38. NEURAL NETWORKS HARDWARE-BASED IMPLEMENTATION

Rusakovich H.V.

Belarusian State University of Informatics and Radioelectronics
Minsk, Republic of Belarus

Liakh Y.V.– Senior Lecturer

The paper deals with the problem of implementing hardware based neural networks, describes the examples of activation functions implementation.

In recent decades, an urgent problem in the field of neural technologies has been the speed of hardware implementation of neural networks, as well as their performance in conditions of limited resources.

The hardware implementation of a neural network is characterised by the structure of a neuron and the activation function. A neuron consists of the following elements:

inputs are a set of input signals that can be either single-bit or multi-bit;

input weights are memory registers that store the values of the weights of the neuron inputs;

a block of multipliers perfoms the function of multiplying the input signal by its corresponding weighting factor;

adders are devices that add products of input signals and weighting coefficients;

the block implementing the activation function includes hardware activation function;

the output is the result of neuron calculations [1].

When implementing hardware neural networks, there is a problem of ensuring functional configurability and limited chip area. Therefore, the architecture with efficient use of the area for the activation function is necessary for the full use of parallel processing capabilities. One of the ways to solve this problem is the hardware implementation of the activation function based on the CORDIC (Coordinate Rotation Digital Computer) algorithm. A universal configurable activation function is developed using the CORDIC algorithm and implements a hyperbolic function, a tangent and a sigmoid [2].

The CORDIC algorithm has linear convergence and uses minimal hardware resources. It uses only shift and addition operations, can perform several computational tasks, such as trigonometric calculations, calculation of hyperbolic and logarithmic functions. Since all of the above nonlinear functions are usually used as activation functions, it is advisable to use the CORDIC algorithm to implement activation functions in the development of hardware neural networks.

Using the CORDIC algorithm in a hyperbolic rotation mode, it is possible to implement both tangential and sigmoid functions using the same hardware resources, saving space and energy with better data accuracy.

The architecture of the configurable activation function block uses the CORDIC module in hyperbolic rotation mode. It is used to generate hyperbolic sin and cos functions [2]. The 8-bit precision with a sign CORDIC algorithm is used. The generated trigonometric hyperbolic functions are used to obtain an exponential function. The 8-bit CORDIC output signal is fed to the adder to produce an exponential output signal.

In the course of the research logic circuits for three different types of activation functions were developed: RadBas, LogSig and TanSig. A common feature of activation functions is that they all require calculating the value of exponenta. A calculator of indicators based on the CORDIC algorithm is considered [3]. Each of the 3 activation function constructs accepts and outputs 32-bit floating-point data. Since CORDIC works with fixed-point numbers, data is converted between floating and fixed-point numbers before and after CORDIC in the computation flow. A 32-bit floating-point number has accuracy of up to 8 digits. Due to the transformations between the number systems in the schemes, the correctness of the results deteriorates to 6 characters. Such high accuracy is acceptable for many neural network applications [4].

In conclusion, it should be mentioned that, the hardware implementation of neural networks using the CORDIC algorithm can be easily scaled to work with longer vectors without reducing the speed of operation, which is a difficult task for software implementations. Use of hardware neural networks for data classification is gaining popularity and has practical significance. This makes the possibility to develop compact systems with high performance, which is one of the critical characteristics that can help create real time classification systems.

References:

1. Shymkovych, V. Hardware implementation of radial-basis neural networks with Gaussian activation functions on FPGA / V. Shymkovych, S. Telenyk, P. Kravets — Neural Computing and Applications, 2021. — 9467–9479 pp.

2. Raut, G. A CORDIC Based Configurable Activation Function for ANN Applications / G. Raut, S. Rai, S. Kumar, A. Kumar — 2020 IEEE Computer Society Annual Symposium on VLSI (ISVLSI). — IEEE, 2020.

3. Sahin, I. Design and Implementation of Neural Networks Neurons with RadBas, LogSig, and TanSig Activation Functions on FPGA / I. Sahin, I. Koyuncu. — 120 p.

4. Lezhnev, E.V. Development of the concept of hardware implementation of a neural network based on FPGA / E. V. Lezhnev, A. A. Amerikanov, 2017. — 214 – 215 pp.