Project Description
EAGER: *Synthesizing Signal Processing Functions with Biochemical Reactions*

1 Introduction

This proposal addresses the synthesis of computations and signal processing operations through protein-protein biochemical reactions. A typical signal processing operation produces an output signal by *filtering* or *transforming* an input signal: examples are smoothing a signal with a *moving-average* filter and performing a Fast Fourier Transform (FFT). The past few decades have seen remarkably progress in the design of integrated circuits for digital signal processing (DSP) in applications such as audio and video processing. We aim to apply and extend this expertise to the domain of bio-computation.

In digital signal processing hardware, the inputs and outputs are typically represented in binary form, say as *two's complement numbers*. If the physical quantity is sensed in analog form, then it is sampled and quantized by an *analog-to-digital* (A/D) converter. In this project, we will investigate signal processing in a novel context: protein-protein biochemistry. **Here the inputs and outputs are quantities of different types of proteins**.

Conceptually, the rules of biochemistry are straight-forward: each biochemical reaction is a primitive process that specifies how and at what rate different types of proteins combine to form other types of proteins. The complexity stems from the dynamics at play among the multitude of coupled reactions operating on the different protein types. All the biochemistry executes asynchronously and in parallel. Techniques for *analyzing* such processes are well established [27, 37, 38, 39, 40]. However, *synthesizing computation* with such mechanisms requires new techniques – and an entirely new mindset.

We have advocated a novel view for structuring computation: instead of transforming definite inputs into definite outputs – say, Boolean, integer, or real values into the same – the circuits and biological systems that we design transform probability values into probability values; so, conceptually, real-valued probabilities are both the inputs and the outputs [31, 65]. Note that the underlying behavior is discrete and inherently random. Nevertheless, when cast in terms of probabilities, the computation is robust: inputs and outputs are encoded through the statistical distribution of the signals.

This proposal is forward-looking and positioned in the realm of synthetic biology; it is focused on concepts for designing new functionality with realistic yet abstract mechanisms of biochemistry. The goal of this research is to demonstrate the feasibility of designing signal processing with biochemistry and to develop the tools and the methodology for design. We will demonstrate that biochemistry can implement simple and powerful signal processing operations such as finite impulse response (FIR) and infinite impulse response (IIR) filters, adaptive digital filters, Fourier transforms, and more complex operations such as adaptive decision feedback equalizers [62, 64]. As a proof concept, we present detailed designs for FIR and IIR filters in Sections 2.3.1 and 2.3.2.

It is important to be clear from the outset that the design of these filters and transforms will be formed in an abstract framework. At this time, our research will not attempt to address the experimental application of these ideas *in vitro* or *in vivo*. That is left to experimental biologists; it is beyond our expertise and the scope of this proposal. Nevertheless we remark that if our framework is proven feasible, it will open numerous opportunities in fields such as biochemical sensing and drug delivery.

Imagine a situation where a *decision feedback equalizer* is implemented entirely through biochemical reactions: the inputs and outputs are quantities of proteins; the result is a decision to deliver a drug or not, performed adaptively and autonomously. Or imagine a situation where the biochemistry performs *band-pass filtering* of a time-varying input signal: the quantity of output protein is a highly-tuned function of the frequency of the changes in the quantities of input proteins.

1.1 EAGER Funding

The research outlined in this proposal strives for new and transformative approaches to design in synthetic biology. A broad theme is the application of expertise from established fields, such as digital circuit design, to problems in this nascent field. Marc Riedel brings the requisite expertise in design automation and logic synthesis; Keshab Parhi in integrated circuit design for digital signal processing. The scope of the project is

conceptually new and exploratory. It straddles existing disciplines but does not fit within any existing NSF program.

1.2 Design Framework

This project will develop a modular and extensible design flow for implementing biochemical signal processing functions. Synthesis first will be performed at a conceptual level, in terms of abstract biochemical reactions – a task analogous to **technology-independent synthesis** in integrated circuit design. Then the results will be mapped onto specific biochemical components, selected from libraries – a task analogous to **technology mapping** in integrated circuit design.

The goal of our synthesis methodology is to produce a set of biochemical reactions that satisfies the specified I/O functionality. The input-output (I/O) specification for our methodology can be a functional specification expressing a relationship time-dependent relationship between input and output quantities of proteins; for example it could be a moving average, as shown in Figure 1. Alternatively, it could be a set of data points expressing this relationship as a function of time. In this case, we obtain the requisite functional relationship through standard *curve-fitting* techniques. The resulting set of reactions are the equivalent of the specification of a transistor netlist. Given such a netlist, established simulation methods and tools are used to characterize the chemical kinetics – a task analogous to the simulation of integrated circuits with SPICE [61]. The resulting *waveforms*, specifying quantities of proteins as a function of time, confirm the validity of the design.

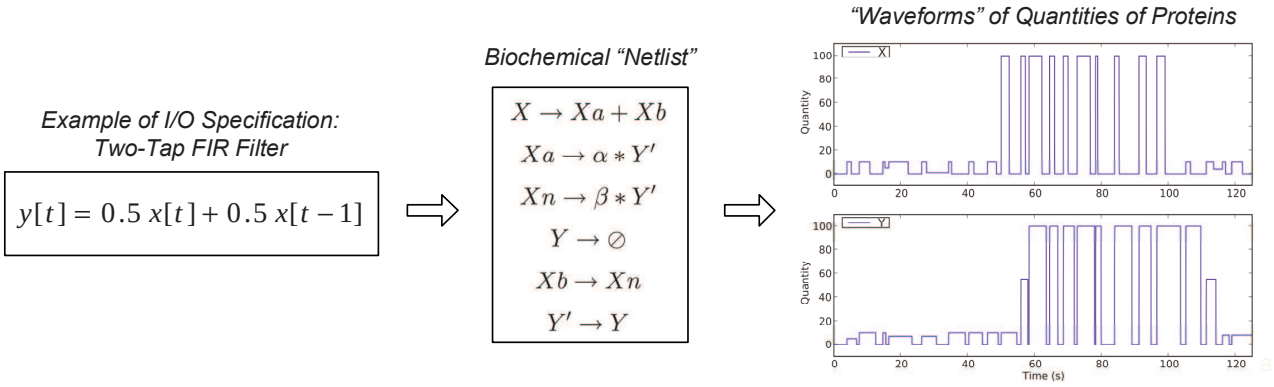

Figure 1: Our synthesis methodology begins with a functional specification, such as that of an FIR filter, or a set of data points. It produces a set of biochemical reactions – the equivalent of a circuit netlist. This netlist is simulated using standard techniques and tools; these produce detailed traces of protein quantities – the equivalent of voltage and current waveforms produced by a circuit simulation tool.

1.3 Context and Related Work

Increasingly, biology has become a computational science as modeling and simulation are applied alongside experimental work in the lab [27]. Further, as researchers are striving for new functionality through genetic manipulations, it is quickly becoming a full-fledged engineering discipline [28]. Recent accomplishments in this area portend of a coming revolution: *Salmonella* that secretes spider silk proteins [86]; yeast that degrades biomass into ethanol [76]; and *E. coli* that produces anti-malarial drugs [72].

The impetus to create synthetic biological systems is, in fact, broader. In both science and engineering, understanding is often achieved by constructing and testing simplified systems from the bottom up, teasing out and nailing down the fundamental principles in the process. Research teams are making significant progress progress toward the goal of artificial life: a living bacterial cell with fully synthetic DNA [36, 41]. In engineering terms, the objective is to assemble a machine (a synthetic bacterium) in which the functionality of all the parts (the genes, the proteins that they code for, and how these interact biochemically) are understood. If the machine works, this vindicates the scientific understanding; if it doesn't – and surely it won't at first – then new understanding can be achieved by examining where and how it breaks. Of

course, with a working blueprint for a synthetic machine, new functionality can be engineered robustly and effectively.

The set of constitutive parts that can be used for genetic manipulation in synthetic systems is vast. Comprehensive repositories of genetic data have been assembled – some public, some commercial – cataloging genes, their DNA sequences, and their products [8]. A concerted effort has been made to assemble repositories of standardized and interoperable parts for synthetic applications [7]. The platforms used will depend on the application, but the technology for synthesizing DNA is becoming routine: in 2008, firms started offering custom-gene synthesis through e-commerce websites; the going rate is $0.49 per base pair [33].

So, in a real sense, the **hardware** for synthetic biology exists, i.e., the technology and infrastructure for obtaining cells with custom-designed genes. The **instruction set** is, to a large extent, known, i.e., genes and their function, cataloged in libraries. The challenge is: *how can we write code with these instructions on this type of hardware*?

One of the great successes of integrated circuit design has been in *abstracting* and *scaling* the design problem. The physical behavior of transistors is understood in terms of differential equations operating on voltage and current values in semiconductor materials. However, the design of circuits proceeds at a more abstract level – in terms of switches, gates, and functional units. This modular approach makes design tractable; furthermore, it permits a systematic exploration of different configurations, leading to optimal designs.

Although driven by experimental expertise, synthetic biology has reached a stage where it calls for a similar automated design flow. This would allow for virtual experimentation: one could vary the inputs and parameters of synthetic designs and observe the outputs – in a manner analogous to traditional *in vitro* and *in vivo* experimentation. (This has been dubbed, somewhat facetiously, as "*in silico*" experimentation [6].) Beyond analysis, by deliberately applying design methodologies, one could engineer **computational control** over biological processes, designing pathways that produce specific outputs in response to different combinations of inputs.

Indeed, there has been considerable research directed at this question of *computation* with biological mechanisms. We do not attempt to catalog all the ideas that have been proposed in this vein; we will only refer to a slice of some of the research that is relevant to this proposal. Thomas Knight and his colleagues first suggested the idea of "*in vivo*" digital circuits in the the 1990's [82, 83]. Much of the research that followed has been directed at manipulating the mechanisms of gene regulation [4, 5, 26, 48]. Considerable mathematical expertise, particularly from the realm of control and dynamical systems, and been developed for and applied to biological systems [24, 25]. A topic of interest to the mathematical and modeling community is *noise* and *randomness* in biochemical systems [37, 31, 56, 57, 59, 66, 65, 75].

DNA and RNA-based computation have been explored theoretically and demonstrated experimentally [1, 6, 85]. Oscillatory mechanisms, suitable for the sort of clocking used in our designs, have been demonstrated experimentally [26]. Samoilov, Arkin and Ross established an analytic framework for the dynamics of biological systems in terms of the signal processing functions that they perform [74].

The NSF has recently awarded Erik Winfree and his colleagues at Caltech the University of Washington an "Expeditions in Computing" award for research on Molecular Programming. The project brings together experimental expertise in DNA computing, together with theoretical strengths and mathematical strengths in control and information theory. (Absent from the team, it should be noted, is expertise in signal processing and integrated circuit design.) Soloveichik, Cook, Winfree and Bruck discuss theoretical aspects of molecular computation [78]. The concepts of register-based computation and clocking that we use are due to [23].

While previous work has established analytic frameworks, this proposal is the first research to tackle the *synthesis* of signal processing functions with biochemical reactions. Previous work has assumed continuous-time processing with mechanisms like negative feedback control. We propose a constructive approach based on *discrete-time processing*. This brings into the scope of the research the vast body of knowledge and experience in circuit design for digital signal processing. Further, whereas prior research has focused on gene regulation as the mechanism for computation, we propose **computation performed entirely with protein-protein interactions**. In spite of the fact that a multitude of such reactions happen asynchronously and in parallel in such system, our method of *clock* and *key generation* enables us to implement robust computation [31].

2 Technical Approach

The project will assemble a flexible toolkit of functional modules: these include signal processing operations such as finite impulse response (FIR) and infinite impulse response (IIR) filters, adaptive digital filters, Fourier transforms, and more complex operations such as adaptive decision feedback equalizers [62, 64].

2.1 Discrete Biochemistry

Interesting biochemistry typically involves complex molecules such as proteins and enzymes. Within the confines of a cell, the *quantities* of such molecules are often surprisingly small: on the order of tens, hundreds, or thousands of molecules of each type. At this scale, individual reactions matter, and the problem must be analyzed discretely [37].

Consider a system with three types of molecules x_1, x_2, and x_3. The *state* of the system is described by the number of molecules: $[|x_1|, |x_2|, |x_3|]$, For instance, the system might be in the state $[3, 3, 3]$ with three molecules of each type. Consider the three reactions:

$$\begin{aligned} R_1: &\quad x_1 + x_2 \rightarrow 3x_3 \\ R_2: &\quad x_1 + 2x_3 \rightarrow 3x_2 \\ R_3: &\quad 2x_2 + x_3 \rightarrow 2x_1. \end{aligned}$$

Note that these reactions are *coupled*: the types appear both as reactants and as products in different reactions. Suppose that the system is in the state $[3, 3, 3]$ and reaction R_1 fires. One molecule of type x_1 and one of type x_2 are consumed; three of type x_3 are produced. This results in the state transition:

$$[3, 3, 3] \xrightarrow{R_1} [2, 2, 6].$$

As reactions fire, a cellular process follows a sequence of such transitions. Figure 2 illustrates the trajectory taken from the state $[3, 3, 3]$ by the sequence R_1, R_2, and R_3.

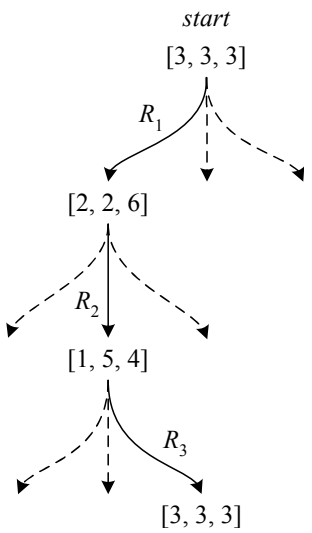

Figure 2: **Biochemical reactions as discrete events.** Beginning from the state $[3, 3, 3]$, R_1 fires, followed by R_2, followed by R_3.

2.2 Probabilistic Biochemistry

Randomness is inherent: at each instant, the exact sequence of reactions that fires next is a matter of chance. Indeed, ignoring environmental changes outside the cell, one can assume cellular biochemistry behaves as a *Markov process*: the probability of future events depends only on the present state of the cell. At each point in time, the probability of a given reaction firing is a function of the quantities of different types of molecules present. Specifically, it is proportional to: 1) the number of ways that the reactants can come together; and 2) and the reaction *rate*. Suppose that the system in the example above is in the state $S = [3, 4, 5]$. There are

$$3 \times 4 = 12, \quad 3 \times \binom{5}{2} = 30, \quad \binom{4}{2} \times 5 = 30$$

ways to choose the reactants of R_1, R_2, and R_3, respectively. Suppose that the rates of reactions R_1, R_2, and R_3 are 1, 2 and 3, respectively. Then the firing probabilities for R_1, R_2, and R_3 are

$$\frac{12 \times 1}{162} = 0.074, \quad \frac{30 \times 2}{162} = 0.370, \quad \frac{30 \times 3}{162} = 0.556, \quad \text{respectively.}$$

Although we will often refer to rates in relative and qualitative terms – e.g., "fast" vs. "slow" – these are, in fact, real-valued parameters that are either deduced from biochemical principles or measured experimentally [49].

Computationally, such discrete probabilistic biochemical systems are characterized through Monte Carlo simulation [34], [35], [37], [58]. Beginning from an initial state, reactions are chosen at random, based on propensity calculations. As reactions fire, the quantities of the different species change by integer amounts. Repeated trials are performed and the probability distribution of different outcomes is estimated by averaging the results.

In biological systems, signaling pathways produce specific output types of proteins in response to input types. The exact sequence and timing of biochemical reactions that fire is random. However, the probability distribution on specific outcomes – for instance, the mutually exclusive production of different signaling molecules – is precise and robust. This view has strong parallels with an efficient new methodology that we have developed for synthesizing nanoscale circuits with probabilistic behavior. Our approach produces circuits that are highly resistant to errors – both in the underlying components and in the signal values. If noise-related faults produce random bit flips, these result in fluctuations in the statistics; accuracy is regained through increased redundancy. Thus, the approach provides *tolerance of faults* that scales gracefully to large numbers of errors [65]. In synthetic biology, our approach produces a precise distribution of different outcomes across a population of organisms or in a sequence of trials. This gives us the ability to fine-tune the response – akin to hedging with a portfolio of investments – in spite of large uncertainties in the underlying mechanisms [31].

2.3 Signal Processing

We have developed modules for computing a variety of functions: **multiplication**, **exponentiation**, **logarithms**, etc. [31]. With our linear and raising-to-a-power modules, our scheme can be used to implement *arbitrary* polynomial functions; hence, in principle, it could be used to approximate complex functions through Taylor series expansions.

In this project, we will demonstrate that biochemistry can implement simple and powerful signal processing operations such as finite impulse response (FIR) and infinite impulse response (IIR) filters. In the project, we will produce a general methodology for the design more complicated signal processing functions such as adaptive digital filters, Fourier transforms, and more and adaptive decision feedback equalizers.

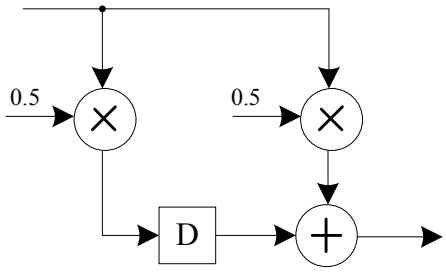

Figure 3: A Two-Tap FIR Filter: it computes a moving average of the input signal. D is a delay element.

2.3.1 A Finite Impulse Response (FIR) Filter Design

One of the basic filters in digital signal processing is a *two-term moving average filter*, shown in Figure 3. It implements the following function:

$$Y[n] = \alpha X[n] + \beta X[n-1].$$

with tap coefficients $\alpha = \beta = 0.5$. Here $x[n]$ is assumed to be an independent input signal that does not depend on previous values of $x[n]$. Similarly, $y[n]$ is a discrete-time output signal that does not depend on previous values of $y[n]$.

First, we will explain the design of the filter *without* rate considerations. The set of reactions is shown in Figure 4. We argue that these reactions "sample" the current value of X and implement the following computation: firstly types X_a and X_b are both set to the current value X; X_a is used in computing $\alpha x[n]$ at the current time and X_b is used in computing $\beta x[n-1]$ at the *next* time; Y' is is set to a linear combination of these; the previous value of Y is cleared; and finally Y set to Y'. If everything were ordered this way, these reactions would achieve the requisite computation: $Y[n] = \alpha X[n] + \beta X[n-1]$.

The challenge in computing with biochemistry, of course, is to enforce such an order of the computation. All these reaction are executing in parallel and asynchronously. We apply the concepts of *module locking* and *clocking* [32] We create a set of reactions that forms a loop; new types of proteins playing the role of "keys" are introduced for unlocking phases of the computation according to the progress through the loop.

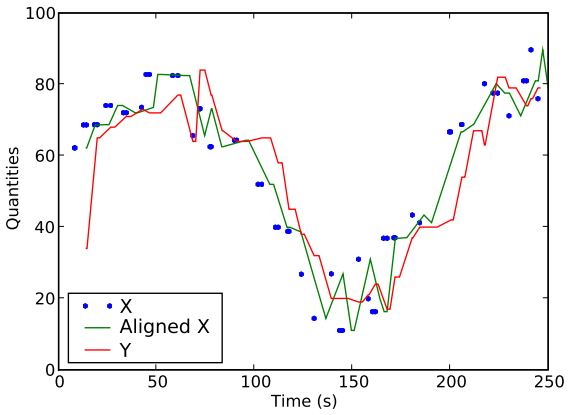

Figure 5: Input and output waveforms for a biochemical FIR filter. The input quantity X is shown both as points from the simulation and as the ideal curve. The output Y is shown in red. The green curve shows where the value of X is after the average delay of the system in order to be time-aligned with the red output curve.

The keys are created by "keysmiths". The six reactions shown in Figure 6 will implement the clock loop. Each "pulse width" of the clock will have identical width if all of the clocking reactions occur at the same rate. (Changing the rates of the reactions only changes the duty cycle of the clock.) These reactions ensure that once a key is present, it cannot be overwritten by a reaction other than one of the clocking reactions.

The full design of the FIR filter in Figure 3 with module locking is shown in Figure 7. The assumption that we make about the rates of these reactions – and the only such assumption – is that they must all be faster that the clock computations. Provided that this holds the relative rates do not matter.

We simulate and validate our designs with transient stochastic simulation [22]. The simulation results for the biochemical FIR filter illustrates the functionality of the design: the moving average smooths high-frequency noise. These results are shown in Figure 5. Here, the input X is shown in green; it is a noisy sinusoid. The output Y is shown in red; note that it is a clearer sinusoidal waveform.

$$\begin{array}{rcc} X & \rightarrow & X_a + X_b \\ \alpha X_a & \rightarrow & Y' \\ \beta X_n & \rightarrow & Y' \\ Y & \rightarrow & \oslash \\ X_b & \rightarrow & X_n \\ Y' & \rightarrow & Y \end{array}$$

Figure 4: Design of the FIR Filter in Figure 3 without rate considerations. Here X_a, X_b and Y' play the role of "temporary variables"; α and β are stoichiometric coefficients – for the moving average $\alpha = \beta = 2$.

2.3.2 Infinite Input Response Filter (IIR)

Expanding on the design of the FIR filter, we present the full design of a Biquad IIR Filter [62, 64], shown in Figure 8. This is a flexible construct that can be used to implement filtering operations. It implements the function:

$$y[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] - a_1 y[n-1] - a_2 y[n-2]$$

For this example, the coefficients were selected for *low pass filtering with some overshoot*, as this demonstrates fluctuations on the output:

$$b_0 = \frac{1}{3}, b_1 = \frac{2}{3}, b_2 = \frac{1}{3}, a_0 = 0, a_1 = \frac{1}{3} \quad (1)$$

We omit the details of the clock and key generation mechanisms; these are a straight-forward generalization of those presented for the FIR Filter in Section 2.3.1. The biochemical design of the Biquad IIR Filer is given in Figure 9. Fractional values for the coefficients are implemented by using reaction (2) for positive and reaction (3) for negative when the coefficient is of the form $\frac{A}{B}$.

$$BX \rightarrow AY \quad (2)$$

$$BX + AY \to \oslash \tag{3}$$

A stochastic transient analysis of the Biquad IIR was performed, with an input square-waveform X with a 200 second period, a low amplitude 50 molecules, and a high amplitude of 100 molecules. The resulting waveforms for the input X and the output Y are shown in Figure 11. To verify the filter's operation, the same filtering function was run numerically to obtain the plot in Figure 10. This figure shows that the output rises slowly compared to the input, but overshoots the input, then rings twice and settles.

$$\begin{aligned}
\text{key}_3 &\xrightarrow{\text{slow}} \text{keysmith}_1 \\
\text{keysmith}_1 &\xrightarrow{\text{slow}} \text{key}_1 \\
\text{key}_1 &\xrightarrow{\text{slow}} \text{keysmith}_2 \\
\text{keysmith}_2 &\xrightarrow{\text{slow}} \text{key}_2 \\
\text{key}_2 &\xrightarrow{\text{slow}} \text{keysmith}_3 \\
\text{keysmith}_3 &\xrightarrow{\text{slow}} \text{key}_3
\end{aligned}$$

Figure 6: The clock design for the FIR Filter.

$$\begin{aligned}
\text{key}_1 + X &\xrightarrow{\text{fast}} X_a + X_b + \text{key}_1 \\
\hline
\text{key}_2 + \alpha X_a &\xrightarrow{\text{fast}} Y' + \text{key}_2 \\
\text{key}_2 + \beta X_n &\xrightarrow{\text{fast}} Y' + \text{key}_2 \\
\text{key}_2 + Y &\xrightarrow{\text{fast}} \text{key}_2 \\
\hline
\text{key}_3 + X_b &\xrightarrow{\text{fast}} X_n + \text{key}_3 \\
\text{key}_3 + Y' &\xrightarrow{\text{fast}} Y + \text{key}_3
\end{aligned}$$

Figure 7: The full locked version design of the FIR filter in Figure 3.

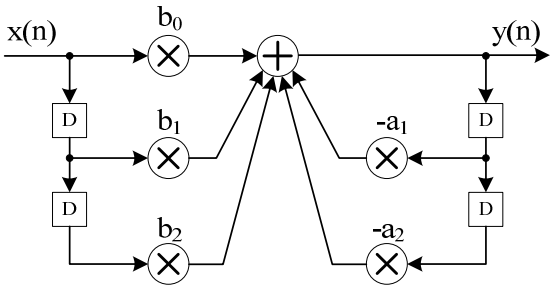

Figure 8: A Biquad IIR Filter.

$$\begin{aligned}
\text{key}_1 + X &\xrightarrow{\text{fast}} Xa + Xb + \text{key}_1 \\
\text{key}_1 + Xn_1 &\xrightarrow{\text{fast}} Xn_1' + Xn_2' + \text{key}_1 \\
\text{key}_1 + Yn_1 &\xrightarrow{\text{fast}} Yn_1' + Yn_2' + \text{key}_1 \\
\hline
\text{key}_2 + Xa &\xrightarrow{\text{fast}} B_0 * Y' + \text{key}_2 \\
\text{key}_2 + Xn_1' &\xrightarrow{\text{fast}} B_1 * Y' + \text{key}_2 \\
\text{key}_2 + Xn_2 &\xrightarrow{\text{fast}} B_2 * Y' + \text{key}_2 \\
\text{key}_2 + Y &\xrightarrow{\text{fast}} \text{key}_2 \\
\hline
\text{key}_3 + A_1 Y' + Yn_1' &\xrightarrow{\text{fast}} \text{key}_3 \\
\text{key}_3 + A_2 Y' + Yn_2 &\xrightarrow{\text{fast}} \text{key}_3 \\
\hline
\text{key}_4 + Xb &\xrightarrow{\text{fast}} Xn_1 + \text{key}_4 \\
\text{key}_4 + Xn_2' &\xrightarrow{\text{fast}} Xn_2 + \text{key}_4 \\
\text{key}_4 + Yn_2' &\xrightarrow{\text{fast}} Yn_2 + \text{key}_4 \\
\text{key}_4 + Y' &\xrightarrow{\text{fast}} Y + Yn_1 + \text{key}_4
\end{aligned}$$

Figure 9: Biochemical Design of the IIR Filter.

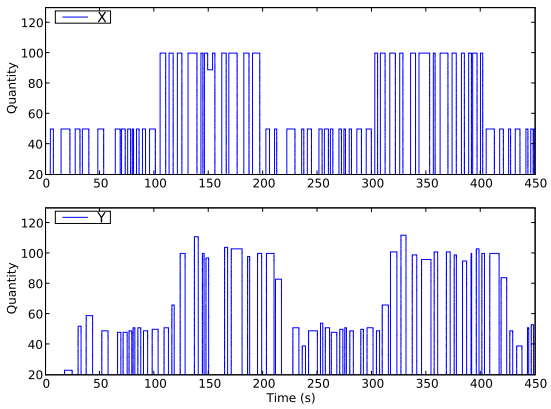

Figure 10: Input vs. Output waveforms for IIR Filter.

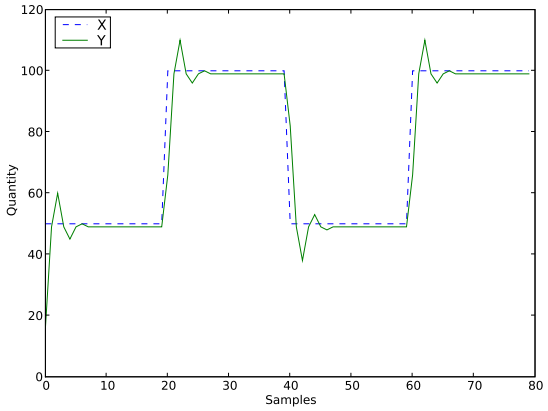

Figure 11: Detailed transient analysis of IIR Filter.

3 Prior NSF Support

- Marc Riedel has been awarded the NSF CAREER award (0845650) *Computing with Things Small, Wet and Random: Design Automation for Digital Computation with Nanoscale Technologies and Biological Processes*, starting July 1, 2009.
- Keshab Parhi was awarded a two-year grant (0811456) *Collaborative Research: CPA-DA: Noise-Aware VLSI Signal Processing: A New Paradigm for Signal Processing Integrated Circuit Design in Nanoscale Era*, starting Sept. 1, 2008.
- We summarize the findings of Keshab Parhi's most recent past NSF grant-0429979 which ended on 8/31/2007: This funding allowed his group to develop pipelined and parallel architectures for Tomlinson-Harashima Precoders, in reducing the complexity of parallel decision feedback equalizers, in reducing complexity of echo and near-end cross-talk cancelers in multi-gigabit ethernet systems, and in developing novel low-power architectures for these cancelers. Two patent applications were filed by the University of Minnesota for the research carried out. Other high-speed building blocks for parallel filters, parallel adaptive filters, and DCTs were published. Various papers supported by this grant are referenced in [11]-[21][41]-[46].

References

[1] L. Adleman, "Molecular Computation Of Solutions To Combinatorial Problems," *Science*, Vol. 266, No. 11, pp. 1021–1024, year 2004.

[2] J. Anderson, E. Clarke, A. Arkin, and C. Voigt, "Environmentally Controlled Invasion of Cancer Cells by Engineered Bacteria," *Journal of Molecular Biology*, Vol. 355, No. 4, pp. 619–627, 2006.

[3] A. Arkin, J. Ross, and H. McAdams, "Stochastic Kinetic Analysis of Developmental Pathway Bifurcation in Phage λ-Infected *E. coli* Cells," *Genetics*, Vol. 149, No. 1633, 1998.

[4] S. Basu, R. Mehreja, S. Thiberge, M. Chen, and R. Weiss, "Spatiotemporal Control of Gene Expression with Pulse-generating Networks," *Proceedings National Academy of Science*, Vol. 101, pp. 6355-6360, 2004.

[5] S. Basu, Y. Gerchman, C. Collins, F. Arnold, and R. Weiss, "A Synthetic Multicellular System for Programmed Pattern Formation," *Nature*, Vol. 434, pp. 1130-1134, 2005.

[6] Y. Benenson, B. Gil, U. Ben-Dor, R. Adar, and E. Shapiro, "An Autonomous Molecular Computer for Logical Control of Gene Expression," *Nature*, Vol. 429, pp. 423–429, 2004.

[7] MIT Registry of Standard Biological Parts.

[8] The BioCyc Collection of Pathway/Genome Databases.

[9] A. Brower, R. Ewing, R. Brower, and H. Abdel-Aty-Zohdy, "SPICE for Modeling Biochemical Networks," *Midwest Symposium on Circuits and Systems*, pp. 491–494, 2005.

[10] Y. Cao, S. Lampoundi, H. Li, and L. Petzold, "A Software Package for Discrete Stochastic Simulation of Chemically Reacting Systems," *SIAM Conference on Computational Science and Engineering*, 2005.

[11] J. Chen, Y. Gu and K. Parhi, "MIMO Equalization and Cancellation for 10Gbase-T", *Proc. of 2006 IEEE Int. Conf. Acoustics, Speech and Signal Processing*, pp. IV–637–IV–640, 2006.

[12] J. Chen and K. Parhi, "Adaptive Tap Management in Multi-Gigabit Echo and NEXT Cancellers", *Proc. 2006 IEEE Workshop on Signal Processing Systems,* pp. 415–419, 2006.

[13] C. Cheng and K. Parhi, "Hardware Efficient Fast Computation of the Discrete Fourier Transform", *Journal of VLSI Signal Processing*, pp. 159–172, Vol. 42, No. 2, 2006.

[14] C. Cheng and K. Parhi, "High-Speed Parallel CRC Implementation Based on Unfolding, Pipelining and Retiming," *IEEE Trans. Circuits and Systems-II: Express Briefs*, Vol. 53, No. 10, pp. 1017–1021, 2006.

[15] C. Cheng and K. Parhi, "Hardware Efficient Fast DCT Based on Novel Cyclic Convolution Structures", *IEEE Trans. on Signal Processing*, Vol. 54, No. 11, pp. 4419–4434, 2006.

[16] J. Chen and K. Parhi, "Fast Computation of MIMO Equalizers and Cancellers in 10GBASE-T Channels", *IEEE Int. Conf. Acoustics, Speech and Signal Processing,* pp. III–201–III–204, 2007.

[17] C. Cheng and K. Parhi, "Low Cost Parallel FIR Filter Structures with 2-Stage Parallelism", *IEEE Trans. Circuits and Systems-I: Regular Papers*, Vol. 54, No. 2, pp. 280–290, 2007.

[18] C. Cheng and K. Parhi, "Low-Cost Fast VLSI Algorithm for Discrete Fourier Transform", *IEEE Trans. Circuits and Systems-I: Regular Papers*, Vol. 54, No. 4, pp. 7910806, 2007,

[19] J. Chen and K. Parhi, "Further Cost Reduction of Adaptive and NEXT Cancellers for High-Speed Ethernet Transceivers", *Proc. of 2008 IEEE Workshop on Signal Processing Systems,* 2008.

[20] J. Chen, Y. Gu and K. Parhi, "Low Complexity and Low-Power ECHO And NEXT Cancellers for High-Speed Ethernet Transceivers", *IEEE Trans. Circuits and Systems-I: Regular Papers*, to appear.

[21] J. Chen, Y. Gu, and K. Parhi, "Novel FEXT Cancellation and Equalization for High Speed Ethernet Transmission," *IEEE Trans. Circuits and Systems-I: Regular Papers*, to appear.

[22] B. Chen and M. Riedel, "Stochastic Transient Analysis of Biochemical Reactions," *Pacific Biocomputing Symposium*, 2009.

[23] , M. Cook, D. Soloveichik, E. Winfree, and J. Bruck, "Programmability of Chemical Reaction Networks," *Technical Report #90*, Caltech Paradise Lab, 208.

[24] H. El-Samad, H. Kurata, J. Doyle, C. Gross, M. Khammash, "Surviving Heat Shock: Control Strategies for Robustness and Performance," *Proceedings of the National Academy of Sciences*, Vol. 102, pp. 2736–2741, 2005.

[25] H. El-Samad, S. Prajna, A. Papachristodoulou, J. Doyle, M. Khammash, "Advanced Methods and Algorithms for Biological Networks Analysis," *Proceedings of the IEEE*, Vol. 94, pp. 832–853, 2006.

[26] M. Elowitz and S. Leibler, "A Synthetic Oscillatory Network of Transcriptional Regulators," *Nature*, vol. 403, pp. 335-338, 2000.

[27] D. Endy and R. Brent,"Modelling Cellular Behaviour," *Nature*, Vol. 409, pp. 391–395, 2001.

[28] D. Endy, "Foundations for Engineering Biology," *Nature*, Vol. 438, pp. 449–453, 2005.

[29] D. Ferber, "Synthetic Biology: Microbes Made to Order," *Science*, Vol. 303. No. 5655, pp. 158–161, 2004.

[30] B. Fett and M. Riedel, "Characterizing the Decisions of Biochemical Systems with Probability Gradients," *International Conference on Systems Biology*, Dec. 2006.

[31] B. Fett, J. Bruck, and M. Riedel, "Synthesizing Stochasticity in Biochemical Systems," *Design Automation Conference*, Vol. 44, pp. 640–645, 2007.

[32] B. Fett and M. Riedel, "Module Locking in Biochemical Synthesis," *International Conference on Computer-Aided Design*, 2008.

[33] Firms performing DNA synthesis commercially: Invitrogen, Genscript, and Mr Gene.

[34] M. Gibson and J. Bruck, "Efficient Exact Stochastic Simulation of Chemical Systems with Many Species and Many Channels," *Journal of Physical Chemistry A*, No. 104, pp. 1876–1889, 2000.

[35] M. Gibson, "Computational Methods for Stochastic Biological Systems," *Ph.D. Dissertation*, California Institute of Technology, 2001.

[36] D. Gibson, G. Benders, C. Andrews-Pfannkoch, E. Denisova, H. Baden-Tillson, J. Zaveri, T. Stockwell, A. Brownley, D. Thomas, M. Algire, C. Merryman, L. Young, V. Noskov, J. Glass, J. Venter, C. Hutchison, and H. Smith, "Complete Chemical Synthesis, Assembly, and Cloning of a *Mycoplasma genitalium* Genome," *Science*, Vol. 319, No. 5867, pp. 1215-1220, 2008.

[37] D. Gillespie, "Exact Stochastic Simulation of Coupled Chemical Reactions," *Journal of Physical Chemistry*, Vol. 81, No. 25, pp. 2340–2361, 1977.

[38] D. Gillespie, "Approximate Accelerated Stochastic Simulation of Chemically Reacting Systems," *Journal of Chemical Physics*, Vol. 115, No. 4, 2001.

[39] D. Gillespie and L. Petzold, "Improved Leap-Size Selection for Accelerated Stochastic Simulation," *Journal of Chemical Physics*, Vol. 119, No. 16, 2003.

[40] D. Gillespie, "Stochastic Simulation of Chemical Kinetics," *Annual Review of Physical Chemistry*, Vol. 58, pp. 35-55, May 2007.

[41] J. Glass, N. Assad-Garcia, N. Alperovich, S. Yooseph, M. Lewis, M. Maruf, C. Hutchison, H. Smith, and J. Craig Venter, "Essential Genes of a Minimal Bacterium," *Proceedings of the National Academy of Sciences*, Vol. 103, No. 2, pp. 425–430, 2006.

[42] Y. Gu and K. Parhi, "Complexity Reduction of the Decoders for Interleaved Trellis Coded Modulation Schemes for 10 Gigabit Ethernet over Copper," *Proc. of 2004 IEEE Workshop on Signal Processing Systems*, pp. 130–135, 2004.

[43] Y. Gu and K. Parhi, "Pipelining Tomlinson-Harashima Precoders", *Proc. of 2005 IEEE Int. Symposium on Circuits and Systems*, pp. 408–411, 2005.

[44] Y. Gu and K. Parhi, "Interleaved Trellis Coded Modulation and Decoder Optimizations for 10 Gigabit Ethernet over Copper," *Journal of VLSI Signal Processing Systems*, Vol. 42, No. 3, pp. 211–221, 2006.

[45] Y. Gu and K. Parhi, "Pipelined Parallel Decision Feedback Decoders (PDFDs) for High-Speed Ethernet over Copper," *IEEE Trans. on Signal Processing*, Vol. 55, No. 2, pp. 707–715, 2007.

[46] Y. Gu and K. Parhi, "High-Speed Architecture Design of Tomlinson-Harashima Precoders," *IEEE Trans. Circuits and Systems-I: Regular Papers*, Vol. 54, No. 9, pp. 1929–1937, 2007.

[47] Y. Gu and K. Parhi, "Design of Parallel Tomlinson-Harashima Precoders", *IEEE Trans. Circuits and Systems-II: Transactions Briefs*, Vol. 55, No. 5, pp. 447–451, 2008.

[48] N. Guido, X. Wang, D. Adalsteinsson, D. McMillen, J. Hasty, C. antor, T. Elston and J. Collins, "A Bottom-up Approach to Gene Regulation," *Nature*, Vol. 439, pp. 856–860, 2006.

[49] R. Heinrich and S. Shuster, "The Regulation of Cellular Systems," *Chapman and Hall*, 1996.

[50] A. Hernday, B. Braaten, and D. Low,"The Intricate Workings of a Bacterial Epigenetic Switch," *Advances in Experimental Medicine & Biology*, Vol. 547, No. 83-9, 2004.

[51] I. Herskowitz, "Life Cycle of the Budding Yeast *Saccharomyces cerevisiae*," *Microbiological Reviews*, Vol. 52, No. 4, pp. 536–553, 1988.

[52] S. Kauffman, C Peterson, B. Samuelsson, and C. Troein, "Random Boolean Network Models and the Yeast Transcriptional Network," *Proceedings of the National Academy of Sciences*, Vol. 100, Dec. 2003.

[53] H. Kuwahara, C. Myers, N. Barker, M. Samoilov, and A. Arkin, "Automated Abstraction Methodology for Genetic Regulatory Networks," to appear in *Transactions on Computational Systems Biology*.

[54] A. Levskaya, A. Chevalier, J. Tabor, Z. Simpson, L. Lavery, M. Levy, E. Davidson, A. Scouras, A. Ellington, E. Marcotte, and C. Voigt, "Synthetic Biology Engineering Escherichia Coli to see Light," *Nature*, Vol. 438, pp. 441-442, 2005.

[55] L. Li and H. Lu, "Explore Biological Pathways from Noisy Array Data by Directed Acyclic Boolean Networks," *Journal of Computational Biology*, Vol. 12, No. 2, 2005.

[56] E. Libby, T. Perkins, and P. Swain, "Noisy information processing through transcriptional regulation," *Proceedings of the National Academy of Sciences*, Vol. 104, No. 17, pp. 7151–7156, 2007.

[57] L. Lok, "The Need for Speed in Stochastic Simulation," *Nature Biotechnology*, Vol. 22, No. 8, 2004.

[58] L. Lok and R. Brent, "Automatic Generation of Cellular Reaction Networks with Moleculizer 1.0," *Nature Biotechnology*, Vol. 23, 2005.

[59] H. McAdams and A. Arkin, "Stochastic Mechanisms in Gene Expression," *Proceedings of the National Academy of Sciences*, Vol. 94, No. 3, 1997.

[60] C. Myers, "*Asynchronous Circuit Design*," John Wiley and Sons, 2001.

[61] L. Nagel and D. Pederson, "Simulation Program with Integrated Circuit Emphasis," *Midwest Symposium on Circuit Theory*, 1973.

[62] A. Oppenheim and R. Schafer, "Discrete-Time Signal Processing," *Prentice Hall*, 1989.

[63] S. Paliwal, P. Iglesias, K. Hilioti, A. Groisman, and A. Levchenko, "MAPK-mediated bimodal Gene Expression and Adaptive Gradient Sensing in Yeast," *Nature*, Vol. 446, Issue 7131, pp. 46–51, 2007.

[64] K. Parhi, "VLSI Digital Signal Processing Systems," Wiley, 1999.

[65] W. Qian and M. Riedel, "The Synthesis of Robust Polynomial Arithmetic with Stochastic Logic," *Design Automation Conference*, pp. 648–653, 2008.

[66] C. Rao, D. Wolf, and A. Arkin, "Control, Exploitation and Tolerance of Intracellular Noise," *Nature*, Vol. 420, No. 6912, pp. 231–237, 2002.

[67] C. Rao and A. Arkin, "Stochastic Chemical Kinetics and the Quasi-Steady-State Assumption: Application to the Gillespie Algorithm," *Journal of Chemical Physics*, Vol. 118, No. 11, pp. 4999–5010, 2003.

[68] M. Rathinam, L. Petzold, Y. Cao, and D. Gillespie, "Stiffness in Stochastic Chemically Reacting Systems: The Implicit Tau-leaping Method," *Journal of Chemical Physics*, Vol. 119, No. 24, pp. 12784–12794, 2003.

[69] M. Riedel and J. Bruck, "Exact Stochastic Simulation with Event Leaping," *International Conference on Systems Biology*, Oct. 2005.

[70] M. Riedel, "Synthesizing Stochasticity in Biochemical Systems," *Synthetic Biology 3.0*, 2007.

[71] M. Riedel, "The Computer-Aided Synthesis of Stochasticity in Biochemical Systems," *Advances in Synthetic Biology*, 2008.

[72] D. Ro, E. Paradise, M. Ouellet, K. Fisher, K. Newman, J. Ndungu, K. Ho, R. Eachus, T. Ham, M. Chang, S. Withers, Y. Shiba, R. Sarpong, and J. Keasling, "Production of the Antimalarial Drug Precursor Artemisinic Acid in Engineered Yeast," *Nature*, Vol. 440, pp. 940–943, 2006.

[73] L. Salwinski and D. Eisenberg, "In Silico Simulation of Biological Network Dynamics," *Nature Biotechnology*, Vol. 22, No. 8, 2004.

[74] M. Samoilov, A. Arkin, and J. Ross, "Signal Processing by Simple Chemical Systems,"Journal of Physical Chemistry, Vol. 106, pp. 10205–10221, 2002.

[75] M. Schwehm, "Parallel Stochastic Simulation of Whole-Cell Models," *International Conference on Systems Biology*, 2001.

[76] M. Sedlak and N. Ho, "Production of Ethanol from Cellulosic Biomass Hydrolysate Using Genetically Engineered Yeast," *Applied Biochemistry & Biotechnology,* Vol. 114, No. 1–3, pp. 403–416, 2004.

[77] I. Shmulevich and S. Kauffman, "Activities and Sensitivities in Boolean Network Models," *Physical Review Letters*, Vol. 93, No. 4, 2004.

[78] D. Soloveichik, M. Cook, E. Winfree, and J. Bruck, "Computation with Finite Stochastic Chemical Reaction Networks," *Natural Computing*, to appear.

[79] T. Tian and K. Burrage, "Binomial Leap Methods for Simulating Stochastic Chemical Kinetics," *Journal of Chemical Physics*, Vol. 121, No. 21, 2004.

[80] E. Voit, "Computational Analysis of Biochemical Systems: A Practical Guide for Biochemists and Molecular Biologists," Cambridge University Press, 2000.

[81] L. Weinberger, J. Burnett, J. Toettcher, A. Arkin, and D. Schaffer, "Stochastic Gene Expression in a Lentiviral Positive-Feedback Loop: HIV-1 Tat Fluctuations Drive Phenotypic Diversity," *Cell*, Vol. 122, pp. 169–182, 2005.

[82] R. Weiss, G. Homsy and T. Knight Jr., "Toward In-vivo Digital Circuits," *Dimacs Workshop on Evolution as Computation Princeton*, 1999.

[83] R. Weiss, S. Basu, S. Hooshangi, A. Kalmbach, D. Karig, R. Mehreja and I. Netravali, "Genetic Circuit Building Blocks for Cellular Computation, Communications, and Signal Processing," *Natural Computing*, Vol. 2, pp. 47–84, 2003.

[84] R. Weiss, T. Knight, and G. Sussman, "Genetic Process Engineering," *Cellular Computing*, Martyn Amos editor, pp.43–73, 2004.

[85] M. Win and C. Smolke, "From the Cover: A modular and extensible RNA-based gene-regulatory platform for engineering cellular function," *Proceedings of the National Academy of Sciences*, Vol. 104, No. 36, pp. 14283, 2007.

[86] E. Zielinska, "Chris Voigt: Biology's Toy Maker," *The Scientist*, Vol. 21, No. 9, 2007.