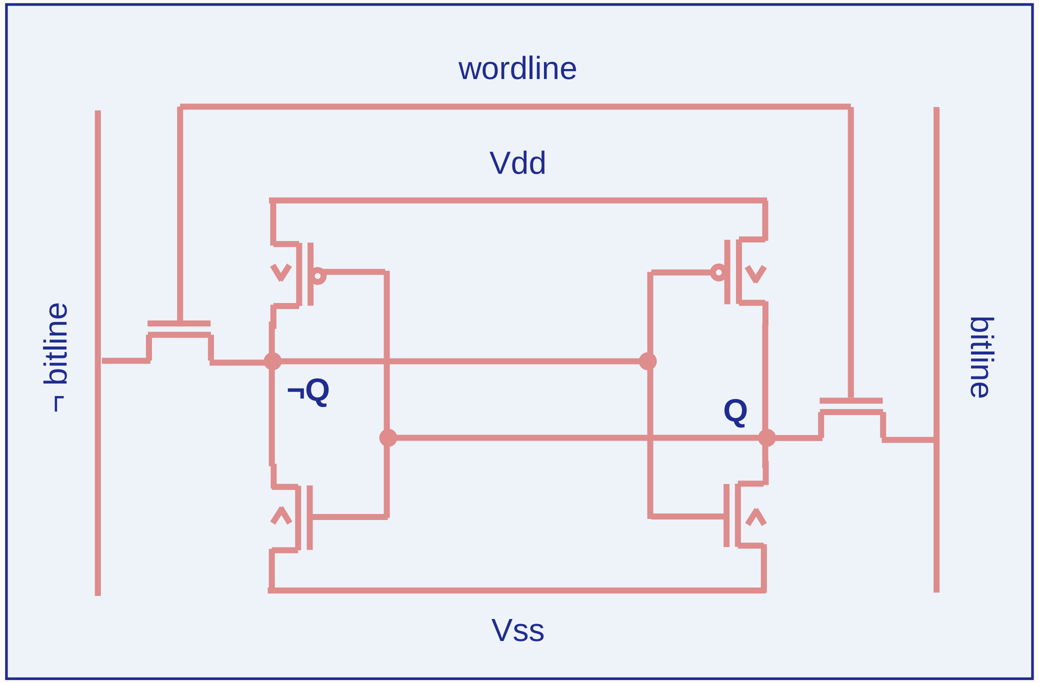
SRAM VS DRAM

Kira Clements, University of Bristol

SRAM

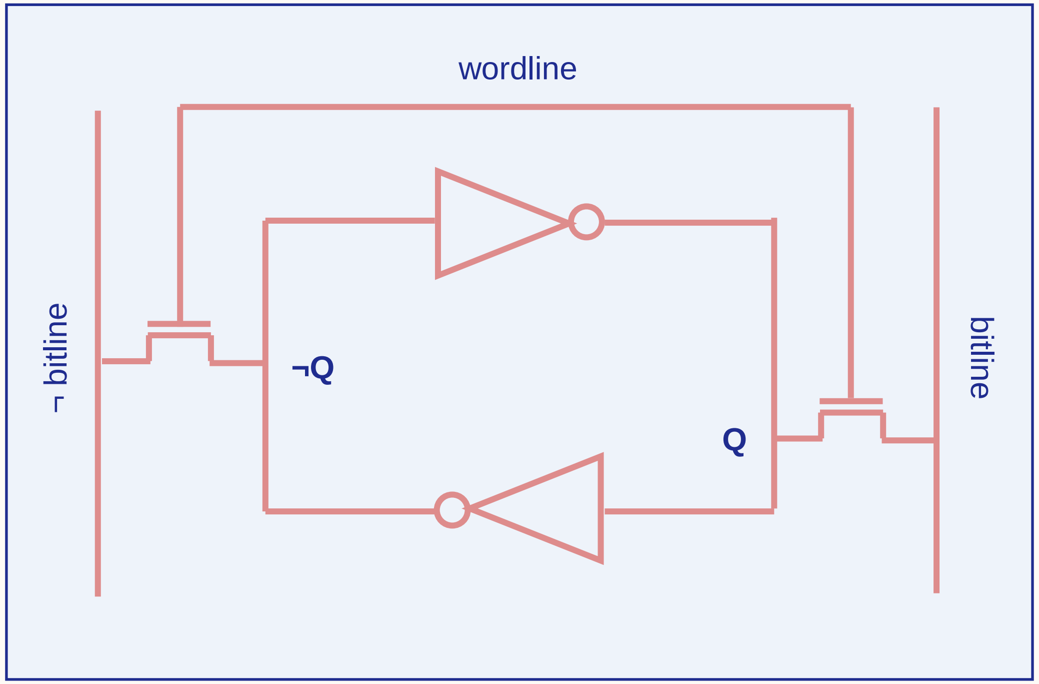
Static Random Access Memory (SRAM) keeps a value in memory as long as power is supplied to the cell, hence the name *static*.



A SRAM cell has two outputs; *bitline* and ¬*bitline*. When the *wordline* input is high, both n-type transistors it's connected to are turned ON, which allows a data value to be transferred to or from the bitlines.

SRAM

We can simplify this diagram by replacing something we've seen previously… Inverters!



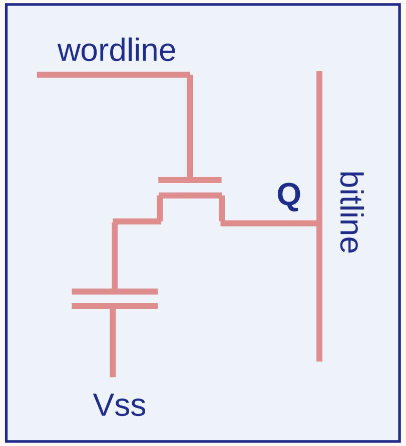
It's the presence of these **cross-coupled inverters**, that creates a bistable circuit and stores the bit value, as if Q = 0, then ¬Q = 1, then Q = 0 etc...

DRAM

Dynamic Random Access Memory (DRAM), on the other hand, stores a value for a short period of time before it needs to be *refreshed*.

The bit value is stored on a **capacitor**, which can be charged to store a 1, or discharged to store a 0.

A n-type transistor is then used to connect the capacitor to the *bitline*.



Turning the transistor ON (to read or write to the DRAM cell) will make the capacitor discharge a little. Even when off, the transistor leaks current, hence the need to **refresh periodically**, even if we don't access a cell.

COMPARING MEMORY

We’ve now seen three different types of volatile memory storage; flip-flops, SRAM, and DRAM. *But how do these compare?*

The data bit stored in a **flip-flop** is available immediately at its output, meaning data access speed is fast. However, flip-flops also take at 12-30 transistors to build, making them relatively expensive in terms of cost, power and size.

DRAM latency is longer than that of SRAM because its bitline is not actively driven by a transistor. DRAM must wait for charge to move (relatively) slowly from the capacitor to the bitline. However, it is built use only 1 transistor plus a capacitor, so is relatively cheap and compact.

SRAM designs use 4-6 transistors, and the access time sits somewhere between flip-flops and DRAM memory components.

The best memory type for a particular design depends on the speed, cost, and power requirements!