A Cryo-CMOS Real-Time Hyperdimensional Decoder for Quantum Error Correction with Current-Based In-Memory Multi-Bit XOR

Rob A. Damsteegt*, Masoud Babaie†, Sebastian Feld*, Fabio Sebastiano*
*Department of Quantum & Computer Engineering & QuTech, TU Delft, The Netherlands
†Department of Microelectronics & QuTech, TU Delft, The Netherlands
Email: r.a.damsteegt@tudelft.nl

***Abstract*—Practical quantum computers (QC) rely on quantum error correction to manage unavoidable physical errors, thus requiring an accurate and fast decoder. This work proposes a low-power hyperdimensional-computing decoder ASIC capable of operating at cryogenic temperatures near the qubits, eliminating long-range high-speed interconnects by adopting local processing. The many multi-bit XOR operations at the heart of the decoder are implemented in-memory by differential current mixing. The decoder's low latency (284 ns) and low power (1.15 mW for a 97-qubit code) pave the way for integration into future error-corrected cryogenic QCs with stringent power constraints.**

***Index Terms*—cryo-CMOS, quantum error correction, decoder, in-memory, hyperdimensional computing**

I. Introduction

Large-scale universal quantum computers require high-fidelity quantum bits (qubits) with error rates orders of magnitude lower than those achievable in practical physical implementations. To improve the error rates, quantum error correction (QEC) schemes [1], Fig. 1, combine multiple physical "data" qubits to form a single logical qubit. Errors on individual data qubits can be detected with periodic parity measurements through additional "stabilizer" qubits. Every QEC cycle, the measured parity information (syndrome) is interpreted by a QEC decoder to find the correction on the logical qubit that is necessary to compensate for data-qubit errors. As a result, the logical error rate (LER) is much lower than the physical error rates of the physical qubits.

To keep up with the parity data generation, the decoders' average processing time should be shorter than t_{cycle}, typically assumed to be 1 µs [2]. Decoders running on CPUs/GPUs can have unpredictable latencies due to scheduling and memory-access variations on top of the syndrome-dependent latency of algorithmic decoders [3]. For hardware implementations, algorithmic decoders suffer from the same latency variability [4], and neural networks (NNs) are difficult to scale and train for large QEC codes [5]. Additionally, these decoders all operate at room temperature (RT), which requires significant data to be transferred out of the qubit cryostat. By operating the decoder in the cryogenic environment, such a wiring data-transfer bottleneck is removed, but the power budget is limited. To address those requirements, a low-power cryo-CMOS decoder is presented that is based on hyperdimensional computing (HDC) and implements the main bit-wise XOR operation in-memory using current mixing to save area and power. The HDC decoder is accurate, has constant decoding latency, and is resilient to CMOS hardware errors, enabling error-prone low-power techniques.

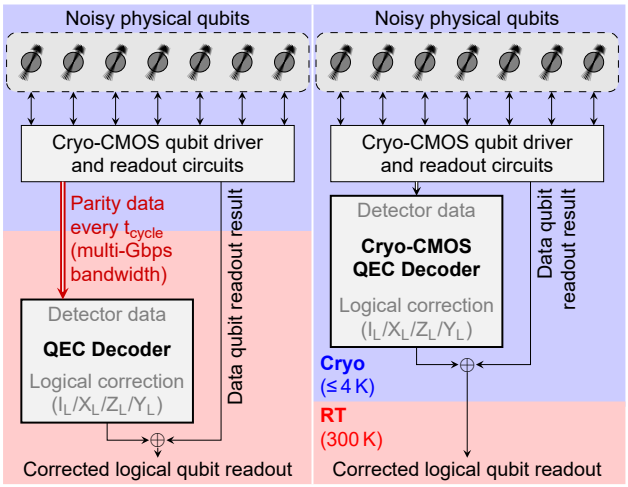

Fig. 1. Left: QEC setup with cryo-CMOS interface electronics and room-temperature decoder. Right: envisioned setup with a cryo-CMOS decoder tightly integrated with the interface electronics.

II. Circuit Description

As shown in Fig. 2, the distance-7 surface code generates 48 detector bits (DET_i) every QEC cycle (t_{cycle}). In this work, we consider a 'depolarizing noise' model where an X, Y, or Z error occurs on each data qubit, each with probability $p/3$, while no error occurs with probability $1-p$ [1]. Each detector bit is mapped to a different 5120-bit hyper-vector (HV), optimized after [6]. The HVs belonging to the detector bits that are "1" are selected (AND gates) and combined bit-wise through multi-bit XOR operations. The resulting syndrome HV is compared with four class HVs, which represent the possible logical corrections, using the Hamming Distance (HD, the number of bits that differ). The most similar class (lowest HD) indicates which logical correction to choose.

The main computation, i.e., the bit-wise selection and XOR of the detector HVs, adopts the circuit shown in Fig. 3. The

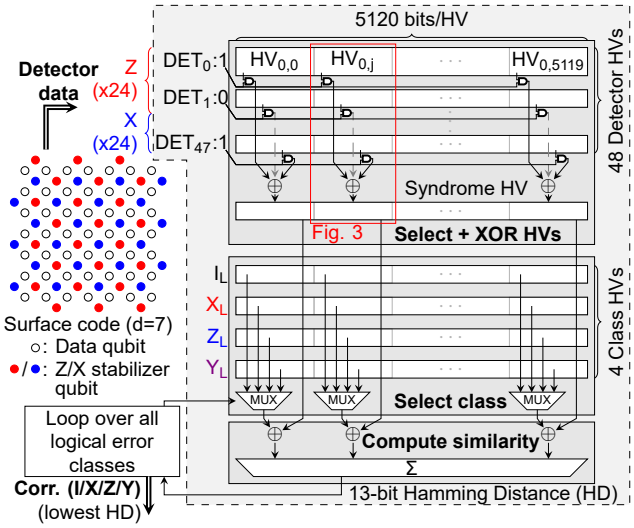

Fig. 2. Overview of the implemented HDC-based decoder for a distance-7 surface-code memory experiment.

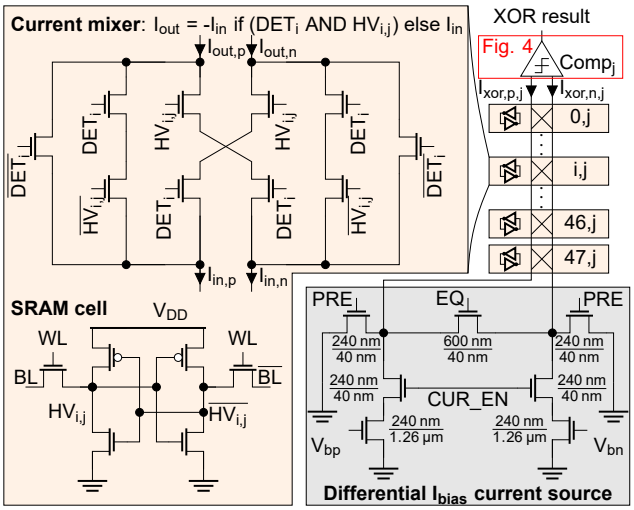

Fig. 3. Schematic of the IXOR for HV bit j. All unlabeled transistors are minimum size: $\frac{W}{L} = \frac{120\,\text{nm}}{40\,\text{nm}}$.

XOR is evaluated in the current domain (IXOR) by swapping the polarity of a differential current I_{bias} through a mixer only if the detector bit and the HV bit are set (equal to 1), thereby implementing the selection functionality. The HV bit is stored in an SRAM bit cell close to the mixer, resulting in a compact compute-in-memory (CIM) cell implementing storage, selection, and the XOR operation that is around 30 % smaller than a standard logic XOR cell. A single multi-bit IXOR consists of a differential current source (DCS) for generating I_{bias} and a stack of 48 CIM cells.

The polarity of the output current is measured by the two-stage current comparator shown in Fig. 4. Since the first

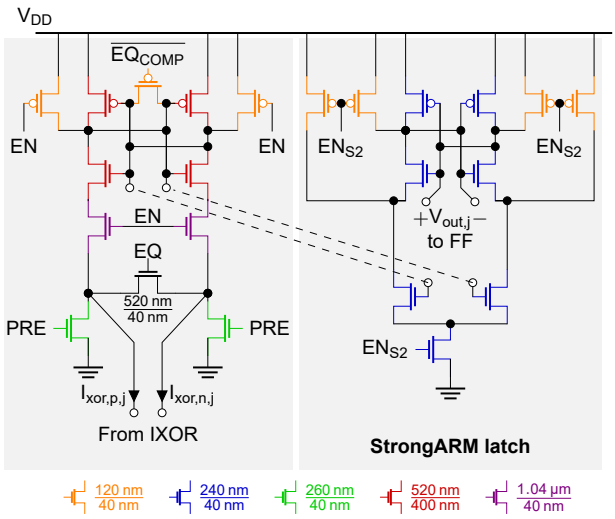

Fig. 4. Schematic of the two-stage current comparator. Unlabeled transistor dimensions are indicated by the transistor color (legend at the bottom).

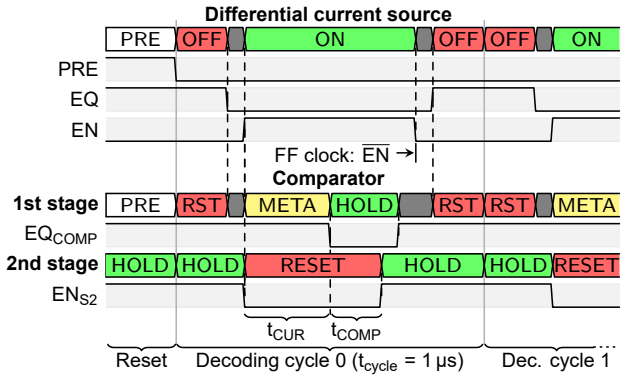

Fig. 5. Control signal timing diagram for the IXOR operation. Looping over the classes for classification at the end of a cycle is not shown.

current-input stage has a limited output swing, a StrongARM latch as the second stage recovers a rail-to-rail output swing.

The timing of the current source and comparator stages is shown in Fig. 5. During the reset phase, both sides of the CIM stack are precharged (PRE) to ground. A decoding cycle starts by enabling the DCS, connecting the first comparator stage to the top of the CIM stack (EN) while keeping it in a metastable state by shorting the latch (EQ_{COMP}), and resetting the second comparator stage (EN_{S2}). After letting the current transients settle during the "current time" (t_{CUR}), the first comparator stage latch is released from its metastable condition and $I_{\text{xor,p/n,j}}$ causes an output voltage to develop. After the "comparator time" (t_{COMP}), the second comparator stage latches and amplifies the output voltage. The comparator output $V_{\text{out,j}}$ is then captured by a standard logic flip-flop (FF) cell that completes the XOR calculation. This process is repeated for each decoding cycle without intermediate resets,

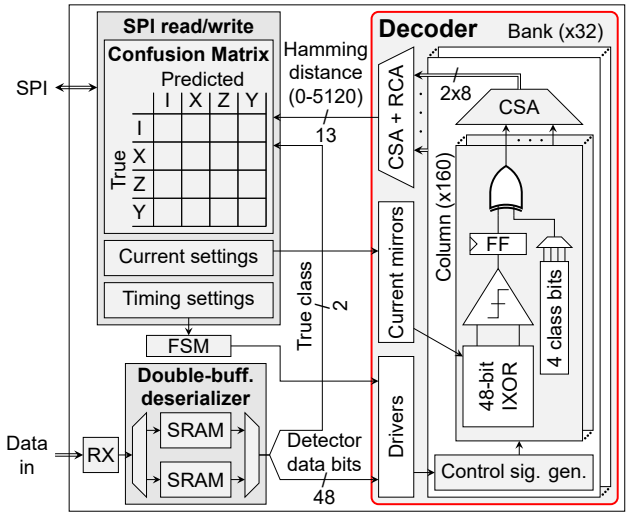

Fig. 6. Overview of the chip architecture, including the test hardware (left) and the hierarchical decoder design (right).

Technology	TSMC 40 nm	
Chip size	1350 x 1395 µm²	
Decoder size	840 x 730 µm²	
Supply voltage	1.1 V	
Detector count	48	
Class count	4	
HV width	5120 bits	
Temperature	300 K	4.2 K
Power (t_{cycle} = 1 µs)	1.1 mW	1.15 mW
Decoding time	245 ns	284 ns

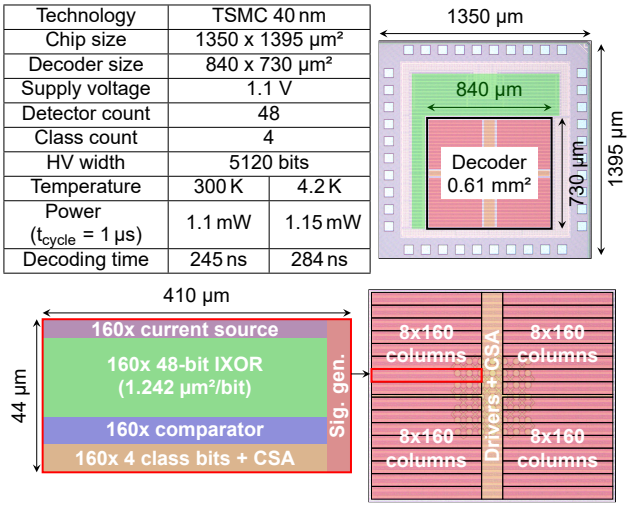

Fig. 7. Performance table, chip micrograph, and physical chip organization.

reducing the power consumption by sharing the charge in the CIM stack through the equalization transistors (EQ) and providing the bias point for the next decoding cycle.

The full chip architecture is shown in Fig. 6. The decoder (right) consists of 32 banks, each having 160 columns for a total of 5120 columns, where each column comprises a single multi-bit IXOR, a FF to store the result, four bits corresponding to the respective bit of each of the four class vectors, and an XOR standard cell to determine the bit similarity. The results of the 160 columns are combined using a bank-level carry-save adder (CSA), and, in turn, the 32 bank outputs are combined through a global CSA and ripple-carry adder (RCA) into a single 13-bit HD. To enable real-time testing, the chip includes programmable test hardware (left), which deserializes and double buffers an NRZ data stream of detector bits and expected classification results, sets the decoder programmable timing/biasing, and constructs a downloadable confusion matrix with the decoder's classification performance.

III. Measurement Results

The design has been fabricated in TSMC 40-nm CMOS technology (Fig. 7) and is measured at real-time speed (t_{cycle} = 1 µs) at both room and cryogenic temperatures using a dipstick setup. When comparing the number of hardware misclassifications with respect to the decoder software implementation (Fig. 8), higher bias currents and longer comparator times reduce the misclassification rate. As long as this misclassification rate is significantly below the decoder LER, the final LER is unaffected, thus allowing power minimization by choosing a low I_{bias} and short IXOR duty-cycle. At cryogenic temperatures, the bias currents need to be increased to avoid the high-mismatch subthreshold operating regime [7]. The actual decoding accuracy in Fig. 9 shows a graceful-degradation behavior for similar settings as in Fig. 8, as errors

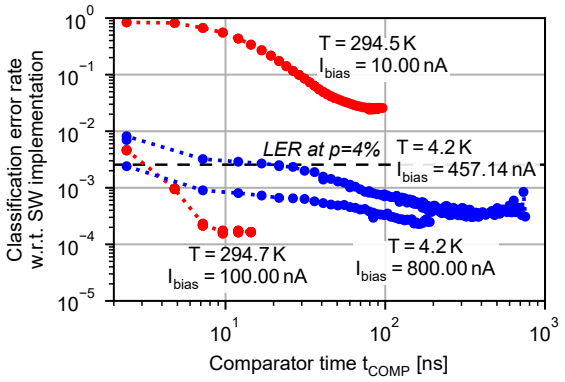

Fig. 8. The rate at which the hardware gives a different logical classification than the software implementation of the decoder as a function of the comparator enable duration (t_{COMP}) for t_{CUR} = 79.33 ns and two I_{bias} levels at room and cryogenic temperatures. Syndrome data for a physical error rate of $p = 4\,\%$ is used since it is sufficiently below the code's threshold while still giving easily measurable error rates. The logical error rate (LER) of the software implementation is indicated by the dashed line.

in the XOR evaluations affect only individual HV bits, thus demonstrating the robustness of the HDC decoder. Since lower bias currents, and hence a lower power, result only in a slightly worse LER, this allows for a gradual tradeoff between performance and power consumption. Fig. 10 shows that the hardware LER is only 2 % higher than the software LER, but up to 80 % lower than the typically adopted benchmark, i.e., the minimum-weight perfect matching (MWPM) algorithm [3] for low physical error probabilities.

Compared to prior hardware decoders in Table I, this work demonstrates the first ASIC implementation, while achieving the lowest latency, in addition to cryogenic operation and a low power for the full decoder.

TABLE I
COMPARISON WITH STATE-OF-THE-ART HARDWARE QEC DECODERS.

	QUNET [8]	BP OSD [9]	GNN [10]	Pinball [11]	This work	
Impl./Tech.	FPGA	FPGA	FPGA	Simulated ASIC (22 nm FDSOI)	ASIC (40 nm bulk)	
Decoder type	NN	Algorithm (BP OSD)	GNN	Predecoder (needs a global decoder)	HDC	
Arch.	Digital	Digital	Digital	Digital	In-memory	
SC distance	7	9	7	21	7	
Error model	Circuit noise	Circuit noise	Circuit noise	Circuit noise	Depolarizing noise	
LER	Worse than MWPM	Better than MWPM	Better than MWPM	N/A	Better than MWPM	
Temp.	300 K	300 K	300 K	4 K	300 K	4 K
Power	400 mW	N/A	N/A	0.56 mW[a]	1.1 mW	1.15 mW
Latency	387 µs	134 µs	989 ns	<1 µs	245 ns	284 ns

[a]: predecoder only, simulated

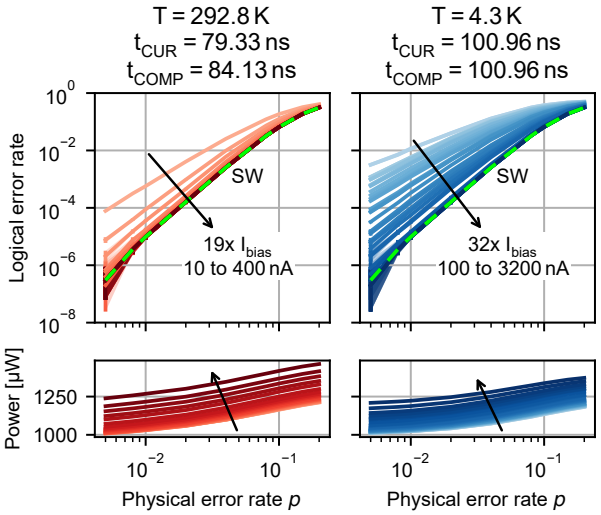

Fig. 9. Hardware decoder logical error rate (LER) and power consumption of the decoder as a function of the physical error rate p at room (left) and cryogenic (right) temperature environments for various I_{bias} compared to the software (SW) implementation (green dashed line).

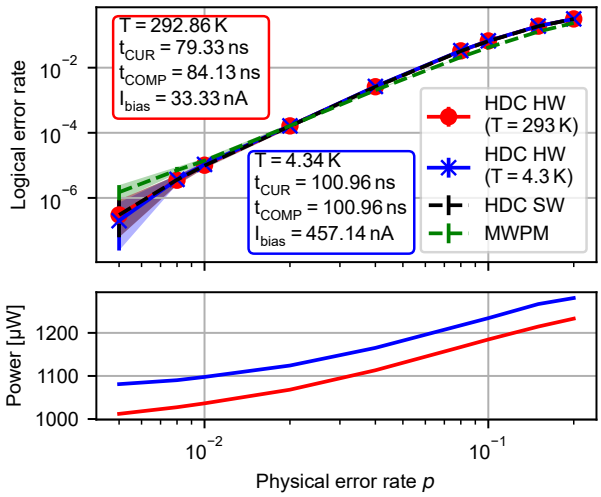

Fig. 10. Logical error rate (LER) and power consumption of the hardware decoder (HDC HW) at room and cryogenic temperatures as a function of the physical error rate p versus a software implementation of the decoder (HDC SW) and a minimum-weight perfect matching decoder (pymatching, MWPM) [3].

IV. CONCLUSIONS

This work demonstrates the first cryo-CMOS real-time decoder, simultaneously achieving low power (1.15 mW), high speed (284 ns), and high accuracy with a moderate silicon footprint thanks to the HDC architecture and its current-based CIM implementation. As a basis for scaling to larger codes and more advanced noise models, it paves a promising path towards fully integrated error-corrected quantum systems.

REFERENCES

[1] B. M. Terhal, "Quantum error correction for quantum memories," *Rev. Mod. Phys.*, vol. 87, pp. 307–346, Apr 2015.

[2] F. Battistel *et al.*, "Real-time decoding for fault-tolerant quantum computing: progress, challenges and outlook," *Nano Futures*, vol. 7, no. 3, p. 032003, Aug 2023.

[3] O. Higgott and C. Gidney, "Sparse blossom: correcting a million errors per core second with minimum-weight matching," *Quantum*, vol. 9, p. 1600, 2025.

[4] N. Liyanage *et al.*, "Scalable quantum error correction for surface codes using FPGA," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 01, 2023, pp. 916–927.

[5] R. W. J. Overwater *et al.*, "Neural-network decoders for quantum error correction using surface codes: A space exploration of the hardware cost-performance tradeoffs," *IEEE Transactions on Quantum Engineering*, vol. 3, pp. 1–19, 2022.

[6] R. A. Damsteegt *et al.*, "A scalable real-time decoder for quantum error correction based on hyperdimensional computing," in *2025 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 01, 2025, pp. 1076–1082.

[7] P. A. 't Hart *et al.*, "Subthreshold mismatch in nanometer CMOS at cryogenic temperatures," *IEEE Journal of the Electron Devices Society*, vol. 8, pp. 797–806, 2020.

[8] R. Huo and J. Nunez-Yanez, "FPGA-accelerated early-exit neural decoder for quantum error correction," in *2025 28th Euromicro Conference on Digital System Design (DSD)*, 2025, pp. 616–624.

[9] D. Báscones *et al.*, "Exploring the FPGA and ASIC design space of belief propagation and ordered statistics decoders for quantum error correction codes," *EPJ Quantum Technology*, vol. 12, no. 1, p. 140, Nov 2025.

[10] A. Cicero *et al.*, "Low latency GNN accelerator for quantum error correction," 2026. [Online]. Available: https://arxiv.org/abs/2603.22149

[11] A. Knapen *et al.*, "Pinball: A cryogenic predecoder for quantum error correction decoding under circuit-level noise," in *2026 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 2026, pp. 1–17.