1. Moore's Law is a classic empirical observation derived from observation in the field of computer devices. By counting the number of crystals in a computer device, Moore found that the density of transistors doubles approximately every 2 years. This law has been proven to be true for 70 years, an exponential increase in transistor density also leads to an exponential increase in computer execution speed.

   As transistors get smaller and smaller, people find that Moore's Law is subject to many physical limitations, Nvidia CEO Jensen Huang also claimed that Moore's law was dead In September 2022

2. Power/Temperature problem: Increasing the density of the transistors, which also means increasing the amount of power the transistor needs to consume. Even tiny transistors use less energy, but density scaling is much faster. As a result, the heat of the transistor will accumulate, which is much greater than that of the low density in the early days. At present, the cheap and popular heat dissipation system is air-cooled heat dissipation, and the weak heat dissipation technology also limits the further improvement of transistor density.

3. Through the above description, we know that under the current mature heat dissipation technology, the energy consumption is constant (otherwise it is difficult for the system to dissipate heat to ensure the stability of the components), we can introduce the following formula

   $$P = \alpha \times CFV^2$$

   P stands for power and is equivalent to alpha (percent of time switching) times C (stands for size), F (stands for clock frequency), and V (stands for voltage swing). As we mentioned before, if the power is fixed, then we want the size and frequency to increase, the only variable we can adjust is the voltage.

4. Here comes out the Dennard Scaling law, a scaling law which states roughly that, as transistors get smaller, their power density stays constant, so that the power use stays in proportion with the area.

   But the voltage cannot be infinitely reduced, because, in any working environment, we all have noise. If the noise is in a certain range (we define it as 0.2-0.3), we define low voltage and high voltage as 0 and 1 respectively. Then even if the most extreme noise reaches 0.3, it will not reach the threshold of 0.5, and the low voltage will be mistaken for high voltage, resulting in a direct error in the binary system. But if we now compress the high voltage to 0.5, then the voltage threshold is 0.25, with the noise it is possible to mistake a low voltage for a high voltage and cause an error at the component level

5. Going back to the equation in 3., if we can't increase frequency infinitely, we can also increase processor cores and hide latency through concurrency. When a program (or thread) needs to wait for other data (e.g. network IO), the CPU is allowed to execute other tasks that do not need to wait during the waiting time. When other data is

obtained, the CPU is reacquired for execution, and every moment of the CPU can be fully utilized.