Fraunhofer IIS

Fraunhofer-Institut für Integrierte Schaltungen IIS

Design-Time Optimization of Deep Neural Networks for Intermittent Learning on Microcontrollers

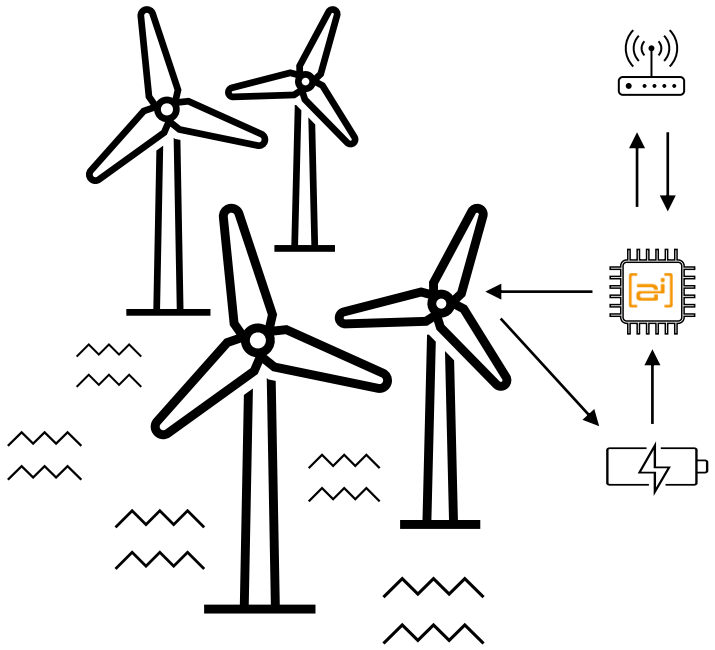
Jakob Schubert, Maximilian Kasper, Maximilian Linke, Benedict Herzog, **Mark Deutel**, Axel Plinge, Dominik Seuss and Christopher Mutschler

ITEM Workshop 2026 | 11.09.2026

Motivation

Learnable Systems in Energy Intermittent Environments

adaptation to slightly changing environment
requires adaptation of the AI model

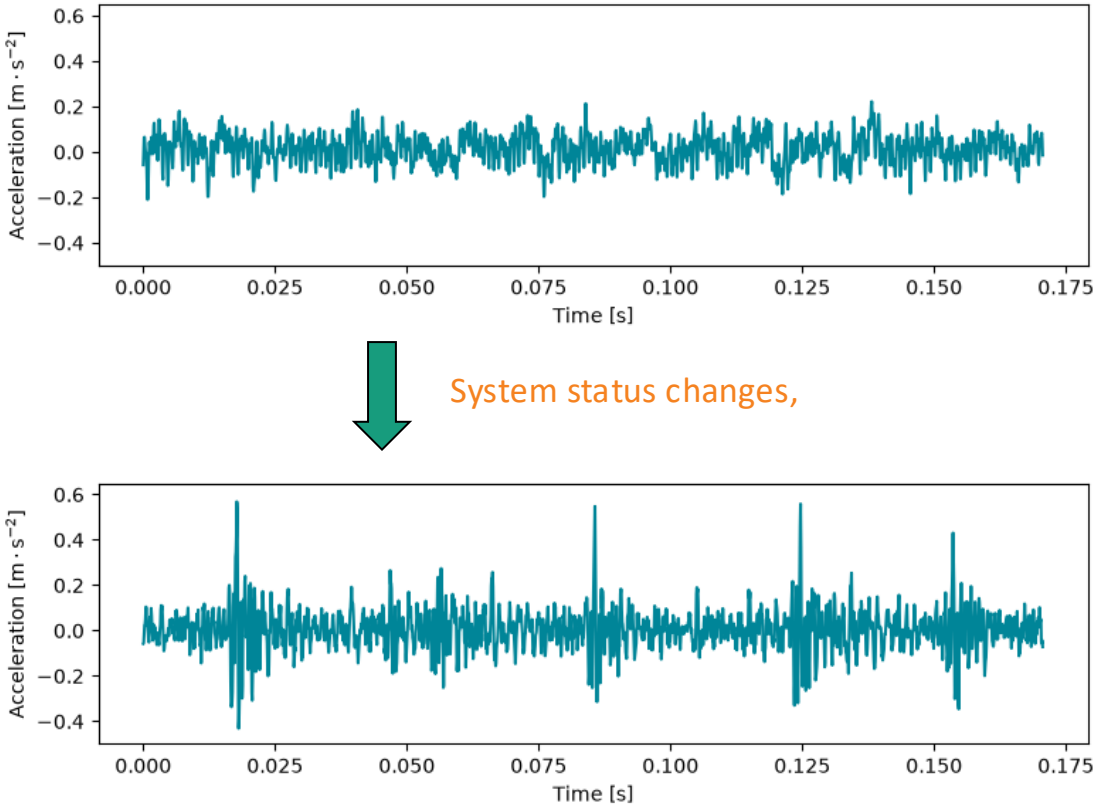


narrowband communication
→ communicate system status

broadband communication
for raw data streaming and
model update

on-device inference
+ on-device training

energy harvesting
e.g., vibration harvesting
→ energy intermittency

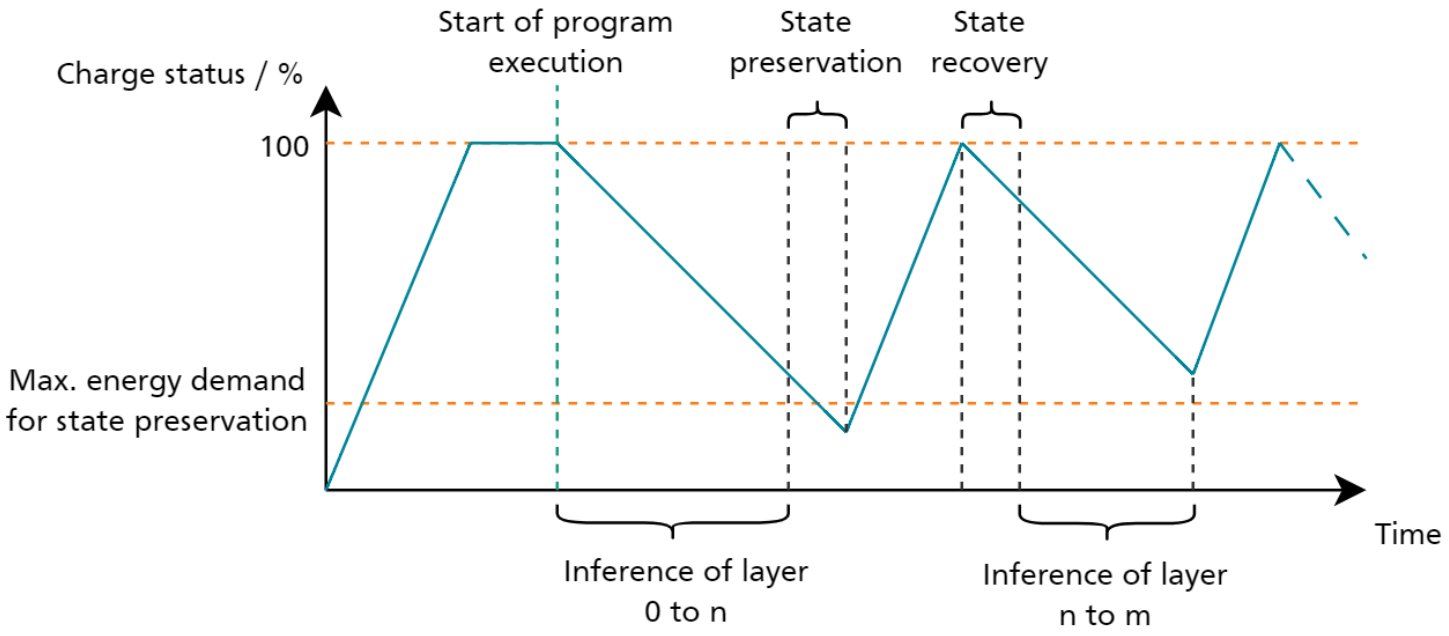


System status changes,

Design of energy intermittency aware trainable deep neural networks

Motivation

On-Device Training with Energy Intermittency



Intermittent Computing

- System operates entirely on **harvested energy** from the environment (e.g., solar, RF, or kinetic).
- Interruption of operation can occur due to **unpredictable power failures**.
- State preservation across power cycles using **Non-Volatile Memory (NVM)** is mandatory.
- Usage of system checkpoints (points from which computation can continue) to prevent corrupt or repeated states.

Intermittent Learning

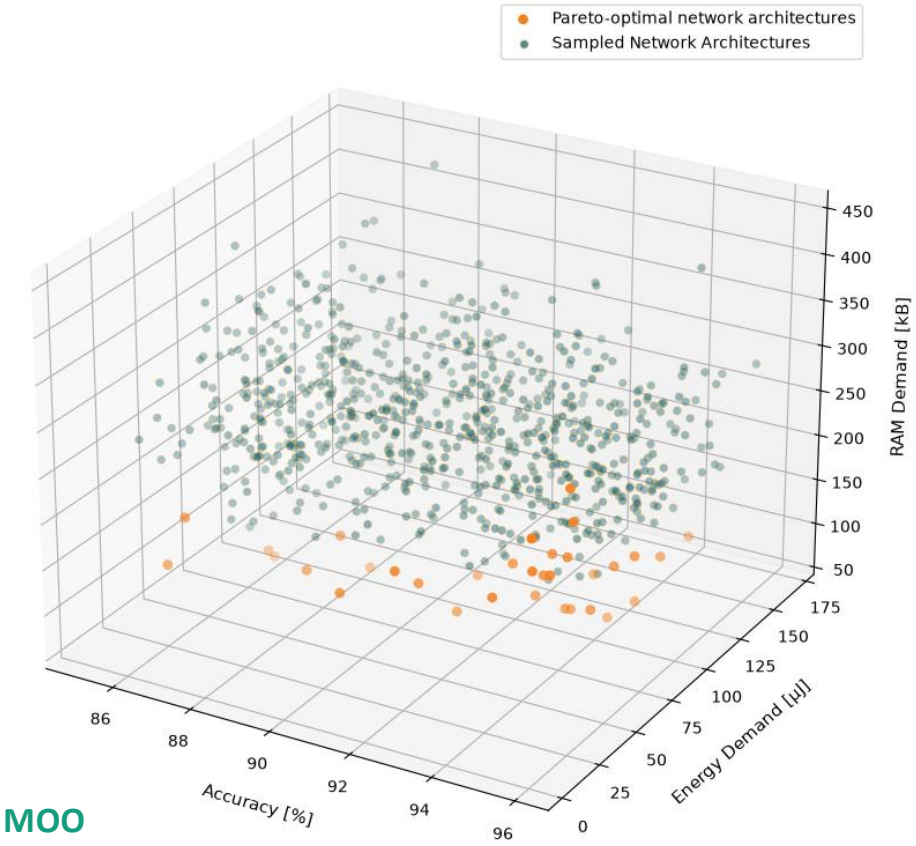
- Add **checkpoints between DNN layers** in the forward and backward pass of the entire network for inference and training.
  - Save **weights, activations and gradients** onto NVM for state preservation
  - Complete forward and backward pass does not have to fit into a single power cycle.

Concept and Purpose of this Work

Design-Time Optimization of Intermittent Learning Aware Networks

Multi-Objective Optimization (MOO) of Deep Neural Network Architectures

- Automated design of DNN architectures that satisfy multiple objectives e.g., accuracy, RAM, ROM, or energy.
  - Goal is to identify Pareto-optimal (or non-dominated) architectures
- Most objectives are **directly inferable** on the remote **development system**
  - Accuracy by model evaluation on test dataset and memory requirements after model conversion.
- **Energy consumption** is **not directly inferable**
  - Deployment on target system and online energy measurements are slow and error-prone



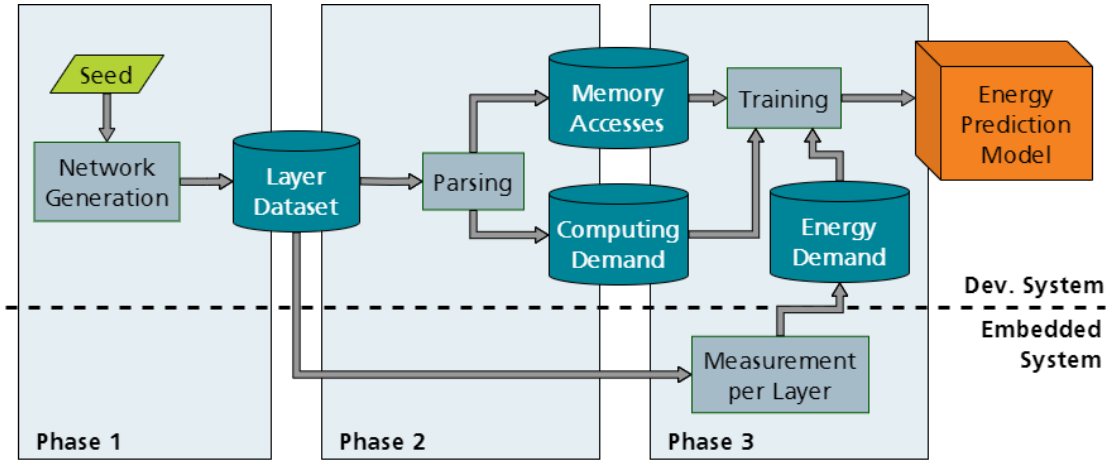
Objective of this work: An accurate and fast to query energy prediction model for MOO to enable efficient intermittent DNN learning

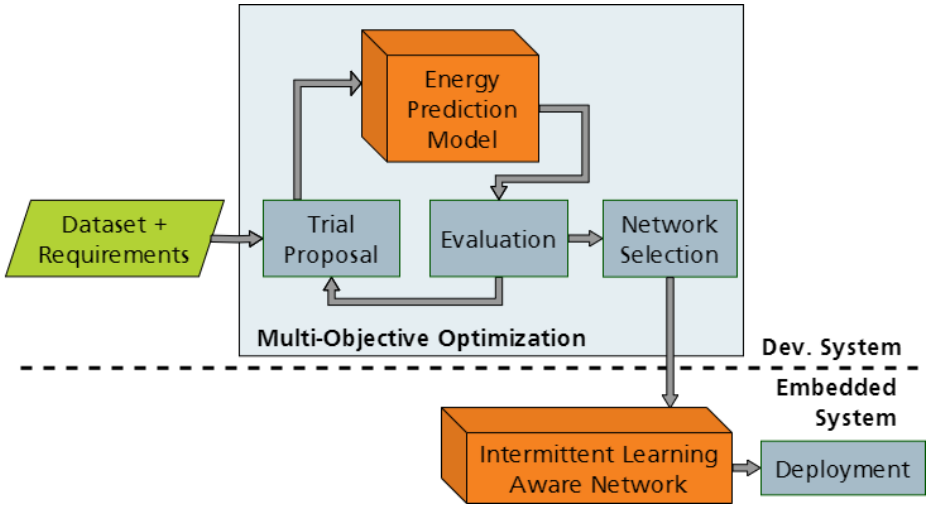
Concept and Purpose of this Work

Overview

Energy consumption of DNN layers is dependent on both implementation and hardware specific aspects:

- FPU?, SIMD instructions?, DSP extension?





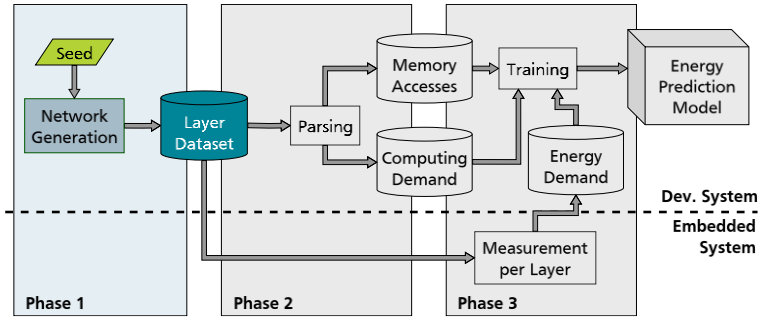
One-time development of an energy prediction model

Multi-Objective Optimization

Short development cycles for energy aware, trainable DNNs in energy intermittent environments.

Energy Prediction Model Development

Phase 1: Layer Dataset Sampling



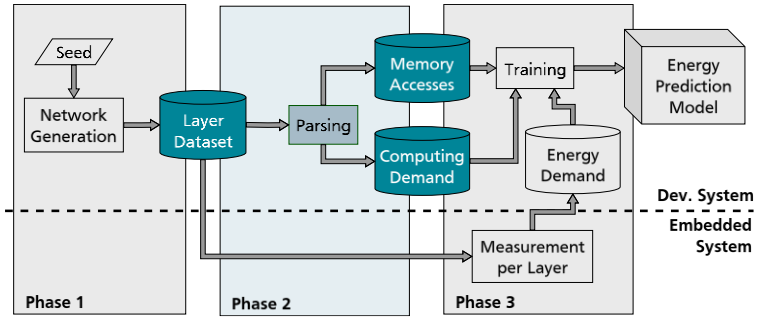
Dataset generation by sampling of network architectures

- Random **sampling** of layer configurations to build valid (quantized) DNN architectures

BatchNormalization	QGemm	QuantizeLinear
DequantizeLinear	QLinearAdd	Relu
Flatten	QLinearAveragePool	Resize
MaxPool	QLinearConv	…

Concept and Purpose of this Work

Phase 2: Model Parsing

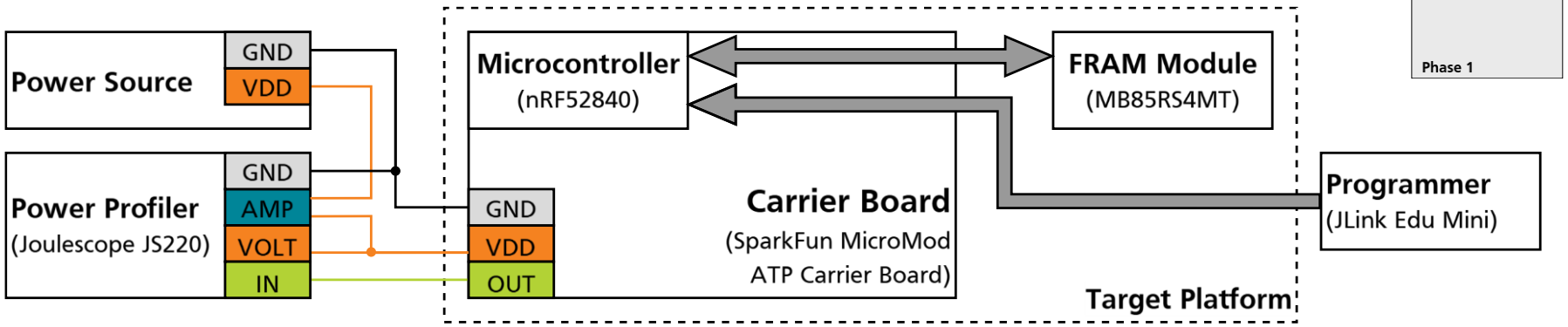


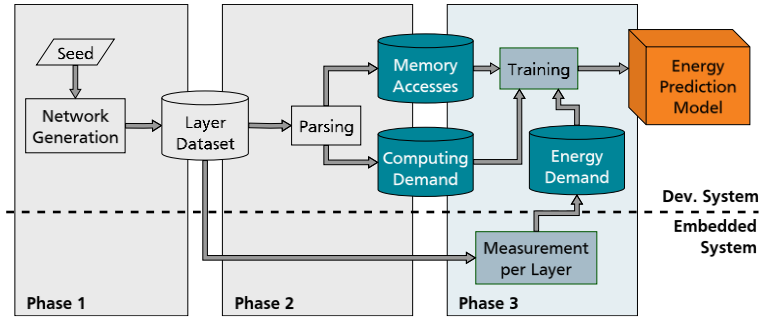
Parsing of Layers and Preparation for on-device measurement

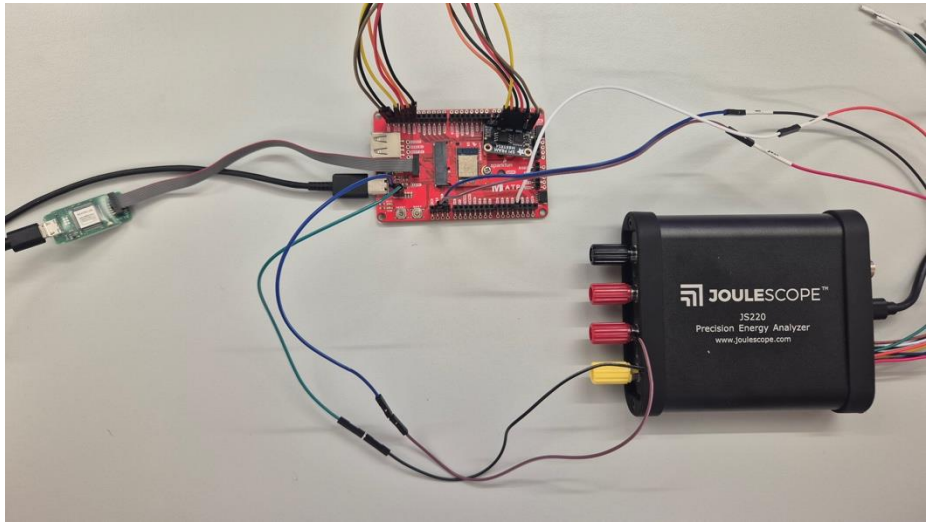
- **Quantization** of DNNs required for on-device inference and training
- **Conversion** of quantized DNNs to C-code and addition of structures required for training (autograd, optimizer, loss)
- **Parsing of tensor sizes** required for **checkpoints** between layers (weights, activations, gradients)
- **Estimation** of DNN memory accesses and MAC operations

Energy Prediction Model Development

Phase 3: Hardware-Setup



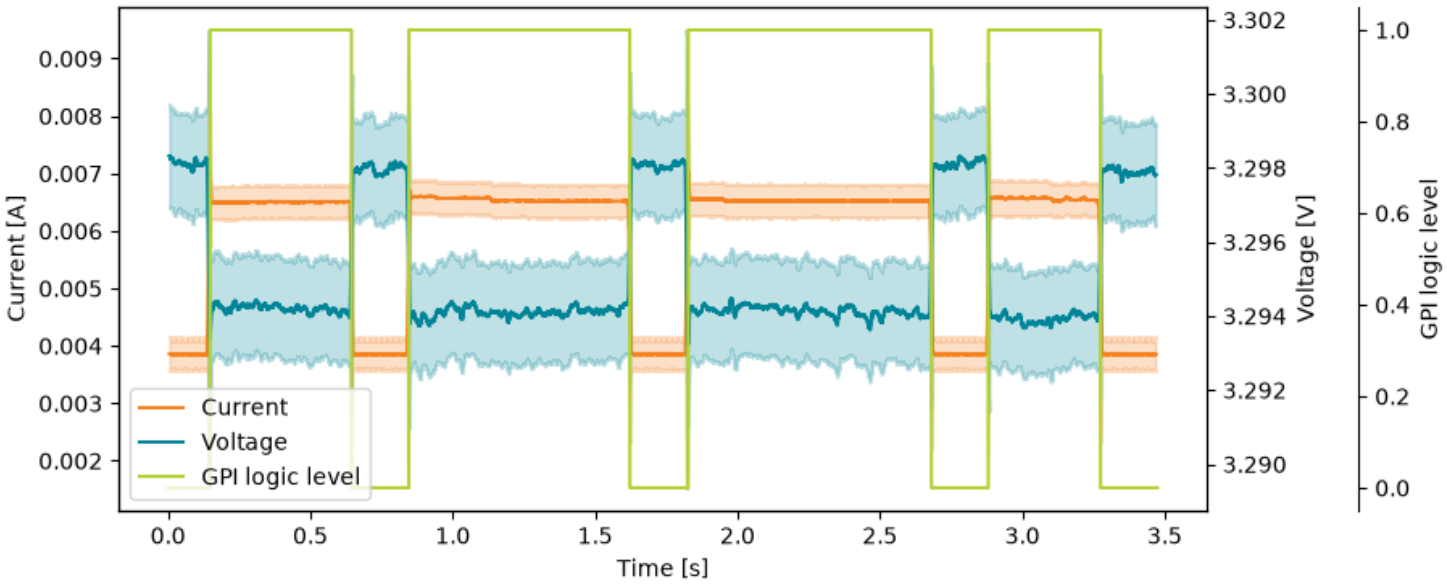


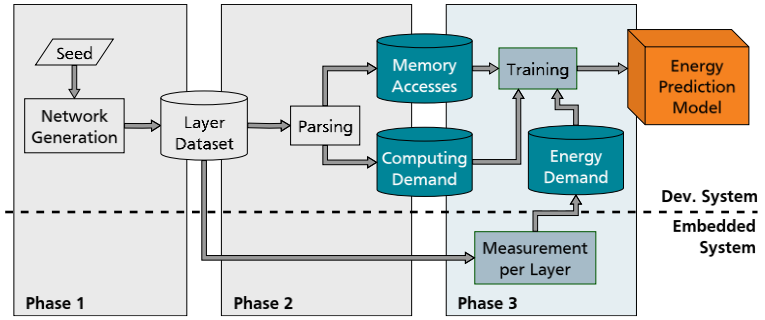


- Microcontroller - Nordic nRF52840
  - 64 MHz ARM **Cortex-M4 with FPU**, 1 MB Flash, 256 kB RAM
- Carrier Board – SparkFun Micromod ATP Carrier Board
  - Measurement pins, interchangeable processor boards
- FRAM Module – **Adafruit MB85RS4MT**
  - 4 Mbit SPI with 512 kB Storage
- Power Profiler – **JouleScope JS220**
  - ±15 V / ±3 A range, 0.5 nA resolution, 2 MHz sampling rate

Energy Prediction Model Development

Phase 3: Energy Measurements

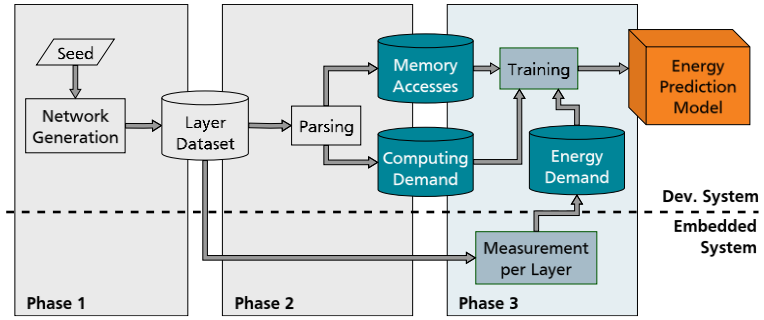




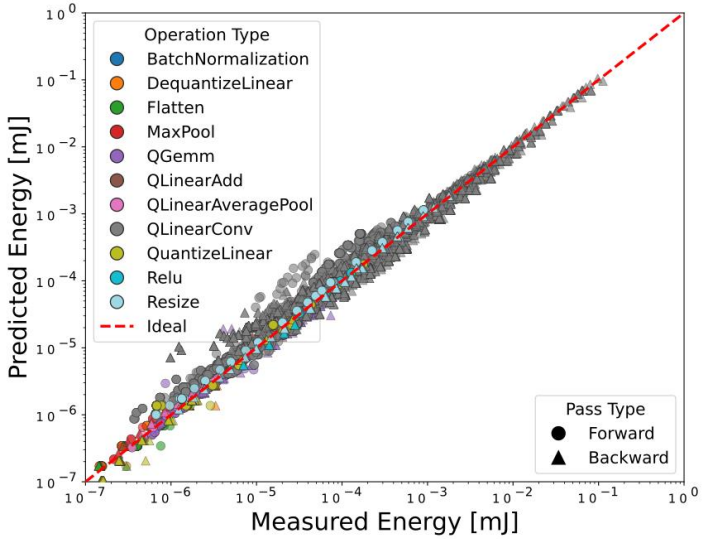
- Insertion of **short sleep phases** (200 ms) between the execution of the forward and backward passes or FRAM write/read process of every tested DNN layer.
- Insertion of **GPIO toggles** directly before and after the execution of each network layer or FRAM write/read process.
  - **Marking** the exact **execution** in voltage and current signal, as well as **inference time** measurement.
- **Compilation, deployment, and measurement** of current, voltage and execution time for inference, training and state preservation/recovery of all sampled network architectures on the target architecture
  - **Calculation of averaged energy** per layer or FRAM read/write access over multiple separate executions

Energy Prediction Model Development

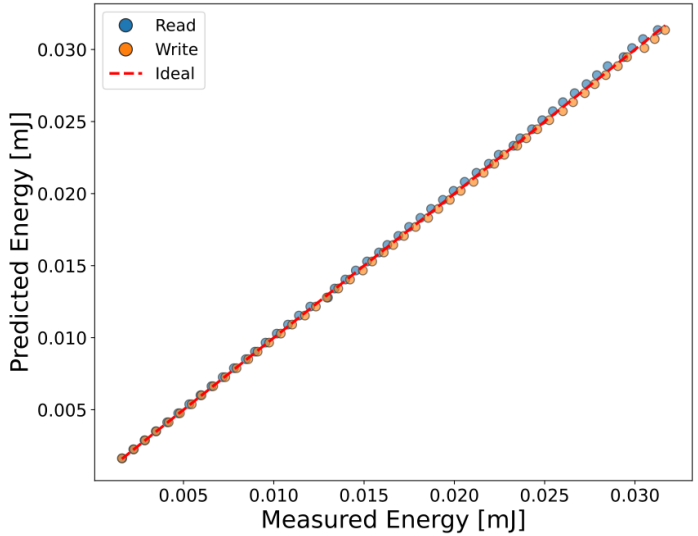
Phase 3: Energy Prediction Model Training



Layer Energy Prediction Model



Checkpoint/NVM Energy Prediction Model



	#	MAE [J] ↓		MAPE [%] ↓		WAPE [%] ↓	
		forward	backward	forward	backward	forward	backward
Layers	907	$2.47 \cdot 10^{-5}$	$3.52 \cdot 10^{-5}$	41.8	22.5	46.9	11.4
Total Layers	1814	$3.00 \cdot 10^{-5}$		32.2		16.6	
FRAM Read	50	$1.51 \cdot 10^{-4}$		0.9		0.9	
FRAM Write	50	$1.54 \cdot 10^{-4}$		0.9		0.9	
Total FRAM	100	$1.48 \cdot 10^{-4}$		0.9		0.9	

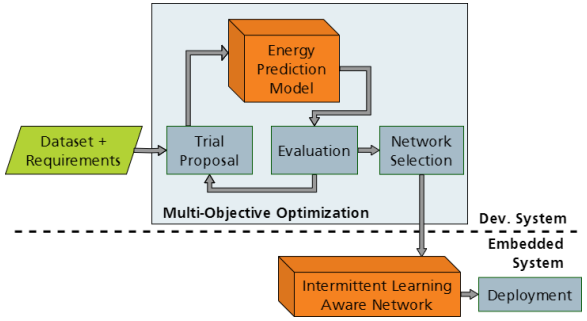
$$MAE = \frac{1}{n}\sum|y - \hat{y}|$$

$$MAPE = \sum\frac{|y - \hat{y}|}{y}$$

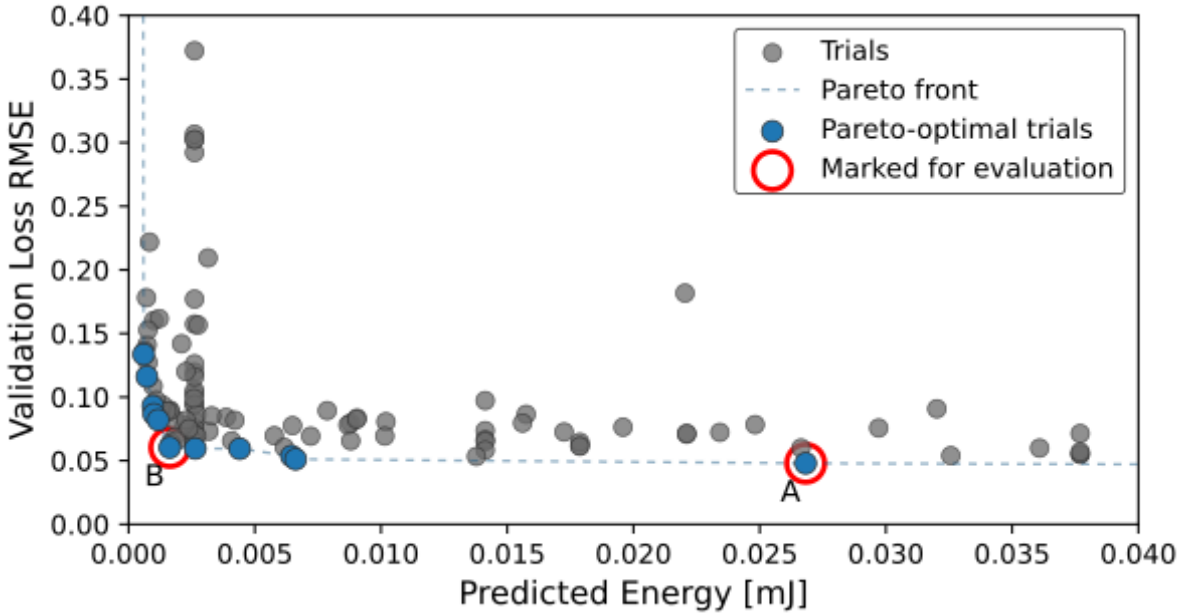
$$WAPE = \frac{\sum|y - \hat{y}|}{\sum|y|}$$

Search of Intermittent Learning Aware Networks

Multi-Objective Optimization for CWRU



- **Example:** CWRU bearing-vibration dataset[1]
- Optimization of convolutional autoencoders for intermittent on-device learning on a **nRF52840 Cortex-M4 MCU**.
- Fixed input shape of 2×256, identical encoder/decoder depth, latent dimension of 16.
  - **Search space:** hidden channels, layer-wise pruning rate.
- **NSGA-II** sampler over 200 trials (using optuna):
  - Minimize validation reconstruction loss on normal data
  - Minimize predicted total energy consumption
- **Training** of each candidate **for 20 epochs** with Adam and MSE loss. **Pruning** was applied **after epoch 10**.



Comparison of non-dominated trials A and B: negligible validation loss increase with **94% energy reduction** from 0.0268 mJ to 0.00163 mJ

1. https://engineering.case.edu/bearingdatacenter

Conclusion

Results and Outlook

Prediction of per-layer energy consumption for the design of intermittent learning DNNs on MCUs

- Per layer energy demand prediction for inference and training on MCUs (16.6% WAPE)
- Energy demand prediction for state preservation and recovery (checkpoints) between layers (0.9% WAPE)

Our results allow offline multi-objective optimization…

- … with per layer energy demand optimization for energy storage system design
- … with optimized state preservation for checkpoints between layers
- … energy demand optimization for inference and on-device training of DNNs