WORKSHOP TALKS

The following table lists the talks presented at the workshop. They fell into three categories:

- “Lessons Learned:” Design lessons to be drawn from silicon
- “Silicon’s Cousins:” Non-silicon technologies with molecular capabilities that look enough like silicon from the design standpoint that many lessons can be ported over.
- “Alien technologies:” non-silicon technologies that will probably require significant developments in design methodologies to bring into practice.

There were 5 talks that fell primarily in the first category, 8 in the second (title in italic), and 4 in the third (name in italic).

<table>
<thead>
<tr>
<th>Name</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kamal Abdali</td>
<td>NSF Initiatives in Molecular Systems</td>
</tr>
<tr>
<td>Lynn Conway</td>
<td>Silicon - the early years</td>
</tr>
<tr>
<td>Mary Jane Irwin</td>
<td>Silicon’s Roadmap: Why the Roadblock</td>
</tr>
<tr>
<td>Michael Fuhrer</td>
<td>Nanoelectronic devices using carbon nanotubes</td>
</tr>
<tr>
<td>Konstantin Likharev</td>
<td>Self-Evolving Networks from Molecular Single-Electron Nano-Switches</td>
</tr>
<tr>
<td>Wolfgang Porod</td>
<td>Circuit-level approaches for field-coupled nano structures</td>
</tr>
<tr>
<td>Seth Goldstein</td>
<td>NanoFabrics</td>
</tr>
<tr>
<td>Paul Weiss</td>
<td>Molecular electronics self-assembly</td>
</tr>
<tr>
<td>Fred Chong</td>
<td>Programming &amp; designing nanoscale devices for useful aggregate behavior</td>
</tr>
<tr>
<td>Thomas Sterling</td>
<td>HTMT &amp; Continuum computing: interaction of technology &amp; architecture</td>
</tr>
<tr>
<td>Thomas Sterling</td>
<td>HTMT: An exercise in developing technology and architecture together</td>
</tr>
<tr>
<td>Glenn Martin</td>
<td>Recent progress on molecular electronics at Yale</td>
</tr>
<tr>
<td>Andre Dehon</td>
<td>Regular architecture for molecular scale arrays</td>
</tr>
<tr>
<td>Jacob Abraham</td>
<td>Dealing with complexity in test &amp; verification</td>
</tr>
<tr>
<td>Chia-Hung Yang</td>
<td>Building a circuit within the elastic mean free path &amp; coherence length</td>
</tr>
<tr>
<td>Craig Lent</td>
<td>Molecular QCA</td>
</tr>
<tr>
<td>Dale Edwards</td>
<td>Design Challenges facing SRC ICSS Science Area</td>
</tr>
<tr>
<td>Peter Kogge</td>
<td>Designing QCA Systems</td>
</tr>
</tbody>
</table>

The rest of this report is divided into three sections: a summary of lessons learned from silicon, a snapshot of some emerging nanotechnologies (both “cousins” and “aliens”) and some summary comments about what will be needed to adapt our design processes to these technologies.

SILICON’S LESSONS

The lessons that were drawn from our experience with silicon were two-fold in direction: what was it that made the initial technology so viable, and what allowed follow-on variants of these silicon technologies to not only survive development but be deployable so quickly. Three of our members (Conway, Irwin, and Abraham) had some particularly telling comments on these topics.

Prof. Lynn Conway of Michigan led off with a discussion on “The Revolution” that made silicon, especially digital CMOS, so ubiquitous. Her comments started with an observation about the successful computer architects of the last few decades. For the most part they don’t care about physics, only on the rules for combination, and the constraints that are imposed on such combinations (power, area, contacts). They tend to look at designs in layers or hierarchies, with each layer designed as some combinations of subsystems from the prior layer. The most successful architects tend to work with others associated with the rest of the chip design process, namely the circuit
designers, CAD, packaging, layout, fab, and test engineering. Finally, they tend to translate how they deal with new technologies into different forms of old ones (the concept of abstraction).

Prof. Conway then commented about the state of affairs before MOSFET, namely when RTL (Resistor Transistor Logic) was the mainstream technology circa 1977. The design flow started with system architects, who developed high level designs and passed them to logic designers. They in turn worked with the boolean equations describing the logic to convert them into combinations of circuits from some library. From such circuit diagrams the fabrication engineers then designed copies of the basic circuits to be placed on a chip, and “wired them up” much as had been done with printed circuit cards and small scale logic. There was very little interaction between any of these groups, especially between the system designers and the fab engineers. Nearly universally, devices were looked at in “cross section,” with a focus on how the device worked. There was very little attention given to the “overhead view” (“it’s just wire”). Such a cross-section focus was clearly not of interest or value to the systems designer, and even most of the logic designers, because very little of that could be influenced by the system design itself. MOSFETs were dismissed as largely irrelevant because they were “slow and sloppy.”

The telling comment was made that this focus on cross-section, not overhead, is exactly what we see today when discussing molecular or nano technologies.

Finally Prof. Conway went through the keys to the MOSFET revolution, for which she and Carver Mead in no small way were responsible. First was the development of the self-aligned gate process that enabled a greatly simplified fabrication process, with just a small series of steps needed to produce functioning circuits at reasonable densities. Next was the realization that scaling of the basic gate down to something on the order of 0.25 micron was feasible, and that with a careful set of groundrules, designs could be done in a largely technology independent fashion. This allowed for long-lived designs and, more importantly, design processes. Together these realizations energized the academic community, and permitted the development of some simple but useful circuit primitives. More importantly, it changed the orientation from the cross-section of the device to a focus on interconnect. The early work in the area was considered toy-like, and unoptimizable, UNTIL the advent of MOSIS enabled inexpensive, adventuresome prototyping. This immediately allowed academic researchers to design, develop, and demonstrate real systems at complexity levels similar to, and sometimes above, that of commercial practice. It also provided a pool of well-educated engineers who were familiar with such design rules and how to deal with the complexity of real systems built from them.

Prof. Jacob Abraham of the Univ. of Texas spoke on the advances in testing and verification that allowed the revolution to continue to grow in scale and complexity. He made several observations. First, if you consider systems with just a puny 200 bits of states, there are more state values than there are protons in the universe. This makes the process of exhaustively testing of even trivial systems totally infeasible. Following up on this, it appears that many key design questions are similarly NP-complete: critical paths, verification, fault coverage, etc. This leads inescapably to two major “lessons learned” from CMOS CAD

- Direct attack on “flat designs” (where there is little structure to the design, and everything is visible at the same level) rapidly becomes impossible
- Hierarchical designs that abstract functionality into layers are much more manageable

To illustrate this, he shared some data, shown below, as to the time to generate test vectors of various designs as a function of the test coverage achieved. The results show a one to three order
of magnitude decrease in test generation time, and an increase in fault coverage, for the more hierarchical designs.

The final set of discussions on silicon were from Prof. Mary Jane Irwin of Pennsylvania State University, and addressed the coming roadblocks that may spell the end of the line for CMOS as we know it, and require a transition to nano-technologies for further advances. Her glimpse at the end of the line included 35 nm gate widths, 4.4 GHz clocks, and 354 sq. mm die. Her major concerns were that devices “won’t work” and there will be significant changes in key ratios crucial to the design process. In particular she identified as problems:

1. Slow Wires: “Long wires” don’t scale as everything else does in CMOS, & more wires in designs are becoming longer. This will place upper limits on system frequencies, especially those that traverse significant fractions of a chip.

2. Chip I/O: the number of signal pins available from a die are growing much less quickly than “Rent’s Rule” (the number of external signals needed for an average block of logic grows as the 0.6 power of the number of gates). This means that data will increasingly become “trapped” on individual chips, with an inability to move it on and off at a rate commensurate with internal processing speeds.

3. Hot Chips: Both power density & die area are increasing, to the point where power densities will approach that of nuclear reactors. In an era where relatively fixed budgets for cooling systems are in place, this constrains considerably our ability to fully utilize the potential of a single chip.

4. Reliability: single defects wipe out more & more functionality, requiring more and more attention to redundancy and other features that consume space without adding functionality.

5. Design Costs: new issues of clock, interconnect, power, IP reuse, and “Systems On a Chip” verification are beginning to increase in importance, and threaten to overwhelm the gains due to hierarchical design.
6. Fab Costs: computationally complex masks, expensive lithography will make future “top down” fabrication facilities extraordinarily expensive, thus reducing our ability to experiment with new designs.

**Nanotechnology Snapshots**

Much of the discussion focused on a variety of new technologies.

Paul Weiss of Penn State discussed the limits of circuits in terms of the mean free paths and coherence lengths of electrons. He theorized that the ultimate computing device would thus probably be realized by a trapped single electron in a quantum dot. He also discussed our growing capability to manipulate the properties of single molecules, especially in monolayer films with sharp boundaries, where molecular alignment and conductance states are heavily influenced by neighboring molecules. Such capabilities may provide the basis for “self-assembly” at the molecular level.

Several attendees (Michael Fuhrer of the University of Maryland, Seth Goldstein of CMU, and Andre Dehon of Cal Tech) all addressed the potentials of nano-tubes and nano-wires. Nano-tubes here were meant to represent nanometer wide metallic or semiconducting molecular cylinders that are now producible in number in the lab. Applications for such devices included interconnect (wire), single electron FETs, micro mechanical switches, levers, relays, and other non-traditional (at the micro level) devices. The most promising circuit structures that may grow out of such devices include arrays, crossbars, FPGAs, and “fabrics.” Such structures, especially ones spread out over very large 2D fabrics, may induce a “compile to space” rather than a “compile to time” philosophy, where the functionality of a program is converted to the interconnection of multiple regions of such fabrics, not to sequences of repeatable instructions. The challenges foreseen for the technologies include alignment, defect recovery and repair, fault tolerance, interfacing between the micro and nano levels, gains and signal restoration, and customization.

Craig Lent, Wolfgang Porod, and Peter Kogge, all of Notre Dame, addressed the potentials of Quantum dot Cellular Automata (QCA), where charge configurations of electrons within confined sets of quantum dots is used to represent information, rather than the use of current flow in silicon systems. Real devices have been built that demonstrate gain, logic, and storage, with a path developing towards molecular implementations. Preliminary estimates based on attempts to design with such devices are that at the molecular level orders of magnitude improvement in density, speed, and power over silicon appear possible. One of the key features of this project has been the early involvement of architects and designers with the device technology people. The result has been a very fruitful exploration of many “real world” issues that otherwise might have been missed in a technology demonstration only program. Future needs foreseen to make QCA viable include demonstrations of larger circuits, a deeper understanding of clocking, defects, and interfaces to the silicon world, the development of Mead/Conway style groundrules, and a prototype fab facility.

**Design Challenges of Nano Technologies**

A variety of attendees addressed explicitly the interplay between technology, especially potential nanotechnologies, and architectures.

Glenn Martin of Yale discussed the state of research in molecular electronics at Yale, and made an impassioned claim that “Silicon ain’t dead yet,” that despite all of nano’s supposed
progress that there is nothing on the horizon that will challenge silicon in fundamental ways. Nano needs to find a niche where it can do things simply not duplicatable by silicon.

A talk by Thomas Sterling of Cal Tech, while it didn’t address nano technologies per se, was an excellent counterpoint to this comment. Dr. Sterling discussed the HTMT project: a multi-year, multi-institutional project to design a petaflops scale computer that would represent a level of computing beyond the cost effectiveness of silicon. The project was an example of attempting to merge multiple new technologies together into a single system design. These technologies ranged from superconducting logic, “latchless” optical networking, processing-in-memory, and holographic storage. A system architecture, and matching execution model, was chosen where each technology could shine in a particular functional subsystem. This was highly successful from the design perspective because it allowed the interfaces between subsystems to become the drivers for the technologies within those subsystems. Similar hierarchical specialization and isolation will probably benefit insertion of future nanotechnologies into real systems.

Three different attendees (Konstantin Likarev of Stonybrook, Wolfgang Porod of Notre Dame, and Fred Chong of UC Davis) addressed the potential of using “neural net”-like architectures as drivers for advances in nano technologies. This architecture was selected because of its natural fault-tolerance, and small-scale (in terms of complexity) building blocks. It also represents one where silicon equivalents simply cannot grow big enough (the human brain has $10^{10}$ basic cells and $10^{14}$ interconnects). Some of the enabling nano technologies may start with single electron transistors (SET) with randomized analog crossbar networks. Growth to complete systems thus might start with a “silicon sandwich” of CMOS interconnect logic, SET processing elements, and “sensors” also constructed from nano technologies. Arrays of such “sensor nets” might then be merged into their environment rather than simply observe it, with programming of smaller nodes for aggregate behavior in the large.

Finally, Dale Edwards of SRC contributed some thoughts from the commercialization perspective of the design challenges of moving into the molecular realm:

- understanding the new failure mechanisms is key to building accurate models of reliability,
- “software reconfigurability” in the field will become more and more important to such systems to allow for both design updates and application to new applications,
- controlling the design time as complexity grows is essential,
- speed of systems: can we maintain Moore’s Law in some form in the new technologies, or are we at yet other sets of roadblocks,
- mixed signal systems will become essential as we move to “systems on a chip” that employ both traditional and nano technologies,
- test will be crucial, both in coverage AND in procedures AND test fixtures (how do you “test” a nano scale chip)
- cost in all its dimensions: development costs, unit costs, and where do the different components of such costs come from.

Summary Comments

As a result of this workshop, a series of overall observations emerged. First, in terms of new insights that need to be developed by the nano community, the following comments are relevant:

- Deeper understanding and modeling of interactions between nearby molecules need to be developed, with an emphasis on abstraction into simple “engineering” models that can be used for at least preliminary design feasibility studies.
A clearer picture needs to be drawn of what exactly is “electronic nanotechnology” and how does it discriminate itself from “classical” micro technology.

There is a dearth of “killer applications” and/or new computing models. Even this workshop, for the most part, ended up discussing neat structures, not applications requiring or being enabled by such structures.

We still need to view systems built from nanotechnology in terms of hierarchies of abstractions, with “interfaces” to conventional environments. The closest to this within the workshop were discussions on two-level FPGA-like arrays.

A focus on reliability and fault models needs to go hand in hand with the device developments.

In terms of making any particular nanotechnology ready for prime time use, three things appeared necessary. First, teams that cover the gamut from device physicists, to architects, including CAD, test, and manufacturing expertise are needed. The device physicists need to develop simplified technology descriptions that are “teachable” to any competent design engineer relatively quickly. This should include description of the “key parameters” that affect circuit level performance, and simple circuits that demonstrate the interactions of the parameters. The architects need to use the above to develop extendable abstractions for hierarchical designs that leverage the technology’s attributes. The CAD engineers need to leverage existing hierarchal design systems wherever possible, taking maximum advantage of the billions invested in algorithms and design tools. Finally, the physical designers need to develop libraries of simple interfaces to conventional technologies and the environment.

Second, any package of tools that are developed in support of a technology, ought be “design-oriented.” This includes:

- engineering oriented groundrules that can project or scale forward into future generations of the technology in question,
- they can be taught quickly to skilled conventional designers.
- they include simple models of speed, power, and area.
- they permit hierarchal designs, with primitive composition rules, reasonable I/O, and leverage existing test and validation technologies,
- a basic tool set be readily available,
- and resulting designs be matched with the capabilities of readily available prototyping services.

Third is the need for a MOSIS-like readily available prototyping service, where experimental designs can be fabricated and evaluated. This facility needs to be open to a wide group of potential users. It needs to permit a spectrum of activities, including student design exercises, new circuit experiments, and new architectural prototypes. The end goal should be to allow designs to rapidly reach the “proof of concept” stage.

In conclusion, the workshop demonstrated that the future is bright for systems designed at the molecular level, especially if we can leverage the enormous store of design experience we have gathered from the silicon world.