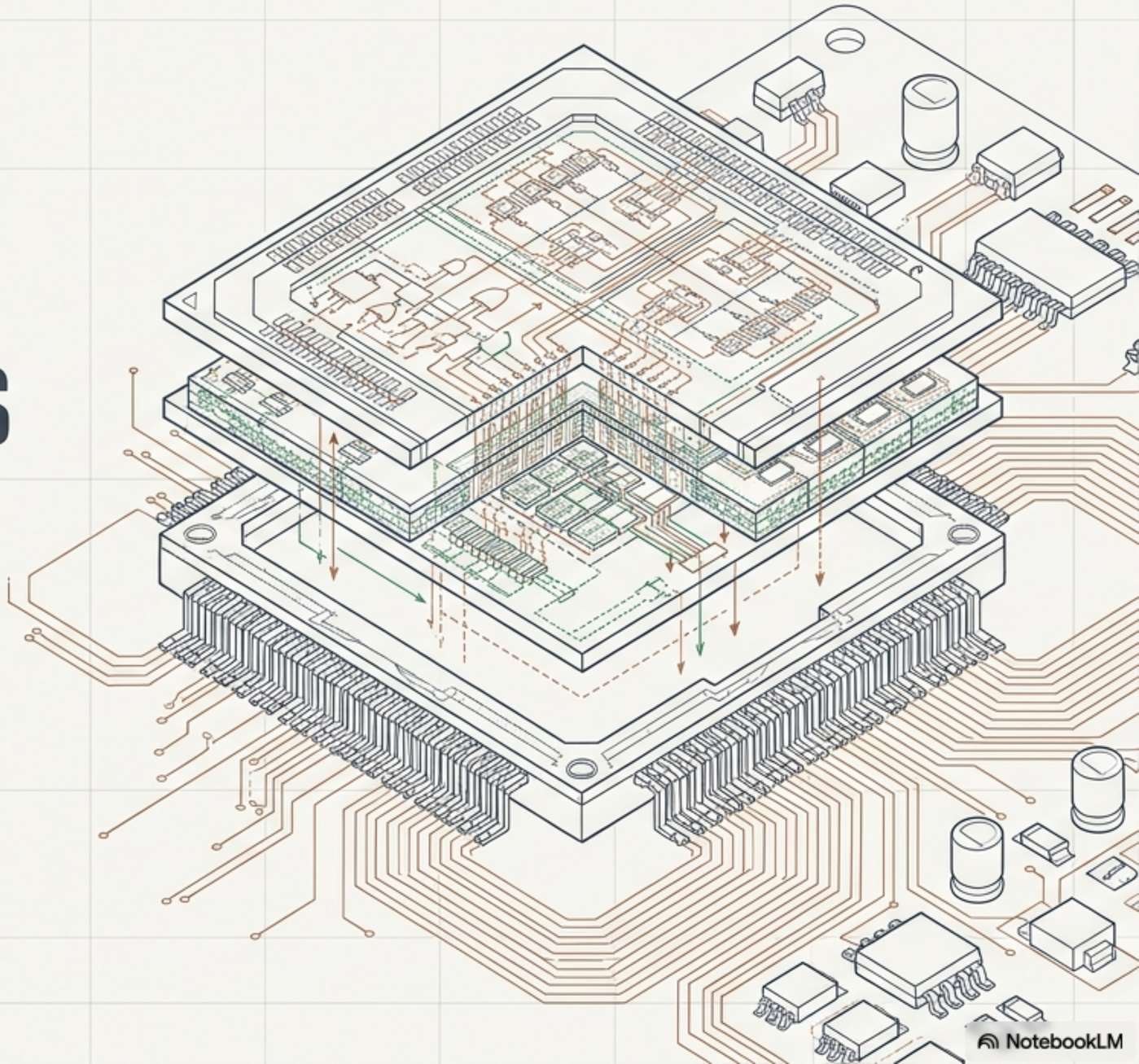
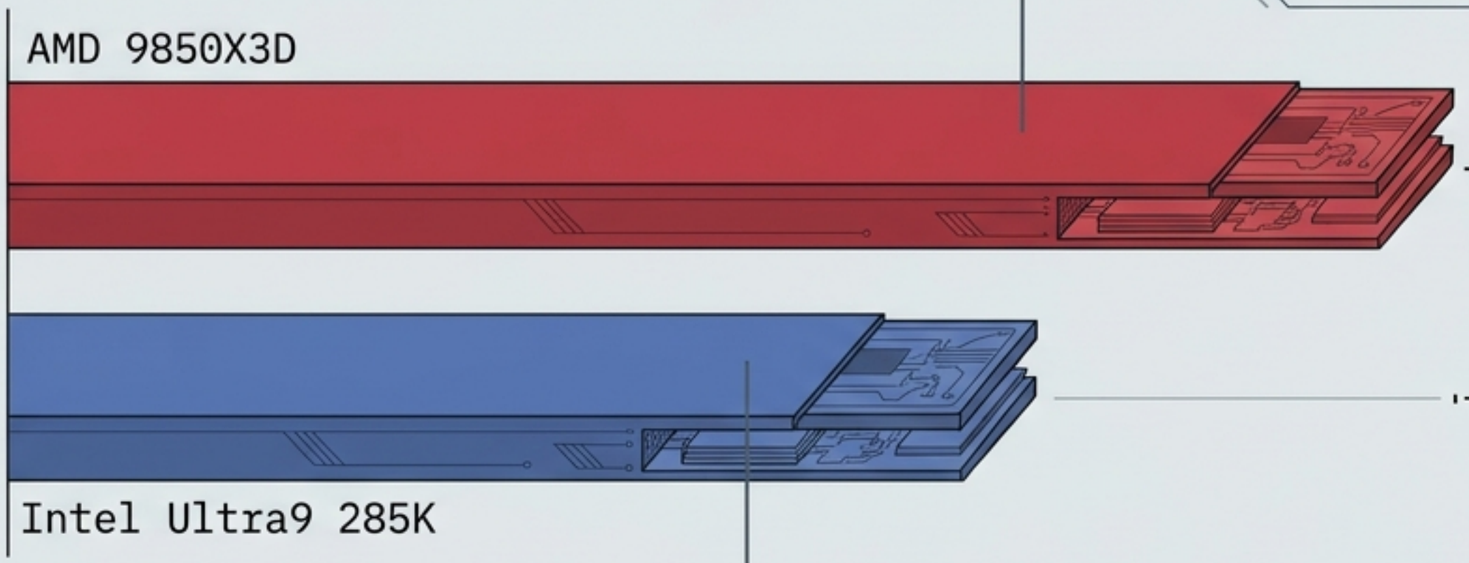
THE PHYSICS OF FRAMERATES

Tracing CPU performance from the motherboard down to the logic gate.



The 27% gaming anomaly

The secret isn't a wild clock speed. It is a **massive physical stockpile of Cache—up to 128MB** stuffed directly into the L3 layer.

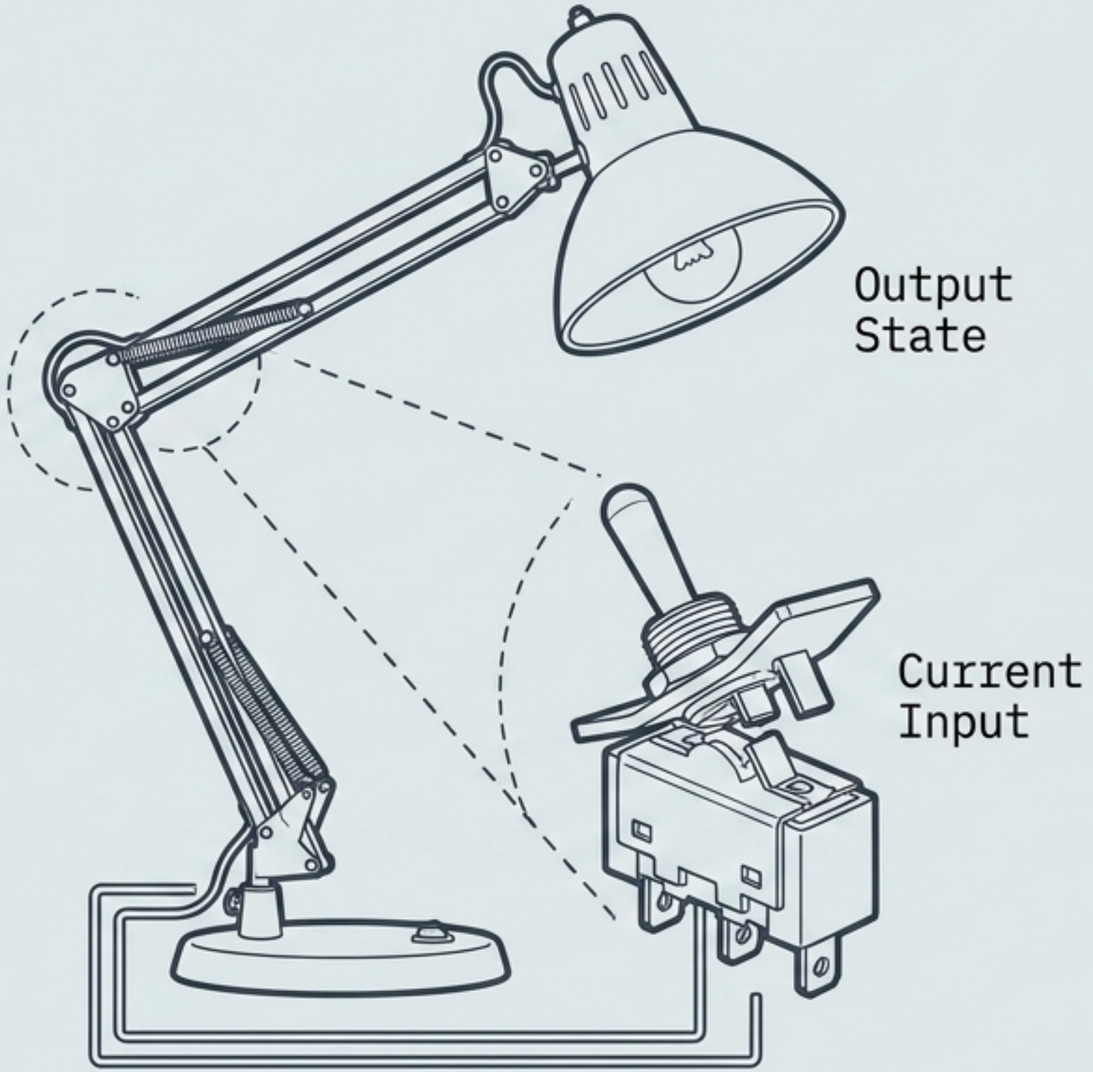


But to understand why cache equals speed, we must first answer a basic physics question:

How does a circuit remember?

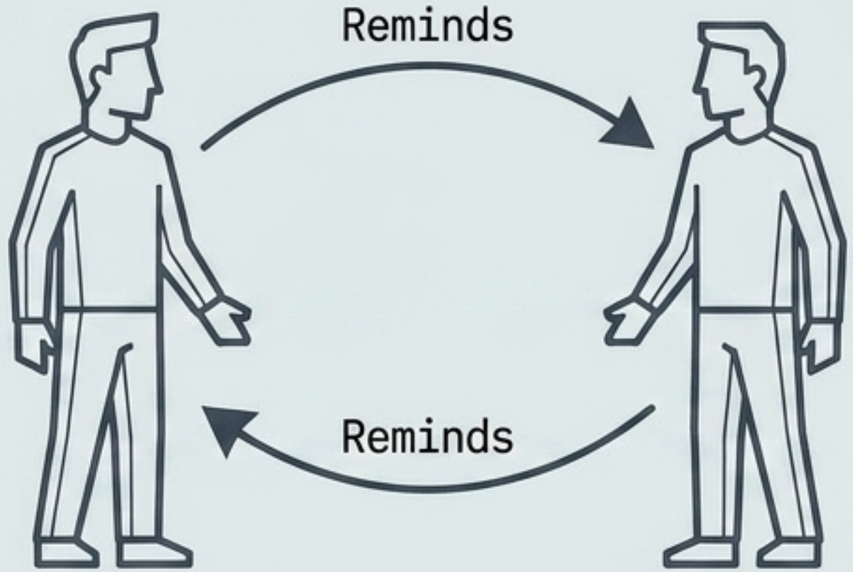
A normal circuit is a goldfish

- When you flip the switch, the lamp lights. Flip it back, it goes dark.
- Its state depends exclusively on the switch right now. It has zero memory of what happened a second ago.
- This is a Combinational Circuit.
- Computers cannot work this way. To process keystrokes and windows, a circuit must hold onto the past.



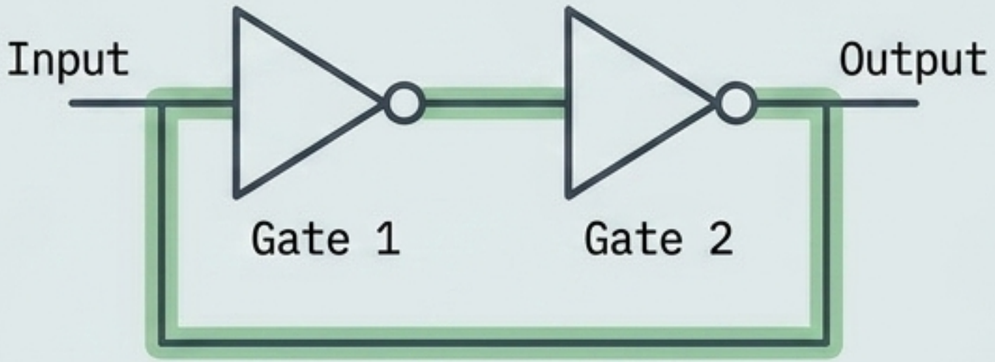
Trapping a signal in a loop

THE ANALOGY



Two people, terrible at remembering, keep a fact alive by constantly reminding each other.

THE CIRCUIT



By looping two NOT gates, a signal flips from high to low and back to high, trapped forever.

This is a Bistable Circuit. It holds one of two states: your 1 or your 0.

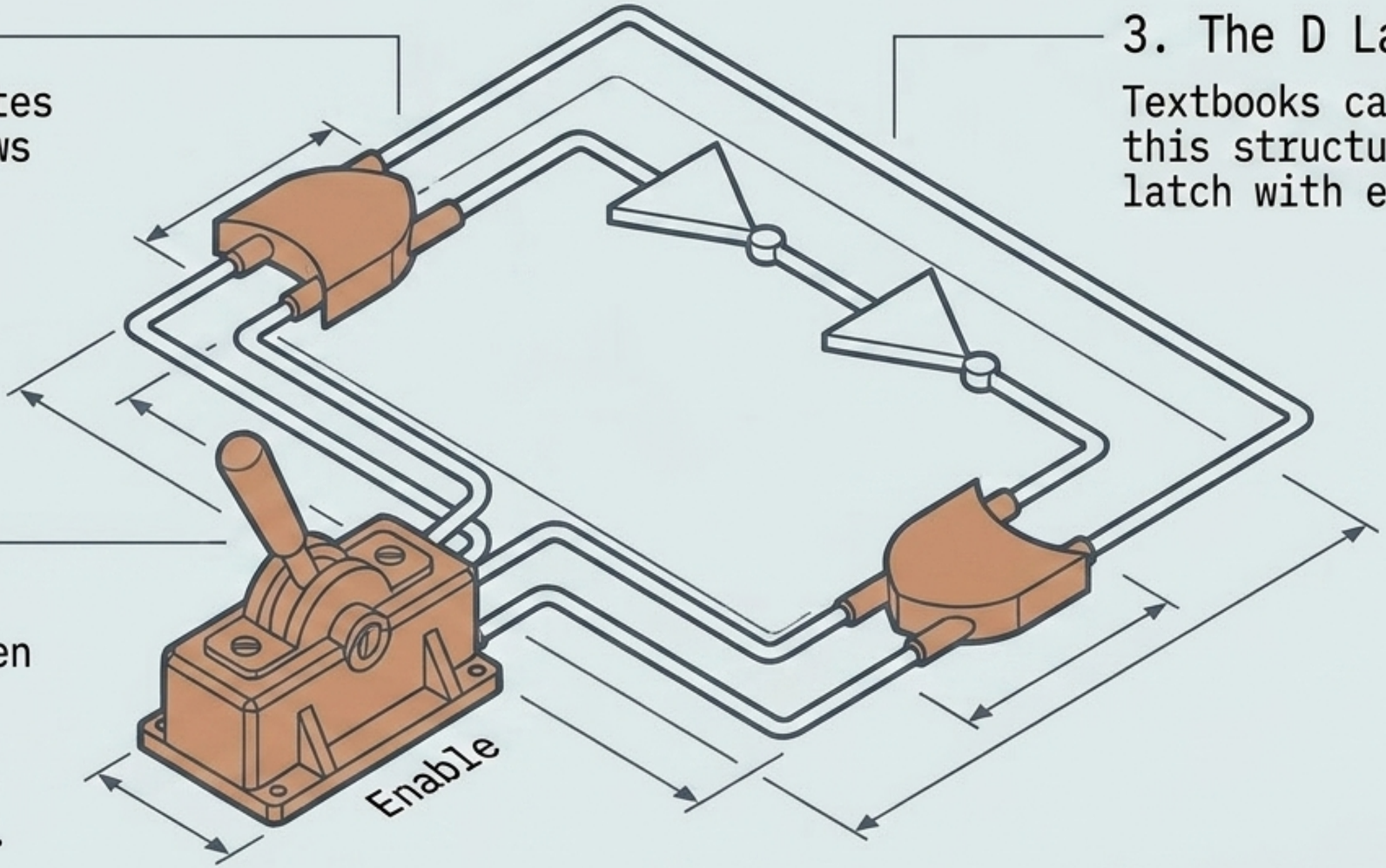
Gaining control with a Write and a Lock

1. The Write:

Dropping two OR gates into the loop allows us to force the stored state to whatever we want.

2. The Lock:

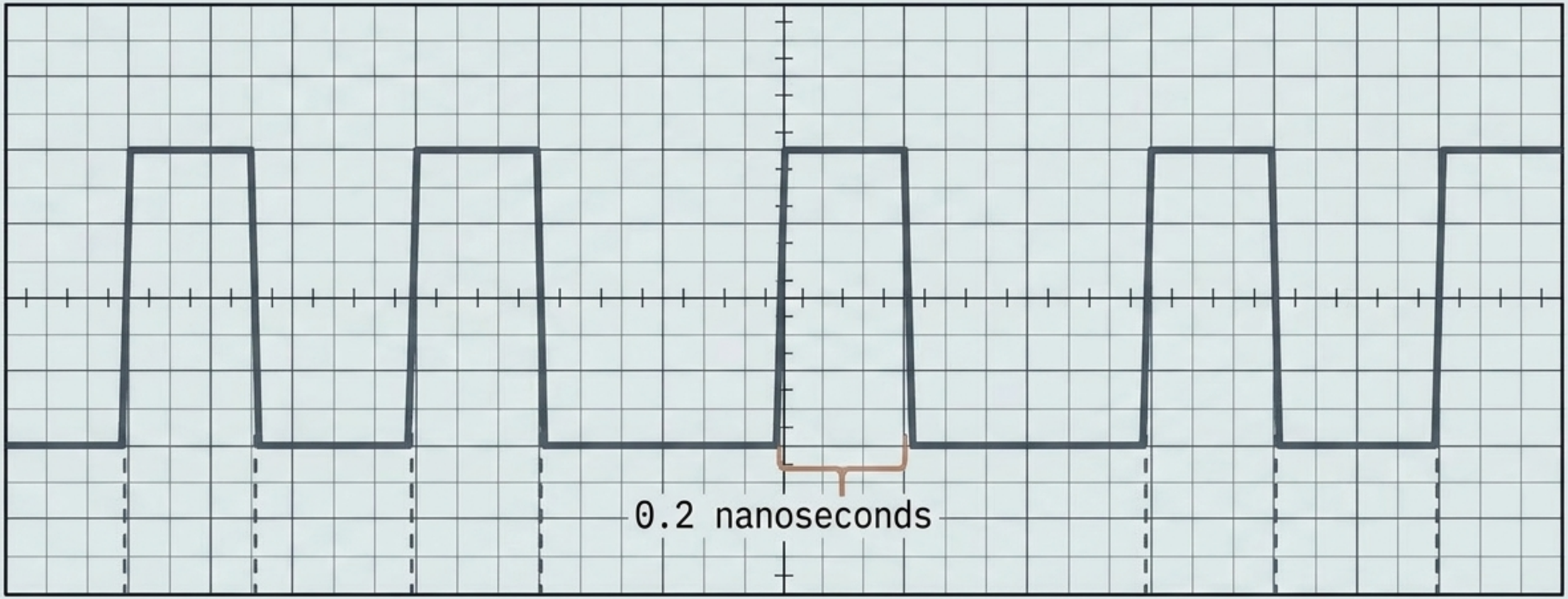
The Enable switch acts as a gate. When High, writing is allowed. When Low, the loop is sealed off and just holds.



3. The D Latch:

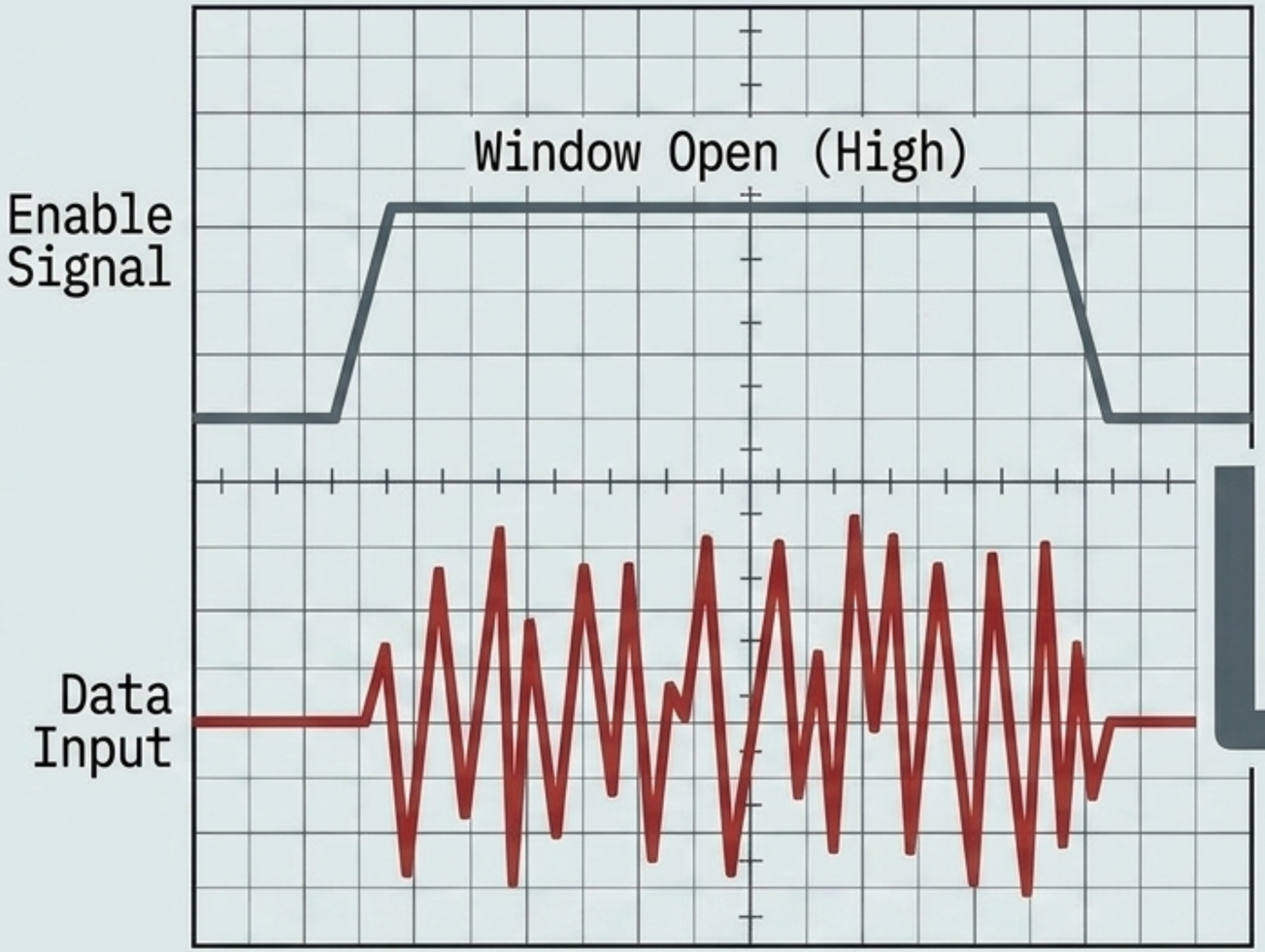
Textbooks call this structure a D latch with enable.

The drill sergeant of the motherboard



Inside a computer, a crystal oscillator produces a steady square wave. One tick, one step forward. Every part of the machine marches to this exact cadence so nothing falls out of step.

The chaos of a continuous write

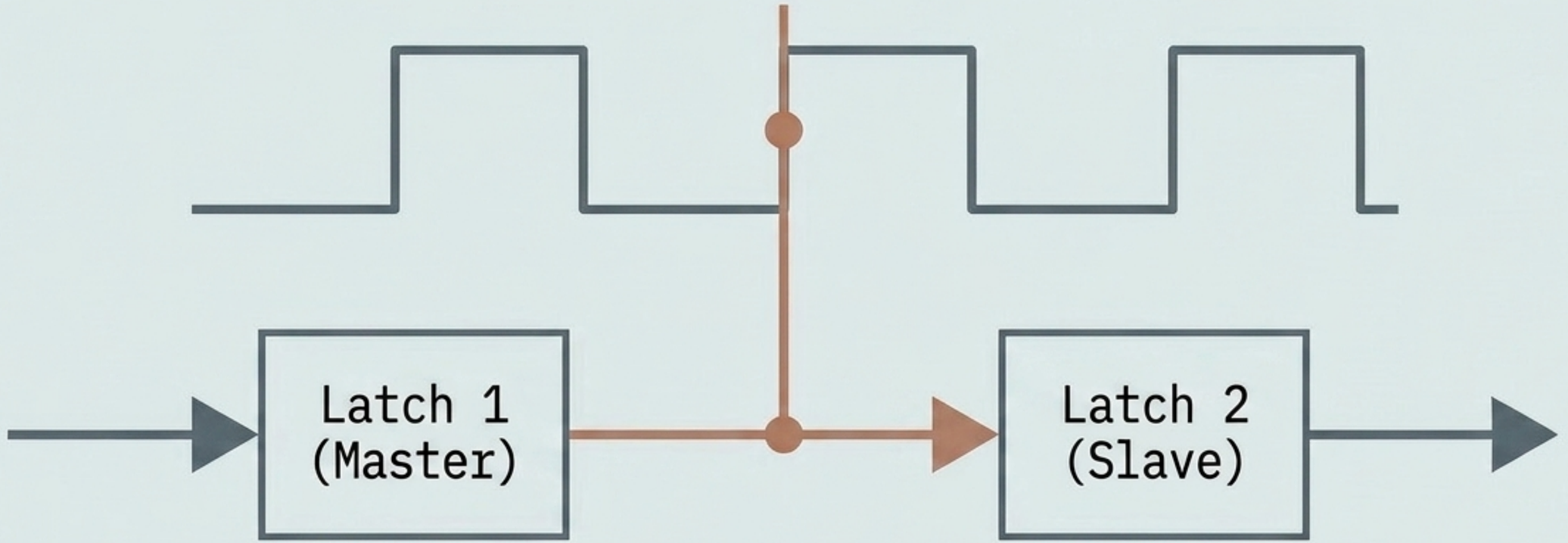


A D latch remains writable the entire time the Enable switch is high.

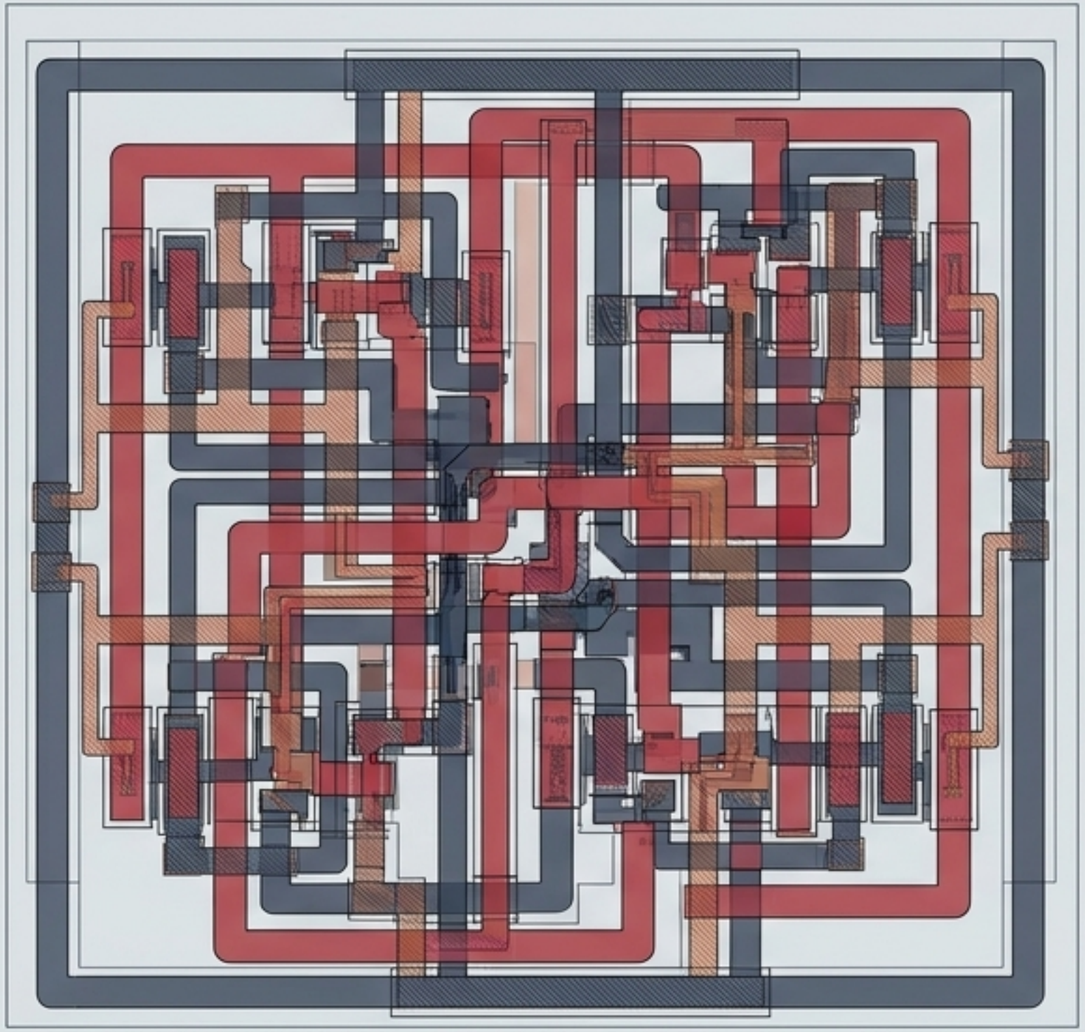
If a fast-fingered input flips the data multiple times within a single clock cycle, the stored value changes erratically, hopelessly confusing the downstream circuits.

Downstream Data Corruption

Exactly one write per cycle



The fix is the Master-Slave Flip-Flop (or D Flip-Flop). By feeding two latches opposite clock levels, new data is only admitted at the exact microsecond the clock jumps. Line up 8 of these, and you have an 8-bit Register.



1 Cell = 12+ Transistors

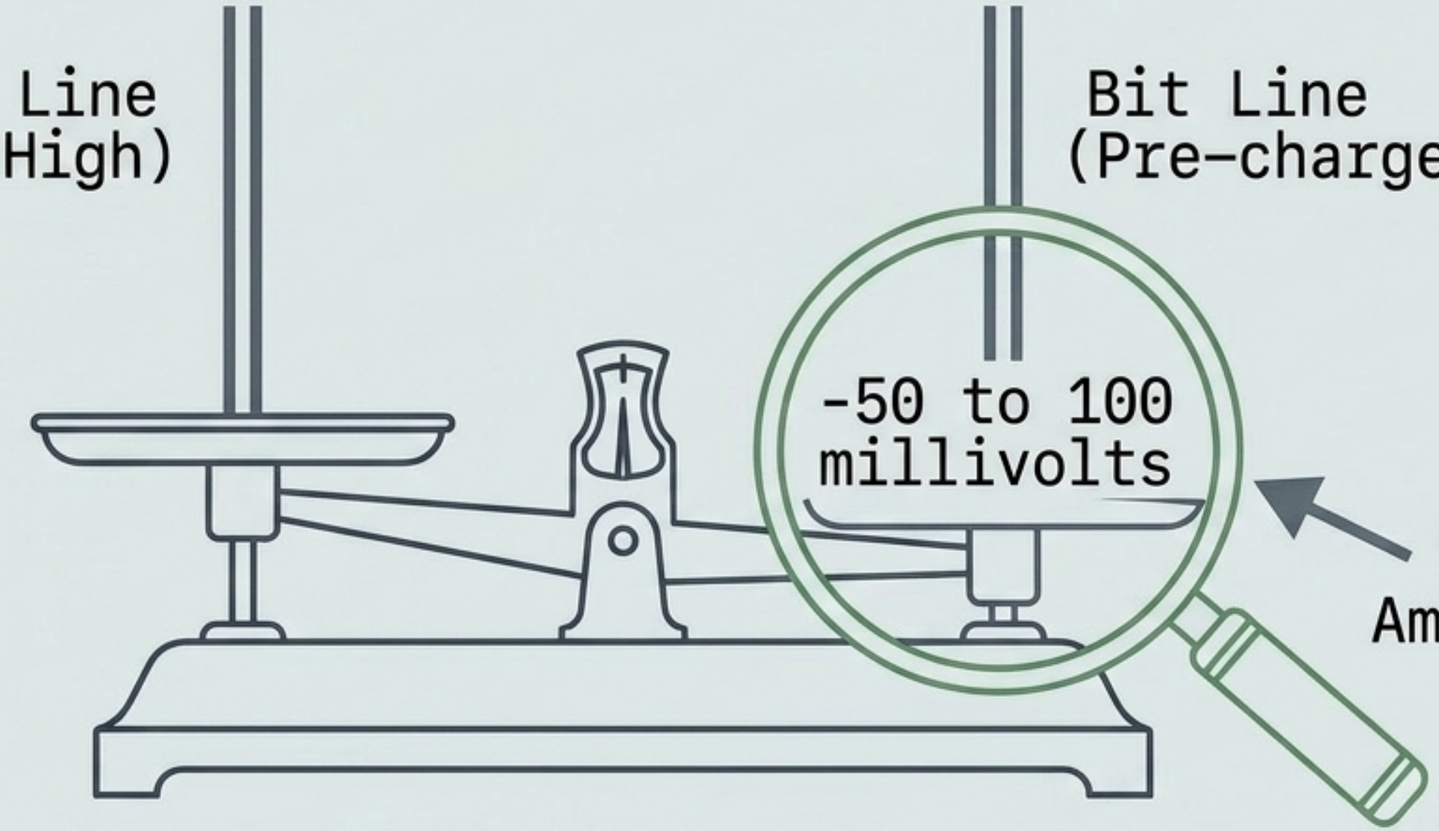
Perfect speed comes with a massive footprint

Registers are absurdly fast because there is nothing in the way—just brute transistors.

The fatal flaw? A single D Flip-Flop requires a dozen-plus transistors.

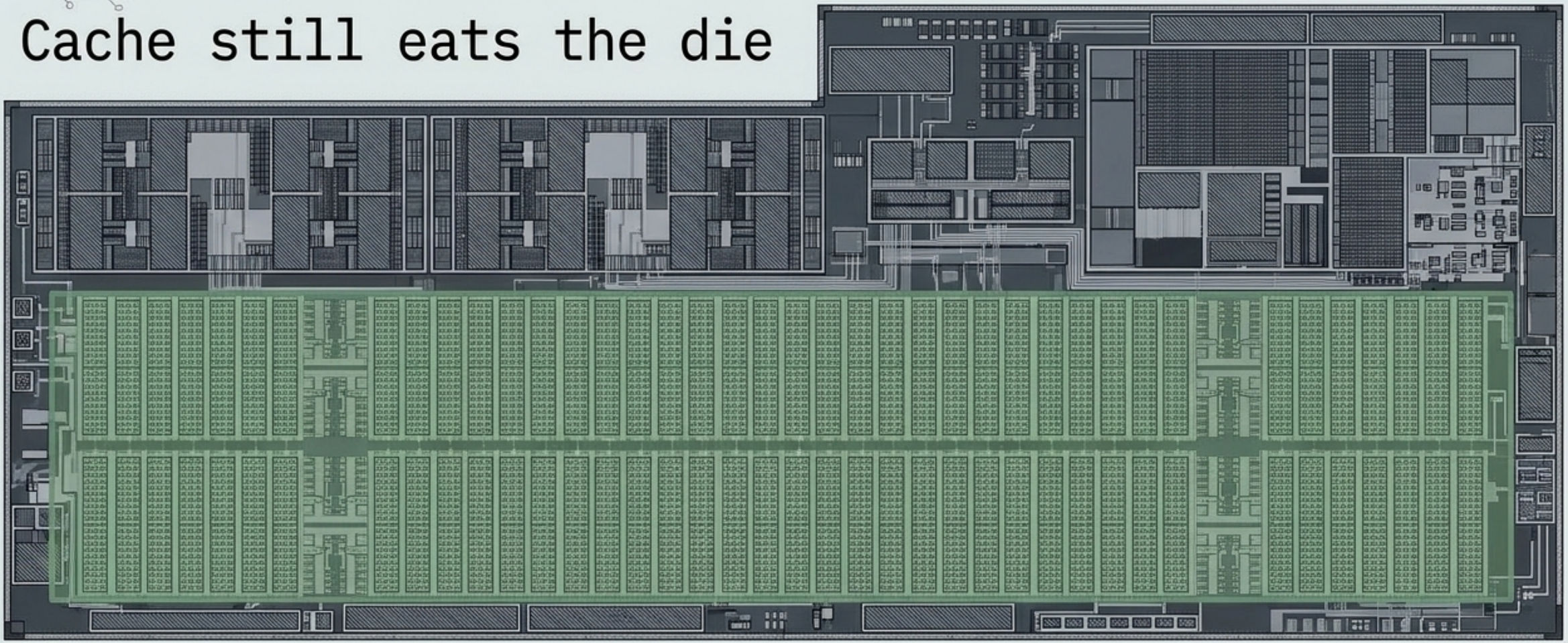
Building megabytes of cache entirely out of registers would physically bankrupt the silicon budget.

The delicate physics of the 6T SRAM read



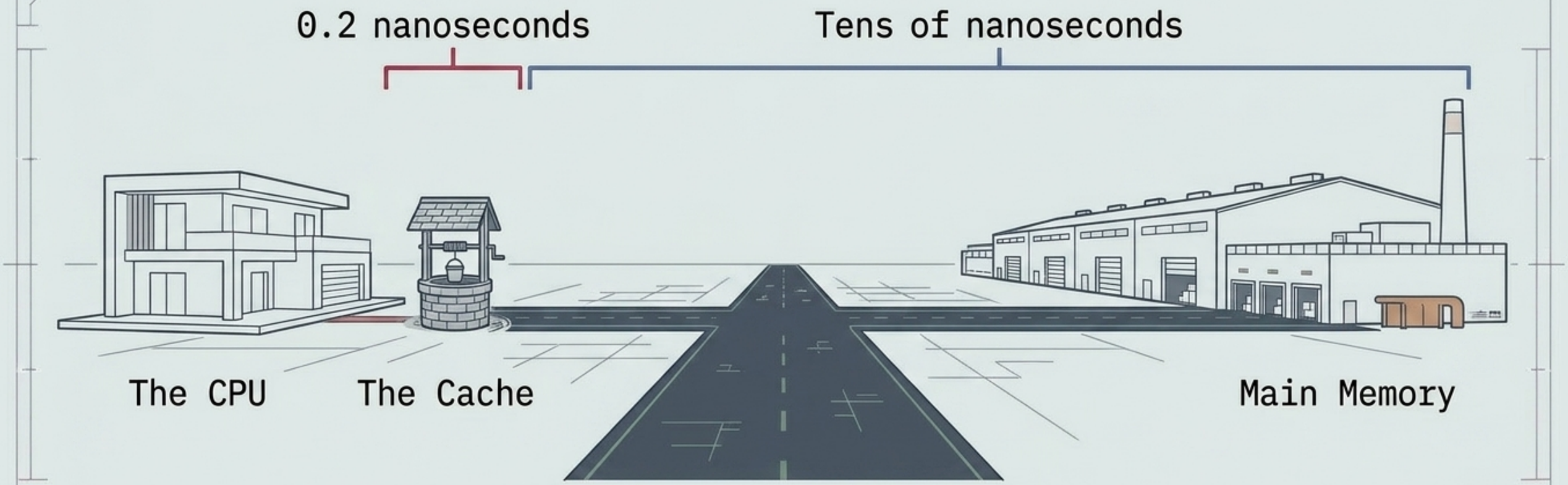
Cache uses a cheaper 6-transistor cell. To read it blisteringly fast, both bit lines are pre-charged. The side storing a ‘1’ holds steady, while the side storing a ‘0’ quietly droops by just 50-100mV. The sense amplifier reads this microscopic voltage tilt instantly.

Cache still eats the die



Even at six transistors per bit, SRAM is an area hog. On modern processors, the L2 and L3 cache devour more than half the total silicon area.

The tyranny of the Memory Wall

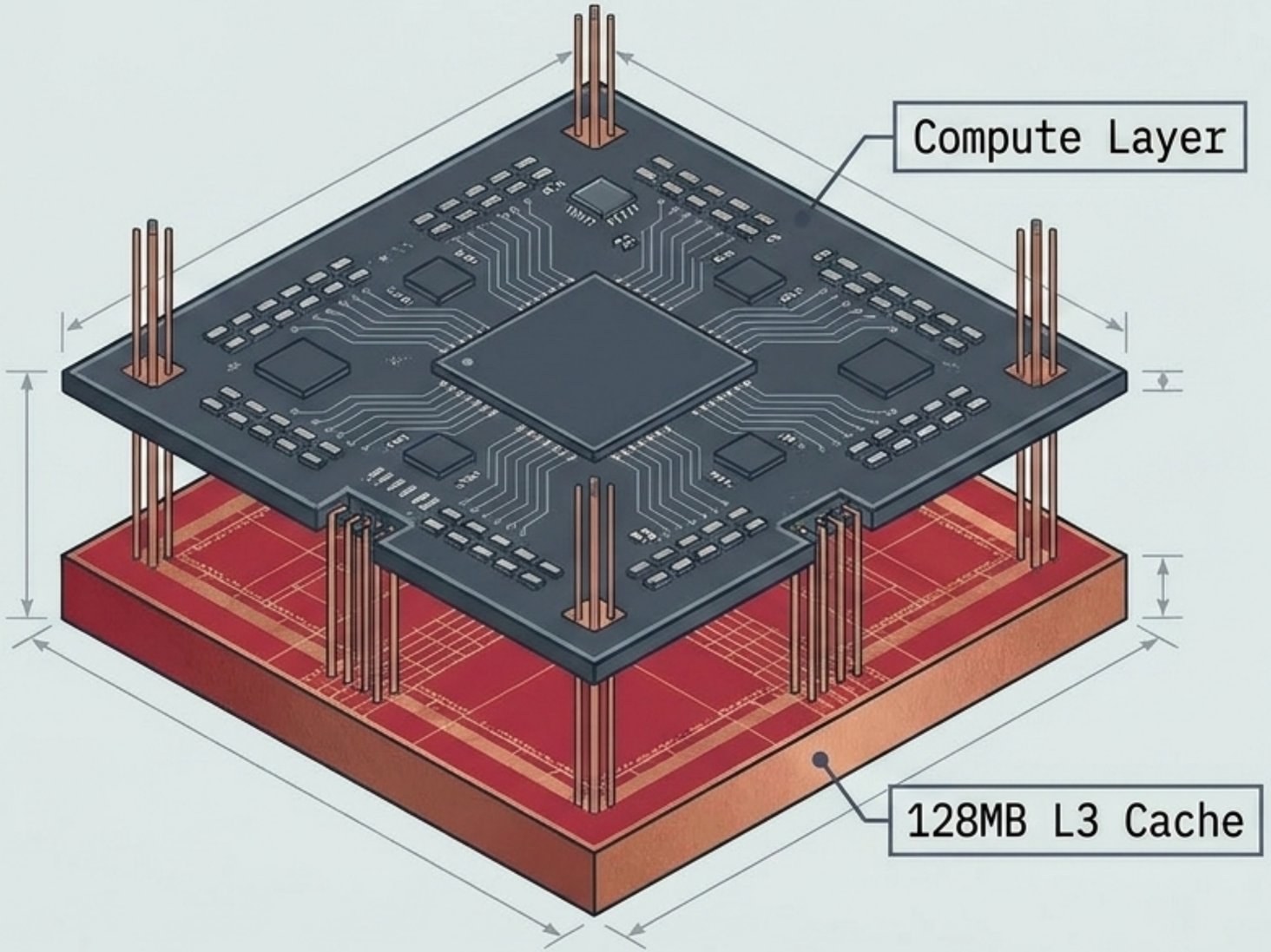


Fetching data from Main Memory takes tens of nanoseconds—long enough for a CPU to run over a hundred instructions while sitting idle. Cache is the well next to the house. More cache means fewer treks down the road.

The Silicon Storage Showdown

Memory Tier	Transistors per Bit	Speed (Latency)	Architectural Role
Registers	12+ Transistors	Sub-nanosecond	The CPU's active hands.
SRAM (Cache)	6 Transistors	~1-3 nanoseconds	The well next to the house.
Main Memory (DRAM)	1 Transistor + Capacitor	Tens of nanoseconds	The warehouse down the road.

If we cannot build out, we build up



Games hammer memory at brutally high frequencies.

AMD's ultimate solution to the physics constraint of SRAM area is the X3D architecture.

They literally build the chip in 3D—stacking a massive, dedicated layer of cache beneath the compute layer to force capacity up to 128MB.

The fatal flaw of speed

Registers and Cache are blistering fast, but they share one absolute weakness: they rely on a constant loop of electrical reminders. Cut the power, and the loops starve. Everything vanishes. To survive a power-off, data must be handed down to main memory and disks—where a whole new physics takes over.

10110100 11101011 01010101 11001100 00110011