

VASP6 GPU

VASP6 GPU N

()

vasp vasp gpu openacc cuda
 vasp6.2 openacc

https://www.vasp.at/wiki/index.php/OpenACC_GPU_port_of_VASP

Openacc gpu Vasp6 NVIDIA HPC-SDK PGI's Compilers & Tools (version
>=19.10) vasp NVIDIA HPC-SDK **20.9**
 vasp bug

NVIDIA HPC-SDK 20.9

<https://developer.nvidia.com/nvidia-hpc-sdk-209-downloads>

wget tarball
 nvidia-smi cuda
 version >=10.0

HPC-SDK 20.9

/usr/software/nv-hpcsdk

```
export NVARCH=`uname -s`_`uname -m`;  
export NVCOMPILERS=/usr/software/nv-hpcsdk #  
export PATH=$NVCOMPILERS/$NVARCH/20.9/compilers/bin:$PATH  
export MANPATH=$MANPATH:$NVCOMPILERS/$NVARCH/20.9/compilers/man  
export LD_LIBRARY_PATH=$NVCOMPILERS/$NVARCH/20.9/compilers/lib/:$LD_LIBRARY_PATH  
export PATH=$NVCOMPILERS/$NVARCH/20.9/comm_libs/mpi/bin/:$PATH
```

gpu vasp6

bashrc

bashrc

intel mpirun

##

NVIDIA HPC-SDK 20.9

CUDA Toolkit, QD,

NCCL, ## FFTW, ## HPC-SDK

FFTW nvhpc-sdk

hpc-sdk

GNU intel

/fs00/software/fftw/3.3.8-ips2019u5

##

vasp6.2

```
cp arch/makefile.include.linux_nv_acc makefile.include`
```

##

```
which nvfortran | awk -F /compilers/bin/nvfortran '{ print $1 }`
```

nvfortran nv-hpc-sdk

vasp openacc+openmp

makefile.include.linux_nv_acc+omp+mkl nccl openacc

openmp

openmp

makefile.include.linux_nv_acc

makefile.include

makefile.include

#Precompiler options

```
CPP_OPTIONS= -DHOST=\"LinuxPGI\" \  
-DMPI -DMPI_BLOCK=8000 -DMPI_INPLACE -Duse_collective \  
-DscalAPACK \  
-DCACHE_SIZE=4000 \  
-Davoidalloc \  
-Dvasp6 \  
-Duse_bse_te \  
-Dtbdyn \  
-Dqd_emulate \  
-Dfock_dblbuf \  
-D_OPENACC \  
-DUSENCCL -DUSENCCLP2P
```

```
CPP      = nvfortran -Mpreprocess -Mfree -Mextend -E $(CPP_OPTIONS) $$$(FUFFIX) > $$$(SUFFIX)
```

```
FC        = mpif90 -acc -gpu=cc60,cc70,cc80,cuda11.0
```

```
FCL       = mpif90 -acc -gpu=cc60,cc70,cc80,cuda11.0 -c++libs
```

```
FREE      = -Mfree
```

```
FFLAGS    = -Mbackslash -Mlarge_arrays
```

OFLAG = -fast

DEBUG = -Mfree -O0 -traceback

#Specify your NV HPC-SDK installation, try to set NVROOT automatically

NVROOT=\$(shell which nvfortran | awk -F /compilers/bin/nvfortran '{ print \$1 }')

#or set NVROOT manually

#NVHPC ?= /opt/nvidia/hpc_sdk

#NVVERSION = 20.9

#NVROOT = \$(NVHPC)/Linux_x86_64/\$(NVVERSION)

#Use NV HPC-SDK provided BLAS and LAPACK libraries

BLAS = -lblas

LAPACK = -llapack

BLACS =

SCALAPACK = -Mscalapack

CUDA = -cudalib=cublas,cusolver,cufft,nccl -cuda

LLIBS = \$(SCALAPACK) \$(LAPACK) \$(BLAS) \$(CUDA)

#Software emulation of quadruple precision

QD = \$(NVROOT)/compilers/extras/qd #□□□□□□

LLIBS += -L\$(QD)/lib -lqdm -lqd

INCS += -I\$(QD)/include/qd

#Use the FFTs from fftw

FFTW = /fs00/software/fftw/3.3.8-ips2019u5 #□□ fftw□□□□□□□□

LLIBS += -L\$(FFTW)/lib -lfftw3

INCS += -I\$(FFTW)/include

OBJECTS = fftmpi.o fftmpi_map.o fftw3d.o fft3dlib.o

#Redefine the standard list of O1 and O2 objects

SOURCE_O1 := pade_fit.o

SOURCE_O2 := pead.o

#For what used to be vasp.5.lib

CPP_LIB = \$(CPP)

```
FC_LIB    = nvfortran
CC_LIB    = nvc
CFLAGS_LIB = -O
FFLAGS_LIB = -O1 -Mfixed
FREE_LIB  = $(FREE)
```

```
OBJECTS_LIB= linpack_double.o getshmem.o
```

```
#For the parser library
```

```
CXX_PARS = nvc++ --no_warnings
```

#Normally no need to change this

```
SRCDIR    = ../../src
```

```
BINDIR    = ../../bin
```

[illegible]

```
/usr/software/nv-hpcsdk  nv-hpc-sdk  
```

```
make std gam ncl
```

■■■■■■■■■■ openacc■■■■■■■■■■ make gpu■■■■■■■■ vasp_std

--	--	--	--	--	--	--

1. ☐ nccl☐☐☐☐ openacc☐ gpu☐☐☐☐☐☐

2.INCAR NCORE openacc 1.

3.INCAR NSIM KPAR KPAR

GPU NSIM cpu .

GPU

VASP_

Revision #8
Created 16 November 2021 16:15:14 by Yao Ge
Updated 8 February 2023 23:08:00 by Yao Ge