Hot Chips 2019

Eitan Medina

Aug 2019
Outline

• Training and Inference requirement differences
• Goya Inference Processor Architecture
• Gaudi Training Processor Architecture
• Gaudi Scale Out Solution
# Training & Inference Architecture Requirements

Performance
Power Efficiency
Programmability
Cost

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Training</th>
<th>Inference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance Metric</td>
<td>Time (Throughput)</td>
<td>Throughput, Latency</td>
</tr>
<tr>
<td>Memory Capacity</td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td>Scale-out</td>
<td>Aggressive (100s)</td>
<td>No/Moderate (1s)</td>
</tr>
<tr>
<td>Data Types</td>
<td>FP</td>
<td>Integer + FP</td>
</tr>
</tbody>
</table>
What We Offer

Purpose-Built for AI **Inference**

**GOYA™**
Available

Purpose-Built for AI **Training**

**GAUDI™**
Sampling
Goya Processor Architecture

- Heterogenous compute architecture
  - 3 Engines: TPC, GEMM and DMA
  - Work concurrently using a shared SRAM
- Tensor Processor Core (TPC™)
  - VLIW SIMD vector core
  - C-programmable
- GEMM operations engine
- Tensor addressing
- Robust to any address stride
- Latency hiding capabilities
- PCIe Gen4.0 x16
- 2 DDR4 channels @ 2.667 GT/s, 40GB/s BW, 16GB capacity
- Dedicated HW and TPC ISA for special functions acceleration (e.g. Sigmoid/GeLU, Tanh)
- Mixed-precision data types: FP32, INT32, INT16, INT8, UINT32, UINT16, UINT8
• Mixed-Precision architecture
• Accuracy-loss tolerance:
  • Controlled by user through our software API in compile time
• ResNet-50 example:
  • Int-8: negligible accuracy loss (0.4%)
  • Int-16: no accuracy loss at all (but would reduce throughput)
  • Model was quantized without fine-tuning or retraining

<table>
<thead>
<tr>
<th></th>
<th>GPU Reference FP32</th>
<th>HL-1000 Result INT8</th>
<th>Diff INT8</th>
<th>HL-1000 Result INT16</th>
<th>Diff INT16</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResNet-50 Accuracy*</td>
<td>75.7%</td>
<td>75.3%</td>
<td>-0.4%</td>
<td>75.7%</td>
<td>0.0%</td>
</tr>
</tbody>
</table>

*Top1 accuracy, higher is better
Habana Labs Software Structure & Tools

Goya supports models trained on any processor (CPU, GPU, TPU, Gaudi etc.)
ResNet-50 Inference Performance

ResNet-50 inference throughput and latency performance

<table>
<thead>
<tr>
<th></th>
<th>Images / sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>1,255</td>
</tr>
<tr>
<td>GPU T4</td>
<td>4,944</td>
</tr>
<tr>
<td>Al Processor (AIP) GOYA</td>
<td>15,393</td>
</tr>
</tbody>
</table>

Latency:
- CPU: Not reported
- GPU T4: 26ms
- Al Processor (AIP) GOYA: 1.01ms

NLP: BERT Inference Performance

- State of the art Natural Language Understanding model
- BERT & Goya Architecture:
  - All BERT operators - natively supported
  - GEMM & TPCs - fully utilized
  - HW accelerated non-linear functions
- A mixed precision implementation
  - GEMM operations in int16
  - Some operators like Layer-Normalization in FP32
  - Providing excellent accuracy - At most 0.11% loss vs. trained model in FP32
    - Verified on SQuAD 1.1 and MRPC tasks
- Software-managed SRAM – optimizing data movement between memory hierarchies while executing
**Task** - Question answering, determining if one sentence is the answer to a second sentence.

**Dataset**: SQuAD

**Topology**: BASE; Layers=12; Hidden Size=768; Heads=12; Intermediate Size=3,072; Max Seq Len = 128

---

**BERT LANGUAGE MODEL PERFORMANCE**

<table>
<thead>
<tr>
<th>BATCH = 12</th>
<th>GOYA</th>
<th>T4</th>
<th>Sentence-per-second throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latency: 9.4 ms</td>
<td>1,273</td>
<td>736</td>
<td></td>
</tr>
<tr>
<td>Latency: 16.3 ms</td>
<td>739</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latency: 15.7 ms</td>
<td>1,527</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latency: 32.4 ms</td>
<td>739</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Goya Configuration**:
- Hardware: Goya HL-100; CPU Xeon Gold 6152@2.10GHZ
- Software: Ubuntu v-16.04.4; SynapseAI v-0.2.0–1173

**GPU Configuration**:
- Hardware: T4; CPU Xeon Gold 6154@3Ghz/16GB/4 VMs
- Software: Ubuntu-18.04.2.x86_64-gnu; CUDA Ver 10.1, cudnn7.5; TensorRT-5.1.5.0;
GAUDI™
AI Training Processor
### Key Goals for Gaudi Training Platform

- **Performance @ scale**
  - High throughput at low batch size
  - High power efficiency

- **Enable native Ethernet Scale-out**
  - Avoid proprietary interfaces
  - On-chip RDMA over Converged Ethernet (RoCE v2)
  - Reduced system complexity, cost and power
  - Leverage wide availability of standard Ethernet switches

- **Promote standard form factors**
  - Open Compute Project (OCP) Accelerator Module (OAM)

- **SW infrastructure and tools**
  - Frameworks and ML compilers support
  - Rich TPC kernel library and user-friendly dev tools to enable optimization/customization
Gaudi Processor Architecture

- Heterogenous compute architecture
  - TPC, GEMM & DMA using a shared SRAM
- VLIW SIMD TPC 2.0 Core (C-programmable)
- GEMM operations engine
- Tensor addressing
- Robust to any address stride
- Latency hiding capabilities
- PCIe Gen4.0 x16
- 4 HBMs: 2GT/s, 32 GB capacity, BW 1 TB/sec
- 10 ports of 100Gb Ethernet, or 20x50 GbE
  - With integrated RDMA over Converged Ethernet (RoCE v2)
- Dedicated HW and TPC ISA for special functions acceleration (e.g. Sigmoid, GeLU, Tanh)
- Mixed-precision data types: FP32, BF16, INT32, INT16, INT8, UINT32, UINT16 and UINT8
Software Infrastructure and Tools

- **Graph Compiler**
- **Run-time**
- **Kernel Mode Driver**

**TPC Tools**
- Compiler
- Assembler
- IDE: Debugger / Simulator

**Rich Performance Library**
- Deep learning operators

**On-board processor Software**
- Debugger (Lauterbach)

**SynapseAI**
- TensorFlow
- Python front end
- Compilation Flows
- Topologies

**PCI Driver**
- Multi device support
- Maintenance features

**Performance Profiling Tool**
- Performance Analysis
- Graphical views
- Real time
The Only AI Processor Integrating RoCE RDMA

10 x 100GbE
“This is the problem of distributed computing... by adding more and more servers ROI started to decline and the reason for that is you’re spending too much on communicating... that’s why networking bandwidth is so important”

Nvidia CEO, Jensen Huang, GTC 2019 Keynote
Over standard Ethernet-based RoCE v2 standard format
....connecting to Standard Ethernet Switches
### Gaudi Scale-Out

- Integrated Compute + Networking
- Parameters, Tensors and sub-tensors transfer over Ethernet
- Advanced Congestion controls
- Supporting Lossless and Lossy fabrics

<table>
<thead>
<tr>
<th>Feature</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port configuration</td>
<td>10 x 100 Gbps, 20 x 50 Gbps – IEEE802.3cd</td>
</tr>
<tr>
<td>Low latency</td>
<td>~ 300ns round trip (back to back connection)</td>
</tr>
<tr>
<td>PFC (IEEE 802.3bb)</td>
<td>4 priorities, enables a lossless fabric (lossy fabric is also supported)</td>
</tr>
<tr>
<td>QoS DCBX/ETS (IEEE 802.1az)</td>
<td>Prevents head-of-the-line blocking in DNN graph distribution over the network</td>
</tr>
<tr>
<td>Jumbo frames</td>
<td>8 KB Payloads</td>
</tr>
<tr>
<td>Congestion control</td>
<td>ECN/DCTCP/TCP CUBIC</td>
</tr>
<tr>
<td>Congestion a avoidance</td>
<td>Rate limiter per flow (QP)</td>
</tr>
<tr>
<td>VLAN tagging and priority</td>
<td>IEEE 802.3q/802.1p</td>
</tr>
<tr>
<td>Standard Eth NIC</td>
<td>Support for standard TCP/IP networking over Gaudi Eth ports</td>
</tr>
</tbody>
</table>
**Gaudi Mezzanine card & System**

**HL-205: Mezzanine Card**

- **Processor Technology**: Gaudi HL-2000
- **Host Interface**: PCIe Gen 4.0 X 16
- **Memory**: 32GB HBM2
- **Memory Bandwidth**: 1TB/s
- **ECC Protected**: Yes
- **Max Power Consumption**: 300W
- **Interconnect**: 2Tbps: 20 56Gbps PAM4 Tx/Rx Serdes (RoCE RDMA 10x100GbE or 20 x 50GbE/25GbE)
- **Form Factor and SKUs**: HL-205: OCP Accelerator Module 0.9 spec compliant.

**HLS-1: 8 Gaudi System**

- **AI Processors**: 8X Gaudi (8x HL-205)
- **Host Interface**: 4 ports of x16 PCIe Gen 4.0
- **Memory**: 256GB HBM2
- **Memory Bandwidth**: 8TB/s
- **ECC Protected**: Yes
- **Max Power Consumption**: 3 kW
- **Interconnect**: 24 X 100Gbps RoCE v2 RDMA Ethernet ports (6 x QSFP-DD)
- **Scale-out Interface**: 20 x 100GbE or 20 x 50GbE/25GbE
- **System Dimensions**: 19", 3U height
- **Operating Temp**: 5C to 35C [41F to 95F]
Ultimate Flexibility

For your application—
- Choose the ratio of CPUs to Gaudis
- # of Gaudis per rack
- Your rack power limit
- Your Cluster size

Buy HLS-1 OR design your own
HLS-1 vs. DGX

**DGX**
- NVLink: GPU with proprietary interfaces
- Blocking internal interconnect
- Using Ethernet/IB RDMA NICs
- Management & Scale-out bottleneck over PCIe

**HLS-1**
- Gaudi: on-chip compute + Standard RDMA RoCE
- Non-blocking, all-2-all internal interconnect
- 24 x 100GbE RDMA RoCE for scale-out
- Separate PCIe ports for external Host CPU traffic
Gaudi PCIe Card: Accelerate Existing Servers

**HL-200: PCIe Card**

Fits existing Servers
Gaudi Topologies for scaling Different Training Models

- Hierarchical reduction with full throughput
- Easily scaled up and out

- Huge bandwidth between model-parallel workers
- 80 x100GbE RoCE for every 8 Gaudis
Model Parallel Training → Leapfrog Performance

- DGX-2 Limitation: Scaling GPUs beyond 16 has huge bottleneck
- Goal: Support model parallelism with many workers @ full throughput
- Example: 64 Gaudi system, fully connected with a single networking hop

- 128-Gaudi system (16 systems of 8-Gaudi) is also possible, with 10 switches
Throughput

Accelerate Training
Boost Productivity
Save Energy

Designed to Scale

Unlimited Scale
Standards Based
No Proprietary Lock-in
Thank You

See the Gaudi & Goya Demos outside!