A spiking neural network for real-time Spanish vowel phonemes recognition

L. Miró-Amarante, F. Gómez-Rodríguez, A. Jiménez-Fernández, G. Jiménez-Moreno

Robotics and Technology of Computers Lab, ETS Ingeniería Informática, University of Seville, Av. Reina Mercedes s/n, Seville, Spain

Keywords:
Neuromorphic engineering
Address event representation (AER)
Event-based processing
FPGA
Digital cochlea
Speech recognition

A B S T R A C T

This paper explores neuromorphic approach capabilities applied to real-time speech processing. A spiking recognition neural network composed of three types of neurons is proposed. These neurons are based on an integrative and fire model and are capable of recognizing auditory frequency patterns, such as vowel phonemes; words are recognized as sequences of vowel phonemes. For demonstrating real-time operation, a complete spiking recognition neural network has been described in VHDL for detecting certain Spanish words, and it has been tested in a FPGA platform. This is a stand-alone and fully hardware system that allows to embed it in a mobile system. To stimulate the network, a spiking digital-filter-based cochlea has been implemented in VHDL. In the implementation, an Address Event Representation (AER) is used for transmitting information between neurons.

1. Introduction

Nature is a notable source of inspiration for researchers around the world. Neuromorphic engineering attempts to integrate biological knowledge into artificial systems [1,2]. This work combines both psychoacoustic knowledge and neuromorphic engineering principles to propose a spiking neural network for Spanish vowel phonemes recognition.

This paper is organized as follows: a brief introduction to Spanish psychoacoustic and neuromorphic processing is presented in the remainder of this section; Section 2 is devoted to describing the spiking digital-filter-based cochlea; Section 3 presents the spiking recognition neural network; the experiments and results are presented in Section 4.

1.1. Spanish psychoacoustic

The biological process of Spanish phoneme recognition is based on the first two formants of the vowel phoneme spectrum. Formants are the resonant frequencies of a vocal tract when pronouncing a voiced sound such as a vowel [3].

There is an intimate relationship between the acoustic properties of speech sounds and their source, the human organs of speech. The acoustic structure of speech is completely predictable (on physical principles) from the position and states of the organs of production. Because vowels are normally voiced, the source (known as the glottal source) is situated in the larynx when the sound arises from the quasi-periodic vibrations of the vocal folds. This near-periodicity of the glottal waveform gives rise to an essentially harmonic spectrum with the energy at the frequency of the vocal fold vibrations and its integer multiples. Hence, as the vocal fold vibration rate increases the frequency of the fundamental component in the spectrum and the inter-harmonic spacing also increase. As it decreases, the harmonics become more densely packed with the fundamental component decreasing in frequency.

Before the sound from the glottal source can reach a listener, it must pass through the vocal tract. For non-nasalized vowels, the vocal tract has no anti-resonances; it only contains resonances. These resonances are called formants, and only the lowest four or five formants are important in speech perception. The formant resonances of the vocal tract give rise (as long energy is present from the source at that frequency region) to peaks in the output spectrum, also known as formants. Numbers increasing from 1 usually identify formants, starting from the formant at the lowest frequency. The spectrum of phonemes can consist of several formants, but the first two formants are the most important for recognition [4].

During the pronunciation of Spanish vowels, in an adult male, the frequency of the first formant (F1) can vary in range from 200 Hz to 1000 Hz. The lower value occurs when the tongue is closer to the roof of the mouth. The vowel /i/ has one of the lowest F1 values – approximately 250 Hz; in contrast, the vowel /a/ has the highest F1 value at approximately 916 Hz. The second formant (F2) can vary from 600 Hz to 2500 Hz; the F2 value is proportional to the front or back of the highest part of the tongue during the production of the vowel. Additionally, lip rounding causes a lower F2 than unrounded lips. /i/ has an F2 of 2400 Hz, the highest F2 of any vowel. In the production of this vowel, the tip of the tongue is quite far forward, and the lips are

Table 1
The dispersion field of F1 of Spanish vowel phonemes in an adult male.

F1	/i/	/e/	/a/	/o/	/u/
Lower limit (Hz)	200	400	800	325	200
Upper limit (Hz)	450	800	1000	800	400
80% field (Hz)	290/330	580/750	915	750	210/330

Table 2
The dispersion field of F2 of Spanish vowel phonemes in an adult male.

F2	/i/	/e/	/a/	/o/	/u/
Lower limit (Hz)	1300	800	1000	700	600
Upper limit (Hz)	2500	2500	1800	2000	2500
80% field (Hz)	2300/2400	1600/1900	1500	950/1050	625/660

unrounded. At the opposite extreme, /u/ has an F2 of 624 Hz; in this vowel, the tip of the tongue is very far back, and the lips are rounded.

The experimental studies conclude that there is a wider frequency range when a listener perceives Spanish vowel phonemes. Tables 1 and 2 show the lower and upper limits as well as the range corresponding to the narrow zone (where the phoneme is correctly identified by 80% of the listeners) for F1 and F2, respectively [5].

1.2. Neuromorphic processing

In 1990, Caver Mead started using the term, neuromorphic engineering [6]; neuromorphic engineering is a field of engineering devoted to neural system design and fabrication, in which architecture and design principles are based in biological neural systems. Different from artificial neural networks, neuromorphic systems attempt to imitate the function of biological neural system such as speech [7,8–10], object recognition [11] or cognitive processes [12–15].

One of the most important neuromorphic developments is address event representation (AER) [16,17]. AER is a communication scheme that was initially proposed for neuromorphic chip communication; today, it is in the basis of several neuromorphic systems [18–20].

AER provides a solution to one of the most difficult questions that neuromorphic engineering has to solve, which is how to imitate the neural-biological connectivity, where each single neuron can connect with tens of thousands of other neurons.

Fig. 1a explains the principle behind AER. The emitter chip contains an array of cells (e.g., neuromorphic cochlea [21]) where each cells shows a state that changes with a slow time constant (on the order of milliseconds). Each cell includes an oscillator that generates pulses of minimum width (a few nanoseconds). Each time a cell generates a pulse (an "event"), it communicates with the periphery, and its address is placed on the external digital bus (the AER bus). Handshaking lines (Acknowledge and Request) are used for completing the communication (Fig. 1b).

An advantage of AER scheme is that computation is event driven and thus can be very fast and efficient. Both the spiking recognition neural network and the spiking digital-filter-based cochlea proposed in this work use AER for representing and transmitting information.

1.3. Proposed spiking neural network overview

The proposed spiking neural network comprises two stages that model the different processes of the human auditory system, from the auditory periphery to the cells in the auditory brainstem. Fig. 2 shows an overview of the proposed spiking neural network.

First, speech is passed through a digital-filter-based cochlea where the speech is split into its frequency components and translated into spikes.

The biological cochlea has the capability of performing a frequency analysis of the incoming vibrations, separating them into overlapping frequency bands [22–24]. The mechanical motion of the basilar membrane leads to the displacement of the inner hair cells. The inner hair cells act as transducers; a neurotransmitter is released at the synaptic cleft, and action potentials are propagated into the auditory brainstem.

The frequency-selective displacement characteristic of the basilar membrane is modelled using an array of 21 band-pass digital filters and a set of spike generators. The band-pass filters, whose cut-off frequencies range over the human speech frequencies (20 Hz to 20 kHz), split the speech signal into different frequency components. Each filter output is connected to a spike generator that models the spike generation. The pair of the band-pass filter and the spike generator is denominated by the Cochlea Band (Section 2).

Secondly, a spiking recognition neural network is used for detecting phonemes and words as sequences of phonemes. This network is composed of several recognition-neurons (RNeuron). RNeurons spike when a set of their inputs spike simultaneously and when a pre-set pattern is detected in their inputs. Because some RNeurons can spike simultaneously, a winner-take-all network, composed by some winner-neurons (WTANeuron), is used for empowering the most active RNeurons and depressing the rest. A third spiking neuron, the delay neuron (DelayNeuron), is needed to detect words such as sequences of phonemes. All of these spiking neuron models are defined in this work in Section 3.

1.4. Hardware platform

A library of IP cores for speech information spiking processing using VHDL were developed for this research [25]. The system is implemented on a Cyclone IV E FPGA [26]. The FPGA chip is on a board, a EP4CE115F29C7 that contains an audio codec, WM8731/WM8731L [27].

A USB2AERmini2 board [28] is also used to communicate between the FPGA chip and the PC. This board was used as an AER monitor to test the output of each stage. The monitoring function involves reading and time-stamping events from AER bus and making them available to

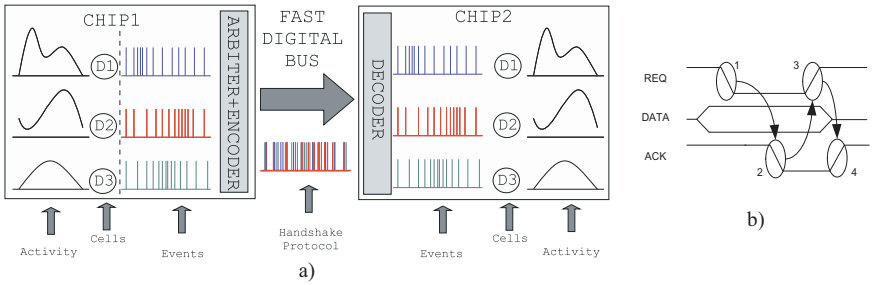


Fig. 1. Address event representation (AER).

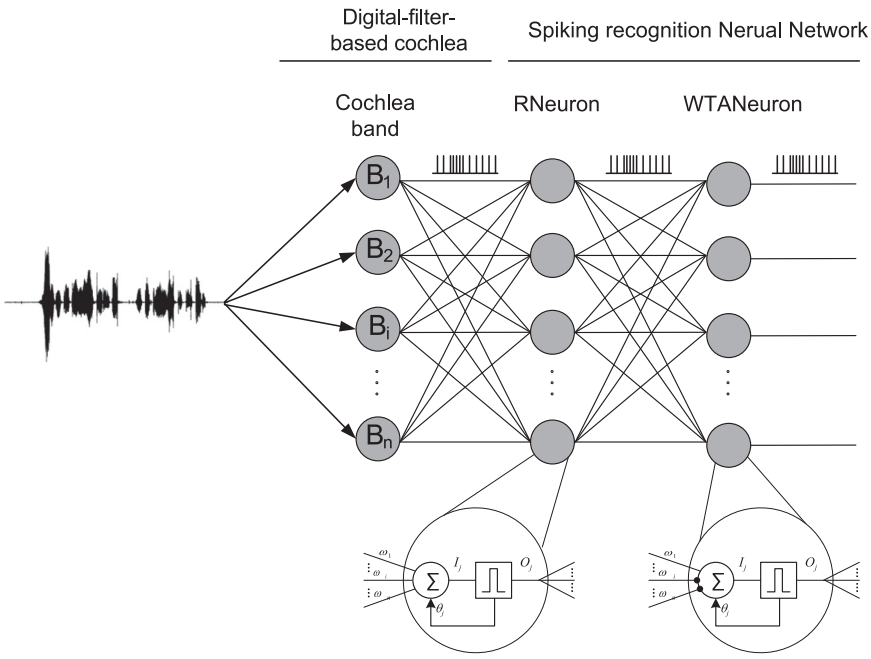


Fig. 2. Spiking recognition neural network overview.

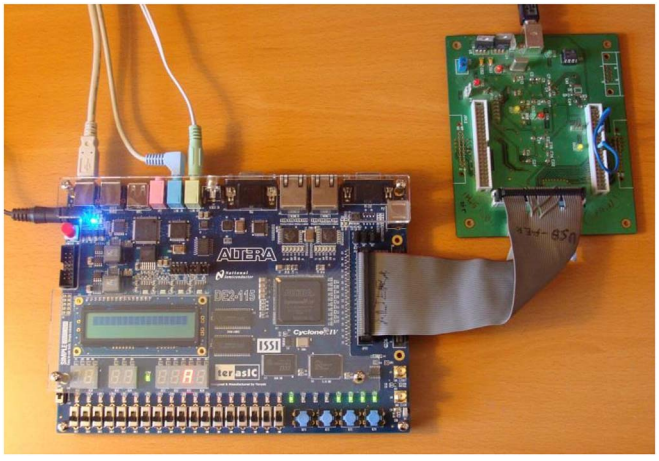


Fig. 3. EP4CE115F29C7 platform and USBAERmini2 board.

the PC. Fig. 3 shows the hardware platform used; a USB2AERmini2 board is on the right upper corner.

2. Spiking digital-filter-based cochlea

The spiking cochlea is composed of 21 bands (the digital filter and the spike generator). The digital cochlea works with an 18-bit PCM signal obtained from the ADC of the WM8731/WM8731L audio codec. The ADC is set at a 48 kHz sample rate.

2.1. Digital filters bank

Because the biological cochlea acts as a spectrum analyser, band-pass filters are used for decomposing complex sound into its frequency components. The band-pass filters show if a signal contains a frequency component within a specific frequency range.

A large number of options are available for filter design and implementation. However, the selected option allows low cost and low complexity in its hardware.

The digital filter bank consists of 21 infinite impulse response (IIR)

Table 3
Subdivision of the audible frequency range into critical bands.

Bands	Centre frequency (Hz)	Frequency range of band-pass filter (Hz)
0	350	200–400
1	450	400–510
2	570	510–630
3	700	630–770
4	840	770–920
5	1000	920–1080
6	1170	1080–1270
7	1370	1270–1480
8	1600	1480–1720
9	1850	1720–2000
10	2150	2000–2320
11	2500	2320–2700
12	2900	2700–3150
13	3400	3150–3700
14	4000	3700–4400
15	4800	4400–5300
16	5800	5300–6400
17	7000	6400–7700
18	8500	7700–9500
19	10500	9500–12000
20	13500	12000–15500

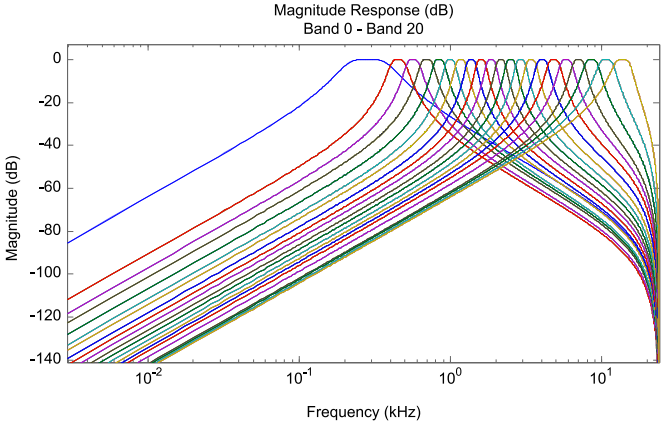
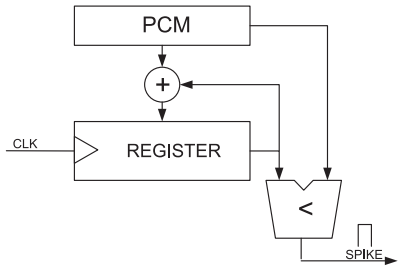


Fig. 4. Magnitude response of the 21 band-pass filters (Bode diagram).



$$threshold\ level = \frac{f_m \times 2^n}{f_{CLK}}$$

f_m 48 KHz, sample rate;
n 18 bits accumulator register;
f_{CLK} 100 MHz.

Fig. 5. Accumulator spiking neuron model.

second order filter sections using parallel form. Each of these filters is set to have a cut-off frequency in the range of 20–20 kHz, as shown in Table 3. The digital filter bank is based on the subdivision of the audible frequency range into critical bands proposed by Zwicker [29].

These 21 band-pass filters are all IIR filters. The FIR filter provides the linear phase; however, cochlear filters do not need a linear phase. Therefore, IIR filters are preferred to FIR filters. Also IIR filters can achieve a given filtering characteristic using less memory and calculations than the similar FIR filters.

Butterworth filters are used in this work. In these filters, the magnitude response is maximally flat in the pass-band and monotonic overall. The filter models are all designed in two's complement fixed-point arithmetic that implies the need for the direct form II transpose structure. All filters are described in VHDL language using the Filter Design HDL Coder [30]. Fig. 4 shows the magnitude response of the 21 band-pass filters.

2.2. Spikes generator

Each digital filter is connected to a spike generator that mimics the functionality of the inner hair cells (IHCs). This spike generator is based on the accumulator spiking neuron model proposed by Gómez Rodríguez et al. [31].

This spiking neuron model (shown in Fig. 5) uses an accumulator register, initialized with a threshold level. This value is increased in the PCM-filtered value every clock cycle. Concurrently, the output from the digital filter, the PCM-filtered value, is compared with the value of the register; if the register value is higher than the PCM-filtered value, a spikes will be generated. The number of spikes and the firing rate of each cochlea band are proportional to the amplitude of the corresponding digital filter output. So, the firing rate changes as the PCM filtered signal changes.

Although, each of the spike generators has its own threshold level that can be set independently, all of the threshold values are set to 127 in this implementation. This value guarantees that a sufficient number of spikes will be generated.

2.3. AER generator and arbiter

As previously mentioned, AER is used for coding and transmitting spikes. So, an AER arbiter encodes the address in each cochlea band from 0 to 20 and sends the AER spike on the time-multiplexed AER bus using a hand-shaking protocol. In this case, each AER spike is identified with an address in range from 0 to 20. Thus, the output of this spiking digital cochlea contains the AER spikes train associated to each cochlea band as an answer to the PCM signal.

2.4. Spiking digital cochlea response to acoustic stimuli

Matlab and the USBAERmini2 board are used to test the behaviour of the spiking digital cochlea. Fig. 6 shows the output histogram of the spiking digital cochlea in response to the swept-sine wave from 200 Hz to 1200 Hz. The maximum number of events of each cochlea band is analysed for each of the sine waves. Fig. 6 shows the most active cochlea band, which is the band whose central frequency matches with the frequency of the sine wave. The maximum number of events of each cochlea band is different because the spike generators do not make a homogeneous distribution of spikes; the number of events depends on the PCM value.

To illustrate the behaviour of the spiking digital cochlea, Fig. 7 shows the cochlear response to acoustic stimuli, a 689 Hz sine wave. The spiking cochleogram (left-hand column) displays the temporal distribution of spikes in the cochlea bands domain, and the histogram (right-hand column) displays the number of spikes per cochlea band.

Like a spectrum analyser, the bandpass filters separate sound into its frequency components. The spiking cochleogram display shows the output of the digital cochlea from a 689 Hz sine wave; each cochlea band fires a sequence of spike trains. The output of the third cochlea band, which ranges from 630 Hz to 770 Hz, indicates that the signal contains a frequency component within this specific frequency range. The histogram determines that the third cochlea band fires the largest number of events and is the most active cochlear band.

3. Spiking recognition neural network

As previously mentioned, the spiking recognition neural network is used to recognize phonemes and words as sequences of phonemes. Three types of neurons are needed to accomplish this recognition,

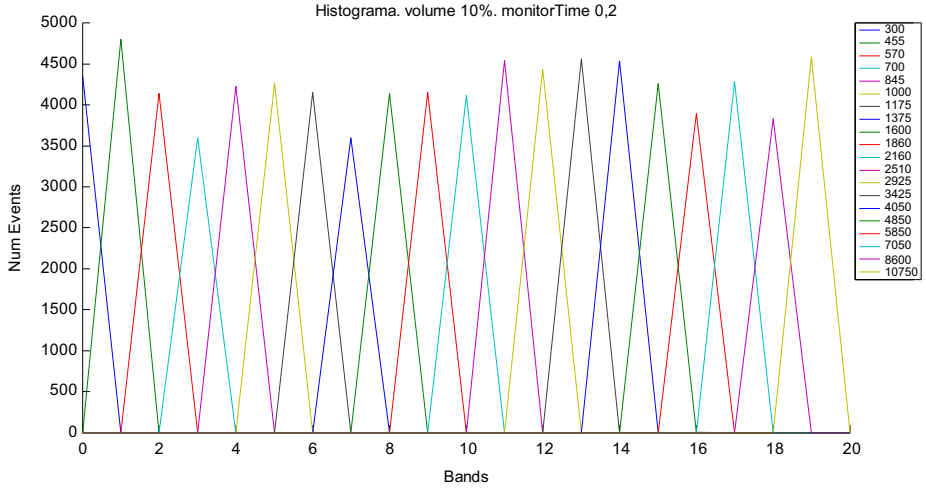


Fig. 6. Histogram plot. A 127-threshold level of the spike generator. The legend shows the frequency of the sine-signals.

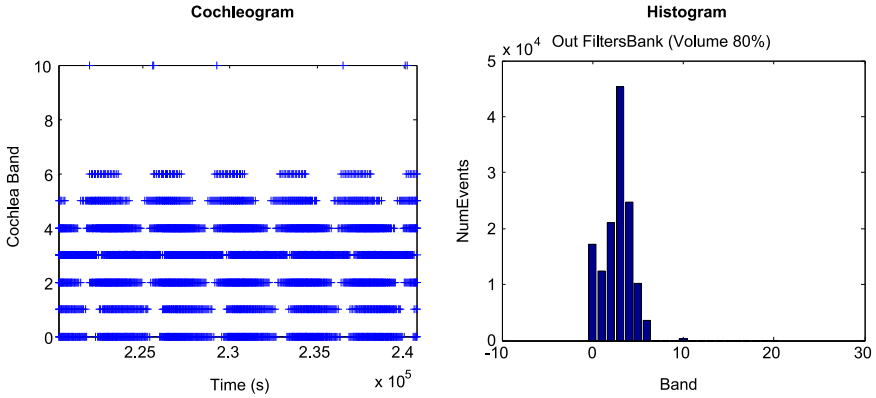


Fig. 7. Response of the spiking digital cochlea to a 689 Hz sine wave.

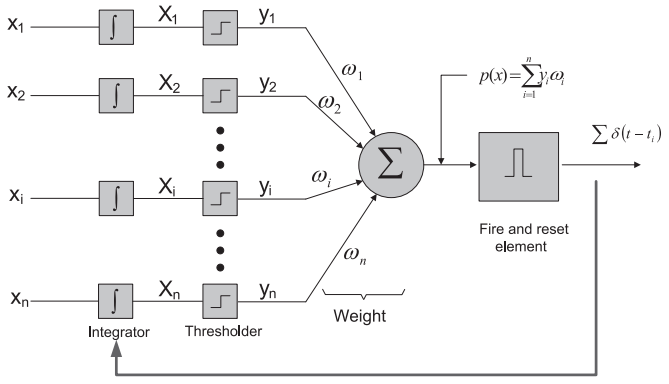


Fig. 8. RNeuron model.

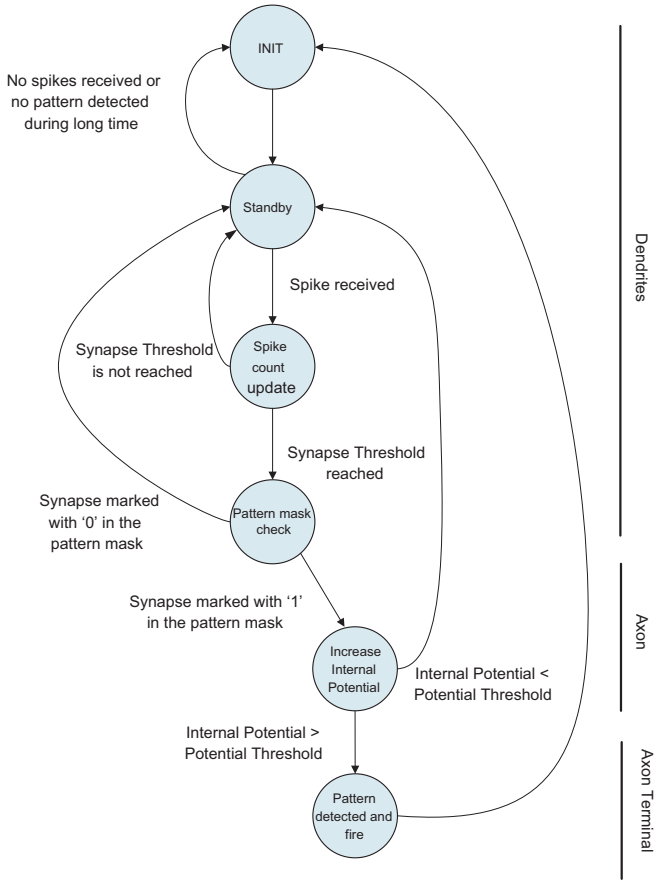


Fig. 9. The finite states machine of RNeurons.

RNeurons, WTANeruons and DelayNeurons.

3.1. RNeuron

The RNeuron is the main element of the recognition neural network. It has the function of identifying a pattern or category. Both phoneme recognition and word recognition are carried out by a set of this RNeuron. There is one RNeuron for each of the patterns to recognize. The behaviour of this neuron is based on artificial neural networks, particularly in the perceptron neuron and its function of pattern recognition, and the integrate-and-fire model.

RNeurons are derived from well-known neuron models, the perceptron [32] and integrate-and-fire models [33,34]. In these models, incoming spikes increase the internal cell potential depending on its input weight. RNeurons fire when the internal potential reaches a threshold. In this implementation, weights take a binary value.

Fig. 8 shows the proposed model. The RNeuron has n spiking inputs, $x_1, x_2,\ldots,x_n$. These spikes are integrated, $X_i = \int x_i$ when enough spikes are received,

$$y_i = \begin{cases} 0 & \text{if } X_i < \textit{Synapse threshold} \\ X_i & \text{if } X_i \geq \textit{Synapse Threshold} \end{cases}$$

These spikes may contribute to the internal potential, depending on its weight:

$$p(x) = \sum_{i=1}^{n} y_i \omega_i$$

The hardware implementation of RNeuron is based on a finite states machine, shown in Fig. 9, with the parameters shown in Table 4.

Fig. 9 shows the state machine diagram that models the behaviour of RNeuron. This figure indicates how the dendrites, axon and axon terminals of the neuron are implemented. The *init* state is the initial state after an asynchronous reset is received. In this state, the RNeuron parameters are initialized with the default values (Table 4). At the next cycle, it transitions to the *standby* state. In this state, if it receives a spike, then it transitions to the *spike count update* state; otherwise, it stays in the *standby* state. There is a timer that detects the absence of activity (Reset time parameter), and it goes back to the *init* state.

In the *Spike count update* state, the number of spikes received from each synapse is update in accordance with the current active synapses. If the number is higher than its synapse threshold, it transitions to the *pattern mask check* state; otherwise, it returns to the *standby* state.

In the *pattern mask check* state, if the synapse is marked with a '1' in the pattern mask, its internal potential is incremented in the *increase internal potential* state; otherwise, it returns to the *standby* state. In the *increase internal potential* state, if its potential threshold is reached, the pattern has been identified. In this case, it transitions to the *pattern detected and fire* state, and a spike is issued. Then, it returns to the *init* state; otherwise, it transitions to the *standby* state.

Table 4
RNeuron parameters.

Name	Parameter description
Pattern mask	Corresponds to the synapse weights. Array of n positions; position i will be '0' if the RNeuron does not expect spikes by the synapse i; '1' indicates that RNeuron expects spikes by this synapse. For example, if the pattern mask is "00000000000000001110", the RNeuron expects to receive spikes by the synapses 1, 2 and 3.
Synapses threshold	Each synapse has an associated a threshold level that indicates when a synapse is active and may contribute to the internal potential
Internal potential	Internal potential is increased when spikes are received by active synapses. This potential defines the behaviour of the RNeuron; while its value is lower than its potential threshold, the RNeuron receives and processes new spikes; when the potential threshold is reached, the RNeuron fires, indicating that a pattern was recognized. Then, its internal potential is reset.
Potential threshold	This threshold determines when RNeuron fires.
Reset time	If RNeuron does not receive a spike during a specified period of time (Reset time), all of internal values will be reset.

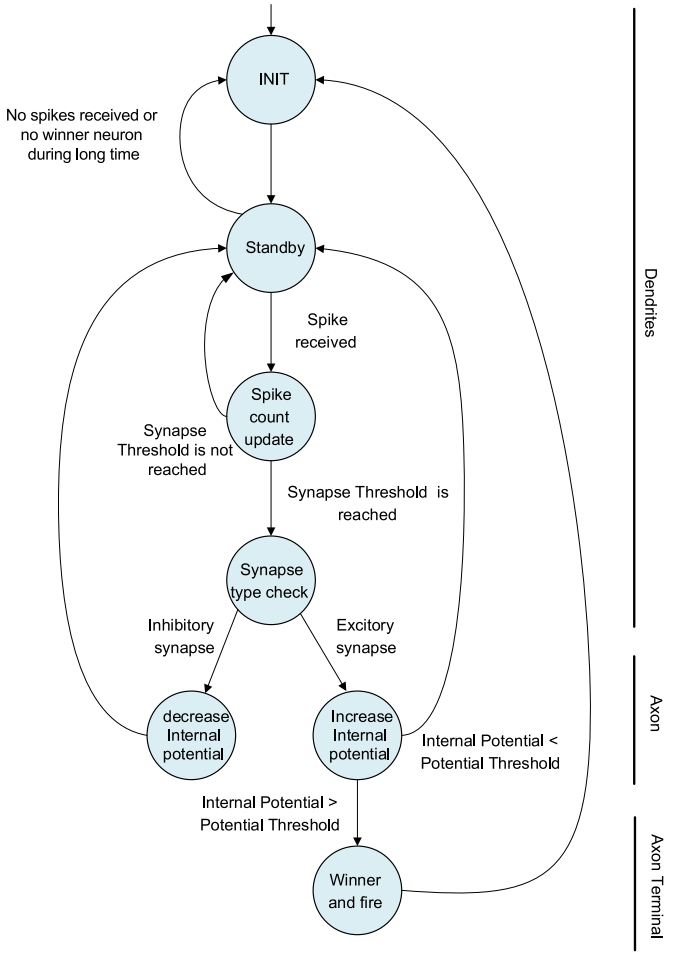


Fig. 10. Finite states machine of WTANeuron.

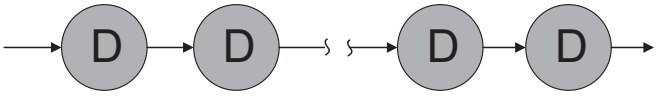


Fig. 11. DelayNeuron chain.

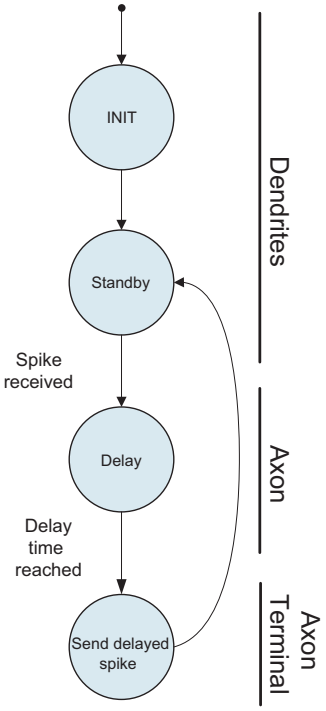


Fig. 12. DelayNeuron finite states machine.

Table 5
WTANeuron parameters.

Name	Parameter description
Synapse weights	Array of n positions that corresponds to the synapse weights. Each neuron has one excitatory synapse and $n-1$ inhibitory synapses. These weights are integers; positive values occur for the excitatory synapses and negative values for the inhibitory synapses.
Synapses threshold	Each synapse has an associated threshold level that indicates when a synapse is active.
Internal potential	When spikes are received by active synapses, the internal potential is increased or decreased depending of synapse weight. This potential defines the behaviour of the WTANeuron; when its value is lower than its potential threshold, the WTANeuron receives and processes new spikes; when the potential threshold is reached, the WTANeuron fires, indicating that it is the winner neuron. Then, its internal potential is reset.
Potential threshold	Each neuron has an associated level, which indicates that it is the winner neuron.
Reset time	If WTANeuron does not receive a spike during a certain period of time (Reset time), all of internal values will be reset.

Table 6
Vowel phonemes for the parameters of RNeuron.

	Id (AER spike address)	Pattern mask	Cochlea bands associated to formant frequencies (F1, F2)
RNeuron A	0	"00000000000001010000"	Bands 4 and 6
RNeuron E	1	"00000000001000000010"	Bands 1 and 9
RNeuron I	2	"00000010000000000001"	Bands 0 and 13
RNeuron O	3	"00000000000000010100"	Bands 2 and 4
RNeuron U	4	"00000000000000001001"	Bands 0 and 3

3.2. WTANeuron

It is possible that several RNeurons fire at the same time. For this reason, a winner-take-all network is needed. The winner-take-all spiking network has been used to enhance the recognition process. It is composed of a layer of spiking neurons, which receives spike trains as inputs. The neuron that receives spikes with the highest rate is selected as the winner after a pre-determined number of input spikes [35]. Although there are two winner-take-all modes, the hard and leaky modes, the leaky winner-take-all mode is chosen in this model, where the winning neuron is active and all other neurons are also active in less proportion.

There is one WTANeuron for each of the patterns to recognize. The WTANeurons inhibit each other while simultaneously activating themselves. Therefore, only one WTANeuron will be active, the one corresponding to the strongest input.

In this model, incoming spikes increase or decrease the internal neuron potential, depending on its input weight and its synapse type (excitatory or inhibitory). WTANeurons fire when the internal potential reaches a threshold. In this case, it is the winner neuron. In this implementation, weights take an integer value, positive value is taken for the excitatory synapses and a negative value for the inhibitory synapses.

The hardware implementation of WTANeurons is also based in a finite states machine, shown in Fig. 10, with the parameters shown in Table 5.

The following state machine diagram models the behaviour of a WTANeuron (Fig. 10). After an asynchronous reset, the WTANeuron begins in the *init* state with its setting parameters with default values (Table 5). At the next cycle, it transitions to the *standby* state. In that state, if it receives a spike, it transitions to the *spike count update* state; otherwise, it stays in the *standby* state. In the *standby* state, there is a timer that detects the absence of activity (the reset time parameter), and it returns to the *init* state.

In the *spike count update* state, the number of spikes received from each synapse is updated in accordance with the current active synapsis. If the number is higher than its synapse threshold, it transitions to the *synapse type check* state; otherwise, it returns to the *standby* state.

In the *synapse type check* state, it transitions to the *decrease internal potential* state or to the *increase internal potential* state according to its synapse weight. In the *decrease internal potential* state, its internal potential is decremented by the inhibitory weight, and it transitions to *standby* state; otherwise, its internal potential is incremented according to the excitatory weight. When its potential threshold is reached, it is the winner neuron. It transitions to the *winner and fire* state where a spike is issued, and it the transitions to the *init* state.

3.3. DelayNeuron

To recognize a word as a sequence of phonemes, it is necessary to delay the spike train associated with the first recognized phoneme. In this case, it ensures both phonemes are closely matched when they arrive to the next spiking recognition neural network (composed by RNeuron and WTANeuron tuned to detect the word).

DelayNeuron is devoted to the spike delay, and its behaviour is implemented with the finite state machine, shown in Fig. 12. DelayNeuron has only one parameter: DelayTime set to 204.8 μs. These experiments ensure that no spikes will be lost with this DelayTime value.

To obtain a larger delay time, several DelayNeuron can be connected in a chain (Fig. 11).

Fig. 12 shows the finite state machine diagram, which models the behaviour of a DelayNeuron. After an asynchronous reset, the DelayNeuron begins in the *init* state with its setting parameters having default values. At the next cycle, it transitions to the *standby* state. In that state, if it receives an incoming spike. Then it transitions to the *delay* state; otherwise, it stays in the *standby* state.

In the *delay* state, there is a timer that measures the delay time, and it goes to *send delayed spike* state where the spike is sent by the axon terminal.

4. Experiments and results

This section describes an experiment where the spiking recognition neural network is configured to categorize a Spanish two-syllable word in real-time from the identification of a sequence of two Spanish vowel phonemes. The Spanish word "letra" is used as an example. This word is considered as a sequence of two Spanish vowels phonemes, /e/ and /a/.

This process is performed in two stages: the first stage identifies the Spanish vowel phonemes, and the second stage identifies two-syllable words as a sequence of two vowel phonemes.

4.1. Vowel phonemes recognition neural network

First, the output of the spiking digital-filter-based cochlea, a spike train, arrives to the vowel phonemes recognition neural network.

This recognition neural network is composed of five RNeurons, one for each Spanish vowel phoneme. Every RNeuron is configured to identify one vowel phoneme (Table 6). As it is described in the previous section, each vowel phoneme is characterized with the F1 and F2 formant; it has been experimentally determined which cochlea band corresponds with each formant. Then, this information is used for setting the pattern mask of the RNeuron. All RNeurons have a reset time of 10 ms, and all of these values were obtained experimentally.

As a result, an AER spike train is obtained, where each AER spike

Table 7
Vowel phonemes for the parameters of WTANeuron.

	WTANeuron A	WTANeuron E	WTANeuron I	WTANeuron O	WTANeuron U
Id (AER spike address)	0	2	4	6	8
We	5	5	5	5	5
Wi	10	10	23	23	15
Potential threshold	16	16	31	16	16

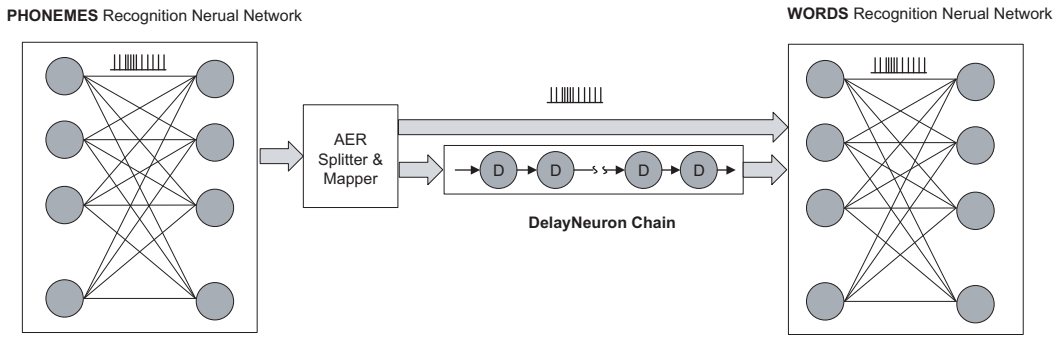


Fig. 13. Word recognition neural network.

Table 8
Words RNeurons parameters.

	Id (AER spike address)	Pattern mask	Address of the two vowel phonemes
RNeuron AE	0	"0000000110"	1 (a_{delay}), 2(e)
RNeuron AO	1	"0001000010"	1 (a_{delay}), 6(o)
RNeuron EA	2	"0000001001"	3 (e_{delay}), 0(a)
RNeuron EO	3	"0001001000"	3 (e_{delay}), 6(o)
RNeuron IA	4	"0000100001"	5 (i_{delay}), 0(a)
RNeuron IE	5	"0000100100"	5 (i_{delay}), 2(e)
RNeuron IO	6	"0001100000"	5 (i_{delay}), 6(o)
RNeuron OA	7	"0010000001"	7 (o_{delay}), 0(a)
RNeuron UA	8	"1000000001"	9 (u_{delay}), 0(a)
RNeuron UE	9	"1000000100"	9 (u_{delay}), 2(e)
RNeuron UI	10	"1000010000"	9 (u_{delay}), 4(i)

has an address from 0 to 4 that is associated with each RNeuron (0 for /a/, 1 for /e/, 2 for /i/, 3 for /o/ and 4 for /u/).

The recognition neural network is also composed of five WTANeurons (Table 7). Each WTANeuron has one excitatory synapse (for one of the five vowel phoneme) and four inhibitory synapses (for the other phonemes). The values of the excitatory weight (We), the inhibitory weight (Wi), the synapses threshold and the potential threshold have been obtained experimentally. All WTANeurons have a reset time of 10 ms.

As a result, an AER spike train is obtained, where each AER spike has an address associated with each WTANeuron (0 for /a/, 2 for /e/, 4 for /i/, 6 for /o/ and 8 for /u/).

4.2. Word recognition neural network

Once the vowel phonemes are identified, the words recognition neural network is able to recognize a two-syllable word as a sequence of two vowels phonemes. This recognition neural network consists of 11 RNeurons, which are set to simultaneously detect two vowel phonemes.

It is necessary to add some delays to the vowel phonemes to ensure that the two vowels phonemes, the first phoneme and the second phoneme, arrive to the RNeurons at the same time. So, it is necessary to delay the first phoneme.

Fig. 13 shows the entire two-syllable word recognition neural network. This network is comprised of three parts: the phonemes recognition neural network, the DelayNeuron chain and the words recognition neural network.

Because of the AER hardware implementation, additional elements are required including an AER_Splitter & Mapper, which replicates and sends each incoming spike through the two output. One spike will go directly to the words recognition neural network, and the other will be delayed. To distinguish between the original spike and the delayed spike, it also changes the addresses of the delayed spike. So, the original spikes have odd addresses (0(a), 2(e), 4(i), 6(o), 8(u)) and the delayed spikes have even addresses (1(a_{delay}), 3(e_{delay}), 5(i_{delay}), 7(o_{delay}), 9(u_{delay})).

In this implementation, the delay time is 200 ms (the experimental value). It is interesting to note that the recognition neural network can be adapted to the speed of the speaker with a change in this value.

The recognition neural network has as many Rneurons as words that it wants to recognize. The implementation identifies 11 words, which means that it recognizes 11 sequences of vowel phonemes (Table 8). The output of this module is also an AER spike train. Each AER spike will have a value that belongs to range 0 to (number of words − 1) associated with each word.

All RNeurons have a reset time of 5 ms. All these values are obtained experimentally.

In this recognition neural network, a set of WTANeurons is also added to improve the recognition process. Table 9 shows the setting parameters for the words WTANeurons. All WTANeurons have a reset time of 2.5 ms.

Fig. 14 shows the full recognition process for the speech signal corresponding to the Spanish word "letra". Each row shows the output of one stage: spiking digital-filter-based cochlea (row 1), phoneme recognition neural network (rows 2 and 3), DelayNeuron chain (row 4) and words recognition neural network (row 5). A row also includes a spiking cochleogram image (left-hand column) and a histogram (right-hand column).

First, the spiking digital-filter-based cochlea transforms the PCM values of this speech signal into spike trains. The spike address has a value associated with a cochlea band within the range of 0–20. In the left-hand column, it displays the spiking representation of the two phonemes of the Spanish word "letra". Corresponding to the first phoneme, the cochlea bands 1 and 9 are the most active bands; they are the cochlea bands which contain the F1 and F2 frequency components of the /e/ phoneme. For the second phoneme, the most active cochlea bands are the bands 3 and 7, which contain the F1 and F2 frequency components of the /a/ phoneme (Table 6). Previous to the /a/

Table 9
Words WTANeurons parameters.

	WTANeuron AE	*WTANeuron* AO, OA, UI	*WTANeuron* EA, EO, IE, UE	*WTANeuron* IA	*WTANeuron* IO	*WTANeuron* UA
We	4	4	4	4	4	4
Wi	2	8	1	4	5	4
Potential threshold	2	15	1	4	1	1

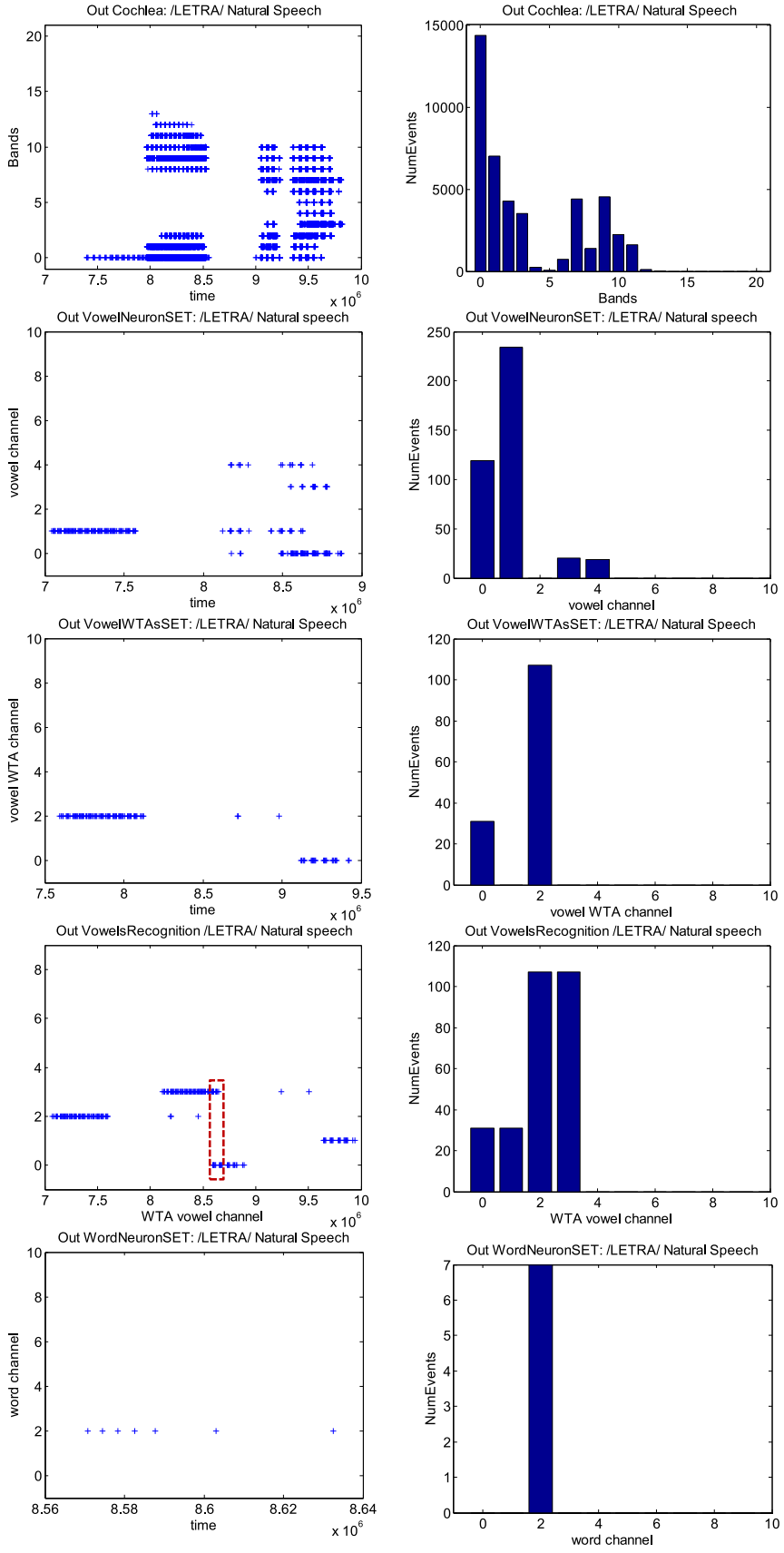


Fig. 14. Recognition of Spanish word "letra".

Table 10
Hit rate (%) and miss rate (%) of the phoneme recognition process.

	Identified phoneme				
	/a/	**/e/**	**/i/**	**/o/**	**/u/**
Stimulus /a/	98.57	0.00	0.00	0.00	1.43
Stimulus /e/	0.00	77.56	1.17	0.00	21.27
Stimulus /i/	0.00	0.00	100.00	0.00	0.00
Stimulus /o/	0.01	0.00	0.00	93.61	6.37
Stimulus /u/	0.00	0.00	0.00	0.64	99.36
Average hit rate: 93.82%					
Average miss rate: **6.18%**					

phoneme, the spiking output is distinguished for the /r/ phoneme.

The second and the third rows display the output of the vowel phoneme recognition neural network. First, each of the 5 RNeurons identify their patterns from the spiking digital cochlea output. The /e/ phoneme is recognized correctly, meaning that the RNeuron with the identifier 1 is the most active; however, the /a/ phoneme is identified with the /e/, /o/ and /u/ phonemes. Although the RNeuron with identifier 0 is the most active, the other 3 RNeurons with identifiers 1, 3 and 4 have also recognized its pattern.

For this reason, the WTANeurons process the RNeuron output to enhance the recognition process. The third row displays the WTANeurons output; the WTANeurons with addresses 2 and 0 are the winner neurons, associated to the first phoneme /e/ and the second phoneme /a/, respectively. For the second phoneme, it some spikes can be observed with address 2 because a leaky winner neural network is used in this implementation.

In the fourth row, the DelayNeurons output is represented. The spikes issued by the previous stage are delayed. This allows the spikes associated to the first phoneme and the second phoneme to match in time, and they arrive to the words recognition neural network at the same time. Enclosed in the red rectangle, the first phoneme is delayed (odd address, 3) and the second phoneme occurs in time (even address, 0).

The output of the words recognition neural network are displayed in the fifth row. The outputs are spikes with identifier 2. The RNeurons and the WTANeurons of this network get to identify the sequence of two phonemes, /e/ and /a/ (Table 8).

Following is a discussion regarding the results of the experiments conducted to evaluate the proposed work. Two sets of experiments were performed, depending on the type of stimulus, for Spanish vowel phonemes and Spanish two-syllable words.

In the first experiments, each speech stimulus was recorded 10 times by 7 speakers. All speakers are Spanish males, aged 20–50. Audio was recorded with a sampling rate of 11025 Hz with 16 bits.

The phoneme recognition neural network was set to identify the 5 Spanish vowel phonemes. These experiments were used to determine the F1 and F2 values for each Spanish vowel phoneme and the parameters of RNeuron and WTANeuron (Tables 6 and 7).

Table 10 summarizes the hit and miss rates of the vowel phonemes recognition process. For each vowel stimulus, it shows the average hit rate and the average miss rate are independent of the speaker. The phoneme /a/ is correctly identified 98.57% of the time; only 1.43% of the time is the phoneme /u/ is wrongly recognized instead of the phoneme /a/. The phoneme /e/ is recognized 77.56% of the time, and the system fails 1.17% of the time because the phoneme /i/ is identified; 21.27% of the time the phoneme /u/ is recognized. The phoneme /i/ is identified 100% of the time. The phoneme /o/ is recognized 93.61% of the time, and in other case, the phoneme /u/ is incorrectly identified. The phoneme /u/ is recognized 99.36% of the time, and only 0.64% of the time is the phoneme /o/ wrongly identified. The phoneme recognition process has a global hit rate of 93.82%.

In the second experiments, each speech stimulus was recorded 10 times by 16 speakers. All speakers are Spanish males, aged 20–50. Audio was recorded with a sampling rate of 11025 Hz with 16 bits.

The recognition neural network is configured for the recognition of 20 two-syllable words, such as sequences of two vowel phonemes. Tables 11 and 12 show the hit rates for each speaker and Table 13 shows the average hit rate of all speakers for each word.

In this implementation, a word is recognized from the identification of its vowel phonemes. So, others consonant phonemes are not taken into account. For this reason, there is a greater difficulty in recognizing some words (with an average hit rate under 40%): “LETRA”, “LISO”, “MUSA”, “PERLA”, “ROSA” or “TEMPLO”, where the phoneme /s/, /r/ and /l/ affects to the F1 and F2 values of the vowel phonemes. This problem can be solved adding more RNeurons configured for the recognition of consonant phonemes.

The other words are recognized with an average hit rate over 40%. Therefore, it proves that this architecture based on spiking neuron can identify two-syllable words.

Table 11
Hit rate (%) obtained in the words recognition process (speaker 1–8).

	Speaker 1	**Speaker 2**	**Speaker 3**	**Speaker 4**	**Speaker 5**	**Speaker 6**	**Speaker 7**	**Speaker 8**
“LETRA”	23.20	38.10	43.30	5.80	48.60	22.40	13.60	11.80
“LISO”	42.20	69.50	92.10	0.00	35.00	22.90	70.60	82.60
“MUSA”	8.50	2.90	18.20	0.00	3.50	0.00	11.10	1.70
“PERLA”	76.90	51.70	50.70	62.50	18.70	14.10	46.80	0.00
“ROSA”	4.30	5.30	0.00	0.00	0.40	0.60	0.00	1.20
“TEMPLO”	9.20	97.50	0.00	0.00	0.00	38.40	0.00	25.60
“CERO”	59.20	67.60	72.20	92.00	26.00	72.20	91.80	28.00
“CHINO”	0.00	39.20	49.80	0.00	0.00	58.40	95.50	0.00
“CITA”	79.90	84.60	93.60	100.00	53.30	9.10	8.00	27.10
“DIQUE”	99.10	100.00	0.00	99.40	0.00	24.60	100.00	100.00
“FASE”	77.50	56.80	20.90	63.50	85.20	59.70	78.10	30.10
“GOMA”	9.80	62.60	89.80	85.60	77.90	5.00	52.70	24.00
“MILLA”	74.40	91.80	97.30	98.00	0.00	11.40	72.60	41.60
“NUBE”	45.30	69.00	92.40	100.00	100.00	44.60	99.30	38.20
“PINO”	77.70	78.10	0.00	0.00	82.60	77.90	100.00	20.20
“PODA”	63.30	84.30	82.60	62.50	24.30	38.70	30.80	38.50
“RIMA”	53.40	78.60	90.40	98.10	68.40	45.80	20.00	77.20
“SEDA”	75.30	67.80	92.30	83.60	43.90	41.80	55.80	69.90
“TIRA”	42.30	69.00	43.70	90.50	10.50	62.80	14.30	4.00
“VEGA”	55.10	54.50	83.30	63.50	65.30	11.40	18.50	81.70
Average hit rate	48.83	63.45	55.63	55.25	37.18	33.09	49.98	35.17

Table 12
Hit rate (%) obtained in the words recognition process (speaker 9–16).

	Speaker 9	Speaker 10	Speaker 11	Speaker 12	Speaker 13	Speaker 14	Speaker 15	Speaker 16
"LETRA"	100.00	36.90	51.90	50.90	31.30	41.90	67.30	0.00
"LISO"	0.00	99.00	52.90	0.00	14.70	84.20	0.00	1.60
"MUSA"	100.00	0.00	0.20	2.00	6.50	9.70	5.30	0.00
"PERLA"	12.50	17.40	9.90	26.00	5.20	48.80	0.00	18.20
"ROSA"	15.80	0.00	0.00	0.00	38.50	8.70	0.00	0.00
"TEMPLO"	0.00	18.10	46.60	0.00	0.00	0.00	0.00	90.60
"CERO"	90.60	92.50	61.30	82.80	94.70	1.40	69.60	2.70
"CHINO"	0.00	92.40	78.00	0.00	0.00	100.00	100.00	50.00
"CITA"	88.40	36.00	70.20	81.00	87.00	67.00	0.00	13.10
"DIQUE"	77.80	100.00	99.70	69.40	100.00	4.00	46.40	100.00
"FASE"	91.80	64.50	52.30	100.00	100.00	2.40	65.80	0.00
"GOMA"	25.80	64.00	37.70	10.90	86.00	81.60	87.80	77.20
"MILLA"	0.00	57.30	57.30	100.00	16.20	15.10	98.00	65.10
"NUBE"	100.00	90.60	94.10	100.00	15.00	29.70	95.50	46.10
"PINO"	56.40	91.90	89.80	11.60	44.20	31.70	0.00	98.40
"PODA"	0.00	91.30	75.70	27.00	41.70	40.00	90.70	57.70
"RIMA"	90.30	78.00	80.10	42.00	10.90	21.10	58.90	16.10
"SEDA"	86.10	63.50	69.10	95.40	82.10	57.70	84.60	31.30
"TIRA"	68.20	74.40	45.60	40.80	46.60	59.40	0.80	12.10
"VEGA"	74.10	41.80	75.80	54.70	55.80	84.90	66.60	18.20
Average hit rate	53.89	60.48	57.41	44.73	43.82	39.47	46.87	34.92

Table 13
All speakers' words recognition average hit rate (%).

"LETRA"	36.69	**"TEMPLO"**	20.38	**"FASE"**	59.29	**"PODA"**	53.07
"LISO"	41.71	**"CERO"**	62.79	**"GOMA"**	54.90	**"RIMA"**	58.08
"MUSA"	10.60	**"CHINO"**	41.46	**"MILLA"**	56.01	**"SEDA"**	68.76
"PERLA"	28.71	**"CITA"**	56.14	**"NUBE"**	72.49	**"TIRA"**	42.81
"ROSA"	4.68	**"DIQUE"**	70.03	**"PINO"**	53.78	**"VEGA"**	56.58

Table 14
FPGA Resources used in the implementation of RNeurons, WTANeurons and DelayNeurons.

	Logic elements		Memory (bits)
	Registers	***LUTs***	
RNeuron	121/114480 (< 1%)	42/114480 (< 1%)	336/3981312 (< 1%)
WTANeuron	104/114480 (< 1%)	63/114480 (< 1%)	336/3981312 (< 1%)
DelayNeuron	3148/114480 (3%)	2788/114480 (2%)	

5. Conclusions

A recognition neural network based on neuromorphic principles is described. This network uses the neuromorphic communication protocol, the AER protocol.

A flexible and scalable solution has been presented. The spiking neural network is based on two types of spiking neurons (RNeuron and WTANeuron), which can be configured to recognize any vowel phoneme. By combining these spiking neurons, a recognition neural network could be built to identify all phonemes of any language that could be recognized from theirs auditory frequency patterns.

Furthermore, with the third type of spiking neuron (DelayNeuron), it is possible to identify any sequence of phonemes. A system capable to recognize two-syllable words as a sequence of two vowel phonemes has also been implemented; however, if new DelayNeuron chains were added, it would be extended to any length.

The feasibility of implementing a neuromorphic system on hardware platform without a personal computer has been proven. It is a low-cost, low-power option, and it allows parallel and real-time processing.

Table 14 shows the resources usage of the Cyclone IV E FPGA chip to implement each of the spiking neurons.

Table 15 summarises the resource usage of the Cyclone IV E FPGA chip to implement the entire system including the spiking digital-filter-

Table 15
FPGA system resources.

	Spiking digital-filter-based cochlea	Spiking recognition neural networks (phonemes & words)	Total
Registers	3099/114480 (3%)	6811/114480 (6%)	9786/114480 (9%)
LUTs	12978/114480 (11%)	4561/114480 (4%)	23521/114480 (21%)
Memory (bits)		10752/3981312 (< 1%)	10752/3981312 (< 1%)
Multiplier 9-bits	210/532 (39%)		210/532 (39%)
PLL			1/4 (25%)

based cochlea composed of 21 cochlea bands, the phonemes recognition neural network and the words recognition neural network. The total resources includes the elements necessary to implement the interface with the audio codec WM8731/WM8731L of the EP4CE115F29C7-developed board.

For the future, there is a need to automatically adapt this recognition process to the characteristics of a speaker, such as their height, vocal tract length and speed.

Moreover, if resources allow, the system could have two sound sources, so redundant information could allow for a more robust system.

Acknowledgment

This work has been supported by the Spanish Grant (with support from the European Regional Development Fund) BIOSENSE (TEC2012-37868-C04-02/01).

References

[1] J. Misra, I. Saha, Artificial neural networks in hardware: a survey of two decades of progress, Neurocomputing 74 (2010) 239–255. http://dx.doi.org/10.1016/j.neucom.2010.03.021.

[2] S.-C. Liu, Event-Based Neuromorphic Systems, John Wiley, Chichester, 2015 http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0470018496.html.

[3] E. Martínez-Celdrán, En torno a las vocales del español: análisis y reconocimiento, Estud. Fonética Exp. 7 (1995) 196–218 http://www.raco.cat/index.php/EFE/article/viewArticle/144415/0.

[4] V. Marrero Aguiar, La fonética perceptiva: trascendencia lingüística de mecanismos neuropsicofisiológicos, Estud. Fonética Exp. XVII (2008) 207–245.

[5] J. Romero, Campos de dispersión auditivos de las vocales del castellano. Percepción de las vocales, Estud. Fonética Exp. (1988) 181–205 http://www.raco.cat/index.php/EFE/article/view/144227.

[6] C.A. Mead, Neuromorphic electronic systems, Proc. IEEE 78 (1990) 1629–1636. http://dx.doi.org/10.1109/5.58356.

[7] T. Yu, A. Schwartz, J.G. Harris, M. Slaney, S.-C. Liu, Periodicity detection and localization using spike timing from the AER EAR, in: Proceedings of the IEEE International Symposium on Circuits and Systems, 2009, pp. 109–112. https://dx.doi.org/10.1109/ISCAS.2009.5117697.

[8] S.C. Liu, N. Mesgarani, J. Harris, H. Hermansky, The use of spike-based representations for hardware audition systems, in: Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS 2010 Nano-Bio Circuit Fabr. Syst., 2010, pp. 505–508. https://dx.doi.org/10.1109/ISCAS.2010.5537588.

[9] M.Abdollahi, S.C.Liu, Speaker-independent isolated digit recognition using an AER silicon cochlea, in: Proceedings of the 2011 IEEE Biomedical Circuits and Systems Conference, BioCAS 2011, 2011, pp. 269–272. https://dx.doi.org/10.1109/BioCAS.2011.6107779.

[10] C. Li, T. Delbruck, S.-C. Liu, Real-time speaker identification using the AEREAR2 event- based silicon cochlea, in: Proceedings of the IEEE International Symposium on Circuits and Systems, 2012, pp. 1159–1162.

[11] A. Cachón, R.A. Vázquez, Tuning the parameters of an integrate and fire neuron via a genetic algorithm for solving pattern recognition problems, Neurocomputing 148 (2015) 187–197. http://dx.doi.org/10.1016/j.neucom.2012.11.059.

[12] L.P. Maguire, T.M. McGinnity, B. Glackin, A. Ghani, A. Belatreche, J. Harkin, Challenges for large-scale implementations of spiking neural networks on FPGAs, Neurocomputing 71 (2007) 13–29. http://dx.doi.org/10.1016/j.neucom.2006.11.029.

[13] G. Indiveri, S.-C. Liu, T. Delbruck, R. Douglas, Neuromorphic systems, in: L.R. Squire (Ed.)Encyclopedia of Neuroscience, Elsevier Ltd, 2009, pp. 521–528. http://dx.doi.org/10.1016/B978-008045046-9.01410-8.

[14] Q. Yu, H. Tang, K.C. Tan, H. Yu, A brain-inspired spiking neural network model with temporal encoding and learning, Neurocomputing 138 (2014) 3–13. http://dx.doi.org/10.1016/j.neucom.2013.06.052.

[15] S. Hussain, A. Basu, R.M. Wang, T. Julia Hamilton, Delay learning architectures for memory and classification, Neurocomputing 138 (2014) 14–26. http://dx.doi.org/10.1016/j.neucom.2013.09.052.

[16] M. Sivilotti, Wiring Considerations in Analog VLSI Systems, with Application to Field-Programmable Networks, California Institute of Technology, 1991.

[17] G. Cauwenberghs, T.S. Lande, Neuromorphic Syst. Eng. (1998). http://dx.doi.org/10.1007/b102308.

[18] S.-C. Liu, T. Delbruck, Neuromorphic sensory systems, Curr. Opin. Neurobiol. 20 (2010) 288–295. http://dx.doi.org/10.1016/j.conb.2010.03.007.

[19] A. Linares-Barranco, M. Oster, D. Cascado, G. Jiménez, A. Civit, B. Linares-Barranco, Inter-spike-intervals analysis of AER Poisson-like generator hardware, Neurocomputing 70 (2007) 2692–2700. http://dx.doi.org/10.1016/j.neucom.2006.07.020.

[20] F. Pérez-Peña, A. Morgado-Estévez, A. Linares-Barranco, Inter-spikes-intervals exponential and gamma distributions study of neuron firing rate for SVITE motor control model on FPGA, Neurocomputing 149 (2015) 496–504. http://dx.doi.org/10.1016/j.neucom.2014.08.024.

[21] S.-C. Liu, A.V. Schaik, B.A. Minch, T. Delbruck, Event-based 64-channel binaural silicon cochlea with Q enhancement mechanisms, in: Proceedings of the IEEE International Symposium on Circuits and Systems, 2010, pp. 2027–2030.

[22] G. von Békésy, Experiments in Hearing, New York, 1960. https://dx.doi.org/10.1121/1.399656.

[23] M.A. Ruggero, Responses to sound of the basilar membrane of the mammalian cochlea, Curr. Opin. Neurobiol. 2 (1992) 449–456. http://dx.doi.org/10.1016/0959-4388(92)90179-O.

[24] M.A. Ruggero, L. Robles, N.C. Rich, Two-tone suppression in the basilar membrane of the cochlea: mechanical basis of auditory-nerve rate suppression, J. Neurophysiol. 68 (1992) 1087–1099 http://jn.physiology.org/cgi/content/abstract/68/4/1087/npapers3://publication/uuid/077842CE-BD83-4F57-BBAB-B9B8C8F64DC3h.

[25] VHDL Analysis and Standardization Group. ⟨http://www.eda.org/vhdl-200x/⟩, 2010.

[26] A. Corporation, 1 . Cyclone IV Device Datasheet, 3, 2010, pp. 1–44.

[27] W. Microelectronics, WM8731/WM8731L datasheet, Audio (2009).

[28] R. Berner, T. Delbruck, A. Civit-Balcells, A. Linares-Barranco, A 5 Meps $100 USB2.0 address-event monitor-sequencer interface, in: Proceedings of the IEEE International Symposium on Circuits and Systems, 2007. ⟨https://dx.doi.org/10.1109/ISCAS.2007.378616⟩.

[29] E. Zwicker, Subdivision the audible frequency range into critical bands, J. Acoust. Soc. Am. 33 (1961) 248. http://dx.doi.org/10.1121/1.1908630.

[30] Mathworks, Filter Design HDL Coder, 2012. ⟨http://www.mathworks.es/products/datasheets/pdf/filter-design-hdl-coder.pdf⟩.

[31] F. Gómez-Rodríguez, R. Paz, L. Miro, A. Linares-Barranco, Two Hardware Implementations of the Exhaustive Synthetic AER Generation Method, 2005, pp. 534–540.

[32] F. Rosenblatt, The Perceptron, A. Probabilistic, Model for information storage and organization in the brain, Psychol. Rev. 65 (1958) 386–408.

[33] C. Kock, Biophysics of Computation, 1st ed., Oxford University Press, Inc., New York, 1999.

[34] C Eliasmith, Computation, Representation, and Dynamics in neurobiological systems, MIT Press, Cambridge, MA, USA 2003. https://dx.doi.org/10.1017/CBO9781107415324.004

[35] M. Oster, S.-C. Liu, Spiking inputs to a Winner-take-all network, Adv. Neural Inf. Process. Syst. 18 (2006) 1051–1058 (http://papers.nips.cc/paper/2852-spiking-inputs-to-a-winner-take-all-network.pdf/nfiles/2408/Oster? Liu-2006-Spiking Inputs to a Winner-take-all Network.pdf/nfiles/2409/2852-spiking-inputs-to-a-winner-take-all-network.html).