AOCL 3.2 Release Notes

Advanced Micro Devices

© 2022 Advanced Micro Devices, Inc. All rights reserved.

The information contained herein is for informational purposes only, and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions and typographical errors, and AMD is under no obligation to update or otherwise correct this information. Advanced Micro Devices, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of noninfringement, merchantability or fitness for particular purposes, with respect to the operation or use of AMD hardware, software or other products described herein. No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of AMD's products are as set forth in a signed agreement between the parties or in AMD's Standard Terms and Conditions of Sale. Any unauthorized copying, alteration, distribution, transmission, performance, display or other use of this material is prohibited.

Trademarks

AMD, the AMD Arrow logo, AMD AllDay, AMD Virtualization, AMD-V, PowerPlay, Vari-Bright, and combinations thereof are trademarks of Advanced Micro Devices, Inc. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

Dolby is a trademark of Dolby Laboratories.

HDMI is a trademark of HDMI Licensing, LLC.

HyperTransport is a licensed trademark of the HyperTransport Technology Consortium.

Microsoft, Windows, Windows Vista, and DirectX are registered trademarks of Microsoft Corporation in the US and/or other countries.

MMX is a trademark of Intel Corporation.

OpenCL is a trademark of Apple Inc. used by permission by Khronos.

PCIe is a registered trademark of PCI-Special Interest Group (PCI-SIG).

USB Type-C® and USB-C® are registered trademarks of USB Implementers Forum.

Reverse engineering or disassembly is prohibited.

USE OF THIS PRODUCT IN ANY MANNER THAT COMPLIES WITH THE MPEG ACTUAL OR DE FACTO VIDEO AND/OR AUDIO STANDARDS IS EXPRESSLY PROHIBITED WITHOUT ALL NECESSARY LICENSES UNDER APPLICABLE PATENTS. SUCH LICENSES MAY BE ACQUIRED FROM VARIOUS THIRD PARTIES INCLUDING, BUT NOT LIMITED TO, IN THE MPEG PATENT PORTFOLIO, WHICH LICENSE IS AVAILABLE FROM MPEG LA, L.L.C., 6312 S. FIDDLERS GREEN CIRCLE, SUITE 400E, GREENWOOD VILLAGE, COLORADO 80111.

Contents

Release Highlights

AOCL-LibM

- Full C99 functions:
  - Single (F32) and double precision (F64) are supported
  - Legacy long-double format (F80) is not supported
- Fast variants (with reduced precision - ULP up to 4) of the routines `acos`, `asin`, `asinf`, `atan`, `atanf`, `expf`, `log`, `logf`, `powf`, `tan`, and `tanf`
- New complex number variant functions – Most frequently used functions exp/log/pow

AOCL-FFTW

- Upgraded AOCL-FFTW to align with the open source master FFTW 3.3.10 from MIT
- Dynamic dispatcher for AOCL-FFTW
- Microsoft Windows 11 and 12 support added for building wisdom binary, generating separate include folder and libraries for ST, MT, and MPI

AOCL-BLIS

- New features:
  - Extended BLAS function - DZGEMM
  - Progress feature for xGEMM and xTRSM APIs: Time taken to complete the mathematical operations tends to increase exponentially with large input problem sizes; this feature provides users a periodic update on the operation progress.
  - Runtime Threading control using OpenMP APIs
  - Dynamic Dispatch covers APUs
  - Improved detection of standard x86-64 feature support
  - Minor bug fixes
- Performance improvements in the following single-threaded and multi-threaded functions:
  - DGEMM, SGEMM, ZGEMM, and CGEMM
  - DTRSM, DGEMMT, ZTRSM, CTRSM, and DTRMM
  - SGEMV, DHER2, ZTRSV, and DSYMV
  - ?AXPBYV, SSCALV, DSCALV, ?DOTXV, and ZAXPY2V

AOCL-libFLAME

- Improved performance of the following for AMD “Zen” architecture:
  - Eigen Value routines (DSYEVD and DSTEQR)
  - SVD routines (DGESVD)
- Increased coverage of tracing and logging support for libFLAME APIs
- Added AOCL_FLA_PROGRESS that provides progress update on API computations running for a long time; this support is available for double precision LU Factorization
- Improvements in libFLAME build system with new config flag for enabling optimization specific to AMD CPUs
- Improved test coverage

AOCL-ScaLAPACK

- Feature AOCL_ScaLAPACK_PROGRESS that provides progress update on API computations running for a long time; this support is available for double precision LU Factorization
- Support to link with custom BLACS library on Linux

AOCL-Sparse

- API for multiplying two Sparse matrices (aoclsparse_xcsr2m)
- API aoclsparse_xilu_smoother that acts as a preconditioner to compute an update to the iterative solution x of Ax=b
- Improved performance of single thread SPMV routine that supports hint and optimize functions analyzing the sparsity pattern for better optimization
- Multi-thread support for SPMV routine

AOCL-RNG

- AVX2 based kernels for Mersenne Twister RNG API for better performance
- New Base Generator - SIMD based Fast Mersenne Twister (SFMT) that is faster than the native Mersenne Twister

AOCL-Cryptography

- New cryptography library with AMD "Zen" support and optimizations for standard encryption algorithms, including the following:
  - AES-CBC, CFB, GCM, and OFB modes
  - Support for 128/192/256-bit keys
  - SHA-2 hashing functions (224, 256, 384, and 512)
  - Hardware acceleration is supported only on Intel Advanced Encryption Standard New Instructions (AES-NI) enabled platforms
- Libcrypto API support for OpenSSL
- Gtest-based testing infrastructure

AOCL-LibMem

- New library for optimized memory and string functions
- Supports memcpy, mempcpy, memmove, memset, and memcmp functions
- Choice of instructions based on alignment and cacheability
- Tunable threshold parameters for ERMS and Non-Temporal instructions

AOCL Enabled MUMPS Library

- CMake-based build system on Windows for AOCL-enabled MUMPS sparse solver library; for more information, refer to GitHub MUMPS-build (*https://github.com/amd/mumps-build*)
- Spack-based recipe on Linux for AOCL enabled MUMPS sparse solver library
- Enablement of MUMPS 5.4.1