

Implementing a Self-Checking Neural System for Photon Event Identification by SRAM-based FPGAs

Monica Alderighi¹, Sergio D'Angelo¹, Vincenzo Piuri², and Giacomo R. Sechi¹

¹*Istituto di Fisica Cosmica “G. Occhialini”, Consiglio Nazionale delle Ricerche, Via Bassini 15, 20133 Milano, Italy, {monica, sergio, giacomo}@ifctr.mi.cnr.it.*

²*Dipartimento di Elettronica e Informazione, Politecnico di Milano, Piazza L. da Vinci 32, 20133 Milano, Italy, piuri@elet.polimi.it.*

Abstract

The paper presents and evaluates the design and the implementation of a self-checking neural system for photon event identification in Intensified Charge-Coupled Devices detectors. The neural approach reveals more effective than classical algorithmic approaches thanks to its learning through example ability. Implementation is accomplished by SRAM-based FPGAs, which have generated increasing interest in the space community. The adoption of suitable on-line fault detection techniques is illustrated taking into account in specific way SEU induced faults. The techniques are based on AN coding, particularly 3N coding, which constitutes a reasonable trade-off between circuit complexity and computational delay. Estimations of circuit area overhead and fault coverage are reported.

1. Introduction

The paper deals with the design and implementation of an innovative system for identification of photon events in Intensified Charged Coupled Devices (ICCD) detectors for space applications [1] by means of a neural architecture. The huge amount of CCD images to be analyzed and the small number of significant photon events suggest the use of onboard processing and dedicated high-performance architectures to limit the storage needs and support the continuous real-time control of the experiment.

The complexity of the photon event identification task, in which many factors concur and interfere to define both the characteristics of the object that must be detected and the experimental environment, discourages the use of algorithmic approaches. Indeed, algorithmic approaches are difficult to be specified at reasonable computational complexity because of uncertainty and noise, while many examples of CCD images can be easily collected. Artificial neural networks represent an interesting approach, due to their intrinsic computational parallelism and configurability through learning by examples. In the literature many papers are available on image classification and feature extraction by artificial neural networks, but none specifically deals with the image characteristics of our application [2-5].

A novel neural photon event identification system has been designed in order to satisfy the stringent requirements dictated by our application. The system has been developed using Field Programmable Gate Arrays (FPGAs), which have generated interest in the space community due to their ability to implement custom hardware solutions while still maintaining flexibility of digital signal processors through device programming. They also

offer the advantage of reduced development costs with respect to Application Specific Integrated Circuits (ASICs), especially in case of low production volume. Moreover, SRAM-based FPGAs allow unlimited in-system reprogrammability, thus increasing system flexibility.

Some fault-detection capabilities are needed in our photon event identification system, like in any digital system in critical applications and hostile environments, in order to guarantee the system operation and validate the outputs. Neural networks are claimed to have intrinsic fault-tolerance capabilities. Unfortunately, only a limited amount of intrinsic correction capacity is actually available [6] for some very specific classes of faults, some neural paradigms, and only if a suitable learning procedure is adopted. However, this capacity does not provide any information both on the correctness of each output and on the location of the faulty component. Intrinsic error masking may fail for some input sets or from a given time on, but no signal is generated to prevent subsequent use of erroneous data [7].

Besides, although rad-hard FPGAs are available to implement systems intended for the space radiation environment, sensitivity to Single Event Upset (SEU) induced faults is still an issue that needs to be dealt with. Concurrent error detection is mandatory to check every result in on-line processing systems. Physical modular redundancy and data codes (namely, AN codes [8, 9]) have been shown effective to achieve this goal at a limited circuit complexity increase. Experiments on SEU sensitivity of neural networks are presented in [10], and preliminary results about the SEU susceptibility of antifuse-based FPGA are reported in [11]. A preliminary design of a neuron with fault-detection capabilities is presented in [12], while the neural approach for event identification is discussed in [13].

The paper is organized as follows. Section 2 summarizes the photon event identification problem and the proposed neural approach and highlights some general properties of SRAM-based FPGAs for space astronomy applications. The self-checking neural architecture is presented in Section 3, while the analysis and coverage of SEU induced faults are given in Section 4. Some conclusions are drawn in Section 5.

2. The photon event identification system

The photon event identification system (Figure 1) consists of a high-gain electron multiplier based on MicroChannel Plates (MCP), a read-out system composed by a phosphor-screen fiber-optically coupled to a fast-scanned CCD camera, and a processing system for event identification [14]. On the CCD matrix, each photon event appears as a charge distribution having approximately a Gaussian profile and covering a 5x5-pixel area (called event window). Valid photon events are identified against bad or spurious ones by morphological analysis of event profile.

An *ad hoc* neural paradigm was defined for the event identification [13]. The neural network consists of 25 neurons, corresponding to pixels in the event window. Each neuron has 24 connections to the neighboring neurons and one feedback loop. The dynamic behavior of the neuron is defined by:

$$y_j(t+1) = \sigma_r \left[\sum_{i=1}^{25} \omega_i s_i(t) - \vartheta \right] \quad j = 1, \dots, 25$$

$$s_j(t+1) = \begin{cases} s_j(t) & \text{if } |y_j(t+1) - s_j(t)| < \alpha s_j(t) \\ 0 & \text{otherwise} \end{cases} \quad j = 1, \dots, 25$$

where $s_i(t)$ is the state of the i^{th} neuron at time t , $\alpha=0.25$ is the stability threshold, ω_i is the i^{th} input synaptic weight, ϑ is the neuron threshold, T is the temperature controlling the neuron evolution, and

$$\sigma_r(x_j) = \begin{cases} 0 & \text{if } x_j \leq -128T \\ \text{int}[128 + x_j/T] & \text{if } -128T < x_j < 128T \\ 255 & \text{if } x_j \geq 128T \end{cases} \quad j = 1, \dots, 25$$

Experiments showed that the event identification is accomplished in three iterations only. The neuron operates in 8-bit integer precision, which was proved to produce results practically equivalent to the floating-point representation.

The event identification problem is invariant under translation: the same set of synaptic weights can thus be used for all event windows. Moreover, the rotation symmetry of the charge distribution allows defining only six independent weights.

The architecture for event identification is mainly composed by the CCD Interface and the Event Identification Unit. The former prepares the event window, while the latter detects the significant events. A suitable neural network controller responsible for downloading the configuration parameters, saving and transmitting results, and communicating possible operating errors, manages the operation of the Event Identification Unit.

The CCD interface consists of four 512*9 FIFOs and five 9-bit registers (Figure 2). Register E loads pixels from the CCD, while the other registers load data from the previous FIFO in the chain. Register E contains the last pixel in the current row; registers D , C , B , and A hold the pixels in the same column but in the previous four rows. Registers contain, therefore, a column of the event window. A Eurocard-standard board hosts the CCD interface and the neural network controller.

The neural-based event identification unit consists of five identical Eurocard-standard boards, hosting five neurons each (Figure 3). Every neuron is implemented by one FPGA. Each neural board processes one row of the event window. The bus CMD is shared by all devices and is used by the controller to send commands to the neurons and download the network parameters. The bus $SIN25$ represents the neuron synapses: each neuron uses one bit of this bus to send serially its state to the other neurons, while the other 24 bits are input synapses. The neurons' states are initialized with the corresponding pixel values through the bus STA at the beginning of each window analysis cycle. Pixels from the CCD interface are loaded into $FPGA-1$. Pixels for the initialization of the other FPGAs are transferred through the chain of adjacent FPGAs so as to propagate the event window. Each initialization forces the window to move one pixel through the neurons. Possible run-time errors are sent to the controller through the bus $ERROR$ and, then, forwarded to a host computer. The bus NZ collects the event flags of all neurons: an event flag is activated if the corresponding neuron's state is non zero. Event flags are delivered through the NZ bus to the neural network controller that assesses the presence of a photon event by analyzing the event flag activation pattern.

In the current prototype, the neural network training is performed off-line and off-board; at the end of training, the network parameters are downloaded from the host computer. The space system will have autonomous learning capability to guarantee further adaptability.

For the actual application the Xilinx XC4013E [15] devices have been used. The specific technology adopted satisfies the requirements of the present application in the atmospheric environment and allows the implemented system to be retargeted for space

applications without additional design costs, by using the corresponding hardened devices XQR4013XL [16] so as to cope with radiation effects.

One of the major concerns in the design of space environment systems is the radiation sensitivity of electronic devices. Radiation induced effects are mainly due to protons and electrons trapped in the Van Allen belts, cosmic ray protons and heavy ions, and protons and heavy ions from solar flares. They are generally described in terms of Total Ionizing Dose (TID), which is related to the long-term absorption of radiation, and require the use of suitable radiation hardened electronics devices. However, the passage of a single ionizing particle through a device may cause some effects, termed Single Event Effects (SEEs), that has begun to be noticed and investigated only recently. SEEs may be distinguished in latch-up effects or Single Event Latchup, (SEL), and the so-called Single Event Upsets (SEU). The former may have a destructive nature, while the latter are generally transient pulses in combinatorial logic or bitflips in memory cells or latches.

Two main classes of FPGAs have gained interest for space applications, each one offering trade-offs in functionality, performance and reliability. Antifuse-based FPGAs are configured by fusing appropriate switches. SRAM-based FPGAs, instead, are configured by programming appropriate memory switch elements, usually composed of a memory cell and a pass-transistor.

SRAM-based have been said more susceptible to SEU than antifuse-based FPGAs, because of the high number of programmable memory locations, resulting in an increased SEU cross-section by approximately one order of magnitude. However, it should be noticed that latches and flip-flops in antifuse devices are equally sensitive to radiation-induced upset as the latches of SRAM-based FPGAs.

Xilinx recently published results about the probability of radiation-induced upsets as well as the likelihood of latch-up in commercial XC4000-series devices in the atmospheric radiation environment [17]. At increasingly energy levels, from 10 to 100 MeV, no latch-up was found and a few SEUs were detected at the highest radiation levels. The SEU cross-section was calculated as about 1.3 to 4.4×10^{-15} cm^2/bit , which is an order of magnitude below the lowest limit reported for commercial SRAMs. Also results were reported for the new rad-hard XQR4000-series devices in Low Earth Orbit missions [18], which proved hardness beyond 60,000 rad of TID, SEU resistance, and latch-up immunity at 100 MeV, that is a Linear Energy Transfer > 100 $\text{MeV} \cdot \text{cm}^2/\text{mg}$, limit beyond which a device is considered SEL immune.

In the current prototype FPGA configuration programs are stored into EPROMs; in the space flight system static RAMs will be used to allow for modifications during the mission.

3. The self-checking neuron

Suited techniques were adopted to implement neuron operations with concurrent error detection. The traditional single fault model [19] is appropriate for our application and also holds when radiation effects are considered. Parity code and AN codes (with $A=3$) [9, 19] are particularly suited for the memory storage and the arithmetic units of the self-checking neuron respectively. Data coding is considered a good compromise between circuit complexity and computational delay. For the CCD interface the parity coding is effective. Encoding should be performed in the CCD camera to guarantee protection of the interconnection path. In the present system release, coding is performed at the CCD interface input since we were not allowed to modify the CCD camera circuits. Communication between the CCD interface and neurons, as well as among neurons, is protected by physical duplication with output comparison. A TMR is used to protect the neural network controller.

The self-checking neuron in the Event Identification Unit consists of three main parts (Figure 4): the Weighted Sum Section (WSS), the Sigmoidal Function Section (SFS), and the Initialization Interface.

The Initialization Interface acquires pixel values and sets the initial neuron's state into the *OUT_Reg* register. The initialization phase is executed at the beginning of each event window analysis cycle. The WSS produces the neuron's activation signal by computing the weighted sum of inputs. Neurons communicate serially to limit the board wiring. Inputs are grouped into 6 classes (one for each independent weight) and summed up within each class. Each of these sums is then multiplied by the corresponding weight. The resulting products are finally added to generate the activation signal. Arithmetic operations are pipelined to achieve high throughput. The SFS generates the neuron's output from the activation signal. If this signal is out of a pre-defined range, the neuron's output is forced to the corresponding saturation value. Otherwise, the neuron's output is computed by dividing the activation signal by the temperature parameter. Division is performed by iterated subtractions. Due to the pipelined architecture, output generation is overlapped to the computation of the subsequent weighted sums.

Implementation of the specific coding techniques for the neuron is now described. Each FPGA receives the initialization data by the Initialization Interface (Figure 4). The parity code is checked at this point and the $3N$ -coded value is determined to be stored in the output *OUT_Reg* register. The $3N$ -coded weights are stored in the *Serial_in_Param_Reg* registers. The arithmetic operations generating the weighted sums are automatically protected by the use of the $3N$ code for both inputs and weights. Weighted inputs, weighted sums, threshold and temperature values, and non-linear activation functions are represented in the $9N$ code. It is worth noting that coding both inputs and weights leads to a circuit complexity higher than the one required by single-fault detection. However, this is acceptable to protect also the logical data paths. No intermediate checking or decoding is required to preserve the single-error assumption since no aliasing is induced either by possible reconvergent data paths or cycles. Possible pipeline registers' faults are considered as errors in the subsequent arithmetic units in this approach.

The output function is protected by the $3N$ code as well. Since division is performed by iterated subtractions, checking is needed after every iteration to avoid aliasing. Comparison to the saturation values and also the NZ flag are protected by circuit duplication with output comparison. Checking is performed within each neuron concurrently with the nominal computation by verifying the divisibility by the code generator (3 or 9 for the $3N$ or $9N$ codes, respectively) [9, 19]. Self-checking checkers are adopted to avoid error masking due to errors in the checkers themselves. The two-rail logic [19, 20] is used to protect the error signal propagation to the control board.

The circuit complexity of the self-checking neuron has been evaluated in terms of FPGA resources, i.e. the number of Configurable Logic Blocks (CLBs) and I/O pins. The neuron without fault-detection features occupies 327 CLBs and 110 I/O pins of the XC4013E device, while the self-checking neuron needs 415 CLBs (72% of the total CLBs) and 118 I/O pins (61% of the total pins). The circuit complexity overhead due to concurrent error detection is thus about 26.9%, while the interconnection complexity increase is 7.2%.

4. SEU fault analysis and coverage

SEUs and SELs for commercial and rad-hard Xilinx 4000-series devices have been measured in laboratory and estimates about SEU and SEL rates have been calculated for applications in the atmospheric and space radiation environments [18]. For a typical

application in the atmospheric environment using commercial devices, the SEU rate is on the average one bit error per 250,000 to 1 million hours, depending on the device family adopted. For a typical space application using rad-hard devices, the SEU rate is approximately one bit upset every 100 hours.

SEUs can be distinguished depending on whether they occur in the user memory cells or configuration cells (possible SEU-induced transient effects on data lines are not considered in this work). In the first case, Look-Up-Tables and Flip-Flops may be affected resulting in corrupted data. The $3N$ coding technique ensures a full coverage of such faults. In the second case, the internal multiplexers' selection bits of CLBs and programmable interconnections' switches may be subjected to SEU. Modifications in CLB's internal data paths result in wrong data and, therefore, are fully covered by the technique adopted.

As far as interconnections are concerned, an SEU may induce a bitflip in one of the six configuration cells of a programmable interconnection point. This affects the user defined connection scheme of the specific design implemented. The following three situations are possible: i) a line break, which causes a wrong value in some unit of the neuron and, as such, is covered by the $3N$ coding; ii) a connection of an unused line with an internal data line, which has no effect, i.e. the fault is intrinsically masked; iii) a short circuit between two data lines, which represents a possible bridging fault. In this last case, for wired-AND and wired-OR bridging, a wrong data flows through the neuron and, thus, can be detected. Any other kind of bridging producing an illegal logic level is difficult to be treated with on-line techniques and is not covered by the present approach. Off-line techniques, for instance I_{DDQ} techniques [21], are more appropriate to detect such kind of faults.

5. Conclusions

A configurable computing system for photon event identification was presented, consisting of a self-checking implementation of a neural network by Xilinx XC4000E devices. The system was conceived for onboard applications of astronomy instrumentation. The susceptibility to radiation induced faults was considered for the chosen technology. The on-line fault detection technique adopted, based on $3N$ coding, was proved effective to cover almost all types of SEU faults, with a circuit complexity increase of 26.9% CLBs and 7.2% I/O pins. Specifically, all user memory cell faults are covered. Regarding configuration cells, only some types of possible bridging faults, due to interconnection shorts, cannot be detected by the chosen approach.

Acknowledgement

We wish to thank Emiliano L. Gummati for his helpful technical assistance.

References

- [1] S. Bowyer and R.F. Malina, eds., *Extreme UltraViolet Astronomy*, Pergamon Press, 1991.
- [2] C.E. Cox, W.E. Blanz, "GANGLION - A Fast Field-Programmable Gate Array Implementation of a Connectionist Classifier", *IEEE Jour. of Solid State Circuits*, vol. 27, March 1992, pp. 288-299.
- [3] J.G. Eldredge and B.L. Hutchings, "Density Enhancement of a Neural Network Using FPGAs and Run-Time Reconfiguration", in *Proc. IEEE Workshop on FPGAs for Custom Computing Machines*, Napa, CA, USA, April 1994, pp. 180-188.
- [4] V. Salapura, M. Gschwind, and O. Maischberger, "A Fast FPGA Implementation of a General Purpose Neuron", in: [R. Hartenstein and M.Z. Servit editors], *Field-Programmable Logic: Architectures, Synthesis and Applications, Proc. of the 4th Int'l. Workshop on Field-Programmable Logic and Applications*, Prague, Czech Republic, Sept. 1994, Springer-Verlag.

- [5] M. Gschwind, V. Salapura, and O. Maischberger, "RAN²SOM: a Reconfigurable Neural Network Architecture based on Bit Stream Arithmetic", in *Proc. of the 4th Conference on Microelectronics for Neural Networks and Fuzzy Systems*, Turin, Italy, Sept. 1995.
- [6] C. Alippi, V. Piuri, and M. Sami, "Sensitivity to Errors in Artificial Neural Networks: a Behavioral Approach", *IEEE Trans. on Circuits and Systems - 1: Fundamental theory and applications*, vol. 42, June 1995, pp. 358-361.
- [7] S. Bettola and V. Piuri, "High-Performance Fault-Tolerant Digital Neural Networks", *IEEE Transactions on Computers*, vol. 47, March 1998, pp. 357-363.
- [8] T.R.N. Rao, *Error Coding for Arithmetic Processors*, Academic Press, NY, 1974.
- [9] V. Piuri, M. Sami, and R. Stefanelli, "Arithmetic Codes for Concurrent Error Detection in Artificial Neural Networks: the Case of AN+B Codes", in *Proc. Int'l. Workshop on Defect and Fault Tolerance in VLSI Systems*, Dallas, TX, 1992.
- [10] R. Velazco, A. Assoum, Ph. Cheynet, and M. Olmos, "S.E.U. Experiments on an Artificial Neural Network Implemented by means of Digital Signal Processors", *IEEE Trans. on Nuclear Science*, vol. 43, Dec. 1996, pp. 2889-2896.
- [11] R. Katz, R. Barto, P. McKerracher, and R. Koga, "SEU Hardening of Field Programmable Gate Arrays (FPGAs) for Space Applications and Device Characterization", *IEEE Transactions on Nuclear Science*, vol. 41, Dec. 1994, pp. 2179-2186.
- [12] M. Alderighi, E.L. Gummati, V. Piuri, and G.R. Sechi, "A FPGA-based Implementation of a Fault-Tolerant Neural Architecture for Photon Identification", in *Proc. ACM/SIGDA Symposium on Field Programmable Gate Arrays*, Monterey, CA, USA, February 1997, pp.166-172.
- [13] M. Alderighi, S. D'Angelo, F. d'Ovidio, E. Gummati, and G.R. Sechi, "An Advanced Neuron Model for Optimizing the SIREN Network Architecture", in *Proc. IEEE Int'l Conf. on Algorithms and Architectures for Parallel Processing*, Singapore, June 1996, pp. 194-200.
- [14] E.G. Tanzi, "Photon Counting and Analog Intensified Imagers for UV and X-Ray Radiation", *IFCTR-CNR Internal Report*, January 1995.
- [15] Xilinx, *The programmable Logic Data Book*, 1996.
- [16] Xilinx, *QPROTM XQR4000XL Radiation Hardened Field programmable Gate Arrays*, 1998.
- [17] M. Ohlsson, P. Dyreklev, K. Johansson, and P. Alfke, "Neutron Single Event Upsets in SRAM-based FPGAs", in (C. Cogn Ed.)*Workshop Record for the 1998 IEEE Radiation Effects Data Workshop*, pp. 177-180 1998, also available at http://www.xilinx.com/appnotes/FPGA_NSREC98.pdf.
- [18] P. Alfke, and R. Padovani, "Radiation Tolerance of High-Density FPGAs", *1998 Military and Aerospace Applications of Programmable Devices and Technologies (MAPLD) Conference*, Nasa Goddard Space Flight Center in Greenbelt, Maryland, September 14-17, 1998, available at <http://www.xilinx.com/appnotes/HiDensityFPGAs.pdf>.
- [19] D.P. Siewiorek and R.S. Swarz, *The Theory and Practice of Reliable System Design*, Digital Press, 1982.
- [20] C. Metra, M. Favalli, B. Ricco', "Embedded Two-Rail Checkers with On-Line Testing Ability", in *IEEE Proceedings of 14th IEEE VLSI Test Symposium*, Princeton (New Jersey), 28 April-1 May, 1996, pp.145-150.
- [21] L. Zhao, D.M.H. Walker, and F. Lombardi, "Bridging Fault Detection in FPGA Interconnects Using I_{DDQ} ", in *Proc. of 1998 ACM/SIGDA Sixth International Symposium on Field Programmable Gate Arrays*, Monterey, CA, February 22-24, 1998, pp. 95-104.

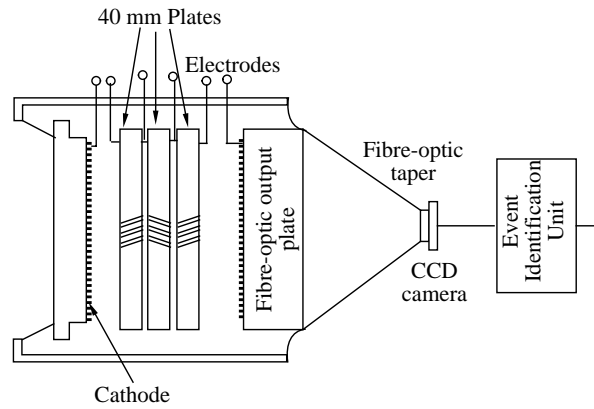


Figure 1. The ICCD detector scheme. The CCD camera is a 512x512 15_μm-pixel matrix and operates at 60 frame/sec in full-frame mode. Frames are read sequentially (one pixel at a time) from the CCD camera digital output at 20 MHz.

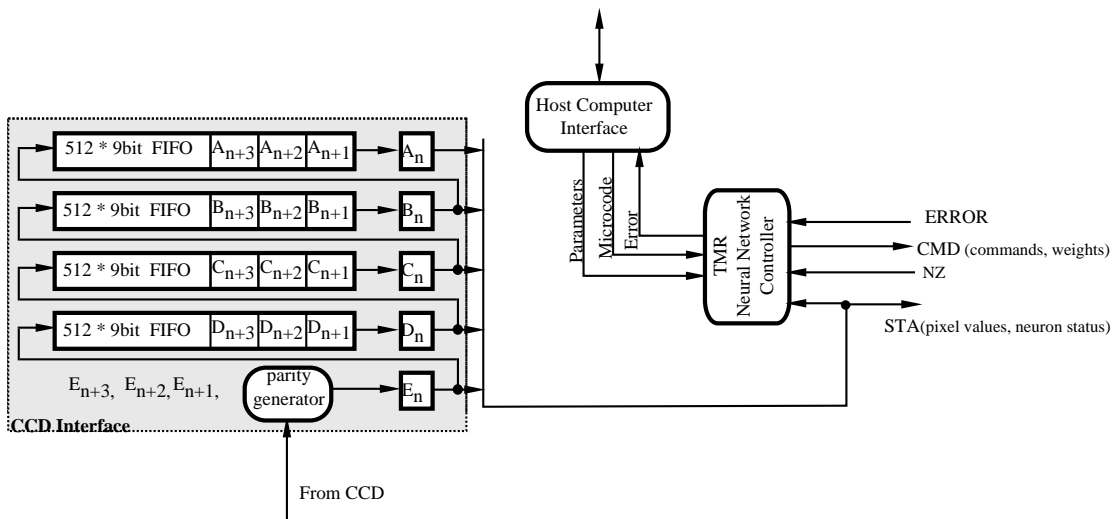


Figure 2. The architecture of the controller board.

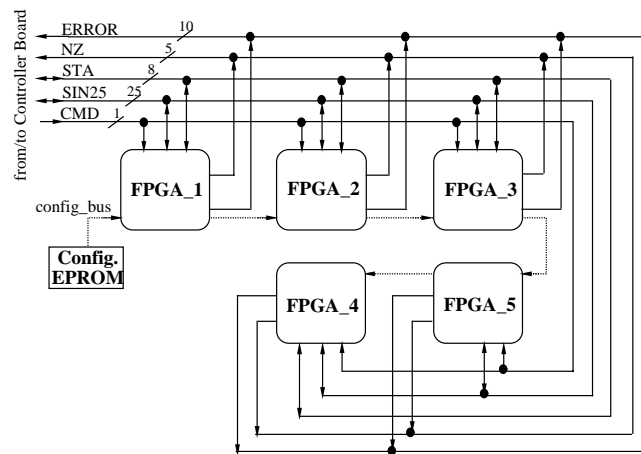


Figure 3. The neural-based identification unit.

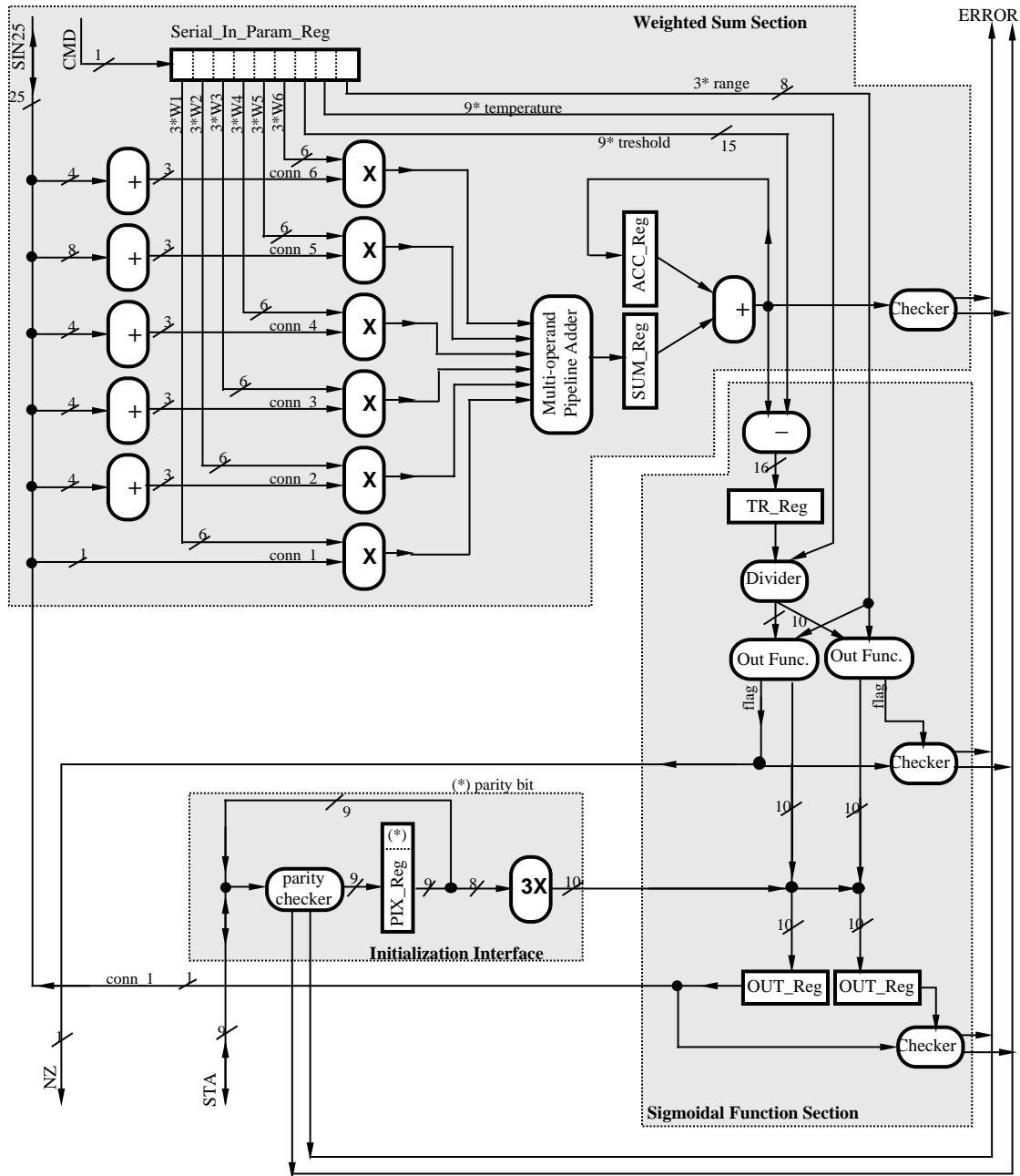


Figure 4. The scheme of the self-checking neuron.