Neuromorphic solutions: digital implementation of bio-inspired spiking neural network for electrocardiogram classification

Dze Rynn Chen, Yan Chiew Wong  
Centre for Telecommunication Research and Innovation (CeTRI), Faculty of Electronics and Computer Engineering (FKEKK), Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia

ABSTRACT

Conventional techniques of off-chip processing for wearable devices cause high hardware resource usage which leads to heat generation and increased power consumption. Hence, edge computing methods such as neuromorphic computing are considered the most promising modern technology to replace conventional processing. It is beneficial to employ neuromorphic processing in electrocardiogram (ECG) classification, enabling engineers to overcome the constraints of heat generation caused by hardware utilization. Thus, this work aims to investigate common building blocks in a spiking neural network (SNN), analyze the spike-based plasticity mechanism and implement ECG classification on a neuromorphic circuit. The MIT-BIH Arrhythmia database (MITDB) is preprocessed in MATLAB, then used to train and test an SNN designed for field programmable gate arrays (FPGA), employing spike-based plasticity and Izhikevich neurons. The behaviour of spike timing dependent plasticity (STDP) in a neuromorphic circuit is also visualized in this work. The state-of-the-art performance of this work lies in providing a generic mechanism to adapt ECG classification into a neuromorphic solution, a non-Von Neumann architecture. The proposed digital design utilizes 1.058% of hardware resources on a Zedboard. Application-wise, this work provides a foundation for development of neuromorphic computing in wearable medical devices that perform continuous monitoring of ECG.

Keywords: Digital hardware, Edge computing, Electrocardiogram, Field programmable gate array, Field programmable gate arrays, Neuromorphic, Spiking neural network

This is an open access article under the CC BY-SA license.

1. INTRODUCTION

Cardiac arrhythmias refer to the impairment of the electrical impulses coordinating human heartbeats and are used to identify the presence of cardiovascular diseases (CVDs). Due to the nature of arrhythmias that can reflect electrical activities in the heart, they can be detected by analyzing electrocardiogram (ECG) signals taken from the body [1]. To do this, the conventional way was for medical professionals to manually inspect results of an ECG test. However, arrhythmias occur intermittently, and thus are difficult to detect based solely on ECG tests. Therefore, continuous monitoring of ECGs is crucial in early detection of potential cardiovascular problems. In status quo, most wearable devices employ the mechanism of collecting data, then transmitting it to external servers which perform off-chip processing [2]. There are several problems with this. Firstly, conventional techniques of using remote servers and signal processing requires intensive computation and processing, causing higher hardware resource usage. Due to
neuromorphic circuits are considered the most promising modern technology beyond Von Neumann processing [3]. SNNs used in neuromorphic computing do not employ the fetch-and-execute cycle of Von Neumann architectures but rather process data in the form of event-driven spikes, with an emerging learning mechanism which is spike-based plasticity. However, a downside of neuromorphic circuits is that they are not general-purpose, but rather need to be designed specifically, or customized for their applications. The customization of neuromorphic circuits for the purpose of ECG classification on field programmable gate arrays (FPGAs) is an area that lacks of research, thus this work aims to fill that gap by proposing a novel generic flow to employ neuromorphic methods in ECG classification. The objectives of this work are to investigate common building blocks and techniques used for a neuromorphic circuit based on an SNN, analyze spike-based plasticity mechanism in neuromorphic circuit, and implement ECG classification on a neuromorphic circuit.

2. RELATED WORKS

2.1. Discrete wavelet transform

With a continuous input signal such as ECG, it is common practice to discretize it [4] before processing because it is computationally impossible to analyse a continuous signal using all wavelet coefficients. Recent studies [5], [6] have shown the effectiveness of the discrete wavelet transform (DWT). Unlike the Fourier transform, the DWT domain is able to describe both frequency and time representation. Another way to interpret the process of performing wavelet transform (WT) is as a tool for signal decomposition. Agil et al. [7] and Shem and Shareena [8], the DWT is used to decompose the noisy ECG signal, before removing the noisy elements and reconstructing it. Ahmed et al. [9], the nominal wavelet for MIT-BIH database is the Daubechies wavelet family. Syama et al. [10] found that Daubechies wavelet to be able to accurately filter ECG signals with its ability to retrieve lost data, reducing distortion of the compressed ECG data.

2.2. Spiking neural network

Spiking neural network (SNNs) are considered the third generation [11] of artificial neural networks (ANNs), after McCulloch-Pitts neurons and continuous activation. SNNs were originally modeled as a means of biological signal processing in the human brain, where data and information are passed around through neurons via spikes. Information processing in SNN depends on the timing of the spikes [12], which will influence the weights of the synapse connection in transmitting the information to other neurons. The use of SNNs for electrocardiogram classification has been explored in multiple research papers as of late, as listed in Table 1. The proposed design employs the Izhikevich (IZH) neuron model. The synaptic currents from presynaptic neurons increase the neuron’s membrane potential when they arrive. This then introduces two possible outcomes. The first is if the total presynaptic currents arriving at the membrane are not sufficient to cause the neuron to spike, then the membrane voltage is reset to its original value before any of the currents arrived. The second possibility is if the total presynaptic currents do in fact suffice to cause a spike, then the membrane voltage, v and the recovery variable, u are reset. Recent studies Elnahawy et al. [13] and Ismail et al. [14] show successful low-power implementations of the Izhikevich model on hardware, proving that the Izhikevich model is indeed practical for complex hardware implementation as intended by Izhikevich himself. The coordinate rotation digital computer (CORDIC) algorithm [15] which employs SNN has been shown to enable the design of low power digital circuits. Recent works have also further proven the low-power and low-resource advantages of SNN on hardware [16], [17].

![Table 1. Past research using SNNs for ECG classification, without hardware implementation](image)

<table>
<thead>
<tr>
<th>Paper</th>
<th>ANN type</th>
<th>SNN model</th>
<th>Preprocessing</th>
<th>Training database</th>
<th>Abnormal categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yan et al. [18] (2021)</td>
<td>SNN, CNN</td>
<td>LIF</td>
<td>Spike rate encoding</td>
<td>MIT-BIH Arrhythmia</td>
<td>4</td>
</tr>
<tr>
<td>Rana et al. [19] (2021)</td>
<td>SNN</td>
<td>LIF</td>
<td>Probabilistic encoding</td>
<td>PTB Diagnostic ECG</td>
<td>1</td>
</tr>
<tr>
<td>Bauer et al. [20] (2019)</td>
<td>RNN</td>
<td>-</td>
<td>Sigma-delta encoding</td>
<td>MIT-BIH Arrhythmia</td>
<td>5</td>
</tr>
<tr>
<td>Amurshahi et al. [21] (2019)</td>
<td>SNN</td>
<td>LIF</td>
<td>Gaussian layer</td>
<td>MIT-BIH Arrhythmia</td>
<td>1</td>
</tr>
<tr>
<td>Kolaçagioğlu [22] (2018)</td>
<td>SNN</td>
<td>LSM</td>
<td>Gaussian layer</td>
<td>MIT-BIH Arrhythmia</td>
<td>16</td>
</tr>
</tbody>
</table>

SNNs employ spike timing dependent plasticity (STDP) to operate. The transfer and storing of signals are determined by the strength of the connection of the synapses, also known as the synaptic plasticity. Modifications of this synaptic connectivity based on activity is the core of the human brain’s learning process [23] as well as memory. It has been proven by recent research [24], [25] that repetitive simulation of a presynaptic cell immediately before spikes occur in a postsynaptic cell will cause synaptic

Neuromorphic solutions: digital implementation of bio-inspired spiking ... (Dze Rynn Chen)
Neuromorphic computing

In 1965, Moore’s law [27] was introduced, stating that the number of transistors able to fit on a chip doubles every two years. Since then, it has been a defining rule pertaining to the scaling of transistor size. However, as of late, the process of shrinking transistors has been replaced by newer technologies and computing methods such as analog computing and stochastic computing [28]. Neuromorphic computing is one of the more advanced techniques being explored. A defining feature of neuromorphic computing is the emulation of the human brain’s behaviour to complete tasks. Neuromorphic computing sends information through networks of synapses, in line with McCulloch and Pitts’ description of neurons being an arithmetic function of its synapses [29]. Neuromorphic hardware can be divided into two types, where ANNs and brain-inspired neural networks are placed in different categories. While ANNs are also inspired by human brain activity in terms of their working principle, the main difference between ANNs and brain-inspired networks is the training methodology, where the latter employs learning mechanisms that are also derived from neurobiological systems, such as STDP. Semiconductor companies such as Intel and IBM have produced their own versions of neuromorphic chips, namely Intel’s Loihi [30] and IBM’s TrueNorth chip [31].

3. RESEARCH METHOD

This work aims to propose a digital design for implementation of neuromorphic computing on an FPGA. An SNN algorithm is developed for the purpose of ECG classification which employs spike-based plasticity to train neurons. Using electronic design automation (EDA) tools which are Vivado as well as MATLAB, raw ECG data is obtained from the MIT-BIH Arrhythmia database (MITDB), which is obtained via Physionet. The database contains 48 ECG records taken on two leads, which are the MLII lead and V5 lead. Only the data from the MLII lead is used, which accurately depicts the overall condition of the ECG record. In this work, the duration of records taken is 10 seconds long containing 3500 samples. This may present limitations due to the short length of the records, however it ensures that the preprocessing step does not over compress the data. This scope may be widened in future works as the design is very much portable and can be used on a bigger scale. The raw ECG data undergoes preprocessing in MATLAB and is compressed into 35-bit binary stream so that it can be fed into the SNN. The SNN algorithm is developed based on adaptations of methods used in past research on developing SNNs [10]-[14]. With 35 normal records and 13 abnormal records for the 10-second-long MITDB records, the algorithm is trained to identify normal and abnormal heartbeat rhythms by using 10 records for training and 3 records for testing in each category. The algorithm is developed in a hardware description language (HDL) which is very high speed integrated circuit (VHDL), so that it can be synthesized and optimized for the target field programmable gate array (FPGA) which is Zedboard. The design is represented in the form of a neuromorphic circuit, which is then analyzed in terms of hardware resource utilization.

3.1. Preprocessing

To make the ECG signal usable in an SNN, the preprocessing stage requires a step to ensure the ECG signal is in a digitized form that is readable by the network. The ECG preprocessing steps are performed using the MATLAB software before moving the digitized data into the SNN in Vivado. The proposed preprocessing algorithm is illustrated in Figure 1 and consists of two techniques which are the discrete wavelet transform (DWT) and binarization. In the proposed method, DWT is applied to the signal twice. As observed in Figure 1, signal A is the input for the first DWT and signal B is the input for the second. While the purpose of transforming signal A is mainly for denoising and compressing the signal, the discrete wavelet transformation on signal B is mainly for compressing the signal. The proposed DWT serves to compress the data by down sampling it using Daubechies wavelet, adapting the methodology used in [32]. The DWT process compresses the data so that it is in a lower frequency than the original signal. Generally, the approach in selecting the vanishing moments is that when the vanishing moments are lower, more signal detail is lost, leading to more distortion of the original signal [33].

In this work, db6 is used for the first round of DWT and db2 for the second round. This is to minimize distortion of the signal while at the same time ensuring the size of the ECG signal fitted into the SNN is large enough for detail. The normalization process rescales data so that its maximum amplitude is 1 while its minimum is 0. This is done so that the data and its changes can be easily observed throughout the
preprocessing steps. In a digital system, data is processed in a digital form. As observed from the output of the DWT, the transformed ECG signal is still in analog form. Thus, the data has to go through binarization to convert it to 0’s and 1’s that can be processed by the SNN. The threshold values are selected based on the amplitude of the normalized R peaks, which are generally between 0.6 to 0.7 (60% to 70%) of the R peak’s amplitude after the first round of DWT and 0.4 to 0.5 (40% to 50%) after the second round.

![Figure 1. Overall preprocessing stages](image)

**3.2. Spike timing-dependent plasticity**

Employing STDP in an SNN addresses the drawbacks of off-chip processing, commonly used in ECG classification applications such as the Apple Watch [34]. The practicality of STDP lies in its ability to enable on-chip processing through deep learning, resulting in lower computational power consumed. Thus, adapting the approach in [12], the digital block in Figure 2 is modelled to represent the STDP module. Figure 2(a) shows the three outputs write to the random access memory (RAM) of the neurons, where each neuron has its own RAM. The clock (CLK) input is for the clock signal, which coordinates and paces the module’s actions. EN is responsible for receiving the enable signal which activates the learning process, and EN_Addr changes the neuron connections accordingly while applying the STDP rule. Figure 2(b) shows Pre_Spikes reads the spikes of the previous neuron, and Post_Spike of the next neuron. The degree of synaptic weight change that occurs based on impulse time between the pre- and postsynaptic neurons is expressed in (1) where A+ and A− refer to the amount of change in synaptic weight at t=0+ and t=0− respectively, and τ+ and τ− show the decrease in change in synaptic weight.

![Figure 2. STDP learning module (a) working principle and (b) digital architecture](image)
3.3. Neuron model

The neuron model emulated in this work is the Izhikevich (IZH) neuron as shown in Figure 3. The neuron module is adapted from [12] that is a digital model of the IZH neuron to simplify computations. Setting different parameter values results in different patterns of intrinsic activation as shown in Figure 3(a), allowing the behavior emulation of diverse real biological neurons. In this work, RS regimen is used, applying (1). The architecture is shown in Figure 3(b), where the CLK signal provides pace and coordination, the EN signal for neuron activation, the WE signal for Write Enable, Addr signal for the neuron addresses, AER_Bus signal which is the input carrying information about which neuron spiked previously, and an output signal labelled Spike_out to show whether or not the neuron spiked. If more than one neuron spiked simultaneously, the address event representation (AER) bus halts all operations in the neuron because it can only transmit one spike at any given time. The input align block limits the synaptic weight’s negative value to -140 mV to avoid spikes being generated when it is not supposed to.

\[
W(x) = \begin{cases} 
    A_+ \exp \left( -\frac{x}{\tau_+} \right) & \text{if } x > 0 \\
    A_- \exp \left( \frac{x}{\tau_-} \right) & \text{otherwise.}
\end{cases}
\]  

(1)

3.3. Neuron model

The neuron model emulated in this work is the Izhikevich (IZH) neuron as shown in Figure 3. The neuron module is adapted from [12] that is a digital model of the IZH neuron to simplify computations. Setting different parameter values results in different patterns of intrinsic activation as shown in Figure 3(a), allowing the behavior emulation of diverse real biological neurons. In this work, RS regimen is used, applying (1). The architecture is shown in Figure 3(b), where the CLK signal provides pace and coordination, the EN signal for neuron activation, the WE signal for Write Enable, Addr signal for the neuron addresses, AER_Bus signal which is the input carrying information about which neuron spiked previously, and an output signal labelled Spike_out to show whether or not the neuron spiked. If more than one neuron spiked simultaneously, the address event representation (AER) bus halts all operations in the neuron because it can only transmit one spike at any given time. The input align block limits the synaptic weight’s negative value to -140 mV to avoid spikes being generated when it is not supposed to.

\[
W(x) = \begin{cases} 
    A_+ \exp \left( -\frac{x}{\tau_+} \right) & \text{if } x > 0 \\
    A_- \exp \left( \frac{x}{\tau_-} \right) & \text{otherwise.}
\end{cases}
\]  

(1)

4. RESULTS AND ANALYSIS

4.1. Preprocessing

The DWT is applied to the ECG data obtained from the MIT-BIH arrhythmia database (MITDB). Figure 4 shows the results of each individual step of preprocessing performed on normal and abnormal ECG signals respectively. Due to the R peaks in the ECG signal still being prominent after the first round of DWT as shown in the figure, the peaks are preserved before the second round of DWT by binarizing the signal at

Figure 3. IZH neuron (a) effect of neuron’s parameters and variables [35] and (b) digital architecture

Figure 4. DWT results of ECG data obtained from the MIT-BIH arrhythmia database (MITDB).
this point. By doing this, the R peaks are binarized as 1’s and the rest of the signal as 0’s. The second round of DWT is performed, resulting in a 35-bit representation of the signal. This is binarized to get a 35-bit binary representation of the ECG signal.

4.2. Training

With the 35-bit binary ECG data, the SNN in [16] can be adapted to classify the data as shown in Figure 5. The pseudocode for increment of array pointer is shown in Figure 5(a), where the switching of data is controlled by Image_Signal(0) and Image_Signal(1). The data is arranged in an array in the Top script and fed into the network via the Digit signal. Two counter variables are introduced, n_counter for normal data and a_counter for abnormal data. The normal data is inserted when Image_Signal(0) is 1, otherwise abnormal data is inserted if Image_Signal(1) is 1. The pseudocode for feeding the data into the network is shown in Figure 5(b) and the result showing insertion of multiple ECG records for training is shown in Figure 6.

![Figure 4. Preprocessing ECG data using DWT](image)

![Figure 5. Pseudocode (a) increment of array pointer and (b) insertion of multiple data](image)

![Figure 6. Insertion of training data automated by counters moving through normal and abnormal data arrays](image)

4.3. Testing

For testing the network, three ECG records are used for each category. The test data is inserted into the Digit_Noise signal, which is modified through the testbench of the Top script. The output of the SNN is observed at neurons 37 and 38, which are the output neurons for normal and abnormal categories respectively. When normal data is inserted into the network via Digit_Noise, neuron 37 spikes as shown in Figure 7. On the other hand, if abnormal data is inserted into the network via Digit_Noise, neuron 38 spikes as shown in Figure 8. The accuracy of this classification is generally correct, although the exact figure for accuracy percentage is unable to be determined solely based on the behavioral simulation, because the results at this stage are only observed at the neuron’s waveform. In this situation, there is a rare instance of both neurons spiking at the same time, which cannot be quantified. As an example, in Figure 8, neuron 37 may...
spike for a duration of 1us at random moments, while neuron 38 is spiking the whole time. This condition is likely to happen when the duration of training time is either insufficient or excessive. This shows that training phase time is the key in successful implementation of SNNs, as proven in [12] where different patterns entail different training times to trigger the correct neuron with precisely the right timing.

Figure 7. Neuron 37 spikes to classify record number 103 correctly as normal

Figure 8. Neuron 38 spikes to classify record number 217 correctly as abnormal

4.4. Spike-based plasticity

Applying (1), the mechanism and working principle of STDP is visually represented in Figures 9. The simulation shows a neuron connected to a previous layer of three neurons. The Pre_Spike signal contains three binary numbers representing the synapses of that previous layer of neurons. The Post_Spike signal goes high when the neuron fires. In Figure 9(a), one of the synapses on Pre_Spike signal spikes at around 20ns. At 150 ns, the Post_Spike signal goes high to indicate that the post-synaptic neuron has fired. The theory applied here is that due to the neuron firing after the pre-synaptic neuron fired, it is assumed that the pre-synaptic neuron contributed to the firing of the neuron by increasing the voltage at the membrane of the neuron, making it closer to the threshold value. From this theory, the weight of the synapse between the post-synaptic neuron and the first pre-synaptic neuron is increased from 0 to 1, indicating that the synapse has strengthened. Figure 9(b) shows the opposite scenario of Figure 9(a), where the Post_Spike signal goes high before the Pre_Spike signal does. According to the STDP rule, it is assumed that the synapse between the first pre-synaptic neuron and the post-synaptic neuron are less significant, thus the weight of that synapse is decreased from 0 to -2. In this work, STDP rule is applied to train the neurons, where neurons 37 and 38 are the neurons in the output layer trained to recognize normal and abnormal ECG data respectively, neurons 0 to 34 are the input neurons where each neuron corresponds to one bit of binarized ECG data, and neurons 35 and 36 are training neurons. Figure 10 shows a portion of the elaborated design of this work, where the connections of neuron modules in the physical design can be observed.

Figure 9. STDP when the pre-synaptic neuron fires (a) after the post-synaptic neuron and (b) after the post-synaptic neuron
4.5. Neuromorphic implementation

The utilization of this design is tabulated in Table 2. In total, 1877 resources making up approximately 1.058% of the Zedboard FPGA resources are utilized by the design. This indicates that less hardware is needed in the implementation of this design [36], making it more cost-effective. Previous research implementing ECG classification on an FPGA is tabulated in Table 3. It is observed that in terms of hardware resource utilization, the proposed design is on par with results of previous research.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Available</th>
<th>Utilized</th>
<th>Utilization percentage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUT</td>
<td>53200</td>
<td>908</td>
<td>1.71</td>
</tr>
<tr>
<td>LUTRAM</td>
<td>17400</td>
<td>38</td>
<td>0.22</td>
</tr>
<tr>
<td>FF</td>
<td>106400</td>
<td>919</td>
<td>0.86</td>
</tr>
<tr>
<td>DSP</td>
<td>220</td>
<td>2</td>
<td>0.91</td>
</tr>
<tr>
<td>IO</td>
<td>200</td>
<td>10</td>
<td>5.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Publication</th>
<th>Lookup tables</th>
<th>RAM</th>
<th>Slice registers</th>
<th>DSP</th>
<th>IO</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gu et al.  [37] (2016)</td>
<td>10116</td>
<td>91</td>
<td>3433</td>
<td>20</td>
<td>-</td>
<td>13660</td>
</tr>
<tr>
<td>Zhai et al. [38] (2017)</td>
<td>16133</td>
<td>17</td>
<td>11797</td>
<td>12</td>
<td>-</td>
<td>27959</td>
</tr>
<tr>
<td>Madiraju et al. [39] (2018)</td>
<td>4324</td>
<td>-</td>
<td>1540</td>
<td>125</td>
<td>30</td>
<td>6019</td>
</tr>
<tr>
<td>Proposed</td>
<td>908</td>
<td>38</td>
<td>919</td>
<td>2</td>
<td>10</td>
<td>1877</td>
</tr>
</tbody>
</table>

5. CONCLUSION

In this paper, a digital design for a neuromorphic circuit employing an SNN for ECG classification is proposed, which fills the research gap of customizing a non-Von Neumann architecture for the purpose of classifying ECG data. The three building blocks of the system are preprocessing the ECG data, training the SNN with it and testing the SNN. For training, the STDP mechanism is employed which allows the IZH neurons to learn. After the implementation stage, the digital design is analyzed in terms of hardware resource utilization, coming up to about 1.058% of hardware used on the Zedboard. The outcome of this work will be able to further advance the technology of wearable devices in the health and medical industry.

ACKNOWLEDGEMENTS

The authors acknowledge the technical and financial support by Universiti Teknikal Malaysia Melaka (UTeM) and the Ministry of Higher Education, Malaysia, under the research grant no. FRGS/1/2020/ICT02/UTEM/02/1.
REFERENCES


Neuromorphic solutions: digital implementation of bio-inspired spiking

Dze Rynn Chen is a final year bachelor’s degree student at Universiti Teknikal Malaysia Melaka (UTeM), Malaysia. She obtained her diploma in electronic engineering from UTeM in 2019. Her research interests include front end IC design and system design. She can be contacted at e-mail: b021820082@student.utem.edu.my.

Yan Chiew Wong is Associate Professor at Universiti Teknikal Malaysia Melaka (UTeM). She has completed her doctorate from The University of Edinburgh, United Kingdom in 2014. Yan Chiew has more than 8 years industry experience in semiconductor design. She has previously worked at Infineon, Intel and Sofant Technologies. Yan Chiew has involved intensively in the research of integrated circuit (IC) design, Artificial Intelligence (AI) and energy harvesting. She involved in ARMOURS European project, Adaptive Impedance Module (AIM) SMART project, SWARM Horizon project and Demand-Driven Development (3D). She is a recipient of different awards such as ITEX, INNOVATE, IEM, and EDS societies. She has been able to successfully design and develop low power and high voltage CMOS control chips and miniature RF harvesting circuitries. She is equipped with both design and implementation skills. She can be contacted at email: ycwong@utem.edu.my.