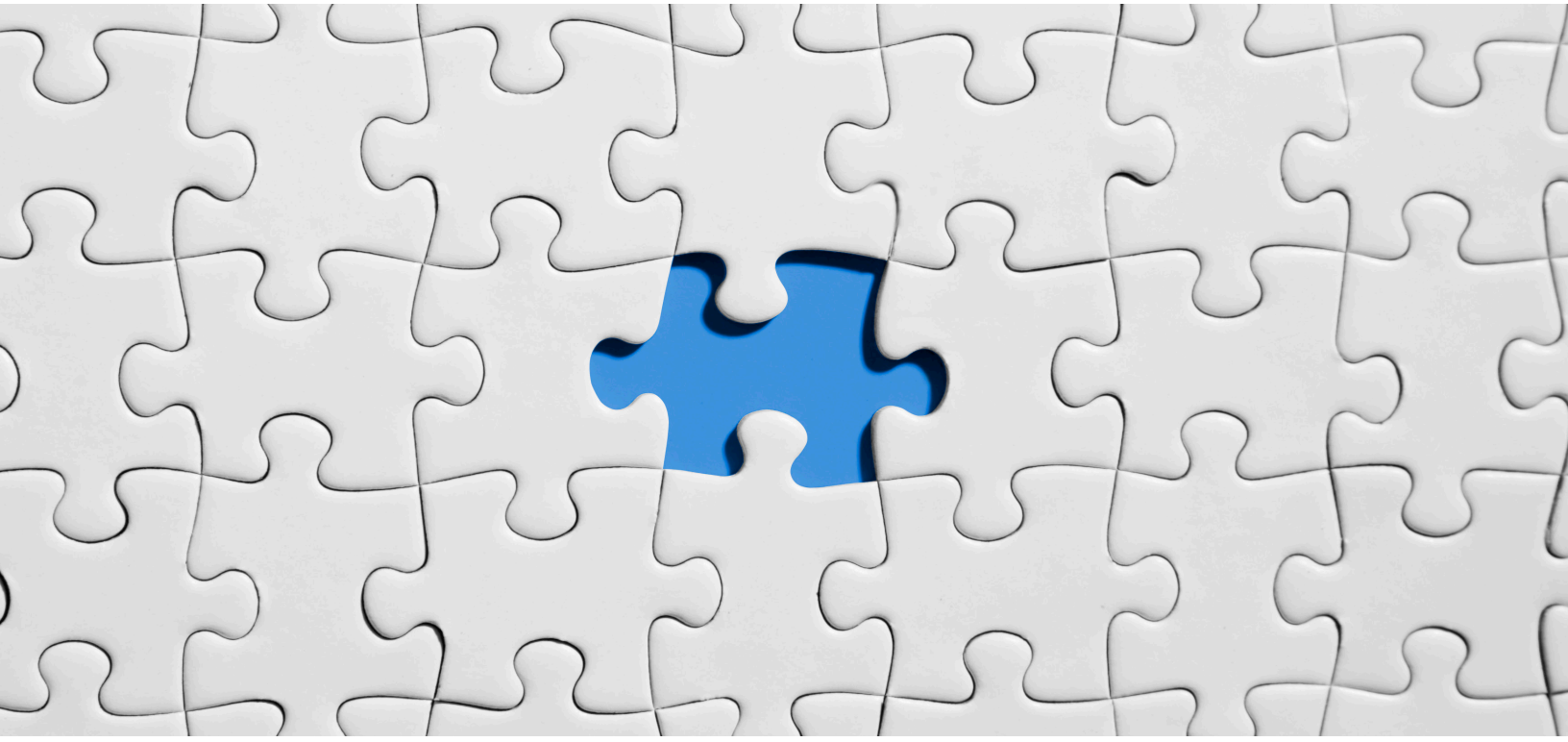
DELL Technologies

SAVING THE FUTURE OF MOORE'S LAW



Bruce Yellin

Bruceyellin@yahoo.com

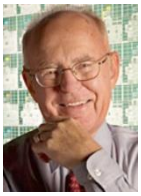
Knowledge Sharing Article

DELL Technologies Proven Professional

Table of Contents

Introduction

The transistor is one of the most amazing inventions of the last hundred years. It is the fundamental building block of our low-cost, high-powered computer chips that have transformed the social, economic, scientific, and technical aspects of today's world. Powerful and evolving new chips come to market with twice as many transistors or for a lower price than the previous year or two like clockwork – the essence of Moore's Law.



Gordon Moore's Law is not a law, such as Newton's Law of Physics, but a running commentary on miraculous miniature technology. It's been over 50 years since his original observations on the world of silicon semiconductors. In that time, the planet and to some extent the universe has evolved.

Mathematician John von Neumann began the transformation in the 1940s followed by Nobel Prize winners John Bardeen, Walter Brattain, and William Shockley with their invention of the transistor. Shockley, a Caltech alum, formed Shockley Semiconductor in 1955 and for his 18th employee hired Moore in 1956, a 27-year old Caltech chemistry doctorate.[1,2] In 1965, Moore predicted how many transistors, resistors, capacitors, and diodes could fit in a computer processor, and the rest is history.

The industry has shrunk the size of the transistor on the chip, making them faster and at a lower cost while remaining true to Moore's prediction. New engineering techniques have kept Moore's ideas going strong. Nonetheless, over the last decade, it has taken longer to shrink a component and overall costs have increased.

This paper explores the computing world from the end of World War II and traces the observations of one of the industry's true visionaries, Gordon Moore, over the last few decades. The central issue is whether Moore's original predictions still hold and if they do, will they continue for another five, ten or twenty years as technology continues to evolve. The paper reviews the engineering know-how that aided his predictions and future technologies that have yet to make it to the market.

John von Neumann

Rooted at the beginning of modern computing was a brilliant Hungarian mathematician, John von Neumann. He immigrated to the U.S. in 1930 and taught mathematics at Princeton University. By 1940, he was applying his

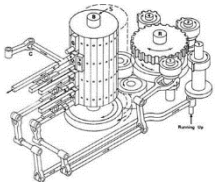
expertise in shaped charge mathematical modeling to the Manhattan Project. Von Neumann would go on to invent the computer architecture we use today.

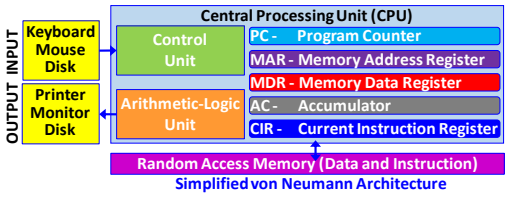
In 1946, the first electronic computer – ENIAC (Electronic Numerical Integrator and Computer) – came online. Built to calculate World War II artillery trajectory firing tables, von Neumann used the machine to help design the hydrogen bomb.[3]

ENIAC was a decimal machine (versus today's binary computers) that used 36 vacuum tubes to store a single digit. At 100 feet long and 30 tons, ENIAC needed 160 kWh of electricity to power 18,000 tubes, 70,000 resistors, and 10,000 capacitors.[4] It added 5,000 numbers or multiplied 357 two 10-digit numbers a second with a 200-microsecond clock. The machine was a "fixed program" computer that used switches and huge plugboards that took days or weeks to program.[5] ENIAC had card reader input and card punch output. In many respects, it was like a giant calculator – it did a good job adding and multiplying, but it couldn't do word processing.

Von Neumann helped design ENIAC's successor called EDVAC (Electronic Discrete Variable Automatic Computer). EDVAC was a binary machine that treated programs the same as data, storing them both in a single memory. It ran different programs, something difficult for ENIAC to do and was easier to debug.[6] Likely influenced by Alan Turing's universal machine, the "von Neumann architecture" is the basis for PCs, smartphones, and tablets in use today.[7] Modeled after human language, it follows the logical and linear sequential manner of how we speak. The machine had an arithmetic unit, a central processor to execute instructions, an instruction pointer (also called a program counter) that helps sequence the program flow, memory to store instructions and data, and input and output capabilities to bring information into the system and generate output reports.

Some of von Neumann's ideas leveraged Charles Babbage's 1830 pegged-cylinder and punched card "Analytical Engine."[8,9] Babbage defined memory, input/output (I/O), arithmetic units, and a decision mechanism based on computation results to solve computing problems.



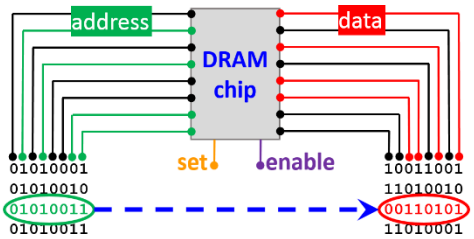


Simplified von Neumann Architecture

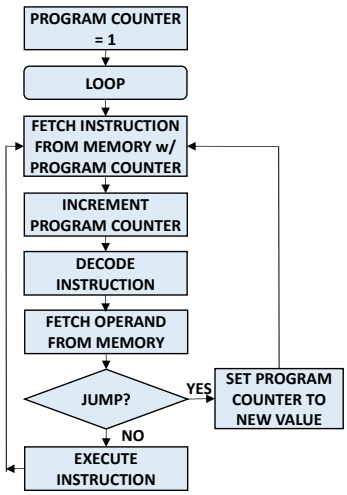
EDVAC made programming easier. As shown here, the **Central Processing Unit** (**CPU**) fetches an instruction from **Random Access Memory** (**RAM**), decodes and executes it, repeating the cycle based on the CPU's clock speed. ENIAC had a 100 KHz clock (100,000 hertz) in contrast to today's 3GHz PC clock (3,000,000,000 hertz) or three billion cycles a second.

A **Program Counter** (**PC**) has the address of the next instruction and registers temporarily hold data. The **Memory Address Register** (**MAR**) has the current data location while the **Memory Data Register** (**MDR**) holds data transferred to or from memory. The **Accumulator** (**AC**) holds calculation and logic results while the **Current Instruction Register** (**CIR**) tracks the instruction execution by the **Arithmetic and Logic Unit** (**ALU**). The **ALU** also handles memory retrieval, adding, subtracting, multiplying, and dividing a set of numbers, and Boolean AND, OR, and NOT operations. The **Control Unit** (**CU**) fetches and decodes instructions and coordinates **ALU**, memory, and keyboard or disk devices (input and output).

In this **Dynamic Random-Access Memory** (**DRAM**) chip illustration, an address maps to chip leads. A "**set**" command writes data while "**enable**" retrieves the **data or instruction stored** at a **memory location**. In this example, **memory address "01010011"** has **data** "**00110101**".

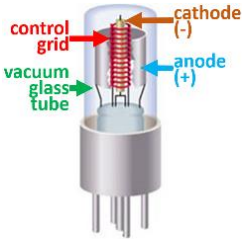


Here is a closer view of the entire cycle:



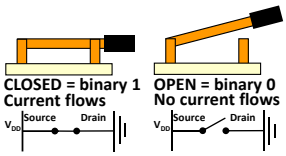
- Set Program Counter to first program instruction address.
- Fetch instruction from memory using PC address.
- Update the PC.
- Decode the instruction to get the number of operands.
- Fetch the operands (data) from memory.
- Test if the PC should be set to a new address.
- If not new address, execute the arithmetic or logic instruction.

Transistors

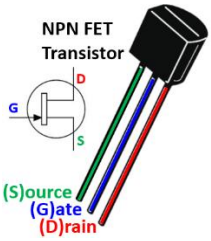


Invented in 1906 by Lee de Forest, ENIAC's triode vacuum tubes acted like on/off switches.[10] Tubes had a heated **cathode** (**-**) electron source, a **control grid**, and an **anode** (**+**) plate. A voltage applied to the **control grid** filament causes electrons to flow between **cathode** and **anode**, turning it ON (or OFF). They used a lot of power, got very hot, and burnt out relatively quickly.

In 1947, Shockley, Bardeen, and Brattain created a germanium transistor replacement for triode vacuum tubes used in the Bell Labs telephone network. Compared to tubes, transistors need less power, last longer, and are smaller. Their new semiconductor was both a conductor with a positive current or an insulator when current is absent.



In a computer, a digital Negative-Positive-Negative (NPN) Field Effect Transistor (FET) acts as an electrical relay or knife switch to turn a circuit ON (binary "1") if closed or OFF (binary "0") if opened. Millions of ON/OFF transistors store binary instructions and data. The NPN FET to the right uses a semiconductor such as silicon (Si) because of its superior switching speed.[11] It is a "1" when the **gate** gets a voltage, causing electrons to flow from **source** (current in) to **drain** (current out) electrodes. Without a **gate** voltage, it is OFF. CPUs use transistors that individually turn ON or OFF to create Boolean logic gates, the basis of a computer.

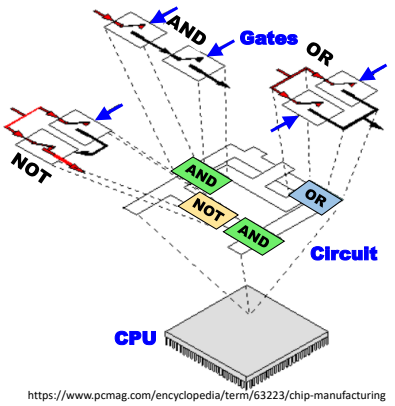


Boolean NOT	
X	Y
0	1
1	0

Boolean AND		
X	Y	Value
0	0	0
0	1	0
1	0	0
1	1	1

Boolean OR		
X	Y	Value
0	0	0
0	1	1
1	0	1
1	1	1

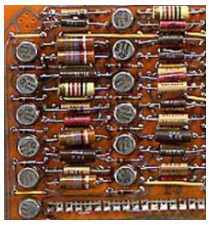
To the left are some Boolean logic "truth table" gates commonly created by combining NPN transistors in a defined way. For example, a transistor **NOT** gate with no voltage is a logical "1" or ON. On the other hand, if it gets +5V, the output value is "0" or OFF. Complex circuits are created by combining gates and wind up in a processor – the illustration to the right represents those integrated circuit logic gates.[12]



https://www.pcmag.com/encyclopedia/term/63223/chip-manufacturing

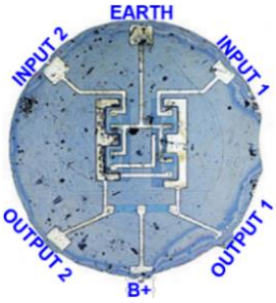
Integrated Circuits

Through the late 1950s, computer circuits contained discrete transistors, resistors, diodes, and capacitors inserted on a circuit board and soldered by hand. Transistorized computers of that era might have had a thousand 1" tall transistors.[13]



Computers with individual transistors were relatively large and power-hungry, and connecting them created a wiring mess.[14] In 1959, Fairchild Semiconductor's Robert Noyce and Shockley, transistor co-creator, changed the computer world. They improved upon Jack Kilby's Texas Instrument design with a silicon integrated circuit (IC) – a single device with a set of small

transistorized circuits.[15,16] To the right is the first silicon resistor-transistor IC. Used in the Apollo spacecraft's guidance computer, it had four transistors in the center and "white line" metal traces for interconnections.[17] The original IC is round compared to today's rectangular design to accommodate a transistor's can and leads package shape like this TO-5 on the left.

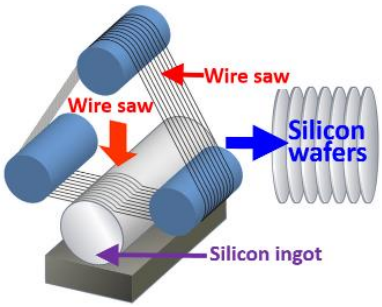


By 1965, Fairchild had produced an IC with 64 components.[18] That year, their research and development director, Gordon Moore, noted the number of IC components doubled every year – a fact he would shortly write about, cementing his name into computer science history.[19]

Year	Processor	# transistors	nm Proc
1971	4004	2,300	10,000
1972	8008	3,500	10,000
1974	8080	4,500	6,000
1976	8085	6,500	3,000
1978	8086	29,000	3,000
	⋮		
2000	Pentium III	21,000,000	180
	⋮		
2016	E5-2600 v4	7,200,000,000	14

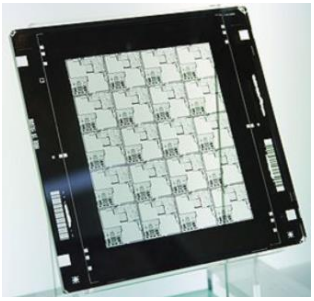
en.wikipedia.org/wiki/Transistor_count

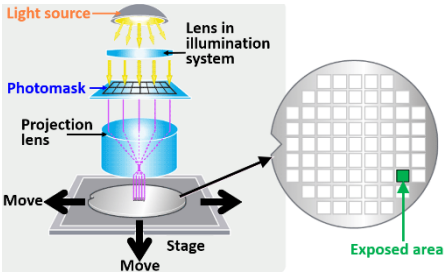
Moore witnessed techniques improve as components kept shrinking. Each rendition had greater complexity resulting in higher development costs, yet IC prices continued to drop. In this abbreviated Intel processor chart, the 8080 had 4,500 transistors built from a 6000 nanometer (nm) design process in 1974. By 2000, a chip could have 21 million transistors, and in 2016, Intel's high-end 22-core E5-2600 v4 had over 7 billion transistors using a 14nm process.[20] A nanometer is one billionth of a meter or 0.00000004 of an inch and the size your fingernail grows in one second. As a reference, the period on this sentence is large enough to contain 6 million transistors.[21]



Chip-making uses silicon, which makes up 28% of the earth's crust and found in sand.[22] Purified silicon, heated to a molten state, is electrically altered with other materials. To the left, a 300-600mm (1-2 foot) long and 200-300mm (8-12 inch) diameter **silicon ingot** is **sawed** into thin, round semiconducting discs 0.1 inches thick called **wafers**.

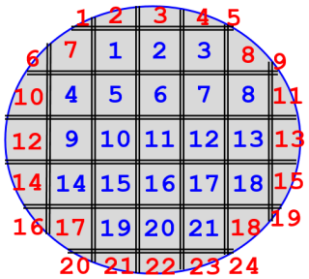
Engineers construct an IC using software libraries of logic gate functions to produce a GDSII (Graphic Data System) or OASIS (Open Artwork System Interchange Standard) design. The circuit design of transistors, resistors, and diodes transfer to a lithographic glass plate as an opaque "photomask" template shown to the right.[23] Each photomask is a single layer of a multi-layer circuit design and the size of a chip.[24] A chip is one physical package of circuits.

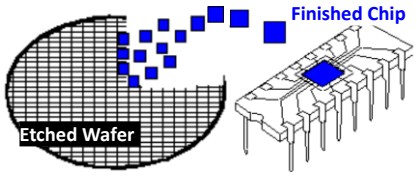




A wafer gets a silicon dioxide coating to enhance its electrical insulation properties. **Light** shines through a **photomask** to etch the **exposed silicon**. The process of building a populated multi-layer wafer with microscopic circuits requires robotic control.

In 1965, wafers were one inch (25mm) in diameter. Over time, 300mm wafers were commonplace with 450mm wafers due out this year.[25] A wafer's diameter is selected based on IC size and waste. A die is a rectangular shaped device sliced from a wafer. Rectangles don't fit well on a circular wafer and in this illustration **21** of them **fit** while **24** are **incomplete**. Yield is the number of viable dies produced after discarding **incomplete** and non-functional dies. A 300mm wafer yields 148 20mm dies.[26] In general, smaller die or larger wafers have the highest yield and lowest waste.





Each tested and certified die mounts in a chip which connects the IC leads to pins in a range of insertable shapes such as the one shown to the left.[27] Chips are plugged into motherboard sockets or soldered onto circuit boards.

Enter Gordon Moore

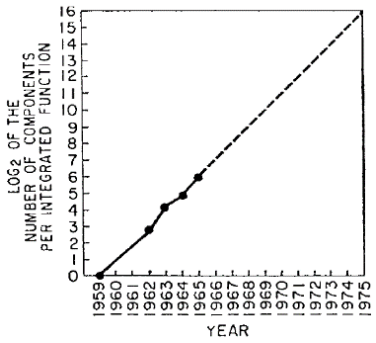
Dr. Gordon Moore was born in 1929.[28] With a year of Shockley Semiconductor experience, he became one of "the Traitorous Eight" who left Shockley in 1957 to co-found Fairchild Semiconductor.[29] Their first product was a high-performance silicon transistor designed by Moore. Measuring 1/4" wide by 1/4" high, they sold 100 of them for

$15,000 ($150 each) to IBM to help build the XB-70 Valkyrie bomber.[30,31] He left Fairchild in 1968 with six others to start Intel Corporation, which is short for "integrated electronics."

What Exactly Is Moore's Law?

While Moore was Fairchild's Research and Development Director, Electronics magazine asked him to predict the future of the semiconductor industry.[32] In 1965, he described the miniaturization, clock frequency, and heat generation of transistors

to date and predicted their ten-year growth based on cost and count variables. He observed that from the first silicon IC in 1959 to the then current 1965, the number of components in integrated circuits doubled every year. His article “Cramming more components onto integrated circuits” included a y-axis logarithmic chart that is among history's greatest predictions.[33] He didn’t call it “Moore’s Law” but he outlined a semiconductor roadmap with just five data points:



1959 - first planar transistor ($2^{\wedge 0}$),
1962 - 8 transistor chip ($2^{\wedge 3}$),
1963 - 16 transistor chip ($2^{\wedge 4}$),
1964 - 32 transistor chip ($2^{\wedge 5}$), and
1965 - 64 transistor chip ($2^{\wedge 6}$).[34]

(When a quantity uniformly doubles every *n* months or years it has exponential growth, and its logarithmic graph is a straight line.) His Law explained the engineering economics, trends, and innovation relationship. Moore felt that the reliability, price, yield, and heat hurdles facing future IC development were solvable.

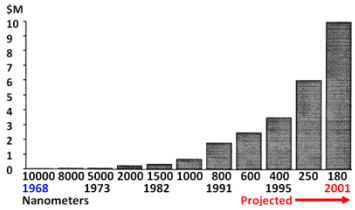
Some oversimplified his article to say the IC transistor and resistor counts double each year. “The complexity for minimum component costs has increased at a rate of roughly a factor of two per year. Certainly, over the short term, this rate can be expected to continue, if not to increase. Over the longer term, the rate of increase is a bit more uncertain, although there is no reason to believe it will not remain nearly constant for at least ten years.” In other words, insane computing power at ever lowering prices. CalTech professor Carver Mead in 1970 coined the phrase “Moore’s Law,” describing Moore’s efforts to describe transistor costs and their impact.[35]

The 1971 Intel 4004 on the left had 192 transistors per square millimeter and equaled the performance of the 18,000 vacuum tube ENIAC. In 2016 Intel’s 14nm E5-2600 v4 processor had 15+ million transistors per sq mm. Wafers for these larger dies grew 36 fold, chips got 3,500 times faster, transistor energy efficiency improved 90,000 times, and transistor cost dropped 60,000 times.[36] Exponential improvements influenced by chip density, engineering cleverness, connection dimensions, and innovation led to reduced transistor cost and added features. It was vital to cut the power profile of small transistors operating at high speed. As clock frequency increased and surface area decreased, the clever use of multiple cores allowed for slower clocking while increasing the total number of instructions per second.

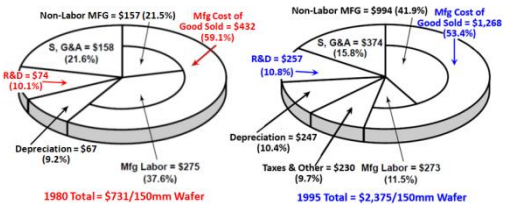
Moore forecasted a quarter-inch semiconductor would hold 65,000 components by 1975 and the density trend would continue at “[…] roughly a factor of two per year.”[37] He included

visionary statements such as “Integrated circuits will lead to such wonders as home computers—or at least terminals connected to a central computer—automatic controls for automobiles, and personal portable communications equipment.”

By 1975, computer-aided design (CAD) was helping build device and circuit cleverness into chips. However, Moore still felt semiconductor complexity was problematic, and his 1965 annual double density growth calculations were optimistic. He revised them to double every two years.[38,39] His prediction, backed by a stable transistor density relationship and straightness of the logarithmic line, has remained true for over the last fifty years. As referenced in an earlier Intel processor chart, the number of transistors in an Intel chip grew at a staggering rate. Chip prices have remained relatively constant, transistor prices steadily dropped, and chip feature sizes decreased to near atomic levels.

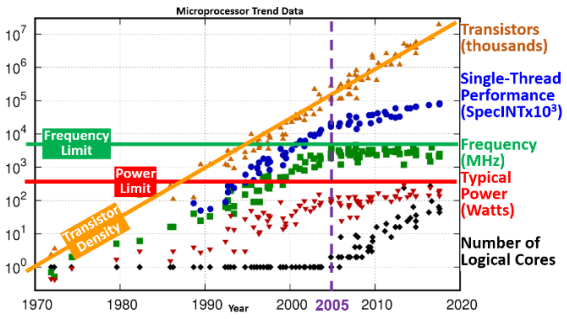


Several Laws attributed to Gordon Moore discussed the exponential nature of transistor density, size and clock speed. In 1995, Moore published his third “Law” paper which in part said the biggest threat to his projection was not miniaturization hurdles, believing the engineering problem could be solved, but the financial aspect of that progression.[40] He noted a piece of equipment cost $12,000 in **1968** and $12 million by **2001** – a trend of increased costs threatened innovation.



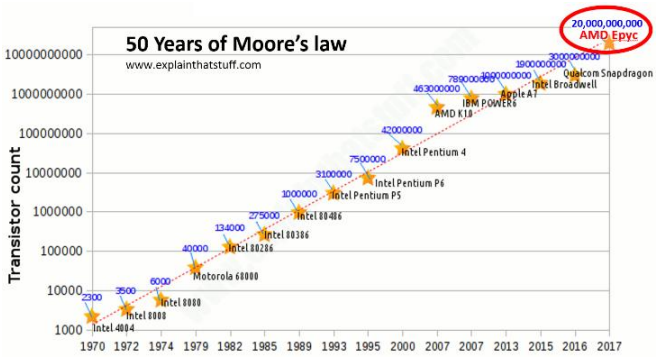
A 1997 Integrated Circuit Engineering Corporation report supports Moore’s conclusions.[41] The cost of 150mm wafers tripled from **$731** in **1980** to **$2,375** by **1995**, due mostly to increases in research and development (R&D) and manufacturing costs of goods sold. Non-R&D labor expense showed little change during that period. Moore encouraged engineers to innovate with an eye towards affordability. His Laws evolved as data became available.

This plot shows the exponential growth in **transistor density**, and the status of **performance**, **clock speed**, **power consumption** and **logical core count** from 1971 to today.[42] You can see Moore’s **transistor** prediction still holds, even as **operating frequency** and **power** limits were reached around **2005** when it became impractical to cool ultra-dense

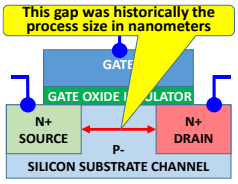


power-hungry silicon ICs running faster than 5GHz. Chip density should increase with the next generation of 7-10nm integrated circuits. In general, more transistors on a chip means more parallel operations per second. With annual **processing power** likely increasing, though not exponentially, and **power consumption** (heat) staying relatively constant, additional **transistors** translates into higher **performance** through mainly more **cores** and larger caches.

Advanced Micro Devices (**AMD**) was founded in 1969 by Jerry Sanders. Sanders left Fairchild Semiconductor a year after Moore departed.[43] In 2017, **AMD** introduced the **EPYC** processor with 19.2 billion transistors as shown here. The industry was maintaining the Law's exponential density and performance even as this plot slightly and occasionally deviates from a straight line.[44]

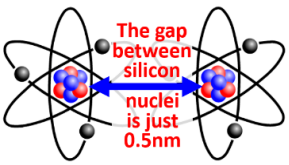


Exponential growth sounds simple, but anything repetitively doubling can be unattainable. For example, in David Birch's story "The King's Chessboard," a King rewards a wise man for his service to the kingdom.[45] The man requests a grain of rice for the first square of the King's chessboard, with double the amount in the next square on the second day, continuing for all the squares. The King agrees, but within weeks, the kingdom is out of rice – it does not have nine quintillion grains to put on the 64th square. In the real world, it is also impossible to build a device every two years with an exponentially growing number of transistors without reaching a physical or financial end. Over the past 50+ years, engineers have delayed the end of Moore's projection. Moore said "The nature of exponentials is that you push them out and eventually disaster happens."[46] Complexity and cost may cause the Law's next revision – perhaps density will double every three years. Still an amazing feat.



Historically, nanometer process size was the source and drain gap, and the distance electrons travel to complete a circuit – a critical part of the Law. Sometime after the 20nm process, the dimension became a chip generation reference and defined the tiniest feature span.[47,48]

Over the last decade, a 34nm (~68 silicon atoms) process shrunk to 7nm (~14 silicon atoms). Smaller gaps mean more transistors on a chip, increased processing power, or more chips on a wafer at lower prices. As shown here, a **7nm** separation equals a 0.5nm silicon nuclei distance (about 4X smaller than



a strand of human DNA), and 14 atoms can't carry enough current.[49,50]

Moore's Law depends on the size of a silicon atom. Silicon transistors cannot exist below 1nm since the source and drain gap would only be two atoms. Combining germanium with silicon allows electrons to have added mobility and permits faster current flow, extending the Law.

The von Neumann Controversy

As revolutionary as von Neumann's architecture was, it's reliance on a single, relatively small path separating the CPU's control unit and main memory forces instruction and execution cycles to alternate. Processor speed and memory density increased yet lagging transfer rates created a bottleneck. His design forced the CPU to have idle periods as it waits on memory retrieval. By 2010, CPUs operated sub-optimally, "starved" for data as they executed instructions 100 times faster than retrieving items from memory.[51]

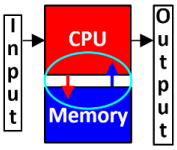
Bottleneck - the part of a bottle that slows the flow of liquid.

Chip makers guided by Moore's vision and commitment to von Neumann compatible software found ways to cut the time CPUs waited for memory transfers.

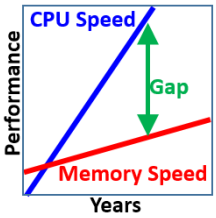
Von Neumann Memory Bottlenecks

The von Neumann design has three main bottlenecks that all involve memory access:

1. Combining instructions and data in the same memory requires the processor to get its next instruction from memory. The request traverses the memory bus (motherboard wires) with data latency traffic jams and slow retrieval compared to CPU speed. Regardless of the CPU horsepower, memory bus access speed limits every request. It takes longer to fetch an instruction than it does to execute it, and when a data-hungry program demands data, the memory bus delays the processor.[52]



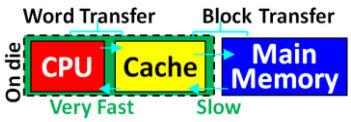
2. A CPU handling large data volumes have relatively few registers to store it. For instance, to spell check a document, a program compares each word to a memory-resident dictionary. It can cause an I/O bottleneck if the processor is idle waiting on memory. A broader memory bus decreases traffic but can increase the die package and cause electrical crosstalk.[53]
3. While the amount of physical memory has increased, its **speed** relative to **CPU speed** has created a **gap**. Often, adding memory doesn't increase processing throughput. Instruction and data memory retrieval speed can throttle the processor, and this gap has grown.



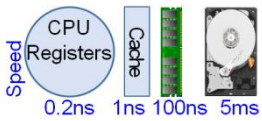
Von Neumann Memory Fixes

Here are some of the many solutions that engineers created to address those bottlenecks:

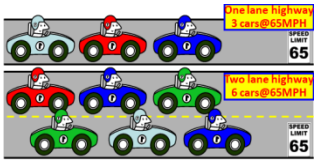
A. Cache – Assume your 3GHz CPU has a 400MHz memory bus. When it needs a memory-resident instruction or data, it waits a relatively long time to get it. A fast, dedicated, small cache (from the French Canadian trapper slang, “hiding place for stores and provisions”[54]) memory was added to von Neumann’s design to improve memory access speed and increase processor performance. Algorithms determine which instructions and data should reside in the cache. Programmer/program invisible, the CPU retrieves what it needs from within its die instead of traversing the memory bus to main memory. A "cache hit" occurs when the CPU finds what it needs in the cache. “Cache misses” takes longer because of “miss” time, memory access time, housekeeping, and the new value put into the cache for future use.



Caches can reside on and off-die, with some dedicated for instructions or memory. CPU models can have different size caches and locations, and a CPU core can have a small Level L1 cache while sharing a large Level L2 cache amongst other cores. Integrated cache, accessed at clock speed, is 100X faster than retrieval from memory outside the CPU and a million to five million times faster than from a 1ms solid-state drive or a 5ms hard drive.[55]Cache is analogous to borrowing a library book for a research project. Assume your desk holds three books. You borrow three books and refer to them as you work through the assignment. If you need additional books, you return some of them to make room for new titles.

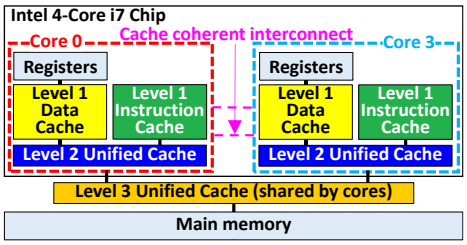


B. Multicore – Another way to bypass the bottleneck is to shrink the CPU size and create another instance of it – basically, two integrated parallel processing cores in a single chip package. Each core has dedicated registers and cache, and they share memory and a large unified cache. Intel’s 2007 dual-core Pentium CPU ran at near uniprocessor’s clock speed, but two cores with twice the componentry performed near twice the work.[56] Analogous to a 65 MPH single versus a double-lane road, the double-lane carries twice the cars while at the same speed.



Some programs or tasks don't benefit from parallelization. For example, a file “open” and “close” runs at the same speed on single or multicore processors. Logic such as C=A+B can't run in parallel because it would be disastrous for another processor to change B until the calculation takes place. Further, it is difficult to write a parallel program.

Personal computers might have 2-4 cores, while Intel and AMD-based servers have up to 32 cores on a single chip.[57] Can a processor with 1,000 cores sustain Moore's Law? Core limits can depend on how many tasks need to run, the ability to break tasks into smaller chunks, or main memory throughput constraints.



In this Intel Core i7 diagram, each core has independent L1 **data** and **instruction** caches, and a shared L2 data and instruction **unified cache**.[58] A **cache coherent interconnect** ensures an update is available to all cores simultaneously. Cores share an L3 **unified cache** that connects to main memory over a memory bus.

Multicore chips multitask by alternately running tasks with the appearance of concurrency. As a result, your computer runs a word processor and plays music on one core while a second core handles the internet browser and downloads data.

C. Pipelining – Optimizes processor activity keeping it busy while stalled tasks wait on the memory bus. The CPU fetches instructions from the same thread ("The smallest sequence of programmed instructions that can be managed independently by a scheduler"[59]) while executing non-memory instructions such as add or subtract. It is like an auto assembly line where wheel installation is independent of windshield installation.

   Processors predict with 90% certainty whether an instruction causes a jump, allowing the CPU to process multiple instructions simultaneously and stay occupied. When a branch misprediction occurs, execution speed slows as the CPU discards incomplete instructions and the pipeline restarts from the point of the new branch instruction program counter.

   Conditional and unconditional "branch instructions" can reset the program counter. A conditional branch, such as "**if then else**" or an unconditional branch like "**go to**" can change the value. Branches can slow a program on a pipelined processor. Studies found that 20-30% of all instructions involve branching, with 65% of them followed.[60] Intel's Pentium Pro processor introduced branch prediction in 1995.[61]

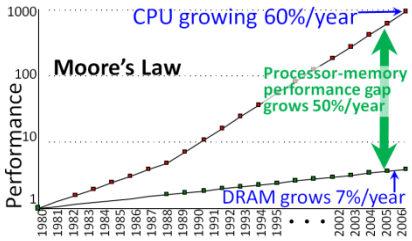
D. Multithreading (Intel's Hyper-Threading) – A thread "blocks" when waiting on cache or memory, letting the operating system run another task's thread such as anti-virus checking, a data download, web browser activity, a word processor displaying text, etc.. Tasks broken

into threads have dedicated program counters and memory blocks that include the section of executable code, a stack that tracks functions, security, and other data.

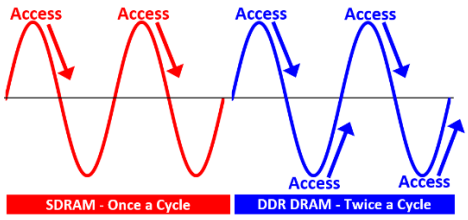
Modern operating systems multitask threads, meaning they support the apparent simultaneous parallel execution of multiple programs. Ready to run threads reduce stalled workloads close to 30% or exacerbate the problem when those threads all need memory resources.[62] Multithreaded applications can run on different cores.

E. Various random access memory – Memory such as this Dual In-line 

Memory Module (DIMM) uses DRAM chips that pair a transistor with a capacitor to store one bit of information. New types of memory were introduced to decrease latency and increase bandwidth as processor performance followed the Law. By 2003, the **performance gap reached 50%** as faster CPUs waited longer for memory access as shown to the left.[63] Intel processors from 1987-1995 used 32-bit Fast Page Module DRAM (FPM DRAM) which performed four consecutive reads using a single address.[64] In 1996, **Synchronous DRAM** (**SDRAM**) leveraged the system clock and handled 64 bits of data versus 32 bits.



In 2002, **DDR SDRAM** (Double Data Rate SDRAM) was twice the speed of **SDRAM**, using the clock's rising and falling edges to double output as shown here. Memory evolved over the years with DDR2, DDR3, and DDR4 SDRAM, each one reducing the bottleneck.

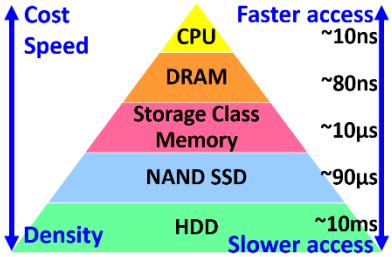


F. Dual Processor – Unlike multicore architectures that integrate all the cores into one chip with shared memory, dual-processor designs use two chips and independent memory all on the same motherboard. They could perform double the work of a single processor in the same amount of time. Dual processor chipsets also had multiple cores to speed up processing.

Invention, Ingenuity and Innovation

The Law's prophecy of greater miniaturization at lower cost continued at a predictable pace. As with the von Neumann bottlenecks, engineers sped up chips, reduced the per transistor price and cut heat profiles. Here are some of the ways they applied invention, ingenuity, and innovation towards that goal.

Storage-Class Memory

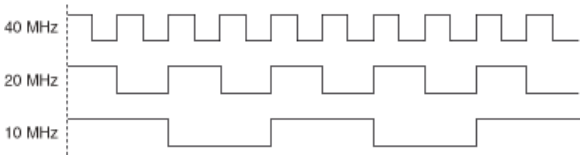


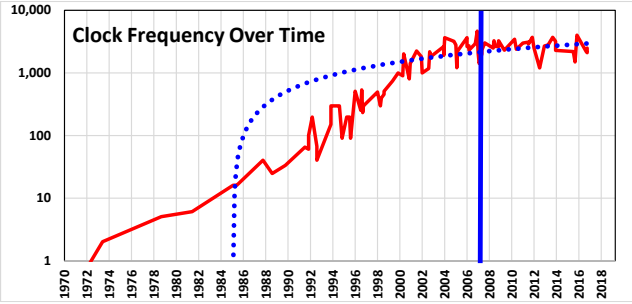
Regardless of processor speed, if a chip's instructions aren't in its registers, they load from slower subsystems. This chart shows some of the innovation that allows Moore's Law to continue. A CPU processes an instruction in under 10ns. Memory access adds an 80ns delay or the time it takes to process eight instructions.[65] Intel "Optane" storage class memory resides between DRAM and an SSD or HDD disk.[66] It is a little slower than main memory but it is persistent like an SSD or HDD. Rather than wait for 90µs+ for an SSD or 10ms+ for an HDD to fetch data and instructions, Optane memory responds in 10µs – 10X faster than the fastest SSD.[67]

Increasing Clock Speed

As we discussed earlier, accessing off-chip memory impacts a computer's processing power. Key architectural issues such as instruction set richness, caching, execution units, pipelining, and branch prediction impact the workload a system can accomplish. Most people focus on the processor's clock speed.

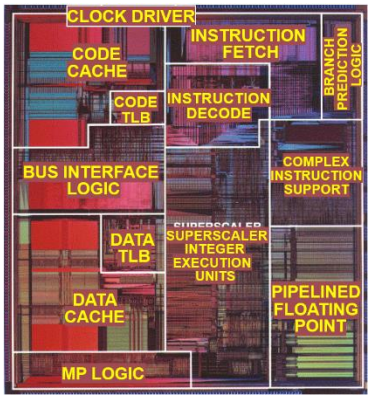
Your personal computer's processor has a measurement of clock speed in gigahertz (GHz or 1,000,000,000Hz) such as 2.4GHz. That translates into a system clock of 2.4 billion ticks or pulses per second. It was the speed a CPU could perform operations, like adding two numbers. A faster clock executes more instructions every second than a slower one. It is analogous to a conductor who uses their baton to increase or decrease the tempo of a composer's sheet music. Here is a clock cycle comparison of 10, 20 and 40MHz. One hertz is one full cycle in one second. Clock speed is not unlimited.





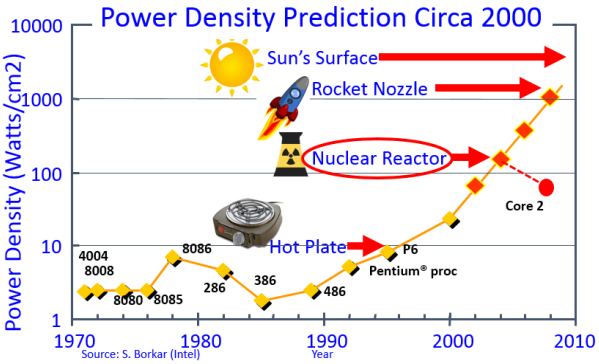
Electromagnetic wave speed (not actual electrons) governs processors and is set by the system clock.[68] This "Clock Frequency Over Time" logarithmic trend chart shows clock frequency leveled in 2007 at 4.6GHz.[69] Wave speed is dependent on IC wire gauge

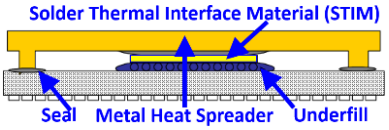
and its material composition. It can travel at 50-100% the speed of light (186,000 mi/s) or 6 to 12 inches in 1ns. A 4GHz CPU's clock pulses every 0.25ns, so electricity travels 1.5 - 3 inches (3.8 - 7.6cm) per pulse.



A CPU die has many logic sections as this map shows. The clock driver at the top sends a synchronized clock pulse to every area.[70] Precise circuit lengths and routes avoid pulse propagation delays (clock skew) at higher clock frequencies or greater transistor density. Engineers ensure a short, simple path receives a pulse at the same time as a long, intricate path.

Increasing the clock speed poses a thermodynamic challenge making it hard to cool dense, flat silicon ICs using a fan. In 2000, Intel's Pentium 4 had a power density of 46 watts per square centimeter (cm^2), or 7X the power density of the old 486.[71] Shrinking the feature size used to mean increased chip performance, such as when the Pentium was faster than the 486, the Pentium II was faster than the Pentium, and so on, but by 2003-2005, clock speed increases came to an end. Engineers were unable to cool the estimated 200W/cm2 of power (twice that of a **reactor**) and were forced to limit the clock speed. They focused on density by adding transistors, larger caches, and multi-core parallelism, allowing a chip to execute more instructions per clock cycle. Hot machines running above perhaps a 4GHz clock speed need liquid cooling similar your car's engine radiator, while four 3GHz cores run fine. Intel's 2018 14nm Core i9-9900K used a solder-based thermal interface material (STIM), a metal heat spreader on top of the die, and a liquid cooling fan to help dissipate heat and allow overclocking to 5GHz.[72]





Increased parallelism meant more calculations completed every clock pulse and lower power consumption allowing for longer battery life in portable devices. Processing 64 data bits at a time is more efficient than older 8, 16 or 32 bit processors. Smarter algorithms solved problems faster than inefficient ones.

As chips get denser and perhaps adopt 3D techniques to limit the distance electrons need to travel, chip cooling likely necessitates a cooling fluid for heat dissipation.[73]

Parallelism and Cache Coherency

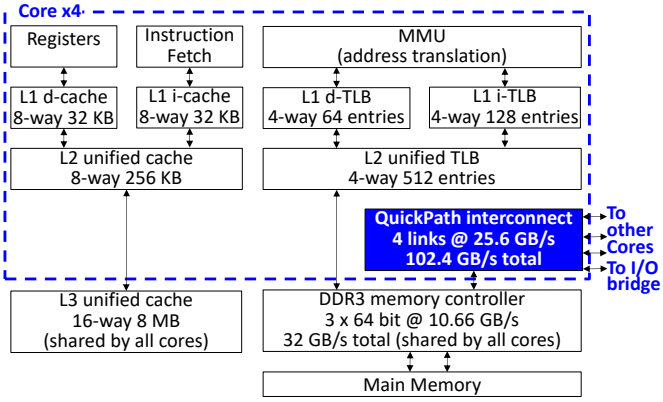
A major architectural shift occurred three decades after Intel solved a Nippon Calculating Machine Corporation chip design problem with their 4004. Engineers realized that maintaining the von Neumann single instruction, single data stream (SISD) approach meant that ever smaller transistors would run into a physics obstacle or be expensive.[74] They began exploring ways to execute instructions in parallel and increase the number or size of CPU-memory paths.

Parallelism is one way to increase overall system performance with many ways to achieve it. Rather than shrinking components and making them power efficient, the premise was to find higher-level solutions. Using a design called multiple instruction, multiple data (MIMD), each processor or core can use an independent set of instructions on different sets of data. Your data center server running a database or other workload uses MIMD such that in a single clock cycle, tasks split into multiple sub-tasks are autonomously run in parallel. This brought products with:

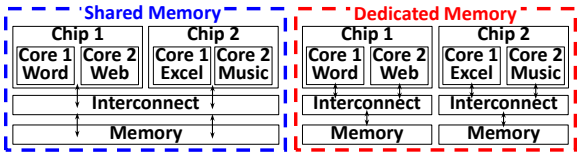
1. Multiple physical CPUs in a single machine that share linked caches and shared or individual system memory.
2. Multiple CPU cores on a single chip sharing cache and system memory.
3. Multiple processing threads that allow the CPU to switch to an alternate instruction sequence when the current one waits for memory access.
4. Parallel instructions that run on other processors in the system or grid. Results are "assembled" from all of them.

Multiple processors and cores presented new engineering hurdles. A cache coherence problem arose when one CPU updated its private cache without notifying other CPUs of those changes. For example, if a program on core #1 changes “X” from “1” to “2” without communicating it to other cores, and core #2 reads “X” from its cache, it is still equal to “1”. It was necessary to actively keep caches synchronized and have memory behave as a single shared memory.

At a high level, a cache has three states. **Invalid** means the cache lacks the requested value (cache miss) or the value is wrong (stale). It must be retrieved from memory or a valid cache. A **shared** status means a value is reusable such as a state abbreviation that never changes (e.g. Ohio is always “OH”). When a shared value changes, it becomes invalid. The **modified** state means no other cache is using a value, so it is usable if it remains “local” to that processor.



Intel developed the QuickPath processor interconnect in 2008 to maintain cache coherency through techniques such as source snooping. Snooping permits multiple caches to trigger an invalid state when a monitored memory address changes. In this Intel Core i7, QuickPath high-speed point-to-point links coordinate with other cores and the I/O path to main memory.[75] Other processor protocols use "directory-based coherence" to maintain shared data in a common directory and "snarfing" which allows cache controllers to update their memory location when contents change.



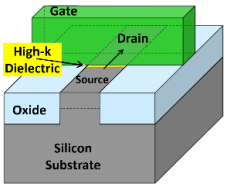
Caches help systems that wait on memory and is essential for multiple processor chips. Cores **share memory** in the design on the far left.[76] A cache "unfriendly" core could inundate a single memory path, which would cause other cores to wait for the jam to clear. **Dedicated memory** on the right gives each chip shareable **memory** using a Non-Uniform Memory Access (NUMA) technique.[77] One chip's core could access another core's memory, keeping in mind that access time depends on whether the information is in the memory of the local or remote chip.

Three-dimensional Integrated Circuits – Taller, Not Smaller

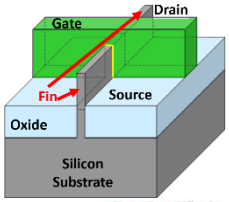
Modern IC design uses two-dimensional (2D) chips to build computers. The height of an IC was tiny and needed to accommodate the physical world of packaging and other constraints.

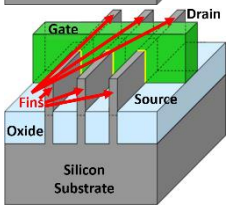
While miniaturization gives us faster, cheaper, denser, and low power components, ultra-dense 2D chips create a host of problems. Shrinking a transistor increases its electrical resistance and "leakage" since opposing surfaces have reduced contact area, and leaking chips need additional power, which adds heat. Transistor gate oxide leakage, called quantum tunneling, occurs when the source and drain gap is so small that electrons pass through the thin barrier instead of staying at their logic gate, preventing it from turning OFF. Heisenberg's uncertainty principle says ultra-tiny transistors could permit wires to leak electrons.[78] If you can't control electron flow, you can't reliably tell a "0" from a "1", and you no longer have a viable switch. Gate thickness cannot shrink below one nanometer.[79] As IC surface area shrinks, it is harder to dissipate the increased heat, so it requires large heat sinks, fans, and/or liquid cooling.

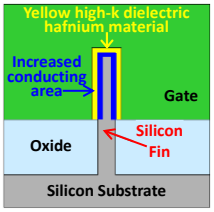
In 2003, Intel used a 90nm “strained silicon” process to fight the resistance and leakage problems to extend the life of Moore’s Law.[80] When the nuclei distance of two silicon atoms stretches (strained) by depositing them on a silicon germanium substrate, the increased inter-nuclei distance of the silicon atoms permits electrons to move 70% quicker, allowing transistors to switch 35% faster.[81]



In 2011, Intel tackled quantum tunneling with its 22nm “Tri-Gate” 3D transistor, allowing the Law to continue.[82] As discussed, a positive voltage applied to the gate allows positive-charged substrate electrons to move to the gate's oxide layer, creating a source and drain channel and turning it ON. The yellow high-k dielectric hafnium (Hf) decreases leakage since it can store electrical energy.[83,84]







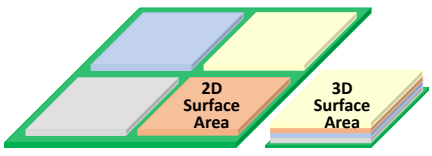
The Tri-Gate transistor, also known as a Fin Field-Effect Transistor (FinFET) is on the left. Its **vertical fin** has a large **conductive area** as shown to the right.[85] FinFETs need half the power and are 37% faster than planar transistors. With higher density, it reduced leakage and heat with a small 2-3% cost increase. FinFETs can have multiple **fins** for enhanced performance as shown to the left. For its time, FinFETs brought a substantial amount of innovation to the Law.

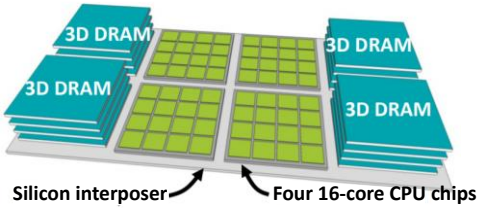
The integrated circuit likewise transitioned from a planar 2D IC to three dimensions, allowing it to keep the same linear footprint (or smaller) by adding height. Stacking ICs continues the density progression, helping to reduce some of the von Neumann bottlenecks by cutting the processor-memory distance. It supports the technical and economic aspects of the Law without requiring expensive manufacturing changes.

An example of this packaging concept is the modern high-rise apartment building. Taking the space of a large single floor building an elevator connects floors (vertical paths.) Hallways (horizontal paths) allow for greater z-direction density without additional 2D space.

22nm node	2D-IC	3D-IC 2 Layers	Decrease
Avg wire length micrometer	6	3.1	48%
Avg gate size (wire length)	6	3	50%
Active area die size mm^2	50	24	52%
Total power (Watt)	1.60	0.80	50%

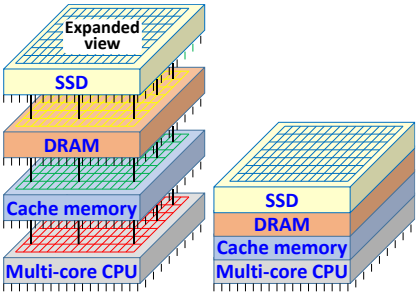
MonolithIC 3D Inc. studied 2D and 3D layouts and found multiple stacked transistor layers using vertical connectivity reduced wire path circuit length and created denser transistor areas by 50% without adverse side effects.[86] To the right, the 2D surface area drops by 75% with a 4-layer 3D package.



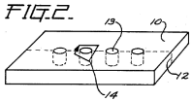


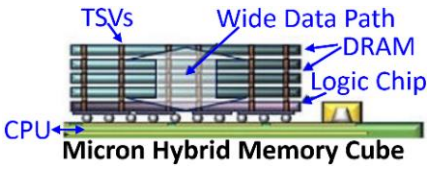
A silicon interposer routes signals between chips allowing for the use of different miniaturization levels in a single package – i.e. a 14nm CPU with 20nm memory. As shown to the left, it can integrate 3D memory stacks and 2D multicore processors for a powerful, low-cost System-on-Chip (SoC) design at current nanometer scales.[87] In this example, a 64 core processor leverages four banks of 3D memory.

While the SoC is useful, it tends to use a relatively large amount of two-dimensional space than an alternate "apartment building" approach called a System-in-Package (SiP). SiP reduces system board space and capital outlay by 3D stacking and integrating different mass-produced dies without excessive complex integration.



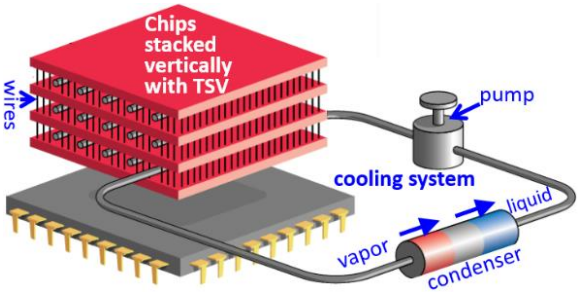
One way to create 3D ICs is "Through-Silicon Via" (TSV).[88] TSV is a SiP using vertical copper wires ("elevator") to connect and stack each chip layer. TSV creates shorter yet higher bandwidth pathway connections, faster memory transfers because of the reduced distances, and lower energy consumption/heat. Transistor co-inventor William Shockley filed a patent in 1958, pictured to the right, for an initial TSV that he envisioned would connect two sides of a wafer.[89]





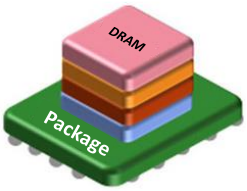
Micron's Hybrid Memory Cube (HMC) on the left has separate TSV DRAM layers with a parallel memory design that was 6X faster than DDR4 memory.[90]

As we've discussed, with a smaller surface area than 2D chips, 3D heat dissipation is a problem. One approach pumps coolant fluid through horizontal pipes ("hallway") in the 3D chip. The pipes, near the diameter of a hair, increase heat dissipation and keep the circuit cooler than a 2D integrated circuit.[91] Liquid pumping systems might someday actually provide chips with power based on a "flow battery" idea.[92] TSV reduces the 3D component paths, shortening the distance electrical signals must travel by as much as 1,000 times and reduces the amount of energy needed and heat generated.[93] On a traditional



motherboard, memory can be a few centimeters from the CPU, so shortening that data path, including 3D closest path routing, causes a dramatic memory speed improvement.

From Moore's perspective, it might be possible to double transistor density by adding layers every two years. He wrote in 1965, "It may prove to be more economical to build large systems out of smaller functions, which are separately packaged and interconnected. The availability of large functions, combined with functional design and construction, should allow the manufacturer of large systems to design and construct a considerable variety of equipment both rapidly and economically."[94] The storage industry did this when it moved from planar 2D NAND to 3D NAND SSDs. With up to 96 layers, storage has become denser, cheaper, and faster than planar NAND.

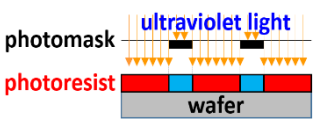


The 3D approach encourages a layered system on a chip functionality. A full system chip could layer a **DRAM** memory module, a **graphics processing unit** (GPU), a **multicore CPU** and **other components** into a **single package**. It reduces planar chip separation from centimeters to millimeters. Building vertically allows engineers to adhere to Moore's Law by shrinking the die by adding layers, which reduces some of the von Neumann computer architecture memory bottlenecks.

The move to 3D is inevitable. Like the rice and chessboard example, a cubic millimeter of silicon has 5x10^19 atoms.[95] Mathematically, a 2D silicon slice has 13x10^12 atoms (thirteen trillion) – i.e. the number of transistors on a millimeter die if they were one atom in size. The Law shows the 2D limit reached in 20 years. Modern processors are a few hundred square millimeters in size, so they will someday reach a limit on how many transistors can fit on a flat die.

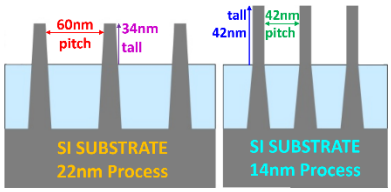
Photolithography

Photolithography prints integrated circuits on a silicon wafer using light, similar to how a film camera takes a picture. In 1826, Joseph Niépce created a process to permanently copy a photograph using asphalt.[96] Bell Labs engineers Jules Andrus and Walter Bond used a photoengraving technique to reproduce circuit designs on a silicon wafer in 1955.[97]

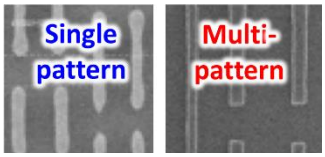


Advances in photolithography equipment, **photoresist** materials, and ingenuity contribute to the success of Moore's Law. Engineers start with a "**photomask**" circuit image data file. An argon fluoride **ultraviolet** (UV) **light** illuminates the **mask** leaving the circuit image on a photosensitive wafer as shown.[98] **Photoresist** liquid polymer applied to the wafer sticks to light exposed areas and

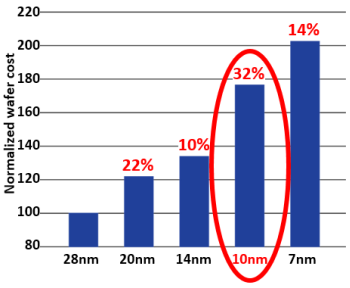
acid removes the **uncoated** material. Multiple iterations of this process, including the application and removal of metal, produce the n-type and p-type required to build transistors and other components.



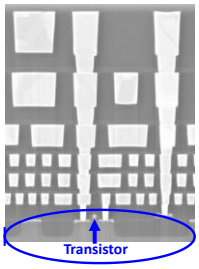
As shown here, smaller pitch gaps mean better performance. Ultraviolet light can build tri-gate **22nm** components with a **60nm pitch** and **34nm tall** as well as 2nd generation **14nm** process tri-gates with **42nm tall** fins and a **42nm** pitch.[99]



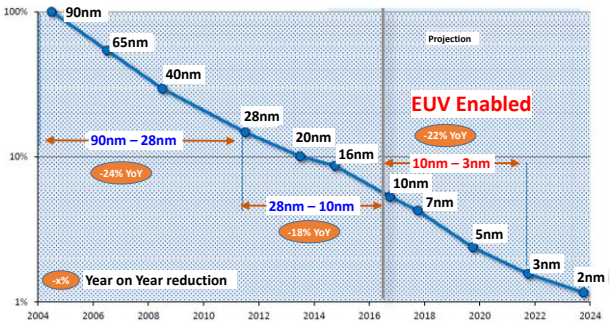
However, UV's 193nm photolithograph wavelength ran into print resolution problems trying to create 10nm processors with a 34nm fin pitch and 53nm height.[100] In this image, UV created a "blurry" 7nm component using a **single pattern**. It needed a **multi-pattern**, multi-exposure process to create sharp images.[101,102] Multi-pattern separates images into distinct patterns through litho-etch-litho-etch (LELE) that require multiple UV passes to produce small chip features.



Multiple patterns extended the Law, but significantly increase manufacturing costs as this chart shows. Each size reduction created expensive multi-pattern challenges. Wafers with **10nm** parts are **32%** more expensive than 14nm dies, due in part to the number of layers required.[103]

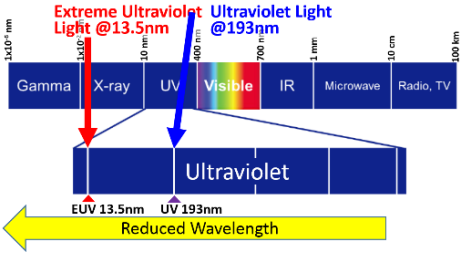


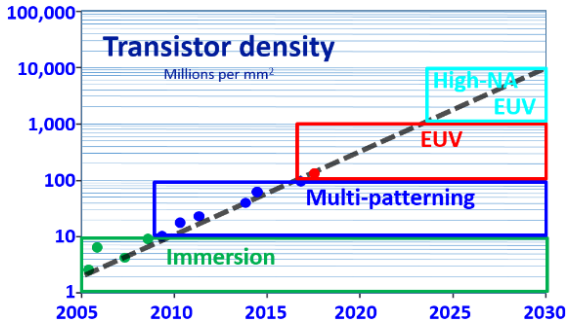
A 3D IC to the right has multiple mask layers. Using UV lithography a 28nm device needs 40-50 mask layers, a 10-14nm device needs 60, and a 7nm device 80-85 layers.[104] Moore wrote about smaller devices having higher cost and greater complexity. As IC manufacturers abandon UV, they are adopting atomic accuracy through extreme ultraviolet (EUV) lithography.



Carrying on with the Law, **EUV** 13.5nm wavelength lithography as shown achieves the same results or better than **UV** without **multi-patterning**.[105] It creates a denser package of smaller transistors with a simple and significantly quicker manufacturing process than **multi-pass 193nm**. **EUV** and **UV** may have roles in the same project based on the resolution required.

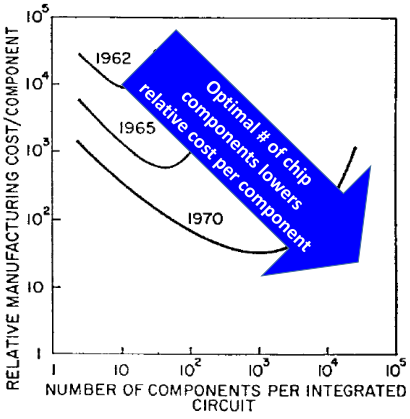
EUV light is created when a powerful laser vaporizes drops of tin at a million degrees Celsius.[106] Tin plasma emits a 13.5nm wavelength of radiation in the **EUV** band as shown to the right. This wavelength is 14 times shorter than UV light and near that of X-rays so it can handle complex patterning in less time and at lower overall costs. GlobalFoundries's CTO cited a 30-day reduction for wafers using **EUV** over the 60-90 days using **UV** due to multi-pattern savings.[107,108] IBM used EUV to build a 5nm transistor.[109] As shown to the left, **EUV** should scale Moore's Law into the 3nm feature size available in the mid-2020s.[110]





The industry should see lower IC prices using single EUV exposure and a billion transistors per mm^2.[111] EUV still has challenges – for example, it doesn't use lenses since they absorb EUV, so processing uses mirrors. In 2024, producing sub-3nm components may require High-Numerical-Aperture (High-NA) EUV techniques, which should scale to 1nm.[112]

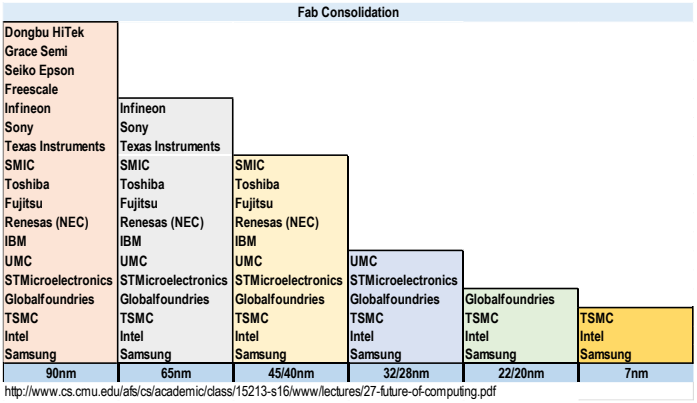
The Cost



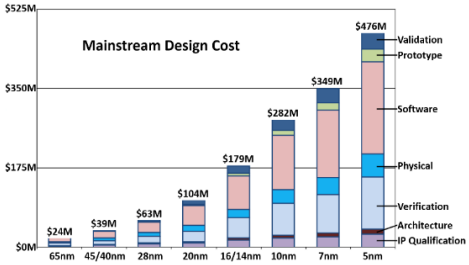
One of the Law's tenets is that transistor cost drops while computing power increases. This 1965 graph illustrates the "sweet spot" relationship where the price of making a component decreases as component density increases. Processes to produce ever-shrinking higher-powered ICs become more complex with each advancement, and the economics to support this makes it harder to justify. While companies like Intel design and manufacture their own ICs, others chose for economic reasons to build their designs at third-party foundries (fabs.) The techniques employed fall into categories such as:

- Shrink the feature size through 193nm argon-fluoride laser, multi-patterning, and EUV.
- Manufacture using 300mm and 450mm wafers.
- Shift from planar to 3D chip layouts.

The R&D, licensing, pre-production outlays, and the move to 3D is expensive and impacts profit margins. Completed designs must go through simulations to make sure the features work at the anticipated performance level and ensure it meets market price points prior to full production.



Foundry ownership is expensive, and with fixed and miniaturization expenses, smaller ICs mean higher costs. That's OK if yield increases, but yield plateaus. Die reduction means more chips per wafer and fewer wafers – not a great business model for a fab. Some believe it isn't physics that kills the Law, but the economics as is evident by unprofitable fabs that no longer build state-of-the-art chips as shown by this chart. GlobalFoundries is a multi-customer contract fab that estimates it would cost $10-12B to build 7nm components and $14-18B to make 5nm products.[113] In August 2018, GlobalFoundries decided against building a 7nm fab and a major customer shifted its entire 7nm production to the Taiwan Semiconductor Manufacturing Company (TSMC).[114,115]



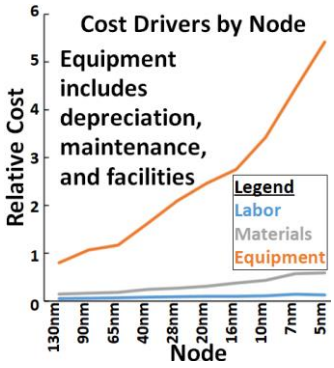
Node reduction has higher design expenses. For example, dense circuits can cause electromagnetic crosstalk and heat dissipation issues from a smaller surface area. As shown to the right, International Business Strategies pegged the 28nm die design and software costs at $63 million and $179 million for a 14nm component.[116] Outlays approached $500 million for 5nm dies, creating a hurdle for small companies struggling with high volume market acceptance of older designs. While there may be production cost savings from EUV and other technologies, 5-7nm may only be attractive to high-end applications at the outset.

Fabrication plant costs are a corollary to Moore's Law, called Rock's Law (sometimes referred to as Moore's Second Law).[117] Arthur Rock, venture capitalist and Intel's first chairman, wrote "The cost of capital equipment to build semiconductors will double every four years." Rock noted a fab cost $12,000 in 1968, and by the mid-1990s, had reached $12 million. The price tag climbed to $3 billion by 2005 and Samsung spent $14 billion for its 2017 plant.[118] Put in context, in 2018 Intel had $17 billion in 2nd quarter revenue and if they build a 5nm fab, the cost would be one quarter their yearly revenue.[119] If this expenditure continues, to paraphrase the wise man and the king, the world may be unable to pay for a chip plant as we know it in twenty years. It means the market for faster chips must continue to grow exponentially, however, at some point the market doesn't need an infinite number of new chips.

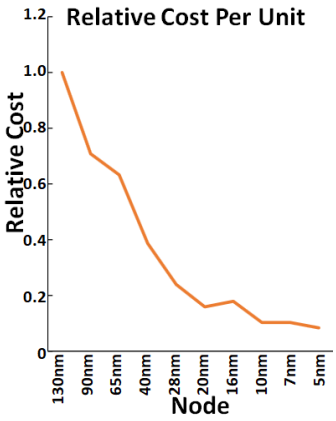
The raw material expense is low given silicon is abundant. A 300mm wafer uses ounces of sand and costs $500.[120] Finished wafers start at $5,000 based on lithography tools, depositing equipment to layer functionality, plasma etchers to burn off unwanted wafer areas, clean-room expenses, fab depreciation, and profit margins.

	45nm	28nm	14nm	10nm
Lithography processs	Single Pattern	Double Pattern	Multiple Pattern	Multiple Pattern
Mask layers	30 -40	40+	60+	80+

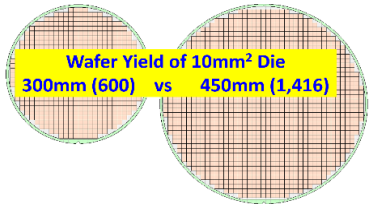
As components get smaller and denser, interconnection space gets tighter. If you can't connect point A to B on the same IC plane, you add a layer to run the connection above or below it. This chart depicts the number of masking layers needed at each progressively smaller manufacturing process.[121] Adding photomask patterns increases complexity, manufacturing expense, and power consumption while taking additional time to produce a finished wafer. With 130nm technology, a set of masks cost from $450,000 to $700,000, and at a 14nm process, the price jumps from $10 million to $18 million.[122] Intel, the largest processor company, amortizes these lithography mask expenses over 100 million processors per quarter.[123]



The chief fabrication expense for shrinking components and larger wafers is the equipment to mass produce items at a decreasing size. On the left, **labor** and **materials** experience a relatively minor increase with shrinking node sizes, with **equipment** costs such as depreciation, maintenance and facilities skyrocketing. Moore's Law and the chart to the right show the positive effect of greater miniaturization is the relative decrease in manufacturing costs per unit.



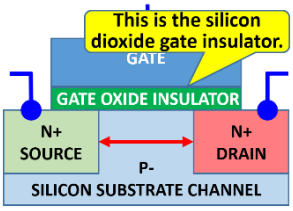
Long amortization periods help offset high costs and help extend Moore's Law. For example, a chip that costs $100 during a two-year production run might drop to $66 if the production cycle stretched to three years. While elongating a production run contradicts the Law's two-year doubling cycle, it significantly lowers unit prices.



Another way to lower chip costs is to use larger wafers. On the far left, a 300mm wafer can create 600 10mm^2 dies while a larger 450mm wafer can create 1,416 dies – a 2.3X increase due to the greater surface area of the wafer and constant die size.

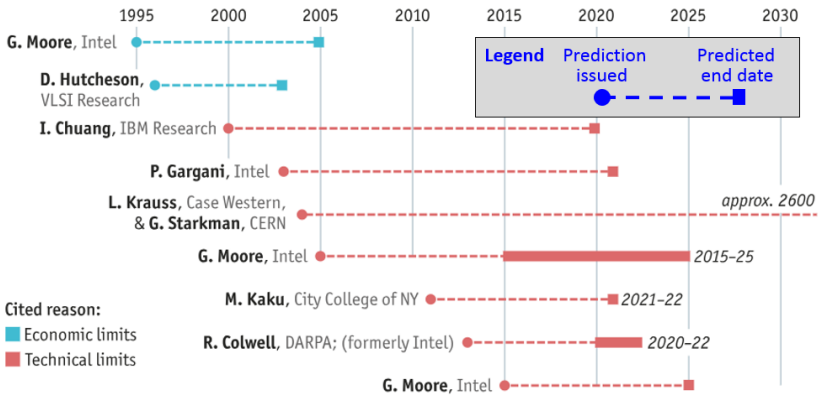
Wafer defects can cause chip defects, and the larger the wafer, the more defective chips they tend to produce. If those defective chips still function, perhaps at a lower frequency or with two fewer cores, they can be sold as low-price components in a process called “binning” – for instance, a faulty Intel i7 can be a fully functional Pentium chip.[124]

The Road Ahead



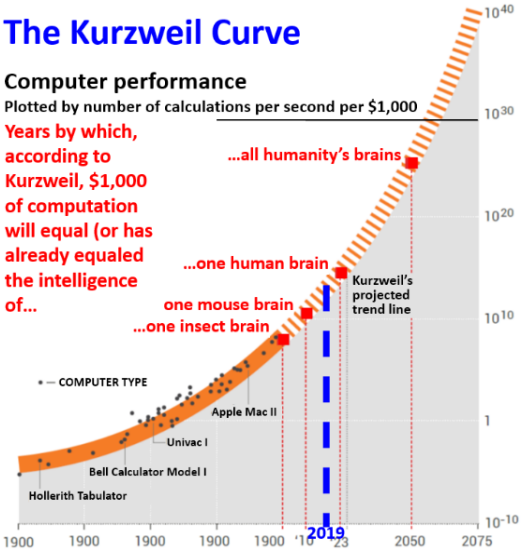
Pundits have long claimed the end of Moore’s Law was in clear sight. For instance, in 1999, Bell Labs researchers concluded the silicon dioxide gate’s insulating properties would break down at 0.7nm.[125] As shown to the left, they believed that by 2012, the gate oxide that insulates the current-carrying electrode from the voltage electrode would be 4-5 silicon atoms thick. It would cause a tunneling current to flow through it and prevent the transistor from reliably changing its on/off state. Their research was undeniable, so they concluded that the fundamental physical limits would halt Moore’s Law and prevent processors from getting faster.

Experts, including Gordon Moore, predict the Law will end in the 2020s.[126] This graphic shows the year they made their prediction and when they expect it to come true. While it remains to be seen, history shows that not all expert predictions come true:



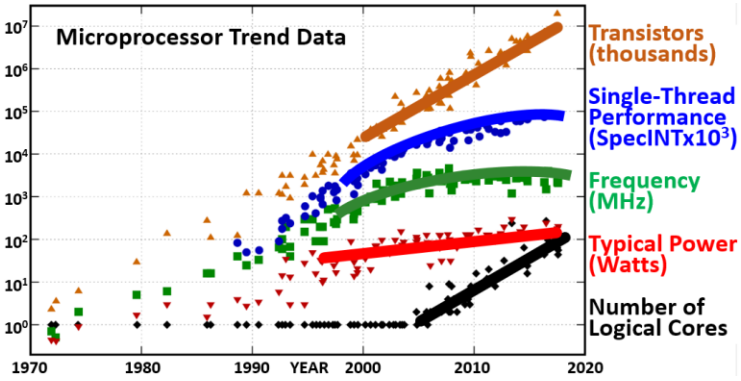
- According to the 1899 U.S. Commissioner of Patents Charles Duell, "Everything that can be invented has been invented."[127]
- "I think there is a world market for maybe five computers," said IBM's chairman and CEO Thomas Watson in 1943.[128]
- Bill Gates said "640KB ought to be enough for anybody" when discussing the amount of memory in the new 1981 IBM PC.[129]

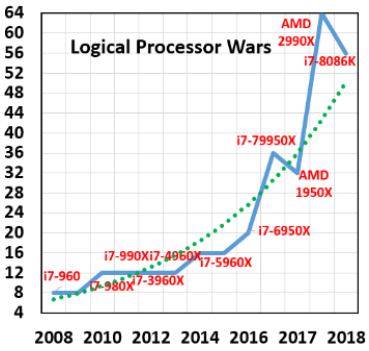
Based on the views of futurist Ray Kurzweil, Gordon Moore’s Law is at its beginning.[130] Ray believes computer power continues its exponential growth as shown to the left. In **2019**, $1,000 worth of performance exceeds the intelligence of a mouse brain but less than a human brain.



Kurzweil projects that by 2050, computational power will be incredibly powerful and inexpensive allowing us to buy a device that equals the computational capability of all humanity for the price of today's modern refrigerator.

The previous Microprocessor Trend chart shows **transistor counts** increasing, while **single thread performance** levels off and **clock frequency** reached its maximum in 2005 as fundamental physical laws were proving difficult to surpass. While **clock frequency** could go higher, heat dissipation and use in battery-powered devices limits **power consumption**.

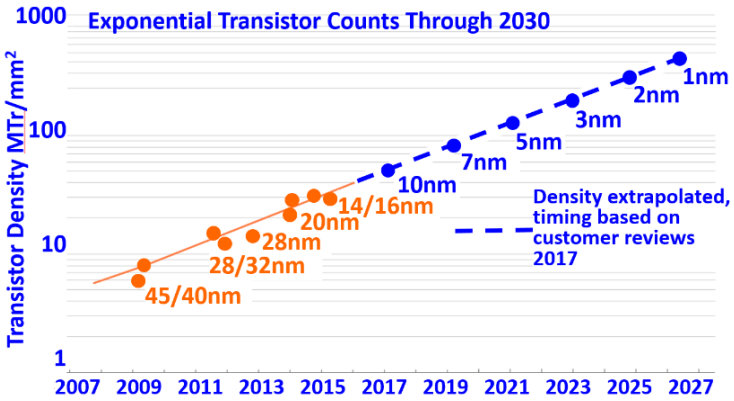




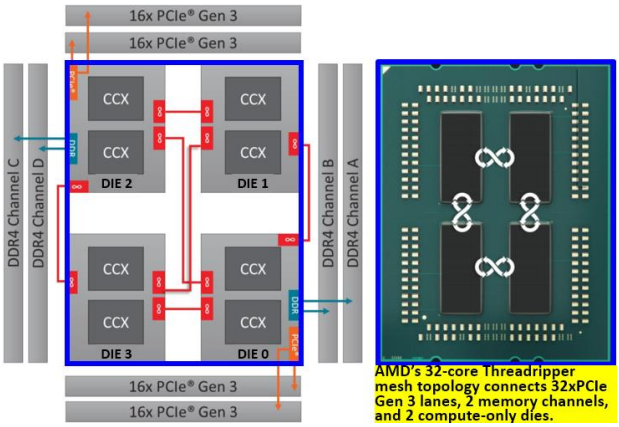
Billions of research dollars went towards creating breakthroughs in physics, engineering, and chemistry. **Logical core** counts rose with heat, power usage and inter-core communication coherency issues addressed.[131] Popular processors from Intel and AMD in "**Logical Processor Wars**" doubled the number of threads in their virtual cores, and the green trend line shows a clear upward "Moore" pattern.

Other advances have occurred in the cache size. Cache memory built from Static Random-Access Memory (SRAM) is faster than DRAM. As discussed, a "cache hit" increases processor throughput. SRAM needs six FETs to store one bit while denser yet slower DRAM uses one FET and capacitor for each bit.[132] While cache memory helps the Law's journey, a 90-95% cache hit rate is good and SRAM is expensive compared to DRAM, so its use is limited.

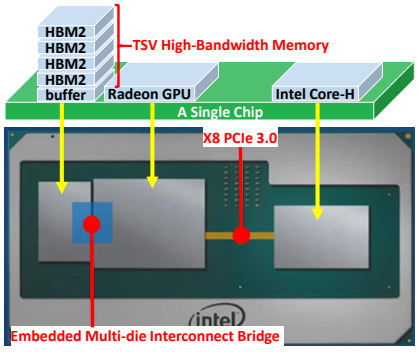
Despite all the controversy, IC advances and architectural tweaking of von Neumann's design have allowed Moore's vision to remain true. This forecasted transistor graph follows the Law's exponential growth line through 2030.[133] For instance, fewer than two years after Intel's Broadwell 22-core 14nm line 7 billion transistor processor, AMD announced their 14nm EPYC server processor with 32 cores and 19.2 billion transistors or more than double the Broadwell count.[134,135]



AMD's core and memory "Infinity Fabric" connects chips at 25-50GBps. Some cores access "near memory" in 64ns and "far memory" in 105ns, addressing a von Neumann memory bottleneck. AMD's next EPYC iteration code-named "Rome" uses the same basic design but leverages a 7nm process for double the density, 64 cores/128 threads, reduced power consumption and PCIe 4.0 for double the transfer rate of 2018 PCIe 3.0 designs.[136]



With Moore's doubling transistor counts and lowering prices, consider that each transistor in the Valkyrie bomber cost $150, Intel's 8008 CPU with 3,500 transistors cost 3.4¢ each, and the latest AMD Ryzen desktop processor with 19.2 billion transistors cost under a penny each ($0.000000094.)[137,138] For proof of the Law's purchasing power, Intel's 14nm Core i7-6950X 10-core processor cost $1,723 in 2016 while the 2018 AMD 12nm TR 2990WX shown above has 32 cores for near the same price.[139,140]



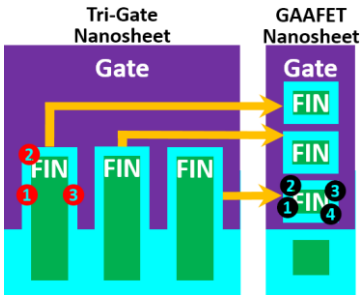
Designed for gamers, Intel's 3D Kaby Lake G follows Moore's Law by combining the processor, GPU and dedicated graphics memory stack in one "2.5D" (the connection of dies in one package) package as shown to the left. An Embedded Multi-die Interconnect Bridge (EMIB) interposer unites the GPU with four 1GB HBM2 DRAM dies stack using TSV connections.[141]

The packaging reduces the overall motherboard real estate by 50% and uses 80% less power than previous GPUs using GDDR5 memory.[142]

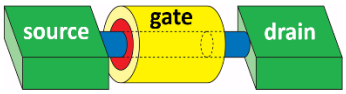
AMD's Zen miniaturization roadmap beyond the 12nm Ryzen calls for a 7nm Zen 2 node in 2019 and a Zen 3 in 2020 using 5nm. Intel, which has tried to maintain Moore's Law of doubling density every two years, has begun shipping 10nm Cannon Lake dual-core i3-8121U laptop chips produced with UV lithography and multi-patterning, after which it is expected to switch to EUV lithography for 7nm designs in 2020.[143,144]

Industry Technology Node	2014	2017	2019	2021	2024
	14nm	10nm	7nm	5nm	3nm

The International Roadmap for Devices and Systems shows 7nm transistors becoming available in 2019 and 3nm by 2024.[145] International Technology Roadmap for Semiconductors forecast 2.5nm and 1.5nm units in 2027 and 2030, while the nanoelectronics company, IMEC, believes we will have 2.5nm by 2024.[146]



Creative engineers shrunk the FinFET fin to 5nm and reached the sub-4 atom threshold further extending the life of the Law.[147] They rotated the vertical fins 90º to the horizontal position as shown in this illustration, and reduced the quantum tunneling issues, allowing them to make even smaller transistors. Gate-All-Around FET (GAAFET) 3D nanosheets increased the number of gates from ❶ and ❸ on the sides with ❷ on top of a Tri-Gate, to four gates with ❶, ❷, ❸, and ❹ on the GAAFET surrounding the fin. In theory, the fins can stack even higher.[148] To the right is a picture of a GAAFET.

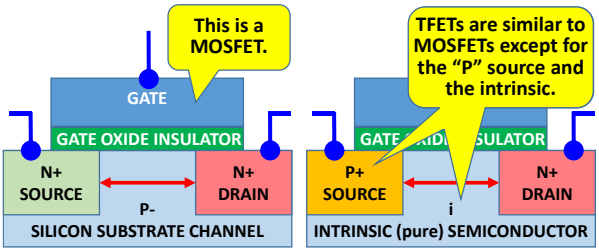


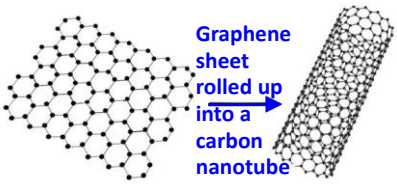
With a gate surrounding the channel, IBM's solution to the left looks like a silicon nanowire with a larger round conducting channel around it. Their Moore's Law solution uses EUV to build a chip with 30 billion of these 5nm stacked silicon nanosheet transistors. Compared to a 7nm FinFET, this design is 40% faster and uses 75% less power than 10nm chips.[149,150,151] IBM believes this process with a silicon-germanium (SiGe) alloy for the channel (same as the Bell Labs transistor) could be in use by 2021 to 2023.

If GAAFET extends to 3nm, a chip in the 2024-2026 timeframe could yield a 60-80% performance gain and a 90-100% power efficiency improvement over 10nm. Silicon GAAFET challenges include a 3nm silicon gate just six atoms thick. At some thickness, it may be physically impossible based on today's science or economically infeasible to produce ever-shrinking silicon transistors. The size may require IBM's SiGe innovation or gallium arsenide

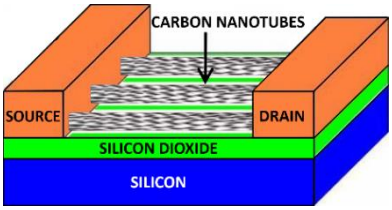
(GaAs) as described by Moore in 1965, both of which allow electrons to move faster through its semiconductor material and ultimately permit transistors to operate at higher frequencies.

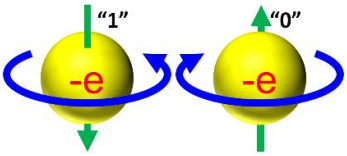
Tunnel FET (TFET), like FinFET, overcomes quantum tunneling leakage by using a large surface area.[152] With a carbon nanotube channel found in GAAFET to support lower voltages and shorter gates, TFET uses quantum mechanical electron tunneling to switch ON and OFF at significantly lower voltages than MOSFETs. Unlike a MOSFET's source and drain which are doped with a boron or arsenic coating to have the same properties, a TFET is doped to have opposite properties.[153] N-type semiconductors have extra electrons, p-type has an electron shortage and intrinsic (i-type) pure semiconductors lack an added conductive coating.





Others think future computers will use materials such as a Carbon NanoTube FET (CNTFET).[154] A single nanometer thick carbon sheet called graphene, shown to the left, is rolled into a hollow cylinder. To the right is a three carbon nanotube transistor showing the source and drain, all sitting on a silicon dioxide/silicon gate.[155] Compared to silicon, nanotubes have extremely high electrical and thermal conductivity, so they use a lot less power, produce less heat, and require less cooling than a FinFET.[156] Engineers could boost clock rates from 4GHz to 10GHz or higher, and build computers that are perhaps twice as fast.[157] A graphene smartphone in 2025 might be classified as a supercomputer by today's standards and have a battery charge that lasts a month.[158]





Another promising approach involves storing information as the subatomic spin of an electron. Electrons spin in a specific direction and are represented by a **north-south pointed arrow** as shown to the left. As discussed, transistors are a "1" when current flows and a "0" when it doesn't. If we disregard an electron's charge and instead focus on their spin, down is a "1" and up is a "0".[159] Spintronics (**spin** **tr**ansport electr**onics**) uses less energy to change direction than a transistor. Altering a spin is quick, and the device is non-volatile when power is turned off. Spintronics has been around since the 1980s and used in Giant Magneto-Resistive (GMR) hard drive heads to sense the magnetic field encoded on drive platters. If we change the classical design of a transistor, then it is possible to extend Moore's Law for a few more decades.[160]

Perhaps further out, engineers will commercialize some of the breakthroughs occurring in the labs. For example, researchers at Columbia University recently revealed a 0.5nm multistate molecular transistor that functions at room temperature and changed states with a single electron.[161] We may someday see a nanocomputer use single phosphorus atom transistors.[162]

Von Neumann's design needs a memory fetched instruction, a decode/execute cycle, and memory to store the result. Engineers believe that while new quantum and neuromorphic computing architectures will break von Neumann software application compatibility, the time is approaching for new ways to solve problems.

Machine Learning & Computer Science	Security & Mission Planning
• Detecting statistical anomalies	• Detecting computer viruses & network intrusion
• Finding compressed models	• Scheduling resources and optimal paths
• Recognizing images and patterns	• Determining set membership
• Training neural networks	• Analyzing graph properties
• Verifying and validating software	• Factoring integers
• Classifying unstructured data	**Healthcare & Medicine**
• Diagnosing circuit faults	• Detecting fraud
• Election modeling	• Generating targeted cancer drug therapies
Optimization	• Optimizing radiotherapy treatments
• Traffic flow / congestion relief	• Creating protein models
• Web advertising	**Financial Modeling**
• Telecommunications network	• Detecting market instabilities
• eCommerce item listing	• Developing trading strategies
Materials Simulation	• Optimizing trading trajectories
• Simulating quantum systems	• Optimizing asset pricing and hedging
• Materials prototyping	• Optimizing portfolios

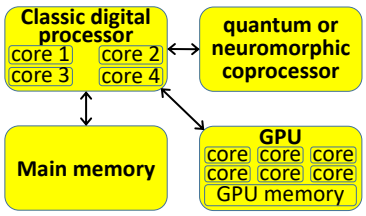
https://www.dwavesys.com/sites/default/files/D-Wave 2000Q Tech Collateral_1029F.pdf

Quantum computing is an alternate approach that relies on the behavior of atomic and subatomic particles and not transistors.[163] A quantum bit (qubit) can be “0” and “1” at the same time. Quantum computers are still in their infancy and require a rewrite of algorithms that focus on solving business problems that deal with many possible parallel outcomes. Binary computers follow Boolean logic while quantum computers are probabilistic, meaning their solution is a likelihood. Presently, there aren’t many quantum computing use cases, unless you need “Shor’s algorithm” for prime number factoring which is useful for cryptography and cybersecurity, “Grover’s algorithm” to search an unordered or unstructured database or a handful of algorithms D-Wave Systems identifies in this chart. A quantum computer requires cryogenic refrigeration to maintain near absolute zero temperatures. It will not speed up your web browsing or word processing and is unlikely to fit on your desk. In the future, quantum computing offers the promise to solve problems that are difficult for traditional computers, such as artificial intelligence and weather prediction.

Neuromorphic machines try to solve problems the way our brains do.[164] This chart shows our brain processes information differently than a CPU, focusing on rapid decision making and creativity rather than computing speed and precision. Digital “neurons” replace transistors, work in parallel without clock frequency timers, and intercommunicate through electrical spikes. Biological networks solve unique problems, such as language comprehension and reasoning,

Mammalian Brains	Computers
• Parallel distributed architecture	• Serial architecture
• Spontaneously active	• No activity unless instructed
• Asynchronous (no global clock)	• Synchronous (global clock)
• Analog computing, digital communication	• Digital computing & communication
• Integrated memory & computation	• Memory & computation are separated
• Low power (25W), small footprint (1 liter)	• High power (9100MW), large footprint (40M liters)
• Learn thru brain-body-environment interaction learning	• Learn via programmed algorithms/rules
• Noisy components operate at low speeds (<10 Hz)	• Precision components operate at very high speeds (GHz)

using far less power than a machine. For example, IBM's Watson 2011 supercomputer that played "Jeopardy!" against contestants used ninety 8-core 3.5GHz servers and 80kW of power versus 25W for a human brain.[165,166] However, like quantum computing, a neuromorphic machine would be challenged to run today's algorithms and software efficiently if at all.



Some engineers believe digital processors can leverage quantum and neuromorphic co-processors, offloading algorithms and code paths that are best suited to specialized processing. Offloading specific complex tasks to a dedicated coprocessor alleviates some von Neumann bottlenecks. Similar to how a GPU coprocessor with specialized cores and dedicated memory excels at rendering real-time video, a linked quantum or neuromorphic coprocessor could speed through unique business problems. While a CPU could perform factoring calculations, the task is accomplished much faster on a specialized processor. Likewise, specialized processors cannot efficiently replace a generalized processor.

Perhaps a decade or two out is photon computing. Photons have no mass, do not generate heat, and compared to electrons pushing through silicon, photons are much faster.[167] However, photonic or optical transistors that feed the output of one to the input of another one without changing wavelength doesn't exist. While it is feasible to build a hybrid photonic computer by swapping out wires for fiber optic fibers, ultimately light must be converted back into electrons to function in an integrated circuit. When it is practical to build a 100% optical computer, we may have yet another Moore's Law on our hands.[168]

Conclusion

In the fifty plus years since integrated circuit pioneer Gordon Moore made his famously insightful and accurate prediction that the number of low-cost transistors placed on a single chip would double every year, daunting technical and economic obstacles that seemed to signal the end of innovation and his Law have been met and overcome by evolving dramatic transformations. The "Law" is fundamentally a commentary on humanity's creativity. Moore made economic observations about the rate of progress of combining smaller transistors in the same silicon area and its corollaries, noting that chip performance increases while price decreases. Based solely on what his article says, his silicon Law must slow down since:

1. Electrons can tunnel through the barrier of all-silicon transistors smaller than 5nm.
2. Increasing transistor density increases power density, which leads to intense heat.
3. The economics imposed by each new miniaturized chip iteration represents a significant increase in design and implementation costs.

All exponentials in the physical world must come to an end, yet computer technology will almost certainly continue after the Law ultimately ends. It's not about transistor density but economics, such as lowering the circuitry price or doubling the circuitry in the same space at the same price. The first 30 years of chip development tackled many daunting problems as processor speeds raced to near 4GHz. The next 20 years saw more elegant (and expensive) ways of getting more performance out of the von Neumann architecture through additional cores, caches, memory technology, the introduction of the solid-state disk, faster networks and more elegant coding.

With billions in profits at stake, expect to see an emphasis on hardware refinements such as wide memory channels and longer instruction pipelines to increase the number of instructions completed per clock cycle, higher core counts, 3D designs, and software parallelism advances over the next 5-10 years. Designing and producing smaller chips with new features is difficult and expensive, so expect Moore's Law to embrace 3D designs. The Law gets an immediate and permanent boost of perhaps 1000-fold if the industry shifts from planar 2D structures to 3D processors and stacked main memory, even while maintaining process size.[169]

One of the things Moore speculated on in 1965 was, "It may prove to be more economical to build large systems out of smaller functions, which are separately packaged and interconnected." He identified the cost of innovation as its biggest threat. His vision included 3D, so it is of little surprise to see manufacturers use three dimensions in their designs. Moore also noted, "Silicon is likely to remain the basic material, although others will be of use in specific applications." Engineers are exploring ways to shrink transistors using additional materials that still incorporate silicon such as silicon germanium or perhaps graphene, and lithography techniques needed to get transistors below 5nm.

Using current techniques of EUV, larger 450mm wafers, co-processors, and 2.5D/SoC 3D, we can see Moore's Law evolving well into the next decade. There is no reason to believe our resourcefulness in building systems has ended. Eventually, modern physics and other disciplines will allow sub-nanometer designs to flourish, carrying on the spirit of the 1965 vision. Smart minds are working on shrinking 5nm transistors by a factor of ten or even a hundred. A liberal interpretation of the Law through the use of new materials, 3D components, and innovation allow us to restate a new version of his Law. We will witness engineers doubling the number of transistors that fit in a given area of IC for another 50 years – the best is yet to come. Then again, exponentials will catch up with transistor engineering. After all, there is a limit to the rice you can put on a chessboard.

Footnotes

[1] "Understanding Moore's Law", Edited by David C. Brock, ISBN 0-941901-41-6, p14
[2] http://calteches.library.caltech.edu/3777/1/Moore.pdf
[3] http://www.newworldencyclopedia.org/entry/John_von_Neumann
[4]https://www.thoughtco.com/history-of-the-eniac-computer-1991601
[5] http://www.columbia.edu/cu/computinghistory/eniac.html
[6]https://history-computer.com/Library/edvac.pdf
[7] http://www.wikiwand.com/en/Von_Neumann_architecture
[8] https://pdfs.semanticscholar.org/9f3c/e4f59f0315d3600cff10d0828bc80561bfff.pdf
[9] https://history-computer.com/Babbage/AnalyticalEngine.html
[10]http://www.tomshardware.com/reviews/upgrade-repair-pc,3000-2.html
[11]http://www.dictionary.com/browse/semiconductor?s=t
[12]https://www.pcmag.com/encyclopedia/term/63223/chip-manufacturing
[13]https://computer.howstuffworks.com/moores-law1.htm
[14]https://en.wikipedia.org/wiki/History_of_computing_hardware
[15]https://en.wikipedia.org/wiki/Integrated_circuit
[16]https://en.wikipedia.org/wiki/Jack_Kilby
[17]https://www.technologyreview.com/s/411485/moores-law/
[18]https://spectrum.ieee.org/semiconductors/processors/the-multiple-lives-of-moores-law
[19]https://www.cs.utexas.edu/~fussell/courses/cs352h/papers/moore.pdf
[20]http://www.tomshardware.com/reviews/intel-xeon-e5-2600-v4-broadwell-ep,4514-2.html
[21]http://download.intel.com/newsroom/kits/22nm/pdfs/22nm_Fun_Facts.pdf
[22] https://www.space.com/17777-what-is-earth-made-of.html
[23] http://www.blackholelab-soft-lithography.com/su-8-photolithography-and-pdms-soft-lithography-products/photolithography-mask
[24]http://www.nikon.com/products/semi/technology/story02.htm
[25]https://en.wikipedia.org/wiki/Wafer_(electronics)
[26]https://www.eetimes.com/author.asp?doc_id=1282825
[27] https://www.yourdisctionary.com/chip-manufacturing
[28]"Understanding Moore's Law", Edited by David C. Brock, ISBN 0-941901-41-6, p14
[29] "Moore's Law: The Life of Gordon Moore, Silicon Valley's Quiet Revolutionary", by Arnold Thackray, David C. Brock, and Rachel Jones. ISBN 0465055648, Prelude
[30]http://semiconductormuseum.com/PhotoGallery/PhotoGallery_2N697.htm
[31]https://wikivisually.com/wiki/Fairchild_Semiconductor
[32]https://newsroom.intel.com/editorials/moores-law-electronics-magazine/
[33] http://www.monolithic3d.com/uploads/6/0/5/5/6055488/gordon_moore_1965_article.pdf
[34]ieeexplore.ieee.org/iel7/5/7270357/07270405.pdf
[35] https://www.technologyreview.com/s/521501/three-questions-for-computing-pioneer-carver-mead/
[36] https://www.youtube.com/watch?v=LbdwbsBbODM
[37]https://www.cs.utexas.edu/~fussell/courses/cs352h/papers/moore.pdf
[38]https://en.wikipedia.org/wiki/Moore's_law
[39] http://ai.eecs.umich.edu/people/conway/VLSI/BackgroundContext/SMErpt/AppB.pdf
[40] "Lithography and the Future of Moore's Law" www.scribd.com/document/6759916/1995-SPIE-Speech
[41] smithsonianchips.si.edu/ice/cd/CEICM/SECTION2.pdf
[42]https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data/
[43] https://wccftech.com/8-facts-intel-amd-nvidia/
[44] https://www.explainthatstuff.com/integratedcircuits.html
[45] "The King's Chessboard" by David Birch, Pufflin Books, ISBN 9780140548808
[46] https://en.wikipedia.org/wiki/Moore%27s_law#Ultimate_limits_of_the_law
[47] https://semiengineering.com/moores-law-a-status-report/
[48] http://nanoscale.blogspot.com/2015/07/what-do-ibms-7-nm-transistors-mean.html
[49] http://faculty.washington.edu/trawets/vc/theory/dna/index.html
[50] https://arstechnica.com/gadgets/2015/07/ibm-unveils-industrys-first-7nm-chip-moving-beyond-silicon/
[51] "An Introduction to Parallel Programming" by Peter Pacheco, ISBN 978-0-12-374260-5", p. 17
[52] https://www.computer.org/cms/Computer.org/magazines/whats-new/2017/04/mcs2017020041.pdf
[53] Crosstalk - component signals that interfere with other components.
[54]https://www.etymonline.com/search?q=cache
[55]http://www.ias.ac.in/article/fulltext/reso/022/12/1175-1192
[56]https://en.wikipedia.org/wiki/Pentium_Dual-Core
[57] https://www.pcworld.com/article/3279264/components-processors/amd-tops-intel-with-32-core-threadripper-2-computex.html
[58] "Computer Systems A Programmer's Perspective", 2nd Edition, ISBN 978-0-13-610804-7, P. 23
[59] https://en.wikipedia.org/wiki/Thread_(computing)
[60] "Computer Organization and Architecture" by Linda Null and Julia Lobur, ISBN 0-7637-0444-X, p. 478

[61] “Computer Organization and Architecture” by Linda Null and Julia Lobur, ISBN 0-7637-0444-X, p. 185
[62]https://www.gizbot.com/computer/features/what-does-hyper-threading-in-cpu-mean-044909.html
[63]https://hal.inria.fr/inria-00615493/document
[64]http://test.lssc.edu/faculty/maurice_a_morgan/Microcomputer Hardware/LectureNotes/LectureNotes_04_DRAM.pdf
[65]
https://www.upgreat.pl/uploads/AKTUALNOSCI/Prezentacje_Smaczne_kaski_w_menu_HPE_20170919/HPE_storage_20170919.pdf
[66] https://ark.intel.com/products/series/99743/Intel-Optane-Memory-Series
[67] https://hothardware.com/reviews/intel-optane-memory-with-3d-xpoint-review-and-performance
[68]http://www.basicknowledge101.com/categories/electricity.html
[69] https://github.com/karlrupp/microprocessor-trend-data/blob/master/42yrs/frequency.dat
[70] http://web.engr.oregonstate.edu/~mjb/cs491/Handouts/parallel.cs419g.6pp.pdf
[71] faculty.chemeketa.edu/ascholer/cs160/Lectures/Week09/2-ParallelProcessing.pptx
[72] https://www.pcper.com/news/Processors/Intel-announces-9th-Generation-Core-processors-8-cores-16-threads
[73]https://arstechnica.com/gadgets/2016/07/itrs-roadmap-2021-moores-law/
[74] https://www.intel.com/content/www/us/en/history/museum-story-of-intel-4004.html
[75]“Computer Systems A Programmer’s Perspective”, 2nd Edition, ISBN 978-0-13-610804-7, P. 799
[76]http://www.cas.mcmaster.ca/~nedialk/COURSES/4f03/Lectures/intro.pdf
[77] https://en.wikipedia.org/wiki/Non-uniform_memory_access
[78] https://www.pcworld.com/article/222704/death_of_moores_law_will_cause_economic_crisis.html
[79] https://spectrum.ieee.org/semiconductors/devices/the-tunneling-transistor
[80] http://download.intel.com/pressroom/kits/advancedtech/pdfs/Mark_Bohr_story_on_strained_silicon.pdf
[81] https://en.wikipedia.org/wiki/Strained_silicon
[82] http://newsroom.intel.com/docs/DOC-2032
[83] http://computer.howstuffworks.com/small-cpu1.htm
[84] https://en.wikipedia.org/wiki/Hafnium
[85] http://download.intel.com/newsroom/kits/22nm/pdfs/22nm-Details_Presentation.pdf
[86] http://www.monolithic3d.com/why-monolithic-3d.html
[87] http://www.eecg.toronto.edu/~enright/Kannan_MICRO48.pdf
[88]https://en.wikipedia.org/wiki/Three-dimensional_integrated_circuit
[89] https://patentimages.storage.googleapis.com/6e/de/d9/bef80c03afb252/US3044909.pdf
[90] https://www.allaboutcircuits.com/news/hybrid-memory-cube-technology/
[91]http://www.kurzweilai.net/3d-chip-stacking-to-take-moores-law-past-2020
[92] https://www.economist.com/technology-quarterly/2016-03-12/after-moores-law
[93]https://www-03.ibm.com/press/us/en/pressrelease/21350.wss
[94] https://www.cs.utexas.edu/~fussell/courses/cs352h/papers/moore.pdf
[95] https://www.futuretimeline.net/forum/topic/14506-they-did-the-math/
[96]https://en.wikipedia.org/wiki/Photolithography
[97]http://www.computerhistory.org/siliconengine/photolithography-techniques-are-used-to-make-silicon-devices/
[98] http://www.me.ntut.edu.tw/introduction/teacher/lu/IC%20fabrication_GA/IC_ch06.pdf
[99] https://www.sec.gov/Archives/edgar/data/50863/000005086314000078/exhibit99_2.pdf
[100] https://semiengineering.com/racing-to-107nm/
[101]https://en.wikipedia.org/wiki/Multiple_patterning
[102] https://www.extreametech.com/computing/160509-seeing-double-tsmc-adopts-new-lithography-technique-to-push-moores-law-to-20nm
[103] http://www.techdesignforums.com/practice/technique/assessing-the-true-cost-of-node-transitions/
[104] https://semiengineering.com/moores-law-a-status-report/
[105] https://staticwww.asml.com/doclib/investor/presentations/2018/asml_20180314_2018-03-14_BAML_Taiwan_March_2018_FINAL.pdf
[106] https://bits-chips.nl/artikel/boxing-with-tin-droplets-to-generate-euv-light-48638.html
[107] https://en.wikipedia.org/wiki/Extreme_ultraviolet_lithography
[108] https://forums.anandtech.com/threads/7nm-euv-in-2019.2497719/
[109] https://semiengineering.com/extending-euv-to-2nm-and-beyond/
[110] https://staticwww.asml.com/doclib/investor/presentations/2017/asml_20170322_2017-03-22_BAML_Taiwan.pdf
[111] https://staticwww.asml.com/doclib/investor/presentations/2018/asml_20180314_2018-03-14_BAML_Taiwan_March_2018_FINAL.pdf
[112] https://semiengineering.com/extending-euv-to-2nm-and-beyond/
[113] https://venturebeat.com/2017/10/01/globalfoundries-next-generation-chip-factories-will-cost-at-least-10-billion/
[114] https://www.bloomberg.com/news/articles/2018-08-27/globalfoundries-gives-up-on-advanced-chip-production-technology
[115] https://www.pcworld.com/article/3300620/components-processors/amd-loses-another-key-executive-jim-anderson-as-it-shifts-manufacturing-to-tsmc.html
[116] https://semiengineering.com/whats-after-finfets/
[117] “The Essentials of Computer Organization and Architecture” by Linda Null and Julia Lobur,

ISBN 0-7637-0444-X, P. 25
[118] https://www.tomshardware.com/news/samsung-14-billion-chip-plant,29058.html
[119] https://www.intc.com/investor-relations/investor-education-and-news/investor-news/press-release-details/2018/Intel-Reports-Second-Quarter-2018-Financial-Results/
[120] http://www.semi.org/en/node/50856
[121] https://semiengineering.com/mask-maker-worries-grow/
[122] https://www.imf.org/~/media/Files/Conferences/2017-stats-forum/session-6-kenneth-flamm.ashx
[123] https://www.imf.org/~/media/Files/Conferences/2017-stats-forum/session-6-kenneth-flamm.ashx
[124] https://linustechtips.com/main/topic/850608-is-this-what-happenes-to-i7-cpus-when-they-dont-meet-the-advertised-perfomance/
[125] https://www3.nd.edu/~gtimp/images/Nature.pdf
[126] https://www.economist.com/technology-quarterly/2016-03-12/after-moores-law
[127] https://en.wikipedia.org/wiki/Charles_Holland_Duell
[128] https://en.wikipedia.org/wiki/Thomas_J._Watson
[129] https://www.computerworld.com/article/2534312/operating-systems/the--640k--quote-won-t-go-away----but-did-gates-really-say-it-.html
[130] www.aspiresys.com/sites/default/files/WhitePapers/Multidimensional-Framework-for-Digital-Transformation_0_0.pdf
[131] https://www.pcworld.com/article/3295003/components-processors/amd-2nd-gen-32-core-ryzen-threadripper-2-price-specs-features.html
[132] https://www.computer.org/cms/Computer.org/magazines/whats-new/2017/04/mcs2017020041.pdf
[133] https://staticwww.asml.com/doclib/investor/presentations/2018/asml_20180314_2018-03-14_BAML_Taiwan_March_2018_FINAL.pdf
[134]https://ark.intel.com/products/91317/Intel-Xeon-Processor-E5-2699-v4-55M-Cache-2_20-GHz
[135]https://www.pcworld.com/article/3279264/components-processors/amd-tops-intel-with-32-core-threadripper-2-computex.html
[136] https://www.top500.org/news/amd-takes-aim-at-performance-leadership-with-next-generation-epyc-processor/
[137] https://wikivisually.com/wiki/List_of_Intel_Core_i9_microprocessors
[138] https://www.pcworld.com/article/3295003/components-processors/amd-2nd-gen-32-core-ryzen-threadripper-2-price-specs-features.html
[139] https://www.pcworld.com/article/3296378/components-processors/2nd-gen-threadripper-review-amds-32-core-cpu-is-insanely-fast.html
[140] https://www.pcworld.com/article/3297499/components-processors/should-you-buy-a-32-core-threadripper-2990wx.html
[141] https://www.anandtech.com/show/12003/intel-to-create-new-8th-generation-cpus-with-amd-radeon-graphics-with-hbm2-using-emib
[142] http://hexus.net/tech/news/cpu/113891-intel-8th-gen-core-chips-now-house-radeon-rx-vega-graphics/
[143] https://www.eetimes.com/document.asp?doc_id=1333230
[144] https://en.wikichip.org/wiki/7_nm_lithography_process
[145] https://e3s-center.berkeley.edu/wp-content/uploads/2017/12/1-1-1_Gargini.pdf
[146] https://semiengineering.com/transistor-options-beyond-3nm/
[147] https://www.elektormagazine.com/news/world-record-5-nm-gaafet-ic-from-ibm-samsung-globalfoundries
[148] https://semiengineering.com/kc/knowledge_center/Gate-All-Around-FET/192
[149] https://www-03.ibm.com/press/us/en/pressrelease/52531.wss
[150] https://www.anandtech.com/show/8223/an-introduction-to-semiconductor-physics-technology-and-industry/7
[151] https://www.pcgamesn.com/ibm-5nm-chip
[152] https://spectrum.ieee.org/semiconductors/devices/the-tunneling-transistor
[153] https://www.revolvy.com/page/Tunnel-field%252Deffect-transistor?uid=1575
[154] https://en.wikipedia.org/wiki/Carbon_nanotube_field-effect_transistor
[155] http://miscircuitos.com/field-effect-transistors-based-carbon-nanotubes-cntfets/
[156] https://en.wikipedia.org/wiki/Carbon_nanotube
[157] https://www.sciencedaily.com/releases/2016/05/160518120130.htm
[158] https://www.youtube.com/watch?v=UUO-f0kIgVU
[159] https://en.wikipedia.org/wiki/Spintronics
[160] https://www.pctechguide.com/hard-disks/hard-disk-gmr-technology
[161] http://engineering.columbia.edu/news/latha-venkataraman-single-molecule-transistor
[162] https://spectrum.ieee.org/semiconductors/nanotechnology/a-singleatom-transistor
[163] https://en.wikipedia.org/wiki/Quantum_computing
[164] https://en.wikipedia.org/wiki/Neuromorphic_engineering
[165] https://en.wikipedia.org/wiki/Watson_(computer)
[166] https://www.economist.com/technology-quarterly/2016-03-12/after-moores-law
[167] https://whatis.techtarget.com/definition/optical-computer-photonic-computer
[168] https://youtu.be/eFhgb5CqAy8
[169] https://www.livescience.com/52207-faster-3d-computer-chip.html