Loihi – a brief introduction

Mike Davies
Director, Neuromorphic Computing Lab | Intel Labs
Loihi at a Glance

Key Properties

- 128 neuromorphic cores supporting up to 128k neurons and 128M synapses with an advanced SNN feature set.

- **Scalable on-chip learning** capabilities to support a range of learning paradigms (unsupervised, supervised, reinforcement-based, and others)

- Supports **highly complex neural network topologies** (up to 2000-way fan-out between neurons)

- Fully digital **asynchronous** implementation

- Fabricated in Intel's **14nm FinFET process** technology
**Chip Architecture**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology</td>
<td>14nm</td>
</tr>
<tr>
<td>Die Area</td>
<td>64 mm²</td>
</tr>
<tr>
<td>Core area</td>
<td>0.41 mm²</td>
</tr>
<tr>
<td>NmC cores</td>
<td>128 cores</td>
</tr>
<tr>
<td>x86 cores</td>
<td>3 LMT cores</td>
</tr>
<tr>
<td>Max # neurons</td>
<td>128K neurons</td>
</tr>
<tr>
<td>Max # synapses</td>
<td>128M synapses</td>
</tr>
<tr>
<td>Transistors</td>
<td>2.07 billion</td>
</tr>
</tbody>
</table>

**Low-overhead NoC fabric**
- 8x16-core 2D mesh
- Scalable to 1000's cores
- Dimension order routed
- Two physical fabrics
- 8 GB/s per hop

**Neuromorphic core**
- LIF neuron model
- Programmable learning
- 128 KB synaptic memory
- Up to 1,024 neurons
- Asynchronous design

**Parallel off-chip interfaces**
- Two-phase asynchronous
- Single-ended signaling
- 100-200 MB/s BW

**Embedded x86 processors**
- Efficient spike-based communication with neuromorphic cores
- Data encoding/decoding
- Network configuration
- Synchronous design
Mesh Operation

Time step $T$ begins.
Cores update dynamic neuron state and evaluate firing thresholds

Above-threshold neurons send spike messages to fanout cores
(Two neuron firings shown.)

All neurons that fire in time $T$ route their spike messages to all destination cores.

**Barrier Synchronization**
Messages exchanged between all cores.
When complete, time advances to time step $T+1$. 

**N-bound Messages**
**S-bound Messages**

---

1 2 3 4 5 6 7 8 9 10

---
Neuromorphic Core Architecture

Discrete time LIF neuron model (CUBA)

Multi-compartment dendritic trees up to 1K compartments

Intrinsic excitability homeostasis

Shared output routing table
4K axon routes

Axon delays
Refractory delays (+ random)

Filtered spike train traces

Graded “reward spikes”

Flexible synaptic plasticity with microcode-programmable rules

Flexible 3-tuple synaptic variables
(1-9b weight, 0-6b delay, 0-8b tag)

Synaptic delays

Synaptic eligibility traces

Synaptic mapping representations

Sparse, dense, and hierarchical

All synaptic connections pooled
128KB shared memory

Sum-of-products rule semantics

Plasticity rules target any synaptic variable

Synaptic delays

Synaptic eligibility traces

Flexible 3-tuple synaptic variables (1-9b weight, 0-6b delay, 0-8b tag)

Graded “reward spikes”

Flexible synaptic plasticity with microcode-programmable rules

Sum-of-products rule semantics

Plasticity rules target any synaptic variable

Filtered spike train traces

Shared output routing table
4K axon routes

Axon delays
Refractory delays (+ random)
Trace-Based Programmable Learning Rules

Short time scale trace correlations => STDP regime

Trace: Exponentially filtered spike train

Presynaptic spike 'X' traces

Postsynaptic spike 'Y' traces

Weight, Delay, and Tag learning rules programmed as sum-of-product equations

\[ w' = w + \sum_{i=1}^{N_P} \prod_{j=1}^{n_i} (V_{i,j} + C_{i,j}) \]

Synaptic Variables
Wgt, Delay, Tag (variable precision)

Variable Dependencies
\[ X_0, Y_0, X_1, Y_1, X_2, Y_2, \]
Wgt, Delay, Tag, etc.

Traces are low precision (7-9b) and may decay stochastically for implementation efficiency

Long time scale traces respond to correlations in activity rates

X_1(t) \( \tau = 20 \)

Y_1(t) \( \tau = 20 \)

X_2(t) \( \tau = 200 \)

Y_2(t) \( \tau = 200 \)
Physical Implementation

Bundled Data Asynchronous Implementation
- Event-driven with integrated flow control
- Fully automated design flow from CSP
- Supports FPGA emulation
- Integrates with synchronous x86 CPUs

One asynchronous controller’s associated pipeline logic
Application Results

**Supervised Learning on Loihi**
- Spiking neural network rapidly learns to recognize labeled real-world objects (Our "Hello World" network)
  - Efficient supervised learning is achieved with a spiking neural network using spike-timing dependent plasticity (STDP) rules.
  - Heterogeneous compute platform: Demonstration on fast, pre-processing of foveated embbeded video data and training & inference on Loihi’s neuromorphic cores.
  - Input images are converted to spatiotemporal spike patterns encoded by neuron spikes.

**Efficient Sparse Coding on Loihi**
- Solve LASSO optimization with orders of magnitude lower energy and runtime
  - Solving LASSO optimization is foundational for many sparse coding problems in ML/DA, signal processing, statistics, etc.
  - Loihi solves LASSO within 1% of optimal solution with significantly lower energy and shorter runtime than classical solvers on FPGA (great for real-time applications).
  - Scaling advantage: Sparsity in energy/time grows with problem size due to integrated compute & memory and event-based communication.

**Parallel Path Search with Propagating Spikes and STDP**
- A Loihi network efficiently computes shortest paths in arbitrary graph topologies
  - Neurons represent nodes of the graph and synapses represent their interconnections.
  - Target node(s) is stimulated resulting in a propagating chain of spikes.
  - STDP acts on the propagating spikes to encode paths to the target(s) in the synaptic weights of the network. Paths are decoded by reading out the synaptic weights or by subsequent stimulation of the network.
  - The solution generalizes to arbitrary graph topologies.
  - Example Path Computations

---

**Reinforcement learning with spiking neurons**
- Use Loihi to perform Q-learning on a 4-armed bandit problem.
- The agent learns to choose the arm with the highest reward.

**Neuromorphic keyword detection**
- Demonstration of a neuromorphic system for keyword detection.
- The system processes audio data in real-time, recognizing keywords associated with specific sounds.

**Adaptive control of a robot arm with Loihi**
- Use Loihi to control a robotic arm, demonstrating adaptive control algorithms.
- The system learns to adjust the arm's movements based on feedback from sensors.

---

**Intel Neuromorphic Computing Lab**
- Intel’s ongoing research in neuromorphic computing.
- Collaboration with academia and industry for advancing neuromorphic technologies.
More to come – Thursday morning
Also see our posters and demos for more

M. Davies et al., “Loihi: A Neuromorphic Manycore Processor with On-Chip Learning,”
in *IEEE Micro*, vol. 38, no. 1, pp. 82-99, January/February 2018
LEGAL INFORMATION

This presentation contains the general insights and opinions of Intel Corporation ("Intel"). The information in this presentation is provided for information only and is not to be relied upon for any other purpose than educational. Statements in this document that refer to Intel's plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel's results and plans is included in Intel's SEC filings, including the annual report on Form 10-K.

Any forecasts of goods and services needed for Intel's operations are provided for discussion purposes only. Intel will have no liability to make any purchase in connection with forecasts published in this document. Intel accepts no duty to update this presentation based on more current information. Intel is not liable for any damages, direct or indirect, consequential or otherwise, that may arise, directly or indirectly, from the use or misuse of the information in this presentation. Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer.

Copyright © 2018 Intel Corporation.
Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.
*Other names and brands may be claimed as the property of others