

Storage

Everything in the computer is stored using binary patterns, but there are several different methods of doing this. For each type of storage, we need to consider how it will store the binary patterns, whether it is volatile, the cost, and the speed, as well as other characteristics specific to each type of storage. The main types of computer storage are electrical, magnetic, optical, and solid state.

Electrical Storage

Electrical storage is what we typically picture when we think of computer storage: binary is stored using electrical power off for 0 and electrical power on for 1. We can read data based on whether power comes out, and write data by charging or not charging.

Main memory typically uses a kind of electrical storage called DRAM – Dynamic Random Access Memory¹. This type of storage is very simple, using a simple electrical component called a capacitor, which can hold an electrical charge briefly. However, the charge fades, so to keep the parts of the pattern that should be on *on*, we have to regularly refresh – charging them up again. This is why it is called “dynamic.” The time spent refreshing can’t

¹ Remember that the “random” part of Random Access Memory is no longer very meaningful. At one time, much computer storage used large magnetic tapes. These were “sequential access”: if you were currently reading data at one end of the tape, to get to data at the other end required a long process of winding through the tape. So, storage that allowed jumping right from the first address to the last to anything in between was a great time savings. However, except for long term backup, we almost never use sequential access anymore.

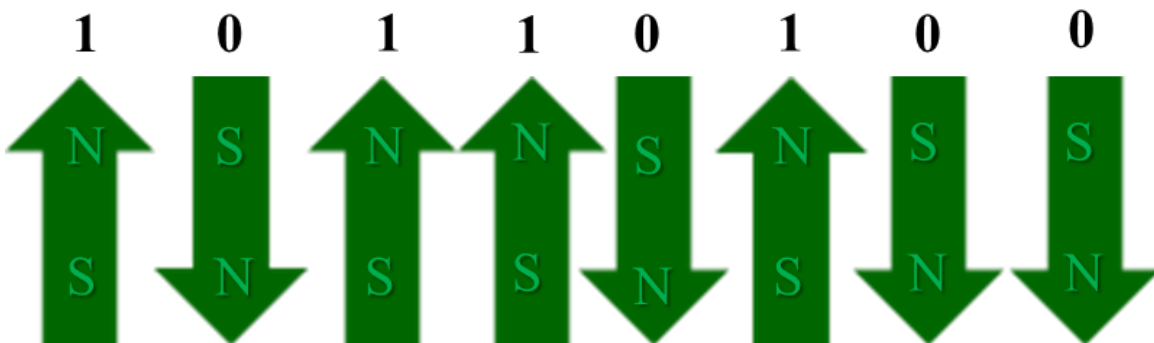
be spent reading and writing data, and there are various tricks (SDRAM, DDR3, etc.), to speed DRAM up to make up for this.

The cache typically uses SRAM – Static Random Access Memory. SRAM is more complex than DRAM, with a circuit keeps the electrical charge constantly cycling through. This means that SRAM does not require refreshing; as long as there is power to the system, on stays on. This makes it faster than DRAM, but being more complex means it is also more expensive.

All electrical storage is volatile. Compared to other types of storage, electrical storage is fast, and expensive.

Magnetic Storage

The most common type of storage to use for long term, such as hard drives, is magnetic storage. To picture how we store data in magnetic storage, you can picture a bar magnet, with North and South Poles. If we take a row of these, we can rotate them with South end up to store 0 and North end up to store 1. We can then use another magnet hovering above these to read North vs South (since two Norths repel while North and South attract each other), and we can write by flipping a magnet over².



² In modern magnetic storage, we in fact use magnetized vs. demagnetized for 1 vs 0, but the North vs South model is easier for most people to picture.

Magnetic storage such as a hard drive has a surface covered by very tiny magnetic particles which can be read and written by a magnet hovering just over the surface, called the magnetic read-write head. Once magnets have been set, they do not need power to stay that way, so magnetic storage is non-volatile.

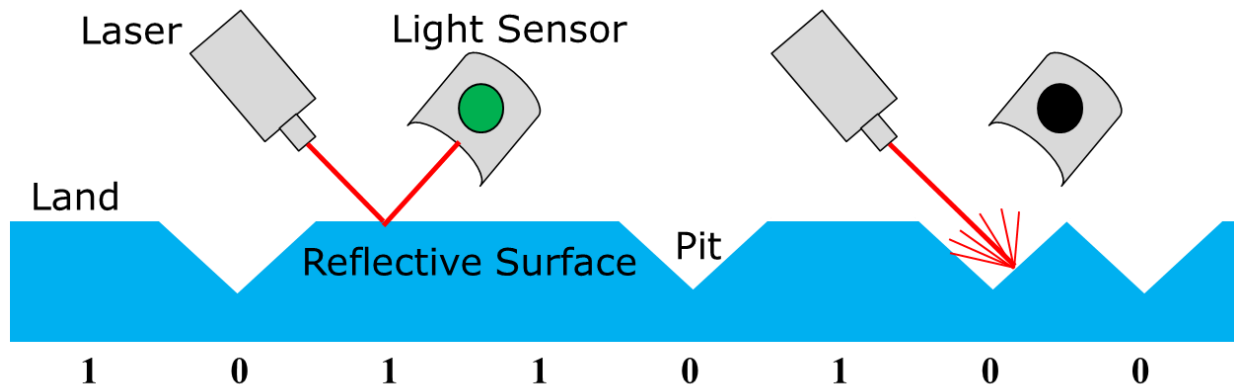
Since all we need is a surface covered in particles, this is a cheap type of storage, but it is slow compared to electrical. This makes it appropriate for hard drives where we need to be able to afford lots of storage space, but we can afford to be a little slower, since any program that runs will first be copied into faster MM when it is to be run.

Since the magnetic read-write head has to be very close to the surface to read the particles, and the particles are very tiny, it is easy for the head to scrape particles from the surface, so magnetic storage is prone to damage, especially if it is moved while the head is reading or writing.

Optical Storage

CDs, DVDs, BluRay discs, etc, use optical storage, which depends on the way light behaves. A laser is a focused beam of light, and we can predict its behavior very accurately. If we bounce a laser off a flat reflective surface at an angle, we can predict exactly where it will bounce to, and place a light-detector at that location. If we make a dent in the reflective surface, the light will instead be broken up and not hit the light detector.

So, we can use the reflective surface to store our data, leaving the surface flat in a small area – a *land* -- for 1, and denting it – a *pit* - for 0. Then we can read the data back by passing a laser over the surface and having the light detector report whether or not it was hit by light.



The amount of data we can get onto the surface of a disc depends on how tiny we make the dents. To read smaller dents, we need a smaller, more focused beam of laser light, so a CD's laser will be unable to read a Blu-Ray disk because it hits an area with several lands and pits at once.

This is a very cheap method of storage – all we need is the reflective surface. It is about as fast to read as magnetic storage. It is also non-volatile – no power is needed to keep the dents in place.

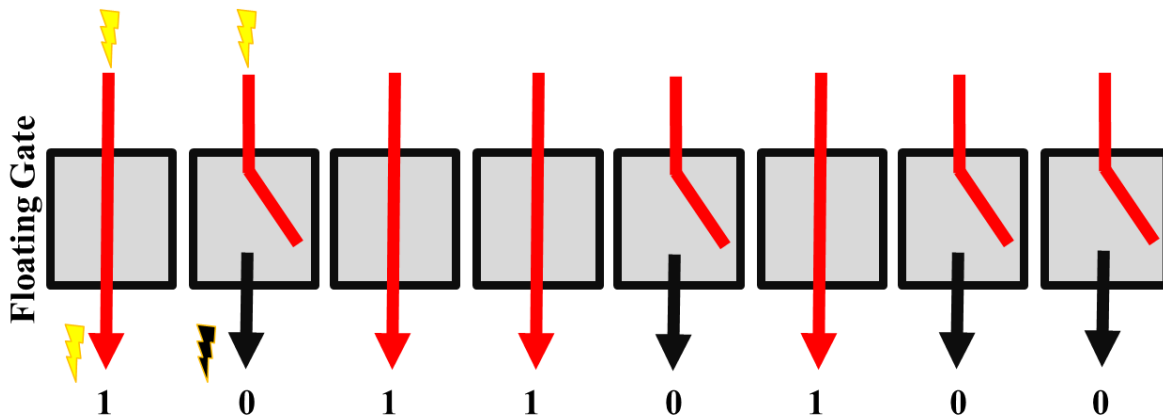
Most optical storage is read only – it can be written once when it is manufactured, but can then only be read, not changed. To make perfect tiny dents requires high-end industrial machinery.

For home use, discs can be written using a slightly different approach. Instead of actual lands and pits, a mostly-light reflective surface is darkened by burning it. The dark spots cause a laser to refract just as a dent does. The result may be quite as reliable as a disc created in a factory.

CD-RW and similar technologies work by allowing the surface to be dented and melted smooth again, making them read-write instead of read-only, so the data stored on the disc can be change. In general people have not found these reliable enough for them to become very popular.

Solid State Storage

Also known as flash storage, solid state storage uses a fairly complex system for storing data, the physics of which is outside the level of this class. However, if you picture a number of boxes with wires going in and coming out, with the wires being either connected or unconnected inside the box, you'll have the right kind of idea³. These are called floating gates. If we put electricity in one wire of a gate, then if they are connected inside, the electricity will come out the other side of the gate, but if they are unconnected it will not.



No power is required to keep the gate connected or not, so solid state storage is non-volatile, but since we read the state of the gates just based on whether or not we detect an electrical charge coming out, this kind of storage is nearly as fast to read as electrical.

It is slightly slower to write, which requires putting a larger charge through the gate to connect a disconnection or disconnect a connection. This larger charge causes a tiny amount of damage each time, so eventually these little gates wear out. So we say

³ The explanation of what's really going on involves the words "field effect transistor" and "quantum tunneling"...

that solid state storage has a *write-endurance issue* – it can only endure so many writes⁴ – but no such problem with reads.

The gates are complex enough that solid state storage is expensive compared to other types of volatile storage, but unlike magnetic storage it is very portable, and not prone to damage, even while reading and writing. We are beginning to use solid-state storage for hard drives since it is fast, non-volatile, and not damage-prone, but this is still fairly expensive.



Storage Sizes

A single binary digit – a 0 or 1 – whether it is represented by a capacitor, a land or pit, a gate, or a magnetic particle, is called a bit. To store anything of interest, we need a pattern of bits. A pattern that is 8 bits long is called a byte. This is enough to store patterns for letters and small numbers. When writing units, we use b for bit and B for byte⁵.

⁴ The average thumb drive could survive having every gate in it re-written once a day every day for several years. In practice, people tend to lose them before they get close to wearing them out.

⁵ We measure almost everything using bytes. Consider why internet service providers would generally choose to measure their speed in *bits* per second...

To store even a text file, we need enough space for thousands of bytes, so we began to use larger units. A kilobyte (KB) is 1024 bytes⁶. 1024 kilobytes is a megabyte (MB). 1024 megabytes is a gigabyte (GB). 1024 gigabytes is a terabyte (TB). Most storage now is measured in GB, with larger storage, such as hard drives, measured in TB.

Word Size

The size of binary patterns used throughout a computer (registers, cache, MM, etc) is called the *word size*.

Suppose you were filling out a form that only had space for 3 digit numbers; you could write 966, but not 2020. Similarly if a computer has word size 4, it can store 1011 but not 11010. So the word size limits the size of the numbers a computer can store.

Modern computers use 32-bit or 64-bit word sizes. Remember that Main Memory has a numbered address for each memory location. A computer that uses 32 bit words can only store numbers up to 4,294,967,295, which is only enough addresses for 4GB of memory space. A computer with a 64 bit word size can store numbers up to 18,446,744,073,709,551,615, which means it could use 1,000,000,000GB of memory space. So changing to a bigger word size also allows computers to use a larger main memory.

Cache

Most modern computers have several small caches, called L1 (Level 1) Cache, L2 Cache, etc. Most of the cache's speed comes

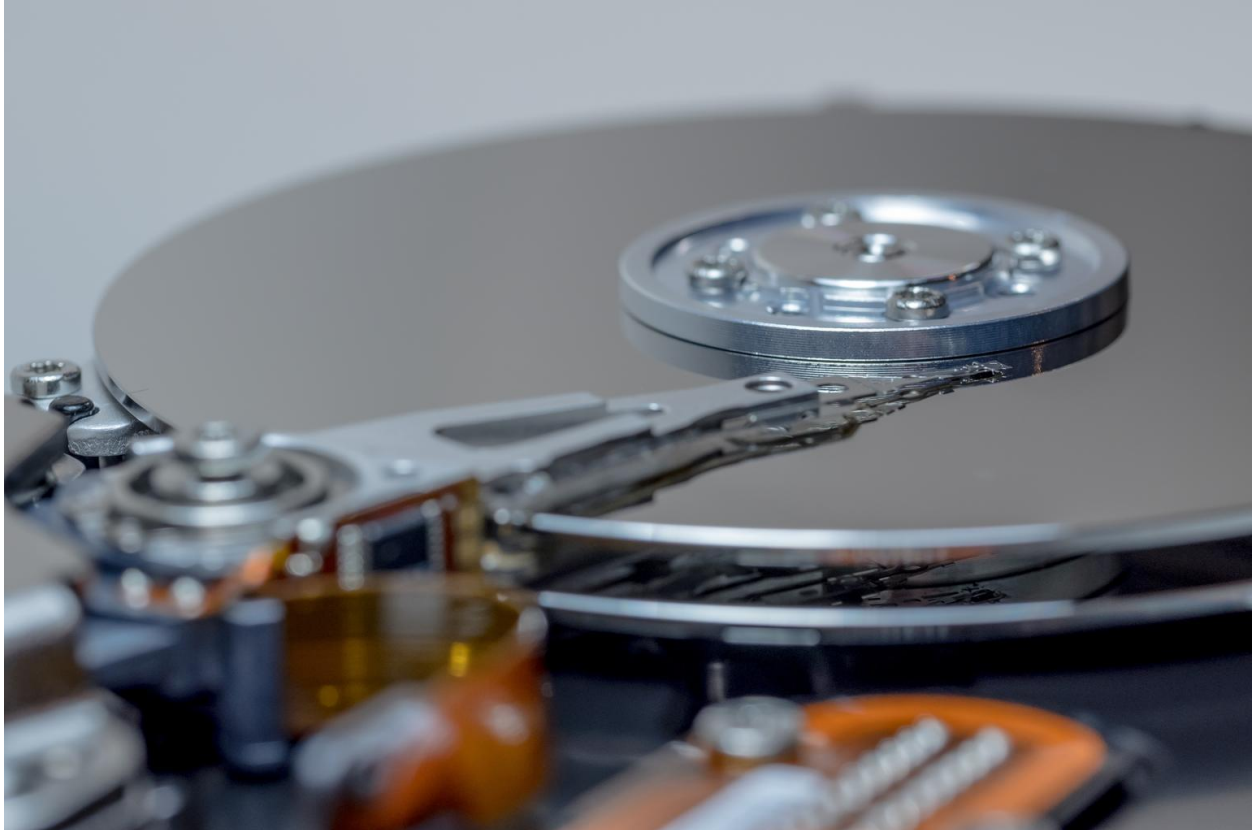
⁶ 1024 is a round number when written in binary. For the purposes of this class, if you think of it as 1000, you will be close enough.

from being made of SRAM, however an odd effect of the way computer storage works is that storing in a *smaller* space actually also gives a very slight boost to the speed.

So if we can afford a certain amount of cache space, it is better to split this into several small caches instead of one large one, to increase this small speed boost. The smallest cache holds the most frequently used data currently in use, and larger caches hold increasingly less-used data. When the CPU needs to load data, it looks first in the smallest cache, then the next, then the next, working its way out to the main memory. This way we save the most time on the data that is used most frequently.

Magnetic Hard Drives

The hard drive of a computer is generally magnetic storage. Inside the drive, tiny magnetic particles sit on round flat platters. To maximize the amount of storage, there are several platters with particles on both sides. The surfaces are divided into blocks which are numbered (like main memory addresses). Within a block are several sectors – the amount that can be read or written by a head at once.

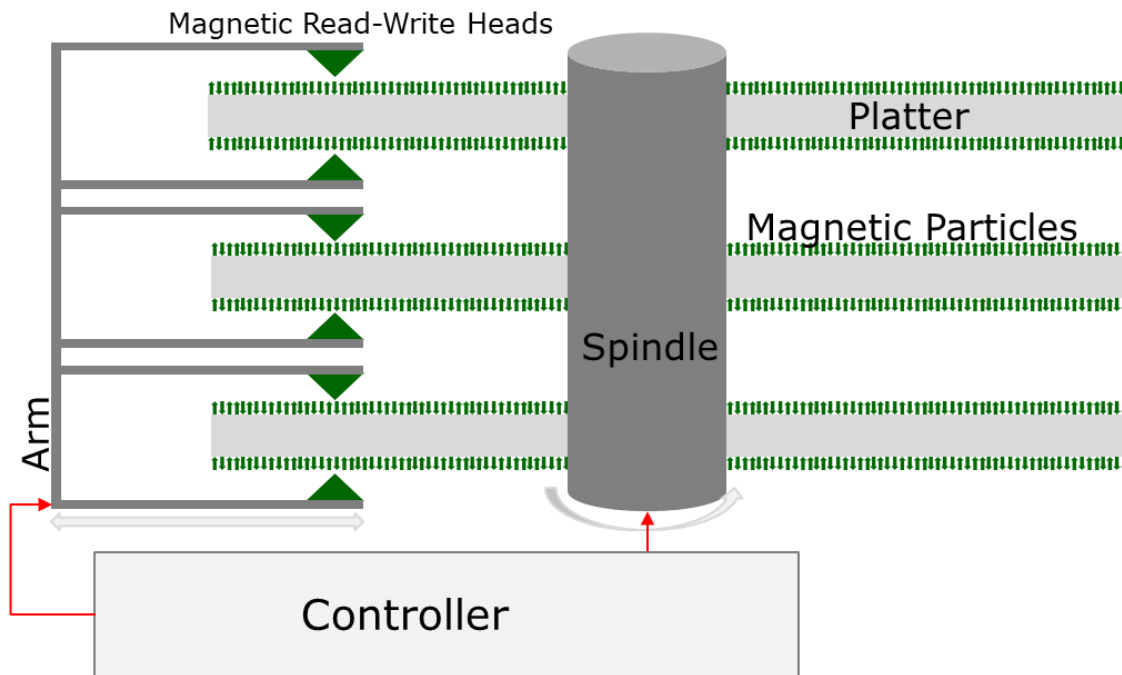


The magnetic read-write head, a magnet that reads and writes the state of the magnetic particles on the hard drive, must go very, very close to the surface to read the tiny particles. But if it actually touches the surface, it will do permanent damage. So the hard drive must be carefully designed to limit the movement of the heads and platters as much as possible – the less movement, the less chance of damage.

For this reason, instead of just having one head, hard drives have a head for each side of each platter, so no head ever has to move from one platter to another. As a bonus, this also means that hard drives can read more than one block at the same time.

The platters rotate all together around a central spindle, and the heads, which all move together on a single structure called the

arm, move in and out. This allows a head to reach any location on the platter.



Waiting for a particular position on the drive to rotate into position under the head takes longest (so this has the most effect on the speed of the hard drive) while the movement of the heads on the arm is faster. The time spent waiting for the heads to get from one location to the next (rotation and motion of the heads) is called *latency*. A lot of latency means that reading and writing to the hard drive will be slower.

The controller of the hard drive is in charge of spinning the platters on their spindle, moving the heads in and out on their arm, telling the heads to read or write, and communicating with main memory through the bus.

Reading from the Hard Drive

When we want to read data from the hard drive, for instance when we open a file, we need to tell the controller which block has the data we want, by sending a block address down the bus to the controller. To read the block, we need the heads to be over the right locations. The controller positions the heads over all the sectors in the chosen block by spinning the platters and moving the heads. Then the heads read the data they are over. The controller then sends this data up the bus to be stored in main memory.

Writing the Hard Drive

When we want to write data to the hard drive, for instance when we save a file, we need to give it that data (currently in main memory) and also tell the controller which block the data should be stored in. So we send the data and the block address down the bus to the controller. Then the controller positions the heads over the right locations (spinning the platters and moving the heads). Then the heads write the data into the locations on the drive.