Scaling Challenges for the Cross-point Resistive Memory Array to Sub-10nm Node – An Interconnect Perspective

Jiale Liang, Stanley Yeh, S. Simon Wong, and H. -S. Philip Wong
Dept. of Electrical Engineering and Center for Integrated Systems, Stanford University, Stanford, CA 94305, USA
liangjl@stanford.edu

***Abstract*— The impact of Cu interconnect scaling on the write/read margin, energy dissipation, speed and reliability of resistive cross-point memory array are quantitatively examined for wire sizes down to the sub-10nm node. The impending resistivity increase due to wire scaling results in significantly degraded write and read windows, substantial interconnect energy, and increased wire latency. The growing current density required for programming exacerbates the Cu electromigration and is a reliability concern for deeply-scaled technology nodes. Performance degradations are strongly dependent on the memory device parameters and memory array sizes: r_{on} below 100KΩ and array size > 1Mb lead to write margin < 55%, read margin < 5%, and wire energy > 1pJ for wire size smaller than 20 nm. Also, a large r_{on} value can tolerate a small r_{off}/r_{on} ratio, although a too high r_{on} would result in a slow speed. This work points to the importance of a careful device and interconnect co-optimization to meet the performance specifications for cross-point memory arrays at sub-10nm nodes.**

Key words*: *cross-point, interconnect, scaling, write/read margin, energy, latency, reliability, resistive switching memory

I. Introduction

Cross-point resistive memory, with a passive bistable material sandwiched between periodically crossed wordlines and bitlines, is widely regarded as the targeted architecture for the next generation non-volatile memory [1]. To achieve the advantage of the highest device density with minimum feature size offered by the cross-point concept, the wordline/bitline in an array has to be scaled at the same rate as the memory elements. This means the wordline/bitline width will soon be smaller than the intrinsic electron mean free path of Cu (~39 nm) and approach sub-10 nm regime. As a result, the "size effect" of Cu wire is triggered which can significantly increase the Cu resistivity over the bulk value [2]. Moreover, for emerging memories, the programming current is often not scaling down with the device dimensions. Thus, the current density in the Cu wires could rise well above its electromigration threshold and raise reliability concerns. All these issues point out the importance to study the scaling challenges and performance limiters for the cross-point memory arrays from an interconnect perspective.

This work explores the impacts of Cu wire scaling on the: 1) write/read margin, 2) energy consumption, 3) speed and 4) reliability of the cross-point memory at different array sizes and device parameters for technology nodes down to sub-10 nm regime. The dramatic degradations of the write/read margin and speed, the substantial energy dissipation of wires, and the imminent reliability issue present big challenges for realizing >1Mb cross-point memory array with minimum feature size beyond the 10 nm node.

II. "Size Effect" of Cu Wires

As the dimension and grain size of Cu wires scale down to a regime that is comparable to the mean free path of Cu, surface scattering and grain boundary scattering become substantial and result in the size-dependent resistivity of Cu wires [3]. In our analysis, we quantified these two effects using the well-known combined Fuchs-Sondheimer and Mayadas-Shatzkes (FS-MS) model [4, 5], with specular scattering fraction = 0.25 and probability of reflection at the grain boundary=0.3[2, 3], Without loss of generality, the wire height is assumed to be same as the width, and the grain size is assumed to be equal to the wire width [6]. The barrier thickness of Cu wire is chosen as 2nm for each side [7]. Fig. 1(a) shows the calculated resistance per unit length (r_{unit}) of Cu wires as wire dimension scales and Fig. 1(b) gives the corresponding resistance per unit cell (r_c) in the cross-point memory array. As we can see, Cu wire experiences a significant increase in r_{unit} (and hence r_c) as wire size shrinks.

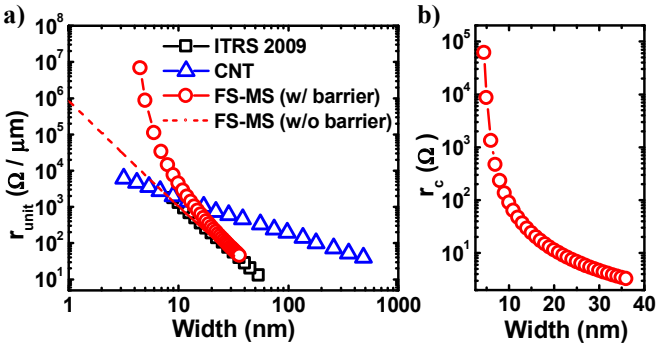

Fig. 1. (a) r_{unit} of Cu wire for different wire dimensions and (b) r_c in the cross-point memory array. The ITRS projection of Cu wire [8] and the modeled single-wall CNT r_{unit} (MFP=1μm) [9] are also plotted in (a) as a comparison (see discussion in Section III.D).

III. Effect of Wire Scaling on Cross-point Memory

A. Write and Read Performance

The increase of wire resistance at smaller size may cause mis-writing and mis-reading in cross-point memory arrays. In this work, the cross-point memory array is modeled as a purely passive m × n resistor network. The bistable switching material is considered as a resistance with two distinct states (r_{on} and r_{off}), and the memory array is assumed to be driven from one end. A generic illustration of the cross-point memory array is given in Fig. 2a. Note that the supply voltages for all the unselected wordlines or bitlines are the same during the write/read operation, and hence can be combined. For simplicity, we will only examine the worst case scenarios. The worst case selected cell for both write and read operations is

978-1-4673-1081-9/12/$31.00 ©2012 IEEE

the one that suffers the largest parasitic wire resistance (r_{wire}), and this cell is located in the farthest corner of the array (r_{src}) [10]. To simplify the model, we divide the whole memory array into four groups: r_{sr}, r_{src}, r_{sc} and r_u, as shown in different colors in Fig. 2a. The cell states are assumed to be same in each group, but they can be different from each other. Although this four-group division is a great simplification, it is able to outline the worst case write and read margins [10]. With all the conditions given, to examine the cross-point memory array with practical size (~Mb) in a time-efficient and flexible manner, the resistor network can be further simplified to a 2×2 matrix using a reduced circuit model as shown in Fig. 2b. The model provides a closed form analytical solutions to the voltage and current in primary nodes, and is applicable to both the write and the read operations in cross-point arrays with arbitrary size and resistance values.

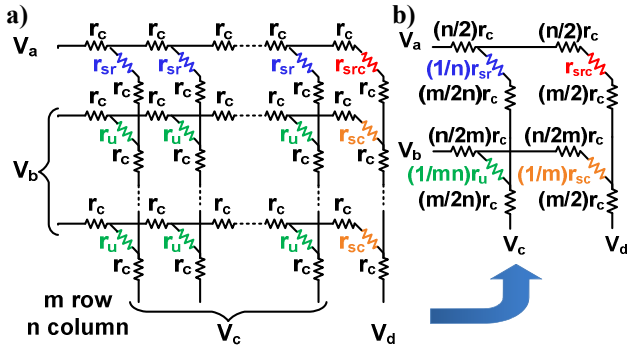

Fig. 2 (a) Schematic view of cross-point memory array. (b) Reduced circuit model of a m×n cross-point memory array including the wire resistance for analytical calculation.

TABLE 1
BASELINE RESISTANCE VALUES AND APPLIED VOLTAGES FOR THE WORST CASE WRITE AND READ OPERATIONS

Parameter	Write	Read "0"	Read "1"
V_a	V_{dd}	V_r	V_r
V_b	$V_{dd}/2$	0	0
V_c	$V_{dd}/2$	0	0
V_d	0	0	0
r_{src}	100 KΩ	100 KΩ	1 MΩ
r_{sr}	10 MΩ	100 KΩ	1 MΩ
r_{sc}	10 MΩ	100 GΩ	100 GΩ
r_u	100 GΩ	100 GΩ	100 GΩ

Write operation

For the write operation, we use the $V_{dd}/2$ write scheme [11]. The worst case array pattern occurs when all the memory cells are at r_{on}, and hence r_{src} sees the smallest accessed voltage (V_{access}). The corresponding baseline resistance values and applied voltages are summarized in Table 1.To focus our study on the impact of wire scaling, we will assume an idealized cell selector with a very high, 100GΩ off-resistance, 10MΩ half-on resistance and zero on-resistance is used in series with each unselected, half-selected and selected memory cell.

Fig. 3 shows the HSPICE-simulated and analytically-modeled write margin (V_{access}/V_{dd}) vs. r_c for different array sizes with good matching. The increase of r_c with size scaling causes a sharp increase in parasitic voltage drop along the wires and hence diminishes the V_{access} required for a successful write. Applying the model to a 1Mb square array (Fig. 4a), we can see that the write margin suffers a larger degradation compared to the small array sizes in Fig. 3. Also from Fig. 4a, the write margin strongly depends on the value of r_{on}. The smaller the r_{on}, the worse the V_{access}/V_{dd}. This is due to the increased voltage division between the wire and the memory cell on-resistance as r_{on} decreases. Therefore, in order to achieve a successful write in a deeply-scaled cross-point memory array, the memory array will either need to be operated at a higher V_{dd} (and hence higher power) to compensate for the voltage loss on the wires, or it has to be divided into smaller memory partitions or the bitline/wordline needs to be strapped by a wider metal layer above/below the bitline/wordline at the cost of increased chip area (decreased array efficiency). For the latter two approaches, Fig. 4b illustrates the maximum allowable memory partition size (m × m) or the longest distance between vias for the metal strap as wire shrinks for different V_{access}/V_{dd} ratios. Memory partition smaller than 230×230 or via distance < 230 cell long has to be used to ensure $V_{access}/V_{dd} > 0.6$ at the 10 nm technology node.

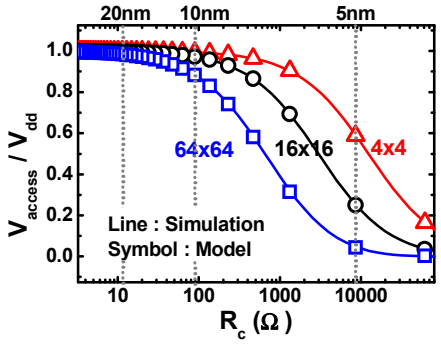

Fig. 3 HSPICE-simulated and analytically modeled V_{access} vs. r_i showing a good match.

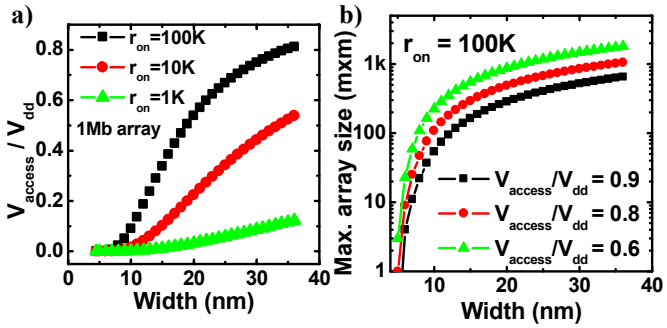

Fig. 4. (a) V_{access} degradation with wire scaling for different r_{on} values in a 1Mb square array. (b) Maximum memory partition size (m×m) or longest distance between vias for the metal strap as wire dimension scales for different V_{access}/V_{dd} ratios.

Read operation

For the read operation, we supply V_r on the selected wordline and ground all other wordlines and bitlines to enable the parallel read out of all the bits on the same row. The worst case array patterns when reading r_{on} and reading r_{off} are listed in Table 1. The current difference (ΔI) when reading the two states of the memory cell is sensed by sense amplifiers (SAs) which are connected in series with each bitline and function as current-to-voltage converters. The trans-resistance (r_{tran}) of the SAs is assumed to match r_{on} to achieve a rail-to-rail output voltage ($\Delta V = r_{tran} * \Delta I$). As shown in Fig. 5, our reduced circuit

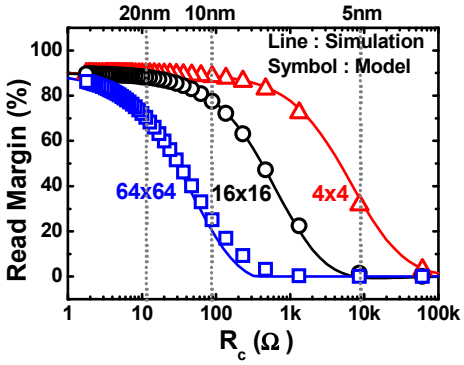

Fig. 5. HSPICE-simulated and analytically modeled read margin vs. r_c showing a good match

model achieves a good match with the HSPICE-simulated worst case read margins ($\Delta V/V_r$). A strong reduction in $\Delta V/V_r$ with wire size is observed. There are two sources of degradation: first, the increase of Cu wire resistance due to the size effect results in the decrease of the effective resistance ratio $(r_{wire}+r_{off})/(r_{wire}+r_{on})$; this makes it harder to discriminate the two states of the memory cell. Secondly, due to the larger voltage loss on wires at a larger read current, the effective read voltage across r_{src} is much smaller when reading r_{on} than reading r_{off}. This results in a significant reduction in read current of r_{on} and hence diminishes ΔI. To compare these two degradation sources, the read margin of a 1Mb square cross-point memory array with increasing r_{on} but fixed r_{off} is calculated by the reduced circuit model (Fig. 6a). Two competing effects are noted here: on one hand, the increase of r_{on} results in a smaller voltage loss on wires, and is beneficial to the read margin; on the other hand, the increase of r_{on} at fixed r_{off} causes a decrease in r_{off}/r_{on} ratio, and it is detrimental to the read margin. The improvement of $\Delta V/V_r$ at larger r_{on} shown in Fig. 6a illustrates the dominant effect of resistance value for the array size and r_{off} value considered here. For a more comprehensive analysis, the variation of normalized read margin as wire scales for different fixed r_{off} values and array sizes is shown in Fig. 6b. Here the read margin at large r_{off}/r_{on} (100×) but small r_{on} is normalized to the case with small r_{off}/r_{on} (10×) but large r_{on}. We can see this normalized read margin is > 1 only for wire width > 15nm and r_{off} >10GΩ. This indicates that r_{on} is the more dominant factor than r_{off}/r_{on} for determining the read margin for most cross-point memory arrays with reasonable r_{off} values and small wire sizes.

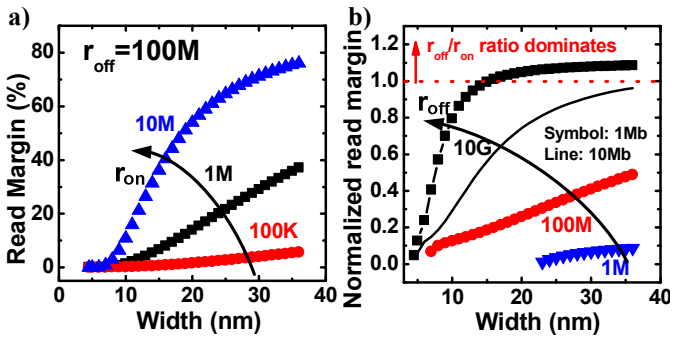

Fig. 6 (a) Read margin of a 1Mb square cross-point memory array at various r_{on} but fixed r_{off} value. (b) Normalized read margin vs. wire size for different fixed r_{off} values and array sizes. The read margin with large r_{off}/r_{on} (100×) but small r_{on} is normalized to the case with small r_{off}/r_{on} (10×) but large r_{on}. The impact of r_{off}/r_{on} is apparent only for <1Mb cross-point memory array with >10GΩ r_{off} values beyond 15nm.

B. Energy Consumption of Wires

There are two main sources of energy dissipation in wires: one is the dynamic energy which is dissipated whenever the wordlines/bitlines are charged and discharged. The other is the static energy which is consumed when the memory cell is being read or written. We will only consider the write operation since the read energy is typically much smaller. To compute the dynamic energy of wordlines/bitlines, we modeled the wire capacitance as in [12] which includes both the overlap capacitance between wordlines and bitlines and the coupling capacitance between parallel wires. The dynamic energy can then be simply calculated as $\sum_{i=1}^{m+n} C_i V_i^2$, where C_i and V_i are the total capacitance and supplied voltage for each wordline and bitline in an m×n array. For the static energy, it can be approximated as $\sum I_{rc} \cdot V_{rc} \cdot t$, where I_{rc} and V_{rc} are the current flowing through and voltage drop on each r_c. t is the time duration when the cell is powered on and is assumed to be 10ns for illustration purposes.

The wire energy of a 1Mb square cross-point memory array with size scaling is shown in Fig. 7. We can see that the static wire energy strongly depends on the memory r_{on} values, and can become substantial for smaller r_{on}. It is also a parameter that scales up with the time duration required to program a memory cell. It is worthy pointing out that the sharp decrease of static energy at extremely scaled device sizes is correlated with a significant degradation in write margin (all voltage drops on the wire), and hence we cannot utilize this region in real application. For the dynamic wire energy, it gradually decreases with the pitch size because of the reduction of the wire capacitance with down scaling, and can be as high as several pJ for wire size < 20nm. This value is greater than the per-cell energy consumption of various emerging memories that have been reported [13, 14]. Therefore, the energy consumption of wires is becoming the real bottleneck to total energy reduction in the cross-point memory array.

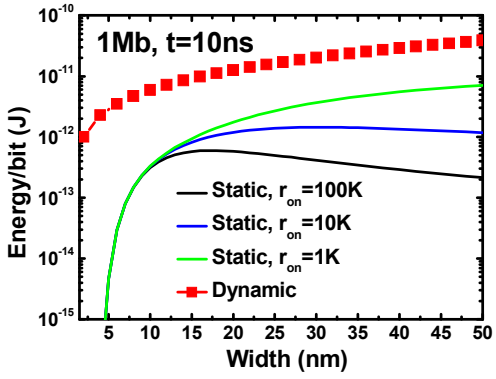

Fig. 7. Dynamic energy and static energy of wordlines/bitlines in a 1Mb square cross-point memory array. The static wire energy strongly depends on the memory r_{on} values, and can become substantial for smaller r_{on} and longer t.

C. Wire Latency and Speed

Latency is an essential performance metric of interconnect. However, due to the dramatic increase of wire resistance with

scaling, wire latency severely deteriorates and has become a primary limiter to achieving fast access of memory cells. The wire latency can be modeled as: $\tau = r_{unit} \cdot C_{unit} \cdot L^2$, in which r_{unit} and C_{unit} are the wire resistance and capacitance per unit length, respectively [15]. It can be inferred from the latency equation that the larger the memory array size and/or the smaller the wire width, the longer the wire delay. Fig. 8 compares the wire latency with the speed of various memory types for a 1Mb cross-point memory array. Here, the speed of memory is defined as the faster of the write and the read operation. We can see that the wire latency increases significantly as wire size scales and can approach 100 ns at wires with 5 nm width. The severe deterioration of wire latency obliterates the high speed advantage of various emerging memories such as RRAM (300ps) [16], STTRAM (< 2ns) [14], CBRAM (5ns) [17], and PCM (30ns) [18]and is becoming one of the most critical issue facing the future interconnect technology. Similar to the case for the write margin (Section 3A), strapping the bitlines/wordlines with wider wires will be a possible solution with a modest impact on array efficiency.

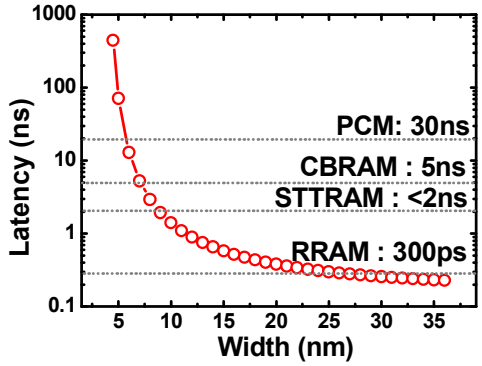

Fig. 8. Wire latency vs. wire width for a 1Mb cross-point memory array.

D. Reliability

A high current carrying capability is critical to interconnect application and reliability. For Cu wires, electromigration failure can readily occur when current density approaches 10MA/cm^2. This imposes big challenges for the scaling of cross-point memory arrays when large current density is required to program the cell. Fig. 9 illustrates the increase of current density with wire scaling for different programming currents in a 1Mb square array. We can see that, in order to keep the maximum current density below the electromigration threshold of Cu, the programming current has to be engineered below 0.1μA for wire size <10 nm. Moreover, further decrease in programming current is required for larger-scale arrays. Such low programming current, associated with an even lower read current, will lead to write/read speeds too slow to be practical. It is worthwhile mentioning that for memory cells whose programming is dependent on the current density (such as PCM and STTRAM), the feasibility of achieving the 4F^2 cross-point array architecture is determined by the difference between the minimum current density required for memory switching and the maximum current density allowed in the Cu wires. For PCM, the programming current density value can be as high as 10~40MA/cm^2 [19]. Careful engineering of the PCM material and structure is hence critical to ensure the current density below the wire electromigration limit. PCM

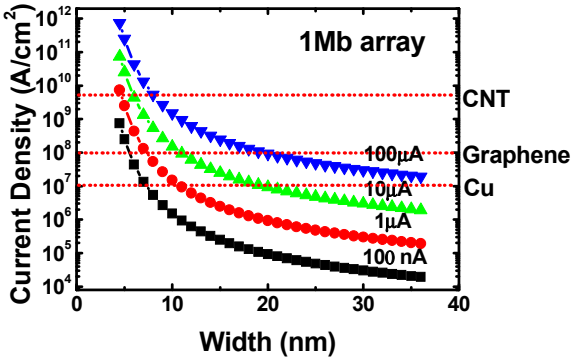

Fig. 9. Maximum wordline/bitline current density in a 1Mb square cross-point memory array as wire size scales down

cells with 2MA/cm^2 current density is recently reported in [20]. As another approach, novel interconnect materials such as graphene and carbon nanotubes (CNTs) may be an alternative solution because of the better current driving capability [21, 22] (Fig. 9). They offer orders of magnitudes lower resistance [9] (Fig. 1a) scalable to small dimensions, which facilitates the scaling of wordline/bitline toward the single-digit nm regime.

IV. Conclusion

An analysis of the size effect of Cu wire on the write and read margin, energy dissipation, speed and reliability of the cross-point memory array is quantitatively presented for wire sizes down to single-digit-nm regime. Substantial degradations in the array performance and their strong dependencies on the memory device parameters and array sizes point out the necessity to rethink the design methodology of cross-point resistive memory to mitigate the wire scaling effects. Possible solutions include the using of wires with better conductivity and scalability (e.g. graphene and CNTs), memory arrays with smaller partition sizes, strapping the bitlines/wordlines with wider wires, and memory elements with larger resistance values and ratios.

Acknowledgement

This work is supported in part by the member companies of the Stanford Non-Volatile Memory Technology Initiative (NMTRI) and FCRP C2S2. J.L. is additionally supported by the Stanford Graduate Fellowship.

Reference

[1] R. Bez and A. Pirovano, *Materials Science in Semiconductor Processing,* vol. 7, pp. 349-355, 2004. [2] S. M. Rossnagel and T. S. Kuan, *Journal of Vacuum Science & Tech. B: Microelectronics and Nanometer Structures,* vol. 22, pp. 240-247, 2004. [3] W. Steinhogl et al., *Journal of Applied Physics,* vol. 97, p. 023706, 2005. [4] K. Fuchs, *Mathematical Proceedings of the Cambridge Philosophical Society,* vol. 34, pp. 100-108, 1938. [5] A. F. Mayadas and M. Shatzkes, *Physical Review B,* vol. 1, p. 1382, 1970. [6] W. Wu, et al., *Applied Physics Letters,* vol. 84, pp. 2838-2840, 2004. [7] S. M. Rossnagel et al., *IEDM* 2005, pp. 89-91. [8] *ITRS(emerging devices chapter), 2010 Update*. [9] A. Naeemi et al, *Electron Device Letters, IEEE,* vol. 26, pp. 84-86, 2005. [10] J. Liang and H. S. P. Wong, *IITC 2010*. [11] J. Mustafa, Ph.D. thesis, Department of Electrical Engineering, RWTH Aachen University, Aachen, Germany, 2006. [12] T. Sakurai and K. Tamaru, *Electron Devices, IEEE Transactions on,* vol. 30, pp. 183-185, 1983. [13] C. H. Cheng et al., *IEDM 2010*, pp. 19.4.1-19.4.4. [14] Z. M. Zeng et al., *Applied Physics Letters,* vol. 98, pp. 072512-3, 2011. [15] J. D. Meindl, *Micro, IEEE,* vol. 23, pp. 28-35, 2003. [16] H. Y. Lee et al., *IEDM 2010* , pp. 19.7.1-19.7.4. [17] K. Aratani et al., *IEDM 2007*, pp. 783-786. [18] I. S. Kim et al., *VLSI 2010*, pp. 203-204. [19] H.-S. P. Wong et al., *Proceed. of the IEEE,* vol. 98, pp. 2201-2227, 2010. [20] J. Y. Wu et al., *IEDM 2011*. [21] B. Q. Wei et al., *Applied Physics Letters,* vol. 79, pp. 1172-1174, 2001. [22] R. Murali et. al, *Applied Physics Letters,* vol. 94, pp. 243114-3, 2009.