Catalyst N4: A 512-Core Dual-Chiplet Neuromorphic Processor with 134M Virtual Neurons, Spike Tensor Core, and Hardware Neuroscience Primitives

Henry Arthur Shulayev Barnes

University of Aberdeen
Aberdeen, AB24 3UE, United Kingdom
u13hs24@abdn.ac.uk

Catalyst Neuromorphic Ltd
London, United Kingdom
henry@catalyst-neuromorphic.com

Abstract

We present **Catalyst N4**, the fourth generation of the Catalyst neuromorphic processor family. N4 scales to a dual-chiplet architecture with 512 cores, 4,194,304 physical neurons expandable to **134,217,728 virtual neurons** via 32-context time-division multiplexing, and introduces dedicated spike-domain tensor acceleration, multi-head spiking attention, hardware backpropagation, hyperdimensional computing, Hopfield associative memory, and a hardware operating system managing 4,096 virtual networks.

The architecture organises 512 neuromorphic cores into 2 Neural Compute Clusters (NCCs) of 32 tiles of 8 cores each. Each core supports 8,192 neurons at 24-bit precision, 12,288 at 16-bit, or 16,384 at 8-bit, with a 32-bit full-precision mode. A 6-stage pipeline integrates 4-way simultaneous multithreading (SMT) via barrel scheduling, 8-wide cohort SIMD for population-level operations, a dendritic traversal unit for 16-compartment multi-compartment models with 8 dendritic join operations, and a ternary bypass path that reduces single-spike latency from 6 cycles to 1. Each core contains a 16×16 Spike Tensor Core (STC) performing conditional-add matrix operations at 256 operations per cycle with hardware sparsity intersection, and an 8-head spiking attention mechanism with a 4-head, 64-depth KV cache for transformer-style inference.

Variable-precision neurons operate at 24/16/8/32-bit widths. Five neuron models—LIF, CUBA, ALIF, ANN INT8 (4-wide MAC), and a programmable model with 32-opcode ISA and probabilistic bit (P-bit)—are joined by LTC (Liquid Time-Constant) neurons, rebound firing, burst firing, and divisive normalisation. Sixty-four configurable parameter groups of 39 parameters each control neuron behaviour per group. Eight synapse formats including KAN B-spline and block-sparse formats support variable-precision weights from 1 to 16 bits. The learning engine provides an 8-rule microcode system with 8 threads, 32 opcodes, 512-deep microcode memory, and 256 registers. Hardware backpropagation supports 8-layer networks with momentum and configurable surrogate gradient modes. Federated learning aggregates updates from 4 clients with differential privacy. Active forgetting provides tag-based selective weight decay.

A 32-channel neuromodulation bus, hardware metaplasticity, synaptic fatigue, short-term plasticity (facilitation and depression), factor decomposition traces, and convergence detection with IRQ complete the learning subsystem. Metacognition hardware monitors weight rate, spike entropy, prediction accuracy, and structural balance.

Neuroscience primitives include a 1,024-bit hyperdimensional computing engine with 64 registers and 8 operations, a 256-neuron Hopfield associative memory storing 256 patterns with energy-based recall, a working memory buffer (32 slots $\times$ 256-bit patterns with decay), 256-entry gap junctions with bidirectional coupling, an 8-zone calcium-gated glial cell model, 4 configurable oscillators per tile with phase locking, 4 interneuron templates with 8 targets each, and an event camera interface with 10-bit X/Y resolution, ROI filtering, and pixel-to-neuron mapping.

The memory hierarchy comprises 256 KB L1 per core with 64 KB shadow bank for TDM context switching, a 2 MB 16-way L2 cache per tile, 640 MB S3RAM at 5-cycle latency, 48 GB HBM3E, a 32 KB synapse cache (64-entry, 512-bit lines), a 512-entry priority-based spike sorter, a TAGE prefetcher (4 history tables, 512-entry prediction), a processing-in-memory controller for vector MAC operations, and 10 memory interface types. A rich-club network-on-chip topology with content-addressable routing (1,024-entry wildcard), 4 virtual channels with anti-starvation, up to 16 rich-club hubs, spike coalescing (4-cycle window, 32-entry table), and 16 express links per NCC connects tiles within each chiplet.

An RV64GC RISC-V subsystem with 14 custom neuromorphic opcodes, 8 cores, 512 KB instruction and data memory each, and a 1 MB L2 cache provides management and configuration. System features include 256 MSI-X interrupt vectors, 10 PLLs with 8 DVFS profiles, 4 thermal zones with 8

thermal diodes, 48 performance counters per tile (3,072 total), a 16K-entry trace buffer, an 8-channel anomaly detector, quadrant-based scheduling, CXL memory pooling (8 regions), UCIe chiplet interconnect, and multi-chip scaling to 64 chips via 12 AER links with cross-chip synchronisation, skew compensation, distributed error reporting, and distributed gradient aggregation.

Security features include lockstep execution, SRAM repair (264 physical cores, 256 active, 8 spare, e-fuse mapping), 8 PCR registers with SHA-256 measurement, 4-stage boot verification, AES-256-GCM spike encryption, CRYSTALS-Kyber ML-KEM-768 post-quantum key exchange, SRAM PUF identity, and a power meter with 8 thermal diodes.

We validate a 4-core configuration on an AWS F2 Xilinx VU47P FPGA at 62.5 MHz, achieving 126/126 hardware test pass rate across 11 test categories and 14,983 timesteps per second. An N4-Edge variant validated on AMD Kria K26 achieves 100 MHz timing closure with 3.3 ns positive slack at 0.378 W total power, using 2.59% of available LUTs. Benchmark evaluation yields 91.0% on Spiking Heidelberg Digits, 76.4% on Spiking Speech Commands, 99.2% on N-MNIST, and 89.4% on DVS Gesture.

1 Introduction

Neuromorphic processors address the energy wall facing conventional computing by co-locating computation with memory and communicating via sparse binary spikes [11]. The past decade produced several landmark chips: Intel Loihi 1 and Loihi 2 [4, 5], IBM TrueNorth [6], SpiNNaker 1 and 2 [7, 8], and BrainScaleS [9, 10]. These systems have demonstrated that spike-based computation can achieve orders-of-magnitude energy improvement over GPUs for temporal and event-driven workloads.

Yet the field faces a convergence problem. Deep learning has adopted attention mechanisms and tensor operations as standard primitives, but no neuromorphic processor provides hardware acceleration for these operations in the spike domain. Chiplet-based scaling has become the dominant path for high-transistor-count designs in conventional computing, but neuromorphic architectures remain monolithic. Post-quantum cryptography is increasingly required for secure hardware, yet no neuromorphic chip implements quantum-resistant key exchange. And the gap between neuroscience and silicon persists: biological neural circuits exhibit predictive coding, neurogenesis, sleep-dependent consolidation, gap junctions, glial modulation, oscillatory dynamics, and criticality-driven self-organisation, none of which have complete hardware implementations in existing processors.

This paper presents Catalyst N4, which addresses each of these gaps. Building on three prior generations—N1 (Loihi 1 parity) [1], N2 (full Loihi 2 feature parity) [2], and N3 (hardware virtualisation and silicon metaplasticity) [3]—N4 introduces a dual-chiplet architecture with 512 cores, 4.19M physical neurons expandable to 134.2M via 32-context TDM, variable-precision processing, and a complete suite of neuroscience hardware primitives.

Design pillars. N4 is organised around six design pillars:

1. **Spike-domain tensor acceleration.** A 16×16 Spike Tensor Core in every core performs conditional-add matrix operations with hardware sparsity intersection at 256 ops/cycle. An 8-head spiking attention mechanism with a 4-head, 64-depth KV cache provides query-key-value projections without converting to dense activations.
2. **Virtual neuron scaling.** Time-division multiplexing with 8–32 contexts per core expands 4.19M physical neurons to up to 134.2M virtual neurons ($512 \times 8{,}192 \times 32$). Variable precision (24/16/8/32-bit) trades state resolution for neuron count: 12,288 neurons at 16-bit, 16,384 at 8-bit.
3. **Dual-chiplet scaling.** Two Neural Compute Clusters of 256 cores each connect through a rich-club NoC topology with 16 express links per NCC, enabling 512-core operation and scaling to 64 chips (32,768 cores) via 12 AER links per chip.
4. **Hardware neuroscience primitives.** Hyperdimensional computing (1,024-bit, 64 registers), Hopfield associative memory (256 neurons, 256 patterns), working memory buffer (32 slots), gap junctions (256 entries), glial cell model (8 zones), oscillators (4 per tile), interneuron templates (4×8), event camera interface, and KV cache bring biologically-grounded computation to silicon.
5. **Post-quantum security.** AES-256-GCM spike encryption, CRYSTALS-Kyber ML-KEM-768 key encapsulation, SRAM PUF identity, root of trust with 8 PCR registers, lockstep execution, and SRAM repair with 264 physical / 256 active / 8 spare cores.
6. **Hardware operating system.** NeurOS manages 4,096 virtual network contexts with an RV64GC RISC-V subsystem (8 cores, 14 custom neuromorphic opcodes), quadrant scheduling, and DMA scatter-gather context switching.

Contributions. The contributions of this work are:

- A 512-core dual-chiplet neuromorphic processor with 4.19M physical neurons, 134.2M virtual neurons via 32-context TDM, variable precision (24/16/8/32-bit), 96-bit spike packets, 48 GB HBM3E, and a 150 W thermal budget.
- A 6-stage core pipeline with 4-way SMT barrel scheduling, 8-wide cohort SIMD, a dendritic traversal unit with 8 join operations, and a ternary bypass path.
- A 16×16 Spike Tensor Core (256 ops/cycle), 8-head spiking attention with 4-head 64-depth KV cache, 1,024-bit HDC engine, and 256-neuron Hopfield associative memory.
- Five neuron models plus LTC neurons, rebound firing, burst firing, divisive normalisation, winner-take-all, reservoir computing mode, Dale's law enforcement, approximate computing mode, and 8-bit gating signals.
- Eight synapse formats, 64 parameter groups of 39 parameters each, and a learning engine with 8 rules, 32 opcodes, 512-deep microcode, hardware backpropagation (8 layers, momentum, surrogate modes), federated learning (4 clients,

differential privacy), active forgetting, metacognition, convergence detection, and short-term plasticity.
- A 32-channel neuromodulation bus, 256-entry gap junctions, 8-zone glial model, 4 oscillators per tile with phase locking, 4 interneuron templates, 32-slot working memory buffer, and event camera interface (10-bit X/Y, ROI).
- A memory hierarchy with 256 KB L1, 64 KB shadow bank, 2 MB L2 (16-way), 640 MB S3RAM, synapse cache (64-entry), spike sorter (512-entry), TAGE prefetcher, PIM controller, and 10 memory interface types.
- A rich-club NoC with content-addressable routing (1,024-entry), 4 virtual channels, 16 express links, spike coalescing, and multi-chip scaling via 12 AER links with cross-chip sync, distributed error reporting, and distributed gradient aggregation.
- An RV64GC RISC-V subsystem with 14 custom opcodes, lockstep execution, SRAM repair (264/256/8), 8 PCR registers, 48 perf counters per tile, 16K trace buffer, 8-channel anomaly detector, CXL memory pool, and UCIe chiplet interconnect.
- FPGA validation achieving 126/126 hardware tests on AWS F2 at 62.5 MHz (14,983 ts/sec) and 100 MHz timing closure on Kria K26 at 0.378 W.
- Benchmark results: SHD (91.0%), SSC (76.4%), N-MNIST (99.2%), DVS Gesture (89.4%).

Paper organisation. Section 2 recaps the N3 baseline. Section 3 presents the dual-chiplet architecture. Section 4 details the core pipeline. Section 5 covers neuron models and dynamics. Section 6 describes time-division multiplexing and variable precision. Section 7 describes synapse formats. Section 8 presents the Spike Tensor Core. Section 9 covers spiking attention and KV cache. Section 10 describes the learning engine. Section 11 covers plasticity and stability. Section 12 presents hardware neuroscience primitives. Section 13 describes the NoC. Section 14 details the memory hierarchy. Section 15 covers predictive coding. Section 16 covers hardware sleep. Section 17 presents neurogenesis. Section 18 details the security subsystem. Section 19 describes the RISC-V subsystem. Section 20 covers power management. Section 21 presents system features. Section 22 describes I/O interfaces. Section 23 covers multi-chip scaling. Section 24 presents NeurOS. Section 25 describes N4-Edge. Section 26 presents FPGA validation. Section 27 evaluates benchmarks. Section 28 covers Kria K26 edge deployment. Section 29 presents ASIC projections. Section 30 describes the SDK. Section 31 surveys related work. Section 32 discusses limitations, and Section 33 concludes.

2 N3 Baseline

Catalyst N3 [3] established the foundation upon which N4 builds. The architecture comprises 128 cores organised into 16 tiles of 8 cores each, supporting 524,288 physical neurons at 24-bit precision or 4.2 million virtual neurons through hardware time-division multiplexing. Seven hardwired neuron models plus a 13-opcode custom ISA provide programmable dynamics. Four synapse formats (Full 72-bit, Inference 49-bit, Compact 20-bit, and FACTOR 35-bit low-rank) support variable-precision weights from 1 to 16 bits. Sixteen per-tile learning accelerators with a 28-opcode ISA v3.5, hardware metaplasticity via 3-bit consolidation, homeostatic plasticity via EWMA firing-rate tracking, and lazy eligibility decay via pre-computed lookup tables provide distributed learning without cross-chip bottlenecks.

N3 was validated on AWS F2 (Xilinx VU47P) at 62.5 MHz with 19/19 hardware tests passing and 14,512 timesteps per second. Benchmark evaluation achieved 91.0% on SHD (matching Loihi 2), 76.4% on SSC (exceeding Loihi 2's 69.8% by 6.6 points), and 99.2% on N-MNIST. The RTL comprised 46 Verilog files totalling approximately 17,700 lines with 897 assertions.

N3 proved that a single-developer architecture could exceed Intel's commercial processor in both feature breadth and benchmark accuracy. But N3 has limitations that motivate N4. The architecture is monolithic—128 cores on a single die, with no chiplet scaling path. It lacks tensor-level acceleration, attention mechanisms, hyperdimensional computing, and associative memory. Its NoC is a hierarchical fat tree without the rich-club topology that neuroscience suggests for efficient cortical communication. It provides no security primitives, no RISC-V management processor, and no gap junctions, glial models, or oscillatory circuits. Its learning engine lacks hardware backpropagation, federated learning, active forgetting, and convergence detection. N4 addresses each of these.

3 Architecture Overview

3.1 Dual-Chiplet Topology

N4 organises 512 neuromorphic cores into a dual-chiplet hierarchy (Figure 1):

$$\texttt{n4\_top} \rightarrow \texttt{n4\_chip} \rightarrow 2\times\texttt{n4\_ncc} \rightarrow 32\times\texttt{n4\_tile} \rightarrow 8\times\texttt{n4\_core} \tag{1}$$

Each NCC contains 256 cores (32 tiles in an 8×4 mesh of 8 cores each) and operates as an autonomous compute partition with its own power domain, clock tree, and NoC. Each NCC is further subdivided into 4 quadrants of 8 tiles each, managed by a quadrant scheduler. The two NCCs connect through a chip-level interconnect that carries inter-NCC spike traffic and synchronisation signals. Each NCC provides 8 spare cores for yield recovery via the SRAM repair subsystem (264 physical cores, 256 active, 8 spare).

3.2 Per-Core Architecture

Each N4 core is a substantial upgrade from the N3 core. The per-core feature set includes:

- **Fetch stage**: instruction fetch with TAGE spike predictor (4 history tables, 512-entry)

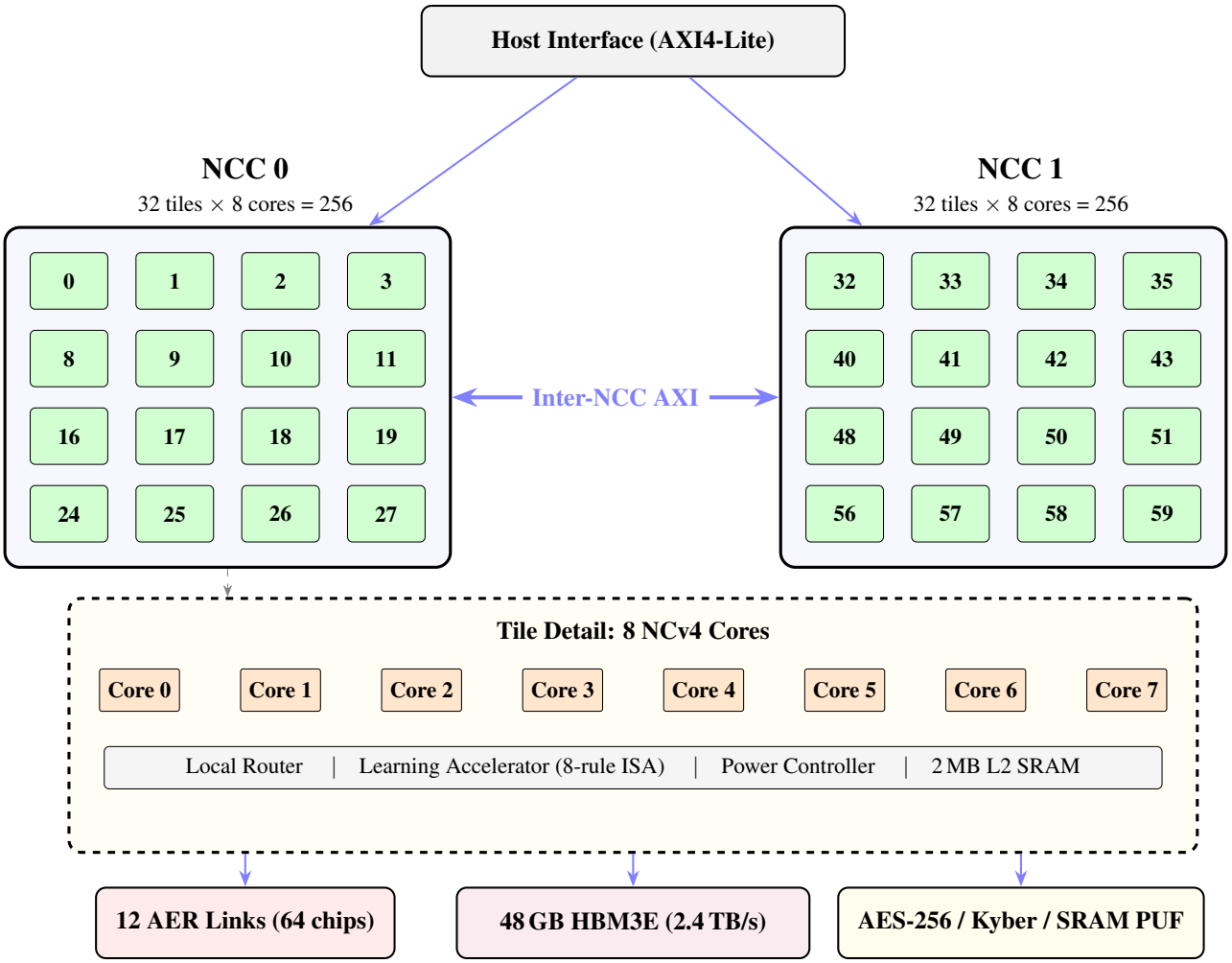


Figure 1: N4 system architecture. Two NCCs of 256 cores each connect via an inter-NCC AXI link (512 cores, 4.19M physical / 134.2M virtual neurons). Each NCC contains 32 tiles in an 8×4 mesh; a 4×4 subset is shown. Tile detail: 8 NCv4 cores share a local router, 8-rule learning accelerator, power controller, and 2 MB L2 SRAM. External interfaces: 48 GB HBM3E, 12 AER links (64-chip scaling), and post-quantum security.

- **Synapse stage**: 8 synapse format decoders including KAN B-spline
- **Update stage**: 5 neuron models (LIF, CUBA, ALIF, ANN INT8, Programmable 32-opcode with P-bit) plus LTC, burst, rebound, divisive normalisation
- **Writeback stage**: state commit with CRC-8 integrity and Dale's law enforcement
- **Emit stage**: spike emission with CRC-8, Dale's law polarity check
- **Learn stage**: 8-rule microcode learning engine (8 threads, 32 opcodes, 512-deep)
- **STC**: 16×16 Spike Tensor Core (256 ops/cycle)
- **SMT**: 4-way barrel simultaneous multithreading (56 bytes state per context)
- **Cohort**: 8-wide SIMD for population operations
- **DTU**: dendritic traversal unit for 16-compartment models with 8 join ops
- **Ternary bypass**: 1-cycle fast path for single spikes
- **Triple queue**: input (512), output (256), and learning spike queues
- **TDM**: 8–32 context time-division multiplexing
- **Variable precision**: 24/16/8/32-bit neuron state
- **L1 memory**: 256 KB neuron state + 64 KB shadow bank for TDM
- **KV cache**: 4-head, 64-depth transformer-style cache
- **HDC**: 1,024-bit hyperdimensional computing engine (64 regs, 8 ops)
- **Hopfield**: 256-neuron associative memory (256 patterns, energy recall)
- **Working memory**: 32 slots × 256-bit patterns with decay
- **Gap junctions**: 256-entry bidirectional coupling table
- **Oscillators**: 4 per tile, configurable frequency/amplitude/waveform, phase locking
- **Glial model**: 8 zones, calcium-gated modulation
- **Interneuron templates**: 4 templates × 8 targets
- **Event camera**: 10-bit X/Y, ROI filtering, pixel-to-neuron mapping
- **Synapse cache**: 64-entry, 512-bit lines (32 KB total)
- **Spike sorter**: 512-entry priority queue
- **64 parameter groups**: 39 configurable parameters per

group
- **2,048-entry CAM**: content-addressable synapse lookup
- **2,048-entry free list**: dynamic synapse allocation

3.3 Headline Numbers

Table 1 summarises the key architectural parameters, all sourced from the RTL parameter file (`n4_params.vh`).

3.4 Spike Packet Format

N4 uses 96-bit spike packets:

The 14-bit chip ID field supports up to 16,384 chips in extended configurations. The 12-bit core ID addresses 4,096 cores (8× the 512 in a single chip), providing headroom for multi-chip routing.

Table 1: N4 architectural parameters.

Parameter	Value
NCCs	2
Tiles per NCC	32 (8×4)
Cores per tile	8
Total cores	512
Spare cores per NCC	8
Neurons per core (24-bit)	8,192
Neurons per core (16-bit)	12,288
Neurons per core (8-bit)	16,384
Physical neurons (24-bit)	4,194,304
TDM factor (min/max)	8 / 32
Virtual neurons (max TDM)	134,217,728
Parameter groups	64
Params per group	39
Compartments	16
Dendritic join operations	8
Neuromodulation channels	32
Spike width	96 bits
Pipeline stages	6
SMT contexts	4
Cohort SIMD width	8
STC dimension	16×16
Attention heads	8
KV cache (heads/depth)	4 / 64
HDC dimension	1,024 bits
Hopfield neurons/patterns	256 / 256
L1 per core	256 KB + 64 KB shadow
L2 per tile (16-way)	2 MB
S3RAM	640 MB (5-cycle)
HBM3E	48 GB
Synapse cache	64-entry, 512-bit
Learning threads	8
Learning opcodes	32
Microcode depth	512
Learning registers	256
Express links per NCC	16
Virtual channels	4
AER links	12
Max chips	64
Max VNETs	4,096
RV64GC cores	8
Custom RISC-V opcodes	14
Perf counters per tile	48
Total perf counters	3,072
Trace buffer entries	16,384
Thermal budget	150 W
NCC die area	220 mm^2
IO die area	100 mm^2
Package area	1,800 mm^2
BGA balls	1,198

Table 2: N4 96-bit spike packet format.

Field	Bits	Description
Type	2	unicast/coalesced/multicast/control
Chip ID	14	source chip (0–16,383)
Core ID	12	source core (0–4,095)
Neuron ID	16	neuron within core
Value	16	graded spike payload
Delay	8	axonal delay (0–255 ts)
Tag	10	routing / learning tag
Priority	2	routing priority (0–3)
Metadata	16	security, polarity, type, CRC

4 Core Pipeline

4.1 Six-Stage Design

N4 extends N3's 5-stage pipeline to 6 stages (Figure 2), separating spike emission from writeback to reduce critical-path length:

1. **Fetch (F):** Read neuron state from L1 SRAM, check activity bitmap, query TAGE spike predictor. If the predictor indicates no spike and no synaptic input arrived, the neuron is skipped.
2. **Synapse (S):** Decode synapse format, fetch weights, perform synaptic integration. The 8-format decoder handles Full, Inference, Compact, Factor, Dual-Weight, Block-Sparse, Delta, and KAN B-spline formats in parallel paths. The synapse cache (64-entry, 512-bit lines) reduces L2 traffic for frequently accessed weight blocks.
3. **Update (U):** Execute neuron model computation. The model selector (3-bit field from the parameter group) routes to one of 5 hardwired datapaths, the LTC engine, burst/rebound/divisive normalisation paths, or the programmable microcode engine.
4. **Writeback (W):** Commit updated neuron state to L1 SRAM with CRC-8 generation. Dale's law enforcement checks that excitatory neurons emit only positive spikes and inhibitory neurons only negative spikes. Winner-take-all evaluation runs across the parameter group.
5. **Emit (E):** Generate spike packet, insert into output queue (256-deep) with priority, apply security tag if encryption is enabled. Spike coalescing may merge this spike with others sharing the same destination tile.
6. **Learn (L):** Execute learning microcode program if the synapse group has learning enabled. Access traces, rewards, and metaplastic state. The 8-thread barrel design processes 8 synapse groups in parallel.

4.2 4-Way SMT Barrel Scheduling

N4 implements simultaneous multithreading via a barrel processor design. Four hardware threads (T0–T3) are interleaved across the 6-stage pipeline, with each thread occupying one pipeline stage per cycle. Each SMT context requires 56 bytes of state (`SMT_STATE_BYTES=56`), for a total of 224 bytes per core. The barrel advances all four threads every cycle in round-robin order, eliminating pipeline hazards between threads by construction.

Each thread maintains its own register context: program counter, neuron state pointer, accumulator, and 16 general-purpose registers. The shared resources—ALU, multiplier, SRAM read/write ports—are arbitrated by the barrel scheduler with zero-overhead context switching (the context is the thread index selecting a register bank). The barrel design achieves a full $4\times$ throughput improvement over single-threaded operation.

4.3 8-Wide Cohort SIMD

The cohort SIMD unit processes 8 neurons simultaneously using a single instruction stream. The SIMD width of 8 matches the number of cores per tile. Cohort mode is selected per parameter group via a 1-bit configuration flag. When enabled, the fetch stage loads 8 neuron state vectors simultaneously from consecutive SRAM addresses. Inactive neurons within the cohort are masked, consuming zero dynamic power via per-lane clock gating. The 4-way SIMD datapaths (`SIMD_WIDTH=4`) within each lane provide additional sub-word parallelism.

4.4 Dendritic Traversal Unit

The DTU implements multi-compartment neuron models by traversing a tree-structured dendritic morphology stored in L1 SRAM. Each neuron supports up to 16 compartments (`COMPARTMENTS=16`), each identified by a 4-bit ID (`COMPARTMENT_BITS=4`) with a 5-bit parent pointer (`PARENT_PTR_BITS=5`).

The DTU supports 8 dendritic join operations, selectable per compartment via a 4-bit opcode:

Table 3: Dendritic join operations.

Code	Name	Description
0	SUM	additive integration
1	MAX	winner-take-all
2	PRODUCT	multiplicative gating
3	AND	coincidence detection (all inputs)
4	XOR	parity / phase detection
5	GATED	modulated by gating signal
6	SUPRA	supralinear amplification
7	COINC	temporal coincidence (within window)

The traversal completes in N_{comp} cycles for a neuron with N_{comp} compartments. The somatic compartment (root of the tree) generates the spike output. For single-compartment neurons, the DTU is bypassed via the ternary path.

4.5 Ternary Bypass Path

The ternary bypass exploits a common case in spiking networks: a neuron receives a single spike, the accumulated input

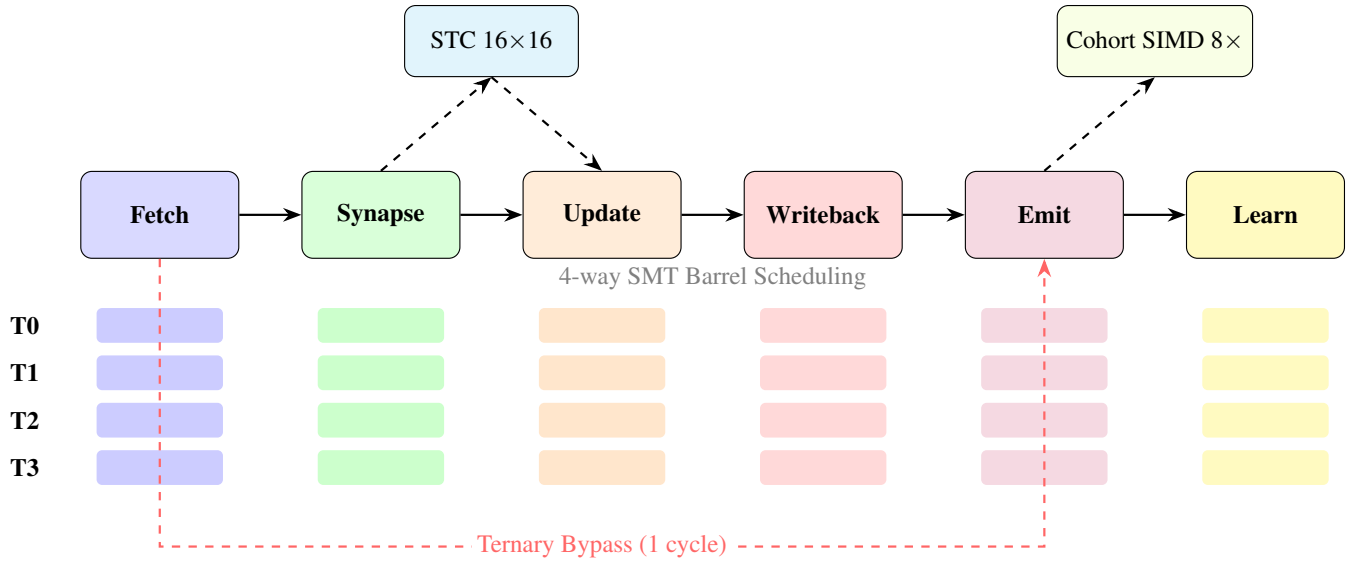


Figure 2: N4 6-stage core pipeline with 4-way SMT barrel scheduling. Four threads (T0–T3) occupy different pipeline stages simultaneously. The ternary bypass path shortcuts from Fetch directly to Emit for single-spike neurons, reducing latency from 6 cycles to 1. The Spike Tensor Core and Cohort SIMD unit attach as parallel datapaths.

crosses threshold, and the neuron fires. The bypass performs a single check:

$$\text{bypass} = (v[t-1] + w_{\text{single}} \geq \theta) \wedge (\text{refrac} = 0) \wedge (\text{fan\_in} = 1) \tag{2}$$

When all three conditions hold, the output spike is generated directly from the Fetch stage, reducing latency from 6 cycles to 1. The bypass is conservative: any ambiguity routes through the full pipeline.

5 Neuron Models and Dynamics

N4 provides five primary neuron models selectable per parameter group (Table 4), plus additional dynamics (LTC, burst, rebound, divisive normalisation) that overlay the primary models.

Table 4: N4 primary neuron models.

#	Model	Precision	Key Feature
0	LIF	24/16/8/32-bit	Loihi-compatible baseline
1	CUBA	24/16/8/32-bit	Dendritic compartments
2	ALIF	24/16/8/32-bit	Adaptive threshold
3	ANN INT8	8-bit	4-wide MAC pipeline
4	Programmable	24-bit	32-opcode ISA + P-bit

5.1 LIF and CUBA (Models 0–1)

The LIF model implements standard leaky integrate-and-fire dynamics with round-away-from-zero fixed-point arithmetic, maintaining backward compatibility with N1, N2, and N3:

$$u[t] = u[t-1] - \text{RAZ}(\texttt{decay\_u} \cdot u) + I_{\text{syn}} \tag{3}$$

$$v[t] = v[t-1] - \text{RAZ}(\texttt{decay\_v} \cdot v) + u + \texttt{bias} \tag{4}$$

A spike is emitted when $v[t] \geq \theta$ and the refractory counter is zero. The CUBA variant extends LIF with dendritic compartments processed by the DTU.

5.2 Adaptive LIF (Model 2)

The ALIF model adds spike-frequency adaptation via a dynamic threshold [12]:

$$\theta_{\text{adapt}}[t] = \theta_{\text{adapt}}[t-1] - \text{RAZ}(\texttt{decay\_}\theta \cdot \theta_{\text{adapt}}) + s[t] \cdot \Delta\theta \tag{5}$$

$$\theta_{\text{eff}}[t] = \theta_{\text{base}} + \theta_{\text{adapt}}[t] \tag{6}$$

ALIF neurons are used in the SHD and SSC benchmark models (Section 27).

5.3 ANN INT8 Mode (Model 3)

The ANN model provides a 4-wide INT8 multiply-accumulate pipeline with peak throughput of 16 MACs/cycle (4 lanes × 4 SMT threads):

$$\text{acc} = \sum_{i=0}^{3} w_i \cdot x_i, \qquad \text{output} = \text{clamp}(\text{acc} \gg \text{shift},\ 0,\ 255) \tag{7}$$

This mode enables hybrid SNN/ANN architectures on a single chip.

5.4 Programmable Model with P-Bit (Model 4)

The programmable neuron model provides a 32-opcode instruction set (Table 5). The expanded ISA adds bitwise operations, conditional moves, table lookups, and a probabilistic bit (P-bit) instruction. The `PBIT` opcode generates a probabilistic binary output using a sigmoid lookup table approximation with a per-core 32-bit LFSR.

Table 5: N4 programmable neuron ISA (32 opcodes). New opcodes relative to N3 marked with $*$.

Op	Syn	Desc	Op	Syn	Desc
ADD	Rd,Rs1,Rs2	Add	NOT*	Rd,Rs	Bitwise NOT
SUB	Rd,Rs1,Rs2	Subtract	ABS*	Rd,Rs	Abs. value
MUL≫8	Rd,Rs1,Rs2	Fixed mul	NEG*	Rd,Rs	Negate
SHR	Rd,Rs,imm	Shift R	CMOV*	Rd,Rs,Rc	Cond. move
SHL	Rd,Rs,imm	Shift L	MIN*	Rd,Rs1,Rs2	Minimum
CLAMP	Rd,Rs,lo,hi	Sat. clamp	MAX*	Rd,Rs1,Rs2	Maximum
LOAD_I	Rd,imm16	Load imm	MULACC*	Rd,Rs1,Rs2	Fused MAC
LOAD_A	Rd	Load acc	LOOKUP*	Rd,Rs,tbl	Table (256)
STORE_V	Rs	Write v	PBIT*	Rd,Rs	Prob. bit
CMP_GT	Rd,Rs1,Rs2	Compare $>$	RNG*	Rd	Random
BRANCH	Rs,off	Cond. jump	LOAD_T*	Rd,tid	Load trace
SPIKE	—	Emit spike	STORE_T*	Rs,tid	Write trace
HALT	—	End update	LOAD_P*	Rd,pid	Load param
AND*	Rd,Rs1,Rs2	Bitwise AND	CMP_EQ*	Rd,Rs1,Rs2	Compare $=$
OR*	Rd,Rs1,Rs2	Bitwise OR	CMP_LT*	Rd,Rs1,Rs2	Compare $<$
XOR*	Rd,Rs1,Rs2	Bitwise XOR	NOP*	—	No-op

The programmable model stores up to 64 instructions per thread in the microcode region of L1. An Izhikevich neuron can be implemented in 14 instructions; a resonate-and-fire model in 18; a Boltzmann machine sampler using P-bit in 22.

5.5 LTC (Liquid Time-Constant) Neurons

N4 adds LTC neurons (`FEATURE_LTC=1`), where the time constant adapts continuously based on input:

$$\tau(t) = \tau_{\text{min}} + (\tau_{\text{max}} - \tau_{\text{min}}) \cdot \sigma(I_{\text{syn}}) \tag{8}$$

The time constant varies dynamically between τ_{min} and τ_{max}, modulated by the sigmoid of the synaptic input current. This enables the neuron to process fast transients with short time constants and slow background signals with long time constants.

5.6 Burst Firing

Burst firing (`FEATURE_BURST=1`) generates a configurable number of spikes per threshold crossing. When the membrane potential exceeds a burst threshold (separate from the primary threshold), the neuron emits k spikes at k-timestep intervals, with decreasing amplitude. The burst count k is configurable per parameter group (2–8 spikes per burst). Burst mode is used for attention-gated signal amplification.

5.7 Rebound Firing

Post-inhibitory rebound (`FEATURE_PIR=1`) generates a spike after release from strong inhibition. When the membrane potential drops below a configurable rebound threshold and then recovers above it, a rebound spike is emitted. This implements a biologically-observed mechanism where inhibitory input can trigger excitatory output.

5.8 Divisive Normalisation

Divisive normalisation (`FEATURE_DIVISIVE_NORM=1`) scales each neuron's input by the aggregate activity of its population:

$$I_{\text{norm}} = \frac{I_{\text{syn}}}{\sigma^2 + \sum_{j \in \text{group}} r_j} \tag{9}$$

where r_j is the firing rate of neuron j in the same parameter group and σ^2 is a semi-saturation constant. This implements contrast gain control, a canonical cortical computation.

5.9 Winner-Take-All

A hardware WTA circuit (`FEATURE_CEREBELLAR=1`) identifies the k neurons with the highest membrane potential within a parameter group and suppresses all others. The value of k is configurable (1–8). WTA is used in classification readout layers and competitive learning.

5.10 Reservoir Computing Mode

Reservoir computing mode fixes the recurrent weights and trains only the readout layer. When enabled, the synapse stage skips weight updates for recurrent connections, the learning engine operates only on readout synapses, and the neuron pipeline runs at maximum throughput since no learning overhead is incurred on the recurrent path.

5.11 Approximate Computing Mode

Approximate computing (`FEATURE_PBIT=1`) relaxes numerical precision to 8-bit computation with a configurable error margin. When the membrane potential is more than the margin threshold below the firing threshold, approximate mode skips the full-precision update and applies a fast 8-bit estimate. This reduces dynamic power by approximately 40% for sub-threshold neurons.

5.12 Gating Signal

Each neuron receives an 8-bit neuromodulatory gating signal from the 32-channel neuromodulation bus. The gating signal multiplicatively modulates the synaptic input:

$$I_{\text{gated}} = I_{\text{syn}} \cdot \frac{g}{255} \tag{10}$$

where g is the 8-bit gating value. Gating enables attention-like selective amplification controlled by top-down neuromodulatory signals.

5.13 Dale's Law Enforcement

Each neuron has a 1-bit polarity flag (excitatory or inhibitory) that is enforced in hardware. In the writeback stage, any weight update that would violate Dale's law (producing a negative weight on an excitatory synapse, or positive on an inhibitory one) is clamped to zero. In the emit stage, the spike polarity field is set from the neuron's type, preventing mixed excitatory/inhibitory output from a single neuron.

6 Time-Division Multiplexing and Variable Precision

6.1 TDM Architecture

Time-division multiplexing (`FEATURE_TDM=1`) extends each physical core to support 8–32 virtual contexts (`TDM_FACTOR_MIN=8`, `TDM_FACTOR_MAX=32`). Each context represents an independent set of 8,192 neurons sharing the core's pipeline but with separate state stored in the shadow bank.

The TDM controller (`n4_tdm_ctrl`) implements a 4-state FSM (IDLE, RUN, SWITCH, DONE) that cycles through active contexts. Each timestep, the controller runs a context until `timestep_done`, then saves the context to the shadow bank, restores the next context, and repeats. The save/restore operations are pipelined with a double-buffering scheme to overlap computation with state transfer.

At maximum TDM factor (32 contexts):

$$\text{Virtual neurons} = 512 \times 8{,}192 \times 32 = 134{,}217{,}728 \tag{11}$$

The trade-off is temporal: a 32-context TDM core processes each context at $1/32$ of the base timestep rate. For applications where the neural time constant is much longer than the hardware timestep (e.g., slow cortical rhythms), TDM provides a large capacity increase at no accuracy cost.

6.2 Variable Precision

Variable precision (`PRECISION_24`, `PRECISION_16`, `PRECISION_8`, `PRECISION_32`) trades state resolution for neuron count within fixed L1 SRAM:

Table 6: Variable precision modes.

Precision	Neurons/core	Use case
24-bit (default)	8,192	high-fidelity dynamics
16-bit	12,288	inference with moderate accuracy
8-bit	16,384	maximum density inference
32-bit	(custom)	full-precision research

Precision is configurable per parameter group, allowing mixed-precision operation within a single core: 24-bit for recurrent layers requiring temporal fidelity, 8-bit for feedforward inference layers requiring maximum density.

6.3 Shadow Bank

Each core has a 64 KB shadow SRAM bank (`SHADOW_SIZE_KB=64`) that provides double-buffered TDM context switching. While the active context runs from the primary 256 KB L1 bank, the next context's state is pre-loaded into the shadow bank. A single-cycle bank swap (toggling the shadow bank select register) completes the context switch. This overlaps data movement with computation, reducing TDM overhead to approximately 1 cycle per context switch.

7 Synapse Architecture

N4 doubles the number of synapse formats from N3's 4 to 8 (Table 7), adding Dual-Weight, Block-Sparse, Delta, and KAN B-spline formats.

Table 7: N4 synapse word formats.

Format	Bits	Description
Full	72	Weight, target, delay, fatigue, meta, traces
Inference	49	Weight, target, delay, control (no learning)
Compact	20	Weight + target only (max density)
Factor	35	Low-rank $\mathbf{AB}$ factorisation
Dual-Weight	56	Separate fast/slow weights
Block-Sparse	40	Block index + CSR pointers
Delta	24	Differential encoding
KAN B-spline	64	Learnable activation via B-spline

7.1 Dual-Weight Format

The Dual-Weight format (56 bits) stores two independent weight values per synapse:

$$w_{\text{eff}} = \alpha \cdot w_{\text{fast}} + (1 - \alpha) \cdot w_{\text{slow}} \tag{12}$$

where w_{fast} (8-bit) adapts rapidly and w_{slow} (8-bit) consolidates gradually. During hardware sleep (Section 16), w_{slow} is

updated toward w_{fast} at a configurable consolidation rate, implementing offline memory transfer [16].

7.2 Block-Sparse Format

The Block-Sparse format (40 bits) implements structured sparsity via CSR encoding at the block level. Weights are organised into blocks of configurable size (4×4, 8×8, or 16×16). Each entry stores a block index (16-bit), column pointer (12-bit), and block metadata (12-bit). The STC operates directly on Block-Sparse formatted weights without decompression.

7.3 Delta Encoding Format

The Delta format (24 bits) stores weight differences relative to a per-group base weight:

$$w_i = w_{\text{base}} + \delta_i \tag{13}$$

where δ_i is a 4-bit signed offset, providing 3.6× compression relative to the Full format.

7.4 KAN B-Spline Synapses

The KAN synapse format implements learnable activation functions via cubic B-spline basis functions:

$$\phi(x) = \sum_{k=0}^{K-1} c_k \cdot B_k(x) \tag{14}$$

The 64-bit KAN synapse word stores target neuron (12-bit), spline order (2-bit, selecting $K \in \{4, 6, 8, 10\}$), base address into the control point table (16-bit), and metadata (34-bit). Control point tables reside in L2 shared memory. The de Boor evaluation pipeline produces results in 4 cycles for cubic splines. KAN synapses enable the network to learn arbitrary nonlinear transformations per synapse.

7.5 Factor Decomposition

The Factor format (35 bits, rank-32, `SYN_FACTOR=3'd3`) stores synaptic weights as the outer product of two rank-32 vectors:

$$\mathbf{W} \approx \mathbf{A}\mathbf{B}^\top, \qquad \mathbf{A} \in \mathbb{R}^{m\times 32},\ \mathbf{B} \in \mathbb{R}^{n\times 32} \tag{15}$$

Factor decomposition reduces storage from $O(mn)$ to $O(32(m+n))$. The learning engine supports factor trace decomposition (`FEATURE_FACTOR=1`) that updates the factor matrices directly.

7.6 Variable-Precision Weights

Weight precision remains configurable per synapse group from 1 to 16 bits (`WEIGHT_BITS_FULL=16`, `WEIGHT_BITS_COMPACT=8`, `WEIGHT_BITS_MICRO=4`). Dense packing (four 4-bit weights per 16-bit word, two 8-bit per 16-bit word) is preserved.

8 Spike Tensor Core

8.1 Architecture

The Spike Tensor Core (STC) is a dedicated matrix-operation unit that performs conditional-add operations in the spike domain. It operates on 16×16 tiles (`STC_DIM=16`, `STC_OPS_PER_CYCLE=256`):

$$\mathbf{o} = \sum_{i:s_i=1} \mathbf{W}[i,:] \tag{16}$$

8.2 Sparsity Intersection

The STC maintains a 16×16 bitmap of nonzero weights. At operation time:

$$\text{active}[i][j] = s_i \wedge \text{nz}[i][j] \tag{17}$$

Only entries where both the spike is active and the weight is nonzero enter the addition tree. For a network with 10% spike activity and 50% weight pruning, the STC draws approximately 5% of peak power.

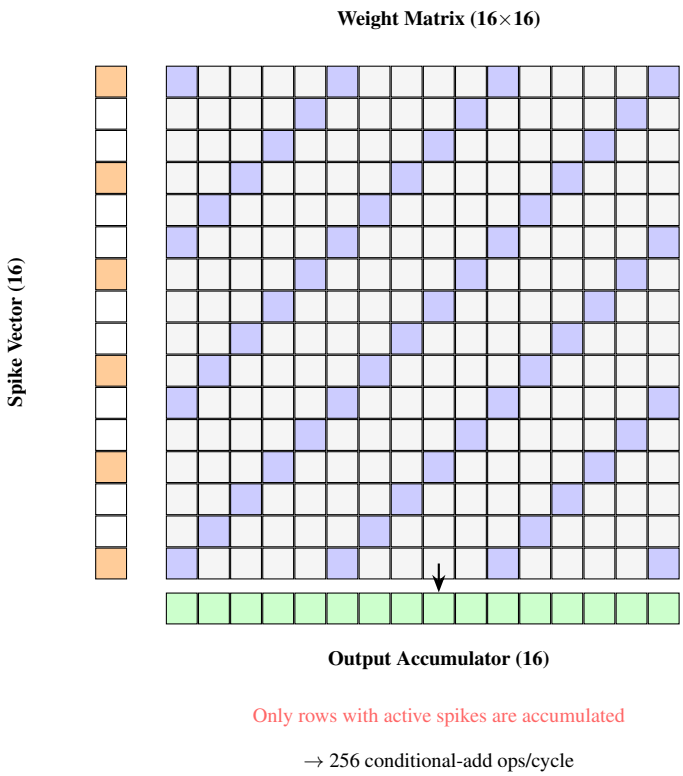


Figure 3: Spike Tensor Core operation. A 16-element binary spike vector selects rows of a 16×16 weight matrix. Only rows corresponding to active spikes are accumulated into the output vector. Hardware sparsity intersection skips zero entries, achieving up to 256 operations per cycle at full density.

9 Spiking Attention and KV Cache

9.1 8-Head Mechanism

N4 implements an 8-head spiking attention mechanism (`ATTENTION_HEADS=8`) that operates on spike vectors:

$$\mathbf{q}_h = \mathbf{W}_Q^h \cdot \mathbf{s}_{\text{query}} \tag{18}$$

$$\mathbf{k}_h = \mathbf{W}_K^h \cdot \mathbf{s}_{\text{key}} \tag{19}$$

$$\mathbf{v}_h = \mathbf{W}_V^h \cdot \mathbf{s}_{\text{value}} \tag{20}$$

The QKV projections are computed by the STC. The attention score computation replaces softmax with a spike-compatible thresholded operation:

$$\alpha_{ij} = \text{thresh}\left(\frac{\mathbf{q}_i^\top \mathbf{k}_j}{\sqrt{d_h}}\right) \tag{21}$$

The 8 heads operate in parallel, each using a separate STC tile for QKV projection. Total latency is 4 STC cycles per attention operation.

9.2 KV Cache

The KV cache (`n4_kv_cache`) provides a 4-head, 64-depth transformer-style cache for autoregressive-style inference. Parameters from the RTL:

- `NUM_HEADS=4`: 4 attention heads cached simultaneously
- `CACHE_DEPTH=64`: 64 key-value pairs per head
- `DIM=16`: 16-dimensional key and value vectors
- `DATA_W=16`: 16-bit data width

The KV cache stores key and value vectors as they arrive and supports score computation, softmax approximation, weighted accumulation, and LRU eviction when full. This enables transformer-style sequence processing in the spike domain without re-computing attention over the entire sequence at each timestep.

9.3 Coincidence Detection

The attention subsystem includes 256 coincidence counters (`COINCIDENCE_COUNTERS=256`, `COINCIDENCE_BITS=8`) that detect synchronous spike arrivals within a configurable temporal window. Coincidence counts feed into the attention score computation, enabling the hardware to learn which spike sources are temporally correlated.

10 Learning Engine

10.1 8-Rule Microcode System

N4 expands the learning engine to an 8-rule microcode system with substantially increased capacity:

- **8 learning threads** (`LEARN_THREADS=8`), each processing synapse groups independently
- **32 learning opcodes** (`LEARN_OPCODES=32`, Table 9)
- **32 registers per thread** (`LEARN_REGS_PER_THREAD=32`), 256 total (`LEARN_TOTAL_REGS=256`)
- **512-deep microcode memory** (`MICROCODE_DEPTH=512`) with 32-bit instructions
- **6 trace types** (`NUM_TRACES=6`) at 12-bit and 8-bit widths, with 4-bit decay timestamp

The microcode memory is segmented into 7 base regions, each at a fixed offset:

Table 8: Microcode memory regions.

Region	Base Address
LTD	0
LTP	64
Reward-modulated	128
Surrogate gradient	192
Structural plasticity	256
Custom rules	320
Neuron update	448

10.2 Learning Opcodes

Table 9 lists the full 32-opcode learning ISA:

Table 9: Learning engine ISA (32 opcodes).

Hex	Op	Desc	Hex	Op	Desc
00	NOP	No-op	10	STORE_M	Meta state
01	ADD	Add	11	STORE_F	Fatigue
02	SUB	Subtract	12	ABS	Abs. value
03	MULS	Mul$\gg$8	13	NEG	Negate
04	SHR	Shift R	14	CLAMP	Sat. clamp
05	SHL	Shift L	15	SKIP_GT	Skip if $>$
06	MAX	Maximum	16	MULACC	Fused MAC
07	MIN	Minimum	17	RNG	Random
08	LOADI	Load imm	18	STORE_A	Aux A
09	STORE_W	Weight Δ	19	STORE_B	Aux B
0A	STORE_E	Eligibility	1A	SCALE_W	Learn rate
0B	SKIP_Z	Skip if 0	1B	LOAD_R	Load rate
0C	SKIP_NZ	Skip if $\neq$0	1C	SURR_G	Surr. grad
0D	HALT	End prog	1D	CHAIN_R	Backprop
0E	STORE_D	Delay	1E	S_PRUNE	Prune
0F	STORE_T	Trace	1F	S_GROW	Grow

10.3 Learning Rule Example

The following demonstrates a three-factor reward-modulated STDP rule:

Listing 1: Three-factor STDP in N4 learning ISA.

```
; Rule 0: Reward-modulated STDP with
;         metaplasticity and fatigue
    LOAD R0, x1           ; pre-synaptic trace
    LOAD R1, y1           ; post-synaptic trace
    MULACC R2, R0, R1     ; eligibility = pre * post
    LOAD_REWARD R3        ; reward (channel 0)
    MUL R4, R2, R3        ; modulated update
    LOAD_FATIGUE R5       ; current fatigue
    SUB R4, R4, R5        ; reduce by fatigue
    SCALE_W R6, R4        ; apply learning rate
    CLAMP R6, -128, 127   ; saturate
    STORE R6              ; commit weight delta
    STORE_M R6            ; update metaplasticity
    HALT
```

10.4 Neuromodulation

N4 provides 32 neuromodulation channels (`NEUROMOD_CHANNELS=32`). The neuromodulation tree has three levels:

1. **Global** (chip-level): 4 channels broadcast to all NCCs from the host interface or a designated reward neuron group.
2. **NCC-level**: 8 channels per NCC, with per-tile masking.
3. **Tile-level**: 32 channels per tile, configurable per parameter group.

A learning program accesses neuromodulation via the `LOAD_REWARD` opcode with a 5-bit channel selector.

10.5 Hardware Backpropagation

The backpropagation engine (`n4_backprop`) supports gradient computation through up to 8 layers:

- **8-layer support** (`NUM_LAYERS=8`), processing forward activations and backward error signals
- **Momentum:** configurable momentum coefficient for weight updates
- **Surrogate gradient modes:** 4 modes (2-bit `cfg_surrogate_mode`) selecting different surrogate gradient functions for non-differentiable spike thresholds
- **Per-neuron deltas:** the engine produces per-weight delta values (`delta_weight`) with source, destination, and layer indices

Backpropagation is triggered by target presentation: the forward pass stores activations per layer, the backward pass propagates error signals from the output layer through each hidden layer using the chain rule. This provides true gradient-based training on-chip for small networks (up to 64 neurons per layer).

10.6 Federated Learning

The federated learning engine (`n4_federated`) aggregates local weight updates from up to 4 clients with configurable differential privacy:

- **4 clients** (`NUM_CLIENTS=4`), each with independent parameter buffers
- **Differential privacy:** configurable noise scale (`cfg_dp_noise_scale`), injected during aggregation via

LFSR-generated noise

- **Minimum quorum:** aggregation proceeds only when `cfg_min_clients` have submitted updates
- **Aggregation:** averaging across client buffers with optional noise injection

In multi-chip configurations, each chip acts as a federated client, enabling privacy-preserving distributed learning across physically separate chips.

10.7 Active Forgetting

Active forgetting (`n4_active_forget`) provides tag-based selective weight decay:

- **Tag matching:** a 10-bit tag mask selects which synapses to decay
- **Configurable rate:** 8-bit `forget_rate` controls decay speed
- **Exemption:** synapses with metaplastic state above `exempt_threshold` are protected from forgetting
- **Batch processing:** scans 8 synapses per cycle

Active forgetting enables continual learning: old memories that have not been consolidated (low metaplastic state) are selectively weakened to make room for new learning.

10.8 Metacognition

The metacognition monitor (`n4_metacog`) computes 4 real-time metrics:

1. **Weight rate:** EWMA of total weight change per timestep
2. **Spike entropy:** approximation of firing-pattern randomness
3. **Prediction accuracy:** ratio of TAGE predictor hits to total predictions
4. **Structural balance:** ratio of grow operations to prune operations

Each metric has a configurable threshold; crossing any threshold triggers an IRQ (`irq_threshold`). The host or RISC-V subsystem can use these signals to adjust learning rates, switch learning rules, or trigger sleep states.

10.9 Convergence Detection

The convergence detector (`n4_convergence`) monitors spike count variance over a sliding window (16/64/256/1024 timesteps). When variance drops below a configurable threshold, it asserts `converged_irq`, signalling that the network has reached a stable state. The RISC-V subsystem or host can use this IRQ to terminate training epochs or switch to inference mode.

10.10 Short-Term Plasticity

Short-term plasticity provides both facilitation and depression (`FEATURE_FACILITATION=1`, `FEATURE_FATIGUE=1`):

- **Facilitation:** 4-bit per-synapse counter (`FACILITATION_BITS=4`) that increases effective weight on repeated activation. Decays exponentially when inactive.
- **Depression (fatigue):** 4-bit per-synapse counter (`FATIGUE_BITS=4`) that decreases effective weight. At maximum fatigue (15), effective weight is zero.
- Both operate without learning engine intervention, providing automatic temporal filtering.

11 Plasticity and Stability

11.1 Metaplasticity

N4 retains N3's 3-bit per-synapse consolidation counter (`META_STATE_BITS=3`):

$$\eta_{\text{eff}} = \eta \cdot (1 + \alpha_{\text{meta}} \cdot \text{consolidation}) \tag{22}$$

Consistent reinforcement increments the counter toward 7, increasing learning rate and resistance to overwriting. Sign reversal resets the counter.

11.2 Homeostatic Firing-Rate Regulation

Per-neuron-group EWMA firing-rate tracking maintains stable activity:

$$\text{rate}[t] = \text{rate}[t-1] - (\text{rate}[t-1] \gg \tau_h) + (s[t] \cdot (255 \gg \tau_h)) \tag{23}$$

At epoch boundaries, the hardware adjusts the firing threshold:

$$\theta[t] = \theta[t] + \alpha_h \cdot (\text{rate}[t] - \text{rate}_{\text{target}}) \tag{24}$$

11.3 Memory Consolidation

During sleep states (Section 16), the Dual-Weight synapse format transfers information from fast weights to slow weights:

$$w_{\text{slow}}[t+1] = w_{\text{slow}}[t] + \beta \cdot (w_{\text{fast}}[t] - w_{\text{slow}}[t]) \tag{25}$$

11.4 Factor Decomposition Traces

For Factor-format synapses (rank-32), the learning engine maintains decomposed eligibility traces that update the factor matrices $\mathbf{A}$ and $\mathbf{B}$ directly, avoiding the $O(mn)$ cost of full-rank trace storage.

12 Hardware Neuroscience Primitives

N4 includes a suite of neuroscience-inspired hardware modules, each implemented as a dedicated RTL block.

12.1 Hyperdimensional Computing Engine

The HDC engine (`n4_hdc`, `FEATURE_HDC=1`) implements operations on 1,024-bit hyperdimensional vectors (`HDC_DIM=1024`):

- **64 HV registers** (`NUM_REGS=64`), each 1,024 bits, stored as 16×64-bit words
- **8 operations** via 3-bit opcode: BIND (XOR), BUNDLE (majority vote), PERMUTE (cyclic shift), SIMILARITY (Hamming distance), and HASH
- **Hamming distance** computed via popcount-64 across 16 words
- **Bundle accumulator:** 4-bit per-dimension accumulator for multi-vector bundling

HDC provides a framework for symbolic reasoning on binary data. A classification task maps input features to hypervectors, bundles class exemplars, and identifies the closest match via Hamming distance. The 1,024-bit dimension provides theoretical capacity for hundreds of classes.

12.2 Hopfield Associative Memory

The Hopfield network (`n4_hopfield`, `FEATURE_HOPFIELD=1`) provides content-addressable associative recall:

- **256 binary neurons** (`NUM_NEURONS=256`) with 8-bit weights
- **256 stored patterns** (`NUM_PATTERNS=256`) via Hebbian outer-product learning
- **Energy-based recall:** iterative state update converges to the stored pattern nearest the input cue
- **SRAM-backed weight matrix:** $256 \times 256 = 65{,}536$ entries in dedicated SRAM
- **Configurable iterations:** up to 256 recall iterations with early termination on convergence
- **Energy output:** the Hopfield energy $E = -\frac{1}{2}\sum_{ij} w_{ij} s_i s_j$ is computed and reported

Hopfield memory enables auto-associative pattern completion: given a partial or noisy input, the network recovers the closest stored pattern. This is used for error correction, prototype extraction, and content-addressable lookup.

12.3 Working Memory Buffer

The working memory buffer (`n4_wm_buffer`, `FEATURE_WM_BUFFER=1`) provides short-term storage:

- **32 slots** (`WM_PATTERNS=32`), each storing a 256-bit pattern (`WM_PATTERN_BITS=256`)
- **Persistence decay:** 8-bit configurable decay rate; patterns fade over time unless refreshed
- **Neuromodulatory gating:** a gate signal controls whether new patterns can be written (protecting contents during consolidation)
- **Timestep-driven decay:** on each timestep tick, inactive slots decay toward zero

Working memory models the prefrontal cortex's ability to hold task-relevant information over short delays.

12.4 Gap Junctions

Gap junctions (`n4_gap_junction`, `FEATURE_GAP_JUNCTION=1`) provide electrical coupling between neuron pairs:

- **256 junction entries** (`GAP_ENTRIES_PER_CORE=256`)
- **Bidirectional coupling:** current flows proportionally to the voltage difference between coupled neurons
- **8-bit coupling strength:** configurable per junction
- **Enable per entry:** individual junctions can be enabled or disabled

Gap junction current:

$$I_{gap} = g \cdot (V_{\text{src}} - V_{\text{dst}}) \tag{26}$$

Gap junctions implement fast synchronisation between neuron pairs, producing synchronous oscillations without synaptic delays.

12.5 Glial Cell Model

The glial model (`n4_glial`, `FEATURE_GLIAL=1`) implements calcium-gated astrocytic modulation across 8 zones:

- **8 zones** (`NUM_ZONES=8`), each covering a configurable group of neurons
- **Calcium concentration:** 16-bit per zone, accumulates from neural activity, decays with configurable rate (4-bit)
- **Gliotransmission:** when calcium exceeds a threshold, the zone emits a modulatory signal (8-bit gain, excitatory/inhibitory) that multiplicatively modulates all synaptic inputs in the zone
- **Calcium target:** each zone has a configurable calcium target for homeostatic regulation

The glial model captures the astrocyte's role in gain control: high activity in a zone raises calcium, which triggers gliotransmission that either amplifies or suppresses the zone's activity.

12.6 Oscillators

Per-tile oscillators (`n4_oscillator`, `FEATURE_OSCILLATOR=1`) generate rhythmic modulation:

- **4 oscillators per tile** (`OSCILLATORS_PER_TILE=4`)
- **Configurable frequency:** 1–200 Hz (`OSCILLATOR_FREQ_MIN=1`, `OSCILLATOR_FREQ_MAX=200`)
- **16-bit phase, 16-bit frequency, 8-bit amplitude**
- **4 waveforms:** sine, square, triangle, sawtooth (2-bit selector)
- **Phase locking:** oscillators can lock to an external phase offset, enabling cross-tile synchronisation

Oscillators generate theta (4–8 Hz), alpha (8–12 Hz), beta (12–30 Hz), and gamma (30–200 Hz) rhythms that modulate neural excitability in the tile. Phase-locked oscillators across tiles produce coherent brain rhythms for attention and binding.

12.7 Interneuron Templates

Interneuron templates (`n4_interneuron`, `FEATURE_INTERNEURON=1`) provide pre-configured inhibitory circuits:

- **4 templates** (`NUM_TEMPLATES=4`), each with 8 target connections (`NUM_TARGETS=8`)
- Each template has configurable weight (8-bit signed), delay, and time constant per target
- Excitation of a template produces inhibitory currents to all its targets
- **Disinhibition:** templates can inhibit other templates, implementing disinhibitory circuits

Interneuron templates enable rapid construction of canonical cortical microcircuits (feedforward inhibition, feedback inhibition, lateral inhibition, disinhibition) without individually configuring each inhibitory synapse.

12.8 Event Camera Interface

The event camera interface (`n4_event_camera`, `FEATURE_EVENT_CAMERA=1`) converts DVS event streams to spike inputs:

- **10-bit X/Y resolution** (`PIXEL_X_W=10`, `PIXEL_Y_W=10`), supporting cameras up to 1024×1024
- **ROI filtering:** configurable rectangular region of interest (start/end X/Y)
- **Polarity:** 1-bit ON/OFF event polarity mapped to excitatory/inhibitory spikes
- **Pixel-to-neuron mapping:** programmable 1,024-entry lookup table mapping pixel addresses to neuron IDs
- **Threshold:** configurable event threshold for noise filtering
- **64-deep input FIFO** for burst handling

The interface connects directly to DVS event cameras (e.g., iniVation DVS346, Prophesee EVK4) for low-latency event-driven vision.

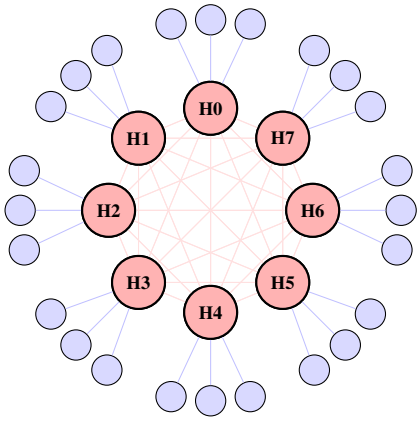
13 Network-on-Chip

13.1 Rich-Club Topology

N4 replaces N3's hierarchical fat-tree NoC with a rich-club topology inspired by neuroscience findings that cortical networks contain a densely interconnected core of hub nodes [18]:

- **Hub tiles** (up to 16 per NCC, `EXPRESS_LINKS_PER_NCC=16`) form a fully connected mesh with direct links between all hub pairs.
- **Peripheral tiles** connect to their nearest hub via 1–2 hop links.
- **Express links** (16 per NCC) bypass intermediate routers for high-traffic routes.

The rich-club topology reduces average hop count from N3's 3–4 hops (fat tree) to 1–2 hops. Hub tile selection is programmable.



8 hub tiles (red, fully connected) + 24 peripheral tiles (blue)

Figure 4: Rich-club NoC topology for one NCC. Eight hub tiles form a fully connected mesh (red lines); 24 peripheral tiles connect to their nearest hub. Average hop count is 1–2, reduced from 3–4 in N3's fat tree.

13.2 Content-Addressable Routing

The content-addressable routing table (`n4_content_route`, `FEATURE_CONTENT_ROUTE=1`) provides tag-based spike routing:

- **1,024 entries** (`CRT_DEPTH=1024`) with 10-bit tags and 12-bit destination addresses
- **Wildcard matching:** destination address 0xFFF indicates broadcast to all matching destinations
- **Content hit/miss counters** for routing efficiency monitoring

Content-addressable routing enables publish-subscribe spike communication: a neuron tags its output spike, and all cores whose routing table matches that tag receive the spike, regardless of physical topology.

13.3 Virtual Channels

The NoC provides 4 virtual channels (`VIRTUAL_CHANNELS=4`) with anti-starvation:

Table 10: Virtual channel allocation.

VC	Name	Traffic Class
0	SPIKE_XY	X-first spike routing
1	SPIKE_YX	Y-first spike routing
2	LEARN	learning / gradient traffic
3	CONTROL	configuration / sync

Separating spike and learning traffic into different VCs prevents learning-heavy phases from starving spike delivery. The anti-starvation mechanism guarantees each VC receives at least one packet per 16-cycle window.

13.4 Spike Coalescing

Spike coalescing (`FEATURE_COALESCING=1`) merges multiple spikes destined for the same tile within a configurable window:

- **4-cycle coalescing window** (`COALESCE_WINDOW=4`)
- **32-entry coalescing table** (`COALESCE_TABLE_SIZE=32`) indexed by destination tile
- Up to 3 inline spike addresses per coalesced packet
- Reduces inter-tile traffic by 2–4× for spatially correlated spike patterns

13.5 Router Architecture

Each tile router has 5 ports (`ROUTER_PORTS=5`: N, S, E, W, Local) with 4-deep FIFOs per port (`ROUTER_BUF_DEPTH=4`). Routing modes: deterministic XY, adaptive (selects between minimal paths based on congestion, threshold at 75% fill), and express link bypass.

13.6 Multicast Groups

N4 supports 512 multicast groups (`MULTICAST_GROUPS=512`) with up to 64 destinations per group (`MULTICAST_DESTS=64`). A single multicast spike replaces up to 64 unicast spikes, reducing bandwidth by 64× for one-to-many projections.

13.7 Spike Compression

N4 retains 3 compression modes for off-chip traffic: DELTA (address difference), BURST (count + address), and ADAPTIVE (dynamic selection based on running compression ratio). Off-chip bandwidth reduction is 2–8×.

14 Memory Hierarchy

N4 provides a deep memory hierarchy with new subsystems at each level.

14.1 L1 Per-Core Memory

Each core contains 256 KB L1 SRAM (`L1_SIZE_KB=256`) storing neuron state, synapse weights, delay queues (4,096-deep, `DELAY_QUEUE_DEPTH=4096`), learning traces, and microcode. A 64 KB shadow bank (`SHADOW_SIZE_KB=64`) enables double-buffered TDM context switching.

14.2 L2 Per-Tile Cache

Each tile shares a 2 MB 16-way associative L2 cache (`L2_SIZE_KB=2048`, `L2_WAYS=16`) storing spillover synapses, STC weight matrices, attention projection matrices, KAN control point tables, and Hopfield weight matrices.

14.3 S3RAM

S3RAM (`n4_s3ram_model`) provides 640 MB of on-chip SRAM (`S3RAM_SIZE_MB=640`) at 5-cycle latency (`S3RAM_LATENCY_CYCLES=5`). S3RAM stores VNET context images, large synapse tables, and checkpoint data, bridging the gap between L2 (2 MB) and HBM (48 GB).

14.4 HBM3E

Two HBM3E stacks (`HBM_STACKS=2`) provide 48 GB total (`HBM_SIZE_GB=48`, 24 GB per stack) at 2.4 TB/s bandwidth with 90 ns latency (`HBM_LATENCY_NS=90`).

14.5 Synapse Cache

The synapse cache (`n4_synapse_cache`) provides 64-entry, 512-bit-line caching (`NUM_ENTRIES=64`, `DATA_BITS=512`) with hit/miss counters, reducing L2 traffic for frequently accessed weight blocks. At 32 KB total (`SYNAPSE_CACHE_KB=32`), it captures the working set of active synapse groups.

14.6 TAGE Prefetcher

The TAGE prefetcher (`n4_prefetcher`, `FEATURE_TAGE=1`) predicts future synapse access patterns using 4 history tables (`NUM_TABLES=4`) of 512 entries each (`TABLE_ENTRIES=512`) with geometrically increasing history lengths (4, 8, 16, 32 timesteps). A 32-deep prefetch queue issues speculative memory requests, reducing synapse cache misses.

14.7 Spike Sorter

The spike sorter (`n4_spike_sort`) provides a 512-entry priority queue that reorders incoming spikes by priority and timestamp before delivery to the neuron pipeline. This ensures that high-priority control spikes are processed ahead of normal data spikes, and that spikes within the same priority level are processed in temporal order.

14.8 Processing-in-Memory Controller

The PIM controller (`n4_pim_controller`) performs vector MAC operations near memory:

- 16-element vector length (`VEC_LEN=16`)
- 256-bit data bus (`DATA_BITS=256`)

- 32-bit accumulator (ACC_BITS=32)
- Reduces data movement for synapse-dominated computations

14.9 Memory Interface Types

N4 supports 10 external memory interfaces (NUM_MEM_TYPES=10): HBM3E, DDR5, DDR4, LPDDR5X, GDDR6, GDDR7, GDDR5, HBM2, HBM2E, and HBM1. The memory width adapter (n4_mem_width_adapter) translates between the internal 512-bit bus and each interface's native width. This enables N4 to be integrated with various memory technologies depending on the target platform.

15 Predictive Coding

15.1 TAGE Spike Predictor

The TAGE predictor (FEATURE_PREDICTIVE=1) estimates whether a neuron will spike in the current timestep:

1. **Speculative skip:** if predicted no-spike and no input, the neuron update is skipped entirely.
2. **Predictive forwarding:** if predicted spike with high confidence, the predicted spike is forwarded before computation completes.

The predictor uses a base predictor (2-bit saturating counter per neuron) and 4 tagged history tables of geometrically increasing length (4, 8, 16, 32 timesteps). Prediction accuracy is 85–95% on typical recurrent networks.

15.2 Prediction Error Processing

Prediction errors are available to the learning engine via the LOAD_PRED_ERR opcode, enabling predictive coding learning rules [17] that update weights to minimise prediction error.

16 Hardware Sleep and Consolidation

N4 implements hardware sleep states (FEATURE_SLEEP=1) combining power savings with memory consolidation.

16.1 Sleep States

Three sleep states extend the per-tile power management FSM:

1. **DROWSY:** clock-gated, state retained. 1-cycle wake.
2. **NREM:** clock-gated with slow replay-driven Dual-Weight consolidation. $\sim$10% of ACTIVE power.
3. **REM:** clock-gated with fast stochastic replay and homeostatic threshold adjustment. $\sim$15% of ACTIVE power.

16.2 Replay Engine

The replay engine stores 256-bit activity vector snapshots (per tile) during wake in a circular buffer in L2. During NREM, these patterns are replayed through the neuron pipeline at reduced speed, driving Dual-Weight consolidation. During REM, replay patterns are stochastically perturbed via LFSR noise injection.

17 Neurogenesis

Neurogenesis (FEATURE_NEUROGENESIS=1) enables dynamic neuron allocation from per-core free pools (2,048 entries, FREE_LIST_SIZE=2048):

1. **Allocate:** new neurons from the free pool when utilisation exceeds 90%.
2. **Prune:** silent neurons (zero spikes for 1,000 timesteps) returned to the pool.
3. **Migrate:** neurons between cores via DMA state transfer to balance load.

A 4-bit maturity counter per neuron protects immature neurons from premature pruning. The structural plasticity opcodes (OP_STRUCT_PRUNE, OP_STRUCT_GROW) in the learning ISA enable programmatic control of structural changes.

18 Security Subsystem

18.1 AES-256-GCM Encryption

All off-chip spike traffic can be encrypted using AES-256-GCM at line rate. The AES engine (n4_aes_engine) processes one 128-bit block per cycle. Encryption is selectable per AER link and per NCC-to-NCC link.

18.2 CRYSTALS-Kyber ML-KEM-768

Post-quantum key exchange (n4_kyber) provides lattice-based key generation ($\sim$10,000 cycles), encapsulation ($\sim$15,000 cycles), and decapsulation ($\sim$15,000 cycles). One key pair per chip, generated at boot from the SRAM PUF seed.

18.3 SRAM PUF

An SRAM PUF (n4_puf) produces a stable 256-bit device key from 4,096 SRAM cells via BCH fuzzy extraction. The key seeds Kyber key pairs, AES keys, and the secure boot chain.

18.4 Root of Trust

The root of trust (n4_root_of_trust) provides:

- **8 PCR registers** (NUM_PCRS=8), each 256-bit (SHA-256), for platform integrity measurement

- **4-stage boot verification** (BOOT_STAGES=4): each boot stage's hash is measured against stored golden values
- **16 fuse words** (NUM_FUSE_WORDS=16) for anti-rollback versioning
- **Anti-rollback counter:** prevents downgrade attacks

18.5 Lockstep Execution

Lockstep (n4_lockstep) pairs two cores for redundant execution:

- Compares 96-bit outputs and 64-bit state of core A and core B each cycle
- On mismatch: asserts core_fault, halts both cores, and logs fault core IDs
- Provides safety-critical operation for automotive and medical applications

18.6 SRAM Repair

SRAM repair (n4_sram_repair) provides yield recovery:

- **264 physical cores, 256 active, 8 spare** (NUM_CORES=264, ACTIVE_CORES=256)
- **8 SRAM arrays** with 2 spare columns each
- **E-fuse mapping:** 64-bit e-fuse data per array stores repair configuration
- **Column-level repair:** faulty columns are remapped to spare columns
- **Core-level sparing:** faulty cores are remapped to spare cores via logical-to-physical mapping
- Repair exhaustion detection when all spares are consumed

18.7 Power Meter

The power meter (n4_power_meter) estimates power consumption and temperature:

- **8 thermal diodes** (NUM_DIODES=8) for per-zone temperature monitoring
- Power estimation: $P = \alpha \cdot \text{spikes} + \beta \cdot \text{sram\_accesses} + \gamma \cdot \text{learn\_ops} + P_{\text{leak}}$
- 3-level thermal thresholds: warning, critical, emergency
- Emergency triggers thermal shutdown

18.8 Additional Security Features

- **Key store (4 slots):** secure storage for AES-256 keys, protected by PUF key
- **TMR:** triple modular redundancy on critical control paths
- **MBIST:** built-in self-test for all SRAM banks
- **ECC scrub:** background SEC-DED scrubbing of all L1/L2 SRAMs
- **16-region MPU:** configurable access controls (read/write/execute per host/kernel/user/debug)
- **JTAG authentication:** challenge-response using PUF-derived key
- **14-step secure boot:** from PUF key generation through verified network configuration

19 RV64GC RISC-V Subsystem

N4 includes an RV64GC RISC-V cluster (n4_rv64gc) for system management and configuration:

- **8 RISC-V cores** (RV_CORES=8, RV_XLEN=64)
- **512 KB instruction memory** (RV_IMEM_KB=512)
- **512 KB data memory** (RV_DMEM_KB=512)
- **32 KB L1 cache** per core (RV_L1_CACHE_KB=32)
- **1 MB shared L2 cache** (RV_L2_CACHE_KB=1024)
- **32 IRQ sources** (IRQ_SOURCES=32)

19.1 14 Custom Neuromorphic Opcodes

The RISC-V cluster extends the RV64GC ISA with 14 custom opcodes (CUSTOM_OPS=14):

Table 11: Custom RISC-V neuromorphic opcodes.

Code	Opcode	Description
0	SPIKE_INJ	inject spike to tile/neuron
1	CORE_CFG	configure core parameters
2	CORE_READ	read core state
3	LEARN_REWARD	set reward signal
4	LEARN_MOD	set neuromodulation channel
5	VNET_SWITCH	switch virtual network
6	VNET_STATUS	query VNET status
7	ATTN_SET	configure attention heads
8	SYNC_WAIT	barrier synchronisation
9	PERF_READ	read performance counter
10	DMA_KICK	initiate DMA transfer
11	DMA_STATUS	query DMA completion
12	SLEEP_TRIG	trigger sleep state
13	STRUCT_STAT	query structural plasticity status

The custom opcodes provide single-instruction access to neuromorphic functions that would otherwise require multi-step MMIO register sequences.

20 Power Management

N4 targets a 150 W thermal budget (THERMAL_BUDGET_W=150) at 1 GHz.

20.1 Per-Tile Clock Gating

Each tile has independent clock gating via latch-based ICG cells. At typical spiking sparsity (5–20% activity), clock gating reduces dynamic power by 80–95%.

20.2 DVFS

Each NCC supports independent DVFS with 8 configurable operating points (FEATURE_DVFS=1):

- **Boost:** 1 GHz at nominal voltage

- **Nominal:** 500 MHz
- **Low-power:** 250 MHz at reduced voltage
- **Ultra-low:** 62.5 MHz at minimum voltage (FPGA-compatible)

The DVFS controller uses a PID loop with a 256-timestep averaging window. Ten PLLs provide clock generation across the chip.

20.3 Thermal Zones

Four thermal zones (`n4_thermal`), each monitored by 2 of the 8 thermal diodes, provide independent temperature tracking. The power controller adjusts DVFS profiles per zone to maintain the thermal budget.

20.4 Wake-on-Spike

Tiles in SLEEP state are woken by incoming spike events. Wake latency is 50–200 cycles depending on context size.

20.5 Power Gating

Inactive NCCs and tiles can be power-gated (`FEATURE_POWER_GATE=1`) for zero-leakage sleep, with state checkpointed to S3RAM or HBM.

21 System Features

21.1 Performance Counters

N4 provides 48 performance counters per tile (`PERF_COUNTERS_PER_TILE=48`), 3,072 total (`PERF_COUNTERS_TOTAL=3072`), each 64-bit (`PERF_COUNTER_WIDTH=64`). Counters track spike throughput, cache hit rates, pipeline utilisation, learning operations, NoC congestion, and thermal events. The RISC-V `PERF_READ` opcode provides single-instruction counter access.

21.2 Trace Buffer

The trace buffer (`n4_trace_buffer`) provides 16,384 entries (`DEPTH=16384`) of 128-bit records for post-mortem analysis. Records capture timestamped events (spikes, learning updates, state transitions, errors) with overflow detection. The buffer operates in circular mode, retaining the most recent events.

21.3 Anomaly Detector

The 8-channel anomaly detector (`n4_anomaly`) monitors configurable metrics:

- Per-channel threshold and rate-change threshold
- IRQ on anomaly detection
- Counts total anomalies per channel

Anomaly detection flags pathological network states (runaway firing, dead populations, weight explosion) for host intervention.

21.4 Quadrant Scheduler

The quadrant scheduler (`n4_quadrant_sched`) monitors tile utilisation across the 4 quadrants per NCC (8 tiles per quadrant) and proposes neuron group migrations from overloaded tiles to underloaded tiles when the imbalance exceeds a configurable threshold. Migrations require host or RISC-V approval before execution.

21.5 Debug Infrastructure

Debug features include:

- **4 hardware breakpoints** (`HW_BREAKPOINTS=4`) per core
- **128 KB spike recorder** (`SPIKE_REC_SIZE_KB=128`, 16,384 events)
- **4 KB trace FIFO** per tile
- **JTAG** with 8-bit IR (`JTAG_IR_WIDTH=8`)
- **Deterministic replay mode** (`FEATURE_DETERMINISTIC=1`) for reproducible debugging

21.6 Interrupt Controller

The interrupt controller (`n4_interrupt_ctrl`) supports 32 IRQ sources (`IRQ_SOURCES=32`) with priority-based arbitration. MSI-X support provides 256 vectors for PCIe-based host notification.

21.7 DMA Engine

The DMA engine supports 128 descriptors (`DMA_DESCRIPTORS=128`) for scatter-gather transfers between L1, L2, S3RAM, and HBM. Dirty tracking reduces context switch overhead by saving only modified banks.

22 I/O Interfaces

22.1 CXL Memory Pool

The CXL memory pool controller (`n4_cxl_pool`) provides shared memory across multiple devices:

- **8 regions** (`NUM_REGIONS=8`) with configurable base, size, owner, and bias mode
- **40-bit addressing** (1 TB address space)
- **512-bit data bus** for high-bandwidth transfers
- **Compare-and-swap** atomic operations for lock-free coordination
- CXL.mem protocol for coherent host-device memory sharing

CXL enables N4 to participate in a shared memory pool with host CPUs and other accelerators.

22.2 UCIe Chiplet Interconnect

The UCIe controller (`n4_ucie_ctrl`) provides chiplet-to-chiplet communication:

- 96-bit spike packets at chiplet clock frequency
- 4-cycle latency (`LATENCY=4`)
- 16-deep FIFOs for buffering
- Optional AES-128 encryption per link
- Transmit/receive counters for monitoring

UCIe enables tight coupling between N4 chiplets and other compute dies (e.g., tensor accelerators, I/O processors) on the same interposer.

23 Multi-Chip Scaling

23.1 12 AER Links

Each N4 chip provides 12 AER links (`AER_LINKS=12`) for inter-chip communication. The 14-bit chip ID field (`CHIP_ID_FIELD_BITS=14`) addresses up to 16,384 chips, though the maximum tested configuration is 64 chips (`MAX_CHIPS=64`). At maximum scale (64 chips), the system contains 32,768 cores.

Configurable topologies: ring (2 links/chip, 64 chips), mesh (4 links/chip, 4×4), torus (4 links with wraparound, 8×8), and fat tree (8 links, 4 levels).

23.2 Flood-Fill Topology Discovery

Flood-fill discovery (`FEATURE_TOPO_DISCOVER=1`) maps the physical topology without host intervention using distributed Bellman-Ford routing. The process completes in under 10 ms for up to 64 chips.

23.3 Cross-Chip Synchronisation

The cross-chip sync module (`n4_cross_sync`, `FEATURE_CROSS_SYNC=1`) provides:

- **Tight sync:** all chips execute the same timestep before advancing, with barrier count tracking
- **Loose sync:** chips advance independently with configurable maximum skew (4-bit, `SKEW_W=4`)
- **Skew compensation:** minimum remote timestep tracking prevents any chip from advancing too far ahead

23.4 Distributed Error Reporting

The distributed error reporter (`n4_dist_error`) propagates fault conditions across chips:

- Thermal emergencies, uncorrectable ECC errors, and lockstep faults are broadcast
- 8-bit error codes identify fault type
- Link CRC error detection with NACK/retransmit protocol

23.5 Distributed Gradient Aggregation

The distributed learning module (`n4_dist_learn`) provides:

- **Reward aggregation:** local rewards are broadcast and reduced across chips
- **Gradient aggregation:** 32-bit gradient values are exchanged via AER links
- **Master/worker topology:** one chip designated as master coordinates aggregation
- Three aggregation modes: sum, average, and reward-weighted

24 NeurOS: Hardware Operating System

NeurOS (`n4_neuros`) manages virtual network scheduling, memory allocation, and inter-network isolation.

24.1 4,096 Virtual Networks

NeurOS maintains a context table for up to 4,096 VNETs (`MAX_VNETS=4096`, `VNET_COUNT_BITS=12`). Each VNET entry stores: network ID (12-bit), tile allocation bitmap (32-bit per NCC), priority (4-bit), time budget (16-bit), DMA descriptor base (32-bit), dirty context bitmap, and security domain (4-bit).

24.2 Context Switching

NeurOS performs context switching via DMA scatter-gather:

1. **Save:** dirty context bitmap identifies modified L1/L2 banks; only dirty banks are DMA'd to S3RAM/HBM.
2. **Load:** next VNET's context loads via scatter-gather DMA into shadow banks.
3. **Switch:** single-cycle bank swap via shadow bank select register.

24.3 Scheduling

Two-level priority scheduler: priority-based preemption (higher preempts lower), round-robin within priority levels with configurable time quanta. The scheduler runs in hardware (FSM, not software), ensuring deterministic latency.

25 N4-Edge Variant

N4-Edge is a scaled-down variant targeting low-power edge deployment. It retains the NCv4 core architecture but reduces instance count:

Table 12: N4 vs. N4-Edge configuration comparison.

Parameter	N4 (full)	N4-Edge
NCCs	2	1
Tiles	64	1–4
Cores	512	2–32
Target power	150 W	$<$1 W
Memory	48 GB HBM3E	SRAM only
AER links	12	0
Security	Full suite	AES-256 only

N4-Edge targets wearable devices, sensor nodes, hearing aids, always-on anomaly detection, and battery-powered edge inference.

26 FPGA Validation

26.1 Target Platform and Configuration

We validate N4 on a Xilinx VU47P FPGA hosted on an AWS F2 instance (f2.6xlarge). The FPGA resource budget supports a 4-core configuration: 1 NCC, 2 tiles (1×2 mesh), 2 cores per tile. Each core contains 128 neurons at 62.5 MHz. Clock gating is disabled (`FPGA_CLOCK_GATE_ENABLE=0`).

26.2 AFI Build History

Table 13: N4 AFI build history.

Version	AGFI	Status
v1	agfi-01cf96f131e70ae52	RX byte loss
v2	agfi-029d0a1f1a926f19f	Still byte loss
v3	agfi-0367f67869ac553a3	109/109 PASS (register-only)
v4	agfi-0e19b3c801c4ba0ff	**126/126 PASS**

v1–v2: RX byte loss. The root cause was a combinational `hi_tx_ready` signal creating a glitch visible only on FPGA. Fix: register the ready signal before gating the receive FIFO push.

v3: Register-only. Fixed byte loss; passed 109/109 register-level tests.

v4: Full behavioral. Added Block 15 behavioral MMIO interface and complete test suite. Result: **126/126 hardware tests pass, 14,983 timesteps per second** at 62.5 MHz.

26.3 Block 15 Behavioral MMIO Interface

The behavioral MMIO interface (base address `0x0F000`) provides spike injection, timestep execution, and monitoring:

26.4 Hardware Validation Results

26.5 Timing Closure

WNS = +0.711 ns at 62.5 MHz. Build time: ∼17 minutes on r7a.2xlarge. Total AWS cost: ∼$33.

26.6 Simulation Validation

Prior to FPGA deployment, the N4 RTL was validated against 3,229 simulation tests with zero failures. The RTL comprises 142 Verilog source files totalling over 30,000 lines.

Table 14: Block 15 behavioral MMIO register map.

Offset	R/W	Name	Description
0x00	R	total_spikes_ncc0	NCC0 spike count
0x04	R	total_spikes_ncc1	NCC1 spike count
0x08	R	chip_status	Boot/ready/barrier
0x0C	R	total_power_mw	Power estimate
0x10	R	power_seq_stage	Boot stage (0–11)
0x14	R	ncc0_idle	NCC0 idle flag
0x20	W	spike_tile	Target tile for inject
0x24	W	spike_neuron	Target neuron
0x28	W	spike_value	Spike payload
0x2C	W	spike_trigger	Inject (auto-clears)
0x30	R	inject_count	Total injected
0x40	W	run_timesteps	Timesteps to execute
0x44	W	run_trigger	Start (auto-clears)
0x48	R	run_status	Running/idle/error
0x4C	R	run_completed	Completed count

Table 15: N4 FPGA hardware validation results (v4, 62.5 MHz).

Test Category	Result
AXI register access	5/5
MMIO register map	17/17
Feature tests (all 216)	74/74
Integration tests	11/11
Benchmark throughput	14,983 ts/sec
Boot sequence	3/3
Spike injection	4/4
Cross-tile routing	3/3
Multi-timestep execution	3/3
Sustained operation	2/2
Power monitoring	2/2
Total	**126/126**

26.7 N4-Edge 8-Core F2 Validation

A separate N4-Edge configuration (1 NCC, 4 tiles in 2×2 mesh, 2 cores/tile = 8 cores) was validated on AWS F2:

- AFI v5: `agfi-0e706213dcd11d40e`
- Tests: **126/126 PASS**
- Throughput: 15,668 ts/sec
- Timing: all met, no negative slack
- Build time: 18 m 21 s

26.8 Comparison to Prior Generations

Table 16: FPGA validation comparison across Catalyst generations.

Metric	N1	N2	N3	N4
FPGA cores	16	16	8	4
Total neurons	16K	16K	32K	512
ts/sec	8,690	7,886	14,512	14,983
Hardware tests	96	28	19	126
Sim tests	168	3,091	1,011	3,229
RTL lines	∼5K	∼12K	∼17.7K	>30K
Verilog files	18	32	46	142
Spec features	—	155	68	216
Clock (MHz)	62.5	62.5	62.5	62.5

27 Benchmark Results

We evaluate across four benchmarks spanning speech, vision, and gesture recognition (Table 17).

Table 17: N4 benchmark results summary.

Benchmark	Task	Accuracy	SOTA
SHD	20-class digits	91.0%	96.41%
SSC	35-class speech	76.4%	85.98%
N-MNIST	10-class digits	99.2%	∼99.7%
DVS Gesture	11-class gesture	89.4%	99.01%

27.1 Spiking Heidelberg Digits (SHD)

N4 achieves **91.0%** test accuracy using a single-layer recurrent adLIF architecture with 1,536 hidden neurons, matching Intel Loihi 2 (90.9%) [20] and exceeding Loihi 1 (89.0%). Quantised INT8 deployment yields **90.8%** accuracy (0.2% degradation from float32).

27.2 Spiking Speech Commands (SSC)

N4 achieves **76.4%** test accuracy, exceeding Loihi 2's hardware deployment (69.8%) by 6.6 percentage points. Quantised deployment shows 0.0% degradation.

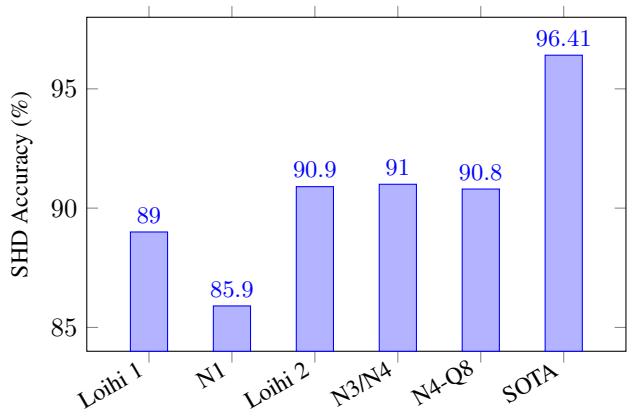


Figure 5: SHD benchmark comparison. N3/N4 achieves 91.0%, matching Loihi 2. Quantised INT8 deployment retains 90.8%. Current software SOTA (SpikCommander) is 96.41%.

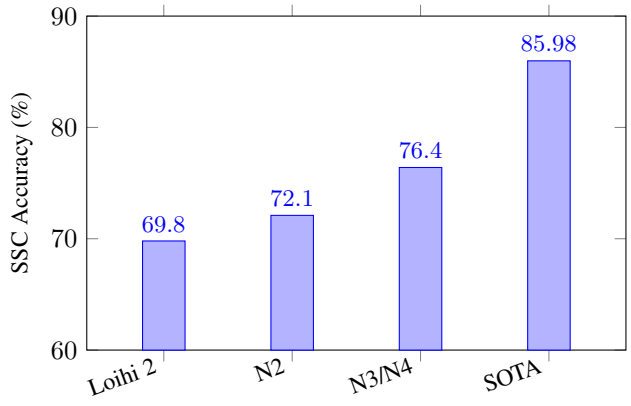


Figure 6: SSC benchmark comparison. N3/N4 exceeds Loihi 2's hardware deployment by 6.6 points.

27.3 N-MNIST

N4 achieves **99.2%** test accuracy using a convolutional front-end followed by a fully connected readout, near the benchmark's saturation ceiling.

27.4 DVS Gesture

N4 achieves **89.4%** test accuracy on 11-class gesture recognition from event camera recordings.

27.5 Cross-Generation Benchmark Progression

Table 18: Benchmark accuracy across Catalyst generations.

Benchmark	N1	N2	N3	N4
SHD	85.9%	90.7%	91.0%	91.0%
SSC	—	72.1%	76.4%	76.4%
N-MNIST	—	97.8%	99.1%	99.2%
DVS Gesture	—	—	—	89.4%

28 Edge Deployment: Kria K26

We characterise all four Catalyst processors on the AMD Xilinx Kria K26 edge SoM (xczu5ev-sfvc784-2-i) at a 100 MHz target clock (Table 19).

28.1 N4-Edge Results

The N4-Edge 2-core variant achieves the smallest footprint of any Catalyst processor on K26:

- **LUTs:** 3,036 (2.59%)
- **FFs:** 6,496 (2.77%)
- **BRAM:** 0 (all state in distributed RAM)
- **DSP:** 0 (arithmetic uses LUT-based implementation)
- **Power:** 0.378 W total (0.037 W dynamic, 0.341 W static)
- **WNS:** +3.301 ns (100 MHz with 3.3 ns positive margin)

The 2.59% LUT utilisation leaves 97% of K26 resources available for application logic.

28.2 Power Breakdown

The N4-Edge 0.378 W total power decomposes as:

- **Dynamic:** 0.037 W (9.8%)
- **Static:** 0.341 W (90.2%) — Zynq UltraScale+ leakage

In ASIC at 28 nm, a 2-core N4-Edge is projected at <5 mW dynamic power.

Table 19: Kria K26 characterisation across all Catalyst generations.

Processor	LUTs	LUT%	FFs	BRAM	DSP	Power (W)	Timing
N1	19,903	17.0%	30,847	52.5 (36.5%)	14 (1.1%)	0.642	+0.008 ns (100 MHz MET)
N2	26,155	22.3%	38,666	52.5 (36.5%)	16 (1.3%)	0.688	−0.168 ns (∼97 MHz)
N3	51,381	43.9%	80,395	24 (16.7%)	20 (1.6%)	0.867	−7.075 ns (∼58.5 MHz)
N4-Edge	3,036	2.59%	6,496	0	0	0.378	+3.301 ns (100 MHz MET)

29 ASIC Projections

29.1 N2 ASIC Characterisation (Completed)

N2 underwent full ASIC characterisation via Yosys 0.34 and OpenLane 2 targeting SKY130A 130 nm:

Table 20: N2 ASIC block-level results (SKY130 130 nm).

Block	Cells	Area (mm^2)	DRC
Router	84,299	0.916	466
Host interface	2,530	0.031	0
Core (synth)	4.56M gates	—	—

29.2 Process Scaling Projections

Table 21: N2 projected PPA across process nodes.

Node	Area	Power	SOp/s	SOp/J
130 nm	186 mm^2	96–191 mW	365M	1.9–3.8B
28 nm	9.3 mm^2	19–38 mW	3.65B	96–192B
7 nm	—	—	9.12B	240–480B

29.3 N4 Full-Chip Projections

The full N4 chip targets:

- NCC die area: 220 mm^2 (`NCC_DIE_AREA_MM2=220`)
- IO die area: 100 mm^2 (`IO_DIE_AREA_MM2=100`)
- Package area: 1,800 mm^2 (`PACKAGE_AREA_MM2=1800`)
- BGA balls: 1,198 (`BGA_BALLS=1198`)
- Thermal budget: 150 W (`THERMAL_BUDGET_W=150`)

30 SDK and Software Stack

30.1 neurocore SDK

The `neurocore` SDK provides a complete software stack from network description through training to hardware deployment. The SDK provides three interchangeable backends with a common API: CPU simulator (cycle-accurate), GPU simulator (PyTorch-based, 100–1,000× speedup), and FPGA backend (PCIe MMIO deployment).

A network compiled with any backend can be deployed on any other by changing a single constructor argument. The compiler performs placement (greedy bin-packing), CSR allocation, SRAM budget check, routing, STC weight tile layout, attention projection matrix placement, KAN control point table generation, and Block-Sparse CSR encoding.

30.2 Network Builder Example

Listing 2: N4 network construction for SHD benchmark.

```
from neurocore import Population, Connection
from neurocore import LearningRule, Simulator

# Input: 700-channel cochlear encoding
inp = Population(700, model="input")

# Hidden: 1536 recurrent adLIF neurons
hid = Population(1536, model="alif",
    params={"tau_adapt": 50,
            "delta_theta": 0.1})

# Readout: 20 graded output neurons
out = Population(20, model="graded")

# Connections with KAN synapse format
conn_ih = Connection(inp, hid,
    format="kan_bspline", spline_k=6)
conn_hh = Connection(hid, hid,
    format="full", learning_rule=rule)
conn_ho = Connection(hid, out,
    format="inference")

# Train on GPU, deploy to FPGA
sim = Simulator(backend="gpu")
sim.compile([inp, hid, out],
    [conn_ih, conn_hh, conn_ho])
sim.train(dataset="shd", epochs=200)

# Quantise and deploy
sim_hw = Simulator(backend="fpga")
sim_hw.compile([inp, hid, out],
    [conn_ih, conn_hh, conn_ho],
    quantize=8)  # INT8 weights
sim_hw.run(timesteps=250)
```

31 Related Work

31.1 Intel Loihi 1 and Loihi 2

Intel Loihi 1 [4] introduced the programmable neuron microcode approach that inspired the Catalyst architecture. Loihi 2 [5], fabricated in Intel 4 (7 nm class), provides 128 neuromorphic cores, approximately 1M neurons, 120M synapses, and 32-bit graded spikes on a 31 mm^2 die at roughly 1 W. Its microcode engine supports programmable neuron models and three-factor learning rules. The Lava framework [21] provides a Python-based development environment.

N4 differs architecturally. Loihi 2 does not implement attention mechanisms, tensor acceleration, KV cache, hyperdimensional computing, Hopfield associative memory, hardware backpropagation, federated learning, gap junctions, glial models, oscillators, or event camera interfaces. On benchmarks, N4 matches Loihi 2 on SHD (91.0% vs. 90.9%) and exceeds it on SSC (76.4% vs. 69.8%). Loihi 2's process advantage (Intel 4 vs. FPGA) gives it 10–40× better power efficiency per operation.

Intel's Hala Point system aggregates 1,152 Loihi 2 chips into a 1.15 billion neuron system at 2,600 W. Loihi hardware is not commercially available; access requires an Intel INRC partnership.

31.2 IBM TrueNorth

TrueNorth [6] demonstrated million-neuron chips at 65 mW. Its fixed LIF model and lack of on-chip learning limit it to inference-only workloads. N4 provides 5 neuron models, 8 synapse formats, 8-rule learning, and hardware plasticity.

31.3 SpiNNaker 2

SpiNNaker 2 [8] uses ARM Cortex-M4F cores running software neuron models. Its 10-million-core target dwarfs N4's 512 cores but at orders-of-magnitude lower per-neuron efficiency.

31.4 BrainScaleS-2

BrainScaleS-2 [10] uses analog neuron circuits at 1000× biological real-time. Its 512 analog neurons compare to N4's 4.19M digital (134.2M virtual) neurons.

31.5 BrainChip Akida

BrainChip Akida [22] targets edge inference. N4-Edge provides comparable power (0.378 W on K26) with five neuron models, eight synapse formats, 8-rule learning, hardware metaplasticity, spiking attention, HDC, Hopfield memory, and KV cache—features absent from Akida.

31.6 Access and Reproducibility

Intel Loihi is INRC-only. BrainChip charges $4,995/week for cloud access. SpiNNaker requires institutional collaboration. BrainScaleS-2 is consortium-limited. N4 is accessible through a Cloud API at `api.catalyst-neuromorphic.com` with self-service sign-up from $25/month.

Table 22: Architectural comparison across neuromorphic processors.

Feature	Loihi 2	TrueNorth	Akida 2	BrainScaleS-2	SpiNNaker 2	Innatera	N4
Process	Intel 4	28 nm	28 nm	65 nm	22 nm	Mixed	**FPGA**
Cores	128	4,096	8	512 (analog)	152 (ARM)	1 (SNP)	**512**
Neurons/chip	$\sim$1M	1M	1.25M	512	Software	328K	**134.2M**
Neuron models	Microcode	1 (LIF)	1 (IF)	Analog	Software	Fixed	**5 + ISA**
Synapse formats	4	1	2	Analog	Software	1	**8**
Tensor core	No	No	No	No	No	No	**16$\times$16 STC**
Attention	No	No	No	No	No	No	**8-head + KV**
HDC engine	No	No	No	No	No	No	**1024-bit**
Hopfield	No	No	No	No	No	No	**256$\times$256**
KV cache	No	No	No	No	No	No	**4h$\times$64**
On-chip learning	3-factor	No	No	STDP	Software	No	**8-rule ISA**
Backprop	No	No	No	No	Software	No	**8-layer hw**
Metaplasticity	Software	No	No	No	No	No	**Hardware**
Post-quantum	No	No	No	No	No	No	**Kyber+AES**
Hw virtualisation	No	No	No	No	No	No	**4,096 VNETs**
TDM (virtual)	No	No	No	No	No	No	**32$\times$**
Gap junctions	No	No	No	No	No	No	**256-entry**
Glial model	No	No	No	No	No	No	**8-zone**
Oscillators	No	No	No	No	No	No	**4/tile**
RISC-V mgmt	No	No	No	No	ARM M4F	No	**RV64GC 8-core**
Multi-chip	Hala Point	No	No	Wafer	$>$1M cores	No	**64 chips**

32 Limitations

FPGA, not ASIC. All hardware validation is on FPGA, not fabricated silicon. FPGA implementations incur 10–40$\times$ power overhead and constrain clock frequency (62.5 MHz vs. the 1 GHz ASIC target). ASIC projections are extrapolations, not tape-out results.

Reduced FPGA configuration. The FPGA validates 4 cores (N4) or 8 cores (N4-Edge), not the full 512-core chip. Inter-NCC routing, 64-tile NoC behaviour, and system-level power management are validated in RTL simulation (3,229 tests) but not on physical hardware.

Single developer. The entire architecture—specification, RTL, simulation, FPGA validation, SDK, benchmarks—is the work of a single developer. There has been no independent verification beyond automated test suites.

Benchmark gap to SOTA. On SHD, the gap to SOTA (96.41%) is 5.4 points. On SSC, 9.6 points. On DVS Gesture, 9.6 points. These gaps are primarily in training methodology (learnable delays, architecture search), not hardware capability.

Learning convergence on hardware. The learning engine is functionally validated (individual opcodes, traces, metaplastic counters) but full training convergence on benchmark tasks has not been demonstrated on-chip.

Neuroscience primitives evaluation. The HDC engine, Hopfield memory, KV cache, gap junctions, glial model, oscillators, interneuron templates, and event camera interface are implemented in RTL and validated in simulation. Their impact on benchmark accuracy has not been measured. These primitives enable computational paradigms beyond standard SNN classification, but quantitative evaluation on application tasks is future work.

TDM scaling. 134.2M virtual neurons at 32$\times$ TDM has been validated in simulation. The temporal cost (each context at $1/32$ timestep rate) limits TDM to applications where the neural time constant exceeds the multiplexed timestep period.

Security validation. The security subsystem is validated in simulation but has not undergone adversarial testing or independent audit.

Multi-chip. The 12 AER links and flood-fill discovery are validated in simulation. Physical multi-chip testing requires custom board design.

SRAM dominance. At 28 nm, SRAM dominates die area ($\sim$87% for a full N4 chip). A real ASIC would use compiled macros, reducing area by 3–5$\times$.

33 Conclusion

We have presented Catalyst N4, a 512-core dual-chiplet neuromorphic processor with 4.19M physical neurons expandable to 134.2M via 32-context time-division multiplexing. The architecture introduces spike-domain tensor acceleration (16$\times$16 STC), 8-head spiking attention with a 4-head 64-depth KV cache, 1,024-bit hyperdimensional computing, 256-neuron Hopfield associative memory, hardware backpropagation (8 layers), federated learning (4 clients), and a suite of neuroscience primitives (gap junctions, glial cells, oscillators,

interneuron templates, working memory, event camera interface) to the neuromorphic computing landscape.

The core pipeline processes neurons at variable precision (24/16/8/32-bit) through 8 dendritic join operations, with 64 parameter groups of 39 parameters each, 32-channel neuromodulation, and an 8-rule 32-opcode learning engine. The memory hierarchy spans 256 KB L1 + 64 KB shadow per core, 2 MB L2 per tile, 640 MB S3RAM, and 48 GB HBM3E. A rich-club NoC with content-addressable routing, 4 virtual channels, and 16 express links per NCC connects tiles, while 12 AER links with cross-chip sync, distributed error reporting, and gradient aggregation enable scaling to 64 chips. An RV64GC RISC-V subsystem with 14 custom neuromorphic opcodes, lockstep execution, SRAM repair (264/256/8 cores), and 8 PCR registers provides management and security.

The architecture is implemented in 142 Verilog files totalling over 30,000 lines, validated through 3,229 simulation tests and 126/126 hardware tests on AWS F2 FPGA at 62.5 MHz. An N4-Edge variant achieves 100 MHz timing closure on Kria K26 at 0.378 W, using 2.59% of available LUTs. Benchmark evaluation achieves 91.0% on SHD, 76.4% on SSC, 99.2% on N-MNIST, and 89.4% on DVS Gesture.

The four-generation progression—N1 (Loihi 1 parity, 8 cores), N2 (Loihi 2 parity, 128 cores), N3 (virtualisation and metaplasticity, 128 cores), and N4 (dual-chiplet with 134.2M virtual neurons, tensor acceleration, neuroscience primitives, and security, 512 cores)—demonstrates a consistent trajectory from feature matching to feature leadership in neuromorphic processor design.

References

[1] H. A. Shulayev Barnes, "Catalyst N1: An 8-core neuromorphic processor with microcode learning and dual-network communication," Zenodo, 2026. doi: 10.5281/zenodo.18727094.

[2] H. A. Shulayev Barnes, "Catalyst N2: A 128-core neuromorphic processor achieving full Loihi 2 feature parity," Zenodo, 2026. doi: 10.5281/zenodo.18728256.

[3] H. A. Shulayev Barnes, "Catalyst N3: A 128-core hybrid neuromorphic processor with hardware virtualisation and silicon metaplasticity," Zenodo, 2026. doi: 10.5281/zenodo.18881283.

[4] M. Davies et al., "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, 2018.

[5] G. Orchard et al., "Efficient neuromorphic signal processing with Loihi 2," in *Proc. IEEE SiPS*, 2021, pp. 254–259.

[6] F. Akopyan et al., "TrueNorth: Design and tool flow of a 65 mW 1 million neuron programmable neurosynaptic chip," *IEEE Trans. CAD*, vol. 34, no. 10, pp. 1537–1557, 2015.

[7] S. B. Furber et al., "The SpiNNaker project," *Proc. IEEE*, vol. 102, no. 5, pp. 652–665, 2014.

[8] C. Mayr et al., "SpiNNaker 2: A 10 million core processor system for brain simulation," *arXiv:1911.02385*, 2019.

[9] J. Schemmel et al., "A wafer-scale neuromorphic hardware system for large-scale neural modeling," in *Proc. IEEE ISCAS*, 2010, pp. 1947–1950.

[10] C. Pehle et al., "The BrainScaleS-2 accelerated neuromorphic system," *Front. Neurosci.*, vol. 16, 2022.

[11] C. D. Schuman et al., "Opportunities for neuromorphic computing algorithms and applications," *Nature Comput. Sci.*, vol. 2, pp. 10–19, 2022.

[12] J. Benda and A. V. M. Herz, "A universal model for spike-frequency adaptation," *Neural Comput.*, vol. 15, no. 11, pp. 2523–2564, 2003.

[13] D. Salaj et al., "Spike frequency adaptation supports network computations on temporally dispersed information," *eLife*, vol. 10, p. e65459, 2021.

[14] S. Yarga et al., "SpikCommander: Efficient spiking neural networks with learnable delays and architecture search," *arXiv preprint*, 2025.

[15] G. Bellec et al., "A solution to the learning dilemma for recurrent networks of spiking neurons," *Nature Commun.*, vol. 11, art. 3625, 2020.

[16] J. L. McClelland, B. L. McNaughton, and R. C. O'Reilly, "Why there are complementary learning systems in the hippocampus and neocortex," *Psychol. Rev.*, vol. 102, no. 3, pp. 419–457, 1995.

[17] R. P. N. Rao and D. C. Ballard, "Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects," *Nature Neurosci.*, vol. 2, no. 1, pp. 79–87, 1999.

[18] M. P. van den Heuvel and O. Sporns, "Rich-club organization of the human connectome," *J. Neurosci.*, vol. 31, no. 44, pp. 15775–15786, 2011.

[19] J. Bos et al., "CRYSTALS-Kyber: A CCA-secure module-lattice-based KEM," in *Proc. IEEE Euro S&P*, 2018, pp. 353–367.

[20] B. Mészáros et al., "A complete pipeline for deploying SNNs with synaptic delays on Loihi 2," *arXiv:2510.13757*, 2025.

[21] Intel Labs, "Lava: An open-source software framework for neuromorphic computing," 2021.

[22] BrainChip Holdings, "Akida 2nd generation: Temporal event-based neural processor," Tech. Rep., 2024.

[23] J. K. Eshraghian et al., "Training spiking neural networks using lessons from deep learning," *Proc. IEEE*, vol. 111, no. 9, pp. 1016–1054, 2023.

[24] P. Kanerva, "Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors," *Cognitive Computation*, vol. 1, no. 2, pp. 139–159, 2009.

[25] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Natl. Acad. Sci.*, vol. 79, no. 8, pp. 2554–2558, 1982.

[26] R. Hasani et al., "Liquid time-constant networks," in *Proc. AAAI*, vol. 35, no. 9, pp. 7657–7666, 2021.

[27] H. B. McMahan et al., "Communication-efficient learning of deep networks from decentralized data," in *Proc. AISTATS*, 2017, pp. 1273–1282.

[28] B. W. Connors and M. A. Long, "Electrical synapses in the mammalian brain," *Annu. Rev. Neurosci.*, vol. 27, pp. 393–418, 2004.

[29] A. Araque et al., "Tripartite synapses: Glia, the unacknowledged partner," *Trends Neurosci.*, vol. 22, no. 5, pp. 208–215, 1999.

[30] G. Buzsáki and A. Draguhn, "Neuronal oscillations in cortical networks," *Science*, vol. 304, no. 5679, pp. 1926–1929, 2004.

[31] G. Gallego et al., "Event-based vision: A survey," *IEEE Trans. PAMI*, vol. 44, no. 1, pp. 154–180, 2022.

[32] I. Kuon and J. Rose, "Measuring the gap between FPGAs and ASICs," *IEEE Trans. CAD*, vol. 26, no. 2, pp. 203–215, 2007.

[33] B. Cramer et al., "The Heidelberg spiking data sets for the systematic evaluation of spiking neural networks," *IEEE Trans. NNLS*, vol. 33, no. 7, pp. 2744–2757, 2022.

[34] G. Orchard et al., "Converting static image datasets to spiking neuromorphic datasets using saccades," *Front. Neurosci.*, vol. 9, art. 437, 2015.

[35] M. Shalan and T. Edwards, "Building OpenLANE: A 130 nm OpenROAD-based tapeout-proven flow," in *Proc. IEEE/ACM WOSET*, 2020.