A Cytomorphic Chip for Quantitative Modeling of Fundamental Bio-Molecular Circuits

Sung Sik Woo, Jaewook Kim, and Rahul Sarpeshkar

Abstract—**We describe a 0.35 μm BiCMOS silicon chip that quantitatively models fundamental molecular circuits via efficient log-domain cytomorphic transistor equivalents. These circuits include those for biochemical binding with automatic representation of non-modular and loading behavior, e.g., in cascade and fan-out topologies; for representing variable Hill-coefficient operation and cooperative binding; for representing inducer, transcription-factor, and DNA binding; for probabilistic gene transcription with analogic representations of log-linear and saturating operation; for gain, degradation, and dynamics of mRNA and protein variables in transcription and translation; and, for faithfully representing biological noise via tunable stochastic transistor circuits. The use of on-chip DACs and ADCs enables multiple chips to interact via incoming and outgoing molecular digital data packets and thus create scalable biochemical reaction networks. The use of off-chip digital processors and on-chip digital memory enables programmable connectivity and parameter storage.**

We show that published static and dynamic MATLAB models of synthetic biological circuits including repressilators, feed-forward loops, and feedback oscillators are in excellent quantitative agreement with those from transistor circuits on the chip. Computationally intensive stochastic Gillespie simulations of molecular production are also rapidly reproduced by the chip and can be reliably tuned over the range of signal-to-noise ratios observed in biological cells.

Index Terms—**cytomorphic, molecular circuits, reaction networks, transistor models, quantitative cellular models, stochastic simulation, synthetic biology, systems biology.**

I. INTRODUCTION

THE modeling and simulation of biochemical reaction networks in living cells that involve small molecules, DNA, RNA, and proteins is challenging for at least two reasons: 1) We still have a vast ignorance of the pathways and parameters of such networks, though our knowledge about them is rapidly increasing every day; 2) A fast, high-throughput simulation-and-modeling tool that can rapidly enable exploration, validation, learning, and constraining of the vast parameter and connectivity spaces with known experimental data is lacking. Just as Google helps us search for solutions amongst big-data spaces today, constantly learning to do better, if such a tool existed for the vast space of reaction networks, it could help us predict and design new experiments for further discovery.

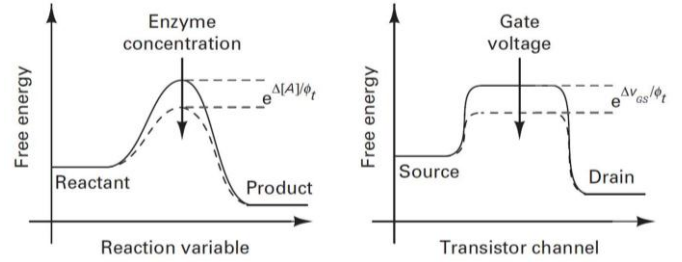It has been estimated that the stochastic Gillespie simulation

Fig. 1. Poisson chemical reaction flux and Poisson electron flow in sub-threshold transistors obey the same Boltzmann exponential laws of thermodynamics, and are therefore mathematically similar [13]. Electron copy number is analogous to molecular copy number and temperature represents itself.

[1]–[3] of the low-molecular-count networks of just one cell cycle of one *E. coli* cell would involve 10^{14} biochemical reactions and is incredibly computationally intensive to simulate [4]. A recent whole-cell simulation of the smallest bacterium of earth, with highly oversimplified, largely deterministic, synchronous, non-Poisson models took 10 hours on a 128-node Linux cluster to simulate just one cell cycle [5]. However, it is computationally intensive and daunting to faithfully model cell-to-cell variability; Poisson stochastics due to modest molecular counts or high Fano factors in multiple state variables; host and environmental context; the feedback "loading" of downstream state variables on upstream state variables; diffusion and compartmentalization; cell-cycle and cell-division effects; unpredictable "cross talk" via shared polymerase, ribosome, ATP, protease, and RNAase resources; molecular toxicity due to high-copy-number, cross reactive, or drug effects; and interactions between cell-cell communication, regulatory, metabolic, developmental, and signaling pathways. Yet, many of these effects are important in systems and synthetic biology and in actual disease: For example, stochastics is important for tumor drug resistance [6] and for antibiotic drug resistance of "persister cells" [7]. Metabolic loading and molecular toxicity [8], [9], loading of one circuit by another [10], the breakdown of logic abstractions, and interactions between cellular resources [8], [11], have all prevented synthetic circuits from exceeding even six logic parts in one cell after almost two decades of research [12]. Therefore, there is motivation to create a fast-and-

S. S. Woo, J. Kim, and R. Sarpeshkar are with the Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: rahuls@mit.edu).

flexible simulation, modeling, and design tool where the addition of complexity in modeling does not drastically compromise the speed of the simulation as it often does today. For example, even 6-state-variable stochastic simulations can take hours.

Fortunately, the exponential Boltzmann equations that govern stochastic electron flow in a transistor and stochastic biochemical reaction flux in a chemical reaction are deeply mathematically similar as suggested intuitively in Fig. 1 and described quantitatively in [13]. *This similarity enables us to map biochemical circuits to log-domain transistor circuits with a few handfuls of transistors per gene or protein* [8], [13]–[17]. In turn, log-domain circuit motifs for linearization [18] have been mapped to create highly part-count efficient circuits in living cells [19]. The **"cytomorphic"** mapping of Fig. 1, can enable rapid and highly parallel stochastic simulations of a single cell on a few chips and speedup multi-cell simulations on multi-chip electronic boards [13]. Therefore, we were motivated to create a few fundamental molecular circuits that could be composed and scaled to model large biochemical reaction networks in the future. This paper describes these fundamental building-block cytomorphic circuits. The paper focuses on establishing quantitative agreement of the cytomorphic chip measurements with prior biological measurements and models.

The building-block circuits may be classified into seven important categories:

1) Basic BiCMOS current-mode analog circuits exploit the log-domain cytomorphic mapping to capture the exact dynamics of **fundamental mass-action molecular kinetics** such as association, dissociation, and degradation. These fundamental circuits are general enough to capture subtle effects such as loading, fan-out, feedback, and substrate depletion through the use of a few explicit connections and Kirchhoff's current law. A wide dynamic range of operation and low power consumption are achieved through the use of bipolar and subthreshold MOS transistors that function at low current levels.

2) The ability to model cooperative binding is enabled by **tunable Hill-function** building-block circuits.

3) An "ITD" block built by a composition of above current-mode circuits enables mapping of the exact differential equations of **inducer-transcription-factor binding and transcription-factor-DNA binding** including forward and reverse reactions, degradation, protection from degradation of transcription factors bound to DNA, and the change in DNA binding affinity of transcription factors when bound by an inducer.

4) An "analogic" current-mode circuit determines the transcription rate of genes based on the probability of multiple transcription-factor DNA binding sites being occupied or unoccupied in a combinatorial fashion with the relative mRNA production rate of each such combinatorial state programmable by the user. This strategy enables us to implement any arbitrary **"analogic promoter function"** that is capable of complex saturating digital logic or probabilistic analog behavior depending on the molecular
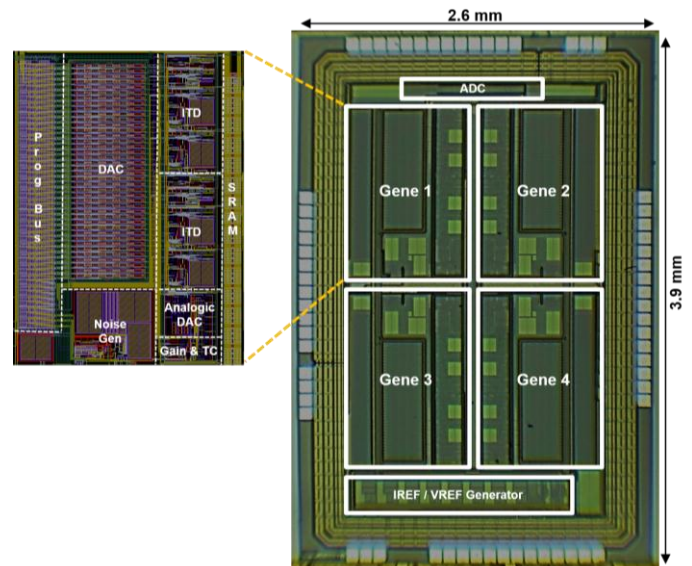

Fig. 2. Die micrograph of the 2.6 mm × 3.9 mm cytomorphic chip fabricated in an AMS 0.35 μm BiCMOS process. The left inset is a layout screen capture of one gene block (2x magnification).

concentration.

5) A current-mode low pass filter (LPF) circuit enables the **gain and dynamics of mRNA and protein production and degradation** to be represented.

6) **A stochastics circuit** intentionally amplifies analog Poisson noise in transistors to represent biological fluctuations in mRNA and protein concentrations at very low copy numbers and at relatively high noise levels. The Poisson nature of biochemical reaction fluxes automatically maps biological noise to electronic noise at high copy numbers and relatively low noise levels [13]. Thus, these circuits are most useful for reliably modeling highly stochastic and relatively low signal-to-noise-ratios in biological cells.

7) **ADCs and DACs** convert between analog currents and digital bits to enable our chips to communicate with each other via digital input/output (I/O), and with off-chip digital processors and computers.

Off-chip digital processors synergistically interact with our chips to carry out various functions: reading digital data from the chips; decoding the data; performing high-speed digital signal processing as necessary (e.g., scaling, diffusion, time delay, and error correction); encoding the data to create or modify molecular data packets via address and data strings; storing the programmable address connectivity amongst gene and protein circuits; and communicating data to other chips or to a computer. For simplicity, the data in this paper were collected with a data-acquisition board (NI PXI-6541) that interacted with our chip and with MATLAB on a computer. A high-performance FPGA (e.g. from the Xilinx Spartan family) could perform all of these functions in the future as well. **Shift registers, SRAM blocks, and switches on the chip** enable programmability of parameters (e.g., reaction rates, dissociation constants, Hill coefficients, and time constants) as well as connectivity.
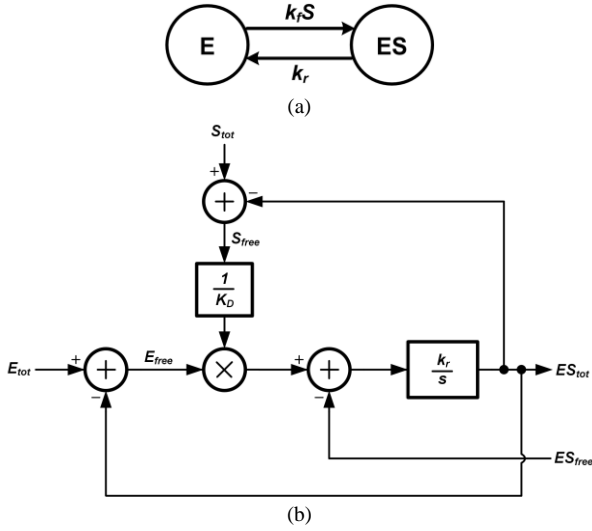
Fig. 3. (a) Simple representation of an enzyme-substrate binding reaction, $E + S \rightleftharpoons ES$. (b) Block diagram representation of the same reaction, based on "total" variables.

The organization of this paper is as follows: Section II describes the core building-block circuits of the chip along with chip measurement data. Fig. 2 shows a die micrograph of the chip that reveals the various building blocks. Section III discusses design considerations that are important in BiCMOS cytomorphic chip design. Section IV discusses how cytomorphic building-block circuits may be composed to simulate synthetic biochemical reaction networks and architects this composition for three concrete biological examples. Section V summarizes the overall specifications of the chip. Section VI concludes the paper by summarizing our contributions.

II. BUILDING-BLOCK CIRCUITS

This section describes the details of the seven building blocks that can be programmed to capture the essential dynamics of molecular basis-function circuits in cells. These blocks were implemented in a proof-of-concept VLSI chip fabricated in an AMS 0.35 μm BiCMOS process. As shown in Fig. 2, the chip contains identical gene blocks. To prove the functionality of on-chip building block circuits, the input-output characteristics for each basis-function circuit created on the chip was compared

with ideal simulation results produced by MATLAB. MATLAB was also used to perform chip programming and readout of chip data via a NI PXI-6541, data-acquisition board. To obtain values of chip data shown in this section, the outputs of ADC's from the chip were converted into actual current levels via measured scale factors.

A. Mass Action and Michaelis-Menten Reaction Block

Among various different approaches to represent gene-protein networks, including directed graphs, Bayesian networks, Boolean networks, differential equations, and master equations [20], our circuit schematics are based on ordinary differential equations (ODEs) that automatically incorporate Poisson stochastics into electron current fluxes [13]. Electron copy number is analogous to molecular copy number. We assume that the laws of mass action determine chemical reaction kinetics. The most fundamental molecular basis functions that need to be implemented are the rate equations of the following four elementary reactions:

1)   Production: $\phi \rightarrow E$
2)   Degradation: $E \rightarrow \phi$
3)   Association: $E + S \rightarrow ES$
4)   Dissociation: $ES \rightarrow E + S$

All complicated biochemical reaction networks can be decomposed into a series of these reactions. Fig. 3(a) shows a simple diagram to depict a network with an association (forward) and a dissociation (reverse) reaction. A block diagram we use to model the same network is shown in Fig. 3(b). For the two diagrams, typical symbols to describe Michaelis-Menten kinetics are used – E for enzyme, S for substrate, ES for enzyme-substrate complex, k_f and k_r for rate constants of forward and reverse reactions, respectively, and $K_D = k_r/k_f$ for the dissociation constant of ES. This block is the main building block of our system, which is capable of simulating all of the four elementary reactions listed above. The production and degradation reactions can often be more efficiently simulated by simpler blocks as well.

The block diagram of Fig. 3(b) basically solves the following three differential equations:
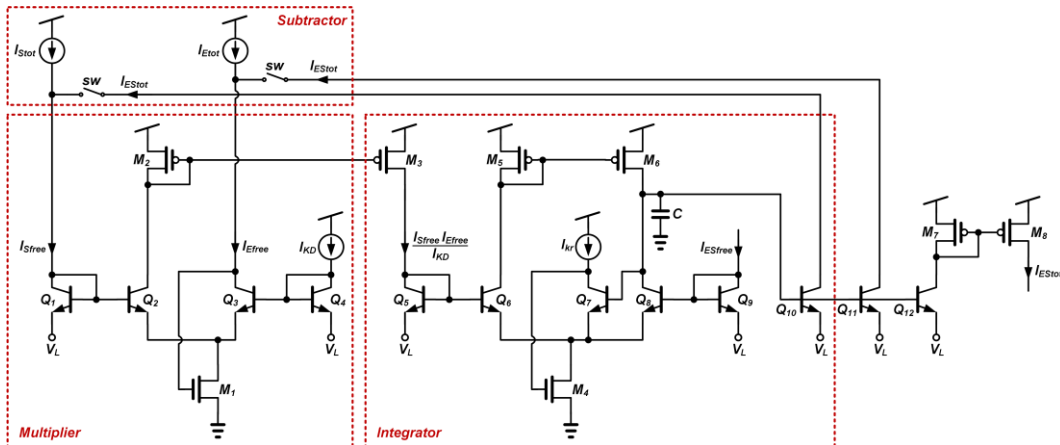

Fig. 4. Transistor schematic of the reaction block of Fig. 3(b).

$$\frac{d[ES_{tot}]}{dt} = k_f \left[E_{free} \right]\left[S_{free} \right] - k_r \left[ES_{free} \right] \qquad (1)$$

$$\frac{d[E_{free}]}{dt} = -k_f \left[E_{free} \right]\left[S_{free} \right] + k_r \left[ES_{free} \right] \qquad (2)$$

$$\frac{d[S_{free}]}{dt} = -k_f \left[E_{free} \right]\left[S_{free} \right] + k_r \left[ES_{free} \right]. \qquad (3)$$

Note that E_{tot}, S_{tot}, and ES_{tot} in Fig. 3(b) are "total" variables that include all downstream quantities necessary for "loading." They are used to compute the "free" variables, E_{free} and S_{free}. ES_{free} is the free amount of ES available that is not bound up in any downstream reactions. If ES is not used anywhere, ES_{tot} becomes equal to ES_{free}, a special case in which the steady state output of this network is given by

$$[ES_{tot}] = [E_{tot}] \left(\frac{[S_{free}]/K_D}{1 + [S_{free}]/K_D} \right). \qquad (4)$$

Fig. 4 shows the transistor schematic of the block diagram in Fig. 3(b), constructed based on BiCMOS current-mode circuits [13]. The circuit consists of a multiplier, an integrator, and two subtractors. The multiplier block composed of Q_1-Q_4 and M_1 calculates $I_{Sfree}I_{Efree}/I_{KD}$ from two variables I_{Efree} and I_{Sfree} and one parameter I_{KD}. The differential integrator block formed by Q_5-Q_{10} and M_4-M_6 gives

$$\frac{dI_{EStot}}{dt} = \left(\frac{I_{Sfree}I_{Efree}}{I_{KD}} - I_{ESfree} \right) \cdot \frac{I_{kr}}{C\phi_t} \qquad (5)$$

which is equivalent to (1). Multiple copies of its output current are created, which are used for subtraction to compute E_{free} and S_{free} or sent to other blocks. Each of the two subtractors is easily created by the two currents gathering into a node. Thus, current-mode circuits effectively implement multiplication and integration through Kirchhoff's voltage law (KVL) and addition and subtraction through Kirchhoff's current law (KCL).

Fig. 5(a) and 5(b) show the MATLAB simulation results and the chip data, respectively, of the block in Fig 3(b). It was assumed that $ES_{free} = ES_{tot}$. The steady-state values of E_{free} and ES_{tot} are plotted, while S_{tot} is varied from 100 pA to 1 μA and E_{tot} and K_D are fixed at 100 nA and 50 nA, respectively. It can be seen from Fig. 5(c) that with the concentration scale mapped from 100 nM to 115 nA for ES_{tot}, the MATLAB plot becomes a reasonable fit to the chip data. Although not shown, this Michaelis-Menten reaction block in the chip mimics the dynamics of the MATLAB model as well.

The dotted traces of Fig. 5 correspond to the results when substrate depletion (substrate being used up whenever it binds to an enzyme) is ignored, i.e., when the switch in Fig. 4 to subtract ES from S_{tot} is programmed as "off" and loading effects such as substrate depletion are ignored. By comparing the dotted and solid traces, the effect of substrate depletion can be clearly observed. Generally, this effect is ignored by many researchers,
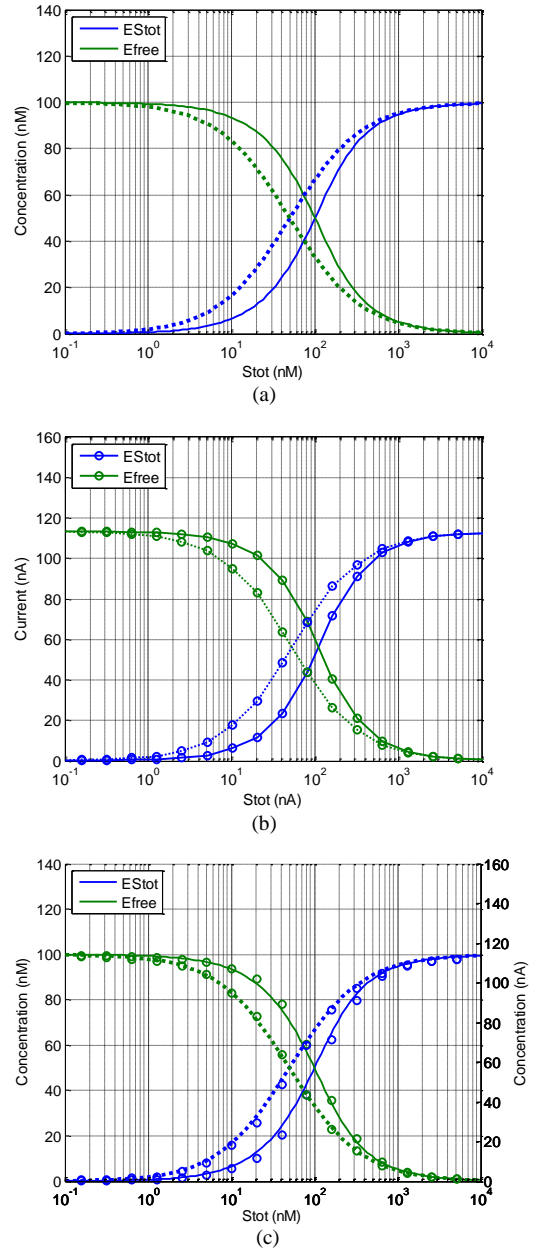

Fig. 5. (a) MATLAB and (b) chip data for the Michaelis-Menten reaction block in steady state, with (solid lines) and without (dotted lines) substrate depletion, which represents a typical loading effect. Lines shown in (b) are connection between points, not any fits. (c) Chip data (circles) plotted on top of MATLAB data (lines). Note that chip data are ADC outputs divided by the scale factors to obtain current levels. Chip parameters: I_{Etot} = 100 nA, I_{Stot} = 100 pA-10 μA, and I_{KD} = 50 nA.

under the assumption that substrate is much more abundant than enzyme, or to simplify analytic solutions. However, substrate depletion may exert a noticeable effect when the level of substrate is comparable to enzyme. Common protease, RNAase, ATPase, polymerase, or ribosome resources that are shared amongst many circuits exhibit "resource depletion" that manifests in an analogous fashion to the substrate-depletion example shown here. Fig. 5 shows that such depletion manifests as an increase in the effective K_D and Hill coefficient values [19]. It should be noted that, since physical cytomorphic circuits represent biomolecular circuits efficiently, this loading effect is
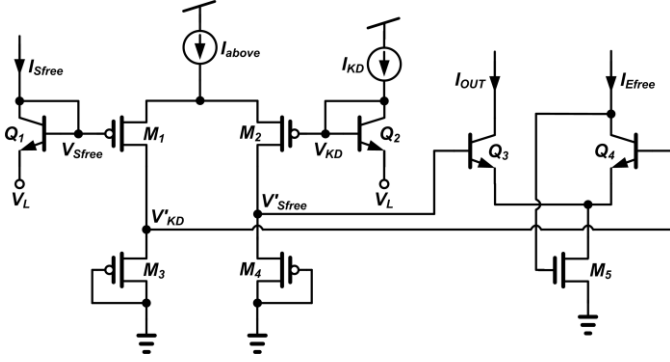
Fig. 6. A cytomorphic circuit implements a Hill coefficient greater than 1 by amplifying the voltage difference between V_{Sfree} and V_{KD}.

captured by the simple addition of a wire and a switch in Fig. 4.

B. Hill Block

The Hill coefficient is a parameter used in biochemistry to characterize the effect of cooperative binding. Note that (4), which is realized by the circuit of Fig. 4 has a Hill coefficient of 1. Fig. 6 shows how we can build steeper Hill-coefficient circuits. The voltage difference between the logarithm of I_{Sfree} and I_{KD} is amplified to yield

$$V'_{Sfree} - V'_{KD} = n\left(V_{Sfree} - V_{KD}\right) \qquad (6)$$

that then gives the term $(I_{Sfree}/I_{KD})^n$ in place of I_{Sfree}/I_{KD} in (4). This amplification is done by a differential amplifier operating in the above-threshold regime (created by the above-threshold current I_{above}), where its gain is set by the ratio between $\sqrt{W/L}$ of upper (M_1 and M_2) and lower (M_3 and M_4) PMOS transistors. By digitally programming the effective size of the upper transistors between 1x to 16x through switching bits, we can alter the Hill coefficient between 1 and 4. This range of Hill coefficients enables us to represent almost all cases of biological operation.

Fig. 7(a) and 7(b) show the MATLAB simulation results and the chip data, respectively, of the Michaelis-Menten reaction block in Fig 3(b), for different Hill coefficients. All variables are the same as in Section II-A, except that the Hill coefficient is set as 1, 2, or 4 respectively. As Fig. 7 reveals, the chip data exhibit the same behavior as the MATLAB model.

C. ITD Block

Each of the two ITD blocks shown in Fig. 2 simulates the dynamics of inducer-transcription factor binding and transcription factor-DNA binding. Fig. 8(a) is a simplified diagram of the ITD block depicting the key reactions among inducer molecules (*I*), transcription factors (*TF*), and DNA binding sites (*DNA*). The names of the molecules and bound complexes are self-explanatory. Both *TF* and *I-TF* are allowed to bind to DNA with programmably different binding affinities. The binding of an inducer to a transcription factor causes its binding affinity to DNA to typically change by a factor of 10 to 100 in cells. Some transcription factors (e.g., AraC) can act both as an activator and as a repressor, depending on whether
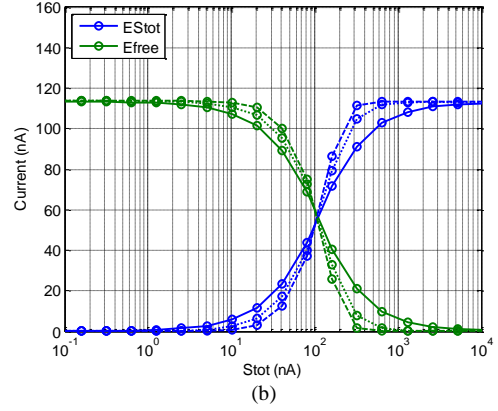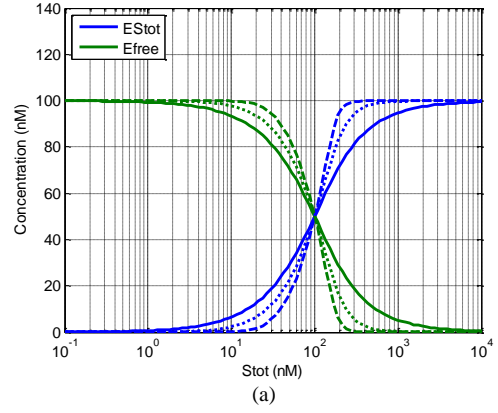

(a)

(b)

Fig. 7. (a) MATLAB and (b) chip data for the Hill block in steady state, for three Hill coefficients, 1 (solid lines), 2 (dotted lines), and 4 (dashed lines). Chip parameters: $I_{Etot} = 100$ nA, $I_{Stot} = 100$ pA-10 µA, and $I_{KD} = 50$ nA.

inducers are bound or not. All of these cases can be modeled by our ITD and analogic DAC blocks (described in the next section).

A more detailed block diagram representation of the same network is shown in Fig. 8(b). First, the MM_Static block models inducer-transcription factor binding and is a modified version of the circuit described in [14]. Since the dynamics of inducer and transcription factor binding is usually a few orders of magnitude faster than the dynamics between transcription factors and DNA binding [21], [22], such binding is assumed to reach steady state nearly instantaneously, and is therefore neglected, as in most models.

The two MM_Basic blocks represent the basic Michaelis-Menten reaction blocks shown in Fig. 3 and 4. The S_{tot} input for the two blocks is represented by TF_{bnd} and TF_{free}, equivalent to *I-TF* and *TF* in Fig. 8(a) respectively. Since a DNA binding site can be bound with either *I-TF* or *TF*, TF_{bnd} and TF_{free} represent "total substrate variables" for the DNA, and Fig. 8(a) and 8(b) effectively implement a competitive fan-out condition for the DNA binding site. The amount of *TF-DNA* from the bottom MM_Basic block is subtracted from DNA_{tot} to yield the total number of DNA sites available for *I-TF* binding in the top MM_Basic block. The net unbound DNA from the top MM_Basic block is the E_{free} input to the bottom MM_Basic block. It is worth noting that the top MM_Basic block explicitly uses the switch in Fig. 4 to achieve correct loading since its input is an E_{tot} input. In contrast, the
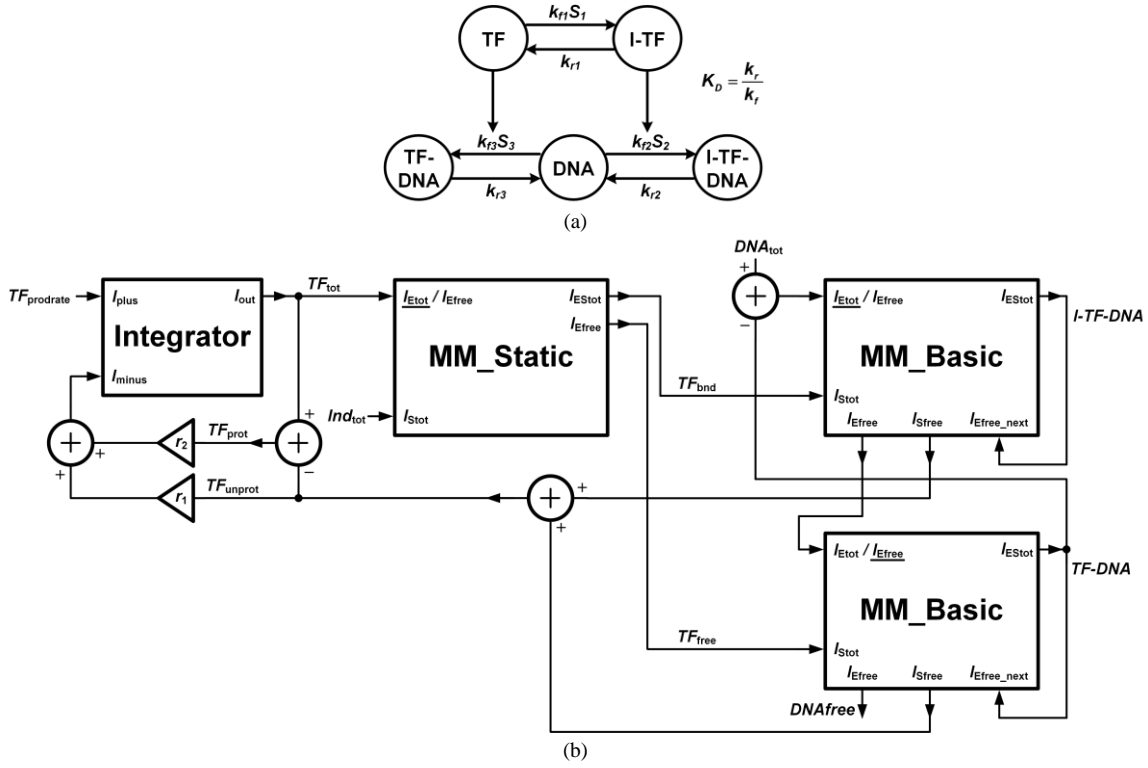
Fig. 8. (a) Simplified diagram showing the key reactions of the ITD block. (b) Block diagram of the ITD block. The integrator models the production and degradation of transcription factor; MM_Static models the steady-state behavior of inducer-transcription factor binding; the two MM_Basic blocks model the dynamics of transcription factor-DNA binding when the transcription factor is bound to an inducer (TF_{bnd} input) or not (TF_{free} input).

bottom MM_Basic block, does not use this switch since its input is already an E_{free} input.

The integrator block in Fig. 8(b) is a differential integrator with production-rate and degradation-rate inputs that determine the total amount of the transcription factor. A circuit similar to the integrator included in the MM_Basic block and described in [13] is used. We modeled the protection effect of DNA on transcription factor degradation [19]: The degradation rate of a transcription factor can drop when it binds to DNA because a certain part of it is hidden and it becomes harder for a protease to digest the transcription factor. Thus, in Fig. 8(b) different degradation rate constants (r_1 and r_2) multiply the amount of transcription factor that is unbound (TF_{unprot}) or bound (TF_{prot}) to DNA, respectively. The sum of the two multiplied results is the negative input of the integrator block.

Our ITD model was chosen because it captures important gene-protein dynamics seen in practice. Fig. 8(b) shows that it is constructed by a straightforward arrangement and wiring of blocks in Fig. 4 and 8. Here we list the assumptions inherent in our ITD model, which are explicitly or implicitly used by most other models that fit biological data as well, e.g., those described in [22]:

1) The binding of inducers to transcription factors occurs at a much faster rate than the rest of the network and can thus be assumed to be nearly instantaneous.
2) The affinity of inducers to transcription factors remains the same, regardless of whether transcription factors are bound to DNA or not.
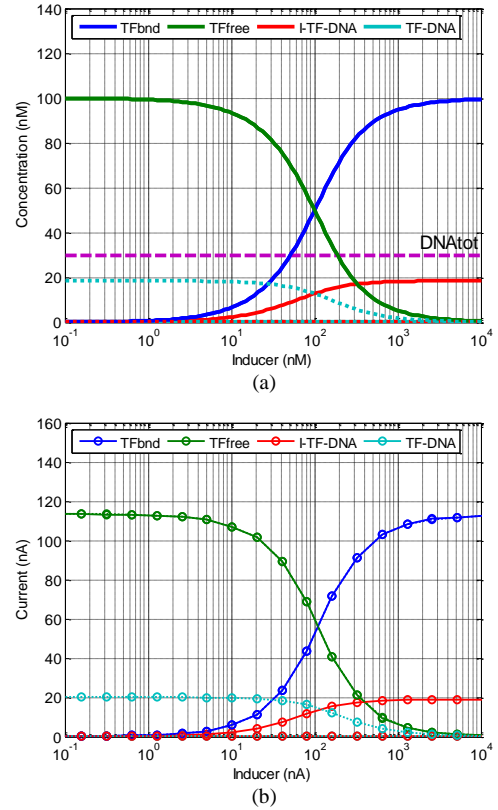3) The degradation of inducers is ignored; when a transcription

Fig. 9. (a) MATLAB and (b) chip data for the ITD block in steady state. Chip parameters: I_{Etot1} (TF_{tot}) = 100 nA, I_{Stot1} (Ind_{tot}) = 100 pA-10 μA, I_{KD1} = 50 nA. I_{Etot2} (DNA_{tot}) = 30 nA. Also, I_{KD2} = 50 nA and I_{KD3} = 10 μA for solid lines, and I_{KD2} = 10 μA and I_{KD3} = 50 nA for dotted lines. The dashed line in (a) indicates the level of DNA_{tot}, from which DNA_{free} can be estimated.
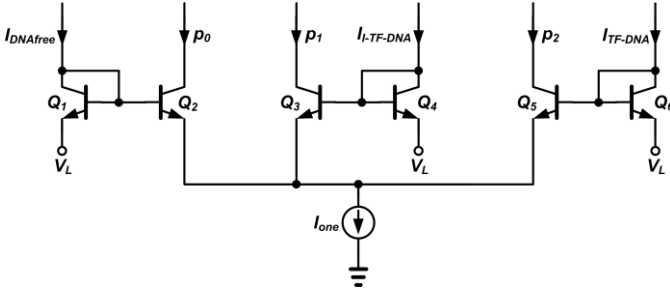
Fig. 10. Circuit used in the analogic DAC to compute the three probabilities (p_0, p_1, and p_2) of a DNA binding site.

factor degrades, its associated inducer is automatically freed.
4) When multiple binding sites for the same transcription factor exist on the DNA, each binding event is independent of other events.

Though it is not very useful for fitting biological data in most situations, it is possible to not make any of the above assumptions and to design more complex cytomorphic ITD circuits that capture other scenarios.

Fig. 9(a) and 9(b) show the steady-state simulation results of the ITD block using MATLAB and the chip, respectively, for equivalent parameter sets. As for the inducer-transcription factor binding, the same variables are used as in Section II-A. As for the transcription factor-DNA binding, $DNA_{tot} = 30$ nA, and two different parameter sets are used for K_{D2} and K_{D3}: 1) $K_{D2} = 50$ nA, $K_{D3} = 10$ µA (solid traces) and 2) $K_{D2} = 10$ µA, $K_{D3} = 50$ nA (dotted traces). The former is the case when only TF_{bnd} can bind DNA, and the latter is when only TF_{free} can bind DNA. The amount of DNA_{tot} is drawn in Fig. 9(a) as a dashed line, so readers can estimate the amount of $DNA_{free} = (DNA_{tot} - I\text{-}TF\text{-}DNA - TF\text{-}DNA)$. It can be seen that the chip produces outputs that are very similar to the plots generated by MATLAB.

D. Analogic DAC

Our models assume two sites in a DNA promoter where transcription factors can bind, which is typical for most microbes. Different transcription factors can bind different binding sites. The rate of mRNA synthesis varies depending on the probabilistic state of each binding site. In the ITD model in Fig. 8(a), binding sites can be bound by either *I-TF* or *TF*, or by neither. Correspondingly, there are three outputs of each ITD block, *I-TF-DNA*, *TF-DNA*, and DNA_{free}, which are output to the analogic DAC. Then, using the circuit shown in Fig. 10, the three binding probabilities of a site, p_0, p_1, and p_2, are computed as:

$$p_0 = I_{one}\left(\frac{I_{DNAfree}}{I_{DNAfree} + I_{I-TF-DNA} + I_{TF-DNA}}\right)$$

$$= I_{one}\left(\frac{I_{DNAfree}}{I_{DNAtot}}\right) \qquad (7)$$

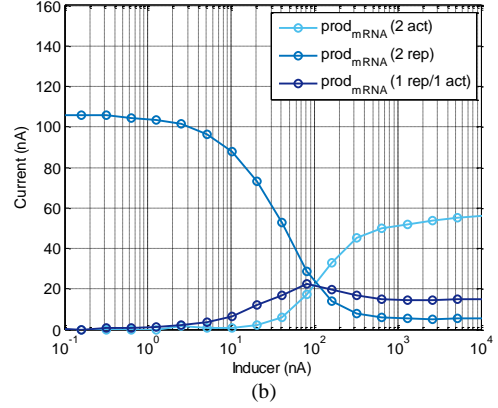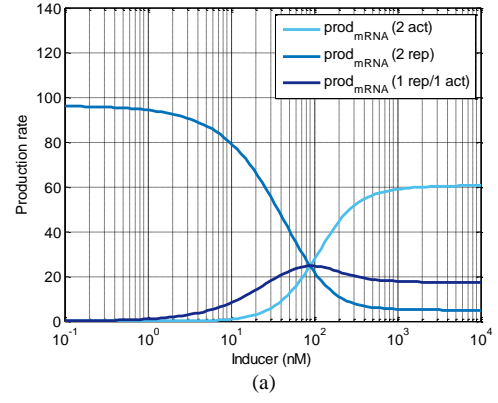$$p_1 = I_{one}\left(\frac{I_{I-TF-DNA}}{I_{DNAtot}}\right) \qquad (8)$$

(a)

(b)

Fig. 11. (a) MATLAB and (b) chip data for the analogic DAC in steady state. Chip parameters (two ITD blocks): I_{Etot1} (TF_{tot}) = 200 nA, I_{Stot1} (Ind_{tot}) = 100 pA-10 µA, I_{KD1} = 50 nA, I_{Etot2} (DNA_{tot}) = 30 nA, I_{KD2} = 50 nA and I_{KD3} = 10 µA. Chip parameters (Analogic DAC): 1) β_{11} = 100 nA, other β_{ij} = 0 (two activators), 2) β_{00} = 100 nA, other β_{ij} = 0 (two repressors), and 3) β_{01} = 100 nA, other β_{ij} = 0 (one repressor and one activator).

$$p_2 = I_{one}\left(\frac{I_{TF-DNA}}{I_{DNAtot}}\right) \qquad (9)$$

where I_{one} is a constant reference current which corresponds to 100% probability. An identical circuit is used for a second binding site, with p_0, p_1, and p_2 used in place of I_{one} such that all nine probabilities, p_{00}, p_{01}, p_{02}, p_{10}, p_{11}, p_{12}, p_{20}, p_{21}, and p_{22}, for two binding sites can be computed. For example, p_{00} is given by

$$p_{00} = I_{one}\left(\frac{I_{DNAfree1}}{I_{DNAtot1}}\right)\left(\frac{I_{DNAfree2}}{I_{DNAtot2}}\right). \qquad (10)$$

The net mRNA synthesis rate is determined by the sum of the probabilities of the promoter being in each of the nine states multiplied by the relative mRNA synthesis rate, from β_{00} to β_{22}, corresponding to that state [23]. That is,

$$prod_{mRNA} = \sum_{i=0}^{2}\sum_{j=0}^{2}\frac{p_{ij}\beta_{ij}}{I_{one}} \qquad (11)$$

where $p_{ij}\beta_{ij}/I_{one}$ can be obtained by using the multiplier included in Fig. 4.

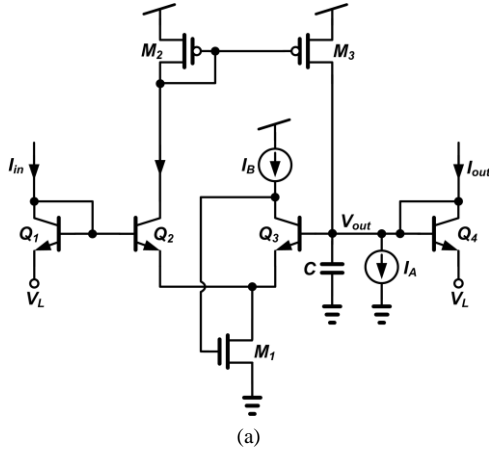Fig. 11(a) and 11(b) show the plots of the mRNA production

(a)

Fig. 12. Current-mode low-pass filter (LPF) circuit used to set the gain and the time constant of transcription (and translation).

rate simulated by MATLAB and the chip, respectively. Three different configurations were made for the analogic DAC: 1) every $\beta_{ij} = 0$ except $\beta_{11} = 100$ nA (i.e., two activators), 2) every $\beta_{ij} = 0$ except $\beta_{00} = 100$ nA (i.e., two repressors), and 3) every $\beta_{ij} = 0$ except $\beta_{01} = 100$ nA (i.e., one repressor and one activator). Since the parameters for the two ITD blocks were set such that *I-TF-DNA* increases as inducer level increases, for each configuration, the level of $prod_{mRNA}$ 1) increases; 2) decreases; and 3) increases and then decreases, as shown in both Fig. 11(a) (MATLAB) and 11(b) (Chip).

E. Gain & Time Constant Block

Fig. 12 shows a current-mode low-pass filter capable of setting the gain and time constant of transcription. The input current I_{in} is mRNA synthesis rate generated by the analogic DAC, the output current I_{out} is mRNA concentration, and the two programmable currents I_A and I_B set the gain (I_B/I_A) and the time constant ($C\phi_t/I_A$) [13]. That is,

$$\frac{I_{out}(s)}{I_{in}(s)} = \frac{I_B/I_A}{1 + s\left(C\phi_t/I_A\right)} \tag{12}$$

which is equivalent to the following differential equation:

$$\frac{d[mRNA]}{dt} = \alpha_{mRNA} - \gamma_{mRNA}[mRNA] \tag{13}$$

where $\alpha_{mRNA} = I_{in}I_B/(C\phi_t)$ and $\gamma_{mRNA} = I_A/(C\phi_t)$ are the production and degradation rate of mRNA, respectively. The gain term may also include a protein production-rate constant (whose unit is proteins/mRNA·s). In this case, I_{out} becomes the rate at which a protein is produced, which can directly be used as the $TF_{prodrate}$ input of an ITD block. Occasionally, the kinetics of mRNA and protein molecules are lumped into a single production rate and a degradation rate, under the condition that the time scale of protein dynamics is much slower than mRNA. Then, I_{out} corresponds to the protein concentration.

Fig. 13(a) and 13(b) show the MATLAB simulation data and the chip data, respectively. The gain & time constant block were
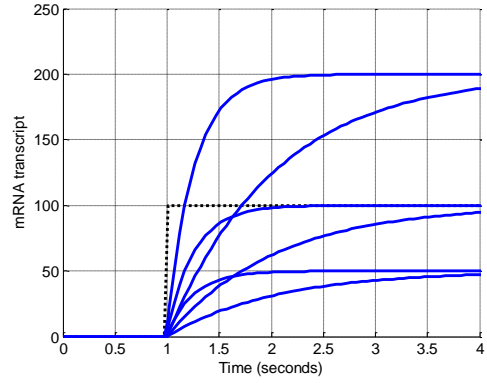
(a)
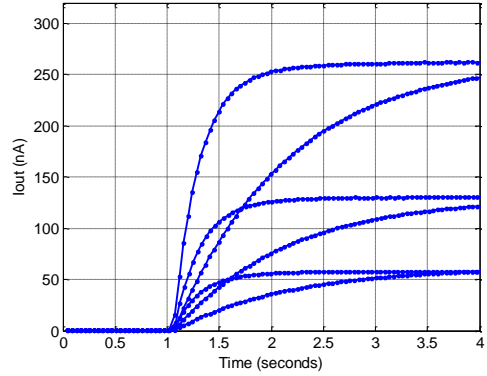
(b)

Fig. 13. (a) MATLAB and (b) chip data over time for the gain & time constant block, for three gains (0.5, 1, and 2) and two time constants (0.25s and 1s). The step function shown in (a) is the input to the block. Chip parameters: $C = 1$ μF and $(I_A, I_B) = (25, 12.5), (25, 25), (25, 50), (100, 50), (100, 100),$ and $(100, 200)$ (units are nA).

tested for three different gains, 0.5, 1, and 2, and two time constants, 0.25 s and 1 s, for a step input (shown as the dotted trace in Fig. 13(a)).

F. Noise Generator

Intrinsic noise in cells, also known as molecular noise, arises from the probabilistic arrivals and collisions of molecules in biochemical reactions. Such noise is well modeled by the Poisson shot noise of electronic current in subthreshold transistors with electron copy number analogous to molecular copy number and physical temperature representing biological temperature [13]. Important stochastic characteristics including the "burst factor" observed in genetic networks [24] are faithfully reproduced in cytomorphic circuits [8], [13], [15] as an effective current gain. In the gain & time constant block presented in Section II-E, the signal-to-noise ratio (SNR) at the output (I_{out}) is proportional to the size of the capacitor C and the current level; by adjusting the capacitor or current, any desired SNR can be achieved [13]. However, for SNR's below 15-25dB (which is the case when the number of molecules is relatively small) capacitor sizes and the current levels can be small in electronics deteriorating the reliability and controllability of the noise level [13], [15]. A noise generator which can create artificially high levels of noise is therefore desirable.

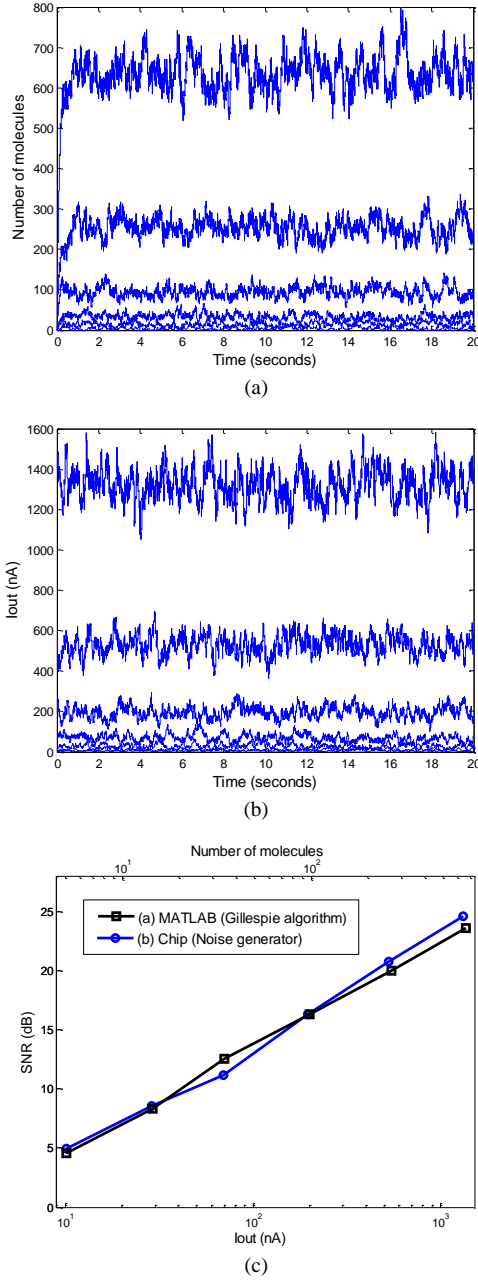The noise generator that we created on chip is similar to that

(a)

(b)

(c)

Fig. 14. Noise generated by (a) a MATLAB simulation using the Gillespie algorithm and (b) the noise generator block in the chip, for relatively low numbers of molecules. 1 nA of I_{out} is mapped to correspond to approximately two molecules. (c) Comparison of SNR obtained from the MATLAB data and the chip data.

reported in [15]. The major difference is that instead of using pseudo-random digital noise generation, our circuit produces noise by amplifying analog thermal noise that inherently exists in any transistor. The noise generator is implemented inside the gain & time constant block in Fig. 12. Noise is generated by the turning on and off of I_A using a "random" clock to charge and discharge the capacitor node. The clock exhibits pure Poisson characteristics with a duty cycle of 0.5, and its mean frequency is proportional to the output current I_{out} and the gain factor β_{SNR} [13], [15]. By increasing β_{SNR}, artificially generated shot noise is increased, which mimics the high levels of noise seen in biology.

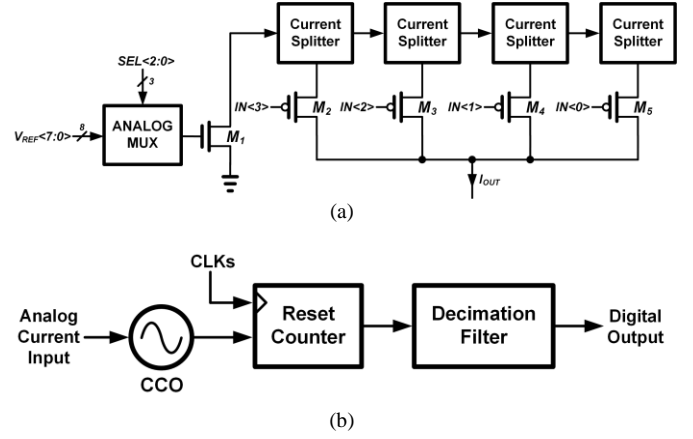Fig. 14(a) and 14(b) show noise generated in MATLAB using

(a)

(b)

Fig. 15. Block diagram of (a) the digital-to-analog converter (DAC) and (b) the analog-to-digital converter (ADC).

a Gillespie algorithm for molecular production and degradation [1] and using the noise generator in the chip, respectively, for relatively low numbers of molecules. In this simulation, we arranged for 1 nA of I_{out} to correspond to approximately two molecules (per cell). The comparison between the SNR obtained from the MATLAB data and the chip data is shown in Fig. 14(c). It can be seen that the artificial noise generator implemented in the chip generates noise that is very similar to that from a Gillespie algorithm, both visually and quantitatively. The good agreement with the Gillespie algorithm, a widely used method to model biological noise, is in accord with noi7se measurements and theories of Poisson noise in transistors and in cells [8], [13], [24], [25]. Such work allows us to achieve fast-and-accurate stochastic Gillespie simulations on custom analog chips. Thus we can enable improvements over methods such as tau leaping [37], Langevin-noise addition, or limited-moment simulations of actual Fokker-Planck equations [38, 39] that necessarily require accuracy to be compromised for speed on general-purpose digital computers

G. DAC and ADC

The analog circuits inside the cytomorphic chip primarily use currents to represent variables and parameters. On the other hand, chip-to-processor and chip-to-computer communication is done digitally. It is thus important to have a scheme to convert between analog currents and digital bits.

The gene block in Fig. 2 includes 40 digital-to-analog converters (DACs), which requires us to design a power- and area-efficient DAC, while preserving reasonable precision, dynamic range, and linearity. To this end, we first took a stable bias current of 11.5 µA created by an on-chip current generator and then used a current splitter similar to that described in [26] to divide the current into halves, 13 times in succession; the 1st, 3rd, 5th, 7th, 9th, 11th, and 13th bias voltage outputs served as global reference voltages for the whole chip.

Fig. 15(a) shows the block diagram of a single DAC. It selects one of the eight reference voltages ($V_{REF}<7:0>$) with a 3-bit digital input ($SEL<2:0>$) and an 8-to-1 analog multiplexer, applies it to the gate of M_1 to convert it back to a current, and additionally divides this current into halves, four times in

succession. The sum of the four resulting currents, turned on or off by another 4-bit digital input (*IN<3:0>*), is the output current of the DAC. In summary, each DAC uses seven bits (stored in SRAM or shift registers) to determine its output current level – three bits to select a range in an exponential fashion and four bits to select a value in a linear fashion. This mechanism of coarse and fine selection enables an experimentally measured output dynamic range of the DAC from 29 pA to 21 uA; we only use a range of 100 dB from 100 pA to 10 uA in our circuits to ensure robustness to leakage and voltage-headroom effects. Furthermore, distributing reference voltages instead of currents allows us to save area and power which would otherwise be consumed by numerous current mirrors and distributing wires. We were careful about using sufficiently large transistors for converting between reference voltages and currents to mitigate the effect of mismatch, and about using thick-and-wide metal wires to minimize voltage drops.

The number of analog-to-digital converters (ADCs) depends on the maximum number of variables that need to be monitored or digitally processed off chip. For example, one may want to monitor 10 to 50 variables per chip. Hence, the top consideration when choosing an ADC topology is to minimize the number of output bits, so that the pin count devoted to this ADC on the chip is minimized. Therefore, we choose a current controlled oscillator-based (CCO-based) ADC shown in Fig. 15(b), which is a first-order noise-shaping oversampling ADC [27]. The advantages of this ADC include compactness that comes from its simple architecture and its inherent noise shaping property. The current-to-frequency nonlinearity in the CCO, the major drawback of this architecture, can be easily compensated for by the use of a mapping table that is stored off chip. In our implementation, each CCO-based ADC operates at a sampling frequency of 5 MHz, and generates one-bit digital outputs that are averaged and down-sampled off chip to achieve analog-to-digital conversion.

III. DESIGN CONSIDERATIONS IN BiCMOS CYTOMORPHIC DESIGN

We chose to use a BiCMOS process technology to leverage good matching, high Early voltage, and most importantly, the nearly ideal wide-dynamic-range exponential current-voltage (I-V) characteristics of bipolar transistors, an essential basis function in log-domain cytomorphic chips. We achieved 100 dB dynamic range, from 100 pA to 10 µA, for most on-chip variables represented as current levels.

Extra care is needed to avoid potentially negative effects of bipolar transistors due to finite base currents effects and forward biasing of the base-collector junction. For example, we used the modified circuit shown in Fig. 16 to solve practical problems that we encountered in the circuit of Fig. 12: In this circuit, M_3 is inserted as a buffer (source follower) which supplies the base currents of Q_1 and Q_2 on behalf of I_{in}. This buffer is also useful in that it can rapidly charge the base node, thereby reducing parasitic-capacitance effects if any. The npn transistor Q_5 minimizes base-current effects in Q_3 and Q_4 without any body-effect voltage losses, while the PMOS transistor M_4 (whose source is tied to the well) minimizes base-current losses in Q_5
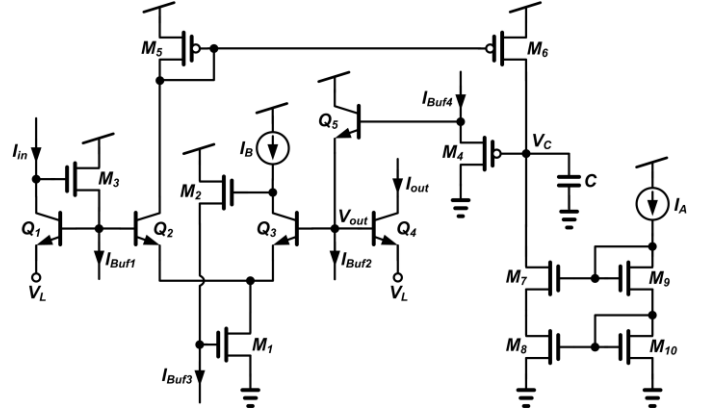

Fig. 16. Modified LPF to solve practical problems in the circuit of Fig. 12.

itself, while drawing no current from the V_C node. The overall cascaded buffer formed by Q_5 and M_4 also keeps V_C at a good DC operating point, which is necessary for the current mirror coupled to V_C to function in saturation. The cascode current mirror formed by M_7-M_{10} changes the direction of I_A since every current on chip is assumed to be supplied as a source current. Finally, M_2 serves as a buffer that enables Q_3 to always operate in its forward active region. The techniques described in this paragraph were also applied to all current-mode circuits in the chip, e.g., multipliers and integrators shown in Fig. 4.

IV. SIMULATION OF SYNTHETIC GENETIC CIRCUITS

To show the potential of our chip as a simulation tool for quantitatively modeling bio-molecular circuits, we emulated the models of several well-known synthetic genetic networks. To do so, we composed and programmed the basis-function circuits described in Section II to architect the network. This section illustrates how the simulations of three bio-molecular networks – the repressilator [28], a feed-forward loop network [29], and a delay-induced oscillator [30] – can be implemented with the chip.

A. Repressilator

The repressilator network shown in Fig. 17(a) is a circuit that helped pioneer the field of synthetic biology [28]. It creates a self-sustaining oscillation by using three repressors in a ring form, similar to a ring oscillator made with three cascaded inverters in electronic systems.

First, from the two dimensionless equations and parameters given in Box 1 of [28], we derived the original differential equations that determine the dynamic behavior of mRNA and protein concentrations of each repressor species:

$$\frac{dm_i}{dt} = -5.8 \times 10^{-3} m_i + \frac{0.5}{1 + \left(p_j / 40 \right)^2} + 5 \times 10^{-4} \quad (14)$$

$$\frac{dp_i}{dt} = -1.2 \times 10^{-3} p_i + 116 \times 10^{-3} m_i \quad (15)$$

Next, scale factors for time and concentration scaling were selected. The time constant for transcription factor-DNA
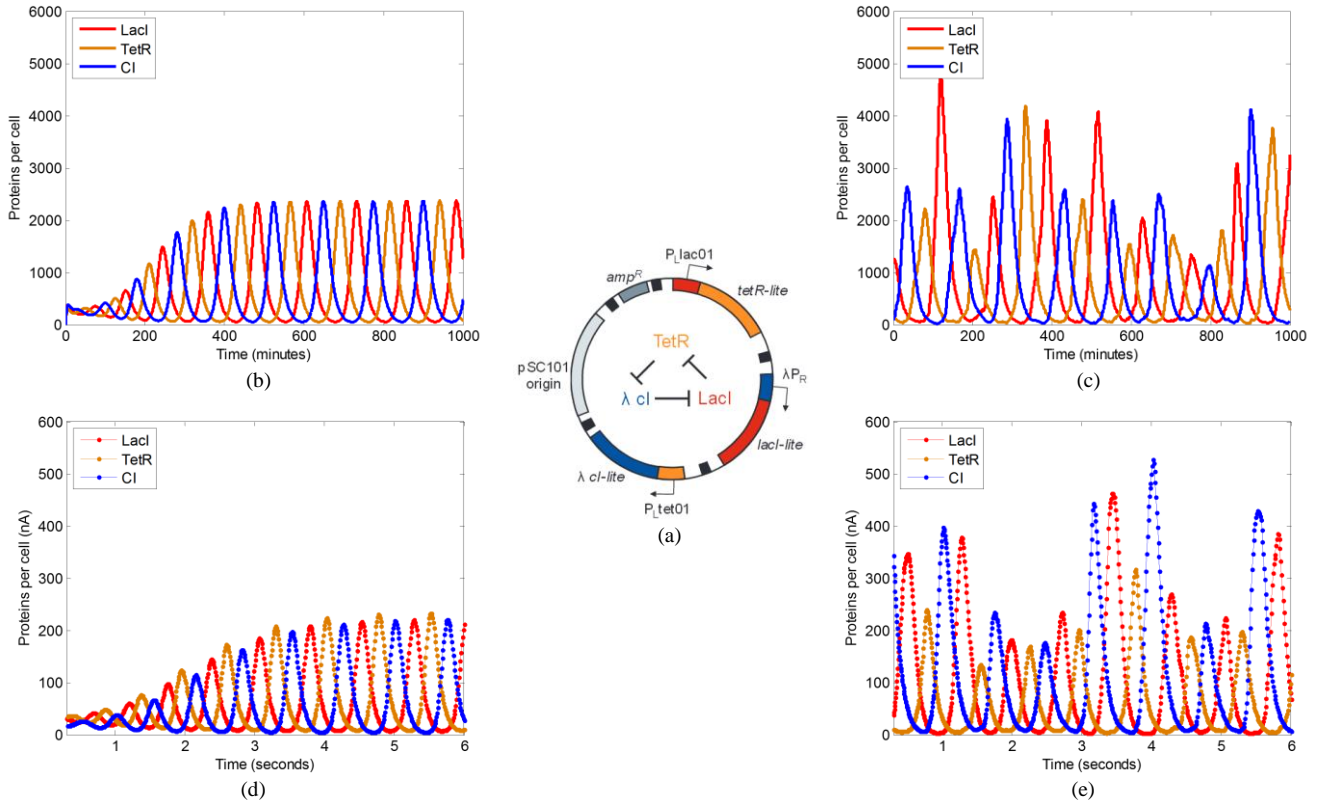
Fig. 17. (a) Repressilator circuit [28]. (b) Deterministic and (c) stochastic simulation results of MATLAB, using the mathematical model provided in [28]. (d) Deterministic and (e) stochastic simulation results of the chip, which is programmed using the parameters in Table I. (a) Reprinted by permission from Macmillan Publishers Ltd: [*Nature*] (M. B. Elowitz and S. Leibler, "A synthetic oscillatory network of transcriptional regulators," *Nature*, vol. 403, no. 6767, pp. 335–338, Jan. 2000), Copyright 2000.

binding is ~1.25 ms in the chip (when $C = 50$ pF and $I_{kr} = 1$ nA are used for the MM_Basic blocks in the ITD circuit), and the mRNA time constant is assumed to be 20 times bigger (25 ms) such that the former dynamics can be seen as nearly instantaneous. Then, $1/(5.8 \times 10^{-3}) = 172$ s of biological time is mapped into 25 ms of electronic time, corresponding to approximately 7000x speedup. Concentration scaling was performed such that 10 monomers of protein or mRNA were mapped into 1 nA of current in the chip. Based on these scaling parameters, all other biological parameters were scaled to electronic chip parameters, as summarized in Table I.

After programming these parameters and the connectivity of the circuit into the SRAM and shift registers of the chip, we simulated the network on the chip. Fig. 17(b) and 17(c) show the deterministic and stochastic simulation results of MATLAB, respectively, using the biological values given in Table I; Fig. 17(d) and 17(e) show the deterministic and stochastic simulations results of the chip, respectively, using the chip parameters given in Table I. For the stochastic simulation, the noise generator in Section II-F was turned on to produce artificial noise in each mRNA concentration. It can be seen that the waveforms from MATLAB and the chip are highly correlated, with the scale factors close to those described in the previous paragraph. As opposed to software simulations, running stochastic simulations does not increase the simulation time of our highly parallel circuits.

Parameter	Biological Value		Chip Value
mRNA degradation rate	5.8×10^{-3} s^{-1}	$\rightarrow$	40 s^{-1}
Protein degradation rate	1.2×10^{-3} s^{-1}	$\rightarrow$	8 s^{-1}
Transcription rate (fully induced)	0.5 transcripts/s	$\rightarrow$	346 nA/s
Transcription rate (repressed)	0.5×10^{-4} transcripts/s	$\rightarrow$	346 pA/s
Translation rate	116×10^{-3} proteins/(mRNA·s)	$\rightarrow$	800 s^{-1}
Dissociation constant	40 proteins	$\rightarrow$	4 nA
Hill coefficient	2	$\rightarrow$	2

Block	Chip Parameters
ITD – Integrator	I_{plus} = mRNA output of adjacent repressor, $I_{r1} = 200$ nA, $I_{r2} = 200$ nA, $C = 1$ μF
ITD – MM_Static	$I_{Stot1} = 0$, $I_{Etot1} = TF_{tot}$, $I_{KD1} = 10$ μA, $Hill1 = 1$
ITD – MM_Basic (upper)	$I_{Stot2} = TF_{bnd}$, $I_{Etot2} = 1$ μA, $I_{KD2} = 10$ μA, $I_{kr} = 1$ nA, $C = 50$ pF, $Hill2 = 1$
ITD – MM_Basic (lower)	$I_{Stot3} = TF_{free}$, $I_{Etot3} = I_{Efree}$ of MM_Basic (upper), $I_{KD3} = 4$ nA, $I_{kr} = 1$ nA, $C = 50$ pF, $Hill3 = 2$
Analogic DAC	$\beta_{00} = 865$ nA, $\beta_{20} = 865$ pA, other $\beta_{ij} = 0$
Gain & time constant	$I_A = 1$ μA, $I_B = 1$ μA, $C = 1$ μF

B. Feed-Forward Loop Network

The feed-forward loop network shown in Fig. 18(a) has two repressors, LacI and TetR, that repress the expression of yEGFP, and TetR also represses the production of LacI [29]. The inputs of the network that regulate yEGFP expression are ATc and
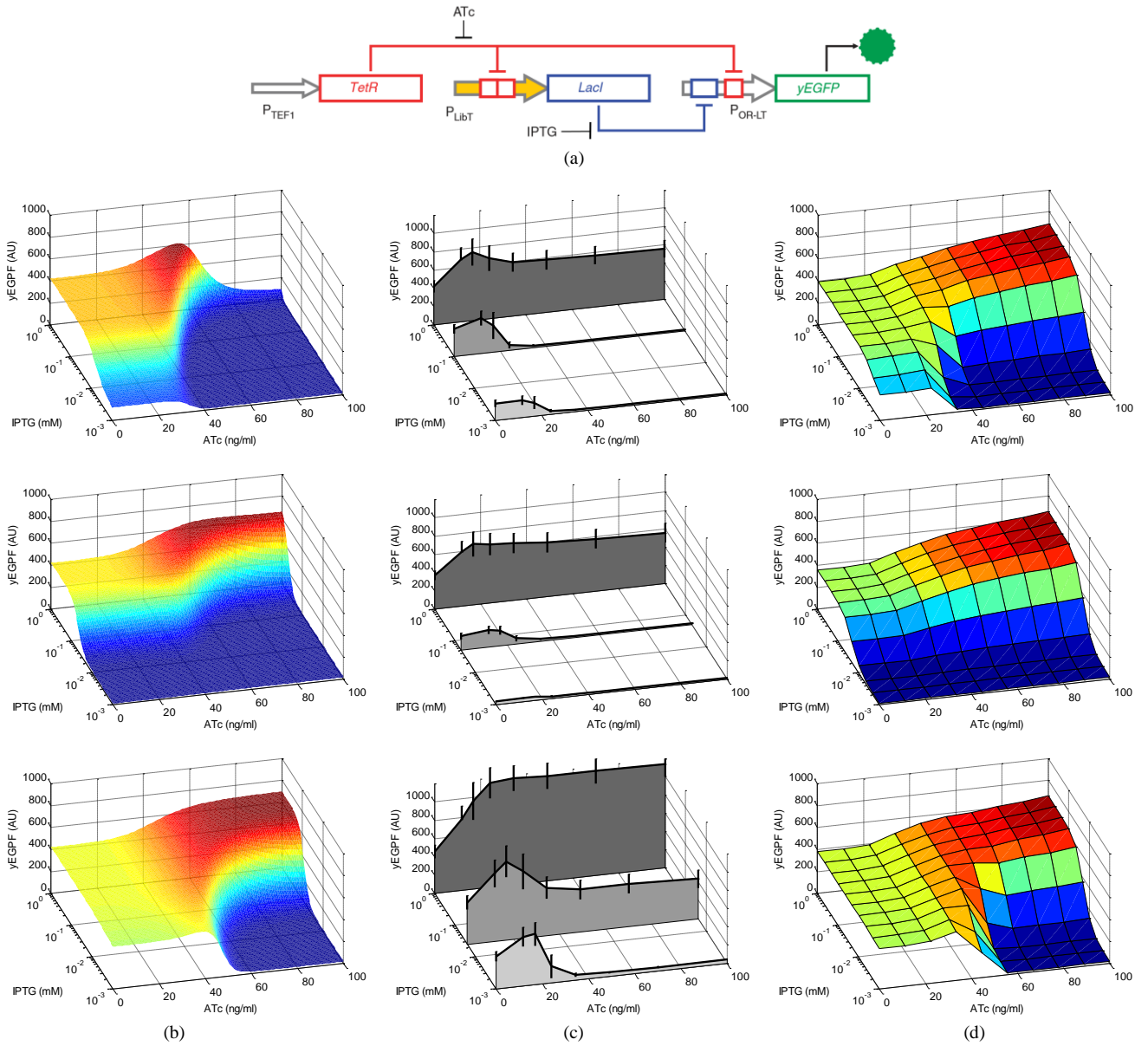
Fig. 18. (a) Feed-forward loop network [29]. (b) MATLAB simulation results of the mathematical model and (c) experimental data from the feed-forward loop network constructed in *S. cerevisiae*, for three different promoters, TX, T8, and T18 [29]. (d) Simulation results of the chip; the tetramerization and concentration-limiting functions were implemented in MATLAB using molecular data packets from the chip. (a)-(c) Reprinted by permission from Macmillan Publishers Ltd: [*Nature Biotechnology*] (T. Ellis, X. Wang, and J. J. Collins, "Diversity-based, model-guided construction of synthetic gene networks with predicted functions," *Nature Biotechnology*, vol. 27, no. 5, pp. 465–471, May 2009.), Copyright 2009.

IPTG inducers. Experiments were conducted with three different TetR-regulated promoters to study the effect of promoter strength.

Just as in the repressilator simulation of Section IV-A, concentration scale factors and chip parameters were determined from mathematical models and parameters described in the supplementary material of [29]. 1 ng/ml of ATc, 1 µM of IPTG, 1 AU (arbitrary unit for fluorescence level) of yEGFP, and 1 nM of repressor protein concentration were mapped to 1 nA of current. Each biological model that represented inducer-repressor binding or repressor-DNA binding was approximated on the chip using a dissociation constant and a Hill coefficient. In addition, to prevent conditions where LacI tetramer concentration becomes excessively high due to the $[\text{LacI}]^4$ term

in the mathematical model, we assumed an upper concentration bound of 1 µM (which is consistent with the typical values given in [21]). To accomplish these goals, MATLAB read the value of LacI monomer concentration from a first gene block on our chip, processed it with tetramerization and concentration-limiting functions, and fed the result back to the chip as the I_{Etot} input of the MM_Static portion of a second gene block. This example illustrates how our chip can be synergistically used with off-chip tools such as MATLAB or an FPGA that can extend its capabilities via traditional software if needed. Mathematical models yield better fits of biological data if concentration-limiting functions are not applied, but only at unphysically high values of LacI that are not implementable either in biology or on our chips over a dynamic range of even 100 dB. Thus, our chip
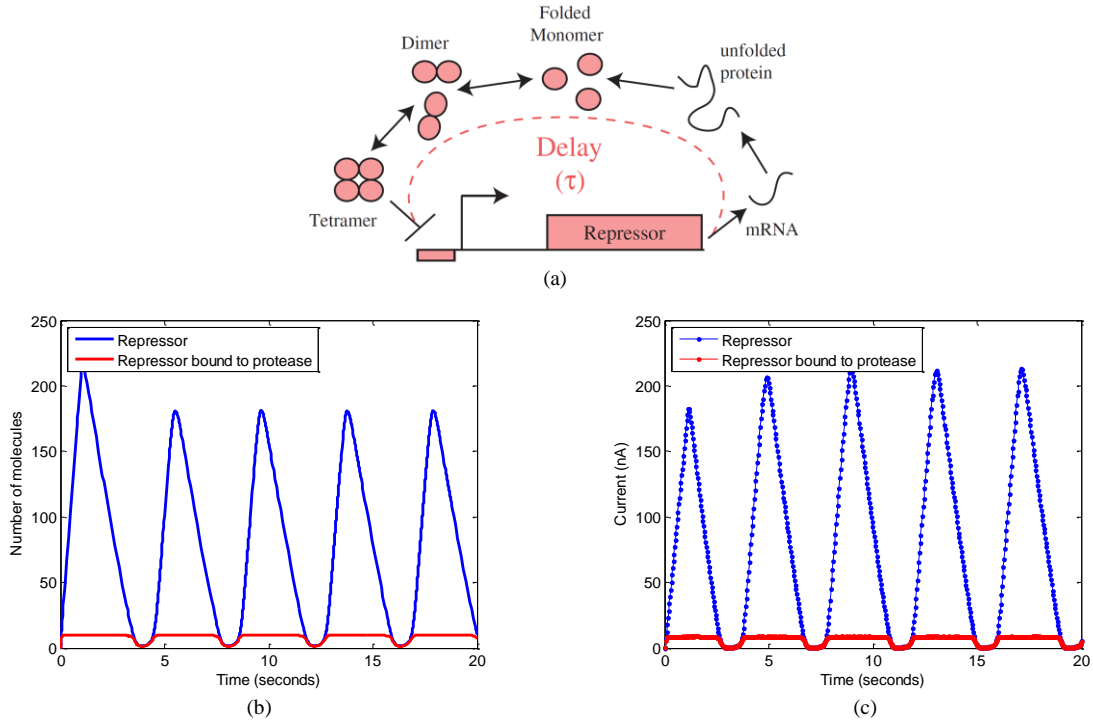
Fig. 19. (a) Delay-Induced Oscillator Network [30]. (b) MATLAB simulation results of the mathematical model in (16) and (c) chip simulation results. (a) Reprinted figure with permission from [W. Mather, M. Bennett, J. Hasty, and L. Tsimring, "Delay-induced degrade-and-fire oscillations in small genetic circuits," *Phys. Rev. Lett.*, vol. 102, p. 068105, Feb 2009.] Copyright 2009 by the American Physical Society.

automatically enables discovery of unphysical parameters in mathematical models.

Fig. 18(b), 18(c), and 18(d) show MATLAB simulation results from a mathematical model, experimental results from the network constructed in *S. cerevisiae*, and chip simulation results. In all cases, where concentrations of molecules were at physically plausible levels, our chip produced results that matched the biological data at least as well as the published mathematical model (the average percentage error for six cases was 64% in the published mathematical model [29] vs. 62% in our chips). In one case where concentration limiting functions were applied, the chip produced lower average percentage error than the mathematical model (24% vs. 34% for the top row of Fig. 18 (b), (c), and (d) at an IPTG concentration of 1 mM). In other cases where the concentration limiting functions were applied, both the chip and mathematical models performed poorly (errors that exceeded 200% in both cases).

C. Delay-Induced Oscillator

A pure delay is hard to implement efficiently with analog continuous-time systems since it requires an infinite number of state variables or a Pade approximant circuit. However, a pure delay is extremely easy to implement with digital systems. Although our system exploits analog and stochastic computation for its computationally intensive portions, our chips communicate digitally with external devices via on-chip ADCs and DACs. Thus, we have the flexibility to add digital basis functions like delays to supplement our on-chip analog basis functions. To demonstrate such flexibility, we simulated a delay-induced oscillator shown in Fig. 19(a), where the time delay in the negative feedback loop due to transcription,

translation, protein folding, and multimerization causes oscillations in the autorepression network [30]. The model in [30] is described by

$$\frac{dr}{dt} = \frac{\alpha}{1 + \left(r_\tau / C_0\right)^2} - \frac{\gamma_r}{1 + r/R_0} - \beta r \qquad (16)$$

where r is the number of repressor molecules, r_τ is a delayed version of r (i.e., $r_\tau(t) = r(t-\tau)$), τ is the delay time, α is the production rate of the repressor, C_0 is the dissociation constant for repressor-DNA binding, γ_r is the maximum rate of degradation due to protease, R_0 is the dissociation constant for repressor-protease binding, and β is the rate of degradation due to dilution.

This network was simulated by the chip with the following configuration: One ITD block in Fig. 8(b) provides an integrator to model production and degradation of repressor. The I_{r1} and I_{r2} inputs of the integrator correspond to γ_r and β, respectively. MM_Static is utilized to mimic the binding between repressor and protease and TF_{bnd} and (TF_{tot}-TF_{bnd}) are directed to the I_{r1} and I_{r2} sides of the integrator, respectively. The output of the integrator generates r, which is conveyed off chip as digital bits via the operation of the ADC. MATLAB reads the value of r, delays it, and programs the chip to send the delayed value to another ITD block, where repressor-DNA binding can be computed via an MM_Static circuit within the ITD. Finally, the TF_{free} output of the latter ITD block is sent to the $TF_{prodrate}$ input of the first ITD block to complete the auto-repressory feedback loop.

Fig. 19(b) and 19(c) show the MATLAB and chip simulation

TABLE II
TYPICAL PARAMETER VALUES IN CELLULAR NETWORKS

Parameter	Bacteria (*E. coli*)	Yeast (*S. cerevisiae*)	Mammalian cell (*Human Fibroblast*)
Cell volume	0.5–5 μm^3	20–160 μm^3	100–10000 μm^3
Concentration of one molecule/cell	~1 nM	~1 pM	~0.1 pM
Cell cycle time	20–40 min	70–140 min	15–30 hr
Transcription time	~1 min	~1 min	~30 min
Translation time	~2 min	~2 min	~30 min
mRNA lifetime	2–5 min	10 min–1 hr	10 min–10 hr
I-TF binding time	~1 ms	~1 s	~1 s
TF-DNA binding time	~1 s		
Concentration for a signaling protein	10 nM–1 μM	10 nM–1 μM	10 nM-1 μM
Signal-to-noise ratio	0-20 dB	0-20 dB	0-20 dB

TABLE III
PERFORMANCE CHARACTERISTICS OF THE CHIP

Parameter	Value
Technology	AMS 0.35 μm BiCMOS
Supply voltage	3.3 V
Dynamic range of variables (DAC output)	100 dB (100 pA–10 μA)
Number of DACs per gene block	40
Hill coefficient	1–4
Time constant for TF-DNA binding	1.25 ms–12.5 ms
Time constant for mRNA, protein	1.25 ms–12.5 s
Programming clock frequency	1 MHz
Programming time	0.3 ms
ADC sampling clock frequency	5 MHz
ADC readout time	2 ms–20 ms
ADC input range	1 nA–10 μA
Number of ADCs per chip	12
Signal-to-noise ratio	4–35 dB
Power consumption	< 10 mW
Chip size	2.6 mm × 3.9 mm

results, respectively. The parameters used in MATLAB are $\tau = 1$ s, $\alpha = 300$, $C_0 = 10$, $\gamma_r = 80$, $R_0 = 1$, and $\beta = 0.1$ (same as Fig. 2 presented in [30]). Their corresponding chip parameters were also selected, with a one-to-one time mapping and 1 molecule-to-1 nA concentration mapping. Taking these scale factors into account, it can be seen that fairly similar results were produced by MATLAB and by our chip.

V. SPECIFICATIONS OF THE CHIP

Table II shows typical values of parameters in cellular networks that help us set the design requirements of our system [21], [22], [31]. For example, to simulate a bacterial genetic network, programmable timescales of 70 dB dynamic range, at least 40 dB dynamic range of protein concentration, and 20 dB of signal-to-noise ratio are necessary. When studying the dynamics of transcriptional networks, the dynamics of inducer-transcription factor binding can be well approximated to be at steady state [22].

Table III summarizes the important performance characteristics of the chip. The values of the parameters suggest that our chip can model the wide range of parameters found in cells.

Digitally programmable analog systems have been reported [32-34, 41-54], and have shown tremendous potential in medical and computing applications [13]. Work described in [8,13,40] contains an extensive discussion of the pros and cons of analog vs. digital computation including issues relevant to speed, precision, power, flexibility, and programmability. Purely custom digital implementations have also been used to model protein folding [35] or to model visual neuronal networks [36]. Our work suggests that prior work on deterministic analog computation could be efficiently extended to deterministic and stochastic simulations of living cells via our digitally programmable cytomorphic circuits.

VI. CONCLUSION

Our work suggests that biological design, simulation, and analysis of circuits in living cells, which is very important in synthetic and systems biology, can greatly benefit from a cytomorphic transistor circuit approach. Such an approach captures the analog, digital, probabilistic, dynamic, stochastic, nonlinear, non-modular, and complex network and circuit behavior of biochemical reaction networks on compact transistor circuits very efficiently and naturally. Our chip's circuits function over more than five orders of magnitude of molecular concentration, which is more than adequate for representing the dynamic range of any given protein, DNA, RNA, or small-molecule state variable in cells.

REFERENCES

[1] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," *The Journal of Physical Chemistry*, vol. 81, no. 25, pp. 2340–2361, 1977.

[2] M. A. Gibson and J. Bruck, "Efficient exact stochastic simulation of chemical systems with many species and many channels," *The Journal of Physical Chemistry A*, vol. 104, no. 9, pp. 1876–1889, 2000.

[3] D. T. Gillespie, "Approximate accelerated stochastic simulation of chemically reacting systems," *The Journal of Chemical Physics*, vol. 115, no. 4, pp. 1716–1733, 2001.

[4] M. Schwehm, "Parallel stochastic simulation of whole-cell models," *Proceedings of the 2nd International Conference on Systems Biology (ICSB 2001)*, pp. 333–341, 2001.

[5] J. R. Karr, J. C. Sanghvi, D. N. Macklin, M. V. Gutschow, J. M. Jacobs, B. Bolival, N. Assad-Garcia, J. I. Glass, and M. W. Covert, "A whole-cell computational model predicts phenotype from genotype," *Cell*, vol. 150, no. 2, pp. 389–401, Jul. 2012.

[6] S. L. Spencer, S. Gaudet, J. G. Albeck, J. M. Burke, and P. K. Sorger, "Non-genetic origins of cell-to-cell variability in TRAIL-induced apoptosis," *Nature*, vol. 459, no. 7245, pp. 428–432, May 2009.

[7] K. Lewis, "Persister cells," *Annual Review of Microbiology*, vol. 64, no. 1, pp. 357–372, 2010.

[8] R. Sarpeshkar, "Analog Synthetic Biology," *Philosophical Transactions of the Royal Society A,* 372:20130110, 2014. http://dx.doi.org/10.1098/rsta.2013.0110.

[9] A. Wagner, "Energy constraints on the evolution of gene expression," *Molecular Biology and Evolution*, vol. 22, no. 6, pp. 1365–1374, 2005.

[10] D. Del Vecchio, A. J. Ninfa, and E. D. Sontag, "Modular cell biology: retroactivity and insulation," *Molecular Systems Biology*, vol. 4, no. 1, 2008.

[11] S. Cardinale and A. P. Arkin, "Contextualizing context for synthetic biology–identifying causes of failure of synthetic biological systems," *Biotechnology Journal*, vol. 7, no. 7, pp. 856–866, Jul. 2012.

[12] P. E. M. Purnick and R. Weiss, "The second wave of synthetic biology: from modules to systems," *Nature Reviews Molecular Cell Biology*, vol. 10, no. 6, pp. 410–422, Jun. 2009.

[13] R. Sarpeshkar, "Cytomorphic electronics: cell-inspired electronics for systems and synthetic biology," in *Ultra Low Power Bioelectronics: Fundamentals, Biomedical Applications, and Bio-Inspired Systems*. Cambridge, U.K.: Cambridge University Press, 2010, ch. 24. Also, Chapters 1, 2, 7, 14, 19, 22, 23, 26 provide useful background for this article. http://www.rle.mit.edu/acbs/wp-content/uploads/2014/03/ULPB_2010_bkprvw.pdf

[14] R. Danial, S. S. Woo, L. Turicchia, and R. Sarpeshkar, "Analog transistor models of bacterial genetic circuits," in *Proceedings of the 2011 IEEE Biological Circuits and Systems (BioCAS) Conference*, San Diego, CA, pp. 333–336, Nov. 2011.

[15] S. Mandal and R. Sarpeshkar, "Circuit models of stochastic genetic networks," in *IEEE Symposium on Biological Circuits and Systems (BioCAS)*, Beijing, China, pp. 109–112, Nov. 2009.

[16] S. Mandal and R. Sarpeshkar, "Log-domain circuit models of chemical reactions," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, Taipei, Taiwan, pp. 2697–2700, May 2009.

[17] S. Mandal and R. Sarpeshkar, "Electronic system for modeling chemical reactions and biochemical processes," U.S. Patent 8 285 523, Oct. 9, 2012.

[18] M. Tavakoli and R. Sarpeshkar, "A sinh resistor and its application to tanh linearization," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 2, pp. 536–543, Feb. 2005.

[19] R. Danial, J. R. Rubens, R. Sarpeshkar, and T. K. Lu, "Synthetic analog computation in living cells," *Nature*, vol. 497, no. 7451, pp. 619–623, May 2013.

[20] H. de Jong, "Modeling and simulation of genetic regulatory systems: a literature review," *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology*, vol. 9, no. 1, pp. 67–103, 2002.

[21] U. Moran, R. Phillips, and R. Milo, "Snapshot: Key numbers in biology," *Cell*, vol. 141, no. 7, pp. 1262–1262.e1, 2010.

[22] U. Alon, *An Introduction to Systems Biology: Design Principles of Biological Circuits*. Boca Raton, Florida: Chapman and Hall, 2007.

[23] N. J. Guido, X. Wang, D. Adalsteinsson, D. McMillen, J. Hasty, C. R. Cantor, T. C. Elston, and J. J. Collins, "A bottom-up approach to gene regulation," *Nature*, vol. 439, no. 7078, pp. 856–860, Feb. 2006.

[24] E. M. Ozbudak, M. Thattai, I. Kurtser, A. D. Grossman, and A. van Oudenaarden, "Regulation of noise in the expression of a single gene," *Nature Genetics*, vol. 31, no. 1, pp. 69–73, May 2002.

[25] R. Sarpeshkar, T. Delbruck, and C. Mead, "White noise in MOS Transistors and Resistors," *Circuits and Devices Magazine, IEEE*, vol. 9, no. 6, pp. 23–29, Nov. 1993.

[26] T. Delbruck and A. Van Schaik, "Bias current generators with wide dynamic range," in *Circuits and Systems, 2004. ISCAS '04. Proceedings of the 2004 International Symposium on*, vol. 1, pp. I–337–I–340, May 2004.

[27] J. Kim, T.-K. Jang, Y.-G. Yoon, and S. Cho, "Analysis and design of voltage-controlled oscillator based analog-to-digital converter," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 57, no. 1, pp. 18–30, Jan. 2010.

[28] M. B. Elowitz and S. Leibler, "A synthetic oscillatory network of transcriptional regulators," *Nature*, vol. 403, no. 6767, pp. 335–338, Jan. 2000.

[29] T. Ellis, X. Wang, and J. J. Collins, "Diversity-based, model-guided construction of synthetic gene networks with predicted functions," *Nature Biotechnology*, vol. 27, no. 5, pp. 465–471, May 2009.

[30] W. Mather, M. Bennett, J. Hasty, and L. Tsimring, "Delay-induced degrade-and-fire oscillations in small genetic circuits," *Phys. Rev. Lett.*, vol. 102, p. 068105, Feb. 2009.

[31] M. B. Elowitz, A. J. Levine, E. D. Siggia, and P. S. Swain, "Stochastic gene expression in a single cell," *Science*, vol. 297, no. 5584, pp. 1183–1186, 2002.

[32] R. Sarpeshkar, M. Baker, C. Salthouse, J.J. Sit, L. Turicchia, and S. Zhak, "An Analog Bionic Ear Processor with Zero-Crossing Detection," *Proceedings of the IEEE International Solid State Circuits Conference* (ISSCC), vol.1, pp. 78-79, Feb. 2005.

[33] Sarpeshkar, R., C. Salthouse, J.J. Sit, M. Baker, S. Zhak, T. Lu, L. Turicchia, and S. Balster, "An Ultra-Low-Power Programmable Analog Bionic Ear Processor," *IEEE Transactions on Biomedical Engineering*, Vol. 52, No. 4, pp. 711-727, April 2005.

[34] G.E.R. Cowan, R.C. Melville, and Y.P. Tsividis, "A VLSI Analog Computer/Digital Computer Accelerator", ", *IEEE Journal of Solid-State Circuits,* vol. 41, no. 1, pp. 42-53, Jan. 2006.

[35] D.E. Shaw, M.M. Deneroff, R.O. Dror, J.S. Kuskin, R.H. Larson, J.K. Salmon, C. Young et al. "Anton, a special-purpose machine for molecular dynamics simulation." *Communications of the ACM, vol.* 51, no. 7, pp. 91-97, Jul. 2008.

[36] J. Seo, B. Brezzo, Y. Liu, B.D. Parker, S.K. Esser, R.K. Montoye, B. Rajendran, J.A. Tierno, L. Chang, D.S. Modha, D.J. Friedman, "A 45nm CMOS neuromorphic chip with a scalable architecture for learning in networks of spiking neurons." IEEE *Custom Integrated Circuits Conference (CICC)*, pp. 1-4, Sep. 2011.

[37] D.T. Gillespie, "Approximate accelerated stochastic simulation of chemically reacting systems." *The Journal of Chemical Physics,* vol. 115, no. 4, pp. 1716-1733, Apr. 2001.

[38] Fokker, A. D. "Die mittlere Energie rotierender elektrischer Dipole im Strahlungsfeld." *Annalen der Physik,* vol. 348, no. 5, pp. 810-820, 1941.

[39] Planck, M. "Sitzungsber." *Preuss. Akad. Wiss. Phys. Math. Kl,* vol. 325, pp. 3, 1917.

[40] R. Sarpeshkar, "Analog versus digital: extrapolating from electronics to neurobiology." *Neural computation,* vol. 10, no. 7, pp. 1601-1638, Oct. 1998.

[41] B. Gilbert, "Translinear circuits: an historical overview." *Analog Integrated Circuits and Signal Processing* , vol. 9, no. 2, pp. 95-118, Mar 1996.

[42] E. Seevinck, "Companding current-mode integrator: A new circuit principle for continuous-time monolithic filters." *Electronics Letters*, vol. 26, no. 24, pp. 2046-2047, Nov 1990.

[43] Y. P. Tsividis, V. Gopinathan and L. Toth, "Companding in signal processing." *Electronics Letters*, vol. 26, no.17, pp. 1331–1332, Aug 1990.

[44] D. R. Frey, "State-space synthesis and analysis of log-domain filters." *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 45, no. 9, pp. 1205–1211, Sep 1998.

[45] C. Toumazou, F. J. Lidgey and D. Haigh, "Analogue IC design: The Current-Mode Approach." Presbyterian Publishing Corp, no. 2, 1990.

[46] A. G. Andreou, K. A. Boahen, P. O. Pouliquen, A. Pavasovic, R. E. Jenkins and K. Strohbehn, "Current-mode subthreshold MOS circuits for analog VLSI neural systems." *IEEE Transactions on Neural Networks*, vol. 2, no. 2, pp. 205–213, Mar 1991.

[47] E. Seevinck, E. A. Vittoz, M. du Plessi, T. H. Joubert and W. Beetge, "CMOS translinear circuits for minimum supply voltage." *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 47, no. 12, pp. 1560–1564, Dec 2000.

[48] B. A. Minch, "Synthesis of multiple-input translinear element log-domain filters." *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS),* Orlando, FL, vol. 2, pp. 697–700, Jul 1999.

[49] Y. Tsividis, N. Krishnapura, Y. Palaskas and L. Toth, "Internally varying analog circuits minimize power dissipation." *IEEE Circuits and Devices Magazine,* vol. 19, no. 1, pp. 63–72, Jan 2003.

[50] J. Lazzaro, S. Ryckebusch, M. A. Mahowald and C. A. Mead, "Winner-take-all networks of O(n) complexity." *Advances in Neural Information Processing Systems*, vol. 1, pp. 703–711, 1989

[51] R. H. R. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas and H. S. Seung, "Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit." *Nature*, vol. 405, pp. 947–951, Jun 2000.

[52] M. W. Baker and R. Sarpeshkar, "Low-Power Single-Loop and Dual-Loop AGCs for Bionic Ears." *IEEE Journal of Solid-State Circuits*, vol. 41, no.9, pp. 1983–1996, Sep 2006.

[53] K.I. Papadimitriou, G. B. V. Stan, and E. M. Drakakis. "Systematic computation of nonlinear cellular and molecular dynamics with low-power CytoMimetic circuits: a simulation study." *PloS one,* vol. 8, no. 2, Feb 2013.

[54] E. R. Timothy, and G. Cauwenberghs. "Synthesis of log-domain filters from first-order building blocks." *Research perspectives on dynamic translinear and log-domain circuits*. Springer US, vol. 550, pp. 71-80, 2000.

Sung Sik Woo is a graduate student at the Analog Circuits and Biological Systems group at MIT. His PhD thesis is on the design and implementation of cytomorphic chips for synthetic and systems biology. His bachelor's degree was in Electrical Engineering from Korea Advanced Institute of Science and Technology (KAIST).

Jaewook Kim is a Postdoctoral Research Fellow at the Analog Circuits and Biological Systems group at MIT. He works on analog circuits for electrochemistry and synthetic biology. His bachelor's, Masters, and PhD degrees were in Electrical Engineering from Korea Advanced Insitute of Science and Technology (KAIST).

Professor Rahul Sarpeshkar heads the Analog Circuits and Biological Systems group at MIT, where he is currently a tenured professor (http://www.rle.mit.edu/acbs/). His group designs advanced molecular circuits in living cells and on analog supercomputing chips. He obtained Bachelor's degrees in Electrical Engineering and Physics at MIT. After completing his PhD at CalTech, he joined Bell Labs as a physics member of technical staff in their department of biological computation. His text book *Ultra Low Power Bioelectronics: Fundamentals, Biomedical Applications, and Bio-inspired Systems* pioneered the concept of cytomorphic systems, and later led to the use of log-domain circuits in living cells and the launch of the field of analog synthetic biology. He has authored 127 publications, holds 35 awarded patents, and is a winner of several awards including the NSF Career, ONR Young Investigator, and Packard Fellow Awards.