

In the format provided by the authors and unedited.

Modern microprocessor built from complementary carbon nanotube transistors

Gage Hills^{1,3}, Christian Lau^{1,3}, Andrew Wright¹, Samuel Fuller², Mindy D. Bishop¹, Tathagata Srimani¹, Pritpal Kanhaiya¹, Rebecca Ho¹, Aya Amer¹, Yosi Stein², Denis Murphy², Arvind¹, Anantha Chandrakasan¹ & Max M. Shulaker^{1*}

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

Supplemental Information:

Advancement of State-of-the-art:

A detailed summary and comparison with prior work is given below and in Supplemental Table 1. In 2013, progress in the field of CNTs resulted in the first miniature CNT computer. While an important milestone for the technology, there were major and fundamental disconnects between that work and a modern microprocessor today. All FETs were PMOS (it did not use energy-efficient CMOS logic), and it was limited to <60 logic gates totaling 178 CNFETs. Moreover, operationally, it only operated on a single bit of data, and could only execute a signal instruction (the SUBNEG instruction, subtract and branch if negative), and thus avoided many decoding and multiplexing (MUX) operations). While limited, this work demonstrated key advances for the field, such as a complete digital system integrating both combinational logic as well as sequential logic. In stark contrast, this work demonstrates a processor based off of the modern RISC-V processor which is already gaining acceptance in commercial products. The side-by-side comparison of these two works is shown below:

Supplemental Table 1		
	Prior CNT miniature computer	This work
Number of logic gate stages	<60	3,762
Number of transistors (CNFETs)	178	14,702
Logic family	PMOS only	Static CMOS
Design flow	Manual	Industry-standard design flow
Number of bits in data path	1	16
Number of instructions	1 (SUBNEG)	31 (RV32-I)
Number of bits in instruction	10-bit	32-bit

More recently, the largest system fabricated from CNFETs was demonstrated in 2017. This is a three-dimensional (3D) nanosystem, integrating vertical layers of logic, memory, and sensing, and was another important demonstration (illustrating the new monolithic 3D systems enabled by CNTs). However, while this system contained >2 million CNFETs, the vast majority of these CNFETs were used as gas sensors (e.g., not for logic, which are a completely different technology with completely different - *and substantially relaxed* - constraints). Moreover, similar to the prior miniature CNT computer discussed

above, it used PMOS-only only and was not a complete system (e.g., the 3D nanosystem only performed sensing and computing, all control flow was performed off-chip with an external computer).

To avoid significant static (i.e., constant) power dissipation, energy-efficient CMOS (complementary metal-oxide-semiconductor) logic design requires complementary and symmetric FETs (*p*-FETs and *n*-FETs, e.g., for logic design using energy-efficient logic families such as static CMOS, which only consume significant energy primarily during the momentary switching of the logic gate states).

While prior works have demonstrated CNT CMOS, no complete CNT CMOS digital system has ever been fabricated. The most complex CNFET CMOS demonstration resulted in a 4-bit adder, comprising of 132 CNFETs. Our manufacturing methodology for CNTs is essential for transforming CNT CMOS from small-scale demonstrations to complex digital systems such as our RV16X-NANO (RV16X-NANO actually also contains a 16-bit adder within its design within the execute unit, Figure 2).

Work	Description	Type of logic	# of logic gate stages/ # of FETs	Additional points
THIS WORK	RV16X-NANO	Full CMOS	3,762 gates/ 14,702 CMOS CNFETs	See manuscript for details.
[1]	first single-bit, single instruction CNT miniature computer	PMOS	<60 gates/ 178 CNFETs	
[2]	3D sensing chip	PMOS	2 million <i>p</i> -CNFET gas sensors	Used as gas sensors, not as cascaded logic. All <i>p</i> -CNFETs, no CMOS.
[3]	Hyper-dimensional chip	PMOS	1932 <i>p</i> -CNFETs	78% stuck-at faults (faulty logic)
[4]	Hardware accelerator chip	PMOS	768 <i>p</i> -CNFETs	

[5]	Sensor interface circuit	PMOS	20 gates/ 45 <i>p</i> -CNFETs	
[6]	4-bit full adder	CMOS (pass-gate logic)	132 CNFETs	Pass-gate logic
[7]	inverter	CMOS	2	
[8]	SRAM cell	CMOS	6	Not solid state, no yield reported.
[9]	Ring oscillator	CMOS	12	34% yield

[1] Shulaker, M.M., Hills, G., Patil, N., Wei, H., Chen, H.Y., Wong, H.S.P. and Mitra, S., 2013. Carbon nanotube computer. *Nature*, 501(7468), p.526.

[2] Shulaker, M.M., Hills, G., Park, R.S., Howe, R.T., Saraswat, K., Wong, H.S.P. and Mitra, S., 2017. Three-dimensional integration of nanotechnologies for computing and data storage on a single chip. *Nature*, 547(7661), p.74.

[3] Wu, T.F., Li, H., Huang, P.C., Rahimi, A., Hills, G., Hodson, B., Hwang, W., Rabaey, J.M., Wong, H.S.P., Shulaker, M.M. and Mitra, S., 2018. Hyperdimensional Computing Exploiting Carbon Nanotube FETs, Resistive RAM, and Their Monolithic 3D Integration. *IEEE Journal of Solid-State Circuits*.

[4] Hills, G., Bankman, D., Moons, B., Yang, L., Hillard, J., Kahng, A., Park, R., et al. "Trig: hardware accelerator for inference-based applications and experimental demonstration using carbon nanotube FETs." In *Proceedings of the 55th Annual Design Automation Conference*, p. 74. ACM, 2018.

[5] Shulaker, M.M., Van Rethy, J., Hills, G., Wei, H., Chen, H.Y., Gielen, G., Wong, H.S.P. and Mitra, S., 2014. Sensor-to-digital interface built entirely with carbon nanotube FETs. *IEEE Journal of Solid-State Circuits*, 49(1), pp.190-201.

[6] Yang, Y., Ding, L., Han, J., Zhang, Z. and Peng, L.M., 2017. High-performance complementary transistors and medium-scale integrated circuits based on carbon nanotube thin films. *ACS Nano*, 11(4), pp.4124-4132.

[7] Geier, M.L., McMorrow, J.J., Xu, W., Zhu, J., Kim, C.H., Marks, T.J. and Hersam, M.C., 2015. Solution-processed carbon nanotube thin-film complementary static random access memory. *Nature nanotechnology*, 10(11), p.944.

[8] Han, S.J., Tang, J., Kumar, B., Falk, A., Farmer, D., Tulevski, G., Jenkins, K., Afzali, A., Oida, S., Ott, J. and Hannon, J., 2017. High-speed logic integrated circuits with solution-processed self-assembled carbon nanotubes. *Nature nanotechnology*, 12(9), p.861.

[9] Zhang, H., Xiang, L., Yang, Y., Xiao, M., Han, J., Ding, L., Zhang, Z., Hu, Y. and Peng, L.M., 2018. High-Performance Carbon Nanotube Complementary Electronics and Integrated Sensor Systems on Ultrathin Plastic Foil. *ACS nano*, 12(3), pp.2773-2779.

Standard Cell Library

As part of the CNFET PDK, parameterized cells (Pcells) are created using Cadence Virtuoso® for both n-CNEETs and p-CNFETs, with the following open access (OA) layers: CNFET gate, goxcut (via between CNFET gate and source/drain), active, sdp (p-CNFET source/drain), pp (passivation over p-CNFETs to shield from doping oxide for n-CNFETs), sdn (n-CNFET source/drain), dopecut (etch doping oxide over p-CNFETs), and m5 (top-layer metal). CNFET Pcells offer the following user-controlled component description format (CDF) parameters, which are both provided as inputs by the designer, and extracted using design rule check (DRC) and layout vs. schematic (LVS) rules: CNFET width, physical gate length, channel length, gate underlap, source/drain/gate extension width over the horizontal edge of the CNT active region, and source/drain extension length over the vertical edge of the CNT active region. These CDF parameters are also automatically checked using DRC, along with other design rules such as minimum spacing between CNT active region and goxcut via (connecting layers: gate and sdp). CNFET devices extracted using LVS (using Standard Verification Rule Format (SVRF) compatible with Mentor Graphics Calibre® integrated within Cadence Virtuoso®) instantiate CNFETs using the widely-used virtual source FET model calibrated to experimentally measured data from CNFETs (compact model written in Verilog-A).

The CNFET Pcell is used in conjunction with other OA layers in the PDK (e.g., m1 = metal routing layer, v1g = via between m1 and CNFET gate, v_sd_m5 = via between NMOS source/drain and metal 5 power distribution) to create 63 cells in the standard library cell; images, layouts, schematics, and experimentally measured waveforms for each standard cell are shown in Extended Data Figure 3. To facilitate automated and compact placement and routing, standard cells are designed with the convention of having standard cell height equal to 16 metal tracks, including one shared power rail (V_{DD}) and one shared ground (V_{SS}) rail (shared between vertically adjacent rows of standard cells), and some cells are “double height cells” comprising 32 metal tracks and conforming to the same conventional site-based placement method used by placement-and-routing tools (e.g., Synopsys IC Compiler® and Cadence Innovus®). Standard library

cells also conform to conventional CMOS-based layout styles with PMOS FETs aligned horizontally (e.g., on the top half of the layout towards the V_{DD} rail) and NMOS FETs aligned horizontally (e.g., on the bottom half of the layout toward the V_{SS} rail). Of the 16 metal tracks for standard cells, 3 are used for power rails, 3 are used for each of n-CNFET and p-CNFET width, and 7 are used for signal routing (2 tracks between p-CNFET and V_{DD} , 2 tracks between n-CNFET and V_{SS} , and 3 tracks between n-CNFET and p-CNFET), creating routing resources for local interconnects (e.g., within standard cells and between standard cells), and for global signal routing (for place-and-route tools).

Layouts for each cell are shown in Extended Data Figure 3, as well as schematics for each standard cell that are automatically extracted using LVS (and verified against drawn schematics in Cadence Virtuoso®). These netlists are then used in conjunction with Cadence Liberate® for library power/timing characterization, e.g., to compute rise/fall delay/output slew rate/energy consumption as functions of output load capacitance and input slew rate. Additional metrics quantified by Cadence Liberate® include cell leakage power and timing constraint tables for sequential logic (e.g., setup time, hold time, minimum clock pulse width). The results are written to standard file formats (LIBERTY file: .lib) for synthesis & place-and-route. In addition to .lib files, library exchange format (.lef) files (defining abstracted standard cell views for place and route, including pin locations, power rail locations, and blockage information) are created using Cadence Abstract®. These .lib and .lef files for standard library cells are then used for synthesis & place and route, in conjunction with other technology files in the PDK (.lef for technology layers, e.g., defining layers and rules for metal routing, and .ict/.qrcTech files defining parasitic information between technology layers).

To experimentally characterize and calibrate the standard cell library, we fabricate each of the standard cells alongside the RV16X-NANO. As shown in Extended Data Figure 2, each die contains the RV16X-NANO processor core in its center, surrounded by test circuitry on the perimeter. For packaging, these test structures are removed during dicing to isolate the RV16X-NANO processor core. The test circuitry contains both test structures for process monitoring throughout the >100 fabrication process steps utilized for RV16X-NANO, but also contains all of the standard cells. Extended Data Figure 3 shows images, schematics, and measurements for each of the standard cells in the library, illustrating correct functionality, high output voltage swing (>99% V_{DD} swing), and high gain (>15). Output voltage swing is defined as the difference between the maximum output voltage and the minimum output voltage as the input voltage is swept from 0 to V_{DD} . Gain is defined as the maximum value of $|dV_{OUT}/dV_{IN}|$ as V_{IN} is swept from 0 to V_{DD} .

RISC-V: Operational Details

The RV16X-NANO processor is implemented as a finite state machine (FSM) with 3 different states: INST_FETCH (when it is requesting an instruction from memory), EXECUTE (when it is executing an instruction), and LOAD_FINISH (when it is loading data from memory). It comes out of reset initialized in INST_FETCH, with the program counter set to 0, and so it fetches the instruction from address 0 in the memory and advances to the EXECUTE state. Adhering to RISC-V specification, the 32-bit instruction is one of 4 base instruction types (R-type, I-type, S-type, or U-type) with 5 immediate variations within the instruction (I-immediate, U-immediate, S-immediate, B-immediate, J-immediate), forming 6 possible instruction formats (type-immediate): R, I-I, I-U, S-B, S-S, U-J: see Extended Data Table 1 for bit-level descriptions of these instructions. The instruction is decoded as one of these formats in the “decode” block (Figure 2), which then access to the register file (e.g., which registers to read) and provides inputs the execute block. These inputs include: 16-bit values read from the register file, immediate values decoded from the instruction, the current value of the program counter, and control signals to select instructions from the sub-blocks including: add, shift, bitwise-and, bitwise-or, bitwise-xor, less than comparison, equal comparison (Figure 2). The outputs of these control blocks determine which values to write back into registers (e.g., for arithmetic operation such as add/subtract), and values to write back into memory. The execute block also computes the next value of the program counter, e.g., based on either incrementing ($\text{next pc} = \text{pc} + 4$) or otherwise for conditional branch or unconditional jump. For loads and stores, the specified memory access is performed in the EXECUTE state. For loads, the processor continues to the LOAD_FINISH state to write the load response to the register file and then request the next instruction. For stores, the processor continues to the INST_FETCH state to fetch the next instruction. All other instructions fetch the next instruction as part of the EXECUTE state. This mode of operation continues during full execution of programs and adheres to the specification in riscv.org according to RV32E specification, but a reduced number of registers (4 instead of 16) and with 16-bit data words instead of 32-bit data words (all registers in the register file are 16-bit instead of 32-bit), though instructions remain 32-bits. A full list of all instructions is in Extended Data Table 1, which is the full list for the RV32E in the RISC-V specification. To adjust RV32E to 16-bit, the instruction length remains 32 bits, and all data operations are performed on 16-bit values instead of 32-bit values, by truncating the upper-32 bits.

The full 32-bit RISC-V processor from which the 16-bit processor was derived was formally verified using the riscv-formal [1] suite of RISC-V specifications and tools for SMT-based hardware model checking. The riscv-formal suite contains a set of specifications that describe correct behavior of RISC-V processors. These specifications are in the form of checks performed on packets emitted from a trace-

emitting interface added to processors under test called the RISC-V Formal Interface (RVFI). There are specifications for each RISC-V instruction that check individual RVFI packets to make sure each instruction is implemented correctly, e.g., the add instruction correctly implements addition. There are also specifications to ensure the microarchitecture is implemented correctly that check bounded sequences of RVFI packets for necessary properties, e.g., reading a register returns the last value written to it. Both groups of specifications are implemented in Verilog with simple synthesizable code and immediate assertion statements. The specifications in the riscv-formal suite have been verified against the reference RISC-V software simulator Spike [2] and another formal specification written in Haskell [3].

The riscv-formal suite also contains tools to perform SMT-based hardware model checking on each of these specifications using the SymbiYosys framework [4]. A script within the riscv-formal suite produces a SymbiYosys configuration file for each piece of the specification. These configuration files can be used with SymbiYosys to perform bounded model checking for each piece of the specification separately.

To verify our 32-bit RISC-V processor against riscv-formal, we added an RVFI port to emit information about each instruction executed and implemented a wrapper for our processor to match the rest of the interface expected by riscv-formal. We then used the SymbiYosys framework to run bounded model checking with a depth of 30 clock cycles using the Boolector SMT solver [5] resulting in all 42 tests passing. The free variables used by the SMT solver were the memory responses and the register file reset values. This allowed the SMT solver to explore arbitrary sequences of instructions from arbitrary initial states of the register file up to 30 clock cycles. Using a depth of 30 clock cycles is more than sufficient to cover all reachable microarchitectural states of the processor.

[1] Clifford Wolf, riscv-formal, <https://github.com/cliffordwolf/riscv-formal>

[2] RISC-V ISA-sim, <https://github.com/riscv/riscv-isa-sim>

[3] riscv-semantics, <https://github.com/mit-plv/riscv-semantics>

[4] Clifford Wolf, SymbiYosys, <https://github.com/YosysHQ/SymbiYosys>

[5] “Boolector 2.0,” Journal on satisfiability, Boolean Modeling and Computing, vol.9, pp. 53-59, 2015.

RINSE Method Characterization

In addition to demonstrating the ability for RINSE to reduce CNT aggregate defect density by $>250\times$, we further characterize RINSE to show that the RINSE process does not negatively impact CNFET performance. As shown in Extended Data Figure 5, scanning electron microscopy (SEM) images of CNT deposition pre- and post-RINSE show negligible change in the CNT density. Therefore, RINSE does not remove CNTs from the wafer surface, even after >60 minutes of performing RINSE sonication. Moreover, CNFETs fabricated on samples that have and have not undergone RINSE exhibit minimal change in electrical characteristics (Extended Data Figure 5b, note negligible change in the I_D - V_{GS} for a set of CNFETs). Importantly, not only does RINSE not impact the p-CNFETs, the CNFETs can still undergo the MIXED doping process (including the electrostatic doping oxide) to realize n-CNFETs (Extended Data Figure 5c).

MIXED Method Characterization

Extended Data Figure 6 provides additional characterization of the MIXED CNFET CMOS process, realizing the first demonstration of large-area (150 mm substrates), uniform, and high-yield CNFET CMOS logic.

Figure 5b shows the 150 mm wafer with the test patterns (each die has 10,400 CNFET CMOS two-input “not-or” (nor2) logic gates). We fabricate and measure separate nor2 logic gates as it enables characterization of key gate-level metrics, *e.g.* the voltage transfer curves (VTCs, which are used to compute the static noise margin of cascaded logic gates)¹, gain, and output voltage swing. Definitions of these key metrics are shown in Extended Data Figure 6a. Extended Data Figure 6b shows the extracted distributions of these key metrics for all of the experimentally measured VTCs from the 10,400 logic gates shown in Figure 5 (importantly, these are the VTCs for all 10,400 CNFET CMOS nor2 logic gates within a die, no logic gates have been excluded). To illustrate uniformity, the functional yield (output voltage swing $>70\%$) for all logic gates within the die is 10,400/10,400, and the average output voltage swing is $>98\%$ with a standard deviation of $<2\%$. Importantly, by extracting V_{OH} , V_{OL} , V_{IH} , V_{IL} from the VTCs associated with each of the 10,400 logic gates, we are able to calculate the static noise margin (SNM) of all combinations of two cascaded nor2 gates (*i.e.*, with *driving* logic gate and *loading* logic gate). Extended Data Figure 6c shows the distributions of the SNM for the >100 million combinations of cascaded logic gate pairs (*i.e.*, with a *driving logic gate* and a *loading logic gate*): 99.93% of logic gate

¹ VTCs allow extracting V_{OH} , V_{OL} , V_{IH} , V_{IL} (Extended Data Figure 6a), which are essential for analyzing the noise resilience of cascaded digital logic gates.

pairs have positive noise margin (*i.e.*, $V_{OH}^{(DR)} > V_{IH}^{(LD)}$ and $V_{OL}^{(DR)} < V_{IL}^{(LD)}$, where (DR) is for the driving logic gate and (LD) is for the loading logic gate), indicating correct cascaded logic gate functionality. All of this characterization is performed at the scaled supply voltage of $1.2 V_{DD}$.

Moreover, Extended Data Figure 6d shows the noise margin violations ($SNM < 0$) can partially be attributed to the wafer processing (*i.e.*, not the CNTs), and therefore can be improved by optimizing processing (e.g., ALD doping oxide uniformity over the 150 mm wafer). As can be seen in Extended Data Figure 6d, there is a spatial dependence of V_{IH} (as an example parameter to compute SNM; the other parameters exhibit similar spatial dependence): V_{IH} increases across the die (increasing 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 uniformly distributed, 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 such as optimizing uniformity of the ALD oxide deposition used for electrostatic doping to further improve SNM for realizing VLSI circuits.

Additionally, to demonstrate that MIXED is wafer-scalable across 150 mm substrates, Extended Data Figure 6e shows the first 150 mm wafer-scale CNT CMOS characterization: measurements across 4 additional dies distributed across the 150 mm wafer (1,000 CNT CMOS nor2 logic gates are sampled randomly from the 10,000 ensemble in each die, with no logic gates omitted from our analysis) illustrating yield across the 150 mm wafer (total yield across wafer: 14,400 logic gates, 57,600 CNFETs). This is the first demonstration of large-area, uniform, and high-yield CNT CMOS logic, enabled by combining RINSE with our robust MIXED CNT CMOS process.

Prior work for overcoming metallic CNTs

The presence of m-CNTs has been a major obstacle in the field of CNTs since the first CNFET demonstrations over a decade ago. While a wide range of techniques have been developed in response to m-CNTs, no technique achieves the required s-CNT purity for realizing CNFET digital VLSI systems. While previous works have set p_s requirements based on limiting m-CNT-induced leakage power, *no existing works have provided guidelines for p_s based on both increased leakage and degraded SNM due to m-CNTs for physical designs of VLSI circuits*; while 99.999% p_s is sufficient to limit EDP degradation to <5%, Extended Data Figure 7 shows that SNM imposes far stricter requirements on purity: $p_s \sim 99.999,999\%$ to achieve $p_{NMS} \geq 99\%$ (for circuits with one million logic gates).

To quantify the impact of m-CNTs on VLSI circuits, we analyze circuit modules synthesized from the processor core of OpenSparc T2, a large multi-core chip that closely resembles the commercial Oracle/SUN Niagara 2 system [1]; thus, our results account for effects present in realistic VLSI circuits – such as wire parasitics, buffer insertion to meet timing constraints, and SNM for cross-coupled logic stages in sequential logic elements – that are not present in small circuit benchmarks. We leverage standard cell libraries derived from the reference library “asap7sc7p5t” included with the ASAP7 process design kit (PDK) to create physical designs of the OpenSparc modules at an example 7 nm technology node [2], to compare optimized energy, delay, area, and p_s required to achieve $p_{NMS} = 99\%$ for VLSI circuits. The baseline case (“non-DREAM”), which permits all pairs of logic stages, requires $p_s \geq 99.999,996,3\%$ to achieve $p_{NMS} \geq 99\%$ (for $SNM_R = V_{DD}/5$). See Methods for additional details on digital VLSI system analysis.

[1] *OpenSparc T2 Processor Core*. Online: <http://www.opensparc.net/opensparc-t2>.

[2] Clark, Lawrence T., Vinay Vashishtha, Lucian Shifren, Aditya Gujja, Saurabh Sinha, Brian Cline, Chandarasekaran Ramamurthy, and Greg Yeric. "ASAP7: A 7-nm finFET predictive process design kit." *Microelectronics Journal* 53 (2016): 105-115.

Below is a table summarizing s-CNT purity achieved through a range of techniques. None meets the required target of 99.999,999% purity to achieve $p_{NMS} > 99\%$ for circuits with one million logic gates; state-of-the-art purity is 99.99% s-CNT: a factor of over 10,000 $\times$ below the required s-CNT purity. The majority of prior CNT circuits have dealt with m-CNTs through either removing m-CNTs pre- or post-deposition. Pre-deposition techniques include both solution-based sorting to selectively remove m-CNTs while the CNTs are suspended in solution, as well as selective s-CNT growth. Unfortunately, while there has been extensive work on both approaches, s-CNT purity remains at $\sim 99.99\%$. Commercially available purified CNTs also approach 99.99% s-CNT purity; none exceeds this threshold. Alternatively, CNTs can be removed post-CNT deposition. For this approach, an ensemble of s-CNTs and m-CNTs (conventional CNT synthesis yields $\sim 30\%$ m-CNTs) is deposited on the wafer, and the m-CNTs are then selectively removed. Electrical breakdown is performed by biasing the gates of the CNFETs in the off-state (low V_{GS}) and applying a large source-drain bias (high V_{DS}). The s-CNTs do not conduct significant current due to the gate biasing (off-state leakage current), but the m-CNTs do flow significant current (by virtue of being a m-CNTs whose conductance is not significantly modulated by the CNFET gate). With high source-drain bias, the m-CNTs flow sufficient current that they heat up due to Joule self-heating, oxidize and are removed from the circuit (similar to a fuse). However, electrical breakdown (and techniques

leveraging electrical breakdown) requires CNFETs to be fabricated to withstand large breakdown voltages (which requires thicker oxides, for instance, degrading CNFET performance during normal circuit operation), and also removes and degrades s-CNTs inadvertently, further degrading CNFET circuit performance. Variations on electrical breakdown (such as VMR or thermocapillary reflow) require either significant additional processing steps and time, or the use of unconventional and non-silicon CMOS compatible materials.

Work	Technique	Highest Demonstrated Purity	Challenge
This work	DREAM	Leverages DREAM design to enable VLSI systems with commercially-available 99.99% s-CNT purity.	
[1]	Thermocapillary reflow	99.99%	Non-standard processing; insufficient purity
[2]	Electrical breakdown (VMR)	99.99%	Non-standard processing; high voltage damages circuits; inadvertent s-CNT removal and damage while removing m-CNTs; insufficient purity
[3]	Solution-based sorting	99.99% s-CNT	Insufficient purity
[4]	Column chromatography	99.99% s-CNT	Insufficient purity
[5]	Gas phase etching	<98% s-CNT	Insufficient purity, low CNT density
[6]	Selective CNT growth	99.9% s-CNT	Insufficient purity, low CNT density

- [1] Shulaker, M.M., Hills, G., Wu, T.F., Bao, Z., Wong, H.S.P. and Mitra, S., 2015, December. Efficient metallic carbon nanotube removal for highly-scaled technologies. In *Electron Devices Meeting (IEDM), 2015 IEEE International* (pp. 32-4). IEEE.
- [2] Patil, N., Lin, A., Zhang, J., Wei, H., Anderson, K., Wong, H.S.P. and Mitra, S., 2009, December. VMR: VLSI-compatible metallic carbon nanotube removal for imperfection-immune cascaded multi-stage digital logic circuits using carbon nanotube FETs. In *Electron Devices Meeting (IEDM), 2009 IEEE International* (pp. 1-4). IEEE.
- [3] Tang, J., Cao, Q., Tulevski, G., Jenkins, K.A., Nela, L., Farmer, D.B. and Han, S.J., 2018. Flexible CMOS integrated circuits based on carbon nanotubes with sub-10 ns stage delays. *Nature Electronics*, 1(3), p.191.
- [4] Cao, Q., Han, S.J., Tulevski, G.S., Zhu, Y., Lu, D.D. and Haensch, W., 2013. Arrays of single-walled carbon nanotubes with full surface coverage for high-performance electronics. *Nature nanotechnology*, 8(3), p.180.
- [5] Zhang, G., Qi, P., Wang, X., Lu, Y., Li, X., Tu, R., Bangsaruntip, S., Mann, D., Zhang, L. and Dai, H., 2006. Selective etching of metallic carbon nanotubes by gas-phase reaction. *Science*, 314(5801), pp.974-977.
- [6] Wang, J., Jin, X., Liu, Z., Yu, G., Ji, Q., Wei, H., Zhang, J., Zhang, K., Li, D., Yuan, Z. and Li, J., 2018. Growing highly pure semiconducting carbon nanotubes by electrotwisting the helicity. *Nature Catalysis*, p.1.

Performance Cost of Manufacturing Methodology for CNTs

Importantly, the Manufacturing Methodology for CNTs (RINSE, MIXED, and DREAM) overcomes the major obstacles facing a future CNT technology with minor cost in performance or energy efficiency. First, RINSE does not degrade CNFET performance: this is validated experimentally in Extended Data Figure 5. Second, MIXED enables CMOS CNFETs without significant (or costly) changes to transistor structure (*e.g.*, additional FET gates or 3D FET geometries that would increase parasitics), and without imposing additional design rules that would impact circuit layouts (in fact, although it is not discussed in this manuscript, CNFET CMOS can potentially enable more compact circuit layouts vs. silicon, due to the lack of design constraints enforced by n/p-wells, well taps, or doping implants in many silicon-based technologies). Finally, DREAM also enables CNFET EDP benefits to be maintained in the presence of m-CNTs using s-CNT purity that is available today. While a perfect, variation-free CNFET technology can

offer 9.0× EDP benefit vs. silicon at a 7 nm node [3], DREAM would still enable – in the worst case – a 7.4× EDP benefit despite only 99.99% s-CNT purity (i.e., while an ideal CNT technology promises a 9× EDP benefit, a worst-case 10% energy and 10% delay cost reduces this 9× EDP benefit to $9 \times / ((1+0.1) \times (1+0.1)) = 7.4 \times$, since energy and delay are both multiplied by 1.1×). Although we quantify the cost of DREAM in this particular case, importantly, DREAM is a general technique to trade-off circuit noise immunity with energy, delay, and area in the presence of variations (such as m-CNTs). Thus, DREAM offers circuit designers multiple solutions with varying costs and noise immunity improvements to meet design goals. There are consequently multiple paths to further reduce the cost of DREAM: (1) as s-CNT purity continues to improve, the cost of DREAM will naturally improve as well (<10% energy, <10% delay, <20% area, is for only 99.99% s-CNT purity that is already commercially available today, improving s-CNT purity will decrease the cost imposed by DREAM); and (2) our analysis considers DREAM implemented using a DREAM-enforcing standard cell library (as described in Methods), so that standard EDA tools today (even though they are currently unaware of DREAM) can still implement DREAM by avoiding the use of prohibited logic stage during logic synthesis and place-and-route. However, integrating DREAM directly into synthesis and place-and-route algorithms can further optimize power/timing/area while simultaneously meeting noise margin constraints, further reducing any cost associated with DREAM.

Therefore, as discussed in the manuscript, the clock frequency and power consumption of RV16X-NANO is not limited by the techniques described above (the combined impact of all techniques is <10% delay, <10% energy, and <20% area in the worst case). To further improve these metrics, there are several key areas that should be pursued. For instance, our RV16X-NANO is implemented at a relaxed ~1 μm technology node due to the lithographic limitations of an academic fabrication facility (the physical channel length is ~1.5 μm, the minimum width CNFET in the standard cell library 5 μm, and the average CNT length is ~3 μm). Therefore, scaling the CNFET technology (e.g., Dennard Scaling) will provide orders of magnitude improvements in energy efficiency (in fact, EDP benefits of scaling have been experimentally demonstrated for CNFETs at the transistor level in multiple works^{4,5}). In addition, continued transistor-level improvement (e.g., minimizing traps at the CNT-dielectric interface, minimizing CNT-metal contact resistance, scaling gate dielectrics, etc.) should continue to be pursued to further to improve the underlying CNFET technology. We expect that with continued efforts, this manuscript sets the foundation for realizing a high-performance CNFET technology.