Quantized NN Workflow for Low-Latency Multi-Qubit State Discrimination on FPGA

Pradeep Kumar Gautam
Department of Electronic Systems Engineering
Indian Institute of Science
Bengaluru, India
pradeepgk@iisc.ac.in

Shantharam Kalipatnapu
Department of Electronic Systems Engineering
Indian Institute of Science
Bengaluru, India
shantharamk@iisc.ac.in

Shankaranarayanan H
Department of Electronic Systems Engineering
Indian Institute of Science
Bengaluru, India
shankaranara@iisc.ac.in

Ujjawal Singhal
Department of Physics
Indian Institute of Science
Bengaluru, India
ujjawals@iisc.ac.in

Benjamin Lienhard
Department of Chemistry
Princeton University
Princeton, USA
blienhard@princeton.edu

Vibhor Singh
Department of Physics
Indian Institute of Science
Bengaluru, India
vsingh@iisc.ac.in

Chetan Singh Thakur
Department of Electronic Systems Engineering
Indian Institute of Science
Bengaluru, India
csthakur@iisc.ac.in

Abstract

Measuring a qubit state is a fundamental but error-prone task in quantum computing. As the number of qubits increases, scalable readout demands multiplexed qubit measurement, which often suffers from crosstalk and other nonidealities. The conventional approach of qubit state detection often struggles with scalability and noise resilience. Neural network (NN)-based discriminators have emerged as a promising alternative, offering superior performance for inferring qubit states from multiplexed signals. However, their deployment on hardware platforms like Field programmable gate arrays (FPGA) is hindered by computation complexity, resource constraints, and latency. In this work, we present an end-to-end methodology for designing and deploying quantized NN-based multi-qubit state discriminators on FPGAs. We demonstrate that implementing a fully connected neural network accelerator for multi-qubit readout is advantageous, balancing computational complexity with low latency requirements without significant loss in accuracy. We significantly reduce the FPGA footprint and latency of the neural network by employing quantization-aware training (QAT) to quantize weights, activations, and inputs, and leveraging automated mapping flows for efficient, hardware-aware design-space exploration and optimization. The hardware accelerator performs frequency-multiplexed readout of five superconducting qubits in 26.7 ns, on a radio frequency system on chip (RFSoC) ZCU111 FPGA. These modules can be implemented and integrated into existing FPGA based quantum control and readout platforms, making them ready for experimental deployment.

Keywords

Quantum computing, superconducting qubits, multi-qubit readout, FPGA, RFSoC, machine learning, neural networks, Accelerator

ACM Reference Format:
Pradeep Kumar Gautam, Shantharam Kalipatnapu, Shankaranarayanan H, Ujjawal Singhal, Benjamin Lienhard, Vibhor Singh, and Chetan Singh Thakur. 2026. Quantized NN Workflow for Low-Latency Multi-Qubit State Discrimination on FPGA. In *Proceedings of the 16th International Symposium on Highly Efficient Accelerators and Reconfigurable Technologies (HEART 2026), June 17–19, 2026, Heidelberg, Germany*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3814576.3814580

1 Introduction

Quantum processors are expected to solve specific computational tasks significantly more efficiently than their classical counterparts [2, 8, 16, 21]. To reduce resource requirements, especially from a readout perspective, multi-qubit readout is often achieved using frequency-division multiplexing [7]. Various methods, including matched filters (MFs), support vector machines (SVMs), and neural networks (NNs), have been used to process these frequency-multiplexed signals for inferring qubit states [4, 11–15, 20]. When implementing such methods, minimizing latency in post-processing for qubit-state inference is crucial for enabling on-the-fly corrective actions or providing active feedback loop [27]. Additionally, NN-based discriminators scale efficiently, as they do not require qubit-specific processing, such as digital demodulation, for each qubit [11].

Scalable field-programmable gate array (FPGA) systems, particularly radio frequency system on chip (RFSoC), have gained

HEART 2026, Heidelberg, Germany

ACM ISBN 979-8-4007-2452-7/26/06
https://doi.org/10.1145/3814576.3814580

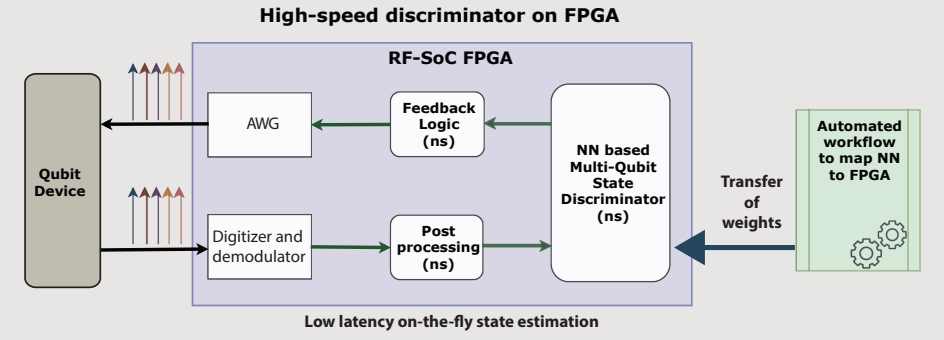

Figure 1: The figure shows an integrated approach using a high speed FPGA system for the control and readout of a multi-qubit system. A NN-based qubit state discriminator on the board helps in reducing the latency and allow on-the-fly real-time multi-qubit readout.

significant attention for integrated qubit control and readout in quantum computing [18, 22, 23, 25]. In this context, NN-based state discriminators promise to enhance frequency-multiplexed readout by improving fidelity against crosstalk. However, FPGA deployment of NNs faces severe resource and latency constraints for large models [14]. Although Quantization-aware training (QAT) effectively reduces model precision—even to the binary level, with minimal accuracy loss [1, 10], existing approaches [3, 6, 9, 14, 19, 27] remain limited by qubit-specific pre-processing, hand-crafted implementations, or lack of mixed-precision quantization. These constraints hinder scalability for larger qubit systems.

We address these gaps with an end-to-end automated workflow from Brevitas QAT [17] to FINN-R [1] FPGA mapping. Fig. 1 shows the superconducting qubit readout setup and the methodology demonstrated in this work to significantly reduce the signal processing delays. A typical experimental setup of superconducting qubits includes a dilution refrigerator maintained at a temperature of around 20 mK housing a qubit quantum processor and an RFSoC board at room temperature used to generate microwave pulses for qubit control and readout and post-process readout signals. Specifically, the readout probe pulse is passed through a radio frequency (RF) digital-to-analog converter (RF-DAC) and, after interacting with the quantum processor to acquire a qubit-state-specific signal characteristic, is fed into a high-speed RF analog-to-digital converter (RF-ADC). The RF-ADC and RF-DAC form the interface between the analog signals to and from the quantum processor and the digital RFSoC processing elements.

Major contributions of this work include the following.

- Complete end-to-end pipeline from QAT to FPGA synthesis, making large NN models (even 1.6M parameters) practical on resource-constrained hardware.
- Single NN handling all qubits simultaneously, without the need for qubit-specific preprocessing or networks.
- Novel hidden layer segmentation with FINN-R adaptations, achieving sub-50 ns latency for 30K+ parameter model.
- Piecewise processing with mixed-precision quantization (4-bit input, 1-bit weights, and 4-bit activations) delivering 26.7 ns inference latency for a 5-qubit system.

The article is organized as follows: Section 1 covers the design methodology and implementation of NN-based qubit-state discriminators. Section 2 presents the results and compares them with state-of-the-art methods. Finally, section 3 concludes the work.

1.1 Neural Network Model Optimization and Quantization

We attempt the proposed workflow on a multi-qubit superconducting quantum processor with 5 transmon qubits. The details of the multi-qubit setup and data acquisition are given in Appendix-A. The potential for mixed-precision quantization of inputs, weights, and activations down to one bit within a large NN design necessitates careful architectural choices. To streamline design exploration, we adopted a strategy similar to that in Ref. [24] and started with a baseline model presented in Ref. [11]. This model has an input size of 1000, which includes 500 samples of each of the in-phase (I) and quadrature (Q) components corresponding to 1 µs of measurement time. The model features three hidden layers with 1000, 500, and 250 nodes, respectively, and an output layer with 32 nodes. Each hidden layer includes a linear transformation followed by batch normalization, a dropout layer, and a rectified linear unit (ReLU). We systematically reduced the input size and the number of nodes in the hidden layers. The various model archetypes are represented as $N_I \times N_{H_1} \times \cdots \times N_{H_k} \times N_O$, where N_I, N_{H_i}, and N_O denote the number of nodes in the input layer, i^{th} hidden layer ($i \in \mathbb{Z}^+$), and output layer, respectively.

The hyperparameters used for the NN training are summarized in Appendix—B. The effect of model size variation and various quantization combinations is given in Appendix-C, based on which we choose a model with architecture $512 \times 64 \times 5$ having a quantization scheme of 4-bit for the input and 2-bit for both weights and activations.

1.2 FPGA Acceleration

To achieve low-latency readout, the dataflow architecture-based framework of FINN-R [1] stands out. FINN-R implements quantized matrix multiplication of low-precision data using multi-vector threshold units (MVTU), with one such unit used for each layer of

the QNN. A significant hurdle in achieving complete parallelism for moderate-size models is the limitation imposed by the AXI-stream interconnect's connection width [28], which limits both inter-layer and intra-layer connectivity.

Low-latency novel architecture: To maximize parallelism, we address this key bottleneck from the AXI-stream interconnect width. The modified hardware architecture is based on a NN with a configuration of $512 \times 64 \times 5$ as described earlier. The first hidden layer of the model, consisting of 64 nodes, is divided into eight equal segments, each containing eight nodes. These segmented nodes can operate in parallel on the FPGA, eliminating the need for time-multiplexing and reducing total latency. However, the new architecture has segment-specific *Mul* nodes with different multiplication factors, preceding the *Concat* layer. These nodes cannot propagate beyond *Concat* layer and be absorbed into downstream MVTUs, resulting in partial FPGA mapping and forcing software execution of the downstream operations. To address this issue, we insert a unified *Quantidentity* layer in each segment prior to the *Concat* layer, ensuring identical scaling factors across segments. This enables *Mul* node propagation past *Concat* and subsequent absorption into MVTUs. This novel architecture fully parallelizes the FPGA implementation of the model, reducing latency by eliminating the need for time-multiplexing. Fig. 2(a) displays the modified NN design generated by FINN-R.

As an alternate approach to achieve low latency while maintaining fidelities, we adopted a deeper neural network architecture that processes the input data in a piecewise fashion [19]. In this scheme, the readout signal is segmented and processed as it arrives. Each network layer handles a specific portion of the signal, along with the output from the preceding layer. The final segment of the readout signal, which marks its completion, is provided as input to only the last layer. Consequently, only the last layer contributes to the readout latency, as all other layers are evaluated in parallel with readout data acquisition. We utilize this property to use higher bit-width for the weights and activations of inner hidden-layers, while keeping the last hidden-layer at lower bit-widths. The architecture for the model with a size of $256 \times 128 \times 128 \times 128 \times 128 \times 5$ is displayed in Fig. 2(b). This design enables the use of a deeper network with compact hidden layers, balancing low latency with high classification performance.

2 Results and Discussion

2.1 Performance and Resource Utilization

To demonstrate the effectiveness of our approach in implementing large models, we applied it to the NN model described in Ref. [11]. This network architecture consists of an input dimension of $1,000$ nodes, three hidden layers with $1,000$, 500, and 250 nodes, and 32 output nodes for each possible state. The total number of learnable parameters sums up to $1,634,782$, each requiring 4 bytes of storage in floating-point representation. This translates to 50 MB of storage just for the weights and biases. Using this standard floating-point approach, the implementation exceeds the resource limits of most available FPGA devices, as reported in Ref. [14].

To address this challenge, we adopted the methodology described in the previous section to implement the model in Brevitas and convert it to a dataflow graph using FINN-R. The input, weight, and activation quantization used are 4, 2, and 2 bits, respectively. The model achieves a geometric mean fidelity (see Appendix-A) of 0.904, which is only 0.9% below the one reported in Ref. [11]. The storage requirement reduces by 16×, making model implementation on FPGA feasible even with 1.6 million parameters.

Latency and resource utilization comparisons for various NN model architectures and quantizations are presented in Table in Appendix. Reducing input size and hidden layers have a significant impact on resource usage and drastically improve latency with a minimal drop in fidelity. The Latency reported in the table includes the one clock cycle required for the boxcar operation to provide input to the NN.

The novel approach of splitting the first hidden layer of Arch-5 ($512 \times 64 \times 5$) into eight parallel segments results in Arch-7, reducing latency from 33 cycles to 19 cycles, an improvement of 42%. However, this comes with a significant increase in resource consumption, likely due to the additional *QuantIdentity* layers introduced before the *Concat* layer and the heightened level of parallelism. The piecewise processing architecture exhibits a latency of 9 cycles, corresponding to 26.7 ns, with one cycle dedicated to the boxcar operation. Importantly, the boxcar operation in all the variants is applied directly to the incoming signal without demodulation, thereby aligning with our design principle of non-qubit-specific processing. As a result, only one instance of the filter is required, irrespective of the number of qubits being discriminated.

2.2 Comparison

As an alternative to traditional signal processing, recent work has utilized NNs for their robustness to signal perturbations and their capability for multi-qubit state discrimination [11], and a few works have implemented light-weight NN on FPGA for real-time deployment [3, 6, 14, 19, 27]. Table 1 summarizes our networks' latency, resource usage, fidelity and number of learnable parameters compared to the state-of-the-art. We compare the results of our approach with Maurya *et al.* [14] and Guo *et al.* [9], who attempt multi-qubit state discrimination on the same 5-qubit readout data, while comparison with single-qubit works is reported subsequently.

Result on 5-Qubit: Maurya *et al.* [14] implemented a NN-based discriminator for frequency-multiplexed qubit readout. Their approach uses a matched filter per qubit for dimensionality reduction before processing the data with a lightweight NN. This model has $1,112$ learnable parameters, while the latency for this approach is not reported in time. Guo *et al.* [9] employ knowledge distillation to compress a neural network for FPGA-based multi-qubit readout. To preserve classification accuracy, they enhance the student model's inputs by appending MF output. They train two separate neural network architectures—one for qubit group Q1, Q4, Q5 and another for more challenging qubits Q2 and Q3. The two NN variants have 1972 and 6754 learnable parameters respectively, and a latency of 32 ns. They deploy a separate NN for each qubit, requiring 5 NNs in total for the task. Both these approaches require appending the output of the trained matched-filter, necessitating qubit-specific processing.

Our NN qubit-state discriminator, based on Arch-7, Arch-8, and Arch-9 implemented on FPGA as shown in Table in Appendix, has latencies of 20 cycles, 12 cycles, and 10 cycles, respectively,

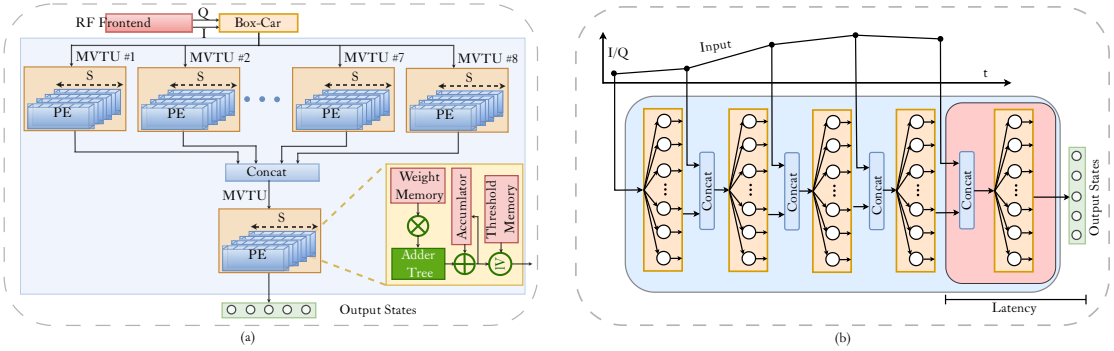

Figure 2: Quantized neural network (QNN) archetypes. (a) Fully parallel hardware architecture of the QNN $((512 \times 8) \times 8 \times 5)$**. The input feature of size** 1×512 **is connected to all the** 8 **segments. (b) Piecewise layered QNN architecture** $256 \times 128 \times 128 \times 128 \times 128 \times 5$**. The first layer receives an initial input size of** 1×256**, while subsequent layers take an input size of** 1×128**, consisting of** 1×64 **from the previous layer's output and** 1×64 **from the corresponding input segment.**

Table 1: Comparison of proposed discriminator using neural networks with the state-of-the-art.

		Discriminator	# of Parameters	Fidelity	Latency (ns)	Resources			Design Methodology
						LUTs	FFs	DSPs	
For 5-Qubit Readout									
Maurya *et al.* [14]		Demodulation +Matched Filter + NN	1112	89.3	Not Reported	17917	3456	32	hls4ml
Guo *et al.* [9]		Demodulation+ Integrator + MF	6754	90.4	32	99272	82159	656	Hand-crafted (Separate NN for each qubit)
This Work	Arch-7	Integrator + NN	**33295**	89.7	**49.6**	107026	60696	0	**QAT, FINN-R**
	Arch-9	Integrator + NN	**42124**	90.0	**26.7**	104527	95945	0	
For Single Qubit Readout†									
Guglielmo *et al.* [3]		Integrator + NN	3208	96.0	32	66669	34369	0	QAT, hls4ml
Farooq *et al.* [6]		Integrator + NN	~200	95.9	23.82	6095	9688	0	QAT, hls4ml
This Work		Integrator +NN	**419**	**95.9**	**25.88**	**2529**	**3747**	0	**QAT, FINN-R**

† The reported results are for single-qubit readout data, available at [5].

including 1 cycle for the boxcar operation to provide input to the NN. Consequently, Arch-7, Arch-8, and Arch-9 achieve latencies of 49.6 ns, 35.16 ns, and 26.7 ns, respectively, which are comparable to or better than the state-of-the-art implementations. Meanwhile, the proposed approach has at least 6 times more learnable parameters, making it more robust and versatile for complex readout scenarios. The fidelities achieved for individual qubits with QNN, and their comparison with similar works on 5 qubits, are reported in Table 2. The geometric mean fidelities are reported for both 5-qubit and 4-qubit configurations, with the second qubit excluded in the latter due to its reduced readout accuracy caused by noise and short coherence time. Our approach achieves fidelities comparable to prior works for the 5-qubit case.

Result for Single Qubit Readout dataset: Our approach attempts a multi-qubit readout system, but to have a fair comparison of results with a single-qubit implementation, we also implement a NN for single-qubit state discrimination for readout data of Ref. [3] available at [5]. We implement a NN with $100 \times 4 \times 1$ architecture

Table 2: Comparison of fidelities with other similar works for 5-qubit system.

Fidelity	F_{GM}	F_{4GM}†	$Q1$	$Q2$	$Q3$	$Q4$	$Q5$
Lienhard★ *et al.* [11]	0.912	0.956	0.969	0.753	0.943	0.946	0.970
Maurya *et al.* [14]	0.893	0.940	0.965	0.730	0.908	0.934	0.953
Guo *et al.* [9]	0.904	0.947	0.968	0.748	0.929	0.934	0.959
This Work (Arch-9)	0.90	**0.947**	0.959	0.734	0.930	0.940	0.961

† Geometric Mean Fidelity for 4 qubits, excluding the second qubit.
★ Fidelities reported are for the floating point NN model.

with quantization of 4, 2, 2 for input, weight and activation respectively. The results are summarized and compared with [3, 6] in Table 1. Comparison with [19, 27] is not given as the readout data for these works are not available.

Guglielmo *et al.* [3] have implemented a quantized NN using hls4ml with QAT using QKeras framework. It has 3208 parameters consisting of a single hidden layer and achieves a fidelity of 96%. The implemented NN has a latency of 32 ns. Farooq *et al.* [6] builds

on the work of Guglielmo *et al.* [3] and uses LogicNet [26] for implementing NN on to FPGA. These approaches offer flexibility in terms of changing the NN architectures with short design time, but lack the the flexibility of mixed-precision quantization for weights and activation. Our implementation for this dataset gives fidelity of 95.9% with a model having 419 parameters and a latency of 25.88 ns. The resultant network on FPGA uses 2516 LUTs. This is 26.3× and 2.4× improvement over [3] and [6] respectively, while achieving the same fidelity with a marginal increase in latency.

2.3 Discussion

This work demonstrates a practical end-to-end methodology that successfully maps large-scale 1.6M-parameter neural networks onto RFSoC FPGAs using Brevitas QAT [17] and FINN-R [1] synthesis. Hidden-layer segmentation with FINN-R adaptations, piecewise processing, and mixed-precision quantization achieves ultra-low latency for 5-qubit discrimination with minimal drop in fidelitiy. The single-NN topology scales naturally with N output nodes for N qubits, avoiding exponential complexity and qubit-specific processing. The framework generalizes well to single-qubit readout, achieving the same fidelity with atleast 2.4× less resources than prior single-qubit implementations.

Neural networks trained to infer each qubit independently in a multi-qubit system do not effectively capture inter-qubit crosstalk and correlations as discussed in Ref. [9]. So, we feel that using a single-NN for a multi-qubit system is the way forward. In future work, augmentation of input features for handling noisy qubits like Q2 of the current system, can be explored to achieve overall more robust fidelities for the combined system. Readout trace evolution over time can be used to identify qubit relaxation events, as demonstrated in Ref. [14], without requiring qubit-specific processing.

3 Conclusion

We present NN accelerators for state discrimination of frequency-multiplexed qubit readout traces of a superconducting multi-qubit processor. We address key challenges of resource, latency, qubit-specific processing, and lengthy design cycle by employing end-to-end workflow with mixed quantization, and architectural adaptations. Notably, our 42K-parameter models (Arch-9) represent 6–36× larger scale than prior deployed discriminators while achieving low latency of 26.7 ns. This robustness enables better crosstalk mitigation in complex multi-qubit scenarios. The novel approach of segmented layer implementation with FINN-R compatible layer adaptations allows for achieving sub 50 ns latency. The demonstrated substantial reduction in latency while maintaining qubit-state discrimination fidelity enables the implementation of active feedbacks, thereby significantly improving the performance of quantum processors.

Acknowledgments

This work was supported by the NQM Program within the Department of Science and Technology (DST), Government of India (GoI).

References

[1] Michaela Blott, Thomas B Preußer, Nicholas J Fraser, Giulio Gambardella, Kenneth O'brien, Yaman Umuroglu, Miriam Leeser, and Kees Vissers. 2018. FINN-R: An end-to-end deep-learning framework for fast exploration of quantized neural networks. *ACM Transactions on Reconfigurable Technology and Systems (TRETS)* 11, 3 (2018), 1–23.

[2] Sergio Boixo, Sergei V Isakov, Vadim N Smelyanskiy, Ryan Babbush, Nan Ding, Zhang Jiang, Michael J Bremner, John M Martinis, and Hartmut Neven. 2018. Characterizing quantum supremacy in near-term devices. *Nature Physics* 14, 6 (2018), 595–600.

[3] Giuseppe Di Guglielmo, Botao Du, Javier Campos, Alexandra Boltasseva, Akash V Dixit, Farah Fahim, Zhaxylyk Kudyshev, Santiago Lopez, Ruichao Ma, Gabriel N Perdue, et al. 2025. End-to-end workflow for machine learning-based qubit readout with QICK and hls4ml. *arXiv preprint arXiv:2501.14663* (2025).

[4] Zi-Han Ding, Jin-Ming Cui, Yun-Feng Huang, Chuan-Feng Li, Tao Tu, and Guang-Can Guo. 2019. Fast high-fidelity readout of a single trapped-ion qubit via machine-learning methods. *Physical Review Applied* 12, 1 (2019), 014038.

[5] Botao Du. 2024. *Data for "End-to-end workflow for machine learning-based qubit readout with QICK and hls4ml".* doi:10.5281/zenodo.14427490

[6] Muhammad Ali Farooq, Giuseppe Di Guglielmo, Abhi Rajagopala, Nhan Tran, VA Chhabria, and Aman Arora. 2025. LUNA: LUT-Based Neural Architecture for Fast and Low-Cost Qubit Readout. *arXiv preprint arXiv:2512.07808* (2025).

[7] Austin G Fowler, Matteo Mariantoni, John M Martinis, and Andrew N Cleland. 2012. Surface codes: Towards practical large-scale quantum computation. *Physical Review A* 86, 3 (2012), 032324.

[8] Lov K Grover. 1996. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. 212–219.

[9] Xiaorang Guo, Tigran Bunarjyan, Dai Liu, Benjamin Lienhard, and Martin Schulz. 2025. KLiNQ: Knowledge Distillation-Assisted Lightweight Neural Network for Qubit Readout on FPGA. *arXiv preprint arXiv:2503.03544* (2025).

[10] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. 2018. Quantized neural networks: Training neural networks with low precision weights and activations. *journal of machine learning research* 18, 187 (2018), 1–30.

[11] Benjamin Lienhard, Antti Vepsäläinen, Luke CG Govia, Cole R Hoffer, Jack Y Qiu, Diego Ristè, Matthew Ware, David Kim, Roni Winik, Alexander Melville, et al. 2022. Deep-neural-network discrimination of multiplexed superconducting-qubit states. *Physical Review Applied* 17, 1 (2022), 014024.

[12] Easwar Magesan, Jay M Gambetta, Antonio D Córcoles, and Jerry M Chow. 2015. Machine learning for discriminating quantum measurement trajectories and improving readout. *Physical review letters* 114, 20 (2015), 200501.

[13] Yuta Matsumoto, Takafumi Fujita, Arne Ludwig, Andreas D Wieck, Kazunori Komatani, and Akira Oiwa. 2021. Noise-robust classification of single-shot electron spin readouts using a deep neural network. *npj Quantum Information* 7, 1 (2021), 136.

[14] Satvik Maurya, Chaithanya Naik Mude, William D Oliver, Benjamin Lienhard, and Swamit Tannu. 2023. Scaling Qubit Readout with Hardware Efficient Machine Learning Architectures. In *Proceedings of the 50th Annual International Symposium on Computer Architecture*. 1–13.

[15] Rohit Navarathna, Tyler Jones, Tina Moghaddam, Anatoly Kulikov, Rohit Beriwal, Markus Jerger, Prasanna Pakkiam, and Arkady Fedorov. 2021. Neural networks for on-the-fly single-shot state classification. *Applied Physics Letters* 119, 11 (09 2021), 114003. doi:10.1063/5.0065011

[16] Peter JJ O'Malley, Ryan Babbush, Ian D Kivlichan, Jonathan Romero, Jarrod R McClean, Rami Barends, Julian Kelly, Pedram Roushan, Andrew Tranter, Nan Ding, et al. 2016. Scalable quantum simulation of molecular energies. *Physical Review X* 6, 3 (2016), 031007.

[17] Alessandro Pappalardo. 2023. Xilinx/brevitas. doi:10.5281/zenodo.3333552

[18] Kun Hee Park, Yung Szen Yap, Yuanzheng Paul Tan, Christoph Hufnagel, Long Hoang Nguyen, Karn Hwa Lau, Patrick Bore, Stavros Efthymiou, Stefano Carrazza, Rangga P Budoyo, et al. 2022. ICARUS-Q: Integrated control and readout unit for scalable quantum processors. *Review of Scientific Instruments* 93, 10 (2022).

[19] Kevin Reuer, Jonas Landgraf, Thomas Fösel, James O'Sullivan, Liberto Beltrán, Abdulkadir Akin, Graham J Norris, Ants Remm, Michael Kerschbaum, Jean-Claude Besse, et al. 2023. Realizing a deep reinforcement learning agent for real-time quantum feedback. *Nature Communications* 14, 1 (2023), 7138.

[20] Alireza Seif, Kevin A Landsman, Norbert M Linke, Caroline Figgatt, Christopher Monroe, and Mohammad Hafezi. 2018. Machine learning assisted readout of trapped-ion qubits. *Journal of Physics B: Atomic, Molecular and Optical Physics* 51, 17 (2018), 174006.

[21] Peter W Shor. 1999. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review* 41, 2 (1999), 303–332.

[22] Ujjawal Singhal, Shantharam Kalipatnapu, Pradeep Kumar Gautam, Sourav Majumder, Vaibhav Venkata Lakshmi Pabbisetty, Srivatsava Jandhyala, Vibhor Singh, and Chetan Singh Thakur. 2023. SQ-CARS: A Scalable Quantum Control and

Readout System. *IEEE Transactions on Instrumentation and Measurement* (2023).
[23] Leandro Stefanazzi, Kenneth Treptow, Neal Wilcer, Chris Stoughton, Collin Bradford, Sho Uemura, Silvia Zorzetti, Salvatore Montella, Gustavo Cancelo, Sara Sussman, et al. 2022. The QICK (Quantum Instrumentation Control Kit): Readout and control for qubits and detectors. *Review of Scientific Instruments* 93, 4 (2022).
[24] Wonyong Sung, Sungho Shin, and Kyuyeon Hwang. 2015. Resiliency of deep neural networks under quantization. *arXiv preprint arXiv:1511.06488* (2015).
[25] Mats O Tholén, Riccardo Borgani, Giuseppe Ruggero Di Carlo, Andreas Bengtsson, Christian Križan, Marina Kudra, Giovanna Tancredi, Jonas Bylander, Per Delsing, Simone Gasparinetti, et al. 2022. Measurement and control of a superconducting quantum processor with a fully integrated radio-frequency system on a chip. *Review of Scientific Instruments* 93, 10 (2022).
[26] Yaman Umuroglu, Yash Akhauri, Nicholas J Fraser, and Michaela Blott. 2020. LogicNets: Co-Designed Neural Networks and Circuits for Extreme-Throughput Applications. In *Proceedings of the International Conference on Field-Programmable Logic and Applications*. IEEE Computer Society, Los Alamitos, CA, USA, 291–297.
[27] Neel R Vora, Yilun Xu, Akel Hasim, Neelay Fruitwala, Nam Nguyen, Horan Liao, Jan Balewski, Abhi Rajagopala, Kasra Nowrouzi, Qing Ji, K. Brigitta Whaley, Irfan Siddiqi, Phuc Nguyen, and Gang Huang. 2024. QubiCML: ML-Powered Real-Time Quantum State Discrimination Enabling Mid-Circuit Measurements. In *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Vol. 02. 414–415. doi:10.1109/QCE60285.2024.10332
[28] Xilinx. 2024. AXI4-Stream Interconnect v1.1. https://docs.amd.com/v/u/en-US/pg035_axis_interconnect. [Online; accessed 20-Apr-2022].