RESEARCH ARTICLE SUMMARY

SYNTHETIC BIOLOGY

Cellular checkpoint control using programmable sequential logic

Lauren B. Andrews, Alec A. K. Nielsen, Christopher A. Voigt*

INTRODUCTION: Modern computing is based on sequential logic, in which the state of a circuit depends both on the present inputs as well as the input history (memory). Implementing sequential logic inside a living cell would enable it to be programmed to progress through discrete states. For example, cells could be designed to differentiate into a multicellular structure or order the multistep construction of a material. A key challenge is that sequential logic requires the implementation of regulatory feedback, which has proven difficult to design and scale.

RATIONALE: We present a quantitative method to design regulatory circuits that encode sequential logic. Our approach uses NOT gates as the core unit of regulation, in which an input promoter drives the expression of a repressor protein that turns off an output promoter. Each gate is characterized by measuring its response function, in other words, how changing the input affects the output at steady state. Mathematically, the response functions are treated as nullclines, and tools from nonlinear dynamics (phase plane and bifurcation analyses) are applied to predict how combining gates leads to multiple steady states and dynamics. The circuits can be connected to genetic sensors that respond to environmental information. This is used to implement checkpoint control, in which the cell waits for the right signals before continuing to the next state. Circuits are built that instruct *Escherichia coli* to proceed through a linear or cyclical sequence of states.

RESULTS: First, pairs of repressors are combined to build the simplest unit of sequential logic: a set-reset (SR) latch, which records a digital bit of information. The SR latches can be easily connected to each other and to sensors because they are designed such that the inputs and outputs are both promoters. Each latch requires two repressors that inhibit each other's expression. A total of 11 SR latches were designed by using a phase plane analysis. The computation accurately predicts the existence of multiple steady states by using only the empirical NOT gate response functions. A set of 43 circuits was constructed that connects these latches to different combinations of sensors that respond to small molecules in the media. These circuits are shown to reliably hold their state for >48 hours over many cell divisions, only switching states in response to the sensors that connect to the set and reset inputs of the latch.

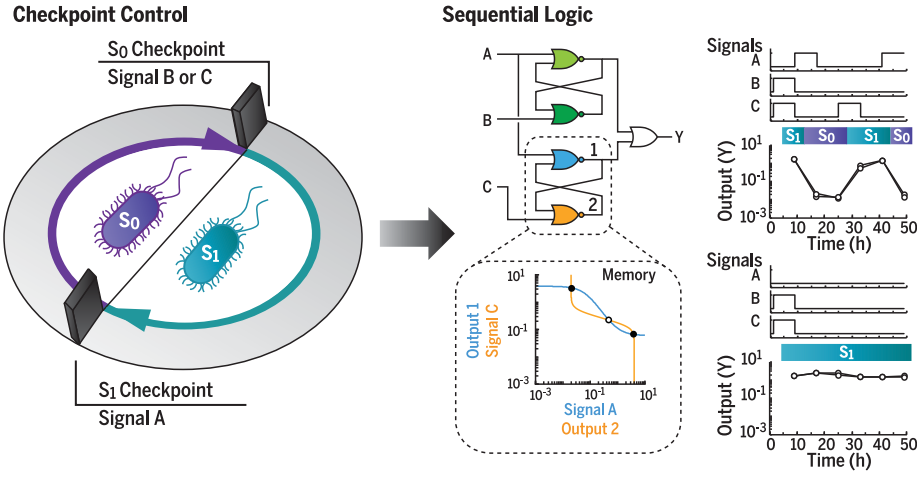
ON OUR WEBSITE

Read the full article at http://dx.doi.org/10.1126/science.aap8987

Larger circuits are constructed by combining multiple SR latches and additional feedback loops. A gated data (D) latch, common in electronic integrated circuits, is constructed where one input sets the state of the circuit and the second input locks this state. Up to three SR latches (based on six repressors) are combined in a single cell, thus allowing three bits to be reversibly stored. The performances of these circuits closely match those predicted by the responses of the component gates and a bifurcation analysis.

Circuits are designed to implement checkpoint control, in which cells wait indefinitely in a state until the correct signals are received to progress to the next state. The progression can be designed to be cyclical, analogous to cell cycle phases, during which cells progress through a series of states until returning to the starting state. The length of time in each state is indefinite, which is confirmed by demonstrating stability for days when the checkpoint conditions are not met.

CONCLUSION: This work demonstrates the implementation of sequential logic circuits in cells by combining reliable units of regulation according to simple rules. This approach is conducive to design automation software, which can use these rules to combine gates to build larger circuits. This provides a designable path to building regulatory networks with feedback loops, critical to many cellular functions and ubiquitous in natural networks. This represents a critical step toward performing advanced computing inside of cells. ■



Quantitative design of sequential logic in living cells. Cells can be genetically programmed to respond to temporal stimuli by using complex sequential logic circuits. (Left) Checkpoint control is one such example in which the circuit state (s_0 and s_1) transitions when the specified input signals are presented. (Middle) Sequential logic circuits can be designed from simple steady-state response functions measured in relative promoter units by using principles of nonlinear dynamics. Bistable latches are used as rewritable memory. The colored symbols represent gates. (Right) The circuit output (Y) was measured for cells that were grown in inputs that were varied over time. The square waveforms indicate the presence or absence of the input signals. Over multiple days, the cells can be cycled through the circuit states or held waiting for the next checkpoint.

The list of author affiliations is available in the full article online.
*Corresponding author. Email: cavoigt@gmail.com
Cite this article as L. B. Andrews *et al.*, *Science* **361**, eaap8987 (2018). DOI: 10.1126/science.aap8987

RESEARCH ARTICLE

SYNTHETIC BIOLOGY

Cellular checkpoint control using programmable sequential logic

Lauren B. Andrews[1,2]*, Alec A. K. Nielsen[2], Christopher A. Voigt[1,2]†

Biological processes that require orderly progression, such as growth and differentiation, proceed via regulatory checkpoints where the cell waits for signals before continuing to the next state. Implementing such control would allow genetic engineers to divide complex tasks into stages. We present genetic circuits that encode sequential logic to instruct *Escherichia coli* to proceed through a linear or cyclical sequence of states. These are built with 11 set-reset latches, designed with repressor-based NOR gates, which can connect to each other and sensors. The performance of circuits with up to three latches and four sensors, including a gated D latch, closely match predictions made by using nonlinear dynamics. Checkpoint control is demonstrated by switching cells between multiple circuit states in response to external signals over days.

The most complex processes in biology use checkpoint control to synchronize cells in the face of noisy and asynchronous conditions (*1*). Checkpoints are implemented between stages of a cascade via a gene regulatory network that integrates environmental and cellular information before triggering a switch instructing the cell to proceed. Checkpoints are ubiquitous across eukaryotic and prokaryotic processes, such as the cell cycle (*2*–*4*), cell development pathways (*5*, *6*), inflammation (*7*), stress response (*8*–*11*), pathogenesis (*12*), and the assembly of molecular machines (*13*). From the perspective of cellular engineering, a generalizable method for implementing checkpoint control would allow the division of complex tasks into stages from which cells only proceed when the proper conditions are met (nutrient or precursor availability, cell-cell signaling, environmental niche recognition, etc.).

A progression of events can be implemented via a cascade in which one transcription factor regulates the next. Cycles are formed when the last transcription factor regulates the first in the cascade. To study these regulatory features in isolation, synthetic cascades and cycles have been constructed with heterologous transcription factors (*14*–*18*). Once initiated, the cascades and cycles progress continuously through each transcription factor in series, either concluding at the end of a cascade or oscillating via a stable limit cycle.

By contrast, checkpoints use bistable switches to control the transitions between discrete stages. This allows the cell to stop and wait until signals that regulate progression cross a threshold. When multiple environmental and cellular signals are required, they can be integrated by regulation encoding Boolean logic operations (AND, OR, etc.) (*19*). A bistable switch introduces hysteresis into the transition, which stops the cells from reversing to the prior state when the stimuli are removed, and it waits for the necessary signals to progress forward (*20*–*22*). Checkpoint control leads to variability in the time spent in each stage but synchronizes the requirements for progression across cells and buffers against fluctuations (*1*). The switch acts as a form of memory to keep track of the cell state during this period.

In electronics, circuits containing logic and rewritable memory are referred to as "sequential logic." Nearly universal across microprocessors, sequential logic is necessary to control the timing and order of processing steps within algorithms (*23*). Information is stored regarding the past state of the circuit, and this is integrated with input signals to determine the circuit's next state transition. The input signals are integrated by "combinational logic," whose output is only a function of the inputs. Latches are analogous to bistable switches and have two stable states that are used to store one digital bit of information. The simplest is a set-reset (SR) latch, composed of cross-coupled NOR gates that integrate two inputs to switch the latch between two states, which are read by two outputs. This architecture suffers from having a forbidden state (both inputs on), which can lead to instabilities in the circuit due to timing effects. More complex latches store a bit of information without having a forbidden combination of inputs. A common example is the data (D) latch used in microprocessors for temporary data storage and to synchronize asynchronous signals (*24*, *25*).

Synthetic genetic circuits have been built that encode the core functions required for sequential logic. Combinational logic has been implemented inside cells through a variety of biochemical mechanisms (*26*–*32*). Here we focus on logic gates that are implemented on the level of transcription, in which the inputs and outputs of the gates are promoters regulated by transcription factors. This defines RNA polymerase (RNAP) flux as the common signal carrier, which facilitates the implementation of more complex circuits by layering simpler gates (*33*). It also simplifies the connection of genetic sensors to the circuit when the sensor responds to a stimulus by activating a promoter. Following this paradigm, we have shown that a library of repressor-based NOR gates can be rationally connected in different permutations to apply arbitrary combinational logic operations to up to four sensors (*34*). Central to this approach are the empirical gate response functions (how the output changes as a function of input at steady state) that are used to computationally identify those gates that can be functionally connected.

Rewritable memory has also been implemented as a transcriptional circuit (*35*–*39*). One implementation is a "genetic toggle switch" that consists of a pair of repressors that regulate each other's promoter (*40*–*44*). The resulting cross-repression leads to the bistability required for an SR latch. The inputs of the original implementation (*40*) are the small molecule inducers of the repressors, which complicates the connection of latches to each other, to combinational logic, or to sensors. The toggle switch has been connected to a single sensor by having the output promoter drive the expression of one repressor, and this has been used to remember transient exposure to a sugar, quorum signal, or an antibiotic (*45*–*47*). SR latches with two promoter inputs have been built based on DNA-inverting enzymes and sigma: anti-sigma factor sequestration, but we avoided these architectures because of toxicity, a dearth of orthogonal variants, or the absence of hysteresis (*35*, *48*, *49*). No more than a single SR latch has been shown to operate in a cell. More complex switches, such as the D latch, have been proposed and studied computationally but have not been implemented (*50*–*53*).

Here we present a theory-guided approach to building complex sequential logic circuits. Repressor-based NOR gates are connected to each other and sensors by signal matching their response functions. Latches are designed by recognizing that these empirical functions can serve as nullclines to identify gate combinations that will exhibit bistability (*54*). Further, the latch quality can be predicted quantitatively by the separation of the stable and unstable steady states. This is used to design 11 SR latches, a D latch, and circuits with up to three latches in a single cell. We demonstrate sequential logic that encodes a series of circuit states, separated by checkpoints, where progression is controlled by up to four sensors responding to different signals.

Design of SR latches

Bistability is a necessary criterion for building an SR latch. This is achieved by arranging two repressors to regulate each other's expression. For

[1]Broad Institute of MIT and Harvard, Cambridge, MA 02142, USA. [2]Synthetic Biology Center, Department of Biological Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA.
*Present address: Department of Chemical Engineering, University of Massachusetts, Amherst, MA 01003, USA.
†Corresponding author. Email: cavoigt@gmail.com

the latch to be extensible, it must have two promoter inputs and two promoter outputs. If these are characterized with the same units, the information can be used to connect the latch to genetic sensors (Fig. 1A), other circuits, and cellular responses. Previously, we characterized insulated NOR gates that have two input promoters in series driving the expression of a repressor that turns the output promoter off (*55*, *56*). Because the promoters are arranged in series, their activity is summed, and the gate response function is treated as a single-input NOT gate (Fig. 1B). Our latch design uses this gate type, in which one of the NOR inputs is the set input promoter (S) to the latch and the other is a promoter regulated by a second NOR gate (Fig. 1C). In turn, the two inputs to the second NOR gate are the reset input promoter (R) to the latch and a promoter repressed by the first gate.

The response function of a NOT gate captures how the gate's output changes as a function of its input at steady state. Because the input and output promoter activities of the gates are measured in relative promoter units (RPUs) (see

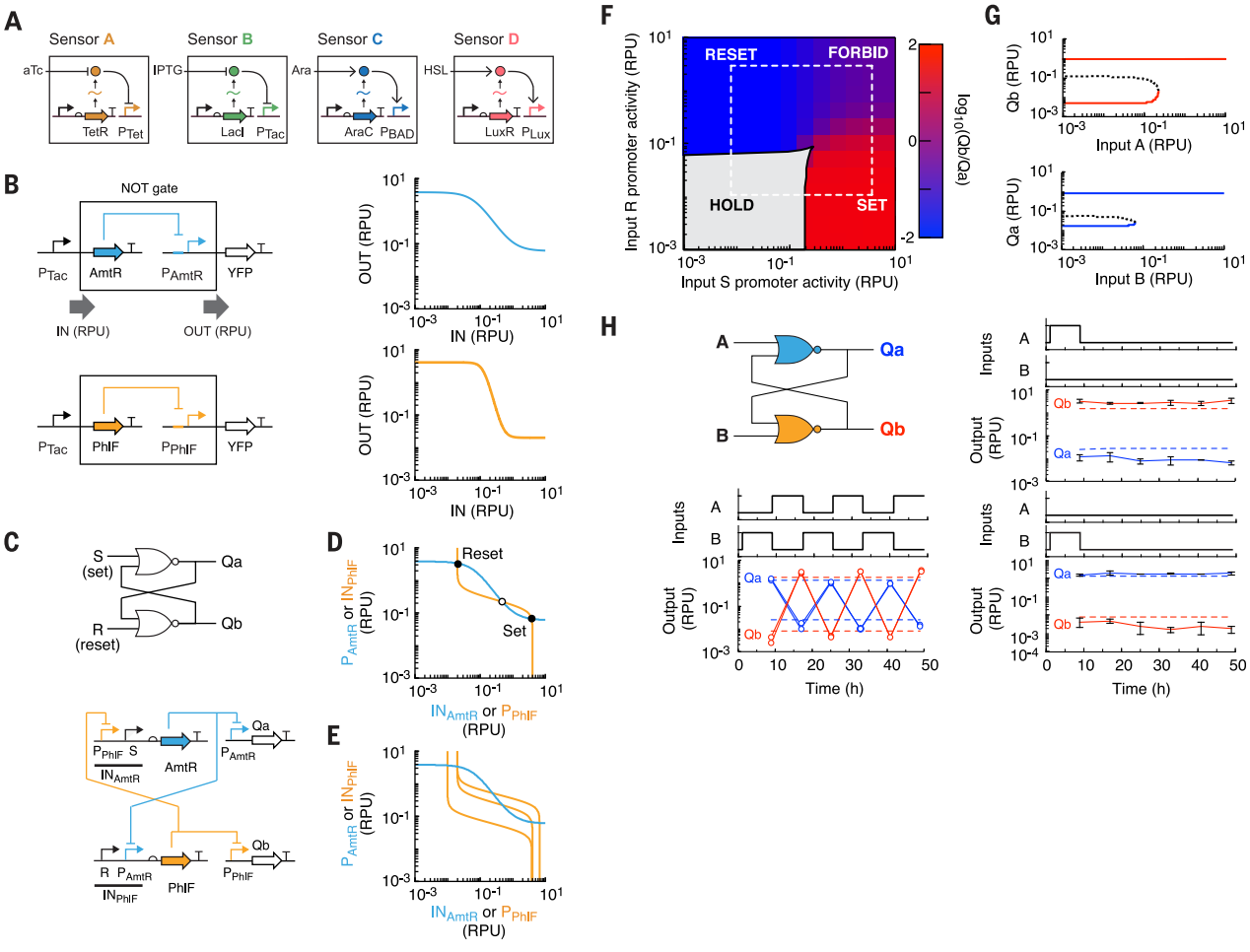


Fig. 1. Design of a bistable SR latch. (**A**) The genetic sensors (A, B, C, and D) used in this work are shown. The one-letter abbreviations are used consistently in the circuit diagrams shown in the figures. The sensor output promoters serve as the inputs to the circuits. P, promoter; Ara, arabinose; HSL, *N*-(3-oxohexanoyl)-L-homoserine lactone. (**B**) Two repressor-based NOT gates and their corresponding empirical response functions are shown (table S1) (*34*). The promoter activities, measured in RPU, are used as indirect measures of the RNAP flux (gray arrow) into and out of the gates. (**C**) The wiring diagram (top) and genetic implementation (bottom) of an SR latch is shown. IN_{AmtR} and IN_{PhlF} are the total (summed) input promoter activities into each gate. The output promoter activities (Qa and Qb) are measured with yellow fluorescent protein (white gene arrows) and two reporter plasmids. (**D**) A phase plane analysis is performed by plotting the response functions from (B) as nullclines (supplementary materials). The intersections are stable and unstable steady states, shown as open and solid circles, respectively. (**E**) The orange lines represent the nullclines for three strengths of RBSs controlling the expression of the PhlF repressor that alter the threshold of the response function and the number of intersections (multistability). (**F**) A bifurcation analysis is shown for the AmtR-PhlF SR latch. This was computed using an ODE model based on the NOT gate response functions and AUTO software (supplementary methods). The bistable region (gray) indicates memory by holding the latch state (HOLD). The SET, RESET, and FORBIDDEN states are contained in the monostable regions (color). The dashed lines correspond to the OFF and ON outputs of sensors A and B (input S and input R, respectively) at the inducer concentrations used in the experiments. The model is provided in Systems Biology Markup Language (SBML) (supplementary file) (*89*). (**G**) Hysteresis plots are shown for the transitions between HOLD and SET (top) or HOLD and RESET (bottom). Stable and unstable steady states are indicated as solid and dashed lines, respectively. The transitions correspond to the white dashed lines in (F). (**H**) Experiments validating the AmtR-PhlF SR latch are shown. The blue and yellow symbols are gates. The methods are described in the text and supplementary materials. The input waveforms correspond to no inducer when low and either 2 ng/ml aTc (sensor A) or 1 mM IPTG (sensor B) when high. The model-predicted outputs are shown as dashed lines. Data points and error bars for the graphs on the right are shown as the average and standard deviation of three replicates performed on different days. The switching experiments to the left show two trajectories performed on different days. Additional experiments with additional input waveforms are provided in fig. S3.

methods), the two NOT functions can be plotted as nullclines on the same phase plane (Fig. 1D) (see the supplementary materials for the biophysical model) (*54*). The circuit is monostable when the nullclines only intersect once, meaning it does not exhibit the bistability necessary for a latch. When there are three intersections, the latch has two stable states separated by an unstable steady state. An exemplar is shown in Fig. 1C, where gates based on two repressors (AmtR and PhlF) are combined to design an SR latch. These gates use repressors that are Tet repressor (TetR) homologs and orthogonal; in other words, they do not bind to each other's operators (*34*, *55*). The ribosome binding site (RBS) that controls repressor expression can be used to change the threshold of a gate. Thus, by changing the RBS, the response function can be shifted in relation to the second gate in the latch until bistability is achieved (Fig. 1E) (*57*).

A full bifurcation analysis was performed for the AmtR-PhlF SR latch from an ordinary differential equation (ODE) model (methods and supplementary files). This demarcates the boundaries of the SET, RESET, and HOLD regions within phase space (Fig. 1F). Note that there is also a FORBIDDEN region corresponding to the simultaneous activation of both inputs where the output signals degrade to an intermediate value. This bifurcation graph fully characterizes the SR latch and can be used to quantitatively predict the response obtained from connecting any two sensors, provided that their dynamic range is known in RPUs. Each sensor performs the SET and RESET functions and alternating their activity switches the latch repeatedly between the two stable states. Once in either state, the reduction of the input (e.g., input A when in the SET state) is not able to switch the latch state without turning on the other input. Thus, this acts as memory, where the state is held indefinitely (Fig. 1G) (*58*, *59*).

The AmtR-PhlF SR latch design was constructed by using a two-plasmid system, one carrying the circuit and the other carrying an output promoter fused to yellow fluorescent protein (YFP) (fig. S1). The sensors that respond to anhydrotetracycline hydrochloride (aTc) (sensor A) and isopropyl-β-D-1-thiogalactopyranoside (IPTG) (sensor B) were used as inputs (Fig. 1A). Two output plasmids were constructed that independently report the Qa and Qb outputs (separate strains and experiments). Cells containing the circuit and one output plasmid were initialized by growing them to exponential phase in the absence of inducer and then diluting them into fresh media with either 2 ng/ml aTc or 1 mM IPTG (methods). Every 8 hours, the samples were analyzed by flow cytometry to measure YFP, and an aliquot was diluted into fresh media containing the same or different inducers. Figure 1H shows the response of the outputs of the circuit to changes in the activity of the sensors, shown as the square waveform. By alternating between inducers, the circuit can be switched back and forth repeatedly for 2 days and more than 60 generations without breaking. When one inducer is pulsed, the circuits hold their state over this period. Notably, the model quantitatively predicts the level of both outputs (dashed lines in Fig. 1H), including when the circuit is in an intermediate FORBIDDEN state and the steady state that is reached after the inducers are removed (fig. S4).

A library of SR latches was then designed by combining gates based on 10 orthogonal repressors from the TetR repressor family that cooperatively bind their respective operator DNA sequence (*34*, *55*). Each gate has a different response function that, when combined in a phase plane analysis, can be analyzed for bistability (Fig. 2A). Forming a bistable switch requires that nullclines intersect at three points. Of the 45 combinations, 23 are predicted to exhibit bistability (blue boxes in Fig. 2A); however, not all are expected to behave equally. In some cases, the distance between the stable and unstable steady states (d_1 and d_2 in Fig. 2B), referred to as the equilibria separation, is small (*60*). Another measure is transversality or, in other words, the degree to which the nullclines do not overlap. Switches with short equilibria separation and poor transversality are sensitive to fluctuations, which can drive the switch into the opposite state (fig. S5). As expected, cooperative gates with steeper thresholds are more likely to lead to bistable switches; for example, the PhlF gate (Hill coefficient n = 4.2) is predicted to form a bistable switch with all other repressors. The cooperativity of each repressor was measured empirically in the context of the gate. Cooperativity could arise from the formation of multimers and/or the binding to multiple sites in a promoter.

We built the 15 SR latches that were predicted to be bistable, including some with short separation and poor transversality (Fig. 2C). These were constructed and evaluated, as described above (methods). Each latch was initialized in the first state (SET or RESET) by growing cells with the appropriate inducer. To test the latch's memory, the cells were then grown in the absence of inducer (HOLD), and, after 8 hours, the outputs were assayed by flow cytometry. The measurements are compared to the predicted outputs in Fig. 2D. Eleven of the SR latches exhibited at least a 10-fold dynamic range and held both latch states. Four had one state that spontaneously flipped during the memory assay (red arrows in fig. S5), which could be predicted from the equilibrium separation (Fig. 2E). Intriguingly, the cutoff separation corresponds to the standard deviation of the cytometry population when converted to RPU (fig. S2). For functional SR latches, the average dynamic range is 162-fold repression.

An additional 26 SR latch circuits were constructed by permuting the sensors connected to the latches and RBS variants of the component gates. Because the sensors are measured in standardized units, in theory, it can easily be determined whether they can be functionally connected to a particular latch (Fig. 1F). For the AmtR-HlyIIR SR latch, six permutations of sensors were tested, and all were able to functionally connect as predicted (Fig. 2F). This was repeated for different permutations of sensors connected to latches comprised of different pairs of repressors (fig. S5). Out of this set of 26 SR latches, 19 latches were functional, and the measured outputs agreed with the predictions (fig. S6A). The latches that did not hold both states correlated with poor predicted equilibria separation (fig. S6B).

Genetic D latch and memory registers

We sought to construct larger circuits containing multiple latches and feedback using the principles of signal matching and nonlinear dynamics (*54*). First, we designed a gated D latch, which has two inputs corresponding to the set (S) and lock (L) inputs (Fig. 3A). The latch state is determined by whether S is low or high when the lock is off. When locked, the state of the latch is stored irrespective of the input S. In contrast to the SR latch, this architecture has no FORBIDDEN state and associated instabilities. As such, it is commonly used in electronics to synchronize multiple latches to record at the same time through a shared clock signal that serves as the L input.

The D latch contains four NOR gates, two of which are cross-coupled to introduce positive feedback that results in cross-repression (Fig. 3A). The other two gates provide the upstream logic to control locking and unlocking of the latch and relaying the input S. Signal matching was performed to assign repressors to each gate by using a modified version of the Cello software (*34*) that used a hill-climbing algorithm to identify an assignment with the designed response by simulating all state transitions and applying a circuit score cutoff of at least 10-fold for valid assignments (methods). After selecting the repressors for each gate, a bifurcation analysis was performed by using the corresponding model to determine how the latch switches between states (Fig. 3B) (supplementary materials). The activation of the S input switches between the outputs (Qa and Qb) that are active. When the lock is off, the transition between the two is bistable. The bistable region expands as the lock signal increases, until it encompasses the entire range of S, thus storing the bit of memory. It is noteworthy that the bifurcation diagram differs substantially from that of the SR latch (Fig. 1F), and this complex behavior can be predicted by using the simple gate response functions.

The design for the D latch was constructed with the aTc-inducible sensor A serving as the S signal and the arabinose-inducible sensor C serving as the lock L (Fig. 3C). The circuit was constructed by using the same two-plasmid system as before, with two versions of the output plasmid to measure Qa and Qb by using separate strains. The D latch was evaluated by the same protocol as the SR latches, except no inducer was used during the initialization step (methods). The transitions between all combinations of states were evaluated, and they closely matched the predicted outputs (Fig. 3, D and E, and fig. S9). The D latch switched states over six cycles, correctly recorded in response to the lock signal, and maintained the memory states for longer than 2 days.

Multibit rewritable memory registers were then constructed by combining multiple parallel

SR latches in a circuit (Fig. 3F). To operate together in a cell, each latch requires two repressors orthogonal to the others in the circuit. From the set of 11 functional SR latches (Fig. 2C), up to three are fully orthogonal to each other. Based on this set, we constructed five circuits containing different combinations of two latches (figs. S10 and S11) and one containing three latches (Fig. 3F). The quantitative response of all the outputs to changes in the activity of the three sensors was predicted by extending the approach used for individual SR latches. A phase plane analysis was performed on the basis of an ODE model parameterized by the response functions of the repressor-based gates (supplementary materials).

Three sensors for IPTG, aTc, and arabinose serve as the inputs to multiple latches, following an architecture similar to memory-register design in electronics. The construction of the circuit is based on the same two-plasmid system as before, except that there are six possible output plasmids so that the circuit outputs (Qa, Qb, Qc, Qd, Qe, and Qf) can be measured by using separate strains

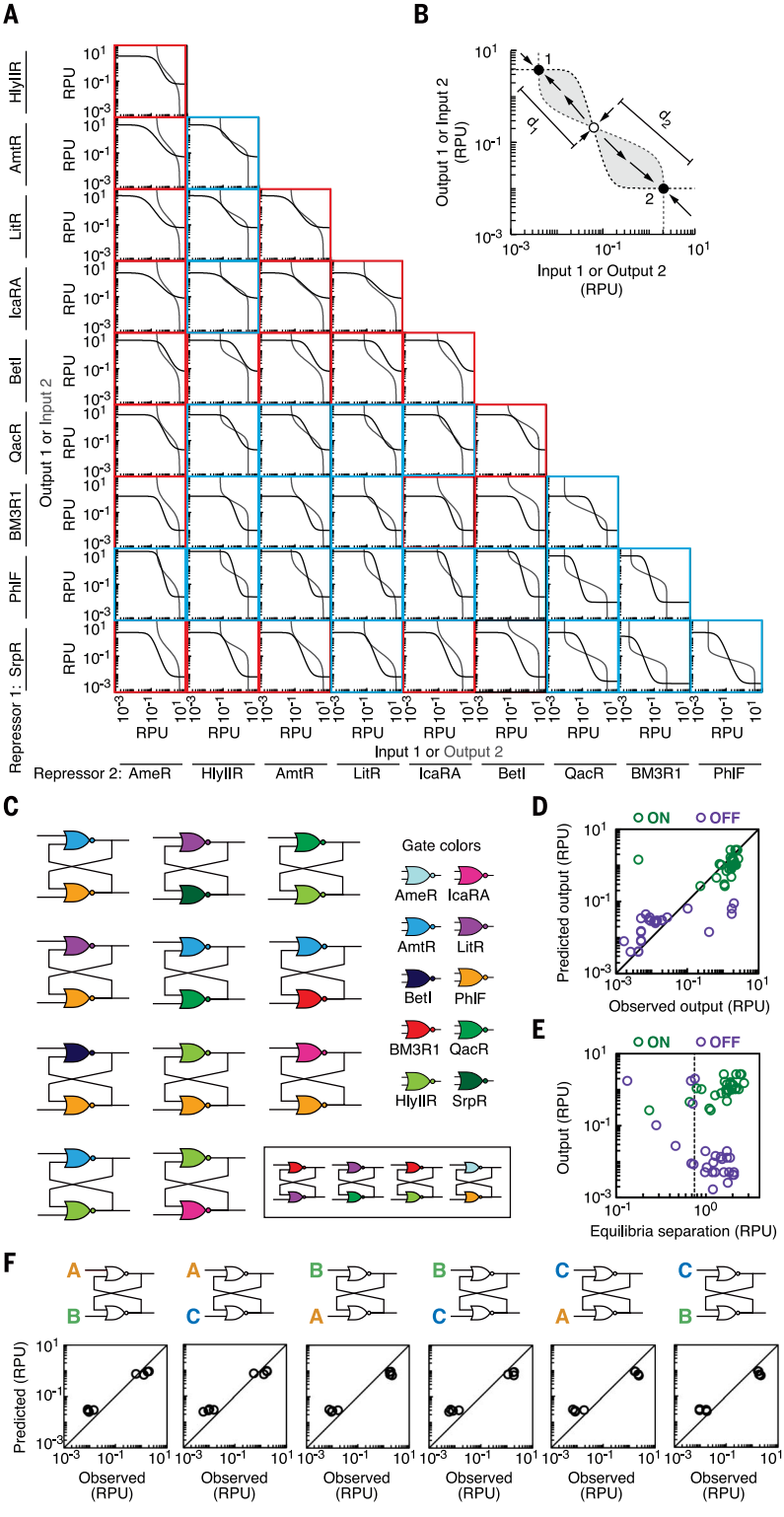


Fig. 2. Connecting gates to build SR latches. (**A**) Phase planes for all combinations of 10 repressors are shown. The nullclines are the empirical response functions measured previously (*34*). Repressor pairs predicted to be monostable or bistable are boxed in red or blue, respectively. For some gates, multiple RBS variants are available. In these cases, we show the pair of gates that produce the greatest transversality, calculated as the area between the nullclines. (**B**) The phase plane analysis is shown for the SrpR and PhlF repressors. The equilibria separation is given by the distance in phase space of the stable state from the unstable state (d_1 and d_2), and the transversality is shaded in gray. (**C**) The wiring diagrams are shown for the set of 11 SR latches that were constructed and validated experimentally. The box surrounds four additional combinations found to be nonfunctional. Sensors A and B were used in these experiments to assay the SR latch library. Colors correspond to the repressor of each gate. (**D**) Quantitative comparison of the predicted and measured output (both Qa and Qb) for the set of 15 latches, as determined by phase plane analysis. The black line is $x = y$. For each SR latch, both outputs for each latch state are plotted after holding the state (absence of either input). Experimental measurements were performed in duplicate, and the mean observed value is plotted (fig. S5). Each output promoter was measured in the ON (green) and OFF (purple) state. The four SR latches that spontaneously switched are marked with red arrows in fig. S5, and sensor permutations were also tested for these SR latches. (**E**) The observed outputs in the ON (green) and OFF (purple) states are plotted against the corresponding equilibria separation for the state [d_1 or d_2 in (B)] for the SR latch library in (C). The dashed line is the mean standard deviation of the cytometry distributions when the output is ON (fig. S2). (**F**) Six permutations of sensors A, B, and C were connected to the AmtR-HlyIIR SR latch. The dynamic range of the sensors in the presence and absence of inducer was used to compute the ON and OFF states of the latch, and these states were compared to those determined experimentally. The observed output data points are the average of duplicate experiments. The black lines are $x = y$.

(Fig. 3G). Growth and induction were performed as before. All six outputs were found to be quantitatively correct for input pulses of each sensor (Fig. 3F). Different patterns of square waveforms for each input were tested over 2 days, and the circuit performed quantitatively as predicted, except for one trajectory after 2 days (Fig. 3H). After a single pulse, the memory could be held over a day, after which at least one output became unstable (fig. S11). We also tested five simpler circuits containing different combinations of two SR latches and sensors, and these were found to function quantitatively as predicted (figs. S10 and S11).

Designing checkpoint control

Sequential logic circuits were designed to implement checkpoint control. This requires that genetic circuits remain in their current state until they receive the necessary sensory inputs to transition to the next state. In computer science, this is referred to as a finite-state machine, which is defined as being able to exist in one of a finite number of circuit states at a given time and where external inputs control state transitions. In this context, a state diagram is used to enumerate the states (s_0, s_1, ... s_i), and allowed transitions are

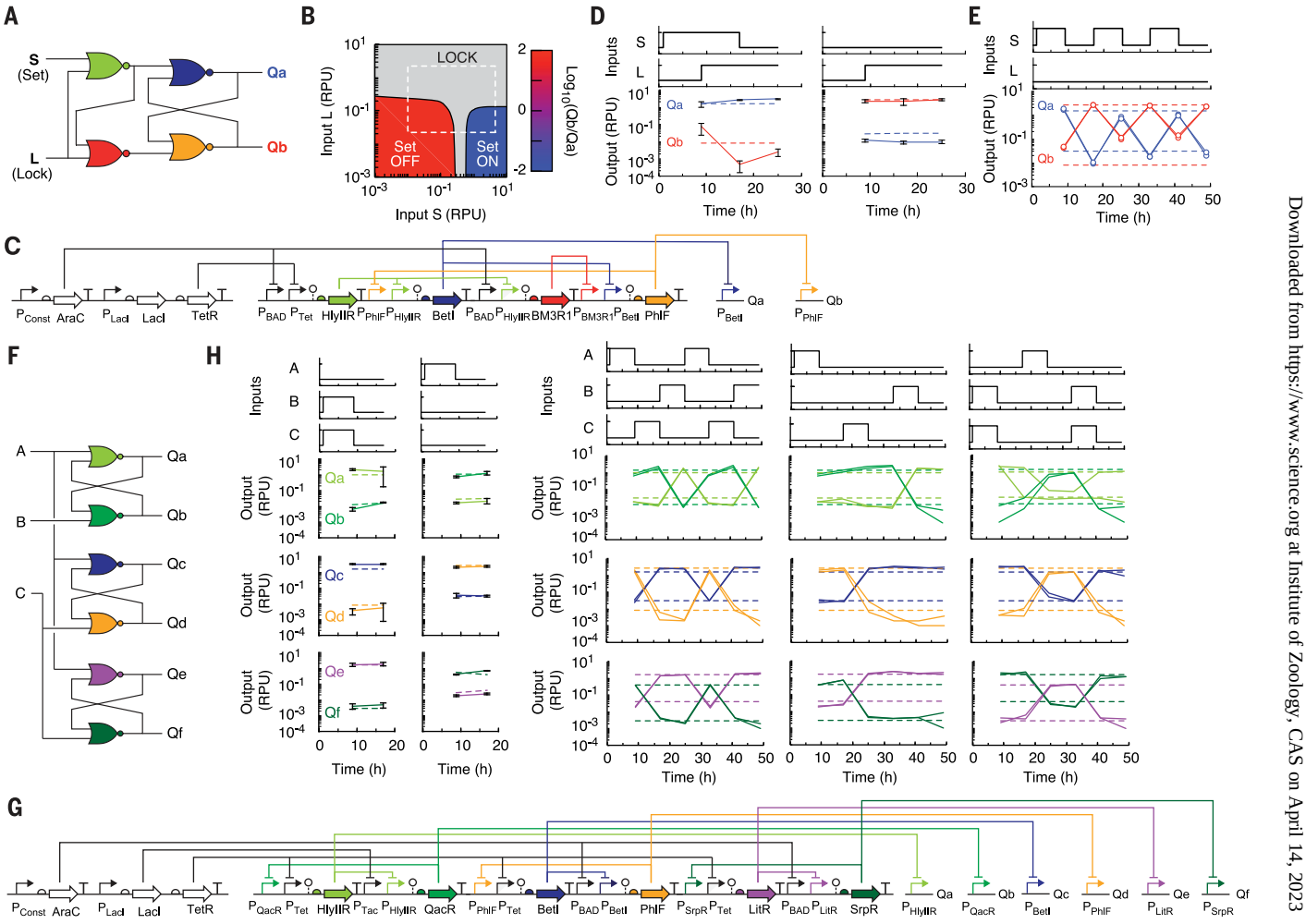


Fig. 3. Circuits comprising multiple latches. (**A**) The D latch circuit wiring diagram. Colors correspond to the repressor assigned to each gate (see Fig. 2C). (**B**) Bifurcation analysis of the D latch from an ODE model (supplementary materials). Bistable regions are shown in gray, and the monostable regions are colored according to the ratio of the two outputs (Qb/Qa). The white dashed lines mark the ON and OFF inputs used in the experiments, and sensors A and C are connected to the S and L inputs, respectively. Corresponding phase plane analysis and hysteresis curves are provided in fig. S8. (**C**) Genetic diagram of the D latch, including sensors, gates, and outputs. The two output promoters (Qa and Qb) were fused to *yfp* and carried on a separate plasmid. (**D**) Experiments verifying the performance of the D latch. The waveforms correspond to the presence and absence of inducers for the two sensors (sensor A is 2 ng/ml aTc, and sensor C is 5 mM L-arabinose). The dashed lines show the predicted outputs from a steady-state model. The data points were calculated from three replicates performed on different days, and the error bars are the standard deviation. (**E**) Cells were repeatedly switched between states by adding and removing the set signal (aTc) over 2 days. No arabinose was added to the media, keeping the lock in the OFF state. Two trajectories are shown, which were performed on different days. The responses to additional waveforms are shown in fig. S9. (**F**) A memory register was constructed from three SR latches connected to sensors A, B, and C. Colors correspond to the repressor assigned to each gate (see Fig. 2C). (**G**) Genetic diagram for the three-latch memory register. Six output plasmids were constructed to measure the Qa, Qb, Qc, Qd, Qe, and Qf outputs in separate experiments. (**H**) The responses of the three-latch memory register to different input waveforms are shown. The sensor inputs A, B, and C were 2 ng/ml aTc, 1 mM IPTG, and 5 mM L-arabinose, respectively. The dashed lines are the steady-state levels predicted by the model (supplementary materials). (Left graphs) The ability to hold a state (memory) is shown for combinations of inputs. Each point represents an average and standard deviation calculated from three experiments performed on different days. (Right graphs) The ability to hold and switch states over 2 days is shown. The lines show the trajectories for two experiments performed on different days, and the colors indicate the output (Qa to Qf) and corresponding output promoter and repressor. Additional experiments showing the circuit response to other waveforms are shown in fig. S11.

indicated by arrows that are labeled by the status of the inputs required (sensors A, B, C, and D) and the state of the output (Y) during the transition, assuming an instantaneous response (Fig. 4). The current state is maintained until signals are received to transition to a different state.

Three circuits were designed to have two, three, and four states, respectively. They are all based on a core circuit containing two SR latches whose outputs are integrated by a final gate that implements OR, NOR, or three-input NOR logic. The two-latch register can exist in four possible states (2^2); however, the state diagram can be simplified by collapsing equivalent states that have the same output and transitions. For each circuit, repressors were assigned to the gates via signal matching, and the DNA sequence was designed and constructed as before (methods).

The circuit shown in Fig. 4A has two states (s_0 and s_1). The checkpoint for the $s_1 \rightarrow s_0$ transition is controlled by sensor A (aTc). The checkpoint for the $s_0 \rightarrow s_1$ transition is controlled by the logic B OR C (the presence of either IPTG or arabinose). Pulses of the inducer(s) switch the cells between states (Fig. 4A) (methods). The cells can be made to oscillate between the circuit states by adding the appropriate inducers back and forth between states. However, unlike a limit cycle oscillator, the cells remain in the current state in the absence of the checkpoint signals or when inputs are pulsed that do not participate in the checkpoint (see Fig. 4A and fig. S12 for additional trajectories). We demonstrate that the states can be held stably for days.

The addition of a NOR gate to the circuit expands the number of states to three (s_0, s_1, and s_2) (Fig. 4B). This can be viewed as a cycle in which the transitions between states are controlled by checkpoints: $s_0 \rightarrow s_1$ (arabinose), $s_1 \rightarrow s_2$ (IPTG), and $s_2 \rightarrow s_0$ (aTc). The cycle cannot proceed in the reverse direction because there is no transition from s_2 returning to s_1. All state transitions showed the predicted circuit output for all experiments, including repeatable switching and stable memory propagation (Fig. 4B and fig. S13).

To increase the number of circuit states to four, an additional sensor (sensor D) that responds to acyl-homoserine lactone and a three-input NOR gate were added (Fig. 4C). The three-input NOR gate was constructed by placing a third input promoter upstream of a NOR gate (fig. S15). The resulting state diagram is complex and cannot be reduced to a simple cycle. The SR latches act to store the past history of three inputs (A, B, and C), and A serves as the shared reset signal for the register. The circuit was tested with pulses of each input, and the output response behaved as expected over 2 days (Fig. 4C and fig. S14).

Discussion

Natural regulatory networks are rife with complex interconnected feedback loops that implement multistable switches and dynamical behaviors (*61*). For engineering purposes, it would be valuable to be able to recreate these functions (*62*). To be designable, the approach must be based on reliable units of regulation that can be combined by following simple rules. Here we demonstrate the core functions of sequential logic that are central to computation and representative of the simplest units of feedback control. Surprisingly, we can predict how to build complex multilatch circuits with feedback using only the steady-state response functions of the underlying NOT gates. These functions are simple to measure and report in standard promoter activity units. Circuit-design software can use these functions to automate the combination of gates to build user-defined sequential logic. Now, this is specified by using structural Verilog to define the desired circuit (supplementary files). However, this work provides the groundwork for implementing higher-level tools that will allow a user to specify the desired sequential logic or finite-state machine behavior with a state table or state diagram (*63*–*65*).

This designability is not intrinsic to biology, but rather comes from intense prior engineering efforts to design gates that are modular and insulated, such that the underlying function is context independent (*34*, *56*, *66*). These gates need to have cooperative thresholds to build circuits that exhibit multistability and complex dynamics. Our gates are based on TetR homologs that exhibit cooperativity, and we have found that having at least one gate with a Hill coefficient $n \geq 2.4$ is required to consistently build a robust bistable switch. By contrast, other means of repression, including transcription activator–like (TAL) effector proteins and CRISPR interference (CRISPRi), offer the possibility of many orthogonal gates but are noncooperative ($n \cong 1$) (*67*–*69*). Incorporating positive feedback into the latch design is a means by which the bistable region could be expanded (*43*, *70*–*72*), but this would complicate design automation.

There are many potential applications for programmable sequential logic that can be modularly connected to sensors. The sensors in this work respond to small molecules, but others could respond to diverse signals, such as dissolved gases, light or color, environmental stresses, and temperature (*73*). The outputs of the circuits are also modular and can be used to control metabolic pathways (*74*), regulate host gene expression using CRISPRi (*68*), or implement permanent memory to irreversibly record the state (*75*). This work demonstrates the programmable ability to implement checkpoint control as a state machine. Importantly, this is not an irreversible progression for a sequence of signals; the cycle of states can be repeated in a single cell and its progeny. This enables the design of cycles and developmental networks for engineering applications that require that cells exist in a particular state for an unspecified amount of time. For example, therapeutic cells could be built to sense transient stimuli, such as throughout the gastrointestinal tract, and switch to a new state when the next signal is encountered. There are similar applications for diagnostic cells (*48*, *76*–*81*), pathways to complex chemicals and materials that require cycles of ordered operations (*82*), and sentinel plants and microbes with responsive traits (*31*, *83*, *84*).

Materials and methods

Strains, media, and inducers

E. coli NEB 10-beta [*Δ(ara-leu) 7697 araD139 fhuA ΔlacX74 galK16 galE15 e14-ϕ80dlacZΔM15 recA1 relA1 endA1 nupG rpsL* (Str^R) *rph spoT1 Δ(mrr-hsdRMS-mcrBC)*], which is a derivative of *E. coli* DH10B (*85*), was used for experimentally measuring circuits. *E. coli* was cultured in LB Miller medium (Sigma-Aldrich, L3152) for routine cloning and propagation of plasmids. Genetic circuits were assayed in M9 minimal media (Sigma-Aldrich, M6030; 6.78 g/liter Na_2HPO_4, 3 g/liter KH_2PO_4, 1 g/liter NH_4Cl, 0.5 g/liter NaCl final concentration) with 0.34 g/liter thiamine hydrochloride (Sigma-Aldrich, T4625), 0.2% Casamino acids (Acros, AC61204), 2 mM $MgSO_4$ (Sigma-Aldrich, 230391), 0.1 mM $CaCl_2$ (Sigma-Aldrich, 449709), and 0.4% D-glucose (Sigma-Aldrich, G8270). Antibiotics used to select for circuit plasmids were 100 μg/ml carbenicillin (Corning, 46-100), 50 μg/ml kanamycin (Sigma-Aldrich, K4000), and 50 μg/ml spectinomycin (Sigma-Aldrich, S9007). The inputs used for the sensor promoters were isopropyl β-D-1-thiogalactopyranoside (IPTG; Sigma-Aldrich, I6758), anhydrotetracycline hydrochloride (aTc; Sigma-Aldrich, 37919), L-arabinose (L-ara; Sigma-Aldrich, A3256), and *N*-(3-oxohexanoyl)-L-homoserine lactone (3OC6-HSL; Sigma-Aldrich, K3007).

Genetic circuit assembly

Genetic circuits were built using the repressor gates, three-input circuit backbone, and output plasmid as in the Eco1C1G1T1 UCF file (*34*). Genetic circuits were constructed by hierarchical Type IIS assembly in two sequential assembly reactions (fig. S1B). In the first assembly reaction, transcriptional unit constructs were assembled by joining the input promoter fragment(s) to the fragment containing the insulated repressor (ribozyme, RBS, repressor gene, terminator). A BsaI Type IIS Assembly reaction was performed for this step. The promoter and repressor fragments were supplied to the reaction as purified plasmid DNA, and the destination vector was supplied as a purified PCR product. These Type IIS assembly reactions were performed in 5 μl total volume containing 20 fmol of each purified part plasmid, 10 fmol of the purified destination vector PCR product, 5 U BsaI enzyme (New England Biolabs, R0535), and 2.5 U T4 DNA ligase (20 U/μl HC; Promega, M1794) in 1X T4 DNA Ligase Buffer (Promega). The reaction mixture was incubated in a thermal cycler (Bio-Rad C1000 thermal cycler, 80°C lid) with the protocol: 37°C for 5 hours, followed by 50°C for 30 min, and inactivated at 80°C for 10 min. Then, 1 μl of the assembly reaction was transformed into 5 μl chemically competent cells (*E. coli* NEB 5-alpha, New England Biolabs, C2988J). After recovery in SOC, the cells were plated on LB agar containing carbenicillin. Colonies were grown, and the constructed transcriptional unit plasmids were sequenced by Sanger sequencing. In the second assembly reaction, these transcriptional unit constructs were assembled together into the circuit backbone containing the sensors to generate the

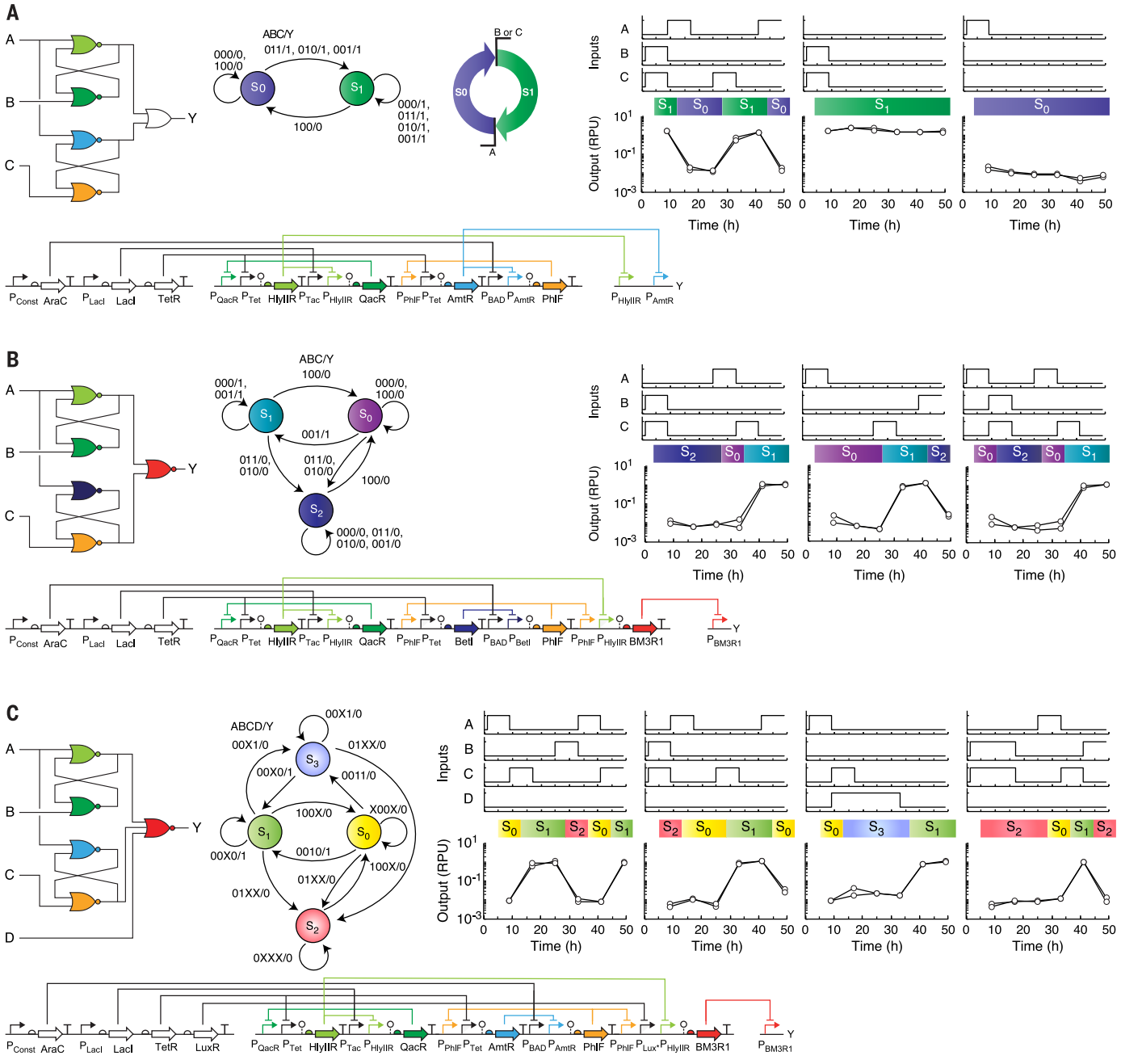


Fig. 4. Implementing checkpoint control by using sequential logic. Circuit-wiring diagrams are shown with the gate colors indicating the assigned repressors (colors as in Fig. 2C); the uncolored gate in (A) is an OR gate. Sensors A, B, C, and D serve as the inputs, and Y is the circuit output promoter. The standard graphical description is shown for each finite-state machine generated by the circuits. The circuit states s_i are shown in circles, and the arrows mark state transitions, where the input causing the transition and the resulting output response are labeled (ABC/Y in binary digits 0 or 1, corresponding to OFF or ON, respectively). (**A**) A circuit designed to have two nonredundant states. A simplified "checkpoint" diagram is shown, in which the colored arrows represent the states and the bent lines show the sensors and corresponding logic required to progress to the next state. Experiments were performed for different input waveforms, showing the ability to switch between states (left) and hold states (right) as memory in the absence of the correct sensory inputs. The output Y is shown, as measured with YFP and cytometry (methods). The color bars in the graph show the predicted state for that set of inputs. (**B**) A circuit designed to have three nonredundant states is shown, following the same format as in (A). (**C**) A circuit designed to have four nonredundant states is shown. The X in the state diagram indicates that the input can be either 0 or 1 for the state transition. Plasmids for the four-input circuit are shown in figs. S16 and S17. For sequential circuits in all panels, two experimental trajectories are shown, performed on different days. Additional waveforms were evaluated for all the circuits (figs. S12 to S14). The inducer concentrations used were 2 ng/ml aTc, 1 mM IPTG, 5 mM L-arabinose, and 10 μM 3OC6-HSL.

final circuit plasmid constructs. For these Type IIS assemblies, the 5-μl reaction volume contained 20 fmol of each purified transcriptional unit plasmid (one plasmid for each gate in the circuit design), 10 fmol of the purified circuit backbone plasmid (three-input or four-input backbone, fig. S19), 5 U BbsI enzyme (New England Biolabs, R0535), and 2.5 U T4 DNA ligase (20 U/μl HC; Promega, M1794) in 1X T4 DNA Ligase Buffer (Promega). The reaction mixture was incubated in a thermal cycler for 6 hours at 37°C, followed by 30 min at 50°C, and then 10 min at 80°C. Then, 2 μl of the assembly reaction was transformed into 8 μl chemically competent cells (*E. coli* NEB 10-beta, New England Biolabs, C3019I). After recovery in SOC, the cells were plated on LB agar containing kanamycin and X-gal. Plasmid DNA was purified from clonal cultures, and the constructed circuits plasmids were sequenced by Sanger sequencing.

Part plasmids (*86*) and destination vectors were assembled by Type IIS restriction-ligation (also referred to as Golden Gate assembly) using the SapI, BsaI, or BbsI restriction enzymes or constructed by PCR site-directed mutagenesis (also referred to as inverse PCR or round-the-horn PCR) using Q5 polymerase (New England Biolabs, M0493). Repressor part plasmids were constructed by Type IIS assembly. Promoter part plasmids and destination vectors were prepared by inverse PCR. For Type IIS assembly of part plasmids, fragments were PCR amplified with Q5 polymerase, digested with DpnI (New England Biolabs, R0176), purified (Qiagen QIAquick column or Agencourt AMPure XP beads), and assembled in a 5-μl reaction containing 10 fmol each purified fragment, 5 U Type IIS enzyme, and 2.5 U T4 DNA ligase (20 U/μl HC; Promega, M1794) in 1X T4 DNA Ligase Buffer (Promega) with the protocol: 37°C for 6 hours, 50°C for 20 min, and 80°C for 15 min (Bio-Rad C1000 thermal cycler, 80°C lid). Type IIS enzymes used were BsaI (New England Biolabs, R0535), BbsI (New England Biolabs, R0539), and SapI (New England Biolabs, R0569). PCR site-directed mutagenesis was used to construct small parts ≤80 bp (e.g., promoters, terminators, and spacers) with half of the part to be inserted added to the 5′ end of the forward and reverse primer. PCR was performed with Q5 polymerase according to manufacturer's recommended protocol using NEB Tm Calculator, followed by DpnI digestion, purification, and simultaneous phosphorylation and ligation with T4 polynucleotide kinase (New England Biolabs, M0201) and T4 DNA ligase (New England Biolabs, M0202) in 1X ligase buffer at room temperature for 1 hour and inactivation 65°C for 10 min before chemical transformation. Part plasmids contained the ColE1 high-copy origin of replication and a standard antibiotic resistance cassette (Kan^R or Amp^R). Parts were flanked by BsaI cut sites and the 4-bp linker sequences GGAG, TACG, AATG, AGGT, GCTT, or CTGA. Plasmids and assembly reactions were chemically transformed cells, clonally amplified, and sequence verified by Sanger sequencing. All part sequences are listed in table S2 and provided in the supplementary SBOL file.

Circuit induction assays

Circuit plasmids were cotransformed with an output plasmid containing YFP expressed by the circuit output promoter (fig. S19D: pPAmeR-YFP, pPAmtR-YFP, pPBetI-YFP, pPBM3R1-YFP, pPHlyIIR-YFP, pPIcaRA-YFP, pPLitR-YFP, pPPhlF-YFP, pPQacR-YFP, pPSrpR-YFP, or pPAmtR-PHlyIIR-YFP). When a circuit contains multiple output promoters, these were measured independently in multiple strains, where each strain contains a reporter for one output. To cotransform the circuit and output plasmids, 0.5 μl of the purified circuit plasmid and 0.5 μl of the purified output plasmid were transformed into 5 μl chemically competent cells (*E. coli* NEB 10-beta, New England Biolabs, C3019I) and plated on LB agar containing kanamycin and spectinomycin. One colony was inoculated into 200 μl M9 minimal media with appropriate antibiotics in a V-bottom 96-well microtiter plate (ThermoFisher Scientific, 249952) sealed with a breathable seal (WorldWide Medical Products Inc., 41061023) and incubated for 16 hours (overnight) at 37°C with shaking (1000 rpm) in an Elmi DTS-4 Digital Thermostatic Microplate Shaker (Elmi Ltd., Riga, Latvia). After overnight incubation, cells were diluted with two serial dilutions of 15 μl culture into 185 μl M9 minimal media with antibiotics (178-fold total dilution) and incubated for 3 hours at 37°C with shaking (1000 rpm) in a V-bottom plate (ThermoFisher Scientific, 249952). Then cells were diluted by two serial dilutions (resulting in 658-fold dilution), the first with 15 μl culture into 185 μl M9 minimal media with antibiotics, and then 15 μl diluted culture into 145 μl M9 minimal media with antibiotics and inducers. The inducer concentrations used for the sensors were 2 ng/ml aTc, 1 mM IPTG, 5 mM L-arabinose, and 10 μM 3OC6-HSL. The cells were incubated for 8 hours at 37°C with shaking (1000 rpm) in a V-bottom plate with a breathable seal. To measure cell fluorescence, a 5-μl aliquot of the cell culture was diluted into 195 μl phosphate buffered saline (PBS, pH 7.4) with 2 mg/ml kanamycin and incubated for 1 hour at room temperature before flow cytometry analysis.

To change the inducers or continue growing the cells, the cells were diluted (658-fold total) by two serial dilutions: (i) 15 μl culture into 185 μl M9 minimal media with antibiotics, then (ii) 15 μl diluted culture into 145 μl M9 minimal media with antibiotics and inducer inputs as needed. For media containing 3OC6-HSL, the cells were washed by pelleting and resuspending in fresh media before carrying out the two dilutions. The diluted cells were then incubated for 8 hours at 37°C with shaking (1000 rpm) in a V-bottom plate with a breathable seal. Then to measure cell fluorescence, an aliquot of cells was diluted into phosphate buffered saline (PBS, pH 7.4) with 2 mg/ml kanamycin and incubated at room temperature for 1 hour before flow cytometry analysis.

Flow cytometry analysis

Fluorescence was measured using an LSRII Fortessa flow cytometer (BD Biosciences, San Jose, CA) and the BD FACSDiva software. The settings used were: 437-V FSC voltage, 289-V SSC voltage, and 425-V green laser (488 nm) voltage. For each culture, fluorescence was measured on the FITC channel, and the data for all events were collected with a cutoff of 20,000 gated events. A flow rate of approximately 1000 events/s or less was used for analysis. The cells were gated with a rectangular gate for cell-sized particles using the FlowJo software (TreeStar Inc., Ashland, OR). The median cell fluorescence was calculated in FlowJo using the geometric median statistical tool.

Conversion of fluorescence AU to RPU

The arbitrary units of fluorescence from the cytometry measurements were converted to relative promoter units, as described previously (*34*, *87*). The conversion follows: $RPU = (\langle YFP \rangle - \langle YFP_0 \rangle)/(\langle YFP_{RPU} \rangle - \langle YFP_0 \rangle)$, where $\langle YFP \rangle$ is the median fluorescence of the promoter, $\langle YFP_0 \rangle$ is the median fluorescence of white cells (*E. coli* NEB 10-beta lacking plasmids), and $\langle YFP_{RPU} \rangle$ is the median fluorescence of the standard promoter. The standard plasmid that we use for RPU measurements is pJSBS_RPU_standard in fig. S19. In this manuscript, we used either pAN1717 or pAN1730, depending on whether the circuit contained 2 or ≥3 sensors to measure the standard fluorescence (because of a concern over the potential impact of the burden of expressing additional regulators). In theory, we could convert the fluorescence values to that of the standard plasmid, pJSBS_RPU_standard. However, in our hands, all three plasmids produced statistically indistinguishable fluorescence measurements, so the conversion factor is unity.

Construction and characterization of four-input circuit backbone

The circuit shown in Fig. 4C required that four sensors be carried on a single plasmid backbone. Therefore, we constructed four-input circuit backbones by adding the LuxR regulator to the three-input circuit backbone (pAN871). Permutations of the sensor operons were constructed (fig. S16). For each circuit backbone, the sensors were characterized, and the toxicity of the backbone was tested by measuring the growth (OD600) in a plate reader (Biotek H1 Synergy Hybrid Multi-Mode Reader) with shaking at 37°C (fig. S16). Sensor characterization was performed to determine the dynamic range of each sensor on the circuit backbone (*34*). The ON and OFF activity of each sensor input promoter (P_{Tet}, P_{Tac}, P_{BAD}, or P_{Lux}) was measured in RPU by growing the cells with and without the appropriate inducer (2 ng/ml aTc, 1 mM IPTG, 5 mM L-arabinose, or 10 μM 3OC6-HSL) (fig. S16). Cells containing each sensor characterization plasmid were grown as described above for the circuit assays and measured by flow cytometry. The pLW555 plasmid was selected as the best four-input circuit backbone. The plasmid map is shown in fig. S16. The sensor response functions for pLW555 were measured by testing additional inducer concentrations (fig. S17). The sensor input promoter values for the three-input circuit backbone pAN871 were P_{Tet} (ON = 3.66 RPU, OFF = 0.0078 RPU),

P_{Tac} (ON = 2.96 RPU, OFF = 0.011 RPU), and P_{BAD} (ON = 2.25 RPU, OFF = 0.022 RPU). The measured sensor input promoters for the four-input circuit backbone pLW555 were P_{Tet} (ON = 3.16 RPU, OFF = 0.0097 RPU), P_{Tac} (ON = 1.61 RPU, OFF = 0.0072 RPU), P_{BAD} (ON = 1.48 RPU, OFF = 0.020 RPU), and P_{Lux} (ON = 1.32 RPU, OFF = 0.0056 RPU).

Computational analysis of circuits

The ODE models for the SR latch, D latch and three-latch memory register are provided as supplementary files in the .ode input file format that was used. Numerical analysis of the ODE models was performed using the XPP AUTO software (XPPAUT 8.0 version) with the Runge-Kutta numerical method for the phase plane analyses and bifurcation analyses (*88*). The numeric parameters were set as follows: step size dt = 0.001, nullcline grid size = 40, output storage integration steps n = 1, and total time = 500. To simulate state transitions, the initial conditions were set to the previous state's steady-state output values, and the sensor input values were updated corresponding to the inducer inputs present. The measured sensor input promoter values from the sensor characterization were used for the inputs to the model. The previously measured gate response functions were used (*34*). The bifurcation analyses were performed using the AUTO interface and starting at a steady-state point. The AUTO numeric parameters were the default values, except: the maximum number of steps = 1000, the initial step size = 0.01 (sign was changed to represent increasing or decreasing inducer input), minimum step size = 0.001, and maximum step size = 0.01. To account for the experimentally measured copy-number difference between the circuit backbone and the output plasmid, the XPPAUT predictions were multiplied by a scaling factor ε = 0.25. The equilibria separation was calculated as the distance d between the x-y coordinates of the unstable equilibrium U and a stable equilibrium (S1 or S2) in log-log scale on the phase plane plot as:

$$d_1 = \sqrt{(x_{S1} - x_U)^2 + (y_{S1} - y_U)^2}$$

and

$$d_2 = \sqrt{(x_{S2} - x_U)^2 + (y_{S2} - y_U)^2}$$

Transversality was calculated as the area between the nullcline curves between the three intersecting points in log-log coordinate space using trapezoidal numerical integration in Matlab. A modified version of the Cello software (*34*) was used to assign gates to the D latch. The Cello software and code is available open source (https://github.com/CIDARLAB/cello). In this version, the hill-climbing assignment algorithm was selected for the gate assignment. To specify the sequential logic, we split the D latch's state table into a time series of truth tables that sampled all state transitions. We used structural Verilog to specify how the NOR gates are wired together (supplementary files). Cello was run locally using the Cello API, and the growth score cutoff was changed to 0.10. The input UCF file was the Eco1C1G1T1.UCF.json file (*34*) with the LmrA and PsrA gates removed. The Cello code is written in Java (version 1.8.0_31) and uses the Apache Maven (version 3.2.1) software project management.

REFERENCES AND NOTES

1. H. H. McAdams, A. Arkin, It's a noisy business! Genetic regulation at the nanomolar scale. *Trends Genet.* **15**, 65–69 (1999). doi: 10.1016/S0168-9525(98)01659-X; pmid: 10098409
2. L. H. Hartwell, T. A. Weinert, Checkpoints: Controls that ensure the order of cell cycle events. *Science* **246**, 629–634 (1989). doi: 10.1126/science.2683079; pmid: 2683079
3. A. Verdugo, P. K. Vinod, J. J. Tyson, B. Novak, Molecular mechanisms creating bistable switches at cell cycle transitions. *Open Biol.* **3**, 120179 (2013). doi: 10.1098/rsob.120179; pmid: 23486222
4. G. Yao, T. J. Lee, S. Mori, J. R. Nevins, L. You, A bistable Rb-E2F switch underlies the restriction point. *Nat. Cell Biol.* **10**, 476–482 (2008). doi: 10.1038/ncb1711; pmid: 18364697
5. A. Ciliberto, J. J. Tyson, Mathematical model for early development of the sea urchin embryo. *Bull. Math. Biol.* **62**, 37–59 (2000). doi: 10.1006/bulm.1999.0129; pmid: 10824420
6. J. E. Ferrell Jr. *et al*., Simple, realistic models of complex biological processes: Positive feedback and bistability in a cell fate switch and a cell cycle oscillator. *FEBS Lett.* **583**, 3999–4005 (2009). doi: 10.1016/j.febslet.2009.10.068; pmid: 19878681
7. C. Nathan, Points of control in inflammation. *Nature* **420**, 846–852 (2002). doi: 10.1038/nature01320; pmid: 12490957
8. H. Maamar, D. Dubnau, Bistability in the *Bacillus subtilis* K-state (competence) system requires a positive feedback loop. *Mol. Microbiol.* **56**, 615–624 (2005). doi: 10.1111/j.1365-2958.2005.04592.x; pmid: 15819619
9. C. A. Voigt, D. M. Wolf, A. P. Arkin, The *Bacillus subtilis sin* operon: An evolvable network motif. *Genetics* **169**, 1187–1202 (2005). doi: 10.1534/genetics.104.031955; pmid: 15466432
10. B. B. Zhou, S. J. Elledge, The DNA damage response: Putting checkpoints in perspective. *Nature* **408**, 433–439 (2000). doi: 10.1038/35044005; pmid: 11100718
11. A. Tiwari, J. C. Ray, J. Narula, O. A. Igoshin, Bistable responses in bacterial genetic networks: Designs and dynamical consequences. *Math. Biosci.* **231**, 76–89 (2011). doi: 10.1016/j.mbs.2011.03.004; pmid: 21385588
12. S. Koirala *et al*., A nutrient-tunable bistable switch controls motility in *Salmonella enterica* serovar Typhimurium. *mBio* **5**, e01611–e01614 (2014). doi: 10.1128/mBio.01611-14; pmid: 25161191
13. P. Aldridge, K. T. Hughes, Regulation of flagellar assembly. *Curr. Opin. Microbiol.* **5**, 160–165 (2002). doi: 10.1016/S1369-5274(02)00302-8; pmid: 11934612
14. M. B. Elowitz, S. Leibler, A synthetic oscillatory network of transcriptional regulators. *Nature* **403**, 335–338 (2000). doi: 10.1038/35002125; pmid: 10659856
15. J. Stricker *et al*., A fast, robust and tunable synthetic gene oscillator. *Nature* **456**, 516–519 (2008). doi: 10.1038/nature07389; pmid: 18971928
16. S. Hooshangi, W. E. Bentley, From unicellular properties to multicellular behavior: Bacteria quorum sensing circuitry and applications. *Curr. Opin. Biotechnol.* **19**, 550–555 (2008). doi: 10.1016/j.copbio.2008.10.007; pmid: 18977301
17. N. Rosenfeld, J. W. Young, U. Alon, P. S. Swain, M. B. Elowitz, Gene regulation at the single-cell level. *Science* **307**, 1962–1965 (2005). doi: 10.1126/science.1106914; pmid: 15790856
18. J. Müller, C. Kuttler, B. A. Hense, M. Rothballer, A. Hartmann, Cell-cell communication by quorum sensing and dimension-reduction. *J. Math. Biol.* **53**, 672–702 (2006). doi: 10.1007/s00285-006-0024-z; pmid: 16897015
19. H. Bolouri, E. H. Davidson, Modeling transcriptional regulatory networks. *BioEssays* **24**, 1118–1129 (2002). doi: 10.1002/bies.10189; pmid: 12447977
20. C. D. Thron, Bistable biochemical switching and the control of the events of the cell cycle. *Oncogene* **15**, 317–325 (1997). doi: 10.1038/sj.onc.1201190; pmid: 9233766
21. J. J. Tyson, Models of cell cycle control in eukaryotes. *J. Biotechnol.* **71**, 239–244 (1999). doi: 10.1016/S0168-1656(99)00027-9; pmid: 10483109
22. M. Laurent, N. Kellershohn, Multistability: A major means of differentiation and evolution in biological systems. *Trends Biochem. Sci.* **24**, 418–422 (1999). doi: 10.1016/S0968-0004(99)01473-5; pmid: 10542403
23. A. P. Chandrakasan, W. J. Bowhill, F. Fox, Eds., *Design of High-Performance Microprocessor Circuits* (Wiley-IEEE Press, 2001).
24. S. Hauck, S. Burns, G. Borriello, C. Ebeling, An FPGA for implementing asynchronous circuits. *IEEE Des. Test Comput.* **11**, 60–69 (1994). doi: 10.1109/MDT.1994.303848
25. S. W. Moore, G. S. Taylor, P. A. Cunningham, R. D. Mullins, P. Robinson, "Self calibrating clocks for globally asynchronous locally synchronous systems," in *Proceedings 2000 International Conference on Computer Design* (IEEE Computer Sociecty, 2000), pp. 73–78.
26. R. Weiss, G. E. Homsy, T. F. Knight, "Toward in vivo digital circuits," in *Evolution as Computation*, L. F. Landweber, E. Winfree, Eds. (Natural Computing Series, Springer, 2002), pp. 275–295.
27. N. E. Buchler, U. Gerland, T. Hwa, On schemes of combinatorial transcription logic. *Proc. Natl. Acad. Sci. U.S.A.* **100**, 5136–5141 (2003). doi: 10.1073/pnas.0930314100; pmid: 12702751
28. R. S. Cox 3rd, M. G. Surette, M. B. Elowitz, Programming gene expression with combinatorial promoters. *Mol. Syst. Biol.* **3**, 145 (2007). doi: 10.1038/msb4100187; pmid: 18004278
29. W. A. Lim, Designing customized cell signalling circuits. *Nat. Rev. Mol. Cell Biol.* **11**, 393–403 (2010). doi: 10.1038/nrm2904; pmid: 20485291
30. B. Wang, M. Buck, Customizing cell signaling using engineered genetic logic circuits. *Trends Microbiol.* **20**, 376–384 (2012). doi: 10.1016/j.tim.2012.05.001; pmid: 22682075
31. J. A. Brophy, C. A. Voigt, Principles of genetic circuit design. *Nat. Methods* **11**, 508–520 (2014). doi: 10.1038/nmeth.2926; pmid: 24781324
32. J. Chappell, K. E. Watters, M. K. Takahashi, J. B. Lucks, A renaissance in RNA synthetic biology: New mechanisms, applications and tools for the future. *Curr. Opin. Chem. Biol.* **28**, 47–56 (2015). doi: 10.1016/j.cbpa.2015.05.018; pmid: 26093826
33. B. Canton, A. Labno, D. Endy, Refinement and standardization of synthetic biological parts and devices. *Nat. Biotechnol.* **26**, 787–793 (2008). doi: 10.1038/nbt1413; pmid: 18612302
34. A. A. K. Nielsen *et al*., Genetic circuit design automation. *Science* **352**, aac7341 (2016). doi: 10.1126/science.aac7341; pmid: 27034378
35. J. Bonnet, P. Subsoontorn, D. Endy, Rewritable digital data storage in live cells via engineered control of recombination directionality. *Proc. Natl. Acad. Sci. U.S.A.* **109**, 8884–8889 (2012). doi: 10.1073/pnas.1202344109; pmid: 22615351
36. C. M. Ajo-Franklin *et al*., Rational design of memory in eukaryotic cells. *Genes Dev.* **21**, 2271–2276 (2007). doi: 10.1101/gad.1586107; pmid: 17875664
37. D. R. Burrill, P. A. Silver, Making cellular memories. *Cell* **140**, 13–18 (2010). doi: 10.1016/j.cell.2009.12.034; pmid: 20085698
38. D. R. Burrill, M. C. Inniss, P. M. Boyle, P. A. Silver, Synthetic memory circuits for tracking human cell fate. *Genes Dev.* **26**, 1486–1497 (2012). doi: 10.1101/gad.189035.112; pmid: 22751502
39. T. Shopera *et al*., Robust, tunable genetic memory from protein sequestration combined with positive feedback. *Nucleic Acids Res.* **43**, 9086–9094 (2015). doi: 10.1093/nar/gkv936; pmid: 26384562
40. T. S. Gardner, C. R. Cantor, J. J. Collins, Construction of a genetic toggle switch in *Escherichia coli*. *Nature* **403**, 339–342 (2000). doi: 10.1038/35002131; pmid: 10659857
41. W. Chen, J. E. Bailey, Communication to the editor. Application of the cross-regulation system as a metabolic switch. *Biotechnol. Bioeng.* **43**, 1190–1193 (1994). doi: 10.1002/bit.260431124; pmid: 18615532
42. T. Ellis, X. Wang, J. J. Collins, Diversity-based, model-guided construction of synthetic gene networks with predicted functions. *Nat. Biotechnol.* **27**, 465–471 (2009). doi: 10.1038/nbt.1536; pmid: 19377462
43. T. Lebar *et al*., A bistable genetic switch based on designable DNA-binding domains. *Nat. Commun.* **5**, 5007 (2014). doi: 10.1038/ncomms6007; pmid: 25264186
44. J. W. Lee *et al*., Creating single-copy genetic circuits. *Mol. Cell* **63**, 329–336 (2016). doi: 10.1016/j.molcel.2016.06.006; pmid: 27425413
45. H. Kobayashi *et al*., Programmable cells: Interfacing natural and engineered gene networks. *Proc. Natl. Acad. Sci. U.S.A.* **101**, 8414–8419 (2004). doi: 10.1073/pnas.0402940101; pmid: 15159530
46. C. Lou *et al*., Synthesizing a novel genetic sequential logic circuit: A push-on push-off switch. *Mol. Syst. Biol.* **6**, 350 (2010). doi: 10.1038/msb.2010.2; pmid: 20212522
47. J. W. Kotula *et al*., Programmable bacteria detect and record an environmental signal in the mammalian gut. *Proc. Natl.*

Acad. Sci. U.S.A. **111**, 4838–4843 (2014). doi: 10.1073/pnas.1321321111; pmid: 24639514
48. D. Chen, A. P. Arkin, Sequestration-based bistability enables tuning of the switching boundaries and design of a latch. *Mol. Syst. Biol.* **8**, 620 (2012). doi: 10.1038/msb.2012.52; pmid: 23089683
49. J. Fernandez-Rodriguez, L. Yang, T. E. Gorochowski, D. B. Gordon, C. A. Voigt, Memory and combinatorial logic based on DNA inversions: Dynamics and evolutionary stability. *ACS Synth. Biol.* **4**, 1361–1372 (2015). doi: 10.1021/acssynbio.5b00170; pmid: 26548807
50. G. Rodrigo, A. Jaramillo, Computational design of digital and memory biological devices. *Syst. Synth. Biol.* **1**, 183–195 (2007). doi: 10.1007/s11693-008-9017-0; pmid: 19003443
51. G. Fritz, N. E. Buchler, T. Hwa, U. Gerland, Designing sequential transcription logic: A simple genetic circuit for conditional memory. *Syst. Synth. Biol.* **1**, 89–98 (2007). doi: 10.1007/s11693-007-9006-8; pmid: 19003438
52. P. Hillenbrand, G. Fritz, U. Gerland, Biological signal processing with a genetic toggle switch. *PLOS ONE* **8**, e68345 (2013). doi: 10.1371/journal.pone.0068345; pmid: 23874595
53. C. H. Chuang, C. L. Lin, Synthesizing genetic sequential logic circuit with clock pulse generator. *BMC Syst. Biol.* **8**, 63 (2014). doi: 10.1186/1752-0509-8-63; pmid: 24884665
54. S. H. Strogatz, *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering* (Addison-Wesley, 1994).
55. B. C. Stanton *et al*., Genomic mining of prokaryotic repressors for orthogonal logic gates. *Nat. Chem. Biol.* **10**, 99–105 (2014). doi: 10.1038/nchembio.1411; pmid: 24316737
56. A. Tamsir, J. J. Tabor, C. A. Voigt, Robust multicellular computing using genetically encoded NOR gates and chemical 'wires'. *Nature* **469**, 212–215 (2011). doi: 10.1038/nature09565; pmid: 21150903
57. S. Chen *et al*., Automated design of genetic toggle switches with predetermined bistability. *ACS Synth. Biol.* **1**, 284–290 (2012). doi: 10.1021/sb300027y; pmid: 23651251
58. G. M. Guidi, A. Goldbeter, Bistability without hysteresis in chemical reaction systems: A theoretical analysis of irreversible transitions between multiple steady states. *J. Phys. Chem. A* **101**, 9367–9376 (1997). doi: 10.1021/jp972244k
59. J. E. Ferrell, W. Xiong, Bistability in cell signaling: How to make continuous processes discontinuous, and reversible processes irreversible. *Chaos* **11**, 227–236 (2001). doi: 10.1063/1.1349894; pmid: 12779456
60. J. L. Cherry, F. R. Adler, How to make a biological switch. *J. Theor. Biol.* **203**, 117–133 (2000). doi: 10.1006/jtbi.2000.1068; pmid: 10704297
61. J. E. Ferrell Jr., Self-perpetuating states in signal transduction: Positive feedback, double-negative feedback and bistability. *Curr. Opin. Cell Biol.* **14**, 140–148 (2002). doi: 10.1016/S0955-0674(02)00314-9; pmid: 11891111
62. T. Afroz, C. L. Beisel, Understanding and exploiting feedback in synthetic biology. *Chem. Eng. Sci.* **103**, 79–90 (2013). doi: 10.1016/j.ces.2013.02.017
63. V. A. Pedroni, *Circuit Design and Simulation with VHDL* (MIT Press, ed. 2, 2010).
64. A. T. Abdel-Hamid, M. Zaki, S. Tahar, in *Canadian Conference on Electrical and Computer Engineering 2004* (IEEE Canada, 2004), vol. 4, pp. 1907–1910.
65. P. Coussy, A. Morawiec, *High-Level Synthesis: From Algorithm to Digital Circuit* (Springer, 2008).
66. C. Lou, B. Stanton, Y. J. Chen, B. Munsky, C. A. Voigt, Ribozyme-based insulator parts buffer synthetic circuits from genetic context. *Nat. Biotechnol.* **30**, 1137–1142 (2012). doi: 10.1038/nbt.2401; pmid: 23034349
67. A. Garg, J. J. Lohmueller, P. A. Silver, T. Z. Armel, Engineering synthetic TAL effectors with orthogonal target sites. *Nucleic Acids Res.* **40**, 7584–7595 (2012). doi: 10.1093/nar/gks404; pmid: 22581776
68. L. S. Qi *et al*., Repurposing CRISPR as an RNA-guided platform for sequence-specific control of gene expression. *Cell* **152**, 1173–1183 (2013). doi: 10.1016/j.cell.2013.02.022; pmid: 23452860
69. A. A. Nielsen, C. A. Voigt, Multi-input CRISPR/Cas genetic circuits that interface host regulatory networks. *Mol. Syst. Biol.* **10**, 763 (2014). doi: 10.15252/msb.20145735; pmid: 25422271
70. B. Huang *et al*., Interrogating the topological robustness of gene regulatory circuits by randomization. *PLOS Comput. Biol.* **13**, e1005456 (2017). doi: 10.1371/journal.pcbi.1005456; pmid: 28362798
71. M. Lu, J. Onuchic, E. Ben-Jacob, Construction of an effective landscape for multistate genetic switches. *Phys. Rev. Lett.* **113**, 078102 (2014). doi: 10.1103/PhysRevLett.113.078102; pmid: 25170733
72. A. H. Chau, J. M. Walter, J. Gerardin, C. Tang, W. A. Lim, Designing synthetic regulatory networks capable of self-organizing cell polarization. *Cell* **151**, 320–332 (2012). doi: 10.1016/j.cell.2012.08.040; pmid: 23039994
73. F. Zhang, J. Keasling, Biosensors and their applications in microbial metabolic engineering. *Trends Microbiol.* **19**, 323–329 (2011). doi: 10.1016/j.tim.2011.05.003; pmid: 21664818
74. K. Temme, R. Hill, T. H. Segall-Shapiro, F. Moser, C. A. Voigt, Modular control of multiple pathways using engineered orthogonal T7 polymerases. *Nucleic Acids Res.* **40**, 8773–8781 (2012). doi: 10.1093/nar/gks597; pmid: 22743271
75. L. Yang *et al*., Permanent genetic memory with >1-byte capacity. *Nat. Methods* **11**, 1261–1266 (2014). doi: 10.1038/nmeth.3147; pmid: 25344638
76. J. C. Anderson, E. J. Clarke, A. P. Arkin, C. A. Voigt, Environmentally controlled invasion of cancer cells by engineered bacteria. *J. Mol. Biol.* **355**, 619–627 (2006). doi: 10.1016/j.jmb.2005.10.076; pmid: 16330045
77. E. J. Archer, A. B. Robinson, G. M. Süel, Engineered *E. coli* that detect and respond to gut inflammation through nitric oxide sensing. *ACS Synth. Biol.* **1**, 451–457 (2012). doi: 10.1021/sb3000595; pmid: 23656184
78. K. A. Haynes, F. Ceroni, D. Flicker, A. Younger, P. A. Silver, A sensitive switch for visualizing natural gene silencing in single cells. *ACS Synth. Biol.* **1**, 99–106 (2012). doi: 10.1021/sb3000035; pmid: 22530199
79. A. Courbet, D. Endy, E. Renard, F. Molina, J. Bonnet, Detection of pathological biomarkers in human clinical samples via amplifying genetic switches and logic gates. *Sci. Transl. Med.* **7**, 289ra83 (2015). doi: 10.1126/scitranslmed.aaa3601; pmid: 26019219
80. S. Slomovic, K. Pardee, J. J. Collins, Synthetic biology devices for in vitro and in vivo diagnostics. *Proc. Natl. Acad. Sci. U.S.A.* **112**, 14429–14435 (2015). doi: 10.1073/pnas.1508521112; pmid: 26598662
81. L. Morsut *et al*., Engineering customized cell sensing and response behaviors using synthetic notch receptors. *Cell* **164**, 780–791 (2016). doi: 10.1016/j.cell.2016.01.012; pmid: 26830878
82. M. J. Smanski *et al*., Synthetic biology to access and expand nature's chemical diversity. *Nat. Rev. Microbiol.* **14**, 135–149 (2016). doi: 10.1038/nrmicro.2015.24; pmid: 26876034
83. M. S. Antunes *et al*., A synthetic de-greening gene circuit provides a reporting system that is remotely detectable and has a re-set capacity. *Plant Biotechnol. J.* **4**, 605–622 (2006). doi: 10.1111/j.1467-7652.2006.00205.x; pmid: 17309732
84. M. S. Antunes *et al*., Programmable ligand detection system in plants through a synthetic signal transduction pathway. *PLOS ONE* **6**, e16292 (2011). doi: 10.1371/journal.pone.0016292; pmid: 21283542
85. T. Durfee *et al*., The complete genome sequence of *Escherichia coli* DH10B: Insights into the biology of a laboratory workhorse. *J. Bacteriol.* **190**, 2597–2606 (2008). doi: 10.1128/JB.01695-07; pmid: 18245285
86. L. B. A. Woodruff *et al*., Registry in a tube: Multiplexed pools of retrievable parts for genetic design space exploration. *Nucleic Acids Res.* **45**, 1553–1565 (2017). pmid: 28007941
87. J. R. Kelly *et al*., Measuring the activity of BioBrick promoters using an in vivo reference standard. *J. Biol. Eng.* **3**, 4 (2009). doi: 10.1186/1754-1611-3-4; pmid: 19298678
88. B. Ermentrout, *Simulating, Analyzing, and Animating Dynamical Systems: A Guide to XPPAUT for Researchers and Students* (Software, Environments, and Tools Series, Society for Industrial and Applied Mathematics, Philadelphia, 2002).
89. N. Roehner, Z. Zhang, T. Nguyen, C. J. Myers, Generating systems biology markup language models from the synthetic biology open language. *ACS Synth. Biol.* **4**, 873–879 (2015). doi: 10.1021/sb5003289; pmid: 25822671

ACKNOWLEDGMENTS

We thank N. Roehner for help with converting the models to SBML. **Funding:** This work was supported by funding from the Defense Advanced Research Projects Agency (DARPA) Living Foundries program (awards HR0011-13-1-0001 and HR0011-15-C-0084). C.A.V. and A.A.K.N. were supported by the Office of Naval Research Multidisciplinary University Research Initiative grant N00014-13-1-0074. **Author contributions:** L.B.A. and C.A.V. conceived of the study, designed experiments, and wrote the manuscript. L.B.A. and A.A.K.N. developed the computational analysis. L.B.A. performed the experiments. **Competing interests:** C.A.V. and A.A.K.N. are founders of Asimov, Inc., a company related to this work. **Data and materials availability:** All data are available in the manuscript or the supplementary materials.

SUPPLEMENTARY MATERIALS

www.sciencemag.org/content/361/6408/eaap8987/suppl/DC1
Supplementary Text
Figs. S1 to S19
Tables S1 to S3
References (90–97)
Data Files S1 to S8

8 September 2017; accepted 3 August 2018
10.1126/science.aap8987

Cellular checkpoint control using programmable sequential logic

Lauren B. Andrews, Alec A. K. Nielsen, and Christopher A. Voigt

Science, **361** (6408), eaap8987.
DOI: 10.1126/science.aap8987

Building smarter synthetic biological circuits

Synthetic genetic and biological regulatory circuits can enable logic functions to form the basis of biological computing; synthetic biology can also be used to control cell behaviors (see the Perspective by Glass and Alon). Andrews *et al.* used mathematical models and computer algorithms to combine standardized components and build programmable genetic sequential logic circuits. Such circuits can perform regulatory functions much like the biological checkpoint circuits of living cells. Circuits composed of interacting proteins could be used to bypass gene regulation, interfacing directly with cellular pathways without genome modification. Gao *et al.* engineered proteases that regulate one another, respond to diverse inputs that include oncogene activation, process signals, and conditionally activate responses such as those leading to cell death. This platform should facilitate development of “smart” therapeutic circuits for future biomedical applications.

Science, this issue p. eaap8987, p. 1252; see also p. 1199