LA-UR -82-2326

Conf-820908--2

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36

LA-UR--82-2326

DE82 021769

MASTER

TITLE: IMPLEMENTATION OF A FULLY RECONFIGURABLE MULTIMICROPROCESSOR

AUTHOR(S): Vito A. Trujillo

SUBMITTED TO: 1982 IEEE International Conference on Circuits and Computers, New York, September 29-October 1, 1982.

Los Alamos Los Alamos National Laboratory
Los Alamos, New Mexico 87545

FORM NO. 836 R4
ST. NO. 2629 5/81

IMPLEMENTATION OF A FULLY RECONFIGURABLE MULTIMICROPROCESSOR

Vito A. Trujillo

Computing Division
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

ABSTRACT

The fully reconfigurable multimicroprocessor is an experimental configuration designed specifically as a research tool for implementing and evaluating parallel-processing algorithms on various multiprocessor architectures. Basically, the system is a shared-memory MIMD (multiple instruction-multiple data stream) machine that supports reconfiguration between processor and memory nodes to permit experimentation on architectures sharing common memory, networks of processors with only local memory, etc. This experimental computer system is currently under development within the Computing Division at the Los Alamos National Laboratory.

Summary

This paper presents the hardware architecture of an experimental multiprocessor system that incorporates a reconfigurable array of commercially available microprocessing and memory elements. The system is designed specifically as a research tool for evaluating parallel-processing algorithms on various multiprocessor architectures to predict which problems are best implemented on specific architectures. Consequently, the principal design objective is to provide a multiprocessor with fully reconfigurable processor-to-memory and processor-to-processor interconnections in order to allow direct comparison of algorithms for a wide range of multiprocessor architectures.

Basically, the system is a tightly coupled, shared-memory MIMD machine [1-3] that supports reconfiguration between processor and memory nodes to permit experimentation with common memory architectures and with various processor network structures, such as rings, trees, and stars. The proper combination of processors and memories can be selected based on the parallel-processing objective.

As illustrated in Figure 1, the system consists of numerous processor and memory nodes that are directly interconnected using multiple processor-to-memory buses and multiported global memory nodes. The multiple bus/multiported memory arrangement functionally implements a full crossbar switch between the processor and memory nodes [4]. This multiple-bus architecture allows processor-to-processor communications to occur concurrently with processor execution from either local memory or global memory. Reconfiguration between processor and global memory nodes is accomplished through a memory mapping facility included within each processor node.

Three types of processor nodes are included within the system: (1) system control processor (Pc), (2) general floating-point processors (Pi), and (3) dedicated data transfer processors (Pxi). The system control processor performs system initialization (downloading of global memory, configuration control, etc.), initiation of parallel-processing applications code, performance measurements, and memory error processing and incorporates an interprocessor interruption facility. In addition, because the multiprocessor is strictly an execution environment, the system control processor provides communication with an external local area network that includes development workstations. Each general floating-point processor includes Intel iAPX 86/87 microprocessing elements [5], 48k bytes of local dedicated ROM/RAM, real-time interrupt facility, and memory mapping logic that allows sixty-one 16k-byte memory segments to be permanently and/or dynamically allocated within the system global memory. Each data transfer processor is a high-speed controller specifically designed for implementing processor-to-processor communications by performing data movement between global memory segments.

The system global memory consists of multiple memory nodes, each having a 256k-byte RAM array accessible from the system control processor, and a multiported memory controller. The port for the system control processor supports downloading and memory error reporting functions. The multiported memory controller includes interface logic for 20 ports, memory arbitration logic that implements a last-granted-lowest-priority algorithm, and a high-speed memory access controller. Memory mapping logic within each processor node allows each memory node segment to be allocated as either private or public memory for each processor node.

The processor-memory interconnection is accomplished with memory mapping logic at each processor node, a multiported memory controller at each glob-

al memory node, and a multiple bus interconnection backplane that allows an orthogonal arrangement of processor and global memory boards. As illustrated in Figure 2, an orthogonal packaging scheme uses minimal bus lengths in providing complete physical interconnection between the processor and global memory nodes. Basically, the processor-memory interconnection provides fully reconfigurable processor-to-memory connections, resolves access arbitration when multiple processors are simultaneously accessing a common global memory node, and supports mutual exclusion to shared memory. Control of shared memory is accomplished through an extension of the lock/unlock mechanism available with the iAPX 86/87 microprocessor [5].

Processor-to-processor communication is implemented indirectly through the processor-memory interconnection by specialized data transfer processors that perform data movement between global memory nodes. Each data transfer processor includes memory mapping logic similar to the general floating-point processors; consequently, these nodes can access any segment within system global memory. In addition, the data transfer processor nodes include high-speed control, buffer, and translation logic that permit both contiguous and noncontiguous memory block transfers. The data transfer processors are controlled by linked structures within global memory and include maskable interrupt capability indirectly through the system-control processor. Functionally, the data transfer processors can be visualized as multiple intelligent buses for interprocessor communications.

Currently, the system accommodates a single-system control processor, 20 processor nodes that can include either general floating-point processors or data transfer processors, and 32 global memory nodes. However, a typical maximum configuration consists of 16 general floating-point processors and 2 data transfer processors, which are sufficient for handling 16 iAPX 86/87 interprocessor communications.

References

1. P. H. Enslow, Jr., "Multiprocessor Organization - A Survey," ACM Computing Surveys (March 1977), Vol. 9, pp. 103-129.

2. S. H. Fuller, J. K. Ousterhoot, et al., "Multi-Microprocessors: An Overview and Working Example," Proc. IEEE (February 1978), Vol. 66, pp. 216-228.

3. M. Satyanarayanan, Multiprocessors: A Comparative Study, Prentice-Hall, Inc., Englewood Cliffs, N.J., (1980).

4. J. H. Patel, "Processor-Memory Interconnections for Multiprocessors," Proc. 6th Annual Symposium on Computer Architecture (April 1979), New York, N. Y., pp. 168-177.

5. Intel iAPX 86,88 User's Manual (August 1981), Intel Corporation, Santa Clara, CA.

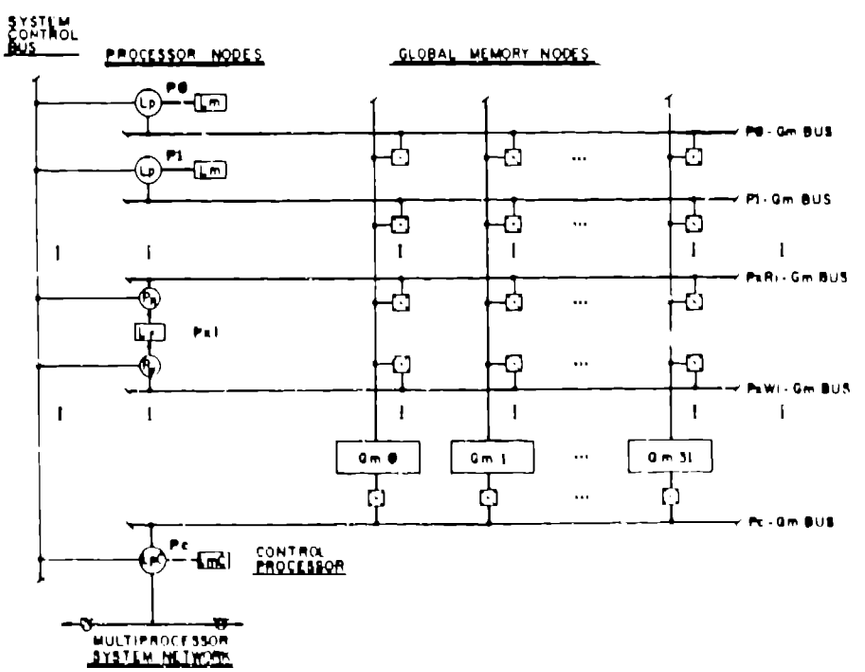

Figure 1. System architecture.

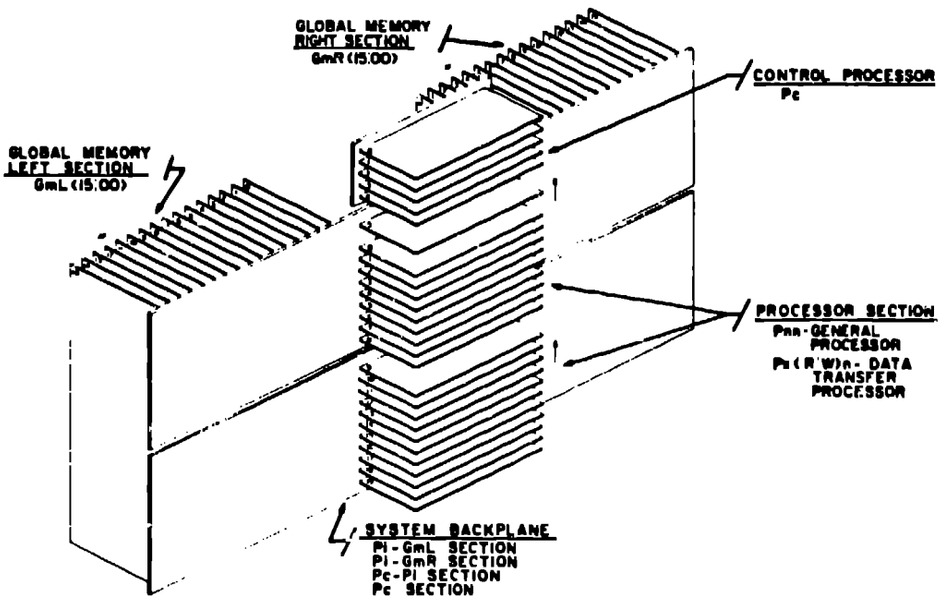

Fig. 2. Orthogonal packaging diagram.