**Building HPL using Ampere Oracle Blis Libraries.**

**rbapat@amperecomputing.com**

Building HPL with Ampere-Oracle Blis libraries is very easy and should not take a lot of time. It’s a 2-step process:

Step 1 where we build the Math libraries found on the Ampere branch of Oracle Blis libraries and Step 2 where we build the HPL binaries.

A detailed guide is below:

**Step 1: Prerequisites:**

To ensure a seamless build process, both the math libraries and the benchmark are built inside the /opt directory.

1. Downloading and installing Ampere Oracle Blis Libraries:

pushd /opt

git clone https://github.com/flame/blis.git MyBlisDir

pushd MyBlisDir

#Switch to the new ampere branch

git checkout ampere

./QuickStart.sh altramax

#Ensure that the test bench contains Ampere Oracle Blis exported to PATH and LD\_LIBRARY\_PATH appropriately.

source ./blis\_build\_altramax.sh

source blis\_setenv.sh

export LD\_LIBRARY\_PATH=/opt/MyBlisDir/lib/altramax

popd

popd

1. OpenMPI: Along with Ampere Oracle Blis, we will also need openmpi. We have used openmpi 4.1.4. An installation guide for openmpi can be found inside the tarball: https://download.open-mpi.org/release/open-mpi/v4.1/openmpi-4.1.4.tar.gz

If OpenMPI is installed in a non-default location. Add the <bin> directory location to PATH and the <lib> directory location to LD\_LIBRARY\_PATH using the following commands

export PATH=<PATH\_TO\_OPENMPI\_BIN\_DIR>:$PATH

export LD\_LIBRARY\_PATH=< PATH\_TO\_OPENMPI\_LIB\_DIR>:$LD\_LIBRARY\_PATH

#Ensure successful installation of openmpi by executing the following commands.

mpirun --version #(That should bring up the openmpi version. 4.1.4 in this case)

mpicc --version #(That should bring up the installed gcc version)

mpic++ --version #(That should bring up the installed g++ version)

mpifort --version #(That should bring up the installed gfortran version)

#If any of the above 3 commands do not return the version for gcc/g++/gfortran, install the missing gcc/g++/gfortran for your distro using “sudo apt/yum/dnf install <package\_name>”

**Step 2: Building HPL Benchmark**

1. Downloading and Installing HPL 2.3

pushd /opt

wget https://netlib.org/benchmark/hpl/hpl-2.3.tar.gz

tar -xzf hpl-2.3.tar.gz

popd

#Copy the Makefile attached with this document to /opt/hpl-2.3 folder.

cp Make.Altramax\_oracleblis /opt/hpl-2.3

#Compile the HPL binary

make arch=Altramax\_oracleblis -j

#Upon success, a bin folder will be created. This folder should contain 2 files xhpl (which is the HPL #binary) and HPL.dat (which is the standard input file).

pushd /opt/hpl-2.3/bin/Altramax\_oracleblis

**Step 3: Creating the HPL input file**

Sample HPL.dat file attached

Copy the attached HPL.dat file to “/opt/hpl-2.3/bin/Altramax\_oracleblis “

**Step 4: Run the benchmark**

mpirun -np 96 --bind-to core --map-by core ./xhpl &> out.log

**Step 5: Performance Expectations**

If your system differs from our testbench, the HPL.dat file will need to be modified (line #6) to match your respective Altra Max config.

|  |  |  |
| --- | --- | --- |
| Line Number | Value | Description |
| 6 | 150000 | Ns |

Table 1: Showcasing the line which needs to be modified in HPL.dat

The value of N when changed to 150K should take approximately 180 gigs of memory and would run on a machine having 256 GB memory. Table2 with differing values of Ns is shown below with our reference numbers.

Our observed Results for AltraMax 128 cores @ 3.0GHz:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Input Param | Input Param | Ns | Memory Used | Results (Gflops) |
| NB=256 | P=8 Q=16 |
| Input Problem Size (Ns) | | 150K | 177 gig | 1528 |
| 200k | 312 gig | 1552 |
| 250k | 480 gig | 1597 |

Table 2: Varying sizes on Ns w.r.t available system memory.