

Gromacs 5.0 for UV2000

ウェブページ

<http://www.gromacs.org/>

バージョン

5.0

ビルド環境

- ▶ Intel Compiler 13.1.1
- ▶ MPT 2.0.7
- ▶ cmake 2.8.11

ビルドに必要なファイル

- ▶ [gromacs-5.0.tar.gz](#)
- ▶ [no-gcc-version.patch](#)
- ▶ [no-mpifinalized.patch](#)

パッチファイル

no-gcc-version.patch

```
--- cmake/gmxManageNvccConfig.cmake.orig 2014-07-09 09:43:31.108489646 +0900
+++ cmake/gmxManageNvccConfig.cmake 2014-07-09 09:44:24.668688300 +0900
@@ -109,16 +109,16 @@
 # as even with icc used as host compiler, when icc's gcc compatibility
 # mode is higher than the max gcc version officially supported by CUDA,
 # nvcc will freak out.
- if (UNIX AND CMAKE_C_COMPILER_ID MATCHES "Intel" AND
-   CUDA_HOST_COMPILER_AUTOSET)
-   if (CUDA_VERSION VERSION_LESS "4.1")
-     message(STATUS "Setting Intel Compiler compatibility mode to gcc 4.4 for nvcc host compilation")
-     set(CUDA_HOST_COMPILER_OPTIONS "${CUDA_HOST_COMPILER_OPTIONS};-Xcompiler;-gcc-version=440;")
-   else()
-     message(STATUS "Setting Intel Compiler compatibility mode to gcc 4.5 for nvcc host compilation")
-     set(CUDA_HOST_COMPILER_OPTIONS "${CUDA_HOST_COMPILER_OPTIONS};-Xcompiler;-gcc-version=450;")
-   endif()
- endif()
+# if (UNIX AND CMAKE_C_COMPILER_ID MATCHES "Intel" AND
+#   CUDA_HOST_COMPILER_AUTOSET)
+#   if (CUDA_VERSION VERSION_LESS "4.1")
+#     message(STATUS "Setting Intel Compiler compatibility mode to gcc 4.4 for nvcc host compilation")
+#     set(CUDA_HOST_COMPILER_OPTIONS "${CUDA_HOST_COMPILER_OPTIONS};-Xcompiler;-gcc-version=440;")
+#   else()
+#     message(STATUS "Setting Intel Compiler compatibility mode to gcc 4.5 for nvcc host compilation")
+#     set(CUDA_HOST_COMPILER_OPTIONS "${CUDA_HOST_COMPILER_OPTIONS};-Xcompiler;-gcc-version=450;")
+#   endif()
+#   endif()
set(CUDA_HOST_COMPILER_OPTIONS "${CUDA_HOST_COMPILER_OPTIONS}"
  CACHE STRING "Options for nvcc host compiler (do not edit!)." FORCE)
```

no-mpifinalized.patch

```
--- src/gromacs/utility/init.cpp.orig 2014-07-10 11:07:50.617514888 +0900
+++ src/gromacs/utility/init.cpp 2014-07-10 11:09:20.845939159 +0900
@@ -67,8 +67,6 @@
 {
 #ifdef GMX_LIB_MPI
   int isInitialized = 0, isFinalized = 0;
-   MPL_Finalized(&isFinalized);
-   GMX_RELEASE_ASSERT(!isFinalized, "Invalid attempt to initialize MPI after finalization");
   MPI_Initialized(&isInitialized);
   if (!isInitialized)
   {
```

注意事項

GMX_CPU_ACCELERATION=AVX_256にするとregressiontestsのfreeenergyの2つのテストでFailとなる。
そのため、SSE4.1も別途用意した。

ビルド手順

```
#!/bin/csh -f
umask 022
```

```

set file_gromacs=/home/users/${USER}/build/gromacs500/gromacs-5.0.tar.gz
set file_patch1=/home/users/${USER}/build/gromacs500/no-gcc-version.patch
set file_patch3=/home/users/${USER}/build/gromacs500/no-mpifinalized.patch
set prefix_avx256=/local/apl/uv/gromacs500
set prefix_sse41=/local/apl/uv/gromacs500_sse41
set work=/work/users/${USER}
source /opt/intel/composer_xe_2013.3.163/bin/compilervars.csh intel64
#-----
cd ${work}
if (-d gromacs-5.0) then
  mv gromacs-5.0 gromacs-erase
  rm -rf gromacs-erase &
endif
tar xzf ${file_gromacs}
cd gromacs-5.0
patch -p0 < ${file_patch1}
patch -p0 < ${file_patch3}
#
setenv CC icc
setenv CXX icpc
setenv F77 ifort
setenv F90 ifort
setenv FC ifort
mkdir rccs-s
cd rccs-s
cmake .. -DCMAKE_INSTALL_PREFIX=${prefix_avx256} \
  -DCMAKE_VERBOSE_MAKEFILE=ON \
  -DGMX_MPI=OFF \
  -DGMX_GPU=OFF \
  -DGMX_DOUBLE=OFF \
  -DGMX_FFT_LIBRARY=mkl \
  -DGMX_SYMLINK_OLD_BINARY_NAMES=OFF \
  -DREGRESSIONTEST_DOWNLOAD=OFF
make -j 12
make install
cd ..
#
mkdir rccs-d
cd rccs-d
cmake .. -DCMAKE_INSTALL_PREFIX=${prefix_avx256} \
  -DCMAKE_VERBOSE_MAKEFILE=ON \
  -DGMX_MPI=OFF \
  -DGMX_GPU=OFF \
  -DGMX_DOUBLE=ON \
  -DGMX_FFT_LIBRARY=mkl \
  -DGMX_SYMLINK_OLD_BINARY_NAMES=OFF \
  -DREGRESSIONTEST_DOWNLOAD=OFF
make -j 12
make install
cd ..
#
mkdir rccs-s-sse41
cd rccs-s-sse41
cmake .. -DCMAKE_INSTALL_PREFIX=${prefix_sse41} \
  -DCMAKE_VERBOSE_MAKEFILE=ON \
  -DGMX_MPI=OFF \
  -DGMX_GPU=OFF \
  -DGMX_DOUBLE=OFF \
  -DGMX_FFT_LIBRARY=mkl \
  -DGMX_CPU_ACCELERATION=SSE4.1 \
  -DGMX_SYMLINK_OLD_BINARY_NAMES=OFF \
  -DREGRESSIONTEST_DOWNLOAD=OFF
make -j 12
make install
cd ..
#
mkdir rccs-d-sse41
cd rccs-d-sse41
cmake .. -DCMAKE_INSTALL_PREFIX=${prefix_sse41} \
  -DCMAKE_VERBOSE_MAKEFILE=ON \
  -DGMX_MPI=OFF \
  -DGMX_GPU=OFF \
  -DGMX_DOUBLE=ON \
  -DGMX_FFT_LIBRARY=mkl \
  -DGMX_CPU_ACCELERATION=SSE4.1 \
  -DGMX_SYMLINK_OLD_BINARY_NAMES=OFF \
  -DREGRESSIONTEST_DOWNLOAD=OFF

```

```

make -j 12
make install
cd ..
#
setenv CC mpicc
setenv CXX mpicxx
setenv F77 mpif90
setenv F90 mpif90
setenv FC mpif90
mkdir rccs-mpi
cd rccs-mpi
cmake .. -DCMAKE_INSTALL_PREFIX=${prefix_avx256} \
-DMAKE_VERBOSE_MAKEFILE=ON \
-DGMX_MPI=ON \
-DGMX_GPU=OFF \
-DGMX_DOUBLE=OFF \
-DGMX_FFT_LIBRARY=mkl \
-DMPIEXEC=/opt/sgi/mpt/mpt-2.07/bin/mpirun \
-DGMX_SYMLINK_OLD_BINARY_NAMES=OFF \
-DREGRESSIONTEST_DOWNLOAD=OFF
make -j 12
make install
cd ..
#
mkdir rccs-mpi-d
cd rccs-mpi-d
cmake .. -DCMAKE_INSTALL_PREFIX=${prefix_avx256} \
-DMAKE_VERBOSE_MAKEFILE=ON \
-DGMX_MPI=ON \
-DGMX_GPU=OFF \
-DGMX_DOUBLE=ON \
-DGMX_FFT_LIBRARY=mkl \
-DMPIEXEC=/opt/sgi/mpt/mpt-2.07/bin/mpirun \
-DGMX_SYMLINK_OLD_BINARY_NAMES=OFF \
-DREGRESSIONTEST_DOWNLOAD=OFF
make -j 12
make install
cd ..
mkdir rccs-mpi-sse41
cd rccs-mpi-sse41
cmake .. -DCMAKE_INSTALL_PREFIX=${prefix_sse41} \
-DMAKE_VERBOSE_MAKEFILE=ON \
-DGMX_MPI=ON \
-DGMX_GPU=OFF \
-DGMX_DOUBLE=OFF \
-DGMX_FFT_LIBRARY=mkl \
-DGMX_CPU_ACCELERATION=SSE4.1 \
-DMPIEXEC=/opt/sgi/mpt/mpt-2.07/bin/mpirun \
-DGMX_SYMLINK_OLD_BINARY_NAMES=OFF \
-DREGRESSIONTEST_DOWNLOAD=OFF
make -j 12
make install
cd ..
#
mkdir rccs-mpi-d-sse41
cd rccs-mpi-d-sse41
cmake .. -DCMAKE_INSTALL_PREFIX=${prefix_sse41} \
-DMAKE_VERBOSE_MAKEFILE=ON \
-DGMX_MPI=ON \
-DGMX_GPU=OFF \
-DGMX_DOUBLE=ON \
-DGMX_FFT_LIBRARY=mkl \
-DGMX_CPU_ACCELERATION=SSE4.1 \
-DMPIEXEC=/opt/sgi/mpt/mpt-2.07/bin/mpirun \
-DGMX_SYMLINK_OLD_BINARY_NAMES=OFF \
-DREGRESSIONTEST_DOWNLOAD=OFF
make -j 12
make install
cd ..

```