Technology Brief 10
Computer Memory Circuits

The storage of information in electronically addressable devices is one of the hallmarks of all modern computer systems. Among these devices are a class of storage media, collectively called ***solid-state*** or ***semiconductor memories***, which store information by changing the state of an electronic circuit. The state of the circuit usually has two possibilities (0 or 1) and is termed a ***bit*** (see Technology Brief 8). Values in memories are represented by a string of ***binary*** bits; a 5-bit sequence $[V_1 V_2 V_3 V_4 V_5]$, for example, can be used to represent any integer decimal value between 0 and 31. How do computers store these bits? Many types of technologies have emerged over the last 40 years, so in this Brief, we will highlight some of the principal technologies in use today or under development. It is worth noting that memory devices usually store these values in arrays. For example, a small memory might store sixteen different 16-bit numbers; this memory usually would be referred to as a 16×16 block or a 256-bit memory. Of course, modern multi-gigabyte computer memories use thousands of much larger blocks to store very large numbers of bits (**Fig. TF10-1**).

Read-Only Memories (ROMs)

One of the oldest, still-employed, memory architectures is the ***read-only memory*** (ROM). The ROM is so termed because it can only be "written" once, and after that it can only be read. ROMs usually are used to store information that will not need to be changed (such as certain startup information on your computer or a short bit of code always used by an integrated circuit in your camera). Each bit in the ROM is held by a single MOSFET transistor.

Consider the circuit in **Fig. TF10-2(a)**, which operates much like the circuit in **Fig. 4-25**. The MOSFET has three voltages, all referenced to ground. For convenience, the input voltage is labeled V_{READ} and the output voltage is labeled V_{BIT}. The third voltage, V_{DD}, is the voltage of the dc power supply connected to the drain terminal via a resistor R. If $V_{\text{READ}} \ll V_{\text{DD}}$, then the output registers a voltage $V_{\text{BIT}} = V_{\text{DD}}$ denoting the binary state "1," but if $V_{\text{READ}} \geq V_{\text{DD}}$, then the output terminal shorts to ground, generating $V_{\text{BIT}} = 0$ denoting the binary state "0." But how does this translate into a permanent memory on a chip? Let us examine the 4-bit ROM diagrammed in **Fig. TF10-2(b)**. In this case, some bits simply do not have transistors; V_{BIT2}, for example, is permanently connected to V_{DD} via a resistor. This may seem trivial,

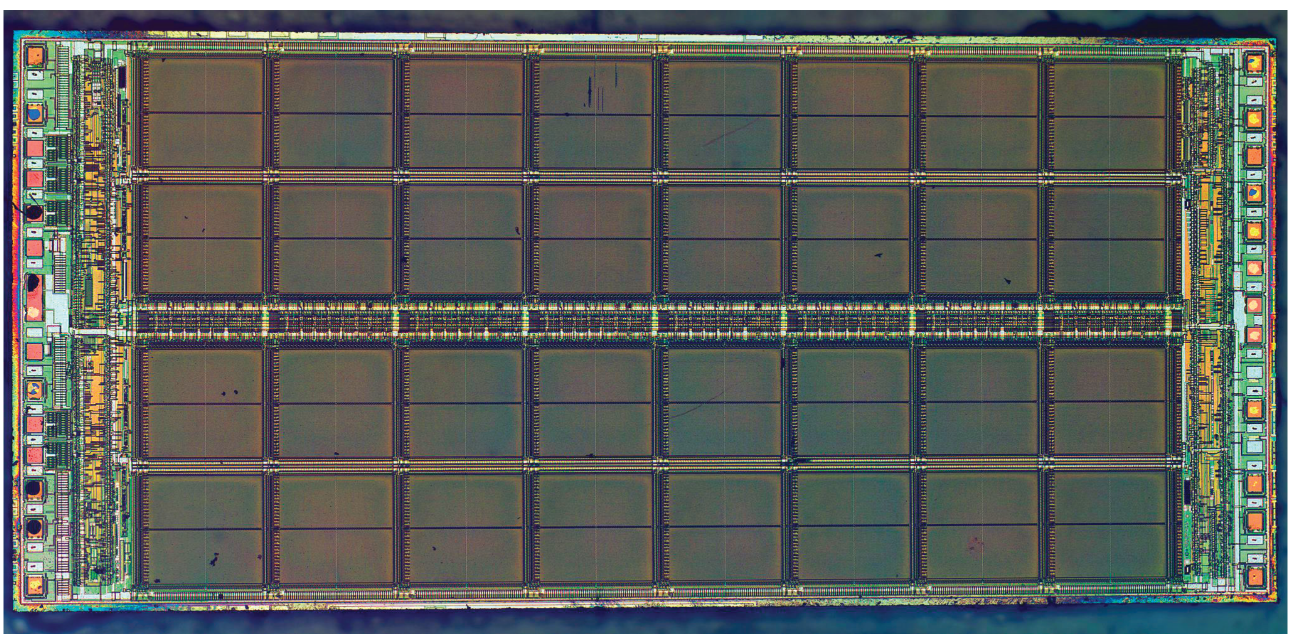
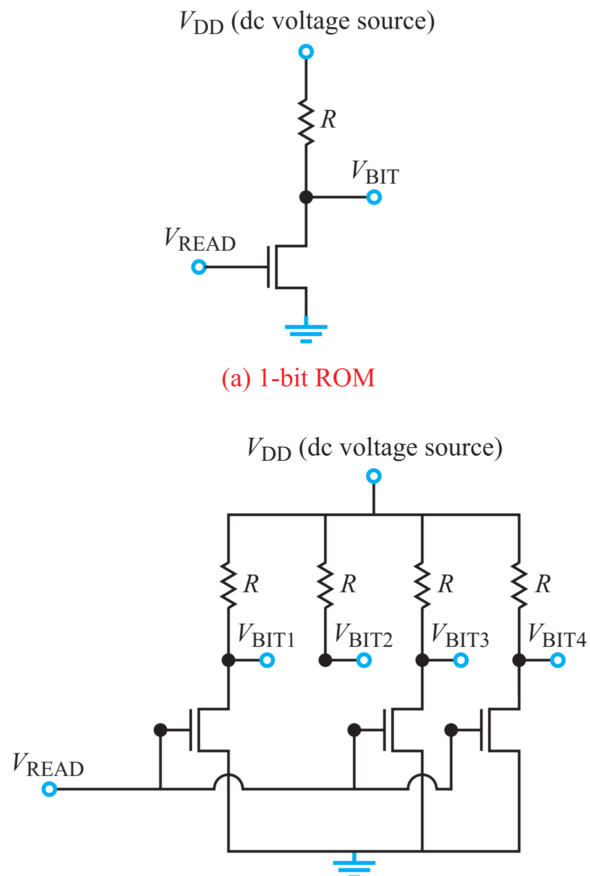


Figure TF10-1: Integrated circuit die photo of a Micron MT4C1024 2^{20}-bit DRAM chip. Die size is 8.662 mm $\times$ 3.969 mm. (Courtesy of ZeptoBars.)



(a) 1-bit ROM

(b) 4-bit ROM

Figure TF10-2: (a) 1-bit ROM that uses a MOSFET transistor, and (b) 4-bit ROM configured to store the sequence [0100], whose decimal value is 4.

but this specific 4-bit memory configuration always stores the value [0100]. In this same way, thousands of such components can be strung together in rows and columns in $N \times N$ arrays. As long as a power supply of voltage V_{DD} is connected to the circuit, the memory will report its contents to an external circuit as [0100]. Importantly, even if you remove power altogether, the values are not lost; as soon as you add power back to the chip, the same values appear again (i.e., you would have to break the chip to make it forget what it is storing!). Because of the permanency of this data, these memories also often are called ***nonvolatile memories*** (NVM).

Random-Access Memories (RAMs)

RAMs are a class of memories that can be read to and written from constantly. RAMs generally fall into two categories: ***static RAMs*** and ***dynamic RAMs*** (DRAMs). Because RAMs lose the state of their bits if the power is removed, they are termed ***volatile memories***. Static RAMs not only can be read from and written to, but also do not forget their state as long as power is supplied. These circuits also are composed of transistors, but each single bit in a modern static RAM consists of four transistors wired up in a bi-stable circuit (the explanation of which we will leave to your intermediate digital components classes!). Dynamic RAMs, on the other hand, are illustrated more easily. Dynamic RAMs usually hold more bits per area than static RAMs, but they need to be refreshed constantly (even when power is supplied continuously to the chip).

Figure TF10-3 shows a simple one-transistor dynamic RAM. Again, we will treat the transistor as we did in

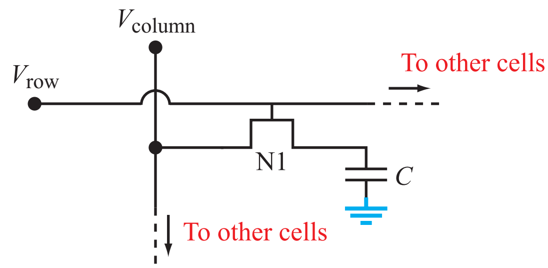


Figure TF10-3: 1-bit DRAM cell.

Section 4-11. Note that if we make $V_{ROW} > V_{DD}$, then the transistor will conduct and the capacitor *C* will start charging to whatever value we select for V_{COLUMN}. When writing a bit, V_{COLUMN} usually is set at either 0 (GND) or 1 (V_{DD}). We can calculate how long this charging-up process will require, because we know the value of *C* and the transistor's current gain g (see Section 5-7). When the capacitor is charged to V_{DD}, a value of 1 is stored in the DRAM. Had we applied instead a value of zero volts to V_{COLUMN}, the transistor would have discharged to ground (instead of charged to V_{DD}) and the bit would have a value of 0. However, note that unlike the ROM, the state of the bit is not "hardwired." That is, if even tiny leakage currents were to flow through the transistor when it is not on (that is, when $V_{ROW} < V_{DD}$), then charge will constantly leak away and the voltage of the transistor will drop slowly with time. After a short time (on the order of a few milliseconds in the dynamic RAM in your computer), the capacitor will have irrecoverably lost its value. How is that mitigated? Well, it turns out that a modern memory will read and then re-write every one of its (several billion) bits every 64 milliseconds to keep them refreshed! Because each bit is so simple (one transistor and one capacitor), it is possible to manufacture DRAMs with very high memory densities (which is why 1-Gbit DRAMs are now available in packages of reasonable size). Other variations of DRAMs also exist whose architectures deviate slightly from the previous model—at either the transistor or system level. ***Synchronous Graphics RAM*** (SGRAM), for example, is a DRAM modified for use with graphics adaptors; ***Double Data Rate 4 RAM*** (DDR4RAM) is a fourth-generation enhancement over DRAM which allows for faster clock speeds and lower operating voltages.

Advanced Memories

Several substantially different technologies are emerging that likely will change the market landscape—just as Flash memories revolutionized portable memory (like your USB memory stick). Apart from the drive to increase storage density and access speed, one of the principal drivers in today's memory research is the development of non-volatile memories that do not degrade over time (unlike Flash).

The ***Ferroelectric RAM*** (FeRAM) is the first of these technologies to enter mainstream production; FeRAM replaces the capacitor in DRAM (**Fig. TF10-3**) with a ferroelectric capacitor that can hold the binary state even with power removed. While FeRAM can be faster than Flash memories, FeRAM densities are still much smaller than modern Flash (and Flash densities continue to increase rapidly). FeRAM currently is used in niche applications where the increased speed is important. ***Magnetoresistive RAM*** (MRAM) is another emerging technology, currently commercialized by Everspin Technologies (spun out from Freescale Semiconductor), which relies on magnetic plates to store bits of data. In MRAM, each cell is composed of two ferromagnetic plates separated by an insulator. The storage and retrieval of bits occurs by manipulation of the magnetic polarization of the plates with associated circuits. Like FeRAM, MRAM currently is overshadowed by Flash memories, but improvements in density, speed, and fabrication methods may make it a viable alternative in the mainstream consumer market in the future. Even more speculative is the idea of using single carbon nanotubes to store binary bits by changing their configuration electronically; this technology is currently known as ***Nano RAM*** (NRAM).