

Semiconductor Technology from A to Z

Digital technology

The SRAM Cell	3
<i>Vom Schaltbild zum Chip: EDA</i>	3
<i>Two Cross-Coupled Inverters as the Storage Core</i>	3
<i>The Access Transistors</i>	4
<i>Writing</i>	4
<i>From a Single Bit to an Array</i>	4
<i>Reading</i>	5

The SRAM Cell

Vom Schaltbild zum Chip: EDA

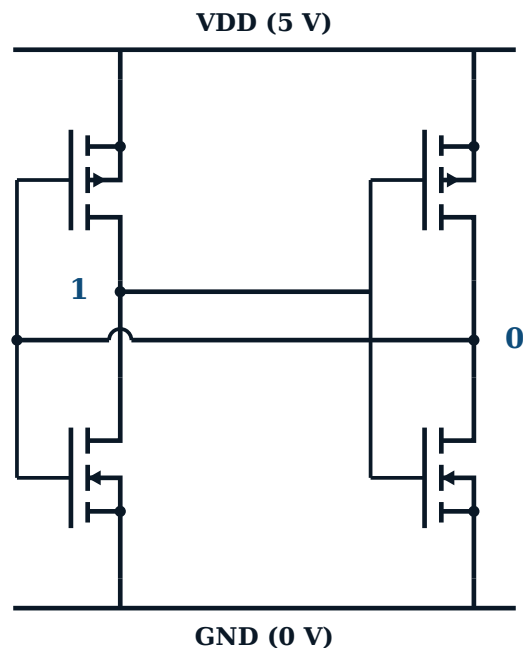
All the circuit diagrams in this section — from the single transistor to the SRAM cell — were created using Electronic Design Automation (EDA): the software toolchain used to design integrated circuits today. An EDA schematic like the one shown here is more than an illustration — it's the input for the next design steps: simulation verifies the electrical behaviour, place & route generates the physical layout on the wafer, and verification checks that the two match. An SRAM cell is a particularly good example of this, because it's repeated identically millions of times on a modern chip — exactly the kind of regular, highly optimised structure that makes EDA tools indispensable.

Storing a single bit sounds like a trivial task at first — but it's actually one of the most common circuit blocks of all. Every processor cache, every register file consists of millions to billions of such cells. The SRAM cell (Static Random Access Memory) doesn't store its bit as charge on a capacitor like DRAM, but as the stable switching state of two cross-coupled inverters. As long as the supply voltage is present, the stored value is retained — no refresh needed, but also without DRAM's high packing density, since a single bit takes six transistors instead of one.

Two Cross-Coupled Inverters as the Storage Core

At the heart of the cell are two CMOS inverters (familiar from chapter 3: one PMOS as pull-up, one NMOS as pull-down), whose inputs and outputs are cross-connected — the output of one inverter drives the input of the other. This feedback creates two stable states: if node 1 is High, that forces the second inverter to hold node 0 Low, which in turn confirms the first inverter's node 1 stays High. The state stabilises itself, with no clock or refresh needed — hence "static". Flipping it requires writing against this feedback from outside with sufficient drive strength.

Storage core: two cross-coupled inverters



The Access Transistors

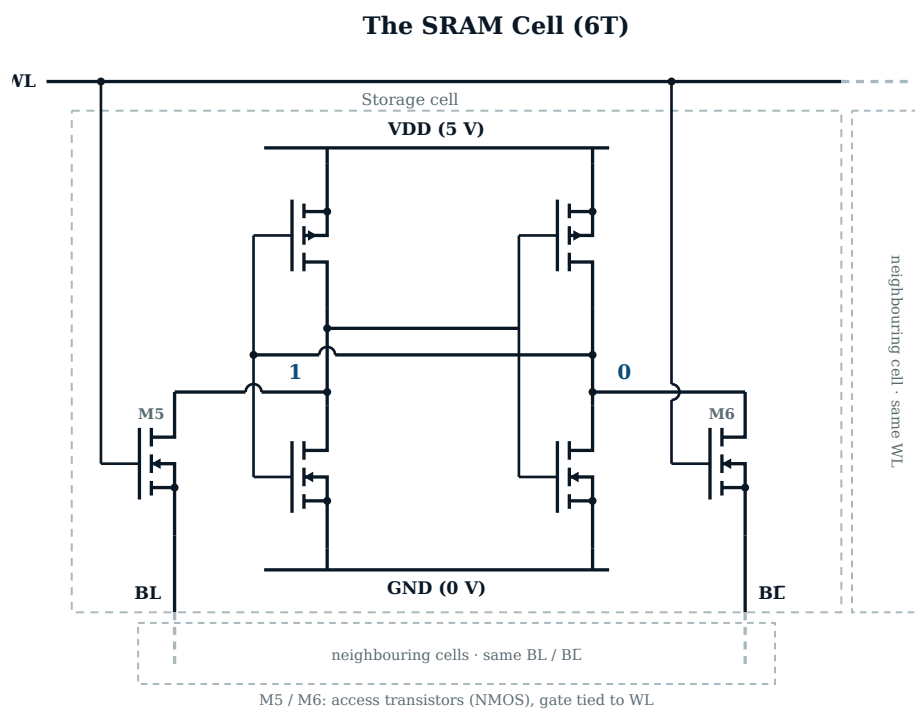
The two stored nodes 1 and 0 aren't connected directly to the cell's input/output lines, but each through its own NMOS access transistor to the bit lines BL and $\overline{BL}$. Their gates are tied together to the word line (WL). When WL is Low, both access transistors are fully off — the cell is disconnected from the bit lines and holds its state in isolation. Only when WL is pulled High do both transistors turn on simultaneously, connecting node 1 to BL and node 0 to $\overline{BL}$. This one shared word line is later used to address a whole row of cells in the memory array at once.

Writing

To write, BL and $\overline{BL}$ are driven externally, low-impedance, to an inverted pair of levels (e.g. BL = 5 V, $\overline{BL}$ = 0 V for a "1"), while WL is activated. The external drivers are deliberately made stronger than the cell's own feedback loop — they override the current state through the access transistors and force the inverters into the new, desired state. Once WL falls Low again, the two inverters hold the freshly written value on their own.

From a Single Bit to an Array

A cell's word line (WL) isn't wired individually — it runs through an entire row of neighbouring cells, and every cell in that row shares the same WL. Perpendicular to it, each bit-line pair (BL/BL) runs through every cell in a column. This forms a grid: only the cell at the intersection where WL is active *and* the matching bit lines are present is actually addressed during a write or read — every other cell in the same column stays isolated by its own, inactive word line, even though it hangs off the same bit lines. This row/column addressing lets a comparatively small number of lines individually address an array of millions of cells.



Reading

Before every read access, both bit lines are first precharged to a shared mid-level potential. Activating WL then pulls one of the two lines slightly toward the stored state through the cell's access transistors, while the other stays close to the precharge level — a small voltage differential builds up between BL and BL. A downstream sense amplifier detects even this tiny difference and amplifies it to a clean digital level. Importantly, the read itself only weakly disturbs the stored state — unlike DRAM, the cell's content is left unchanged.