

Mechanical integrated circuit materials

<https://doi.org/10.1038/s41586-022-05004-5>

Received: 13 December 2021

Accepted: 21 June 2022

Published online: 24 August 2022

Charles El Helou¹, Benjamin Grossmann², Christopher E. Tabor², Philip R. Buskohl² & Ryan L. Harne^{1✉}

Recent developments in autonomous engineered matter have introduced the ability for intelligent materials to process environmental stimuli and functionally adapt^{1–4}. To formulate a foundation for such an engineered living material paradigm, researchers have introduced sensing^{5–11} and actuating^{12–16} functionalities in soft matter. Yet, information processing is the key functional element of autonomous engineered matter that has been recently explored through unconventional techniques with limited computing scalability^{17–20}. Here we uncover a relation between Boolean mathematics and kinematically reconfigurable electrical circuits to realize all combinational logic operations in soft, conductive mechanical materials. We establish an analytical framework that minimizes the canonical functions of combinational logic by the Quine–McCluskey method, and governs the mechanical design of reconfigurable integrated circuit switching networks in soft matter. The resulting mechanical integrated circuit materials perform higher-level arithmetic, number comparison, and decode binary data to visual representations. We exemplify two methods to automate the design on the basis of canonical Boolean functions and individual gate-switching assemblies. We also increase the computational density of the materials by a monolithic layer-by-layer design approach. As the framework established here leverages mathematics and kinematics for system design, the proposed approach of mechanical integrated circuit materials can be realized on any length scale and in a wide variety of physics.

Unconventional practices of information processing in engineered materials have recently been explored, including mechanical computing that abstracts digital bits according to mechanism and material configurations^{17,18,21–29}. For scalable computing capabilities, integrated circuits (ICs) are the conventional platform for computing digital electrical signals. Efforts are growing to embed mechanically strain-gated switching elements in ICs for the development of logic operations in soft matter^{19,20}. By combining mechanical computing bit abstraction with reconfigurable electrical networks, ref.³⁰ realized all of the universal logic gates in soft and conductive mechanical metamaterials. Yet, a method to develop combinational logic ICs in soft materials has yet to be established. Mechanical computing networks are currently limited by the type and quantity of inputs that an output signal can drive^{20,30}, and by the damping of the signal propagation in mechanical materials^{24,27}. A design approach is required whereby scalable, higher-level computing operations are cultivated in soft, autonomous matter.

Inspired by the mathematical principles of switching theory for ICs³¹, here we introduce a robust strategy, grounded in Boolean mathematics, to guide the design of all combinational logic operations in soft, conductive mechanical materials. The approach builds on the logic gate components fashioned by ref.³⁰. We create an IC-synthesis procedure whereby any arbitrary combination of logic gates used in a combinational logic process can be realized. This facilitates information decoding and advanced arithmetic operations such as n -bit addition, subtraction, multiplication and magnitude comparators. These materials are programmed through a design process that supports

automation through minimized sets of Boolean functions and multilayer monolithic fabrication techniques to increase computational density. We demonstrate reconfigurable IC materials that sense and process mechanical loading to compute higher-level n -bit arithmetic and decoding operations that could be used for intelligent response, actuation and communication.

The IC-synthesis strategy builds on the fundamental mechanical computing 1-bit-unit-cell material cross-section shown in Fig. 1a. When uniaxially compressed, a bifurcated kinematic collapse stage is reached with two possible compact configuration states that are controlled through left or right shear. A mechanics-based bit abstraction is exploited to represent the 1-bit mechanical shear input as ‘1’ for the anticlockwise (left shear) rotational state and ‘0’ for the clockwise (right shear) rotational state (Fig. 1a). A row of adjacent unit cells that are connected in parallel deform homogeneously with the same digital input owing to the mechanical disconnections between centre square tiles. To develop an n -bit material system with 2^n unique configuration states, n unit-cell rows must be serially connected. For instance, a 2-bit material is shown in Fig. 1a with two mechanical input rows, A_1 and A_2 .

Our embodiment of digital logic³⁰ depends on the discrete mechanical behaviour of a unit cell coupled with an electrical network to develop two kinematic 1-bit switching elements. A mechanical material buffer switch (Fig. 1b) connects at the compact state to output an electrical digital signal Q_b of 1 when an input of 1 is applied, and yields an output Q_b of 0 for an input of 0. The second switch is a NOT switch (Fig. 1c):

¹Department of Mechanical Engineering, The Pennsylvania State University, University Park, PA, USA. ²Materials and Manufacturing Directorate, Air Force Research Laboratory, Wright-Patterson AFB, OH, USA. ✉e-mail: ryanharne@psu.edu

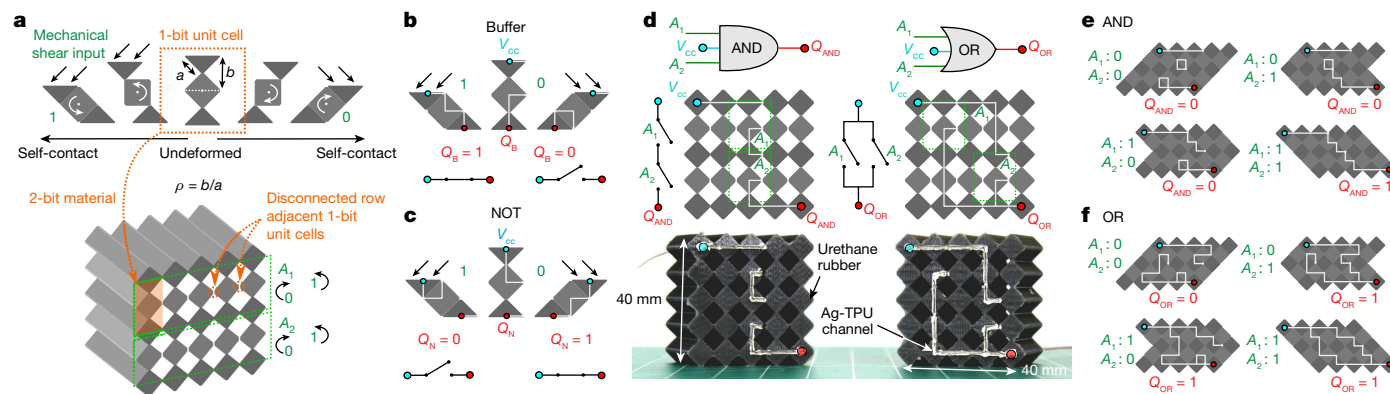


Fig. 1 | Bit abstraction through switching elements and gates. **a**, One-bit-unit-cell geometry with the 1 (anticlockwise) and 0 (clockwise) mechanical configurations to reach the corresponding self-contact states. The 2-bit material is also shown with two mechanical input rows connected in series, A_1 and A_2 . ρ refers to the relative size of the unit cell having dimensions a and b . **b,c**, Schematic of the buffer (**b**) and NOT (**c**) elementary switches on a 1-bit unit cell with respective conductive networks. Digital mechanical inputs are

an inverted buffer switch with output Q_N . With the 2-bit material platform in Fig. 1a and electrical networks consisting of the buffer and NOT switches, ref.³⁰ realized all the fundamental 2-bit logic gates, namely AND, NAND, OR, NOR, XOR and XNOR. Figure 1d shows the implementation of the AND and OR gates on a soft urethane rubber substrate with conductive surface channels containing silver thermoplastic polyurethane (Ag-TPU) composites (see Methods for fabrication details). Figure 1e,f validates the four possible configurations for the AND and OR gates, respectively.

Leveraging the fundamental buffer and NOT logic gates³⁰, Figure 2 illustrates the design process created through this work to program any higher-level combinational logic sequence in conductive mechanical networks. The process leverages the foundations of canonical Boolean functions and their algebraic combinations. A means to automate the mechanical-material design is given in Supplementary Information.

To exemplify the design process established here, we use the full adder as a model logic combination. The full adder in Fig. 2a includes three digital inputs A , B and C_{in} . This logic operation adds the 1-bit numbers A and B , and allows for the 1-bit C_{in} input as a carry bit that is conventionally passed along in a parallel binary adder circuit. The full adder yields two digital outputs Q_{Sum} and Q_{Cout} , which correspond to the first-bit and second-bit digits in the binary number ($Q_{Cout}Q_{Sum}$)₂, respectively. Each output is a separate logic path in the design strategy created here. Thus, distinct truth-table outputs are obtained for a specific combinational logic process (Fig. 2b). The truth tables are utilized to extract the canonical standard sum of product (SSoP) Boolean functions for each output. An SSoP is devised of minterms containing the product (&) of all the input Boolean terms that are added together (I). Each Boolean term in the function corresponds to a switch in the circuit. Minimizing the SSoP corresponds to a reduction in the quantity of switches and, as a result, a reduction of the size of the mechanical material. The canonical function minimization Quine–McCluskey (QM) algorithm³² is applied here to the SSoP to obtain the reduced Q_{Sum} and Q_{Cout} QMSoP functions shown in Fig. 2c. Further details on the Boolean functions are presented in Methods.

We then employ the symbolic representation of switching circuits presented by ref.³¹. In this approach, we construct the conductive network by utilizing the QMSoP functions composed using only the two elementary switches, buffer and NOT. As shown in Fig. 2c, each simplified minterm corresponds to a material column of three serially connected unit cells. An inverted term (') corresponds to a NOT switch, and

highlighted in green. Digital electrical output terminals are highlighted in red with labels Q , electrical paths are marked as white lines and the circuits are powered by the cyan-coloured V_{cc} terminal. **d**, Schematics (top) and experimental images (bottom) of the AND and OR logic gates designed on a 2-bit material structure with their respective switching circuitry. **e,f**, The AND (**e**) and OR (**f**) gates shown in the four possible mechanical configurations with the appropriate digital outputs for each.

an ordinary term corresponds to a buffer switch. For instance, the initial minterm in the Q_{Sum} function ($A' \& B' \& C_{in}$) contains two NOT switches on the A and B rows, and a buffer switch on the C_{in} row that are serially connected. Such a gate sequence is shown in the left column of the Q_{Sum} in Fig. 2c. By this method, each simplified minterm is represented in a distinct material column, where the buffer and NOT switches are used in exact agreement with the QMSoP functions for a given combinational logic process. Then the material columns are connected in parallel as shown in Fig. 2d to form the IC material system. Both outputs are powered by the same low voltage signal at the V_{cc} (cyan) terminal. The output terminals are separated to the Q_{Sum} and Q_{Cout} .

The mechanical IC material is then fabricated (Methods) with the full adder logic operation programmed on the surface as shown in Fig. 2e. This design is validated by the example addition arithmetic sequences shown in Fig. 2f. In practice, each layer undergoes shear and uniaxial compression to determine the digital input sequence (Methods). The column networks that electrically conduct and output a value of '1' upon compaction are highlighted in yellow or green in Fig. 2f. As the conductive Ag-TPU ink is confined by the substrate geometry, electrical connections are only allowed through the switches in the same column. All eight addition computations possible for the full adder are exemplified in Extended Data Fig. 1 by simulated and experimental results.

The extensibility of this implementation is grounded in the generality of canonical Boolean functions coupled with the kinematics of the platform. Implementing this IC formulation requires (1) independent bi-state self-contact and (2) a reconfigurable mechanical-electrical network realized in the mechanical-electrical material unit cell. Thus, the mathematical foundation established here is not unique to the unit-cell geometry in Fig. 1a and may be generalized to other material platforms with the two features (1) and (2). Extended Data Fig. 2 introduces two alternative 1-bit-unit-cell geometries with the two features that ensure similar computing capabilities albeit with distinct mechanical force inputs.

To illustrate the scalability of our formulation, three fundamental 2-bit arithmetic operators are constructed with the QMSoP functions. These operators are the 2-bit adder, 2-bit subtractor and 2-bit multiplier, as shown in Fig. 3. The operators are applied between the two 2-bit binary operands, A and B , that are mechanically applied on the 4-bit material in the row order A_1, B_1, A_2 and B_2 from top to bottom. See Extended Data Figs. 3–5 for the logic diagrams for each arithmetic operator, respectively.

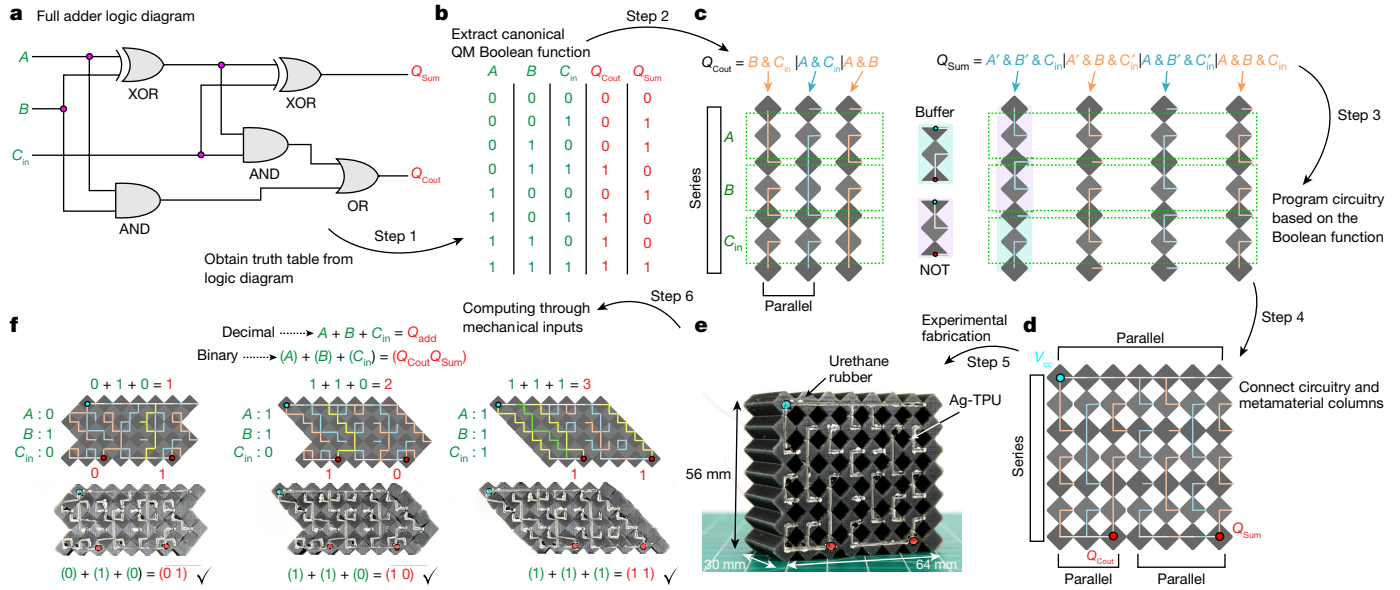


Fig. 2 | Combinational logic design synthesis. **a**, The full adder logic diagram with the mechanical inputs (A , B and C_{in}) shown in green, and the electrical outputs (Q_{Sum} and Q_{Cout}) highlighted in red. **b**, The truth table corresponding to the full adder operation. **c**, A schematic illustrating the relationship between the QMSoP Boolean function minterms, and the 3-bit material columns that contain either NOT or buffer connected in series. **d**, **e**, A schematic (**d**) and experimental image (**e**) of the full adder material design that is constructed by

connecting the material columns in parallel. **f**, Experimental digital output results of three possible configurations with their respective experimental images and schematics of the reconfigured network. Decimal inputs and outputs are shown at the top; binary inputs and outputs shown at the bottom. Connected paths that result in an output of 1 are highlighted in green or yellow. Electrical connections only occur through switches in the same column.

The 2-bit adder in Fig. 3a contains 32 electrical switches and 44 material unit cells. This is accounted for by 11 material columns each including 4 unit cells stacked vertically in series. By comparison, a 2-bit adder designed with the SSOP functions requires 60 switches and 64 unit cells. The Boolean function minimization through the QM algorithm allows for the same computing function with a significantly reduced quantity of switches. On the basis of the adder operation scalability analysis in Extended Data Fig. 6, the QMSoP reduces the material size by 31% in the 2-bit adder and by 92% in the 6-bit adder. For the specific material fabrications realized here, this corresponds to reduced mechanical energy stored per computing operation to achieve the compact state. Furthermore, a third method of designing combinational logic is created in this work, termed the substitution method

(Supplementary Information). As shown in Extended Data Fig. 7, the 2-bit adder designed from the substitution method has 24 switches and 36 unit cells, which is still fewer than the number of switches and unit cells through the QMSoP. Yet, the substitution method is dependent on a sequence of manipulations to the switching circuit and is tedious to scale and automate compared with the QMSoP that exploits the canonical functions.

The QMSoP 2-bit subtractor shown in Fig. 3b contains 32 switches and 44 unit cells, which is the same quantity as for the QMSoP 2-bit adder owing to De Morgan's theorem inversion of the switching network when a NOT gate is added to the logic diagram³⁰ (Extended Data Figs. 3 and 4). Finally, the 2-bit multiplier shown in Fig. 3c requires 24 switches and 32 unit cells. With increasing n -bits, the order of the computational

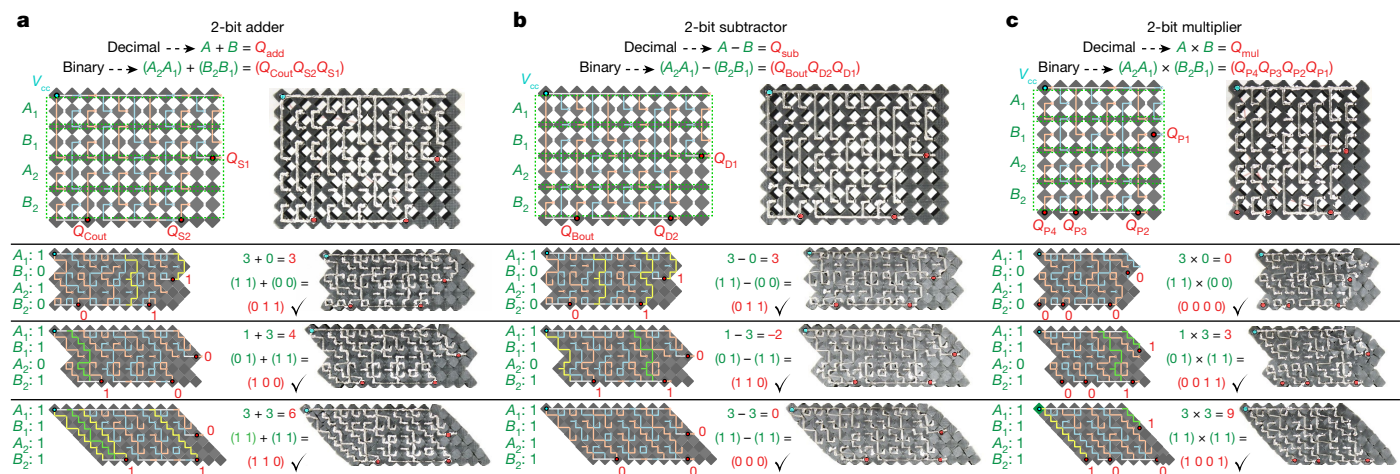


Fig. 3 | Two-bit arithmetic operators. **a–c**, A chart illustrating the schematic and experimental images of the 4-bit material and computing network design for a 2-bit adder (**a**), a 2-bit subtractor (**b**) and a 2-bit multiplier (**c**). The operations are computed between two operands A ($A_2 A_1$)₂ and B ($B_2 B_1$)₂

that are mechanically entered (green) through the three representative material configurations. The binary digital outputs (denoted by Q in red font) for each configuration are compared with the decimal value for validation. The Q_{Bout} binary digit in the subtractor output corresponds to a decimal value of -4.

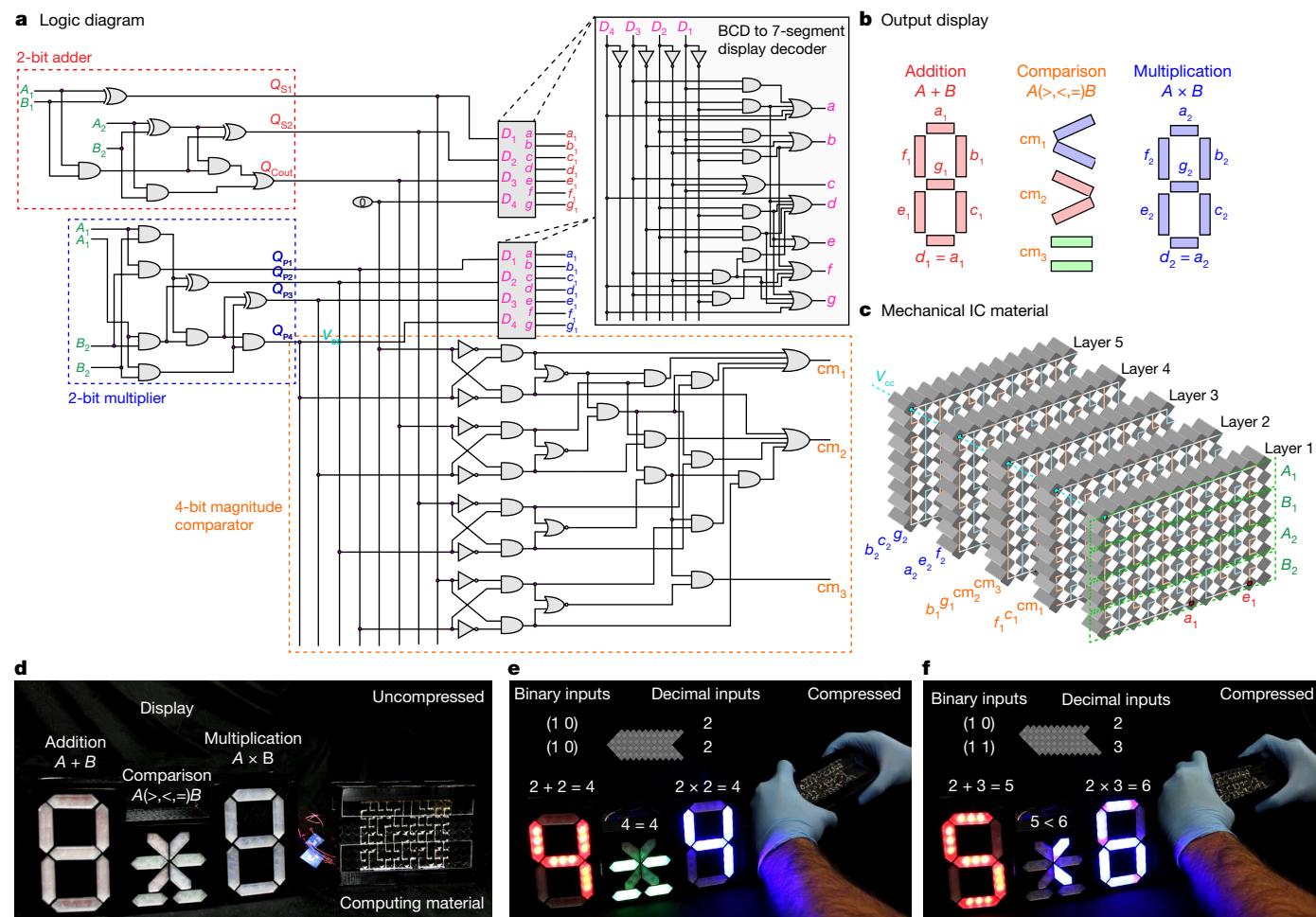


Fig. 4 | Demonstration of the sensing, computing and actuating functionalities of a soft mechanical IC material. **a**, The logic diagram for the combinational operation with 4 mechanical inputs (A_1, B_1, A_2, B_2) and 15 electrical outputs. The operation includes a 2-bit adder, 2-bit multiplier, 4-bit magnitude comparator and two binary-coded decimal (BCD) 7-segment display decoders. **b**, A schematic illustrating the multi-input display with the number segments and comparator symbols that correspond to the independent computing material outputs. The number corresponding to the sum ($A + B$) and product ($A \times B$) are

illuminated in red and blue, respectively. **c**, A schematic of the programmed mechanical IC material that is constructed with five 4-bit layer assembly. **d**, Experimental image of the display set-up directly connected to the uncompressed computing material. A segment or symbol on the display is lit (ON) when the corresponding output is 1, and vice versa. **e, f**, Application of digital inputs to the mechanical IC material to enter the $(0011)_2$ (**e**) and $(0111)_2$ (**f**) arithmetic calculations show that the appropriate inequalities are displayed.

complexity governs the IC network and material size as shown through the scalability analysis in Extended Data Fig. 6.

These three, arithmetic-specific IC materials are fabricated (Methods) and experimentally validated through the representative arithmetic calculations shown in Fig. 3. The 2-bit adder and 2-bit subtractor result in three digital outputs corresponding to the binary numbers $(Q_{\text{Cout}}Q_{\text{S2}}Q_{\text{S1}})_2$ and $(Q_{\text{Bout}}Q_{\text{D2}}Q_{\text{D1}})_2$, respectively. It is important to note that the two's complement binary representation is utilized for the 2-bit subtractor output, and thus the Q_{Bout} binary digit corresponds to a decimal value of -4 . The 2-bit multiplier results in four digital outputs that correspond to the binary number $(Q_{\text{P4}}Q_{\text{P3}}Q_{\text{P2}}Q_{\text{P1}})_2$. As shown in Fig. 3, the binary outputs for the three representative mathematical calculations agree with the decimal operations. See Extended Data Figs. 3–5 for all 16 possible mathematical computations able to be determined by the 2-bit adder, 2-bit subtractor and 2-bit multiplier, respectively.

We program an advanced mechanical IC material that communicates with a visual display (Fig. 4) to demonstrate the comprehensive sensing, information processing and response functionalities of mechanical IC materials. The corresponding logic diagram is shown in Fig. 4a, which includes 86 gates to govern 15 digital outputs. This operation contains four digital inputs: A_1, B_1, A_2 and B_2 . The 2-bit number A is simultaneously

added and multiplied with the 2-bit number B . The binary outputs are decoded to seven-segment number displays with a, b, c, d, e, f and g independent segments. As shown in Fig. 4b, the outputs corresponding to the addition and multiplication number are illuminated in red ($a_1, b_1, c_1, d_1, e_1, f_1$ and g_1) and blue ($a_2, b_2, c_2, d_2, e_2, f_2$ and g_2), respectively. Furthermore, the sum and product numbers are compared through a 4-bit magnitude comparator operation incorporated into the electrical switching network. Such operation results in three outputs, $cm_1 (<)$, $cm_2 (>)$ and $cm_3 (=)$ that govern each of the corresponding comparative symbols.

A five-layer mechanical IC material with 137 switches is fabricated (Fig. 4c). As the digital outputs in a logic operation are independent of each other, it is possible to separate computing material layers that are networked by stacked assembly. The outputs are distributed in the material based on the QMSoP minterms to optimize the quantity of layers and unit-cell columns for specific force applications. Increasing the quantity of stacked layers decreases the quantity of unit-cell columns per layer. The network design for each of the 12 column layers can be found in Extended Data Fig. 8. According to the binary inputs, the current flow from the V_{cc} (9 V) source connects the appropriate light-emitting diode (LED) number segments and comparator symbols

on the display. In Fig. 4d, the material initially has no digital input and correspondingly has no output. By manually applying a combination of uniaxial and shear traction, the $(0011)_2$ digital input sequence is applied to all the layers. This corresponds to the decimal operations of $(2 + 2)_{10}$ and $(2 \times 2)_{10}$. As shown in Fig. 4e, the material outputs the correct equality in the LED displays: $4 = 4$. As these materials have volatile memory, the bit information is lost when the digital inputs are removed, which automatically resets the material. Further digital input of $(0111)_2$ or corresponding decimal operations of $(2 + 3)_{10}$ and $(2 \times 3)_{10}$ is then applied as shown in Fig. 4f to output the appropriate inequality through the LED displays: $5 < 6$. See Supplementary Video 1 for demonstrations of the functionality of the mechanical IC material.

In Fig. 4, we show separated stacked layer assemblies that uniformly collapse while held in a material casing. Yet, monolithic material systems with multilayer logic operations that embed conductive networks through the material depth can reduce the quantity of unit cells and increase the computational density. Two fabrication techniques are demonstrated in Extended Data Fig. 9, which use casting and multi-material additive manufacturing methods to fabricate a 2-bit adder with 46% reduction in two-dimensional substrate area. See Methods for further fabrication details.

The formulation of combinational logic created here is grounded in Boolean mathematics³¹ and kinematically reconfigurable electrical networks³³. Consequently, similarly complex and scalable information processing could be achieved in other length scales and in other physics where such principles may be harnessed. Therefore, the sense of ‘touch’ realized here could be augmented with a sense of ‘sight’ through photo-responsive hydrogels⁵ in the material system design process, or with a sense of ‘hearing’ through discretized spectral signalling like that used in the cochlea³⁴. Certainly, the soft computing framework formulated here does not compete with the speed and complexity of conventional semi-conductor-based microprocessors. Yet, the information processing cultivated here is intrinsically scalable and sufficiently advanced to provision future engineered living materials with ample adaptability as they navigate the environment in pursuit of a programmed objective.

In this research, we introduce a robust strategy that exploits canonical Boolean functions and kinematically reconfigurable structures to program combinational logic into soft, mechanical IC materials. We demonstrate sensing, processing and responding functions in a monolithic material platform that thinks about digitized mechanical loading through higher-level combinational logic operations. Any arbitrary combinational logic process may be synthesized by this method and any degree of computational density may be achieved by the layer-by-layer fabrication approach. This foundation can be extended further by the formulation of analogue-to-digital conversion layers that help the materials to interface between the analogue mechanical loads natural to the environment and the requirement for digital sensory inputs. Nevertheless, as one embodiment of mechanical computing¹⁷, our approach offers advanced decision-making functionality and a roadmap for intelligence for researchers pursuing engineered living systems across a wide range of length scales.

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at <https://doi.org/10.1038/s41586-022-05004-5>.

- Kaspar, C., Ravoo, B. J., van der Wiel, W. G., Wegner, S. V. & Pernice, W. H. P. The rise of intelligent matter. *Nature* **594**, 345–355 (2021).
- McEvoy, M. A. & Correll, N. Materials that couple sensing, actuation, computation, and communication. *Science* **347**, 1261689 (2015).
- Wehner, M. et al. An integrated design and fabrication strategy for entirely soft, autonomous robots. *Nature* **536**, 451–455 (2016).
- Pishvar, M. & Harne, R. L. Foundations for soft, smart matter by active mechanical metamaterials. *Adv. Sci.* **7**, 2001384 (2020).
- Li, L., Scheiger, J. M. & Levkin, P. A. Design and applications of photoresponsive hydrogels. *Adv. Mater.* **31**, 1807333 (2019).
- Han, D., Lu, Z., Chester, S. A. & Lee, H. Micro 3D printing of a temperature-responsive hydrogel using projection micro-stereolithography. *Sci. Rep.* **8**, 1963 (2018).
- Sears, N. C., Berrigan, J. D., Buskohl, P. R. & Harne, R. L. Flexible hybrid electronic material systems with programmable strain sensing architectures. *Adv. Eng. Mater.* **20**, 1800499 (2018).
- Valentine, A. D. et al. Hybrid 3D printing of soft electronics. *Adv. Mater.* **29**, 1703817 (2017).
- Gong, J., Seow, O., Honnet, C., Forman, J. & Mueller, S. MetaSense: integrating sensing capabilities into mechanical metamaterial. In *34th Annual ACM Symposium on User Interface Software and Technology* 1063–1073 (2021).
- Barri, K. et al. Multifunctional meta-tribomaterial nanogenerators for energy harvesting and active sensing. *Nano Energy* **86**, 106074 (2021).
- Nick, Z. H., Tabor, C. E. & Harne, R. L. Liquid metal microchannels as digital sensors in mechanical metamaterials. *Extreme Mech. Lett.* **40**, 100871 (2020).
- Otake, M., Kagami, Y., Inaba, M. & Inoue, H. Motion design of a starfish-shaped gel robot made of electro-active polymer gel. *Rob. Autom. Syst.* **40**, 185–191 (2002).
- Ye, Z., Hou, P. & Chen, Z. 2D maneuverable robotic fish propelled by multiple ionic polymer-metal composite artificial fins. *Int. J. Intell. Robot. Appl.* **1**, 195–208 (2017).
- Yang, D. et al. Buckling pneumatic linear actuators inspired by muscle. *Adv. Mater. Technol.* **1**, 1600055 (2016).
- Wani, O. M., Zeng, H. & Priimagi, A. A light-driven artificial flytrap. *Nat. Commun.* **8**, 15546 (2017).
- Jin, H. et al. Soft and smart modular structures actuated by shape memory alloy (SMA) wires as tentacles of soft robots. *Smart Mater. Struct.* **25**, 085026 (2016).
- Yasuda, H. et al. Mechanical computing. *Nature* **598**, 39–48 (2021).
- Pal, A., Restrepo, V., Goswami, D. & Martinez, R. V. Exploiting mechanical instabilities in soft robotics: control, sensing, and actuation. *Adv. Mater.* **33**, 2006939 (2021).
- Xu, B., Chen, D. & Hayward, R. C. Mechanically gated electrical switches by creasing of patterned metal/elastomer bilayer films. *Adv. Mater.* **26**, 4381–4385 (2014).
- Chae, S. et al. Stretchable thin film mechanical-strain-gated switches and logic gate functions based on a soft tunneling barrier. *Adv. Mater.* **33**, 2104769 (2021).
- Yasuda, H., Tachi, T., Lee, M. & Yang, J. Origami-based tunable truss structures for non-volatile mechanical memory operation. *Nat. Commun.* **8**, 962 (2017).
- Trembl, B., Gillman, A., Buskohl, P. & Vaia, R. Origami mechanologic. *Proc. Natl Acad. Sci. USA* **115**, 6916–6921 (2018).
- Meng, Z. et al. Bistability-based foldable origami mechanical logic gates. *Extreme Mech. Lett.* **43**, 101180 (2021).
- Song, Y. et al. Additively manufacturable micro-mechanical logic gates. *Nat. Commun.* **10**, 882 (2019).
- Zanaty, M., Schneegans, H., Vardi, I. & Henein, S. Reconfigurable logic gates based on programmable multistable mechanisms. *J. Mech. Robot.* **12**, 021111 (2020).
- Ion, A., Wall, L., Kovacs, R. & Baudisch, P. Digital mechanical metamaterials. In *Proc. 2017 CHI Conference on Human Factors in Computing Systems* 977–988 (2017).
- Raney, J. R. et al. Stable propagation of mechanical signals in soft media using stored elastic energy. *Proc. Natl Acad. Sci. USA* **113**, 9722–9727 (2016).
- Preston, D. J. et al. Digital logic for soft devices. *Proc. Natl Acad. Sci. USA* **116**, 7750–7759 (2019).
- Jin, Y. et al. Materials tactile logic via innervated soft thermochromic elastomers. *Nat. Commun.* **10**, 4187 (2019).
- El Helou, C., Buskohl, P. R., Tabor, C. E. & Harne, R. L. Digital logic gates in soft, conductive mechanical metamaterials. *Nat. Commun.* **12**, 1633 (2021).
- Shannon, C. E. A symbolic analysis of relay and switching circuits. *Electr. Eng.* **57**, 713–723 (1938).
- McCluskey, E. J. Minimization of Boolean functions. *Bell Syst. Tech. J.* **35**, 1417–1444 (1956).
- Grima, J. N., Alderson, A. & Evans, K. E. Auxetic behavior from rotating rigid units. *Phys. Status Solidi B* **242**, 561–575 (2005).
- Ma, F., Wu, J. H., Huang, M., Fu, G. & Bai, C. Cochlear bionic acoustic metamaterials. *Appl. Phys. Lett.* **105**, 213702 (2014).
- Vuyk, P., Cui, S. & Harne, R. L. Illuminating origins of impact energy dissipation in mechanical metamaterials. *Adv. Eng. Mater.* **20**, 1700828 (2018).

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

© The Author(s), under exclusive licence to Springer Nature Limited 2022

Material substrate fabrication

The elastomeric material substrate is cast in a two-part mould. The mould components are designed in the computer-aided design (CAD) software SOLIDWORKS 2019 and three-dimensionally printed (FlashForge Creator Pro) with acrylonitrile butadiene styrene. When assembled together, the two-part mould realizes the negative of the material substrate shape with the appropriate networked channels for the conductive traces. The moulds used for the embedded network layer fabrication method include general gridded channels that may serve as a substrate platform for any 4-bit operation as shown in Extended Data Fig. 9b.

The channel cross-section for the conductive traces is 2.00-mm deep and 0.75-mm wide. The liquid urethane rubber (Smooth-On VytaFlex 60) material is mixed by hand for 3 min. Black dye (Smooth-On So-Strong) is added in the mixture (1% to 3% of the total urethane mass) to provide a visible colour contrast between the substrate and conductive path. The material is allowed to cure at room temperature for 16 h after being poured into the mould. The substrate is then demoulded and prepared for the conductive ink deposition. This moulding fabrication technique has been reported in previous studies^{30,35}.

Conductive ink fabrication and deposition

After the urethane rubber substrate is demoulded, enamel-coated copper wire (22 gauge) is passed to each terminal surface through a 2.00-mm-diameter channel in the substrate depth. The enamel is removed from the extremities of the wires as they are utilized to power the networks (V_{cc}) and measure the digital outputs (Q) at the terminals. The wires are secured by applying a small amount of silicone adhesive (DAP All-Purpose) at the back surface. Using a 3.0-cc dispensing syringe with a 27-gauge needle, the Ag-TPU ink is then deposited in the channels and allowed to cure around the copper wires for 24 h.

A similar process is required for each independent layer in the embedded network fabrication method. For the latter case, after the Ag-TPU is cured on each independent layer, the layers are moulded together by applying a thin uncured coat of urethane rubber between adjacent layers. An additional mould is utilized to align and secure the layers together during the 16-h curing period. After careful demoulding, the layers behave homogeneously as a monolithic material.

The composition of the applied conductive ink is 35% (volume % (v%)) Ag microflakes (Inframat Advanced Materials, 47MR-10F) and 65% (v%) TPU elastomer (BASF Elastollan Soft 35A). Appropriate quantities of Ag microflakes and *N*-methyl-2-pyrrolidone (NMP) solvent are first mixed in a glass vial, and sonicated (Branson M2800 Ultrasonic Cleaner) for 60 min. TPU granules are then added to the Ag-NMP sonicated mixture, and planetary mixed (KK 300SS Mazerustar) at 2,000 rpm for 2-min increments. The planetary mixing process is repeated three times with gentle hand-stirring in between to ensure that the walls of the vial do not collect aggregates. The Ag-TPU is ready for deposition after the mixture has had 48 h for the NMP to evaporate at room temperature. This Ag-TPU fabrication procedure can be found in previous studies^{7,8}.

Sample fabrication using additive manufacturing technique

To fabricate the 2-bit adder with the embedded networked layers, an additive manufacturing technique is explored that allows for the printing of the flexible substrate and conductive network simultaneously. The material and embedded network are designed as separate bodies in the CAD software SOLIDWORKS 2019. The conductive terminals extend to the back surface of the material to allow for power and output measurements. A dual extrusion fused deposition modelling three-dimensional printer (FlashForge Creator Pro 2) is utilized to print the substrate with non-conductive flexible TPU filament (NinjaTek Cheetah TPU) and the network with flexible conductive-carbon-based

TPU composite filament (NinjaTek EEL TPU). After the printing process is complete, the material is ready for computation of mechanical input signals.

Experimental digital state characterization

The arithmetic operations used in the formulation of conductive mechanical IC materials such as the full adder, 2-bit adder, 2-bit subtractor, 2-bit multiplier and multilayer 86-gate operation are experimentally examined to validate the digital outputs with the corresponding truth tables. Each of the soft material systems is compressed to reach a specific compact state. A combination of shear perturbations during uniaxial compression controls the mechanical input or the configuration state that the material enters. This loading technique is shown in greater detail in Supplementary Video 1. Through this process, all the configurations are examined, and the digital outputs are measured at each state. A 5-V power supply is connected to the V_{cc} terminal through the enamel wires that power the conductive network, yet any voltage supply may be connected depending on the actuator. For instance, a 9-V supply is utilized to power the LED display in Fig 4 and Supplementary Video 1. Each of the voltage outputs Q are measured relative to a common ground utilizing a voltmeter (AstroAI DM6000AR). A voltage threshold near the V_{cc} of 5 V is considered a digital output of 1, and a voltage reading of 0 V is considered a digital output of 0 (ref.³⁰).

QMSoP functions

This section describes the method utilized to obtain the QMSoP functions. The calculations here are exemplified on the full adder operation with two digital outputs Q_{Sum} and Q_{Cout} . Yet, such techniques are automated in this research and may be applied to directly obtain the QMSoP for all combinational logic operations. A canonical SSoP function is initially extracted from the truth tables for each digital output. An SSoP is devised of minterms corresponding to each bit output in the truth table. Each minterm contains the product (&) of all the input Boolean terms. If the input is '0' in that specific configuration, the inverted term (') is included in the product, and vice versa. All the minterms are added together (|) to form the SSoP form of the Boolean function. The SSoP functions for the Q_{Sum} and Q_{Cout} are described by equations (1) and (2), respectively.

$$Q_{Sum}(A, B, C_{in}) = A' \& B' \& C_{in} | A' \& B \& C'_{in} | A \& B' \& C'_{in} | A \& B \& C_{in} \quad (1)$$

$$Q_{Cout}(A, B, C_{in}) = A' \& B \& C_{in} | A \& B' \& C_{in} | A \& B \& C'_{in} | A \& B \& C_{in} \quad (2)$$

As shown in equations (1) and (2), both Q_{Sum} and Q_{Cout} in the SSoP contain four minterms with A , B and C_{in} . The canonical function minimization QM algorithm³² is applied here to the SSoP. The modified Q_{Sum} and Q_{Cout} QMSoP functions are demonstrated in equations (3) and (4), respectively.

$$Q_{Sum}(A, B, C_{in}) = A' \& B' \& C_{in} | A' \& B \& C'_{in} | A \& B' \& C'_{in} | A \& B \& C_{in} \quad (3)$$

$$Q_{Cout}(A, B, C_{in}) = B \& C_{in} | A \& C_{in} | A \& B \quad (4)$$

Using the QM algorithm, the Q_{Sum} in equation (1) is already in the optimized canonical form. However, the Q_{Cout} expression reduces significantly as shown by comparing equations (2) and (4).

Data availability

All data are available in the main text or the Supplementary Information and are available from the corresponding author upon reasonable request.

Code availability

All code is available in the main text or the Supplementary Information and is available from the corresponding author upon reasonable request.

Acknowledgements This research is supported in part by a US Air Force Research Lab (AFRL) Summer Faculty Fellowship, in part by the Air Force Office of Scientific Research, in part by the National Science Foundation (NSF) Faculty Early Career Development Award (number 2054970), and in part by funds from the Department of Mechanical Engineering at Penn State University.

Author contributions C.E.H. and R.L.H. designed the research. C.E.H. and B.G. performed the research. C.E.H., B.G., C.E.T., P.R.B. and R.L.H. analysed the data. C.E.H., B.G., C.E.T., P.R.B. and R.L.H. wrote the paper.

Competing interests The authors declare no competing interests.

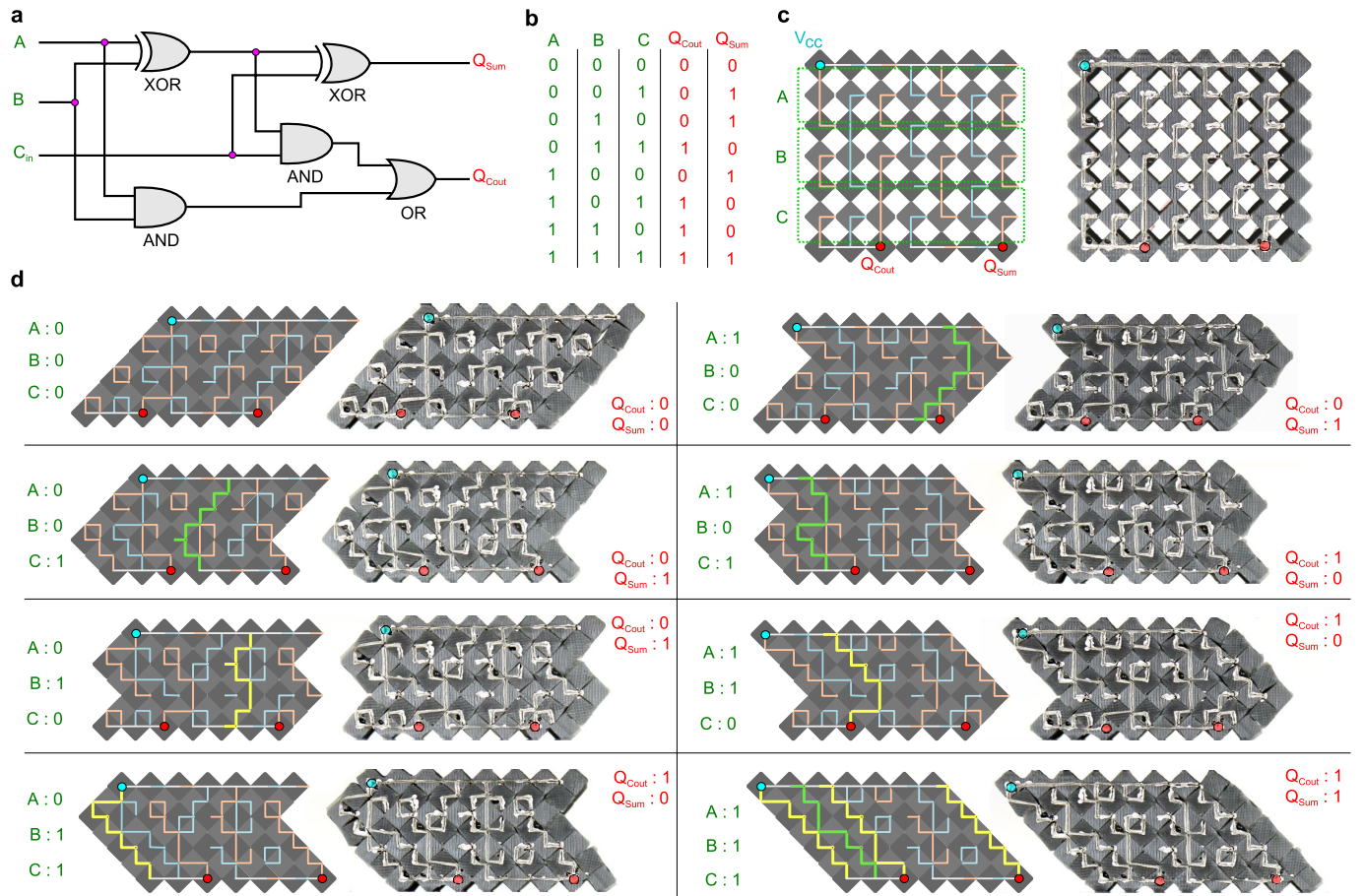
Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41586-022-05004-5>.

Correspondence and requests for materials should be addressed to Ryan L. Harne.

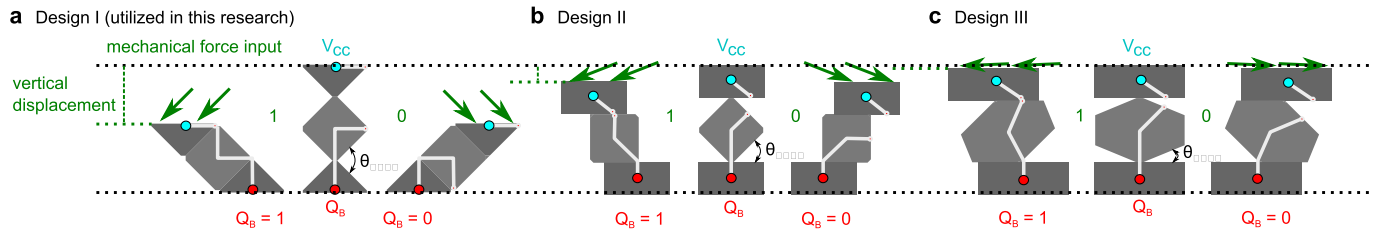
Peer review information *Nature* thanks Soosang Chae and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at <http://www.nature.com/reprints>.



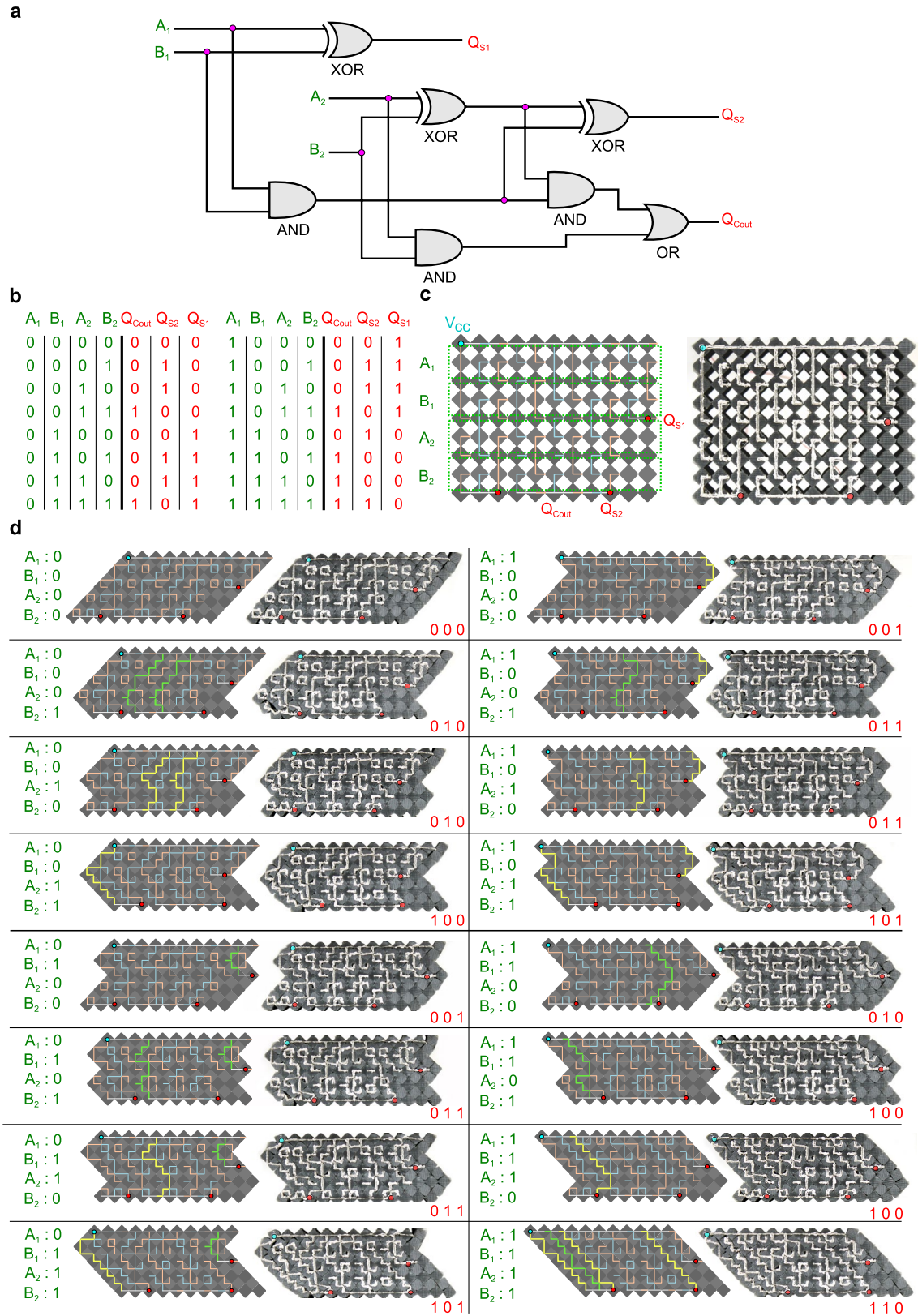
Extended Data Fig. 1 | Full adder digital states. (a) Logic diagram of the full adder with its corresponding (b) truth table. (c) A schematic (left) and experimental image (right) of the material and conductive network for the full adder operation as determined from the design process. (d) A schematic (left)

and experimental image (right) of the 8 possible configuration states with the binary inputs $(ABC)_2$ shown in green and output $(Q_{Cout} Q_{Sum})_2$ in red. Connected networks for each output are bolded and highlighted in green or yellow.



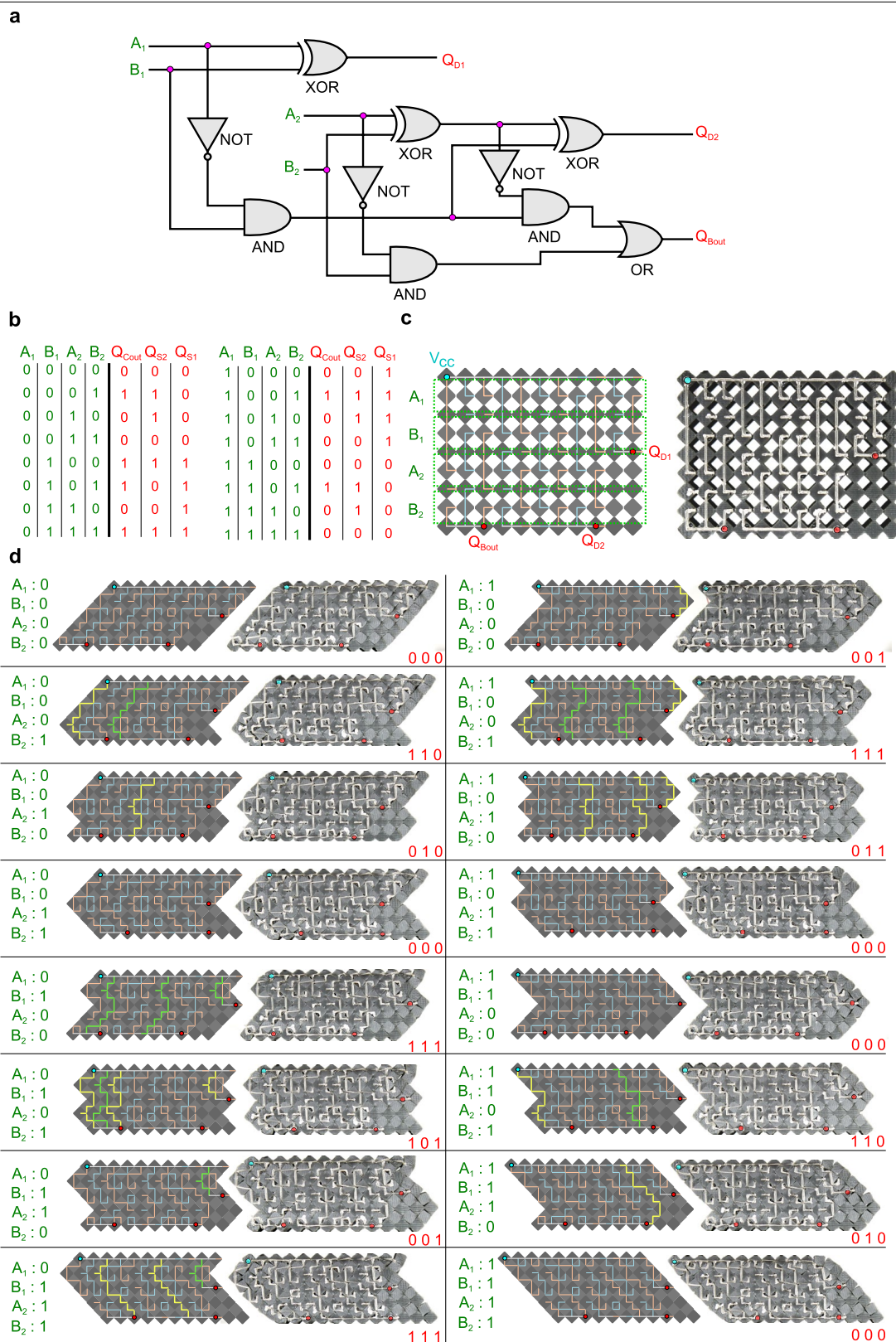
Extended Data Fig. 2 | Analysis of unit cell designs. Three 1-bit unit cell designs with a Buffer gate conductive network are explored to tailor the mechanical uniaxial and shear force for the digital bit input. Schematics of the three unique designs are illustrated here: (a) Design I, (b) Design II and (c) Design III. The designs demonstrate independent bi-state self-contact and reconfigurable mechanical-electrical networks. The contact angle design

parameter θ_{cont} and the centre shape geometry may be cultivated to design unit cells that require shear-dominated forces and low vertical displacements, such as Design III. The Boolean mathematical framework introduced in this research may be applied to various n-bit material systems that are sensitive to broad force environments.



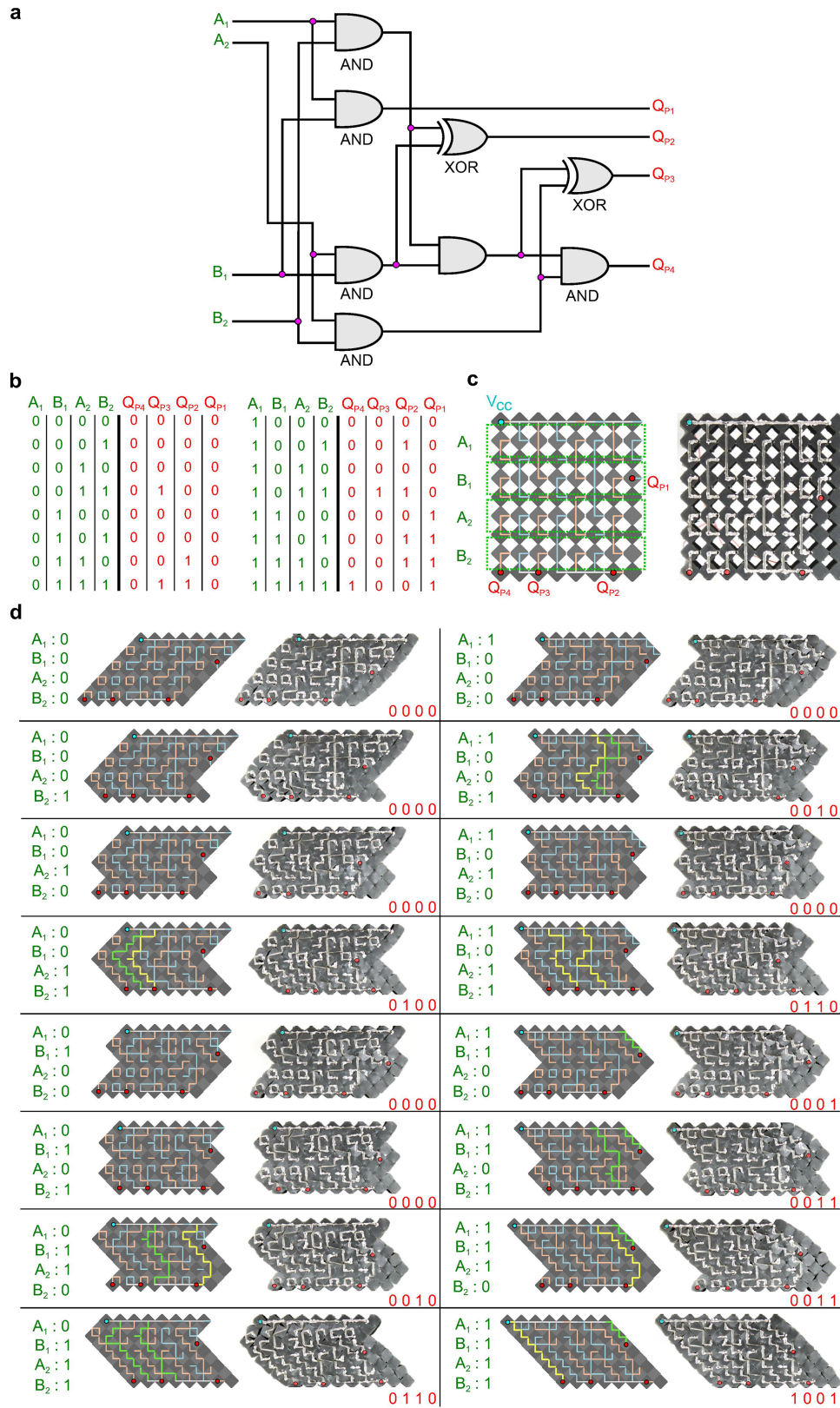
Extended Data Fig. 3 | 2-bit adder digital states. (a) Logic diagram of the 2-bit adder with its corresponding (b) truth table. (c) A schematic (left) and experimental image (right) of the material and conductive network for the 2-bit adder operation as determined from the design process. (d) A schematic (left)

and experimental image (right) of the 16 possible configuration states with the binary inputs $(A_1 B_1 A_2 B_2)_2$ shown in green and output $(Q_{Cout} Q_{S2} Q_{S1})_2$ in red. Connected networks for each output are bolded and highlighted in green or yellow.



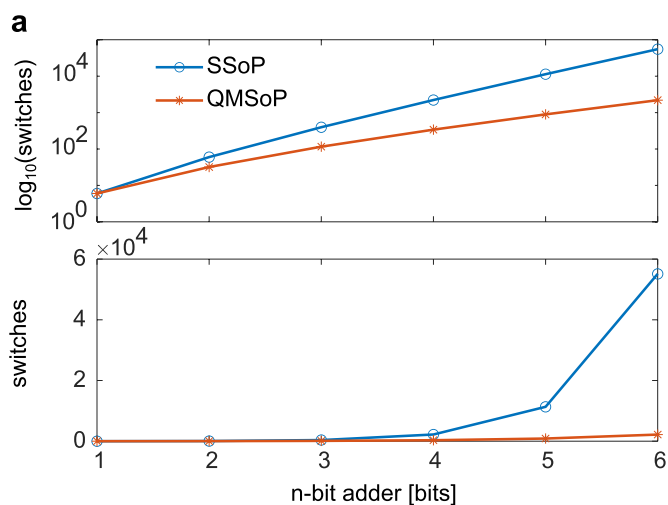
Extended Data Fig. 4 | 2-bit subtractor digital states. (a) Logic diagram of the 2-bit subtractor with its corresponding (b) truth table. (c) A schematic (left) and experimental image (right) of the material and conductive network for the 2-bit subtractor operation as determined from the design process. (d) A schematic

(left) and experimental image (right) of the 16 possible configuration states with the binary inputs ($A_1 B_1 A_2 B_2$)₂ shown in green and output ($Q_{Bout} Q_{D2} Q_{D1}$)₂ in red. Connected networks for each output are bolded and highlighted in green or yellow.

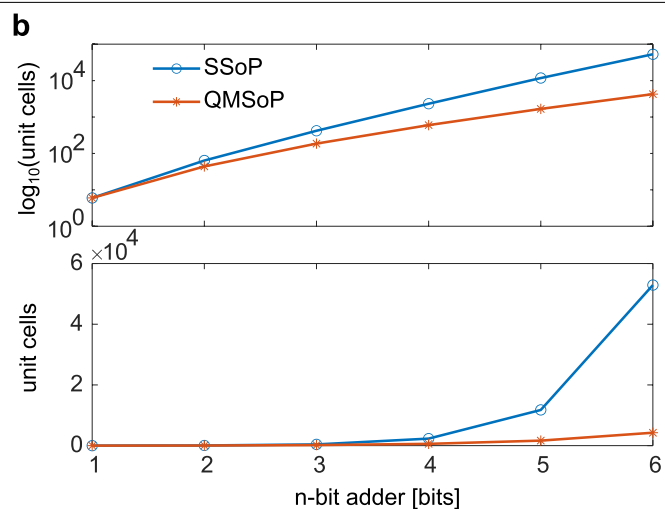


Extended Data Fig. 5 | 2-bit multiplier digital states. (a) Logic diagram of the 2-bit multiplier with its corresponding (b) truth table. (c) A schematic (left) and experimental image (right) of the material and conductive network for the 2-bit multiplier operation as determined from the design process. (d) A schematic

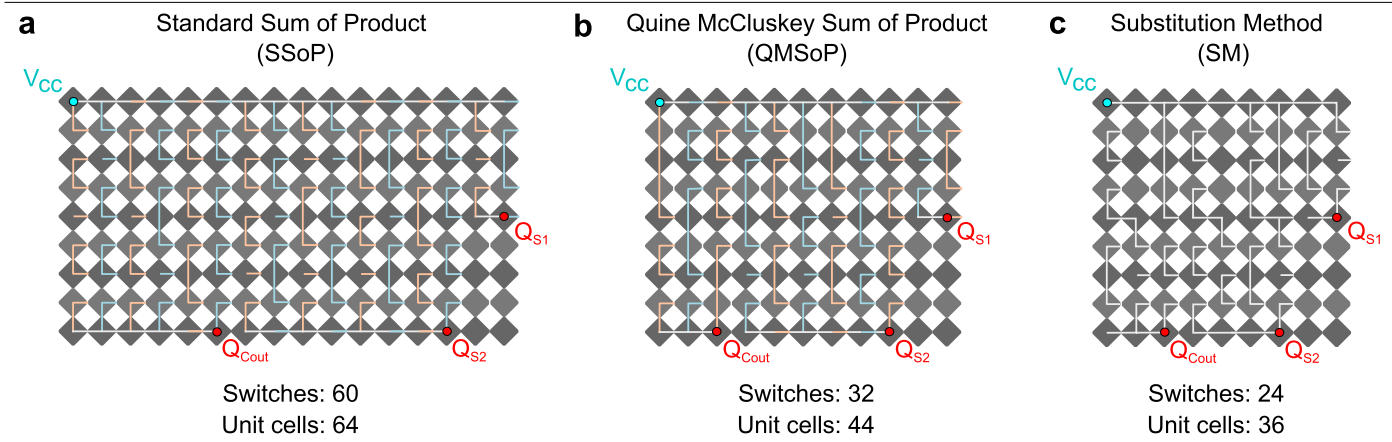
(left) and experimental image (right) of the 16 possible configuration states with the binary inputs ($A_1 B_1 A_2 B_2$) shown in green and output ($Q_{p4} Q_{p3} Q_{p2} Q_{p1}$) in red. Connected networks for each output are bolded and highlighted in green or yellow.



Extended Data Fig. 6 | Analysis of n-bit adder scalability. We utilize the automated design tool to design 1- to 6- bit adders to understand the material scalability with increasing n-bits in the adder operation. Two plots illustrating the relationship between the quantity of bits in an n-bit adder operation and the quantity of (a) switches and (b) material unit cells for the Standard Sum of

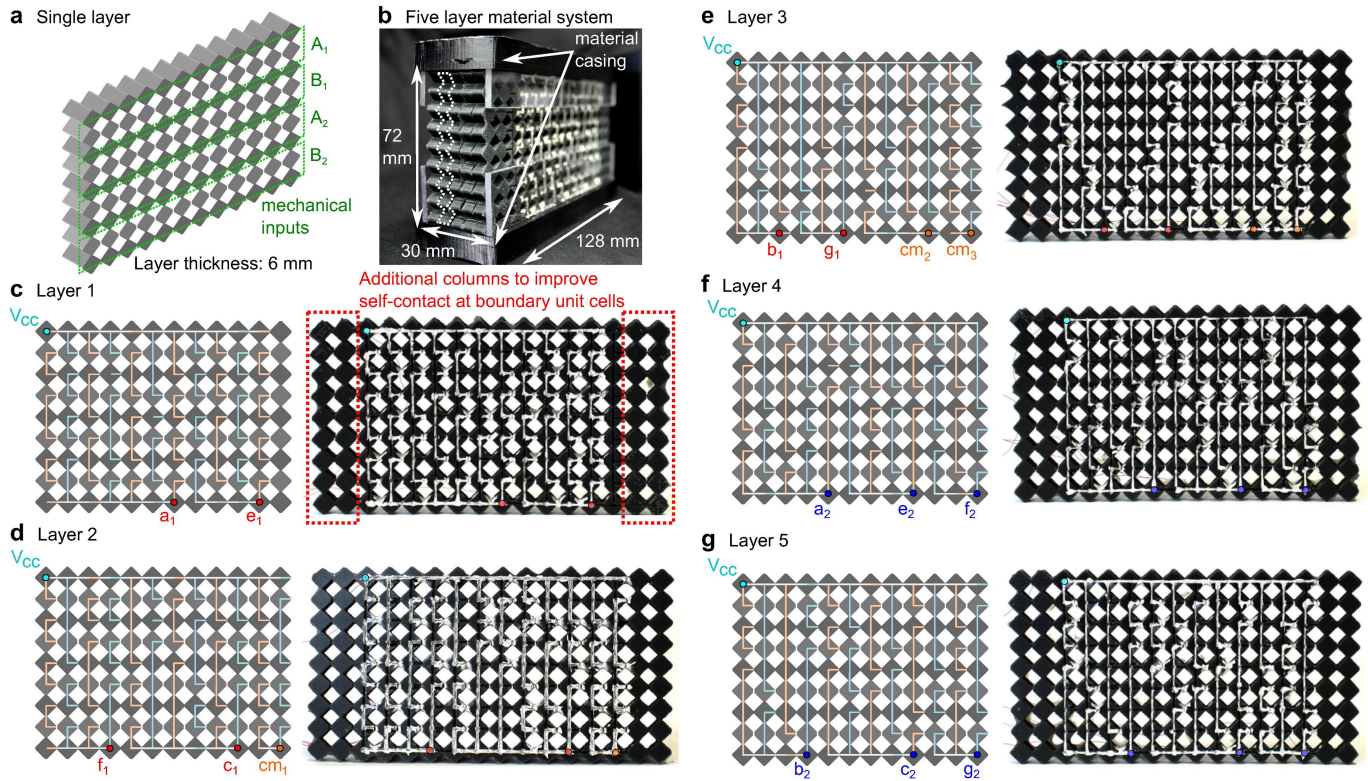


Product (SSoP) and the Quine–McCluskey Sum of Product (QMSoP) design methods. The top plots illustrate a $\log_{10}$ scale and the bottom plots illustrate a linear scale for the quantity of switches and unit cells. The QMSoP reduces the material size by 31% in the 2-bit adder and by 92% in the 6-bit adder.



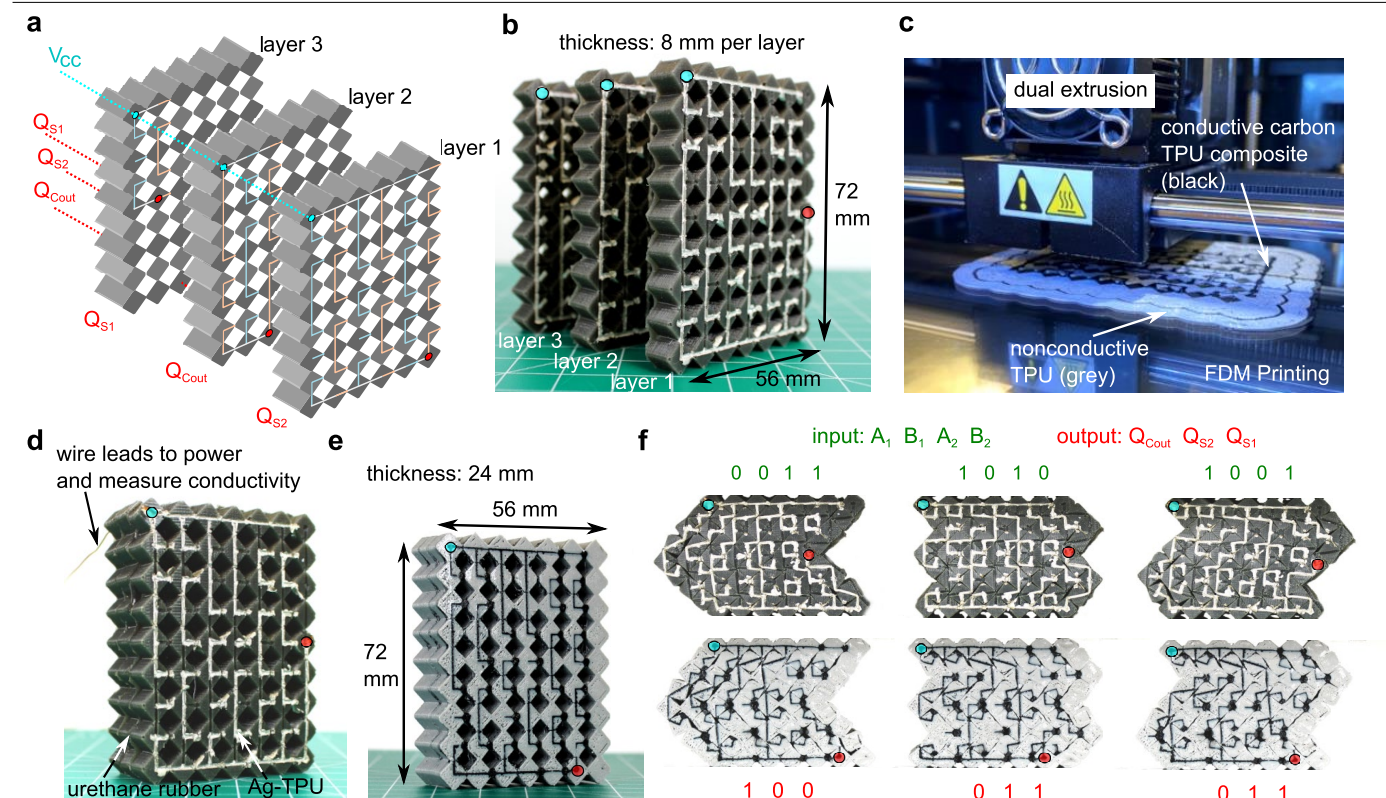
Extended Data Fig. 7 | Comparison of material design methods. Three methods are explored in this research that are capable of combinational logic network programming on soft material substrates. Schematics of the material and conductive network for the 2-bit adder by utilizing (a) the Standard Sum of Product (SSoP), (b) the Quine–McCluskey Sum of Product

(QMSoP) and (c) the Substitution Method (SM). Compared to the SSoP, the QMSoP 2-bit adder significantly reduces both the quantity of switches and unit cells to 32 and 44, respectively. The SM further reduces the 2-bit adder to 24 switches and 36 unit cells.



Extended Data Fig. 8 | Multilayer integrated circuit material. A five-layer material system is utilized for the 86-gate combinational logic operation. (a) A schematic of a single layer 4-bit mechanical platform. (b) An experimental image illustrating the five-layer material system in the casing with respective dimensions. Five layers are chosen to distribute the 15 total digital outputs. A schematic (left) and experimental image (right) of the material and conductive

network for layer (c) 1, (d) 2, (e) 3, (f) 4 and (g) 5 as determined from the design process. Each layer is only 12 conductive columns wide, which permits more straightforward digital actuation based on the urethane rubber material chosen here. In experiment, 16 column layers are utilized. The outer 2 columns are added without conductive traces on each extremity to improve self-contact at the boundary unit cells.



Extended Data Fig. 9 | Embedded layers for computing-dense mechanical integrated circuit material. (a) A schematic of a 2-bit adder with three stacked layers that contain the Q_{S2} , Q_{Cout} , and Q_{S1} on layers 1, 2 and 3, respectively. All layers are powered by the same V_{cc} terminal (cyan). (b) Experimental image showing the three layers made with cast urethane rubber substrates and Ag-TPU networks. (c) Image of commercial dual extrusion printer as it

simultaneously prints the substrate and conductive network of a 2-bit adder by the layer-by-layer fabrication method. Photos of 2-bit adders as fabricated using the (d) casting and (e) additive manufacturing techniques. (f) Experimental results showing three digital input and output combinations for the cast (top) and additive manufactured (bottom) samples.

Supplementary information

Mechanical integrated circuit materials

In the format provided by the
authors and unedited

Supplementary Information for Title: Mechanical Integrated Circuit Materials

Authors: Charles El Helou¹, Benjamin Grossmann², Christopher E. Tabor², Philip R. Buskohl², and Ryan L. Harne^{1*}

1. Department of Mechanical Engineering, The Pennsylvania State University, University Park, PA 16802 USA

2. Materials and Manufacturing Directorate, Air Force Research Laboratory, Wright-Patterson AFB, OH 45433 USA

** Corresponding author: ryanharne@psu.edu*

1 Substitution method of network topology formulation

Since the class of soft, conductive materials introduced in this research accept mechanical inputs and output electrical digital signals, the conventional serial assembly of logic gates is not feasible. Instead, the canonical Boolean functions allow for such assemblies by way of establishing the relationship between the truth tables of a logic gate assembly and a network of binary switches.

Here, an alternative method of formulating the conductive network topology for any arbitrary combinational logic sequence is introduced. The substitution method (SM) allows for the construction of a combinational logic gate assembly using a visual design approach.

For a Boolean function $f : \{0,1\}^m \rightarrow \{0,1\}$ with inputs $A_1, \dots, A_m$, a circuit with (1 input and 1 output) is called an f -switch if it is conductive if and only if $f(A_1, \dots, A_m) = 1$. For $A \in \{0,1\}$, let A' denote the negation of A , as shown in Equation 13.

$$A' = \begin{cases} 0 & A = 1 \\ 1 & A = 0 \end{cases} \quad (13)$$

For any $f : \{0,1\}^m \rightarrow \{0,1\}$, f' denotes the function defined by $f'(X) = f(X)'$.

All material network designs may be represented as circuits with switches that depend on the mechanical inputs to the system. In particular, each mechanical circuit with inputs $A_1, \dots, A_m$ may be represented as a

circuit built from A_i -switches and A_i' -switches. The overall behavior of the circuit is that of an f -switch for some $f: \{0,1\}^m \rightarrow \{0,1\}$. All the fundamental gates such as AND, NAND, OR, NOR, XOR, and XNOR are shown with the corresponding switching circuit in Figure S1(a) to (f), respectively [1]. Each gate may also be represented as a single f -switch.

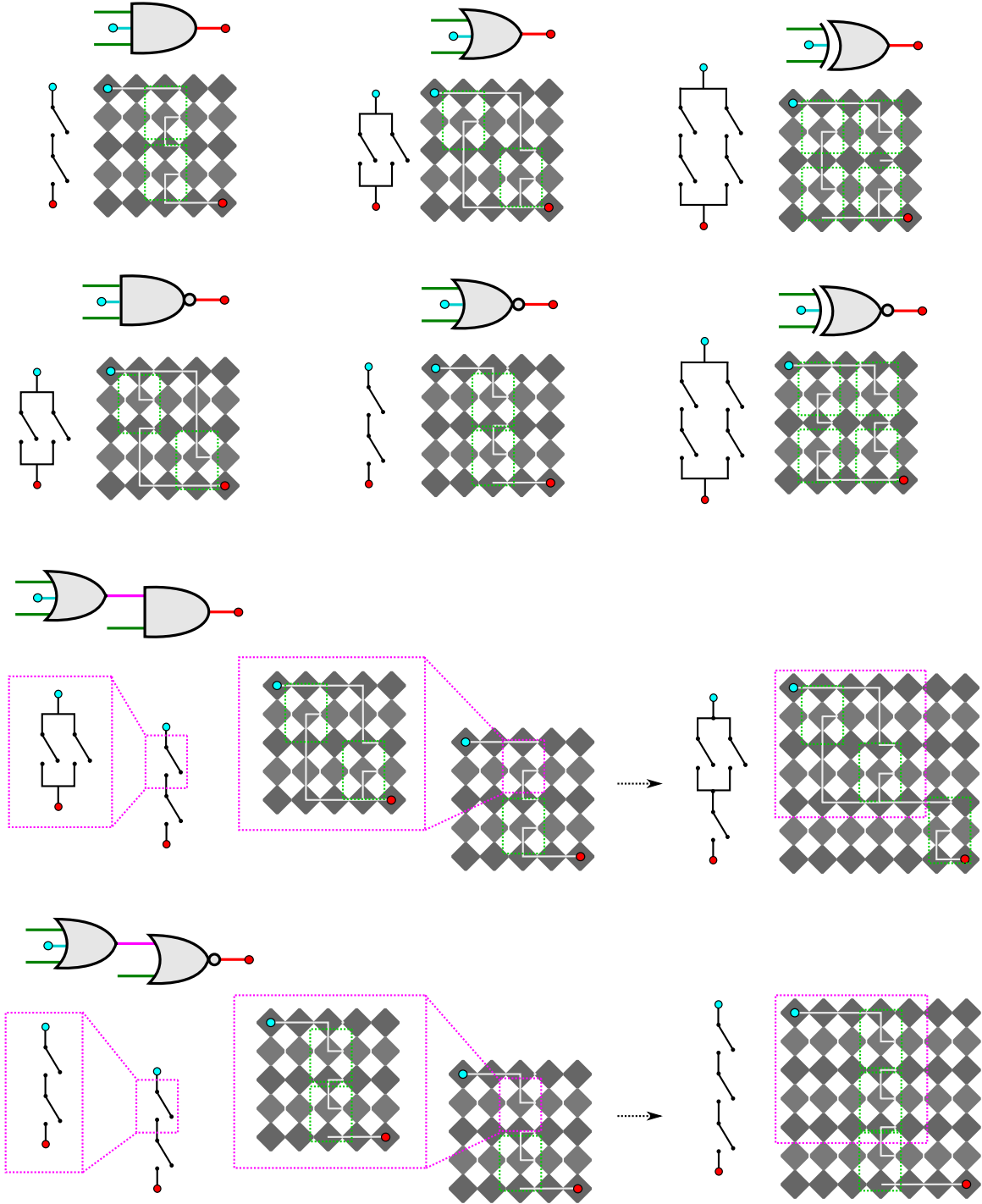


Figure S1. Schematics of the material system for the fundamental 2-bit gates with the corresponding switch circuit: (a) AND, (b) NAND, (c) OR, (d) NOR, (e) XOR, and (f) XNOR. (g) Rule 1 of the substitution method (SM) that shows an OR gate substituted for a buffer switch in the AND gate to form the 2-gate OR-AND combinational logic. (h) Rule 2 of the SM that shows a NOR gate substituted for a NOT switch in the NOR gate to form the 2-gate OR-NOR combinational logic.

Based on such mathematical switching representations [2], a proposition is made to describe the general substitution method (SM) for any logic gate connections.

Proposition 0.1 (General SM). *Let $f: \{0,1\}^m \rightarrow \{0,1\}$ and $g: \{0,1\}^n \rightarrow \{0,1\}$ be Boolean functions with inputs denoted $A_1, \dots, A_m$ and $B_1, \dots, B_n$, respectively. Define $h: \{0,1\}^m \times \{0,1\}^{n-1} \rightarrow \{0,1\}$ by Equation 14.*

$$h(A_1, \dots, A_m, B_2, \dots, B_m) = g(f(A_1, \dots, A_m), B_2, \dots, B_m) \quad (14)$$

Taking a g-switch and replacing B_1 -switch with an f -switch and each B'_1 -switch with a f' -switch yields an h-switch.

Proof for proposition 0.1. Suppose that $A_1, \dots, A_m$ are such that $f(A_1, \dots, A_m) = 1$. By design, all B_1 -switches within the g-switch are closed and all B'_1 -switches within the g-switch are open. By the definition of a g-switch, the resulting configurations will be conductive if and only if $g(1, B_2, \dots, B_m)$. That is, the overall configuration is conductive if and only if $h(A_1, \dots, A_m, B_2, \dots, B_m)$, as is desired. By a similar argument, we can observe that $A_1, \dots, A_m$ such that $f(A_1, \dots, A_m) = 0$, the resulting configuration is conductive if and only if $h(A_1, \dots, A_m, B_2, \dots, B_m)$. Thus, in all cases, the combined circuit behaves like an h-switch.

Such general description of the SM may be utilized to illustrate a 2-gate combination in a simplified proposition.

Proposition 0.2 (2-gate SM). *Let $f, g: \{0,1\}^2 \rightarrow \{0,1\}$ be Boolean functions. Define $h: \{0,1\}^3 \rightarrow \{0,1\}$ by Equation 15.*

$$h(A_1, A_2, B_2) = g(f(A_1, A_2), B_2) \quad (15)$$

Taking a g-switch and replacing each B_1 -switch with an f -switch and each B'_1 -switch with a f' -switch yields an h-switch.

Proposition 0.2, which is a simplification of Proposition 0.1, is demonstrated in Figure S1(g) and (h). For instance. Figure S1(g) shows an f_{OR} replacing the B_1 -switches in the g_{AND} to yield an $h_{OR-AND}(A_1, A_2, B_2)$. Furthermore, Figure S1(h) shows an f'_{OR} or f_{NOR} (since $f'_{OR} = f_{NOR}$) replacing the B'_1 -switches in the g_{NOR} to yield an $h_{OR-NOR}(A_1, A_2, B_2)$.

Such a mathematical representation of the Substitution Method may be generalized through two rules.

1. The first rule: substitute the B_1 switch of gate 1 with gate 2 if the B_1 switch of gate 1 is a Buffer switch.
2. The second rule: substitute the B_1 switch of gate 1 with the inverted gate 2 if the B_1 switch of gate 1 is a NOT switch.

Through the SM, any combinational logic gate combination may be designed without the extraction of the canonical Boolean function. A comparison of the SM with the QMSoP method is discussed in more detail in the following section.

2 Design strategy tutorial: Quine-McCluskey Sum of Product (QMSoP)

To illustrate the Quine-McCluskey Sum of Product (QMSoP) design strategy tool capabilities, a tutorial of the MATLAB and Simulink model code is provided here through an exemplary design procedure of a full adder computing material. Even though a full adder is the demonstration case used, any combinational logic operation may be constructed using the following models. A three-step procedure is described in the following sections 2.1, 2.2, and 2.3.

2.1 **Step 1:** Simulink models

The logic diagram for a full adder is shown in Figure S2(a), which contains three digital inputs: A, B, and C_{in} . The full adder results in two digital outputs, the Q_{Sum} and Q_{Cout} . Here, the logic path that results in each output is considered an independent operation that may be designed separately. By utilizing the Simulink *logic and bit operations* library, the Q_{Cout} model is developed as shown in Figure S2(b).

(a) The input blocks are a *constant* block parameter corresponding to $cond(n,:)$, with n representing the material row number. In this example, the inputs A, B, and C_{in} correspond to row number 1, 2, and 3, respectively. If the inputs are consistent with the logic diagrams in all output Simulink models, the row numbers may be chosen arbitrarily to control the input order.

(b) After the input blocks have been created, the *logic and bit operations* library is utilized again to create the combinational logic path with the appropriate gates (found in the *Logic Operator* block), and connections.

(c) The Q_{Cout} output terminal is then represented by the *To Workspace* (found in *sinks*) output block parameter labeled as *out.Truth*. Such block conveys the truth table binary results to the MATLAB workspace.

(d) Finally, the Simulink model for the Q_{Cout} output is saved as '*Logic_Design_Full_Adder_Cout*'. Similar procedures and variable naming conventions are followed to formulate the Q_{Sum} output Simulink model that is saved as '*Logic_Design_Full_Adder_Sum*', Figure S2(c).

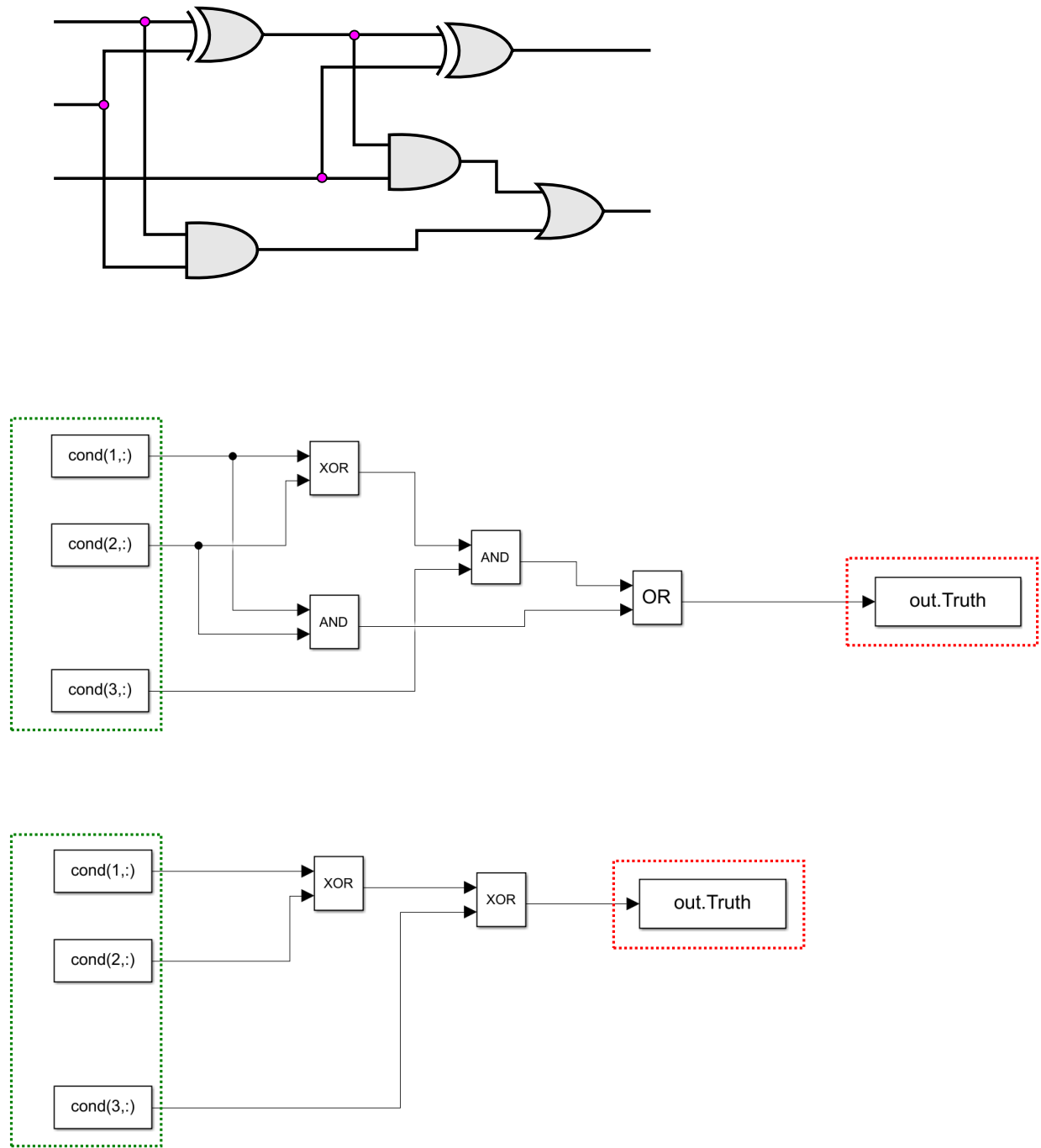


Figure S2. Step 1: Simulink models. (a) A schematic of the full adder logic diagram. Simulink model snapshots of the (b) Q_{Cout} and (c) Q_{Sum} output operations.

2.2 Step 2: MATLAB program

After step 1 is complete and a Simulink model is developed for each digital output, the 'QMSSoP_Main_Code' MATLAB program is utilized to obtain the computing network and metamaterial design. As shown in Figure S3, two parameters are edited in the MATLAB code for each output operation.

(a) The first parameter labeled as *inputs* corresponds to the quantity of inputs that are involved in each operation. For the full adder Q_{Cout} output, such parameter is set to 3.

(b) The Simulink model filename is varied depending on the output operation being designed. Here, the Q_{Cout} model is named 'Logic_Design_Full_Adder_Cout'.

(c) The MATLAB code is then run to obtain a figure (shown in step 3) of the material design. The code is capable of constructing the truth table, extracting the canonical Boolean function, applying the Quine-McCluskey minimization [3] and finally connecting the appropriate switches to program the metamaterial and network for each output. Similar procedure is repeated with *inputs* set at 3 and the file name changed to 'Logic_Design_Full_Adder_Sum' to obtain the design for the Q_{Sum} output.

```
7 - inputs = 3; %% CHANGE NEEDED: insert quantity of inputs
8 - cond = getcondvects(inputs)';
9
10 %% Run Simulink
11 - out = sim('Logic_Design_Full_Adder_Cout'); %% CHANGE NEEDED: insert logic design Simulink file name;
```

Figure S3. Step 2: MATLAB program. A snapshot of the initial code lines in the 'QMSSoP_Main_Code' MATLAB code highlighting the two parameters that are edited by the user.

2.3 Step 3: Results

This step involves the utilization of the results from step 2 to formulate the final material design. The design results from step 2 for the Q_{Cout} and Q_{Sum} outputs are shown in Figure S4(a) and (b), respectively. To develop the final full adder material system:

(a) The 3- and 4-column metamaterial platforms of the two output operations are mechanically connected in parallel to form the 7-column architecture in Figure S4(c).

(b) The cyan V_{cc} electrical terminals are connected to merge a single power supply to both operations. Through the three step procedure described above, the final full adder computing material is developed as illustrated in Figure S4(c).

A similar 3-step procedure is followed to formulate all the operations demonstrated in this research.

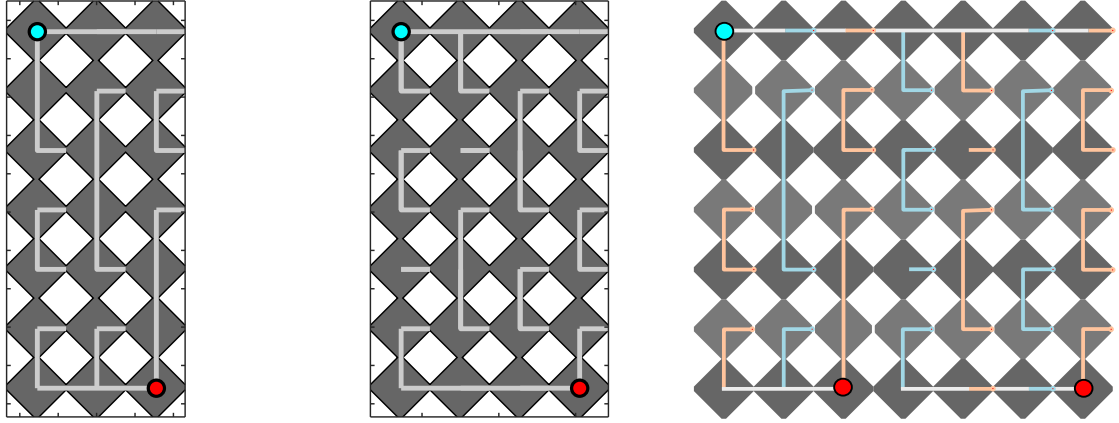


Figure S4. Step 3: Results. The material design figures resulting from the MATLAB program for the (a) Q_{Cout} and (b) Q_{Sum} output operations. (c) A schematic of the complete full adder computing material with both digital outputs.

References

- [1] C. El Helou, P.R. Buskohl, C.E. Tabor, R.L. Harne, Digital logic gates in soft, conductive mechanical metamaterials, *Nature Communications* **12**, 1633 (2021).
- [2] C.E. Shannon, A symbolic analysis of relay and switching circuits, *Electrical Engineering* **57**, 713-723 (1938).
- [3] GitHub. [Online].; 2017 [cited 2021. Available from: <https://github.com/JuiHsiu/Quine-McCluskey-Algorithm-in-Matlab>.