Artificial Intelligence HW: Research Needs
May 2020

Semiconductor Research Corp. (SRC) Global Research Collaboration (GRC) is soliciting white papers in the Artificial Intelligence Hardware (AI Hardware) research program. The principal goal of this program is to create new highly efficient AI platforms to enable neuro-inspired, cognitive, and learning abilities which will be required to address the vast range of future data types and workloads as intelligence is enabled from edge devices to the cloud.

The AIHW research needs are described in five major categories:

- Architectures for Power Efficient AI Acceleration
- Modeling, Analysis, and Simulation/Emulation of AI Hardware for Early System Exploration
- HW/SW Co-design of AI Compute Systems
- Fairness, Robustness, Privacy, and Explainability of Models and Algorithms for AI Hardware
- Interplay of AI and System Architecture/Microarchitecture Design

Each of these major categories are broken down into several sub-categories which describes the need in more detail. Even so, these are written to be broad in nature in order to not restrict the investigator's approach. There is no priority order for either the major or minor needs that follow. In each category, there may be applications from large systems to small (datacenter and the edge/end node) and investigators should consider this in their submissions. Members are looking for significant innovations, for example, 100X improvement in energy-performance efficiency or other key metrics for systems for emergent AI applications.

The use of appropriate benchmarks and metrics to assess how far the effort advances the state-of-the-art will be a key part of the evaluation process. It is important that performance and efficiency metrics such as "TOPS/W" (tera ops/Watt) and "% utilization" of hardware be qualified as "peak," "sustained," or "average". The primary metrics should include a performance metric, a power efficiency metric, and a mapping efficiency metric. For example, the end-to-end wall-clock execution time for a set of benchmarks, the energy consumed by the hardware on a benchmark set, and the utilization of the hardware resources during the execution. Breakdown of any metrics for training vs. inference helps identify the suitability of the innovation for deployment in different settings such as cloud, edge, mobile, etc. Appropriate metrics should be used to establish the impact of the advances in each setting. For instance, total throughput and throughput per watt might be metrics for datacenter applications while optimal energy usage might be more appropriate for the edge/end node. Accuracy of the results and/or reporting the metrics at iso-accuracy becomes an important factor for understanding the benefits of approximate computing techniques such as reduced precision FP.

In addition to what is mentioned above, some metrics for consideration include

- Inference accuracy
- Inference robustness to antagonistic inputs
- Inference/unit of energy (per uJ/mJ/J/kJ)
- Training/unit of energy
- Throughput: inferences per unit time, training per unit time
- HW cost metric: MACs (or equivalent) required per unit time
- Memory metrics: local/global memory requirements (access time, latency, bandwidth, average per unit time and total energy per inference)

The needs in the AIHW space cover a broad range of applications, including high performance processors for data centers, automotive, industrial, mobile and edge node computing and communication, and healthcare. Investigators are encouraged to link the results of their work with a potential application to help show the relevance of the proposed work.

This needs document is driving the AIHW solicitation. It is issued to universities worldwide, may be addressed by an individual investigator or a research team. Our selection process is divided into two stages. The interested party is requested to submit a brief 1-page white paper. The white paper should clearly identify what can be done in three years, and a successfully selected white paper will result in an invitation to submit a full proposal. These proposals will be further down-selected for research contracts. The number and size of the contracts awarded will be determined by the amount of available funds, and by the number of high-quality proposals.

Investigators who are funded will be expected to publish at top-tier conferences, including but not limited to ISSCC, VLSI, HPC, ISCA, MICRO, HPCA, ESSCIRC, and ESWEEK (CASES, CODESISSS, & EMSOFT).

White Papers for all the categories below will be considered for funding. Investigators are limited to participation in two white papers in this solicitation (either as a PI or Co-PI) and submissions should highlight which category of need is addressed, such as “A2.3”.

CONTRIBUTORS

AMD	Ganesh Dasika
Arm	Jose Joao, Dam Sunwoo, Matthew Mattina
GLOBALFOUNDRIES	Ted Letavic
IBM	Krishnan Kailas, Matt Ziegler
Intel	Michael Kishinevsky, Greg Chen, Rosario Cammarota, Erik Norden
Mentor Graphics	Russell Klein, Duaine Pryor
NXP	Ben Eckermann, Brian Kahne, Adam Fuks
Qualcomm	Ramesh Chauhan
TI	Steven Bartling, Mahesh Mehendale, Jim Wieser, Clive Bittlestone
SRC	David Yeh

2020 Artificial Intelligence HW Research Needs

A1 Architectures for Power Efficient AI Acceleration

Accelerating future AI systems may benefit from architectures, circuits, and/or devices beyond today’s conventional computing approaches. New architectures or extensions of existing approaches that depart from the deep learning neural network paradigm may provide significant performance and/or power improvements for certain applications. Novel circuits and/or devices may also unlock capabilities unattainable from conventional circuit design and CMOS technology. At the datacenter system level, the challenge of integrating multiple chips or approaches to achieve the equivalent of multi-chip systems are of high importance for the future of AI computing.

General challenges include but are not limited to: Energy-efficient end-to-end system architectures and partitioning (cloud to sensor) and optimizing energy/bandwidth/latency tradeoffs at all levels within the computational hierarchy (data center, gateway, and edge/end node).

These challenges are most acute at both ends of the AI computational hierarchy (e.g. Datacenter and Edge AI). Devices on the edge/end nodes are typically heavily resource constrained with stringent cost, performance, power, communication latency, and bandwidth limitations. Also, all edge/end node AI and microcontroller functionality typically resides on a single die and is implemented on older process nodes to gain access to integrated NVM and high-performance analog, creating additional area/power efficiency challenges. Research is needed to optimize the interplay of on-chip sensing, compute, and off-chip communication requirements at the edge/end node.

In the datacenter, high throughput is crucial, but must be balanced by power efficiency. Datacenter computing environments must combine energy efficient processor designs, multi-chip/module communication for data movement and memory access, and the flexibility/programmability to support diverse workloads. Center of cloud AI computation is highly data access limited (bandwidth, latency, storage), data movement limited (I/O bandwidth, power), and often thermally bounded. Extensions to existing approaches as well as novel architectures such as AI compute-in-memory that address fundamental limitations are of interest.

A1.1	New AI architectures, including but not limited to those using emerging devices and circuits, e.g., reduced precision/dynamic range computation, in-memory and near-memory computing based on charge-based and resistance-based memory devices, other NVM devices, mixed signal techniques, compute-in-DRAM, compute-in-cache, etc.
A1.2	System-level integration solutions for emerging architectures, e.g., SoC, 3D, packaging, inter-chip / module communication, partitioning, etc.
A1.3	Neuromorphic computing: algorithms and hardware for biologically plausible neuron models and learning rules, such as spiking neural networks, spike timing dependent plasticity, and bio-plausible deep learning
A1.4	Use of approximate computing (beyond relaxed precision) for AI/Machine Learning architectures
A1.5	High/Hyper-dimensional computing
A1.6	AI architectures using quantum computing

A1.7	Resource efficient training and inference at the edge: self teaching/adaptation/optimization of initial algorithms to local application conditions/needs within the strict computational/memory/power/costs constraints imposed by edge hardware/software
A1.8	End-to-end optimization schemes that span system-algorithm-architecture-circuit-technology stacks for minimizing energy per decision without compromising accuracy, throughput and cost (power, area, performance), security/privacy constraints for AI systems consisting of sensors, pre- and post-processors, communication networks, and AI computer hardware
A2	**Modeling, Analysis, and Simulation/Emulation of AI Hardware for Early System Exploration**
End-to-end performance and energy efficiency of AI systems are determined by various components including memory subsystem, I/O, on-chip and off-chip network, in addition to core AI computation. Challenges include, but are not limited to, characterizing and modelling long running AI computations that often take days/weeks to complete. Novel methods for modelling, simulation and emulation are essential for early design-space exploration of next generation AI systems. Finally, a better understanding of the theoretical behavior and limits of AI to better guide a design of AI systems is needed.	
A2.1	AI workload analysis and characterization
A2.2	Efficient techniques for end-to-end performance/power/reliability modelling (cycle-accurate and analytical), simulation, emulation, and prototyping for exploration of AI systems
A2.3	Benchmarks for emerging AI applications, and metrics for comparing AI systems
A2.4	Application-level understanding and profiling of new AI applications including (a) recent deep learning networks (e.g. graph convolutional networks, energy-based models) (b) techniques for machine reasoning and (c) neuro-symbolic approaches
A2.5	Modeling infrastructure and techniques for AI computation at the edge/end node, including sensors
A2.6	Analysis and comparison of theoretical limits of algorithms and compute efficiency of AI systems (e.g. understanding theoretical limits of precision, sparsity, and compression)
A3	**HW/SW Co-design of AI Compute Systems**
Interactions and dependencies between hardware and software are integral for achieving high performance on AI workloads. These two fields of study cannot be decoupled. Topics of interest include compilers that map deep learning models to CPU, GPU, and accelerator hardware with reduced data movement, training algorithms (e.g. NAS) that are hardware-cognizant in their optimizations and enabling traditionally non-AI Applications with AI.	
A3.1	Compilers and run-time management that map AI algorithms/computations to homogeneous or heterogeneous compute platforms including CPU/GPU/hardware accelerators
A3.2	Compilers and run-time management that optimize data storage in compute in/near memory for reduced data movement
A3.3	Run-time management of large number of accelerators including virtualization and security of AI computation
A3.4	Co-design of AI exploration, sensing, and training at the edge/end node
A3.5	Automated labeling of data sets for self-supervised learning
A3.6	Co-design of AI and HPC and other scientific applications, e.g. AI-based surrogate models
A3.7	Co-design of CPU-friendly neural network training algorithms

<table>
<tr><th>A4</th><th>Fairness , Robustness, Privacy, and Explainability of Models and Algorithms for AI Hardware</th></tr>
<tr><td colspan="2">Machine Learning has made enormous strides in recent years in its ability to train models and infer results with higher degrees of accuracy than many other types of algorithms. However, one of the potential stumbling blocks for machine learning adoption in many applications is the issue of fairness, robustness, privacy, and explainability. Many machine learning algorithms are somewhat of a “black box”, with no easy way to determine why the algorithm produced the specific output. Explainability is key to challenge an AI/ML-based decision, especially in safety-critical applications from a SOTIF (Safety of The Intended Functionality) perspective. This may be required, for example, to understand whether a correct decision was made in scenarios such as why a loan application was rejected by an AI/ML-based application, or why an autonomous vehicle in an accident decided to drive the route it did. Another important vector is achieving privacy in AI hardware architectures.</td></tr>
<tr><td>A4.1</td><td>Methods and architectures that return a result and a rationale for that result, or that add explainability to existing AI/ML-based solutions</td></tr>
<tr><td>A4.2</td><td>Architectures and algorithms to add fairness into machine learning algorithms and architectures while maintaining best possible performance and accuracy, even when trained with biased data</td></tr>
<tr><td>A4.3</td><td>Architectures robust against both natural variations of input data and adversarial attacks to ensure stability of machine learning and AI decisions. Also, included under this are architectures capable of uncovering corruptioñ/bias of training phase data and model integrity</td></tr>
<tr><td>A4.4</td><td>Enhancing robustness by building prior knowledge about the task to be learned and/or about the training data into the ML solution, e.g. training with a potentially limited set of input data supplemented by rules-based data, and/or pre-wiring the neural network, and/or data synthesis to enlarge training data sets</td></tr>
<tr><td>A4.5</td><td>Architectures with the ability to assess the functionality of its AI/ML process, so that a system with functional safety requirements can identify a malfunction and establish appropriate safety actions</td></tr>
<tr><td>A4.6</td><td>Privacy and confidentiality preserving AI architectures and systems. Included in this are methods for anonymizing and securing training data. (e.g. Homomorphic Deep Learning)</td></tr>
<tr><th>A5</th><th>Interplay of AI and System Architecture/Microarchitecture Design</th></tr>
<tr><td colspan="2">Advances in AI/ML can significantly impact system design in at least two ways. First, AI/ML-based or AI/ML-inspired components can be directly used in hardware designs. For example, branch predictors, prefetchers, and other hardware predictors can be based on ML models or can be optimized using ML models; scheduling and resource management at the core, chip, node and data center levels can be based on ML and improve over heuristic-based approaches. Second, AI/ML can be part of the system design process itself, e.g., providing optimizations at the system, architecture and micro-architecture levels that improve over traditional hardware design methods and flows.

On the other hand, hardware and systems for AI/ML can benefit from groundbreaking advances in system-level architecture, memory systems and optimizations across multiple levels of the hardware/software stack that can directly impact future AI hardware on different design targets: performance, energy efficiency, security, etc. This interplay of AI and system level design is fundamental for design, construction and management of intelligent self-optimizing systems.</td></tr>
<tr><td>A5.1</td><td>AI-based or AI-inspired components that can be used in hardware designs: e.g., hardware predictors, resource management controllers, etc.</td></tr>
<tr><td>A5.2</td><td>AI methods for optimization of hardware designs at the system, architecture and micro-architecture levels, excluding CAD software optimizations (which are part of the CADT thrust)</td></tr>
<tr><td>A5.3</td><td>AI-based design and optimization of AI accelerators and their integration in bigger systems</td></tr>
<tr><td>A5.4</td><td>Synergistic advances in system design and AI/ML to improve performance, energy-efficiency, reliability/robustness and security</td></tr>
<tr><td>A5.5</td><td>AI-assisted operating system, run-time system, and hardware for thread scheduling, DVFS, power state transitions and other hardware resource management</td></tr>
</table>