ARM Cluster: The Research Tool

Andrew Hoover, Christine Sorensen and Christer Karlsson
Mathematics and Computer Science
South Dakota School of Mines and Technology
Rapid City, SD 57701
christine.sorensen@mines.sdsmt.edu

Abstract

A common trend in computing over the last few years has been to create more computational power by connecting cheap, less powerful computing platforms into cluster. The initial purpose of this project was to build an ARM cluster of 6-12 homogeneous single board computers to make it the fastest and most efficient in cost and energy with the intention to show a proof of concept.

We decided that the metric to measure efficiency in cost and energy would be the number of floating-point operations per dollar per unit of power. Three types of single board computers were tested: Raspberry Pi 2B, PcDuino, and ODROID XU4. PcDuino was immediately dropped due to the cost of $160 per board and problems with the operating system. Therefore, it was inferred that the computer would not excel in the tests.

An Open Multi-Processing (OpenMP) benchmark was installed on each of the computers running various mathematic equations on all cores. In the results of the benchmark, the Raspberry Pi performed best with 0.000217 Gigaflops/Dollar/Watt compared to ODROIDs 0.000203 Gigaflops/Dollar/Watt. However, the ODROID XU4 was chosen because bandwidth potential was higher and it ran 7.4 times faster. Eight ODROIDs were purchased and connected in a star topology using an unmanaged switch. To benchmark the cluster, it was decided to use LINPACK, a software that performs numerical linear algebra that is commonly used for cluster benchmarking. However, LINPACK benchmarking software did not exist for an ARM cluster. An open source project was adjusted to perform on the ARM architecture. This led to a side goal; to be able to release an ARM version of the LINPACK testing package.

LINPACK utilized Message Passing Interface (MPI) software to run in parallel on all cores and computers, testing every combination of number of cores and matrices. The benchmark was designed to fill as much available memory on the eight devices as possible and use every core on each machine. The results of the initial benchmarking with communication over Ethernet was a maximum of 13.38 GFLOPS using all eight cores on all eight devices. This was compared to a standard i7 desktop which benchmarks to about 47.5 GFLOPS.

The next stage of this project is to compare the results using Ethernet connection against connections using the other means of communications, specifically the Universal Serial Bus (USB) and General-Purpose Input/Output (GPIO).

The purpose of this project has over time drifted into using the cluster as an education tool. Answering questions such as how computers work, how do we build and setup networks, and how can we communicate between the computers and benchmark the performance? Also, how can we make this a cost efficient educational tool for teaching parallel computing, networks, or assembly, for institutions and organizations that normally do not have access to clusters?