Abstract

The Bath Concurrent LISP machine is a general purpose multiprocessor designed for research in symbolic computation and expert systems. The system is particularly suited to tree structured computations with dynamic data requirements and complex interactions. It accepts conventional LISP programs, performs a data flow analysis on them, and creates a compiled module in which available concurrency is exploited. This paper presents an overview of the system hardware and software and the directions in which research is proceeding.

Introduction

The Bath Multiprocessor was built by the Electrical Engineering and Mathematics Schools at the University of Bath starting in the spring of 1982. It consists of six Motorola 68000 microprocessors connected in a topology to facilitate dynamic task scheduling and minimize communication overhead. Its intended tasks are processing of quadtrees for a computer graphics system, and general high-speed computations in symbolic computation and expert systems. This report overviews the progress we have made towards building an operational concurrent processing environment.

Hardware

The initial hardware configuration is six individual Motorola 68000 computer systems, each with 256K of 500 nanosecond memory, up to 64K bytes of ROM, and a 16 bit data path. For machines with no peripherals there is room for up to 1.5 megabytes of high speed store, soon to be expanded to 3 megabytes for a total possible configuration of
over 16 megabytes. One machine has a DMA floppy disk controller with access to 2.4 megabytes of transportable storage. Through high speed parallel and serial interfaces, the other 5 machines are connected to a 26 megabyte winchester disk operated by an intelligent disk controller. The department also possesses a standalone 68000 system with 512k bytes of RAM, 15 megabytes of disk storage, network connections, floppy disks and so on. By connection through the intelligent disk server we were able to bring up the first versions of the operating system and LISP. The intelligent disk controller also runs LISP and served as a test bed for much of the language processor and data flow analysis software. The machine is also connected to a bit mapped color graphics display [1], a 68000 based quad tree display [13], and soon to the departmental token ring network [14]. The current configuration is the four processor network shown in Figure 1.

Interprocessor communication is conducted through memory windows: 2K byte blocks of dual ported memory shared by adjacent processors. There are three such windows per processor. The effective throughput under program control is somewhat under 1 megabyte per second per processor for a total machine capacity of 6 megabytes per second.

The numbering scheme for processors places the storage for each set of three windows in even numbered processors and an empty interface in odd. The four processor configuration of Figure 1 uses only two memory windows per processor while the six processor configuration in Figure 2 utilizes all three [16].

The properties of this network have been investigated by simulation [1]. The results indicate is that the valency of a node (the number of outward edges) influences the scheduling and duration of a computation. Too low a valence inhibits potential concurrency and to high causes more overhead in task scheduling and storage. The simulation indicates that a valence of 3 is optimal for many tasks.

Supporting System

A number of programs support the operation of the concurrent machine. Many of these were developed and debugged on the standalone processor before the multiprocessor was completed. Each machine runs a complete copy of the TRIPPOS operating system [11] which occupies about 90k bytes of available storage. This supports a hierarchical filing system, a multitasking supervisor, a resident disassembler, and debugging program. These have proven invaluable during the initial implementation effort. As more of the system becomes operational, subsidiary processors will no longer need large parts of the operating system to the point where well over 200K bytes of storage should be available for user programs.