Artificial Intelligence Solutions
The key steps behind Neural Networks

1. Capture data
2. Train NN Model
3. Clean, label data
   Build NN topology
4. Convert NN into
   optimized code for MCU
5. Process & analyze new
data using trained NN
ST toolbox for Neural Networks

Capture data

Clean, label data
Build NN topology

Convert NN into optimized code for MCU

Process & analyze new data using trained NN

STM32

STM32 Cube.AI
STM32CubeMX extension
AI conversion tool

Input your framework-dependent, pre-trained Neural Network into the STM32Cube.AI conversion tool

Automatic and fast generation of an STM32-optimized library

STM32Cube.AI offers interoperability with state-of-the-art Deep Learning design frameworks

Any framework that can export models in ONNX open format can be imported

Train NN Model

Convert NN into optimized code for MCU

Process & analyze new data using trained NN
Possible conversion strategies: Network code generation and interpreter

More Flexible: TensorFlow Lite interpreter mode

- Pre-trained model
- Flat buffer .tflite
- TFLite interpreter
- User app
- Model is interpreted and executed by pre-built ops

More optimized: Optimized C code generated by STM32 Cube.AI

- Pre-trained model
- STM32 Cube.AI
- NN C files
- STM32.AI lib
- STM32 BSP
- STM32 device
- User app
- Model is pre-compiled and linked only with used ops

TensorFlow Lite run-time on STM32
Collecting data & architecting a NN topology

Services provided by Partners
- Capture data
- Clean, label data
- Build NN topology

ST tools to support
- ST BLE Sensor mobile phone application
  Collect and label data from the SensorTile.

Selected partners
- Neural Networks engineering services support.
- Data scientists and Neural network architects.
ST toolbox for neural networks more than just a conversion tool

• Function packs for quick prototyping
• Audio, Motion and Vision examples

• STM32 Community with dedicated Neural Networks topic
• For support and idea exchange

Process & analyze new data using trained NN

Convert NN into optimized code for MCU
STM32 solutions for AI
More than just the STM32Cube.AI

An extensive toolbox to support easy creation of your AI application

1. **Software examples** for Quick prototyping
   Audio, Motion and Vision Function packs
   On ST development Hardware

2. **STM32 Community** with dedicated
   Neural Networks topic

3. Trainings, hands on, MOOCs and
   partners videos

4. **STM32 AI Partner Program**
   with dedicated Partners providing
   Machine or Deep Learning engineering services

---

1. Capture Data
2. Label Data
3. Train NN
4. Run on STM32 Cube.AI

---

**STM32 Cube.AI**

AI extension for STM32CubeMX to map pre-trained Neural Networks
Example form factor hardware to capture and process data

Capture data

STM32L4

STM32L476: Cortex-M4

Balun Filter

Motion MEMS

MP34DT04: Microphone

LPS22HB: Barometer

BlueNRG-MS: Bluetooth low-energy

Motion MEMS

LSM6DSM: 3DAcc+3Dgyro

LSM303AGR: 3DAcc+3DMag

Process & analyze new data using trained NN

www.st.com/SensorTile

www.st.com/SensorTile-edu
Fast go to market module to capture data with more accuracy

More advanced, high accuracy and low power sensors
- First Inertial module with Machine Learning capabilities.
- Motion (accelerometer and gyroscope, magnetometer) and slow motion (inclinometer)
- Altitude (pressure), environment (pressure, temperature, humidity, compass) and sound (sound and ultrasound analog microphone)
- Microsoft IoT services ready to make available on a web dashboard the result of the embedded processing

www.st.com/SensorTileBox
Distributed AI: sensor + STM32
Optimize performance and power consumption

**Smart Sensor**
with Machine Learning Core

- Best ultra-low-power sensing at high performance
  - 550µA (gyroscope and accelerometer)
    ➔ 200µA less than closest competitor
  - 20~40µA (Accelerometer only for HAR)
- Efficient Finite State Machines: 2µA
- Configurable Machine Learning Core: 4~8µA

**Smart STM32**
Second level of AI processing

- More advanced and complex NNs
- Decisions on multiple sensors
- NN input can be sensor data and/or sensor Machine Learning decisions
- Multiple Neural Networks support
- Actuation & communication

- FSM up to 16
- MLC up to 8

Inertial Sensor
New LSM6DSOX

Raw Data
Event Decision
FSM and MLC Re-configuration

Deep Learning
Neural Networks
Machine Learning

- 550µA (gyroscope and accelerometer)
- 200µA less than closest competitor
- 20~40µA (Accelerometer only for HAR)
Form factor hardware
AI IoT Node for more connectivity

- IoT Node
- Capture data
- Process & analyze new data using trained NN

More debug capabilities
- Integrated ST-Link/V2.1
- PMOD extension connector
- Arduino Uno extension connectors

https://www.st.com/IoTnode
OpenMV integration
Fast machine vision prototyping

Configure Machine Vision in real-time over USB in Python

Run and validate optimized Neural Network

OpenMV CAM
Running MicroPython over STM32

https://github.com/openmv/openmv
Function Packs

Simple, fast, optimized
Audio scene classification (ASC)

Audio example in FP-AI-SENSING1 package

Audio Data capture

Labelling controlled by smartphone application

Data stored on the device SD card for future learning

Indoor, Outdoor, In vehicle labelling

Embedded audio pre-processing

Inferences running on the microcontroller

NN & example dataset provided

Inference result displayed on mobile app

Demo available
Human activity recognition (HAR) Motion example in FP-AI-SENSING1 package

Motion Data Capture

Labelling controlled by smartphone application

Data stored on the device SD card for future learning

Stationary, walking, running, biking, driving labelling

Embedded motion pre-processing

Inferences running on the microcontroller

NN & example dataset provided

Inference result displayed on mobile app

Demo available
Image classification
Vision example in FP-AI-VISION1 package

Enjoy the food classification demo
- Default demo based on 18 classes (224x224 RGB pictures)
- Several camera image output size possible

Full end-to-end optimized software example
- from camera acquisition to image pre-processing before feeding the NN
- Multiple memory mapping possibilities to optimize and test impact on performances
- Retrain this NN with your own dataset
- Quantize your trained network to optimized inference time and memory usage

Embedded image pre-processing (SW) on the STM32H747

Inferences running on the microcontroller

Pizza 99%
150ms

Demo available
## Making AI Accessible Now

### Leader in Arm® Cortex®-M 32-bit General Purpose MCU

<table>
<thead>
<tr>
<th>ARM Cortex® core</th>
<th>STM32F0</th>
<th>STM32G0</th>
<th>STM32F1</th>
<th>STM32F2</th>
<th>STM32F3</th>
<th>STM32F4</th>
<th>STM32F7</th>
<th>STM32G4</th>
<th>STM32L0</th>
<th>STM32L1</th>
<th>STM32L2</th>
<th>STM32L4</th>
<th>STM32L4+</th>
<th>STM32F7</th>
</tr>
</thead>
<tbody>
<tr>
<td>-M0</td>
<td>106 CoreMark 48 MHz</td>
<td>142 CoreMark 64 MHz</td>
<td>177 CoreMark 72 MHz</td>
<td>398 CoreMark 120 MHz</td>
<td>245 CoreMark 72 MHz</td>
<td>324 CoreMark 180 MHz</td>
<td>1024 CoreMark 216 MHz</td>
<td>550 CoreMark 170 MHz</td>
<td>75 CoreMark 32 MHz</td>
<td>93 CoreMark 32 MHz</td>
<td>424 CoreMark 110 MHz</td>
<td>273 CoreMark 80 MHz</td>
<td>409 CoreMark 120 MHz</td>
<td>161 CoreMark 48 MHz</td>
</tr>
<tr>
<td>-M0+</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-M3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-M33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-M4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-M7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dual -A7 &amp; -M4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **STM32MP1**: 4158 CoreMark, 650 MHz Cortex -A7, 209 MHz Cortex -M4

Compatible with **Machine Learning** Partner ecosystems

Compatible with **Deep Learning** STM32Cube.AI ecosystem

More than 40,000 customers  
Over 4 Billion STM32 shipped since 2007
AI solutions for STM32MP1

Running AI on ST Microprocessors
STM32MP1 microprocessor
Augmented intelligence

- STM32Cube.AI to convert pre-trained NNs for the Cortex-M4 core
- TensorFlow Lite STM32MP1 support up streamed for native NN inferences support on the dual Cortex-A side
Application examples in C/C++ and Python
- Image classification: 1000 objects classified
- Multiple object detection: 90 classes

Includes code for camera acquisition and image pre-processing

USB camera or built-in camera module

**Inferences** running on the microprocessor in 80ms for image classification

Displayed on STM32MP1-DK2, STM32MP1-EV1 and Avenger96 board

► 2x demos available
For more information

www.st.com/STM32CubeAI
Releasing your creativity

/STM32
@ST_World
community.st.com
www.st.com/STM32CubeAI