Asynchronous Cellular Automaton Based Modeling of Nonlinear Dynamics of Neuron

Hiroyuki Torikai and Takashi Matsubara

Abstract A modeling approach of nonlinear dynamics of neurons by an asynchronous cellular automaton is introduced. It is shown that an asynchronous cellular automaton neuron model can realize not only typical nonlinear response characteristics of neurons but also their underlying occurrence mechanisms (i.e., bifurcation scenarios). The model can be implemented as an asynchronous sequential logic circuit, whose control parameter is the pattern of wires that can be dynamically updated in a dynamic reconfigurable FPGA. An on-FPGA learning algorithm (i.e., on-FPGA rewiring algorithm) is presented and is used to tune the model so that it reproduces nonlinear response characteristics of a neuron.

1 Introduction

The neuron is one of the most sophisticated nonlinear dynamical systems and its mathematical and hardware modelings have been investigated intensively [1–10]. Motivations for the hardware neuron include development of a neural prosthesis chip for clinical applications [9, 10] and development of an artificial neural network chip for engineering applications [1, 5–7]. Major classical approaches of hardware spiking neurons include: (i) implementation of a nonlinear *ordinary differential equation* (ab. ODE) by an analog circuit [1–4] and (ii) implementation of a numerical integration by a digital processor [5–8]. Recently, an alternative neuron modeling approach has been proposed, where the nonlinear dynamics of a neuron is modeled by an asynchronous cellular automaton that is implemented by an asynchronous sequential logic

H. Torikai (✉) · T. Matsubara
Graduate School of Engineering Science, Osaka University, 1-3 Machikaneyama-cho, Toyonaka, Osaka 560-8531, Japan
e-mail: hiroyuki.torikai@ieee.org

T. Matsubara
e-mail: matubara@hopf.sys.es.osaka-u.ac.jp

V. In et al. (eds.), *International Conference on Theory and Application in Nonlinear Dynamics (ICAND 2012)*, Understanding Complex Systems,
DOI: 10.1007/978-3-319-02925-2_9,

circuit [11–15]. The asynchronous sequential logic spiking neuron model (ab. ASN) consists of registers, logic gates, and reconfigurable wires, where the pattern of the wires among the registers and the gates is a control parameter that determines its nonlinear dynamics. Some types of ASNs (e.g., integrate-and-fire type and rotate-and-fire type) have been presented so far and their neuron-like properties have been analyzed intensively [11–15]. In this paper, it is shown that the ASN can realize typical nonlinear responses of neurons [11]. Also, some of our recent new results on learnings of the ASN are presented. Significances of the ASN include the following points. (a) The analog circuit neuron has a continuous time and a continuous state, and the digital processor neuron has a discrete time and a discrete state. On the other hand, the ASN has a discrete state and a continuous (state transition) time. Hence the ASN belongs to a different class of nonlinear dynamical systems from the major classical hardware spiking neuron models. We emphasize that investigation of such a new neuron modeling approach is an important fundamental nonlinear problem. (b) An important control parameter of the analog circuit neuron is a nonlinearity of a circuit element. Hence, its dynamic parameter adjustment is often troublesome. An important control parameter of the digital processor neuron is a coefficient of a nonlinear function. Hence, its dynamic parameter adjustment is possible but needs a numeric data processor. On the other hand, the control parameter of the ASN is the wiring pattern that can be dynamically adjusted based on a dynamic reconfigurable FPGA technology. (c) The ASN can be implemented with less hardware resources (i.e., smaller number of configuration logic blocks) than the digital processor neuron for some reasonable parameter cases. Such a low hardware cost property and the dynamic reconfiguration ability will be significantly useful to develop future applications of the ASN, e.g., a neural prosthesis chip whose area is small and whose parameters can be dynamically updated after implantation and an artificial neural network chip with an on-chip learning capability.

2 Asynchronous Sequential Logic Neuron Model

In this section, an *asynchronous sequential logic spiking neuron model* (ab. ASN), whose diagram is depicted in Fig. 1, is introduced [11–15]. The ASN has the following four registers whose bit lengths are denoted by positive integers N, M, K, and J, respectively.

(1) The *membrane register* is an N-bit bi-directional shift register having an integer state $V \in \mathbf{Z}_N \equiv \{0, \ldots, N-1\}$ encoded by the one-hot coding manner, where "$\equiv$" denotes "is defined by" hereafter. From a neuron model viewpoint, the state V can be regarded as a *membrane potential*.
(2) The *recovery register* is an M-bit bi-directional shift register having an integer state $U \in \mathbf{Z}_M \equiv \{0, \ldots, M-1\}$ encoded by the one-hot coding manner. From a neuron model viewpoint, the state U can be regarded as a *recovery variable*.

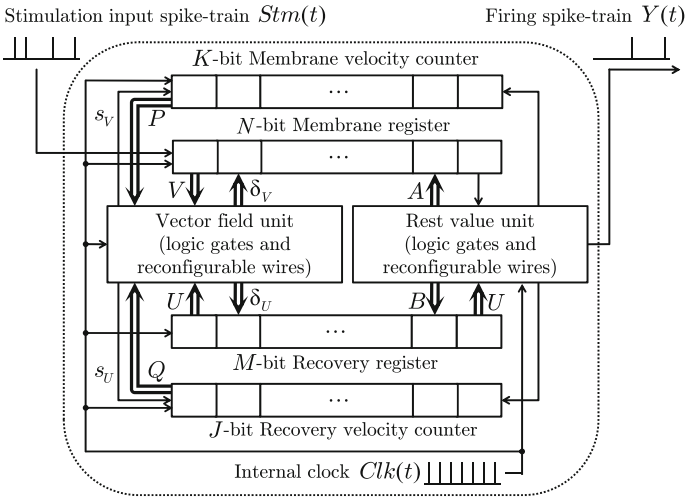

Fig. 1 Asynchronous sequential logic spiking neuron model (ab. ASN)

(3) The *membrane velocity counter* is a K-bit register having an integer state $P \in \mathbf{Z}_K \equiv \{0, \ldots, K-1\}$ encoded by the thermometer coding manner. The state P controls a velocity of the membrane potential V.
(4) The *recovery velocity counter* is a J-bit register having an integer state $Q \in \mathbf{Z}_J \equiv \{0, \ldots, J-1\}$ encoded by the thermometer coding manner.

The state Q controls a velocity of the recovery variable U. The states V, U, P, and Q are clamped to the ranges $[0, N-1]$, $[0, M-1]$, $[0, K-1]$, and $[0, J-1]$, respectively. As shown in Fig. 1, the registers and the velocity counters are connected to each other via the following two memoryless units. (i) The *vector field unit* consists of logic gates and reconfigurable wires. This unit determines the characteristics of a vector field of the states (V, U) as its name implies. (ii) The *reset value unit* consists of logic gates and reconfigurable wires. From a neuron model viewpoint, this unit determines values to which the states (V, U) are reset when the ASN fires, as its name implies. The ASN accepts a periodic *internal clock* $Clk(t)$ described by

$$Clk(t) = \begin{cases} 1 & \text{if } t \pmod 1 = 0, \\ 0 & \text{otherwise}, \end{cases}$$

where $t \in [0, \infty)$ is a continuous time. In the next subsection, autonomous behaviors of the ASN (i.e., behaviors when no stimulation input spike-train $Stm(t)$ is applied) are investigated. After that, in the next subsection, non-autonomous behaviors of the ASN (i.e., behaviors when a stimulation input spike-train $Stm(t)$ is applied) are investigated.

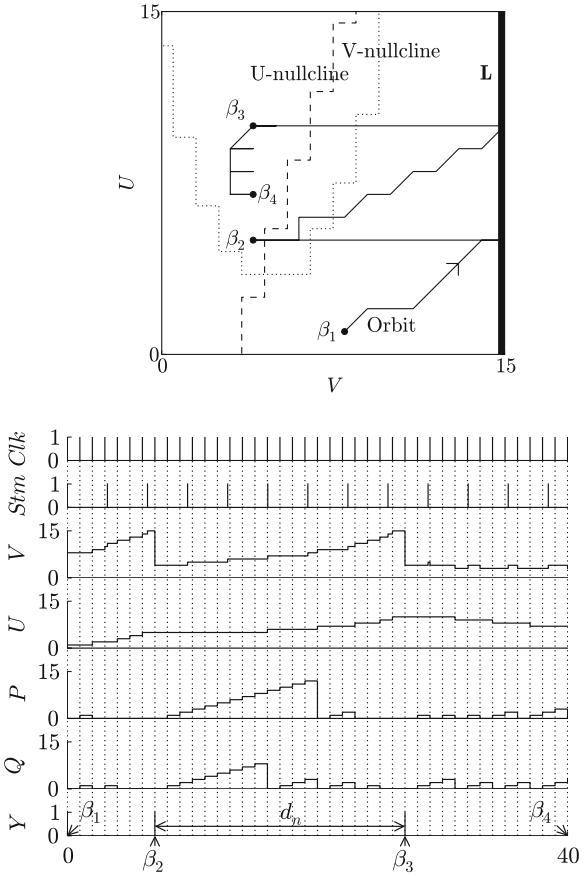

Fig. 2 A phase plane and state transitions. V-nullcline (U-nullcline) is a border between $D_V \in \{-1, 0\}$ and $D_V = 1$ ($D_U \in \{-1, 0\}$ and $D_U = 1$). The bit lengths of the ASN are $N = M = K = J = 16$. The parameters are $\boldsymbol{\Gamma} = (7, 0.3, 0.2, 3, 0.1, 16, 0.5, 0.3, 0)$ defined in (10). A periodic stimulation input spike-train $Stm(t)$ with a frequency 0.312 via the synaptic weight $W = 1$ is applied to the ASN

2.1 Autonomous Behaviors

Let us begin with defining the following subset $\mathbf{L}$ in the state space $\mathbf{Z}_N \times \mathbf{Z}_M$ (see also Fig. 2).

$$\mathbf{L} \equiv \{(V, U) | V = N - 1, U \in \mathbf{Z}_M\} \subset \mathbf{Z}_N \times \mathbf{Z}_M. \tag{1}$$

From a neuron model viewpoint, $\mathbf{L}$ can be regarded as a *firing threshold*. First, let us consider the case of $(V, U) \notin \boldsymbol{L}$. In this case, the vector field unit in Fig. 1 triggers the transition of the states (P, Q) of the velocity counters and the states (V, U) of the registers through signals $(s_V, s_U) \in \{0, 1\}^2$ and $(\delta_V, \delta_U) \in \{-1, 0, 1\}^2$ as follows.

$$
\begin{aligned}
P(t^+) &= \begin{cases} P(t)+1 & \text{if } s_V(t)=0, Clk(t)=1, \\ 0 & \text{if } s_V(t)=1, Clk(t)=1, \\ P(t) & \text{otherwise}, \end{cases} \\
Q(t^+) &= \begin{cases} Q(t)+1 & \text{if } s_U(t)=0, Clk(t)=1, \\ 0 & \text{if } s_U(t)=1, Clk(t)=1, \\ Q(t) & \text{otherwise}, \end{cases} \\
V(t^+) &= \begin{cases} V(t)+\delta_V & \text{if } Clk(t)=1, \\ V(t) & \text{otherwise}, \end{cases} \\
U(t^+) &= \begin{cases} V(t)+\delta_V & \text{if } Clk(t)=1, \\ V(t) & \text{otherwise}, \end{cases}
\end{aligned}
\tag{2}
$$

where $t^+ = \lim_{\varepsilon \to +0} t + \varepsilon$, the velocity counters accept the internal clock $Clk(t)$ and the signals (s_V, s_U), and the registers accept the signals (δ_V, δ_U) from the vector field unit. The signals (s_V, s_U) and (δ_V, δ_U) are generated as follows.

$$
\begin{aligned}
s_V &= \begin{cases} 1 & \text{if } P \geq P_h(V,U), \\ 0 & \text{if otherwise}, \end{cases} \\
s_U &= \begin{cases} 1 & \text{if } Q \geq Q_h(V,U), \\ 0 & \text{if otherwise}, \end{cases} \\
\delta_V &= \begin{cases} D_V(V,U) & \text{if } P \geq P_h(V,U), \\ 0 & \text{otherwise}, \end{cases} \\
\delta_U &= \begin{cases} D_U(V,U) & \text{if } Q \geq Q_h(V,U), \\ 0 & \text{otherwise}, \end{cases} \\
P_h &: \mathbf{Z}_N \times \mathbf{Z}_M \to \mathbf{Z}_K, \ Q_h : \mathbf{Z}_N \times \mathbf{Z}_M \to \mathbf{Z}_J, \\
D_V &: \mathbf{Z}_N \times \mathbf{Z}_M \to \{-1, 0, 1\}, \ D_U : \mathbf{Z}_N \times \mathbf{Z}_M \to \{-1, 0, 1\},
\end{aligned}
\tag{3}
$$

where the functions $P_h(V,U)$, $Q_h(V,U)$, $D_V(V,U)$, and $D_U(V,U)$ are discrete functions that are designed by the following rule.

$$
\begin{aligned}
&\mathscr{F}(V,U) = N(\gamma_1 (V/N - \gamma_2)^2 + \gamma_3 - U/M)/\lambda, \\
&\mathscr{G}(V,U) = \mu M(\gamma_4 (V/N - \gamma_2) + (\gamma_3 + \gamma_5) - U/M)/\lambda, \\
&P_h(V,U) = \lfloor |\mathscr{F}^{-1}(V,U)| \rfloor - 1, \ Q_h(V,U) = \lfloor |\mathscr{G}^{-1}(V,U)| \rfloor - 1, \\
&D_V(V,U) = \mathrm{sgn}(\mathscr{F}(V,U)), \ D_U(V,U) = \mathrm{sgn}(\mathscr{G}(V,U)),
\end{aligned}
\tag{4}
$$

where $(\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5)$ are parameters that characterize nullclines, (λ, μ) are parameters that work as time constants, the function $\lfloor x \rfloor$ gives the integer part of a real number x, $P_h(V,U)$ and $Q_h(V,U)$ are clamped to the ranges $[0, K-1]$ and

Table 1 Summary of the implementation and execution methods of Eqs. (1)–(9), and their orders of hardware costs

Equation	Implement or not	Hardware cost
Eq. (2)		$O(1)$
Eq. (3)	Implemented as	$O(N \times M)$
Eqs. (1) and (5)	logic gates and	$O(N + M)$
Eq. (6)	reconfigurable wires	$O(N + M)$
Eq. (8)		$O(1)$
Eq. (9)		$O(1)$
Eq. (4)	Wiring pattern setting rules	
Eq. (7)	(not implemented as a part of ASN)	

$[0, J-1]$, and the signum function $\mathrm{sgn}(x)$ gives the sign of a real number x, respectively.

Second, let us consider the case of $(V, U) \in \mathbf{L}$. In this case, the reset value unit in Fig. 1 triggers the reset of the states (P, Q) of the velocity counters and the states (V, U) of the registers through integer signals $(A, B) \in \mathbf{Z}_N \times \mathbf{Z}_M$ encoded by the one-hot coding manners as follows.

$$\begin{aligned}&(P(t^+), Q(t^+), V(t^+), U(t^+)) = \\ &\begin{cases}(0, 0, A, B) & \text{if } (V, U) \in \mathbf{L}, Clk(t) = 1, \\ (P(t), Q(t), V(t), U(t)) & \text{otherwise},\end{cases}\end{aligned} \tag{5}$$

where the signals (A, B) are generated as follows.

$$\begin{aligned}&(A, B) = (\mathscr{A}, \mathscr{B}(U)) \\ &\mathscr{A} \in \mathbf{Z}_N, \quad \mathscr{B}(U) : \mathbf{Z}_M \to \mathbf{Z}_M,\end{aligned} \tag{6}$$

where $\mathscr{A}$ is an integer and $\mathscr{B}(U)$ is a discrete function that are designed by the following rule.

$$\mathscr{A} = \lfloor \rho_1 N \rfloor, \quad \mathscr{B}(U) = U + \lfloor \rho_2 M \rfloor, \tag{7}$$

where (ρ_1, ρ_2) are parameters, and $\mathscr{A}$ and $\mathscr{B}(U)$ are clamped to the ranges $[0, N-1]$ and $[0, M-1]$, respectively. Repeating the resets, the ASN generates the following *firing spike-train* $Y(t)$.

$$Y(t) = \begin{cases}1 & \text{if } (V(t), U(t)) \in \mathbf{L}, Clk(t) = 1, \\ 0 & \text{otherwise}.\end{cases} \tag{8}$$

Note that Eqs. (2) and (5) represent the discrete state transitions triggered by the discrete signals, and Eqs. (3) and (6) represent the discrete functions. Also, Eq. (8) with Eq. (1) is the discrete function. Hence, they can be implemented by logic gates and reconfigurable wires, where the functional relations are determined by the wiring

patterns in the vector field unit and the reset value unit (see Table 1). On the other hand, Eqs. (4) and (7) represent the parameter (i.e., wiring pattern) setting rules and are not implemented as a part of the ASN (see Table 1).

2.2 Non-autonomous Behaviors

Let us now apply the following stimulation input spike-train $Stm(t)$ to the ASN.

$$Stm(t) = \begin{cases} 1 & \text{if } t = t_1, t_2, \ldots, \\ 0 & \text{otherwise,} \end{cases}$$

where $t = t_1, t_2, \ldots$ are input spike positions. From a neuron model viewpoint, the stimulation input spike-train $Stm(t)$ can be regarded as a *stimulation input*. A stimulation input spike $Stm = 1$ induces the transition of the membrane potential V as follows.

$$V(t^+) = V(t) + W \cdot Stm(t), \tag{9}$$

where $W \in \{1, -1\}$ is a parameter that can be regarded as a *synaptic weight* and $W = 1$ ($W = -1$) implies that the stimulation weight is excitatory (inhibitory). Note that the membrane register of the previous model [14] accepts the single signal $Stm(t)$ only and the model has the excitatory synaptic weight $W = 1$ only. In contrast, the membrane register of the ASN accepts the two signals W and $Stm(t)$ and the ASN has both the excitatory and the inhibitory synaptic weights $W \in \{1, -1\}$. Note also that Eq. (9) represents the discrete state transitions and thus is implemented by logic gates and reconfigurable wires (see Table 1). Figure 2 shows basic non-autonomous behaviors of the ASN, where the V-nullcline (U-nullcline) is a border between $D_V \in \{-1, 0\}$ and $D_V = 1$ ($D_U \in \{-1, 0\}$ and $D_U = 1$). As a result, the dynamics of the ASN is described by Eqs. (2)–(9), and is characterized by the following parameters.

$$N,\ M,\ K,\ J,\ \boldsymbol{\Gamma} = (\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \lambda, \mu, \rho_1, \rho_2). \tag{10}$$

3 Reproduction of Inhibitory Dynamic Response Behaviors

For simplicity, we focus on the following periodic stimulation input spike-train $Stm(t)$.

$$Stm(t) = \begin{cases} 1 & \text{if } (t + \theta_0) \pmod{f_S^{-1}} = 0, \\ 0 & \text{otherwise,} \end{cases}$$

where f_S is an input frequency, $\theta_0 \in [0, f_S^{-1})$ is an initial input phase, and a *post-synaptic stimulation input* I to the ASN is defined as

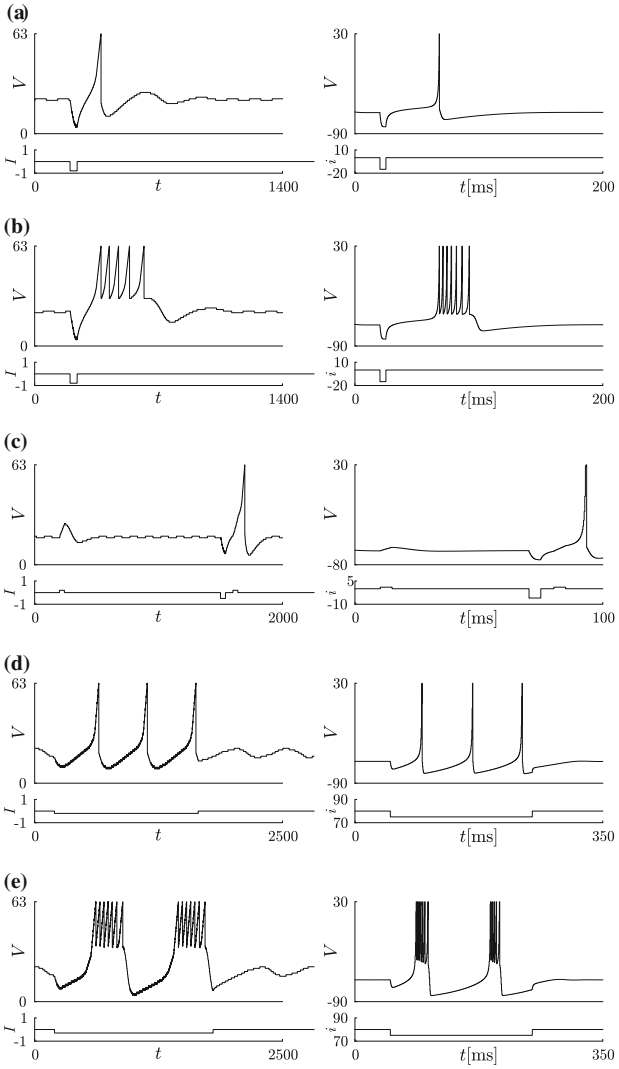

Fig. 3 Numerical simulation results of inhibitory dynamic response behaviors of the ASN (each *left* figure) and those of the Izhikevich simple model [16] (each *right* figure), where v denotes the membrane potential and i denotes the current input. **a**–**e** correspond to (**a**)–(*e*) in Table 2, respectively. The bit lengths of the ASN are $N = M = K = J = 64$. The parameters $\boldsymbol{\Gamma}$ of the ASN and the heights of the post-synaptic stimulation I are as the followings. **a** $\boldsymbol{\Gamma} = (7, 0.3, 0.2, 3, -0.1, 64, 0.5, 0.3, 0)$, $I = -0.8$. **b** $\boldsymbol{\Gamma} = (7, 0.3, 0.2, 3, -0.1, 64, 0.5, 0.48, -0.4)$, $I = -0.8$. **c** $\boldsymbol{\Gamma} = (7, 0.3, 0.2, 3, 0.1, 64, 0.5, 0.3, 0)$, $I = 0.2, -0.5$. **d** $\boldsymbol{\Gamma} = (7, 0.3, 0.5, -5, 0, 64, -0.2, 0.3, 0.1)$, $I = -0.2$. **e** $\boldsymbol{\Gamma} = (7, 0.3, 0.5, -5, 0, 64, -0.1, 0.55, -0.1)$, $I = -0.3$. The parameter values of the Izhikevich simple model can be found in [16]

$$I = f_S \cdot W.$$

According to [16, 17], biological and model neurons typically exhibit dynamic response behaviors (i.e., waveforms of the membrane potential in response to the stimulation input) that can be classified into fifteen excitatory types and five inhibitory types as shown in Table 2. It has been shown the ASN can reproduce all the twenty types of dynamic response behaviors. In this paper, we focus on the following inhibitory ones (a)–(e), which are shown in Table 2 [11]. (a) *A rebound spike* is a spike induced by an inhibitory stimulation input. (b) *A rebound burst* is a burst induced by an inhibitory stimulation input. (c) *Threshold variability* is a phenomenon that whether the stimulation input of the same strength induces a spike depends on the preceding inhibitory stimulation input. (d) *Inhibition-induced spiking* is persistent spike-train generation while an inhibitory stimulation input is applied. (e) *Inhibition-induced bursting* is persistent burst-train generation while an inhibitory stimulation input is applied. Numerical simulation results corresponding to the above behaviors (a)–(e) are shown in Fig. 3a–e, respectively.

4 Learning and Neural Responses

This section shows that the ASN can reproduce desired responses characteristics (i.e., relationships between the stimulation strength and the average firing frequency) obtained from a biological or model neuron, which is called a *teacher* neuron. As the teacher neuron, Izhikevich's resonator model [17] is used, whose stimulation strength is denoted by I. The control parameter, i.e., the wiring pattern, of the ASN is dynamically updated to reproduce the response characteristics of the teacher, where the distance between the responses of them is defined by using the metric-space analysis [18]. This procedure is called a *learning* hereafter. Figure 4 shows the responses characteristics of the teacher, the ASN before the learning, and the ASN after the learning. The response characteristics of the ASN before the learning is different from that of the teacher. The response characteristics of the ASN after the learning is similar to that of the teacher. This indicates that the learning enables the ASN to reproduce the responses characteristics of the teacher. More detailed investigations on the learning will be presented in our future works.

The ASN is implemented on Xilinx's FPGA Veitex-5 XUPV5-LX110T mounted on Digilent's OpenSPARC evaluation platform. The FPGA-implemented ASNN occupies 123 slices (each slice includes four 6-input LUTs and four FFs) of the FPGA device. For comparison, the teacher is also implemented on the same FPGA device using a forward Euler numerical integration with a time step 3 kHz. The implemented teacher occupies 532 slices, where the resolutions of the states are 16-bit binary fixed point numbers. These facts indicate that the ASN requires less hardware resources then a digital processor neuron. More detailed investigations on the hardware will be presented in our future works.

Table 2 The table summarizes reproduction abilities of dynamic response behaviors by typical neuron models and our models, where "+" denotes "reproducible", "−" denotes "not reproducible", and "·" denotes "partially reproducible". Each empty square denotes that sufficient parameter and initial value conditions are unknown but the model satisfies necessary conditions in principle [16]. Each dynamic response behavior is as follows. **a** Rebound spike. **b** Rebound bursting. **c** Threshold variability. **d** Inhibition-induced spiking. **e** Inhibition-induced bursting. **f** Tonic spiking. **g** Phasic spiking. **h** Tonic bursting. **i** Phasic bursting. **j** Mixed mode. **k** Spike frequency adaptation. **l** Class 1 excitable. **m** Class 2 excitable. **n** Spike latency. **o** Subthreshold oscillation. **p** Resonator. **q** Integrator. **r** Bistability. **s** Depolarizing after-potential. **t** Accommodation

	Inhibitory dynamic response behaviors					Excitatory dynamic response behaviors														
Neuron model	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)	(l)	(m)	(n)	(o)	(p)	(q)	(r)	(s)	(t)
Izhikevich [17]	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
Hodgkin Huxley [19]	+	+	+	+		+	+	+			+	+	+	+	+	+	+	+	+	+
ASN [11–13, 15]	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

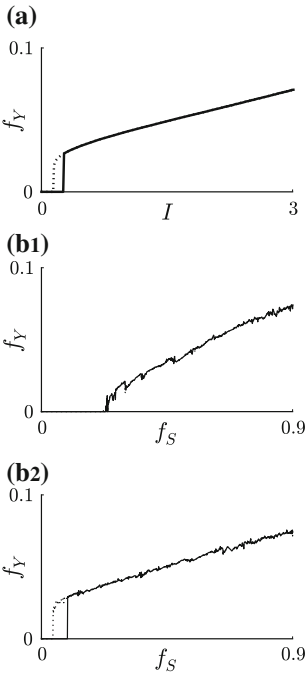

Fig. 4 Learning. **a** Teacher neuron = Izhikevich's resonator model. **b1** The ASN before learning. The sizes of registers are 32 and the internal clock frequency is $f_C = 3\,\text{kHz}$. **b2** The ASN after learning

5 Conclusions

The asynchronous sequential logic spiking neuron model (ab. ASN) was introduced. It was shown that the ASN can reproduce the typical twenty types of the dynamic response behaviors of neurons. Especially, in this paper, it was demonstrated that

the ASN can reproduce the five inhibitory dynamic response behaviors. Furthermore, it was shown that the learning algorithm enables the ASN to automatically reproduce the nonlinear response characteristics of an ODE-based neuron model. It was also confirmed that the ASN can be implemented with less hardware resources than a digital processor neuron for a reasonable parameter case. These reproduction abilities of neural dynamics and the low hardware cost property will be the keys to developing future applications of the ASN. Future problems include: (a) bifurcation analyses of the ASN, (b) clarification of relationships between the parameters of the ASN and experimentally measurable parameters of biological neurons, (c) development of a multi-compartment neuron model based on the ASN, including register-dynamics models of synaptic connections, and (d) development of a network of multi-compartment ASNs and its bio-inspired learning mechanisms such as the spike-timing dependent plasticity.

The authors would like to thank Professor Toshimitsu Ushio of Osaka University for valuable discussions. This work is partially supported by Toyota Riken Scholar and KAKENHI (24700225).

References

1. R. Vogelstein, U. Mallik, T. Vogelstein, G. Cauwenberghs, Dynamically reconfigurable silicon array of spiking neurons with conductance-based synapses. IEEE Trans. Neural Networks **18**(1), 253–265 (2007)
2. H. Chen, S. Saïandghi, L. Buhry, S. Renaud, Real-time simulation of biologically realistic stochastic neurons in vlsi. IEEE Trans. Neural Networks **21**(9), 1511–1517 (2010)
3. A. Basu, S. Ramakrishnan, C. Petre, S. Koziol, S. Brink, P.E. Hasler, Neural dynamics in reconfigurable silicon. IEEE Trans. Biomed. Circuits Syst. 4(5), 311–319 (2010)
4. J.V. Arthur, K.A. Boahen, Silicon-neuron design: a dynamical systems approach. IEEE Trans. Circuits Syst. I, **58**(5), 1034–1043 (2011)
5. E. Ros, E. Ortigosa, R. Agís, R. Carrillo, M. Arnold, Real-time computing platform for spiking neurons (RT-spike). IEEE Trans. Neural Networks **17**(4), 1050–1063 (2006)
6. R. Weinstein, M. Reid, R. Lee, Methodology and design flow for assisted neural-model implementations in FPGAs. IEEE Trans. Neural Syst. Rehabil. Eng. **15**(1), 83–93 (2007)
7. M. Pearson, G. Pipe, B. Mitchinson, K. Gurney, C. Melhuish, I. Gilhespy, M. Nibouche, Implementing spiking neural networks for real-time signal-processing and control applications: a model-validated FPGA approach. IEEE Trans. Neural Networks **18**(5), 1472–1487 (2007)
8. J.M. Nageswaran, N. Dutt, J.L. Krichmar, A. Nicolau, A.V. Veidenbaum, A configurable simulation environment for the efficient simulation of large-scale spiking neural networks on graphics processors. Neural Netw. **22**(5–6), 791–800 (2009)
9. T. Berger, D. Glanzman, *Toward Replacement Parts for the Brain: Implantable Biomimetic Electronics as Neural Prostheses* (The MIT Press, Cambridge, 2005)
10. Z. Zumsteg, C. Kemere, S. O'Driscoll, G. Santhanam, R. Ahmed, K. Shenoy, T. Meng, Power feasibility of implantable digital spike sorting circuits for neural prosthetic systems. IEEE Trans. Neural Syst. Rehabil. Eng. **13**(3), 272–279 (2005)
11. T. Matsubara, T. Torikai, Neuron-like responses and bifurcations of a generalized asynchronous sequential logic spiking neuron model. IEICE Trans. Fundam. **E95-A**(8), 1317–1328 (2012)
12. T. Hishiki, H. Torikai, A Novel Rotate-and-Fire Digital Spiking Neuron And its Neuron-like Bifurcations and Responses. IEEE Trans. Neural Networks **22**(5), 752–767 (2011)

13. T. Matsubara, H. Torikai, T. Hishiki, A generalized rotate-and-fire digital spiking neuron model and its on-FPGA learning. IEEE Trans. Circuits Syst. II Express Briefs **58**(10), 677–681 (2011)
14. T. Matsubara, H. Torikai, A novel asynchronous digital spiking neuron model and its various neuron-like bifurcations and responses, in *Proceedings of International Joint Conference on, Neural Networks*, 2011, pp. 741–748
15. S. Hashimoto, H. Torikai, A novel hybrid spiking neuron: bifurcations, responses, and on-chip learning. IEEE Trans. Circuits Syst. I **57**(8), 2168–2181 (2010)
16. E.M. Izhikevich, Which model to use for cortical spiking neurons? IEEE Trans. Neural Networks **15**(5), 1063–1070 (2004)
17. E.M. Izhikevich, *Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting* (The MIT Press, Cambridge, 2006)
18. J. Victor, K. Purpura, Metric-space analysis of spike trains: theory, algorithms and application. Netw. Comput. Neural Syst. **8**(2), 127–164 (May 1997)
19. A.L. Hodgkin, A.F. Huxley, A quantitative description of membrane current and its application to conduction and excitation in nerve. J. physiol. **117**(4), 500–544 (1952)