From Microchips to “Microbe”chips? (Toh Keng Hian, Computer Science)

Personal Motivation for Choice of Topic

I picked this topic because… have you seen the price of RAM recently due to the AI boom? If I can't afford parts for my dream gaming PC, I have no choice but to build a computer from the ground up myself. What is free and all around us? That’s right, microbes!

Jokes aside, modern computing is facing a very real crisis, that being Moore's Law has been dead for over a decade. Moore's Law is the historical observation that the number of transistors on a microchip doubles approximately every two years. Microchips are the “brains” of a device, and transistors are the tiny silicon on/off switches, controlled by electricity, that allow them to calculate. For decades, the industry strategy was simple, to cram as many transistors onto a chip as physically possible to increase speed and power efficiency.

Due to fundamental physical limits and economic concerns, we can no longer rely on shrinking transistors to improve computing power. While this means waiting an extra millisecond for a webpage to load, the true travesty lies in the impact on critical fields like climate change modeling, drug discovery, and bioinformatics. These disciplines require processing massive datasets, from global temperature grids to billions of DNA sequences. For many, the continued pursuit of computing performance is literally a matter of life and death.

Furthermore, our reliance on silicon presents a severe sustainability crisis. As global digitization accelerates, toxic e-waste containing heavy metals like mercury is polluting the environment. Simultaneously, the massive energy consumption required to train Artificial Intelligence models is driving up carbon emissions. Given the physical and environmental limits of silicon, this essay explores whether carbon-based biological solutions can offer a more powerful and sustainable computing alternative.

The Intersection of Biology and Tech: Cellular Computing

We will specifically explore Cellular Computing, which is the intersection of computer science and biology that studies the use of living cells as computational hardware.

At first glance, this crossover might seem strange, but the two fields run perfectly parallel. A conventional computer is a complex machine that communicates and executes instructions based on written code. Similarly, cells and microorganisms are simply microscopic machines that execute instructions based on the code written in their DNA. To build a biological computer, we must replicate a conventional computer's two primary components: the Central Processing Unit (CPU) and the data storage device (like an SSD or HDD).

Biological CPU

CPUs perform calculations using fundamental logical operations: AND, OR, and NOT. These operations take inputs and produce a specific output using hardware called logic gates (built from transistors). For example, an AND gate can perform basic multiplication (1 AND 0 outputs 0; 1 AND 1 outputs 1).

It turns out we can modify microbes to perform these exact logical operations. Green et al. (2017) and a team at Harvard designed strands of RNA mathematically engineered to fold and bind in highly predictable ways, essentially building biological AND, OR, and NOT gates. The "inputs" are specific RNA molecules floating in the cell. When the RNA logic gate detects these inputs, it unfolds and produces a specific target protein, which is the "output." Using this method, they successfully programmed an *E. coli* cell to compute a multi-input logic expression.

While an individual logic gate is mathematically simple, combining them allows us to tackle complex problems. Tamsir et al. (2011) demonstrated this by engineering a NOR gate (a combination of NOT gate and OR gate) using two distinct bacterial colonies. They programmed the output of the first colony to be a specific quorum-sensing chemical. This chemical naturally diffused into the environment and triggered the input of the second colony. By using chemical signals as biological "wires," it is theoretically possible to network basic biological logic gates into an entire microbial CPU.

Biological Storage

If microbes execute instructions specified by their DNA, can we draw a parallel from silicon architecture and store our digital data in DNA as well?

Modern SSDs store data using the binary numeral system (1s and 0s). DNA, however, is constructed from four nitrogen base pairings: A-T, T-A, G-C, C-G. From a computer science perspective, this is a quaternary numeral system. Because data can be translated between any mathematical base systems, digital files can theoretically be converted into genetic sequences.

Church (2012) proved the feasibility of this by successfully translating a book, which includes text and images, into synthetic DNA, and then reconstructing the digital files with minimal error. This makes DNA the most information-dense storage medium humanity possesses.

Goldman et al. (2013) highlighted another major benefit: extreme longevity. If kept cool, DNA can last for thousands of years without degrading, far outlasting magnetic tapes or hard drives that fail within decades. It is also entirely passive, requiring none of the constant electricity or cooling demanded by massive data centers.

While storing data *in vitro* (in a test tube) is efficient, it lacks resilience. Consequently, frontier research is exploring *in vivo* storage that puts encoded DNA directly into living microbes. Shipman (2017) demonstrated that a living population of *E. coli* could capture, store, and pass down complex digital information over multiple generations. If future innovations can

perfectly guard against mutation, we could store human history inside resilient extremophile microbes that could survive a nuclear catastrophe, creating a true "living archive." While microbes have historically altered human history, we are approaching an era where we can alter microbes to passively safeguard the history of our coexistence.

Shortcomings

Now that we have the blueprint for a biological PC, can I launch my favorite game? Not exactly. Biological computers have a glaring flaw: they are astronomically slower at executing sequential instructions compared to silicon chips.

Computing tasks are generally divided into sequential computing (executing instructions one after another) and parallel computing (executing multiple instructions simultaneously). While parallel computing is faster, sequential processing is unavoidable due to dependencies. Your computer must sequentially fetch this report from your SSD before it can render it on the monitor. Modern software relies heavily on a combination of both to maximise performance.

Because biological computing relies on the physical diffusion of molecules rather than the speed-of-light travel of electrons (electricity), sequential tasks take time. Gao et al. (2026) demonstrated that a simple sequential biological instruction takes roughly 31 minutes to execute. In contrast, a modern laptop can execute an instruction every 0.25 nanoseconds. Because biological computing is trillions of times slower at linear tasks, it is currently infeasible to replace silicon microchips for everyday use.

Opportunities

However, the biological paradigm is not without immense advantages. While silicon is constrained to the flat, 2D wiring of a circuit board, biological computers operate as a swarm in a 3D liquid environment. Without the need for physical wires, trillions of RNA logic gates can interact simultaneously, meaning biological computers have the potential to vastly outperform silicon specifically in massively parallel computing tasks.

Furthermore, biological computing offers a radical solution to tech's sustainability crisis. Molecular chemistry is exponentially more energy-efficient than electrical resistance, generating virtually zero heat. Most importantly, when a biological computer finishes its lifespan, it doesn't become toxic landfill waste as it can be safely neutralized and composted back into the earth.

Reflections on the Interdisciplinary Journey

While I began writing this essay partially as a joke, the research process revealed how deeply intertwined my major is with the biological sciences. I saw how computing concepts I’ve studied, like dynamic programming and AI, are actively solving microbiology's grand challenges, such as protein folding and DNA sequence analysis.

More surprisingly, I discovered how the "wet" science of microbiology has directly inspired the "cold, hard" logic of Computer Science. Beyond cellular processors, biology has inspired Artificial Immune Systems for network security and Genetic Algorithms that mimic microbial mutation to optimize complex code. This revelation has not only made me more appreciative of the elegance of Computer Science, but it has opened my eyes to the incredible potential of interdisciplinary innovation. Given how crucial microbiology is to public health, this module has inspired me to consider how my computational skills might one day contribute to this life-saving research.

References

- Church, G. M., Gao, Y., & Kosuri, S. (2012). Next-generation digital information storage in DNA. *Science (New York, N.Y.)*, 337(6102), 1628. https://doi.org/10.1126/science.1226355
- Gao, R., Dong, W., Xie, A., & Wei, X.-S. (2026). Programmable DNA Strand-Displacement Circuits for Emulating Digital Sequential Logic Devices (Version 1). *ACS Publications*. https://doi.org/10.1021/acsami.6c00972.s001
- Green, A., Kim, J., Ma, D. et al. (2017). Complex cellular logic computation using ribocomputing devices. *Nature* 548, 117–121. https://doi.org/10.1038/nature23271
- Goldman, N., Bertone, P., Chen, S. et al. (2013). Towards practical, high-capacity, low-maintenance information storage in synthesized DNA. *Nature* 494, 77–80. https://doi.org/10.1038/nature11875
- Shipman, S., Nivala, J., Macklis, J. et al. (2017). CRISPR–Cas encoding of a digital movie into the genomes of a population of living bacteria. *Nature* 547, 345–349. https://doi.org/10.1038/nature23017
- Tamsir, A., Tabor, J. & Voigt, C. (2011). Robust multicellular computing using genetically encoded NOR gates and chemical ‘wires’. *Nature* 469, 212–215. https://doi.org/10.1038/nature09565