CONDITIONAL DYNAMICAL SYSTEMS FOR IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Image generation is dominated by deep generative models on GPUs, whose computational and energy costs raise sustainability concerns. Emerging non-von Neumann substrates, including quantum, compute-in-memory, photonic, and thermodynamic platforms, promise greater efficiency, yet most work ports conventional neural network architectures onto them and mainly accelerates operations such as matrix multiplication. This does not fully exploit a native capability of many hardware substrates: relaxation toward low-energy states can itself compute at little cost. Building on this primitive, we develop dynamical systems for image generation based on Ising spins and Kuramoto oscillators. The internal states of both models evolve under dynamics with an explicit Lyapunov energy, and a compact decoder renders the final states as images. For conditional generation, we introduce local conditioning: programmed interactions remain fixed, and one class-dependent bias per node reshapes the landscape, avoiding additional reprogramming and calibration costs. On CIFAR-10, the Ising and Kuramoto models reach clean-FID scores of 6.40 and 5.37, respectively, with 4096 nodes. With a CMOS-compatible hardware realization, the dynamical-system core is projected to cost ~ 110.6 nJ per image, and the full computation, even with the decoder on a GPU, outperforms all GPU baselines in latency and energy. These results suggest that energy-descending dynamics on non-von Neumann substrates can serve as generative computation, offering an efficient path beyond GPUs. Our code is available at `https://anonymous.4open.science/r/CF9E`.

1 INTRODUCTION

Image generation is a central task in machine learning, with applications in content creation, data augmentation, and design. Generative adversarial networks, diffusion models, and flow-matching models have achieved high generation quality (Goodfellow et al., 2014; Ho et al., 2020; Lipman et al., 2023). Their success, however, relies on increasingly capable GPU infrastructure. Generation requires substantial computation and energy, motivating interest in substrates whose native physical behavior differs from that of conventional processors (Strubell et al., 2019; Patterson et al., 2021).

Several classes of non-von Neumann substrates have been explored. Quantum annealers and Ising machines encode an objective in physical interactions and evolve toward low-energy configurations (Johnson et al., 2011; Moy et al., 2022; Zhang et al., 2024). Analog compute-in-memory systems based on memristive crossbars or phase-change memory perform matrix operations where parameters are stored, reducing data-movement cost (Ielmini & Wong, 2018; Sebastian et al., 2020; Mannocci et al., 2026). Photonic processors exploit optical propagation and interference for ultrafast computation (Shen et al., 2017; Kalinin et al., 2025), while thermodynamic, spintronic, and stochastic hardware uses physical fluctuations as a native sampling resource for computation (Borders et al., 2019; Melanson et al., 2025).

For machine learning, these substrates are mostly used as accelerators: the neural network keeps its GPU-oriented architecture, and selected operations are offloaded to specialized devices (Shen et al., 2017; Ambrogio et al., 2023; Oguz et al., 2024). This approach can be effective, but it does not exploit the native physical primitive, which often runs at negligible energy and latency cost. Offloading may also incur overhead and lose accuracy to programming error, quantization, parasitics, and noise, reducing part of the efficiency gain (Xiao et al., 2022a).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We take a different direction: designing the model around the hardware substrate's computational primitives (Momeni et al., 2025; Bösch et al., 2026; Whitelam, 2026; Jelinčič et al., 2026). An initial demonstration shows that a coupled-oscillator system can produce nontrivial image distributions (Unconventional AI, 2026). Beyond this first step, designing and realizing generative models for such substrates raises two challenges. First, what computation should the model perform so that existing hardware can run it natively? Second, how should the image class be introduced when the interaction array is written to physical devices and is expensive to change?

We address both challenges with one design principle: the substrate dynamics descend an explicit energy, and conditioning reshapes this energy landscape instead of reprogramming the interactions. Figure 1 illustrates the idea: when the class *cat* is selected, the energy landscape attracts the state toward cats and repels it from horses. Generation becomes energy relaxation, the primitive that the hardware performs natively at little cost. For conditional generation, we introduce *local conditioning*: all pairwise couplings are fixed, and class information enters only through one local bias per node, which tilts the shared landscape. Switching the class updates only these values and leaves the programmed interaction array untouched. This avoids reprogramming and calibration costs, which could otherwise dominate the cost of generation.

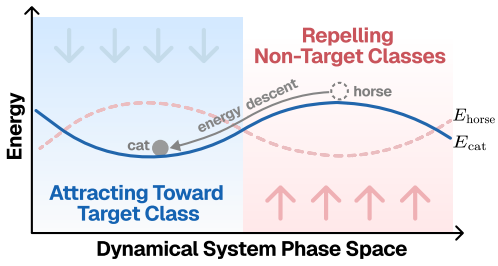


Figure 1: Generation by energy descent. Local conditioning reshapes the landscape, attracting the evolving state toward the target class.

The design principle is not limited to one substrate. We study both continuous Ising dynamics and coupled Kuramoto oscillators and map them to Ising machines. Both descend a learned energy landscape with the local conditioning mechanism, and a compact convolutional decoder turns the final state into an image. The energy is defined over the internal state rather than pixels: energy descent produces the representation the decoder reads.

Many existing Ising machines, quantum and digital annealers, and thermodynamic, memristive, and photonic substrates already provide most of the primitives our design requires. Our results suggest that such systems could gain generative capability with modest architectural changes. Several of these substrates are CMOS-compatible, so the same dynamics could also be built as a dedicated generative core inside a GPU or as a standalone chip. For now, our goal is not to replace mature GPU-native generators, whose parameter budgets and primitives differ. Instead, we ask whether hardware built for energy minimization can generate images with its native relaxation dynamics, and at what cost. Our contributions are:

- We formulate conditional image generation as dynamics that descend an explicit Lyapunov energy, with two models: continuous Ising spins and coupled Kuramoto oscillators.
- We introduce local conditioning, which selects the class through one constant bias per node while the pairwise interactions stay fixed, avoiding reprogramming and calibration costs.
- With $n = 4096$ nodes, the Ising and Kuramoto models reach clean-FID scores of 6.40 and 5.37 on CIFAR-10, respectively, outperforming existing substrate-native baselines.
- We present a case study of a CMOS-compatible hardware design. The dynamical-system core is projected to cost $\sim$110.6 nJ per image. With a compute-in-memory decoder accelerator, the projected energy and latency of the full computation are 5.9 µJ and 57.4 µs per image, respectively, far below those of GPU-based generators.

2 Method

We first formulate our model (Section 2.1) and its two dynamics: continuous Ising spins (Section 2.2) and Kuramoto oscillators (Section 2.3). Next, we present our local conditioning mechanism, which enables conditional generation while keeping the pairwise couplings fixed across classes (Section 2.4). Finally, we describe the training procedure (Section 2.5) and show how the resulting model maps onto physical hardware (Section 2.6).

2.1 Model Overview

The generator maps one of C class labels, $y \in \{1, \dots, C\}$, and a random initial state ξ to an image,

$$G(y, \xi) = \mathrm{Dec}_\psi(\Phi_\phi(y, \xi)), \tag{1}$$

where Φ_ϕ returns the terminal state of a physical evolution and Dec_ψ is a convolutional decoder. The random initial state ξ is the source of sample diversity. We consider two choices for Φ_ϕ: continuous Ising spins and coupled phase oscillators. In both cases, a pairwise interaction is shared by all classes, while one bias per node selects y.

2.2 Continuous Ising Dynamics

The continuous-spin model begins from the binary Ising energy (Ising, 1925)

$$E_{\mathrm{Ising}}(s) = -\frac{1}{2}\sum_{i \neq j} J_{ij} s_i s_j - \sum_i b_i s_i, \qquad J_{ij} = J_{ji}, \quad J_{ii} = 0, \tag{2}$$

where the spins are discrete, $s \in \{-1, +1\}^n$, J contains pairwise couplings, and b contains the coefficients of the linear term. Quadratic unconstrained binary optimization problems map directly to this energy (Lucas, 2014). In Max-Cut, for example, negated edge weights become couplings, and the lowest-energy state is the best cut. An Ising machine solves such problems physically: its state settles toward low energy, and the final configuration is a good solution.

We let each spin take a continuous value, $x \in \mathcal{X} = [-1, 1]^n$, as in most analog Ising machines (Yamamoto et al., 2017; Afoakwa et al., 2021). We define the system energy as

$$E^{\mathrm{I}}_y(x) = -\frac{1}{2} x^\top A x - \gamma b_y^\top x, \qquad A = A^\top \in \mathbb{R}^{n \times n}, \quad b_y \in \mathbb{R}^n. \tag{3}$$

The vertices of $\mathcal{X}$ are binary Ising configurations. The interaction A, which extends J with a diagonal, is shared across classes. Here b_y is the local conditioning vector of class y, and the fixed scalar γ sets its strength.

Starting from $x(0) = \xi \sim \mathcal{U}(\mathcal{X})$, the relaxation is

$$\dot{x} = -\nabla_x E^{\mathrm{I}}_y(x) = Ax + \gamma b_y, \tag{4}$$

so the energy descends,

$$\frac{d}{dt} E^{\mathrm{I}}_y(x(t)) = -\|\nabla_x E^{\mathrm{I}}_y(x(t))\|^2 \leq 0. \tag{5}$$

Each spin is clipped at ± 1, as a node voltage saturates at the supply rails of an analog Ising machine. This only removes terms from Eq. (5), so E^{I}_y remains non-increasing. The decoder reads $x(T)$ at a fixed horizon T and produces the final image.

2.3 Kuramoto Oscillator Dynamics

The Kuramoto oscillator model uses n coupled phases $\theta \in \mathbb{T}^n$ (Kuramoto, 1975). We follow the oscillator-based Ising machine formulation (Wang & Roychowdhury, 2019). Starting from independent random uniform phases, the oscillator dynamics are

$$\theta_i(0) \sim \mathcal{U}([0, 2\pi)), \qquad \dot{\theta}_i = \sum_j K_{ij} \sin(\theta_j - \theta_i) - b_{y,i} \sin \theta_i, \tag{6}$$

where $K = K^\top \in \mathbb{R}^{n \times n}$ is the learned, class-shared coupling and $b_y \in \mathbb{R}^n$ is the local conditioning vector, which absorbs the strength γ. The final term couples oscillator i to a reference phase fixed at zero. Since K is symmetric, Eq. (6) is the descent of a Lyapunov energy,

$$E^{\mathrm{K}}_y(\theta) = -\frac{1}{2}\sum_{i \neq j} K_{ij} \cos(\theta_j - \theta_i) - \sum_i b_{y,i} \cos \theta_i, \tag{7}$$

because $\dot{\theta} = -\nabla_\theta E^{\mathrm{K}}_y$. Consequently,

$$\frac{d}{dt} E^{\mathrm{K}}_y(\theta(t)) = -\|\nabla_\theta E^{\mathrm{K}}_y(\theta(t))\|^2 \leq 0. \tag{8}$$

Oscillator-based Ising machines perform this descent for optimization problems, and we apply it to generation. We stop the dynamics at a fixed horizon T. The final state $[\cos \theta(T), \sin \theta(T)]$ is fed to the decoder.

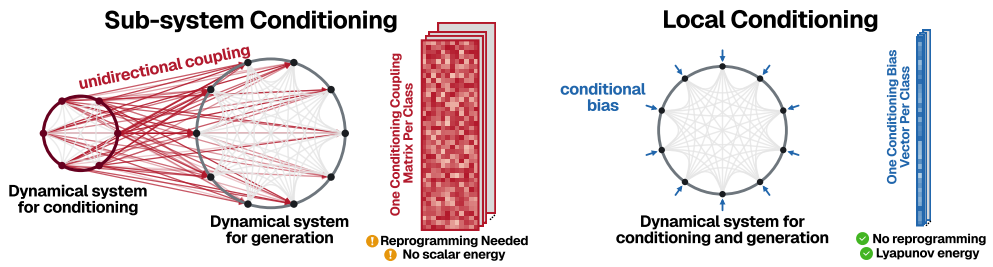


Figure 2: Comparison of conditioning models. Sub-system conditioning adds an auxiliary system and class-specific unidirectional couplings. Each class needs its own coupling, so reprogramming is required, and the dynamics generally do not descend a scalar energy. Our approach instead applies one bias per node for each class, keeping a fixed pairwise interaction and a Lyapunov energy.

2.4 Local Conditioning Mechanism

One naive way to build conditional generative models from hardware primitives is to train one model per class. On a physical substrate, this choice is costly: each model has its own coupling matrix, so one model per class means either one chip per class or rewriting the full interaction in the hardware (e.g., memristive or photonic arrays) at every class switch. Prior work (Unconventional AI, 2026) reduces the changes to a subset of the interactions (Figure 2, left). It evolves a separate group of conditioning oscillators alongside the n oscillators for generation and connects the two systems with a class-dependent, unidirectional coupling. This design has several drawbacks: it adds dynamical states, requires coupling reprogramming, and generally lacks the energy-descent property favored by many hardware substrates. Reprogramming changes what the substrate was built to hold fixed and adds recalibration cost. This leaves a gap between models and hardware.

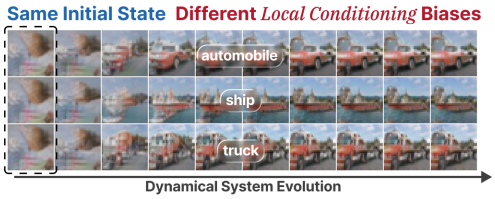


Figure 3: One random initial state evolves under different local conditioning biases, producing images of different classes.

We close this gap by moving the class conditioning out of the interaction and into the local term of each node (Figure 2, right). The pairwise interaction is programmed once and never changed, and no auxiliary system is added. Selecting class y injects a term into each node i: the bias $\gamma b_{y,i}$ in the Ising model and the reference coupling $b_{y,i}$ in the Kuramoto model. The mechanism has a direct interpretation: the interaction defines an energy landscape shared by all classes, and the conditioning vector tilts it. Every class descends the same landscape under a different tilt, so the Lyapunov property, the programmed hardware, and its calibration are all reused. The label enters the energies of Eq. (3) and Eq. (7) only through the local term. For the same state, x or θ, and two labels y and y', the shared interaction A or K cancels in the energy difference:

$$\begin{aligned} E^{\mathrm{I}}_y(x) - E^{\mathrm{I}}_{y'}(x) &= -\gamma(b_y - b_{y'})^\top x, \\ E^{\mathrm{K}}_y(\theta) - E^{\mathrm{K}}_{y'}(\theta) &= -\sum_i (b_{y,i} - b_{y',i}) \cos\theta_i. \end{aligned} \tag{9}$$

Only the biases remain. Figure 3 shows the effect of the conditioning mechanism. Starting from one random initial state, we run the dynamics with the biases of three classes: automobile, ship, and truck. As the dynamical-system state evolves, its decoded image settles to the selected class.

2.5 Training

We focus on efficient inference on the substrate. Model training is performed only once on conventional hardware. We train the dynamics and the decoder jointly using an objective that aligns generated and real feature distributions of each class in frozen DINOv2 feature spaces (Oquab et al.,

Table 1: Results on CIFAR-10, grouped by system size n. † denotes reported results. Our rows show the best of three seeds. **Reprog.**: requires reprogramming at each class switch. **Dec. Params**: Decoder parameters. **IS**: Inception Score.

Model	Reprog.	Total Params	Dec. Params	IS ↑	clean-FID ↓
Un-0.n1024†	Yes	1.29M	0.16M	9.84	11.16
Ising-1024	No	0.60M	0.07M	9.32	11.57
Kuramoto-1024	No	0.69M	0.16M	8.99	**10.86**
Un-0.n2048†	Yes	4.94M	0.58M	10.23	9.38
Ising-2048	No	2.28M	0.16M	10.01	**7.62**
Kuramoto-2048	No	2.70M	0.58M	9.94	8.26
Un-0.n4096†	Yes	19.43M	2.33M	10.40	8.86
Ising-4096	No	9.01M	0.58M	10.12	6.40
Kuramoto-4096	No	10.75M	2.33M	10.34	**5.37**

2024), which encourages both image fidelity and within-class diversity. Following the drifting objective of Deng et al. (2026), we match generated features toward targets constructed from Sinkhorn displacements with a self-matching correction (He et al., 2026), motivated by the debiased Sinkhorn divergence (Genevay et al., 2018; Feydy et al., 2019). More details are provided in Appendix A.

2.6 Non-von Neumann Hardware Realization

Our dynamics build on primitives found in many non-von Neumann substrates: relaxation, programmable pairwise couplings, and per-node biases. We take Ising machines as an example. For the Kuramoto realization, oscillator-based Ising machines implement the pairwise term of Eq. (6). Coupling each oscillator to a shared reference adds the local conditioning mechanism, as CMOS oscillator chips already do (Lo et al., 2023). For the Ising realization, resistively coupled electronic Ising machines (Liu et al., 2025e) store x as capacitor voltages, and programmable couplers inject currents proportional to Ax that charge each capacitor. Per-node bias inputs supply γb_y for local conditioning. Section 4 details this design. Furthermore, photonic (Inagaki et al., 2016; Kalinin et al., 2025), compute-in-memory (Cai et al., 2020; Jiang et al., 2023), thermodynamic (Melanson et al., 2025), and superconducting (Razmkhah et al., 2024) substrates already offer most of these primitives, making them natural candidates as well.

3 Evaluation

Because the proposed model is not tied to a single physical substrate, we separate model-level capability from implementation-level performance. At the model level (Section 3.1), we compare against substrate-native models. At the implementation level (Section 3.2), we pair our model with a concrete hardware design and compare its energy and latency against conventional models on GPUs.

3.1 Model-Level Capability

Prior non-von Neumann substrates for generative tasks have mainly been evaluated on simple benchmarks such as MNIST (Lecun et al., 1998). On CIFAR-10 (Krizhevsky & Hinton, 2009), the closest baseline is Un-0 (Unconventional AI, 2026), which uses coupled oscillators with varied frequencies and changes the cross-system interactions for class conditioning. We compare the proposed Ising and Kuramoto models with Un-0 at system sizes $n \in \{1024, 2048, 4096\}$.

Results. Table 1 reports the comparison. Regarding image generation quality, our models match the baseline at $n = 1024$, and the margin grows with system size. At $n = 4096$, both lower clean-FID (Parmar et al., 2022) by more than 2 while keeping a competitive Inception Score (IS) (Salimans et al., 2016). In terms of modeling, our Kuramoto dynamics do not require varied natural frequencies, which eases hardware implementation, and both proposed models descend an explicit Lyapunov energy. For conditioning, the baseline reprograms cross-system couplings at every class switch, whereas our local conditioning updates only one bias per node. These benefits come without

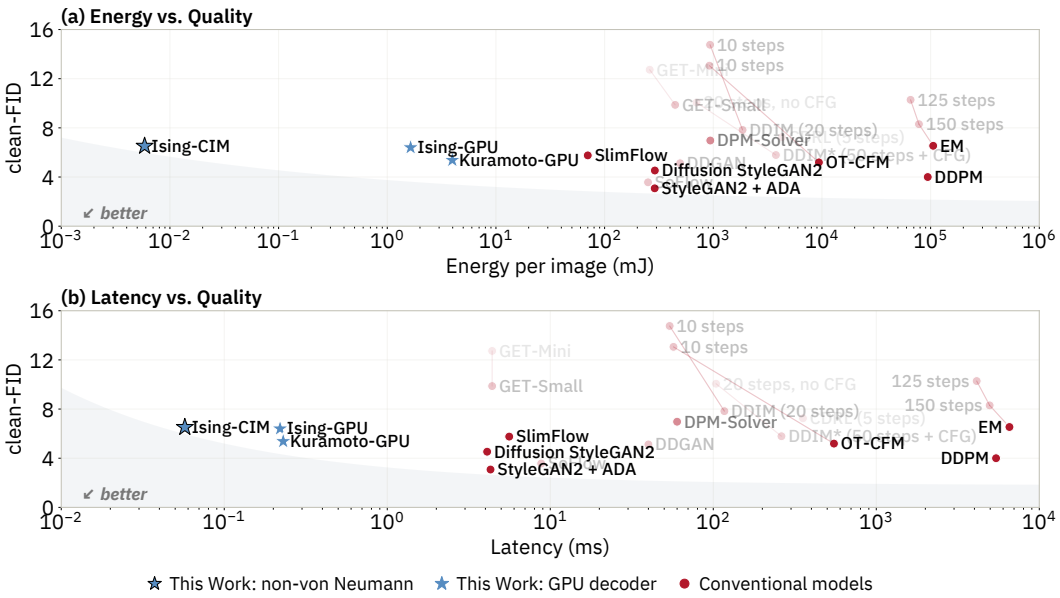


Figure 4: (a) Energy per generated image vs. clean-FID. (b) Latency vs. clean-FID. Some data points are faded for better visualization.

sacrificing image quality, which even improves at larger n. The proposed approach thus narrows the gap between generative models and hardware realization and requires only minor modifications to existing substrates.

3.2 Implementation-Level Performance

To evaluate implementation-level efficiency, we pair the proposed model with a concrete hardware design (Section 4) and compare its performance with models running on GPUs. Our data are projected from post-layout transistor-level simulation results of a CMOS-compatible dynamical system. Appendix D provides the circuit details.

Baselines. We compare against the conventional generators shown in Figure 4: DDPM (Ho et al., 2020), DDIM (Song et al., 2021), DPM-Solver (Lu et al., 2022), OT-CFM (Tong et al., 2024), CDRL (Zhu et al., 2024a), Energy Matching (Balcerak et al., 2025), DDGAN (Xiao et al., 2022b), SlimFlow (Zhu et al., 2024b), SoFlow (Luo et al., 2026), GET (Geng et al., 2023), Diffusion-StyleGAN2 (Wang et al., 2023), and StyleGAN2-ADA (Karras et al., 2020). We additionally train a class-conditional DDPM U-Net whose parameter count approximately matches that of Ising-4096 and sample it with DDIM (DDIM*). All GPU measurements use an NVIDIA L40S GPU.

Our models. Based on projections from circuit simulations (Section 4 and Appendix C), the Ising-4096 core draws 1.37–8.05 W and is read out after 10.1–80.8 ns, costing $\sim$110.6 nJ per image. Relaxation costs orders of magnitude less than GPU inference. The reason is physical: the substrate itself carries out the computation, with no clocked arithmetic and no data movement between memory and processor (Zhang et al., 2024). The Kuramoto model behaves similarly (Appendix C). Local conditioning maintains this advantage across classes. The compact decoder is far smaller than every baseline model and can run on GPUs or dedicated accelerators.

Results. Figure 4(a) compares the energy cost per generated image against clean-FID. When the decoder is mapped onto compute-in-memory (CIM) hardware, the Ising-CIM (4096 spins) is located at the extreme low-energy end of the Pareto frontier, requiring only 5.9 µJ per image. Under decoder quantization constraints, it achieves a clean-FID of 6.52. With the decoder executed on a GPU, the resulting Ising-GPU and Kuramoto-GPU (4096 nodes) realizations consume energy at the millijoule scale, still orders of magnitude less than most GPU baselines. For the latency comparison in Figure 4(b), our implementations operate at sub-millisecond latency, more than an order of

magnitude faster than the GPU baselines. Overall, our models deliver strong generation quality with substantially lower energy and latency than GPU-based generators.

3.3 Ablation Study

To measure what the learned dynamical system adds beyond the decoder, we compare the full Ising and Kuramoto models at $n = 4096$ with two variants. **Reservoir** keeps the dynamical-system evolution but freezes the coupling matrix at its random initialization, and only the local conditioning vectors and decoder are trained. **Decoder Only** removes the dynamical system and feeds the decoder a random state with a learned class-specific offset, preserving conditional generation.

For each model, all variants use the same decoder architecture, per-seed initialization of common components, and the same full training schedule. We report the mean clean-FID over three random seeds. Table 2 shows that the learned dynamical system improves generation quality in both families. **Reservoir** and **Decoder Only** increase clean-FID over the full models by 4.6 to 11.7, with larger gaps for Ising, even though these variants still train their conditioning vectors and decoders.

Table 2: Contribution of the dynamical system to image generation quality. Mean clean-FID (↓) over three seeds.

Variant	Kuramoto-4096	Ising-4096
Full	5.41	6.53
Reservoir	10.02	13.96
Decoder Only	10.14	18.19

4 Case Study: A CMOS-Compatible Dynamical System

4.1 Hardware Design

System architecture. The proposed principle applies to many emerging hardware substrates. As a concrete example, we map the Ising model onto the core of the CMOS-compatible Ising machine of Liu et al. (2025e). As shown in Figure 5(a), programmable resistive couplers connect the spins. The analog voltage on each spin capacitor drives this network through a unity-gain buffer. The system state evolves continuously. To support the local conditioning mechanism, a digitally controlled bias current is applied to each spin. The hardware therefore encodes the energy defined in Eq. (3). A scheduler coordinates the operation. After initialization, the system starts to evolve along an energy landscape defined by the coupling coefficients and reshaped by the bias currents. Finally, the continuous-spin voltages are read out, buffered, and sent to the decoder to render the image.

Conditioning. A key advantage is that the coupling coefficients stay fixed across image-class requests. The coupler array is therefore programmed and calibrated only once. For each request, the corresponding bias currents are set in parallel. For a fully connected 4096-spin Ising substrate with symmetric 5-bit couplings, switching to a different image class requires changing only the conditioning currents. No reprogramming or recalibration follows.

Circuit design. The continuous-spin circuit in Figure 5(b) uses a current conveyor that holds the coupling node X near the reference voltage V_g while conveying the bidirectional coupling current

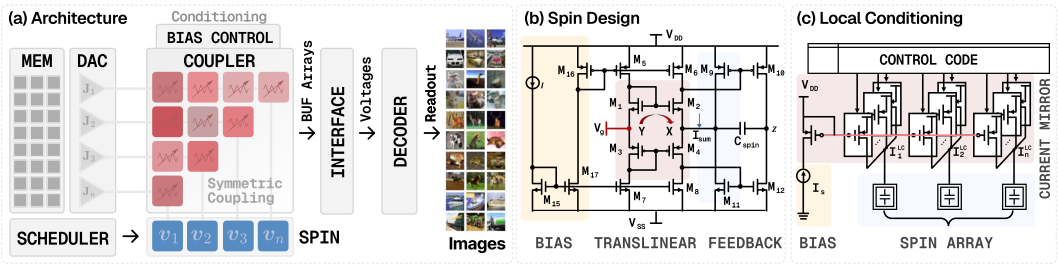


Figure 5: CMOS-compatible dynamical-system implementation. (a) Overall architecture. (b) Circuit design for spins. (c) Circuit design for local conditioning.

Table 3: Robustness of the proposed models to analog hardware non-idealities at $n = 4096$. Every stored weight of the dynamical system is quantized to k bits and read back with element-wise multiplicative Gaussian noise.

Weight precision	**Full precision**			**5-bit quantized**			**4-bit quantized**		
Noise level	None	2%	5%	None	2%	5%	None	2%	5%
Kuramoto-4096									
clean-FID ↓	5.37	5.39	5.27	5.24	5.34	5.13	5.37	5.30	5.34
Change vs. ideal	–	+0.02	−0.10	−0.13	−0.03	−0.24	0.00	−0.07	−0.03
Ising-4096									
clean-FID ↓	6.40	6.72	6.66	6.40	6.33	6.64	6.56	6.87	6.87
Change vs. ideal	–	+0.32	+0.26	0.00	−0.07	+0.24	+0.16	+0.47	+0.47

I_{sum} to the spin capacitor C_{spin}. The capacitor integrates the net coupling current, and its voltage represents the continuous-spin state. Figure 5(c) shows the local conditioning circuit, where a reference current I_S biases a current-mirror array and locally stored binary control codes independently set the bias current applied to each spin.

Decoder. The decoder is a seven-layer convolutional network that requires 26M, 34M, and 86M multiply-accumulates (MACs) per generated image for $n = 1024$, 2048, and 4096, respectively. Its convolution operations can be mapped to existing mixed-signal and CIM accelerators as matrix-vector multiplications with locally stored weights (Shafiee et al., 2016; Sebastian et al., 2020). For a hardware-based energy estimate, we use results from fabricated CIM chips: a 55-nm 3D-RRAM CIM reports 29.94 TOPS/W for 4-bit inputs and 5-bit weights (Huo et al., 2022), while a 55-nm Twin-8T SRAM-CIM reports 37.5 TOPS/W for 2-bit inputs and 5-bit weights (Si et al., 2020). Counting two operations per MAC, these measured efficiencies give estimated decoder energies of 1.39–1.74 µJ, 1.81–2.27 µJ, and 4.59–5.74 µJ per image for $n = 1024$, 2048, and 4096, respectively. For latency, a fabricated 16-nm multibit CIM accelerator reports a measured peak MAC throughput of 3.0 TOPS at 8-bit precision and 11.8 TOPS at 4-bit precision, and demonstrates an end-to-end throughput of 7,815 images/s for a fully mapped 11-layer 4-bit CNN (Jia et al., 2022). Using the measured 3.0-TOPS point as a conservative throughput reference gives compute-only latency estimates of approximately 17.3 µs, 22.7 µs, and 57.3 µs for the three decoder sizes.

Validation. To validate the proposed architecture at the circuit level, we perform post-layout transistor-level simulations of the dynamical-system core in a 65-nm CMOS technology using Cadence Virtuoso AMS with Spectre as the analog simulator. A 64-spin model for MNIST generation serves as the test case (Appendix D). The continuous-spin states are read out after 10.1 ns, a time set by the model dynamics and the circuit RC time constant. The simulated energy consumption of the dynamical-system core is 0.216 nJ per image.

4.2 Mitigating Hardware Non-Idealities

Emerging hardware substrates, such as analog computing platforms, often suffer from non-idealities (Xiao et al., 2022a), including limited programming precision, circuit variations, and noise. We therefore train and evaluate the dynamics under these effects. Every stored weight, namely the coupling A or K and the local conditioning vectors b_y, is quantized to k bits by a quantizer Q_k scaled to the tensor's largest magnitude, and read back as $W_{\mathrm{hw}} = Q_k(W) \odot (\mathbf{1} + \sigma\varepsilon)$ with $\varepsilon \sim \mathcal{N}(0, I)$ and noise level $\sigma \in \{0, 2\%, 5\%\}$. During training, gradients pass through the rounding via a straight-through estimator (Bengio et al., 2013). Table 3 shows that both models tolerate these non-idealities. In every tested setting, clean-FID increases by less than 0.5. With quantization- and noise-aware training, the models therefore operate under realistic hardware constraints at little cost in generation quality.

5 Related Work

Hopfield networks and Boltzmann machines. Hopfield networks connect recurrent dynamics to an energy landscape (Hopfield, 1982), and the graded-response extension carries energy descent

over to continuous states (Hopfield, 1984). Their purpose is associative memory: stored patterns are attractors, and a corrupted cue evolves to the matching fixed point. Modern Hopfield networks raise memory capacity and relate their update rule to transformer attention (Ramsauer et al., 2021; Krotov, 2023; Hu et al., 2023; Masumura & Taki, 2025), and the Energy Transformer minimizes a global energy with recurrent attention (Hoover et al., 2023). Boltzmann machines use a similar pairwise energy to define an equilibrium distribution, $p(x) \propto \exp[-E(x)]$, and learn by matching equilibrium statistics (Ackley et al., 1985), often approximated by contrastive divergence (Hinton, 2002). Our models share this pairwise energy but use it differently. Our energy landscape does not define a likelihood, and generation does not require reaching equilibrium. Instead, E_y^{I} and E_y^{K} specify a descent that turns a random state into a representation for the decoder, and training optimizes the resulting samples.

Generative models. Diffusion models generate samples by reversing a prescribed forward noising process using a learned time-dependent denoising or score function (Sohl-Dickstein et al., 2015; Ho et al., 2020; Yang et al., 2023). Flow matching (Liu et al., 2023; Lipman et al., 2023; Davis et al., 2024) instead learns time-conditioned velocities that transport a base distribution along a predefined probability path. Both explicitly supervise dynamics in data or latent space during training. Equilibrium Matching (Wang & Du, 2025) learns a time-invariant gradient over noise–data interpolations so that sampling descends a fixed energy landscape. Our energy has a different role: it is defined over the physical system's internal state space, and a decoder maps the final states to images. It neither models the image density nor supplies a supervised transport. Conditioning is a local term for each node that tilts a shared energy landscape. The generator has no forward noising process or learned time-dependent transport.

Ising machines and dynamical systems. Ising machines were first built for combinatorial optimization: they map a problem to an Ising energy and search for low-energy states. Implementations include photonic coherent Ising machines (Inagaki et al., 2016; Yamamoto et al., 2017; Hamerly et al., 2019), oscillator-based Ising machines (Chou et al., 2019; Wang & Roychowdhury, 2019; Vaidya et al., 2022; Moy et al., 2022; Albertsson & Rusu, 2023; Cılasun et al., 2025) and their theoretical models (Erementchouk et al., 2022; Ekanayake & Shukla, 2025), and CMOS (Afoakwa et al., 2021; Su et al., 2023; Wang & Fong, 2024; Liu et al., 2025e;d) and superconducting (Razmkhah et al., 2024; Ucpinar et al., 2024) circuits. They have been applied to Max-Cut (McMahon et al., 2016; Haribara et al., 2016; Wang & Roychowdhury, 2019; Bashar et al., 2020; Ucpinar et al., 2024; Shaglel et al., 2026), SAT (Bashar & Shukla, 2023; Bybee et al., 2023; Sharma et al., 2023; Su et al., 2023; Sikhakollu et al., 2024; De Prins et al., 2025; Sun et al., 2025), MIMO detection (Singh et al., 2022; Sreedhara et al., 2023), and other optimization problems (Tanahashi et al., 2019; Matsumoto et al., 2022; Parizy et al., 2022; Bao et al., 2024; Kirihara et al., 2023; Mao et al., 2023; Tsuyumine et al., 2024). Related Ising-derived dynamical systems have been trained for graph learning, function learning, and differential equation solving (Song et al., 2024; Wu et al., 2024; Liu et al., 2025a;b;c).

6 Conclusion

This work presents a design principle for efficient image generation on non-von Neumann hardware. We develop continuous Ising and Kuramoto models that turn physical energy minimization into generative computation. Both models descend explicit energy functions and map naturally to existing hardware. For conditional generation, we close the gap between models and hardware with local conditioning: a local term for each node selects the class while the programmed interactions stay fixed. This is attractive for physical systems such as memristive and photonic arrays, where writing and calibrating pairwise couplings costs more than updating biases. Our evaluation shows that high-quality image generation can emerge from a different and restricted computational primitive, with projected efficiency far beyond that of GPUs.

Several questions remain open. Richer energy functions may make the model more expressive but need more complex hardware. Our current implementation is fully connected: exploring sparse or structured connectivity could substantially reduce coupler, programming, and storage requirements. Finally, validating the models on fabricated hardware under measured noise, precision, programming, and calibration constraints is the next step toward the hardware-efficiency goal of this work.

AI USE STATEMENT

In this work, we used generative AI tools to polish the writing, debug and verify code, search related work and polish the code documentation. We have not used generative AI tools for generating synthetic data sets, helping develop theoretical models or conceptual frameworks, designing or providing feedback on research methodology or experiments, implementing methods, interpreting results, or plotting figures. And the rest of the required disclosure tasks are not applicable to this work. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Our code is available at `https://anonymous.4open.science/r/CF9E`.

REFERENCES

David H. Ackley, Geoffrey E. Hinton, and Terrence J. Sejnowski. A learning algorithm for Boltzmann machines. *Cognitive Science*, 9(1):147–169, 1985.

Richard Afoakwa, Yiqiao Zhang, Uday Kumar Reddy Vengalam, Zeljko Ignjatovic, and Michael Huang. BRIM: Bistable resistively-coupled Ising machine. In *IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, pp. 749–760, 2021.

Dagur I Albertsson and Ana Rusu. Highly reconfigurable oscillator-based ising machine through quasiperiodic modulation of coupling strength. *Scientific Reports*, 13(1):4005, 2023.

Stefano Ambrogio, Pritish Narayanan, Atsuya Okazaki, Andrea Fasoli, Charles Mackin, Kohji Hosokawa, Akiyo Nomura, Takeo Yasuda, An Chen, A Friz, et al. An analog-ai chip for energy-efficient speech recognition and transcription. *Nature*, 620(7975):768–775, 2023.

Michal Balcerak, Tamaz Amiranashvili, Antonio Terpin, Suprosanna Shit, Lea Bogensperger, Sebastian Kaltenbach, Petros Koumoutsakos, and Bjoern Menze. Energy matching: Unifying flow matching and energy-based models for generative modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 8583–8609, 2025. URL `https://proceedings.neurips.cc/paper_files/paper/2025/hash/0cbbdfb0a4098af8dc7a497a5e59aff7-Abstract-Conference.html`.

Siya Bao, Masashi Tawada, Shu Tanaka, and Nozomu Togawa. An ising-machine-based solver of vehicle routing problem with balanced pick-up. *IEEE Transactions on Consumer Electronics*, 70 (1):445–459, 2024. doi: 10.1109/TCE.2023.3335392.

Mohammad Khairul Bashar and Nikhil Shukla. Designing ising machines with higher order spin interactions and their application in solving combinatorial optimization. *Scientific Reports*, 13(1): 9558, 2023.

Mohammad Khairul Bashar, Antik Mallick, Daniel S Truesdell, Benton H Calhoun, Siddharth Joshi, and Nikhil Shukla. Experimental demonstration of a reconfigurable coupled oscillator platform to solve the max-cut problem. *IEEE Journal on Exploratory Solid-State Computational Devices and Circuits*, 6(2):116–121, 2020.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

William A. Borders, Ahmed Z. Pervaiz, Shunsuke Fukami, Kerem Y. Camsari, Hideo Ohno, and Supriyo Datta. Integer factorization using stochastic magnetic tunnel junctions. *Nature*, 573: 390–393, 2019.

Connor Bybee, Denis Kleyko, Dmitri E Nikonov, Amir Khosrowshahi, Bruno A Olshausen, and Friedrich T Sommer. Efficient optimization with higher-order ising machines. *Nature Communications*, 14(1):6033, 2023.

Cyrill Bösch, Geoffrey Roeder, Marc Serra-Garcia, and Ryan P. Adams. Local learning rules for out-of-equilibrium physical generative models, 2026. URL https://arxiv.org/abs/2506.19136.

Fuxi Cai, Suhas Kumar, Thomas Van Vaerenbergh, Xia Sheng, Rui Liu, Can Li, Zhan Liu, Martin Foltin, Shimeng Yu, Qiangfei Xia, J. Joshua Yang, Raymond Beausoleil, Wei D. Lu, and John Paul Strachan. Power-efficient combinatorial optimization using intrinsic noise in memristor Hopfield neural networks. *Nature Electronics*, 3(7):409–418, 2020. doi: 10.1038/s41928-020-0436-6.

Jeffrey Chou, Suraj Bramhavar, Siddhartha Ghosh, and William Herzog. Analog coupled oscillator based weighted ising machine. *Scientific reports*, 9(1):14786, 2019.

Hüsrev Cılasun, William Moy, Ziqing Zeng, Tahmida Islam, Hao Lo, Alex Vanasse, Megan Tan, Mohammad Anees, Ramprasath S, Abhimanyu Kumar, et al. A coupled-oscillator-based ising chip for combinatorial optimization. *Nature Electronics*, 8(6):537–546, 2025.

Oscar Davis, Samuel Kessler, Mircea Petrache, Ismail I Ceylan, Michael Bronstein, and Avishek J Bose. Fisher flow matching for generative modeling over discrete data. *Advances in Neural Information Processing Systems*, 37:139054–139084, 2024.

Robbe De Prins, Guy Van der Sande, Peter Bienstman, and Thomas Van Vaerenbergh. How to incorporate higher-order interactions in analog ising machines. *arXiv preprint arXiv:2507.23621*, 2025.

Mingyang Deng, He Li, Tianhong Li, Yilun Du, and Kaiming He. Generative modeling via drifting, 2026. URL https://arxiv.org/abs/2602.04770.

E.M.H.E.B. Ekanayake and Nikhil Shukla. Different paths, same destination: Designing physics-inspired dynamical systems with engineered stability to minimize the ising hamiltonian. *Phys. Rev. Appl.*, 24:024008, Aug 2025. doi: 10.1103/cdc9-y234. URL https://link.aps.org/doi/10.1103/cdc9-y234.

Mikhail Erementchouk, Aditya Shukla, and Pinaki Mazumder. On computational capabilities of ising machines based on nonlinear oscillators. *Physica D: Nonlinear Phenomena*, 437:133334, 2022.

Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouvé, and Gabriel Peyré. Interpolating between optimal transport and mmd using sinkhorn divergences. In *The 22nd international conference on artificial intelligence and statistics*, pp. 2681–2690. PMLR, 2019.

Aude Genevay, Gabriel Peyre, and Marco Cuturi. Learning generative models with sinkhorn divergences. In Amos Storkey and Fernando Perez-Cruz (eds.), *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 1608–1617. PMLR, 09–11 Apr 2018. URL https://proceedings.mlr.press/v84/genevay18a.html.

Zhengyang Geng, Ashwini Pokle, and J. Zico Kolter. One-step diffusion distillation via deep equilibrium models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 41914–41931, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/82f05a105c928c10706213952bf0c8b7-Abstract-Conference.html.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pp. 2672–2680, Cambridge, MA, USA, 2014. MIT Press.

Ryan Hamerly, Takahiro Inagaki, Peter L McMahon, Davide Venturelli, Alireza Marandi, Tatsuhiro Onodera, Edwin Ng, Carsten Langrock, Kensuke Inaba, Toshimori Honjo, et al. Experimental investigation of performance differences between coherent ising machines and a quantum annealer. *Science advances*, 5(5):eaau0823, 2019.

Yoshitaka Haribara, Shoko Utsunomiya, and Yoshihisa Yamamoto. *A Coherent Ising Machine for MAX-CUT Problems: Performance Evaluation against Semidefinite Programming and Simulated Annealing*, pp. 251–262. Springer Japan, Tokyo, 2016. ISBN 978-4-431-55756-2. doi: 10.1007/978-4-431-55756-2_12. URL https://doi.org/10.1007/978-4-431-55756-2_12.

Ping He, Om Khangaonkar, Hamed Pirsiavash, Yikun Bai, and Soheil Kolouri. Sinkhorn-drifting generative models. *arXiv preprint arXiv:2603.12366*, 2026.

Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002. doi: 10.1162/089976602760128018.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Benjamin Hoover, Yuchen Liang, Bao Pham, Rameswar Panda, Hendrik Strobelt, Duen Horng Chau, Mohammed Zaki, and Dmitry Krotov. Energy transformer. *Advances in neural information processing systems*, 36:27532–27559, 2023.

John J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982.

John J. Hopfield. Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences*, 81(10):3088–3092, 1984.

Jerry Yao-Chieh Hu, Donglin Yang, Dennis Wu, Chenwei Xu, Bo-Yu Chen, and Han Liu. On sparse modern hopfield model. *Advances in neural information processing systems*, 36:27594–27608, 2023.

Qiang Huo, Yiming Yang, Yiming Wang, Dengyun Lei, Xiangqu Fu, Qirui Ren, Xiaoxin Xu, Qing Luo, Guozhong Xing, Chengying Chen, Xin Si, Hao Wu, Yiyang Yuan, Qiang Li, Xiaoran Li, Xinghua Wang, Meng-Fan Chang, Feng Zhang, and Ming Liu. A computing-in-memory macro based on three-dimensional resistive random-access memory. *Nature Electronics*, 5:469–477, 2022. doi: 10.1038/s41928-022-00795-x.

Daniele Ielmini and H-S Philip Wong. In-memory computing with resistive switching devices. *Nature electronics*, 1(6):333–343, 2018.

Takahiro Inagaki, Yoshitaka Haribara, Koji Igarashi, Tomohiro Sonobe, Shuhei Tamate, Toshimori Honjo, Alireza Marandi, Peter L McMahon, Takeshi Umeki, Koji Enbutsu, et al. A coherent ising machine for 2000-node optimization problems. *Science*, 354(6312):603–606, 2016.

Ernst Ising. Beitrag zur theorie des ferromagnetismus. *Zeitschrift für Physik*, 31(1):253–258, 1925.

Andraž Jelinčič, Owen Lockwood, Akhil Garlapati, Peter Schillinger, Isaac L. Chuang, Guillaume Verdon, and Trevor McCourt. An efficient probabilistic hardware architecture for diffusion-like models. *npj Unconventional Computing*, 3:30, 2026. doi: 10.1038/s44335-026-00075-3.

Hongyang Jia, Murat Ozatay, Yinqi Tang, Hossein Valavi, Rakshit Pathak, Jinseok Lee, and Naveen Verma. Scalable and programmable neural network inference accelerator based on in-memory computing. *IEEE Journal of Solid-State Circuits*, 57(1):198–211, 2022. doi: 10.1109/JSSC.2021.3119018.

Mingrui Jiang, Keyi Shan, Chengping He, and Can Li. Efficient combinatorial optimization by quantum-inspired parallel annealing in analogue memristor crossbar. *Nature Communications*, 14(1):5927, 2023. doi: 10.1038/s41467-023-41647-2.

Mark W Johnson, Mohammad HS Amin, Suzanne Gildert, Trevor Lanting, Firas Hamze, Neil Dickson, Richard Harris, Andrew J Berkley, Jan Johansson, Paul Bunyk, et al. Quantum annealing with manufactured spins. *Nature*, 473(7346):194–198, 2011. doi: 10.1038/nature10012.

Kirill P. Kalinin, Jannes Gladrow, Jiaqi Chu, James H. Clegg, Daniel Cletheroe, Douglas J. Kelly, Babak Rahmani, Grace Brennan, Burcu Canakci, Fabian Falck, Michael Hansen, Jim Kleewein, Heiner Kremer, Greg O'Shea, Lucinda Pickup, Saravan Rajmohan, Ant Rowstron, Victor Rühle, Lee Braine, Shrirang Khedekar, Natalia G. Berloff, Christos Gkantsidis, Francesca Parmigiani, and Hitesh Ballani. Analog optical computer for AI inference and combinatorial optimization. *Nature*, 645:354–361, 2025.

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Kenta Kirihara, Hidetaka Imai, Eisuke Kuroda, Jun Yamazaki, and Akiko Masaki-Kato. Exploring potential applications of ising machines for power system operations. *IEEE Access*, 11:68004–68017, 2023.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Dmitry Krotov. A new frontier for hopfield networks. *Nature Reviews Physics*, 5(7):366–367, 2023.

Yoshiki Kuramoto. Self-entrainment of a population of coupled non-linear oscillators. In *International Symposium on Mathematical Problems in Theoretical Physics*, volume 39 of *Lecture Notes in Physics*, pp. 420–422. Springer, 1975.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Chuan Liu, Ruibing Song, Chunshu Wu, Pouya Haghi, and Tong Geng. Instatrain: Adaptive training via ultra-fast natural annealing within dynamical systems. In *International Conference on Learning Representations*, volume 2025, pp. 42729–42741, 2025a.

Chuan Liu, Chunshu Wu, Ruibing Song, Ang Li, Ying Nian Wu, and Tong Geng. An expressive and self-adaptive dynamical system for efficient function learning. In *Forty-second International Conference on Machine Learning*, 2025b. URL https://openreview.net/forum?id=KU7jlB2MPN.

Chuan Liu, Chunshu Wu, Ruibing Song, Guangyan Sun, Ying Nian Wu, Yousu Chen, Ang Li, and Tong Geng. Ds-tide: Harnessing dynamical systems for efficient time-independent differential equation solving. In *Proceedings of the 58th IEEE/ACM International Symposium on Microarchitecture*, pp. 1690–1703, 2025c.

Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

Yongchao Liu, Lianlong Sun, Matthew Burns, Michael Huang, and Hui Wu. Ising machine based on charge re-distribution. In *2025 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, 2025d. doi: 10.1109/ISCAS56072.2025.11044217.

Yongchao Liu, Lianlong Sun, Michael Huang, and Hui Wu. Integrated hardware annealing based on langevin dynamics for ising machines. In *2025 Design, Automation & Test in Europe Conference (DATE)*, pp. 1–6, 2025e. doi: 10.23919/DATE64628.2025.10993070.

Hao Lo, William Moy, Hanzhao Yu, Sachin Sapatnekar, and Chris H Kim. An ising solver chip based on coupled ring oscillators with a 48-node all-to-all connected array architecture. *Nature Electronics*, 6(10):771–778, 2023.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 5775–5787, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/260a14acce2a89dad36adc8eefe7c59e-Abstract-Conference.html.

Andrew Lucas. Ising formulations of many np problems. *Frontiers in Physics*, Volume 2 - 2014, 2014. ISSN 2296-424X. doi: 10.3389/fphy.2014.00005. URL https://www.frontiersin.org/journals/physics/articles/10.3389/fphy.2014.00005.

Tianze Luo, Haotian Yuan, and Zhuang Liu. SoFlow: Solution flow models for one-step generative modeling. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/a654298de44ec95dec28aa37e8d0af51-Abstract-Conference.html.

Piergiulio Mannocci, Giacomo Larelli, Marco Bonomi, and Daniele Ielmini. Achieving high precision in analog in-memory computing systems. *npj Unconventional Computing*, 3(1):1, 2026.

Zetian Mao, Yoshiki Matsuda, Ryo Tamura, and Koji Tsuda. Chemical design with gpu-based ising machines. *Digital Discovery*, 2(4):1098–1103, 2023.

Tsubasa Masumura and Masato Taki. On the role of hidden states of modern hopfield network in transformer. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 68607–68626. Curran Associates, Inc., 2025. doi: 10.52202/085713-2308. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/634ddc254c9e739eea6c5393e3abe5c9-Paper-Conference.pdf.

Nasa Matsumoto, Yohei Hamakawa, Kosuke Tatsumura, and Kazue Kudo. Distance-based clustering using qubo formulations. *Scientific reports*, 12(1):2669, 2022.

Peter L McMahon, Alireza Marandi, Yoshitaka Haribara, Ryan Hamerly, Carsten Langrock, Shuhei Tamate, Takahiro Inagaki, Hiroki Takesue, Shoko Utsunomiya, Kazuyuki Aihara, et al. A fully programmable 100-spin coherent ising machine with all-to-all connections. *Science*, 354(6312): 614–617, 2016.

Denis Melanson, Mohammad Abu Khater, Maxwell Aifer, Kaelan Donatella, Max Hunter Gordon, Thomas Ahle, Gavin Crooks, Antonio J Martinez, Faris Sbahi, and Patrick J Coles. Thermodynamic computing system for ai applications. *Nature Communications*, 16(1):3757, 2025. doi: 10.1038/s41467-025-59011-x.

Ali Momeni, Babak Rahmani, Benjamin Scellier, Logan G Wright, Peter L McMahon, Clara C Wanjura, Yuhang Li, Anas Skalli, Natalia G Berloff, Tatsuhiro Onodera, et al. Training of physical neural networks. *Nature*, 645(8079):53–61, 2025.

William Moy, Ibrahim Ahmed, Po-wei Chiu, John Moy, Sachin S Sapatnekar, and Chris H Kim. A 1,968-node coupled ring oscillator circuit for combinatorial optimization problem solving. *Nature Electronics*, 5(5):310–317, 2022. doi: 10.1038/s41928-022-00749-3.

Ilker Oguz, Niyazi Ulas Dinc, Mustafa Yildirim, Junjie Ke, Innfarn Yoo, QIFEI WANG, Feng Yang, Christophe Moser, and Demetri Psaltis. Optical diffusion models for image generation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=RY3rDQV0tQ.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt. Featured Certification.

Matthieu Parizy, Przemyslaw Sadowski, and Nozomu Togawa. Cardinality constrained portfolio optimization on an ising machine. In *2022 IEEE 35th International System-on-Chip Conference (SOCC)*, pp. 1–6, 2022. doi: 10.1109/SOCC56010.2022.9908082.

Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On Aliased Resizing and Surprising Subtleties in GAN Evaluation . In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11400–11410, Los Alamitos, CA, USA, June 2022. IEEE Computer Society. doi: 10.1109/CVPR52688.2022.01112. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52688.2022.01112.

David A. Patterson, Joseph Gonzalez, Quoc V. Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David R. So, Maud Texier, and Jeff Dean. Carbon emissions and large neural network training. *CoRR*, abs/2104.10350, 2021. URL https://arxiv.org/abs/2104.10350.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=tL89RnzIiCd.

Sasan Razmkhah, Mehdi Kamal, Nobuyuki Yoshikawa, and Massoud Pedram. Josephson parametric oscillator based ising machine. *Physical Review B*, 109(1):014511, 2024.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pp. 2234–2242, Red Hook, NY, USA, 2016. Curran Associates Inc. ISBN 9781510838819.

Abu Sebastian, Manuel Le Gallo, Riduan Khaddam-Aljameh, and Evangelos Eleftheriou. Memory devices and applications for in-memory computing. *Nature nanotechnology*, 15(7):529–544, 2020.

Ali Shafiee, Anirban Nag, Naveen Muralimanohar, Rajeev Balasubramonian, John Paul Strachan, Miao Hu, R. Stanley Williams, and Vivek Srikumar. ISAAC: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars. In *ACM/IEEE International Symposium on Computer Architecture (ISCA)*, pp. 14–26, 2016.

Salwa Shaglel, Markus Kirsch, Marten Winkler, Christian Münch, Stefan Walter, Fritz Schinkel, and Martin Kliesch. A comprehensive benchmark of an ising machine on the max-cut problem. *New Journal of Physics*, 2026.

Anshujit Sharma, Matthew Burns, Andrew Hahn, and Michael Huang. Augmenting an electronic ising machine to effectively solve boolean satisfiability. *Scientific Reports*, 13(1):22858, 2023.

Yichen Shen, Nicholas C Harris, Scott Skirlo, Mihika Prabhu, Tom Baehr-Jones, Michael Hochberg, Xin Sun, Shijie Zhao, Hugo Larochelle, Dirk Englund, et al. Deep learning with coherent nanophotonic circuits. *Nature photonics*, 11(7):441–446, 2017.

Xin Si, Jia-Jing Chen, Yung-Ning Tu, Wei-Hsing Huang, Jing-Hong Wang, Yen-Cheng Chiu, Wei-Chen Wei, Ssu-Yen Wu, Xiaoyu Sun, Rui Liu, Shimeng Yu, Ren-Shuo Liu, Chih-Cheng Hsieh, Kea-Tiong Tang, Qiang Li, and Meng-Fan Chang. A twin-8t sram computation-in-memory unit-macro for multibit cnn-based ai edge processors. *IEEE Journal of Solid-State Circuits*, 55(1): 189–202, 2020. doi: 10.1109/JSSC.2019.2952773.

Venkata Pavan Sumanth Sikhakollu, Shreesha Sreedhara, Rajit Manohar, Alan Mishchenko, and Jaijeet Roychowdhury. High quality circuit-based 3-sat mappings for oscillator ising machines. In Da-Jung Cho and Jongmin Kim (eds.), *Unconventional Computation and Natural Computation*, pp. 269–285, Cham, 2024. Springer Nature Switzerland. ISBN 978-3-031-63742-1.

Abhishek Kumar Singh, Kyle Jamieson, Peter L McMahon, and Davide Venturelli. Ising machines' dynamics and regularization for near-optimal mimo detection. *IEEE Transactions on Wireless Communications*, 21(12):11080–11094, 2022.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 2256–2265, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/sohl-dickstein15.html.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://openreview.net/forum?id=St1giarCHLP.

Ruibing Song, Chunshu Wu, Chuan Liu, Ang Li, Michael Huang, and Tony Tong Geng. Ds-gl: Advancing graph learning via harnessing nature's power within scalable dynamical systems. In *2024 ACM/IEEE 51st Annual International Symposium on Computer Architecture (ISCA)*, pp. 45–57, 2024. doi: 10.1109/ISCA59077.2024.00014.

Shreesha Sreedhara, Jaijeet Roychowdhury, Joachim Wabnig, and Pavan Koteshwar Srinath. Mu-mimo detection using oscillator ising machines. In *2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD)*, pp. 1–9, 2023. doi: 10.1109/ICCAD57390.2023.10323680.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in NLP. In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3645–3650, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1355. URL https://aclanthology.org/P19-1355/.

Yuqi Su, Tony Tae-Hyoung Kim, and Bongjin Kim. A reconfigurable cmos ising machine with three-body spin interactions for solving boolean satisfiability with direct mapping. *IEEE Solid-State Circuits Letters*, 6:221–224, 2023.

Lianlong Sun, Matthew X. Burns, and Michael C. Huang. General oscillator-based ising-machine models with phase-amplitude dynamics and polynomial interactions. *Phys. Rev. Appl.*, 24: 044035, Oct 2025. doi: 10.1103/7s4v-4hs4. URL https://link.aps.org/doi/10.1103/7s4v-4hs4.

Kotaro Tanahashi, Shinichi Takayanagi, Tomomitsu Motohashi, and Shu Tanaka. Application of ising machines and a software development for ising machines. *Journal of the Physical Society of Japan*, 88(6):061010, 2019.

Alexander Tong, Kilian FATRAS, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=CD9Snc73AW. Expert Certification.

Yui Tsuyumine, Kenichi Masuda, Takeshi Hachikawa, Tsuyoshi Haga, Yuta Yachi, Tatsuhiko Shirai, Masashi Tawada, and Nozomu Togawa. Optimization of practical time-dependent vehicle routing problem by ising machines. In *2024 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–5, 2024. doi: 10.1109/ICCE59016.2024.10444436.

Beyza Zeynep Ucpinar, Sasan Razmkhah, Mehdi Kamal, and Massoud Pedram. Scalable superconductor ising machine for combinatorial optimization problems. In *2024 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pp. 565–570, 2024. doi: 10.1109/ISVLSI61997.2024.00107.

Unconventional AI. Introducing un-0: Generating images with coupled oscillators, June 2026. URL https://unconv.ai/blog/introducing-un-0-generating-images-with-coupled-oscillators/.

Jaykumar Vaidya, RS Surya Kanthi, and Nikhil Shukla. Creating electronic oscillator-based ising machines without external injection locking. *Scientific Reports*, 12(1):981, 2022.

Runqian Wang and Yilun Du. Equilibrium matching: Generative modeling with implicit energy-based models, 2025. URL https://arxiv.org/abs/2510.02300.

Tianshi Wang and Jaijeet Roychowdhury. Oim: Oscillator-based ising machines for solving combinatorial optimisation problems. In *Unconventional Computation and Natural Computation: 18th International Conference, UCNC 2019, Tokyo, Japan, June 3–7, 2019, Proceedings*, pp. 232–256, Berlin, Heidelberg, 2019. Springer-Verlag. ISBN 978-3-030-19310-2. doi: 10.1007/978-3-030-19311-9_19. URL https://doi.org/10.1007/978-3-030-19311-9_19.

Yimin Wang and Xuanyao Fong. Energy-efficient ising machines using capacitance-coupled latches for maxcut solving. In *2024 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, 2024. doi: 10.1109/ISCAS58744.2024.10558647.

Zhendong Wang, Huangjie Zheng, Pengcheng He, Weizhu Chen, and Mingyuan Zhou. Diffusion-GAN: Training GANs with diffusion. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=HZf7UbpWHuA.

Stephen Whitelam. Generative thermodynamic computing. *Physical Review Letters*, 136:037101, 2026. doi: 10.1103/kwyy-1xln.

Chunshu Wu, Ruibing Song, Chuan Liu, Yunan Yang, Ang Li, Michael Huang, and Tong Geng. Extending power of nature from binary to real-valued graph learning in real world. In *International Conference on Learning Representations*, volume 2024, pp. 57041–57053, 2024.

T. Patrick Xiao, Ben Feinberg, Christopher H. Bennett, Venkatraman Prabhakar, Prashant Saxena, Vineet Agrawal, Sapan Agarwal, and Matthew J. Marinella. On the accuracy of analog neural network inference accelerators. *IEEE Circuits and Systems Magazine*, 22(4):26–48, 2022a. doi: 10.1109/MCAS.2022.3214409.

Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion GANs. In *International Conference on Learning Representations (ICLR)*, 2022b. URL https://openreview.net/forum?id=JprM0p-q0Co.

Yoshihisa Yamamoto, Kazuyuki Aihara, Timothee Leleu, Ken-ichi Kawarabayashi, Satoshi Kako, Martin Fejer, Kyo Inoue, and Hiroki Takesue. Coherent Ising machines—optical neural networks operating at the quantum limit. *npj Quantum Information*, 3(1):49, 2017.

Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM computing surveys*, 56(4):1–39, 2023.

Tingting Zhang, Qichao Tao, Bailiang Liu, Andrea Grimaldi, Eleonora Raimondo, Manuel Jiménez, María José Avedillo, Juan Nuñez, Bernabé Linares-Barranco, Teresa Serrano-Gotarredona, Giovanni Finocchio, and Jie Han. A review of ising machines implemented in conventional and emerging technologies. *IEEE Transactions on Nanotechnology*, 23:704–717, 2024. doi: 10.1109/TNANO.2024.3457533.

Yaxuan Zhu, Jianwen Xie, Ying Nian Wu, and Ruiqi Gao. Learning energy-based models by cooperative diffusion recovery likelihood. In *International Conference on Learning Representations (ICLR)*, 2024a. URL https://openreview.net/forum?id=AyzkDpuqcl.

Yuanzhi Zhu, Xingchao Liu, and Qiang Liu. Slimflow: Training smaller one-step diffusion models with rectified flow. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part LXXXII*, pp. 342–359, Berlin, Heidelberg, 2024b. Springer-Verlag. ISBN 978-3-031-73006-1. doi: 10.1007/978-3-031-73007-8_20. URL https://doi.org/10.1007/978-3-031-73007-8_20.

A Training Objective

We jointly train the dynamics and the decoder by matching generated and real image distributions within each class. This distribution-level supervision encourages both image fidelity and within-class diversity. We perform the matching in frozen DINOv2 feature spaces (Oquab et al., 2024), so that the loss reflects semantic content and local structure rather than pixelwise similarity alone.

Feature regression. Following the drifting objective (Deng et al., 2026), for each class y and feature view F_v, let $g_i = F_v(G(y, \xi_i))$ denote the feature of generated sample i. We compute a displacement V_i that specifies how g_i should change to better match the real feature distribution. Its normalized form $\widehat{V}_i$ defines a desired step in feature space. We use the shifted feature as a temporary regression target:

$$\tau_i = \mathrm{sg}\Big(g_i + \widehat{V}_i\Big), \qquad \ell_i = \|g_i - \tau_i\|_2^2. \tag{10}$$

The target represents a desired new feature position, not a paired real image. The stop-gradient operator sg holds τ_i fixed during backpropagation, yielding $\nabla_{g_i}\ell_i = -2\widehat{V}_i$. We average the losses over samples, classes, and feature views.

Sinkhorn displacement. To construct V_i, following He et al. (2026), we use Sinkhorn-based distribution matching to align generated features $\{g_i\}$ with same-class real references $\{r_j\}$, favoring nearby features while balancing assignments across all real references. Each generated feature is guided toward a weighted average of its matched references. This averaging, however, can pull distinct features together even when the generated and real distributions already agree. To correct this effect, we also match the current generated batch to an independent batch from the same generator and class, which estimates the attraction that soft matching introduces within the generated distribution itself. Let $M_{\mathrm{real}}(g_i)$ and $M_{\mathrm{self}}(g_i)$ denote the weighted reference means obtained from Sinkhorn matching to real and generated references, respectively. We construct

$$V_i = \big(M_{\mathrm{real}}(g_i) - g_i\big) - \big(M_{\mathrm{self}}(g_i) - g_i\big) = M_{\mathrm{real}}(g_i) - M_{\mathrm{self}}(g_i). \tag{11}$$

The first term points toward real data, and the second subtracts the self-attraction estimated from generated data. This correction is motivated by the debiased Sinkhorn divergence (Genevay et al., 2018; Feydy et al., 2019).

B Conditioning for Asymmetric Coupling

Local conditioning also applies to asymmetric coupling: one bias per node still selects the class, although the dynamics are no longer guaranteed to descend the energies. Asymmetric coupling may not be available on some hardware substrates. We evaluate this setting in Table 4. The dynamical system then requires roughly twice as many parameters as in the symmetric case.

Table 4: Asymmetric Ising and Kuramoto models on CIFAR-10, grouped by system size n.

Model	Reprogramming Needed	Total Params	Decoder Params	clean-FID ↓
Ising-1024-asym	No	1.13M	0.07M	9.71
Kuramoto-1024-asym	No	1.22M	0.16M	14.15
Ising-2048-asym	No	4.37M	0.16M	7.15
Kuramoto-2048-asym	No	4.80M	0.58M	8.23
Ising-4096-asym	No	17.40M	0.58M	5.96
Kuramoto-4096-asym	No	19.14M	2.33M	5.39

C Dynamical-System Core Cost

Table 5 summarizes the power and energy scaling of the continuous-spin dynamical-system core. The 64-spin post-layout simulation of Section 4 serves as the reference. For larger systems, the

Table 5: Power and energy per image of the dynamical-system core with $R_0 = 40\ \mathrm{k}\Omega$. The Reference row is obtained from post-layout transistor-level simulation, while Cases A and B are projections. P_{spin}, P_{coup}, and P_{bias} denote the power of the continuous-spin, coupler, and bias circuits, respectively. P_{tot} is their sum, and t_{read} is the evolution time.

n	C_{spin}	Case	P_{spin}	P_{coup}	P_{bias}	P_{tot}	t_{read}	E/image
64	250 fF	Reference	19.20 mW	1.657 mW	0.538 mW	21.40 mW	10.1 ns	0.216 nJ
128	353.6 fF	A	38.40 mW	6.628 mW	1.076 mW	46.10 mW	10.1 ns	0.466 nJ
128	353.6 fF	B	38.40 mW	3.314 mW	1.076 mW	42.79 mW	14.28 ns	0.611 nJ
1024	1.000 pF	A	307.2 mW	424.2 mW	8.608 mW	740.0 mW	10.1 ns	7.47 nJ
1024	1.000 pF	B	307.2 mW	26.51 mW	8.608 mW	342.3 mW	40.4 ns	13.83 nJ
2048	1.414 pF	A	614.4 mW	1.697 W	17.22 mW	2.329 W	10.1 ns	23.52 nJ
2048	1.414 pF	B	614.4 mW	53.02 mW	17.22 mW	684.6 mW	57.13 ns	39.11 nJ
4096	2.000 pF	A	1.229 W	6.787 W	34.43 mW	8.050 W	10.1 ns	81.31 nJ
4096	2.000 pF	B	1.229 W	106.0 mW	34.43 mW	1.369 W	80.8 ns	110.6 nJ

spin capacitance is conservatively scaled as $250\ \mathrm{fF}\sqrt{n/64}$ to account for the increased aggregate coupling current. Two representative coupling scenarios are considered. Case A approximately keeps the coupling density and resistance distribution of the reference network, yielding stronger aggregate coupling and a similar evolution time as the system scales. Case B instead holds a roughly constant number of nonzero couplings per spin, reducing coupling power at the cost of a longer evolution time. These cases represent two practical scaling scenarios rather than strict bounds. The same projection method is applied to the Kuramoto oscillator systems. From 64 to 128 oscillators, the simulated energy per image increases from 0.396 to $0.832\ \mathrm{nJ}$, with the evolution time increasing from 170 to $180\ \mathrm{ns}$. For larger systems, we conservatively assume that the coupling power scales with n^2 and the bias power scales with n. The projected energy per image is 10.32, 27.71, and $84.22\ \mathrm{nJ}$ for 1024, 2048, and 4096 oscillators, respectively.

D Circuit Simulation

D.1 Circuit-to-Dynamics Correspondence

In our circuit implementation, each continuous spin is represented by the voltage v_j stored on C_{spin} relative to the fixed reference voltage V_{g}. A differential amplifier generates the two polarities $V_{\mathrm{g}} + v_j$ and $V_{\mathrm{g}} - v_j$, such that the sign of each coupling coefficient is implemented by selecting the corresponding polarity at the input of the current-conveyor-based coupling path. The magnitude of the dimensionless coefficient A_{ij} is mapped to the physical conductance through a normalization resistance R_0. With the local bias current $I_{b,j}$ represented equivalently by $b_j = R_0 I_{b,j}$, the relevant circuit quantities and the resulting spin dynamics are

$$\begin{aligned} v_j &= V_j - V_{\mathrm{g}}, \qquad G_{ij} = \frac{|A_{ij}|}{R_0}, \qquad b_j = R_0 I_{b,j}, \\ \frac{dv_j}{dt} &= \frac{dV_j}{dt} = \frac{1}{R_0 C_{\mathrm{spin}}} \left(\sum_i A_{ij} v_i + b_j \right). \end{aligned} \tag{12}$$

Here, the selection of the positive or negative spin polarity gives the signed coupling contribution $A_{ij} v_i / R_0$, while the programmed conductance determines its magnitude.

For the symmetric interaction matrix $A_{ij} = A_{ji}$, the circuit dynamics descend the energy

$$E_{\mathrm{c}}(v) = -\frac{1}{2} \sum_j \sum_i A_{ij} v_i v_j - \sum_j b_j v_j = -\frac{1}{2} v^\top A v - b^\top v, \tag{13}$$

for which

$$-\frac{\partial E_{\mathrm{c}}}{\partial v_j} = \sum_i A_{ij} v_i + b_j, \qquad \frac{dv_j}{dt} = -\frac{1}{R_0 C_{\mathrm{spin}}} \frac{\partial E_{\mathrm{c}}}{\partial v_j}. \tag{14}$$

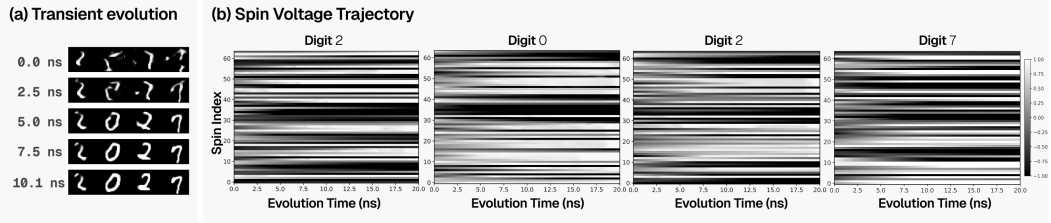


Figure 6: (a) Transient evolution of the decoded "2027" image. (b) Circuit-level spin-voltage trajectories of the four samples over the first 20 ns. The final spin states are sampled at 10.1 ns.

The continuous-spin network therefore follows the dynamics in the unsaturated operating region. Along its state trajectory, the energy satisfies

$$\frac{dE_{\mathrm{c}}}{dt} = \sum_j \frac{\partial E_{\mathrm{c}}}{\partial v_j}\frac{dv_j}{dt} = -R_0 C_{\mathrm{spin}} \sum_j \left(\frac{dv_j}{dt}\right)^2 \leq 0, \tag{15}$$

showing descent of $E_{\mathrm{c}}(v)$ under the circuit dynamics. The physical voltage range of the spin nodes bounds the feasible state space, while the programmed resistive coupling network determines the energy landscape through A. With $v = V_s x$ for a voltage scale V_s and $b_j = \gamma V_s b_{y,j}$, we have $E_{\mathrm{c}}(v) = V_s^2 E_y^{\mathrm{I}}(x)$, and Eq. (12) becomes Eq. (4) with time unit $R_0 C_{\mathrm{spin}}$.

D.2 Transient Mixed-Signal Operation

We perform a post-layout transient simulation of the hardware implementation in Section 4. The simulation consists of initialization, autonomous evolution, and readout. During initialization, the prescribed continuous-spin states are loaded onto the spin capacitors while the coupling coefficients and local conditioning currents are configured. The initialization paths are then disconnected simultaneously, after which the network evolves continuously through the interaction of the couplers, current-conveyor interfaces, conditioning currents, and spin capacitors. The analog solver integrates these circuit dynamics, whereas the event-driven digital domain is used only to sequence initialization, configuration, and readout. At readout, all continuous-spin voltages are simultaneously transferred to the hold capacitors through unity-gain input buffers. After the sampling switches are opened, the held voltages are isolated from the continuously evolving state nodes and buffered for subsequent processing by the image decoder.

All 64 continuous-spin trajectories are recorded throughout the circuit evolution. Figure 6(b) shows the circuit-level spin trajectories of the four samples that generate the digits "2027" over the first 20 ns. To visualize the corresponding evolution in the image domain, the circuit-extracted latent states are passed through the trained decoder at five representative time points, as shown in Figure 6(a). The decoded images progressively approach the target digit patterns, with the continuous-spin states sampled for final readout at 10.1 ns.

D.3 Large-Scale Circuit Validation

Although the preceding results use post-layout transistor-level simulation, extending the same Cadence setup to larger systems becomes computationally demanding. For the large-scale simulation, selected circuit blocks are therefore replaced by ideal elements parameterized from their corresponding circuit implementations, while the network dynamics and coupling topology are unchanged. This model enables simulation of a 1024-spin system in Cadence for CIFAR-10 image generation. Figure 7 shows the evolution of one generated image. The observed evolution times fall within the range estimated in Table 5, providing a consistency check between the large-scale simulation and the circuit-based performance projection.

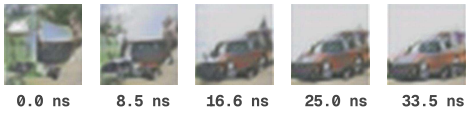


Figure 7: Evolution of one generated *automobile* image from a 1024-spin circuit simulation.