A Highly Fault Tolerant PLA Architecture for Failure-Prone Nanometer CMOS and Novel Quantum Device Technologies

Alexandre Schmid and Yusuf Leblebici
Swiss Federal Institute of Technology EPFL
Microelectronic Systems Laboratory
CH - 1015 Lausanne Switzerland

Abstract

Nanometer-scale CMOS devices, as well as new devices based on quantum technologies are expected to gradually replace current CMOS devices within the next ten to fifteen years. However it is expected that these devices will be prone to failures of several types, until radically new fabrication processes yet to be developed stabilize, and error absorbing methodologies adapted to expected failures are developed and massively applied. This paper proposes a method and the underlying system architecture for improving the fault-tolerance, based on a feed-forward four layer structure which can accommodate deep submicron CMOS circuits and novel quantum structures. Simulation results show a significant improvement of yield with respect to widely applied triple-redundancy and majority voting techniques. A programmable logic array arrangement of the proposed architecture is demonstrated.

1. Introduction

Aggressively downsized CMOS circuits expected to be available in the next technology generations, as well as nanometer-scale quantum-electronic devices exploit very different device properties to achieve similar switching characteristics. While the fabrication technologies and the device characteristics are fundamentally different from each other, it is envisioned that these devices will coexist in future systems with specific inherent properties to be best exploited in different but complementary fields or applications.

Devices realized in CMOS are very mature, their parameters can be controlled and stabilized. The reliability of CMOS circuit has been studied over the thirty years of its industrial use, and solutions to increase the probability of correct system operation include all level of development abstraction, ranging all the way up from technology and device improvement through system-level failure safe architectures, and algorithms that allow identifying and recovering erroneous data. CMOS technology relies on the gate modulation of a conductive channel to vary the transconductance of the device.

Nanotechnologies are envisioned to establish as industrially compliant technologies within ten years. So far, several devices have been theoretically demonstrated to be functional, and constructed and measured, mainly as proof-of-concept isolated devices. The theoretical foundations of nanodevices lie in the physics of quantum devices, and were initiated by Nobel Prize Richard P. Feynman in his famous 1959 talk summarized as “There is plenty of room at the bottom.” One relevant nanoelectronic device candidate is the single electron device (SET), which consists of a conductive island of nanometric dimension isolated by two tunneling junctions [1]. The tunneling rate is probabilistic, and can be adapted by a gate voltage level.

The operation of these two devices is clearly very different. Despite of this fact, it is expected that both technologies will suffer from low reliability in future technology generations. Current CMOS devices already suffer from deep-submicron effects such as short-channel

effects and static leakage. The impact of these effects is expected to be very significant for sub-50nm technologies, where additional parasitic effects such as gate oxide leakage and process variation induced effects will contribute to lower the yield. On the other hand, SETs are known to be extremely sensitive to temperature, due to the dependence of tunneling rate to temperature. Construction of room-temperature operating device is beyond currently available fabrication technologies. Moreover, guaranteeing the long term temperature stability during various phases of operation has not yet been considered. SETs which can be operated in a single tunneling of electron mode are expected to be affected by transient and random movement of charges that are trapped in the underlying lattice, called background charge effect.

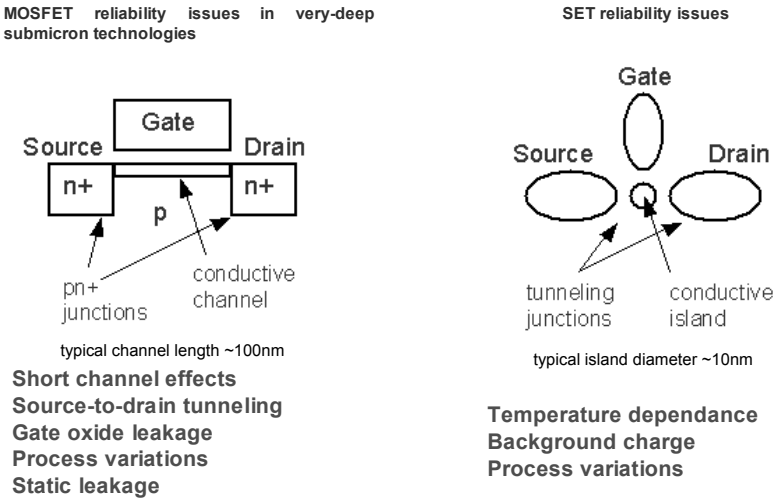


Figure 1. CMOS and SET technologies, showing some failure mechanisms.

Thus, technologies that will be available in the near future are expected to suffer from device disruption prone to cause catastrophic system behavior of digital systems which rely on perfect operation of every constituting elements to reconstruct correct results. Permanent and transient errors, together with localized and random failures are expected to influence proper device and circuit operation with predicted device failure rates of 10% to 30%. While reliability is considered and treated at the technology and device levels, it appears clearly that a circuit and systems-level approach will be needed in order to overcome such random errors. Constructing operators with the built-in ability to survive and absorb these errors is a key element in the development of failure safe systems, built with failure prone technologies.

In this paper, a system architecture exhibiting enhanced failure absorbtion is described in Section 2, and simulation results are described in Section 3. Considering the need of a dense and regular design as a way to increase system reliability, a programmable logic array was designed using the proposed architecture, as described in Section 4. Some considerations on yield assessment are taken in Section 5, and the issue of threshold adjustment is discussed in Section 6.

2. Layered arrangement of a Boolean circuit

The proposed fault-tolerant architecture consists of four layers in which the data is strictly processed in a feed-forward manner (Fig. 2.a). The first layer is denoted as the input layer, accepting conventional Boolean (binary) signal levels. The core operation is performed in the second layer, which consists of a number of identical, redundant units implementing the

desired logic function. It will be seen that the fault immunity increases with the number of redundant units, yet the operation is quite different from the classical majority-based redundancy. In contrast to classical n-tuple redundancy, the proposed architecture is expected to be significantly more immune to multiple device failures, in the form of stuck-on or stuck-off faults. The third layer receives the outputs of the redundant logic units in the second layer, creating a weighted average with re-scaling. Note that the output of the third layer becomes a multiple-valued logic level. Finally, the fourth layer is the decision layer where a binary output value is extracted using a simple threshold function. Fig. 2.b shows the typical weighted-averaging and re-scaling function that is performed by one of the third-layer blocks. It was already shown in the literature that this particular type of weighted-sum functions can be implemented quite easily with SET devices [2]. Similarly, proposals have been made to exploit the particular characteristics of SETs for the implementation of multiple-valued logic functions [3].

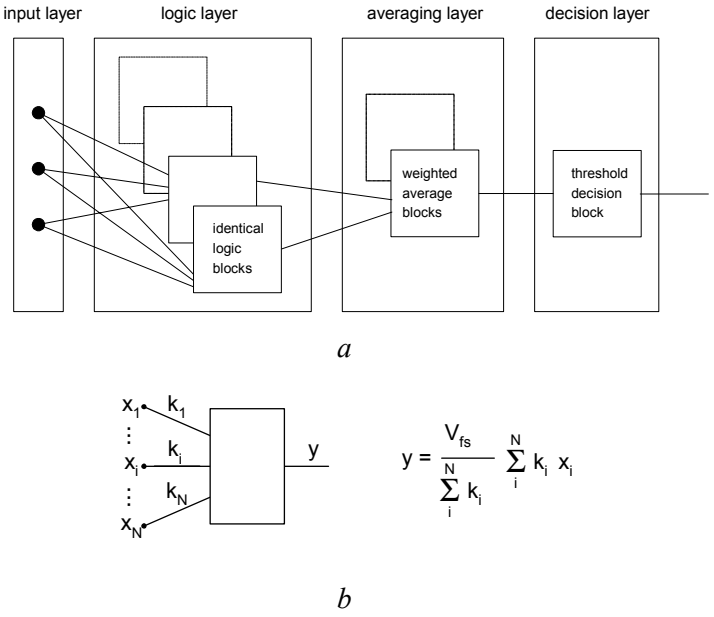


Figure 2. a) Four layer fault-tolerant operator, and b) weighted averaging and rescaling function used in the third layer.

Considering only the pull-down network that realizes a multi-variable Boolean function, the condition for correct restoration of the output function under multiple device failures can be expressed as:

$$V_{NM} \le \min_{\substack{\forall V_j \in \mathbb{L} \\ \forall V_{out\,i} \in \mathbb{P}}} \left\{ \frac{1}{2} \left(V_{H\min} - \frac{V_{fs}}{\sum_{i=1}^{N} k_i} \sum_{i=1}^{N} k_i V_{out\,i} \left(V_j \,|\, j = 1, .., m \right) \right) \right\}$$

where V_{NM} is the allowable output noise margin, $V_{H\min}$ is the lowest level for "logic 1" output, V_{fs} is the nominal full-scale range for the output voltage, k_i is the weight coefficient, $V_{out\,i}$ is the actual output voltage of each unit in the second layer under the assumption of device failures, V_j is the input voltage vector, and N is the number of redundant units in the second layer. Note that $\mathbb{L}$ represents the set of all m input combinations that are expected to produce a "logic 0" output in the undamaged gate, and $\mathbb{P}$ represents the set of all outputs for each unit in the second layer, with or without

device failures. It is assumed that the thresholding decision in the fourth layer is done at the half-way (50%) point between the lowest "logic 1" level and the highest "logic 0" level. A dual expression similar to the one given above can also be derived to define the conditions for the pull-up network.

3. Simulation results for simple gates

The first example consists of two identical logic blocks in the second layer, and one averaging block in the third layer. It is assumed that the NOR function blocks in the second layer are realized using the straightforward construction approach based on dual (series-parallel) switch networks, similar to classical CMOS logic gates. Each NOR block in the second layer receives two binary inputs, and produces one binary output. The outputs of the second layer are processed further in the averaging block to produce the multiple-valued output (Fig. 3).

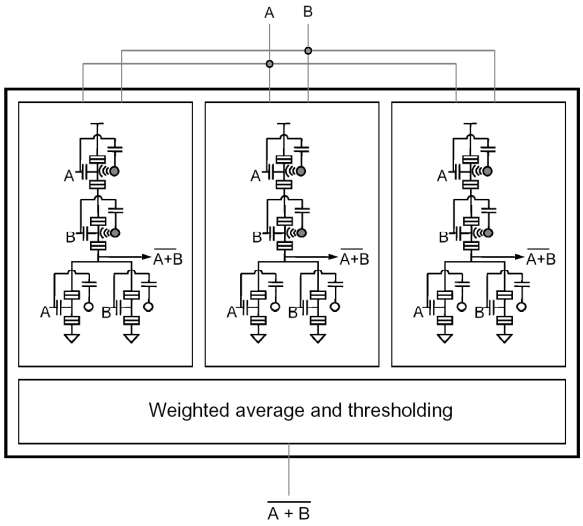
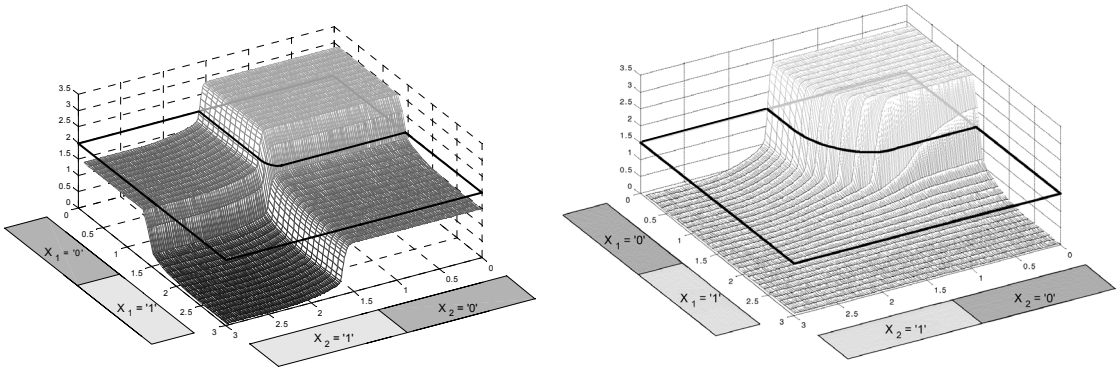


Figure 3. Structure of a two-input NOR cell built with three identical logic units, where each unit is implemented as a complementary SET circuit. (NOR circuit design after Uchida [4].)

As long as all devices operate correctly, three of the four possible input combinations will produce a logic-zero output in both of the second-layer logic blocks, and only one input combination, "00" will produce a logic-one output. Considering random device failures, it can be shown that the proposed architecture can successfully absorb all single-faults occurring anywhere in the second layer, as long as there are two or more identical logic units in the second layer. This is a property that can only be achieved using three or more redundant units in the conventional approach based on majority decisions. Furthermore, the new circuit architecture is capable of producing correct output behavior even when some devices in the third layer (averaging block) are faulty.

Fig.4.a shows the output transfer function surface of the circuit described above (two identical NOR blocks in the second layer, one averaging block in the third layer) where a total of four devices are assumed to be faulty. It can be seen that the correct output behavior can be extracted by setting the decision threshold level as shown. A fixed decision threshold level appears to be sufficient in most cases, while dynamically adjustable decision threshold levels may further increase the flexibility of the proposed approach.



a *b*

Figure 4. a) Output transfer function generated by the averaging layer of the two-input NOR circuit, with a total of four device failures in both of the second-layer logic blocks. The fourth-layer decision threshold is also indicated. b) Output transfer function showing correct operation (no device failures).

Simulation results in Fig. 5 show the graceful degradation of the probability of correct operation using the weighted averaging approach, where classical triple redundancy with majority voting results in sharply declining probabilities for large number of faulty devices. Moreover, the superiority of the proposed method over classical triple redundancy voting can be seen on Fig. 5, where the probability of correct operation under a large number of random errors affecting a system of three redundant units remains at a much higher level.

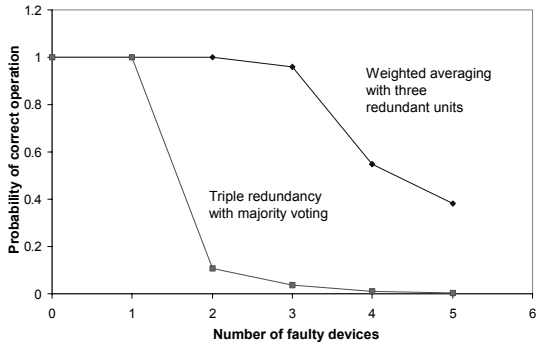


Figure 5: Probability of correct operation for the two-input NOR circuit with three redundant units in the 2nd layer, as a function of the device failure probability. The curve was obtained with exhaustive application of all possible device failure scenarios.

4. Robust system-level design

4.1 Hierarchical design

The proposed four-layer architecture can be applied at various levels in the abstraction hierarchy. Clearly, it is expected that device-level failure recovery will be crucial for new, aggressively downsized technologies. All examples shown in Section 3 demonstrate the application of the proposed principle at the level of logic gates. There is no theoretical limitation to the application of this method to higher levels of abstraction. In this case again,

error absorbtion has to rely on bitwise computation. Following this approach, the proposed method can be applied to several layers, hierarchically (Fig 6).

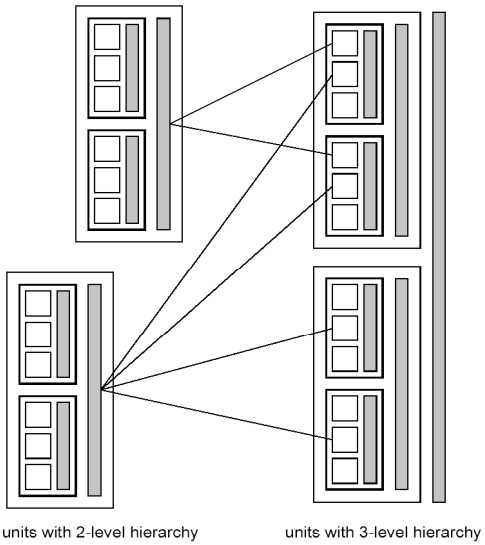
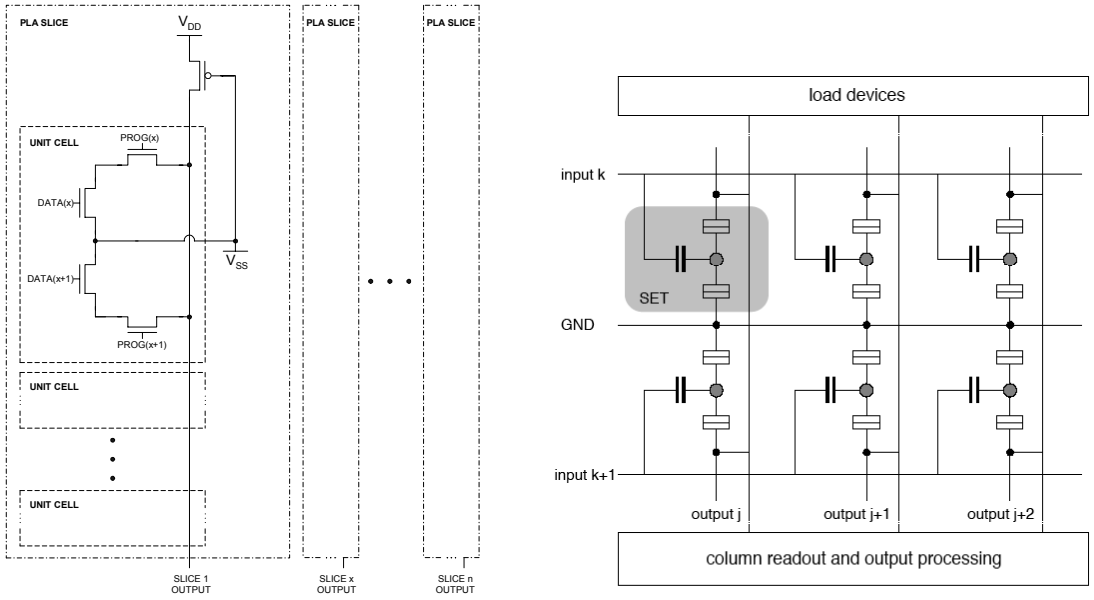


Figure 6. Hierarchical application of the proposed fault-tolerant architecture. Grey areas correspond to the averaging and tresholding layers.

4.2 Circuit-level programmable logic array design style

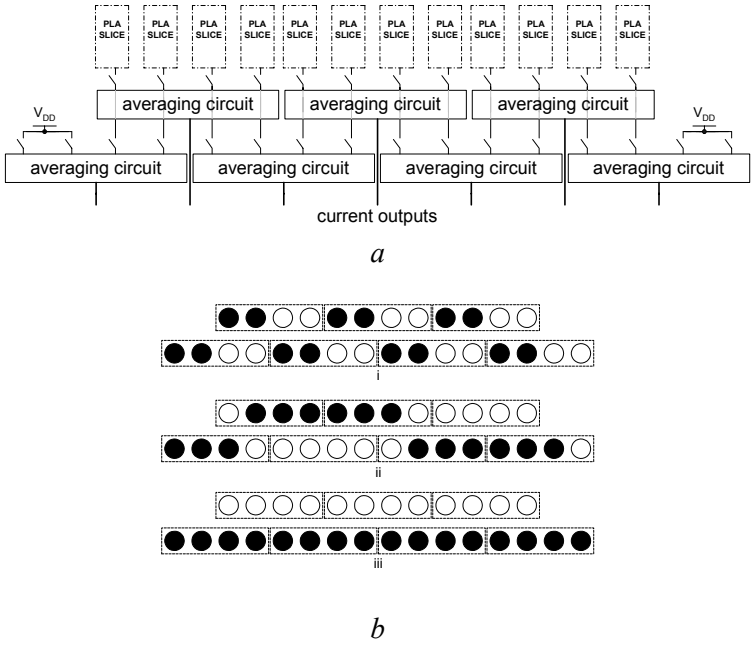
The programmable logic array design style offers a compact and dense realization of a layered Boolean operator circuit [5]-[6]. A regular programmable logic array (PLA) of unit building blocks has been adapted to provide fault-tolerance capability in the second layer using SETs or nanometer CMOS devices. The PLA is used for performing a programmable NOR Boolean operation of its inputs. Fig. 7 shows the structure of the array, made from one unit cell being replicated in the vertical direction to form the logic function as a slice. A number of slices are appended in the horizontal direction and share the same input variables to be connected to the DATA inputs.

In a standard PLA array, every slice outputs a different logic function. However, in order to achieve fault-tolerance, redundancy in the slices is demanded. Applying the concepts presented in Section 2, a number of slices performing an identical Boolean logic function are to be connected to an averaging unit. In order to provide soft programmability of the redundancy factor, each output of a slice is connected to two four-inputs averaging units, according to the scheme depicted in Fig. 8.a for a PLA consisting of twelve slices. The programmability scheme of the switches granting access to the averaging units allows redundancy factors of two, three or four for each logic function. It has been shown previously that the proposed four-layer architecture has the capability of absorbing errors which occur with a high-density pattern much more efficiently than majority voting schemes usually applied, even with a low redundancy factor, typically two or three. Fig. 8.b shows three programmation schemes, where a black dot represents an active switch connecting the PLA slice output to the averaging unit depicted as a dotted box gathering up to four connections. A white dot represents an open switch.



a *b*

Figure 7. Conceptual description of a PLA design based on a NOR network, where two design styles are shown, a) based on CMOS logic, and b) based on SET logic.



a

b

Figure 8. Connectivity programming of the third layer, showing a) the PLA building blocks (slices), and b) three examples of connectivity with various level of redundancy applied, where R=2 (i), R=3 (ii), and R=4 (iii).

5. Considerations on the yield assessment

The derivation of an analytical expression for the yield of a specific arrangement is strictly limited to a probability operation. Several rules have to be applied in order to specify the

correct terms to be applied. The specific design, and hardware arrangement play a significant role in the determination of the exact set of rules, to be translated into probabilities. The impact of failure in all possible combination of transistors in the specific circuit must be assessed. A probability of correct operation under a selected number of errors can be attached to each of these combinations.

The decision on the "correct" behavior of a gate under a given number of failures depends on the possibility for the subsequent thresholding operation to be performed within the limits of the fourth layer precision. Thus, a gate which would be considered as faulty from a digital logic perspective (using digital noise margin criteria), can still be considered as operating in an acceptable range, where separability into binary values remains possible. Typically, output range compression is supported by the proposed method, although the digital output levels are reduced.

The probability of correct operation under worse-case massive disruption of the circuit elements is straightforward to derive. The probability expression is related to the minimal number of circuit elements that must operate in order to create an acceptable transfer function surface, for which a fourth layer threshold function can be found within the limits of circuit precision. All possible combinations leading to the same system state must then be found to derive the final probability result.

However, in the case of spatially uncorellated circuit failure, the probability of correct operation must be extracted from the rules which dictate the conditions of correct operation for the specific circuit design, involving the careful study of all possible cases. For example, the disruption of a transistor may be tolerated, under the assumption that its counterpart belonging to a redundant unit of the same function operates without failure.

6. System-level approach to fourth layer threshold adaptability

Correct binary output of the proposed failure-absorbing architecture is obtained by thresholding the output of the third layer. The fourth layer consists of a hard-limiting thresholding module, which make a decision based on the transfer function surface. The actual value of the threshold is a key parameter for proper operation of the system. The correct value of the threshold can be extracted from analyzing the transfer function surface. This process is straightforward for a human operator working on a single gate.

Automatization of the threshold setting process is required in order to propose a method that can effectively be applied to real-life cases. We propose two possible approaches to solving this critical issue, both based on the theory of artificial neural network training.

One prerequisite is that a complex Boolean operator can be broken down into clusters consisting of simple functions where the dataflow direction is in a strict feedforward path, and where the input-output conditions are known. Under these conditions, it is possible to consider the simpler function derived in the preceding step as an artificial neural network which has to be trained in order to achieve a predetermined function. Here every neuron is represented as a Boolean gate having the threshold as the only adaptable parameter, and a hard-limiting function as the activation function. Learning can be applied to these units using an adapted version of the Backpropagation learning rule [7], or weight-perturbation learning [8] and the threshold level properly adapted to absorb any possible errors in the first three layers, under the condition that the fourth layer be spared from any fault. A chip-in-the-loop arrangement of the training system vs. trained system seems to be appropriate for this

approach, which requires systematic testing under the constraints of a predefined set of test patterns consisting of input-expected output pairs of data.

The training process described in the preceding paragraph requires a relatively long phase to be reserved to system configuration prior to any computation. Moreover, it is capable to correct permanent errors only. Dynamic adaptation of the threshold aiming at absorbing transient errors requires a control and correction process to be applied in real time to every gate. Weight-perturbation algorithms can be applied to update the weights on-the-fly. However, this requires to integrate the training sets and some control on-chip. Hence, the granularity of the hardware to be trained results from a trade-off with the extra hardware that must be incorporated, resulting in a solution half-way between the chip-in-the-loop and the actual on-chip learning training methods.

7. Conclusion

In this paper, design challenges for nanometer-scale devices and single-electron transistors are discussed, concentrating on the functional robustness and fault tolerance point-of-view. A robust architecture based on a four layer feed-forward circuit, capable of absorbing the effects of multiple simultaneous device failures and still offering a high probability of correct operation is presented. A PLA arrangement applying the proposed fault-tolerant concepts has been developed. The very regular and repetitive array arrangement of the implemented Boolean functions is expected to impact positively on the reliability of the system. The proposed approach is different from conventionally applied techniques based on multiple redundancy and majority voting, and offers improved immunity to permanent and transient faults occurring at the transistor level.

8. Acknowledgments

The authors acknowledge the support of the Swiss National Science Foundation under Grant No. 200021-101847/1.

9. References

[1] C. Wasshuber, Computational Single–Electronics, Springer Ver., Wien, 2001.

[2] C. Lageweg, S. Cotofana, S. Vassiliadis, “A Linear Threshold Gate Implementation in Single Electron Technology,” in Proc. IEEE Comp. Soc. Workshop on VLSI, 2001, pp. 93-98.

[3] H. Inokawa, A. Fujiwara, Y. Takahashi, “A Multiple-Valued Logic with Merged Single-Electron and MOS Transistors,” Electron Device Meeting IEDM, 2001.

[4] K. Uchida et al., “Programmable single-electron transistor logic for low-power intelligent Si LSI,” Digest of ISSCC 2002, pp. 206-207, 2002.

[5] A. DeHon, “Array-Based Architecture for FET-Based, Nanoscale Electronics,” Proc. IEEE Nanotechnology, Vol. 2, No. 1, March 2003, pp. 23-32.

[6] M. M. Ziegler, M. R. Stan, “CMOS/Nano Co-Design for Crossbar-Based Molecular Electronic Systems,” Proc. IEEE Nanotechnology, Vol. 2, No. 4, December 2003, pp. 217-230.

[7] D. E. Rumelhart, G. E. Hinton, R. J. Williams, “Learning Internal Representations by Error Propagation, Parallel Distributed Processing: Exploration in the Microstructures of Cognition,” Vol. 1, D. E. Rumelhart, J. L. McClelland, Edts., MIT Press, Cambridge, MA, 1986, pp. 318-362.

[8] G. Cauwenberghs, “Analog VLSI Stochastic Perturbative Learning Architectures,” Neuromorphic Systems Engineering: Neural Networks in Silicon, T. S. Lande, Edt., Kluwer Academic Publishers, Boston, MA, 1998, pp. 409-435.