ARTICLE

https://doi.org/10.1038/s41586-019-1493-8

Modern microprocessor built from complementary carbon nanotube transistors

Gage Hills[1,3], Christian Lau[1,3], Andrew Wright[1], Samuel Fuller[2], Mindy D. Bishop[1], Tathagata Srimani[1], Pritpal Kanhaiya[1], Rebecca Ho[1], Aya Amer[1], Yosi Stein[2], Denis Murphy[2], Arvind[1], Anantha Chandrakasan[1] & Max M. Shulaker[1]*

Electronics is approaching a major paradigm shift because silicon transistor scaling no longer yields historical energy-efficiency benefits, spurring research towards beyond-silicon nanotechnologies. In particular, carbon nanotube field-effect transistor (CNFET)-based digital circuits promise substantial energy-efficiency benefits, but the inability to perfectly control intrinsic nanoscale defects and variability in carbon nanotubes has precluded the realization of very-large-scale integrated systems. Here we overcome these challenges to demonstrate a beyond-silicon microprocessor built entirely from CNFETs. This 16-bit microprocessor is based on the RISC-V instruction set, runs standard 32-bit instructions on 16-bit data and addresses, comprises more than 14,000 complementary metal–oxide–semiconductor CNFETs and is designed and fabricated using industry-standard design flows and processes. We propose a manufacturing methodology for carbon nanotubes, a set of combined processing and design techniques for overcoming nanoscale imperfections at macroscopic scales across full wafer substrates. This work experimentally validates a promising path towards practical beyond-silicon electronic systems.

With diminishing returns of silicon field-effect transistor (FET) scaling[1], the need for FETs leveraging nanotechnologies has been steadily increasing. Carbon nanotubes (CNTs, nanoscale cylinders made of a single sheet of carbon atoms with diameters of approximately 10–20 Å) are prominent among a variety of nanotechnologies that are being considered for next-generation energy-efficient electronic systems[2–4]. Owing to the nanoscale dimensions and simultaneously high carrier transport of CNTs[5,6], digital systems built from FETs fabricated with CNTs as the transistor channel (that is, CNFETs) are projected to improve the energy efficiency of today's silicon-based technologies by an order of magnitude[3,7,8].

Over the past decade, CNT technology has matured: from single CNFETs[9] to individual digital logic gates[10,11] to small-scale digital circuits and systems[7,12–16]. In 2013, this progress led to the demonstration of a complete digital system: a miniature computer[2] comprising 178 CNFETs that implemented only a single instruction operating on only a single bit of data (see Supplementary Information for a full discussion of previous work). However, as with all emerging nanotechnologies, there remained a substantial disconnect between these small-scale demonstrations and modern systems comprising tens of thousands of FETs (for example, microprocessors) to billions of FETs (for example, high-performance computing servers). Perpetuating this divide is the inability to achieve perfect atomic-level control of nanomaterials at macroscopic scales (for example, yielding CNTs of consistent 10-Å diameter uniformly across industry-standard wafer substrates of diameter 150–300 mm). The resulting intrinsic defects and variations have made the realization of such modern systems infeasible. For CNTs, there are three major intrinsic challenges: material defects, manufacturing defects and variability.

(1) Material defects. Although semiconducting CNTs form energy-efficient FET channels, the inability to precisely control CNT diameter and chirality results in every CNT synthesis containing some percentage of metallic CNTs. Metallic CNTs have little to no bandgap and therefore their conductance cannot be sufficiently modulated by the CNFET gate, resulting in high leakage current and potentially incorrect logic functionality[17].

(2) Manufacturing defects. During wafer fabrication, CNTs inherently 'bundle' together, forming thick CNT aggregates[18,19]. These aggregates result in CNFET failure (reducing CNFET circuit yield), as well as prohibitively high particle contamination rates for very-large-scale integration (VLSI) manufacturing.

(3) Variability. Energy-efficient complementary metal–oxide–semiconductor (CMOS)[20] digital logic requires the ability to fabricate CNFETs of complementary polarities (p-CNFETs and n-CNFETs) with well-controlled characteristics (for example, tunable and uniform threshold voltages, and p- and n-CNFETs with matching on- and off-state current). Previous techniques for realizing CNT CMOS have relied on either extremely reactive, non-air-stable, non-silicon CMOS-compatible materials[21–25] or have lacked tunability, robustness and reproducibility[26]. This has severely limited the complexity of CNT CMOS demonstrations (a complete CNT CMOS digital system has not yet been fabricated).

Although much previous work has focused on overcoming these challenges, none meets all of the strict requirements for realizing VLSI systems. In this work, we overcome the intrinsic CNT defects and variations to enable a demonstration of a beyond-silicon modern microprocessor: RV16X-NANO, designed and fabricated entirely using CNFETs. RV16X-NANO is a 16-bit microprocessor based on the open-source and commercially available RISC-V instruction set processor, running standard RISC-V 32-bit instructions on 16-bit data and addresses. It integrates >14,000 CMOS CNFETs, and operates as modern microprocessors do today (for example, it can run compiled programs; in addition, we demonstrate its functionality by executing all types and formats of instructions in the RISC-V instruction-set architecture). This is made possible by our manufacturing methodology for CNTs (MMC)—a set of original processing and circuit design techniques that are combined to overcome the intrinsic CNT challenges. The key elements of MMC are:

[1]Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology (MIT), Cambridge, MA, USA. [2]Analog Devices, Inc. (ADI), Wilmington, MA, USA. [3]These authors contributed equally: Gage Hills, Christian Lau. *e-mail: shulaker@mit.edu

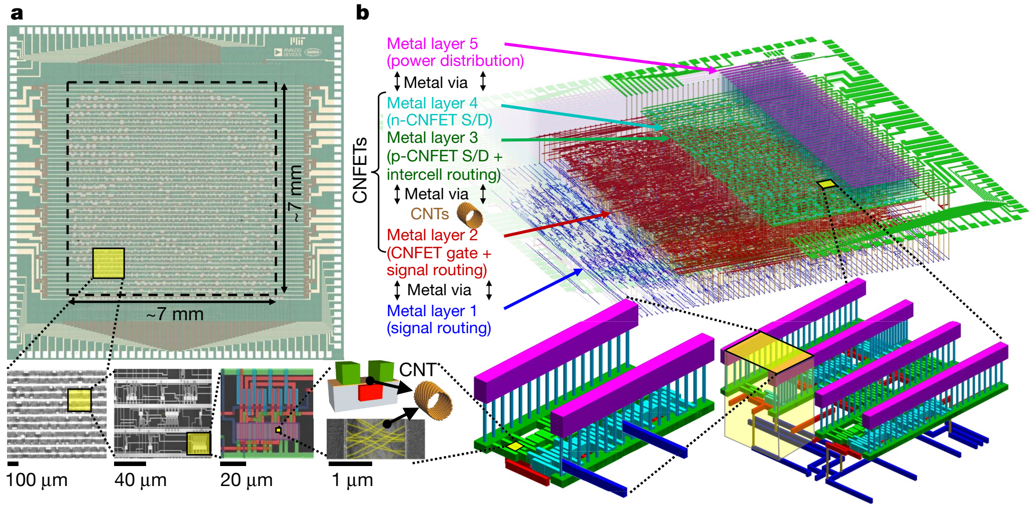


Fig. 1 | RV16X-NANO. a, Image of a fabricated RV16X-NANO chip. The die area is 6.912 mm × 6.912 mm, with input/output pads placed around the periphery. Scanning electron microscopy images with increasing magnification are shown below (one image is false-coloured to match the colouring in the schematic in **b**). RV16X-NANO is fabricated entirely from CNFET CMOS, in a wafer-scalable, VLSI-compatible, and silicon-CMOS compatible fashion. **b**, Three-dimensional to-scale rendered schematic of the RV16X-NANO physical layout (all dimensions are to scale except for the *z* axis, which is magnified to clarify each individual vertical layer). RV16X-NANO leverages a new three-dimensional (3D) physical architecture in which the CNFETs are physically located in the middle of the stack, with metal routing both above and below.

(1) RINSE (removal of incubated nanotubes through selective exfoliation). We propose a method of removing CNT aggregate defects through a selective mechanical exfoliation process. RINSE reduces CNT aggregate defect density by >250× without affecting non-aggregated CNTs or degrading CNFET performance.

(2) MIXED (metal interface engineering crossed with electrostatic doping). Our combined CNT doping process leverages both metal contact work function engineering as well as electrostatic doping to realize a robust wafer-scale CNFET CMOS process. We experimentally yield entire dies with >10,000 CNFET CMOS digital logic gates (2-input

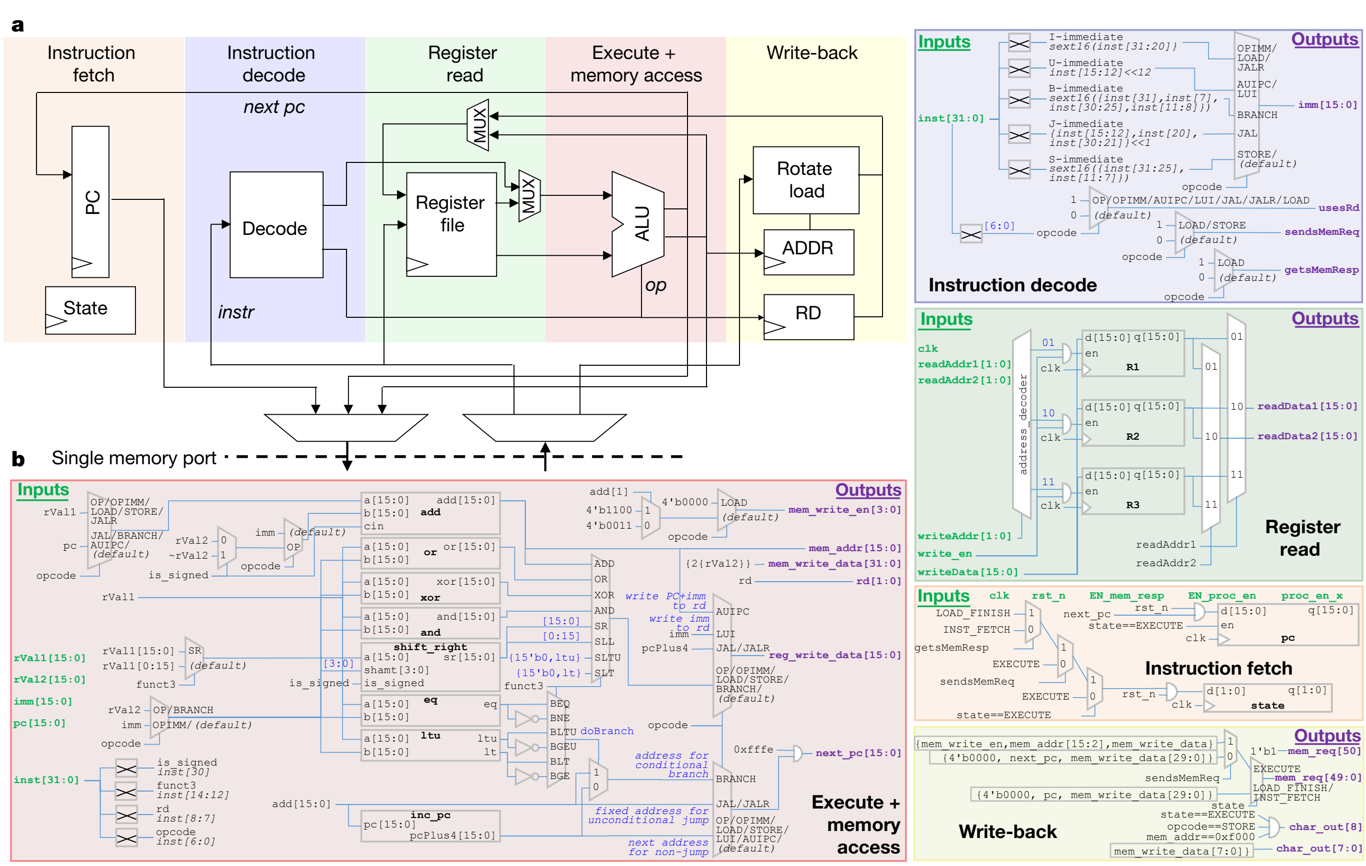


Fig. 2 | Architecture and design of RV16X-NANO. a, Block diagram showing the organization of RV16X-NANO, including the instruction fetch, instruction decode, register read, execute + memory access, and write-back stages. See Supplementary Information section 'RISC-V: Operational Details' for definitions of terms. **b**, Schematics describing the high-level register transfer level (RTL) description of each stage, including inputs, outputs and signal connections. Additional information on the RV16X-NANO is in the Supplementary Information.

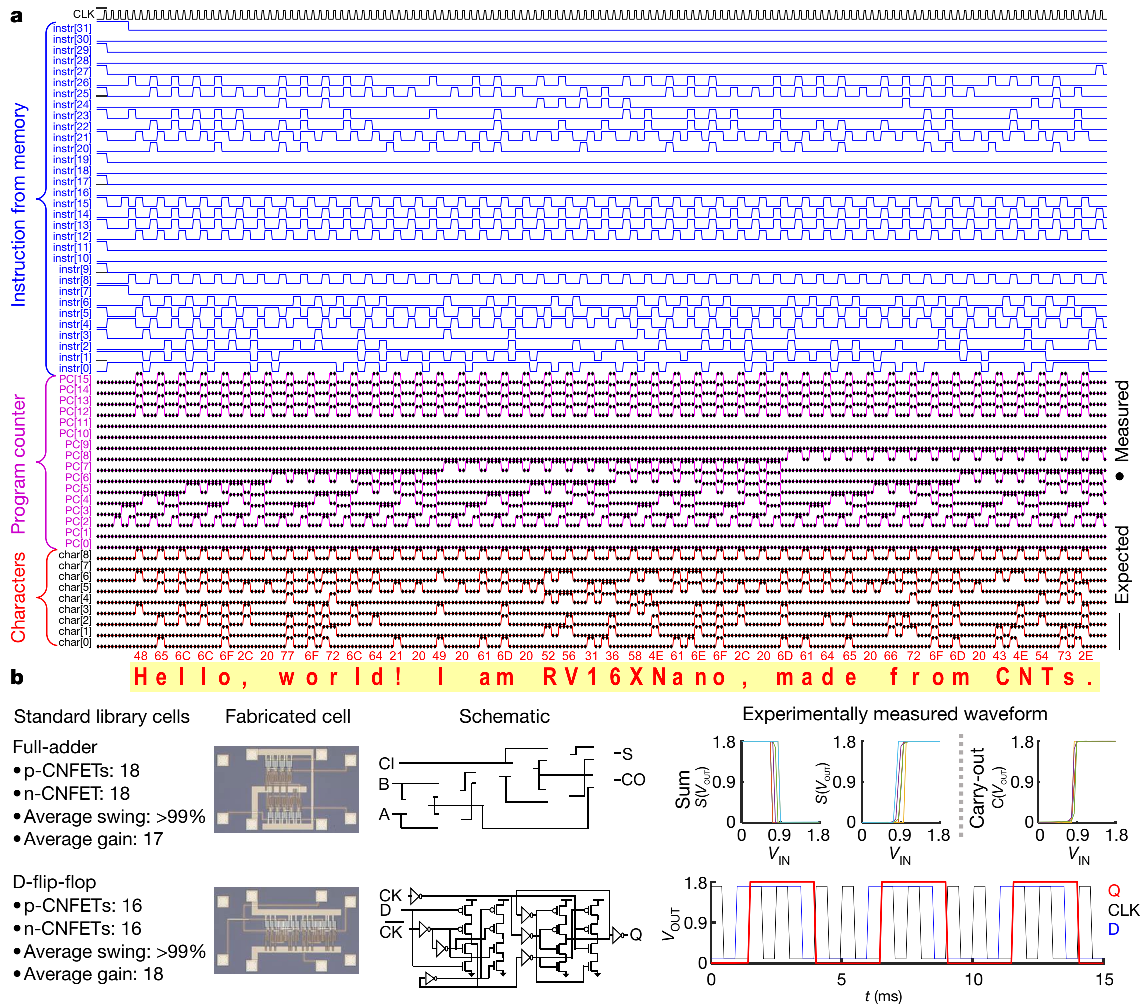


Fig. 3 | RV16X-NANO experimental results. a, Experimentally measured waveform from RV16X-NANO, executing the famous 'Hello, World' program. The waveform shows the 32-bit instruction fetched from memory, the program counter stored in RV16X-NANO, as well as the character output from RV16X-NANO. Below the waveform, we convert the binary output (shown in red in hexadecimal code) to their ASCII characters, showing RV16X-NANO printing out "Hello, world! I am RV16XNano, made from CNTs." In addition to this program, we test functionality by executing all of the 31 instructions within RV32E (see Supplementary Information). **b**, RV16X-NANO is designed using conventional electronic design automation (EDA) tools, leveraging our CNT process design kit and CNT CMOS standard cell library. An example combinational cell (full-adder) and example sequential cell (D-flip-flop) are shown alongside an optical microscopy image of the fabricated cells, their schematics, as well as their experimentally measured waveforms. For the full-adder, we show the outputs (sum and carry-out outputs) for all possible biasing conditions in which sweeping the voltage of input (from 0 to V_{DD}) causes a change in the logical state of the output (that is, for the full adder, with $C_{OUT} = A*B + B*C_{IN} + A*C_{IN}$, with A = logical '0' and B = logical '1', then sweeping C_{IN} from '0' to '1' causes C_{OUT} to change from logical '0' to logical '1'). (CI indicates C_{IN} and CO indicates C_{OUT}.) For the sum output $S(V_{OUT})$, there are 12 such conditions: six where V_{OUT} has the same polarity as the swept input (positive unate) and six where V_{OUT} has the opposite polarity to the swept input (negative unate). For the carry-out output $C(V_{OUT})$ there are six such conditions (all positive unate); the measurements are overlaid over one another in **b**). Gain for all transitions is >15, with output voltage swing >99%. The D-flip-flop waveform (voltage versus time) illustrates correct functionality of the positive edge-triggered D-flip-flop (output state Q shows correct functionality based on data input D and clock input CLK). CK and $\overline{CK}$ are the clock input and the inverse of the clock input, respectively.

'not-or' gates with functional yield 14,400/14,400, comprising 57,600 total CNFETs), and present a wafer-scale CNFET CMOS uniformity characterization across 150-mm wafers (such as analysing the yield for more than 100 million possible combinations of cascaded logic gate pairs).

(3) DREAM (designing resiliency against metallic CNTs). This technique overcomes the presence of metallic CNTs entirely through circuit design. DREAM relaxes the requirement on metallic CNT purity by about 10,000× (relaxed from a semiconducting CNT purity requirement of 99.999999% to 99.99%), without imposing any additional processing steps or redundancy. DREAM is implemented using standard electronic design automation (EDA) tools, has minimal cost, and enables digital VLSI systems with CNT purities that are available commercially today.

Importantly, the entire MMC is wafer-scale, VLSI-compatible and is seamlessly integrated within existing infrastructures for silicon CMOS—both in terms of design and of processing. Specifically, RV16X-NANO is designed with standard EDA tools, and leverages only materials and processes that are compatible with and exist within commercial silicon CMOS manufacturing facilities. Together, these contributions establish a robust CNT CMOS technology and represent a major milestone in the development of beyond-silicon electronics.

RV16X-NANO

Figure 1 shows an optical microscopy image of a fabricated RV16X-NANO die alongside three-dimensional to-scale rendered schematics of the physical layout. It is the largest CMOS electronic system

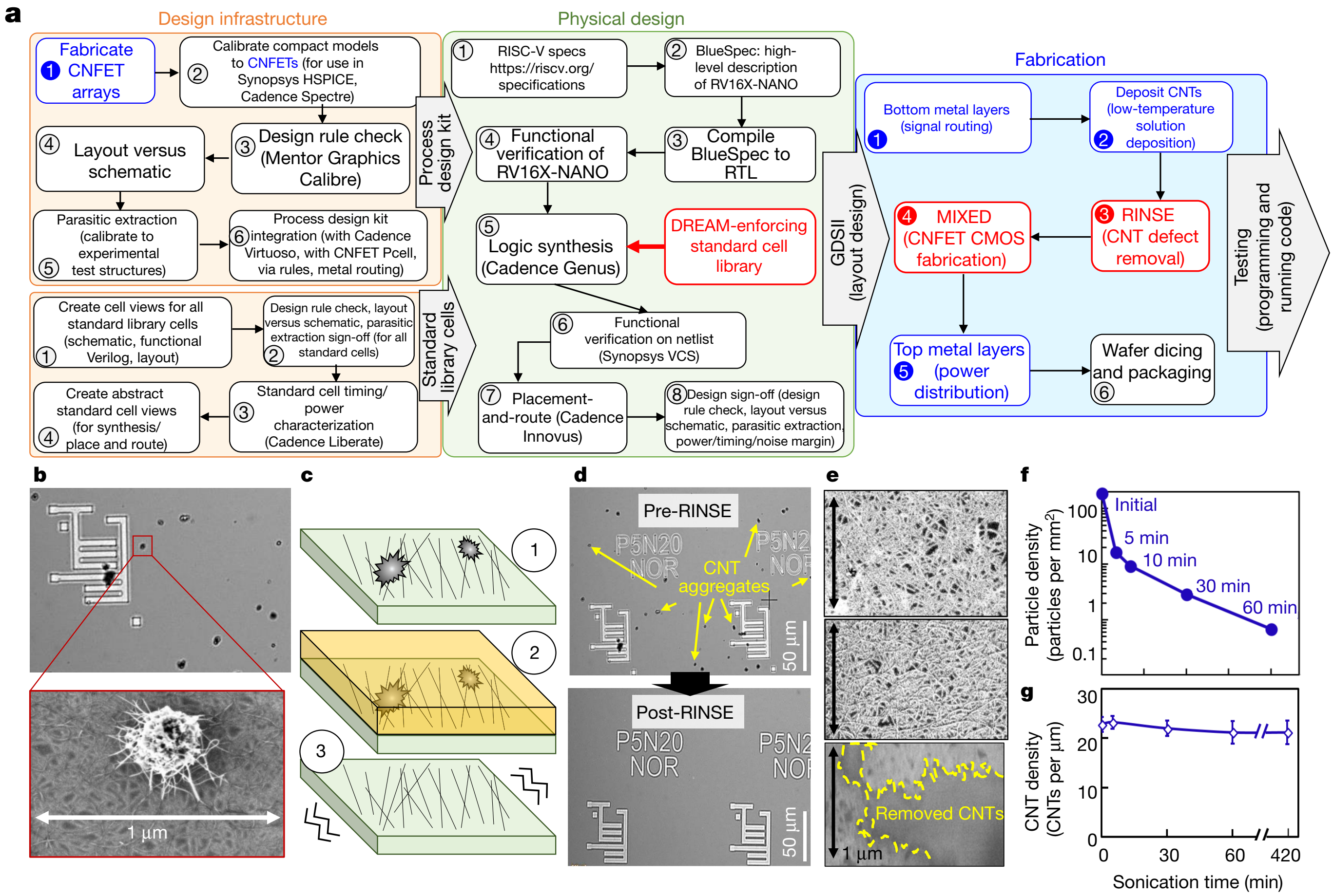


Fig. 4 | MMC. **a**, Design and manufacturing flow for RV16X-NANO, illustrating how MMC seamlessly integrates within conventional silicon-based EDA tools. Black boxes show conventional steps in silicon-CMOS design flows. Blue text indicates steps that are adjusted for CNTs instead of silicon, and red text represents the additions needed to implement the MMC. RV16X-NANO is the first hardware demonstration of a beyond-silicon emerging nanotechnology leveraging a complete RTL-to-GDS physical design flow that uses only conventional EDA tools. Software packages are from Synopsys (https://www.synopsys.com/), Cadence (https://www.cadence.com/) and Mentor Graphics (https://www.mentor.com/). **b**, RINSE. As shown in the scanning electron microscopy images, CNTs inherently bundle together, forming thick CNT aggregates. These aggregates result in CNFET failure (reduced CNFET yield) as well as prohibitive particle contamination for VLSI manufacturing. **c**, The RINSE process steps: (1) CNT incubation, (2) adhesion coating, (3) mechanical exfoliation (see text for details). **d**, **e**, RINSE results. After performing RINSE, CNT aggregates are removed from the wafer (as shown in **d**). Importantly, the individual CNTs not in aggregates are not removed from the wafer, while without RINSE, sonication inadvertently removes large areas of all CNTs from the wafer (in **e**, where the top shows CNT incubation pre-RINSE, the middle shows CNTs left on the wafer post-RINSE, and the bottom shows CNTs inadvertently removed from the wafer after sonicating a wafer to remove CNT aggregates without performing the critical adhesion-coating step in RINSE). **f**, Particle contamination reduction due to RINSE: RINSE decreases particle density by >250×. **g**, Ideally, individual CNTs are not inadvertently removed during RINSE; increasing the time of step 3 (sonication time) to over 7 h results in no change in CNT density across the wafer.

realized using beyond-silicon nanotechnologies: comprising 3,762 CMOS digital logic stages, totalling 14,702 CNFETs containing more than 10 million CNTs, and includes logic paths comprising up to 86 stages of cascaded logic between flip-flops (that is, that must evaluate sequentially in a single clock cycle). It operates with supply voltage (V_{DD}) of 1.8 V, receives an external referenced clock (generating local clock signals internally), receives inputs (instructions and data) from and writes directly to an off-chip main memory (dynamic random-access memory, DRAM), and stores data on-chip in a register file. No other external biasing or control signals are supplied. Furthermore, RV16X-NANO has a three-dimensional (3D) physical architecture, as the metal interconnect layers are fabricated both above and below the layer of CNFETs; this is in contrast to silicon-based systems in which all metal routing can only be fabricated above the bottom layer of silicon FETs (see Methods). In RV16X-NANO, the metal layers below the CNFETs are primarily used for signal routing, while the metal layers above the CNFETs are primarily used for power distribution (Fig. 1c, d). The fabrication process implements five metal layers and includes more than 100 individual processing steps (see Methods and section 'MMC' for details). This 3D layout, with routing above and below the FETs promises improved routing congestion (a major challenge for today's systems[27]), and is uniquely enabled by CNTs (owing to their low-temperature fabrication; see Methods).

Physical design

The design flow of RV16X-NANO leverages only industry-standard tools and techniques: we create a standard process design kit (PDK) for CNFETs as well as a library of standard cells for CNFETs that is compatible with existing EDA tools and infrastructure without modification. Our CNFET process design kit includes a compact model for circuit simulations that is experimentally calibrated to our fabricated CNFETs. The standard cell library comprises 63 unique cells, and includes both combinational and sequential circuit elements implemented with both static CMOS and complementary transmission-gate digital logic circuit topologies (see Supplementary Information for a full list of standard library cells, including circuit schematics and physical layouts). We use the CNFET process design kit to characterize the timing and power for all of the library cells, which we experimentally validate by fabricating and measuring all cells individually (see Supplementary Information for full description and experimental characterization of the standard

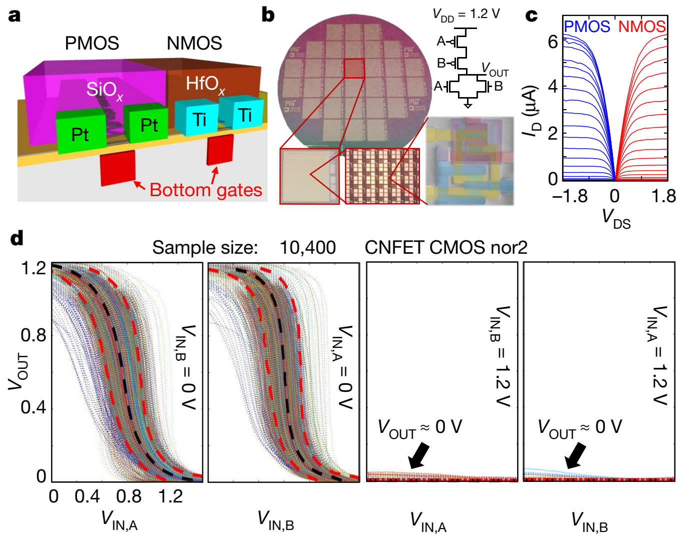


Fig. 5 | MIXED. a, Schematic of CNFET CMOS fabricated using MIXED. MIXED is a combined doping process that leverages both metal contact work-function engineering as well as electrostatic doping to realize a robust wafer-scale CNFET CMOS process. We use platinum contacts and SiO_x passivation for p-CNFETs, and titanium contacts and HfO_x passivation for n-CNFETs (see Methods for details). To characterize MIXED, we fabricated dies with 10,400 CNFET CMOS digital logic gates across 150-mm wafers (**b**). **c**, **d**, Experimental results. **c**, I_D versus V_{DS} characteristics showing p-CNFETs and n-CNFETs that exhibit similar I_D–V_{DS} characteristics (for opposite polarity of input bias conditions, for example, $V_{DS,P} = -V_{DS,N}$), achieved with MIXED. The gate-to-source voltage V_{GS} is swept from $-V_{DD}$ to V_{DD} in increments of 0.1 V. See Supplementary Information for I_D–V_{GS} and additional CNFET characteristics. **d**, Output voltage transfer curves (VTCs, V_{OUT} vs V_{IN}) for all 10,400 CNT CMOS logic gates (nor2) within a single die. Each VTC illustrates V_{OUT} as a function of the input voltage of one input (V_{IN}), while the other input is held constant. For each nor2 logic gate (with logical function OUT = !(IN_A|IN_B), we measure the VTC for each of two cases: V_{OUT} versus $V_{IN,A}$ with $V_{IN,B} = 0$ V and V_{OUT} versus $V_{IN,B}$ with $V_{IN,A} = 0$ V). All 10,400/10,400 exhibit correct functionality (which we define as having output voltage swing >70%). The black dotted line represents the average VTC (average V_{IN} across all measured VTC for each value of V_{OUT}), while the red dotted line represents the boundary of ±3 standard deviations (again, across all V_{IN} values for each value of V_{OUT}). See Supplementary Information for extracted distributions of key metrics from these experimental measurements (gain, output voltage swing and SNM analysing >100 million possible cascaded logic gates pairs formed from these 10,400 samples), as well as uniformity characterization across the 150-mm wafer. Importantly, despite the high yield and robust CNFET CMOS enabled by MIXED and RINSE, we note that there are outlier gates with degraded output swing (the blue lines in **d**). These outliers are caused by CNT CMOS logic gates that contain metallic CNTs; the third component of the MMC (DREAM; see Fig. 6), is a design technique that is essential for overcoming the presence of these metallic CNTs.

cell library). A full description of our industry-practice VLSI design methodology, including how we implement DREAM during logic synthesis and place-and-route, is provided in the Methods.

Computer architecture

Figure 2 illustrates the architecture of RV16X-NANO, which follows conventional microprocessor design (implementing instruction fetch, instruction decode, register read, execute/memory access, and write-back stages). It is designed from RISC-V, a standard open instruction-set architecture used in commercial products today and gaining widespread popularity in both academia and industry[28,29]; see https://riscv.org/wp-content/uploads/2017/05/Tue1345pm-NVIDIA-Sijstermans.pdf and https://www.westerndigital.com/company/innovations/risc-v). RV16X-NANO is derived from a full 32-bit RISC-V microprocessor supporting the RV32E instruction set (31 different 32-bit instructions, see Supplementary Information), while truncating the data path width from 32 bits to 16 bits, and reducing the number of registers from 16 to 4. It is designed using the publicly available software Bluespec (https://bluespec.com/), and is verified using a Satisfiability Modulo Theories (SMT)-based bounded model checking against a formal specification of the RISC-V instruction-set architecture (see Supplementary Information). To demonstrate the correct functionality of the microprocessor, we experimentally run and validate correct functionality of all types and formats of instructions on the fabricated RV16X-NANO. Figure 3 shows the first program executed on RV16X-NANO: the famous 'Hello, World'. See Methods and Supplementary Information for schematics, operational details and experimental measurements.

MMC

Here we describe our MMC—a set of combined processing and design techniques that are the foundation for enabling the realization of RV16X-NANO (Fig. 4a). All design and fabrication processes are wafer-scale and VLSI-compatible, not requiring any per-unit customization or redundancy.

RINSE

The CNFET fabrication process begins by depositing CNTs uniformly over the wafer. 150-mm-diameter wafers (with the bottom metal signal routing layers and gate stack of the CNFET already fabricated for the 3D design) are submerged in solutions containing dispersed CNTs (Methods). Although CNTs are uniformly deposited over the wafer, the CNT deposition also inherently results in manufacturing defects: CNT aggregates deposited randomly across the wafer (Fig. 4b). These CNT aggregates act as particle contamination, reducing die yield. Several existing techniques have attempted to remove these aggregates before CNT deposition, but none is sufficient to meet wafer-level yield requirements for VLSI systems: (1) excessive high-power sonication for dispersing aggregates in solution damages CNTs, which results in degraded CNFET performance and does not disperse all CNTs; (2) centrifugation, which does not remove all smaller aggregates (and aggregates can re-form post-centrifugation), (3) excessive filtering, which removes both aggregates and the CNTs themselves from the solution, and (4) etching the aggregates, which is not feasible owing to lack of selectivity versus the underlying CNTs themselves. Instead, to remove these aggregates, we developed a process that we call RINSE, consisting of three steps (Fig. 4c):

(1) CNT incubation. Solution-based CNTs are deposited on wafers pre-treated with a CNT adhesion promoter (hexamethyldisilazane, bis(trimethylsilyl)amine).

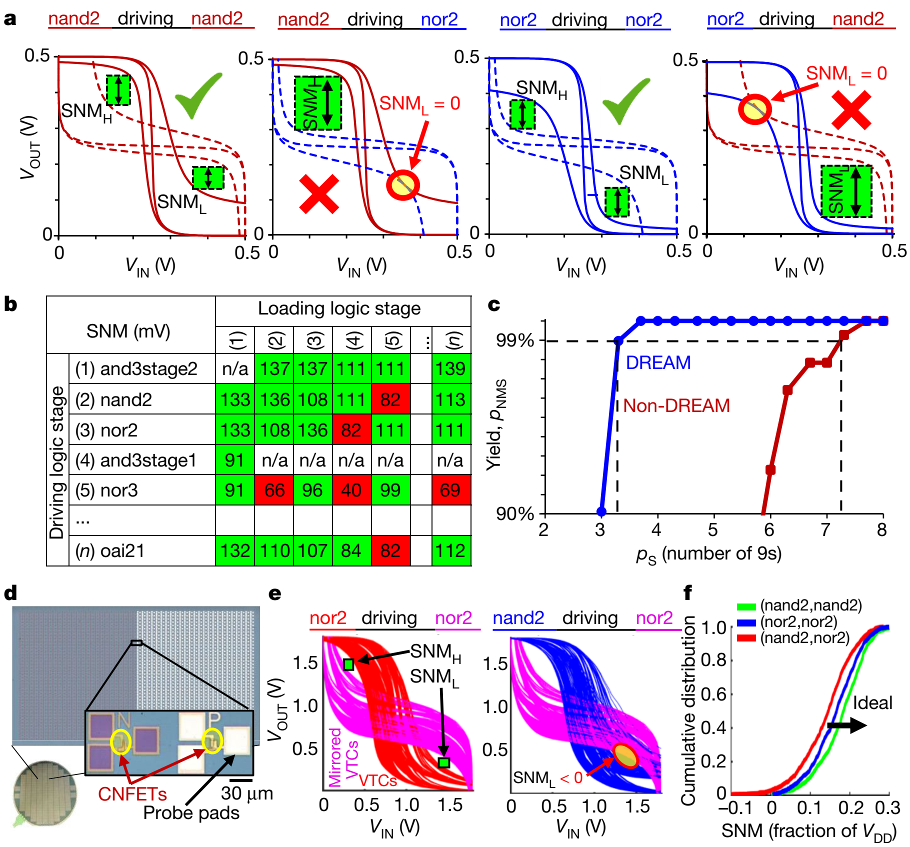
(2) Adhesion coating. A standard photoresist (polymethylglutarimide) is spin-coated onto the wafer and cured at about 200 °C.

(3) Mechanical exfoliation. The wafer is placed in solvent (*N*-methylpyrrolidone) and sonicated.

The key to RINSE is the adhesion coating (step 2): without it, sonicating the wafer inadvertently removes sections of CNTs in addition to the aggregates (Fig. 4d). The adhesion coating leaves an atomic layer of carbon that remains after step 3, which exerts sufficient force to adhere the CNTs to the wafer surface while still allowing for the removal of the aggregates. Experimental results for RINSE are shown in Fig. 4d–g; by optimizing the adhesion-coating cure temperature and time as well as the sonication power and time, RINSE reduces the CNT aggregate density by >250× (quantified by the number of CNT aggregates per unit area) without damaging the CNTs or affecting CNFET performance (see Supplementary Information).

MIXED

After using RINSE to overcome intrinsic CNT manufacturing defects, CNFET circuit fabrication continues. Unfortunately, while energy-efficient CMOS logic requires both p-CNFETs and n-CNFETs with controlled and tunable properties (such as threshold voltage), techniques for realizing CNT CMOS today result in large FET-to-FET



SNM (mV)	Loading logic stage (1)	(2)	(3)	(4)	(5)	⋮	(n)
Driving logic stage (1) and3stage2	n/a	137	137	111	111		139
(2) nand2	133	136	108	111	82		113
(3) nor2	133	108	136	82	111		111
(4) and3stage1	91	n/a	n/a	n/a	n/a		n/a
(5) nor3	91	66	96	40	99		69
...							
(n) oai21	132	110	107	84	82		112

Fig. 6 | DREAM. DREAM overcomes the presence of metallic CNTs entirely through circuit design, and is the final component of the MMC. DREAM relaxes the requirement on metallic CNT purity by about 10,000×, without imposing any additional processing steps or redundancy. DREAM is implemented using standard EDA tools, has minimal cost (≤10% energy, ≤ 10% delay and ≤ 20% area), and enables digital VLSI systems with CNT purities that are available commercially today (99.99% semiconducting CNT purity). **a**, VTCs for driving logic stages and mirrored VTCs for loading logic stages, showing SNM simulated for 4 different logic stage pairs (SNM is defined in the Supplementary Information), with up to two metallic CNTs in all CNFETs. The logic stage pairs: (nand2, nand2) and (nor2, nor2) have better SNM than do (nand2, nor2) and (nor2, nand2) despite all logic stages having exactly the same VTCs. We note that we distinguish logic stages (for example, an inverter) from logic gates (for example, a buffer, by cascading two inverters); a logic gate can comprise multiple logic stages. **b**, Example DREAM SNM table (see Methods for details, analysed for a projected 7-nm node with a scaled V_{DD} of 500 mV), which shows the minimum SNM for each pair of connected logic stages. As an example, values less than 83 mV are highlighted in red, indicating that these combinations would not be permitted during design, to reduce overall susceptibility to noise at the VLSI circuit level. **c**, Yield (p_{NMS}) versus semiconducting CNT purity for a required SNM level (SNM_R) of $SNM_R = V_{DD}/5$, shown for the OpenSparc 'dec' module designed using the 7-nm node CNFET standard library cells derived from the ASAP7 process design kit with a scaled V_{DD} of 500 mV (details in Methods). **d**, Fabricated CNT CMOS die, comprising 1,000 NMOS CNFETs and 1,000 PMOS CNFETs. Semiconducting CNT purity is $p_S \approx 99.99\%$, with around 15–25 CNTs per CNFET. **e**, **f**, Experimental demonstration of DREAM. VTCs for nand2 and nor2 generated by randomly selecting two NMOS and two PMOS CNFETs from **d** (some of which contain metallic CNTs). This is repeated to form 1,000 unique nor2 and nand2 VTCs. We then analyse the SNM for over one million logic stage pairs (shown in **f**), corresponding to all combinations of 1,000 VTCs for the driving logic stage and 1,000 VTCs for the loading logic stage. **e**, A subset of these logic stage pairs; the (nor2, nor2) maintains minimum SNM > 0, while (nand2, nor2) suffers from minimum SNM < 0 in the presence of metallic CNTs; >99.99% of (nor2, nor2) and (nand2, nand2) logic stage pairs achieve SNM > 0 V, while only about 97% of (nand2, nor2) achieve SNM > 0 V. **f**, Cumulative distributions of SNM over one million logic stage pairs.

variability that has made the realization of large-scale CNFET CMOS systems infeasible. Moreover, the vast majority of existing techniques are not air-stable (for example, they use materials that are extremely reactive in air[23]), are not uniform or robust (for example, they do not always successfully realize CMOS[22]), or rely on materials not compatible with conventional silicon CMOS processing (for example, molecular dopants that contain ionic salts prohibited in commercial fabrication facilities[24,25]).

These challenges are overcome by our processing technique, MIXED, described in Fig. 5. The key to MIXED is a combined doping approach that engineers both the oxide deposited over the CNTs to encapsulate the CNFET as well as the metal contact to the CNTs[30]. First, we encapsulate the CNFETs in oxide (deposited by atomic-layer deposition) to isolate them from their surroundings. By leveraging the atomic-layer control of atomic-layer deposition, we also engineer the precise stoichiometry of this oxide encapsulating the CNTs, which enables us to simultaneously electrostatically dope the CNTs (the stoichiometry dictates both the amount of redox reaction at the oxide–CNT interface and the fixed charge in the oxide). In addition, we engineer the metal source/drain contacts to the CNTs to further optimize the p- and n-CNFETs. We use a lower-work-function metal (titanium) for the contacts to n-CNFETs and a higher-work-function metal for the contacts to p-CNFETs (platinum), improving the on-state drive current of both (for a given off-state leakage current). In contrast to previous approaches, MIXED has the following key advantages: it leverages only silicon CMOS-compatible materials, it allows for precise threshold voltage tuning through controlling the stoichiometry of the atomic-layer deposition doping oxide, and it is robust owing to tight process control by using atomic-layer deposition and only air-stable materials.

Figure 5c shows the current–voltage (I–V) characteristics of p-CNFETs and n-CNFETs, demonstrating well-matched characteristics (such as on- and off-state currents). To demonstrate the reproducibility of MIXED at the wafer scale, Fig. 5 shows measurements from 10,400/10,400 correctly functioning 2-input 'not-or' (nor2) CNFET logic gates within a single die, and 1,000/1,000 correctly functioning nor2 gates randomly selected from across a 150-mm wafer. Additional characterization results (including output voltage swing, gain, and SNM for >100 million possible combinations of cascaded logic gate pairs), are in Supplementary Information. This demonstrates solid-state, air-stable, VLSI- and silicon-CMOS compatible CNFET CMOS at the wafer scale.

DREAM

Despite the robust CNFET CMOS enabled by RINSE and MIXED, a small percentage (around 0.01%) of CNTs are metallic CNTs. Unfortunately, a metallic CNT fraction of 0.01% can be prohibitively large for VLSI-scale systems, owing to two major challenges—increased leakage power, which degrades energy-delay product (EDP) benefits, and degraded noise immunity, which potentially results in incorrect logic functionality. To quantify the noise immunity of digital logic, we extract the static noise margin (SNM) for each pair of connected logic stages, using the voltage transfer curves (VTCs) of each stage (details in Extended Data Fig. 8). The probability that all connected logic stages meet a minimum SNM requirement (SNM_R, typically chosen by the designer as a fraction of V_{DD}, for example, $SNM_R = V_{DD}/4$) is p_{NMS}: the probability that all noise margin constraints are satisfied (Methods). Although previous works have set requirements on semiconducting-CNT purity (p_S) based on limiting metallic-CNT-induced leakage power, no existing works have provided VLSI circuit-level guidelines for p_S based on both increased leakage and the resulting degraded SNM. Although p_S of 99.999% is sufficient to limit EDP degradation to ≤5%, SNM imposes far stricter requirements on purity: p_S must be about 99.999999% to achieve $p_{NMS} \geq 99\%$ (analysed for 1 million gate circuits, Supplementary Information).

Unfortunately, typical CNT synthesis today achieves a p_S value of only about 66%. While many different techniques have been proposed to overcome the presence of metallic CNTs (Supplementary Information), the highest reported purity is a p_S of about 99.99%: this is 10,000× below the requirement for VLSI circuits[31–33]. Moreover, these techniques have substantial cost, requiring either additional processing steps (for example, applying high voltages for electrical 'breakdown' of metallic CNTs during fabrication[10]) or redundancy (incurring substantial energy-efficiency penalties[34]). Here we present and experimentally validate a new technique, DREAM, that overcomes the presence of metallic CNTs entirely through circuit design. The key contribution of DREAM is that it reduces the required p_S by around 10,000×, allowing 99% p_{NMS} with $p_S = 99.99\%$ (for circuits with one million logic gates). This enables digital VLSI circuits to use CNT processing available today: $p_S = 99.99\%$ is already commercially available (and can also be achieved through several means, including solution-based sorting, which we use in our process for fabricating RV16X-NANO; see Methods).

The key insight for DREAM is that metallic CNTs affect different pairs of logic stages uniquely depending on how the logic stages are implemented (considering both the schematic and physical layout). As a result, the SNM of specific combinations of logic stages is more susceptible to metallic CNTs. To improve overall p_{NMS} for a digital VLSI circuit, DREAM applies a logic transformation during logic synthesis to achieve the same circuit functionality, while prohibiting the use of specific logic stage pairs whose SNM is most susceptible to metallic CNTs. As an example, let (G_D, G_L) be a logic stage pair with driving logic stage G_D and loading logic stage G_L. Figure 6 shows that some logic stage pairs have better SNM in the presence of metallic CNTs than others, despite using exactly the same VTCs for the logic stages comprising the circuit (in this instance, logic stage pairs (nand2, nand2) and (nor2, nor2) have better SNM than (nand2, nor2) or (nor2, nand2)). Thus, a designer can improve p_{NMS} by prohibiting the use of logic stage pairs that are more susceptible to metallic CNTs, while permitting logic stage pairs that maintain better SNM despite the presence of metallic CNTs.

Beyond this simple example to illustrate DREAM, we also quantify the benefit of DREAM using both simulation and experimental analysis for VLSI-scale circuits; in simulation, we leverage a compact model for CNFETs (derived from ref. [8]), which accounts for both semiconducting CNTs and metallic CNTs, to analyse the effect of metallic CNTs on the leakage power, energy consumption, speed and noise susceptibility of physical designs of VLSI-scale circuits at a 7-nm technology node designed using standard EDA tools, with and without DREAM (results are shown in Fig. 6; see additional discussion in Supplementary Information). Experimentally, we fabricate and characterize 2,000 CMOS CNFETs fabricated with MIXED (1,000 p-type metal-oxide-semiconductor (PMOS) and 1,000 n-type metal-oxide-semiconductor (NMOS) CNFETs; see Fig. 6). Using *I*–*V* measurements from these 2,000 CNFETs, we analyse one million combinations of CNFET digital logic gates (whose electrical characteristics are solved using the *I*–*V* characteristics of the measured CNFETs; Extended Data Fig. 8) to show the benefits of DREAM in reducing circuit susceptibility to noise. In the Methods, we provide extensive details of these analyses and the implementation of DREAM for arbitrary digital VLSI circuits, including how to implement DREAM using standard industry-practice physical design flows, how we implement DREAM for RV16X-NANO, and an efficient algorithm to satisfy target p_{NMS} constraints (such as $p_{NMS} \geq 99\%$), while minimizing energy, delay and area costs.

Outlook

These combined processing and design techniques overcome the major intrinsic CNT challenges. Our complete manufacturing methodology for CNTs (MMC) enables a demonstration of a beyond-silicon modern microprocessor fabricated from CNTs, RV16X-NANO. In addition to demonstrating the RV16X-NANO microprocessor, we thoroughly characterize and analyse all facets of MMC, illustrating the feasibility of our approach and more broadly of a future CNT technology. This work is a major advance for CNTs, paving the way for next-generation beyond-silicon electronic systems.

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/s41586-019-1493-8.

Received: 16 January 2019; Accepted: 3 July 2019;
Published online 28 August 2019.

1. Khan, H. N., Hounshell, D. A. & Fuchs, E. R. H. Science and research policy at the end of Moore's law. *Nat. Electron*. **1**, 14–21 (2018).
2. Shulaker, M. et al. Carbon nanotube computer. *Nature* **501**, 526–530 (2013).
3. Hills, G. et al. Understanding energy efficiency benefits of carbon nanotube field-effect transistors for digital VLSI. *IEEE Trans. NanoTechnol*. **17**, 1259–1269 (2018).
4. Franklin, A. et al. Sub-10 nm carbon nanotube transistor. *Nano Lett*. **12**, 758–762 (2012).
5. Brady, G. J. et al. Quasi-ballistic carbon nanotube array transistors with current density exceeding Si and GaAs. *Science* **2**, e1601240 (2016).
6. Javey, A., Guo, J., Wang, Q., Lundstrom, M. & Dai, H. Ballistic carbon nanotube field-effect transistors. *Nature* **424**, 654–657 (2003).
7. Aly, M. M. S. et al. Energy-efficient abundant-data computing: the N3XT approach to energy-efficient abundant-data computing. *Proc. IEEE* **107**, 19–48 (2019).
8. Lee, C.-S., Pop, E., Franklin, A. D., Haensch, W. & Wong, H.-S. P. A compact virtual-source model for carbon nanotube FETs in the sub-10-nm regime-Part I: Intrinsic elements. *IEEE Trans. Electron Devices* 62.9 3061-3069 (2015).
9. Tans, S. J., Verschueren, A. R. M. & Dekker, C. Room-temperature transistor based on a single carbon nanotube. *Nature* **393**, 49–52 (1998).
10. Patil, N. et al. VMR: VLSI-compatible metallic carbon nanotube removal for imperfection-immune cascaded multi-stage digital logic circuits using carbon nanotube FETs. In *IEEE Int. Electron Devices Meet*. https://doi.org/10.1109/IEDM.2009.5424295 (IEEE, 2009).
11. Cao, Q., Kim, H., Pimparkar, N., Kulkarni, J. & Wang, C. Medium-scale carbon nanotube thin-film integrated circuits on flexible plastic substrates. *Nature* **454**, 495–500 (2008).
12. Shulaker, M., Saraswat, K., Wong, H. & Mitra, S. Monolithic three-dimensional integration of carbon nanotube FETs with silicon CMOS. In *Symp. VLSI Technology Digest Tech. Pap*. https://doi.org/10.1109/VLSIT.2014.6894422 (IEEE, 2014).
13. Shulaker, M. et al. Carbon nanotube circuit integration up to sub-20 nm channel lengths. *ACS Nano*. **8**, 3434–3443 (2014).
14. Shulaker, M. et al. Experimental demonstration of a fully digital capacitive sensor interface built entirely using carbon-nanotube FETs. In *IEEE Int. Solid-State Circuits Conf. Digest Tech. Pap*. https://doi.org/10.1109/ISSCC.2013.6487660 (IEEE, 2013).

15. Shulaker, M. et al. Sensor-to-digital interface built entirely with carbon nanotube FETs. *IEEE J. Solid-State Circ.* **41**, https://doi.org/10.1109/JSSC.2013.2282092 (2014).
16. Ding, L. et al. CMOS-based carbon nanotube pass-transistor logic integrated circuits. *Nat. Commun.* **3**, 677 (2012).
17. Shulaker, M. et al. Efficient metallic carbon nanotube removal for highly-scaled technologies. *In IEEE Int. Electron Devices Meet.* https://doi.org/10.1109/IEDM.2015.7409815 (IEEE, 2015).
18. Shulaker, M., Wei, H., Patil, N., Provine, J. & Chen, H. Linear increases in carbon nanotube density through multiple transfer technique. *Nano Lett.* **11**, 1881–1886 (2011).
19. Won, Y. et al. Zipping, entanglement, and the elastic modulus of aligned single-walled carbon nanotube films. *Proc. Natl Acad. Sci. USA* **110**, 20426–20430 (2013).
20. Kang, S.-M. & Leblebici, Y. *CMOS Digital Integrated Circuits* (Tata McGraw-Hill Education, 2003).
21. Zhang, Z. et al. Doping-free fabrication of carbon nanotube based ballistic CMOS devices and circuits. *Nano Lett.* **7**, 3603–3607 (2007).
22. Shahrjerdi, D. et al. High-performance air-stable n-type carbon nanotube transistors with erbium contacts. *ACS Nano* **7**, 8303–8308 (2013).
23. Ding, L. et al. Y-contacted high-performance n-type single-walled carbon nanotube field-effect transistors: scaling and comparison with Sc-contacted devices. *Nano Lett.* **9**, 4209–4214 (2009).
24. Xu, J.-L. et al. Efficient and reversible electron doping of semiconductor-enriched single-walled carbon nanotubes by using decamethylcobaltocene. *Sci. Rep.* **7**, 6751 (2017).
25. Geier, M. L., Moudgil, K., Barlow, S., Marder, S. R. & Hersam, M. C. Controlled n-type doping of carbon nanotube transistors by an organorhodium dimer. *Nano Lett.* **16**, 4329–4334 (2016).
26. Zhang, J., Wang, C., Fu, Y., Che, Y. & Zhou, C. Air-stable conversion of separated carbon nanotube thin-film transistors from p-type to n-type using atomic layer deposition of high-κ oxide and its application in CMOS logic circuits. *ACS Nano* **5**, 3284–3292 (2011).
27. Markov, I. L., Hu, J. & Kim, M.-C. Progress and challenges in VLSI placement research. *Proc. IEEE* **103**, 1985–2003 (2015).
28. Celio, C., Patterson, D. A. & Asanovic, K. The Berkeley Out-Of-Order Machine (BOOM): an Industry-Competitive, Synthesizable, Parameterized RISC-V Processor. Technical Report No. UCB/EECS-2015-167 (University of California at Berkeley, 2015); http://www2.eecs.berkeley.edu/Pubs/TechRpts/2015/EECS-2015-167.pdf.
29. Patterson, D. 50 Years of computer architecture: From the mainframe CPU to the domain-specific tpu and the open RISC-V instruction set. In *IEEE Int. Solid-State Circuits Conf.* (IEEE, 2018).
30. Lau, C., Srimani, T., Bishop, M. D., Hills, G. & Shulaker, M. M. Tunable n-type doping of carbon nanotubes through engineered atomic layer deposition HfO_x films. *ACS Nano* **12**, 10924–10931 (2018).
31. Brady, G. et al. Polyfluorene-sorted, carbon nanotube array field-effect transistors with increased current density and high on/off ratio. *ACS Nano* **8**, 11614–11621 (2014).
32. Wang, J. et al. Growing highly pure semiconducting carbon nanotubes by electrotwisting the helicity. *Nat. Catal.* **1**, 326–331 (2018).
33. Si, J. et al. Scalable preparation of high-density semiconducting carbon nanotube arrays for high-performance field-effect transistors. *ACS Nano* **12**, 627–634 (2018).
34. Lin, A., Patil, N., Wei, H., Mitra, S. & Wong, H.-S. P. ACCNT—A metallic-CNT-tolerant design methodology for carbon-nanotube VLSI: concepts and experimental demonstration. *IEEE Trans. Electron Dev.* **56**, 2969–2978 (2009).

METHODS

Fabrication process. The fabrication process is shown in Extended Data Fig. 1, and a final fabricated 150-mm wafer is shown in Extended Data Fig. 4. It uses five metal layers and over 100 individual processing steps.

Bottom metal routing layers. The starting substrate is a 150-mm silicon wafer with 800-nm-thick thermal oxide for isolation. The bottom metal wire layers are defined using conventional processing (for example, lithographic patterning, metal deposition, etching, and so on). After the first metal layer is patterned (Extended Data Fig. 1a), an oxide spacer (300 °C) is deposited to separate this first metal layer from the subsequent second metal layer (Extended Data Fig. 1b). To produce interlayer vias between the first and second metal layer, vias are lithographically patterned and etched through this spacer dielectric using dry reactive ion etching (RIE) that stops on the bottom metal layer (Extended Data Fig. 1c). The second metal layer is then defined lithographically and deposited. The vias are formed simultaneously with the second metal wire layer, because the vias are filled during the metal deposition (Extended Data Fig. 1d). RV16X-NANO has two bottom metal layers, which are used for signal routing. The second metal layer also acts as the bottom gate for the CNFETs.

Bottom gate CNFETs. The second metal layer (Extended Data Fig. 1d) provides both signal routing (local interconnect) as well as the bottom gate for the CNFETs. To fabricate the remaining bottom gate CNFET structure, a high-*k* (*k* is the dielectric constant) gate dielectric (a dual-stack of AlO_2 and HfO_2) is deposited through atomic layer deposition (at 300 °C) over the bottom metal gates (Extended Data Fig. 1e). The HfO_2 is used for the majority of the dielectric stack owing to its high-*k* dielectric constant, while the AlO_2 is used for its improved seeding and increased dielectric breakdown voltage. Following gate dielectric deposition, contact vias through the gate dielectric are patterned, and again RIE is used to etch the contact vias, stopping on the local bottom gates (Extended Data Fig. 1f). These contact vias are used by the top metal wiring to contact and route to the bottom gates and bottom metal routing layers. Post-etch, the surface is cleaned with both a solvent rinse as well as oxygen plasma, in preparation for the CNT deposition. Before CNT deposition, the surface is treated with hexamethyldisilazane, a common photoresist adhesion promoter, which improves the CNT deposition (both density and uniformity) over the high-*k* gate dielectric. The 150-mm wafer is then submerged in a toluene-based solution of purified CNTs (similar to the commercial Isosol-100 available from NanoIntegris; http://nanointegris.com/), containing approximately 99.99% semiconducting-CNTs. The amount of time the wafer incubates in the solution, as well as the concentration of the CNT solution, both affect the final CNT density; this process is optimized to achieve approximately 40–60 CNTs per linear micrometre (Extended Data Fig. 1g). Immediately before CNT incubation, the CNT solution is diluted to the target concentration and is horn-sonicated briefly to maximize CNT suspension (importantly, some CNT aggregates will always remain). Post-CNT deposition, we perform the RINSE method (the first step of our MMC) to remove CNT aggregates that deposit on the wafer, leaving CNTs uniformly deposited across the 150-mm wafer. Importantly, RINSE does not degrade the remaining CNTs or remove the non-aggregated CNTs on the wafer (Extended Data Fig. 5). After CNT incubation, we perform the CNT active etch in order to remove CNTs outside the active region of the CNFETs (that is, the channel region of the CNFETs). To do so, we lithographically pattern the active region of the CNFETs (protecting CNTs in these regions with photoresist), and etch all CNTs outside these regions in oxygen plasma. The photoresist is then stripped in a solvent rinse, leaving CNTs patterned only in the intended locations (that is, in the channel regions of the CNFETs) on the wafer (Extended Data Fig. 1h). We use solution-based CNTs here, but an alternative method for depositing CNTs on the substrate is aligned growth of CNTs on a crystalline substrate followed by transfer of the CNTs onto the wafer used for circuit fabrication; both methods have shown the ability to achieve high-drive-current CNFETs[5,17].

MIXED method for CNT CMOS. After the active etch of the CNTs (described in the paragraph above), the p-CNFET source and drain metal contacts are lithographically patterned and defined. We deposit the p-CNFET contacts (0.6-nm-thick titanium for adhesion followed by 85-nm-thick platinum) using electron-beam evaporation, and the contacts are patterned through a dual-layer lift-off process (Extended Data Fig. 1i). This third metal layer acts as both the p-CNFET source contact and the p-CNFET drain contact, as well as the local interconnect. After establishing the p-CNFET source and drain contacts, we passivate the p-CNFETs by depositing 100-nm-thick SiO_2 over only the p-CNFETs (Extended Data Fig. 1j). Following p-CNFET passivation, the wafer undergoes an oxide densification anneal in forming gas (dilute H_2 in N_2) at 250 °C for 5 min. This concludes the p-CNFET fabrication. To fabricate the n-CNFETs, the fourth metal layer (100-nm-thick titanium, n-CNFET source and drain contacts) are defined (Extended Data Fig. 1k, similar to the p-CNFET source and drain contact definition). For the electrostatic doping, nonstoichiometric HfO_x is deposited through atomic-layer deposition at 200 °C uniformly over the wafer. Finally, we lithographically pattern and etch contact vias (Extended Data Fig. 1m) through the HfO_x for metal contacts to the bottom metal layers, and then etch the HfO_x covering the p-CNFETs (the p-CNFETs are protected during this etch by the SiO_2 passivation oxide deposited previously). Additional experimental characterization of the MIXED method (step two of our MMC) is shown in Extended Data Fig. 6.

Back-end-of-line metal routing. Following the CNT CMOS fabrication, conventional back-end-of-line metallization is used to define additional metal layers over the CNFETs (for example, for power distribution and signal routing). As the metal layers below the CNFETs are primarily used for signal routing, we use the top (fifth) metal layer in the process for power distribution (Extended Data Fig. 1n). Additional metal can be deposited over the input/output pads for wire bonding and packaging. At the end of the process, the wafer undergoes a final anneal in forming gas at 325 °C. The finished wafer is diced into chips, and each chip can be packaged for testing or probed for standard cell library characterization.

This 3D physical architecture (with metal routing below and above the CNFETs) is uniquely enabled by the low-temperature processing of the CNFETs. The solution-based deposition of the CNTs decouples the high-temperature CNT synthesis from the wafer, enabling the entire CNFET to be fabricated with a maximum processing temperature below 325 °C. This enables metal layers and the gate stack to be fabricated before the CNFET fabrication takes place. This is in contrast to silicon CMOS, which requires high-temperature processing (for example, >1,000 °C) for steps such as doping activation annealing. This prohibits the fabrication of silicon CMOS over pre-fabricated metal wires, as the high-temperature silicon CMOS processing would damage or destroy these bottom metal layers[35,36].

Experimental measurements. A supply voltage (V_{DD}) of 1.8 V is chosen to maximize the noise resilience of the CNT CMOS digital logic, given the experimentally measured transfer characteristics of the fabricated CNFETs (noise resilience is quantified by the SNM metric (see main-text section 'DREAM'). To interface with each RV16X-NANO chip, we use a high channel count data acquisition system (120 channels) that offers a maximum clock frequency of 10 kHz while simultaneously sampling all channels. This limits the frequency we run RV16X-NANO at to 10 kHz, at which the power consumption is 969 μW (dominated by leakage current). However, this is not the maximum clock speed of RV16X-NANO; during physical design, using an experimentally calibrated CNFET compact model and process design kit in an industry-practice VLSI design flow, the maximum reported clock frequency is 1.19 MHz, reported by Cadence Innovus following placement-and-routing of all logic gates. Future work may improve CNFET-level metrics (for example, improvements in contact resistance, gate stack engineering, CNT density and CNT alignment to increase CNFET on-current) to further speed up clock frequency.

VLSI design methodology. The design flow of RV16X-NANO leverages only industry-standard tools and techniques. We have created a standard process design kit for CNFETs as well as a library of standard cells for CNFETs that is compatible with existing EDA tools and infrastructure without modification. This enables us to leverage decades of existing EDA tools and infrastructure to design, implement, analyse and test arbitrary circuits using CNFETs, which is important to enable CNFET circuits to be widely adopted in the mainstream. This is the first experimental demonstration of a complete process design kit and library for an emerging beyond-silicon nanotechnology.

A high-level description of RISC-V implementation is written in Bluespec and then compiled into a standard RTL hardware description language: Verilog. Bluespec enables testing of all instructions (listed in Extended Data Table 1) written in assembly code (for example, using the assembly language commands) to verify proper functionality of the RV16X-NANO. The functional tests for each instruction are also compiled into waveforms and tested on the RTL generated by Bluespec, they are verified using Verilator to verify proper functionality of the RTL (inputs and outputs are recorded and analysed as value change dump (.vcd) files). RTL descriptions of each module are shown in Fig. 2.

Next is the physical design of RV16X-NANO, including logic synthesis with a DREAM-enforcing standard cell library (see Methods section 'DREAM method implementation'), placement and routing, parasitic extraction, and design sign-off (that is, design rule check, layout versus schematic, verification of the final Graphic Database System, GDSII), as shown in Fig. 4. The RTL is synthesized into digital logic gates using Cadence Genus, using the following components of the CNFET process design kit and standard cell library: the LIBERTY file (.lib) containing power/timing information for all standard library cells, the cell macro library exchange format file (.macro.lef) containing abstract views of all standard library cells (for example, signal/power pin locations and routing blockage information), the technology library exchange format file (.tech.lef) containing metal routing layer information (for example, metal/via width/spacing), and the back-end-of-line parasitic information (.qrcTech file). To enforce DREAM, we use a subset of library cells in the standard cell library, including cells with inverter- and nand2-based logic stages (for combinational logic), and logic stages using tri-state inverters (for sequential logic), as well as fill cells (to connect power rails) and decap cells (to increase capacitance between power rails V_{DD} and V_{SS}); specifically, these 23 cells comprise (see Extended Data Fig. 3): and2_x1, buf_x1, buf_x2, buf_x4, buf_x8, decap_x3, decap_x4, decap_x5, decap_x6, decap_x8, dff2xdlh_x1, fand2stk_x1, inv_x1, inv_x2, inv_x4, inv_x8, inv_x16, mux2nd2_x1, nand2_x1, nor2nd2_x1, or2nd2_x1, xnor2nd2_x1 and xor2nd2_x1. During synthesis, all output pads are buffered with library cell buf_x8 to drive the output pad so that no signal simultaneously drives an output pad as well as another logic stage to prevent excessive capacitive loading in the core. Also, to minimize routing congestion in preparation for place-and-route, the register file (containing four registers, as described in Fig. 2) is directly synthesized from the Verilog hardware description language (instead of being designed 'by hand' or using a memory compiler) so that the D-flip-flops (dff2xdlh_x1: Extended Data Fig. 3) comprising the state elements (registers) can be dispersed throughout the chip to lower the overall total wire length. The final netlist is flattened so there is no hierarchy, and so logic can be optimized across module boundaries, and is then exported for place and route.

Placement-and-routing is performed using Cadence Innovus, loading the synthesized netlist output from Cadence Genus. The core floorplan for standard library cells is defined as 6.912 mm × 6.912 mm. Given the standard cell library and logic gate counts from synthesis (and2_x1: 188, buf_x1: 3, buf_x8: 82, buf_x16: 25, dff2xdlh_x1: 68, fand2stk_x1: 15, inv_x1: 75, inv_x2: 15, inv_x4: 10, inv_x8: 27, mux2nd2_x1: 189, nand2_x1: 625, nor2nd2_x1: 27, or2nd2_x1: 211, xnor2nd2_x1: 14 and xor2nd2_x1: 8), the resulting standard cell placement utilization is 40%. The pad ring for input/output is defined as another cell with 160 pads: 40 on each side, with minimum width 170 μm and minimum spacing 80 μm, totalling pitch 250 μm. Inputs are primarily towards the top of the chip, outputs are primarily on the bottom, and power/ground (V_{DD}/V_{SS}) pads are on the sides (Fig. 1). 1. In addition to the core area, an additional boundary of 640 μm is permitted for signal routing around the core area (containing all standard library cells), for example, for relatively long global routing signals. Placement is performed while optimizing for uniform cell density and low routing congestion. The power grid is defined on top of the core area using the fifth metal layer (as shown in Fig. 1), while not consuming any additional routing resources within the metal layers for signal routing. The clock tree is implemented as a single high-fanout net loaded by all 68 D-flip-flops (for each of CLK and the inverted clock: CLKN), which is directly connected to an input pad, to minimize clock skew variations between registers. All routing signals and vias are defined on a grid, with routing jogs enabled on each metal layer to enable optimization targeting maximum spacing between adjacent metal traces. After this stage of routing, incremental placement is performed to further optimize congestion, and then filler cells and decap cells are inserted to connect the power rails between adjacent library cells and to increase capacitance between V_{DD} and V_{SS} to improve signal integrity. After this incremental placement, the final routing takes place, reconnecting all the signals and routing to the pads, including detailed routing to fix all design rule check violations (for example, metal shorts and spacing violations). Finally, parasitic resistance and capacitances are extracted to finalize the power/timing analysis, and the final netlist is output to quantify the SNM for all pairs of connected logic stages. The GDSII is streamed out from Cadence Innovus and is imported into Cadence Virtuoso for final design rule check and layout versus schematic, using the standard verification rule format files with Mentor Graphics Calibre. The synthesized netlist is again used in the RTL functional simulation environment to verify proper functionality of all instructions, using Synopsys VCS, with waveforms for each test stored in a value change dump (.vcd) file. We note that these waveforms constitute the input waveforms to test the final fabricated CNFET RV16X-NANO, as well as the expected waveforms output from the core, as shown in Fig. 3.

Once the GDSII for the core is complete, it is instantiated in a full die, which contains the core in the middle, alignment marks and test structures (including all standard library cells, CNFETs and test structures to extract wire/via parasitic resistance and capacitance) around the outside of the core as shown in Extended Data Fig. 2. This die (2 cm × 2 cm) is then tiled onto a 150-mm wafer, each of which comprises 32 dies (6 × 6 array of dies minus 4 dies in the corners). Each layer in the GDS is flattened for the entire wafer and then released for fabrication.

DREAM method implementation. To implement DREAM:

1) Generate the DREAM SNM table—for each pair of logic stages in the standard cell library, quantify the susceptibility of the pair to metallic CNTs as follows: use the variation-aware CNFET SNM model (Extended Data Fig. 9) to compute SNM for all possible combinations of whether or not each CNFET comprises an metallic CNT (for example, in a (nand2, nor2) logic stage pair, there are 256 such combinations because there are 8 total CNFETs ($2^8 = 256$)). Record the minimum computed SNM in the DREAM SNM table (Fig. 6b, Extended Data Fig. 9).

2) Determine prohibited logic stage pairs—choose an SNM cut-off value (SNM_C), such that all logic stage pairs whose SNM in the DREAM SNM table is less than SNM_C are prohibited during physical design (see example in Fig. 6b: green entries satisfy SNM_C whereas red entries prohibited cascaded logic gate pairs). The method of choosing SNM_C is described below.

3) Physical design—use industry-practice design flows and EDA tools to implement VLSI circuits without using the prohibited logic stage pairs. Ideally, EDA tools will enable designers to set which logic stage pairs to prohibit during power/timing/area optimization, but this is currently not a supported feature. To demonstrate DREAM in this work, we create a DREAM-enforcing library that comprises a subset of library cells such that no possible combination of cells can be connected to form a prohibited logic stage pair.

To choose SNM_C, we use a bisection search. A larger SNM_C prohibits more logic stage pairs, resulting in better p_{NMS} with higher energy/delay/area cost (and vice versa). To satisfy target p_{NMS} constraints (for example, $p_{NMS} \geq 99\%$), while minimizing cost, we optimize SNM_C as follows. Step 1: Initialize a lower bound L and upper bound U for SNM_C. $L = 0$, and U is the maximum value of SNM_C that enables EDA tools to synthesize arbitrary logic functions (for example, prohibiting all logic stage pairs except (inv, inv) would be insufficient). Step 2: Find p_{NMS} using $SNM_C = (L + U)/2$, using the design flow in Extended Data Fig. 9. Record the set of prohibited logic stage pairs, as well as the circuit physical design, p_{NMS}, energy, delay and area. Step 3: If p_{NMS} satisfies the target constraint (for example, $p_{NMS} \geq 99\%$), set $U = SNM_C$. Otherwise set $L = SNM_C$. Step 4: Set $SNM_C = (L + U)/2$. If p_{NMS} has already been analysed for the resulting set of prohibited logic stage pairs, terminate. Otherwise, return to step 2.

For all physical designs recorded in step 2 we choose the physical design that satisfies the target p_{NMS} constraint with minimum energy/delay/area cost. Importantly, the cost of implementing DREAM is ≤10% energy, ≤10% delay and ≤20% area. To integrate DREAM within EDA tools—enabling p_{NMS} optimization simultaneously with power/timing/area optimization—is a goal for future work on improving p_s versus power/timing/area trade-offs. The effect that the remaining metallic CNTs have on EDP is shown in Extended Data Fig. 7.

Data availability

The data that supports the findings of this study are shown in Figs. 1–6, Extended Data Figs. 1–9, and Extended Data Table 1, and are available from the corresponding author on reasonable request.

35. Batude, P. et al. Advances, challenges and opportunities in 3D CMOS sequential integration. *In IEEE Int. Electron Devices Meet.* https://doi.org/10.1109/IEDM.2011.6131506 (IEEE, 2011).
36. Shulaker, M. et al. Monolithic 3D integration of logic and memory: Carbon nanotube FETs, resistive RAM, and silicon FETs. In *IEEE Int. Electron Devices Meet.* https://doi.org/10.1109/IEDM.2014.7047120 (IEEE, 2014).
37. Clark, L. T. et al. ASAP7: A 7-nm finFET predictive process design kit. *Microelectron. J.* **53**, 105–115 (2016).
38. Zhang, J. et al. Carbon nanotube correlation: promising opportunity for CNFET circuit yield enhancement. In *Proc. 47th Design Autom. Conf.* https://doi.org/10.1145/1837274.1837497 (IEEE, 2010).
39. Sherazi, S. M. et al. Track height reduction for standard-cell in below 5nm node: how low can you go? In *Design-Process-Technology Co-optimization for Manufacturability XII* **10588** 1058809 (International Society for Optics and Photonics, 2018).
40. Hills, G. et al. Rapid co-optimization of processing and circuit design to overcome carbon nanotube variations. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **34**, 1082–1095 (2015).

Acknowledgements We acknowledge Analog Devices, Inc. (ADI), the Defence Advanced Research Projects Agency (DARPA) Three-Dimensional System-on-Chip (3DSoC) program, the National Science Foundation and the Air Force Research Laboratory for support. We thank S. Feindt, A. Olney, T. O'Dwyer, S. Gupta and S. Knepper (all at ADI), and Dimitri Antoniadis and Utsav Banerjee (both at MIT) for collaborations.

Author contributions G.H. performed all VLSI design aspects of this project (developing and analysing DREAM, creating the CNFET process design kit and designing all standard cells in the CNFET library; he performed the entire RV16X-NANO RTL-to-GDS physical design and led experimental calibration and testing). C.L. performed all fabrication aspects of this project (developing and experimentally demonstrating RINSE, developing, experimentally demonstrating and characterizing MIXED; he developed the fabrication process, and fabricated all of the RV16X-NANO wafers and their subsequent packaging to chips). A.W. led the architectural definition of RV16X-NANO (including Bluespec, the Verilog hardware description language and the instruction-set architecture; he also wrote the test programs). S.F. contributed to the architectural definition, system design and implementation. M.D.B., T.S., P.K. and R.H. contributed to developing the fabrication process and establishing the CNFET fabrication flow. A.A. contributed to circuit design. Y.S. and D.M. contributed to project development. A., A.C. and M.M.S. were in charge, advised, and led on all aspects of the project.

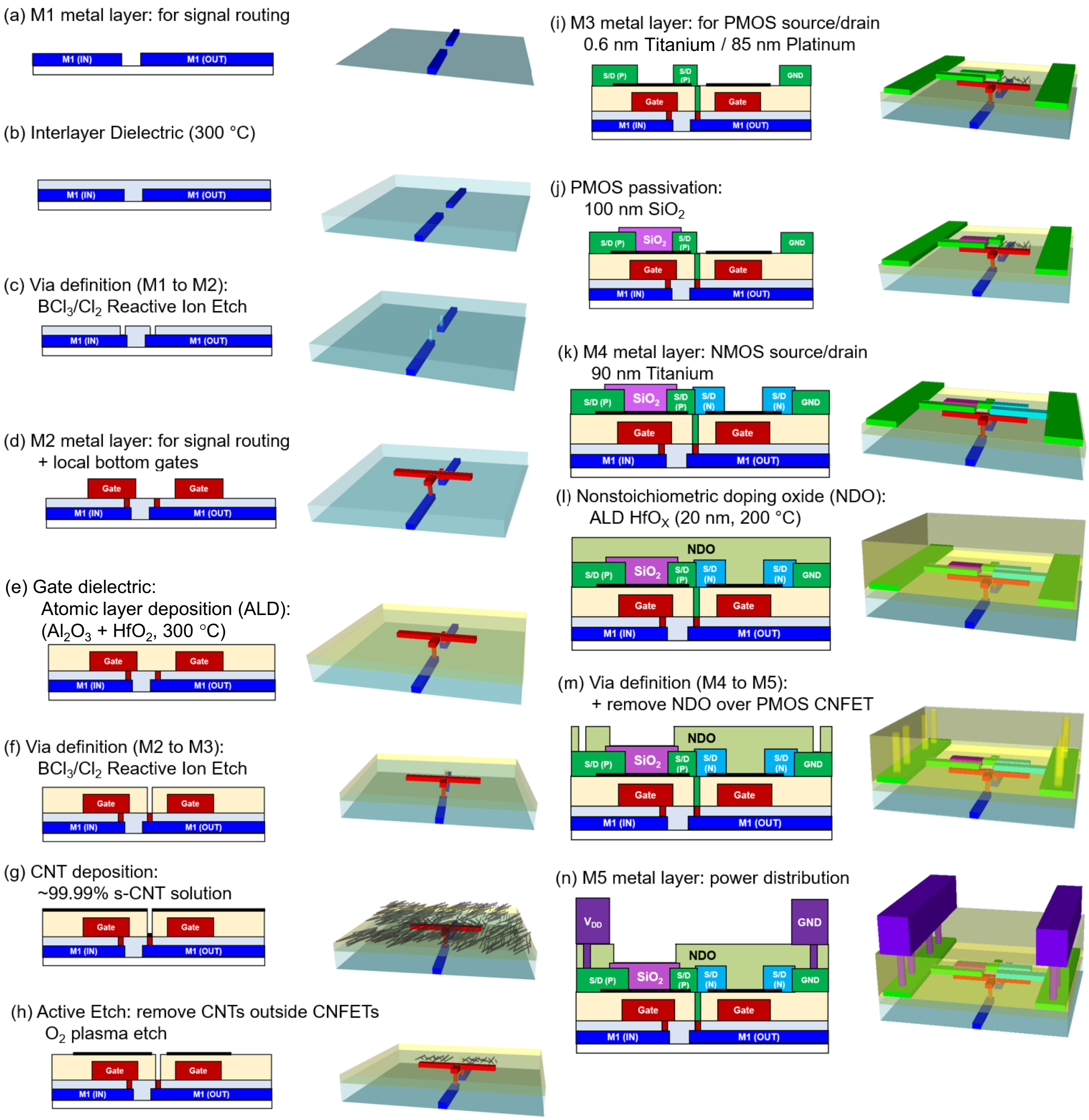
Competing interests A.C. is a board member at Analog Devices, Inc., and this work was sponsored in part by Analog Devices, Inc.

Additional information

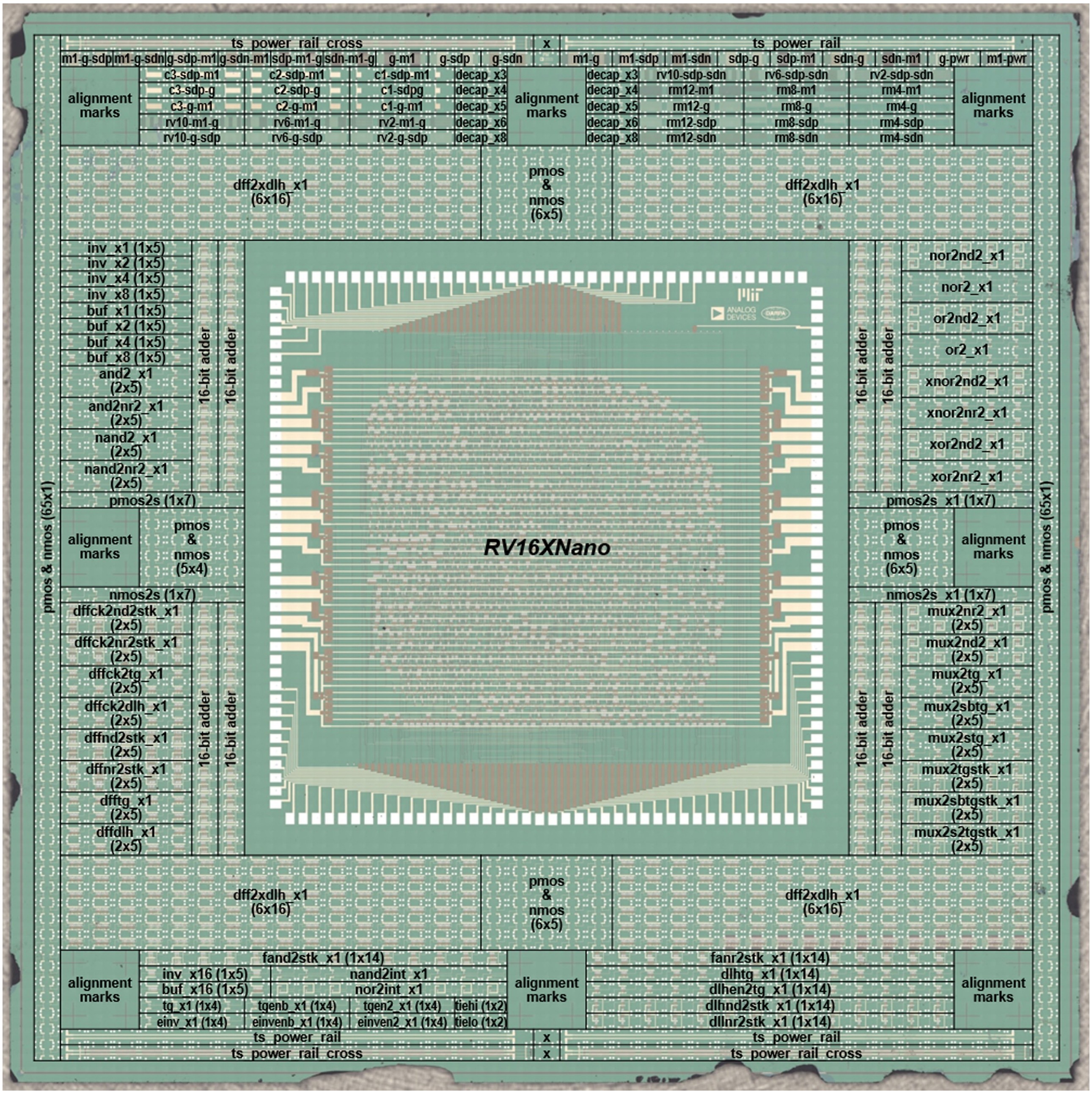
Supplementary information is available for this paper at https://doi.org/10.1038/s41586-019-1493-8.

Correspondence and requests for materials should be addressed to M.M.S.

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

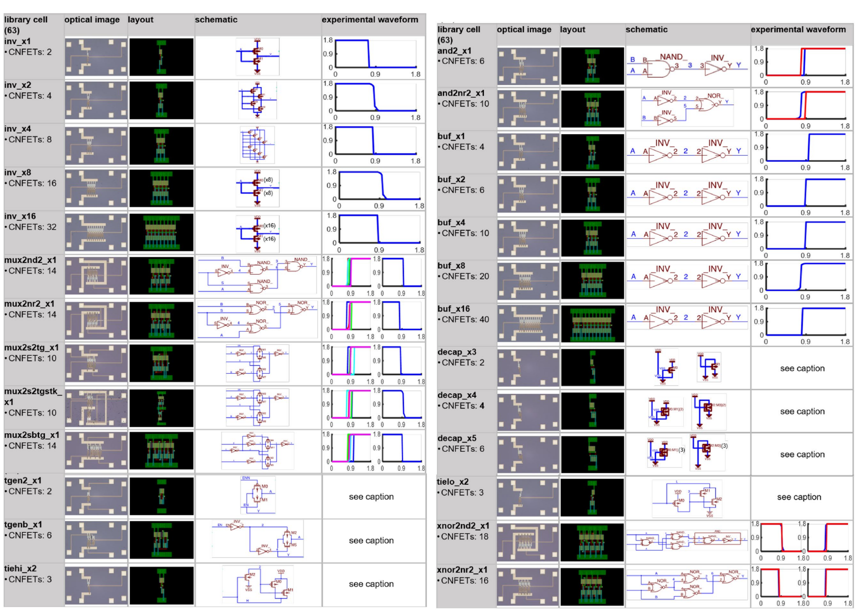


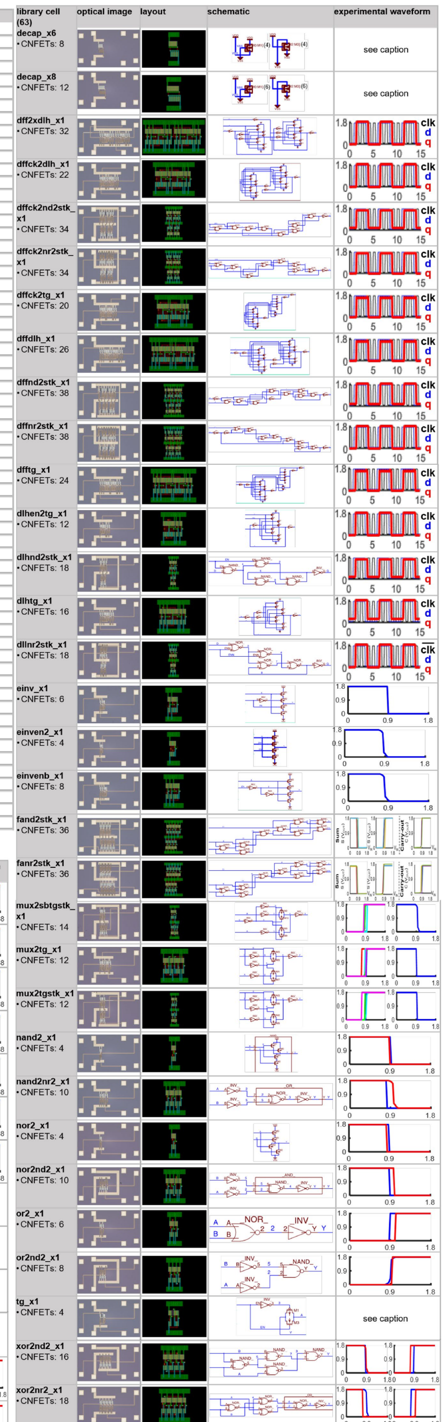
Extended Data Fig. 1 | Fabrication process flow for RV16X-NANO. The fabrication process is a 5-metal-layer (M1 to M5) process and involves >100 individual process steps. s-CNT, semiconducting CNT; S/D, source/drain.



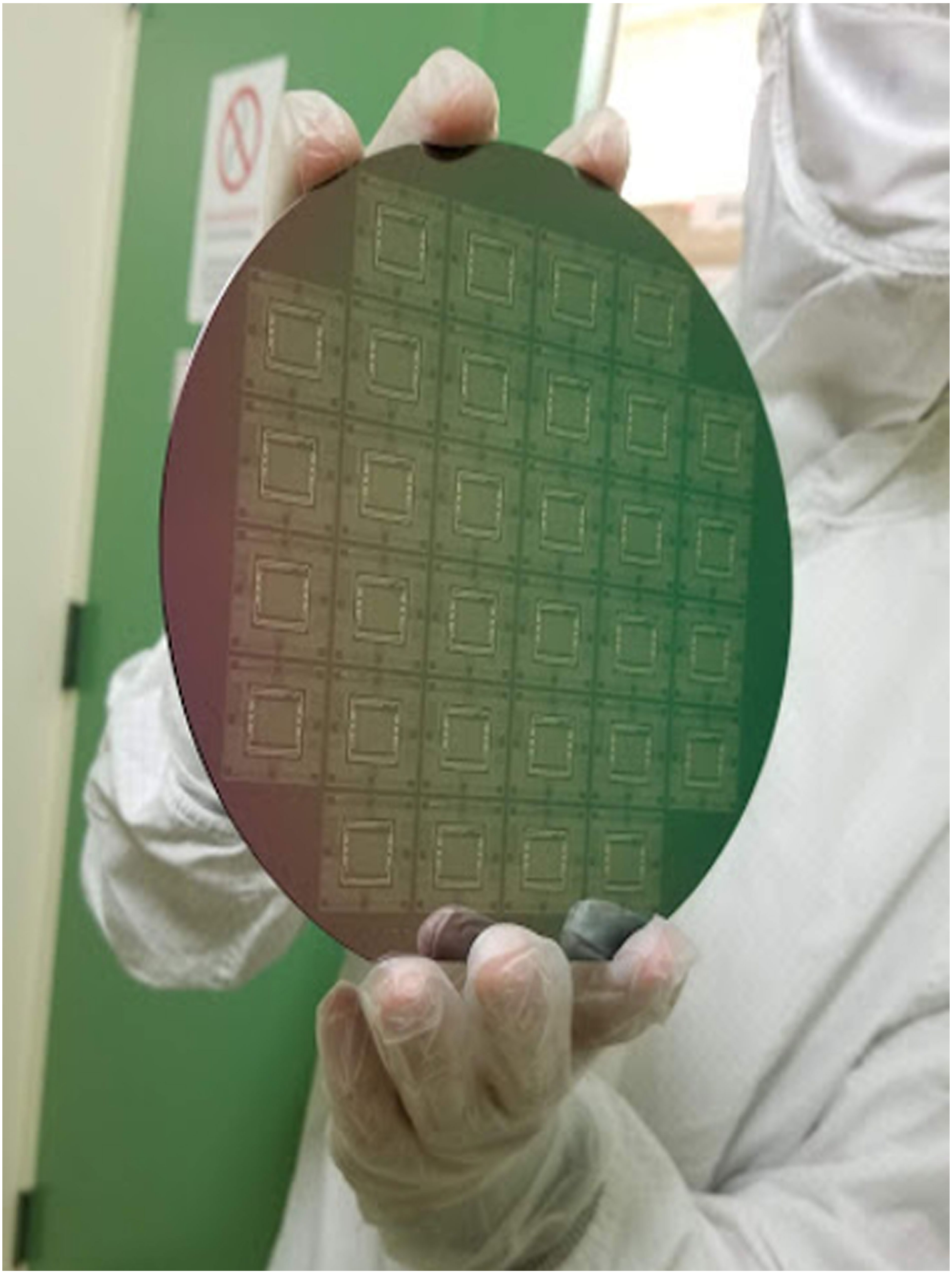
Extended Data Fig. 2 | Microscopy image of a full fabricated RV16X-NANO die. The processor core is in the middle of the die, with test circuitry surrounding the perimeter (when the RV16X-NANO is diced for packaging, these test structures are removed). The test structures include test structures for monitoring fabrication, as well as for measuring and characterizing all of the 63 standard cells in our standard cell library.

library cell (63)	description
and2_x1	2-input AND
and2nr2_x1	2-input AND (comprising nor2/inv logic stages)
buf_x1	buffer, drive strength 1x
buf_x2	buffer, drive strength 2x
buf_x4	buffer, drive strength 4x
buf_x8	buffer, drive strength 8x
buf_x16	buffer, drive strength 16x
decap_x3	capacitance between power rails, size 1x
decap_x4	capacitance between power rails, size 2x
decap_x5	capacitance between power rails, size 4x
decap_x6	capacitance between power rails, size 8x
decap_x8	capacitance between power rails, size 16x
dff2xdlh_x1	positive edge-triggered D-flip-flop (comprising 2x D-latches), input separate clocks for master/slave
dffck2dlh_x1	positive edge-triggered D-flip-flop (comprising 2x D-latches), input clock and inverted clock
dffck2nd2stk_x1	positive edge-triggered D-flip-flop (comprising nand2/inv logic stages), input clock and inverted clock, 2x cell height
dffck2nr2stk_x1	positive edge-triggered D-flip-flop (comprising nor2/inv logic stages), input clock and inverted clock, 2x cell height
dffck2tg_x1	positive edge-triggered D-flip-flop (comprising D-latch and transmission gate), input clock and inverted clock
dffdlh_x1	positive edge-triggered D-flip-flop (comprising 2x D-latches), inverted clock generated locally
dffnd2stk_x1	positive edge-triggered D-flip-flop (comprising nand2/inv logic stages), 2x cell height
dffnr2stk_x1	positive edge-triggered D-flip-flop (comprising nand2/inv logic stages), 2x cell height
dfftg_x1	positive edge-triggered D-flip-flop (comprising D-latch and transmission gate), inverted clock generated internally
dlhen2tg_x1	high-enable D-latch (comprising transmission gates), input enable and inverted enable
dlhnd2stk_x1	high-enable D-latch (comprising nand2/inv logic stages)
dlhtg_x1	high-enable D-latch (comprising transmission gates), inverted enable generated internally
dllnr2stk_x1	high-enable D-latch (comprising nor2/inv logic stages)
einv_x1	tri-state inverter, inverted enable generated internally
einven2_x1	tri-state inverter, input enable and inverted enable
einvenb_x1	tri-state inverter, enable (and inverted enable) buffered internally
fand2stk_x1	full-adder (comprising nand2/inv logic stages)
fanr2stk_x1	full-adder (comprising nor2/inv logic stages)
fill_x1	fill cell (extends power rails), size 1x
fill_x2	fill cell (extends power rails), size 2x
fill_x4	fill cell (extends power rails), size 4x
fill_x8	fill cell (extends power rails), size 8x
fill_x16	fill cell (extends power rails), size 16x
inv_x1	inverter, drive strength 1x
inv_x2	inverter, drive strength 2x
inv_x4	inverter, drive strength 4x
inv_x8	inverter, drive strength 8x
inv_x16	inverter, drive strength 16x
mux2nd2_x1	2-input multiplexer (comprising nand2/inv logic stages)
mux2nr2_x1	2-input multiplexer (comprising nor2/inv logic stages)
mux2s2tg_x1	2-input multiplexer (comprising transmission gates), input select and inverted select
mux2s2tgstk_x1	2-input multiplexer (comprising transmission gates), input selected and inverted select, 2x cell height
mux2sbtg_x1	2-input multiplexer (comprising transmission gates), select (and inverted select) buffered internally
mux2sbtgstk_x1	2-input multiplexer (comprising transmission gates), select (and inverted select) buffered internally, 2x cell height
mux2tg_x1	2-input multiplexer (comprising transmission gates), inverted select generated internally
mux2tgstk_x1	2-input multiplexer (comprising transmission gates), inverted select generated internally, 2x cell height
nand2_x1	2-input NOT-AND
nand2nr2_x1	2-input NOT-AND (comprising nor2/inv logic stages)
nor2_x1	2-input NOT-OR
nor2nd2_x1	2-input NOT-OR (comprising nand2/inv logic stages)
or2_x1	2-input OR
or2nd2_x1	2-input OR (comprising nand2/inv logic stages)
tg_x1	transmission gate, inverted enable generated internally
tgen2_x1	transmission gate, input enable and inverted enable
tgenb_x1	transmission gate, enable (and inverted enable) buffered internally
tiehi_x2	output is tied high (to VDD)
tielo_x2	output is tied low (to VSS)
xnor2nd2_x1	2-input EXCLUSIVE-NOT-OR (comprising nand2/inv logic stages)
xnor2nr2_x1	2-input EXCLUSIVE-NOT-OR (comprising nor2/inv logic stages)
xor2nd2_x1	2-input EXCLUSIVE-OR (comprising nand2/inv logic stages)
xor2nr2_x1	2-input EXCLUSIVE-OR (comprising nor2/inv logic stages)

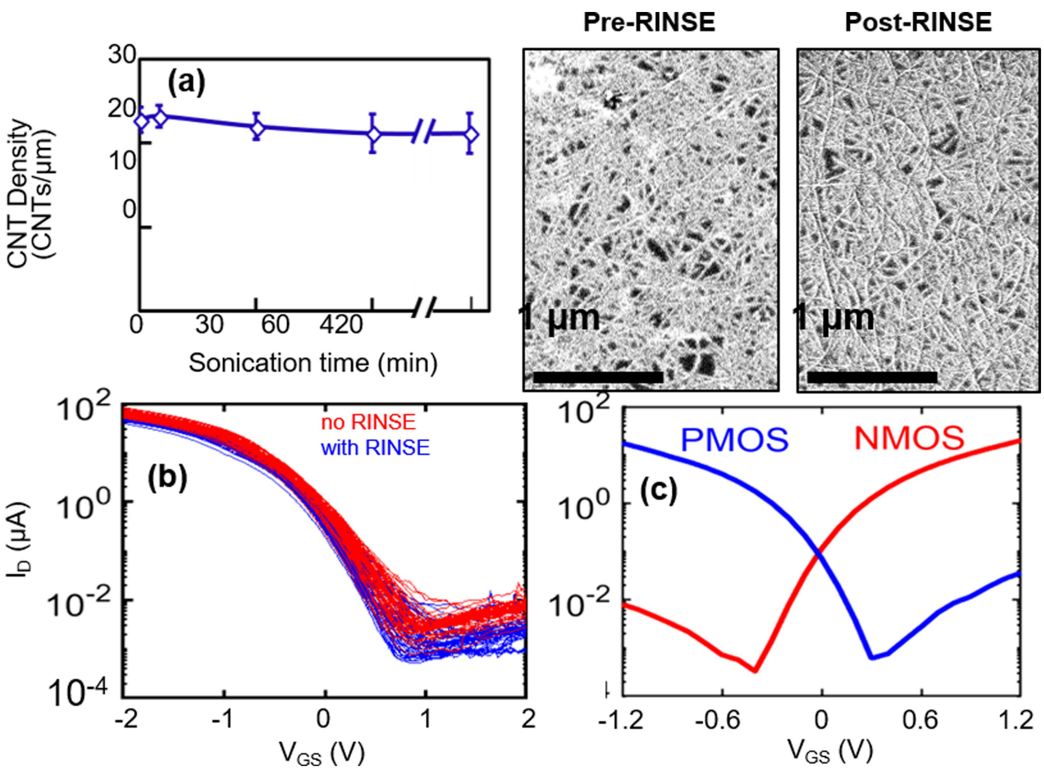




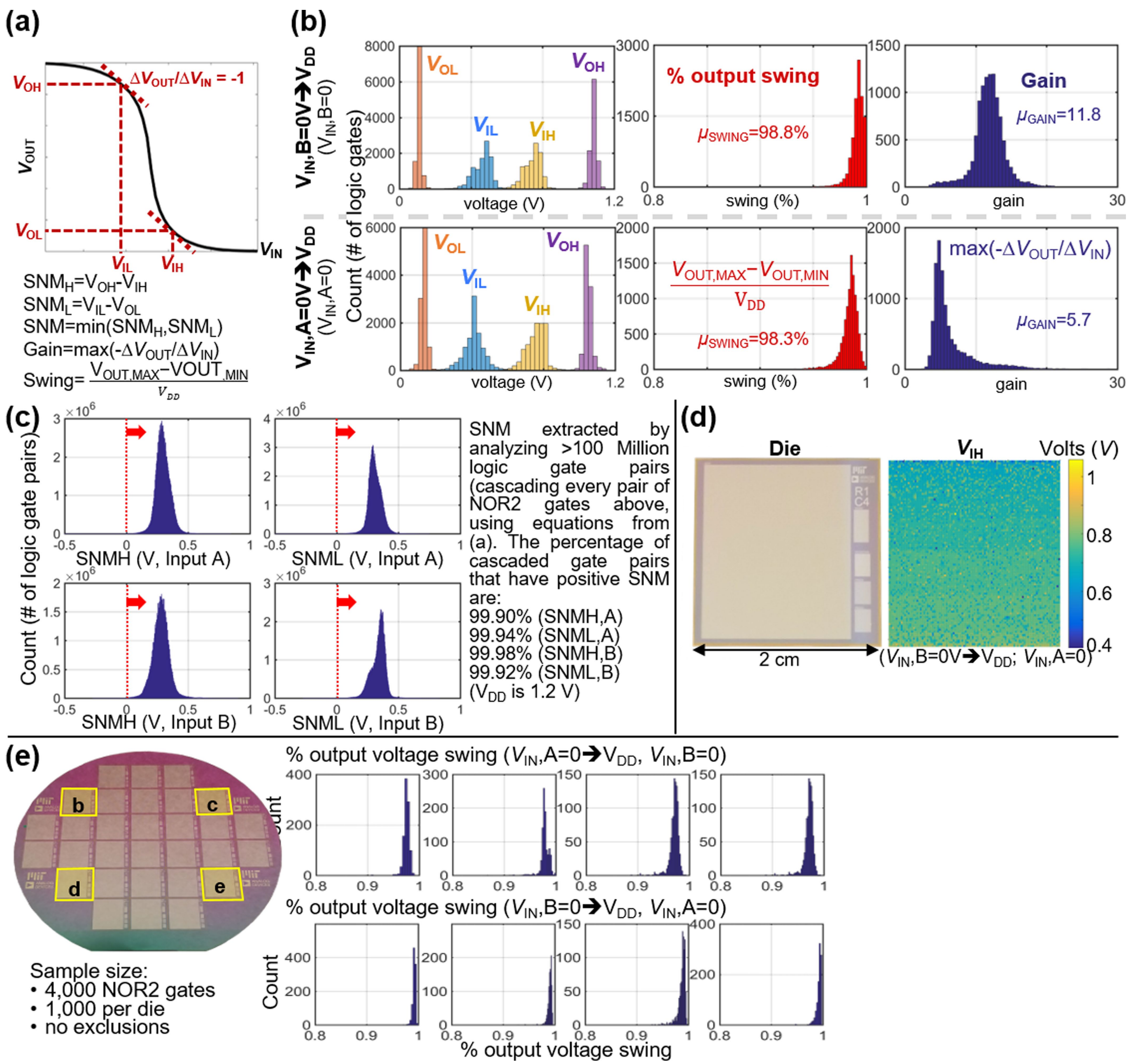
Extended Data Fig. 3 | CNFET standard cell library. List of all of the standard cells comprising our standard cell library, along with a microscopy image of each fabricated standard cell, the schematic of each cell, and a typical measured waveform from each fabricated cell. As expected for static CMOS logic stages, the CNFET logic stages exhibit output voltage swing exceeding 99% of V_{DD}, and achieve gain of >15. Experimental waveforms are not shown for cells whose functionality is not demonstrated by output voltage as a function of either input voltage or time; for example, for cells without outputs (for example, fill cells: cell names that start with 'fill_' or decap cells: cell names that start with 'decap_'), for cells whose output is constant (tied high/low: cell names that start with 'tie_'), or for transmission gates (cell names that start with 'tg_').



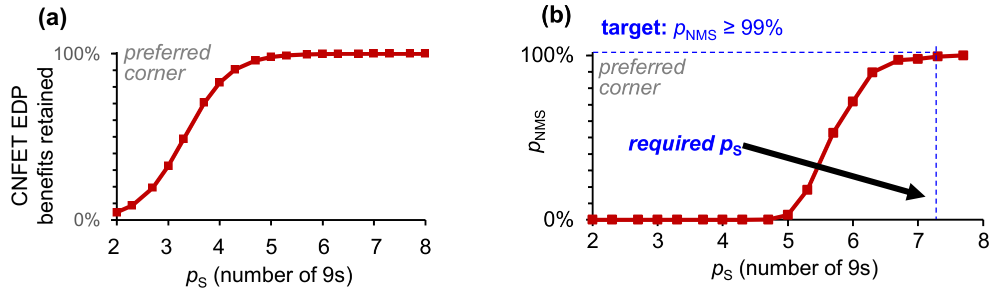
Extended Data Fig. 4 | Image of a completed RV16X-NANO 150-mm wafer. Each wafer includes 32 dies (single die shown in Extended Data Fig. 2).



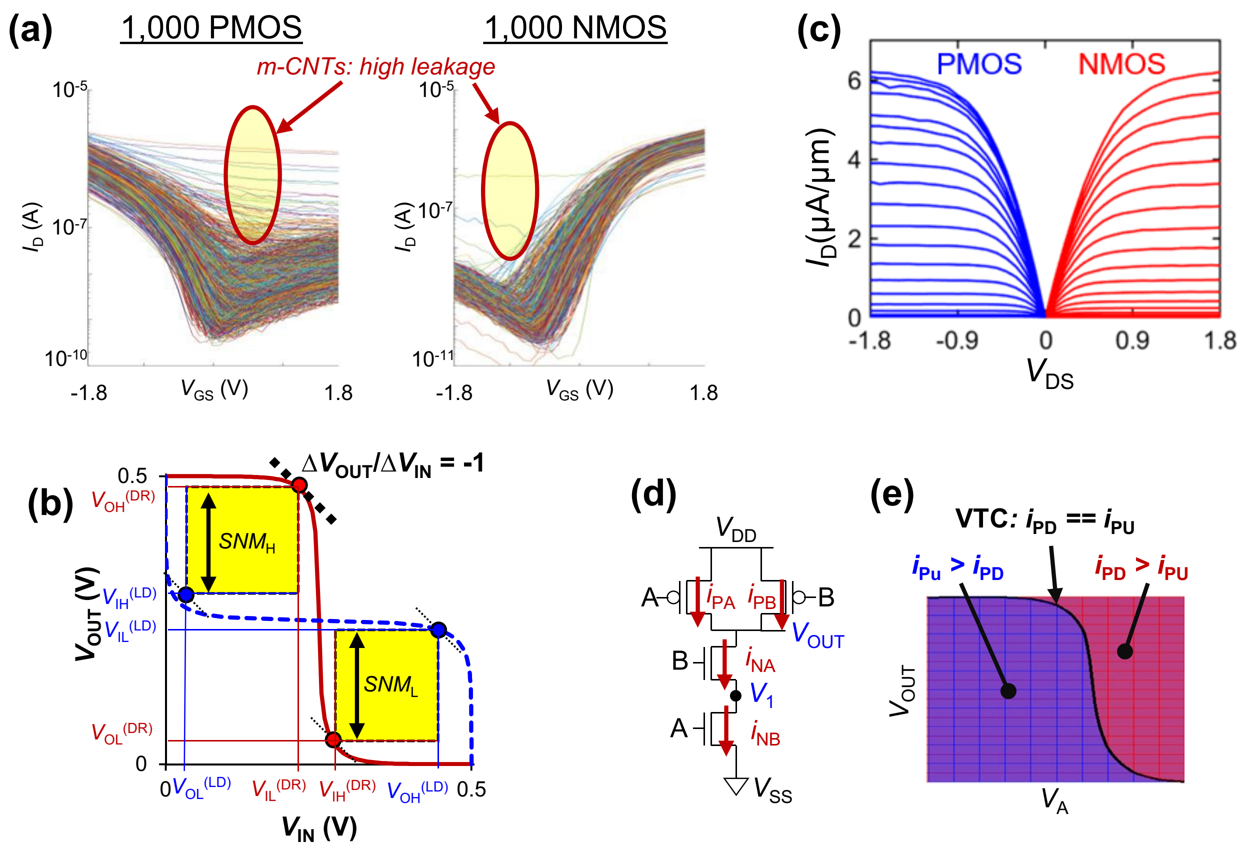
Extended Data Fig. 5 | Negligible effect of RINSE on CNTs and CNFETs. **a**, CNT density is the same pre- versus post-RINSE. **b**, CNFET I_D–V_{GS} exhibit minimal change for sets of CNFETs fabricated with and without RINSE ($V_{DS} = -1.8$ V for all measurements shown). Both samples came from the same wafer, which was diced after the CNT deposition but before the RINSE process. One sample underwent RINSE while the other sample did not. **c**, CNFETs can still be doped NMOS after the RINSE process, leveraging our MIXED process ($V_{DS} = -1.2$ V for all measurements shown).



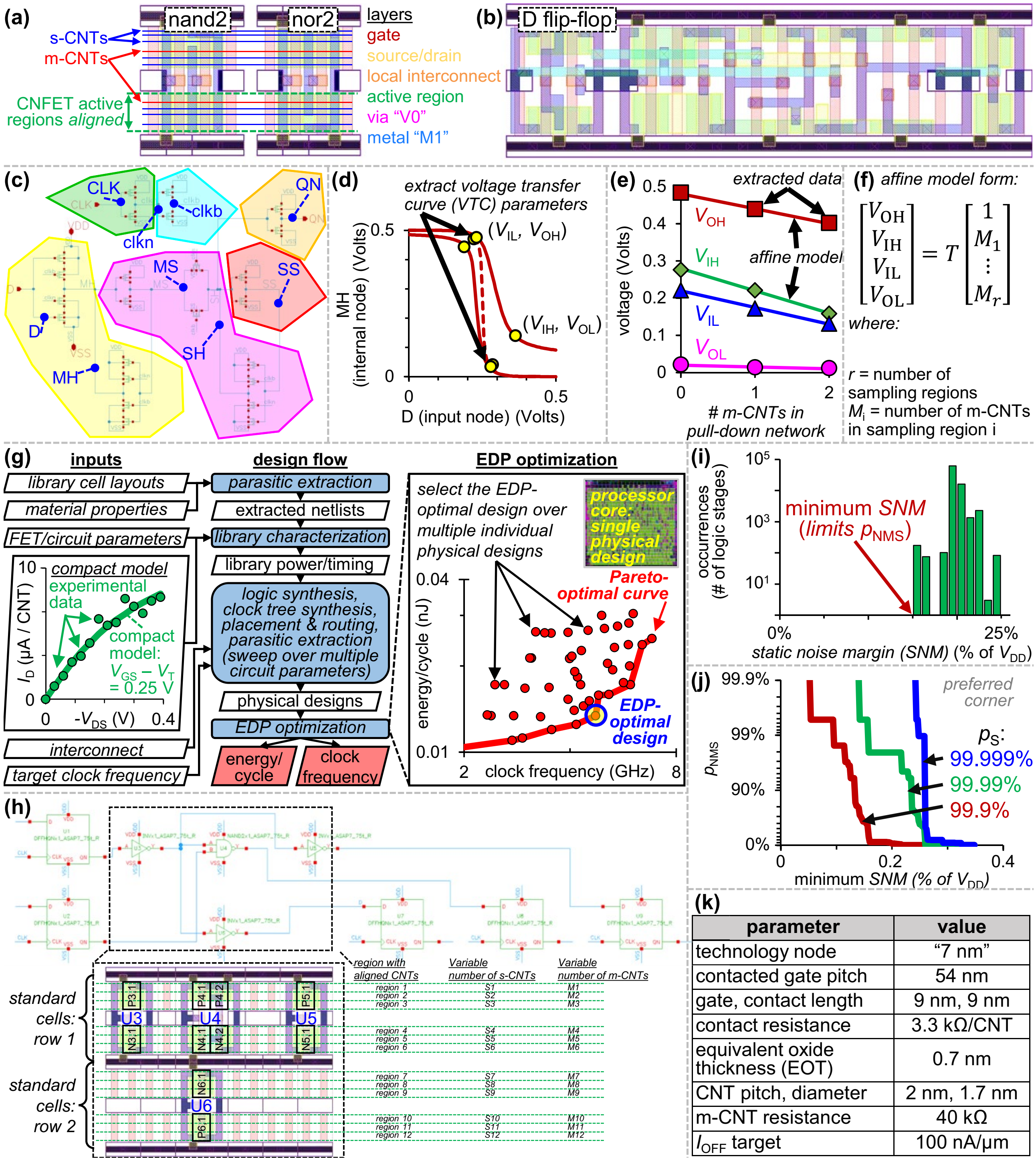
Extended Data Fig. 6 | MIXED CNFET CMOS characterization. **a**, Definitions of key metrics for characterizing logic gates, including SNM, gain and swing. V_{OH}, V_{IH}, V_{IL} and V_{IL} (labelled on the VTCs in **a**, where (V_{IL}, V_{OH}) and (V_{IH}, V_{OL}) are the points on the VTC where $\Delta V_{OUT}/\Delta V_{IN} = -1$) are used to extract the noise margin: SNM = min(SNM_H, SNM_L). **b**, Key metrics extracted for the 10,400 CNFET CMOS nor2 logic gates measured in Fig. 5 (metrics defined in **a**). This is the largest CNT CMOS demonstration to date, to our knowledge. V_{DD} is 1.2 V. **c**, SNM is extracted based on the distributions from **b**. We analyse >100 million logic gate pairs based on these experimental results. **d**, Spatial dependence of V_{IH} (as an example parameter to compute SNM). Each pixel represents the V_{IH} of the nor2 at that location in the die. Importantly, V_{IH} increases across the die (from top to bottom). The change in V_{IH} corresponds with slight changes in CNFET threshold voltage. The fact that the threshold voltage variations are not independently and identically distributed (i.i.d.), but rather have spatial dependence, illustrates that a portion of the threshold voltage variations (and therefore variation in SNM) is due to wafer-level processing-related variations (CNT deposition is more uniform across the 150-mm wafer). Future work should optimize processing steps, for example, increasing the uniformity of the atomic-layer-deposition oxide deposition used for electrostatic doping to further improve SNM for realizing VLSI circuits. **e**, Wafer-scale CNFET CMOS characterization. Measurements from 4 dies across 150-mm wafer (1,000 CNFET CMOS nor2 logic gates are sampled randomly from the 10,400 such logic gates in each die). No outliers are excluded. Yield and performance variations are negligible across the wafer, illustrated by the distribution of the output voltage swing.



Extended Data Fig. 7 | Effect of metallic CNTs on digital VLSI circuits. **a**, Reduction in CNFET EDP benefits versus p_S (metallic CNTs increase I_{OFF}, degrading EDP). $p_S \approx$ 99.999%, sufficient to minimize EDP cost due to metallic CNTs to ≤5%. **b**, p_{NMS} versus p_S (metallic CNTs degrade SNM), (shown for $SNM_R = V_{DD}/5$, and for a circuit of one million logic gates). Although 99.999% p_S is sufficient to limit EDP degradation to ≤5%, panel **b** shows that SNM imposes far stricter requirements on purity: $p_S \approx$ 99.999999% (that is, number of 9s is 8) to achieve $p_{NMS} \geq$ 99% (number of 9s is 2). Results in panels **a** and **b** are simulated for VLSI circuit modules from a 7-nm node processor core (see Supplementary Information and Methods for additional details).



Extended Data Fig. 8 | Methodology to solve VTCs using CNFET *I*–*V* measurements. **a**, Experimentally measured I_D versus V_{GS} for all 1,000 NMOS ($V_{DS} = 1.8$ V) and 1,000 PMOS CNFETs ($V_{DS} = -1.8$ V), with no CNFETs omitted. Metallic CNTs (m-CNTs) present in some CNFETs result in high off-state leakage current ($I_{OFF} = I_D$ at $V_{GS} = 0$ V). **b**, VTC and SNM parameter definitions, for example, for (nand2, nor2). DR is the driving logic stage; LD is the loading logic stage. SNM = min(SNM$_H$, SNM$_L$), where $SNM_H = V_{OH}^{(DR)} - V_{IH}^{(LD)}$ and $SNM_L = V_{IL}^{(LD)} - V_{OL}^{(DR)}$. **c**–**e**, Methodology to solve VTCs (for example, for nand2) using experimentally measured CNFET *I*–*V* curves. **c**, Example I_D versus V_{DS} for NMOS and PMOS CNFETs (V_{GS} is swept from −1.8 V to 1.8 V in 0.1-V increments). **d**, Schematic. To solve a VTC (for example, V_{OUT} versus V_A with $V_B = V_{DD}$): for each V_A, find V_1 and V_{OUT} such that $i_{PA} + i_{PB} = i_{NA} = i_{NB}$ (DC, direct current, convergence). **e**, Current in the pull-up network (i_{PU}, where $i_{PU} = i_{PA} + i_{PB}$, and i_{PA} and i_{PB} are the labelled drain currents of the PMOS FETs gated by A and B, respectively) and current in the pull-down network (i_{PD}, where $i_{PD} = i_{NA} = i_{NB}$, and i_{NA} and i_{NB} are the labelled drain currents of the NMOS FETs gated by A and B, respectively) versus V_{OUT} and V_A. The VTC is seen where these currents intersect. CNFETs are fabricated at a ~1 μm technology node, and the CNFET width is 19 μm in panel **a**.



(k)

parameter	value
technology node	"7 nm"
contacted gate pitch	54 nm
gate, contact length	9 nm, 9 nm
contact resistance	3.3 kΩ/CNT
equivalent oxide thickness (EOT)	0.7 nm
CNT pitch, diameter	2 nm, 1.7 nm
m-CNT resistance	40 kΩ
I_{OFF} target	100 nA/μm

Extended Data Fig. 9 | See next page for caption.

Extended Data Fig. 9 | DREAM implementation and methodology. **a,** Standard cell layouts (derived using the 'asap7sc7p5t' standard cell library[37]), illustrating the importance of CNT correlation: because the length of CNTs (which can be of the order of hundreds of micrometres) is typically much longer compared with the CNFET contacted gate pitch (CGP, for example about 42–54 nm for a 7-nm node[37]), the number of s-CNTs and m-CNTs in CNFETs can be uncorrelated or highly correlated depending on the relative physical placement of CNFET active regions[38]. For many CMOS standard cell libraries at sub-10-nm nodes (for example refs [37,39]), the active regions of FETs are highly aligned, resulting in highly correlated number of m-CNTs among CNFETs in library cells, further degrading VTCs (because one m-CNT can affect multiple CNFETs simultaneously). **b–f**, Generating a variation-aware CNFET SNM model, shown for a D-flip-flop (dff) derived from the asap7sc7p5t standard cell library[37]. **b**, Layout used to extract netlists for each logic stage. **c**, Schematic: CNFETs are grouped by logic stage (with nodes arbitrarily labelled 'D', 'MH', 'MS', 'SH', 'SS', 'CLK', 'clkn', 'clkb' and 'QN' for ease of reference). **d**, For each extracted netlist, there can be multiple VTCs: for each logic stage output, a logic stage input is sensitized if the output state (0 or 1) depends on the state of that input (given the states of all the other inputs). For example, for a logic stage with Boolean function: Y = !(A*B+C), C is sensitized when (A, B) = (0, 0), (0, 1) or (1, 0). We simulate all possible VTCs (over all logic stage outputs and sensitized inputs), and also in the presence of m-CNTs. For example, panel **d** shows a subset of the VTCs for the logic stage in panel **b** with output node 'MH' (labelled in panel **c**), and sensitized input 'D' (with labelled nodes ('clkb', 'clkn', 'MS') = (0, 1, 0)). The dashed line indicates VTC with no m-CNTs, and the solid lines are example VTCs in the presence of m-CNTs (including the effect of CNT correlation). In each case, we model V_{OH}, V_{IH}, V_{IL} and V_{OL} as affine functions of the number of m-CNTs (M_i) in each of r regions (M_1, ..., M_r), with calibration parameters in the static noise margin (SNM) model matrix T (shown in panel **f**). **e**, Example calibration of the SNM model matrix T for the VTC parameters extracted in panel **d**; the symbols are VTC parameters extracted from circuit simulations (using Cadence Spectre), and solid lines are the calibrated model. **f**, Affine model form. **g–j**, VLSI design and analysis methodology. **g**, Industry-practice physical design flow to optimize energy and delay of CNFET digital VLSI circuits, including: (1) library power/timing characterization (using Cadence Liberate) across multiple V_{DD} and using parasitics extracted from standard cell layouts (derived from the asap7sc7p5t standard cell library), in conjunction with a CNFET compact model[8]. (2) Synthesis (using Cadence Genus), place-and-route (using Cadence Innovus) with back-end-of-line (BEOL) wire parasitics from the ASAP7 process design kit (PDK). (3) Circuit EDP optimization: we sweep both V_{DD} and target clock frequency (during synthesis/place-and-route) to create multiple physical designs. The one with best EDP is used to compare design options (for example, DREAM versus baseline). **h**, Subset of logic gates in an example circuit module, showing the effect of CNT correlation at the circuit level (for example, the m-CNT counts of CNFETs P3,1 and P5,1 are both equal to $M_1 + M_2 + M_3$)[40]. **i**, Distribution of SNM over all connected logic stage pairs, for a single sample of the circuit m-CNT counts. The minimum SNM for each trial limits the probability that all noise margin constraints in the circuit are satisfied (p_{NMS}). **j**, Cumulative distribution of minimum SNM over 10,000 Monte Carlo trials, shown for multiple target p_S values, where p_S is the probability that a given CNT is a semiconducting CNT. These results are used to find p_{NMS} versus p_S for a target SNM requirement (SNM_R), where p_{NMS} is the fraction of trials that meet the SNM requirement for all logic stage pairs. We note that p_{NMS} can then be exponentiated to adjust for various circuit sizes based on the number of logic gates. **k**, CNFET compact model parameters (for example, 7-nm node).

Extended Data Table 1 | RISC-V instruction set architecture implementation details

inst	category	summary	assembly
addi	register-immediate arithmetic	add constant, no overflow exception	addi rd, rs1, imm
add	register-register arithmetic	addition with 3 GPRs, no overflow exception	add rd, rs1, rs2
andi	register-immediate arithmetic	bitwise AND with constant	andi rd, rs1, imm
and	register-register arithmetic	bitwise AND with 3 GPRs	and rd, rs1, rs2
auipc	register-immediate arithmetic	load (pc + constant) into GPR	auipc rd, imm
beq	conditional branch	branch if 2 GPRs are equal	beq rs1, rs2, imm
bgeu	conditional branch	branch based on unsigned comparison of 2 GPRs	bgeu rs1, rs2, imm
bltu	conditional branch	branch based on unsigned comparison of 2 GPRs	bltu rs1, rs2, imm
bge	conditional branch	branch based on signed comparison of 2 GPRs	bge rs1, rs2, imm
blt	conditional branch	branch based on signed comparison of 2 GPRs	blt rs1, rs2, imm
bne	conditional branch	branch if 2 GPRs are not equal	bne rs1, rs2, imm
jalr	unconditional jump	jump to relative address, place return address in GPR	jalr rd, rs1, imm
jal	unconditional jump	jump to address, place return address in GPR	jal rd, imm
lh	memory instruction	load short from memory into GPR	lh rd, imm(rs1)
lui	register-immediate arithmetic	load upper bits of constant into GPR	lui rd, imm
ori	register-immediate arithmetic	bitwise OR with constant	ori rd, rs1, imm
or	register-register arithmetic	bitwise OR with 3 GPRs	or rd, rs1, rs2
sh	memory instruction	store short into memory	sh rs2, imm(rs1)
slli	register-immediate arithmetic	shift left logical by constant	slli rd, rs1, imm
sll	register-register arithmetic	shift left logical by GPR value	sll rd, rs1, rs2
sltiu	register-immediate arithmetic	set GPR based on unsigned comparison of GPR and constant	sltiu rd, rs1, imm
slti	register-immediate arithmetic	set GPR based on signed comparison of GPR and constant	slti rd, rs1, imm
sltu	register-register arithmetic	set GPR based on unsigned comparison of 2 GPRs	sltu rd, rs1, rs2
slt	register-register arithmetic	set GPR based on signed comparison of 2 GPRs	slt rd, rs1, rs2
srai	register-immediate arithmetic	shift right arithmetic by constant	srai rd, rs1, imm
sra	register-register arithmetic	shift right arithmetic by GPR value	sra rd, rs1, rs2
srli	register-immediate arithmetic	shift right logical by constant	srli rd, rs1, imm
srl	register-register arithmetic	shift right logical by GPR value	srl rd, rs1, rs2
sub	register-register arithmetic	subtraction with 3 GPRs, no overflow exception	sub rd, rs1, rs2
xori	register-immediate arithmetic	bitwise XOR with constant	xori rd, rs1, rs2
xor	register-register arithmetic	bitwise XOR with 3 GPRs	xor rd, rs1, rs2

inst	format (type -imm)	instruction format																													
		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
addi	**I-I**	imm[11:0]												~~rs1[4:2]~~			rs1		funct3=ADD			~~rd[4:2]~~			rd		opcode=OPIMM						
add	**R**	0	0	0	0	0	0	0	~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=ADD			~~rd[4:2]~~			rd		opcode=OP						
andi	**I-I**	imm[11:0]												~~rs1[4:2]~~			rs1		funct3=AND			~~rd[4:2]~~			rd		opcode=OPIMM						
and	**R**	0	0	0	0	0	0	0	~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=AND			~~rd[4:2]~~			rd		opcode=OP						
auipc	**I-U**	~~imm[31:16]~~																imm[15:12]				~~rd[4:2]~~			rd		opcode=AUIPC						
beq	**S-B**		imm[10:5]						~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=BEQ			imm[4:1]					opcode=BRANCH						
bgeu	**S-B**		imm[10:5]						~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=BGEU			imm[4:1]					opcode=BRANCH						
bltu	**S-B**		imm[10:5]						~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=BLTU			imm[4:1]					opcode=BRANCH						
bge	**S-B**		imm[10:5]						~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=BGE			imm[4:1]					opcode=BRANCH						
blt	**S-B**		imm[10:5]						~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=BLT			imm[4:1]					opcode=BRANCH						
bne	**S-B**		imm[10:5]						~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=BNE			imm[4:1]					opcode=BRANCH						
jalr	**I-I**	imm[11:0]												~~rs1[4:2]~~			rs1		0	0	0	~~rd[4:2]~~			rd		opcode=JALR						
jal	**U-J**		imm[10:1]											~~imm[19:16]~~				imm[15:12]				~~rd[4:2]~~			rd		opcode=JAL						
lh	**I-I**	imm[11:0]												~~rs1[4:2]~~			rs1		funct3=LH			~~rd[4:2]~~			rd		opcode=LOAD						
lui	**I-U**	~~imm[31:16]~~																imm[15:12]				~~rd[4:2]~~			rd		opcode=LUI						
ori	**I-I**	imm[11:0]												~~rs1[4:2]~~			rs1		funct3=OR			~~rd[4:2]~~			rd		opcode=OPIMM						
or	**R**	0	0	0	0	0	0	0	~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=OR			~~rd[4:2]~~			rd		opcode=OP						
sh	**S-S**	imm[11:5]							~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=SH			imm[4:0]					opcode=STORE						
slli	**I-I**	0	0	0	0	0	0	0		imm[3:0]				~~rs1[4:2]~~			rs1		funct3=SLL			~~rd[4:2]~~			rd		opcode=OPIMM						
sll	**R**	0	0	0	0	0	0	0	~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=SLL			~~rd[4:2]~~			rd		opcode=OP						
sltiu	**I-I**	imm[11:0]												~~rs1[4:2]~~			rs1		funct3=SLTU			~~rd[4:2]~~			rd		opcode=OPIMM						
slti	**I-I**	imm[11:0]												~~rs1[4:2]~~			rs1		funct3=SLT			~~rd[4:2]~~			rd		opcode=OPIMM						
sltu	**R**	0	0	0	0	0	0	0	~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=SLTU			~~rd[4:2]~~			rd		opcode=OP						
slt	**R**	0	0	0	0	0	0	0	~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=SLT			~~rd[4:2]~~			rd		opcode=OP						
srai	**I-I**	0	1	0	0	0	0	0		imm[3:0]				~~rs1[4:2]~~			rs1		funct3=SR			~~rd[4:2]~~			rd		opcode=OPIMM						
sra	**R**	0	1	0	0	0	0	0	~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=SR			~~rd[4:2]~~			rd		opcode=OP						
srli	**I-I**	0	0	0	0	0	0	0		imm[3:0]				~~rs1[4:2]~~			rs1		funct3=SR			~~rd[4:2]~~			rd		opcode=OPIMM						
srl	**R**	0	0	0	0	0	0	0	~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=SR			~~rd[4:2]~~			rd		opcode=OP						
sub	**R**	0	1	0	0	0	0	0	~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=ADD			~~rd[4:2]~~			rd		opcode=OP						
xori	**I-I**	imm[11:0]												~~rs1[4:2]~~			rs1		funct3=XOR			~~rd[4:2]~~			rd		opcode=OPIMM						
xor	**R**	0	0	0	0	0	0	0	~~rs2[4:2]~~			rs2		~~rs1[4:2]~~			rs1		funct3=XOR			~~rd[4:2]~~			rd		opcode=OP						

The top panel shows all supported instructions implemented in RV16X-NANO, adhering to RISC-V format specifications for RV32E, with high-level description summary for each. Each instruction is categorized into one of six formats, including instruction type (R-type, I-type, S-type, U-type) and immediate variant (I-immediate, U-immediate, B-immediate, J-immediate, S-immediate), forming one of six formats (type immediate): R, I-I, I-U, S-B, S-S, U-J (shown in the bottom panel). For the assembly code, 'rd' is the destination register, 'rs1' is the source register 1, 'rs2' is the source register 2, 'imm' is immediate. The bottom panel shows the bit-level description of each instruction format. The bottom 7 bits (inst[6:0]) are always the OPCODE, and then the remaining bits are decoded depending on the instruction format (determined by the OPCODE). Values that are crossed out indicate bits that are not used for the 16-bit data path implementation (RV16E) with four registers, instead of 32-bit data path implementation (RV32E) with 16 registers. For example, for instruction 'auipc', only 2 of the 5 reserved bits for 'rd' are required to address the register file for register 'rd' (because there are only $2^2 = 4$ registers instead of $2^5 = 32$), and also the upper 16 bits of the 32-bit immediate (that is, imm[31:16]) are not used because the data path is truncated to 16 bits.