

## Gromacs 2024.2-CUDA

### Webpage

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

### Version

2024.2

### Build Environment

- GCC 13.1.1 (gcc-toolset-13)
- Open MPI 4.1.6 (CUDA-aware)
- CUDA 12.4 Update 1

### Files Required

- gromacs-2024.2.tar.gz
- regressiontests-2024.2.tar.gz
- fftw-3.3.8.tar.gz

### Build Procedure

Built and tested on ccgpu.

```
#!/bin/sh

VERSION=2024.2
INSTALL_PREFIX=/apl/gromacs/${VERSION}-CUDA

BASEDIR=/home/users/${USER}/Software/Gromacs/${VERSION}/
GROMACS_TARBALL=${BASEDIR}/gromacs-${VERSION}.tar.gz
REGRESSION_TARBALL=${BASEDIR}/regressiontests-${VERSION}.tar.gz
WORKDIR=/gwork/users/${USER}
REGRESSION_PATH=${WORKDIR}/regressiontests-${VERSION}

PARALLEL=12
export LANG=C

FFTW_VER=3.3.8
FFTW_PATH=${BASEDIR}/fftw-${FFTW_VER}.tar.gz

#-----
umask 0022

module -s purge
module -s load gcc-toolset/13
module -s load cuda/12.4u1
module -s load openmpi/4.1.6/gcc13-cuda12.4u1

unset OMP_NUM_THREADS

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

if [ -d regressiontests-${VERSION} ]; then
  mv regressiontests-${VERSION} regressiontests_erase
  rm -rf regressiontests_erase &
fi

tar xzf ${GROMACS_TARBALL}
```

```
tar xzf ${REGRESSION_TARBALL}
cd gromacs-${VERSION}

# single precision, no MPI
mkdir rccs-s
cd rccs-s
cmake .. \
-DMAKE_INSTALL_PREFIX=${INSTALL_PREFIX} \
-DMAKE_VERBOSE_MAKEFILE=ON \
-DMAKE_C_COMPILER=gcc \
-DMAKE_CXX_COMPILER=g++ \
-DGMX_MPI=OFF \
-DGMX_GPU=CUDA \
-DGMX_DOUBLE=OFF \
-DGMX_THREAD_MPI=ON \
-DGMX_USE_CUFFTMP=OFF \
-DGMX_BUILD_OWN_FFTW=ON \
-DGMX_BUILD_OWN_FFTW_URL=${FFTW_PATH} \
-DREGRESSIONTEST_DOWNLOAD=OFF \
-DREGRESSIONTEST_PATH=${REGRESSION_PATH}
make -j${PARALLEL} && make check && make install
cd ..

# single precision, with MPI
mkdir rccs-mpi-s
cd rccs-mpi-s
cmake .. \
-DMAKE_INSTALL_PREFIX=${INSTALL_PREFIX} \
-DMAKE_VERBOSE_MAKEFILE=ON \
-DMAKE_C_COMPILER=mpicc \
-DMAKE_CXX_COMPILER=mpicxx \
-DGMX_MPI=ON \
-DGMX_GPU=CUDA \
-DGMX_DOUBLE=OFF \
-DGMX_THREAD_MPI=OFF \
-DGMX_USE_CUFFTMP=OFF \
-DGMX_BUILD_OWN_FFTW=ON \
-DGMX_BUILD_OWN_FFTW_URL=${FFTW_PATH} \
-DREGRESSIONTEST_DOWNLOAD=OFF \
-DREGRESSIONTEST_PATH=${REGRESSION_PATH}
make -j${PARALLEL} && make check && make install
cd ..
```

## Tests

All the tests have passed successfully.

## Notes

- (Please also check [notes for CPU version](#).)
- As for cuFFTMp, we have confirmed that it can be built with gcc9 or 12 + Open MPI 4.1.6 (cuda-aware) + NVIDIA HPC SDK 24.3 (CUDA 12.3 + NVSHMEM etc.). However, that executable failed on FFTMpiUnitTests test with transport error. It also failed in some of complex tests. This is due to the absence of [nv\\_peer\\_memory](#)? We have not evaluated `nv_peer_memory` so far. For `nvidia_peermem` in the error message below, there is a kernel module with that name.
  - (cuda-aware feature is not necessary for cuFFTMp gromacs? cuFFTMp enabled one can be built with Open MPI without cuda-aware feature. We don't know whether the binary without cuda-aware MPI works, though.)

```
nvshmemt_init:1811: neither nv_peer_mem, or nvidia_peermem detected. Skipping transport.
```

```
nvshmemi_transport_init:215: init failed for remote transport: ibrc
```