Journal of Zhejiang University SCIENCE A
ISSN 1009-3095 (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Parallel programming characteristics of a DSP-based parallel system*

GAO Shu†, GUO Qing-ping

(*School of Computer Science, Wuhan University of Technology, Wuhan 430063, China*)

†E-mail: gshu418@163.com

Received Jan. 23, 2006; revision accepted July 18, 2006

Abstract: This paper firstly introduces the structure and working principle of DSP-based parallel system, parallel accelerating board and SHARC DSP chip. Then it pays attention to investigating the system's programming characteristics, especially the mode of communication, discussing how to design parallel algorithms and presenting a domain-decomposition-based complete multi-grid parallel algorithm with virtual boundary forecast (VBF) to solve a lot of large-scale and complicated heat problems. In the end, Mandelbrot Set and a non-linear heat transfer equation of ceramic/metal composite material are taken as examples to illustrate the implementation of the proposed algorithm. The results showed that the solutions are highly efficient and have linear speedup.

Key words: Parallel algorithm, Multi-grid, Domain decomposition, Virtual boundary forecast (VBF), DSP-based parallel system
doi:10.1631/jzus.2006.A1669 **Document code:** A **CLC number:** TP39

INTRODUCTION

DSP-based parallel system is a kind of high-performance parallel computer which uses parallel accelerating boards based on SHARC DSP chip (Kuo and Gan, 2005) as plug-in boards and inserts them into the military reinforcing computer, military fault-tolerant computer, even the personal computer. The parallel accelerating board based on SHARC DSP is a high-performance parallel processing board-level product developed by the Wuhan Digital Engineering Institute and is suitable for various mainframe computer environments (including Multibus, ISA, PCI and other standard industrial buses). As it can be plugged into different kinds of computers, and although its volume is small, it is light weight, its power consumption is low, its computation speed is fast, it can be used in the high performance computer mounted on vehicles, aircraft, ships and crafts, satellites and missiles, and in the ground or underground operation under adverse circumstances. Therefore, we investigated the programming method and characteristics of this parallel system, and put forward a domain-decomposition-based complete multi-grid parallel algorithm with virtual boundary forecast (VBF), which serves to solve large-scale and complicated heat problems, for example, heat stress analysis for ceramic/metal composition material, laser quenching, and so on. At the same time, we can also apply the parallel-optimized approach to solve the famous Mandelbrot Set and make its speedup linear. The results showed that the approach provides a high-efficiency and feasible parallel scheme for problems in fields such as national defense scientific research, intelligence process and analysis, simulated training and simulation, real-time industrial control.

The remainder of this paper is organized as follows. In Section 2, we briefly introduce the structure of the DSP-based parallel system. Section 3 investigates the programming characteristics of the parallel system. Section 4 describes a domain-decomposition-

* Project (No. 60173046) supported by the National Natural Science Foundation of China

based complete multi-grid parallel algorithm with VBF and gives two examples to illustrate the feasibility of the proposed algorithm. Section 5 gives a case study. Section 6 discusses conclusion and future work.

STRUCTURE OF THE PARALLEL SYSTEM BASED ON DSP

Structure of the system

The parallel system uses the parallel accelerating boards based on SHARC DSP as plug-in board and connects them to the host computer through various buses, such as Multibus, ISA. Its block diagram can be simplified as Fig.1.

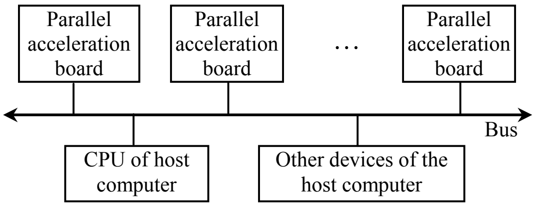
Parallel acceleration board
Parallel acceleration board
...
Parallel acceleration board
Bus
CPU of host computer
Other devices of the host computer

Fig.1 Simplified structure diagram of the DSP-based parallel system

Structure of the parallel accelerating board based on SHARC DSP

There are four Analog Devices ADSP-2106*x* SHARC DSP chips on the parallel accelerating board, which are connected to form a Multiprocessing cluster through a shared bus. It can achieve computing power of up to 480 MFLOPS and also has a 1 M×32 bits zero waiting shared SRAM that is used as external shared memory. At the same time, in order to support system expansion, the board provides 12 external link ports, each of which is 40 MB/s, and 2 serial ports of TDM mode, each of which is 40 Mb/s. There is a JTAG interface which is used by the EZ-ICE simulator and a 512 k×8 bits Flash ROM which supports the modular electrifying bootstrap (Analog Devices, Inc., 1995). Its block diagram is shown in Fig.2.

SHARC DSP chip

SHARC is the abbreviation of Super Harvard Architecture Computer. It includes the ADSP-2106*x* series chips, which are designed for high performance signal processor for communications, Graphics and Imaging Applications. The chips in this series have similar architecture. The differences among them are the capacity of the on-chip memory and the ability of the communication. The parallel accelerating board uses ADSP-21062, which is composed of the 2 Mb dual-ported on-chip memory, I/O processor and the dedicated function core processor whose speed is up to 40 MIPS. There are four chips on a board, which are connected to form an integrated system through

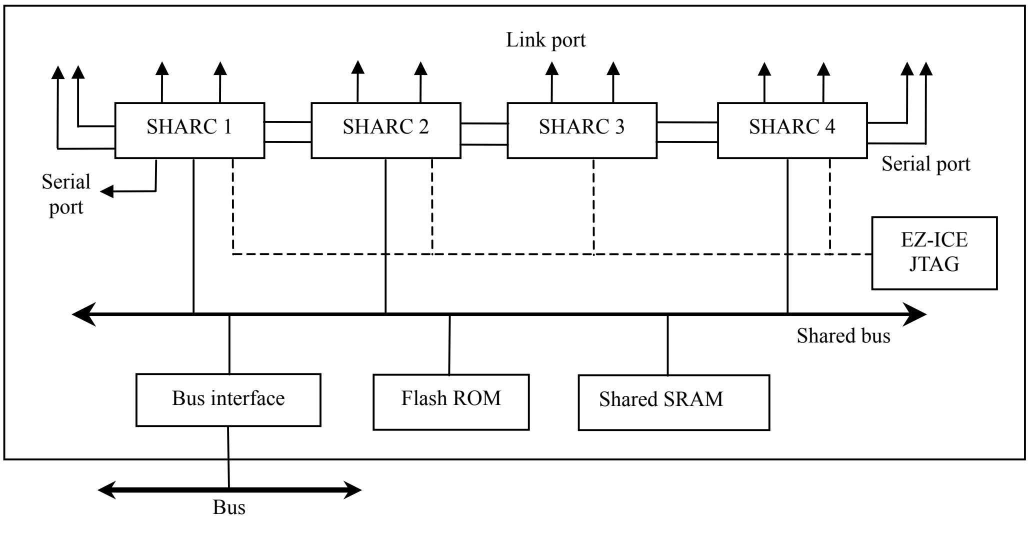

Fig.2 Block diagram of the parallel accelerating board based on SHARC DSP

four independent buses, namely DM bus, PM bus, I/O bus, EP bus. The dual-ported on-chip memory of the chip is organized as two blocks of 1 Mb each, which can be configured for different combinations of code and data storage. Each memory block is dual-ported for single-cycle, independently accessed by the core processor and I/O processor or DMA controller. In addition, the I/O processor includes six 4-bit link ports, each of which has its own double-buffered input and output registers, two serial ports, a DMA controller. The ADSP-21062 chip also contains an external port that provides the processor's interface to off-chip memory and peripherals. The chips can form a multiprocessing cluster by the external port's connection to the external system bus. The ADSP-21062 offers powerful features tailored to multi-processor DSP systems. The unified address space allows direct core processor accesses of each ADSP-21062's internal memory. Distributed bus arbitration logic is also included on-chip for simple connection of systems containing up to six ADSP-21062's and a host processor. Bus arbitration is selectable as either fixed or rotating priority. Bus lock allows indivisible read-modify-write sequences for semaphores. A vector interrupt is provided for core processor commands. Maximum throughput for processor data transfer is 240 MB/s over the link ports or external port. Broadcast wires allow simultaneous transmission of data to all ADSP-21062's and can be used to implement reflective semaphores (Analog Devices, Inc., 1996).

RESEARCH ON THE PROGRAMMING CHARACTERISTICS OF THE PARALLEL SYSTEM BASED ON DSP

In order to design a parallel program of the parallel system, the following problems should be considered:

(1) How to distribute the data to every processor?

(2) How to balance the amount of computation among the processors?

(3) How to communicate among the processors?

In view of the system characteristics, the algorithm should adopt the master-slave mode. The property of the mode is that it has a main-control program called master process and several slave processes (Li and Jiang, 1992). The master runs on the host computer and is responsible for generating, initializing the slaves, and distributing the computing task, receiving and displaying the result, with the slaves running on the SHARCs and performing the computation. They may or may not communicate with each other (Makowski *et al.*, 2004).

As far as the communication is concerned, there are several modes as follows.

(1) The mode of link port. It is an effective mode to realize the communication among the SHARCs. Each SHARC has six link ports which can implement the point-to-point communication with the other five SHARCs simultaneously. It uses the on-chip DMA to send and receive the data at speed of 80 MB/s. However, only 4 bits of data can be put into the data line at every time.

(2) The mode of external port. It is more flexible, which can only set the sender or receiver. Moreover, it can communicate between the SHARC and host computer, and between the SHARCs as well. Its speed is up to 240 MB/s because all of the external ports share the bus. However, its limitation is the communication has to be completely serial. At the same time, it has data-packing ability, namely, it can pack the 32/48 word data on the chip into 16/32 word ones in order to be accessed by the host computer or the peripherals.

(3) The mode of direct access to bus. It means the holder of the bus can access the multiprocessor memory space and external memory space of the other chips. For example, general registers MSGR0-MSGR7 can be used to transfer data among the SHARCs because the address of the registers may be mapped into the IOP space and are known and fixed. Others can get them if only both SHARCs want to communicate through the used register and if one of them puts the data into the register. Similarly, by means of the mode, SHARC or host computer can also write directly the address of the interrupt service routine to vectored interrupt register on the other SHARCs in order to make the vectored interrupt to be generated on those SHARCs, and execute the routine to control those SHARCs (Gan and Kuo, 2006).

In consideration of the above characteristics, we investigated the non-numeric computation and large-scale numeric computation problems respectively. For the former, we took the famous Mandelbrot Set as an example. We optimized it by means of the method of distributing coordinates in the plane alternately to maintain the balancing of load on every SHARC. For the later, we took the heat conduction model of the ceramic/metal composition material as an example to investigate the high-efficiency parallel algorithm for large-scale and complicated heat problems.

PARALLEL ALGORITHM ON THE DSP-BASED PARALLEL SYSTEM

Parallel algorithm for the Mandelbrot Set

1. Mathematical model

Mandelbrot Set is the set M which is composed of the following complex number:

$$M=\{C:|M_n(c)|<\infty, \forall n\in N\},$$

where

$$M_0(c)=0\,,\ M_{n+1}(c)=M_n^2(c)+c.$$

It can also be expressed as follows:

$$\text{if } \exists\, n:|M_n(c)|>2,\ \text{then } c\notin M.$$

The set can be drawn on the display because a complex number can be represented by the coordinate of a point in the complex plane. The method is that if a complex number is not in the set, the color value of the corresponding point is the loop number in the program which decides the number is not in the set, while if the loop number is more than a fixed value and the module of the complex number is still less than 2, the complex number is considered to be in the set and the color value of the point is set black (i.e. zero). It can be proved that all of the complex numbers represented by the points in the square are in the set if the complex numbers represented by the points at the edge of a square are all in the set (Phil *et al.*, 2004).

2. Implementation of the Mandelbrot Set

(1) Design of the algorithm

In order to draw the picture of Mandelbrot Set, it is necessary to calculate the color value of every point in the complex plane. As we all know, the calculation is independent. Therefore, the master-slave algorithm is available for this problem.

With respect to the communication mode, the data transmission between the SHARC and host computer adopts the mode of external port. And the mode of the direct access to bus is also used to set up simple handshaking control using the general register MSRG6.

The parallel program consists of the “master.c” and “newslave.c”, in which the “master.c” is the main process and runs on the host computer, and the “newslave.c” is the sub-process and runs on each SHARC. The main process is responsible for loading the “newslave.ldr” to every SHARC, initiating the data and sending them to each sub-process by means of the function *sendsharc*, receiving the calculation results from every sub-process by means of the function *receivesharc* and drawing the picture. Meanwhile, each sub-process receives the task by function *receive*, calculates the color value of points and sends the results to the main process by function *send*, and gets the new task from it.

(2) Implementation of the algorithm

In order to reduce the communication between the main process and sub-process, we used the method of distributing coordinates in the complex plane alternately to maintain the balancing of load on every SHARC, which means the complex plane will be divided into *x_block***y_block* blocks, which are represented by coordinates of the leftmost point. Every block will be treated as a computing task. In order to prevent some sub-process from being the bottleneck of the whole computation, we adopted the polling scheme rather than sequential scheme to test which SHARC will send the result when the main process waits for the results. At the same time, an address table, namely “config.dat” file, is set up to store the addresses of the parallel accelerating board and its SHARCs in order to make the program support expansion. When the program runs, the table reads the addresses to CGF array from file in order to provide the addresses of the board and SHARCs. If a board is added into the system, the only thing we need to do is to add an address to that file, while if a SHARC is broken, we only need to delete its address from that file. All of these do not influence the execution of the program.

Research and implementation of the complete multi-grid parallel algorithm of a non-linear transient equation in the temperature field

1. Math model

We consider a non-linear heat transfer equation of ceramic/metal composite material

$$\frac{\partial u}{\partial t}=\frac{\partial}{\partial x}\left(k\frac{\partial u}{\partial x}\right),\quad 0<x<5,\ t>0,$$
$$u(0,t)=\cos(2\pi t),\quad u(5,t)=0,\quad u(x,t)=0.$$

The equation can be rewritten as

$$u_t=ku_{xx}+k_xu_x \quad \text{or} \quad u_t=ku_{xx}+G,$$

where

$$G=G(x,t)=k_xu_x,$$
$$k=(1-f_m)(0.1-0.01u+0.001u^2)$$
$$+f_m\cdot(1.0+0.1u+0.01u^2),$$
$$f_m=\begin{cases}x^2, & x<1,\\ 1.0, & x\geq 1.\end{cases}$$

2. Implementation of the parallel algorithm

Investigating the multi-grid parallel algorithm with VBF proposed by Guo *et al.*(2000), we found that if multi-grid method can be applied into every sub-domain, that is to say, if the method of VBF, domain decomposition method and the method of multi-grid on each sub-domain can be combined, we will get a higher efficiency algorithm called domain-decomposition-based complete multi- grid parallel algorithm with virtual boundary forecast. Its key ideas are as follows:

(1) Forecasting virtual boundary values using historical values of boundary points on each sub-domain by some methods, for example, linear forecast method, 2-order polynomial forecast method, maximum value of 2-order polynomial forecast method, etc. In this stage there is no communication between sub-domains.

(2) Performing multi-grid calculation on each sub-domain separately, independently in parallel without any communications between sub-domains. In this stage every sub-domain solves its own independent boundary problem.

(3) Smoothing result from Step (2) on the finest grid of the whole domain in a few cycles to reduce errors derived by Step (2). In this stage there is communications between corresponding sub-domains, passing the whole information and updating the virtual boundary values.

(4) Using distributed convergence model to determine whether a sub-domain calculation should be terminated or not. If it cannot be terminated the procedure returns to Step (1); otherwise a termination token is sent to the neighbor computers, informing them the calculation on the sub-domain has been finished, and there are no more communications between it and its neighbors. Meanwhile, results on the sub-domain are sent to master computer for assembling the final results.

(5) The whole calculation is terminated when all sub-domains have finished their own computations.

In the above algorithm, the following techniques are adopted to avoid the situation that the computation does not converge or converges falsely and to speed up the computation.

(1) False convergence means that the computation should have converged according to the error, but in fact it has not converged yet. Therefore, a constant $N_{\min}$ is defined as the minimum loop number in every sub-domain computation. Only when the loop number is greater than a certain constant can the computation be terminated.

(2) To speed up the computation, we take two steps to solve the problem. First, we calculate the approximate initial values on a very coarse grid, whose step is very large. The values need not be precise values, and they may be just initial guess values for virtual boundary, whose goal is to get the good initial values, because the approach can rapidly transfer the information on the known boundary. In this stage, every computer performs the same computation, gets the same initial results, then takes out the results which are related to its sub-domain. Then, the above Steps (1)~(5) are performed.

The algorithm is based on the domain decomposition method. It forecasts the virtual boundary values (the virtual boundaries are produced by the domain decomposition method), and implements the multi- grid method on each sub-domain. All of these quicken the convergence, reduce the communication, and improve the speedup.

As for the algorithm on the parallel system based on DSP, the whole domain is divided into some sub-domains according to the number of SHARCs. Each sub-domain has its boundary and its initial values. Every SHARC calculates a sub-domain independently. SHARCs exchange the boundary data by means of the link port. First all SHARCs calculate the initial values on the very coarse grid. Then, the host computer sends the computation task to each SHARC, which then performs the task, and every SHARC sends the results to the host computer to draw the graph. If necessary, SHARC can communicate with the neighbor SHARCs to exchange data on the boundary.

CASE STUDY

The two algorithms considered in this paper are useful in various military and business parallel systems with Multi-bus, ISA, PCI, developed by the Wuhan Digital Engineering Institute. The tested results are satisfying. We only take a 3U military reinforcing computer based on DSP as case study.

Results on 3U military reinforcing computer

The computer has two 3U-DSP boards, each of which has four SHARCs, and a VDSP board with two SHARCs. Using 1~10 SHARCs to draw the same picture of Mandelbrot Set respectively, we tested the results with respect to runtime, clock number, and the calculated speedup, parallel efficiency, as shown in Table 1.

Table 1 Related parameters of Mandelbrot Set on a 3U military reinforcing computer based on DSP

Number of SHARCs	Runtime (s)	Clock number	Speedup	Parallel efficiency
1	197.51	3597	1.000	1.000
2	98.86	1800	1.998	0.999
3	66.08	1203	2.990	0.997
4	49.71	905	3.975	0.994
5	40.05	729	4.934	0.987
6	33.45	609	5.906	0.984
7	28.78	524	6.865	0.981
8	25.26	460	7.820	0.977
9	22.52	410	8.773	0.975
10	20.54	374	9.618	0.962

Meanwhile, Table 2 shows the tested results only using a host computer.

Table 2 Runtime, clock number of Mandelbrot Set on a host computer

CPU	Runtime (s)	Clock number
STPC (80486)	1974.29	35945

For the problem of non-linear transient temperature field, the speedup gained respectively by the domain-decomposition-based complete multi-grid parallel algorithm with virtual boundary forecast (Method 1 for short) and multi-grid parallel algorithm with virtual boundary forecast (Method 2 for short) are shown in Table 3.

Table 3 Comparison of speedup of Method 1 and Method 2 used to solve a non-linear transient equation in the temperature field

Method	Speedup			
	1 SHARC	2 SHARCs	3 SHARCs	4 SHARCs
1	1.000	1.992	2.994	3.924
2	1.000	1.973	2.844	3.728

Discussion

From the results above, we can see that the speedup approaches linear with the increase of the number of nodes, which shows that the optimized approach to the Mandelbrot Set is feasible. And the speedup gained by Method 1 is faster than that gained by Method 2. With respect to the large-scale and complex heat problems on the parallel system based on DSP, the former is a more effective parallel algorithm.

CONCLUSION AND FUTURE WORK

We have investigated the parallel programming characteristics of the DSP-based parallel system, proposed an approach to draw Mandelbrot Set effectively, and used the method, called the domain-decomposition-based complete multi-grid parallel algorithm with virtual boundary forecast, to solve a non-linear heat transfer equation of ceramic/metal composite material. The method can also be applied to a lot of large-scale and complicated heat problems. It will improve the speedup and reduce the computation time. At the same time, we have discussed how to avoid the situation that the computation does not converge or converges falsely.

Aside from striving to seek more effective methods to forecast virtual boundary values using historical values of boundary points on each subdomain, our future research involves extending the domain-decomposition-based complete multi-grid parallel algorithm with virtual boundary forecast to solve more sophisticated and large-scale problems.

References

Analog Devices, Inc., 1995. ADSP-21000 C Runtime Library Manual (3rd Ed.), p.46-50.

Analog Devices, Inc., 1996. SHARC User's Manual (2nd Ed.), p.32-40.

Gan, W.S., Kuo, S.M., 2006. Teaching DSP software development: from design to fixed-point implementations. *IEEE Transactions on Education*, **49**(1):122-131. [doi:10.1109/TE.2005.863425]

Guo, Q.P., Zhang, S.S., Wei, J.N., 2000. Optimum tactics of parallel multi-grid algorithm with virtual boundary forecast (VBF) method running on a local network with the PVM platform. *Journal of Computer Science and Technology*, **89**(4):56-77.

Kuo, S.M., Gan, W.S., 2005. Digital Signal Processors. Prentice-Hall, Upper Saddle River, NJ.

Li, X.M., Jiang, Z.R., 1992. Parallel Algorithms. Hunan Science and Technology Press, Changsha, p.89-102 (in Chinese).

Makowski, D., Grecki, M., Jablonski, G., 2004. Application of a Genetic Algorithm to Design of Radiation Tolerant Programmable Devices. Proceedings of the 11th Mixed Design of Integrated Circuits and Systems. Poland, p.463-467.

Phil, J., Paul, S., Charlie, R., 2004. A Single Program Multiple Data Parallel Processing Platform for FPGAs. Proceedings of the 12th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM'04), p.115-117.