Nuclear Instruments and Methods in Physics Research A262 (1987) 366–370
North-Holland, Amsterdam

A MULTIPROCESSOR SYSTEM FOR RAPID DATA ANALYSIS

J. ELSON, O. EL-GHAZZAWY and D.G. SARANTITES

Department of Chemistry, Washington University, St. Louis, MO 63130, USA

Received 5 May 1987

A multiprocessor system designed for rapid analysis of data is described and its performance is evaluated in a complex histogramming application. The system consists of several general purpose processors coupled to the Multibus with a shared memory that is used for intercommunication between the processors and a host VAX-11/780 computer.

1. Introduction

In recent years a constant increase in complexity of the experiments in nuclear physics has been observed. For nuclear structure and nuclear reaction mechanism studies large arrays of detectors are being used, which provide experimental data on an event-by-event basis with several hundreds of correlated parameters. Examples of such detector arrays include the 4π γ-ray multidetector spectrometers such as the spin spectrometer at HHIRF [1,2], the crystal ball at Heidelberg [3], the anti-Compton Ge arrays at HHIRF, Daresbury [4], Berkeley [5], and others. Other multidetector devices for charged particle counting such as the Dwarf Ball [6] are under construction. The analysis of such multiparameter data usually requires extensive histogramming to be done with considerable manipulation of the parameters of each event, which may include event-by-event energy calibrations, linearizations, Doppler shift corrections, reaction kinematics, etc. These are extremely CPU time consuming operations. It is therefore highly desirable to develop inexpensive computer systems that are capable of handling high CPU loads without impeding the operation of time sharing systems in multiuser environments.

Since the data consist of self-contained events of correlated addresses, it is possible to process the events using a large number of low cost processors that are operating independently on different groups (blocks) of data appropriately supplied by a host computer. In this paper we present the architecture and performance of a system based on the National Semiconductor 32016 boards operating in a Multibus environment that is interfaced to a host VAX 11/780 computer [7]. A description of the package for data analysis in the host computer is given briefly in sect. 2. The description of the multiprocessor system and its components is given in sect. 3. The basic operation and performance results are given in sects. 4 and 5.

0168-9002/87/$03.50

2. The data analysis system

In order to provide the user with a flexible data processing system which is suitable for efficient use of a multiprocessor system we have developed a scanning package that handles all the input–output operations according to a simple instruction list provided by the user.

For each specific task the user must supply a preprocess, a block-process and a postprocess routine. The function of the preprocess routine is to read files and set up the configuration and all the conditions for the tape scanning or histogramming as required. In the present system the host computer is a VAX-11/780 operating under VMS. The preprocess subroutine is written in Fortran. The host handles the tape I/O using double buffering and supplies blocks of data (8192 bytes) to the block-process routine written in Pascal, which in turn operates on the events for analysis or histogramming. It is this block-process routine which is executed in each processor of the multiprocessor system. Finally, a postprocess routine written in Fortran is called for a variety of output information, such as scalers, other derived tabular information, data arrays defined by the user in addition to four groups of one-dimensional and four groups of two-dimensional sets of spectra that are internally defined by the data analysis package, etc.

3. Description of the multiprocessor system

The multiprocessor system consists of a variable number (up to seven) of National Semiconductor (NS)

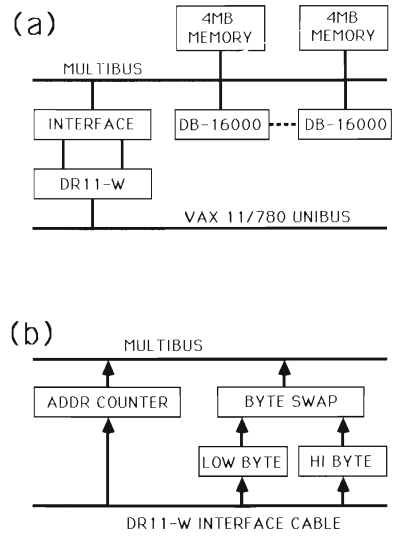

Fig. 1. Schematic diagram of multiprocessor system. The coupling of the Unibus and Multibus is made through a DR11-W and a custom-made interface as shown in (a). The custom interface includes the address counter for the control of the system and the byte swapping section done during the data transfer (b).

[8] 32016 processor boards. Each processing unit (board) contains an NS-32016 general purpose processor, an NS-32081 Floating Point processor, and 128 Kbytes of local memory. The boards are interfaced to the Multibus. All processors share an 8 Mbyte common memory on the Multibus (R). A small part of the common memory is used for intercommunication between the multiprocessor system and the host computer, a DEC VAX-11/780 [7]. This intercommunication is provided through an MDB Systems DR11-W direct memory access (DMA) interface and a custom made Multibus–Unibus DMA interface which is totally controlled by the host as shown schematically in fig. 1a.

3.1. The DB-16000 board

The National Semiconductor NS-32016 processor is a 32 bit computer capable of speeds similar to the VAX-11/750. It has a very flexible instruction set with substantial similarity to the VAX, and is designed to permit efficient code to be generated by high-level language compilers. National Semiconductor provides an assembler and a Pascal compiler which runs on the VAX and generates code for the NS-32016. The floating point processor is fast, so that a 32 bit multiply is performed in under 5 μs. The NS-32016 has an instruction prefetch queue which keeps the next several instructions ready for execution by the processor. The original DB-16000 board from National Semiconductor contains 128 Kbytes of random access memory (RAM), the NS-32016, and the NS-32201 timing chip. The NS-32081 floating point unit (FPU) can be plugged into the available socket. Additional sockets for the NS-32202 interrupt controller and the NS-32082 memory management unit are provided, but are not needed in the present application. There are also serial and parallel ports. The parallel ports are used to operate status lights on each board. The serial port can be used by a debug program. It connects an NS supplied debugging program on the VAX to the NS-32016 monitor program for communication when a NS-32016 program is being interactively debugged. This mode allows interactive halting of the program at specific line numbers, examination of variables, and tracing of the program flow. During normal operation the serial port is not used.

The processor board required substantial modification (the addition of two ICs) to make it comply with Multibus multimaster protocol. New boards (DB-32016) have been manufactured by NS to replace the old DB-16000 boards, but these have not been incorporated into this system.

3.2. The MDB DR11-W board

The MDB DR11-W board is a 16-bit parallel-DMA interface for the DEC Unibus. It transfers data between the VAX and the Multibus memory at a rate of 200 000 16-bit words per second. It is designed to use the standard DEC device driver for DEC's version of this board, but for efficiency we rewrote the driver to accomplish all operations required for a data transfer in one request (QIO). This also required a slight wiring modification to the DR11-W to bring out one internal signal.

3.3. The Multibus interface

We have designed a Multibus interface to connect the multiprocessor system to the VAX DMA interface DR11-W. This interface handles all Multibus protocol, but its special feature is that it is controlled completely from the DR11-W, so that the VAX can perform operations on the Multibus with no assistance from the NS-32016 processors. This is necessary for initially loading the program, but it also means that there is no need for the NS-32016 processors to monitor the interface and keep it ready to receive new data buffers. This also means that there is no need to break the symmetry of the NS-32016 processors such that one has a different or an additional task.

The Multibus interface permits the VAX to select the address in Multibus memory at which the data transfer will begin. In addition this unit performs byte-swapping during the data transfer as shown schematically in fig. 1b. Byte-swapping takes 4.7 μs per word for the VAX processor to perform, but is done in zero time by the Unibus–Multibus interface.

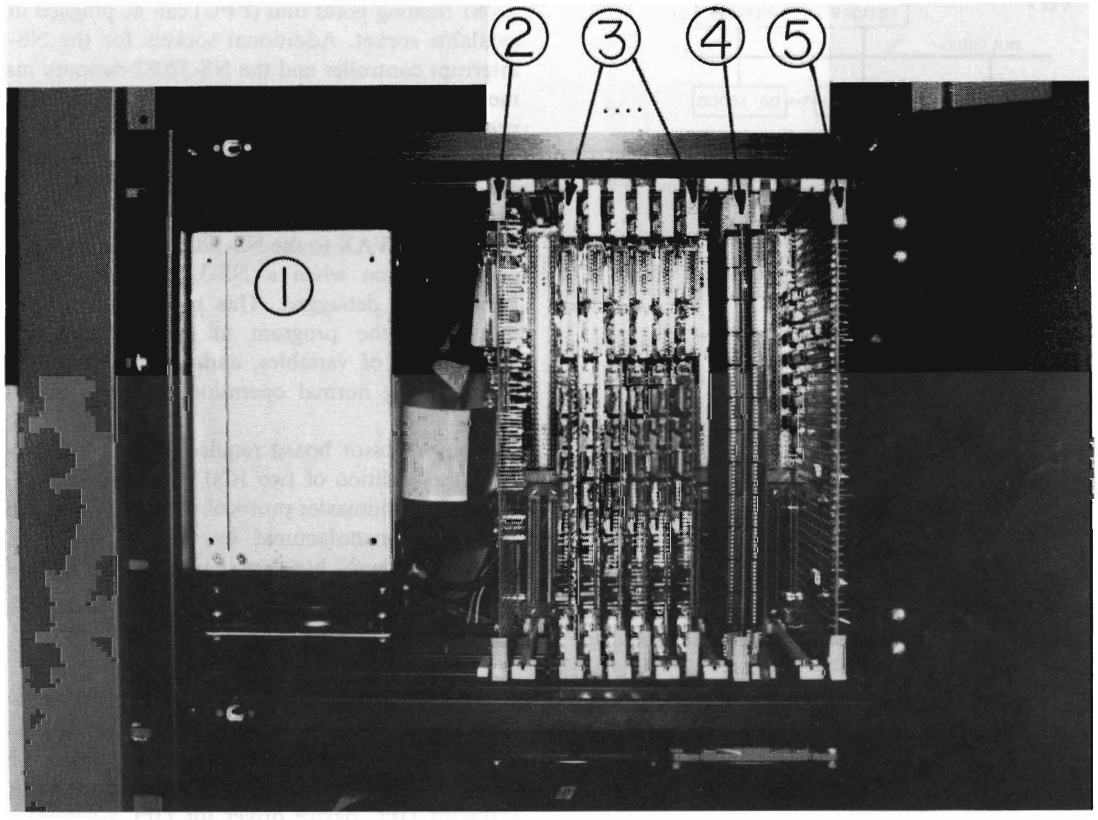

Fig. 2. Photograph of the multiprocessor system. The power supply is seen on the left (1). The first board from the left is the interface (2). The following six boards are the DB-16000 boards (3), followed by the two memory boards (4). The last board to the right is used for LED display of the function of the system (5).

3.4. Multibus card cage

The Multibus card cage is a 15-slot [9] card cage, with P2 connectors and parallel priority resolution. Hand wiring of the P2 connector allows the use of the full 24 bit (16 Mbyte) address space of the Multibus, and also synchronizes the clocks of all the CPU's. Parallel priority resolution is an optional feature of the Multibus, which permits two or more bus masters to operate in one card cage. This cage was modified for the large power demands of the CPU boards by adding external bus bars to the circuit board power and ground busses on the backplane. A 100 A, 5 V switching power supply was installed in the unused area of the cage, to the side of the connector area. The processor boards consume about 6 A each and the Multibus interface draws 3 A of current. Fig. 2 shows a photograph of the multiprocessor system.

3.5. Memory

The initial system was assembled and tested with a single 1 Mbyte board. The system was then upgraded to 8 Mbytes by replacing the 64K memory chips with 256K ones and adding a second 4 Mb memory board to the system. This allows up to 4 megachannel histogramming to be done in core memory.

4. Basic operation

As it was discussed in sect. 2 only the block-process routine is executed in the multiprocessor system. The preprocess routine is first executed by the host computer. It reads setup information from VAX files and builds tables of parameters, sizes, counts and similar information which gives the block-process routine the necessary parameters to operate correctly on the data stream. This allows the block-process routine to have considerable generality without any need to rewrite it for every application that uses the same type of operations.

The block-process routine in each different application is written in Pascal, with some special parts in assembly language. These routines are compiled (or assembled) and linked on the VAX with the National

Semiconductor NSX-Pascal package. Then several programs modify these data to turn them into a "core image" file. The core image file is loaded into the shared multibus memory by the VAX. A modified version of the standard debugging monitor checks the value in the first four bytes of the Multibus memory, and if it matches a given value, then the program is copied by all processors from shared to local memory. These programs then are started, and they wait for another certain value. The VAX at this point writes the setup values into the Multibus memory, and sets the first 4 bytes to the "constant" code word. When this value is seen, the program in each processor copies the setup information from Multibus memory to local memory, and then begins to scan semaphore words in shared memory for a buffer that has data to be processed. When if finds a 1 in one of these locations it checks again under bus interlock, and sets it to a 2, indicating that the buffer is now in use. The system currently uses 32 8 Kbyte buffers. When the data in the buffer have been processed the semaphore is set to 0, which indicates to the VAX that fresh data can be put into this buffer.

5. Performance results

We have used a general gating and histogramming program called GATE to evaluate the performance of the multiprocessor system. The reference for comparison is the VAX-11/780 operated in a single user mode, while the multiprocessor was operated with the host in normal multiuser time sharing environment. An elaborate gating and histogramming task, with a large fraction of integer additions and multiplications was chosen for comparison. The same program was run on the VAX with and without the array boundary checks and on the Pascal version of the program for the multiprocessor with and without the boundary checks. Briefly, we find that the NS-32016 boards when running at 8 MHz perform at about $\frac{1}{3}$ of the speed of the VAX-11/780.

The comparison is illustrated in detail in fig. 3. The ordinate gives the number of processed events per second as a function of the number of processor units inserted in the system. It is seen that with 7 processors an increase in performance by a factor of 2.5 over the VAX is obtained when the checks are activated in both systems. An increase in performance by a factor of 2.0 over the VAX was achieved when the checks were removed from either system.

It is interesting to note that no indication of saturation in performance was observed in going from one to seven processors. Furthermore, it should be noted that the multiprocessor performance refers to clock time as opposed to CPU single user time for the VAX-11/780.

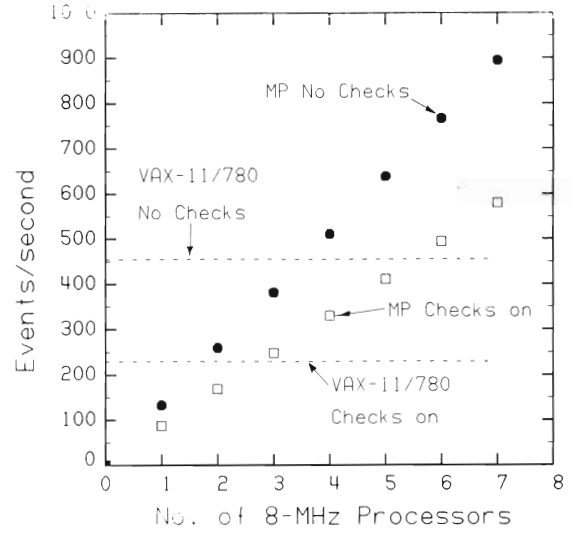

Fig. 3. Comparison of the performance of the multiprocessor system with the VAX-11/780 for a histogramming task. The performance is given in events processed per second as a function of the number of processor boards inserted into the system. The open squares and closed circles correspond to having boundary checks on and off on the MP system. The throughput is essentially linear for up to 7 processors in the system. For comparison horizontal lines are drawn for the VAX-11/780 performance with the checks on and off. It is seen that when 7 processors are used, a factor of 2.0 improvement is obtained over the VAX-11/780 without the checks and a factor of ~ 2.5 when the checks are on.

For the majority of the tasks anticipated in gating and histogramming applications the VAX can supply the multiprocessor system with data with only a minimal 5–10% load, thus making the VAX available for other tasks, while the multiprocessor is being used. For typical prime time usage with a 50% on the average available time per single histogramming task it is clear that a factor of 4 in throughput can be achieved. A further enhancement of about 30% is obtained over the VAX in applications where byte flipping is required, giving a typical throughput increase for this case of a factor of ~ 5 over the VAX-11/780.

In conclusion we have developed a powerful and inexpensive data analysis and histogramming system based on the multiprocessor concept. This system increases considerably the throughput of data analysis particularly under multiuser environments.

References

[1] D.G. Sarantites, R. Lovett and R. Woodward, Nucl. Instr. and Meth. 171 (1980) 503.

[2] M. Jääskeläinen, D.G. Sarantites, R. Woodward, F.A. Dilmanian, J.T. Hood, R. Jääskeläinen, D.C. Hensley, M.L. Halbert and J.H. Barker, Nucl. Instr. and Meth. 204 (1983) 385.