Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 11 (2012) 43 – 54

Procedia Computer Science

Proceedings of the 3rd International Conference on Computational Systems-Biology and Bioinformatics (CSBio 2012)

Design and analysis of genetically encoded counters

Pakpoom Subsoontorn[a,]*, Drew Endy[a]

[a]*Bioengineering Department, Y2E2-269B, 473 Via Ortega, Stanford University, Stanford, CA, 94305, USA*

Abstract

The ability to store and to act upon modest amounts of information within living systems would enable new approaches to the study and control of biological processes. We develop a hierarchical composition framework supporting the design and analysis of higher-order genetically encoded information storage systems using combinatorial counters as a test case. We first develop a set-reset latch design based on DNA inversion, a toggle flip-flop design based upon two set-reset latches having gated outputs, and an N-bit 2^N state counter design built from N toggle flip-flops. Using computational modeling, we then show how increasing the output gating speed of set-reset latches extends the operable ranges of toggle flip-flops with respect to set-reset latch switching thresholds and toggle flip-flop input pulse height and frequency. Finally, we show how coupling of input/output operable ranges of toggle flip-flops determines the operable ranges of counters. Such frameworks support the comparative analysis of competing low-level designs and further enable engineers to make the best use of limited genetic components while avoiding unnecessary compositional failure.

Selection and/or peer-review under responsibility of the Program Committee of CSBio 2012.

Keywords: synthetic biology, data storage, memory, integrase, excisionase, counter

* Corresponding author. Tel.: +1-650-721-6373; fax: +1-650-721-6602.
E-mail address: tons@stanford.edu

1877-0509 © 2012 Published by Elsevier Ltd.
doi:10.1016/j.procs.2012.09.006

1. Introduction

The ability to store and process modest amounts of programmable information storage inside living cells would enable new approaches in studying and controlling biological processes [1]. For example, with only 8 bits of genetically encoded data storage one could implement a combinatorial counter that could count and report up to 256 discrete biochemical events, such as cell division. Such counting capacity would be sufficient to track and control entire cell lineages during the development and aging across many known organisms.

While the ability to store one bit of information at a genetic level by engineering DNA inversion, or at an epigenetic level by implementing bistable gene regulatory systems, was achieved more than two decades ago [2-3], the complexity and performance of synthetic genetically encoded information storage systems have increased and improved slowly. For example, current information storage demonstrations have not exceeded two bits while attempts to engineer higher-order information storage systems such as a counter and a push-on push-off switch have limited success. More specifically, a recent genetically encoded counter cannot be reset and will fail if the input signal sustains for too long [4]; a recent push-on push-off toggle has a switching efficiency of less than fifty percent for each operation cycle [5].

Three classes of challenges need to be addressed in order to advance genetically encoded information storage. First, more well-characterized and functionally independent instances of key types of genetic encoded functions are required, including transcription operators and their cognate regulators, recombinases and their cognate sites, and so on. Second, tools and approaches that make the physical process of assembling genetically encoded information storage systems more reliable are needed, including sequence level compositional calculators, genetic layout architectures, and so on. Third, given that the first two challenges cover many basic research questions, design and analytical frameworks that support the best organization of scarce starting materials and efficient use of limited project resources become practically essential. For example, many existing one-bit information storage systems are not modular in that they rely on specific inputs such as chemical inducers or radiation to induce state switching; such lack of modularity prevents the straightforward comparison or composition of one-bit information storage systems into higher-order systems. As a second example, many one-bit information storage systems have not included a mechanism to rewrite system state; the inability to rewrite state prevents the implementation of combinatorial counting systems. Practically, while theoretical work has been done to relate how device architecture and molecular kinetics impact the properties of individual one-bit information storage systems, no framework has been developed to explore and prescribe how the properties of one-bit information storage systems might best support the engineering of higher-order systems.

Here, we use a combinatorial pulse counter as a design and computational study case for higher-order information storage system engineering. The counter takes a single input and transitions from one state to the next depending on the number of input pulses received. An N-bit combinatorial counter can count up to 2^N states. We propose an abstraction hierarchy for engineering such combinatorial counters from modular and rewritable one-bit genetically encoded information storage devices. The resulting modularity allows counters to be composed from several different information storage elements, including DNA inversion and bistable transcriptional regulatory systems. Via kinetic modeling we predict that scalable combinatorial counter designs can be achieved within realistic parameter ranges. We further explore how the kinetic parameters and properties of individual bits are expected to define the operable ranges of the resulting counters.

2. Result

2.1 The use of abstraction in combinatorial counter design

We first developed an abstraction hierarchy for combinatorial counters consisting of three functional levels: set-reset latch, toggle flip-flop, and counter (Figure 1). "Set-reset latches" are binary-state information storage devices with two inputs, "input-set" and "input-reset," and two outputs, "output-Q" and "output-Q'." Input-set and input-reset pulses switch a set-reset latch to state-1 (high output-Q; low output-Q') and state-0 (low output-Q; high output-Q'), respectively. "Toggle flip-flops" are single-input single-output binary-state information storage devices built from set-reset latches. A toggle flip-flop can be toggled between state-0 (low output) and state-1 (high output) with an input pulse. "N-bit counters" are single-input N-output 2^N state information storage devices built from N toggle flip-flops. The combinatorial state of the N-toggle flip-flops corresponds to the number of input pulses received by the counter.

There are many ways to implement qualitatively equivalent functional devices at a given level within the abstraction hierarchy. A useful abstraction hierarchy should also provide flexibility for engineers to compare the performance of and to connect devices regardless of their internal mechanisms. To enable such flexibility we define device boundaries so that the number of RNA polymerase molecule passing a specific position on DNA per second (Polymerase Per Second, or PoPS) can be used to define device input and output signal levels [7-8]. While in the next section we present the development of a single example of a set-reset latch, toggle flip-flop, and counter.

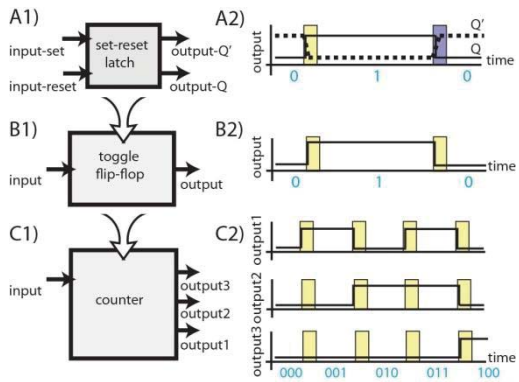

Fig 1. A set-reset latch, a toggle flip-flop, and a counter. (A1-C1) depict a set-reset latch, toggle flip-flop, and three-bit counter, respectively. Black arrows represent signals received or sent by each device. (A2-C2) depict the idealized input/output response for each type of device. Blue numbers depict the internal state of each device over time. Note that counter output1 represents the state for the toggle flip-flop encoding the least significant bit within a counter. Yellow areas represent a pulse of input-set to a set-reset latch or of input to a toggle flip-flop or a counter. The dark blue area represents a pulse of input-reset to a set-reset latch.

2.2 Designs and operations of devices comprising genetically encoded combinatorial counters

2.2.1 Implementation of a set-reset latch

A set-reset latch requires three functional elements: a "storage element," an "input element," and an "output element." A storage element is a binary state system that can maintain its state in the absence of inputs. An input element receives two PoPS input signals, input-set and input-reset, and switches the storage element state accordingly. An output element converts the storage element state to the values of two complementary PoPS output signals, output-Q and output-Q.'

Here we propose an implementation of a set-reset latch based on a bacteriophage integrase-excisionase system. We call this implementation a "DNA inversion (DI)" set-reset latch (Figure 2A1). The input element of a DI set-reset latch consists of an integrase gene expressed under input-set and an integrase-excisionase polycistron expressed under input-reset. The storage element is a pair of recombination sites arranged such that an integrase tetramer inverts DNA between the sites in one direction, called the "ON configuration," while a tetramer of integrase-excisionase reverts the DNA between the sites to the other direction, called the "OFF

configuration" [6]. The output element is a promoter located between the recombination sites such that transcription is directed to output-Q or output-Q' direction when the storage element is in the ON or OFF configuration, respectively.

A DI set-reset latch operates as the following (Figure 2A2). In the absence of inputs there is no production of integrase or excisionase. Therefore, the storage element remains in its current configuration. An input-set pulse leads to expression of integrase that then transitions the storage element to the ON configuration thereby switching the set-reset latch to state-1 (high output-Q low output-Q'); an input-reset pulse leads to expression of both integrase and excisionase whose combined activity transitions the storage element to the OFF configuration thereby switching the set-reset latch to state-0 (low output-Q high output-Q').

2.2.2 Implementation of a toggle flip-flop

A toggle flip-flop is more complicated than a set-reset latch due to a requirement that a toggle flip-flop must alternate its output signal in response to a single repeating input signal. Thus, the response of the toggle flip-flop must be different depending on internal state of the flip-flop. As a result, a toggle flip-flop must somehow retain information about its previous state even while switching to a next state. Such information can be maintained if a toggle flip-flop itself encodes two bits of information storage. The first bit stores flip-flop state in the absence of an input pulse; the second bit stores state during an input pulse.

Here we propose an implementation of a toggle flip-flop which we call an "activation-inhibition (AI)" toggle flip-flop (Figure 2B1). An AI toggle flip-flop consists of a pair of orthogonally-operating set-reset latches called an "A-latch" (Figure 2B1 blue area) and an "I-latch" (Figure 2B1 purple area). The I-latch outputs, Q and Q', are connected to the A-latch inputs, set and reset, respectively, while the A-latch outputs, Q and Q', are connected to I-latch inputs, reset and set, respectively. The input to a toggle flip-flop expresses an activator, A, and a repressor, R, which can activate and repress output element promoters of the A- and I-latches, respectively. The A-latch has another output element with a constitutive promoter connecting to the toggle flip-flop output.

An AI toggle flip-flop operates by having its A- and I-latches take turn switching state (Figure 2B2). In the absence of an input pulse, the A level is low and there is no output from the A-latch to the I-latch, regardless of the A-latch state. Consequently, the I-latch holds its current state. At the same time, the R level is low so that the I-latch can send outputs to switch A-latch state. Since the I-latch outputs, Q and Q', are connected to the A-latch inputs, set and reset, respectively, the A-latch will be switched to the same state as the current I-latch state. During an input pulse the R level is high and there is no output from the I-latch to the A-latch regardless of the I-latch state and thus the A-latch holds its current state. At the same time, the A level is high and the A-latch can drive switching of I-latch state. Since the A-latch outputs, Q and Q', are connected to the I-latch inputs, reset and set, respectively, the I-latch will be switched to the state opposite the current A-latch state.

Given a toggle flip-flop with both A- and I-latch in state-0, during the first pulse the A-latch holds state-0 while the I-latch switches from state-0 to state-1 (Figure 2B2). After the first pulse, the I-latch holds state-1 while the A-latch switches from state-0 to state-1. Since the toggle flip-flop output is connected to the A-latch the first input pulse switches the toggle flip-flop from state-0 (low output) to state-1 (high output). During the second pulse the A-latch holds state-1 while the I-latch switches from state-1 to state-0. After the second pulse the I-latch holds state-0 while the A-latch switches from state-1 to state-0. Therefore, the second input pulse switches the toggle flip-flop from state-1 (high output) to state-0 (low output). Taken together, the mechanisms described above allow a toggle flip-flop to switch back and forth between state-0 and state-1 using the same recurring input pulse.

2.2.3 Implementation of a counter

A combinatorial counter can be assembled from toggle flip-flops if the flip-flops are connected such that the flip-flop encoding the k^{th} bit is triggered to change state every 2^{k-1} input pulses to the counter. For example, a 3-

bit counter counts up as the following: 000, 001, 010, 011, 100, 101, 110 and 111. Notice that the least significant bit is toggled between 0 and 1 every count, the next significant bit is toggled every two counts (e.g., 000 to 010), and the most significant bit is toggled every 4 counts (e.g., 000 to 100).

An N-bit asynchronous counter has an output from each toggle flip-flop connecting directly to an input of the toggle flip-flop encoding the next significant bit (Figure 2C1). An asynchronous counter exploits the fact that every two consecutive input pulses into a toggle flip-flop give rise to a single output pulse. If 2^N input pulses enter an asynchronous counter, bit-1, the least significant bit of the counter, will receive 2^N input pulses directly, bit-2 will receive $2^N/2^1$ pulses from bit-1, bit-3 will receive $2^N/2^2$ pulses from bit-2 and so on.

Therefore, a toggle flip-flop of the k^{th} bit changes state every 2^{k-1} input pulses (Figure 2C2). Since the toggle flip-flop encoding each bit can only switch state following a switch in the state of the toggle flip-flop encoding the next less significant bit, state switching times within asynchronous combinatorial counters grow with the number of bits.

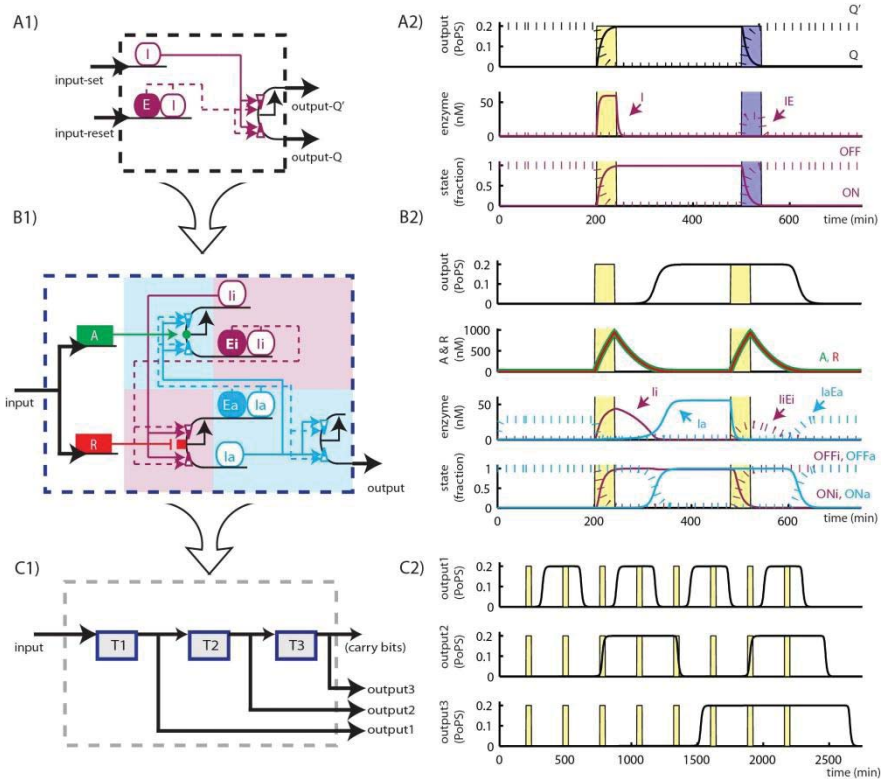

Fig. 2. DNA inversion based set-reset latches, toggle flip-flops and combinatorial counters. (A1-C1) DNA inversion set-reset latch, activation-inhibition toggle flip-flop, and three bit asynchronous counter architectures, respectively. Horizontal black arrows represent PoPS transcription signals received or sent by devices. Blue and purple areas in (B1) denote A- and I-latches, respectively. A generic set-reset latch, A-latch and I-latch are shown in the OFF configuration. Bent black arrows represent promoters. Round rectangles labelled "I," "Ii" and "Ia" represent integrase genes within a generic set-reset latch, an I-latch, and an A-latch, respectively; round rectangle labeled "E", "Ei" and "Ea" represent excisionase genes within a generic set-reset latch, an I-latch and an A-latch, respectively. Rectangles labelled "A" and "R" represent transcriptional activator and repressor genes, respectively. Paired triangles represent recombination sites. Small squares and circles at the bottoms of promoters represent repressible and activatable operator sites, respectively. For (C1), gray rectangles labeled "T1", "T2" and "T3" represent toggle flip-flops within a counter. (A2-C2) Simulated operational time-courses for a DNA inversion set-reset latch, activation-inhibition toggle flip-flop, and asynchronous counter, respectively. The yellow and dark blue areas in (A2) mark the duration of input-set and input-reset pulses, respectively. Yellow areas in (B2) and (C2) mark the duration of input to a toggle flip-flop and a counter, respectively. Subplots labelled "enzyme" show the levels of integrase and integrase-excisionase complex over time. Subplots labelled "state" show the fractions of latch storage elements in each configuration over time. See appendix for the full description of reaction mechanisms and parameter values.

2.3 Performance analysis of devices comprising genetically encoded combinatorial counters

To be useful counters must be designed to operate under constraints set by their application(s). For example, a counter for tracking cell division cycles must be able to count at least as fast as the cell cycle frequency. Generally, applications determine the range of "input signal parameters" that counters must process including, for example, the height and width of input pulses or the gaps between pulses. The range of input signal parameters under which a counter can operate, in turn, depends on the kinetic and other parameters of the molecular components used in their construction including, for example, transcription, translation and degradation rates for different gene products. In practice, the kinetic parameters should result from engineering choices – which specific promoters, ribosome binding sites, degradation tags, recombinases, and so on, are used. Thus, to enable the design of counters to meet specific application requirements it is important to understand the relationship between the operable input ranges a counter must meet and the kinetic parameters of its molecular components.

The levels and width of input pulses are the two key external factors influencing the operation of a set-reset latch. For example, for the DI set-reset latch developed above, the input-set and input-reset levels together determine integrase and excisionase expression levels which in turn determine if the set-reset latch will set, reset or hold state. How such a DI set-reset latch will respond to different combinations of input-set and input-reset levels can be represented by a phase-plane for latch state fate (Figure 3A1). To operate properly, the basal input level in the absence of a formal input pulse must be within the "state holding" region of the phase-plane. At other times, such as during a set or a reset pulse, input levels must be within the "set" or "reset" regions of the phase-plane, respectively.

The operable range of input pulse width for a DI set-reset latch depends on the state switching speed of the latch. If the time required for state switching is short, the minimal input pulse width required to avoid state switching failure will also be short. The time required for state switching, in turn, depends on the input pulse height (Figure 3A2, solid curve). For the DI set-reset latch considered here, increasing pulse height increases recombinase expression and, thus, decreases state switching time; the nonlinearity of this curve results from the four integrase or integrase-excisionase complex catalyzing DNA inversion. At the lower limit for input pulse height the switching time curve approaches a vertical asymptote corresponding to a "switching threshold" for input pulse height that matches the border of the state holding region in the phase-plane.

Internal to the DI set-reset latch, the Michealis and catalytic rate constants of the recombinase itself are the two key parameters determining the relationship between switching time and input pulse height (Figure 3A2, dotted and dashed lines). A lower Michealis constant means that the recombinase binds to its target site with a higher affinity; a higher catalytic rate constant means that recombination proceeds faster once the recombinase binds to its cognate target. In both cases, state switching time will be shorter for a given recombinase expression level. However, the state switching threshold is more sensitive to a change in the Michealis constant when the input pulse height is low, since at low recombinase levels recombinase-target site binding is expected to be rate limiting. As the input pulse height increases, switching time is expected to become more sensitive to changes in the catalytic rate as the recombination sites themselves become saturated with recombinase complexes.

Next, in considering toggle flip flops we noted that the two set-reset latches, A- and I-, within a properly operating AI toggle flip-flop alternate switching state during each toggle cycle. From this we defined a critical "time window" during which an A- or I-latch must switch state while its input-set or input-reset signal is outside the state holding levels. As a result, a toggle flip-flop will be expected to fail to operate if either the A- or I-latch state switching time windows are too short such that a latch does not switch state, leading to "incomplete switching failures," or the state switching time windows for the A- and I-latches overlap, leading to "mutual switching failures." Avoiding both incomplete and mutual switching failure modes will together constrain the operating ranges of toggle flip-flops.

Practically, the operable range of toggle flip-flops is related to the properties of the underlying set-reset latches (Figure 3B1 and 3B2). For example, the switching thresholds of the set-reset latches, the A- and I-latch,

determine the range of their holding regions. A higher switching threshold corresponds to a larger state holding region and thus a shorter switching time window. At the limit of high A- or I-latch switching thresholds (high Michaelis or low catalytic rate constant) the time window of A- or I-latch state switching is too short and the toggle flip-flop fails due to incomplete switching. Alternatively, at the limit of low A- or I-latch switching thresholds (low Michealis or high catalytic rate constant) the time windows of A- and I-latch state switching overlap and the toggle flip-flop fails due to mutual switching. Note that the lower bounds of the A- and I-latch switching thresholds (lower bound and upper bound of their Michealis and catalytic rate constants, respectively) are coupled. Stated differently, we can increase the operable switching time window of the A-latch by reducing the operable switching time window of the I-latch, or vice versa, so long as the A- and I-latch switching time windows do not overlap.

We next considered the relationship between the magnitude (or height) of an input signal to a toggle made from any particular A- and I- latches. To do this we defined the input level during a pulse as "pulse height" and the input level in the absence of a formal pulse as "gap height." Physically, these signal levels drive the expression of activator, A, and repressor, R, levels during a pulse and a gap, respectively. A and R levels, in turn, determine input-set and input-reset levels to the I- and A-latches, respectively. Operationally, the gap height needs to be low enough so that A and R levels are low enough for the I- and A-latches to hold and switch states, respectively, and the pulse height needs to be high enough so that A and R levels are high enough for the I- and A-latch to switch and hold states respectively (Figure 3B3). Note that the operable pulse height also has an upper bound; if the pulse height is too high there will be too much A and R left over during the gap for the I-latch to hold state or for the A-latch to switch state.

We then considered the relationship between the duration (or width) of an input signal to a toggle made from any particular A- and I- latches (Figure 3B4). To do this we defined signal duration of an input pulse as the "pulse width" and the time between signals as the "gap width." Lower bounds of pulse and gap widths need to be long enough for I- and A-latches, respectively, to switch state. Moreover, the lower bound of gap width increases with pulse width due to the fact that the longer a pulse sustains the more A and R are produced during a pulse; the higher A and R levels at the end of the pulse, the longer gap width is required for A and R to decay. Note that the lower bound of gap width eventually levels off when the pulse width is so long that A and R reach steady-states.

We next explored how the turnover rates of activator, A, and repressor, R, might act as key parameters in determining the operating range for toggles. Specifically, given faster A and R turnover rates the input-set and input-reset to the A- and I-latches become more responsive to changes in toggle flip-flop input. This increasing responsiveness sharpens the switching time windows for the A- and I-latches and results in an expansion of toggle operable range given low Michealis and high catalytic rate constant values (Figure 3B1 and 3B2). In addition, faster A and R turnover rates mean that both A and R can be cleared faster following a pulse, resulting in a higher upper bound of operable pulse height and a reduced lower bound of operable gap width (Figure 3B3 and 3B4).

Counters function properly when the input to the toggle flip-flop encoding each bit is within the operable input range of the toggle flip-flop. Therefore, an operable input range of a counter is contained within the operable input range of a single toggle flip-flop (Figure 3C1 and 3C2). However, note that a certain region of the operable input range for a single toggle flip-flop falls outside the operable range of a counter (Figure 3C2, lower left grey region). For example, we show that for activation-inhibition toggle flip-flops the pulse height, gap height, rise width and fall width (defined as time duration a signal takes to rise and fall at the beginning and at the end of a pulse, respectively) remain nearly constant as a signal propagates through a counter (Figure 3C3). However, as expected, the pulse and gap widths double in duration across each toggle flip-flop since two input pulses are required to produce a single output pulse. Stated differently, if the output pulse and gap widths of a toggle fall outside the allowable input pulse and gap widths for the same toggle then such toggles cannot be combined to produce higher order combinatorial counters.

3. Discussion and conclusion

While one-bit synthetic genetically encoded information storage systems have long been achieved, design and analysis frameworks that support the engineering of higher-order information storage systems have been lacking. In this work, we present examples of designs and analyses of such higher-order information storage systems using combinatorial counters as a test case.

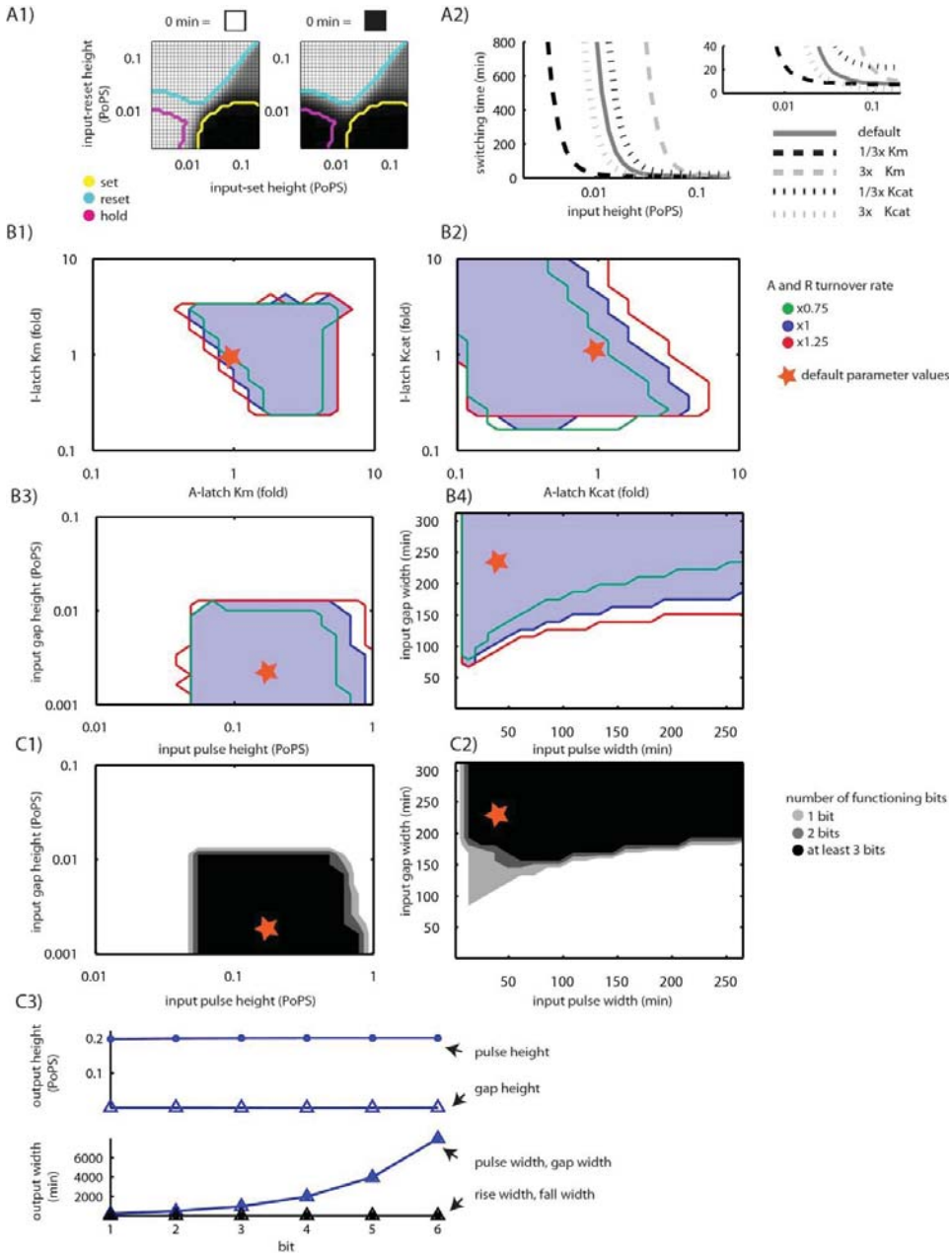

Fig 3. The dynamic properties of genetically encoded set-reset latches and working ranges of toggle flip-flops define the operating range and performance of combinatorial counters. (A1) Expected phase state diagram for a DNA inversion set-reset latch fate. Each grid shows

the final state, defined as the output-Q level, of a set-reset latch that has received a particular combination of input-set and input-reset values for 1100 minutes. The darker grid corresponds to a state closer to state-1. The initial state of a set-reset latch is either state-0 (left) or state-1 (right). The regions are marked as set (yellow), reset (blue) and hold (pink) if the final output-Q level is within 10 percent of the maximal level, the minimal level and the level at time zero, respectively. (A2) Expected switching time of DNA inversion set-reset latches, from state-0 to state-1, as a function of the input-set pulse height. Different lines are expected switching time curves of DI set-reset latches with different recombinase Michaelis (Km) or catalytic (Kcat) constants. (Inset) Zoomed-in switching time curves highlighting the effects of varying Km and Kcat at high input pulse height. The switching time is defined as the duration from the arrival of an input pulse until when output-Q and output-Q' levels pass half of their maximal values. The input-reset value is fixed at zero. Switching time curves with respect to input-reset pulse height are similar to curves shown here (data not shown). (B1-B4) Expected activation-inhibition toggle flip-flop operable ranges with respect to kinetic parameters or input signal parameters. The overlaying red, blue, and green contours are the operable ranges at different relative turnover rates of activator A and repressor R. Toggle flip-flop is scored as operating if the toggle flip-flop changes from state-0 to state-1 following an odd number of input pulses, and to state-0 following an even number of input pulses. Orange stars mark default parameter values. (C1-C2) Predicted asynchronous counter operable ranges with respect to input signal parameters. (C3) Expected changes in propagating pulse and gap height and width during asynchronous counter operation. Pulse widths and gap widths are defined as the average duration for which output level from each bit is above 75% and below 5% of the maximal output level, respectively. Pulse height and gap height are defined as the average output value during pulse width and gap width, respectively. Rise width and fall width are defined as the average duration for which the output levels rise from 5% to 75% and fall from 75% to 5% of the maximal output level, respectively.

Earlier studies have demonstrated the use of DNA inversion or bistable genetic systems to store one bit of information. The set-reset latch proposed here differs from these earlier designs in that it provides mechanisms for both setting and resetting latch state using signals from an arbitrary transcriptional regulatory circuit upstream. The toggle flip-flops proposed here are similar in function to the Lou et al. push-on push-off switch in that they depend on one bit information storage devices that can be toggled between two states via a single input [5]. However, the Lou et al. push-on push-off switch relies on a delay in gene expression kinetics to temporarily store state during a pulse. Such toggle flip-flops will fail if an input pulse is too long and are likely impractical for use in implementing asynchronous counters. In contrast, the activation-inhibition toggle flip-flops considered here use two independent set-reset latches to store state during and between input pulses and thus have no theoretical upper bound with respect to operable input pulse width.

The DNA inversion cascade counter by Friedland et al., like our asynchronous counter, is built by connecting the output from a device storing one-bit to the device storing the next bit [4]. However, each bit within our asynchronous counters is a toggle flip-flop that can be set and reset, whereas each bit within the Friedland et al. counter is a latch that can only be set. This design difference has two functional consequences. First, the Friedland et al. counter functions as a thermometer-code counter that can count only up to N given N bits, while our design functions as a combinatorial counter that can count up to 2^N using N bits. Second, the Friedland et al. counter does not have a mechanism to prevent a long input pulse from switching multiple bits; if an input pulse persists once a bit switches to state-1 it sends a signal to switch the next bit to state-1 and so on. In contrast, the asynchronous counters considered here use a negative edge-trigger flip-flop that can hold the current state until an input pulse ends. Such counter designs prevent multiple-bit switching per pulse.

Conventionally, input and output signals for genetically encode devices are defined via the concentrations of some molecular species. Since different device designs typically make use of different molecules it is difficult if not impossible to directly compare the performances across designs. Instead, our design framework requires all devices to receive and send information via a common signal carrier, allowing for direct comparison of input/output functionality across different designs. Thus, we can directly compare attributes such as switching thresholds, switching times, and operable ranges of devices at the same functional level regardless of their internal architectures. The ability to directly compare any device implementation simplifies future device design, characterization, standardization, and reuse. Moreover, other designs for latches, toggles, or counters, not considered here, could be added in the future. For example, alternative set-reset latch designs might store information via the bistability of DNA methylations, protein phosphorylation, or chromatin remodeling; alternative toggle flip-flop designs may use riboswitches or proteases to gate signal flow between latches.

Our studies emphasize the significance of characterizing and tuning state-switching thresholds, device speeds, and operable ranges across multiple levels of a functional abstraction hierarchy. For example, we show that a toggle flip-flop can operate only when its set-reset latch state-switching thresholds and speeds allow set-

reset latches to take turn switching state for a given toggle input, thus avoiding incomplete switching and mutual switching failures. We also show that a counter can operate only when the counter input that propagates through each toggle flip-flop in the counter remains within the toggle flip-flop operable ranges. This finding is important given that state-switching thresholds and speeds have rarely been characterized in earlier studies of genetically encoded information storage devices and thus deserve more attention in the future.

Given our initial framework and analysis presented herein and the overall goal of developing a capacity to engineer scalable genetically encoded information storage systems, we believe that future experimental work should first focus on implementing and characterizing rewritable-modular one-bit information storage devices. The design architectures of DNA inversion or mutual inhibition set-reset latches in this study, together with earlier studies of integrase-excisionase systems and other modular one-bit information storage devices, could serve as starting points [6, 9]. Standardized and efficient procedures for characterizing and screening devices with desirable properties need to be developed. To scale information storage capacity to multiple bits, larger sets of functionally independent (i.e., orthogonal) parts need to be engineered and characterized.

In addition, future theoretical work should address the limitations of our analyses. First, our abstraction hierarchy requires that each device receives and sends a PoPS signal, requiring that device operations involve transcriptional regulation. Next generation set-reset latches, toggle flip-flops, and counters may all operate at post-transcriptional levels or use other, to-be-defined common signal carriers, such as ribosomes per second (RiPS). Second, our analysis does not take into account still poorly understood effects arising from apparent stochasticity in gene expression and other cellular phenomena. In the context of our higher order information storage systems, stochastic effects could undermine the ability of the device to hold state or influence the timing of state switching. The extent to which such effects might interfere with or improve the operation of multiple-bit information storage systems needs be addressed in future studies.

Acknowledgements

We thank M Ortiz, J Bonnet, F St-Pierre, R Murray, L Martin and C Rodriguez for discussion. PS is supported by Stanford Bio-X Bioengineering Graduate Fellowship. DE acknowledges generous support from NSF Synthetic Biology Engineering Research Center and Stanford University.

References

[1] Burrill DR & Silver PA. Making cellular memories. *Cell; 2010,* 140(1):13-18

[2] Podhajska AJ, *et al* (1985) Control of cloned gene expression by promoter inversion in vivo: construction of the heat-pulse-activated att-nutL-p-att-N module. *Gene; 1985,* 40(1):163-168.

[3] Toman Z, *et al*. A system for detection of genetic and epigenetic alterations in Escherichia coli induced by DNA-damaging agents. *J Mol Bio; 1985,l* 186(1):97-105.

[4] Friedland AE*, et al.* Synthetic gene networks that count. *Science; 2009,* 324(5931):1199-1202 .

[5] Lou C*, et al.* Synthesizing a novel genetic sequential logic circuit: a push-on push-off switch. *Mol Syst Biol; 2010,* 6:350

[6] Bonnet J, *et al.* Rewritable digital data storage in live cells via engineered control of recombination directionality. Proc Natl Acad Sci USA; 2012, 109(23): 8884-9

[7] Canton B, *et al*. Refinement and standardization of synthetic biological parts and devices. *Nat Biotechnol; 2008,* 26(7):787-793

[8] Kelly JR*, et al.*Measuring the activity of BioBrick promoters using an in vivo reference standard. *J Biol Eng; 2009,* 3:4.

[9] Kobayashi H*, et al.* Programmable cells: interfacing natural and engineered gene networks. *Proc Natl Acad Sci USA; 2004,* 101(22):8414-8419

Appendix A. Kinetic models and parameters

All simulation results presented in this study were generated from deterministic mass-action kinetic models. Cell volume and copy number of genes were assumed to be constant. Gene expressions were modelled as

single-step protein productions from genes. Protein production rates thus encapsulate transcription, translation and mRNA degradation rates. If a gene in a device is expressed under a PoPS input, its production rate was assumed to be proportional the PoPS input value. Transcriptional regulator degradation rates were assumed to be proportional to its monomer concentrations. Integrase and excisionase degradation rates were assumed to be proportional to their total concentrations. All molecular association and dissociation reactions were assumed to be at the equilibriums. Recombinase concentrations were assumed to be much higher than substrate concentrations. Thus, total recombinase concentration (free recombinase + substrate-bound recombinase) and free recombinase concentration were assumed to be the same. All models were represented as sets of ODEs and numerically solved using MATLAB ® ode45 solver.

Particular kinetic parameter values shown in tables below values were chosen from their biologically realistic ranges to simplify the presentation. PoPS inputs in this study were assumed to be equivalent to transcription rates. These values can range from zero, if a device is not connected to any input, to maximal promoter clearance rate times the number of DNA template. Number of protein molecules produced per transcript is approximately equal to its translation rate divided by its mRNA degradation rate. This value could range from zero, for mRNA that cannot be translated, up to tens of protein molecules per mRNA. Protein half-life ranges from a few minutes, for rapidly degraded proteins with strong degradation tags, to several hours, for stable proteins which can only be diluted by cell growth [A1]. We assume that transcription factors in this study have perfect cooperative stability, i. e., only protein monomer can be degraded [A2]. The degradation rates of the monomer are chosen so that total proteins have ~20 minute half-life. Protein dimer dissociation equilibrium constants are in order of 10 nM [A2]. Protein-DNA dissociation equilibrium constants are between 1-1000 nM [A3]. Fold changes in activation or inhibition by transcription factor are between 10-1000 folds [A4]. Michealis constants and catalytic constants of recombinase were chosen to be within the same order of magnitudes from the values derived from the studies of FLP and Cre recombination kinetics [A5].

Table A1. DNA inversion set-reset latch model and parameters

Reactions	Descriptions	Default values
$\emptyset \rightarrow I$	Integrase production from set input	1.5 /PoPS set
$\emptyset \rightarrow I$	Integrase production from reset input	0.5 /PoPS reset
$\emptyset \rightarrow E$	Excisionase production from reset input	1.5 /PoPS reset
$I \rightarrow \emptyset$	Free integrase degradation	0.3/ min
$IE \rightarrow E$	Integrase degradation from complex	0.3/ min
$E \rightarrow \emptyset$	Free excisionase degradation	0.3/ min
$IE \rightarrow I$	Excisionase degradation from complex	0.3/ min
$IE \leftrightarrow I + E$	Integrase-excisionase binding	1 nM
$4I + OFF \leftrightarrow OFFI_4 \rightarrow ON + 4I$	Integrase catalyzes set	Km 10 nM; Kcat 0.1/min
$4IE + ON \leftrightarrow OFF(IE)_4 \rightarrow OFF + 4\text{IE}$	Integrase excisionase catalyzes reset	Km 10 nM; Kcat 0.1/min
$OFF \rightarrow OFF + Q'$	OFF state latch produces output	12 nM/min
$ON \rightarrow ON + Q$	ON state latch produces output	12 nM/min

Table A2. DNA inversion activation-inhibition toggle flip-flop

Reactions	Descriptions	Default values
$\emptyset \rightarrow A$	Activator production	3/ PoPS input
$\emptyset \rightarrow R$	Repressor production	3/ PoPS input
$A \rightarrow \emptyset$	Activator degradation	0.2/ min
$R \rightarrow \emptyset$	Repressor degradation	0.2/ min
$A_2 \leftrightarrow 2A$	Activator dimerization	20 nM
$B_2 \leftrightarrow 2B$	Repressor dimerization	20 nM
$OFFaA_2 \leftrightarrow OFFa + A_2$	Activator-A-latch binding	150 nM
$ONaA_2 \leftrightarrow ONa + A_2$	Activator-A-latch binding	150 nM
$OFFiR_2 \leftrightarrow OFFi + R_2$	Repressor-I-latch binding	6 nM
$ONiR_2 \leftrightarrow ONi + R_2$	Repressor-I-latch binding	6 nM

$OFFa \rightarrow OFFa + Ii$	Basal I-latch set integrase production	0.18 / min
$OFFaA_2 \rightarrow OFFaA_2 + Ii$	Activated I-latch set integrase production	18 / min
$ONa \rightarrow ONa + Ii$	Basal I-latch reset integrase production	0.06 / min
$ONaA_2 \rightarrow ONaA_2 + Ii$	Activated I-latch reset integrase production	6/ min
$ONa \rightarrow ONa + Ei$	Basal I-latch excisionase production	0.18 / min
$ONaA_2 \rightarrow ONaA_2 + Ei$	Activated I-latch excisionase production	18 / min
$Ii \rightarrow \emptyset$	I-latch free integrase degradation	0.3/ min
$IiEi \rightarrow Ei$	I-latch integrase degradation from complex	0.3/ min
$Ei \rightarrow \emptyset$	I-latch free excisionase degradation	0.3/ min
$IiEi \rightarrow Ii$	I-latch excisionase degradation from complex	0.3/ min
$IiEi \leftrightarrow Ii + Ei$	Integrase-excisionase binding	1 nM
$ONi \rightarrow ONi + Ia$	Basal A-latch set integrase production	18 / min
$ONiR_2 \rightarrow ONiR_2 + Ia$	Repressed A-latch set integrase production	0.18 / min
$OFFi \rightarrow OFFi + Ia$	Basal A-latch reset integrase production	6/ min
$OFFiR_2 \rightarrow OFFiR_2 + Ia$	Repressed A-latch reset integrase production	0.06 /min
$OFFi \rightarrow OFFi + Ea$	Basal A-latch excisionase production	18 / min
$OFFiR_2 \rightarrow OFFaiR_2 + Ea$	Repressed A-latch excisionase production	0.18 / min
$Ia \rightarrow \emptyset$	A-latch integrase degradation	0.3/ min
$IaEa \rightarrow Ea$	A-latch integrase degradation	0.3/ min
$Ea \rightarrow \emptyset$	A-latch excisionase degradation	0.3/ min
$IaEa \rightarrow Ia$	A-latch excisionase degradation	0.3/ min
$IaEa \leftrightarrow Ia + Ea$	Integrase-excisionase binding	1 nM
$4Ii + OFFi \leftrightarrow OFFiIi_4 \rightarrow ONi + 4Ii$	I-latch set	Km 10 nM; Kcat 0.1/min
$4Ii + OFFiR_2 \leftrightarrow OFFi\ Ii_4R_2 \rightarrow ONiR_2 + 4Ii$	I-latch set	Km 10 nM; Kcat 0.1/min
$4IiEi + ONi \leftrightarrow ONi(IiEi)_4 \rightarrow OFFi + 4IiEi$	I-latch reset	Km 10 nM; Kcat 0.1/min
$4IiEi + ONiR_2 \leftrightarrow ONi(IiEi)_4R_2 \rightarrow OFFiR_2 + 4IiEi$	I-latch reset	Km 10 nM; Kcat 0.1/min
$4Ia + OFFa \leftrightarrow OFFa(Ia)_4 \rightarrow ONa + 4Ia$	A-latch set	Km 10 nM; Kcat 0.1/min
$4Ia + OFFaA_2 \leftrightarrow OFFa(Ia)_4A_2 \rightarrow ONaA_2 + 4Ia$	A-latch set	Km 10 nM; Kcat 0.1/min
$4Ia + OFF \leftrightarrow OFF(Ia)_4 \rightarrow ON + 4Ia$	A-latch output element set	Km 10 nM; Kcat 0.1/min
$4IaEa + ONa \leftrightarrow ONa(IaEa)_4 \rightarrow OFFa + 4IaEa$	A-latch reset	Km 10 nM; Kcat 0.1/min
$4IaEa + ONaA_2 \leftrightarrow ONa(IaEa)_4A_2 \rightarrow OFFaA_2 + 4IaEa$	A-latch reset	Km 10 nM; Kcat 0.1/min
$4IaEa + ON \leftrightarrow ON(IaEa)_4 \rightarrow OFF + 4IaEa$	A-latch output element reset	Km 10 nM; Kcat 0.1/min
$ON \rightarrow ON + Q$	ON state latch produces output	12 nM/min

Each bit of an asynchronous counter consists of only a toggle flip-flop. The output of one toggle flip-flop is connected to the input of a toggle flip-flop in the next bit. No additional component or interaction is required. To simulate combinatorial counter operation, we simply simulate the toggle flip-flop of the least significant bit and then use its output to as an input to simulate the next significant bit and so on.

Appendix references

[A1] Gottesman S, Maurizi M R. Regulation by proteolysis: energy-dependent proteases and their targets. *Microbiol. Rev.; 1992,* 56:592-621.

[A2] Buchler N E, *et al*. Nonlinear protein degradation and the function of genetic circuits. *Proc Natl Acad Sci USA; 2005,* 102(27):9559-9564.

[A3] Gerland U, *et al*. Physical constraints and functional characteristics of transcription factor-DNA interaction. *Proc Natl Acad Sci USA; 2002,* 99:12015-12020.

[A4] Lutz R, Bujard H. Independent and tight regulation of transcriptional units in Escherichia coli via the LacR/O, the TetR/O and AraC/I1-I2 regulatory elements. *Nucleic Acids Res; 1997,* 25: 1203-1210.

[A5] Ringrose L, *et al*. Comparative kinetic analysis of FLP and cre recombinases: mathematical models for DNA binding and recombination. *J Mol Biol; 1998,* 284(2):363-384.