A report for

US National Science Foundation (NSF)

On

Sustainability in Computing

Based on

NSF Sponsored Workshop Series on Sustainability in Computing, 2021-2023

Web: https://nsf-suscomp.org/

Acknowledgments

This workshop was sponsored by the National Science Foundation CISE/CSR Division under grant CSR-2126011. We thank the NSF Program Director, Dr. Alex Jones, for the support of this workshop and for providing valuable feedback to the drafts of this report. We are grateful to all the workshop speakers, panelists, and roundtable participants for their insightful and stimulating presentations and discussions. Many of the participants have directly contributed to the writing of this report. The workshop program and a complete list of the speakers, panelists, and roundtable participants are provided in the appendix.

Sincerely,

Amlan Ganguly, Sudeep Pasricha, Massoud Pedram, Fabrizio Lombardi, Wuchun Feng
Organizers
NSF Sponsored Workshop Series on Sustainability in Computing, 2021-2023

* For any questions or comments, please write to Amlan Ganguly at axgeec@rit.edu.

Contents

Chapter 1: Executive Summary

● Introduction

Sponsored by NSF funding a series of workshops and activities were conducted to bring experts together to brainstorm and discuss open problems and challenges in making computing more sustainable in the broadest sense of the term. It has emerged that in order to improve sustainability of computing, researchers and practitioners must consider not only operational phase energy or carbon footprint but also that of the design and manufacturing phase. Dramatic increase of accelerators and heterogeneous platforms utilizing novel technologies has their impact on the manufacturing phase and therefore traditional technologies when used in novel ways to achieve similar performance as novel technologies will become more important in the future. Through the series of activities, open challenges in all levels of the computing hierarchy such as application development, system design, computer architectures, components such as Integrated Circuits (IC) and workforce development were discusses and captured in this report.

● Activities

The activities were divided into four segments:

1. A Series of Keynote speeches hosted online from luminary experts on various aspects of sustainability in computing.
2. A workshop day broken up into breakout rooms based on the hierarchical levels of computing where attending experts led by a domain-lead discussed challenges in each of those areas immersively.
3. A phase of scribing the discussions of the workshop day by scribes of the workshop day hierarchical domains.
4. A working roundtable in-person meeting where the individual components of the reports were coalesced together into the final form.

● Recommendations for NSF

The recommendations to NSF center around sponsoring activities towards increasing awareness for sustainability in computing, encouraging sustainability focused research and research that impacts sustainability of computing systems and emphasizing the principles of reduce, reuse or recycle. Development of a cadre of sustainability-aware engineers, researchers and practitioners through workforce development activities.

Chapter 2: Introduction

Computing is the enabler of modernization and has a profound impact on human life, society, and the environment. The scope of computing systems spans from minuscule sensor/processor devices to massive data centers and supercomputers that find applications in households, battlefields, mines, and even space. It is estimated that information and computing technologies (ICT) would consume a significant amount of energy, totaling 20% of the total global energy consumption by 2030 [1]. The energy consumption of computing equipment not only reaches staggering levels during its operational phase but also contributes to significant embodied costs throughout its entire life cycle. These costs encompass carbon emissions, the production of toxic chemicals and waste, carcinogens, pollutants, and water consumption. The entire value and production chains associated with computing also have substantial and direct implications for human rights and the environment. Therefore, it is crucial for the community of computer designers, builders, and users to be cognizant of the sustainability concerns related to computing and to prioritize sustainability as a fundamental goal in their practice rather than regarding it as an afterthought [2, 3].

Sustainable computing refers to the practice of designing, manufacturing, utilizing, and disposing of computing technologies and computer systems in an environmentally friendly and resource-preserving manner. It involves minimizing the adverse effects of computing on the environment and encouraging best sustainability practices throughout the full lifecycle of computing systems.

More precisely, sustainable computing spans three important principles: (i) Energy efficiency, which refers to the reduction of energy consumption by computing systems, including optimizing hardware and software design to minimize electrical energy usage during operation and while in standby mode. (ii) Resource conservation, which includes the minimization of non-renewable resources, such as rare metals and minerals, in the manufacturing and disposal of computing equipment and computer systems. (iii) Design for sustainability, which encourages the integration of best sustainability practices in the design and/or development of computing systems, including the use of renewables and clean energy, life cycle assessment, and e-waste management. In addition, education and outreach to increase awareness among users and developers about the importance of sustainable computing is critically important.

To be precise, key metrics for sustainable computing technologies are essential for evaluating the environmental, social, and economic impact of these technologies. Some of the commonly used metrics to assess the sustainability of computing technologies are Energy Efficiency, Power Usage Effectiveness, Material Efficiency, Product Longevity and Upgradability, Carbon Footprint, E-waste Generation, Social Impact, Economic Viability, and Water Usage. These metrics collectively provide a thorough evaluation of the sustainability features of computing technologies, helping government and industry stakeholders make informed decisions that align with their environmental and social responsibility goals.

The full lifecycle cost of computing solutions encompasses several factors that directly relate to sustainable infrastructure, products, and services, i.e., Capital Expenditure (CAPEX), Operational

Expenditure (OPEX), and Embodied Cost [1]. OPEX comprises the electrical energy consumption, cooling costs, and water usage of the operating computing infrastructure and hardware. Sustainable computing products with energy-efficient components contribute to lower OPEX by minimizing electricity bills. CAPEX is influenced by the initial investment in the computing infrastructure and hardware. The sustainable computing infrastructure may have higher upfront CAPEX due to the incorporation of energy-efficient technologies and renewable energy sources. However, over the long term, these investments result in cost savings through reduced energy consumption and lower maintenance costs. Embodied cost refers to the environmental impact and resource usage associated with the design, manufacturing, transportation, and disposal of computing products. Sustainable computing products are designed to minimize embodied costs by using eco-friendly materials, reducing waste generation, and enabling longer product lifespans. Sustainable practices in manufacturing and supply chain management help lower the embodied costs of computing products.

The differences between end-user devices (e.g., laptops, smartphones, and other personal electronics) and cloud infrastructure (e.g., server clusters and data centers) are significant, particularly in terms of their CAPEX-OPEX split, which is 80%-20% split for end-user devices and 20%-80% split for cloud infrastructure [1, 4]. The difference in CAPEX-OPEX split between end-user devices and cloud infrastructure is primarily due to the nature of their usage and lifecycle. End-user devices are standalone devices that require a one-time purchase, and their usage is relatively infrequent, resulting in a higher CAPEX share. However, the cloud infrastructure requires continuous operation, maintenance, and upgrades, leading to a higher share of OPEX over time. This difference in cost distribution is a crucial consideration for organizations when making investment decisions and financial planning related to end-user devices and cloud infrastructure.

The cloud infrastructure has higher embodied costs compared to end-user devices. This is because building data centers requires significant amounts of raw materials, including steel, concrete, and other construction materials. Data centers house numerous servers and networking equipment, which have substantial embodied costs due to the manufacturing and assembly processes, as well as the extraction and processing of raw materials for their components. The cooling systems and power infrastructure used in data centers contribute greatly to the embodied cost of data centers. In contrast, the embodied cost of end-user devices is generally lower due to their smaller size and fewer components compared to large server farms. While laptops and smartphones still have embodied costs associated with their manufacturing, use of materials, and disposal, they have a smaller environmental footprint compared to the data center infrastructure, although the collective carbon footprint of all end-user devices is still significant due to their widespread usage and sheer numbers globally. Note also that the carbon footprint of both categories can vary depending on factors such as the region's energy mix, the efficiency of devices, and the adoption of sustainable practices.

The purpose of this workshop was to unite visionaries, experts, and practitioners with the aim of reviewing the state-of-the-art in sustainable computing and identifying areas that are lacking or need more attention. The workshop was designed to cater to a diverse audience, including

policymakers and investors, computer scientists and engineers, educators, and skilled practitioners. The participants were interested in gaining insights into the full lifecycle cost of existing and emerging computing technology solutions, products, and services. The workshop attendees strived to forecast the evolution of sustainable computing by formulating and addressing the challenges that must be overcome while also exploring innovative solutions. During the workshop, participants engaged in interactive sessions and gained insights from leaders in sustainable computing technologies and each other through keynote speeches, discussions, and brainstorming sessions.

One significant aspect of this effort involved exploring and understanding the sustainability implications of evolving generative artificial intelligence (AI) solutions. For example, comparing the carbon footprint cost of training large language and vision models with the potential benefits of using these models for inference, forecasting, and decision-making can yield valuable insights [5, 6]. As an illustrative example, consider the environmental impact of GPT-4, emphasizing its staggering carbon footprint. The training process alone demanded 100 billion petaflops of compute power and consumed 33.3 million kWh of energy, resulting in 12,987 metric tons of CO_2 equivalent. This is akin to the combined output of 20,000 NVIDIA DGX A100 GPUs, each boasting an FP16 performance rating of 5 petaflops per second and a 6kW power rating, running continuously for 12 days. To provide context, the average US household produces 7.5 metric tons of CO_2 equivalents per year. In summary, the carbon emission cost of training GPT-4 equates to the total annual carbon footprint of 1,730 US households. While the environmental impact is substantial, it is important to acknowledge the considerable benefits of utilizing GPT-4-like models, including heightened productivity, efficiency, creativity, and job creation opportunities. Another example is that by leveraging such models, we can reduce the carbon footprint of other activities that would otherwise be resource-intensive and energy-hungry. In the same context of AI based computing, due to the carbon footprint of GPU based computing hardware novel accelerators which are either significantly OPEX efficient or leverage the principles of reduce, reuse and recycle in creating novel carbon-efficient hardware. Some examples are novel memory-centric architectures such as Processor-in-Memory (PIM), computing near memory or utilizing memory such that movement of data from memory to processing engines is either reduced or eliminated as data movement is several orders of magnitude more power-hungry than computing itself. However, careful consideration must be given to reuse of existing hardware infrastructure, fabrication and foundries leveraging existing semiconductor memories such as SRAM or DRAM rather than indiscriminately producing novel material-based memory subsystems to achieve comparable performance gains [7].

One critical aspect of this effort involved exploring and understanding the sustainability implications of emerging generative artificial intelligence (AI) solutions. For instance, conducting a comparison between the carbon footprint cost of training large language and vision models and the potential benefits of utilizing these models for knowledge discovery, dissemination, and intelligent decision-making can provide valuable insights toward creating a more sustainable future. By considering the environmental impact of AI training and usage, we can identify ways to optimize AI technologies for a positive societal impact while minimizing their environmental

footprint. This approach ultimately benefits both society and the environment, fostering a more sustainable and responsible integration of AI solutions into various domains.

The findings of the workshop hold significant value for various stakeholders, providing them with valuable information to make informed decisions. Government agencies, investors, and company research and development teams can leverage the workshop findings to prioritize and incentivize the development of specific technology solutions over others. By understanding the full lifecycle cost of various computing technologies, these stakeholders can make strategic choices that align with sustainability goals and economic considerations. This information can guide them in directing resources towards more eco-friendly and economically viable technology solutions, ensuring a positive impact on both society and the environment.

The main workshop findings underscore the importance of dedicating significant efforts to advancing sustainable practices across the whole society. To achieve this, a multifaceted approach is necessary, encompassing research and development that focuses on creating sustainable technologies and products, as well as initiatives promoting their widespread adoption. Moreover, educational efforts play a crucial role in informing and inspiring people to embrace the best sustainability practices. Government agencies, companies, and individuals all have important roles to play in this endeavor. Embracing new sustainability-driven approaches and mindsets is essential to prioritize sustainable development and incentivize the creation and advancement of eco-friendly products and services and their adoption. By collectively fostering a culture that promotes sustainability in every aspect of our lives, we can contribute to a more sustainable future for the planet.

Bibliography

1. U. Gupta, Y. G. Kim, S. Lee, J. Tse, H.-H. S. Lee, G.-Y. Wei, D. Brooks, and C.-J. Wu, "Chasing Carbon: The Elusive Environmental Footprint of Computing," *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, Seoul, Korea (South), 2021, pp. 854-867, doi: 10.1109/HPCA51647.2021.00076.

2. U.S. National Science Foundation, *NSF Program on Design for Environmental Sustainability in Computing (DESC)*, 2022. [Online]. Available: https://www.nsf.gov/pubs/2023/nsf23532/nsf23532.htm

3. Semiconductor Research Corporation (SRC), *Decadal plan and sustainability charter*, 2022. [Online]. Available: https://www.src.org/about/sustainability/

4. D. Kline Jr, N. Parshook, X. Ge, E. Brunvand, R. Melhem, P. K. Chrysanthis, and A. K. Jones, “GreenChip: A tool for evaluating holistic sustainability of modern computing systems,” *Sustainable Computing: Informatics and Systems*, vol. *22*, pp. 322-332, 2019.

5. C. J. Wu, R. Raghavendra, U. Gupta, B. Acun, N. Ardalani, K. Maeng, G. Chang, F. Aga, J. Huang, C. Bai, and M. Gschwind, “Sustainable ai: Environmental implications, challenges and opportunities,” *Proceedings of Machine Learning and Systems*, *4*, pp.795-813, 2022.

6. D. Patterson, J. Gonzalez, Q. Le, C. Liang, L.M. Munguia, D. Rothchild, D. So, M. Texier, and J. Dean, “Carbon emissions and large neural network training,” *arXiv preprint arXiv:2104.10350,* 2021.

7. P. R. Sutradhar, Sathwika Bhavikadi, Mark Connolly, Savankumar Prajapati, Mark Indovina, Sai Manoj P Dinakarrao, Amlan Ganguly, "Look-up-Table Based Processing-in-Memory Architecture With Programmable Precision-Scaling for Deep Learning Applications," in IEEE Transactions on Parallel and Distributed Systems, vol. 33, no. 2, pp. 263-275, 1 Feb. 2022, doi: 10.1109/TPDS.2021.3066909.

Chapter 3: Applications

3.1. Executive Summary

The advancements in computing systems and artificial intelligence (AI) techniques have ushered in a new era of computing applications. However, they impose considerable costs and challenges in terms of higher processing requirements, surge in carbon emissions, and reusability of applications. To address these challenges, a paradigm shift in software development is crucial, which could make sustainability an integral part of the application design process. This chapter summarizes such needs and opportunities. For example, the application could be designed to be delay-tolerant for relaxed processing or rescheduling in large-scale or multi-tenancy systems. Plus, they need to be developed such that less data communication is required for large-scale processing, e.g., in generative AI models. Moreover, application developers can cautiously introduce AI techniques in their applications while being mindful of the functionality-processing trade-offs. Further, we discuss how applications and their programming frameworks can be developed considering the portability of applications, such that applications can be ported seamlessly across devices of different scales and leverage recent advancements in emerging hardware/software techniques, e.g., near-data processing capabilities or data compression. Lastly, the needs and opportunities for enhancing the longevity of applications are described.

3.2. Background

The advent of current embedded systems with hardware-software co-design led to the evolution of smart systems supporting a plethora of computing applications [1-2]. These systems heavily focus on meeting real-time performance requirements and facilitating a plethora of functionalities. The performance enhancements are obtained at the expense of increased power consumption, advanced manufacturing, and other challenges, eventually leading to endangering the sustainability of the devices and increased carbon dioxide (CO_2) emissions.

Traditional techniques that address sustainability challenges are often dealt with from the hardware perspective, such as the development of power management techniques, architectural innovations, or advanced manufacturing techniques [3-5]. Though such a vision aligns well with the thermal, power, clock, and battery limitations, it spurs Jevon's paradox [6]. In other words, energy-efficient embedded system design has fueled the wide adoption of many devices in the past decade, leading to a larger carbon footprint [7]. Such a wide adoption led to the collection and processing of massive amounts of data either on the device or at the data centers, creating a negative impact on sustainability.

In terms of data processing and analysis, one misconception is that data centers, servers, or complex computing systems are the main cause of carbon footprint and endanger sustainability. Thereby relaxing the constraints and optimization for sustainability in the design of edge devices and systems. Contrarily, recent research [8-10] showcases that the rapid increase in the adoption of edge devices such as smartphones and communication infrastructure tends to outperform

carbon emissions compared to data centers. For instance, in the case of video streaming such as Netflix and other applications, the amount of carbon emission by the data center is comparable to or even less than the carbon emission by the edge devices such as mobile phones or Televisions, and the networking infrastructure [11-13].

Traditional hardware-based solutions have a limited impact on addressing this challenge due to limited capacity and minimal feasible savings. Design of effective data analysis techniques and software applications, on the contrary, have multi-fold impacts on sustainable computing and carbon footprint reduction [14-16]. The introduction of artificial intelligence (AI) or machine learning (ML) computing paradigms at hardware and software levels, though it enables smarter functionality, increases the number of required computations drastically. The computations performed by such applications have increased orders in magnitude with the introduction of applications such as ChatGPT [17]. As per the statistics [M.M.], in the past two years, the data required and/or processed by artificial intelligence (AI) or machine learning (ML) engines have doubled, whereas the AI models have increased by 20x in terms of their depth and architecture. The carbon footprint from training and inference has increased by nearly 2x [18]. In terms of carbon footprint, training of the current models leads to nearly 45.2 million tons of CO2, which is nearly the same as the carbon footprint of 5 homes [7].

Thus, the design of applications and data processing (especially by AI/ML applications) must be an integral part of understanding the needs of sustainability and the environment. However, one main challenge in such a design of applications is the gap that exists between the hardware and software designers, i.e., hardware architects focus on energy efficiency and the footprint of the system, and software engineers focus on usability and features provided by their applications. This void often makes the software power-hungry and non-sustainable despite the best efforts from both hardware and software designers. Furthermore, sustainability is an expensive option to be considered in application development. Considering the limited resources and experience of the software developers in small and medium-scale businesses, the design of applications for sustainability is expensive. Therefore, it is non-trivial to identify the solution for sustainability without sacrificing the application specifications. In the context of AI/ML or deep learning applications that traditionally rely on huge amounts of data an alternative would be to explore the reduction of data dependence on training. Techniques like “few-shot”, “one-shot” or “zero-shot” learning where only a few, one or no samples are required for at least one class in the classification job respectively. However, as of now, these methods do not have high accuracies as in traditional data reliant training methods. Novel research to reduce training data dependence without compromising accuracy is needed to reduce training phase footprint.

Green software [11] is proposed as a panacea and is in the early stages of development. One of the main causes is the lack of understanding between the software engineers and the sustainability metrics. Green software also needs improvement in terms of design and understanding. For example, a study among multiple CS 101 students has shown that different codes lead to different power consumption, though they all meet the functional requirements [11]. Bringing this green software needs to be emphasized right from the beginning of the programming rather than as an afterthought.

In addition, the processing of data by AI/ML techniques is another critical factor that significantly impacts sustainability and carbon footprints. As aforementioned, with the increase in depth and width of the AI/ML architectures, the number of computations and the required data to train such applications are increasing tremendously [19]. Traditional techniques such as precision scaling [20], pruning [21], and carry-skip computations [22] can aid in reducing carbon footprint; considering the scale of the AI/ML techniques, the impact is still smaller. This indicates the need for novel low-data-driven AI/ML techniques and AI/ML techniques that can perform multi-task learning. Sustainability in applications also leads to more opportunities in terms of development and management. Applications have an impact on the overall carbon footprint. The performance and the specifications of the application induce a lot of cost and complexity, leading to a higher carbon footprint. An application that has complex specifications often leaves a carbon footprint irrespective of its performance constraints or requirements.

Thus, the key challenges in the development of applications for sustainability can be outlined as follows:

- Integration of software and hardware from the sustainability perspective.
- Selecting the right programming methodology and framework to make an application sustainable.
- Design of efficient ML/AI techniques with minimal data.
- Improving the longevity of the applications and minimizing the unwanted resources for application execution.
- Application optimization for sustainability.

3.3. Pathway for the Design of Sustainable Applications: Opportunities and Challenges

We outline the path for the development of sustainable applications to enhance the positive environmental impact. We also present some of the feasible solutions for the development of sustainable applications and the challenges that still need to be addressed in the era of edge computing and AI/ML.

3.3.1. Sustainability-aware Application Development Process

Sustainability has to be an integrated part of application design and development rather than an afterthought process. Multiple reasons for such a design strategy include reducing the reengineering costs and avoiding the introduction of bugs in the later stages. The application design process needs to be sustainability-aware and reconfigurable, along with balancing performance and sustainability.

One of the recommended embeddings in application development is delay tolerance. For instance, shifting or rescheduling the operations depending on the application usage time can minimize the carbon footprint. For efficient and sustainable computing platform design, the techniques to introduce such optimizations need to be enabled both from the system design and

application perspectives. The present-day software applications do not support such a paradigm and are solely performance-centric.

Another major source of carbon footprint during application processing is data communication. As most of the current and emerging applications work on large amounts of data, they need to communicate frequently between memory and logic units. For example, in the case of ChatGPT, nearly few GBs of data are communicated between data and logic blocks. Such large amounts of data cause significant energy consumption, eventually leading to an increased carbon footprint. The current schedulers lead to limited energy efficiency due to the traditional von Neumann architectures and traditional memory access techniques. For instance, better schedulers and memory management units are pivotal for memory usage optimization, which eventually leads to lower power and carbon footprint. Thus, one needs to design applications that are aware of underlying hardware architecture and the impact it leaves on the hardware in terms of energy efficiency.

3.3.2. Design of AI/ML Applications for Sustainability

With the advent of AI/ML, most of the current systems and applications support AI/ML applications. Despite the benefits offered by these applications, they are becoming one of the major sources of resource consumption and carbon footprint due to their memory and compute-intensive traits.

Bringing awareness to the software developers is one of the primary steps to be taught to the program/application developers. One of the best examples is the case study that demonstrates the power consumption for a program in the CS 101 course by one of the panelists, as described earlier. Depending on the functions used and the libraries loaded, the overall footprint will change. So, the developers need to be mindful of the libraries loaded or used in their applications, and they should be optimized. For this purpose, application developers need to have a way to measure efficiency in terms of carbon footprint or power consumption. Such feedback will help in designing applications that are sustainable. The design of applications also needs to be tailored to the underlying system and its architecture. Such awareness also enables efficient application mapping and execution, resulting in sustainable application execution.

Furthermore, with the advent of non-von Neumann architectures, the hardware architectures have drastically changed. For instance, in-memory computing architectures have emerged in recent times, which alleviates the need for frequent data transfer between compute and memory units. The applications need to be redesigned for better energy efficiency and lower carbon footprint for such advanced and emerging architectures. In a similar manner, the memory schedulers and the application mapping techniques need to be revamped to match the architectural modifications. Several alternatives exist for designing Processor-in-Memory (PIM) or Processor-near-Memory (PnM) where the memory may be SRAM, DRAM, or emerging technologies such as memristors or other non-volatile memories (NVMs) [26]. Each technology has its pros and cons from performance and reliability perspectives, whereas DRAM or SRAM has some distinct advantages from a sustainability perspective. SRAM and DRAM technology will enhance sustainability in the

accelerator design space as this will enable the reuse of existing SRAM or DRAM foundries or manufacturing steps as opposed to the embodied costs of requiring new fabrication infrastructure necessary for emerging memory technologies with exotic materials. Among SRAM and DRAM, SRAM would be faster in-memory access latencies while DRAM can scale up more elegantly to support PIMs handling large workloads in the future world of Hyperscale AI or Big Data [27, 23, 24]. Moreover, DRAM-based systems are likely to be manufactured in older technologies that are generally cleaner [25, 28] and have less embodied Carbon related to the manufacturing process.

3.3.3. Improving the Longevity of the Applications

Another critical aspect in developing applications for sustainability is performing the market study on current and future requirements rather than merely relying on current needs. This indicates the trend of requirements and expectations of the users, paving the pathway for applications to remain in the market for the current and future.

In terms of application development strategy, the design or adaptation of certain application elements will offer better sustainability compared to others. Such design trends need to be considered for future needs. The optimization choices also need to be carefully made. For instance, a specific design trend can achieve minor power saving, say 1%. However, the larger question is: Is that minor saving worth it when the longevity of the application cannot be well served? In other words, application optimization might lead to smaller power savings for current architecture and operating systems. However, an upgrade of architecture or the user might lead to larger overheads and/or redesign costs, which will indirectly impact the sustainability of the application and the carbon footprint. Thus, it is pivotal to consider the current and future needs while optimizing the applications for sustainability.

3.3.4. Co-Design of Sustainable Applications and IDEs:

Software applications are developed and executed in Integrated Development Environments (IDEs) for efficient monitoring and development. However, the IDEs are traditionally bulkier compared to the applications they run due to their requirement to support development, execution, and debugging. However, such feature-rich IDEs lead to large overheads and memory accesses. If one observes the IDEs carefully, they consume a large amount of energy by loading a diverse set of libraries, though not necessary or required by the application. Most IDEs are highly bloated and consume large amounts of power due to loading libraries and other supporting scripts. Optimizing IDEs can significantly reduce energy consumption and eventually lead to higher sustainability and a lower carbon footprint.

Application Development and Optimization based on the Use Case: In addition to designing IDEs and applications based on the hardware, the design of applications also needs to consider the usage scenarios, such as enterprise vs. consumer domain. Depending on the enterprise or consumer domain applications, the constraints and the opportunities to optimize the application for sustainability vary.

In the case of the enterprise domain, certain standards have to be followed and can be controlled, which makes it easy to target for sustainability. Furthermore, as enterprise applications have wider applicability and users, sustainability has a larger impact. However, the development of applications based on sustainability needs significant domain knowledge, expertise, and skillset, and expensive (in terms of price), as discussed earlier. As such, small businesses might not be able to afford such application developers and do not have the resources to validate and optimize for sustainability goals. On the other hand, as enterprise applications are expected to last for a few years or decades, the applications might not be developed considering the future hardware optimizations and user demands. As such, frequent updates and optimizations need to be performed, which further contributes to large energy consumption and carbon footprint. This indicates the need for the design of applications based on the current and future demands and architectural changes.

On the contrary, consumer applications need not be verified, nor do there exist stringent rules on standardization for application development. This makes it hard to achieve sustainability. Furthermore, the return on investment for sustainable application development is minimal.

3.3.5. Metrics to Measure Sustainability

In addition to the development and optimization of the applications for sustainability, there needs to be a standardized manner to measure and benchmark sustainability. Energy consumption, though it can be considered as a metric, does not reflect the application's sustainability, as it only reflects the computations rather than the application's sustainability or carbon consumption. Thus, a set of standard metrics to measure sustainability rather than mere energy efficiency is required. Also, a standard set of applications and benchmark suites are required to validate and benchmark the applications for sustainability.

3.4. Summary and Recommendations for NSF

We summarize the key points from the panel discussion and the presentations in a nutshell for quick and easy grasping as follows:

- Application development must be an integral part of understanding the environment
- Green software design methodology must be incubated in application development for better sustainability
- Current software applications do not support delay tolerance. Software with delay tolerance capability to trade-off performance for sustainable resource consumption must be encouraged.
- Improved resource utilization needs to be an integral part of application development with a careful understanding of relevant trade-offs.
- Sustainable application is the need of the hour, but sustainability in applications is non-trivial. Therefore, early-stage research should be encouraged and funded.

- Challenges: improving the longevity of applications and better understanding customers' choices and architectural paradigms. Identifying solutions for sustainability without sacrificing the application specifications and performance is required.
- Sustainability is not a priority in most small/medium businesses. Therefore, the sustainability of application software should be encouraged and incentivized. Methods and strategies to achieve this end need to be developed or researched.
- Changing the application design strategy leads to better sustainability. Software application developers need to be educated on green software right from the initial stages rather than sustainability as an afterthought. Outreach and educational efforts to non-traditional students and practitioners on sustainable application development need to be sponsored.

3.5. Bibliography

1. J. Park, "Chiplets and heterogeneous packaging are changing system design and analysis," in Cadence Whitepaper, 2020.

2. A. Mastroianni, B. Kerr, J. Nasrullah, K. Cameron, H. J.Wong, D. Ratchkov, and J. Reynick, "Proposed standardization of heterogenous integrated chiplet models," in IEEE International 3D Systems Integration Conference (3DIC), 2021.

3. T. Benz, L. Bertaccini, F. Zaruba, F. Schuiki, F. K. GÅNurkaynak, and L. Benini, "A 10-core SoC with 20 fine-grain power domains for energy-proportional data-parallel processing over a wide voltage and temperature range," in IEEE European Solid State Circuits Conference (ESSCIRC), 2021.

4. Y. Wang, W. Zhang, M. Hao, and Z. Wang, "Online power management for multi-cores: A reinforcement learning based approach," IEEE Transactions on Parallel and Distributed Systems, vol. 33, no. 4, pp. 751–764, 2022.

5. W. Kim, M. S. Gupta, G.-Y. Wei, and D. Brooks, "System level analysis of fast, per-core DVFS using on-chip switching regulators," in IEEE International Symposium on High Performance Computer Architecture, 2008.

6. C.-J. Wu, R. Raghavendra, U. Gupta, B. Acun, N. Ardalani, K. Maeng, G. Chang, F. A. Behram, J. Huang, C. Bai, M. Gschwind, A. Gupta, M. Ott, A. Melnikov, S. Candido, D. Brooks, G. Chauhan, B. Lee, H.-H. S. Lee, B. Akyildiz, M. Balandat, J. Spisak, R. Jain, M. Rabbat, and K. Hazelwood, "Sustainable AI: Environmental implications, challenges and opportunities," ArXiv, 2021.

7. S. M. P. Dinakarrao, H. Homayoun, D. Schien, C.-J.Wu, and Z. Zong, "Impact of applications on sustainable computing - panel summary," in NSF Sustainable Computing Workshop (Tech Report), 2022.

8. M. Zaki, B. Theodoulidis, and D. Diaz, “Carbon footprint innovation through environmental information management,” in Annual SRII Global Conference, 2011.

9. D. Schien, P. Shabajee, J. Chandaria, D. Williams, and C. Preist, “Using behavioural data to assess the environmental impact of electricity consumption of alternate television service distribution platforms,” Environmental Impact Assessment Review, vol. 91, 2021.

10. D. Schien, P. Shabajee, H. Akyol, L. Benson, and A. Katsenou, “Help, I shrunk my savings! assessing the carbon reduction potential for video streaming from short-term coding changes,” in IEEE International Conference on Image Processing, 2023.

11. J. a. Saraiva, Z. Zong, and R. Pereira, “Bringing green software to computer science curriculum: Perspectives from researchers and educators,” in ACM Conference on Innovation and Technology in Computer Science Education, 2021.

12. A. K. Jones, Y. Chen, W. O. Collinge, H. Xu, L. A. Schaefer, A. E. Landis, and M. M. Bilec, “Considering fabrication in sustainable computing,” in IEEE/ACM International Conference on Computer-Aided Design (ICCAD), 2013.

13. A. K. Jones, L. Liao, W. O. Collinge, H. Xu, L. A. Schaefer, A. E. Landis, and M. M. Bilec, “Green computing: A life cycle perspective,” in International Green Computing Conference Proceedings, 2013.

14. E. Brunvand, D. Kline, and A. K. Jones, “Dark silicon considered harmful: A case for truly green computing,” in International Green and Sustainable Computing Conference (IGSC), 2018.

15. U. Gupta, Y. Kim, S. Lee, J. Tse, H. S. Lee, G. Wei, D. Brooks, and C. Wu, “Chasing carbon: The elusive environmental footprint of computing,” in IEEE International Symposium on High-Performance Computer Architecture (HPCA), 2021.

16. V. Justafort, R. Beaubrun, and S. Pierre, “On the carbon footprint optimization in an intercloud environment,” IEEE Transactions on Cloud Computing, vol. 6, no. 03, pp. 829–842, Jul 2018.

17. Frieder, S., Pinchetti, L., Griffiths, R.R., Salvatori, T., Lukasiewicz, T., Petersen, P.C., Chevalier, A. and Berner, J., 2023. Mathematical capabilities of chatgpt. arXiv preprint arXiv:2301.13867.

18. D. Patterson, J. Gonzalez, U. Holzle, Q. Le, C. Liang, L. Munguia, D. Rothchild, D. R. So, M. Texier, and J. Dean, “The carbon footprint of machine learning training will plateau, then shrink,” Computer, vol. 55, no. 07, pp. 18–28, Jul 2022.

19. M. M. Rathore, S. A. Shah, D. Shukla, E. Bentafat and S. Bakiras, "The Role of AI, Machine Learning, and Big Data in Digital Twinning: A Systematic Literature Review, Challenges, and Opportunities," in IEEE Access, vol. 9, pp. 32030-32052, 2021.

20. Xu, X., Ding, Y., Hu, S.X. et al. Scaling for edge inference of deep neural networks. Nat Electron 1, 216–222 (2018).

21. Blalock, D., Gonzalez Ortiz, J.J., Frankle, J. and Guttag, J., 2020. What is the state of neural network pruning? Proceedings of machine learning and systems, 2, pp.129-146.

22. Mao, Xiaojiao, Chunhua Shen, and Yu-Bin Yang. "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections." Advances in neural information processing systems 29 (2016).

23. P. R. Sutradhar, S. Bavikadi, S. M. P. Dinakarrao, M. A. Indovina and A. Ganguly, "3DL-PIM: A Look-up Table oriented Programmable Processing in Memory Architecture based on the 3-D Stacked Memory for Data-Intensive Applications," in IEEE Transactions on Emerging Topics in Computing, doi: 10.1109/TETC.2023.3293140.

24. P. Gu et al., "DLUX: A LUT-Based Near-Bank Accelerator for Data Center Deep Learning Training Workloads," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 40, no. 8, pp. 1586-1599, Aug. 2021, doi: 10.1109/TCAD.2020.3021336.

25. Y. -C. Kwon et al., "25.4 A 20nm 6GB Function-In-Memory DRAM, Based on HBM2 with a 1.2TFLOPS Programmable Computing Unit Using Bank-Level Parallelism, for Machine Learning Applications," 2021 IEEE International Solid- State Circuits Conference (ISSCC), San Francisco, CA, USA, 2021, pp. 350-352, doi: 10.1109/ISSCC42613.2021.9365862.

26. Sathwika Bavikadi, Purab Ranjan Sutradhar, Khaled N. Khasawneh, Amlan Ganguly, and Sai Manoj Pudukotai Dinakarrao. 2020. A Review of In-Memory Computing Architectures for Machine Learning Applications. In Proceedings of the 2020 on Great Lakes Symposium on VLSI (GLSVLSI '20). Association for Computing Machinery, New York, NY, USA, 89–94. https://doi.org/10.1145/3386263.3407649

27. Q. Deng, Y. Zhang, M. Zhang and J. Yang, "LAcc: Exploiting Lookup Table-based Fast and Accurate Vector Multiplication in DRAM-based CNN Accelerator," 2019 56th ACM/IEEE Design Automation Conference (DAC), Las Vegas, NV, USA, 2019, pp. 1-6.

28. UPMEM, https://www.upmem.com/ (accessed Oct. 11, 2023).

Chapter 4: Systems

4.1. Executive Summary

This chapter discusses sustainability challenges in ICT systems, primarily focusing on the operational and life cycle efficiency metrics. First, we introduce key personas involved in the system-level sustainability of computing: system integrators, data center designers, operators, and workload owners. To foster a sustainability mindset, we advocate incorporating life cycle assessment into system design, considering workload characteristics and component longevity. Transparency in data, especially related to carbon costs and workload behavior, is crucial. Therefore, standardized metrics and policies are recommended to aid decision-making, addressing challenges like system upgrades, repairability, and component-specific design. We also discuss the potential of disaggregated and composable systems, emphasizing modularity, repairability, standardization, and the dynamicity of workloads to contribute to power grid decarbonization. Finally, the implementation of incentives, regulations, and programming models are suggested, which can enable the efficient movement of workloads and promote sustainable practices in data centers.

4.2. Background

An IT system is any organized assembly of resources and procedures united and regulated by interaction or interdependence to accomplish a set of specific functions.

Our goal in this section is to illuminate the issues pertaining to sustainability in the integration of components to systems, as well as the design and operation of data centers comprising systems.

The two main metrics that we are concerned with are:

- Operational Efficiency: The energy and associated carbon per unit of work performed at the runtime (operations) phase of the system.
- Life Cycle Efficiency: The energy/carbon associated with the entire life cycle of the system, including material extraction, manufacturing (of all parts comprising it), assembly, transportation, and end of life.

While it makes sense to consider these two metrics (#1) and (#2) in isolation, sometimes there are tradeoffs between them that require consideration. For example, upgrading a server is associated with a step function in carbon life cycle cost inherited (as scope 3) by the owner. However, potential operational efficiency gains relative to the old system may very well compensate for the step increase in lifetime cost. Whether there will be sufficient operational efficiency gains depends on the expected workloads, their characteristics (e.g., ratio of read/write), and their volume. All these tradeoffs must be well understood, data must be available, methodology defined, and incentives in place for the key stakeholders to make such decisions.

4.3. Key Personas for Systems Sustainability

One of the key goals is to foster a sustainability mindset with the stakeholders in each of the areas, termed ‘personas.

There are four personas that are key in this discussion.

1. **System integrators:** work to design new systems by integrating IT components such as accelerators, memory, and storage devices, and networks. The key decisions that they make are the selection of the components, e.g., NVIDIA A100 or V100, and the system configurations. Key considerations include operational efficiency (speed, power) and CAPAX.
2. **Data Center Designers:** A data center comprises systems and can be thought of as a system, especially as we move towards disaggregated and composable computing (See later). A data center designer is responsible for designing a new data center or expanding an existing one. The key decisions they make are the location of the data center (this is critical since it will have a huge effect on cooling and renewable energy), cooling technology to be used, and systems. Multiple concerns include the SLAs of their clients, the facility owner’s ability to support requirements, e.g., for cooling (if not the same), and the costs associated with various technologies.
3. **Data Center Operator:** Once a data center is already put in place, it needs to be operated. A data center operator is tasked with policies governing application placement on systems and system upgrades or repairs. Concerns include the SLAs of applications, system efficiency and optimization, and costs.
4. **DevOps or workload owners:** This persona makes decisions about where to run a workload. In a hybrid cloud context, some workloads can be placed dynamically, and there is a choice. Concerns include privacy, compliance, cost, and sustainability.

4.3.1. Fostering a Sustainability Mindset

To foster a sustainability mindset, we need to have sustainability as a first-class goal for all three personas and to develop technologies that will help them perform their role.

For example, System designers who care about Sustainability will want to factor in the life cycle expectancy of the components they use (e.g., the lifetime expectancy of SSDs), as well as the Life Cycle Carbon Cost, aka LCA (Life Cycle Assessment). The Life Cycle Assessment of a product is concerned with calculating the carbon cost of manufacturing, transportation, and end-of-life. For example, some solid-state drive technology is notorious for high manufacturing costs. To make informed decisions, they may also need to know something about the characteristics of the workloads. For example, a high number of writers will adversely affect the lifetime expectancy of some SSD drives [1].

As another example, a center administrator will want to identify components that are malfunctioning and thus inefficient. Once such a component is identified, they will need to decide

if to repair or replace it. Hot spot or outlier detection mechanisms to identify such malfunctions must be developed. Additionally, they will need to decide when to upgrade the computer system. There may be some tradeoffs between the embodied carbon of a new system purchased vs. the potential operational efficiency gains obtained by the new system. To make informed decisions, they will need data, methodology, and policy drive incentives.

4.3.2. Key Requirements and Recommendations for Sustainable Systems

(1) Data and Transparency

- **(a)** Part manufacturers need to publish the Life Cycle Assessment of each part according to the standard [2]
- **(b)** Part manufacturers need to publish benchmark data. At the point of writing this document, SPEC benchmarks mostly consist of the number of FLOPs per second. There is limited data on power behavior. This should be included as part of the benchmarks (power behavior is not a linear curve, and it also depends on the workload characteristics.
- (c) Part manufacturers and/or third parties need to publish information about the lifetime expectancy of components and how workload characteristics affect the lifetime expectancy (for example, read vs. write).
- (d) Data Center owners, and in particular cloud providers, must be transparent about the hardware they use, their cooling overhead and the renewable energy they use, and the method of calculating the carbon associated with each user workload. This is necessary to allow users to make choices on where to run the workloads. At the point of writing, some cloud providers expose such data at granularity too coarse and with no way of verifying/auditing.

(2) Metrics

Standards are needed to help the key stakeholders with decisions, especially ones that involve tradeoffs.

- (a) How to calculate the actual amortized cost of a job executing on a system in a data center, factoring in the operational cost, including overheads such as cooling and power losses, and the amortized lifetime cost (embodied ‘tax’). There have been some attempts towards this goal (e.g., [3]) that are not completely consistent and need to be brought together (e.g., [4)] and supported by a standard body. This is needed because it will help the system integrator select components for a new system based on the expected characteristics of the workload, expected volume, and SLAs.
- (b) How to associate carbon cost with workloads in a multi-tenant cloud environment? This item is very much related to (a). However, there is a need to factor in the division of roles between a cloud provider and a user, the use of multi-tenant platform services, and their provenance chain. The standard needs to be easy to understand and verifiable. At the point of writing this paper, every cloud provider uses their own method. Some do not provide enough details on the method, so any comparison and reasoning are like ‘apples and oranges.’ An initial proposal to drive towards a standard across cloud providers can be found here [5].

(3) Policy and Regulations

Policy and regulations are needed to provide the necessary incentives for the various stakeholders to make decisions with a sustainability mindset.

(a) The policy should work to incentivize part manufacturers to publish the data in 1(a), 1(b), and 1(c).

(b) Regulations must mandate reporting of relevant and meaningful data center metrics that absolutely must include embodied carbon (scope 3) of all systems used, as well as aspects such as recycling, repair, and re-use.

4.4. Research Opportunities

4.4.1. Opportunities in Designing Disaggregated and Composable Systems

The disaggregated and composable system design provides flexibility to perform a variety of workloads. The system offers a dynamic co-design platform that allows experiments and measurements in a controlled environment, which not only speeds up the system design but also gives an opportunity for software evolution (e.g., [6]). It also decouples the lifecycles of different components used in the system from each other. Disaggregation also enables power, cooling, and networking resources to be shared by multiple subsystems, including network, CPU, memory, and storage resources. The design consideration includes adopting available technology with the understanding of application characteristics. This results in operational efficiency improvements because the pooled resources can support multiple computing subsystems simultaneously, rather than each system having its own. Users will have the opportunity to control how and which resources to use.

One of the challenging questions for the IT administrator is whether to upgrade or repair when a system is not performing at its optimum level. Every product has its own life cycle. During the lifecycle, there will always be scenarios where the product is not able to provide the performance demanded by the system. Especially in this current age of development of ASIC manufacturing following Moore’s law, the computational power of the integrated circuits is doubling every eighteen months, which virtually makes the previous generation obsolete for some applications. But most of the remaining parts of the system will remain virtually the same. So, it is an ideal era to start focusing on disaggregated and composable systems.

4.4.2. Modular System

Especially in the computing industry, the trend is to design the whole system on a chip (SOC), which, on the one hand, makes the system more compact and operationally efficient, but on the other hand, if anything breaks, it is virtually impossible to repair. There is very limited or no provision for upgrading the system if desired. The only option for an upgrade is a replacement, which makes the product lifecycle very small and has a huge adverse impact on the overall carbon footprint. But not all sections of the system evolve at the same pace. Hence, the system designer should consider this during the design phase and design the system in a more modular way so

that they can only upgrade a few components and reuse or repurpose the other parts of the system. This will elongate the average lifecycle of the product.

4.4.3. Availability of the Repairability Data

The presence of different advanced technology parts in modern equipment has enabled the manufacturers to reduce access to overall system design to the end users, proclaiming protecting their "Proprietary" rights limits the repairability of a system. There has been a movement of “Right to Repairability” in the USA to make legislation which will force the original equipment manufacturers (OEMs) to provide consumers and independent repair businesses equal access to repair documentation, diagnostics, tools, service parts as their direct or authorized repair providers. This has the potential to enhance the life cycle of a product drastically and reduce the overall carbon footprint of the product.

4.4.4. Too Specific Design

Most of the industry in the computing sector is moving towards designing and manufacturing their own ASIC rather than going for a generic off-the-shelf CPU or GPU. It is true that these ASICs can perform tasks much faster and more efficiently compared to the generic CPUs or GPUs. Still, if they are designed too specifically to perform only a few tasks, they practically cannot be repurposed for any other jobs. Hence, there is a potential for a huge trade-off between embodied carbon footprint (ECFP) and operational carbon footprint (OCFP). ASICs designed too specifically can improve the operational carbon footprint in the short run, but considering the overall lifecycle, they have drastic adverse effects on embodied carbon footprint. During the designing phase, the system designer must consider the trade-off between the ECFP and OCFP.

4.4.5. More Standardization

Product standardization is a manufacturing and marketing process that ensures uniformity and consistency among the several specific products available in various regions across the industry. It entails ensuring that a product meets specified criteria for item quality, design, service delivery, or appearance in each area. The purpose of standardization is to ensure that specific procedures or processes within a given context are uniform and exact. If a user wants to move away from one manufacturer to another manufacturer, if there is no standardization, the previously existing system cannot be repurposed or integrated with the new system. Virtually, the lifecycle of all the components used in that system becomes drastically short. Whereas if there were standardization of different components, many of the components used in the system could have been repurposed in the new system, which would have improved the embodied carbon footprint.

4.4.6. How to Design Datacenters that Expand and Contract Contributing to the Power Grid Decarbonization

As stated before, a data center comprises systems integrated into an infrastructure that includes a power conversion and transformation system and cooling technology. Workloads run on

systems in the said data center. In many cases, businesses own more than one data center, and their workloads run on a mix of on-prem data centers and third-party clouds.

One of the main characteristics of IT workloads is that, in many cases, they can be moved dynamically to run in different environments. This may necessitate a unified abstraction layer, such as Kubernetes, which provides for containerized workloads. In the case of Kubernetes, it is provided as a service by all exa-scalers. Thus, any containerized workload can be easily moved.

A decision to move a workload must respect multiple constraints, such as compliance, security, and privacy. In addition, one must carefully examine performance (such as latency).

The dynamicity of workload in the modern cloud environment offers opportunities for data centers to play an important role in the de-carbonization of the power grid.

Specifically, it is well known that the main issue with renewable energy is its unpredictability. Namely, it comes and goes based on climate conditions. The lack of effective energy storage implies that to reach the goal of full decarbonization, we must have means to match supply and demand. The best opportunity to do that is with IT workloads since they can be dynamically moved based on the availability of workloads (respecting other constraints).

Consider a sustainable city that attempts to rely 100% on renewable energy. In a case of shortage, you absolutely do not want to shut off essential services such as a hospital, but you can get some energy back to the grid from data centers that can contract by moving workloads somewhere else.

From a technological standpoint, there are no big inhibitors to achieve this vision. We go back to the issue of incentives and data. Data centers must publish the relevant metric as a time series in fine granularity.

Workload owners must be incentivized to enable their workloads to move, and this can be accomplished by standards and regulations. Programming models may be required to break applications into components based on their SLAs.

In a sustainable data center, there may be other means that can be applied. For example, by configuring systems with dynamic voltage/frequency scaling, some workloads can be made to tolerate a hit on latency in order to conserve energy at times when renewable energy is scarce.

4.5. Bibliography

[1] Tannu, Swamit, and Prashant J. Nair. "The dirty secret of SSDs: Embodied carbon." *arXiv preprint arXiv:2207.10793* (2022).

[2] “ISO 14040:2006,” ISO, https://www.iso.org/standard/37456.html (accessed Oct. 16, 2023).

[3] Gandhi, Anshul, et al. "Metrics for sustainability in data centers." *Proceedings of the 1st Workshop on Sustainable Computer Systems Design and Implementation (HotCarbon'22)*. 2022.

[4] Tamar Eilam, Pedro Bello-Maldonado, Bishwaranjan Bhattacharjee, Carlos Costa, Eun Kyung Lee, and Asser Tantawi. 2023. Towards a Methodology and Framework for AI Sustainability Metrics. In Proceedings of the 2nd Workshop on Sustainable Computer Systems (HotCarbon '23). Association for Computing Machinery, New York, NY, USA, Article 13, 1–7. https://doi.org/10.1145/3604930.3605715

[5] [1] Green-Software-Foundation, “Green-software-foundation/real-time-cloud,” GitHub, https://github.com/Green-Software-Foundation/real-time-cloud (accessed Oct. 16, 2023).

[6] I-Hsin Chung, Bulent Abali, and Paul Crumley. 2018. Towards a Composable Computer System. In Proceedings of the International Conference on High Performance Computing in Asia-Pacific Region (HPC Asia 2018). Association for Computing Machinery, New York, NY, USA, 137–147. https://doi.org/10.1145/3149457.3149466

[7] The decadal plan for semiconductors: a pivotal roadmap outlining research priorities. SRC. (n.d.). https://www.src.org/about/decadal-plan/

Chapter 5: Sustainable Computer Architectures

5.1. Executive Summary

Computer architectures and their design methodologies have played an important role in making the processing of computing applications faster and highly energy-efficient, which has made computing more accessible and integral to our daily tasks. Their role becomes more important in sustaining such efficiency, especially with the diminishing returns from transistor technology (post-Moore's law scaling) and the emergence of new large-scale workloads, such as for artificial intelligence (AI) and scientific computing tasks. However, these advancements primarily reduce carbon emissions due to architecture designs in their operation phase only, whereas most emissions (e.g., more than 80%-90%) could be accounted for by their non-operational phases throughout their life cycle, such as design and non-recurring engineering, materials, manufacturing, transportation, recycling, and disposal. This marks the need for a holistic approach to improving the sustainability of architecture and its design practices. It becomes even more crucial, given the rise in shortened usage or faster upgrades of consumer-scale devices and data centers with evolving needs of applications.

In this chapter, we outline multiple areas of challenges and opportunities for sustainable computing architectures, including full-life analysis quantification of their carbon emissions, developing a sustainability-aware mindset for architecting new processors or their design methodologies, incentivizing sustainability-aware practices, and the unique role of the architectures and architects in advancing sustainability across the computing stack. We discuss making the sustainability metrics first-hand in architecture design practices without compromising quality-of-service and making their reporting more accessible for sustainability-aware choices by design practitioners and end-users. Especially for designing sustainable architectures, we discuss the trade-offs of using architectural specialization, fault-tolerant architectures, and open-source technologies. We also discuss opportunities that can be enabled by resource-disaggregation and modular architecture designs, using artificial intelligence to improve the sustainability of architectures and their design process and reducing the non-recurring engineering efforts for novel architectures. We end with thoughts and recommendations for educating the next generation of computer architecture researchers and practitioners and the need for long-term, forward-looking research support and practices.

5.2. Background

5.2.1. Energy-efficient Architectures

Over the past decades, computer architecture researchers and designers have devoted substantial effort to designing energy-efficient personal computers, mobile architectures, and data center systems. These advancements include accelerating system performance, power

management, optimizing energy efficiency, and ensuring reliable computing. Several architectural innovations that have driven high-performance and energy-efficient computing include processor pipelining [1], caching [2-7], data prefetching [8], memory systems [9], speculative execution [10, 11], branch prediction [12], instruction-level parallelism [13], SIMD and vector processing [1, 14], multi-core and heterogeneous processing [15-18], on-chip communication [19, 20], dynamic voltage and frequency scaling (DVFS) [21, 22], power gating and low-power designs [23, 24], fault-tolerant systems and reliable computing [25, 26], near-data processing [27, 28], and domain-specific architectures [29-31]. More recently, advancements in artificial intelligence (AI) have driven the development of more specialized architectures tailored to the unique computational requirements of machine learning and deep learning tasks, including tensor cores [32], wafer-scale processing, exploiting sparsity and mixed data precisions [31, 33], transformer engines [32], and neuromorphic processing [34].

5.2.2. Need for Sustainability of Architecture Design, Use, and Reuse

Processor architectures have been traditionally designed and utilized with a primary focus on performance or energy efficiency. It has considerably reduced the operational carbon footprint of processing applications on cloud and/or mobile devices [35]. However, this approach neglects the sustainability aspect of their full life cycle, especially for the design and manufacturing phases of computer architectures that correspond to capital expenditure or capex [35-38]. Therefore, there remain major open challenges for designing sustainable computer architectures and improving the sustainability of architectural design practices.

With the growing demand for computing in society, there has been an increasing focus on studying and improving the sustainability of designing and operating computer architectures and systems [35, 36, 38-40]. For example, technology companies are pledging to be carbon neutral; they use neutrality practices such as carbon offsetting, carbon capture, and carbon sequestration. Major technology companies are actively working to reduce the carbon footprints of their operations by designing/employing more energy-efficient processors, infrastructures, and applications and using carbon-free energy resources for their operations [39, 41]. For computer architectures, recent studies and tools like [35-37, 42-44, 69] have considered quantifying the carbon footprint of not just the operation phase of processors but their full life cycle analysis (LCA). For instance, GreenChip [36] provided an evolution flow for determining carbon implications of choosing architectural components such as process core counts and memory/storage configurations. A recent first-order model in [43] uses proxies for estimating both embodied and operational carbon. For example, it uses chip size for calculating embodied carbon and power/energy estimates about fixed work-time scenarios for determining operational emissions. Most recently, ACT [44] enabled a more comprehensive quantification of carbon footprint LCA. It estimates emissions from hardware manufacturing (i.e., embodied carbon) based on information about environmental factors, hardware specifications, semiconductor fab, and workload characteristics, and it can provide a detailed breakdown of emissions due to various life cycle phases.

Future research on computer architectures and their design automation can help minimize capex-related carbon emissions. For example, one of the primary challenges is finding ways to

holistically quantify the carbon footprint of processors throughout their life cycle and the related resources consumed, from a processor's design development and evolution, manufacturing, testing, deployment, and operational usage, upgrades and reuse for low-tier processing, and its disposal at the end of life. The lack of reporting for related data makes it hard to achieve reasonable quantification. Further, these metrics need to be defined and quantified correctly (e.g., calculations of emissions could span across industries) and embedded as first-hand goals/constraints in the design process. Another challenge is to minimize the environmental impact of these life-cycle stages associated with computer hardware. For instance, higher-performance hardware tends to result in higher carbon emissions during manufacturing [35]. Additionally, prior approaches have started investigating the footprint associated with materials and manufacturing processes. In contrast, there can be significant emissions associated with the architecture design methodologies and non-recurring engineering efforts for every new architecture. Similarly, introducing specialization can help achieve operational efficiency, but it requires significant development/manufacturing efforts and the related carbon emissions could easily outweigh the gains achieved from operational efficiency. Therefore, exploring alternative manufacturing methods and reusable architectural design and design methodologies, especially for specialization, becomes crucial in reducing these emissions and promoting/adopting sustainability goals. Furthermore, as devices with billions of transistors experience low utilization, architectural design and optimization play a critical role in balancing the concept of "dark silicon" with manufacturing emissions [35]. Dark silicon refers to the portion of a chip that is inactive due to power and thermal constraints. Architectural optimizations can directly reduce CO2 output by judiciously provisioning resources, selectively introducing redundancy, and incorporating specialized yet reasonably programmable logic. Finally, resource disaggregation and designing architectures in a modular manner can offer opportunities for the reuse of architectural resources and related tooling in their design methodologies, respectively.

The round table discussion on improving the sustainability of computer architectures has considered such challenges and the potential next steps, including quantifying the sustainability of architecture design more effectively, incentivizing sustainability as a first-hand design metric, strategies for designing architectures with sustainability as a goal, and the unique role of architectural design in advancing sustainability across the computing stack. In particular, the round table spanned the following topical questions:

- How to incorporate sustainability as a first-hand design metric for processors?
- What are the current approaches for quantifying the sustainability of computing architectures? What are their limitations?
- How does architectural design uniquely impact the sustainability of computing?
- How to design computer architectures with sustainability as a goal?
- What are the implications of specialized computing architectures, error-resilient architectures, and open-source technologies on the sustainability of computing?
- How can architectural advancements help aspire sustainable options across the computing stack?
- How can the end-users of the architectures be made aware of the sustainability implications of their computing choice?

- How can architecture designers be incentivized to adopt sustainability as a design goal?
- What are the next steps to make sustainability a more tangible notion in architectural design?

5.3. Open Challenges and Opportunities

5.3.1. Accounting for the Full Life Cycle in Quantification

While the operational efficiency of computer architectures/systems is well studied, the sustainability of architectures is not fully quantified throughout their life cycle phases. For instance, a major factor in quantifying sustainability metrics is considering – What is the impact of life cycle phases on the hardware that we have designed? How are they being designed? How are they being manufactured? How are they being disposed of? What materials are we using? How sustainable are design methods, including those for customizable computing? Can we make processors and their design methods more reusable and accessible? Accounting for all such phases can help achieve a more realistic quantification and, thereby, carbon-aware designs and optimization.

Note that while recent studies for carbon LCA of architectures (e.g., [35, 36] and follow-up works) include embodied carbon corresponding to materials/manufacturing, the sustainability of the design methods and design practices themselves (recurring/non-recurring engineering efforts) has not been investigated [45]. This becomes crucial when exploring new domain-specific architectures, as such overheads would be non-trivial. This is because significant efforts get spent on finding a new architecture organization that works best for domain workloads, architectural characterizations, code optimization and generation for workloads, runtime management, etc. [46]. However, quantification of the design phase and accounting for their carbon life-cycle implications can inspire innovation in new design automation methodologies, thereby reducing the embodied carbon footprint related to designing architectures such as novel specialized processors or new memory systems.

5.3.2. Making Sustainability a First-hand Design Metric

Current approaches for designing architectures do not treat sustainability as a preliminary requirement. There remains a lack of quantifiable targets and metrics for such adoption among computing systems architects and application developers. For example, processor designers have primarily focused on metrics like throughput, latency, energy consumption, and resource requirements for their designs. To encourage the designers to make more sustainable choices, it is necessary to quantify the sustainability metrics for the architecture life cycle (embodied and operational emissions) and integrate such quantification throughout the design and usage process. When the notions of metrics and their milestone values are well-defined, they can inspire the community to actively pursue the goals, e.g., exa-FLOPS targets used in the HPC community or latency/EDP for application QoS requirements. Contrarily, for achieving sustainable computing, individuals and business entities might be setting their own qualitative notion of what is better

sustainability-wise. The lack of clear, holistic, or quantifiable metrics hinders their active involvement and sustainability improvements at their fullest potential.

5.3.3. Designing Architectures with Sustainability as a Goal

Current design methodologies and optimization frameworks for processors target metrics like performance or energy consumption. However, for designing high-performance, energy-efficient architectures that can meet the Quality of Service (QoS) requirements, the designers must consider the sustainability implications of their design choices. This is especially true for designing and manufacturing specialized architectures, as it introduces an excessive carbon footprint and increases embodied carbon. The embodied carbon footprint due to specialization needs to be balanced against the benefits gained by the usage of specialization over the new architecture's lifetime [35, 55]. Additionally, when resilience mechanisms are introduced, the design frameworks need to account for carbon footprint implications associated with the lifespan improvements and execution overheads of resilience mechanisms.

5.3.4. Aspiring for Sustainable Architectural Options Across the Computing Stack

Application developers have typically relied on using more accurate options, such as large AI models or large-scale simulations in scientific computing. However, this approach leads to excessive computing requirements, resulting in much higher operational footprints of systems and applications. With architectural innovations and the evolution of application algorithms, it is now possible to approximate these models with a reasonable degradation in task accuracy. For example, using lower precision of data or sparsely activating large AI models can significantly reduce the processing requirements while compromising accuracy by only 0.1% for tasks such as object detection or weather forecasting [31, 33, 47]. Although these approaches are gaining traction and relevant architectural innovations have been proposed, they are not yet commonplace. Therefore, business entities and developers, e.g., those in the AI and HPC domain, need to strive for more sustainable alternatives at the architecture and application levels.

5.4. Moving Forward

5.4.1. Full Life Cycle Quantification for Sustainability

The sustainability and carbon footprint of an architecture should be determined considering its full life cycle, including the material collection/transportation, manufacturing, design, and NRE efforts, and not just the carbon footprint related to the device operation. The LCA also needs to account for reuse of the architectural components [48-50] and disposal [51]. There needs to be new estimations, tools, and data for enabling such characterizations over full life cycle phases.

While current LCA techniques analyze emissions for processor architectures throughout several phases, such as materials collection/transportation, manufacturing, and operation, there still needs to be additional studies for quantifying some more phases, including their design efforts

(recurring and non-recurring engineering) and possible reuse of the architectures in low-tier operations beyond their first deployment (e.g., in datacenters). This can help estimate and mitigate the total emissions in a holistic and more accurate manner.

5.4.2. Reporting Breakdown of Emissions over Life Cycle Phases and Related Sources

To analyze the carbon footprints of architectures more effectively and develop appropriate countermeasures, it is important to differentiate between embodied carbon and operational carbon in reporting. Embodied carbon refers to the “pre-use” carbon that is already emitted before a computing task begins its operation, while operational carbon refers to the “through-use” footprint. High levels of embodied carbon pose new challenges. For example, if embodied carbon emissions constitute a significant portion, such as 80% [35, 52], reducing operational carbon alone may not improve sustainability significantly. Therefore, operational carbon footprint and embodied carbon footprint over different phases of lifespan need to be reported separately for designing new architectures, along with their major sources for carbon emissions.

Quantifying embodied carbon remains challenging, as industries often consider this as proprietary information, and the attempted quantification may not be holistic [43, 44]. Availability of limited data [43, 44, 50, 53] and double-counting across LCA phases could also lead to inaccuracies in quantifications [54]. Therefore, more reporting and quantification efforts are required to wholistically and accurately report the embodied carbon footprint of the processors.

5.4.3. Metrics for Evaluating Sustainability

A holistically defined carbon footprint over the lifetime of an architecture is usually the primary metric that can guide the design of architecture. However, additional metrics can be needed that can be more insightful or serve as a primary factor for various use cases. For example, embodied carbon is typically significant for processors or even higher than the operational carbon emissions expected through the assumed lifespan. However, based on the lifespan expectancy (for primary deployment and reuse in low-critical tasks) and the target domains for processors, the operational emissions can outweigh embodied carbon emissions [40]. Thus, their quantifications/mitigation should not be marginalized by the significance of embodied carbon. To address this challenge, the design processes can also embed the carbon breakeven point (CBEP) as a metric, which can be calculated in terms of when the carbon footprint of the operational and reuse phase outweighs embodied carbon [40, 55]. The integration of a breakeven analysis in the LCA quantification can guide architects and automatic design processes about how changes in the early lifespan of architecture (e.g., design/manufacturing) could affect the potential breakeven point.

Several phases of the architectural lifecycle can involve one-time carbon emissions related to non-recurring aspects of an architecture. For instance, there would be significant efforts to introduce specialization in novel architectural designs. If carbon emissions throughout the full life cycle for this new architecture are considered as the metric, it could turn out to be significant and outweigh operational benefits for a single architecture. However, this approach would penalize the technology advancements and represent an incomplete quantification, as the new architecture

could likely find broader adoption later (e.g., accelerators [30-32]), and there can be significant benefits through operational phases for next-generation architectures. Therefore, there needs to be separate quantification and reporting for recurring and non-recurring aspects of the life-cycle phases, and additional factors and metrics are required that can reflect the carbon projection based on the likely future reuse of newly introduced technologies for a life-cycle phase.

5.4.4. Using Sustainability Metrics in the Design Process

Sustainability metrics such as carbon footprint could serve as a first-hand design metric to guide the architectural exploration or hardware/software codesign space optimization of an architecture. Just like the performance, energy efficiency, or area efficiency metrics, designers can use these sustainability metrics as either a design constraint or a minimization objective for the overall architectural optimization. For instance, most recently, design space exploration frameworks [56, 57, 58] target data-center designs [56] and architectural choices for extended reality systems [57] or edge processing [58] while considering carbon emissions as a first-hand metric. Like energy–delay product or performance/watt, metrics like carbon–delay product [44, 57] or performance/carbon could also be used as one of the design constraints or objectives instead of using carbon footprint standalone. The designers can also use additional metrics discussed previously, i.e., carbon break-even point (CBEP) or carbon estimates with future utilization (CAREFUL).

5.4.5. The Unique Role of Architects and Architecture Design in Improving Sustainability

From a full LCA perspective, architecture design sits at a unique juncture in the sustainable computing stack, as it can impact sustainability beyond conventional carbon reduction practices like using carbon-free energy or reducing operational energy. In particular,

i) Architectural measurements for LCA of carbon footprint can serve as a high-level sustainability model, providing the necessary data for quantification at higher levels, such as runtime systems and applications, enabling a cross-layer sustainable design and operation.

ii) Architecture designers can play a unique role beyond carbon neutrality practices, as they need to design sustainable architectures in a sustainable way without compromising the QoS requirements of target applications. They can do so by opening the design space of new systems for highly sustainable operations and adopting more sustainable design processes going forward.

Architectures could seamlessly measure and provide feedback on the carbon impact of computing, just like the performance monitoring capabilities found in today's architectures. Sustainability monitoring through the architectures could be invoked at reasonable intervals, e.g., every few days in end-user devices. It can report and help keep track of how effectively applications have utilized the computing fabric. Such architectural-level estimations can be abstract enough to encompass the footprint of lower-level components like manufacturing/operating circuits while providing a breakdown of the footprint for architectural components and their invocations for system-level analysis.

5.4.6. Design Considerations for Sustainable Architectures

When designing high-performance, energy-efficient architectures to meet the QoS requirements, architecture designers must consider the lifespan improvements and execution overheads of resilience mechanisms, as well as sustainability implications of designing and manufacturing specialized architectures as compared to the benefits gained by the usage of specialization over the lifetime of new architectures. Architecture designers can strive for the following goals.

- **Reducing dark silicon:** Dark silicon in architecture can have a detrimental impact on sustainability. The silicon used in the chip, i.e., the total chip area or the number of transistors, directly affects the carbon footprint of the architecture design and its overall lifetime usage. For example, consider a specialized architectural component that may offer up to orders of magnitude higher operational efficiency [30, 31]. If it is not as heavily used as a general-purpose counterpart like a CPU (e.g., thousands of times), the carbon footprint associated with its design and manufacturing can easily outweigh the reduction in the footprint achieved through its execution over the device's lifetime [35, 43]. This example highlights the importance of sustainability awareness when designing domain/application-specific solutions and aiming for generality and reusability whenever possible.

- **Increasing generality and programmability of specialized computing architectures:** Maintaining some level of generality in specialized architectures, and especially their programmability, can increase their usage and adoption, as demonstrated by the success of GPUs over the past decades. If customized-computing architectures provide reasonable programmability, the designers and users can explore ways to efficiently program/process their applications within similar domains on such computing platforms. Thus, they can be more broadly adopted and used over a longer time, amortizing the embodied carbon costs [35, 40] and, thereby, higher sustainability.

- **Selective redundancy:** While incorporating redundancy into a system can increase its lifespan, it comes with energy and performance overheads. Therefore, sustainability throughout the design and operation should be considered, and redundancy should be introduced selectively by considering the related overheads and typical usage of the device for applications versus the potential lifetime improvements gained.

- **Use of open-source vs. proprietary technologies:** When deciding whether to utilize open-source or proprietary technology in architecture/system design, various factors should be considered, including the maturity and reusability of the technology and the sustainability-related life-cycle implications of its deployment. For example, if a vendor aims to build a limited number of systems for in-house usage, the operational efficiency and NRE costs may be primary concerns; using open-source technologies that mitigate such costs and their carbon footprint can be highly beneficial. However, when building massive data centers with thousands or tens of thousands of processors, operational overheads and system longevity become key challenges, favoring off-the-shelf and time-tasted hardware/software systems over an open-source alternative that is less robust. Thus, the design choice is a case-

dependent and challenging task, necessitating research on such quantification and design-space specification and exploration.

- **Resource disaggregation and modular designs:** Moving forward, resource disaggregation in data centers plays a crucial role in sustainability. Often, hardware platforms can only be used for a few to several years, but it does not render all components of a hardware platform unusable at the end of its lifetime. Disaggregating resources [59], such as decoupling computational resources, storage, and networking, can enable effective utilization throughout their lifespan. This approach allows repairing or replacing resources on-demand over time, thereby amortizing embodied carbon overheads through long-lasting operational usage. For example, a recent study [50] estimates a significant reduction in embodied carbon due to reusing pre-designed chiplet IP blocks across several designs through heterogeneous integration technologies.

- **Modular design methodologies:** Designing architectures from components in a modular manner can help design domain-specific architectures effectively and reduce the embodied carbon due to related design efforts. For instance, recent studies show that such methodologies can be applicable for automatically characterizing, simulating, and synthesizing various domain-specific architectures [46, 60-62]. By developing such tasks for design automation of processors in a modular manner, researchers cannot just lower design efforts for a specific architecture template, but they can also find novel architectures for new workloads in an automated and sustainable manner [45].

- **Less intense architectural analysis models:** With the growth of computation- and data-intensive applications, such as large-scale neural networks, using existing computational models for execution characterization and simulations become infeasible from an energy-consumption perspective, which directly impacts recurring/non-recurring design efforts and embodied carbon. Designers can develop or adopt new abstractions/models that require significantly less processing and are generalizable for various architectures while maintaining estimation accuracy. They can enable much faster estimation, characterization, and simulation of processing large workloads.

- **Using AI for improving the sustainability of architectures and their design methodologies:** AI models can help improve the sustainability of architectures through learning from historical data and making better data-driven estimations about the lifetime of architectures (primary deployment and reuse) as well for carbon emissions throughout different life cycle phases. Moreover, AI models can be used to make design methodologies sustainable, especially for designing newer architectures. For example, current approaches use AI models for design placement [63], approximating execution costs of workloads on processors [64, 65] as well as for exploring design parameters of an architecture [66, 67]. The creation of processor design-centric data and AI-based methodologies for characterizing and exploring novel architectures and code optimization [45, 67, 68] could help reduce the design costs and embodied carbon in a significant manner.

5.4.7. End-user Awareness About Carbon Implications of Computing Choices

Currently, less sustainable design choices/practices are challenging to improve as they are not measured. For instance, users may currently prefer streaming services that are relatively cheaper. However, streaming music from distant, large cloud services over a wide-scale network could be extraordinarily expensive in terms of energy consumption and carbon footprint than processing the music on a local device. Providing end-users with sustainability metrics as a first-hand measure can help them and service providers make more sustainable choices. Hyper-scale data centers like Microsoft Azure or Google GCP have recently introduced dashboards that inform their customers about the carbon footprint of their cloud resources used – even approximately. Similar reporting should be enabled to account for the full life-cycle footprint of processing on end-user devices and the edge-cloud continuum. This information may encourage users to opt for buying/renting the devices manufactured with sustainable materials and designed with sustainable design methods and practices that have incurred minimal carbon implications (while having a similar or little more purchase cost and operational cost).

5.4.8. Incentivizing Sustainability of Architectures

Several steps can be taken next to improve the sustainability of designing and operating computers. They include encouraging relevant large-scale business entities to explore carbon-friendly approaches, fostering joint efforts between industry, governments, and research communities, implementing the reporting and budgeting of carbon usage, and promoting the reuse of computational resources.

- **Persuading large-scale business entities to adopt carbon-friendly practices and technologies:** To make a tangible impact, computing professionals, and especially large-scale business entities, should identify the implications of their existing practices/technologies and replace them with carbon-friendly alternatives. For example, chip manufacturing companies can explore advanced material designs to reduce toxic emissions. Similarly, vendors designing processors for heterogeneous computing can develop design methods that enable the automatic search of effective architectures for the target set of applications, improving the sustainability of the design flows and limiting carbon footprints associated with repeated NRE efforts.

- **Joint efforts between industry, government, and academia:** Designing architectures in a sustainable manner can be influenced by global supply-chain challenges and geopolitical issues. In this regard, government agencies should encourage industry partners to develop and adopt more sustainable solutions/technologies while collaborating closely with academic researchers to explore the adoption of newer alternatives developed by them. Industries need to participate in such efforts more actively and help advance them. For instance, the National Science Foundation (NSF) has recently called upon researchers to engage in sustainable computing [70], established a dedicated program [71] to expedite research on

sustainable computing challenges and solutions, and launched a new research program on sustainable digital infrastructures in partnership with VMWare [72].

- **Sustainability budgets for end-user computing:** Recent research and industrial efforts have increasingly focused on improving the energy efficiency of computing and raising awareness about it. For example, in massive-scale and supercomputing infrastructures, allocations to users are now provided based on energy consumption budgets rather than their usage time. Similarly, resource usage reports can be equipped with sustainability metrics, and resource budgeting of computing systems can be provisioned based on carbon footprints.

- **Encouraging sustainability reporting:** For the research community, additional processes incentivizing sustainability need to be introduced. For instance, researchers should be encouraged to report the computing efforts, human efforts, and carbon footprint associated with their research work. This reporting can be initiated by allowing authors of research works to report on sustainability aspects, such as in conjunction with the existing, well-established artifact evaluation process for computing systems research venues. Initially, authors can report the carbon footprint of running evaluations for their artifacts (e.g., for all experiments), focusing on the operational phase, and gradually extend the reporting to include the overall life-cycle metric, encompassing computing resources and human hours spent from conceptualization to deliverable. Accepted papers can include sustainability metrics, and the artifact evaluation process can validate the reported numbers – at least for the operational phase.

- **Improving reusability of resources:** Additional mechanisms and regulations can be established, particularly for large-scale computing infrastructures, that enhance the reusability of computing resources. For instance, recent studies [48, 49] found that computational resources in data centers could be suitable for reasonable reuse for data center workloads and other non-critical tasks, even preserving end-to-end service performance in certain load conditions [49]. Thus, the data center companies/agencies can cautiously and explicitly plan for reusing computational, memory, and networking resources during partial/full upgrades (or deposit them through a relevant agency for reuse in less-performance-critical operations). Likewise, a system can be made available to academic researchers for depositing computing resources purchased from government funding (e.g., unutilized resources bought from NSF CRI grants) in government-managed facilities or educational institutions. However, caution must be exercised in the logistic trade-offs, as recycling resources with low-tier operational quality may prolong the use of energy-inefficient computers. In addition to efforts to promote the reuse of older resources, incentivizing efforts for repairability [73] and making repairs more affordable can be helpful, as it can prolong the use of electronic devices. The efforts and incentives for reusability and repairability cannot just be helpful for prolonged usage of components (and fewer emissions), but they can also be helpful in mitigating shortages of computing resources due to higher chip/semiconductor demands or global uncertainties [74].

5.5. Recommendations to NSF

5.5.1. Raise Awareness

NSF should help increase awareness about sustainable computer architecture designs through co-sponsored workshops at conferences, reach out to industrial partners, and encourage researchers for detailed reporting about carbon-related metrics.

- Justification: Efforts need to be made to make designers aware of the higher embodied carbon and the required efforts for lowering them.

  Recommendation: NSF could help and encourage researchers to organize workshops that can emphasize the sustainability implications of processors and their full life cycle phases to the processor design communities.

- Justification: Currently, industry and practitioners do not report/estimate a detailed breakdown of emissions over the life cycle phases of a processor. For effective quantification and mitigation of carbon implications of architectural techniques, carbon reporting needs to be distinguished between different life-cycle phases of processors as well as their recurring and non-recurring aspects.

  Recommendation: NSF should consider efforts to raise awareness about more detailed reporting of data through workshops as well as through reaching out to industrial partners. This would be important to effectively quantify emissions and determine effective mitigation strategies.

- Justification: Currently, research findings for designing new architectures lack reporting about the computing efforts, human efforts, and carbon footprint associated with the research work. Like research reproducibility, reporting on sustainability metrics needs to be made a part of the research reporting and evaluation process.

  Recommendation: For NSF-funded projects, NSF should encourage computing systems researchers to report sustainability aspects/implications of their proposed research (e.g., CNS and CCF programs under the CISE directorate). For reporting project outcomes and research findings, principal investigators (PIs) can provide supplementary information about the implications of their proposed/developed research on introducing or reducing embodied and operational carbon. For new solicitations from NSF that specifically focus on sustainable computing [70-72], this can be a key required aspect for the proposed/funded projects.

- Justification: Currently, end-users of computing platforms lack information about the carbon implications of their choice of computing.

  Recommendation: NSF should consider working with industrial partners and agencies to encourage reporting about sustainability metrics with end-user devices, which can help increase end-user awareness and more informed decisions.

5.5.2. Infrastructure

- NSF may consider working with government agencies, industry, and educational institutes to establish a common platform for reusing computing resources. Processors and computing equipment purchased from NSF-sponsored projects, large-scale computing infrastructures at universities, and excess resources after upgrades in industrial data centers could be deposited for reuse, e.g., in low-performance-critical tasks at educational institutions.

- NSF could call for the development of open-source infrastructures and computing ecosystems targeting sustainable processing. Like existing programs calling for large-scale tools for community usage, the sustainability-related infrastructure can provide computing systems researchers with a common platform for quantifying and exploring the carbon implications of their new technologies. NSF could also foster joint efforts between industry, governments, and research communities for the same.

5.5.3. Research Topics and Funding

NSF should consider funding dedicated projects to expedite research on sustainable computing challenges and solutions. For example, NSF could consider funding new research projects that focus on the holistic sustainability quantification of processors, effective design metrics, sustainability-aware design methodologies, and efforts for reducing embodied carbon and improving the reuse of processors. The related research topics can include, but are not limited to:

- Wholistic quantification of carbon emissions throughout the full life cycle of processors, including the design, material collection/transportation, manufacturing, operation, reuse, and disposal. Efforts related to dataset curations and case studies are also encouraged.

- Effective sustainability metrics for reducing emissions for various processor life cycle phases. These metrics and their calculations can cover various use cases of processor deployment and different design goals for sustainability, such as total emissions throughout processor lifetime, carbon breakeven point between embodied/operational carbon, carbon estimates with future utilization of technology across several processor generations, etc.

- Carbon-aware design frameworks for general-purpose and domain-specific processors that consider various sustainability metrics and architectural design factors such as domain-specific specialization, dark silicon, programmability, and overheads of fault tolerance mechanisms while improving operational lifetime.

- Efforts for reducing embodied carbon, e.g., with modular and resource-disaggregated architectural designs and high reuse of processors.

- Novel methodologies for lowering carbon footprint related to architecture design methodologies for recurring and non-recurring processor design efforts.

- Data curation and case studies for reuse of processors at scale.

- Improve resource reusability and repairability in large-scale computing infrastructures to reduce electronic waste and carbon emissions.

5.5.4. Education

Integrate discussion about sustainability metrics and case studies into curricular development for undergraduate and graduate courses, e.g., computer architecture, circuits design, computing systems, and related special topics such as reconfigurable computing, accelerators, embedded systems, etc.

5.6. Bibliography

1. J. L. Hennessy and D. A. Patterson, *Computer architecture: a quantitative approach*. Elsevier, 2011.

2. M. D. Hill and A.J. Smith, "Evaluating associativity in CPU caches," in *IEEE Transactions on Computers*, vol. 38, no. 12, pp. 1612-1630, Dec. 1989, doi: 10.1109/12.40842.

3. M. H. Lipasti, C. B. Wilkerson, and J. P. Shen. “Value locality and load value prediction,” in *Proceedings of the seventh international conference on Architectural support for programming languages and operating systems (ASPLOS VII)*: Association for Computing Machinery, New York, NY, USA, 1996, pp. 138–147. doi: 10.1145/237090.237173

4. R. Balasubramonian, D. Albonesi, A. Buyuktosunoglu, and S. Dwarkadas, “Memory hierarchy reconfiguration for energy and performance in general-purpose processor architectures,” in *Proceedings of the 33rd annual ACM/IEEE international symposium on Microarchitecture (MICRO 33)*: Association for Computing Machinery, New York, NY, USA, 2000, pp. 245–257. doi: 10.1145/360128.360153

5. M. D. Powell, A. Agarwal, T. N. Vijaykumar, B. Falsafi and K. Roy, "Reducing set-associative cache energy via way-prediction and selective direct-mapping," *Proceedings. 34th ACM/IEEE International Symposium on Microarchitecture. MICRO-34*, Austin, TX, USA, 2001, pp. 54-65, doi: 10.1109/MICRO.2001.991105.

6. D. Chandra, F. Guo, S. Kim and Y. Solihin, "Predicting inter-thread cache contention on a chip multi-processor architecture," *11th International Symposium on High-Performance Computer Architecture*, San Francisco, CA, USA, 2005, pp. 340-351, doi: 10.1109/HPCA.2005.27.

7. M. K. Qureshi, A. Jaleel, Y. N. Patt, S. C. Steely, and J. Emer, “Adaptive insertion policies for high performance caching”, in Proceedings of the 34th annual international symposium on

Computer architecture (ISCA '07): Association for Computing Machinery, New York, NY, USA, 2007, pp. 381–391. doi: 10.1145/1250662.1250709

8. K. J. Nesbit and J. E. Smith, "Data Cache Prefetching Using a Global History Buffer," *10th International Symposium on High Performance Computer Architecture (HPCA'04)*, Madrid, Spain, 2004, pp. 96-96, doi: 10.1109/HPCA.2004.10030.

9. B. Black, M. Annavaram, N. Brekelbaum, J. DeVale, L. Jiang, G. H. Loh, D. McCaule, P. Morrow, D. W. Nelson, D.l Pantuso, P. Reed, J. Rupley, S. Shankar, J. Shen, C. Webb, "Die Stacking (3D) Microarchitecture," in *2006 39th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO'06)*, Orlando, FL, USA, 2006, pp. 469-479, doi: 10.1109/MICRO.2006.18

10. S. A. Mahlke, D. C. Lin, W. Y. Chen, R. E. Hank and R. A. Bringmann, "Effective Compiler Support For Predicated Execution Using The Hyperblock," In *Proceedings the 25th Annual International Symposium on Microarchitecture MICRO 25*, Portland, OR, USA, 1992, pp. 45-54, doi: 10.1109/MICRO.1992.696999.

11. J. D. Collins, D. M. Tullsen, H. Wang and J. P. Shen, "Dynamic speculative precomputation," *Proceedings. 34th ACM/IEEE International Symposium on Microarchitecture. MICRO-34*, Austin, TX, USA, 2001, pp. 306-317, doi: 10.1109/MICRO.2001.991128.

12. D. A. Jimenez and C. Lin, "Dynamic branch prediction with perceptrons," *Proceedings HPCA Seventh International Symposium on High-Performance Computer Architecture*, Monterrey, Mexico, 2001, pp. 197-206, doi: 10.1109/HPCA.2001.903263.

13. B. R. Rau, "Dynamically scheduled VLIW processors," in *Proceedings of the 26th Annual International Symposium on Microarchitecture*, Austin, TX, USA, 1993, pp. 80-92, doi: 10.1109/MICRO.1993.282744.

14. C. Kozyrakis and D. Patterson, "Overcoming the limitations of conventional vector processors," in *Proceedings of the 30th Annual International Symposium on Computer Architecture, 2003*, San Diego, CA, USA, 2003, pp. 399-409, doi: 10.1109/ISCA.2003.1207017.

15. D. Culler, J. P. Singh, and A. Gupta, *Parallel computer architecture: a hardware/software approach*. Gulf Professional Publishing, 1999.

16. J. Nickolls and W. J. Dally, "The GPU Computing Era," in *IEEE Micro*, vol. 30, no. 2, pp. 56-69, 2010, doi: 10.1109/MM.2010.41.

17. K. V. Craeynest, A. Jaleel, L. Eeckhout, P. Narvaez, and J. Emer. “Scheduling heterogeneous multi-cores through Performance Impact Estimation (PIE),” in *Proceedings of the 39th Annual International Symposium on Computer Architecture (ISCA '12)*, 2012, IEEE Computer Society, USA, pp. 213–224.

18. A. Arunkumar, E. Bolotin, B. Cho, U. Milic, E. Ebrahimi,O. Villa,A. Jaleel, C.J. Wu, and D. Nellans, "MCM-GPU: Multi-chip-module GPUs for continued performance scalability," *2017 ACM/IEEE 44th Annual International Symposium on Computer Architecture (ISCA)*, Toronto, ON, Canada, 2017, pp. 320-332, doi: 10.1145/3079856.3080231.

19. S. Pasricha, and N. Dutt, 2010, *On-chip communication architectures: system on chip interconnect*. Morgan Kaufmann.

20. R. Marculescu, U. Y. Ogras, L. -S. Peh, N. E. Jerger and Y. Hoskote, "Outstanding Research Problems in NoC Design: System, Microarchitecture, and Circuit Perspectives," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, no. 1, pp. 3-21, Jan. 2009, doi: 10.1109/TCAD.2008.2010691

21. S. Herbert and D. Marculescu. 2007. "Analysis of dynamic voltage/frequency scaling in chip-multiprocessors," *Proceedings of the 2007 international symposium on Low power electronics and design (ISLPED '07)*, Portland, OR, USA, 2007, pp. 38-43, doi: 10.1145/1283780.1283790.

22. A. Pathania, Qing Jiao, A. Prakash and T. Mitra, "Integrated CPU-GPU power management for 3D mobile games," in *2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC)*, San Francisco, CA, USA, 2014, pp. 1-6, doi: 10.1145/2593069.2593151.

23. Z. Hu, A. Buyuktosunoglu, V. Srinivasan, V. Zyuban, H. Jacobson, and P. Bose, "Microarchitectural techniques for power gating of execution units," *Proceedings of the 2004 International Symposium on Low Power Electronics and Design*, Newport Beach, CA, USA, 2004, pp. 32-37, doi: 10.1145/1013235.1013249.

24. J. M. Rabaey and M. Pedram, *Low power design methodologies*. Vol. 336. Springer Science & Business Media, 2012.

25. I. Koren and C. M. Krishna, 2020. *Fault-tolerant systems*. Morgan Kaufmann.

26. J. Henkel, L. Bauer, N. Dutt, P. Gupta, S. Nassif, M. Shafique, M. Tahoori, and N. Wehn. 2013. Reliable on-chip systems in the nano-era: lessons learnt and future trends. In Proceedings of the 50th Annual Design Automation Conference (DAC '13). ACM, New York, NY, USA, Article 99, 1–10.

27. J. Ahn, S. Hong, S. Yoo, O. Mutlu and K. Choi, "A scalable processing-in-memory accelerator for parallel graph processing," *2015 ACM/IEEE 42nd Annual International Symposium on Computer Architecture (ISCA)*, Portland, OR, USA, 2015, pp. 105-117, doi: 10.1145/2749469.2750386.

28. P. Chi, S. Li, C. Xu, T. Zhang, J. Zhao, Y. Liu, Y. Wang, and Y. Xie, “PRIME: a novel processing-in-memory architecture for neural network computation in ReRAM-based main memory”, in *Proceedings of the 43rd International Symposium on Computer Architecture (ISCA '16)*, IEEE Press, 2016, pp. 27–39. doi: 10.1109/ISCA.2016.13

29. J. Cong, V. Sarkar, G. Reinman and A. Bui, "Customizable Domain-Specific Computing," in *IEEE Design & Test of Computers*, vol. 28, no. 2, pp. 6-15, 2011, doi: 10.1109/MDT.2010.141.

30. N. P. Jouppi, C. Young, N. Patil, and D. Patterson, "A domain-specific architecture for deep neural networks," in *Communications of the ACM,* vol. 61, no. 9, 2018, pp. 50-59.

31. W. J. Dally, Y. Turakhia, and S. Han, “Domain-specific hardware accelerators,” in *Communications of the ACM*, vol. *63*, no. 7, pp. 48-57.

32. M. Andersch, G. Palmer, R. Krashinsky, N. Stam, V. Mehta, G. Brito and S. Ramaswamy, *NVIDIA hopper architecture in-depth*, 2022. [Online]. Available: https://developer.nvidia.com/blog/nvidia-hopper-architecture-in-depth/

33. S. Dave, R. Baghdadi, T. Nowatzki, S. Avancha, A. Shrivastava and B. Li, "Hardware Acceleration of Sparse and Irregular Tensor Computations of ML Models: A Survey and Insights," in *Proceedings of the IEEE*, vol. 109, no. 10, pp. 1706-1752, Oct. 2021, doi: 10.1109/JPROC.2021.3098483.

34. M. Davies, A. Wild, G. Orchard, Y. Sandamirskaya, G. A. F. Guerra, P. Joshi, P. Plank, and S. R. Risbud, "Advancing neuromorphic computing with loihi: A survey of results and outlook," in *Proceedings of the IEEE*, vol. *109*, no. 5, pp. 911-934, 2021.

35. U. Gupta, Y. G. Kim, S. Lee, J. Tse, H.-H. S. Lee, G.-Y. Wei, D. Brooks, and C.-J. Wu, "Chasing Carbon: The Elusive Environmental Footprint of Computing," *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, Seoul, Korea (South), 2021, pp. 854-867, doi: 10.1109/HPCA51647.2021.00076.

36. D. Kline Jr, N. Parshook, X. Ge, E. Brunvand, R. Melhem, P. K. Chrysanthis, and A. K. Jones, "GreenChip: A tool for evaluating holistic sustainability of modern computing systems," *Sustainable Computing: Informatics and Systems*, vol. *22*, pp. 322-332, 2019.

37. C. J. Wu, R. Raghavendra, U. Gupta, B. Acun, N. Ardalani, K. Maeng, G. Chang, F. Aga, J. Huang, C. Bai, and M. Gschwind, "Sustainable ai: Environmental implications, challenges and opportunities," *Proceedings of Machine Learning and Systems*, *4*, pp.795-813, 2022.

38. B. Li, S. Samsi, V. Gadepally, and D. Tiwari, "Sustainable HPC: Modeling, Characterization, and Implications of Carbon Footprint in Modern HPC Systems," *arXiv preprint arXiv:2306.13177*, 2023.

39. D. Patterson, J. Gonzalez, Q. Le, C. Liang, L.M. Munguia, D. Rothchild, D. So, M. Texier, and J. Dean, "Carbon emissions and large neural network training," *arXiv preprint arXiv:2104.10350,* 2021.

40. S. Sudhakar, V. Sze, and S. Karaman,"Data centers on wheels: emissions from computing onboard autonomous vehicles," in *IEEE Micro*, vol. *43*, no. 1, pp. 29-39, 2022.

41. B. Acun, B. Lee, F. Kazhamiaka, A. Sundarrajan, K. Maeng, M. Chakkaravarthy, D. Brooks, and C.J. Wu, "Carbon Dependencies in Datacenter Design and Management," in *1st Workshop on Sustainable Computer Systems Design and Implementation (HotCarbon '22)*, 2022.

42. R. Bennis, "Life cycle assessment of dell poweredge r740," Dell, 2019, https://corporate.delltechnologies.com/content/dam/digitalassets/active/en/unauth/data-sheets/products/servers/lca poweredge r740.pdf.

43. L. Eeckhout, "A First-Order Model to Assess Computer Architecture Sustainability," in *IEEE Computer Architecture Letters*, vol. *21*, no. 2, pp.137-140, 2022.

44. U. Gupta, M. Elgamal, G. Hills, G. Y. Wei, H. H. S. Lee, D. Brooks, and C. J. Wu, "ACT: Designing sustainable computer systems with an architectural carbon modeling tool," in

Proceedings of the 49th Annual International Symposium on Computer Architecture, pp. 784-799, 2022.

45. S. Dave, and A. Shrivastava, “Design space description language for automated and comprehensive exploration of next-gen hardware accelerators,” in *Workshop on Languages, Tools, and Techniques for Accelerator Design (LATTE’22)*, 2022.

46. S. Dave, A. Marchisio, M. A. Hanif, A. Guesmi, A. Shrivastava, I. Alouani, and M. Shafique, "Special Session: Towards an Agile Design Methodology for Efficient, Reliable, and Secure ML Systems," *2022 IEEE 40th VLSI Test Symposium (VTS)*, San Diego, CA, USA, 2022, pp. 1-14, doi: 10.1109/VTS52500.2021.9794253.

47. Carbon emissions and large neural network training. David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. arXiv preprint arXiv:2104.10350 (2021).

48. A. Tomlinson and G. Porter, “Something Old, Something New: Extending the Life of CPUs in Datacenters,” in *1st Workshop on Sustainable Computer Systems Design and Implementation (HotCarbon ‘22)*, 2022.

49. J. Wang, U. Gupta, and A. Sriraman, “Giving Old Servers New Life at Hyperscale,” in *2nd Workshop on Sustainable Computer Systems (HotCarbon ‘23)*, 2023.

50. V. A. Chhabria, C. C. Sudarshan, S. Vrudhula, and S. S. Sapatnekar, “Towards Sustainable Computing: Assessing the Carbon Footprint of Heterogeneous Systems,” *arXiv preprint arXiv:2306.09434*, 2023.

51. V. Arroyos, M. L. Viitaniemi, N. Keehn, V. Oruganti, W. Saunders, K. Strauss, V. Iyer, and B. H. Nguyen, “A tale of two mice: Sustainable electronics design and prototyping,” in *CHI Conference on Human Factors in Computing Systems Extended Abstracts*, pp. 1-10, 2022.

52. S. Tannu and P. J. Nair, “The dirty secret of SSDs: Embodied carbon,” in *1st Workshop on Sustainable Computer Systems Design and Implementation (HotCarbon ‘22)*, 2022.

53. A. Gandhi, K. Ghose, K. Gopalan, S. R. Hussain, D. Lee, D. Liu, Z. Liu, P. McDaniel, S. Mu, and E. Zadok, “Metrics for sustainability in data centers,” in *1st Workshop on Sustainable Computer Systems Design and Implementation (HotCarbon'22)*, 2022.

54. N. Bashir, D. Irwin, and P. Shenoy, “On the Promise and Pitfalls of Optimizing Embodied Carbon,” in *2nd Workshop on Sustainable Computer Systems (HotCarbon ‘23)*, 2023

55. S. Ollivier, S. Li, Y. Tang, C. Chaudhuri, P. Zhou, X. Tang, J. Hu, and A. K. Jones, "Sustainable AI Processing at the Edge," in *IEEE Micro*, vol. 43, no. 1, pp. 19-28, 2023, doi: 10.1109/MM.2022.3220399.

56. B. Acun, B. Lee, F. Kazhamiaka, K. Maeng, U. Gupta, M. Chakkaravarthy, D. Brooks, and C. J. Wu, “Carbon explorer: A holistic framework for designing carbon aware datacenters,” in *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, pp. 118-132, 2023.

57. M. Elgamal, D. Carmean, E. Ansari, O. Zed, R. Peri, S. Manne, U. Gupta, G. Y. Wei, D. Brooks, G. Hills, and C. J. Wu, “Design Space Exploration and Optimization for Carbon-Efficient Extended Reality Systems,” *arXiv preprint arXiv:2305.01831*, 2023.

58. Y. G. Kim, U. Gupta, A. McCrabb, Y. Son, V. Bertacco, D. Brooks, and C. J. Wu, “GreenScale: Carbon-Aware Systems for Edge Computing,” *arXiv preprint arXiv:2304.00404*, 2023.

59. S. Blagodurov, M. Ignatowski, and V. Salapura, *The Time is Ripe for Disaggregated Systems*, In ACM SIGARCH Blog on Computer Architecture Today, 2021. [Online]. Available: https://www.sigarch.org/the-time-is-ripe-for-disaggregated-systems/

60. J. Weng, S. Liu, V. Dadu, Z. Wang, P. Shah and T. Nowatzki, "DSAGEN: Synthesizing Programmable Spatial Accelerators," in *2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture (ISCA)*, 2020, pp. 268-281, doi: 10.1109/ISCA45697.2020.00032.

61. S. Dave and A. Shrivastava, "Automating the Architectural Execution Modeling and Characterization of Domain-Specific Architectures", in *TECHCON*, 2023.

62. R. Venkatesan, Y. S. Shao, M. Wang, J. Clemons, S. Dai, M. Fojtik, B. Keller, A. Klinefelter, N. Pinckney, P. Raina, and Y. Zhang, “Magnet: A modular accelerator generator for neural networks,” in *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 1-8, 2019.

63. A. Mirhoseini, A. Goldie, M. Yazgan, J. W. Jiang, E. Songhori, S. Wang, Y. J. Lee, E. Johnson, O. Pathak, A. Nazi, and J. Pak *et al.*, “A graph placement methodology for fast chip design,” in *Nature*, vol. *594*, pp. 207-212, 2021.

64. C. Mendis, A. Renda, S. Amarasinghe, and M. Carbin, “Ithemal: Accurate, portable and fast basic block throughput estimation using deep neural networks,” in *International Conference on machine learning*, pp. 4505-4515, 2019.

65. Y. Zhou, X. Dong, T. Meng, M. Tan, B. Akin, D. Peng, A. Yazdanbakhsh, D. Huang, R. Narayanaswami, and J. Laudon, “Towards the co-design of neural networks and accelerators,” in *Proceedings of Machine Learning and Systems*, vol. *4*, pp.141-152, 2022.

66. D. Zhang, S. Huda, E. Songhori, K. Prabhu, Q. Le, A. Goldie, A. Mirhoseini, “A full-stack search technique for domain optimized deep learning accelerators,” in *Proceedings of the 27th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 27-42, 2022.

67. V. Janapa Reddi and A. Yazdanbakhsh, *Architecture 2.0: Why Computer Architects Need a Data-Centric AI Gymnasium*, In ACM SIGARCH Blog on Computer Architecture Today, 2023. [Online]. Available: https://www.sigarch.org/architecture-2-0-why-computer-architects-need-a-data-centric-ai-gymnasium/

68. S. Krishnan, A. Yazdanbakhsh, S. Prakash, J. Jabbour, I. Uchendu, S. Ghosh, B. Boroujerdian, D. Richins, D. Tripathy, A. Faust, and V. Janapa Reddi, “ArchGym: An Open-Source Gymnasium for Machine Learning Assisted Architecture Design,” in *Proceedings of the 50th Annual International Symposium on Computer Architecture*, pp. 1-16, 2023.

69. E. Brunvand, D. Kline and A. K. Jones, "Dark Silicon Considered Harmful: A Case for Truly Green Computing," *2018 Ninth International Green and Sustainable Computing Conference (IGSC)*, Pittsburgh, PA, USA, 2018, pp. 1-8, doi: 10.1109/IGCC.2018.8752110.

70. U.S. National Science Foundation, *Dear Colleague Letter: Design for Sustainability in Computing*, 2022. [Online]. Available: https://www.nsf.gov/pubs/2022/nsf22060/nsf22060.jsp

71. U.S. National Science Foundation, *NSF Program on Design for Environmental Sustainability in Computing (DESC)*, 2022. [Online]. Available: https://www.nsf.gov/pubs/2023/nsf23532/nsf23532.htm

72. VMWare News, *VMware and the NSF Announce Academic Awards*, 2021. [Online]. Available: https://news.vmware.com/sustainability/national-science-foundation-sustainable-infrastructure

73. M. Moeslinger, K. Almasy, M. Jamard, and H. De Maupeou, “Towards an Effective Right to Repair for Electronics,” Publications Office of the European Union, Luxembourg, 2022, doi:10.2760/42722, JRC129957.

74. R. Kanungo, S. Siva, N. Bleier, M. H. Mubarik, L. Varshney, and R. Kumar, “Understanding Interactions Between Chip Architecture and Uncertainties in Semiconductor Supply and Demand,” *arXiv preprint arXiv:2305.11059*, 2023.

Chapter 6: Systems-on-Chips and Integrated Circuits

6.1. Executive Summary

This section summarizes the findings and recommendations of a panel discussion on improving SOC/IC (System-on-Chip/Integrated Circuit) design and fabrication with sustainability in mind. The panel focused on addressing the environmental impacts of computing technologies beyond energy consumption, including greenhouse gas emissions, depletion of rare earth elements, and e-waste. Designing SoCs/ICs with sustainability as a goal involves considering various factors throughout the design process, including power optimization, energy-efficient architectures, system-level optimization, materials selection and end-of-life considerations. The discussion highlighted the need for holistic approaches to sustainability throughout the entire lifecycle of computing devices. The panelists provided insights into various important topics such as consumer behavior, manufacturing processes, reliability, emerging technologies, and open-source initiatives. Based on the discussion, this section presents a set of specific recommendations to the National Science Foundation (NSF) to guide future research and development efforts in sustainable SOC/IC design.

6.2. Background

The rapid growth in computing demands has raised environmental concerns due to increased greenhouse gas emissions [1, 2], depletion of rare earth elements [3, 4], and the generation of e-waste [5]. The disposability of computing systems and consumer behavior toward seeking the latest gadgets exacerbate these issues. Additionally, the manufacturing process and end-of-life disposal of electronic devices contribute significantly to their environmental impact. Therefore, a holistic approach that encompasses the entire lifecycle of computing devices is necessary to address sustainability challenges in SOC/IC design. The following set of questions was used to ignite the panel discussions:

1. How to design SoC/IC with sustainability as a goal?
2. What is the role of SoC/IC design in the Life-cycle analysis/optimization of the computing system?
3. What is the impact on the carbon footprint of optimization of SoC/IC?
4. What is the role of interconnection architectures in computing sustainability? Can we envision designs being informed by LCA, supply chain, or even geopolitics?
5. How can ML help design a sustainable SOC/IC?

6. Can you comment on the role of software/hardware co-design for sustainable SOC design?

7. What are the most promising emerging technologies for future SOC design?

8. How can the increasing use of ML/AI affect sustainable SoC design and its design tools?

9. How should SoC designers apply sustainability principles such as reduce, reuse, recycle/share, etc., for various steps of SoC design and for SoC components?

6.3. Consumer Behavior and Recycling

The panel discussed the challenge of changing consumer behavior to encourage longer usage of SOC/ICs, reducing the environmental impact of frequent device replacements. Consumer behavior plays a significant role in the lifecycle and sustainability of SOC/ICs. The constant pursuit of the latest gadgets and features leads to a culture of frequent device upgrades, resulting in the premature disposal of functional electronics. This trend contributes to increased greenhouse gas emissions, resource depletion, and e-waste generation. Additionally, the lack of recycling further exacerbates the environmental impact, with a significant portion of recyclable materials ending up in landfills. Addressing this issue requires a multifaceted approach that focuses on promoting extended usage of SOC/ICs and encouraging responsible recycling practices.

6.3.1. Design Considerations for SoC/IC

The panelists discussed the creation of energy-efficient and long-lasting SOC/IC architectures that meet the computational needs of future applications. Emphasize the benefits of using devices with optimized designs, highlighting their ability to handle emerging technologies *without the need for frequent upgrades.* There was a recognition of the need to promote software/hardware co-design methodologies that optimize algorithms, power management, and task management policies. This approach can lead to energy savings and improved performance, making devices *more appealing for long-term use.*

6.3.2. Sustainability in Manufacturing of SoC/IC

The panel also discussed the needed support for research on sustainable manufacturing processes for SOC/ICs. This includes developing device and circuit-level models to *measure the carbon footprint of different fabrication steps.* Encourage the adoption of environmentally friendly techniques such as *reducing energy-intensive processes and optimizing material usage.* Another idea was to explore the development of *"sustainable" design variants of standard logic cells* that prioritize energy efficiency, recyclability, and reduced environmental impact. The idea of *how to use carbon credits* in electronics manufacturing was also discussed. It is well known that simply planting trees to offset carbon is not practical as we will run out of space in the process. Therefore, we must be creative in this process.

6.4. Architecture and Algorithm Optimization

To address sustainability concerns, there is a need to focus on developing computing architectures that are both high-performing and energy-efficient, particularly for demanding workloads like AI and big data applications. By achieving higher throughput while reducing the number of units and energy consumption, it becomes possible to meet computing requirements while minimizing the environmental impact. Emerging technologies such as in-memory computing and 3D/M3D integration offer promising avenues for improving performance and energy efficiency.

6.4.1. In-Memory Computing

The panel discussed the significant potential of in-memory computing, which has shown the ability to outperform traditional non-Von Neumann architectures by orders of magnitude [6-8]. In-memory computing can offer significant improvements in data processing speed and energy efficiency, thereby enabling more sustainable computing solutions. However, to fully achieve the potential of in-memory computing, we need to develop novel memory technologies and architectures that are optimized for in-memory computing, which includes exploring materials and devices with improved data storage and retrieval capabilities. However, the reuse of existing fabrication or manufacturing foundries where novel architectures and systems leverage the reuse of existing technologies like DRAM or SRAM should not be neglected as the reuse of existing foundries greatly fosters reuse, one of the cornerstones of sustainability.

6.4.2. 3D/M3D and 2.5D Integration

The panel emphasized the advantages of 3D/M3D integration techniques that enable the stacking of multiple layers of components, resulting in better performance and energy efficiency compared to conventional 2D systems since such integration can lead to shorter interconnect lengths, reduced power consumption, and improved data transfer rates [9]. Chip designers, packaging, and fabrication experts need to collaborate to develop reliable and cost-effective 3D/M3D integration methodologies. 3D/M3D is known to have lower yields compared to 2D or 2.5D integration. 2.5D integration with interposers can provide comparable performance to that of M3D while improving yield and enabling the reuse of older and cleaner manufacturing technologies in older nodes for the interposer [10, 11]. The interposer is a large die that allows chiplets or dielets to be integrated into sockets. Interposers are either entirely passive or partially active with a limited number of active transistors or devices while providing wiring real estate for dense interconnect architectures. Chiplets or dielets are smaller chips/SoCs that can be pre-designed or pre-packaged to be used from a library of chiplets to be integrated into a larger system-on-interposer for a system-in-package. This methodology encourages reuse and sustainability and needs to be considered for sustainable computing systems.

6.5. Reliability and Lifetime Extension

To ensure the sustainability of SOC/ICs, it is crucial to consider their reliability in the face of environmental variations, workload variations, and other factors [12]. By extending the lifetime of SOC/IC systems, the environmental costs associated with manufacturing and end-of-life can be amortized. Drawing parallels from the automotive industry, where cars are often used for more than ten years, similar longevity can be achieved for SOC/ICs. For this, we need to come up with *proactive reliability mechanisms* like the following.

6.5.1. Early Defect Detection

The panel discussed the need for research on proactive reliability mechanisms that can detect defects early in SOC/IC systems. Like sensors in cars that notify users or designers about potential issues, incorporating similar mechanisms into SOC/IC designs can enable suitable mitigation actions to prevent catastrophic failures in the future. Extensive studies are needed on advanced fault detection and diagnosis techniques that leverage real-time monitoring, machine learning, and data analytics to identify potential issues before they cause system failures.

6.5.2. Lifelong Testing

The panel also discussed the knowledge gap regarding reliability in SOC/ICs and proposed potential remedies. It is important to periodically test to detect defects and enable timely mitigation actions. However, there has been relatively limited exploration of proactive in-field testing. The analogy drawn to cars illustrates the concept: just as the engine light on a car's dashboard indicates a possible issue with the engine, a similar approach could be applied to SOC/ICs. Proactive in-field testing would identify failing or soon-to-fail components, enabling early intervention to prolong the chip's lifespan. The panelists advocated for the adoption of 2.5D chiplet-based architectures as a potential solution. Such architectures offer the ability to replace or bypass defective components, making them a promising avenue for further investigation.

6.5.3. Reconfigurable SOC/ICs for Extended Lifetime

The panel discussed the development of reconfigurable SOC/ICs that offer the flexibility to vary hardware functionality to a certain extent. This adaptability is essential to address future challenges that are currently unknown. For instance, in the case of cryptography accelerators, if an underlying algorithm is defeated, a reconfigurable SOC/IC would allow for modifications to support a more secure algorithm, thereby extending its useful life. More research is needed on reconfigurable hardware architectures, such as field-programmable gate arrays (FPGAs) and programmable logic devices (PLDs), which enable runtime modifications to hardware functionality without the need for complete hardware replacement.

6.5.4. ASICs vs. Reconfigurable SOC/ICs

The panel also addressed the debate surrounding the environmental impact of application-specific integrated circuits (ASICs) compared to reconfigurable SOC/ICs. While ASICs may be more energy-efficient for specific applications, their lack of flexibility can lead to the design and production of multiple units for different applications, resulting in higher costs and environmental impact. Reconfigurable SOC/ICs can be optimized for multiple target applications as needed, which reduces the need for separate hardware designs, minimizes waste, and improves resource utilization, leading to a more sustainable SOC/IC ecosystem.

6.6. Manufacturing Process and Carbon Footprint

The cost of manufacturing SOC/ICs has become increasingly challenging with scaling, particularly regarding energy usage [14]. The panel discussed a bottom-up approach to develop device and circuit-level models specifically tailored for measuring the *carbon footprint* of a process node. This approach would be an extension of the methodology used in Design Technology Co-Optimization (DTCO). There is a need for research to carefully quantify the carbon emissions associated with specific fabrication steps, considering factors such as the use of extreme UV lithography versus multiple rounds of quadruple patterning lithography. Accurate carbon footprint measurements can provide valuable insights into the environmental impact of different manufacturing approaches.

6.6.1. Developing Sustainable Logic Cells

The panel discussed the *bottom-up* development of “sustainable” design variants of standard logic cells. These sustainable design variants should focus on reducing energy consumption, waste generation, and the use of rare materials during SOC/IC manufacturing. Another direction is to encourage research on novel circuit architectures and design techniques that prioritize energy efficiency, recyclability, and the use of environmentally friendly materials.

6.6.2. Sustainability in 2.5D and 3D Integration

As heterogeneous integration is proliferating, it is important to develop a *top-down* approach to incorporate sustainability considerations into the process of 2.5D and 3D integration. This will enable the replacement or bypassing of defective units without the need to discard the entire architecture, thus extending the lifetime of the device. There is also a need for research on advanced testing, diagnosis, and reconfiguration techniques that facilitate the identification and isolation of faulty components in 2.5D and 3D integrated systems. This will allow for targeted repairs or replacements, minimizing waste and improving resource utilization.

6.7. Holistic view of an SoC/IC product phases

Most of the work on sustainability usually focuses on making the operational phase more energy efficient, but there is a need to address the environmental impact of the manufacturing and end-

of-life phases. The SOC/IC industry can effectively address the increasing carbon footprint and environmental impact associated with SOC/ICs. Implementing sustainable manufacturing practices, exploring reuse and extended lifetimes, and establishing responsible end-of-life strategies are crucial steps toward achieving a more environmentally friendly and sustainable SOC/IC ecosystem.

6.7.1. Manufacturing Phase

The panel recognizes the need to assess and quantify the environmental impact of SOC/IC manufacturing processes, considering factors such as energy consumption, resource utilization, waste generation, and emissions. For this, we need to encourage the exploration of innovative manufacturing technologies, such as cleaner fabrication processes, recycling of materials and reduced use of hazardous substances, as well as support the development of energy-efficient manufacturing equipment and techniques to improve overall sustainability in the production of SOC/ICs.

6.7.2. Reusing Old Hardware

The panel discussed the possibility of reusing SOC/ICs, especially those that are still functional but no longer suitable for cutting-edge applications, and recognizing that not all applications require the latest computing systems, and repurposing old SOC/ICs for less demanding tasks can significantly extend their lifetime. Research is needed into the development of methodologies, standards, and compatibility frameworks that facilitate the reuse of SOC/IC components across different applications and domains.

6.7.3. Redistribution of Components

One idea suggested in the discussion was to promote initiatives that facilitate the redistribution of SOC/IC components from outdated or decommissioned systems to areas where computing power requirements are lower, such as in educational institutions, non-profit organizations, or low-resource settings. This approach could be supported through collaborations between industry, academia, and non-profit organizations to establish networks and platforms for the efficient exchange and redistribution of SOC/IC components.

6.7.4. Responsible Disposal and Recycling

Sustainable end-of-life strategies require the development of environmentally responsible and economically viable methods for the disposal and recycling of SOC/ICs that have reached the end of their usable life. Specialized recycling facilities could be established, equipped to handle SOC/IC components, ensuring proper handling and extraction of valuable materials while minimizing environmental harm. This approach could be supported through research on innovative recycling techniques, including the recovery of rare earth elements and other valuable resources from SOC/ICs.

6.7.5. Material Selection and Design for Recycling

The panel discussed the use of materials in SOC/IC design that are more recyclable or environmentally friendly, enabling easier and more efficient recycling processes. The industry should adopt design practices that facilitate the disassembly and separation of components during end-of-life processing and explore the development of standardized labeling or identification systems that provide information on the recyclability and environmental impact of SOC/IC components.

6.7.6. Reduce, Reuse, Recycle/Share and Open Source

SoC designers can apply sustainability principles at various steps of SoC design and for SoC components.

To *Reduce*, designers should focus on reducing power consumption, resource usage, and environmental impact throughout the design process. This can include optimizing power management techniques, using low-power components, and minimizing the use of hazardous materials. To *Reuse,* designers should try to reuse existing design blocks, IP cores, and verification suites to minimize unnecessary duplication of effort and reduce the environmental impact associated with new designs. To *Recycle,* there is a need to have recyclability in mind by considering the selection of materials that can be easily recycled or using components with higher recyclability. Designers should also consider end-of-life scenarios and design for easy disassembly and recycling. Finally, to *Share,* we should foster collaboration and knowledge-sharing within the design community to promote sustainable practices and innovations. Open-source initiatives and shared design resources can contribute to reducing redundancy and environmental impact.

During the discussion, the panelists highlighted the issue of limited availability of open-source intellectual property (IPs) and electronic design automation (EDA) tools. Currently, the process of designing a new SOC/IC involves extensive simulations on cloud-based EDA tools, which often require significant time and resources. Although many IPs already exist, they are not openly accessible, resulting in a repetitive and carbon-intensive design process for each new project. This approach contributes to a substantial environmental impact due to the energy consumption, cooling requirements, and communication overhead of cloud computing infrastructure. The need for further development of open-source EDA tools was emphasized, together with promoting the reuse of software and IP blocks to help mitigate the economic and environmental costs associated with manufacturing. However, economic challenges exist in realizing such a model, as it may not align with the profitability objectives of EDA or IC design companies.

6.7.7. Emerging Technologies and Sustainable Computing

Several emerging technologies show promise for future SoC design with sustainability considerations. 3D integration enables the stacking of multiple chip layers, reducing interconnect

length, improving performance, and potentially reducing power consumption. Neuromorphic computing is inspired by the human brain and can provide energy-efficient solutions for specific tasks, leveraging principles of low-power computation and event-driven processing. Approximate computing allows controlled errors within acceptable bounds, thus trading off accuracy for energy savings, making it suitable for certain applications. Emerging non-volatile memory technologies, such as resistive RAM (RRAM) or phase-change memory (PCM), offer the potential for energy-efficient and high-density on-chip storage solutions. Energy harvesting, such as RF, solar cells, or vibration sensors, can enable autonomous power generation and reduce reliance on external power sources.

6.7.8. AI/ML's Role in Sustainable SoC/IC Design and Fabrication

Artificial Intelligence/Machine learning (AI/ML) can aid in designing sustainable SoCs/ICs in several ways. First, in design optimization, AI/ML techniques can automate design space exploration, identifying optimal configurations and parameters that result in energy-efficient and sustainable SoCs. Second, for power management, AI/ML algorithms can dynamically adapt power management strategies based on workload characteristics, leading to more efficient power consumption [14]. Third, fault detection and reliability AI/ML techniques can be used to detect and predict failures in SoCs, enabling proactive measures to improve reliability and reduce resource waste. Finally, for design tool optimization, AI/ML can enhance design tools by automating tasks, improving accuracy, and reducing the time and effort required for design iterations.

6.8. Recommendations to NSF

6.8.1. Sustainable SoC Design Consortium

Help create a consortium that brings together industry, academia, and government agencies to collaborate on research, development, and knowledge sharing in sustainable SoC/IC design. This consortium can provide funding, resources, and a platform for researchers and industry experts to work together on specific sustainability-focused projects.

6.8.2. Sustainable SoC Design

Allocate specific funding programs that prioritize research on sustainable SoC design. This funding can support projects focusing on low-power architectures, power optimization techniques, and power management strategies, with the goal of minimizing energy consumption in computing systems.

6.8.3. Design Tools for Sustainability

Support the development of design tools and methodologies that enable designers to evaluate the sustainability impact of their SoC designs. This can include tools for estimating carbon

footprint, energy efficiency analysis, and life-cycle assessment. Funding can be provided to research groups working on such tools to enhance their development and adoption.

6.8.4. Collaboration with Material Science and Engineering

Foster collaboration between SoC designers and material scientists/engineers to explore the development of sustainable materials and components for SoCs. Encourage research on environmentally friendly materials, such as biodegradable or recyclable substrates, low-impact manufacturing processes, and novel materials with reduced energy requirements.

6.8.5. Encourage Design for Upgradability and Repairability

Promote research and development of SoCs that are designed for upgradability and repairability. This can include modular designs that allow for component-level upgrades or repairs, reducing electronic waste and extending the lifespan of computing devices.

6.8.6. Support Open-Source Hardware Initiatives

Provide funding and resources to support open-source hardware initiatives focused on sustainable SoC/IC design. Open-source hardware encourages collaboration, knowledge sharing, and the reuse of design blocks, reducing duplication of effort and enabling more sustainable design practices.

6.8.7. Integration of ML/AI in Sustainable Design

Support research on the use of AI/ML techniques for sustainable SoC/IC design. This can involve the development of AI/ML algorithms for power optimization, system-level energy management, and intelligent decision-making to enhance sustainability metrics in SoC design.

6.8.8. Establish a Sustainable Design Certification

Collaborate with industry partners and standards organizations to develop a certification program for sustainable SoC design. This certification can provide a framework for assessing and benchmarking the sustainability performance of SoCs, guiding designers and companies towards more sustainable design practices.

6.8.9. Foster International Collaboration for Supply Chain Sustainability

Facilitate international collaborations to address supply chain sustainability in SoC design. Partner with international organizations and academic institutions to share best practices, establish guidelines for responsible sourcing of materials, and promote fair labor practices throughout the supply chain.

6.8.10. Support Interdisciplinary Research Projects

Encourage interdisciplinary research projects that explore the intersection of sustainability, SoC/IC design, and other domains such as renewable energy, IoT, or smart cities. Funding and resources can be allocated to projects that investigate the synergies and potential for sustainable design solutions in these areas.

References

1. *https://physicsworld.com/a/the-huge-carbon-footprint-of-large-scale-computing/*
2. *https://www.it.ox.ac.uk/article/environment-and-it*
3. *https://www.sciencenews.org/article/rare-earth-elements-properties-technology*
4. *https://www.anl.gov/article/rare-earth-supply-disruptions-have-longrange-impacts-computer-model-shows*
5. *https://www.epa.gov/international-cooperation/cleaning-electronic-waste-e-waste*
6. *K. Roy, I. Chakraborty, M. Ali, A. Ankit and A. Agrawal, "In-Memory Computing in Emerging Memory Technologies for Machine Learning: An Overview," 2020 57th ACM/IEEE Design Automation Conference (DAC), San Francisco, CA, USA, 2020, pp. 1-6, doi: 10.1109/DAC18072.2020.9218505.*
7. *Carlos Ríos et al. In-memory computing on* a photonic platform.Sci. Adv.5,eaau5759(2019).DOI:10.1126/sciadv.aau5759.
8. Sathwika Bavikadi, Purab Ranjan Sutradhar, Khaled N. Khasawneh, Amlan Ganguly, and Sai Manoj Pudukotai Dinakarrao. 2020. A Review of In-Memory Computing Architectures for Machine Learning Applications. In Proceedings of the 2020 Great Lakes Symposium on VLSI (GLSVLSI '20). Association for Computing Machinery, New York, NY, USA, 89–94. https://doi.org/10.1145/3386263.3407649.
9. https://www.sciencedirect.com/topics/engineering/three-dimensional-integrated-circuits
10. N. E. Jerger, A. Kannan, Z. Li and G. H. Loh, "NoC Architectures for Silicon Interposer Systems: Why Pay for more Wires when you Can Get them (from your interposer) for Free?," 2014 47th Annual IEEE/ACM International Symposium on Microarchitecture, Cambridge, UK, 2014, pp. 458-470, doi: 10.1109/MICRO.2014.61.
11. M. M. Ahmed, M. S. Shamim, N. Mansoor, S. A. Mamun and A. Ganguly, "Increasing interposer utilization: A scalable, energy efficient and high bandwidth multicore-multichip integration solution," *2017 Eighth International Green and Sustainable Computing Conference (IGSC)*, Orlando, FL, USA, 2017, pp. 1-6, doi: 10.1109/IGCC.2017.8323583.
12. S. Kundu, "Managing reliability of integrated circuits: Lifetime metering and design for healing," 2016 IEEE 25th Asian Test Symposium (ATS), Hiroshima, Japan, 2016, pp. 227-227, doi: 10.1109/ATS.2016.80.
13. https://hal.science/hal-04112708v1/file/semiconductor_ghg.pdf
14. https://www.researchgate.net/publication/371512509_Towards_Green_Automated_Machine_Learning_Status_Quo_and_Future_Directions

Chapter 7: Devices and Materials for Computing

7.1. Executive Summary

This chapter discusses the environmental impact of semiconductor manufacturing, highlighting the carbon footprint associated with the production of computing components. It delves into the questions and challenges, including the role of the semiconductor fabs in the sustainable manufacturing of computers and how different materials used in semiconductor devices, such as silicon, germanium, and gallium arsenide, impact sustainability and carbon emissions. We detail the challenges and emissions associated with different stages in the semiconductor fabrication process, including scope 1, scope 2, and scope three emissions. Then, we describe several strategies proposed by the panel for sustainable computing, including reducing energy consumption during fabrication, utilizing older CMOS nodes, exploring biomaterial-based computing, incorporating non-CMOS materials and devices, and emphasizing material lifecycle analysis. Further, interdisciplinary collaboration and a deep convergence of knowledge between fields are advocated for finding holistic solutions to sustainability challenges. Lastly, we also emphasize the role of stakeholders, such as fab designers, material production companies, and transistor architects, in adopting sustainable practices and incentivizing their efforts.

7.2. Background

The proliferation of IoT, edge, and cloud computing has led to tens of billions of processing cores as well as memory, network, and storage modules in use today. Each of these components makes use of several million to billions of devices, such as transistors, that are built using a variety of CMOS-compatible and other types of materials. The semiconductor fabs that fabricate these components expend a significant amount of embodied carbon and other resources (e.g., water use during manufacture) that depend on the devices and materials being used by the computing components. It has been estimated that over 80% or more of the carbon footprint of many electronics comes from their manufacturing [1][2]. This is partly due to the extremely high energy costs of semiconductor manufacturing equipment (e.g., 2000 °F furnaces, specialized lithography tools, etc.). The use of computing components also expends operational carbon and other resources (e.g., water used for cooling large data center computing facilities), which depends on the devices and materials used in the computing components [3].

Semiconductor devices are the key components in integrated circuits (ICs), such as computer processors, microcontrollers, and memory chips (such as NAND flash and DRAM) that are present in everyday electrical and electronic devices. The most common semiconductor device in the world is the MOSFET (metal–oxide semiconductor field-effect transistor), also called the MOS transistor. MOSFETs account for at least 99.9% of all transistors, and there have been an estimated 13 sextillion MOSFETs manufactured between 1960 and 2018 [4]. Many other types of semiconductor devices are also used in electronic systems, including diodes, bipolar junction transistors, thyristors, and photocells.

All transistor types can be used as the building blocks of logic gates, which are fundamental in the design of digital circuits. In digital circuits such as processors, transistors act as on-off switches. Transistors used for analog circuits (e.g., amplifiers, oscillators) do not act as on-off switches; rather, they respond to a continuous range of inputs with a continuous range of outputs. Circuits that interface or translate between digital circuits and analog circuits are known as mixed-signal circuits and are also widely used. Power semiconductor devices are discrete devices or integrated circuits intended for high current or high voltage applications. Power integrated circuits combine IC technology with power semiconductor technology.

From a materials perspective, silicon (Si) is the most utilized substance for the fabrication of computer chips. Silicon is a naturally occurring semiconductor, and typical beach sand has a large concentration of this element. The injection of imperfections into silicon can change its electrical characteristics, a technique known as doping. Owing to these properties, it is an effective substance for the fabrication of transistor devices. Silicon's combination of low raw material cost, relatively simple processing, and a useful temperature range makes it currently the best compromise among the various competing materials. Silicon used in semiconductor device manufacturing is currently fabricated into boules that are large enough in diameter to allow the production of 300 mm (12 in.) wafers [5][6]. The silicon wafers that serve as the foundation of computing chips are composed of silicon, while the metal wires used to connect the sections of circuitry are typically made of aluminum or copper [7].

Several other elements are also employed in the design of computing chips. Germanium (Ge) was a widely used early semiconductor material, but its thermal sensitivity makes it less useful than silicon. Today, germanium is often alloyed with silicon for use in very-high-speed SiGe devices (such as those made by IBM) [8]. Gallium arsenide (GaAs) is also widely used in high-speed devices [9][10], but so far, it has been difficult to form large-diameter boules of this material, limiting the wafer diameter to sizes significantly smaller than silicon wafers, thus making mass production of GaAs devices significantly more expensive than silicon. Gallium Nitride (GaN) is gaining popularity in high-power applications, including power ICs, light-emitting diodes (LEDs), and RF components due to its high strength and thermal conductivity [11]. Compared to silicon, GaN's band gap is more than three times wider at 3.4 eV, and it conducts electrons 1,000 times more efficiently [11]. Silicon carbide (SiC) has found some application as the raw material for blue LEDs and is being investigated for use in semiconductor devices that can withstand very high operating temperatures and environments with the presence of significant levels of ionizing radiation [12]. Various indium compounds (indium arsenide [13], indium antimonide [14], and indium phosphide [15]) are also being used in LEDs, solid-state laser diodes, and photodetectors. Selenium sulfide is being studied in the manufacture of photovoltaic solar cells [16].

Semiconductor fabrication facilities that utilize the abovementioned materials to build devices and integrate them into components that become part of computing and electronic systems have a significant environmental impact. As an example, TSMC alone uses more than 5% of all of Taiwan's electricity, according to figures from Greenpeace [17], with estimates indicating a rise to 7.2% in 2022, and it used about 63m tons of water in 2019. The company's water use became a controversial topic during Taiwan's drought in 2019, the country's worst in a half-century, which

pitted chipmakers against farmers [17]. In the US, a single fab, Intel's 700-acre campus in Ocotillo, Arizona, produced nearly 15,000 tons of waste in the first three months of 2021, about 60% of it hazardous [17]. It also consumed 927m gallons of fresh water, enough to fill about 1,400 Olympic swimming pools, and used 561m kilowatt-hours of energy [17].

The environmental impacts from semiconductor facilities involve scope 1, scope 2, and scope 3 overheads [18]. Scope 2 emissions, which represent the highest proportion of greenhouse gases (GHG) from semiconductor companies, are linked to the energy required to run their extensive production facilities [18]. The sources of these emissions include tool fleets containing hundreds of manufacturing tools, such as lithography equipment, ion implanters, and high-temperature furnaces; large clean rooms requiring climate and humidity control with overpressure and particle filtration extensive subfab facilities for gas abatement, exhaust pumps, water chillers, and water purification [18]. Scope 1 emissions, which also significantly add to fabs' GHG emission profile, arise from process gases used during wafer etching, chamber cleaning, and other tasks. These gases, which include PFCs, HFCs, NF3, and N20, have high global-warming potential (GWP); they rise as manufacturing node size shrinks [18]. Scope 1 emissions may also arise from high-GWP heat transfer fluids that may leak into the atmosphere when fabs use them in chillers to control wafer temperature during manufacturing processes [18]. Together, scope one and Scope 2 emissions account for 80% of emissions from semiconductor fabrication facilities. Additional emissions may come from upstream scope three sources, such as suppliers, chemicals, and raw materials, or from transportation to customer facilities [18]. These upstream emissions generally account for about 20 percent of fabs' GHG profile. There are also many toxic materials used in the semiconductor fabrication process. These include poisonous elemental dopants, such as arsenic, antimony, and phosphorus; poisonous compounds, such as arsine, phosphine, tungsten hexafluoride, and silane; and highly reactive liquids, such as hydrogen peroxide, fuming nitric acid, sulfuric acid, and hydrofluoric acid.

The round table on sustainable devices and materials posed the following questions:

- What is the role of semiconductor fabs in the sustainable manufacture of devices?
- What devices provide greater sustainability benefits than others?
- What materials provide greater sustainability benefits than others?
- How do we assess the sustainability benefits of specific devices and materials?
- Where do we get data to help make sustainable device and material selection decisions?

7.3. Opportunities and Challenges

7.3.1. Role of Sustainable Practices at Semiconductor Fabs

While some semiconductor companies have created ambitious targets for reducing their emissions and remaining on a 1.5°C pathway, many others have been less ambitious. The pressure to act may soon increase, however, since businesses across industries are now scrutinizing emissions along their entire supply chain—and in many cases, semiconductor

companies will account for a substantial amount of them. Already, some of the semiconductor industry's most important end customers, including Apple, Google, and Microsoft, have committed to reaching net-zero emissions for their full value chain and set aggressive timelines for achieving their goals [19][20][21]. Some semiconductor companies have responded by setting their own emissions goals. For instance, Infineon plans to reduce greenhouse gas (GHG) emissions by 70 percent by 2025, compared with its 2019 baseline, and aspires to reach carbon neutrality for emissions directly under its control by the end of 2030 [22]. Intel recently committed to net-zero GHG emissions in its global operations by 2040 and has targeted achieving 100 percent use of renewable electricity as an interim milestone in 2030 [23]. Several semiconductor players have also committed to science-based targets, including STMicroelectronics [24], NXP [25], and UMC [26].

To achieve substantial emissions reductions and accelerate decarbonization, semiconductor companies must focus on fabricating more sustainable devices and materials, as well as many additional steps. There is a need to reduce energy consumption during fabrication, which may come from new approaches for reducing tool-related energy consumption—for instance, by upgrading and replacing tools with more energy-efficient ones and implementing smart control systems to enable coupling and regulation of facilities and tools. The use of energy from renewable sources, greater energy efficiency of buildings, and replacing existing lighting in fabs with LED fixtures may aid in this goal. To reduce energy consumption in clean rooms, new mechanisms are needed that can simultaneously employ strategies for reducing air pressure, increasing humidity, limiting air exchange in unused areas, or eliminating leaks in air-supply lines. Semiconductor fabs may be able to reduce emissions by adjusting process parameters, such as temperature and chamber pressure. Process engineers often overlook this lever and instead focus solely on yield during optimization efforts, partly because they lack the knowledge and operational experience required to identify strategies for reducing GHG emissions. Similarly, the suppliers involved in daily tool operations and maintenance may prioritize cost and uptime targets over energy savings. If fabs address knowledge gaps and collaborate more closely with tool suppliers, they may improve emissions—for instance, by simultaneously optimizing yield and energy consumption during cleaning protocols. Fabs must also lower emissions by switching to chemicals that have a lower environmental impact. However, this is not easy; for example, it can be difficult to get suppliers on board with their plans. In addition, developing new solutions is both costly and time-consuming, as is the process for qualifying new chemicals on existing processes and tools. While some fabs have already implemented some major improvements, such as increased use of NF3, many other shifts, including the replacement of NF3 with F2 or ozone, are still nascent. Another direction involves capturing unutilized process gases and by-products through various means, such as membrane separation, cryogenic recovery, adsorption, and desorption. The fabs can then refine them into pure process gases that can be used again, potentially reducing process-gas emissions. For this approach to become economically viable, researchers will need to address major challenges related to the separation of process-gas outflows and purification.

7.4. Use of New, Scaled CMOS Transistors/Devices for Centralized versus Ubiquitous Computing

To make the vision of ubiquitous computing a reality, the industry has pushed for employing scaled CMOS transistors/devices in IoT systems to attain higher performance generation by generation. However, this approach often provides higher computing capabilities to IoT systems than they need. It also dramatically increases the overall carbon footprint of manufacturing and distributing such IoT systems. This is mainly because the newer/scaled CMOS transistors incur a substantially high embodied carbon footprint due to the increased number of processing steps and complexity of lithography patterns [27]. Moreover, the relatively low runtime of IoT systems does not let this high embodied carbon footprint break even with the operational carbon footprint of the systems [28], which in turn results in a lot of wasted carbon.

One solution to this problem could be to leverage the economy of scale of data centers by employing scaled CMOS transistors/devices in subsystems of data centers to commoditize computing. The main idea could be as follows. People would perform the sort of computations that they need without as much redundancy or inefficiency or wastefulness in the computation—so, moving to a model where, when you want to compute something, it goes to the most efficient place, which would probably be a highly centralized cloud-based computing center with the lowest possible operational carbon footprint. The trend of making everybody's smartphone, or by extension every microcontroller and every IoT device, as powerful as a PC now is unsustainable. That must shift. The only thing we need for highly intense computing is a terminal with an interface (e.g., a keyboard or voice dictation or something similar); the actual computing happens on the cloud, and we only get the result back. In such ecosystems, terminal computing systems can be made employing old CMOS transistors/devices and technology nodes that have a low overall carbon footprint, even if that means the performance and deployed count of these terminal systems are not at the highest achievable level. Avoiding the deployment of IoT terminal systems whenever possible would help minimize or reduce the environmental cost (carbon cost) of manufacturing and transporting/distributing these systems. The commoditization of computing through the use of centralized data centers also has an added advantage. For instance, by employing energy-efficient design practices (e.g., mixing more renewable energy sources in the power grid of the data center) the operational carbon cost of such data centers can be dramatically reduced.

7.4.1. Aim for the Longevity of Old CMOS Nodes and Move Away from Inventing New “Dirtier” Nodes

This may be achieved in numerous ways. For instance, to digitize the planet with IoT terminal systems, the older CMOS nodes, such as 28 nm and 32 nm nodes, might be ideal due to their relatively low embodied carbon footprints [28]. As another example, chiplets made of older CMOS nodes may be integrated together in a 2.5D or 3D assembly using older substrates to gain energy efficiency through the heterogeneous operation of the chiplet assembly, consequently reducing the operational carbon footprint [29]. Moreover, the use of older CMOS nodes, such as 65 nm

and 32 nm nodes, in chiplet-based systems may reduce investments in materials and infrastructure compared to the new process nodes. As a result, chiplet systems employing older CMOS nodes can reduce the embodied carbon as well [29], thereby proving to be more environmentally sustainable.

Another interesting way may be to employ old CMOS nodes, such as 45 nm and 32 nm SOI CMOS nodes, to manufacture silicon photonics-based computing systems. Recent breakthroughs have made it possible to employ 45 nm and 32 nm SOI CMOS nodes with a “zero-change” approach (without changing the native CMOS process) to fabricate silicon photonic circuits [30]-[32]. Efficient reuse of existing CMOS manufacturing infrastructure may reduce the embodied carbon footprint of silicon photonics-based computing systems. It may also raise new opportunities for reducing the operational carbon footprint due to the established energy efficiency benefits of silicon photonic circuits for computing and communication [33]-[37].

Atop aiming for the longevity of old CMOS nodes, the reusability of computing subsystems into multiple lifecycles should also be targeted. For that, modularization of computing systems and chiplet assemblies seems to be the solution. For example, it could be very promising for sustainability if we could change a few components of cell phones on demand instead of mandatorily changing the entire cell phones in the future. A similar approach can be taken with chiplet assemblies as well. Chiplets realized using older CMOS nodes can be disintegrated from old assemblies, and then a new life can be brought into them by re-integrating them into new chiplet assemblies. However, several technical challenges remain. For example, how to achieve such flexibility in chiplet assemblies? What specific substrates can be used to integrate chiplets of possibly different CMOS nodes?

7.4.2. Eco-inspired Design: Computing Using Biomaterials Directly from the Environment

Nature has evolved over millions to hundreds of millions of years to solve some of the same exact issues we now face regarding sustainability. Therefore, maybe we can look to nature for solutions; explicitly, the world of industrial ecology, which employs the ideas of biomimicry and eco-inspired designs, might offer some promising and compelling solutions.

As a potential solution, DNA or molecular computing is something that people in academia, and to some extent in industry, have been pursuing. And that's the idea that we could organize computation, not in the form of a silicon lattice where electrons are being segregated in a silicon lattice, but through other forms, by molecular confirmations, by changing the shapes of molecules. We can organize computation as we've learned how to do it over the past century using this kind of medium. So, what is the impetus? In principle, this form of computation is one that could be more directly embedded in the environment. Our bodies are biological materials, and with the knowledge that we've gained from computer science and computer engineering, perhaps we could learn to choreograph the sort of state changes that now occur with electrons in a silicon lattice in the form of molecular conformation changes but to do so embedded in the material. Such computation may not be particularly fast or powerful, but it can be embedded directly into the

environment. Moreover, with chemical sensing, one might be able to sense a change in the environment, affect the computation, and respond to it. This could be in the environment outside of our bodies or the environment inside our bodies.

Although computing with DNA is touted to be very energy efficient, the technology is still very futuristic. All of that has been purely in the academic realm, with no practical significance whatsoever. But recently, there's been a move towards building storage systems out of DNA [38]. DNA storage is still a long way off, but there's an investment now on the part of the industry [39]. With the industry buying in, DNA storage and DNA computing might move towards becoming more practical. When it comes to realizing DNA computing, one can approach this from two ends. On the one hand, theoretically, there is proof that, in the form of DNA, one can construct very powerful computers that have a remarkably low environmental footprint. For example, our brains; biological evolution has really created wonderful processing devices in terms of the watts per operation. Also, in terms of interaction with the environment, it could just be based on calories and excrement of some kind. Very intelligent creatures exist in the environment, and by many measures, they are better than our computers.

But, in approaching this from a more practical standpoint, DNA computing is very far from being deployed in any practical sense. People currently use liquid handling robots, these big machines with servo motors, to move small amounts of liquid. To just store and process a few megabytes of data requires bathtubs full of reagents. Thus, the current state-of-the-art is very, very far. DNA computing and storage will start to have an angle in a purely invented environment if people want to collect information (the sensing aspect) from the environment directly in the environment, store it directly in the environment, and then actuate based upon the environment. Those sites of applications in the environment could become mainstream to custom design molecular circuits for very specific functions in situ, and from there, we will have to progress to make this to be a full-blown alternative for computation. In another complementary direction to DNA-based storage and computing, an integrated circuit from wood or cellulose was developed in 2015 [40] that bears the promise of non-toxic waste materials at the end-of-life of computing infrastructure.

7.4.3. Role of Non-CMOS Materials and Devices

The term sustainable computing has become effectively synonymous with low-power/low-energy computing. However, for computing to be truly sustainable, all phases of the system life cycle must be considered. In addition to addressing the use-phase energy consumption issue, it is very crucial to pay due attention to the considerable energy consumption and environmental impacts of semiconductor fabrication. Current research indicates that fabrication is responsible for a significant factor of the energy utilized by a wide range of computing systems, from battery-powered embedded systems to data center servers, throughout their life cycle. The trends of technology scaling coupled with developing hybrid fabrication solutions for integration of emerging computing solutions that require non-CMOS materials and devices, while beneficial for use-phase power consumption, exacerbate these increasing environmental impacts from fabrication.

Examples of culprits include emerging technological trends, e.g., 3D integration for silicon and hybrid processes that integrate non-volatile memories [41]. Realizing these technologies requires the integration of devices that are made of hybrid CMOS and non-CMOS elements/materials [41]. Considering an example from magnetic storage, data is stored as the resistance of the magnetic tunneling junction (MTJ) device in magnetic memory cells. An MTJ device employs many elements, including Co, Fe, B, etc. [41]. These elements do not commonly exist in conventional CMOS processes, though they are very popular materials in the fabrication of magnetic devices, e.g., recording heads. Experience with a leading foundry (reported in [41]) shows that the contamination qualification process alone may require between nine and 12 months for hybrid integration processes that include magnetic memory. The required fabrication effort will also be significantly increased in the most popular low-cost design cycles for these technologies as follows. First, the back-end CMOS devices are fabricated in the foundry; second, the preparation of the magnetic device is conducted at a third-party facility and integrated atop the back-end CMOS devices; and finally, the wafer is sent back to the original foundry to complete top-level interconnects and pads.

To complicate the process, protective cover layers are required whenever the wafer is transferred between the CMOS foundry and the magnetic foundry. The foundry reports that the cleaning process alone for each of the required additional layers in a hybrid process, including 3D CMOS, increases the disbursed gases (CO_2 and volatile organic compounds) as well as wastewater generation. The mechanics and potential complexity of the non-CMOS fabrication stages can further exacerbate environmental impacts. For instance, an energy-efficient realization of MTJ devices requires a round or octagonal shape of MTJ cells. Introducing such diagonal/octagonal shapes requires a more complicated CMOS lithography process, which reduces yield [41]. A reduced yield could increase environmental impact by requiring additional units to be fabricated (including their impacts) to meet the need.

The biggest fear in our fight for sustainability is that we have a limited amount of time. We must solve these problems at an accelerated pace before irreversible changes in the climate set in. Also, we have to solve these problems with no negative impact on the environment whatsoever. So, we probably do not have the luxury of bringing in new technologies, devices, or materials. Because when we bring in new technologies, even if they're compatible with CMOS or even if we try to make them compatible with CMOS, anything new that we bring in may have an overwhelming initial manufacturing cost and environmental impact, which could be unacceptable. Therefore, whatever we may do to achieve the same sustainability goals, we have to think very conservatively about using CMOS. We have to try to see if we can do what the new technologies support and promise with existing technologies, both at the device/transistor level as well as the circuit level.

7.4.4. Role of Material Lifecycle Analysis

The design of sustainable computing systems requires systematic and holistic thought processes. In the absence of such solid processes for decision-making, choosing a particular material and technology for merely a one-faceted sustainability advantage might become regrettable with

unintended consequences. The choice of a specific material can oppositely impact the sustainability outcomes of the extraction, usage, and recycling phases of the material. For instance, one of the major shifts that we've seen over time is the shift from plastics in components to light metals like aluminum and magnesium. Aluminum has a strong recycling infrastructure, but it takes a lot of energy to manufacture at the front end [43]. We see this also with nanotechnology, nanoscale materials being used in electronics. These are materials that might confer advantages later because they're light, they perform well, and they can be recycled. In some cases, they can reduce the energy impact of using the product, but they have a huge energy footprint in their material extraction. Therefore, to avoid such undesirable outcomes, the ramifications of specific material choices should be considered holistically. For that, a collaboration between computing and sustainability practitioners is key.

To avoid making regrettable decisions, there are methodologies like lifecycle assessment and a whole suite of other methodologies [44], along with a wide range of material management metrics [45], that could be looked at to put numbers around these decisions. However, these methodologies and metrics may not be one-size-fits-all [45]. The selection of a specific set of methodologies and metrics could be very context-specific.

7.4.5. Role of Deep Convergence of Knowledge and Infusion Among Fields

What we need to focus on is to push for this idea of a change in thinking away from a singular form of solution. A singular solution will probably never change things in a way that leads to meaningful impacts for sustainability. This is because sustainability challenges at their inherent core are complex and messy wicked problems. Therefore, a holistic, systemic approach to sustainability is required. A deep Convergence of knowledge between disciplines is required. It is important to understand not just how other computer folks think about sustainability but also how material scientists (who are at the ground level of developing some of these materials) and social scientists (who know about human behavior) think about sustainability. Interdisciplinary workshops on sustainable computing provide perfect opportunities to bring these domains together, and leverage what each of them understands about their respective areas, and create something new about sustainability that is greater than the sum of the parts.

7.5. Stakeholders and Incentives

The following stakeholders should be incentivized to adopt sustainable practices. One way of appropriating incentives is to provide financial benefits. For instance, for-profit organizations could receive tax benefits, while employees could receive salary bonuses and/or formal recognition for their efforts.

Fab designers and process engineers. If fabs address knowledge gaps and collaborate more closely with tool suppliers, they may improve emissions—for instance, by simultaneously optimizing yield and energy consumption during cleaning protocols. Fabs must also lower emissions by switching to chemicals that have a lower environmental impact. Process engineers

may be able to reduce emissions by adjusting process parameters, such as temperature and chamber pressure. For that focus should not be solely on yield during optimization efforts. Instead, appropriate emphasis should be given to imparting to the process engineers the knowledge and operational experience required to identify strategies for reducing GHG emissions.

Material production companies. Material production companies should aim to extract materials from more sustainable resources. If sustainable material resources are unknown, the companies should fund expeditions to discover new sources while minimizing the adverse environmental impacts of mining/procuring materials from existing resources. They should also meticulously quantify and create a database of the environmental impacts of their activities (e.g., in terms of carbon and greenhouse gas footprints). Such databases should be used to spread awareness and made available publicly to help catalyze further research.

Post-Moore era transistor architects. Architects of novel transistors in the post-Moore era should aim to invent materials and transistor architectures that are environmentally sustainable. For that, the objective of transistor innovation and design should be to minimize the overall carbon footprint (embodied + operational footprints) while maximizing the transistor lifetime. Architects should think very conservatively about using old CMOS nodes and try to see if old CMOS nodes support and promise what new CMOS nodes could do, both at the device/transistor level as well as the circuit level. The need to do this arises from the urgency of the climate change problem, which must be resolved at an accelerated pace before irreversible changes in the climate set in. So, we probably do not have the luxury of bringing in new technologies, devices, or materials. Because when we bring in new technologies, even if they're compatible with CMOS or even if we try to make them compatible with CMOS, anything new that we bring in may have an overwhelming initial manufacturing cost and environmental impact, which could be unacceptable.

7.6. Recommendations to NSF

NSF may consider the following recommendation to encourage sustainability in the domain of materials and device research for computing.

15. Create programs for cross-cutting research for the discovery or invention of new non-electronic computing substrates. High-risk, Radical, and revolutionary research, even without traditionally rigorous preliminary results, should be encouraged.
16. Collaboration and infrastructure proposals for novel materials and devices should be incentivized to enable material scientists and Microelectronics experts to collaborate with the CISE community.
17. Consortiums may be established to exchange ideas and form collaborative teams spanning multiple disciplines to enable research across several disciplines to be necessary to create an impact in this domain.
18. Life-cycle awareness of new materials for computing needs to be encouraged. A conscious choice for more sustainable substrates or materials and devices needs to be rewarded by NSF, even if a thorough analysis and modeling is not present in proposals or research.

7.7. Bibliography

1. Udit Gupta, Young Geun Kim, Sylvia Lee, Jordan Tse, Hsien-Hsin S Lee, Gu-Yeon Wei, David Brooks, and Carole-Jean Wu. 2021. Chasing Carbon: The Elusive Environmental Footprint of Computing. In 2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA). IEEE, 854–867.

2. Paul Teehan and Milind Kandlikar. 2013. Comparing Embodied Greenhouse Gas Emissions of Modern Computing and Electronics Products. Environmental Science & Technology 47, 9 (May 2013), 3997–4003.

3. S. B. Boyd, Life-Cycle Assessment of Semiconductors. New York, NY: Springer, 2012. [Online]. Available: http://link.springer.com/10.1007/978-1-4419-9988-7

4. David Laws “13 SEXTILLION & COUNTING: THE LONG & WINDING ROAD TO THE MOST FREQUENTLY MANUFACTURED HUMAN ARTIFACT IN HISTORY,” April 2018. [Online]. Available:https://computerhistory.org/blog/13-sextillion-counting-the-long-winding-road-to-the-most-frequently-manufactured-human-artifact-in-history/

5. BOSE (2013). IC Fabrication Technology. McGraw Hill Education (India) Pvt Ltd. p. 53. ISBN 978-1-259-02958-5

6. Dhanaraj, Govindhan; Byrappa, Kullaiah; Prasad, Vishwanath; Dudley, Michael (2010). Springer Handbook of Crystal Growth. ISBN 9783540747611. Retrieved February 25, 2017.

7. J.-P. Colinge (29 February 2004). Silicon-on-Insulator Technology: Materials to VLSI: Materials to Vlsi. Springer Science & Business Media. p. 12. ISBN 978-1-4020-7773-9

8. J.-P. Colinge (29 February 2004). Silicon-on-Insulator Technology: Materials to VLSI: Materials to Vlsi. Springer Science & Business Media. p. 12. ISBN 978-1-4020-7773-9

9. Dennis Fisher; I. J. Bahl (1995). Gallium Arsenide IC Applications Handbook. Vol. 1. Elsevier. p. 61. ISBN 978-0-12-257735-2.

10. Dennis Fisher; I. J. Bahl (1995). Gallium Arsenide IC Applications Handbook. Vol. 1. Elsevier. p. 61. ISBN 978-0-12-257735-2.

11. Kiarash Ahi, "Review of GaN-based devices for terahertz operation," Opt. Eng. 56(9) 090901 (11 September 2017) https://doi.org/10.1117/1.OE.56.9.090901

12. Liu, L., Liu, A., Bai, S. et al. Radiation Resistance of Silicon Carbide Schottky Diode Detectors in D-T Fusion Neutron Detection. Sci Rep 7, 13376 (2017). https://doi.org/10.1038/s41598-017-13715-3

13. Yongli Wang, Bojie Ma, Jian Li, Zhuoliang Liu, Chen Jiang, Chuanchuan Li, Hao Liu, Yidong Zhang, Yang Zhang, Qi Wang, Xinyu Xie, Xiaolang Qiu, Xiaomin Ren, and Xin Wei, "InAs/GaAs quantum-dot lasers grown on on-axis Si (001) without dislocation filter layers," Opt. Express 31, 4862-4872 (2023)

14. Jinchao Tong, Heng Luo, Fei Suo, Tianning Zhang, Dawei Zhang, and Dao Hua Zhang, "Epitaxial indium antimonide for multiband photodetection from IR to millimeter/terahertz wave," Photon. Res. 10, 1194-1201 (2022).

15. Crosnier, G., Sanchez, D., Bouchoule, S. et al. Hybrid indium phosphide-on-silicon nanolaser diode. Nature Photon 11, 297–300 (2017). https://doi.org/10.1038/nphoton.2017.56

16. Rasmus Nielsen, Tomas H. Youngman, Andrea Crovetto, Ole Hansen, Ib Chorkendorff, and Peter C. K. Vesborg. "Selenium Thin-Film Solar Cells with Cadmium Sulfide as a Heterojunction Partner," ACS Applied Energy Materials 2021 4 (10), 10697-10702. DOI: 10.1021/acsaem.1c01700.

17. Pádraig Belton, "The computer chip industry has a dirty climate secret", September 2021. [Online]. Available: https://www.theguardian.com/environment/2021/sep/18/semiconductor-silicon-chips-carbon-footprint-climate

18. G. Protocol, "Greenhouse Gas Protocol," Mar. 2021. [Online]. Available: https: //www.wri.org/initiatives/greenhouse-gas-protocol

19. Google. "Sustainability Reports & Case Studies." Google Sustainability. Accessed March 11, 2023. https://sustainability.google/reports/.

20. Apple. "Environment." Apple. Accessed March 11, 2023. https://www.apple.com/environment/.

21. Microsoft Sustainability. "Microsoft 2022 Environmental Sustainability Report." Accessed August 1, 2023. https://www.microsoft.com/en-us/corporate-responsibility/sustainability/report.

22. Infineon, Infineon Technologies. "CSR Reporting - Infineon Technologies." Accessed March 11, 2023. https://www.infineon.com/cms/en/about-infineon/sustainability/csr-reporting/.

23. Intel. "Corporate Social Responsibility." Accessed August 1, 2023. https://www.intel.com/content/www/us/en/corporate-responsibility/corporate-responsibility.html.

24. STMicroelectronics Sustainability Report 2022. "STMicroelectronics Sustainability Report 2022 - Home." Accessed August 1, 2023. https://sustainabilityreports.st.com/sr22.

25. "Sustainability." Accessed August 1, 2023. https://www.nxp.com/company/about-nxp/sustainability-and-esg:CORP_SOCIAL_RESP.

26. "Sustainability Reports - UMC." Accessed August 1, 2023. https://www.umc.com/en/Download/corporate_sustainability_reports.

27. Garcia Bardon, M., P. Wuytens, L.-Å. Ragnarsson, G. Mirabelli, D. Jang, G. Willems, A. Mallik, A. Spessot, J. Ryckaert, and B. Parvais. "DTCO Including Sustainability: Power-Performance-Area-Cost-Environmental Score (PPACE) Analysis for Logic Technologies."

In 2020 IEEE International Electron Devices Meeting (IEDM), 41.4.1-41.4.4, 2020. https://doi.org/10.1109/IEDM13553.2020.9372004.

28. Gupta, Udit, Mariam Elgamal, Gage Hills, Gu-Yeon Wei, Hsien-Hsin S. Lee, David Brooks, and Carole-Jean Wu. “ACT: Designing Sustainable Computer Systems with an Architectural Carbon Modeling Tool.” In Proceedings of the 49th Annual International Symposium on Computer Architecture, 784–99. ISCA ’22. New York, NY, USA: Association for Computing Machinery, 2022. https://doi.org/10.1145/3470496.3527408.

29. Chhabria, Vidya A., Chetan Choppali Sudarshan, Sarma Vrudhula, and Sachin S. Sapatnekar. “Towards Sustainable Computing: Assessing the Carbon Footprint of Heterogeneous Systems.” arXiv, June 15, 2023. https://doi.org/10.48550/arXiv.2306.09434.

30. Stojanović, Vladimir, Rajeev J. Ram, Milos Popović, Sen Lin, Sajjad Moazeni, Mark Wade, Chen Sun, et al. “Monolithic Silicon-Photonic Platforms in State-of-the-Art CMOS SOI Processes,” Optics Express 26, no. 10 (May 14, 2018): 13106–21. https://doi.org/10.1364/OE.26.013106.

31. Moazeni, S., A. Atabaki, D. Cheian, S. Lin, R. J. Ram, and V. Stojanovic. “Monolithic Integration of O-Band Photonic Transceivers in a ‘Zero-Change’ 32nm SOI CMOS.” In 2017 IEEE International Electron Devices Meeting (IEDM), 24.3.1-24.3.4, 2017. https://doi.org/10.1109/IEDM.2017.8268452.

32. Sun, Chen, Mark Wade, Michael Georgas, Sen Lin, Luca Alloatti, Benjamin Moss, Rajesh Kumar, et al. “A 45 Nm CMOS-SOI Monolithic Photonics Platform With Bit-Statistics-Based Resonant Microring Thermal Tuning.” IEEE Journal of Solid-State Circuits 51, no. 4 (April 2016): 893–907. https://doi.org/10.1109/JSSC.2016.2519390.

33. R. Meade, S. Ardalan, M. Davenport, J. Fini, C. Sun, M. Wade, A. Wright-Gladstein, and C. Zhang, “TeraPHY: A High-Density Electronic-Photonic Chiplet for Optical I/O from a Multi-Chip Module,” in 2019 Optical Fiber Communications Conference and Exhibition (OFC), Mar. 2019, pp. 1–3.

34. P. Kennedy, “Lightmatter Mars SoC AI Inference Using Light,” Aug. 2020. [Online]. Available: https://www.servethehome.com/lightmatter-mars-soc-ai-inference-using-light/

35. H. Zhou, J. Dong, J. Cheng, W. Dong, C. Huang, Y. Shen, Q. Zhang, M. Gu, C. Qian, H. Chen, Z. Ruan, and X. Zhang, “Photonic matrix multiplication lights up photonic accelerator and beyond,” Light: Science & Applications, vol. 11, no. 1, p. 30, Feb. 2022, number: 1 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41377-022-00717-8

36. B. J. Shastri, A. N. Tait, T. Ferreira de Lima, W. H. P. Pernice, H. Bhaskaran, C. D. Wright, and P. R. Prucnal, “Photonics for artificial intelligence and neuromorphic computing,” Nature Photonics, vol. 15, no. 2, pp. 102–114, Feb. 2021, number: 2 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41566-020-00754-y

37. V. J. Sorger, “Photonic tensor cores for machine learning,” in Emerging Topics in Artificial Intelligence 2020, vol. 11469. SPIE, Aug. 2020, p. 114690Q. [Online]. Available:

https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11469/114690Q/Photonic-tensor-cores-for-machine-learning/10.1117/12.2568234.full

38. D. Soloveichik et al., “DNA as a universal substrate for chemical kinetics,” PNAS, vol. 107, no. 12, pp. 5393– 5398, 2010.

39. S. K. Tabatabaei et al., “DNA punch cards for storing data on native DNA sequences via enzymatic nicking,” Nature Communications, vol. 11, no. 1, p. 1742, 2020.

40. URL: https://www.smithsonianmag.com/innovation/these-new-computer-chips-are-made-from-wood-180955471/

41. Jones, Alex, Yiran Chen, William Collinge, Haifeng Xu, Laura Schaefer, Amy Landis, and Melissa Bilec. Considering Fabrication in Sustainable Computing. IEEE/ACM International Conference on Computer-Aided Design, Digest of Technical Papers, 2013. https://doi.org/10.1109/ICCAD.2013.6691120.

42. M. A. Yao, T. G. Higgs, M. J. Cullen, S. Stewart, and T. A. Brady, “Comparative assessment of life cycle assessment methods used for personal computers.” Env. Sci. & Tech., vol. 44, no. 19, pp. 7335–46, Oct. 2010.

43. Raabe, Dierk, Dirk Ponge, Peter J. Uggowitzer, Moritz Roscher, Mario Paolantonio, Chuanlai Liu, Helmut Antrekowitsch, et al. “Making Sustainable Aluminum by Recycling Scrap: The Science of ‘Dirty’ Alloys.” Progress in Materials Science 128 (July 1, 2022): 100947. https://doi.org/10.1016/j.pmatsci.2022.100947.

44. “Life Cycle Assessment Methodology - an Overview | ScienceDirect Topics.” Accessed August 1, 2023. https://www.sciencedirect.com/topics/engineering/life-cycle-assessment-methodology.

45. “Materials Management KPIs & Metric Definitions | OpsDog.” Accessed August 1, 2023. https://opsdog.com/categories/kpis-and-metrics/materials-management.

Chapter 8: Workforce Development, Education and Curriculum

8.1. Executive Summary

In order to enforce the multi-faceted approach to sustainability in computing discussed in prior sections and to consolidate sustainable practices, it is of paramount importance to develop a workforce that is trained in sustainable practices. With such a workforce available, sustainability will gradually become more normalized and widely accepted. Having these goals in mind, this panel observes and evaluates existing practices in sustainability efforts as well as proposes newer approaches to build a sustainable workforce consisting of students, faculty, and researchers. This entails a thorough discussion and analysis of multiple proposed approaches, both discrete and overlapping, to find optimal solutions to training and developing a sustainable workforce in the upcoming decades.

8.2. Background

Sustainable computing is an important goal that has received increasing attention. The scope of sustainability concerns has expanded over the years. The current understanding of sustainable computing includes several elements:

- Full lifecycle costs and effects.
- Economic effects caused by the design, manufacturing, operation, and disposal of the computing system.
- Physical and biological constraints and effects.
- Ethical and legal considerations related to the effects of the computing system.
- Consideration of sustainability in engineering design.

Potential harm resulting from a failure to take into account sustainability includes physical or psychological harm to individuals or groups, physical damage or lack of access to property, and damage to the commons.

Potential advantages to sustainability-enhanced computing include improved health, lower costs, and enhanced revenue.

8.3. Current Status

We do not know of systematic studies of the study of sustainability in the computer science or computer engineering curricula. Anecdotal reports indicate that sustainability receives little to no attention in the typical CS/CE curriculum. In the existing curricular structure, there is a noticeable pattern of increasing siloing of knowledge as a student climbs the ladder of educational hierarchy:

from high school to undergraduate level to post-graduate level. Students and industry practice would benefit from a more holistic awareness of topics such as sustainability and environmental impacts among the students alongside their primary focus or interest in studies.

8.4. Curricular Methods

A variety of sustainability-enhancing techniques have been identified. Certain materials or methods may be avoided or used to enhance the sustainability of manufacturing, operation, and disposal. Design for recyclability is a related approach that takes into account the effort required to recycle a device thanks to its components and materials, ease of disassembly, etc. A complementary approach is designed for longevity and maintainability; legal considerations such as right-to-repair play into the engineering approaches used for longevity and maintainability. The role of artificial intelligence in these approaches is not yet clear.

Curricular techniques can be applied to teach specific sustainability methods and reinforce the importance of sustainability at multiple levels in undergraduate [2, 3] as well as graduate programs [3, 9]. Major design projects, especially Capstone projects for undergraduates and project courses for graduate students, can incorporate sustainability goals and practice sustainability methods. Introductory courses can introduce sustainability to students to motivate their continued attention to the topic throughout their education [1, 3].

- Capstone (undergraduate) + Project-based courses (graduate)
- Introductory courses.
- Build into curriculum.

8.5.1. K-12

Universities can work with K-12 schools to identify ways to incorporate sustainable computing into their curricula, for example, by relating computing to other aspects of sustainability via integration in traditional coursework [8] and projects [6]. A prime example of such an initiative is the 1M NSF-funded, 3-year project from UC San Diego that aims to promote sustainable practices in Computer Science among K-12 students across three school districts in San Diego, CA [7].

8.6.1. Undergraduate Studies

The curriculum should be updated to motivate sustainable practices among the students from a social point of view. Foundational topics—environmental science, ethics, social sciences, public policy, law, economics, physics, chemistry, *etc*.---help students to contextualize the design and manufacturing techniques they learn in CS/CE [2]. However, our curricula are already tightly packed, and incorporating these background topics into curricula will require trade-offs.

Sustainability modules can be incorporated into existing courses at all levels [1, 3]. This may be performed by including relevant aspects of sustainability studies in various existing undergraduate-level STEM courses. This will be accompanied by real-life

examples/demonstrations of sustainable practices and why these are essential in classroom discussions [5]. A specific example of such an initiative would be the consideration of sustainability-oriented topics, such as energy efficiency, end-of-life cycle cost, and the environmental impact of fabrication, etc., in the curriculum of VLSI design and test, computer architecture, IoT, digital design, and high-performance computing coursework.

Motivation toward sustainable development can be encouraged by a shift in the primary objectives/priorities of the coursework. The course assignments/projects may be redesigned to include sustainability considerations [1, 3]. More sustainable and energy-efficient ones may be incorporated into project goals and grading criteria. One great example of such initiatives is Green Software designing and Greencodes, which report quantifiable impacts of environment-friendly and sustainable coding practices [5].

Capstone is an important venue for the interdisciplinary experience of sustainable computing. Projects can also demonstrate and quantify the impact of sustainable development [4]. This can be performed by developing improved simulation platforms that report quantitative implications of sustainable design practices, essentially creating 'Digital Twins' of research projects/developments with a simulated projection of sustainability factors.

Ultimately, sustainable computing should be incorporated into the ACM and IEEE model curricula, including both computing-specific topics and background.

8.6.3. Graduate Studies

Graduate studies are an important venue for sustainability studies [9]. Students can build upon their foundational knowledge to understand how to incorporate sustainability into all aspects of computing systems. Sustainability can be reflected in both coursework and research. Some M.S. thesis and Ph.D. dissertation projects may be pivoted on the theme of sustainability. Other theses/dissertations may focus on sustainability in a chapter.

8.5. Broadening Participation in Computing by DEI Efforts

Novel and innovative efforts for broadening participation in computing through programs and outreach activities are much needed not only in research but also in educational activities. One approach could be to broaden the student or researcher archetypes that are appealed to in our programs. Traditionally, the archetype that computing sciences have appealed to are tech enthusiasts or problem solvers. However, educators can leverage the broad goals of sustainability in computing to appeal to students of other archetypes, such as those with a service mentality for the greater good or those with a creative mindset. This will enhance the diversity in student populations from different backgrounds as that influences their archetypes. This should be augmented with resource investment to conduct outreach to traditionally underrepresented populations groups in computing, such as women, gender, racial, and ethnic minorities.

8.6. Sustainable Domestic Semiconductor Industry

A critical part of developing a sustainable domestic semiconductor industry, under the recent CHIPS act, is to develop a diverse workforce skilled and knowledgeable in sustainable practices [10, 11]. This includes training the extant workforce as well as building a sustainability-aware workforce in collaboration with education institutions [10] via well-laid-out training programs and upgrading of curricula. Alongside, funding for R&D in sustainability in semiconductor technologies should be encouraged.

8.7. Recommendations to NSF

1. CISE should consider tracking sustainability efforts pursued by grants, including intellectual merit, outreach, and broader impacts.
2. CISE should consider incorporating sustainability topics into programs that are not solely focused on sustainability.
3. CISE should consider providing sustainability to review panels for consideration in their deliberations.
4. CISE should collaborate with professional societies (IEEE, ACM, etc.) to develop curricular guidelines related to sustainable computing. These collaborative efforts could link to research and outreach efforts in grants.
5. CISE should encourage investigators to incorporate sustainability into their outreach plans at all levels: K-12, undergraduate, and graduate.
6. CISE should collaborate with other directorates on the full spectrum of sustainable computing topics, including research, outreach, and education.

8.8. Bibliography

1. J. L. Aurandt and E. C. Butler, ‘Sustainability Education: Approaches for Incorporating Sustainability into the Undergraduate Curriculum,’ *Journal of Professional Issues in Engineering Education and Practice*, vol. 137, no. 2, pp. 102–106, 2011.

2. National Academies of Sciences, Engineering, and Medicine. 2020. *Strengthening Sustainability Programs and Curricula at the Undergraduate and Graduate Levels*. Washington, DC: The National Academies Press. https://doi.org/10.17226/25821.

3. Electrical and Computer Engineering,” College of Engineering - The University of Iowa, https://ece.engineering.uiowa.edu/undergraduate/focus-areas/ece-focus-area-sustainablity-computer (accessed Oct. 10, 2023).

4. Hayashi, V. T., Arakaki, R., Almeida, F. V., & Ruggiero, W. V. (2023). The Development of Sustainable Engineering with PjBL during the COVID-19 Pandemic. *International*

journal of environmental research and public health, *20*(5), 4400. https://doi.org/10.3390/ijerph20054400

5. João Saraiva, Ziliang Zong, and Rui Pereira. 2021. Bringing Green Software to Computer Science Curriculum: Perspectives from Researchers and Educators. In Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V. 1 (ITiCSE '21). Association for Computing Machinery, New York, NY, USA, 498–504. https://doi.org/10.1145/3430665.3456386

6. S. L. Robinson and J. A. Mangold, *Implementing Engineering and Sustainability Curriculum in K-12 Education*, vol. 5: Education and Globalization. 11 2013, p. V005T05A033.

7. L. Meyer06/02/15, "UC San Diego develops sustainable computer science courses for K-12," THE Journal, https://thejournal.com/articles/2015/06/02/uc-san-diego-develops-sustainable-computer-science-courses-for-k12.aspx (accessed Oct. 10, 2023).

8. W. Z. Bernstein, A. Ramani, X. Ruan, D. Ramanujan, and K. Ramani, *Designing-In Sustainability by Linking Engineering Curricula With K-12 Science Projects*, vol. 7: 9th International Conference on Design Education; 24th International Conference on Design Theory and Methodology. 08 2012, pp. 305–312.

9. Electrical and Computer Engineering," McCormick School of Engineering, https://www.mccormick.northwestern.edu/electrical-computer/academics/graduate/masters/electrical-engineering/ (accessed Oct. 10, 2023).

10. Workforce development," NIST, https://www.nist.gov/chips/workforce-development (accessed Oct. 10, 2023).

11. M. Kindling, "Building the workforce to power the Semiconductor Revolution," SEMI, https://semi.org/en/blogs/semi-news/building-the-workforce-to-power-the-semiconductor-revolution (accessed Oct. 10, 2023).

12. E. J. Felsberg, "Nist reveals government strategy to support U.S. Semiconductor Research and Development," Jackson Lewis, https://www.jacksonlewis.com/insights/nist-reveals-government-strategy-support-us-semiconductor-research-and-development (accessed Oct. 10, 2023).

Chapter 9: Conclusions and Recommendations to NSF

9.1. Improving Awareness of Sustainability Challenges

There is a need for the NSF to emphasize sustainability challenges in computing by organizing and coordinating federal and state-level awareness campaigns. Specifically, the importance of sustainable computing towards meeting increasingly pressing climate goals and to reduce carbon and greenhouse emissions and water use must be highlighted. There are currently many outstanding challenges and open problems at the level of applications, systems, computer architectures, systems-on-chip/integrated circuits, devices/materials, and workforce development, as outlined in Chapters 3-8. US competitors (China and the European Union) are investing heavily in the domains of foundational sustainability technologies and the micro and macro computing scales. To ensure economic competitiveness, technological leadership, and national security, the US must invest heavily in this area. To ensure the investment needed, NSF must lead efforts to motivate sustainability in research and innovation via joint inter-agency programs, media outreach, and workforce development initiatives. The NSF's Directorate for Computer and Information Science and Engineering (CISE) is uniquely positioned to lead these efforts, specifically across its Division of Computer and Network Systems (CNS) and Division of Computing and Communication Foundations (CCF).

9.2. Data and Infrastructure for Sustainability Initiatives

To further the goals of sustainable computing research, there is a need for facilitating access to sustainability data for 1) operational and 2) embodied overheads of computing. Specifically, data from large cloud computing providers (e.g., Amazon, Google, Microsoft, Meta), embedded, mobile, and IoT solution providers (e.g., Apple, Samsung, Motorola, Nokia), and hardware chip designers (e.g., Intel, AMD, Nvidia) would cover the vast ecosystem of computing that matters for sustainability. The data from these enterprises would relate to manufacturing-related and end-of-life-related metrics such as carbon emissions, water use, and wastewater generation, as well as (in cases where it is possible) operational statistics along the same metrics. Access to such data is critical to ensure meaningful research outcomes as a direct outcome. Indirectly, this access to relevant data will ensure a robust workforce pipeline by attracting students to work in these fields, the competitiveness of U.S. graduates for employment in leading companies, and the competitiveness of U.S. research and innovation in sustainable computing. NSF should also consider partnering with companies, including semiconductor fabrication and cloud hosts, with agreements to obtain operational and embodied sustainability data. It may be possible to engage research facilities such as CEA LETI and IMEC to obtain such data in the short term. In the long-term, NSF can consider helping to establish and support facilities for semiconductor manufacturing, system assembly, and cloud computing (at a reasonable scale) in collaboration with other programs such as nanosystems (whose report from the NSF Workshop on Micro/Nano Circuits and Systems outlined the need for NSF to establish/support facilities to fabricate new types of nanosystems). NSF can also help create a consortium that brings together industry, academia, and government agencies to collaborate on research, development, and knowledge sharing in sustainable computing. This consortium can provide funding, resources, and a platform for researchers and industry experts to work together on specific sustainability-focused projects.

9.3. Key Sustainable Computing Research Thrusts

9.3.1. Applications

Applications are the drivers of computing at all scales, and as such, they have a significant impact on computing sustainability. NSF should initiate new research programs focusing on:

- New green software design methodologies
- New tools for sustainability-aware application design
- New methods for resource management of applications that can balance sustainability with traditional goals, such as reliability and real-time performance
- New approaches to educate developers and practitioners on the challenges and benefits of designing sustainable application

We have provided justifications for these recommendations in Chapter 4.

9.3.2. Systems

At the system level, holistic sustainability initiatives can have a meaningful impact across both minor (e.g., embedded and IoT) and large (e.g., cloud data center) scales. NSF should initiate new research programs focusing on:

- New system design approaches that rely on disaggregation and modular components, which decouple the lifecycles of different components used in the system from each other
- New approaches for efficient repair of computing systems to improve lifetimes
- New analyses on the sustainability of ASIC, ASIP, and CPU/GPU platforms
- New techniques for standardization of interfaces to support efficient repurposing
- New data center designs that can efficiently adapt to energy grid/supply variations

We have provided justifications for these recommendations in Chapter 5.

9.3.3. Sustainable Computer Architectures

Computer architectures and their design methodologies have played an essential role in making the processing of computing applications faster and highly energy-efficient, which has made computing more accessible and integral to our daily tasks. Their role becomes more critical in realizing a future where computing can be sustainable. NSF should initiate new research programs focusing on:

- New efforts on lifecycle analysis of conventional and emerging computer architectures
- New techniques for the design of sustainable computer architectures
- New sustainability-aware design exploration and automation frameworks for general-purpose and domain-specific processors
- New metrics for evaluating the sustainability of computer architectures
- New approaches to educate end users and consumers on the sustainability implications of computer architecture design choices

- New ways to incentivize sustainability considerations for computer architects and the computing industry as a whole

We have provided justifications for these recommendations in Chapter 6.

9.3.4. Systems-on-Chips and Integrated Circuits

Designing SoCs/ICs with sustainability as a goal involves considering various factors throughout the design process, including power optimization, energy-efficient architectures, system-level optimization, materials selection and end-of-life considerations. NSF should initiate new research programs focusing on:

- New techniques for the creation of energy-efficient and long-lasting SOC/IC architectures that meet the computational needs of future applications
- New approaches to realize sustainable manufacturing processes for SOCs/ICs
- New design automation tools for sustainability
- New open-source SOC/IC design with sustainable design practices
- New designs and optimization of emerging paradigms such as in-memory computing and 2.5D/3D stacking with sustainability as a first-class design objective
- New methods for improving SOC/IC reliability and lifetime
- New approaches for lifelong testing over extended component lifetimes
- New techniques for reconfigurable designs that enable extended lifetimes
- New methods for efficient repurposing and recycling of SOCs/ICs

We have provided justifications for these recommendations in Chapter 7.

9.3.5. Devices and Materials for Computing

Semiconductor devices are the critical components in integrated circuits (ICs), such as computer processors, microcontrollers, and memory chips (such as NAND flash and DRAM) that are present in everyday electrical and electronic devices. From a materials perspective, several materials, such as silicon (Si), Germanium (Ge), Gallium Nitride (GaN), and various indium compounds, are in wide use today for fabricating IC components and SOCs. NSF should initiate new research programs focusing on:

- New methods for life-cycle analysis of new materials and devices for computing
- New approaches to salvage older CMOS nodes and fab infrastructure
- Cross-cutting research for the discovery or invention of new computing substrates, such as biomaterials for computing, with more significant sustainability potential
- Collaboration and infrastructure proposals for novel materials and devices to enable material scientists and Microelectronics experts to collaborate with the CISE community with a sustainability focus

We have provided justifications for these recommendations in Chapter 8.

9.3.6. Workforce Development, Education, and Curriculum

In order to promote sustainability in computing and to consolidate sustainable practices, it is of paramount importance to develop a workforce that is trained in sustainable practices. With such a workforce available, sustainability will gradually become more normalized and widely accepted. To promote the proliferation of sustainability in computing, NSF should:

- track sustainability efforts pursued by grants, including intellectual merit, outreach, and broader impacts.
- consider incorporating sustainability topics into programs that are not solely focused on sustainability.
- provide sustainability criteria to review panels for consideration in their deliberations.
- collaborate with professional societies (IEEE, ACM, etc.) to develop curricular guidelines related to sustainable computing
- encourage investigators to incorporate sustainability into their outreach plans at all levels: K-12, undergraduate, and graduate.
- collaborate across directorates on the full spectrum of sustainable computing topics, including research, outreach, and education.

We have provided justifications for these recommendations in Chapter 8.

Appendix A: Workshop Series Information

A.1: Organizing Committee

The following are members of the Organizing Committee of the workshop series:

- Amlan Ganguly, Rochester Institute of Technology
- Sudeep Pasricha, Colorado State University
- Massoud Pedram, University of Southern California
- Fabrizio Lombardi, Northeastern University
- Wuchun Feng, Virginia Institute of Technology

Webmaster

- Sayed Ashraf Mamun (Cisco)

Cognizant NSF Program Director

- Alex Jones (NSF)

A.2: Workshop Series Activities

The following is a list of all activities conducted as parts of the workshop series:

- Online Keynote Speeches
    - When Climate Meets Machine Learning: Edge to Cloud ML Energy Efficiency
        - Speaker: Diana Marculescu (UTA)
        - Date: 24 September 2021
        - Time: 1:00 PM- 2:00 PM (US Eastern Time)
    - Information Technology in the Context of Life-Cycle and Sustainability Assessment
        - Speaker: Arpad Horvath (UCB)
        - Date: 22 October 2021
        - Time: 2:00 PM- 3:00 PM (US Eastern Time)
    - Domain-specific Accelerators for Sustainable Machine Learning
        - Speaker: Norman Jouppi (Google)
        - Date: 28 October 2021
        - Time: 2:00 PM- 3:00 PM (US Eastern Time)
    - Carbon Footprint Quantification and Reduction in Cloud Environment
        - Speaker: Tamar Eilam (IBM)
        - Date: 4 November 2021
        - Time: 12:00 PM- 1:00 PM (US Eastern Time)

    - Sustainability and Corporate Responsibility of technology sourcing represent challenges and impact opportunities across the value chain
        - Speaker: Adam Schafer (Intel)
        - Date: 18 November 2021
        - Time: 1:00 PM- 2:00 PM (US Eastern Time)
- Online Workshop Day
    - Agenda
        - Date: 12/12/2022
        - Time: 11:00 am to 4:00 pm (Eastern Time)
        - Timeline:
            - Welcome Remarks – 11 am – 12 pm
            - Introductions – Amlan Ganguly (RIT)
            - Expectations and Vision for the day – Alex Jones (NSF)
            - Keynote talk – Massoud Pedram (USC)
            - Logistics for the day – Amlan Ganguly (RIT)
            - Panels: 12-3 pm
            - Panel Summaries and Discussions on future events: 3-4 pm
                - Moderator - Fabrizio Lombardi (NEU)
            - Open discussions
            - Wrap-up with future plans and timeline – Amlan Ganguly (RIT)
    - Panels

Panels	*Applications*	*System*	*Architectures*	*SoC/IC*	*Devices/ Materials*	*Workforce Development*
Time (**eastern**)	**12-1pm**	**1-2pm**	**12-1pm**	**12-1pm**	**1-2pm**	**2-3pm**
Moderator/ Lead	Yanzhi Wang (NEU)	Tamar Eilam (IBM)	Wuchun Feng (Virginia Tech)	Mircea Stan (U of Virginia)	Sudeep Pasricha (Colorado State U)	Marilyn Wolf (U of Nebraska - Lincoln)
Panelists	Daniel Schien (U of Bristol), Carole-Jean Wu (Meta Platforms), Ziliang Zong (TX State U), Houman Homayoun (UC Davies)	Yiran Chen (Duke), Helen Li (Duke), Arpad Horvath (UC Berkeley), Dakai Zhu (UTSA)	Murali Annavaram (USC), Udit Gupta (Cornell), Baris Taskin (Drexel)	Partha Pande (WSU), Nikhil Dutt (UC Irvine), Azadeh Davoodi (U Wisconsin)	Marc Riedel (U of Minnesota), Callie Babbitt (RIT), Sarma Vrudhula (ASU)	Yanzhi Wang (NEU), Sandeep Gupta (ASU), Sandeep Gupta (USC) Roman Sobolewski (U of Rochester)
Scribe	Sai Manoj PD (GMU)	Ahyoung Lee (Kennesaw)	Shail Dave (ASU)	Biresh Joardar (U of Houston)	Ishan Thakkar (U of Kentucky)	Purab Sutradhar (RIT)

- In-person Working Roundtable Meeting
    - Attended by
        - Amlan Ganguly (RIT)
        - Sudeep Pasricha (CSU)
        - Tamar Eilam (IBM)
        - Marilyn Wolf (UNL)
        - Mircea Stan (UVA)
        - Sai Manoj Pudukotai Dinakarrao (GMU)
        - Shail Dave (ASU)
        - Ishan Thakkar (UKY)
        - Sayed Ashraf Mamun (Cisco)
        - Purab Ranjan Sutradhar (RIT)

- Workshop Series Website
    - https://nsf-suscomp.org/