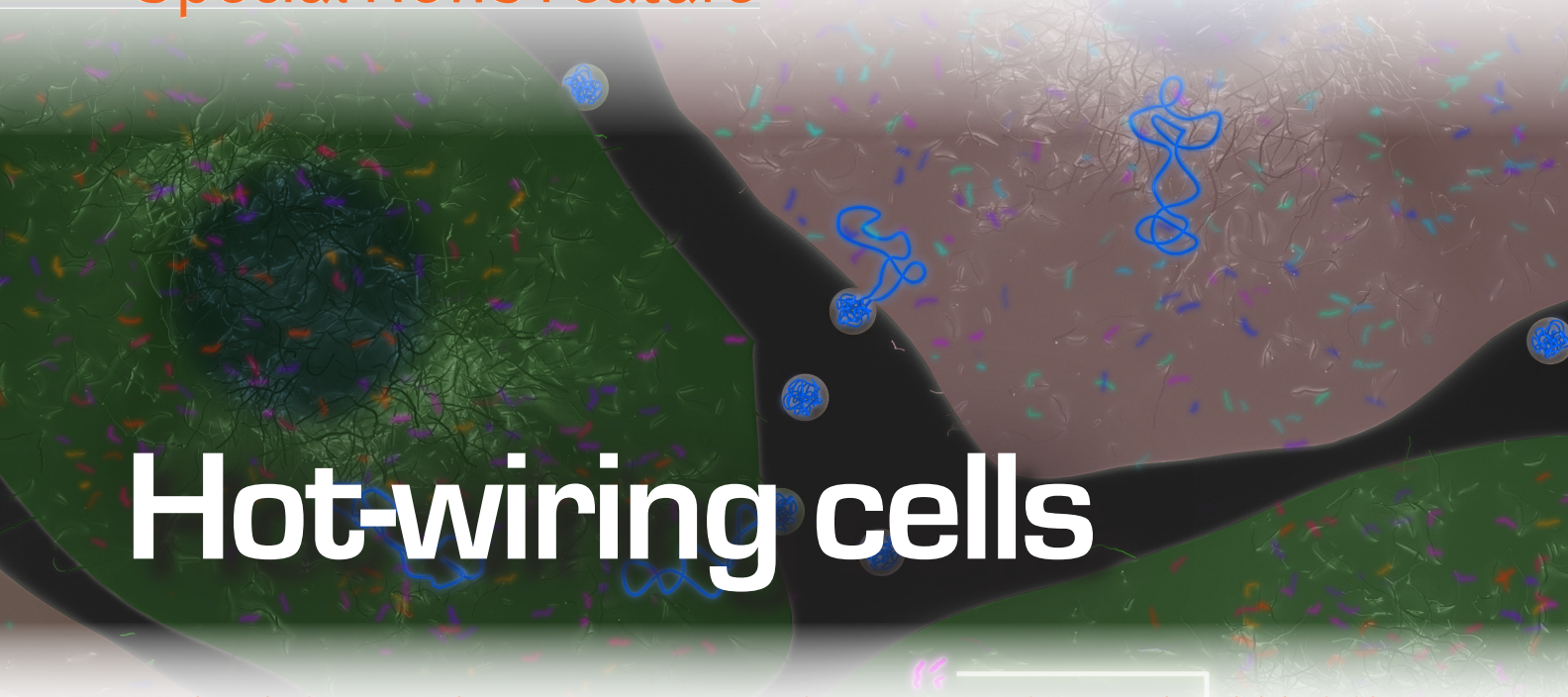
Special News Feature



Hot-wiring cells

Synthetic biologists are designing genetic circuits of increasing complexity. But how did the field get to this point, and where is it going? Nathan Blow examines the challenges, and potential applications, of engineering gene circuits.

According to Jim Collins, a synthetic biologist at Boston University who is exploring the use of transcription factors as key components in synthetic gene circuits, bioengineers come in two flavors.

"In their youth, those that take things apart become systems biologists, while those that put things together and tinker go into synthetic biology." Collins finds himself in the latter class.

A physicist by training, Collins was led somewhat inadvertently into the world of synthetic biology by his BU colleagues. "It was suggested that I take my physics background and apply those skills to reverse engineer natural genetic networks," recalls Collins. This was in 1996 and while microarray technology had been developed, there were still very few large-scale genetic datasets that could be used as a starting point in any reverse engineering effort. "In the end, we ran away from the problem."

But not entirely. While Collins did not have the tools to take apart and reverse engineer naturally occurring networks, he and his graduate student at the time Tim Gardner came to realize that they did have the capabilities to assemble some basic molecular components, proteins, into genetic circuits that could then function in cells — akin in some ways to an electrical engineer wiring a light switch in a house.

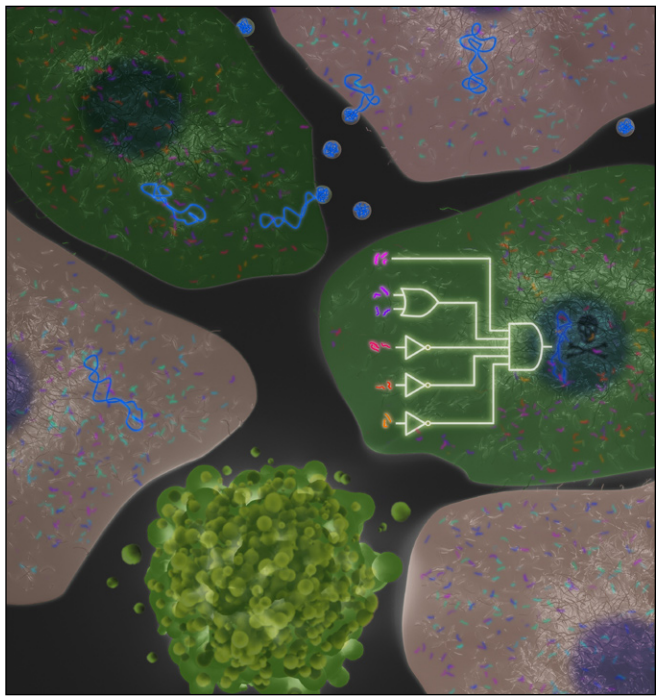
In 2000, Collins and Gardner's forward engineering efforts paid off, resulting in a publication in *Nature* detailing the construction of a genetic toggle switch in *Escherichia coli*, the first synthetic, bistable gene-regulatory network to be described. That same issue of *Nature* also featured a report by Michael Elowitz and Stanislas Leibler describing a synthetic oscillating gene network in *E. coli* that periodically induced synthesis of green fluorescent protein as a readout of cell state. These two studies were the first demonstrations of the potential of engineering basic gene circuits, and with their publication, a new era of genetic circuit design in synthetic biology started.

Lessons in complexity

Collins' bistable genetic toggle was simple, a switch composed of two repressors and two promoters with each promoter inhibited by the repressor that is transcribed by the opposing promoter. But this simple toggle also turned out to be robust (i.e. exhibiting bistability over a range of parameters), a key property in circuit design. In fact, in the concluding paragraph of the article, Collins even makes mention of the idea that such toggles could find use in gene therapy and biotechnology applications in the future. As it turns out, in hindsight though, moving from this *E. coli* toggle toward more complex circuitry introduces new problems and challenges — as well as possibilities.

A genetic circuit needs a couple basic components to function properly. First, there needs to be a sensor module capable of identifying a specific input. From there, the sensor needs to be connected to a computational module. These modules, sometimes

Jim Collins, a synthetic biologist at Boston University, is exploring the use of transcription factors as key components in synthetic gene curcuits. Source: J. Collins



Visual representation of an RNAi-based gene circuit for detection in HeLa cells. Source: Y. Benenson

referred to as logic gates, take the input and "calculate" the appropriate output. Logic gates can make basic decisions based on the input provided. For example, a NOT logic gate will result in one output over another while the AND logic gate will result in the same output. Finally, an output component is needed in the circuit to register the activation or repression of the circuit (for example, GFP was the output used in that early oscillator network study from Elowitz and Leibler). And that's not all, these circuit components should be both robust (exhibiting responses over a range of parameters like Collins' bistable toggle) and sensitive, not to mention tunable or adaptable to new functions.

If it is not obvious by this point, creating a synthetic genetic circuit is no small task. The good news here is that parts or building blocks, basic circuit components, are available — the bad news is that components generated for other studies cannot simply be added to new circuit designs — this is often not a direct plug-and-play deal.

"If I purchased 1000 transistors, I would not test every transistor prior to use," says Collins. "However, when it comes to synthetic biology, you do need to test every genetic component prior to use in a particular circuit." Why? Synthetic gene circuits, unlike the electrical circuits in your house, operate in the context of a whole cell; they interact with other proteins and metabolites as well as the unique cellular environment. And this is the reason that increasing circuit complexity, well, increases circuit complexity.

But even in the face of these challenges, researchers have developed unique, clever circuit designs with components capable of logical computations, filtering, oscillation, noise propagation and memory from a variety of inputs.

Massive input

Yaakov (Kobi) Benenson, an assistant professor in the department of Biosystems Science and Engineering at the Swiss Federal Institute of Technology (ETH Zurich), initially became interested in the possible applications of gene circuits during his doctoral work in the lab of Ehud Shapiro at the Weizmann Institute of Science.

At the time, Benenson was focused on creating programmable biochemical-based computing systems.

"Even then, I always had medical applications in mind," says Benenson. "Exploring the disease states of single cells, interrogating information from inside cells — this interested me."

His work in the Shapiro lab had focused on in vitro biochemical computing systems. But to extend these studies to medical applications, Benenson realized that he needed to go inside the cell with his programmable computing systems — leading him directly into the world of synthetic biology and engineered gene circuits.

While gene circuits provide the opportunity to monitor the state of a cell (input coming from the cell, output that can be directly read like in the case of the first oscillating gene network), these engineered circuits also provide the opportunity for other interesting applications. For example, gene circuits could be engineered to activate expression of therapeutic transgene in response to a specific input. Similarly, bacteria can be engineered with specific metabolic gene circuits to respond to cues and produce novel outputs. Benenson saw this as the next step for his DNA computing research started in the Shapiro lab.

After graduating from Shapiro's lab, Benenson moved to Harvard University as a Bauer Fellow at the FAS Center for Systems Biology. It was here that Benenson, teaming up with Ron Weiss who at the time was at Princeton University, made his mark in circuit engineering with the demonstration that RNAi could act as a component in gene circuits in human kidney cells.

"The hallmark of decision making in cells is the requirement for a lot of inputs," explains Benenson. "And how do we build large circuits to process multiple inputs? RNAi presents a scalable mechanism to accomplish this."

To take advantage of RNAi as a circuit component, Benenson and Weiss first had to construct two or more mRNAs encoding the same proteins but with different non-coding regions, implementing an OR logic between the levels of these mRNAs. Next, sets of siRNA targets were added to the 3′ ends of the mRNAs. Endogenous inputs and their effects on these siRNAs were evaluated to confirm robust output from the mRNAs. From this starting point, the basic system could evaluate signals in a logical manner: in the cases where inputs block all siRNAs targeting the same mRNA, an mRNA will be transcribed, creating an AND logic. In addition, if the input activates an siRNA, the mRNA will be targeted by this siRNA and the output not transcribed (and vice versa), creating a NOT logic. The combination of OR, AND, and NOT gates can effectively support any logic computation.

Yaakov (Kobi) Benenson, an assistant professor in the department of Biosystems Science and Engineering at the Swiss Federal Institute of Technology (ETH Zurich), initially became interested in the possible applications of gene circuits during his doctoral work in the lab of Ehud Shapiro at the Weizmann Institute of Science. Credit: ETh Zurich/Tom Kawara

"Another nice thing about RNAi-based circuits is the fact that it is easy to develop sensors for every miRNA in a cell," notes Weiss who is quick to add that with nearly 1400 mammalian miRNAs currently identified, a lot of information can be gleaned from this approach. Weiss and Benenson demonstrated this nicely in a 2011 follow-up article published in *Science* where the pair described multi-input RNAi-based logic circuits that enable the cells themselves to describe who, or what, they are. In this circuit design, cell classification is accomplished when the circuit senses the expression levels of a set of endogenous miRNAs and then triggers a cellular response if those expression levels hit a certain threshold.

"The nice part is that this approach can be easily generalized and adapted to profiling different miRNAs and therefore different cell types," explains Weiss, thus enabling the monitoring and identification of a potentially wide range of cells. In fact, Weiss says that with minor tweaks to the circuit, they are now able to even more robustly classify a growing range of cells using different endogenous miRNAs.

Transcription factor-based components

The use of short RNAs in gene circuits is a relatively new development in circuit engineering, the traditional circuit players have been proteins, with directed evolution approaches playing a significant role in adapting these biomolecules for use in specific circuit configurations. Critically, according to Collins and others, any biomolecule to be used as a component or module in a gene circuit should be reliable in behavior, able to adapt to specific parameters, and interface with other biomolecules (as in the case of siRNA).

For some researchers, transcription factors (TFs) present the perfect component to meet these requirements. TFs can be activated at specific times, they can transcribe specific genes, and they interact with protein components of the transcriptional machinery, thus harnessing these molecules has been a goal for many circuit builders.

The first obstacle to overcome with TFs was figuring out how to get synthetic versions of these molecules to specific genomic location to transcribe other circuit components. Enter one of the new workhorses of modern synthetic biology — the zinc finger. Combine a zinc finger with a TF (a so-called synthetic TF or "*s*TF") and the result is a TF possessing target site specificity.

In early 2012, Harvard University synthetic biology researcher Pamela Silver described in *Nucleic Acids Research* how tunable zinc finger TFs could be incorporated into genetic circuits to perform logic computations. Silver and her colleagues generated 15 transcriptional activators and 15 repressors that displayed various levels of repression and induction and then used these to perform a variety of simple OR, AND, NOR, and NAND logic operations.

On the heels of this work, and with the realization that expanding the number of *s*TFS could expand circuit possibilities, another team of researchers detailed a framework for creating novel *s*TFs. This work, published in *Cell* in August 2012

by Collins and his colleagues Tim Lu from Massachusetts Institute of Technology (MIT) and Mo Khalil at Boston University, describes the ways in which output strength and transcriptional cooperativity of these TFs can be "tuned." Surprisingly, the researchers found that even subtle changes in these properties allowed for distinct functional roles.

Although much of the work up to now has focused on creating either RNAi-based or sTF-based circuits, these components are not mutually exclusive. According to Weiss, hybrid approaches where the advantages and specificities of both components are exploited for circuit design could yield novel circuit designs.

While providing a new set of tools to engineer sTFs and, therefore, create new circuits, the *Cell* paper, along with other recent articles describing increasingly complex circuits, also demonstrates another outcome of today's circuits engineering efforts — these circuits and how they function in cells enhances our basic understanding of how natural cellular networks sense and analyze signals.

Insulation from the elements

In August 2012, Chris Voigt, a researcher who works across town from Collins at MIT, and his colleagues described the most complex synthetic genetic circuit to date. Voigt's group engineered the first layered genetic circuits where the circuit components do not interfere with one another. Instead, four sensors for different molecules work together without interfering with one another.

Although layered and complex, at the core, this new circuit is not terribly different in principle from either Collins' bistable genetic toggle or Elowitz and Leibler's oscillatory network. However, in Voigt's case, the circuit components do not simply not interact, they are engineered to be "insulated" from one another and the rest of the cell — no small feat in the complex environment of a cell.

While many circuit components exist today, one challenge that will always remain when building a genetic circuit is the environment, and how the components of any circuit interact with other molecules, genomic locations, and physical conditions within the cell. "It is impossible to completely isolate circuits, but unwanted interactions can be brought to a minimum," notes Benenson.

According to Benenson, a crucial component of proper circuit function is a rational design and integration of any gene circuit (see sidebar:"Getting it all together"). And as Collins noted, components should be tested prior to use. But might it be possible to predict how a particular circuit will function in the cell even before integration and experimental testing?

Computing circuits

Weiss started his career as a computer scientist. In the mid-1990s, around the time Collins was postulating his bistable toggle switch, Weiss was wondering if it might be possible to program cells like one would program a computer. "Or maybe even use biology as a way to program a computer," recalls Weiss.

This idea led Weiss into his doctoral studies with Tom Knight, a senior researcher in the MIT Computer Science and Artificial Intelligence Laboratory and a pioneer in the field of artificial intelligence. Around this time, Knight was also making his move into synthetic biology — actually taking classes at MIT to learn key biology concepts and molecular biology lab techniques. It was Weiss who helped Knight establish a "wet-lab" in Knight's computer lab at MIT, and then together, they began exploring genetic circuit design.

Since his time in Knight's lab, Weiss has gone on to design a number of genetic circuits of increasing complexity, but he's still part computer scientist, which explains his interest in trying to inform circuit design through computation analysis.

Computer-assisted design and analysis of gene circuits (think CAD for genetic engineering) is by most accounts in its infancy, something Weiss also acknowledges to some extent. "In principle, I agree. But the prediction capabilities are getting better."

In fact Weiss says that simple toggles,

Getting it all together

One part of circuit engineering that has become easier for researchers in recent years is getting all those molecular components into cells in the proper location. "Two years ago, cloning was a major challenge, taking much time and effort," says Yaakov (Kobi) Benenson. But the continuing development of ligation independent cloning (LIC) methodologies, alongside the refinement of nuclease-based approaches for genome engineering (i.e. zinc finger nucleases and TALENs), is greatly enhancing the front-end of circuit engineering.

LIC is a cloning approach wherein the assembly of DNA fragments is accomplished through the use of small overlapping ends and enzymes. In most cases, this approach works with a small number of fragments over a limited size range. But in 2009, Daniel Gibson and colleagues at the J. Craig Venter Institute described an assembly reaction where a larger number of fragments (upward of 10) could be assembled in a single reaction without the need for restriction digests. "Gibson assembly was really transformative for what is possible in a short amount of time," says Ron Weiss. By using this approach, Weiss and others are now able to assemble and clone a variety of circuit components in a week's time — a rate that was unheard of prior to 2009.

Following cloning and assembly, the decision between transient transfection or stable integration of a given circuit has to be made. Many investigators still use transient transfection since any therapies would require transient transfection. When it comes to stable integration though, genome engineering tools — including ZFNs and TALENs — can be used to integrate a synthetic circuit into specific loci in almost any cell type. Specific integration is important as the genome context can play a role in circuit functionality.

Still, stable integration can cause issues for circuit builders. "Turns out that, as the circuits get longer, there is a greater chance of silencing," explains Weiss.

Yet another development when it comes to testing your circuits is the iterative plug-and-play method described in 2012 by Jim Collins and his group. Here, a series of vectors were developed to enable the rapid construction and modification of larger gene circuits. The approach is also compatible with LIC and Gibson assembly approaches, thus providing enhancements in both upstream cloning of circuits as well as in their downstream analysis.

All these developments make Benenson hope that one day DNA cloning will become an even more routine task in synthetic biology. "Hopefully, you will be able to type the sequence and get the DNA four weeks later," he says. "In this way, most of the time will be spent on looking at the circuits rather than cloning." Which in the end, he adds, is much more time intensive — not to mention interesting. —NB

cascades, and oscillators can now be computationally predicted fairly well. The breakdown in computational predictions comes as the complexity of the circuit increases and more components are added.

The circuits themselves are a large part of the prediction problem. "Current circuits are not particularly robust," notes Collins, as cellular conditions can impact their output and values, especially in larger circuit designs.

But in July 2012, Weiss along with several MIT colleagues described a set of computational tools to guide (but not predict) large-scale circuit design in an article in *PLoS Computational Biology*. This study examined the use of computational tools to determine the impact of combinations of functional modules on optimal system performance in large circuits. In essence, Weiss and his colleagues programs don't predict circuit output — instead they provide information on the possible impacts components can exert on the circuit.

"Computational analysis here is a parameter; modify a component and it has a large impact on the circuit," explains Weiss. Perform such analysis repeatedly, and you begin to optimize your circuit design. In this case a modular network for artificial tissue homeostasis was optimized and in the process the researchers discovered that features previous associated with robust circuits (noise attenuation for example) actually were detrimental for this network — a result and observation that Weiss's team owes in no small part to their computational analyses.

Engineering for the masses?

For the moment, engineering genetic circuits is still the province of synthetic biologists. Although a growing number of components and modules are available to try from repositories such as The Standard Registry of Biological Parts (partsregistry.org), stringing these together into a working circuit is not as straightforward as one might hope at the moment.

"Getting parts into cells works well now," says Collins. But he is quick to add that designing circuits that function in the way one would like in a given cell remains the big challenge at the moment.

And while electrical engineers can quickly model and test their systems and designs prior to implementation, it is clear that synthetic biologists do not have this luxury at the moment — a further roadblock for the would-be novice circuit builder.

"Progress is not at that level, therefore experimental characterization of components in isolation must be done," says Collins. Still, from Weiss's computational strategies for identifying critical components to actually testing component experimentally prior to building a circuit, approaches aimed at understanding and defining circuit design principles seem to be in vogue now. Collins recently reported on what he calls a "gene circuit breadboard" in the journal *Nature Methods*; a series of vectors to enable rapid construction, and post-assembly tuning, of basic synthetic gene networks (also see sidebar: "Getting it all together").

In the end, circuit designers are making tremendous progress in understanding how to construct synthetic circuits. The question now is when will design give way to applications? Only the synthetic oscillator in the cell will tell.

Written by Nathan Blow, Ph.D.

BioTechniques 53:351-355 (December 2012) doi 10.2144/000113965