



Article CMOS Perceptron for Vesicle Fusion Classification

Mariusz Naumowicz 🗅, Paweł Pietrzak 🗅, Szymon Szczęsny * 🗅 and Damian Huderek 🗅

Institute of Computing Science, Faculty of Computing and Telecommunications, Poznan University of Technology, Piotrowo 3A Street, 61-138 Poznań, Poland; mariusz.naumowicz@put.poznan.pl (M.N.); pawel.pietrzak@doctorate.put.poznan.pl (P.P.); damian.huderek@put.poznan.pl (D.H.) * Correspondence: szymon.szczesny@put.poznan.pl

Abstract: Edge computing (processing data close to its source) is one of the fastest developing areas of modern electronics and hardware information technology. This paper presents the implementation process of an analog CMOS preprocessor for use in a distributed environment for processing medical data close to the source. The task of the circuit is to analyze signals of vesicle fusion, which is the basis of life processes in multicellular organisms. The functionality of the preprocessor is based on a classifier of full and partial fusions. The preprocessor is dedicated to operate in amperometric systems, and the analyzed signals are data from carbon nanotube electrodes. The accuracy of the classifier is at the level of 93.67%. The implementation was performed in the 65 nm CMOS technology with a 0.3 V power supply. The circuit operates in the weak-inversion mode and is dedicated to be powered by thermal cells of the human energy harvesting class. The maximum power consumption of the circuit equals 416 nW, which makes it possible to use it as an implantable chip. The results can be used, among others, in the diagnosis of precancerous conditions.

Keywords: vesicle fusion; exocytosis; edge computing; neural networks; CMOS accelerators; weak-inversion mode



According to estimates by Gartner Research, in the next three years, approximately 75% of data will be processed outside the cloud, i.e., with the use of edge devices [1]. The use of edge devices is the basis of the concept of edge computing, i.e., a distributed data computing environment that works close to the data source (at the edge of the network). The main goal of this concept is to preprocess data before it is sent to the cloud, reducing latency and saving bandwidth [2]. It has been proven many times that edge computing is an approach that particularly increases the efficiency of the computing environment [3,4]. The techniques of edge-computing are one of the most frequently used approaches in the analysis of medical data [5]. The integration of artificial intelligence with mobile devices is used, among others, in hospital intensive care unit [6]. On the other hand, the development of edge-computing techniques makes it also possible to monitor the health of patients without the need to hospitalize them [7]. Examples of edge-computing applications in healthcare include areas such as: analysis and therapy for voice disorder [8], coronary heart disease diagnosis [9], detecting behavior of dementia [10], and even management of medical infrastructure [11]. Regardless of the application, all edge processing systems are characterized by computing efficiency suitable for processing a specific type of data close to its source. In this article, we focused on the implementation of an exocytosis signal processing system (the vital processes of the cell) using an analog CMOS (Complementary Metal-Oxide-Semiconductor) preprocessor. This is an example of analog data processing as close to the sensors as possible.

We took up this topic for two independent reasons. Primarily, the processing of exocytosis signals is an extremely important medical issue, as it allows for early diagnosis of pre-cancerous conditions, before they are visible in the histopathological images [12]. It also allows the diagnosis and monitoring of many other diseases. Exocytosis, its monitoring



Citation: Naumowicz, M.; Pietrzak, P.; Szczęsny, S.; Hudere, D. CMOS Perceptron for Vesicle Fusion Classification. *Electronics* **2022**, *11*, 843. https://doi.org/10.3390/ electronics11060843

Academic Editor: Claus Pahl

Received: 31 January 2022 Accepted: 5 March 2022 Published: 8 March 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). methods, medical applications and the related vesicular fusion process are described in the next section. The second reason why the authors decided to take up this topic are the engineering aspects of the problem. In this case, the source of medical data are individual patient cells, which requires the implementation of the preprocessor as an implantable chip class device [13]. Admittedly, hardware implementations of neural networks for applications in signal processing from sensors or in edge computing are known as: semiconductor circuits [14], based on DAC-based (Digital-to-Analog Converter) circuits [15] or memristor-based circuits [16]. The implementations mentioned operate in a frequency band suitable for the task of analyzing biomedical signals such as vesicle fusion signals. However, they are characterized by power consumption of several dozen to several hundred μ W and are not suitable for processing current signals with an amplitude below 1 nA. In the case of analysis of vesicle fusion signals such the narrow range of signals observed by means of dedicated carbon nanotube (CNT) sensors necessitates the use of reduced supply voltage modes in the computing circuits. This is a big challenge, especially since the monitoring of precancerous conditions is a long process and should take place without the need to hospitalize the patient. This requires providing a system that, despite high computing efficiency, will be able to be powered by cells with low efficiency. Strong reduction of power consumption while maintaining high computing efficiency was the main goal of the presented approach. For all these reasons, the paper presents the methodology of designing an edge processing system that implements the functionality of a neural network for the classification of exocytosis signals. Particular emphasis was placed on ensuring low power consumption, with the possibility of supplying the system with thermal cells used in human energy harvesting techniques [17]. Efforts were also made in the network training phase to ensure minimal use of hardware resources which goes hand in hand with designing small, non-invasive computing systems.

The article is organized as follows. Section 2 describes the vesicular fusion process which is the basic mechanism of cell exocytosis. Various methods of monitoring fusion have been described and the sensor techniques used for this purpose. Section 3 describes the weak-inversion mode of CMOS semiconductor circuits. Its use in the implementation of an analog preprocessor for processing fusion signals near cells has been justified. Section 4 is devoted to the problem of training a neural network model for the fusion signal classification task. The limitations of the learning process resulting from the hardware implementation of the classifier were discussed. Section 5 presents parameters of the fusion signal classifier implemented as an analog CMOS system in weak-inversion. A summary of all the advantages of implementing analog CMOS circuits as edge devices is provided in Section 6.

2. Vesicle Fusion

In multicellular organisms, two basic processes of substance transport between cells take place: exocytosis and endocytosis [18]. For many years, research has been carried out to understand these mechanisms, as they constitute the basis of many life processes, making it possible to, among others, monitor and diagnose cancer [19], in particular cancer metastases [20]. There is also evidence linking these processes to precancerous conditions in cells [12]. Monitoring of exocytosis and endocytosis is also the basis for diagnosing Alzheimer's disease [21] or thrombosis [22]. The mechanism of exocytosis and endocytosis is based on the transport of the so-called vesicles [23], which carry chemical messages [24]. In case of exocytosis, vesicles move outside cells, and in case of endocytosis, inside cells. Communication between cells is based on both of these processes—a vesicle formed in the process of exocytosis in one cell is absorbed in the process of endocytosis in another cell. A particular contribution to understanding the vesicle-based transport process was done by James E. Rothman, Randy W. Schekman and Thomas C. Südhof, who for their research received the Nobel Prize in Physiology or Medicine in 2013. There are basically two types of vesicle fusion: a full fusion and a partial (kiss-and-run) fusion [25]. In case of the full fusion, the entire content of the vesicle is released in the process of exocytosis. In the kiss-and-run fusion, only part of the vesicle content is released in the process of exocytosis, while the remainder is absorbed by the same cell in the process of endocytosis.

There are two methods of detecting vesicle fusions. The first approach is based on the analysis of image sequence obtained using Total Internal Reflection Fluorescence (TIRF) Microscopy [26]. This approach uses machine learning methods mainly based on Convolutional Neural Network [27] or Hierarchical Convolutional Neural Network (HCNN) [28]. The precision of both types of networks in the task of classification of the full and the partial fusion is higher than 95%. Due to the algorithm of deep networks and the need to analyze a sequence of photos, this approach is, however, very expensive for an application in edge computing systems. Alternative and less computationally expensive methods of detecting vesicle fusions are based on the analysis of amperometric signals [29]. An example of an amperometric AFE (Analog Front-End) system for monitoring vesicle fusion signals during exocytosis is described in paper [30]. The basis of the system is a sensor in the form of a Carbon Nanotube electrode array (CNT) [31], electrodes of which are attached directly to the tissue. This technology is covered by a NASA patent [32] and is being applied, among others, in medicine, early disease diagnosis, implantable sensors, analytical instruments [33]. A single AFE for the task of detecting fusions consists of three electrodes: a reference electrode (representing the reference potential), an anode and a cathode, which oxidize charged vesicles [34]. An application with voltammetric AFE [35] is also possible. In such a situation, the use of a voltage-to-current CMOS converter is additionally required [36]. Examples of current signals observed at the electrodes are presented in Section 4, describing the dataset used in the research for training the classifier of signals of the full and the partial fusion. The mentioned section presents the architecture of the perceptron network, which is the base for the classifier. It is worth noting that the computational complexity of the perceptron-based classifier is in this case much lower than that of systems used in the task of detecting fusion in an image sequence. This is because the calculations were moved as close as possible to the data source – near cells, next to the sensor matrix.

3. Weak Inversion Mode

This paper presents the implementation of the classifier as a CMOS circuit. Processing the biological signals of vesicle fusion requires the use of an unusual mode of operation of the semiconductor structure. CMOS circuits can work in three different modes depending on the supply voltage level [37]. Most applications use the strong inversion mode, which means that the gate voltage of the MOS transistor with respect to the source voltage (V_{GS}) satisfies the condition: $V_{GS} > V_T + 100$ mV, where V_T is the technology threshold voltage. Lowering the V_{GS} voltage to the range $V_T + 100$ mV > $V_{GS} > V_T - 100$ mV will enter the moderate inversion mode [38]. By lowering the supply voltage further to the $V_{GS} < V_T - 100$ mV range, it enters the weak inversion mode. This mode is often used to implement analog computing circuits: multiplier/divider circuits [39], amplifiers [40] or comparators [41]. In this work, the weak inversion mode was used, which results from the need to process currents below 1 nA (typical for amperometric applications) and the need to ensure low power consumption (typical for implantable chips class applications). In weak inversion mode, the drain currents of the MOS transistors are described by the Equations (1) and (2) [37]:

$$I_{Dn} = I_0 \frac{W}{L} e^{\frac{\kappa V_G}{U_T}} \cdot \left(e^{\frac{V_S}{U_T}} - e^{\frac{V_D}{U_T}} \right)$$
(1)

$$I_{Dp} = I_0 \frac{W}{L} e^{\frac{\kappa V_W - V_G}{U_T}} \cdot \left(e^{\frac{-(V_W - V_S)}{U_T}} - e^{\frac{-(V_W - V_D)}{U_T}} \right)$$
(2)

where I_0 is a process dependent constant, κ is the gate coupling coefficient and U_T is the thermal voltage. For most CMOS technologies, these parameters take the following values: $\kappa = 0.7$, $U_T = 26$ mV and I_0 has a value below 1 pA for NMOS transistor and below 10 fA for PMOS transistor [42]. Contrary to the strong inversion mode in the weak inversion

mode, the drain current I_D of the MOS transistor does not depend on the gate voltage in relation to the source potential (V_{GS}) and the drain voltage in relation to the source potential (V_{DS}), but directly on the potentials in the nodes: source (V_S), drain (V_D), gate (V_G) and in the case of of the PMOS transistor on the well potential (V_W). The exponential dependence of the drain current on the D, S, G, W potentials of the transistor results in an increase in the steepness of the drain characteristic in the triode region and a complete flattening of the drain characteristic in the saturation region. This makes it easier to drive current-mode circuits whose operating principle is based on maintaining the polarization of the transistors in the saturation area. The implementation presented in this paper is largely based on the use of current mirrors operating in a saturated mode. Additionally, it is worth emphasizing that in the weak inversion mode, the figure of merit described by the Equation (3) and understood as the relationship between the maximum frequency f_{MAX} and the maximum power consumed P_{MAX} , is more favorable than in the strong inversion mode.

$$FoM = \frac{f_{MAX}}{P_{MAX}} \tag{3}$$

This means that as the supply voltage is lowered, the power consumed decreases faster than the maximum processing frequency. This is another argument for using weak inversion mode in a data processing task close to its source. The FoM parameter for the structures used in this work is listed at the end of the current section.

To implement a neural network as a VLSI circuit, the following three circuits were used: a programmable multiplier described in paper [43] which implements the neuron weight, shown in Figure 1, a circuit described in paper [42] which implements a nonlinear neuron activation function, shown in Figure 2, and a circuit for removing the concurrent component (CMRR) described in paper [43] and shown in Figure 3.



Figure 1. Circuit that implements the weight of the neuron [43].



Figure 2. Circuit that implements the activation function of the neuron [42].



Figure 3. Circuit removing the concurrent component [43].

The multiplier shown in Figure 1 consists of two series-connected six-output current mirrors controlled by twelve keys. The output current depends on setting the keys and its value is in accordance with Equation (4).

$$OUT = IN \cdot \sum_{i=1}^{6} B_{1i} w_{1i} \cdot \sum_{j=1}^{6} B_{2j} w_{2j}$$
(4)

Parameters of scaling factors w1i, w2j are selected during the design stage of the multiplier IPcore. Their value depends on the ratio of the device transconductance of transistors at the input stage and at the *i*- or *j*- output stage according to the principle described in more detail in paper [43]. At the implementation stage of the neural network using IPCores, *B*1, *B*2 bit words are selected so that the multiplier implements the neuron weight learned using the framework, which is described in detail in Section 4.3.

As for the hardware implementation of nonlinear neuron activation function, the most commonly used method is to estimate the function using a linear function continuous on intervals. While this implementation is generally satisfactory with respect to the fidelity of reconstructing the hyperbolic activation function, it appears to be relatively expensive. This implementation uses the circuit shown in Figure 2 designed and described by authors in paper [42]. The advantage of the circuit is its simple architecture based on only six transistors. However, the disadvantage is the need to use a dedicated activation function in the learning process. The activation function form is shown in Section 4.2, in which limitations resulting of using the function are discussed in greater detail.

The last circuit used for the implementation is the schematic shown in Figure 3 used to remove the CMRR (Common Mode Rejection Ratio) component described in paper [43]. Its structure is based on a circuit of three two-output current mirrors with scaling coefficients of 1 or 0.5. The principle of operation of the circuit is to determine the value of the common mode component and then to subtract it from the useful signal. This is done both for the non-negated signal and the negated signal simultaneously.

There are some limitations in using these IPcores. The structure of the final preprocessor must contain classic current mirrors with scaling factors of 1, whose task is to duplicate current signals in the perceptron. The circuit implementing weights introduces limitations concerning the permissible dispersion of weights in the network, while the circuit implementing the activation function imposes the need to use a dedicated activation function. The procedure of training the network with the above limitations will be described in more detail in the next chapter.

At the end of the description of the structures used, let us analyze their FoM parameter from the Equation (3). Figure 4b shows the dependence of the parameter on the supply voltage VDD for the circuit from Figure 4a, i.e., implementing a single neuron weight in a balanced structure, with removal of the common mode component. The analysis uses a mirror programmed to implement the highest weight, which means the activity of all output stages in the mirrors. The FoM parameter achieves the best value, i.e., the highest value, for the 0.3 V supply voltage. This is the supply voltage that was used to implement the perceptron.



Figure 4. FoM parameter: (a) the analyzed circuit, (b) dependence of FoM on the supply voltage VDD.

4. Network Structure

4.1. Dataset Description

Data used to train the neural network consisted of artificially generated time series of currents, resembling CNT sensors' readings from vesicles. The dataset was generated using a script that was previously used in [30] for data generation purposes and is described there in more detail. Overall, our dataset consists of 3600 examples. Generated time series are of 180 ms in length and represent three classes: full fusion, kiss-and-run fusion and no fusion. Class examples are presented in Figure 5.



Figure 5. Examples of generated fusion classes.

Dataset was designed so that vesicle fusions are detected only in specific moments in time as mentioned in [30]. Translations of fusions larger than -10 to +10 ms were treated as no fusion class in addition to no-activity time series depicted in the last plot in Figure 5. Dataset consisted of 600 full fusion examples, 600 kiss-and-run fusion examples and 2400 no fusion examples. The only data preprocessing step that was performed was normalization to <-1, 1> range for the numerical stability of the neural network. It was performed with the formula from Equation (5). Constant value 325 in the equation is the largest possible current value in the dataset.

$$vec = vec/325 \times 2 - 1 \tag{5}$$

4.2. Architecture

A 10 ms sampling period of given time series was employed, which resulted in network input data being in the form of vectors with 18 elements. Network structure was designed with simplicity and minimization of resources in mind. Eventually a two-layer 12-2 architecture was settled on. As mentioned before, network's task was a three class classification but due to the minimization of resources, two neurons in the output layer were employed, one representing full fusion and the other kiss-and-run fusion class. These two output neurons technically perform the task of multilabel classification with labels in the form of binary vectors (00, 01, 10, 11). In multilabel classification one input can have multiple classes. We prevented that from happening though by using labels that have at most one class per example (00, 01, 10). Overall this technique allows us to use two output neurons instead of three in the last layer of the network, as it would have to be with a network performing multiclass classification with softmax activation. It is also worth noting that the third class, no fusion, was considered to be detected when both output neurons answered with values below chosen decision thresholds (examples for no fusion were given a 00 label).

Based on [42], all of the neurons have a uniform activation function described by the Equation (6) and Figure 6.

$$\sigma = 0.0560 \times x + 0.9802 \times tanh(0.0029 + (\frac{x}{0.0530})) - 0.00089146$$
(6)



Figure 6. Custom activation function plot.

This is not a classic sigmoidal function. It describes a relation between output and input currents from the circuit presented in Figure 2. This relation is also known as a static characteristic of a circuit. It's parameters were matched based on IPcore simulation results. The main advantage of using this circuit is its size. It is made of only six transistors. The learning process with a dedicated activation function is more difficult, but the complexity of the system is ultimately much less. For the same reason, none of the neurons had any biases. This made the learning process a bit more difficult, but it reduced the number of required multipliers in the hardware implementation. Overall, the network contained 240 trainable parameters.

4.3. Learning

The network was trained using Tensorflow Keras framework. The approach to training was unorthodox. Training was performed with sigmoid activation function in the last layer and custom activation from Figure 6 in the first layer. Sigmoid activation returns values from the finite range of <0, 1> unlike the custom activation. Range <0, 1> is needed for crossentropy loss functions and metrics such as AUC (area under the curve) to function

correctly. They require input values (output of the network) to be of finite range of <0, 1>. This approach however, enforced translation of the class decision thresholds back to the custom activation function post-training. This method was only possible due to both activation functions being centered around zero and having similar monotonicity.

Before training, a hard constraint on the weights of the network had to be taken into account. The constraint was that the weights need to reside in two compartments: <-1.136377, -0.02> and <0.02, 1.136377>. This limitation results from the range of the multiplier coefficient realized by the circuit from Figure 1. It is possible to increase this range, but it would have to result in adding additional output stages in the circuit. This is disadvantageous as the subsequent stages would have to have extremely long transistor channels. The authors decided to present an implementation with a small use of the area, for applications in thermoelectric energy harvesting techniques. For training purposes, the weights were limited to the <-1.136377, 1.136377> range. After training, however, those from (-0.02, 0.02) range were manually set to zeros. This approach resulted in a small loss on testing metrics. When it comes to negative weight values, guaranteeing them in a balanced structure is not a problem and is limited only to modifying the routing of negated and non-negated signals.

Binary crossentropy was chosen as the loss function and Adam as the optimizer with the learning rate of 0.0001 and batch size equal to 8. In both layers, L2 regularization was employed during training to penalize large weights and promote small non-zero ones. The main metric that we used for measuring the network's performance was ROC AUC (Area under the ROC Curve). It has one advantage over standard accuracy—it does not assume the decision threshold to be static. This means that manual selection of the best threshold after training was needed. ROC AUC is the Riemann sum of the ROC curve (receiver operating characteristic curve). For each class there is a single ROC curve, and the final AUC score is averaged over all binary classes (in our case two classes: full fusion and partial fusion). ROC curve is a plot of TPR (true positive rate) to FPR (false positive rate) calculated for uniformly distributed class decision thresholds from <0,1> range for a given dataset. In our case, the step for subsequent thresholds equals 0.5, resulting in 200 data points on the ROC curve for each class. TPR and FPR equations are shown in Equations (7) and (8). TP (true positive), FN (false negative), TN (true negative) and FP (false positive) stand for the number of classifications of particular type on a given dataset with current decision threshold.

$$TPR = \frac{TP}{TP + FN} \tag{7}$$

$$FPR = \frac{TN}{TN + FP} \tag{8}$$

The dataset was split so that 2400 examples were used during training and 600 examples each were used for validation and testing. Network was trained for 272 epochs. It scored 99.31% ROC AUC on the test set. ROC AUC and loss over epochs are presented in Figures 7 and 8, while weights' histograms over epochs are presented in Figures 9 and 10.

Zeroing weights from (-0.02, 0.02) range caused ROC AUC to drop by approximately 0.2% on the test set. Final ROC curves for full fusion and partial fusion classes are shown in Figures 11 and 12.

The procedure of zeroing weights from the (-0.02, 0.02) compartment also caused some neurons to have all input connections with weights equal to 0 so they were removed. This resulted in network shrinking in size from 12–2 architecture to 8–2. Lastly, sigmoid activation functions in the last layer were changed back to custom activation from Figure 6 and optimal decision thresholds for output layer neurons were calculated. They equalled 1.05 nA for full fusion and 0.9975 nA for kiss-and-run fusion. When a neuron that was assigned a class answers with a current equal or above the threshold, class is considered to be detected. Final accuracy of the network with both class thresholds equalled 96.17%.



Figure 7. ROC AUC over epochs. Orange color indicates training and blue indicates validation.



Figure 8. Loss over epochs. Orange color indicates training and blue indicates validation.



Figure 9. Weights' histograms over training epochs for the first layer.



Figure 10. Weights' histograms over training epochs for the second layer.



Figure 11. ROC curve for full fusion classification on the test set. Ranges of TPR and FPR were limited for visualization purposes.



Figure 12. ROC curve for partial fusion classification on the test set. Ranges of TPR and FPR were limited for visualization purposes.

5. CMOS Classifier Parameters

This section describes the parameters of the CMOS circuit implementing the functionality of the aforementioned perceptron, which is a fusion signal classifier. The circuit is made of the IP cores described in Section 3. To implement the circuit we used the TSMC 65 nm CMOS technology. The process of generating the network structure was partly performed using proprietary EDA (Electronics Design Automation) tools [44]. The final circuit works in a balanced structure, consists of 20 blocks implementing the activation function and 320 reconfigurable multiplier blocks. Additionally, the structure embeds 320 blocks for removing the concurrent component. Due to the use of the current mode, it is also required to use 36 8-output mirrors and 16 2-output mirrors to duplicate the signals connecting the perceptron layers. The simplified connection diagram of the circuits in the preprocessor is shown in Figure 13. The diagram corresponds to the two-layer structure of the classifier and contains in each of the two layers a cascade of four blocks: duplicators, multipliers, CMRR blocks and circuits that perform the activation function. The number of connections in individual preprocessor stages is shown in the diagram in rectangular frames drawn with a dashed line. Each of these connections carries current signals. The entire preprocessor is made up of 52,904 transistors in total.



Figure 13. The structure of the preprocessor with a division into blocks: multipliers, duplicators, CMRR removal circuits and blocks implementing activation functions.

The analyses were performed using the Eldo simulator which is part of the Mentor Graphics software. Figure 14 shows the example response of the implemented CMOS circuit to signals from the electrodes for two fusion cases: full and partial. According to the adopted perceptron learning method, fusion classification is carried out by exceeding the classification threshold. A new common threshold has been established for both types of fusion at 595 pA—which is marked on the output waveforms in Figure 14. In the presented example, for both cases of fusion, only one of the outputs exceeds the classification threshold.

The classifier is supplied with the voltage of 0.3 V. The range of the analyzed input currents is limited to range <-1 nA, 1 nA> and the range of output currents to range <-1.03 nA, 1.03 nA>. The circuit samples fusion signals with a frequency of 875 samples/s. The average power consumption of the circuit is 410 nW, and the maximum power consumption is 416 nW. The FoM parameter from Equation (3) is 2.098 $\frac{1}{nJ}$ based on average power consumption. The area of the active part of the circuit equals 1.429 mm². The estimated area of the thermoelectric cell for human energy harvesting techniques needed to power the preprocessor equals 1.39 mm² with a typical efficiency of $30 \frac{\mu W}{cm^2}$, which is 97% of the active part area. This means that the cell required for the full operation of

the preprocessor is of comparable size to the surface of its topography. Therefore, the above implementation can be classified as a circuit of the implantable chips class operating without an external power source.



Figure 14. Examples of CMOS classifier responses: (a) for full fusion, (b) for partial fusion. I(IN)—input current, $I(OUT_FULL)$ —output current corresponding to full fusion, $I(OUT_PARTIAL)$ —output current corresponding to partial fusion.

The applications presented in the literature provide the power consumed in terms of the number of calculation channels (NC). For example, there was presented an analog multilayer perceptron for a portable electronic nose application with the power consumption per channel equal to 27.65 μ W [45]. In another biologically-inspired approach for pattern recognition, the power consumption per channel was achieved at the level of 17 nW [46]. As the third example, we can give a semiconductor implementation of a network with the WTA (Winner Takes All) mechanism, for which the power consumption per channel equals 18.3 μ W [47]. As for the implementation of the preprocessor described in the current article, the number of channels is 320, which is the number of multipliers. The preprocessor power consumption per number of channels is 1.28 nW, which is at least an order of magnitude lower power consumption compared to similar implementations.

The accuracy of the classifier was verified on a set of 150 full fusion patterns and 150 partial fusion patterns. Sets of negative patterns corresponded to them—150 patterns in each of the sets. The average accuracy of the classification was 93.67%. This means a decrease in the accuracy of the CMOS classifier by only 2.5% compared to the learned model. The avarage precision determined on the same set is 96.33%. This is comparable to advanced methods based on deep networks analyzing image sequences, which allow, based on [27], a precision of 95.0% for full fusion and 96.7% for partial fusion, and based on [28], a precision of 95.2% for full fusion and 96.1% for partial fusion. These parameters were obtained using an analog implementation of a very simple two-layer network consisting of 10 neurons.

6. Conclusions

The paper presents a hardware CMOS implementation of a perceptron network algorithm for the task of processing medical data in the form of time waveforms. The approach described in the paper is an example of edge processing, as the data is processed very close to the data source, i.e., immediately after it is obtained from the cells. The work uses techniques to obtain the best possible ratio of data processing speed to power consumed. In particular, the following can be enumerated: using a simple perceptron network as a classifier, the use of simple CMOS structures to implement a perceptron, limiting the dispersion of weights and omitting biases in the network training process and finally the use of the weak inversion mode. Despite the use of a very simple perceptron network architecture the accuracy of the classifier is comparable to much more complex machine learning methods. The preprocessor consumes so little power that no additional power source is required, apart from typical thermal cells of the human energy harvesting class. Both the precision parameters of the classifier itself and the electrical parameters of its hardware implementation fit it into the class of TinyML solutions [48], which are the next step in the development of the Edge-AI concept. Medical data processing applications using implantable chips, due to their low production costs and low power requirements, can improve the level of medical care in many countries with enormous challenges in terms of connectivity, energy and cost [49]. The results of the work can be used in the early diagnosis of precancerous conditions thanks to the possibility of tracking life processes in cells. The approach described in the paper is a consensus between computational efficiency and consumed power. It increases the patient's comfort thanks to the significant reduction of the system dimensions (both the size of the preprocessor and the dimensions of the power source). This approach will certainly work well in various wearable devices applications, especially in combination with the sensor fusion concept. This is a further area of the authors' research.

Author Contributions: Conceptualization, S.S. and P.P.; methodology, S.S. and P.P.; software, M.N. and P.P.; validation, M.N., P.P. and D.H.; formal analysis, S.S. and D.H.; investigation, P.P. and M.N.; resources, P.P.; data curation, P.P.; writing—original draft preparation, P.P. and S.S.; writing—review and editing, D.H.; visualization, P.P.; supervision, S.S.; project administration, S.S. and D.H.; funding acquisition, D.H. and S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Statutory Activities No. 0311/SBAD/0708 and 0311/SBAD/0714 of the Faculty of Computing and Telecommunications at the Poznan University of Technology in Poland.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Edge-AI	Edge Artificial Intelligence
AFE	Analog Front-End
AUC	Area Under the Curve
CMOS	Complementary Metal-Oxide-Semiconductor
CMRR	Common Mode Rejection Ratio
CNT	Carbon Nanotube
DAC	Digital-to-Analog Converter
EDA	Electronic Design Automation
FoM	Figure of Merit
HCNN	Hierarchical Convolutional Neural Network
IP	Intellectual Property
NC	Number of Channels
ROC	Receiver Operating Characteristic
TinyML	Tiny Machine Learning
TIRF	Total Internal Reflection Fluorescence
TSMC	Taiwan Semiconductor Manufacturing Company
VLSI	Very Large-Scale Integration
WTA	Winner Takes All

References

- 1. van der Meulen, R.; Gartner Research. What Edge Computing Means for Infrastructure and Operations Leaders. 2018. Available online: https://www.gartner.com/smarterwithgartner/what-edge-computing-means-for-infrastructure-and-operationsleaders (accessed on 30 January 2022).
- Weisong, S.; Hui, S.; Jie, C.; Quan, Z.; Wei, L. Edge Computing—An Emerging Computing Model for the Internet of Everything Era. J. Comput. Res. Dev. 2017, 54, 907–924.
- Xu, J.; Palanisamy, B.; Ludwig, H.; Wang, Q. Zenith: Utility-Aware Resource Allocation for Edge Computing. In Proceedings of the 2017 IEEE International Conference on Edge Computing (EDGE), Honolulu, HI, USA, 25–30 June 2017; pp. 47–54.
- 4. Cui, G.; He, Q.; Li, B.; Xia, X.; Chen, F.; Jin, H.; Xiang, Y.; Yang, Y. Efficient Verification of Edge Data Integrity in Edge Computing Environment. *IEEE Trans. Serv. Comput.* 2021. [CrossRef]
- 5. Sun, L.; Jiang, X.; Ren, H.; Guo, Y. Edge-Cloud Computing and Artificial Intelligence in Internet of Medical Things: Architecture, Technology and Application. *IEEE Access* 2020, *8*, 101079–101092. [CrossRef]
- Zida, S.I.; Lin, Y.-D.; Lee, C.L.; Tsai, Y.L. Evaluation of an Intelligent Edge Computing System for the Hospital Intensive Care Unit. In Proceedings of the IEEE 3rd Eurasia Conference on Biomedical Engineering, Healthcare and Sustainability (ECBIOS), Tainan, Taiwan, 28–30 May 2021.
- 7. Javaid, S.; Zeadally, S.; Fahim, H.; He, B. Medical sensors and their integration in Wireless Body Area Networks for Pervasive Healthcare Delivery: A Review. *IEEE Sensors J.* 2022, 22, 3860–3877. [CrossRef]
- Chandrasekhara Reddy, T.; Sirisha, G.; Reddy, A.M. Smart Healthcare Analysis and Therapy for Voice Disorder using Cloud and Edge Computing. In Proceedings of the 4th International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), Mangalore, India, 6–8 September 2018.
- Liu, X.; Zhou, P.; Qiu, T.; Wu, D.O. Blockchain-Enabled Contextual Online Learning Under Local Differential Privacy for Coronary Heart Disease Diagnosis in Mobile Edge Computing, *IEEE J. Biomed. Health Inform.* 2020, 24, 2177–2188. [CrossRef]
- 10. Barua, A.; Dong, C.; Al-Turjman, F.; Yang, X. Edge Computing-Based Localization Technique to Detecting Behavior of Dementia. *IEEE Access* 2020, *8*, 82108–82119. [CrossRef]
- Namee, K.; Panong, N.; Polpinij, J. Integration of IoT, Edge Computing and Cloud Computing for Monitoring and Controlling Automated External Defibrillator Cabinets in Emergency Medical Service. In Proceedings of the 5th International Conference on Information Management (ICIM), Cambridge, UK, 24–27 March 2019.
- 12. Palm, W.; Thompson, C.B. Nutrient acquisition strategies of mammalian cells. Nature 2017, 546, 234-242. [CrossRef]
- 13. Jou, A.Y.-S.; Pajouhi, H.; Azadegan, R.; Mohammadi, S. A CMOS integrated rectenna for implantable applications. In Proceedings of the IEEE MTT-S International Microwave Symposium (IMS), San Francisco, CA, USA, 22–27 May 2016
- 14. Abden, S.; Azab, E. Multilayer Perceptron Analog Hardware Implementation Using Low Power Operational Transconductance Amplifier. In Proceedings of the 32nd International Conference on Microelectronics (ICM), Aqaba, Jordan, 14–17 December 2020
- Ishiguchi, Y.; Isogai, D.; Osawa, T.; Nakatake, S. A Perceptron Circuit with DAC-Based Multiplier for Sensor Analog Front-Ends. In Proceedings of the New Generation of CAS (NGCAS), Genova, Italy, 6–9 September 2017
- 16. Kumar, P.; Zhu, K.; Gao, X.; Wang, S.D.; Lanza, M.; Thakur, C.S. Hybrid architecture based on two-dimensional memristor crossbar array and CMOS integrated circuit for edge computing. 2D Mater. Appl. 2022, 6, 1–10. [CrossRef]
- 17. Wong, H.P.; Dahari, Z. Human body parts heat energy harvesting using thermoelectric module. In Proceedings of the IEEE Conference on Energy Conversion (CENCON), Johor Bahru, Malaysia, 19–20 October 2015.
- 18. Oh, N.; Park, J.H. Endocytosis and exocytosis of nanoparticles in mammalian cells. Int. J. Nanomed. 2013, 9 (Suppl. 1), 51–63.
- 19. Ivan, A.I. (Ed.) *Exocytosis and Endocytosis*, 2nd ed.; International Institute of Anticancer Research; Humana Press : Totowa, NJ, USA ; Springer Science and Business Media: Berlin/Heidelberg, Germany, 2014; Volume 3, p. 435.
- 20. Lucien, F.; Leong, H.S. The role of extracellular vesicles in cancer microenvironment and metastasis: Myths and challenges. *Biochem. Soc. Trans.* 2019, 47, 273–280. [CrossRef] [PubMed]
- Zoltowska, K.M.; Maesako, M.; Lushnikova, I.; Takeda, S.; Keller, L.J.; Skibo, G.; Hyman, B.T.; Berezovska, O. Dynamic presenilin 1 and synaptotagmin 1 interaction modulates exocytosis and amyloid β production. *Mol. Neurodegener.* 2017, 12, 15. [CrossRef] [PubMed]
- Michels, A.; Albánez, S.; Mewburn, J.; Nesbitt, K.; Gould, T.J.; Liaw, P.C.; James, P.D.; Swystun, L.L.; Lillicrap, D. Histones link inflammation and thrombosis through the induction of Weibel–Palade body exocytosis. J. Thromb. Haemost. 2016, 14, 2274–2286. [CrossRef] [PubMed]
- 23. Liang, K.; Wei, L.; Chen, L. Exocytosis, Endocytosis, and Their Coupling in Excitable Cells. *Front. Mol. Neurosci.* 2017, 10, 109. doi:10.3389/fnmol.2017.00109 [CrossRef]
- 24. Ahmed, K.A.; Xiang, J. Mechanisms of cellular communication through intercellular protein transfer. *Cell. Mol. Med.* 2011, 15, 1458–1473. doi: 10.3389/fnmol.2017.00109 [CrossRef] [PubMed]
- 25. Ren, L.; Mellander, L.J.; Keighron, J.; Cans, An.; Kurczy, M.E.; Svir, I.; Oleinick, A.; Amatore, C.; Ewing, A.G. The evidence for open and closed exocytosisas the primary release mechanism. *Q. Rev. Biophys.* **2016**, *49*, e12. [CrossRef]
- 26. Schneckenburger, H. Total internal reflection fluorescence microscopy: technical innovations and novel applications. *Curr. Opin. Cell Biol.* **2005**, *16*, 13–18. [CrossRef]
- 27. H. Li et al., A deep learning framework for automated vesicle fusion detection. In Proceedings of the IEEE International Symposium on Biomedical Imaging, Melbourne, Australia, 18–21 April 2017.

- Li, H.; Mao, Y.; Yin, Z.; Xu, Y. A hierarchical convolutional neural network for vesicle fusion event classification. *Comput. Med. Imaging Graph.* 2017, 60, 22–34. [CrossRef]
- Amine, A.; Mohammadi, H.; Amperometry, H. Encyclopedia of Analytical Science, 3rd ed.; Elsevier: New York, NY, USA, 2019; pp. 85–98.
- 30. Szczęsny, S.; Pietrzak, P. Exocytotic vesicle fusion classification for early disease diagnosis using a mobile GPU microsystem. *Neural Comput. Appl.* **2022**, *34*, 4843–4854. [CrossRef]
- 31. Fan, S.; Liang, W.; Dang, H.; Franklin, N.; Tombler, T.; Chapline, M.; Dai, H. Carbon nanotube arrays on silicon substrates and their possible application. *Physica E* 2000, *8*, 179–183. [CrossRef]
- 32. Li, J.; Meyyappan, M.; Cassell, A.M. Biochemical Sensors Using Carbon Nanotube Arrays. U.S. Patent 7,939,734, 10 May 2011.
- Biochemical Sensors Using Carbon Nanotube Arrays, Available online: https://technology.nasa.gov/patent/TOP2-104 (accessed on 17 January 2022).
- 34. Fathail, H.; Cans, A.S. Amperometry methods for monitoring vesicular quantal size and regulation of exocytosis release. *Pflug. Arch.* **2018**, *470*, 125–134. [CrossRef] [PubMed]
- Chen, Y.C.; Lu, S.Y.; Tsai, J.H.; Liao, Y.T. A Power-Efficient, Bi-Directional Readout Interface Circuit for Cyclic-Voltammetry Electrochemical Sensors. In Proceedings of the 2019 International Symposium on VLSI Design, Automation and Test (VLSI-DAT), Hsinchu, Taiwan, 22–25 April 2019.
- Handkiewicz, A.; Szczesny, S.; Kropidłowski, M. Over rail-to-rail fully differential voltage-to-current converters for nm scale CMOS technology. *Analog. Integr. Circuits Signal Process.* 2018, 94, 139–146. [CrossRef]
- 37. Harrison, R. MOSFET Operation in Weak and Moderate Inversion; EE5720; University of Utah: Salt Lake City, UT, USA, 2014.
- Szczęsny, S.; Kropidłowksi, M.; Naumowicz, M. 0.50-V Ultra-Low-Power ΣΔ Modulator for Sub-nA Signal Sensing in Amperometry. *IEEE Sensors J.* 2020, 20, 5733–5740. [CrossRef]
- Cracan, A.; Bonteanu, G.; Bozomitu, R.G. A Weak-Inversion CMOS Analog Multiplier/Divider Circuit. In Proceedings of the IEEE 24th International Symposium for Design and Technology in Electronic Packaging (SIITME), Iasi, Romania, 25–28 October 2018.
- Aiyappa, B.N.; Madhusudan, M.; Yashaswini, B.; Yatish, R.; Nithin, M. Amplifier design in weak inversion and strong inversion— A case study. In Proceedings of the International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 6–8 April 2017.
- 41. Fan, H.; Lei, P.; Yang, J.; Feng, Q.; Wei, Q.; Su, H.; Wang, G. A high-efficient dynamic comparator with low-offset in weak inversion region. *Analog. Integr. Circuits Signal Process.* **2021**, *110*, 175–183. [CrossRef]
- Szczęsny, S. 0.3 V 2.5 nW per Channel Current-Mode CMOS Perceptron for Biomedical Signal Processing in Amperometry. *IEEE Sensos J.* 2017, 17, 5399–5409. [CrossRef]
- 43. Szczesny, S. High speed and low sensitive current-mode CMOS perceptron. Microelectron. Eng. 2016, 165, 41–51. [CrossRef]
- Szczęsny, S. HDL-Based Synthesis System with Debugger for Current-Mode FPAA. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 2018, 37, 915–926. [CrossRef]
- 45. Pan, H.C.; Hsieh, H.Y.; Tang, K.T. An analog multilayer perceptron neural network for a portable electronic nose. *Sensors* **2013**, *13*, 193–207. [CrossRef]
- Rasouli, M.; Yi, C.; Basu, A.; Thakor, N.V.; Kukreja, S. Spike-based tactile pattern recognition using an extreme learning machine. In Proceedings of the 2015 IEEE Biomedical Circuits and Systems Conference (BioCAS), Atlanta, GA, USA, 22–24 October 2015; pp. 1–4.
- 47. Talaśka, T.; Kolasa, M.; Długosz, R.; Pedrycz, W. Analog programmable distance calculation circuit for winner takes all neural network realized in the CMOS technology. *IEEE Trans. Neural Netw. Learn. Syst.* 2016, 27, 661–673. [CrossRef]
- 48. Shafique, M.; Theocharides, T.; Reddy, V.J.; Murmann, B. TinyML: Current Progress, Research Challenges, and Future Roadmap. In Proceedings of the 58th ACM/IEEE Design Automation Conference (DAC), San Francisco, CA, USA, 5–9 December 2021.
- 49. Ooko, S.O.; Ogore, M.M.; Nsenga, J.; Zennaro, M. TinyML in Africa: Opportunities and Challenges. In Proceedings of the IEEE Globecom Workshops (GC Wkshps), Madrid, Spain, 7–11 December 2021.