The Rise of Open Programming Frameworks

JC BARATAULT
IWOCL May 2015
1,000+ OpenCL projects

SourceForge
GitHub
Google Code
BitBucket
Interactive GPU Navier-Stokes fluid simulation implemented in OpenCL

TUM.3D Virtual Wind Tunnel

- 10K C++ lines of code, 30 GPU kernels
- CUDA 5.0 to OpenCL 1.2 port in less than a day
- 30 fps with one FirePro S9150
- Multi-GPU & Linux version in June

1 million fluid cells in a 256x64x64 grid
Explore programming methodologies for the next generation hardware to achieve performance portability in current, emerging, and tomorrow’s computational resources.

N-body performance on wide range of architectures
## DNNs Everywhere

<table>
<thead>
<tr>
<th>Supercomputers</th>
<th>Datacenters</th>
<th>Tablets, smartphones</th>
<th>Wearable devices</th>
</tr>
</thead>
</table>

- 1000s GPUs
- 100k-1m servers
- 700m (in China)
- Billions?

Supercomputer used for training

Trained DNNs then deployed to data centers (cloud), smartphones, and even wearables and IoTs
OpenCL-based Open ECO-SYSTEM

- Diverse industry participation, from cell phones to supercomputers
  - Processor vendors, system OEMs, middleware vendors, application developers.
- OpenCL is the industry standard embraced by many companies.
DNN – Anywhere, Anytime

- DNN-based image recognition on mobile device
- No connectivity needed
- Real time, directly works on video stream
- Everything is done within the device
- What you point is what you get

- OpenCL based, highly optimized
- Large deep neural network models
- Thousands of objects, flowers, dogs, and bags etc
- Unleashed the full potential of the device hardware

- World’s first in-place mobile DNN app?
- And the best!
Open source clBLAS

github.com/clMathLibraries/clBLAS

AMD FirePro S9150

• 16GB GDDR5
• 320 GB/s memory bandwidth
• Full OpenCL + OpenGL
• 4 TFlops SGEMM
• 2 Tflops DGEMM

>80% efficiency
#1 GSI L-CSC cluster
600 FirePro S9150
5.27 GFlops/W
AMD clFFT vs Nvidia cuFFT 6.0
on AMD W9100 & S9150 vs Nvidia K40c
1D single precision complex batched FFTs

FirePro W9100 for workstation
FirePro S9150 for server
Advanced Hands On OpenCL™
Simon McIntosh-Smith
James Price
Tom Deakin
Mike O'Connor

HP DL380 G9 server

Remote machines
- HP/AMD: ssh user@192.168.2.2
  Connect via FireProS or FirePro24 WiFi network.
REQUIREMENT: Memory and performance

- **Memory Availability**
  - 8TB
  - 64MB
  - 12 GB
  - 16 GB
  - 512GB
  - 1TB

- **dGPU**
  - NVIDIA max per ASIC - Tesla

- **CPU**
  - Intel Haswell 1TF *est

- **3D RTM**

- **dGPU**
  - FirePro S9150
  - AMBER14
  - NAMD
  - FastROC
  - RTM
  - XFdtd

- **Raw performance TF Double**
  - 3TF
  - 2TF
  - 1TF
  - 0
AMD HPC Roadmap Trends

**FirePro GPU**
- S9150 2TF DGEMM

**APU**
- Kaveri 32-bit
- Carrizo 64-bit

**Server CPU**
- Opteron 8/16 x86 cores

- Next Gen GPU
- Next Gen APU
- Next gen CPU
- HPC APU Multi TFlops

- Next Gen

---

AMD | IWOCL 2015
AMD HPC APU delivers memory and performance

- dGPU
- CPU
- HPC APU

<table>
<thead>
<tr>
<th>Raw performance (TF Double)</th>
<th>Memory Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1TF</td>
<td>64MB</td>
</tr>
<tr>
<td>2TF</td>
<td>12 GB</td>
</tr>
<tr>
<td>3TF</td>
<td>16 GB</td>
</tr>
<tr>
<td>3D RTM</td>
<td>512GB</td>
</tr>
<tr>
<td>1TB</td>
<td>1TB</td>
</tr>
<tr>
<td>8TB</td>
<td>SVM based</td>
</tr>
</tbody>
</table>

- FirePro S9150
- FastROC
- RTM
- XFdtd
- NAMD
- Intel Haswell 1TF *est
- Hadoop

NVIDIA max per ASIC - Tesla
CUSTOMER CENTRIC
Smooth Transition to Heterogeneous Computing

Hardware Agnostic Open Programming Frameworks

C/C++  C++AMP  Fortran

OpenCL 2.0  OpenMP 4.0

Python  Java
Develop your code now for tomorrow’s platforms

**Now**
- Best performance with FirePro GPU

**Summer 2015**
- AMD Carrizo APU x86 64-bit laptop for code testing
Shared Virtual Memory on APU vs. PCIe data transfer on dGPU

APU SVM

//CL_MEM_SVM_FINE_GRAIN_BUFFER means host and device can concurrently access the buffer, thus no more data transfer...

float* Buffer = (float*)clSVMAlloc(ctx, CL_MEM_READ_WRITE | CL_MEM_SVM_FINE_GRAIN_BUFFER, 1024 * sizeof(float), 0);

//fill the buffer from host, no data transfer
for (int i = 0; i < 1024; i++)
    Buffer[i] = ...;

// use your SVM buffer in you OpenCL kernel on device directly
clSetKernelArgSVMPointer(my_kernel, 0, Buffer);

clEnqueueNDRangeKernel(queue, my_kernel, ...)

dGPU PCIe

//create device buffer

cl_mem DeviceBuffer = clCreateBuffer(ctx,
    CL_MEM_READ_WRITE, 1024 * sizeof(float), NULL, &err);

//create host buffer
float* hostBuffer = new float[1024];
for (int i = 0; i < 1024; i++)
    hostBuffer[i] = ...;

//data transfer happens here
clEnqueueWriteBuffer(queue, DeviceBuffer, ... , hostBuffer);

//use our device buffer on device
clSetKernelArg(my_kernel, 0, sizeof(cl_mem), &DeviceBuffer);

clEnqueueNDRangeKernel(queue, my_kernel, ...)

AMD | IWOCL 2015
OpenCL 2.0 support in AMD Compute SDK 1.0

OpenCL 2.0 Core features

- Shared Virtual Memory Coarse grain, Buffer mode
- Device-side enqueue (kernels enqueueing kernels, dynamic parallelism, ...)
- C11 atomics
- Generic address space
- Program scoped variables
- Pipes
- Non-uniform workgroups (flexible ND-range)
- sRGB image reads
- Create an Image2D from buffer
- New workgroup built-in functions (all, any, broadcast, reduce, scan)
- Precision for Math built-in native functions

OpenCL 2.0 Optional features **APU only**

- Shared Virtual Memory Fine Grain Buffer Mode
- Platform Atomics

More info in AMD Blog Series

CodeXL helps SW developers get the best performance on AMD platforms

- Debug, Profile and Analyze applications
  - On local and remote hosts

- Power Profiling

- System level “white box” view

- AMD CPUs, GPUs and APUs

- Multiple platforms and Operating Systems
  - Standalone application for Windows® and Linux®
  - Integrated into Microsoft® Visual Studio®

- Free to download and use

Start developing for OpenCL 2.0 **today** with latest AMD APUs and GPUs

- **Hawaii**: 44 GCN CUs
  - 4 TF SGEMM, 2 TF DGEMM GPU
- **Carrizo**: 4 x86 cores, 8 GCN CUs
  - x86 64-bit APU
- **HPC APU**: Multi Tflops
  - Multi Tflops x86 64-bit APU

**Code portability**

- OpenCL
- HSA Foundation
- OpenMP
Thank you
DISCLAIMER & ATTRIBUTION

The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions and typographical errors.

The information contained herein is subject to change and may be rendered inaccurate for many reasons, including but not limited to product and roadmap changes, component and motherboard version changes, new model and/or product releases, product differences between differing manufacturers, software changes, BIOS flashes, firmware upgrades, or the like. AMD assumes no obligation to update or otherwise correct or revise this information. However, AMD reserves the right to revise this information and to make changes from time to time to the content hereof without obligation of AMD to notify any person of such revisions or changes.

AMD MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND ASSUMES NO RESPONSIBILITY FOR ANY INACCURACIES, ERRORS OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION.

AMD SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL AMD BE LIABLE TO ANY PERSON FOR ANY DIRECT, INDIRECT, SPECIAL OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF AMD IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

ATTRIBUTION

© 2015 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD Arrow logo, AMD Opteron, Radeon and combinations thereof are trademarks of Advanced Micro Devices, Inc. in the United States and/or other jurisdictions. Adobe is a registered trademark of Adobe Systems Inc. ARM is a registered trademark of ARM Limited. Linux is a registered trademark of Linus Torvalds. OpenCL is a trademark of Apple Inc. used by permission of Khronos. Windows and Microsoft registered trademarks of Microsoft Corporation. Other names are for informational purposes only and may be trademarks of their respective owners.