1. Change working directory to “/home/vbidari/Term\_Project/programs”

Use command : *cd home/vbidari/Term\_Project/programs*

Source files of each parallel program is placed in working directory as follows

1. MPI\_Send\_Recieve.c : Program uses MPI send and receive functions to perform communication.
2. MPI\_Collective\_Communication.c : Program uses MPI collective communication such as scatter and gather to communication.
3. MPIopenMP.c : This program is based hybrid programming of MPI and Openmp.
4. Task\_and\_Data\_Parallel\_Model.c: This program uses Task and data parallel model.

Folder with name “sample” is provided with this project.

1. The compile and execution command for each above program as follows.
2. To compile MPI\_Send\_Recieve.c as follows

*mpicc –c MPI\_Send\_Recieve.c*

*mpicc –o MPI\_Send\_Recieve MPI\_Send\_Recieve.o*

#Run

*qsub –pe mpich 2 MPI\_Send\_Recieve.bash*

*qstat*

1. To compile MPI\_Collective\_Communication.c as follows

*mpicc –c MPI\_Collective\_Communication.c*

*mpicc –o MPI\_Collective\_Communication MPI\_Collective\_Communication.o*

#Run

*qsub –pe mpich 2 MPI\_Collective\_Communication.bash*

*qstat*

1. To compile MPIopenMP.c as follows

Here we add flag fopenmp for including OpenMP compilation.

*mpicc –c MPIopenMP.c -fopenmp*

*mpicc –o MPIopenMP MPIopenMP.c.o -fopenmp*

#Run

*qsub –pe mpich 2 MPIopenMP.bash*

*qstat*

1. To compile Task\_and\_Data\_Parallel\_Model.c as follows

*mpicc –c* Task\_and\_Data\_Parallel\_Model*.c*

*mpicc –o* Task\_and\_Data\_Parallel\_ModelTask\_and\_Data\_Parallel\_Model*.o*

#Run

*qsub –pe mpich 2 Task\_Data\_Parallel.bash*

*qstat*