Theory and Applications of Mathematics & Computer Science

(ISSN 2067-2764, EISSN 2247-6202)
http://www.uav.ro/applications/se/journal/index.php/tamcs

Theory and Applications of Mathematics & Computer Science 7 (1) (2017) 47 – 58

On Threshold Voltage Variation-Tolerant Designs

Valeriu Beiu[a,*], Mihai Tache[b]

[a]*Department of Mathematics and Computer Science, "Aurel Vlaicu" University of Arad, Romania*
[b]*University Politehnica of Bucharest, Bucharest, Romania*

Abstract

Scaling CMOS transistors has been used to achieve *smaller*, *faster*, and *cheaper* integrated circuits. However, with CMOS transistors moving deep towards the nanometer range, the effects *threshold voltage* (V_{TH}) variations (besides other variations and noises) play on their reliabilities and that of the gates they are forming are worrying. For mitigating against this trend, *sizing* can be used to improve on the reliability of the CMOS gates. Simultaneously, sizing can also reduce power or maintain speed while only marginally affecting area. For evaluating the advantages sizing still holds, inverters of different sizings are compared in this paper with reliability enhanced inverters using well-known redundancy schemes like triple modular redundancy and hammock networks. Simulation results show that, *at the same reliability*, *sizing can lead to designs outperforming those obtained by the other methods on any of the design parameters (i.e., area, power or delay)*. These are reinforcing previous reports showing that *space redundancy* applied at the device-level outperform gate-level solutions.

Keywords: CMOS, sizing, reliability, redundancy, area, delay, power.
2010 MSC: 34M10, 30D35.

1. Introduction

Over half a century the semiconductor industry has relied on CMOS scaling as the basis for its growth, implementing *smaller*, *faster*, and *cheaper* integrated circuits (ICs). However, with sizes approaching *10nm*, industry is facing several fundamental limitations. One of these is the randomness of the number and locations of doping atoms (Asenov, 1998), (Asenov *et al.*, 2003), which together with imprecisions in fabrication are leading to device-to-device fluctuations/variations in key parameters, including V_{TH}.

When adding intrinsic and extrinsic noises (on top of variations), reliability looks like one of the greatest threats to the design of future ICs (SIA, 2014). The expected higher *probabilities of failures* (PFs), due to higher sensitivity to noises and variations, could make future ICs

*Corresponding author
Email address: valeriu.beiu@uav.ro (Valeriu Beiu)

prohibitively unreliable. In this context, ITRS (SIA, 2014) predicted that CMOS scaling would become difficult when trying to go beyond *10nm* as more "*errors [will] arise from the difficulty of providing highly precise dimensional control needed to fabricate the devices and also from interference from the local environment.*" That is why VLSI designers should consider reliability as an extra design parameter, in addition to area, power, and delay.

The well-established approach for improving reliability is to add *redundancy* (von Neumann, 1956), (Moore & Shannon, 1956), (Winograd & Cowan, 1963), (Wakerly, 1976). Redundancy can be either in *space*, *time, information*, or a combination of some of these. *Space (hardware) redundancy* can be most easily understood in relation to voting and includes: modular redundancy (von Neumann, 1956), (Wakerly, 1976), (Abraham & Siewiorek, 1974), cascaded modular redundancy (Lee *et al.*, 2007), (Hamamatsu *et al.*, 2010), as well as multiplexing (e.g., von Neumann multiplexing (von Neumann, 1956), enhanced von Neumann multiplexing (Roy & Beiu, 2004), (Roy & Beiu, 2005), and parallel restitution (Sadek *et al.*, 2004)). Still, voters are not necessarily needed. In fact, besides multiplexing, others schemes which do without voting include: quadded logic (Tryon, 1960), (Jensen, 1963); interwoven logic (Pierce, 1964); radial logic (Klaschka, 1967), (Klaschka, 1969); *n*-safe-logic (Mine & Koga, 1967), (Das & Chuang, 1972); dotted logic (Freeman & Metze, 1972); as well as solutions at the device/transistor level. *Time redundancy* is trading space for time (e.g., alternating logic, re-computing with shifted operands or with swapped operands, etc.), while *information redundancy* is based on error detection and error correction codes.

The focus of this paper is on space redundancy. Space redundancy can be applied at the system-, module-, gate-, or device-level. Applying space redundancy at the device-level is much more efficient than applying it at higher levels (as explained in (Moore & Shannon, 1956); see also (Beiu & Ibrahim, 2011)), while the common expectation is that spatial redundancy should always degrade performances, i.e., increase *area, power*, and *delay*. In this paper we will show that redundancy applied at the device-level can improve redundancy without increasing *area*, and even while reducing *power* or *delay.*

Sizing has already been suggested as a way to enhance tolerance to variations (Sulieman *et al.*, 2010), (Ibrahim *et al.*, 2011), (Keller *et al.*, 2011), (Ibrahim & Beiu, 2011). In fact, sizing gives the VLSI designer options for optimizing the trade-offs between reliability and *area-power-delay*, while, in particular, it can enhance reliability and reduce power within the same area. For getting a better understanding of the advantages sizing can bring to reliability, the performances of differently sized inverters will be weighted against those obtained by using reliability improvement schemes including triple modular redundancy (TMR) and four-transistor hammock networks (H_{22}). The paper is organized as follows. The effect sizing plays on tolerating V_{TH} variations is discussed in Section 2. A brief review of space redundancy methods is presented in Section 3. Sizing is revisited in Section 4, followed by simulation results in Section 5 and concluding remarks in Section 6.

2. How Sizing Affects Variations

VLSI designers have normally adjusted the sizing of nMOS and pMOS transistors (i.e., width W and length L) in order to balance the driving currents when either the pMOS (I_{pMOS}) or the

nMOS (I_{nMOS}) stacks are switched ON. This requires the balancing of the ON resistances of the pMOS and nMOS stacks (R_{pMOS}, R_{nMOS}), which is achieved by adjusting (sizing) the transistors because pMOS conduction relies on holes which have slower mobility than electrons. Fig. 1 shows four different sizing options for a transistor. Although all of them have the same area $W \times L = 6a^2$, they have different ON resistances. In case of Fig. 1(a) there are 6 squares (a^2) connected in parallel, therefore $R_{ON} = R_{\square}/6$ (where $R_{\square}$ is the resistance of a square, e.g., $W = a$, $L = a$). In case of Fig. 1(d), the six squares are connected in series, hence $R_{ON} = 6R_{\square}$. Similarly, R_{ON} for Fig. 1(b) and 1(c) can be estimated as $3R_{\square}/2$ and $2R_{\square}/3$.

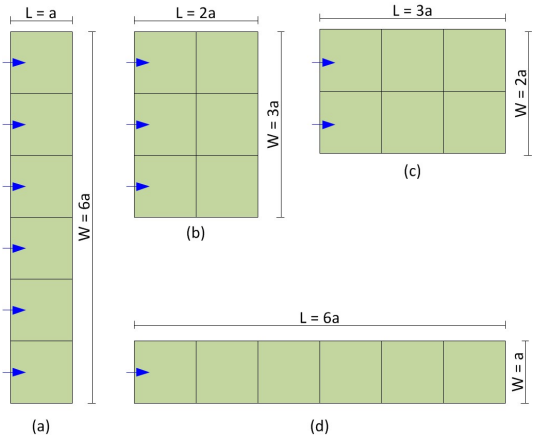


Figure 1. Four different sizing options having the same *area* ($A = 6a^2$).

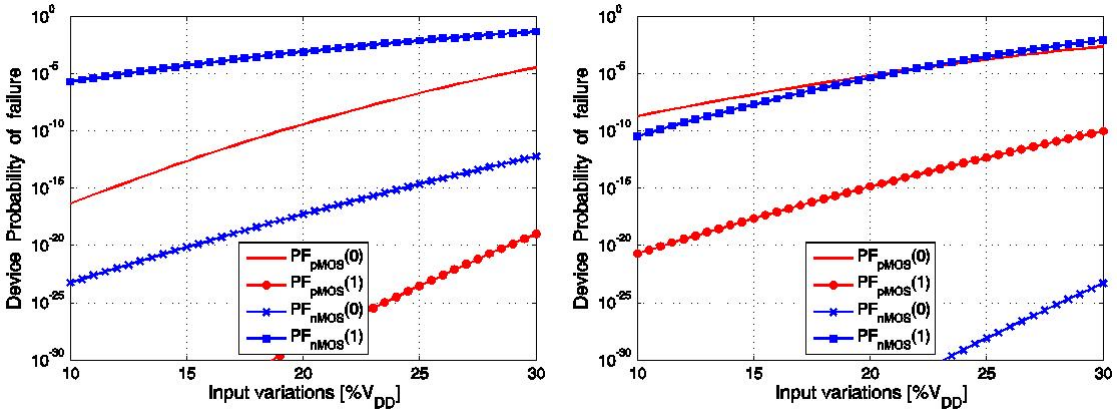
With CMOS scaling approaching *10nm*, it becomes difficult to reproduce V_{TH} over the large number of transistors in a chip. This is due to the random fluctuations of both the number of dopants and of their physical locations. V_{TH} variations can be approximated (see (Asenov *et al.*, 2003)) by a normal distribution with standard deviation:

$$\sigma_{V_{TH}} \simeq 3.19 \times 10^{-8} t_{ox} N_A^{0.4} (L_{eff} \times W_{eff})^{-0.5} [V] \tag{2.1}$$

where t_{ox} is the oxide thickness, N_A is the channel doping, W_{eff} is the effective channel width, and L_{eff} is the effective channel length. In the following we will use normalized dimensions for L and W.

Eq.(2.1) shows that increasing the transistors area (by increasing L and/or W) will always reduce V_{TH} variations. While the four sizing options in Fig. 1 are expected to exhibit similar probabilities of switching (meaning that the transistor fails to open/close, see (Beiu & Ibrahim, 2011), (Ibrahim & Beiu, 2011), (Ibrahim *et al.*, 2012)) as they have the same area, they will lead to very different performances, as their R_{ON} is between $R_{\square}/6$ and $6R_{\square}$.

For classical sizing VLSI designers set $L_{nMOS} = L_{pMOS} = min$ and $W_{nMOS} = 2 \times L_{nMOS}$ (i.e., $R_{nMOS} = R_{\square}/2$). To balance I_{pMOS} and I_{nMOS}, W_{pMOS} is then increased, such as R_{pMOS} matches R_{nMOS}. This also increases the area of the pMOS ($A_{pMOS} = L_{pMOS} \times W_{pMOS}$), improving their reliability. Fig. 2(a) shows that a pMOS transistor is more reliable than an nMOS. As classical sizing increases the area of the pMOS transistors it makes them even more reliable than nMOS transistors.



(a) (b)

Figure 2. PF_{TRS} with respect to variations: (a) classical sized ($L_{nMOS} = L_{pMOS} = 1, W_{nMOS} = 2$, and $W_{pMOS} = 4$); (b) reverse sized ($W_{nMOS} = W_{pMOS} = 1, L_{nMOS} = 4$, and $L_{pMOS} = 2$).

For enhancing PF_{GATE}, it is essential to improve the reliability of the nMOS stack, ideally matching the reliability of the pMOS stack (similar to matching R_{pMOS} to R_{nMOS}). For doing this A_{nMOS} should be enlarged (see eq. (2.1)). Classical sizing is using $W/L > 1$ and $L = min$, so it follows that W_{nMOS} has to be increased. Subsequently, this requires increasing W_{pMOS} (to compensate for the slower mobility of the holes). Hence, relying on classical sizing W_{nMOS} has to be increased, which leads to enlarging all transistors and degrading the gates *area, delay*, and *power* consumption.

3. Space Redundancy

Classical space redundancy schemes start from an unreliable system and use divide-and-conquer in a top-down fashion as follows. The unreliable system is divided into several sub-systems which are interconnected by a *network*. Each sub-system is further divided into several sub-sub-systems, which are also interconnected by a network. This continues down to the elementary transistors, and the level where redundancy will be applied has to be decided. Four levels are well-established: *system, module, gate*, and *device*. Redundancy can be applied simultaneously at more than one level even using different schemes at different levels. This implies that the optimization space is very large. Fundamentally, using space redundancy at any level translates into replicating all of the sub-systems at that level by a *redundancy factor R*. This *R*-times larger redundant system needs to be connected by a modified network. Most space redundancy methods are done at this point, while some space redundancy methods require additional blocks (e.g., voters) for connecting the original sub-systems. In the following we shall briefly review space redundancy methods by classifying them with respect to the need for voters, while also suggesting how complex is the connectivity pattern (modified network) they use.

3.1. Higher Level Methods

The most well-known high level redundancy methods are *triple modular redundancy* (TMR) and *n-modular redundancy* (NMR). TMR was proposed by von Neumann [4]. It divides a system into modules (sub-systems) and triplicates each module (Fig. 3). A voter is used to combine the outputs of the $R = 3$ modules operating in parallel (Lyons & Vanderkulk, 1962), (Gurzi, 1965), (Longden *et al.*, 1966), (Wakerly, 1975), (Stroud, 1994), Morgan *et al.* (2007). TMR is able to mask failures that affect one module by taking the majority of three modules. The interconnectivity pattern is simple (Fig. 3(a)), while it might get slightly more complex if more voters are used in parallel (Fig. 3(b)).

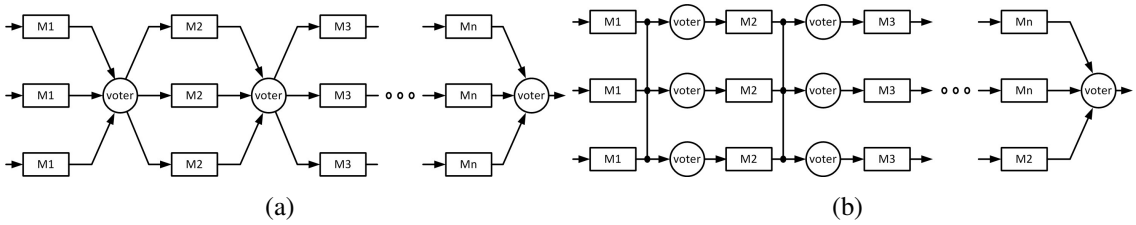


Figure 3. Triple modular redundancy: (a) one voter per stage; (b) three voters per stage.

NMR is an extension of TMR to any odd number n. It requires replicating all the modules n times ($R = n$), and also using larger voters (with n inputs), but it could tolerate $n/2$ module failures (Ness *et al.*, 2007). The connectivity pattern gets more complex and the length of the wires increases as n is increased, and if more voters are being used in parallel. The early analyses have assumed that voters are very reliable. Later it was realized that even assuming that the reliability of a voter is independent of the number of inputs n is unrealistic, and could lead to wrong conclusions. This has motivated research into space redundancy methods which could do without voters.

A gate-level redundancy method without voting was also introduced by von Neumann in [4], and is known as *multiplexing*. Other gate-level space redundancy methods without voting are *quadded* (Tryon, 1960), (Jensen, 1963), *interwoven* (Pierce, 1964), *radial* (Klaschka, 1967), (Klaschka, 1969) and *n-fail-safe* (Mine & Koga, 1967), (Das & Chuang, 1972) logic. All of these exhibit simpler connectivity patterns then multiplexing, as being more regular and local, i.e., having shorter wires. Another gate-level method which does not require voting is *dotted logic* (Freeman & Metze, 1972), which is bridging the gap between gate- and device-level methods. The reason is that dotted logic took advantage of implementations which use wired AND and OR functions. This is not entirely gate-level anymore, but it is not yet device-level either.

3.2. Device-Level Methods

Device-level methods have also been introduced in a seminal article (Moore & Shannon, 1956) (relays in the original paper). The main conclusions of that work have been that:

- redundant relay (device-level) structures are able to outperform redundant gate-level schemes at significantly (orders of magnitude) smaller R; and that
- the modified networks (there are different ways to connect the redundant relays) have a strong influence on reliability.

All the subsequent publications inspired by the original study of Moore and Shannon (Moore & Shannon, 1956) have detailed particular applications of those ideas. They rely on series-and-parallel networks of (a few) devices. The most widespread network used is a series-parallel network of 4 devices/relays/transistors which is the simplest hammock network (Moore & Shannon, 1956). This has been named a *quad configuration* by many of the later papers Suran (1964), Bolchini *et al.* (1996), Abid & El-Razouk (2006), Anghel & Nicolaidis (2007), El-Maleh *et al.* (2008). Here we shall use hammock network (hence the H abbreviation) as we do not want to create any confusion with respect to gate-level *quadded logic* (Tryon, 1960), (Jensen, 1963). A few papers have looked at simpler hammock networks (Djupdal & Haddow, 2007), or at hammock networks having more than four transistors (Anghel & Nicolaidis, 2007), (Aunet *et al.*, 2005). It looks like this trend will be taking up due to developments on carbon nano tubes (Zarkesh-Ha & Shahi, 2010), (Zarkesh-Ha & Shahi, 2011).

4. Transistor Sizing Revisited

While sizing has been used for a very long time to balance driving currents, its use for enhancing reliability has only recently started to be explored for $W/L > 1$ (classical sizing) (Keller *et al.*, 2011). Still, a *reverse sizing* ($W/L < 1$) has been proposed in (Sulieman *et al.*, 2010) for overcoming the problems mentioned in Section 2. This sizing method keeps all W minimum ($W_{nMOS} = W_{pMOS} = min$), and increases L. Normally, this is not used for digital circuits, but has been used in analog circuits as "*better matching can be obtained without consuming additional area, simply by changing the W/L aspect ratio*" (Drennan & McAndrew, 2003). To make A_{nMOS} larger than A_{pMOS}, L_{pMOS} should be kept small ($L_{pMOS} = 2W_{pMOS}$), while L_{nMOS} should be increased. While occupying the same area, the reverse sizing method enhances the gates reliability but diminishes its performances.

This is because increasing L increases R_{ON} and hence the delay, but power is reduced as I_{ON} is reduced. Fig. 2(b) shows PF_{TRS} for reverse sizing. Increasing the area of the nMOS transistors improves their reliability and (more importantly) allows matching the reliability of the pMOS transistors (see $PF_{nMOS}(1)$ and $PF_{pMOS}(0)$ in Fig. 2(b)).

Aiming to simultaneously optimize *reliability* and *power-delay-area*, an exhaustive sizing search was suggested in (Ibrahim *et al.*, 2011). Instead of using L_{min} (classical sizing) or W_{min} (reverse sizing), different sizings are obtained by analyzing all the possible A_{nMOS} and A_{pMOS} combinations, lower than a maximum area A_{max}, and achieving a PF_{GATE} lower than a target PF_{target}. This method is exhaustive as iterating through all the possible nMOS area combinations from $W_{nMOS} \times L_{nMOS} = 1 \times A_{max}$ to $A_{max} \times 1$. For each nMOS sizing combination, the algorithm tries to find all the corresponding pMOS sizing combinations ($W_{pMOS} \times L_{pMOS}$) such that R_{pMOS} matches R_{nMOS} and $A_{pMOS} \leq A_{max}$. If a pMOS sizing combination is found, Gate Reliability EDA (GREDA) (Ibrahim *et al.*, 2012) is used to quickly and accurately estimate PF_{GATE}.

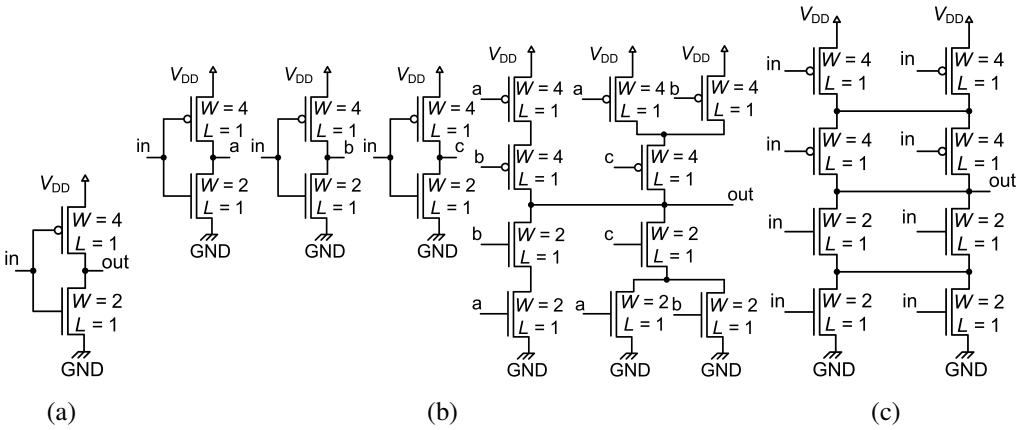


Figure 4. (a) Classical INV; (b) INV-TMR with a MIN-3 as voter; (c) H_{22}-INV.

If $PF_{GATE} \leq PF_{target}$, the method stores the identified nMOS and pMOS sizing combination in a list of candidate combinations. Finally, the method checks the list of candidate combinations. If the list is empty it means that PF_{GATE} cannot achieve PF_{target} with transistors of up to A_{max}. Otherwise, the design process is continued by using Spice to estimate the *delay, power*, and *power-delay-product (PDP)* for each candidate combination. The best combination that optimizes *delay, power* or *PDP* can then be selected. In all cases the *reliability* and the *area* constraints are always going to be satisfied.

5. Simulation Result

We have used 16*nm* PTM v2.1 incorporating high-k/*metal* gates and stress effects (Zhao & Cao, 2007), (PTM, 2011), as this is strongly affected by variations, and simulated at $V_{DD} = 700mV$ (nominal voltage) and $T = 27°C$. The TMR-INV circuit has three INVs followed by a mirrored MIN-3 gate as voter. This MIN-3 implementation was preferred as it is considered the most reliable one (Sulieman, 2009). All the transistors for TMR-INV (Fig. 4(b)) and H_{22}-INV (Fig. 4(c)) were sized using classical sizing, and the mobility of the electrons was assumed to be twice the mobility of the holes.

5.1. Reliability Results

In the first set of simulations GREDA was used to calculate the reliability of INVs with transistors of different sizings as well as TMR-INV (Fig. 4(b)) and H_{22}-INV (Fig. 4(c)). For all these simulations the input variations were assumed to be 15%, i.e., logic "1" $= 0.85V_{DD} = 595mV$ and logic "0" $= 0.15V_{DD} = 105mV$.

In the case of a classical INV, the simulation results show $PF_{INV}(0) = 7.25E-21$ and $PF_{INV}(1) = 5.33E-05$. The large difference between these values is due to $PF_{INV}(1)$ being dominated by PF_{nMOS}. For a reverse sized INV the simulation results show that increasing the area of the nMOS by increasing L_{nMOS} reduces $PF_{INV}(1)$ to $1.58E-07$ (i.e., 2 orders of magnitude better than $PF_{INV}(1)$ for classical sizing).

For TMR-INV the simulations show $PF_{TMR-INV}(1) = 8.51E-09$ (4 orders of magnitude better than classical) and $PF_{TMR-INV}(0) = 5.67E-09$, which although 12 orders of magnitude worse then classical, is balanced with respect to $PF_{TMR-INV}(1)$. This is due to the fact that the output of the INV gate ($PF_{INV}(0) = 7.25E-21$) is a logic "1" input for the MIN-3 gate, being significantly degraded (to $5.67E-09$) as affected by $PF_{MIN-3}(1)$, which is determined by PF_{nMOS}.

For H_{22}-INV we have seen $PF_{H_{22}-INV}$ being improved significantly for both logic "0" and logic "1": from $PF_{INV}(0) = 7.25E-21$ and $PF_{INV}(1) = 5.33E-05$ (classical INV) to $PF_{H_{22}-INV}(0) = 2.10E-40$ and $PF_{H_{22}-INV}(1) = 5.67E-09$ respectively.

For a fair comparison of the performances of sizing versus the other space redundancy methods considered, the PF_{target} was set to $1.0E-09$ (range achieved by TMR-INV and H_{22}-INV). The maximum transistor area was limited to $A_{max} = 10a^2$. Table 1 shows the seven different sizing combinations (with W/L aspect ratios above and below 1) satisfying both of these requirements and also matching R_{nMOS} to R_{pMOS}. All of them achieve reliabilities of the order $1E-10$.

5.2. Performance Results

The second set of simulations has used Spice to estimate the performances of the different solutions. These are reported in Table 1, starting with the classically sized INV having an average *delay* of 5.54ps and an average *power* consumption of 0.24μW.

Table 1. Reliability enhanced INV circuits as well as differently sized INVs.

	A_{nMOS} ($W\times L$)	A_{pMOS} ($W\times L$)	Area ($\sum A_{trs}$)	Worst PF_{INV}	Delay [ps]	Power [µW]	PDP [aJ]
Classical	2×1	4×1	**6**	5.33E−05	**5.54**	0.24	**1.34**
Reversed	1×4	1×2	**6**	1.58E−07	30.54	**0.06**	**1.70**
TMR	2×1	4×1	48	8.51E−09	20.01	3.55	71.03
H_{22}	2×1	4×1	24	5.67E−09	35.71	0.79	28.34
This paper	1×5	1×3	**8**	4.47E−10	55.12	**0.06**	3.49
	1×6	1×3	9	1.89E−10	62.65	0.07	4.20
	3×2	3×1	9	1.89E−10	13.34	0.25	3.39
	1×7	1×3	10	1.89E−10	70.71	0.07	5.01
	5×1	9×1	14	4.47E−10	**5.45**	0.64	3.49
	5×1	10×1	15	4.47E−10	5.62	0.69	3.89
	1×5	2×5	15	4.47E−10	153.04	0.15	22.57

Table 1 clearly shows that adding more gates (TMR-INV) or adding more transistors (H_{22}-INV), while improving the reliability over the classical INV by 4 orders of magnitude (from $1E-5$ to $1E-9$), significantly degrades both power and delay: TMR-INV increases the average delay by 3.6×, while the average power and PDP are increased by 14.8× and 53× respectively; H_{22}-INV is about 6.5× slower while consuming about 3.3× more power and having a 21× higher PDP.

The reverse sized INV improves redundancy by 2 orders of magnitude while also reducing power by 4×, but degrades delay and PDP by 5.5× and 1.3× respectively. Among other possible sizings, $[3\times 2, 3\times 1]$ improves PF_{INV} by more than 5 orders of magnitude (over the classical $[2\times 1, 4\times 1]$ sizing) at about the same power, while increasing delay and PDP by only 2.4×. For high-performance applications, one should select $[5\times 1, 9\times 1]$ which improves reliability by 5 orders of magnitude while being as fast as a classical INV (in fact it is a shy 2% faster), and

consumes about 2.7× more power. Alternatively, $[1 \times 5, 1 \times 3]$ could be selected for low-power applications, with reliability being improved by 5 orders of magnitude, and power being reduced 4×, while delay is increased $10 \times$.

6. Conclusions

This paper has compared the performances of different sized inverters with classical and reverse sized inverters, as well as with two redundancy methods at the gate-level (TMR) and device-level (H_{22}) (Mukherjee & Dhar, 2015) (Sheikh *et al.*, 2016), (Robinett *et al.*, 2007), (El-Maleh *et al.*, 2009). The main conclusions are:

- Sizing can outperform both TMR and H_{22} methods with respect to reliability.
- Improving the reliability of a CMOS gate can be achieved without increasing *area.*
- Improving the reliability of a CMOS gate should not necessarily lead to penalties in *power* or *delay*, or even on the contrary, i.e., there are reliability enhanced solutions which can achieve either lower *power* or shorter *delays* but not both.

Sizing can be used to improve tolerance to variations, and it is possible to design CMOS gates trading reliability versus *area-power-delay*. The disadvantages are represented by very large libraries of gates and a much more complex design.

Future work will analyze re-sized solutions for other CMOS gates (e.g., NAND, NOR, XOR, etc.), of different *fan-ins* (see (Gemmeke & Ashouei, 2012), (Gemmeke *et al.*, 2013)). These should be compared not only with classical sized gates, TMR, and H_{22}, but also with quadded, interwoven, radial, *n*-safe, and dotted logic solutions, and evaluated jointly with advanced CMOS (Berge & Aunet, 2009) (Liu & Moroz, 2007), (Maly, 2007), (Geppert, 2002), and even beyond-CMOS technologies (Courtland, 2016), (Desai *et al.*, 2016).

Acknowledgements. This work was supported in part by Intel (*ULP-NBA = Ultra Low-Power Application-specific Non-Boolean Architectures, 2011-05-24G*) and in part by the European Union through the European Regional Development Fund under the Competitiveness Operational Program (*BioCell-NanoART = Novel Bio-inspired Cellular Nano-architectures*, POC-A1-A1.1.4-E nr. 30/2016).

References

Abid, Z. and H. El-Razouk (2006). Defect tolerant voter design based on transistor redundancy. *J. Low Power Electr.* **2**, 456–463.

Abraham, J. A. and D. P. Siewiorek (1974). An algorithm for the accurate reliability evaluation of triple modular redundancy networks. *IEEE Trans. Comp.* **23**, 682–692.

Anghel, L. and M. Nicolaidis (2007). *Defect Tolerant Logic Gates for Unreliable Future Technologies*. Sandoval, F., Prieto, A., Cabestany, J., Graa, M. (eds.) Computational and Ambient Intelligence, LNCS. Springer, Heidelberg.

Asenov, A. (1998). Random dopant induced threshold voltage lowering and fluctuations in sub-0.1μm MOSFETs: A 3-D "atomistic" simulation study. *IEEE Trans. Electr. Dev.* **45**, 2505–2513.

Asenov, A., A. R. Brown, J. H. Davies, S. Kaya and G. Slavcheva (2003). Simulation of intrinsic parameter fluctuations in decananometer and nanometer-scale MOSFETs. *IEEE Trans. Electr. Dev.* **50**, 1837–1852.

Aunet, S., Y. Berg and V. Beiu (2005). *Ultra Low Power Redundant Logic Based on Majority-3 Gates*. da Luz Reis, R.A., Osseiran, A., Pfleiderer, H.-J. (eds.) From Systems to Silicon. Springer, Heidelberg.

Beiu, V. and W. Ibrahim (2011). Devices and input vectors are shaping von Neumann multiplexing. *IEEE Trans. Nanotech.* **10**, 606–616.

Berge, H. K. O and S. Aunet (2009). Benefits of decomposing wide CMOS transistors into minimum size gates. In: *Proc. NORCHIP, Trondheim, Norway,* art. 5397795.

Bolchini, C., G. Buonanno, D. Sciuto and R. Stefanelli (1996). Static redundancy techniques for CMOS gates. In: *Proc. Intl. Symp. Citc. and Syst. (ISCAS)*. pp. 576–579.

Courtland, R. (2016). The next high-performance transistor. *IEEE Spectrum* **53**, 11–12.

Das, S. and Y. H. Chuang (1972). Fault restoration using N-fail-safe logic. *Proc. IEEE* **60**, 334–335.

Desai, S. B., S. R. Madhvapathy, A. B. Sachid, J. P. Pablo Llinas, Q. Wang, G. H. Ahn, G. Pitner, M. J. Kim, J. Bokor, C. Hu, H. S. P. Wong and A. Javey (2016). MoS_2 transistors with 1-nanometer gate length. *Science* **354**, 99–102.

Djupdal, A. and P. C. Haddow (2007). Defect tolerant ganged CMOS minority gate. In: *Proc. NORCHIP,* art. 4481060.

Drennan, P. G. and C. C. McAndrew (2003). Understanding MOSFET mismatchfor analog design. *IEEE J. Solid-State Circ.* **38**, 450–456.

El-Maleh, A. H., B. M. Al-Hashimi, A. Melouki and F. Khan (2009). Defect-tolerant n^2-transistor structure for reliable nanoelectronic design. *IET Comput. and Digital Tech.* **3**, 570–580.

El-Maleh, H., B. M. Al-Hashimi and A. Melouki (2008). Transistor-level based defect tolerance for reliable nanoelectronics. In: *Proc. Intl. Conf. Comp. Syst. and Appls. (AICCSA)*. pp. 53–59.

Freeman, H. A. and G. Metze (1972). Fault-tolerant computers using "dotted logic" redundancy techniques. *IEEE Trans. Comp.* **C-21**, 867–871.

Gemmeke, T. and M. Ashouei (2012). Variability aware cell library optimization for reliable sub-threshold operation. In: *Proc. European Solid-State Circ. Conf. (ESSCIRC), Bordeaux, France*. pp. 42–45.

Gemmeke, T., M. Ashouei, B. Liu, M. Meixner, T. G. Noll and H. de Groot (2013). Cell libraries for robust low-voltage operation in nanometer technologies. *Solid-State Electr.* **84**, 132–141.

Geppert, L. (2002). The amazing vanishing transistor act. *IEEE Spectrum* **39**, 28–33.

Gurzi, K. J. (1965). Estimates for best placement of voters in a triplicated logic network. *IEEE Trans. Electr. Comp.* **EC-14**, 711–717.

Hamamatsu, M., T. Tsuchiya and T. Kikuno (2010). On the reliability of cascaded TMR systems. In: *Proc. Pacific Rim Intl. Symp. Dependable Comp. (PRDC)*. pp. 184–190.

Ibrahim, W. and V. Beiu (2011). Using Bayesian networks to accurately calculate the reliability of complementary metal oxide semiconductor gates. *IEEE Trans. Reliab.* **60**, 538–549.

Ibrahim, W., V. Beiu and A. Beg (2012). GREDA: a fast and more accurate CMOS gates reliability EDA tool. *IEEE Trans. CAD* **31**, 509–521.

Ibrahim, W., V. Beiu and H. Amer (2011). Reliability optimized CMOS gates. In: *Proc. IEEE Intl. Conf. Nanotech. (IEEE-NANO)*. pp. 730–734.

Jensen, P. A. (1963). Quadded NOR logic. *IEEE Trans. Reliab.* **12**, 22–31.

Keller, S., S. S. Bhargav, C. Moore and A. J. Martin (2011). Reliable minimum energy CMOS circuit design. In: *European Workshop CMOS Variability (VARI). Grenoble, France*.

Klaschka, T. F. (1967). Two contributions to redundancy theory. In: *Proc. Annual Symp. Switching and Autom. Th. (SWAT)*. pp. 175–183.

Klaschka, T. F. (1969). *Reliability Improvement by Redundancy in Electronics Systems. Part II: An Efficient New Redundancy SchemeRadial Logic.Tech. Rep. 69045*. Royal Aircraft Establishment, Farnborough, UK.

Lee, S., J. Jung and I. Lee (2007). Voting structures for cascaded triple modular redundant modules. *IEICE Electr. Exp.* **4**, 657–664.

Liu, T. J. King and V. Moroz (2007). Segmented channel MOS transistor. *US Patent* 7,247,887.

Longden, M., L. J. Page and R. A. Scantlebury (1966). An assessment of the value of triplicated redundancy in digital systems. *Microelectr. and Reliab.* **5**, 39–55.

Lyons, R. E. and W. Vanderkulk (1962). The use of triple-modular redundancy to improve computer reliability. *IBM J. R and D* **6**, 200–209.

Maly, W. (2007). Integrated circuit, device, system, and method of fabrication. *WO Patent* 133775.

Mine, H. and Y. Koga (1967). Basic properties and a construction method for fail-safe logical systems. *IEEE Trans. Electr. Comp.* **EC-16**, 282–289.

Moore, F. and C. E. Shannon (1956). Reliable circuits using less reliable relays. *J. Frankl. Inst.* **262**, 191–208 and 281–297.

Morgan, K. S., D. L. McMurtrey, B. H. Pratt and M. J. Wirthlin (2007). A comparison of TMR with alternative fault-tolerant design techniques for FPGAs. *IEEE Trans. Nuclear Sci.* **54**, 2065–2072.

Mukherjee, A. and A. S. Dhar (2015). Fault tolerant architecture design using quad-gate-transistor redundancy. *IET Circ. Dev. and Syst.* **9**, 152–160.

Ness, D. C., C. J. Hescott and D. J. Lilja (2007). Modeling failure reduction for combinational logic using gate level NMR. In: *Proc. Annual Reliab. and Maintain. Symp. (RAMS)*. pp. 208–213.

Pierce, W. H. (1964). Interwoven redundant logic. *J. Frankl. Inst.* **277**, 55–85.

PTM, Predictive Technology Model (2011). http://ptm.asu.edu/.

Robinett, W., P. J. Kuekes and R. S. Williams (2007). Defect tolerance based on coding and series replication in transistor-logic demultiplexer circuits. *IEEE Trans. Circ. and Syst. I* **54**, 2410–2421.

Roy, S. and V. Beiu (2004). Multiplexing schemes for cost-effective fault-tolerance. In: *Proc. IEEE Intl. Conf. Nanotech. (IEEE-NANO)*. pp. 589–592.

Roy, S. and V. Beiu (2005). Majority multiplexing-economical redundant fault-tolerant designs for nanoarchitectures. *IEEE Trans. Nanotech.* **4**, 441–451.

Sadek, S., K. Nikolić and M. Forshaw (2004). Parallel information and computation with restitution for noise-tolerant nanoscale logic networks. *Nanotech* **15**, 192–210.

Sheikh, A. T., A. H. El-Maleh, M. E. S. Elrabaa and S. M. Sait (2016). A fault tolerance technique for combinational circuits based on selective-transistor redundancy. *IEEE Trans. VLSI Syst.* in press.

SIA, Intl. Tech. Roadmap Semicon (2014). http://public.itrs2.net/.

Stroud, C. E. (1994). Reliability of majority voting based VLSI fault-tolerant circuits. *IEEE Trans. VLSI Syst.* **2**, 516–521.

Sulieman, M. H. (2009). *On the Reliability of Interconnected CMOS Gates Considering MOSFET Threshold-Voltage Variations*. Schmid, A., Goel, S., Wang, W., Beiu, V., Carrara, S. (eds.) Nano-Net, LNICST, vol. 20. Springer, Heidelberg.

Sulieman, M. H., V. Beiu and W. Ibrahim (2010). Low-power and highly reliable logic gates: Transistor-level optimizations. In: *Proc. IEEE Intl. Conf. Nanotech. (IEEE-NANO)*. pp. 254–257.

Suran, J. J. (1964). Use of circuit redundancy to increase system reliability. In: *Proc. Intl. Solid-State Circ. Conf. (ISSCC)*. pp. 82–83.

Tryon, J. G. (1960). Redundant logic circuit. *US Patent* 2,942,193.

von Neumann, J. (1956). *Probabilistic Logics and the Synthesis of Reliable Organisms from Unreliable Components*. Shannon, C.E., McCarthy, J. (eds.) Automata Studies, pp. 43-98. Princeton Univ. Press, Princeton, NJ.

Wakerly, J. F. (1975). Transient failures in triple modular redundancy systems with sequential modules. *IEEE Trans. Comp.* **C-24**, 570–573.

Wakerly, J. F. (1976). Microcomputer reliability improvement using triple-modular redundancy. *Proc. IEEE* **64**, 889–895.

Winograd, S. and J. D. Cowan (1963). *Reliable Computation in the Presence of Noise*. MIT Press, Cambridge.

Zarkesh-Ha, P. and A. A. M. Shahi (2010). Logic gates failure characterization for nanoelectronic EDA tools. In: *Proc. Intl. Symp. Defect and Fault Tolerance VLSI Syst. (DFT)*. pp. 16–23.

Zarkesh-Ha, P. and A. A. M. Shahi (2011). Stochastic analysis and design guidelines for CNFETs in gigascale integrated circuits. *IEEE Trans. Electr. Dev.* **58**, 530–539.

Zhao, W. and Y. Cao (2007). Predictive technology model for Nano-CMOS design exploration. *ACM J. Emerg. Tech.* **3**, 1–17.