NÂNG CAO HIỆU NĂNG HỆ VI XỬ LÝ ĐƠN CPU TRONG HỆ XỬ LÝ SONG SONG ĐA CPU

Phạm Xuân Bách[1], Đỗ Xuân Tiến[2], Chu Đức Toàn[3*]
[1]Đại học Sư phạm Kỹ thuật Nam Định, [2]Học viện kỹ thuật Quân sự, [3]Đại học Điện Lực

TÓM TẮT

Các hệ xử lý song song đa CPU chuyên dụng, do được phân rã tốt nên các hệ vi xử lý đơn CPU sẽ đảm nhiệm phần lớn thời gian làm việc của toàn hệ, đồng nghĩa với việc hiệu năng toàn hệ sẽ phụ thuộc chủ yếu vào hiệu năng của các hệ đơn CPU. Việc tổ chức bộ nhớ của chúng đóng vai trò quyết định đến hiệu năng của toàn hệ. Bài báo này xây dựng một kiến trúc bộ nhớ song song cùng phương pháp tổ chức cơ sở dữ liệu kiểu vector giúp cải thiện đáng kể hiệu năng cho kiến trúc bộ nhớ song song. Khi khảo sát hệ thống theo mô hình này, chúng tôi thu được quan hệ định lượng cho các thông số kiến trúc, thiết lập được mô hình tính toán hiệu năng của hệ xử lý song song chuyên dụng.

Từ khóa: *Hệ xử lý song song đa CPU chuyên dụng; hiệu năng; bộ nhớ song song; luồng dữ liệu; tốc độ dữ liệu*

ĐẶT VẤN ĐỀ

Các hệ xử lý song song đa CPU chuyên dụng thường xử lý một bài toán cụ thể hoặc một lớp bài toán cùng thuộc tính [3,4]. Khi đó khả năng phân rã chức năng được cho là rất tốt, đồng nghĩa với việc nhiệm vụ được chia đều cho các hệ vi xử lý thành phần. Thời gian thực hiện nhiệm vụ của cả hệ thống có thể coi là thời gian làm việc của các hệ đơn CPU có trong hệ thống. Tính toán hiệu năng cho các hệ đơn CPU chính là tính toán hiệu năng cho toàn hệ.

Trong các hệ xử lý song song đa CPU, thành phần CPU có tốc độ vượt trội nhiều lần so với bất kỳ thành phần nào của hệ, kể cả bộ nhớ. Vì vậy, điều tiên quyết để hệ có thể đạt hiệu năng yêu cầu thì trước hết phải tổ chức một kiến trúc song song cho không gian nhớ của hệ đơn CPU. Kiến trúc này từ lâu đã được biết đến là kiến trúc bộ nhớ đan xen [5]. Vấn đề là lựa chọn một kiến trúc hợp lý. Bài báo này lựa chọn kiến trúc có tính tới độ tin cậy làm việc của bộ nhớ, đặc biệt khi chúng phải làm việc ở tốc độ quét cao. Vì vậy kiến trúc bộ nhớ đan xen kiểu C-access được sử dụng thay vì kiến trúc S-access như thông thường (hình 1). Tuy có phức tạp hơn, nhưng khi tốc độ quét module nhớ cao, cơ cấu chốt địa chỉ của nó giúp lưu địa chỉ của từng ngăn nhớ của từng module nhớ. Điều này là rất quan trọng vì CPU làm việc theo từng chu kỳ máy, kể cả các chu kỳ máy tham chiếu bộ nhớ. Khi tham chiếu bộ nhớ, CPU chỉ cấp thông tin địa chỉ, thông tin điều khiển, cấp thông tin dữ liệu hoặc thu thông tin dữ liệu theo nhịp clock của riêng mình mà không cần biết bộ nhớ đã chốt được các thông tin của mình hay chưa. Vì vậy kiến trúc C-access đặc biệt hữu ích khi nó kịp phản ứng với bất kỳ tốc độ tham chiếu nào, vì cơ cấu chốt địa chỉ của nó được thực hiện bằng phần cứng. Chú ý cách địa chỉ hoá cho các ngăn nhớ được tiến hành trong kiến trúc C-access là tập hợp n bit của kênh địa chỉ được chia thành 2 phần, các bit địa chỉ thấp a_{m-1}- a_0 (*m* bit,) chỉ rõ số hiệu các modul bộ nhớ còn các bit địa chỉ cao a_{n-1}- a_m (*n-m* bit) quy định vị trí các ô nhớ trong một module.

MÔ HÌNH HIỆU NĂNG HỆ XỬ LÝ SONG SONG ĐA CPU

Hellerman [1,2] đã giới thiệu một mô hình, trong đó một luồng tham chiếu được quét theo thứ tự đến của chúng cho đến khi tìm thấy module nhớ lặp lại đầu tiên. Vì vậy, chuỗi *k* yêu cầu riêng biệt đầu tiên này được truy nhập song song. Hiệu năng E đối với bộ xử lý đơn CPU có 2^m modul nhớ đan xen là:

* *Tel: 0982 917093, Email: toancd@epu.edu.vn*

$$E(2^m) = \sum_{k=1}^{2^m} kP(k) = \sum_{k=1}^{2^m} \frac{k^2.(2^m-1)!}{(2^m)^k.(2^m-k)!} \quad (1)$$

Trong đó, P(k) là xác suất để luồng tham chiếu có chuỗi yêu cầu với độ dài bằng k.

Nhóm tác giả, trong công trình [1] cũng tìm được mô hình tính hiệu năng thành phần nhằm tăng độ chính xác bằng công thức, khi dữ liệu tham chiếu có kiểu vô hướng:

$$E(2^m) = \sum_{k=1}^{2^m} kP(k) = \sum_{k=1}^{2^m} \sum_{j=0}^{k-1} (\frac{1}{2^m}) \chi^j \theta^{k-j-1} C(j,k) \quad (2)$$

Luận giải các mô hình trên cũng khá phức tạp, tuy nhiên có thể nhận xét: (i) Mô hình Hellerman (1) chỉ phù hợp với việc tính toán hiệu năng đối với máy tính đa nhiệm kiểu SuperComputer hay MiniComputer. (ii) mô hình (2) quá phức tạp để có thể dùng nó như công cụ để điều tiết và điều khiển kiến trúc bộ nhớ đan xen C-access để có hiệu năng cao cho các hệ đa CPU chuyên dụng. Mô hình (3) là một cố gắng trong việc đơn giản hóa mô hình hiệu năng mà [1] đã đạt được. Tuy vậy nếu mô hình này phải tích hợp thêm ảnh hưởng của tính vô hướng trong tổ chức dữ liệu trong các module nhớ thì hiệu năng còn giảm nữa [2] và trong nhiều nhiều nhiệm vụ sẽ không đáp ứng yêu cầu về hiệu năng của hệ thống.

Trong trường hợp hệ có kiến trúc Harvard thì hiệu năng của hệ phụ thuộc vào hiệu năng của 2 luồng tham chiếu là luồng lệnh và luồng dữ liệu. Mặt khác luồng dữ liệu là luồng có cấu trúc vô hướng nên hiệu năng rất thấp.

Đối với luồng lệnh [1] nhận thấy trong mỗi chu kỳ bộ nhớ, CPU thực hiện thao tác đọc các lệnh chứa trong các mođun nhớ do bộ đếm lệnh trỏ tới. Giả thiết rằng nếu một lệnh rẽ nhánh xuất hiện với xác suất λ trong luồng lệnh và gọi P(k) là xác suất mà k trong số n lệnh sẽ được giải mã thì xác suất để k=1 sẽ là P(1)= λ. Khi k ≠1, [1] cho hiệu năng tính cho luồng lệnh là:

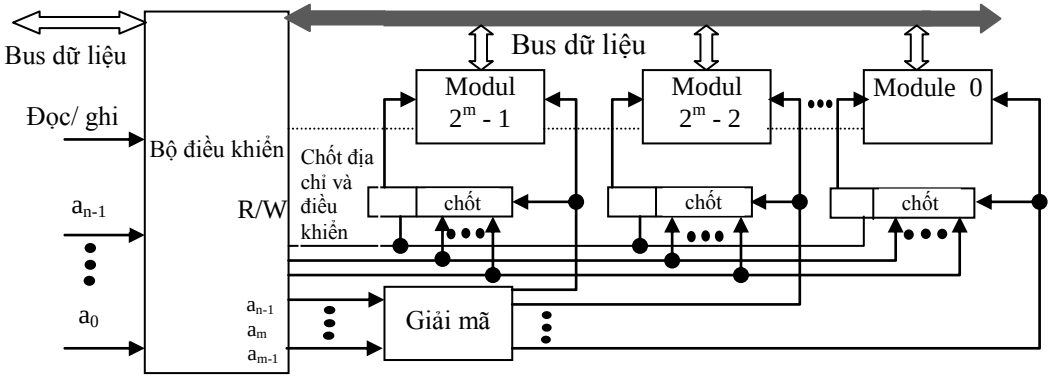
$$IE = \frac{1-(1-\lambda)^n}{\lambda} \quad (3)$$

Đây là mô hình tốt nhất của [1] khi nó bỏ qua phân bố của dữ liệu trong bộ nhớ, là loại phân bố vô hướng làm suy giảm đáng kể hiệu năng chung của hệ vi xử lý. Khảo sát IEm theo λ theo số lượng module nhớ M được thể hiện trên hình 2.

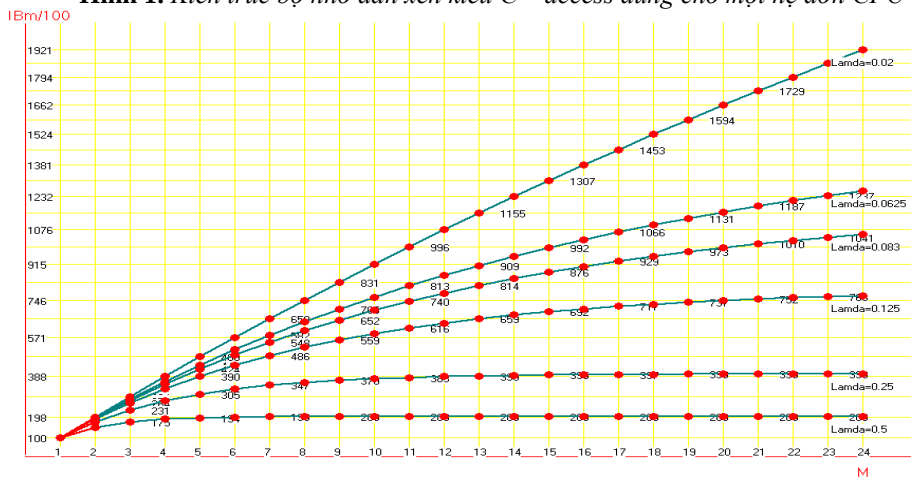
MÔ HÌNH ĐỀ XUẤT CHO HIỆU NĂNG HỆ XỬ LÝ ĐƠN CPU TRONG HỆ XỬ LÝ SONG SONG ĐA CPU CHUYÊN DỤNG

Để giải quyết, bài báo sử dụng các đặc điểm của hệ xử lý song song đa CPU chuyên dụng. Ngoài đặc trưng đã nêu ở trên, còn một đặc điểm rất quan trọng là dữ liệu các lớp bài toán chức năng là dễ vector hóa như dữ liệu ảnh số, dữ liệu multimedia, dữ liệu truyền thông...; kích thước của 1 vector có thể quy định trong hệ chuyên dụng; dung lượng không gian nhớ không quá lớn. Nếu dữ liệu là kiểu vector thì vấn đề còn lại là tổ chức kích thước cho phù hợp với kiến trúc module nhớ song song. Xét nguyên lý hoạt động của luồng dữ liệu bộ nhớ trên hình 3. Nếu kích thước của các vector không đồng nhất sẽ xảy ra tình trạng như hình 3 a, b, c. Nếu kích thước các dữ liệu vector là đồng nhất thì hiệu năng sẽ đạt cực đại khi kích thước vector bằng đúng số lượng module nhớ M. Để thực hiện được điều đó chỉ cần điền thêm giá trị 0 vào phần thiếu của kích thước vector dữ liệu. Các dữ liệu vô hướng sẽ chỉ tồn tại trong quá trình xử lý và gia công ở các khâu trung gian, mà những khâu xử lý này chủ yếu xảy ra ở trường thanh ghi đa năng của CPU.

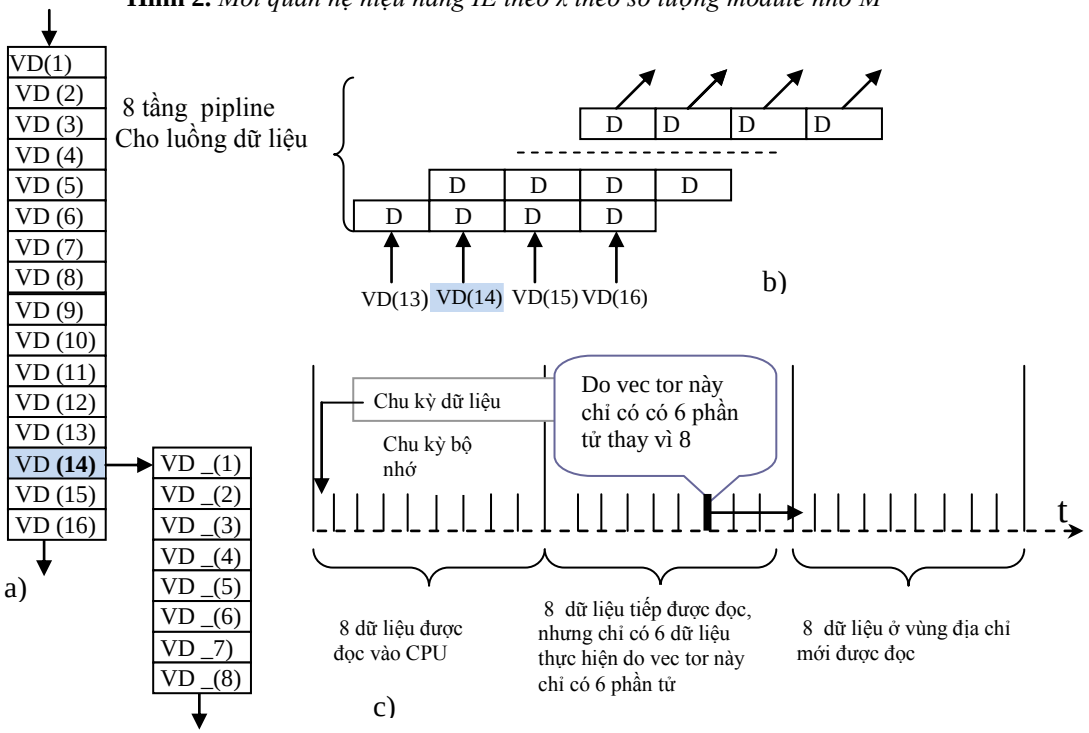
Bằng cách này, kích thước của một vector luôn bằng bội của số lượng module nhớ trong hệ đơn CPU. Thường thì bằng chính lượng module nhớ. Do đó truy cập vào bộ nhớ dữ liệu luôn đạt hiệu năng tối đa, tức là bằng tốc độ truy cập của CPU và bằng số lượng module 2^m nhớ được tổ chức song song trong hệ: DEm= 2^m.Trong thực tế số lượng module nhớ thường được tổ chức bằng 2,4,6... nên để tiện cho tính toán cần thay công thức tính hiệu năng luồng tham chiếu dữ liệu là DEm=M, với M là số lượng module nhớ được tổ chức song song theo kiểu C-access cho từng hệ đơn CPU.



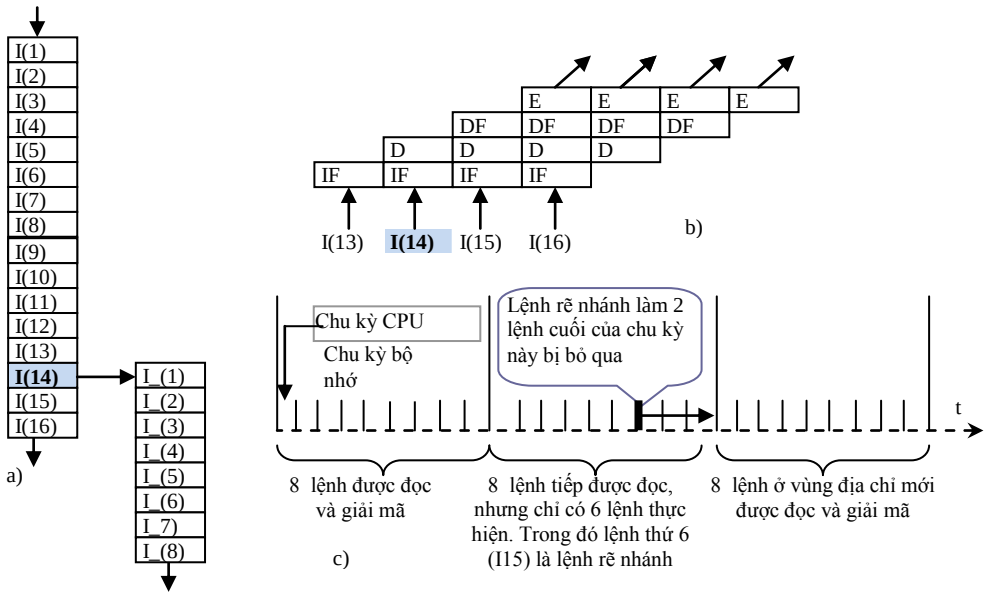
Hình 1. *Kiến trúc bộ nhớ đan xen kiểu C – access dùng cho một hệ đơn CPU*



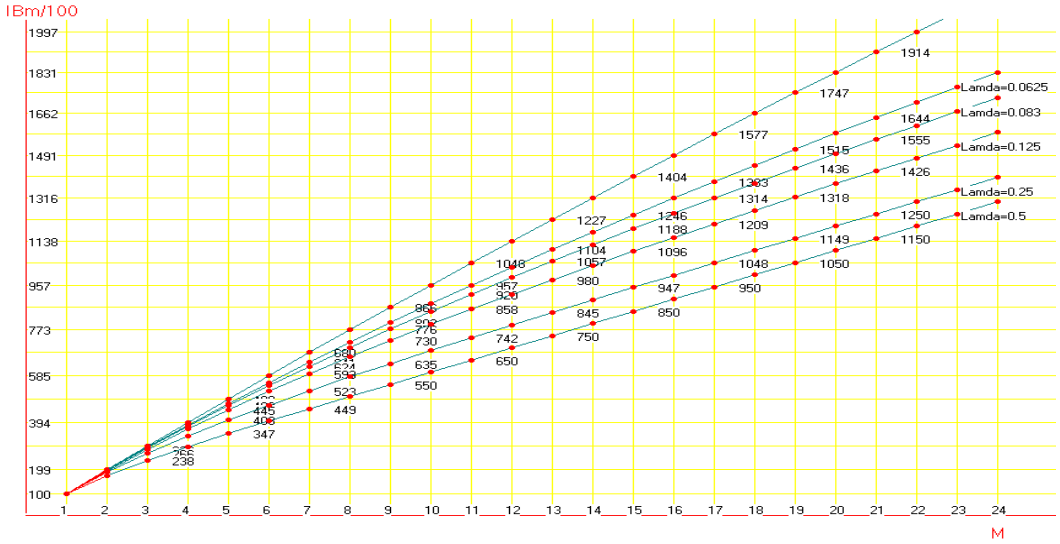
Hình 2. *Mối quan hệ hiệu năng IE theo λ theo số lượng module nhớ M*



Hình 3. *Nguyên lý hoạt động của luồng dữ liệu bộ nhớ*



Hình 4. *Nguyên lý hoạt động của luồng lệnh*



Hình 5. *Mối quan hệ hiệu năng E_M theo λ theo số lượng module nhớ M*

Bảng 1. *λ =0,02*

M	06	08	10	12	14	16	18	20	22
IE_m theo mô hình [1]	5,71	7,46	9,15	10,76	12,32	13,81	15,24	16,62	17,94
Mô hình đề xuất	5,85	7,73	9,57	11,38	13,16	14,91	16,62	18,31	19,97

Bảng 2. *λ =0,5*

M	06	08	10	12	14	16	18	20	22
IE_m theo mô hình [1]	1,94	1,98	2,08	2,08	2,08	2,08	2,08	2,08	2,08
E_M theo mô hình đề xuất	3,47	4,49	5,50	6,50	7,50	8,50	9,50	10,50	11,50

Đối với luồng lệnh, tại thời điểm bắt đầu một chu kỳ lệnh, bộ quét trỏ tới n lệnh liên tiếp kể từ địa chỉ bộ đếm lệnh hiện tại. Do đó n module nhớ sẽ bận trong chu kỳ này.

Nếu một lệnh rẽ nhánh được giải mã trong chu kỳ dữ liệu tiếp theo, thì n lệnh cho chu kỳ lệnh tiếp theo sẽ được bắt đầu từ địa chỉ của lệnh rẽ nhánh đó. Đối với hàng chứa lệnh, nếu coi λ là xác suất để một lệnh là rẽ nhánh thì hiển nhiên tất cả các lệnh đọc được sau lệnh rẽ nhánh của chu kỳ lệnh này sẽ không được giải mã (hình 4).

Mô hình đề xuất sẽ là tổng của hiệu năng luồng tham chiếu lệnh IE_M và luồng tham chiếu dữ liệu DE_M trong kiến trúc Harvard và được xác định:

$$E_M = IE_M + DE_M \quad (4)$$

Trong đó, $DE_M = M$ như đã phân tích còn IE_M được xác định như nguyên lý tính xác suất thông thường. Cho rằng xác suất để k trong số n lệnh được giải mã tương đương k-1 lệnh trước đó không phải lệnh rẽ nhánh và lệnh thứ k là lệnh rẽ nhánh sẽ là:

$P(k)=(1-\lambda)^{k-1}\ \lambda$; $1<k<n$, nghĩa là cho toàn chuỗi lệnh n thì xác suất đó sẽ là $P(n)=(1-\lambda)^{n-1}$

Từ đó ta tính được

$$IEn = \sum_{k=1}^{n} \mathbf{kP(k)} = \lambda + 2p(1-\lambda) + 3p(1-\lambda)^2 + \ldots + n(1-\lambda)^{n-1}$$ và cũng bằng $\frac{1-(1-\lambda)^n}{\lambda}$

Từ đó, hiệu năng toàn phần sẽ là:

$$E_M = IE_M + DE_M = \sum_{k=1}^{M} kP(k) + M \quad (5)$$

Khảo sát mô hình (5) thu được một tập hợp giá trị E_M hợp giá trị khi cho λ và M biến thiên trong miền giá trị thực tiễn mà các hệ đa CPU chuyên dụng hay sử dụng.

Lập bảng so sánh mối quan hệ hiệu năng E_M theo số lượng module nhớ M khi λ=0,02 và λ=0,5, thể hiện ở bảng 1 và bảng 2.

Rõ ràng, trong tất cả các trường hợp khi có cùng giá trị λ và M thì hiệu năng E_M của mô hình đề xuất đều lớn hơn hiệu năng IEm của mô hình cũ khi mô hình đề xuất thành công trong việc vector hóa cơ sở dữ liệu chứa trong bộ nhớ C-access.

KẾT LUẬN

Với mô hình đề xuất, hiệu năng tổng của các hệ xử lý song song đa CPU chuyên dụng tăng lên đáng kể, đặc biệt khi số lượng các module nhớ M lớn và sác xuất gặp lệnh rẽ nhánh λ thấp. Điều này là phù hợp với các hệ xử lý chuyên dụng khi mà cơ sở dữ liệu được vector hóa dễ dàng với cấu trúc xác định trước. Khi đó chỉ cần tổ chức bộ nhớ song song kiểu C-access với số lượng đúng bằng kích thước của dữ liệu vector thì hiệu năng truy cập đối với dữ liệu loại này đạt xấp xỉ 100%.

Đối với truy cập vào vùng chứa lệnh, thuật toán cho các hệ chuyên dụng thường hạn chế các lệnh ngắt (đặc biệt các lệnh ngắt cứng) để tránh việc dừng đột đột ngột một tiến trình nào đó đang xử lý, nên xác suất gặp lệnh rẽ nhánh λ thấp. Vì vậy các đường đặc tuyến của hình 5 thường là ở phía trên nên hiệu năng của IE là khá cao.

TÀI LIỆU THAM KHẢO

1. Nguyễn Minh Ngọc, Đỗ Xuân Tiến, Vũ Hoàng Gia.Về Thông lượng trung bình của hệ lưu trữ song song. Tạp chí Khoa học và Kỹ thuật, HVKTQS số 115, II-2006.

2. Hellerman, H., and Smith, H. J Throughput Analysis of some Idealized Input, Output, and Compute Overlap Configurations." Computing Surreys, 2, June 1980, pp. 111-118.

3. M. V. Wilkes. Slave memories and dynamic storage allocation. IEEE Transactions Electronic Computers Vol EC-14. No 2 April 2005.

4. Васильев А.Е., До Суан Тьен, Кабесас Д., Садин Я.Д., Донцова А.В. Методологические аспекты и инструментальные средства автоматизированного Информатика. Телекоммуникации. Управление. Номер 6(169) 2013 г.Стр. 123-134.

5. Бородин, А.М. Сравнительный анализ возможностей и скорости обработки многомерных данных программными средствами бизнес-аналитики на основе индексирующих структур основной памяти [Текст]/А.М. Бородин, С.В. Телекоммуникации, Управление.–2010.–№ 1.– С. 99–102.

SUMMARY

TO IMPROVE THE EFFICIENCY OF SINGLE CPU SYSTEM IN THE PARALLEL MULTI-CPU SYSTEM

Pham Xuan Bach[1], Do Xuan Tien[2], Chu Duc Toan[3*]
[1]Nam Dinh University of Technology Education,
[2]Academy of Technology and Military, [3]Electric Power University

The specialized parallel multi-CPU systems, so should be good decompose then the single-CPU processors will undertake the majority of a time working system, meaning that the system efficiency will depend primarily on the efficiency of single CPU systems. The organization of our memory plays a crucial role in the efficiency of the all system. This paper build a parallel memory architecture and method of vector database organization significantly improve efficiency for parallel memory architecture. When surveying system according to this model, we obtained quantitative relation to architectural parameters, created a efficient model of the specialized parallel multi-CPU systems, the result emulation cho precision best.

Key words: *The specialized parallel multi-CPU systems; efficiency; parallel memory; data stream; computational speed*

Ngày nhận bài:28/3/2014; Ngày phản biện:01/4/2014; Ngày duyệt đăng: 09/6/2014
Phản biện khoa học: *PGS.TS Nguyễn Huy Hoàng – Học viện Kỹ thuật Quân sự*

[*] *Tel: 0982 917093, Email: toancd@epu.edu.vn*