Erasable and Field Programmable DNA Circuits Based on Configurable Logic Blocks

Xianjin Xiao (xiaoxianjin@hust.edu.cn)
Insititute of Reproductive Health, Tongji Medical College, Huazhong University of Science and Technology https://orcid.org/0000-0003-0455-1080

Yizhou Liu
School of Life Science and Technology, Wuhan Polytechnic University, Wuhan

Yuxuan Zhai
School of Life Science and Technology, Wuhan Polytechnic University, Wuhan

Hao Hu
Insititute of Reproductive Health, Tongji Medical College, Huazhong University of Science and Technology

Yuheng Liao
Insititute of reproductive health, Tongji Medical College, Huazhong University of Science and Technology

Huan Liu
Insititute of reproductive health, Tongji Medical College, Huazhong University of Science and Technology

Xiao Liu
Insititute of reproductive health, Tongji Medical College, Huazhong University of Science and Technology

Jiachen He
Insititute of reproductive health, Tongji Medical College, Huazhong University of Science and Technology

Limei Wang
School of Life Science and Technology, Wuhan Polytechnic University

Hongxun Wang
School of Life Science and Technology, Wuhan Polytechnic University

Longjie Li
School of Life Science and Technology, Wuhan Polytechnic University

Xiaoyu Zhou
City University of Hong Kong

Article

Keywords:

Posted Date: May 10th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-2873793/v1

Abstract

DNA is commonly employed as a substrate for the building of artificial logic networks due to its excellent biocompatibility and programmability. Till now, DNA logic circuits have been rapidly evolving to accomplish advanced operations. Nonetheless, the process of creating DNA logic circuits according to personal needs (logical truth table) requires extensive knowledge on digital circuits. Moreover, even after the researchers endeavor to build a DNA circuit, it lacks field programmability and thereby being disposable and inconvenient. Herein, inspired by the Configurable Logic Block (CLB) paradigm in silicon digital circuits, we present the CLB-based field-programmable DNA circuit that uses clip strands as its operation-controlling signals. It substantially simplifies the construction of desired circuits by establishing the relationship between circuits and operation-controlling strands. Additionally, the field programmability enables users to realize diverse functions with limited hardware. We firstly constructed CLB-based basic logic gates (OR and AND), and effectively demonstrate their eras ability and field programmability. Furthermore, by simply adding the appropriate operation-controlling strands, we achieved multiple rounds of switch among 5 different logic operations on a single two-layer circuit. In addition, we successfully built a circuit to implement two fundamental binary calculators: half-adder and half-subtractor, proving that our design could imitate silicon-based binary circuits. Finally, we built a comprehensive CLB-based circuit that enabled multiple rounds of switch among 7 different logic operations including half-adding and half-subtracting. Overall, the CLB-based field-programmable circuit greatly streamlines the process to build DNA circuits and immensely enhances their practicability. We believe our design could be widely used in DNA logic networks due to its efficiency and convenience.

Introduction

As one of the most important branches in the field of bioengineering, the artificial logic networks are committed to using molecules to simulate the logic operations of silicon-based digital circuits with the ultimate goal of constructing submicroscopic computers[1–5]. Compared with the existing artificial molecules, DNA is naturally biocompatible[6], so it is easier to accommodate to the aqueous biological environment[7–9]. At the same time, the Watson-Crick base pairing principle gives DNA a high degree of programmability. Therefore, researchers may precisely regulate the degree of binding between DNA strands and the reaction rate by designing the arrangement of base pairs[10–17]. Owing to the advantages listed above, DNA is frequently used as a substrate for the construction of logic circuits[18–22]. When building a DNA-based logic circuit, researchers frequently take the following procedures: First of all, the researchers make the logical truth table according to their own needs, and then design the digital circuit diagram. After that, by ingeniously designing the sequence of DNA strands, the researchers can transform the digital circuit diagram into a DNA reaction network[23,24], which is often based on toehold-mediated strand displacement (TMSD)[13, 24–29] or enzyme-catalyzed reactions[30–32].

Until now, several kinds of DNA reaction networks with their own advantages have been proposed, and their construction pathways have been considerably mature. The seesaw gate[4,27], which was proposed

by Winfree, could complete the basic OR and AND logic operations by setting the threshold gate to different concentrations. And the low leakage guaranteed its excellent cascading ability. While the polymerase-based single-strand gate provided a new idea, that is, to complete the strand displacement by an enzyme-catalyzed reaction[32]. Benefiting from the high efficiency of enzyme catalysis, the speed of DNA calculation has been greatly improved so that the 4-bit square operation can be completed within 1 hour. Furthermore, DNA-based switching circuits manage to simplify the circuit design so that it used as few DNA strands as possible to achieve equivalent operations[33]. As a result, the complexity of the DNA circuit is greatly reduced. Overall, the transformation from digital logic diagrams to DNA logic circuits and the construction of DNA circuits have been fully accomplished and will be further enriched and optimized. However, the process of generating digital logic circuits based on personal needs (logical truth table) still depends on the researchers' personal experience and knowledge in the field of digital circuits. Therefore, it has become the rate-limiting step for the construction of DNA circuits. Typically, researchers have to calculate the type and number of logic gates needed by the circuit, as well as the cascade relationship between them. Such calculations are very complex and time-consuming. For example, the construction of a dual-rail four-bit square-root circuit already involves the arrangement and routing of 12 AND/OR gates[33]. It is too complex for researchers in the field of bioengineering, not to mention that DNA circuits are still developing to be more complex and cascaded[34,35], so the difficulty and time cost of corresponding circuit design will be further increased. To sum up, DNA circuits have been developing rapidly so that they could perform advanced functions. However, there is an urgent need to establish a simpler and highly adjustable design paradigm to realize the rapid design and construction of DNA circuits.

To achieve the forementioned goal, digital circuits give us a good inspiration: the Configurable Logic Block (CLB). Essentially, it is a kind of circuit design paradigm, based on which circuits have the potential to perform a variety of logic operations, and the actual logic of the circuit is determined by a set of operation-controlling signals[36,37]. And the corresponding relationship between the operation-controlling signals and the logic operation is presented in the form of a table. Thus, while designing the circuit, the researcher can simply look up the table to find out the corresponding operation-controlling signals of the target logic and then add it to the CLB to get the desired circuit[38,39]. As an example, Fig. 1a shows a dual-rail two-input CLB. A, A', B and B' are the input signals, while O is the output signal, and (C0,C1,C2,C3) is a 4-bit operation-controlling signal. Only when the value of the signal is 1, its corresponding logical relationship can be transmitted to the output. As shown in Table S1, the operation-controlling and the corresponding logic operation are listed.. If a researcher wants to obtain an XOR circuit, he will not have to consider the arrangement and routing of the basic logic gate but will instead directly look up the table and input (0,0,1,1) to the CLB at the operation-controlling end. Hence, the CLB helps researchers avoid the difficulty of artificially designing circuits by establishing the relationship between circuits and operation-controlling signals directly. And with the advantage that electronic circuits can be reset and reprogrammed multiple times, the performance and strengths of the CLB design paradigm can be fully demonstrated. Users can reprogram a CLB-based circuit multiple times, allowing users to realize unlimited functions with limited hardware resources (i.e., a limited number of CLB-based circuits). The

paradigm of CLB has been widely applied in silicon-based computation systems and has greatly facilitated the booming development of silicon-based computers[40]. Conclusively, constructing CLB-based DNA circuits with reference to digital circuits would be an important development direction in the field of molecular computing.

However, constructing a CLB-based DNA circuit would face two major difficulties. First, the sequence design is needed to guarantee an accurate linkage between the logic operation and the operation-controlling. Second, such an exquisite multi-strand reaction system should be able to be reprogrammed so that the power of CLB could be fully unleashed. As far as we know, only a few DNA circuits could be reset[41–47], letting alone being reprogrammed.

Herein, we have utilized the allosteric-clip-toehold mediated strand displacement reaction, which was previously proposed by our group[41], to realize the CLB-based field-programmable DNA circuit. As shown in Fig. 1b, by regarding the clip strands as the operation-controlling signals, we realized the precise correspondence between the logic functions and the controlling signals. In cascaded DNA circuits, we could set up multiple clip addition sites, just like the positions C0 ~ C3 in Fig. 1a. At each site, only clip strands could provide the toehold for the invading strand to complete the strand replacement reaction. Consequently, adding clips with different toeholds may result in different reactions, i.e., different logic functions. Finally, we established the relationship between the clip and the logic operation, and listed them in a table for the researchers' query. On the other hand, our previous experiments have proved that complementary strands of clips and inputs can reset the circuit to its initial state. Taking advantages of this, we could add another set of clips according to the table after the erasion to switch the circuit to another logic function. To sum up, researchers can obtain desired logic circuits more easily with our design, and the field programmability further reduces the time and material costs of building DNA circuits.

Result and Discussion

1. Construction of CLB-based primary logic gates

For logic gate circuits, AND and OR gates are the most fundamental and simple logic gates. Theoretically, any circuit can be realized by the combination of the above two gates. So, first of all, we explored how to construct such basic logic gates in the CLB mode. We used the previously reported allosteric clip-mediated strand displacement reaction as the basic principle to realize those 2 logic gates. As shown in Fig. S1a, an allosteric clip consists of the following domains from 5' to 3': domain T with 14-nt for binding with gate:output, domain A with 7-nt for the allosteric reaction, domain I with 14-nt for binding with input, and finally domain R with 8-nt for the reset reaction. The clip was first hybridized with gate:output through domain T. Then the domain A and gate:output would undergo an allosteric reaction, a strand displacement reaction that would open the proximal end of the dsDNA and thereby assist the input strand to invade the dsDNA by taking domain I as a toehold. Fig. S1b-c illustrated that when the

concentration of FAM:BHQ duplex was 100 nM, the addition of a 120-nM clip and 240-nM input would be enough to obtain the maximum output.

Here, we chose the allosteric clip over the traditional clip so as to speed up the reaction while reducing crosstalk. Fig. S2a showed the principle of a secondary circuit realized by clips and allosteric clips respectively. The principle based on the traditional clip may cause crosstalk between input and output, thereby interfering with the desired reactions. Moreover, according to the fluorescent intensity shown in Fig. S2b, the allosteric clip apparently increased the conversion ratio of the reaction significantly.

After determining the reaction principle, we then tried to construct the two basic logic gates: OR gate and AND gate. As shown in Fig. 1c, the OR gate was made up of two clips with different domain I and a gate:output duplex. Each clip was able to hybridize with the gate:output, and its corresponding input could initiate the clip-mediated strand displacement, executing the function of an OR gate. As for the AND gate, it consisted of a gate:output duplex, a clip strand (7-and1), and two input strands (8-input-a1 & 9-input-a2). The first step was the combination of clip and gate:output as well, then part of the input-a1 would hybridize with domain I. While input-a2 could hybridize with input-a1 through a complementary section of 14-nt and provide the strand-displacement section. Consequently, together with input-a1, input-a2 could invade gate:output and displace the output strand. In other scenarios, when either input existed alone, the conversion ratio and rate of the strand displacement reaction would be extremely low due to the lack of a toehold or strand displacement section. The experimental results in Fig. 1d-e demonstrated the function of the proposed logic gate in dealing with different inputs. For the basic one-layer circuit, allosteric clip-based strand displacement reactions possessed a considerable reaction rate, so that either logic could be completed within half an hour. At the same time, its low leakage created a good signal-to-noise ratio, which made it possible to implement subsequent reprogramming and cascaded circuits. Furthermore, the reaction speeds of the two logic gates were essentially equal, despite the fact that the AND gate required more complexity than the OR gate since the inputs needed to be complementary. And this ensured that in a cascaded circuit, the output time of the logic gates at the same layer was basically identical. In a word, using allosteric clips, we demonstrated the feasibility of our proposed basic CLB-based logic gates: the AND gate and the OR gate.

We would also like to discuss the advantages of our proposed logic gates. Compared with the traditional AND gate, the most significant advantage of the newly proposed AND gate was that it could share the same gate:output strands with the OR gate. Therefore, while constructing an AND gate and an OR gate, our design only needs 2 single strands due to its high integration, whereas the traditional circuit needs 5. Much more importantly, those two logics can be transformed back and forth by simply adding different clips. Such field programmability was the foundation for building CLB-based DNA circuits, and it would become more and more powerful with the increase in circuit complexity.

2. The erasability and field programmability of CLB-based circuits

We then would like to verify the eras ability and field programmability of the CLB-based circuit. According to our previous research, the initial state of the circuit could be erased by adding the complementary strands of clip (C-clip) and input (C-input). The detailed mechanism was illustrated in Fig. 1c. C-clip, taking Domain R as a toehold, reacted with the clip strand to form a waste duplex Clip:C-clip. The output strand then dissociated the input via TMSD. At the same time, the hybridization of C-input and input also promoted the separation of input and gate. In this way, any kind of logic can be erased to its "initial state", where only gate:output and wasted dsDNA existed. Gate:output can be further used, but wasted Clip:C-clip and Input:C-input are blunt-ended and inert. So far, the reaction system has been reset. Then, clips corresponding to other logic according to the logic operation table could be added so that the circuit could be reprogrammed. The results of reusing and switching the circuit's logic were shown in Fig. 2a-c, respectively. First, 100-nM FAM:BHQ duplex was added. Then, 120-nM operation-controlling strand (3-or1 & 4-or2 for OR gate, 7-and1 for AND gate) and 240-nM their corresponding input (5-input-o1 & 6-input-o2 for OR gate, 8-input-a1 & 9-input-a2 for AND gate) were added. The reaction was considered sufficient when the fluorescence curve reached a plateau. After that, C-clip (10-c-clip-1 & 11-c-clip-2 for OR gate, 14-c-clip-3 for AND gate) and C-input 12-c-input-1 & 13-c-input-2 for OR gate, 15-c-input-3 & 16-c-input-4 for AND gate with the same concentration as clip/input were added to erase the circuit’s logic functions to its initial state. Hence, a new round of operation-controlling strands and input strands could be added, which could either be different or the same to the last round. In order to demonstrate the versatility of this field programmability, we enumerated all possible inputs for each logic: (0,0) (0,1) (1,0) (1,1). The output results of each round were in line with the predicted outcomes. As the clip and C-clip were sequentially added to the reaction system, the fluorescence signals rose and fell accordingly, and the amplitude of the fluorescence signal did not significantly attenuate. The fluorescent signal of each round was close to 100% yield, accompanied by a relatively low leakage. In terms of reaction rate, all of the forward reactions were completed within 30 minutes. Overall, the above data proved that a CLB-based circuit could be erased and field programmed between AND and OR logic gates.

It's important to note that we increased the clip and input concentrations in each round by 10 nM compared to the previous round in order to lessen the impact of waste strands on the reaction rate and extent. And we noted that during the reprogramming process, the volume increase caused by sample addition may dilute the fluorescence intensity. So, we corrected the fluorescence signals according to the volume variation and then calculated the output percentage of each forward reaction. The formula was as follows: $S_A = \frac{S_0 \times V_A(\mu l)}{50(\mu)}$. where S_0 stood for the raw fluorescence signal value,V_A for the volume of the sample at the time of measurement, andS_A for the corrected fluorescence signal.

3. A two-layer "X-AND" cascaded circuit based on CLB logic gates to realize multiple logic operations

After verifying the feasibility of basic CLB-based logic gates, we tried to build a cascaded circuit containing more logical operations to further reveal the power of the CLB design. The cascaded circuit was illustrated in Fig. 3a and named as "X-AND". In the first layer, we chose to place an AND gate rather than an OR gate. Because the AND logic must guarantee a low leakage when the input is (0,0), (0,1), or

(1,0), while the OR gate only face the leakage problem when the input is (0,0). Therefore, the "X-AND" gate posed a greater challenge to our leakage-control strategy. By exploring the construction principle of "X-AND", we could improve the anti-leakage techniques, and apply them to the more complex circuits so as to strengthen the robustness of the CLB-based circuits. As shown in Fig. 3a, there existed two positions in the "X-AND" circuit where the operation-controlling strand could be added, and both of them were located in the second layer of the circuit. By adding different clips, each location was able to realize the function of OR, AND, or simply Translation. Its complete logic operation table was shown in Fig. 3b. Take "A AND B AND C" for example. 19-or11 and 24-and21 should be added at positions 1 and 2, respectively. When 30-input-B and 31-input-C coexisted, they would form a duplex through a 14-nt complementary region. The input duplex then invaded gate2:output2, eventually dissociating output2. At the same time, 25-input-A, using 19-or11 as a toehold, invaded gate1:output1 so that output1 could be dissociated. After that, output1 and output2 would form a duplex just like input-B and input-C. Finally, the FAM strand could be displaced off, and the whole circuit executed the function of "A AND B AND C". As for the reprogramming process, it could be accomplished by adding C-clips (strand number 33, 36) and C-inputs (strand number 37, 40, 41) to the circuit. Then, the FAM strand, as the invader, disassociated And1:And2 from the BHQ strand. Consequently, output1 and output2 took their 14-nt single-strand segment as toehold to hybridize with gate1 and gate2 respectively, completing the reset of the second layer. Since the entire circuit have restored to the "initial state", a new set of clip and input strands could be added to reprogram the circuit.

Figure 3d-e and Fig. S3,S4 showed that all the five logic operations of "X-AND" listed in the truth table can be successfully realized. In all logic operations, the signal reached to plateau within 90 minutes, which was slower than that of basic logic gates but still relatively fast as a cascaded circuit. Moreover, in some scenarios, the leakage of the "X-AND" circuit was even below the detection limit of the instrument, so it presented a straight line with a slope close to 0.

The reprogramming process was similar as described above. When forward reactions were completed, C-clips and C-inputs were added to initiate the erasing reaction, so the fluorescence signal began to decrease. However, as a secondary circuit, the decline of the fluorescence signal could only indicate that the reporter, not the whole system, had been successfully reset. So, we chose to place the system at 37°C for 90 minutes, even though the fluorescence signal had already declined to the baseline level. After that, new clips and inputs could be added to complete the following round of operations. As shown in Fig. S5 and Fig. 3f, we performed reusing and logic switching to (A AND B AND C AND D) for three rounds. Each round's reaction demonstrated clear distinction between the leakage and the major signal due to the excellent control of the leakage. Never did a leakage signal exceed 25% of the main signal. However, as shown in Fig. 3g, the signal's amplitude in reusing displayed a trend of declining. The yield of the second-round reaction was 70%, and it further decreased to 60% for the final round. This might be caused by the high complexity of the reset reaction. As a result, gate and FAM were not sufficiently restored to their initial states for the subsequent round of reaction. In order to maintain the fluorescence signal's amplitude, we attempted to simplify the second and third round of reactions. (A AND B AND C AND D) was operated in the first round; (A AND B AND C) in the second round; and (A AND B) in the third round. By gradually simplifying logic operations, we kept the fluorescence signal amplitude stable with an 80%

output yield in the final round. Therefore, as shown in Fig. 3h, its signal-to-noise ratio was significantly improved compared to reusing. To sum up, we successfully constructed a cascaded CLB-based field-programmable "X-AND" circuit with two operation-controlling sites. The system possessed the potential to implement at least 5 different logic operations, and the optimization of leakage ensured the robustness of the circuit. The above data firmly proved that CLB-based design can be used to build cascaded circuits, which further expands its practicability.

It was worth noting that we adjusted the length of the "incumbent toehold". As shown in Fig. 3c, it referred to the segment whose dissociation was spontaneous rather than caused by being displaced by an invading strand[13]. Such a domain acted like a clamp, hindering the gate-gate leakage between output1 and output2. Moreover, it also provided resistance to the forward reaction, reducing the extent of the forward reaction. According to the tradeoff shown in Fig. S6, we set the length of the incumbent toehold to 5-nt to guarantee a high signal-to-noise ratio.

4. Construction of CLB-based half-adder and half-subtractor

Since the ultimate goal of a DNA logic circuit is to build a biological computer on a submicroscopic scale, it is an inevitable trend to imitate and replace the silicon-based binary operation circuit. Based on the progress we have made, we further wanted to explore whether CLB-based circuits were compatible with binary operation circuits simulated by DNA. We chose half-adder and half-subtractor for the exploration because they have been generally simulated by various principles of DNA circuits[26,41,48–50].

In brief, the most important difference between the previous cascaded circuit and the half-adder/subtractor was that the latter had two outputs: sum and carry for the half-adder, and difference and borrow for the half-subtractor. Both Sum and Difference = (A XOR B), Carry=(A AND B), Borrow=(A' AND B). The process of integrating the above two logic circuits into one comprehensive CLB-based circuit was shown in Fig. 4a. The circuit was designed to have three operation-controlling sites. Position 1 and Position 2 were located in the second layer, where different clips were added to implement Translation/OR/AND, otherwise the corresponding component would be silenced since no clip was added to the site. Position 3 was located at the output end of the Carry/Borrow signal, where different clips decided which upstream components the reporter would connect: the AND gate or the inputs directly. The FAM channel represented SUM/Difference while the HEX channel represented Carry/Borrow. When 53-and12 and 56-and22 were added to Position 1 and Position 2 respectively, and the HEX reporter was connected to the AND gate by 57-trans and 58-and31, a half-adder was built. When the connection was switched to the inputs by adding 59-and32, the circuit would become a half-subtractor. As shown in Fig. 4c-d and Fig. S7, 100-nM FAM:BHQ and HEX:HBHQ, 240-nM gate1:output, gate2:output and gate3:ouputA*, 120-nM or-fi, was added first. Then, adding 240-nM corresponding operation-controlling and 480-nM input strands enabled the circuit to operate correct logic operations. Both FAM and HEX channel reached to a plateau in 90 mins with leakage controlled below 35%. It is worth noting that in order to ensure the performance of the half-adder and half-subtractor, we have made a series of

adjustments. We used fan-in strategy (the outputs of two upstream AND gates were used as inputs of the downstream gate) to replace the OR gate at the FAM reporter. This simplification eliminated the need to create another clip and its corresponding output, making the reprogramming reaction much easier. Furthermore, our preliminary findings shown in Fig. S8 indicated that the 7-nt incumbent toehold in the HEX reporter, which was consistent with the FAM reporter, provided too much resistance to the invader strand. As a result, there was almost no signal observed in the HEX channel. So we shortened length of the incumbent toehold to 4-nt.

We also noted that though the leakage did not hindered the major signal, the half-adder and half-subtractor circuit tended to have a greater leakage problem than previous logic circuits. Because both the input of the half-adder and the half-subtractor required fan-out (the output of one upstream gate was used as the input of two downstream gates at the same time), the input concentration needs to be doubled (480 nM). And in the dual-rail half-adder/ half-subtractor, the inputs first went into the AND gate in the second layer. Consider the case of (0,0), each AND gate had only one input, hence did not output the major signal. However, the higher concentration of input would still push the reaction to proceed, thus increasing the leakage signal. Also, the HEX signal rose more slowly in half-adder. Because compared to the half-subtractor, where inputs directly reacted with the HEX reporter, the HEX signal of the half-adder was output by a secondary circuit, which was more time-consuming.

In summary, using the CLB paradigm, we managed to integrate the half-adder and half-subtractor into one circuit, so researchers only need to add different operation-controlling strands to obtain corresponding computing elements.

5. A comprehensive field-programmable CLB-based DNA device enabling 7 logic operations including half-adding and half-subtracting

Furthermore, we tried to expand the function of the circuit by adding different combinations of operation-controlling strand based on the half-adder/half-subtractor circuit, so that the circuit became a comprehensive field-programmable CLB-based DNA nanodevice. Various combinations of operation-controlling strands in positions 1–3 and their corresponding logic operations were listed in Fig. 4b. Their reaction principles and experimental verifications were illustrated in Fig. S9 and Fig. 4c-i. Besides the half-adder and half-subtractor, other logic operations also showed a good signal-to-noise ratio and were able to complete the reaction within 90 minutes.

After verification of multiple logic operations, we tried to reprogram this circuit. First, we investigated the field programmability of the operations that output FAM only. We selected XOR and (1 OR 2 OR 3 OR 4) as representatives, and the result shown in Fig. 5a and Fig. S10a demonstrated that both logic operations could be reused three times. Figure 5b further proved the above 2 logics could be switched to each other back and forth. Based on this, we next tried the logical switching of the half-adder and the half-subtractor. As shown in Fig. 5c-d and Fig. S10b-c, the circuit could perform three logics in the order of XOR-half

adder/half subtractor-XOR, or tow logics in the order of half adder/half subtractor-XOR. In all, we have built a CLB-based DNA circuit enabling seven logic operations.

It was worth noting that in (1 OR 2 OR 3 OR 4), the output signals of (1,1,0,0) and (0,0,1,1) were significantly higher than 100%. The signal increase might be attributed to the fan-in strategy, which resulted in 480-nM output strand invading the reporter in the two preceding cases. This conflicted with the maximum output concentration we had previously explored, but we consider it acceptable because the results were reproducible and did not prevent us from drawing a cutoff line between leakage and the main signal.

Based on the current results, we foresee that the function of a CLB-based circuit can be further expanded. CLB is frequently used as the basic component in cascading digital integrated circuits. Despite the complexity of the current circuit, each circuit only has one CLB unit. As a result, we can create intricate circuits in the future that contain cascaded CLB units, enabling this circuit to perform more logic operations. On the other hand, the release of the operation-controlling strand (clip) of the circuit is controlled artificially. However, we envision that with a delicate design, the release of the operation-controlling strand can be regulated by upstream DNA strand displacement reactions or aptamers. So, the logic operation of the CLB can also be regulated accordingly. This will encourage the use of DNA circuits in the simulation of cell behavior. Overall, the computing potential and multiple regulatory methods endow the CLB-based DNA circuit with a wide range of application prospects.

Conclusion

In summary, we have built the field-programmable CLB-based DNA circuit for the first time. Utilizing allosteric-clip-toehold based strand displacement reaction, we managed to design and construct two basic logic gates: AND and OR. Additionally, we used C-clip and C-input to complete three consecutive reprogramming of the basic logic gate, which included both erasing and logic-switching. Based on the above design, we further constructed a cascaded CLB circuit "X-AND" that could execute 5 different logic functions and verified the field programmability of the most complex one, “A AND B AND C AND D”. Moreover, using the dual-rail design, we successfully constructed two basic CLB-based binary calculators: a half-adder and a half-subtractor. Finally, through different combinations of operation-controlling strands, we could use the comprehensive CLB-based circuit to realize 7 different logic operations and reprogram it according to our needs.

Conclusively, the CLB design paradigm greatly streamlines the process for scientists to build DNA circuits. The CLB-based DNA circuit has the potential to execute a variety of logic operations, and researchers can obtain the desired circuit only by adding a group of operation-controlling strands, allowing researchers to use the same circuit for multiple attempts and changes. We believe that our new circuit design can make DNA circuit programming more efficient and convenient, thereby accelerating the birth of a real DNA-based molecular computers.

Declarations

Authors Contribution

Yizhou Liu performed the theoretical analysis, designed the experiments, conducted most of the experiments, analysed the data. Yuxuan Zhai, Hao Hu, Yuheng Liao, Huan Liu, Xiao Liu, Jiachen He, Longjie Li and Xiaoyu Zhou conducted some of the experiments and participated in the discussion of the work. Xianjin Xiao supervised the whole work.

Acknowledgements

This work was financially supported by the National Natural Science Foundation of China [81871732]; the National Key Research and Development Program of China [2021YFC2701402]; the Open Research Fund of State Key Laboratory of Bioelectronics, South-east University [Sklb2021-k06]; the Open Foundation of NHC Key Laboratory of Birth Defect for Research and Prevention (Hunan Provincial Maternal and Child Health Care Hospital) [KF2020007]; and the Open Foundation of Translational Medicine National Science and Technology Infrastructure (Shanghai) [TMSK-2021-141].

Conflict of interest statement. None declared.

References

1. Fu, T.; Lyu, Y.; Liu, H.; Peng, R.; Zhang, X.; Ye, M.; Tan, W. DNA-Based Dynamic Reaction Networks. Trends Biochem. Sci. 2018, 43 (7), 547–560.
2. Song, X.; Reif, J. Nucleic Acid Databases and Molecular-Scale Computing. ACS Nano 2019, 13 (6), 6256–6268.
3. Zhao, S.; Yu, L.; Yang, S.; Tang, X.; Chang, K.; Chen, M. Boolean Logic Gate Based on DNA Strand Displacement for Biosensing: Current and Emerging Strategies. Nanoscale Horizons 2021, 6 (4), 298–310.
4. Qian, L.; Winfree, E.; Bruck, J. Neural Network Computation with DNA Strand Displacement Cascades. Nature 2011, 475 (7356), 368–372.
5. Zhang, C.; Ge, L.; Zhuang, Y.; Shen, Z.; Zhong, Z.; Zhang, Z.; You, X. DNA Computing for Combinational Logic. Sci. China Inf. Sci. 2019, 62 (6), 1–17.
6. Fan, D.; Wang, J.; Wang, E.; Dong, S. Propelling DNA Computing with Materials' Power: Recent Advancements in Innovative DNA Logic Computing Systems and Smart Bio-Applications. Adv. Sci. 2020, 7 (24), 1–25.
7. Fern, J.; Schulman, R. Design and Characterization of DNA Strand-Displacement Circuits in Serum-Supplemented Cell Medium. ACS Synth. Biol. 2017, 6 (9), 1774–1783.
8. Groves, B.; Chen, Y. J.; Zurla, C.; Pochekailov, S.; Kirschman, J. L.; Santangelo, P. J.; Seelig, G. Computing in Mammalian Cells with Nucleic Acid Strand Exchange. Nat. Nanotechnol. 2016, 11 (3), 287–294.

9. Chen, Y. J.; Groves, B.; Muscat, R. A.; Seelig, G. DNA Nanotechnology from the Test Tube to the Cell. Nat. Nanotechnol. 2015, 10 (9), 748–760.
10. Yin, J.; Wang, J.; Niu, R.; Ren, S.; Wang, D.; Chao, J. DNA Nanotechnology-Based Biocomputing. Chem. Res. Chinese Univ. 2020, 36 (2), 219–226.
11. Wang, J. S.; Zhang, D. Y. Simulation-Guided DNA Probe Design for Consistently Ultraspecific Hybridization. Nat. Chem. 2015, 7 (7), 545–553.
12. Zhang, J. X.; Fang, J. Z.; Duan, W.; Wu, L. R.; Zhang, A. W.; Dalchau, N.; Yordanov, B.; Petersen, R.; Phillips, A.; Zhang, D. Y. Predicting DNA Hybridization Kinetics from Sequence. Nat. Chem. 2018, 10 (1), 91–98.
13. Zhang, D. Y.; Winfree, E. Control of DNA Strand Displacement Kinetics Using Toehold Exchange. J. Am. Chem. Soc. 2009, 131 (47), 17303–17314.
14. He, G.; Li, J.; Ci, H.; Qi, C.; Guo, X. Direct Measurement of Single-Molecule DNA Hybridization Dynamics with Single-Base Resolution. Angew. Chemie - Int. Ed. 2016, 55 (31), 9036–9040.
15. Olson, X.; Kotani, S.; Yurke, B.; Graugnard, E.; Hughes, W. L. Kinetics of DNA Strand Displacement Systems with Locked Nucleic Acids. J. Phys. Chem. B 2017, 121 (12), 2594–2602.
16. Ang, Y. S.; Yung, L. Y. L. Dynamically Elongated Associative Toehold for Tuning DNA Circuit Kinetics and Thermodynamics. Nucleic Acids Res. 2021, 49 (8), 4258–4265.
17. Mayer, T.; Oesinghaus, L.; Simmel, F. C. Toehold-Mediated Strand Displacement in Random Sequence Pools. J. Am. Chem. Soc. 2023, 145 (1), 634–644.
18. Zadegan, R. M.; Jepsen, M. D. E.; Hildebrandt, L. L.; Birkedal, V.; Kjems, J. Construction of a Fuzzy and Boolean Logic Gates Based on DNA. Small 2015, 11 (15), 1811–1817.
19. Dong, J.; Wang, M.; Zhou, Y.; Zhou, C.; Wang, Q. DNA-Based Adaptive Plasmonic Logic Gates. Angew. Chemie - Int. Ed. 2020, 59 (35), 15038–15042.
20. Yang, J.; Wu, R.; Li, Y.; Wang, Z.; Pan, L.; Zhang, Q.; Lu, Z.; Zhang, C. Entropy-Driven DNA Logic Circuits Regulated by DNAzyme. Nucleic Acids Res. 2018, 46 (16), 8532–8541.
21. Liao, Y.; Hu, H.; Tang, X.; Qin, Y.; Zhang, W.; Dong, K.; Yan, B.; Mu, Y.; Li, L.; Ming, Z.; Xiao, X. A Versatile and Convenient Tool for Regulation of DNA Strand Displacement and Post-Modification on Pre-Fabricated DNA Nanodevices. Nucleic Acids Res. 2023, 51 (1), 29–40.
22. He, S.; Cui, R.; Zhang, Y.; Yang, Y.; Xu, Z.; Wang, S.; Dang, P.; Dang, K.; Ye, Q.; Liu, Y. Design and Realization of Triple DsDNA Nanocomputing Circuits in Microfluidic Chips. ACS Appl. Mater. Interfaces 2022, 14 (8), 10721–10728..
23. Frezza, B. M.; Cockroft, S. L.; Ghadiri, M. R. Modular Multi-Level Circuits from Immobilized DNA-Based Logic Gates. J. Am. Chem. Soc. 2007, 129 (48), 14875–14879.
24. Luo, T.; Fan, S.; Liu, Y.; Song, J. Information Processing Based on DNA Toehold-Mediated Strand Displacement (TMSD) Reaction. Nanoscale 2021, 13 (4), 2100–2112.
25. Seelig, G.; Soloveichik, D.; Zhang, D. Y.; Winfree, E. Enzyme-Free Nucleic Acid Logic Circuits. Science 2006, 314 (5805), 1585–1588.

26. Li, W.; Zhang, F.; Yan, H.; Liu, Y. DNA Based Arithmetic Function: A Half Adder Based on DNA Strand Displacement. Nanoscale 2016, 8 (6), 3775–3784.
27. Qian, L.; Winfree, E. Scaling up Digital Circuit Computation with DNA Strand Displacement Cascades. Science 2011, 332 (6034), 1196–1201.
28. Yurke, B.; Turberfield, A. J.; Mills, A. P.; Simmel, F. C.; Neumann, J. L. A DNA-Fuelled Molecular Machine Made of DNA. Nature 2000, 406 (6796), 605–608.
29. Srinivas, N.; Ouldridge, T. E.; Šulc, P.; Schaeffer, J. M.; Yurke, B.; Louis, A. A.; Doye, J. P. K.; Winfree, E. On the Biophysics and Kinetics of Toehold-Mediated DNA Strand Displacement. Nucleic Acids Res. 2013, 41 (22), 10641–10658.
30. Pan, L.; Wang, Z.; Li, Y.; Xu, F.; Zhang, Q.; Zhang, C. Nicking Enzyme-Controlled Toehold Regulation for DNA Logic Circuits. Nanoscale 2017, 9 (46), 18223–18228.
31. (Koc, K. N.; Stodola, J. L.; Burgers, P. M.; Galletto, R. Regulation of Yeast DNA Polymerase δ-Mediated Strand Displacement Synthesis by 5'-Flaps. Nucleic Acids Res. 2015, 43 (8), 4179–4190.
32. Song, T.; Eshra, A.; Shah, S.; Bui, H.; Fu, D.; Yang, M.; Mokhtar, R.; Reif, J. Fast and Compact DNA Logic Circuits Based on Single-Stranded Gates Using Strand-Displacing Polymerase. Nat. Nanotechnol. 2019, 14 (11), 1075–1081.
33. Wang, F.; Lv, H.; Li, Q.; Li, J.; Zhang, X.; Shi, J.; Wang, L.; Fan, C. Implementing Digital Computing with DNA-Based Switching Circuits. Nat. Commun. 2020, 11 (1), 121.
34. Xiong, X.; Zhu, T.; Zhu, Y.; Cao, M.; Xiao, J.; Li, L.; Wang, F.; Fan, C.; Pei, H. Molecular Convolutional Neural Networks with DNA Regulatory Circuits. Nat. Mach. Intell. 2022, 4 (7), 625–635.
35. Han, W.; Zhou, C. 8-Bit Adder and Subtractor with Domain Label Based on DNA Strand Displacement. Molecules 2018, 23 (11).
36. HOLDSWORTH, B.; WOODS, R. C. Programmable Logic Devices. In Digital Logic Design; HOLDSWORTH, B., WOODS, Eds.; Elsevier: Oxford, 2002; pp 326–366.
37. Harris, S. L.; Harris, D. Digital Building Blocks. In Digital Design and Computer Architecture; Harris, S. L., Harris, D., Eds.; Elsevier, 2022; pp 236–297.
38. Manno, M. M.; Ciletti, M. D. Sequential Programmable Devices. In Digital design; Gilfillan, A., Disanno, S., Eds.; 2016; pp 329–346.
39. H.Roth, C.; L.Kinney, L. Complex Programmable Logic Devices. In Fundamentals of Logic Design; Gowans, H., Dahl, F., Eds.; 2010; pp 268–269.
40. Li, W.; Pan, D. Z. A New Paradigm for FPGA Placement Without Explicit Packing. IEEE Trans. Comput. Des. Integr. Circuits Syst. 2019, 38 (11), 2113–2126.
41. Liu, L.; Hu, Q.; Zhang, W.; Li, W.; Zhang, W.; Ming, Z.; Li, L.; Chen, N.; Wang, H.; Xiao, X. Multifunctional Clip Strand for the Regulation of DNA Strand Displacement and Construction of Complex DNA Nanodevices. ACS Nano 2021, 15 (7), 11573–11584.
42. Goel, A.; Ibrahimi, M. A Renewable, Modular, and Time-Responsive DNA Circuit. Nat. Comput. 2011, 10 (1), 467–485.

43. Li, X.; Sun, X.; Zhou, J.; Yao, D.; Xiao, S.; Zhou, X.; Wei, B.; Li, C.; Liang, H. Enzyme-Assisted Waste-to-Reactant Transformation to Engineer Renewable DNA Circuits. Chem. Commun. 2019, 55 (77), 11615–11618.
44. O'Steen, M. R.; Cornett, E. M.; Kolpashchikov, D. M. Nuclease-Containing Media for Resettable Operation of DNA Logic Gates. Chem. Commun. 2015, 51 (8), 1429–1431.
45. Song, X.; Eshra, A.; Dwyer, C.; Reif, J. Renewable DNA Seesaw Logic Circuits Enabled by Photoregulation of Toehold-Mediated Strand Displacement. RSC Adv. 2017, 7 (45), 28130–28144.
46. Garg, S.; Shah, S.; Bui, H.; Song, T.; Mokhtar, R.; Reif, J. Renewable Time-Responsive DNA Circuits. Small 2018, 14 (33), 1–9.
47. Genot, A. J.; Bath, J.; Turberfield, A. J. Reversible Logic Circuits Made of DNA. J. Am. Chem. Soc. 2011, 133 (50), 20080–20083.
48. Zhao, Y.; Liu, Y.; Zheng, X.; Wang, B.; Lv, H.; Zhou, S.; Zhang, Q.; Wei, X. Half Adder and Half Subtractor Logic Gates Based on Nicking Enzymes. Mol. Syst. Des. Eng. 2019, 4 (6), 1103–1113.
49. Elbaz, J.; Lioubashevski, O.; Wang, F.; Remacle, F.; Levine, R. D.; Willner, I. DNA Computing Circuits Using Libraries of DNAzyme Subunits. Nat. Nanotechnol. 2010, 5 (6), 417–422.
50. Deng, F.; Pan, J.; Liu, Z.; Zeng, L.; Chen, J. Programmable DNA Biocomputing Circuits for Rapid and Intelligent Screening of SARS-CoV-2 Variants. Biosens. Bioelectron. 2023, 223 (September 2022), 115025.

Figures

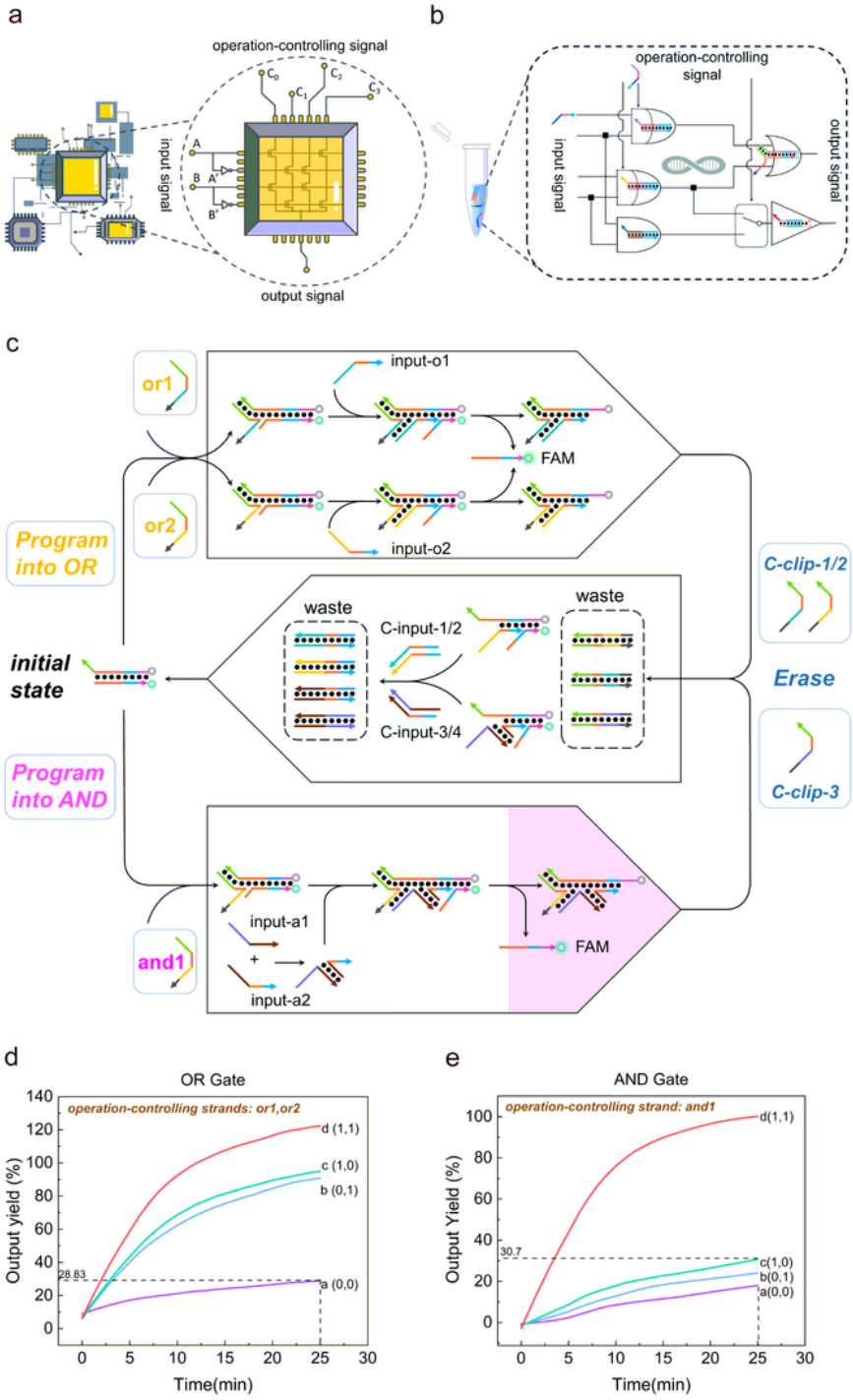
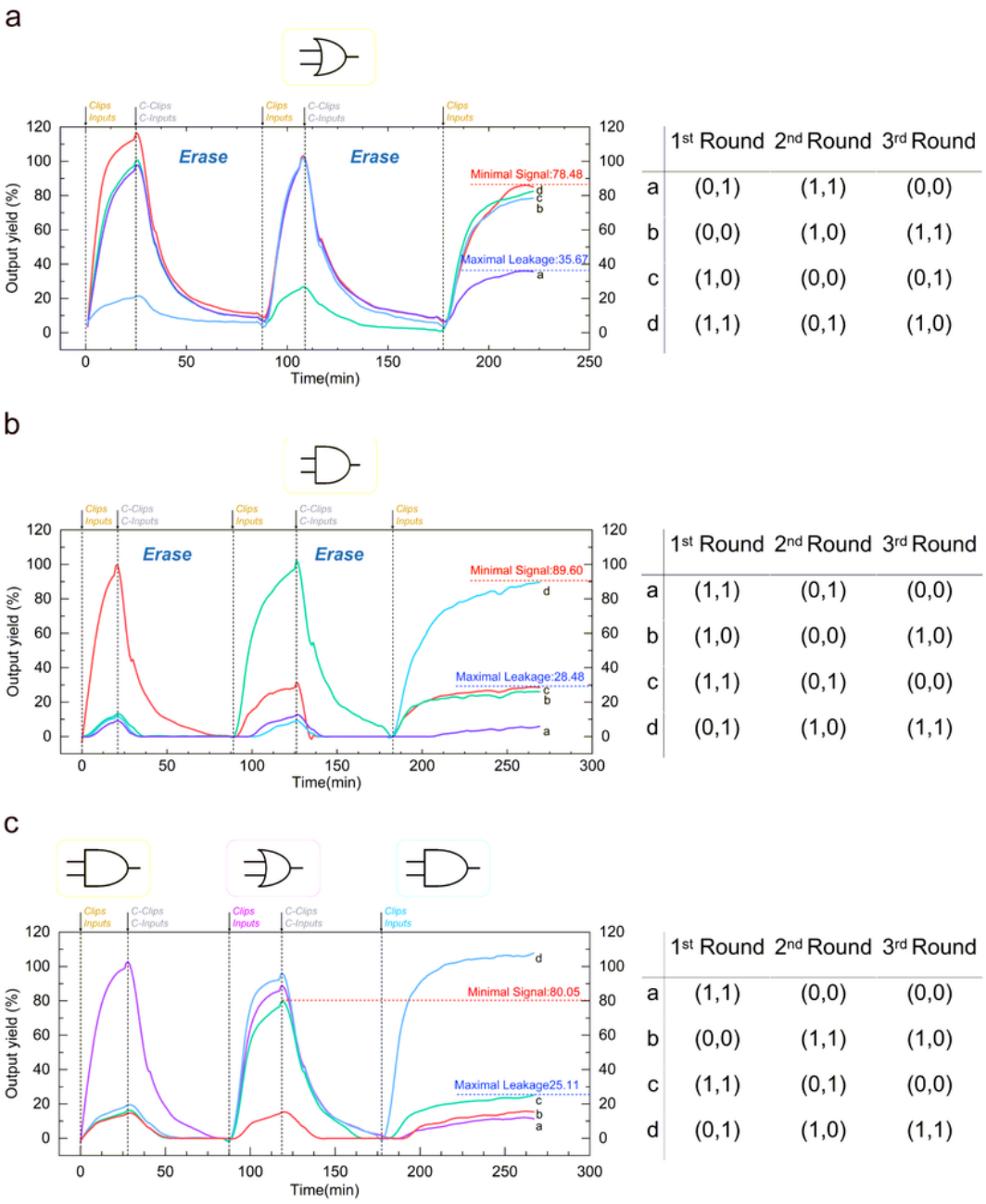


Figure 1

(a) Schematic illustration of the Configurable Logic Block (CLB) in digital circuits. (b) Schematic illustration of CLB-based field-programmable DNA circuits. The clip strands were used as the operation-controlling signals so that adding different clips could result in different logic operations. Also, the clip strands enabled the circuit to be erased by adding their complementary strands. (c) Schematic illustration of CLB-based field-programmable OR and AND gate. Adding different clips (operation-controlling strands)

would lead to corresponding logic operations, and the initial state could be restored by adding C-clips and C-inputs. (d) Experimental verification of OR gate. (e) Experimental verification of OR gate. Reaction setup: 100-nM FAM:BHQ,120-nM clips, 240-nM inputs were added sequentially.



	1st Round	2nd Round	3rd Round
a	(0,1)	(1,1)	(0,0)
b	(0,0)	(1,0)	(1,1)
c	(1,0)	(0,0)	(0,1)
d	(1,1)	(0,1)	(1,0)

	1st Round	2nd Round	3rd Round
a	(1,1)	(0,1)	(0,0)
b	(1,0)	(0,0)	(1,0)
c	(1,1)	(0,1)	(0,0)
d	(0,1)	(1,0)	(1,1)

	1st Round	2nd Round	3rd Round
a	(1,1)	(0,0)	(0,0)
b	(0,0)	(1,1)	(1,0)
c	(1,1)	(0,1)	(0,0)
d	(0,1)	(1,0)	(1,1)

Figure 2

Experimental verification of the field programmability of the CLB-based OR and AND gate. (a) The fluorescent curves of OR gate reusing for 3 times. (b) The fluorescent curves of AND gate reusing for 3 times. (c) The fluorescent curves of the logic gate switched in the order of AND-OR-AND. Reactions setup: 100-nM FAM:BHQ, 120-nM operation-controlling strands, 240-nM inputs for the first-time using; 120-nM C-clips, 240-nM C-inputs for the first-time restoring. The concentration of inputs/clips and C-input/clips would increase 10 nM each round, which means 130-nM clips/C-clips and 250-nM inputs/C-inputs for the second round, and 140-nM clips/C-clips and 260-nM inputs/C-inputs for the third round.

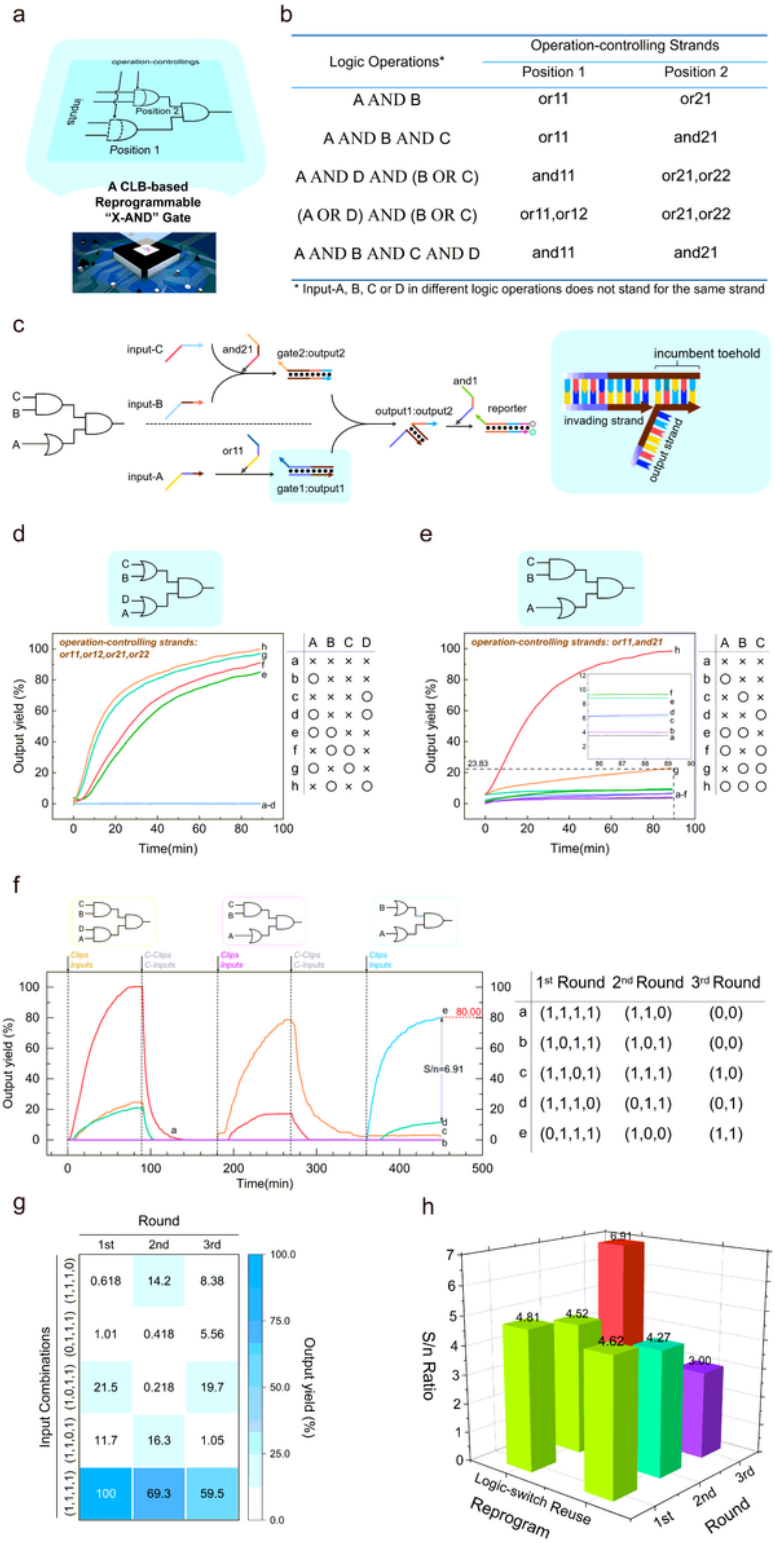
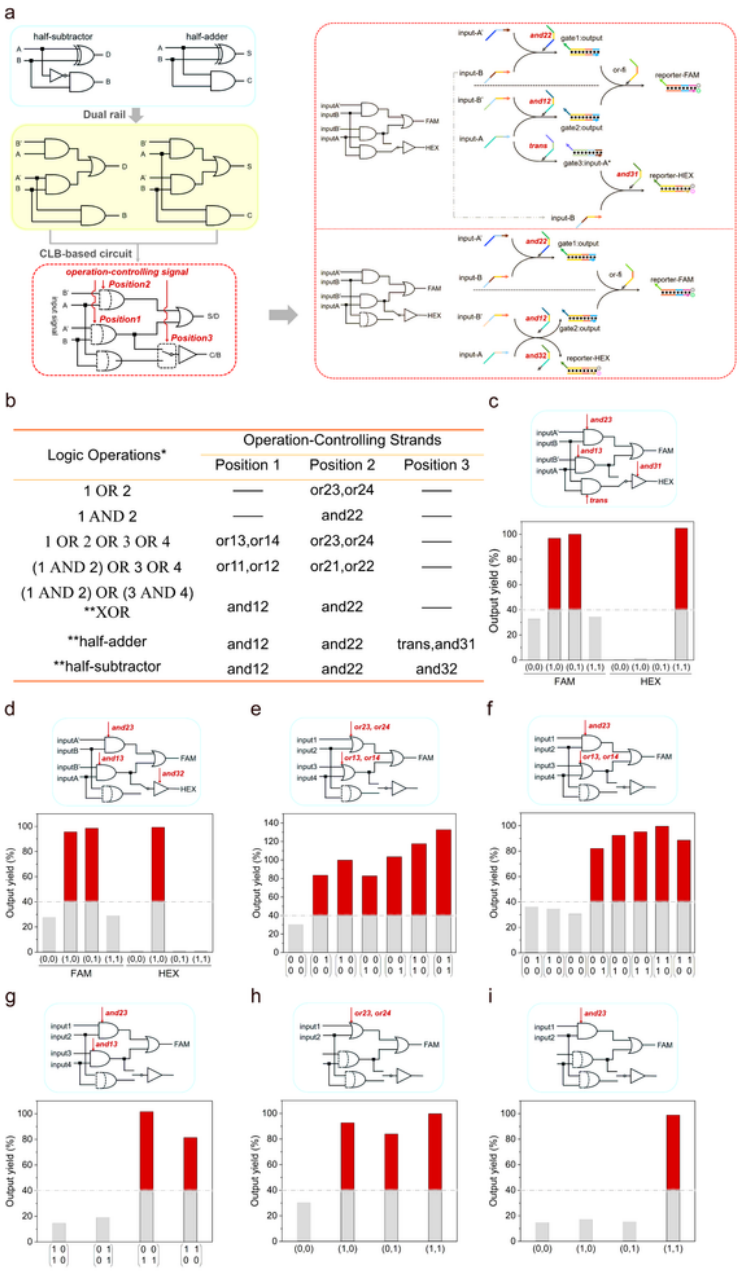


Figure 3

(a-b) Schematic illustration (a) and logic operation table (b) of the CLB-based “X-AND” gate. There are 2 positions for users to add operation-controlling strands to realize 5 different logic operations. (c) Digital circuit diagram of “A AND B AND C” and the schematic illustration of its corresponding CLB-based DNA circuit. The pattern inside the blue box shows the location of the “incumbent toehold”. (d-e) The fluorescent curves of “(A OR D) AND (B OR C)”(d) and “A AND B AND C”(e). Reactions setup: 100-nM FAM:BHQ, 240-nM gate2:output2 and gate1:output1, 120-nM and1, 240-nM operation controlling strands and inputs were added sequentially. (f) The fluorescent curves of “X-AND” gate switched in the order of (A AND D AND B AND C)-(A AND B AND C)-(A AND B). (g) The heat map reflecting the output yield of “A AND B AND C AND D” in a 3-round reusing. (h) The comparison of signal-to-noise ratio between logic-switching and reusing “A AND B AND C AND D”. Reactions setup: 100-nM FAM:BHQ, 240-nM gate2:output2 and gate1:output1, 120-nM and1, 240-nM operation controlling strands and inputs for the first-time using, 120-nM c-and1, 240-nM c-clips/inputs for the first-time restoring. The concentration of inputs/clips and C-input/clips would increase 10 nM each round.



Logic Operations*	Operation-Controlling Strands		
	Position 1	Position 2	Position 3
1 OR 2	——	or23,or24	——
1 AND 2	——	and22	——
1 OR 2 OR 3 OR 4	or13,or14	or23,or24	——
(1 AND 2) OR 3 OR 4	or11,or12	or21,or22	——
(1 AND 2) OR (3 AND 4) **XOR	and12	and22	——
**half-adder	and12	and22	trans,and31
**half-subtractor	and12	and22	and32

Figure 4. (a) Schematic illustration of using CLB paradigm to integrate a half-adder and a half subtractor into one dual-rail comprehensive circuit. Inside the dashed box illustrates the digital circuit diagram of half-adder/subtractor and their corresponding CLB-based DNA circuit. (b) Logic operation table of the integrated comprehensive circuit. * Input-1, 2, 3 or 4 in different logic operations does not stand for the same strand.** The corresponding logic operations were dual-rail. (c-i) The output yield of half-adder (c), half-subtractor (d), "1 OR 2 OR 3 OR 4" (e), "(1 AND 2) OR 3 OR 4" (f), XOR (g), "1 AND 2" (h), "1 OR 2" (i). Note that the matrix were arranged in the order of $\begin{pmatrix} \text{input-1} & \text{input-2} \\ \text{input-3} & \text{input-4} \end{pmatrix}$

Reactions setup: 100-nM FAM:BHQ and HEX:HBHQ, 240-nM gate1:output, gate2:output and gate3:ouputA*, 120-nM or-fi, 240-nM operation-controlling strands and 480-nM input strands were added sequentially.

Figure 4

See image above for figure legend.

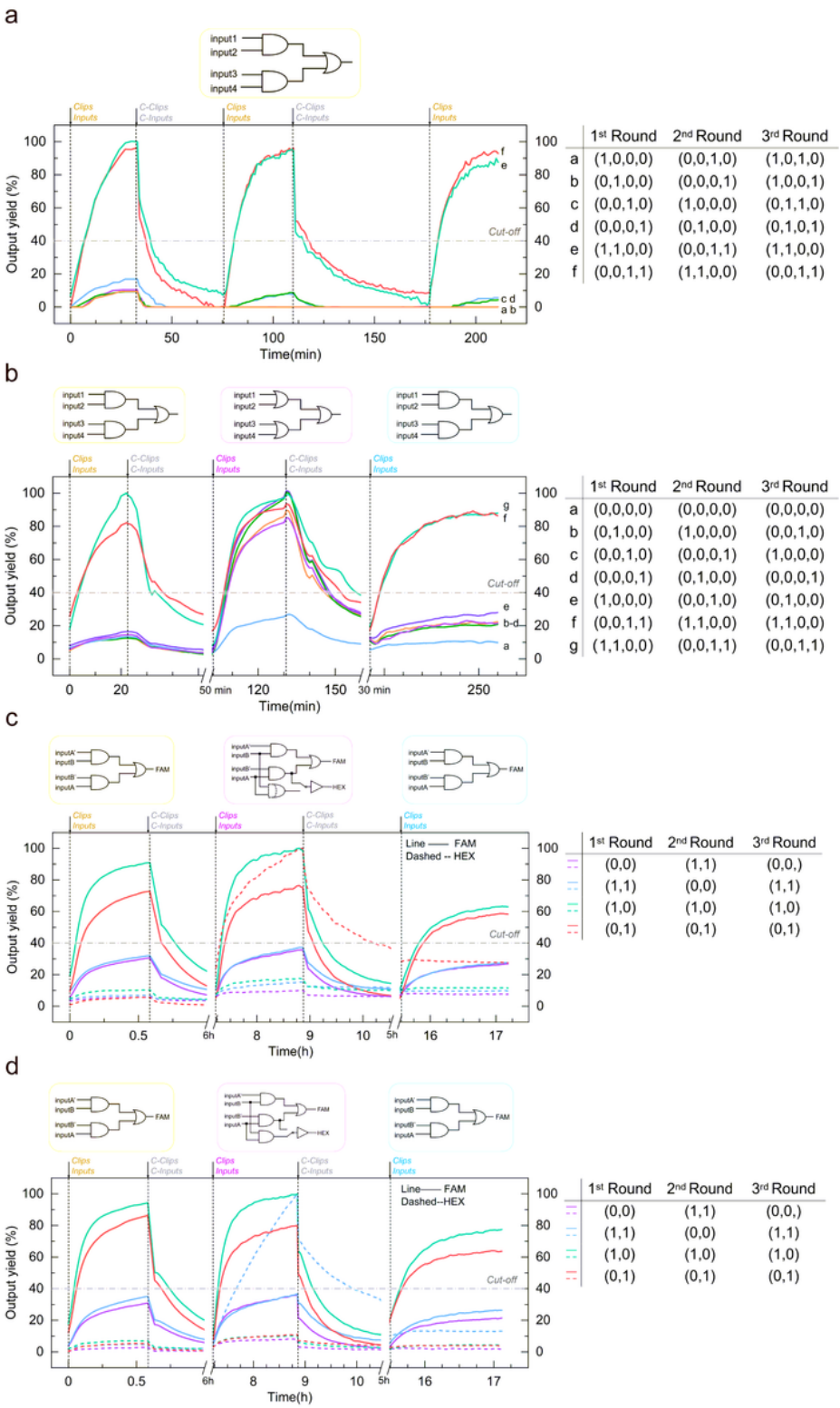


Figure 5

(a) Fluorescent curves of XOR gate reusing for 3 times. (b) Fluorescent curves of the comprehensive logic gate switched in the order of XOR-(1 OR 2 OR 3 OR 4). (c) Fluorescent curves of the comprehensive logic gate switched in the order of XOR-half adder-XOR. (d) Fluorescent curves of the comprehensive logic gate switched in the order of XOR-half subtractor-XOR. Reactions setup: 100-nM FAM:BHQ and HEX:HBHQ, 240-nM gate1:output, gate2:output and gate3:ouputA*, 120-nM or-fi, 240-nM operation-controlling strands

and 480-nM input strands were added for the first-time using; 120-nM c-or-fi, 240-nM C-clips and 480-nM C-inputs were added for the first-time restoring. The concentration of inputs/clips and C-input/clips would increase 10 nM each round.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- CLBsupinfo.docx