

Performance and Reliability Analysis of a Scaled Multi-Switch Junction Crossbar Nanomemory and Demultiplexer

Ayodeji Coker, *Student Member, IEEE* and Valerie Taylor, *Senior Member, IEEE*

Abstract—This paper presents a performance and reliability analysis of a scaled crossbar molecular switch memory and demultiplexer. In particular, we compare our multi-switch junction fault tolerance scheme with a banking defect tolerance scheme. Results indicate that delay and power scale linearly with increasing number of redundant molecular switch junctions. The multi-switch junction scheme was also shown to achieve greater than 99% reliability for molecular switch junction failure rates less than 20%, when a redundancy of at least 3 was implemented. In contrast, the banking scheme was only effective for molecular switch junction failure rates of less than 1%, which requires over three times the number of banking modules.

I. BACKGROUND

HYBRID crossbar nanomemories are showing great potential as future memory candidates. Advances in nano-wire fabrication techniques and synthesis of bistable-molecular switches [1] are making it more practical to fabricate ultra dense molecular switch crossbar nanomemories. Green et al [2] demonstrated a 160-kbit crossbar molecular electronic memory with a fabrication density 10^{11} bits cm^{-2} . Snider et al [3] also reported a defect-tolerant field-programmable nano-wires interconnect architecture for enabling hybrid nano/ CMOS architectures.

II. INTRODUCTION

Crossbar molecular switch memories are grid-like memory structures comprised of intersecting nanowires. Bistable molecules are deposited at the junctions of each intersecting nano-wires. These molecular switch junctions alter their resistive state when exposed to an applied voltage field. The dimensions in which they are fabricated make them highly susceptible to fabrication defects and transients faults, thus

making reliability an important factor in designing nanomemories.

The objective of this work is to analyze the performance using defect tolerant architectures with the design of molecular switch crossbar nanomemories and demultiplexers. In particular, we focus on the following performance metrics; delay, power and reliability. The defects considered are those related to defective molecular switch junctions that occur when not enough molecules are deposited such that switching effects can be observed; transient errors in the molecular switches due to environmental effects are also considered in our model. These defects or faults are nondeterministic, and as such, the probabilistic model checker PRISM [4], was used for reliability measurements. Further, faults related to defective nanowires can occur and will be considered in future work.

Crossbar nanomemories and demultiplexer circuits were modeled using HSPICE [7]. In previous work we developed a Multi-Switch Junction (MSJ) fault tolerant scheme [5], that utilizes redundancy in nanowires of the crossbar nanomemories, we presented results based on a parameterized crossbar nanomemory circuit model, with no demultiplexer. In this work, a more detailed scaled model of the crossbar nanomemory and demultiplexer are studied, and compared with a banking fault tolerant scheme. Results show the MSJ is able to improve the nanomemory reliability to achieve greater than 99% for redundancies of $k \geq 3$. Also, the MSJ was found to work better for smaller nanomemory arrays, because the larger the array size, the larger the number of fault prone molecular switch junctions. The reliability analysis of the crossbar nanomemory and demultiplexer are also presented, with results showing an improvement in reliability as a result of the MSJ scheme.

The MSJ scheme is compared to the banking scheme [6] which achieves defect tolerance through modular redundancy. PRISM was used to simulate a condition where each molecular switch junction has a 1% probability of failing; at this error rate the MSJ can ensure the memory works with a larger 99% probability for an 8×8 array with $k=3$. The reliability of a non-defect-tolerant 8×8 nano-memory array with a molecular switch failure probability of 1% has a nanomemory array reliability of 53%. Using banking to implement a 16×16 array (i.e four 8×8 banks) requires a factor

Contacting Author; Ayodeji Coker is with the Department of Computer Science, Texas A&M University, College Station, TX 77843 USA. Phone: 979-845-3907, (e-mail: coker@cs.tamu.edu).

Valerie Taylor is with the Department of Computer Science, Texas A&M University, College Station, TX 77843, USA. (e-mail: taylor@cs.tamu.edu).

Acknowledgement: Work supported by NASA under award no. NCC 2-1363.

of three increase in the number of banks to achieve larger 99% reliability. In comparison, MSJ requires $k \geq 3$ to achieve the same level of reliability.

The rest of this paper is organized as follows; Section III discusses the MSJ and banking scheme implementation for the crossbar nanomemory. Section IV discusses the demultiplexer implementation. In Section V the circuit analysis, simulation parameters, and results are presented. Section VI discusses the reliability analysis and results. Finally, Section VII concludes the paper.

III. CROSSBAR NANOMEMORY FAULT-TOLERANT SCHEMES

A. MULTI-SWITCH JUNCTION SCHEME

The grid-like geometry of crossbar architecture makes it a natural candidate for implementing redundancy-based, defect tolerant architectures. The MSJ architecture is based on the direct introduction of redundant nano elements into the fabric of the crossbar nanomemory structure. As illustrated in Fig 1, defect tolerance in the molecular switch crossbar memory is achieved by utilizing redundant nanowires in the rows and columns of the device. Redundancy in this work is defined as the number of row or column NWs (k) attached to the same contact electrode; these NWs constitute the bit line or word line. It has been demonstrated in [20] that by taking advantage of the quantized nature of the separation between NWs, which are assembled via the Langmuir-Blodgett method, the ratio of the number of NWs per electrode can be defined, thereby reinforcing the practicality of the MSJ scheme. The MSJ scheme possesses the added advantage of not requiring additional peripheral circuitry for implementation.

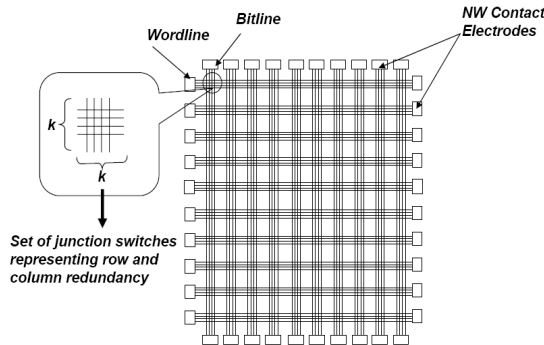


Figure 1. MSJ architecture for the crossbar nanomemory, k is the number of nanowires used to represent a single row or column.

B. BANKING

Banking has been suggested as another viable means of achieving defect-tolerance in crossbar nanomemories [9, 10, 11]. Banking partitions the nanomemory into smaller memory blocks in order to lower access time, and increase power

conservation and defect-tolerance. It involves the use of redundant memory modules to tolerate defects that occur in the main nanomemory module. Choi et al [11] proposed a variation of this scheme that is comprised of two parts; the main molecular memory modules along with spare modules, and the CMOS based circuitry for providing power, address translation and the requisite logic for the molecular memory modules. Given n number of bits, and m molecular memory modules; a $2n \times 2n$ nanomemory can be addressed using $\log_2(2n)$ bits. The number of molecular memory modules and spares are given by $2^{2(n-m)} + s$, where s is the number of spare modules required to achieve defect-tolerance. Modules are selected using a module table and decoder implemented in the CMOS layer. A module is deemed defective if the module cannot form the desired size memory array.

C. MULTI-SWITCH JUNCTION ENHANCED BANKING SCHEME

The MSJ and banking schemes were also combined for optimal reliability, by implementing the MSJ scheme in the redundant bank modules. This approach will increase the reliability of each bank module, thus reducing the total number of redundant bank modules required to achieve the nanomemory reliability. This approach is also important because there is a limit to the degree to which MSJ crossbar nanomemory can be scaled; a fact largely resulting from constraints placed on the lengths of the NW interconnects due to deficiencies in current NW fabrication techniques.

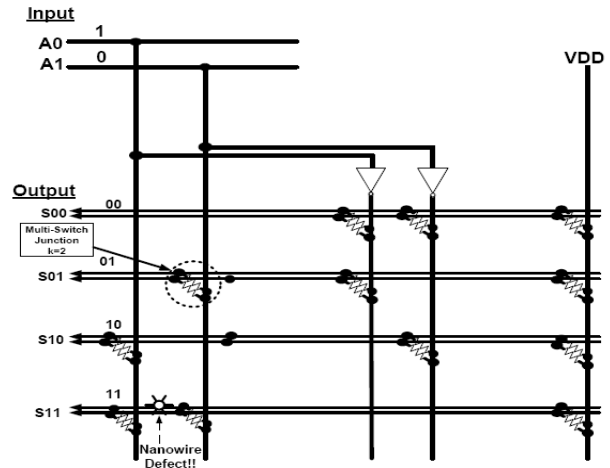


Figure 2. A molecular RAM demultiplexer with an additional NW inserted to each row ($k=2$) to implement the MSJ scheme.

IV. CROSSBAR DEMULTIPLEX DEFECT-TOLERANT SCHEMES

A. MULTI-SWITCH JUNCTION DEMULTIPLEX

Demultiplexers represent one of the most vital components of a nanoscale memory. Faults that originate at the decoder can propagate through the entire memory system. The

demultiplexer architecture is laid out in a crossbar configuration, as illustrated in Fig 2 which is an example of the configuration for a 4×4 crossbar nanomemory demultiplexer; with two input address signals, A0 and A1, which drive four Microwire (MW) column signal lines. The MW and NW connections form an AND gate functionality the address line output. Hence, a NW address line can only be selected if its two input signals are high or set to their “1” state.

The MSJ scheme can also be introduced into the demultiplexer architecture to achieve defect-tolerance. Only redundant NWs are implemented, the MW CMOS interfaces have less than 1% probability of being defective, and as such do not warrant incurring the large area penalty of implementing redundant MWs— Fig 2 illustrates.

V. CROSSBAR NANMOMEMORY AND DEMULTIPLEX CIRCUIT MODEL AND ANALYSIS

In previous work, a parameterized circuit model was utilized to analyze the performance of the crossbar nanomemory [5]. Here, a full scale model of the crossbar nanomemory and demultiplexer are implemented; the motivating factor behind this approach was a more detailed understanding of the fault tolerance induced performance penalties incurred in the crossbar nanomemory and demultiplexer devices. The circuit models were implemented and simulated in HSPICE. The bistable molecular switch junctions were modeled as resistors and the redundant NWs were modeled as interconnecting RLC transmission lines as described by Burke et al [15] and illustrated in Fig 3.

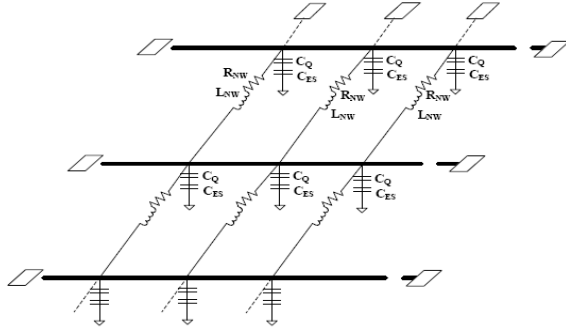


Figure 3. RLC network schematic for crossbar nanomemory.

C_{ES} and C_Q are the electrostatic and quantum capacitance, given in Equations (1) and (2) respectively. Where v_f (m/s) is the Fermi velocity of the NW.

$$C_{ES} = \frac{L^4 \epsilon_0 \epsilon_r}{\pi \ln(4d/w)} \quad (1) \quad C_Q = \frac{2e^2}{hv_f} \quad (2)$$

The capacitance of the MW was modeled as the parallel plate capacitance [Equation (3)] between the MW and the chip substrate as symbolized by the parameter d_{dist_sub} in Equation (3).

$$C_{MW} = \left(\frac{\epsilon h L}{d_{dist_sub}} \right) \quad (3)$$

The NWs were modeled as copper nanowires, with resistance per unit length was computed using the resistivity of bulk copper material [3]—according ITRS specifications—as described in Equation (4). The resistances of the demultiplexer MW address lines were also computed in using Equation (4), adjusted to meet the MW dimension parameters.

$$\frac{R}{L} = \frac{\rho}{A} \quad (4)$$

Only the kinetic inductance L_{k_NW} is used in the aforementioned NW interconnect model because in one dimensional systems kinetic inductance always dominates magnetic inductance [18]. Typical inductance values in CMOS on-chip environments have been found to be approximately 1nH/mm or less [19]. The inductance of the demultiplexer address line MWs were modeled using Equation (5).

$$L_{k_NW} = \frac{h}{2e^2 v_f} \quad (5)$$

The redundant NWs described in Fig 1 are connected to the same contact electrode, they are assumed to be densely spaced and driven by the same voltage source. Further, because the redundant NWs switch in tandem, the effective parasitic capacitance between them is negligible [14].

A. SIMULATION SETUP

An HSPICE nanomemory simulator was designed and constructed to conduct the delay and power analysis. Delay is defined in this work as the time required to access and read stored information from a selected bit. A worst case analysis was performed (i.e. the target bit assessed was the bit farthest from the voltage driver source and the sense amplifiers as indicated in Fig 4).

Nanowire width [3]	15nm
Nanowire pitch	33nm
Microwire width	90nm
Microwire pitch	180nm
“ON” Resistance (R_{ON}) [1]	0.48M Ω
“OFF” Resistance (R_{OFF})	9.2M Ω
Load Resistance (R_{load}) [8]	$\sqrt{(R_{ON})(R_{OFF})}$

Table 1. Model parameters

All bits on the assessed wordline and bit line were set to their low resistive “ON” state, while the rest of the non-accessed bits were reverse biased. The NW contact electrodes were modeled as voltage sources and the sense

amplifiers as load resistors, as (see Fig 4). Model parameters are provided in Table 1.

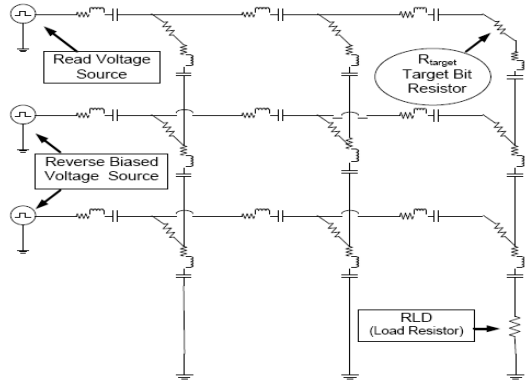


Figure 4. Scaled 3×3 model of the crossbar nanomemory.

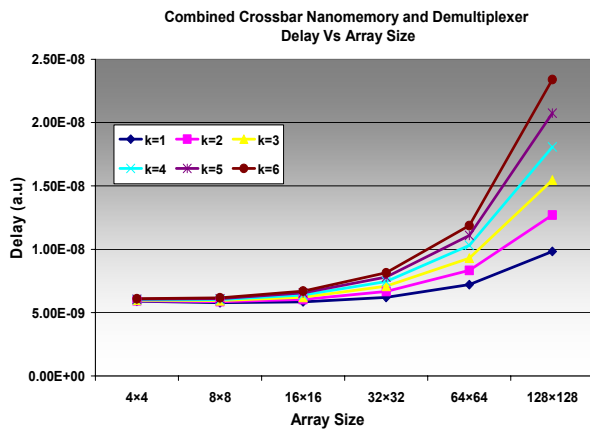
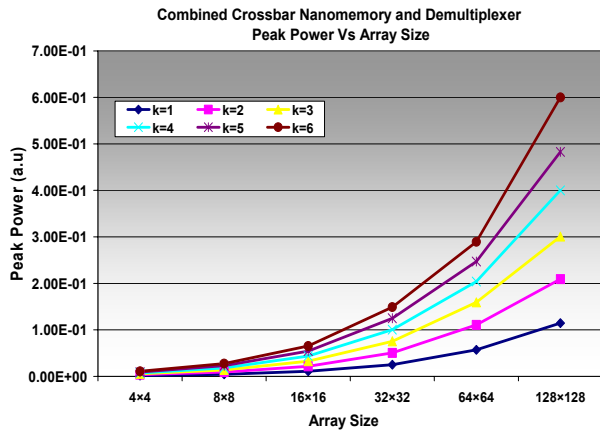


Figure 5. Signal delay measurement of selecting the memory address with the demultiplexer and reading a bit with the crossbar nanomemory.



(a) shows the access time delay of the nanomemory and demultiplexer when increasing levels of k are implemented. (b) shows the peak power dissipation trend with increasing degrees of k .

B. RESULTS

The results indicate that access time delay and power dissipation scale with increasing redundancy. Given that the array sizes increase exponentially, a trend which is further amplified by the inclusion of redundant elements, a somewhat linear scaling as presented in Figs 5 & 6 is a relatively small penalty to pay for improved reliability. Simulation results, not shown here due to space limitations, also indicate that access time delay penalty with increasing redundancy is highest in the crossbar nanomemory when compared with the demultiplexer. On the other hand, peak power dissipation penalty is highest in the demultiplexer, which can be attributed to the microwires present in the demultiplexer.

VI. RELIABILITY ANALYSIS

The probabilistic model checking tool PRISM [16, 17], was used to conduct the reliability analysis of the defect-tolerant schemes. Probabilistic model checking is basically an algorithmic methodology for determining the ability of a given probabilistic system to adhere to specified probabilistic parameters. The defects studied in this work are those that constitute stuck-open faults in the molecular switch junctions. The PRISM model assumes all junctions to have an equal likelihood of failing, which is user defined. PRISM is used in the context of this work, to evaluate stuck-open faults by modeling them as the probability of failure to program at least one of the redundant molecular switch junctions to the desired logic values.

The crossbar nanomemory is generally regarded as a state transition system, and can be modeled as a generic Markov decision process (MDPs). MDPs can be defined as a model of computation that is able to express the non-determinism of choosing the probability distributions from any given state. The MSJ architecture is modeled as an MDP which takes as input memory size ($n \times m$), the molecular switch junction failure probability, and the redundancy (k). The model assumes an independent, identical and unclustered fault rate for all the molecular switch junctions. The PRISM models were verified theoretically using a base case binary model.

A. RESULTS

All simulations results indicate substantial improvements in the defect tolerance capabilities of the crossbar nanomemory and demultiplexer with the implementation of the MSJ scheme. As can be seen in the graphs of Figs 7 & 8, the MSJ scheme has a greater impact on the reliability of smaller nanomemory arrays. This is mainly due to the fact that as the molecular switch junctions are replicated by implementing the MSJ scheme, more defect prone junctions are added to the simulated nano-electronic devices. In the case of the crossbar nanomemory devices, our simulations show that a reliability of greater than 92% can be achieved for molecular switch junction defects greater than 50% and a redundancy $k \geq 4$ is

implemented for array sizes greater than 64×64 —the reliability increases to greater than 99% when $k \geq 5$.

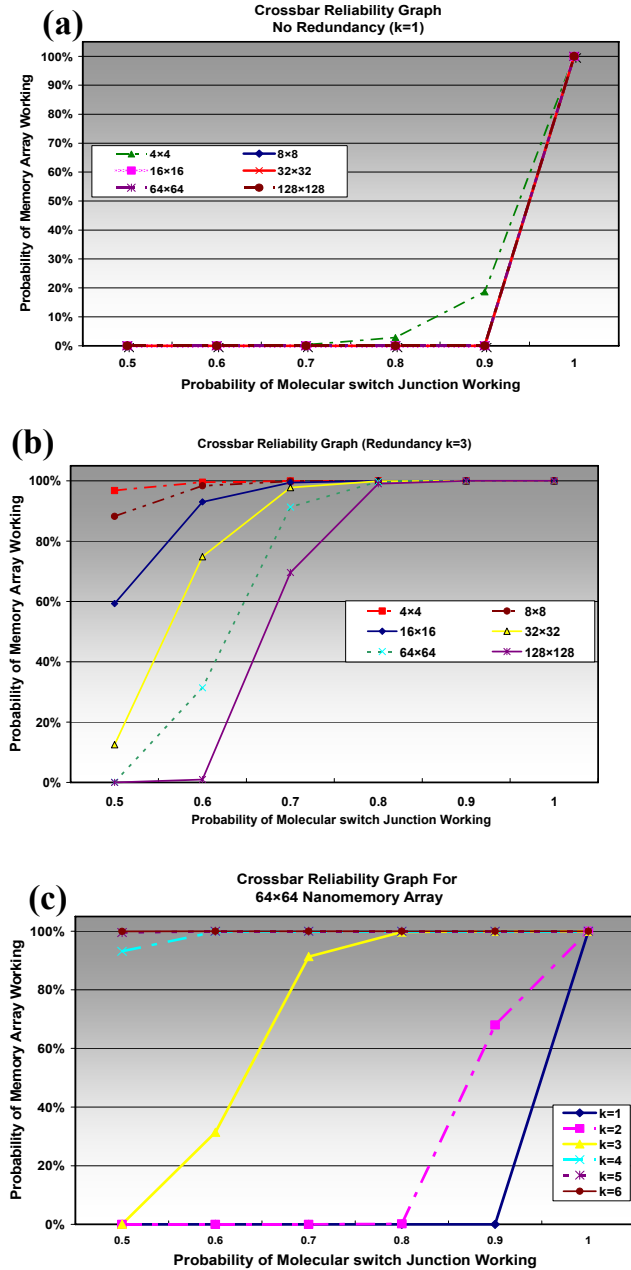


Figure 6. (a) shows the reliability of a crossbar nanomemory with no implemented fault tolerance. (b) shows the clear improvement in the reliability of the nanomemory at redundancy $k=4$. (c) shows the trend in reliability improvement with increasing k for a 64×64 nanomemory array.

In the case of the demultiplexer, a greater degree of redundancy is required to obtain a reliability rate equivalent to that of the crossbar nanomemory. When the MSJ scheme is implemented in the demultiplexer the molecular switches are replicated at a value equal to k . In the crossbar nanomemory

the replicated switches are equal to k^2 , which is a function of the crossbar nanomemory and demultiplexer architecture, as illustrated in Figs 1 & 2 respectively.

Array sizes of up to 128×128 were simulated because large computing running times made it difficult to simulate larger array sizes; however, efforts are being made to find a more efficient testing platform for future works.

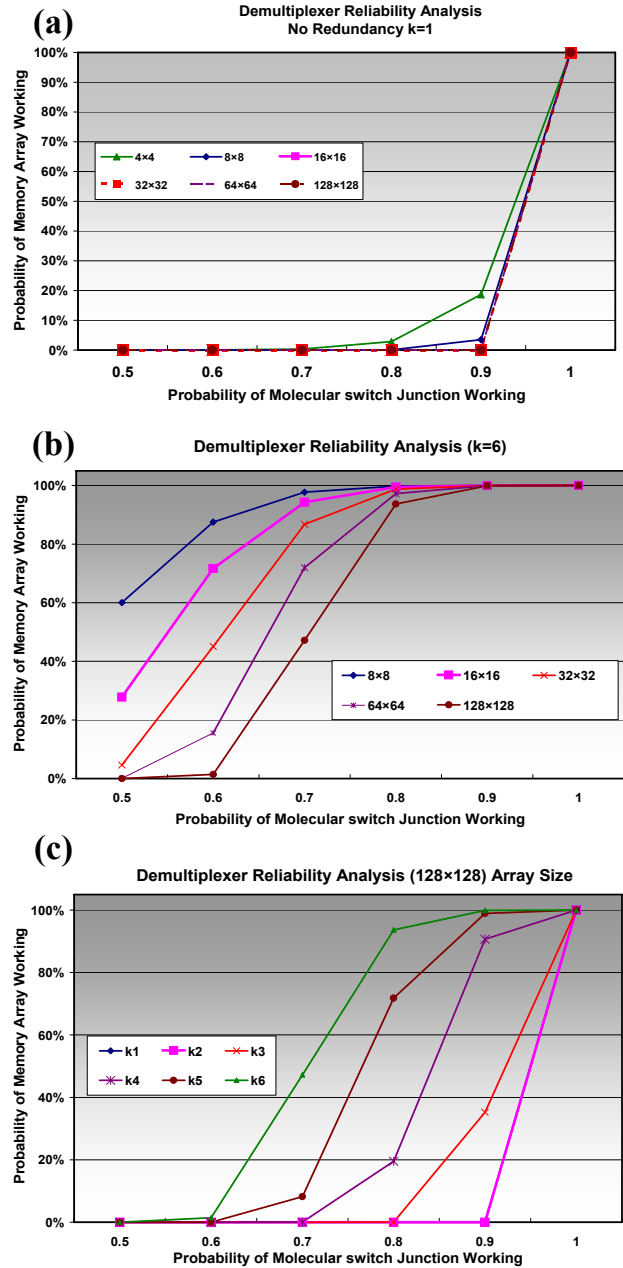


Figure 7. (a) shows the reliability of the nanomemory demultiplexer with no implemented fault-tolerance. (b) shows the clear improvement in the reliability of the demultiplexer at redundancy $k=6$. (c) shows the trend in reliability improvement with increasing k for 128×128 demultiplexer array.

Comparisons between the MSJ and banking schemes were also studied. In this experiment, PRISM was used to simulate a condition where each molecular switch junction has a 1% probability of failing; at this error rate the MSJ can ensure the memory works with a greater than 99% probability for an 8×8 array with a $k=3$, as presented in Fig 7a. The reliability of a non-defect tolerant 8×8 nanomemory array was also simulated— results show that given a molecular switch failure probability of 1%, the reliability of the nanomemory array was 53%. The cumulative binomial distribution formula was then used to determine the probability of choosing a specified number of good banks from a total number of defect prone redundant banks. To implement a 16×16 nanomemory array configuration, four 8×8 banks are required. In the presence of a 1% molecular junction failure probability, over four times the number of required banks are needed to achieve a greater than 99% probability of the nanomemory array working. Similarly, a 128×128 nanomemory array will require 256 (8×8) banks for implementation and over 535 (8×8) banks at a 1% molecular junction failure probability, to achieve greater than 99% probability of the nanomemory array working. In comparison, MSJ requires $k \geq 3$ to achieve the same level of reliability (Fig. 2). Delay and Power simulation results indicate similar performance advantages for the MSJ scheme over the banking scheme.

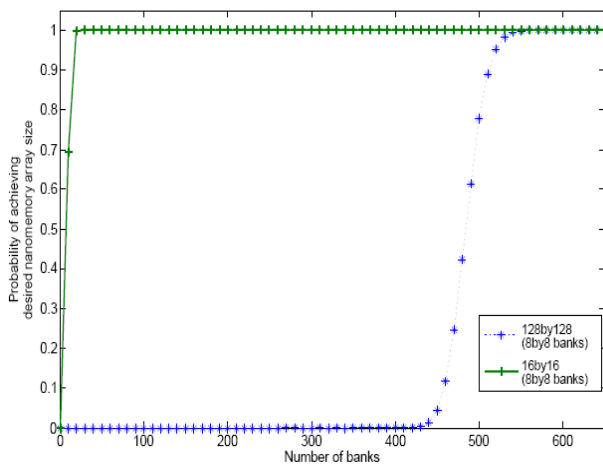


Figure 8. Data showing the number of 8×8 banks required to fabricate a 128×128 and 16×16 nanomemory array with a 1% molecular junction failure probability.

VII. CONCLUSION

The performance analysis of a combined crossbar nanomemory and demultiplexer has been presented. The introduction of the MSJ defect-tolerant scheme into the architecture of the nano-devices was found to improve the reliability of both the crossbar nanomemory and demultiplexer at a minimal cost to access time delay and power dissipated. Simulation results also show that the MSJ was most effective for smaller array sizes.

The banking scheme of defect tolerance was also studied and found to provide less reliability than the MSJ scheme. Current simulation results suggest a combined MSJ and Banking scheme as the best way of achieving the desired optimal reliability. Smaller degrees of redundancy are needed to achieve high rates of reliability in smaller crossbar configurations, by having smaller more reliable redundant crossbar nanomemory and demultiplexer modules, a faster nanomemory with a conceivable parallel operation can be achieved.

REFERENCES

- [1] Y. Chen, G. Jung, D. A. A. Ohlberg, X. Li, D. Stewart, J. O. Jeppesen, K. A. Nielsen, J. F. Stoddart, and R. S. Williams, "Nanoscale molecular-switch crossbar circuits" *Nanotechnology* **14**, 462 (2003)
- [2] J. E. Green, J. Wook Choi, A. Boukai, Y. Bunimovich, E. Johnston-Halperin, E. DeIonno, Y. Luo, B. A. Sheriff, K. Xu, Y. Shik Shin, H-R. Tseng, J. Fraser Stoddart and J. R. Heath, "A 160-kilobit molecular electronic memory patterned at 10^{11} bits per square centimetre," *Nature* **445**, 414-417 (2007).
- [3] G. Snider and R. Williams "Nano/CMOS architecture using a field programmable nanowire interconnect" *IOP Nano* **18** (2007) 035204 (11 pp).
- [4] M. Kwiatkowska, G. Norman, and D. Parker, "PRISM 2.0: A tool for Probabilistic model checking," *QEST'04*, IEEE Computer Society Press, 2004, pp. 322–323.
- [5] A. Coker, V. Taylor, D. Bhaduri, S. Shukla, A. Raychowdhury, K. Roy "Multi-Junction Fault Tolerance Architecture for Nanoscale Crossbar Memories" *IEEE NANO* 2006.
- [6] A. Dehon, S. Goldstein, P. Kuekes and P. Lincoln "Non-photolithographic Nanoscale Memory Density Prospects" *IEEE Transactions on Nanotech*, Vol. 4, NO. 2, March 2005
- [7] Avant! Corporation Star-HSPICE 2001.2 and AvantWaves 2001.2
- [8] C. Amsinck, N. Di Spigna, S. Sonkusale, D. Nackashi, and P. Franzon, "Scaling constraints in nanoelectronic random access memories" *IOP Publishing, Nanotechnology* **16** (2005) 2251-2260
- [9] M. Ziegler, C. Picconatto, J. Ellenbogen, A. Dehon, D. Wang, Z. Zhong C. Lieber "Scalability Simulations for Nanomemory Systems Integrated on the Molecular Scale". *Ann N.Y. Acad. Sci* **1006**: 312-330 (2003)
- [10] A. Dehon, S.C. Goldstein, P.J. Kuekes and P. Lincoln "Nonphotolithographic Nanoscale Memory Density Prospects" *IEEE Transactions on Nanotechnology*, Vol. 4, NO. 2, March 2005
- [11] Y-H Choi and Y.K. Kim "A nanoscale scalable memory architecture for molecular architecture electronics" *Nanotechnology*, October 2004, vol. 15, no. 10, pp. S639-S644(1)
- [12] Philip J. Kuekes, Warren Robinett and R. Stanley Williams, "Improved voltage margins using linear error-correction codes in resistor-logic demultiplexers for nanoelectronics." *IOP Publishing, Nanotechnology* **16** (2005) 1419-1432
- [13] S. Lin, D. Costello, *Essentials of Error-Control Coding techniques, 2nd Edition*, Prentice-Hall, Inc. Upper Saddle River, NJ, 2004
- [14] K. Hirose, H. Yasuura, "A bus delay reduction technique considering crosstalk", *Proceedings of DATE*, p.441-445, March 2000, France.
- [15] P.J. Burke, "Luttinger liquid theory as a model of the gigahertz electrical properties of carbon Nanotubes," *IEEE Transactions on Nanotech*, vol. 1, pp. 129-144, 2002.
- [16] M. Kwiatkowska, M., Norman, G., and Parker, D. Prism: Probabilistic symbolic model checker. In *TOOLS 2002* (April 2002), vol. 2324 of LNCS, Springer-Verlag, pp. 200–204.
- [17] Web Page: www.cs.bham.ac.uk/~dxdp/prism/.
- [18] P.J. Burke, "AC performance of nanoelectronics : towards a ballistic THz nanotube transistor" *Solid-State Electronics* **48** (2004) 1981–1986
- [19] R. Kapur, J.P. McVittie and K.C. Saraswat "Technology and Reliability Constrained Future Copper Interconnects—Part II —Performance Implications." *IEEE Transactions Electron Devices* Vol49, no. 4 April 2002
- [20] S. Jin, D. Whang, M. McAlpine, R. Friedman. Interconnection and Integration of Nanowire Devices without Registration," *Nano Lett.* **4**, 915-919 (2004).