the modern computer

Modle introduction

Isn't the history of computers supper interesting. I love going back in time and seeing how we got to this exciting point in computing. You've already taken the first few steps to building your foundational knowledge of IT. Before we dive deeper, I'll just take a moment to introduce myself. My name is Devan Sri-Tharan. I've been working in IT for 10 years. I'm a Corporate Operations Engineer at Google, where I get to tackle challenging and complex IT issues. Thinking back, my first experience with tech began when I was about nine years old. My dad brought home the families first computer. I remember my dad holding a floppy disk and telling me that there was a game on it. To my dad's amazement, I somehow managed to copy the game from disk onto the computer's hard drive. While it might seem like a trivial task now, this device was just so new to us back then. Sure, I loved the different games I could play but what I really loved tinkering with the machine, trying to get it to do what I want it to do. While that floppy disk in computer might have ignited my passion for technology, was actually my first few job experiences that really started to shape my IT career. One was in retail selling baby furniture and the other was at a postal store where I helped customers chip their packages and became the one-person IT crew. It might sound all that working in retail inspired my career, but I've realized I've really enjoyed communicating with customers, trying to understand their needs and offering a solution. My first experience working directly in IT was in college as an IT support specialist intern. From there, I worked as an IT consultant to decommission an entire IT environment. This was my first experience working directly with large IT infrastructure and pushing myself outside my comfort level as a college student. I bring up these few jobs for a reason. These experiences helped shape my career in IT. I knew at that time that I wanted to go into tech, but I struggled where I want it to focus my career. Starting at Google as an IT journalists allowed me to experience many different areas of technology. It allowed me to figure out the jobs I didn't want to do before I was able to identify exactly what I did want to do. Really passionate about IT infrastructure, but you can't understand infrastructure until you understand hardware. Let's dig in, in IT, hardware is an essential topic to understand. You might find yourself replacing faulty components or even upgrading an entire fleet of machines one day. By the end of this lesson, you'll be able to describe all the physical parts of computer and how they work together. You'll even be able to build your own computer. Once you figured out how one computer works, you'll be able to understand how any type of computer works. Excited, I am. Let's get started.

Introduction to Computer Hardware

Let's face it, computers are everywhere.

You come into contact with them at home, work, the airport, the grocery store. You're using some type of computer to take this course. You know what? There's probably one in your pocket right now. While computers are complex and can seem daunting to learn, they ultimately just calculate, process and store data. In this lesson, we're going to take a peek at what's inside of a computer. We'll spend the next few lessons explaining how each of these components work. But for now, let's check out a typical desktop setup. Desktops are just computers that can fit on or under our desks. Here we have a monitor, a keyboard, a mouse, and a desktop. Sometimes we might even add a webcam, speakers or printer setup. We'll call these physical components hardware. Let's take a look at the back of the computer.

You can see common connectors here, the power outlet here, and the common ports here. Ports are connection points that we can connect devices to that extend the functionality of our computer. We'll go into detail about the ports you see here in a later lesson, but here's a quick rundown. We have a port here to connect to a monitor and a few ports here to plug your keyboard and mouse. There's another important one here for our network connection. With just these ports we're able to have the basic functionality to browse the web and much more. Things look pretty similar on a laptop. Here are some of the same ports, a built-in monitor, and a keyboard. There are also physical components inside the laptop case that are hidden for portability. Once you figure out how one computer works, you can figure out how any other computer works. This is my favorite part. Let's open up this desktop and take a deeper look. Let me first clean up my desk. Get ready for it.

It looks pretty complicated, but that's okay. We'll take you through it. Let's start with a quick tour. Then we'll dive deeper into each of these parts in the next lesson. Right here, this component is a CPU or central processing unit, which is covered by this heat sink. You can think of the CPU as the brain of our computer. The CPU does all the calculations and data processing. It communicates pretty heavily with this component right here, RAM or Random Access Memory. Ram is our computer's short-term memory. We use this component when we want to store data temporarily. Let's say you're typing something to a chat or a piece of text in a word processor, this information is stored in the RAM. Don't worry, we'll cram in more details on RAM in a later lesson. When we want to store anything in long-term memory, we use this component here, the hard drive. The hard drive holds all our data, which can include music, pictures, applications. Let me show you something else interesting. Have you noticed this large slab here? This is our motherboard. It holds everything in place and let's our components communicate with each other. It's the foundation of our computer. You can think of the motherboard as the body or secretory system of the computer that connects all the pieces together. The last component we'll talk about is our power supply, which converts the electricity from our wall outlet onto a format that our computer can use. You know what's interesting? All these components make up most computers, even a mobile phone. While it might look very different from your laptop, a mobile phone, just uses a smaller version of the hardware that we saw in the desktop and laptop today. Understanding how computer hardware works is a really helpful skill set in IT support. Since an IT department maintains the hardware that a company uses, a solid understanding of these computer internals will come in handy when troubleshooting hardware related problems and taking things apart to see how they work, it's just super fine.

Programs, the CPU, and Memory

Before we get our hands dirty with learning how to build a computer, let's talk theory first. In an earlier lesson, we talked about binary and how computers perform calculations. Remember that our computer can only communicate in binary using 1's and 0's. Our computers speak in machine language but we of course speak in human languages like English, Spanish, Mandarin, Hindi, you get the idea. If we want to communicate with our machines we have to have some sort of translation dictionary, just like if I wanted to say something in spanish I'd look it up in an English to Spanish dictionary. Well our computers have a built in translation book. In this lesson we'll dive deeper into how our computer translates the information we give it into instructions that it understands. RIght now you're probably using a web browser, music player, text editor or something else on your computer. We interact with these applications on a daily basis, they're referred to as programs. Programs are basically instructions that tell the computer what to do. We typically store programs on durable media like hard drives, you can think of programs like cooking recipes, we keep these recipes all stored together in a cookbook, just like apps stored in a hard drive. Now we want to make a ton of food, so we hire a chef to follow our recipes and whip up something good. The faster our chef works, the more food she'll prepare, the chef is our CPU she processes the recipes, we send her and makes the food, our chef works super fast so fast that she can cook faster than she can read. So we take copy of the recipes and put them into RAM. Remember that RAM is our computers short term memory, it stores information in a location our CPU can access it faster than they could with our hard drive. Now we can give our chef one or two recipes at a time instead of reciting the entire cookbook to her. Okay, now let's say I want to make a peanut butter and jelly sandwich. I see a pretty good recipe and send it to our chef to make, remember that our chef needs these instructions quickly so I don't send her the entire recipe, I sent her one line at a time. One, get two slices of bread, two, put peanut butter on one slice, three, put jelly on another slice, four, combine the two slices of bread. Now let me throw one more thing at you. Our chef can only communicate with us in 1's and 0's. So instead of sending something readable, like the recipe for peanut butter and jelly sandwich, we have to center something like this. In reality this process is a little more complicated. Our CPU is constantly taking instructions and executing them, these instructions are written in binary. But how do they travel around the computer? In our computer, we have something called the external data bus or EDB. It's nothing like a bus at all, it's a row of wires that interconnect the parts of our computer, kind of like the veins in our body. When you send a voltage to one of the wires, we say the state of the wire is on are represented by a 1, if there's no voltage then we say that the state is off represented by a 0. This is how we send around our 1's and 0's, sound familiar? In the last lesson, we talked about how transistors help us to send voltages. Now we know how our bits physically travel around the computer, the EDB comes in different sizes, a bit, 16-bit, 32, even 64. Can you imagine if you had 64 wires going you can move around a lot more data right now we're just going to stick with using an EDB with 8-bits in our examples, sending one byte at a time. Okay, so now our CPU is receiving a byte and it needs to get to work. Inside the CPU, there are components known as registers, they let us store the data that our CPU works with. If for example our CPU wanted to add two numbers, one number would be stored in a register A, another number will be stored in register B, the result of those two numbers will be stored in register C. Imagine the register is one of our chefs work tables, since she has a place to work, she can start to cook, to do so she uses a translation book to translate her binary into tasks that she can perform. Let's jump back for a second. Remember that our programs are copied into RAM for the CPU to read, RAM is memory that's randomly accessed allowing our CPU to read from any part of RAM as quickly as any other part. We don't actually send data from RAM over the EDB, there would be way too much stuff. RAM can hold millions even billions of rows of data. Despite our sandwich example, most of our recipes aren't simple at all. There can be thousands of lines long, we want to process them and we don't actually go in any particular order. Since we can only send one line of data through the EDB at a time we need the help of another component, the memory controller chip or MCC.

The MCC is a bridge between the CPU and the RAM. You can think of it like a nerve in your brain connecting to your memories, the CPU talks to the MCC and says hey I need the instructions for step number three of this recipe, the MCC finds instructions for step number three in RAM, grabs the data and sends it through the EDB. There's another bus that's nothing like a bus involved in the process called the address bus, it connects the CPU to the MCC and sends over the location of the data but not the data itself, then the MCC takes the address and looks for the data and then data is then sent over the EDB. Believe it or not, RAM isn't the fastest way we can get more data to our CPU for processing. The CPU also uses something known as cache.

Cache is smaller than RAM but it lets us store data that we use often and let's just quickly reference it. Think of RAM like a refrigerator full of food, it's easy to get into but it takes time to get something out, on the flip side of that, cache is like the stuff we have in our pockets, it's used to store recently or frequently accessed data. There are three different cash levels in a CPU, L1, L2 and L3. L1 is the smallest and fastest cache. So now we understand how our RAM interacts with our CPU but how does our CPU know when a set of instructions end and a new one begins? Our CPU has an internal clock that keeps its operations in sync. It connects to a special wire called a clock wire. When you send or receive data, it sends a voltage to that clock wire to let the CPU know it can start doing calculations. Think of our clock wires as the ticking of a clock, for every tick the CPU does one cycle of operations. When you send a voltage to the clock wire as referred to as a clock cycle, if you have lots of data you need to process in the command you need to run lots of clock cycles. Have you ever seen a CPU in the store and has something labeled 3.4 Ghz? This number refers to the clock speed of the CPU, which is the maximum number of clock cycles that it can handle in a certain time period. 3.40 GHz is 3.4 billion cycles per second, that's super fast. But just because it can run at this speed doesn't mean it does, it just means that it can't exceed this number, still, that number doesn't stop some people from trying. There's a way you can exceed the number of clock cycles on your CPU on almost any device, it's referred to as over clocking and it increases the rate of your CPU clock cycles in order to perform more tasks. This is commonly used to increase the performance in low end CPUs, let's say you're a gamer and you want to have better graphics and less lag while playing, you might want to over clock your CPU when you play the game. But there are cons to doing this like potentially overheating your CPU.

Supplemental Reading for CPUs

Supplemental Reading for CPUs

CPU cache and overclocking

In this reading, you will learn about the various levels of cache for central processing units (CPUs) and how a CPU processes and executes instructions. Additionally, you will learn about overclocking CPUs to maximize processing speeds. IT Support professionals may use this information when purchasing, allocating, and/or configuring high-performance servers.

Cache

You may already be familiar with the term “cache”. In computer jargon, cache (pronounced “cash”) refers to a small amount of recently used data that is stored either on hardware or in software. The first time data is accessed, both the initial request for the data and the reply containing the data pass through multiple points on their journey. Depending on several variables, these points might include I/O devices, motherboard busses, RAM, cables, hard drives, applications, networks, the internet, cloud platforms, and more. If a computer needed to use these full paths everytime it tried to access data, the entire transaction could take a relatively long time. Cache speeds up this process by holding a local copy of the most recently accessed data in temporary storage.

CPU cache

CPUs use a system of cache storage to help them quickly access data. A CPU cache is normally stored inside each core of the CPU. Older computers might store CPU cache in a transistor chip that is attached to the motherboard, along with a high-speed bus connecting the chip to the CPU.

CPU levels of cache

There are three levels of CPU cache memory:

- **Level 3 cache:** L3 cache is the largest and slowest of CPU cache. However, it is often twice as fast as RAM. L3 is the first CPU cache location to store data after it is transferred from RAM. L3 cache is often shared by all of the cores in a single CPU.

- **Level 2 cache:** L2 cache holds less data than L3 cache, but it has faster access speeds. L2 holds a copy of the most recently accessed data that is not currently in use by the CPU. Each CPU core normally has its own L2 cache.

- **Level 1 cache:** L1 cache is the fastest and smallest of the three CPU cache levels. L1 holds the data currently in use by the CPU. Each CPU core usually has its own L1 cache.

Overclocking a cpu

Overclocking a CPU sets it to run at a higher CPU clock frequency rate than the manufacturer's original specifications. For example, if a processor is labeled as having a 3.2 GHz base frequency rate, it may be possible to overclock the CPU to run at 3.5 GHz. Achieving a higher CPU clock frequency rate means the CPU can process a higher volume of instructions per nanosecond, resulting in faster performance. A computer user might want to overclock their CPU to improve sluggish speeds when performing processor-intensive tasks, like video editing or gaming.

Overclocking a CPU's frequency involves three variables:

- The base CPU clock frequency, often measured in GHz.
- The core frequency, which is calculated by multiplying the base frequency by the CPU core multipliers.
- The core voltage, which needs to be increased in small increments to meet the increasing power demand of the CPU during the overclocking process.

Warnings on overclocking

Overclocking the CPU can damage the computer if not configured properly. Operating a CPU at a higher speed can overheat the CPU and surrounding hardware, which can cause the computer system to fail. Additionally, overclocking the CPU can shorten the overall lifespan of the computer and void the computer's warranty. It is better to avoid overclocking the CPU and instead purchase the appropriate CPU speed necessary to meet computing demands.

How to overclock a CPU safely

As an IT Support professional, you may be asked to overclock a CPU. There are steps you should follow to do this as safely as possible. Always make sure that the requestor understands the risks of overclocking before agreeing to perform this procedure.

- **Check if overclocking is supported:** First, make sure the CPU is a model that is unlocked for overclocking. Not all CPUs can support overclocking, including most laptop CPUs. Check the CPU manufacturer's documentation to determine if overclocking is possible for the CPU model. Both Intel and AMD provide overclocking guides and tools for supported CPU models (see below for links to these guides). Additionally, check the documentation for the computer's motherboard model to ensure that it can support an overclocked CPU.

- **Clean the inside of the computer:** Turn off and unplug the computer. While wearing an anti-static wristband, open the computer and use compressed air to remove any dust build-up that has accumulated. It is especially important to remove any dust from around the CPU, fans, and intake vents.

- **Ensure an appropriate CPU cooler is installed (critical):** If the computer has a stock CPU cooler, it is most likely insufficient for cooling an overclocked CPU. Replace the stock CPU cooler with an advanced cooling system, like a liquid cooling system.

- **Follow the manufacturer's instructions for overclocking the CPU:** Using the detailed instructions from the manufacturer (see below for links to Intel and AMD's guides):

    - Use benchmarking software to establish a baseline for the normal performance of the computer.

    - Set each CPU core multiplier to the value of the lowest multiplier using either the manufacturer's overclocking software (recommended) or the BIOS. Then reboot the computer.

    - Increase each CPU core multiplier by 1 to increase the CPU frequency.

    - Test each increase for stability using the testing utility provided by the manufacturer.

    - Fix any problems flagged by the testing tools, especially temperature alerts. If the system becomes too unstable, roll back to the last frequency that produced a stable performance and stop overclocking the CPU.

    - If the voltage appears to become insufficient to support the new frequency, increase the voltage by 0.05V. Do not increase the voltage above 1.4V without specialized cooling hardware.

  - If the computer freezes or crashes, it has either become completely unstable or the CPU is not getting enough voltage to support the overclocked frequency. Use the BIOS to return to the last stable frequency or increase the voltage in 0.01V increments until stable.

  - If stable, reboot the computer before attempting the next increase.

Resources for more information

- **Intel:** Overclocking: Maximize Your Performance - Intel's all-inclusive guide to overclocking CPU, RAM, and motherboard. The site also provides utility tools for fine-tuning overclock performance and lists Intel CPU models that support overclocking.

- **AMD:** AMD Ryzen™ Master Utility for Overclocking Control - AMD's toolkit for overclocking Ryzen processors. Note that overclocking support for non-Ryzen models is no longer recommended by AMD.

- **AMD:** Ryzen™ Processor Overclocked Memory Compatibility List - List of AMD Ryzen CPU models that support overclocking.

- **AMD:** How to Overclock Your AMD Ryzen CPU - Instructions for overclocking AMD Ryzen CPUs from PC Magazine.

Joe: Diversity in IT

Changes in diversity with regards to IT support specialist is something I noticed for the last several years. There is a lot of stereotypes in the industry. But I think what was unexpected was how many people actually break that mold? The people I've met in the course of my IT support career have clearly shown that it's not just all male, it's people from all walks of life. That's one thing about IT support in general is that it's just so approachable for everyone. Diversity within the role has exploded. It's a much more diverse team now, both from the gender, the race, background as well as just educational background. I work closely with people from all different experiences all over the world, very different perspectives. It's great for the role, it's great for the company, but is also great to just work with different people. It's an incredible experience to share my experiences with team mate from Romania, team mate from Kenya, it's refreshing, it's fun. We're starting to finally lose those stereotypes associated with IT. We're starting to understand that technology is ubiquitous, everyone uses it, why can't everyone support it as well?
Help Us Translate