The Blue Gene/L Supercomputer: Architecture and Implementation

David Gregg and Jake Johnson
Introduction

On March 11, 2002, the NEC Corporation finished construction of the Earth Simulator, and for two years, it reigned as the fastest computer on the planet. Unleashed in Yokohama, Japan, the supercomputer reached a record speed of 35.86 teraflops. For awhile, it appeared that America had fallen behind in the race to create the world’s fastest machine. But even as the Earth Simulator stood atop the world, IBM and Lawrence Livermore National Laboratory were releasing plans to create its successor, the Blue Gene/L Supercomputer.

The Blue Gene/L was to be the first in a line of computers that would be designed to eventually pass the petaflop mark. Blue Gene/C, Blue Gene/P, and Blue Gene/Q were, and still are, scheduled to follow the Blue Gene/L. The Blue Gene/L would be the computer to break the speed record set by Earth Simulator, with a targeted speed of 360 teraflops\(^1\). It would be a specific-purpose computer used for performing calculations and simulations in the area of life sciences.

This jointly-funded project between IBM and Lawrence Livermore National Laboratory is currently under construction. But even though the supercomputer has not yet been fully installed, it has already surpassed the Earth Simulator to become the world’s fastest machine. By September 29, 2004, a prototype of the Blue Gene/L had surpassed the 35.86 teraflops of Earth Simulator with a speed of 36.01 teraflops. Shortly thereafter, it had reached a mark of 70.72 teraflops. On March 24, 2005, the U.S. Department of Energy, who had recently purchased the Blue Gene/L, announced that the machine had
broken its own speed record, performing 135.5 teraflops. Yet the Blue Gene/L is not yet complete; the machine currently has half the processors intended for the final design.\(^7\)

This document investigates what makes the Blue Gene/L supercomputer so fast, dissects its architecture, and explains the operations that it can and cannot do. It is not intended to be exhaustive, but rather, it is an overview of the basic philosophy behind the architecture and implementation of the Blue Gene/L. For further research on this topic, see “An Overview of the BlueGene/L Supercomputer,” released in 2002 by the BlueGene/L Team from IBM and Lawrence Livermore National Laboratory\(^1\) and “Unlocking the Performance of the BlueGene/L Supercomputer,”\(^2\) released in 2004.

This document will discuss the following topics:

- The general approach of the Blue Gene/L design
- A system overview
- Special features of the nodes used in Blue Gene/L
- The Blue Gene/L operating system
- Programming models used for parallelization
- Applications that may be able to take advantage of the Blue Gene/L
- Comparison of architecture with other supercomputers

In order to understand why the Blue Gene/L supercomputer is so fast, one must understand the somewhat revolutionary approach to the problem.
General Approach

The most obvious goal with supercomputing is to create a computer that runs as fast as possible. The typical way to address the problem is to group a bunch of computers together that are each as fast as current technology would allow and then give them each a large amount of computation responsibility. The BG/L takes an entirely different approach. It had become evident to the BG/L team that the typical solution had outgrown its usefulness in creating increasingly powerful machines. The large, fast SMP’s used in other supercomputers consumed increasingly large amounts of electricity, and the addition of more processors delivered additional processing power at a decreasing rate. Therefore, the BG/L team decided to implement the supercomputer by grouping a “very large number of nodes,” each with a modest clock rate of approximately 700 MHz\(^1\). These nodes would be efficient and cost-effective, making it feasible to group thousands of them together. Each node would then be responsible for relatively simple tasks, accomplished through mass parallelism. These concepts would allow the BG/L to efficiently process large amounts of data.

The BG/L design would have the following features:

- IBM PowerPC embedded CMOS processors
- Embedded DRAM
- System-on-chip techniques
- Dual-processor design

The dual-processor design would utilize one processor for computation and the other for
communication, and these processors would communicate via multiple high-speed networks.\footnote{1}

The BG/L team estimated that the supercomputer’s peak performance would be 360 teraflops, which is only applicable for applications that would take advantage of both processors on the node. For applications that do not use both processors, the expected peak performance would be 180 teraflops.\footnote{1}

**System Overview**

The implementation of this philosophy would require the grouping of several nodes, each with a single Application Specific Integrated Circuit (ASIC) and Memory. The ASIC used by the BG/L is designed for BG/L’s intended application. Each node has about 2 gigabytes of local memory. The overall size of each node is relatively small – about 11.1 mm square die size. This allows for concentrated computation specific to the simple task that the node is assigned.\footnote{1}

The final configuration for the BG/L calls for 65,536 processors, which are packaged in the following manner. Two nodes are grouped on a compute card, each with two processors. There are 16 nodes per node board and 16 node boards per 512-node midplane. The midplanes measure approximately 17”x21”x34.” Two of the midplanes are grouped together in a 1024-node rack.\footnote{7} The following diagram shows the packaging in detail.
As illustrated in the diagram, the final system consists of 64 1024-node cabinets. Currently, only half of these racks have been installed.

Logically, when more cabinets are added, the system is able to reach a higher speed. Each processor can perform about 4 floating point operations per cycle, which leads to a performance of about 1.4 teraflops for a single midplane. Therefore, in theory, each cabinet adds about 2.8 teraflops of compute speed. In practice, the numbers are noticeably less than that, but the pattern holds that doubling the number of racks roughly doubles the compute power. When BG/L surpassed the Earth Simulator as the fastest computer, it reached a speed of 36.01 teraflops. The number of racks was later doubled to reach a nearly-double speed of 70.72 teraflops. Most recently, when the number of
racks was doubled again to 32, the BG/L accomplished 135.5 teraflops, nearly double its previous record.\textsuperscript{7}

In addition to the 65,536 dual-processor compute nodes, the system has 1024 I/O processors, or about one I/O node for every 64 compute nodes. The I/O nodes have the same ASIC as the compute nodes, but with an expanded external memory. Each compute node has a lightweight kernel that handles basic communication and the functions that high performance scientific code would require.\textsuperscript{1}

The nodes are interconnected using 5 different networks:\textsuperscript{1}

- A 3D torus network for point-to-point messaging between nodes
- Global combining and broadcast tree for operations relating to all nodes
- Global barrier and interrupt network
- Gigabit Ethernet to Joint Test Access Group (JTAG) network for machine control
- A second Gigabit Ethernet network for connection to other systems, such as hosts and file systems

The two most important networks for communicating are the torus network and the tree network. The BG/L does its general communicating via the torus network. A torus network essentially connects each node by giving each node 6 adjacent neighbors. There is a node to the left, right, top, bottom, front and back, and then they loop around on the ends. The bandwidth for these links are 2 bits/cycle or 175MB/s at 700 MHz. Each message that is passed is broken up into different packets that can range in size between
32 bytes-256 bytes in 32 byte increments. The BG/L uses a tree network for collective communication patterns that often occur, such as broadcasting or reduction operations. A tree network is a network that combines two or more star networks. Star networks are networks where all of the workstation nodes are linked to one central node. In the BG/L’s tree network, each of these central nodes for the star networks are linked together. The BG/L’s tree network has a bandwidth of 350MB/s, much faster than the 175MB/s link of the torus network.

Between the midplanes, a “link” ASIC is implemented, which serves two purposes:

1) It redrives (and therefore strengthens) the signal between midplanes, and

2) It allows the signals to be redirected between different ports

Thus, the BG/L can be partitioned into many logically separate systems, and traffic interference is nearly eliminated.

**Node Features**

A floating point unit (FPU) is a dedicated execution unit that performs operations on floating point numbers - any numbers that are not integers. The BG/L has a double floating point unit (DFPU) in its architecture. This DFPU was built by taking the architecture from an FPU and merging it with another FPU, using one as the primary and one as the secondary. The secondary FPU is basically a duplicate copy of the primary one, but it has its own set of special parallel instructions. According to the BG/L team, these instructions include “parallel add, multiply, fused multiply-add, and addition
operations to suppose complex arithmetic. All of the SIMD instructions operate on
double-precision floating point data. But that does not mean that the second FPU
cannot do single floating point operations. The instruction set also provides a standard
for this, using extensions.

Code generation for the DFPU can be done with the IBM XL TOBEY compiler. This
compiler translates C, C++ or FORTRAN source code into an intermediate language.
TOBEY recognizes where the complex arithmetic floating point computations occur and
then uses the SIMD-like extensions of the BG/L to efficiently implement the
computations.²

Every node on the BG/L has two PowerPC 440 processors with targeted clock speed of
700MHz. The processors have two execution modes. The default mode is the
communication mode, which involves one processor doing all of the communicating,
while the other does general processing. The second is called the virtual mode. With
this, each processor acts as an independent processor, because the resources are divided
between the two. Each processor gets half of the memory a separate MPI task. These
tasks share use of the network and memory. A special region of shared non-cached
shared memory allows communication between the tasks in the same node.²

**Operating System**

The compute nodes use the Linux operating system, and the I/O nodes use a Linux-based
system⁶. The general approach is to split the operating system’s functionality between
the compute and I/O nodes\textsuperscript{1}.

The I/O nodes provide the physical interface to the file system and various other processes that would be burdensome for the compute nodes. This allows the compute node software to be kept simple, which maintains the BG/L philosophy of distributing minimal instructions to many processors.\textsuperscript{1}

As for the compute nodes, the operating system is simple and lightweight. The kernel can support a dual-threaded application process.\textsuperscript{1}

Because the BG/L is based on Linux, a lot of the testing for it was done on Linux clusters to simulate the BG/L system. The BG/L team created a parallel application to simulate it, called BGLism. This actually let the developers develop and test the BG/L software before the hardware even arrived, which helped speed up the process.\textsuperscript{6}

**Programming Models**

Because the BG/L application is run in parallel, a message passing programming model is used. The BG/L uses an implementation of the MPI message-passing library.\textsuperscript{1} Because of its use of BG/L’s nodes and effective communication, MPI allows for the most efficient use of BG/L’s resources\textsuperscript{6}.

**Applications**

It must be emphasized that the BG/L is not a general-purpose computer; it cannot run all
applications. The BG/L is intended for specific scientific applications involving many floating-point operations. It was designed to solve grid-based problems that involve nodes communicating with the nearest neighbor and the need for a lot of computational power. Therefore only a few applications can take advantage of the BG/L.

To aid in the design of the BG/L’s hardware and software, the design team has borrowed concepts from many applications from the NNSA laboratories of the U.S. Department of Energy.¹

Applications that have a high flops/loads ratio and alternating compute/communicate behavior can take advantage of the BG/L’s unique architecture¹. The extra floating point unit in each node makes it possible to compute floating point operations at a higher rate. So if, for each load, a high number of floating point instructions are called, the system will perform significantly faster than if few floating point operations were used.

Most of the problems that the BG/L will solve are found in high-energy physics, molecular dynamics and astrophysics. The BG/L is based primarily around the problem of molecular simulation, particularly protein dynamics. The intent then, in the words of the BG/L team, is to “obtain a microscopic view of thermodynamics and kinetics of the [protein] folding process.”¹ The BG/L will likely be used in the search for cures to diseases such as Alzheimer’s, cystic fibrosis and Mad Cow Disease by looking at how folded/misfolded proteins effect their development.
Comparison

The BG/L architecture makes it the fastest supercomputer in the world. It is designed to have 65,536 compute nodes that each have two PowerPC 440 processors. The BG/L links these nodes using multiple networks for different types of problems. A three-dimensional torus network is used for point-to-point communication, and a tree network is used for collective operations.

In contrast, the Earth Simulator is based on SX-6 architecture. An SX-6 is a single node system that has up to 8 vector processors that all share up to 64 gigabytes of memory. A vector processor “is a CPU design that is able to run mathematical operations on a large number of data elements very quickly.” The Earth Simulator has 640 nodes with 8 vector processors and 16 gigabytes of memory each.

However, the Earth Simulator is over 5 times faster than the ASCI White. The ASCI White architecture included nodes that each had 16 processors. It had 8192 processors and 6 terabytes of memory.

The BG/L’s approach of using multiple, modest nodes allowed it to surpass the Earth Simulator and ASCI White to become the fastest supercomputer in the world.

Conclusion

The Blue Gene/L supercomputer uses an unconventional approach to the problem of supercomputing. By parallelizing simple tasks among many nodes, the BG/L can achieve
record speeds running the specific applications for which it was developed. The nodes themselves use a dual-processor system, and each processor is given a second floating point unit to increase the speed of floating-point-intensive applications, such as those related to life sciences.

Blue Gene/L now stands atop the world as the fastest supercomputer, even before its completion. Blue Gene is expected to be complete by July or August of 2005. Although it would be too optimistic to think that it will reach its theoretical estimation of 360 teraflops, time will tell how close it will come. Still, whether or not it reaches that goal, the Blue Gene/L computer is a monumental step toward the creation of a petaflop machine.
Works Cited


7 Wikipedia (http://en.wikipedia.org)