UNIWERSYTET TECHNOLOGICZNO-PRZYRODNICZY
IM. JANA I JEDRZEJA SNIADECKICH W BYDGOSZCZY
ZESZYTY NAUKOWE NR 256
TELEKOMUNIKACJA I ELEKTRONIKA 13 (2010) 67-78

COMPARISON OF DIFFERENT HARDWARE REALIZATIONS OF THE WINNER TAKES ALL NEURAL NETWORK

Tomasz Talaśka[1], Paweł Przedwojski[2], Jakub Dalecki[3], Rafał Długosz[1]

[1] Faculty of Telecommunication and Electrical Engineering
University of Technology and Life Sciences (UTP),
Kaliskiego 7, 85-789 Bydgoszcz, Poland
[2] Tieto Poland Sp. z o.o.
Grodzka 20, 70-560 Szczecin, Poland
[3] STAR- Projekt Lech Starczak
Przemysłowa 8, 85-758 Bydgoszcz, Poland
talaska@utp.edu.pl, pawel.przedwojski@wp.pl, dalecki@o2.pl, rafal.dlugosz@gmail.com

Summary: This paper presents realization and the laboratory tests of the Kohonen winner takes all (WTA) neural network (NN) realized on microcontrollers (μC) with the AVR and ARM CortexM3 cores, as well as the comparison with the full custom implementation of analog network of this type in the CMOS technology. The two μCs have been placed on a single testing board to facilitate the comparison. The board allows for switching between the two μCs, it enables selection of either the Euclidean (L2) or the Manhattan (L1) distance measures. It also allows for turning on/off the so-called conscience mechanism. Some signals illustrating the training of the network can be observed directly on the board. The full learning process with all essential parameters can be viewed on PC using the USB port. The prospective application of the system is in on-line analysis of the ECG and EMG biomedical signals in the health care diagnostic systems, as well as in the student laboratories on neural networks and programmable devices.

Keywords: WTA network, digital neural networks, analog neural networks, microcontroller, low energy consumption

1. INTRODUCTION

Artificial NNs are commonly used in such tasks that require processing and classification of "difficult" signals e.g. non-stationary signals, heuristic data etc. in medical health care, telecommunication, electrical engineering and other application areas. In literature one can find various implementation techniques of NNs both the software- and the hardware-based. Considering such criteria as energy consumption, calculation capacity and device size, *full-custom* designed networks are the most efficient solutions [15]. NNs realized as application specific integrated circuits (ASIC) allow, for example, for parallel data processing, and thus consuming significantly less energy are often faster than the software-based networks. The full-custom style allows for a very good matching of the circuit structure to a given task. A disadvantage of this approach is relatively complex design process and large fabrication cost in case of short series. It is also relatively difficult to built-in such a programmable network of this type.

In this paper the authors present realization of the Kohonen WTA NN [2] using an alternative approach based on two μCs, as well as the comparison with the analog WTA NN previously implemented in the CMOS 0.18 μm technology. For this purpose a special testing board has been developed using the Eagle 5.6 environment. To get a better insight into the parameters of the NN realized in this way, two different μCs have been used i.e. the 8-bits AVR and the 32-bits ARM CortexM3. Such realized testing board allows for training the NN with different parameters i.e. with two different measures (L1 / L2) of similarity between the input training patterns X and the weight vectors W of particular neurons. Both measures are shortly presented in Section III. The network can be trained with or without the so-called conscience mechanism [13, 14, 15]. All these modes can be selected by the manually-operated switches placed directly on the board.

Realizations that involve μCs are relatively cheap, which is one of their important advantages. In comparison with custom-designed networks all parameters can be easily reprogrammed. On the other hand, because of serial data processing, networks of this type are relatively slow and thus are suitable for small networks with up to fifty neurons and sampling frequencies not exceeding 100 kHz. However, many applications still can be indicated, in which such parameters are accepted, e.g. in the analysis and classification of the ECG and EMG biomedical signals sampled at max 2 kHz.

Implementation of neural networks on μCs is significantly cheaper than realization of such networks by use of digital signal processors (DSP) or in field programmable gate array (FPGA). Furthermore, microcontrollers offered on the market today aid floating point operations, while multiplication operations are performed in a single clock cycle. As a result, the general parameters of both platforms are often comparable. It is worth mentioning that very often core blocks of DSPs are the same like those used in microcontrollers (the ARM core).

The paper is organized as follows. State-of-the-art in the field of Kohonen WTA NNs realized using programmable platforms and the analog technique is presented in next section. An overview of the WTA NN principle is provided in Section III. Section IV presents realization of the testing board described above. In Section V selected measurement results are shown together with a discussion of the achieved parameters. The conclusions are covered in Section VI.

2. STATE-OF-THE-ART STUDY

2.1. REALIZATIONS OF THE KOHONEN NEURAL NETWORKS BASED ON MICROCONTROLLERS

An idea of the implementation of the self organizing map (SOM) [2] using SIMD (Single Instruction Multiple Data) processors has been described in [3]. These processors allow for parallel information processing using a single instruction. They find the application in the newest computer systems. The authors of [3] focused on different methods of detection of the winning neuron i.e. on the so called WTA circuits.

Using SIMD processors is beneficial in case of large NNs with more than 100 neurons, in which data rate is one of the key parameters. In case of smaller networks with sampling frequencies not exceeding 100 kHz such realizations become uneconomical. The SIMD processors are more expensive and dissipate more power than the AVR/ARM μCs.

The other reason, important in commercial applications, is that AVR/ARM μCs can be easier programmed and require simpler environment (the printed circuit board).

Another implementation has been reported in [4]. This paper presents realization as well as the optimization of selected arithmetic operations, such as multiplication and *tanh* using feed-forward multi-layer network implemented in PIC18F45J10 μC. The authors of [4] conclude that are able to implement up to 256 neuron weights using this μC. Assuming this network to be a multi-layer architecture, this means that relatively small networks can be realized with the number of neurons not exceeding 50. It is difficult to assess the performance of this network as the achievable data rate, power dissipation and the computational capacity are not provided.

In case of hardware implemented NNs the computational capacity depends on either data processing is performed serially or in parallel, the sampling frequency of the μC core, the number of neurons and several other parameters. An important issue is the complexity of the training algorithm offered by a NN. μCs are rather suitable for such networks that require simple arithmetic operations. For example, the WTA algorithm implemented by the authors requires only multiplications, summations and subtractions. For the comparison the network described in [4] requires *tanh* activation functions. It is worth mentioning that the WTA network is trained without supervision that makes this NN relatively faster than their counterparts trained with the supervision, in which an error function is calculated separately for every neuron.

Different applications of NNs realized on μCs have been reported. Such networks are frequently used in control and diagnostic devices. A device described in [5] has been used as an intelligent wireless electronic nose node (WENN) used in classification and quantification of binary gas mixtures NH3 and H2S. In [6] such implemented NN is used in control of the furnace temperature. One can find the applications in which NNs only cooperate with μCs. In [7] software implemented feed-forward Back Propagation (BP) NN exchange data with the PIC16F84A μC that controls a device detecting damages in textiles produced in the factory.

2.2. REALIZATIONS OF KOHONEN NEURAL NETWORKS AS ASICS

Full-custom developments of Kohonen NNs are not common. Several examples can be found in [15, 16, 17]. They are based on different techniques, such us digital, analog or mixed analog-digital.

A fully digital NN realized in the CMOS 0.5 μm process is reported in [16]. The main disadvantage in this case is a relatively large chip area, which makes the implementation of large networks fairly impractical. Each processing element (PE), representing a single neuron contains about 10.000 transistors and occupies an area of 4 mm^2.

Another, mixed analog-digital, implementation has been reported in [17]. In this case some modules of the overall architecture like the distance calculation (L2 in this case) and the WTA blocks are implemented as analog components, while the adaptation process is realized by the use of digital blocks. The memory for the weight storage is implemented as digital counters that can count in both directions, depending on whether an input data x is greater than a neuron's weight w or not. The adaptation mechanism used in this solution differs from the classic algorithm proposed by Kohonen. The learning rate, η, is kept fixed, while its value results from the assumed resolution of the counter (5 bits in this particular case). Moreover, unlike the classic Kohonen's algorithm in

which adaptation process depends on exact values of x and w, in [17] it depends on the value of the sign function, sign(x, w), always updating the counter value by ± 1.

A fully analog WTA NN has been implemented by the authors of this paper in the CMOS 0.18 μm technology [15]. In this implementation all main components, including the adaptation mechanism, the Euclidean distance calculation block, as well as the winning neuron detection block (WTA) ware realized using analog circuitry. A full design cycle of this network, which started with extensive system level simulations performed in the software model realized in the C++ environment, covered also transistor level simulations, full-custom layout design and finally the chip fabrication and experimental verification. The prototype chip has been realized in Canada in TSMC CMOS 0.18 μm process. Layout of this prototype is shown in Figure 1.

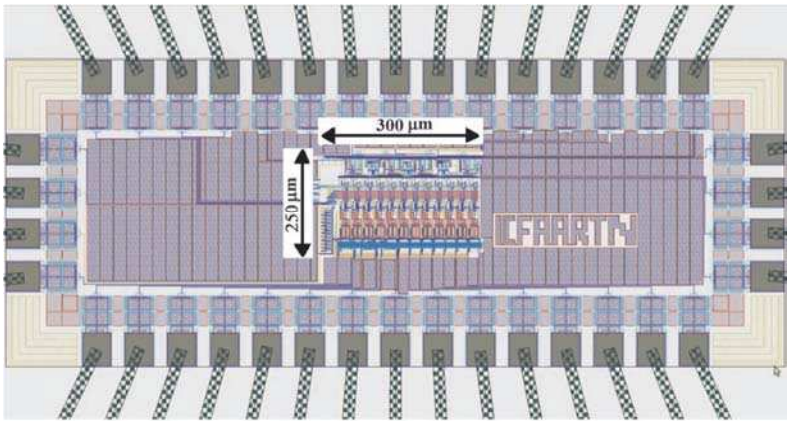


Fig. 1. Layout of the prototype analog WTA NN implemented by the authors in the TSMC CMOS 0.18 μm technology

In measurements the network with 12 channels working in parallel, sampled with 2 MHz clock was able to operate as fast as a standard 2 GHz PC, consuming only 700 μW of power i.e. almost 100 thousands times less than the PC. As this network was a first prototype, only four neurons have been implemented. In case if fifty neurons were realized in a single chip with ten analog inputs, the power dissipation would not exceed 10 mW, while the chip area would equal about 1 mm^2. Operating at 2 MHz the network would realize 8e09 operations/s. This estimation shows that the analog network realized as ASIC surpasses the networks implemented using the programmable platforms.

3. LEARNING PROCESS IN THE WTA NEURAL NETWORK

The feed-forward network that is in the scope of interests of this paper has been originally proposed by Teuvo Kohonen in [2]. This network features a competitive unsupervised learning. The weights of neurons are modified without any feedback that makes the learning algorithm very fast. Such networks are suitable for the applications, in which data rate is a key parameter e.g. in telecommunications [8-11]. Two types of this network are often distinguished. In this paper the WTA learning method is implemented that is a special case of the winner takes most (WTM) learning algorithm for the

neighborhood range equal to zero. The WTA algorithm is less complex than the WTM one and therefore is much simpler in the hardware implementation.

The learning in Kohonen NNs (KNN) relies on presenting the network with learning patterns, X, in order to make the neurons' weight vectors, W, resemble presented data. For each pattern X the network first determines the distance between this vector and the W vector of each neuron. Different measures of the similarity between these vectors are available. One of them is the Euclidean distance defined as follows:

$$d(X,W_i)=\sqrt{\sum_{l=1}^{n}(x_l-w_{il})^2} \tag{1}$$

In this work a modified Euclidean measure has been used, in which the rooting operation has been neglected. The results in both cases are the same, as if $a < b$ then always $\sqrt{a}<\sqrt{b}$, while the rooting operation is more difficult in the hardware realization. The Euclidean measure can be denoted as L2. Another frequently used measure, called the "Manhattan" one and denoted as L1, is defined as:

$$d(X,W_i)=\sum_{l=1}^{n} abs\left|x_l-w_{il}\right| \tag{2}$$

In this case the squaring operations have also been neglected, which enables further simplification of the learning algorithm. Both these measures have been implemented by the authors in the μC and compared with respect to the calculation complexity.

The adaptation of the winning neuron is in the WTA NN performed in accordance with the following formula:

$$W_i(t+1)=W_i(t)+\eta\cdot\left(X(t)-W_i(t)\right) \tag{3}$$

where η is the learning rate. Other neurons in the network that lose the competition remain unchanged in this algorithm.

One of the significant problems encountered in the WTA networks are the, so-called, dead neurons i.e. the neurons that take part in the competition but never win and therefore their weights remain unchanged. One of the reasons of this are badly selected initial values of the weights [12]. Such neurons reduce the number of classes that can be discriminated, thus increasing the mapping (quantization) error of the network. For this reason reducing the number of these neurons is an important design objective. One of the efficient methods in this task is by use of the, so-called, conscience mechanism [13, 14]. Its role is to increase the likelihood of winning for all neurons in the network.

The conscience mechanism proposed earlier by the authors and implemented in their analog NN has been also used in the network realized in both μCs. In this case, the real distance between the weight and the training vectors is made higher by adding a signal that is proportional to the number of the wins:

$$d_{\text{cons}}(X,W)=d_{\text{L1/L2 norm}}(X,W)+L_{\text{count}}\cdot K \tag{4}$$

where $d_{\text{L1/L2}}(X, W)$ is the real distance determined by use of either the L1 or the L2 metric, $d_{\text{cons}}(X, W)$ is a signal modified by the conscience mechanism and then applied to the WTA block. The L_{count} is the number of the wins of a given neuron. The K coeffi-

cient is the gain factor that allows for controlling and optimizing the learning process by adjusting the strength of the conscience mechanism. The quantization error mentioned above is defined as:

$$Q_E = \frac{1}{z} \cdot \sum_{iter=1}^{Z} \left\| X(t) - W_j(t) \right\|^2 \tag{5}$$

where Z is the number of the iterations in each epoch, i.e. the number of all training patterns X in a given input data set. The j index indicates the winning neuron.

4. TESTING BOARD WITH THE WTA NEURAL NETWORKIMPLEMENTED ON MICROCONTROLLERS

The realized testing board with both μCs, shown schematically in Fig. 2, has been designed in the Eagle 5.6 environment. The board is composed of several blocks. One of them is the interface block containing the ADC/DAC standard chips that convert analog learning signals X to digital form used then by the μCs, as well as the neuron weights to analog form for the observation. The 'Switching field' allows for selecting one of the two μCs, as well as the distance measure (L1/L2), as shown in Fig. 3.

The conscience mechanism can be turned on/off as well. The proposed system due to manifold of different options can be used both in commercial application and in education. Further development possibilities of this system still exist.

The neuron weights can be observed on-line either as the analog signals at the output of the DAC blocks or directly as the digital signals. The analog signals allow for a rough verification of the learning process on the oscilloscopes. Digital signals can be acquired on PC by use of the USB and RS232 serial ports, for further off-line analysis.

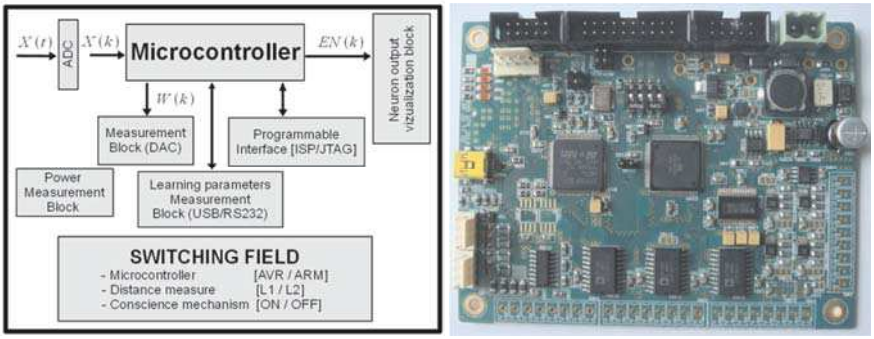


Fig. 2. The proposed testing board of the programmable WTA neural network based on microcontrollers

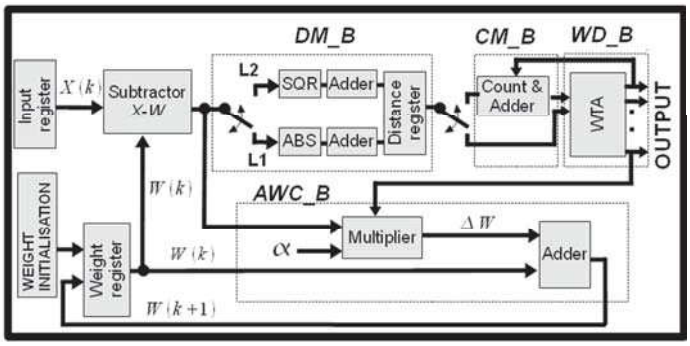


Fig. 3. WTA learning process implemented on μC: DM_B is a distance measurement block, CM_B – the conscience mechanism block, WD_B – the winning neuron detecting block (WTA), AWC_B – weight adaptation block

The board has been designed in such a way to enable measuring on-line the power dissipation, separately for the ADC / DAC blocks and for both μCs. The μCs are programmed by the use of the ISP / JTAG interfaces. The serial ports also allow for acquiring on PC the learning patterns X (as digital signal samples), the calculated distances between the X and the W vectors, the numbers of the wins and the quantization error for detailed analysis of the network performance. The board allows for a full observation of the network with 3 inputs and 4 outputs i.e. 12 neuron weights. To make it possible, a single 4-channel THS1206 ADC and three AD7305 4-channel DACs have been used. The number of neurons can be increased but in this case only selected weights can be observed on-line on the oscilloscopes.

One of the reasons of selecting in this prototype only 12 weights for direct observation was to facilitate a direct comparison with the analog network previously designed by the authors with just 3 inputs and 4 outputs. For a better comparison between both μCs the X and the W signals have in this approach the resolution of 8 bits. This resolution is sufficient in many practical applications. The float type on the ARM μC has not been used in this case, although such a possibility still exists.

5. LABORATORY TESTS OF THE REALIZED WTA NETWORK

The maximum achievable data rate of the network in case of the implementation on μCs depends on the number of the inputs and the outputs, as shown in Fig. 4 for both μCs. The results are present for different parameters of the learning process. The learning speed can be determined using the following formula:

$$f_{\text{data}}(n) = \frac{f_{\max}}{NoC \cdot n} \tag{6}$$

where $f_{\max}$ is the maximum clock frequency of the μC equal 16 MHz for the AVR and 72 MHz for the ARM. NoC is the number of all cycles of the μC required for processing a single vector X in the network with n neurons. Note that the number of the clock cycles in the AVR and the ARM μCs required to complete the same task may differ. The ARM μCs are more efficient, so although $f_{\max}$ is in this case 5 times larger, the network is more than 7 times faster than in case of the AVR μC.

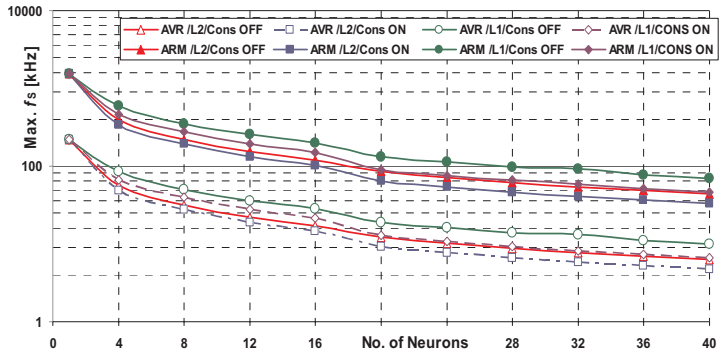


Fig. 4. Achievable maximum data rate of the WTA NN vs. number of neurons (for 3 inputs) for both microcontrollers

The power dissipation of particular system components is shown in Fig. 5. This parameter has been measured for a full performance of the µCs, when the network operates at the highest possible data rate. Comparison of the results for the L1/L2 metrics shows that in the first case the achievable data rate is almost doubled for the same number of neurons, while the learning accuracy is maintained. This shows that elimination of the multipliers at least at the stage of the distance calculation is a good idea.

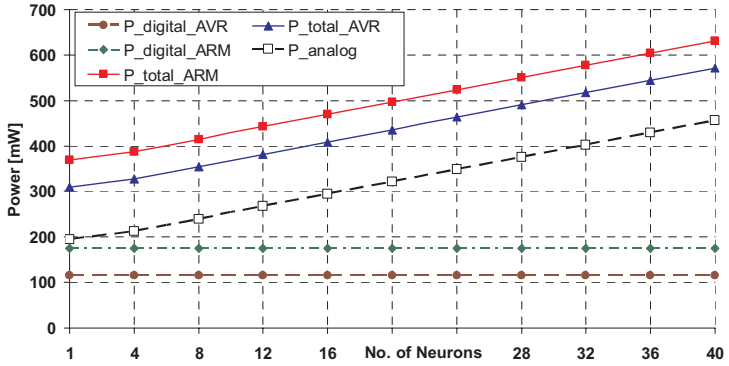
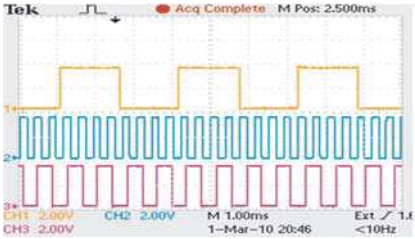
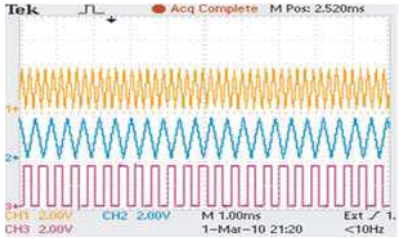


Fig. 5. Estimated power dissipation of the WTA NN vs. the number of neurons for 3 inputs for both used microcontrollers

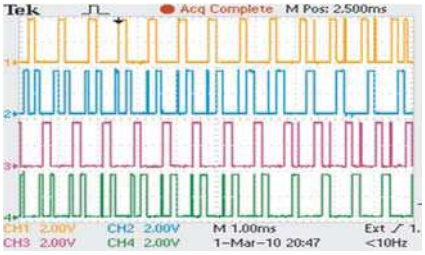
Figure 6 presents selected measurement results of the network with 3 inputs and 4 outputs realized using the ARM µC sampled at 400 kHz. The results are shown for two different settings of the learning process i.e. for the conscience mechanism being turned on (left) and off (right). When this mechanism was turned-off one of the neurons remained inactive for presented input data.



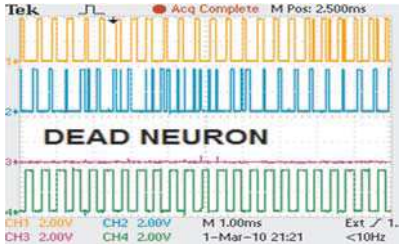
Learning signals (X) – Example 1



Learning signals (X) – Example 2



Neurons output signals – Example 1



Neurons output signals – Example 2

Fig. 6. Selected measurement results of the WTA NN with 3 inputs and 4 outputs. Example 1: the conscience mechanism (Cons) is on; Example 2: the Cons block is off

One of the objectives of this paper was the comparison of the implementation based on μC with the earlier analog realization. The analog NN with the same number of weights, sampled at 1 MHz, dissipated power of 700 μW. This is more than 500 times less than in case of the realization on the μCs. Taking into account also the sampling frequency, it can be demonstrated that the analog network can be even 1000 times more efficient. The main parameters of particular implementations have been collected in Table 1.

Table 1. Comparison between different realizations of WTA NN

	AVR/L1	AVR/L2	ARM/L1	ARM/L2	Analog/L2	PC
P [mW]	310	310	390	390	0.7	2e04
f_S_max [kHz]	71	48	480	320	1000	1800
FOM (f_S /P)	0.23	0.15	1.23	0.82	1428	0.09
device sizes	3x4 cm (board with one μC)				1 x 1.5 mm	Large
Price (€)	40 (with one μC)				5 (long series)	c. 300

The results presented for the PC are estimated on the basis of the simulation results of the network model implemented in C++. If the NN is realized as ASIC or on the μC, as opposed to the PC based realizations, the power dissipation can be controlled in such a way to be approximately linearly dependent on the sampling frequency. This feature is an very important advantage, especially in the wearable systems, for example in Wireless Sensor Networks (WSN), which operate in environments with limited access to energy sources [18].

6. SUMMARY

Two different hardware realizations of the Winner Takes All (WTA) Artificial Neural Network (NN) have been compared in the paper. One of them is an analog network designed earlier by the authors in "full-custom" style in the CMOS 180 nm technology. The second implementation, proposed and described in this paper is based on two microcontrollers with the AVR and the ARM cores.

Both microcontrollers are placed on a single testing board, together with the ADC/DAC blocks, the power supply block. This makes the board a fully autonomous system, with the built-in learning abilities.

The measurement results of the prototype devices show that the μC-based implementation is even ten times more efficient than the PC-based realizations, considering such criteria as the achievable data rate vs. power dissipation. On the other hand the analog networks is thousand times more efficient than the μC-based network, occupying the area less than 1 mm^2.

The main disadvantage of the full-custom, transistor level realization is relatively long design process and high fabrication costs in case of short series. The μC-based realization offers a low cost of the device, medium sizes and large flexibility. If only one μC will be used, e.g. with the ARM core, the cost will not exceed 40 Euro per piece. The overall device sizes of 3 x 4 cm, achievable in this case, make the proposed system suitable for various portable applications, including medical diagnostic tools.

The WTA neural network presented in this paper is going to be used as a base station in wireless body sensor network for the on-line analysis of various biomedical data. To enable such option the final device will be equipped with the filters for data preprocessing and the wireless communication module to enable communication with the low power sensors placed on the patient's body. The platform is still being developed.

BIBLIOGRAPHY

[1] Długosz R., Talaśka T., Dalecki J., Wojtyna R., 2008. Experimental Kohonen neural network implemented in CMOS 0.18 μm technology, International Conference on Mixed Design of Integrated Circuits and Systems, pp. 243-248, Poland.
[2] Kohonen T., 2001. Self-Organizing Maps, Springer Verlag, Berlin.
[3] Mailachalama B., Srikanthan T., 2002. Area–time issues in the VLSI implementation of self organizing map neural networks, Microprocessors and Microsystems, Vol. 26, No. 9-10, pp. 399-406.
[4] Cotton N.J., Wilamowski B.M., Dundar G., 2008. A Neural Network Implementation on an Inexpensive Eight Bit Microcontroller, International Conference on Intelligent Engineering Systems, USA, pp. 109-114.
[5] Young Wung Kim, Jung Hwan Cho, Gi Joon Jeon, 2007. An Intelligent Wireless Electronic Nose Node for Monitoring Gas Mixtures Using Neuro-Fuzzy Networks Implemented on a Microcontroller, IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, Italy, pp.100-104.
[6] Mousa H.M., El-Bardini M.A., Koutb M.A., 2005. On-Line Neurocontroller Based on Microcontrollers, IEEE International Conference on Industrial Technology, Hong-Kong, pp. 1252-1256.

[7] Mursalin T.E., Eishita F.Z., Islam A.R., 2008. Fabric defect inspection system using neural network and microcontroller, Journal of Theoretical and Applied Information Technology, Vol. 4, No. 7.
[8] Amerijckx C., Verleysen M., Thissen P., Legat J., 1998. Image Compression by Self-Organized Kohonen Map, IEEE Transactions on Neural Networks, Vol. 9, No. 3, pp. 503-507.
[9] Chang C., Xu P., Rui Xiao, Srikanthan T., 2005. New Adaptive Color Quantization Method Based on Self-Organizing Maps, IEEE Transactions on Neural Networks, Vol. 16, No. 1, pp. 237-249.
[10] Pedrycz W., Vasilakos A., 2001. Computional intelligence in telecommunications networks, CRC Press, LLC.
[11] Kohonen T., Somervuo P., 1998. Self-Organizing Maps of Symbol Strings with Application to Speech Recognition, Neurocomputing, Vol. 21, No. 1-3, pp. 19-30.
[12] Talaśka T., Długosz R., 2008. Initialization mechanism in Kohonen neural network implemented in CMOS technology, European Symposium on Artificial Neural Networks (ESANN), Belgium, pp. 337-342.
[13] Ahalt S.C., Krishnamurthy A.K., Chen P., Melton D.E., 1990. Competitive learning algorithms for vector quantization, Neural Networks, Vol. 3, pp. 131-134.
[14] DeSieno D., 1988. Adding a conscience to competitive learning, IEEE Conference on Neural Network, Vol. 1, pp. 117-124.
[15] Długosz R., Talaśka T., Pedrycz W., Wojtyna R., 2010. Realization of a Conscience Mechanism in CMOS Implementation of Winner Takes All Neural Networks, IEEE Transactions on Neural Networks, Vol. 21, No. 6, pp. 961–971.
[16] Rajah A., Khalil Hani M., 2004. ASIC design of a Kohonen Neural Network microchip, IEEE International Conference on Semiconductor Electronics, Kuala Lumpur, Malaysia, pp. 148-151.
[17] Chung-Yu Wu., Kuo Wen-Kai, 1993. A new analog implementation of the Kohonen Neural Network, International Symposium on VLSI Technology, Systems, and Applications, Taipei, Taiwan, pp. 262-266.
[18] Dubois P., Botteron C., Mitev V., Menon C., Farine P.-A., Dainesi P., Ionescu A., Shea H., 2009. Ad hoc wireless sensor networks for exploration of Solar-system bodies, Acta Astronautica, 64 (5-6), pp. 626-643.

PORÓWNANIE RÓŻNYCH SPRZĘTOWYCH REALIZACJI SZTUCZNEJ SIECI NEURONOWEJ TYPU WINNER TAKES ALL

Streszczenie

W pracy przedstawiono projekt oraz wyniki badań laboratoryjnych sieci neuronowej Kohonena typu Winner Takes All (WTA) zaimplementowanej na mikrokontrolerach z rdzeniami AVR oraz ARM. W pracy przedstawiono też porównanie z wcześniejszą realizacją podobnej sieci jako specjalizowany analogowy układ scalony. Dwa mikrokontrolery, na których zaimplementowano algorytm uczący umieszczone zostały na jednej płytce testowej aby umożliwić bezpośrednie porównanie ich parametrów. Za pomocą przełączników umieszczonych bezpośrednio na płytce możliwe jest wybranie jednego z mikrokontrolerów, jednej z dwóch miar podobieństwa między wektorami (Euklidesa L2 lub typu Manhattan L1) oraz włączenie lub wyłączenie mechanizmu sumienia. Niektóre sygnały przedstawiające proces uczenia (sygnału sygnalizującego

zwycięski neuron) możemy bezpośrednio obserwować na płytce. Proces uczenia możemy też w całości obserwować na komputerze PC, poprzez złącze USB. Do potencjalnych zastosowań wykonanej płytki testowej oraz sprzętowych realizacji sieci neuronowej należą systemy do ciągłego monitoringu zdrowia pacjentów (obserwacja oraz analiza sygnałów typu EKG oraz EMG), a także jako wyposażenie laboratorium studenckiego.

Słowa kluczowe: sieć typu WTA, cyfrowe sieci neuronowe, analogowe sieci neuronowe, mikrokontrolery, niski pobór energii