

**МЕТОД ПРОЕКТИРОВАНИЯ СТРУКТУРЫ НЕЙРОННЫХ СЕТЕЙ  
НА ОСНОВЕ КЛЕТОЧНЫХ АВТОМАТОВ**

*Дедегкаев Альберт Гажеевич*

*профессор, д-р техн. наук,  
заведующий Кафедрой промышленной электроники  
Северо-Кавказского Горно-металлургического Института  
(Государственного Технологического Университета),  
РФ, г. Владикавказа*

*Рыжков Александр Александрович*

*кафедра промышленной электроники  
Северо-Кавказского Горно-металлургического Института  
(Государственного Технологического Университета),  
РФ, г. Владикавказа  
E-mail: [jurodivij@gmail.com](mailto:jurodivij@gmail.com)*

**THE METHOD OF DESIGNING THE STRUCTURE  
OF NEURAL NETWORKS BASED ON CELLULAR AUTOMATA**

*Dedegkaev Albert Gageevich*

*professor, Doctor of Technical Sciences,  
Head of the Department "Industrial Electronics"  
of the North Caucasian institute of Mining and Metallurgy  
(State Technological University), Russia, Vladikavkaz*

*Ryzhkov Aleksandr Aleksandrovich*

*graduate student of the Department "Industrial Electronics"  
of the North Caucasian institute of Mining and Metallurgy  
(State Technological University), Russia, Vladikavkaz*

**АННОТАЦИЯ**

В статье представлен метод, позволяющий автоматизировать проектирование нейронных структур. В основе метода лежит использование

клеточных автоматов. Аппаратная платформа при реализации метода — программируемая логическая интегральная схема (далее ПЛИС) семейства Cyclon III фирмы Altera. Схема, основанная на базе ПЛИС, позволяет построить эффективную распределённую систему для решения различных задач в области цифровой обработки сигналов.

### **ABSTRACT**

The method, proposed in the paper gives opportunity to automate the design of neural structures. The method relies on the use of cellular automata. The hardware platform for the implementation of the method — is programmable logic integrated circuit (hereinafter FPGA) from Cyclon III family of the Altera firm. The scheme based on FPGA, gives opportunity to build an effective distributed system solutions for various applications in the field of digital signal processing.

**Ключевые слова:** нейронные сети, программируемые логические интегральные схемы, клеточные автоматы, аппаратная реализация.

**Keywords:** neural networks, programmable logic integrated circuits, cellular automata, hardware implementation.

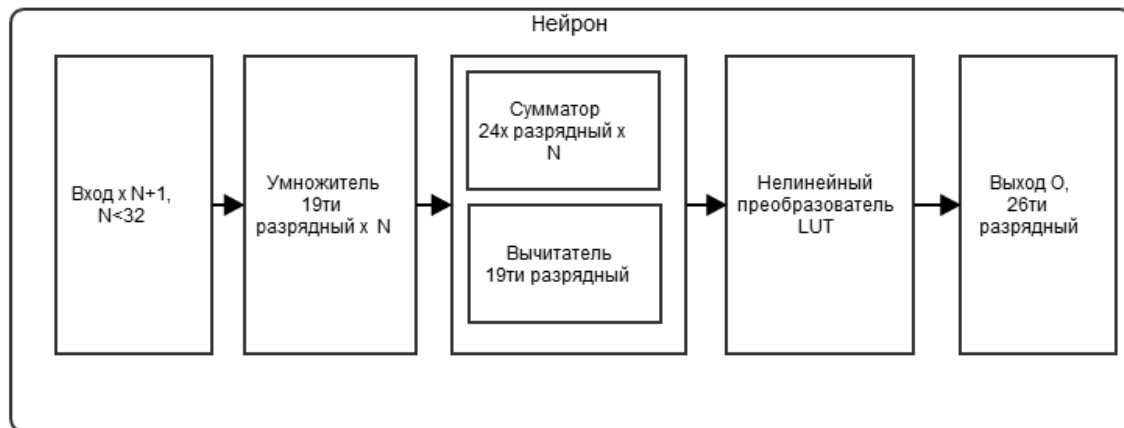
### **Введение**

Одно из основных достоинств нейронных сетей — одновременная обработка большого количества сигналов. Большая часть реализуемых в настоящее время сетей — программная эмуляция, работающая на персональных компьютерах и специализированных серверах [1; 2; 5]. Помимо достоинств, таких как простота, достигаемая программной эмуляцией, и повсеместная доступность персональных компьютеров, эти станции имеют и недостатки, например, избыточность и большую потребляемую мощность. Скорость работы нейронной сети, построенной таким образом, будет напрямую зависеть от ее размерности, так как частота работы единого центрального процессора будет делиться на все использующиеся в данной сети нейроны. Их быстроедействие обратно пропорционально количеству элементов

клеточного автомата, в то время как распределенная структура ПЛИС позволяет более эффективно реализовать нейронные сети средней величины, работающие с большей скоростью [3].

### Структура нейрона

Перед тем как строить топологию нейронной сети, сначала необходимо спроектировать структуру единичного нейрона. Ниже предлагается схема замещения искусственного нейрона, разбитая на пять функциональных блоков: входы, умножители, сумматор, нелинейный преобразователь, выход, далее используемая в ПЛИС с сохранением этих функциональных блоков.



*Рисунок 1. Схема замещения искусственного нейрона*

На основе схемы замещения строится модель единичного нейрона.

Входные данные — знаковые десятиразрядные числа в диапазоне  $[-1;1]$ . Старший разряд содержит знак. Веса сети и значение порога представлены девятнадцатиразрядными числами. Старший разряд знаковый, затем 9 разрядов целой части и 9 разрядов дробной части. Результат произведения весов и входных данных содержится в двадцатичерехразрядном регистре. Выходное значение содержится в двадцатишеститиразрядном регистре.

Далее показан заголовок модуля пятивходового линейного нейрона на языке описания аппаратуры Verilog

```
module soma(  
input wire [9:0] x0,  
input wire [9:0] x1,
```

```
input wire [9:0] x2,  
input wire [9:0] x3,  
input wire [9:0] x4,  
output wire [26:0] y  
);
```

В то время как веса и пороги искусственной нейронной сети реализованы путем простых блоков умножения и суммирования, активационная функция требует использования блока памяти. То есть в область памяти записывается таблица истинности некоторой функции активации. В нашем случае использовалась сигмоидальная функция:

$$\sigma(x) = \frac{1}{(1 + \exp(-tx))},$$

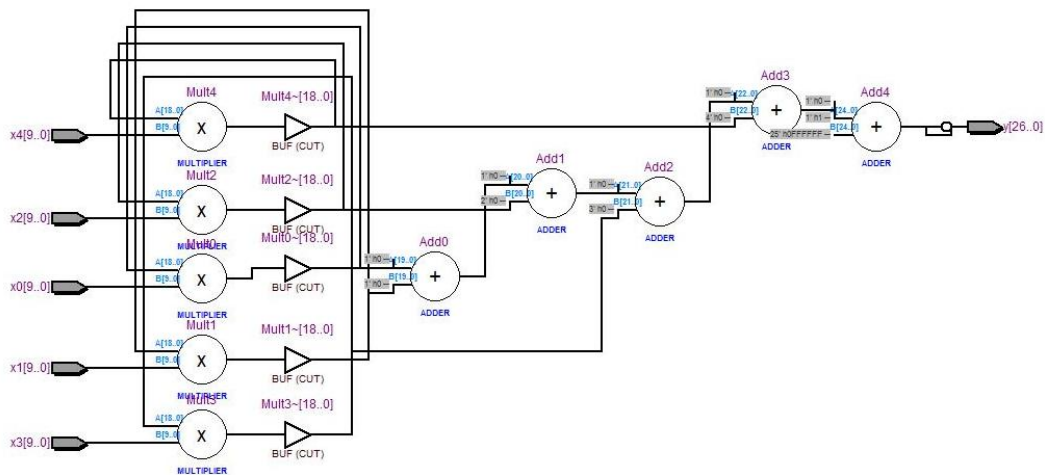
где:  $x$  — результат суммирования, поступающий на вход нелинейного преобразователя,

$t$  — коэффициент, задающий крутизну характеристики. При  $t = 0$ , сигмоида вырождается в пороговую функцию.

Во избежание излишнего потребления памяти ПЛИС, используется свойство симметрии пороговой функции. Таким образом, в памяти хранится 8000 двадцатисемиразрядных значений первой половины сигмоидальной функции и дополнительная логическая схема на основе сумматора для построения второй части.

На базе ПЛИС возможна реализация вычисления данной функции, но это потребует больших аппаратных ресурсов [4].

Схематическое изображение, реализующее устройство нейрона без использования сигмоидальной функции, показано на рис. 2.



**Рисунок 2. Схема нейрона на базе ПЛИС**

Для реализации любой структуры необходимо обеспечить возможность хранения численных величин. Разрядность этих величин определяет число возможных состояний каждого нейрона, а с учетом числа нейронов — автомата в целом. Для этого можно сохранять выходное значение каждого нейрона в переменную типа reg. Так как каждый нейрон представляет собой модуль на языке Verilog, то можно строить топологию сети, оперируя отдельными нейронами, и не углубляться до значений весов.

### **Клеточные автоматы**

Клеточный автомат — дискретная модель, изучаемая в математике, теории вычислимости, физике. Включает регулярную решетку ячеек, каждая из которых может находиться в одном состоянии из конечного множества. Данная решетка может быть любой размерности.

Клеточный автомат можно представить как ориентированный граф, обладающий следующими свойствами:

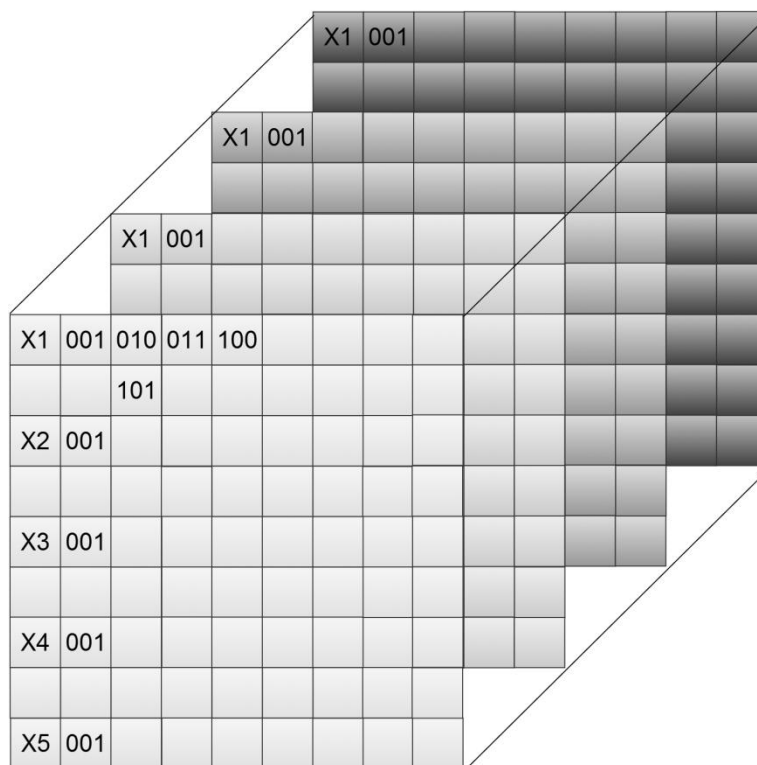
- каждая вершина находится в некотором состоянии из множества возможных состояний;
- в определенное время (определяемое частотой работы задающего генератора) каждая вершина переходит в новое состояние, однозначно обусловленное ее предыдущим состоянием и состоянием связанных с ней вершин;

- правила перехода из одного состояния в другое (функция переходов) для всех вершин одинаковы;
- клеточный автомат синхронизирован по времени.

Важной особенностью является то, что клеточный автомат не имеет памяти, то есть из текущего состояния нельзя однозначно определить предыдущее: если в данный момент клетка пуста, нельзя определить, что в ней было в предыдущий момент времени.

Основываясь на том, что микросхема ПЛИС состоит из множества настраиваемых логических блоков и на том, что структура ИНС равномерно распределена, приходим к выводу, что все используемые логические блоки будут настроены схожим образом, за исключением умножителей, так как они будут содержать настраиваемые веса. Также если допустить, что все нейроны заданной области имеют одну и ту же функцию активации, то этот элемент можно вынести за «пределы» нейрона и использовать один нелинейный преобразователь на несколько нейронов, однако это приведет к падению быстродействия. Поэтому использование этой схемы подключения возможно только при нехватке ресурсов под «полноценные» нейроны.

Согласно концепции клеточных автоматов разобьем все пространство на трехмерную матрицу, где каждый элемент матрицы может находиться в заданном состоянии (см. рис. 4). Состояние этого элемента описывается конечным числом разрядов, в нашем случае — 28. Три старших разряда будут определять состояние ячейки: 000 — входной сигнал (X), 001 — дендрит, 010 — сумматор, 011 — нелинейный преобразователь, 100 — аксон, 101 — порог, 110 — пустая клетка, 111 — выход (Y).



**Рисунок 3. Пример построения 4-х пятивходовых нейронов на основе клеточного автомата**

Построение ИНС осуществляется при задании правил работы клеточного автомата. Обычно клеточный автомат является синхронным, то есть правила применяются для всех клеток одновременно за один такт работы. В нашем случае правила применяются последовательно к каждой плоскости. В рамках одной плоскости правила применяются слева направо, сверху вниз. Изначально все элементы матрицы имеют состояние 110, то есть они пусты. Задается число входов. Обычно входы ПЛИС подключаются к выходу АЦП, который и определяет размерность входных сигналов. Также задается размерность выходного сигнала исходя из количества классов, на которые необходимо разбить входные данные.

Затем клеточный автомат начинает работать согласно следующим правилам.

1. Если клетка пуста и сосед вход или аксон, создается дендрит.
2. Если клетка пуста и сосед дендрит и нет соседа сумматора, создается сумматор.

3. Если клетка пуста и сосед сумматор и нет соседа нелинейного преобразователя, то создается нелинейный преобразователь.

4. Если клетка пуста и сосед нелинейный преобразователь и нет соседа аксона, создается аксон.

5. Если клетка пуста и соседи — дендрит и сумматор и нет соседа порог, создается порог.

6. Если клетка пуста и сосед аксон, создается дендрит.

Для корректной работы автомата необходимо пояснить термин «сосед». Для предлагаемого автомата на заданной плоскости соседями являются все клетки в том же столбце, что и рассматриваемая ячейка и клетки предыдущего столбца. Значение любого аксона отображается на все плоскости.

### **Заключение**

Следуя правилам, заложенным в данном простом клеточном автомате, можно быстро проектировать нелинейные многослойные нейронные сети малых и средних размеров с последующим их размещением на кристалле ПЛИС. Недостатком разработанного метода является ограниченное количество проектируемых архитектур. Их количество можно расширить, дополнив систему правил и впоследствии реализовать обучение сети.

### **Список литературы:**

1. Нейронные сети. Практикум/ Сивохин А.В., Лушников А.А. — М., 2004.
2. Нейронные сети. Statistica Neural Networks. Методология и технологии современного анализа данных/ В. Боровиков. Горячая Линия — Телеком, 2008.
3. Тоффоли Г., Маргулос Н. Машины клеточных автоматов. — М.: Мир, 1991.
4. Neural Network Implementation Using FPGA: Issues and Application/ A. Muthuramalingam, S. Himavathi, E. Srinivasan. International Journal of Information and Communication Engineering 4:6, 2008.
5. Object-Oriented Neural Networks in c++/ Joey Rogers, Morgan Kaufmann, 1996.