DNA Data Storage — Breakthrough Research Proposals

Independent Research Submission

Prepared by: Shimi | April 2026

This document presents a set of original research ideas aimed at addressing the three core barriers blocking DNA data storage from practical deployment: high cost, slow read/write speed, and usability. Each proposal includes a core concept, the deeper mechanism behind it, and suggested implementation pathways.

1. Reducing Cost

[COST-A] Neural compression encoding — write less, store more

Rather than encoding raw binary data into DNA bases at 2 bits per base, a domain-specific neural compression model maps file content into a compact symbolic sequence before synthesis. This reduces the volume of DNA that needs to be synthesized — the primary cost driver.

Deeper mechanism:

A separate compression model is trained per data domain (video, genomic data, documents). Each model learns a latent representation where common patterns are collapsed into short token sequences — analogous to how LLMs represent language as embeddings. Compression ratios of 100:1 to 1000:1 are achievable before synthesis begins.

- File input → domain-specific neural encoder → compressed token stream
- Token stream → DNA synthesis (far fewer bases written)
- Read pipeline: DNA → sequencing → token stream → neural decoder → file
- Error-correcting codes operate on the token stream, not raw bits

Key insight:

Cutting bases written cuts synthesis cost directly. Computation is cheap; chemistry is expensive. Shift the workload accordingly.

[COST-B] LEGO-DNA: reusable pre-synthesized fragment library

Pre-synthesize a fixed library of thousands of short DNA oligos (fragments). Data encoding becomes a fragment selection and ordering problem rather than de-novo synthesis for every write operation. Ligation chemistry assembles selected fragments into full-length data strands.

Deeper mechanism:

A library of ~65,000 unique 20-mer oligo fragments covers a 16-bit address space. Any byte stream can be represented as an ordered selection of fragments. The cost of synthesizing each fragment is amortized across millions of write operations — similar to how a semiconductor mask is expensive to make once but cheap per chip produced.

- One-time synthesis of the full fragment library (capital cost)
- Each write = fragment selection + ligation assembly (marginal cost approaches zero)
- Library fragments are reusable indefinitely if stored correctly
- Supports parallel writes: different files assemble different fragment subsets simultaneously

[COST-C] Enzymatic synthesis with on-chip electrochemical feedback

Enzyme-based DNA synthesis (using Terminal deoxynucleotidyl Transferase, TdT) is biologically cheaper than the standard phosphoramidite chemical process. Adding real-time electrochemical verification at each base addition step eliminates wasted synthesis from undetected errors.

Deeper mechanism:

Current chemical synthesis must discard full strands after discovering errors at the end. An on-chip nanoelectrode sensor detects miscorporation at the single-base level, allowing immediate retry before the strand continues growing. This reduces material waste and increases yield — both lowering cost per correct base synthesized.

2. Improving Read/Write Speed

[SPEED-A] Massively parallel microfluidic write array

A microfluidic chip with 10,000+ independent synthesis wells, each writing a different DNA fragment simultaneously. Instead of physical valves (slow), each well is addressed electrochemically — a voltage signal activates a specific well in sub-millisecond time.

Deeper mechanism:

Current synthesis instruments write sequentially or in small batches. A 10,000-channel chip writing in full parallel could achieve write throughputs comparable to SSD sequential speeds. The architecture mirrors a GPU: many slow cores working simultaneously outperform a few fast ones for parallelizable tasks.

- Electrochemical addressing: no moving parts, sub-millisecond well switching
- Each well synthesizes one fragment independently
- Scheduling algorithm distributes file fragments across wells to maximize utilization
- Fabrication is compatible with existing MEMS/semiconductor manufacturing

[SPEED-B] Direct signal-to-data AI decoder for nanopore reading

The current DNA storage read pipeline has three steps: raw nanopore signal → basecaller → DNA sequence → data decoder. Each step adds latency and accumulates error. A transformer model trained to decode raw ionic current signals directly into stored data symbols eliminates the intermediate basecalling stage entirely.

Deeper mechanism:

The model is trained on paired datasets: known stored data → synthesized DNA → simulated and measured nanopore current signals. Deployed on the sequencer's onboard FPGA or edge GPU, it processes the data stream in real time as strands pass through the nanopore — achieving 3 to 10x lower end-to-end read latency compared to current pipelines.

- Training: synthetic data → DNA → nanopore signal simulation (fully automated)
- Inference: raw current waveform → stored data output (single model, no intermediate steps)
- FPGA deployment: processes signals in real time without buffering full reads
- Fine-tuning on real hardware compensates for device-specific noise profiles

Why this matters:

Basecalling is currently the primary bottleneck in nanopore read latency. Skipping it entirely — rather than making it faster — is a more durable solution as model architectures improve.

3. Making DNA Storage Practical

[USE-A] Two-tier DNA filesystem for random access

The core barrier to practical DNA retrieval is the absence of random access — you cannot seek to a position in a DNA pool the way a hard drive head moves to a sector. A two-pool architecture solves this without requiring new hardware.

Architecture:

- Pool A (metadata pool): small, fast to fully sequence. Contains short DNA strands encoding filename, file size, checksum, and the PCR primer pair that addresses the data in Pool B.
- Pool B (data pool): large archive. Contains all file content, addressed by PCR primers.
- Retrieval: sequence Pool A first (seconds) → identify primers for target file → PCR-amplify only the matching fragments from Pool B → decode data.
- Result: retrieval requires sequencing a tiny fraction of the archive per lookup.

This mirrors the index+data structure of modern filesystems (NTFS, EXT4) but implemented in molecular chemistry rather than magnetic domains.

[USE-B] DNA-SSD hybrid drive with automatic tiering

A single logical storage device combining flash memory (active data) with a DNA cartridge (archival data). Automated tiering software moves files between tiers based on access frequency — completely transparent to the user and operating system.

How it works:

- The OS sees one logical drive — no special software or workflow required from the user
- A background daemon monitors file access timestamps (analogous to Linux atime)
- Files not accessed in a configurable period (e.g. 30 days) are encoded and written to DNA during device idle time
- A retrieval request for a DNA-stored file triggers: PCR amplification → nanopore read → flash restore → file served to user
- Estimated retrieval latency: 15–60 minutes — acceptable for archival content
- A manifest index in flash memory provides instant lookup of what is stored in DNA without sequencing

User experience goal:
DNA storage should feel like a very slow external drive, not a laboratory procedure. The hybrid tier abstraction achieves this.

4. Unified Architecture Vision

The proposals above are individually implementable, but they compose into a coherent end-to-end system:

- Write path: File → neural compression → LEGO-DNA fragment assembly → parallel microfluidic synthesis → DNA cartridge storage
- Read path: Access request → PCR primer lookup (metadata pool) → targeted amplification → nanopore sequencing with direct AI decoding → file restored to flash tier
- Cost model: one-time fragment library synthesis; per-write cost is assembly + ligation only
- Speed model: 10,000-channel parallel write; real-time AI read decoding
- Usability: standard drive interface; auto-tiering handles all DNA interaction invisibly

Each component can be developed and validated independently, making this a modular research roadmap rather than a single monolithic project.