EMBEDDED ARTIFICIAL NEURAL NETWORKS FOR ENERGY-RESTRICTED EDGE COMPUTING APPLICATIONS

Alperen Aksoy[a], Ilja Bekman[a], Sarah Fleitmann[a], Qader Dorosti[b], Jan Vogelbruch[a], Vesselin Dimitrov[b], Fabian Hader[c], Stefan van Waasen[a, d]

[a] Peter Grünberg Institute (PGI), Integrated Computing Architectures (ICA | PGI-4), Forschungszentrum Jülich GmbH, Germany

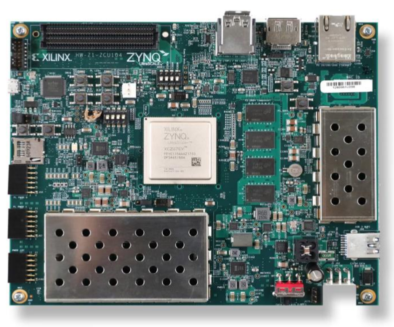
[b] Center for Particle Physics Siegen, Department für Physik, Universität Siegen, Germany

[c] JARA-FIT Institute for Quantum Information, Forschungszentrum Jülich GmbH and RWTH Aachen University, Germany

[d] Faculty of Engineering, University Duisburg-Essen, Germany

1. MOTIVATION

- Edge devices have limited energy and memory
- Conventional neural networks are too heavy for low-power hardware
- Need for **efficient embedded neural networks**



[1]

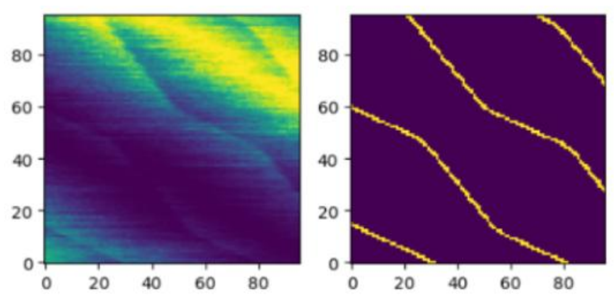
Key Concept: Quantized Neural Networks (QNNs)

- Reduce precision of weights and activations
  - Smaller memory footprint
  - Faster inference
  - Lower energy consumption

3. APPLICATIONS

Automated Qubit Tuning

- QNN model: Quantized U-Net
- Energy-efficient quantum dot calibration
- Charge transfer detection



Search for Hidden Particles (SHiP) - CERN

- QNN model: BNN on FPGA
- Capture particles that interact feebly with ordinary matter
- Processing and classification of silicon photomultiplier (SiPM) signals (Fig. 5)
  → Real-time filtering to reduce volume of transmitted data

Pierre Auger Observatory

- QNN model: BNN on FPGA
- First-level, real-time triggering of radio signals induced by cosmic rays
- Low-latency and resource-efficient classification on detector level
  → Autonomous triggering under realistic noise conditions

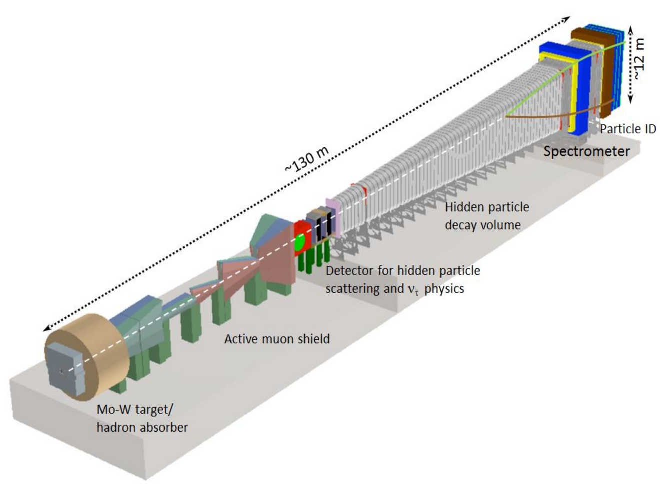


Fig. 3: Overview of the target and experimental area for the SHiP detector as implemented in the physics simulation [3].

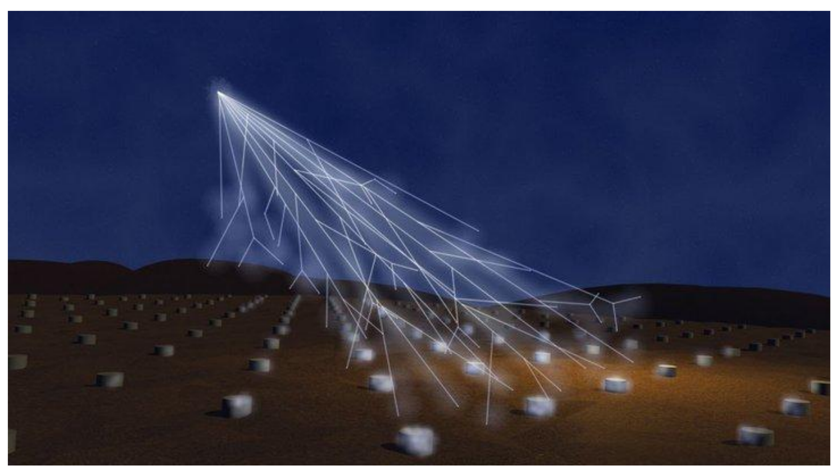


Fig. 4: Illustration of an extensive air shower produced by an ultra-high energy cosmic ray [4].

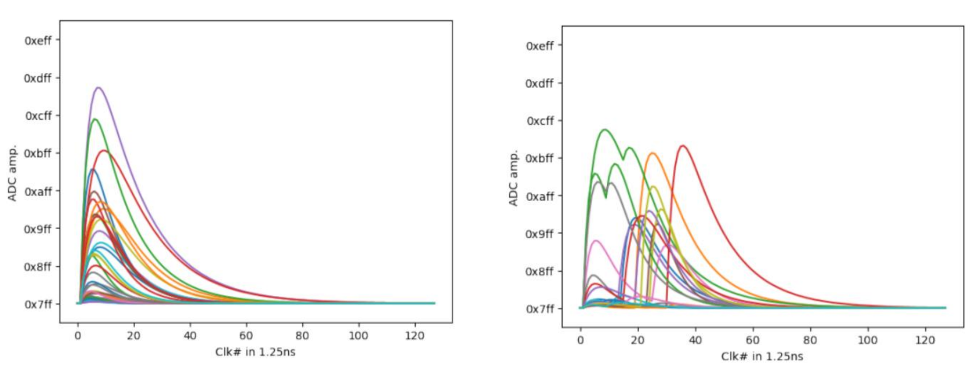
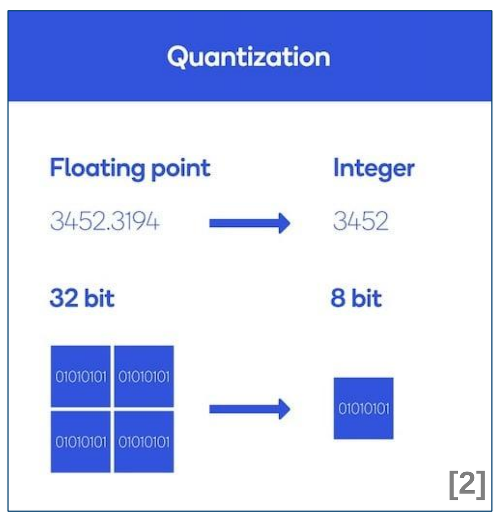


Fig. 5: Plot of signals from SiPM data. On the left it should be classified as good and on the right as bad.

2. METHODS

Post Training Quantization (PTQ)

- Quantize pre-trained full-precision model
- Pros: simple, fast
- Cons: potential accuracy drop



↓ Memory usage
↓ Power consumption
↑ Inference speed

Quantization Aware Training (QAT)

- Simulates the effects of quantization during training
- Pros: maintains accuracy
- Cons: longer training time

Binary Neural Networks (BNNs)

- Weights & activations constrained to 1 or 2 bits
  - Look Up Table (LUT)-based with minimal Flip-Flops (FFs), no BRAMs and no DSPs
  - Extreme memory & computation reduction
- Ideal for ultra-low-power edge devices like FPGAs
- Training with Genetic Algorithm (GA) possible

4. RESULTS

- PTQ achieves significant memory saving while preserving segmentation accuracy in qubit tuning

Model	Unquantized	PTQ	QAT
UNet-447	89%	90%	90%
UNet-38k	99%	99%	24%

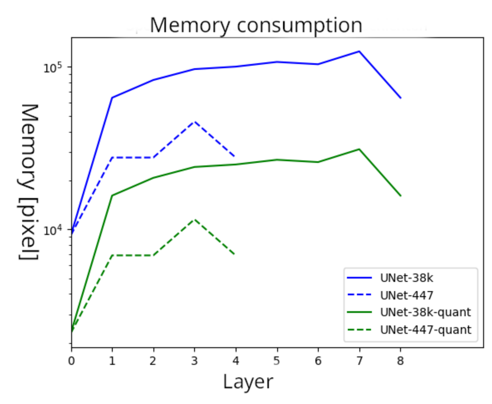


Fig 6: Memory consumption during inference over layers.

- 2-bit LUT-based BNNs show (Table 1):
  - Very low inference latency on FPGA hardware
  - Feasibility for real-time signal processing
- Python-to-VHDL conversion workflow implemented
- Published open-source Python package for implementing BNNs: **Hardware-Constrained Learning for BNNs (HCL4BNN)[5]**

- Compressed and analyzed Convolutional Neural Networks (CNNs)
  - Gained good accuracy with a small parameter count on MNIST (see Table. 2)

	Accuracy	Latency in ns	LUTs x k	FFs x k	DSPs	BRAMs 18K
FINN	74±4%	24850	30	106	106	5
hls4ml	94.9%	3050	186	556	556	120
BNN a)	64±5%	15	58	0	0	0
BNN b)	74±5%	10	23	0	0	0

Table. 1: Results and comparison of FINN 2DCNN ((128-4-6-8-2) , Kernel [5,1], padding 2.0, int8) implementation, hls4ml CNN, BNN a) (128-64-128-2) b) (128-32-32-2) with input quantized to int7. Evaluated on SiPM data (Fig. 5).

Model	# Parameter	Accuracy
BinaryUltraMiniCNN	9,112	82%
TinyBinaryCNN	4,268	91%
UltraTinyBinaryCNN	3,147	85%
MicroBNN	2,062	86%
NanoBNN	1,013	77%

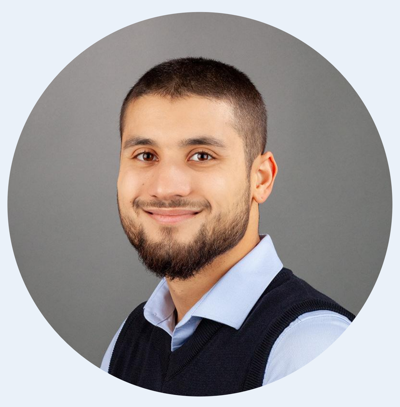
Table. 2: Results of compressed CNNs on MNIST dataset with small amount of parameters and binary weights and activations.

CONCLUSION

Quantization enables deployment of neural networks on energy-constrained edge devices.

Post Training Quantization constitutes a rapid solution with superior accuracy in comparison to unquantized networks.

BNNs are particularly well-suited for ultra-low power scenarios for fast inference requirements. Genetic algorithms are a viable solution for complex or non-differentiable problems.



Alperen Aksoy
Researcher / Software Engineer

a.aksoy@fz-juelich.de

Integrated Computing Architectures (ICA)
www.ica.fz-juelich.de

[1] https://www.amd.com/en/products/adaptive-socs-and-fpgas/evaluation-boards/zcu104.html
[2] https://www.allaboutcircuits.com/technical-articles/neural-network-quantization-what-is-it-and-how-does-it-relate-to-tiny-machine-learning/
[3] https://cds.cern.ch/record/2644153/plots
[4] Friedlander, M. A century of cosmic rays. Nature 483, 400–401 (2012)
[5] https://github.com/fzj-ica/HCL4BNNN