

Renewable DNA Hairpin-based Logic Circuits

Abeer Eshra, Shalin Shah, Tianqi Song, John Reif, *Fellow, IEEE*

Abstract— Developing intelligent molecular systems for the desired functions is a significant current research topic in the field of nanoscience. Several materials have been explored to construct interesting systems such as molecular motors, molecular walkers, and Boolean logic circuits. Deoxyribonucleic acid (DNA) is one of the most widely used materials as the Watson-Crick base pairing makes it a versatile substrate. A significant achievement in DNA nanoscience has been the construction of large-scale logic circuits. However, most of the prior works have focused on developing the single-use DNA logic circuits. These single-use circuits can perform robust computations. However, the computing material, such as gate strands, cannot be reused to achieve the same (or a different) computation again. Such reusable behavior is essential for applications such as feedback and sequential logic computation. In this article, we propose a novel design strategy for building renewable DNA logic circuits. First, we propose a renewable DNA hairpin-based motif and, then, use this motif to implement a Boolean logic gate. Such renewable circuits make the overall sample preparation convenient and straightforward. We believe that our work will serve as a seed for the development of renewable intelligent molecular systems.

Index Terms— renewable DNA computing, DNA hairpin, Boolean logic, DNA nanotechnology, toehold-mediated strand-displacement

I. INTRODUCTION

THE programmable nature of nucleic acids has been exploited for data-storage application [1, 2], cellular and molecular imaging [3, 4], constructing complex nanostructures [5, 6] and targeted drug delivery [7, 8]. Researchers in the computing paradigm, have also substantially used DNA as a substrate for building a finite state machine [9, 10], chemical reaction networks “CRNs” [11, 12], logic and analog circuits and switches [13–16] and neural networks [17, 18]. These architectures either used *toehold-mediated strand displacement* [19, 20], or, in some cases, enzymes [21] for tuning the behavior of DNA. These and several other works were possible because of the programmability of DNA hybridization.

Several computing frameworks for constructing logic gates AND and OR, using nucleic acids, have already been proposed

[13, 22–25]. Since these simple gates can perform any complex logic operations, they are the fundamental units of any computing framework. An important milestone in the field of DNA computing was achieved by Qian et al. who were able to demonstrate a large-scale catalytic DNA circuit that computes the square root of a 4-bit binary number. Their circuit used about 130 DNA strands and was based on their proposed seesaw architecture [13, 23] (described in a supplementary section S2). Their framework can also learn to compute [17, 18]. One of the challenges with the seesaw structure (and others) is that they required several hours of computation time. Efforts have also been made to speed-up the computation time by spatially localizing DNA strands on a nanotrack or a nanostructure [26, 27]. Another critical issue is the reusability of the DNA gates that perform the computation. Most of the previous frameworks can perform robust computing. However, they can use the DNA-based gates only once.

Although dynamic DNA devices such as a DNA tweezer [28] and bi-directional DNA walkers [29, 30] have been demonstrated, not much work has been done to design a computing architecture with reusable gates. Some theoretical frameworks that can reuse gates have been proposed with and without enzymes [31–33]. These have been investigated and found to have a scalability problem [34], low signal restoration [25] or excess circuit concentration affecting the regenerated output [35]. In [33], a theoretical design is built to reuse seesaw circuits using azobenzene photo-regulation on the toeholds. In [36], Garg et al. proposed a renewable time-responsive circuit architecture. Experimental results of their design showed that renewability is possible but with undesirable intermediate structures due to unintended domain interactions. Not only does their design face a significant gate damping problem, which prevents them from reusing their computing material more than twice, but it needs higher temperatures to operate.

In this paper, inspired by the remarkable success of the seesaw architecture, we introduce a DNA hairpin-based motif that can reuse the computing gates strands. Such circuits are defined as *renewable* or *time-responsive* DNA circuits [24, 32, 33, 36] as they can compute a new output in the presence of a different set of input strands. Such responsive property is

Supplementary material for this article is available. A preliminary version of this work is available at A. Eshra, S. Shah, and J. Reif, “DNA Hairpin Gate: A Renewable DNA Seesaw Motif Using Hairpins,” *arXiv preprint arXiv:1704.06371*, 2017. This work is supported by Missions & Cultural Representation Sector, Ministry of Higher Education, Egypt and by NSF CCF1617791.

A. Eshra was with the Department of Computer Science, Duke University, Durham, NC, 27705, USA. She is now with the Department of Computer Science and Engineering, Faculty of Elec. Eng., Menoufia University, Menouf,

Menoufia, 32951, Egypt (e-mail: abeer@cs.duke.edu, abeer.eshra@el-eng.menoufia.edu.eg).

S. Shah is with the Department of Electrical and Computer Engineering, Duke University, Durham, NC, 27705, USA.

T. Song was with the Department of Computer Science, Duke University, Durham, NC, 27705, USA. He is now with the Department of Bioengineering, California Institute of Technology, Pasadena, CA 91125, USA

J. Reif is with the Department of Computer Science, Duke University, Durham, NC, 27705, USA (e-mail: reif@cs.duke.edu).

Copyright (c) 2019 IEEE

crucial since it is required for feedback circuits such as memory and flipflops. Since our framework uses a hairpin-based gate, these gates can be reversed (or renewed) quickly as the reverse process begins. The gate reversing is achieved using a set of DNA strands called *the extractors*. Our experiments show that the proposed gate architecture can be restored and reused multiple times with some signal loss in every cycle. To achieve higher restoration, we had to double the concentration of the inputs and the extractors in each cycle. Although this accumulates a higher waste, the mechanism can potentially help with saving and reusing the circuit components such as the computing gates.

Moreover, our design is simple and faster as compared to the other prior renewable circuits. One way to build large-scale circuits is to implement logic gate AND and OR and use the dual-rail logic [13, 23]. Therefore, in our supplementary document, we also propose a simple modification of our motif, to make it capable of producing either the AND or OR logic.

A. Renewable Hairpin-Gate Motif

A DNA strand containing a complementary sub-sequence can fold itself to form a secondary structure called DNA hairpin. Such a secondary structure consists of two regions, namely, a stem and a loop. The self-hybridized part of a DNA hairpin is usually referred to as its stem while the unhybridized part connecting the two ends of the stem is called its loop. A DNA hairpin with a long stem is exceptionally stable, especially, if there are no mismatches within the double-stranded stem sequence [37]. Therefore, we are introducing a motif which is a hairpin that when inputs invade it, it opens to compute. Afterward, when inputs are taken away, it closes back to be renewed.

Figure 1a shows the basic renewable motif: G is the gate, I is the input, and B is the booster. The stem of our hairpin-based gate is the double-stranded DNA (domain S_2). The loop is a single-stranded DNA (ssDNA) right arm that cascades to the downstream gate (domain S_3). In addition, two universal toeholds are used. G is a hairpin where the left toehold T_1 interacts with its complement in the middle of I to open the hairpin by strand displacement by domain S_2 . The G remains in the OFF mode until input is available to initiate strand displacement. Once the input strand invades the gate and opens the hairpin, the gate is activated (ON mode) to work with the following level. The reaction starts with I displacing the top strand of the hairpin stem with regular strand-displacement. This results in a partial dsDNA complex $G.I$. When the hairpin is open, the middle toehold T_1 becomes vulnerable to hybridization. The booster strand B is designed to initiate a strand exchange and replace I , resulting in complex $G.B$. Therefore, the booster B should aid I to invade new gates and produce more output.

Discussion of the earlier design trials can be found in Supplementary section S7.1. In case of using one universal toehold, it will be double-stranded DNA (dsDNA) within the hairpin loop. Since it is a hairpin, the open arm will always be close enough to compete with the invader I . Competition will be very high between domain S_2 in I and S_2 in G 's top to

hybridize with S_2^* in G 's bottom. Therefore, we decided to use two different toeholds. However, the use of two universal toeholds can potentially increase leaks, because one of the two toeholds in the strand exchange is supposed to be sequestered. Having T_1^* single-stranded inside the loop allows B to easily invade the gate before being opened by I . We tried decreasing that effect by making the two universal toeholds partially complementary to each other. This prevents complete hybridization between the toeholds and partially sequesters the inner toehold as well. We are still experiencing some leaks as discussed in supplementary Figure S3 (blue curve).

To avoid leak at this stage, we considered booster as an intake to the circuit. It is only present if the input is present to help to boost up the output. For instance, a 3-input gate is represented with our design by six intakes. Every intake is a set of I and B . The effect of the booster is demonstrated experimentally and shown in supplementary section S3. In the absence of the input signal, the booster exhibited a significant leak reaction with the rest of the circuit, releasing approximately 40% of the output intensity released by the input alone. This leak reaction is sufficient to account for the additional intensity of the full reaction, with the input and booster both present, and suggests that the current reaction does not act catalytically.

To detect the output, we added another reaction phase after the gate reaction via a reporter complex R . It is labelled with a pair of fluorophore and quencher molecules. The reporter R is a DNA duplex with a toehold T_2 that is complementary to the third toehold in the hairpin. As seen in Figure 1b, once the gate is ON, the third toehold will be exposed, and then R begins interacting with either $G.I$ or $G.B$ by toehold-mediated strand displacement with domain S_3 . This process releases the final detectable output.

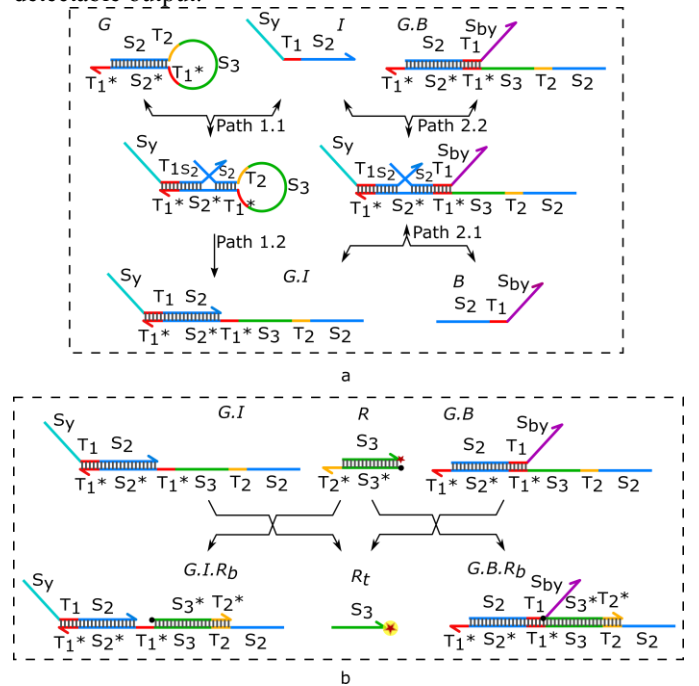


Fig. 1: Forward computation cycle for renewable hairpin-motif. (a) Basic renewable-gate reaction cycle. The gate is in OFF mode until input opens it and makes it ON and ready to work with following structure. Input binds to gate

(G) then is replaced by booster to bind to other gates. Path 1: The input strand binds to the gate to open the hairpin. Path 2: The booster is designed to bind to the input-gate complex to release input and form a booster-gate complex. Arrows indicate the reaction pathways either forward or reverse. (b) Reporting mechanism. Another cascade was added to detect the output. Reporter complex is a dsDNA with a toehold, which has a complement within the hairpin loop. Once hairpin is opened, toehold T_2 is available to react. Via toehold-mediated strand-displacement, the reporter complex hybridizes to the gate. As a result, the dye and quencher molecules are separated, causing the dye to fluoresce. This fluorescence emission indicates the reaction completion. The reporter complex hybridizes to the gate-input or gate-booster complexes. The fluorescence emission is an indication of the completion of the hybridization reaction.

B. Renewing the Hairpin-Gate Motif

To rebuild the hairpin-gate gate, we initiate the process of reversing the forward reaction by adding extracting hairpins. They have a complementary toehold to start strand displacement with either input or booster strands. These extractors remove both input and booster strands, freeing the hairpin-stem regions from hybridization. The resulting strand replacement reaction on the hairpin-gate starts from the end of the hairpin stem arm until reaching a point that weakens the prior hybridization of the attached reporter strand. At this point, a subsequence of the reporter strand becomes single-stranded, which in turn acts as an active toehold to initiate strand-displacement. This reaction will reconstruct both the hairpin-gate and the reporter complex.

Since both input and booster each have distinct arms, we used strand-displacement as an extraction mechanism. As seen in Figure 2, two extracting hairpins were introduced. Each extractor can attach to either the input strand I or booster strand B . A detailed version of the reversal process is shown in supplementary Figure S4. For brevity, we will discuss the extraction of the input strand I (the same scenario would take place with the booster strand B) using the extractor I_{ex} . The effect of the extractor I_{ex} is that two strand displacement processes take place consecutively:

- 1) The first strand-displacement step occurs when the extractor I_{ex} starts hybridization with I 's free arm, and I starts opening the extractor hairpin by displacing stem top strand of I_{ex} .
- 2) The second strand-displacement step starts when the hairpin I_{ex} is open, it starts displacing G from being hybridized with I .

The entire extraction process ends by I being extracted. A similar process is used to extract the booster strand B . After extracting both the input and booster strands; the hairpin motif has two free arms that contain two domains complementary to each other. The restoration process can be expected to proceed as follows:

- 1) The hybridization process of the hairpin stem arms starts from the ends.
- 2) When getting close to the loop (which is hybridized to R bottom strand) edges of the hybridized part will experience a breathing effect and open toeholds from R . This is due to the strength of stem hybridization.
- 3) That will allow the reporter top strand to start hybridizing with its complement, which results in pulling it away from the hairpin loop.

- 4) Finally, this will reform the hairpin and restore the reporter complex.

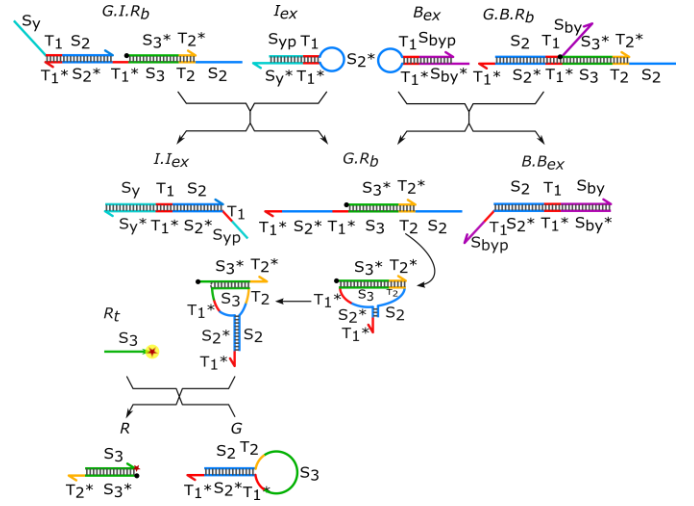


Fig. 2: Reverse mechanism (Renewing). Using the extraction mechanism, two extractors (I_{ex} and B_{ex}) were introduced to pull input and booster out from forward reaction resultant complexes. Once extracted, the gate hairpin begins to close starting from the ends of its branches. This weakens hybridization with the reporter's bottom and opens a toehold for the reporter's top to hybridize with its complement by strand-displacement. Extracted input and booster become waste. Reporter and gate are fully restored to be reused. Detailed extraction steps are shown in supplementary Figure S4.

For extractors, hairpins were chosen over single-stranded complements of input and booster. This choice was made to sequester domain S_2 within the hairpin loop to avoid blocking desired gate restoration. Extraction happens by toehold-mediated strand displacement starting from I or B distinctive arms, S_y and S_{by} respectively. ssDNA extractors would be $S_2^*T_1^*S_y^*$ for the input I and $S_2^*T_1^*S_b^*$ for booster B . In case of using ssDNA extractors, besides the desired hybridization for extraction, domain S_2 will hybridize with its un-sequestered complement in either $G.I.R_b$ or $G.B.R_b$. This hybridization will cause blocking domain S_2 in the gate which is needed later to restore G hairpin. A sketch and experiments of ssDNA extractors are shown in Supplementary Figure S5.

II. RESULTS AND DISCUSSION

The hairpin-gate motif was experimentally verified to be renewable. We used fluorescence spectroscopy to test our system by observing fluorescence from the reporter. Figure 3 shows the results of renewing the motif three times. Both kinetic experiment and polyacrylamide gel electrophoresis (PAGE) analysis show the gradual loss of signal. Those experiments were carried out in three phases. In the following $1 \times$ equals 100 nM . In the first phase, G and R were mixed in solution with concentrations of $1 \times$ and $1.5 \times$ respectively. I and B were then added to the solution with concentrations of $1 \times$ and $2 \times$ respectively, allowing the forward reaction to take place. When signal reached peak saturation, we initiated the reversal process by adding $2 \times$ of I_{ex} and $4 \times$ of B_{ex} . After having output at its lowest saturation level, the second phase

was initiated by adding I and B at a double concentration of I_{ex} and B_{ex} to reverse the backward reaction and to initiate the forward reaction again. The same process was repeated with doubling concentrations of inserts every time. Experiments were performed in Tris-acetate-EDTA buffer with 12.5 mM Mg^{2+} at 22 °C. The kinetic experiment was carried out three times to prove reproducibility. It can be found in Supplementary section S8.

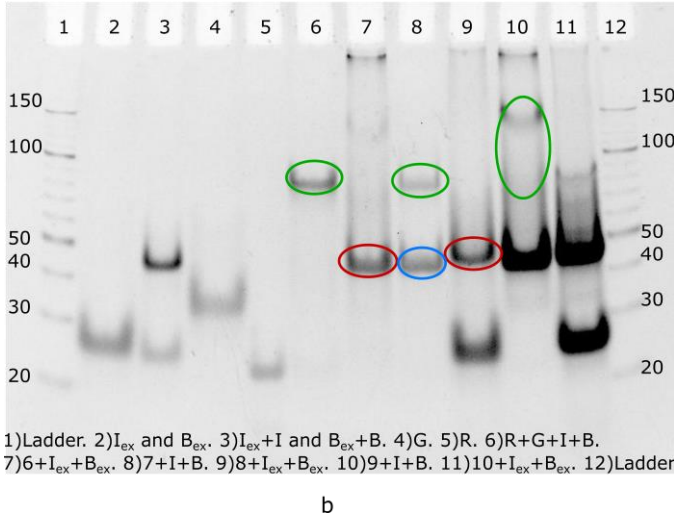
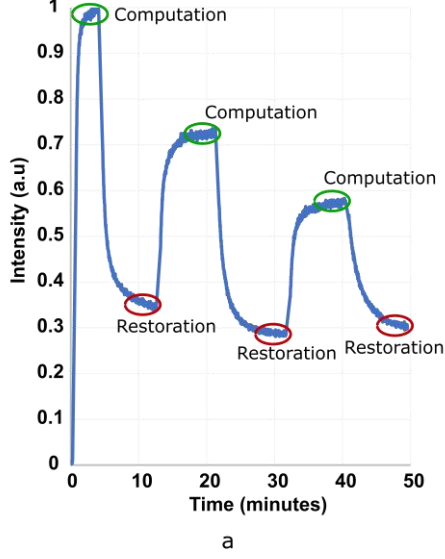


Fig. 3: System recycled three times with a gradual loss of signal. (a) Kinetic experiment. (b) PAGE analysis (all visible bands are DNA with double-stranded regions). In both figures, green circles show the result of the forward reaction after adding I and B ; and red circles show result of adding extractors I_{ex} and B_{ex} to perform the restoration. Computation result in (a) is observed by fluorescence of the FAM dye conjugated to R_b 's end. Computation result in b is the intensity of the band corresponding to structures $G.I.R_b$ and $G.B.R_b$. In (b) a blue circle on lane 8 is showing accumulating waste from the previous restoration. The intensity of the adjacent band in lane 10 shows that the concentration of waste is increasing.

Lanes of the gel in Figure 3b are as follows: Lanes 1 and 12 contain Thermo Scientific O'RangeRuler 10 bp DNA Ladder. Lane 2 contains extracting hairpins. Lane 3 contains input and booster with their extracting hairpins. It is showing two bands. The higher band has dsDNA $I.I_{ex}$ and $B.B_{ex}$. The lower band equal to the band in Lane 2 that has the extracting hairpins.

Lanes 4 and 5 contain G and R respectively. Lanes 6 and 7 contains the full reaction and restoration respectively. Green-marked band in lane 6 shows the computation result. Red marked band in lane 7 shows the extracted input and booster band. Bands that shows gate and reporter are very shallow because of low concentrations in the initial reaction Lane 8 and 9 shows computation and restoration for the second cycle. A new band marked in blue is shown in lane 8, indicating the excess waste of previously extracted input and booster. Also, there is a lower band in lane 9, and it is equal to the lane 2 band which contains only extracting hairpins. This is resulting from excess extracting addition because of concentration doubling every cycle. Lanes 10 and 11 show the third cycle of computation and restoration. It is observed that the computation signal start to fade (marked with a green oval). This fading results from output loss with every cycle. Also, the concentration of wasted extracted input and booster increase and excess of extractors increase. This is shown by band intensity increase. Both kinetic and gel electrophoresis experiments are consistent with the results.

A. Used Material and Produced Waste

In this subsection, we will give a comparison between our proposed renewable motif and the non-renewable seesaw motif. We will discuss making three cycles of computation. Taken into consideration $1 \times = 100\text{nM}$, in presented renewable experiments we used $2.5 \times$ as a circuit (concentration of $G = 1 \times$ and $R = 1.5 \times$) which is 250nM . Intakes were $3 \times$ (concentration of $B = 2 \times$ and $I = 1 \times$) which gives 300nM . Total DNA concentration needed for the first computation cycle is 550nM . For restoration, we need to add extractors in double concentrations of intakes which is $6 \times$ ($I_{ex} = 2 \times$ and $B_{ex} = 4 \times$). That gives 600nM . Second computation needs $12 \times$ that gives 1200nM . Second restoration needs $24 \times$ that gives 2400nM . Third computation needs $48 \times$ that gives 4800nM . From that, it is seen that we used the $2.5 \times$ of the circuit in the three computations. However, we used $95.5 \times$ of the material in total.

On the other hand for a non-renewable seesaw motif presented in [13], they used $5 \times$ for the circuit and $2 \times$ for the input in every cycle of computation. So, to perform three computations, they need $15 \times$ of circuit and $21 \times$ in total. We conclude that the non-renewable circuit requires approximately six times of the material needed for the renewable circuit. But regarding total material consumption, the renewable motif needs ≈ 4.5 times the material needed to perform three non-renewable calculations. Therefore, an avenue to be explored in the future is taking care of the accumulating waste. Waste can be eliminated after each computation and restoration cycle, which leaves only the circuit in solution to be reused and restored. Then we repeat until all gates are consumed. One way to do that is via conventional magnetic bead extraction [38]. We can eliminate the extracted intakes and their extractors, *i.e.* (I, B) and (I_{ex}, B_{ex}). This can be done by labelling the intakes and extractors with biotin. Once the restoration process is finished, they can be extracted from the solution by streptavidin-coupled Dynabeads. Another idea is to spatially

localize the gates on a DNA origami substrate and introduce intakes and extractors as a solution[39]. After extraction, the solution can be washed away, and only the localized circuits remain to be reused. These are open points for future research and are not explored within the scope of this article.

B. Renewable Logic OR Gate

Using the presented reversible motif, we designed a two-input OR gate illustrated in Figure 4a and 4b. It consists of two hairpin gates that act in parallel. If one of the inputs is present,

one of the gates will be opened to work with the reporter and release the output. In case both the inputs are present, both gates should open, and output will be released. In case there is no input, none of the gates will open which means there should be no output release. The four cases were experimentally verified and renewed in Figure 4c. To prove that it is reusable, we computed a combination of inputs, generated the output, restored the gate and then recomputed with a different combination of inputs. Two different experiments are shown in Figure 4d and 4e.

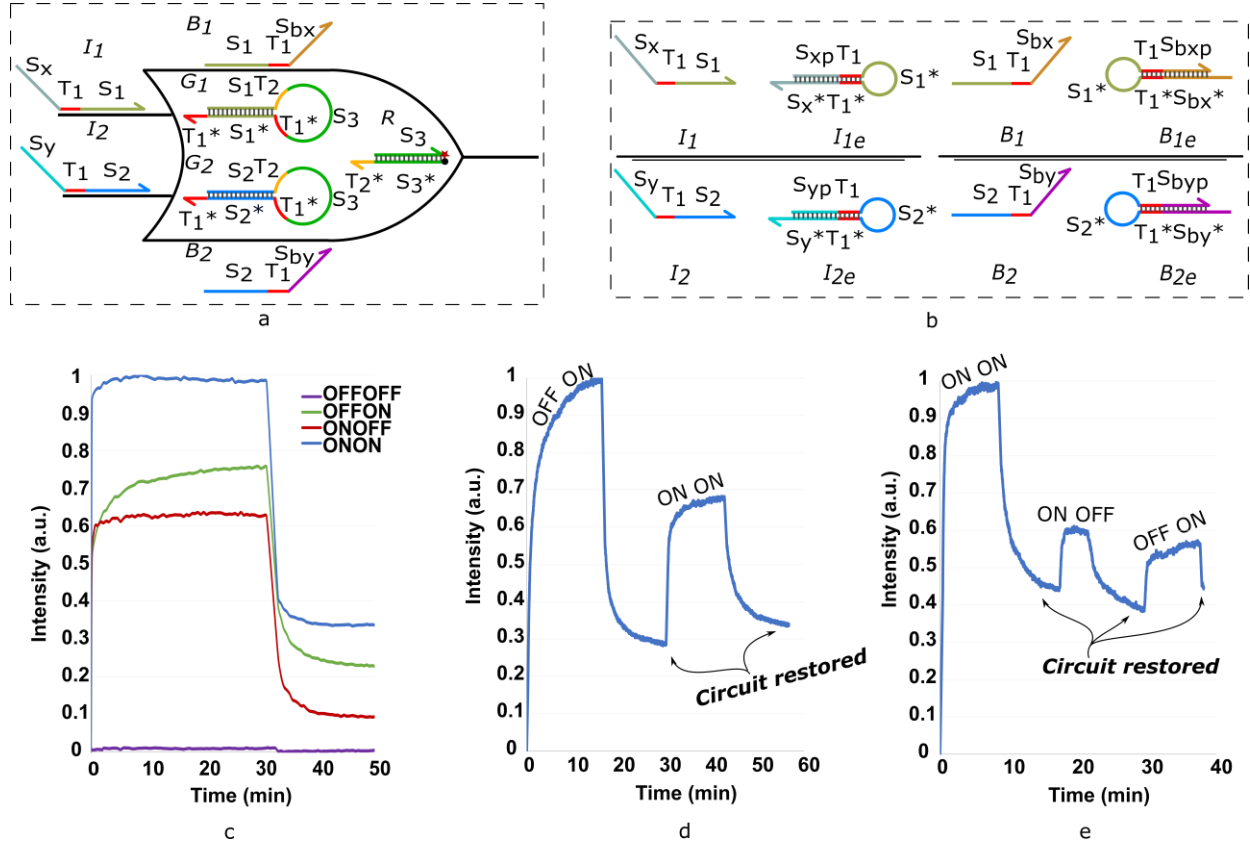


Fig. 4: Renewable 2-input OR gate with experimental results. (a) Abstract design. (b) Inputs and boosters each in-lined with their extractor. (c) Four cases of computation and restoration. OR gate should give high output if one of its inputs is high. It gives low output if all inputs are low. Fluorescence was absent only when there were no inputs. (d and e) OR gate with reversing and changing cases. (d) OFF-ON then ON-ON. (e) OFF-ON, ON-OFF then ON-ON. Reporter and gates were mixed first with relative concentrations 1.5x and 1x respectively. Afterward, input and booster strands were added depending on their presence within the experiment. ON indicates relative concentration 1x for input and 2x for a booster. OFF indicates the absence of a species concentration. For restoration, extractors I_{ex} and B_{ex} were added with concentrations that are double of preceding computation concentrations of I and B respectively. 1x= 100 nM. Experiments were performed at 22°C in Tris-acetate-EDTA buffer containing 12.5 mM Mg^{2+} .

A proposal for a set of two logic gates is presented in supplementary section S10. Using the modified design can help implement AND and OR logic circuits. This modified design was only investigated through simulations rather than experiments. However, our simulation results show that we can involve thresholding mechanism to represent the desired logic element. Using only AND and OR can constitute any combinational logic circuit by the dual-rail mechanism. We also added our extraction technique to renew the gate. Simulations show that renewing is achievable for all four cases of two input logic gate (either OR or AND). Material and Method section is discussed in supplementary material section S7. Sequences of

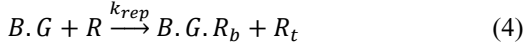
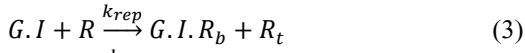
used DNA strands are available in supplementary Table S1. Method of preparing DNA solutions for experimental use is discussed. Native Gel and Kinetic experiment setup and run are explained in subsections S6.1 and S6.2.

C. Modeling of the System's Reaction Kinetics

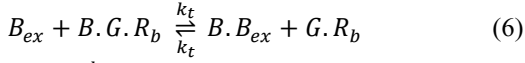
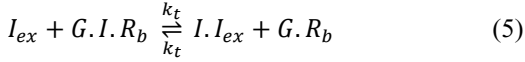
To understand our system better and identify potential leaks, we modeled our system by assuming each DNA strand as a single molecule which can diffuse and bind to the desired complementary DNA strand or undergo toehold mediated strand-displacement. The rate constant for the reporter is denoted as $k_{rep} = 1.3 \times 10^6 / M/s$ and its value if adopted from previous works [26, 40]. Additionally, since our system is

enzyme-free, and all the reaction are based on the principle of strand displacement, we assume that the rate constant for all other reactions is the same and we denote it as k_t . We model our reversible DNA system with a set of reactions shown below. Note that in (2), the forward and backward rate will be slightly different because of the presence of the extra two nucleotides (clamp) at the beginning of the gate external toehold. However, for simplification purposes, we keep the same rates in our model. Additionally, prior works [40, 41] also suggests that the rate of toehold-mediated strand displacement does not change substantially by increasing the toehold length beyond 5 – 6 nt, taking into consideration the binding strength of the strand displacement (G-C ratio).

Computation Reactions:



Restoration Reactions:



Leaky Reaction:



Simulations were performed using Microsoft's Language for Synthetic Biology (LBS) [42] package. To obtain the value of the rate constant k_t , we used maximum likelihood estimation (MLE) with initial values adopted from [13]. The range of values k_t can sweep over were specified as $10^5 - 10^6$ /M/s as reported by [40]. MLE is a common statistical technique to best approximate parameters for a defined model that can generate the observed data. The likelihood of model-parameters that maximizes the probability of generating the observed data should best approximate rate-constant of our reactions. The calculated value of rate constant value for best-fit of the model is 6.7×10^6 /M/s. Note that we assumed a full yield of products for the first cycle and changed the available concentration of input, booster, input extractors and booster extractors in the subsequent cycles to achieve a good fit between our model and data. Additionally, we also assumed that some amount of booster can leak and open the gate without input; however, the rate of leak is much slower than the desired reactions. The loss in reaction yield and booster leaks have been reported in previous studies [28, 43]. The yield of our system drops after every cycle, as observed from gel and electrophoresis data, and therefore our model needs to account for that drop. A reason for that loss is poisoned gates after each cycle, either computation or restoration. By poisoned gates, we mean the gates that were

not fully restored to the hairpin shape and not able to react again. Since there is a competition between hairpin extractor (either I_{ex} or B_{ex}) and gate hybridized with extractant (either I or B) there are two possible results. One is the desired reaction which is described in this article. The other is that extractor hairpin stem keeps fighting back, and remains as a hairpin. Therefore, the gate with the extractant stays hybridized. Another reason is that in each cycle, adding more DNA liquid to solution causes dilution of previously existed DNA. This affects the output signal as well. This can be treated by rescaling and considering the highest obtained signal is our maximum with every cycle. This is not addressed within this work.

Additionally, we used the same model to fit data for OR gate and obtained a rate constant of 6.13×10^6 /M/s which is relatively close to the rate constant for the motif. LBS fitting code for the basic motif three cycle reversal is provided in supplementary section S10. Figure 5 shows experimental and simulated data of basic motif three cycles of reversal and OR gate four cases computation and restoration.

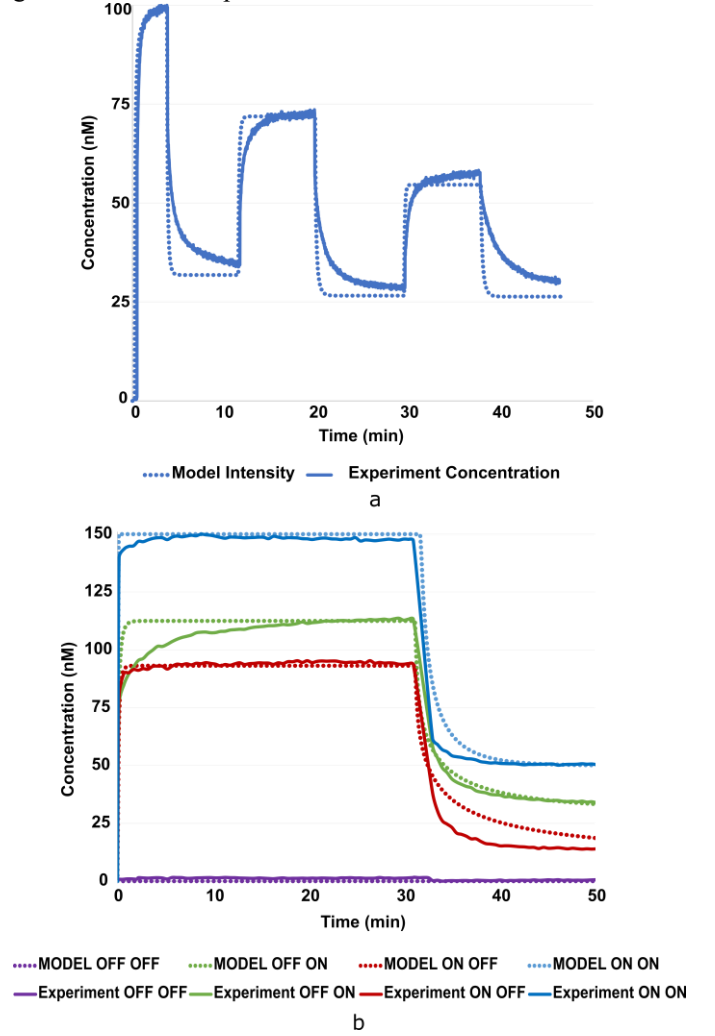


Fig. 5: Experiment modeling. (a) Modeling of three restoration cycles of the motif. Rate constant obtained is 2.743×10^6 /M/s. (b) Modeling of OR gate computation and restoration of four cases. Rate constant obtained is 2.45×10^6 /M/s. In both figures, solid lines are experimental results, and dotted lines are modeling results.

III. CONCLUSION

In this work, we have introduced a DNA computing motif with reusable computing strands. To achieve the renewability or the time-responsive behavior, we designed hairpin-based gates and extractors so they can pull-out the invader input strand. We used fluorescence spectroscopy and PAGE to demonstrate the working of our design experimentally. In addition to the simple gate demonstration, we also designed a 2-input OR gate and reported the experimental results for all the possible input combinations. Note that we used the same hairpin-based gate strands to perform three computations, consecutively, which shows that the renewability is possible. Additionally, by adding a fluorescence reporter, we showed a quasi-two layered circuit demonstrating that our design can work with more than one circuit layer. Both the gate and the reporter are restored after the reversing process and can respond to a new set of input strands.

To build complex logic circuits, we also need to implement the AND logic which, at this point, remains as the future work. Additionally, our design does produce an input and extractor waste during the gate restoration phase, which makes it a challenge to renew the computing strands beyond a few cycles. An important future direction for constructing large-scale enzyme-free renewable DNA circuits is a successful demonstration of the catalytic renewable hairpin motif. However, we believe that this study is a substantial step in that direction as our findings will aid the future research of the renewable DNA computing field.

ACKNOWLEDGMENT

The authors are grateful for the valuable discussions and support of the Reif group members. The authors would also like to thank Dr. Hieu Bui and Reem Mokhtar for their feedback. A. Eshra sincerely thanks Prof. Nawal El-Fishawy and Prof. Ayman El-Sayed for their support and cooperation.

REFERENCES

- [1] N. Goldman *et al.*, "Towards practical, high-capacity, low-maintenance information storage in synthesized DNA," *Nature*, vol. 494, no. 7435, p. 77, 2013.
- [2] S. Shah, D. Limbachiya, and M. K. Gupta, "DNACloud: A Potential Tool for storing Big Data on DNA," *arXiv preprint arXiv:1310.6992*, 2013.
- [3] J. Schnitzbauer, M. T. Strauss, T. Schlichthaerle, F. Schueder, and R. Jungmann, "Super-resolution microscopy with DNA-PAINT," *Nature protocols*, vol. 12, no. 6, p. 1198, 2017.
- [4] S. Shah and J. Reif, "Temporal DNA Barcodes: A Time-based Approach for Single-Molecule Imaging," in *International Conference on DNA-Based Computers*, DNA 24, Jinan, China, 2018: Springer International Publishing.
- [5] G. Tikhomirov, P. Petersen, and L. Qian, "Fractal assembly of micrometre-scale DNA origami arrays with arbitrary patterns," *Nature*, vol. 552, no. 7683, p. 67, 2017.
- [6] L. L. Ong *et al.*, "Programmable self-assembly of three-dimensional nanostructures from 10,000 unique components," *Nature*, vol. 552, no. 7683, p. 72, 2017.
- [7] S. M. Douglas, I. Bachelet, and G. M. Church, "A logic-gated nanorobot for targeted transport of molecular payloads," *Science*, vol. 335, no. 6070, pp. 831-834, 2012.
- [8] E. S. Andersen *et al.*, "Self-assembly of a nanoscale DNA box with a controllable lid," *Nature*, vol. 459, no. 7243, p. 73, 2009.
- [9] Y. Benenson, T. Paz-Elizur, R. Adar, E. Keinan, Z. Livneh, and E. Shapiro, "Programmable and autonomous computing machine made of biomolecules," *Nature*, 10.1038/35106533 vol. 414, no. 6862, pp. 430-434, 2001.
- [10] A. Eshra and A. El-Sayed, "An Odd Parity Checker Prototype Using DNAzyme Finite State Machine," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 11, no. 2, pp. 316-324, 2014.
- [11] Y.-J. Chen *et al.*, "Programmable chemical controllers made from DNA," *Nat Nano*, Article vol. 8, no. 10, pp. 755-762, 2013.
- [12] D. Soloveichik, G. Seelig, and E. Winfree, "DNA as a universal substrate for chemical kinetics," *Proceedings of the National Academy of Sciences*, vol. 107, no. 12, pp. 5393-5398, 2010.
- [13] L. Qian and E. Winfree, "Scaling Up Digital Circuit Computation with DNA Strand Displacement Cascades," *Science*, vol. 332, no. 6034, pp. 1196-1201, 2011.
- [14] D. Fu, S. Shah, T. Song, and J. Reif, "DNA-Based Analog Computing," in *Synthetic Biology*: Springer, 2018, pp. 411-417.
- [15] T. Song, S. Garg, R. Mokhtar, H. Bui, and J. Reif, "Analog Computation by DNA Strand Displacement Circuits," *ACS Synthetic Biology*, vol. 5, no. 8, pp. 898-912, 2016.
- [16] S. Shah, P. Dave, and M. K. Gupta, "Computing Real Numbers using DNA Self-Assembly," *arXiv preprint arXiv:1502.05552*, 2015.
- [17] L. Qian, E. Winfree, and J. Bruck, "Neural network computation with DNA strand displacement cascades," *Nature*, 10.1038/nature10262 vol. 475, no. 7356, pp. 368-372, 2011.
- [18] K. M. Cherry and L. Qian, "Scaling up molecular pattern recognition with DNA-based winner-take-all neural networks," *Nature*, vol. 559, no. 7714, p. 370, 2018.
- [19] G. Seelig, D. Soloveichik, D. Y. Zhang, and E. Winfree, "Enzyme-Free Nucleic Acid Logic Circuits," *Science*, vol. 314, no. 5805, pp. 1585-1588, 2006.
- [20] E. Graugnard *et al.*, "DNA-controlled excitonic switches," *Nano letters*, vol. 12, no. 4, pp. 2117-2122, 2012.
- [21] S. Garg, H. Chandran, N. Gopalkrishnan, T. H. LaBean, and J. Reif, "Directed enzymatic activation of 1-d DNA tiles," *ACS Nano*, vol. 9, no. 2, pp. 1072-1079, 2015.
- [22] M. N. Stojanovic, T. E. Mitchell, and D. Stefanovic, "Deoxyribozyme-Based Logic Gates," *Journal of the American Chemical Society*, vol. 124, no. 14, pp. 3555-3561, 2002.
- [23] L. Qian and E. Winfree, "A simple DNA gate motif for synthesizing large-scale circuits," *Journal of the Royal Society Interface*, vol. 8, no. 62, pp. 1281-1297, 2011.
- [24] T. Song *et al.*, "Improving the Performance of DNA Strand Displacement Circuits by Shadow Cancellation," *ACS Nano*, 2018/10/29 2018.
- [25] A. J. Genot, J. Bath, and A. J. Turberfield, "Reversible logic circuits made of DNA," *Journal of the American Chemical Society*, vol. 133, no. 50, pp. 20080-20083, 2011.
- [26] H. Bui, S. Shah, R. Mokhtar, T. Song, S. Garg, and J. Reif, "Localized DNA Hybridization Chain Reactions on DNA Origami," *ACS Nano*, vol. 12, no. 2, pp. 1146-1155, 2018.
- [27] G. Chatterjee, N. Dalchau, R. A. Muscat, A. Phillips, and G. Seelig, "A spatially localized architecture for fast and modular DNA computing," *Nature Nanotechnology*, vol. 12, no. 9, p. 920, 2017.
- [28] B. Yurke, A. J. Turberfield, A. P. Mills, F. C. Simmel, and J. L. Neumann, "A DNA-fuelled molecular machine made of DNA," *Nature*, 10.1038/35020524 vol. 406, no. 6796, pp. 605-608, 2000.
- [29] W. B. Sherman and N. C. Seeman, "A Precisely Controlled DNA Biped Walking Device," *Nano Letters*, vol. 4, no. 7, pp. 1203-1207, 2004.
- [30] J.-S. Shin and N. A. Pierce, "A Synthetic DNA Walker for Molecular Transport," *Journal of the American Chemical Society*, vol. 126, no. 35, pp. 10834-10835, 2004.
- [31] A. Goel and M. Ibrahimi, "A renewable, modular, and time-responsive DNA circuit," *Natural Computing*, journal article vol. 10, no. 1, pp. 467-485, 2011.
- [32] E. Chiniforooshan, D. Doty, L. Kari, and S. Seki, "Scalable, Time-Responsive, Digital, Energy-Efficient Molecular Circuits Using DNA Strand Displacement," in *DNA Computing and Molecular Programming: 16th International Conference, DNA 16, Hong Kong, China, June 14-17, 2010, Revised Selected Papers*, Y. Sakakibara and Y. Mi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 25-36.

- [33] X. Song, A. Eshra, C. Dwyer, and J. Reif, "Renewable DNA seesaw logic circuits enabled by photoregulation of toehold-mediated strand displacement," *RSC Advances*, 10.1039/C7RA02607B vol. 7, no. 45, pp. 28130-28144, 2017.
- [34] S. Garg, "Programming Molecular Devices using Nucleic Acid Hairpins," Ph.D., Computer Science, Duke University, Duke University Libraries, 2016.
- [35] N. V. DelRosso, S. Hews, L. Spector, and N. D. Derr, "A Molecular Circuit Regenerator to Implement Iterative Strand Displacement Operations," *Angewandte Chemie International Edition*, vol. 56, no. 16, pp. 4443-4446, 2017.
- [36] S. Garg, S. Shah, H. Bui, T. Song, R. Mokhtar, and J. Reif, "Renewable Time-Responsive DNA Circuits," *Small*, vol. 14, no. 33, p. 1801470, 2018.
- [37] S. J. Green, D. Lubrich, and A. J. Turberfield, "DNA Hairpins: Fuel for Autonomous DNA Devices," *Biophysical Journal*, vol. 91, no. 8, pp. 2966-2975, 2006.
- [38] S. Zhang, K. Wang, C. Huang, and T. Sun, "Reconfigurable and resettable arithmetic logic units based on magnetic beads and DNA," *Nanoscale*, 10.1039/C5NR06733B vol. 7, no. 48, pp. 20749-20756, 2015.
- [39] A. K. George and H. Singh, "Design of Computing Circuits using Spatially Localized DNA Majority Logic Gates," in *2017 IEEE International Conference on Rebooting Computing (ICRC)*, 2017, pp. 1-7.
- [40] D. Y. Zhang and E. Winfree, "Control of DNA Strand Displacement Kinetics Using Toehold Exchange," *Journal of the American Chemical Society*, vol. 131, no. 47, pp. 17303-17314, 2009.
- [41] N. Srinivas *et al.*, "On the biophysics and kinetics of toehold-mediated DNA strand displacement," *Nucleic Acids Research*, vol. 41, no. 22, pp. 10641-10658, 2013.
- [42] M. Pedersen and A. Phillips, "Towards programming languages for genetic engineering of living cells," *Journal of the Royal Society Interface*, vol. 6, no. Suppl 4, pp. S437-S450, 2009.
- [43] K. Montagne, R. Plasson, Y. Sakai, T. Fujii, and Y. Rondelez, "Programming an in vitro DNA oscillator using a molecular networking strategy," *Molecular Systems Biology*, vol. 7, no. 1, 2011.