A Hardware-Friendly Real-Time Implementation of the Auditory Attention Based on a Novel Spiking Winner-Take-All Network

Behrooz Abdoli and Saeed Safari

***Abstract*—In recent decades, the modeling and designing of neuromorphic systems have received considerable attention, leading to the development of electrophysiological devices that can mimic the dynamic behavior of cortical networks. Some features, such as parallel processing, real-time performance, reconfigurability, low production cost, and good accuracy, make field-programmable gate arrays (FPGAs) an ideal hardware platform for implementing spiking neural networks (SNNs), which has become very popular in the field of neural computing. In this work, we investigated a high-performance FPGA design of the auditory attention (AA) to simulate the on-chip "cocktail party" effect similar to what happens in the brain. In the first step, we introduce a novel spiking winner-take-all (SWTA) network consisting of a hardware-friendly Izhikevich neuron model and explore its biological characteristics and stability. We then use this network as a receptive field in the primary auditory cortex, which is tuned by attention signals. The desired sound frequency will be selected as the target and transmitted to the higher layers of the auditory cortex for further processing, while other frequencies are severely suppressed. Experimental results show that the proposed network accurately follows the dynamic behavior defined in the AA theory and emphasizes the resource consumption after post place and routes on a fully efficient hardware implementation, where less than 4% of the resources were used to implement a network with 1856 neurons.**

***Index Terms*—Auditory attention (AA), field-programmable gate array (FPGA), neuromorphic hardware, spiking neural network (SNN), winner-take-all (WTA).**

I. Introduction

The brain continuously receives a large number of sequences of the sensory stimuli, and its cognitive functions enable it to parse and process information, then recognize, and ultimately respond correctly. These stimuli which translate into spikes, allow us to see and hear, navigate around, participate in a conversation, and enjoy a special taste. The research on the brain has a very rich history and noteworthy knowledge has been gathered, however, the functionality of the brain remains a mystery and is poorly understood. Despite facing the myriad of sensory inputs, it seems that at any specified time the nervous system processes only a narrow number of them [1]. Throughout the analysis of this scene, attention plays a crucial role in modulating neural selection. This study attempts to shed light on the processing of auditory attention (AA) in the cortex, in particular, the context of the "cocktail party" effect, then implement the results on a digital platform.

Imagine yourself at a crowded party where you are bombarded by a multitude of sound sources. In such circumstances, you are able to focus your attention on a conversation with a specific person. One of the greatest perceptual challenges of the auditory system is selectively paying attention to a speaker, while the other irrelevant competing speech and noise are suppressed. This phenomenon is known as the cocktail party effect [2].

Neurons in A1 are sharply adjusted for sound frequency, and there are iso-frequency bands that respond to a specific sound spectrum [3], [4]. Receptive fields (RFs) of auditory-cortical neurons can reshape according to attention signals to select the desired frequency of the auditory stimuli. Thus, RF effectively filters attended frequency from irrelevant stimuli. Signals received from attention compete with each other to form the desired RF. So what is needed is an approach to design a spiking winner-take-all (SWTA) network so that receives attention signals and filters out unwanted audio signals by identifying the winner in attention.

The SWTA is one of the most prominent computational microcircuits in the cortex that populations of excitatory and inhibitory neurons compete for activation [6], [7]. Max-selection, amplification, and filtering are three major features of SWTA networks that we will greatly benefit from in designing our network. The connectivity structure and intensity of the synapses in an SWTA circuit are vital matters that must be precisely fine-tuned to emerge stable localized attractor dynamics [9].

During the past few decades, researchers in the fields of hardware and software engineering have concentrated on designing frameworks that can mimic biological behaviors, such as vision, audition, and motion. Various CPU-based solutions have been proposed to simulate biological networks. Among them, we can mention Brian2 [12], which is a Python-based framework and operates in the clock-driven mode. Based on the serial nature of CPUs, computation is not well parallelized for large-scale networks, which leads to a slow execution speed. All software frameworks suffer

Manuscript received 18 May 2022; revised 4 October 2022; accepted 18 November 2022. Date of publication 23 November 2022; date of current version 19 July 2023. This article was recommended by Associate Editor Z. Zhang. *(Corresponding author: Behrooz Abdoli.)*

The authors are with the High Performance Embedded Architecture Lab., School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran 0098, Iran (e-mail: abdoli.behrooz@ut.ac.ir; Saeed@ut.ac.ir).

Digital Object Identifier 10.1109/TCAD.2022.3224091

from this issue [13]. Then, GPUs have come to the aid of CPUs because of their parallel computing power. For instance, CARLsim4 [14] is a software framework designed to develop spiking neural network (SNN) networks on GPU, based on C++ and CUDA. An important feature of SNNs is sparsity, where only the neurons with receiving input spike need to be calculated, and other neurons are idle. The GPU architecture is not well designed to support the sparse feature, which results in low computational efficiency [13]. In addition, the power consumption of GPUs is an obstacle to its use in SNN application [15]. Therefore, a field-programmable gate array (FPGA) with its unique features was proposed as a suitable digital hardware platform for this field.

In order to demonstrate the computational power and scalability of the different systems, in [16], hardware performance has been compared with three alternative approaches, including CPU, GPU, and FPGA. A cost function for the computational efficiency is defined, where the experimental computational time on the simulation system is divided by the biological activity time. The results of the implementation on three different platforms showed that in addition to the higher computational speed in the FPGA, this hardware platform has a significant advantage over the other two competitors in terms of scalability. While the computational efficiency of the two frameworks based on CPU and GPU faces a significant drop with increasing the network size, the FPGA is able to follow the execution routine with a relatively constant trend. New research in the field of neuromorphic engineering reveals that the tendency of designers has increased greatly to exploit FPGA as the heart of their accelerator for SNNs [17], [18], [19]. FPGAs provide the user with a scalable and reconfigurable design, a customized and flexible architecture, with reasonable development time and cost. For instance, [18] investigated FPGA implementation of neuron-astrocyte interaction and considered calcium dynamics based on a functional model. This kind of connectivity can facilitate modulation of synapse transmission that can lead to improving the learning process in SNNs. Some studies mimic biological applications such as [21]. They utilized piecewise linear (PWL) approximation and multiplier-less techniques to implement high-performance conductance-based subthalamic nucleus on FPGA. Finally, an interesting approach to efficient hardware implementation of large-scale SNNs is presented by [22]. They simulated 5120 randomly connected HH neurons in real time. Resource sharing and PWL approximations in their architecture improved computational efficiency.

Here, we look into a gap in the digital implementations of the SNN, which is the emulation of the functional dynamics of biological networks. We investigate a high-performance, modular FPGA accelerator to pursue the dynamical behavior of AA in real time. The hardware target is Artix-7 XC7A200T, which is a mid-range FPGA of the Xilinx family. The architecture is entirely manually coded and synthesized in Verilog HDL. The modified IZ neuron model is employed, and a Q10.20 fixed-point number representation is applied to the neuron model and synapses parameters due to the appropriate accuracy. In this article, an SWTA network was initiated based on the interaction between excitatory and inhibitory cells, and its stability was evaluated in detail. Second, the possibility of using this network as a biological filter in the auditory cortex was examined. Eventually, a network that could highlight the role of attention in A1 was designed and implemented on the FPGA.

This article is organized as follows. Section II describes the dynamics of the hardware-friendly (HF) IZ model. Section III presents SWTA analysis, design, and stability. In Section IV, the AA models are reviewed and then the proposed model based on SWTA is discussed. The hardware architecture and its details are brought in Section V. Experimental results and its related analysis are explained in Section VI. In Section VII, the results are concluded.

II. Biological Neuron Model

This article explores the possibility of the hardware implementation of a cognitive function in the brain. It is necessary to exploit a neuronal model that mimics the dynamics of biological neurons. There are various spiking neural models that follow the nonlinear dynamic system, ranging from the simplest model LIF to the most complicated HH. Among them, there are other diverse models, including IZ [23] and FitzHugh–Nagumo [24]. The selected model must be biologically validated for the analytical results to be meaningful. Reciprocally, the neuron computations should be appropriate due to limited hardware resources and real-time implementation.

We implemented the LIF, IZ, and HH models on FPGA and showed the results of hardware consumption of the DSP blocks and the lookup tables (LUTs) in Table I. The IZ neuron model is capable of reproducing the dynamic characteristic of biological neurons, such as regular spike and bursting. It is also one of the simplest possible spiking models that does not require intensive computations. Based on these valuable features, we employ this model to design our SWTA network. It is composed of 2-D differential equations for both excitatory and inhibitory neurons

$$C\frac{dv}{dt} = k(v - v_r)(v - v_t) - u - I_N \tag{1}$$

$$\frac{du}{dt} = a\{b(v - v_r) - u\} \tag{2}$$

where v, v_r, and v_t are the membrane voltage, the resting potential of the neuron, and the threshold voltage, respectively. u is related to the recovery variable, and C refers to the membrane capacitance. I_N consists of the external (I_{ext}), and synaptic (I_{syn}) currents, and a and b are constant coefficients. When the voltage membrane reaches to the peak potential, the neuron fires a spike and the values of v and u will reset to c and $u + d$, where c and d are constant parameters [23].

Previously, nonlinear equations of IZ have been approximated by PWL techniques [25], but in this article, only the constant coefficients are slightly modified to replace the multiplication operations by shifters and adders. Since implementing multiplication operation is costly from the hardware resources point of view, an HF design can be achieved by converting the values of constant coefficients to numbers with a power of 2 and performing shift operation. We employed

TABLE I
HARDWARE RESOURCES CONSUMPTION OF THE NEURAL MODELS

Neuron model	-	DSPs	-	LUTs
LIF		2		95
IZ		2		230
HH		56		2100

TABLE II
PARAMETERS IN ORIGINAL AND MODIFIED MODELS

Parameters	Original model[23]	HF model
a	$\frac{1}{100}$	$\frac{1}{128}+\frac{1}{256}$
b	5	4 + 1
k	3	2 + 1
C	100	64 + 32 + 4

the Euler method for solving differential equations by approximating time intervals $dt = 1/16$ ms which is small enough to solve the equations with suitable accuracy and is also a power of 2 to substitute compute-intensive division with a simple shift operation. Table II represents the values of the original and HF coefficients of the IZ neuron model for the spike regularly.

The synaptic current consists of four different receptors: *AMPA*, *NMDA*, *GABA*$_A$, and *GABA*$_B$

$$I_{\text{syn}} = g_{AMPA}(v-0) + g_{NMDA}(v-0) + g_{GABA_A}(v+70) + g_{GABA_B}(v+90). \tag{3}$$

For each receptor, the conductance is multiplied by individual reversing voltage and as shown in (3), the reversal voltages of *AMPA*, *NMDA*, *GABA*$_A$, and *GABA*$_B$ are 0, 0, -70 mv, and -90 mv, respectively. In (3), the first two receptors are excitatory and the next two components are inhibitory synapses. Each receptor has its own time constant, which can be obtained from the following equation:

$$\frac{dg}{dt} = -\frac{g}{\tau} \tag{4}$$

where $\tau = 5, 150, 6$, and 150 ms for *AMPA*, *NMDA*, *GABA*$_A$, and *GABA*$_B$, respectively.

III. SPIKING WINNER-TAKE-ALL DESIGN

So far, several topologies of the spiking networks have been proposed that follow WTA dynamics, such as surround inhibition, Gaussian inhibition, etc. [9]. However, we need a WTA networks structure which is compatible with the auditory cortex. Accordingly, in addition to the above-mentioned dynamics, our required network should also have columnar organization on the basis of the frequency (similar to the A1 area). Here, we describe the key concepts of our proposed SWTA network.

This network is illustrated in Fig. 1. The parameters of all neurons are defined as regular spike activity. The network comprises N distinctive SWTA that compete each other. Each SWTA consists of three neuronal groups: 1) the input neuron group; 2) the excitatory neurons group (ENG); and 3) the inhibitory neurons group (ING). There is a connection between each input neuron and all corresponding ENG neurons. Neurons within each ENG are linked together as fully connected, and they are also connected to all ING neurons except for their own group. The function of the ING is to transmit the inhibitory synapses to the ENG in the same group to suppress them. Excitatory to inhibitory ratio in each group is set to 4:1 to be more biologically realistic [25].

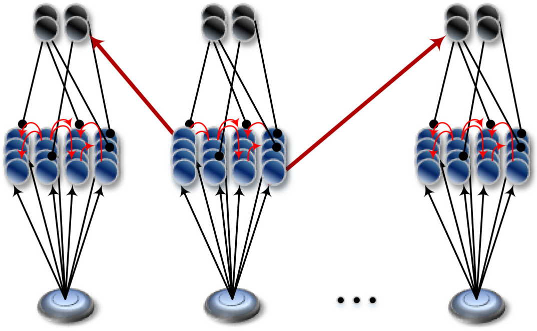


Fig. 1. Proposed SWTA structure. Several groups compete each other.

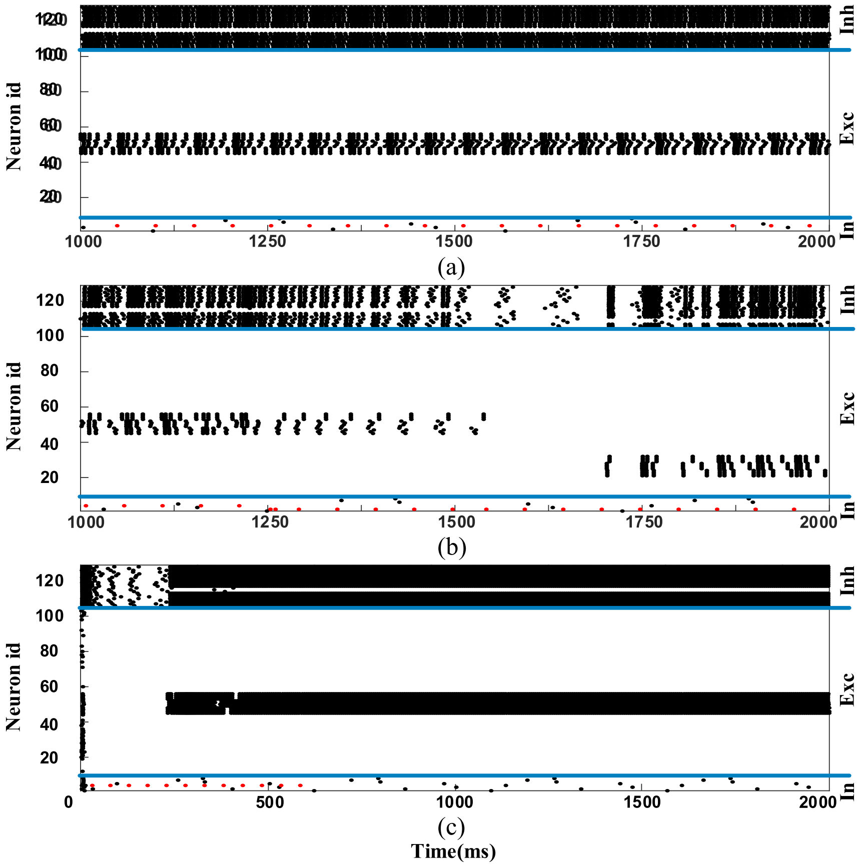
TABLE III
ALL TYPES OF SYNAPSES IN THE NETWORK ARE DESCRIBED

pre	post	type	inside/outside group	weight
input	ENG	Ex	in	w_{e1}
ENG	ENG	Ex	in	w_{e2}
ENG	ING	Ex	out	w_{e1}
ING	ENG	Inh	in	w_i

Overall, we face four different synapse connections that are completely described in Table III. The last column of this table is dedicated to synaptic weights and includes two excitatory weights (w_{e1} and w_{e2}) and an inhibitory weight (w_i). Here, we assume that w_{e1} and w_i are equal, while w_{e2}, which is corresponding to self-excitation inside the ENGs, is more than 50 times smaller (as will be discussed later in Section III-B). The input cells receive specific biological signals, and the neuron with the highest spiking activity can inject the most *NMDA* and *AMPA* into its ENG. Consequently, the corresponding ENG generates the maximum spiking rate and also transfers this trend to ING of other populations. Abundant spiking responses in other INGs lead to *GABA* penetration to their ENGs and subsequently shut them off. Therefore, the active input neuron allows its ENG and other INGs to regularly fire spike while other neurons are entirely quiet. With this trick, if a network combines with our SWTA, the winning ENG will amplify the paired network, while other INGs help the desired filter for that network.

A. Network Characteristics

In this section, we evaluate the behavior and characteristic of the proposed network. For the sake of clarity, we emulated a network that includes a total of 128 IZ neurons, it consists of eight competing clusters of SWTA, each containing one neuron for input, 12 neurons for ENG, and three neurons for ING. Fig. 2(a) illustrates the raster plot of the proposed network. This figure shows spikes recording between 1 and



(a)

(b)

(c)

Fig. 2. Characteristics of the proposed SWTA. (a) Proposed network raster plot, where the fourth group receives stronger input. (b) Switching time investigation. (c) Working memory.

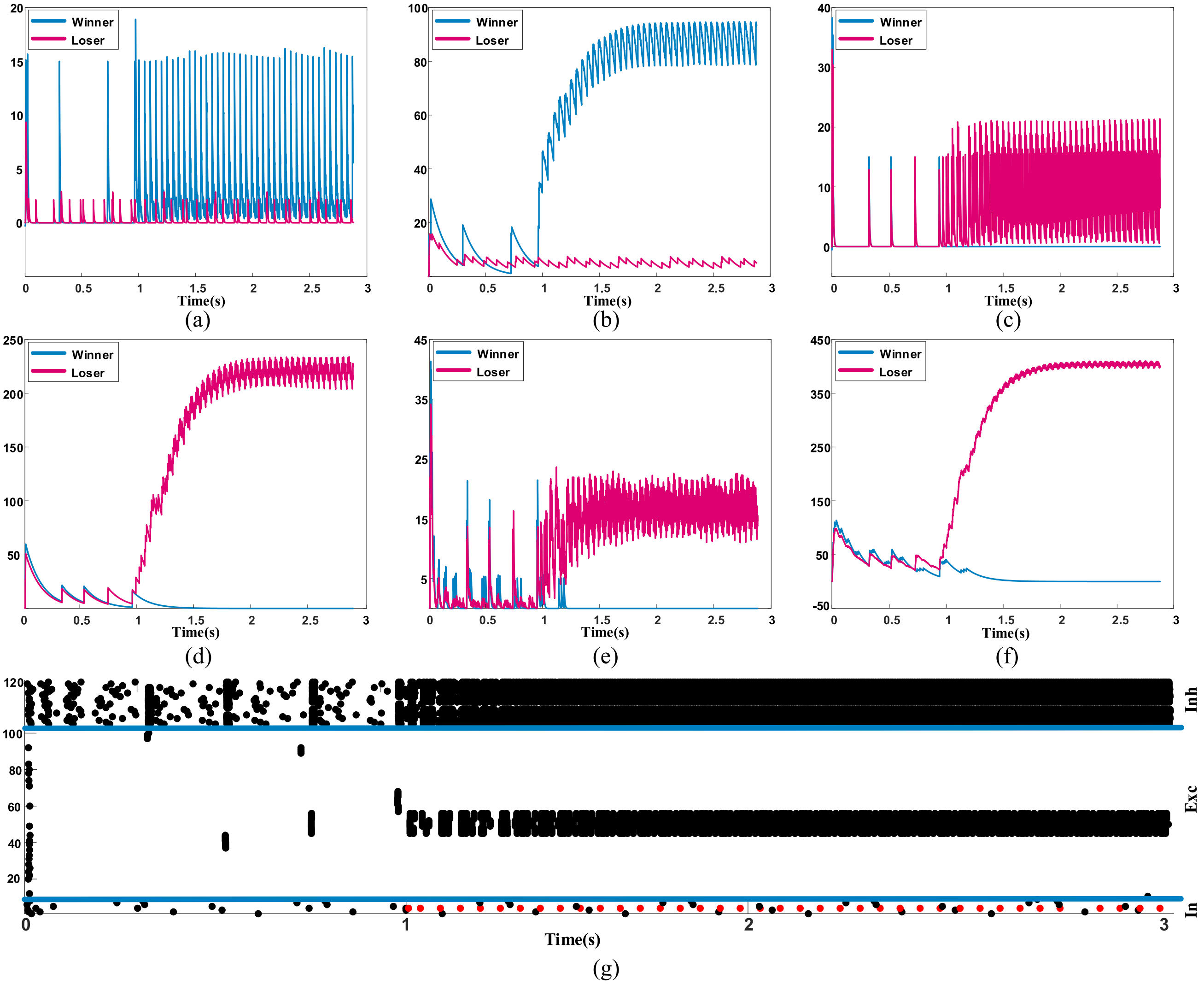
2 s after the onset of the simulation. Neurons with indices 1–8 are the input, the next 96 neurons are excitatory, and the last 24 neurons are inhibitory. All neurons are injected with an external current of 70 pA, except for the fourth input neuron, which receives an additional 500 pA. The spikes of this neuron are marked with red dots in Fig. 2(a), which are at a higher rate compared to the other input neurons. Fig. 2(a) shows that the fourth ENG (indices 45–56) is very active, while the excitatory neurons of the other groups are completely silent. Similarly, the inhibitory cells of the group that captures the most current (with neuron id 114–116) are inactive, while the *NMDA* and *AMPA* receptors spread in the post-synaptic of the fourth ENG have led to the activity of other inhibitory groups.

We have also examined the switching time, which is the time required for the desired output to appear on the network and for the winning group to shift when the input signal changes. Fig. 2(b) depicts this property in a way that from the beginning to the moment of $t = 1.25$ s the fourth input receives the most current, but just at this point, the second input takes up the maximum current [red dots in Fig. 2(b)]. As is visibly apparent in Fig. 2(b), the ENG output of the fourth group has gradually weakened from $t = 1.25$ s, and the neurons of the second group have overcome the others at about $t = 1.7$ s. In the proposed network, it takes about 450 ms to switch the output properly as soon as the input stimuli changes, which is apparently a sensible time for microcircuits in the cortex [26].

The next property of the network is related to the memory function in a steady state, which means that the previous winning population maintains its activity in the absence of a prominent input. This feature has a significant impact on cognition models such as state-dependent computations [27]. As mentioned earlier, w_{e2} is more than 50 times smaller than w_{e1} in the normal mode. For working memory, we can reduce this difference up to $w_{e2} = 0.1w_{e1}$ at the cost of slow switching time. Fig. 2(c) shows the recording of spikes from the start to $t = 2$ s in the working memory mode. The fourth input current in the first 600 ms is equal to 570 pA and after that is equal to 70 pA, like other neurons. Interestingly, once the input currents are equal, the winning group is able to sustain its stable position until one of the currents prevails.

B. Analysis of the Impact of Synaptic Receptors on the Formation of WTA

The neural arrangement along with the connection of different neurons to each other in the proposed structure has differentiated the amount of *AMPA*, *NMDA*, $GABA_A$, and $GABA_B$ receptors. In addition, different time constants of conductance channels can play a very effective role in shaping the stability of bump activity. For clarification, we bring up an example. Consider the situation where all the currents are equal, and after $t = 1$ s the current of the fourth input



(a) (b) (c) (d) (e) (f) (g)

Fig. 3. Comparison of synaptic receptors in the winning group and other groups. (a) Average *AMPA* conductance in excitatory neurons. (b) Similar to (a) except that *NMDA* conductance is considered. (c) Average *AMPA* conductance in inhibitory neurons. (d) Similar to (c) except that *NMDA* conductance is considered. (e) Average $GABA_A$ conductance in inhibitory neurons. (f) Similar to (e) except that $GABA_B$ conductance is considered. (g) Raster plot of the competition of eight groups.

increases. The goal is to observe the conductance of the different channels in all neuronal groups. Fig. 3 shows the details of this event. As can be seen from Fig. 3(g), before $t = 1$ s almost all neurons have irregular spiking activity, but after this time only the fourth ENG along with the ING of the other groups are active. After $t = 1$ s, three important synaptic effects occur.

1) The impact of excitatory synapses of the input neuron to the fourth ENG.
2) The impact of the excitatory synapses of the fourth ENG to the ING of other groups.
3) The impact of inhibitory synapses of all the INGs (except for the fourth) to their related ENGs.

Fig. 3(a) and (b) are related to case 1 and show the *AMPA* and *NMDA* conductances, respectively. The blue curves describe the average status of the winning neurons (fourth group), while the pink curves describe the average losing neurons (all other groups). Fig. 3(c) and (d) shows the excitatory current effect of *AMPA* and *NMDA* channels on inhibitory neurons, respectively. It depicts how the inhibitory neurons of the winning group fail to receive the excitatory currents while the neurons in the second group take a large amount of this current. Also, Fig. 3(e) and (f) illustrates the feedback impact of inhibitory neurons on the excitatory neurons of their groups, depicting the effects of $GABA_A$ and $GABA_B$, respectively. Correspondingly, as presented in these figures, there is no specific repressing effect on the winning ENG after $t = 1$ s. On the other hand, other ENGs are severely suppressed by both fast and slow conductance channels.

Here, we mathematically extract the state of the conductance channels of the excitatory neurons, and for simplicity, we assess only the *NMDA* and $GABA_B$ channels, which possess a longer time constant. We can assume that each ENG is considered as a single neuron that captures a self-excitation with the weight $(neurons - per - group) * w_{e2}$. As previously mentioned the $neurons - per - group$ is equal to 12. The condition of the excitatory neurons conductance over time is directly related to the rate of input excitatory and inhibitory spikes, and (4) can be rewritten as follows:

$$\frac{dg_{NMDA}}{dt} = -\frac{g_{NMDA}}{\tau_{NMDA}} + \nu_{\text{in}} w_{e1} + \nu_{ex}(12 * w_{e2}) \tag{5}$$

$$\frac{dg_{GABA_B}}{dt} = -\frac{g_{GABA_B}}{\tau_{GABA_B}} + \nu_{inh} w_i \tag{6}$$

where ν_{in}, ν_{ex}, and ν_{inh} are average firing rates of the incoming spike of the input, self-excitation, and inhibitory synapses, respectively, and w_x is the prespecified synaptic weight. As mentioned earlier, $w_{e1} = 50 w_{e2}$ and $w_{e1} = w_i$, so (7) can be expressed as follows:

$$\frac{dg_{NMDA}}{dt} = -\frac{g_{NMDA}}{\tau_{NMDA}} + w_{e1}(\nu_{\text{in}} + 0.24\nu_{ex}). \tag{7}$$

According to (6) and (7), average excitatory and inhibitory conductances can be solved for excitatory neurons as follows:

$$\begin{aligned} g_{NMDA}(t) &= w_{e1}(\nu_{\text{in}} + 0.24\nu_{ex})\tau_{NMDA} \\ &\quad + (g_{NMDA}(0) - w_{e1}(\nu_{\text{in}} + 0.24\nu_{ex})\tau_{NMDA})e^{\frac{-t}{\tau_{NMDA}}} \end{aligned} \tag{8}$$

$$\begin{aligned} g_{GABA_B}(t) &= w_i \nu_{inh} \tau_{GABA_B} \\ &\quad + \left(g_{GABA_B}(0) - w_i \nu_{inh} \tau_{GABA_B}\right)e^{\frac{-t}{\tau_{GABA_B}}}. \end{aligned} \tag{9}$$

In (8), ν_{in} and ν_{ex} for the winning group have considered 20 and 90 Hz, respectively, while these values for the losing groups are 3 and 0 Hz, respectively. Also in (9), ν_{inh} is 0 Hz for the winning group and 174 Hz for the losing groups. (These frequency rates are based on theoretical expectations [9].) Furthermore, $\tau_{NMDA} = \tau_{GABA_B} = 150$ ms and the only parameter that remains unknown in these equations is the amount of weight (w_{e1} and w_i). We conducted an extensive search to deduce the amounts of weight in which the network remained stable. According to the simulations, two low and high bounds for the weights were obtained, and between these two values, the network dynamics indicate a completely stable and accurate performance. The results showed that if the weights were less than 3.5 ns, there was no sufficient authority to determine the winner, and also if the weights were greater than 23 ns, the network would proceed toward instability and exhibit an unpredictable behavior.

The results of Fig. 3 are obtained for $w_{e1} = w_i = 15$ ns, and if we place these values in the first part of (8), which is related to the steady state of the *NMDA* conductance channel, 95 ns is extracted, which is approximately the same value as Fig. 3(b) in winning vectors [also, by placing these values in (8) and (9), the steady-state values of *NMDA* and *GABA*$_B$ are obtained according to Fig. 3(d) and (f)]. Therefore, the innovative connectivity topology in the proposed method makes the emergence of bump activity.

IV. Modeling Auditory Attention

In this section, we introduce a simple model of AA based on the proposed SWTA. First, the role of attention in the auditory system and cognitive process is investigated. Then, the proposed model for AA is presented.

A. Auditory Attention

Attention in the auditory system is described as a bottleneck in the brain processing that continuously samples input sensory stimuli and limits the brain resources by directing sensory and cognitive neurons to the most relevant events in the auditory system [28], [29]. Here, the cocktail party effect will be investigated, therefore, task-driven or selective attention is needed. The RF of the auditory neurons in A1 is reshaped by selective attention feedbacks to enhance transfer foreground sound to the secondary auditory cortex while effectively filter the irrelevant background auditory stimuli [5].

Although high-level cognitive signals in attention are ambiguous variables, experimental results show that these signals are constantly reorganizing acoustic RF in order to select attended sounds in the ever-changing environment. In [32], a selective attention model is introduced in which there is a set of descending projections from the nonprimary auditory cortex. Once incoming sounds are processed by these higher areas, the sounds are classified according to their characteristics such as frequency. At this point, the target sound is separated from the background, then an appropriate feedback signal is sent to the subcortical nuclei. This way, attention could tune RF in A1 and neuromodulator activities suppress interferes or facilitate target responses to the incoming streams. Fig. 4(a) shows the model of this process.

In the next section, we attempt to provide a suitable model of AA in the A1 area using the proposed SWTA network.

B. Proposed Model

As previously mentioned, the primary auditory cortex contains isolated neural bands that have fairly similar frequency responses. As shown in Fig. 4(a), input signals from the cochlea tonotopically ascend to the A1, which means the response of cochlear neurons with a specific frequency connects to a column of neurons with the same frequency in the A1 area. The primary auditory cortex performs the initial processing on the sounds received from the cochlea, and then a sample of the audio signals is also sent to high-level cognitive areas. Appropriate feedback signals which are originated from the high-level cognitive areas (selective attention) are returned immediately to A1 after detecting the target sound frequency. The auditory RF is set to conduct the desired signal and, like a filter, attenuate the unwanted frequency bands in A1. Attention to the target sound frequency is the main factor of analysis in our model.

Fig. 4(b) shows the final model of attention in the auditory system. We assume that there are also tonotopic maps between RF (feedback signals) and A1, meaning that at least one cognitive feedback signal is applied to each frequency column in A1. Given that the proposed SWTA behaves like a biological filter. we use this network as the RF in the auditory system, which receives its inputs from the attention section. The number of competing clusters in the SWTA is considered equal to the number of frequency channels in the auditory system so that each cluster establishes a connection between the attention area and A1 for a frequency channel. These clusters compete with each other for the effect of attention on the

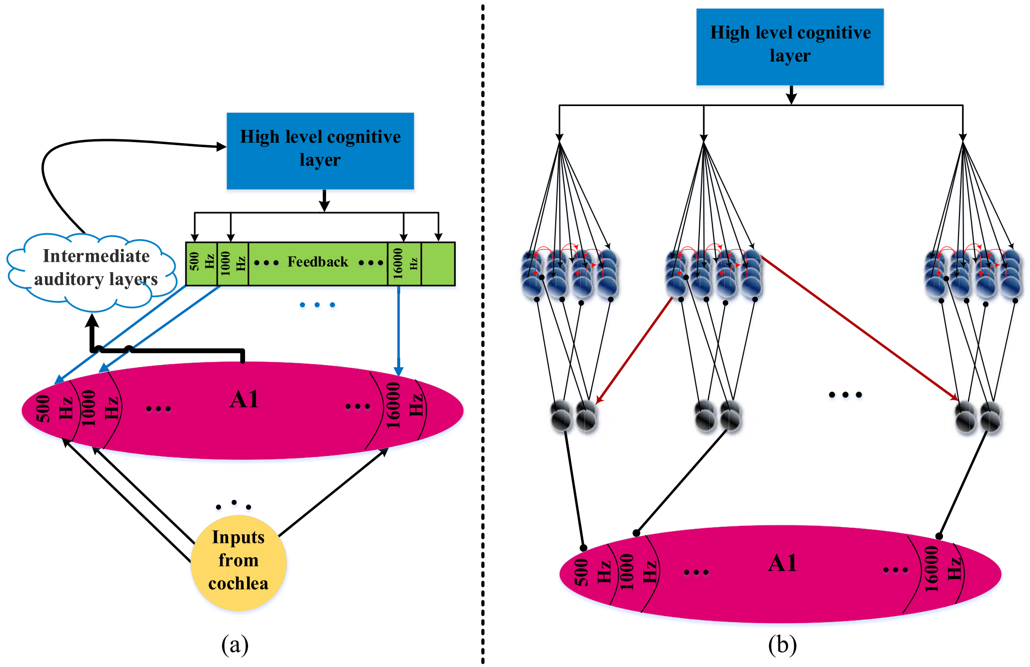


Fig. 4. (a) Attention originates from the high-level cognitive areas and tunes the auditory RF. These feedback signals affect different frequencies in the A1 area to attenuate undesirable signals, like a filter. (b) Proposed SWTA network plays the same role as the RF.

auditory system. After the high-level cognitive area detects the target sound frequency, it applies the distribution of spikes to the clusters in such a way that the group corresponding to the target sound in the SWTA wins the competition.

The proposed biological filter can both amplify or attenuate signals. For instance, if synapses from the ENG are connected to neurons of a frequency channel in A1, it will amplify the signal, but if inhibitory neurons are used, it will suppress the channel. Here, we use the filter attenuation property. Consequently, from the ING of each group, several inhibitory axons are sent to the corresponding column in A1. According to the information in Section III, INGs of all groups are extremely active, except the winner. Accordingly, it attenuates all irrelative frequencies in A1. Thus, the proposed SWTA network, like a filter, attenuates the frequency of noise sounds in A1, while the target sound can be transmitted to higher layers for detail processing without any inhibition. The proposed structure can be effective in the field of AA in the cocktail party effect.

V. Hardware Implementation

The proposed AA network, based on the IZ neuron model, can be efficiently implemented on the FPGA. The designed architecture was thoroughly manually coded in Verilog HDL, so most parameters in the network were generically defined to take advantage of both a scalable and a customizable system. Fig. 5 shows a general overview of the system.

The total number of neurons, the number of SWTA channels, the number of excitatory neurons in each group, the ratio of excitatory to inhibitory neurons, etc., are received (RX line) by the FPGA through packets with specific headers. For example, consider a packet with the value "4A4F-00D2-0A-0F-03." The first two bytes are the header, the next two bytes are the total number of neurons (here 210), the next byte is the number of channels (here 10), the byte after that is the number of excitatory neurons in each group (here 15), and finally the last byte is the ratio of excitatory to inhibitory neurons (here, one inhibitory neuron is considered for every three excitatory ones). Additionally, the output spikes of the neurons are sent to the PC serially (TX line) for graphical display. Eventually, the architecture comprises a datapath and a controller module which are further explained.

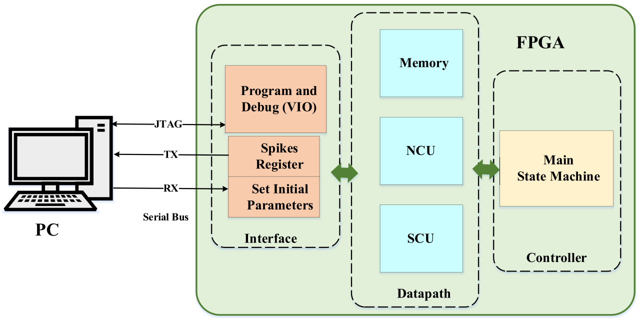


Fig. 5. Architecture of the proposed network. The FPGA is connected to the PC using two USB cables: one for programming and the other for serial communication.

The datapath of the 5-stage pipeline is shown in Fig. 6(a), in which a single computational module is reused multiple times to update several neurons' status. The two RAM blocks store the neurons and synapses variables. For instance, u and v are kept in neuron BRAM (N-BRAM) while the conductance of $AMPA$, $NMDA$, $GABA_A$, $GABA_B$, and I_{ext} is hold on

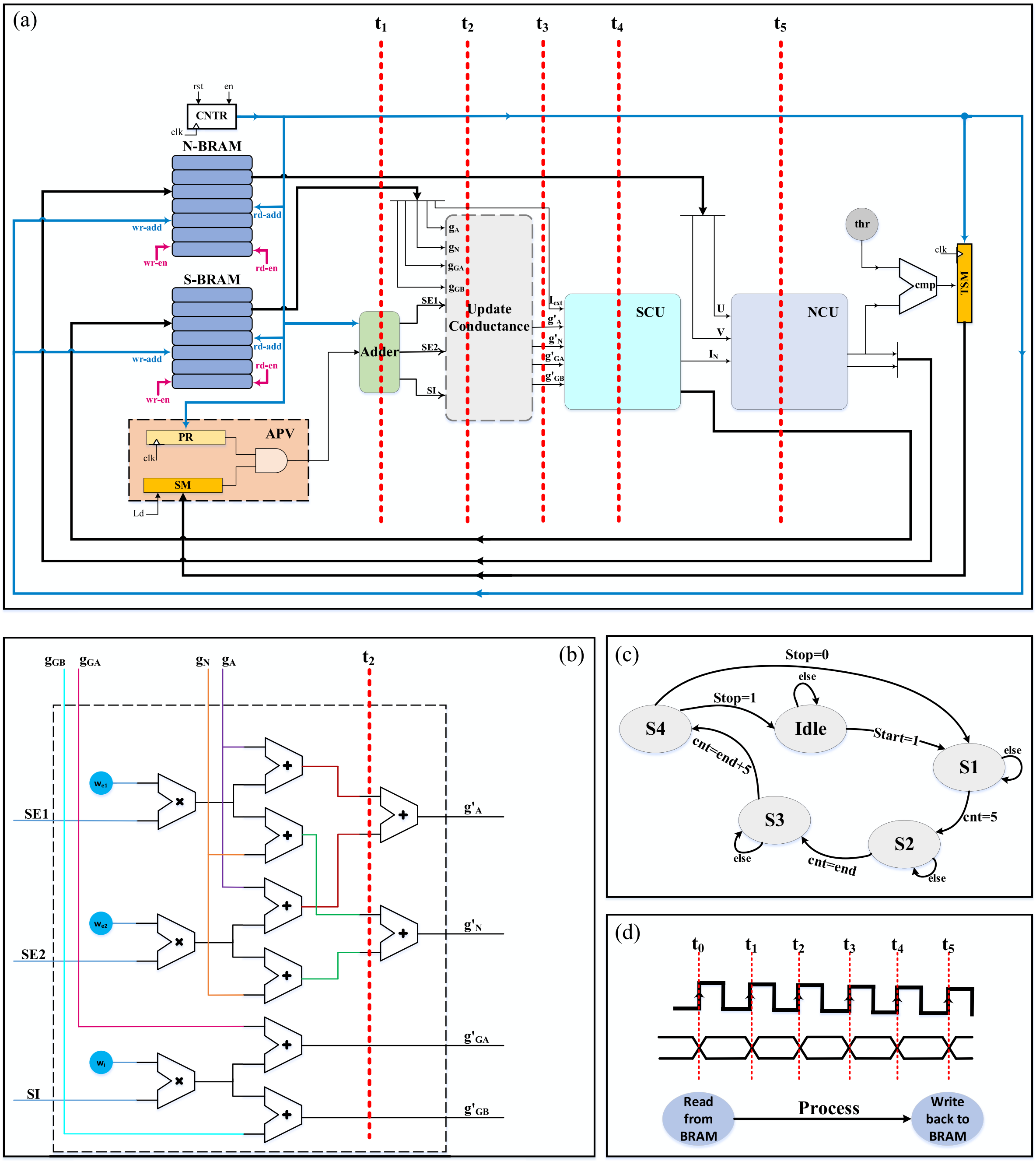


Fig. 6. Detailed pipeline architecture of the proposed SWTA. (a) Various modules in the datapath and their arrangement. (b) Signals and hardware inside the "update conductance" module. (c) Controlling finite-state machine of the system. (d) Timing diagram from the beginning to the end of the pipeline.

synapse BRAM (S-BRAM). Experimental results from the post-synthesis showed that this type of storage is more efficient in terms of resource utilization and routing compared to a single memory block, because they were selected in accordance with the default BRAM architecture in Xilinx 7 series FPGAs. A Q10.20 fixed-point arithmetic is employed to describe neuron and synapse variables. The information of each neuron is stored in a unique row of BRAMs, so each row consists of 60 and 150 bits for N-BRAM and S-BRAM, respectively. The depth of both BRAMs is equal to the number of existing neurons in the network. The initial values are randomly chosen and are distributed equally in a given range.

With any rising edge of the clock, the counter is increased by one. This counter generates a read address for both BRAMs and causes the data of that address to be extracted for processing and updating. At the same time, the active presynaptic vector (APV) must be identified in the previous iteration to correctly update conductance of the desired neuron. In these conditions, the system requires two registers, one to keep the spike status of all the neurons in the previous iteration, and the other to determine which neurons are involved in the presynaptic connection of the target neuron. The spike memory (SM) is a register with a length equal to the total number of neurons and contains 1s and 0s which indicate spikes have occurred or not in the previous iteration, respectively. This register will be up to date at the end of each iteration. A binary $N * N$ matrix is required to determine the presynaptic of the neurons, which N indicates the total number of neurons. Each row of this matrix describes the presynaptic neurons that are connected to the neuron associated with that row. Because storing this matrix consumes a huge amount of memory, alternative methods are proposed.

To solve this problem, we proposed an N bits presynaptic register (PR) to display connected neurons. Based on predefined initial seeds and shift operation, PR for each neuron in each clock cycle generates N-bits binary vector that shows the presynaptic neurons connected to the target neuron. This mechanism drastically diminishes memory usage for storing the connectivity matrix. Active presynaptic neurons for the target neuron are determined by bitwise AND of SM and PR registers.

Then, this vector goes to the adder module, which counts the number of "1"s in APV. As mentioned earlier, three different types of fixed weights were defined for the network that divide each APV vector into three components. For example, in a network with N neurons, including k input neurons, l excitatory neurons, and m inhibitory neurons, the number of 1s that exist in the first k bits of the APV, are aggregated and stored in the sum excitatory 1 (SE1) register. Similarly, for the bits K to $k + l - 1$ and $k + l$ to $N - 1$, the accumulated values are stored in registers sum excitatory 2 (SE2) and sum inhibitory (SI), respectively. This process is performed by three parallel modules that combine the counters and the adder trees.

In the next module, the conductances will be updated, which its internal schematic is shown in Fig. 6(b). Each of the output registers of the adder module is multiplied by the corresponding weight, then accumulated with the prior conductance value. Eventually, since two different excitatory weights are considered for the network, the excitatory conductance of *AMPA* from each of the weights is added together to obtain the final value. This process also happens for *NMDA*.

Following the datapath, the synapse computing unit (SCU) and neuron computing unit (NCU) modules are serially located. In the SCU, (3) is first calculated, then I_{syn} is added to I_{ext} to obtain I_N. Finally, u and v values are updated according to (1) and (2) in the NCU. The new v is always compared to the spike threshold. If the membrane voltage exceeds this threshold, the corresponding bit of the temporary SM (TSM) will be 1; otherwise, it will record 0. The TSM register has the following features.

1) Its length is equal to the total number of neurons.
2) It resets at the beginning of each iteration.
3) During the update of neurons, a one-bit shift to the left occurs with each rising edge of the clock.
4) At the end of each iteration, its value is loaded into the SM register.

A finite-state machine (FSM) controls this datapath and is shown in Fig. 6(c). Upon receiving the start signal, the counter is enabled to start working and the data is read from the BRAM. When the counter shows the number 5, it means that the data related to the zero address has passed through the five-stage pipeline and reached the end of the path and the updated data is overwritten at the relevant address. Therefore, the "write-enable" signal in BRAM is activated while the "write-address" is generated using a five-time delay in the counter. In the next state, the counter reaches the last address of the BRAM, the counting should be stopped. After the five rising edges of the clock, the pipeline is emptied and the status of each neuron is updated. The following steps happen in the last state.

1) The SM register "load" is activated.
2) The TSM register is reset.
3) The counter is reset.

The high-performance implementation could be achieved by the pipeline design that can improve the throughput of the digital circuit. Fig. 6(d) describes the proposed 5-stage pipeline timing diagram in which data is fetched from BRAM in t_0 and passed through the stages sequentially. The pipeline can introduce a higher clock frequency that leads to a real-time performance.

VI. Experimental Results

In order to characterize and evaluate the performance of the real-time digital cocktail party effect based on the SWTA network, the setup discussed in Section IV was implemented on a Xilinx Artix-7 FPGA. The number of channels in this architecture is reconfigurable and here, we have implemented a network with 64 frequency channels in the A1 area. Previously, numerous implementations have been performed to model audio from the input of the ear to the output of the cochlea [33], [34]. Their output neurons can be the inputs of our network, and the firing rate of these neurons is proportional to the intensity of the input sound at a specific frequency.

For each frequency channel, there is an input that receives its data from its co-frequency areas in the cochlea. The purpose of this article is to investigate the spike patterns of the neurons in the A1 area in which several sounds with different frequencies enter this region and using the proposed SWTA will try to apply the attention effect to the A1 area. Therefore, the received spikes are modeled by the analog current (I_{ext}) to become independent of the cochlea synapses. In this implementation, the I_{ext} of all neurons is assumed to be 60 pA, which, due to dynamical simulations, cannot lonely produce the spikes. The serial port is used to apply more I_{ext} to the input neurons. Using serial software, two bytes of hexadecimal value "4D4F" are sent at first to the FPGA as headers. This value indicates the set up of the new external current for the input neurons. Then, one byte is sent as the desired input neuron address. Finally, the next 30 bits determine the I_{ext} value, which store in the S-BRAM. The input neurons indices in this implementation are from 1 to 64.

In this implementation, all the A1 neurons are excitatory and 12 neurons are considered for each of the frequency channels. All 12 neurons in a frequency band receive excitatory synapses of w_{e1} weight from the input neuron of that band. Eventually, the indices of the A1 neurons in this design (for 64 channels, each contains 12 neurons) will be from 65 to 832.

The attention network utilizes the same proposed SWTA, which includes 64 channels. For each A1 channel, there is

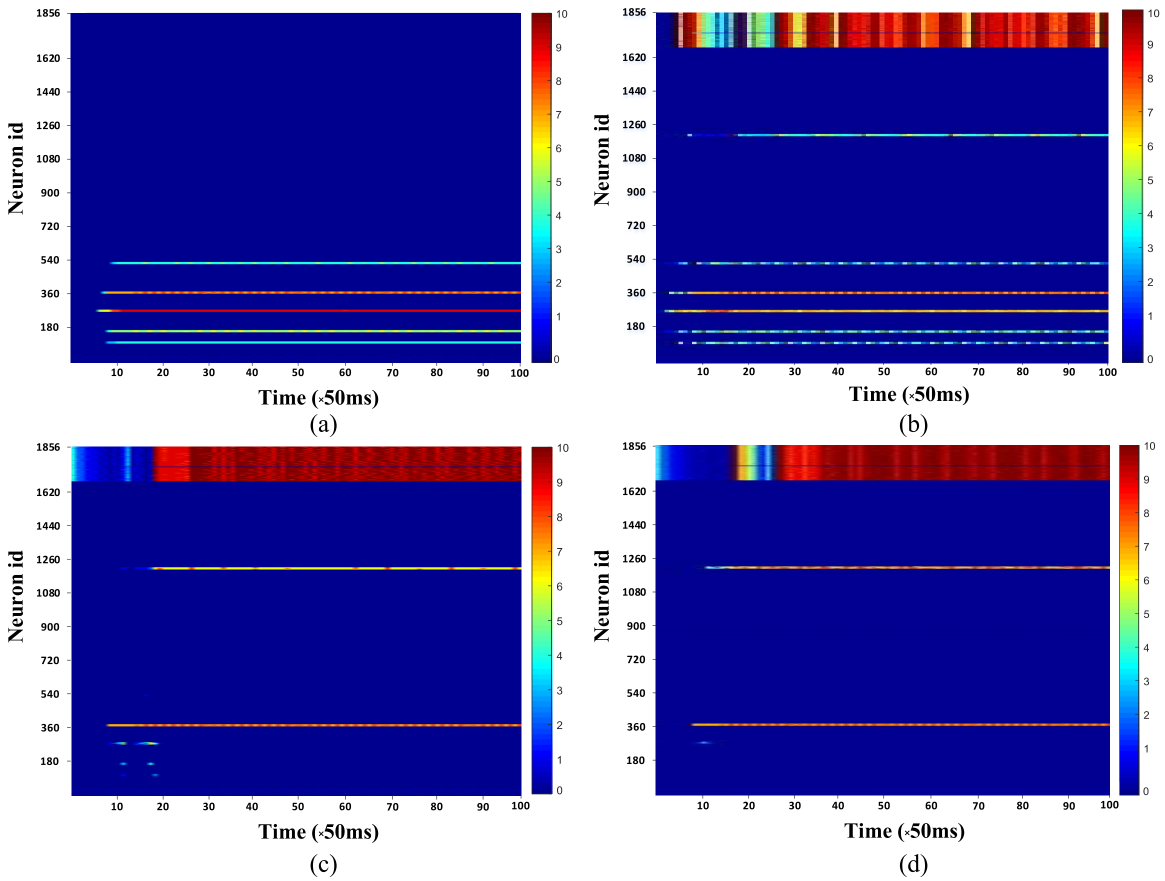


Fig. 7. Spiking activity of neurons is recorded for 5 s and is divided into 50-ms steps. (a) Attention signals are off. (b) Low-level attention to group 26. (c) High-level attention to group 26. (d) Very high-level attention to group 26.

TABLE IV
INDICES OF NEURONS IN THE SIMULATION

Neuron type	Abundance	Indices
Input cochlea	64	1-64
A1	768	65-832
Input attention	64	833-896
ENG	768	897-1664
ING	256	1665-1856

also a channel for attention, and with a peer-to-peer connection between the channels with the same frequencies, the two networks are coupled to each other. Each channel of attention consists of an input neuron that receives its I_{ext} from a high-level cognitive area. Each ENG group consists of 12 excitatory neurons. The INGs are then located in the next layer, which causes coupling between the two networks of the A1 and the attention. ING neurons inhibit both their own channel ENG and A1 neurons which are located at the same frequency channel. Attention input neurons, like cochlea input neurons, receive their I_{ext} through a serial port, except that the header intended for them is the hexadecimal value "4B4F." In this implementation, the attention input neurons indices are 833–896. These values for ENG neurons are 897–1664, and for ING neurons are 1665–1856. For the sake of clarity, Table IV describes the types of the neurons and their indices in the simulation.

The test scenario of the proposed network is as follows: five sound signals with different frequencies and intensities are given to the input neurons of five channels. Channels 4, 9, 18, 26, and 39 were randomly chosen as sound receivers, and the input neurons of these channels received 200, 250, 600, 400, and 200 pA, respectively. These numbers indicate that channel 18 is receiving louder sounds. The target channel in this test is channel 26. It is expected that the functional algorithm of attention swiftly changes the RF to facilitate sound separation in A1 by filtering out the unattended acoustic signals. The state of all neurons was recorded for 5 s to display the FPGA implementation results. For each neuron, the total number of spikes per 50 ms was accumulated, so 100 intervals were obtained. Fig. 7(a) shows the state that the input neurons from the cochlea for channels 4, 9, 18, 26, and 39 receive currents of the aforementioned values, and no channels of attention are active.

After this step, channel 26 is identified as the target frequency band and the attention network related to this channel is activated. Neuron 858 is the input neuron of the 26th channel of attention that will receive external current. This current is proportional to the amount of attention a person pays to this frequency band. For example, we injected three currents of 100, 300, and 500 pA into these neurons, indicating low, high, and very high levels of attention, respectively. The results of injecting these three values of I_{ext} are shown in Fig. 7(b)–(d),

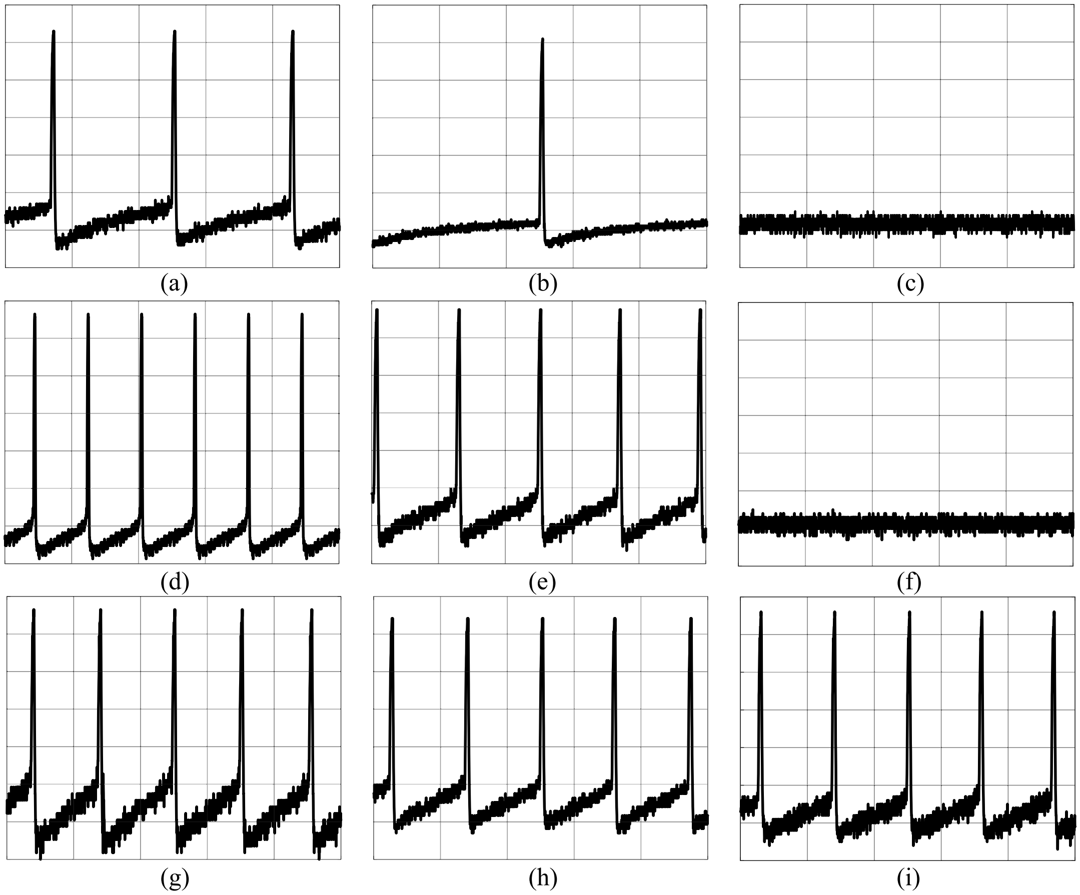


Fig. 8. Membrane voltage of three neurons observed using an oscilloscope. The membrane voltage of neuron with indices of (a)–(c) 110, (d)–(f) 275, and (g)–(i) 370 in the absence of attention, in low level of attention, and in very high level of attention.

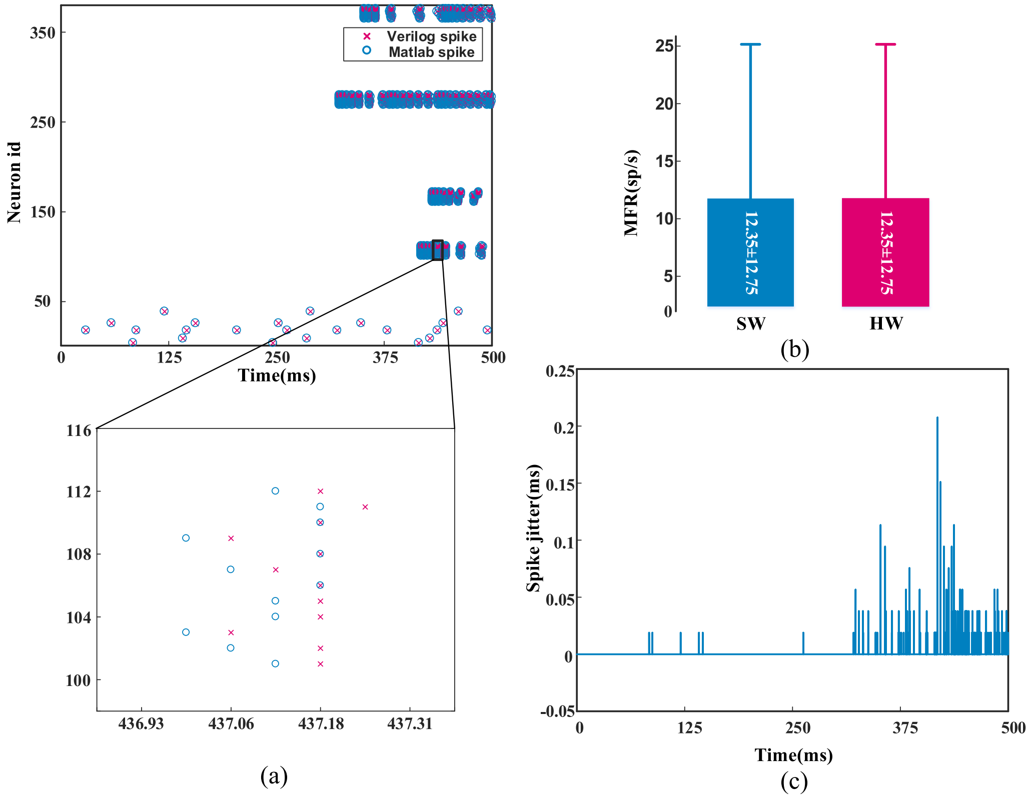
respectively. As can be seen, in all cases, neurons of channel 26 in A1 are transmitted to higher areas at approximately the same firing rate, while by increasing attention levels the intensity of suppression will increase and eliminate more annoying noises.

For better visual understanding, the membrane voltage of some neurons is shown in Fig. 8, where an oscilloscope is used to obtain an analog voltage. A 16-bit, 4-channel digital-to-analog converter (DAC) with part number AD5754 from Analog Devices is employed to generate analog waveforms. The membrane voltages of the neurons with indices of 110 from the 4th channel, 275 from the 18th channel, and 370 from the 26th channel were sent to DAC, and their output were captured in different modes of attention. Fig. 8(a)–(c) shows the membrane voltage of neuron 110, in the absence of attention, the presence of low-level attention to channel 26, and, very high attention to channel 26, respectively. The same scenario is repeated in Fig. 8(d)–(f) for neuron 275, while Fig. 8(g)–(i) for neuron 370 (in the attended group) keeps its firing rate in all states of attention.

To evaluate the precision of the digital primary auditory cortex with the attention network, the hardware implementation result is compared with the MATLAB equivalent. For this purpose, we consider a network like Fig. 7(a) in which the attention network is off. Fig. 9(a) compares the raster plots of the first 376 neurons in the network that emulates for 500 ms. The zoomed-in view of this figure shows a good adherence between hardware and software in terms of the spike timing. For more quantitative comparison, we investigated the raster plots from the mean firing rate (MFR) and jitter criteria point of views.

Regarding the MFR, as clearly visible in the bar graphs of Fig. 9(b), both hardware and software statistically represent the same number of spikes. In order to examine the jitter values for active neurons, we plotted a histogram in Fig. 9(c) showing the average time differences between the spikes in hardware compared to software in each iteration (the width of each column is equal to 1/16 ms). The network has a jitter less than 0.2 ms in all simulation time, verifying proper performance of the hardware implementation in terms of accuracy.

To provide a more comprehensive perspective of performance and resource utilization, we have summarized hardware costs and timing report in Table V. We defined the architecture of Fig. 6(a) as a package, and with minor modifications, we were able to implement a network of 30 packages on this FPGA, which contains about 56 000 neurons. In this case, approximately 95% of the available LUTs were used. The timing analysis of post-place and route (PAR) indicates that the maximum frequency of the proposed network is 210.70 MHz for 5-stages pipeline architecture. The status of all neurons is updated in 1856 clock cycles, which means each iteration takes



(a)

(b)

(c)

Fig. 9. Comparison of software and hardware results. (a) Raster plot of the first 376 neurons in 500 ms for both hardware and software. (b) MFR in both simulations in the presence of the bar graphs. (c) Average time difference between spikes for (a) in every 1/16 ms.

TABLE V
POST PLACE AND ROUTE REPORT

Neurons	**1856**	**56k**	
	Used%	**Used%**	**Available**
LUT	4	94.6	134600
Register	2	54.1	269200
DSP	3.2	68.8	740
BRAM	2.2	66.3	365
Max frq(MHz)	210.70	65.32	

about 8.83 us (1856/210 MHz) to update all the neurons. This time is even much faster than the real-time operation (less than $1/16\,\text{ms} = 62.5\,\text{us}$). For the maximum number of neurons on the chip (56 000 neurons), the maximum frequency is 65.32 MHz, which is again in the real-time range (28.41 us). Therefore, the proposed architecture guarantees real-time performance for the maximum number of neurons that can be placed in the FPGA.

Table VI compares the presented work with state-of-the-art works in terms of the digital hardware implementation of SNN networks. A fixed-point implementation of the IZ neuron model is introduced in [35]. They are able to embed up to 1440 neurons in a fully connected network in a Xilinx Virtex6 FPGA. The lack of resource sharing has severely limited the number of neurons in comparison with the proposed work. In [15], a real-time scalable FPGA implementation of hippocampal SNN with 10 000 IZ neurons is fulfilled, in which neurons are organized into four excitatory groups and one inhibitory group and are connected randomly together. Their network consumes 2 MB of BRAM for 10 000 neurons, which consumes far more memory resources than our work. Other parameters of logical resources and clock frequency are not explicitly reported. The time interval between the iterations is 0.77 ms, which is larger than our time resolution, and it is more difficult for our network to work in real-time performance compared to them.

In [7], a scalable digital hardware accelerator for simulating 4500 IZ neuron is presented. In the proposed network, more efficient hardware programming has reduced resource consumption by 2.8 times compared with them. Also, a much better search for critical paths and finding optimal pipeline locations to break these paths has increased our the clock speed by more than four times.

The network designed in [37] consists of Izhikevich neurons implemented on FPGA to demonstrate competitive Hebbian learning. They did not use any DSP resources, but LUT consumption is inevitably increased. While they are able to implement 110 neurons at a cost of 46k LUTs and have no reference to real-time network performance, the high-performance hardware architecture design allows us to consume 4.6k LUTs for 1860 neurons in real time.

Stimberg et al. [12] proposed a digital hardware architecture for spiking FORCE with some modifications to the original method. They implement a network of 510 Izhikevic neurons on an FPGA with floating-point numbers. The author utilizes customized floating-point (24- and 18-bit) cores for the arithmetic functions. Although the high accuracy of the floating point number system results in reduced hardware jitter

TABLE VI
COMPARISON OF THE PROPOSED IMPLEMENTATION WITH SIMILAR HARDWARE IMPLEMENTATIONS OF IZ NEURONS

Work	FPGA	Num neurons	Max frq	LUT	DSP	Real-time	NS
[35]	Virtex6	1440	100 MHz	56k	408	Yes	144
[15]	Stratix3	10k	-	-	-	Yes	-
[7]	Artix7	4500	52 MHz	30k	36	Yes	234
[36]	Spartan6	110	183 MHz	46.6k	0	-	20.13
[12]	Artix7	510	167MHz	11.1k	99	Yes	85.17
This work	Artix7	56k	65.32 MHz	134k	92	Yes	3657

(which is essential for systems with precise spike timing), design complexity and intensive resource consumption are serious obstacles in choosing this type of computation. While their number of neurons is one-third of our network (compared to 1860 neurons), our LUT and DSP block consumption are one-half and one-fourth, respectively.

For a more quantitative comparison between the presented works, we define an NS parameter which is derived from

$$\frac{(\text{Number of Neurons}) * (\text{Clock Frequency})}{10^{-9}}. \qquad (10)$$

The larger NS means that the state of more neurons is updated at a higher frequency. As the last column of Table IV shows, the proposed work can provide a better hardware platform for a high-speed large-scale network of IZ neurons compared to previous works. Scalability, resource sharing, the use of optimal pipeline stages to improve timing performance, less hardware consumption, and memory management are some of the advantages of our work.

VII. CONCLUSION

In this study, we developed a real-time scalable FPGA accelerator for cortical AA. The main goal was to provide an embedded implementation of the cocktail party effect by simplifying the auditory cortex networks and proposing an SWTA network as the RF in the attention network. The results confirmed that the presented AA was able to reproduce relevant functional dynamics and is a suitable case for digital-based AA. We provided an innovative structure for the SWTA network that is made up of an HF IZ neuron model and analyzed the network in detail from various aspects, such as stability and switching time. Hardware implementation results demonstrated that the optimization techniques, such as how to store data, resource sharing, and the pipeline structure, considerably enhanced the performance in comparison with the state-of-the-art biologically hardware implementations. While here 1856 neurons were placed on the FPGA, reconfigurability of the platform, along with generic programming of the parameters allow designers to easily implement up to 56 000 neurons in real time.

REFERENCES

[1] E. M. Kaya and M. Elhilali, "Modelling auditory attention," *Biol. Sci.*, vol. 372, no. 1714, 2017, Art. no. 20160101. [Online]. Available: https://doi.org/10.1098/rstb.2016.0101

[2] S. Evans, Z. K. Agnew, S. Rosen, and S. K. Scott, "Getting the cocktail party started: Masking effects in speech perception," *Cogn. Neurosci.*, vol. 28, pp. 483–500, Mar. 2016.

[3] R. K. Singh, Y. Xu, R. Wang, T. J. Hamilton, S. L. Denham, and A. van Schaik, "CAR-lite: A multi-rate cochlear model on FPGA for spike-based sound encoding," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 5, pp. 1805–1817, May 2019.

[4] M. F. Bear, B. W. Connors, and M. A. Paradiso, *Neuroscience: Exploring the Brain*. Philadelphia, PA, USA: Wolters Kluwer, 2016.

[5] I. P. Jaaskelainen and J. Ahveninen, "Auditory-cortex short-term plasticity induced by selective attention," *Neural Plast.*, vol. 2014, Jan. 2014, Art. no. 216731. [Online]. Available: https://doi.org/10.1155/2014/216731

[6] J. Shamsi, K. Mohammadi, and S. B. Shokouhi, "A hardware architecture for columnar-organized memory based on CMOS neuron and memristor crossbar arrays," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 12, pp. 2795–2805, Dec. 2018.

[7] B. Abdoli and S. Safari, "A reconfigurable real-time neuromorphic hardware for spiking winner-take-all network," *Int. J. Circuit Theory Appl.*, vol. 48, no. 12, pp. 2141–2152, 2020. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cta.2877

[8] R. Kreiser, D. Aathmani, N. Qiao, G. Indiveri, and Y. Sandamirskaya, "Organizing sequential memory in a neuromorphic device using dynamic neural fields," *Front. Neurosci.*, vol. 12, p. 717, Nov. 2018. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2018.00717

[9] Y. Chen, "Mechanisms of winner-take-all and group selection in neuronal spiking networks," *Front. Comput. Neurosci.*, vol. 11, p. 20, Apr. 2017. [Online]. Available: https://www.frontiersin.org/article/10.3389/fncom.2017.00020

[10] V. A. Filippov, A. N. Bobylev, A. N. Busygin, A. D. Pisarev, and S. Y. Udovichenko, "A biomorphic neuron model and principles of designing a neural network with memristor synapses for a biomorphic neuroprocessor," *Neural Comput. Appl.*, vol. 32, p. 15, Apr. 2020. [Online]. Available: https://doi.org/10.1007/s00521-019-04383-7

[11] K. Minkovich, C. M. Thibeault, M. J. O'Brien, A. Nogin, Y. Cho, and N. Srinivasa, "HRLSim: A high performance spiking neural network simulator for GPGPU clusters," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 2, pp. 316–331, Feb. 2014.

[12] M. Stimberg, R. Brette, and D. Fm Goodman, "Brian 2, an intuitive and efficient neural simulator," *Elife*, vol. 8, Aug. 2019, Art. no. e47314.

[13] S. Li, Z. Zhang, R. Mao, J. Xiao, L. Chang, and J. Zhou, "A fast and energy-efficient snn processor with adaptive clock/event-driven computation scheme and online learning," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 68, no. 4, pp. 1543–1552, Apr. 2021.

[14] T.-S. Chou et al., "CARLSim 4: An open source library for large scale, biologically detailed spiking neural network simulation using heterogeneous clusters," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2018, pp. 1–8.

[15] S. Yang et al., "FPGA implementation of hippocampal spiking network and its real-time simulation on dynamical neuromodulation of oscillations," *Neurocomputing*, vol. 282, pp. 262–276, Mar. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231217318751

[16] S. Yang et al., "BiCoSS: Toward large-scale cognition brain with multigranular neuromorphic architecture," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 7, pp. 2801–2815, Jul. 2022.

[17] E. Jokar, H. Abolfathi, A. Ahmadi, and M. Ahmadi, "An efficient uniform-segmented neuron model for large-scale neuromorphic circuit design: Simulation and fpga synthesis results," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 6, pp. 2336–2349, Jun. 2019.

[18] S. Nazari, M. Amiri, K. Faez, and M. M. Van Hulle, "Information transmitted from bioinspired neuron-astrocyte network improves cortical spiking network's pattern recognition performance," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 2, pp. 464–474, Feb. 2020.

[19] 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.

[20] S. Y. Bonabi, H. Asgharian, S. Safari, and M. N. Ahmadabadi, "FPGA implementation of a biological neural network based on the Hodgkin-Huxley neuron model," *Front. Neurosci.*, vol. 8, p. 379, Nov. 2014. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2014.00379

[21] S. Yang et al., "Efficient hardware implementation of the subthalamic nucleus-external globus pallidus oscillation system and its dynamics investigation," *Neural Netw.*, vol. 94, pp. 220–238, Oct. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893608017301648

[22] K. Akbarzadeh-Sherbaf, B. Abdoli, S. Safari, and A.-H. Vahabie, "A scalable FPGA architecture for randomly connected networks of Hodgkin-Huxley neurons," *Front. Neurosci.*, vol. 12, p. 698, Oct. 2018. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2018.00698

[23] E. M. Izhikevich and G. M. Edelman, "Large-scale model of mammalian thalamocortical systems," *Proc. Nat. Acad. Sci. United States America*, vol. 105, no. 9, pp. 3593–3598, Mar. 2008.

[24] E. M. Izhikevich, "Which model to use for cortical spiking neurons?" *IEEE Trans. Neural Netw.*, vol. 15, no. 5, pp. 1063–1070, Sep. 2004.

[25] 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.

[26] S. Atiani et al., "Emergent selectivity for task-relevant stimuli in higher-order auditory cortex," *Neuron*, vol. 82, no. 2, pp. 486–499, 2014. [Online]. Available: https://doi.org/10.1016/j.neuron.2014.02.029

[27] D. Liang, R. Kreiser, C. Nielsen, N. Qiao, Y. Sandamirskaya, and G. Indiveri, "Neural state machines for robust learning and control of neuromorphic agents," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 9, no. 4, pp. 679–689, Dec. 2019.

[28] L. Whiteley and M. Sahani, "Attention in a Bayesian framework," *Front. Human Neurosci.*, vol. 6, p. 100, Jun. 2012. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnhum.2012.00100

[29] N. Mesgarani and E. F. Chang, "Selective cortical representation of attended speaker in multi-talker speech perception," *Nature*, vol. 485, pp. 233–236, Apr. 2012. [Online]. Available: https://doi.org/10.1038/nature11020

[30] S. Norman-Haignere, N. Kanwisher, and J. McDermott, "Distinct cortical pathways for music and speech revealed by hypothesis-free voxel decomposition," *Neuron*, vol. 88, no. 6, pp. 1281–1296, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0896627315010715

[31] S. Shamma and J. Fritz, "Adaptive auditory computations," *Current Opin. Neurobiol.*, vol. 25, pp. 164–168, Apr. 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0959438814000269

[32] J. B. Fritz, M. Elhilali, S. V. David, and S. A. Shamma, "Does attention play a role in dynamic receptive field adaptation to changing acoustic salience in A1?" *Hearing Res.*, vol. 229, no. 1, pp. 186–203, 2007.

[33] A. Jimenez-Fernandez et al., "A binaural neuromorphic auditory sensor for FPGA: A spike signal processing approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 4, pp. 804–818, Apr. 2017.

[34] Y. Xu, C. S. Thakur, R. K. Singh, T. J. Hamilton, R. M. Wang, and A. van Schaik, "A FPGA implementation of the CAR-FAC cochlear model," *Front. Neurosci.*, vol. 12, p. 198, Apr. 2018. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2018.00198

[35] D. Pani, P. Meloni, G. Tuveri, F. Palumbo, P. Massobrio, and L. Raffo, "An FPGA platform for real-time simulation of spiking neuronal networks," *Front. Neurosci.*, vol. 11, p. 90, Feb. 2017. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2017.00090

[36] M. Heidarpur, A. Ahmadi, M. Ahmadi, and M. R. 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.

[37] N. Salimi-Nezhad, E. Ilbeigi, M. Amiri, E. Falotico, and C. Laschi, "A digital hardware system for spiking network of tactile afferents," *Front. Neurosci.*, vol. 13, p. 1330, Jan. 2020. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2019.01330

Behrooz Abdolil received the B.S. degree in electronic engineering from Razi University, Kermanshah, Iran, in 2009, and the M.S. degree in electronic engineering from the Iran University of Science and Technology, Tehran, Iran, in 2012. He is currently pursuing the Ph.D. degree in digital systems engineering with the School of Electrical and Computer Engineering, University of Tehran, Tehran.

His research interests include bioinspired neural networks, hardware implementation of neural networks, neuromorphic engineering, and computer architecture.

Saeed Safari received the Ph.D. degree in computer architecture from the Computer Engineering Department, Sharif University of Technology, Tehran, Iran, in 2005.

Since then, he has been a Faculty Member with the Electrical and Computer Engineering Department, University of Tehran, Tehran. From May 2009 to September 2010, he collaborated with TeleRobotics and Applications Laboratory, IIT, Genoa, Italy, working on different aspects of low-power parallel implementation of machine vision applications. His research interests are artificial intelligence, high-performance computing, computer architecture, and computer arithmetic.