DNA Punch Cards: Encoding Data on Native DNA Sequences via Topological Modifications

S Kasra Tabatabaei[1], Boya Wang[2], Nagendra Bala Murali Athreya[3]*, Behnam Enghiad[5], Alvaro Gonzalo Hernandez[4†], Jean-Pierre Leburton[3†], David Soloveichik[2†], Huimin Zhao[1,5,6 §], Olgica Milenkovic[3§]

[1] Center for Biophysics and Quantitative Biology, University of Illinois at Urbana-Champaign, Urbana, Illinois, 61801, USA,

[2] Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, Texas, 78712, USA,

[3] Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA,

[4] Roy J. Carver Biotechnology Center, University of Illinois at Urbana-Champaign, Urbana, Illinois, 61801, USA,

[5] Department of Chemical and Biomolecular engineering, University of Illinois at Urbana-Champaign, Urbana, Illinois, 61801, USA,

[6] Carl R. Woese Institute for Genomic Biology, University of Illinois at Urbana-Champaign, Urbana, Illinois, 61801, USA,

*, †: These authors contributed equally.

§: To whom the correspondence should be addresses. Emails: zhao5@illinois.edu, milenkov@illinois.edu.

Abstract

Synthetic DNA-based data storage systems (1-11) have received significant attention due to the promise of ultrahigh storage density. However, all proposed systems suffer from high cost, read-write latency and error-rates that render them impractical. One means to avoid synthesizing DNA is to use readily available native DNA. As native DNA content is fixed, one may adopt an alternative recording strategy that modifies the DNA topology to encode desired information. Here, we report the first macromolecular storage paradigm in which data is written in the form of "nicks (punches)" at predetermined positions on the sugar-phosphate backbone of native dsDNA. The platform accommodates parallel nicking on multiple "orthogonal" genomic DNA fragments, paired nicking and disassociation for creating "toehold" regions that enable single-bit random access and strand displacement. As a proof of concept, we used the multiple-turnover programmable restriction enzyme *Pyrococcus furiosus* Argonaute (12) to punch files into the PCR products of *Escherichia coli* genomic DNA. The encoded data is reliably reconstructed through simple read alignment.

One sentence summary

We propose a novel cost-efficient and low-latency method for DNA-based data storage that uses native DNA and a programmable nickase to record data in the form of nicks and also enables strand displacement computing and bitwise random access.

Main Text

All existing DNA-based data recording architectures store user content in synthetic DNA oligos (1-11) and retrieve desired information via next-generation (e.g., HiSeq and MiSeq) or nanopore sequencing technologies (5). Although DNA sequencing can be performed routinely and at low cost, *de novo* synthesis of DNA strands with a predetermined content is a major bottleneck (14); DNA synthesis protocols add one nucleotide per cycle and are inherently slow and prohibitively expensive compared to existing optical and magnetic writing mechanisms. To address these limitations of DNA-based data storage systems and reduce their cost, we developed a new storage paradigm that represents information via *in vitro* topological modifications on native DNA sequences (e.g., genomic DNA or its cloned or PCR-amplified products).

In the write component of the proposed system (Figure 1, top), binary user information is converted into a positional code that describes where native DNA sequence is to be topologically modified, i.e. nicked. A nick is a cut in the sugar-phosphate backbone between two adjacent nucleotides in double-stranded DNA, and each nick encodes either $\log_2 2 = 1$ bit (if only one strand is allowed to be nicked or left unchanged) or $\log_2 3 = 1.58$ bits (if either of the two strands is allowed to be nicked or both left unchanged). As bacterial cells are easy to handle and grow, the native DNA nicking substrates of choice are the PCR products of one or multiple regions of the bacterial genomic DNA, that can be easily isolated via simple and inexpensive commercially available protocols. Native DNA is organized into *orthogonal registers*, with each register represented by multiple replicas of one isolated genomic region; two registers are termed orthogonal if their sequence edit distance is sufficiently large (>55%). Each register is nicked in a combinatorial fashion, determined by the information content to be stored. To enable fast and efficient data recording, a library of registers with desired nicking site patterns is created in parallel. Registers or orthogonal registers are subsequently placed into grids of microplates that enable random access to registers and spatially organize the data, similar to tracks and sectors on disks and tapes.

In the read component of the proposed system (Figure 1, bottom), nicked DNA may be processed using different protocols: next-generation sequencing (NGS), solid-state nanopore sequencing and strand displacement (16-18). As demonstrated by our molecular dynamics simulations, MoS_2 nanopore sequencers can operate directly on the nicked DNA by correlating ionic and transverse sheet currents (Figures S12-S14).

The register chosen for experimental verification is a DNA fragment of length 450 bps that was PCR-amplified from the genomic DNA of *E. coli* K12 MG1655. The register contains ten designated nicking positions. Although registers as long as 10 Kbps can be easily accommodated, they are harder to read; hence, multiple orthogonal registers are preferred to long registers. The nicking positions are determined based on four straightforward to accommodate sequence composition constraints (Supplementary Information; Section C.1) that enable precise nicking. To prevent disassociation of the two strands at room temperature, the nicking sites are placed at a conservative distance of at least 25 bps apart. The user file is parsed into 10-bit strings which are converted into nicking positions of spatially arranged registers, according to the rule that a '1' corresponds to a nick, while a '0' corresponds to the absence of a nick (the number of bits recorded is chosen based on the density of nicks and the length of the register). As an example, the string 0110000100 is converted into the positional code 238, indicating that nicking needs to be performed at the 2^{nd}, 3^{rd} and 8^{th} positions (Figure 2A). Note that recording the bit '0' does not require any reactions, as it corresponds to the "no nick" ground state. Therefore, nick-based recording effectively reduces the size of the file to be actually recorded by half. This property of native DNA storage resembles that of compact disk (CD) and other recorders.

As the writing tool, we needed to choose a nicking enzyme with optimized programmability and nicking activity. Nicking endonucleases (natural/engineered) are only able to detect specific sequences in DNA strands; they can bind certain nucleotide sequences. Also, *Streptococcus pyogenes* Cas9 nickase (*Sp*Cas9n), as a widely used tool for genetic engineering applications, requires the presence of a protospacer adjacent motif (PAM) sequence (NGG) at the 3' site of the target DNA. The NGG motif constraint limits the nicking space to 1/16 of the available positions. The *Sp*Cas9n complex uses RNA guides (gRNAs) to bind the target, which makes it unstable and hard to handle. Furthermore, *Sp*Cas9n is a single turnover enzyme (15), i.e., one molecule of the enzyme can generate one nick per DNA molecule only. These make *Sp*Cas9n exhibit low efficiency and versatility for storage applications. To address these problems, we used the programmable restriction enzyme *Pf*Ago (12) as our writing tool. *Pf*Ago has significantly larger flexibility in double-stranded DNA cleaving than *Sp*Cas9n and, most importantly, has a high

turnover rate (one enzyme molecule can be used to create a large number of nicks). *Pf*Ago also uses 16 nt 5'-phosphorylated DNA guides (gDNAs) that are more stable and easier to handle *in vitro*. We experimentally demonstrated that under proper reaction conditions, *Pf*Ago can successfully perform simultaneous nicking of multiple prescribed sites with high efficiency and precision within 40 min. A comparison of the nicking performance of *Sp*Cas9n and *Pf*Ago may be found in Table S2 and Figure S3-4.

To facilitate writing multiple user files in parallel, we designed *Pf*Ago guides for all ten nicking positions in the chosen register and created registers bearing all 2^{10} = 1024 nicking combinations (Table S3). Registers were placed in microplates in an order dictated by the content to be encoded. The recording protocols for orthogonal registers, nick placements on both the sense and antisense strands and combinatorial mixing via group testing are described in the Supplementary Information.

Since the length, sequence composition and nicking sites of a register are all known beforehand, reading amounts to detecting the positions of the nicks. The nicked registers are first denatured, resulting in ssDNA fragments of variable length dictated by the nicked positions. These length-modulated ssDNA fragments are subsequently converted into a dsDNA library, sequenced on Illumina MiSeq, and the resulting reads are aligned to the known reference register sequence. The positions of the nicks are determined based on read coverage analysis, the insert size distributions and through alignment with the reference sequence; potential nicking sites that are not covered are declared to be '0's (Figure 2A-C).

We report write-read results for a 272-word file of size 0.4 KB containing Lincoln's Gettysburg Address (LGA) and a JPEG image of the Lincoln Memorial of size 14 KB (Figure S6). Both files were compressed and converted into ASCII and retrieved with perfect accuracy. Given the inherent redundancy of the sequencing process and the careful selection of the nicking sites and register sequences, no error-correction redundancy was needed (Figure 2B, C and Figure S5B-D). Technical details regarding implementations with orthogonal registers and with nicks on both DNA strands are provided in the Supplementary Information.

A potentially more efficient, portable and cost-effective approach to read the nicked DNA registers is via two-dimensional (2D) solid-state nanopore membranes. One approach is to use toeholds, short single-stranded regions on dsDNA created through two closely placed nicks, instead of single nicks. Toeholds can be accurately read using solid-state SiN_x and MoS_2

nanopores, as recently reported in (13). The cost of creating toeholds is twice as high as that of nicks, since one needs two different nicking guides. To mitigate this problem, one may attempt to detect nicks directly. To illustrate the feasibility of this approach, we performed Molecular Dynamics (MD) simulations based on quantum transport calculations. These revealed a strong inverse correlation between the ionic and electronic sheet current signals along the membrane induced by nicks in graphene and MoS_2 nanopores (Figure 2D, Figures S12-S14 & Supplementary Videos 1-5). The regions of strong negative "extremal" correlations between the ionic current and transverse sheet conductance strongly associate with the positions of the nicks.

In addition to allowing for nanopore-based reading, toeholds also enable complex in-memory computations. Toehold-mediated DNA strand displacement is a versatile tool for engineering dynamic molecular systems and performing molecular computations (16). Information is processed through releasing strands in a controlled fashion, with toeholds serving as initiation sites to which these input strands bind to displace a previously bound output strand.

Toeholds are usually generated by binding two regions of synthetic ssDNA and leaving a short fragment unbound. However, with *Pf*Ago, one can easily create toeholds in native DNA. To form a toehold, two nicks are generated within 14 bps. Under appropriate buffer and temperature conditions, in a single reaction the 14 nt strand between the two nicks disassociates, leaving a toehold on the double-stranded DNA (Figure S15).

Fluorescence-based methods can detect the existence of a toehold and the concentration of registers bearing a toehold without modifying the DNA registers. We illustrate this process on a register encoding 0010000000, with a toehold of length 14 nts at the nicking position 3. As shown in Figure 3A, a fluorophore and quencher labelled reporter strand with a sequence complementary to the toehold can hybridize to the toehold segment, producing a fluorescence signal resulting from an increase of the distance between the fluorophore and the quencher. We were also able to reliably measure different ratios of native DNA fragments with and without toeholds within 20 mins (Figure 3B). Since the reporter has a short single stranded overhang, it can be pulled off from the register upon hybridization, making the readout process non-destructive (Polyacrylamide gel electrophoresis analysis, Figure 3C). This feature equips our proposed storage system with unique nondestructive bitwise random access, since one is able to design specific reporters to detect any desired toehold sequence which accompanies a nick. It also enables computations on data encoded in nicks, as described in two recent papers (19,20).

In summary, by reprogramming *Pyrococcus furiosus* Argonaute as a universal nickase and using *E. coli* native DNA sequences, *we* have implemented the first DNA-based storage

system that mitigates the use of costly long synthetic DNA strands for storing user information. Our platform utilizes a parallel writing mechanism that combines an inexpensive nicking enzyme and a small number of short and inexpensive synthetic DNA guides. In addition, this approach enables enzyme driven toehold creation, allowing for bitwise random access and in memory computing via strand displacement.

Nick-based storage outperforms known synthetic DNA technologies in all relevant performance categories except for recording density; but the roughly one order of magnitude loss is insignificant for a system that already compacts petabytes in grams and overcompensated by the four-fold reduction of cost in our proposed system (Table 1; also, see Supplementary Information; Section C.10.). It also allows for cost-efficient scaling as a) long registers and mixtures of orthogonal registers may be nicked simultaneously; b) most uncompressed data files do not contain all possible 10-mers or compositions of orthogonal *k*-mers; c) genomic DNA and *Pf*Ago, as the writing tool, are readily available, and the mass of the created DNA products by far exceeds that of synthetic DNA, significantly increasing the number of readout cycles with NGS devices. This storage system may also be used to superimpose, erase and rewrite categorical and metadata on synthetic DNA oligos, in which case bitwise random access enables efficient non-destructive search and concentration sensing.

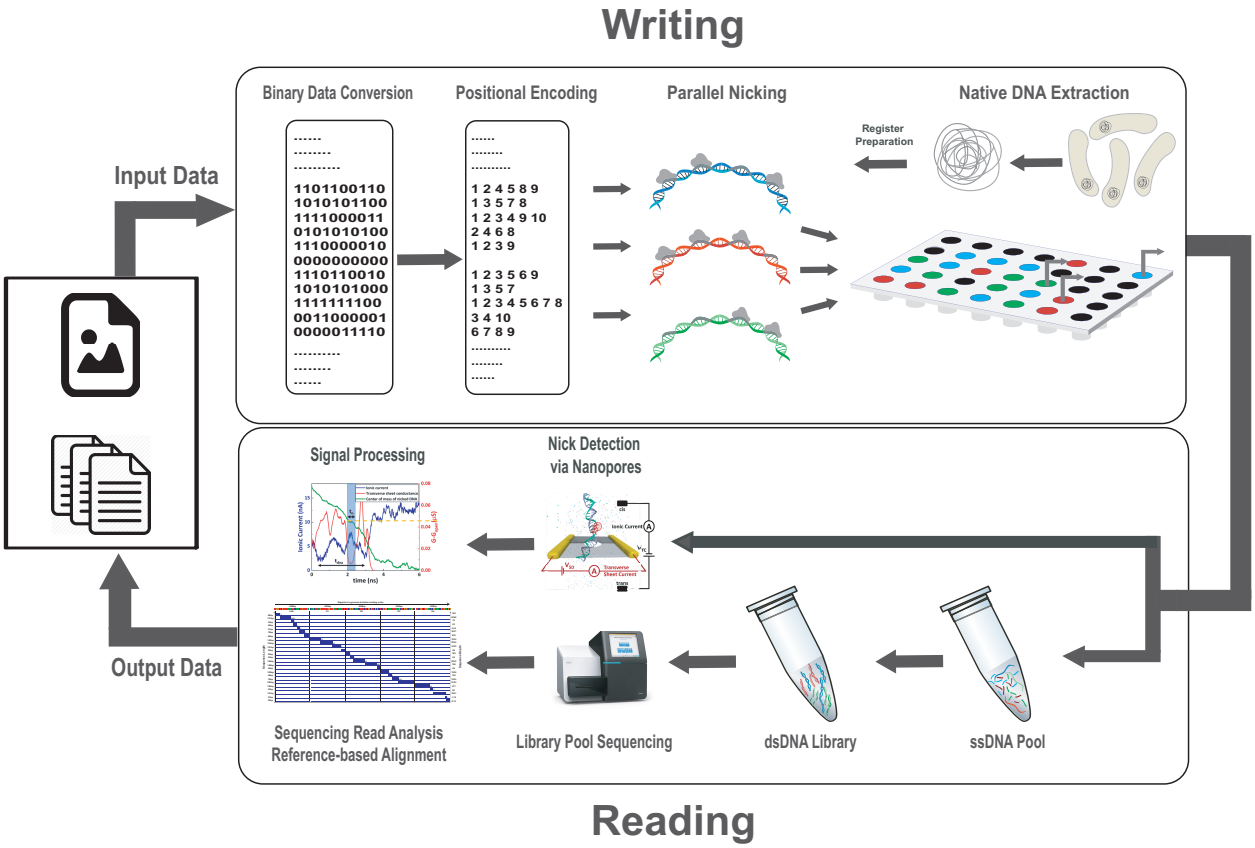
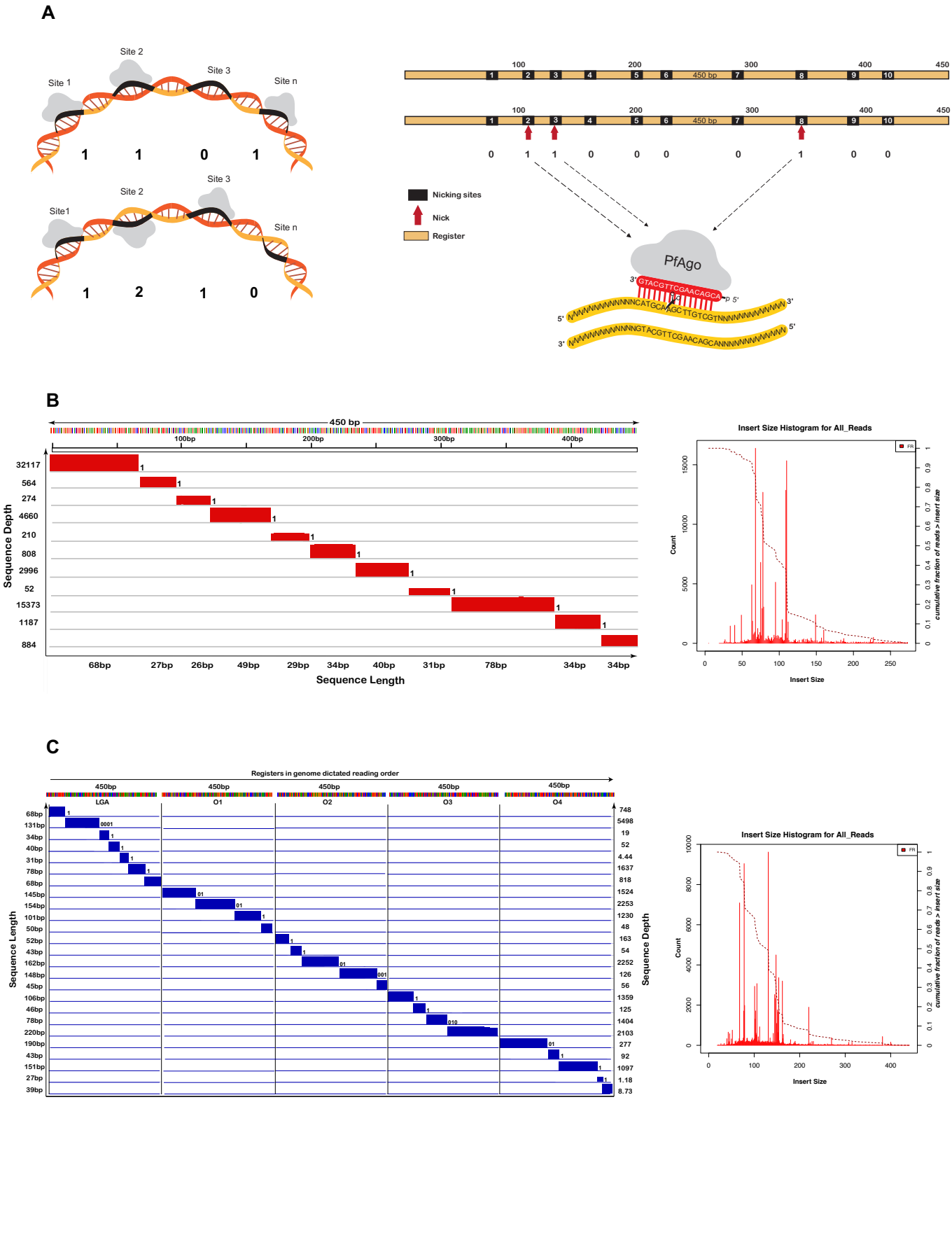


Figure 1 | **The native DNA-based data storage platform**. In the **Write component**, arbitrary user content is converted into a binary message. The message is then parsed into blocks of *m* bits, where *m* corresponds to the number of nicking positions on the register (for the running example, *m* = 10). Subsequently, binary information is translated into positional information indicating where to nick. Nicking reactions are performed in parallel via combinations of *Pf*Ago and guides. In the **Read component**, nicked products are purified and denatured to obtain a pool of ssDNAs of different lengths. The pool of ssDNAs is sequenced via MiSeq. The output reads are processed by first performing reference-based alignment of the reads, and then using read coverage to determine the nicked positions.



D

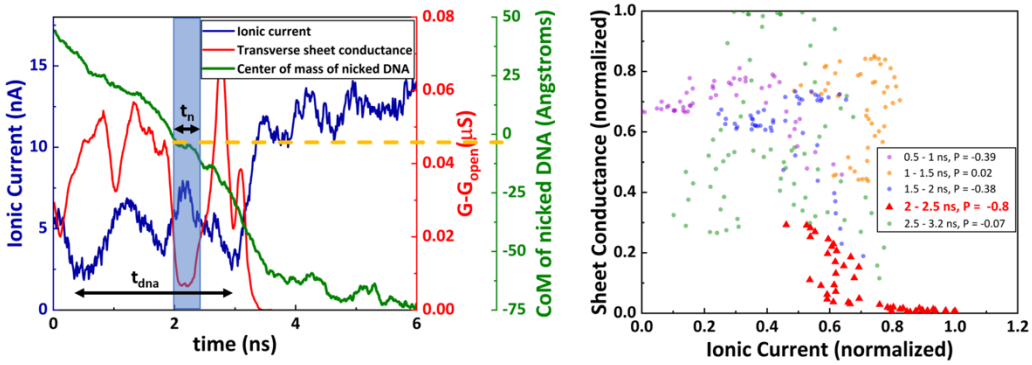


Figure 2 | Writing and reading the encoded data. **A**) *Pf*Ago can nick several pre-designated locations on only one strand (left, **top**) or both strands (left, **bottom**), simultaneously. In the first register, the stored content is 110…1, while in the second register, the content is 1210…0. The chosen register is a PCR product of a 450 bp *E. coli* genomic DNA fragment with 10 pre-designated non-uniformly spaced nicking positions (right, **top**). The positional code 238 corresponds to the binary vector 0110000100 (right**, bottom**). **B**) The MiSeq sequencing reads were aligned to the reference register to determine the positions of the nicks. The size distribution histogram (right) and coverage plots (left) are then generated based on the frequency and coverage depth of the reads. Coverage plots allow for straightforward detection of nicked and unnicked sites. In the example shown, all the ten positions were nicked, resulting in eleven aligned fragments. **C**) Five orthogonal registers used instead of one single register. Each vertical section represents one register in genome dictated reading order, and each row shows the read lengths retrieved after sequencing analysis. Read lengths are recorded on the left and sequencing depths on the right axis. **D) Reading via solid-state nanopores**. Plot of the calculated ionic current (blue), differential transverse sheet conductance (red) and center of mass of the nicked site (green) versus time (t_{dna} represents the translocation time of the entire dsDNA, while t_n represents the dwell time of the nick in the pore (left)). Scatter plot of normalized sheet conductance versus normalized ionic current over t_{dna} (right). A strong inverse correlation with P = -0.8 between the ionic current and transverse sheet conductance is observed at 2 – 2.5 ns, during which time the ion current reaches its global maximum, while the sheet current reaches its global minimum in t_{dna}. This time interval corresponds to the nick translocation event.

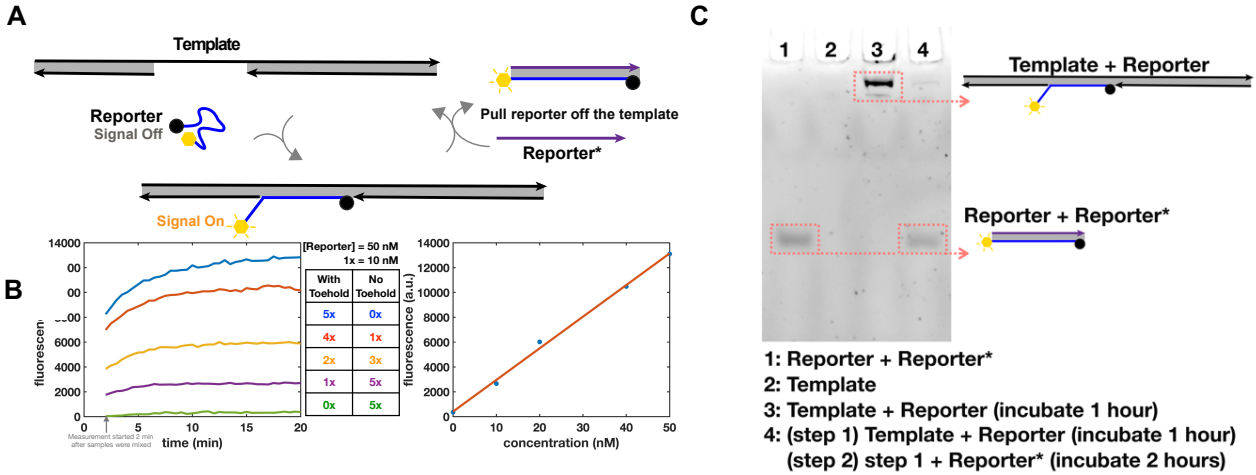


Figure 3 | **Non-destructive detection of a toehold**. **A)** Non-destructive detection of toeholds through a fluorophore and quencher labelled Reporter strand. Once the Reporter hybridizes with the toehold on the register strand, a fluorescence signal is observed due to the increase of the distance between the fluorophore and quencher. The Reporter strand can be pulled off from the register once the Reporter* strand hybridizes with the Reporter. **B)** Kinetics of detecting the concentrations of registers with and without toeholds in a mixtures **(left).** The fluorescence signals saturate within 20 minutes. The samples were mixed no more than 2 min before measurement. The concentration of toehold-ed DNA can be accurately quantified through fluorescence intensity **(right),** as it increases linearly with the concentration of the registers with toehold. **C)** PAGE gel results for non-destructive detection of a toehold. The gel was not stained with other fluorescence dyes, thus only the species with self-fluorescence is observed. After adding the Reporter, a large size complex appears in lane 3, indicating hybridization of the Reporter and the register. After the Reporter* is added, as seen in lane 4, the large size complex in lane 3 no longer exhibits self-fluorescence, indicating that the Reporter strand is pulled off from the register.

Table 1 | Comparison of synthetic and native DNA-based data storage platforms. Native DNA-based platforms outperform synthetic DNA-based approaches in all performance categories, except for storage density.

DNA-based Storage Method	Price per Bit Replica	Writing Latency	Reading Latency	Enables Computation?	Bit-wise Random Access	Maximum achievable physical Density	Information Density	(Optimal) Coding Loss (10)
Synthesis - based (1-11)	>$0.06 <$0.12 (10)	Sequential de novo synthesis/ Hours	NGS/hours	×	×	200 Ebytes/g (9)	< 2 bits/bp (to account for coding loss, usually ~1.5 bits/bp)	21% (9,10)
This work	$<\$1.5 \times 10^{-6}$	Parallel Nicking/ < 40 min	NGS followed by reference alignment/ hours	✓	✓	4 Ebytes/g	0.036 bits/bp	0%

References

1. G. M.Church, Y.Gao, S. Kosuri. *Science* **337,** 1628-1628 (2012).

2. N. Goldman *et al. Nature* **494,** 77-80 (2013).

3. S. H. T. Yazdi, Y.Yuan, J. Ma, H. Zhao, O. Milenkovic. *Sci. Rep.* **5,** 14138 (2015).

4. R. N. Grass, R. Heckel, M. Puddu, D. Paunescu, W. J. Stark, *Angew. Chem. Int. Ed.* **54,** 2552–2555 (2015).

5. S. H. T.Yazdi, R. Gabrys, and O. Milenkovic. *Scientific reports* 7.1 (2017).

6. S.L. Shipman, J. Nivala, J.D. Macklis, G.M. Church. *Nature* **547,** 345–349 (2017).

7. O. Milenkovic, R. Gabrys, H.M. Kiah, S.H.T. Yazdi. *IEEE Spectrum*, *55*(5), 40-45 (2018).

8. V. Zhirnov, R.M. Zadegan, G.S. Sandhu, G.M. Church, W.L. Hughes. *Nat. Mater.* **15,** 366-370 (2016).

9. Y. Erlich, D. Zielinski, D. *Science*, **355**, 950-954 (2017).

10. S.H.T. Yazdi, H.M. Kiah, E. Ruiz-Garcia, J. Ma, H. Zhao, O. Milenkovic. *IEEE Transactions on Molecular, Biological and Multi-Scale Communications,* **1, 3,** 230-248, (2015).

11. C. Laure, D. Karamessini, O. Milenkovic, L. Charles, J.F. Lutz. *Angewandte Chemie International Edition*, ***55***(36), pp.10722-10725 (2016).

12. B. Enghiad, H. Zhao, ACS Synth. Biol., **6**, 752−757 (2017).

13. K. Liu, C. Pan, A. Kuhn, A.P. Nievergelt, G. Fantner, O. Milenkovic, A. Radenovic. *Nature Communications*, **10**, 3 (2019).

14. S. Palluk, D.H. Arlow, T. De Rond, S. Barthel, J.S. Kang, R. Bector,H.M. Baghdassarian, A.N. Truong, P.W. Kim, A.K. Singh, N.J. Hillson, J.D. Keasling. *Nat Biotechnol.*, **36**, 645-650 (2018).

15. C. Andres, M. Jinek. *Methods Enzymol.* **546**, 1-20 (2016)

16. B. Yurke, A.J. Turberfield, A.P. Mills, F.C. Simmel, J.L. Neumann. *Nature*, **406**:605–608 (2000).

17. D.Y. Zhang, G. Seelig. *Nat Chem.,* **3**:103–113 (2011).

18. B. Wang, C. Thachuk, A. Ellington, E. Winfree, D. Soloveichik. *PNAS*, **115** (52), E12182-E12191 (2018).

19. B. Wang, C. Chalk, D. Soloveichik, *DNA 25 Conference,* Seattle, WA, U.S.A. (2019).

20. T. Chen, M. Riedel, *11th International Workshop on Bio-Design Automation (IWBDA),* Cambridge, England, U.K. (2019).