Gromacs 5.0.4 for Primargy

Webpage

http://www.gromacs.org/

Version

5.0.4

Tools for Compiling

- Intel Compiler 15.0.2
- Cuda compilation tools, release 5.0, V0.2.1221
- Intel MPI 4.1.0.030
- cmake 2.8.8

Necessary Files for Compiling

- gromacs-5.0.4.tar.gz
- no-gcc-version.patch
- int128.patch

Patch Files

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 compatibity mode to gcc 4.4 for nvcc host compilation")
-               set(CUDA_HOST_COMPILER_OPTIONS "${CUDA_HOST_COMPILER_OPTIONS};-Xcompiler;-gcc-version=440;")
-           else()
-               message(STATUS "Setting Intel Compiler compatibity mode to gcc 4.5 for nvcc host compilation")
-               set(CUDA_HOST_COMPILER_OPTIONS "${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 compatibity mode to gcc 4.4 for nvcc host compilation")
+#               set(CUDA_HOST_COMPILER_OPTIONS "${CUDA_HOST_COMPILER_OPTIONS};-Xcompiler;-gcc-version=440;"
+#           else()
+#               message(STATUS "Setting Intel Compiler compatibity mode to gcc 4.5 for nvcc host compilation")
+#               set(CUDA_HOST_COMPILER_OPTIONS "${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)
```

int128.patch

```
--- src/external/boost/boost/config/compiler/intel.hpp.orig    2014-06-18 00:14:19.000000000 +0900
+++ src/external/boost/boost/config/compiler/intel.hpp    2014-07-09 09:59:27.000000000 +0900
@@ -35,10 +35,14 @@
 #endif

 #ifdef BOOST_INTEL_STDCXX0X
+#ifndef BOOST_COMPILER
 #define BOOST_COMPILER "Intel C++ C++0x mode version " BOOST_STRINGIZE(BOOST_INTEL_CXX_VERSION)
+#endif
 #else
+#ifndef BOOST_COMPILER
 #define BOOST_COMPILER "Intel C++ version " BOOST_STRINGIZE(BOOST_INTEL_CXX_VERSION)
 #endif
+#endif
 #define BOOST_INTEL BOOST_INTEL_CXX_VERSION

 #if defined(_WIN32) || defined(_WIN64)
@@ -303,10 +307,6 @@
 #  define BOOST_HAS_STDINT_H
 #endif

-#if defined(__LP64__) && defined(__GNUC__) && (BOOST_INTEL_CXX_VERSION >= 1310)
-#  define BOOST_HAS_INT128
-#endif
-
 //
 // last known and checked version:
 #if (BOOST_INTEL_CXX_VERSION > 1310
```

Procedure of Compiling

```
#!/bin/csh -f
umask 022
set file_gromacs=/home/users/${USER}/build/gromacs504/gromacs-5.0.4.tar.gz
set file_patch1=/home/users/${USER}/build/gromacs504/no-gcc-version.patch
set file_patch2=/home/users/${USER}/build/gromacs504/int128.patch
set prefix_avx256=/local/apl/pg/gromacs504
set work=/work/users/${USER}
source /opt/intel/composer_xe_2015.2.164/bin/compilervars.csh intel64
#------------------------------------------------------------------------------
cd ${work}
if (-d gromacs-5.0.4) then
  mv gromacs-5.0.4 gromacs-erase
  rm -rf gromacs-erase &
endif
tar xzf ${file_gromacs}
cd gromacs-5.0.4
patch -p0 < ${file_patch1}
patch -p0 < ${file_patch2}
#
setenv CC icc
setenv CXX icpc
setenv F77 ifort
setenv F90 ifort
setenv FC ifort
mkdir rccs-gpu
cd rccs-gpu
cmake28 .. -DCMAKE_INSTALL_PREFIX=${prefix_avx256} \
    -DCMAKE_VERBOSE_MAKEFILE=ON \
    -DGMX_MPI=OFF \
    -DGMX_GPU=ON \
    -DGMX_DOUBLE=OFF \
    -DGMX_THREAD_MPI=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
cmake28 .. -DCMAKE_INSTALL_PREFIX=${prefix_avx256} \
    -DCMAKE_VERBOSE_MAKEFILE=ON \
    -DGMX_MPI=OFF \
    -DGMX_GPU=OFF \
    -DGMX_DOUBLE=ON \
    -DGMX_THREAD_MPI=OFF \
    -DGMX_FFT_LIBRARY=mkl \
    -DGMX_SYMLINK_OLD_BINARY_NAMES=OFF \
    -DREGRESSIONTEST_DOWNLOAD=OFF
make -j 12
make install
cd ..
#
setenv CC mpiicc
setenv CXX mpiicpc
setenv F77 mpiifort
setenv F90 mpiifort
setenv FC mpiifort
mkdir rccs-mpi
cd rccs-mpi
cmake28 .. -DCMAKE_INSTALL_PREFIX=${prefix_avx256} \
    -DCMAKE_VERBOSE_MAKEFILE=ON \
    -DGMX_MPI=ON \
    -DGMX_GPU=OFF \
    -DGMX_DOUBLE=OFF \
    -DGMX_THREAD_MPI=OFF \
    -DGMX_FFT_LIBRARY=mkl \
    -DMPIEXEC=/opt/intel/impi/4.1.0.030/intel64/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
cmake28 .. -DCMAKE_INSTALL_PREFIX=${prefix_avx256} \
    -DCMAKE_VERBOSE_MAKEFILE=ON \
    -DGMX_MPI=ON \
    -DGMX_GPU=OFF \
    -DGMX_DOUBLE=ON \
    -DGMX_THREAD_MPI=OFF \
    -DGMX_FFT_LIBRARY=mkl \
    -DMPIEXEC=/opt/intel/impi/4.1.0.030/intel64/bin/mpirun \
    -DGMX_SYMLINK_OLD_BINARY_NAMES=OFF \
    -DREGRESSIONTEST_DOWNLOAD=OFF
make -j 12
make install
cd ..
```