AMD EPYC™ 7xx1 Series Processors

Compiler Options Quick Reference Guide

AOCC compiler (with Flang - Fortran Front-End)

Latest release: 1.2.1, Jul 2018

https://developer.amd.com/amd-aocc/

Architecture	
Generate instructions that run on EPYC/RYZEN	-march=znver1
Generate instructions for the local machine	-march=native
Optimization Levels	
Disable all optimizations	-O0
Minimal level speed and code optimization	-O1
Moderate level optimization (default)	-O2/ -O
Aggressive optimizations	-O3
Maximize performance	-Ofast
Additional Optimizations (For DragonEgg , use –fplugin-arg-dragonegg-llvm-option=”<[option without ‘f ’ prefixed], [option without ‘f’ prefix]…>“ eg: -fplugin-arg-dragonegg-llvm-option=” -inline-aggressive, -function-specialize”**)**	
Enable Link Time Optimization	-flto
Enable unrolling	-funroll-loops
Enable aggressive loop optimizations	-enable-loop-versioning-licm -enable-loop-distribute -enable-partial-unswitch -unroll-aggressive -loop-unswitch-aggressive
Enable aggressive inline optimizations	-finline-aggressive -function-specialize
Enable aggressive vectorization	-vectorize-memory-aggressively -disable-vect-cmp -enable-strided-vectorization -enable-epilog-vectorization
Enable memory layout optimizations	-fstruct-layout -fremap-arrays
Profile Guided optimizations	-fprofile-instr-generate and –fprofile-instr-use
OpenMP	-fopenmp
Other options	
Enable faster, less precise math operations	-ffast-math
Compile free form FORTRAN	-ffree-form
OpenMP threads and affinity (N number of cores)	export OMP_NUM_THREADS=N export GOMP_CPU_AFFINITY=“0-{N-1)”
Link to AMD library	-L/libm-install-dir/lib -lamdlibm

GNU compiler collection (gcc, g++, gfortran)

Latest release: 8.2, Jul 2018

Recommended version : 8.2 (at least 6.3 and above)

http://gcc.gnu.org

Architecture	
Generate instructions that run on EPYC/RYZEN	-march=znver1
Generate instructions for the local machine	-march=native
Optimization Levels	
Disable all optimizations (default)	-O0
Minimal level speed and code optimizations	-O1
Moderate level optimizations	-O2/ -O
Aggressive optimizations	-O3
Maximize performance	-Ofast
Additional Optimizations	
Link time optimization	-flto
Enable unrolling	-funroll-all-loops
Generate memory preload instructions	-fprefetch-loop-arrays --param prefetch-latency=300
Profile-guided optimization	-fprofile-generate and –fprofile-use
OpenMP	-fopenmp
Other options	
Enable generation of code that follows IEEE arithmetic	-mieee-fp
Enable faster, less precise math operations	-ffast-math
Compile free form FORTRAN	-ffree-form
OpenMP threads and affinity (N number of cores)	export OMP_NUM_THREADS=N export GOMP_CPU_AFFINITY=“0-{N-1)”
Link to AMD library	-L/libm-install-dir/lib -lamdlibm

GlibC

Latest release: 2.28, Aug 2018

Recommendation : 2.26 or later

https://www.gnu.org/software/libc/

Binutils

Recommendation: 2.26.1 or later

https://www.gnu.org/software/binutils/

Advanced Micro Devices
One AMD Place, P.O. Box 3453
Sunnyvale, CA 94088-3453

AMD EPYC™ 7xx1 Series Processors
Compiler Options Quick Reference Guide

Intel compilers (icc, icpc, ifort)

Latest release: 18.3, May 2018

http://software.intel.com

Architecture	
Generate instructions that run on EPYC/RYZEN	-march=core-avx2
Optimization Levels	
Disable all optimizations	-O0
Speed optimization without code growth	-O1
Enable optimization for speed including vectorization	-O2
Aggressive optimization	-O3
Maximize performance	-fast
Additional Optimizations	
Aggressive unrolling	-unroll-aggressive
Disable improved precision floating divides	-no-prec-div
Enable vectorization	-vec
Inter procedural Optimization	-ipo
OpenMP	-qopenmp
Prefetch optimization	-qopt-prefetch
Profile generated optimization	-prof-gen and -prof-use
Use optimized header definitions	-use-intel-optimized-headers
Other options	
Floating point accuracy tuning	-fp-model
Compile free form FORTRAN	-free

Microsoft Visual Studio 2017

Latest release : 15.7, May 2018

https://www.visualstudio.com/

User Guide

Readme

Architecture	
Generate instructions that run on EPYC/RYZEN	/arch:[AVX|AVX2]
Optimize for 64-bit AMD processors	/favor:AMD64
Optimization Levels	
Disable optimizations	/Od
Maximum optimizations (favor space)	/O1
Maximum optimizations (favor speed)	/O2
[link.exe] Eliminate unreferenced function and/ or data	/OPT:REF
[link.exe] Perform identical COMDAT folding	/OPT:ICF
Output an informational message for loops that are auto-vectorized	/Qvec-report:[1|2]
Enable automatic parallelization of loops, used in conjunction with #pragma loop() directive	/Qpar
Output an informational message for loops that are auto-parallelized	/Qpar-report:[1|2]
Additional Optimizations	
Maintain the precision for floating-point operations through proper rounding	/fp:precise
Optimize floating-point code for speed at the expense of floating-point accuracy and correctness	/fp:fast
Whole Program Optimization (link-time code generation)	/GL
Profile-guided optimization	LTCG:PGI and /LTCG:PGO

AMD Optimized Libraries

Latest release: 1.0, Jun 2018

https://developer.amd.com/amd-cpu-libraries/

AMD µProf (Performance & Power Profiler)

Latest release: 1.2, Jul 2018

https://developer.amd.com/amd-uprof/

Advanced Micro Devices
One AMD Place, P.O. Box 3453
Sunnyvale, CA 94088-3453