tinyML® Talks

Enabling Ultra-low Power Machine Learning at the Edge

"Hardware-aware Edge AI using the parameterizable ML accelerator UltraTrail"

Paul Palomero Bernardo - University of Tübingen

September 22, 2021

www.tinyML.org

tinyML Talks Sponsors and Strategic Partners

TINY ML TALKS webcast

arm
tinyML Strategic Partner

Deeplite
tinyML Strategic Partner

EDGE IMPULSE
tinyML Strategic Partner

emza visual sense
tinyML Strategic Partner

GREENWAVES TECHNOLOGIES
tinyML Strategic Partner

LatentAI
Adaptive AI for a Smarter Edge
tinyML Strategic Partner

HOTG
tinyML Strategic Partner

imagimob
tinyML Strategic Partner

maxim integrated
NOW PART OF ANALOG DEVICES

Qeexo
tinyML Strategic Partner

Qualcomm
tinyML Strategic Partner

RealityAI
tinyML Strategic Partner

seeed studio
The IoT Hardware Enabler
tinyML Strategic Partner

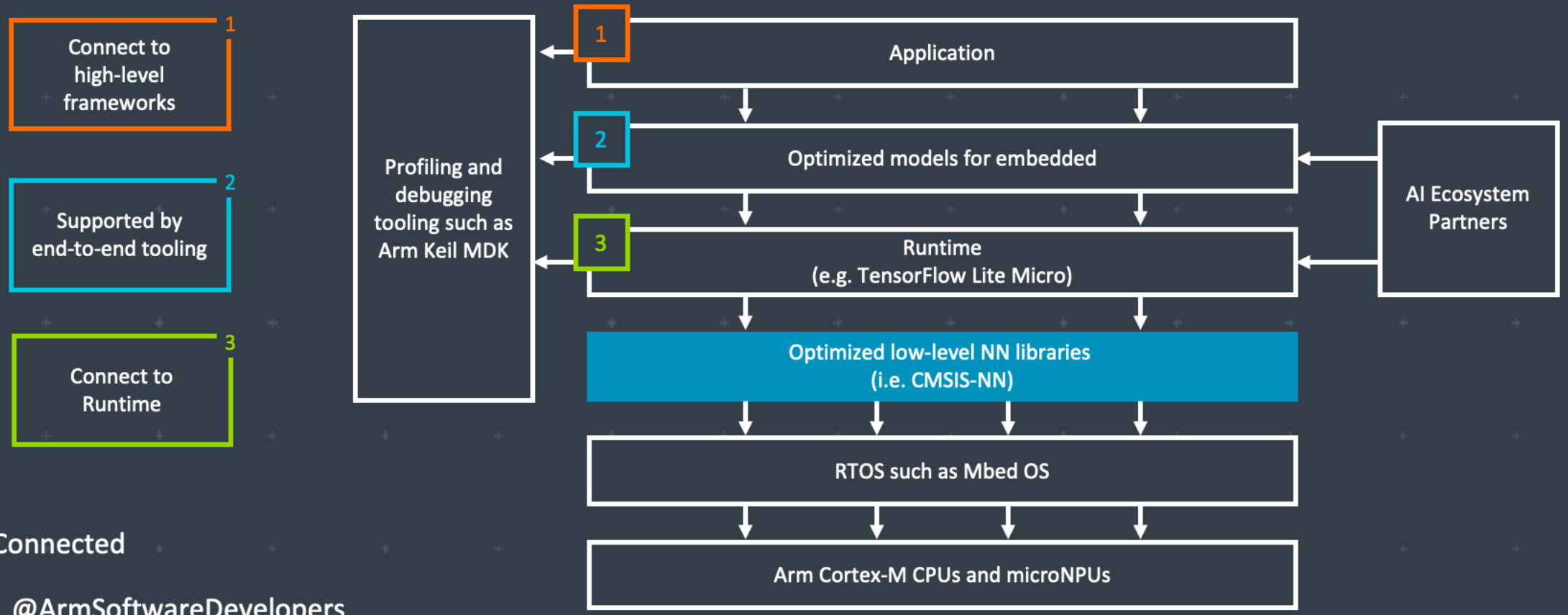
SensiML
tinyML Strategic Partner

SynSense
tinyML Strategic Partner

SYNTIANT
tinyML Strategic Partner

Additional Sponsorships available – contact Olga@tinyML.org for info

Arm: The Software and Hardware Foundation for tinyML



Stay Connected

@ArmSoftwareDevelopers

@ArmSoftwareDev

Resources: developer.arm.com/solutions/machine-learning-on-arm

arm

Deeplite

WE USE AI TO MAKE OTHER AI FASTER, SMALLER AND MORE POWER EFFICIENT

Automatically compress SOTA models like MobileNet to <200KB with **little to no drop in accuracy** for inference on resource-limited MCUs

Reduce model optimization trial & error from weeks to days using Deeplite's **design space exploration**

Deploy more models to your device without sacrificing performance or battery life with our **easy-to-use software**

BECOME BETA USER **bit.ly/testdeeplite**

mobilityXlab arm AI 100 CBINSIGHTS

TinyML for all developers

C++ library

Arduino library

WebAssembly

Dataset

Acquire valuable training data securely

Enrich data and train ML algorithms

Edge Device

Real sensors in real time

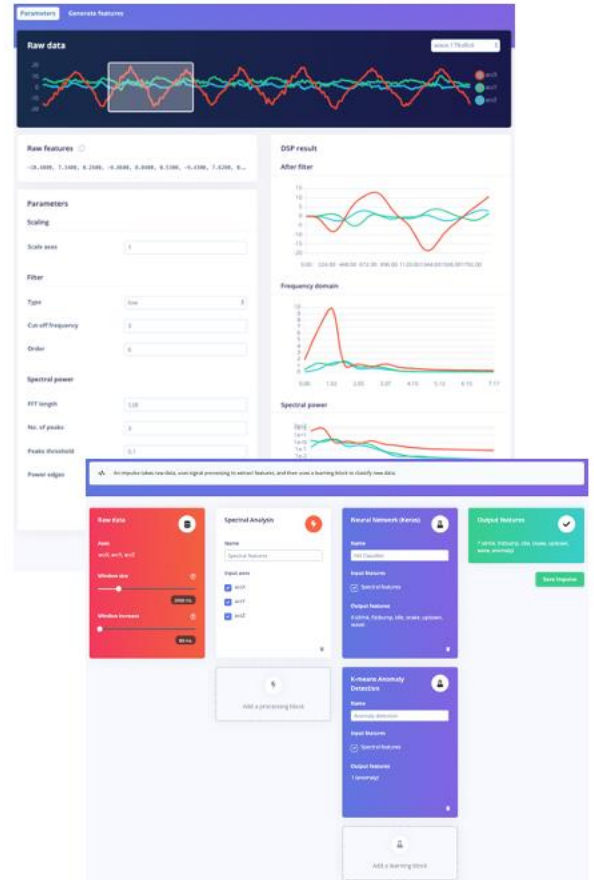
Open source SDK

Impulse

Embedded and edge compute deployment options

Test impulse with real-time device data flows

Test

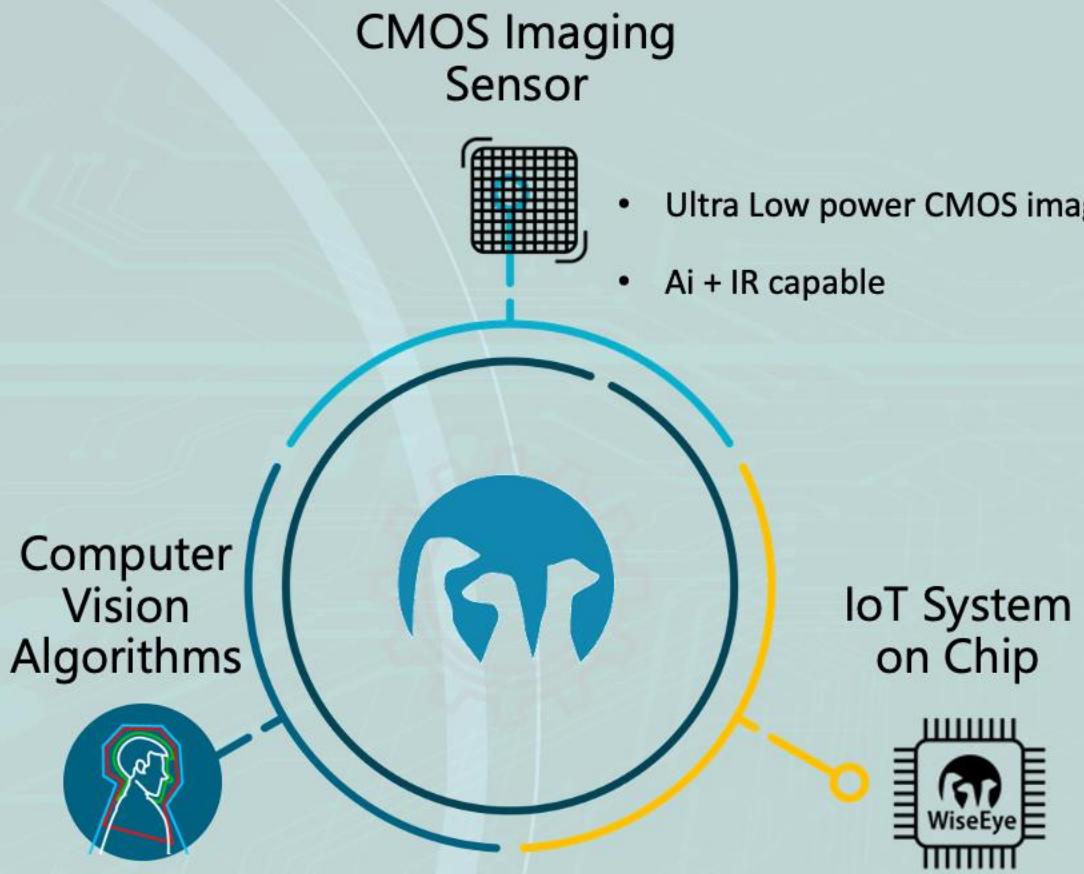


www.edgeimpulse.com

emza
visual sense

The Eye in IoT

Edge AI Visual Sensors



- Ultra Low power CMOS imager
- Ai + IR capable

- Machine Learning algorithm
- <1MB memory footprint
- Microcontrollers computing power
- Trained algorithm
- Processing of low-res images
- Human detection and other classifiers

- Machine Learning edge computing silicon
- <1mW always-on power consumption
- Computer Vision hardware accelerators

info@emza-vs.com

Enabling the next generation of Sensor and Hearable products to process rich data with energy efficiency

Visible Image

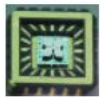
Sound

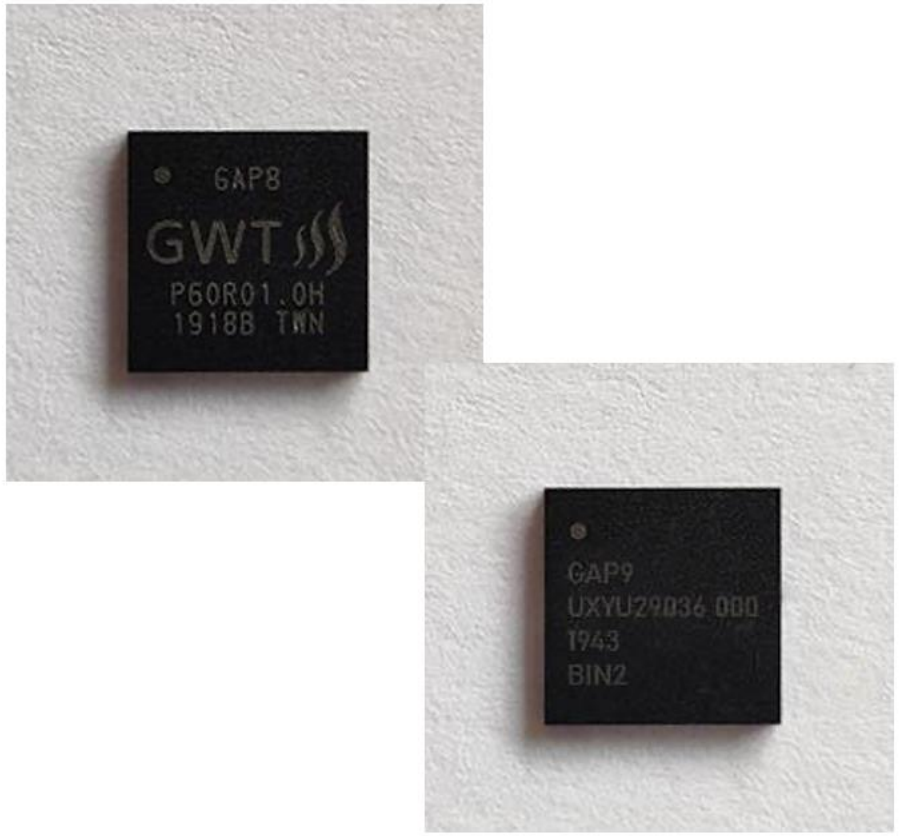
IR Image

Radar

Bio-sensor

Gyro/Accel

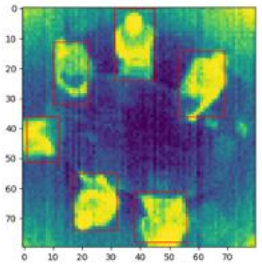




Wearables / Hearables

Battery-powered consumer electronics

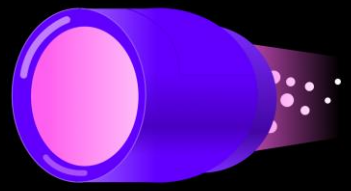
IoT Sensors



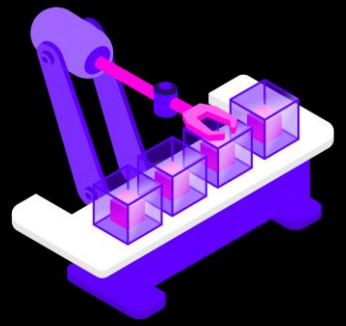
GREENWAVES TECHNOLOGIES

Distributed infrastructure for TinyML apps

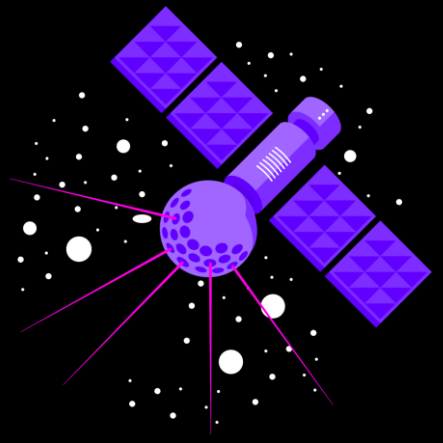
HOTG
Decoupling intelligence



Develop at warp speed



Automate deployments



Device orchestration

HOTG is building the distributed infrastructure to pave the way for AI enabled edge applications

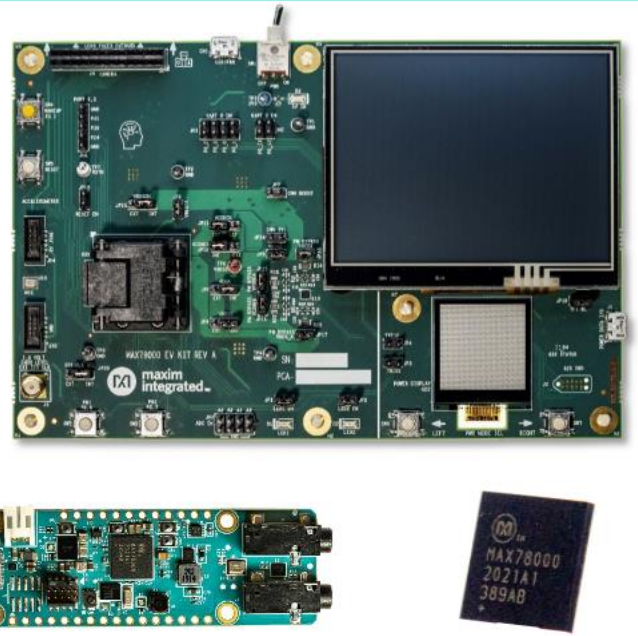
LatentAI

Adaptive AI for the Intelligent Edge

Latentai.com

Maxim Integrated: Enabling Edge Intelligence

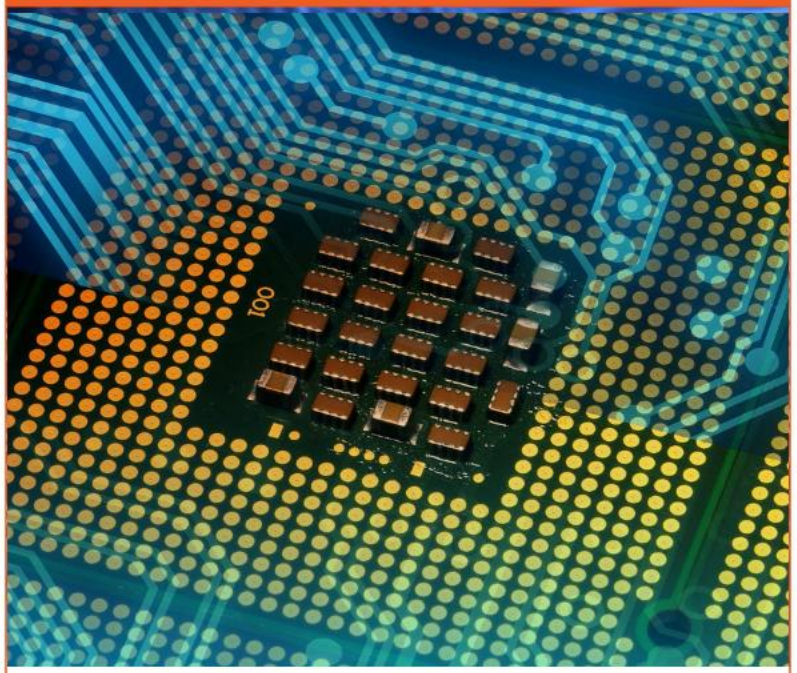
Advanced AI Acceleration IC



The new MAX78000 implements AI inferences at low energy levels, enabling complex audio and video inferencing to run on small batteries. Now the edge can see and hear like never before.

www.maximintegrated.com/MAX78000

Low Power Cortex M4 Micros



Large (3MB flash + 1MB SRAM) and small (256KB flash + 96KB SRAM, 1.6mm x 1.6mm) Cortex M4 microcontrollers enable algorithms and neural networks to run at wearable power levels.

www.maximintegrated.com/microcontrollers

Sensors and Signal Conditioning

Health sensors measure PPG and ECG signals critical to understanding vital signs. Signal chain products enable measuring even the most sensitive signals.

www.maximintegrated.com/sensors

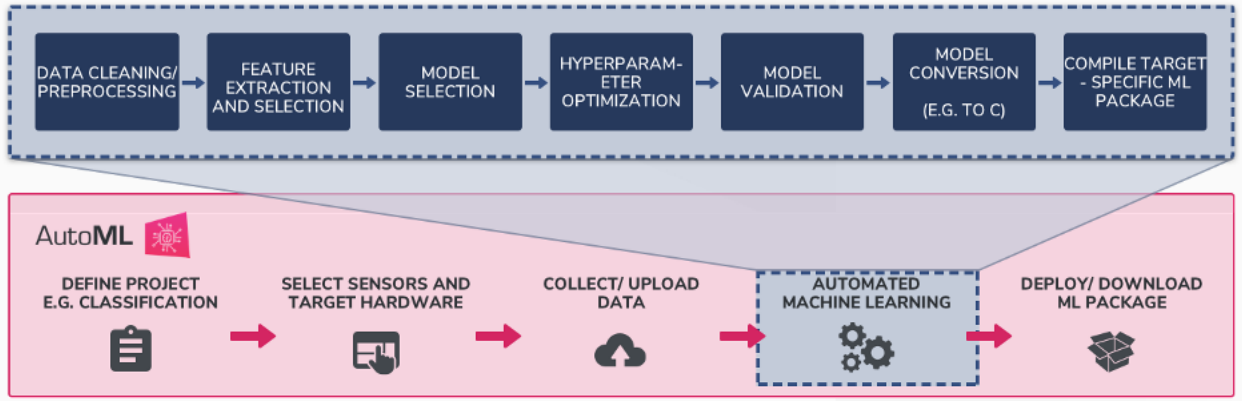
Qeexo AutoML

Automated Machine Learning Platform that builds tinyML solutions for the Edge using sensor data

Key Features

- Supports 17 ML methods:
  - Multi-class algorithms: GBM, XGBoost, Random Forest, Logistic Regression, Gaussian Naive Bayes, Decision Tree, Polynomial SVM, RBF SVM, SVM, CNN, RNN, CRNN, ANN
  - Single-class algorithms: Local Outlier Factor, One Class SVM, One Class Random Forest, Isolation Forest
- Labels, records, validates, and visualizes time-series sensor data
- On-device inference optimized for low latency, low power consumption, and small memory footprint applications
- Supports Arm® Cortex™- M0 to M4 class MCUs

End-to-End Machine Learning Platform



For more information, visit: www.qeexo.com

Target Markets/Applications

- Industrial Predictive Maintenance
- Smart Home
- Wearables
- Automotive
- Mobile
- IoT

Qualcomm
AI research

Advancing AI research to make efficient AI ubiquitous

Power efficiency

Model design, compression, quantization, algorithms, efficient hardware, software tool

Personalization

Continuous learning, contextual, always-on, privacy-preserved, distributed learning

Efficient learning

Robust learning through minimal data, unsupervised learning, on-device learning

A platform to scale AI across the industry

Perception

Object detection, speech recognition, contextual fusion

Reasoning

Scene understanding, language understanding, behavior prediction

Action

Reinforcement learning for decision making

Edge cloud

Cloud

IoT/IIoT

Automotive

Mobile

Reality AI®

Add Advanced Sensing to your Product with Edge AI / TinyML

https://reality.ai info@reality.ai @SensorAI Reality AI

Pre-built Edge AI sensing modules, plus tools to build your own

Reality AI solutions

Prebuilt sound recognition models for indoor and outdoor use cases

Solution for industrial anomaly detection

Pre-built automotive solution that lets cars "see with sound"

Reality AI Tools® software

Build prototypes, then turn them into real products

Explain ML models and relate the function to the physics

Optimize the hardware, including sensor selection and placement

SensiML™

Build Smart IoT Sensor Devices From Data

SensiML pioneered TinyML software tools that auto generate AI code for the intelligent edge.

- End-to-end AI workflow
- Multi-user auto-labeling of time-series data
- Code transparency and customization at each step in the pipeline

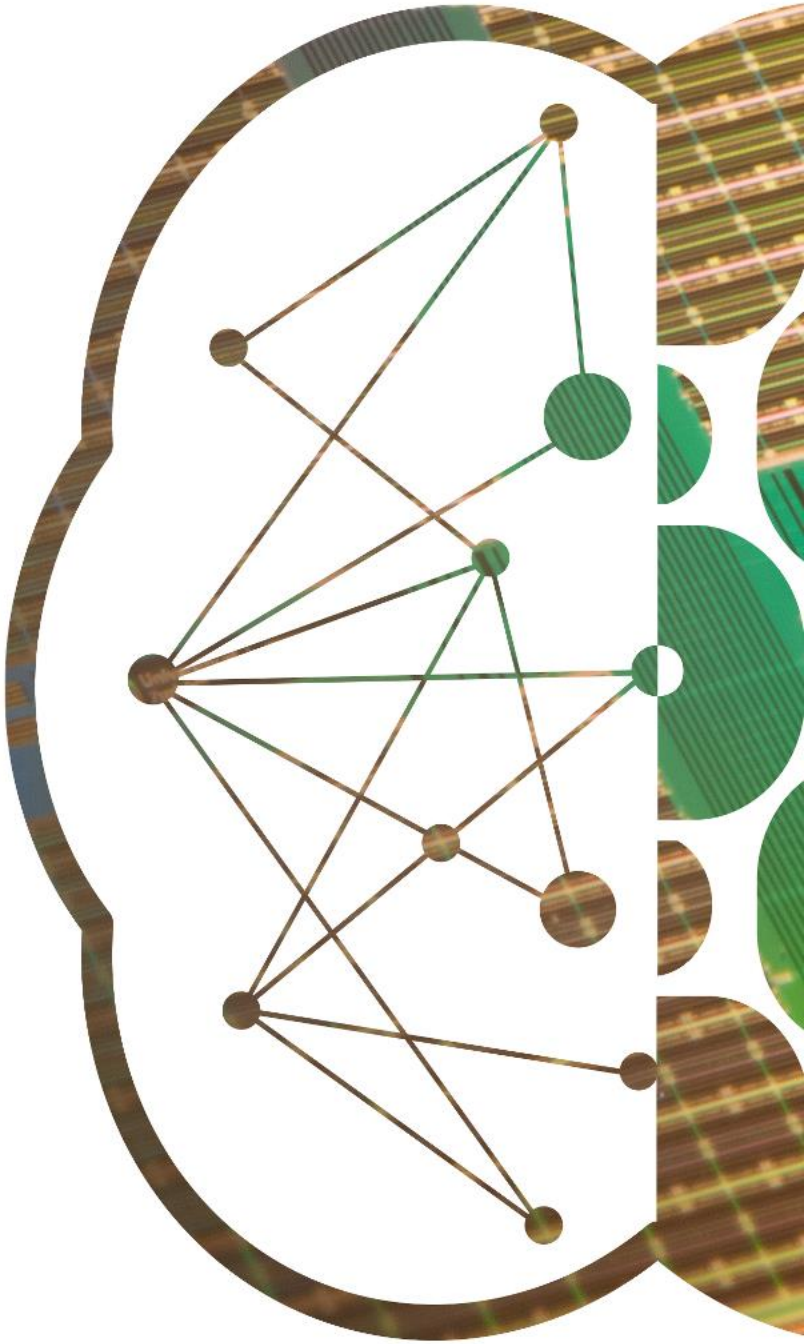
We enable the creation of production-grade smart sensor devices.

sensiml.com

SynSense

SynSense builds **sensing and inference** hardware for **ultra-low-power** (sub-mW) **embedded, mobile and edge** devices. We design systems for **real-time always-on smart sensing**, for audio, vision, IMUs, bio-signals and more.

https://SynSense.ai

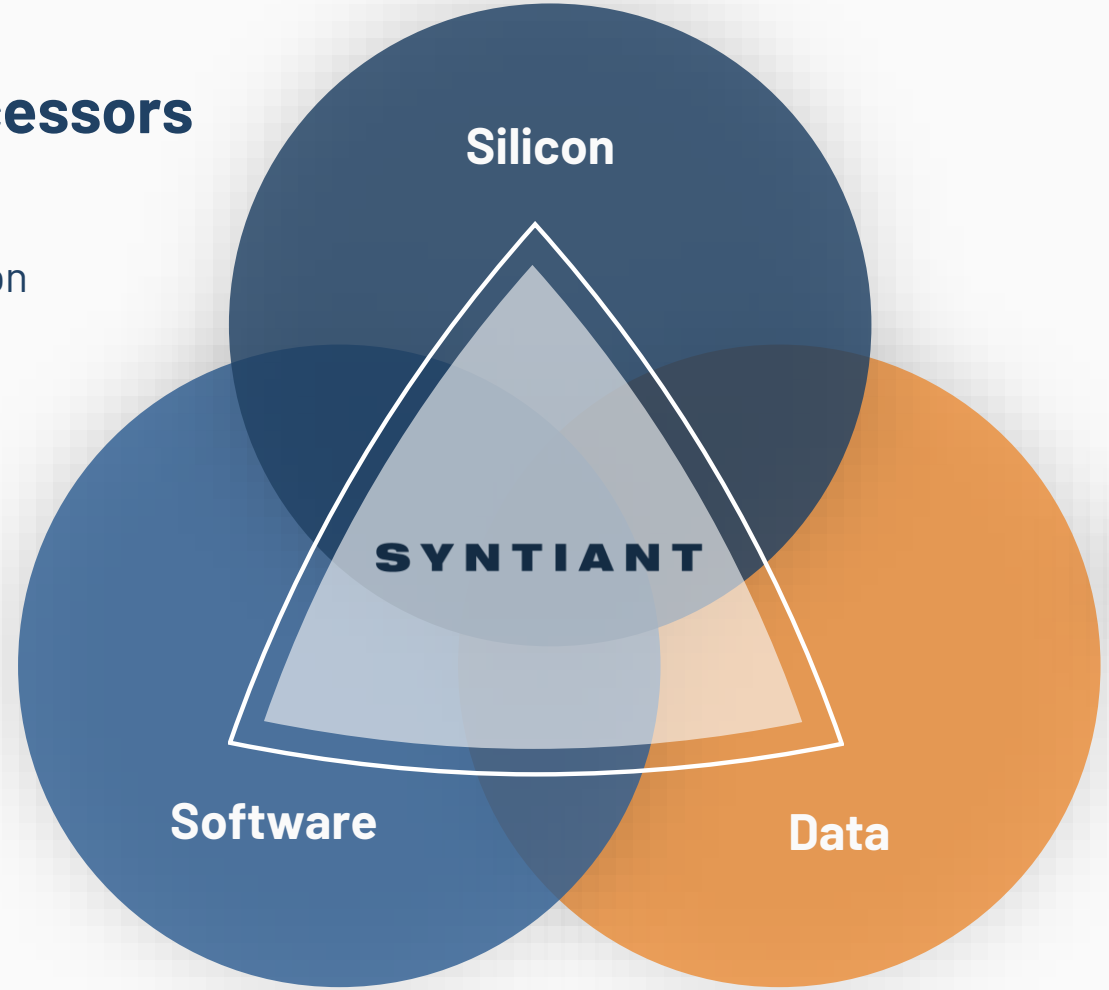


SYNTIANT

End-to-End Deep Learning Solutions for TinyML & Edge AI

Neural Decision Processors

- At-Memory Compute
- Sustained High MAC Utilization
- Native Neural Network Processing



ML Training Pipeline

- Enables Production Quality Deep Learning Deployments

Data Platform

- Reduces Data Collection Time and Cost
- Increases Model Performance

partners@syntiant.com

www.syntiant.com

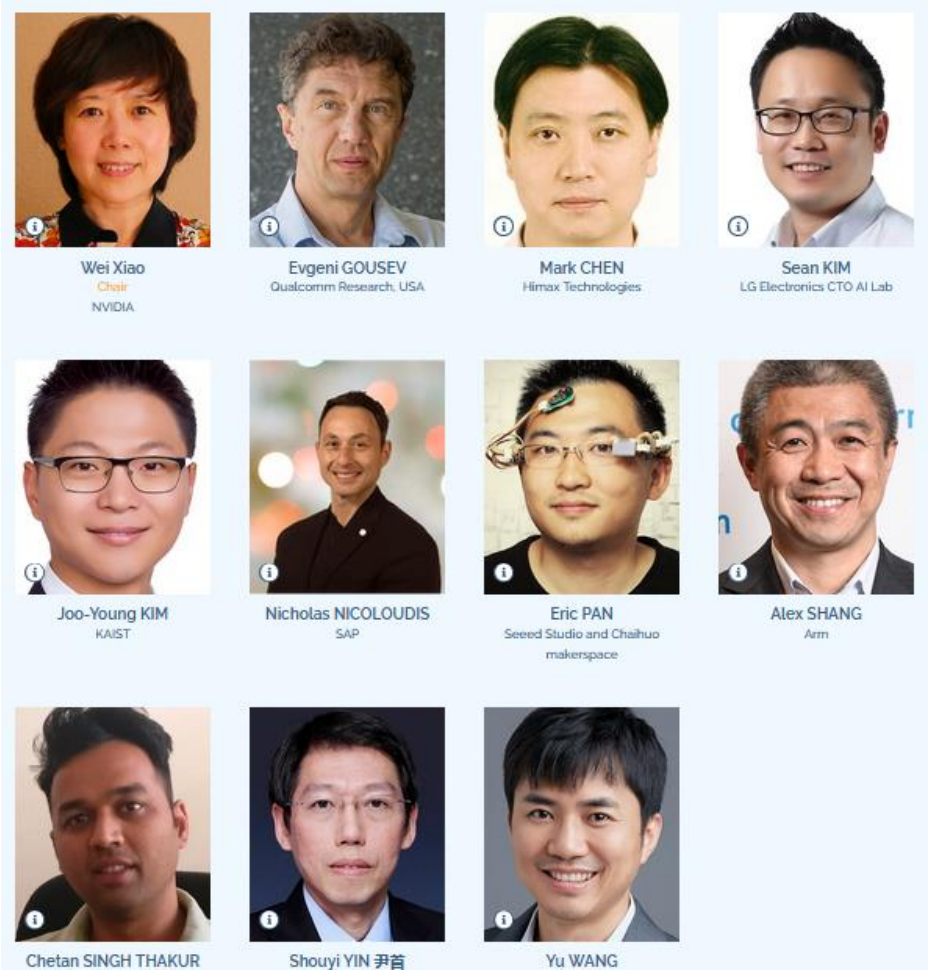
TINY ML ASIA

LIVE ONLINE **November 2-5, 2021**

(9-11:30 am China Standard time)

https://www.tinyml.org/event/asia-2021/

Technical Programm Committee



Register today!

Free event courtesy of our sponsors and strategic partners

arm · EDGE IMPULSE · emza visual sense · GREENWAVES TECHNOLOGIES

HOTG · imagimob · LatentAI Adaptive AI for a Smarter Edge · Qualcomm

Qeexo · RealityAI · seeed studio The IoT Hardware Enabler · SensiML

SynSense · SYNTIANT

More sponsorships are available: sponsorships@tinyML.org

collaboration with hackster.io

AN AVNET COMMUNITY

Focus on:
(i) developing new use cases/apps for tinyML vision; and (ii) promoting tinyML tech & companies in the developer community

Submissions accepted until September 17th, 2021

Winners announced on October 1st, 2021 ($6k value)

Sponsorships available: *sponsorships@tinyML.org*

https://www.hackster.io/contests/tinyml-vision

Next tinyML Talks

Date	Presenter	Topic / Title
Thursday, September 23	**Alexandre Valentian,** CEA LETI	How to design a power frugal hardware for AI - the bio-inspiration path

Webcast start time is 9 am Pacific time

Please contact talks@tinyml.org if you are interested in presenting

TinyML Meetup Germany

- Meetup Group: www.meetup.com/de-DE/tinyml-enabling-low-power-ml-at-the-edge-germany/

- If you like to propose a talk or other meetup event, feel free to get in touch with us: daniel.mueller@tum.de, carloshvp@gmail.com, Marcus.Rueb@hahn-schickard.de

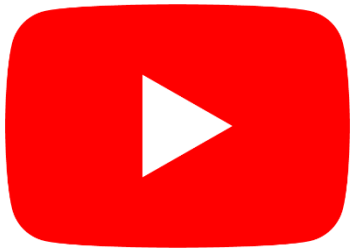
TinyML Meetup Germany Organizers:
Carlos Hernandez-Vaquero (Bosch)
Daniel Mueller-Gritschneder (TU Munich)
Alexis Veynachter (Infineon)
Marcus Rüb (Hahn-Schickard)

TINY ML TALKS webcast

Reminders

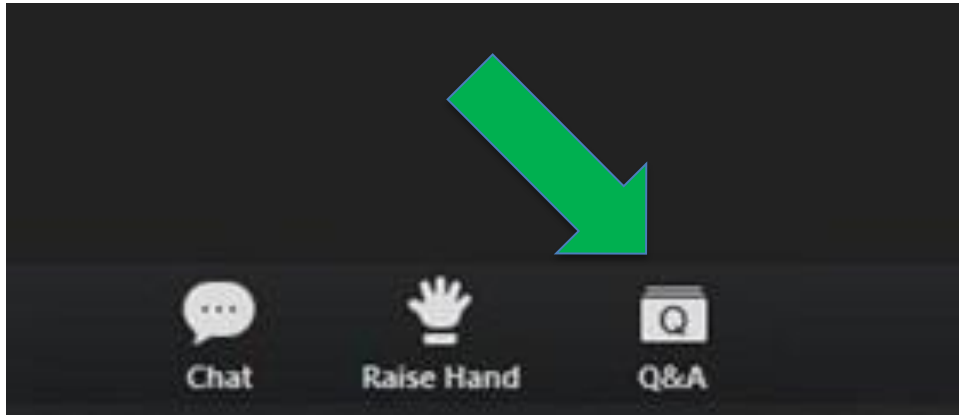
Slides & Videos will be posted tomorrow

tinyml.org/forums



youtube.com/tinyml

Please use the Q&A window for your questions



TINY ML TALKS webcast

Paul Palomero Bernardo

Paul Palomero Bernardo was born in Tübingen, Germany, 1996. He received the B.S. and M.S. degrees in computer science from University of Tübingen, Tübingen, Germany, in 2017 and 2020, respectively, where he is currently pursuing the doctoral degree (Ph.D.) at the Department of Computer Science. His current research interests include neural network hardware and design optimization.

Structure

- **ML accelerator UltraTrail**
  - Architecture
  - Deployment
  - Results
  - SoC platform integration

- **Hardware architecture and neural network search (HANNAH)**
  - Framework overview
  - Search algorithm
  - Results

- **Automatic hardware generation**
  - Motivation
  - TVM-based hardware generation

UltraTrail – Overview

UltraTrail: A Configurable Ultralow-Power TC-ResNet AI Accelerator[1]

- **Architecture:**
  - ASIC accelerator for DNN inference
  - Scalable design for application-specific adaptability

- **Target DNN architecture:**
  - Temporal convolutional neural network (TC-ResNet)
  - 1-dimensional convolution along the temporal dimension

- **Use-cases:**
  - Near-sensor signal processing
  - Keyword spotting, wakeword detection, voice activity detection

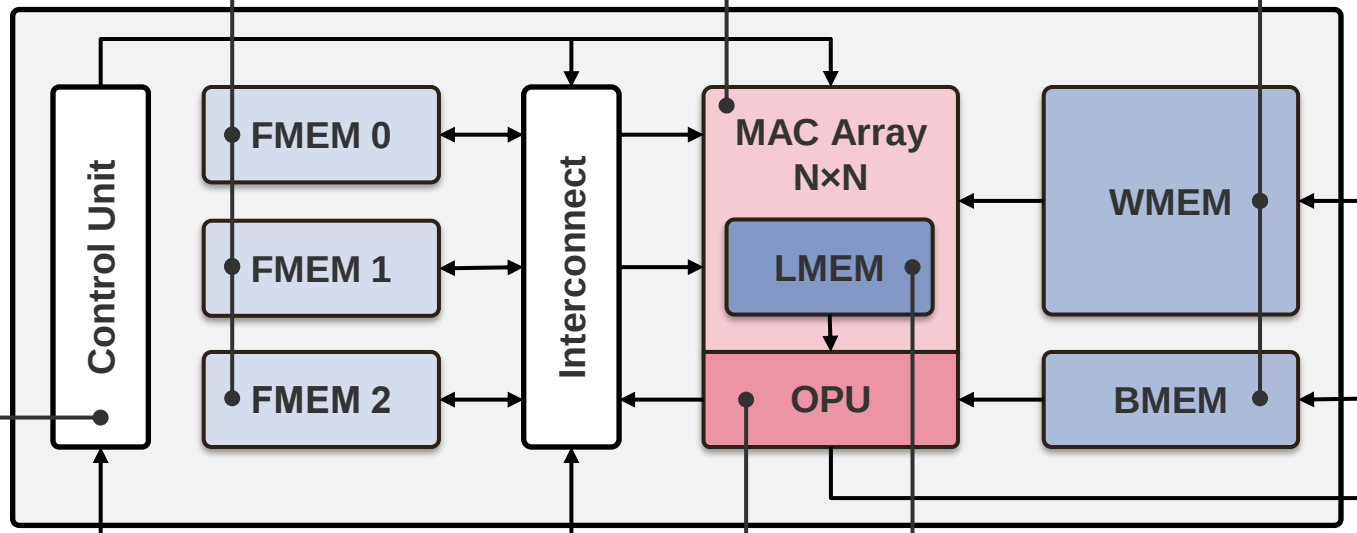
[1] Paul Palomero Bernardo, Christoph Gerum, Adrian Frischknecht, Konstantin Lübeck, Oliver Bringmann: "UltraTrail: A Configurable Ultralow-Power TC-ResNet AI Accelerator for Efficient Keyword Spotting", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 39.11 (2020): 4240-4251.

UltraTrail – Architecture

MAC Array
Combinational array of N×N multiply-and-accumulate units

Feature Memories
Distributed architecture for input features, output features, and residual paths

Weight & Bias Memory
All DNN parameters stored on-chip



Control Unit
Programmable control unit for stand-alone operation

Local Memory
Stores partial sums for accumulation inside the MAC array

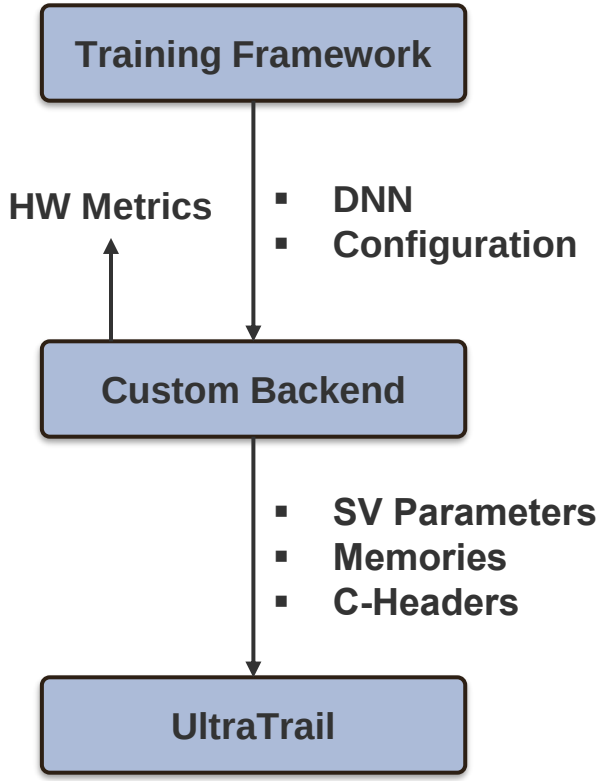
Output processing unit
Bias, Activation, Pooling, Conditional Computing

UltraTrail – Deployment

UltraTrail provides a custom backend for configuration and deployment

- **Custom backend:**
  - Manages data layout transformation
  - Generates fitting memory macros
  - Generates SystemVerilog parameters for design configuration
  - Provides hardware models for power, performance, and area

- **Wrapper interface for PULP-based SoCs:**
  - Hardware Processing Engine (HWPE) interface protocol
  - Manages off-chip memory access and accelerator configuration
  - Manages clock and power gating of the accelerator
  - Support for clock domain crossing



UltraTrail – Results

- **Evaluation on the task of real-time keyword spotting**

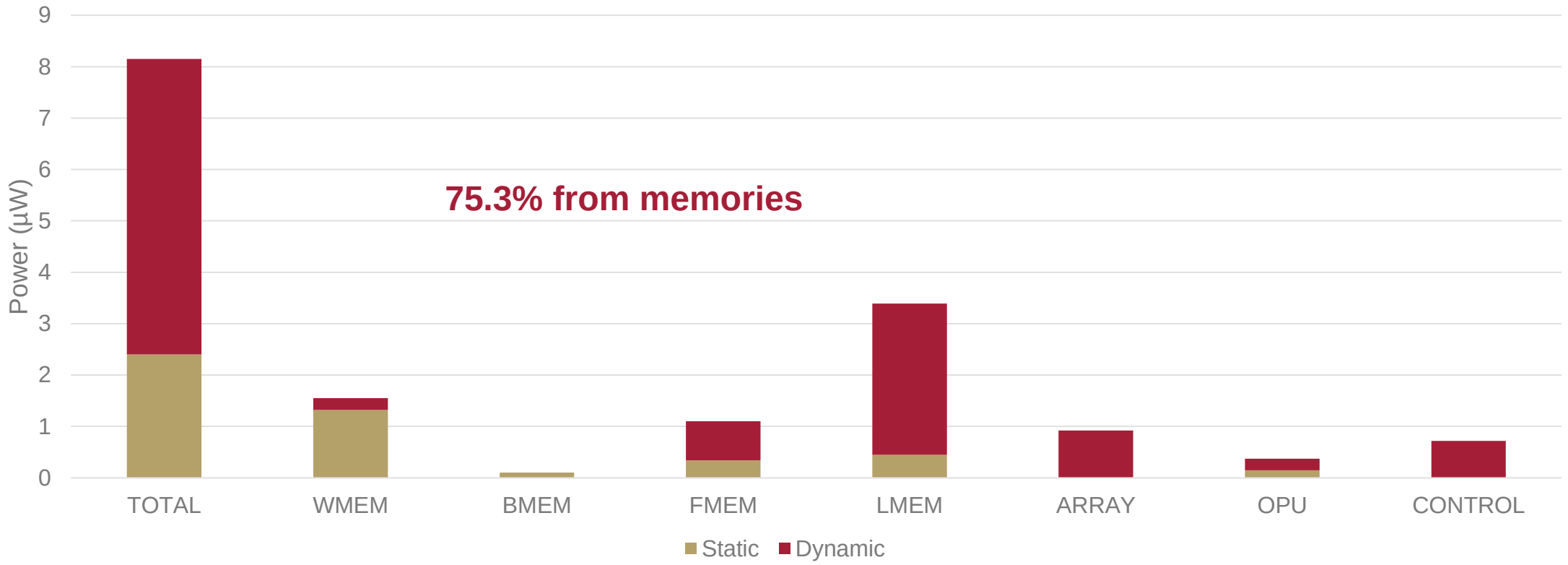
- **DNN architecture:**
  - TC-ResNet
  - 64k parameters, 1.5M MACs
  - 6 bit weights, 8 bit features

- **Requirements:**
  - Always-on → optimize for low power consumption
  - Real-time → 10 inferences per second

- **Methodology:**
  - Results are based on post-layout simulations of 20 consecutive inferences
  - Evaluated for the 22 nm technology 22FDX at typical operation conditions (25°C TT)

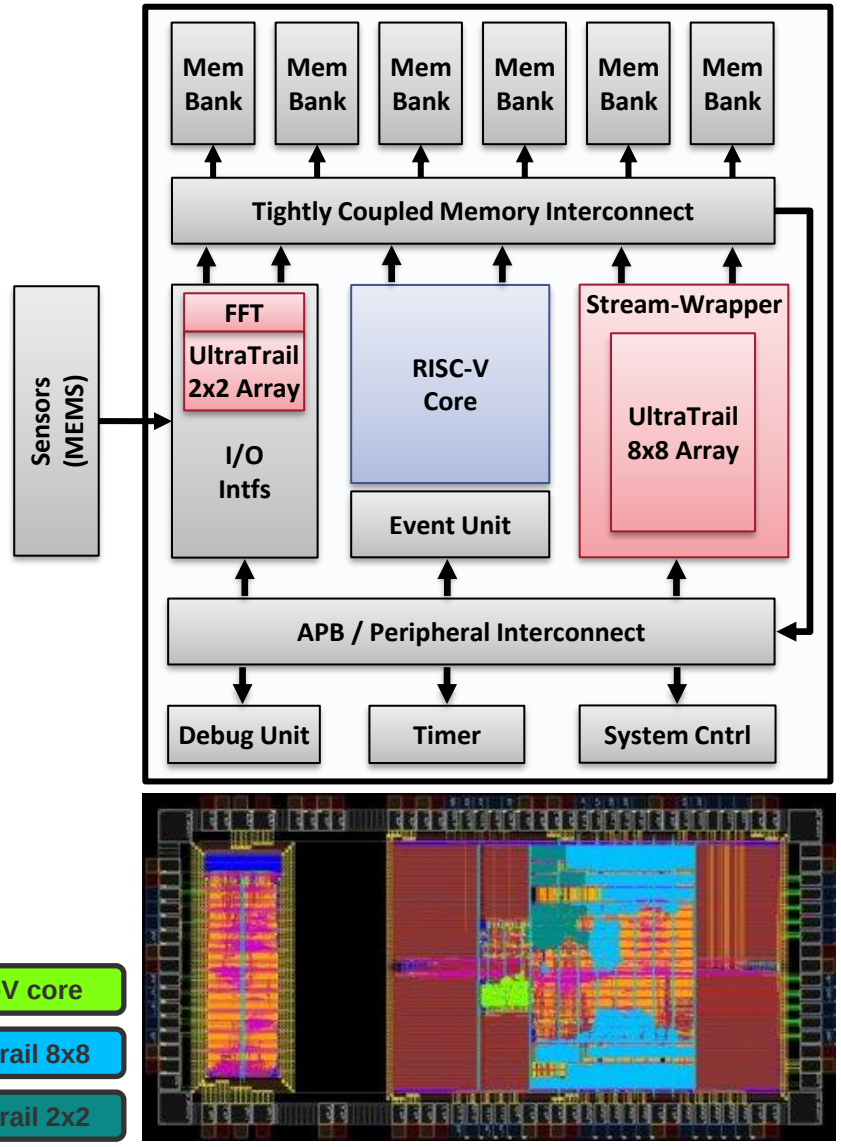
UltraTrail – Results

Technology	Area	Frequency	Latency	Voltage	Word Width (Weights)	Word Width (Features)	Accuracy	Keywords	Power
22 nm	0.2 mm²	250 kHz	100 ms	0.8 V	6	8	93.09 %	10	8.2 µW

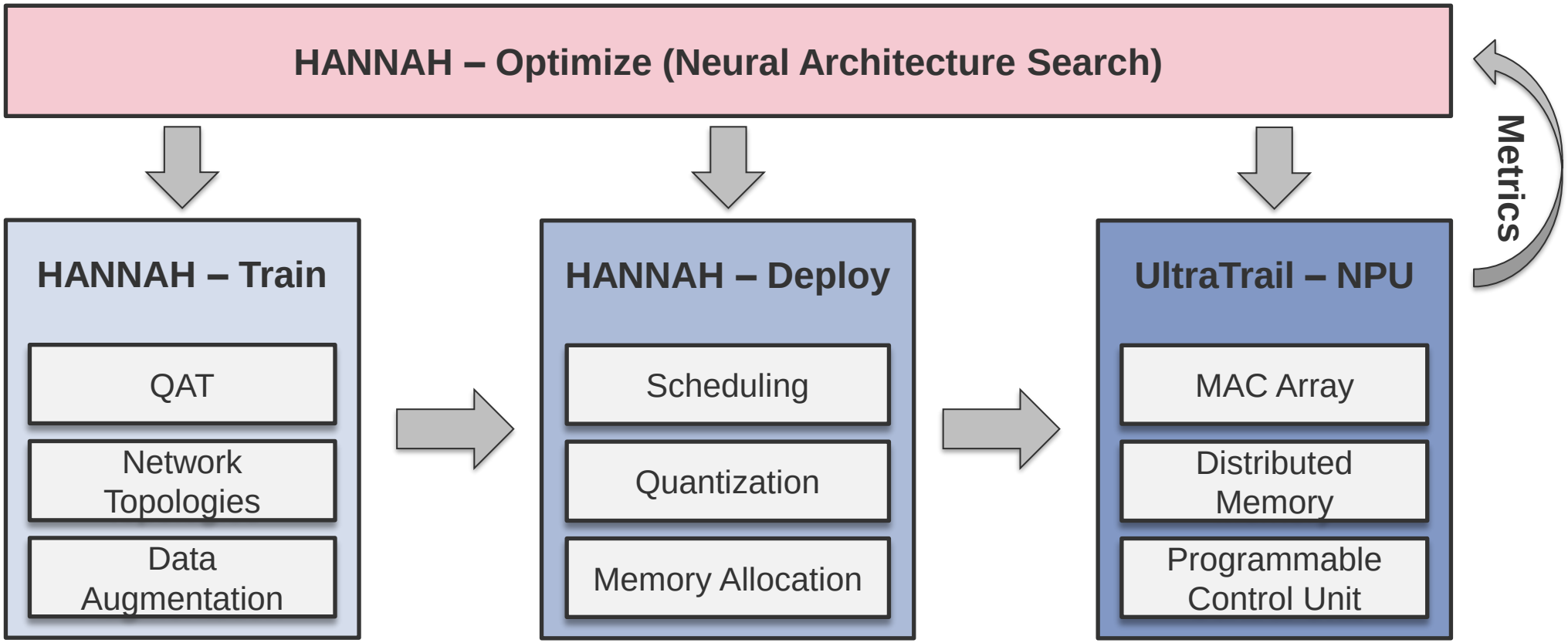


Tübingen Edge Computing AI Platform T-Rax

- **Low-power edge computing platform for intelligent sensor signal processing**
- **SoC platform T-Rax (based on PULPissimo)**
  - Chip in 22FDX, RTL and VP available
  - Tape-out: 1.56 mm², 200 MHz @ 0.8V, 7.5 mW
- **RISC-V CPU core**
  - RV32IMC
  - Used for booting and accelerator setup
- **Customizable AI accelerator UltraTrail**
  - Configurable neural network execution
  - 2x2 UltraTrail for voice activity detection (~500 nW)
  - 8x8 UltraTrail for keyword spotting (8.2 µW)
  - Hierarchical wakeup mechanism



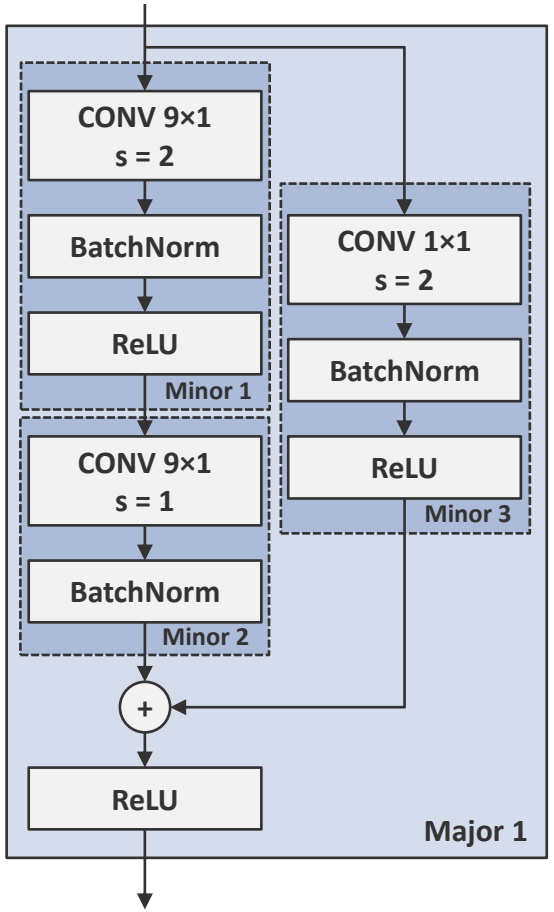
HANNAH – Overview



HANNAH – Search

Input: Design Space S, Search Budget b, Population Size s, Bounds B, Training Set T, Validation Set V
Output: Search History H

```
/* Start with empty search history                          */
H ← ∅
for n ∈ {1 ... b} do
    if b ≤ s then
        /* Sample Random Architecture and Accelerator       */
        a ← sample_random(S)
    else
        /* Sample Fitness Function from target bounds       */
        f ← sample_fitness_function(B)
        /* Sort last p candidates according to Fitness      */
        P ← sort(H[−s :], f)
        /* Apply random mutation to current best architecture */
        a ← mutate_random(last(P))
    /* Quantization Aware Training                          */
    N ← train(a, T)
    /* Evaluate trained architecture                        */
    error_rate ← evaluate(N, V)
    /* Estimate other metrics from hardware model           */
    power, latency, area ← hardware_model(a)
        /* Append architecture and metrics to history       */
    H.append(a, (error_rate, power, latency, area))
```



Major Block 1
- output channels = 24
- stride = 2
- branch = "residual"
- Minor Blocks: (see below)

Minor Block 1
- size = 9
- padding = true
- batchnorm = true
- activation = true
- parallel = false

Minor Block 2
- size = 9
- padding = true
- batchnorm = true
- activation = false
- parallel = false

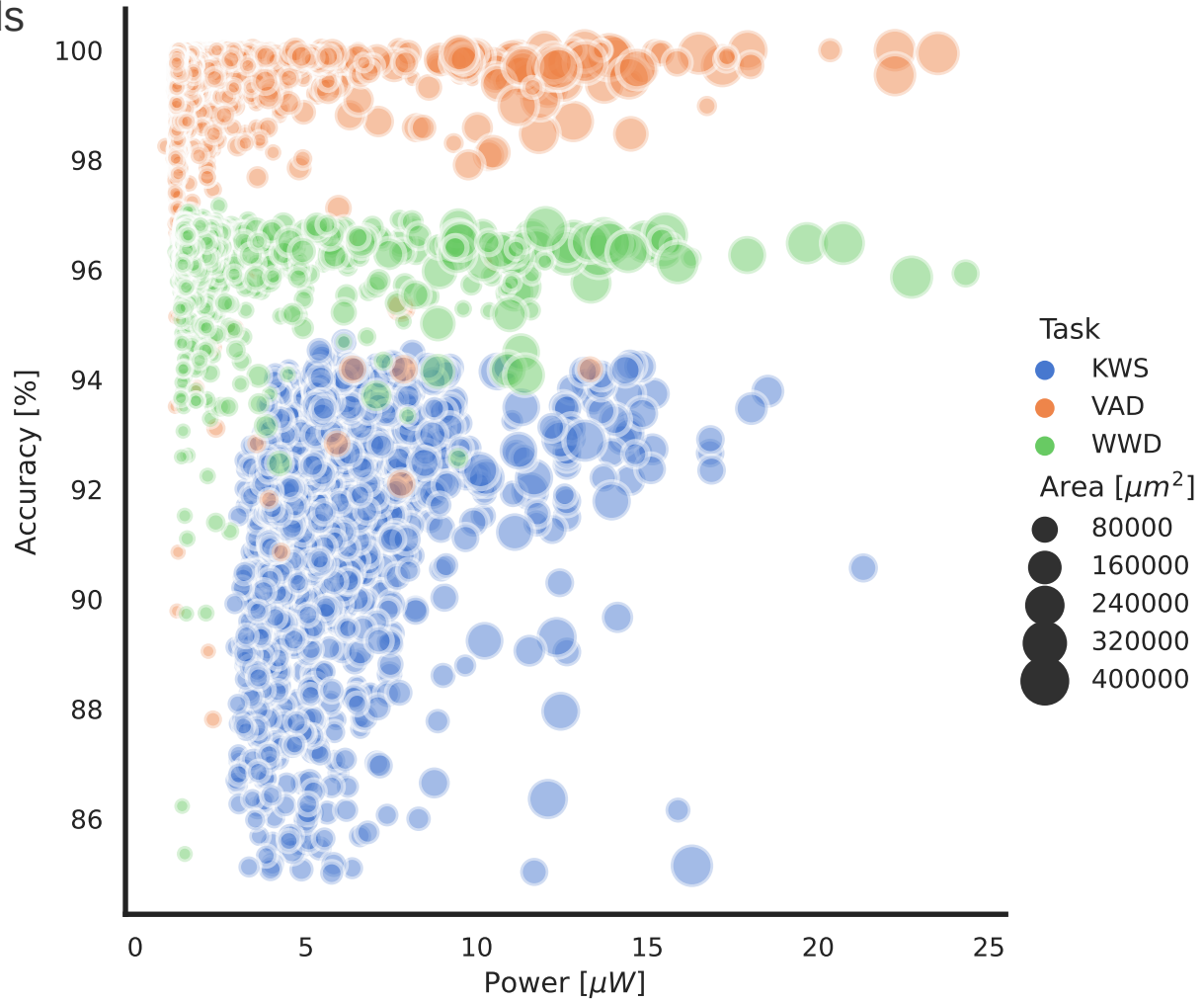
Minor Block 3
- size = 1
- padding = true
- batchnorm = true
- activation = true
- parallel = true

HANNAH – Results

- **KWS:** Keyword Detection on Google Speech Commands
- **VAD:** Voice Activity Detection on UWNU/TUT
- **WWD:** WakeWord Detection on Hey Snips!

→ Search Time: ~ 2 days @4 GPUs

Level	Option	Choices
Network	Word Width Features	4,6,8
Network	Word Width Weights	2,4,6,8
Network	Number of Blocks	1-4
Block	Stride of Blocks	1,2,4,8,16
Block	Type	residual, forward
Block	Number of Convs	1-4
Layer	Kernel Size	1,3,5,7,9,11
Layer	Output Channels	4,8,..,64
Layer	Activation Function	ReLU, None
Accelerator	Array Size	2×2, 4×4, 8×8, 16×16



HANNAH – Results

Design	Area	DNN Structure	Word Width (Weights)	Word Width (Features)	MAC Array Size	Accuracy	Power
Manual	0.20 mm²	TC-ResNet	6	8	8×8	93.09 %	8.20 µW
HANNAH High Accuracy	0.13 mm²	TC-ResNet	6	6	8×8	94.33 %	6.38 µW
HANNAH Low Power	0.09 mm²	CONV+FC	6	6	6×6	93.37 %	3.79 µW

- **1.24 pp increase in accuracy**
- **Over 2x reduction in power**

Current State

- No continuous design flow
- Separate optimization

Automatic Hardware Generation

Current State

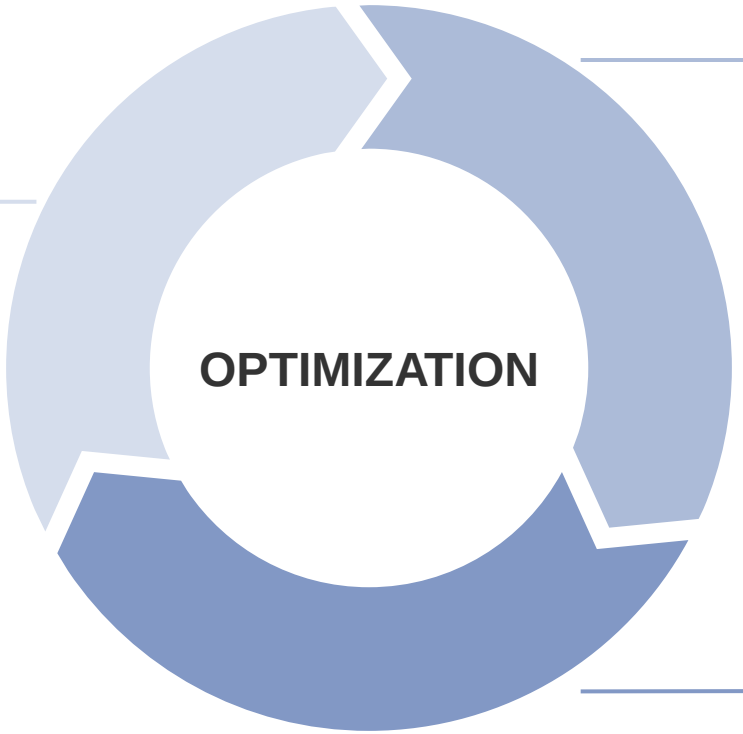
- No continuous design flow
- Separate optimization

Goal

- End-to-end hardware generation
- Joint optimization

TVM-based Hardware Generation

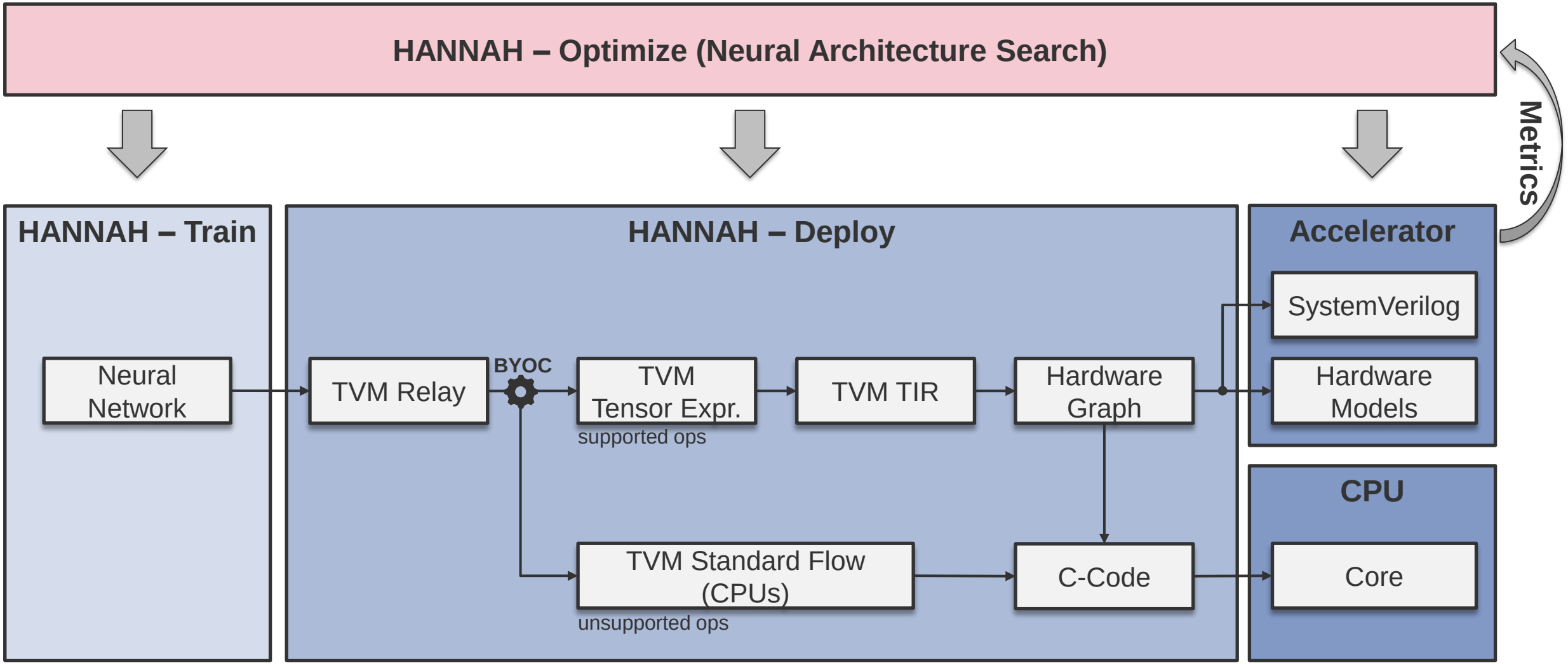
DNN
- Network topology
- Training
- Quantization
- Pruning



Deployment
- Mixed deployment (BYOC)
- Hardware abstraction
- Memory planning
- Code generation

Accelerator
- RTL
- Hardware models
- Driver

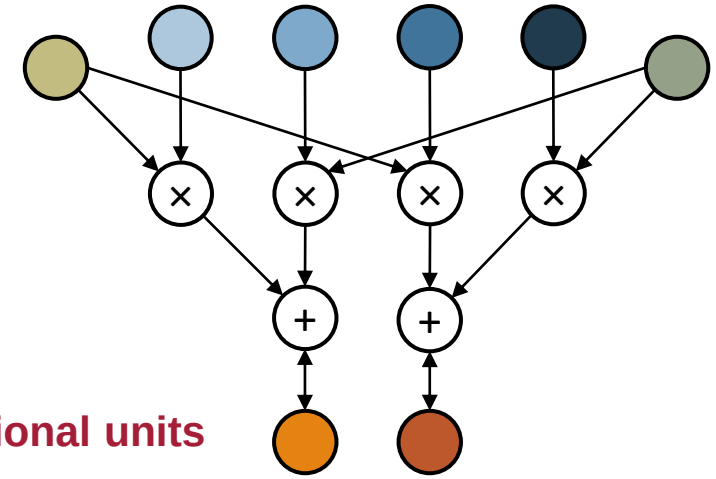
TVM-based Hardware Generation



TVM-based Hardware Generation

```
for g1=0:1 do
    for k1=0:12 do
        for x1=0:12 do
            for c1=0:8 do
                for f_x1=0:9 do
                    o[24·k1+x1]    += i[40·c1+x1+f_x1]    · w[288·k1+18·c1+f_x1]
                    o[24·k1+x1]    += i[40·c1+x1+f_x1+20] · w[288·k1+18·c1+f_x1+9]
                    o[24·k1+x1+12] += i[40·c1+x1+f_x1]    · w[288·k1+18·c1+f_x1+144]
                    o[24·k1+x1+12] += i[40·c1+x1+f_x1+20] · w[288·k1+18·c1+f_x1+153]
```

To hardware graph



Hardware

- loop control
- load
- multiplier
- adder
- load / store

- **Use templates for memories and functional units**
- **Use code generation for interconnects**

Conclusion

- **Summary:**
  - Scalable ML accelerator UltraTrail for extreme-edge applications
  - Joint hardware/software co-design using HANNAH

- **Challenges:**
  - Manual design and extension of ML accelerators is very time consuming and requires expert knowledge
  - Changes to the hardware also translate to the deployment and training process

- **Solutions:**
  - End-to-end design flow including training, deployment, and hardware generation
  - Global optimization over joint search space

Copyright Notice

www.tinyml.org

Copyright Notice

www.tinyML.org