HETEROGENEOUS SYSTEM ARCHITECTURE: PLATFORM FOR THE FUTURE

Haibo Xie, Ph.D.

Chief HSA Evangelist
AMD China
OUTLINE:

The Challenges with Computing Today

Introducing Heterogeneous System Architecture (HSA)

Taking HSA to the Industry
A NEW ERA OF PROCESSOR PERFORMANCE

Single-Core Era

- Enabled by:
  - ✓ Moore’s Law
  - ✓ Voltage Scaling
- Constrained by:
  - x Power
  - x Complexity

- Assembly ➔ C/C++ ➔ Java …

Multi-Core Era

- Enabled by:
  - ✓ Moore’s Law
  - ✓ SMP architecture
- Constrained by:
  - x Power
  - x Parallel SW
  - x Scalability

- pthreads ➔ OpenMP / TBB …

Heterogeneous Systems Era

- Enabled by:
  - ✓ Abundant data parallelism
  - ✓ Power efficient GPUs
- Temporarily Constrained by:
  - x Programming models
  - x Comm.overhead

- Shader ➔ CUDA ➔ OpenCL ➔ !!!

Modern Application Performance

- Time (Data-parallel exploitation)

Single-Thread Performance

- Time

Throughput Performance

- Time (# of processors)
WHAT WE ARE FACING – POWER ISSUE

Reducing POWER consumption is increasingly CRITICAL across all segments of computing.
WHAT WE ARE FACING – PERFORMANCE

Demand constantly improving PERFORMANCE to enable compelling new user EXPERIENCES.
WHAT WE ARE FACING – PROGRAMMABILITY

Programmer

PRODUCTIVITY is another essential element that must be delivered.
WHAT WE ARE FACING – PORTABILITY

Developers can NOT SUSTAIN today’s trend of REWRITING code for an ever expanding number of different platforms.
Current CPUs and GPUs have been designed as separate processing elements and do not work together efficiently...
We need consideration of overall system efficiency

Today’s efficiency problems result from the way computers have evolved

Typically platform builders create innovative new hardware and offer an API for software to access it

That tired thinking has only ever had niche success!
MAINSTREAM A-SERIES AMD FUSION APU: “TRINITY”

A-Series APU
- Up to four x86 CPU cores
  - AMD Turbo CORE frequency acceleration
- Array of Radeon™ Cores
  - Fully GPGPU support
- PCIe® Gen3
- Dual-channel DDR3
- 17–35/65–100 watts TDP

Performance:
- Up to 800 Gflops of Single Precision Compute
INTRODUCING HETEROGENEOUS SYSTEM ARCHITECTURE
Brings All the Processors in a System into Unified Coherent Memory
### APU HSA FEATURE ROADMAP

#### Physical Integration
- Integrate CPU & GPU in silicon
- Unified Memory Controller
- Common Manufacturing Technology

#### Optimized Platforms
- GPU Compute C++ support
- User mode scheduling
- Bi-Directional Power Mgmt between CPU and GPU

#### Architectural Integration
- Unified Address Space for CPU and GPU
- GPU uses pageable system memory via CPU pointers
- Fully coherent memory between CPU & GPU

#### System Integration
- GPU compute context switch
- GPU graphics pre-emption
- Quality of Service
- Extend to Discrete GPU
**HSA SOLUTION STACK**

- **Overall Vision:**
  - Make GPU easily accessible
    - Support mainstream languages
    - Expandable to domain specific languages
    - Complete GPU tool-chain
    - Programming & debugging & profiling like CPU does
  - Make compute offload efficient
    - Direct path to GPU (avoid Graphics overhead)
    - Eliminate memory copy
    - Low-latency dispatch
  - Make it ubiquitous
    - Drive HSA as a standard through HSA Foundation
    - Open Source key components
**HSA INTERMEDIATE LAYER - HSAIL**

- HSAIL is a virtual ISA for parallel programs
  - Finalized to ISA by a JIT compiler or “Finalizer”
  - Low level for fast JIT compilation

- Explicitly parallel
  - Designed for data parallel programming

- Support for exceptions, virtual functions, and other high level language features

- Syscall methods
  - GPU code can call directly to system services, IO, printf, etc

- Debugging support
TASK QUEUING RUNTIMES

- Popular pattern for task and data parallel programming on SMP systems today

- Characterized by:
  - A work queue per core
  - Runtime library that divides large loops into tasks and distributes to queues
  - A work stealing runtime that keeps the system balanced

- HSA is designed to extend this pattern to run on heterogeneous systems
FUTURE COMMAND AND DISPATCH FLOW

- Application codes to the hardware
- User mode queuing
- Hardware scheduling
- Low dispatch times

- No APIs
- No Soft Queues
- No User Mode Drivers
- No Kernel Mode Transitions
- No Overhead!
FUTURE COMMAND AND DISPATCH CPU <-> GPU

Application / Runtime

CPU1

CPU2

GPU
OPENCL™ AND HSA

- HSA is an optimized platform architecture for OpenCL™
  - Not an alternative to OpenCL™

- OpenCL™ on HSA will benefit from
  - Avoidance of wasteful copies
  - Low latency dispatch
  - Improved memory model
  - Pointers shared between CPU and GPU

- HSA also exposes a lower level programming interface, for those that want the ultimate in control and performance
  - Optimized libraries may choose the lower level interface
HSA TAKING PLATFORM TO PROGRAMMERS

- Balance between CPU and GPU for performance and power efficiency

- Make GPUs accessible to wider audience of programmers
  - Programming models close to today’s CPU programming models
  - Enabling more advanced language features on GPU
  - Shared virtual memory enables complex pointer-containing data structures (lists, trees, etc) and hence more applications on GPU
  - Kernel can enqueue work to any other device in the system (e.g. GPU->GPU, GPU->CPU)
    - Enabling task-graph style algorithms, Ray-Tracing, etc

- Clearly defined HSA memory model enables effective reasoning for parallel programming

- HSA provides a compatible architecture across a wide range of programming models and HW implementations.
THE HSA OPPORTUNITY ON MODERN APPLICATIONS

**SOLUTION**
- HSA + Libraries = productivity & performance with low power
- Few M HSA coders
- Few 100Ks HSA apps
- Wide range of differentiated experiences

**PROBLEM**
- GPU/HW blocks hard to program
- Not all workloads accelerate
- ~100K GPU coders
- ~200 apps
- Significant niche value

**Developer Return**
(Differentiation in performance, reduced power, features, time to market)

**Developer Investment**
(Effort, time, new skills)

Historically, developers program CPUs
- ~10+M* CPU coders
- ~4M apps
- Good user experiences

~4M apps
~10+M* CPU coders
~200 apps
~100K GPU coders

*IDC
Taking HSA to the Industry

HSA Technology, Scaling to serve the world.
HSA FOUNDATION INITIAL FOUNDERS

Key Founders of the HSA Foundation

- Imagination
- ARM
- Texas Instruments
- Qualcomm
- AMD
- MediaTek
- Samsung
AMD’S OPEN SOURCE COMMITMENT TO HSA

- We will open source our linux execution and compilation stack
  - Jump start the ecosystem
  - Allow a single shared implementation where appropriate
  - Enable university research in all areas

<table>
<thead>
<tr>
<th>Component Name</th>
<th>AMD Specific</th>
<th>Rationale</th>
</tr>
</thead>
<tbody>
<tr>
<td>HSA Bolt Library</td>
<td>No</td>
<td>Enable understanding and debug</td>
</tr>
<tr>
<td>OpenCL HSAIL Code Generator</td>
<td>No</td>
<td>Enable research</td>
</tr>
<tr>
<td>LLVM Contributions</td>
<td>No</td>
<td>Industry and academic collaboration</td>
</tr>
<tr>
<td>HSA Assembler</td>
<td>No</td>
<td>Enable understanding and debug</td>
</tr>
<tr>
<td>HSA Runtime</td>
<td>No</td>
<td>Standardize on a single runtime</td>
</tr>
<tr>
<td>HSA Finalizer</td>
<td>Yes</td>
<td>Enable research and debug</td>
</tr>
<tr>
<td>HSA Kernel Driver</td>
<td>Yes</td>
<td>For inclusion in linux distros</td>
</tr>
</tbody>
</table>
THE FUTURE OF HETEROGENEOUS COMPUTING

- The architectural path for the future is clear
  - Programming patterns established on Symmetric Multi-Processor (SMP) systems migrate to the heterogeneous world
  - An open architecture, with published specifications and an open source execution software stack
  - Heterogeneous cores working together seamlessly in coherent memory
  - Low latency dispatch
  - No software fault lines
THANK YOU!

Access HSA:
http://developer.amd.com
http://hc.csdn.net

Haibo Xie:
haibo.xie@amd.com
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. There is no obligation to update or otherwise correct or revise this information. However, we reserve the right to revise this information and to make changes from time to time to the content hereof without obligation to notify any person of such revisions or changes.

NO REPRESENTATIONS OR WARRANTIES ARE MADE WITH RESPECT TO THE CONTENTS HEREOF AND NO RESPONSIBILITY IS ASSUMED FOR ANY INACCURACIES, ERRORS OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION.

ALL IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE ARE EXPRESSLY DISCLAIMED. IN NO EVENT WILL ANY LIABILITY TO ANY PERSON BE INCURRED FOR ANY DIRECT, INDIRECT, SPECIAL OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

AMD, the AMD arrow logo, and combinations thereof are trademarks of Advanced Micro Devices, Inc. All other names used in this presentation are for informational purposes only and may be trademarks of their respective owners.

© 2012 Advanced Micro Devices, Inc.