LETTER

doi:10.1038/nature09679

Distributed biological computation with multicellular engineered networks

Sergi Regot[1]*, Javier Macia[2]*, Núria Conde[1,2], Kentaro Furukawa[3], Jimmy Kjellén[3], Tom Peeters[1], Stefan Hohmann[3], Eulàlia de Nadal[1], Francesc Posas[1] & Ricard Solé[2,4,5]

Ongoing efforts within synthetic and systems biology have been directed towards the building of artificial computational devices[1] using engineered biological units as basic building blocks[2,3]. Such efforts, inspired in the standard design of electronic circuits[4–7], are limited by the difficulties arising from wiring the basic computational units (logic gates) through the appropriate connections, each one to be implemented by a different molecule. Here, we show that there is a logically different form of implementing complex Boolean logic computations that reduces wiring constraints thanks to a redundant distribution of the desired output among engineered cells. A practical implementation is presented using a library of engineered yeast cells, which can be combined in multiple ways. Each construct defines a logic function and combining cells and their connections allow building more complex synthetic devices. As a proof of principle, we have implemented many logic functions by using just a few engineered cells. Of note, small modifications and combination of those cells allowed for implementing more complex circuits such as a multiplexer or a 1-bit adder with carry, showing the great potential for re-utilization of small parts of the circuit. Our results support the approach of using cellular consortia as an efficient way of engineering complex tasks not easily solvable using single-cell implementations.

Engineered living cells have been designed to perform a broad variety of functions[8–16] but, with few exceptions, complex computational constructs (such as comparators, bit adders or multiplexers) are difficult to obtain and reuse[17]. Moreover, cell-cell communication requirements rapidly grow with circuit complexity, thus limiting the combinatorial potential of the constructs. One way of overcoming these difficulties is to use cellular consortia[18] based on the idea that external communication between cells in populations involving either single[19–21] or multiple[22–24] cell types would perform functions difficult to be implemented using individual strains. Here, we apply this view to a novel distributed approach based on a reusable, sparse design of synthetic circuits.

A small library of engineered cell types with restricted connections among them was generated, each cell responding to one/two inputs (Fig. 1a–c). The basic two-input and one-output engineered functions include the AND and the inverted IMPLIES (N-IMPLIES, Fig. 1d), which allow implementing any Boolean function. Moreover, some cells define one-input, one-output function (Fig. 1e). The output of each cell type is either a diffusible wiring molecule or the desired output. In contrast with previous works using synthetic consortia[18–22], we have not used cell-cell feedbacks. Instead, cells only respond to an external input and to a single diffusible molecule acting as a wire.

The computation is determined by: (1) the number of cells C involved, (2) the specific function implemented by each engineered cell and (3) the location of cells within the network (see Supplementary Information and Supplementary Fig. 1 for details). Crucially, we allow different engineered cells to produce the output signal, which is thus distributed. Moreover, each cell can be modulated by external inputs, which can either trigger the production of a signal or its inhibition.

The combinatorial nature of our approach is highlighted by calculating, for each C, the number of functions that can be implemented[19]. We have analysed all possible functions with two and three inputs versus C with our approach (see Supplementary Information and Supplementary Fig. 2) and found that most can be constructed using $C = 2–5$ different cells. For instance, in response to three inputs, just three cells results in more than 100 functions and exceed 200 using four cell types. The number of extracellular wires using this approach is significantly lower compared to other standard approximations (Supplementary Fig. 3 and Supplementary Information). With three inputs, over 100 different logical functions can be achieved with only two wires and almost all are obtained with just three to four wires (Supplementary Fig. 3).

As a proof of principle that distributed computation can be implemented *in vivo*, we created a library of engineered yeast cells. Each cell

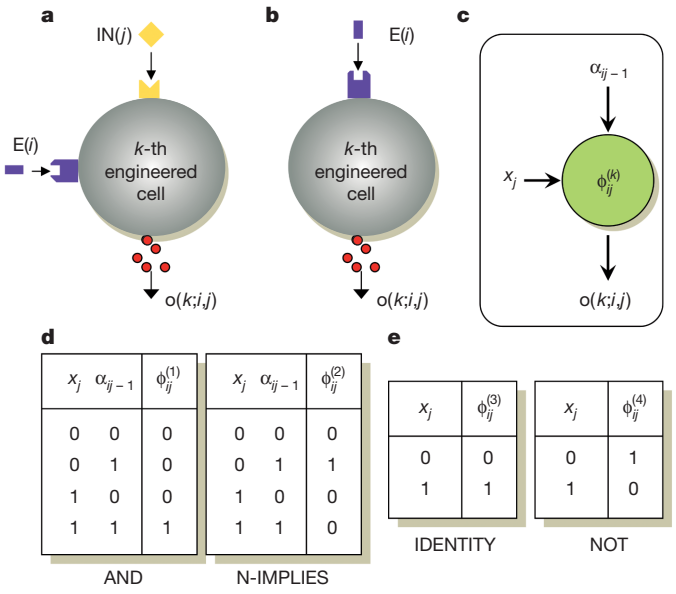

Figure 1 | Basic engineered cells (cell types). **a**, **b**, Cells can receive signals from other cells (IN) and external sources (E) (**a**) or just from external sources (**b**). Cells can also produce diffusive output molecules. **c**, Representation of the cell behaviour is summarized, where each cell c_{ij} responds to two different inputs; external input (x_j) and a signalling molecule (wire) from another cell (α_{ij-1}). The response of k-th cell type o(k;i,j) can be the production of a new wiring molecule (α_{ij}) or the final output. The k-th cells respond to the presence of signals through some Boolean function $\Phi^{(k)}ij$ with $k = \{1,2,3,4\}$ defining the resulting Boolean output. **d**, **e**, The four basic functions implemented in our study are displayed.

[1]Cell signaling unit, Departament de Ciències Experimentals i de la Salut, Universitat Pompeu Fabra (UPF), E-08003 Barcelona, Spain. [2]ICREA-Complex Systems Laboratory, Universitat Pompeu Fabra (UPF), E-08003 Barcelona, Spain. [3]Department of Cell and Molecular Biology/Microbiology, University of Gothenburg, Box 462, 40530 Gothenburg, Sweden. [4]Santa Fe Institute, Santa Fe, New Mexico 87501, USA. [5]Institut de Biologia Evolutiva, CSIC-UPF, Passeig Maritim de la Barceloneta, 37-49, 08003 Barcelona, Spain.
*These authors contributed equally to this work.

responds to an extracellular stimulus (for example, NaCl, doxycycline, galactose, oestradiol) and/or the presence of a wiring molecule (for example, yeast pheromone). The output of the cells was monitored as the expression of a reporter construct under the control of the *FUS1* promoter (for example, green fluorescent protein, GFP). See Supplementary Fig. 4 for relevant genotype and the logic function of each cell of the library is. The ability of cells to respond to external stimuli (inputs) was monitored by fluorescence in single cell (fluorescence-activated cell sorting, FACS) and normalized to the maximal number of cells able to produce output signal (see Fig. 2a and Supplementary Information). Each cell type has been characterized by its ability to respond to the corresponding stimuli (Fig. 2b and Supplementary Fig. 5).

We then implemented all standard 2-input logic functions by combining just a few engineered cell types. We initially designed a basic circuit with an AND logic (Fig. 3a) involving two cell types responding to two stimuli (NaCl and oestradiol) and using a pheromone (alpha factor) as a wiring molecule. The presence of NaCl stimulates Cell 1 to produce pheromone (IDENTITY) that is received by Cell 2. In addition, Cell 2 has the ability to sense another external input (oestradiol) and it is competent, via the production and activation of the Fus3 mitogen-activated protein kinase (MAPK), to produce the final output. Only in the presence of the two inputs the final outcome was produced (Fig. 3a). Similarly, a NOR gate was implemented using a different pair of cell types in which each cell responded to a particular stimulus (doxycycline and 6a, an inhibitor of Fus3as kinase) with yeast pheromone as a wiring. Only in the absence of both stimuli there was positive output (Fig. 3b).

Next, we designed two completely different circuits that involved the use of three independent engineered cell types, by reusing cells from our previous AND and NOR circuits. The first three-cell circuit was an OR logic gate in which the two inputs are NaCl and galactose. In this circuit, engineered Cell 1 and 5 are IDENTITY functions. They respond to the presence of NaCl (input 1) or galactose (input 2) to produce the wiring molecule that induces output production in Cell 6 (GFP). The presence of any input (galactose or NaCl) generated a positive output as it corresponds to an OR gate (Fig. 3c). Similarly, a NAND gate was designed using doxycycline and glucose as inputs. Cell 3 and Cell 5 display NOT logic. Both secreted pheromone in the absence of stimuli. Cell 6 responded to the presence of pheromone from either Cell 3 or Cell 5 inducing a fluorescent output. As expected, only the presence of both stimuli generates the output (Fig. 3d). This illustrates how to increase computational complexity at low cost. Other circuits can be easily built through reuse (Supplementary Fig. 6). Of note, the N-IMPLIES circuit can be implemented in a single cell (Supplementary Fig. 6a) or by combining cells with different logics (Supplementary Fig. 6b). Using other consortia, we obtained the AND, NOR, OR, NAND, XNOR and XOR gates. However, they can be

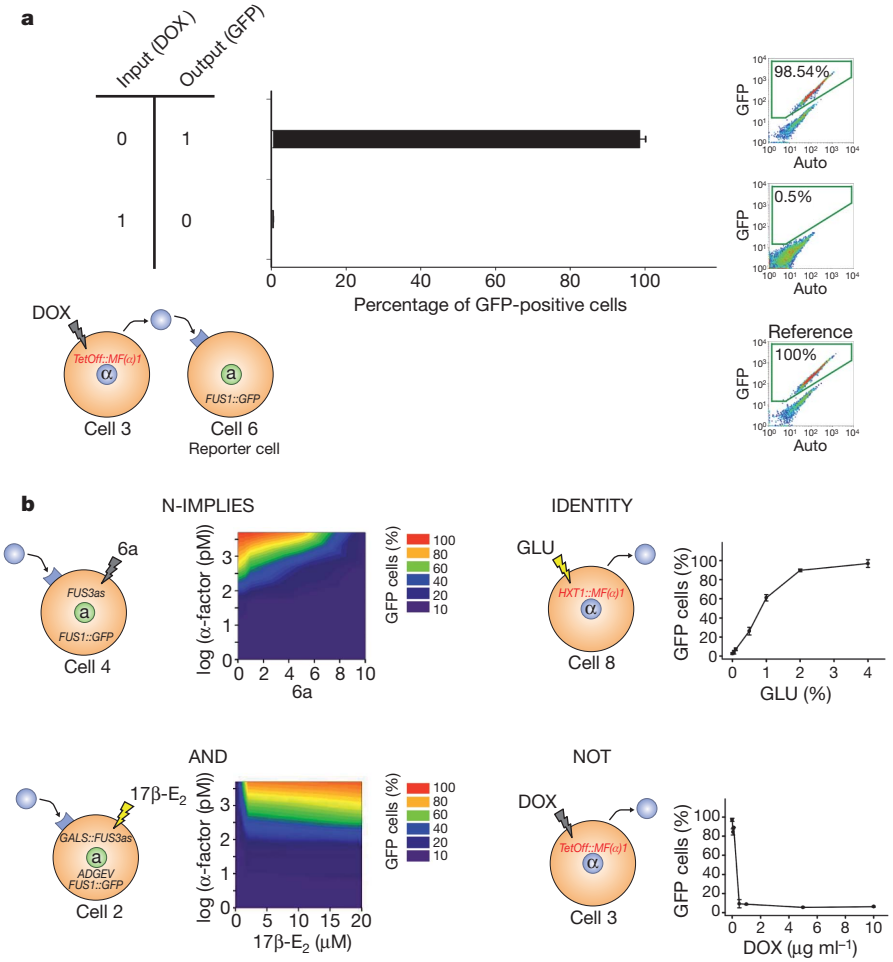

Figure 2 | *In vivo* analyses of engineered cells. **a**, Quantification of single cell computational output. Truth table and schematic representation of a cell with a NOT logic (see Supplementary Information for complete genotype). The NOT function is implemented in Cell 3, and the reporter cell (Cell 6) is used to quantify alpha factor production *in vivo*. Doxycycline (DOX) was added as indicated and cells were analysed by FACS. Data are expressed as the percentage of GFP-positive cells versus cells treated with pheromone. Results represent the mean ± s.d. of three independent experiments. **b**, Transfer functions of basic logic cells. Schematic representation of cells implementing N-IMPLIES, AND, IDENTITY and NOT functions. Indicated cells were treated with indicated input concentrations (2 inputs, left; 1 input, right). 17β-E_2, oestradiol; GLU, glucose.

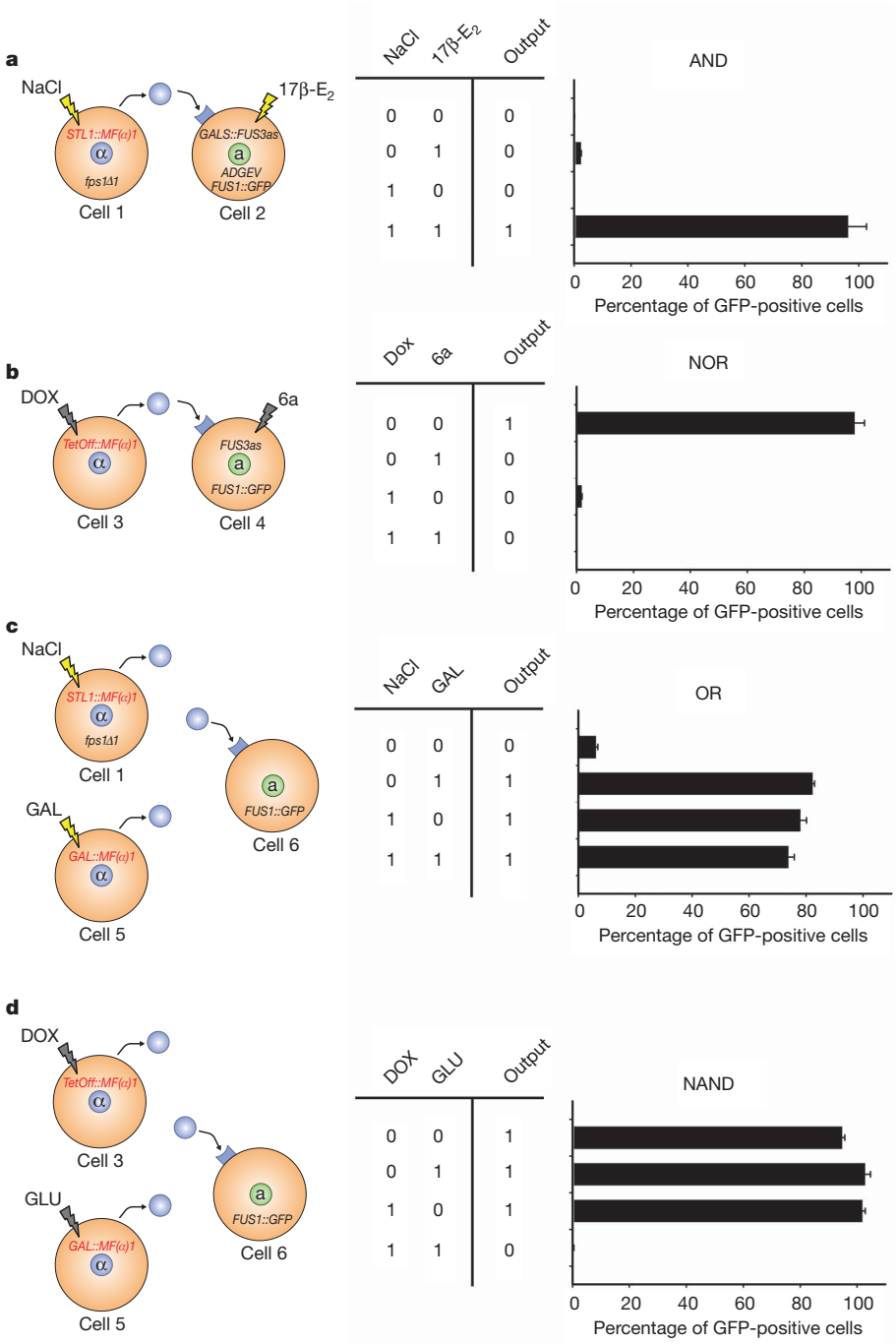

Figure 3 | Engineered cells to implement different logic gates *in vivo*. **a**, Truth table and schematic representation of cells in the AND circuit (see Supplementary Information for complete genotype). Cells were mixed proportionally and inputs (NaCl and oestradiol) were added at the same time. **b**, Panel ordered as in (**a**) following NOR logic. Indicated cells were treated using as inputs doxycycline and 6a. **c**, OR gate. Indicated cells were treated using as inputs 0.4 M NaCl and 2% galactose (GAL). **d**, NAND gate. Indicated strains were treated using as inputs doxycycline and 2% glucose. Data represent the mean and standard deviation of three independent experiments.

implemented using the same inputs for all of them (that is, doxycycline and glucose) (Supplementary Fig. 7a–f). This supports that our approach is adaptable and that multiple functions can be constructed from a small library of reusable cells.

We then analysed the long-term dynamical response of our AND circuit under changing inputs (Supplementary Fig. 7a). We have found that once the circuit is turned on, it can maintain maximal signal for periods beyond 9 h in the presence of stimuli (Supplementary Fig. 8a). Furthermore, once the system has been established, it responds equally well (at least eight generations) while the culture is maintained to a log phase (Supplementary Fig. 8b). We have also experimentally addressed network responsiveness to dynamic changes by means of a microfluidic device containing the cellular types of the AND circuit and have exposed it to changes in the input signals over time. The system was able to dynamically respond to re-stimulation after GFP inactivation (Supplementary Fig. 8c).

Our system can be selectively switched off and partially reprogrammed. For instance, the inhibition of the intracellular signal transduction in Cell 2 of the AND gate, blocks the positive outcome of the circuit (Supplementary Fig. 9a). More interestingly, when reprogramming is applied to complex multicellular circuits, different computations can be obtained with little effort. For instance, when a reprogramming

molecule (glucose) is added to the OR gate shown in Supplementary Fig. 9b it works as IDENTITY for NaCl (input 2). In this context, despite the fact that multicellular circuits might seem more complex, their easy reuse and combination actually makes them more appropriate in many situations.

Finally, we were able to engineer complex circuits by re-using our previous designs. One of them is the multiplexer MUX2to1 circuit that selects one of different input signals and forwards the selected input into a single output. This circuit, if designed in a single cell would be difficult to implement *in vivo* (see Supplementary Fig. 10a). However, using distributed computation, the circuit can be assembled from just three engineered cell types responding to three input signals and a single wiring molecule (Supplementary Fig. 10b). In addition, we also implemented a MUX2to1 circuit that contains four cell types but uses two independent wiring molecules (α-factor from *Saccharomyces cerevisiae* and the α-factor from *Candida albicans).* Cell 10 and Cell 13 respond to doxycycline and produce each one of the wiring molecules. Cell 12 responds to oestradiol and *S. cerevisiae* pheromone whereas Cell 15 responds to galactose and *C. albicans* pheromone. The final output (GFP) is generated by Cell 12 and Cell 15. Here, although the complexity of the circuit required a differential output to eight different input combinations, the *in vivo* results clearly showed that the computation of the three inputs yielded the expected response (Fig. 4a). A second complex circuit, the 1-bit adder with carry, was built by combining XOR and AND gates that respond to the same input (doxycycline and glucose) with two wiring molecules (α-factor from *S. cerevisiae* and *C. albicans*). In addition, output cells express different reporter proteins, a green reporter (adder) or a red reporter (carry) (*FUS1*::*GFP* or *FUS1*::*mCherry* respectively) allowing to detect the outcome of the carry and adder in the same culture. The system responds as an XOR gate (green columns) but presence of the two stimuli induces led to a 1-bit carrier (red columns) (Fig. 4b).

Possible applications as well as some caveats will need to be addressed in future work (such as scalability[25,26], strategies for reducing

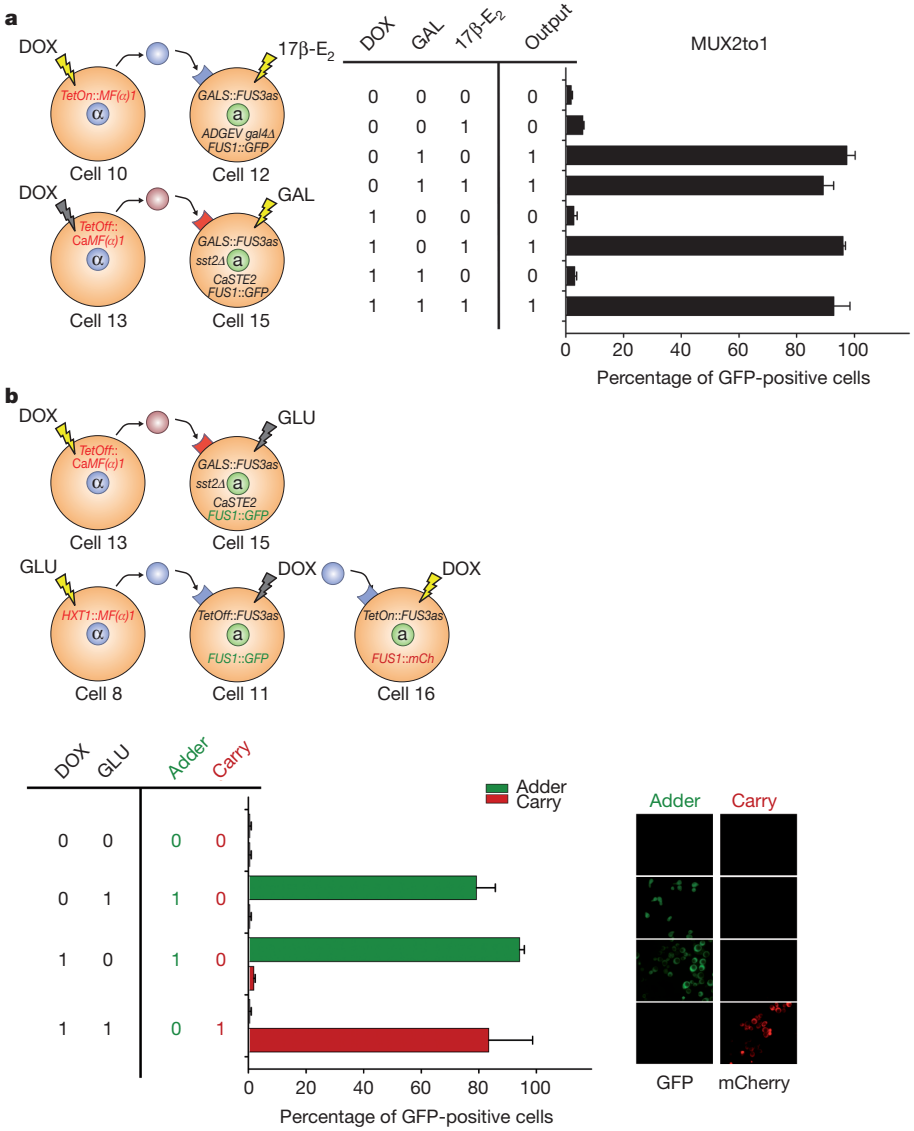

Figure 4 | Design and *in vivo* implementation of a multiplexer (MUX2to1) and 1-bit adder with carry. a, Truth table and schematic representation of the cells used in the MUX2to1. Indicated cells were treated using doxycycline (selector) and the inputs oestradiol and/or 2% galactose. Data are expressed as the percentage of GFP-positive cells using a sample treated with either *S. cerevisiae* or *C. albicans* alpha factor as a reference for Cell 12 or Cell 15, respectively. **b**, Truth table and schematic representation of cells used for 1-bit adder with carry. Four cells with two wiring systems that respond to glucose and doxycycline with an XOR logic were combined with an extra cell that respond to same stimuli but with an AND logic in which instead of GFP, mCherry was expressed as output. The final outcome was measured as in Fig. 3a. Green bars indicate the adder output (GFP) whereas red bars represent the carry bit (mCherry). GFP and mCherry images of cells are shown (right panels). Data represent the mean and standard deviation of three independent experiments.

potential crosstalk[3] and robustness to noise[27]). However, the results reported here show that distributed computation using consortia is a powerful strategy to build complex synthetic constructs, opening a new door to reusable, reprogrammable complex circuits (Supplementary Fig. 11).

METHODS SUMMARY

Complete list of engineered yeast strains and plasmids is described in Supplementary Information. Computational output detection was done in single cells by flow cytometry (FACScalibur, Becton Dickinson) and the dynamical inputs responses to different circuits were analysed in a microscopy based microfluidic platform. See Supplementary Information for details about experimental methods.

Received 18 February; accepted 18 November 2010.
Published online 8 December 2010.

1. Bashor, C. J. & Horwitz, A. A. Peisajovich , S. G. , & Lim, W. A. Rewiring cells: synthetic biology as a tool to interrogate the organizational principles of living. *Ann. Rev. Biophys.* **39,** 515–537 (2010).
2. Amos, M. *Cellular Computing* (Oxford University Press, 2004).
3. Andrianantoandro, E., Basu, S., Karig, D. & Weiss, R. Synthetic biology: new engineering rules for an emerging discipline. *Mol. Syst. Biol.* **2,** E1–E14 (2006).
4. Bray, D. Protein molecules as computational elements in living cells. *Nature* **376,** 307–312 (1995).
5. Hjelmfelt, A., Weinbeger, E. D. & Ross, J. Chemical implementation of neural networks and Turing machines. *Proc. Natl Acad. Sci. USA* **88,** 10983–10987 (1991).
6. McAdams, H. H. & Shapiro, L. Circuit simulation of genetic circuits. *Science* **269,** 650–656 (1995).
7. Arkin, A. & Ross, J. Computational functions in biochemical reaction networks. *Biophys. J.* **67,** 560–578 (1994).
8. Elowitz, M. B. & Leibler, S. A synthetic oscillatory network of transcriptional regulators. *Nature* **403,** 335–338 (2000).
9. Gardner, T. S., Cantor, C. R. & Collins, J. J. Construction of a genetic toggle switch in *Escherichia coli*. *Nature* **403,** 339–342 (2000).
10. Becskei, A. & Serrano, L. Engineering stability in gene networks by autoregulation. *Nature* **405,** 590–593 (2000).
11. Voigt, C. A. Genetic parts to program bacteria. *Curr. Opin. Biotechnol.* **17,** 548–557 (2006).
12. Kramer, B. P., Fischer, C. & Fussenegger, M. BioLogic gates enable logical transcription control in mammalian cells. *Biotechnol. Bioeng.* **87,** 478–484 (2004).
13. Win, M. N. & Smolke, C. D. Higher-order cellular information processing with synthetic RNA devices. *Science* **322,** 456–460 (2008).
14. Greber, D. & Fussenegger, M. An engineered mammalian band-pass network. *Nucleic Acids Res.* **38,** e174 (2010).
15. Weber, W. *et al.* A synthetic mammalian gene circuit reveals antituberculosis compounds. *Proc. Natl Acad. Sci. USA* **105,** 9994–9998 (2008).
16. Rinaudo, K. *et al.* A universal RNAi-based logic evaluator that operates in mammalian cells. *Nature Biotechnol.* **25,** 795–801 (2007).
17. Purnick, P. E. M. & Weiss, R. The second wave of synthetic biology: from modules to systems. *Nature Rev. Mol. Cell Biol.* **10,** 410–422 (2009).
18. Brenner, K., You, L. & Arnold, F. H. Engineering microbial consortia: a new frontier in synthetic biology. *Trends Biotechnol.* **26,** 483–489 (2008).
19. Brenner, K., Karig, D. K., Weiss, R. & Arnold, F. H. Engineered bidirectional communication mediates a consensus in a microbial biofilm consortium. *Proc. Natl Acad. Sci. USA* **104,** 17300–17304 (2007).
20. Danino, T., Mondragón-Palomino, O., Tsimring, L. & Hasty, J. A synchronized quorum of genetic clocks. *Nature* **463,** 326–330 (2010).
21. You, L., Cox, R. S. III, Weiss, R. & Arnold, F. H. Programmed population control by cell-cell communication and regulated killing. *Nature* **428,** 868–871 (2004).
22. Shou, W., Ram, S. & Vilar, J. M. G. Synthetic cooperation in engineered yeast populations. *Proc. Natl Acad. Sci. USA* **104,** 1877–1882 (2007).
23. Weber, W., Daoud-El Baba, M. & Fussenegger, M. Synthetic ecosystems based on airbone inter- and intrakingdom communication. *Proc. Natl Acad. Sci. USA* **104,** 10435–10440 (2007).
24. Balagaddé, F. K. *et al.* A synthetic *Escherichia coli* predator–prey ecosystem. *Mol. Syst. Biol.* **4,** 187 (2008).
25. Lucks, J. B., Qi, L., Whitaker, W. R. & Arkin, A. P. Toward scalable parts families for predictable design of biological circuits. *Curr. Opin. Microbiol.* **11,** 567–573 (2008).
26. Chin, J. W. Programming and engineering biological networks. *Curr. Opin. Struct. Biol.* **16,** 551–556 (2006).
27. Rao, C. V., Wolf, D. M. & Arkin, A. P. Control, exploitation and tolerance of intracellular noise. *Nature* **420,** 231–237 (2002).

Supplementary Information is linked to the online version of the paper at www.nature.com/nature.

Acknowledgements We thank L. Subirana and S. Ovejas for technical support and S. Pellet and M. Peter for their help in setting up the microfluidics platform, and K. Kuchler for the *FUS1*-mCherry construct. S.R. is recipient of a FPU fellowship (Spanish Government). This work was supported by grants from the James McDonnell Foundation to R.S., the MICINN (BIO2009-07762 and FIS2009-12365); Consolider Ingenio 2010 programme (grant CSD2007-0015), from the ESF (ERAS-CT-2003-980409) FP6 as part of a EURYI scheme award (www.esf.org/euryi) to F.P. and the CELLCOMPUT (FP6) project to F.P., R.S. and S.H., and FP7 UNICELLSYS grant (#201142) to F.P. and S.H., and The Santa Fe Institute to R.S.; F.P. and R.S. laboratories are also supported by the Fundación Marcelino Botín (FMB). F.P. is recipient of an ICREA Acadèmia (Generalitat de Catalunya).

Author Contributions All authors shared all the phases of the work. J.M. and R.S. developed the theoretical background for multicellular computing. Circuits were designed by S.R., J.M., N.C., E.N., F.P. and R.S.; S.R., N.C., K.F., J.K. and T.P. did the experimental designs. J.M., F.P., E.N., S.H. and R.S. wrote the paper.

Author Information Reprints and permissions information is available at www.nature.com/reprints. The authors declare no competing financial interests. Readers are welcome to comment on the online version of this article at www.nature.com/nature. Correspondence and requests for materials should be addressed to R.S. (ricard.sole@upf.edu) or F.P. (francesc.posas@upf.edu).