UDC 004.8

Володимир Шимкович, Юрій Бердник

РЕАЛІЗАЦІЯ ТА ДОСЛІДЖЕННЯ НЕЙРОННИХ МЕРЕЖ ПРЯМОГО РОЗПОВСЮДЖЕННЯ НА ПЛІС

IMPLEMENTATION AND RESEARCH OF NEURAL NETWORKS OF DIRECT PROPAGATION ON FPGA

В статті наведені результати реалізації нейронних мереж прямого поширення на ПЛІС за розробленим раніше методом реалізації нелінійних функцій активації та алгоритмами реалізації штучних нейронів. Отримані результати дослідження, а саме використаній ресурс ПЛІС, похибка та швидкодія, представлені в таблиці.

Ключові слова: нейронні мережі, функції активації, ПЛІС, VHDL.

Рис.: 1.Табл.: 1. Бібл.: 16.

This paper presents the results of FPGA implementation of directpropagation neuronal networks for the previously developed method of realization of nonlinear activation functions and artificial neurons implementation algorithms. The obtained results of the research, which are presented in the table, are FPGA source, error and speed.

Key words: neural network, activation function, FPGA, VHDL.

Fig.: 1. Tabl.:1. Bibl.: 16.

Neural network control systems constitute a new high-tech branch of control theory and relate to the class of nonlinear dynamic systems. High speed of parallel processing of information coupled with the ability to teach neural networks makes this technology very attractive for creating effective implementation of automated control systems for complex dynamic objects [1, 2]. Neural networks can be used to construct devices for regulation and correction of control systems of reference, adaptive, nominal and inverse-dynamic object model for observation and evaluation of the object parameters, magnitude of the perturbation, search or computation of the optimal influence change program, object identification, prediction of the object state or another operation [3 - 5]. For realization of all neural network capabilities, the neural network controllers of control systems need to be built on chips such as the FPGA [6-9].

FPGA chips allow implementing complex parallel algorithms. This technology involves the implementation of a plurality of blocks on one crystal with intensive information exchange between them. The implementation of data parallel processing on general-purpose processors or digital signal processors requires considerable effort to build multimicroprocessor systems. But a developer does not encounter any rigid architectural constraints in FPGA and can effectively implement parallel algorithms.

Existing approaches to the digital realization of nonlinear functions use different methods of approximation, such as the tabular method, Taylor series expansions, lump-linear approximation, etc. Taylor series expansion requires many multiplications, and therefore unacceptable for FPGA implementation, because the multiplication block occupies a large amount of resources. The tabular method

involves the creation of a global variable, a table of the target function possible values, unpredictable and uncontrolled access to the tables of all neurons in the network, which in turn creates a large time delay. And the creation of a separate local table for each neural network is inappropriate in terms of the use of the FPGA resource. Therefore, the most optimal method for sigmoidal type activation functions implementing is lump-linear approximation of this function.

Issues related to the implementation of one neuron on an FPGA are considered in [6]. The implementation of 3-input neuron with different activation functions on FPGA Xilinx XCV400hq240 has been investigated. The comparison of performance and occupied neuron resources for different implementation variants is given.

In [7] a prototype of a cascade fragment of a neural network with straight sequential bonds was developed. The prototype is implemented on the basis of FPGA Virtex-E XCV400E-pq240. The fragment is intended for parallel input of 8 component input vector, parallel multiplication by the contents of internal blocks of memory of factors, parallel addition of 8 products and parallel activation of the received amounts in each neuron (8th senior bits of sum are given to the address port of memory in which the discrete values of the selected activation function are written). The following result is obtained. The maximum clock speed of the circuit is 90 MHz. Calculation time of the source vector by input is 70 ns.

In work [8] the analysis of the exponential and sigmoid artificial neurons activation functions by means of FPGA (Xilinx Virtex-5 XC5VLX110T) is analyzed. To calculate the values of the sigmoidal function, a table of function values at the reference points is used and a linear interpolation between these points. As a result, the performance and error of the calculations and FPGA resource are presented.

We propose the following approach to the realization of functions of activation of a sigmoid type with the necessary accuracy of its approximation. In the beginning, the activation function is studied for symmetry with respect to the axes. If the condition

$$1-f(x)=f(-x), \tag{1}$$

isfulfilledthen you can consider the function $f(x)$ only for positive arguments, and for negative ones, determine by formula (1), which accelerates the calculation of the function values and reduce the occupied resource of the FPGA.

Further, proceeding from the given accuracy of approximation, we form a piecewise-linear activation function with an appropriate number of intervals, and each of the intervals $(-\infty;x_1),(x_1;x_2);...(x_n;+\infty)$ represent by a separate formula. A general description of a piecewise-linear activation function is written as:

$$f(x)=\begin{cases} k_0x+b_0,\ x<x_1 \\ k_1x+b_1,\ x_1<x<x_2 \\ \cdots \\ k_nx+b_n,\ x_n<x \end{cases}, \tag{2}$$

Next, for each linear function, we calculate the values of the coefficients *k*and *b*and memorize them. Subsequently, for the calculation of the value of the function *f*(*x*)by the x argument from memory, the corresponding values of the coefficients *k*and

*b*are chosen and the value of $f(x)$is calculated according to the corresponding formula. And for the negative values of the argument, the value of the function is calculated by formula 3.

One of the main features of the neural network is the parallel processing of signals. Multilayer neural networks represent a homogeneous computing environment. According to the neuroinformatics terminology, these are universal parallel computing structures, intended for solving various classes of tasks. With the hardware implementation of artificial neural networks on the FPGA, each layer of the network works in parallel with another, which allows using the principle of the conveyor during calculations. Neurons in each layer also work in parallel on the principle of multiprocessor data processing. That is, each artificial neural network is a separate processor and the processing of information in each neuron passes simultaneously.

The results of simulation of neural networks such as used FPGA resource, speed and error are presented in Table 1.

Table 1

Results of neural networks simulation in FPGA

Neural network	Number of synapses	FPGA resource (LUT)	Output ANN value		Error	Speed, ns
			MatLab	FPGA		
1-1-1	3	484	0.56389366	0.546875	0.017019	120.24
1-2-1	5	585	0.62573413	0.609375	0.016359	126.22
1-3-2	10	723	0.68372392	0.671875	0.011849	127.77
1-5-1	11	523	0.78328235	0.765625	0.017657	128.75
2-2-1	8	488	0.63863534	0.625	0.013635	150.48
2-3-1	11	790	0.70144096	0.6875	0.013941	130.58
2-4-2	18	899	0.75747616	0.75	0.007476	133.60
4-4-4	36	1104	0.79223947	0.765625	0.026614	133.53
3-5-1	23	725	0.82549114	0.796875	0.028616	134.05
2-4-1	14	933	0.75747616	0.75	0.007476	133.54
1-3-1	7	723	0.68372392	0.671875	0.011849	127.71
2-1-1	5	602	0.57070375	0.5625	0.008204	126.22
1-4-1	9	829	0.73651211	0.734375	0.002137	127.63
1-6-1	13	1065	0.82373748	0.796875	0.026862	132.26
3-1-1	7	723	0.57708067	0.5625	0.014581	128.71
3-2-1	11	819	0.65058263	0.625	0.025583	134.14
2-8-2	34	1369	0.90702034	0.890625	0.016395	163.24

In the table, the structure of the studied neural networks is represented by three numbers, where the first is the number of neurons in the input layer, the second is the number of neurons in the hidden layer, and the third is the number of neurons in the output layer. The obtained values in LUTs may vary somewhat when the synaptic weights of the neurons change. As a result, an increase in the efficiency of the neural networks implementation on the FPGA is obtained. The increase in the efficiency based on indicators such as the speed and error of the calculations, and the usage of the FPGA resource.

Each neuron is represented as a separate block, as shown in Figure 1, which consists of several parallel processes, and the neural network is a multiprocessor system. The programming language allows you to explicitly specify the signals that start the process. To start the process of calculating, the input signal of this neuron is used.

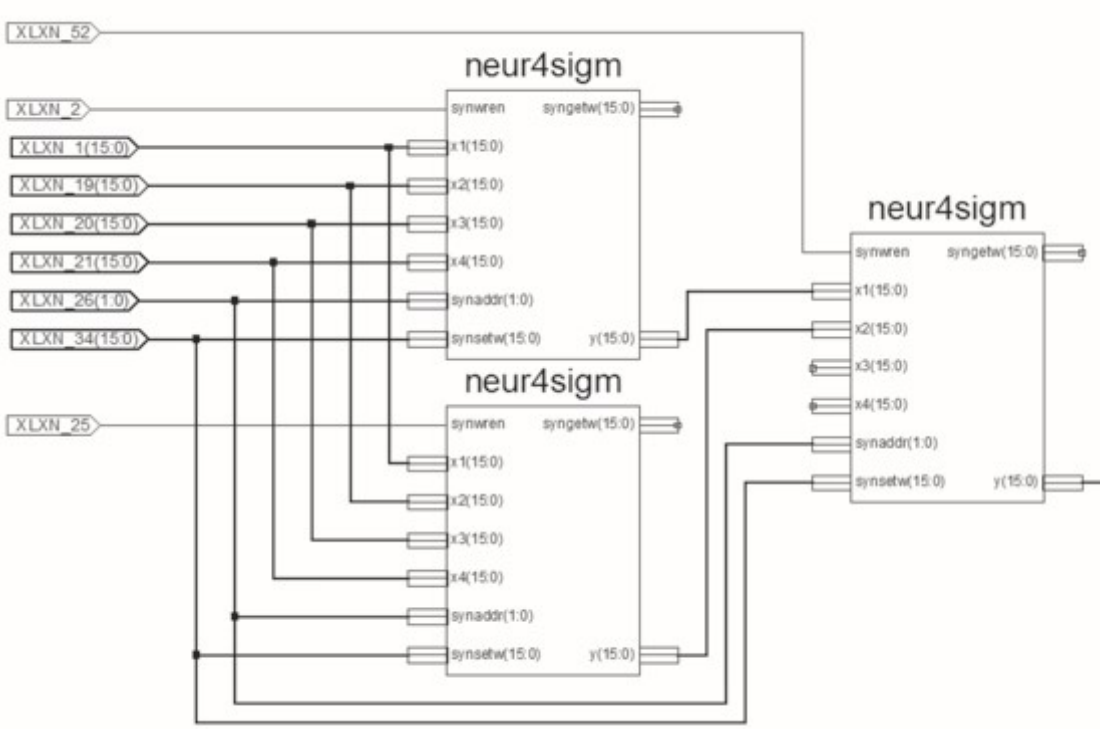

Fig. 1. A fragment of the neural network, implemented on FPGA

To solve the problem of selecting a neural network of minimal structure from a set of possible, [12] describes a technology and software package for the study and evaluation of neural network models (direct and inverse) of multidimensional control objects for their further implementation on the FPGA. Their essence consists in modeling all possible variants of "internal" structures within the "external" structure and determining the structure that provides the lowest mean square error, or the neural network of minimal structure whose mean square error satisfies the condition of the problem.

In order to debug and adapt the neural network models of dynamic objects in real time, it is proposed to use the genetic algorithm and implement it on the FPGA as a parallel search procedure [13].

Realized neural networks can be the basis for further synthesis of neural network components of dynamic control systems based on FPGA. The method of synthesis of hardware components of control systems based on ANN on the example of the implementation of the generalized neural network model of the control object is described in [14-16].

Conclusions. The results of the research of neural networks implemented on FPGAs, according to the developed method, showed their high performance and accuracy with optimal use of computing resource. These neural networks can be the basis for further synthesis of neural network components of dynamic control systems based on FPGA.

References

1. G. Dreyfus. (2005) Neural Networks: Methodology and Applications. SpringerVerlag Berlin Heidelberg. P. 498.

2. A.L. Edelen, S.G. Biedron, B.E. Chase, D. Edstrom, S.V. Milton, P. Stabile. (2016). Neural Networks for Modeling and Control of Particle Accelerators. IEEE Trans.Nucl.Sci. 63 no.2, pp. 878-897

3. M. Lawrynczuk. (2013). Computationally Efficient Model Predictive Control Algorithms: A Neural Network Approach. Springer. Studies in Systems, Decision and Control, vol. 3.

4. Oliver Nelles. (2013). Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models. Springer Science & Business Media, 2013. P. 786.

5. Sigeru Omatu. (2017). Classification of mixed odors using a layered neural network. International Journal of Computing, Volume 16, Issue 1. pp. 41-48.

6. A. Muthuramalingam, S. Himavathi, E. Srinivasan.(2008). Neural network implementation using FPGA: Issues and application. Int. J. Inf. Technol., vol. 4, no. 2, pp. 86-92.

7. Himavathi S., Anitha D. Muthuramalingam A. (2007).Feedforward Neural Network Implementation in FPGA Using Layer Multiplexing for Effective Resource Utilization. IEEE Transactions on Neural Networks. V. 18. № 3. P. 880–888.

8. Ortega-Zamorano F, Jerez J, Juarez G, Perez J, Franco L. (2014). High precision FPGA implementation of neural network activation functions. In: 2014 IEEE symposium on intelligent embedded systems (IES), pp 55–60

9. P. I. Kravets, T. I. Lukina, V. A. Zherebko, V. N. Shimkovich. (2011). Methods of Hardware and Software Realization of Adaptive Neural Network PID Controller on FPGA–Chip. Journal of Automation and Information Sciences. Begell House, New York, USA.Volume 43, Issue: 4, pp. 70-77. DOI: 10.1615/JAutomatInfScien.v43.i4.80

10. Kravets P.I., Shymkovych V.M., Zubenko G.A. (2012).Technology of Hardware and Software Implementation of Artificial Neurons and Artificial Neural Networks by Means of FPGA. Visnyk NTUU ‘KPI’ Informatics, operation and computer systems, , №55, pp. 174-180.

11. Kravets P.I., Shimkovich V.N., Ferens D.A. (2013). Method and algoritms of implementation on PLIS the activation function for artifical neuron chains.Elektronnoe modelirovanie, Vol. 37, no. 4, pp. 63-74.

12. Kravets P.I., Lukina T.I., Shymkovych V.M., Tkach I.I. (2012) Development and research the technology of evaluation neural network models MIMO-objects of control. Visnyk NTUU ‘KPI’ Informatics, operation and computer systems, №57, pp. 144-149.

13. Kravets P.I, Shimkovich V.N. (2013). Method of optimization of weight coefficients of neuron networks by means of genetic algorithm under implementation on programmed logical integral circuits.Elektronnoe modelirovanie, Vol. 35, no. 3, pp. 65-75.

14. Kravets P. I., Shymkovych V.M.,Omelchenko P. (2013) Neuronetwork components of the systems of control of dynamic objects and their hardware-software implementation on FPGA. Visnyk NTUU ‘KPI’ Informatics, operation and computer systems, 2013, №59, pp. 78-85.

15. Kravets P.I., Shymkovych V.M., Fedorchuk V.V., Goy A.A. (2015) Neural controller stability of moving object with the hardware and software realization on FPGA,Visnyk NTUU ‘KPI’Informatics, operation and computer systems, №63, pp. 4–11.

16. P. I. Kravets, V. M. Shymkovych, and V. Samotyy. (2017) Method and technology of synthesis of neural network models of object control with their hardware implementation on FPGA, 2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Bucharest, Romania, pp. 947-951. doi: 10.1109/IDAACS.2017.8095226

ДОВІДКА ПРО АВТОРІВ

Шимкович Володимир Миколайович – асистент, кафедра автоматики та управління в технічних системах, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського».

VolodymyrShymkovych – assistant, Department of Automation and Control in Technical Systems, National Technical University of Ukraine ‘Igor Sikorsky Kyiv Polytechnic Institute’.

E-mail: shymkovych.volodymyr@gmail.com

Бердник Юрій – студент, кафедра автоматики та управління в технічних системах, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського».

Yurii Berdnyk – student, Department of Automation and Control in Technical Systems, National Technical University of Ukraine ‘Igor Sikorsky Kyiv Polytechnic Institute’.

E-mail: berdniky@gmail.com

Володимир Шимкович,
Юрій Бердник

РЕАЛІЗАЦІЯ ТА ДОСЛІДЖЕННЯ НЕЙРОННИХ МЕРЕЖ ПРЯМОГО РОЗПОВСЮДЖЕННЯ НА ПЛІС

Актуальність теми дослідження. Проблема підвищення ефективності адаптивних систем управління на базі нейронних мереж стає більш актуальною в останні дні у зв'язку зі зростаючими вимогами до якості керування складними швидкодіючими об’єктами що працюють в умовах параметричної та інформаційної невизначеності. Дана робота присвячена проблемі підвищення ефективності роботи нейронних мереж при їх апаратній реалізації.

Постановка проблеми. Реалізація та дослідження ефективності роботи нейронних мереж прямого розповсюдження на ПЛІС.

Аналіз останніх досліджень і публікацій. Протягом останніх років з’являється все більше статей присвячених побудові систем управління на основі нейронних мереж та їх апаратній реалізації.

Виділення недосліджених частин загальної проблеми. Дана стаття присвячена реалізації та дослідженню нейронних мереж, реалізованих за раніше розробленим методом і алгоритмами. Дослідженнясфокусовано на показниках ефективності роботи нейронних мереж прямого розповсюдження на ПЛІС.

Постановка завдання. Завданням є дослідження ефективності реалізації ШНМ на ПЛІС за розробленим раніше методом реалізації нелінійних функцій активації та алгоритмами реалізації штучних нейронів.

Викладення основного матеріалу. Побудовано моделі нейронних мереж прямого поширення на ПЛІС за розробленим раніше методом реалізації нелінійних функцій активації та алгоритмами реалізації штучних нейронів у середовищі XilinxISEDesignSuite 14.3. Та досліджено їх. В результаті побудовано порівняльну таблицю, де наведено основні характеристики апаратної реалізації ШНМ: швидкодія, похибка обчислень, а також використаний ресурс ПЛІС.

Висновки. Результати досліджень реалізованих на ПЛІС нейронних мереж, за розробленим методом, показали їх високу швидкодію та точність при оптимальному використанні обчислювального ресурсу. Дані нейронні мережі можуть бути базовими для подальшого синтезу нейромережевих компонентів систем управління динамічними об'єктами на основі ПЛІС.

Ключові слова: нейронні мережі, функції активації, ПЛІС, VHDL.