Programmable Arithmetic Units in the Neural Networks, The IREN Architecture

Timót Hidvégi[$,‡], Péter Keresztes[$]

{hidvegi}{keresztp}@sze.hu
[$] Széchenyi István University, Egyetem tér 1, H-9026 Győr, Hungary
[‡] Budapest Tech, Tavaszmező u. 15-17, H-1084 Budapest, Hungary

Abstract: The neural networks have some solutions of simulator (on PC), analog and emulated digital [1],[2],[3],[4],[5],[6]. In this paper we deal with design of the emulated digital neural networks. The paper shows a universal emulated digital architecture. In the paper the compiled solution will be presented which is testing and simulating using a VIRTEX FPGA development system [15].

Keywords: Neural Networks, FPGA, ASIC solutions, configurable arithmetic units

1   Introduction

The implementation of a neural network model can be a software module, an emulated digital (on silicon or FPGA) or an analog circuit. The analog neurons are very fast but, unfortunately, inaccurate. Therefore, the different emulated digital neurons (e.g. Hopfield, CNN, Perceptron) are preferred in several applications. The accuracy of an emulated digital neural network depends on the resolution of the numbers given in binary digits. For example the solution of a partial differential equation demands not higher resolution then six bits. The IREN's accuracy is eight bits.

In the second section of the paper we show the test environment. The programmable emulated digital (IREN) architecture is shown in section 3. The Chapter 4 shows an application in IREN architecture. The conclusions are given in section 5.

2 The FPGA Environment

We used a VIRTEX XCV300 FPGA from XILINX [13]. The XCV300 FPGA has 322.970 system gates, (2)*32*48 CLBs (slices) and max. available I/O number is equal to 316 (Figure 1). This FPGA was fabricated by 0.22 μm 5-layer CMOS process. The VIRTEX family's key feature is the flexible, regular architecture (dedicated carry logical for high-speed and density).
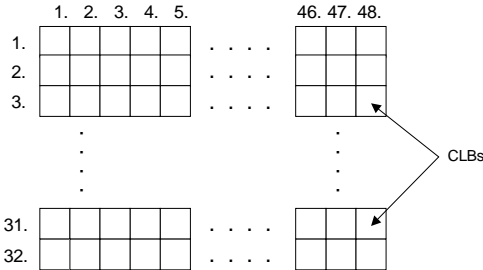

Figure 3

The floorplan of the XCV300 FPGA

The different arithmetic architectures were simulated and measured on the test-panel from XESS [14]. The design specification was given in schematic input. This description is compiled to stream format that can be downloaded into VIRTEX FPGA. The FPGA is controlled by XC95108 CPLD, which can be programmed too. The basic architecture of a test environment is given on Figure 2.
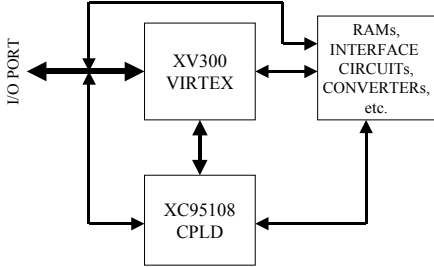

Figure 2

Block diagram of the test-panel

3   The IREN Architecture

The IREN architecture contains a RAM, a Programmable Arithmetic Array [7], a Memory and a Control and Timing Unit (Figure 3) [9].
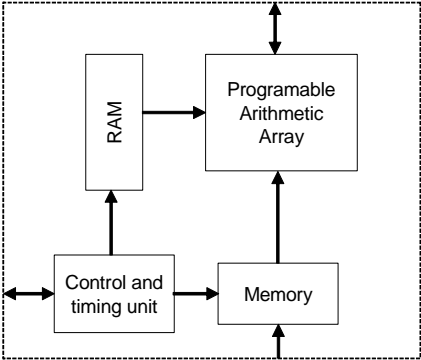
Figure 3

The structure of IREN architecture

The Figure 4 shows the programmable arithmetic array. This array contains some multipliers, adders and registers. The modules are connected by the interconnections. The accuracy of the interconnections is 2x16 bits.
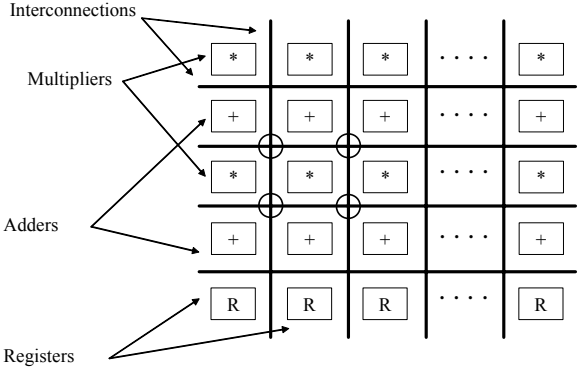
Figure 4

The structure of Programmable Arithmetic Array

The adder with accuracy 8 bits can be seen in Figure 5. The adder builds up two Look Ahead Carry, therefore, the speed is maximal.
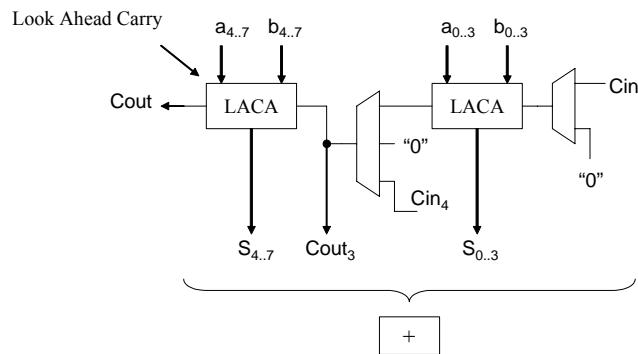
Figure 5
The configurable adder

The configurable adder can be installed into cascaded with the "Cin", "Cout" and "Cout$_3$" signals. Consequently, the accuracy of the adder can be changed it can be 4, 8, 12, etc. bits (Figure 6).
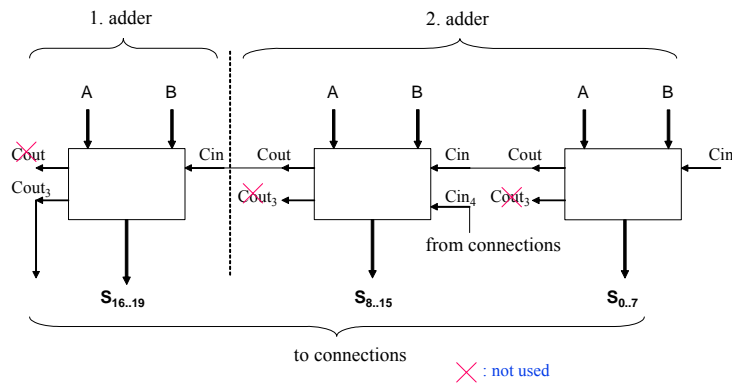

Figure 6
The cascaded configurable adder

The block of the control-unit can be seen in Figure 7. The main elements of control unit are some programmable OR arrays, Final State Machines (FSM) and a clock circuit. The number of states can be changed by the programmable OR arrays.
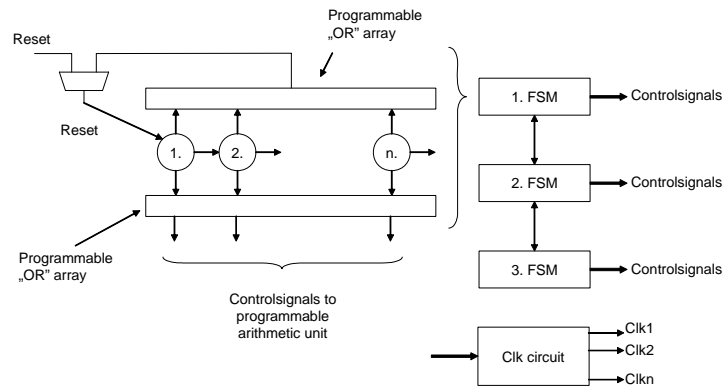
Figure 7

The programmable control unit

The following state graph can be designed with the three FSMs and the programmable OR arrays (Figure 8).
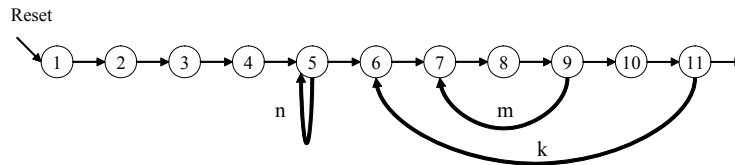

Figure 8

An example of control state graph

4   An Application

The Figure 9 shows a Hopfield Neuron Network [10] with 4 neurons and a neuron cell. This neuron network is downloaded into the IREN architecture.
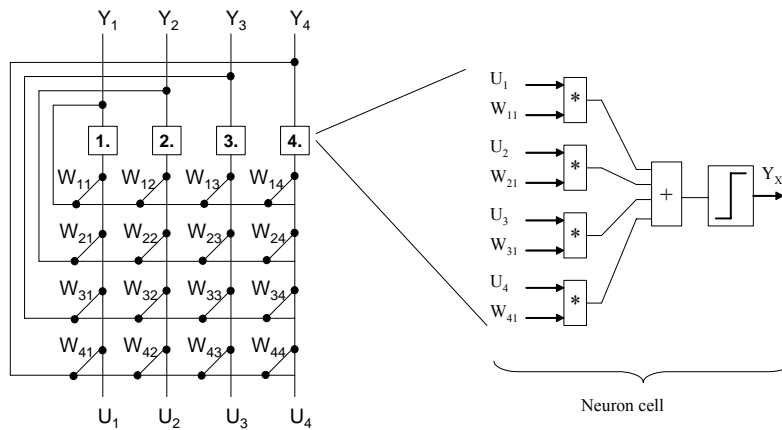
Figure 9

The Hopfield Neuron Network and a neuron cell

We can use 4x4 multipliers but it is not optimal solution. This architecture can be optimalized according to area and dissipation power. The Figure 10 shows the optimized neuron.
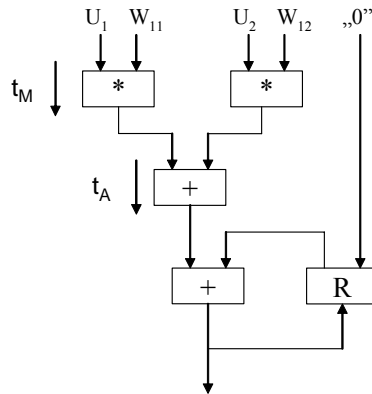

Figure 10

The optimized neuron cell according to area

The Figure 11 shows the used interconnections with "square dot" line. It can be seen that the multiplier 21 cannot be used because there are not interconnections.
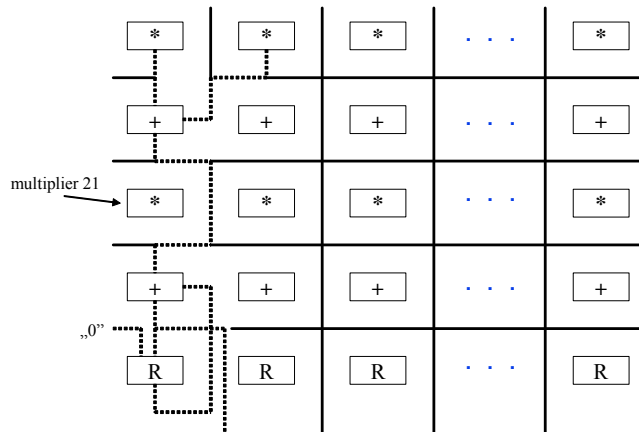
Figure 11
The compiled optimized neuron cell

Conclusions

In this paper a new programmable architecture was shown. This architecture was compiled into a VIRTEX FPGA and analyzed. The different data of the VIRTEX FPGA and the compiled part-variation can be seen in Table 1.

	Number of CLBs	Number of 4 input LUTs	Total equivalent gate
XC300	3072	6144	-
Compiled multiplier	63	118	708
Compiled adder	15	27	162

Table 1
The results of the compiler

References

[1]   T. Hidvégi, P. Keresztes and P. Szolgay, "Enhanced Modified Analized Emulated Digital CNN-UM (CASTLE) Arithmetic Cores", Journal of Circuits, Systems, and Computers, Special Issue on "CNN Technology and Visual Microprocessors

[2]   P. Keresztes, Á. Zarándy, T. Roska, P. Szolgay, T. Bezák, T. Hidvégi, P. Jónás and A. Katona, "An Emulated Digital CNN Implementation", *Journal of VLSI Signal Processing,* Special Issue: Spatiotemporal Signal Processing with Analogic CNN Visual Microprocessors, Vol. 23, No. 2/3, pp. 291-304, Kluwer, 1999

[3] G. Linan, R. Dominguez-Castro, S. Espejo, A. Rodriguez Vazquez, "ACE16k: A Programmable Focal Plane Vision Processor with 128*128 Resolution" Proc. of ECCTD'01, pp. I-345-348, 2001, Espoo, Finland

[4] A. Paasio, A. Kananen, V. Porra, "A 176*144 processor binary I/O CNN-UM chip design" Proc. of ECCTD '99, Stresa, pp. 82-86, 1999

[5] G. Almasi et al., "Cellular Supercomputing with System-On-A-Chip" IEEE Proc. of Solid-State Circuits Conference 2002, San Francisco

[6] Z. Nagy, P. Szolgay, "An emulated digital CNN-UM implementation on FPGA with programmable accuracy" DDECS'01 IEEE pp. 203-208, Győr, Hungary, April 18-20, 2001

[7] Kai Hwang, "Computer Arithmetic Principles, Architecture, and Design" John Wiley & Sons, New York, 1979

[8] P. Arató, T. Visegrády, I. Jankovits, Sz. Szigeti, "High-Level Synthesis of Pipelined Datapaths" Edited by P. Arató, Panem Budapest, 2000

[9] Raul Camposano, Wayne Wolf, "High-Level VLSI Synthesis" Boston, Kluwer Academic Publishers, 1991

[10] Dunay Rezső, Horváth Gábor, Pataki Béla, Strausz György, Szabó Tamás, Várkonyiné Kóczy Annamária, "Neurális hálózatok és műszaki alkalmazásaik", Műegyetemi Kiadó, Budapest, 1998

[11] Enoch O. Hwang, "Microprocessor Design, Principles and Practices with VHDL", Brooks/Cole 2004

[12] Reto Zimmermnn, "Lecture notes on Computer Arithmetic: Principles, Architectures, and VLSI Design", Integrated Systems Laboratory, Swiss Federal Institute of Technology Zurich, 1998

[13] www.xilinx.com

[14] www.xess.com