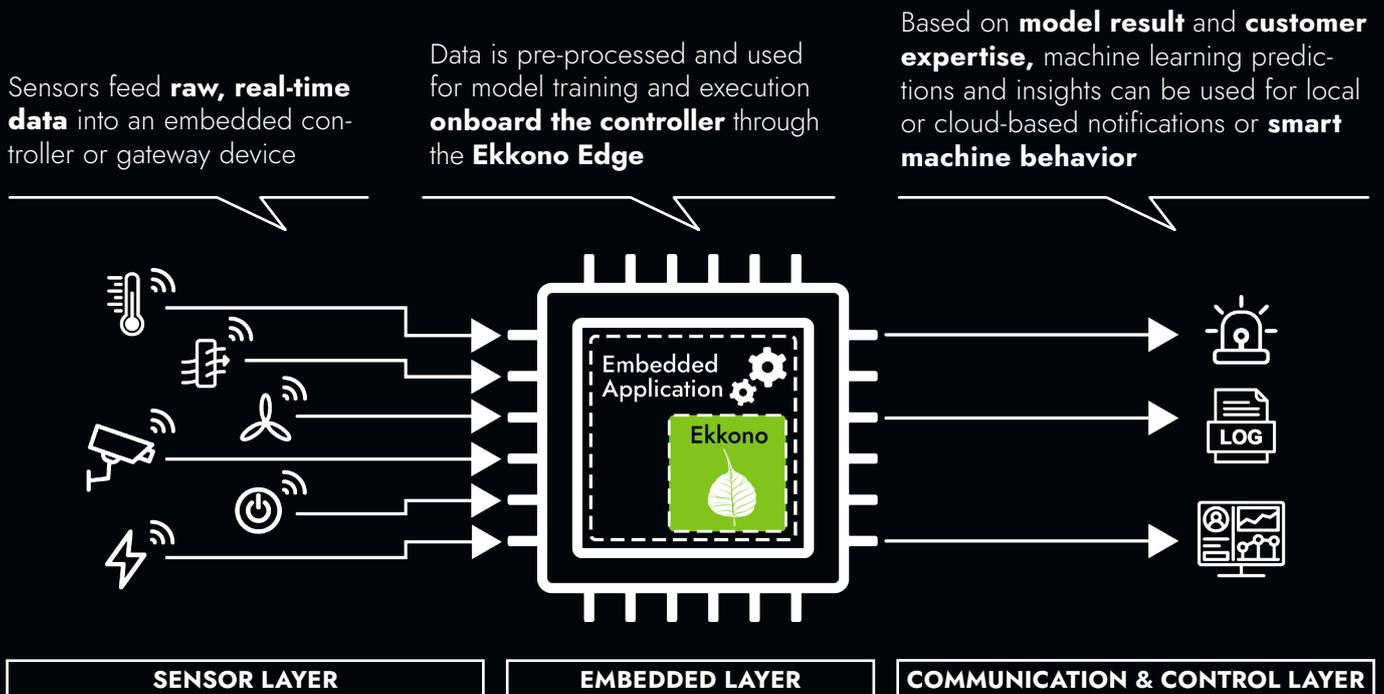
In Ekkono's definition, edge means sensor-close, where one can see real-time sensor data. Most products have a number of sensors and actuators but a constrained capacity and memory. Ekkono's unique ability to train custom models per unit, enables automated individualization of insights and features that range from condition monitoring to predictive control and virtual sensors.



Software Development Kit – How It Works

Ekkono provides a comprehensive software solution for embedded edge machine learning. The main functionality focuses on streaming analytics based on sensor data in combination with on-device learning. Primer is the core library and contains all functionality needed for developing an edge machine learning solution i.e. data wrangling, preprocessing (including signal processing), incremental and batch training of models, AutoML and evaluation techniques. Primer is designed to mainly be used on a laptop or workstation during the design phase of a project and is available for Windows, Mac and Linux.

When deploying a model to an edge device, one of Ekkono's two runtime libraries Edge and Crystal is integrated into an existing embedded application. Crystal is written in pure C and is optimized for the smallest devices without an operating system, e.g. Cortex-M. Edge is written in C++ and designed for slightly more powerful platforms and has more extensive functionality. Thanks to that, Ekkono's models have an integrated preprocessing pipeline, running a model on a device using one of the runtimes is extremely simple and only requires 5 to 15 lines of code, regardless of model or the amount of preprocessing.



Functionality

Functionality

Data Wrangling

- Removal of outliers and missing values
- Filtering of columns and rows
- Feature elimination

Preprocessing pipeline for streaming data

- Lags, moving averages, expressions
- Supports incremental training

Incremental and Batch Modeling

- Neural Networks, Ensembles, Random Forest, Decision Trees, Linear Regression
- Anomaly and Concept Drift Detection
- Conformal Prediction

Evaluation

- X-fold cross validation
- Continuous Evaluation
- What If simulation
- Sensitivity Analysis

Signal Processing

- Signal to noise ratio
- Autocorrelation
- Discrete Fourier Transform
- Discrete Wavelet Transform

Documentation

- API – documentation
- Notebook Tutorials
- Reference Implementations

Libraries

Libraries

The SDK is structured into four libraries.

Primer – Data Wrangling, model configuration, optional pre-training and model evaluation

- Encompasses all functionality including Edge, Crystal and Spectral

Edge – C++ runtime for deployment of models on the edge

- Preprocessing pipeline for streaming data
- Inference and Incremental training of all models
- Anomaly and Concept Drift Detection
- Conformal Prediction
- Continuous Evaluation
- What If simulation
- Sensitivity Analysis

Crystal – C runtime for deployment of models on the edge

- Preprocessing pipeline for streaming data
- Inference and Incremental training of Neural Networks and Linear Regression
- Continuous Evaluation
- Concept Drift Detection

Spectral – Signal Processing with incremental capabilities for feature engineering

- Signal to noise ratio
- Autocorrelation
- Discrete Fourier Transform
- Discrete Wavelet Transform

Runtimes and preprocessing libraries are delivered as source code.

Ekkono Studio

A cloud hosted environment designed to reduce time from start to deployment.

Functionality

- Jupyter hub based environment hosted in the cloud
- Utilization of Primer through Python and Jupyter notebooks
- Code snippets for faster coding, deployed through drag and drop. Support most functionality in Primer
- Interactive notebook tutorials
- Educational videos
- Optional local deployment using Docker

