Gromacs 2016.6-CUDA

Webpage

http://www.gromacs.org/

Version

2016.6

Build Environment

- Intel Parallel Studio 2017 Update 8
- CUDA 9.1

Files Required

- gromacs-2016.6.tar.gz
- (regressiontests-2016.6.tar.gz; testset)

Build Procedure

```
#!/bin/sh
VERSION=2016.6
INSTALL_PREFIX=/local/apl/lx/gromacs2016.6-CUDA

BASEDIR=/home/users/${USER}/Software/Gromacs/${VERSION}/
GROMACS_TARBALL=${BASEDIR}/gromacs-${VERSION}.tar.gz

#WORKDIR=/work/users/${USER}
WORKDIR=/home/users/${USER}/ramdisk
REGRESSION=${WORKDIR}/regressiontests-${VERSION} # unpacked

PARALLEL=12

# intel17+cuda-9.1
module purge
module load intel_parallelstudio/2017update8
module load cuda/9.1

export LC_ALL=C
export LANG=C

#------------------------------------------------------------------
umask 0022

cd ${WORKDIR}
if [ -d gromacs-${VERSION} ]; then
  mv gromacs-${VERSION} gromacs_erase
  rm -rf gromacs_erase &
fi

tar xzf ${GROMACS_TARBALL}
cd gromacs-${VERSION}

# compiler setting
export CC=icc
export CXX=icpc
export F77=ifort
export F90=ifort
export FC=ifort

# single precision, no MPI
mkdir rccs-s
```

```
cd rccs-s
cmake .. \
      -DCMAKE_INSTALL_PREFIX=${INSTALL_PREFIX} \
      -DCMAKE_VERBOSE_MAKEFILE=ON \
      -DGMX_MPI=OFF \
      -DGMX_GPU=ON \
      -DGMX_DOUBLE=OFF \
      -DGMX_THREAD_MPI=ON \
      -DGMX_FFT_LIBRARY=mkl \
      -DGMX_USE_NVML=OFF \
      -DREGRESSIONTEST_DOWNLOAD=OFF #\
      #-DREGRESSIONTEST_PATH=${REGRESSION}
#make -j${PARALLEL} && make check && make install
make -j${PARALLEL} && make install
cd ..

# compiler setting for MPI versions
export CC=mpiicc
export CXX=mpiicpc
export F77=mpiifort
export F90=mpiifort
export FC=mpiifort

# single precision, with MPI
mkdir rccs-mpi-s
cd rccs-mpi-s
cmake .. \
      -DCMAKE_INSTALL_PREFIX=${INSTALL_PREFIX} \
      -DCMAKE_VERBOSE_MAKEFILE=ON \
      -DGMX_MPI=ON \
      -DGMX_GPU=ON \
      -DGMX_DOUBLE=OFF \
      -DGMX_USE_NVML=OFF \
      -DGMX_THREAD_MPI=OFF \
      -DGMX_FFT_LIBRARY=mkl \
      -DREGRESSIONTEST_DOWNLOAD=OFF #\
      #-DREGRESSIONTEST_PATH=${REGRESSION} \
      #-DMPIEXEC=${MPIRUN} \
      #-DMPIEXEC_NUMPROC_FLAG="-np" \
      #-DNUMPROC=${MPI_NPROCS} \
      #-DMPIEXEC_PREFLAGS="" \
      #-DMPIEXEC_POSTFLAGS=""
#make -j${PARALLEL} && make check && make install
make -j${PARALLEL} && make install
cd ..
```

Notes

- Regression tests were performed on a P100 node.
- Gcc version might be faster in some cases. Please check the performance using your input.