

СИСТЕМА КОММУТАЦИИ С РАСПРЕДЕЛЕННЫМ УПРАВЛЕНИЕМ

The article is devoted the decision of problem of removal of such failings as: conflicts of access; difficulty of realization of more effective asynchronous mode of interprocessor exchange, which are caused application of the centralized management in the methods of switching of informative channels.

Введение. Любой общесистемный ресурс, какова бы ни была его природа, ухудшает практически все основные системные характеристики и ограничивает диапазон масштабируемости системы.

Коммутатор всегда рассматривался как один из самых важных узлов любой многопроцессорной системы (шина классифицируется как коммутатор с временным разделением каналов).

Во всех известных методах коммутации информационных каналов, используемых как в сетях, так и в многопроцессорных системах, применяется централизованное управление, как правило, возлагаемое на управляющую ЭВМ [1-2].

Если для сетей такое управление является естественным, то присущие ему недостатки: конфликты доступа, невозможность осуществления более эффективного асинхронного режима межпроцессорного обмена, не только существенно снижают скорость процессов обмена в многопроцессорных вычислительных системах, но и создают трудно решаемую проблему поддержания когерентности данных. Отказ от централизованного управления позволяет попутно упростить и решение других системных вопросов:

- Устраняются конфликты доступа.
- Упрощается решение проблемы когерентности данных.
- Кардинально упрощается программная процедура межпроцессорного обмена.
- Принципиально по-другому и более эффективно решается проблема организация оперативной памяти системы.
- Упрощаются критерии распараллеливания программ.

Предлагаемая методология развивает методы построения ортогональных многопроцессорных вычислительных систем, используя в полном объеме принципы организации памяти обмена, способы адресации к ней и обеспечения когерентности данных [3-4].

Структурная организация системы коммутации. В отличие от сетей в многопроцессорных системах порядок информационного обмена вытекает непосредственно из алгоритма решаемой задачи. При распараллеливании для

каждого потока (thread) можно сформировать последовательность, согласно которой каждый данный поток будет взаимодействовать с остальными потоками системы, и указать вид взаимодействия (приём или передача данных).

Последнее обстоятельство позволяет перейти к распределенному управлению системой коммутации информационных каналов, один из вариантов организации которого приведен на рисунке 1. В состав многопроцессорной вычислительной системы входят процессоры P1, P2, P3, P4 и система коммутации с распределенным управлением. Система коммутации представлена матрицей размерностью 4x4, главную диагональ которой составляют дешифраторы (DC). Остальными элементами являются шинные формирователи (ШФ).

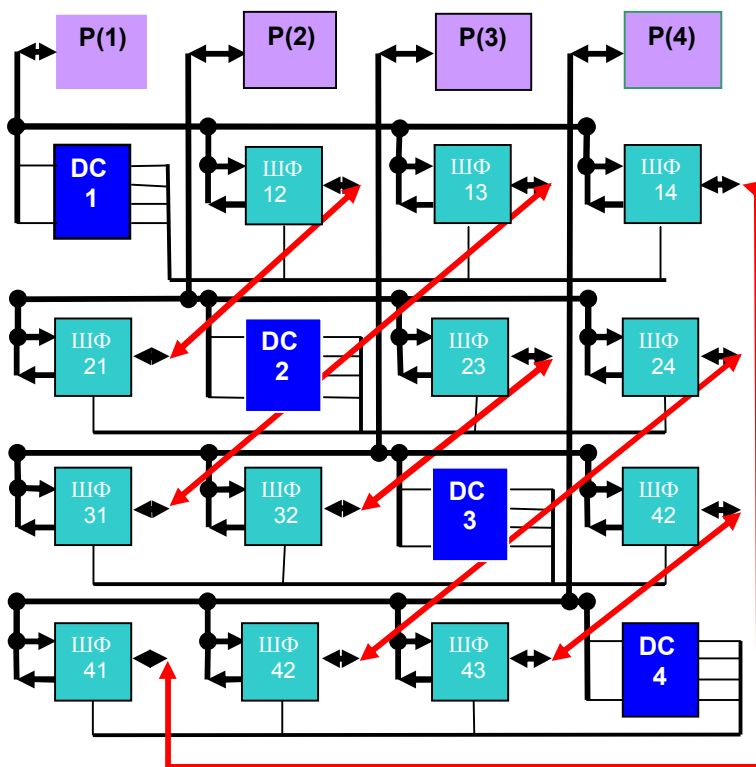


Рис. 1. Структурная организация системы коммутации

Как известно, обычные коммутаторы имеют адресный вход, адрес на который выставляет управляющее устройство. Процессоры являются в этом

процессе пассивными элементами – они только могут запросить обмен с одним из остальных процессоров системы у управляющего устройства.

Как понимается распределенное управление? Каждый процессор управляет соответствующей ему частью системы коммутации, причем, единолично. Любой произвольный шинный формирователь i -й строки матрицы может быть активизирован только i -м процессором системы (под процессом активизации подразумевается открытие шинного формирователя и задание направления передачи информации). Таким образом, каждый процессор управляет только шинными формирователями i -й строки матрицы. Предложенная схема коммутации не предполагает наличие общесистемного устройства управления, хотя оно достаточно просто может быть использовано и в данном случае.

Шинные формирователи разбиты на пары, выходы которых соединены. Каждая такая пара, которую назовем узлом спаренных формирователей (УСФ), отвечает за информационный обмен только между двумя заданными процессорами системы.

Чтобы информационный обмен между двумя процессорами системы стал возможен необходимо программно активизировать в соответствующем УСФ оба шинных формирователя таким образом, чтобы обеспечить заданное алгоритмом направление передачи информации. При этом если один из них работает на запись, то другой должен работать только на чтение.

Для n -процессорной системы при такой системе коммутации могут быть сформированы $n/2$ асинхронных параллельных канала обмена, а общее число независимых каналов равно $n(n-1)/2$.

Принципы управления коммутатором и его функционирование.

Принципы адресации, разработанные для ортогональных многопроцессорных систем, применимы и для данного случая. Чтобы обеспечить непрерывность адресного пространства операций ввода-вывода, в них введена относительная адресация. Для этих целей каждый произвольный процессор системы имеет относительную (внутреннюю) нумерацию остальных процессоров системы.

Относительный номер j -го процессора системы $n(i, j)$ в системе нумерации i -го процессора системы определяется из соотношения

$$n(i, j) = \begin{cases} i + j, & \text{при } i > j, \\ j - i - 1, & \text{при } i < j. \end{cases}$$

Этот номер преобразуется в адресные части инструкций ввода-вывода. В таблице 1 сведены относительные номера всех процессоров системы. По данной таблице можно определить, какие адреса должна установить произвольная пара процессоров для организации канала обмена.

Нумерация процессоров системы

I	1	2	3	4
J				
1			1	0
2	0		2	1
3	1	0		2
4	2	1	0	

Рассмотрим кратко функционирование предлагаемой системы коммутации в составе многопроцессорной системы. Как уже было упомянуто, для каждого процессора системы непосредственно из алгоритма задачи вытекает последовательность его обмена с другими процессорами.

К примеру, после распараллеливания задачи для процессора P(2) можно сформировали запись типа: {P3R, P1W, P4W}, которая будет означать, что процессор P(2), после активации в нем параллельного фрагмента общей программы, должен будет сначала принять сообщение от процессора P(3), затем послать сообщение в процессор P(1) и закончить обмен посылкой сообщения в процессор P(4).

Из этой записи следует, что для той части общей программы, которая исполняется процессором P(2), в некоторой её точке потребуются для дальнейших вычислений данные от процессора P(3). Для получения этих данных процессор P(2) осуществляет операцию ввода, в адресной части которой указывает относительный адрес процессора P(3). При этом процессор P(3) со своей стороны должен будет осуществить операцию вывода, в адресной части которой указать относительный адрес процессора P(2). Эти две операции должны совпасть во времени, но этого в общем случае не происходит.

В большинстве случаев управление операциями ввода-вывода возложено на контроллеры. Если для вывода информации её надо предварительно сформировать и только потом выполнять операцию вывода, то для операций ввода дело обстоит проще. Операцию ввода можно поставить в программе раньше той точки, в которой вводимые данные потребуются, но при этом ввести признак их наличия. В предлагаемой системе операцию ввода целесообразно вводить в программу по возможности раньше.

Обозначим через $A(i, j)$ относительный адрес процессора P(j) в адресном пространстве ввода-вывода процессора P(i).

Из таблицы 1 непосредственно видно, что $A(2, 3)=1$ и $A(3, 2)=2$. Следовательно, для осуществления информационного обмена между процессорами P(2) и P(3) необходимо следующее.

- Процессору P(2) исполнить операцию ввода, относительный адрес которой равен 1. По исполнении этой операции будет активизирован шинный формирователь ШФ(2, 3) и установлено направление передачи данных от шинного формирователя к процессору.

- Процессору P(3) исполнить операцию вывода, относительный адрес которой равен 2. По исполнении этой операции будет активизирован шинный формирователь ШФ(3,2) и установлено направление передачи данных от процессора к шинному формирователю.

При этом оба процессора выполняют свои операции асинхронно, механизм прерываний так же не используется. Канал обмена между двумя процессорами будет сформирован только тогда, когда будут активизированы оба шинных формирователя в соответствующем УСФ, после чего он функционирует как шина. Параллельно с рассмотренным выше каналом обмена может быть сформирован и функционировать другой канал обмена, между процессорами P(1) и P(4). Необходимо заметить, что процесс формирования каналов обмена является полностью асинхронным.

В общем случае в n-процессорной системе может параллельно функционировать любое произвольное число каналов обмена

$$q \leq n/2.$$

С позиций конструктивной целесообразности следует преобразовать приведенную выше структурную организацию системы коммутации следующим образом:

- Дешифраторы ввести в состав соответствующих процессоров.
- УСФ сформировать как единый неделимый узел.

На рисунке 2 приведена структура многопроцессорной системы, где распределенная система коммутации представлена 6-ю узлами УСФ.

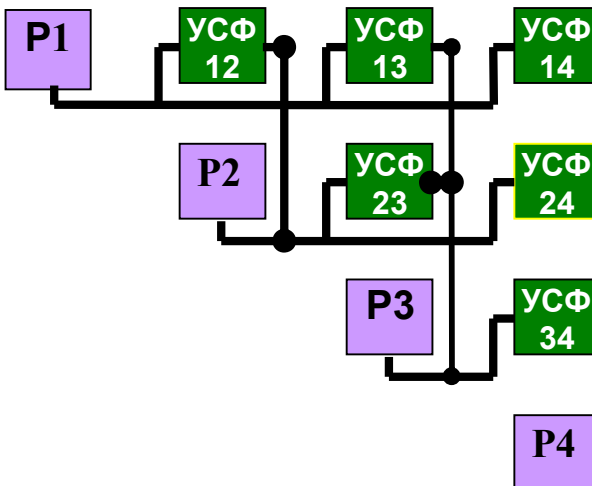


Рис. 2. Модифицированная структура системы коммутации

Свойства системы коммутации. Традиционно в системах коммутации применяются мультиплексоры и демультимплексоры, недостатком которых является наличие централизованного управления и однонаправленное движение информационных потоков.

Рассмотренная система коммутации информационных каналов лишена этих недостатков. Так каждая строка шинных формирователей может рассматриваться как демультимплексор, управляемый соответствующим процессором системы единолично,

Отличительной особенностью такого мультиплексора является возможность одновременной передачи информации любому произвольному числу процессоров системы. Те же самые шинные формирователи в других сочетаниях могут выполнять функции мультиплексоров. Информация в каждом шинном формирователе может протекать в обоих направлениях. Поэтому каждый УСФ может входить в состав и мультиплексора и демультимплексора.

В целом предложенная система коммутации является более универсальной, что дает возможность решать и такие важные проблемы, удовлетворительного решения которых не найдено. К таким проблемам относится проблема когерентности общих переменных при межпроцессорном обмене.

Традиционные системы коммутации в большей степени предназначены для осуществления обмена между пассивными элементами. Даже процессор в таком обмене является почти пассивным элементом. Его отличие от полностью пассивных элементов состоит в том, что он может запросить обмен с тем или иным устройством.

Как правило, традиционные коммутаторы в многопроцессорных системах используются для обмена процессоров с общей или распределенной памятью, в итоге, между собой процессоры общаются опосредованно, через память и общесистемного арбитра, которые, хотим мы этого или нет, выступают в качестве общесистемного ресурса. Последнее обстоятельство является отрицательным фактором, серьезно ограничивающим диапазон масштабируемости системы.

Предложенная и рассмотренная система коммутации рассчитана на взаимодействие активных субъектов информационного обмена, выполняющих общую задачу. Информационный обмен между двумя произвольными субъектами обмена может начаться незамедлительно, как только оба они выразят готовность к обмену, и осуществляется он без каких-либо посредников и промежуточных звеньев.

Имитационная одноразрядная модель УСФ, созданная в пакете Workbench, показала:

- функциональные возможности модели полностью обеспечивают реализацию поставленной задачи – передача информации только в одном из двух возможных направлений;

- простоту структурной организации устройства.

Кроме того, она позволяет достаточно точно оценить аппаратную сложность системы коммутации в целом.

1. *Корнеев В.* Следующее поколение суперкомпьютеров / В. Корнеев // Открытые системы, 2008. – № 8.
2. *Черняк Л.* Многоядерные процессоры и грядущая параллельная революция / Л. Черняк // Открытые системы, 2007. – № 4.
3. *Hwang K., Tseng P.S., Kim D.* An Orthogonal Multiprocessor for Parallel Scientific Computation / Hwang K., Tseng P.S., Kim D. // IEEE Transaction on Computers, January 1989. - 38, № 1. – P. 120-124.
4. *Сигарев А.А., Душеба В.В.* Методы организации ортогональных многопроцессорных систем / А.А. Сигарев, В.В. Душеба // Электронное моделирование, 1999. – т. 21, 1. - № 1. – С. 47-57.