Nano-enhanced Architectures: Using Carbon Nanotube Interconnects in Cache Design

Banit Agrawal Navin Srivastava* Frederic T. Chong Kaustav Banerjee* Timothy Sherwood

Computer Science Department
*Department of Electrical and Computer Engineering
University of California, Santa Barbara
Santa Barbara, CA, 93106-5110
{banit,chong,sherwood}@cs.ucsb.edu {navins,kaustav}@ece.ucsb.edu

Abstract

In current deep submicron (DSM) technologies, circuit designers have to deal with some serious concerns including signal noise, leakage effect, electromigration, scattering effects, variability, and many other circuit-level issues. The problem is getting worse as we move further into the nanotechnology era. Among these, copper wires have been identified as one of the main limiting factor and error-susceptible component due to its increasing resistance and lower reliability at higher temperatures. To address this growing concern, several alternative solutions have been proposed in the form of optical interconnects and carbon nanotubes. While some of the recent works have looked into the architectural impacts of using optical interconnect, there has been no work to study the implications of using carbon nanotubes in architectural space. In this paper, we make a first attempt to present the architectural impacts of using carbon nanotubes interconnect. Specifically, we model the access time and energy consumption of cache memory when implemented with carbon nanotubes and compare with a copper wire based cache implementation. We find that carbon nanotube based cache architecture has the potential to alleviate the problem of increasing gap between processor cycle time and cache cycle time.

Keywords: *Copper, Carbon nanotubes, CNT, wire, Interconnect, cache, capacitance, resistance, access time, energy*

1 Introduction

Recently, there are many architecture-level concerns due to the increasing gap between processor cycle time and cache cycle time. Since the access time of a cache is mostly wire-dominated, increasing wire delay [20] has raised some severe concerns and it has increased the gap further between the processor and cache cycle time. As a result, there have been some recent works addressing this growing problem of wire delay in architectural domain. For example, in [13], it is shown that not considering wire delay in the multicore systems may lead to a non-optimal design solution. Also, in [2], the effect of wire delay on instruction per cycle (IPC) is shown in super-scalar architectures. Similarly, in [24], the effect of wire delay has been studied on simultaneous multi-threading (SMT) architectures. Hence, architects will need to take care of wire delay also as an important design parameter while designing caches, memories, or multicore systems.

As we look deeper into the circuit level issues, we find that as we slowly move from deep sub-micron technology to nanotechnology, the traditional design materials such as copper will not be able to keep up, and we will need to look beyond conventional materials for interconnect design. This is because of many signal noise and inductive effects, low thermal reliability, and resistivity of copper wires. Due to these deep circuit level issues, copper wires have become major limiting factor in improving the performance of the system. Constantly, wire delay is falling behind compared to the logic gate delay [9]. This problem gets worse in the case of global and intermediate wires, where wires need to cross a significant distance on the chip. Hence, there is a need for some unconventional materials that can address this problem of wire delays. Ideally, big chip industries who spend billions of dollars for each new fabrication lab, would like to have some seamless integration of new interconnection materials as a possible replacement for copper wires.

To reduce these architecture-level and circuit-level concerns, recently there have been several alternative solutions to alleviate these problems which include through-vias in 3-D chips [16, 3], optical interconnects [12, 5], and carbon nanotubes (CNTs) [21, 17]. Among the possible solutions for new interconnection materials, carbon nanotubes is found to be a promising candidate and shown to work better than the optical interconnect in some cases [5]. While there have been some works in the architectural domain to study the impact of optical interconnects, there has been no work on understanding the impacts of using carbon nanotubes in architectural design. In this paper, we focus on the architectural impacts of using carbon nanotubes. Specifically, we study the impact of carbon nanotubes in cache design and compare the results with copper wire based cache. From our findings, carbon nan-

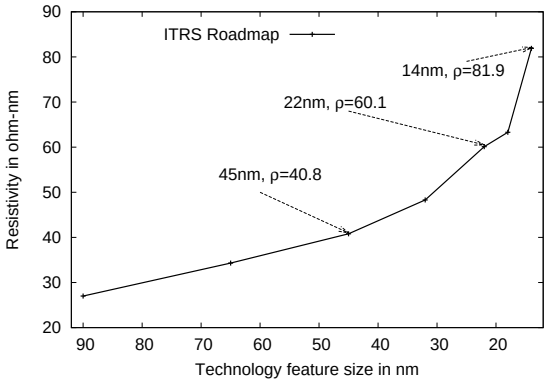

Figure 1: Resistivity increase from ITRS roadmap. There is a steep increase in resistivity as we move into 32nm and lower technology node.

otubes interconnect has the potential to reduce the problem of cache access time and narrow the gap between processor cycle time and cache access time.

The rest of the paper goes as follows. In the next section, we provide a brief background on carbon nanotubes and compare with copper interconnect for different technology nodes. In Section 3, we describe how we implement the cache using CNT interconnects. We also describe how we update the implementation of cache with new ITRS roadmap for copper wires. Then, in Section 4 we show the effects of cache size and technology on the cache access time for both CNT and copper interconnects. We also present power-performance tradeoffs while selecting the best design parameters for a particular cache design. Finally, in Section 5, we conclude.

2 Copper and Carbon Nanotube (CNT) Interconnects

In this section, we first look at the growing concerns of copper wires in detail. Then, we provide the detailed description on carbon nanotube interconnect and how it can address some of the copper wire problems. At the end of this section, we present a comparative study of copper and carbon nanotubes wires for different technology nodes.

2.1 Cu interconnect limitations

Although, in the past we have seen the replacement of aluminum wires with copper wires due to lower resistance, now copper wires are going through the similar problems due to the increasing resistivity and as a result, wire delay is becoming serious concern among circuit designers and architects. To understand the trend of increasing resistivity, we look at the ITRS roadmap [11] and some of the past works [10]. From ITRS reports [11], we find that the copper resistivity for future technologies is increasing at a very fast rate as shown in Figure 1. We find that, the increase in resistivity is not much when we move from 90nm to 32nm technology node, but as we reduce the feature size further from 32nm, we see a sharp resistivity increase. For example, increase in resistivity

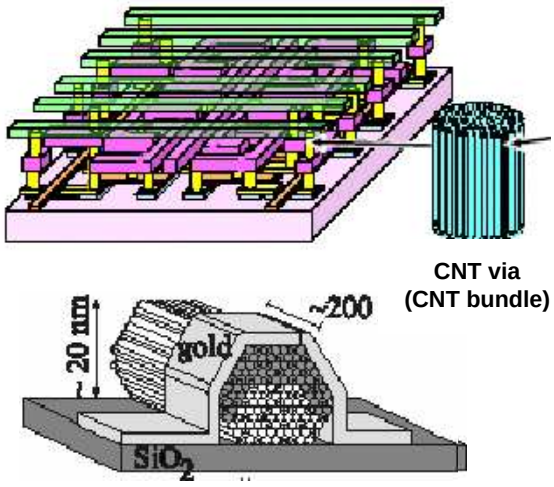

Figure 2: Different mechanism to form vias and interconnect using carbon nanotubes bundles [15][7]

is about 22% as we transision from 18nm to 14nm feature size compared to about 14% when we move from 45nm to 32nm technology node.

Besides increasing resistivity, the wire width is also shrinking with newer technologies. That further increases the overall resistance, since resistance of a wire is inversely proportional to the wire width. Therefore, even though wire length is getting smaller, but decreasing cross section area and increasing resistivity resulting in higher interconnect delay, which in turn leads to serious architectural design concerns while designing memory architecture [2] and multicore systems [13]. Next, we look at how carbon nanotubes can help to solve these emerging interconnect problems.

2.2 Carbon nanotubes (CNTs) interconnect

Recently carbon nanotubes (CNTs) are viewed as a potential replacement for copper wires due to its desirable properties such as high thermal conductivity, thermal stability, and large current carrying capacity. As a result, there have been some recent work on building CNT vias [6, 8] as also shown in Figure 2. In a separate setting, horizontal carbon nanotubes up to $18mm$ length have been grown [19]. While CNTs can address several reliability problems arising in copper wires [20], interconnect performance will also be necessary for its near or future adoption as interconnection material. To better understand the performance impact of using carbon nanotube interconnect, we first present a brief background on carbon nanotubes and its different classifications. Then, we present how some of the past works have calculated the effective resistance and capacitance of carbon nanotubes to understand the impact of CNTs on interconnect delay.

2.2.1 Characterization of CNTs

Single walled CNTs (SWCNTs) have better electron mean free paths which makes them a good choice for being considered as interconnect. However, an isolated CNT usually has

much higher resistance and there have been past works which have focused on using carbon nanotube bundles [21, 14, 4] to reduce this resistance. When carbon nanotubes are used in bundles, the large decrease in resistance leads to smaller interconnect delay. CNT bundles can be further subdivided into different classes based on the separation between the CNT tubes with in a bundle. When the bundle is densely packed, the number of nanotubes in the bundle increases which in turn effectively decreases the overall resistance [21]. We focus on the dense CNT bundles in this paper and for the rest of paper, we will refer to dense carbon nanotube bundles as CNTs for brevity for all of our performance comparison with copper.

2.2.2 CNTs Resistance

For an isolated CNT, the resistance usually comprises of mainly three components: (1) Fundamental resistance of $6.45k\Omega$ (2) Scattering resistance (3) Imperfect metal-nanotube contact resistance. For wire length less than mean free path of electrons ($1\mu m$), the resistance is independent of length. But for wire length greater than $1\mu m$, the resistance increases with length due to scattering effects. Scattering resistance is the only component which is dependent on length, while the imperfect contact resistance is independent of length. Putting it all together, the resistance of an isolated CNT wire (length $> 1\mu m$) can be written as shown in Equation 1.

$$\begin{aligned} R_{CNT} &= 6.45 * (L/L_0) + R_{contact}\ in\ k\Omega \\ where,\ L &= Length\ of\ wire \\ L_0 &= Mean\ free\ path\ of\ electrons\ in\ CNT \\ R_{contact} &= Contact\ Resistance\ in\ k\Omega \end{aligned} \quad (1)$$

While the resistance of an isolated CNT is already in $k\Omega s$ for even shorter wires, CNT bundles have been shown to reduce this effective resistance by packing more CNTs in a bundle. If we put more and more CNTs with in a bundle, the resistance goes down further. Hence, dense CNTs bundles have been shown to work better compared to sparse CNT bundles [21].

The most important difference between the resistance of copper wires and CNT interconnect is the dependence on wire length. In case of copper wires, the resistance is proportional to wire length. But with CNT wires, while one part of the resistance is directly dependent on the wire length, but the large imperfect contact resistance shadows the length-dependent part of the resistance. This is specially true for local wires, where CNT wires will not be very effective. When looking further into contact resistance, we find that there have been significant efforts recently to minimize the contact resistance. Looking at some of the past works to find the value of contact resistance, we find that there are results reporting a contact resistance of about $1k\Omega$ on the lower end [1, 25] and up to $100k\Omega$ on the higher end [23]. For our analysis, we take two conservative values of contact resistance. For upper limit, we select a value of $20k\Omega$ and for lower limit, we take a value of $5k\Omega$. While $5k\Omega$ is still higher for contact resistance, but a further decrease in contact resistance is only going to make CNTs better compared to copper.

2.2.3 CNTs Capacitance

Unlike CNTs resistance, CNTs capacity is directly dependent on length. Hence, an increase in wire length will lead to higher capacitance. While looking into many past works to find the capacitance, we find that there are different models [17, 21]. In [21], a closed form equation is presented and it has been shown that capacitance of bundle increases as we pack more number of CNTs in bundles. In [17] capacitance of dense CNT bundles is shown to be almost equal to copper wires and will not be higher than the copper wires in the worst case. Hence, we can take a conservative lower limit of CNT capacitance as capacitance of copper wires. For upper limit, we consider the results from [21].

Hence, for our analysis, we analyze two cases of CNT bundles. In the worst case (Case 1), we take the contact resistance to be $20k\Omega$ and capacitance from [21]. In the optimized case (Case 2), we take the contact resistance to be $5k\Omega$ and CNT capacitance is taken as copper wire capacitance [17].

2.3 Comparison of Cu and CNT interconnect

In this section, we present how copper and CNT interconnects will perform as we move into future technologies. For copper wire resistance and capacitance calculation, we use the closed form equations from [27]. To do a fair comparison with copper wire, we assume same wire width for both CNT and copper wires for a given technology node. All the wires widths, aspect ratios, resistivity, and dielectric constants are obtained from ITRS roadmap [11] as we further explain it in Section 3.

With the shrinking die size and feature size with newer technology, some of the wires are also becoming smaller. For example, as cache die area get smaller for a given cache size, the wire length inside caches will also become smaller. But there are still intermediate-level or global-level wires which needs to travel a significant distance. This can happen when we increase the cache size with each new technology and wire may need to travel the same length if cache is given the same real-estate budget inside the processor chip. Hence, some intermediate/global wires will be of fixed length even as we move into future technology. But most of the local/intermediate wires will scale with technology. Therefore, we present two case studies to compare copper wires with CNT interconnect. In one case, we keep the wire length fixed, and in the other we scale down the wire length as we move further into deep sub-micron technologies and nanotechnologies. In the end, we show the effect of increasing wire length for a fixed technology node. Then, we look at the combined effects of these case studies in cache design in Section 4.

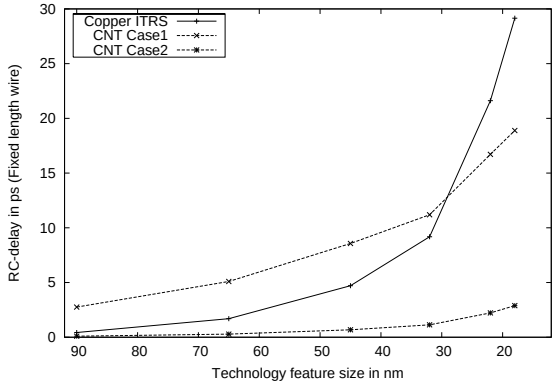

Figure 3: RC-delay comparison of Copper and two cases of Carbon nanotubes with changing technology. The wire length is fixed to 50 μm for all technology nodes.

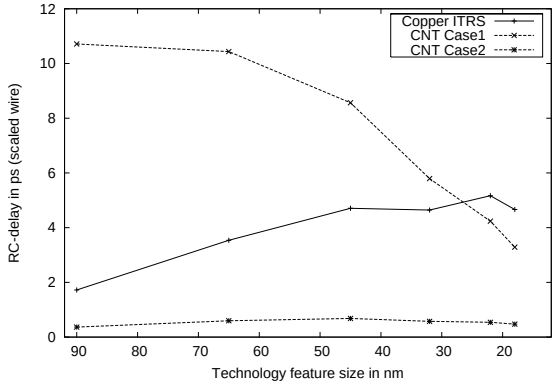

Figure 4: RC-delay comparison of Copper and two cases of Carbon nanotubes with changing technology. The wire length is 100 μm in 90nm technology node and the wire length is scaled for newer technology nodes.

2.3.1 Effects of technology on fixed length wires

We fix the wire length to 50 μm for different technology nodes ranging 90nm to 18nm. We find the effective resistance (R) and capacitance (C) of copper wires and two cases of CNT bundles. We plot the RC-delay for these three cases as shown in Figure 3. In x-axis, we vary the technology feature size from 90nm to 18nm, and y-axis shows the RC-delay in ps. We find that the performance of copper wires lies in between the two cases of CNT. But as we move into lower feature size, the gap between the copper and CNT Case1 is decreasing and at 22nm and lower technology node, copper is found to perform worse. CNT Case2 is found to do better in all technologies and perform much better (more than 5x) compared to copper and CNT Case2. Hence, CNT has the potential to serve the needs of fast interconnect in future technologies.

2.3.2 Effects of technology on scaled wires

In this scenario, we scale the wire length as the feature size decrease in current and imminent technologies. Initially, we fix the wire length to 100 μm in 90nm technology and we scale the wire accordingly. For example, in 22nm technology node, the wire length will be reduced to about 24 μm. Using different lengths for different technology, we find the resistance and capacitance of all three cases. Figure 4 shows the RC-delay plot of all three cases when we change the feature size from 90nm to 18nm. We find that the delay of copper wire is increasing even though the length of wire is decreasing. This is due to sharp increase in resistivity of copper wires. But in case of CNT Case1, the delay goes down significantly and found to do better than copper wires in 22nm and lower technology nodes. Once again, the optimized CNT Case2 performs much better compared to the two cases and found to do 7-8 times better. Hence, for scaled wires as well, CNT can effectively replace copper wires and reduce the gap between logic gate delay and wire delay.

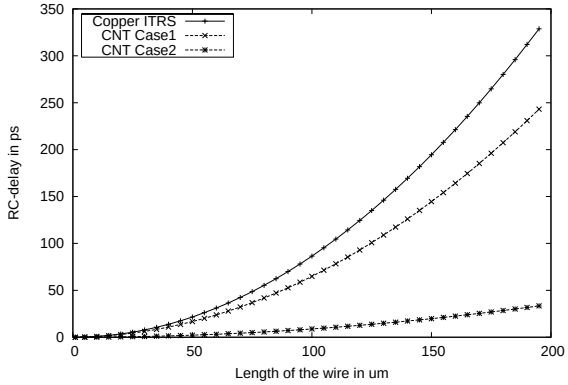

Figure 5: RC-delay comparison of Copper and two cases of Carbon nanotubes with changing wire length. The technology node is fixed to 22nm in this case.

2.3.3 Effects of increasing wire length

Next, we see the effect of increasing wire length for a fixed technology node. For this experiment, we fix the technology node at 22nm and we vary the length of the wire from $1\mu m$ to $50\mu m$. We plot the RC-delay for all the three cases in Figure 5. For all the three cases, we see increase in RC-delay with increasing wire length as expected. But in case of CNT Case2, the RC-delay is much better than copper and CNT Case1. When comparing copper with CNT case 1, we find that they are relatively close to each other. CNT Case1 is working slightly better compared to copper due to 22nm technology node as shown earlier in Figure 3. We make use of these results to understand the impact of CNTs and copper interconnect in cache design.

3 Cache Modeling

In this section, we explain how we model the cache memory using carbon nanotubes. Almost all cache designs have six-transistors 1-bit SRAM cell as building block, where one bit is stored by a pair of inverters and read/write operation

is controlled by the *wordline*. The sense amplifier is used to sense the bit stored in the cell. The length of the *wordline* and *bitline* wires are the main components, which affects delay and power design constraints. A cache is usually divided into sub-arrays to reduce the length of the *wordline* and the *bitline*, which directly reduces the delay and power. Hence, to model the cache design, we need the support of optimal cache partitioning that will give us the best power-performance tradeoffs. To do this analysis, we modify Cacti tool [18] and incorporate the carbon nanotube interconnects, which is explained next.

3.1 Cache Modeling Using Cacti

We now describe how we use Cacti tool to model CNT interconnect based cache. Cacti [18], a powerful cache modeling tool, has been widely used by architects to evaluate various on-chip cache designs. Initial version of this tool [26] was designed in 0.80 μm technology and only supported the access time of set-associative caches. Over the years, there have been several updates to Cacti and now it supports the access time, power and area overhead of set-associative caches and fully-associative caches as well. Recently, in the latest version of Cacti (4.0) [22], support for leakage power and read/write power has been added. Before this latest version, Cacti used a "fudge factor" to approximate the effect of changing technology which is an incredibly important feature for architects and has most certainly added to Cacti's longevity. But in latest version, Cacti has updated all its internal parameters for newer technology and also takes transistor sizing into account as well.

Cacti subdivides the wordline and bitline in tag array, as well as in data array to get the best cache partitioning. We need to find out the best possible combination of subdivisions in wordline and bitline for both data and tag array that gives best area overhead, access time, and energy consumption. By default, Cacti 4.0 uses an objective function to decide the best combination by giving equal weights to area, delay, and power. But, these weights can be changed if the designer is specifically interested to optimize power and performance at the expense of little larger area overhead.

3.1.1 Modeling cache with CNT interconnect

To model CNT interconnect, we need to identify all the major wiring components inside the cache structure. Once we identify all the main wires used to build the cache, we find the length of those wires. Since the resistance of CNT bundles is not directly proportional to wire length due to large contact resistance, we need to find out the length of each wire in our cache design to find the effective resistance and capacitance. Once we find the resistance and capacitance of these wires, we feed these values to Cacti. Cacti combines these results with the results of other circuit components results including decoder, sense amplifier, comparator and multiplexer. Then, we get the final access time and energy consumption of cache when implemented with carbon nanotubes. There is also one more important factor, the transistor sizing of driver transistor which drives the wire load. Since, Cacti explores the cache design space by extensively varying the divisions in bitline and wordline, we get different length wires and hence different loading effects on driving transistor. Hence, based on CNTs interconnect capacitance, the transistor width is adjusted accordingly to take into account the loading effect of CNT wires.

3.1.2 Updating Cacti for newer technologies

We find that while Cacti can provide good estimate of access time and power in current technologies, but for future technologies (45nm and lower) there may be some errors in estimation. This is mainly because there are many wire features which are assumed to be constant inside Cacti for all technology nodes. For example, Cacti assumes that wire width is 1.6 times feature-size. While this assumption holds good for some of the past and current technologies, it will not hold true for future technology where wire width is coming in line with feature size (1.0 * feature-size). Hence, we need to update the wire width for both copper and CNT interconnects in our cache analysis. Similarly, we find that the wire aspect ratio in Cacti has an assumed constant value of 1.8, which will also not hold as the wire aspect ratio is increasing further for intermediate and global wires. For example, in 22nm technology node, the aspect ratio for intermediate wires is found to be 2.0 from ITRS [11]. To do more detailed comparisons with other assumed features inside Cacti, we collect all the relevant interconnect information from ITRS for technology nodes up to 18nm [11]. We tabulate these ITRS data in Table 1. The table shows the wire pitch, wire aspect ratio, resistivity of copper wire for local/intermediate/global wires and for different technology nodes. Wire pitch is usually two times of wire width, since wire width and wire spacing are kept same. In addition to these important wire features, we also include the effective dielectric constant for copper wires which also changes with newer technology nodes. Cacti also assumes a constant value of C_{metal} which is capacitance per micrometer. But we find that C_{metal} decreases with newer technology and we update Cacti so that it can better reflect the future while comparing both copper and CNTs. Cacti also does not take care of increasing copper resistivity. Consequently, we also fix the value of resistance per micrometer parameter for each technology node. Putting it all together, we modify Cacti with ITRS data by setting the wire width, wire height for both copper and CNT wires accordingly. And for copper wires, we also fix resistance and capacitance by taking care of increasing resistivity and decreasing dielectric constant.

4 Results

Using CNT based cache implementation, we analyze various cache configurations and compare the access time and energy consumption with the traditional cache implemented with copper wires. For CNTs, we consider two cases as ex-

Table 1: Relevant ITRS wire features for different technology nodes and for different types of wires

Feature size in nm	Local wires			Intermediate wires			Global wires			Effective dieletric constant
	wire pitch in nm	aspect ratio	resistivity in ohm-nm	wire pitch in nm	aspect ratio	resistivity in ohm-nm	wire pitch in nm	aspect ratio	resistivity in ohm-nm	
90	214	1.70	29.9	275	1.70	27.0	360	2.20	24.5	3.4
65	136	1.70	34.7	140	1.80	34.3	210	2.30	27.3	2.8
45	90	1.80	40.8	90	1.80	40.8	135	2.40	31.0	2.7
32	64	1.90	48.3	64	1.90	48.3	96	2.50	35.2	2.3
22	44	2.00	60.1	44	2.00	60.1	66	2.60	42.0	2.1
18	36	2.00	63.3	36	2.00	63.3	54	2.80	45.8	1.8

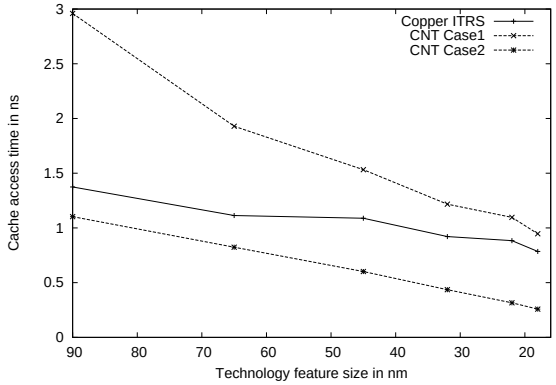

Figure 6: Cache access time for copper and CNT interconnect based cache implementation for different technology nodes

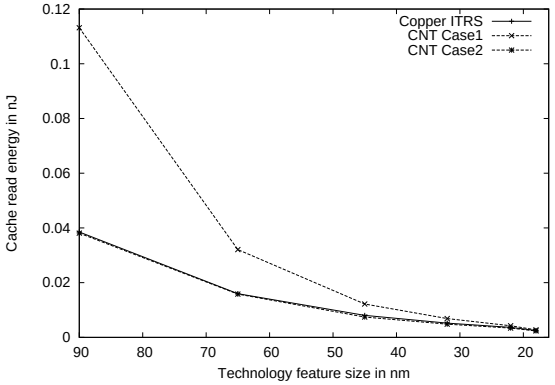

Figure 7: Cache energy for copper and CNT interconnect based cache implementation for different technology nodes

plained earlier. The worst-case of CNT wire has a contact resistance of 20 $k\Omega$ and capacitance is calculated from the results in [21] and we call this as "CNT Case1". The optimized case of CNT has a contact resistance of 5 $k\Omega$ and capacitance is conservatively assumed as equal to copper wire capacitance and we call this case as "CNT Case2". For all the comparative analysis, ITRS data has been used. Using our modified Cacti tool, we look at the technology effects on cache access time and energy consumption. We also vary the cache size for a fixed technology and find how CNTs performs compared to copper. Since, Cacti explores the design space extensively, we look at all the design options and present power-performance tradeoffs for copper and CNTs interconnect.

4.1 Technology Scaling

As the feature size is shrinking and technology is getting better, copper wires are becoming the main limiting factor. So, we look at the technology node from 90nm to 18nm to see how CNTs perform compared to coper. We select a 256KB 4-way set-associative cache with one read-write port for this analysis. Figure 6 shows the cache access time for different technology nodes. From the Figure 6, CNT Case1 is not doing better than the copper wires, but the gap is decreasing as we move towards the future technologies. We also find that CNT Case2 is doing much better compared to copper wires and the performance is getting much better as we move further into lower feature size. For example, a cache implemented with CNT Case2 will be almost four times faster than copper wires based implementation in 18nm technology node. Hence, CNTs have the potential to solve the problems with copper wires and become a possible interconnection material in near future.

In Figure 7, we show the energy consumption of optimal cache design for different technology nodes. We find that the energy consumption for CNT Case2 and copper wires is very similar because we use the same capacitance. But for CNT Case1, we find that the gap between the energy consumption of copper wires and CNT Case1 is reducing due to better decrease in capacitance.

4.2 Increasing Cache Size

In this experiment, we fix the technology node to 22nm and we vary the cache sizes from 8Kbytes to 1Mbytes. The associativity is set to four and number of read/write ports is set to one. We plot the cache access time for the optimal configuration of a fixed cache size. Figure 8 shows the cache access time in ns for different cache sizes. As we increase the cache size, the access time of the caches is increasing for all three cases, but not at the same rate. We can see that CNT Case2 again outperforms the copper wire and CNT Case1. The performance of CNT Case2 is much better compared to copper wires (more than 5-6 times) for larger cache sizes. While CNT Case1 is very close to the copper wires, it still does not good as copper wires which shows that most of wire lengths may be less than $50\mu m$ since CNT Case1 works better than copper in 22nm technology for $50\mu m$ wire length as shown earlier in Figure 3.

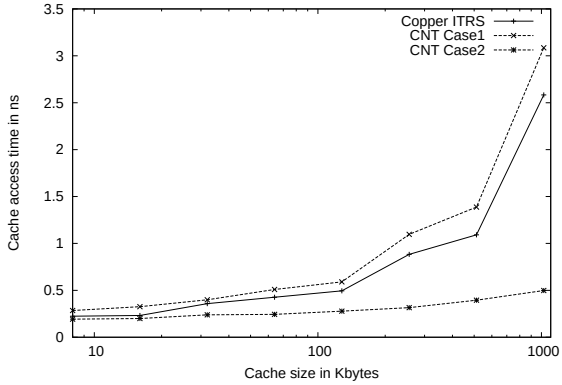

Figure 8: Cache access time for copper and CNT interconnect based cache implementation for different cache sizes

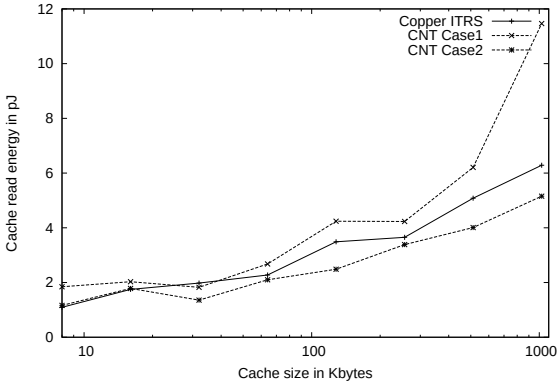

Figure 9: Cache energy for copper and CNT interconnect based cache implementation for different cache sizes

For the same optimal cache configuration, we plot the energy consumption for different cache sizes in Figure 9. While we should have expected similar behavior from CNT Case2 and copper wires, we find that it does not follow it exactly. This is because Cacti optimizes the cache configuration based on an objective function which is equally dependent on area, delay, and power. Therefore, even though the capacitance is same in case of CNT Case2 and copper wires, the access time forces Cacti to select different optimal configuration for both these cases. For CNT Case1, we find that energy consumption is slightly higher than the copper wires for smaller cache sizes, but it is much higher for 1MB cache size. This may be again due to objective function which selects the best optimal design parameters. This motivates us to look at these various design configurations that affects the power and performance which we evaluate in the following subsection.

4.3 Power-performance Tradeoffs

In this section, we study the power-performance tradeoffs of having different partitioning for a particular cache size. Once again, we fix the cache size to 256 KBytes and associativity to four. We select the 22nm technology node to evaluate the tradeoffs for using different number of splits in wordline and bitline of data and tag array. By using different number of splits in wordline and bitline, we can get an optimal partitioning of cache in terms of power and performance. We refer to these splits as $[Ndwl, Ndbl, Ntwl, Ntbl]$, where Ndwl and Ndbl are the number of splits in wordline and bitline of data array respectively. Ntwl and Ntbl represents the subdivisions in wordline and bitline of tag array.

We find the access time and energy consumption by varying these parameters for copper and two CNT cases. Each design point with different number of splits, takes a particular amount of time to access the cache and consumes a particular amount of energy. To understand these power-performance tradeoffs, we plot the access time verses energy consumption for all design points. To illustrate this, we show these design points for CNT Case2 in Figure 10. We find that there are only few pareto points which provides optimal access time with a specific energy constraint. Hence, to find these pareto points, we find the best achievable access time for a given energy budget. Based on this pareto-optimal analysis, we find all the optimal design solutions and we connect these points with pareto lines as shown in Figure 10 (shown as solid lines). Any design configuration above this line will will not be an optimal design since we can always get a better design solution on the pareto lines.

In Figure 11, we combine the results of all the three cases to do a collective comparison. We find that pareto line of CNT Case1 is always above than of copper wires. Hence, we will not be able to achieve better access time for a given energy budget in CNT Case1 compared to copper. To see the optimal partitioning of cache design with copper and CNTs interconnect, we locate the optimal design points on pareto lines and find corresponding subdivisions in wordline & bitline for data and tag array. We show these points in the figure as $(energy, delay)[Ndwl, Ndbl, Ntwl, Ntbl]$. As we look at the three optimal design configurations (one from each case), we find that $Ndbl = 32$ and $Ntbl = 32$ remains fixed for all optimal design points. There are mainly changes in the number of subdivisions in the wordline of data array. We find that the value of $Ndwl = 1$ is least in CNT Case2, while it is $Ndwl = 8$ in CNT Case1 and $Ndwl = 2$ in copper wires. Since the access time for CNT Case1 is higher, Cacti tries to get lower access time by splitting wordlines in data array and the same behavior is seen in copper wires. Hence, for this cache configuration, we find that the number of splits in the wordline of data array have an important role in designing the optimal cache configuration.

5 Conclusion

Due to increasing wire delay and less thermal reliability of copper wires, carbon nanotubes have been identified as possible interconnect for future. While there have been some studies to compare the performance of copper and CNT interconnect. But there has been no work to study the architectural impact of using carbon nanotubes as a replacement of copper. Since cache is one of most wire-dominated component inside the processor, we present an architectural analysis of using

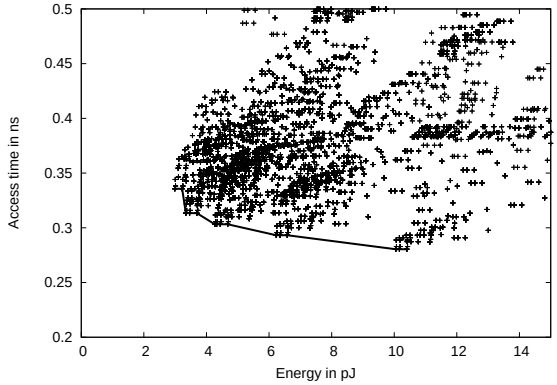

Figure 10: Power-performance tradeoffs for CNT Case2 based cache implementation. The points represent all the design points with an access time and energy consumption, whereas the solid line connects all the pareto points and represents the pareto-line.

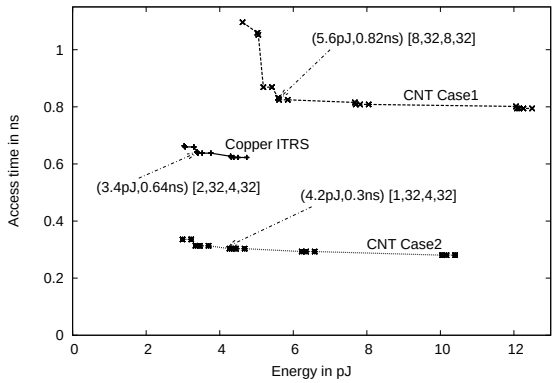

Figure 11: Power-performance tradeoffs for all the three cases. The lines shows the pareto line for three cases and any design configuration above these lines will not be optimal.

carbon nanotube interconnect in a cache design. We find all the major wiring components inside cache structure and we get the equivalent resistance and capacitance separately for each wire and feed these results to Cacti (a detailed cache model) to get the overall access time and energy consumption. When compared with copper interconnect, our findings show that access time of cache when implemented with CNT interconnect is much lesser and it has the potential to reduce the increasing gap between processor and cache cycle time in future technologies.

References

[1] A. Javey, J. Guo, M. Paulsson, Q. Wang, D. Mann, M. Lundstrom, and H. Dai. High-field quasiballistic transport in short carbon nanotubes. *Physical Review Letters*, 92(10), 2004.

[2] V. Agarwal, M. S. Hrishikesh, S. W. Keckler, and D. Burger. Clock rate versus ipc: the end of the road for conventional microarchitectures. In *ISCA '00: Proceedings of the 27th Annual International Symposium on Computer Architecture*, pages 248–259. ACM Press, 2000.

[3] K. Banerjee, S. J. Souri, P. Kapur, and K. C. Saraswat. 3-D ICs: A Novel Chip Design for Improving Deep Submicron Interconnect Performance and Systems-on-Chip Integration. *Proceedings of the IEEE*, 89(5):602–633, May 2001.

[4] K. Banerjee and N. Srivastava. Are carbon nanotubes the future of vlsi interconnections? In *DAC '06: Proceedings of the 43rd annual conference on Design automation*, pages 809–814. ACM Press, 2006.

[5] H. Cho, K.-H. Koo, P. Kapur, and K. C. Saraswat. Modeling of the performance of carbon nanotube bundle, cu/low-k and optical on-chip global interconnects. In *SLIP '07: Proceedings of the 2007 international workshop on System level interconnect prediction*, pages 81–88. ACM Press, 2007.

[6] F. Kreupl et al. Carbon nanotubes in interconnect applications. *Microelectronic Engineering*, 64:399–408, 2002.

[7] H. Stahl et al. Intertube coupling in ropes of single-wall carbon nanotubes. *Physical Review Letters*, 85(24):5186–5189, 2000.

[8] W. Hoenlein, F.Kreupl, G. Duesberg, A. Graham, M. Liebau, R. Seidel, and E. Unger. Integration of carbon nanotubes devices into microelectronics. *Mat. Res. Soc. Symp.*, 772, 2003.

[9] M. Horowitz, R. Ho, and K. Mai. The future of wires, 1999.

[10] S. Im, N. Srivastava, K. Banerjee, and K. E. Goodson. Scaling analysis of multilevel interconnect temperatures for high performance ics. *Proceedings of the IEEE Transactions on Electron Devices*, 52(12):2710–2719, December 2005.

[11] Interconnect Report. International Technology Roadmap for Semiconductors (ITRS), 2006.

[12] N. Kirman, M. Kirman, R. K. Dokania, J. F. Martinez, A. B. Apsel, M. A. Watkins, and D. H. Albonesi. Leveraging optical technology in future bus-based chip multiprocessors. In *MICRO 39: Proceedings of the 39th Annual IEEE/ACM International Symposium on Microarchitecture*, pages 492–503. IEEE Computer Society, 2006.

[13] R. Kumar, V. Zyuban, and D. M. Tullsen. Interconnections in multi-core architectures: Understanding mechanisms, overheads and scaling. In *ISCA '05: Proceedings of the 32nd Annual International Symposium on Computer Architecture*, pages 408–419. IEEE Computer Society, 2005.

[14] J. Li, Q. Ye, A. Cassell, H. T. Ng, R. Stevens, J. Han, and M. Meyyappan. Bottom-up approach for carbon nanotube interconnects. *Applied Physics Letters*, 82(15):2491–2493, 2003.

[15] M. Nihei et al. Carbon Nanotube Vias for future LSI Interconnects. In *IEEE Intl Interconnect Tech Conference*, pages 251–253, 2004.

[16] S. Mysore, B. Agrawal, N. Srivastava, S.-C. Lin, K. Banerjee, and T. Sherwood. Introspective 3D chips. In *ASPLOS-XII: Proceedings of the 12th international conference on Architectural support for programming languages and operating systems*, pages 264–273. ACM Press, 2006.

[17] A. Naeemi and J. D. Meindl. Design and performance modeling for single-walled carbon nanotubes as local, semiglobal and global interconnects in gigascale integrated systems. *Proceedings of the IEEE Transactions on Electron Devices*, 54(1):26–37, January 2007.

[18] P. Shivakumar and N. P. Jouppi. Cacti 3.0: An Integrated Cache Timing, Power and Area Model. Technical Report Western Research Lab (WRL) Research Report, 2001/2.

[19] Smart Materials Nanotechnology Laboratory. UC Researchers Shatter World Records with Length of Carbon Nanotube Arrays, april 2007.

[20] N. Srivastava and K. Banerjee. A comparative scaling analysis of metallic and carbon nanotube interconnections for nanometer scale VLSI technologies. In *VMIC*, pages 393–398, 2004.

[21] N. Srivastava and K. Banerjee. Performance analysis of carbon nanotube interconnects for vlsi applications. In *ICCAD '05: Proceedings of the 2005 IEEE/ACM International conference on Computer-aided design*, pages 383–390. IEEE Computer Society, 2005.

[22] D. Tarjan, S. Thoziyoor, and N. P. Jouppi. Cacti 4.0. Technical Report Western Research Lab (WRL) Research Report, 2006/06.

[23] Th. Hunger et al. Transport in ropes of carbon nanotubes: Contact barriers and luttinger liquid theory. *Physical Review B*, 69, 2004.

[24] T. N. Vijaykumar and Z. Chishti. Wire delay is not a problem for smt (in the near future). In *ISCA '04: Proceedings of the 31st Annual International Symposium on Computer Architecture*, pages 40–51. ACM Press, 2004.

[25] W. Liang et al. Fabry-perot interference in a nanotube electron waveguide. *Nature*, 411:665–669, 2001.

[26] S. Wilton and N. Jouppi. An enhanced access and cycle time model for on-chip caches. Technical Report 93/5, DEC Western Research Lab, 1994.

[27] S.-C. Wong, T. G.-Y. Lee, D.-J. Ma, and C.-J. Chao. An empirical three-dimensional crossover capacitance model formultilevel interconnect vlsi circuits. *IEEE Transactions on Semiconductor Manufacturing*, 13(2):219–227, 2000.