

Experimental realizations of the HR neuron model with programmable hardware and synchronization applications

Nimet Dahasert · İsmail Öztürk · Recai Kiliç

Received: 1 June 2012 / Accepted: 15 September 2012 / Published online: 20 October 2012
© Springer Science+Business Media Dordrecht 2012

Abstract This study presents experimental realizations of the HR neuron model with programmable hardware and synchronization applications. The HR neuron model exhibiting burst, spike, and chaotic dynamics has been implemented with both FPAA (Field Programmable Analog Array) and FPGA (Field Programmable Gate Array) devices. These devices provide flexible design possibilities such as reducing the complexity of design, real-time modification, software control and adjustment within the system. And it is also examined experimentally that how the synchronization of two HR neurons are able to achieve by using these hardware. The experimental results obtained from FPAA and FPGA based realizations agree with the numerical simulations very well.

Keywords Hindmarsh–Rose neuron model · Synchronization · FPAA · FPGA

1 Introduction

Describing the dynamic behavior of an individual neuron mathematically is one of the important issues in computational neuroscience. Many artificial neuron models were proposed to model neurons in the nervous system to express the ion flows through the surface of membrane, examine exactly how brain works, and simulate the activities of the brain. The first neuron model is the classical neuron model proposed by McCulloch and Pitts in 1943 [1]. Several assumptions such as “all or nothing” principle were made in this model for easy calculation, and these assumptions constitute a basis for other models. In 1952, the most successful and widely used neuron model, the Hodgkin–Huxley model was presented in the literature [2]. The ionic mechanism and electrical current on membrane surface were discussed in this comprehensive model. After that, the FitzHugh–Nagumo neuron model, which is the simplified type of the Hodgkin–Huxley neuron model, was proposed [3]. In 1972, Nagumo and Sato [4] defined a neuron model and the weakly coupled Wilson–Cowan neuron model was defined in the same year [5]. The Morris–Lecar neuron model proposed in 1981 is a conductance based neuron model [6]. The Hindmarsh–Rose [7] and Izhikevich [8] neuron models were proposed in 1984 and in 2003, respectively. The studies that deal with the behavior of the collective neurons rather than an individual neuron have come forward, because the biological information process and production of regular rhyth-

N. Dahasert · İ Öztürk · R. Kiliç (✉)
Department of Electrical & Electronic Engineering,
Erciyes University, 38039 Kayseri, Turkey
e-mail: kilic@erciyes.edu.tr

N. Dahasert
e-mail: nimetdahasert@erciyes.edu.tr

İ Öztürk
e-mail: ismail.ozturk@amasya.edu.tr

mic activity are related to the cooperative behavior of neurons [9–14].

To obtain and examine physically the behavior of an individual or collective neuron model is very important, because it is very difficult to measure the individual signal of a real neuron and to identify the interaction of the collective neurons in the living body except for some applications. Several alternative system approaches such as numerical modeling and hardware implementations, which allow us to observe the fire patterns or synchronizations of neurons, have become crucial. While the software examinations of the biological neuron models simulate the behavior of the neurons, hardware realizations are able to emulate the behavior of an individual biological neuron or coupled neurons with real time adaptability. Furthermore, hardware realizations of neuron models can be used in practical applications such as bioinspired robotic systems and CPGs (Central Pattern Generators) [15, 16]. While there are mostly discrete analog and digital hardware implementations of neuron models [17–24], there are few studies about hardware implementation of synchronal neurons [25–27], because hardware realization of the synchronous neural system is very complex and hard to implement in terms of the circuit network structure and parameter adjustability. For these reasons, alternative hardware solutions should be considered. Nowadays, FPAA and FPGA based system solutions offer a good alternative for the versatile implementations of the complex structures such as neural systems. Although there are some studies utilizing FPAA or FPGA implementation of the different neuron models [28–31], there is no comprehensive study about the programmable and reconfigurable implementation of the HR neuron model and its synchronization to the best of our knowledge. In this study, Hindmarsh–Rose neuron model (HR), which exhibits several fire patterns of the neuron and widely used in synchronization studies, is realized with both FPAA and FPGA devices by utilizing their programmable and reconfigurable features. The usage of both FPAA and FPGA for modeling the HR neuron reflects another originality of this study. And also, the synchronization of two HR neurons is implemented with these devices experimentally by using electrical coupling method.

The HR neuron model and its dynamics are briefly introduced in Sect. 2. The synchronization issues of biological neurons and the coupled HR neurons are

handled in Sect. 3. The details of FPAA and FPGA based realizations of HR neuron model and synchronization applications are given in Sect. 4. Some concluding remarks and comparisons are presented in the last section.

2 HR neuron model

A real biological neuron has very different dynamic behavior such as quiescent, spiking, bursting, and chaotic. If the input signal applied to the neuron is below a threshold value, the neuron does not generate an output response and this behavior is known as quiescent. If there is a regular series at the output of the neuron, this behavior is spiking. If the output resembles ensembles of spikes separated by a certain period, the neuron exhibits bursting behavior. And if the output signal of the neuron is produced in the chaos mode; its behavior is also chaotic. In terms of reflecting these properties of neurons, the most successful neuron model is the Hodgkin–Huxley model [2], but it is a very complex model. The FitzHugh–Nagumo neuron model [3] is the simplified type of the Hodgkin–Huxley neuron model. This model is defined by two differential equations, but it does not have the capability of exhibiting certain behavior of neurons. The HR neuron model was derived from the FitzHugh–Nagumo neuron model. Despite its simplicity, the HR neuron model is able to exhibit several dynamic behavior of a real neuron [32–36] and it is defined by the following differential equations:

$$\begin{aligned}\dot{x} &= f_x(x, y, z) = y - x^3 + bx^2 + I - z \\ \dot{y} &= f_y(x, y, z) = 1 - 5x^2 - y \\ \dot{z} &= f_z(x, y, z) = \mu(s(x - x_{rest}) - z)\end{aligned}\quad (1)$$

where (I) represents the membrane input current, (b) controls the transition between bursting and spiking, (μ) controls the spiking frequency and the number of spikes per burst in the case of spiking and bursting respectively, (s) adjusts adaptation, and (x_{rest}) is the resting potential.

By fixing the parameters $\mu = 0.01$, $s = 4$, $x_{rest} = -1.6$, the HR neuron model dynamics are investigated with a numerical simulation tool, namely MATLAB™, depending on parameters (I) and (b). In these numerical simulations, the Dormand–Prince integration algorithm was used. The time step is based

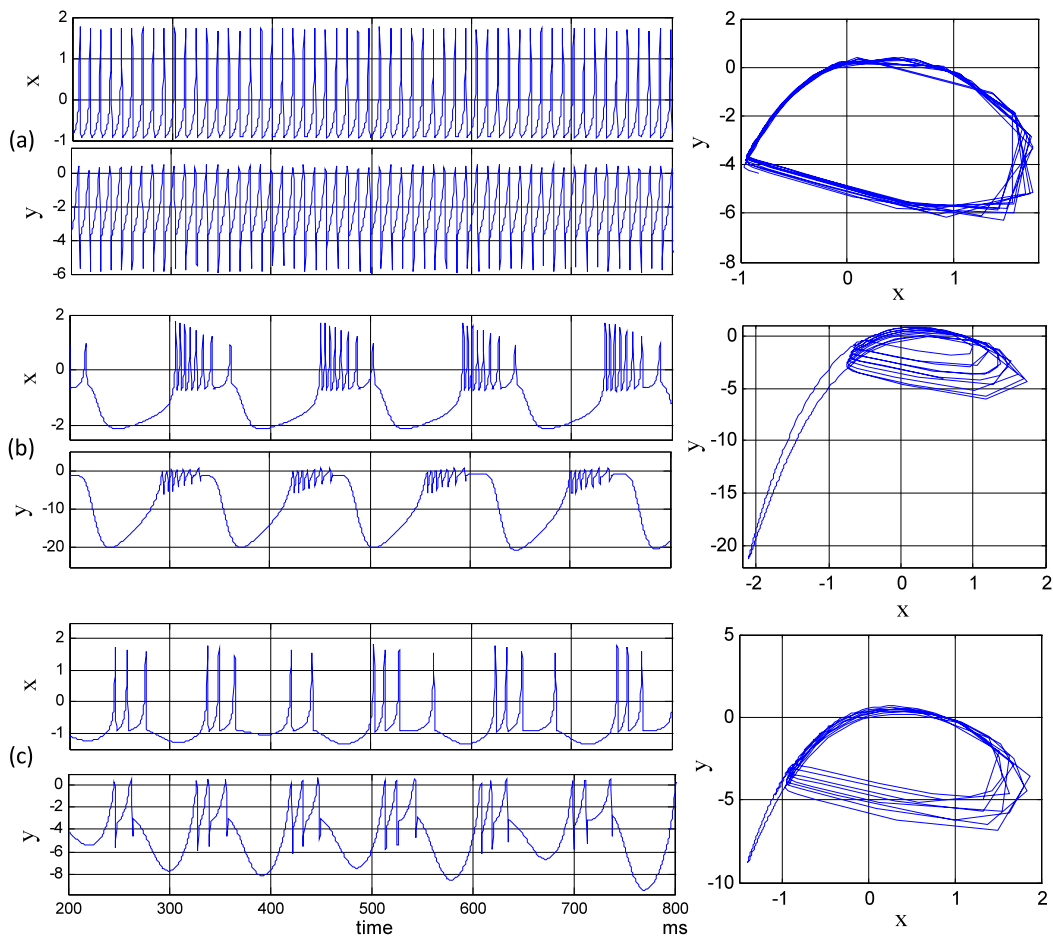


Fig. 1 HR neuron dynamics and x - y plane demonstration for (a) Spiking mode, (b) Bursting mode, and (c) Chaotic mode

on a variable step solver with a relative tolerance of 10^{-3} . The 200 ms transitory responses are discarded for the simulations in Fig. 1. While $b = 2.96$, $I = 5$ are set to observe the spiking behavior as shown in Fig. 1a, the bursting behavior in Fig. 1b is obtained for the parameter values of $b = 2.6$, $I = 2.66$. The (b) and (I) parameters are chosen 2.96 and 3, respectively, to observe the chaotic behavior in Fig. 1c.

3 Electrical coupling of HR neurons and synchronization

Real neurons communicate with each other through their synapses and there are two ways to couple them: chemical coupling and electrical coupling. Chemical coupling is slower than electrical coupling and electrical coupling is used in the operations needed to the fast

transmissions [37]. The experimental studies carried out on the living neurons reveal two important cases about synchronization on neurons:

First case: Even if a neuron is isolated from its partners with different methods such as dc hyperpolarization or photo-inactivating, it still maintains to generate fire patterns by itself [38, 39].

Second case: When two neurons are coupled with an artificial coupling method, the artificial coupling parameters influence the activity patterns and synchronization of neurons. On the other hand, the injecting external dc currents to neurons change their characteristics in both cases [12].

The features of these real neurons and coupling methods are emulated with various studies thanks to the biological neuron models. For example, the artificial coupling method can be constituted with a numerical simulation easily. To investigate the behaviors

of the coupled neurons numerically, one of the most convenient biological model is HR neuron model. The HR neurons can be coupled with each other by using the artificial coupling methods such as electrical and synaptic coupling [40, 41]. Typical coupling mechanism is illustrated in Fig. 2.

The two coupled HR neurons can be defined by the following equations:

$$\begin{aligned} \dot{x}_1 &= f_x(x_1, y_1, z_1) - g_s \sigma(x_1) \hat{c}_{12} \gamma(x_1, x_2) \\ \dot{y}_1 &= f_y(x_1, y_1, z_1) \\ \dot{z}_1 &= f_z(x_1, y_1, z_1) \\ \dot{x}_2 &= f_x(x_2, y_2, z_2) - g_s \sigma(x_2) \hat{c}_{21} \gamma(x_2, x_1) \end{aligned} \quad (2)$$

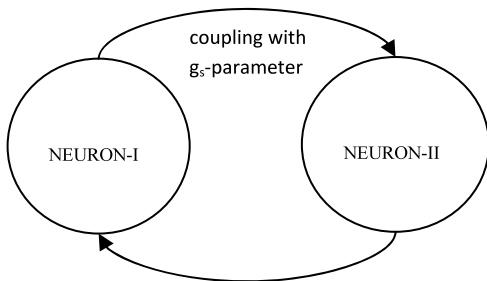


Fig. 2 Coupled two HR neurons

$$\begin{aligned} \dot{y}_2 &= f_y(x_2, y_2, z_2) \\ \dot{z}_2 &= f_z(x_2, y_2, z_2) \end{aligned}$$

where $\hat{c}_{12} = \hat{c}_{21} = 1$. The parameter g_s in Eq. (2) is known as the strength of coupling. In the case of the synaptic coupling, for the first neuron $\gamma(x_1, x_2) = \gamma_s(x_2) = (1/(1 + e^{-k(x_2 - \theta_s)}))$, $\sigma(x_1) = -(x_1 - V_s)$, and for the second neuron, $\gamma(x_2, x_1) = \gamma_s(x_1) = (1/(1 + e^{-k(x_1 - \theta_s)}))$, and $\sigma(x_2) = -(x_2 - V_s)$. Two neurons with synaptic coupling were simulated according to $k = 10, V_s = -2, \theta_s = -0.28$ by using a simulation model where no control method is applied. Numerical simulation results are given in Fig. 3. The neurons coupled with synaptic coupling usually exhibit asynchronous behavior [40] and various control methods are proposed to be synchronized to them [41–43].

In the case of electrical coupling for the first neuron $\gamma(x_1, x_2) = \gamma_e(x_1, x_2) = (x_2 - x_1)$, $\sigma(x_1) = 1$ and the second neuron $\gamma(x_2, x_1) = \gamma_e(x_2, x_1) = (x_1 - x_2)$, $\sigma(x_2) = 1$. When two HR neurons are coupled with electrical coupling, they can exhibit synchronous behaviors, but coupling parameters may affect the behavior. For example, antiphase synchrony behavior is observed between neurons for the negative values of g_s as in Fig. 4. If < 0.52 , the neurons

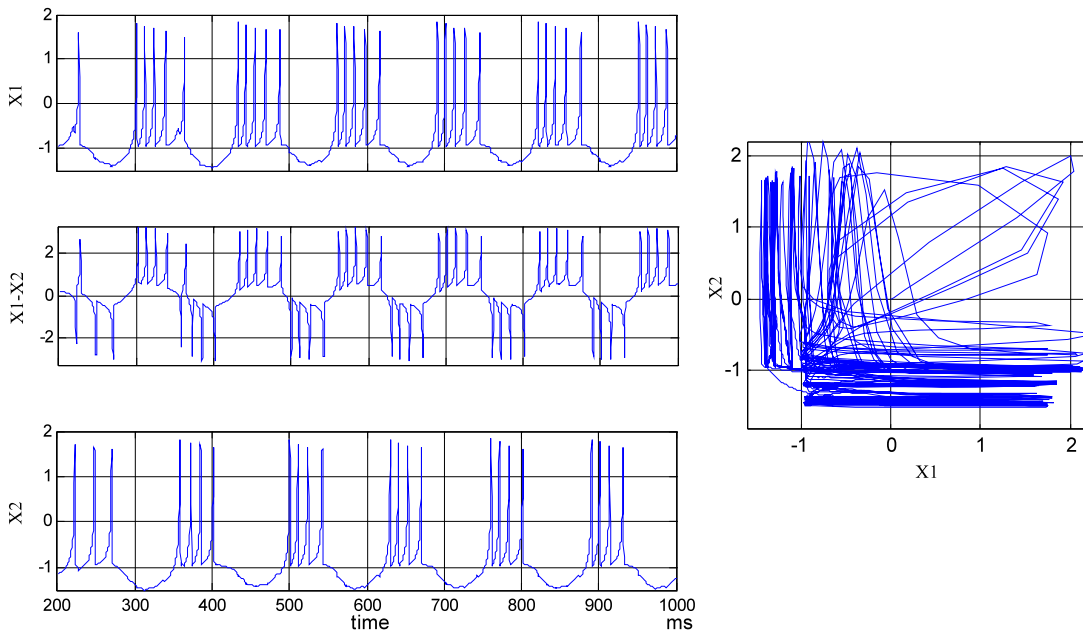


Fig. 3 The behaviors of synaptic coupled HR neurons: membrane potentials (x_1 and x_2) of the coupled neurons, the error between the membrane potentials, and $x_1 - x_2$ plane demonstration

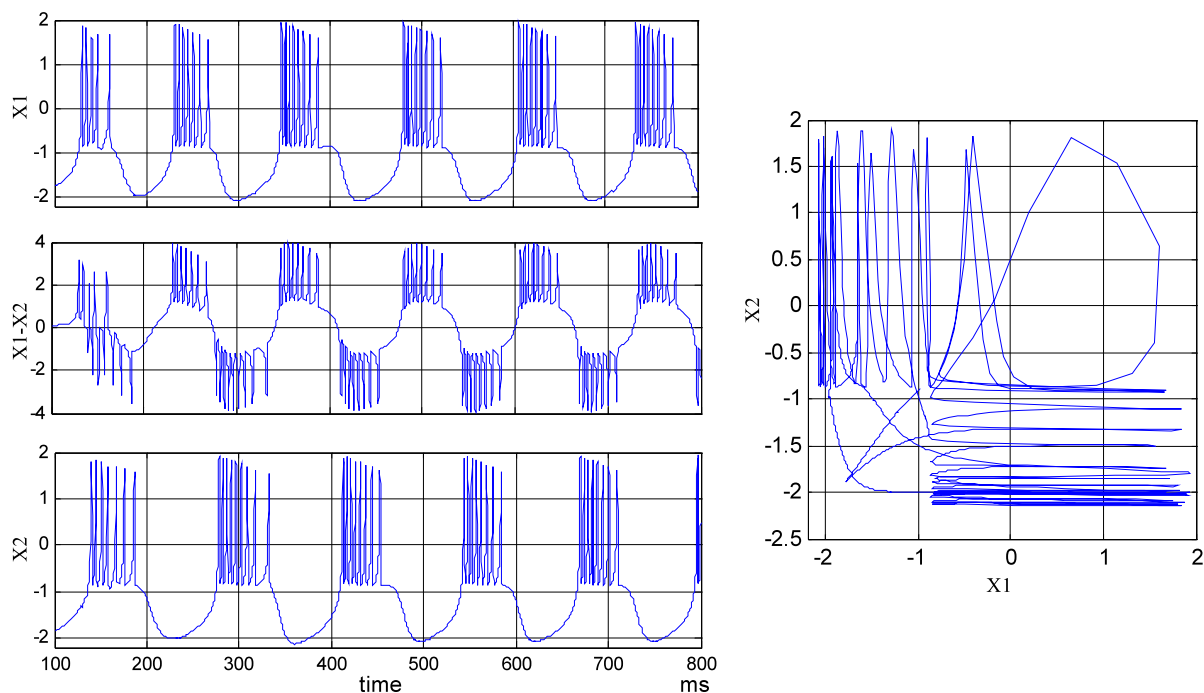


Fig. 4 The antiphase synchrony behavior of electrical coupled HR neurons for $g_s = -0.2$: membrane potentials (x_1 and x_2) of the coupled neurons, the error between the membrane potentials, and $x_1 - x_2$ plane demonstration

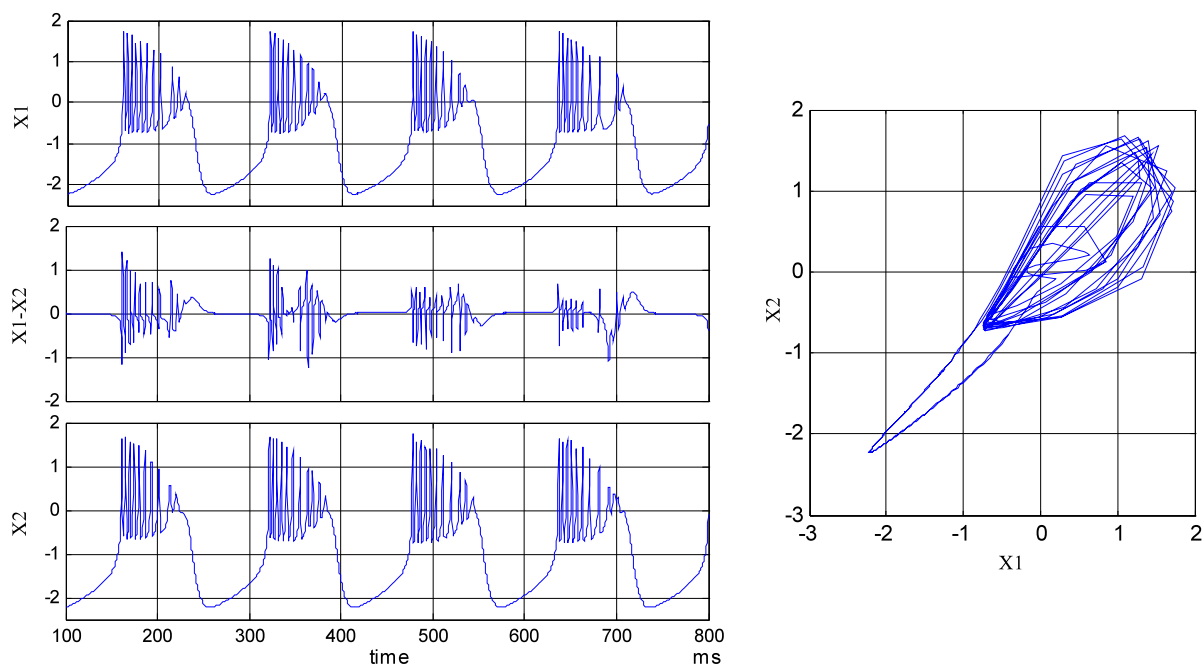


Fig. 5 The asynchronous behaviors of electrical coupled HR neurons for $g_s = 0.2$: membrane potentials (x_1 and x_2) of the coupled neurons, the error between the membrane potentials, and $x_1 - x_2$ plane demonstration

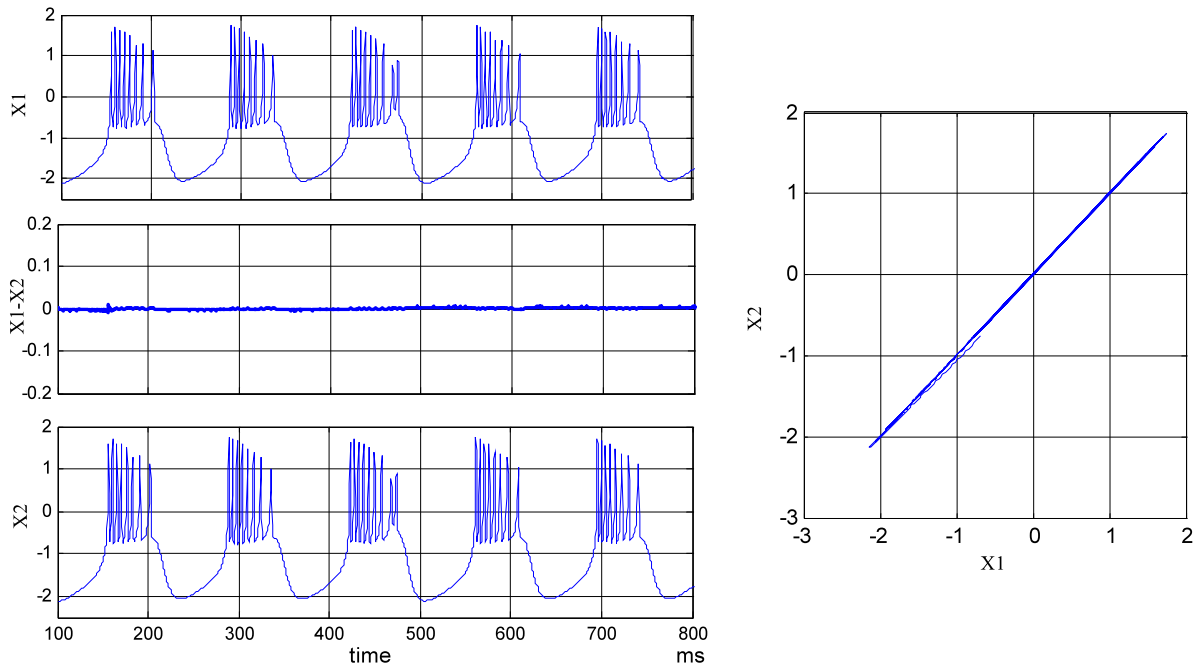


Fig. 6 Synchronous behavior of electrical coupled HR neurons for $g_s = 0.8$: membrane potentials (x_1 and x_2) of the coupled neurons, the error between the membrane potentials, and $x_1 - x_2$ plane demonstration

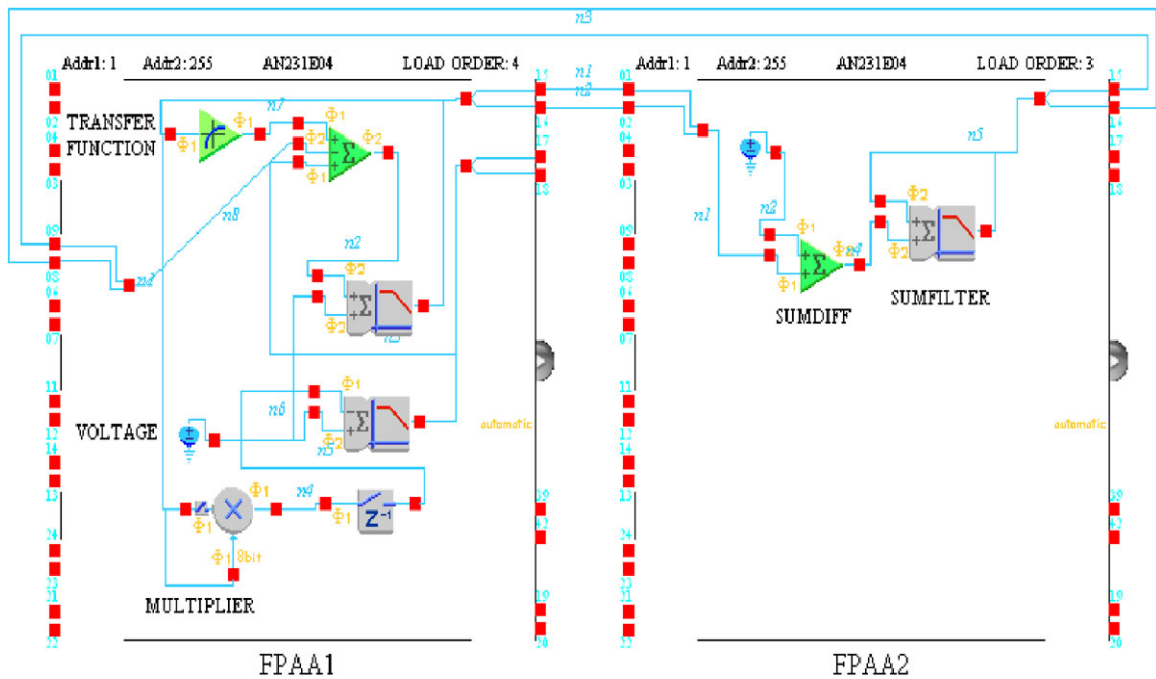
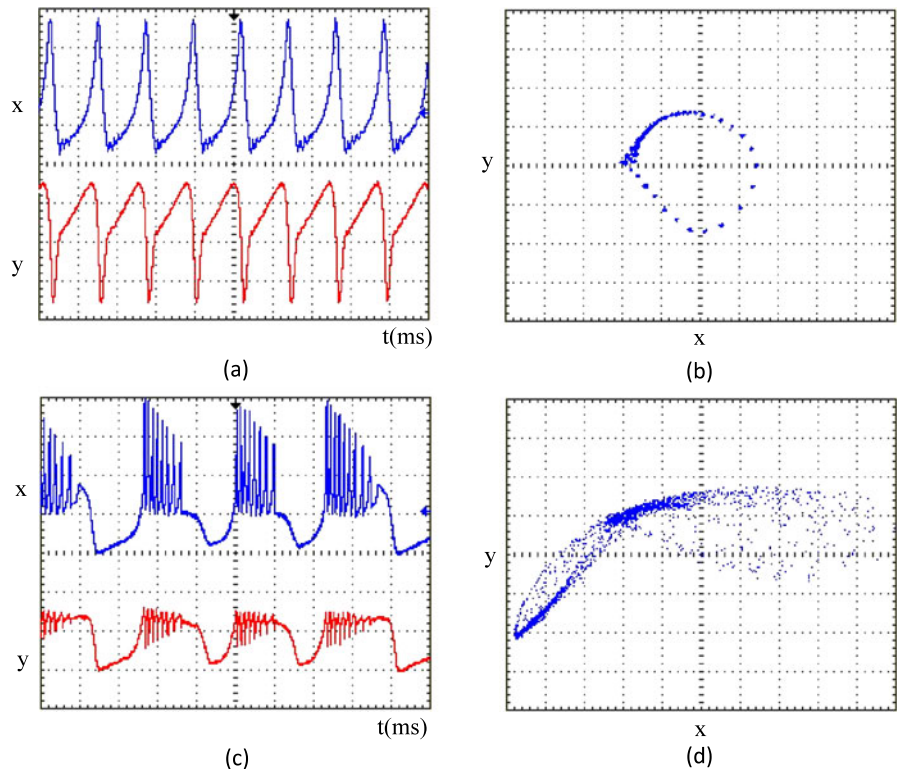


Fig. 7 The FPAA implementation scheme of HR neuron model

Fig. 8 The experimental results of the FPAA based HR neuron model
(a) Spiking mode, x : 0.5 mV/div, y : 0.5 mV/div, time/div: 1 ms, **(b)** x - y phase portrait illustration of the spiking mode, x : 0.5 mV/div, y : 0.5 mV/div
(c) Bursting mode, x : 0.5 mV/div, y : 0.5 mV/div, time/div: 10 ms, **(d)** x - y phase portrait illustration of the bursting mode, x : 0.2 mV/div, y : 0.2 mV/div



are asynchronous as shown in Fig. 5. Otherwise, they exhibit synchronous behavior as in Fig. 6 without no need to use any control method [41].

4 Programmable analog and digital implementations of the HR neuron model and coupled HR neurons

In this study, HR neuron model and the two electrical coupled neurons have been implemented with FPAA as an example of analog realization, and then they have been implemented with FPGA as an example of digital realization for the first time in literature. Here, the design processes of FPAA and FPGA based HR neuron model implementations are given in detail.

4.1 FPAA-based implementation of the HR neuron model and coupled HR neurons

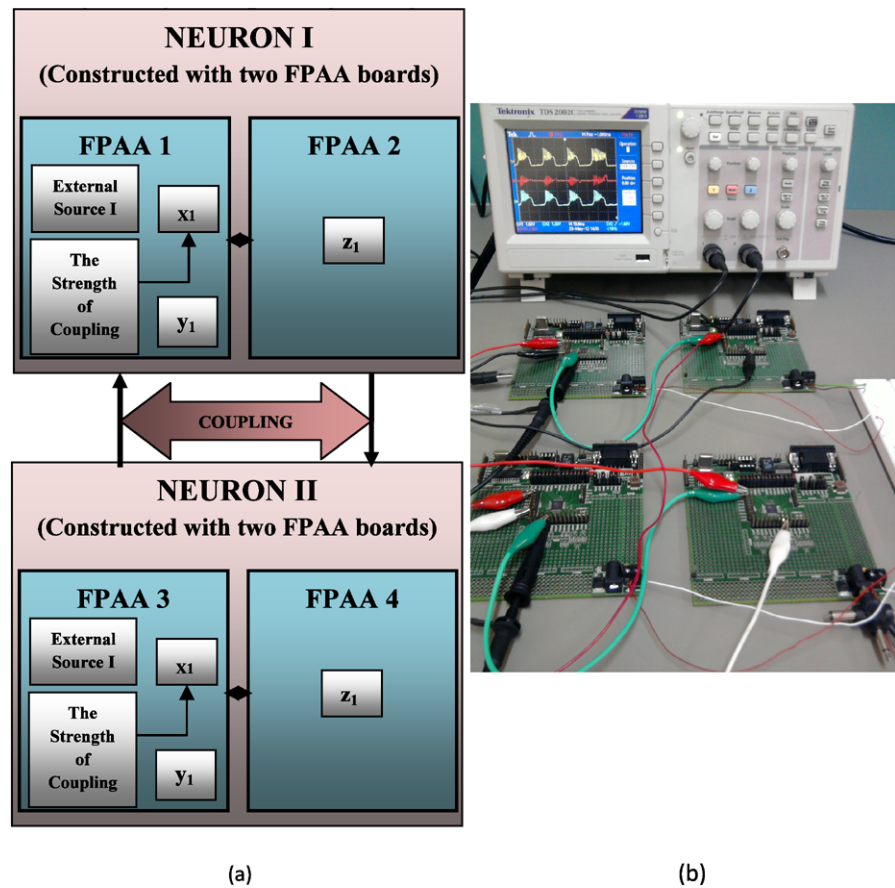
FPAAs are electrically reprogrammable integrated circuits, which contain basic analog building blocks. They are used in implementations of analog and mixed circuits providing high stability, accuracy, and rapid

prototyping techniques. FPAA devices have several CABs (Configurable Analog Blocks), which consist of op-amps, an array of switches, and a capacitor bank and this device use switched-capacitor technology for implementing various analog functions in a CAB. In this study, AN231E04 type FPAA boards were used and each board has four configurable blocks.

Various configurable analog blocks (CAMs-Configurable Analog Modules) are predefined in Anadigm Designer2 software tool. These CAM blocks are used to implement analog functions such as multiplication, addition, filtering, rectification, etc. The comprehensive descriptions about these CAMs are available on this software tool.¹ An FPAA chip has limited capacity and a specific saturation level, (± 2 V). To eliminate the capacity limitation, the FPAA boards can be combined with each other. And, to overcome the saturation problem, the system model must be rescaled according to numerical simulation results and saturation levels in design process. In order to implement

¹Online available: www.anadigm.com.

Fig. 9 (a) A block diagram, (b) Experimental setup, for FPAA based two coupled Hindmarsh–Rose neurons with electrical coupling



the FPAA based HR neuron model and the synchronization of two coupled neurons, their mathematical equations must be modified because of the saturation level of the FPAA. If the numerical simulation results of this neuron model are examined, it is seen that the y -state variable exceeds saturation level. The modified HR neuron definitions are given in Eq. (3).

$$\begin{aligned} \dot{x} &= x - x^3 + bx^2 + I - az + cy \\ \dot{y} &= e - fx^2 \\ \dot{z} &= 0.9976z + (0.016 + 0.02x) \end{aligned} \quad (3)$$

The continuous time modified equations defined by Eq. (3) are the basis for implementing a HR neuron on FPAA. This neuron is designed with FPAA by using predefined SUMFILTER, TRANSFER FUNCTION, MULTIPLIER, and VOLTAGE blocks as shown in Fig. 7. x , y , and z state variables in Eq. (3) are implemented with SUMFILTER blocks. This block consists of a single pole low pass filter and three analog weighted inputs. These inputs can be used both

in addition and subtraction operations. They are applied to a single pole low pass filter part that has a programmable corner frequency. The quadratic and cubic functions of membrane potential and itself, namely $(x - x^3 + bx^2)$ function in x -state variable are embedded on a TRANSFER FUNCTION block. This block produces output voltage with 256 quantization steps according to a function defined by the user. Therefore, a specified output voltage is generated according to the value of the sampled input voltage. Another quadratic function in recovery variable y is realized with the MULTIPLIER block. It is used for performing the multiplication operation. The membrane input current (I) and the other constant parameters are constituted with a DC voltage source called the VOLTAGE block. The remaining addition and subtraction operations are implemented with the SUMDIFF blocks. This block is similar to the SUMFILTER block, but it does not consist of a filter.

To observe the spiking behaviors of the neuron, the FPAA modeling is downloaded to the develop-

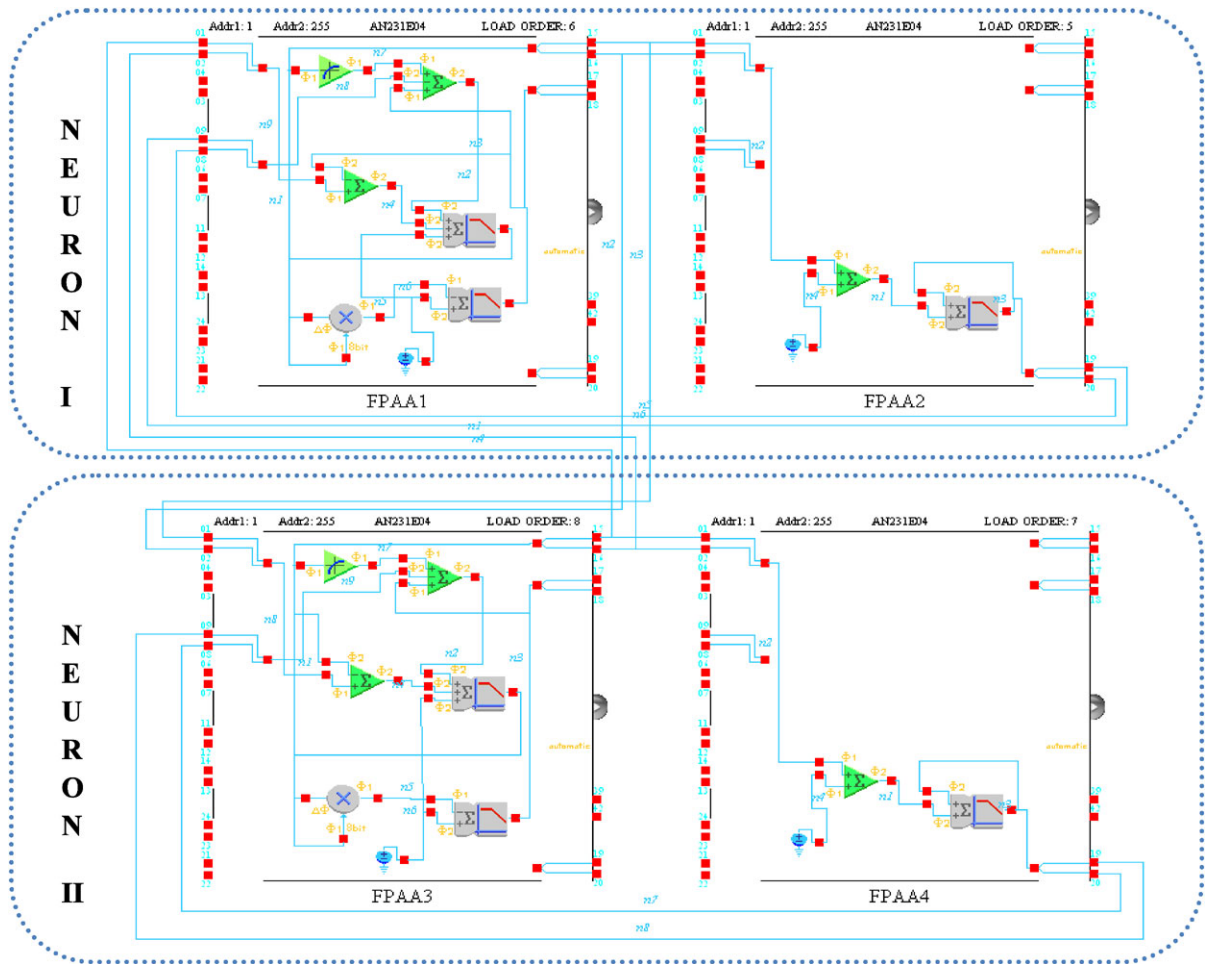


Fig. 10 The FPAA implementation scheme of two HR neurons with electrical coupling

ment board by setting the parameters in Eq. (3) as in follows: $a = 2, b = 2.96, c = 4, f = 3.125, e = 0.125, I = 0.5$. And the experimental results of the FPAA realization were measured via an oscilloscope in time domain and $x-y$ illustration. The experimental result of spiking behavior is shown in Fig. 8a and $x-y$ phase portrait of the HR neuron model is given in Fig. 8b. To obtain the bursting behavior of the FPAA based HR neuron model, only the variable parameters in Eq. (3) are adjusted to the following values: $a = 2, b = 2.6, c = 10, f = 0.055, e = 1, I = 0.8$ on the same FPAA hardware in Fig. 7 in a programmable manner. The bursting behavior of FPAA based HR neuron and $x-y$ phase portrait of this behavior are given in Figs. 8c and 8d.

After the HR neuron is implemented with the FPAA device, two coupled HR neurons are imple-

mented on FPAA devices by using electrical coupling. To realize the FPAA based synchronous two neurons, four FPAA boards are used in this study as in Fig. 9. Here, the neuron structures are constituted separately, Neuron 1 is embedded FPAA1&2 and Neuron 2 is embedded FPAA3&4. The external sources used to change the fire patterns of neurons are realized in FPAA1&3. Electrical coupling operations and the adjustable strength parameter are implemented in FPAA1&3.

The electrical coupling between these neurons is realized only a SUMDIFF block, because $\gamma_e(x_j)$ function consists of a difference expression, $\sigma(x_i) = 1$ is ignored and the values of strength parameter (g_s) are adjusted with the gain of SUMDIFF block. The FPAA circuit scheme of two coupled HR neurons with electrical coupling is presented in Fig. 10.

Fig. 11 The experimental results of the electrical coupled two HR neurons (a) x_1 , x_2 and error signal for $g_s = 0.2$, x : 1 V/div, y : 1 V/div, time/div: 10 ms, (b) x - y phase portrait illustration for $g_s = 0.2$: 1 V/div, y : 1 V/div (c) x_1 , x_2 and error signal for $g_s = 0.8$, x : 1 V/div, y : 1 V/div, time/div: 10 ms, (d) x - y phase portrait illustration for $g_s = 0.8$: 0.5 V/div, y : 0.5 V/div

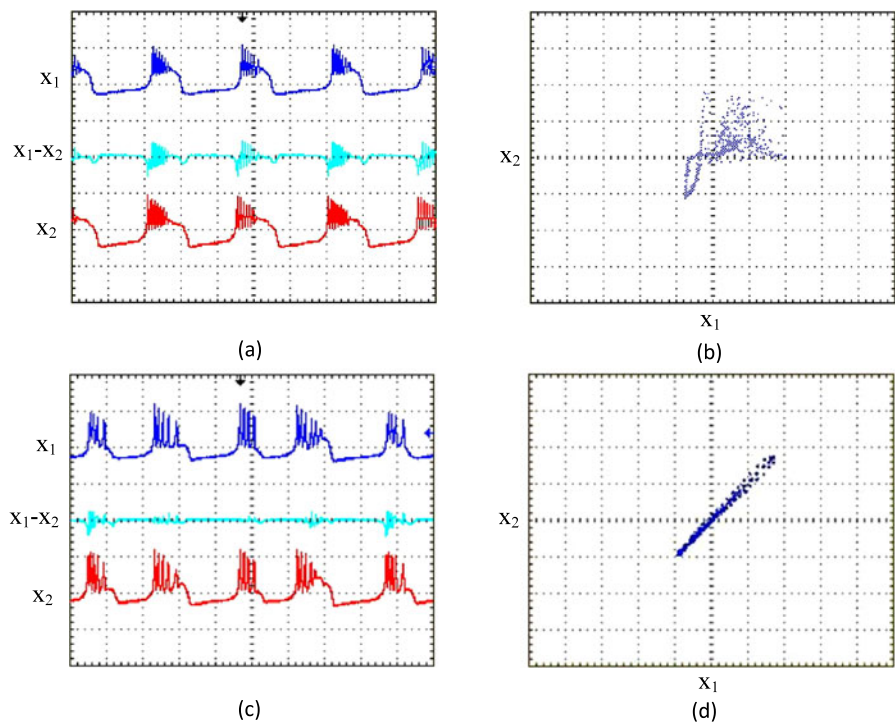


Table 1 The power consumptions and CAB usage capacity of the FPAA based coupled neurons implementation

		Power consumption	CAB1 (used/total capacity)	CAB2 (used/total capacity)	CAB3 (used/total capacity)	CAB4 (used/total capacity)
Neuron I	FPAA 1	128 ± 38 mW	8/8	7/8	5/8	5/8
	FPAA 2	56 ± 17 mW	7/8	0/8	0/8	0/8
Neuron II	FPAA 3	128 ± 38 mW	8/8	7/8	5/8	5/8
	FPAA 4	56 ± 17 mW	7/8	0/8	0/8	0/8

The asynchronous and synchronous situations can be obtained by changing the (g_s) parameter, as mentioned previously. The experimental measurements of the FPAA based realization of two coupled neurons are obtained via an oscilloscope in time domain and their synchronizations can be appeared from the x - y illustrations. These experimental results are given in Figs. 11a, 11b, 11c, and 11d for asynchronous ($g_s = 0.2$) and synchronous ($g_s = 0.8$) situations, respectively.

The systems, which include the nonlinear expressions defined by different mathematical functions, are affected the adjustable parameters within the nonlinear functions. Moreover, these nonlinear functions are difficult to implement with discrete devices. However,

they are embedded on the TRANSFER FUNCTION block in FPAA and the variable parameters in the system are adjusted with FPAA software tool Anadigm Designer2 without any need for extra design effort. For these reasons, quadratic and cubic descriptions in membrane potential and artificial coupling functions are realized with FPAA in a precise way. The results are consistent with numerical simulations. The CAB usage capacity and power consumptions of these implementations are summarized in Table 1.

4.2 FPGA-based implementation of the HR neuron model and coupled HR neurons

FPGAs are reconfigurable integrated circuits, which consists of programmable digital blocks and pro-

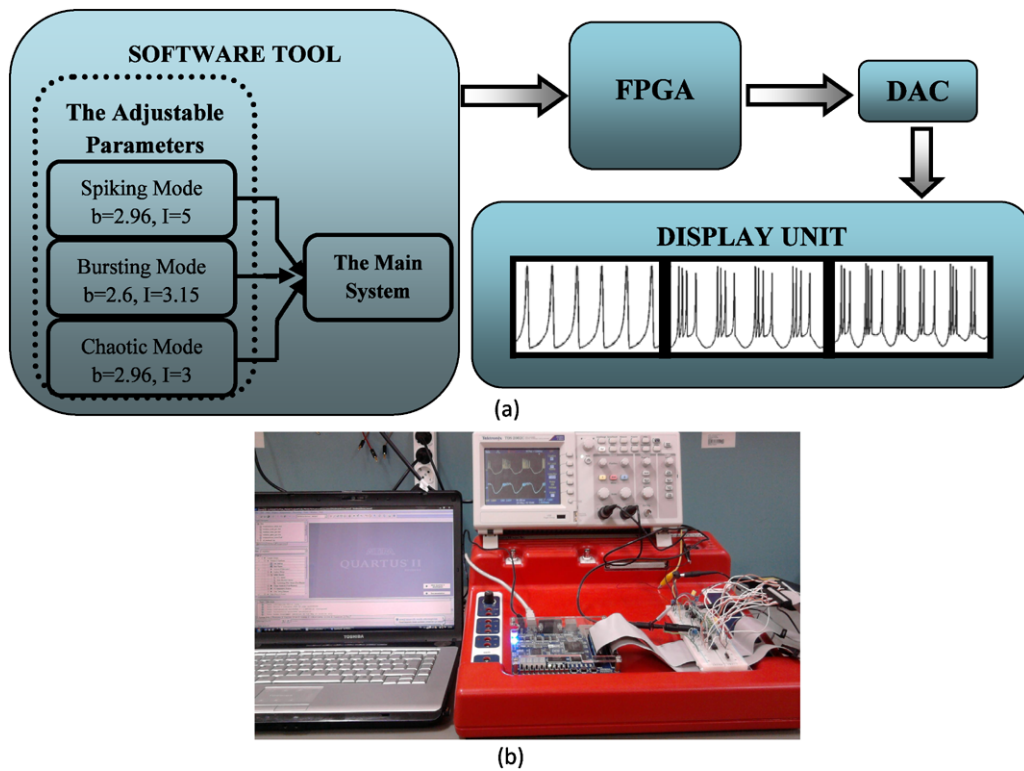


Fig. 12 (a) A block diagram of the implementation of the FPGA based HR Neuron Model, (b) Experimental setup

programmable interconnections between them. Unlike standard digital microprocessors, FPGAs provide massive parallelism in digital implementations. The FPGA device used in these implementations is the Cyclone III IC available on the Altera DE0 educational board. A Cyclone III has 15408 digital blocks called logic elements and 112 9-bit embedded multipliers. Also, an Altera DE0 board has various peripherals like switches and LEDs.²

To implement FPGA based HR neuron model, continuous time equations are discretized using the Euler discretization method by taking discretization constant as 0.01. The other parameter values are the same as that in the numerical simulation. The discrete time equations are as follows:

$$\begin{aligned}
 x_{n+1} &= x_n + \Delta h(y_n - x_n^3 + bx_n^2 + I - z_n) \\
 y_{n+1} &= y_n + \Delta h(1 - 5x_n^2 - y_n) \\
 z_{n+1} &= z_n + \Delta h(\mu(s(x_n - x_{rest}) - z_n))
 \end{aligned}
 \tag{4}$$

²[Online]. Available: www.altera.com.

Derived equations are realized with VHDL (Very high speed integrated circuit Hardware Description Language) on QuartusII software using 32-bit fixed point numerical notation. In the design process, the main system in Eq. (4) is first formed using VHDL. The parameters b and I , which affect the fire patterns, are adjusted via the software tool to observe the different behavior of the neuron with the FPGA. The predefined system model of the neuron is downloaded to this device by changing the variable parameters as similar to FPAA. Since the output of an FPGA device is digital, unlike FPAA, the outputs of the FPGA device are fed to 8-bit DACs. Experimental measurements are monitored through a DAC block. The experimental setup of FPGA based HR neuron is shown in Fig. 12.

Three different behaviors of the neuron model on FPGA are shown in the following figures depending on the adjustable parameters. First of these represents the regular spiking behavior in Fig. 13a. The bursting behavior is given in Fig. 13c and the last one is chaotic dynamic of an FPGA based HR neuron as in Fig. 13e.

Fig. 13 The experimental results of FPGA based HR neuron model (a) Spiking mode, x : 2 V/div, y : 2 V/div, time/div: 50 μ s, (b) x - y phase portrait illustration of the spiking mode, x : 2 V/div, y : 2 V/div, time/div: 50 μ s, (c) Bursting mode x : 2 V/div, y : 2 V/div, time/div: 500 μ s, (d) x - y phase portrait illustration of the bursting mode x : 1 V/div, y : 1 V/div, (e) Chaotic mode x : 2 V/div, y : 2 V/div, time/div: 1 ms, and (f) x - y phase portrait illustration of the chaotic mode x : 1 V/div, y : 1 V/div

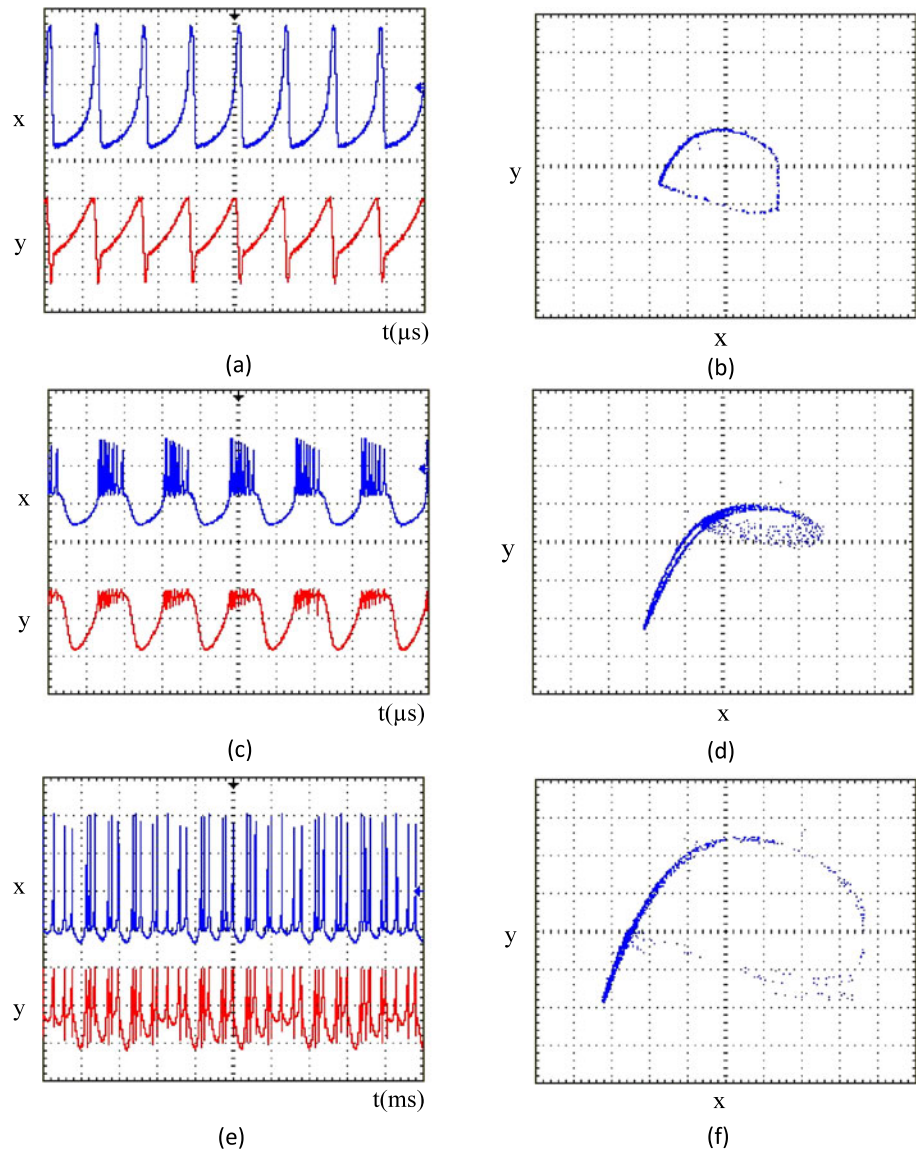


Table 2 The synthesis results and estimated power consumptions of the FPGA based neuron implementation

	Area usage (logic elements)	Embedded multiplier usage	Maximum frequency	Power consumption
I neuron	1156 (%8)	64 (%57)	23.1 MHz	83.02 mW
II neuron	4851 (%31)	104 (%93)	14.19 MHz	83.14 mW

To implement two coupled neurons, which have the same characteristic on the same FPGA device, their initial conditions must be set to different values. Because the clocks on the FPGA are applied to these neurons simultaneously, they already exhibit a

synchronous behavior. When the same initial condition is assigned to the neurons in numerical analyses, the error signal between the neurons is zero as in Fig. 14a. To make these neurons asynchronous, their initial conditions are adjusted to different val-

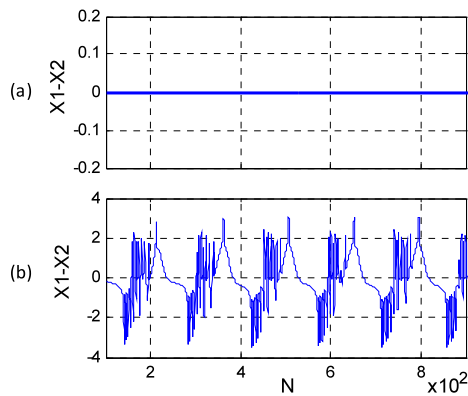


Fig. 14 The numerical simulation results of two HR neurons with the different initial conditions-IC (a) the error signal between the neurons with the same IC, (b) the error signal between the neurons with the different IC

ues and the error signal between them increases as in Fig. 14b.

Similar to numerical simulation, initial conditions of the neurons fulfilled on the same FPGA device are set to different values and the neurons become asynchronous. After two HR neurons are coupled with electrical coupling, the synchronization behavior is adjusted by depending on the values of g_s . For $g_s = -0.2$, $g_s = 0.2$, $g_s = 0.35$, and $g_s = 0.8$, experimental results of FPGA based implementation are given in Fig. 15.

It is noted that when the value of g_s is negative, the system exhibits the antiphase behavior as in Fig. 15a. While the asynchronous behaviors are observed for $g_s \leq 0.52$ as in Figs. 15b–c, the synchronous behavior is shown in Fig. 15d for $g_s > 0.52$. In addition, both the neuron dynamics results and the synchronization results of the FPGA based HR neuron are in agreement with the numerical simulations and FPAA realizations. The synthesis results and estimated power consumption values of FPGA based implementations are given in Table 2.

5 Conclusions

A comprehensive study about the Hindmarsh–Rose neuron model has been presented in this paper. The HR neuron model has been implemented with analog and digital programmable embedded hardware for the first time in literature. In addition, two HR neurons

are coupled with electrical coupling and the synchronization behaviors are investigated in both FPAA and FPGA based experimental setups.

In the FPAA based design process, HR Neuron Model is rescaled because of the saturation level of the device and the continuous time system description is used for FPAA based implementations. A neuron is constructed with two FPAA devices. There is no need for extra process except for the parameter adjustments in the software tool. To observe the synchronization condition of the neurons, four devices are used.

In the FPGA based design process, because of the digital structure of FPGA, the equations of the HR neuron model are discretized. The discrete equations with adjustable parameters are built on FPGA devices by using VHDL. After two neurons, which have two different initial conditions, are constituted on FPGA separately, they are combined with electrical coupling to synchronize on the FPGA.

When the performances of analog and digital HR neuron platforms are compared, it can be seen that both of them reflect the dynamics of a neuron and neural system similar to the numerical analyses. As seen in Tables 1 and 2, while one FPGA board was sufficient for FPGA implementations in terms of capacity usage, multiple FPAA boards were required for FPAA implementations. Also, using multiple boards made FPAA consume more power compared to the power consumption of the FPGA. The outputs of the FPGA were obtained through DACs. Therefore, the performance of the FPGA remained dependent on the bit resolution of the DAC in use. In contrast, there was no requirement for an external device to observe the neural dynamics for FPAA.

The programmability and reconfigurability features of FPAA and FPGA devices provide convenience for observation of fire patterns of the neuron, because the different behaviors of the neuron are predefined in the software tools and the variable parameters are adjusted easily and sensitively. The constructed structures in software tools are downloaded to devices one by one for evaluation. Thus, the experimental tools, which deal with neural system dynamics, are provided to researchers.

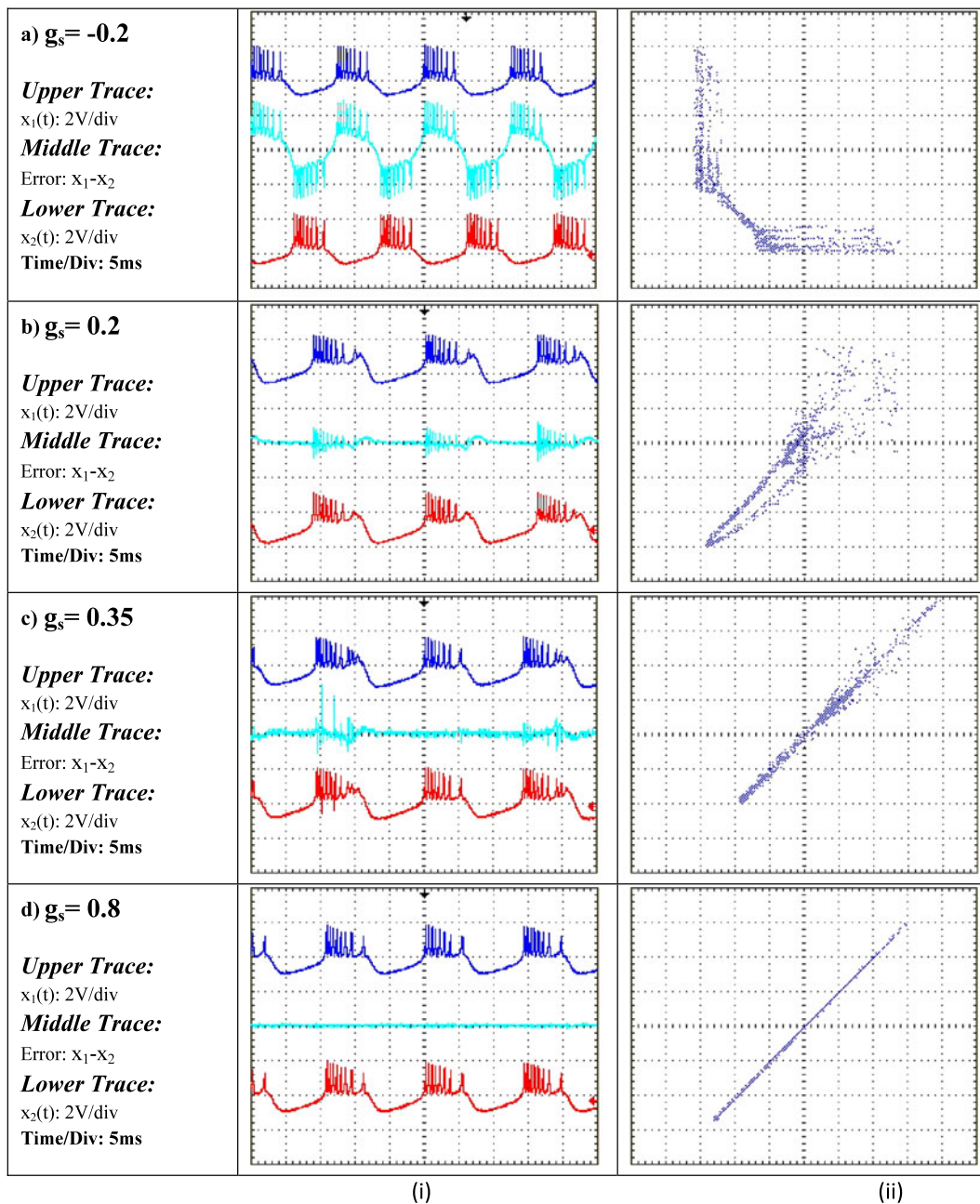


Fig. 15 Experimental results of FPGA based coupled HR neurons for the values of $g_s = \{-0.2, 0.2, 0.35, 0.8\}$. **(i)** The results of neurons states, $x_1(t)$, $x_2(t)$ and the error signal $x_1(t) - x_2(t)$. **(ii)** Phase portraits in $x_1(t) - x_2(t)$ domain

References

1. McCulloch, W.S., Pitts, W.H.: A logical calculus of ideas immanent in nervous activity. *Bull. Math. Biophys.* **5**, 115–133 (1943)
2. Hodgkin, A., Huxley, A.: A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.* **117**, 500–544 (1952)
3. FitzHugh, R.: Mathematical models for excitation and propagation in nerve. In: Schawin, H.P. (ed.) *Biological Engineering*, vol. 1, pp. 1–85. McGraw-Hill, New York (1969)
4. Nagumo, J., Sato, S.: On a response characteristic of mathematical neuron model. *Kybernetik* **10**, 155–164 (1972)
5. Wilson, H.R., Cowan, J.D.: Excitatory and inhibitory interactions in localized populations of model neurons. *Biophys. J.* **12**(1), 1–24 (1972)
6. Morris, C., Lecar, H.: Voltage oscillations in the barnacle giant muscle fiber. *Biophysics J.* **35**, 193–213 (1981)
7. Hindmarsh, J.L., Rose, R.M.: A model of neural bursting using three coupled first order differential equations. *Proc. R. Soc. Lond. B, Biol. Sci.* **221**(1222), 87–102 (1984)
8. Izhikevich, E.M.: Simple model of spiking neurons. *IEEE Transactions on Neural Networks* **14**(6) (2003)
9. Mishra, D., Yadav, A., Ray, S., Kalra, P.K.: Controlling synchronization of modified FitzHugh–Nagumo neurons under external electrical stimulation. *NeuroQuantology* **4**(1), 50–67 (2006)
10. Checco, P., Righero, M., Biey, M., Kocerev, L.: Information processing in networks of coupled Hindmarsh–Rose neurons. In: *International Symposium on Nonlinear Theory and its Applications (NOLTA 2006)*, Bologna-Italy, pp. 671–674 (2006)
11. Checco, P., Biey, M., Righero, M., Kocerev, L.: Synchronization and bifurcations in networks of coupled Hindmarsh–Rose neurons. In: *IEEE International Symposium on Circuits and Systems, ISCAS 2007*, pp. 1541–1544 (2007)
12. Elson, R.C., Selverston, A.I., Huerta, R., Rulkov, N.F., Rabinovich, A.I., Abarbanel, H.D.I.: Synchronous behavior of two coupled biological neurons. *Phys. Rev. Lett.* **81**(25), 5692–5695 (1998)
13. Li, F., Liu, Q., Guo, H., Zhao, Y., Tang, J., Ma, J.: Simulating the electric activity of FitzHugh–Nagumo neuron by using Josephson junction model. *Nonlinear Dyn.* **69**(4), 2169–2179 (2012)
14. Shi, X.R.: Bursting synchronization of Hind–Rose system based on a single controller. *Nonlinear Dyn.* **59**, 95–99 (2010)
15. Li, B., Li, Y., Rong, X.: Gait generation and transitions of quadruped robot based on Wilson–Cowan weakly neural networks. In: *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 19–24 (2010)
16. Lu, J., Kim, Y.B., Ayers, J.: A low power 65 nm CMOS electronic neuron and synapse design for a biomimetic micro-robot. In: *IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 1–4 (2011)
17. Lewis, E.R.: An electronic model of the neuron based on the dynamics of potassium and sodium ion fluxes. In: Reiss, R.F. (ed.) *Neural Theory and Modeling*, pp. 154–189. Stanford University Press, Stanford (1964)
18. Roy, G.: A simple electronic analog of the squid axon membrane: the neurofet. *IEEE Trans. Biomed. Eng.* **19**(1), 60–63 (1972)
19. Merlat, L., Silvestre, N., Merckle, J.: A Hindmarsh and Rose-based electronic burster. In: *Proceedings of Fifth International Conference on Microelectronics for Neural Networks*, pp. 39–44 (1996)
20. Lee, Y.J., Lee, J., Kim, Y.B., Ayers, J., Volkovskii, A., Selverston, A., Abarbanel, H., Rabinovich, M.: Low power real time electronic neuron VLSI design using subthreshold technique. In: *Proceedings of the 2004 International Symposium on Circuits and Systems (ISCAS '04)*, vol. 4, pp. IV 744–747 (2004)
21. Poggi, T., Sciutto, A., Storace, M.: Piecewise linear implementation of nonlinear dynamical systems: from theory to practice. *Electron. Lett.* **45**(19), 966–967 (2009)
22. Charles, G., Gordon, C., Alexander, W.E.: An implementation of a biological neural model using analog-digital integrated circuits. In: *IEEE International Behavioral Modeling and Simulation Workshop (BMAS 2008)*, pp. 78–83 (2008)
23. Gotthans, T., Petrzela, J., Hrubos, Z.: Analysis of Hindmarsh–Rose neuron model and novel circuitry realization. In: *Proceedings of the 18th International Conference Mixed Design of Integrated Circuits and Systems (MIXDES)*, pp. 576–580 (2011)
24. Linaro, D., Poggi, T., Storace, M.: Experimental bifurcation diagram of a circuit-implemented neuron model. *Phys. Lett. A* **374**, 4589–4593 (2010)
25. Steur, E.: On synchronization of electromechanical Hindmarsh–Rose oscillators. Master thesis, Eindhoven University of Technology (2007)
26. Pinto, R.D., Varona, P., Volkovskii, A.R., Szücs, A., Abarbanel, H.D.I., Rabinovich, M.I.: Synchronous behavior of two coupled electronic neurons. *Phys. Rev., E Stat. Phys. Plasmas Fluids Relat. Interdiscip. Topics* **62**, 2644–2656 (2000) (2000)
27. Jacquir, S., Binczak, S., Bilbault, J.M., Kazantsev, V., Nekorkin, V.: Synaptic coupling between two electronic neurons. *Nonlinear Dyn.* **44**, 29–36 (2006)
28. Zhao, J., Kim, Y.B.: Circuit implementation of Fitz–Hugh–Nagumo neuron model using field programmable analog arrays. In: *50th Midwest Symposium on Circuits and Systems (MWSCAS 2007)*, pp. 772–775 (2007)
29. Sekerli, M., Butera, R.J.: An implementation of a simple neuron model in field programmable analog arrays. In: *26th Annual International Conference of the IEEE on Engineering in Medicine and Biology Society (IEMBS '04)*, pp. 4564–4567 (2004)
30. Rice, K.L., Bhuiyan, M.A., Taha, T.M., Vutsinas, C.N., Smith, M.C.: FPGA implementation of Izhikevich spiking neural networks for character recognition. In: *International Conference on Reconfigurable Computing and FPGAs (ReConFig '09)*, pp. 451–456 (2009)
31. Weinstein, R.K., Lee, R.H.: Architectures for high-performance FPGA implementations of neural models. *J. Neural Eng.* **3**, 21–34 (2006)
32. Bizzarri, F., Linaro, D., Storace, M.: PWL approximation of the Hindmarsh–Rose neuron model in view of its circuit implementation. In: *18th European Conference on Circuit Theory and Design (ECCTD 2007)*, pp. 878–881 (2007)

33. González-Miranda, J.M.: Complex bifurcation structures in the Hindmarsh–Rose neuron model. *Int. J. Bifurc. Chaos Appl. Sci. Eng.* **17**(9), 3071–3083 (2007)
34. Storace, M., Lınaro, D., Lange, E.: The Hindmarsh–Rose neuron model: bifurcation analysis and piecewise-linear approximations. *Chaos* **18**(3), 033128 (2008)
35. Innocenti, G., Genesio, R.: On the dynamics of chaotic spiking-bursting transition in the Hindmarsh–Rose neuron. *Chaos* **19**(2), 023124 (2009)
36. Danca, M.F., Wang, Q.: Synthesizing attractors of Hindmarsh–Rose neuronal systems. *Nonlinear Dyn.* **62**(1–2), 437–446 (2010)
37. Kandel, E.R., Schwartz, J.H., Jessell, T.M.: *Principles of Neural Science*, 4th ed. McGraw-Hill, New York (2000) ISBN 0-8385-7701-6
38. Bal, T., Nagy, F., Moulins, M.: The pyloric central pattern generator in Crustacea: a set of conditional neuronal oscillators. *J. Comp. Physiol.* **163**(6), 715–727 (1988)
39. Miller, J.P., Selverston, A.I.: Mechanisms underlying pattern generation in lobster stomatogastric ganglion as determined by selective inactivation of identified neurons. II. Oscillatory properties of pyloric neurons. *J. Neurophysiol.* **48**(6), 1378–1391 (1982)
40. Wu, Y., Xu, J., Jin, W.: Synchronous behaviors of two coupled neurons. In: *Neural Networks (ISNN 2005)*, Lecture Notes in Computer Science Advances, vol. 3496, pp. 121–130 (2005)
41. Shi, Y., Wang, J., Deng, B., Liu, Q.: Chaotic synchronization of coupled Hindmarsh–Rose neurons using adaptive control. In: *2nd International Conference on Biomedical Engineering and Informatics (BMEI '09)*, pp. 1–5 (2009)
42. Che, Y., Zhang, S., Wang, J., Cui, S., Han, C., Deng, B., Wei, X.: Synchronization of inhibitory coupled Hindmarsh–Rose neurons via adaptive sliding mode control. In: *2nd International Conference on Intelligent Control and Information Processing (ICICIP)*, vol. 2, pp. 1134–1139 (2011)
43. Shi, X., Wang, Z.: Adaptive synchronization of time delay Hindmarsh–Rose neuron system via self-feedback. *Nonlinear Dyn.* **69**(4), 2147–2153 (2012)