Dynamically Reconfigurable Computing:
A Novel Computation Technology with Potential to Improve National Security Capabilities by Dramatically Improving High-End Computing

A White Paper Prepared by Star Bridge Systems, Inc.
Brent D. Ward and Allan E. Snavely

May 15, 2003

Introduction

This white paper provides information about a significant architectural innovation – dynamically reconfigurable supercomputing – for the purpose of assisting the HECRTF in implementing the federal government's high-end computing program objectives. Dynamically reconfigurable supercomputing offers a potential pay-off for high-end computing by providing resources, tools, and techniques needed to minimize "time to solution" by users of high-end systems.

Company Background

Starbridge is a Midvale, Utah based start-up company with the vision: "The Shortest Distance from Thought to Solution". Building on fifteen years of research by the company's CTO, the company is developing a new architectural model for high-end computing that inherently supports both fine-grain and coarse-grain parallelism and dynamically tunes its architecture to fit the underlying application – namely a parallel computer system (software and hardware) with both conventional and reconfigurable processing elements and a reconfigurable interconnect. The company's prototype systems are inherently parallel (both hardware and software), ultra-tightly-coupled, high-level programmable, highly scalable reconfigurable computer (RC) systems with the potential to evolve into high-end, general-purpose supercomputers.

Summary of Technologies Under Development by Starbridge

There is evidence that reconfigurable systems can deliver 10X to 100X or greater improvement in computational efficiency (compared to traditional RISC processor based machines) for many problems by tailoring hardware allocations to match the needs of applications. Dynamically

reconfigurable supercomputers can potentially contribute to important value metrics, including time-to-solution, by reducing design cycle time and porting costs, reducing electrical power usage, enhancing fault-tolerance, and providing hardware support for intrusion detection and handling. Built from commodity components such as FPGAs, they also leverage commodity R&D, while providing benefits usually associated with expensive "balanced" HEC systems from the vector line.

FPGAs have traditionally been used for hardware prototyping or as glue logic, but today's gates are clocked at rates that make the prototype nearly as fast as the end product and much more flexible. In sheer density, FPGAs are outpacing Moore's Law. Thus, they have the capability, especially aggregated on specially designed printed circuit boards, to become self-contained, high-end supercomputers. Moreover, their flexibility raises the possibility of *meta-architecture*; "morphing" hardware configurations with software as needed to improve efficiency, robustness, security and capability on-the-fly. With such a system the applications designer can design algorithms to maximize the use of silicon, something not possible with today's type-T (high ratio of computation to communication) and type-C (high communication bandwidth to global memory and between processors) machines. A third order of programmability offers the designer the capacity to dynamically change the make-up and organization of the entire compute substrate – computational elements, communications topology and memory – to optimize the system for the problem at hand. This level of programmability has been reduced to practice in Starbridge's Viva® software. The enhanced programmability of dynamically reconfigurable supercomputers running on Viva can improve time-to-solution by significantly expediting the process of developing, modifying and validating software.

Prototype systems from Starbridge demonstrating many of the benefits of dynamically reconfigurable supercomputing are currently undergoing proof of concept evaluation and testing by (1) NSA at Ft. Meade, Maryland and at various NSA sponsored research institutions; (2) NASA's Langley Research Center in Hampton, Virginia and Marshall Space Flight Center in Huntsville, Alabama; and (3) U.S. Air Force Munitions Directorate at Eglin AFB, Florida.

Benefits of Dynamically Reconfigurable Supercomputing for High-End Computing

Dynamically reconfigurable computing achieves – or holds the promise of achieving – breakthroughs that significantly enhance systems performance. Breakthroughs in reconfigurable computing software will bring a more productive programming environment to high-end computing. Algorithm development will be faster and more intuitive. New tools will permit seamless algorithm design at a high level of abstraction and execution at a high level of efficiency in hardware, synthesizing layers of parallel execution structures all the way to the gate level without the need for unwieldy hardware assembly languages such as VHDL. These tools support both algorithm level programming and highly efficient mapping of algorithms into hardware. They synthesize only the necessary and sufficient FPGA circuitry for each task and dynamically reconfigure tens of millions of gates to perform multiple tasks at the same time. As virtually every gate performs useful information transformative work in each clock cycle, increased circuit-packing density can be achieved. These improvements will fully

exploit the inherent parallelism of FGPAs and leverage it in a way that more than compensates for slower FPGA clock speeds.

Because it is implementation independent and supports infinitely reusable system libraries, Starbridge's new RC software is portable and enables integration of previously disparate technologies – such as dense conventional logic, reprogrammable logic, dense DRAM memories and non-digital interface technologies such as optics – into a single, homogeneous system. Cross platform integration is also possible. Current RC systems already integrate CPUs and reprogrammable devices. Beyond this, entire high-end systems of different types, e.g. vector and clustered SMP systems, may be integrated using software to emulate non-RC systems in reconfigurable circuitry. The new software will also be portable to the quantum computing paradigm. All of these potential benefits are inherent in the Viva software system that has been developed and is being extended for capability at Starbridge.

Breakthroughs in RC hardware already enable ultra-tight coupling of compute, memory and I/O systems in a fractal-like hierarchy yielding shorter wire paths. These breakthroughs bring improvements in memory latency and bandwidth, as well as higher sustained I/O rates at all levels of the system (both on-off chip and to peripherals). Improved memory latency and bandwidth in prototype RC systems promise efficient utilization of larger storage devices, as they become available. High bandwidth/low latency interconnects and switching between logic, memory and storage on both localized and distributed levels are possible to meet the demands of increased distance and complexity in future high-end supercomputers. All of these potential benefits are inherent in the Hypercomputers® under development at Starbridge. Starbridge and SGI are already exploring a merger of technologies to achieve significant memory latency, bandwidth and capacity improvements by combining SGI's advanced memory and access technology with Starbridge's Hypercomputer hardware and Viva programming environment.

The expected results of research and development in dynamically reconfigurable computing will include minimized time-to-solution, faster computational speed, better scalability, lower power consumption, smaller footprint size, reduced cost and improved reliability. In some cases improvements are expected to be orders-of-magnitude in size.

These software and hardware breakthroughs are reflected in early benchmarking of dynamic reconfigurable supercomputer systems in specific applications. For example, a Starbridge HC62 Hypercomputer performed the Smith-Waterman bioinformatics algorithm at a rate of 75 billion Smith-Waterman steps per second, many times faster than conventional solutions. Moreover, the design cycle time to port this algorithm to Viva foreshadows a dramatic improvement in time-to-solution for many high-end supercomputing applications.

Features of Starbridge's RC Software

Starbridge's Viva software runs on a traditional CPU and includes a high-level graphical algorithm description language. It emphasizes high-level control flow and reusable library function calls. The user selects library functions to perform calculations, then interconnects these into a control-flow graph via a "point-and-click" interface. A synthesis tool develops net

lists directly from the high level language, and placement and routing tools place the net lists directly into the reconfigurable Hypercomputer. In this manner, Viva instantiates algorithmic block diagram descriptions of desired behavior directly into system level hardware configurations. Computational, communications and memory objects may all be created by this basic process. The objects created may then be executed in whatever system was previously described as the implementation environment. The result is a set of algorithm design tools for users who lack the time or expertise to create their own application specific circuits using traditional design software.

Data sets in Viva are polymorphic – i.e. they may be of any type, size and precision – resulting in infinitely flexible, reusable data set objects. Each time a data set is used, the developer specifies whether the data set will be an 8-bit integer or a 256-bit integer, or anything in between (any size may be defined). Or, the developer may specify a floating point number, fixed point number, complex number, or vectors of any of these types. Each of these may also be of any size. Data sets are also dynamic, i.e. their size (range or precision) may change within a Viva program, allowing precision changes to be specified anywhere in a program. Viva library objects also "morph" as information rates vary during algorithm execution. At runtime, Viva configures the hardware architecture that best suits the calculations. As the program changes phase, the hardware can be reconfigured beneath it to maintain high efficiency. The programmer has explicit control over the resulting application's performance and hardware size requirements. The programmer may tune the algorithm for intended results, either maximum performance (by using more gates) or smallest physical size (by using fewer gates).

Features of Starbridge RC Hardware

Starbridge has developed a patented FPGA architecture that combines any number from two to hundreds of FPGAs to achieve optimum hardware performance. It is fractal-like in design, i.e. it follows a pattern in which the structure of each level of hardware resources is repeated at the next higher level. Up to eleven FPGAs are arrayed on proprietary PCBs, which in turn may be expanded to multiple boards operating in a PCI-X (or other) communications environment.

One prototype system employs a dual processor motherboard and a single Hypercomputer board with nine Xilinx XC2V6000-BG1152 Virtex-II FPGAs and two XC2V4000-BG1152 Virtex-II FPGAs, yielding approximately 62 million gates per board. Three FPGAs function as the PCI-X bus interface, cross-point switch interface and router interface. Each of the remaining eight FPGAs, along with the cross-point switch, is equipped with four parallel memory channels interfaced to .5 gigabytes of double data rate DRAM per channel, i.e. 2 gigabytes of RAM per FPGA. This eleven-chip system features 18 gigabytes of RAM, configured with 36 64-bit parallel memory channels, yielding aggregate memory bandwidth of greater than 50 gigabytes per second per board. FPGAs are organized in groups of four with 50 I/O connections between each FPGA and the other three FPGAs in the quad. A total of 560 external I/O lines are available for communicating with other boards or digital systems. In a larger system boards are also organized in groups of four with one interconnect board in a similar fractal-like hierarchy.

With this architecture the following communication bandwidths are achievable:

Bus controller to quads:	20 GB/s	I/O board to board:	22.5 GB/s
Cross-point switch to quads:	10 GB/s	FPGA to FPGA:	7.5 GB/s
Router to quads:	30 GB/s	FPGA to FPGA inter-quad:	10.5 GB/s
Inter-quad:	12 GB/s	Aggregate bandwidth:	84 GB/s

These results are highly promising. However, the realization of the full promise of general purpose, dynamic reconfigurable computing will require answers to several key questions. A fundamental question concerns the evolution of current RC prototypes – which have not yet been used in specific applications – into general-purpose high-end supercomputers. Early library development in Viva has concentrated on basic functions and primitive mathematics objects. Development is now producing intermediate reusable building blocks such as high level mathematics, integration, differential equation solving, linear algebra, memory, I/O, control, logic, structure, signal processing and image processing library objects. As it continues to evolve, the library of reusable objects will expand to include basic applications, then complex applications. Objects at each level become building blocks for objects at the next level. This resembles the programming style now used by programmers in the ASCI stockpile stewardship program at the nation's national laboratories.

In time the library set will become comprehensive enough to represent a truly general purpose, "point and click", high-end supercomputer with a wide range of applications, accessible from an object library by the click of a mouse on Viva's graphical interface. These applications will be "tuned" by the user on-screen – for example, by inserting data set and information rate parameters into polymorphic operators within the application. The question is: What will drive the evolution of the library set to the general-purpose level, or at least to a level which addresses national security needs, and how will this take place?

Other questions include: What RC applications will become viable and when? What RC programming model can best exploit the native advantages of RC and still leverage the features of other desirable programming models? What methods can be developed to port legacy applications written in text-based languages to RC platforms to run at high-end speeds? For "classic problems" are there alternative algorithms or problem formulations that better fit RC architecture and outperform time-to-solution requirements? How do RC systems scale? Will available bandwidth keep pace with peak performance of RC devices?

As these and other questions are answered – and they must be answered – it will become apparent to what degree dynamically reconfigurable supercomputing is a significant breakthrough that will help the U.S. maintain its decisive edge in innovation essential to retain dominance in technologies vital to our national security.

Thus far, the extraordinary programming flexibility, hardware adaptability and execution speeds of dynamically reconfigurable supercomputing demonstrate the potential to satisfy the diversity of technology needs identified by the *Report on High Performance Computing for the National Security Community* as critical to national security: comprehensive aerospace vehicle design, signals intelligence, operational weather and ocean forecasting, stealthy ship design,

nuclear weapons stockpile stewardship, signal and image processing, Army future combat systems, electromagnetic weapons development, geospatial intelligence and threat weapon systems characteristics.

This diversity of needs is impossible to satisfy in today's server dominated market, which requires major supercomputer vendors to trade off those needs in favor of market demands. Tradeoffs include programming flexibility in favor of custom devices, a low ratio of computation to communication in favor of a high ratio of computation to communication and optimized global communications in favor of applications that can leverage data caches. If on-the-fly programming flexibility and high-end performance are both inherent in RC technology, as many believe they are, the national security community's current and future HEC needs will no longer be held hostage to such tradeoffs.

REFERENCES

1. Storaasli, Olaf O., Robert C. Singleterry, and Samuel Brown, "Scientific Computations on a NASA Reconfigurable Hypercomputer." Presented at 5th Military and Aerospace Programmable Logic Devices (MAPLD) Conference (paper in preparation). September 10-12, 2002.

2. Snavely, Allan E., Matt Lyons and Xin Zhao (UCSD), "Scientific Computing with FPGAs." Poster presented at SC2001. Details at http://www.sc2001.org/postersdetail.html#42.

3. Singleterry, R., J. Sobieski and Samuel Brown, "Field-Programmable Gate Array Computer in Structural Analysis: An Initial Exploration." Presented at AIAA 43rd SDM Conference, April 22-25, 2002.

4. Fithian, William S., Samuel Brown, Robert C. Singleterry and Olaf O. Storaasli, "Iterative Matrix Equation Solver for a Reconfigurable FPGA Based Hypercomputer." Paper published by NASA's Langley Research Center, September 5, 2002.