Neuromorphic Computing @ UTK

Research website: [http://neuromorphic.eecs.utk.edu/](http://neuromorphic.eecs.utk.edu/)

Fall 2016 Kickoff
Challenges with the Existing Computing Paradigm

• von Neumann / Turing computers “data bottlenecks”.
  – Difficult to process large amounts of data efficiently given computation, memory, and storage are distinct, distant and mismatched.

• Deterioration of Moore’s Law.
  – Single core performance improvements ended in 2004
  – Power density and device feature size limits (manufacturing and atomic-scale/noise) have significantly slowed system scaling.

• Programming High-Performance Computers is HARD.
  – Limited success with increasing computer parallelism.
  – Some problems don’t parallelize
  – Software complexity is a major constraint - due to development costs & lack of tools and skills (parallelization is difficult)

• Data and problem sizes continue to grow.
  – Growth in data volume is increasing exponentially.
In 2015, Global Data Volume

Note: 1000 Exabytes = 1 Zetabyte

By 2015 80% of all data will be uncertain.

By 2015 the number of networked devices will be double the entire global population. All sensor data has uncertainty.

The total number of social media accounts exceeds the entire global population. This data is highly uncertain in both its expression and content.

Data quality solutions exist for enterprise data like customer, product, and address data, but this is only a fraction of the total enterprise data.

Multiple sources: IDC, Cisco
The Promise of Neuromorphic Computing

Leveraging Neuroscience for Computing Architecture

• Potentially orders of magnitude improvement in performance/watt.  
  – Use of: low-voltage analog/digital circuits, low-power non-volatile devices  
    (e.g. memristors) and “spike” based computing.
• Significant increase in processing and storage capacities.  
  – Changes in information structure and processing accuracy.
• Significant reduction in physical system size for target applications.
• Adaptable and Dynamic: Can adjust to changes in: environments,  
  inputs, thresholds, critical conditions, ... - (dynamic learning)
• “training” and/or “learning” paradigm vs programming paradigm
• Primary target – “complex dynamic real-time” applications  
  – Classification - Anomaly Detection  
  – Controls - Analytics
Our Approach – NIDA & DANNA

- **NIDA** = Neuroscience-Inspired Dynamic Dynamic Architecture
  - 3D structure using two simple elements: neurons & synapses
  - Both element types incorporate memory & dynamics.
  - Placement, number, and connections among neurons are not constrained

- **DANNA** = Dynamic Adaptive Neural Network Array
  - 2D adaptation of NIDA
  - Implementable in FPGAs and VLSI chips.
  - Single programmable element (neurons, synapses, pass-thru) replicated across an NxM array (array size dependent on chip size and devices used).
  - Real-time execution.
  - Mixed-signal implementations (lower power & higher element density).

- Design tools based upon genetic algorithms - “learning” paradigm
  - Neural Network topology and function determined by evolutionary optimization & real-time adaptation
DANNA Application Development Platform (ADP)

- DANNA ADP Functions:
  - Neural Networks created using genetic algorithm
    - Evolutionary Optimization and hardware matched simulator
  - Command Interface - Controlled, Configured, and Monitored
  - External input “fire events” created by host command
DANNA ADP System

- A complete hardware system that contains:
  - A DANNA Array Chip (Xilinx 690T or 2000T)
  - A Communications Interface (Cypress FX3)
  - A Host System – ARM based Single Board Computer
- Neural Network Dev. Env.
  - Uses Evolutionary Optimization to build neural networks
  - Hardware matched simulator for off-line app development.
- Allows researchers to build, test and evaluate DANNA based Neural Networks in various applications
Neuromorphic Computing at Tennessee

We are a group of faculty, post-docs, graduate students and undergraduates researching a new paradigm of computing, inspired by the human brain. Our research encompasses nearly every facet of the area, including current and emergent hardware implementations, theoretical models, programming techniques and applications. ([http://neuromorphic.eecs.utk.edu/pages/research-overview](http://neuromorphic.eecs.utk.edu/pages/research-overview))