

The Diffusion Network in Analog VLSI Exploiting Noise-Induced Stochastic Dynamics to Regenerate Various Continuous Paths

Yi-Da Wu and Hsin Chen, *Member, IEEE*

Abstract—The Diffusion Network (DN) is a stochastic recurrent network capable of learning to regenerate various distributions of continuous-valued, continuous-time paths. By generalizing the data variability with internal stochasticity, the DN is found useful for distinguishing time-varying, biomedical signals reliably. In addition, the DN is a generalized form of the hidden Markov model and the Kalman filter, both of which have been useful in many applications. However, the continuous-time dynamics of the DN are governed by stochastic differential equations, making the DN unfavorable for execution in a digital computer. This paper presents the translation of the DN into analog very large scale integration (VLSI). By exploiting the natural differential current-voltage relationship of capacitors, the stochastic differential equations are computed by analog VLSI simultaneously in real time. The stochasticity required by the DN in VLSI is induced by a multi-channel noise generator on-chip, such that the DN actually uses noise-induced stochastic dynamics to generate continuous paths. Moreover, log-domain representation is employed to increase the dynamic range for diffusion processes, as well as to facilitate the subthreshold operation for power reduction. As subthreshold operation is prone to more nonlinear effects, the biasing conditions and operation ranges that minimize the effects are identified. Afterwards, a DN system-on-a-chip with four stochastic units is fabricated with the 0.18 μm CMOS technology. The measurement results reveal the DN in VLSI is able to regenerate a variety of continuous paths with noise-induced stochastic dynamics in VLSI. Moreover, the practical utility of the DN system is demonstrated in the context of recognizing electrocardiograms.

Index Terms—Analog VLSI implementation, Diffusion Network, noise, stochastic dynamics.

I. INTRODUCTION

PROBABILISTIC models are able to use stochasticity to generalize the variability of data. This feature is particularly useful for robust classification of biomedical signals which are inherently noisy and drifting. Therefore, implementing probabilistic models in very large scale integration (VLSI) has been attractive to low-power signal processing in biomedical implants [1], [2]. For example, recognizing multi-channel neural activity on-line is important for implantable brain-machine interfaces to avoid transmitting all data wirelessly, and to control prosthetic devices in real time [3]–[5].

Manuscript received October 01, 2014; revised February 16, 2015; accepted March 07, 2015. Date of publication May 14, 2015; date of current version May 25, 2015. This work is supported by the Ministry of Science and Technology in Taiwan (grant no. MOST 103-2220-E-007-025, NSC 102-2220-E-007-008), and the Chip Implementation Center (Taiwan) for chip fabrication. This paper was recommended by Associate Editor T. S. Gotarredona.

The authors are with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: hchen@ee.nthu.edu.tw).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCSI.2015.2416811

Among various probabilistic models implemented in VLSI [6]–[15], the continuous restricted Boltzmann machine (CRBM) in VLSI uses noise to implement the stochastic units of the CRBM, and the CRBM in VLSI is shown capable of using noise-induced stochastic equilibria to regenerate static data distributions of various forms [15]. In addition, as noise is used to represent the variability of data in the same category, the CRBM is able to cluster or to classify data reliably [16], [17]. These intriguing findings point towards the novel concept of using noise for computation, which is especially attractive for advanced VLSI technologies where noise grows dramatically. However, the CRBM is not suitable for modelling time-varying signals because each time-varying signal needs to be represented as a static vector containing all the signal's samples across time. The number of computing units required for modelling the signal then grows in proportion to the number of samples. Although a large CRBM network in VLSI can be formed by interconnecting multiple CRBM chips in [18], modeling time-varying signals with a large CRBM network is prone to considerable power- and area-consumption.

The Diffusion Network (DN) is a stochastic recurrent neural network, whose dynamics are governed by stochastic differential equations (SDEs). According to [19], the DN is equivalent to a variant of the hidden Markov model, which possesses continuous-valued states and continuous-time dynamics. Basing on the Monte-Carlo expectation-maximization (EM) algorithm, the stochastic dynamics of the DN can be trained to regenerate various distributions of continuous-valued, continuous-time paths [19]. As the continuous-time dynamics of one computing unit in the DN is able to represent a one-dimensional path, a DN with only two computing units is sufficient for regenerating a set of one-dimensional paths. This feature makes the DN much more favorable than the CRBM for modeling time-varying signals. In addition, the DN also uses noise to represent the variability of continuous paths in the same category. The DN is shown suitable for distinguishing noisy, continuous-time, biomedical signals reliably [20]. However, the speed of simulating stochastic differential equations in a digital computer is inherently limited by the serial processing and numerical iterations of the computer. Translating the DN into analog circuits is thus of great interests because all differential equations can be simulated simultaneously in real time by exploiting the natural, differential current-voltage (I-V) relationship of capacitors [21].

The analog VLSI implementation of the DN is first proposed in [22], in which the state of each stochastic unit is represented as a voltage on a capacitor. The continuous-time, stochastic dynamics of the state are then generated by injecting noisy current

inputs to the capacitor. However, the dynamic range of the state is limited by the supply voltage. To alleviate this constraint, this paper derives the log-domain translation of the DN algorithm according to [23]. The translation allows the state of each stochastic unit to be represented as a current in VLSI, so that the DN in VLSI is able to generate stochastic dynamics power-efficiently with a wide dynamic range. In addition, the stochasticity required by each unit is induced by a noise generator. Therefore, the VLSI implementation of the DN allows us to examine the feasibility of using noise-induced stochastic dynamics for computation. While preliminary simulation results are presented in [24], this paper further investigates how nonideal characteristics of VLSI circuits could impede the DN in VLSI from regenerating continuous paths correctly. The biasing conditions and operation ranges for minimizing the nonideal effects are then identified and implemented in our final VLSI design. A DN system-on-a-chip with four stochastic units is fabricated with the 0.18 μm CMOS technology. By measuring the functionality of the DN system, the feasibility of using noise-induced stochastic dynamics to regenerate various distributions of continuous paths are verified and discussed.

Following the brief introduction to the DN in Section II, Section III derives the log-domain representation of the DN, and the corresponding VLSI implementation is described in Section IV. Afterwards, Section V investigates the nonideal effects of VLSI circuits and identifies the design guidelines for minimizing these nonideal effects. Section VI then presents and discusses the measurement results of using the DN system to regenerate different types of continuous paths. Finally, Section VII concludes the contribution and future works.

II. THE DIFFUSION NETWORK

Fig. 1 shows the architecture of the DN, comprising n continuous-time, continuous-valued stochastic units with fully connections. The strength of the connections are defined by an $n \times n$ weight matrix $\{\omega_{ij}\}$ for $i, j = 1, \dots, n$. Let $x_j(t)$ denote the state of the j^{th} unit at time t . The dynamics of $x_j(t)$ during the interval $t \in [0, T]$ are governed by the following SDE.

$$\frac{dx_j(t)}{dt} = \mu_j(t) + \sigma \cdot \frac{dB(t)}{dt}, \text{ for } t \in [0, T], j = 1, 2, \dots, n, \quad (1)$$

where $\mu_j(t)$ is a deterministic *drift* term given in (2), while $B(t)$ represents a one-dimensional *Brownian motion* with the noise amplitude determined by the positive constant σ . The Brownian motion introduces the stochasticity, enriching greatly the representational capability of the DN [19].

$$\mu_j(t) = \kappa_j \cdot \left\{ -\rho_j x_j(t) + \xi_j + \sum_{i=1}^n \omega_{ij} \cdot \varphi(x_i(t)) \right\}. \quad (2)$$

In (2), ω_{ij} defines the *connection strength* from unit i to unit j . κ_j^{-1} and ρ_j^{-1} represent the *input capacitance* and *transmembrane resistance*, respectively, of the j^{th} unit. ξ_j is the *input bias*, and φ is the *sigmoid* function given as (3), where a adapts the slope of the sigmoid function. As shown by Fig. 1, the DN contains both visible (white) and hidden (grey) stochastic units. It is notable that, if φ is replaced by a linear function and the connections from visible to all other neurons are removed, the DN with $\kappa_j = \rho_j = 0$ is equivalent to a Kalman filter [25].

$$\varphi(x_j; a) = \tanh\left(\frac{a}{2}x_j\right) = -1 + \frac{2}{1 + e^{-ax_j}}. \quad (3)$$

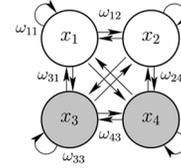


Fig. 1. The architecture of a Diffusion Network with two visible units in white and two hidden units in grey.

Basing on the Monte-Carlo EM algorithm, the DN is able to learn to regenerate at visible units a specific set of continuous paths with a given probability distribution [19]. The number of visible units, hence, equals the dimension of the training data set, while the minimum number of hidden units required for modelling the data satisfactorily is identified by experimental trials. During training, visible units are clamped to the dynamics of the training data. The corresponding dynamics of the hidden units are then Monte-Carlo sampled to estimate the optimal values for $(\omega_{ij}, \kappa_j, \rho_j, \xi_j)$, so as to maximize the expectation of the training data. After training, all units are given initial values at $t = 0$, and then the dynamics of all units during $t \in (0, T]$ are sampled. The similarity between the training data and the dynamics regenerated at visible units indicate how well the DN models the data.

To employ a trained DN as a classifier, calculating the likelihood of regenerating an unknown datum is the principle method. However, the likelihood calculation is not hardware-friendly. Instead, the deterministic dynamics of hidden units generated as the datum is clamped to visible units are shown useful for data identification [20]. This alternative is especially suitable for the DN in VLSI to recognize data in real time, and will be demonstrated in Section VI-E.

III. LOG-DOMAIN TRANSLATION INTO VLSI

A. Log-Domain Translation

To maximize the dynamic ranges for diffusion processes in VLSI, the stochastic state $x_j(t)$ is represented as a current and then logarithmically-compressed into a voltage V_{Xj} in VLSI [24]. The logarithmic compression allows $x_j(t)$ to change over several decades within a limited voltage range for V_{Xj} . On the other hand, the voltage representation V_{Xj} facilitates the exploitation of the nature, differential (I-V) relationship of a capacitor to simulate SDEs in real time and in parallel.

The logarithmic relationship between $x_j(t)$ and V_{Xj} is realized by the exponential I-V characteristics of MOS transistors in subthreshold operation. To maintain $x_j(t)$ a non-negative current in VLSI, an offset x_{off} is added to result in

$$x_j + x_{off} \equiv I_s \cdot e^{\alpha V_{Xj}}, \quad dx_j = \alpha I_s \cdot e^{\alpha V_{Xj}} \cdot dV_{Xj}, \quad (4)$$

where I_s and α are process-dependent constants. Substituting (4) into (1) then gives

$$C_{Xj} \cdot \frac{dV_{Xj}}{dt} = -\rho_j I_s + \left[\xi_j + \sum_{i=1}^n \omega_{ij} \varphi(x_i) \right] \cdot e^{-\alpha V_{Xj}} + \frac{\sigma}{\kappa_j} \frac{dB_j(t)}{dt} \cdot e^{-\alpha V_{Xj}} + \rho_j x_{off} \cdot e^{-\alpha V_{Xj}}, \quad (5)$$

where C_{Xj} equals α/κ_j . Fig. 2 illustrates the block diagram for implementing (5) in VLSI. C_{Xj} is a capacitor and V_{Xj} the voltage across the capacitor. Each term on the right hand side of (5) then corresponds to a current charging or discharging C_{Xj} .

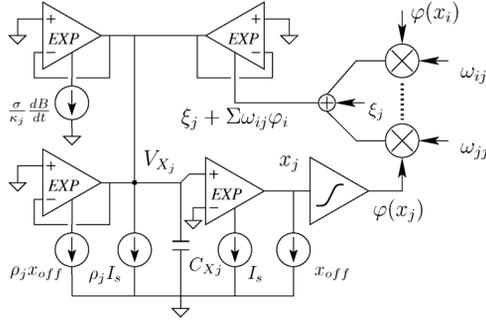


Fig. 2. The block diagram of a stochastic unit of the DN in VLSI.

TABLE I
THE PARAMETER MAPPINGS BETWEEN NUMERICAL SIMULATION AND VLSI IMPLEMENTATION

parameter	numeric	circuit	unit value
x_j	-3~5	-30~50 nA	$I_{unit} = 10$ nA
x_{off}	5	50 nA	offset term in (4)
V_{X_j}	0.773~0.827	773~827 mV	$V_{X_j,unit} = 1$ V
ω, ξ	-30~30	-300~300 nA	$I_{unit} = 10$ nA
$\varphi(x_j)$	-1~1	-400~400 nA	activation function
C_{X_j}	$\alpha/\kappa_j = 30$	30 pF	$C_{unit} = 1$ pF
Δt	0.05	5 μ s	$t_{unit} = 0.1$ ms
ρ	0.5~2	0.5~2	$\rho_{unit} = 1$

The EXP-element in Fig. 2 produces an output current of $I_{out} = I_{VAR} \cdot \exp\{\alpha(V_P - V_N)\}$, where I_{VAR} and $(V_P - V_N)$ stand for the input current and the differential input voltage, respectively. Therefore, the first term in (5), $-\rho_j I_s$, is implemented by a constant current source, while the other three terms by the EXP-elements. Finally, the sigmoid circuit converts x_j into $\varphi(x_j)$, and the multipliers output a total current proportional to $\sum_i \omega_{ij} \cdot \varphi(x_i)$.

B. Parameter Mappings

By extensive simulation, the dynamic ranges of parameters and their mappings from numerical simulation into VLSI implementation are identified and summarized in Table I [24]. The unit values give the current or voltage level corresponding to 1 in numerical simulation. The unit currents of x_j , ω_{ij} , and ξ_j are defined as 10 nA to match the current scale of subthreshold operation, as well as to reduce the power consumption. While the diffusion process in (1) is iterated with $\Delta t = 0.05$ in numerical simulation, Δt is set to be 5 μ s in VLSI, corresponding to a reasonable sampling rate of 200 kHz at which most instruments can sample multiple channels (units) simultaneously. Finally, the unit capacitance for $1/\kappa_j$ is calculated as $C_{unit} = I_{unit} \cdot \Delta t_{unit} / V_{X_j,unit}$, equaling 1 pF and resulting in $C_{X_j} = \alpha C_{unit} = 30$ pF.

IV. CIRCUIT IMPLEMENTATION

A. The EXP-Element

Fig. 3(a) shows the schematics of the EXP-element. With M1 and M2 operated in the subthreshold region, the output current is given as

$$I_{out} = I_B \cdot \exp\left(\frac{\kappa}{U_T}(V_P - V_N)\right), \quad (6)$$

where U_T denotes the thermal voltage and κ the *gate-coupling coefficient*. Transistors M3–M5 form an active biasing circuit, allowing I_{out} to change over several decades. Comparing (6)

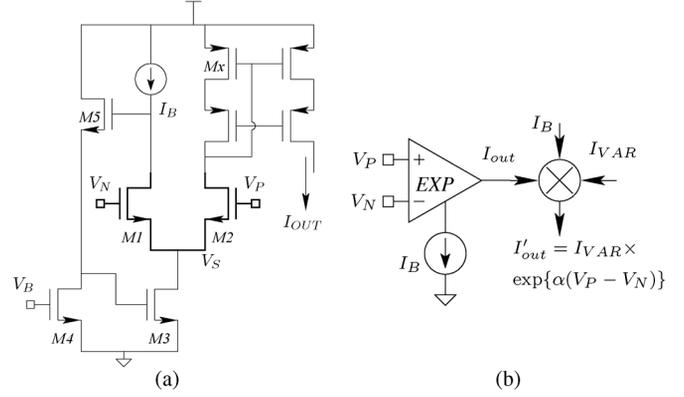


Fig. 3. (a) The schematics and (b) the block diagram of the EXP-element.

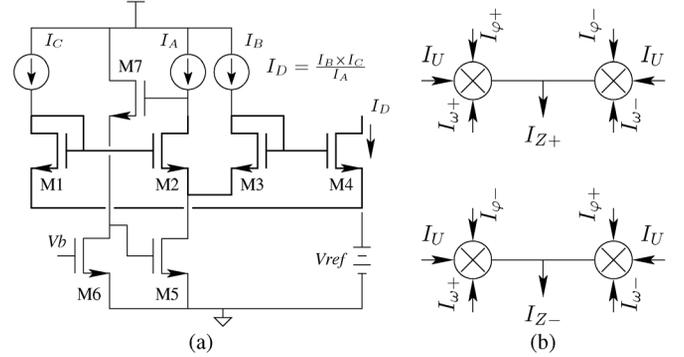


Fig. 4. (a) The schematics of the one-quadrant multiplier. (b) The block diagram of the four-quadrant multiplier consisting of the circuit in (a).

with (4) reveals that $\alpha = \kappa/U_T$. However, κ actually varies with the current level of M2. To prevent the variable α from introducing simulation errors, all EXP-elements of the DN unit are biased with a constant $I_B = 100$ nA. I_{out} of each element is then re-scaled by the one-quadrant current multiplier in Fig. 4(a) to produce $I'_{out} = I_{out} \times I_{VAR}/I_B$ (Fig. 3(b)). I_{VAR} represents the current input to each element in Fig. 2 (e.g., $\sum \omega \varphi$ or ρx_{off}) [26].

B. Current Multipliers

Four-quadrant multipliers basing on translinear loops are employed to calculate $\sum \omega_{ij} \varphi(x_i)$ in (5). Both ω_{ij} and $\varphi(x_i)$ are represented by differential currents as

$$\omega_{ij} = I_{\omega+} - I_{\omega-}, \quad \varphi(x_i) = I_{\varphi+} - I_{\varphi-}. \quad (7)$$

Let the differential current $(I_{Z+} - I_{Z-})$ represent the multiplier's output and I_U a unit current. Equation (8) indicates that the four-quadrant multiplication can be composed of four one-quadrant multipliers in Fig. 4(a), as illustrated by Fig. 4(b).

$$\begin{aligned} I_{Z+} \cdot I_U - I_{Z-} \cdot I_U &= \omega_{ij} \cdot \varphi(x_i) \\ &= (I_{\omega+} \cdot I_{\varphi+} + I_{\omega-} \cdot I_{\varphi-}) - (I_{\omega+} \cdot I_{\varphi-} + I_{\omega-} \cdot I_{\varphi+}). \quad (8) \end{aligned}$$

C. Sigmoid Function $\varphi(\cdot)$

Fig. 5 shows the block diagram for implementing the sigmoid function φ in (3). The current I_{X_i} representing x_i is firstly converted into a voltage V_i by the operational amplifier (OPA) with the voltage-controlled active resistor (VCR) proposed in

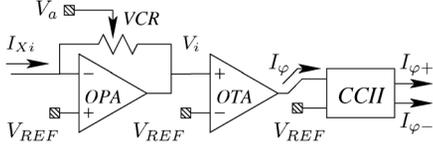


Fig. 5. The block diagram of the sigmoid circuit.

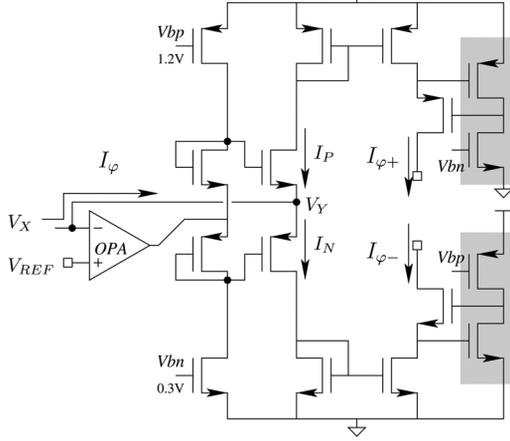


Fig. 6. The circuit diagram of the single-to-differential current conveyor.

[27]. V_i is then sent to an operational transconductance amplifier (OTA) in subthreshold operation, producing an output current of

$$I_\varphi = I_B \tanh \left(\frac{\kappa}{2U_T} (V_i - V_{ref}) \right). \quad (9)$$

Since $V_i - V_{ref} = R_i I_{X_i}$ and R_i is the resistance of the VCR, the voltage V_a controls R_i and thus the slope of the sigmoid function. Finally, the 2nd generation current conveyor (CCII) in Fig. 6 converts the current I_φ into a pair of differential currents ($I_{\varphi+}$, $I_{\varphi-}$) ranging between -400 nA and $+400$ nA [28]. The differential currents are then duplicated for the inputs of four-quadrant multipliers of all DN units.

D. The Noise Generator

The noise generator proposed in [29] is employed to produce multi-channel, uncorrelated noise for DN units. Fig. 7 shows the switch-capacitor circuit and its control signals that performs the following MASH quantization

$$V_{i+1} = f_{\Delta\Sigma}(V_i + V_{i-1}) = \begin{cases} (V_i + V_{i-1}) - vdd & \text{if } V_i + V_{i-1} \geq vdd, \\ (V_i + V_{i-1}) + vdd & \text{if } V_i + V_{i-1} < vdd. \end{cases} \quad (10)$$

By cascading four identical circuits in a ring topology, four channels of nearly-uncorrelated noise with uniform distribution over $[0, V_{dd}]$ are generated. The amplitude of V_i is then reduced to around 100 mV by C3–C4 for the differential pair, which steers 70 nA of noise current into the EXP-element in accordance with Fig. 2. It is notable that, although the original DN formula in (1) incorporates Gaussian noise, the main utility of noise is representing data variability. Therefore, different noise distributions could only cause the DN to regenerate slightly different data distributions. To keep the circuit simple, the uniformly-distributed noise is used directly in VLSI.

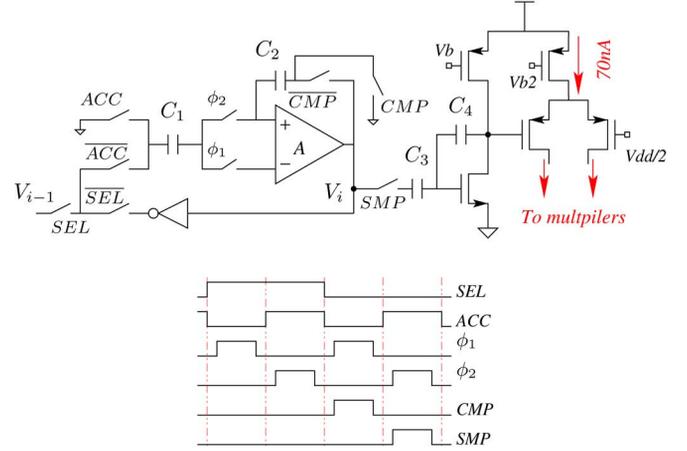


Fig. 7. The schematics of the noise unit and its corresponding timing.

V. CIRCUIT NONIDEALITIES

Although the circuits described in Section IV realizes the log-domain translation of the DN, the SPICE simulation in [24] reveals that the dynamics generated by the circuits could deviate from those in ideal numerical simulation. The main reason is because the subthreshold parameter κ in (6) is dependent on the biasing condition of a transistor. As κ corresponds directly to α in the exponential terms in (5), the errors induced by κ could easily drive the circuit dynamics to deviate from ideal ones. The deviation could even grow dramatically if the errors are accumulated during the recurrent regeneration of dynamics.

To minimize the nonideal effect in real circuits, this section first investigates the nonlinear dependence of κ on the biasing conditions of a transistor. How the nonlinearity affects the precision of the circuits described in Section IV is then evaluated quantitatively. Subsequently, a macromodel of the log-domain DN in VLSI is built for behavioral simulation in SPICE. By replacing individual circuit blocks one at a time with its transistor-level circuit, the circuit affecting dynamics most is identified. Finally, the biasing conditions of transistors and the dynamic ranges of component circuits are selected to minimize these nonideal effects.

A. The Nonlinear Variability of κ

The drain current of a subthreshold MOS in saturation is given as

$$I_D = \lambda I_S \exp \left(\frac{\kappa V_G - V_S}{U_T} \right), \quad (11)$$

where I_S is a technology-dependent constant and λ denotes the W/L ratio of the transistor. The drain current of M2 in the EXP-element (Fig. 3(a)) is simulated in SPICE by sweeping the gate-to-source voltage (V_{GS}) under different drain-to-source voltages (V_{DS}). The effective κ is then calculated according to (12) and illustrated in Fig. 8.

$$\kappa(V_G) = \frac{U_T}{\Delta V} \cdot \ln \left(\frac{I_D(V_G + \Delta V)}{I_D(V_G)} \right), \quad \Delta V \rightarrow 0 \quad (12)$$

Fig. 8 reveals that κ is nearly independent of V_{DS} when V_{GS} is greater than 0.3 V. To reduce the variability of κ , all transistors

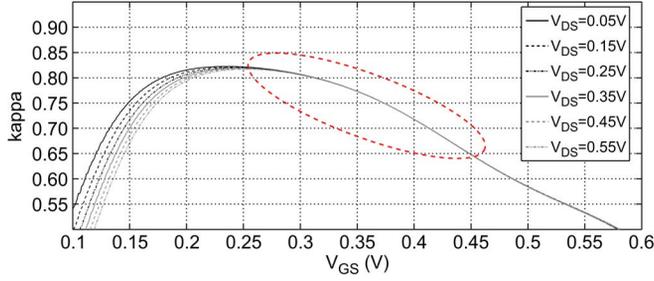


Fig. 8. The kappa simulation of a subthreshold MOS versus V_{GS} with respect to different V_{DS} .

in subthreshold operation are biased in the region circled by the red dashed line in Fig. 8.

B. The EXP Element

As κ still depends on V_{GS} , the effective α implemented by an EXP-element circuit actually varies with its input voltage. According to (6), the effective α is expressed as

$$\alpha_{eff} = \frac{1}{V_P - V_N} \ln \left(\frac{I_{OUT}}{I_B} \right) \quad (13)$$

With V_N fixed to 0.75 V, the output current of the EXP-element circuit is simulated for different V_P . Fig. 9(a) compares the simulated current with the ideal current derived from (4) with $\alpha = 30$. The relative error and α_{eff} are shown in Fig. 9(b). The output current is greater than the ideal value when $V_P < 0.82$ V. This is because α_{eff} becomes greater than 30 in this operation region. The increase of α_{eff} can be reasoned from the fact that M1 and M2 with different V_{GS} have different κ . Let the drain currents of M1 and M2 be written as

$$I_{D1} = \lambda I_S \exp \left(\frac{\kappa_1 V_N - V_S}{U_T} \right), \quad (14)$$

$$I_{OUT} = \lambda I_S \exp \left(\frac{\kappa_2 V_P - V_S}{U_T} \right) \quad (15)$$

Dividing (15) by (14) gives

$$I_{OUT} = I_B \cdot \exp \left\{ \frac{\kappa_1}{U_T} \cdot \left[\frac{\kappa_2}{\kappa_1} \cdot V_P - V_N \right] \right\} \quad (16)$$

κ_1 is constant because V_N is constant in this simulation. So α_{eff} merely depends on κ_2 . According to Fig. 8, decreasing V_P causes κ_2 and α_{eff} to increase. Moreover, since the drain of M2 is connected to a folded-cascode current mirror, V_{DS} of M2 increases significantly when the current of M2 reduces with V_P . The channel-length modulation effect of M2 further augments I_{OUT} and α_{eff} , as indicated by (13).

C. Current Multiplier

The simulation of the one-quadrant multiplier in Fig. 4(a) indicates that the channel-length-modulation effect encountered by M4 and the variability of κ could cause output current to increase. To minimize the channel-length modulation effect and the κ variability, transistors M1–M4 are designed to have long channel lengths and operated in the region circled by the red curves in Fig. 8. Fig. 10(a) compares the simulated and ideal currents of a four-quadrant multiplier, and the corresponding errors are displayed by Fig. 10(b). Although the simulated currents are

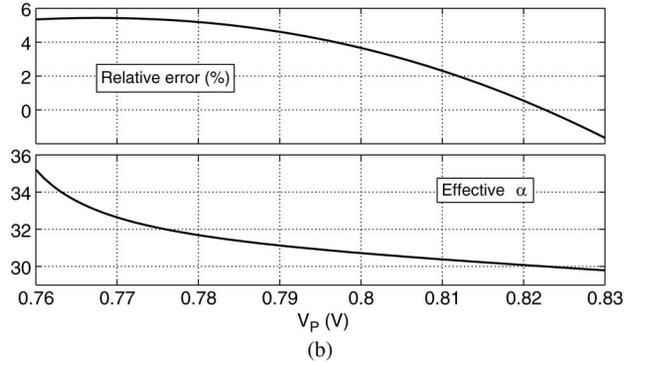
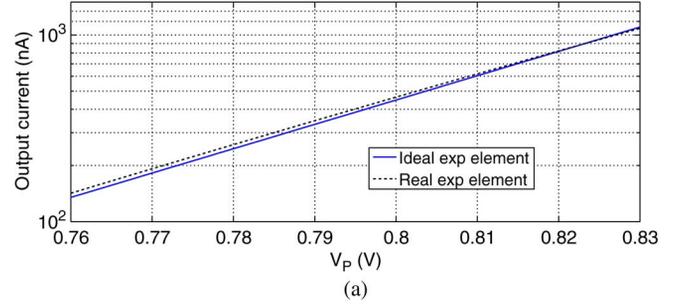


Fig. 9. The nonidealities simulation of the EXP element. (a) The output current deviation. (b) The relative error and effective α .

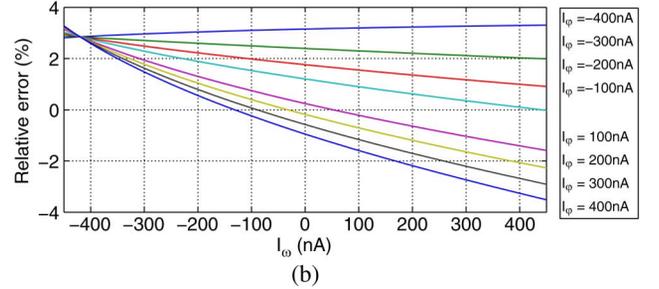
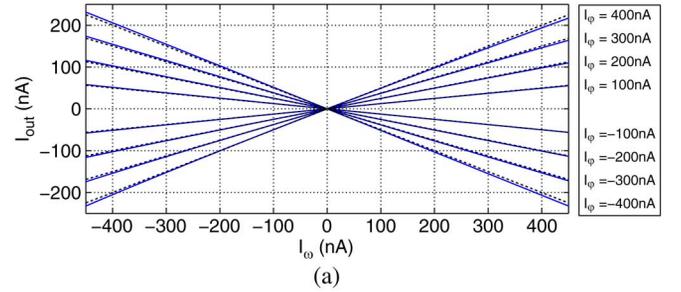


Fig. 10. The nonidealities simulation of the four-quadrant multiplier. (a) The output current deviation. (b) The relative error.

greater than the ideal values for most dynamic ranges, the maximum error is below 4%.

D. Sigmoid Circuit

The transfer curves of the sigmoid circuit in Fig. 5 are simulated and compared with those of the ideal sigmoid function, as illustrated by Fig. 11(a). The corresponding errors are shown in Fig. 11(b). The error mainly results from the fact that the differential-input transistors of the OTA in Fig. 5 have different κ whenever $V_i \neq V_{ref}$. The more V_i differs from V_{ref} , the larger difference in κ is induced. This explains why the largest error occurs around the bending corners of the sigmoid curves, where

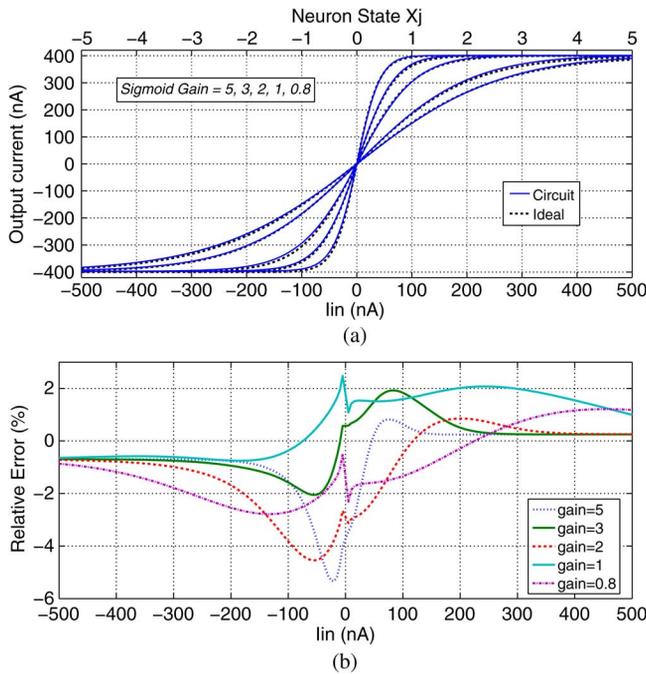


Fig. 11. The nonidealities simulation of the sigmoid function φ . (a) The output current deviation. (b) The relative error.

V_i differs from V_{ref} significantly but has not driven the sigmoid output to its maximum or minimum. Beyond the bending corners, the output current of the OTA reaches its maximum or minimum. As long as the maximum or minimum current complies with mapping defined in Table I, the errors are small, as revealed by Fig. 11(b).

E. Ideal Macromodels of Component Circuits

To investigate how the nonidealities of individual circuits affect the dynamics of a DN in VLSI, the ideal, behavioral macromodel of all component circuits are built for HSPICE simulation. The macromodels allow us to simulate the dynamics of an ideal DN system in VLSI without circuit nonidealities. By replacing one component at a time with its transistor-level circuit in HSPICE, how the nonidealities of this component circuit affect the system dynamics is revealed. As all the building blocks in Fig. 2 generate current outputs, their macromodels are described as *Behavioral Current Sources (BCS)* as follows

1) The EXP element:

Each EXP element sinks a current exponentially proportional to its differential input voltage. With $\alpha = 30$ and $I_B = 100$ nA, the macromodel of the EXP element is described as

$$\begin{aligned} \text{Gexp out gnd} \\ + \text{CUR} = '100e - 9 * \text{EXP}(30 * (V(Vp) - V(Vn)))' \end{aligned}$$

2) The current multiplier:

As each one-quadrant multiplier generates an output current of $(I_B \times I_C / I_A)$, the corresponding model is described as

$$\text{Gout id gnd CUR} = 'i(Vb) * i(Vc) / i(Va)',$$

where V_a , V_b , and V_c are the ideal voltage sources that conduct the input currents I_A , I_B , and I_C , respectively. The four-quadrant multipliers are further built by assembling four one-quadrant multipliers.

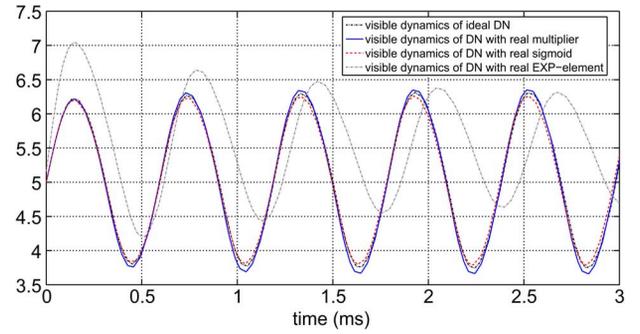


Fig. 12. The impact on the regenerated sinusoidal dynamics by the nonidealities of different component circuits.

3) The sigmoid function φ :

The sigmoid function produces a hyperbolic tangent output according to (3). Therefore, the behavioral model of the sigmoid circuit in Fig. 5 is described as

$$\begin{aligned} \text{Gsig iout gnd CUR} = ' -400n + \\ + 800n / (1 + \text{EXP}(-1 * a * i(Vin) / 1e - 7))' \end{aligned}$$

where V_{in} represents the ideal voltage source that conducts the input current of the sigmoid circuit.

F. Nonideal Effects on System Dynamics

To obviate how circuit nonidealities affect the dynamics generated by a DN in VLSI, the task of regenerating a sinusoidal signal with a DN containing one visible and one hidden neurons is simulated. As the current multipliers are replaced by real circuits, the sinusoidal signal regenerated by the DN is shown by the blue curve in Fig. 12. It agrees with the black dashed curve generated with ideal macromodels. Therefore, the nonidealities of the current multipliers do not disturb the dynamics seriously. The main reason is that the inputs of all multipliers, $\varphi(x_j)$ and ω_{ij} , are confined within $[-200$ nA 200 nA]. According to Fig. 10(b), the relative errors are kept below 2%, such that the dynamics are hardly affected. As the sigmoid circuits are replaced by real circuits, the sinusoidal signal regenerated by the DN is shown by the red curve in Fig. 12. Real sigmoid circuits also have negligible effect on the dynamics because the errors of real circuits are below 2% for most of the dynamic range (Fig. 11(b)). Finally, as real EXP-element circuits are substituted for their macromodels, the sinusoidal signal regenerated by the DN is shown by the grey dashed curve in Fig. 12. The regenerated signal has not only an attenuating amplitude but also an increasing phase delay with respect to the ideal signal. This is mainly attributed to the fact that α_{eff} of the EXP-element circuit becomes greater than 30, the ideal value, for more than half of the dynamic range. As indicated by (5), the dynamics of V_{X_j} are driven by current components proportional to $e^{-\alpha V_{X_j}}$. Large α causes the real currents to be smaller than their ideal values, such that the amplitude and the phase of the sinusoidal signal generated on V_{X_j} is attenuated and delayed.

The simulation results above reveal that the nonidealities of the EXP-element circuit affect the dynamics of a DN system more significantly. To minimize the nonideal effect, the nominal biasing for V_P is set to be 0.82 V and transistors M1 and M2 in the EXP-element circuits are operated within the region indicated by the red dashed circle in Fig. 8. This choice of biasing and operating conditions allow α_{eff} to stay as close to 30 as possible.

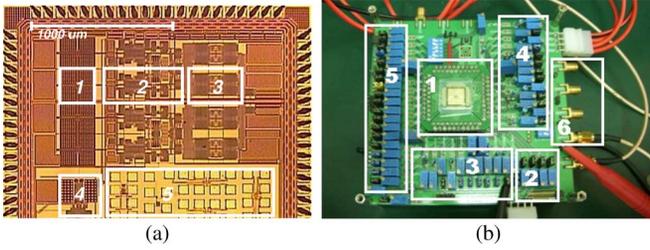


Fig. 13. (a) The chip micro-photograph. 1. Capacitor C_x . 2. Neuron 3. Synapse 4. Noise generator 5. Testkeys. (b) The measurement setup. 1. The Diffusion Network chip 2. Potentiastats for setting initial conditions 3–4. Potentiastats for setting parameters ρ_j , ξ_j , and a_j 5. Potentiastats for setting weights ω_{ij} and reference voltages/currents 6. Connections to the oscilloscope for measuring neurons' dynamics.

TABLE II
THE SPECIFICATION OF THE DN SYSTEM IN VLSI

Specification	Value
Supply voltage	1.5V
Total power consumption	948 μ W
Total core area	1.2 mm^2
Area of component circuits	neuron: 0.1 mm^2 , synapse: 0.05 mm^2 C_{X_j} : 0.06 mm^2
Power consumption of component circuits	neuron: 155 μ W, synapse: 11.2 μ W noise generator: 150 μ W
Capability	Modelling 1D/2D continuous paths

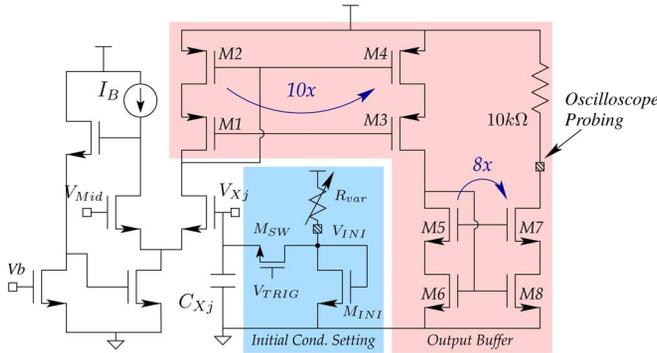


Fig. 14. The circuits for clamping the initial condition of DN units and the current buffer for measuring dynamics.

VI. EXPERIMENTAL RESULTS

A. The Full System

The DN system consisting of four stochastic units is fabricated with the CMOS 0.18 μ m technology provided by the Taiwan Semiconductor Manufacturing Company (TSMC). Fig. 13 shows the chip layout and the measurement setup. The system specification is summarized in Table II. The total power dissipation is 948 μ W, and the total area of the core circuit is 1.2 mm^2 . Testkeys of the component circuits (e.g., the EXP-element) are also included to characterize their dynamic range and linearity. As shown by Fig. 14, the initial condition on each C_{X_j} is determined by an external resistor R_{var} and set through an NMOS switch. The dynamics of each x_j (i.e., the current of M2) are amplified by 800 times and converted into a voltage by an external 10 k Ω resistor to ease measurement with an oscilloscope.

B. Measurement of Component Circuits

Fig. 15 shows the measured characteristics of the EXP-element circuit. As V_X varies from 750 mV to 840 mV, the output

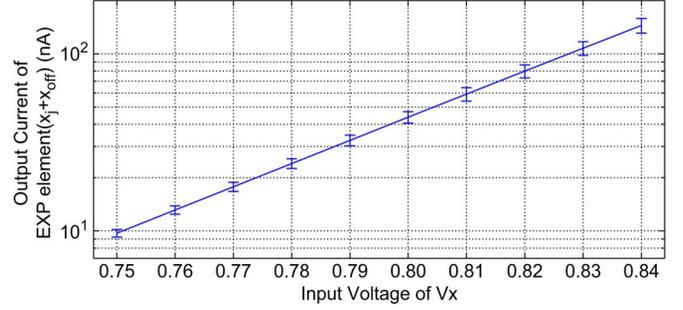


Fig. 15. The DC measurement of an EXP-element. The error-bars indicate the standard deviations measured across four samples.

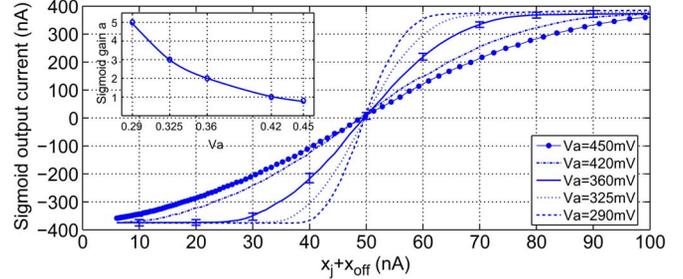


Fig. 16. The measured characteristics of the sigmoid circuit. The error-bars indicate the standard deviations (STD) measured across four samples.

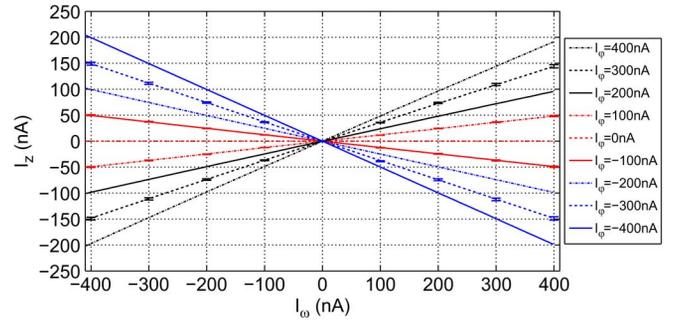


Fig. 17. The measured characteristics of the four quadrant multiplier with $I_U = 800$ nA. The error-bars indicate the STDs across four samples.

current increases from 9.7 nA to 140 nA. The log-scale plot reveals clearly that the EXP-element possesses satisfactory exponential relationship over the desired dynamic range of V_X . Fig. 16 shows the measured characteristics of the sigmoid circuit with various V_a . V_a adapts the sigmoid slope effectively, and the mapping between V_a and the parameter a in (3) is further derived and shown by the inset. Moreover, Fig. 17 shows the measured characteristics of the four-quadrant multiplier. Obviously, the multiplier has satisfactory linearity and precision over the dynamic ranges of I_ω and I_ϕ .

The error bars in Fig. 15–Fig. 17 further reveal the process variations measured across four chips (samples) for each circuit. The mean standard deviations normalized with respect to output currents are 7.2% for the EXP-element circuit, 4.8% for the sigmoid circuit, and 1.86% for the multiplier. The EXP-element circuit exhibits largest variation because $\alpha = \kappa/U_T$ is process-dependent and its variation is amplified by the exponential relationship in (6). Similarly, the output current of the sigmoid circuit also depends on κ exponentially in (9), so that the sigmoid circuit exhibits larger variation than the multiplier. Although the process variations are non-negligible, the error values are comparable to those discussed in Section V. Therefore, it is expected that only the process variation of the EXP-element circuit would

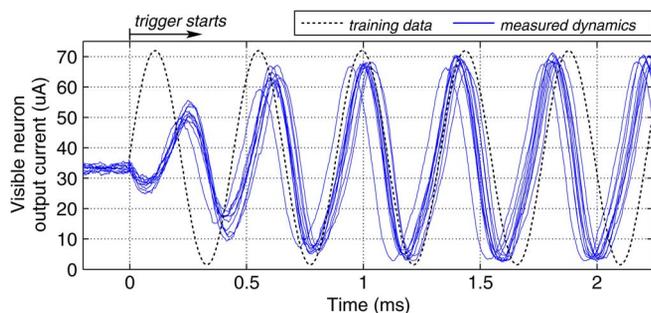


Fig. 18. The desired sinusoidal path (dotted curve) and 10 dynamics regenerated by the DN system (blue curves).

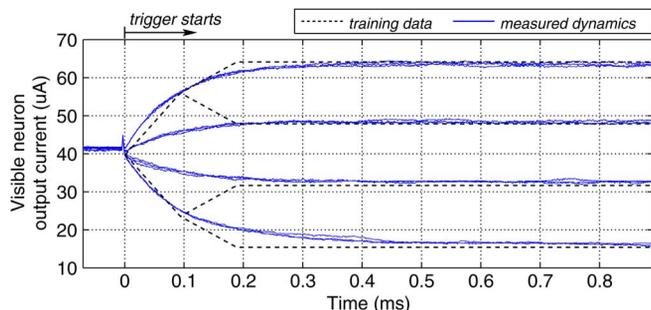


Fig. 19. The desired bifurcating paths (dotted curves) and the dynamics regenerated by the DN system in 12 experimental trials (blue curves).

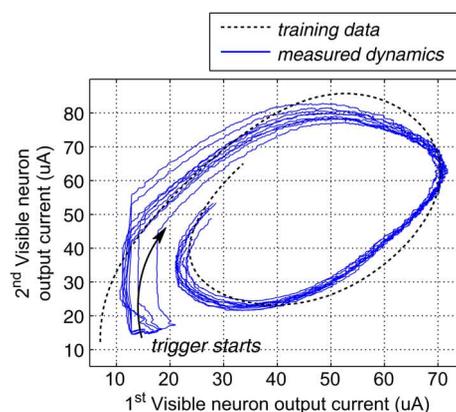
affect the dynamics of the DN system significantly. On the other hand, as the DN system computes with and transmit signals as currents, careful layout design has been adopted to achieve a mismatch error smaller than 1.5% for current mirrors (data not shown). Assume the variation across neighboring circuits on the same chip is much smaller than that across different chips. The effect of process variations can be reduced significantly.

C. Regenerating 1D Stochastic Dynamics

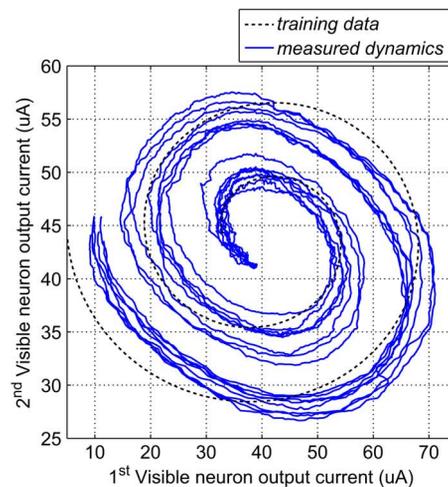
A DN with one visible and one hidden units was first trained to model the sinusoidal wave (the dotted curve in Fig. 18) in MATLAB. The parameter values obtained from numerical simulation were then programmed into the DN system according to Table I. (The same procedure was applied to program the DN system to model different data in the following experiments.) The stochastic dynamics regenerated by the visible unit of the DN system were shown by the blue curves in Fig. 18. Each curve corresponded to the measured dynamics in one experimental trial, and the noise made the dynamics different from one to another. Nevertheless, the amplitude and the frequency agreed with the training data after the first two sinusoid cycles. This indicated that the circuit realized the mapping in Table I faithfully.

To demonstrate the noise-enhanced representational capability, the DN system was further programmed to regenerate the bifurcating dynamics shown by the dotted curves in Fig. 19. The blue curves in Fig. 19 showed the measured dynamics of the visible unit. Even with an identical initial condition, the dynamics were disturbed by noise and converged into one of the four equilibriums after 0.8 ms. The agreement between the training data and the regenerated dynamics demonstrated that the noise-induced stochastic dynamics were well-controlled to regenerate the four bifurcating curves simultaneously.

Both Fig. 18 and Fig. 19 reveal that the dynamics regenerated by the DN system do not follow the training data well



(a)



(b)

Fig. 20. The dynamics regenerated by the DN system in 10 experimental trials (blue curves), as it was programmed to model: (a) the handwritten ρ (the dotted curve); (b) the spiral curve (the dotted curve).

in the beginning. This phenomenon is attributed to the clock-feedthrough errors introduced to V_{X_j} by the initialization circuit. An error of a few millivolts can cause the initial dynamics to be very different because X_j grows exponentially with V_{X_j} . Nevertheless, the dynamics are brought back to the modelled paths within 0.5 ms.

D. Regenerating 2D Stochastic Dynamics

The DN system with two visible units were programmed to regenerate the handwritten ρ (the dotted curve) in Fig. 20(a). With V_{X_j} of both visible units initialized to 760 mV, the regenerated dynamics in 10 experimental trials were shown by the blue curves. The variation among the regenerated dynamics was attributed to the noise of the DN system. The DN system was also able to regenerate spiral curves, as shown by Fig. 20(b). The similarity between the regenerated data and the training data further demonstrated the DN's ability to model two-dimensional paths satisfactorily.

E. Signal Identification

In addition to regenerating different types of signals, the utility of the DN system in identifying biomedical signals was examined. The DN with one visible and one hidden units was first trained in MATLAB to model the QRS segments of 10 normal electrocardiograms (ECGs). The training data were

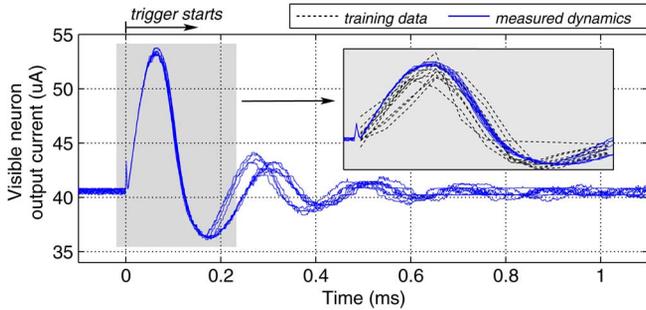


Fig. 21. The dynamics regenerated by the DN system in 10 experimental trials (blue curves). The inset shows the magnification of the grey window as well as the training data (the dotted curves).

extracted from a 30-minute long recording in the MIT-BIH database (dotted curves in Fig. 21). The trained parameters were obtained as

$$\omega = \begin{bmatrix} \omega_{vv} & \omega_{vh} \\ \omega_{hv} & \omega_{hh} \end{bmatrix} = \begin{bmatrix} -9.1 & -22.9 \\ 4.7 & 10.5 \end{bmatrix}, \quad (17)$$

$$\xi = \begin{bmatrix} \xi_v \\ \xi_h \end{bmatrix} = \begin{bmatrix} 3.0 \\ 5.8 \end{bmatrix}, \quad \rho = \begin{bmatrix} \rho_v \\ \rho_h \end{bmatrix} = \begin{bmatrix} 0.8 \\ 0.95 \end{bmatrix}. \quad (18)$$

With the parameter values programmed onto the DN system, the dynamics regenerated by the DN in 10 trials were shown by the blue curves in Fig. 21. The inset revealed that the regenerated dynamics agree with the training data. The prolonged dynamics had their amplitudes attenuated towards zero within 0.8 ms. This phenomenon indicated that, although the QRS segments looked like sinusoidal signals, the modelled dynamics differ significantly from those in Fig. 18.

To test the DN's ability to identify signals, a testing dataset containing the QRS segments of 10 normal and 10 abnormal ECGs was used (the insets of Fig. 22). With the noise of the DN system turned off, the dynamics of visible unit were clamped to each testing datum, and the corresponding dynamics of the hidden neuron were measured and plotted in Fig. 22. For normal ECGs, the dynamics (blue curves) looked like sinusoidal waves with exponential decay towards the minimum output, while for abnormal ECGs, the dynamics (red curves) saturated to the maximum output within 0.1 ms. This significant difference allowed abnormal ECGs to be identified reliably with a simple threshold detector, and the detection can be done at the early onset of ECGs, i.e., before 0.1 ms. The reliability came from the fact that the DN used noise to generalize data variability, such that the connection weight could represent the key features of normal ECGs effectively. Even though the normal and abnormal ECGs had similar waveforms, the recurrent architecture accumulated the difference and resulted in dramatically different dynamics for the hidden neuron. Turning off the noise further allows the DN system to generate nearly the same dynamics “deterministically” for the same type of ECGs. Nevertheless, it is notable that real ECGs exhibit a much greater variety of waveforms than those in Fig. 22. This experiment only demonstrates the utility of the DN system, while a larger DN network will be essential for modelling more complex waveforms.

F. Discussion

The DN system is shown capable of exploiting noise-induced stochastic dynamics to regenerate a variety of continuous paths. Table III summarizes the mean correlation coefficients (CCs)

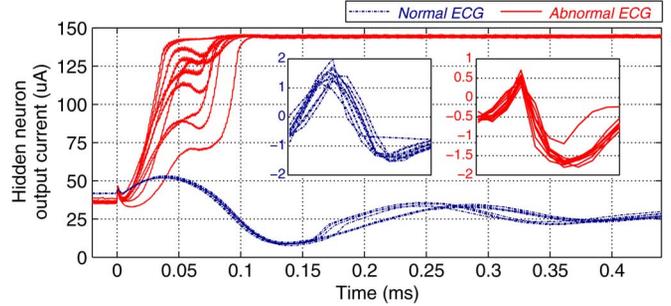


Fig. 22. The dynamics regenerated at the hidden unit as the visible unit was clamped to the testing data shown in the insets.

TABLE III
THE CORRELATION COEF. BETWEEN DESIRED AND REGENERATED PATHS

sinusoidal curve	bifurcation	ρ curve	spiral curve
0.57 (w/i 1 st cyc.)	0.98 (1 st , 4 th curves)	0.78	0.97
0.72 (w/o 1 st cyc.)	0.09 (2 nd , 3 rd curves)		

between desired and regenerated paths. For the sinusoidal signal, the CC is increased to 0.72 if the first sinusoidal cycle is not taken into account. This is because, given zero initial values, it takes time for the regenerated dynamics to catch up with the desired sinusoidal curve (Fig. 18). The same reason explains why the regenerated paths in Fig. 19 do not reproduce the second break points at $t = 0.1$ ms. Instead of regenerating the triangular ramps of the 2nd and 3rd bifurcating curves, the DN simply regenerates paths that have the same equilibrium values as the two bifurcating curves. Therefore, the mean CC for the 2nd and 3rd curves is increased to 0.89 if the first 0.1 ms is not taken into account. For the 2D dynamics, the spiral curve is regenerated better than the hand-written ρ because the former increases gradually from small initial values. The comparison implies that increasing the number of hidden units helps to improve the representational ability of the DN system. For example, a DN with five hidden neurons is able to model the difference between hand-written β and ρ [20]. Certainly, careful design will be crucial to avoid process variations (observed in Section VI-B) from distorting the dynamics regenerated by a large DN network.

VII. CONCLUSION

The Diffusion Network is translated into log domain and implemented as a stochastic system in analog VLSI. The stochasticity comes from the injection of noise into the VLSI system. The log-domain translation further allows the diffusion processes to be simulated power-efficiently with sufficient dynamic ranges. More importantly, the measurement results demonstrates for the first time that noise-induced stochastic dynamics is useful for regenerating a variety of continuous paths. This not only relies on a good definition of the parameter mapping in Table I, but also requires careful consideration of circuit nonidealities in Section V. Based on the DN algorithm, the intriguing utility of noise-induced stochastic dynamics is illustrated as generalizing the variability of ECGs, so that the DN system is able to distinguish ECG signals reliably. These promising results shed the light on the potential of using noise for computation in advanced VLSI technologies. In addition, the DN system exemplifies the advantage of analog VLSI for solving stochastic differential equations in parallel and in real time. Many stochastic dynamic systems (e.g., the brain) could

be simulated more efficiently with dedicated analog VLSI in a similar way.

REFERENCES

- [1] G. Iddan, G. Meron, A. Glukhovskiy, and P. Swain, "Wireless capsule endoscopy," *Nature*, vol. 405, no. 6785, p. 417, Jul. 2000.
- [2] T. W. Berger, M. Baudry, J.-S. L. Roberta Diaz Brinton, V. Z. Marmarelis, A. Y. Park, B. J. Sheu, and A. R. Tanguay, JR., "Brainimplantable biomimetic electronics as the next era in neural prosthetics," *Proc. IEEE*, vol. 89, no. 7, pp. 993–1012, Jul. 2001.
- [3] M. A. Lebedev and M. A. L. Nicolelis, "Brain-machine interfaces: Past, present and future," *Trends Neurosci.*, vol. 29, no. 9, pp. 536–546, 2006.
- [4] V. Gilja, C. A. Chestek, I. Diester, J. M. Henderson, K. Deisseroth, and K. V. Shenoy, "Challenges and opportunities for next-generation intracortically based neural prostheses," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 7, pp. 1891–1899, 2011.
- [5] V. Gilja, P. Nuyujukian, C. A. Chestek, J. P. Cunningham, B. M. Yu, J. M. Fan, M. M. Churchland, M. T. Kaufman, J. C. Kao, S. I. Ryu, and K. V. Shenoy, "A high-performance neural prosthesis enabled by control algorithm design," *Nature Neurosci.*, vol. 15, no. 12, pp. 1752–1757, 2012.
- [6] J. Alspector, R. B. Allen, A. Jayakumar, T. Zeppenfeld, and R. Meir, "Relaxation networks for large supervised learning problems," in *Advances in Neural Information Processing Systems*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA, USA: MIT Press, 1991, vol. 3, pp. 1015–1021.
- [7] J. Alspector, A. Jayakumar, and S. Luna, "Experimental evaluation of learning in a neural microsystem," in *Advances in Neural Information Processing Systems*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA, USA: MIT Press, 1992, vol. 4, pp. 871–872.
- [8] A. Jayakumar and J. Alspector, "A cascaded neural network chip set with on-chip learning using noise and gain annealing," in *Proc. IEEE Custom Integr. Circuits Conf.*, 1992, pp. 19.5.1–19.5.4.
- [9] D. F. Specht, "Probabilistic neural networks," *J. Neural Netw.*, vol. 3, no. 1, pp. 109–118, 1990.
- [10] I. C. Jou, R. Y. Liu, and C. Y. Wu, "Cmos implementation of neural networks for speech recognition," in *Proc. IEEE Asia-Pacific Conf. Circuits Syst. (APCCAS'94)*, pp. 513–518.
- [11] N. Aibe, M. Yasunaga, I. Yoshihara, and J. Kim, "A probabilistic neural network hardware system using a learning-parameter parallel architecture," in *Proc. IEEE 2002 Int. Joint Conf. Neural Netw. (IJCNN)*, pp. 2270–2275.
- [12] J. D. Hsu, S. Bridges, M. Figueroa, and C. Diorio, "Adaptive quantization and density estimation in silicon," in *Advances in Neural Information Processing Systems*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA, USA: MIT Press, 2003, vol. 15, pp. 1083–1090.
- [13] T. G. Clarkson, C. K. Ng, and Y. Guan, "The pram: An adaptive VLSI chip," *IEEE Trans. Neural Netw.*, vol. 4, no. 3, pp. 408–412, 1993.
- [14] T. G. Clarkson, Y. Guan, J. G. Taylor, and D. Gorse, "Generalization in probabilistic ram nets," *IEEE Trans. Neural Netw.*, vol. 4, no. 2, pp. 360–363, 1993.
- [15] H. Chen, P. Fleury, and A. Murray, "Continuous-valued probabilistic behavior in a VLSI generative model," *IEEE Trans. Neural Netw.*, vol. 17, no. 3, pp. 755–770, 2006.
- [16] H. Chen and A. Murray, "A continuous restricted boltzmann machine with an implementable training algorithm," in *Proc. IEE Vis. Image, Signal Process.*, 2003, vol. 150, no. 3, pp. 153–158.
- [17] J. H. Wang and H. Chen, "An embedded probabilistic neural network with on-chip learning capability," in *Proc. IEEE Biomed. Circuits Syst. Conf.*, 2013.
- [18] C. C. Lu and H. Chen, "Current-mode computation with noise in a scalable and programmable probabilistic neural vlsi system," in *Proc. Int. Conf. Artif. Neural Netw.*, 2009.
- [19] J. R. Movellan, P. Mineiro, and R. J. Williams, "A Monte Carlo EM approach for partially observable diffusion processes: Theory and applications to neural networks," *Neural Comput.*, vol. 14, pp. 1507–1544, Jul. 2002.
- [20] Y. S. Hsu, T. J. Chiu, and H. Chen, "Real-time recognition of continuous biomedical signals using the Diffusion Network," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2008, pp. 2628–2633.
- [21] L. O. Chua, T. Roska, T. Kozek, and A. Zandary, "CNN universal chips crank up the computing power," *IEEE Circuits Devices Mag.*, vol. 12, no. 4, pp. 18–28, Jul. 1996.
- [22] C. H. Chien, C. C. Lu, and H. Chen, "Mapping the Diffusion Network into a stochastic system in very large scale integration," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2010.
- [23] T. Serrano-Gotarredona and B. Linares-Barranco, "Log-domain implementation of complex dynamics reaction-diffusion neural networks," *IEEE Trans. Neural Netw.*, vol. 14, pp. 1337–1355, Sep. 2003.
- [24] Y.-D. Wu, S.-J. Lin, and H. Chen, "A log-domain implementation of the Diffusion Network in very large scale integration," in *Advances in Neural Information Processing Systems (NIPS) 23*. Cambridge, MA, USA: MIT Press, 2010, pp. 2487–2495.
- [25] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Trans. ASME J. Basic Eng.*, vol. 83, pp. 95–108, 1961.
- [26] S.-C. Liu, J. Kramer, G. Indiveri, T. Delbrück, and R. Douglas, *Analog VLSI: Circuits and Principles*. Cambridge, MA, USA: MIT Press, 2002.
- [27] M. Banu and Y. Tsvividis, "Floating voltage-controlled resistors in CMOS technology," *Electron. Lett.*, vol. 18, no. 15, pp. 678–679, Jul. 1982.
- [28] C. Toumazou, F. J. Lidgley, and D. G. Haigh, *Analogue IC Design: The Current-Mode Approach*. London, U.K.: Peter Peregrinus Ltd., 1990.
- [29] G. Cauwenberghs, "Delta-sigma cellular automata for analog VLSI random vector generation," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 46, no. 3, pp. 240–250, Mar. 1999.



Yi-Da Wu received his Ph.D degree in 2012 in Department of Electrical Engineering, National Tsing Hua University, Taiwan. His research interests include frequency synthesizer for wireless communication applications, and stochastic neural networks for real-time recognition of biomedical signals. Since 2008, he has been with the Information and Communications Research Laboratories of Industrial Technology Research Institute developing RF circuits for WiMAX applications.



Hsin Chen (M'04) received his Ph.D. degree in Electronics at the Edinburgh University, U.K., in 2004. He is now appointed as an associate professor in the EE Dept., National Tsing Hua University, Taiwan. His research interests include stochastic neuromorphic systems, neuro-electronic interfaces for neural prostheses, and neuro-engineered integrated circuits that could communicate with biological neurons. He is a member of the IEEE CaS and the IEEE EMBS.