

метода распараллеливания DO-циклов. Предложен подход к построению нового тела цикла с минимальным числом дополнительных операторов присвоения, что позволяет получать в результате распараллеливания более эффективный цикл.

Показана применимость координатного метода к конвейерной ЭВМ. Кроме того, расширены возможности его использования на таких компьютерах с учетом обработки рекуррентных соотношений.

Предложены способы ускорения работы алгоритма, позволяющие избежать полного перебора возможных комбинаций распараллеливаемых циклов.

Перечисленные возможности по развитию метода были использованы и апробированы в программной разработке по автоматическому распараллеливанию циклов для процессора А-12* [4]. Опыт обработки реальных циклов для этого процессора показывает, что отсутствующие индексы в пользовательских программах — довольно частое явление. Построение тела цикла, как правило, ведет к появлению не более трех дополнительных операторов.

ЛИТЕРАТУРА

1. Lamport L. The parallel execution of DO loops // Comm. of ACM.— 1974.— V. 17, N 2.
2. Хокни Р., Джесхоун К. Параллельные ЭВМ.— М.: Радио и связь, 1986.
3. Белкин А. Р. Приближенная триангуляция матриц в задачах ранжирования и обработки межотраслевого баланса // Техн. кибернетика.— 1981.— № 1.
4. Бродский И. И. и др. Высокопроизводительный периферийный векторный процессор А-12 // Вопр. кибернетики.— 1985.— № 104.

Поступила в редакцию 21 апреля 1987 г.

УДК 681.3.06

Р. ВЕРБОВА, А. НЯГОЛОВ, П. ТАНЕВА, А. ТРЕНЕВ

(София, Болгария)

ПОДХОДЫ К УПРАВЛЕНИЮ СПЕЦИАЛИЗИРОВАННЫМИ АРИФМЕТИЧЕСКИМИ ПРОЦЕССОРАМИ

Введение. Микрокомпьютерные системы автоматизации занимают в последние годы центральное место в области автоматизации научных исследований, управления непрерывными производственными процессами, САПР и т. д. Разнообразие и непрерывное совершенствование элементной базы приводят к созданию разных типов систем автоматизации в соответствии с конкретными требованиями применений. Использование современных магистрально-модульных интерфейсов обеспечивает их гибкую структуру. Основным требованием к таким системам, работающим, как правило, в реальном масштабе времени, является быстрдействие. Функции управления чаще всего связаны с выполнением большого числа вычислительных операций в широком диапазоне входных данных и выходных реакций. Вычислительные возможности микропроцессоров, однако, весьма ограничены. Все это приводит к необходимости включения

* Публикация по разработке планируется во втором номере журнала «Автоматика» за 1988 г.



Рис. 1

в структуру систем автоматизации специализированных арифметических устройств (арифметических процессоров).

Наиболее распространенные микропроцессорные наборы содержат интегральные схемы для быстрых арифметических операций, ориентированные на подключение к определенной микропроцессорной магистральной и работающие под управлением соответствующего микропроцессора. В последнее время проводятся исследования эффективности использования арифметических устройств, предметом которых являются разные варианты подключения к микропроцессорной конфигурации, анализ их преимуществ и недостатков, проблемы создания программного обеспечения.

Способы включения специализированных арифметических процессоров в структуру систем автоматизации. Отметим несколько вариантов подключения арифметических устройств к модульным системам автоматизации.

Чаще всего встречается так называемый конвенциональный способ подключения: основной микропроцессор (МП) непосредственно управляет арифметическим устройством (АЛУ) через микропроцессорную магистраль (рис. 1). Главным преимуществом этого способа является простота реализации. Основным недостатком состоит в том, что на управление арифметическим устройством уходит большая часть времени микропроцессора.

Подобные проблемы возникают и при подключении арифметического процессора к магистрально-модульным структурам типа КАМАК [1]. Существуют реализации [2], в которых управление арифметическим процессором осуществляется КАМАК-командами (рис. 2). В данном случае подлежащие обработке данные и соответствующая управляющая инфор-

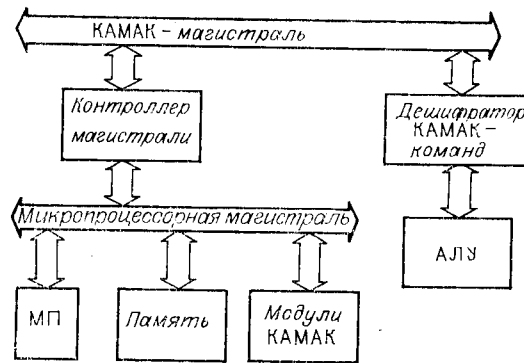
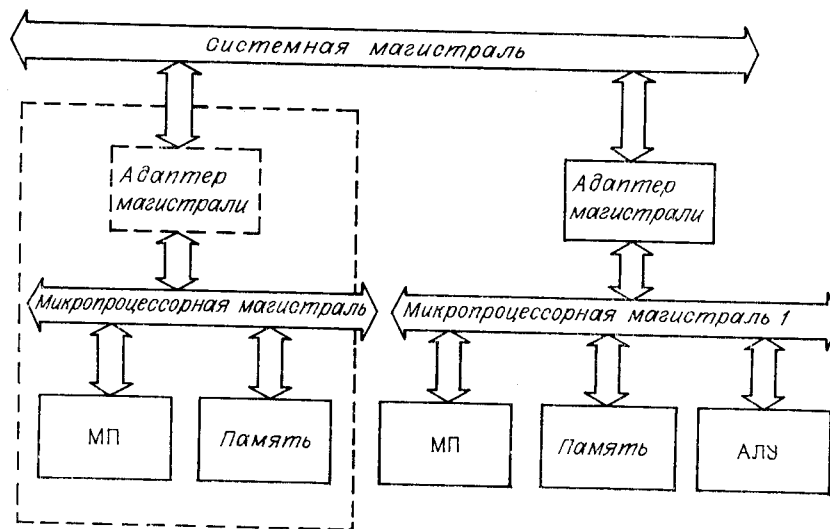


Рис. 2



Основная микрокомпьютерная конфигурация

Рис. 3

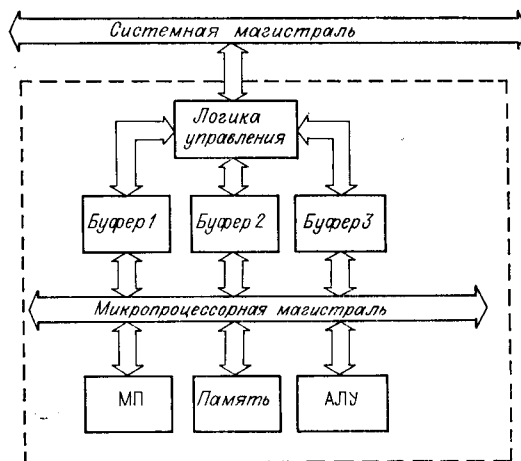


Рис. 4

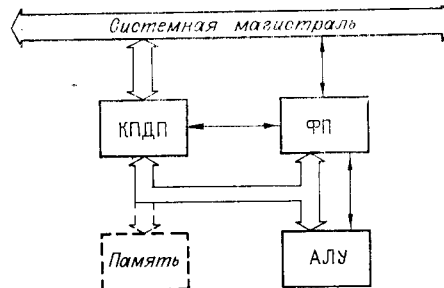


Рис. 5

мация передаются по шинам данных КАМАК-интерфейса. Быстродействие этой передачи зависит и от конкретного решения модуля в КАМАК-системе.

Следующим шагом в области применения арифметических процессоров является создание специализированных модулей для арифметических вычислений. По существу арифметический модуль представляет собой микрокомпьютерную систему с конвенциональным способом подключения арифметического процессора (рис. 3). Модуль можно подключать к магистрально-модульным структурам с помощью адаптера для конкретной системной магистрали (ВМЕ, КАМАК и др.). Преимущества этого способа связаны с сокращением времени на обслуживание арифметического модуля со стороны основного микропроцессора системы. Функции обслуживания сводятся к загрузке обрабатывающих программ и данных в модуль и к получению конечных результатов. Наличие микропроцессора в арифметическом модуле позволяет реализовать некоторые диагностические функции самого модуля, например выполнение встроенных тестовых процедур во время, свободное от заданий и т. д.

Существуют варианты этого способа подключения, увеличивающие эффективность работы арифметического модуля. Известны реализации арифметических модулей с несколькими буферами данных и результатов, что дает возможность совмещать процедуры обработки данных с процедурами обмена данными (рис. 4). В некоторых решениях продолжительность операций передачи внутри модуля снижается путем использования микропрограммного автомата для непосредственного управления арифметическим устройством [3]. В этом случае функции встроенного микропроцессора сводятся только к синхронизации работы микропрограммного автомата, к обеспечению обмена с основной микрокомпьютерной конфигурацией и к выполнению диагностических процедур, т. е. используется весьма ограниченная часть возможностей микропроцессора. Существуют конкретные решения, позволяющие минимизировать структурные схемы арифметических модулей путем предоставления функций синхронизации основному микропроцессору и создания функционально специализированного процессора (ФП), осуществляющего, кроме управления арифметическим процессором, управление каналом прямого доступа к памяти (КПДП) основной системной конфигурации (рис. 5).

Проблемы создания программного обеспечения. Основной целью программного обеспечения для управления специализированными арифметическими процессорами является осуществление обмена между основным микропроцессором и специализированным арифметическим процессором с минимальными затратами времени. Протокол обмена, однако, непосредственно зависит от аппаратной реализации и от степени «интеллигентности» арифметического модуля. Использование магистрали для передачи каждой команды в арифметический модуль и для получения каждого (в том числе и промежуточного) результата сразу становится «узким местом» и снижает быстродействие в несколько раз. При исполь-

зовании специализированных арифметических модулей нужно предусматривать и создание резидентного программного обеспечения, реализующего обмен информацией с основной микрокомпьютерной системой. Заслуживает внимания способ организации обмена, который позволяет использовать встроенный в арифметическом модуле микропроцессор только для подготовки данных и передачи конечных результатов в основной микропроцессор, а функции непосредственного управления арифметическим процессором предоставить микропрограммному автомату.

При разработке программного обеспечения надо учитывать и фактор общения пользователя с системой. В этом смысле программные средства управления арифметическими модулями должны в максимальной степени облегчать доступ к арифметическому процессору со стороны пользователя, сохраняя обычный вид записи арифметических операторов в используемых языках программирования и обеспечивая системные средства генерации с одного и того же исходного кода объектных файлов как

Управление специализированными арифметическими процессорами с помощью средств Ассемблеров и языков высокого уровня. Основной функцией внутреннего уровня (ядра) программных средств управления арифметическими процессорами являются управление обменом информацией с основной микрокомпьютерной системой, запуск вычислительных процедур в соответствии с генерированными уже кодами операций, обработка исключительных ситуаций или передача управления пользовательским программам. На внутреннем уровне должны учитываться и архитектурные особенности подключения арифметических процессоров к системе, наличие специализированных модулей и их конкретная структура. Прямое отношения к пользователю ядро не имеет.

Отсюда следует, что перечисленные выше функции внутреннего уровня программного обеспечения делают его в максимальной степени зависимым как от конкретного типа арифметического процессора, так и от конкретной структуры системы автоматизации. Эту зависимость нельзя обойти, и ядро для каждого типа систем автоматизации приходится создавать заново. Учитывая характерные особенности внутреннего уровня и требования в отношении оптимальности и быстродействия, эту часть программного обеспечения целесообразно разрабатывать на Ассемблере и помещать в постоянную память типа PROM [4].

В процессе разработки следующих уровней программного обеспечения бывает целесообразно использовать в качестве ядра программные арифметические пакеты. Такой подход полезен при создании унифицированных программных продуктов, предусматривающих работу как со специализированным арифметическим модулем в составе системы, так и без него.

Следующий уровень программных средств управления специализированными арифметическими процессорами зависит от пользователя и должен отвечать некоторым требованиям в отношении «дружественности» к непрофессиональному программисту. Обычно на этом уровне используется Ассемблер на основе макроопределений, которые обеспечивают реализацию команд управления арифметическим процессором [5].

Расширенные Ассемблеры строятся по модульному принципу и обычно включают несколько логически обособленных программных единиц. Почти во всех случаях возникает необходимость в создании программного модуля предварительного прохода, во время работы которого на основе исходного текста программы производится обработка определений констант с плавающей запятой, подключается макроаппарат реализации команд управления арифметическим процессором и формируется соот-

ветствующий файл, который обрабатывается обычным Ассемблером. Главная функция этого модуля — генерация программного кода, соответствующего типу выбранного спецпроцессора. Нужно отметить, что на создании расширенных Ассемблеров уже не сказывается зависимость от архитектуры системы и способа подключения арифметического процессора. Таким образом, этот уровень является универсальным, и составляющие его программные средства можно мультиплицировать в широком диапазоне применений.

С точки зрения пользователя программные средства этого уровня воспринимаются как Ассемблеры с дополнительными командами для осуществления арифметических операций с плавающей запятой. Мнемоника и синтаксис дополнительных команд полностью подчиняются синтаксическим правилам соответствующего Ассемблера. Тем не менее программные средства такого типа требуют знания Ассемблера, что ограничивает распространенные алгоритмические языки (Фортран, Паскаль) используют библиотечные структуры для реализации арифметических операций: во время работы транслятора проводится анализ операторов языка и при необходимости генерируется последовательность обращений к библиотеке программной арифметики. Поскольку библиотечные подпрограммы включаются в программный код на этапе его сборки, появляется возможность заменить библиотеку программной арифметики библиотекой управления арифметическим процессором, модули которой генерировали бы программный код внутреннего уровня.

Преимущества такого подхода заключаются в его универсальности: не возникает необходимость в модификации трансляторов; с одной уже транслированной программы можно получить путем связывающего редактирования как программный код с обычной программной арифметикой, так и программный код, управляющий арифметическим процессором. Таким образом, проблемы по адаптации уже существующих прикладных программных продуктов полностью отпадают.

Нужно, однако, иметь в виду, что генерируемые трансляторами последовательности обращений к арифметическим подпрограммам, как правило, не учитывают связь между отдельными арифметическими операциями в одном операторе. Иными словами, промежуточные и конечные результаты обрабатываются одинаковым способом. Этот факт заведомо снижает быстродействие специализированных модулей из-за резкого возрастания операций обмена между управляющим микропроцессором и арифметическим процессором. Следовательно, использование такого подхода оправдано только при конвенциональных способах подключения арифметического процессора.

Если же аппаратное управление арифметическим процессором предусматривает наличие микропрограммного автомата, приходится либо использовать уровень языка Ассемблера, где всю ответственность за операции обмена несет прикладной программист, либо модифицировать трансляторы, что сразу приводит к потере универсальности.

Достигнутые результаты и применения. С 1981 г. в Центральной лаборатории по автоматизации и научному приборостроению БАН ведутся работы по включению специализированных арифметических процессоров в состав микрокомпьютерных магистрально-модульных систем автоматизации в стандартах КАМАК и ВМЕ. Созданы как простые специализированные арифметические модули в стандарте КАМАК, управляемые КАМАК-командами, так и арифметические модули со встроенными микропроцессорами. Разработаны и варианты с микропрограммным управлением арифметическим процессором.

В области программного обеспечения разработаны внутренние уровни для комбинаций 8-битовых и 16/32-битовых микропроцессоров со специализированными арифметическими процессорами, созданы расширенные Ассемблеры для 8-битовых и 16/32-битовых микропроцессоров. Существующие трансляторы Фортран и F77 снабжены библиотечными пакетами для управления специализированными арифметическими процессорами.

Все эти работы направлены на создание мощных прикладных систем автоматизации в разных областях научных исследований и народного хозяйства. Разработаны система анализа состава газов в реальном масштабе времени для конверторного производства стали, система адаптивного управления процессом сгорания угля в парогенераторе ТЭС; ведутся работы по созданию систем машинной графики, обработки изображений и т. д.

Создание технических и программных средств управления специализированными арифметическими процессорами дает возможность повысить вычислительную мощность магистрально-модульных систем автоматизации и в результате расширить сферу их применения.

ЛИТЕРАТУРА

1. Тренев А. Система КАМАК // Автоматика и вычислительная техника.— София.— 1982.— № 6.
2. Челебиев А. Арифметические модули ЦЛАНП // II Междунар. симп. «Автоматизация и научное приборостроение, 83».— Варна, 1983.
3. Trenev A., Varbova R., Njagolov A. Autonomous arithmetic module for microcomputer configurations // II Symposium on Microcomputer and Microprocessor Applications "Microprocessor's 85".— Budapest, 1985.
4. Аврамов И., Няголов А., Атанасова Л. Подход при вычислении арифметических выражений в микрокомпьютерных системах // I Междунар. симп. «Автоматизация и научное приборостроение, 81».— Варна, 1981.
5. Танева П., Вербова Р. Арифметика с плавающей запятой на уровне Ассемблера для микрокомпьютерной системы ИНТЕРЛАБ // II Междунар. шк. по АНИ.— Пущино, СССР, 1985.
6. Вербова Р., Няголов А. Алгоритмический язык FORTRAN с библиотекой плавающей арифметики для МСА ЦЛАНП 0270, расширенной специализированным арифметическим модулем // III Междунар. симп. «Автоматизация и научное приборостроение, 85».— Варна, 1985.

Поступила в редакцию 23 декабря 1986 г.

УДК 681.3 : 621.3

Ю. Н. МАТВЕЕВ, Е. Ф. ОЧИН

(Ленинград)

ВЫПОЛНЕНИЕ ОПЕРАЦИИ СКОЛЬЗЯЩЕГО ВЫРАВНИВАНИЯ ГИСТОГРАММЫ В МАТРИЧНОМ ПРОЦЕССОРЕ

Еще в первых исследованиях по обработке изображений с помощью ЭВМ было установлено, что матричные процессоры, структура которых отображает структуру обрабатываемых данных, могут стать мощным средством обработки изображений [1]. К настоящему времени разработан и изготовлен ряд матричных процессоров таких, как ILLIACIII, CLIP4, DAP [2], MPP [1], ПС-3000 и другие. Основной проблемой при их использовании является разработка алгоритмов и соответствующих программ, эффективно реализуемых этими матричными процессорами.

В ряде исследований показано, что наиболее эффективно на матричных процессорах реализуются операции обработки двумерных массивов данных, обладающих свойством локальности: свертка с малоразмерным