

Neuromorphic Computing

Study Guide

by Richard J. Young, Ph.D.

richard@deepneuro.ai

Chapter 21 of "The NeuroAI Handbook"

Building AI in the brain's image: in-memory computing, spikes, and event-based sensing.

One-sentence thesis. The brain out-computes any AI on 20 watts because it fuses memory with processing and fires only when something changes, and neuromorphic hardware wins by copying those two tricks.

PART 1: AI's Energy Problem

An Unsustainable Appetite

Modern AI has a dirty secret: it is spectacularly wasteful. Training one large model can burn the electricity of a small town, because we run intelligence on hardware, the GPU, that was built for graphics. The brain is the counterexample. It performs feats no AI can match while sipping just **20 watts**, less than a dim bulb. Scaling AI by building ever-larger data centers is a dead end; the fix is to rethink the computer itself.

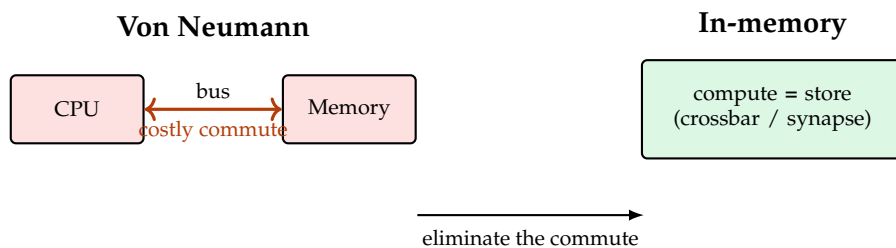
The Von Neumann Bottleneck

The waste has a name. In almost every computer since the 1940s, memory and processor sit in separate places, so computing anything means shuttling data back and forth across a bus. The shuttling, not the

arithmetic, dominates the energy bill. **The core idea.** the brain has no such divide: memory (the synaptic weight) and computation (integrating signals) happen at the very same spot, the synapse. Neuromorphic engineering is the art of building that no-commute architecture in silicon.

ANALOGY

Picture a kitchen with the pantry in the basement and the stove upstairs. For every ingredient the chef sprints down and back. Nearly all the effort is stairs, not cooking. That is the von Neumann bottleneck. The brain is a kitchen where every ingredient sits within arm's reach of the pan.



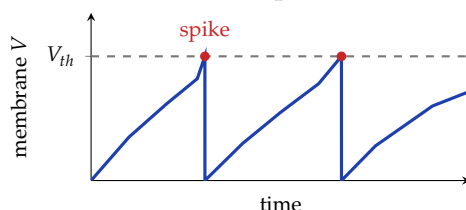
The whole game: stop moving data to the computation and instead compute where the data already lives.

Lesson 1: Compute Where You Store

Put the pantry next to the stove. Neuromorphic hardware does this with a **memristor**, a component whose resistance changes with the voltage you apply, so it physically “remembers” a weight (Strukov et al., 2008). Wire memristors into a grid, a **crossbar array**, and the network’s weight matrix lives in the hardware itself.

Now the magic: drive the rows with input voltages and, by Ohm’s law, the current summed down each column is a matrix-vector multiply, the core operation of deep learning, done in one analog step with almost no energy.

$$I_j = \sum_i G_{ij} V_i$$



The LIF neuron: charge, leak, fire at threshold, reset. Between spikes it costs nothing.

Lesson 3: Sense Changes, Not Snapshots

A normal camera grabs a full frame of millions of pixels 30 times a second even when nothing moves, most of it redundant. The retina does the opposite: its cells fire on *changes* in light. A **dynamic vision sensor (DVS)**, or event camera, copies the retina, each pixel reports asynchronously, and only when its brightness changes past a threshold.

Formally, a pixel emits an event (x, y, t, p) the instant the log-intensity jumps past δ :

$$\log I(t) - \log I(t - \Delta t) > \delta$$

A still scene produces no data at all; a fast object produces a sparse stream that tracks it with **microsecond** timing and a huge (120+ dB) dynamic range. **The payoff.** orders of magnitude less redundant data, which is exactly what a drone or an implant on a tiny power budget needs.

Lesson 2: Talk in Events, Not Numbers

A standard network is a chatterbox: every neuron sends a floating-point number to every neighbor at every step, a dense, constant stream. Real neurons stay quiet and speak only when they have news, firing a **spike**. A **spiking neural network (SNN)** copies this. Information rides in the *timing* of spikes, which buys two things: **sparsity** (most units are silent, so there is little to process) and **efficiency** (no spike means no computation and no energy spent).

The workhorse unit is the **leaky integrate-and-fire (LIF)** neuron: input charges a membrane voltage that leaks away over time; cross a threshold and it fires a spike, then resets.

ANALOGY

A frame camera is a guard who photographs an empty hallway every second all night. An event camera is a motion sensor: silent until something actually moves, then reporting precisely what changed and when.

Chips That Think in Spikes

This is shipping, not theory. Intel's **Loihi 2** (2021) packs a million spiking neurons and 120 million synapses under 100 mW, with learning rules on the chip itself. IBM's **TrueNorth** (2014) runs a million neurons across 4096 cores on 70 mW, less than a hearing-aid battery. Manchester's **SpiNNaker** wires a million ARM cores to simulate a billion neurons in biological real time, and **BrainChip's Akida** brings event-based, on-device learning to commercial edge products.

Where Neuromorphic Wins, and Where It Doesn't

These chips are not GPU replacements for training. They dominate at the **edge**: real-time, low-power sensing on drones, robots, wearables, medical implants, and always-on industrial monitors, where a conventional system would drain the battery or miss a fast event.

Read the caveat. the huge wins hold only for **sparse, event-driven** workloads. For dense matrix math, like training a large transformer, GPUs still win. Neuromorphic is a scalpel, not a bigger hammer.

Real-time object detection, 1000 frames (chapter benchmark)

Platform	Energy (J)	Latency (ms)	Energy-delay product
GPU (V100)	5000	50	250,000
CPU (Xeon)	8000	100	800,000
Loihi 2	5	20	100
TrueNorth	2	10	20

The Training Problem

Here is the sting: neuromorphic chips are wonderful at running a model but awkward at *learning* one. A spike is discontinuous, so it has no clean derivative, and standard backpropagation stalls. Three workarounds dominate. **ANN-to-SNN conversion** trains an ordinary network, then converts activations to spike rates (proven, but loses temporal precision). **Surrogate gradients** swap a smooth stand-in for the spike during the backward pass, enabling direct training that now reaches near-ANN accuracy. And **STDP** learns locally, no backprop at all.

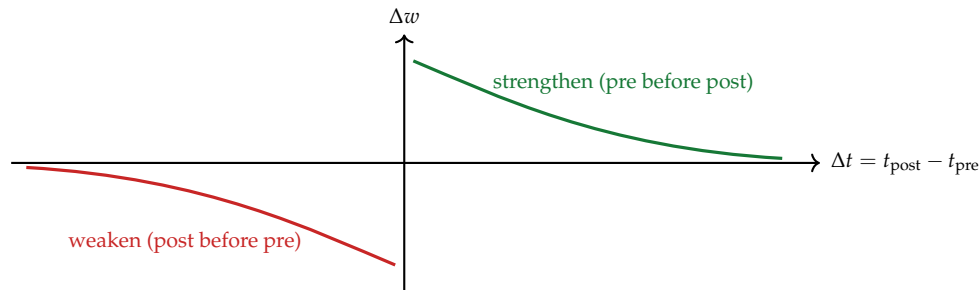
STDP, spike-timing-dependent plasticity, is the brain's own rule: strengthen a synapse when the input spike arrives just *before* the output spike, weaken it when

the order flips.

$$\Delta w = A_+ e^{-\Delta t / \tau_+} \text{ (pre} \rightarrow \text{post)}$$

The Hybrid Future

The likely endgame is not one paradigm winning but a division of labor. Train massive models where power is plentiful, on GPUs in the data center, then deploy the trained intelligence onto hyper-efficient neuromorphic chips at the edge, from phones to retinal implants. The **holy grail** beyond that is on-chip learning: a device that keeps adapting to its user and environment, no cloud round-trip, which also means better privacy and autonomy.



The STDP learning window: timing, not just co-activity, decides whether a synapse grows or shrinks, a fully local rule that needs no backpropagation.

Quick Self-Test

1. What is the von Neumann bottleneck, and which single feature of brain architecture avoids it?
2. How does a memristor crossbar turn a matrix-vector multiply into one analog step?
3. Name the two sources of an SNN's efficiency, and tie each to the LIF neuron's behavior.
4. Why does an event camera produce no data from a static scene, and why is that a feature?
5. The EDP table shows neuromorphic winning by 1000x. State the caveat that keeps GPUs relevant.

If You Remember One Thing

Neuromorphic computing is two ideas, not a hundred: put memory and computation in the same place, and compute only when something changes. Those two moves are why the brain runs on 20 watts, and copying them is why a Loihi or TrueNorth chip can match a GPU's edge task on a thousandth of the energy, for sparse, event-driven work.