

Supplementary Information for

Digital data storage on DNA tape using CRISPR base editors

Afsaneh Sadremomtaz¹, Robert F. Glass², Jorge Eduardo Guerrero¹, Dennis R. Lajeunesse²,
Eric A. Josephs^{2*} and Reza Zadegan^{1*}

1. Department of Nanoengineering, Joint School of Nanoscience and Nanoengineering, NC A&T State University, Greensboro, NC, USA

2. Department of Nanoscience, Joint School of Nanoscience and Nanoengineering, UNC Greensboro, Greensboro, NC, USA

¹Department of Nanoengineering, NC A&T State University, Greensboro, NC, USA, ²Department of Nanoscience, UNC Greensboro, Greensboro, NC, USA, ³Joint School of Nanoscience and Nanoengineering, Greensboro, NC, USA

Table of Contents

S0. General materials.....	3
S1. The calculation of the estimated cost of digital data storage using de novo DNA synthesis....	3
S2. Mechanism of DMOS writer system.....	4
S3. Cleavage efficiency of gRNA to DMOS bits.....	6
S4. Designing address schemes of DMOS registers.....	8
S5. The orthogonality of DMOS writer system.....	10
S6. The called bits.....	11
S7. Software development.....	15
S8. The error-recovery simulation	15

S0. General materials

The DNA barcode primers, 5' phosphate modified reverse primer, 5' phosphorothioate-bond modified forward primer, and gRNA synthesis templates were purchased from Integrated DNA Technologies (IDT; Coralville, IA) and received as dry pellets. The primer oligos were dissolved in nuclease-free H₂O up to a concentration of 100 μ M and stored at -20 C for future use. The DMOS blank tape and the registers (Table S5) used for the school logo and title encoding experiments were purchased from TWIST Bioscience (San Francisco, CA) as dry pellets, resuspended in 0.1X TE buffer from Thermo Fisher Scientific (Waltham, Massachusetts) and stored at -20 C.

The FastDigest EcoRI restriction enzyme (catalog number FD0274) and FastDigest Bsu15I restriction enzyme (catalog number FD0144) were purchased from Thermo Fisher Scientific (Waltham, Massachusetts), and the PBR322 plasmid (catalog# N3033S) was purchased from New England Biolabs (Ipswich, MA) and stored at -20 C. The high-efficiency NEB 5-alpha Competent E. coli cells (catalog# C2987H) were purchased from New England Biolabs (Ipswich, MA) and stored in -80 C. Cloning results were confirmed via Sanger Sequencing by Azenta (South Plainfield, NJ).

The dCas9 (catalog number 1081058) and Cas9 proteins (catalog number 1081066) were purchased from Integrated DNA Technologies (IDT; Coralville, IA). The enGen sgRNA synthesis kit (catalog number E3322), lambda exonuclease (catalog number M0262S), and Q5U Polymerase master mix (catalog number M0597) were all purchased from New England Biolabs (NEB, Ipswich, MA). The APOBEC3A enzyme and BSA buffer were extracted from the NEBNext Enzymatic Methyl-seq Conversion Module (catalog number E7125) purchased from New England Biolabs (NEB; Ipswich, MA). The AMPure XP magnetic particles (catalog number A63881) were purchased in 60 mL volumes from Beckman Coulter (Indianapolis, ID).

S1. The calculation of the estimated cost of digital data storage using de novo DNA synthesis

Based on our calculations, to store 5 minutes of a 1080p YouTube video stream in commercially acquired DNA costs over 7 million dollars, consumes over 100KWh of energy, takes over 4 days, and produces over 15 liters of toxic waste.

The rough size of Youtube HD videos with a resolution of 1080p, based on the video file-size calculator <http://toolstudio.io>, is 160 MB. On another side, Integrated DNA Technologies (IDT) reported the cost of synthesizing one base pair of oligo as \$0.011.

This information provided the possible cost calculation of storing this file in DNA memory using de novo DNA synthesis.

$$160 \text{ MB} = 1.34 \times 10^9 \text{ bits}$$

(SE1)

If each nucleotide stores 2 Bits, 1.34×10^9 bits could be stored in 671,088,640 bp.

The price of storing 5 minutes of a 1080p YouTube video stream is calculated as follows:

$$671,088,640 \text{ bp} \times \$0.011 = \$7,381,975.04$$

(SE2)

S2. Mechanism of DMOS writer and rewriter system

DMOS uses a programmed machinery base-editing tool to write the digital data in the form of sequence edits at the precise locations of DMOS registers. To write the data on DMOS bits, we used a DMOS writer system composed of two enzymes: cytidine deaminase mutagenic protein "APOBEC3A" and a mutant form of Cas9 known as dead Cas9 (dCas9). dCas9 is accompanied by a programmed machinery system of gRNA and creates Ribonucleoprotein (RNP). Each DMOS bit consists of a 23bp-long data 'state' section and a unique 40 bp 'index' sequence (Fig. 1).

During the base-editing reaction, the CRISPR effector Cas9 recognizes a 3 bp portion of state defined as a "protospacer adjacent motif, "; resulting in binding to 20bp-long state section within a double-stranded DNA molecule. Next, the 20 bp corresponding gRNA sequence, known as "spacer", recognizes the state section (Fig.1 and S1) and creates a nucleotide "R-loop" structure at the target positions within DMOS bits (Fig.1, S1, S2, S3). The DMOS writer system selectively programs the mutations in the form of sequence edits across the DMOS registers during the writing process. We encode the new data into the existing DNA bits by editing the sequence in the DNA bits through activation or deactivation of the writer system. The writer system uses a cytidine-deaminase that chemically converts deoxycytosine (dC) to deoxyuracil (dU). Next, dU is converted to deoxythymine (dT) through the active process of writing using Q5U polymerase (Fig.S1A). The product of this process is a DNA molecule that contains dT bases instead of dC bases at target regions; as a result, the state of each domain change from intact to mutated. As a further step, we digested the unmodified nucleotide backbone with lambda exonuclease after mutation to increase the likelihood of the modified register being amplified. After exonuclease digestion, we used Q5U polymerase to amplify the remaining modified register. We selected Q5U polymerase to amplify our modified registers, reading the uracil-containing registers in nanopore sequencing reads.

DMOS registers amplified with a reverse 5'-phosphorylated primer, that showed with the red dot in Fig. S1A. This modification enables the lambda exonuclease to degrade the unmodified strand of our register, leaving the encoded strand as a register for PCR amplification. Finally, we observe a significant presence of dT in each modified register, indicating successful mutations in our selected bits with no footprint of cross-reactivity (Fig. S1A).

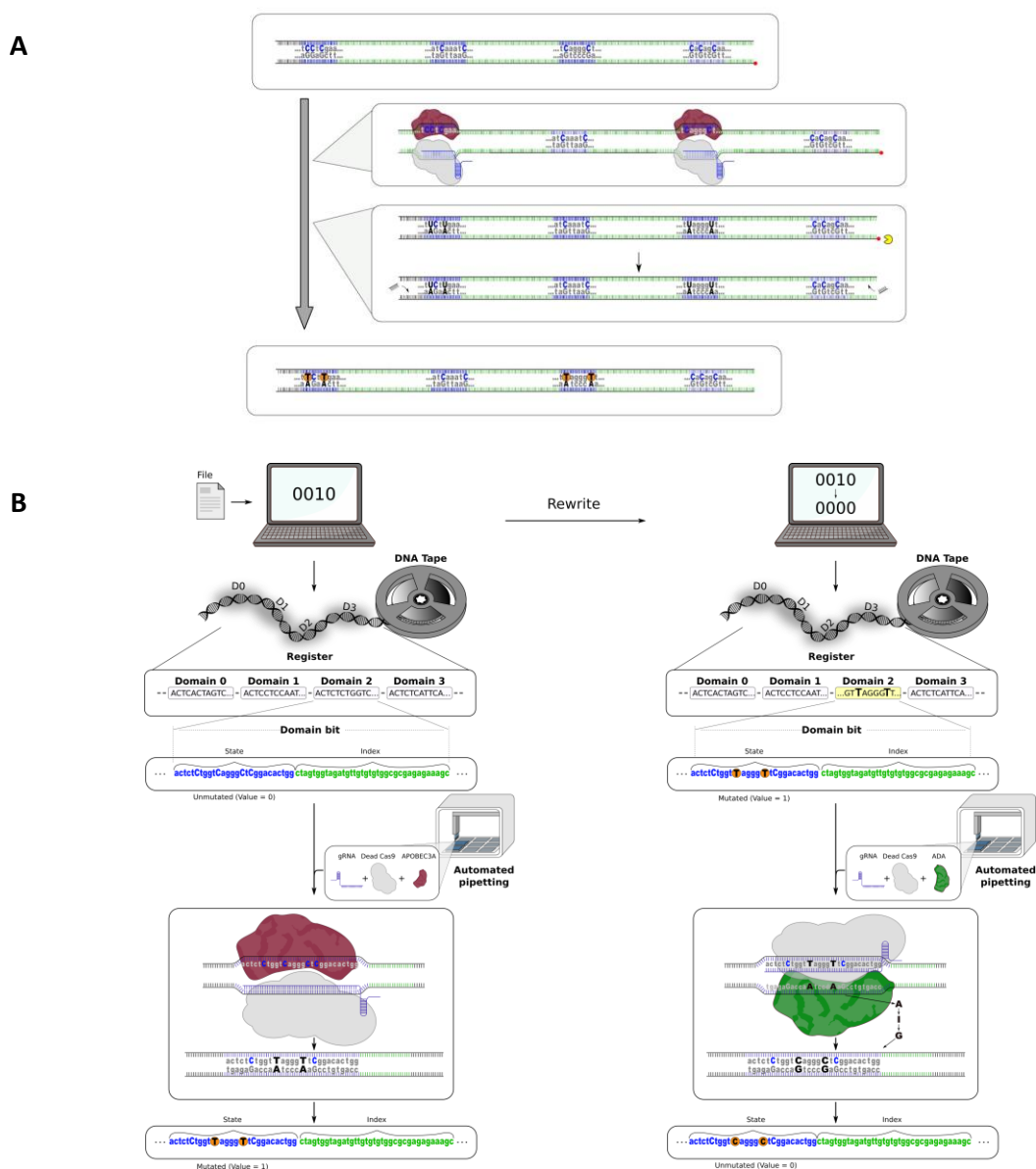


Figure S1. DMOS enzyme encoding protocol. A) A combination of RNPs target the DMOS register at the desired positions to write the data via a programmable APOBEC3A-driven mutation reaction. The modified register addressed to lambda exonuclease digestion (yellow) and amplified in the active process of writing using Q5U polymerase, and read the uracil nucleotides as thymines in the final sequencing read. B) Potential DMOS rewriting mechanism to convert the A:T to G:C.

S3. Cleavage efficiency of gRNA to DMOS bits

To visualize the cleavage efficiency of the spacer sequences of gRNA, we used the active Cas9 proteins and confirmed the targeting of the writer system at the precise location of their complementary domains (Fig. S2). Unlike the dCas9 proteins used in our writing system, the active Cas9 proteins will cleave the DMOS tape at complementary regions. Each digestion splits the tape into two fragments, indicating that the spacer sequences of gRNA show high binding efficiency to the respective state. Therefore, it demonstrates that the gRNA will accord the bits to the R-loop conformation required for APOBEC3A to recognize and deaminate the nucleotides within the loop. We validated the results of digested tape fragments on 2% agarose gel to determine the length of the fragments compared to the control 1108 bp template labeled "C" in the Fig. S2. As shown in Fig. S2, we observed bands of a smaller length relative to the negative DMOS register control. This result indicates that digestion only occurs in precise locations as a function of Cas9. We also observed an "X" pattern across all digestion sites consistent with where we expect the Cas9 protein to cut, showing likely no significant off-target cleavage across the register. This result indicates that the targeting of bits occurs only at the precise location of interest.

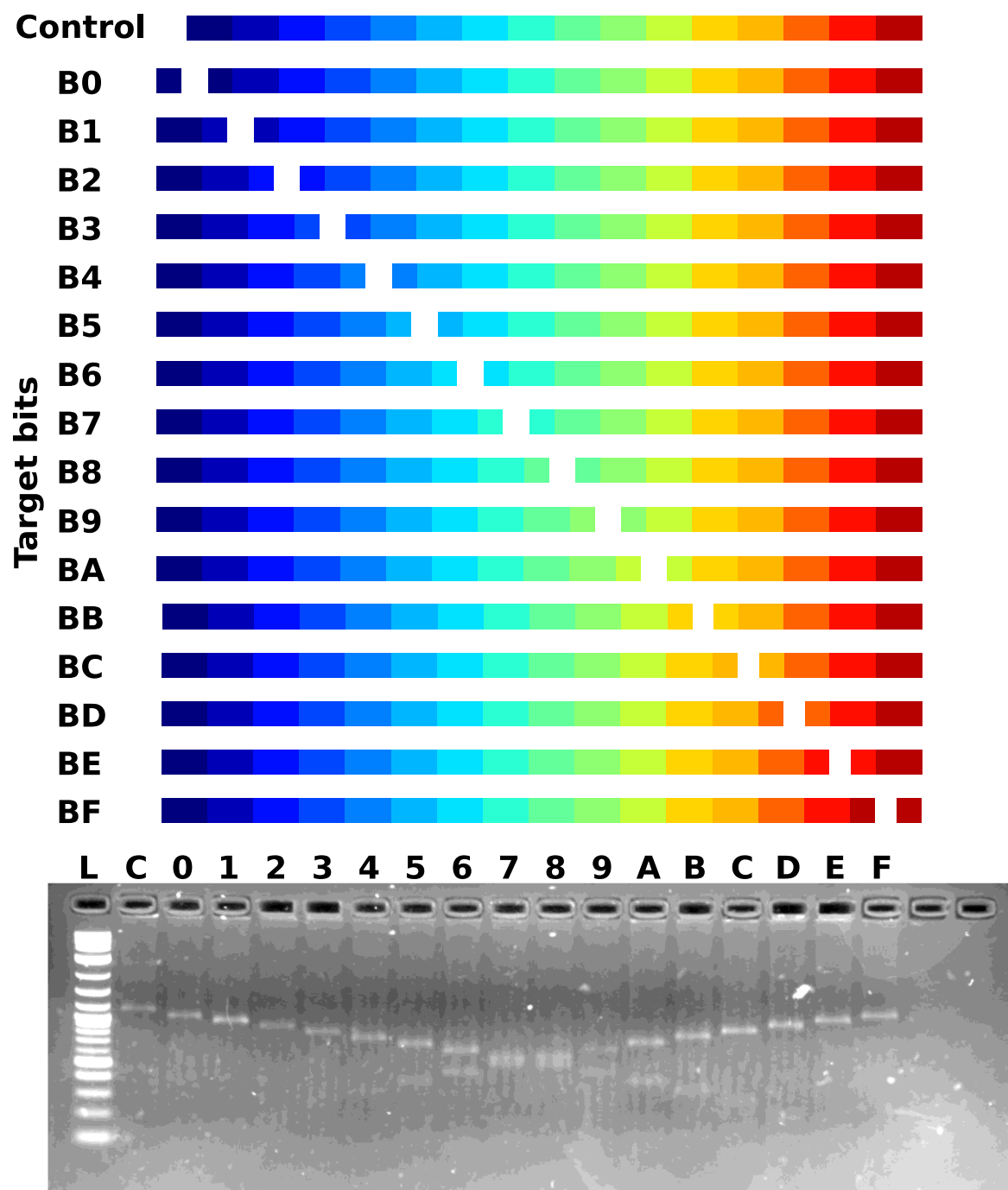


Figure S2. Schematic of efficiency of gRNA to DMOS bits. The efficacy of the writer system was evaluated in the form of Cas9 cleavage activity at the desired bit positions. The gel results confirm that there is no evidence of off-target cleavage, showing the high efficiency of dCas9 for our writing system.

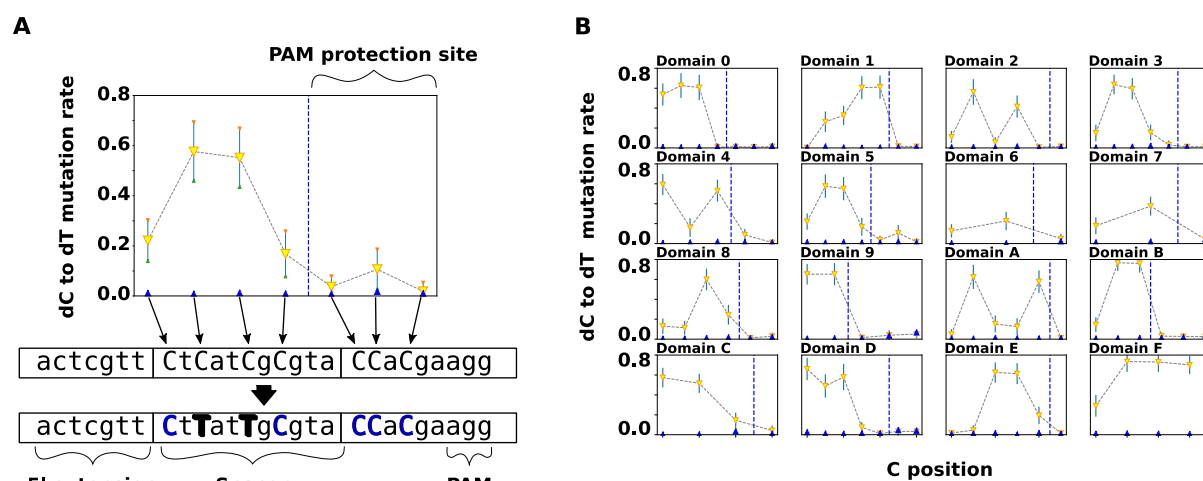


Figure S3. dC to dT orthogonality analysis: A) Schematic view of mutational signature for one DMOS bit. A mutational signature is depicted as conversion dC's to dT's across the entire 'state' section of a DMOS bit during the base editing reaction. The evidence of mutation appears through changing the state of the bit from '0' (unmutated) to '1' (mutated). The DMOS algorithm deterministically programs the bits and stores the data in the domain states by writing the data on the DMOS bits and making mutations more efficient. To improve the design of DMOS domains, iterative trinucleotides, indicated here as TCA or TCG, were placed at least six nucleotides upstream of the PAM site. B) In vitro APOBEC3A-driven mutational signature of dCs positions tracked across entire DMOS bits (16 domains of DNA register) as a function of dC-to-dT mutation rate (%). Conversion rates showed significant increases up to 80% when the corresponding dCas9 was utilized. Dash line indicated the PAM protection site and number of Cs located in this region. Data are represented as average values $\pm$ SD. Source data are provided as a Source Data file.

S4. Designing address schemes of DMOS registers.

To produce a set of differentiable DMOS registers (Fig. 2A), we initially used the DMOS addressing model as a "Barcode-based addressing scheme" using 36 random barcodes that identify one register from another in the pool (Fig. 2B). In this system, the number of barcodes informs the maximum capacity. To increase data storage capacity, we defined a unique shuffling (permutation) addressing model for the domain position orders (bits) in each individual DMOS register (Fig. 2B), allowing the generation of up to 2.09×10^{13} different DMOS registers with high-entropy of permutations (Fig. 2B). The order of DMOS bits determines the "DMOS register address" map. DMOS bits permute independently in random order across the register to identify one register from another in the pool. We refer to a collection of DMOS registers in the pool as "DMOS block," or "DMOS tape" in this study, which contains a library of 48 DMOS registers. (Fig. 2A).

Thus, we defined a lexicographic representation of the permutation order (Fig. 2B), where we map the domain positions as B0 = 0, B1 = 1, ..., B15 = F. For instance, DMOS register addresses of 0, 1, and 2 represent 0123456789ABCDEF, 0123456789ABCED, and 0123456789ABDFCE, respectively. The maximum theoretical number of DMOS register combinations in one single tube is $\sim 2.09 \times 10^{13}$, corresponding to the upper limit for addressing locations. Therefore, the permutation addressing scheme enhanced the DMOS system's capacity. According to this scheme, the registers were shuffled only in the order of the last six domains. The issue with this approach is the minimum possibility of variations (low-entropy) between the order of consecutive register addresses. The low-entropy of this addressing scheme leads to miss-addressing and increases the computation cost (Fig. 2B).

To decrease computational cost during decoding, we suggest the High-entropy addressing scheme with the lexicographic representation. This method permutes the order of bit positions in the entire register (Fig. 2B). The High-entropy permutation addressing scheme ensures consecutive addresses in the DMOS block with insignificant correlation. Herein, we designed a library of DMOS registers in which the order of domains shuffled across entire registers. For example, the register addresses 0, 1, and 2 represent the combinations such as 915760E3CB2D84AF, 6A51BF32EC074D98, and BC85F6E3072A4D19, respectively. We hypothesized that the High-entropy addressing scheme reduces the noise and increases the confidence in our measurements (Fig. 2B).

In this study, we defined one DMOS block as a blank drive containing 48 DMOS registers, in which 32 registers use the High-entropy permutation addressing scheme, and the last 16 registers use the Low-entropy permutation addressing scheme (Fig. 2A).

S5. The orthogonality of the DMOS writer system

We introduce a DMOS tape, which uses a programmable, independent, and orthogonal writer system to encode digital data in sequence edits, enabling higher density than the conventional methods. We induced mutations at each bit to analyze the orthogonality and distribution of dC to dT conversions within each bit. We replicated this experiment across the registers, containing 32 different mutation reactions. We observed significant dC to dT conversion rates in multiple dC positions within the written bits. Each DMOS tape in our trials is addressed with barcodes to distinguish registers within the pool and nanopore sequencing run. We collected the dC and dT population data to compare the conversions across the registers from our sequencing (Fig. 2A).

Figure.S3 shows significant dC to dT conversion rates in multiple dC positions within the affected spacers. We also confirmed that dC nucleotides within 6 bp of the protospacer adjacent motif showed an insignificant conversion rate compared to the negative control (Fig. S3A). However, the DMOS uses a domain-based coding system that enables information storage in DNA domains rather than individual bases. Thus, we reduce the number of bit errors that may missed in the PAM protection sites within the DMOS bits (Fig. S3). This fact emphasizes the robustness of the methods. It bypasses the controversial issues of the current DNA-based memory error correction methods, excluding any bit strings containing insertions, deletions, and degraded information, then performing majority voting to recover the data.⁶ The orthogonality experiments have shown efficient conversion of dC to dT within the bits where we applied the corresponding dCas9 complex versus our negative control group. This result also demonstrated no off-target activity in any dCs outside our affected bits, confirming that our encoding strategy is independent, accurate, and precise in encoding digital information into the respective registers (Fig. S4A and B).

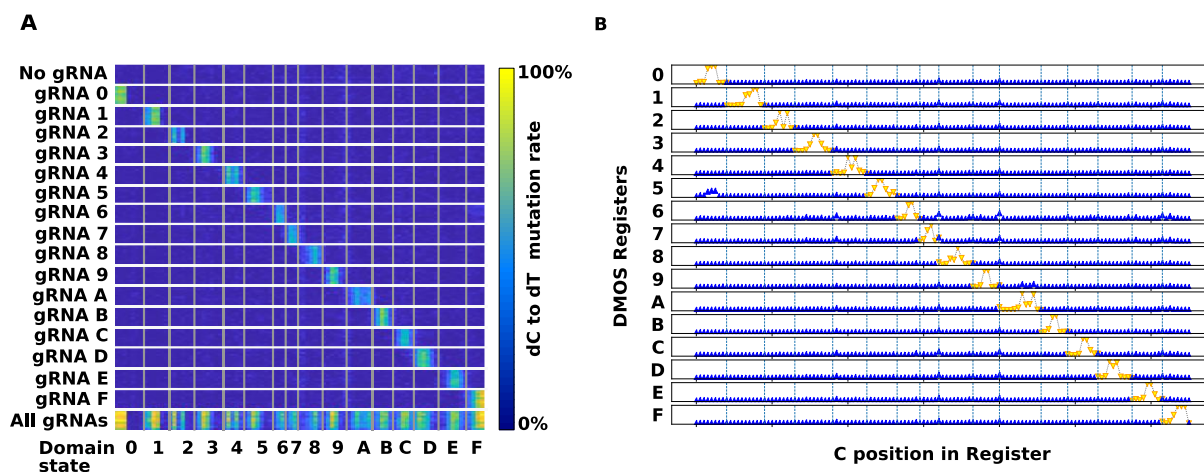


Figure S4. Nanopore sequencing records the orthogonality of DMOS writer system across the domains of DMOS registers.

A) Position-independency of dC to dT ratio determines the modularity of DMOS system to locate the edits at the precise locations across the DMOS registers with no footprint of cross-reactivity. B) Orthogonality analysis of DMOS writer enzyme to assign the cross-reactivity of APOBEC3A/CRISPR-dCas9 across all cytosines as a function of the neighbor bits.

S6. The called bits

We used a classification model to identify the DMOS register addresses and then mutational states accordingly (Fig. 4). The decoder was developed and written in C++. We first use the Smith-Waterman align algorithm to perform the alignment.⁴⁵ Next, the decoder takes the FASTA sequences of the domains B0, B1, ..., B15 and determines $(pn, sn, cigar) = \text{swalign}(B_n, \text{Seq})$. Where "p" is defined as positions, "s" as score, "n" as the number of positions [0,1,2,...,15], " B_n " is the nucleotide sequence of the n-th domain, and "Seq" is a single nucleotide sequence obtained from the nanopore. Thus, the algorithm generates a list of positions; $[p_0, p_1, \dots, p_{15}]$ and sorts them to determine the order of the domains in the DMOS register. Moreover, the decoder uses the Compact Idiosyncratic Gapped Alignment Report (CIGAR report) to generate a string of nucleotides aligned with the cytosines of the bit's states in the FASTA reference. Using the maximum likelihood, the decoder determined each domain's position and alignment score on the sequenced strand. Given the bit order, the decoder generates the lexicographic representation of the address and converts it to an integer number. A valid address number corresponds to our DMOS register's the address library. Whether the address is valid, the decoder determines the values of the mutational states. The DMOS decoder then determines the mutation state. Our DMOS decoder employs two Bayesian steps in a classification decision tree to accomplish this model.

The schematic view of our decoder is shown in Figure. 4, demonstrating how the first Bayesian classifier counts all Ts and Cs in the nucleotide sequence of the domain's states. We then accumulate these values for all reads of the same states. Finally, the values used to calculate the dC to dT ratio are as follows:

$$\text{dC to dT} = \text{sum}(T) / (\text{sum}(T) + \text{sum}(C)) \quad (\text{SE3})$$

We used dC to dT ratio curves to show the states' probability and represent them as unmutated or mutated. We found the regions that curves overlap, creating an uncertain region. Herein, we establish the dC to dT threshold values t_u , and t_m for unmutated and mutated domain-calls, respectively (Table S2). The classifier determines the state of the mutational state according to the following:

Unmutated:	$0 < \text{dC to dT} < t_u$,	CS = -10
Uncertain:	$t_u < \text{dC to dT} < t_m$,	CS = Use second classifier
Mutated:	$t_m < \text{dC to dT}$,	CS = 10

(SE4)

Where CS is the classification score (CS).

We assessed the called-bits based on the results of the first Bayesian classifier as to whether it determines the state as uncertain. The uncertain calls address further analysis under the next Bayesian classifier. Else, it will assign the CS values as -10 for unmutated and 10 for mutated.

Next, we performed a classification training process for each combination of the domain state's nucleotides according to the expected mutational state. As a result, we generated a library containing the score values per each combination to classify the domain's state as either unmutated or mutated. The second Bayesian classifier receives the scores from the library of the nucleotide string and the CIGAR score.

$$(S_u, S_m) = \text{CheckScores}(\text{nucleotide position}) \quad (\text{SE5})$$

Where S_u = unmutated score, S_m = mutated score, and iteratively adds the values to the classification score (CS), as follows:

$$\begin{aligned} u\text{Fraci} &= (S_u / (S_u + S_m)) \\ m\text{Fraci} &= (S_m / (S_u + S_m)) \\ \text{CS} &= \sum_{n=i \text{ to } k} (\log(m\text{Fraci}) - \log(u\text{Fraci})) \end{aligned} \quad (\text{SE6})$$

The CS values are in the range [-10, 10]. The Bayesian classifier determines the state of the mutational state according to the following:

$$\begin{aligned} \text{Unmutated:} & \quad \text{CS} < 0 \\ \text{Mutated:} & \quad \text{CS} \geq 0 \\ \text{Uncalled:} & \quad -5 < \text{CS} < 5 \end{aligned} \quad (\text{SE7})$$

The decoder uses the CS values to transduce the sequence of nucleotides to the corresponding binary values. The results represent binary 0 and 1 for negative and positive CS values and eliminate the uncalled.

The correctly-called-bits

To identify the correctly called bits, we first choose the set of Called bits as determined by the equation:

$$\text{Called-bits} = 100 * ((\text{TotalBits} - \text{UncalledBits}) / \text{TotalBits}) \% \quad (\text{SE8})$$

We then utilize a reference file to recognize the errors and subtract them from the set of Called bits:

$$\text{Correctly-called-bits} = 100 * ((\text{TotalBits} - \text{UncalledBits} - \text{Errors}) / \text{TotalBits}) \% \quad (\text{SE9})$$

The aforementioned equations are statistically evaluated using bootstrapping analysis of 250 iterations to produce graphs of Called-bits and correctly-called-bits and the curves were fitted using Origin (as shown in Fig. 3 and Fig. 4).

Our decoder employs to explore the applications for the tradeoff writing/reading recovery rate and determines whether a long read is required to label a particular state as mutated or unmutated. Hence, we configured our system to take snapshots every 100 intervals, generating the correctly-called-bits analysis as per the reads function in Figures 3 and 4. Depending on whether the DMOS encoder uses error correction, the Python script informs the LDPC decoder algorithm and records the retrieval of stored data. We next add error correction codewords to improve data storage's reliability. We used low-density parity-check (LDPC) codes that set the error threshold close to Shannon capacity limits. The code includes a pre-processing step that adds LDPC error correction to the file's binary contents.⁴¹⁻⁴⁴

The DMOS decoder was written in C++, including a Python interface to generate a real-time visualization of snapshots, and is available on GitHub.

(<https://github.com/SBML-LAB/DMOSEncoder>, <https://github.com/SBML-LAB/DMOSDecoder>).

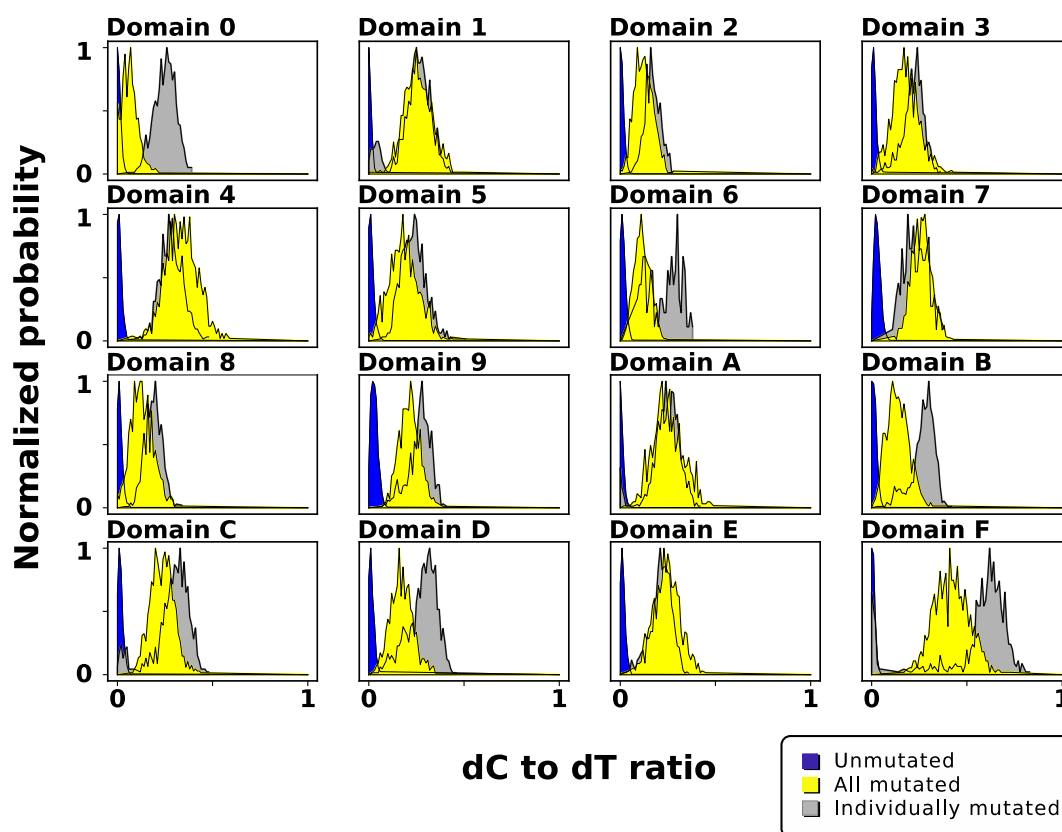


Figure S5. Normalized probability curves of dC to dT ratio to determine the mutation state at the DMOS bit states. Normalized probability distribution of individually mutated (gray) and all domains mutated (yellow) domains compared to unmutated domains (blue).

Curve fitting equation of the called-bits and the correctly-called-bits for Bitmap representation of our school's logo on DMOS tape:

The called bits:

$$y = A_1 e^{-reads/t_1} + A_2 e^{-reads/t_2} + y_0$$

$$A_1 = -54.7551$$

Supplementary Information

$$\begin{aligned}A_2 &= -35.14309 \\t_1 &= 625.6624 \\t_2 &= 3059.59351 \\y_0 &= 92.14322 \\R^2 &= 0.99966\end{aligned}\tag{SE10}$$

The correctly-called-bits:
 $y = A_1 e^{-\text{reads}/t_1} + A_2 e^{-\text{reads}/t_2} + y_0$

$$\begin{aligned}A_1 &= -57.49084 \\A_2 &= -34.94659 \\t_1 &= 633.75741 \\t_2 &= 3088.54435 \\y_0 &= 91.08862 \\R^2 &= 0.9996\end{aligned}\tag{SE11}$$

Curve fitting equation of the called-bits and the correctly-called-bits for encoding the title of this study on DMOS tape:

The called bits:

$$\begin{aligned}y &= A_1 e^{-\text{reads}/t_1} + A_2 e^{-\text{reads}/t_2} + y_0 \\A_1 &= -54.23359 \\A_2 &= -34.61551 \\t_1 &= 1373.51658 \\t_2 &= 6571.78245 \\y_0 &= 89.08068 \\R^2 &= 0.999995\end{aligned}\tag{SE12}$$

The correctly-called-bits:

$$\begin{aligned}y &= A_1 e^{-\text{reads}/t_1} + A_2 e^{-\text{reads}/t_2} + y_0 \\A_1 &= -55.06187 \\A_2 &= -35.42389 \\t_1 &= 1447.93899 \\t_2 &= 6596.35643 \\y_0 &= 86.06278 \\R^2 &= 0.99996\end{aligned}\tag{SE13}$$

S7. Design of LDPC codes for error protection

We used the Protograph LDPC library⁴³ (<https://github.com/shubhamchandak94/ProtographLDPC>), to design the LDPC codes for simulation and obtain our error-recovery model subsequently. We designed two types of LDPC codes: Protograph and Regular.

We modified the number of check nodes for the Regular LDPC codes to achieve redundancies ranging from 20% to 90%. For the Protograph LDPC codes, we selected the Protograph type accumulative repeat by 4 jagged accumulate (AR4JA) to define the Generator and Parity Check matrices, with a message-code ratio of 3/4, 2/3, and 1/2 to obtain 25%, 33%, and 50% redundancy, respectively. We noticed that the storage capacity of each LDPC code is inversely proportional to the amount of maximum redundancy overhead added. Our first demonstration of the DMOS writer system encodes 576 bits with 25% redundancy, while it encodes only 77 bits with 90% redundancy (Fig. S7). This result justifies the tradeoff writing/reading to achieve an industrially acceptable data recovery rate.

Protograph LDPC with 25% redundancy uses 576 bits per message (72 bytes) and 768 bits per codeword (96 bytes). We developed a Python script to communicate with the LDPC library that allows the conversion of the intermediate binary files for input/output and records the signals of the LDPC decoder (Fig. S7, S8).

S8. The error-recovery simulation

We ran an error-recovery simulation for different LDPC codes that calculates the decoder's recovery rate per number of errors. The simulator creates a random message according to the capacity of the LDPC codes and creates the codeword. We iteratively inject errors into the codeword and test if the decoder recovers the data. The simulator repeats this procedure 1000 times before increasing the number of errors and recording the chance of recovery per read (Fig.S7).

$$y=A2+(A1-A2)/(1+ \exp(\text{Errors}-x_0) /dx). \quad (\text{SE14})$$

Where y is Recovery rate (Errors)%

A1 = Initial value of Recovery rate when Errors is 0, A2 = Final value of Recovery rate when Errors is infinity,

Xo = number of Errors where Recovery rate (Xo) approximately reaches 50%, and dx = constant number which determines the slope of the curve close to Xo

This simulation creates the error-recovery rate model for different LDPC codes. Given the received errors from the experiments (Fig. 3 and 4), we add reads to a number of errors, feed the model, and represent the DMOS recovery rate (%) per read. Figures 3 and 4 show the error-recovery rate per reading length using the error-rate from the corresponding experiments. Figures S7 show the error-recovery rate per errors % for 1000 iterations by increasing the number of errors and recording the chance of recovery rate % per read. The decoder inform the LDPC codes can recover nearly 100% of the cases. Simulation results imply Protograph LDPC shows a better setting than Regular LDPC with a similar redundancy ratio.

In the Bitmap representation of our School's logo, Protograph LDPC with 25% redundancy showed being capable of recovering the data in nearly 5K reads. We selected this redundancy

for further experiments due to the higher number of bits stored in the codeword and assessed the recovery capabilities. Therefore, we encode this study's title using the 25% overhead added. We observed an increased error-rate that required long sequencing reads to achieve a valid recovery, as shown Figure 3. Given this study's error-rate, we simulated the different LDPC codes to determine the amount of required overhead. In good agreement with computational analysis, the experimental result confirmed that using Protograph LDPC with 25% redundancy, we recover the data after 100K (Fig. 4), with a 67% probability (Fig. S7). This result justifies the tradeoff between redundancy and data capacity. The simulation study offers the Protograph LDPC code with 50% redundancy and the best error recovery performance (data recovered at 3000 reads) while preserving a decent storage capacity for future works.¹⁹⁻²²

Curve fitting equation of the bootstrapping error-recovery simulation for Regular LDPC of Bitmap representation of our school's logo on DMOS tape:

$$y=y_0+A1*Read^n/K^n+Read^n \quad (SE15)$$

Where y_0 is statrt point of graph, and $A1$ is defined as End point of graph and K and N were used as a constant number for curve fitting:

Curve fitting equation of the bootstrapping error-recovery simulation for Protograph LDPC of our school's logo on DMOS tape: $y=y_0+A1*Read^n/K^n+Read^n$ (SE16)

Curve fitting equation of the bootstrapping error-recovery simulation for Regular LDPC of encoding the title of this study on DMOS tape: $y=y_0+A1*Read^n/K^n+Read^n$ (SE17)

Curve fitting equation of the bootstrapping error-recovery simulation for Protograph LDPC of encoding the title of this study on DMOS tape: $y=y_0+A1*Read^n/K^n+Read^n$ (SE18)

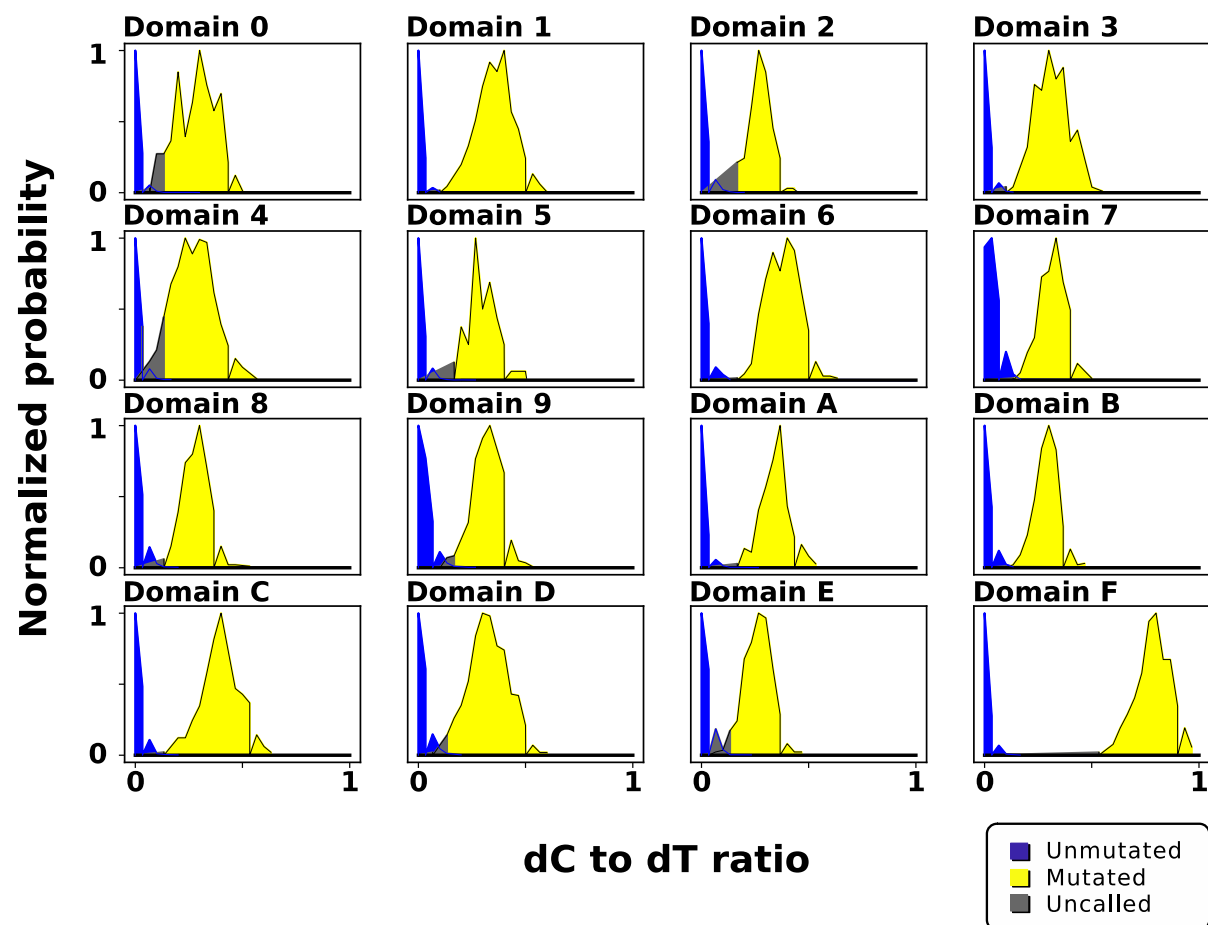


Figure. S6. Normalized probability curves of dC to dT ratio for each individual domain of DMOS registers at bitmap representation of our school's logo. Interception of the intact and mutated curves shows uncertainty in the classification decision tree analysis (threshold shows as gray), and the limits of the interception allow us to determine the threshold values using the first Bayesian classifier.

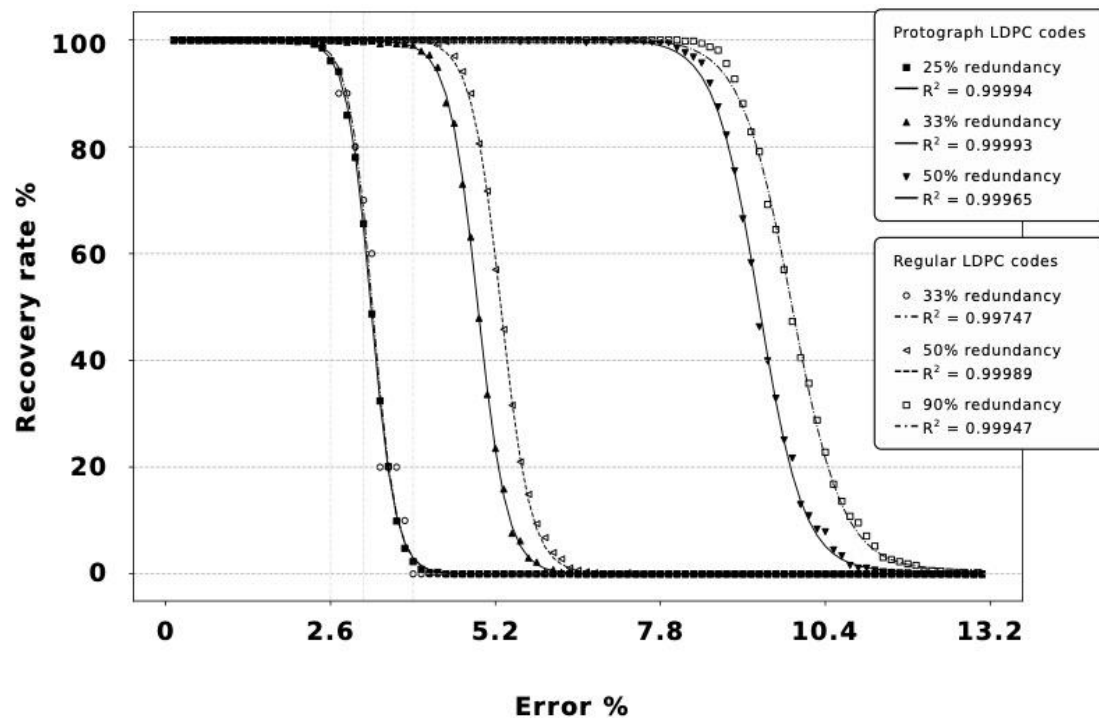


Figure. S7. DMOS error-recovery simulation model for different LDPC codes. The decoder calculates the recovery rate per number of errors for 1000 iterations by increasing the number of errors and recording the chance of recovery rate % per read. Data are represented as average values $\pm$ SD. Source data are provided as a Source Data file.

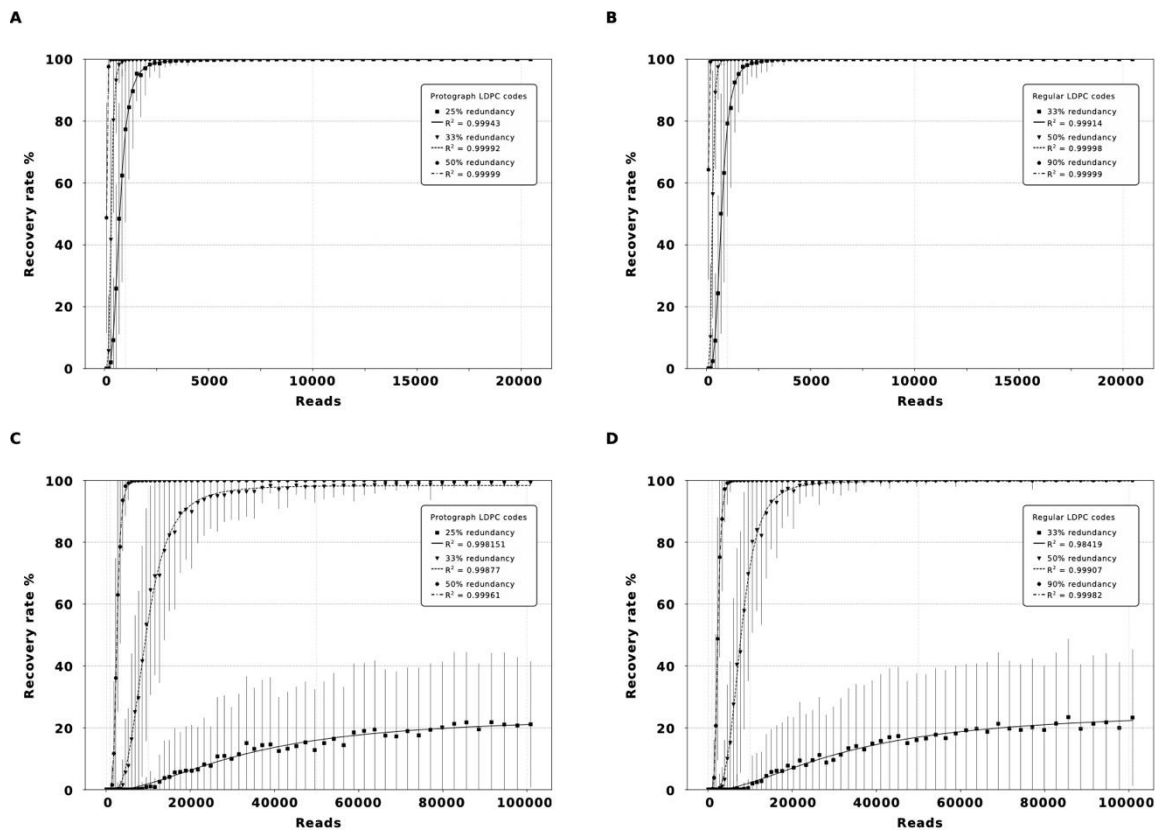


Figure S8. DMOS error-recovery simulation per reads. Error-recovery simulation for different LDPC codes using 1000 repetitions to obtain an error-recovery rate model for each decoder. Data recovery simulation for A) JSNN_Protograph, B) JSNN_Regular, C) Title of this study_Protograph, D) Title of this study_Regular. 3D representation of sequencing reads vs. Protograph LDPC for 33%, and 50% redundancy in the codeword per reads as a function of recovery rate (%). For the Protograph LDPC codes, we used a message-code ratio of 3/4, 2/3, and 1/2 to obtain 25%, 33%, and 50% redundancy, respectively, and 3D representation of sequencing reads vs. Regular LDPC redundancy ratio (%) for 33%-90% redundancy in the codeword as a function of recovery rate (%). Data are represented as average values $\pm$ SD. Source data are provided as a Source Data file.

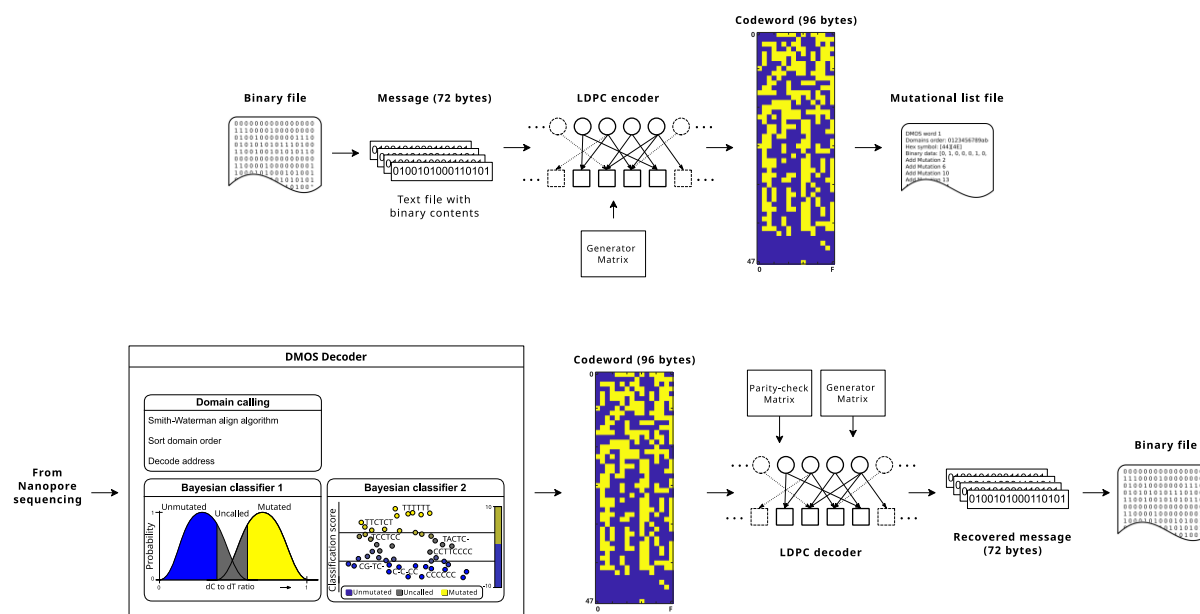


Figure. S9. DMOS coding and encoding scheme. The DMOS encoder uses binary files and converts it to the byte arrays with codewords. The DMOS encoder then depends on including error correction, transduces the coding protocol to DMOS writer, and change the state of domains from 0 to 1 on DNA tape.

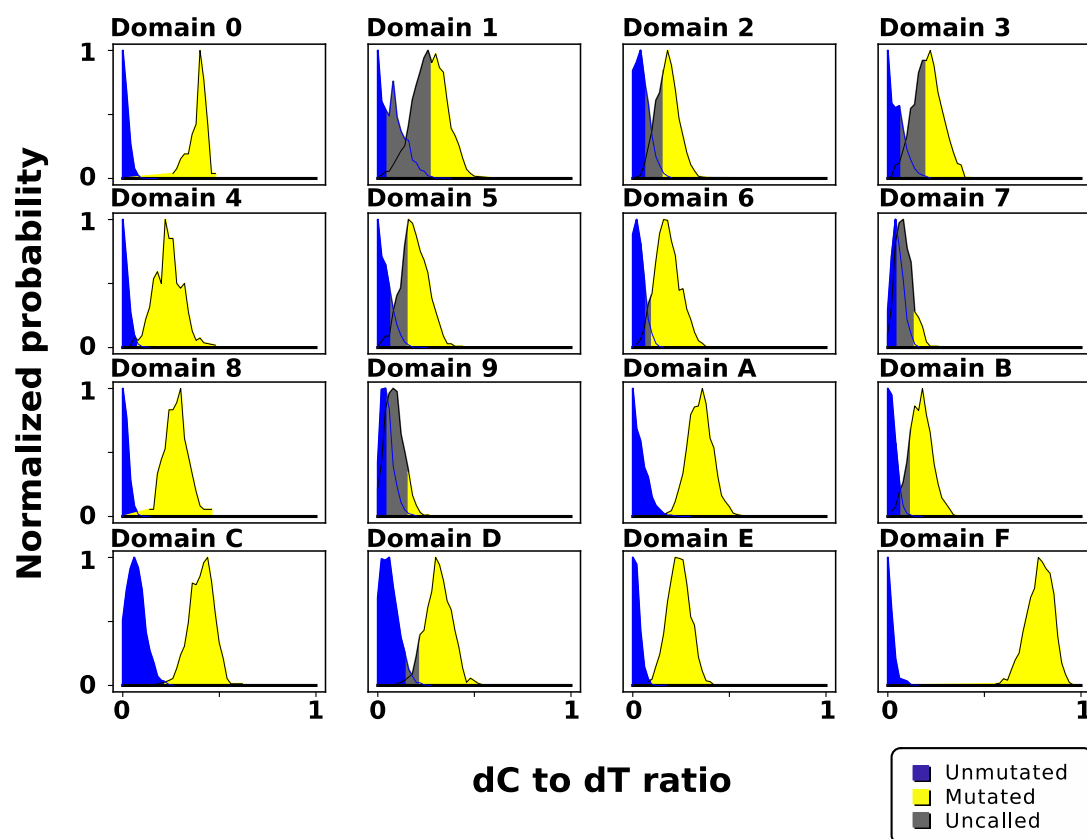


Figure. S10. Normalized probability curves of dC-to-dT ratio for each bit of DMOS registers at the experiment of encoding the title of this study.

Supplementary Information

Table S1: DNA-based memory technologies

DNA writing technologies	Density	Scalability	Toxic waste
An example of Phosphoramidite-based synthesis ²³	1.57 bits/nt	Yes	Yes
TdT enzymatic synthesis ^{10,12}	1.57 bits/nt	Yes	No
<i>In vivo</i> DNA writing in living cell populations ³⁵	3 bits/cell genome (genome size: 4,530,564 bp)	No	No
DNA nanostructures ³⁸	0.01 bits/nt	<u>Yes</u>	Yes
DNA backbone nicks ¹³	0.036 bits/nt	<u>Yes</u>	No
DMOS	0.014 bits/nt	Yes	No

***Table S2: The dC to dT threshold values for the first Bayesian classification step**

Block	Unmutated (t_u)	Mutated (t_m)
0	0.077050	0.144550
1	0.129758	0.178004
2	0.071650	0.151862
3	0.113244	0.168885
4	0.082900	0.166900
5	0.044773	0.139677
6	0.070100	0.157850
7	0.042900	0.094650
8	0.118000	0.178000
9	0.024487	0.102552
A	0.092800	0.160550
B	0.080100	0.147600
C	0.080100	0.147600
D	0.068250	0.135750
E	0.092200	0.177700
F	0.192950	0.288200

*Thresholds determine the state of the domain as Unmutated (dC to $dT < t_u$) and Mutated (dC to $dT > t_m$). The dC to dT values in between ($t_m < dC$ to $dT < t_u$) are considered uncertain.

Table S3: The sequences of barcode primers

dBC1F	TTTCTGTTGGTGCTGATATTGCGTTGTCGGTGTCTTTGTGATCACGAGGCCCTTTTCG
dBC2F	TTTCTGTTGGTGCTGATATTGCCCCTTTGTAGTCGTCTGTATCACGAGGCCCTTTTCG
dBC3F	TTTCTGTTGGTGCTGATATTGCTGTGTCCCAGTTACCAGGATCACGAGGCCCTTTTCG
dBC4F	TTTCTGTTGGTGCTGATATTGCTTCTATCGTGTTTCCCTAATCACGAGGCCCTTTTCG
dBC5F	TTTCTGTTGGTGCTGATATTGCCAGGGTTTGTGTAACCTTATCACGAGGCCCTTTTCG
dBC6F	TTTCTGTTGGTGCTGATATTGCGAACAACCAAGTTACGTATCACGAGGCCCTTTTCG
dBC1R	TTGCCTGTCGCTCTATCTTCCCGTGGGAATGAATCCTTTGATAAACTACCGCATTAAAGC
dBC2R	TTGCCTGTCGCTCTATCTTCCAAAGGCAGAAAGTAGTCTGATAAACTACCGCATTAAAGC
dBC3R	TTGCCTGTCGCTCTATCTTCGCACAGCGAGTCTTGGTTTGATAAACTACCGCATTAAAGC
dBC4R	TTGCCTGTCGCTCTATCTTCTGAAACCTTTGTCCTCTCTGATAAACTACCGCATTAAAGC
dBC5R	TTGCCTGTCGCTCTATCTTCTCTATCGGAGGGAATGGATGATAAACTACCGCATTAAAGC
dBC6R	TTGCCTGTCGCTCTATCTTCGAAAGAAGCAGAATCGGATGATAAACTACCGCATTAAAGC

Table S4: The sequences of DMOS bits and indexes

Initiator	atcacgaggcccttcgtcttcaagaattc
Index	TTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCT
State 0	ACTCactagtcctcgaaaacctcgtagg
Index 0	TGTCCTACTATGTCTTCTCTCTTCTACTACTTACCT
State 1	ACTCctccaatcaaatcagtcactagg
Index 1	GGATGGATGATCCCACACCTCACACGCAGGAGAGAA
State 2	ACTCcttggtcagggtcggacactgg
Index 2	CTAGTGGTAGATGTTGTGTGTGGCGCGAGAGAAAAGC
State 3	ACTCtattcacagcaactgcagcagg
Index 3	TTGCGACGATGACTGACGACTGCACGAAAAGCTGGA

State 4	ACTCatggtcaactcaatccaaaatgg
Index 4	GTGAGGAGGAGAAGTAAAAGAAAGCTTCGAGAGAGT
State 5	ACTCgttctcatcgctaccacgaagg
Index 5	AGTTTACACGGCGCTCTTTCCGGTTTGATCTTGCAC
State 6	ACTCatcaatagtgtcatggcatgtgg
Index 6	ATGTTTACGCACGCGTTTTCCACCCACGATGTTGT
State 7	ACTCtcgggagaaaggtcgctgtgagg
Index 7	CTGTTTGCACACACACCCGCACACCCTGTTCCCTCG
State 8	ACTCatcacgagttcacgataccgtgg
Index 8	ATGCGTTGCGTTGTTTTGCGTTCCACACCACACGTT
State 9	ACTCttgtggtcaatgtcactccgagg
Index 9	ATCCAAAGAGAACTGGGATTTCTAAAAGAGAGAGAA
State A	ACTCaagctcagcctcgtaaactgtgg
Index A	GTTTTACCTTTTTGCGCTTTTGTCTTCGTTTCGCCCT
State B	ACTCgaacagatcatcaaccattagg
Index B	GGCTCCCTACCACACACCACGTTTTGATGATAGTTG
State C	ACTCattcaatcaagctgcaaaggtgg
Index C	TACGAGAGGAAGCTTCACACACCACCACGATCGGAT
State D	ACTCgattcgaatatctctcttcgagg
Index D	CTTGCGCACACCTCACACACGTGTTTGTGTTGTGTT
State E	ACTCGcctcatcagcagaacaagttgg
Index E	CGATCCGCACACGCACGTCACACCTATCTTACGTGT
State F	ACTCtcattccagtcfaatgtggaaagg
Index F	GAAGAAAAGAAAGAGAAGAGAAAACCTCAAAAAGATGA
Terminator	ACTCatcgataagctttaatgcggtagtttatca

NATURE COMMUNICATIONS | <https://www.nature.com/naturecommunications>

Supplementary Information

	TTTCCGGTTTGATCTTCGCACACTCgctctcatcagcagaacaattggCGATCCGCACACGCAGCTCACACCTATCTTACGTGTACTCatcaatagtgctatggcatgggATGTTTACGCACGCGTTTCCCA ACCCACGATGTTGTACTCatcgataagctttaatgcgtagttatca
4	Register 4: [741C36FEA59D0B28] AtcacgaggcccttcgcttcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCtgggagaaggtcgtgtaggCTGTTTGACACACACCCGCACACCTGTTCCCTCGACTCat ggctcaactcaatcaaatggGTGAGGAGGAGAAGTAAAAAGAAAGCTTCGAGAGAGTACTCtccaatcaatcagtcactaggGGATGGATGATCCACACCTCACACCGAGGAGAAACTCattcaatc aagctcaaaagggTACGAGAGGAAGCTTCACACACCACGATCGGATACTCtcatcagcagcaactgcagcaggTTGCGACGATGACTGACGATCGACGAAAGAAAGCTTGGAAACTCatcaatagtgctat ggcatgggATGTTTACGCACGCGTTTTTCCCAACCACGATGTTGACTCtcatcagtcagtcagggaaggAAGAAAAGAAAGAGAAGAAACTCAAAGATGAACTCgctctcatcagcagaacaagt tggCGATCCCGCACACGCGACGTACACCTATCTTACGTGACTCaagtcagctctgttaaacgtrggGTTTACCTTTTGGCGTTTGTCTTCGTTGCTCGCTACTCgttctcatcgcgtaccacgaaggATT ACACGGCGCTTCTTCGGTTTGATCTTGCACACTCtggggtcaatgcactccgaggATCCAAAGAGAAGACTGGGATTTCTAAAAGAGAGAGAAACTGatcgaatctctctcctggtTCGCGCACAC CTCACACACGTTGTTGTGTGTACTCactagctcgaaaactcgggTGTCTACTATGTTCTTCTTCTACTACTTACCTACTCgaacgatcatcaaccattaggGGCTCCCTACACACACC ACGTTTGTATGATAGTTGACTCtctgctcagggtcgacacatggCTAGTGGTAGATGTTGTGTGTGGCGGAGAGAAGCACTCatcagagttcacgaltccggttATGGCTTGGCTTGTTCGCGTTC CACACCAACAGTTACTCatcgataagctttaatgcgtagttatca
5	Register 5: [4D725B98E3C0FA61] AtcacgaggcccttcgcttcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCatggccaactcaatcaaatggGTGAGGAGGAGAAGTAAAAAGAAAGCTTCGAGAGAGTACTC gattcgaatctctctcaggCTTGGCGACACCTCACACACGTTGTTGTGTGTTACTCtgggagaaggtcgtgtaggCTGTTTGACACACACCCCGCACACCTGTTCCCTCGACTCtctggcagg gtcctggaactggCTAGTGGTAGATGTTGTGTGTGGCGGAGAGAAGCACTGttctcatcgcgtaccacgaaggAGTTTACACGGCGCTCTTCCGGTTTGATCTTGCACACTCgaacgatcatcaacca ttaaggGCTCCCTACACACACACAGCTTTTGATGATGTTGACTCtgggtcaatgcactccaggATCCAAAGAGAAGCTGGGATTTCTAAAAGAGAGAGAAACTCatcagagttcacgaltccggttAT CGGTTGGCTTGTTCGCTTCCACACACACGTTACTCgctctcatcagcagaacaattggCGATCCGCACACGACGTCAACACTTCTTACGTGACTCtctcaatcaatcagtcactaggGGA TGATGATGTTGTGTGGCGGAGAGAAGCACTCtgggagaaggtcgtgtaggCTGTTTGACACACACCCGCACACCTGTTCCCTCGACTCactagttcgaaaactcgggTGTCTACTATGT TCTTCTCTCTCTACTACTTACCTACTCtcatcagtcagtcagggaaggAAGAAAAGAAAGAGAAGAAACTCAAAGATGAACTCaagtcagctcgtttaaacgtrggGTTTACCTTTTGGCG TTTTGTTGTTGCTTGCCTACTCatcaatagttcatggcatgggATGTTTACGCACGCGTTTTCCACCCACGATGTTGTACTCtccaatcaatcagtcactaggGGATGGATGATGCCAACCTCAC ACGCAGGAGAGAAACTCatcgataagctttaatgcgtagttatca
6	Register 6: [C63F5AE182704B9D] AtcacgaggcccttcgcttcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCatccaatcaagtcgaaagggTACGAGAGGAAGCTTCACACACCACCATCGGATAGTCA tcaatggtgctatcgatggATGTTTACGCACGCGTTTTCCCAACCACAGTGTGTACTCtcatcagcagcagtcagcaggTTGCGAGGTAGTACGACGTACGACGAAAGCTCGAACTCtcatcctcagtc taagtggaagGAGAAAAGAGAAGAGAAGAAACTCAAAGATGAAGTCTGttctcatcgcgtacacgaaggAGTTTACACGGCGCTCTTCCGGTTTGATCTTGCACACTCaagtcagctcgttga acggtgGTTTACCTTTTGGCGTTTGTCTTCGTTCTGCTCCCTACTCgctctcatcagcagaacaattggCGATCCGCACACGACGTCAACACTTCTTACGTGACTCtccaatcaatcagtcactaggGGA TGATGATGTTGTGTGGCGGAGAGAAGCACTCtgggagaaggtcgtgtaggCTGTTTGACACACACCCGCACACCTGTTCCCTCGACTCactagttcgaaaactcgggTGTCTACTATGT TCTTCTCTCTCTACTACTTACCTACTCtcatcagtcactcaatcaaatggGTGAGGAGGAGAAGTAAAAAGAAAGCTTCGAGAGAGTACTCgaacgatcatcaaccattaggGGCTCCCTACACACACC ACGTTTGTATGATAGTTGACTCtgggtcaatgcactccgaggATCCAAAGAGAAGCTGGGATTTCTAAAAGAGAGAGAAACTCgattcgaatctctctcaggCTTGGCGACACCTCACACAGTGT TTGTGTTGTGTTACTCatcgataagctttaatgcgtagttatca
7	Register 7: [8A2057B6FEC9D341] AtcacgaggcccttcgcttcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCatcagagttcacgataccgggATGUGTTGCGTTGTTTGGGTTCCACACACAGTTACTCaa ctcagctcgtttaaacgggGTTTACCTTTTGGCGTTTGTCTTCGTTCTGCTCCCTACTCtctggtagggctggacactggCTAGTGGTAGATGTTGTGTGTGTGGCGGAGAGAAGCACTCactagttcgtga aacctcgggtTGCTCTACTAGTCTTCTCTCTCTACTACTTACCTACTCtctcatcgcgtacacgaaggAGTTTACACGGCGCTCTTCCGGTTTGATCTTGCACACTCtgggagaaggtcgtgtagg CTGTTTGACACACACCCGCACACCTGTTTCCCTCGACTCgaacgatcatcaaccattaggGGCTCCCTACACACACACAGTGTGATGATGATGACTCatcaatagttcatggcatgggATGTTT ACGCACCGTTTTCCCAACCACAGTGTGTACTCtcatcaatcaagtcgaaagggTACGAGAGAAGCTTCACACACCACCGATCGGATACTCtctgggtcaatgcactccgaggCATCTCCGCA CACGCACTCTCACACTCTTACCTACTCtcatcaatcaagtcgaaagggTACGAGAGAAGCTTCACACACCACCGATCGGATACTCtctgggtcaatgcactccgaggATGTTTCAAGAGAAGT GGGATTTCTAAAAGAGAGAGAAACTCgattcgaatctctctcgggCTTGGCGCACACCTCACACAGTGTGTTGTGTGTTACTCtcatcagcagcagtcgagggTTGCGACGATGACTGACGA TGACGAAAGACTTGAAGTCTCattgccaactcaatcaaatggGTGAGGAGGAGAAGTAAAAAGAAAGCTTCGAGAGAGTACTCtccaatcaatcagtcactaggGGATGGATGATCCCAACCTCAC ACGCAGGAGAGAAACTCatcgataagctttaatgcgtagttatca
8	Register 8: [0DEA9213864B75FC] AtcacgaggcccttcgcttcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCactagtcctcgaaaactcgggTGTCTACTATGTTCTTCTCTCTACTACTTACCTACTCgattc gaatatactctcgggCTTGGCGACACCTCACACAGTGTGTTGTGTGTTACTCgctctcatcagcagaacaattggCGATCCGCACACGACGTCAACACTATCTTACGTGTACTCaagtcagctcgtta aacgggtGTTTACCTTTTGGCGTTTGTCTTCGTTCTGCTCCCTACTCtgggtcaatgcactccgaggATCCAAAGAGAAGCTGGGATTTCTAAAAGAGAGAGAAACTCtctggtcagggtcggaactgg TAGTGGTAGATGTTGTGTGGCGGAGAGAAGCACTCtccaatcaatcagtcactaggGGATGGATGATGCCACACCTCACACGACGAGAGAAACTCtcatcagcagcagtcagcaggTTGGC ACGATGACTGACGACGTGACGAGAAAGCTGGAAGTCTCatcagagttcacgataccgggATGCGTTTGGCGTTGTTTGGGTTCCACACACAGTACTCatcaatagttcatggcatgggATGTTTACGC ACGCGTTTTTCCCAACCACAGTGTGTTACTCattgccaactcaatcaaatggGTGAGGAGGAGAAGTAAAAAGAAAGCTTCGAGAGAGTACTCgaacgatcatcaaccattaggGGCTCCCTACACCA CACACAGCTTTTGTATGATAGTTGACTCtgggagaaggtcgtgtaggCTGTTTGACACACACCCGCACACCTGTTCCCTCGACTCgttctcatcgcgtacacgaaggAGTTTACACGGCGCTCTTT CGGTTTGTATCTTGACACTCTcattccagtcagggaaggAAGAAAAGAGAAGAAACTCAAAGATGAACTCatcaatcaagtcgaaagggTACGAGGAGGAAGCTTCAACGAGCACTGGGATTTCT CACACAGTCCGATACTCatcgataagctttaatgcgtagttatca
9	Register 9: [D2574CA8913FE0B6] AtcacgaggcccttcgcttcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCgattcgaatctctctcaggCTTGGCGACACCTCACACAGTGTGTTGTGTGTTACTCtctgg tcagggtcggaactggCTAGTGGTAGATGTTGTGTGTGGCGGAGAGAAGCACTCgttctcatcgcgtacacgaaggAGTTTACACGGCGCTCTTCCGGTTTGATCTTGCACACTCtgggagaaggt tcgtgtaggCTGTTTACGACACACCCGCACACCTCTGCTCGACTCatggtcagtcagtcactccaggGTGAGGAGGAGAAGTAAAAAGAAAGCTTCGAGAGAGTACTCatcaatagttcatgca tgggATGTTTACGCACGCGTTTTTCCCAACCACGATGTTGACTCgctctcatcagcagaacaattggCGATCCGCACACGACGTCAACACTTCTTACGTGACTCtccaatcaatcagtcactaggGGA TGATGATGTTGTGTTGCGTTCCACACACAGTGTGTTACTCtgggtcaatgcactccgaggATCCAAAGAGAAGCTGGGATTTCTAAAAGAGAGAGAAACTCtcatcagcagcagtcagcaggTTGGC ATGATCCACACCTCACACGAGGAGAGAAACTCtcatcagcagcagtcagcaggTTGGCAGCATGACTACGCACTGACGAGAAAGCTTGAAGTCTCtcatcagtcagtcagggaaggAAGAAAAGAA AGAGAAAGAGAAACTCAAAGATGAACTCgctctcatcagcagaacaattggCGATCCGCACGACGTCAACACTATCTTACGTGTTACTCactagtcctgaaactcgggTGTCTACTATGTTCTT CTCTCTCTACTACTTACTACTCgaacgatcatcaaccattaggGGCTCCCTACACACACAGCTTTTGATGATAGTTGACTCatcaatagttcatggcatgggATGTTTACGCACGCGTTTTTCCCA CCACAGTGTGTTACTCatcgataagctttaatgcgtagttatca
10	Register 10: [BC85F6E3072A4D19] AtcacgaggcccttcgcttcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCgaacgatcatcaaccattaggGGCTCCCTACACACACAGCTTTTGTATGATGTTACTCatt caatcaagtcgaaagggTACGAGAGGAAGCTTCACACACCACCATCGGATACTCtcatcagtcagtcagggaaggAAGAAAAGAGAAGAGAGAAACTCAAAGATGAACTCgttctcatcgcgtac gcacgaaggAGTTTACACGGCGCTCTTCCGGTTTGATCTTGCACACTCtcatcagtcagtcagggaaggAAGAAAAGAGAAGAGAGAAACTCAAAGATGAACTCatcaatagttcatgca tgggATGTTTACGCACGCGTTTTTCCCAACCACGATGTTGACTCgctctcatcagcagaacaattggCGATCCGCACACGACGTCAACACTTCTTACGTGACTCtccaatcaatcagtcactaggGGA TGATGATGTTGTGTTGCGTTCCACACACAGTGTGTTACTCtgggtcaatgcactccgaggATCCAAAGAGAAGCTGGGATTTCTAAAAGAGAGAGAAACTCtcatcagcagcagtcagcaggTTGGC AGCATGACTGACGACGTGACGAGAAAGCTGGAAGTCTCatcagagttcacgataccgggATGCGTTTGGCGTTGTTTGGGTTCCACACACAGTACTCatcaatagttcatggcatgggATGTTTACGC ACGCGTTTTTCCCAACCACAGTGTGTTACTCattgccaactcaatcaaatggGTGAGGAGGAGAAGTAAAAAGAAAGCTTCGAGAGAGTACTCgattcgaatctctctcaggCTTGGCGACACCTCACAC ACACAGCTTTTGTATGATAGTTGACTCtgggagaaggtcgtgtaggCTGTTTGACACACACCCGCACACCTGTTCCCTCGACTCgttctcatcgcgtacacgaaggAGTTTACACGGCGCTCTTT CGGTTTGTATCTTGACACTCTcattccagtcagggaaggAAGAAAAGAGAAGAAACTCAAAGATGAACTCatcaatcaagtcgaaagggTACGAGGAGGAAGCTTCAACGAGCACTGGGATTTCT CACACAGTCCGATACTCatcgataagctttaatgcgtagttatca
11	Register 11: [35F029BEA8C167D4] AtcacgaggcccttcgcttcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCtcatcagcagcagtcagcaggTTGCGACGATGACTGAGACTGACGAGAAAGCTCGGAAGTCT gttctcatcgcgtaccagaaggAGTTTACACGGCGCTCTTCCGGTTTGATCTTGCACACTCtcatcagtcagtcagggaaggAAGAAAAGAGAAGAGAGAAACTCAAAGATGAACTCgttctcatcgcgtac gcacgaaggAGTTTACACGGCGCTCTTCCGGTTTGATCTTGCACACTCtcatcagtcagtcagggaaggAAGAAAAGAGAAGAGAGAAACTCAAAGATGAACTCatcaatagttcatgca tgggATGTTTACGCACGCGTTTTTCCCAACCACGATGTTGACTCgctctcatcagcagaacaattggCGATCCGCACACGACGTCAACACTTCTTACGTGACTCtccaatcaatcagtcactaggGGA TGATGATGTTGTGTTGCGTTCCACACACAGTGTGTTACTCtgggtcaatgcactccgaggATCCAAAGAGAAGCTGGGATTTCTAAAAGAGAGAGAAACTCtcatcagcagcagtcagcaggTTGGC AGCATGACTGACGACGTGACGAGAAAGCTGGAAGTCTCactagtcctgaaactcgggTGTGCTACTATGTTCTTCTTCTTACTACTTACTCtgggagaaggtcgtgtaggCTGTTTG CACACACACCCGCACACCTGTTCCCTCGACTCtctgtagggtcgacactggCTAGTGGTAGATGTTGTGTGTGGCGGAGAGAAGCACTCtccaatcaatcagtcactaggGGCTCCCTACACAG CGGCTTTTGTCTTCGTTCTGCTCCCTACTCattgccaactcaatcaaatggGTGAGGAGGAGAAGTAAAAAGAAAGCTTCGAGAGAGTACTCgattcgaatctctctcaggCTTGGCGACACCTCACAC ACGTTGTTGTGTTGTTGTTACTCtccaatcaatcagtcactaggGGATGGATGATGCCACACCTCATggttcaatcagtcagcaggATCCAAAGAGAAGCTGGGATTTCT AAAAGAGAGAGAAACTCatcgataagctttaatgcgtagttatca
12	Register 12: [4BF039815E27C6DA] AtcacgaggcccttcgcttcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCatggtccaactcaatcaaatggGTGAGGAGGAGAAGTAAAAAGAAAGCTTCGAGAGAGTACTC gaacgatcatcaaccattaggGGCTCCCTACACACACACAGTTTTGATGATAGTTGACTCtcatcagtcagtcagggaaggAAGAAAAGAGAAGAGAGAAACTCAAAGATGAACTCactagtc ctcgtggaggctggTGTCTACTATGTTCTTCTTCTTCTACTACTTACCTACTCtcatcagcagcagtcagcaggTTGCGACGATGACTGACGATGACGATCGGAACTCtgggtcaatgcact cgaggATCCAAAGAGAAGCTGGGATTTCTAAAAGAGAGAGAAACTCatcagagttcacgataccgggATGCGTTTGGCGTTTGTGTCTTCGTTGCTCCCTACTCatcagagttcacgataccgggATGCGTT GATGGATGATGCCACACTCACACGAGGAGAGAAACTCgttctcatcgcgtacacgaaggAGTTTACCTTTTGGCGTTTGTGTCTTCGTTGCTCCCTACTCatcagagttcacgataccgggATGCGTT GCATCCGCACACGACGTGACACCTATCTTACGTGACTCaagtcagctcgttgaatcgggGTTTACCTTTTGGCGTTTGTGTCTTCGTTGCTCCCTACTCatcagagttcacgataccgggATGCGTT GCGTTTGTGGGTTTCCACACACAGTGTACTCatcaatcaagtcgaaagggTACGAGAGGAAGCTTCACACACCACCGATCGGATACTCtccaatcaatcagtcactaggGGATGATGATGATC CACACCTCACACGAGAGAGAACTCatcaatagttcatggcatgggATGTTTACGACGCGTTTTTCCCAACCACGATGTTGACTCtctgggagaaggtcgtgtaggCTGTTTGACACACACCC CGCACACCTGTTCCCTCGACTCgattcgaatctctctcaggCTTGGCGACACCTCACACAGCTGTTGTGTGTGTTACTCattgccaactcaatcaaatggGTGAGGAGGAGAAGTAAAAAGAA GCTTCCGAGAGAGTACTCatcgataagctttaatgcgtagttatca
13	Register 13: [12B0A46EF5387DC9] AtcacgaggcccttcgcttcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCtccaatcaatcagtcactaggGGATGGATGATGCCACACTCTCACCGAGGAGAAACTCgt ctggtcagggtcggaactggCTAGTGGTAGATGTTGTGTGTGGCGGAGAGAAGCACTCgaacgatcatcaaccattaggGGCTCCCTACACACACACGTTTGTATGATGATGACTcatgctct cgaaactcgtggTGTCTACTATGTTCTTCTTCTTCTACTACTTACCTACTCaagtcagctcgttgaatcgggGTTTACCTTTTGGCGTTTGTGTCTTCGTTGCTCCCTACTCattggtccaatcaatcaaat gGTGAGGAGGAGAAGTAAAAAGAAAGCTTCGAGAGAGTACTCatcaatagttcatggcatgggATGTTTACGCAACGCGTTTTTCCCAACCACAGTGTGATGACTCgctctcatcagcagaacaattggCGAT CCGCACACGACGCTACACACTATCTTACGTGACTCtcatcagtcagtcagggaaggAAGAAAAGAGAAGAGAGAAACTCAAAGATGAACTCgttctcatcgcgtaccagaaggAGTTTACCA CGCGCTCTTCTTCGGTTTGTATCTTGACACTCtcatcagcagcagtcagcaggTTGCGACGATGACTGACGATGACGAGAAAGCTGGAAGTCTCatcagagttcacgataccgggATGCGTTGCGTTG TTTTGCGTTCCACACACAGTGTACTCtgggagaaggtcgtgtaggCTGTTTGACACACACCCGCACACCTGTTCCCTCGACTCgattcgaatctctctcaggCTTGGCGACACCTCACACA CGTGTGTTGTGTTGTTACTCatcaatcaagtcgaaagggTACGAGAGGAAGCTTCACACACCACAGCATCGGATACTCtgggtcaatgcactccgaggATCCAAAGAGAAGCTGGGATTTCTTA AAAGAGAGAGAAACTCatcgataagctttaatgcgtagttatca

[illegible]

	AtcacgaggcccttgccttcaagaattctttatagaaaacgcttttgaagaagaagatgatctctactCtctggccagggtcggagacgttgTAGTGGTAGATGTTGTGTGTGGCGGAGAGAAAGCACTg ccttagccagcaagcaggtagGCATCCGCACAGCCAGCTCAACCACTATCTCACTGCTACTCattcaatcaagctgcgaagggTACGAGAGGAAGCTTCACACACCACCGATCGGTGACTCgattgcaac tcaatccaataggTGGAGGAGGAGAAATAAAGAAAGCTTCGGAGAGATCACTcatcagagttcacgatacgttggAGTGGCGTTCGCTGTGTTTCGGCTTCACACCACCACTTACTCtgggagaagggcgc tggATGTTTGTCGACACACCCGACACCCCTGCTCCGAGTGTGTACTCagttcgaatctctctcgggaaggGAGAAAGAAAGAGAAAGAAAGCTCAAAGATGAACCTCgaacagatcaatcacccatg TGCCCTCCCTCACACACACCTGTTTGATGATAGTTGATCTCtctcaatcaatcagttacaggGGATGGATGATCCCACTCAGACCGCAGGAGAGAAACTCaaagtcagcctgttgaacgtggGTTT TACCTTTTGGCGTTTGTCTTCGTTGCTTCCCTCACTCgattggaatctctctcgggaaggTGTGGCACACCTCACACAGCGTTGTGTGTGTGTTACTCtgggtcaatgtcactccagagATCCACAAAGAA CTCGGGAATTTCTAAAGAGAGAGAAACTCataatagttgcagtcagttggATGTTTACGCACGGTTTTCGCCACCCAGCATGTTGTGTGTTACTCtattcacagcaactgcagagGTTGGCAGAGTACTGAC GACTGCAGAAAGCTGGAACCTCagttcctctcgaaaactcgggTGTCTACTATGTTCTCTCTACTACTACTCTCtctcatcgctgcagcaagaaggATTTTACCGCGCGCTTTCGGG TTTGATCTTGCACTCatcgataaagctttaatgcgtagttatca
25	Register 25: [20CDF64A358791BE] AtcacgaggcccttgccttcaagaattctttatagaaaacgcttttgaagaagaagatgatctctactCtctggccagggtcggagacgttgTAGTGGTAGATGTTGTGTGTGGCGGAGAGAAAGCACTCa ctagctctcgaaaactcgggTGTCTCTATGCTTCTCTCTCTCTCTACTACTACTCattcaatcaagctgcgaagggTACGAGAGGAAGCTTCACACACCACCGATCGGTGACTCgattgcaac tctctcggcTGTTCGGGACACCACTGACACACGTTGTTGTGTGTGTTACTCtcttcacgtcatggtgaaggGAGAAAGAAAGAGAAAGAAAGCTTCAGAGAGATGAACCTCataatagttgcagtcagtt ggATGTTTACGCACCGCTTTCGCCACCCAGTGTGTACTCagttcgaatctctcgggaaggTGGAGGAGGAAGTAAAGAAAGCTTCAGAGAGATGAACCTCaaagtcagcctgttgaacgtggGTTT TACCTTTTGGCGTTTGTCTTCGCTGCTCCCTCACTCtcttcacagcaactgcagcggTTGGCGAGTACTGACGATCGACGAGAAAGCTGGAACCTCgntctcagctgcagcaagaaggAGTTTACAC GGCGCTCTTTCGGGTTTGTCTTGCACTACTCagttcagttcagatacgttggATGTTGCGGTGTGTTTGGCTTTCACACACCACTGACTCtgggagaagggctcgtggagGTTTGTGACACACA CCGCGACACCCCTGTTCCCTCGACTCtgggttcaatgcactcggagATCCAAAGAGAACTGGGATTTCTAAAGAGAGAGAGAAACTCtctcaatcaatcagttacgtaggGTTGGATGATCCACCACT CACACCGAGGAGAGAAACTCgaacagatcaatcaatcaggTGTCTCTACCCACACCAACCGTTTGTGATGATGTTGACTCgctctcagcagcaagaaggCGATCCGCACACCGCAGCTACACCG TATCTTACGTGTAACCTCagtgataagctttaatgcgtagttatca
26	Register 26: [79814FAD6C305E2B] AtcacgaggcccttgccttcaagaattctttatagaaaacgcttttgaagaagaagatgatctctactCtctgggagaagaaggtcgtgtgaggCTGTTTGCACACACACCCGCCAACCCTGTTCCCTGACTCtt gggttcagttcagtcaggATTCCAAAGAACTGGGATGTTCTAAAGAGAGAGAAACTCatacagagttcacgatacctggTAGTGGTGTGGCTTGTGCGTTCACACCACTGTTACTCtctcaatcaa atcagtcaggTAGGATGATGCCACCTCACACGAGGAGAGAAACTCagttgcaactcaatcaaggTTCAGGAGGAGAAGTAAAGAAAGCTTCGAGAGAGTACTCgattcagttcagtt ggaggGAGAAAGAAAGAGAGAGAAACTCAAAGATGAACCTCaaagtcagcctgttgaacgtggGTTTATCTTACCTTTTGGCGTTTGTCTTCGCTGCTCCCTACTCgattcgaatctctctcggc TGTGGCCACCTCACACACGTTGTTGTGTTGTTACTCataatagttgcagtcagttggATGTTTACGCACGGTTTTCGCCACCCAGTGTGTACTCgattcagttcgaacggggTACGAGA GGAAGGTTTCACACACCACTCGGATGATCTCtcttcacagcaactgcagcaggTTGCGACGATGACTGACGATCGACGAGAAAGCTGGAACCTCagttcagctcgaaaactcgggTGTCTACTAT GTCTCTCTCTTCTACTACTACTACTCtcttcagctcagtcagcaaggAGTTTACACGGCGGTTCTTCCCGGTTGTGTTGTGCTGACACTCgctcaatcagcagcaagaaggCGATCCGCACACCGCAGC TCACACCTATCTTACGTGTAACCTCtctggccaggctcggcagcggTAGTGGTAGATGTTGTGTGTGGCGGAGAGAAAGCAACTCgaacagatcaatcaatcaggGGCTCCTTACCACACCAACGCT TTTGTAGTAGTGTACTCatacgaatgctttaatgcgtagttatca
27	Register 27: [38419275ADECBF06] AtcacgaggcccttgccttcaagaattctttatagaaaacgcttttgaagaagaagatgatctctactCtcttcacagcaactgcagcaggTTGGCAGCAGTACTGACGACTCGACGAAAGCTGGAACCTCa tcaagttcagatacctgggATGCGTGGCTGTTTGGCTGTCCACACCACTGTAACCTCagttcgaatcaatcaatcaaggTTCAGGAGGAGAAGTAAAGAAAGCTTCGAGAGAGTACTCtctcaatcaa atcagtcacaggCGATGGATGATGCCACACTCACACCGCAGGAGAAACTCtgggttcaatgtcagtcaggATCCAAAGAAAGAACTGGGATTTCTAAAGAGAGAGAAACTctgtgcagggctgc gacagctCTAGTGGTAGATGTTGTGTGGCGGAGAGAAAGCACTCtctggaagaagctcagtcaggCTGTTTGCACACACCCGCCAACCCTGTTCCCTGACTCgttctcagctgcatacagcaagg AGTTTTCGCGCGCGCTTTCGCGTTTGATCTTCGCACTCAAGtcagctcgttgaacgtggGTTTATCTTTCGCGTTTGTGCTGTCTGCTCCCTACTCgattggaatctctcggcgttGGCTTCGGC AACCTTCACACAGCTGTTTGTGTGTTGTTACTCgctctcagcagcaagaaggGATCCGCCACACCGCAGCTGACACTATCTTACGTTGACTCattcaatcagctcgaagggTACGAGAGGAAG CTTCACACACCCACCATGCGGATGATCtCgaacagatcaatcaatcaggGGCTCCTTACCACACCACTGTTGATGATAGTTGACTCtcttcacagcaatggtgaaggGAGAAAGAAAGAGAA AGAAAGAACTCAAAGATGAACCTCagttcagctcgaaaactcgggTGTCTACTATGTTCTCTCTACTACTACTCataatagttcagttcagttggATTCGAAGACCGGCTTTCCTCA CCACAGTGTTGTACTCatcgataaagctttaatgcgtagttatca
28	Register 28: [63C7AF18542BD0E9] AtcacgaggcccttgccttcaagaattctttatagaaaacgcttttgaagaagaagatgatctctactCattcaatagttgcagtcagttggATGTTTACGCACGGCTTTCGCCACCCACAGATGTTTGTACTCtcat tcacagcaactgcagcaggTTGGCAGCATGACTCAGCAGCTGCACGAGAAAGCTGGAACCTCattcaatcaagctcgaaggTTCAGAGAGGAAGCTTCACACACCACCGATCGGTACTCtgggaga aggctcgtcaggCTGTTTGCACACACCCGACACCCCTGTCCTCGACTCaaagtcagcctgttgaacgtggGTTTATCTTTCGCGTTTGTCTTCGCTGCTCCCTACTCtcttcacgatacgttgggaa ggGAGAAAGAAAGAAAGAGAGAAACTCAAAGATGAACCTCtctcaatcaatcagttacaggTAGGATGATGCCACCTCACACCGAGAGAAAGCTCatacagttcagttcgaacgggTACGAGAGGA TGGCTTGGCTGTTTGTGGCTTCACACACCACTGTAACCTCgntctcagctcagcaagaaggAGTTTACACGGCGCTTTCGCGTTGTGATCTTGCACTACTCagttcgaatcaatcaatgaaggGTTGAGGA CTGGAAGTAAAGAAAGAAAGCTTCGAGAGAGTACTCtctggccaggctcggacagctgCTAGTGGTAGATGTTGTGTGTGGCGGAGAGAAAGCACTCgaacagatcaatcaatcaggGGCTCCTTAC CACACACACCACTGTTTGATGATAGTTGACTCgattcagatactctctcggcGTTCGCGACACTCACACAGCTGATGTTGTGTGTGTTGTTGTTACTCagttcagctcgaaaactcgggTGTCTACTATGTCTTCTC TCTCTTCTACTACTACTACTCgctctcagcagcaagaaggCGATCCGCACACCGCAGCTCACACCTATCTTACGTGTAACCTCtgggttcaatggtgaaggATTCCAAAGACCGGCTTTCCTCA AAAAGAGAGAGAAACTCatcgataaagctttaatgcgtagttatca
29	Register 29: [08CB5E7F2946AD31] AtcacgaggcccttgccttcaagaattctttatagaaaacgcttttgaagaagaagatgatctctactCactagttcctcgaaaactcgggTGTCTACTATGTTCTTCTCTTCTACTACTTACCTACTCatac gagttcacgatacgttggATGCGTGTGGCTTGTTGGCTGTCCACACCACTGTAACCTCattcaatcaagctcgaaggTTCAGAGAGGAAGCTTCACACACCACTCGGATGACTCgaacagatcatca accataggGGCTCCTCACACACCACTGATGTTTGATGATAGTTGACTCgntctcagctcagtcagcaggAGTTTACCGGGCTTTCGCGTTGTGTTGTGCTACTTGCACTCgctctcagctcagtcagttg cGGATCCGCACACCGCAGCTCACCTTACTTACGTGTAACCTCtgggagaagaaggtcgtgtaggCTGTTTGCACACACCCGCCAACCCTGTTCCCTCGACTCtcttcagttcagttcgggaaggGAAG AAAAGAAAGAGAGAGAGAAACTCAAAGATGAACCTCtctggccaggctcggacagctgCTGTTGGTAGATGTTGTGTGTGGCGGAGAGAAAGCACTCtgggttcaatggtcagtcaggATCCAAAG GAGAACTGGGATTTCTAAAGAGAGAGAAACTCagttgcactcaatcraaaggTGGAGGAGGAGAAGTAAAGAAAGCTTCGAGAGAGTACTCataatagttgcagtcagttggATGTTTACCGCA CCGGTTTTTCGCCACCCAGCTGTGTACTCaaagtcagctcgttgaacgggGTTTATCTTTCGCGTTTGTGCTGTGCTCCCTACTCgattggaatctctcggcGTTCGGCACACCTCACACA CGTGTGTTGTGTGTTACTCtcttcacagcaactgcagcaggTTGCGACGATGACTGACGACTCGAGAAAGCTGGAACCTCtctcaatcaatcagttacaggGGATGGATGATCCACACCTCACAC CGCAGGAGAGAAACTCatcgataaagctttaatgcgtagttatca
30	Register 30: [82D73FC10AE4956B] AtcacgaggcccttgccttcaagaattctttatagaaaacgcttttgaagaagaagatgatctctactCatacagagttacgatacgttggATGCGTGTGGCTGTTGTTTTCGCTTCACACACCACTGACTCtct gttcggggctcggacagctgCTGTGGTAGATGTTGTGTGTGGCAGAGAAAGCACTCgattggaatctctctcgggTGTGGCACACCTCACACCGTGTGTGTGTTGTACTCtctcgggagaag gtcgtggaggCTGTTTGCACACACCCGACACCTGCTTCCCTGCACTCtcttcacagcaactgcagcaggTTCGCACAGTACTCAGCAGCTCGACGAGAGAAAGCTTGAACCTCattcagttcagttgga aggGAGAAAGAAAGAGAGAGAGAAACTCAAAGATGAACCTCattcaatcaatcagtcgaagggTACGAGAGGAAGCTTCACACACCACCGATCGGTACTCtctcaatcaatcagtcagc GGATGGATGATGCCACACTCACACCGCAGGAGAAACTCagttcctcgaaaactcgggTGTCTACTATGTTCTCTCTCTACTACTACTCtcttcagctcagttcagttcagttggTTTTA CTTTTTC

	actgcgacgagTTGGCAGGATGACTGACGACTGACGAGAAAGCTTGAAGACTCattgccaactcaatcaaaaggTTGAGGAGGAGAAAGTAAAGAAAGCTTGACAGAGAGTACTGgttcatcgcgttacca cgaaggAGTTTACACGGCGGCTCTTCCGGTTGATCTTGCCACTCATcaataatggtcatggcattggAGTTTACCGCAGCGGTTTCCACCACGAGTGTGTACTTCruggaagaagcgtcgtgaggCT GTTTGCCACACACCCGACACCCCTGTCCTCTCGACTCatacgaagtacgataccgagATGCGGTTGCGTTGTGTTGGGTTCCACACACAGCTGATTCTCnrtgggtcaatgcactccaggATCTCAAAAG AGAACTGGGATTTCTAAAGAGAGAGAAGAACTCaagtcagctcgttgaacggggGTTTACCTTTTGGCGTTTGTCTTGTCTGCTCCTACTCgaacgatatacaaccattaggGGCTCCCTACACCA CACCACGTTTGTATGATAGTTGACTCattcaatacagctgcaaaagggtTACGAGAGGAAGCTTCACACACCCACACGATCGGATGTAATCTCgctctcatcagagaacaaagggtGCATTCGCCACACCGACGT CACCACTATCTTACGTGTACTCattccagtcattgggaaggGAGAAAGAAAGAGAGAAGAACTCAAAGATGGAATCTCgattgcaatctctctcaggCTTGGCCACACCTCACACACGTG CTCAAAAGATGAACTCatcgaataagcttaatgcggagtattca
35	Register 35: [0123456789ABCEFD] AtcacgaggcccttgccttcaagaattTTTATAGAAAACGTTTGAAGAAGAAGATGATCTCTACTCactagtcctcgaaaacctcgggTGTCTCTACTATGTCTTCTCTCTTCTACTACTTACCTACTCTctca atacaatcagtcactaggGGATGGATGATCTCCACACCTCACACCGAGGAGAAACTCTgttgcaggctcggaacctggCTAGTGGTAGTGTGTGTGTGGCGCGGAGAGAAGCACTCattcacagca actcagcaggTTGGCAGCATGACTCAGCACTGCAGCAAAAGCTGGAACCTCattgccaactcaatcaaaagggtTGAGGAGGAGAGAATAAAGAAAGCTTCGAGAGAGTACTGgttcatcgcgttacca cgaaggAGTTTACACGGCGGCTCTTCCGGTTGATCTTGCCACTCATcaataatggtcatggcattggAGTTTACCGCAGCGGTTTCCACCACGAGTGTGTACTTCruggaagaagcgtcgtgaggCT GTTTGCCACACACCCGACACCCCTGTCCTCTCGACTCatacgaagtacgataccgagATGCGGTTGCGTTGTGTTGGGTTCCACACACAGCTGATTCTCnrtgggtcaatgcactccaggATCTCAAAAG AGAACTGGGATTTCTAAAGAGAGAGAAGAACTCaagtcagctcgttgaacggggGTTTACCTTTTGGCGTTTGTCTTGTCTGCTCCTACTCgaacgatatacaaccattaggGGCTCCCTACACCA CACCACGTTTGTATGATAGTTGACTCattcaatacagctgcaaaagggtTACGAGAGGAAGCTTCACACACCCACACGATCGGATGTAATCTCgctctcatcagagaacaaagggtGCATTCGCCACACCGACGT CACCACTATCTTACGTGTACTCattccagtcattgggaaggGAGAAAGAAAGAGAGAAGAACTCAAAGATGGAATCTCgattgcaatctctctcaggCTTGGCCACACCTCACACACGTG TTTGTGTGTGTACTCatcgaataagcttaatgcggagtattca
36	Register 36: [0123456789ABCFDE] AtcacgaggcccttgccttcaagaattTTTATAGAAAACGTTTGAAGAAGAAGATGATCTCTACTCactagtcctcgaaaacctcgggTGTCTCTACTATGTCTTCTCTCTTCTACTACTTACCTACTCTctca atacaatcagtcactaggGGATGGATGATCTCCACACCTCACACCGAGGAGAAACTCTgttgcaggctcggaacctggCTAGTGGTAGTGTGTGTGTGGCGCGGAGAGAAGCACTCattcacagca actcagcaggTTGGCAGCATGACTCAGCACTGCAGCAAAAGCTGGAACCTCattgccaactcaatcaaaagggtTGAGGAGGAGAGAATAAAGAAAGCTTCGAGAGAGTACTGgttcatcgcgttacca cgaaggAGTTTACACGGCGGCTCTTCCGGTTGATCTTGCCACTCATcaataatggtcatggcattggAGTTTACCGCAGCGGTTTCCACCACGAGTGTGTACTTCruggaagaagcgtcgtgaggCT GTTTGCCACACACCCGACACCCCTGTCCTCTCGACTCatacgaagtacgataccgagATGCGGTTGCGTTGTGTTGGGTTCCACACACAGCTGATTCTCnrtgggtcaatgcactccaggATCTCAAAAG AGAACTGGGATTTCTAAAGAGAGAGAAGAACTCaagtcagctcgttgaacggggGTTTACCTTTTGGCGTTTGTCTTGTCTGCTCCTACTCgaacgatatacaaccattaggGGCTCCCTACACCA CACCACGTTTGTATGATAGTTGACTCattcaatacagctgcaaaagggtTACGAGAGGAAGCTTCACACACCCACACGATCGGATGTAATCTCattccagtcattgggaaggGAGAAAGAAAGAGAGAAG AGAAACTCAAAGATGAACCTCgattgcaatctctcaggCTTGGCCACACCTCACACACGTGTTTGTGTGTGTACTCgctctcatcagagaacaaagggtGCATTCGCCACACCGACGTCTCAC CTATCTTACGTGTACTCatcgaataagcttaatgcggagtattca
37	Register 37: [0123456789ABCFED] AtcacgaggcccttgccttcaagaattTTTATAGAAAACGTTTGAAGAAGAAGATGATCTCTACTCactagtcctcgaaaacctcgggTGTCTCTACTATGTCTTCTCTCTTCTACTACTTACCTACTCTctca atacaatcagtcactaggGGATGGATGATCTCCACACCTCACACCGAGGAGAAACTCTgttgcaggctcggaacctggCTAGTGGTAGTGTGTGTGTGGCGCGGAGAGAAGCACTCattcacagca actcagcaggTTGGCAGCATGACTCAGCACTGCAGCAAAAGCTGGAACCTCattgccaactcaatcaaaagggtTGAGGAGGAGAGAATAAAGAAAGCTTCGAGAGAGTACTGgttcatcgcgttacca cgaaggAGTTTACACGGCGGCTCTTCCGGTTGATCTTGCCACTCATcaataatggtcatggcattggAGTTTACCGCAGCGGTTTCCACCACGAGTGTGTACTTCruggaagaagcgtcgtgaggCT GTTTGCCACACACCCGACACCCCTGTCCTCTCGACTCatacgaagtacgataccgagATGCGGTTGCGTTGTGTTGGGTTCCACACACAGCTGATTCTCnrtgggtcaatgcactccaggATCTCAAAAG AGAACTGGGATTTCTAAAGAGAGAGAAGAACTCaagtcagctcgttgaacggggGTTTACCTTTTGGCGTTTGTCTTGTCTGCTCCTACTCgaacgatatacaaccattaggGGCTCCCTACACCA CACCACGTTTGTATGATAGTTGACTCattcaatacagctgcaaaagggtTACGAGAGGAAGCTTCACACACCCACACGATCGGATGTAATCTCattccagtcattgggaaggGAGAAAGAAAGAGAGAAG AGAAACTCAAAGATGAACCTCgctctcatcagagaacaaagggtGCATTCGCCACACCGACGTCTCACACCTATCTTACGTGTACTCgattgcaatctctctcaggCTTGGCCACACCTCACACACGTG TTTGTGTGTGTACTCatcgaataagcttaatgcggagtattca
38	Register 38: [0123456789ABDCEF] AtcacgaggcccttgccttcaagaattTTTATAGAAAACGTTTGAAGAAGAAGATGATCTCTACTCactagtcctcgaaaacctcgggTGTCTCTACTATGTCTTCTCTCTTCTACTACTTACCTACTCTctca atacaatcagtcactaggGGATGGATGATCTCCACACCTCACACCGAGGAGAAACTCTgttgcaggctcggaacctggCTAGTGGTAGTGTGTGTGTGGCGCGGAGAGAAGCACTCattcacagca actcagcaggTTGGCAGCATGACTCAGCACTGCAGCAAAAGCTGGAACCTCattgccaactcaatcaaaagggtTGAGGAGGAGAGAATAAAGAAAGCTTCGAGAGAGTACTGgttcatcgcgttacca cgaaggAGTTTACACGGCGGCTCTTCCGGTTGATCTTGCCACTCATcaataatggtcatggcattggAGTTTACCGCAGCGGTTTCCACCACGAGTGTGTACTTCruggaagaagcgtcgtgaggCT GTTTGCCACACACCCGACACCCCTGTCCTCTCGACTCatacgaagtacgataccgagATGCGGTTGCGTTGTGTTGGGTTCCACACACAGCTGATTCTCnrtgggtcaatgcactccaggATCTCAAAAG AGAACTGGGATTTCTAAAGAGAGAGAAGAACTCaagtcagctcgttgaacggggGTTTACCTTTTGGCGTTTGTCTTGTCTGCTCCTACTCgaacgatatacaaccattaggGGCTCCCTACACCA CACCACGTTTGTATGATAGTTGACTCgattgcaatctctcaggCTTGGCCACACCTCACACACGTGTTTGTGTGTGTACTCgctctcatcagagaacaaagggtTACGAGAGGAAGCTTCACAC ACCCACGATCGGGATACTCgctctcatcagagaacaaagggtGCATTCGCCACACCGACGTCTCACACCTATCTTACGTGTACTCattccagtcattgggaaggGAGAAAGAAAGAGAGAAGAA ACTCAAAAGATGAACTCatcgaataagcttaatgcggagtattca
39	Register 39: [0123456789ABDCFE] AtcacgaggcccttgccttcaagaattTTTATAGAAAACGTTTGAAGAAGAAGATGATCTCTACTCactagtcctcgaaaacctcgggTGTCTCTACTATGTCTTCTCTCTTCTACTACTTACCTACTCTctca atacaatcagtcactaggGGATGGATGATCTCCACACCTCACACCGAGGAGAAACTCTgttgcaggctcggaacctggCTAGTGGTAGTGTGTGTGTGGCGCGGAGAGAAGCACTCattcacagca actcagcaggTTGGCAGCATGACTCAGCACTGCAGCAAAAGCTGGAACCTCattgccaactcaatcaaaagggtTGAGGAGGAGAGAATAAAGAAAGCTTCGAGAGAGTACTGgttcatcgcgttacca cgaaggAGTTTACACGGCGGCTCTTCCGGTTGATCTTGCCACTCATcaataatggtcatggcattggAGTTTACCGCAGCGGTTTCCACCACGAGTGTGTACTTCruggaagaagcgtcgtgaggCT GTTTGCCACACACCCGACACCCCTGTCCTCTCGACTCatacgaagtacgataccgagATGCGGTTGCGTTGTGTTGGGTTCCACACACAGCTGATTCTCnrtgggtcaatgcactccaggATCTCAAAAG AGAACTGGGATTTCTAAAGAGAGAGAAGAACTCaagtcagctcgttgaacggggGTTTACCTTTTGGCGTTTGTCTTGTCTGCTCCTACTCgaacgatatacaaccattaggGGCTCCCTACACCA CACCACGTTTGTATGATAGTTGACTCgattgcaatctctcaggCTTGGCCACACCTCACACACGTGTTTGTGTGTGTACTCgctctcatcagagaacaaagggtTACGAGAGGAAGCTTCACAC ACCCACGATCGGGATACTCattccagtcattgggaaggGAGAAAGAAAGAGAGAAGAAACTCAAAGATGAATCTCgctctcatcagagaacaaagggtGCATTCGCCACACCGACGTCTCACAC CTATCTTACGTGTACTCatcgaataagcttaatgcggagtattca
40	Register 40: [0123456789ABDECF] AtcacgaggcccttgccttcaagaattTTTATAGAAAACGTTTGAAGAAGAAGATGATCTCTACTCactagtcctcgaaaacctcgggTGTCTCTACTATGTCTTCTCTCTTCTACTACTTACCTACTCTctca atacaatcagtcactaggGGATGGATGATCTCCACACCTCACACCGAGGAGAAACTCTgttgcaggctcggaacctggCTAGTGGTAGTGTGTGTGTGGCGCGGAGAGAAGCACTCattcacagca actcagcaggTTGGCAGCATGACTCAGCACTGCAGCAAAAGCTGGAACCTCattgccaactcaatcaaaagggtTGAGGAGGAGAGAATAAAGAAAGCTTCGAGAGAGTACTGgttcatcgcgttacca cgaaggAGTTTACACGGCGGCTCTTCCGGTTGATCTTGCCACTCATcaataatggtcatggcattggAGTTTACCGCAGCGGTTTCCACCACGAGTGTGTACTTCruggaagaagcgtcgtgaggCT GTTTGCCACACACCCGACACCCCTGTCCTCTCGACTCatacgaagtacgataccgagATGCGGTTGCGTTGTGTTGGGTTCCACACACAGCTGATTCTCnrtgggtcaatgcactccaggATCTCAAAAG AGAACTGGGATTTCTAAAGAGAGAGAAGAACTCaagtcagctcgttgaacggggGTTTACCTTTTGGCGTTTGTCTTGTCTGCTCCTACTCgaacgatatacaaccattaggGGCTCCCTACACCA CACCACGTTTGTATGATAGTTGACTCgattgcaatctctcaggCTTGGCCACACCTCACACACGTGTTTGTGTGTGTACTCgctctcatcagagaacaaagggtGCATTCGCCACACCGACGTCTCAC CACTATCTTACGTGTACTCattccagtcattgggaaggGAGAAAGAAAGAGAGAAGAACTCAAAGATGAATCTCattcaatacagctgcaaaagggtTACGAGAGGAAGCTTCACACCT CTCAAAAGATGAACTCatcgaataagcttaatgcggagtattca
41	Register 41: [01234

Supplementary Information

	GTTTGCACACACCCGCACACCTGTTCCCTCGACTCatcacgagttcacgataccgaggATGCGTGTGCGTTGTTTTCGTTCCACACCACACGTTACTCttggttcagtcactccgaggATCCAAAG AGAAGCTGGGATTCTAAAAGAGAGAGAAACTCaagctcagctcgttaaacgaggGTTTTACCTTTTTCGCGTTTGTCTTCGTTTCGTCCTACTCgaacagatcatcaacccattaggGGCTCCCTACCACA CACCACGTTTGTATGATAGTTGACTCgctcatcagcagaacaagttggCGATCCGCACACGCACGTCACACCTATCTTACGTGTACTCattcaatcaagctcgaaggtgTACGAGAGGAAGCTTCACA CACCACACGATCGGATACTCgattcgaatatctcttcgaggCTTGCGCACACCTCACACACGTGTTTGTGTGTACTCtattccagtcagtgaggGAAGAAAGAAAGAGAGAGAAAA CTCAAAGATGAACTCatcgataagctttaatgcggtagttatca
45	Register 45: [0123456789ABECFD] AtcacgaggcccttcgtctcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCactagtcctcgaaaacctcgggTGTCTACTATGTCTTCTCTCTCTACTACTTACCTACTCctcca atcaaatcagtcactaggGGATGGATGATCCACACCTCACACGCAGGAGAGAAACTCtctgtrcagggtcgtgacactggCTAGTGGTAGATGTTGTGTGTGGCGCGAGAGAAAGCACTCtattcacagca actgcagcaggTTGCGACGATGACTGACGACTGCACGAAAAGCTGGAACCTCatggtcaactcaatcaaatggGTGAGGAGGAGAAGTAAAGAAAGCTTCGAGAGAGTACTCgttctcatcgcgtacca cgaaggAGTTTACACGGCGCTCTTTCCGGTTTGTATCTTGACACTCatcaatagtcagtgatggATGTTTACGCACGCGTTTTCCACCCACGATGTTGTACTCtgggagaaggtcgtctgaggCT GTTTGCACACACCCGCACACCTGTTCCCTCGACTCatcacgagttcacgataccgaggATGCGTGTGCGTTGTTTTCGTTCCACACCACACGTTACTCttggttcagtcactccgaggATCCAAAG AGAAGCTGGGATTTCTAAAAGAGAGAGAAACTCaagctcagctcgttaaacgaggGTTTTACCTTTTTCGCGTTTGTCTTCGTTTCGTCCTACTCgaacagatcatcaacccattaggGGCTCCCTACCACA CACCACGTTTGTATGATAGTTGACTCgctcatcagcagaacaagttggCGATCCGCACACGCACGTCACACCTATCTTACGTGTACTCattcaatcaagctcgaaggtgTACGAGAGGAAGCTTCACA CACCACACGATCGGATACTCtattccagtcagtgaggGAAGAAAGAAAGAGAGAGAAAGTCAAAAGATGAACTCgattcgaatatctcttcgaggCTTGCGCACACCTCACACACGTG TTTGTGTTGTGTACTCatcgataagctttaatgcggtagttatca
46	Register 46: [0123456789ABEDCF] AtcacgaggcccttcgtctcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCactagtcctcgaaaacctcgggTGTCTACTATGTCTTCTCTCTCTACTACTTACCTACTCctcca atcaaatcagtcactaggGGATGGATGATCCACACCTCACACGCAGGAGAGAAACTCtctgtrcagggtcgtgacactggCTAGTGGTAGATGTTGTGTGTGGCGCGAGAGAAAGCACTCtattcacagca actgcagcaggTTGCGACGATGACTGACGACTGCACGAAAAGCTGGAACCTCatggtcaactcaatcaaatggGTGAGGAGGAGAAGTAAAGAAAGCTTCGAGAGAGTACTCgttctcatcgcgtacca cgaaggAGTTTACACGGCGCTCTTTCCGGTTTGTATCTTGACACTCatcaatagtcagtgatggATGTTTACGCACGCGTTTTCCACCCACGATGTTGTACTCtgggagaaggtcgtctgaggCT GTTTGCACACACCCGCACACCTGTTCCCTCGACTCatcacgagttcacgataccgaggATGCGTGTGCGTTGTTTTCGTTCCACACCACACGTTACTCttggttcagtcactccgaggATCCAAAG AGAAGCTGGGATTTCTAAAAGAGAGAGAAACTCaagctcagctcgttaaacgaggGTTTTACCTTTTTCGCGTTTGTCTTCGTTTCGTCCTACTCgaacagatcatcaacccattaggGGCTCCCTACCACA CACCACGTTTGTATGATAGTTGACTCgctcatcagcagaacaagttggTACGAGAGGAAGCTTCACACACCACACGTCACACCTATCTTACGTGTACTCgattcgaatatctcttcgaggCTTGCGCACACCTCACACAC GTGTTGTGTGTGTACTCattcaatcaagctcgaaggtgTACGAGAGGAAGCTTCACACACCACACGATCGGATACTCtattccagtcagtgaggGAAGAAAGAAAGAGAGAGAAAA CTCAAAGATGAACTCatcgataagctttaatgcggtagttatca
47	Register 47: [0123456789ABEDFC] AtcacgaggcccttcgtctcaagaattcTTTATAGAAAACGTTTTGAAGAAGAAGATGATCTCTACTCactagtcctcgaaaacctcgggTGTCTACTATGTCTTCTCTCTCTACTACTTACCTACTCctcca atcaaatcagtcactaggGGATGGATGATCCACACCTCACACGCAGGAGAGAAACTCtctgtrcagggtcgtgacactggCTAGTGGTAGATGTTGTGTGTGTGGCGCGAGAGAAAGCACTCtattcacagca actgcagcaggTTGCGACGATGACTGACGACTGCACGAAAAGCTGGAACCTCatggtcaactcaatcaaatggGTGAGGAGGAGAAGTAAAGAAAGCTTCGAGAGAGTACTCgttctcatcgcgtacca cgaaggAGTTTACACGGCGCTCTTTCCGGTTTGTATCTTGACACTCatcaatagtcagtgatggATGTTTACGCACGCGTTTTCCACCCACGATGTTGTACTCtgggagaaggtcgtctgaggCT GTTTGCACACACCCGCACACCTGTTCCCTCGACTCatcacgagttcacgataccgaggATGCGTGTGCGTTGTTTTCGTTCCACACCACACGTTACTCttggttcagtcactccgaggATCCAAAG AGAAGCTGGGATTTCTAAAAGAGAGAGAAACTCaagctcagctcgttaaacgaggGTTTTACCTTTTTCGCGTTTGTCTTCGTTTCGTCCTACTCgaacagatcatcaacccattaggGGCTCCCTACCACA CACCACGTTTGTATGATAGTTGACTCgctcatcagcagaacaagttggCGATCCGCACACGCACGTCACACCTATCTTACGTGTACTCgattcgaatatctcttcgaggCTTGCGCACACCTCACACAC GTGTTGTGTGTGTACTCtattccagtcagtgaggGAAGAAAGAAAGAGAGAGAAAGTCAAAAGATGAACTCattcaatcaagctcgaaggtgTACGAGAGGAAGCTTCACACACCA CCACGATCGGATACTCatcgataagctttaatgcggtagttatca