A Breakthrough Innovation in MEMS Sensors

Introducing LSM6DSOX, iNEMO 6DoF Inertial Measurement Unit (IMU), with Machine Learning Core

Edoardo Gallizio

Sr. Product Marketing Manager

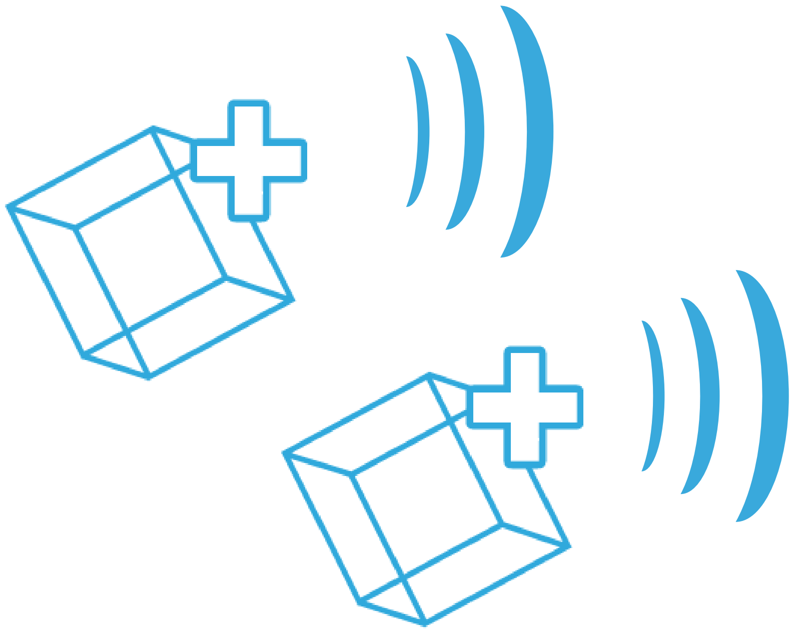
life.augmented

ST Developers Conference
September 12th, 2019
Santa Clara Convention Center - Mission City Ballroom
Santa Clara, CA

Agenda

- Artificial Intelligence in IoT
- ST Smart Sensor: 6-axis IMU with Machine Learning Core
- Why new sensors solutions are attractive?
- Solutions and Comparisons
- Tools
- Conclusions

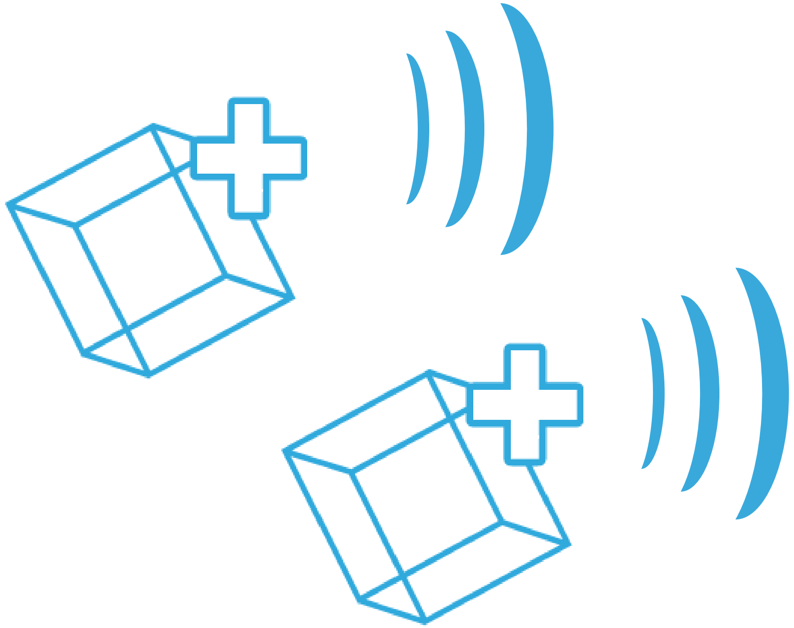
Artificial Intelligence in IoT



Logic Programming
Machine Learning
Decision Tree
A.I.
Search Algorithms
Deep Learning
Data Mining
Neural Networks

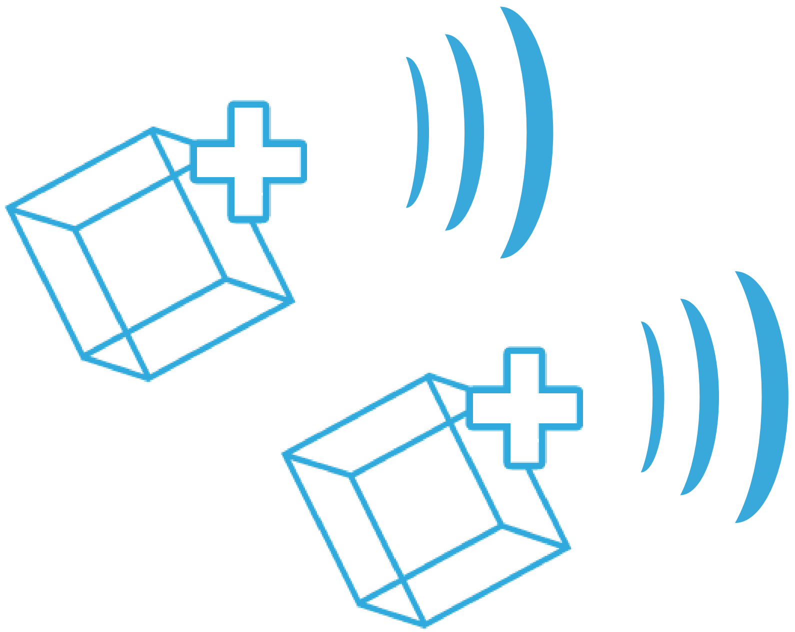
life.augmented

Artificial Intelligence in IoT



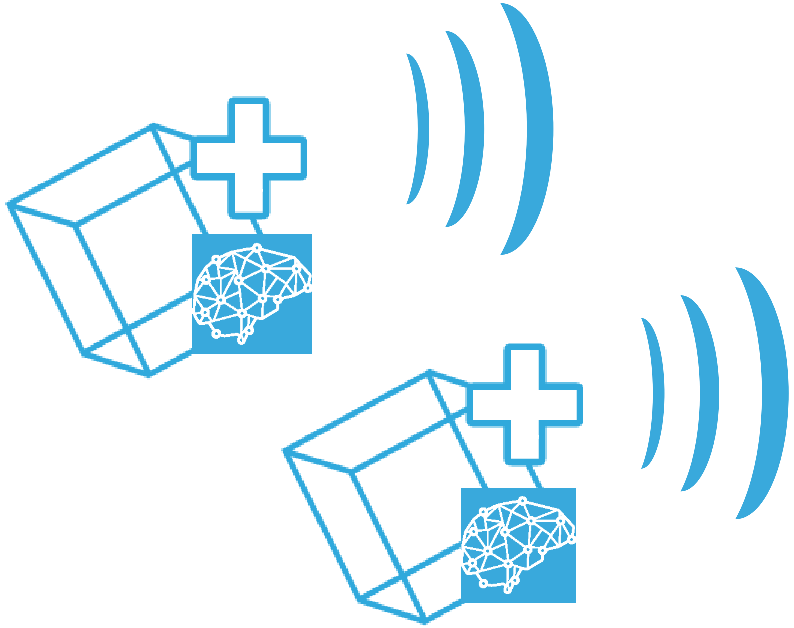
Analysis of vast
amount of information

Artificial Intelligence in IoT



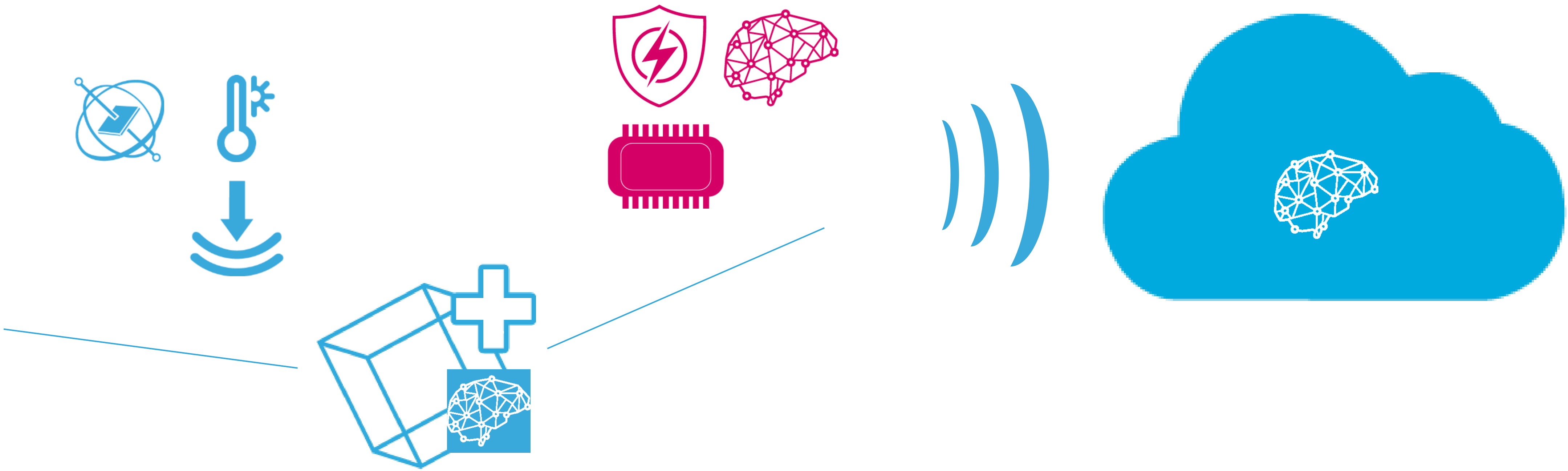
Centralized A.I. Architecture

Artificial Intelligence in IoT



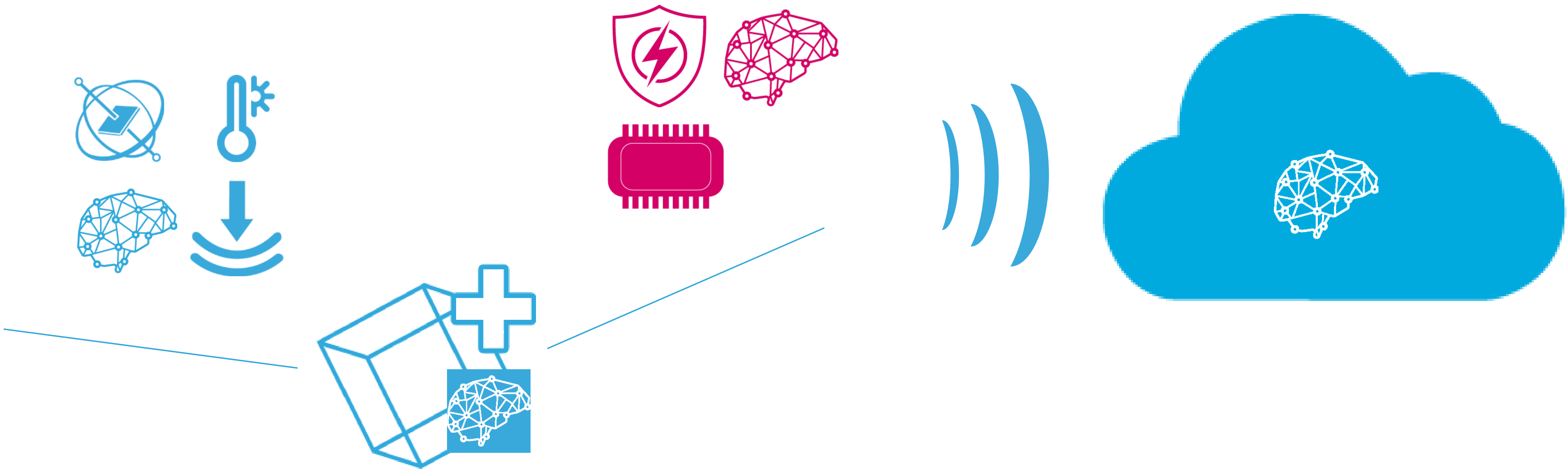
Edge A.I. Architecture

Artificial Intelligence in IoT



Edge A.I. Architecture

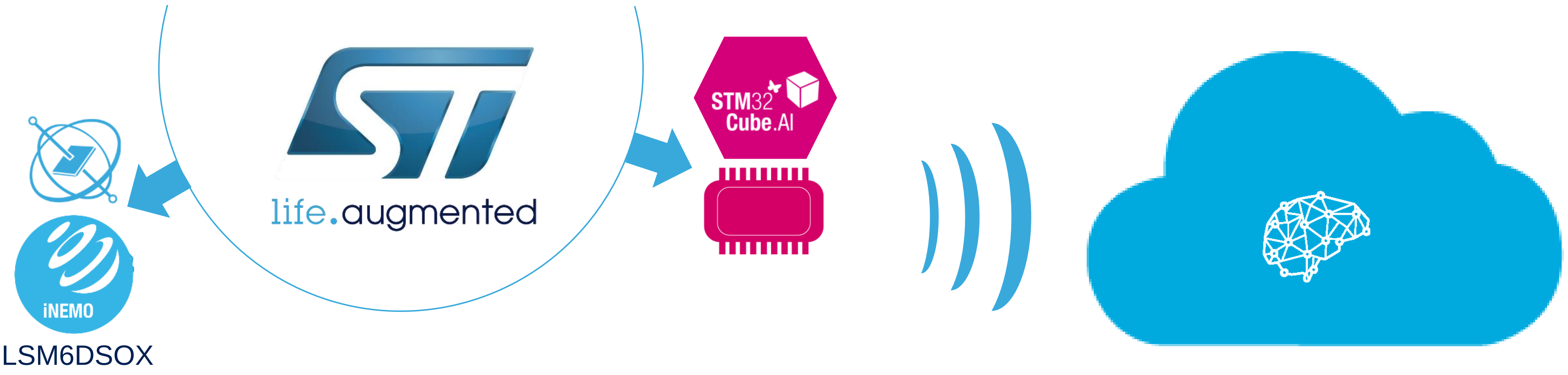
Artificial Intelligence in IoT



Edge A.I. Architecture

Artificial Intelligence in IoT

Artificial Intelligence in IoT



Data processing inside the sensor:

+ Local processing
+ Real time analysis
+ Reduced cost of bandwidth
+ Ultra Low Power
+ Intrinsic Security
= Configurable Logic
- Simple computation (Dec. Tree)

Data processing inside the STM32:

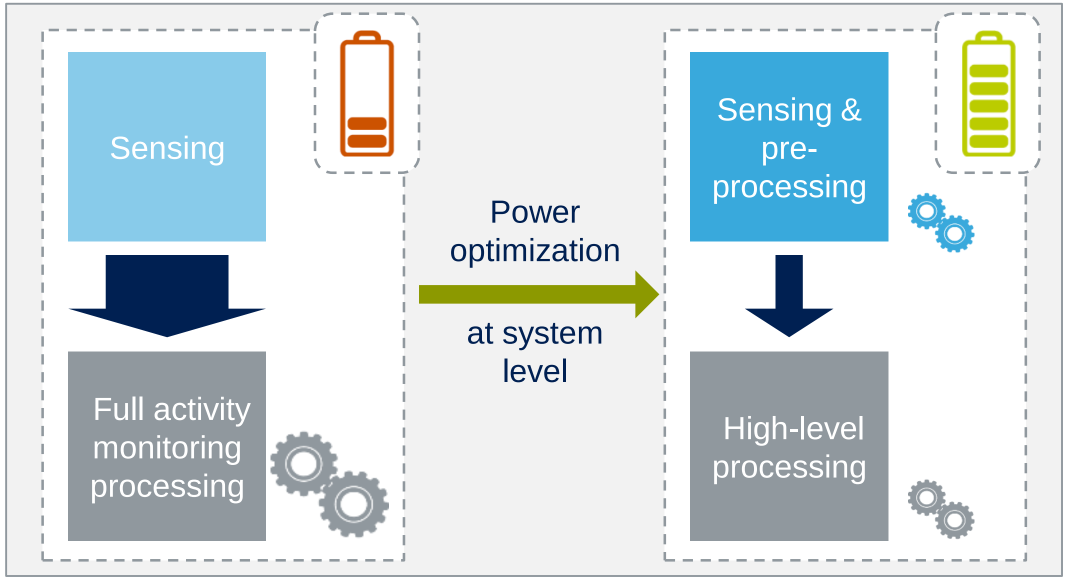
+ Local processing
+ Real time analysis
+ Reduced cost of bandwidth
= Low Power
= Improved security
= Configurable Logic
= Std Computation (Neural Net.)

Data processing inside the Cloud:

+ Advanced Computation
+ Availability of Wide Amount of Data
+ Continuous Algo Improvement
= Remote processing
- Data Transfer Latency
- High cost of bandwidth
- Very High current consumption

LSM6DSOX Smart Sensor

The First IMU Sensor with embedded Machine Learning Core



LSM6DSOX Unique Performance

Improved Accuracy, Optimized System Power

High Accuracy
Noise: Gyro 3.8 mdps/√Hz
Accelerometer 70 µg/√Hz

Low Current Consumption
0.55mA HP combo
-15% vs. LSM6DSL/M

New ultra low power Axl:
14uA @100Hz ODR

LSM6DSOX

2.5x3x0.86 mm

Sensor HUB
& compressed 9kB FIFO

Embedded Pedometer 2.x
WeChat Compliant

OIS Stabilization Core

New Standard Protocol
I3C / I2C / SPI

Finite State Machine &
Machine Learning Core

From Low Power Sensor

to Low Power System

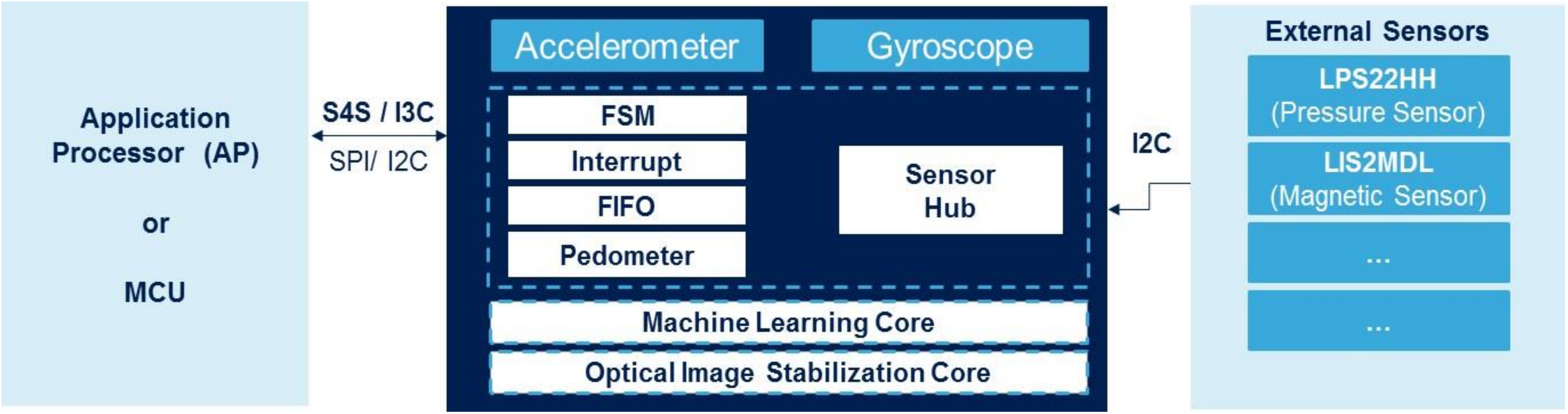
Best in class IMU **Power Consumption** (0.55 mA combo High Perf. Mode, 9uA Accel. only)

10 to 1000 time energy saving by running Machine Learning on Sensor HW (vs. AP)

Simplified And Lean System Concept based on 2 mainstreams:

1. Configurable power mode and high speed communication
2. Flexible HW solution running accurate algorithms

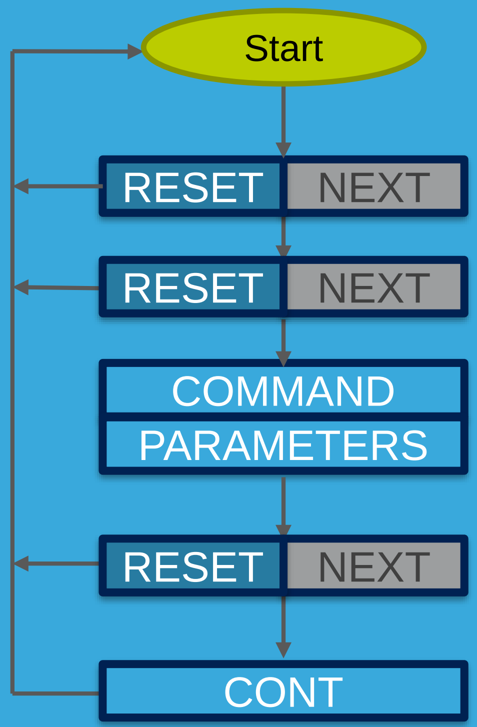
LSM6DSOX



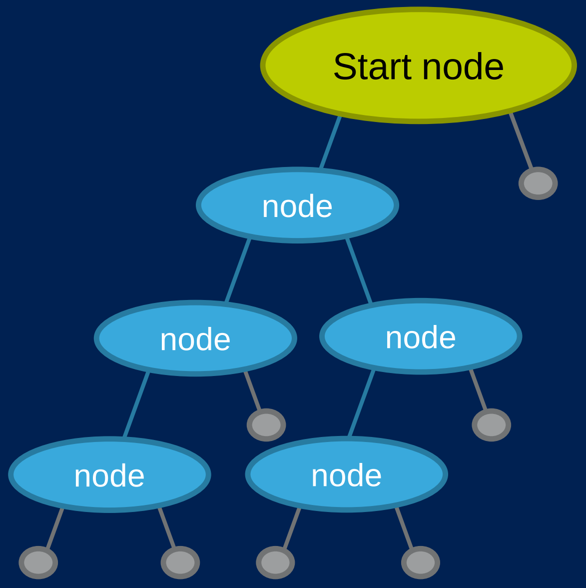
LSM6DSOX Smart Sensor

The first Smart Sensor Embedding:

Finite State Machine



Machine Learning Core



FSM & MLC allows sensors to process data with reduced help of a host MCU

Deductive vs. Inductive algorithms

Inertial Algorithms Overview

DEDUCTIVE

Hypothesis

↓

Rules

i.e. Wake Up

Finite State Machine

INDUCTIVE

Observation

↓

Model

i.e. Activity Rec

Machine Learning Processing

Finite State Machines overview

LSM6DSO/X - Finite State Machine

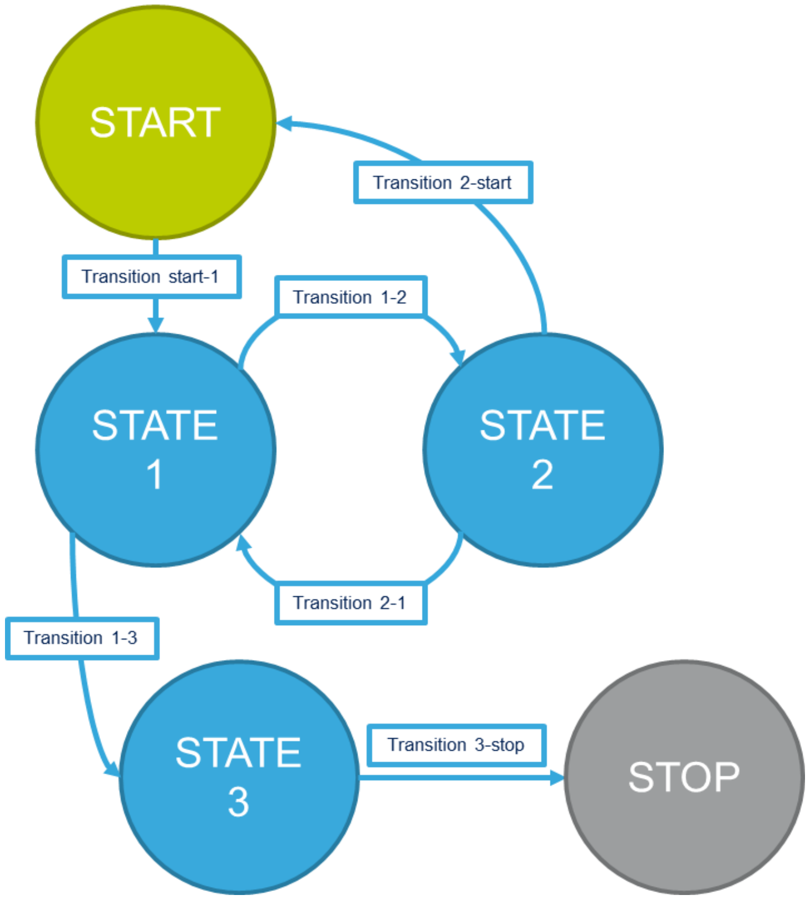
FSM is composed of a series of states, configurable parameters/resources and variables

Possibility to implement up to 16 independent FSM

FSM are executed simultaneously or sequentially

FSM can use data from accelerometer, gyro or an external sensor

FSM outputs:
- Interrupt
- Source information



Finite State Machine structure

In LSM6DSOX

Sensors Data
- Accelerometer
- Gyroscope
- External Sensor
- MLC Filtered Data

Data Section
- Fixed Data
- Variable Data

Instructions
- Conditions
- Commands
- Parameters

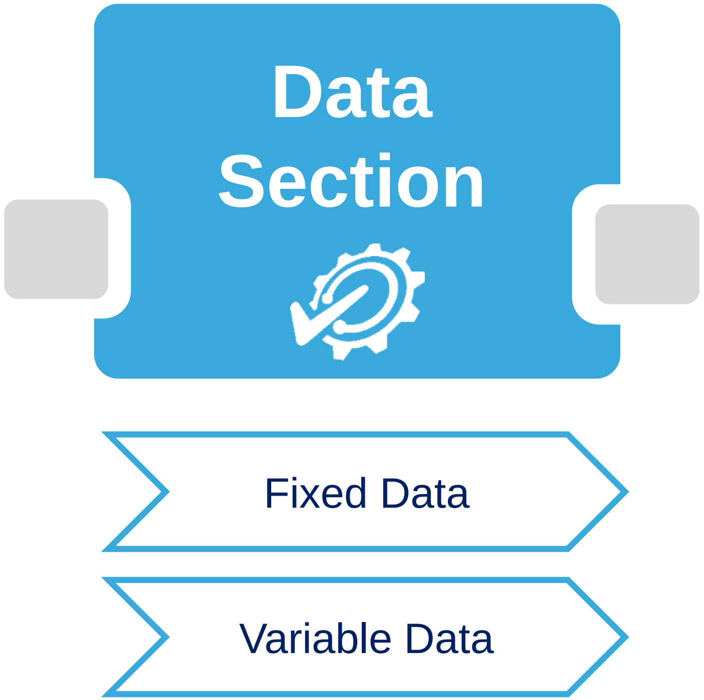
Finite State Machine structure

In LSM6DSOX

There are 2 Data Sections in each FSM program:

- The **Fixed Data Section** of the program is used to specify the resources that are needed in the instructions section. Resources examples:
    - Thresholds
    - Timers
    - Masks
    - Hysteresis
    - Decimation factor

- The **Variable Data Section** is used to define all the values for the used resources.



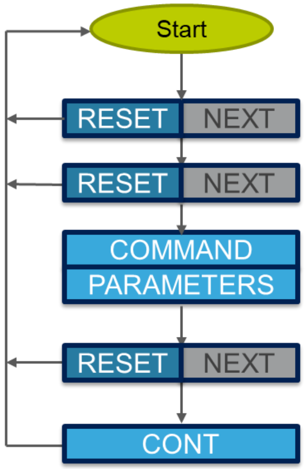
life.augmented

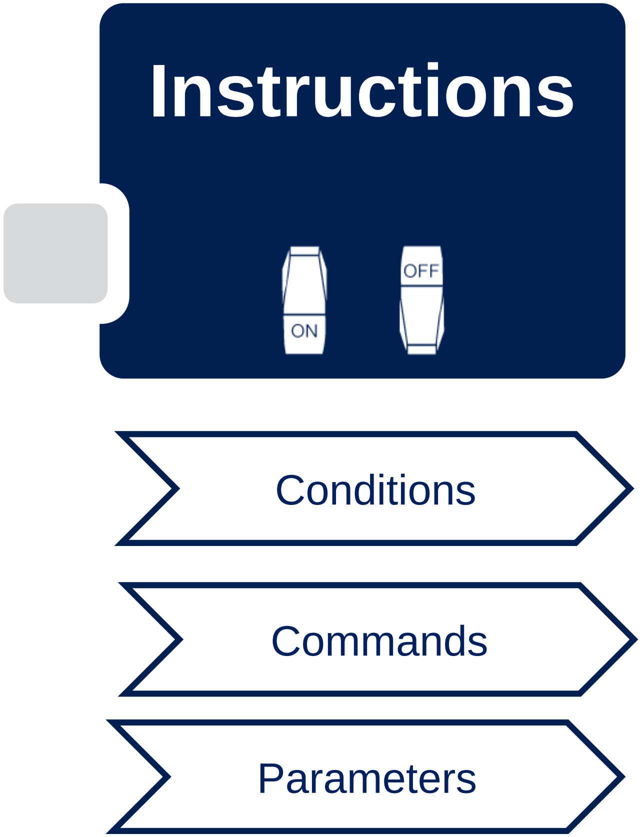
Finite State Machine structure

In LSM6DSOX

In the Instruction Section the user defines the algorithm by using available commands and conditions:

- **Commands** can be followed by parameters. Once executed the program pointer is set to the next command/condition line that is immediately evaluated.
- **Parameter** are application masks, threshold and timers.
- **Conditions** can be:
  - Reset. If true → pointer set to last Reset Point
  - Next. If true → pointer set to Next Line
  - None → interpreter keeps evaluating the same line when next sample occurs

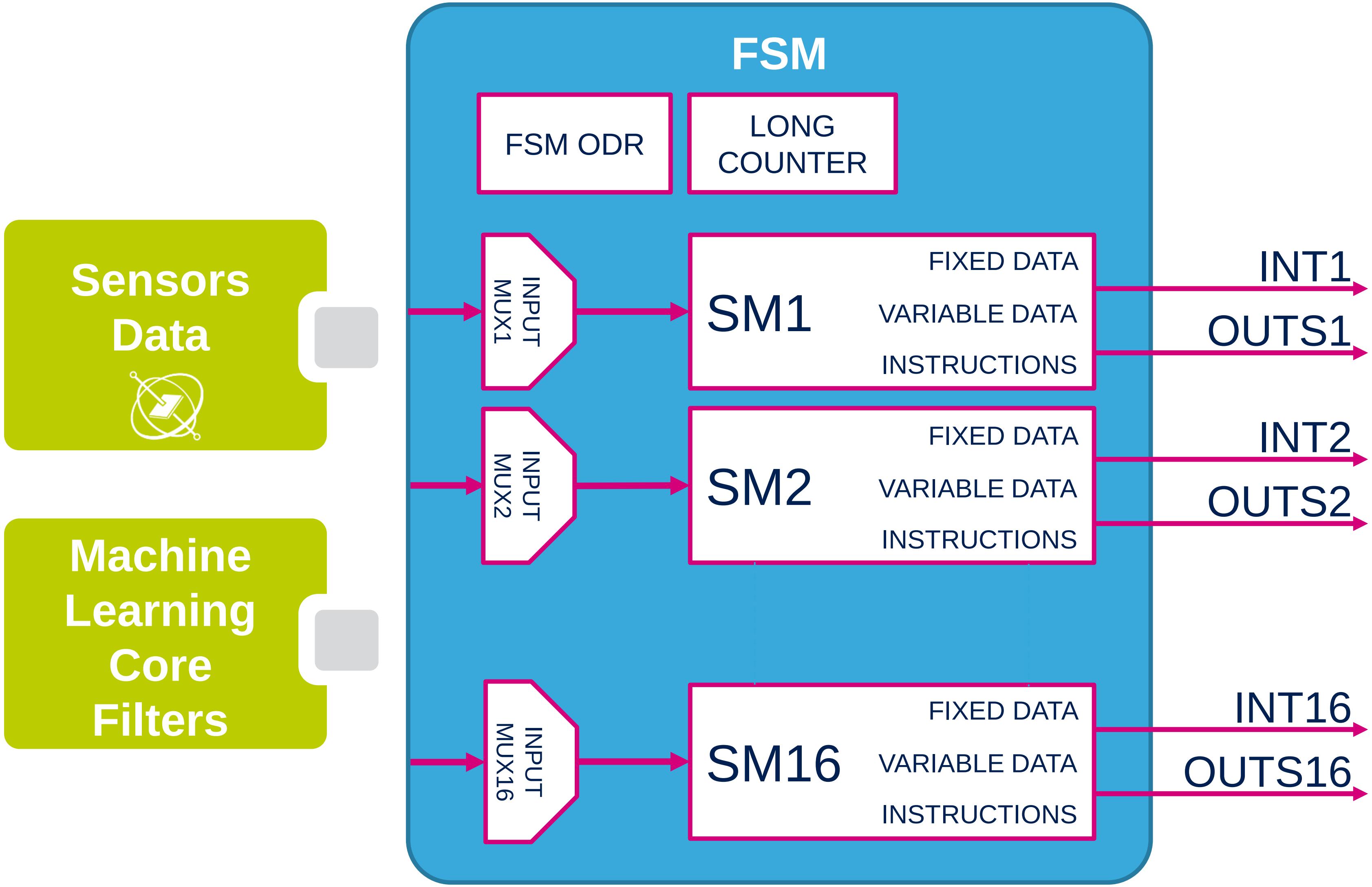




life.augmented

Finite State Machine Blocks

Summary



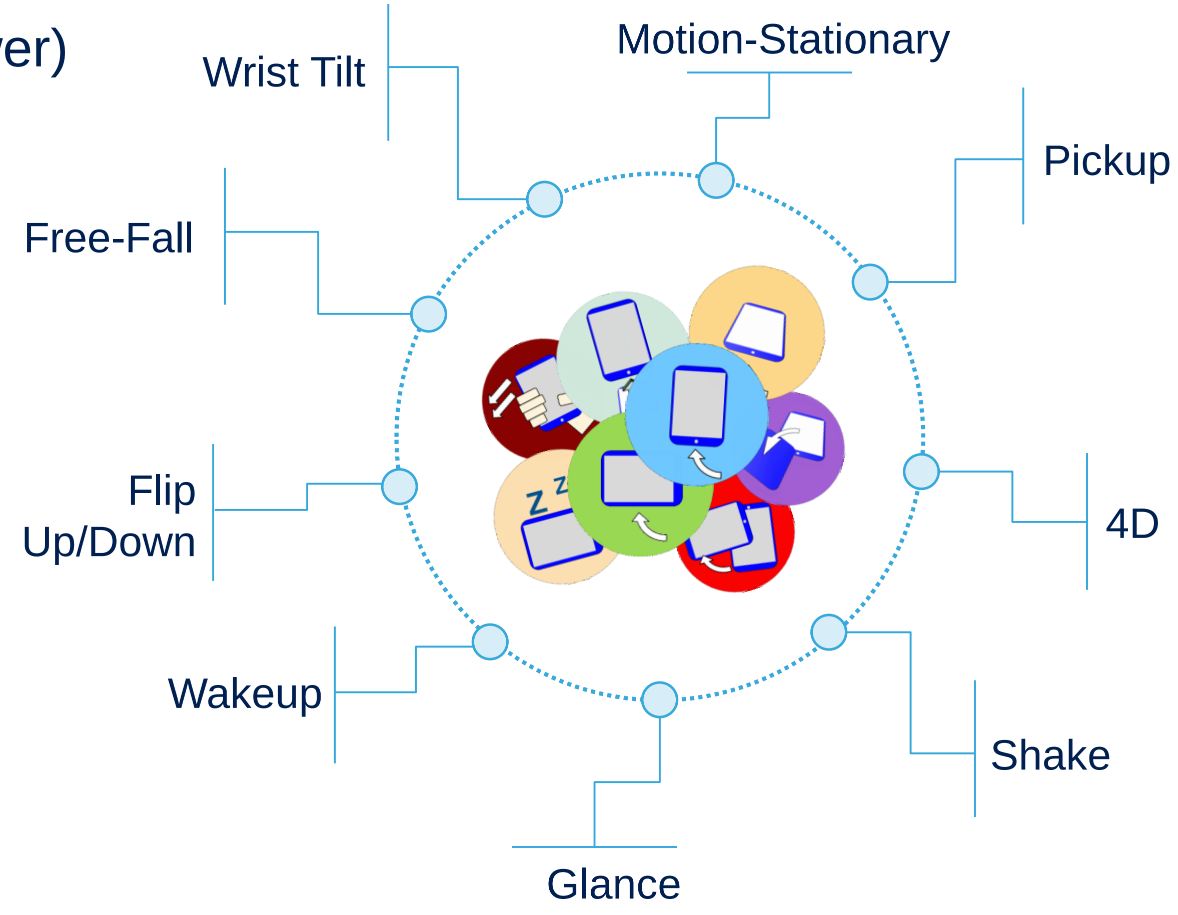
life.augmented

Finite State Machine

Examples of Libraries

Exploiting Finite State Machine capabilities

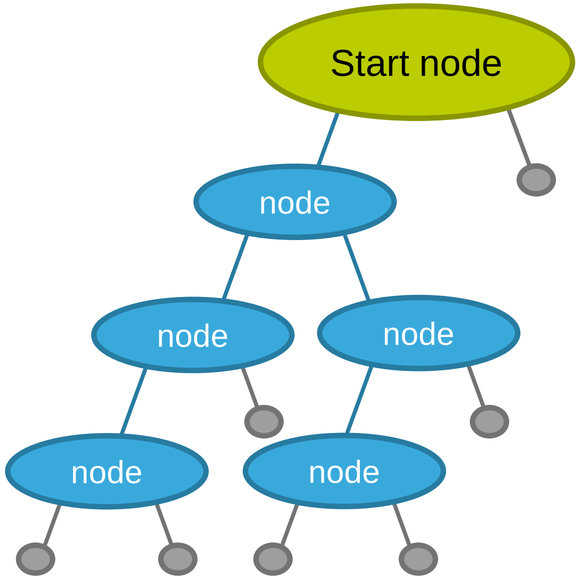
- High Efficiency (Computation Resources, Power)
- High Customization level
- Library of example gestures available:



life.augmented

Machine Learning Core Overview

LSM6DSOX - Machine Learning Core



MLC is a programmable logic.

LSM6DSOX embeds a Decision Tree Logic composed by a series of configurable nodes.

Each node is characterized by one “if-then-else” condition.

Up to 8 decision trees can be configured to run simultaneously in LMS6DSOX.

Machine Learning approach

Machine Learning Core – LSM6DSOX

- User defines Classes to be recognized

- Collects multiple Logs for each class

- Defines Features that better characterized the defined classes

- Based on Logs and Features, the Decision Tree can be generated using machine learning tools

- Configure the device and run the application

Envisioning

Self Learning Approach

✓ Log continuous acquisition enabling Decision Tree fine tuning

Machine Learning Core (MLC)

In LSM6DSOX

Sensors Data

- Accelerometer
- Gyroscope
- External Sensor

Computation Block

- Filters
- Digital Pre-defined Features

Decision Tree

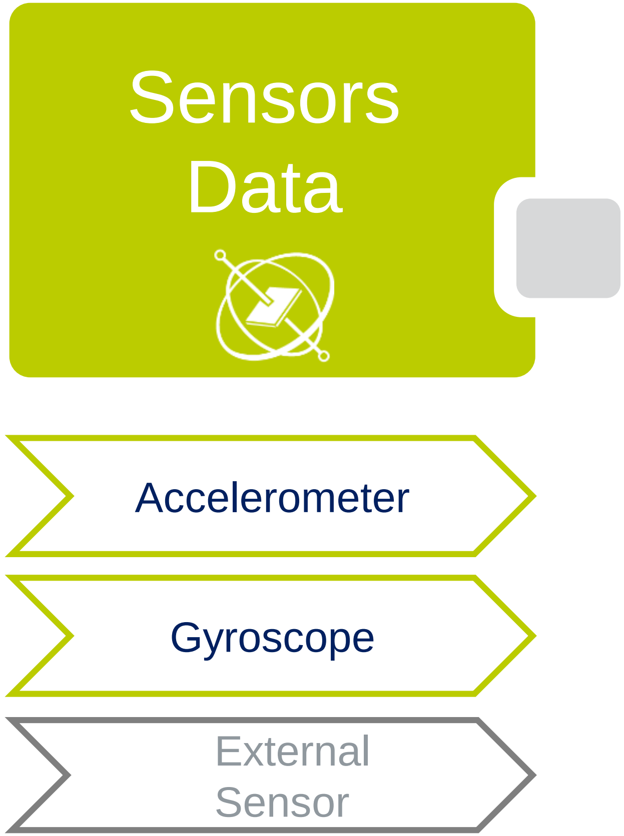
- Meta-classifier
- Node

life.augmented

Machine Learning Core (MLC)

In LSM6DSOX

- Wide set of Inputs available:
  - **Accelerometer** → $[a_x\ a_y\ a_z\ ], [a_v], [a^2{}_v]$
  - **Gyroscope** → $[g_x\ g_y\ g_z\ ], [g_v]\ , [g^2{}_v]$
  - **External sensor** → $[m_x\ m_y\ m_z\ ], [m_v], [m^2{}_v] (i.e. magnetometer)$
  - **Magnitude** → $V = \sqrt{X^2 + Y^2 + Z^2}$

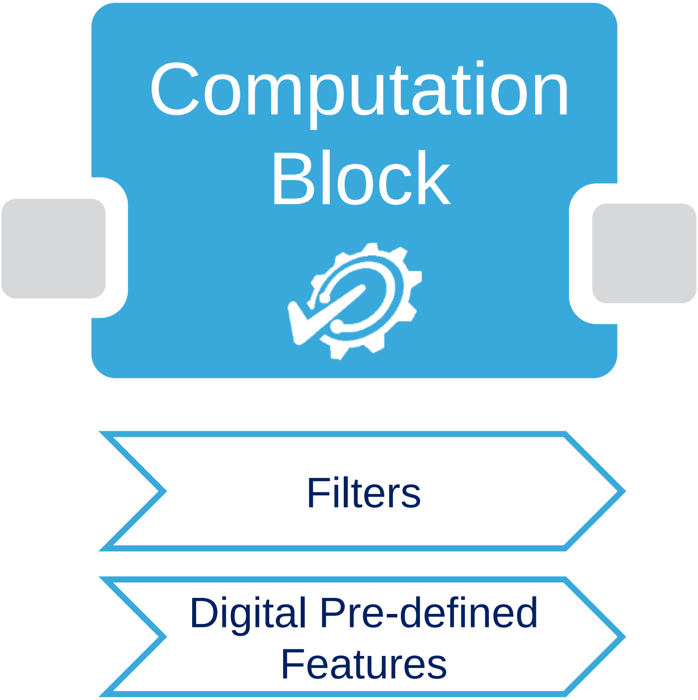


life.augmented

Machine Learning Core (MLC)

In LSM6DSOX

- Sensors data can be filtered with a 2nd order IIR **Filter**.
- **Features** are statistical parameter calculated from:
    - Input Data
    - Filtered Input Data
- Examples of features are: Mean, Variance, Energy, Peak to Peak, …

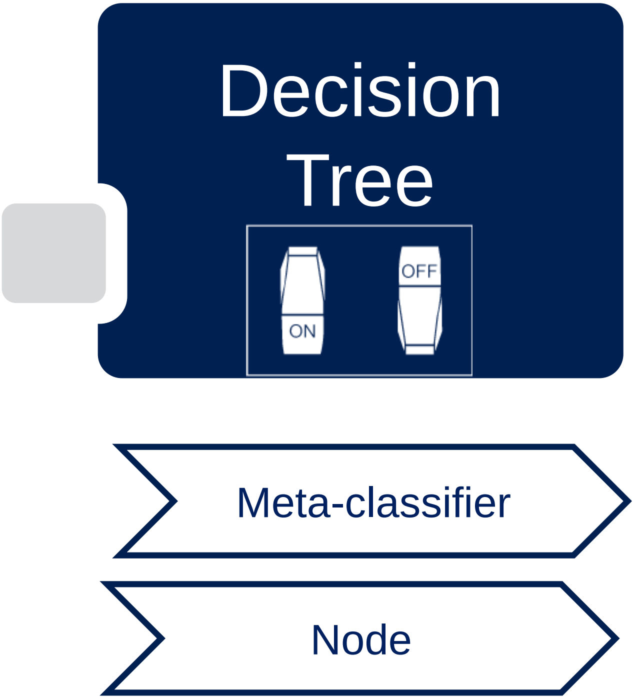


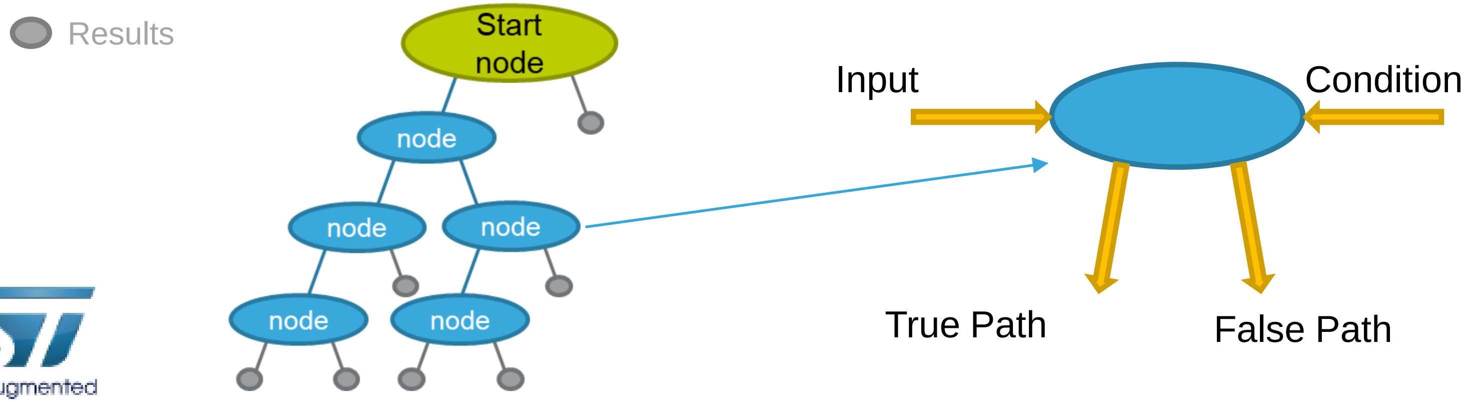
life.augmented

Machine Learning Core (MLC)

In LSM6DSOX

- The **Decision Tree** is a predictive model built from training data. The outputs of the computation blocks are the inputs of the decision tree.
- Each **Node** is characterized by one «if-then-else» condition. Some examples of conditions:
  - Mean on Acc_X < 0.5 g
  - Variance on Gyro_Z < 200 dps
- Decision tree can either generate a result at every sample or filter the results with a **Meta-classifier**, to have a more robust output.





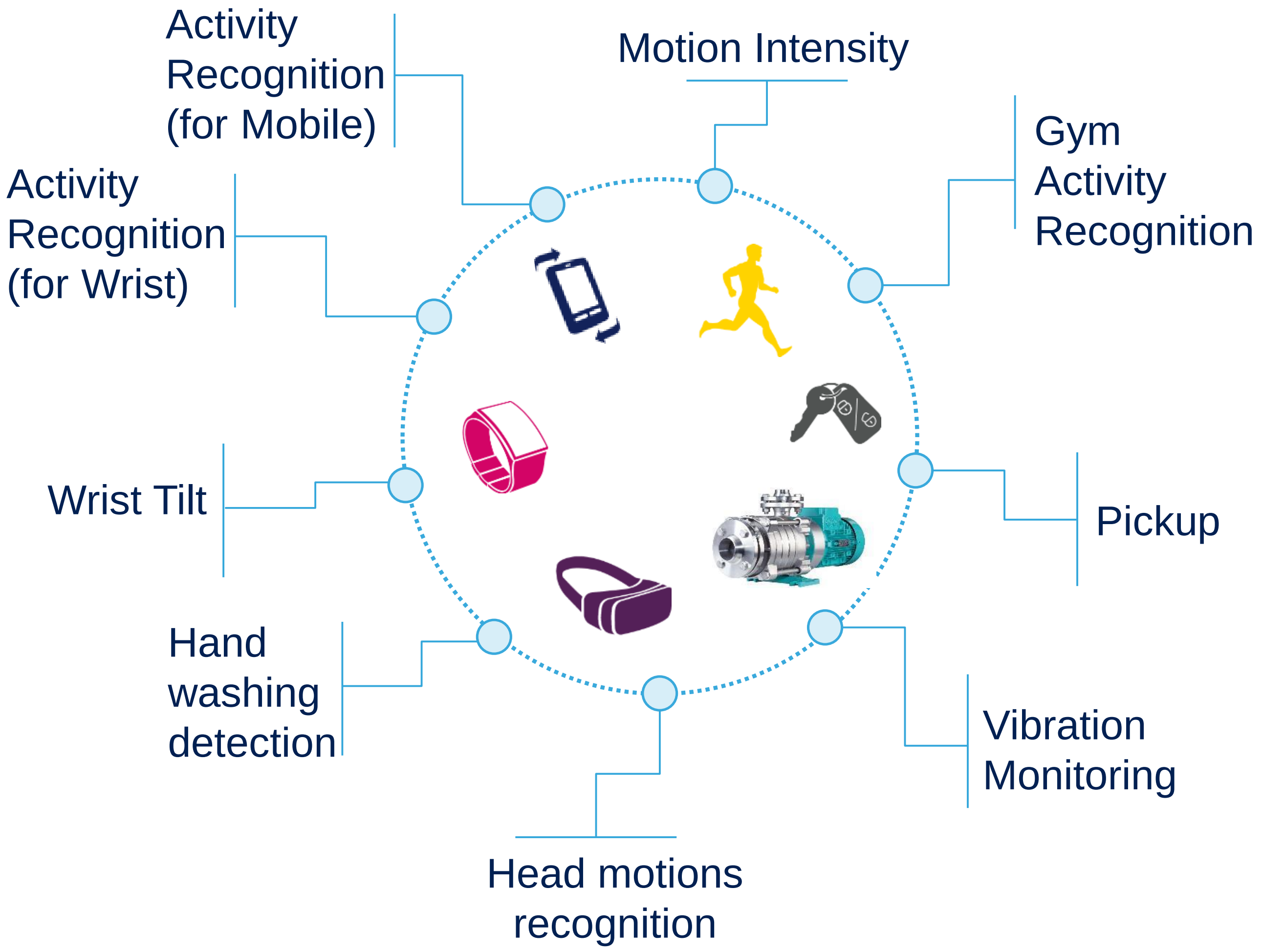
life.augmented

Machine Learning Core

Examples of Libraries

Exploiting Machine Learning Core capabilities

- High Efficiency (Computation Resources, Power)
- High Customization level
- Higher Complexity than FSM
- Example Libraries available:



life.augmented

Comparison of Solutions

LSM6DSOX – Smart Sensor

The First IMU Sensor with embedded Machine Learning Core

Industrial Robots

Personal Electronics

Sensors Data → **Computation Block** → **Decision Tree**

ON OFF

life.augmented

LSM6DSOX Lowest Current Consumption

FSM and MLC are incredibly efficient in current consumption needs:

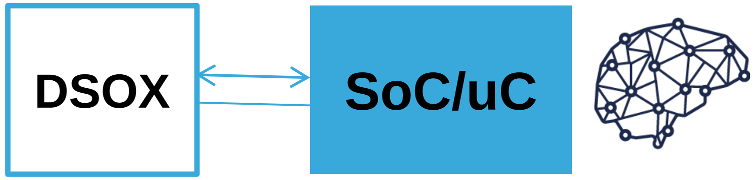
+ 3 µA for each Finite State Machine

+ 1-15 µA Machine Learning Core

as low power features to recognize customizable conditions and generate interrupts.

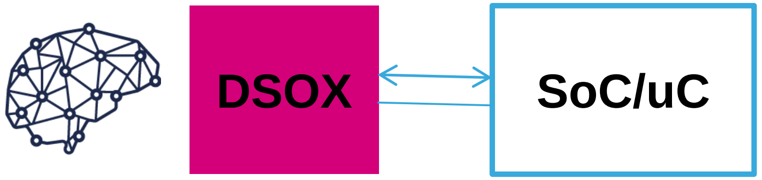
System Level Benefits using LSMDSOX

Activity Recognition in **SW**



Real Time Data

Activity Recognition in **DSOX**



Interrupt

Parameter	Activity Recognition in SW	Activity Recognition in DSOX	**Benefit in using MLC**
DSOX current	●	●	**Similar DSOX consumption**
MIPS uC/SoC	● ● ● ● ●	● ●	**Offload SoC / uC**
Traffic	● ● ● ● ●	●	**Less traffic on Bus**
Battery usage	● ● ● ● ●	● ●	**Reduction of System current**

life.augmented

Computational Current Consumption

Applications has been developed on Machine Learning Core

Algorithm	Sensors used	ODR	Number of decision trees	Number of nodes	MLC additional Current consumption
Vibration Monitoring	A	26 Hz	1	2	1 uA
Motion Intensity	A	12.5 Hz	1	7	1 uA
6D position recognition	A	26 Hz	1	8	2 uA
Activity Recognition for mobile	A	26 Hz	1	126	4 uA

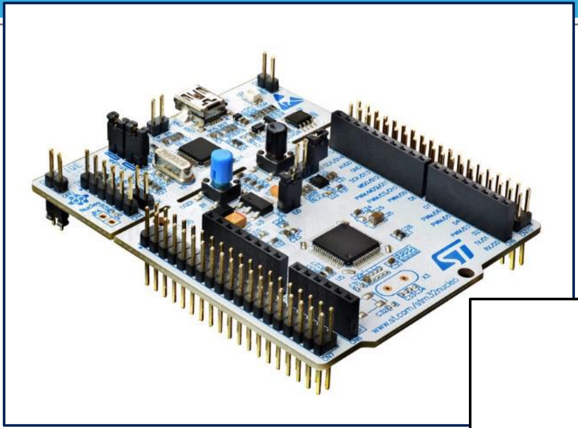
Application Case

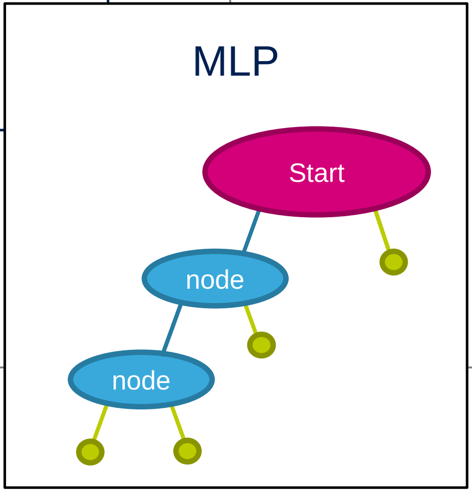
Same application developed on low power microcontroller and on Motion Learning Processing

Activity Recognition Algorithm Classes

- Stationary
- Walking
- Fast Walking
- Jogging

…



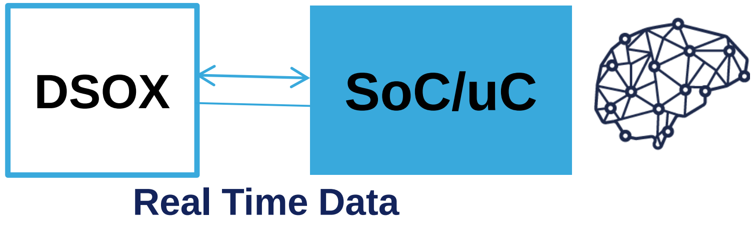


Additional Current Consumption Required to Run the Application

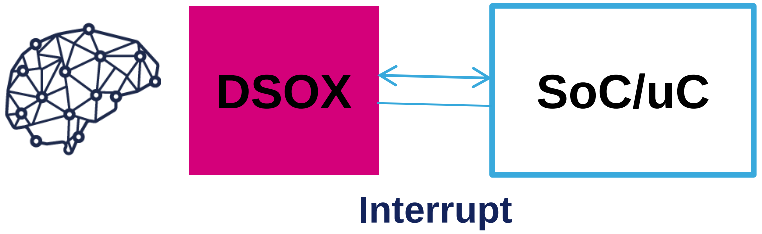
Running on:	[µA]
Cortex-M3 STM32L152RE@32MHz	240
MLC on LSM6DSOX	7

System Level Benefits using LSMDSOX

Activity Recognition in **SW**



Activity Recognition in **DSOX**



Same performance on Confusion Matrix on SW Vs. algo running in DSOX

Detected as ->	Stationary	Walking	Fast Walking	Jogging
Stationary	99.1%	0.9%	0.0%	0.0%
Walking	0.0%	99.4%	0.2%	0.0%
Fast Walking	0.0%	3.7%	95.9%	0.2%
Jogging	0.0%	0.6%	0.7%	98.5%

50x less current for same performance

Tools

ST MEMS Sensor evaluation kit

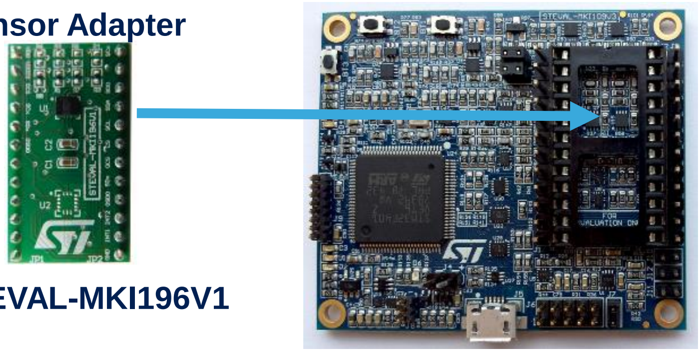
Professional Unico software Tool and ProfiMEMS

HW board

- High level tool for device configuration, data logging and programming
- Support FSM & Machine Learning Core Configuration Tool
- FSM / Decision Tree outputs visualization and Logging
- Different configuration of Power Supply (Vdd & Vdd_IO separately)
- Device power consumption measurement (with configurable power mode by SW)

STEVAL-MKI109V3

Sensor Adapter



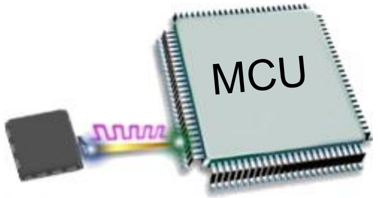
STEVAL-MKI196V1

Software tool

Device Configuration

Runtime verification

CONFIGURATION .UCF file



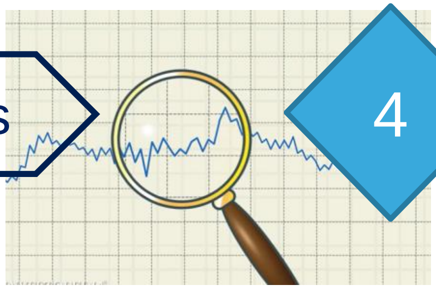
ST Unico GUI software

- Unico Lite GUI (or STSW-MEMS034) is an user interface that assists users to graphically and easily:
  - configure registers,
  - see the sensor's behavior on a chart,
  - export settings in a header file to include in the source code of an application.

1 Registers configuration

2 Data Collection

3 Pattern analysis



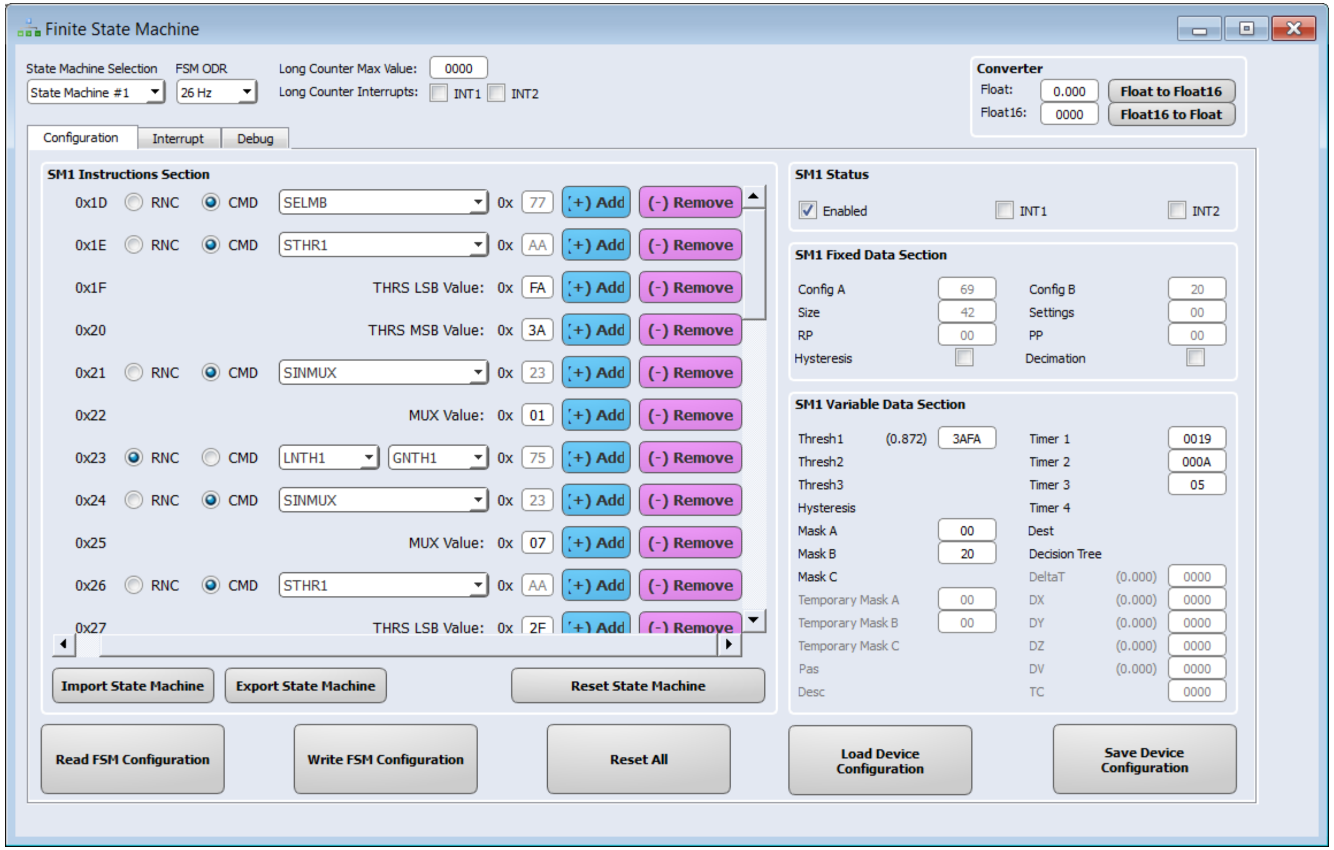
4 FSM Implementation

5 MLC Configuration

life.augmented

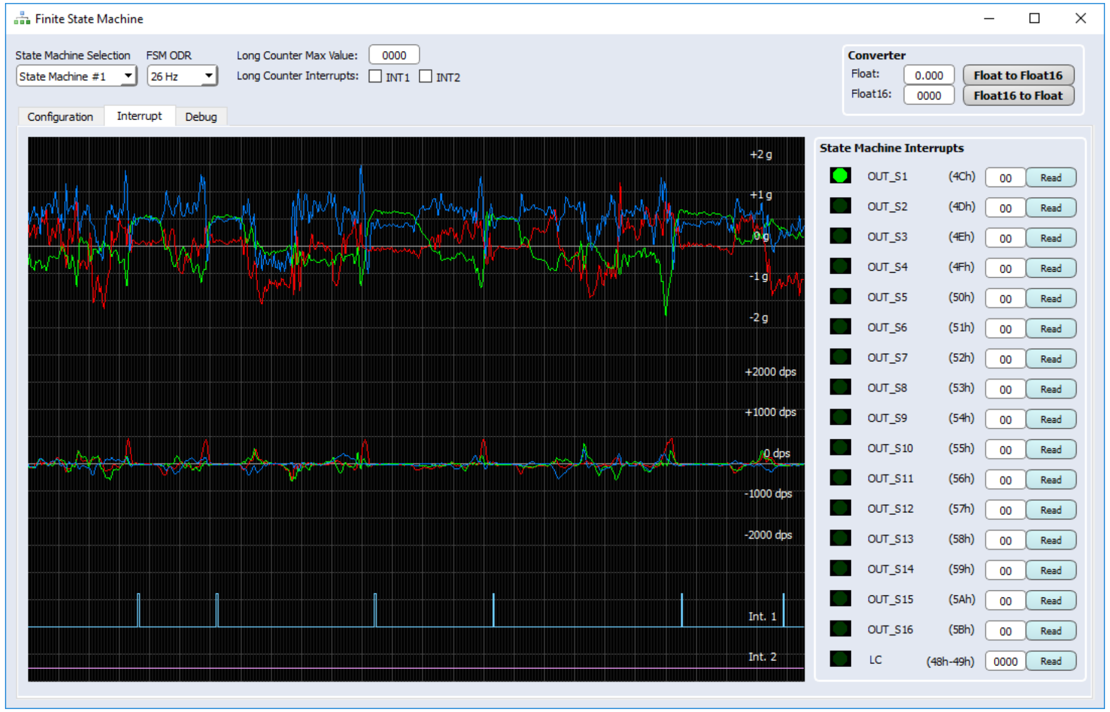
FSM examples

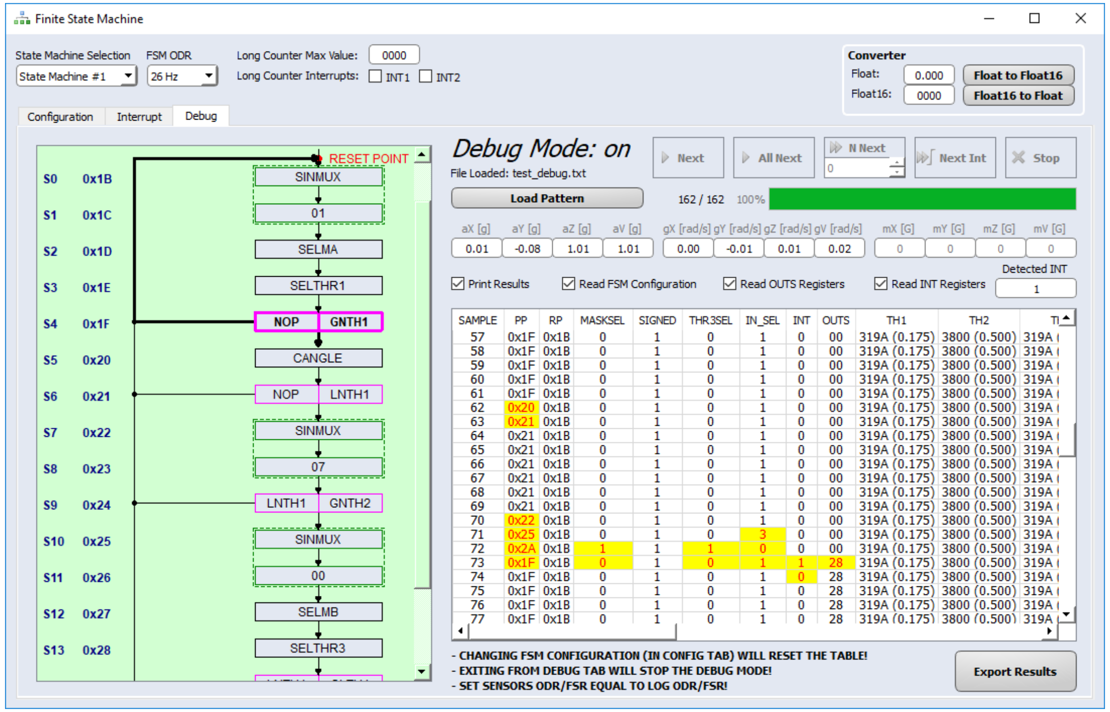
- ST provides FSM examples to start experimenting and test.



Fine Tuning

It is possible to debug in real time the FSM behavior playing back previously recorded data.



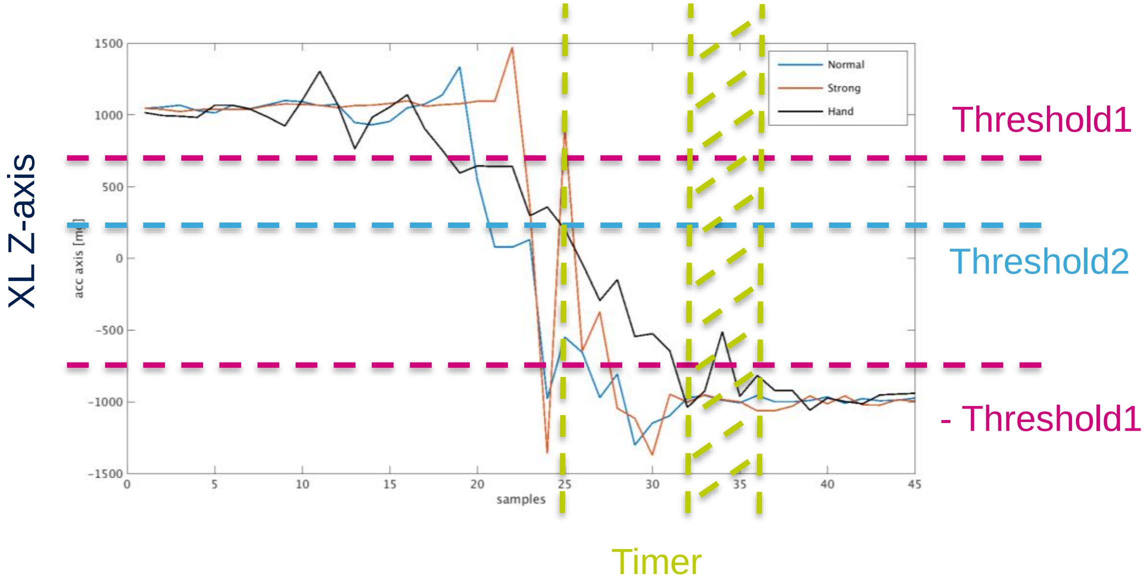


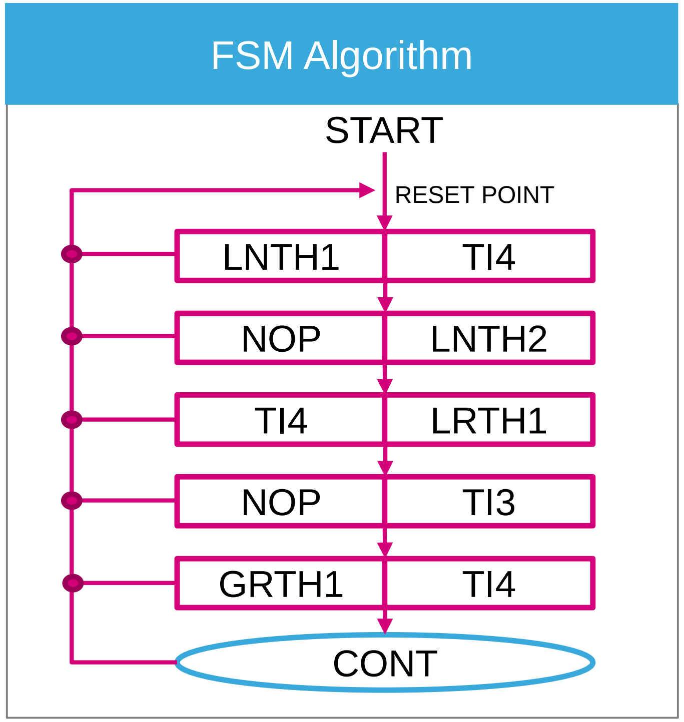
FSM can detect

when the user picks up a product

- Option 1: Programmers can load the example, write it onto the sensor, and start experimenting with it right away by changing threshold values or timers, for instance.

- Option 2: You can replay a series of events to see if the FSM would respond appropriately and throw an interrupt at the right moment.

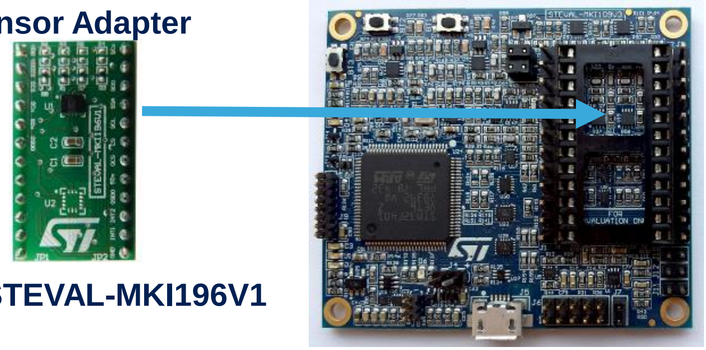




life.augmented

Start Working with ProfiMEMS Board

- The best way to start experimenting and prototyping with the LSM6DSO/X is:
  - STEVAL-MKI196V1 daughter board.
  - STEVAL-MKI109V3 motherboard.



- Users need to plug the sensor board into the main board and connect it to a PC using the onboard USB port.

- Unico v8.0 supports FSM & MLC on Windows, Linux, Mac

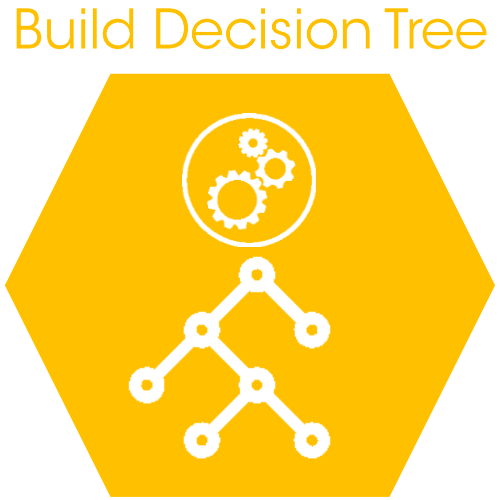
Part Number	Software Version	Marketing Status	Supplier	Download
STSW-MKI109W	8.0.0	Active	ST	Get Software

life.augmented

Build the Decision Tree through ML tools

External tools for decision tree generation

- The decision tree can be generated by a dedicated Machine Learning tools:
  WEKA
  - Attributes selection (from .ARFF file)
  - Data filtering
  - Decision tree generation
  - Decision tree performance evaluation (e.g. number of nodes, accuracy, confusion matrix, etc…)



WEKA The University of Waikato

rapidminer

MATLAB

python™

- Or other alternative tools:
  - RapidMiner → Example available in AN5259 rev2
  - Matlab → Scripts available at https://github.com/STMicroelectronics/STMems_Machine_Learning_Core/tree/master/tools/matlab
  - Python → Scripts available at https://github.com/STMicroelectronics/STMems_Machine_Learning_Core/tree/master/tools/python

life.augmented

LSM6DSOX

Reference & Support

- LSM6DSOX ST datasheet
  - LSM6DSOX: LSM6DSOX.html
- Application Notes:
  - AN5272 (LSM6DSOX): DM00571818.pdf
  - AN5273 (LSM6DSOX Finite State Machine): DM00572971.pdf
  - AN5259 (LSM6DSOX Machine Learning Core): DM00563460.pdf
- Video Tutorial
  - Getting Started with LSM6DSOX: LSM6DSOX.html
- Boards:
  - Professional MEMS tool board - STEVAL-MKI109V3: mems-motion-sensor-eval-boards/steval-mki109v3.html

  - LSM6DSOX Adapter board - STEVAL-MKI197V1: mems-motion-sensor-eval-boards/steval-mki197v1.html

- Unico GUI package v8.0 - SW package for boards
  - STSW-MKI109W for Windows OS: evaluation-tool-software/stsw-mki109w.html
  - STSW-MKI109L for Linux OS: evaluation-tool-software/stsw-mki109l.html
  - STSW-MKI109M for Mac OS: evaluation-tool-software/stsw-mki109m.html
- MEMS Community
  - Community: community.st.com/s/group/CollaborationGroup
  - Q&A: mems-and-sensors

life.augmented

Takeaways

- LSM6DSOX completely transforms what a sensor can process without help of a host microcontroller (MCU), thanks to its unique Machine Learning Core.
- Exceptional advantage in terms of Power consumption and Flexibility.
- ST is supporting your projects by refence design, dedicated tools and example libaries.

Thank You!

life.augmented