International Journal of Engineering & Technology, 7 (4.36) (2018) 29-31

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Structure Selection Technique of Multi-Processor Computing Systems

S.M. Kovalenko, G.V. Petushkov, O.V. Platonova

MIREA - Russian Technological University, Moscow, Russia

Abstract

This paper discusses construction of modern microprocessor structures and computing modules, including microprocessors and RAM, as well as computing systems containing computing modules combined by network means. Also, the "overhead" expenditure of time used for data transmission in the considered subsystems is analyzed. Estimates of the time spent on calculations and data exchange in multiprocessor computing systems (VS) are considered. An expression is derived for estimating the optimal number of computational modules in an aircraft that provides the minimum program execution time. Implementation of an aircraft with the structure recommended in the work, allows us to increase the overall performance of the system when performing the tasks that require intensive data exchange between individual computing modules. The results of research are expected to be used in the design and development of high-performance aircraft structures**.**

Keywords: *Multiprocessor computing systems, data exchange, computing modules, microprocessors, cache memory, RAM, performance of computing systems, data exchange rate*

1. Introduction

The solution of many currently important tasks imposed on computer systems requires high performance from such systems. Such tasks also include modeling in CAD of complex products, systems and tasks from such areas of science as nuclear physics, biology, pharmacology, climatology, and others.

The basis of a modern high-performance computing system (CS) is the most complex product of modern digital microelectronics, a modern multi-core microprocessor (MP), which provides performance when performing computational operations in 1-100 billion operations per second. However, the required performance of a computing system for solving complex problems can be ensured only by tens, hundreds or even thousands of such MPs assembled into a system. Choice of the structure of such aircraft largely determines its characteristics and should be made considering the influence of both the attributes of processors and the attributes of communication system uniting them.

The purpose of this study is to develop recommendations on the choice of high-performance computing systems structures which are highly efficient when executing computer programs requiring intensive data exchanges between individual program fragments. Recommendations should take into account both the attributes of the microprocessors themselves and attributes of network tools that unite them.

Modern high-performance parallel aircraft, including dozens and hundreds of MPs, are created on standard equipment including: mass-produced MPs and network equipment. Such aircrafts are built according to a hierarchical principle: at the first level of the hierarchy, several MPs (from 1 to 8) together with the necessary RAM and network adapters are combined into a computing node or module (VM).

2. Research Method

According to the common classification, there are three basic schemes for organizing computations in a VM: calculations under control of the control flow, under control of the data flow, and under control of the flow of requests.

For calculations under control of the control flow, selection phase corresponds to the selection of a command at the address indicated by contents of the command counter and in accordance with place of the command in program structure. Being selected, the command, bypassing the verification phase, is automatically received for execution. As a result of the command execution, contents of the common part of memory are changed. Then contents of the command counter are incremented, and the next command is sampled; or in the case of a transition, a corresponding change in the command counter contents and a selection of the command indicated by it occur.

In machines based on data flow, commands are executed as soon as all their arguments are ready, regardless of their place in the program structure. This means that as soon as they are ready for execution, teams should be distributed over corresponding computing elements. We can assume that the selection phase here logically consists in assigning a computing element to each team. At the next phase, readiness check of all arguments is carried out, and they should be data, not expressions, for which calculations have not yet been carried out. If the values have not yet been obtained, the computational element does not execute the command, while remaining awaiting data. Execution phase leads to a change in the state of command being executed and the set of its successor teams. The team as if consumes its arguments and puts the result in each successor command. There are, however,

slightly different options for organizing an aircraft based on data flow.
For calculations under control of request flow, there can also be several different variants of the organization of aircraft. We confine ourselves to the most common.
In a reduction machine, instruction selection rule determines the distribution of computational elements for start of each computational cycle. In the next phase, availability of arguments is checked from the position of possibility of executing a command. If the test is positive, command is executed; if it is negative, the computational element takes actions aimed at obtaining the required element, i.e. sends relevant requests. After receiving the corresponding arguments in form of data, the reverse process of receiving and transmitting results is carried out in the execution phase.
So, in machines on the basis of control flow, an unambiguous selection of commands is made according to their place in program structure, and, once selected, they are immediately executed. In machines based on data flow, any command can, in principle, be activated regardless of its place in program structure, but it is executed only when its arguments are ready. In machines based on the flow of requests, commands are activated by requests with a delay of execution to search for a command with ready-made arguments, after which the process is reversed - execution of commands with the corresponding transmission of results.
The advantage of machines based on the flow of control is complete and unambiguous control of sequence of commands executed, without waiting for any arguments. The advantage of machines based on data flow is the implementation of a high degree of completeness of internal parallelism of computations, since any command can begin to execute as soon as its arguments become ready. The disadvantage is that teams can uselessly waste time waiting for arguments that are not really necessary.
The advantage of a machine based on the request flow is that only those commands are executed, and the results of which are needed immediately. Mechanism for calling procedures laid down naturally in them, determines blocks of program commands. At the same time, the query mechanism for performing even a single operation spans many commands, activating them with a time delay before execution. This requires a memory organization with a complex control scheme (stacking).
Considering that the MPs themselves are multi-core and have a structure where the cores are combined either with a common cache memory or an intracrystal communication network, a structure arises at the VM level, where the MPs and the RAM modules are combined using a cross-switch (PC) [1 , 2]. At the next level, several VMs are combined into a structurally complete node: a block or rack, and functionally VMs are combined by connecting them to the ports of switches or network routers [1,3,4]. Most often, the AF uses two networks: a data network and a service network (a network that supports the operation of a network operating system). Nowadays, InfiniBand with low latency and high bandwidth is most often used as a data transmission network, as Fast — or GigabitEth [5, 6] as a service network.
An example of such a multiprocessor VS implementation is, for example, the RoadRunner system developed by IBM, or the Lomonosov domestic supercomputer installed in the MSU computer center. Lomonosov.
The T-Platforms company has implemented its new development of the T-Blade2 computing node for cluster architecture systems in the MVS Lomonosov. The system chassis occupies 7U and is intended for installation in a standard hardware cabinet. There are 16 blades installed in the chassis (motherboards with components), each of which has two dual-processor boards with support for Intel Xeon 5500 processors. Thus, 32 dual-processor computing nodes are installed in the chassis. When installing Xeon 5570 processors, the peak chassis performance is 3 Teraflops (10 ** 12), which allows you to achieve 18 Teraflops in a standard cabinet (42U).
Computing node includes two central processors, each of which is connected to a removable memory module type DDR3-1333 (three-channel). The node contains the QDR Infiniband controller chip, and the node is connected to the IBA network via a backplane with a switch installed in the chassis. Two switches are installed in the chassis, each of which has 20 external QSFP standard ports and 16 internal ones. Thus, 16 computing nodes are connected to each switch and the possibility of building an IBA network is provided. Two networks of the Ethernet standard are used to control the node: GigabitEthernet for control at the OS level and Fast Ethernet, to which the naval controller is connected for control via the IPMI protocol.
The high density of placement of compute nodes in the chassis did not allow adding ability to install a hard disk drive to the compute node (the system has a common high-performance data storage system), but it is possible to install a micro SD standard drive in each node.
An important role in system is played by the T-Blade2 control module, built on the basis of low-voltage Intel Yonah processor. The control module contains Ethernet switch chips and an FPGA chip to support an additional barrier network and synchronization networks, as well as a 2.5-inch hard disk drive. The control node (like a compute node) provides the ability to remotely control via IPMI via an IUD chip installed on the board.
Increase in productivity of aircraft with an increase in the number of MPs and VMs is non-linear, since escalating the number of VMs in the armed forces and the "overheads" associated with the time is required to implement data exchange between the VMs grow. The time spent on data exchange is most significant when exchanging the network between VM, because there is more delay and traffic.

3. Results Anda Nalysis

To analyze the impact of such "overheads", we will consider some idealized variant of a well-parallel program that has N parallel executable fragments on a K VM, with N more than K. The program execution time in this case can be estimated as:

$$T_{\text{пр}} = T_{\text{выч}} + T_{\text{обм}}, \quad (1)$$

where Твыч – is the time spent on calculations in the VM,
Тобм – the time spent on the exchange of data between the VM.
Твыч can be estimated taking into account the performance of one VM Рвм and their number K as:

$$T_{\text{выч}} = \frac{G}{(K * P_{\text{ВМ}})}, \quad (2)$$

where G –is the estimate of the number of operations in the program.
The value of G can be obtained by analyzing the program's algorithm, using, for example, the Halstead metrics.
The time spent on the exchange of a Тп data packet in a data network can be obtained as:

$$T_{\text{п}} = T_{\text{н}} + T_{\text{з}} + \frac{Q}{V}, \quad (3)$$

where Тн – the delay in the formation of a packet of data in the network adapter,
Тз – packet transmission delay in the network, associated with delays in the switch,
Q - the amount of data transmitted in the data packet
V - data transfer rate in the network.
We consider a data network as part of the Armed Forces, since the volume of traffic in it is much larger than in the service network. Analyzing the time of data exchange as "overhead" in the

calculation process, in the first approximation can be estimated taking into account the limited network bandwidth as:

$$T_{\text{обм}} = K * \frac{Q}{V}, \quad (4)$$

Consider getting the expression (4) in somewhat more detail using the example of a system that includes K VM. The volume of traffic of each VM will be reduced in proportion to the number of VMs, however, we believe that each VM sends packets to the others after the execution of its fragment of the program. Combining the traffic of all VMs sent over the network, we obtain an estimate of the time given in expression (4).
The amount of data in bytes of an exchange packet is related to the number of operations in the G program executed (computational complexity of the algorithm) as:

$$Q = C * G * L, \quad (5)$$

where C is a coefficient characterizing the class of executable algorithms in terms of connectivity according to data [7], algorithms with greater connectivity according to data are characterized by a higher intensity of exchanges between the VM and a large value of the coefficient C, and L is the share of calculation operations in the program fragment (L≈1).
Taking into account expressions (2) - (5), expression (1) can be rewritten in the form:

$$T_{\text{пр}} = \frac{G}{(K*P_{\text{ВМ}})} + \frac{(G*C*L)}{V} * K, \quad (6)$$

As follows from the above expression (6), as K increases, the time for computing decreases and the time for data exchange in the aircraft increases. Thus, the characteristics of the data network will greatly affect the work of the aircraft.
From the presented expression (6) it follows that it is possible to estimate the optimal value of the number of VMK, which provides the minimum value of the execution time of the program Tпр. The value of k is determined from the condition dTпр/dk=0 and is estimated as:

$$K_{\text{оп}} = \sqrt{\frac{V}{C*P_{\text{ВМ}}*L}}, \quad (7)$$

For modern aircraft, $K_{\text{оп}}$ values are in the unit area for V - 1 Gbyte / s and Рвм– 1000 million op / s, and only for C values - 0.01 byte / op, which characterize algorithms with weak data connectivity, the KOP will be about 10. As the data transmission network capacity increases, the number of VMs in the aircraft can be increased further without fear of loss of the aircraft's performance due to the increase in time for data exchange. We call this expression for Cop the rule of “choosing the effective structure of the Armed Forces” and emphasize that it is of great practical importance in the design of computing systems.
Main result of the study is the expression to estimate optimal value of the number of VMs in the Armed Forces. On this basis, when the number of VMs should be chosen to be less than or equal to the optimal value a rule has been developed for choosing an effective structure of an Armed Forces.

4. Conclusion

The ANALYSIS performed allows us to conclude that for effective scaling of aircraft reducing the “overhead” time spent on data exchange in the system is required. This needs the use of new network implementation technologies for connecting VMCs with shorter delay times in data exchange.

Additional studies are needed to assess influence of the characteristics of the task performed in VS, namely, the number of parallel fragments of the task and the required volume of data exchange, to determine more accurate and correct values of the optimal number of VMs in the VS.

References

[1] Hennessey, John L., Patterson, David A. Computer architecture. Quantitative approach. Edition 5th. M .: Technosphere. 2016. 936s.
[2] Kalachev A.V. Multi-core processors. M .: Internet University of Information Technology. 2010. 247s.
[3] Bogdanov A.V., Korkhov V.V., Mareev V.V., Stankova E.N. Architecture and topology of multiprocessor computing systems. M .: Internet University of Information Technology. 2012. 176s.
[4] S. Orlov, B. Ya. Tsilker Computer organization and systems. 3rd edition. SPb .: Peter. 2016. 688c.
[5] Makagon D., Syromyatnikov E. Networks for supercomputers. // Open systems. № 7. 2011.
[6] Kulagin V.P.Problems of analysis and synthesis of structures of parallel computing systems. // Russian Technological Journal, 2013. № 1, P. 1-19.
[7] VV Voevodin, Vl. Voevodin. B. Parallel computing. SPb .: BHV-Petersburg. 2002. 608c.