Cellular Gate Technology

Thomas F. Knight, Jr.
Gerald Jay Sussman

MIT Artificial Intelligence Laboratory

July 1997

A Scientist discovers that which exists. An Engineer creates that which never was. – *Theodore von Karman*

Abstract

We propose a biochemically plausible mechanism for constructing digital logic signals and gates of significant complexity within living cells. These mechanisms rely largely on co-opting existing biochemical machinery and binding proteins found naturally within the cell, replacing difficult protein engineering problems with more straightforward engineering of novel combinations of gene control sequences and gene coding regions.

The resulting logic technology, although slow, allows us to engineer the chemical behavior of cells for use as sensors and effectors. One promising use of such technology is the control of fabrication processes at the molecular scale.

1 Introduction

Cells provide an isolated, controlled environment for carrying out complex chemical reactions. Moreover, they self-reproduce, allowing the creation of many copies with little manufacturing effort. The ability to control cellular function will provide important capabilities in computation, materials manufacturing, sensing, effecting, and fabrication at the molecular scale.

This work is part of an effort to learn how to control the chemical mechanisms of the cell, by co-opting the existing biological mechanism and by constructing novel mechanism. One particular short-term goal is to engineer chemical mechanisms which can be used to implement the digital abstraction—the notion that chemical signals can represent logical true and false (or zero and one) values.

Like any good abstraction, the digital abstraction allows us to ignore the fine details of a complex phenomenon, and concentrate on the essentials of the control process.

The essential features of any digital logic implementation include the ability to distinguish and maintain two distinct values of some physical representation of a signal. This requires the presence of adequate noise margins—an ability to produce outputs whose physical values more perfectly represent a given logical value than the physical representation of their input. Adequate noise margins allow noise and imperfections in a digital system to be reduced, rather than amplified, during complex information processing.

This work attempts to define a series of biologically plausible chemical reactions which can implement such a digital abstraction.

2 Building on Available Biological Mechanisms

Our strategy builds upon existing biological mechanisms. We are fortunate that the natural world has evolved mechanisms similar to those a good engineer would design. By using a modified version of these naturally occurring mechanisms we avoid potentially very challenging issues of engineering complex protein-DNA interactions, allowing us to build interesting structures with a mix and match approach, combined with limited modification.

Here we show the feasibility of building a family of logic gates where the signals are represented by concentrations of naturally-occurring DNA-binding proteins, and where the nonlinear amplification is implemented by *in vivo* DNA-directed protein synthesis.

But first we review the essential aspects of protein manufacturing in the cell.

2.1 Making Proteins within a Cell

Proteins are ordered molecular polymers of 50-1000 amino acids, of 20 different types. Each of the approximately 500-10,000 protein types in a typical cell consists of a unique sequence of the 20 amino acids. Moreover, each protein chain folds into a characteristic three-dimensional structure, which is necessary for its activity.

Many proteins, called enzymes, act as exquisitely selective catalysts for specific chemical reactions, allowing these reactions to take place dramatically faster than they would under normal circumstances. The presence or absence of an enzyme effectively switches reactions on and off within a cell.

The amino acid sequence (and thus the properties) of a particular protein is controlled by the sequence of DNA codons in the associated gene. Triplets of the four DNA nucleotides, A, T, G, and C specify one of 64 code words. These 64 code words specify a start code, three stop codes, and a redundant specification of which of the 20 amino acids should be inserted next into a partially constructed protein molecule.

The information in the DNA is not directly used to manufacture protein. Instead, the primary DNA sequence for a gene is first copied, in a process called *transcription*, into an intermediate form of RNA, called messenger RNA (mRNA). This copying process is under the control of an enzyme complex called *RNA polymerase*. The copying process is not automatic, and control is carefully exercised over which portions of the DNA are copied into mRNA. We will use these control mechanisms as the basis for our logic gates.

The mRNA transcript is then (often in a pipelined manner) used by an enzyme/RNA complex called the *ribosome* to manufacture proteins. This process of manufacturing proteins from mRNA transcripts is called *translation*. The protein manufacturing process is sometimes used as a control mechanism, but is far less widely used than control of mRNA synthesis. mRNA is degraded quite rapidly by the cell, and requires continual replenishment by the creation of new mRNA copies by RNA polymerase from the primary DNA gene.

Proteins are also gradually degraded within the cell, at a sequence dependent rate. The continuing presence of a particular protein thus depends upon its creation by the translation of mRNA transcripts.

$$\text{DNA Gene} \xrightarrow[controlled]{RNApolymerase} \text{mRNA Transcript} \xrightarrow{Ribosome} \text{Protein}$$

2.2 Control of Gene Expression

The creation of mRNA transcripts is carefully controlled within the cell. Each DNA coding sequence (gene) is accompanied by an upstream control region, consisting of non-coding DNA sequences. Some of these sequences signal the binding location for RNA polymerase, the enzyme which catalyzes the creation of mRNA. Other sequences are the binding sites for either repressors or promoters, which are proteins that selectively bind to specific DNA sequences within the control region.

Repressor binding sites typically overlap the RNA polymerase binding site—a protein bound to this site physically interferes with the binding of RNA polymerase. Promoter binding sites are typically located some distance from the RNA polymerase binding site, and the binding of a promoter to such a site makes it easier for RNA polymerase to bind and initiate mRNA production.

In many cases, a single gene-control sequence contains several promoter and repressor regions. In this proposal, we make use of only repressor DNA binding proteins.

2.3 Protein Dimers and Cooperative Binding

An essential aspect of any digital logic gate is a high-quality nonlinearity. In essence, a digital gate must exhibit low gain for signals near a logical zero or one, while exhibiting high gain for signals within the transition regions.

The biological world utilizes two chemical techniques to achieve this highly nonlinear behavior.

The first technique is the use of protein dimers as the biologically active form. The active form of many enzymes, including the DNA binding proteins we propose using, is a bound combination of two copies of the protein. In equilibrium, the concentration of the dimer is proportional to the square of the protein concentration. Higher power nonlinearities can be achieved with tetramers, hexamers, or even higher multimers. Such multimers are common in biologically active protein complexes.

A similar power law behavior is obtained in cooperative binding of proteins to a substrate. Cooperative binding refers to mechanisms in which the first of several protein binding reactions occurs at a relatively low rate, while subsequent binding reactions occur more rapidly, because the presence of the already bound protein enhances the binding affinity.

2.4 Lambda Phage Lysogeny

A naturally occurring genetic switch which controls the lytic/lysogenous switch in bacterial cells infected with the λ phage has been extensively studied. Ptashne's classic book on the subject is required reading to instill modesty among those who would engineer these systems.

While the details of the switch mechanism are (typically) substantially more complex than the techniques proposed here, it is striking that both the CRO and λ repressor are dimeric protein complexes, and that both interact cooperatively with gene control sites.

3 A Biologically Plausible Gate

We can use the naturally occurring mechanisms of controlled mRNA transcription, repressors, cooperative binding, and the degradation of mRNA and proteins as a way to implement a logical inverter.

The "signals" in our logic system consist of concentrations of specific DNA binding proteins, which act as repressors. These concentrations can be thought of as a simple integer count of how many protein molecules of a particular type exist within a single cell.

The "inverters" in our logic system consist of genes—specific DNA coding regions, along with their control sequences—which code for the production of specific proteins. Normally, the coded proteins are themselves DNA binding proteins, which are used as inputs to other such inverters. They could, of course, code for enzymes which effect some other action within the cell, such as motion, illumination, or chemical reactions. Similarly, the input of our gates could consist, not of the output of another logic gate, but of a sensor which creates a DNA binding protein in response to illumination, a chemical in the environment, or the concentration of specific intracellular chemicals.

One additional feature, also present in naturally occurring transcription control mechanisms, is used in our gate to control the output level of the DNA binding protein product. Specifically, the presence of large concentrations of DNA binding protein produced by a particular gene is used to inhibit the transcription of more copies of its mRNA. This results in a predictable amount of the gene product when the gene is turned on.

Logic functions are performed in this model by constructing multiple inverters (genes), each having the same binding protein as a product, but with different control inputs. In many respects, this is similar to the I^2L integrated circuit logic family.

4 The Chemical Model Reactions and their Kinetic Equations

In this section, we consider the detailed chemical reactions of a single inverter. A detailed understanding of the static and dynamic behavior of the inverter is the key to digital gate design. More complex gates are simple extensions of this understanding. In the case of our proposal, complex gates are formed simply by constructing inverters with distinctive control sequences, but which produce the identical gene products.

Consider the inverter with DNA binding protein input A and DNA binding protein output B. B is manufactured by ribosomes acting on mRNA copied from a DNA coded gene, B_G. B is destroyed by scavenging mechanisms which continually degrade cellular protein.

The rate of production of the mRNA transcripts is controlled by promoter and repressor regions upstream from the coding region of the gene for B, B_G. In this example, we will assume that both A and B act as repressors for the transcription of the B_G.

We will model the binding of the DNA binding proteins A and B as reversible chemical reactions, transforming the gene B_G into a repressed (inactive) form B_GA and B_GB respectively. Note that we require n molecules of A to inactivate the gene—this is the source of the needed nonlinearity. We will obtain excellent inverter behavior with $n = 4$.

$$B_G + nA \xrightarrow[\text{binding}]{k_1} B_GA$$

$$B_G + B \xrightarrow[\text{binding}]{k_2} B_GB$$

$$B_GA \xrightarrow[\text{dissociation}]{k_3} B_G + nA$$

$$B_GB \xrightarrow[\text{dissociation}]{k_4} B_G + B$$

Only the active form, B_G will be transcribed and translated to form B in the essentially irreversible reaction

$$B_G \xrightarrow[\text{dna pol, ribosome}]{k_5} B_G + B.$$

The breakdown of A and B is modelled by the equations

$$A \xrightarrow[\text{breakdown}]{k_6}$$
$$B \xrightarrow[\text{breakdown}]{k_7} .$$

The kinetic equations for this chemical mechanism can be written as a set of coupled differential equations corresponding to the conservation of material. We added a "drive" term, which is not part of the chemistry, to allow us to manipulate the system numerically, by setting a schedule for production of the protein A. This would not be present in any biological implementation, but then the input variable $[A]$ would be coupled to the rest of the system by being the output variable of some other gate or sensor.

$$\frac{d[B_G]}{dt} = -k_1[A]^n[B_G] - k_2[B][B_G] + k_3[B_GA] + k_4[B_GB] \tag{1}$$
$$\frac{d[B_GA]}{dt} = k_1[A]^n[B_G] - k_3[B_GA] \tag{2}$$
$$\frac{d[B_GB]}{dt} = k_2[B][B_G] - k_4[B_GB] \tag{3}$$
$$\frac{d[B]}{dt} = k_5[B_G] + k_4[B_GB] - k_2[B_G][B] - k_7[B] \tag{4}$$
$$\frac{d[A]}{dt} = \text{drive} + nk_3[B_GA] - k_6[A] - nk_1[A]^n[B_G] \tag{5}$$

The bound and unbound copies of the B gene are conserved, and equal to the gene copy number in the cell, so we may define $[B_0]$ as the concentration of any form of B_G:

$$[B_0] = [B_G] + [B_GA] + [B_GB]. \qquad \text{(from 1,2,3)} \tag{6}$$

In equilibrium, the time derivatives of concentrations are zero, giving us the equations:

$$
\begin{aligned}
[B_G A] &= \frac{k_1}{k_3}[A]^n[B_G] && (7)\\
[B_G B] &= \frac{k_2}{k_4}[B][B_G] && (8)\\
[B_G] &= [B_0] - [B_G A] - [B_G B]\\
&= [B_0] - \frac{k_1}{k_3}[A]^n[B_G] - \frac{k_2}{k_4}[B][B_G]\\
&= \frac{[B_0]}{1 + \frac{k_1}{k_3}[A]^n + \frac{k_2}{k_4}[B]} && (9)\\
k_5[B_G] + k_4[B_G B] &= k_2[B_G][B] + k_7[B] && (10)\\
\text{drive} + nk_3[B_G A] &= k_6[A] + nk_1[A]^n[B_G] && (11)
\end{aligned}
$$

Simplifying further, we arrive at the equilibrium input/output relationship between $[A]$ and $[B]$:

$$
\begin{aligned}
[B_G B] &= \frac{k_2}{k_4}[B][B_G] && \text{(from 10, 8)} && (12)\\
\frac{k_7}{k_5}[B] &= \frac{[B_0]}{1 + \frac{k_1}{k_3}[A]^n + \frac{k_2}{k_4}[B]} && \text{(from 12,9)} && (13)\\
[B_0] &= \frac{k_7}{k_5}[B] + \frac{k_7 k_1}{k_5 K_3}[A]^n[B] + \frac{k_7 k_2}{k_5 k_4}[B]^2 && && (14)\\
[A]^n &= \frac{[B_0] - \frac{k_7 k_2}{k_5 k_4}[B]^2 - \frac{k_7}{k_5}[B]}{\frac{k_7 k_1}{k_5 k_3}[B]} && && (15)\\
[A]^n &= \frac{k_5 k_3 [B_0]}{k_7 k_1 [B]} - \frac{k_2 k_3}{k_4 k_1}[B] - \frac{k_3}{k_1} && && (16)
\end{aligned}
$$

Letting $\alpha = \frac{k_1}{k_3}$, $\beta = \frac{k_2}{k_4}$ and $\gamma = \frac{k_5}{k_7}$, we arrive at the equation

$$
\alpha[A]^n = \gamma\frac{[B_0]}{[B]} - \beta[B] - 1 \qquad (17)
$$

This relationship defines the input/output transfer function of our system. The parameters α, β, and γ are ratios of kinetic constants, and $[B_0]$ is the conserved concentration of the gene.

5 Equilibrium Behavior

We set $n = 4$. We have found that this value produces a good quality inverter.

We need to put further constraints on our concentrations to make a useful inverter. In particular, we want the output of an inverter to be a signal in the same analog range as the input, and we may require the output to go through the halfway point just when the input does. These constraints will restrict the possible values of the parameters α, β, γ, and $[B_0]$.

Let's start by declaring that the range of signals is the interval $[0, 1]$. So we require that $[A] = 0$ when $[B] = 1$. This is a free choice, because it just determines the units that we use to measure the concentrations. We can also require that $[A] = 1/2$ when $[B] = 1/2$. This is an actual restriction on our transfer characteristic. Plugging these constraints into the equation (17) results in the relationships:

$$\beta = \gamma[B_0] - 1 \tag{18}$$

$$\alpha = 24\gamma[B_0] - 8 \tag{19}$$

These leave us with exactly one free parameter for the transfer characteristics of our inverter, $\gamma[B_0]$. Do we get a good inverter for reasonable values of this parameter? The answer is yes.

Figure 1 shows that the relationship in equation (17) between the input and output concentrations yields the classic transfer curve of a good digital inverter. (Here $n = 4$ and $\gamma[B_0] = 2.5$.) Note, in particular, the low gain for high and low input concentrations, separated by a relatively high-gain transition region. This nonlinearity is the essence of digital gates, and forms the basis for effectively rejecting small variations in the input signals—that is, for attenuating the input noise.

It is encouraging that the proposed inverter shows promising low sensitivity to variations in the chemical rate constants. The sets of curves shown in figures 2 and 3 display the results of halving and doubling of each of the constants α, β in equation (17), while holding $\gamma[B_0]$ constant. Of course, changing these constants violates our constraints (equations (18,19)) but the resulting system is still quite serviceable.

What is even more spectacularly encouraging is that the value of $\gamma[B_0]$ can be varied over a huge range without making this inverter unusable. In figure 4 we see that varying $\gamma[B_0]$ over the entire range $[1, 50]$ has almost no effect on the characteristic shape of our inverter.

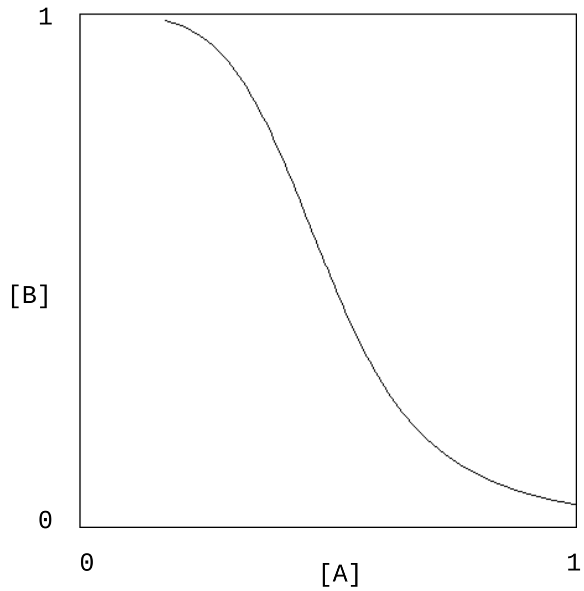

Figure 1: The DC transfer curve for our mechanism is similar to the characteristic of an NMOS inverter.

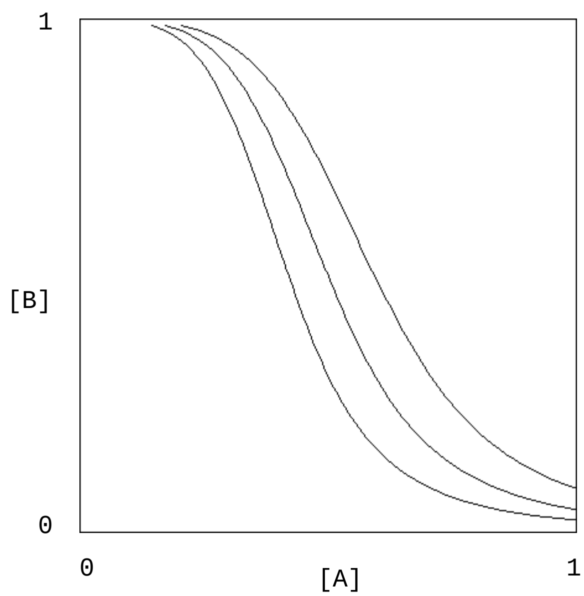

Figure 2: Variation in the parameter α affects the inverter threshold of the gate. Here we see the effect of a factor of four variation in α.

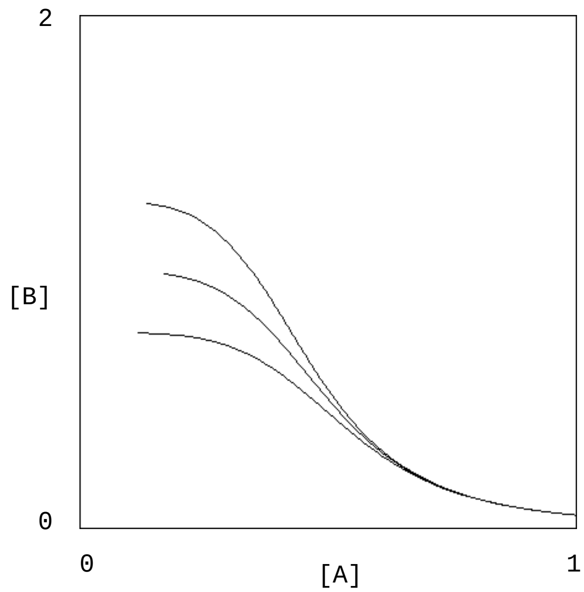

Figure 3: Variation in the parameter β affects the concentration representing a logic *one*. Here we see the effect of a factor of four variation in β.

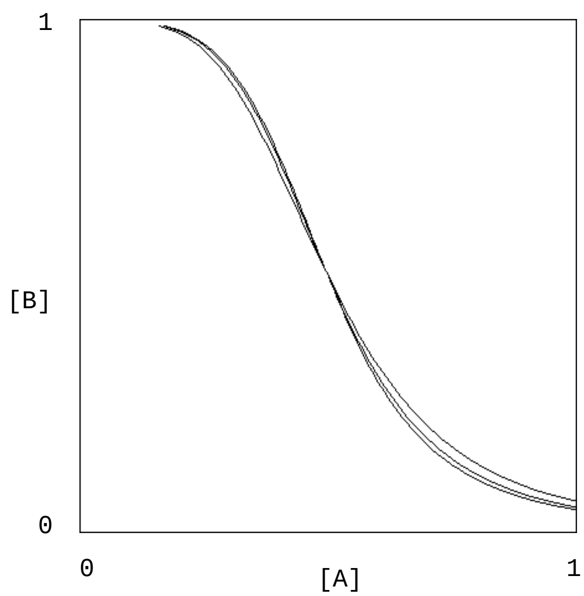

Figure 4: Variation in the parameter $\gamma[B_0]$ has almost no effect on the inverter characteristic. Here we vary $\gamma[B_0]$ over a factor of fifty

Of course, this is still useless unless the kinetic constants for biologically feasible reactions are within these ranges. We have not yet done this analysis, but we hope to have results to report shortly.

6 Dynamic Behavior

Static transfer curves do not address the dynamic behavior of logic circuits. Although the static transfer characteristic of the system was determined by one free parameter, the dynamic characteristics are much more complicated. Even given the constraints on the transfer characteristic of the last section the dynamics depends on choices of γ, $[B_0]$, k_3, k_4, k_6, and k_7. We have not yet explored this space, so we show here the behavior of only one choice. In the simulations we have chosen the following values for the free parameters—we have not tried to scale these for biologically plausible time scales and concentrations. But similar behavior can be obtained over very wide ranges of variations of these parameters.

$$n = 4, \gamma = 25, B_0 = 0.1, k_3 = 0.2, k_4 = 0.03, k_6 = 1.0, k_7 = 1.5$$

In figure 5 we show behavior of our gates when stimulated with a abrupt transition (square wave) input signal, by turning on and off production of protein A. The concentration of A then changes, as shown in the simulation, due to the sudden production change, in combination with the concentration-dependent destruction rate, and its binding to B_G. The binding of A to DNA inhibits production of B, entailing the changes in the concentration of B, thus yielding the inverter behavior desired.

Note that the simulation shown in figure 5 the inverter is not loaded by connection to the next stage. We must show that the inverters work with an attached load. One way to do this, which also shows other important features of the dynamics, is to hook three of these inverters up to make a ring oscillator. Simulation of this oscillator (see figure 6) demonstrates that we actually have working inverting amplifiers. In this simulation we started with a random initial condition. We see that our parameters are not particularly optimal, but the system oscillates with reasonable swing.

7 Implementation Issues

Any new digital logic family has important characteristics and limitations which must be understood in order to effectively design with them. In this section, we

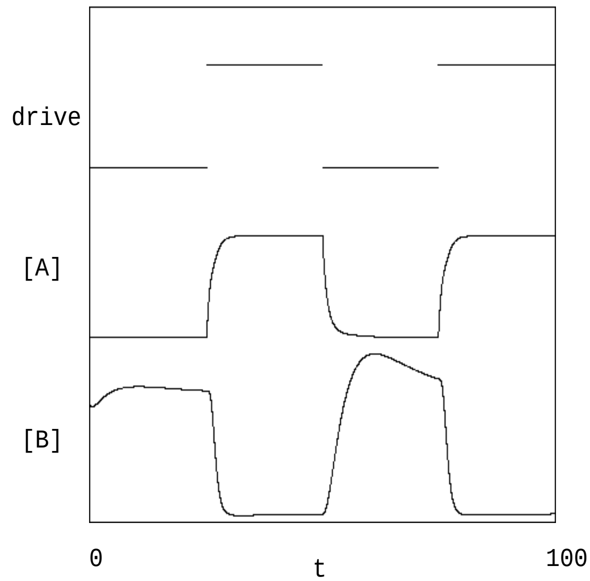

Figure 5: The top trace shows the externally-imposed drive, the rate of production of protein A. The middle trace shows the response of $[A]$. The bottom trace shows the response of $[B]$. The two rises of $[B]$ do not look the same because the simulation was started with an arbitrary initial state. Subsequent rises of $[B]$ would look very much like the second one.

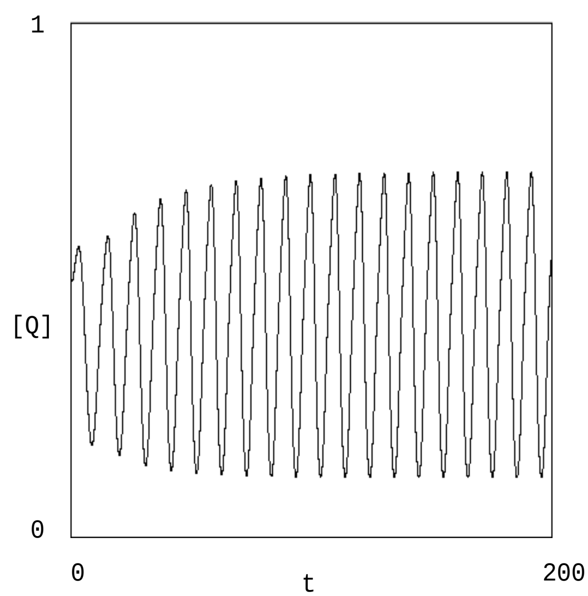

Figure 6: The ring oscillator oscillates at a rather high frequency, limited by the rise time, the fall time, and the storage time of the inverter.

have attempted to predict some of these issues, but inevitably there will be many which we will overlook until real implementations of these gates is underway.

7.1 These Gates are Slow

Perhaps the most dramatic difference between our biological gates and conventional logic gates is the speed differential. Electrical gates now function with delays of tens of picoseconds. Biological gates constructed using this methodology will have delays governed by the speed of protein manufacturing—perhaps many minutes. Roughly speaking, we should think of this logic family as functioning at frequencies measured in millihertz, rather than at rates measured in Megahertz.

While other biological mechanisms might be constructed which could function (optimistically) at kilohertz rates, such structures will require a degree of engineering finesse which we believe will not be available in the short term. Protein design, and, especially, protein-complex design, required for such high-speed gates, is not sufficiently well understood at the present time. Indeed, the major requirement may be, ironically, better and faster computational tools.

7.2 Complexity Limitations

A critical resource in the design of complex logic circuits within a cell is the availability of a sufficient number of distinct DNA binding proteins. Not only must many such proteins be found, but the set employed must *not* be used elsewhere within the host cell control mechanisms. We estimate that tens to hundreds of such proteins exist naturally. With luck, we can perhaps learn how to engineer many others within this limited domain.

A second limitation comes from the requirement of finite concentrations of these proteins within the cell. Cells have finite volume, and the requirement that many copies of a protein exist to have effect, together with the complexity requirement for many distinct proteins, leads to an upper limit on the logic complexity which can be performed within a single cell. We are confident, however, that logic circuits of an interesting level of complexity can be constructed.

7.3 Cell Cycle Coordination

Cells, particularly bacterial cells, are not static objects. They undergo a growth and division cycle which has a profound effect on the biochemistry. For example, the value of $[B_0]$, the total number of copies of the gate gene within the cell, varies over time because the DNA is being replicated prior to cell division. Gates must be robust against such potentially major changes in internal cell chemistry.

7.4 What Organism?

The choice of organism is critical. One view is that we should experiment with the simplest possible organism, perhaps Mycoplasma capricolum, which has the advantage of a small genome, and is already sequenced. But, it is s difficult to culture, and few workers have experience with it.

Another view espouses the standard biological prokaryote, Eschericia coli. It, too is now sequenced, although with a genome twelve times as large. Extensive experience and the wide availability of tools argue strongly for its adoption.

A third view proposes that we use a eukaryote, specifically the yeast, S. cervisiae, as an ideal organism, because of its richer genetics, isolated nucleus, and more complex transcription mechanisms.

Our current approach favors the use of E. coli due to its wide availability and overwhelming popularity. Eventually, the advantages of using a dramatically simplified cell seem large. Engineering such a simplified cell may be an important early goal.

8 Applications

Cellular Computing opens a new frontier of engineering that will dominate the technology of the next century. Employing information technology, the future holds promise for the development of means to organize and control biological processes that are just as effective as our current mastery of electrical processes.

In particular, biological cells are self-reproducing chemical factories that are controlled by a program written in the genetic code. Current progress in biology will soon provide us with an understanding of how the code of existing organisms produces their characteristic structure and behavior. As engineers we can take control of this process by inventing codes (and more importantly, by developing automated means for aiding the understanding, construction, and debugging of such codes) to make novel organisms with particular desired properties.

Besides the obvious application of control of biological processes to medicine, we will be able to co-opt biological processes to manufacture novel materials and structures at a molecular scale. The biological world already provides us with a variety of useful and effective mechanisms, such as flagellar motors. If we could co-opt cells to build organized arrays of such motors, with accessible interfaces for power and control, we could see how this could be of engineering significance. Common, biologically available conjugated polymers, such as carotene, can conduct electricity, and can be assembled into active components. If we, as engineers, can acquire mastery of mechanisms of biological differentiation, morphogenesis, and pattern formation, we can use biological entities of our own design as construction agents for building and maintaining complex ultramicroscopic electronic systems. Such systems will have better performance and reliability then technologies based on less precisely controlled chemical processes. Of course, one of the most important products of mass-produced molecular-scale engineering will be extremely compact, efficient, and effective computing mechanisms.

Thus, in spite of the long gate delays in cellular computing mechanisms, the fact that cells can reproduce and organize into precisely arranged and differentiated tissues means that we can use them as the (very slow) agents of molecular-scale manufacturing of macroscopic objects. It is the resulting objects that we desire—they may contain electrical circuitry with picosecond cycle times. The slow biological systems are our machine shops, with proteins as the machine tools, and with DNA as our control tapes.

This work is supported, in part, by the DARPA/ONR Ultrascale Computing Program under contract N00014-96-1-1228 and by the DARPA Embedded Computing Program under contract DABT63-95-C130.

9 References

- Lodish, Harvey; Baltimore, David; Berk, Arnold; Zipursky, S. Lawrence; Matsudaira, Paul; and Darnell, James; "Molecular Cell Biology," third edition, Scientific American Books, New York, 1995.
- Watson, James D.; Hopkins, Nancy H.; Roberts, Jeffery W.; Steitz, Joan A.; and Weiner, Alan M.; "Molecular Biology of the Gene," fourth edition, Benjamin/Cummings, Menlo Park, 1987.
- Ptashne, Mark, "A Genetic Switch: Phage λ and Higher Organisms," second edition, Cell Press, Cambridge, 1992.
- Palmer, Trevor, "Understanding Enzymes," fourth edition, Prentice Hall, Hertfordshire, 1995.