Artificial Neural Networks: a Review of Commercial Hardware

Fernando Morgado Dias[a], Ana Antunes[a], Alexandre Manuel Mota[b]

[a]*Escola Superior de Tecnologia de Setúbal, Departamento de Engenharia Electrotécnica, Campus do IPS, Estefanilha, 2914-508 Setúbal, Portugal*
Tel: +351 265 790000, Email:{ fmdias,aantunes}@est.ips.pt
[b]*Departamento de Electrónica e Telecomunicações, Universidade de Aveiro, 3810 Aveiro, Portugal, Tel: +351 234 370383, Email: alex@det.ua.pt*

Proposed in 9 October 2003

Abstract

Artificial Neural Networks became a common solution for a wide variety of problems in many fields, such as control and pattern recognition to name but a few. Many solutions found in these and other Artificial Neural Network fields have reached a hardware implementation phase, either commercial or with prototypes. The most frequent solution for the implementation of Artificial Neural Networks consists of training and implementing the Artificial Neural Networks within a computer. Nevertheless this solution might be unsuitable because of its cost or its limited speed. The implementation might be too expensive because of the computer and too slow when implemented in software. In both cases dedicated hardware can be an interesting solution.
The necessity of dedicated hardware might not imply building the hardware since in the last two decades several commercial hardware solutions that can be used in the implementation have reached the market.
Unfortunately not every integrated circuit will fit the needs: some will use lower precision, some will implement only certain types of networks, some don't have training built in and the information is not easy to find.
This article is confined to reporting the commercial chips that have been developed specifically for Artificial Neural Networks, leaving out others solutions.
This option has been made because most of the other solutions are based on cards which are built either with these chips, Digital Signal Processors or Reduced Instruction Set Computers.

Keywords: Artificial Neural Networks, Hardware.

1. Introduction

The field of Artificial Neural Networks (ANN) has crossed different stages of development. One of the most important steps was achieved when Cybenko (Cybenko, 1989) proved that they could be used as universal approximators.

A negative stage was brought by the book of Minsky and Papert called Perceptrons (Minsky et al., 1969). This negative phase was overcome when algorithms for training of multilayer ANN where proposed in the decade of the 80s. Since then much work has been done regarding ANN and their application to many different fields (Dias et al., 2001). Naturally the successful application to some areas led to commercial or specific applications that can work without having a computer attached.

The need for leaving the most common implementation of ANN with a computer might arise from a number of reasons: reducing the cost of the implementation, achieving higher processing speed or simpler implementations.

Reducing cost or having simpler implementations can be achieved simply by replacing the computer by specific hardware.

Unlike the conventional von-Neumann architecture of computers that is sequential in nature, ANN profit from massively parallel processing (Liao). This can be exploited by specific hardware to increase processing speed.

For these applications that share the necessity of working without a computer, some dedicated hardware has already been built avoiding the difficulty of producing hardware for each new application.

The hardware produced has been a result of different needs and therefore has different uses.

In order to choose hardware for a specific application, details about each circuit will be needed.

The different solutions might be useful or not depending on the precision used for the weights, maximum number of weights, type of network implemented, availability of on circuit training algorithms and other characteristics.

This article is confined to reporting the commercial chips that have been developed specifically for Artificial Neural Networks, independently of the technology used (Application Specific Integrated

Circuits, Field Programmable Gate Arrays, Sea of Gates or others), leaving out other solutions. This option has been made because, aside from some hybrid solutions, most of the other solutions are based on cards which are built either with these chips, Digital Signal Processors or Reduced Instruction Set Computers.

The utility of this survey can therefore be summarized in two different directions: a short reference for those who need hardware for a specific implementation and information about the existing solutions for those who seek to develop a new implementation.

2. Hardware Specification

From the point of view of the user, the first step towards selecting a hardware solution is a short hardware specification. This specification includes the type of ANN (feedforward multi-layer, Radial Basis Function, Kohonen, etc.), the number of neurons, number of external input and outputs, the number of connections to each neuron, the precision, the speed of operation or performance and other characteristics that can be more or less important depending on the application.

Most of these characteristics can be derived directly from the application to be developed, while others deserve a closer look.

The precision used should be an important parameter to take into account. There might be a different precision for a number of parts: inputs, outputs, weights, internal calculations, accumulators and multiplication.

The performance of the circuit can be measured in many different ways and is an issue that is far from being consensual.

The most common performance rating is Connection Per Second (CPS) (Lindsey, et al., 1994) which is defined as the number of multiply and accumulate operations per second during the recall or execution phase. An equivalent measure exists for the learning phase: Connection Update Per Second (CUPS) and rates the number of weight changes per second.

Other measures exist as well. The value of CPS can be normalized dividing it by the number of weights N_w (equation 1) obtaining the Connection Per Second Per Weight (CPSPW), which was suggested as a better way of rating the performance of each solution (Holler, 1991).

$$CPSPW = CPS/N_w \quad (1)$$

Another measure is Connection Primitives Per Second (CPPS), which can be calculated as:

$$CPPS = b_{in} x b_w x CPS \quad (2)$$

where b_{in} is the number of bits used for the input and b_w is the number of bits used for the weights. This measure allows the precision to be included in the performance measure (Keulan et al., 1994) (Schüffny, et al., 1999). These two measures can also be applied to CUPS.

Another parameter that can be used to analyse performance is power dissipation (Schüffny, et al., 1999). Depending on technology, clock frequency, number of processing elements, accuracy, etc, each hardware solution has a measure of power dissipation, which cannot be compared directly. An energy per connection measure was proposed in (Schüffny, et al., 1999). This measure is an indicator of the energy efficacy in the circuits and is becoming more important as the integration in the solutions increases because the need to carry away dissipated power limits integration density at system level. Unfortunately this measure is frequently not available for most of the chips, which made it impossible to include it in this survey.

For a more detailed specification other information may be taken into account: learning facilities, cascadability, type of storage of the weights, type of implementation of the activation functions, clock rate, number of inputs and outputs and technology or type of implementation of the circuit (analogue, digital or mixed).

3. Types of Implementation

Classify the implementation types of the hardware solutions is a controversial task, as can be seen from the examples present in the literature. For example in (Aybay et al., 1996) a classification is proposed which divides the hardware solutions in Neuro-chips and Neurocomputers having both subcategories of special and general purpose. Another classification, although more general since it is meant for ANN hardware components, is given in (Caviglia), where the main types are: neurocomputers, personal computer accelerators, chips, cell libraries and embedded microcomputers.

In (Heemskerk, 1995) the classification is made according to figure 1. Here the global hardware solution is called Neurocomputers, which can be made of Standard Chips or Neurochips. The Standard Chips can be classified either by Sequential and Accelerator boards or Multi-Processor solutions.

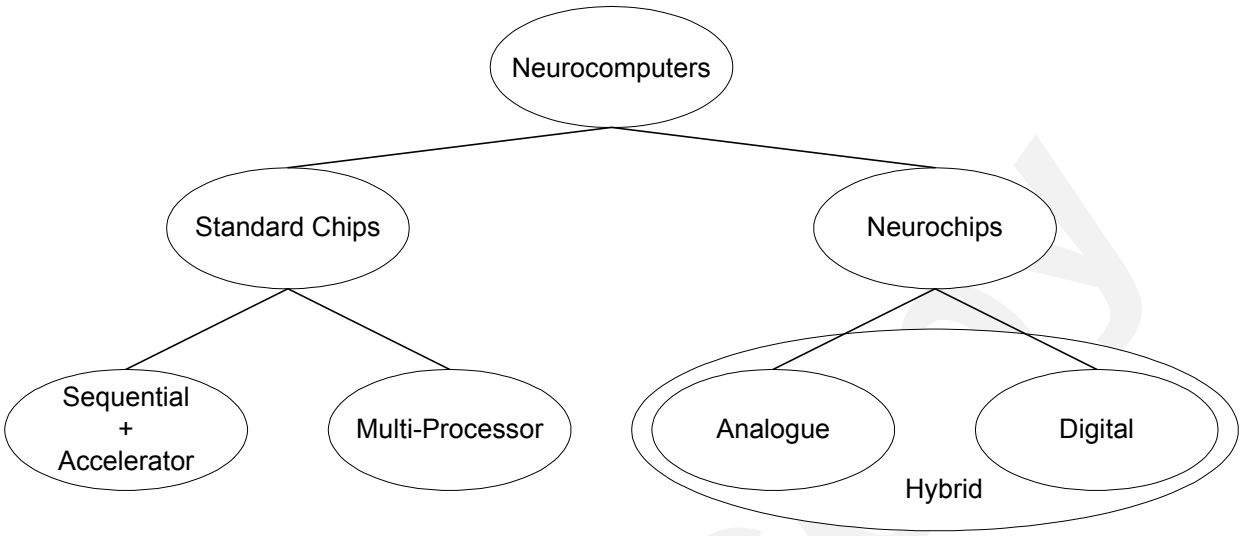

Figure 1- Neural networks' hardware categories

The Neurochips, which are constituted by Application Specific Integrated Circuits (ASICs) can be classified either as Analogue, Digital or Hybrid.

The present article deals only with the category of Neurochips with a special emphasis on the commercially available solutions.

Therefore the classification used here is similar to the proposed in (Heemskerk, 1995) for Neurochips since it is a separation of the solutions according to their nature: analogue, digital or hybrid.

4. The Commercial Hardware

According to the previously defined simplified classification the existing hardware will now be presented.

The solutions found are presented with information about the architecture, implemented capacity to learn, precision, number of neurons available, number of synapses or memory available to implement them and a measure of speed of operation or performance.

It should be noted that some of the hardware presented is not being produced at present but is still included both as a reference and because it might still be available from some suppliers (please read section 5).

For each item the information presented in the table is later detailed in the text and the abbreviations explained.

Architecture: In this topic the information refers to the type of network that can be implemented and in some situations additional information about the type of implementation.

Examples are: Feedforward Multilayer networks, Matrix operation circuits, etc.

The abbreviations used are: Multilayer (ML), General Purpose (GP), Single Instruction Multiple Data (SIMD) Floating Point (FP), Integer (Int), Radial Basis Function (RBF), Fully Connected and Recurrent (FCR).

Learn: In this topic information is given about the possibility of on-chip learning for the ANN.

Examples are: Program, when the algorithm to use can be programmed, Hopfield, Boltzmann, etc.

The abbreviations used are: Backpropagation (BP), Region of Influence (ROI), Restricted Coulomb Energy (RCE), Probabilistic Neural Networks (PNN), K-Nearest Neighbour (KNN), L1 and LSUP (two norms to use with RBF (ZISC78)).

Precision: The information about precision is presented in the format number of bits for input times number of bits for output. The information about internal precision is seldom available, which is not very satisfying from a scientific or from an industrial point of view.

Neurons: This field indicates the number of neurons available in the circuit. In some cases the circuit contains Processing Elements (PE) instead of neurons.

Synapses: This field indicates the number of connections or the amount of memory available for storing the values of the weights.

Speed: In this field a measure of performance is indicated. Most of the abbreviations have already been mentioned in section 2 and pat/s stands for the number of patterns to be processed in each second.

In all the fields N.A. stands for information not available.

The information presented has been collected either directly from the manufacturers or available in previous studies about hardware for ANN.

4.1. Analogue Implementations

In general, analogue implementations have the advantage of obtaining high speed and density while having several technological drawbacks. These difficulties are mainly related to: store the weights, stability with temperature variations and obtaining linear multipliers over a wide operating range. Naturally these problems may result in a loss of precision when compared to other types of implementation.

Table 1 shows the resume of the characteristics of the analogue implementations.

ETANN- The Intel chip, 80170NX also known as Electrically Trainable Analog Neural Network (ETANN) was the first commercially available. It stores the weights as electrical charge in floating gates and uses Gilbert multipliers of 4-quadrants. It has 64 neurons and achieves a performance of 2 GCPS.

Synaptics Silicon Retina- This chip is a special case in the hardware implementation of ANN since instead of implementing a conventional architecture it tries to emulate biological neurons as close as possible (Lindsey, et al., 1994).

4.2. Digital Implementations

The digital implementations, in general, have the following advantages: weight storage in memory, easy to integrate with other applications, less difficult to implement learning algorithms and exact within the number of bits of the operands and accumulators. In a simple way, when compared with the analogue solutions, the digital ones can be said to be more precise.

On the other hand digital implementations are in general slower (due to the need of conversion of analogue inputs and to the operation of digital multipliers) and present also difficulties to implement the activation functions.

Digital implementations are usually divided into four classes: slice architectures, single instruction multiple data (SIMD), systolic array devices (Baratta et al., 1998) (Lindsey, et al., 1994) and RBF, although some other classes are sometimes assumed like multiprocessor chips or others that result from the technology used (PLD and FPGA) or the similarity with other type of hardware (For example Digital Signal Processors).

The characteristics of each class and the division of the circuits in classes are presented below and Table 2 shows the resume of the characteristics of the digital implementations.

Slice Architectures: The principle of the slice architectures is to provide blocks to build networks of arbitrary size, although they can be used to implement an ANN by themselves. In most cases these blocks implement themselves a neuron and their name derives from the bit slice concept of conventional digital processors (Lindsey, et al., 1994).

Because of the type of architecture this solution usually does not include on-chip training.

Examples of this type are:

MD-1220- This chip from Micro Devices was one of the very first implementations of commercial hardware for ANN. It contains eight neurons with hard-limit activation functions and eight sixteen bit weights with only one bit input. The multipliers used are bit-serial.

NLX-420- Each chip contains sixteen neurons with variable precision. The sixteen bits available for inputs and weights can be selected as sixteen-one bit, four-four bits, two-eight bits or one-sixteen bits. The activation functions are user defined and internal feedback is available.

Lneuro-1- Composed of sixteen PE this chip also has sixteen bit weights of variable precision like the NLX-420 solution. The weights are to be holded in on-chip cache (1Kbyte) and the transfer functions are to be implemented off-chip.

Lneuro-2.3- The Lneuro 2.3 is an evolution of the Lneuro 1. It consists of 12 PE that can be used either in parallel or pipelined modes. Each PE contains 128 16-bit registers that can be used either for storing the weights or the states. The PEs also contain a 16 to 32-bit multiplier, a 32-bit ALU and a barrel shifter. Microcode can be used for adapting the chip to the applications.

Systolic Arrays and SIMD solutions could be included into a category named multiprocessor chips.

SIMD: As the name itself explains each processor executes the same instruction in parallel in different data.

Examples of this type are:

Inova N64000- This chip contains 64 PE each one having a 9x16 bit integer multiplier, a 32 bit accumulator and 4Kbytes of on-chip memory for weight storage. The circuit includes common control and data buses for each processor, which makes it easy to combine multiple chips.

Table 1 – Neural networks analogue hardware implementations.

Name	Architecture	Learn	Precision	Neurons	Synapses	Speed
Intel ETANN	Feedforward, ML	No	6bx6b	64	10280	2 GCPS
Synaptics Silicon Retina	Neuromorphic	No	N.A.	48x48	Resistive net	N.A.

HNC100-NAP- The Hecht-Nielsen Computers 100 Neurocomputer Array Processor is a 32 bit floating point processor. The choice of a 32 bit floating point representation limits the number of PEs to 4. The weights are stored off-chip.

Hitachi WSI- There are two different circuits developed by Hitachi which have some common and some distinct characteristics. In common they have the 9bit x 8 bit precision and the type of architecture. Distinct is the built in learning (Hopfield and

Table 2 – Neural networks digital hardware implementations.

Name	Architecture	Learn	Precision	Neurons	Synapses	Speed
Slice Architectures						
Micro Devices MD-1220[2]	Feedforward, ML	No	1bx16b	8	8	1.9MCPS
NeuraLogix NLX-420[2]	FeedForward, ML	No	1-16b	16	Off chip	300CPS
Philips Lneuro-1	Feedforward, ML	No	1-16b	16 PE	64	26MCPS
Philips Lneuro-2.3	N.A.	No	16-32b	12 PE	N.A.	720MCPS
SIMD						
Inova N64000[1]	GP, SIMD, Int	Program	1-16b	64 PE	256k	870MCPS 220MCUPS
Hecht-Nielson HNC 100-NAP[2]	GP,SIMD,FP	Program	32b	4 PE	512k Off chip	250MCPS 64MCUPS
Hitachi WSI	Wafer, SIMD	Hopfield	9bx8b	576	32k	138MCPS
Hitachi WSI	Wafer, SIMD	BP	9bx8b	144	N.A.	300MCUPS
Neuricam Nc3001 Totem	Feedforward, ML, SIMD	No	32b	1-32	32k	1GCPS
Neuricam Nc3003 Totem	Feedforward, ML, SIMD	No	32b	1-32	64k	750MCPS
RC Module NM6403	Feedforward, ML	Program	1-64bx1-64b	1-64	1-64	1200MCPS
Systolic Array						
Siemens MA-16	Matrix ops	No	16b	16 PE	16x16	400MCPS
Radial Basis Functions						
Nestor/Intel NI1000[3]	RBF	RCE, PNN, program	5b	1 PE	256x1024	40kpat/s
IBM ZISC036	RBF	ROI	8b	36	64x36	250kpat/s
Silicon Recognition ZISC 78	RBF	KNN, L1, LSUP	N.A.	78	N.A.	1Mpat/s
Other Chips						
SAND/1	Feedforward, ML, RBF, Kohonen	No	40b	4 PE	Off-Chip	200MCPS
MCE MT19003	Feedforward, ML	No	13b	8	Off chip	32MCPS

[1] No longer available according to (Lindsey)
[2] No longer available.
[3] No longer available according to (Neuronet)

Backpropagation with momentum term (Ienne et al., 1995)), the number of neurons and the memory available for synapses.

Neuricam Nc3001 Totem- The Nc3001 is designated as a Parallel Signal Processor which is composed of 32 digital signal processors. The architecture allows to have 1 to 32 neurons implemented. The precision used is 32 bits, with circuitry for external Look Up Table (LUT) RAM to implement the activation functions. The multipliers used are Baugh-Wooley with pipeline.

Neuricam Nc3003 Totem- The Nc3003 is an evolution of the Nc3001 though the two chips are very similar. Looking only to the characteristics chosen to analyse the chips the difference is a larger memory available for storing the weights. There is also improvement in the internal control of the circuit.

RC Module NM6403- This circuit was designed by RC Module and produced by Samsung. The circuit is composed of 32/64 bit RISC (Reduced Instruction Set Computer) Core and a 64-bit co-processor to support vector operations of variable length and is a combination of Very Long Instruction Word (VLIW)/SIMD architectures.

Systolic Array: In systolic array solutions each processor performs one step of the calculation (always the same step) before passing the result to the next processor, forming a pipeline.

Examples of this type are:

Siemens MA-16- This chip performs fast matrix operations of 4x4 matrices with 16 bit elements. It has 16 complex PEs (called elementary chain) (Ienne et al., 1995) and the weights and activation functions are stored off-chip (the latter with a Look Up Table).

Radial Basis Functions: This type of NN usually requires specific hardware though until now not many commercial solutions are available.

Examples of this type are:

Nestor/Intel NI 1000- This circuit developed jointly by Nestor and Intel can store up to 1024 prototypes with 256 dimensions, 5 bits per dimension. The RCE and PNN algorithms are available for on-chip learning, with the possibility of microcode any other. According to (Neuronet) this chip has been withdrawn from marketing by Intel but will continue to be manufactured for Nestor under a 15-year agreement.

IBM ZISC036- The IBM Zero Instruction Set Computer contains 36 neurons of 8 bit precision. The chip implements on-chip learning for ROI and has built in facilities to easily cascade multiple chips.

Silicon Recognition ZISC 78- This chip prepared for automatic pattern recognition through the use of KNN and the norms L1 and LSUP has 78 neurons. The input vectors can have 1 to 64 components of 8 bits.

Other Chips: Some of the developed circuits do not fit in the classes above because of their characteristics. An example is the SAND circuit that can be classified either as a systolic array or RBF.

SAND/1- The Simply-Applicable Neural Device is a solution designed with a sea-of-gates technology that could be classified in several classes. It consists of four PE, off-chip weights and transfer functions and forty bits precision.

This circuit also needs an additional sequencer circuit to perform memory management and control the SAND circuit.

MCE MT19003- This circuit is classified as a NISP (Neural Instruction Set Processor), which is a very simple RISC with only seven instructions that are optimised for the implementation of multilayer feedforward ANN.

4.3. Hybrid Implementations

The hybrid implementations appear as a mean to try to obtain the best of analogue and digital systems. Typically the external input/outputs are digital to facilitate integration into digital systems, while internally some or all the processing is analogue (Lindsey, et al., 1994).

Table 3 shows the resume of the characteristics of the hybrid implementations.

AT&T ANNA- AT&T Artificial Neural Network ALU is digital from the user's point of view but analogue in the inside. The weights are stored as capacitor charge refreshed periodically. This chip

Table 3 – Neural networks hybrid hardware implementations.

Name	**Architecture**	**Learn**	**Precision**	**Neurons**	**Synapses**	**Speed**
AT&T ANNA	Feedforward, ML	No	3bx6b	16-256	4096	2.1GCPS
Bellcore CLNN-32	FCR	Boltzmann	6bx5b	32	992	100MCPS 100MCUPS
Mesa Research Neuralclassifier	Feedforward, ML	No	6bx5b	6	426	21GCPS
Ricoh RN-200	Feedforward, ML	BP	N.A.	16	256	3.0GCPS

provides a variable number of neurons (from 16 to 256) and has no built in learning algorithm.

Bellcore CLNN-32- This circuit is similar to the ANNA since the inputs are also digital and internal processing is analogue. In this case the learning of the Boltzmann algorithm is built in. It implements fully connected recurrent networks (Alspector et al., 1992).

Mesa Research Neuroclassifier- this circuit built by the Mesa research Institute at the University of Twente has analogue inputs and outputs with digital weights of 5 bits. The speed that is claimed reaches the 21 GCPS, which is the highest rate of performance announced.

Ricoh RN-200- This chip implements a different method of emulating ANN: pulse rates or pulse widths (Lindsey, et al., 1994). This is an evolution of the Ricoh RN-100, which also had an adapted Backpropagation algorithm. An array of 12 Ricoh RN-100 was used to learn how to balance a 2-D pendulum in just 30 seconds (Lindsey, et al., 1994). This version has 16 neurons each with 16 synapses and achieves a performance of 3GCPS.

5. Difficulties Found in this Work

It is worth to point out the difficulties found to develop this work because they are related to its utility.

The main problem is to obtain information from the manufacturers. As strange as it might seem and resulting probably from the fact that these products are not the most sold on retail basis, most of the enquiries made to the manufacturers where left unanswered.

This results in some of the information present in this paper being based on previous surveys instead of the supplier.

Therefore the authors were unable to verify that all the circuits are still available and to cross validate all the information using the product datasheets.

Some of the chips included in this paper are included as a reference since they are no longer available from the manufacturer. Nevertheless, some of these circuits might still be purchased from some suppliers.

Because of the difficulty to gather information on these products, a few more recent circuits have not been included, since it was not possible to find enough information to classify them in order to make them available for a selection decision.

On the other hand, these difficulties reported here make this kind of surveys even more important to anyone who starts to seek information on this field, since it would take an enormous effort to gather all this information from scratch.

6. Conclusions

A few new neurochips are reported in this survey while the information collected indicates that more neurochips are no longer available commercially.

The new solutions that have appeared indicate that this field is still active, but the removal of the market of other solutions does not seem to be good news.

As (Heemskerk, 1995) indicates, neurocomputer building is expensive in terms of development time and resources, and little is known about the real commercial prospects for working implementations. Moreover, there is no clear consensus on how to exploit the currently available VLSI and even ultra large-scale integration (ULSI) technological capabilities for massively parallel neural network hardware implementations. Another reason for not actually building neurocomputers might lie in the fact that the number and variety of (novel) neural network paradigms is still increasing rapidly. For many paradigms the capabilities are hardly known yet. Paradoxically, these capabilities can only be tested in full when dedicated hardware is available.

These might be the reasons for the slow development of the ANN hardware market in the last years, but the authors believe that this situation will change in the near future with the appearance of new hardware solutions.

In the user perspective, taking into account the information given here about the existing market, it should be noted that there is no “best” solution for every case but the most suitable solution should be found for each case. This is the reason why the authors decided not to make a performance comparison.

7. References

Alspector, J., T. Jayakumar, and S. Luna, “Experimental Evaluation of Learning in a Neural Microsystem”, Advances in Neural Information Processing Systems - 4, ed. J. Moody et al., 871-878, Morgan-Kaufmann Pub., San Mateo, Ca. 1992.

Aybay, I., S. Çetinkaya and U. Halici, “Classification of Neural Network Hardware”, Neural Network World, IDG Co., Vol. 6, Nº1, pp11-29, 1996.

Baratta, D., G. M. Bo, D. D. Caviglia, f. Diotalevi and M. Valle, “Microelectronic Implementation of Artificial Neural Network”, Invited paper in the proceedings of the 5th Electronics Devices and Systems International Conference, Brno, Czech Republic, June 1998.

Caviglia, D., “NeuroNet Roadmap - Preliminary Draft”, part a) Hardware Implementations, available in the internet at

http://www.kcl.ac.uk/neuronet/about/roadmap/hardware.html

Cybenko, G., "Approximation by Superposition of a Sigmoidal Function, Mathematics of Control, Signals and Systems, 2 pp.492-499,1989.

Dias, F. M. and A. M. Mota, "Comparison between different Control Strategies using Neural Networks", 9th Mediterranean Conference on Control and Automation, Dubrovnik, Croatia, 2001.

Heemskerk, J. N. H., "Overview of Neural Hardware. Neurocomputers for Brain-Style Processing. Design, Implementation and Application", PhD Thesis, Unit of Experimental and Theoretical Psychology, Leiden University, the Netherlands, Draft version available from ftp.mrc-apu.cam.ac.uk/pub/nn, 1995.

Holler, M., "VLSI Implementation of Learning and Memory Systems: A Review", Advances in Neural Information Processing Systems 3, Morgan Kaufmann, San Mateo, Ca. 1991.

Hopfield, J.J., "Neural Networks and Physical Systems with Emergent Collective Computational Abilities", Proceedings of the National Academy of Sciences USA, 79, 2554-2558, 1982.

Ienne, P. and G. Kuhn, "Digital Systems for Neural Networks", P. Papamichalis and R. Kerwin, editors, Digital Signal Processing Technology, volume CR57, of Critical Reviews Series, pag. 314-345, SPIE Optical Engineering, Orlando, Fla. 1995.

Jedrasik, P., "Hardware implementation for real time RBF-based proximity effects neurocorrector", chapter 8 of "Proximity Effects Correction in Electron Beam Nanolithography, University of Anrwerpen, Departement Natuurkunde, Antwerpen, 1998, available in the internet at http://143.129.203.3/visielab/theses/jedrasik/chap80.pdf

Keulan, E. van, S.Colak, H. Withagen and H. Hegt, "Neural Networks Hardware Performance Criteria", Proceedings of the IEEE Conference on Neural Networks, III, 1885-1888, 1994.

Kohonen, T., "Self-organization and associative memory", 3rd edition, Berlin, Germany: Springer-Verlag, 1989.

Liao, Y., "Neural Networks in Hardware: A Survey", available in the internet at http://wwwcsif.cs.ucdavis.edu/~liaoy/research/NNhardware.pdf

Lindsey, C. and T. Lindblad, "Review of Hardware Neural Networks: A User's Perspective", Proceeding of 3rd Workshop on Neural Networks: From Biology to High Energy Physics, Isola d'Elba, Italy, Sept. 26-30, 1994.

Lindsey, C., B. Denby and T. Lindblad, "Neural Networks in HEP", available in the internet at http://www1.cern.ch/NeuralNets/nnwInHepHard.html.

Lindsey, C., "Neural Networks in Hardware: Architectures, Products and Applications", available in the internet at http://www.particle.kth.se/~lindsey/HardwareNNWCourse/home.html

Lindsey, C., B. Denby and T. Lindblad, " FutureAI.com – Artificial Intelligence – Neural Network Hardware", available in the internet at fttp://www.futureai.com/tutorials/hardware.php

Lindsey, C., T. Lindblad, "Survey of Neural Network Hardware", Proc. SPIE Vol. 2492, p. 1194-1205, Applications and Science of Artificial Neural Networks, Steven K. Rogers; Dennis W. Ruck; Eds. , March 1995.

Minsky, M. and S. Papert, "Perceptrons", Cambridge, MA: MIT Press,1969.

Rumelhart, D. E., and J.L. McCelland (Eds.), "Parallel Distributed Processing", Volume I: Foundations, Cambridge MA: MIT Press, 1986.

Schoenauer, T., A. Jahnke, U. Roth and H. Klar, "Digital Neurohardware: Principles and Perspectives", Neuronal Networks in Applications, Magdeburg, 1998.

Schüffny, R., A. Graupner and J. Schreiter, "Hardware for Neural Networks", 4th International Workshop Neural Networks in Applications, Magdeburg, Germany, 1999.

Sussner, P., "Perceptrons," in Encyclopedia for Electrical and Electronics Engineering, John Wiley & Sons, Vol. 16, pp. 44-59, 1999.

Taveniku, M., and A. Linde, " A Reconfigurable SIMD Computer for Artificial Neural Networks", Technical Report nº189L, Chalmers University of Technology.

Adaptive Solutions home page, http://www.asi.com/

"Digital Neural Networks in VLSI", Module Research Center, available in the internet at http://www.module.ru

"FAQ on Neural Networks part 7 – Hardware NNW", available in the internet at ftp//:ftp.sas.com/pub/neural/FAQ.html

"NC3001 Parallel Signal Processor (PSP)", Data Sheet, available in the internet at http://www.neuricam.com

"NC3001 TOTEM - A Digital Processor for Neural Networks", Product description – 14 August 1998, available in the internet at http://www.neuricam.com

"NC3003 TOTEM Digital Processor for Neural Networks", Data Sheet Rel. 12/99.

"Neural Networks: Commercial Hardware Tools", available in the internet at http://www..emsl.pnl.gov:2080/proj/neuron/neural/systems/commercial.html

NEuroNet Research Committee, "A European Strategy for Neural Networks", available in the internet at http://www.kcl.ac.uk/neuronet/about/roadmap/hardware.html

NM6403 Digital Signal Processor, RC Module Data Sheet v1.0

"SAND NeuroChip", Infosheet, February 1997

ZISC78 Datasheet, Silicon Recognition INC, May 2002.

Fernando Morgado Dias received his *Diplôme D'Études Approfondies* in Microelectronics from the University Joseph Fourier in Grenoble, France (1995) and currently teaches at the Escola Superior de Tecnologia de Setubal, Portugal and is preparing his PhD degree at the University of Aveiro, Portugal.

Ana Antunes received her *Diplôme D'Études Approfondies* in Microelectronics from the University Joseph Fourier in Grenoble, France (1995) and currently teaches at the Escola Superior de Tecnologia de Setubal, Portugal and is preparing her PhD degree at the University of Aveiro, Portugal.

Alexandre Manuel Mota received his PhD in Automatic Control from the University of Aveiro, Portugal (1993) and currently teaches at the University of Aveiro. He has published more than 40 papers in the fields of Automatic Control and Distributed Systems.