SANDIA REPORT
SAND201X-XXXX
Unlimited Release
Printed Month and Year

Post Moore's Law Report

Authors
Erik P. DeBenedictis, Sandia
Anthony Lentine, Sandia
Matthew Marinella, Sandia
R. Stanley Williams, HPE
Thomas M. Conte, Georgia Tech
Paolo Gargini, IRDS

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by National Technology and Engineering Solutions of Sandia, LLC.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from

U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@osti.gov
Online ordering: http://www.osti.gov/scitech

Available to the public from

U.S. Department of Commerce
National Technical Information Service
5301 Shawnee Rd
Alexandria, VA 22312

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.gov
Online order: http://www.ntis.gov/search

SAND201X-XXXX
Printed May 2018
Unlimited Release

Post Moore's Law Report

Erik P. DeBenedictis, Sandia
Anthony Lentine, Sandia
Matthew Marinella, Sandia

R. Stanley Williams, HPE
Thomas M. Conte, Georgia Tech
Paolo Gargini, IRDS

Sandia National Laboratories
P. O. Box 5800
Albuquerque, New Mexico 87185-MSXXXX

Abstract

Moore's law is driving an information revolution, worldwide economic growth, and is a tool for national security. This report explains how dire proclamations that "Moore's law is ending" are due to a natural redefinition of the phrase, but computing remains positioned to both drive economic growth and support national security.

The computer industry used to be led by the semiconductor companies that made ever faster microprocessors with larger memories. However, control is shifting to new ways of designing computers, notably based on 3D chips and new analog and digital architectures. While artificial intelligence and quantum computing research have become mainstream pursuits, these latter two areas seem destined split off from Moore's law rather than become a part of it.

We include a discussion of recent developments and opportunities in optical communications and computing.

Table of Contents

Executive Summary ... 6
- Background on Moore's law ... 7
- A change was coming ... 8
- Why the quest failed ... 8
- Government and industry responses ... 9
- Moore's law continues anyway ... 10
- Organizing computing approaches by physical limits ... 10
  - Artificial intelligence ... 10
  - Quantum computing ... 11
  - Approaches limited by the thermodynamics of computation ... 11
  - Reversible computing ... 12

Device physics research for computing ... 13

Turning 3D into a scaling resource ... 14
- Path #1 z-dimension layers and #2 tightness of interface ... 14
- Path #3 logic layer scaling and heat removal ... 15
- Path #4 algorithms ... 15
- Path #5 Programming model ... 16
- Path #6 diversification of devices ... 16

Photonics for Beyond Moore's Law Computing ... 17

Conclusions ... 18

Deep dive I: Combining device physics and computation ... 19
- The need to consider devices in context ... 19
- Generalizing solution of the traveling salesman problem ... 20
- Energy minimization as physics-based programming primitive ... 21
- Hard computing problems and using noise as a resource ... 22
- An architecture based on an energy minimization subsystem ... 23
- Conclusions ... 23
- Recommendations ... 24

Deep dive II: Future scaling of Sandia applications ... 25
- Basic changes due to 3D chips (#1, #2) ... 25
- The general connection between 3D and algorithms ... 28
- Matching algorithm scaling to the new improvement paths (#4) ... 29
- Programming model (#5) ... 30
- Where we are today ... 31
- Recommendations ... 31

Deep dive III: CMOS Optical Interconnect ... 33
- On-roadmap solutions ... 33
- Off-roadmap solutions ... 34
- Optical transceivers directly integrated with high-value electronics ... 35
- Optical media: fibers ... 36
- Optical printed circuit boards ... 38
- Free space optical interconnects ... 39
- On-chip interconnects: an opinion ... 39
- Optical interconnects for BMC transistors ... 39

Electrically controlled optical switching 40
Optical packet switching 41
Optical Computing 42
Digital logic 42
Vector-matrix multiplication and neural-inspired computing 42
Discussion and recommendations: optics 43
References 45

Executive Summary

While some people say "Moore's law is ending," table 1 shows the world's five largest companies are all based on the fruits of Moore's law, up from just one of the five when we wrote the previous version of this report in 2008. This report attempts to reconcile the purported end of Moore's law with resulting success.

Table 1: Top five companies worldwide by market cap, IT sector companies in red

Dec. 31, 2008 (last report or study)	June 30, 2017 (date of this writing)
Exxon	Apple
PetroChina	Google
Wal Mart	Microsoft
China Mobile	Amazon.com
Procter and Gamble	Facebook

When we wrote the previous version of this report, society believed Moore's law made microprocessors exponentially faster and their memory exponentially larger from one generation to the next. These two factors together made computers improve over time. The increasing capabilities of computer hardware over time enabled new applications, and the applications drove an information revolution and changed the world. Great story, but it couldn't go on forever.

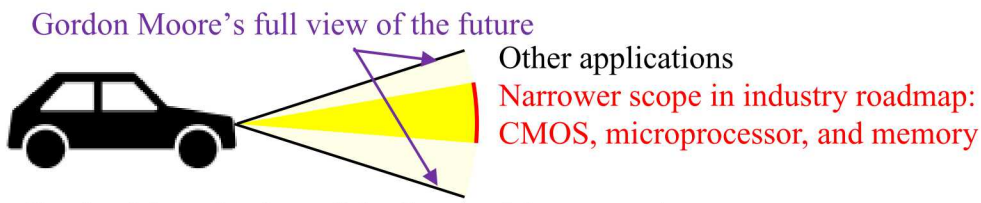
The exponential improvement rate slowed in the nine years since the previous report, but we believe progress will continue with the new set of drivers in table 2, although industry is in a transition period right now and it may take some time for the drivers to become clear. However, it looks like progress in artificial intelligence and quantum computing will not be seen as part of Moore's law because of society's choice of word definitions.

Table 2: Continuation of progress in computing, including Moore's law

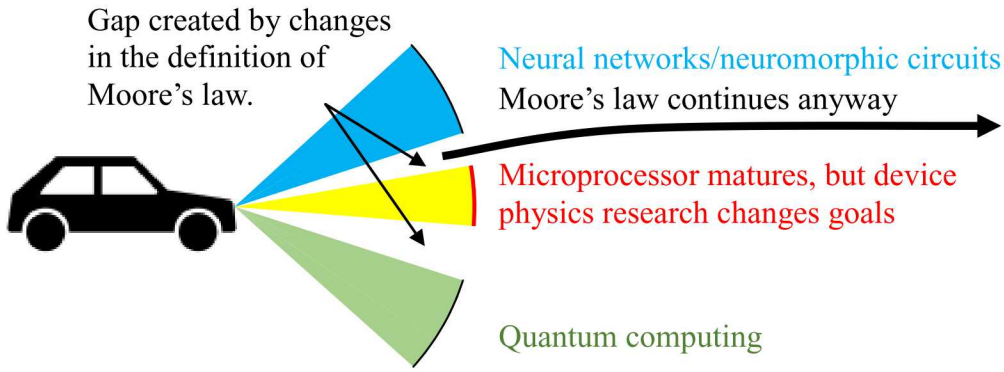
Artificial intelligence	Today's deep learning, neuromorphic computing, and other approaches will develop into a new computer type that improves via "smartness" instead of traditional metrics.
Quantum computing	A new type of computer will emerge that uses quantum information principles to solve a limited number of problems e^N times faster than a computer based on standard Boolean logic, where N is the number of (logical) qubits. If the number of (logical) qubits grows linearly with time, quantum computers will speed up exponentially over time.
New device physics strategy	Device physics research will continue, but its purpose will broaden from just making devices for use in Boolean logic gates and memory. New devices will support many circuits and architectures.

Background on Moore's law

Moore's law is universally credited to a 1965 article by Gordon Moore,[1] an article that did not mention Moore's law or any other law. The article was part of an Electronics Magazine column called "the experts look ahead" and projected that integrated circuits would lead to many computer-based consumer applications. Figure 1a illustrates the scope of ideas as Gordon Moore in his car looking ahead, able to see the area illuminated by the headlights.



a. Gordon Moore's view of the future of the computing



b. View of the future from Moore's law + Post-Moore's law era

Figure 1: Scope of Moore's law and Post-Moore's law era based on Gordon Moore looking ahead, which is the literal origin of the "law."

About a decade later, the term "Moore's law" was attached to the third figure in Moore's article, which projected an exponential rise in the number of components per integrated structure (circuit) from 1959-1975. Since Moore's law refers only to a subset of the ideas in Moore's article, people could create their own definitions of Moore's law using Moore's third figure plus other ideas in the article of their choosing.

The increase in the number of components followed the third figure in Moore's article through 1975 and continues to this day. However, industry exploited the changeable definition of Moore's law to narrow its definition to just the key products of the era, the microprocessor and memory, as opposed to any chip that could contribute to the application-level vision in Moore's paper. For example, the International Semiconductor Roadmap for Semiconductors (ITRS)[2] roadmapped microprocessors and memory, but not FPGAs, signal processors, or computers in general. This is shown in figure 1a as the narrowing scope of the headlights looking into the future.

A change was coming

Microprocessor systems improved exponentially from the 1990s through the early 2000s, but industry was forced to change to multi-core processors around 2003. Without multiple cores, microprocessor power dissipation would have reached and exceeded 200 W per chip, which was above what customers were willing to accept. Customers ultimately accepted multiple cores at a lower clock rate, where the lower clock rate raised energy efficiency and brought power dissipation into an acceptable range. This was a weak solution because most software only runs on one core, in which case an entire multi-core microprocessor runs only as fast as one core.

To illustrate one of industry's attempts to remedy the situation, the ITRS met in April 2008 in Koenigswinter, Germany. At this meeting, the semiconductor industry's roadmap group and arguably the industry's leadership, made an, ultimately unsuccessful, decision to transition the semiconductor industry to a new technology called "Beyond CMOS." The following quote is from the meeting minutes, with an acronym key below:[3]

> *"The IRC has requested ERD/ERM to begin to narrow options for 'Beyond CMOS' technologies. Of the various options for new Beyond CMOS Information processing technologies (including various charge based – SETs, QCA, RTD, etc. - , molecular, spintronics, nanomechanical, etc.) we are asked to:*
>
> - *"Recommend one of the major classes as being most promising by no later than Dec. 31, 2008*
> - *"Identify one or two devices or approaches within the recommended class to pursue with a detailed roadmap with a time line. We will define a process for accomplishing this task by arriving (hopefully) at a consensus with ERD…"*
>
> Legend: IRC=International Roadmap Committee; ERD=Emerging Research Devices; ERM=Emerging Research Materials; SETs=Single electron transistors; QCA=Quantum cellular automata; RTD=Resonant tunneling diodes.

Why the quest failed

The technology change did not occur because there was no suitable Beyond CMOS device, but the technical reason was not immediately clear. Five years later, a group led by Nanotechnology Research Institute (NRI) and Intel published a comprehensive study of performance of the Beyond CMOS technology options as of that date.[4] The study analyzed many proposed devices in an effort to find the one with the best speed and energy efficiency. The results in Figure 1[4] show a "Pareto frontier" where leading options approach a line from one side, or frontier, that trades off speed for energy efficiency. None of the options cross the frontier and CMOS is close enough to the frontier that it would be difficult to recoup the billions of dollars needed to develop an alternative technology from such a small advantage.

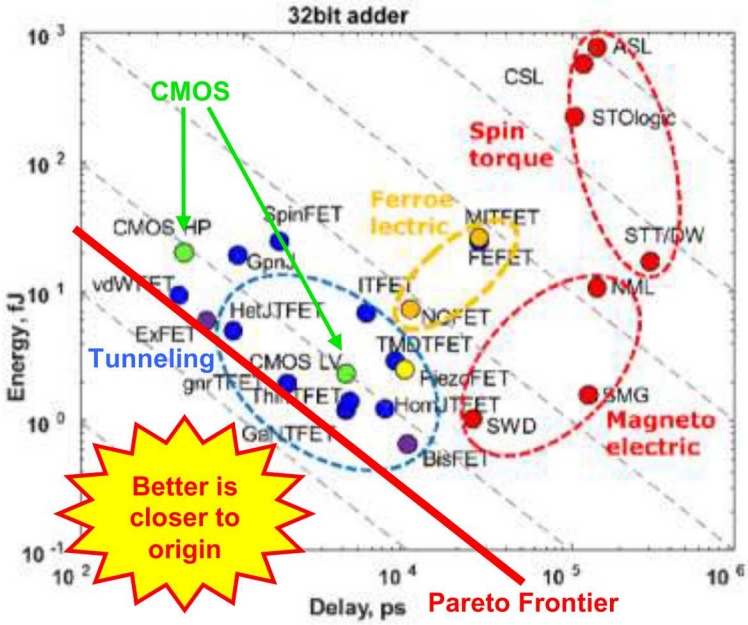


Figure 2. Performance of Beyond CMOS devices. Two variants of CMOS are plotted with green dots labeled CMOS HP and CMOS LV. The graph shows a tradeoff between speed and energy efficiency, but no device has as much as an order of magnitude benefit over CMOS.

Government and industry responses

High Performance computing (HPC) and supercomputers are based on microprocessors, so the slowing rate of microprocessor improvement became an obstacle to building an ExaFLOPS supercomputer. This led to the National Strategic Computing Initiative (NSCI),[5] which sought to establish, "over the next 15 years, a viable path forward for future HPC systems even after the limits of current semiconductor technology are reached (the 'post- Moore's Law era')."

As shown in figure 1b, NSCI led to new research in neural and quantum computing on the top and bottom of figure 1b, which are clearly outside Moore's 1965 vision, or outside the range of the original headlights on Moore's car. It also led to more intense research on device physics intended to find the missing Beyond CMOS device and continue microprocessor scaling. However, the shifting definition of Moore's law left a gap, which is visible in figure 1b.

There has been a change in perspective since 2008 that could make this failure look like the inevitable cost of progress. In 2008, finding the successor to the transistor seemed a reasonable assignment for two dozen volunteers to complete by "Dec. 31, 2008," a concept that seems laughable today when the problem is being addressed by the NSCI[5] in the US with its 15-year timeframe.

Moore's law continues anyway

There is a joke that Moore's law keeps ending, but industry always figures how to resurrect it. In lieu of following the government-designated path, some parts of industry expect to match historical growth rates by shifting manufacturing from 2D to 3D, adopting new architectures, and inventing devices with novel functions. These changes would seem to be within the vision in Moore's original article, but they are outside the narrower interpretation of Moore's law as just microprocessors and memory. As illustrated in figure 1b, it appears industry has resurrected Moore's law by rolling back its definition to what Moore was thinking when he wrote the article defining it.

Industry has reorganized and seems poised to continue the information revolution, although some companies will not make the transition. This report will try to repurpose industry's continuing optimism to Sandia's uses of computing.

Organizing computing approaches by physical limits

Moore's law is based on ongoing exponential growth, but the laws of physics set a performance limit to all but one of the approaches being considered today. So "ongoing exponential growth" actually means "exponential growth that goes on until it approaches a physical limit." If we group the approaches by their limits, as in Table 3, and accept one group where the limit is not known, we get a rigorous basis for organizing the future of computing.

Table 3: Theoretical limits of computing approaches under discussion

Name of approach, grouped by background color	Performance limit or other capability	Approximate investment ($)
Artificial intelligence, such as deep learning and digital neural networks	Smartness and ability to learn[6]	Billion
Quantum computing	Quantum speedup	Billion
"3D+architecture," i. e. continuation of Moore's law	**Thermodynamic (*kT*)[7]**	**Trillion?**
Neuromorphic computing (e. g. analog neural networks)	Thermodynamic (*kT*)[6]	Millions
Novel devices: Spintronics, Carbon Nanotubes, Josephson Junctions, new memories, etc.	Thermodynamic (*kT*)[8]	Millions (each)
Analog computing	Thermodynamic (*kT*)[9]	Millions
Reversible computing	Initial purchase cost[10]	Millions

Artificial intelligence

Table 3's single blue row represents an exciting new area typified by deep learning, but including machine learning and artificial intelligence software running on CPUs, GPUs, and specialized processors. Typical examples are playing the Go board game,[11] tracking user preferences in eCommerce, self-driving cars,[12] and other leading and newsworthy applications.

When these applications appear in the news, they are touted for their "smartness" or driving safety rather than energy efficiency. We know no physical limit to smartness and safety.

This entry has been criticized as being "just software" and with the same goals as neuromorphic computing, which we put in a different group. Perhaps the neuromorphic computing and AI groups will merge in the future, and deep dive I includes an example circuit-level comparison between a neuromorphic circuit and a digital equivalent, concluding that each could be superior under the right circumstances. However, neuromorphic computing has not been demonstrated at sufficient scale to warrant measurement by smartness.

Quantum computing

Table 3's single green row comprises quantum computers, which only seem viable for the limited class of applications where they achieve quantum speedup, and probably only for the further subset where the speedup is exponential.

Exponential quantum speedup makes a quantum algorithm faster than the equivalent microprocessor-based algorithm by a ratio that grows exponentially with problem size. In essence, the speed of the quantum algorithm on a problem of size N, which will require N qubits to solve, will be equivalent to a classical computer whose clock rate is e^N times higher. Since problems can be arbitrarily large, a quantum computer is, in the abstract, equivalent to an ∞ GHz microprocessor. While this is a very big improvement, it applies to only a limited class of problems.

Approaches limited by the thermodynamics of computation

Table 3's yellow group contains devices subject to thermodynamic limits. Landauer[13] developed an implementation-independent theory of minimum heat dissipation in computing due to thermodynamics.

Landauer's theory is most often applied to Boolean logic in microprocessors,[14] so let's discuss it in that context first. Landauer's theory identifies the minimum energy for a Boolean logic operation to be on the order of kT, where $kT \approx 4.14\times10^{-21}$ J = 4.14 zeptojoules at room temperature, k being Boltzmann's constant and T being temperature in Kelvins.

Today's digital computers are closer to Landauer's minimum dissipation than commonly appreciated. A CMOS logic gate in one of today's microprocessors operates at about 100,000 kT,[15] a figure that includes energy attributable to the long wires required by the microprocessor's architecture. Roadmaps and existing industry plans suggest energy efficiency will improve to about 3,000 kT[4] per gate operation, yet this analysis includes only the short wires that would be encountered in a subsystem, specifically a 32-bit adder. This would seem to make the end of current roadmaps 3,000-100,000× above the theoretical limit for energy efficiency and create a substantial opportunity for further improvement. However, most of 3,000-100,000× improvement range is inaccessible to the Boolean logic in microprocessors for the following reasons:

- A microprocessor's architecture requires long wires to move data between the processor and the memory, where energy in the wire accounts for high end of the 3,000-100,000× improvement range. Long wires can be eliminated by breaking them into short segments with amplifiers in between. This may reduce the average energy per gate operation, but the gain is illusory because the amplifiers are essentially gates and the need for more gates offsets the lower average energy per gate.

- Electrical noise, manufacturing variability, and other factors call for operating margins, which are addressed by increasing system energy levels to accommodate the weakest device or the largest noise transient expected in a system.

- Energy efficiency drops as speed increases according to the adiabatic principle in the physics of computation, which is the source of the Pareto frontier in figure 2. Slowing a computer's clock rate to increase energy efficiency is also illusory because a larger computer will be needed to solve the original problem. The energy savings will be offset by the higher cost to buy a larger computer in the first place.

The reasoning above supposedly justifies the argument that "Moore's law is ending" due to fundamental physical principles, but the argument is flawed. Physics is no longer fundamental if it is only applied within a restricted domain, such as Boolean logic and the microprocessor. Many of the devices in the yellow group function differently from transistors and their unique functions can be used to compute directly, potentially sidestepping the limitations of Boolean logic and the microprocessor. For example, the energy attributable to wires could be recovered and reused. Landauer's implementation-independent minimum energy will still apply, but other devices may be able to capture more of the 3,000-100,000× improvement range.

Reversible computing

Table 3's single red row contains reversible computing, an approach with enormous potential and deserving of research support, but for certain specialized applications rather than mainstream computing.

Landauer showed minimal energy to be "typically on the order of kT for each irreversible function,"[13] adding the qualifying term "irreversible" for rigor but (we later find out) without any idea that his colleagues would develop a theory of computation that only uses "reversible" operations and hence has a far lower minimum. Unfortunately, the energy savings of reversible computing is offset by increased complexity that, for the time being, makes reversible computing appropriate only where low energy consumption is extremely important compared to the cost of the computer in the first place.

Quantum computer control electronics is an example of such an application. Some quantum computing approaches require the qubits to be very cold, such as 10 mK, where the ratio of room temperature to the qubit temperature is 300K/0.01K = 30,000. Given a hypothetical perfectly Carnot-efficient refrigerator, energy released near the qubits must be multiplied by this ratio of 30,000 to compute the total energy drawn from the power company. This effectively raises the cost of energy by 30,000×, affecting tradeoffs..

Device physics research for computing

For decades, the dynamo of Moore's law made transistors exponentially cheaper, faster, and more power efficient – in other words, their "performance" by some metric varied over time as e^t. System performance rose exponentially as well, because the combination of improvements became a "tide that lifts all ships."

The actual relationship is that the performance of a computer system is the product of efficiencies at multiple levels as illustrated in figure 1a. So developing an algorithm that is twice as fast makes just as much difference as doubling the clock rate. In the past, system performance improved due to coupled improvements in architectures, transistors, and other integrated circuit advances such as larger chips. Because multi-core processors have been disappointing, Beyond CMOS was expected to drive the entire stack, as illustrated in figure 3a.

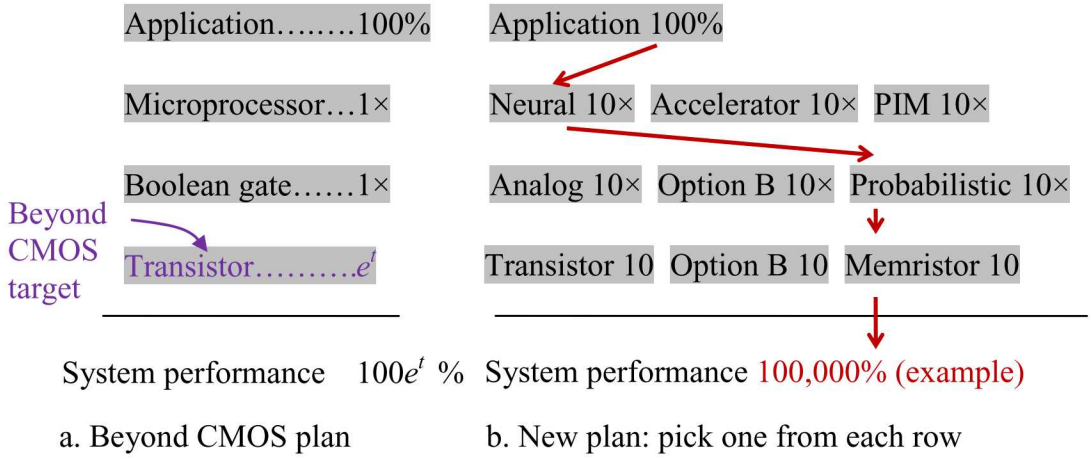


Figure 3: Scaling past and future. (a) If an application runs at a baseline performance of 100% today, traditional scaling predicts it will run at 100e^t % at some later time *t*. (b) In the future, the game will be to pick from multiple options, each of which as an "upside potential." This may yield a performance boost sufficient to drive the economy, but still fall short of the physically implausible e^t dependence.

We observe a new strategy where computer design acquires an additional level of combinatorics, as shown in figure 3b. The computer designer becomes both a device physicist and computer scientist, designing computers by choosing options at various levels of computer abstraction – much like a chemist creates compounds out of a fixed set of chemical elements. Systems must include at least one technology at each level, with system performance being the product of performance at all the levels. Figure 3b shows an example from later in this document where the red arrows tap one technology at each level. If each technology has a 10× improvement potential, the combined potential is 1,000× current levels or a 100,000% improvement.

Pan[16] and deep dive I later in this report are examples.

Turning 3D into a scaling resource

We see industry developing new technical directions that could maintain the historical improvement rate for computers. Instead of finding the desired Beyond CMOS device,[3] 3D memory turned up, but we see 3D memory kicking off a larger set of advances that may ultimately yield the same improvement factor as Beyond CMOS.

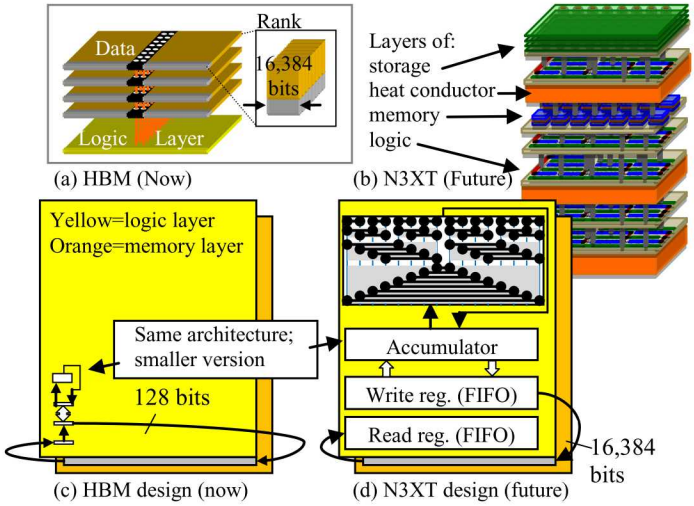
We see examples of the extra dimension being used as a resource for additional scaling, including both 3D chips *per se*,[17, 18, 19, 20, 21, 22] other architectural developments that could apply to 3D,[23, 24, 25] and design principles that translate the additional degrees of freedom into increased performance and energy efficiency.[26, 27]

This introductory section will describe a collection of six improvement paths that follow from 3D memory, with a deep dive II giving more detail and an example. The six paths below may replace some semiconductor parameters that contributed to scaling in the past, like linewidth, metal layer resistance, subthreshold slope, and some others.

#1. Number of total layers, or devices in the *z* dimension
#2. *Z*-direction bandwidth, or tightness of interface in *z* direction
#3. Number logic layers, principally enabled by improved heat removal
#4. Algorithmic adaptation to new logic-memory-communications ratios
#5. Programming model: (a) on-the-fly data routing, (b) static layout, (c) dynamic layout
#6. Diversification of devices: logic, volatile/non-volatile memory, sensors/actuators

Path #1 z-dimension layers and #2 tightness of interface

Setting aside architecture for the moment, industry now roadmaps[2] rising sophistication in the use of the third dimension in chip manufacture. As detailed in deep dive II, roadmaps show physical design advancing from stacked structures like HBM[17] in figure 4a to fully integrated, or monolithic, structures like the N3XT vision from Stanford[28] in figure 4b (the authors are not advocating N3XT over other integrated structures). We assume the number of devices in the *z* dimension will rise over time, and call this the #1 improvement path. Tighter integration along the evolutionary path from HBM to N3XT will raise the efficiency of data movement in the *z* direction, forming the #2 improvement path.[7]



HBM Row: 16,384 bits =

Interface width ×	Cycles	Possible network size
128 bits (HBM)	128	8 = 4 × 2
256	64	24 = 8 × 3
512	32	64 = 16 × 4
1,024	16	160 = 32 × 5
16,384 (N3XT)	1	4,608 = 512 × 9

(e) Scaling scenario

Figure 4. (a) Today's HBM, (b) future N3XT structure, (c and d) an architecture parameterized by the tightness of coupling, (e) a hypothetical scale-up path based on doubling interface with, halving clock cycles, and increasing depth of a functional component.

Path #3 logic layer scaling and heat removal

The orange heat removal layers in figure 4b[28] enable the #3 improvement path, which is the number logic devices in the *z* dimension. The simplified line of reasoning is that logic devices dissipate heat but memory devices do not, so memory devices can be stacked arbitrarily deep in the *z* dimension, at least from the heat dissipation perspective. However, increasing the number of logic devices in the *z* dimension will quickly overload standard cooling, tying the #3 improvement path to increased heat removal capability.

Heat removal layers are technically challenging and may stop scaling soonest.

Path #4 algorithms

Since a chip's purpose is to support computing, the additional capability offered by 3D is useful only to the extent it can be exploited by architectures and algorithms, which vary tremendously in the ratios of logic operations, memory size, and data movement bandwidth and latency.

The deep dive section will explain in detail how 3D affects algorithms. In summary, 3D alleviates two issues from the perspective of algorithms. The first issue is the obvious shortening of wires due to more objects fitting in a 3D sphere than a 2D circle of equal radius. While not fundamental to 2D chips, the second issue is that today's chips are all 2D and come in two types,

logic and DRAM. At the system level, the fact that there are two chip types imposes a large speed and energy penalty when information must flow between types, i. e. between logic and a memory too large to be implemented with registers or SRAM. This "information modality" restricts the designers' choice of algorithms. 3D will alleviate the restriction, but the benefit will vary widely depending on the ratios of logic operations, memory size, and data movement bandwidth and latency.

With the background just presented, figures 4c & d show a notional view of 3D scaling using sorting network. A sorting network is an example of a structure that becomes feasible only with 3D due to excessive memory bandwidth requirements, as explained in deep dive II. Figure 4e shows a hypothetical scale up sequence where the interface gets wider and bandwidth increases, supporting a sorting network that grows in size and throughput over multiple generations.

Path #5 Programming model

The discussion of 3D above treated the algorithms as a given, but rising sophistication in the way 3D is programmed or otherwise controlled will be the #5 improvement path. We project that there will be multiple levels of programming sophistication for 3D:

(1) Standard programming with higher performance memory, which is the way HBM is used today. This is a baseline level and does not unleash the broader range of algorithms enabled by the #4 improvement path.

(2) Design like an Application Specific Integrated Circuit (ASIC) where function and memory are lain out in 3D prior to manufacture. This level unleashes full performance but is inflexible.

(3) A new level that lets software allocate functions and memory on the fly. We will call this level "dynamic layout."

Path #6 diversification of devices

The scaling path from figure 4a to b also includes functional diversification of layers, which we call the #6 improvement path. From Sandia's perspective, we see 3D structures comprising logic and memory (i. e. DRAM-equivalent memory) to be the first version of 3D to be useful to Sandia, as it will resolve the von Neumann bottleneck and address current issues with energy efficiency. After this, 3D with logic and non-volatile storage could be useful for data intensive problems, as it would allow, for example, computation on stored data without reading the data from storage first, saving time and energy. One or more layers of (silicon) photonic integration could improve long distance communication within a module or as I/O from it to the outside. Then, 3D integration that includes a sensor array could create an energy efficient smart sensor.

However, industry roadmaps include heterogeneous integration of, for example, RF transistors, where the projected benefit is to save a few dollars on the manufacturing cost of a smartphone. This is an example of an industry initiated direction we find interesting, but not a Sandia priority.

Photonics for Beyond Moore's Law Computing

It's an interesting question as to where one might draw the line between Photonics for (CMOS) Computing and Photonics for Beyond Moore's Law Computing (BMC); since there has been little work associating photonics with BMC computing, we are somewhat free to choose. But, in general, researchers familiar with BMC computing are not all that familiar with photonics, so we cover the topic from near-term trends in transceiver-based interconnect through optical computing.

As supercomputer architects typically focus on power consumption as being the largest obstacle to future progress, the near-term trends are not progressing in a way that satisfies their needs. Data centers are focused primarily on cost, with bandwidth density being a key factor, and power consumption being a distant 3rd in importance. Hence any tradeoff between cost (more optics) and power (more electronics) tends to favor more electronics in today's transceiver market. The true way to achieve low power consumption is intimate integration of photonics and electronics, not so much for on-chip interconnects as off-chip ones, but avoiding the interface issues between the photonics and electronics. The approach is largely similar for interfacing to CMOS as it is interfacing to ICs comprised of BMC transistors, but some of the subtle differences are described in this section.

As networking plays a key role in any supercomputer, be it for connecting nodes or processors to memories, researchers are starting to investigate electrically controlled, optical switching to achieve low power scalable bandwidth solutions. Optical packet switching, while in theory even more useful, is a long way from being scalable in complexity and power with the approaches that have been demonstrated in research labs today.

Perhaps the most promising development in optical computing recently is to make use of arbitrary linear transformations using cascaded interferometers to provide a lossless method of proving arbitrary weighted connectivity between inputs and outputs, for example in a vector matrix multiply. As a key component of neural-inspired computer architectures, this shows promise to revolutionize optical approaches to these architectures. However, a more detailed analysis is required to fully analyze how this compares to other electrical approaches, for example, those based on memristors as well as CMOS logic, not only in terms of power consumption, but also in terms of equivalent operations per unit area and cost.

Lastly, we discuss a few areas that we think might warrant some small investments in technology to help move the field forward in areas that the data-center/communications markets are not addressing.

Conclusions

The authors' opinion is that "Moore's law" is a marketing slogan based on an important but imprecisely defined technical idea. As long as "Moore's law" helps sell products, we believe product marketing forces of will try to redefine it so it remains true.

The traditional scaling path based on reducing line width may be reaching its limits, but it hasn't reached them yet. We believe the community can expect some modest improvement due to narrower linewidths and lower energy per function.

We introduced the idea above that the traditional device physics scaling path will give way to a more diverse set of technologies, an idea developed in deep dive I below. We do not see any device or architecture beating the others, like the transistor and microprocessor did in the past. Instead, we see dozens of devices, interconnect methods, chip designs (e. g. 3D) becoming tools in a toolbox of options. Designers will solve problems in part by picking and choosing the right options. The most valuable technologies in the toolbox will be the ones that can form the greatest number of combinations with others, as opposed to some notion of universal goodness.

Sandia may have a role at the system level similar to the development of parallel computers in the 1980s and 1990s. While industry was perfecting the personal computer, the government assumed primary responsibility for parallel software that could gang together "killer micros" into "hypercubes" that could solve larger problems. Repeating this scenario, industry is now shifting to the roadmapping of 3D structures that support a broad variety of architectures. We suggest the Sandia understand how this new roadmap will support architectures and algorithms that meet the Sandia's specific needs.

We don't see a reversal to the rising cost of chip fabrication plants (fabs), which make 2D chips today. To sidestep path #3's concern about excessive heat dissipation from 3D logic, the highest performing module would have 2D logic with 3D memory and/or storage. Balancing the amount of 2D logic with 3D memory will put continued pressure to create large and highly sophisticated 2D logic chips, which is the direction of the most expensive fabs.

Each of the following three deep dives includes its own conclusions.

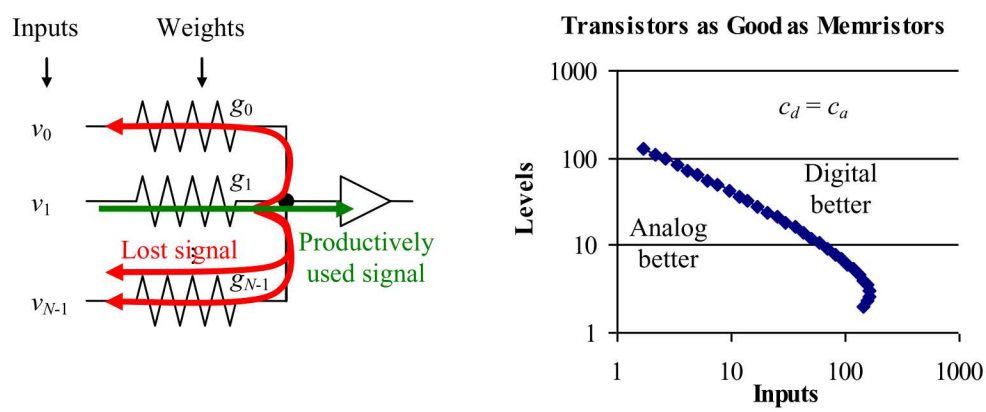
Deep dive I: Combining device physics and computation

There are just a dozen or so approaches to computing known, such as the microprocessor, neural network, quantum computer, some other human-created approaches, and biological processes. Having failed to find a Beyond CMOS device that could extend microprocessor scaling, device physicists are now looking more broadly. This deep dive will focus on devices with novel functions that would extend the number of known approaches to computing (see also Pan[16]).

The need to consider devices in context

Making better devices has been a top goal in the semiconductor industry, but we will use the analog neural network as a counterexample to show that the concept of "better" is too simplistic. To compare devices as shown in figure 2, device performance must be abstracted to a few numbers. However, our example, from one of our articles,[9] shows that the identity of the best device changes based on the problem being solved. For the analog neural network, it depends on scale and precision requirements. Scale and precision depend on a user's needs, and numbers assigned to a device in isolation cannot capture the scale and precision needed by a user.

The example compares the minimum energies of a memristor-based analog neural network and a digital circuit performing the same function. The example compares a single column of an artificial neural network computing the vector dot product of analog input voltages v_i and analog conductances g_i stored in memristors (conductance = 1/resistance) shown in figure 5a with a functionally equivalent digital circuit. Since the analog circuit is not reprogrammable beyond changing the conductances, fair comparison does not require the digital equivalent to be changeable to any greater extent, so we used a fixed linear array of digital adders and multipliers.



(a) Analog and why it scales as it does (b) Dividing line analog/digital better

Figure 5. Analog versus digital. (a) Circuit and intuitive explanation of why analog and digital scale differently. (b) Partition of parameter space into regions where digital is better than analog and vice versa.

The article[9] compensates for "maturity bias" by using theoretical limits, but we will add two new factors here: c_a (analog) and c_d (digital) representing the amount by which actual energy consumption is greater than the physical limit, which makes them the improvement factors in figure 1b.

Circuit equations rigorously tell us how a circuit works, so comparing them appropriately should tell us which device is better. The article[9] included the minimum energies of digital (E_{digital}) and analog (E_{analog}) computations, with the circuit equations reproduced in table 4, although with the c_a and c_d factors. The equations are for an N-element dot product, or a problem scaled to size N, with the precision of each signal being L distinguishable levels. Both approaches produce errors if energy is reduced too far, so variable p_{error} is the allowable error rate.

Table 4: Minimum energy of an *N*-element dot product of *L*-level or $\log_2 L$-bit values

Approach	Computational complexity expression for minimum energy				
	Constant factor ×	Reliability ×	Accuracy ×	Scale ×	Energy unit
E_{digital} =	24 c_d	$\ln(1/p_{\text{error}})$	$\log_2^2(L)$	N	kT
E_{analog} =	(1/24) c_a	$\ln(1/p_{\text{error}})$	L^2	N^2	kT

A straightforward way to figure out which is better is to set $E_{\text{digital}} = E_{\text{analog}}$ and plot. Digital will be better on one side of the curve and analog on the other. Algebraically rearranging the equations in Table 4 yields $N = c_d/c_a\ 24^2 \log_2^2(L)/L^2$, which his plotted in figure 5b for $c_d = c_a$. The plot in figure 5b shows analytically that digital is better than analog at high precision and large scale, a widely known and published subjective insight,[29] but we have not seen another analytical derivation in the literature.

The arrows in figure 5a explain the difference intuitively. Digital logic gates restore Boolean signals to proper voltage levels. If Boolean 0 is supposed to be 0 V, but noise or a circuit imperfection causes it to be 0.032 V, the gate's amplification and thresholding will restore the signal to 0 V. However, analog circuits do not have digital restoration. Since amplification consumes energy, this explains why the analog circuit in figure 5a, which doesn't have amplification, wins in the lower left figure 5b.

However, the red arrows show the input signal flowing backwards into the other N-1 inputs, reducing signal strength at the amplifier by a factor of about N. Increasing all energy levels by a factor of N fixes the signal strength issue, but raises the minimum energy by a factor of N. This is why table 4 has an N term for E_{digital} and N^2 term for E_{analog}.

Failure to find the desired Beyond CMOS device actually makes it easier to assess new computing devices. Years ago, deciding whether the memristor was better than the transistor would have required a double guess, i. e. guessing how much money would be invested in each device in order to assess its energy efficiency. The fact that CMOS is approaching its thermodynamic limit eliminates one guess. So the analysis could, if true, conclude "memristors will beat transistors once you invest $*N*."

Generalizing solution of the traveling salesman problem

Williams and his colleagues at HPE developed a physical and computational system for solving the Traveling Salesman Problem (TSP), which we will generalize into a vision of how device physics research may be used in the future. Their process is an outgrowth of traditional analog computing, but uses novel nanoelectronic devices instead of circuits like op amps, potentially

yielding systems that are more compact and consume less energy. The generalization will be a toolbox of options and a constructive process for assembling devices into circuits that solve broad classes of problems specified in advance.

The toolbox can be expand with new devices if needed to solve a problem, but the toolbox for the TSP example contained:

- Arraying devices to perform vector and matrix operations
- Using devices both with and without persistent state
- Using both digital and analog signals or values
- Using noise (actually chaos) as a part of the computation instead of noise being just a parasitic that limits performance
- Using circuits with feedback, which has some similarity to processor executing multiple instructions
- Creating a hybrid between an analog circuit and an external program
- Using an energy minimization circuit as a primitive, where energy minimization is a more complex primitive than available in digital computing

The circuit by Suhas Kumar[30] in figure 6a solves the TSP by energy minimization as shown in figure 6b, but the correct answer appears only probabilistically, so the circuit is run repeatedly and monitored for the best answer, as shown in figure 6c & d.

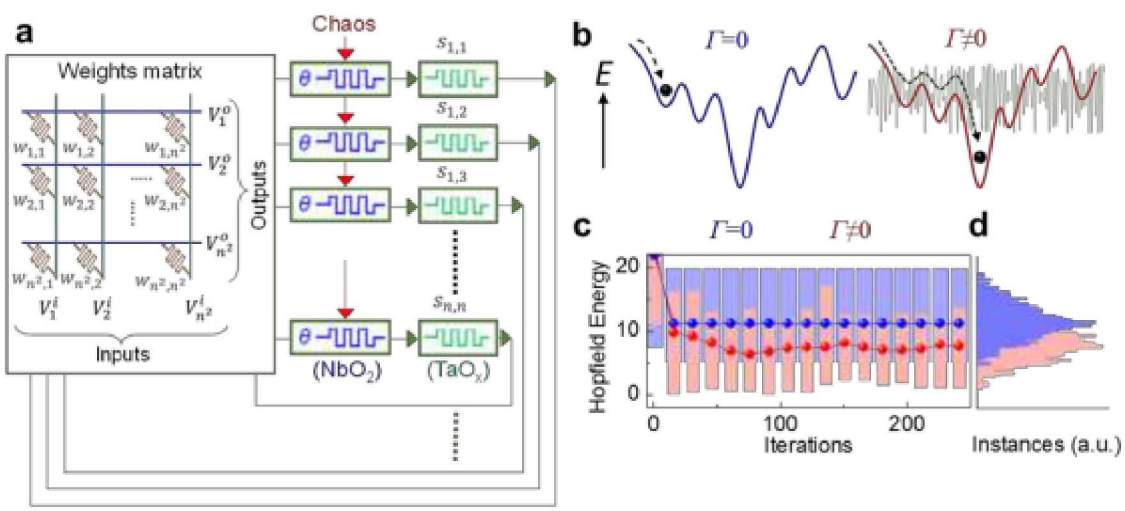


Figure 6: Hopfield network. (a) Schematic. (b) Energy minimization. (c) and (d) solutions to the Traveling Salesman Problem (TSP)

Energy minimization as physics-based programming primitive

There are century-old mathematical proofs that Boolean AND, OR, and NOT gates form a universal computational basis,[31, 32] meaning a network of universal gates can compute any function. However, even in pure digital logic, the resulting functions are not minimal in terms of

gate count, energy, or any other metric. For example, an AND gate can be created from a CMOS NAND gate followed by an inverter at the cost of using two gates to create one.

Furthermore, single analog devices typically process signals that are equivalent to, say, an 8-bit binary number. This makes a single analog device equivalent to the dozens of digital gates needed for an 8-bit digital adder or multiplier.

Energy minimization is much more complex in digital electronics than analog. The Hopfield network,[33] one of the older neural network formulations, performs energy minimization at the circuit level. A Hopfield network is essentially the neural network column in figure 3a, replicated to become an array, and with its output fed back to the input in real time. Figure 6a is a Hopfield network if the blue and green memristors are taken to be amplifiers.

To use the circuit in figure 6a to minimize energy, the weights are set up with a description of the energy potential and the v_n^i's are forced to some initial guess for an answer. When the forcing is removed, the circuit becomes a group of interacting feedback loops, which settle in several time constants of the analog circuit. The mathematics of Hopfield networks says the system will find the minimal energy as shown in figure 6b. A Hopfield network may also find a local minimum energy, which is a limitation addressed below.

Many software subroutines solve problems by simulating physical energy minimization, suggesting a computer with an energy minimization "instruction" could perform such a subroutine in one step. This could be more efficient than digital energy minimization, because every memristor would be the equivalent of a multi-bit digital multiplier. On the other hand, energy minimization problems vary in the amount of precision needed to specify the energy potential. If the problem needs energies specified at high precision or over a large range, like floating point, the analog circuit would suffer the same issue with precision shown in figure 5b. So analog energy minimization may work better for some problems, but digital for others.

Hard computing problems and using noise as a resource

The TSP is easy to state but can be very hard to solve. TSP's objective is to minimize the total distance traveled by a salesman visiting n cities exactly once based on specified city-to-city distances. In the notation of computational complexity, TSP software solutions have complexity between $O(n^4)$ and $O(n^3 2^n)$,[34] depending the way cities are distributed. The high end of the range includes a 2^n factor, making it NP-hard and hence intractable to today's computers, including quantum computers and the system in figure 6. Computational complexity theory thus tells us that no system can solve TSP in general and at large scale – unless the system periodically gives a wrong or approximate answer.

The system in figure 6 uses methods equivalent to some software solutions, all of which get the best answer for a given resource usage. Both software algorithms and the system in figure 6 run energy minimization in a loop, with a software random number generator or the blue chaos devices causing each cycle to be slightly different from the others. The system monitors the solutions as they appear, as shown in figure 6c, recording the best solution over many trials.

An architecture based on an energy minimization subsystem

Today's computers have functional subsystems, such as floating point units, but perhaps future computers could have subsystems for other functions? Let's consider how to integrate a memristor array into a computer, either a simple one like shown in figure 3a or the full circuit in figure 6.

1. Deep learning is the popular accelerator vision today, where a conventional computer would contain a neural network circuits like the one in Figure 3. This would let the computer do deep learning more efficiently.

2. The TSP problem in Figure 6 is another example. In lieu of recognizing features in images (a popular use of deep learning), a conventional computer could load inter-city distances into registers or memory. The registers or memory would actually be the memristor weights in figure 6. A new machine instruction would run the TSP problem, perhaps returning the first city in the accumulator and a list of other cities in memory. By interpreting inter-city distances in different ways, this computer instruction could be used for scheduling movement of a robot arm between points in space or looking through a list of tasks in discrete event simulation and returning the most appropriate task to execute next. If applied to the internal operation of a CPU, it could decide whether to flush cache at a given instant or prefetch more data from memory.

3. The energy minimization underlying TSP could be generalized to, for example, molecular dynamics simulation. The approximate positions of molecules would be loaded in registers or memory and the Hopfield network would adjust their positions for minimal energy.

4. Data compression, like zip and gif files, use simplistic "greedy" encoders. For higher compression rates, the encoder would have to make intelligent choices, on the fly, about whether to create a new code word for the decoding dictionary or reusing an existing one. Codewords are like cities, so the problem is like deciding whether to go to a specific city early in the path or later.

However, the new computing system will only be useful if the speed, energy efficiency, or some other metric is higher than other approaches, including implementation in software on a conventional computer. In this last regard, an analysis like the one in figure 3 should be performed.

Conclusions

We started a line of reasoning with a description of the failed quest for a Beyond CMOS device and how the Pareto frontier in figure 2 shows no such device exists. While this led some people to claim "Moore's law was ending," figure 5b shows no simple 2-axis plot like figure 2 can identify the device we're looking for. Others have made this same argument, such as Pan,[16] who plots a cellular neural network in lieu of an adder but uses the same axes as figure 2. This argument also applies to the popular neuromorphic crossbar circuit, where proponents do not use the axes in figure 2, but do not make comparisons against CMOS either.

Recommendations

Physical science research into materials and devices in isolation seems to giving way to research on devices in circuits that address specific problems in computing, specifically where performance limits can be identified to see if the circuit has a chance of beating CMOS.

The proposed top-down analysis has been performed before with profound results. For example, Alan Turing and John von Neumann went through the process above for what became today's microprocessor[35] – albeit focused on ability to solve the broadest possible range of problems rather than speed or energy efficiency. Richard Feynman[36] and Peter Shor[37] went through the process for the quantum computer, specifically focused on computational efficiency, while Hinton[38] and many others made contributions for machines that learn.

It should be possible to seek top-down solutions directly. Today Wikipedia has concise descriptions of the advances in the paragraph above – the microprocessor, quantum computer, and neuromorphic computing, including the all the relevant issues. While few people could write such a description in advance, the world needs just a handful of new approaches.

We suggest a research program where responders would be expected to address three requirements:

1. Successful proposers would identify a new computing approach, defined as a principle of physics that modifies information, where the way information is modified can be connected to an important problem in computing.

2. An architectural concept for the computing approach, or a way to encapsulate the physical principle into a computational subsystem that would presumably be connected to conventional computer. This may include a toolbox of devices, their function, and how to combine them to solve a range of problems.

3. An analysis of the speed, energy consumption, and cost to manufacture a system showing how it can beat a CMOS computer in when both are assessed in a consistent way, given sufficient investment.

Deep dive II: Future scaling of Sandia applications

This deep dive will be oriented towards Sandia's computing needs such as supercomputers, variants of supercomputing like graph processing,[39] and highly focused needs that currently use ASICs for, say, signal processing.[40]

In this report, we are making an original effort to apply new consumer product roadmaps to high performance computing and ASICs. We observe industry developing a scale up path for 3D chips at the device level[41] and plans to use the technology for better consumer products.[7] Separately, we see people associated with Sandia's benchmarking supercomputer prototype node boards containing new architectures with 3D memory.[42] There is also benchmarking of graph algorithms, such as the DARPA HIVE program.[43] The government efforts are for a specific point in time, which we will generalize to apply to a time range like a roadmap.

Our quest to apply the 3D roadmap to Sandia's applications led to a situation like the development of parallel supercomputers from consumer PCs. Starting with hypercubes in the 1980s, Sandia took responsibility for developing the theory of parallel algorithm scaling and developed software infrastructure that would scale to progressively larger supercomputers over time. While all scaling principles have the same basis, the results for parallel computers and 3D chips are not the same. So we took an initial step of developing a 3D test architecture, called Superstrider, simulating it,[44] and developing enough theory to explain the test architecture. However, this report goes further and includes a vision of a general theory of scaling that would apply to the six improvement paths.

We will also reference a 2014 document[14] created by Joneckis, et. al. for the government. This document expressed concern about the future of specialized processing in a section titled "Theory and Practice of Processor-in-Memory Computing," notably recommending the community try and develop the "ability to program increasingly heterogeneous computers that are approaching a PIM architecture."[14] Although we didn't know about Joneckis' report until this document was nearly complete, we follow his suggestion.

Basic changes due to 3D chips (#1, #2)

Industry made two types of chips until recently, logic and memory. The DRAM fabrication process leads to about 100× more data per unit area than the equivalent logic process. However, dividing chips into two classes leads to the previously mentioned "information modality." At any given time, a unit of information must be in what we call "logic mode," or located in a logic chip, or "memory mode," and stored in a memory chip. Figure 7a shows a yellow logic chip, orange memory chip, a data pathway between them in gray, and a green curve showing information switching modes. The exponential improvement rate of Moore's law applies within logic and memory chips, but only weakly to data pathways between them, so algorithms that move information between the two modes may scale poorly because the interface will become a bottleneck. We use DRAM to represent all memory and storage types, because the other types have the same issue.

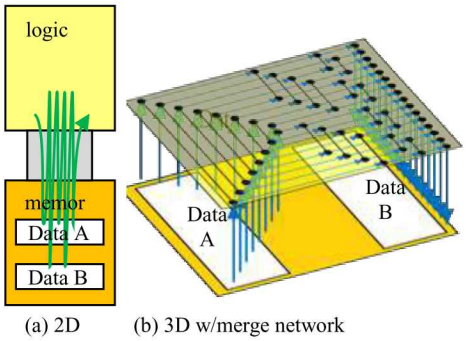


Figure 7. (a) 2D systems comprise logic and memory chips, with the green curve illustrating a mixed logic-memory calculation that is inefficient precisely because of the partitioning into logic and memory. (b) A 3D system with tight coupling between logic and memory avoids high latency paths, bandwidth bottlenecks, and conversion of signals to high energy levels for off-chip interconnects. The blue curve shows a representative data movement step in sorting. (c) The sorting network used for the 3D example.

Figure 7b shows the basic structure of a 3D chip, characterized by logic and DRAM devices on the different layers. While information will still be in "logic mode" or "memory mode" at any point in time, the layers connect across their large faces, making room for more, shorter, layer-to-layer connections that reduce the impact of information modality. Figure 7b shows an orange memory layer with information Data A and Data B located in specific regions. A logic layer is located on top, with the example pattern of for a bitonic sorting network (discussed later). Information flow from logic to memory is shown through blue arrows. The length of the blue arrows is very much exaggerated in the diagram.

The logic layer in figure 7b shows the typical design style for a special purpose chip or FPGA, where the design is defined as a register-transfer diagram or layout comprising registers, arithmetic units, sorting modules, data storage, and so forth. However, in figure 7b, the logic is shown in a yellow logic layer and the data storage is on an orange layer shown below it in the 3D diagram. The location of the data on the memory layer should coincide with the placement of the logic that creates or uses the data in order to achieve maximum efficiency.

The HBM diagram in figure 4a is the principal 3D option available now. Figures 4c & d intend to show the same architecture resized for the difference in the tightness of the 3D interface between HBM and future visions. For example, HBM transmits a 16,384-bit DRAM row as 128 128-bit data packets, which will take 128 cycles to transmit, and will require more energy and take more time than the fully integrated N3XT vision diagram that would presumably move the same data in one cycle. We have tried to scale the register transfer diagram between 6a & b to show the relative size of the logic needed to fully utilize the memory interface.

We created a hypothetical 3D scaling path in figure 4e, bounded by the structures in figure 4a and 4b. We have no control over the HBM standard or the N3XT vision, but we created a table of multiple generations of 3D interface where the width gets wider and faster from one

generation to the next. We also used the sizes of sorting networks to estimate relative chip area for a network that is a wide as the interface.

How does our projection in figure 4 coincide with industry roadmaps? Figure 8 is based on the current working draft of the 2017 IRDS roadmap. Details of figure 8 are beyond the scope of this document, but it shows a series of semiconductor processes going from today's FinFETs to nanowires, then vertical nanowires, and then a monolithic 3D similar to N3XT.

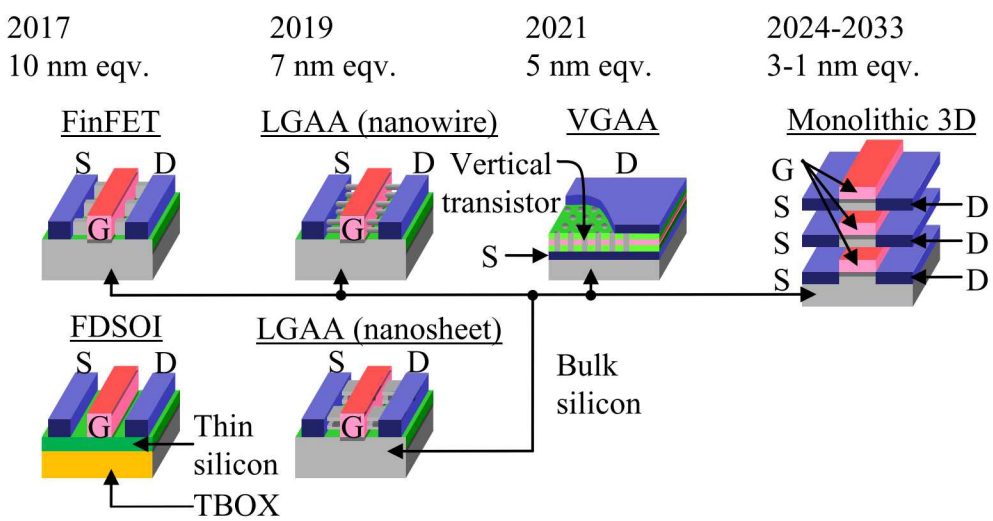


Figure 8. Roadmap of 3D structure previewed for 2017 International Roadmap for Devices and Systems (abbreviated). Legend: S = source, G = gate, D = drain, FDSOI= Fully depleted silicon on insulator, LGAA = Lateral Gate-All-Around, VGAA = Vertical Gate-All-Around, TBOX=Thin Buried Oxide.

The IRDS roadmap will also include a projection of functional changes across the scale-up path corresponding to figure 4c & d, but industry projections are very different from figure 4. The draft IRDS projection has five levels, approximately as shown in table 5.[7] We find no discussion of algorithms.

Table 5: Industry plans for generations of 3D VLSI by generation

Generation 1	A high-performance digital ASIC stacked with a layer of non-scaling components like analog, power management, and I/O.
2	Two layers, one with p-type transistors and the other with n-type, allowing vertical gates for higher density.
3	A logic and a memory layer.
4	A layer for special devices, such as RF.
5	3D VLSI, like N3XT like in figure 4d with applications like neuromorphic computing and sensor interfacing.

The general connection between 3D and algorithms

The shift from 2D to 3D chips affects algorithms by alleviating the information modality discussed above, in addition to shortening wires and making memories larger.

If a problem can be solved with one of several algorithms, picking the right one might reduce the number of mode changes to the point where they don't matter. However, in certain problem classes, such as sorting, there is a minimum number of mode changes over all possible algorithms, and the minimum can be so high that mode changing becomes the rate limiting issue.

Just about every programmer will know that Quicksort[45] is one of the best general sorting algorithms, with an average time complexity of $O(n \log n)$. This information is readily confirmed in many textbooks and the Wikipedia article. However, if one scrolls the Wikipedia about halfway down, another table appears with entries like bitonic sort[46] with the much faster speed of $O(\log^2 n)$.

For background, we are using big-O notation for algorithmic performance. $O(f(n))$ for function f and problem size n means a value proportional to $f(n)$, so, for example, $O(\log^2 n)$ steps might be $234.4 \log^2 n$ steps. This method of rating algorithms separates the behavior of the algorithm from clock rate and other properties of the underlying computer.

The discrepancy is that the terminology for sorting algorithms follows software conventions, which are based on a von Neumann computer implemented with the structure in figure 7a. A von Neumann computer was originally viewed as doing one thing at a time – although recent multicore microprocessors can do up to, say, a dozen things at a time. However, a bitonic sort requires simultaneous access and comparison of every record in one half the list being sorted with a different record from the other half. When a von Neumann computer tries to run a software implementation of a bitonic sort, the processor must do the compare and swap operations one at a time, raising the algorithm's complexity from $O(\log^2 n)$ to $O(n \log^2 n)$, not due to a change in the algorithm, but due to a difference in the computational model. This means the best sort algorithm is different depending on whether the implementation platform is figure 7a or b:

Figure 7a (2D): Quicksort O($n \log n$) faster than bitonic sort O($n \log^2 n$)
Figure 7b (3D): bitonic sort O($\log^2 n$) faster than Quicksort O($n \log n$)

Sorting networks like bitonic sort are well known in the ASIC design community, but the capacity of an ASIC is limited unless it is attached to a DRAM, in which case the software argument applies. Let's see how much faster the integrated structure in figure 7b can sort compared to a von Neumann processor. Say we want to sort a billion $\approx 2^{30}$ numbers, which is about the capacity of a memory chip today. Quicksort has O($n \log n$) steps or about $10^9 \times \log_2 2^{30} \approx 30$ billion steps. The bitonic sorter[46] (containing a repetition of the diagram in figure 7b) would have O($\log^2 n$) = $(\log_2 2^{30})^2 = 900$ steps. The ratio of these numbers is 33 million.

While it is theoretically possible for a bitonic sorting network to operate 33 million times faster than a von Neumann computer, the structure in figure 7b would overheat if it tried to sort a sort a billion numbers in 900 steps. However, we see improvements of 1000× in the literature for fully-integrated 3D.[28, 20, 44]

In conclusion, 3D chips offer a fundamental change to algorithms, but the change will be seen differently by programmers and ASIC designers. 3D chips will allow programmers to use more efficient algorithms, like sorting networks. The algorithms are not new, but they could not be executed at full efficiency on systems built from separate logic and memory chips. ASIC designers will see things differently. They have been able to use the algorithms all along, but only for modest-size problems where the logic and data (at SRAM density) fit on a single die. Thus, for the ASIC designer, 3D chips can scale further, to the point where the logic fill the entire logic die and the data is off-plane in the third dimension.

Matching algorithm scaling to the new improvement paths (#4)

Table 6 shows our vision of how the #2 and #3 improvement paths for hardware would enable a #4 improvement path for algorithms.

Table 6: 3D physical scaling and representative algorithms, with staged development path #4.

#3 improvement path →

#2 improvement path →	n=1 logic layer, m-1 memory layers	n logic layers, m-n memory layers
Low z-direction bandwidth (stacking)	Stage 1: microprocessor or GPU and memory in a common package	Stage 3: 3D systolic array or finite difference equation
High z-direction bandwidth (monolithic)	Stage 2: Sorting network	Stage 3: Radiation transport Long distance forces

The upper left entry is the stage 1 baseline available in currently shipping products such as high-end graphics cards and prototype supercomputer nodes, both with HBM memory. The purpose of the 3D memory is to increase memory performance for existing algorithms.

Let table 6's vertical direction represents the #2 improvement path or the amount of bandwidth available in the *z* direction. This improvement path only benefits algorithms where a fixed amount of logic operates on a progressively larger amount of memory – because algorithms with growing amounts of logic would cause overheating. The sorting networks described above and illustrated in figure 4c & d have the required property, as do FFTs[27] and other algorithms.

Let table 6's horizontal direction represent more layers of logic, or the #3 improvement path. A systolic array is a 1-, 2-, or 3-dimensional array of ALUs. A 3D systolic array would benefit from 3D chips. Irrespective of the number of dimensions, these structures are known to ASIC designers and parallel computer programmers as the fastest and most efficient way to compute certain functions. ASIC designers will readily understand 3-dimensional systolic arrays in academic papers, such as matrix multiplication that requires a 3D interconnection pattern.[47] These arrays have been impossible to implement because 3D chips didn't exist.

Table 6's lower right entry is for simultaneous high *z*-direction bandwidth and a large number of logic layers. Further description is beyond the scope of this document.

Programming model (#5)

As discussed in the introduction, we see programming or design methods for 3D as the #5 improvement path. 3D is being programmed today at what we call levels 1 and 2, neither of which unleashes the full potential of the advance. Let us discuss the first two levels and give a vision for combination of the two.

Level 1: The reader certainly knows about memory addressing, but we need to describe it to create a baseline for improvement. The von Neumann architecture insulates the programmer from knowledge of where data is located, which makes programming easier. By the time a program is running on a computer, the program will identify data by a memory address, which is a number typically located in the accumulator or a general register. To act on data, the computer sends the address through an address decoding network to the actual storage location on another chip, receiving the data back via a bus.

Addressing has a high cost in terms of execution time and energy consumption. In fact, the cost of address decoding is similar to the mode change cost when information goes from logic to memory chips and back. Since the costs are similar, there has been no need to separate the cost of addressing from the cost of mode changing. Reducing the cost of mode changing through 3D chips will not help overall system performance if the cost of address decoding cannot be reduced as well.

Level 2: ASIC layout model, which is used in architecture research today.[19, 20] The structures in figure 4a & b can be controlled, or programmed, via layout principles ubiquitous in ASIC and logic design. In logic design, logic and memory are placed or laid out on the chip's surface

during the design phase so data is as close as possible to the logic that produces it or needs to use it. This makes much of the overhead associated with data movement simply disappear. The extension to 3D is obvious, although today's design tools do not readily support 3D.

Level 3: "Dynamic layout," is something we observe in other forms of computing and is easy to project as a consistent vision. It is common for one software subroutine to invoke another recursively and on the fly by (a) assessing resource requirements for the subroutine, (b) finding adequate resources from a pool of available resources, such as memory on the stack, and (c) running the subroutine using the resource. This method is also used in parallel supercomputer job scheduling,[48] for example, where jobs are lain out on or allocated to contiguous supercomputer nodes. The underlying algorithms could be extended to the structures along the scale up path of figure 4 if the resource being allocated includes both logic and memory. The complication is that somebody will have to create an algorithm that can figure out the resources needed for a subroutine in advance and then essentially map or place the components of the subroutine on the available resources before execution. There are some easy examples of this process, but sometimes it will be hard.

Where we are today

At present, there are two main commercially available 3D DRAMs: HBM[17] and Hybrid Memory Cube.[18] 3D Flash is widely available commercially for Solid State Disks – and it is monolithic 3D as in figure 4b with up to 72 layers.[49]

3D logic can be stacked, but we are not aware of any commercial products in widespread use. There are companies that will make general 3D stacks on a case-by-case basis.

Also, Xilinx[50] and Altera/Intel[51] have announced FPGA-based products that could implement the system in figure 4a, both companies claiming engineering samples in the first half of 2018. In one case, the product will be FPGAs containing a native (hard) microprocessor and an on-package HBM stack. In other words, they could implement figure 4a without additional real hardware, provided that somebody creates the FPGA programming for the unique architectural structures – which is hardware design.

Recommendations

While Moore's law may be ending for 2D chips, industry has a 3D roadmap that may lead to equally good results. However, Sandia's staff and management will need to understand differences in applying 2D and 3D chips. This is similar to Joneckis's 2014 recommendation that Sandia develop "theory and practice of PIM computing,"[14] but developments in the last three years let us put a plan into this document whereas the earlier document just expressed the need.

Sandia may be able to steer industry investment by demonstrating the value of 3D in unexplored domains. It would be helpful for Sandia to develop a few architecture families that scale with 3D roadmaps. These block diagrams of physical structures, architecture, and software would be parameterized by positions on paths #1-#6 of the 3D roadmap, solving the same problem albeit at larger size as the roadmap progresses. Industry is investing heavily in 3D for consumer

applications, yet Sandia-generated examples could provide "applications pull" to encourage industry to better tune the technology to Sandia's needs. (Of course, paths #1-#6 and figure 4 are examples developed for this report, with reality likely to be different in detail.)

For paths #1, #2, and #4, work on design principles can begin right away by using FPGAs that have been announced by the two companies previously mentioned[50, 51] for prototyping. The authors did a simulation for one example called Superstrider,[44] showing a FPGA prototype system outperforming current chips by 50×. As the first step in a scaling sequence, 50× is pretty good.

Industry is likely to fund R&D for path #3, multiple logic layers, but the underlying technology for 3D logic is likely to be much more difficult to use and may require additional attention, including from Sandia.

While industry is likely to develop programming tools and software libraries to support path #5, industry is likely to focus on consumer products. Just as Sandia has primary responsibility for developing supercomputer software tools and libraries, we recommend the Sandia consider developing design tools for complex applications of 3D chips.

While both government and industry will have considerable interest in the #6 improvement path, 3D chips with sensors and/or actuators, the government's sensors and actuators are likely to go beyond those in the commercial sector and therefore require specific investment.

3D standards. While industry seems to have the constituent technologies to transition 2D to 3D, many of the advances cannot be deployed on the same chip. Sandia should encourage the development of standards so, for example, any of a number of 2D logic type could be designed and manufactured with any of a number of 3D memory types.

Deep dive III: CMOS Optical Interconnect

On-roadmap solutions

About 10 years ago, the advent of very high speed analog to digital converters ushered in a new era of long distance and metropolitan communications. The initial purpose was to replace dispersion compensation fiber with electronic circuitry, but within a few years, the same technology allowed multi-level signaling to increase the bit rate by transmitting multiple bits per symbol at a reasonable clock rate, at that time, and still today, 25 GHz. Advanced adaptive forward error correction (FEC) followed, taking the state of the art at the time from about 8 dB to more than 10 dB, increasing link margin. Polarization encoding and decoding, signal equalization, and digital clock recovery followed. Today, all commercial metro and long distance communications make use of this advanced digital electronics to transmit up to 200 Gb/s using 16-QAM on a single wavelength on a 25 GHz carrier with 50 GHz channel spacing, yielding about 16 Tb/s on a single fiber with 80 wavelengths.

A few years earlier, electronic analog equalization chips allowed increased fidelity on backplane printed circuit board materials initially at 2.5 Gb/s, but steadily increasing to 10 – 25 Gb/s today. The same and similar equalization techniques could be applied to shorter reach optical communications, enabling faster link speeds with slower components,[52] increased link margin, and even direct modulation of semiconductor lasers at high bit rates.[53] The result has seen a steady increase to a per-channel bit rate of short reach interconnects today of 25 Gb/s, generally occurring as 4-fibers in each direction to implement the 100 GbE standard of today.

Interestingly, the Ethernet standards have always evolved in factors of 10, except for a brief flirtation with 40 GbE around the year 2007 – 2009. However, the power of digital signal processing (DSP) has eased the transition very quickly to 4 channels at 25 Gb/s, which in some ways is easier than the 40 Gb/s single lane before that. An interesting note is that the 100 GbE standard should have been implemented in 2010 (10 GbE was 2000, GbE was 1990), but like the electronic Moore's law, optical transceiver progress has slowed a bit. The next step in standards committees beyond 100 GbE is 400 GbE, similar to the intermediate step taken in the mid-late 2000s, and product announcements are expected in 2018 – 2020. Incidentally, the step beyond that seems to be 1.6 Tb/s, not 1 Tb/s, as the factor of 4 in each iteration better satisfies the insatiable bandwidth demand of data centers.

For some time, four fibers in each direction bundled as one has had some tractability, going back to early implementations of 10 GbE as 4 channels of 2.5 Gb/s, using the MTP multifiber connectors and ribbon cables, multiplied by 10 to 4 channels at 25 Gb/s for 100 GbE today. In VCSEL multimode interconnects, the products have, except for some small volume outliers) used parallel fibers versus polarization or wavelength multiplexing. For silicon photonics, in addition to multi-fiber systems, a trend has been to use 4-wavelengths with wider channel spacing, the so-called coarse wavelength division multiplexing (CWDM).

For higher bit-rates, to increase the bit rate, but not the clock rate, the trend is to use pulse amplitude modulation (4 signal levels versus 2) with quite extensive error correction and equalization to enable link margin in this reduced SNR data format, implementing 400 GbE as 8

fibers or CWDM wavelengths at 56 Gb/s, but still with a 28 GHz clock rate, the added overhead from 64b/66b data encoding for error correction. Error correction is required on a 4-PAM link, but not on an on-off keyed link at the same clock rate (but ½ the bit rate). A couple demonstrations at 56 GHz clock rates enables 100 Gb/s per fiber with only a single wavelength, without requiring the more complex coherent communications that require a receive laser in the long distance and metro networks. It's easy to imagine with 10 CWDM wavelengths or fibers, one can achieve 1,000 GbE (or 1 TbE) links per fiber without resorting to dense wavelength division multiplexing or coherent communications. Other modulation schemes besides 4-PAM have been proposed,[54] such as duobinary, optical frequency division multiplexing (OFDM), carrierless amplitude phase modulation (CAP); these all have some merit.

However, the trend in increasing the symbol rate (b/s/Hz) using advanced electronic signal processing has two detrimental effects. First, the use of FEC increases the latency of the link from 10s of ns to 100 ns or so. This is likely fine for data centers where the data center computing latency is not as demanding as for example in a purposely built high performance computing machine. Second, the power used by FEC and a heavy dose of electronic equalization is high compared what additional links or wavelengths use. In spite of this, we believe that the trend will continue to add more and more electronics to avoid additional fibers or wavelengths. ***That is contrary to what the HPC machines need, but in the interest of providing the lowest cost solutions to the data center market, we should expect it to continue.***

To gain additional insight into why, today's large switching chips have 64 ports at 100 Gb/s. The IO is implemented as 256 IO at 25 Gb/s. Since a port has both transmit and receive channels and differential signaling, these chips have over 1,000 data pins over 2,000 pins including power and ground! It is not likely that the pin-count will continue to grow substantially (say by 2×, 4×), because the assembly infrastructure cannot handle smaller pins. So, the trend in chips is to increase data rate per pin even higher. Electronics designers can imagine IO using PAM-4 at 56 Gbaud (112 Gb/s) per pin for a total IO of 25.6 Tb/s per chip. However, at that rate and with such a poor SNR, the electronics required per pin for FEC and DSP equalization is staggering and the interconnection length is limited to significantly less than a foot, because of the increased loss at higher frequencies. The next step beyond this might be where intimate optics is needed in the package, even for the data center market. When that solution becomes a reality, it makes sense to abandon multilevel signaling in the interest of removing the FEC and DSP functions, saving power and reducing latency, both detriments to high performance computing.

Off-roadmap solutions

The alternate approach is to provide many channels either through multicore fiber ribbons (say 8 fibers with 5 cores each) or dense wavelength division multiplexing, where 40 channels fit in the telecommunications C-band. Technologically, the two require different breakthroughs. In the first case with many completely independent channels, an advancement in fiber manufacturing is the key challenge to keep the fiber-cost low. ***In the second case, there are two principal challenges, first, the control of silicon photonics variations over temperature and manufacture and second, the implementation of a DWDM laser subsystem with a cost-point not too much greater than a single laser.*** We outlined these challenges for the case of silicon photonics micro-

resonators.[55] While these approaches are the subject of quite a bit of research, neither is really trending today toward products that vendors can use in high performance computing.

Before moving on, it is instructive to say a few words about silicon photonics microresonators as filters (WDM mux/demux) and modulators. To our knowledge, no products use them, owing to their wavelength sensitivity with manufacturing and environmental (mainly temperature) variations. However, the circuitry required for closed loop control of resonant wavelengths is simpler than, but not too dissimilar from, that required to bias a VCSEL. Recently closed loop control of multiple devices implementing DWDM on a common bus[56, 57] as well as control of higher-order filters[58] has been demonstrated, although not at the same time. If we can build on this work, 40 wavelengths of silicon photonics devices are compact, low power, and completely feasible. Because telecommunications C-band is a consideration in amplified systems, going beyond that to 100 nm optical bandwidth systems may be feasible, enabling the technology to satisfy 4 TbE in a single fiber at 25 Gb/s serial data rates, with 10 TbE feasible making use of increased bit rate, dual polarization systems, and closer channel spacing. Given laboratory demonstrations have topped out at 8 – 16 wavelength channels, it is a bit absurd to consider 200 wavelength systems. Comb lasers, consisting of a single pump laser and a non-linear microresonator with channel spacing given by the free spectral range of the comb resonator can in theory implement a cheap laser with many channels that doesn't cost too much more than a single laser.

Optical transceivers directly integrated with high-value electronics

Today's model of optical interconnects connects optical transceivers to high-value electronics through printed circuit board traces. This has a few constraints. First, the bandwidth is limited by the IO rate of the electrical traces and the number of IOs is limited by the pitch and the fact that electrical IO is differential and requires a fair number of powers and grounds owing to the high current in the low voltage devices. Today, that number is about 6 Tb/s (256 ports of 25 Gb/s), as we mentioned earlier, it's easy to imagine electrical lines going to PAM-4 at 50 or 56 clock rates for 100 Gb/s per differential pair giving upwards of 25 Tb/s IO electrical bandwidth, albeit at quite high power consumption. It's difficult to imagine going much beyond it, because printed circuit board technology is not likely to significantly increase the pin count, and the SNR constraints will limit the data rate per pin from greatly exceeding 100 Gb/s. A reasonable limit in the 10-year time frame is probably in the 25 – 100 Tb/s range.

Intimately integrated optics with electronics has the advantage of eliminating the requirement to equalize the transmission channel (i. e. backplane trace) and allowing the electrical drive circuit to drive a low capacitance, high impedance load versus a low impedance 50-ohm line. Silicon photonics micro-resonators can be directly driven differentially at voltages comparable to those at the end of the CMOS end (~ 0.5 V). This difference was elegantly stated by Miller[59] as a quantum impedance transform; that is the conversion from electrical signals to photons makes the impedance of the transmission medium irrelevant, avoiding the relatively high power consumption of driving 50 ohms for both transmit and receive. The consequence is drastically lower power at the transmitter and receiver.

The effect of drastically reducing the capacitance and eliminating 50 ohm lines is greater on the receiver. Drastically reducing the capacitance of the photodiode itself and connection to electronics to the few femtofarad range allows higher values of transimpedance in the receiver. This, in turn, enables lower noise floor, lower laser power, elimination of AC-coupling, threshold balance compensation and limiting amplifiers in the receiver, and of course the elimination of line equalization to overcome the large frequency dependent loss in an electrical line. Hence, the result again is drastically reduced power consumption, with increased link margin at the same time.

Today, this is well known in the interconnect research community, but few demonstrations of the receiver performance have been made that exceed the sensitivities of commercial receivers. Perhaps top on the reasons why not is that no one has really tried to reduce capacitance to single femtofarad levels and increasing the transimpedance to the high levels that enable these sensitive receivers. The best demonstrations point to sub-pJ scale receivers, including clock recovery,[60] which is a couple orders of magnitude below current data communications optical receivers. But, the intimate integration with electronics should lead to receiver energies in the 1 – 10 fJ/bit range, excepted of clock recovery.

While this has been proposed in the research community for decades,[59] it hasn't made it to products. There are many reasons for this. These are:

- Electrical signaling rates have increased dramatically allowing IO to 'keep up' with internal capability. That's likely to continue to increase to above 100 Gb/s/IO using multilevel signaling such as PAM-4 or PAM-8.
- The design infrastructure for optical transceiver design and electrical chip design are not the same. That's changing as Mentor, Cadence and other vendors incorporate design environments for silicon photonics, but not highly integrated (in design software) with fine line CMOS and heterogeneous integration of the two.
- Packaging something like an Ethernet switch chip is different from a transceiver. It's not likely optics can beat electrical signaling rates, so to get more bandwidth off a chip optically, we need significantly more IO. So, for example, 4,000 IO at 25 Gb/s is 100 Tb/s. With 40 wavelengths, that's 100 channels or fibers. There is a product with a 64 IO optical connector (Molex MXC), but that is likely not quite compatible with a high power large chip. So, likely a custom package would need to be developed.
- Cost of optics is a big issue; a lot of it is packaging, but the cost of the lasers as well.

In summary, if we had to guess, we wouldn't expect optics intimately integrated with high-value electronic ICs (processing, networking, memory chips) to materialize until Industry really hits a power wall (not wishful researchers who base their research on it) or when the aggregate rate of electrical interfaces hits a wall where increases become extremely costly. Our opinion is that we're probably about a single order of magnitude away from that, but it may be a decade.

Optical media: fibers

Today, virtually all optical interconnects use fiber as the interconnection medium. Multimode is dominant with VCSELs as the sources, owing to the low cost and simple alignment of the VCSEL with the multimode optical fiber. Single mode is growing for use in data centers,

perhaps for a couple reasons. First, the reach is longer, so hyperscale data centers can use a single fiber type for all reaches. Second, as data rates increase from 100 Gb/s to 400 Gb/s and to 1.6 Tb/s, single mode systems have advantages in the ease of multiplexing multiple logical channels on a single fiber using wavelengths or complex signaling. That is, single mode is viewed as future proof.

A reasonable assumption in a BMC environment is that the off-chip bandwidth requirements will continue to grow, if not accelerate. In terms of numbers, a reasonable target to start the discussion is 100 Tb/s, with growth required to 1 Pb/s or beyond. Why? Electronic networking chips today have 64 logical ports at 100 Gb/s or 6.4 Tb/s of aggregate IO. This is implemented as 256 IO at 25 Gb/s. We discussed earlier, the progression of aggregate data rates to 25 Tb/s – 100 Tb/s per chip, before integrated optics directly with CMOS. In a beyond Moore's technology, optics requires either more channels or higher rates. The latter is difficult. In fact, there are advantages to operating at inherent chip clock rates if we can cost effectively provide fundamental clock rate interconnects. While an end of roadmap processor or networking chip might operate at 2 – 5 Gb/s, a BMC chip may operate much faster, owing to the much lower voltage. Hence, we may get higher data rates off chips without an extremely high serial rate multiplexing.

The fundamental bandwidth of fiber is quite high. In single mode fiber, there are two main degrees of freedom, polarization and wavelength. Today, local interconnect products are limited to 4 wavelengths, with potentially 8 around the corner. Yet, long distance networks typically have 80. Long distance networks are limited by the bandwidth of an EDFA, but local interconnects really aren't very limited at all. In fact, if we could use channels with 50 GHz optical channel spacing, we could fit many hundred channels in the bandwidth of the fiber. At data rates of 100 Gb/s/channel, a path to 10 Tb/s per fiber or single mode waveguide could be envisioned. With 1,000 IO, ***we can project to 10 Pb/s of optical IO per chip***.

Multimode has more limitations with respect to wavelengths. It is more difficult to provide filtering functions in multimode, because the modal dispersion doesn't map into a simple optical filtering system. However, achieving 8 – 10 wavelengths appears feasible. But, it's not likely to be a BMC solution.

Mode multiplexing and multi-core fiber have been the subject of quite a bit of research lately in the desire to add more bandwidth to a single fiber infrastructure. Multicore fiber is a simpler concept, where the signals on the individual cores have low crosstalk and can be considered effectively as separate fibers. Examples include Chiral's PROFA fiber[61] with fiber outputs with 12 micrometer linear spacing vs. 127 um or 250 um for standard fibers. Such fibers have not found widespread use yet. It's a bit of a catch-22, because they are undoubtedly more expensive than standard fiber ribbons; however, volume is a key cost factor in any comparison. Nonetheless, it is an approach that can increase the density of interconnect quite readily.

Mode mixing will occur if there is coupling between cores of a multicore fiber. In this case, there must be a mode-demultiplexing approach to separate out the mixed channels. In long distance communications, this has been the subject of quite a lot of research. However, in short reach interconnects, the signal processing required to deconstruct the mode mixing is likely be viewed

as prohibitive. A real breakthrough would be an optical device to perform the mode deconstruction. Indeed, such a device has been proposed,[62] but how we would apply this to a 2D system is not clear. Even harder than a multicore fiber with coupling between cores (few mode-system) is a full multimode fiber. If we could utilize the full number of modes supported in the fiber, it would have a really large capacity. This is a long way off.

Optical printed circuit boards

Optical PCBs were common in the late 90s for telecom gear to simplify front panel routing. However, the cost of such elements was somewhat prohibitive, and they largely disappeared following the telecom meltdown of the 2000s. However, research has continued at a slow pace for quite some time. The main advantage of an optical PCB is packaging and cost relative to fiber. By routing optics like electronics, one can envision costs approaching electronics. However, challenges remain such as reducing loss and reducing cost. The loss of single mode is indeed quite difficult to reduce for long distances, yet single mode has the same advantage as single mode fibers relative to multimode fibers. It may be more feasible to use a multimode system without getting a strong degree of mode-coupling owing to the fact that the infrastructure is rigid compared to fibers.

While there have been 20+ years of demonstrations of optical interconnects on printed circuit boards, there is no low cost commercial infrastructure to do this. The ability of electrical interconnects to supply the needed connectivity bandwidth has kept this infrastructure from further developing.

Most of the demonstrations of optical PCBs have focused on 50 um polymer guides for interfacing to VCSELs. Because VCSEL data rates don't exceed that of electrical connections and WDM is rare, there is a weak case for developing multimode optical printed circuit interconnects because the bandwidth density doesn't exceed that of electrical interconnects by much. The power dissipation of multimode receivers is also too high because of the large photodiode size required to collect multimode light.

That said, a single mode optical waveguide PCB would have definite advantages in power and density compared to electrical interconnects, the latter from closer spacing of single mode connections and WDM placing more bandwidth per connection compared to electrical interconnect. There has been a substantial amount of work on single mode PCB connections over the last 30 years when taken collectively.[63] There were products with single mode optical PCBs in the late 90s and early 2000s (e. g. Ciena's Core Stream) but many of these PCBs were ripped out because the cost exceeded that of simple optical patch chords.

The advent of optical PCBs are likely to usher in a few changes to the optical interconnect world. One of these is easier implementation of differential optical connections which can potentially eliminate data encoding and DC balance requirements as in electrical LVDS interconnects (albeit not 25 Gb/s ones).

Free space optical interconnects

There's a good argument for interconnecting at the fundamental data rate of the chip as opposed to a high degree of time multiplexing.[64] As an example, a 100 Tb/s interconnect at 4 Gb/s requires 25,000 independent connections. This might be handled with wavelengths and routing paths, say 100 wavelengths, 2 polarizations, and 125 paths. Now, let's consider an interconnection with 10 Pb/s. This may become intractable using 12,500 spatial guided wave paths without time multiplexing.

However, free space suffers from physical constraints relative to guide wave systems. While guided wave optics can implement arbitrary topologies, free space optics ones are rather constrained in their ability to route signals from *n* different locations to *m* different locations – even though all-to-all matrices can be implemented rather simply in lenses. How would we for example interconnect an entire data center using free space optics? Over what distance, does free space optics make sense? **There is almost no work today in this area. It's worth investigating the scale and applicability of such systems.**

On-chip interconnects: an opinion

Currently, there are a core group of researchers that propose on-chip interconnects using optics. This aims to solve two problems with electrical interconnects; first the limited bandwidth of thin electrical wires traversing mm-scale distances and second the energy consumption in those interconnects which is ¼ CV^2 for a data line and CV^2 for a clock line. The capacitance of a line is typically in the range of 1 pF/cm, at 0.5V signaling, the energy of a 1 mm – 1 cm interconnect is 25 – 250 fJ/bit for a clock line and 6.25 – 62.5 fJ/bit for a (random) data line. (A data line has ¼ of the energy of a clock line, because it has ¼ as many transitions for random data) As long lines will either need equalization or repeaters, the actual numbers are likely higher. So, optics has an opportunity here. But, the principal impediment is cost; it's hard to see optics being as cost effective as an on-chip trace. However, in a system with BMC transistors, the power consumption of electrical interconnects is drastically reduced as the supply voltage is reduced. **In a chip utilizing low-voltage post-CMOS transistors, the need for on-chip optics is likely to be reduced, not increased.** However, interfacing optical modulators directly to low voltage transistors is a challenge without the use of amplifiers.

Optical interconnects for BMC transistors

Since about 2004, clock speed has plateaued in highly integrated CMOS circuits, primarily from power consumption. Although resistance in lines is a limiting factor, the availability of high-bandwidth serialized interfaces at 25 Gb/s shows that speed is not an intrinsic limit. As technology nodes have scaled since then, circuits have become more densely packed with an increasing number of cores per processor chip and increased DRAM sizes per chip. Performance improvements have come about through more cores, more memory, and architectural changes versus increased memory bandwidth, network bandwidth, or clock speed. At the 'end' of Moore's law, size no longer scales, putting an end to the first two reasons for improved performance. Interestingly, most implementations of beyond Moore's transistors implementing digital logic attempt to solve the limit in clock speed by lowering the voltage, reducing the dynamic dissipation, CV^2, while at the same time, reducing leakage currents, essentially solving

the problems that have plagued CMOS in the last decade. Proposals exist to reduce voltages from a fraction of a volt to a few millivolts; of course, there's little success to date in doing so.

It is instructive to consider how to interface optical modulation devices and receivers to these low voltage logic devices. Today's modulation devices require at least a fraction of a volt for a decent optical modulation output.[65] Therefore, we must amplify the output signals from millivolt to hundred millivolt levels. There has not been much research into low power voltage amplifiers for this application, so there are no data points for how much power such a driver would consume. A lower power solution might be obtained by developing a low voltage optical output device (e. g. a modulator) that can interface directly with low voltage (millivolt) logic. This area of research has very little emphasis right now.

On the receiver side, today's transimpedance amplifiers amplify signals from millivolt to volt levels. A solution exists to that; if we drastically reduce the capacitance from 100s of fF to 1 fF values, the transimpedance can be drastically increased (from ~ 1k to > 100k), allowing receiverless or single stage transimpedance amplifiers with drastically reduced power dissipation, performing the amplification in a single stage. In the limit, the optical signals can switch logic levels directly.[66] In a BMC world, this problem becomes even easier, because the required output of the receiver is only at millivolt levels, thus even less optical power is required to reach logic levels. Indeed, with low enough capacitance at low enough voltages, receivers become shot noise limited, with complete data links below 1 fF being feasible, pending the invention of an appropriate modulator device.[67]

Today, the QCSE is about the strongest effect and it requires on the order of a few V/um in electric field change to effectively modulate light for a reasonable optical modulation output. To efficiently modulate light at millivolt levels, we either need new materials with drastic improvements in electro-optic coefficients or devices that operate with ultra-small dimensions that amplify the field with low voltages. There's likely limits to how well resonant devices can improve a weak effect (e. g. silicon photonics resonant modulators), although like CMOS, that technology can be pushed rather hard because the materials are easy.

Electrically controlled optical switching

Today's primary application of optical switching is in the metropolitan networks where optical reconfigurable add drop multiplexers provision network bandwidth among multiple paths. There is quite a lot of interest now in optical switching for data center applications. The reason is to reduce expensive and power hungry OEO conversions as well as increase throughput from generation to generation without requiring an upgrade to the switch fabric.

Optical switches can be categorized as broadband and wavelength selective. Broadband switches switch multiple wavelength channels concurrently; wavelength selective switches allow one to route wavelengths in a DWDM system independently. Nearly all the switches in the metropolitan area network use wavelength selective switches to set up routes for individual wavelengths. Both applications exist in data centers and computing, because there may be a need for finer grain switching than a whole 400 Gb/s or larger stream.

Different technologies of optical switches have different features, **but no one technology is ideal for future BMC networks**. MEMs/free-space optical switches can scale to large sizes, but their relatively long reconfiguration time makes them less suitable for switching short packets in a high-performance computing system. Integrated MEMs switches[68] have similar characteristics and tradeoff loss and scalability for smaller physical size. Silicon photonics Mach-Zehnder switches can be made with fast (< 1 ns) or slow (> 1 μs) reconfiguration rates, but scalability is an issue in large networks comprised of many 2 × 2 switches. The faster ones are physically larger and have higher loss. These switches are broadband. Ring resonator switches can also be made fast or slow, they are physically more scalable from their small size, but loss is still often a limiting factor, not to mention wavelength stabilization. Scaling has been theoretically studied to 128 ports.[69] These can be either wavelength selective or broadband in theory.

There aren't really any current other technologies being considered. **A missing link is an electrically controlled optical switch that is integrated with small size, high speed, low loss, broadband, and non-resonant that is robust over environmental conditions and manufacturing tolerances.**

Today, both HPC systems and data centers use packet switches. Packet switches have three important components. These are (1) buffering of data, for example if inputs 1 and 2 both want to transmit data to port 3 at the same time (2) a control plane that includes determination of routing paths through the switch or network of multiple switches, and the arbitration technique that determines the dynamic order of priority among which inputs pass through the switch or network and (3) the physical transport of data through the switch or network itself. An electrically controlled optical switch only addresses the third item. While very high bandwidth can be supported optically, for HPC networks which often consist of small messages, it isn't obvious that even a holy-grail optical switch that satisfies all the requirements (low loss and crosstalk, fast reconfiguration, low power, scalable to large sizes) will make much of a dent in extending the range of packet switches beyond what is achievable with electronic switching.

Optical packet switching

Optical packet switching is really a subset of digital optical logic, but easier cost, power, and scalability targets compared to digital optical logic. Most work on optical packet switching has involved the use of semiconductor optical amplifiers as logic gates to decode the packet header information[70] This approach is very power hungry. Packet buffering has been implemented in a variety of technologies.[71] Both the logic and buffering are cumbersome compared their commercial electronic alternatives, although even though SOAs have high power dissipation, the low energy of using a fiber as a buffer make the approach potentially more power efficient than today's electronic packet switching. Very little work is published in the research community today on optical packet switching.

Thus, the research community is primarily focused on optical interconnect and electrically controlled optical switching. Compounding the difficulty achieving low energy logic, for the most part, optical logic tends not to have the desirable characteristics of CMOS.[72] Several years ago, the research community tended to evolve into electronics for logic and optics for

communications. It is worth revisiting optical computing, in light of recent advancements in silicon photonics.

Optical Computing

Digital logic

It was observed many years ago that light tends not to interact strongly with materials, thus the elements for logic tend to be weak. For more than 30 years, there has been some small effort in digital logic. In the late 80s, the most promising devices were optoelectronic,[71] but it became clear to those in the optical computing research community by the early 90s that electronic computation with optical communications was a more likely outcome. Hence conferences, such as the OSA's optical computing conference were abandoned in favor of optical switching and optical interconnects. One interesting approach to digital logic that makes use of integrated quantum optics is described below.

Quantum optical logic gates can be made using microring resonators to generate entangled photons.[73] However, you cannot perform logic directly with single photons at an acceptable error rate, because of their probabilistic nature (for example, QKD uses a very large overhead to do this). Using coherent quantum optical feedback, a device can be stabilized to have known states (with a certain error probability), yielding circuits that comprise quantum optical devices that can perform logic at very low energies ($\ll$1 fJ/bit), albeit at much larger energies than that of a single photon.[74] A research quantum hardware descriptor language (QHDL)[74] has been developed that allows researchers to construct circuits using these techniques. The work is still in its infancy, but shows promise for a class of integrated optical devices and circuits with low energies, something completely lacking in previous implementations of digital optical logic. But, whether this can cost-effectively scale to perform a useful function better than CMOS remains to be seen.

Vector-matrix multiplication and neural-inspired computing

In computation, the proposed use of analog optical multiplication has been around for years dating back to systems using spatial light modulators in free space optics,[75] but more recently with silicon photonics (see e. g. Watts[76] and Shen[77]).

Vector matrix multipliers require {C} = {B} {A} require a linear vector A, a matrix B and the result is a linear matrix {C}. In an original implementation, an optical linear array of fiber outputs along a column (vertical) was fanned out horizontally and imaged through a 2-D spatial light modulator that applies weights to the different inputs. The outputs of the spatial light modulator are summed along the vertical dimension by fanning the outputs into a linear horizontal array to input fibers and ultimately detectors that perform the sum. The main problem (among many) is that this architecture has an N^2 loss to image to a single mode small detector or an N loss if we can make use of spatial position on the output by using a large detector or a multimode fiber. An approach in silicon photonics using wavelength division multiplexing uses an array of modulators, cascaded through a 2D array of filters and uses a WDM mux to provide the output. This still has the N loss term, but at least not N^2. A new approach[77] using an arbitrary

linear transformation is more promising, because it provides a theoretically lossless multiplication; the result is an optical architecture that can potentially be more energy efficient than its electronic counterpart.

Like their communications counterparts, optical analog multipliers require an increase in signal to noise ratio as the number of analog levels increases. The sensitivity limit at the summing detector is increased dramatically for these systems compared to a binary signal, because there is a background level that needs to be overcome. For example, if we are to distinguish between two adjacent levels in an 8-bit analog signal, say from 254 to 255 at the high end of the range, then the background shot noise is 254 times larger than the 'modulation' on the signal. Hence the shot-noise limit to the computation increases dramatically compared to individual binary logic systems. However, in a neural inspired computing implementation, the SNR requirements are not likely as high as in a digital system; hence that improves the minimum detectable signal on the receiver photodiode(s).

However, the analog matrix multiplier above,[77] shows the potential for drastically improved performance compared to its electronic counterpart. This is a good example of where a major architectural change can lead to a performance enhancement, greater than that of developing a BMC transistor with existing architectures. However, the technology assumptions that lead to the performance enhancement may or may not be realized.

There are other optical computing approaches that have been and are currently being pursued. Given the advancements in silicon photonics and integrated optics in the last 20 years, it's worth taking an updated look at optical computing to see if new technology can make it become practical in providing co-processing functions now, where it wasn't in the past.

Discussion and recommendations: optics

The maturation of integrated optics, principally, but not exclusively silicon photonics, enables cost/performance points in the use of optics in computing that will become more and more compelling. Over the last decade, optics has firmly moved beyond the telecommunications space where it is now the go-to interconnect between racks of equipment in high performance data communications and computing systems. This 'shortening' of the distance where optics will play a role will continue as will increased functionality out of that optics. This will first be transceivers within racks, transceivers on chips, optical switching and lastly potentially optical computing. Below we summarize the progression and highlight a few key challenges where applicable.

Discrete optical transceivers: There isn't a technology out there that we know about that can bring about a fundamental breakthrough for discrete transceivers. Even removing FEC and electronic digital signal processing equalization to overcome limitations in the SNR of multilevel modulation may not impact the power consumption or cost much, because those components are likely to be needed to overcome limitations in the electrical links between high-value CMOS and the optical transceivers. In long run, we need intimate integration of the two.

Optical transceivers intimately integrated with high-value CMOS: This approach should look to scale to 100 Tb/s - 1 Pb/s aggregate IO near the end of the roadmap. Because per-channel data rates will be below 100 Gb/s, and advantageously closer to 10 Gb/s, 1k – 100k parallel channels may be needed. DWDM, perhaps with 50 - 100 channels per fiber, is the nearest term solution to enabling a drastic reduction of the number of physical channels. If we had to identify one key missing link, it would be the laser multi-wavelength source. For DWDM to ever permeate the data center, the cost of these 50 – 100 lasers must be a little more (say 1.5×) than the cost of a single laser. Otherwise, people will continue to search for solutions that require extensive electronics to minimize the number of wavelength channels as much as possible

Optical Switching The degree to which optical switching makes sense in a data-com or HPC environment depends greatly on the characteristics of the technology. Optical switches have high data path bandwidth and dissipate no power when passing data (other than loss), but depending on technology, they have one or more limitations in physical size (too big), reconfiguration time (too slow), crosstalk (too high), loss (too much), network size scalability (too small), device manufacturing and environmental variability (too much) and cost. Developing a switch that can meet all these goals is a holy grail.

Optical Packet Switching: The control plane and buffering side of the equation needs a scalable approach to the buffering problem. Because optical logic is in its infancy (and has been for some time), it's difficult to even compare an all-optical router to an electrical one (or electrically controlled optical one). Currently, it's hard to define an area where a little investment could produce a profound effect.

Optical Computing: The new approach implementing deep learning[77] is promising on the surface in that it solves a problem in loss that optics previously had no solution for. While Shen[77] performed a comparison based on energy per bit, a scalability analysis can determine if the optical solution, even if it's more energy efficient, can scale to the complexity of an electronic one. On the other hand, there may be cases where you want a faster answer to a "small" problem. Digital optical computing is worth thinking about, but is likely to have a longer time horizon.

References

1. Moore, G. E. Cramming more components onto integrated circuits. Electronics, 38 (8), April 1965. *VLSI Technol. Archit.* (2010).
2. IRDS - IEEE International Roadmap for Devices and Systems. Available at: http://irds.ieee.org/. (Accessed: 28th July 2017)
3. DeBenedictis, E. P. Meeting minutes, Emerging Research Device Working Group FxF meeting, Koenigswinter (Bonn), Germany April 2-4, 2008. (2008).
4. Nikonov, D. E. & Young, I. A. Overview of beyond-CMOS devices and a uniform methodology for their benchmarking. *Proc. IEEE* **101,** 2498–2533 (2013).
5. Obama, Barak. 'Executive Order—Creating a National Strategic Computing Initiative.' - Google Search. Available at: https://obamawhitehouse.archives.gov/the-press-office/2015/07/29/executive-order-creating-national-strategic-computing-initiative. (Accessed: 28th July 2017)
6. DeBenedictis, E. P. Rebooting Computers as Learning Machines. *Computer* **49,** 84–87 (2016).
7. DeBenedictis, E. P. Sustaining Moore's Law with 3D Chips. *Computer* **50,** (2017).
8. DeBenedictis, E. P. The Boolean logic tax. *Computer* **49,** 79–82 (2016).
9. DeBenedictis, E. P. Computational Complexity and New Computing Approaches. *Computer* **49,** 76–79 (2016).
10. DeBenedictis, E. P., Mee, J. K. & Frank, M. P. The Opportunities and Controversies of Reversible Computing. *Computer* **50,** 76–80 (2017).
11. Silver, D. *et al.* Mastering the game of Go with deep neural networks and tree search. *Nature* **529,** 484–489 (2016).

12. Administration, N. H. T. S. & others. Federal Automated Vehicles Policy: Accelerating the Next Revolution in Roadway Safety. *Wash. DC US Dep. Transp.* (2016).
13. Landauer, R. Irreversibility and heat generation in the computing process. *IBM J. Res. Dev.* **5,** 183–191 (1961).
14. Joneckis, L., Koester, D. & Alspector, J. *An Initial Look at Alternative Computing Technologies for the Intelligence Community*. (INSTITUTE FOR DEFENSE ANALYSES ALEXANDRIA VA, 2014).
15. Frank, D. "Reversible adiabatic classical computation—an overview".
16. Pan, C. & Naeemi, A. A proposal for energy-efficient cellular neural network based on spintronic devices. *IEEE Trans. Nanotechnol.* **15,** 820–827 (2016).
17. Standard, J. High bandwidth memory (HBM) DRAM. *JESD235* (2013).
18. Consortium, H. M. C. & others. Hybrid memory cube specification 1.0. *Last Revis. Jan* (2013).
19. Zhu, Q. *et al.* A 3D-stacked logic-in-memory accelerator for application-specific data intensive computing. in *3D Systems Integration Conference (3DIC), 2013 IEEE International* 1–7 (IEEE, 2013).
20. Zhu, Q., Graf, T., Sumbul, H. E., Pileggi, L. & Franchetti, F. Accelerating sparse matrix-matrix multiplication with 3D-stacked logic-in-memory hardware. in *High Performance Extreme Computing Conference (HPEC), 2013 IEEE* 1–6 (IEEE, 2013).
21. Azarkhish, E., Rossi, D., Loi, I. & Benini, L. Design and evaluation of a processing-in-memory architecture for the smart memory cube. in *International Conference on Architecture of Computing Systems* 19–31 (Springer, 2016).

22. Vermij, E., Fiorin, L., Hagleitner, C. & Bertels, K. Sorting big data on heterogeneous near-data processing systems. in *Proceedings of the Computing Frontiers Conference* 349–354 (ACM, 2017).

23. Ahn, J., Hong, S., Yoo, S., Mutlu, O. & Choi, K. A scalable processing-in-memory accelerator for parallel graph processing. in *Computer Architecture (ISCA), 2015 ACM/IEEE 42nd Annual International Symposium on* 105–117 (IEEE, 2015).

24. Dysart, T. *et al.* Highly scalable near memory processing with migrating threads on the emu system architecture. in *Irregular Applications: Architecture and Algorithms (IA3), Workshop on* 2–9 (IEEE, 2016).

25. Chang, J. K.-T., Liu, C. & Gaudiot, J.-L. Enhancement for Potential Target in Cryptography Algorithms by Applying Processor-in-Memory Architecture. in *Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), 2013 IEEE 27th International* 2035–2044 (IEEE, 2013).

26. Loh, G. H. Three-Dimensional Microprocessor Design. in *Three Dimensional Integrated Circuit Design* 161–188 (Springer, 2010).

27. Akin, B., Franchetti, F. & Hoe, J. C. Understanding the design space of dram-optimized hardware FFT accelerators. in *Application-specific Systems, Architectures and Processors (ASAP), 2014 IEEE 25th International Conference on* 248–255 (IEEE, 2014).

28. Aly, M. M. S. *et al.* Energy-efficient abundant-data computing: The N3XT 1,000x. *Computer* **48,** 24–33 (2015).

29. Boahen, K. A Neuromorph's Prospectus. *Comput. Sci. Eng.* **19,** 14–28 (2017).

30. Suhas Kumar, Strachan, J. P. & Williams, R. S. Chaotic Dynamics in Nanoscale NbO2 Mott Memristors for non-Boolean Computing. *Nature* 318–321 (2017). doi:10.1038/nature23307

31. Sheffer, H. M. A set of five independent postulates for Boolean algebras, with application to logical constants. *Trans. Am. Math. Soc.* **14,** 481–488 (1913).

32. Wernick, W. Complete sets of logical functions. *Trans. Am. Math. Soc.* **51,** 117–132 (1942).

33. Hopfield, J. J. Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci.* **79,** 2554–2558 (1982).

34. Li, Y. A New Exact Algorithm for Traveling Salesman Problem with Time Complexity Interval (O (n 4), O (n 3* 2 n)). *ArXiv Prepr. ArXiv14122437* (2014).

35. Isaacson, W. *The Innovators: How a Group of Hackers, Geniuses, and Geeks Created the Digital Revolution.(2014)*. (Simon & Schuster, New York, NY).

36. Feynman, R. P. Simulating physics with computers. *Int. J. Theor. Phys.* **21,** 467–488 (1982).

37. Shor, P. W. Algorithms for quantum computation: Discrete logarithms and factoring. in *Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on* 124–134 (Ieee, 1994).

38. Hinton, G. E., Osindero, S. & Teh, Y.-W. A fast learning algorithm for deep belief nets. *Neural Comput.* **18,** 1527–1554 (2006).

39. Kepner, J. & Gilbert, J. *Graph algorithms in the language of linear algebra*. (SIAM, 2011).

40. Shamir, A. & Tromer, E. Factoring Large Number with the TWIRL Device. in *Crypto* **3,** 1–26 (Springer, 2003).

41. Badaroglu, M. & Xu, J. Interconnect-aware device targeting from PPA perspective. in *Computer-Aided Design (ICCAD), 2016 IEEE/ACM International Conference on* 1–6 (IEEE, 2016).

42. Deveci, M., Trott, C. & Rajamanickam, S. Performance Portable Sparse Matrix-Matrix Multiplication on Intel Knights Landing and NVIDIA GPUs. *Unpubl. Sandia Doc.* (2016).

43. Tran, T. DARPA Hierarchical Identify Verify Exploit (HIVE).

44. DeBenedictis, E. Superstrider Associative Array Architecture. *Submiss. GraphChallenge*

45. Hoare, C. A. Quicksort. *Comput. J.* **5,** 10–16 (1962).

46. Batcher, K. E. Sorting networks and their applications. in *Proceedings of the April 30–May 2, 1968, spring joint computer conference* 307–314 (ACM, 1968).

47. Lakhani, S., Wang, Y., Milenković, A. & Milutinović, V. 2D matrix multiplication on a 3D systolic array. *Microelectron. J.* **27,** 11–22 (1996).

48. Lo, V., Mache, J. & Windisch, K. A comparative study of real workload traces and synthetic workload models for parallel job scheduling. in *Workshop on Job Scheduling Strategies for Parallel Processing* 25–46 (Springer, 1998).

49. Tanaka, H. *et al.* Bit cost scalable technology with punch and plug process for ultra high density flash memory. in *VLSI Technology, 2007 IEEE Symposium on* 14–15 (IEEE, 2007).

50. Xilinx Virtex Ultrascale+.

51. Intel/Altera Stratix 10 MX.

52. Kuchta, D. M. *et al.* 70+ Gb/s VCSEL-Based Multimode Fiber Links. in *Compound Semiconductor Integrated Circuit Symposium (CSICS), 2016 IEEE* 1–4 (IEEE, 2016).

53. Nakahara, K. *et al.* 28-Gb/s directly modulated InGaAlAs ACPM DFB lasers with high mask margin of 22% at 55° C. in *Optical Fiber Communication Conference* OTh4H–3 (Optical Society of America, 2013).

54. Wei, J., Cheng, Q., Penty, R. V., White, I. H. & Cunningham, D. G. 400 Gigabit Ethernet using advanced modulation formats: performance, complexity, and power dissipation. *IEEE Commun. Mag.* **53,** 182–189 (2015).

55. Lentine, A. L. & DeRose, C. T. Challenges in the implementation of dense wavelength division multiplexed (DWDM) optical interconnects using resonant silicon photonics. in *Proceedings of SPIE* **9772,** 977207 (2016).

56. Lee, D. Y. *et al.* Error-free operation of a polarization-insensitive 4λ x 25 Gbps silicon photonic WDM receiver with closed-loop thermal stabilization of Si microrings. *Opt. Express* **24,** 13204–13209 (2016).

57. Dong, P. *et al.* Simultaneous wavelength locking of microring modulator array with a single monitoring signal. *Opt. Express* **25,** 16040–16046 (2017).

58. Mak, J. C. *et al.* Automatic resonance alignment of high-order microring filters. *IEEE J. Quantum Electron.* **51,** 1–11 (2015).

59. Miller, D. A. Optics for low-energy communication inside digital processors: quantum detectors, sources, and modulators as efficient impedance converters. *Opt. Lett.* **14,** 146–148 (1989).

60. Zheng, X. *et al.* Ultra-efficient 10Gb/s hybrid integrated silicon photonic transmitter and receiver. *Opt. Express* **19,** 5172–5186 (2011).

61. Kopp, V. I. *et al.* Pitch reducing optical fiber array for dense optical interconnect. in *Avionics, Fiber-Optics and Photonics Technology Conference (AVFOP), 2012 IEEE* 48–49 (IEEE, 2012).

62. Miller, D. A. Perfect optics with imperfect components. *Optica* **2,** 747–750 (2015).

63. Dangel, R. *et al.* Polymer waveguides for electro-optical integration in data centers and high-performance computers. *Opt. Express* **23,** 4736–4750 (2015).

64. Miller, D. A. Attojoule optoelectronics for low-energy information processing and communications. *J. Light. Technol.* **35,** 346–396 (2017).

65. Timerdogan, E. *et al.* A one femtojoule 'athermal'modulator. *Nat. Commun.* **5,** (2014).

66. Debaes, C. *et al.* Receiver-less optical clock injection for clock distribution networks. *IEEE J. Sel. Top. Quantum Electron.* **9,** 400–409 (2003).

67. Lentine, A. L. & DeRose, C. T. Challenges for optical interconnect for beyond Moore's law computing. in *Rebooting Computing (ICRC), IEEE International Conference on* 1–5 (IEEE, 2016).

68. Seok, T. J., Quack, N., Han, S., Muller, R. S. & Wu, M. C. Large-scale broadband digital silicon photonic switches with vertical adiabatic couplers. *Optica* **3,** 64–70 (2016).

69. Cheng, Q., Bahadori, M., Rumley, S. & Bergman, K. Highly-scalable, low-crosstalk architecture for ring-based optical space switch fabrics. in *IEEE Optical Interconnects Conference (OI), 2017* 41–42 (IEEE, 2017).

70. Touch, J., Bannister, J., Suryaputra, S. & Willner, A. E. A design for an Internet router with a digital optical data plane. *Appl. Sci.* **7,** 143 (2017).

71. Burmeister, E. F., Blumenthal, D. J. & Bowers, J. E. A comparison of optical buffering technologies. *Opt. Switch. Netw.* **5,** 10–18 (2008).

72. Miller, D. A. Are optical transistors the logical next step? *Nat. Photonics* **4,** 3–5 (2010).

73. Silverstone, J. W., Bonneau, D., O'Brien, J. L. & Thompson, M. G. Silicon quantum photonics. *IEEE J. Sel. Top. Quantum Electron.* **22,** 390–402 (2016).

74. Tezak, N., Niederberger, A., Pavlichin, D. S., Sarma, G. & Mabuchi, H. Specification of photonic circuits using quantum hardware description language. *Phil Trans R Soc A* **370,** 5270–5290 (2012).

75. Goodman, J. W., Dias, A. R. & Woody, L. M. Fully parallel, high-speed incoherent optical method for performing discrete Fourier transforms. *Opt. Lett.* **2,** 1–3 (1978).

76. Watts, M. R. *Integrated optic vector-matrix multiplier*. (Google Patents, 2011).

77. Shen, Y. *et al.* Deep learning with coherent nanophotonic circuits. *Nat. Photonics* **11,** 441–447 (2017).

Page intentionally blank

DISTRIBUTION

1	MS0899	Technical Library	9536 (electronic copy)

Sandia National Laboratories