AGENDA

- AMD’s Graphics Core Next (GCN) Architecture
- Processors Featuring the GCN Architecture
- IEEE 754 Floating-Point Arithmetic in GCN
- Summary and Future Directions
A NEW GPU DESIGN FOR HETEROGENEOUS COMPUTING

- Leading-edge graphics performance and features
- Optimized for heterogeneous computing
  - Full support for IEEE-754 floating-point
  - Ability to share virtual x86-64 address space with CPU cores
  - Architectural support for traps, exceptions & debugging
- Designed for excellent performance and power efficiency
- Flexible floating-point arithmetic implementations
Graphics Processing Units (GPUs) for Gaming and Compute
Accelerated Processing Units (APUs) that combine GPUs and x86 CPUs
Game consoles: Sony PS4 AND MICROSOFT XBOX ONE

44 Compute Units (CUs)
- 5.6 SP TFLOPS
- 2.8 DP TFLOPS
4 Geometry Processors
- 4 billion primitives/sec
704 32b Load/Store Units
- Up-to 2.8 TB/sec L1 bandwidth
1MB L2 Cache
- Up-to 1 TB/sec L2/L1 bandwidth
6.2 billion transistors

RADEON™ R9 290X FROM 2013
Each Compute Unit (CU) contains 4 SIMD Vector Units, each with:
- A 16-lane integer and floating-point vector ALU
- 64KB Vector General Purpose Register (VGPR) file
- A 48-bit Program Counter
- Instruction buffer for 10 wavefronts
  - A wavefront is a group of 64 threads: the size of one logical VGPR

A 64-thread wavefront issues to a 16-lane SIMD Unit over four cycles
Each 16-lane SIMD Vector Unit
- Supports half, single, and double precision floating-point arithmetic
- Issues one half or single precision instruction per-lane per-clock (including FMA)
- Issues double-precision operations at a reduced rate
VECTOR UNIT FLOATING-POINT ARITHMETIC DATATYPES AND MODES

Floating-point Arithmetic Datatypes
- 16-bit, 32-bit, and 64-bit floating-point numbers as defined by IEEE 754-2008

IEEE Rounding and Sub-normal Handling Modes
- Round to nearest even, Round toward +Infinity, Round toward –Infinity, Round toward zero
- Control for input sub-normal flush to zero and underflow flush to zero
- Provided under software control anywhere in the program
- Separate rounding mode and sub-normal control for single and double precision numbers

Exceptions Support
- Inexact, underflow, overflow, division by zero, sub-normal, and invalid operation
- Provided in hardware with mechanisms for software recording and reporting
VECTOR UNIT FLOATING-POINT ARITHMETIC OPERATIONS

▶ **FMA** (Fused Multiply Add)
  - Single cycle issue instruction
  - IEEE 754-2008 precise with all round modes and proper handling of Nan/Inf/Zero
  - Full sub-normal support in hardware

▶ **MULADD** (Multiply Add)
  - Single cycle issue instruction
  - IEEE MUL followed by IEEE ADD with round and normalization after both multiplication and subsequent addition

▶ **VCMP** (Vector Compare)
  - A full set of vector compare operations designed to fully implement all the IEEE 754-2008 comparison predicates
  - Results written to a vector mask register

▶ **3 Operand Selection Operations**
  - `V_MIN3`, `V_MAX3`, `V_MED3`
VECTOR UNIT FLOATING-POINT ARITHMETIC OPERATIONS

▲ FP Conversion Ops
- Between 16-bit, 32-bit, and 64-bit floating-point values with full IEEE 754-2008 rounding

▲ 64-bit Transcendental Approximations
- Hardware based double precision approximations for reciprocal, reciprocal square root and square root

▲ 16-bit and 32-bit Transcendental Approximations
- Hardware based approximations for reciprocal, reciprocal square root, square root, exponent, logarithm, sine, cosine

▲ Several other FP Arithmetic Operations
- Basic arithmetic operations, min and max, interpolation, rounding and truncation, extract exponent or mantissa, etc.

▲ Graphics Core Next ISA Documentation
- http://amd-dev.wpengine.netdna-cdn.com/wordpress/media/2013/07/AMD_GCN3_Instruction_Set_Architecture.pdf

▲ Open Source GPU Software (compilers, tools, drivers, libraries, applications)
- http://gpuopen.com
SUMMARY AND FUTURE DIRECTIONS

- Support for IEEE 754-2008 floating-point arithmetic is essential for GPU compute
- Wide range of operations required for graphics, multimedia, and scientific computing
- GPU computing can help solve key challenges in science, engineering, medicine, and manufacturing
  - Very high performance and power efficiency

Future directions
- Efficient support for multiple precisions and new operations
- Efficient vector floating-point reductions and fused operations
- Power-efficient floating-point arithmetic for exascale computing
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.