FPGA Realization of Hodgkin-Huxley Neuronal Model

Farzin Shama, Saeed Haghiri, and Mohammad Amin Imani

***Abstract*— One of the appealing cases of the neuromorphic research area is the implementation of biological neural networks. The current study offers Multiplierless Hodgkin-Huxley Model (MHHM). This modified model may reproduce various spiking behaviors, like the biological HH neurons, with high accuracy. The presented modified model, in comparison to the original HH model, due to its exact similarity to the original model, has more top performances in the case of FPGA saving and more achievable frequency (speed-up). In this approach, the proposed model has a 69 % saving in FPGA resources and also the maximum frequency of 85 MHz that is more than other similar works. In this modification, all spiking behaviors of the original model have been generated with low error calculations. To validate the MHHM neuron, this proposed model has been implemented on digital hardware FPGA. This approach demonstrates that the original HH model and the proposed model have high similarity in terms of higher performance and digital hardware cost reduction.**

***Index Terms*— FPGA, Hodgkin-Huxley, neuron.**

I. Introduction

NEUROMORPHIC engineering is one of the new and attractive branches for researchers who are interested in the brain process information in recent years [1]–[3].

In comparison with conventional processors, the brain is much more complicated and capable. Cognition, comparison, and reasoning capability to learn and develop new concepts are significant characteristics of a human's brain [1]–[5]. Various scientists, such as mathematicians, neuroscience, and biomedical researchers, and many other researchers have handled some significant works in this field. They have covered diverse consequences in their probing, like behavioral and cellular analysis of the neural system [4]–[7]. Among different previous studies, the realization of the Spiking Neural Networks called (SNNs) has considerable necessities in neuro studies. Understanding the model well depends on the hardware implementation of diverse neuron models. The human nervous system is composed of neurons. The data in these cells have been coded, processed, and transferred with electrochemical signals. For better perception of the system function, neuron models might be so helpful.

Some mathematical and biological models for brain neurons are HH model, Morris-Lecar model, Izhikevich model, and so on [4]–[16]. Among all the various neuron models, the HH model is the most valuable study in neuromorphic. The Nobel award in medicine 1963 has been won by A. L. Hodgkin and his co-worker, A. F. Huxley for their significant findings of the ionic mechanisms of stimulation and deterrence in the structure of brain cells or neurons. Thus, the most precise brain cells mathematical model is the HH neuron model as yet. Unfortunately, this model has composed of numerous complicated equations that make it difficult for possible implementations of this model.

FPGA-based neural network implementations have been discussed in many papers [11]–[19]. For instance, presenting an implementation of the HH neuron model has been investigated in Yaghini *et al.* [19]. In their work, the CORDIC algorithm and gradual integration that are computational techniques were applied in the realization of arithmetic circuits. The present research proposes a multiplierless digital realization of the HH [1] neuron model with hyperbolic approximations. Modifications, as a new model, have applied to the original HH model to simplify it besides reducing the implementation costs. The proposed model in this paper is called as Multiplierless Hodgkin-Huxley Model (MHHM). In analogy with the original HH model, the MHHM has higher performances and more achievable frequency (speed-up) due to its perfect similarity with the original model. In this modification, all spiking behaviors of the original model are generated with low error calculations. Furthermore, some studies have been conducted on the original HH model and the proposed MHHM in the rest.

Behaviors of single biological neurons, as well as the dynamic discussions and the synapse interactions of HH neurons, are some major studies in this research. Implementation of Spiking Neural Networks (SNNs) have been reported in many papers [11]–[19]. For example, Gomar and Ahmadi [11] proposes a multiplierless design of the Adex neuron model based on realization of exponential function, Imani *et al.* [12] presents two coupled Wilson neuronal model on an FPGA board, Haghiri *et al.* [13] implements the novel model of Izhikevich with noisy stimulus, Nouri *et al.* [15] explains the realization of FitzHugh–Nagumo (FHN) neuron model based on implementation of exponential function, Soleimani *et al.* [17] proposed an implementation of Izhikevich model in multiplierless form. Hayati *et al.* [18] present

Manuscript received October 21, 2019; revised December 20, 2019 and March 2, 2020; accepted March 9, 2020. Date of publication March 16, 2020; date of current version May 8, 2020. This work was supported by the Kermanshah Branch, Islamic Azad University, Kermanshah, Iran. *(Corresponding author: Farzin Shama.)*

Farzin Shama is with the Department of Electrical Engineering, Kermanshah Branch, Islamic Azad University, Kermanshah, Iran (e-mail: f.shama@aut.ac.ir).

Saeed Haghiri is with the Department of Electrical Engineering, Kermanshah University of Technology, Kermanshah, Iran (e-mail: s.haghiri@kut.ac.ir).

Mohammad Amin Imani is with the Medical Biology Research Center, Kermanshah University of Medical Sciences, Kermanshah, Iran (e-mail: ma.imani90@yahoo.com).

Digital Object Identifier 10.1109/TNSRE.2020.2980475

1534-4320 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on May 11,2020 at 01:08:10 UTC from IEEE Xplore. Restrictions apply.

digital FPGA implementation of the Morris Lecar (ML) neuronal model with low-cost design. Also, different papers have been discussed about the digital implementation of the neuron models in [20]-[22]. Yang *et al.* [20] presented basal ganglia digital realization using linear functions. Heidarpour *et al.* [21] propose digital implementation of STDP learning based on the Izhikevich neuron model using the CORDIC algorithm. Also, Yang *et al.* [22] proposed an efficient digital implementation of a conductance-based globus pallidus neuron using linear functions.

In comparison with the previous works, in this work, a combination of two methods has been applied for a more complex neuron model (HH). The first one is a piece-wise linear approximation. Since the original HH model was made by a large number of high-degree polynomial functions, this piece-wise method is a right and low-cost choice for the realization of the nonlinear terms. The second is the converting multiplications of the original model to logical shifts and adds. Using power-2 based functions leads to achieving equations without any multipliers. Consequently, the proposed MHHM model can be implemented in a hardware platform in low-cost and high-speed level. In all works, the original mathematical model has been converted to a new model with the lowest number of multiplications. The reason for this conversion is that the multiplications need a high calculation time. This fact reduces the frequency of the neuronal system (speed of the proposed system). On the other hand, the existence of multipliers leads to a high-cost operational neuronal system. Thus, to achieve a low-cost and high-speed system, the multiplications have been eliminated. Therefore, the advantage of the multiplierless implementation of neurons has observed. Among the different neuronal models, the HH neuron is a highly accurate model with a high degree of biological levels and is a reliable choice for realizing biological neural networks in a real state. In this paper, at the first step, a multiplierless model of the HH neuronal model as an MHHM neuron is proposed. In this approach, all nonlinear terms of the original HH model were converted to the power of 2 functions, and the other division functions are replaced by piece-wise linear approximations. In our method, all nonlinear terms were reformulated as a power of 2 functions that lead to some digital shifts to right or left and a number of adders. Consequently, we have a simple digital system without any multipliers. Thus, this leads to achieving a high-speed and low-cost digital system. Since the original HH model has a large number of multipliers, in comparison to our proposed model, the frequency of this proposed model will be increased significantly. In the second step, two coupled MHHM neurons are realized on the FPGA board, and the synaptic transmission has been evaluated. At the third step, the behavior of 1000 randomly connected neurons considered for testing a large-scale implementation. In large-scale realization of neural networks, two main issues must be evaluated: the speed of the system and overhead cost. In our proposed model, since all multiplications have been ignored, these two primary factors are solved. In other words, since the overall saving in FPGA is more than the original HH model that means more implemented neurons on an FPGA board in a large-scale system. Besides, the FPGA embedded multiplier number is the limiting factor of the number of neurons, and that's because of the requirement of high-performance multipliers in the original HH model. On the other hand, all FPGA boards have several multipliers that can be used in the digital design of neurons. In this state, it is noticeable that although these embedded multipliers can be applied in our digital systems for implementation of the original HH model, this leads to reduce the maximum number of implemented neurons on an FPGA board, and it is a problem in large-scale implementation. As a consequence, our proposed model is a high-speed and low-cost neuron and capable of applying in the large-scale neural networks. In comparison with [19], since the overall saving in the FPGA area for our proposed model is more than their model, the maximum number of digital neurons in an FPGA will be increased in large-scale implementation.

The composition of this article will be explained as follows. The original HH neuron model is narrated in the next part. In part III, the modified Hodgkin-Huxley neuron model is presented. In the next part, Hopf bifurcation of the HH neuron has been discussed. After that, in section V, the behaviors of two coupled HH models and networking have been explained. This section consists of synapse model evaluations of the HH coupling model and the network behavior discussions. HH and astrocyte interactions considered in part VI. Digital hardware design procedures for the modified HH model are presented in Section VII. This section also includes equation discretization, bit-width definition, hyperbolic function realization, and resource comparison. The results of the implementation are reported in Section VIII. Part IX consists of a conclusion of the paper.

II. HH Model

Hodgkin-Huxley model (HH) is a mathematical conductance-based neuron model, which describes how propagation and initiation of action potentials in neurons happen. This complex neuron model consists of four coupled differential equations and some internal functions [1]. This accurate model is given by

$$\begin{cases} c_m \dfrac{\mathrm{du}}{\mathrm{dt}} = -\mathrm{g_{Na}}\,(\mathrm{u}-\mathrm{v_{Na}})\,m^3 h \\ \quad -\mathrm{g_k}\,(\mathrm{u}-\mathrm{v_k})\,\mathrm{n}^4 - \mathrm{g_L}(\mathrm{u}-\mathrm{v_L}) + \mathrm{I} \\ \dfrac{\mathrm{dm}}{\mathrm{dt}} = \alpha_\mathrm{m}\,(1-\mathrm{m}) - \beta_m \mathrm{m} \\ \dfrac{\mathrm{dn}}{\mathrm{dt}} \alpha_\mathrm{n}\,(1-\mathrm{n}) - \beta_n \mathrm{n} \\ \dfrac{\mathrm{dh}}{\mathrm{dt}} \alpha_\mathrm{h}\,(1-\mathrm{h}) - \beta_h \mathrm{h} \end{cases} \tag{1}$$

where

$$\begin{cases} \alpha_\mathrm{m} = \dfrac{2.5-0.1\mathrm{u}}{\exp(2.5-0.1\mathrm{u})-1} \\ \alpha_\mathrm{n} = \dfrac{0.1-0.01\mathrm{u}}{\exp(1-0.1\mathrm{u})-1} \\ \alpha_\mathrm{h} = 0.07\exp(\dfrac{-u}{20}) \\ \beta_m = 4\exp(\dfrac{-u}{18}) \\ \beta_n = 0.125\exp(\dfrac{-u}{80}) \\ \beta_h = \dfrac{2.5-0.1\mathrm{u}}{\exp(2.5-0.1\mathrm{u})-1} \end{cases} \tag{2}$$

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on May 11,2020 at 01:08:10 UTC from IEEE Xplore. Restrictions apply.

In these equations, cm = 1 and HH model has been described by other parameters that are given below:

- u: Membrane potential;
- cm: Effective capacity per area of the membrane;
- g_{Na}; g_k; g_L: conductivity of Na, K, and the leak channels;
- V_{Na}; V_k; V_L: Reversal potentials of Na, K, and the leak channels;
- I: excitation current per area;
- m; n; h: Coefficients;

III. Modified HH

A modification is suggested to simplify the Hodgkin-Huxley model. In this approach, for achieving a neuronal system with high-speed and lower cost, we will propose a modified model with minimum quadratic terms. In this approach, two papers are presented with their digital implementations [23], [24]. In [23], scalable digital neuromorphic architecture for a large-scale biophysically meaningful neural network with multi-compartment neurons is presented. Its approximation is based on the linear functions for nonlinear terms of the original neuron model. Also, in [24], digital implementations of thalamocortical neuron models and its application in thalamocortical control using FPGA for Parkinson's disease are proposed with piecewise linear modifications. Indeed, in our proposed model in this paper, the piece-wise linear functions are applied for realizing the nonlinear functions of the HH neuron model. Moreover, the power-2 based functions are considered for implementing the multiplications of the original model by Shifts and Adds, only.

From the other side, the presented neuronal model is linear and multiplierless. The main motivations in this way, are the implementation cost and speed-up in analogy with the original.

HH model and by this means, multiplierless HH Model (MHHM) can be presented. The membrane potential equation and other coupling differential equations are given by

$$\text{MHHM}: \begin{cases} c_m \dfrac{\mathrm{du}}{\mathrm{dt}} = -\mathrm{g_{Na}}\,(\mathrm{u}-\mathrm{v_{Na}})\sinh(h)\,\mathrm{f}\,(\mathrm{m}) \\ \quad -\mathrm{g_k}\,(\mathrm{u}-\mathrm{v_k})\,\mathrm{g(n)} - \mathrm{g_L}(\mathrm{u}-\mathrm{v_L}) + \mathrm{I} \\ \dfrac{\mathrm{dm}}{\mathrm{dt}} = \alpha_{\mathrm{m-appx}} sinh\,(1-\mathrm{m}) - \beta_{m-appx} sinh\,(\mathrm{m}) \\ \dfrac{\mathrm{dn}}{\mathrm{dt}} = \alpha_{\mathrm{n-appx}} sinh\,(1-\mathrm{n}) - \beta_{n-appx}\,\sinh\,(n) \\ \dfrac{\mathrm{dh}}{\mathrm{dt}} = \alpha_{\mathrm{h-appx}} sinh\,(1-\mathrm{h}) - \beta_{h-\mathrm{appx}}\,\sinh\,(\mathrm{h}) \end{cases} \tag{3}$$

where

$$f(m) = sinh(m)sinh(m)sinh(m) \tag{4}$$

$$g(n) = sinh(n)sinh(n)sinh(n)sinh(n) \tag{5}$$

In this procedure, the hyperbolic functions turn to binary functions [12], [13]. As depicted in Fig. 1 (a), the hyperbolic function $f(m)$ can match the original function with good accuracy. Therefore, the hyperbolic function can be rewritten as below:

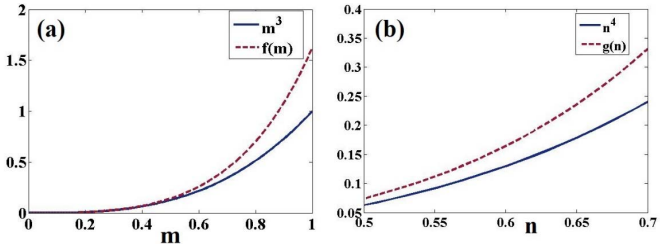

Fig. 1. Matching accuracy. (a) Matching accuracy between m^3 and $f(m)$. (b) Matching accuracy between n^4 and $g(n)$.

$$sinh\,(\mathrm{m}) = \frac{2^m - 2^{-m}}{2} \tag{6}$$

$$sinh\,(\mathrm{n}) = \frac{2^n - 2^{-n}}{2} \tag{7}$$

Accordingly, after simplifications, the hyperbolic function $f(m)$ can be given by

$$f\,(m) = \frac{1}{8}[2^{3m} + 2^{-3m} + 2^m - 2^{-m} + 2^{1-m} - 2^{1+m}] \tag{8}$$

Based on this method, the multiplication between m^3 and $sinh(h)$ will be converted to logical shifts and adds, as demonstrated in Section IV. As depicted in Fig.1 (b), the original function and hyperbolic function $g(n)$ have a good matching precision. In this way, for approximating the function $g(n)$, after simplifications, the hyperbolic function can be given by:

$$g\,(n) = \frac{1}{16}[2^{4n} + 2^{-4n} - 2^{2+2n} - 2^{2-2n} + 6] \tag{9}$$

Based on this method, the multiplication between n^4 and $g_k\ (u\text{-}v_k)$ will be converted to logical shifts and adds, as demonstrated in Section IV.

On the other hand, for implementing the α_i and β_i functions (internal functions), we will use the piece-wise linear functions.

$$\alpha_{\mathrm{m-appx}} = \begin{cases} (\alpha m_1)\,u + (\alpha k_1)\,; & 0 < u < u_1 \\ (\alpha m_2)\,u + (\alpha k_2)\,; & u_1 < u < u_2 \\ (\alpha m_3)\,u + (\alpha k_3)\,; & else \end{cases} \tag{10}$$

$$\alpha_{\mathrm{n-appx}} = (\alpha n_1)\,u + (\alpha p_1) \tag{11}$$

$$\alpha_{\mathrm{h-appx}} = \begin{cases} (\alpha h_1)\,u + (\alpha q_1)\,; & 0 < u < u_1 \\ (\alpha h_2)\,u + (\alpha q_2)\,; & u_1 < u < u_2 \\ (\alpha h_3)\,u + (\alpha q_3)\,; & else \end{cases} \tag{12}$$

$$\beta_{\mathrm{m-appx}} = \begin{cases} (\beta m_1)\,u + (\beta k_1)\,; & 0 < u < u_1 \\ (\beta m_2)\,u + (\beta k_2)\,; & u_1 < u < u_2 \\ (\beta m_3)\,u + (\beta k_3)\,; & else \end{cases} \tag{13}$$

$$\beta_{\mathrm{m-appx}} = (\beta n_1)\,u + (\beta p_1)\,; \tag{14}$$

$$\beta_{\mathrm{m-appx}} = \begin{cases} (\beta h_1)\,u + (\beta q_1)\,; & 0 < u < u_1 \\ (\beta h_2)\,u + (\beta q_2)\,; & u_1 < u < u_2 \\ (\beta h_3)\,u + (\beta q_3)\,; & else \end{cases} \tag{15}$$

As depicted in Fig. 2, the α_{i-appx} and β_{i-appx} functions have good matching accuracy with the original functions. Also, the approximation coefficients are given by Table II. In this approximation, $u_1 = 20$, $u_2 = 40$, $u_3 = 60$, and $u_4 = 80$.

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on May 11,2020 at 01:08:10 UTC from IEEE Xplore. Restrictions apply.

TABLE I
DIFFERENT NEURON PARAMETERS FOR HH MODEL

$V_{Na} = 115$	$g_{Na} = 120$
$V_K = -12$	$g_K = 36$
$V_L = 10.60$	$g_L = 0.30$

TABLE II
DIFFERENT NEURON PARAMETERS FOR HH MODEL

$\alpha_{m1} = 0.0350$	$\alpha_{m2} = 0.0780$	$\alpha_{m3} = 0.0970$	$\alpha_{k1} = 0.2200$	$\alpha_{k2} = -1.070$	$\alpha_{k3} = -2.200$
$\alpha_{h1} = -0.0017$	$\alpha_{h2} = -0.0004$	$\alpha_{h3} = -0.0001$	$\alpha_{q1} = 0.0620$	$\alpha_{q2} = 0.0260$	$\alpha_{q3} = 0.0098$
$\alpha_{n1} = 0.0088$	$\alpha_{p1} = -0.0110$	$\beta_{m1} = -0.1300$	$\beta_{m2} = -0.0430$	$\beta_{m3} = -0.0140$	$\beta_{m4} = -0.0047$
$\beta_{k1} = 3.700$	$\beta_{k2} = -2.100$	$\beta_{k3} = 0.9700$	$\beta_{k4} = 0.4100$	$\beta_{n1} = 0.0009$	$\beta_{p1} = 0.1200$
$\beta_{h1} = 0.0110$	$\beta_{h2} = 0.0220$	$\beta_{h3} = 0.0058$	$\beta_{q1} = 0.0240$	$\beta_{q2} = -0.1500$	$\beta_{q3} = 0.5800$

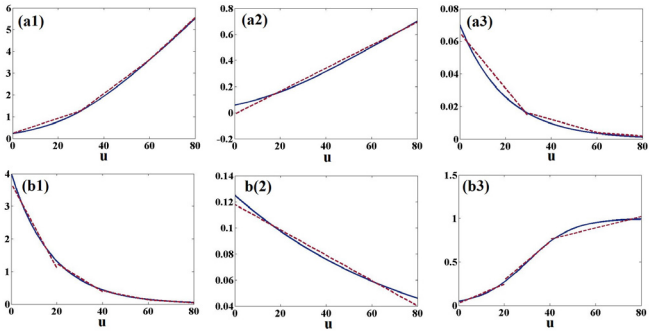

Fig. 2. Matching accuracy. (a1)-(a3) Matching accuracy between α_i and α_i -appx functions. (b1)-(b3) Matching accuracy between β_i and β_i-appx functions.

For validating the proposed MHHM system, errors are obtained in different states based on the parameters of the mathematic model of the neuron for the original and the suggested. In the present case, two main procedures can be used: 1. In terms of resemblances of the output signals, the Root Mean Square Error (RMSE), which has been considered as a comparison of the HH model measurement and [12], [15]. 2. The second procedure for comparing predictions with their eventual outcomes is Mean Absolute Error (MAE), which describes how much foreseen values and observed values are far from [13]. These two solutions are shown below:

$$\mathrm{RMSE}(u_{MHHM}, u_{HH}) = \sqrt{\frac{\sum_{i=1}^{n} (u_{MHHM} - u_{HH})^2}{n}} \quad (16)$$

$$\mathrm{MAE} = \frac{1}{n} \sum_{i=1}^{n} |u_{MHHM} - u_{HH}| \quad (17)$$

Consequently, these methods can calculate errors for different values of stimulus $dt = 10ms$. The acceptable accuracy of the MHHM model is obvious from Table III. Also, different spiking patterns which are generated by different levels of stimulation depicted in Fig. 3. As can be shown in this figure, the proposed MHHM reproduces the original HH model patterns in high precision.

IV. HOPF BIFURCATION OF HH MODEL

The interplays of rest and spike states have the importance of evaluating the dynamic behaviors of the HH neural model [27]-[29]. In this approach, we should consider a two-variable dynamical system. Thus, the voltage variable (u) is considered as a base variable, and other variables (m, n, h) are evaluated with the voltage values. On the other state, the stability process

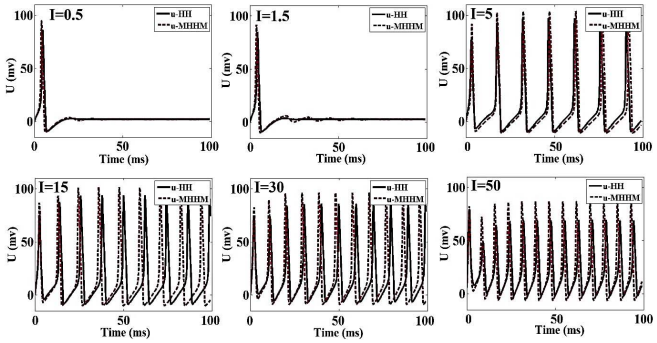

Fig. 3. Different spiking patterns for the proposed MHHM (u-MHHM) and original HH (u-HH) models based on different stimulus currents.

TABLE III
MAE COMPUTATIONS AND RMSE FOR VARIOUS STIMULUS CURRENTS

Stimulus Current	RMSE	MAE
I=0.5	0.02	0.01
I=1.5	0.03	0.02
I=5	0.36	0.06
I=15	0.41	0.03
I=30	0.41	0.05
I=50	0.41	0.03
Mean Error	**0.27**	**0.03**

TABLE IV
EQUILIBRIUM POINTS FOR THE HOPF BIFURCATION DIAGRAM

Stimulus Current(μA)	Fixed point type	Fixed point value
0	Spiral Sink	(-0.097, 0.316)
2	Spiral Sink	(1.45, 0.34)
6	Spiral Sink	(3.82, 0.38)
7	Spiral Source	(4.3, 0.385)
50	nodal Source	(19.46, 0.612)
100	nodal Source	(25.52, 0.684)
300	Spiral Source	(32.165, 0.748)
400	Spiral Sink	(34.141, 0.764)

investigated based on the different levels of input stimulus and the effects of currents on the membrane potentials. The nullclines are given as

$$\begin{cases} \dfrac{du}{dt} = 0 \rightarrow -g_{Na}\,(u - v_{Na})\,m^3 h - g_k\,(u - v_k)\,n^4 \\ \quad -g_L(u - v_L) + I \\ \dfrac{dm}{dt} = 0 \rightarrow \alpha_m - m\,(\alpha_m + \beta_m) = 0 \\ \dfrac{dn}{dt} = 0 \rightarrow \alpha_n - n\,(\alpha_n + \beta_n) = 0 \\ \dfrac{dh}{dt} = 0 \rightarrow \alpha_h - h\,(\alpha_h + \beta_h) = 0 \end{cases} \quad (18)$$

The interplays between two nullclines have a major significance in explaining bifurcation or the switching from the rest state to spiking [27], [29]. For describing the u and m variables coupling, these points are given by

$$\begin{cases} \dfrac{du}{dt} = \dfrac{1}{c_m}[-g_{Na}\,(u - u_{Na})\,m^3 h - g_k\,(u - u_k)\,n^4 \\ \quad -g_L(u - v_L) + I] \\ \dfrac{dm}{dt} = \alpha_m - m\,(\alpha_m + \beta_m) \end{cases} \quad (19)$$

Jacobian matrix and eigenvalues are needed for bifurcation analysis of the equilibrium points [29]. This matrix is shown

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on May 11,2020 at 01:08:10 UTC from IEEE Xplore. Restrictions apply.

TABLE V
SPECIFIC VALUES OF THE SYNAPSE PARAMETERS

$\tau_s = 10$		$S_s = 1$	$d_s = 3$	$h_s = -70$	$k_s = 10$	$z_0 = 0$

below

$$J(u,m) = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \tag{20}$$

where

$$\begin{cases} A = -\dfrac{1}{c_m}\left[\mathrm{g_{Na}} m^3 h + \mathrm{g_k n^4} + \mathrm{g_L}\right] \\ B = -\dfrac{1}{c_m}\left[3\mathrm{g_{Na}} m^2 h\,(\mathrm{u}-\mathrm{u_{Na}})\right] \\ C = \dfrac{-0.1\left[\exp(2.5-0.1\mathrm{u})-1\right]}{\left[\exp(2.5-0.1\mathrm{u})-1\right]^2}(1-\mathrm{m}) \\ \quad + \dfrac{0.1\left[\exp(2.5-0.1\mathrm{u})\right][2.5-0.1\mathrm{u}]}{\left[\exp(2.5-0.1\mathrm{u})-1\right]^2}(1-\mathrm{m}) \\ \quad + \dfrac{4}{m}\left[\exp\left(\dfrac{-u}{18}\right)\right] \\ D = \dfrac{-2.5+0.1\mathrm{u}}{\exp(2.5-0.1\mathrm{u})-1} - 4\exp\left(\dfrac{-u}{20}\right) \end{cases} \tag{21}$$

Here the Jacobian matrix for interaction of u and n is given by:

$$\begin{cases} A = -\dfrac{1}{c_m}\left[\mathrm{g_{Na}} m^3 h + \mathrm{g_k n^4} + \mathrm{g_L}\right] \\ B = -\dfrac{1}{c_m}\left[4\mathrm{g}_k\,(\mathrm{u}-\mathrm{u_k})\,n^3\right] \\ C = \dfrac{-0.01\left[\exp(1-0.1\mathrm{u})-1\right]}{\left[\exp(1-0.1\mathrm{u})-1\right]^2}(1-\mathrm{n}) \\ \quad + \dfrac{0.1\left[\exp(1-0.1\mathrm{u})\right][1-0.1\mathrm{u}]}{\left[\exp(1-0.1\mathrm{u})-1\right]^2}(1-\mathrm{n}) \\ \quad + \dfrac{0.125n}{80}\left[\exp\left(\dfrac{-u}{80}\right)\right] \\ D = \dfrac{-1+0.1\mathrm{u}}{\exp(1-0.1\mathrm{u})-1} - 0.125\exp\left(\dfrac{-u}{80}\right) \end{cases} \tag{22}$$

And the Jacobian matrix for interaction of u and h is given by:

$$\begin{cases} A = -\dfrac{1}{c_m}\left[\mathrm{g_{Na}} m^3 h + \mathrm{g_k n^4} + \mathrm{g_L}\right] \\ B = -\dfrac{1}{c_m}\left[\mathrm{g_{Na}}\left(\mathrm{u}-\mathrm{u_{Na}} m^3\right)\right] \\ C = -0.07\left[\exp\left(\dfrac{-u}{20}\right)\right] \\ \quad (1-\mathrm{h}) - \mathrm{h}\dfrac{0.1\left[\exp(3-0.1\mathrm{u})+1\right]}{\left[\exp(3-0.1\mathrm{u})+1\right]^2} \\ D = -0.07\left[\exp\left(\dfrac{-u}{20}\right)\right] - \dfrac{1}{\exp(3-0.1\mathrm{u})+1} \end{cases} \tag{23}$$

In this case, the stability of the fixed point is determined by the trace of *J(u,m), J(u,n)*, and *J(u,h)*: the fixed points will be stable if A+D is minus and unstable if $A + D > 0$. If both of the eigenvalues of this matrix have minus real parts, the fixed point is stable. However, if at the minimum of the eigenvalues has a real positive part, it is unstable. Bifurcation theory has been explained with how solutions change as parameters in a model are altered. By applying the bifurcation theory, the types of transitions occurring as the parameters have changed can be classified. For which values of I, fixed point loses its stability and fluctuations appear. In particular, we can predict which amount of I the fixed point loses its stability and oscillations emerge (as can be seen in the next step).

HH neuron model, the Hopf bifurcation, will be observed. The Hopf bifurcation is a mechanism that can be used to switch from a stable fixed point to the oscillation. So, the stable point gets unstable, initially. In this state, the bifurcation chart might be characterized by changes of I, in the HH model. In this case, increments of the input stimulus obtain two fixed points for the current, which a Hopf bifurcation takes place at $I = I_1$, $I = I_2$. It is expected that a fixed point to be stable when all the eigenvalues have a minus real part, and it is unstable if at least one of the eigenvalues has a positive real part. With increments of the current excitement intensity, the nullcline goes upward, and the intersection gently goes to the right. The intersection stays stable so long as excitement strength reaches a critical value. As depicted in Fig. 4, the Hopf bifurcation for the interaction of u and n can occur by varying the stimulus current, I. As can be shown in this figure, the fixed point will be stable for $I < I_1 = 7\ \mu$A or $I > I_2 = 380\ \mu$A. Also, it is unstable for other regions. Therefore, a Hopf bifurcation occurs at $I_1$and I_2. Also, in Table I, we can see that for $I < I_1$, $I > I_2$, the fixed points are stable, and for $I_1 < I < I_2$, there are unstable fixed points. The equilibrium points for the Hopf bifurcation are presented in Table IV.

V. BEHAVIORS OF TWO COUPLED HH MODEL AND NETWORKING

A. Synapse Model

A synapse model has been used for describing the interaction between two coupled HH neurons. In this model, a presynaptic HH neuron has joined to a postsynaptic one. The model is defined as below:

$$\begin{cases} \tau_s \dfrac{dz}{dt} = [1 + \tanh(\mathrm{S_s}\,(u_1 - h_s))]\,(1-z) - \dfrac{z}{d_s} \\ \mathrm{I_{synapse}} = \mathrm{k_s}(\mathrm{z}-\mathrm{z_0}) \end{cases} \tag{24}$$

Here, z is the synaptic activation value. The other coefficients are given by

- τ_s: Time delay (s);
- S_s: The Variable for Activation and Relaxation of z
- d_s: The Value for Relaxation of z
- h_s: Threshold Value for Activation of z
- k_s: Conductivity
- z_0: Reference Level of z

In the present equation, Transmission of the signal between the two neurons becomes complete when the presynaptic neuron (u_1) takes its threshold value (h_s). In the present case, synaptic current, $I_{synapse}$ triggers the postsynaptic neuron.

Table V demonstrates the synapse parameters. In the present paper, the proposed synapse model described in [30] has been used.

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on May 11,2020 at 01:08:10 UTC from IEEE Xplore. Restrictions apply.

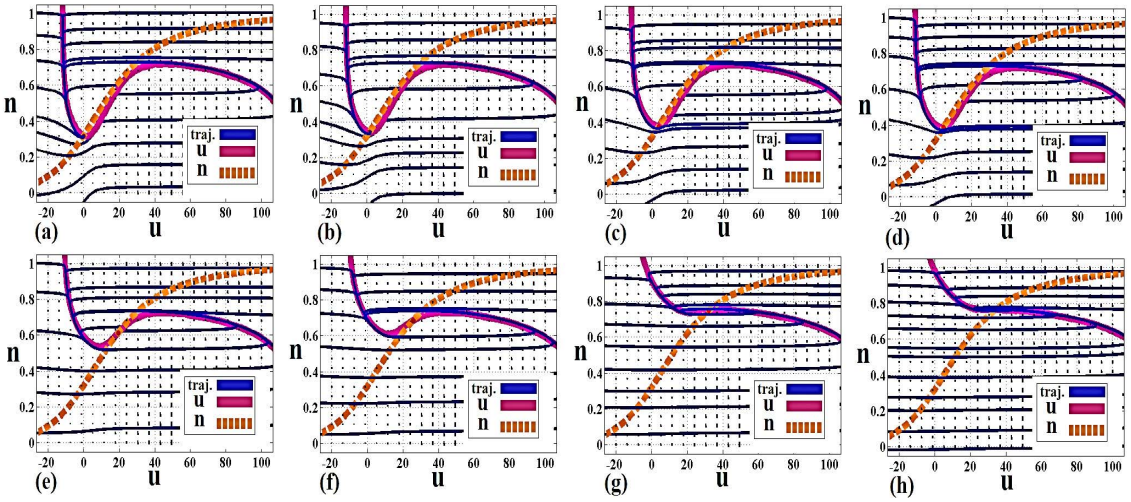

Fig. 4. The Hopf bifurcation for the HH neuron model. (a)-(c) Stable region by increasing the stimulus current from $I = 0\mu$A to $I = 6\mu$A. (d)-(g) Unstable region in the Hopf bifurcation from $I = 7\mu$A to $I = 380\mu$A. (h) Stable region in the Hopf bifurcation after $I = 380\mu$A.

B. HH Coupling Model

The neurons and synchronization of them have an important duty in neural network defects. With systematic vision, synchronization factors can be adjusted with a suitable choice of the synaptic factors. The following section has evaluated the behavior of the coupled neurons. Two coupled neurons are incorporated by a terminal for this purpose. Each presynaptic neuron is triggered with various excitement currents, and the transmission path will be created as the spiking patterns that earn their threshold amount. The coupled HH and MHHM models are defined as below:

$$\text{HH}: \begin{cases} c_m \dfrac{\mathrm{du}\left(1/2\right)}{\mathrm{dt}} = -\mathrm{g_{Na}}\left(\mathrm{u}\left(1/2\right) - \mathrm{v_{Na}}\right)\mathrm{m}^3\left(1/2\right)h\left(1/2\right) \\ \quad -\mathrm{g_k}\left(\mathrm{u}\left(1/2\right) - \mathrm{v_k}\right)\mathrm{n}^4\left(1/2\right) \\ \quad -\mathrm{g_L}\left(\mathrm{u}\left(1/2\right) - \mathrm{v_L}\right) + \mathrm{I} \\ \dfrac{\mathrm{dm}\left(1/2\right)}{\mathrm{dt}} = \alpha_\mathrm{m}\left(1-\mathrm{m}\left(1/2\right)\right) - \beta_m h\left(\mathrm{m}\left(1/2\right)\right) \\ \dfrac{\mathrm{dn}\left(1/2\right)}{\mathrm{dt}} = \alpha_\mathrm{n}\left(1-\mathrm{n}\left(1/2\right)\right) - \beta_n \mathrm{n}\left(1/2\right) \\ \dfrac{\mathrm{dh}\left(1/2\right)}{\mathrm{dt}} = \alpha_\mathrm{h}\left(1-\mathrm{h}\left(1/2\right)\right) - \beta_h \mathrm{h}\left(1/2\right) \\ \tau_s \frac{dz}{dt} = \left[1 + \tanh\left(\mathrm{S_s}\left(u_1 - h_s\right)\right)\right](1-z) - \frac{z}{d_s} \\ \mathrm{I_{synapse}} = \mathrm{k_s}\left(\mathrm{z}-\mathrm{z_0}\right) \end{cases} \tag{25}$$

Different patterns have obtained by these formulas. Various behaviors of two coupled neurons for the HH and MHHM models have been described in Fig. 5. As illustrated, neurons might have various states: coupled, uncoupled, and

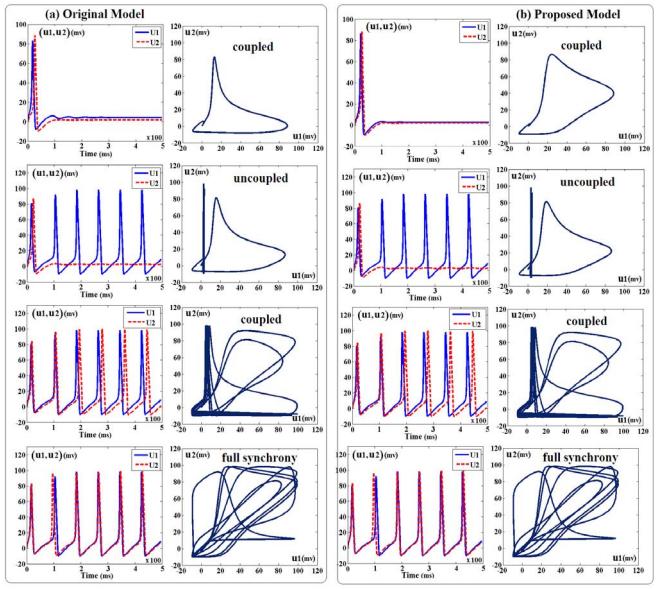

Fig. 5. The coupling behaviors of the original and proposed models. (a) Different synchronization types for the original HH model based on the different levels of stimulus current and the effect of the synapse. (b) Different synchronization types for the proposed MHHM model.

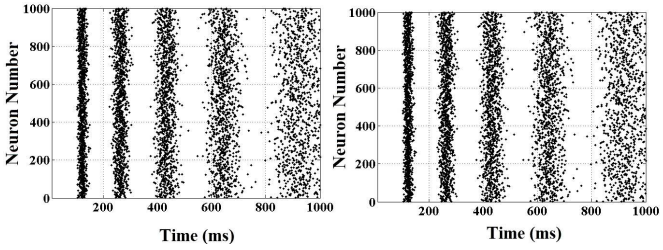

Fig. 6. Raster plot demonstrating the activity of 1000 neurons for tonic neurons in the HH model (a) the MHHM model (b). Random connection of neurons.

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on May 11,2020 at 01:08:10 UTC from IEEE Xplore. Restrictions apply.

full synchrony.

$$\text{MHHM}: \begin{cases} c_m \dfrac{\mathrm{du}\left(1/2\right)}{\mathrm{dt}} = -\mathrm{g_{Na}}\left(\mathrm{u}\left(1/2\right) - \mathrm{v_{Na}}\right)\mathrm{f}\,(\mathrm{m})^{*}\left(1/2\right) \\ \sinh\left(h\left(1/2\right)\right) - \mathrm{g_k}\left(\mathrm{u}\left(1/2\right) - \mathrm{v_k}\right)\mathrm{g(n)}\left(1/2\right) \\ -\mathrm{g_L}(\mathrm{u}\left(1/2\right) - \mathrm{v_L}) + \mathrm{I} \\ \dfrac{\mathrm{dm}\left(1/2\right)}{\mathrm{dt}} = \alpha_{\mathrm{m-appx}} sinh\left(1 - \mathrm{m}\left(1/2\right)\right) \\ -\beta_{m-\mathrm{appx}} \sinh\left(\mathrm{m}\left(1/2\right)\right) \\ \dfrac{\mathrm{dn}\left(1/2\right)}{\mathrm{dt}} = \alpha_{\mathrm{n-appx}} sinh\left(1 - \mathrm{n}\left(1/2\right)\right) \\ -\beta_{n-\mathrm{appx}} \sinh(\mathrm{n}\left(1/2\right)) \\ \dfrac{\mathrm{dh}\left(1/2\right)}{\mathrm{dt}} = \alpha_{\mathrm{h-appx}} sinh\left(1 - \mathrm{h}\left(1/2\right)\right) \\ -\beta_{h-\mathrm{appx}} \sinh(\mathrm{h}\left(1/2\right) \\ \mathrm{I_{synapse}} = \mathrm{k_s} or \mathrm{I_{synapse}} = 0 \end{cases} \tag{26}$$

C. Network Behavior

Neuron population network behavior is also very substantial in a system. A network of 1000 randomly connected HH and MHHM neurons is simulated to comprehend the model on a network scale. Fig. 6 represents the raster plots of the simulations. The MHHM model and the network behaviors of the original HH model have high similarity in structure.

The relative error is computed for evaluating the differences between MHHM models in network behavior and HH. This value is calculated for each spike fired, and the average of these values achieves the MRE error as below:

$$\mathrm{MRE\,(MHHM)\,\%} = \frac{\sum_{i=1}^{N} \left|\Delta t_{MHHM_i}\right|}{|t_{si}|} * 100 \tag{27}$$

That Δt is a time difference between the i_{th} spike in the MHHM model and the HH model, while N is the number of samples. The MRE is less than *1.38* for different levels of stimulus.

VI. HH and Astrocyte Interaction

The Postnov model [30], [31] describes the interactions between two coupled neurons with a simplified and generalized form. In the present model, the activation pathways of astrocytes have activated, and oscillatory behavior has considered mutually as the neurons start to fire, returning of calcium oscillations affects the postsynaptic neuron and the synaptic terminal. This event might adjust synaptic transmission as well as the postsynaptic neuron behavior. The formulas of this model are as below:

$$\tau_c \frac{dc}{dt} = -\mathrm{c} - \mathrm{c_4 f(c, ce)} + [\mathrm{r} + \alpha(\mathrm{m_2} + \mathrm{n_2} + \mathrm{h_2}) + \beta \mathrm{S_m}] \tag{28}$$

$$\varepsilon_c \tau_c \frac{dc_e}{dt} = \mathrm{f(c, ce)} \tag{29}$$

$$\mathrm{f\,(c, ce)} = \mathrm{c_1}\frac{c^2}{1+c^2} - \left(\frac{c_e^2}{1+c_e^2}\right)\left(\frac{c^4}{c^4 + c_2^4}\right) - c_3 \mathrm{c_e} \tag{30}$$

$$\tau s_m \frac{ds_m}{dt} = \left[1 + \tanh\left(\mathrm{S}_{s_m}\left(z - h_{s_m}\right)\right)\right](1 - s_m) - \frac{s_m}{d_{s_m}} \tag{31}$$

$$\tau G_m \frac{dG_m}{dt} = \left[1 + \tanh\left(\mathrm{S}_{G_m}\left(z - h_{G_m}\right)\right)\right](1 - G_m) - \frac{G_m}{d_{G_m}} \tag{32}$$

In these formulas, describing calcium concentrations has been shown by c and c_e symbolized the concentration of calcium in the ER or the endoplasmic reticulum. The term $[r + \alpha(m_2 + n_2 + h_2) + \beta S_m]$ represents the calcium invasion from the external space. Furthermore, the interplays of the cytoplasmic calcium (c) and internal calcium (c_e) are depicted by two-variable function, $f(c;c_e)$. The threshold value for the Sm production, which is operated with synaptic activity (z), Distinguishing of the activated and inactivated states of the values u1; z and c are done by threshold parameters h_s; h_{Sm}, and h_{Gm}, respectively. Astrocyte can have an important duty in the adjustment of synaptic transmission of the two neurons. In this manner, the activity of the postsynaptic neurons can be synchronized, if the astrocyte regulation factors are not adjusted. On the other side, the spiking activity of coupled neurons will be regulated with the appropriate election of the astrocyte factors. Fig. 7 shows the astrocyte effects on regulating the patterns of spikes. Thus, for evaluating the traces of astrocyte in adjusting the activity of the neurons, two coupled MHHM neurons with synaptic connecting and astrocyte interacting (proposed astrocyte model in [31]) are considered. As can be seen in this figure, the neuronal activity will be regulated by the appropriate selection of the astrocyte coefficients.

VII. Digital Hardware Design

In this part, a multiplierless digital design for implementing the MHHM neuronal system in the hardware stage has been presented. Implementation of this hardware comprises of logic Add and Sub as well as shifts, which are evident in Fig. 8. Also, for implementing the hyperbolic functions, the Exponential Unit (Exp. Unit) has been used.

A. Equation Discretization

The differential equations must be discretized for digital implementation by the time-step of dt = 1/128. For reductions in complexity, the Euler method is used. Using this method leads to low-error system implementation.

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on May 11,2020 at 01:08:10 UTC from IEEE Xplore. Restrictions apply.

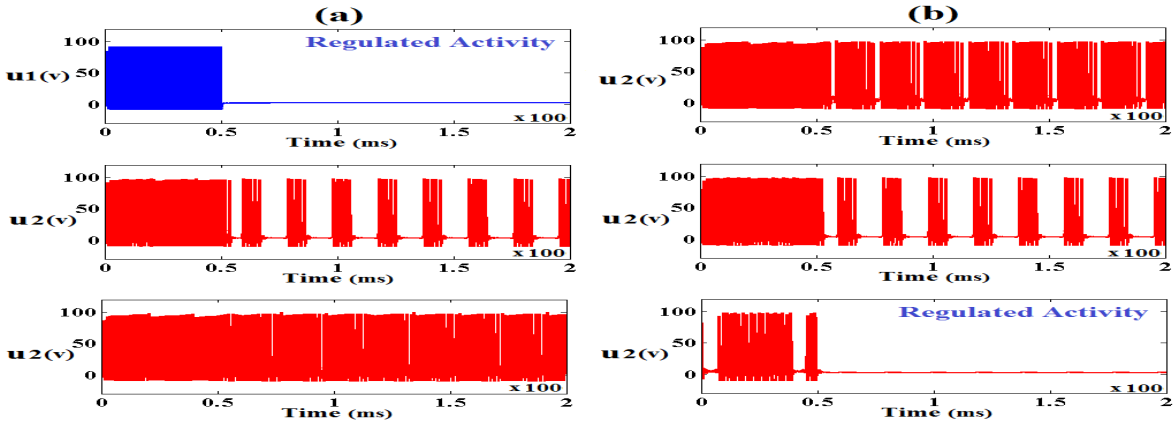

Fig. 7. Astrocyte interaction and simulation outcomes for the MHHM postsynaptic neuron activity and the presynaptic represented in Column (a). In this state, increasing γ and presume that $\delta = 0$ will increase the postsynaptic MHHM neuron excitation current, and the time interval of spiking activity is independent of the presynaptic neuron. Therefore, long-term potentiation of the postsynaptic neuron has been led by this. Effect of astrocyte on the synaptic transmission demonstrated in column (b). In this state, by increasing, the firing patterns of the postsynaptic neuron are similar to the presynaptic neuron and can be adjusted.

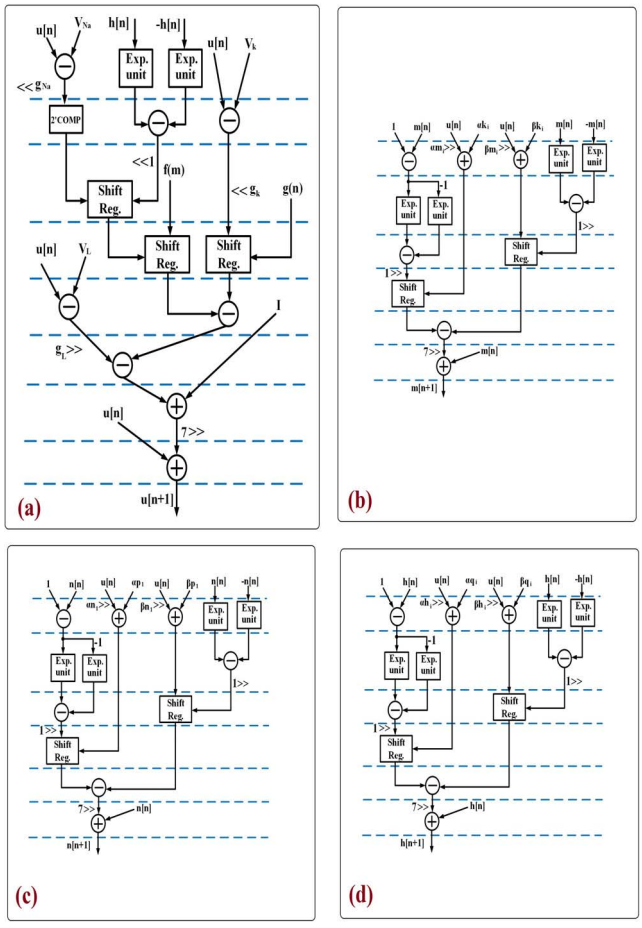

Fig. 8. Hardware implementation for the proposed MHHM neuronal system.

B. Bit-Width Definition

Discretizing differential equations is necessary due to implement digital functions. Also, using the Euler method reduced the computational overhead and complexity. To determine parameters and variables bit-width, taking two fundamental factors into account must be done. The logic shifts spans, and minimum bound and maximum bound of the parameter values are the mentioned factors. In the proposed HH model, the spans of U (voltage) are from -20 mV to 90 mV, and 8 bits are required for implementing the membrane potentials at least. Since the range of parameters can be varied from 2(-4n) to 2(+4n), this leads to a bit-width of 16. In the determination of bit-width, overflow may happen if the maximum logic shifts to left or right have not been considered. There is a bit-width of 40 bit, 20 bits for the integer section and 20 bits for the fraction part, this segmentation increases calculations accuracy and overflow avoiding.

C. Hyperbolic Function Realization

On the other hand, converting the hyperbolic functions to modified functions (with the capability of implementation of the shift and add modules) is necessary. The new modified function will be used for implementing the proposed MHHM neuronal system. In MHHM models, hyperbolic functions have the exponential terms that there is a possibility to change them in digital design by powers of 2. In [11] has been discussed about a multiplierless implementation method. The exponential functions in this approach are converted to 2^x functions that have been realized by logic shifts. If multipliers replace with logic shift operations, it will lead to an implementation that is significant and low-cost. Hence, based on this approach, the hyperbolic functions can be computed.

To implement $f(m)$ and $g(n)$ functions, this method is applicable in differential equations, as depicted in Fig. 9.

D. Resources Comparison

As mentioned, the original HH model is a complex model with a large number of nonlinear terms such as multipliers, divisions, etc. On the other hand, for implementing a high speed and low-cost system, these terms have been eliminated in the proposed MHHM model. Indeed, by using add and shift operations, all multiplications and nonlinear terms in the original HH have been replaced. Thus, an efficient neuronal system can be implemented. This approach is essential for large-scale neuronal implementation in case of the maximum

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on May 11,2020 at 01:08:10 UTC from IEEE Xplore. Restrictions apply.

TABLE VI
LOW-LEVEL DEVICE UTILIZATION OF XILINX VIRTEX-II PRO AND BASIC ELEMENTS USED PERCENTAGE FOR IMPLEMENTATION OF NEURON MODEL

Resources	Yaghini et al. [19] (Virtex7)	Akbarzadeh et al. [25] (Artix-7)	Khoyratee et al. [26] (Kintex-7)	Proposed MHHM (Virtex-2)
Slice Flip Flops Number Flops	50228(9%)	25430(9%)	1552(NA)	2840(10%)
4 input LUTs Number	86032(33%)	29130(21%)	4735(NA)	5660(21%)
Overall Saving in FPGA	58%	70%	NA	69%
Max speed (Frequency)	63.386MHz	71.4 MHz	100 MHz	85MHz

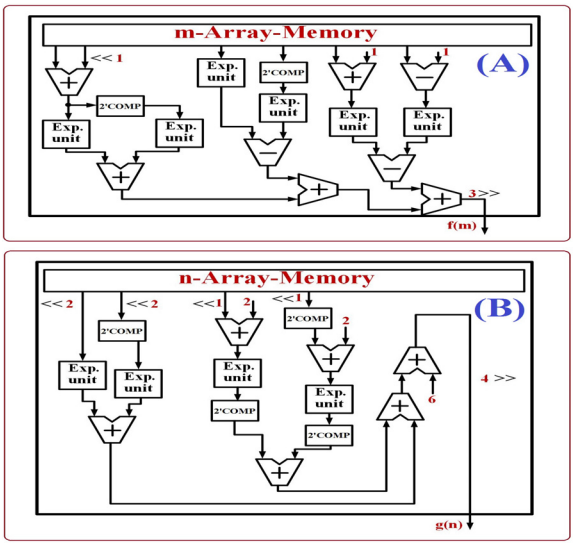

Fig. 9. *f*(*m*) and *g*(*n*) Calculation: By using the Exponential Unit (*Exp. Unit*) [12], the hyperbolic terms in the proposed model can be implemented.

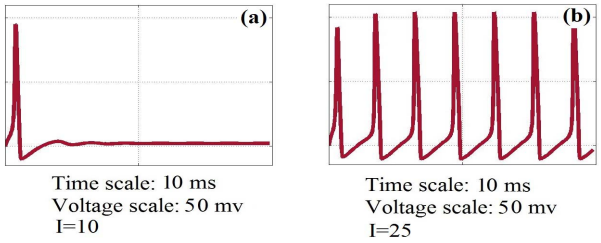

Fig. 10. The output of implementing presented MHHM neuron on XILINX Virtex-II Pro XC2VP30.

number of implemented neurons on the FPGA board. Consequently, the proposed MHHM neuron is an excellent choice for large-scale realization in comparison with the original HH model because all nonlinear terms and multiplications in this original model have been removed.

VIII. IMPLEMENTATION RESULTS

To the substantiation of concept, the MHHM neuronal system has been fulfilled on an FPGA device with this model: XILINX XUP Virtex-II Pro. The mentioned system produces a hardware platform comprising a high-performance Virtex-II Pro XC2VP30 FPGA. Spiking activity for the MHHM neuronal system has been displayed in Fig. 10. To generate this analog output, an 8-bit Digital to Analog Converter (DAC) has been used. An effectual comparison of the CORDIC-based HH model (Yaghini *et al.* [19]). And our suggested MHHM model has been shown in Table VI. Utilizing the device has been evaluated for implementing these models. The overall saving makes this model possible to implement more neurons in SNNs. The overall saving in this paper is 69%. Also, the presented work in [19] shows the 58 percent saving in the FPGA board that is less than the proposed model, and the presented work in [25] depicts the 70 percent saving in the FPGA board that is close to the proposed model, although it operates at less frequency. It should be noted that the proposed model is implemented on the Virtex-2 FPGA board with fewer resources than other works that use from high-performance platforms such as Virtex-7, Artix-7, and Kintex-7 FPGA boards. The number of 150 proposed MHHM neurons in this paper is realized in one core of the Virtex-2 development board. On the other hand, Yaghini *et al.* [19] implemented 150 HH neurons on its Virtex-7 FPGA board, Akbarzadeh *et al.* [25] realized the number of 500 HH neurons on its Artix-7 FPGA board and Khoyratee *et al.* [26] implements 1034 neurons (FS mode in one core) on its Kintex-7 FPGA board. As can be seen in Table VI, our proposed model is capable of being applied in large-scale neural networks. According to other papers ([19], [25], and [26]), that are used high-performance FPGA boards for implementing their neuronal models, in this paper, the proposed MHHM neuron has been implemented on VirtexII FPGA board by fewer resources compared with other FPGA boards. It is noted that the resource usage in our implementation is more than or equal to other high-performance FPGA boards, but we succeed in implementing it in the Virtex2 FPGA board. Furthermore, we are not using DSP blocks compared to [19], [25], and [26]. These two advantages prove that our realization is acceptable. Clearly, by using the high-performance FPGA boards, the number of implemented neurons will be increased significantly. Based on the proposed system implementation, it is demonstrated that our model is capable of reproducing all patterns of the original model in high accuracy and low-cost state. This operation is predictable due to eliminating a large number of high-cost multipliers in the original model. On the other hand, the division operation also is a complex and high-cost module. Consequently, the original model is not suitable for large-scale neural networks because of its vast realization compared to the proposed MHHM model. As can be shown in Fig. 8 and Fig. 9, to implement the proposed digital system, the ADD, SUB, and SHIFT modules only employed. It leads to achieving a high-speed and low-cost architecture that is used in real biological neural networks.The aim of this work was primarily to provide a scalable and tunable platform allowing the study of neurological diseases. The precise studying of the brain and understanding the possibilities of connecting nerve cells with the machine is a very important issue in neuroscience [32]-[34]. By implementing the large-scale

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on May 11,2020 at 01:08:10 UTC from IEEE Xplore. Restrictions apply.

neural networks, containing a large number of MHHM neuronal models and biological synapses, which have been connected to biological astrocytes, a real central nervous system will be achieved. This model is applied in investigating neuronal diseases such as epilepsy, Alzheimer's, essential tremor, Parkinson, etc. In other words, by applying these basic biological blocks in hardware cores, studying the curing of the diseases is achievable.

IX. Conclusion

A Multiplierless Hodgkin-Huxley Model (MHHM) has been presented in this paper. Results validate that the proposed MHHM neuron investigation in the time and phase domains follows similar dynamical behavior as the original HH model. This model is a multiplierless form of the original HH neuron and only requires logical shifts and adds. The number of implemented neurons on an FPGA chip can be increased by multiplierless realization of MHHM neuron on FPGA. The proposed model is far faster and consumes less area than the original HH model.

Acknowledgment

The authors would like to thank the Kermanshah Branch, Islamic Azad University, Kermanshah, Iran; for the financial support of this research project.

References

[1] L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *J. Physiol.*, vol. 117, no. 4, pp. 500–544, Aug. 1952.

[2] R. B. Szlavik, A. K. Bhuiyan, A. Carver, and F. Jenkins, "Neural–Electronic inhibition simulated with a neuron model implemented in SPICE," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 14, no. 1, pp. 109–115, Mar. 2006.

[3] S. Monk and H. Leib, "A model for single neuron activity with refractory effects and spike rate estimation techniques," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 4, pp. 306–322, Apr. 2017.

[4] J. Nagumo and S. Sato, "On a response characteristic of a mathematical neuron model," *Kybernetik*, vol. 10, no. 3, pp. 155–164, Mar. 1972.

[5] K. Rajković *et al.*, "Mathematical modeling of the neuron morphology using two dimensional images," *J. Theor. Biol.*, vol. 390, pp. 80–85, Feb. 2016.

[6] B. H. Jansen, G. Zouridakis, and M. E. Brandt, "A neurophysiologically-based mathematical model of flash visual evoked potentials," *Biol. Cybern.*, vol. 68, no. 3, pp. 275–283, 1993.

[7] T. Hishiki and H. Torikai, "A novel Rotate-and-Fire digital spiking neuron and its neuron-like bifurcations and responses," *IEEE Trans. Neural Netw.*, vol. 22, no. 5, pp. 752–767, May 2011.

[8] C. Bardy *et al.*, "Neuronal medium that supports basic synaptic functions and activity of human neurons *in vitro*," *Proc. Nat. Acad. Sci. USA*, vol. 112, no. 20, pp. E2725–E2734, May 2015.

[9] A. S. Cassidy, J. Georgiou, and A. G. Andreou, "Design of silicon brains in the nano-CMOS era: Spiking neurons, learning synapses and neural architecture optimization," *Neural Netw.*, vol. 45, pp. 4–26, Sep. 2013.

[10] M. F. Simoni and S. P. DeWeerth, "Adaptation in a VLSI model of a neuron," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 46, no. 7, pp. 967–970, Jul. 1999.

[11] S. Gomar and A. Ahmadi, "Digital multiplierless implementation of biological adaptive-exponential neuron model," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 4, pp. 1206–1219, Apr. 2014.

[12] M. A. Imani, A. Ahmadi, M. RadMalekshahi, and S. Haghiri, "Digital multiplierless realization of coupled wilson neuron model," *IEEE Trans. Biomed. Circuits Syst.*, vol. 12, no. 6, pp. 1431–1439, Dec. 2018.

[13] S. Haghiri, A. Zahedi, A. Naderi, and A. Ahmadi, "Multiplierless implementation of noisy Izhikevich neuron with low-cost digital design," *IEEE Trans. Biomed. Circuits Syst.*, vol. 12, no. 6, pp. 1422–1430, Dec. 2018.

[14] F. Grassia, T. Kohno, and T. Levi, "Digital hardware implementation of a stochastic two-dimensional neuron model," *J. Physiol.-Paris*, vol. 110, no. 4, pp. 409–416, Nov. 2016.

[15] M. Nouri, R. Karimi, A. Ahmadi, and D. Abbott, "Digital multiplierless implementation of the biological FitzHugh–Nagumo model," *Neurocomputing*, vol. 165, pp. 468–476, Oct. 2015.

[16] T. Levi, F. Khoyratee, S. Saïghi, and Y. Ikeuchi, "Digital implementation of Hodgkin–Huxley neuron model for neurological diseases studies," *Artif. Life Robot.*, vol. 23, pp. 10–14, Sep. 2018.

[17] H. Soleimani, A. Ahmadi, and M. Bavandpour, "Biologically inspired spiking neurons: Piecewise linear models and digital implementation," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 12, pp. 2991–3004, Dec. 2012.

[18] M. Hayati, M. Nouri, S. Haghiri, and D. Abbott, "Digital multiplierless realization of two coupled biological morris-lecar neuron model," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 62, no. 7, pp. 1805–1814, Jul. 2015.

[19] S. Yaghini, H. Asgharian, S. Safari, and M. Nili, "FPGA implementation of a biological neural network based on the Hodgkin–Huxley neuron model," *Frontiers Neurosci.*, vol. 8, p. 379, Nov. 2014.

[20] S. Yang *et al.*, "Cost-efficient FPGA implementation of basal ganglia and their parkinsonian analysis," *Neural Netw.*, vol. 71, pp. 62–75, Nov. 2015.

[21] M. Heidarpur, A. Ahmadi, M. Ahmadi, and M. Rahimi Azghadi, "CORDIC-SNN: On-FPGA STDP learning with izhikevich neurons," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 7, pp. 2651–2661, Jul. 2019.

[22] S. Yang, X. Wei, B. Deng, C. Liu, H. Li, and J. Wang, "Efficient digital implementation of a conductance-based globus pallidus neuron and the dynamics analysis," *Phys. A, Stat. Mech. Appl.*, vol. 494, pp. 484–502, Mar. 2018.

[23] S. Yang *et al.*, "Scalable digital neuromorphic architecture for large-scale biophysically meaningful neural network with multi-compartment neurons," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 1, pp. 148–162, Jan. 2020.

[24] S. Yang *et al.*, "Digital implementations of thalamocortical neuron models and its application in thalamocortical control using FPGA for Parkinson's disease," *Neurocomputing*, vol. 177, pp. 274–289, Feb. 2016.

[25] K. Akbarzadeh-Sherbaf, B. Abdoli, S. Safari, and A.-H. Vahabie, "A scalable FPGA architecture for randomly connected networks of hodgkin-huxley neurons," *Frontiers Neurosci.*, vol. 12, p. 698, Oct. 2018.

[26] F. Khoyratee, F. Grassia, S. Saïghi, and T. Levi, "Optimized real-time biomimetic neural network on FPGA for bio-hybridization," *Frontiers Neurosci.*, vol. 13, p. 377, Apr. 2019.

[27] E. M. Izhikevich, *Dynamical Systems in Neuroscience*. Cambridge, MA, USA: MIT Press, 2007.

[28] H. K. Khalil, *Nonlinear Systems*, vol. 3. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[29] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge, U.K.: Cambridge Univ. Press, 2002.

[30] D. E. Postnov, L. S. Ryazanov, and O. V. Sosnovtsev, "Functional modeling of neural–glial interaction," *Biosystems*, vol. 89, pp. 84–91, May/Jun. 2007.

[31] D. E. Postnov, R. N. Koreshkov, N. A. Brazhe, A. R. Brazhe, and O. V. Sosnovtseva, "Dynamical patterns of calcium signaling in a functional model of neuron–astrocyte networks," *J. Biol. Phys.*, vol. 35, pp. 425–445, Jun. 2009.

[32] S. Buccelli *et al.*, "A neuroprosthetic system to restore neuronal communication in modular networks," *iScience*, vol. 19, pp. 402–414, Jan. 2019.

[33] H. Keren, J. Partzsch, S. Marom, and C. G. Mayr, "A biohybrid setup for coupling biological and neuromorphic neural networks," *Frontiers Neurosci.*, vol. 13, p. 432, May 2019.

[34] T. Levi, "Closed-loop systems for next-generation neuroprostheses," *Frontiers Neurosci.*, vol. 12, p. 26, Feb. 2018.

Authorized licensed use limited to: University of North Carolina at Charlotte. Downloaded on May 11,2020 at 01:08:10 UTC from IEEE Xplore. Restrictions apply.