Enzyme-free nucleic acid dynamical systems

Supplementary Materials

Niranjan Srinivas,[1*] James Parkin,[2] Georg Seelig,[3]
Erik Winfree,[1,2,4] and David Soloveichik[5*]

[1]Computation and Neural Systems, California Institute of Technology, Pasadena, CA 91125,
[2]Bioengineering, California Institute of Technology, Pasadena, CA 91125,
[3]Electrical Engineering, University of Washington, Seattle, WA 98195,
[4]Computer Science, California Institute of Technology, Pasadena, CA 91125,
[5]Electrical and Computer Engineering, UT Austin, 1616 Guadalupe St., Austin, TX 78701.
*To whom correspondence should be addressed;
E-mail: niranjan@dna.caltech.edu, david.soloveichik@utexas.edu

Contents

S1 CRN-to-DNA implementation scheme: Additional details

S1.1 History domain

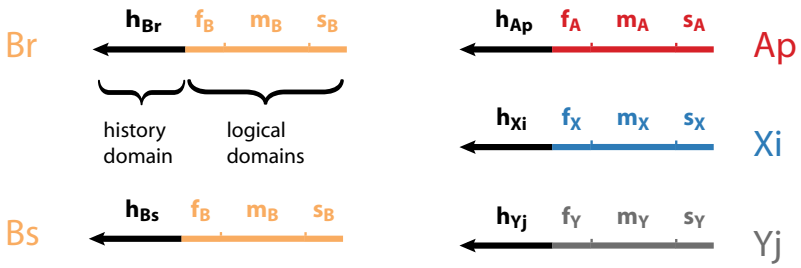

Figure S1: Each signal strand comprises a history domain in black (e.g. $\mathrm{h_{Br}}$) and a logical unit that comprises three domains: the first toehold (e.g. $\mathrm{f_B}$), a branch migration region (e.g. $\mathrm{m_B}$), and the second toehold (e.g. $\mathrm{s_B}$). The logical unit is common to all signal strands that represent a particular formal species; the history domain is specific to a particular position on the Produce complex. The domains in the logical unit participate in all desired strand displacement reactions (e.g. Fig. S5). The history domain facilitates correct annealing of the Produce complex. Signal strands with the same logical unit (e.g. Br and Bs) represent the same formal species (B) and are designed to behave identically in solution.

S1.2 Naming scheme

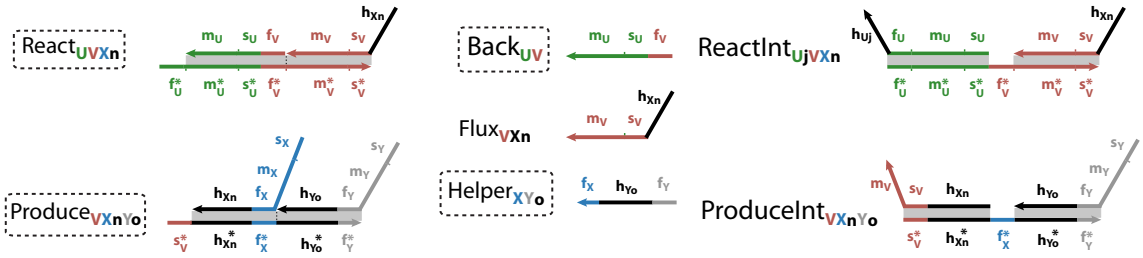

Figure S2: Our naming scheme is illustrated with the molecules used in the general reaction $\mathrm{U + V \rightarrow X + Y}$ as an example. The scheme is both precise and general – the name and the molecule fully determine each other. Note that each molecule only captures the reactivity and logical function of that molecule: e.g. neither the name $\mathrm{React_{UVXn}}$ nor the corresponding molecule encode any information about the second output released downstream during the produce step. Similarly, $\mathrm{Flux_{VXn}}$ does not include any information about the species U that was consumed to release it. Also, each molecule is fully determined by its name: e.g. $\mathrm{ReactInt_{UjVXn}}$ includes the specific history domain $\mathrm{h_{Uj}}$, which keeps track of the particular signal strand Uj. The waste complexes (not shown) also have precise and general names.

S2 Molecular implementation of autocatalytic single-reaction CRNs

S2.1 Detailed specification of the desired reaction pathways

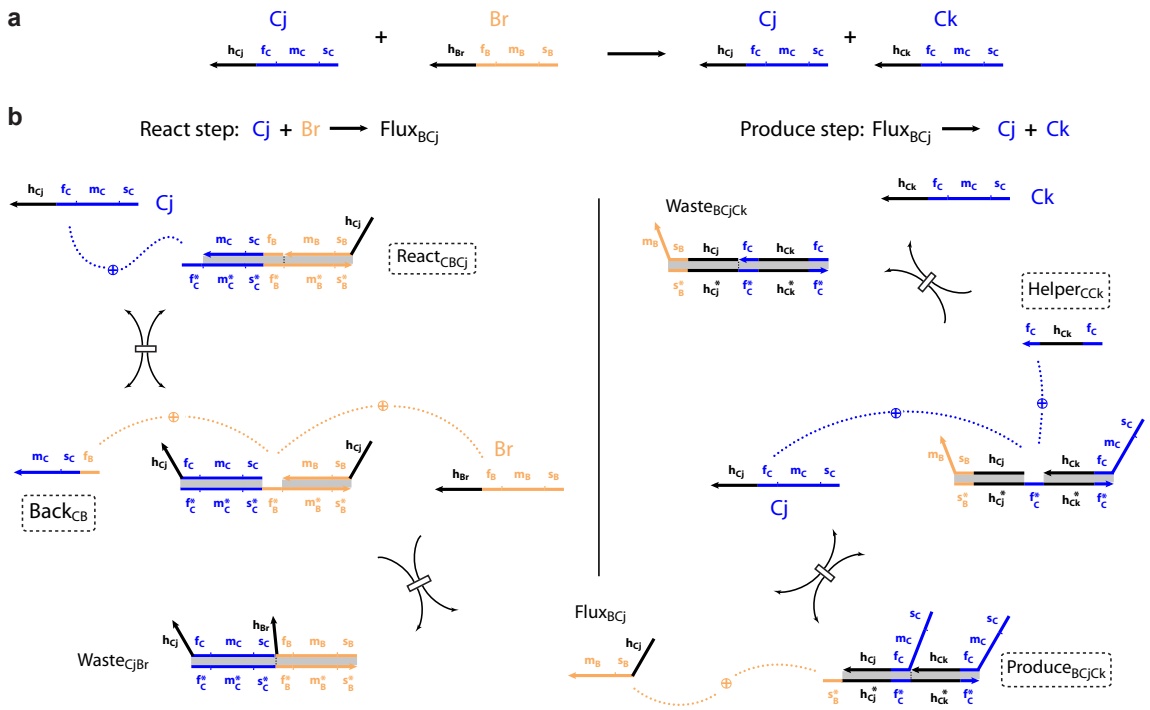

Figure S3: DNA implementation for $\mathrm{C} + \mathrm{B} \rightarrow 2\mathrm{C}$. Dashed box indicates fuel species.

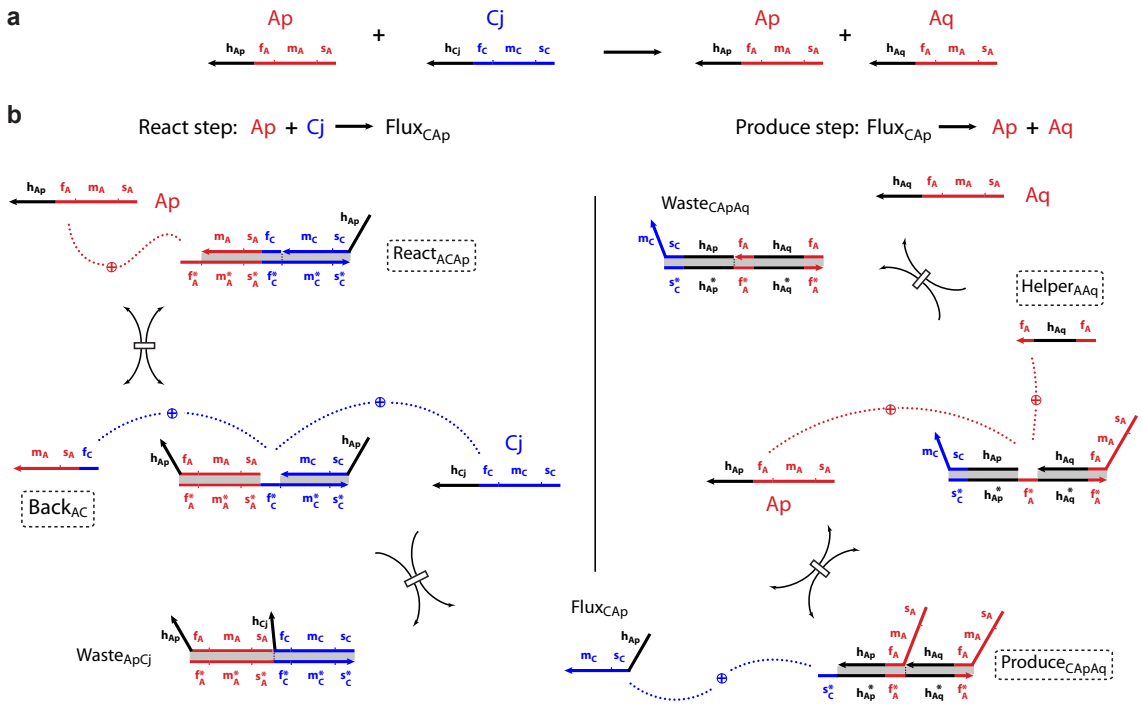

Figure S4: DNA implementation for A + C → 2A. Dashed box indicates fuel species.

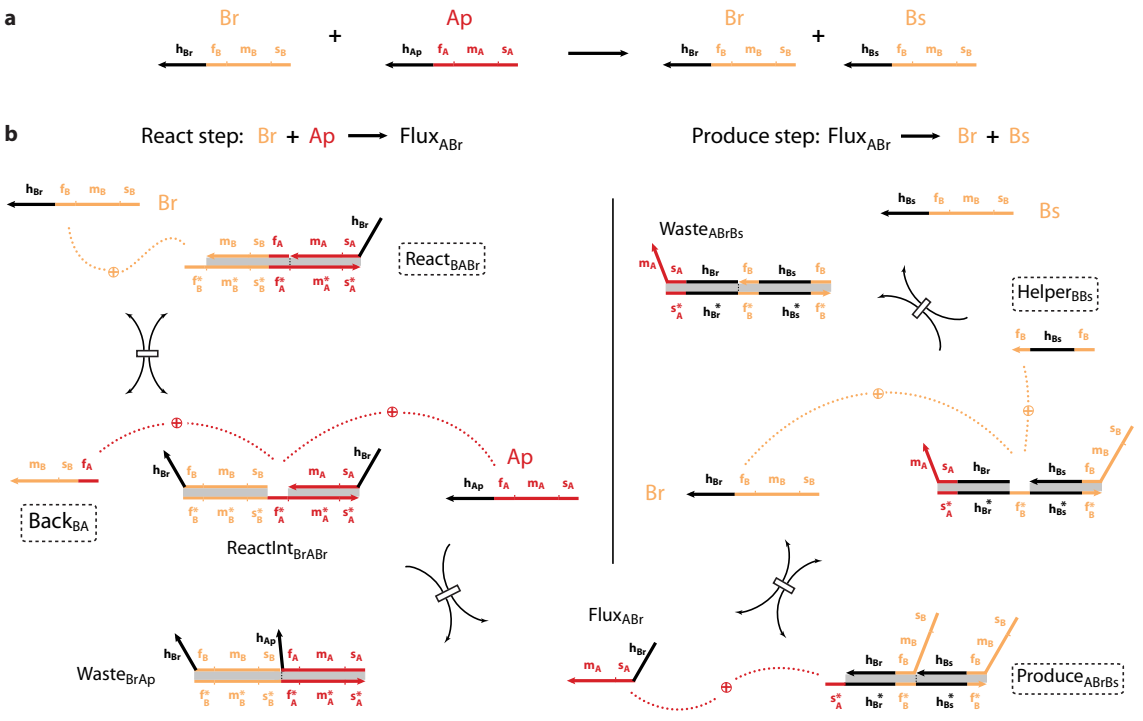

Figure S5: DNA implementation for B + A → 2B. Dashed box indicates fuel species.

S2.2 Proposed molecular mechanisms for leak reactions

These leak pathways were first encountered in the context of our efforts to engineer autocatalytic single-reaction CRNs, and are therefore presented in that context. However, these leak pathways are more general - they would be present in any DNA implementation constructed using the general scheme presented in this work.

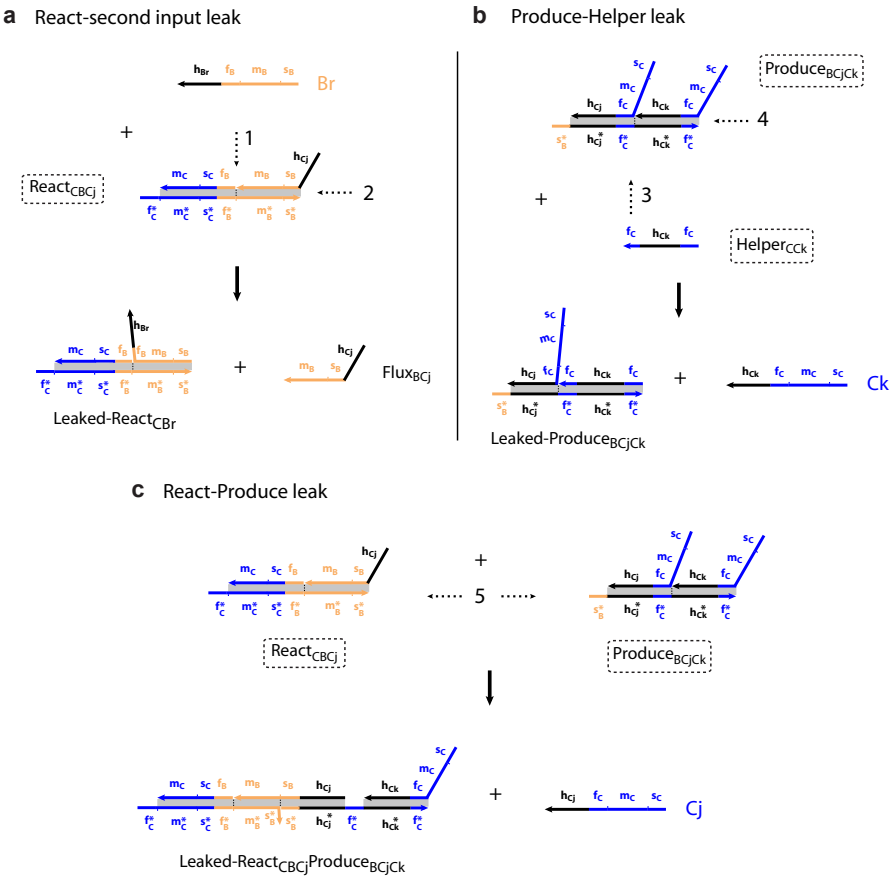

Figure S6: Examples of leak pathways. Numbered dashed arrows indicate locations of invasion. a. The second input (here, Br) can invade at locations 1 (the junction) and 2 (the end of the helix) in the React complex. Once strand displacement finishes, the Flux strand and a spurious complex are formed. b. The Helper strand and Produce complex can react similarly to release the second output (here, Ck) and a spurious complex. c. Spontaneous thermal fraying at the end of the helix in the React complex may enable the Produce complex to invade at location 5, resulting in the release of the first output (here, C_j) and a spurious complex. Spurious complexes may participate in downstream reactions through legitimate strand displacement pathways (Fig. S7).

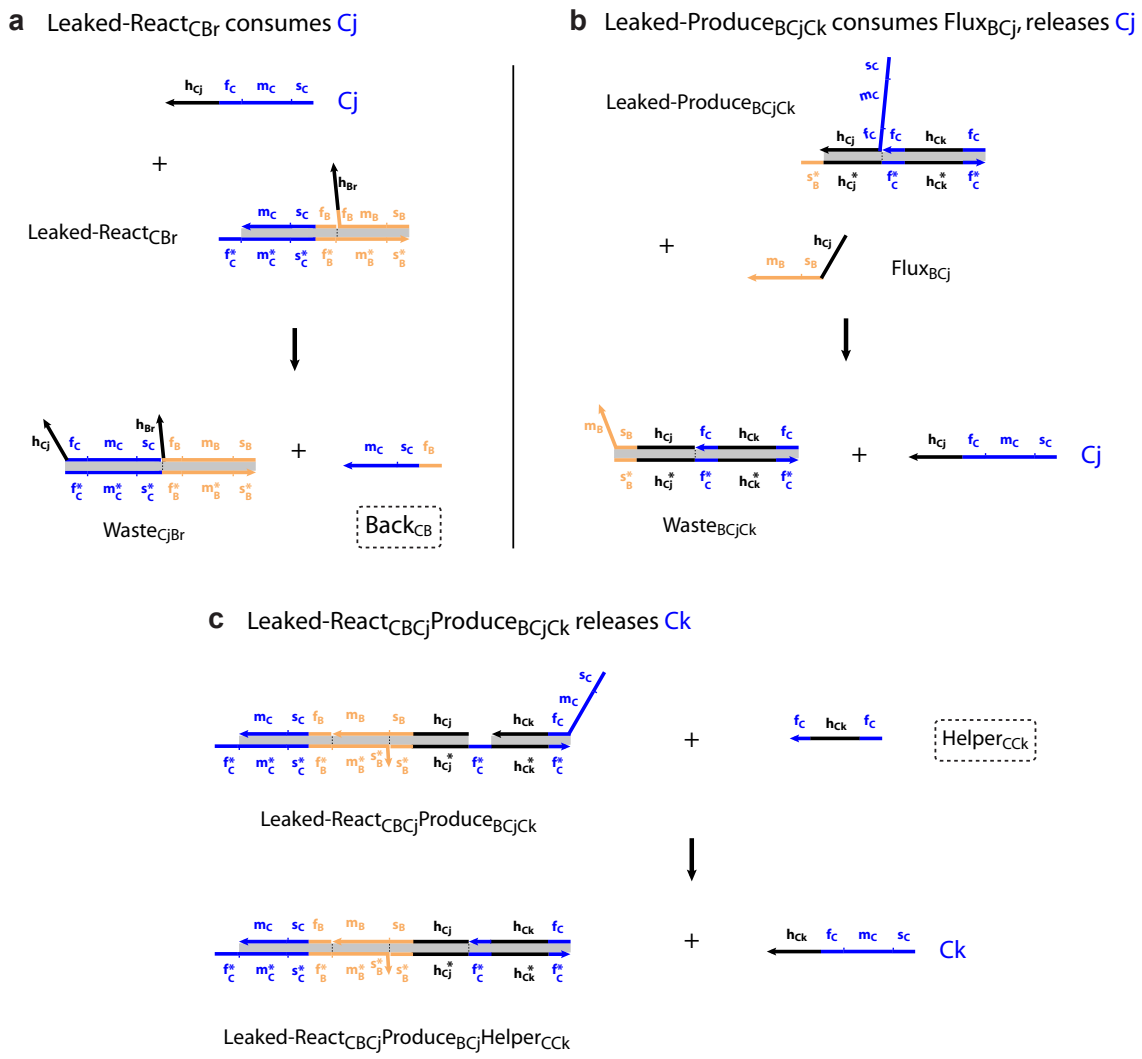

Figure S7: Spurious products that are formed due to leak pathways may undergo reactions that are legitimate steps in desired reaction pathways. Therefore leak reactions may affect dynamical behavior in ways that are more complex than merely the unexpected release of signal strands or Flux strands.

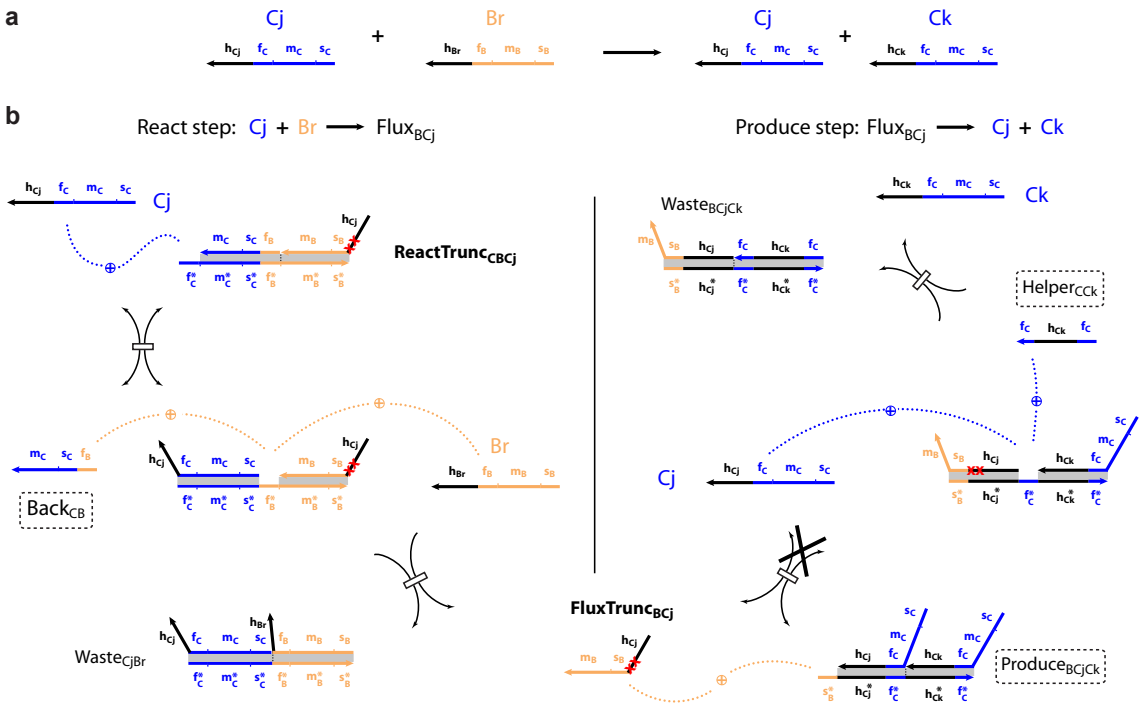

Figure S8: Synthesis errors, such as truncations, can adversely affect reaction stoichiometry in the DNA implementation. In particular, synthesis errors in toehold regions and the initial bases of branch migration domains can dramatically reduce the rates of desired strand displacement reactions. For example, truncations in the Flux strand (indicated by red crosses) can prevent efficient triggering of the produce step, thereby resulting in consumption of signal strands in the react step without the corresponding release of signal strands in the produce step. This leads to incorrect reaction stoichiometry.

S2.3 Toehold occlusion

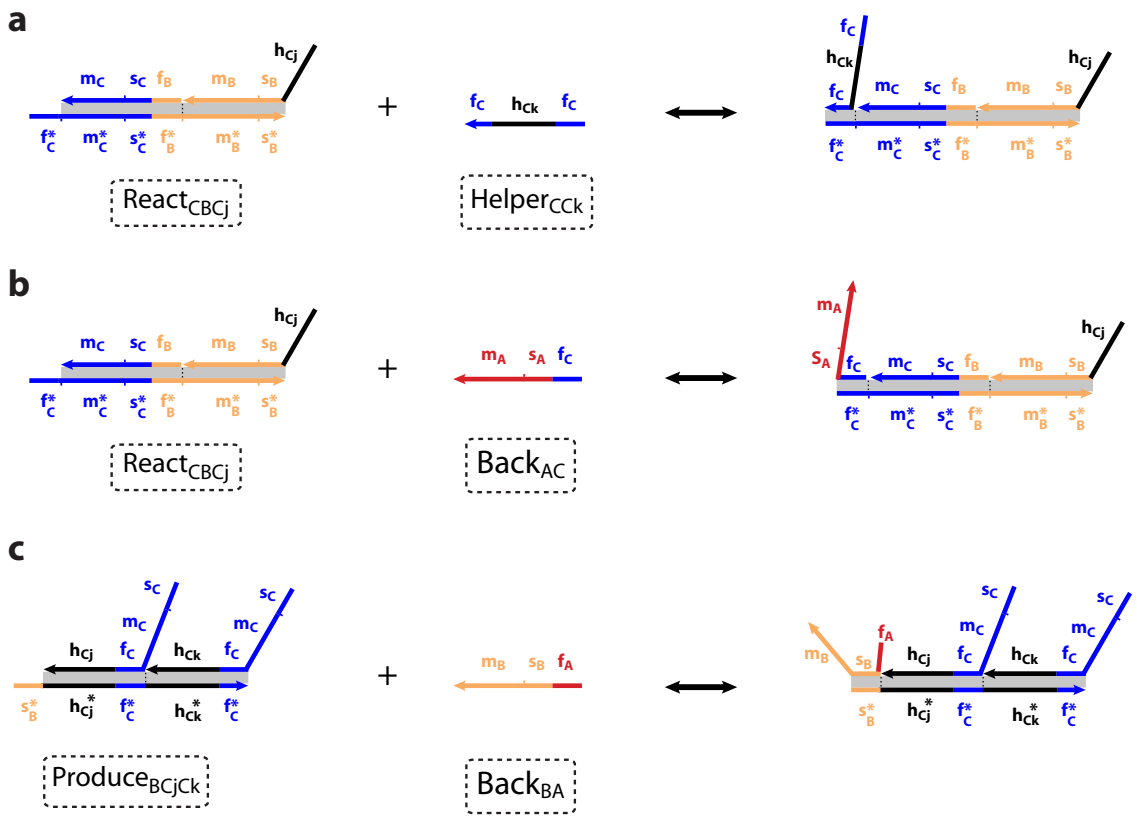

Figure S9: Illustrative (but not exhaustive) examples of toehold-only interactions in the molecular implementation of our oscillator. Note that all these interactions (but not all interactions) are between fuel species at high concentration. There are interactions within the same autocatalytic module (e.g. (a)) and across modules (e.g. (b), (c)).

S2.4 Counteracting damping: Catalytic helper mechanism

One way of counteracting damping and "tuning up" our Displacillator would be to engineer an alternative mechanism for the produce step where, in addition to the second output of the Produce complex, the Flux strand is also released. If that can be engineered, the Flux strand could interact with more Produce complexes and release more outputs, thereby effectively increasing output stoichiometry. Inspired by Zhang et.al. (1)'s entropy-driven catalyst, we introduced the "catalytic Helper" mechanism (see Fig. S10) to mediate such an alternative pathway. Note that in our case, the entropic benefit is a bonus and not the only reason our desired pathway is thermodynamically favorable.

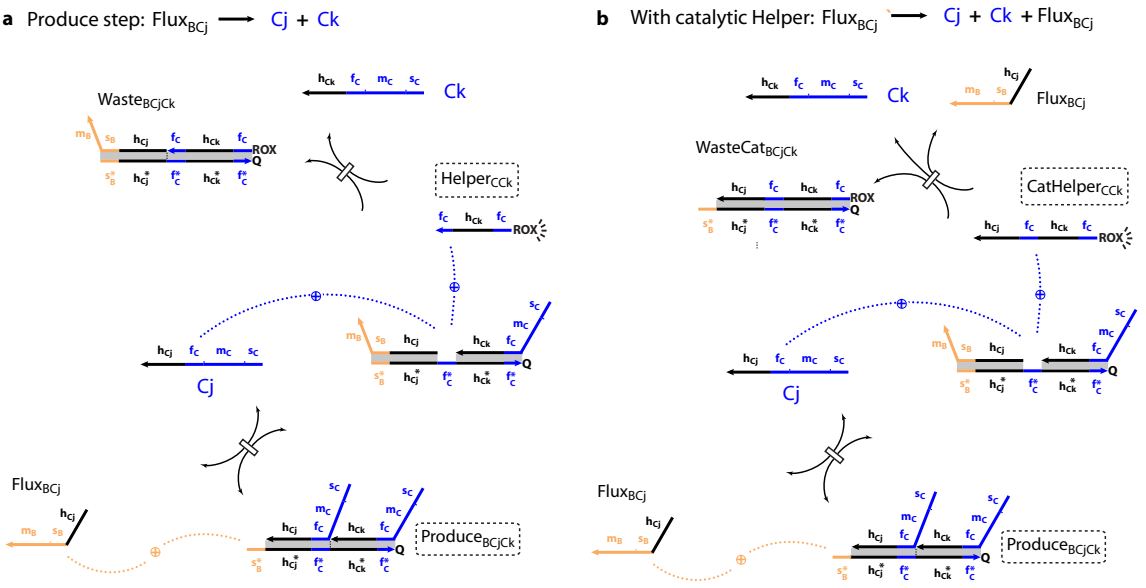

Figure S10: Catalytic Helper mechanism. a. Produce step for the reaction $\mathrm{C} + \mathrm{B} \rightarrow 2\mathrm{C}$ facilitated by the traditional Helper strand. b. Produce step for the reaction $\mathrm{C} + \mathrm{B} \rightarrow 2\mathrm{C}$ using the "catalytic Helper" strand. The CatHelper strand is simply the Helper strand extended at the 3' end with the history domain of the first output of the Produce complex (here, $\mathrm{h_{Cj}}$). Apart from releasing the second output (here, Ck) the catalytic Helper also displaces the Flux strand by toehold exchange, which is then free to interact with another Produce complex to release more outputs, thereby effectively "tuning up" the output stoichiometry of the desired CRN. Note that both the Helper and CatHelper strands are tagged with a fluorophore (here, ROX) at the 5' end. Since the Produce complex is tagged with a quencher at the 3' end of the bottom strand, the consumption of the Helper and CatHelper strands can be quantitatively measured through fluorimetry (Sec. S4.1).

S2.5 Comparing the kinetics of the autocatalytic modules

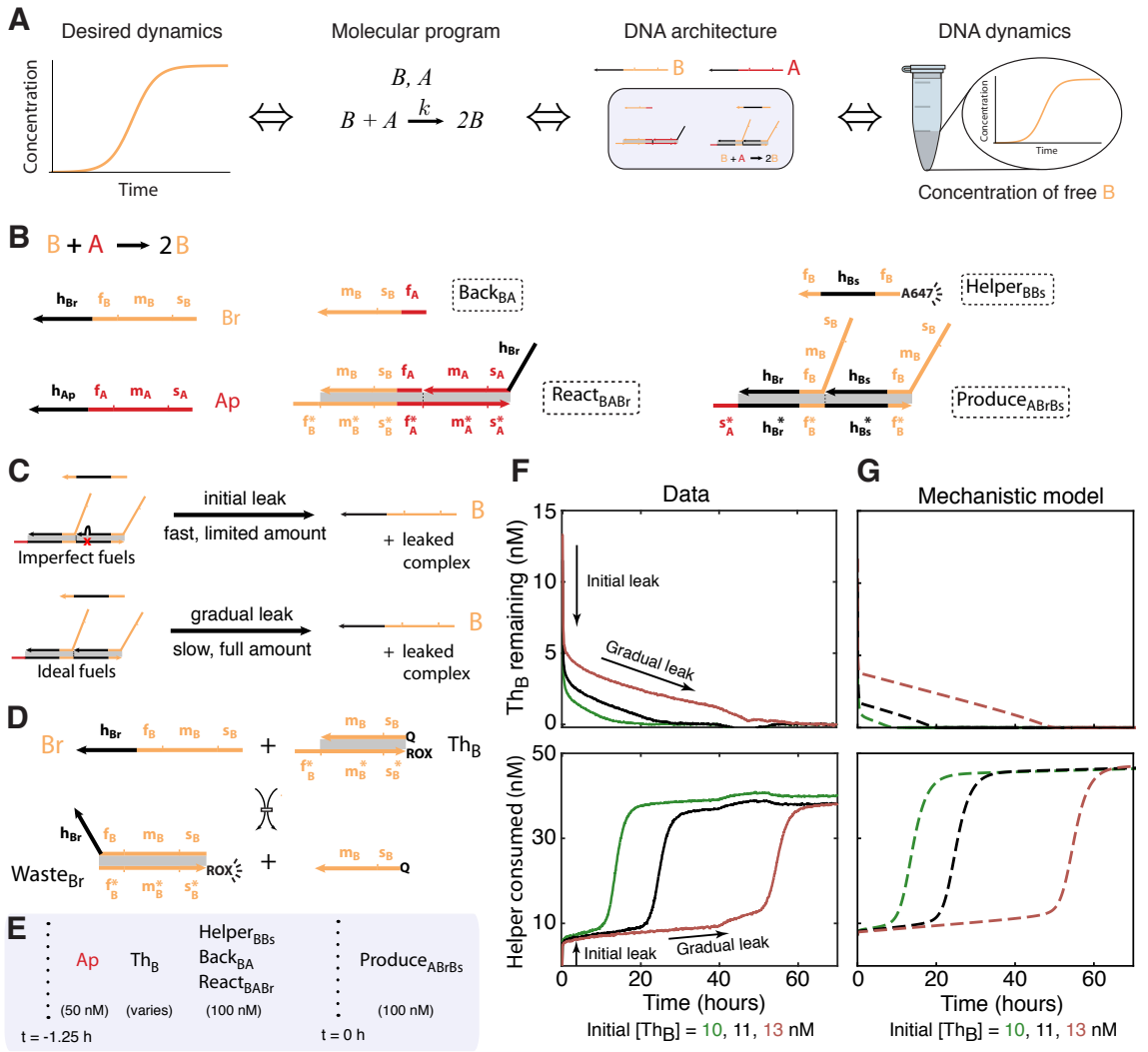

Figure S11: (**A**) Schematic for engineering a single-reaction CRN with exponential amplification using our systematic pipeline. (**B**) Domain-level illustration of the DNA species involved (fuel species indicated by dashed boxes). (**C**) A limited amount of imperfect fuel species, such as those with DNA synthesis errors, release signal strands and waste products through fast spurious pathways ("initial leak"). Ideal fuel species release similar products through slow "gradual" leak. (**D**) A Threshold complex ($\mathrm{Th_B}$) is designed to consume leaked autocatalyst. (**E**) Experimental setup. Vertical dotted lines separate initial contents of the test tube and timed additions. Addition of Produce complexes kickstarts release of autocatalyst through initial and gradual leak. (**F**) Experimental data showing concentration of $\mathrm{Th_B}$ (top) and the amount of $\mathrm{Helper_{BBs}}$ consumed (bottom) for three independent samples with differing initial amounts of $\mathrm{Th_B}$. (**G**) The essential features of the autocatalytic dynamics were captured by a quantitative mechanistic model at the level of individual strand displacement reactions (see S5). Besides independently measured parameters and empirical parameters that were fit to the full oscillator data, only the initial threshold amounts were fit to the autocatalytic data shown here. These additional parameters capture the uncertainty in pipetting and initial leak, and effectively control the "triggering" time of autocatalysis. Note that to minimize the number of fit parameters, a global substoichiometric yield parameter was used, which could not fully account for the final "Helper consumed" amount in each autocatalytic module.

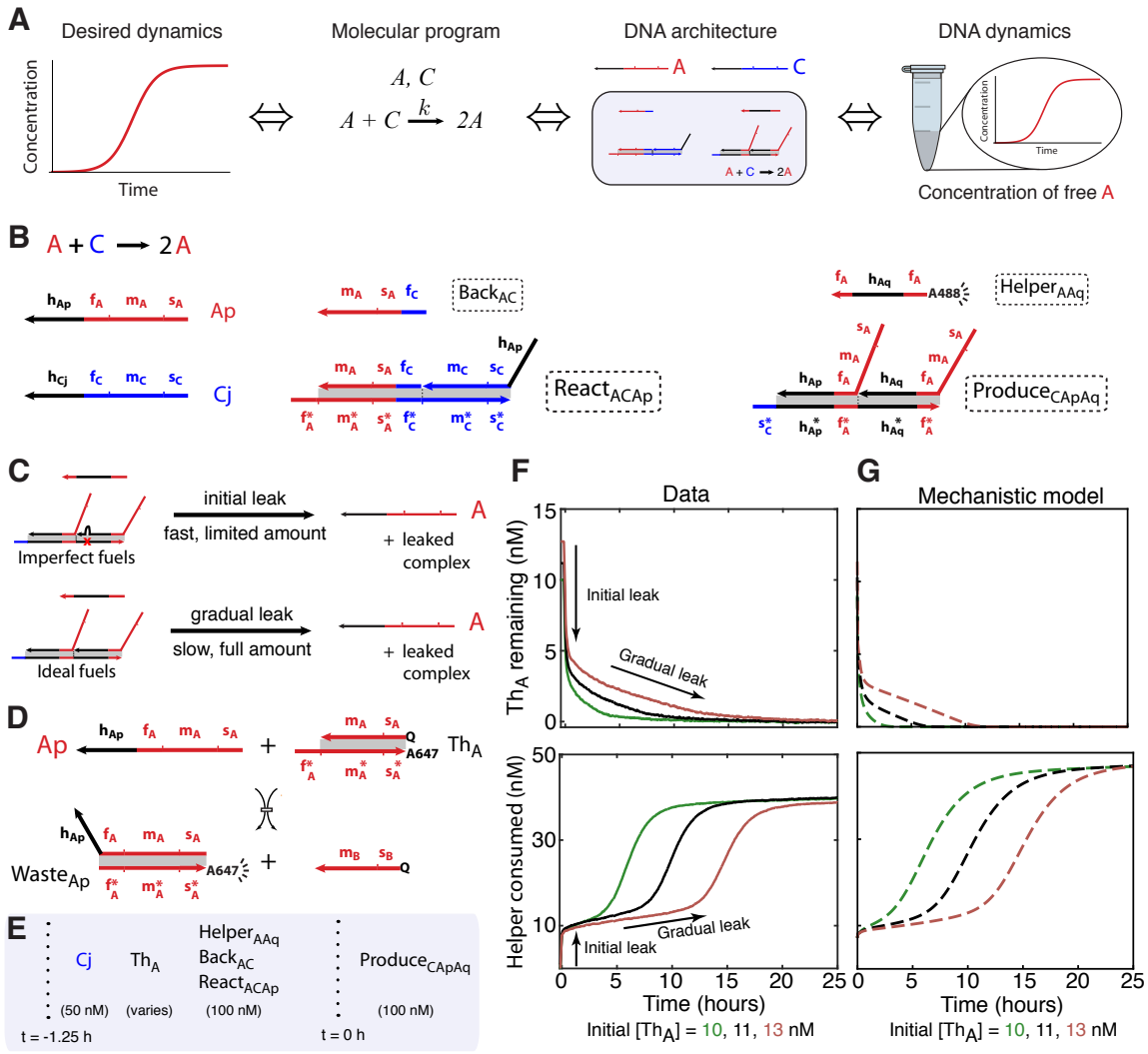

Figure S12: (**A**) Schematic for engineering a single-reaction CRN with exponential amplification using our systematic pipeline. (**B**) Domain-level illustration of the DNA species involved (fuel species indicated by dashed boxes). (**C**) A limited amount of imperfect fuel species, such as those with DNA synthesis errors, release signal strands and waste products through fast spurious pathways ("initial leak"). Ideal fuel species release similar products through slow "gradual" leak. (**D**) A Threshold complex ($\mathrm{Th_A}$) is designed to consume leaked autocatalyst. (**E**) Experimental setup. Vertical dotted lines separate initial contents of the test tube and timed additions. Addition of Produce complexes kickstarts release of autocatalyst through initial and gradual leak. (**F**) Experimental data showing concentration of $\mathrm{Th_A}$ (top) and the amount of $\mathrm{Helper_{AAq}}$ consumed (bottom) for three independent samples with differing initial amounts of $\mathrm{Th_A}$. (**G**) The essential features of the autocatalytic dynamics were captured by a quantitative mechanistic model at the level of individual strand displacement reactions (see S5). Besides independently measured parameters and empirical parameters that were fit to the full oscillator data, only the initial threshold amounts were fit to the autocatalytic data shown here. These additional parameters capture the uncertainty in pipetting and initial leak, and effectively control the "triggering" time of autocatalysis. Note that to minimize the number of fit parameters, a global substoichiometric yield parameter was used, which could not fully account for the final "Helper consumed" amount in each autocatalytic module.

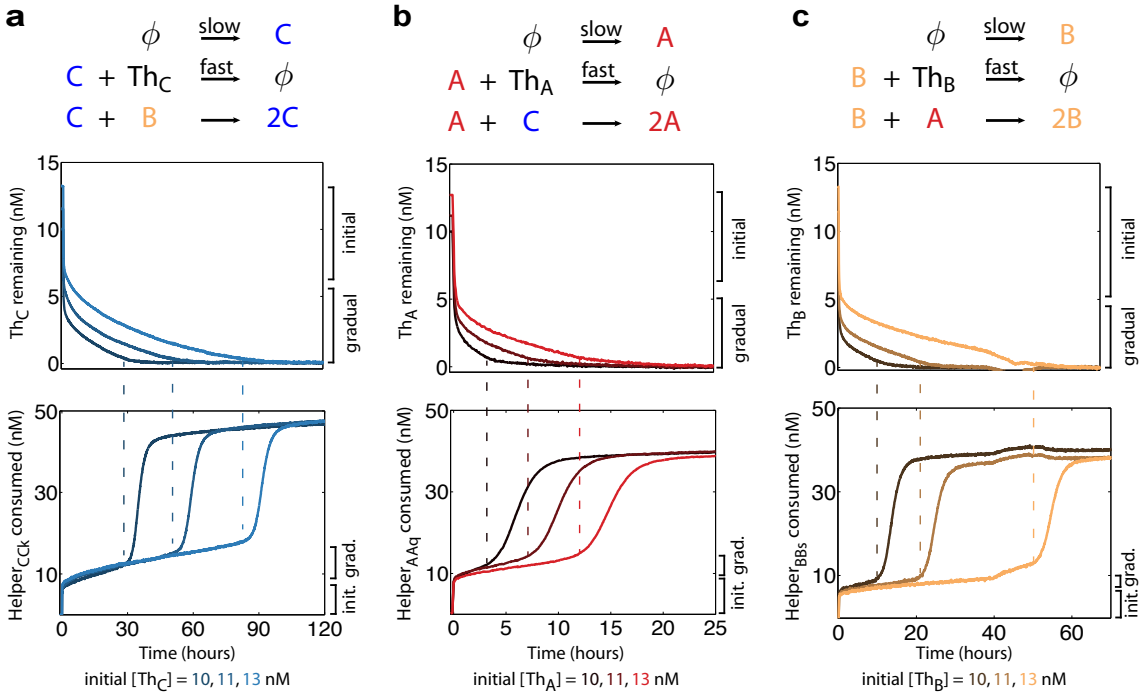

Figure S13: Exponential kinetics of all three autocatalytic modules with delays tuned by Threshold concentrations. Fig. S14 illustrates our method for estimating leak rates for each module; Table S3 summarizes these estimates. Note that the time-axes are different for the three modules. The data illustrates the substoichiometric yield effect (Fig. S8): in panel (b), for example, the total consumption of $\mathrm{Helper}_{\mathrm{AAq}}$, including that consumed by initial leak, is approximately 40 nM. This total value is 20% less than what we would expect, given that the total initial concentration of C was approximately 50 nM. The substoichiometric yield effect is probably even larger, since some of the initial consumption of $\mathrm{Helper}_{\mathrm{AAq}}$ would arise from initial leak with $\mathrm{Produce}_{\mathrm{CApAq}}$, which is independent of C (e.g. Fig. S6b). Substoichiometric yield is also observed in experiments with individual fuel complexes (e.g. Fig. S22).

Complex	k_{fwd1}	k_{back}	k_{fwd2}
$\mathrm{React}_{\mathrm{ACApi2}}$	2.7×10^5	1.1×10^6	1.4×10^5
$\mathrm{React}_{\mathrm{BABr}}$	1.8×10^5	6.2×10^5	2.7×10^5
$\mathrm{React}_{\mathrm{CBCj}}$	8.6×10^4	9.8×10^5	3.0×10^5
$\mathrm{Produce}_{\mathrm{CApAq}}$	2.1×10^5	2.2×10^5	1.2×10^6
$\mathrm{Produce}_{\mathrm{ABrBs}}$	6.0×10^5	4.6×10^5	1.5×10^6
$\mathrm{Produce}_{\mathrm{BCjCk}}$	1.6×10^6	2.4×10^5	2.6×10^6

Table S1: Independently measured rate constants (all in /M /s) for designed strand displacement and toehold exchange reactions in the Displacillator (Design 4). The reactions corresponding to the notation for rate constants are specified in Equations 1 - 4. Note that rate constants involving the catalytic Helper pathway have not been characterized. Sec. S8.2 discusses the experimental methods and modeling used for estimating these rate constants.

Threshold	Measured rate constant (/M /s)
k_{ThA}	7.4×10^5
k_{ThB}	1.7×10^6
k_{ThC}	1.2×10^6

Table S2: Independently measured rate constants (all in /M /s) for the consumption of signal strands (Ap, Br, and Cj) by thresholds ($\mathrm{Th_A}$, $\mathrm{Th_B}$, and $\mathrm{Th_C}$, respectively), from Design 4. Equation 5 specifies the reaction and the notation used for the rate constant. Sec. S8.2 discusses the experimental methods and modeling used for estimating these rate constants.

S2.5.1 Kinetic measurements for individual strand displacement and toehold exchange reactions

Rate constants for the following reactions comprising the DNA implementation of the autocatalytic module $\mathrm{B} + \mathrm{A} \rightarrow 2\mathrm{B}$ were measured independently. Similar measurements were made for the other autocatalytic modules. All these measured rate constants are specified in Table S1. Figs. S3- S5 provide diagrams that illustrate these reactions.

$$\mathrm{B} + \mathrm{React_{BABr}} \underset{k_{back}{}^{BABr}}{\overset{k_{fwd1}{}^{BABr}}{\longleftrightarrow}} \mathrm{ReactInt_{BABr}} + \mathrm{Back_{BA}} \tag{1}$$

$$\mathrm{ReactInt_{BABr}} + \mathrm{A} \xrightarrow{k_{fwd2}{}^{BABr}} \mathrm{Flux_{ABr}} + \mathrm{Waste_{BA}} \tag{2}$$

$$\mathrm{Flux_{ABr}} + \mathrm{Produce_{ABrBs}} \underset{k_{back}{}^{ABrBs}}{\overset{k_{fwd1}{}^{ABrBs}}{\longleftrightarrow}} \mathrm{B} + \mathrm{ProduceInt_{ABrBs}} \tag{3}$$

$$\mathrm{ProduceInt_{ABrBs}} + \mathrm{Helper_{BBs}} \xrightarrow{k_{fwd2}{}^{ABrBs}} \mathrm{B} + \mathrm{Waste_{ABrBs}} \tag{4}$$

In addition, the kinetics of the consumption of Br by the threshold $\mathrm{Th_B}$ was also measured. Note that the analogous reaction for the consumption of Bj was not characterized; it was assumed that the rate constant for that reaction would be similar. The kinetics of consumption of Cj and Ap were also characterized. All these rate constants are listed in Table S2. Fig. 2D of the main text and Fig. S11 and Fig. S12 provide diagrams that illustrate these reactions.

$$\mathrm{Br} + \mathrm{Th_B} \xrightarrow{k_{ThB}} \mathrm{Waste_{Br}} \tag{5}$$

S2.6 Estimating leak rates for the three autocatalytic modules

Each designed strand displacement reaction pathway in a reaction network contributes unintended, or leak, reaction pathways (see Sec. S3.1). Some leak pathways are active before all reaction components are mixed together in solution and alter the initial species concentrations. Therefore, we must estimate the extent that leak impacts initial conditions for each reaction pathway, separately, and account for these errors when selecting initial species concentrations for experiments involving the full rock-paper-scissors reaction network. In this section, we describe our method of measuring leak parameters from autocatalyst reactions of the type shown in Fig. S13.

These experiments show the relationship between initial Threshold concentration and the time delay before entering the exponential growth phase of the reaction. To estimate the initial leak from these data, we assume the following:

1. Two sources of leak
   (a) Initial leak, which instantly consumes an Helper and releases output strand.
   (b) Gradual leak, which is a constant rate of Helper consumption and output strand release.
2. Thresholding is faster than the autocatalytic reaction. That is, gradual leak interacts only with Threshold complexes until they are all spent.
3. When there are no longer active Threshold complexes, the system enters the autocatalytic phase and rapidly accelerates Helper consumption.

Assumptions 1 and 2 imply a linear relationship between initial Threshold concentration and delay, the time to trigger all of the Threshold complexes and initiate the autocatalytic phase of the reaction. Assumption 3 suggests that t_c, the moment all Threshold complexes have been triggered, can be defined as the moment the Helper consumption rate surpassses a threshold rate. For gradual leak α, initial leak β, and threshold-crossing time t_c

$$[\text{Threshold}](t) = [\text{Threshold}]_0 - \alpha t - \beta$$

$$[\text{Threshold}](t_c) = 0$$

$$[\text{Threshold}]_0 = \alpha t_c + \beta$$

We find parameters α and β by determining the threshold-crossing times t_c for each experimental sample and performing linear regression on the data points $(t_c, [\text{Threshold}]_0)$. These leak parameters were used in the mechanistic model described in Sec. S5. This estimation procedure assumes that actual Threshold concentration is close to the desired Threshold concentration, i.e. that pipetting errors are low. The clear linear relationship between initial Threshold concentration and observed time delay (Fig. S14; bottom row) suggests that this assumption holds.

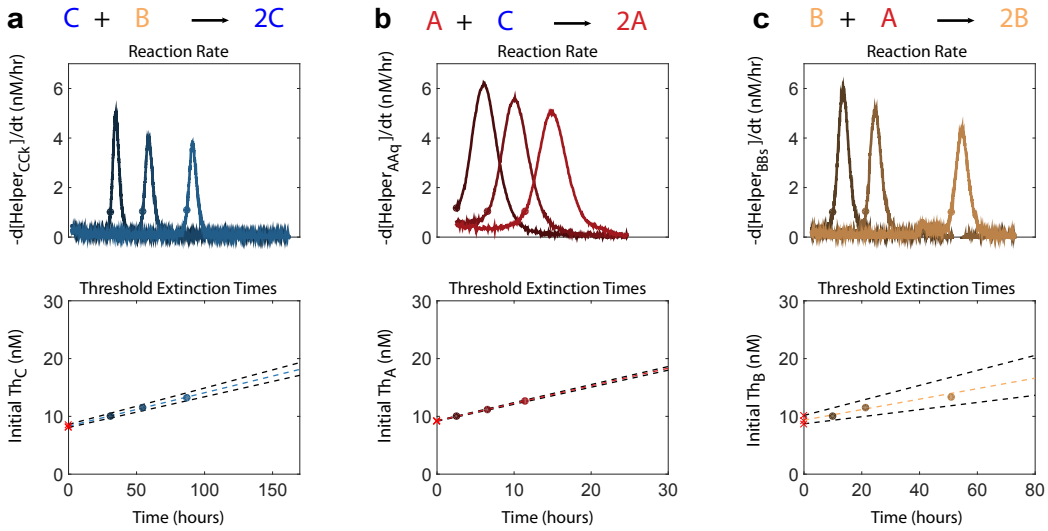

Figure S14: Leak estimation from Design 4 autocatalyst reactions. The initial and gradual leak parameters are estimated by fitting a line to the initial Threshold concentration and the time delay before the exponential phase. We define the time delay to be the first time-point at which the rate of Helper consumption is larger than 1 nM/hr, an arbitrary threshold close to the onset of exponential phase. This is represented in the first row of plots. The second row shows the threshold-crossing time-points matched with their respective initial Threshold concentrations. The dashed, colored lines are linear regression fits for each reaction pathway. The red x-marks on the Initial Threshold axes indicate the fitted initial leak parameter. To get a sense for the range of reasonable leak parameters, we performed the linear fit on all distinct pairs of data-points for each reaction, generating three estimates for each parameter. We then took the maximum and minimum value for each parameter, regardless of which data-points they are associated with, as the maximum and minimum leak parameter sets implied by the data. These extrema sets are plotted as black dashed lines with red x-marks in the second row of plots.

Module	Initial leak (nM)	Gradual leak velocity (nM/hr)	Gradual leak rate constant (/M/s)
B + A → 2B	9.35	0.091	3.13
C + B → 2C	8.31	0.058	1.99
A + C → 2A	9.36	0.295	10.07

Table S3: Leak parameters derived from autocatalyst experiments using Design 4.

S3 Discovery of the design principles through multiple iterations of the design pipeline

This section summarizes our process of discovery through multiple rounds of design. Sec. S3.1 provides a brief overview of the major sequence design considerations and challenges; Sec. S3.2 discusses Design 1; Sec. S3.3 presents the *in silico* heuristics and design process we formulated for evaluating later designs; Sec. S3.5 and Sec. S3.6 discuss Design 3 and Design 4 respectively. **All data in this work is from Design 4, unless explicitly stated otherwise in the text or in a figure caption.**

S3.1 Sequence design challenges

As described in the main text, an experimental implementation is judged based on how faithfully it captures the dynamics specified by the domain-level model. Sequence design is the process of finding DNA sequences such that the resulting molecular system stays as faithful to the domain-level abstraction as possible. The following major issues were considered during the design process:

- **Achieving desired rates for intended pathways.** In general, the problem of designing sequences for strand displacement reactions under kinetic constraints, in order to achieve a prescribed rate constant, is challenging - well-characterized rules and design tools do not exist (2–4). Therefore we attempt to control kinetics through thermodynamic proxies, such as toehold strength. At 25° C, rate constants increase exponentially with the toehold binding strength up to about 8 to 9 kcal/mol, whereupon the rate constant saturates (see Fig. 3 of Zhang & Winfree (5)). We chose our six toeholds, $\mathrm{f_A}$, $\mathrm{f_B}$, $\mathrm{f_C}$, $\mathrm{s_A}$, $\mathrm{s_B}$, and $\mathrm{s_C}$, to be roughly equally strong. For three reasons, we chose toehold energies such that strand displacement rates with those toeholds would be just within the saturation regime, i.e., between 8 and 9 kcal/mol. First, such strong toeholds result in fast strand displacement rates relative to gradual leak rates. Second, small variations in toehold energy would not be expected to result in large variation in (relative) strand displacement rates in the saturation regime. Third, stronger binding would prevent fast dissociation and lead to more toehold occlusion, as discussed below.

- **Balancing toehold strengths in different contexts**. Each toehold occurs in multiple local contexts: e.g. the internal and external contexts illustrated in Fig. S22. In particular, toehold exchange reactions in the React step involve the "forward" toehold in an external context and the "backward" toehold in an internal context. The relative rates of the forward and reverse toehold exchange steps depends crucially on the relative binding energies of the two toeholds (see Fig. 4 of Zhang & Winfree (5)). In particular, if the stronger toehold initiates displacement with roughly the same rate constant as "irreversible" strand displacement (i.e. it is in the saturation regime), then the rate constant for the reverse direction mediated by the weaker toehold will be slower than the forward direction by a factor of $e^{|\Delta\Delta G^\circ|/RT}$, where $|\Delta\Delta G^\circ|$ is the absolute difference in toehold energies. In balancing toehold strengths, we concluded that it is important to consider the energetic contributions of co-axial stacking at nicks (6), dangling single-stranded nucleotides at helix ends (7), and single-stranded tails protruding at nicks (2), in addition to the standard nearest-neighbor base-pair stacking energies (8).

- **Achieving fast unimolecular dissociation rates.** Strong toeholds are not without drawbacks. Toeholds need to be weak enough to ensure that toehold dissociation rates are fast (else, unimolecular dissociation steps would become rate-limiting). Moreover, fast dissociation rates would ensure that complexes that are not designed to interact with each other but that have complementary toehold regions will not be co-localized significantly, an effect we call toehold occlusion (see Fig. S9).

- **Avoiding unintended secondary structure.** Several species (the signal strands, Flux, Back, and Helper strands) need to be almost completely single-stranded, with no intra-molecular base pairing, most of the time. Secondary structure in key locations, such as toeholds or the first few base pairs involved in branch migration, is known to slowdown strand displacement rates (2, 9, 10).

- **Minimizing initial and gradual leaks.** As discussed below, several design rules and heuristics aim to minimize the magnitude of initial and gradual leaks. These strategies are summarized at a high level in Fig. 3 of the main text and in detail in Sec. S3.3– S3.6.

For reasons discussed below, designs 1 and 2 have opposite 5'-3' orientation for all the strands and multi-stranded complexes relative to designs 3–5 and all the domain-level diagrams.

S3.2 Sequence design 1

S3.2.1 Design criteria

Design 1-PRE (preliminary) employed the following criteria:

- All "top" strands (signal strands, Flux, Back, and Helper strands) were designed to use the 'ACT' alphabet (no 'G's). This is standard practice in dynamic DNA nanotechnology (9, 11, 12) for avoiding unintended intramolecular secondary structure.

- Toeholds were designed to be of the form "WWWSSSW" where W stands for a "weak" base (A or T) and S stands for a "strong base" (C or G). By ensuring that each toehold has exactly 3 strong base-pairs, we attempted to achieve similar toehold binding strengths. When combined with the choice of 'ACT' alphabet for top strands, this rule forces all toeholds in the top strands to be "WWWCCCW" and all toeholds in the complementary "bottom" strands to be "WWWGGGW".

- To minimize the initiation of spurious branch migration, we designed domains intended for branch migration ($\mathrm{m_A}, \mathrm{m_B}, \mathrm{m_C}$, and history domains) to be as different as possible at each end. Further, the first and last base of every branch migration domain was constrained to be a strong base, in order to reduce fraying of the helices at the ends. Sequences in the middle of branch migration regions were essentially randomly generated.

- Lastly, Design 1-PRE was verified by eye with the NUPACK web interface (13) to ensure that the desired multi-stranded complexes and intermediates were well-formed and that the top strands were (mostly) free of secondary structure.

S3.2.2 Design 1: Results

We encountered very high initial leaks (8 - 15% of the fuel concentration) with Design 1-PRE. Despite our best efforts at purification of fuel complexes, which included the use of PAGE-purified strands, ultramers, modified annealing and gel-purification protocols, we could not reduce this initial leak to much less than 10%. Further, the amount of initial leak scaled monotonically with the concentration of the fuel complex (experiments not shown). This observation suggested that the initial leak arises due to a fraction of mis-folded fuel complexes.

We also observed very high gradual leaks (20-40 /M/s). We hypothesized that gradual leaks were due to blunt-end strand displacement initiated by invasion at the ends of helices (Fig. S6C) due to fraying (2). To minimize such spurious pathways, we added 2-nucleotide clamps to the React and Produce complexes (Fig. S15). We call this design, augmented with 2-nucleotide clamps, Design 1.

Although Design 1 had lower gradual leak, the reduction was not substantial (Fig. S16). No reduction in initial leak was observed. **The 2-nt clamps present in Design 1 are also included in all subsequent sequence designs, even if they may sometimes be omitted, for convenience, in domain level diagrams**. Sequences that comprise Design 1 are provided in Sec. S8.3.

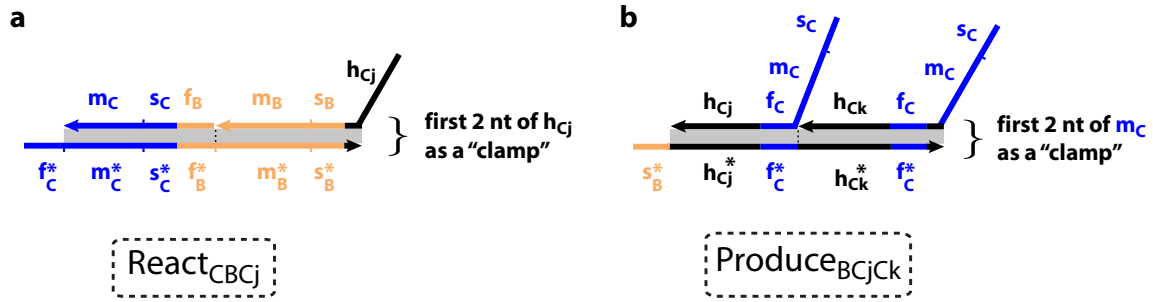

Figure S15: After preliminary experiments with Design 1-PRE demonstrated very high gradual leak rates, we added 2-nucleotide "clamps" to all React and Produce complexes as illustrated here, to obtain Design 1. These clamps are meant to mitigate some of the gradual leak pathways shown in Fig. S6, such as the React-Produce gradual leaks in panel (c). The clamps did reduce gradual leak, but not substantially. **They are included in Design 1 and all subsequent designs, even if they may sometimes be omitted, for convenience, in domain level diagrams.**

S3.2.3 Estimating gradual leak rates

Even with the 2-nucleotide clamps, we observed Produce-Helper gradual leak rates as high as 150 /M /s and React-second input gradual leak rates as high as 50 /M /s in Design 1 (Fig. S16). We now describe the procedure used for obtaining numerical estimates of gradual leak rate constants with the $\mathrm{Produce}_{\mathrm{CApAq}}$-$\mathrm{Helper}_{\mathrm{AAq}}$ leak (Sample 3 in Fig. S16) as an example.

First, the total amount "leaked" (say l, in M) was measured within an (arbitrary) time window within which the slope of the curve can be approximated to be constant. For Sample 3 in Fig. S16, this could be between 5 and 7 hours. Then, we divide l by the length of the time window (in s) to obtain the instantaneous flux f (in M/s). Assuming that the concentrations of the species involved ($\mathrm{Produce}_{\mathrm{CApAq}}$ and $\mathrm{Helper}_{\mathrm{AAq}}$) would not change significantly within the time window,

and assuming a bimolecular rate law for the gradual leak, we then estimate the leak rate constant k_{leak} (in /M /s) by dividing f by the estimated concentrations of the species involved at the beginning of the time window (5 hours). We found that this rough estimate did not usually vary more than by a factor of 2-3 for reasonable choices of the time window.

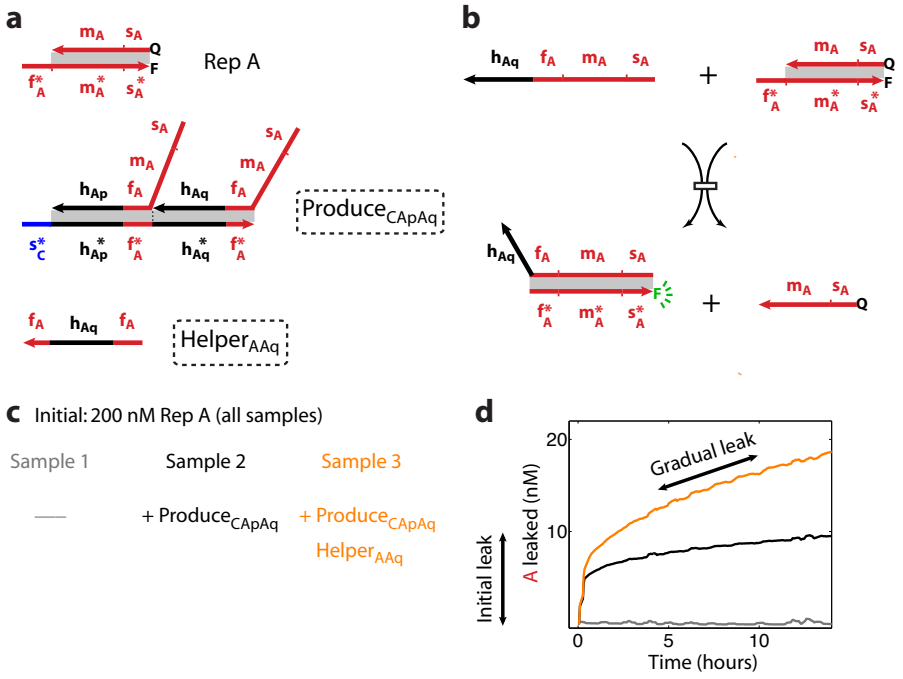

Figure S16: Leaks in Design 1, with $\text{Produce}_{\text{CApAq}}$-$\text{Helper}_{\text{AAq}}$ leak as an example. a. The molecules involved. "Rep A", short for "Reporter for A", is used for fluorescence readout of the concentration of A using the mechanism shown in panel b. b. Aq displaces the strand with the quencher (Q), leading to a waste product where the Fluorophore (F) can emit light at its characteristic wavelength. This mechanism works for Ap as well, yielding a quantitative readout of A. c. Experimental setup. All three samples are essentially negative controls for the function of $\text{Produce}_{\text{CApAq}}$, since no Flux_{CAp} is present. Sample 1 shows stable baseline fluorescence of Rep A. In Samples 2 and 3, the addition of 100 nM of $\text{Produce}_{\text{CApAq}}$ causes initial leak of A. The initial leak is higher in Sample 3 because 50 nM of $\text{Helper}_{\text{AAq}}$ was also added, which suggests that $\text{Helper}_{\text{AAq}}$ facilitates this leak. After the initial leak, the slow but persistent gradual leak of A in Sample 3 is the $\text{Produce}_{\text{CApAq}}$-$\text{Helper}_{\text{AAq}}$ leak. A much smaller $\text{Produce}_{\text{CApAq}}$- Rep A gradual leak is observed in Sample 2.

S3.3 Heuristics for evaluating sequence designs *in silico*

Our experience with Design 1 suggested that we did not really know what we should design for (or against). To understand that better, we developed some heuristic measures which attempted to quantitatively capture various undesired spurious interactions. Using these heuristic measures, we hoped to quantify the "goodness" of candidate designs and sequence design algorithms (by comparing the best designs we could obtain from each algorithm). We now introduce the heuristic

measures.

First, we define a "NUPACK interaction score", I (S1, S2), between single-stranded species S1 and S2. I (S1, S2) is the percentage of strands that are in any dimer (S1:S1, S1:S2, or S2:S2), as predicted by NUPACK when S1 and S2 are each at a concentration of 1 μM. For this calculation (i) the temperature is set to be 25°C and (ii) the salt concentration to be 1 M Na^+, and (iii) complexes comprising at most 2 strands are considered. Therefore, with some algebra,

$$\mathrm{I}\ (\mathrm{S1}, \mathrm{S2}) = 100 * \left(\frac{[\mathrm{S1} : \mathrm{S1}] + [\mathrm{S1} : \mathrm{S2}] + [\mathrm{S2} : \mathrm{S2}]}{\mathrm{c}} \right), \tag{6}$$

where c = 1 μM.

The "Top Strand Interactions (TSI)" score is the sum of interaction scores for every distinct pair of top strands (signal strands, Flux, Back, and Helper strands). That is, if $\{S_i\}$ are the top strands,

$$\mathrm{TSI} := \sum_{\substack{(i,j) \\ i \leq j}} \mathrm{I}\ (\mathrm{S_i}, \mathrm{S_j}). \tag{7}$$

The "Toehold Occlusion (TO)" score is the sum of I (t^*, S) for every toehold-complement t^* and top strand S, assuming S does not contain toehold t. If S does contain toehold t, I (t^*, S) is replaced by I ($t^*, S_{<t}$) + I ($t^*, S_{>t}$) where $S_{<t}$ is the subsequence of S, starting at the 5' end, before encountering domain t and $S_{>t}$ is the subsequence after domain t. Strictly, the above definition of $S_{>t}$ could contain an occurrence of t if t occurs multiple times in the original strand S; if that is the case, iteratively apply the same rule. That is,

$$\mathrm{TO} := \sum_{\substack{(t,S) \\ t \in \text{ Toeholds} \\ S \in \text{ Top Strands}}} \mathrm{F}\ (\mathrm{t}, \mathrm{S}), \tag{8}$$

where

$$F(t, S) := \begin{cases} I\ (t^*, S) & \text{, if S does not contain } t \\ F(t^*, S_{<t}) + F(t^*, S_{>t}) & \text{, if S contains } t. \end{cases}$$

The "Weighted Sum - Branch Migration (WS-BM)" score identifies subsequence matches (not necessarily aligned by position) between distinct branch migration domains, which include the $\mathrm{m_A}, \mathrm{m_B}, \mathrm{m_C}$ and history domains. WS-BM is essentially a weighted count of such subsequence matches. (Note that there are no intended matches between branch migration domains.) A subsequence match of length 5 contributes 1 point, length 6 contributes 2 points, length 7 contributes 4 points, and so on until lengths greater than or equal to 10 contribute 32 points.

If $\{R_i\}$ is the set of branch migration domains,

$$\text{WS-BM} := \sum_{\substack{(R_i, R_j) \\ i < j}} \left(\sum_{(s,s') \in \Phi(R_i) \times \Phi(R_j)} W_{exp}(s, s') \right), \tag{9}$$

where $\Phi(R_i)$ is the multiset of all subsequences of the sequence R_i and W_{exp} is the weight function

defined by

$$W_{exp}(s, s') := \begin{cases} 0 & \text{if } |s| < 5 \text{ or } s \neq s' \\ 2^{|s|-5} & \text{if } |s| \in [5, 10] \text{ and } s = s' \\ 32 & \text{if } |s| > 10 \text{ and } s = s'. \end{cases}$$

Note that the multiset cross-product will count each distinct pair of occurrences of subsequences. For example, if $s = TCGGA$ appears 5 times in R_i and appears 3 times in R_j, then the overall contribution due to this subsequence will be 15 for this pair of branch migration domains. Also note that subsequences of s, such as CGG will be tallied independently and additionally; we do not count just the maximal matching subsequences in this score.

The "Maximum Branch Migration subsequence (Max-BM)" score seeks to quantify the same kind of non-ideality in the design but measures the length of the longest subsequence match between distinct branch migration domains, rather than a weighted sum. That is,

$$\text{Max-BM} := \max_{\substack{(R_i, R_j) \\ i < j}} \left(\max_{s \in \Phi(R_i) \cap \Phi(R_j)} |s| \right). \tag{10}$$

The "Weighted Sum Inter-Strand (WSIS)" evaluates unintended subsequence matches between all pairs of strands in a DNA reaction network, weighted by the matching subsequence's length. This calculation finds subsequences of a strand that are exact matches to (or exact complements of) a subsequence another strand. Only spurious matches, as opposed to those guaranteed by the design specification, contribute to the WSIS. A subsequence match (based on identity or complementarity) of length 6 contributes 1 point, length 7 contributes 2 points, and so on up to matches of length 12 and above, which contribute 7 points.

$\{S_i\}$ is a set from which pairs of strands are drawn during WSIS calculation. For reasons not discussed here that relate to the operating details of our sequence design software, this set contains one copy of each fuel strand and two copies of each signal strand. As a result, all spurious matches involving one signal strand are counted twice and matches involving two different signal strands are counted four times. The software is aware of which members in the set are identical copies of a signal strand and so does not consider matches in these comparisons to be spurious.

$$\begin{aligned} \{S_i\} = \{&\mathrm{Ap}, \mathrm{Ap}, \mathrm{Aq}, \mathrm{Aq}, \mathrm{Br}, \mathrm{Br}, \mathrm{Bs}, \mathrm{Bs}, \mathrm{Cj}, \mathrm{Cj}, \mathrm{Ck}, \mathrm{Ck}, \\ &\mathrm{Back}_{\mathrm{BA}}, \mathrm{Back}_{\mathrm{CB}}, \mathrm{Back}_{\mathrm{AC}}, \mathrm{Flux}, \mathrm{Flux}_{\mathrm{BCj}}, \mathrm{Flux}_{\mathrm{CAp}}, \\ &\mathrm{Helper}_{\mathrm{BBs}}, \mathrm{Helper}_{\mathrm{CCk}}, \mathrm{Helper}_{\mathrm{AAq}}, \\ &\mathrm{ReactBot}_{\mathrm{BABr}}, \mathrm{ReactBot}_{\mathrm{CBCj}}, \mathrm{ReactBot}_{\mathrm{ACAp}}, \\ &\mathrm{ProduceBot}_{\mathrm{ABrBs}}, \mathrm{ProduceBot}_{\mathrm{BCjCk}}, \mathrm{ProduceBot}_{\mathrm{CApAq}}\} \end{aligned} \tag{11}$$

$$\text{WSIS} := \sum_{\substack{(S_i, S_j) \\ i < j}} \left(\sum_{(s, s') \in \Phi(S_i) \times \Phi(S_j)} W_{lin}(s, s') \right),$$

where $\Phi(S_i)$ is the multiset of all subsequences of strand S_i and W_{lin} is the weight function defined by

$$W_{lin}(s, s') := \begin{cases} 0 & \text{if } |s| < 6 \text{ or if } s \text{ does not match } s' \text{ or if } s \text{ is an intended match} \\ |s| - 5 & \text{if } |s| \in [6, 12] \text{ and } s \text{ matches } s' \text{ unintentionally} \\ 7 & \text{if } |s| > 12 \text{ and } s \text{ matches } s' \text{ unintentionally.} \end{cases}$$

In the implementation, matches of length up to 12 were considered.

As an example, suppose that the 3′ end of domain m_A has 2 nucleotides unintentionally identical to the 3′ end of domain h_{Aq}. Then if strand S_i is signal strand Ap and strand S_j is the catalytic $\mathrm{Helper}_{\mathrm{AAq}}$, there will be an exact identity match between the 3′ half of signal strand Ap (domains h_{Ap}, f_A, and 2 nt of m_A) and the corresponding part of the catalytic Helper strand (domains h_{Ap}, f_A, and 2 nt of h_{Aq}). Accordingly, this would be identified as an unintentional match of length greater than 12 and scored as specified above (note however that our implementation considers spurious matches of length up to 12 only). The merit of scoring this "2 nucleotide" sequence design error as a "12 nucleotide unintentional match" is that sequence uniqueness near junctions can be especially important for self-assembly and structural integrity (14).

The "Weighted Sum Inter-Strand Mismatch (WSIS-M)" is identical to the WSIS score, except that it only counts subsequence matches (either based on identity or complementarity) that have a 1-base mismatch. To illustrate with a particular example, subsequences "ATAACCA" and "ATATCCA" would be considered a match.

Lastly, this design process involving custom heuristic scores was further updated before it was incorporated into the automated Piperine compiler for enzyme-free nucleic acid dynamical systems. These changes, which include inferences from debugging experiments performed with Designs 3 and 4, are outlined in Sec. S6.

S3.3.1 Candidate sequence design methods

Since (i) toehold strengths are critical for controlling reaction kinetics and (ii) excessive spurious interaction between toeholds assumed to be orthogonal at the domain level is undesirable, we decided to design the toeholds first. Our objective was to obtain toeholds strong enough to elicit fast kinetics, with toehold strengths being as close as possible, yet as mutually orthogonal as possible. For this purpose we used StickyDesign (15), a software package used to design "sticky ends" for experimental DNA-tile based self-assembly (16). Without accounting for flanking bases (which do affect nearest neighbor contributions to the toehold strength), for Design 2 we targeted toehold strengths between 8.0 and 9.0 kcal/mol and toehold cross-talk binding energies of less than or equal to 2.5 kcal/mol. (Note that for Design 5, and the Piperine compiler, a more sophisticated energy model was used, which better accounts for flanking bases, toehold truncations, neighboring tails, etc., as described in Sec. S6).

We also enforced the following constraints in order to mitigate leak reactions. Since fraying due to thermal fluctuations at the ends of helices is thought to contribute to gradual leak pathways of the kind in Fig. S6 (2, 9, 10), we constrained helices and junctions to end with two strong (C/G) base pairs. As a direct consequence of this decision, we had to choose between (i) allowing 1-2 base pairs to continually branch migrate back and forth on either side of the junction in the Produce complexes or (ii) violate the 'ACT' alphabet rule by incorporating Guanines (G) at specific places in top strands (see Fig. S17). Since option (i) could complicate toehold energy calculations, we chose to violate the ATC alphabet.

Once the toeholds were chosen, we tried multiple sequence design algorithms that were available at that time to design the rest of the system. Among these were SpuriousSSM (17), which minimizes unintended subsequence matches within and between complexes (a variant on "sequence symmetry minimization" (14)); DomainDesign (9), which focuses on designing domains of user-specified lengths to be as orthogonal as possible to each other based on thermodynamic calculations;

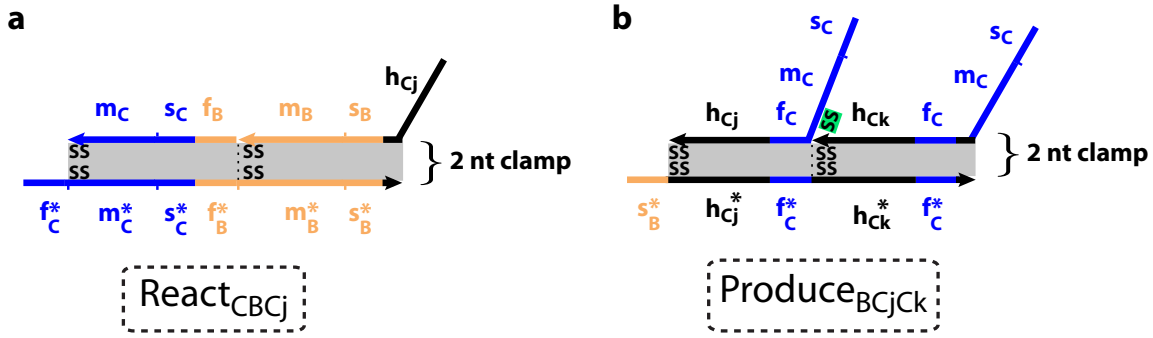

Figure S17: To reduce gradual leaks that are thought to be facilitated by thermal fraying of base pairs at junctions or ends of helices, we attempted to close such helices and junctions with two consecutive "strong" (S) base pairs. This strategy is illustrated with the React (a) and Produce (b) complexes. b. In this strategy, there is a trade off between ACT alphabet and preventing branch migration at the junction in the Produce complexes. With the ACT alphabet, back and forth branch migration of 2 nucleotides around the junction is unavoidable since both $\mathrm{m_c}$ and $\mathrm{h_{ck}}$ will need to begin with "CC". To avoid this, we violated ACT alphabet by fixing the first base of $\mathrm{m_C}$ (first of the two highlighted bases in this figure) to be a 'G'. In some situations, e.g. $\mathrm{React_{ACAp}}$, the second highlighted base was also a 'G' (optimized by the designer to reduce spurious scores).

and NUPACK (13, 18) which minimizes ensemble-defect (19) to match the user-specified structure for each complex based on thermodynamic calculations.

The best designs we could obtain from each candidate algorithm were compared with each other and Design 1 on the basis of our custom heuristic measures. The results are summarized in Table S4. Note that there are three different designs based on NUPACK: "NUPACK" was run without any artificial negative-design constraints. "NUPACK-ND" included additional negative design constraints, intended for minimizing spurious interactions between single strands and toehold complements, in the form of artificial target "complexes" between species that were not supposed to interact: wherein the "desired" structure was specified to be without any base pairing between the two species. "NUPACK-ND-BM" included additional such constraints between branch migration domains. In addition, the following sequences were excluded by the use of NUPACK's "prevent" constraints: "GGGG" (to avoid G-potential quartets) and "WWWWWW" (6 contiguous weak A/T base-pairs).

Table S4 suggests that, given our toeholds, SpuriousSSM and DomainDesign gave the best designs according to our heuristic measures. Some caveats are in order while interpreting the results. First, our heuristic measures include measures that focus on spurious matches at the level of sequence identity, without a thermodynamic or kinetic evaluation of how physically important those spurious sequence identity matches might be in the test tube. Second, at the time this analysis was performed, NUPACK's second generation algorithms for "test tube design" (20), which perform negative design against competing complexes which could form in a user-specified test tube, had not been released. We have not tested the performance of the second generation NUPACK sequence design algorithms in this analysis. Further, we have no compelling evidence that, with respect to experimental performance, the heuristic design methods adopted here are preferable than the alternatives; we just know that they are good enough. Design 4 and Design 5 – designed using a

Method	TSI	TO	WS-BM	Max-BM	WSIS	WSIS-M
Design 1	0.8	**466.0**	18	6	**12469**	**8009**
SpuriousSSM (Design 2)	7.2	21.6	10	6	30	2514
DomainDesign	12.2	19.6	7	6	1239	4016
NUPACK	1.6	56.2	**73**	**10**	**13574**	**8674**
NUPACK-ND	1.4	14.8	**306**	**13**	**23529**	**27578**
NUPACK-ND-BM	2.2	34.0	9	**8**	**25872**	**8429**

Table S4: Performance of the best designs from various sequence design methods on our custom heuristic measures. The heuristic measures and sequence design methods used are described in the text. Scores in bold were thought to be unacceptably high relative to other methods; this determination was based on a subjective intuition acquired by repeated design attempts.

refinement of the approach described here, as discussed in Sec. S3.6 and Sec. S6 respectively – both resulted in functional DNA strand displacement oscillators. A future experimental study systematically comparing design methods for DNA strand displacement dynamical systems would be highly desirable.

S3.4 Sequence design 2

Based on the analysis in Sec. S3.3.1, we decided to use SpuriousSSM for designing our second attempt. Experimentally, we found that Design 2 had much less gradual leak — as low as 1-10 /M /s. Gradual leaks of this magnitude are illustrated in Fig. S18 with an example from Design 3. However, Design 2 did not show any reduction in initial leak, which remained high (10% of the fuel concentration).

S3.5 Sequence design 3

Typically, a domain level design specification can function with either orientation (5'-3') for the DNA strands involved. However, it has been suggested in the literature (21) that certain 5'-3' orientations for the molecules may be preferable in practice because of asymmetries in the distribution of synthesis errors along the 5'-3' axis. In the hope that reversing 5'-3' orientations might change the distribution of synthesis errors to a more favorable one, which may in turn result in lower initial leak, we decided to try Design 3, which is the same as Design 2 but with 5'-3' orientations reversed.

Since the free energy contributions of individual nearest-neighbor base pair stacks towards double helix stability are not symmetric with respect to 5'-3' orientation, reversing the orientation of our design would perturb the thermodynamics of all our domains, including toeholds. This is undesirable as it could potentially alter the kinetics of desired strand displacement pathways. In spite of this, we went ahead with testing Design 3.

Experimentally, we found that Design 3 did have much lower initial leaks — reduced to 3-5% of the fuel concentration (see Fig. S18). This was an improvement from 10% in Designs 1 and 2. In addition, gradual leaks remained low, except in one particular case, that of $\mathrm{Produce}_{\mathrm{CApAq}}$ and $\mathrm{Helper}_{\mathrm{AAq}}$, where it was very high, approximately 150 /M /s.

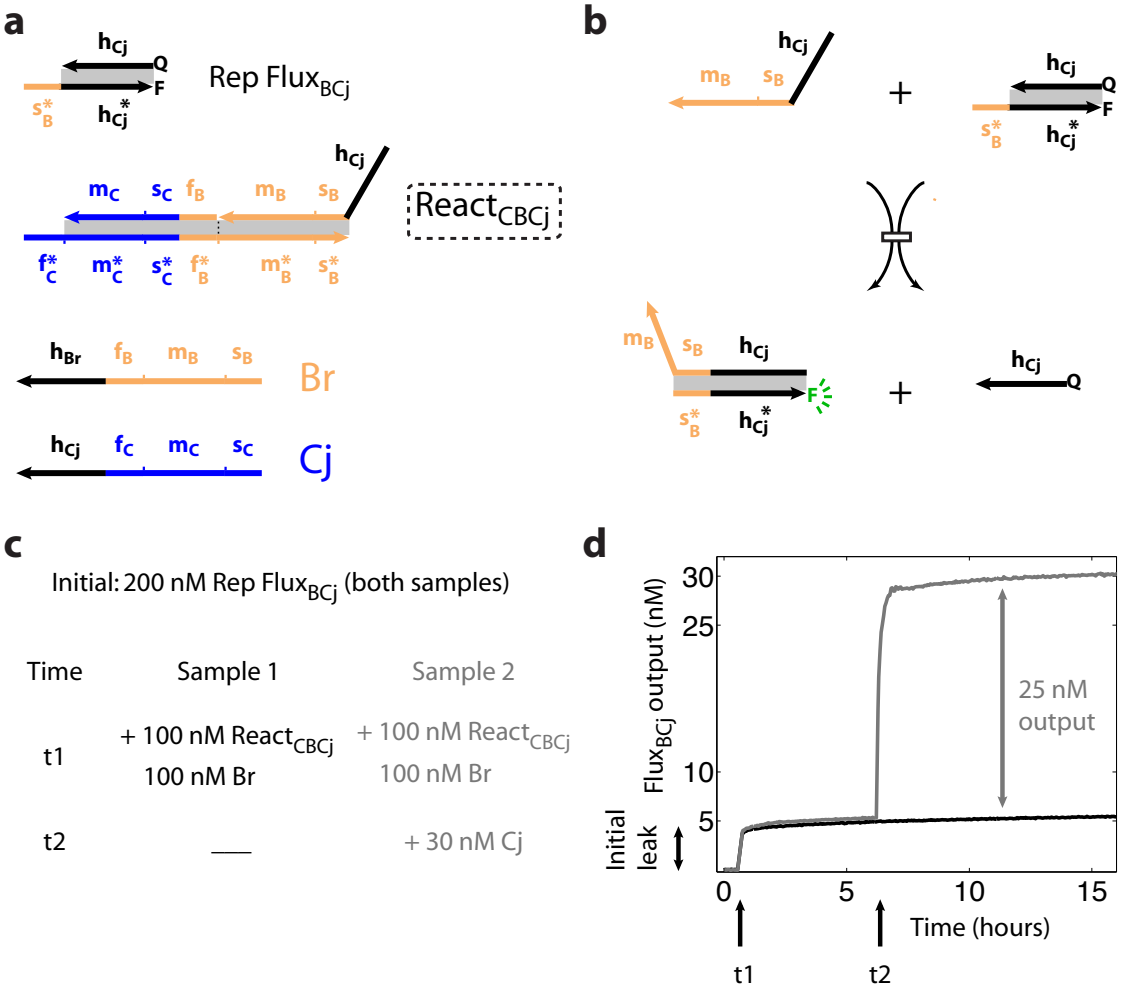

Figure S18: Reduced initial and gradual leaks in Design 3, illustrated with $\mathrm{React_{CBCj}}$-Br leaks. a. The molecules involved. Rep $\mathrm{Flux_{BCj}}$ is a reporter for $\mathrm{Flux_{BCj}}$, which works like Rep A in Fig. S16. For convenience the mechanism is illustrated in (b). c. Experimental setup. Both sample 1 and 2 contain 200 nM Rep $\mathrm{Flux_{BCj}}$ initially, and 100 nM $\mathrm{React_{CBCj}}$ and 100 nM Br are added to both at time $\mathrm{t_1}$. The initial leak is under 5%, which is 2x-3x lower than in Designs 1 and 2. After the initial leak goes to completion, only Sample 2 is triggered with 30 nM of Cj at time $\mathrm{t_2}$. Note the fast triggering in Sample 2 and the much reduced gradual leak in Sample 1 (in general, 5x-10x lower than Design 1; see Fig. S16 for an example).

S3.5.1 Understanding the exceptionally high Produce-Helper leak

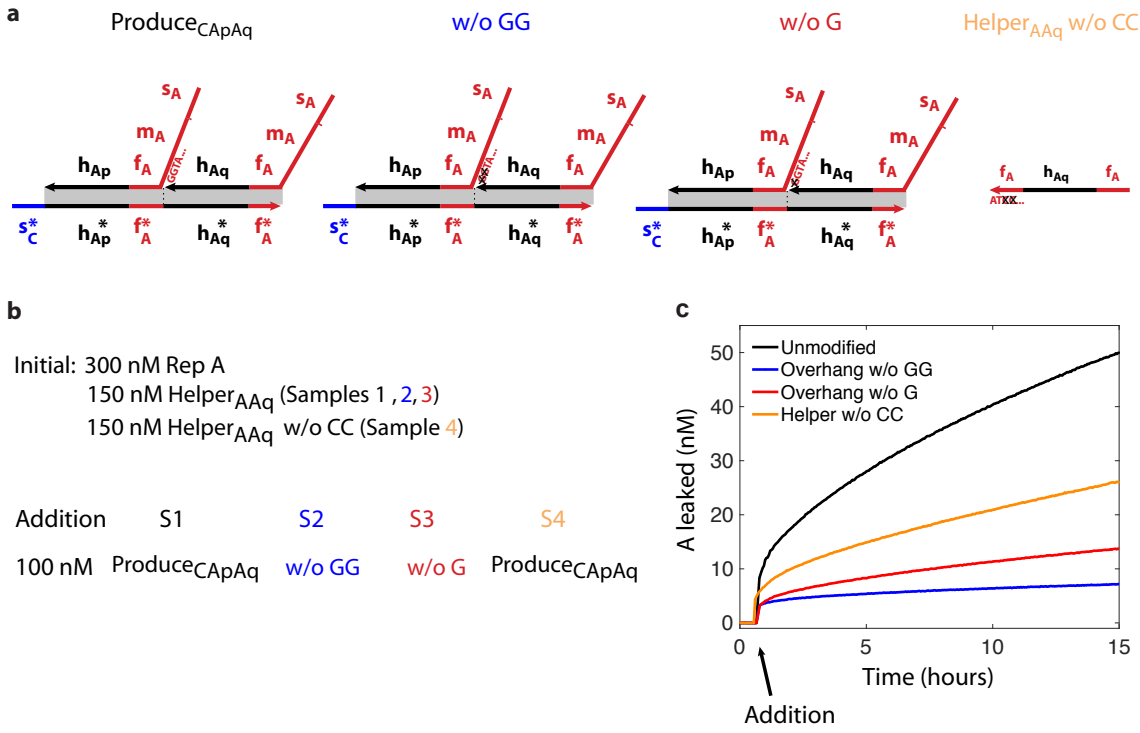

Figure S19: Experiments measuring gradual leak rates between modified versions of $\mathrm{Produce_{CApAq}}$ and $\mathrm{Helper_{AAq}}$ from Design 3. Quantitative estimates of bimolecular rate constants for these leak pathways, calculated according to the methodology described in Sec. S3.2.3, are listed in Tab. S5. In panel b, Rep A is a fluorescent "reporter" for the signal strand A; the readout mechanism is shown in Fig. S16.

In order to understand the exceptionally high leak between $\mathrm{Produce_{CApAq}}$ and $\mathrm{Helper_{AAq}}$ (150 /M/s in 12.5 mM $\mathrm{Mg^{++}}$ and 80 /M/s in 0.5 M $\mathrm{Na^{+}}$), we performed careful debugging experiments where we measured the gradual leak between various modified versions of the $\mathrm{Produce_{CApAq}}$ and $\mathrm{Helper_{AAq}}$ species, including 1-2 base deletions. These experiments are summarized in Fig. S19; the inferred bimolecular leak rate constants, according to the methodology described in Sec. S3.2.3, are summarized in Tab. S5. These results suggest that an interaction between the partially complementary bases shown in Fig. S20 ("GGTA" on the overhang near the junction in $\mathrm{Produce_{CApAq}}$ and "ATCC" in the $\mathrm{Helper_{AAq}}$ strand) could be responsible for accelerating the leak reaction between these two fuel species. We hypothesize a remote-toehold mechanism for this leak: the complementarity between 'CC' of $\mathrm{Helper_{AAq}}$ and 'GG' of $\mathrm{Produce_{CApAq}}$ could co-localize the molecules fleetingly to accelerate strand displacement, acting similarly to a strong 2-base pair toehold (22). We design against this particular gradual leak pathway in Design 4 (Sec. S3.6).

Species 1	Species 2	k_{leak} (/M/s)
$\text{Produce}_{\text{CApAq}}$	$\text{Helper}_{\text{AAq}}$	82
$\text{Produce}_{\text{CApAq}}$ w/o G	$\text{Helper}_{\text{AAq}}$	11
$\text{Produce}_{\text{CApAq}}$ w/o GG	$\text{Helper}_{\text{AAq}}$	3
$\text{Produce}_{\text{CApAq}}$	$\text{Helper}_{\text{AAq}}$ w/o CC	28

Table S5: This table lists quantitative estimates of bimolecular rate constants for the leak rates measured by the experiments described in Fig. S19, from Design 3.

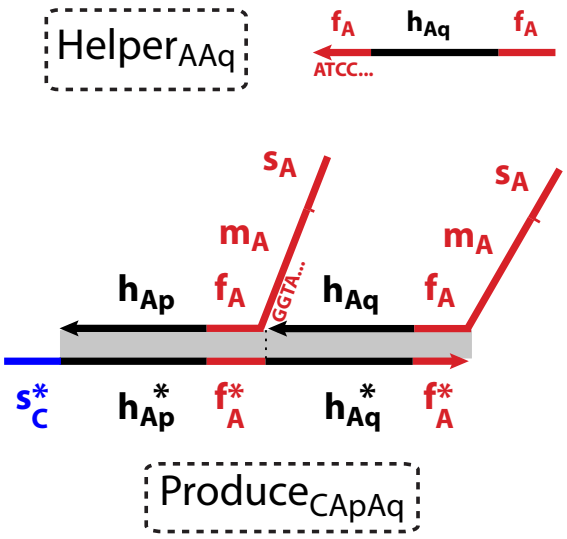

Figure S20: Based on experiments measuring gradual leak rates with single-base changes at the positions illustrated ('ATCC' in $\text{Helper}_{\text{AAq}}$ and 'GGTA' in $\text{Produce}_{\text{CApAq}}$; see Tab. S5), these bases contribute to the high gradual leak between $\text{Produce}_{\text{CApAq}}$ and $\text{Helper}_{\text{AAq}}$ in Design 3. We hypothesize a remote-toehold mechanism for this leak: the complementarity between 'CC' of $\text{Helper}_{\text{AAq}}$ and 'GG' of $\text{Produce}_{\text{CApAq}}$ could co-localize the molecules fleetingly to accelerate strand displacement, acting similarly to a strong 2-base pair toehold (22).

S3.5.2 The role of cat-ion choice in gradual leak: Mg^{++} vs Na^{+}

While discussing leak measurements for Design 3 with Paul W. K. Rothemund, he suggested that performing these experiments with 0.5 - 1 M Na^{+} (as opposed to 12.5 mM Mg^{++}) may result in lower gradual leak since Mg^{++} is known to stabilize DNA-DNA junctions (23–25) (which could in turn facilitate gradual leak pathways, by, for example, stabilizing the invasion of Helper strands at the junction of the Produce complex).

When we repeated gradual leak measurements with Design 3 in 0.5 M Na^{+}, we found that there was a reduction (across the board) by approximately a factor of 2. So, we altered our protocol at this stage to use 0.5 M Na^{+} instead of 12.5 mM Mg^{++}. Even though DNA strand displacement kinetics in high sodium (0.5 -1 M Na^{+}) (26, 27) was observed to be quite similar to that in 12.5 mM Mg^{++} (5), we experimentally verified that the kinetics of our desired pathways did not slow down significantly due to the change in salt conditions (experiments not shown).

S3.5.3 Kinetics of desired pathways

Apart from investigating leaks, we also experimentally checked whether the desired strand displacement reactions were occurring with (roughly) the expected kinetics. We found that $\mathrm{React}_{\mathrm{BABr}}$ and $\mathrm{React}_{\mathrm{ACAp}}$ were slow to "trigger" — that is, when both their inputs were present, $\mathrm{React}_{\mathrm{BABr}}$ and $\mathrm{React}_{\mathrm{ACAp}}$ were much slower to release their outputs than expected. Compared to $\mathrm{React}_{\mathrm{CBCj}}$, where this triggering process takes about 20 minutes (Fig. S18), $\mathrm{React}_{\mathrm{BABr}}$ and $\mathrm{React}_{\mathrm{ACAp}}$ took about 10 hours (slower by a factor of 30).

We re-examined the MFE structures and partition functions of all the complexes involved in these slow strand displacement reactions in NUPACK. We found that the signal strands Ap and Aq, which serve as inputs to both $\mathrm{React}_{\mathrm{BABr}}$ and $\mathrm{React}_{\mathrm{ACAp}}$, had formed unintended hairpin structures in Design 3 (Fig. S21). In addition to the MFE structures, we found that the first two bases of the branch migration domain $\mathrm{m_A}$, both G's, were bound almost all the time in some (weak) hairpin or the other. This location is especially critical, as initiation of the first branch migration step is known to be among the slowest unimolecular steps in the strand displacement process and is important in determining kinetics (2). Given that Ap and Aq serve as inputs to both React complexes, we hypothesized that this unintended secondary structure was responsible for the slowdown in triggering both React steps.

S3.6 Sequence design 4

In Design 4, we wanted to modify Design 3 to (i) speed up the triggering of $\mathrm{React}_{\mathrm{BABr}}$ and $\mathrm{React}_{\mathrm{ACAp}}$ by removing unintended secondary structure in Ap and Aq (Fig. S21) and (ii) reduce the gradual leak between $\mathrm{Produce}_{\mathrm{CApAq}}$ and $\mathrm{Helper}_{\mathrm{AAq}}$ by eliminating the remote-toehold mechanism postulated in Fig. S20.

First, we constructed a new heuristic score that seeks to quantify the presence of significant secondary structure in strands that are intended to be single-stranded. For a given base b in a strand S, let $\mathrm{p}_{\mathrm{unpaired}}(b, S)$ be the fraction of the population that is unpaired at equilibrium in a solution where only S is present, as predicted by NUPACK (considering single-stranded complexes only). Then the heuristic measure, which we call Single-Strand Unpaired (SSU), is the minimum of $\mathrm{p}_{\mathrm{unpaired}}$ across all bases in all strands that are supposed to be free of secondary structure. That is, if T is the set of all strands that are supposed to be free of secondary structure,

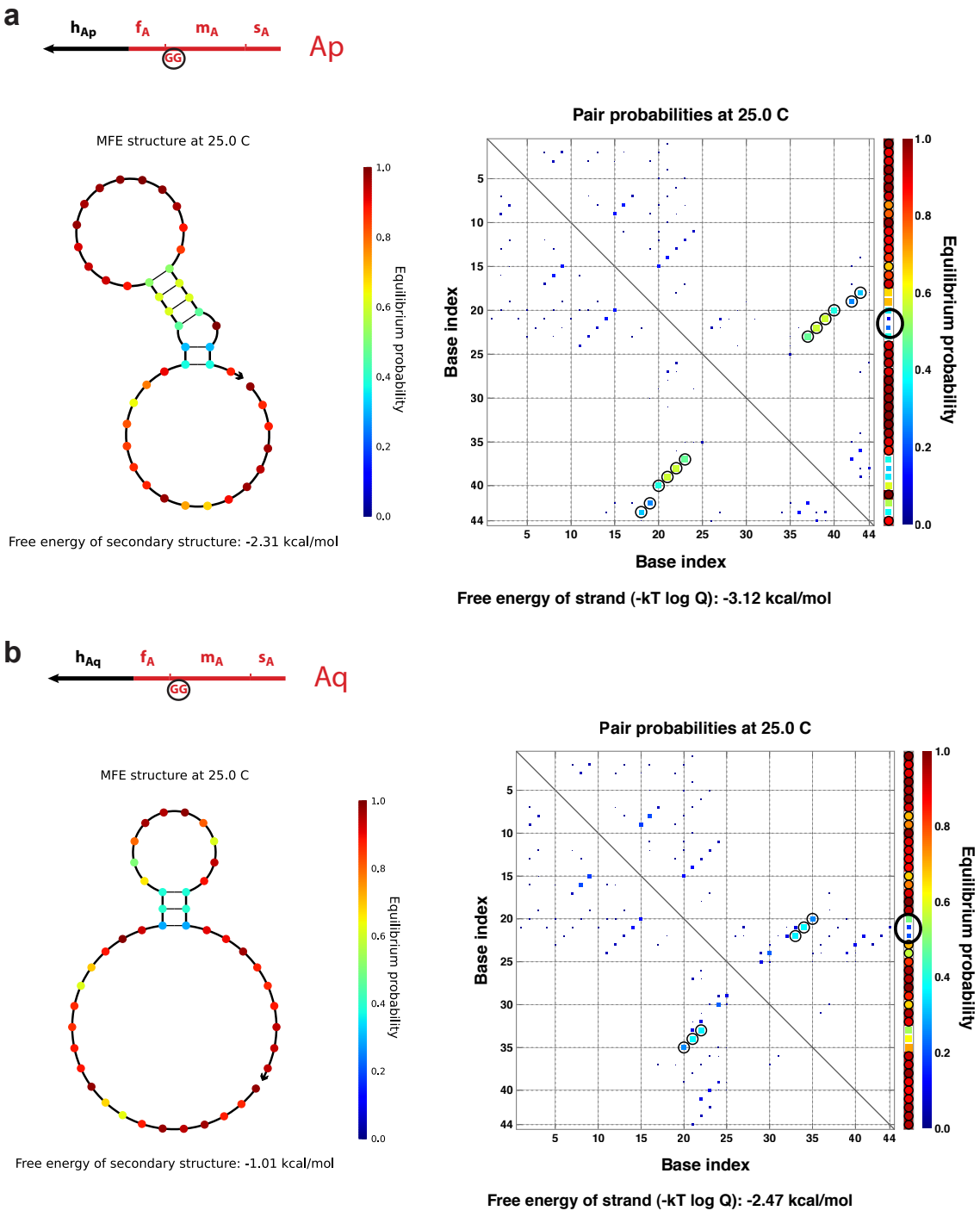

Figure S21: NUPACK predicted MFE structure and pair-probabilities matrix for Ap (a) and Aq (b) from Design 3. This secondary structure could slow down desired strand displacement pathways involving these strands. In particular, the first two bases (GG) of the branch migration region are base-paired most of the time (as a part of several weak hairpins).

$$\mathrm{SSU} := \min_{S \in T} \min_{b \in S} \mathrm{p}_{\mathrm{unpaired}}(b, S). \tag{12}$$

For Design 3, the SSU was only 0.13, and the base which was paired the most was one of the first two bases in the branch migration region $\mathrm{m_A}$ (illustrated in Fig. S21). This in itself could account for a 10-fold slowdown in strand displacement reactions involving Ap or Aq. Note that, among all the heuristic measures we have defined, SSU is the only measure where "larger is better".

We decided to modify Design 3 "by hand" to mitigate the problems described above. We identified 8 bases that, if changed, seemed to mitigate one or both of those issues, and identified by trial and error what degrees of freedom those bases had (that is, which options for their new identity did not introduce new secondary structure or malformed complexes as predicted by NUPACK). We changed 3 bases by hand as there seemed to be a clear "best choice" for those 3 bases. For the other 5 bases, we evaluated all possible candidates by brute force on our heuristic measures and chose a design that seemed satisfactory on all those measures. In particular, the SSU for the new design, which we call "Design 4-PRE" increased to 0.76, which was encouraging.

Experimentally, we found (i) a dramatic reduction in the $\mathrm{Produce_{CApAq}}$-$\mathrm{Helper_{AAq}}$ gradual leak, which reduced 10-fold from approximately 150 /M /s to 15 /M /s, and (ii) a dramatic (30-fold) speedup in the triggering of $\mathrm{React_{BABr}}$, which was now comparable to $\mathrm{React_{CBCj}}$. However, surprisingly, there was no speedup in the triggering of $\mathrm{React_{ACAp}}$, which suggested that another factor was responsible for the slow triggering of $\mathrm{React_{ACAp}}$.

On close re-examination of the thermodynamics of all the complexes, including intermediates, involved in triggering $\mathrm{React_{ACAp}}$, we noticed that the initial toehold exchange reaction involved the two most imbalanced toeholds. The "forward" toehold, $\mathrm{f_A}$, was the weakest and the "backward" toehold, $\mathrm{f_C}$, was the strongest. In addition to that, $\mathrm{f_C}$, being an "internal" toehold, would have an additional (coaxial) stack when bound (see Fig. S22). According to the principles of toehold exchange kinetics first explained in Zhang & Winfree (5), this imbalance of $\Delta\Delta G = \Delta G_{\mathrm{f_C}} - \Delta G_{\mathrm{f_A}}$ would slow down the forward toehold exchange rate by a factor of about $10^{\Delta\Delta G}$.

We attempted to balance these toehold energies by removing two base pairs from the internal toehold as shown in Fig. S22, effectively modifying $\mathrm{React_{ACAp}}$ and $\mathrm{Back_{AC}}$ to have a truncated toehold $\mathrm{f_C}$. We named the resulting species $\mathrm{React_{ACApi2}}$ and $\mathrm{Back_{ACi2}}$. We call this new design, augmented with the 2-base truncations described above, Design 4. Experimentally, the two base-pair truncation resulted in a dramatic 30-fold speedup. All 3 React and all 3 Produce steps in Design 4 had, according to our modeling, low enough gradual leak rates and high enough triggering rates that the DNA implementation could show oscillatory behavior.

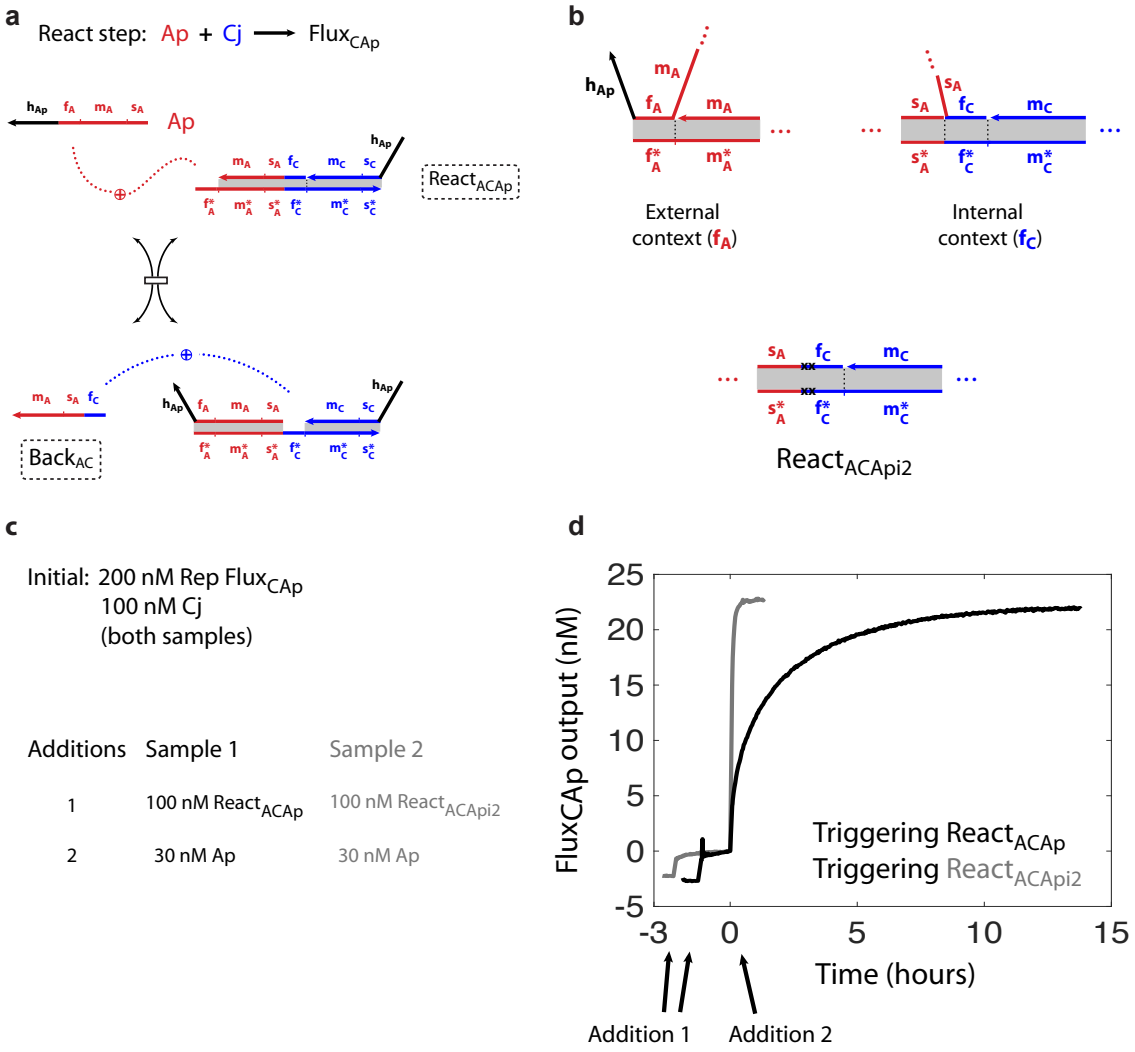

Figure S22: Energy imbalance between the external toehold ($\mathrm{f_A}$) and the internal toehold ($\mathrm{f_C}$) causes slow triggering of $\mathrm{React_{ACAp}}$. a. Recap of the toehold exchange reaction in triggering $\mathrm{React_{ACAp}}$. b. External and internal contexts. The external toehold, when bound, has only one coaxial stack, whereas the internal toehold has one on either side. In order to balance the energies, we removed two base pairs from the internal toehold as indicated by truncating the $\mathrm{Back_{AC}}$ strand and the bottom strand of $\mathrm{React_{ACAp}}$ to yield $\mathrm{Back_{ACi2}}$ and $\mathrm{React_{ACApi2}}$. c, d. Experiment illustrating that triggering $\mathrm{React_{ACApi2}}$ (Design 4) is much faster than $\mathrm{React_{ACAp}}$ (Design 4-PRE). Notice also the substoichiometric yield: addition of 30 nM of Ap results in the release of about 23 nM of $\mathrm{Flux_{CAp}}$, which is approximately 23% less than expected.

S4 Experimental implementation of the Displacillator

S4.1 Quantitative fluorescent readouts for measuring DNA implementation kinetics

Quantitative measurement of the dynamics of the DNA implementation is challenging because irreversibly consuming the signal strands A, B, C (e.g. using reporter complexes of the kind described in Fig. S16) would in itself alter the kinetics one seeks to measure. An alternative method would be to tag the signal strands directly with fluorophores; however, the only distinction between free signal strands and bound signal strands in our CRN-to-DNA scheme is the state of the first toehold (e.g. f_A). Therefore, implementing this method would involve placing a fluorophore in the middle of every first toehold, which could perturb the kinetics of the DNA implementation since toehold energies are critical for strand displacement kinetics (2, 5).

Instead, we chose to observe the consumption of the fuel species, which provides an indirect measurement of the kinetics of the DNA implementation. In particular, we tag the Helper and CatHelper strands with fluorophores at the 5' end, and the Produce complexes with quenchers at the 3' end of the bottom strand (see Fig. S10). By using a distinct fluorophore for each pair of Helper and CatHelper strands, we can measure the kinetics of all three autocatalytic modules simultaneously. In all Displacillator experiments, the CatHelper strand is also present, and the term "Helper concentration" (e.g. as used in Fig. S23, column b, bottom) refers to the sum of Helper and CatHelper concentrations.

a

Initial: 100 nM React and Back species (both samples)

Sample 1	Sample 2
10 nM Th_A, Th_B, Th_C	10 nM $Th_A^\dagger$, $Th_B^\dagger$, $Th_C^\dagger$
75 nM Helpers AAq†, BBs†, CCk†	75 nM Helpers AAq, BBs, CCk
25 nM PL-CatHelpers AAq†, BBs†, CCk†	25 nM CatHelpers AAq, BBs, CCk

Addition 1: + 100 nM Produce species to both samples

Addition 2: + (0, 10, 13) nM of (Ap, Br, Cj) to both samples

b

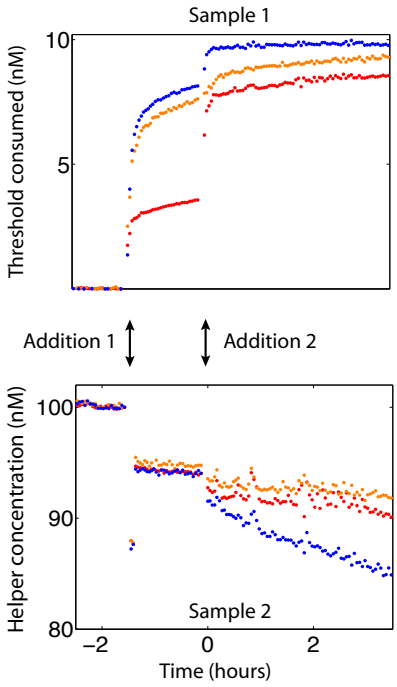

Figure S23: Experimental setup for Displacillator experiments. This data is from Design 4. Two samples are used for each experiment: Sample 1 uses "plain" versions of Helper and CatHelper (indicated with a †), which do not contain fluorophores, and thresholds $\mathrm{Th_A}$, $\mathrm{Th_B}$, and $\mathrm{Th_C}$ with fluorophores. Sample 2 uses plain versions of thresholds (indicated with a †), which do not contain fluorophores, and Helper and CatHelper strands with fluorophores. Otherwise Samples 1 and 2 are identical. (In particular, Produce complexes in both samples are labeled with a quencher on the bottom strand). The ratio of Helper:CatHelper initial concentrations was experimentally optimized to the value 3:1. a. All fuel species except Produce complexes are present initially, along with 10 nM of each threshold. After measuring the background, a mixture containing 100 nM of each Produce complex is added to each sample. This causes initial leak of signal strands A, B, and C, which is reflected in (i) the initial drop in Helper concentration (Sample 2), since Helper and CatHelper strands are consumed and quenched during initial leak and (ii) the initial spike in consumption of Thresholds (Sample 1) as the leaked signal strands are consumed. Once the initial leak stabilizes, we add (A, B, C) to kickstart oscillation with the initial conditions we desire.

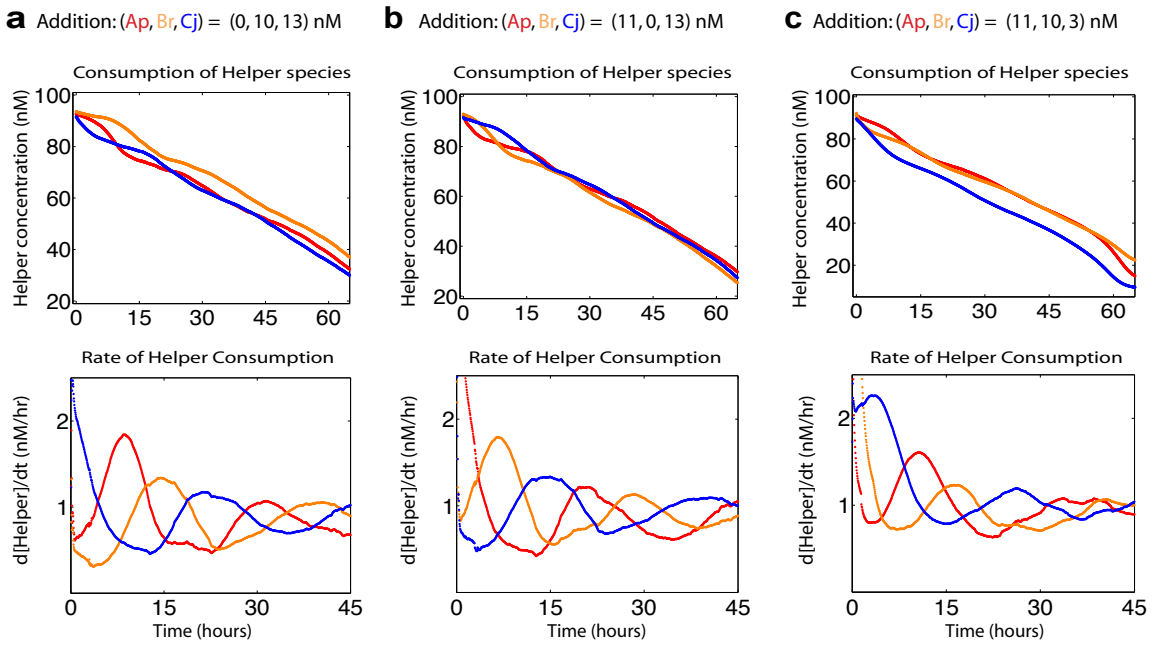

Figure S24: Displacillator experiments from multiple initial conditions (Design 4). Helper concentrations are presented in the top row; time derivatives of those measurements are presented in the bottom row. Time derivatives indicate the instantaneous rate (in nM/hr) for each autocatalytic module, apart from the reduction of Helper concentration due to gradual leak. The sequential 'firing' of the autocatalytic modules in the expected order (as long as the fuel species are at significant concentration) indicates oscillatory behavior.

S5 Mechanistic modeling

The predictable nature of the strand displacement reaction allows for kinetic modeling at the mechanistic level. Even the undesired non-idealities (leak, substoichiometric yield, toehold occlusion) are sufficiently understood that they can be captured in a mechanism-level kinetic model. In this section we describe how we constructed the mechanistic model of both the intended reactions and the undesired pathways, and how we fitted parameters to obtain a good fit with the measured oscillator dynamics. Importantly, this mechanistic model was used to obtain the extrapolated signal concentrations plotted in Fig. 4(D) of the main text.

Formal CRNs as molecular programs vs. mechanistic-level CRNs as models. It is important to keep in mind the two distinct roles that CRNs have in this work. First, a formal CRN is the molecular program that specifies the dynamics we wish to realize. For example, the rock-paper-scissors CRN (Fig. 1(A) of the main text) is the target of implementation. This is not to be confused with the mechanistic level CRN which describes the individual strand displacement steps. Our process of constructing the latter is described in Sections S5.1–S5.4 below.

The theory developed in ref. (28) for a closely related CRN implementation scheme, argues that the mechanistic CRN should approximate the formal CRN in the limit of high concentration of fuel species. However, that work did not explicitly model imperfections—leak reactions, incomplete yield, and toehold occlusion—as we do here. In Sec. S5.5 we numerically confirm that our mechanistic model well-approximates the rock-paper-scissors CRN in our regime. This allows us to argue that the cause of the oscillations in the strand displacement implementation is captured by the ideal rock-paper-scissors CRN.

Software for simulation and fitting. Mathematica (Wolfram Research) was used to simulate the mechanistic model and perform parameter fitting. The model was constructed with the help of the CRNSimulator package[1], which automatically constructs and simulates the system of ODEs for numerical simulation of a CRN.

S5.1 Desired pathway modeling

Each intended strand displacement interaction (Fig. S5–S4) is modeled as a bimolecular reaction

$$[\text{complex}]_1 + [\text{strand}]_1 \xrightarrow{k} [\text{complex}]_2 + [\text{strand}]_2, \tag{13}$$

where $[\text{strand}]_1$ and $[\text{strand}]_2$ are the displacing and displaced strands respectively, and $[\text{complex}]_1$ and $[\text{complex}]_2$ are the complexes before and after displacement. The rate constants k are potentially unique for every displacement reaction and were determined in separate experiments (see Table S1). Here we think of reversible strand displacement reactions as two separate reactions with different k. If $[\text{complex}]_2$ does not have an open toehold, we consider it to be effectively inert and omit it in the model.

As discussed in the main text and Fig. S13, we observed that the intended strand displacement reactions had less than expected yield (substoichiometric yield). This was consistent with the hypothesis that some fraction of bad fuel species consume inputs without releasing functioning outputs. To model this, we assume that a fraction of outputs of React and Produce complexes are inert

[1] Available at: `http://users.ece.utexas.edu/~soloveichik/crnsimulator.html`.

(truncated). Specifically, fraction τ of React complexes has a truncated Flux strand, and fraction $1-\tau$ is error-free. For Produce complexes, fraction τ has a truncated first output signal, fraction τ has a truncated second output signal, and the remaining fraction $1-\tau$ is error-free. (For small τ it is unlikely that both outputs are truncated.)

Together with separate reactions for React and Produce complexes with truncated fuels, the strand displacement interactions shown in Fig. S5–S4 are described by 84 chemical reactions in the mechanistic model.

S5.2 Leak pathway modeling

Of the 3 types of gradual leak shown in Fig. S6, our mechanistic model includes only the "Produce-Helper" leak. The exclusion of other leak types is based on theoretical and experimental considerations as follows.

The modeled "Produce-Helper" gradual leak is expected to be significant because it occurs between two high concentration fuel species. Further, the putative gradual leak mechanism, where the Helper strand invades at the nick of the Produce complex (between the two outputs), cannot be mitigated by clamps. Indeed, this leak was experimentally shown to be significant (Table S5).

In contrast, the omitted "React-second input" gradual leak occurs between a high concentration auxiliary species (React) and a low concentration signal species. Further, although the "React-Produce" gradual leak occurs between two high concentration auxiliary species, the clamp on the React complex is expected to significantly diminish the rate of the blunt-end strand displacement. Indeed, experimental measurements of this leak confirmed that it was at least a factor of 10 less than the "Produce-Helper" leak (data not shown).

Leak interactions result not only in undesired release of otherwise sequestered strands, but also the production of unwanted complexes that may have downstream effects. The complex produced as a result of the "Produce-Helper" leak (Fig. S6(b)) can undergo a strand displacement interaction with a Flux strand generating one signal strand (Fig. S7(b)). Note that this is fewer than the expected 2 signal strands produced by a fresh Produce complex. Thus, leak can contribute to apparent substoichiometric yield. We incorporated this reaction into our model; however, it was not sufficient to explain the dampening we observed, suggesting that other mechanisms are responsible for substoichiometric yield (see Sec. S5.1).

The 3 leak reactions (one for each of the three formal reactions) and their consequences contribute an additional 6 reactions to the mechanistic model.

In addition to the gradual leak discussed above, we observed substantial amount of initial leak (see e.g., Table S3 and Fig. S22). Initial leak mechanisms are not well understood and thus in general present a challenge to mechanistic level modeling. To avoid unnecessarily complicating our mechanistic model, we chose to simulate initial leak as, effectively, instantaneous gradual leak. In other words, we start the simulation with fraction λ of Produce and Helper fuels already leaked, resulting in a corresponding increase in initial signal along with the leaked complexes (Fig. S6, S7).

S5.3 Occlusion modeling

An open toehold domain and its complement occur not only between a complex and its displacing strand, but also in contexts where a strand displacement cannot occur (Fig. S9). These *toehold occlusion* interactions are expected to have an overall slowdown effect on the kinetics (see the

discussion of spurious toehold binding in ref. (12)). We model toehold occlusion interactions by a reversible bimolecular reaction specific to each toehold shared between complex and strand:

$$[\text{complex}] + [\text{strand}] \underset{k_{\text{off}}}{\overset{k_{\text{on}}}{\rightleftharpoons}} [\text{inactive-complex-strand}]. \tag{14}$$

The [inactive-complex-strand] species do not interact in strand displacement reactions; in this way, the amount of available [complex] and [strand] is reduced. Since leak reactions are toehold-independent, Helper strands and Produce complexes sequestered in [inactive-complex-strand] can still leak in our model. Thus the leak rate is not a function of toehold occlusion.

Kinetic parameter k_{on} was (somewhat arbitrarily) set to 2×10^6/M/s. Kinetic parameter k_{off} was fit (see below).

S5.4 Parameter fitting

Full oscillator. We fit three global parameters: τ, the fraction of truncated outputs; λ, the fraction of Produce and Helper fuels involved in initial leak; and k_{off}, the spurious toehold binding dissociation rate. Further, we fit the initial amounts of signal strands (A, B, C) for each of the three experiments that were initialized with different amounts of the signals. These additional parameters compensate for uncertainty in initial signal concentration due to leak and pipetting, as well as for inaccuracies of assuming that each reaction module is governed by identical truncation, initial leak, and toehold occlusion parameters. These initial signal values modify (add or subtract to) the amount of signals generated in the initial leak.

The Helper traces in all three initial conditions quickly reached a value consistent with roughly 0.075 initial leak λ, the value that we used throughout (see Fig. S25). The other parameters were fit to the Helper numerical derivatives (Fig. 4(B) of the main text), which are expected to be less susceptible to systematic offsets than the absolute Helper signal. The fitting is performed by Mathematica's `FindMinimum` function, with a mean squared error loss function.

In order to obtain relatively noise-free numerical derivatives of Helper consumption to use in fitting, we smoothed the collected data using a 30 datapoint moving average. The numerical derivatives were then computed using Mathematica's `DerivativeFilter` function. In order to focus on the most meaningful part of the trajectory—ignoring initial transients and the dynamics after fuels are consumed—we removed the first 20 data points (corresponding to ~ 52 minutes), and set the end of fitting at data point 1000 corresponding to ~ 50 hours.

The fitted parameters are shown in Table S6.

Autocatalytic modules. We used the three global parameters τ, λ, and k_{off} from the full oscillator fit. Since the time of onset of exponential amplification is sensitive to the exact threshold concentration, for each of the three modules we additionally fit the initial amount of threshold. (Alternatively, we could have fit three "$\Delta\lambda$" parameters to account for module-dependent initial leak; we make no claim as to the origin of threshold concentration perturbations, be they pipetting inaccuracy, leak pathways, or other effects.) The fit was to the numerical derivatives of the Helper strand concentration, computed as in the full oscillator.

τ	λ	k_{off}
0.11	0.075	0.226 /s

initial condition	A	B	C
1	-2.82 nM	-0.09 nM	-0.07 nM
2	-6.32 nM	-0.94 nM	1.06 nM
3	-3.28 nM	-4.92 nM	2.74 nM

Table S6: Fitted parameters for the full oscillator. τ: truncated output fraction, λ: initial leak fraction, k_{off}: spurious toehold binding dissociation rate. Meaning of the initial A, B, C parameters: We expect that the initial signal concentrations correspond to the amount leaked, minus the amount consumed by the Threshold complexes, plus the amount manually added. Parameter λ captures the initial leak, while the initial A, B, C parameters capture the difference between the amount manually added and the Threshold complexes. Note that the interaction of Threshold with the signals is assumed to be fast compared with the rest of the system dynamics and is not explicitly modeled here. Thus negative signal concentrations correspond to the net removal of leaked signal due to Threshold complexes. The fitting is done to minimize mean squared error of the Helper derivatives (Fig. 4(B) of the main text) as described in this section.

S5.5 Agreement of the mechanistic model with the molecular program CRN

Our pipeline starts with the formal CRN specification: in this case, the rock-paper-scissors CRN. Are the realized signal dynamics quantitatively consistent with the specification? Although we did not have direct access to signal concentrations in the test tube, we used the signal concentrations extrapolated from the mechanistic model and checked their consistency with the ideal CRN dynamics.

In our pipeline we used the topology of the rock-paper-scissors CRN as specification, but did not attempt to target particular reaction rate constants. The theoretical treatment developed in ref. (28) derives the reaction rate constants of the simulated CRN as a function of the rate constants of each strand displacement step and fuel concentrations. Thus, we utilize this analysis in order to obtain the rate constants of the ideal rock-paper-scissors CRN that should be most consistent with the mechanistic model, and thus with the test tube implementation. Note that analytically deriving the rock-paper-scissors rate constants from the mechanistic model, rather than through numerical fitting, reiterates the efficaciousness of our understanding of the mechanism.

We modify the treatment of ref. (28) in two ways. First, we take into account toehold occlusion, which slows down the kinetics. Second, we disregard the "buffering effect" that occurs when a large fraction of signals is sequestered in $\mathrm{ReactInt}$ complexes. We are justified in doing so because the measured forward (k_{fwd1}) and backward (k_{back}) rate constants for the first react step (Table S1) are strongly biased backward, suggesting that most of the signals remain free.

The mechanistic model makes the following predictions regarding toehold occlusion and fuel concentrations. The fraction of occluded signal strands varies roughly between 0.25 (initially) and 0.15 (at 60 hours). The amounts of non-occluded React fuels ranges from about 40 nM to 20 nM, and the amount of non-occluded $\mathrm{Backward}$ fuels ranges from about 60 nM to 100 nM. The variation occurs roughly linearly with the progress of the reaction over 60 hours. For each of these fractions and concentrations, we take the average of the regime limits: let $\phi = 0.8$ be the average fraction

of signal strands that are not occluded, let $c_{\text{React}} = 30$ nM and $c_{\text{Back}} = 80$ nM be the average concentrations of non-occluded React complex and Backward strand fuels, respectively.

In the following discussion, we derive the formal rate constant for the reaction $B + A \rightarrow 2B$ (see Fig. S5). The formal rate constants for the other two reactions are determined in the same manner. Let $[A]$ be the total concentration of signal strands A, with any history domain. This amount consists of two populations: the non-occluded (free) strands A_{free}, and the occluded strands A_{occ}. Thus, $[A] = [A_{\text{free}}] + [A_{\text{occ}}]$. In the same fashion, we will use subscripts "free" and "occ" to indicate non-occluded and occluded versions of other complexes and strands as well. Let $[\text{ReactInt}_{\text{free}}]$ be the concentration of non-occluded ReactInt complex (summing over the two possible history domains of the sequestered B input). The displacement of the Flux strand from ReactInt by signal A is expected to be the rate limiting step of the entire module, as argued in ref. (28). (This is essentially because all other forward reactions involve at least one fuel, which is present in high concentration. In contrast in this step both ReactInt and A are non-fuel, low concentration species.) The instantaneous rate of ReactInt + A strand displacement reaction is $k_{\text{fwd2}} \cdot [\text{ReactInt}_{\text{free}}] \cdot [A_{\text{free}}]$ which can be re-written as (by multiplying by 1):

$$k_{\text{fwd2}} \cdot \frac{[B_{\text{free}}]}{[B]} \cdot \frac{[A_{\text{free}}]}{[A]} \cdot \frac{[\text{ReactInt}_{\text{free}}]}{[B_{\text{free}}]} \cdot [B] \cdot [A], \tag{15}$$

We assume signal B (again summed over the different history domains) is in pseudoequilibrium with ReactInt, which occurs through the strand displacement reaction $B_{\text{free}} + \text{React}_{\text{free}} \underset{k_{\text{back}}}{\overset{k_{\text{fwd1}}}{\rightleftharpoons}} \text{Backward}_{\text{free}} + \text{ReactInt}_{\text{free}}$. If we further assume that the concentrations of fuels (React complex and Backward strand) are constant, we obtain the relationship:

$$\frac{[\text{ReactInt}_{\text{free}}]}{[B_{\text{free}}]} = \frac{k_{\text{fwd1}} \cdot c_{\text{React}}}{k_{\text{back}} \cdot c_{\text{Back}}}.$$

Further, assuming A_{free} and B_{free} are in pseudo-equilibrium with their occluded versions such that a fixed fraction ϕ is non-occluded, we have $\phi = \frac{[A_{\text{free}}]}{[A]} = \frac{[B_{\text{free}}]}{[B]}$. Thus, eq. 15 can be written as:

$$k_{\text{fwd2}} \cdot \phi^2 \cdot \frac{k_{\text{fwd1}} \cdot c_{\text{React}}}{k_{\text{back}} \cdot c_{\text{Back}}} \cdot [B] \cdot [A]. \tag{16}$$

Thus the rate constant for the effective reaction $B + A \overset{k}{\rightarrow} 2B$, acting on total signal strand concentrations, is:

$$k = k_{\text{fwd2}} \cdot \phi^2 \cdot \frac{k_{\text{fwd1}} \cdot c_{\text{React}}}{k_{\text{back}} \cdot c_{\text{Back}}}. \tag{17}$$

Instantiating ϕ, c_{React}, and c_{Back} with the averages discussed above, and using the separately measured k_{fwd1}, k_{back}, k_{fwd2} for each reaction module (from Table S1), we derive the following formal CRN with numerical rate constants (units of /M/s):

$$B + A \overset{18813}{\longrightarrow} 2B$$

$$C + B \overset{6318}{\longrightarrow} 2C$$

$$A + C \overset{8247}{\longrightarrow} 2A.$$

Deterministic mass-action ODE simulations of this three-reaction CRN (Fig. S26), using the same initial concentrations $[A]$, $[B]$, and $[C]$ that appear in the mechanistic model, we obtain qualitatively similar trajectories, with comparable frequencies, amplitudes, offsets, and phases.

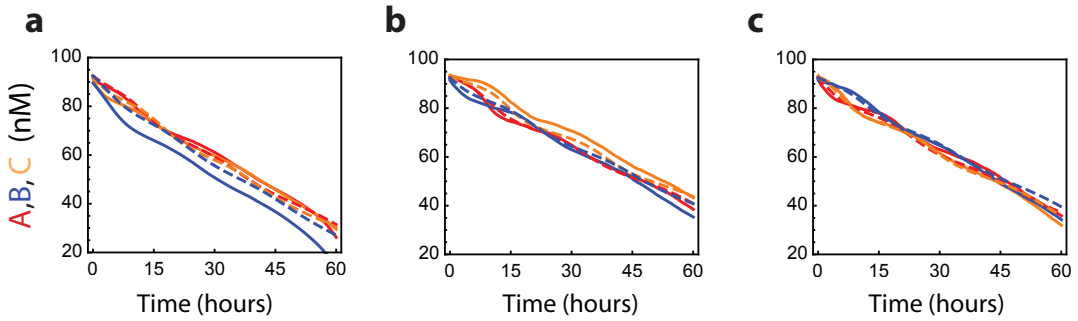

Figure S25: Measured Helper A, B, C concentrations (solid) versus model fit (dashed). (a)-(c) Three sets of initial conditions. Note that the fit is performed not to the Helper concentrations but to their derivatives (see Fig. 4(B) in the main text).

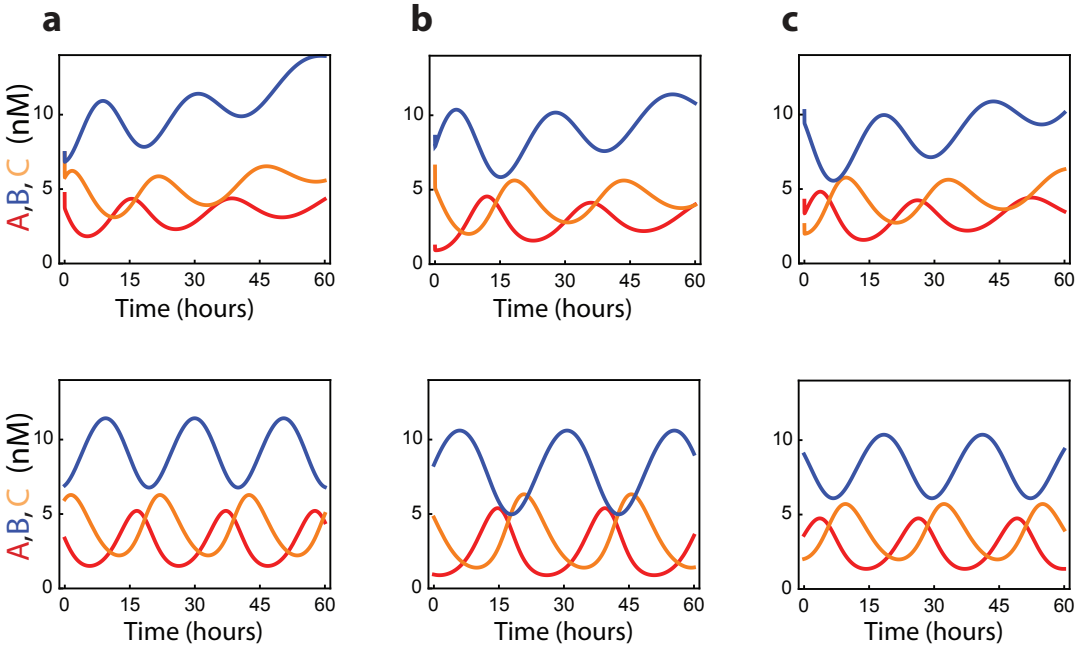

Figure S26: Extrapolated A, B, C signal concentrations (top) versus formal CRN (bottom). (a)-(c) Three sets of initial conditions.

S6 Piperine: A CRN-to-DNA compiler

Through the four rounds of sequence design and experimental debugging presented in Sections S3.2, S3.4, S3.5, and S3.6, we identified domain-level and sequence-level design principles for mitigating spurious "leak" reactions and balancing rate constants in DNA strand displacement reaction cascades. We translated these design principles into a set of quantitative heuristic functions. Each individual function in this set calculates a number that describes the violations of one design principal in a given DNA strand displacement cascade. Taken together, they detect sequence-level motifs that may contribute to poor experimental performance (See Sec. S3.3).

During the many design-experiment iterations mentioned above, these heuristics allowed us to track the incremental improvements made with each design alteration. This encouraged us to construct an automated sequence design pipeline that incorporates these heuristic functions to help ensure good experimental performance of the DNA systems it generates. To test this possibility, we wrote an automated CRN-to-DNA compiler named "Piperine", which interfaces intimately with the PepperSuite toolkit [2]. The Piperine compiler accepts a plain-text file describing a formal CRN and returns the sequences of a DNA implementation intended to approximate its dynamical behavior. This section describes the basic principles underlying our Piperine compiler; Sec. S7 describes our use of an earlier version of the software to generate another DNA implementation of the rock-paper-scissors formal CRN which demonstrated oscillatory dynamics in the test-tube with no subsequent optimization or re-design.

Piperine generates a list of candidate sequence designs implementing the target CRN, ranks them according to the heuristic measures, then selects the optimal design for the user to purchase and use in experiments. This process begins with converting lines of text, each describing a component reaction of the input CRN, into the domain-level complementarity rules for the DNA implementation. Piperine compiles these rules into nucleotide-level complementarity constraints and populates the toehold domains with sequences designed to be energetically balanced in all relevant toehold contexts. Piperine then generates nucleotide sequences satisfying these constraints through sequence symmetry minimization (14). The algorithms performing toehold design and sequence symmetry minimization are stochastic, meaning each execution using the same set of constraints yields a different sequence design. Piperine generates many candidate designs in this manner before calculating their scores according to the heuristic measures. The best sum-of-ranks over the heuristic scores is selected as the "best" design, which Piperine reports to the user. Fig. S27 shows a detailed overview of this process.

[2] Available at `http://www.dna.caltech.edu/DNA_Sequence_Design_Tools/`

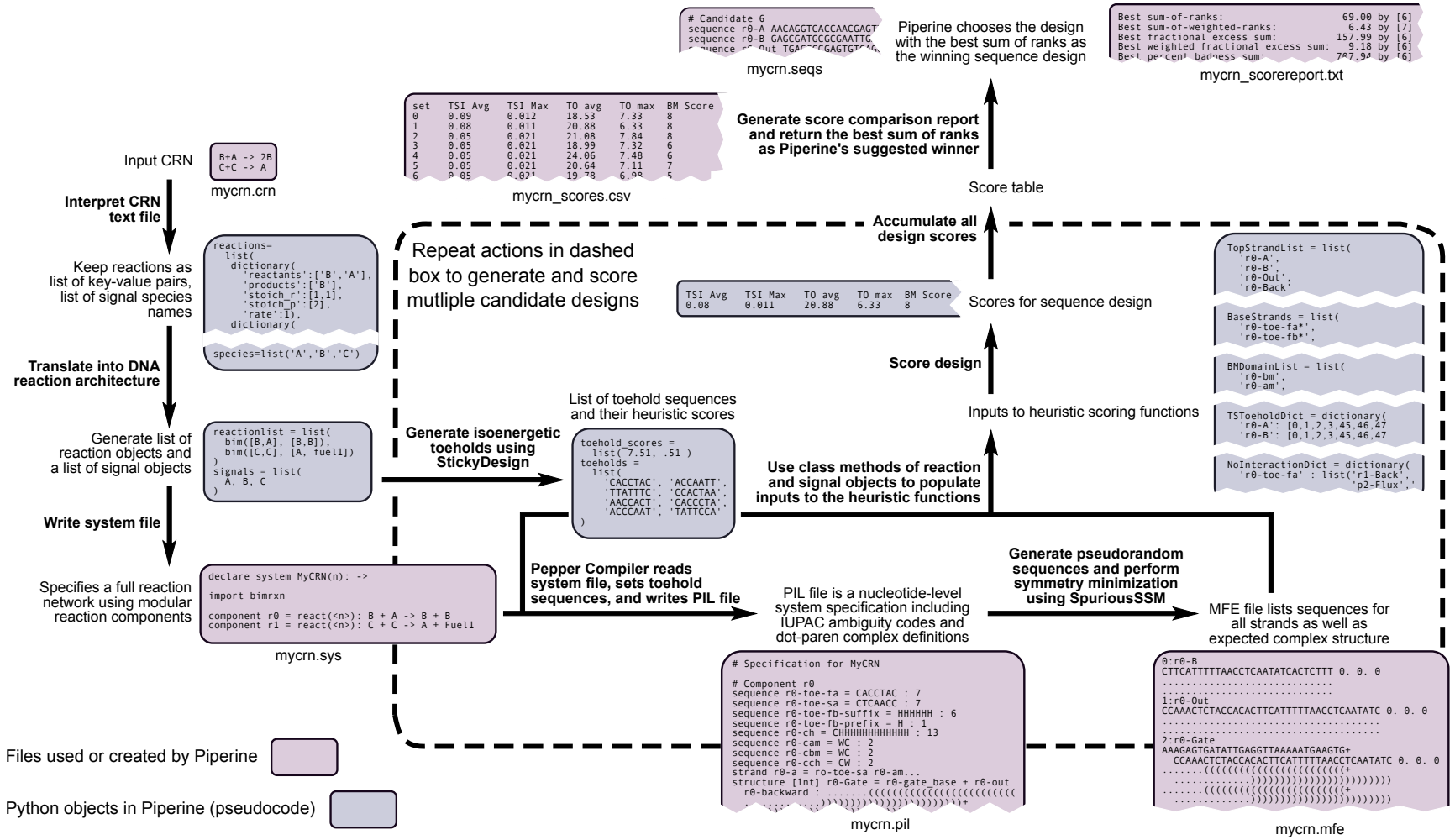

Figure S27: An overview of the Piperine sequence design pipeline, including internal data and text files from an example run of the software. Bold statements describe operations performed on data and standard-case statements explain the contents of that data. Sawtooth breaks in text bubbles indicate that a portion of that data or text is hidden for display purposes.

S6.1 Translating an abstract CRN into domain-level specification of DNA strands and complexes

Piperine translates component reactions into the strand displacement architecture shown in Fig. 1D&E, so it may only design sequences for CRNs whose component reactions have at most two reactants and two products. For formal reactions with less than two input or output strands, Piperine adds placeholders so that the final reaction has two reactants and two products. These placeholder strands have the same domain structure as signal strands, but they do not represent formal species of the CRN. They interact with only one complex in the strand displacement cascade; either they are sequestered by a React complex or released by a Produce complex. Following these rules, the DNA implementation is composed entirely of bimolecular reactions that follow the form shown in Fig. 1E of the main text. This mapping of abstract reactions to DNA strand displacement cascades, together with the design principles shown in Fig. 3 of the main text, constitute the domain-level specification for the DNA system.

Piperine accepts arguments defining rate constants and recommends experimental approaches to set the relative kinetic rates of each bimolecular reaction. Rather than exploiting differences in toehold binding energies to tune rate constants, Piperine balances the binding energies of all toeholds in all kinetically relevant contexts (Fig. S28). Of particular importance is the reversible toehold-exchange reaction that controls Backwards strand release in the React step. This is the rate-determining reaction step for the overall bimolecular architecture; all other rate dependence is a function of species concentrations (28). When toehold energies are balanced, the expected rate constant of a bimolecular reaction will be proportional to the concentrations of its respective fuel complexes. Furthermore, each rate may be tuned independently as the fuel complexes are unique to each reaction. When users specify rate constants for CRN component reaction, Piperine suggests concentration ratios for fuel species that will experimentally emulate the user's specification.

S6.2 Generating candidate sequence designs from domain-level specifications

The toehold binding energies in the reversible portion of the React step must be balanced between all bimolecular reactions. Otherwise, the rates of each React step will differ significantly between each bimolecular reaction (a pathological example is shown in Fig. S22). Piperine ensures similar rate constants for all React steps by generating toehold sequences that are within a user-defined range of binding energies. In doing so, Piperine considers both contexts shown in Fig. S28 and records the one furthest from the user-specified energy as that toehold's binding energy. The software that generates toehold sequences for Piperine, Constantine Evans' Stickydesign, restricts crosstalk in the set of toeholds and toehold complements and ensures mis-aligned toehold binding events are low-energy (16).

Piperine generates sequences for branch migration domains through sequence symmetry minimization. As described above, Piperine translates each formal reaction to a domain-level representation of a bimolecular strand displacement cascade. Piperine then passes the domain-level specification to SpuriousSSM, a sequence symmetry minimizer. SpuriouSSM stochastically produces DNA sequences satisfying the domain-level constraints and then iteratively edits undesirable sequence motifs and unintended subsequence matches until the occurrence of these features falls below a threshold. Because this process is stochastic, SpuriousSSM produces different sequences for repeated executions with identical arguments. Piperine takes advantage of this to produce multiple candidate sequence designs for the system (14).

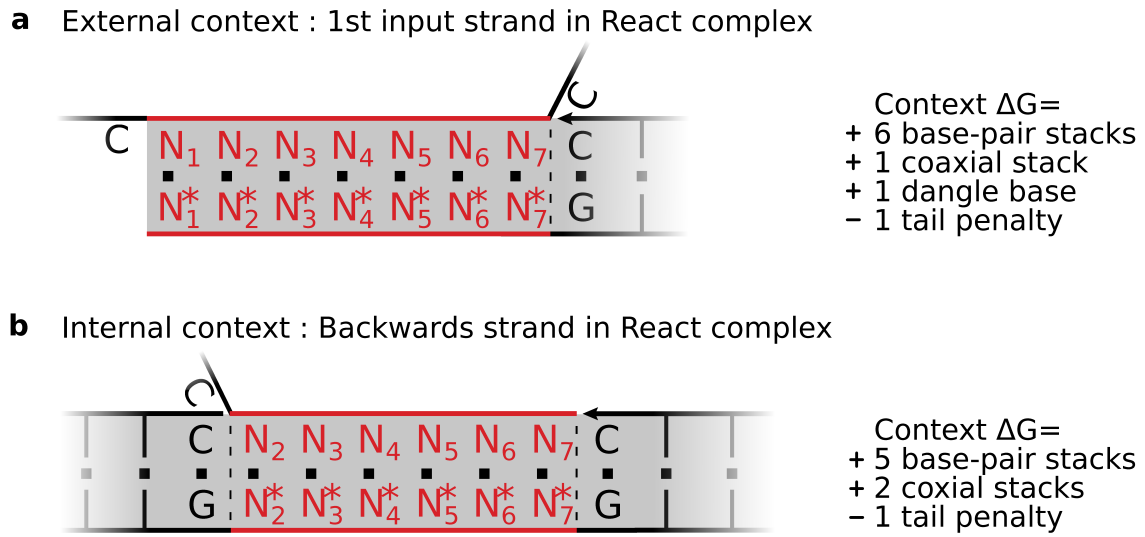

Figure S28: The expected binding energy of a toehold depends on the characteristics and identities of its flanking nucleotides, not only its nucleotide composition. Subfigures **a** and **b** show the binding contexts and energetic features of the two toehold-binding events that significantly impact a bimolecular reaction's overall kinetic rate. The indexed **N** characters represent the seven nucleotide bases, and asterixed characters their complements, that compose a toehold. The internal context uses a shorter toehold (lacking base $\mathbf{N}_1$) because a flanking coaxial stack and a base-pair similarly stabilize the bound toehold.

S6.3 Selecting a winning candidate

The heuristics described in Sections S3.3 and S3.6 are chiefly summations of a given metric over a set of strands or a set of strand groups. However, this approach may disguise the most egregious violations that significantly distort reaction dynamics on their own. Piperine determines both the summation and most extreme value under each of the heuristics to capture both the overall and worst-case non-idealities.

In addition to these heuristic measures, Piperine considers scores based on Single-Strand Toehold Unpaired (SSTU) and Bad Nucleotide Percent (BN%) scores. Based on the SSU heuristic, SSTU scoring involves finding the minimum unpaired probability for nucleotides within toehold regions and the first three bases to participate in strand displacement. This heuristic reflects the contribution of top strands' secondary structure to toehold occlusion.

The SSTU score is calculated as below, where T is the set of all strands designed to be free of secondary structure and R is a function that maps a strand to its nucleotides composing the toehold and three initial strand-displacement positions.

$$\text{SSTU} := \min_{S \in T} \min_{b \in R(S)} \text{p}_{\text{unpaired}}(\text{b}, \text{S}). \tag{18}$$

The BN% heuristic captures how well complexes form at equilibrium from 1 µM of each component strand. The two components of the numerator account for structural and concentration defect

which, when summed and divided by the target nucleotide concentration, gives the fraction of nucleotides expected to form incorrect base-pairs. Structural defect refers to the nucleotides that form incorrect base-pairs within a complex. However, the expectation is for no less than 1 μM of perfect complexes. Therefore any strands not that do participate in the target complex contribute incorrect nucleotides, called the concentration defect.

The following definitions pertain to the equation below. ϕ_j is the sequence of complex j and s_j is its target base-pairing structure. $n(\phi_j, s_j)$ is NUPACK's estimate for the number of nucleotides in sequence ϕ_j that are not in the state prescribed by complex structure s_j, x_j is the estimated concentration of complex s_j, and y_j is its target concentration.

$$\mathrm{BN\%}(\phi_j, s_j) := \frac{n(\phi_j, s_j)\min(x_j, y_j) + |\phi_j|\max(y_j - x_j, 0)}{y_j|\phi_j|}$$

Piperine evaluates this measure over each complex, including waste and intermediate complexes, and counts as scores the maximum BN% value, the identity of the complex contributing the maximum value (for debugging rather than scoring comparison purposes), and the mean value. We adopted this score to estimate each complexes' propensity to engage in leak pathways that are initiated through remote toeholds in malformed complexes and, generally, how far the designed complexes are from perfect complex formation.

Our heuristic measures capture the tendency of a candidate design to participate in leak reactions or exhibit slowed reaction kinetics due to toehold occlusion. Although experimental evidence suggests that the heuristic measures predict non-ideal behavior well enough to allow comparisons between candidate designs, we do not have a model that relates heuristic scores to absolute levels of non-ideal behavior. Our strategy is to apply the heuristics to select the relative optimum candidate from a set of sequence designs that all fall below a threshold for sequence symmetry and undesirable motif counts. Piperine calculates a rank for each candidate according to each heuristic measure and selects the candidate with the best sum-of-ranks as the winner.

S7 Putting the design pipeline to the test

The design-experiment feedback approach to engineering an oscillating DNA strand displacement system demanded significant effort and time. To convince ourselves that the design principles used to generate that DNA system were general, we attempted to design a second oscillating DNA system using only the principles described in the main paper and no design iterations. Furthermore, we formalized these design principles in a software package that automates the process of compiling abstract CRNs to DNA systems that emulate them, presented above in Sec. S6. This section describes the use of an early version of this compiler to generate a new set of DNA strands implementing the rock-paper-scissors formal reaction network and also summarizes its experimental characterization.

S7.1 Generating and Selecting Candidates

We used a precursor of Piperine to generate and score ten candidate strand sets. This precursor software was not automated, but otherwise had all features described in Sec. S6. These steps were performed identically with a single exception: we included the catalytic helper in strand and complex definitions while designing this sequence set whereas this kind of strand was absent from the sequence design steps while designing Designs 1-4. After generating ten candidates, we decided to select a promising candidate (# 9 in Table S7) to improve by selective mutation of problematic nucleotides. While this design appeared favorable to others, we were concerned about balance in toehold availability. All candidates had large differences between the average and maximum values of TO and SSTU scores, suggesting that kinetic bottlenecks caused by unintended base-pairing in toehold regions may disrupt the intended balance between all reaction pathways.

Using the scoring software, we identified the nucleotides responsible for the TO and SSTU Max values. Keeping all other nucleotides fixed, these positions were re-designed using SpuriousSSM to generate the "9-mut" sequence set. The "mutation" improved the TO and SSTU scores and the design ended up the best sum-of-ranks candidate, which we selected to be Design 5 and ordered for experiments. Critically, the selection by discussion and the directed alteration of winning candidate before purchasing were both manual operations. However, both were necessary to establish how an automated candidate selection should operate and to emphasize that heuristics allow one to detect and correct sequence defects before purchasing DNA.

Table S7: Heuristic scores for sequence design candidates.

Legend: TSI: Top Strand Interactions; TO: Toehold Occlusion; BM: Branch Migration subsequence match; SSU: Single Strand Unpaired; SSTU: Single Strand Toehold Unpaired; TED: Tube Ensemble Defect; WS-Inter-S: Weighted Sum Inter-Strand; WS-Intra-S: Weighted Sum Intra-Strand; ΔG: Average toehold binding energy; ΔG range: Maximum minus minimum toehold binding energies

Design candidate	**TSI avg**	**TSI max**	**TO avg**	**TO max**	**BM sum**	**BM max**	**SSU avg**	**SSU min**	**SSTU avg**	**SSTU min**
1	0.08	0.11	6.33	20.88	8	5	0.94	0.41	0.91	0.42
2	0.11	0.19	5.84	17.75	10	6	0.93	0.17	0.90	0.23
3	0.11	0.20	5.76	20.06	6	5	0.93	0.43	0.90	0.43
4	0.12	0.19	11.95	25.37	6	5	0.94	0.35	0.90	0.35
5	0.10	0.17	7.68	28.96	3	5	0.95	0.34	0.93	0.34
6	0.13	0.20	6.76	21.20	7	5	0.94	0.55	0.92	0.55
7	0.11	0.18	4.10	9.92	8	5	0.94	0.51	0.92	0.51
8	0.10	0.15	9.32	25.56	12	5	0.92	0.30	0.89	0.36
9	0.10	0.16	5.43	19.97	7	5	0.95	0.57	0.92	0.57
9-mut	0.07	0.11	4.70	9.77	14	6	0.97	0.66	0.95	0.67
10	0.08	0.15	21.71	37.90	9	6	0.94	0.35	0.90	0.35

Design candidate	**TED avg**	**TED max**	**WS Inter-S** ($\times 10^3$)	**WS Intra-S** ($\times 10^6$)	**WS Inter-S-1**	**WS Intra-S-1** ($\times 10^6$)	**Verboten**	**ΔG**	**ΔG range**	**Rank sum**
1	0.11	0.040	4.44	7.30	17.93	1.69	1070	7.76	0.42	118.5
2	0.07	0.039	4.35	7.30	10.00	3.38	862	7.66	0.31	128.5
3	0.10	0.042	4.41	7.30	17.59	3.38	34.3	7.66	0.31	118.5
4	0.11	0.040	5.68	10.3	14.66	2.55	864	7.66	0.11	141.5
5	0.06	0.033	6.21	5.20	22.41	4.22	34.2	7.69	0.11	112.5
6	0.09	0.041	4.50	8.57	19.31	3.38	35.1	7.70	0.72	138.5
7	0.11	0.037	4.40	8.99	9.14	0.006	36.3	7.67	0.31	101.5
8	0.08	0.042	4.44	8.57	11.55	1.70	35.2	7.69	0.16	143
9	0.10	0.037	4.41	7.30	20.69	1.69	1070	7.77	0.42	106.5
9-mut	0.06	0.032	4.71	7.32	5.86	2.55	24.3	7.77	0.42	95
10	0.10	0.038	4.43	5.19	3.97	3.38	33.3	7.68	0.15	116

S7.2 Quantifying Leak for Design 5

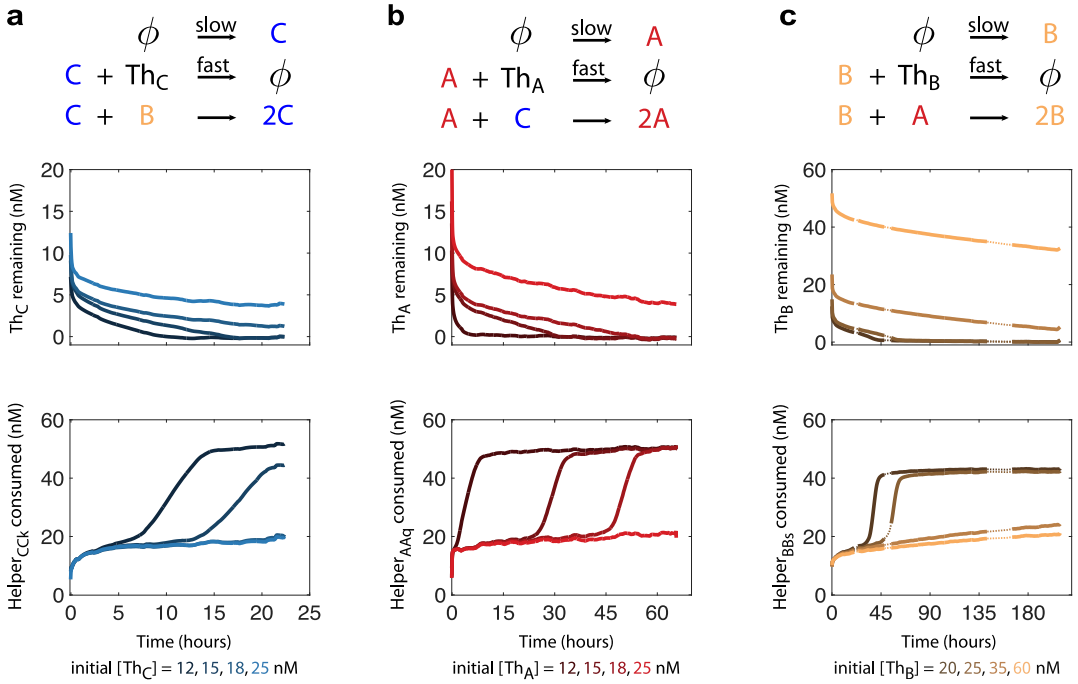

Figure S29: Columns **a**, **b**, **c** show data from experiments on three different DNA implementations of an autocatalytic CRN from Design 5. At the top of each column is a formal CRN composed of, in descending order, the gradual leak, thresholding, and the formal autocatalytic reaction being implemented. The first row of plots show concentrations of Threshold complexes. Threshold complexes sequester the autocatalyst, delaying the onset of the exponential phase. Higher initial concentrations of Threshold, then, cause longer delays. The second row shows the consumption of Helper strands. Initial concentrations of Helper, Produce, and React species are all 100 nM, while the second input is 50 nM. Thus, ideal and leakless reactions would terminate with Helper completion at 50 nM. Missing data was linearly interpolated before smoothing and is represented by dotted line segments in the plots.

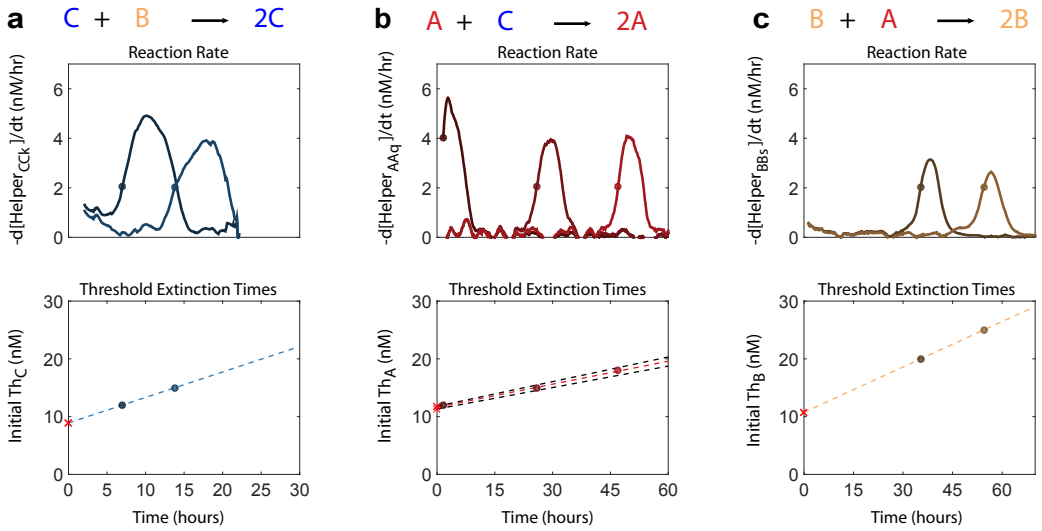

Figure S30: Leak estimation from Design 5 autocatalyst reactions. The analysis performed here is identical to that described in the caption of Fig. S14, except that the threshold for Helper Consumption Rate chosen to indicate the moment when Threshold complexes have all reacted is 2 nM/hr instead of 1 nM/hr. This adjustment was made because, as discussed in Sec. S7.3, the Design 5 data were much noisier than previous experiments.

Module	Initial leak (nM)	Gradual leak velocity (nM/hr)	Gradual leak rate constant (/M/s)
B + A → 2B	10.74	0.26	7.22
C + B → 2C	8.91	0.44	12.22
A + C → 2A	11.64	0.13	3.61

Table S8: Leak parameters derived from autocatalyst experiments using Design 5.

S7.3 A note on normalization

As described in Sec. S7.2, we characterize the leak pathways by investigating each reaction separately as autocatalytic reactions. There are two fluorescent species in these experiments: Helper strands and strands released from Threshold complexes. In analyzing data from experiments on the Designs 1-4, fluorescence readouts were normalized relative to their minimum and maximum values. Both the Helper and Threshold fluorescence timecourse data are monotonic, being either exclusively absorbed or released during an experiment. Maximum fluorescence, then, corresponds to the total concentration of a species added to a sample. Zero-concentration levels are set by adding fluorescence quenching molecules or by measuring pre-reaction, quenched fluorescence. However, the lamp noise during experiments on the Design 5 set was worse than the earlier experiments and in some cases maximum fluorescence levels were not recorded long enough to find an average accurate enough for Maximum-Minimum normalization.

A new normalization approach, called Post-Produce Normalization, was developed to compensate for the noisy data. This approach assumes that all experimental samples of an autocatalytic reaction experience the same amount of initial leak, as we observed in data from experiments on Design 4 (Sec. S3.6). We first normalize the data such that their traces overlap immediately following the addition of Produce complexes and overlap once again when all Helper fluorescence is quenched. Then, we assign a concentration of 0 nM to each samples' minimum value before uniformly scaling their maximum values such that the average of all concentration data before the addition of Produces complexes is 100 nM. See Fig. S31 for a side-by-side comparison of Post-Produce and Maximum-Minimum Normalization.

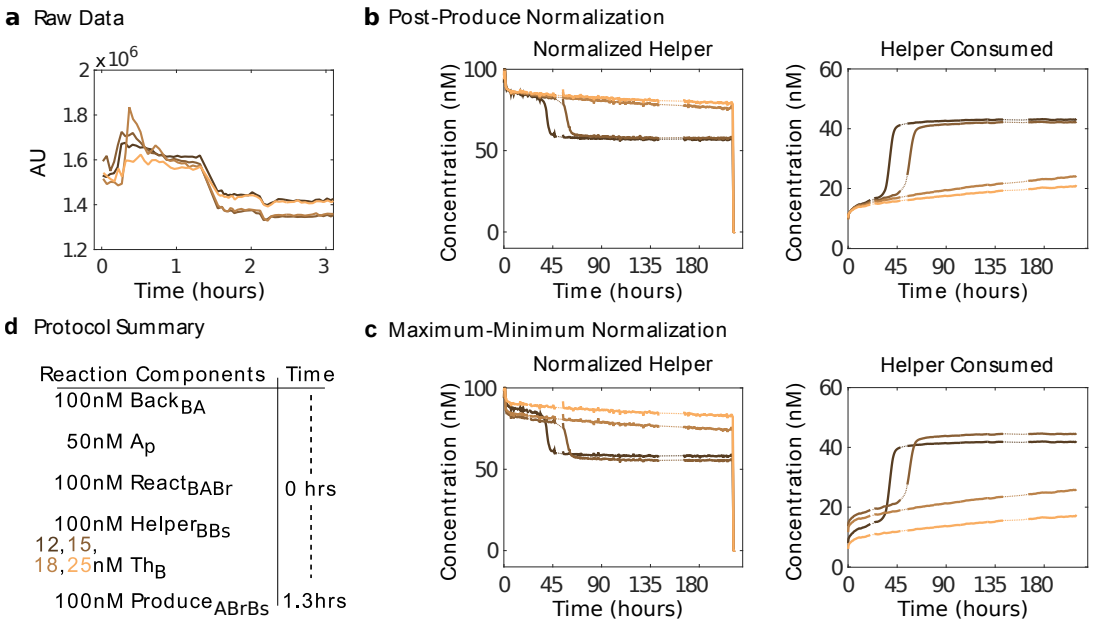

Figure S31: A comparison of the two normalization methods using Module $\mathrm{B} + \mathrm{A} \rightarrow \mathrm{B} + \mathrm{B}$ from Design 5 as an example. Subplot **a** is the raw fluorescence data from the four samples with initial conditions shown in subplot **d**. Plot **a** presents only the first few hours of data so that the effect of lamp noise is clear. Subplots **b** and **c** show the results from using Post-Produce or Maximum-Minimum normalization methods, respectively.

S7.4 Oscillator experiments using Design 5

Data from every experiment of the Design 5 Displacillator showed oscillatory behavior, validating Piperine as a tool for sequence design of dynamical strand displacement cascades. The Rate of Helper Consumption plots shown in Fig. S33 show that the reaction produces signal strands in the same $\mathrm{A} \rightarrow \mathrm{B} \rightarrow \mathrm{C}$ cycle as the formal CRN. However, the significant variation between initial conditions of the different Design 5 oscillator experiments produced a diverse set of trajectories that, in some cases, contradict the expected trajectory. In each sample, we aimed to establish an initial condition with two signal strands at high concentration and one signal strand at low concentration such that only one individual autocatalytic reaction would be active at the onset of the reaction. Where samples **a,b,d,e,f** are those where the fastest initial reaction is clear in Fig. S33, **b,d,f** are in agreement with expectations based on the initial conditions. This indicates that the approaches we took to set an initial condition for the Displacillator reaction were inaccurate, especially for initial conditions with low signal concentrations.

Theoretically, the initial species concentrations of a formal CRN determine its trajectory through phase space. In the DNA dynamical systems considered here, a reaction's initial condition is set by two experimenter-controlled parameters (the concentrations of Threshold complexes and signal strands added to the test tube) and one uncontrollable parameter (concentrations of signal strands released through leak reactions). Using the estimates of initial and gradual leak derived from the autocatalyst experiments described in Sec. S7.2, we attempt to account for leak by first including threshold complexes at concentrations greater than the estimated leak contribution and then adding signal strands to simultaneously trigger the remaining threshold complexes and kickstart oscillations at a known, non-zero initial condition. Fig. S32 explains these calculations for the six rock-paper-scissors oscillator experiments performed using Design 5. Unless specified, experimental parameters are identical to those described in the main text. The results corresponding to the experimental parameters in Fig. S32 are shown in Fig. S33.

a Initial: (Ap,Br,Cj) = (8.8, 7, 2.7)nM

	A	B	C
Threshold added (nM)	13	16	12
Signal leaked (nM)	10.8	13	11.7
Signal added (nM)	11	10	3

b Initial: (Ap,Br,Cj) = (-0.3, 10, 13)nM

	A	B	C
Threshold added (nM)	13	16	12
Signal leaked (nM)	12.7	16	12
Signal added (nM)	0	10	13

c Initial: (Ap,Br,Cj) = (3.4, 2, -3.7)nM

	A	B	C
Threshold added (nM)	13	16	12
Signal leaked (nM)	3.4	5	5.3
Signal added (nM)	13	13	3

d Initial: (Ap,Br,Cj) = (3.7,-7.9, 6.9)nM

	A	B	C
Threshold added (nM)	13	16	12
Signal leaked (nM)	3.7	5.1	5.9
Signal added (nM)	13	3	13

e Initial: (Ap,Br,Cj) = (-0.3, 3.8, 5.4)nM

	A	B	C
Threshold added (nM)	10	10	10
Signal leaked (nM)	3.7	3.8	5.4
Signal added (nM)	6	10	10

f Initial: (Ap,Br,Cj) = (3.7,-0.1, 5.7)nM

	A	B	C
Threshold added (nM)	10	10	10
Signal leaked (nM)	3.7	3.9	5.7
Signal added (nM)	10	6	10

Figure S32: Tables **a**-**f** list the experimental values that determined the initial signal strand concentrations for each execution of the full Design 5 oscillator shown in Fig. S33, respecting figure labels. "Threshold added" values are the concentrations of each signal's threshold complex prior to the addition to Produce complexes. Following the manual addition of Produce complexes, both initial and gradual leak pathways are active and introduce signal strands that react with Threshold complexes. "Signal leaked" values are the concentrations of reacted threshold complexes, measured directly through their fluorescent output strands, at the moment before the manual addition of signal strands. "Signal added" values are the concentration of each signal strand manually added to the reaction, intended to exhaust all remaining threshold complexes and set the initial condition of the chemical oscillations. The initial species concentrations, then, are the sum of leaked and added signal strand conentrations minus the initial threshold concentrations, where negative values indicate surplus threshold complexes. Experiment pairs **a** & **b**, **c** & **d**, and **e** & **f** were each prepared and executed simultaneously and each pair was executed on different days. The experimental chronology matches the alphabetical order of the labels used.

For the first four experiments (**a-d** of Fig. S32 and Fig. S33), threshold complexes were added to the preliminary reaction mixtures at concentrations higher than those expected from the initial and gradual leaks stemming from the individual autocatalytic reactions. While the first experiments conducted (pair **a-b**) exhibited initial leaks as expected, leak dropped significantly in the following experimental pair **c-d**. Most likely this is due to faster preparation of the later experiment pairs and thus less gradual leak accumulation ahead of the first fluorescence readings. Overestimation of leak in experimental pair **c-d** lead to a large surplus of threshold complexes after manual addition of signal strands and, as a result, a long delay before onset of oscillatory behavior in both experimental samples.

Having observed that initial leak estimates based on individual autocatalytic experiments may not predict initial leaks in the rock-paper-scissorss reactions, the concentrations of signal strands added to experimental samples **e-f** were adjusted as data were collected, rather than prior to the experiment, in order to more accurately establish the desired initial signal strand concentrations. From the raw threshold fluoresence data we estimated the concentration of leaked signal strands, observing whether the leak appeared more similar to samples **a-b** or **c-d**, and then tuned the concentration of added signal strands accordingly. This approach allowed us to reduce surplus threshold concentrations in experimental pair **e-f**.

Unlike Design 4, we did not determine the overall kinetic rates and yield of each autocatalytic reaction component of Design 5. Therefore these data are not amenable to the same comparisons to model simulations. There are trends, however, that stand out. The observed peak order matches that of the theoretical system, $\mathrm{A} \rightarrow \mathrm{B} \rightarrow \mathrm{C} \rightarrow \mathrm{A}$. Larger initial concentrations yield higher reaction velocities (e.g. **b** compared to **a**). Helper consumption appears to accelerate throughout all trials except **b**. This may indicate that concentration of Catalytic Helper strands used in these experiments had adjusted the stoichiometry of the autocatalytic reactions above the intended ratio of two output strands for two input strands and lead to a net increase, rather than conservation, of total signal strand concentration in solution as the experiment progressed.

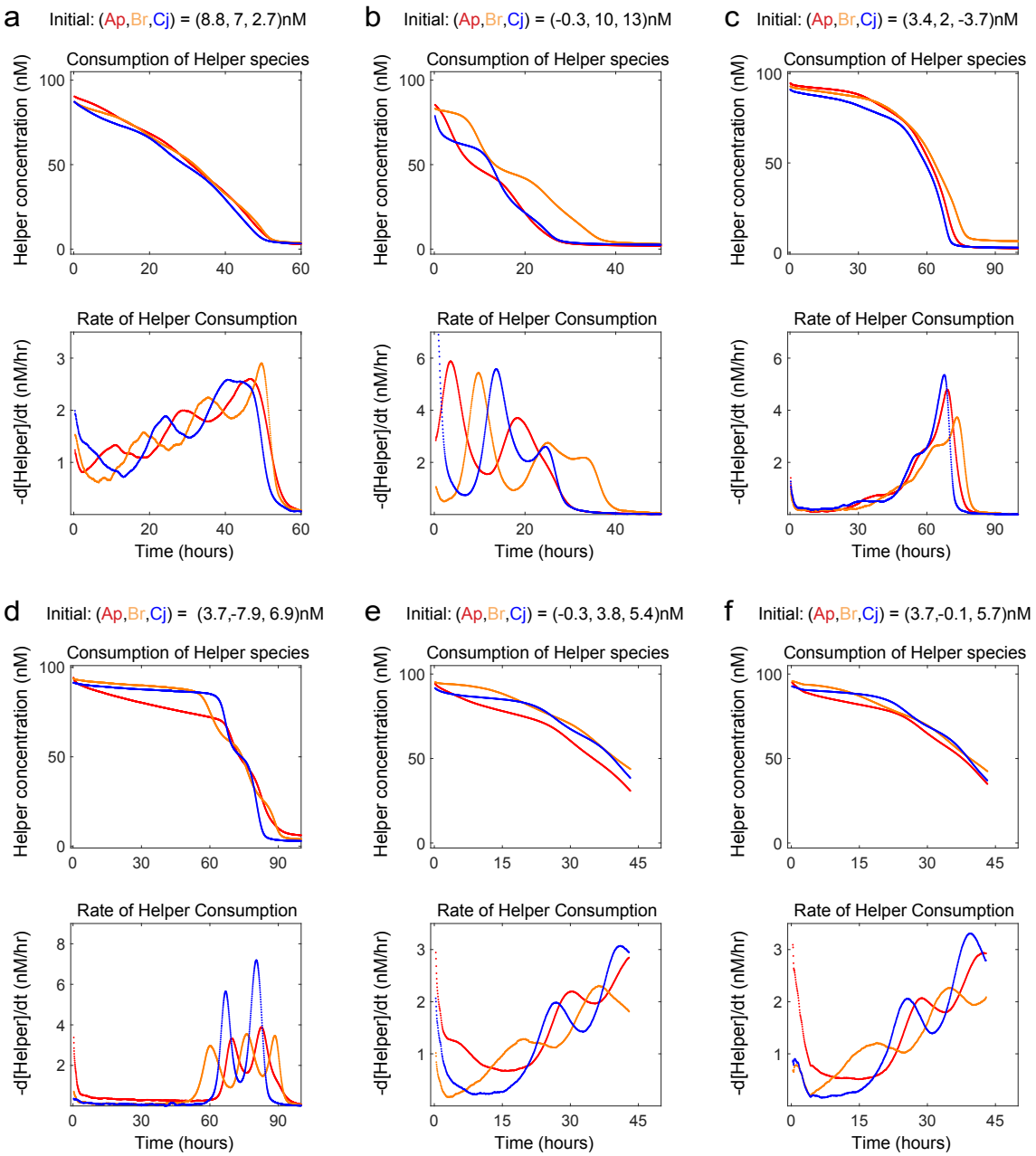

Figure S33: **a**-**f** show Helper concentrations and consumption rates along with initial signal strand concentrations for six different runs of the Design 5 Displacillator. Fig. S32 shows the calculations that yield the initial conditions listed here. Negative values occur when there are expected to be surplus Threshold complexes in solution at time $t = 0$.

S8 Materials and Methods

DNA oligonucleotides. All DNA oligonucleotides used in this study were purchased from Integrated DNA Technologies (IDT). Oligonucleotides of length less than 60 bases were ordered with HPLC purification, while those 60 bases or longer were ordered with IE-HPLC purification. Where applicable, fluorophores and quenchers were attached by IDT as well.

Buffer conditions. All DNA oligonucleotides were stored at 4 °C in TE buffer (10 mM Tris.HCl pH balanced to 8.0, with 1 mM EDTA.Na_2, purchased as 100x stock from Sigma-Aldrich). We define two buffer conditions. First, "TE/Mg^{++}", which was prepared by adding TE buffer containing 62.5 mM $MgCl_2$ in a ratio of 1:4 to the sample, thereby achieving a final $MgCl_2$ concentration of 12.5 mM, out of which 1 mM is bound to EDTA. Second, "TE/Na^{+}", which was prepared by adding the appropriate quantity of dry NaCl salt (purchased 99% pure from EM Science, lot number 43076317) to a given volume of TE buffer to achieve a final NaCl concentration of 0.5 M. All buffer solutions were pH adjusted to 8.0 and filtered with a 1 micron filter (Nalgene rapid-flow).

All spectrofluorimetry experiments with Designs 3 and 4 reported here were performed in TE/Na^{+} buffer. Spectrofluorimetry experiments with earlier Designs 1 and 2 were performed in TE/Mg^{++} buffer.

Quantitation of single strands. Single strands were quantitated from absorbance at 260 nm (measured using a Thermo Scientific NanoDrop cuvette-free spectrophotometer) using calculated extinction coefficients (29). After thorough vortexing to ensure homogeneity in concentration, 3 samples of 2 μL each were typically used to measure absorbance. Two readings were taken from each sample and all data points were averaged. Typically, readings were within 2-5% of each other.

Annealing protocol. All annealing steps in this study were identical and were performed with an Eppendorf Mastercycler Gradient thermocycler. The samples were first heated up to 95 °C and then slowly cooled to 20 °C at the constant rate of 1 °C/min.

Annealing Reporters. All reporter complexes were annealed with a 20% excess of top strand (which is labeled with quencher in each case). For experiments with Designs 3 and 4, reporters were annealed in TE/Na^{+} buffer. For experiments with Designs 1 and 2, reporters were annealed in TE/Mg^{++} buffer.

Reporter complexes were not gel purified after annealing. Reporter complexes were annealed to have a bottom strand concentration of 5 μM, which also determines nominal concentration of reporter since the top strand was added in excess. This procedure was chosen because of two reasons. First, accurate quantification of bottom strands leads to accurate estimates of the concentration of reporter complex. This is important since the total concentration of reporter complex is used for normalization of spectrofluorimetry data. In contrast, quantifying the concentration of multistranded complexes is relatively less accurate because of larger errors in estimating extinction coefficients. Second, since reporter top strands have no toehold domains and are modified with quenchers, the excess addition of top strands ensures that all bottom strands form complexes, even with somewhat imperfect stoichiometry. This mitigates the chances of any active single-stranded DNA being present and ensures a stable fluorescence baseline.

Annealing and purification of multistranded fuel complexes. Each multistranded fuel complex (React and Produce) was prepared as follows. First, 5 nanomoles of the bottom strand was annealed with 6 nanomoles (20% excess) of each of the top strands in TE/Mg^{++} buffer (at an approximate concentration of 25 μM).

After annealing, complexes were purified by nondenaturing (ND) polyacrylamide gel elec-

trophoresis (PAGE) by running the samples on 12% gel at 150 V for approximately 6 hours. The purpose of the purification was (i) to remove the excess top strands that were added and (ii) remove multimers of the desired complexes that form due to the high concentrations in the annealing step. The acrylamide (19:1 acrylamide:bis) was diluted from 40% acrylamide stock purchased from Ambion. ND loading dye containing xylene cyanol FF in 50% glycerol was added to the React complexes, resulting in a final glycerol concentration of 10% by volume. Since the Produce complexes contain a quencher, 50% glycerol was added directly (rather than the ND loading dye) to achieve a final gycerol concentration of 10% by volume.

For Designs 3 and 4, the appropriate bands were cut out and eluted in 1 mL of TE/Na^+ buffer for 18-24 hours. For Designs 1 and 2, the elution was done in TE/Mg^{++} buffer instead.

Dialysis of multistranded fuel complexes. For Designs 3 and 4, since the experiments were to be performed in TE/Na^+ buffer, a further reduction (approximately 2500 fold) in Mg^{++} concentration was achieved using 2 rounds of dialysis. Each round of dialysis is expected to achieve a reduction of approximately 50 fold, since 1 ml of purified multistranded fuel complex was dialyzed with approximately 50 ml of TE/Na^+ buffer for 2 hours using a 2 ml Thermo Scientific Slide-A-Lyzer MINI dialysis device with a 10K MWCO membrane.

Quantitation of multistranded fuel complexes. The procedure for quantitating multistranded fuel complexes is essentially identical to the procedure for single strands, except for the calculation of extinction coefficients, which involves corrections for hyperchromicity (29). We expect the inferred concentrations to be less accurate because of larger uncertainties in the estimated extinction coefficients. Typical yields after purification ranged from 40% to 60%.

Experimental protocols for spectrofluorimetry. Spectrufluorimetry experiments were performed using a commercial SPEX Fluorolog-3 from Horiba, equipped with a water bath temperature controller. All spectrofluorimetry experiments were performed at 25 °C unless otherwise mentioned. Synthetic round-top quartz cuvettes (119-004F) from Hellma, with a total volume 1.6 ml, were used.

Prior to each experiment, all cuvettes were cleaned thoroughly. Each cuvette was washed 10 times in Milli-Q water, twice in 70% ethanol, and finally another 5 times in Milli-Q water. After washing, cuvettes were dried by gently tapping them on a Kimtech Science wipe placed on paper towels for cushioning. They were subsequently left to air-dry for about one hour. Cuvette caps were washed once thoroughly with Milli-Q water, once with 70% ethanol, and once again with Milli-Q water. They were then dried with Kimtech Science wipe and left to air-dry for an hour. After adding the sample, the exterior of the cuvette was washed with the same procedure as the caps.

For experiments involving the ROX fluorophore, excitation was at 584 nm, while emissions were at 602 nm. For experiments involving the Alexa-488 fluorophore, excitation was at 492 nm, while emissions were at 517 nm. For experiments involving the Alexa-647 fluorophore, excitation was at 650 nm, while emissions were at 670 nm. Band pass value of 2 nm was used for both excitation and emission monochrometers for all experiments except those measuring individual strand displacement and toehold exchange rate constants, for which a bandpass of 4 nm was used. All experiments were done with integration time of 10 seconds for each data point.

For experiments involving one fluorophore, measurements were taken every minute; for those with multiple fluorophores, measurement interval increased proportionally because each excitation/emission channel was allotted 1 minute for measurement.

Dilution correction. Spectrofluorimetry experiments involving the autocatalytic modules (e.g. Fig. 2 of the main text, Figs. S11, S12) and the Displacillator (e.g. Fig. S23) require the addition of Pro-

duce complexes while fluorescence data is being recorded. The volume of liquid added along with the Produce complexes is usually not negligible; in Displacillator experiments, it can be as high as 8-10% of the total volume of the reaction. This volume addition would reduce the concentration of all the fluorophores in solution, and thereby lead to a decrease in fluorescence signal that is purely due to dilution. To compensate for this effect, we introduce a dilution correction. If the volume of Produce complexes added is v_P, and the total volume of the reaction before addition of the Produce complexes was v_1, then we multiply all fluorescence counts after the addition by $(1 + \frac{v_P}{v_1})$.

S8.1 Normalization details for spectrofluorimetry.

Here we describe our procedure for normalizing fluorescence data (in AU) to get information about concentrations of the respective species (in nM). Fig. S34 describes the procedure for an experiment with an autocatalytic module; Fig. S35 illustrates the procedure for a Displacillator experiment.

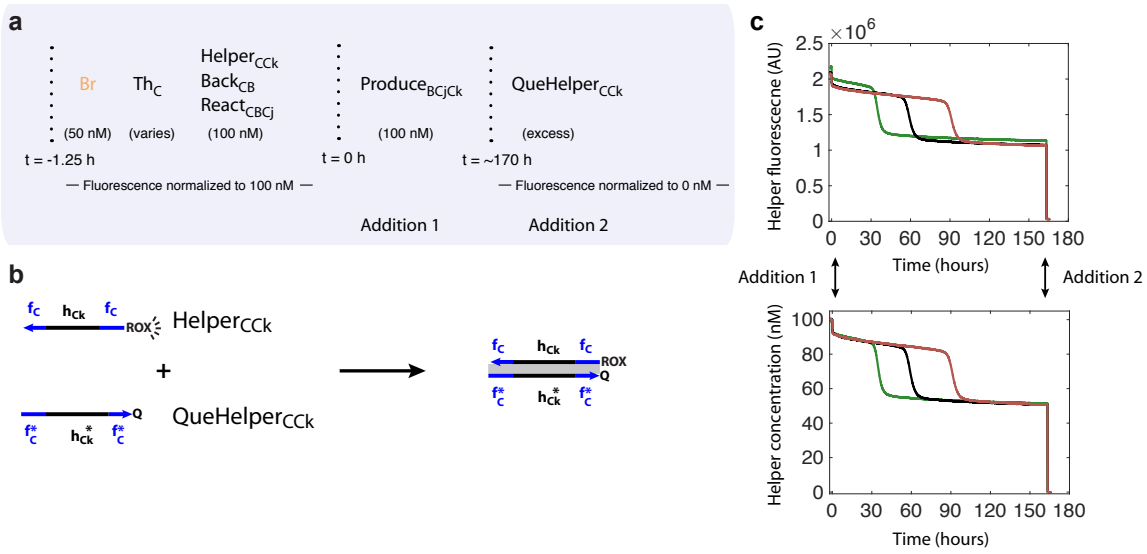

Figure S34: a. Experimental setup for the autocatalytic module C + B → 2C (Design 4). Median fluorescence of the Helper strand measured for about 1.25 h (at least 25 data points) before the addition of the Produce complex was normalized to 100 nM. After the exponential amplification completes, excess QueHelper strand was added around 170 h in order to quench fluorescence of free Helper strands that remain in solution. After the quenching proceeds to completion, the median fluorescence (measured for at least 25 data points) was normalized to 0. b. Mechanism of quenching by the QueHelper strands, which are essentially complements of the Helper strands with a quencher attached. c. Smoothed raw data and normalized data, for comparison.

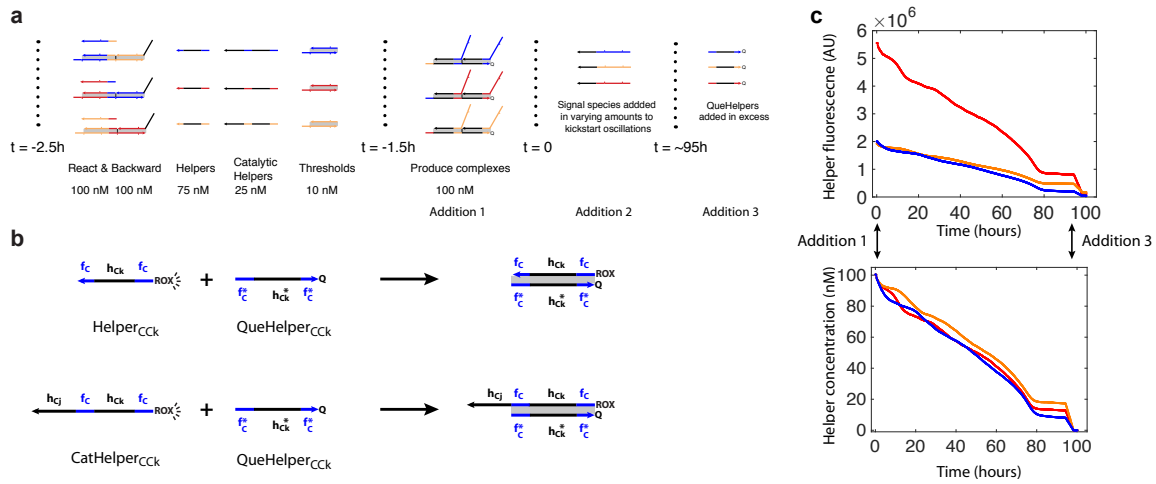

Figure S35: a. Procedure for a Displacillator experiment (data shown from Design 4). Median fluorescence of the Helper and CatHelper strands measured for about 1.25h (at least 25 data points) before the addition of the Produce complexes were normalized to 100 nM. After the fuel species are consumed, excess QueHelper strands were added around 95h in order to quench the fluorescence of free Helper and CatHelper strands that remain in solution. After the quenching proceeds to completion, the median fluorescence (measured for at least 25 data points) were normalized to 0. b. Mechanism of quenching of Helper and CatHelper strands by the QueHelper strands, shown for $\mathrm{Helper}_{\mathrm{CCk}}$ and $\mathrm{CatHelper}_{\mathrm{CCk}}$ as an example. c. Smoothed raw data and normalized data, for comparison.

Threshold	Measured rate constant (/M /s)
$k_{\mathrm{RepFluxCAp}}$	7.7×10^5
$k_{\mathrm{RepFluxBCj}}$	3.1×10^6
$k_{\mathrm{RepFluxABr}}$	2.2×10^6

Table S9: Independently measured rate constants (all in /M /s) for the consumption of FluxCAp, FluxBCj, and FluxABr by the corresponding reporters (Design 4).

S8.2 Characterizing individual rate constants

All experiments for measuring individual rate constants were performed at low concentrations, where the toehold occlusion effect will be negligible.

We emphasize that all data in Sec. S8.2 was normalized to the total concentration of the Reporter complex. This is good practice for two reasons. First, the Reporter complexes were annealed with an excess of top strand and were not gel purified. Therefore, their concentration is inferred directly from the concentration of the bottom strand. Since extinction coefficient calculations are more accurate for single strands, we believe that concentrations of the Reporter complexes are in general more accurately measured compared to fuel complexes. (Since fuel complexes are gel purified, their concentration estimates rely on extinction coefficients calculated for multi-stranded complexes, which are prone to larger errors).

S8.2.1 Measuring reporter rate constants

Here we describe how rate constants for triggering reporter complexes (which also act as Thresholds for consuming initial leak of signal strands) were measured. Fig. S36 summarizes the experimental and analysis procedure for measuring k_{RepA}. Similar experiments were performed for measuring k_{RepB} and k_{RepC}. All these individual rate constants are summarized in Tab. S2. Measurements similar to these were performed for reporters for the Flux strands (see Fig. S18). Measured rate constants for $\mathrm{RepFlux_{CAp}}$, $\mathrm{RepFlux_{BCj}}$, and $\mathrm{RepFlux_{ABr}}$ are summarized in Tab. S9.

S8.2.2 Measuring rate constants for desired strand displacement reactions

Tab. S1 summarized the independently measured rate constants for all desired strand displacement reactions. In this section we describe the experimental and analysis procedures with $\mathrm{k_{fwd1}^{CApAq}}$ and $\mathrm{k_{back}^{CApAq}}$ as illustrative examples.

Fig. S37 presents the experimental setup and data for measuring $\mathrm{k_{fwd1}^{CApAq}}$ and $\mathrm{k_{back}^{CApAq}}$. We note that exact amounts of each addition were inferred after normalization (e.g. the amounts of $\mathrm{Flux_{CAp}}$ added to Samples 1, 2, and 3 in panels b and c were inferred to be 2.2, 3.1, and 3.9 nM, respectively). The following model (say model A) was used to estimate $\mathrm{k_{fwd1}^{CApAq}}$:

$$\mathrm{Flux_{CAp}} + \mathrm{Produce_{CApAq}} \underset{\mathrm{k_{back}}^{CApAq}}{\overset{\mathrm{k_{fwd1}}^{CApAq}}{\longleftrightarrow}} \mathrm{ProduceInt_{CApAq}} + \mathrm{Ap} \tag{19}$$

$$\mathrm{Ap} + \mathrm{Rep\,A} \xrightarrow{\mathrm{k_{RepA}}} \mathrm{Fluorescence}. \tag{20}$$

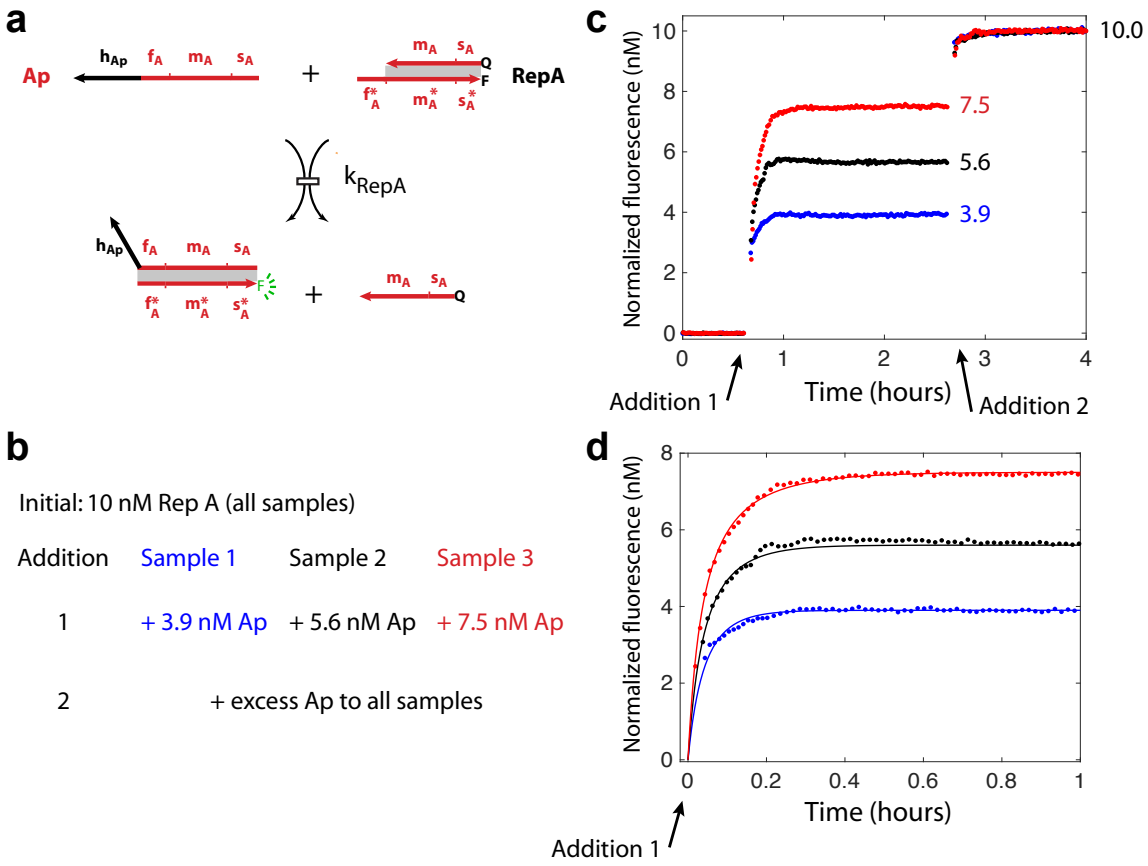

Figure S36: a. Mechanism by which Ap triggers Rep A. b. Experimental setup. c. Normalized fluorescence data (Design 4). Median baseline fluorescence of Rep A before addition of Ap was normalized to 0; median fluorescence after addition of excess Ap was normalized to the initial concentration of Rep A, which was 10 nM. The exact amounts of Ap added in Addition 1 were backed out from this normalization (target additions were 4, 6, and 8 nM respectively). d. By fitting Ap + RepA $\xrightarrow{k_{\mathrm{RepA}}}$ Fluorescence to all three curves simultaneously, we obtained $k_{\mathrm{RepA}} = 7.4 \times 10^5$ /M/s.

Similarly, the model below (model B) was used to estimate $\mathrm{k}_{\mathrm{back}}^{\mathrm{CApAq}}$:

$$\mathrm{Flux_{CAp}} + \mathrm{Produce_{CApAq}} \underset{\mathrm{k_{back}}^{\mathrm{CA}p\mathrm{Aq}}}{\overset{\mathrm{k_{fwd1}}^{\mathrm{CA}p\mathrm{Aq}}}{\longleftrightarrow}} \mathrm{ProduceInt_{CApAq}} + \mathrm{Ap} \tag{21}$$

$$\mathrm{Flux_{CAp}} + \mathrm{Rep} \xrightarrow{\mathrm{k_{RepFlux_{CAp}}}} \mathrm{Fluorescence}. \tag{22}$$

Since k_{RepA} and $k_{\mathrm{RepFlux_{CAp}}}$ were measured independently (Tab. S2 and S9 respectively), the only unknown parameters were $\mathrm{k}_{\mathrm{fwd1}}^{\mathrm{CApAq}}$ and $\mathrm{k}_{\mathrm{back}}^{\mathrm{CApAq}}$. These were estimated as follows.

First, the fluorescence curves predicted by model A with $\mathrm{k}_{\mathrm{back}}^{\mathrm{CApAq}} = 0$ were fit to the data in Fig. S37d. This generated an initial estimate for $\mathrm{k}_{\mathrm{fwd1}}^{\mathrm{CApAq}}$. Similarly, the fluorescence curves predicted by model B with $\mathrm{k}_{\mathrm{fwd1}}^{\mathrm{CApAq}} = 0$ were fit to the data in Fig. S37g to generate an initial estimate of $\mathrm{k}_{\mathrm{back}}^{\mathrm{CApAq}}$. Then, model A was fit again to the data in Fig. S37d, with $\mathrm{k}_{\mathrm{back}}^{\mathrm{CApAq}}$ set to its initial estimate, to generate an updated estimate for $\mathrm{k}_{\mathrm{fwd1}}^{\mathrm{CApAq}}$. This new estimate of $\mathrm{k}_{\mathrm{fwd1}}^{\mathrm{CApAq}}$ was used to re-fit model B to the data in Fig. S37g to generate an updated estimate for $\mathrm{k}_{\mathrm{back}}^{\mathrm{CApAq}}$. This procedure was repeated until both estimates changed by less than 10%. The final values are summarized in Tab. S1.

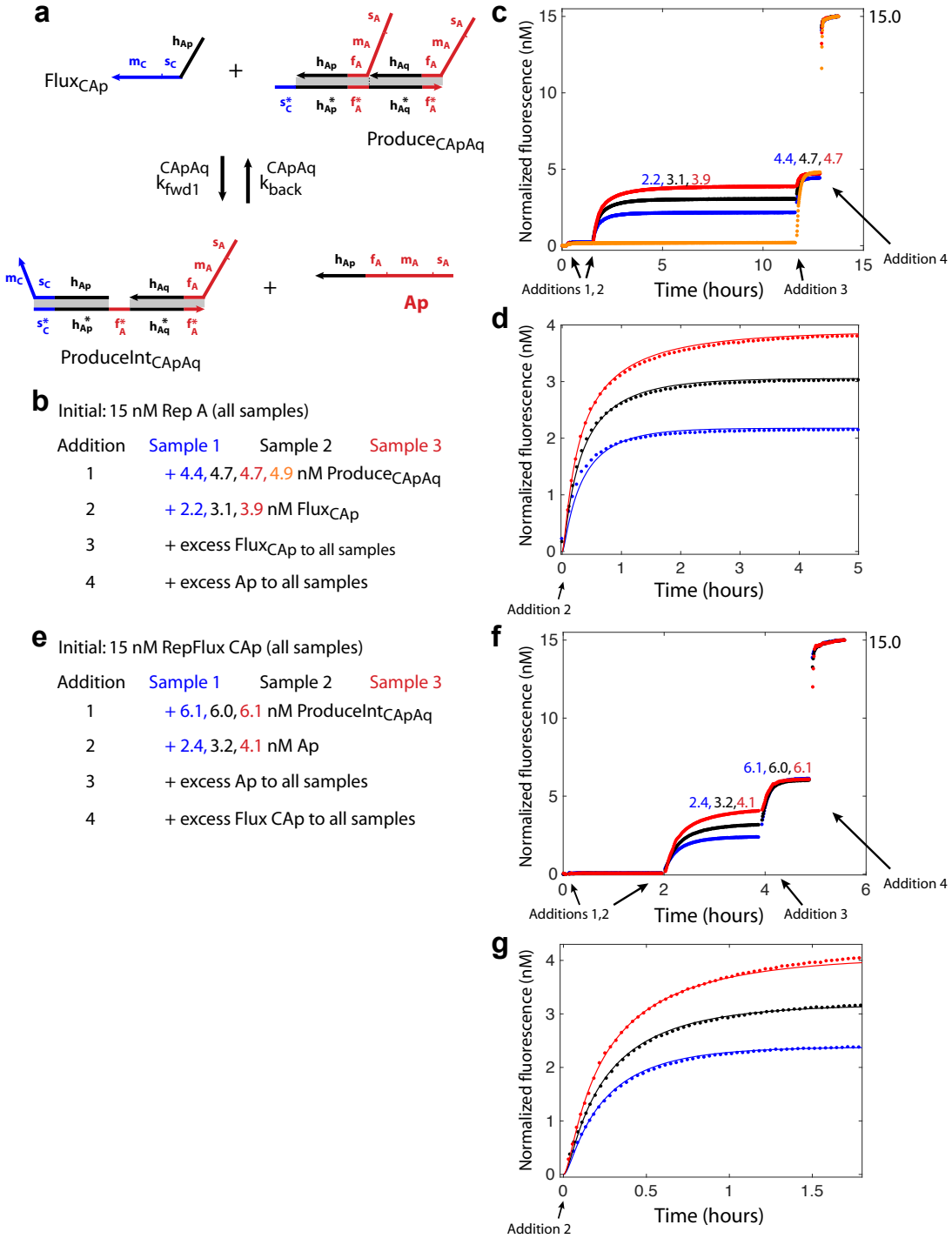

Figure S37: a. Reversible toehold exchange reaction in the Produce step for A + C $\rightarrow$ 2A. b. Experimental setup for measuring k_{fwd1}^{CApAq}. c. Fluorescence data (Design 4). The orange curve represents Sample 4, which is an un-triggered control. Median baseline fluorescence of Rep A is normalized to 0; median fluorescence after Addition 4 is normalized to initial concentration of Rep A (15 nM). d. Model fits as described in Sec. S8.2.2. e. Experimental setup for measuring k_{back}^{CApAq}. f. Fluorescence data. Median baseline fluorescence of RepFlux CAp is normalized to 0; median fluorescence after Addition 4 is normalized to the initial concentration of RepFlux CAp (15 nM). g. Model fits, as described in Sec. S8.2.2.

S8.3 Sequences from Designs 1, 2, 3, 4, and 5

DNA sequences used in this study are provided below.

Strand name	Sequence
D1_React_BOT_CBCj	AGTGGGTTAGTAGAGAGTTGTTAGTGGGAAATGGGAATGTTGTGAGGAATGAGAGGGTAT
D1_Back_CB	CTCATTCCTCACAACATTCCCATTTCCCA
D1_Flux_BCj	CTAACAACTCTCTACTAACCCACTTCATACCTTATCC
D1_Produce_BOT_BCjCk	AGAGGGTATGAAGGTGTAAGAAGGAGGGTATGGATAAGGTATGAAGTGGGTTA
D1_Cj	CTTCATACCTTATCCATACCCTCTCATTCCTCACAACATTCCCA
D1_Ck	CCTTCTTACACCTTCATACCCTCTCATTCCTCACAACATTCCCA
D1_Helper_CCk	ATACCCTCCTTCTTACACCTTCATACCCT
D1_React_BOT_BABr	TGAGGGTTAGAGGTTTGAAGAGTGAGGGATTTGGGTTAGTAGAGAGTTGTTAGTGGGAAA
D1_Back_BA	CTAACAACTCTCTACTAACCCAAATCCCT
D1_Flux_ABr	CACTCTTCAAACCTCTAACCCTCATTCAAATCTCACC
D1_Produce_BOT_ABrBs	AGTGGGAAAGGAGAGAATGAATGGTGGGAAAGGTGAGATTTGAATGAGGGTTA
D1_Br	CATTCAAATCTCACCTTTCCCACTAACAACTCTCTACTAACCCA
D1_Bs	CCATTCATTCTCTCCTTTCCCACTAACAACTCTCTACTAACCCA
D1_Helper_BBs	TTTCCCACCATTCATTCTCTCCTTTCCCA
D1_React_BOT_ACAp	TGTGGGAATGTTGTGAGGAATGAGAGGGTATAGGGTTAGAGGTTTGAAGAGTGAGGGATT
D1_Back_AC	CACTCTTCAAACCTCTAACCCTATACCCT
D1_Flux_CAp	CTCATTCCTCACAACATTCCCACACAATACTATCATC
D1_Produce_BOT_CApAq	TGAGGGATTGTGTTTGAGTTTAGGAGGGATTGATGATAGTATTGTGTGGGAAT
D1_Ap	CACAATACTATCATCAATCCCTCACTCTTCAAACCTCTAACCCT
D1_Aq	CCTAAACTCAAACACAATCCCTCACTCTTCAAACCTCTAACCCT
D1_Helper_AAq	AATCCCTCCTAAACTCAAACACAATCCCT

Table S10: DNA sequences from Design 1.

Strand name	Sequence
D2_React_BOT_CBCj	TCGGGTAAAGAGATTGATTGGTGGGATATGGAGAAATGAGGAAGTTGAGAGGCTTGTTGT
D2_Back_CB	GCCTCTCAACTTCCTCATTTCTCCATATC
D2_Flux_BCj	CCACCAATCAATCTCTTTACCCGACACCTCCCTTCTA
D2_Produce_BOT_BCjCk	GCTTGTTGTAGGAGTGTGTTTGCGTTGTTGTTAGAAGGGAGGTGTCGGGTAAA
D2_Cj	GACACCTCCCTTCTAACAACAAGCCTCTCAACTTCCTCATTTCT
D2_Ck	CGCAAACACACTCCTACAACAAGCCTCTCAACTTCCTCATTTCT
D2_Helper_CCk	ACAACAACGCAAACACACTCCTACAACAA
D2_React_BOT_BABr	CCGTAGTGATAGTTAGTATGTACCAAAGGATGGGTAAAGAGATTGATTGGTGGGATATGG
D2_Back_BA	CCACCAATCAATCTCTTTACCCATCCTTT
D2_Flux_ABr	GGTACATACTAACTATCACTACGGCATTACATTCAAA
D2_Produce_BOT_ABrBs	GGGATATGGGAAGAATAGGTTGCCGATATGGTTTGAATGTAATGCCGTAGTGA
D2_Br	GGCATTACATTCAAACCATATCCCACCAATCAATCTCTTTACCC
D2_Bs	GGCAACCTATTCTTCCCATATCCCACCAATCAATCTCTTTACCC
D2_Helper_BBs	CCATATCGGCAACCTATTCTTCCCATATC
D2_React_BOT_ACAp	CCAGAAATGAGGAAGTTGAGAGGCTTGTTGTGTAGTGATAGTTAGTATGTACCAAAGGAT
D2_Back_AC	GGTACATACTAACTATCACTACACAACAA
D2_Flux_CAp	GCCTCTCAACTTCCTCATTTCTGGTCACCACACTTCT
D2_Produce_BOT_CApAq	CCAAAGGATTAGGGTTAGTTGTGGAAAGGATAGAAGTGTGGTGACCAGAAATG
D2_Ap	GGTCACCACACTTCTATCCTTTGGTACATACTAACTATCACTAC
D2_Aq	CCACAACTAACCCTAATCCTTTGGTACATACTAACTATCACTAC
D2_Helper_AAq	ATCCTTTCCACAACTAACCCTAATCCTTT

Table S11: DNA sequences from Design 2.

Strand name	Sequence
D3_React_BOT_CBCj	TGTTGTTCGGAGAGTTGAAGGAGTAAAGAGGTATAGGGTGGTTAGTTAGAGAAATGGGCT
D3_Back_CB	CTATACCTCTTTACTCCTTCAACTCTCCG
D3_Flux_BCj	ATCTTCCCTCCACAGCCCATTTCTCTAACTAACCACC
D3_Produce_BOT_BCjCk	AAATGGGCTGTGGAGGGAAGATTGTTGTTGCGTTTGTGTGAGGATGTTGTTCG
D3_Cj	TCTTTACTCCTTCAACTCTCCGAACAACAATCTTCCCTCCACAG
D3_Ck	TCTTTACTCCTTCAACTCTCCGAACAACATCCTCACACAAACGC
D3_Helper_CCk	AACAACATCCTCACACAAACGCAACAACA
D3_React_BOT_BABr	GGTATAGGGTGGTTAGTTAGAGAAATGGGTAGGAAACCATGTATGATTGATAGTGATGCC
D3_Back_BA	TTTCCTACCCATTTCTCTAACTAACCACC
D3_Flux_ABr	AAACTTACATTACGGCATCACTATCAATCATACATGG
D3_Produce_BOT_ABrBs	AGTGATGCCGTAATGTAAGTTTGGTATAGCCGTTGGATAAGAAGGGTATAGGG
D3_Br	CCCATTTCTCTAACTAACCACCCTATACCAAACTTACATTACGG
D3_Bs	CCCATTTCTCTAACTAACCACCCTATACCCTTCTTATCCAACGG
D3_Helper_BBs	CTATACCCTTCTTATCCAACGGCTATACC
D3_React_BOT_ACAp	TAGGAAACCATGTATGATTGATAGTGATGTGTTGTTCGGAGAGTTGAAGGAGTAAAGACC
D3_Back_AC	AACAACACATCACTATCAATCATACATGG
D3_Flux_CAp	TCTTCACACCACTGGTCTTTACTCCTTCAACTCTCCG
D3_Produce_BOT_CApAq	GTAAAGACCAGTGGTGTGAAGATAGGAAAGGTGTTGATTGGGATTAGGAAACC
D3_Ap	CATCACTATCAATCATACATGGTTTCCTATCTTCACACCACTGG
D3_Aq	CATCACTATCAATCATACATGGTTTCCTAATCCCAATCAACACC
D3_Helper_AAq	TTTCCTAATCCCAATCAACACCTTTCCTA

Table S12: DNA sequences from Design 3.

Strand name	Sequence
D4_React_BOT_CBCj	TGTTGTTTGGAGAGTTGAAGGAGTAAAGAGGTATAGGGTGGTTAGTTAGAGAAATGGGCG
D4_Back_CB	CTATACCTCTTTACTCCTTCAACTCTCCA
D4_Flux_BCj	ATCTTCCCTCCACCGCCCATTTCTCTAACTAACCACC
D4_Produce_BOT_BCjCk	AAATGGGCGGTGGAGGGAAGATTGTTGTTGCGTTTGTGTGAGGATGTTGTTTG/3IAbRQSp/
D4_Cj	TCTTTACTCCTTCAACTCTCCAAACAACAATCTTCCCTCCACCG
D4_Ck	TCTTTACTCCTTCAACTCTCCAAACAACATCCTCACACAAACGC
D4_Helper_CCk	/56-ROXN/AACAACATCCTCACACAAACGCAACAACA
D4_Cat_Helper_CCk	/56-ROXN/AACAACATCCTCACACAAACGCAACAACAATCTTCCCTCCACCG
D4_React_BOT_BABr	GGTATAGGGTGGTTAGTTAGAGAAATGGGTAGGAAAAGATGTAGGATTGATAGTGATGCG
D4_Back_BA	TTTCCTACCCATTTCTCTAACTAACCACC
D4_Flux_ABr	AAACTTACATTACCGCATCACTATCAATCCTACATCT
D4_Produce_BOT_ABrBs	AGTGATGCGGTAATGTAAGTTTGGTATAGCGGTTGGATAAGAAGGGTATAGGG/3IAbRQSp/
D4_Br	CCCATTTCTCTAACTAACCACCCTATACCAAACTTACATTACCG
D4_Bs	CCCATTTCTCTAACTAACCACCCTATACCCTTCTTATCCAACCG
D4_Helper_BBs	/5Alex647N/CTATACCCTTCTTATCCAACCGCTATACC
D4_Cat_Helper_BBs	/5Alex647N/CTATACCCTTCTTATCCAACCGCTATACCAAACTTACATTACCG
D4_React_BOT_ACApi2	TAGGAAAAGATGTAGGATTGATAGTGATGTTGTTTGGAGAGTTGAAGGAGTAAAGAAG
D4_Back_ACi2	AACAACATCACTATCAATCCTACATCT
D4_Flux_CAp	TCTTCACACCACTCTTCTTTACTCCTTCAACTCTCCA
D4_Produce_BOT_CApAq	GTAAAGAAGAGTGGTGTGAAGATAGGAAAGGTGTTGATTGGGATTAGGAAAAG/3IABkFQ/
D4_Ap	CATCACTATCAATCCTACATCTTTTCCTATCTTCACACCACTCT
D4_Aq	CATCACTATCAATCCTACATCTTTTCCTAATCCCAATCAACACC
D4_Helper_AAq	/5Alex488N/TTTCCTAATCCCAATCAACACCTTTCCTA
D4_Cat_Helper_AAq	/5Alex488N/TTTCCTAATCCCAATCAACACCTTTCCTATCTTCACACCACTCT

Table S13: DNA sequences from Design 4.

Strand name	Sequence
D4_Rep_BOT_C	TGTTGTTTGGAGAGTTGAAGGAGTAAAGA/3AlexF488N/
D4_Rep_TOP_B	/5IAbRQ/CCCATTTCTCTAACTAACCACC
D4_Rep_BOT_B	GGTATAGGGTGGTTAGTTAGAGAAATGGG/3Rox_N/
D4_Rep_TOP_A	/5IAbRQ/CATCACTATCAATCCTACATCT
D4_Rep_BOT_A	TAGGAAAAGATGTAGGATTGATAGTGATG/3AlexF647N/
D4_Helper_CCk†	AACAACATCCTCACACAAACGCAACAACA
D4_Cat_Helper_CCk†	AACAACATCCTCACACAAACGCAACAACAATCTTCCCTCCACCG
D4_Helper_BBs†	CTATACCCTTCTTATCCAACCGCTATACC
D4_Cat_Helper_BBs†	CTATACCCTTCTTATCCAACCGCTATACCAAACTTACATTACCG
D4_Helper_AAq†	TTTCCTAATCCCAATCAACACCTTTCCTA
D4_Cat_Helper_AAq	TTTCCTAATCCCAATCAACACCTTTCCTATCTTCACACCACTCT
D4_Rep_TOP_C†	TCTTTACTCCTTCAACTCTCCA
D4_Rep_BOT_C†	TGTTGTTTGGAGAGTTGAAGGAGTAAAGA
D4_Rep_TOP_B†	CCCATTTCTCTAACTAACCACC
D4_Rep_BOT_B†	GGTATAGGGTGGTTAGTTAGAGAAATGGG
D4_Rep_TOP_A†	CATCACTATCAATCCTACATCT
D4_Rep_BOT_A†	TAGGAAAAGATGTAGGATTGATAGTGATG
D4_Rep_TOP_C	/5IABkFQ/TCTTTACTCCTTCAACTCTCCA
D4_QUE_Helper_CCk	TGTTGTTGCGTTTGTGTGAGGATGTTGTTTG/3IAbRQSp/
D4_QUE_Helper_BBs	GGTATAGCGGTTGGATAAGAAGGGTATAGGG/3IAbRQSp/
D4_QUE_Helper_AAq	TAGGAAAGGTGTTGATTGGGATTAGGAAAAG/3IABkFQ/

Table S14: Additional DNA sequences from Design 4. † signifies "plain version", without fluorophores or quenchers attached. Complexes Rep A, Rep B and Rep C, comprising the corresponding top and bottom strands, also act as thresholds.

Strand name	Sequence
D4_Rep_Back_CB_Top	TCTTTACTCCTTCAACTCTCCA/3IAbRQSp/
D4_Rep_Back_CB_Bot	/56-ROXN/TGGAGAGTTGAAGGAGTAAAGAGGTATAG
D4_Rep_Back_BA_Top	CCCATTTCTCTAACTAACCACC/3IAbRQSp/
D4_Rep_Back_BA_Bot	/56-ROXN/GGTGGTTAGTTAGAGAAATGGGTAGGAAA
D4_Rep_Back_ACi2_Top	CATCACTATCAATCCTACATCT/3IAbRQSp/
D4_Rep_Back_ACi2_Bot	/56-ROXN/AGATGTAGGATTGATAGTGATGTTGTT
D4_Rep_Flux_ABr_Top	/5IAbRQ/CAAACTTACATTACCG
D4_Rep_Flux_ABr_Bot	AGTGATGCGGTAATGTAAGTTTG/3Rox_N/
D4_Rep_Flux_BCj_Top	/5IAbRQ/CATCTTCCCTCCACCG
D4_Rep_Flux_BCj_Bot	AAATGGGCGGTGGAGGGAAGATG/3Rox_N
D4_Rep_Flux_BCj_Top	/5IAbRQ/CATCTTCCCTCCACCG
D4_Rep_Flux_BCj_Bot	AAATGGGCGGTGGAGGGAAGATG/3Rox_N
D4_Rep_Flux_CAp_Top	/5IAbRQ/CTCTTCACACCACTCT
D4_Rep_Flux_CAp_Bot	GTAAAGAAGAGTGGTGTGAAGAG/3Rox_N

Table S15: DNA sequences used for characterizing individual rate constants in Design 4.

Strand name	Sequence
D5_Aq	CACATCATCCAACACCTTCCTCACTTCTCCATAACCAATCCACA
D5_Ap	CACATCATCCAACACCTTCCTCACTTCTCCTCTCTATACTTACA
D5_Br	CTTACCTTCATCCTCCACAATCTTCCATTCACCAACCTCAAACA
D5_Bs	CTTACCTTCATCCTCCACAATCTTCCATTCCTAACCATACTACA
D5_Cj	CTCAACATTTCCACTATCCTTCAACTCTTCCCTCCTACCACCCA
D5_Ck	CTCAACATTTCCACTATCCTTCAACTCTTCATATCTACTAAACA
D5_FLUX_ABr	CACCAACCTCAAACACACATCATCCAACACCTTCCTC
D5_FLUX_BCj	CCCTCCTACCACCCACTTACCTTCATCCTCCACAATC
D5_FLUX_CAp	CATAACCAATCCACACTCAACATTTCCACTATCCTTC
D5_BACK_BA	ACTTCTCTTACCTTCATCCTCCACAATC
D5_BACK_CB	TTCCATCTCAACATTTCCACTATCCTTC
D5_BACK_AC	AACTCTCACATCATCCAACACCTTCCTC
D5_REACT_BOT_BABr	AATGGAAGATTGTGGAGGATGAAGGTAAGAGAAGTGAGGAAGGTGTTGGATGATGTGTG
D5_REACT_BOT_CBCj	AAGAGTTGAAGGATAGTGGAAATGTTGAGATGGAAGATTGTGGAGGATGAAGGTAAGTG
D5_REACT_BOT_ACAp	GAGAAGTGAGGAAGGTGTTGGATGATGTGAGAGTTGAAGGATAGTGGAAATGTTGAGTG
D5_HELP_CAT_BBs†	TTCCATTCCTAACCATACTACATTCCATTCACCAACCTCAAACA
D5_HELP_CAT_CCk†	AACTCTTCATATCTACTAAACAAACTCTTCCCTCCTACCACCCA
D5_HELP_CAT_AAq†	ACTTCTCCTCTCTATACTTACAACTTCTCCATAACCAATCCACA
D5_HELP_BBs†	TTCCATTCCTAACCATACTACATTCCATT
D5_HELP_CCk†	AACTCTTCATATCTACTAAACAAACTCTT
D5_HELP_AAq†	ACTTCTCCTCTCTATACTTACAACTTCTC
D5_HELP_CAT_BBs	/5Alex647N/TTCCATTCCTAACCATACTACATTCCATTCACCAACCTCAAACA
D5_HELP_CAT_CCk	/56-ROXN/AACTCTTCATATCTACTAAACAAACTCTTCCCTCCTACCACCCA
D5_HELP_CAT_AAq	/5Alex488N/ACTTCTCCTCTCTATACTTACAACTTCTCCATAACCAATCCACA
D5_HELP_BBs	/5Alex647N/TTCCATTCCTAACCATACTACATTCCATT
D5_HELP_CCk	/56-ROXN/AACTCTTCATATCTACTAAACAAACTCTT
D5_HELP_AAq	/5Alex488N/ACTTCTCCTCTCTATACTTACAACTTCTC
D5_PROD_BOT_ABrBs	TGATGTGTGTTTGAGGTTGGTGAATGGAATGTAGTATGGTTAGGAATGGAAGA/3IAbRQSp/
D5_PROD_BOT_BCjCk	AGGTAAGTGGGTGGTAGGAGGGAAGAGTTTGTTTAGTAGATATGAAGAGTTGA/3IAbRQSp/
D5_PROD_BOT_CApAq	TGTTGAGTGTGGATTGGTTATGGAGAAGTTGTAAGTATAGAGAGGAGAAGTGA/3IABkFQ/
D5_QUE_HELP_BBs	AATGGAATGTAGTATGGTTAGGAATGGAAGA/3IAbRQSp/
D5_QUE_HELP_CCk	AAGAGTTTGTTTAGTAGATATGAAGAGTTGA/3IAbRQSp/
D5_QUE_HELP_AAq	GAGAAGTTGTAAGTATAGAGAGGAGAAGTGA/3IABkFQ/
D5_REP_A_TOP	/5IAbRQ/CACATCATCCAACACCTTCCTC
D5_REP_A_BOT	GAGAAGTGAGGAAGGTGTTGGATGATGTG/3AlexF647N/
D5_REP_B_TOP	/5IAbRQ/CTTACCTTCATCCTCCACAATC
D5_REP_B_BOT	AATGGAAGATTGTGGAGGATGAAGGTAAG/3Rox_N/
D5_REP_C_TOP	/5IABkFQ/CTCAACATTTCCACTATCCTTC
D5_REP_C_BOT	AAGAGTTGAAGGATAGTGGAAATGTTGAG/3AlexF488N/
D5_REP_A_TOP†	CACATCATCCAACACCTTCCTC
D5_REP_A_BOT†	GAGAAGTGAGGAAGGTGTTGGATGATGTG
D5_REP_B_TOP†	CTTACCTTCATCCTCCACAATC
D5_REP_B_BOT†	AATGGAAGATTGTGGAGGATGAAGGTAAG
D5_REP_C_TOP†	CTCAACATTTCCACTATCCTTC
D5_REP_C_BOT†	AAGAGTTGAAGGATAGTGGAAATGTTGAG

Table S16: DNA sequences from Design 5.

References

[1] D. Y. Zhang, A. J. Turberfield, B. Yurke, and E. Winfree. Engineering entropy-driven reactions and networks catlayzed by DNA. *Science*, 318:1121–1125, 2007.

[2] N. Srinivas, T. E. Ouldridge, P. Sulc, J. M. Schaeffer, B. Yurke, A. A. Louis, J. P. K. Doye, and E. Winfree. On the biophysics and kinetics of toehold-mediated DNA strand displacement. *Nucleic Acids Res.*, 41(22):10641–10658, 2013.

[3] Yuan-Jyue Chen, Neil Dalchau, Niranjan Srinivas, Andrew Phillips, Luca Cardelli, David Soloveichik, and Georg Seelig. Programmable chemical controllers made from DNA. *Nature Nanotech.*, 8:755, 2013.

[4] J. M. Schaeffer, C. Thachuk, and E. Winfree. Stochastic simulation of the kinetics of multiple interacting nucleic acid strands. *DNA Computing and Molecular Programming LNCS*, 9211:194–211, 2015.

[5] D.Y. Zhang and E. Winfree. Control of DNA strand displacement kinetics using toehold exchange. *J. Am. Chem. Soc.*, 131(47):17303–17314, 2009.

[6] E. Protozanova, P. Yakovchuk, and M. D. Frank-Kamenetskii. Stacked-unstacked equilibrium at the nick site of DNA. *J. Mol. Biol.*, 342(3):775 – 785, 2004.

[7] J. SantaLucia, Jr. and D. Hicks. The thermodynamics of DNA structural motifs. *Annu. Rev. Biophys. Biomol. Struct.*, 33:415–40, 2004.

[8] J. SantaLucia, Jr. A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics. *Proc. Natl. Acad. Sci. U.S.A*, 17(95(4)):1460–5, 1998.

[9] David Yu Zhang. Toward domain-based sequence design for DNA strand displacement reactions. *DNA Computing and Molecular Programming*, pages 162–175, 2011.

[10] J. M. Schaeffer. *Stochastic Simulation of the Kinetics of Multiple Interacting Nucleic Acid Strands*. PhD thesis, California Institute of Technology, 2013.

[11] D.Y. Zhang and E. Winfree. Control of DNA strand displacement kinetics using toehold exchange. *J. Am. Chem. Soc.*, 131(47):17303–17314, 2009.

[12] L Qian and E. Winfree. Scaling up digital circuit computation with DNA strand displacement cascades. *Science*, 332(6034):1196–1201, 2011.

[13] J. N. Zadeh, C. D. Steenberg, J. S. Bois, B. R. Wolfe, M. B. Pierce, A. R. Khan, R. M. Dirks, and N. A. Pierce. NUPACK: analysis and design of nucleic acid systems. *J Comput. Chem.*, 32:170–173, 2011.

[14] N.C. Seeman. De novo design of sequences for nucleic acid structural engineering. *J. Biomol. Struct. Dyn.*, 8(3):573–81, 1990.

[15] Constantine G. Evans and Erik Winfree. DNA sticky end design and assignment for robust algorithmic self-assembly. *DNA Computing and Molecular Programming LNCS*, 8141:61–75, 2013.

[16] C. Evans. *Crystals that count! Physical principles and experimental investigations of DNA tile self-assembly*. PhD thesis, California Institute of Technology, 2014.

[17] Erik Winfree. Unpublished software for designing dna sequences based on sequence symmetry minimization.

[18] R. M. Dirks, M. Lin, E. Winfree, and N. A. Pierce. Paradigms for computational nucleic acid design. *Nucl. Acids Res.*, 32(4):1392–1403, 2004.

[19] J. N. Zadeh, B. R. Wolfe, and N. A. Pierce. Nucleic acid sequence design via efficient ensemble defect optimization. *J Comput. Chem.*, 32:439–452, 2011.

[20] B. R. Wolfe and N. A. Pierce. Sequence design for a test tube of interacting nucleic acid strands. *ACS Synthetic Biology*, doi: 10.1021/sb5002196, 2014.

[21] D.Y. Zhang and E. Winfree. Robustness and modularity properties of a non-covalent DNA catalytic reaction. *Nucleic Acids Research*, 38:4182–4197, 2010.

[22] A. J. Genot, D. Yu. Zhang, J. Bath, and A. J. Turberfield. Remote toehold: A mechanism for flexible control of DNA hybridization kinetics. *2011*, 133(7):2177–2182, JACS.

[23] I. G. Panyutin and P. Hsieh. The kinetics of spontaneous DNA branch migration. *Proc. Natl. Acad. Sci. U.S.A*, 91:2021–2025, 1994.

[24] I. G. Panyutin, I. Biswas, and P. Hsieh. A pivotal role for the structure of the holliday junction in DNA branch migration. *The EMBO Journal*, 14(8):1819–1826, 1995.

[25] S. A. McKinney, A. D. J. Freeman, D. M. J. Lilley, and T. Ha. Observing spontaneous branch migration of holliday junctions one step at a time. *Proc. Natl. Acad. Sci. U.S.A*, 102(16):5715–5720, 2005.

[26] B. Yurke, A. J. Turberfield, A. P. Mills, F. C. Simmel, and J. Neumann. A DNA-fueled molecular machine made of DNA. *Nature*, 406:605–608, 2000.

[27] Bernard Yurke and Allen Mills. Using DNA to power nanostructures. *Genet. Prog. Evol. Mach.*, 4:111–122, 2003.

[28] D. Soloveichik, G. Seelig, and E. Winfree. DNA as a universal substrate for chemical kinetics. *Proc. Natl. Acad. Sci. U.S.A*, 107(12):5393–5398, 2010.

[29] J. D. Puglisi and I. Tinoco Jr. Absorbance melting curves of RNA. *Methods Enzymol.*, 180:304–325, 1989.