SEU FAULT TOLERANCE IN ARTIFICIAL NEURAL NETWORKS†

R. Velazco, A. Assoum, N.E. Radi
Laboratoire de Génie Informatique (LGI /IMAG)
46 Av. Félix Viallet, 38031, Grenoble 1, France

R. Ecoffet
Centre National d'Etudes Spatiales (CNES)
18 Av. Edouard Belin, 31055, Toulouse, France

X. Botey
Universtat Politécnica de Catalunya (UPC/ETSETB)
C. Gran Capità, s/n 08034, Barcelona, Spain

Abstract

In this paper we investigate the robustness of Artificial Neural Networks when encountering transient modification of information bits related to the network operation. These kinds of faults are likely to occur as a consequence of interaction with radiation. Results of tests performed to evaluate the fault tolerance properties of two different digital neural circuits are presented.

I. INTRODUCTION

Since their appearance, integrated circuits implementing "classical" digital processors based on Von Neuman architecture (micro-processors, digital signal processors, micro-controllers) are increasingly being used in all fields of human life and in every kind of thinkable application.

In view of their utilisation in space applications, Very Large Scale Integration (VLSI) processors have been the subject of numerous studies. These studies focused mainly on the examination of space radiation effects on their behaviour, the development of test strategies and the qualification of some existing circuits.

Besides the processors mentioned above, an alternative architecture of information processing systems has been developed: the so-called *Artificial Neural Networks* (ANN) [1]. Inspired by biological organisation schemes, ANN offer powerful and compact solutions to a wide range of problems, some of them difficult to be tackled without massively parallel systems. As examples of such application fields we can mention: speech and image recognition, control, robotics, signal processing, optimization and data classification.

Most studies concerning ANN and their applications were accomplished by means of software simulations. But recently, there has been more interest in the development of hardware implementations (analog and digital) for ANN models designed for commercial and industrial applications [2].

Among the few published ANN implementations which were devoted to space applications, one can mention the work presented in [3] where a thermal noise receiver and a time domain sampler[1] were implemented as neural networks on a digital signal processor and a general purpose micro-processor[2]. Nevertheless, despite the potential utilisation of ANN for powerful data processing on board satellites and in other harsh environments, there are few reported studies on the effects of radiation on neural circuits, especially relating to Single Event Upsets (SEU). To our knowledge, the only published work in this area concerned the effects of absorbed doses on an analog neural network [4].

In this paper, we study the fault tolerance of some simple ANN digital implementations when coming up against faults that can be modelled by the modification of one bit of information. The modified information may correspond either to the network input data (stimuli) or to the parameters defining the network configuration and is supposed stored within the integrated circuit in static memory cells.

The paper is organised as follows: the basic principles of Artificial Neural Networks and their ability to tolerate certain faults are briefly presented in §II. For the purpose of performing SEU tests, two VLSI digital neural circuits have been chosen and trained to solve different problems; their main features and the experimental setup are presented in §III. Results of ground tests as well as SEU simulations are presented and discussed in §IV. Finally, conclusions and future work are given in §V.

II - BACKGROUND

A) Basic principles of ANNs

From a structural point of view, an Artificial Neural Network is composed of a number of processing units, the *artificial neurons*, interconnected through variable synaptic strengths, called *weights*. The number of neurons depends essentially on the problem to be solved. The response of a neuron is a non-linear function of the weighted sum of its inputs. Neurons are connected according to different organisational principles to build the proper network architecture [5].

The simplest ANN is the *perceptron;* it consists of a single neuron that receives signals of an input pattern from N input units, as shown in fig. 1. The output state, σ_O, of the perceptron is a function *f*, usually a step or a sigmoidal

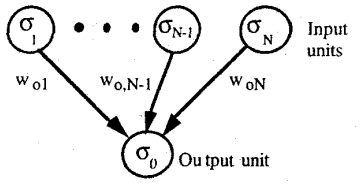
† Work supported in part by the CNES under contract 8442950027.

[1] The WAVES experiment implemented on the WIND satellite

[2] The ADSP-2100 from Analog Devices and the SA-3300 from Sandia respectively

0018-9499/95$04.00

function, of the weighted sum of its inputs: where w_{oi} is the synaptic weight of unit i on the output neuron, and θ_o is the output's threshold (eq. (1)).

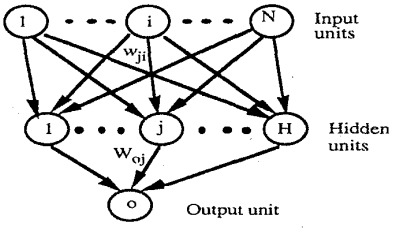


$$\sigma_o = f\left[\sum_{i=1}^{N} w_{oi}\sigma_i - \theta_o\right] \quad (1)$$

Figure 1: A perceptron with N inputs and a single output

More complicated architecture are mere assemblies of perceptrons, in which some of them play the role of input units for the others, depending on the connectivity pattern. For example, a feed forward Multiple Layered Perceptron (MLP) with one layer of H *hidden units* is shown fig. 2.

The response σ_0, to an input pattern σ_i, $1 \le i \le N$, of an MLP with one hidden layer is given by eq. (2), where the activation functions f and g are generally taken to be the same. The weights w_{ji} are associated to connections between input and hidden units, while W_{oj} are those between hidden units and the output. The thresholds θ_j, and θ_o correspond to the hidden units and the output unit respectively. MLPs with more than an output neuron are a straightforward generalization of the one of fig. 2.



$$\sigma_o = g\left[\sum_{j=1}^{H} W_{oj}\, f\left(\sum_{i=1}^{N} w_{ji}\sigma_i - \theta_j\right) - \theta_o\right] \quad (2)$$

Figure 2: An MLP with one hidden layer of H units.

Usually, weights and thresholds have to be determined to adapt the ANN to a given task. For example, in character recognition, if the input pattern is a more or less noisy bitmap of a character, the aim is to design an ANN such that its output is the code of the character.

The search for the ANN adapted to solve a given problem is called *training*. It is done through a learning algorithm which aims to find weights such that the network gives the correct response to a set of examples of the problem. Clearly, the solution obtained depends on the algorithm itself, and on the size of the training set. In particular, the perceptron is able to learn without errors only one class of problems: those which are linearly separable (the outputs can be separated by a hyperplan in patterns' space). For more involved tasks, more complex architectures (MLP or others) are needed.

There are thus three degrees of freedom when defining the neural network functionality: the type of non-linearity of the processing units, the connectivity patterns between units and the training algorithm to be used. The different neural network models available are obtained by combining appropriate choices for those parameters.

B) Fault tolerance of ANNs

Among the main properties of Artificial Neural Networks, their fault tolerance is an intrinsic characteristic often cited as one of their relevant advantages in comparison with other information processing systems. The fault tolerance measures their capability to perform the desired task under fault conditions (erroneous information) and to maintain their computing ability when a part of the network is damaged or removed. This property stems mainly from the distributed nature of ANN parallel computation: as each node contributes to the final response, damage has to be extensive before the network response degrades seriously. The network therefore demonstrates gradual performance degradation rather than catastrophic failure [6].

Studying the ANNs behaviour under fault conditions can be useful in three ways: (i) it gives information about the network's robustness in relation to internal perturbations; (ii) it can be used as a criterion for selecting among multiple network solutions to a given problem; and (iii) it provides a framework for relating the performance of the network as a whole to the performance of its components (the neurons).

Previous research on the area of ANN fault tolerance focused on the following aspects:

(i) neuron errors: disabling one or more neurons during the training phase [7],

(ii) weight errors: destroying (zeroing) weights between neurons during the recall phase [8] and introducing weight perturbations either during the recall phase [9] or the training phase [10],

(iii) input pattern errors: injecting either input perturbations during the recall phase [11] or training data perturbations during the training phase [12-14].

All the studies mentioned above proved that the considered perturbations have practically no incidence on the network performance when they were small in comparison with the initial information (the noise percentage should not exceed 50 % of the initial information). In addition, it should be noted that in some cases, perturbations during training could have beneficial effects on the ANN performances: reducing the training time [11] and improving the quality of training as measured by the network generalization ability during the recall phase [11-15].

The SEU effects on ANN have not been thoroughly studied yet in spite of the good response time and theoretical fault tolerance featured by presently available dedicated VLSI. This would make them potential candidates for applications designed to operate in harsh environments.

A block used in space applications for power current monitoring was modelled using an ANN approach in [16].

SEUs were then simulated on the model. The results obtained revealed two facts: firstly, the ANN features a good fault tolerance for the injected SEUs faults (inversion of each neuron output), secondly the response time was better under fault conditions than during standard operation.

Although it may be considered as a particular case of noise, the effects of SEU on ANNs must be carefully studied. Indeed, the magnitude of the perturbation induced by an SEU on a bit of information will depend on both the data format and the location of the perturbed bit within the coded information. It is quite easy to show that in the case of integer format, an SEU may lead to a corrupted value exceeding largely the initial non perturbed value. For instance, if the information is an n-bit 2's complement binary coded value, an SEU occurring on its most significant bit will lead to the addition or subtraction of 2^{n-1} depending on whether the initial value is negative or positive. The perturbation ratio is related to the initial value and can reach 2^{n-1} when the initial value is 1. This corresponds to a noise of 800% when n = 4. For floating point format the consequences of SEUs will depend on the corrupted field: they could be catastrophic for the exponent, serious (200%) for the sign bit, and irrelevant for the less significant bits of the mantissa.

III. Experimental Details

A) Target circuits

It is in hardware implementations that the particular properties of neural networks are best exploited [17-18]. The first circuits have been based on analog technologies, with the advantage of speed and compactness, but presenting severe limitations. The main drawbacks concern their programmability (possibility to implement different networks), the accuracy of synaptic coefficients (implemented as resistors) and the sensitivity to noise (especially crosstalk) [19-22].

Various digital circuits have emerged in the last years. They are capable of emulating neural networks in a flexible way and with a good immunity to noise, thanks to digital technology that allows the implementation of any function with arbitrary accuracy and quasi-infinite storage time [23-28]. Components with in-circuit learning capabilities have also been developed [17, 26, 27].

We performed SEU experiments on two digital neural circuits. Both circuits studied can be used either alone or as a part of an array of processors (massively parallel architecture) to achieve high speed performances. The experiments presented next sections were performed on single neural chips architecture.

The UTAK chip[3] (Unit for Trained Adaptive Knowledge)

The UTAK chip [28] is a general purpose neural circuit built around a full custom digital processor. Its first version, which does not implement the training algorithm, has a set of 16 instructions and an internal RAM of 128 bytes used to store the synaptic weights and thresholds needed to perform the neural computations.

To emulate a neuron, the UTAK chip is exercised by an external host that feeds it with the corresponding weights, inputs and thresholds. In practice, a single UTAK can be programmed to emulate any ANN. When high performances are looked for, many UTAKs can be used together (in an array shape) to support complicated networks. The connection between UTAKs is accomplished by means of 3 programmable communication ports. As the UTAK operation is based on a fully synchronous processor, all the chips in an array will work in parallel, thus optimizing the operation speed.

The L-Neuro chip [4] (Learning Neurochip).

The L-Neuro chip [27] is a commercially available circuit, manufactured by Philips as a hardware accelerator (co-processor) for neural calculations. It can be seen as a general purpose neural circuit which implements in parallel the training phase as well as the recall phase. L-Neuro has a set of 45 instructions and an internal RAM of 1 Kb where the synaptic weights are to be stored.

To emulate a given neuron, the synaptic weights and the corresponding inputs should be stored in the internal RAM and neural status registers respectively. The emulation of an entire neural network is achieved by a proper sequence of data loading and running of the L-Neuro. The latter can be easily controlled by an external host processor via a built-in interface bus. It should be noted that many L-Neuro may be used together, thus offering a faster global performance and allowing the neural computing system to be used for complicated applications with strong real time constraints.

The two target circuits comprise a variable number of storage cells (internal RAM, registers, latches, counters, etc.) potentially sensitive to SEU. Tab. 1 gives for each circuit the sensitive zones and the portion accessible to the user by executing the appropriate instructions.

Circuit	RAM	Registers	Total	Access. %
UTAK	128 bytes	87 bits	1111 bits	94.69
L-Neuro	1024 bytes	608 bits	8800 bits	99.63

Table 1: Sensitive zones of the circuits under test

B) Experimental setup

The SEU experimental setup (fig. 3) for the chips under study was developed around the Functional and Upset Tester (FUTE-16) system, which was specifically designed to cope with the SEU testing of VLSI processors [29].

[3] Manufactured by UPC& SIDSA (Madrid) in CMOS 1.2 µm.

[4] Manufactured by t he LEP Philips (Paris)-in CMOS 1.6 µm.

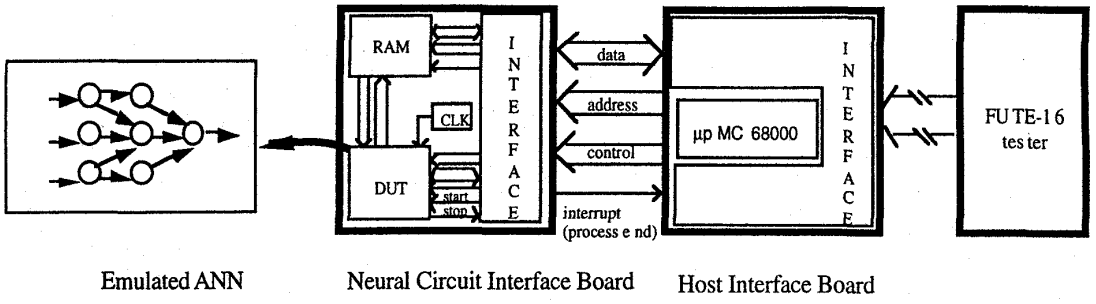


Figure 3: Experimental setup for neural chips SEU testing

To respond to the particularities of the considered neural chips, a host board was built around a well-known microprocessor (the MC68000 in our case). This host board facilitates the communications between the tester and the specific interface boards built around each of the target chips. Thus, the control and the observation of the activity of the Device Under Test (DUT) are achieved via the execution of a 68000 program. This program, which includes all the necessary information (instructions and data) needed for the ANN emulation, is stored in the tester memory.

Exercising the chip under test requires the following three steps:

(a) initialisation: weights and neuron input data are loaded by the host in the corresponding memory area,

(b) execution: the neural chip is run to perform the neural computations,

(c) result collection: the results obtained are stored by the host in the FUTE-16 tester memory, for ulterior processing.

The previous three-step procedure should be repeated as many times as necessary to obtain the final response of the emulated ANN. In practice, the synaptic memory should be used without reinitialisation for the emulation of portions of the neural network.

C) Test sequences

Two kinds of test sequences were applied to the circuits during the experiments: a *static test*, where the circuit is considered as a set of memory cells (thus ignoring all the existing sensitive areas accessible only through ANN operation), and a *dynamic test*, where an ANN application is executed by the circuit. Previous experiments performed on classical processors [30] proved that static test sequence led to a significant over-estimation of SEU cross-section, compared to the ones obtained using dynamic test sequences.

The static test sequence consisted in loading all the accessible internal memory cells with known data, performing a wait loop to allow SEUs to occur, and finally downloading the memory content to compare it with initial values.

For dynamic tests, simple ANNs evaluating 3-input Boolean functions have been implemented (fig. 4). Aiming at investigating the influence of the network architecture (number of layers), on the SEU sensitivity, we have chosen to calculate the Boolean functions using three different Multiple Layer Perceptrons (MLP) with 1, 2 and 3 hidden layers respectively. These fully-connected networks (i.e. neurons of each layer are connected to all the neurons of the forward adjacent layer) were trained by the classical "back propagation algorithm" to determine the weights and thresholds. As an example we give in fig. 5 the MLP with 2 hidden layers.

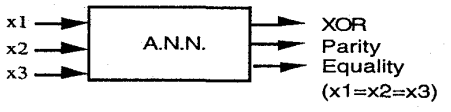


Figure 4: Implemented Boolean functions

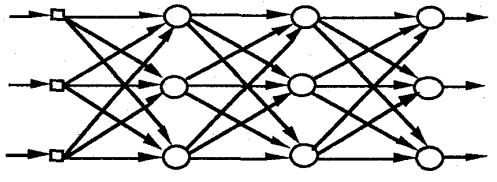


Figure 5: One of the tested ANNs:
A feed-forwarded MLP with 2 Hidden layers

IV. Obtained Results

A) SEU simulations

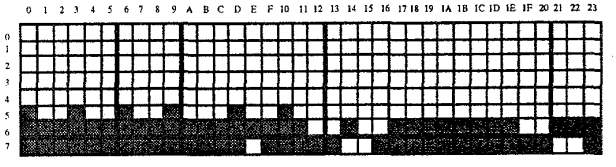
A first estimation of the SEU fault tolerance for the considered neural circuits was achieved by SEU-like fault injection in a subset of the sensitive memory cells. This fault simulation was carried out using the bread-board capabilities, and thus without the need of ground tests. These "off-beam" experiments consisted in the simulation of SEUs, by inverting single bits of the memory area accessible to the host board, prior to the exercising of the ANN application. Thus, the circuit under test will run the network with corrupted data.

The tested circuit thus operates under typical SEU conditions: if it issues a correct output to the corrupted configuration, SEUs occurring during "real life" in the corresponding memory cell location will be tolerated, otherwise, the SEU should be taken into account for the final

cross-section. An estimation of the final application SEU sensitivity, can be established from both the mapping of SEU-sensitive bits and a given SEU sequence (either randomly chosen or issued from static ground tests).

It should be pointed out that this kind of simulation would not be feasible for most classical processors, where the sensitive area is mainly composed of register banks that will be dynamically used when executing the application program (instructions and data are usually stored in an external RAM). SEUs occurring during a program execution will have different effects depending on the significance of the register content making it necessary to take into account, in addition to its random location, its random instant of occurrence.

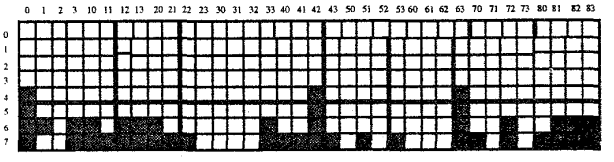
We performed such SEU simulations for the two considered circuits, obtaining the memory mapping of the SEU-sensitive cells. As an example, figs. 6 and 7 show the SEU-sensitive memory mapping of MLP2 for the UTAK and L-Neuro chips respectively. Memory cells are identified by their address: bit number (row) and byte number (column).



■ the SEU leads to an error on at least one ANN output

□ the SEU is tolerated by the ANN

Figure 6: SEU-sensitive mapping for the occupied area of the internal RAM of the UTAK (67 sensitive bits out of 1K bits).



■ the SEU leads to an error on at least one ANN output

□ the SEU is tolerated by the ANN

Figure 7: SEU-sensitive mapping for the occupied area of the internal RAM of the L-Neuro (34 sensitive bits out of 8192 bits).

Tab. 2 gives the results for the implemented ANNs, as the percentage of the entire RAM locations (where injected SEUs have no effects on the network's output).

tested network	UTAK	L Neuro
MLP1	93.9%	99.6%
MLP2	93.4%	99.5%
MLP3	86.8%	99.0%

Table 2: Results of SEU software injection

As expected, both circuits tolerate a large proportion of SEU-like faults in their internal memory used to store synaptic weights. The observed difference is mainly related to the data format used by each of the chips. Indeed, for the UTAK, synaptic weights and neuron thresholds are coded with a specific 8-bit fixed-point format, while the L-Neuro uses a standard 8 bits 2's complement format.

B) Ground tests

Unlike commonly performed ground tests, whose goal is the determination of the complete cross-sections (threshold and saturation cross-section) for Single Event Phenomena, we only aimed at getting evidence of the SEU-tolerance for different ANN implementations.

Achieving this, needs at least one particle beam having a Linear Transfer Energy (LET) able to induce enough SEUs, without risks of circuit damage by latchup. So, we performed ground tests with one Cf^{252} equipment[5] and a particle accelerator[6]. Since none of the studied chips existed in versions guaranteed to be latchup free, a first set of experiments was devoted to determining the range of LET suitable for our experimentation. The results of these preliminary "qualitative" tests can be summarised as follows:

- despite a high sensitivity to SELs, the SEU cross-section of the UTAK was evaluated with the Cf^{252} source. Owing to scheduling details, accelerator tests could not be completed.
- even if a few latchups were detected for the L-Neuro-chip, the corresponding Single Event Latchup (SEL) cross-section revealed one order of magnitude lower, compared to its SEU cross-section, allowing to get results about its SEU fault tolerance for different LETs.

Results obtained for the UTAK

As stated before, only Cf^{252} tests have been performed for this circuit. The SEU and SEL cross-sections derived from the static test of the sensitive area were respectively: $4x10^{-4}$ and $2{,}8\ x10^{-4}cm^2$/device. The analysis of the SEUs mapping showed that all the upsets have occurred on the internal memory.

Dynamic tests were performed with the three versions of Multiple Layer Perceptrons (referred as MLPi / i=1, 2 or 3). Tab. 3 gives the cross-sections derived from dynamic tests, as well as those extrapolated from the memory mapping of SEU sensitive cells, and the actual SEU sequence of static test (89 upsets). The small differences between measures and estimations showed in Tab. 3 are mainly due to the fact that neither the sequences of SEUs used to determine both figures were the same, nor the test program durations.

These results did not show big differences among the dynamic cross-sections of the emulated ANNs, the highest measured cross-section being obtained for the ANN with more hidden layers. So, redundancy (additional resources) which is,

[5] The CIRIL source available at CERT/DERTS (Toulouse, France)

[6] The Tandem Van de Graaff facility of the Insitut of Nuclear Physics (IPN, Orsay, France)

in general a positive factor [31], is not necessarily a feature of ANNs systems. This fact has been mentioned also in [9].

σSEU	MLP1	MLP2	MLP3
Measured	$2.25x10^{-5}$	$3.96x10^{-5}$	$7.6x10^{-5}$
Evaluated	$1.78x10^{-5}$	$8.9x10^{-6}$	$2.23x10^{-5}$

Table 3: Measured and extrapolated dynamic SEU cross-sections (cm2/device) for the UTAK-chip

Results obtained for the L-Neuro

Preliminary tests performed with the Cf^{252} facility showed that the L-Neuro has an SEU static cross-section of $1,1x10^{-3}$ cm^2/device. As for the UTAK, the analysis of the SEUs mapping showed that all the upsets have occurred on the internal memory.

More complete tests were performed with the accelerator. Static cross-sections for the used ions are given in Tab. 4.

Results are summarised in Tab. 4 show us quite good behaviour of the L-Neuro chip in relation to SEU: few latchups were detected, SEU saturation cross-section is probably of the order of $10^{-3}cm^2$/device for a LET of 36 Mev/mg/cm^2

ION	LET	σ_{SEU}	σ_{SEL}
Fl (55°)	7.3	$1.4x10^{-4}$	$1.6x10^{-5}$
Cl	12.7	$3.3x10^{-4}$	$2.2x10^{-5}$
Br	36	$3.5x10^{-3}$	$1.6x10^{-4}$

Table 4: SEU and SEL cross-sections (cm^2/device) obtained for L-Neuro with a static test strategy

Dynamic tests for each of the three MLPs were run by the L-Neuro while exposing it to Chlorine beam. The results are similar to those obtained for the UTAK: the number of layers is not necessarily a positive factor in an ANN system (Tab. 5).

L-Neuro chip	MLP1	MLP2	MLP3
Measured σ_{SEU}	$2.6x10^{-6}$	$2x10^{-6}$	$5x10^{-6}$
Estimated σ_{SEU}	$<4.1x10^{-6}$	$<4.1x10^{-6}$	$<4.1x10^{-6}$

Table 5: Measured and Estimated dynamic SEU cross-sections (cm^2/device) obtained with Chlorine for the L-Neuro-chip

None of the SEUs occurring during irradiation (while running the static tests) corresponds to a sensitive memory cell of the dynamic test sequences. Thus, the estimated cross-sections given in Tab. 5 are upper-bounds.

The static and dynamic SEU cross-sections for different ion species are summarised in Fig. 9. Dynamic cross-sections are averaged on extrapolated results obtained for the three MLPs. The drop-off observed at LET 12,7 Mev/mg/cm^2 (Chlorine) can be explained by the difference between the SEU sequences used to perform the extrapolation.

C) Discussion

The correlation between measured and estimated cross-sections for both studied circuits is quite good. This shows that static test results could be used for ANNs to extrapolate the SEU cross-section of the final application.

As for standard processors, irradiations performed while running the static test sequence lead to an over-estimation of the SEU risk: 20 times for the UTAK chip, 100 times for the L-Neuro. The difference comes from the specific data format used by the UTAK which is more sensitive than the one used by L-Neuro. In fact, the over-estimation values mentioned above should be considered as worst case figures. Indeed, no SEUs were detected for the dynamic test sequences, so, idle loops were introduced to increase the SEU probability (to cope with accelerator timing constraints).

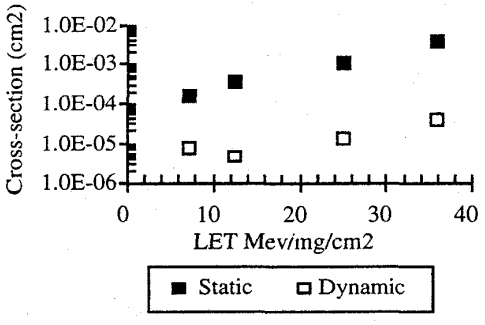


Figure 9: Static cross-section and extrapolated dynamic cross-sections for the L-Neuro

To get a rough idea, more than 90% of the neural circuit execution time was devoted to the waiting loops. The actual cross-section is then probably two orders of magnitude lower.

Moreover this performance is achieved without any hardening, neither at the process level nor at the design level. Furthermore, the dynamic test sequences used all along this study, were probably not complex enough to drastically bring out the inherent ANN's Fault Tolerance mechanisms.

V. Conclusions & Future Work

This paper aimed at presenting some evidence on the fault tolerance of Artificial Neural Networks hardware implementations. SEU injection experiments, performed on two digital neural circuits, allowed us to identify, for three simple ANNs, the bit locations sensitive to SEUs within the internal memory area used to store the network parameters. Depending on both the considered circuit and the implemented ANNs, this SEU simulation reveals a significant Fault Tolerance ranging between 86% and 99% (given as the

percentage of memory cells where SEUs do not provoke output errors). The obtained mapping of the SEU-sensitive locations was used to estimate the application's cross-section. Then, irradiation tests are to be performed only once, both to determine the static cross-section and to collect actual SEU sequences.

Ground tests (Cf^{252} and accelerator) allowed us to determine the actual and the estimated SEU cross-sections for simple artificial neural networks. Owing to the accelerator schedule and the sensitivity to latchup, we could obtain test data only for limited ion species. The measured and estimated cross-sections show a quite good correlation and ranged between 10^{-6} and 10^{-5} cm^2/device. Moreover, these values are to be considered as worst case making these circuits potential candidates for space application (if manufactured with a process immune to the latchup phenomenon).

Future work includes the implementation and test of a complex Artificial Neural Network, designed to be used in space applications (identification of X-ray spectra). We will also investigate the influence of the used training strategy on the SEU sensitivity to determine the weights values

ACKNOWLEDGEMENTS

This work is supported by the CNES, the authors are grateful to M. Labrunée, head of the Quality department, for his encouragement. We would also like to thank M. Duranton & J. Cabestany, from LEP Philips and UPC respectively, for their help with L-Neuro chip and UTAK chip.

REFERENCES

[1] S. Haykin, Neural networks: a comprehensive foundation, *Macmillan College Publishing Company, Inc.*, 1994.

[2] B. Angeniol, Industrial Applications of Neural Networks, *Neural networks: Biological Computers or Electronic Brains, Springer-Verlag Ed.*, 1990, pp. 65-70.

[3] Les réseaux de neurones formels et leurs applications spatiales , *Journée Thématique du CNES* , 1993.

[4] H. Castro, & M. Sweet, Radiation Exposure Effects on the Performance of an Electrically Trainable Artificial Neural Network (ETANN) *IEEE Trans. on Nuclear Science, vol. 40, no. 6,* 1993, pp. 1575-1583.

[5] Rumelhart D.E., McClelland, J.L. & PDP Research Group. *Parallel Distributed Processing. Volume 1: Foundations.* 1, Cambridge: MIT Press, 1986.

[6] R. Beale, & T. Jackson, Neural Computing: An introduction, *Bristol, UK, Adam Hilger,* 1990.

[7] C.H. Sequin,& R.D. Clay, Fault Tolerance in Feedforward Artificial Neural Networks, *Neural Networks: Concepts, Applications and Implementations*, 1991, vol. 4 pp. 111-114.

[8] M.D. Emmerson, R.I. Damper, Determination and Improving the Fault Tolerance of Multilayer Perceptrons in a Pattern-Recognition Application, *IEEE Trans. on neural networks. vol. 4, no. 5*, 1993 pp. 788-793.

[9] M. Stevenson, R. Winter, & B. Widrow, Sensitivity of Feedforward Neural Networks to Weights Errors., *IEEE Trans. on Neural Networks, vol. 1, no. 1,* 1990, pp. 71-80.

[10] A.F. Murray, & P.J. Edwards, Enhanced MLP Performance and Fault Tolerance Resulting from Synaptic Weight Noise During Training, *IEEE Trans. on Neural Networks,vol. 5, no. 5*, 1994, pp. 792-802.

[11] A.F. Murray, Multi-layer perceptron learning optimised for on-chip implementation - A noise robust system, *Neural Computation, 1992, vol. 4, no. 3,* pp. 366-381.

[12] L. Holmstrom, & P. Koistinen, Using additive noise in back-propagation training, *IEEE Trans. on Neural Networks, 1992, vol. 3, no. 1.*

[13] C. Bishop, Curvature driven smoothing in back-propagation neural networks, *IJCNN, vol. 2,* 1990, pp. 749-752.

[14] K. Matsuoka, Noise injection into inputs in back-propagation learning, *IEEE Trans. on Syst. Man Cyber., 1992 vol. 22, no. 3*, pp. 436-440.

[15] C.T. Chiu, K.Mehrotra, C.K. Mohan & S. Ranka, Training techniques to obtain fault-tolerant neural networks, *Proc. FTCS-1994 pp. 360-369.*

[16] S. W.Welch, et al Fault tolerance behaviour of I^2t parallel computing network, *Proc. IJCNN, vol II, 1990, pp. 712-715.*

[17] M. Weinfeld, Neural Networks as Specialized Integrated Circuits: an Academic Exercice or the Promise of New Machines?, *Neural networks: Biological Computers or Electronic Brains, Springer-Verlag Ed.*, 1990, pp. 173-180.

[18] M. Weinfeld, Integrated artificial neural networks: components for higher level architectures with new properties, *NATO Adv. Workshop on Neurocomputing,* 1990.

[19] E. Spencer, Programmable bistable switches and resistors for ANN, *Neural Networks for Computing*, 1986, pp. 151-155.

[20] H.P. Graf, & P. de Vegvar, A CMOS associative memory chip based on neural networks, *International Solid State Circuits Conference*, 1987.

[21] M. Verleysen et al, Neural networks for high storage content-addressable memory: VLSI circuit and learning algorithm,.*IEEE journal of solid state circuits.* Vol. 24, pp. 562-569, 1989.

[22] M. Holler, S. Tam, H. Castro, An Electrally Trainable Artificial Neural Network (ETANN) with 10240 floating gate synapses, *Proc. of the IJCNN 1989*, Vol. 2, pp. 191-196.

[23] S. Jones, M. Thomaz, & K. Sammut, Linear systolic neural network machine, *IFIP Workshop on Parallel Architecture on Silicon*, Grenoble, 1989.

[24] V., Peiris et. al., A versatile digital building block for fast simulation of neural networks, *IFIP Workshop on Parallel Architecture on Silicon, Grenoble*, 1989.

[25] D. Jacquet, & G. Saucier, Design of a Digital Neural Network Chip: Application to Optical Character Recognition, *EuroAsic-EDAC-ETC Conf.*, Paris, 1994.

[26] M. Weinfeld, A fully digital CMOS integrated Hoppfield network including the learning algorithm, *VLSI for artificial intelligence. Delgado-Frias, J.G. & Moore, W. eds.* 1989 Kluwer Academic.

[27] N. Mauduit et al., L-Neuro 1.0: A piece of hardware LEGO for building Neural Network System, *IEEE Trans. on Neural Networks. Vol. 3, N° 3, 414-422,* 1992.

[28] F. Castillo , J.M. Moreno, & J. Cabestani, Digital VLSI implementation of a neural processor *Proc. Melecon* , pp. 307-310, *1991.*

[29] R. Velazco, et al, A low cost functional test system: the FUTE 16 tester, *Proceedings of the International Conference of Microelectronics (ICM),* pp. 5.1.1.1-5.1.1.4, 1992.

[30] Estreme, F. & al., SEU and latch-up results for SPARC processors, *IEEE Radiation Efects Data Workshop*, 1993. pp. 13-19.

[31] P.K.Lala, Fault toleant and fault testable hardware design, *Englewood Cliffs, NJ: Prentice-Hall*, 1985.