Nanoparticle-based computing architecture for nanoparticle neural networks

Sungi Kim, Namjun Kim, Jinyoung Seo, Jeong-Eun Park, Eun Ho Song, So Young Choi, Ji Eun Kim, Seungsang Cha, Ha H. Park, Jwa-Min Nam*

The lack of a scalable nanoparticle-based computing architecture severely limits the potential and use of nanoparticles for manipulating and processing information with molecular computing schemes. Inspired by the von Neumann architecture (VNA), in which multiple programs can be operated without restructuring the computer, we realized the nanoparticle-based VNA (NVNA) on a lipid chip for multiple executions of arbitrary molecular logic operations in the single chip without refabrication. In this system, nanoparticles on a lipid chip function as the hardware that features memory, processors, and output units, and DNA strands are used as the software to provide molecular instructions for the facile programming of logic circuits. NVNA enables a group of nanoparticles to form a feed-forward neural network, a perceptron, which implements functionally complete Boolean logic operations, and provides a programmable, resettable, scalable computing architecture and circuit board to form nanoparticle neural networks and make logical decisions.

INTRODUCTION

The earliest electronic computing machines were only able to run a fixed program, and laborious and extensive physical rewiring and restructuring of the entire machine were required to reprogram such computers. In 1945, von Neumann (1) developed the von Neumann architecture (VNA), which refers to a stored-program computer in which instructions and data are stored in the memory, enabling the execution of a set of instructions, i.e., a program, by design. The VNA processes information by sequentially fetching stored data and the instructions from the memory, which are from the users. Following the operation, the result is stored to generate outputs. Owing to the powerful programmability, the VNA is applied to most modern computers and quantum computing (2).

Molecular computing with nanostructures has the potential to enable a wide array of technologies, including nanoparticle logic gates (3, 4), single-molecule biosensors (5), and logic sensing inside/on living cells (6, 7). However, as with early electronic computing machines, nanostructure-driven molecular computing systems are limited to operating a single program because the function (software) of most nanostructures is defined with their structure (hardware). Therefore, implementing other operations requires extensive redesign and remixing of the nanostructures, and it is often challenging or almost impossible to scale up computing powers. Moreover, the single operation quickly consumes fuel molecules and yields irreversible structural changes in the nanostructures, which substantially hampers reoperation and reversibility.

Lipid bilayers offer a systematic platform for molecules and nanoparticles with lateral mobility, phase transition, and compartmentalization (8–10). We have previously demonstrated a molecular computing platform with nanoparticles on a lipid bilayer as the lipid nanotablet (LNT) (11). Although the interfacial ligand and nanoparticle network designs have enabled modular logic circuit design to a certain extent, functional completeness of the Boolean logic that can express all possible logic operations has not yet been achieved. A predefined set of nanoparticle is required for each logic circuit, indicating that only a fixed logic circuit can be operated in a single LNT. Thus, general computing architecture can largely improve applicability, versatility, and practicality of nanoparticle-driven molecular computing.

Here, we designed and realized the nanoparticle-based VNA (NVNA) for molecular computing on the LNT platform. To create a stored-program device for the facile programming on a molecular computing platform, we applied the VNA with nanoparticles by incorporating the concept of memory that stores molecular information (Fig. 1A). The stored information is then processed once the instruction codes are introduced from a user. We separate hardware and software, and the conceptual separation contributes to the modularity and scalability of information processing in LNT in that only “updating” software allows a user to perform multiple computational tasks without fabricating a device every single time.

RESULTS

Hardware and software in NVNA

The LNT hardware chip is composed of three types of DNA-modified nanoparticles: Nano-Memory (NM), Nano-Floater (NF), and Nano-Reporter (NR) (Fig. 1, A to C). The NM and NR are immobile nanoparticles that function as a molecular information storage device and an output unit, respectively. The NM and NR can be distinguished by their distinct scattering colors of green and blue from gold and silver nanoparticles (Fig. 1C and fig. S1). The mobile nanoparticles, referred to as NFs, freely diffuse and collide with immobile particles. The NFs act as a processing unit by assembling to the NM or NR in response to molecular information stored in NM and external instruction codes (“software”) made of Instruction DNAs.

The software was realized using a set of Instruction DNAs in solution (Fig. 1, A and B). Two types of Instruction DNAs, namely, Trap and Report DNAs, provide instructions for the NFs to bind to either the NM or NRs, respectively, via DNA hybridization. The Instruction DNAs provide different rates for the NF to bind to the NM with different storage state or NR. Thus, a mixture of multiple
Instruction DNAs creates logical decision-making strategies according to the storage condition. Hence, arbitrary logic computing can be programmed on a single LNT hardware chip. As the solution inside the reaction chamber can easily be exchanged with the remaining nanoparticles tethered to the lipid membrane, multiple instructions can be transmitted to the LNT.

The logic operation follows three steps. First, the molecular information is stored on the NM unit via DNA hybridization (Fig. 1B). A single NM particle is a 1-bit memory device, where the bistable state is represented by either "0" or "1" (Fig. 1D). Initially, NM_A (memory for input A) has a null value of "0," NM_{A0}, exposing a single-stranded domain, a_{0}. When the user stores input "1" to NM_A, the Input DNA_A hybridized with NM_A, hiding the a_{0} domain and exposing the a_{1}*. The different single-stranded domains in NM_0 and NM_1 are exploited in the operation step. Second, the logic operation was performed by providing instruction signals in the form of a combination of Instruction DNAs, which initiate competitive nanoparticle–nanoparticle assembly with different kinetics based on the NM state. With the output signal "0," all NFs are trapped to the memory unit, NM. However, if the result of the logic operation is "1," the NFs are assembled to the output unit, reporting NF—NR binding (Fig. 1E). Last, following the single execution, the computing chip can be reset to the initial state for the next operation by adding a reset solution (i.e., low salt and high temperature), which detaches all of the DNA base pairings including the inputs and Instruction DNAs.

Programming strategy using Instruction DNAs
In the NVNA-LNT, a program, i.e., a set of instructions for logic circuits, is composed of a combination of multiple Trap and Report DNAs (Fig. 2A and fig. S2). To render nanoparticles capable of making logical decision, we prepared two types of Trap DNA: NM_{0} Trap DNA and NM_{1} Trap DNA, which bind NFs to the NM_{0} and
NM1, respectively. The hybridization domains in the Trap DNA ($a_0$, $a_1$, and $f_1$) are 14 nucleotides whose melting temperatures are >40°C, allowing for fast trapping. On the other hand, another type of Instruction DNA, namely, Report DNA, was designed to make the NFs bind to the NR much more slowly than the logic-allowed trapping for difference in binding kinetics. To induce fast trapping kinetics as compared with reporting, we optimize the concentration of Instruction DNAs and the density of each nanoparticle (Materials and Methods; Fig. 2A and Fig S2, A and B).

For Report DNA, we introduced an eight-base sequence for domain $r^*$, which has a lower melting temperature of ~16°C. Thus, multivalent links in the Report DNA between the NF and NR are required for stable assembly to one another in this case (fig. S2E). When 1 nM Report DNA was added, the initial binding of the NF to the NR was suppressed with a lag time of ~5 min. Later, the binding kinetics became faster as multiple Report DNA strands started binding to an NF, as represented by an increasing slope over time (fig. S2C). Despite slow initial reporting kinetics, Report DNA enabled more than 85% of the NFs to be assembled to the NR within 30 min. For all nanoparticle reactions, detailed experimental conditions (reaction time, DNA concentrations, and buffer conditions), DNA design (domain length and sequence), and kinetics data ($\tau_{1/2}$, $k_i$) are described in the Supplementary Materials.

Such competitive trapping and reporting behaviors using the difference in binding kinetics can be expressed as an If-Then-Else statement, which first searches whether the If condition satisfies TRUE or FALSE, and then operates the Then or Else statement. The initial search for an If-Then statement is analogous to the function of Trap DNAs that consume NFs with a faster rate, and the Else statement can be represented by the function of the Report DNA that prints output “1” depending on the result of the conditional trapping.

Fig. 2. Software programming strategy using Instruction DNAs. (A) Reaction kinetics of three types of Instruction DNAs. The addition of 8 nM NM0 and NM1 Trap DNAs allows fast logic-allowed trapping (solid lines) of NFs to NM with the “0” and “1” states, respectively, and no or slow logic-forbidden binding (dotted lines). The 1 nM Report DNA addition shows binding of NFs to NRs with a lag time. (B) Programming of NOT gate from an If-Then-Else statement to a combination of Instruction DNAs coding the NNN. (C) NOT gate operation in the LNT. For input “0,” the NF has no specific interaction with M0 and generates NF—NR assemblies (cyan dotted circle) as the output “1” (reporting ratio > 0.2, green box). For DNA input “1” stored in the NM, the NFs are trapped to the NM1 (yellow dotted circle), resulting in the output “0” (reporting ratio = < 0.2, green box).
When a mixture of Trap DNA and Report DNA is introduced to the NVNA-LNT chip, the NF binding generates different results depending on the NM storage states and logic operation, thereby implementing a logical operation. For example, NOT logic gates can be implemented with a molecular state form using a mixture of NM Trap DNA and Report DNA (Fig. 2, B and C). We analyzed all the binding events in the microscopic view (100 μm by 100 μm) and calculated the reporting ratio, [(Reporting)/(Reporting + Trapping)], to reliably identify output “1” and output “0.” The reporting ratio over 0.2 was found to report output “1” for all cases. As the Input DNAs attached to the NM remain thermodynamically stable on the surface of the NM particles for >12 hours (Fig. S3), the stored NM state can be used for the information processing step after washing the input solution from the LNT (12). Given that the architecture easily allows for NOT function as we separated the data storage step from the information processing step, arbitrary logic gates can be executed without dual-rail transition logic by which the system without inverting properties can improve the computing power (13).

It should be noted that a few logically forbidden assemblies between the NF and NM1 with the addition of the NM0 Trap DNA were also observed. Although we designed the Input DNA to be captured on the α0 domain (24 nucleotides in length, Fig. 2A) on the NM with a sufficiently high melting temperature, several detached and unbound α0 domains may still exist on the NM1, owing to the high local concentration and charge repulsion of DNA near the nanoparticle surface. While leaky nonspecific trapping may occur for the reaction kinetics of the NF to the NM1 with the NM0 Trap DNA, the output signal “1” is always reported because the leaky reaction is substantially slower than the reporting kinetics. Another false-positive leak may also exist when early reporting occurs before trapping finishes. To reduce such leaks, we attempted to clearly differentiate the true positive reaction kinetics from this false leaking kinetics within the first 5 min. Overall, considering nonspecific reactions, we set the reporting ratio over 0.2 as output “1” and below 0.2 as output “0.”

### Nanoparticle neural network

As the reaction network between multiple nanoparticles connected by the Instruction DNAs can be represented by a perceptron, which is a type of artificial neural network for a binary classifier, we can expand the programming strategy to construct the nanoparticle neural network (NNN) on the LNT platform (fig. S4) (14). In this case, the NM, NF, and NR correspond to the input, hidden, and output layers, respectively, and the Instruction DNAs correspond to the concept of weights. The way that each floater decides TRUE or FALSE based on the memory state is analogous to the activation function that only activates the hidden layer when the weighted sum of the cross-products between the inputs and weights exceeds a certain threshold. The quantitative values of weights for the NNN are summarized in fig. S4.

Using the NNN, we can implement arbitrary Boolean logic circuits in a scalable and modular manner. We tested the software programmability by executing the functionally complete Boolean logic set for two-bit inputs (table S1). As there are two types of NMs, which store inputs A and B, respectively, the NF clearly differentiates the relative kinetics on three binding channels (two NMs and one NR) with the input combinations, generating the result of logic gates. The logic gates that yield the single output “1” in only one of the four input combinations, such as AND, INH, and NOR, can be implemented using a single-layer perceptron (Fig. 3, A and C, and fig. S5A). Thereafter, we use multiple nodes in the hidden layer to construct a multilayer perceptron by introducing multiple NFs, which have different DNA domains to capture the Trap DNAs but shared the same domain for binding with the Report DNAs (Fig. 3D and fig. S5B). Using the nanoparticle-based multilayer perceptron, the two-input OR, NAND, XOR, and XNOR can be implemented by changing the combination of the Trap DNAs. Owing to the two types of NF, output “1” can have a maximum assembled ratio of 50%. Although the multilayer perceptron exhibited certain density differences between the two NFs, a reporting ratio from 0.2 to 0.6 can represent the output “1,” indicating a notable difference between TRUE and FALSE.

We can calculate the number of nanoparticle nodes required for functional completeness of the Boolean logic operators. Logic circuits dealing with n-bit inputs have 2n input combinations, each of which may have an output of TRUE or FALSE. Thus, the possible number of logic circuits is 22n. Using the Karnaugh map, which simplifies Boolean algebra expressions (15), any n-bit logic circuits can be simplified into OR wiring of 2n−1 AND gates, each of which can be represented with one NF. Therefore, 2n−1 NFs are required to operate arbitrary n-bit logic circuits on the LNT. For example, to form a functionally complete set of all 256 (= 28) logic circuits for 3 (n = 3)–bit input logic circuits, the total particle set of the 3 (= n) NMs, 4 (= 23−1) NFs, and 1 NR with the 25 types of Instruction DNAs (24 Trap DNAs and 1 Report DNA) is required.

### Reset and reusability

As the hardware, relying on the covalently modified nanostructures on a lipid chip, allows for multiple executions by returning back to the initial state, we tested the reset function for reusability (Fig. 3B). We dehybridized all DNA assemblies by exchanging the solution with a 5 mM phosphate buffer at 50°C for 30 min. At such a low salt concentration and high temperature, all the Instruction DNAs added at the logic operation and inputs stored on the NM are detached due to the low melting temperatures of DNA domains at the reset condition. Following the resetting, only thiolated DNAs remained on the nanoparticles, returning to the initial state the next operation. We tested two chips showing multiple YES and NOT gate operations as well as data storage with promising reusability up to four times for the LNT chip.

### Decision-making process

We further exploited the system with a sequential decision-making process using a decision tree by combining the NNN operation with the reset function. The decision tree resembles a flowchart that follows each node with multiple questions from top to bottom based on the result, i.e., YES or NO, to produce a final decision. We executed 2-bit comparators that take two 2-bit inputs (4-bit inputs in total) to compare the quantity of the two inputs (Fig. 4). We first show that the first decision-making node, “Is input AB greater than input BC?” represented with the electric logic circuit diagram can be the NNN diagram with four hidden layer nodes (Fig. 4A). The results of 16 input cases successfully generated output “1” only when input AB was greater than input CD. Although the number of nanoparticle nodes and accompanying complexity of the logic circuit increased, the reaction kinetics remained identical owing to the parallel reactions of the multilayer perceptron. Using the powerful programmability and reset function of the NVNA, it is possible to sequentially operate the 2-bit comparator (Fig. 4B). If
the result of the first operation (AB > CD) were false, we execute the next operation (AB = CD, fig. S6) to determine whether input AB was greater than input CD, input AB was equal to input CD, or input CD was greater than input AB. The three representative inputs for each case (AB > CD, AB = CD, and AB < CD) were tested, identifying the relative quantity between AB and CD with up to two results of “Yes” and/or “No.”

**Fan-out logic gate**
In the LNT, the plasmonic nanoparticle core is critical for computing due to their nanoscale geometric features and optical properties. The time-delayed assembly between the NF and NR with the Report DNA is attributable to the DNA sequence design and multivalent DNA strand interactions on the single nanoparticle surface (16). Furthermore, we can expand the number of outputs by adding other pairs of nanoparticle interactions with different scattering colors (17). With the introduction of blue-scattering NFs along with green-scattering NFs, we executed fan-out logic circuits that operate YES and NOT gates simultaneously with two outputs distinguished by optical signals (fig. S7).

**DISCUSSION**
The nanoparticle perceptron was realized with the NVNA on a lipid chip, and we further exploited the system with a sequential
decision-making process using a decision tree by combining the system with the reset function that allows for reusability. It should be noted that nanoparticle-based computing architecture is now fully built and shown here, and the NNN sets up the stage for truly modular and scalable molecular computing with a single set of nanoparticles and Instruction DNAs. It is notable that we simplified the basic reaction modules by only using the hybridization between single-stranded DNAs with kinetics control, without relying on enzymatic or strand displacement reaction (18). Therefore, the computing strategy in the LNT can be intuitively applicable to other types of molecules, such as proteins, peptides, ionic species, small molecules, and RNA. As the main purpose of the Instruction DNA is to create and control links between nanoparticles with molecular information, other chimeric linkers, such as DNA-antibody or chemical ligand-aptamer conjugates, can serve as the Instruction molecules with binding moieties on both ends (19).

The potential offered by the LNT platform can be expanded in various directions. In the aspect of NNN, instead of using digital values of inputs and weights, varying the weights and inputs to analog values can be realized with the biochemical molecules via controlling assembly kinetics and ligand density/modification, enabling sophisticated neural network operations for molecular pattern recognition (20). It should be noted that the number of NFs required for functional completeness exponentially increases as the number of inputs increases, but we can address this issue by adopting more than one hidden layer by which the fewer number of NFs can cover the same function.

Now, most nanodevices rely on a new design of nanoparticles for a specific function as the structure defines its function in nanotechnology/bioscience. However, we created full diversity in function with a network of single nanoparticles set here. The NVNA enables scalable and practical programmability in a wide range of

Fig. 4. Execution of a 2-bit comparator with decision tree on a single chip. (A) Digital logic circuit and NNN diagram for \(AB > CD\), and operation result of 16 combinations of two 2-bit input \(AB\) and \(CD\). (B) Decision trees for the magnitude comparator. The two-layered tree structure generates three results, indicating the relative magnitude of two 2-bit binary inputs. Four-bit inputs of 1111, 0110, and 1000 result in \(AB = CD\), \(AB < CD\), and \(AB > CD\), respectively. Scale bars, 1 \(\mu m\).
Preparation of SUVs

After sonication cleaning of a 50-ml round bottom flask with 99.5% chloroform (Daejung, South Korea), 97.2 mol% dioleoylphosphatidylcholine, 0.4 mol% biotinylated dioleoylphosphatidylethanolamine (DOPE), and 2.5 mol% poly(ethylene glycol) 1000–DOPE (all three lipids were purchased from Avanti, USA) were mixed in the solution of 99.8% chloroform (Samchun, South Korea). Chloroform solvent was removed via a rotary evaporator, which left a lipid mixture ring film in the round bottom flask. To ensure removal of all solvent was removed via a rotary evaporator, which left a lipid mixture of 99.8% chloroform (Samchun, South Korea). Chloroform solvent was removed via a rotary evaporator, which left a lipid mixture ring film in the round bottom flask. To ensure removal of all solvent was removed via a rotary evaporator, which left a lipid mixture ring film in the round bottom flask. To ensure removal of all

Preparation of SLB chamber

Supported lipid bilayer (SLB) was formed on hydrophilic supporting substrates. We used the vesicle fusion method to get SLBs in the flow chamber. We used a flow chamber consisting of a top slide glass, a Parafilm spacer (4 mm by 50 mm by 200 µm), and a bottom cover glass (both glasses are purchased from Paul Marienfeld GmbH & Co. KG, Germany). The inner volume of the flow chamber is ~40 µl. The top slide glass, which has inlet and outlet holes, and the bottom cover glass were cleaned by 10-min sonication in DIW and 10-min piranha etching in H2SO4/H2O2 (3:1) and were thoroughly rinsed with DIW. To prevent SLB formation on the top glass, we passivated the top slide glass with bovine serum albumin (BSA) (10 mg/ml) in 150 mM NaCl phosphate-buffered saline (1× PBS) for 30 min. The flow chamber was assembled by placing a double layer of Parafilm spacer between the two glasses and heat sealing at 105°C. Next, the SUV solution was diluted to 1 mg/ml in 1× PBS solution and sonicated additionally for 15 min. The SUV solution is injected into the flow chamber and incubated for 40 min to form a lipid bilayer on the bottom of the chamber. The flow chamber was gently washed by injection of DIW (200 µl, twice) and 1× PBS (200 µl, once) and passivated with BSA (30 µg/ml) in 1× PBS for 30 min. Streptavidin (40 nM) in 1× PBS was injected into the flow chamber and incubated for 90 min to modify biotinylated DOPE in SLBs. Last, the flow chamber was washed with 1× PBS (200 µl, twice) and modified with DNA-functionalized nanoparticles for further experiments.

Synthesis and characterization of plasmonic nanoparticles

Spherical gold nanoparticles (AuNPs) (diameter, 50.5 ± 3.5 nm) for green-scattering color were purchased from BBI Solutions (Cardiff, UK). Gold-silver core-shell nanoparticles (Au@Ag NPs) (diameter, 50.3 ± 4.7 nm) for blue-scattering color were synthesized by following the seed-mediated growth method. We prepared cetyltrimethylammonium bromide (CTAB)–capped seeds. CTAB solution (9.75 ml, 100 mM) was mixed with HAuCl4 solution (250 µl, 10 mM). Freshly made ice-cold NaBH4 solution (600 µl, 10 mM) was added quickly to the mixture with vigorous stirring for 3 min. The synthesized 1- to 2-nm seeds were incubated at 27°C for 3 hours before the next step. Gold core nanoparticles (10 nm) were then synthesized with the seeds. CTAC (2 ml, 200 mM), l-ascorbic acid (1.5 ml,100 mM), and the previously prepared CTAB-capped seed solution (50 µl) were sequentially mixed. HAuCl4 solution (2 ml, 0.5 mM) was injected with one shot while the solution was being mixed and incubated at 25°C for 15 min with stirring at 300 rpm. After incubation, the 10-nm gold cores were washed by centrifugation and dispersed in CTAC solution (1 ml, 20 mM). Last, we grew a silver shell on the gold core. We mixed the gold core solution (5 µl, 1.77 pM) with cetyltrimethylammonium chloride (CTAC) solution (100 µl, 100 mM), AgNO3 solution (30 µl, 1 mM), and NH4OH solution (5 µl, 28.0 to 30.0% NH3 basis) sequentially. After the gold core solution and the l-ascorbic acid solution (100 mM) were mixed and heated to 50°C, the l-ascorbic acid solution (50 µl) was quickly injected and rapidly mixed. The solution was washed by centrifugation and dispersed in 1% polyvinylpyrrolidone solution for further oligonucleotide modification. The grown silver shell thickness was ~20 nm. All NPs were characterized by transmission electron microscopy (TEM) (JEM-2100, JEOL Ltd., Japan), ultraviolet-visible (UV-Vis) spectrophotometry (Agilent 8453, Agilent Technologies, USA), and dark-field microscopy (DFM) (Axiocvert 200 M, Carl Zeiss, Göttingen, Germany). TEM imaging was carried out at the National Center for Inter-University Research Facilities (Seoul National University, Seoul, South Korea).

Functionalization of plasmonic nanoparticles

Gold and silver nanoparticles are modified with thiolated DNA oligonucleotides via strong gold-thiol and silver-thiol bond. To cleave the dithiol bond, thiol-modified oligonucleotides (Integrated DNA Technologies, USA) are incubated with dithiothreitol (100 mM) in pH 8.0 phosphate buffer for 1 hour. The DNA was purified through size exclusion chromatography using a NAP-5 column (GE Healthcare, Buckinghamshire, UK), and the concentration of monothiolated DNA was measured by UV-Vis spectroscopy. Nanoparticles (10 fmol) were mixed with the thiolated oligonucleotides (200 pmol), at 0.1% (w/v) sodium dodecyl sulfate (SDS) solution (300 µl), and incubated for 1 hour at 25°C. The ratios of thiolated strands for NF, NM, and NR are summarized in table S2. Three aliquots of 1 M NaCl, 0.1% SDS, and 10 mM phosphate buffer (PB) solution were added with a 1-hour interval to achieve a final concentration of 0.3 M NaCl for NF and NR and 0.25 M NaCl for NM. The solution was sonicated for 10 s after each salt aging and incubated overnight at 25°C. The nanoparticles were centrifuge-washed and redispersed in 10 mM PB solution.

Data storage on NM and NNN execution on the LNT

NF, NM, and NR at a concentration range between 1 and 10 pM in 1× PBS were loaded on the LNT for 10 min to reach a proper density of particles (0.1 to 0.2 µm−2 for NM, ~0.02 µm−2 for NF, and ~0.02 µm−2 for NR) on SLB. We loaded 5- to 10-fold higher density of NMs (0.1 to
0.2 \mu m^{-2}) than that of NRs (~0.02 \mu m^{-2}) for 80% of the NF to be trapped to the NM within 10 min, which is close to diffusion-controlled reaction. To achieve trapping faster than reporting, we loaded the density of NM 5 to 10 times higher than that of NF. To fully store molecular information of Input DNAs on NM, we incubated each Input DNA at 50 nM concentration for 30 min and washed with 1x PBS. The mixture of Instruction DNAs (8 nM Trap DNAs and 1 nM Report DNA) in 1x PBS was injected to the chamber. The nanoparticle logic circuit operation was monitored via DFM for 30 min. The sequences of Instruction DNAs are summarized in table S3.

To reset the LNT and reuse for the next operation, all the hybridized Input DNAs and the Instruction DNAs need to be removed from the nanoparticles. We increased the temperature and decreased the salt concentration of the solution over the melting temperature to make the solution temperature greater than the melting temperature of DNA hybridization. PB solution (5 mM) was injected, and the LNT chamber was incubated at 50°C for 30 min to detach the hybridized DNAs. We then washed with PB solution (5 mM, 50°C). For the next operation, 1x PBS solution was injected to recover the salt concentration and temperature.

Analyzing dark-field time-lapse images
To analyze the obtained dark-field time-lapse images, we used the previously developed custom MATLAB code from our laboratory. The images were registered with the StackReg plugin in Imagej, choosing the area of interest (100 \mu m by 100 \mu m) and correcting the movement, and all the particles within the area were analyzed. The image sequences were processed by an image analysis algorithm that classifies NPs into NRs, NFs, and NMs. The nanoparticle assemblies were counted by monitoring scattering color change at the position of NR and NM (immobile NPs) upon NF binding.

SUPPLEMENTARY MATERIALS
Supplementary material for this article is available at http://advances.sciencemag.org/cgi/content/full/6/35/eabb3348/DC1

REFERENCES AND NOTES
1. J. von Neumann, First draft of a report on the EDVAC. IEEE Ann. Hist. Comput. 15, 27–75 (1993).
2. M. Mariantoni, H. Wang, T. Yamamoto, M. Neeley, R. C. Bialczak, Y. Chen, M. Lenander, E. Lucero, A. D. O’Connell, D. Sank, M. Weides, J. Wenner, Y. Yin, J. Zhao, A. N. Korotkov, A. N. Cleland, J. M. Martinis, Implementing the quantum von Neumann architecture with superconducting circuits. Science 334, 61–65 (2011).
3. T. Song, H. Liang, Synchronized assembly of gold nanoparticles driven by a dynamic DNA-fueled molecular machine. J. Am. Chem. Soc. 134, 10803–10806 (2012).
4. M. P. Nikitin, V. O. Shipunova, S. M. Deyev, P. I. Nikitin, Biocomputing based on particle disassembly. Nat. Nanotechnol. 9, 716–722 (2014).
5. Y. Zhang, Z. Shuai, H. Zhou, Z. Luo, B. Liu, Y. Zhang, L. Zhang, S. Chen, J. Chao, L. Weng, Q. Fan, C. Fan, W. Huang, L. Wang. Single-molecule analysis of microRNA and logic operations using a smart plasmatic nanobioenvosensor. J. Am. Chem. Soc. 140, 3988–3993 (2018).
6. R. Yue, Z. Li, G. Wang, J. Li, N. Ma, Logic sensing of MicroRNA in living cells using DNA-programmed nanoparticle network with high signal gain. ACS Sens. 4, 250–256 (2018).
7. R. Peng, X. Zheng, Y. Lyu, L. Xu, X. Zhang, G. Ke, Q. Liu, C. You, S. Huan, W. Tan, Engineering a 3D DNA-Logic gate nanomachine for bispecific recognition and computing on target cell surfaces. J. Am. Chem. Soc. 140, 9793–9796 (2018).
8. Y. Lyu, C. Wu, C. Heinke, D. Han, R. Cai, I. T. Teng, Y. Liu, H. Liu, X. Zhang, Q. Liu, W. Tan, Constructing smart protocols with built-in DNA computational core to eliminate exogenous challenge. J. Am. Chem. Soc. 140, 6912–6920 (2018).
9. S. Kim, J. Seo, H. H. Park, N. Kim, J. W. Oh, J. M. Nam, Plasmonic nanoparticle-interfaced lipid bilayer membranes. Acc. Chem. Res. 52, 2793–2805 (2019).
10. M. Langecker, V. Arnaut, J. List, F. C. Simmel, DNA nanostructures interacting with lipid bilayer membranes. Acc. Chem. Res. 47, 1807–1815 (2014).
11. J. Seo, S. Kim, H. H. Park, D. Y. Choi, J.-M. Nam, Nano-bio-computing lipid nanotablet. Sci. Adv. 5, eaau2124 (2019).
12. P. S. Randeria, M. R. Jones, K. L. Kohlstedt, R. J. Banga, M. Olvera De La Cruz, G. C. Schatz, C. A. Mirkin, What controls the hybridization thermodynamics of spherical nucleic acids? J. Am. Chem. Soc. 137, 3486–3489 (2015).
13. T. Song, A. Eshra, S. Shah, H. Bui, D. Fu, M. Yang, R. Mokhtar, J. Reif, Fast and compact DNA logic circuits based on single-stranded gates using strand-displacing polymerase. Nat. Nanotechnol. 14, 1075–1081 (2019).
14. Y. Freund, R. E. Schapire. Large margin classification using the perceptron algorithm. Mach. Learn. 37, 277–296 (1999).
15. M. Karnaugh, The map method for synthesis of combinational logic circuits. Trans. Am. Inst. Electr. Eng. Part I Commun. Electron. 72, 593–599 (1953).
16. C. R. Laramy, M. N. O’Brien, C. A. Mirkin, Crystal engineering with DNA. Nat. Rev. Mater. 4, 201–224 (2019).
17. S. Kim, J. E. Park, W. Hwang, J. Seo, Y. K. Lee, J. H. Hwang, J. M. Nam, Optokinetically encoded nanoprobe-based multiplexing strategy for microRNA profiling. J. Am. Chem. Soc. 139, 3559–3566 (2017).
18. L. Qian, E. Winfree, Scaling up digital circuit computation with DNA strand displacement cascades. Science 332, 1196–1201 (2011).
19. S. A. Kazane, D. Sok, E. H. Cho, M. Loressa, P. Kuhn, P. G. Schultz, V. V. Smider, Site-specific DNA-antibody conjugates for specific and sensitive immuno-PCR. Proc. Natl. Acad. Sci. U.S.A. 109, 3731–3736 (2012).
20. K. M. Chen, L. Qian, Scaling up molecular pattern recognition with DNA-based winner-take-all neural networks. Nature 559, 370–376 (2018).
21. C. Liu, J. Zhao, F. Tian, L. Cai, W. Zhang, Q. Feng, J. Chang, F. Wan, Y. Yang, B. Dai, Y. Cong, C. Ding, J. Sun, W. Tan, Low-cost thermophoretic profiling of extracellular-vesicle surface proteins for the early detection and classification of cancers. Nat. Biomed. Eng. 3, 183–193 (2019).
22. Y. Yang, Y. Song, X. Bo, J. Min, O. S. Pak, L. Zhu, M. Wang, J. Tu, A. Kogan, H. Zhang, T. K. Hsiai, Z. Li, W. Gao, A laser-engraved wearable sensor for sensitive detection of uric acid and tyrosine in sweat. Nat. Biotechnol. 38, 217–224 (2019).
23. T. Wiegand, M. Fratini, F. Frey, K. Yierentant, Y. Li, E. Weber, K. Galiar, I. Ohmes, F. Braun, D. P. Herten, S. Boullant, U. S. Schwarz, K. Salaata, E. A. Cavalcanti-Adam, J. P. Spatz, Forces during cellular uptake of viruses and nanoparticles at the ventral side. Nat. Commun. 11, 32 (2020).

Acknowledgments: We thank J. Son for insightful discussion. Funding: This work was supported by Samsung Research Funding and Incubation Center of Samsung Electronics (SRFC-MA1502-51). Author contributions: S.K. and J.-M.N. conceived the initial idea and study. S.K. designed the experiment under the guidance of J.-M.N. K.S., J.S., J.-E.P., and J.-M.N. discussed the results. S.K., N.K., J.-E.P., E.H.S., S.Y.C., and J.E.K. performed the experiments. S.K., N.K., E.H.S., S.Y.C., J.E.K., S.C., and H.H.P. analyzed the data under the guidance of J.-M.N. S.K. and J.-M.N. wrote the manuscript with help from all authors. Competing interests: S.K., N.K., and J.-M.N. are coinventors of a patent filed by Seoul National University on related works. All other authors declare that they have no competing interests. Data and materials availability: All data needed to evaluate the conclusions in the paper are present in the paper and/or the Supplementary Materials. Additional data related to this paper may be requested from the authors.

Submitted 17 February 2020
Accepted 14 July 2020
Published 26 August 2020
10.1126/sciadv.abb3348

Citation: S. Kim, N. Kim, J. Seo, J.-E. Park, E. H. Song, S. Y. Choi, J. E. Kim, S. Cha, H. H. Park, J.-M. Nam, Nanoparticle-based computing architecture for nanoparticle neural networks. Sci. Adv. 6, eabb3348 (2020).