

ОРГАНИЗАЦИЯ РАСПРЕДЕЛЕННОЙ СИСТЕМЫ УПРАВЛЕНИЯ НА ОСНОВЕ МИКРОПРОЦЕССОРНЫХ МОДУЛЕЙ

Н.Н. Алексеев, В.П. Пискунов, А.С. Рябцев

Институт теоретической и экспериментальной физики, Москва

Введение

Настоящая работа посвящена вопросам совершенствования действующих автоматизированных систем управления, построенных на основе мини-ЭВМ и традиционной структуры интерфейсов типа КАМАК-ВЕКТОР.

На примере автоматизированной системы управления протонным синхротроном ИТЭФ У-10 рассматривается один из возможных вариантов построения распределенной системы управления ускорительным комплексом с помощью микропроцессорных модулей. Особенностью предлагаемой архитектуры является простота аппаратуры, а также возможность эволюционного совершенствования действующей системы.

Структура интерфейсов АСУ У-10 / 1, 2/ представляет собой достаточно разветвленную сеть, которая объединяет несколько десятков интерфейсных каркасов и управляет от двух связанных межмашинной связью ЭВМ типа ЕС-1010. Интерфейсные каркасы удалены от ЭВМ на расстояние до 200 м и размещаются в непосредственной близости от контролируемого оборудования ускорителя. АСУ У-10 обеспечивает автоматизированное управление и контроль нескольких технологических систем ускорителя (магнитной оптики ионопровода, системы инжекции, системы коррекции магнитного поля, системы синхронизации, системы управления радиочастотой ускоряющего ВЧ- поля, вакумной системы), а также диагностику пучка на выходе инжектора, в ионопроводе и в кольце. Объем информации, вырабатываемой в каждом цикле ускорения аппаратурой автоматизированного контроля оборудования У-10 и диагностики пучка, составляет в настоящее время около 30 кбайт, более двухсот параметров систем ускорителя управляются и оптимизируются через ЭВМ.

Эффективность использования аппаратных средств АСУ У-10 ограничивается производительностью процессоров ЭВМ ЕС-1010. При трехсекундном цикле ускорения значительная часть процессорного времени (около 1,5 с) тратится на получение, обработку и вывод на экраны дисплеев минимального объема оперативной информации, остальное время распределяется между тремя-четырьмя задачами по выбору обслуживающего персонала ускорителя, включая задачи автоматической оптимизации режимов, выборочного или непрерывного контроля параметров систем, громоздких измерений, тестирования и калибровки аппаратуры, подготовки данных, расчетов и т.п.

При создании интерфейсной структуры АСУ У-10 предполагалось, что на определенном этапе развития автоматизированной системы для управления и контроля отдельных информационных ветвей будут применены микропроцессорные контроллеры (МПК), которые заменят пассивные блоки управления интерфейсными каркасами.

В настоящее время разработка МПК завершена и они начинают использоваться для выполнения отдельных функций оперативного управления и контроля параметров синхротрона У-10. За счет применения МПК в АСУ У-10 мы предлагаем уменьшить информационную нагрузку основных процессоров и соответственно расширить функциональные возможности автоматизированной системы.

Основное назначение МПК заключается в выполнении функций непрерывного контроля и управления подсистемами ускорителя, обеспечения предварительной обработки и преобразования информации, передаваемой от подчиненных модулей в основные ЭВМ и обратно, а также автономном управлении подчиненными модулями в режиме

прямого интерактивного взаимодействия с оператором. Другие функции МПК мы связываем с организацией на их основе межкаркасных информационных магистралей, которые позволят перераспределить задачи контроля и управления распределенным оборудованием ускорителя между основными процессорами АСУ и микропроцессорными модулями.

Для эффективного функционирования многопроцессорной АСУ необходимы адекватные средства программирования. Проблему реализации средств программирования, удовлетворяющих заданным требованиям, мы предполагаем решить на основе используемой на протяжении ряда лет в АСУ У-10 интерпретирующей системы м'Нодал.

Аппаратура

В качестве интерфейсов в АСУ У-10 применяется стандарт "Вектор", а также многоуровневый специализированный интерфейс, выполненный в конструктиве "Вишня" / 2/.

На основе микропроцессорной (МП) серии К580 для управления каркасом "Вектор" был разработан МП контроллер (МПКВ), заменяющий стандартный блок местного управления (БУМ). Блок МПКВ выполняет все основные функции БУМ и обеспечивает асинхронный доступ к информационной магистрали каркаса со стороны ветви "Вектор" и порта в/в МП. Для управления МП используются адреса "Вектор", зарезервированные для управления контроллером (N25+M31). Блок обрабатывает до 8 внешних прерываний: два - со стороны ветви "Вектор", шесть - со стороны лицевой панели. Команды управления МПКВ обеспечивают инициацию МП (бросок всех регистров с передачей управления по нулевому адресу), прерывание МП, а также прямой доступ к памяти МП. Типовая конфигурация блока МПКВ включает 2+16 кбайт ППЗУ и ОЗУ до 48 кбайт. На лицевой панели блока установлен многоконтактный разъем расширения системы, который может использоваться для подключения дополнительных внешних устройств.

Для управления стандартным мультиплексорным каркасом специалитета АСУ У-10 "Вишня" разработан на основе МП К580 микропроцессорный блок связи (МБС1), заменяющий пассивный блок связи с мультиплексорным каркасом (БСМК) с сохранением всех первоначальных функций. Структура, логика управления и конфигурация МБС1 аналогична МПКВ, отличие - в конструктиве исполнения, а также в схемах, выполняющих роль адаптеров с интерфейсными магистралями.

Первые образцы разработанных блоков изготовлены и находятся в опытной эксплуатации.

Один из блоков МБС1 используется для сбора и предварительной обработки оперативной информации, а также для накопления статистических данных о работе ускорителя. Перечисленные функции в ограниченном объеме выполнялись в одной из двух мини-ЭВМ ЕС-1010 и занимали заметную часть процессорного времени.

Конфигурация измерительной системы, работающей под управлением блока МБС1, показана на рис. I. Основная программа, выполняемая МП, контролирует параметры пучка в ионопроводе, обрабатывает оцифрованные динамические сигналы тока циркулирующего пучка и интенсивности ускоряемого пучка, по данным измерителей рассчитывает значения частоты и крутизны ускоряющего ВЧ- поля на момент инъекции. Обработанная информация заносится в таблицу оперативной информации, которая формируется в ОЗУ МП и содержит отмасштабированные значения контролируемых параметров за десять последних циклов ускорения, а также статистические данные по основным эксплуатационным параметрам ускорителя за последнюю минуту, час, смену, весь период от начала сеанса. Вся накапливаемая таким образом информация периодически считывается в основную ЭВМ для сохранения на внешнем носителе информации либо отображения на экране дисплея. Основная ЭВМ может

инициировать дополнительно к основной программе параллельное выполнение в МП нескольких программ (до четырех).

Для того чтобы возложить на МП выполнение описанной выше задачи, в подчиненных одному МБСИ каркасах собраны все необходимые измерительные модули. При этом остается проблема представления информации: блоки привязки средств отображения информации либо должны размещаться в тех же каркасах, либо функции вывода должны выполняться в основной ЭВМ. И тот, и другой вариант не представляется оптимальным: первый вариант приводит к ограниченному применению средств отображения информации, второй - связан с затратами времени основного процессора.

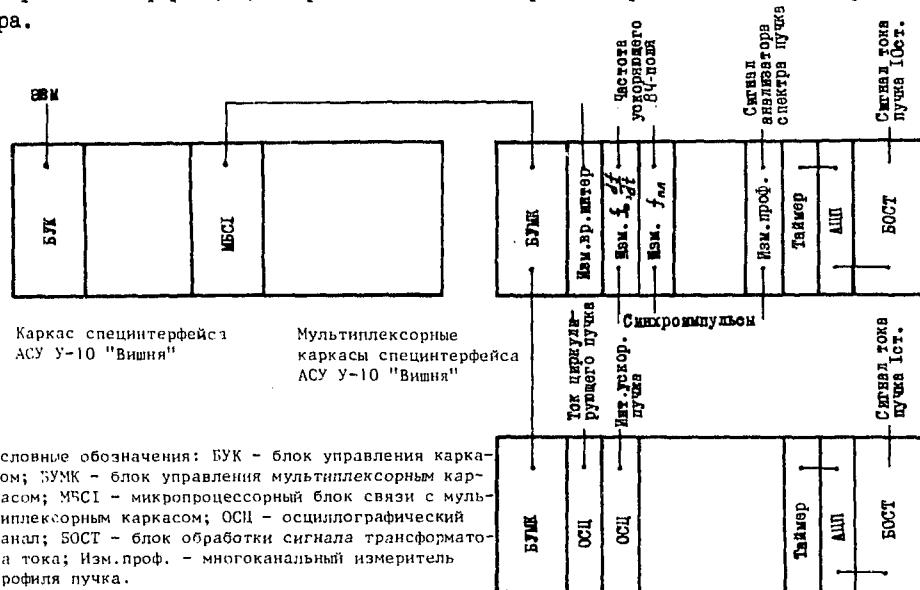
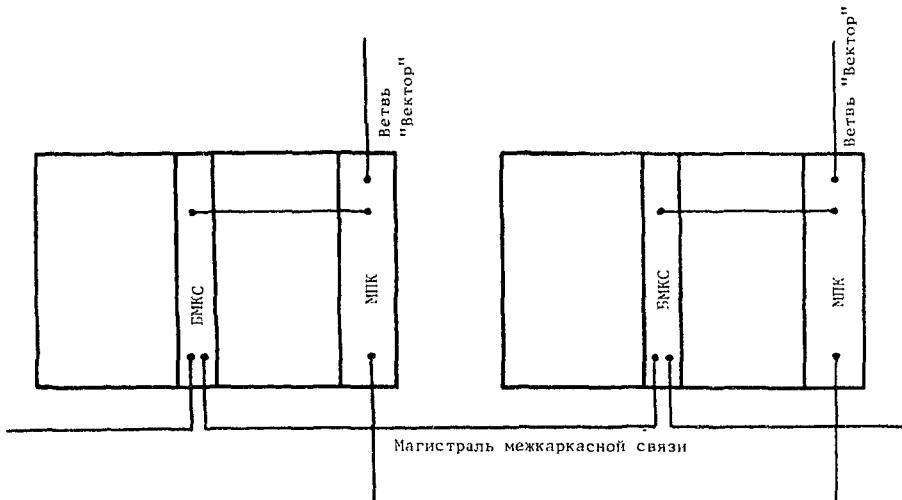


Рис. I. Схема измерительной системы на основе микропроцессорного контроллера.

Для расширения функциональных возможностей МПК была разработана схема межкаркасной связи (МКС), позволяющая МПК разных интерфейсных каркасов обмениваться информацией по линиям прямой связи, минуя центральные процессоры. Блоки МКС занимают место одиночной ширины в каркасах с МПК и соединяются между собой по лицевой панели 10-парным кабелем, по которому реализуется побайтная передача информации (рис. 2). Каждый блок МКС подключается также перемычкой по лицевой панели к одному из уровней прерывания своего МПК. Функции управления процессом приема/передачи информации возлагаются на прерываемую программу МПК. Один из блоков МПК выполняет функции арбитра, являющегося одновременно посредником при обмене информацией между другими МПК. Арбитр может одновременно обслуживать несколько линий МКС, пересекающихся в его каркасе. Распределение функций между МПК осуществляется основная ЭВМ. Реальная скорость обмена информацией между МПК зависит от количества связанных контроллеров и интенсивности обмена. В режиме диалога двух блоков обеспечивается скорость передачи данных до 100 кбайт/с.

Эффективность использования МКС зависит от производительности МПК и имеющихся средств программирования. На У-10 мы предполагаем использовать МКС для выполнения МП функций распределенного контроля и управления, а также для прямого вывода графической информации на устройства отображения центрального пульта.



Условные обозначения: МПК - микропроцессорный контроллер;
БМКС - блок межкаркасной связи.

Рис. 2. Схема межкаркасной связи.

Последняя задача особенно актуальна в связи с развитием техники оцифровывания динамических сигналов (цифровых осциллографических каналов) и связанным с этим значительным возрастанием объема графической информации.

Программное обеспечение

На современном этапе использования МП в многопроцессорных системах управления ускорительными комплексами задачи программирования МП решаются традиционными крос-средствами последовательного программирования, включающими трансляторы распространенных языков (Ассемблера, Бейсика, Фортрана и др.), поддерживаемых библиотеками стандартных программ, простейшими мониторами и операционными системами.

Дальнейшее развитие многопроцессорных АСУ связывается с реализацией языковых средств параллельного программирования, которые позволили бы упростить программирование параллельных информационно-связанных процессов, выполняемых многочисленными МП в реальном масштабе времени.

Средства параллельного программирования начали развиваться сравнительно недавно. Пожалуй, первым языком параллельного программирования можно считать Нодал, разработанный в 1974 году для многопроцессорной автоматизированной системы управления ускорителем SPS (ЦЕРН). В настоящее время синтаксические возможности программирования параллельных процессов введены в несколько языков: модификации Паскаля, Модула-2 /5/, ОКРАМ /6/, Ада /7/. Наибольшие усилия были связаны с разработкой языка Ада, который в настоящее время считается наиболее перспективным.

Применение современных языков программирования высокого уровня в многопроцессорных системах ограничивается отсутствием удобных реализаций языков, адекватных архитектуре автоматизированной системы. Разрыв между возможностями аппаратуры АСУ и программным обеспечением (ПО) пока сократить не удается, несмотря на все большие средства, вкладываемые в разработку ПО. Реально оценивая сложившуюся ситуацию, можно надеяться на доступность первых стандартных языковых средств параллельного программирования многопроцессорных систем только

ко через несколько лет, поэтому в настоящее время приходится искать собственные пути реализации таких возможностей.

В АСУ У-10 в качестве основного средства программирования прикладных задач на протяжении нескольких лет используется интерпретирующая система м'Нодал (модификация языка Нодал, разработанная для ЭВМ ЕС-1010). Средства параллельного программирования м'Нодала реализованы четырьмя командами (**IMEX**, **EXEC**, **REMIT, WAIT**), обеспечивающими из ведущей программы распределение сателлитных подпрограмм между несколькими процессорами, параллельное выполнение ведущей программы и всех сателлитных подпрограмм в реальном масштабе времени, передачу ведущей программе заказанной информации, синхронизацию выполнения операций ведущей программы со всеми сателлитными. Схема взаимодействия выполняемых параллельно процессов показана на рис. За, и требует наличия в каждом процессоре интерпретатора базового языка, на котором программируются распределенные задачи.

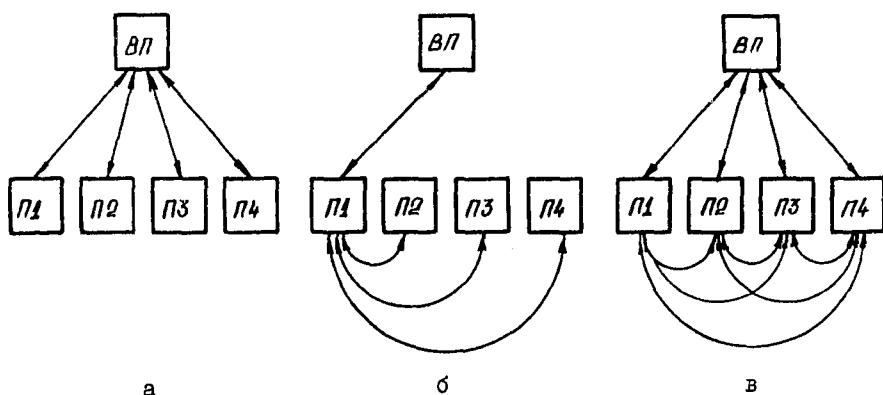


Рис. 3. Варианты взаимодействия параллельных процессов.

Такой базовый язык, представляющий из себя сокращенную версию м'Нодала, разрабатывается в настоящее время на У-10 для применения в МП. Применяемую схему параллельного программирования предполагается усовершенствовать для приведения ее в соответствие с развивающейся архитектурой АСУ: сателлитные подпрограммы смогут инициировать новые параллельные процессы (рис. 3б) и обмениваться информацией между собой (рис. 3в). Функции межпроцессорного обмена информацией мы сохраняем полностью за интерпретаторами, которые обеспечивают наглядность и логическую простоту всех применяемых процедур. Двоичные программные модули для МП, получаемые в результате трансляции, представляются как данные интерпретируемых программ как в основных процессорах АСУ, так и в МП. Такое представление позволяет легко манипулировать с объектными модулями МП средствами интерпретатора, включая компоновку сложных программ, их загрузку, тестирование и т.п. Совместное использование в МП интерпретируемых и транслируемых программ позволяет обеспечить необходимые скорости выполнения операций при управлении в реальном масштабе времени быстрыми процессами, а также сохранить все сервисные возможности и простоту восприятия программного обеспечения, свойственные интерпретаторам.

Заключение

Рассмотренные принципы организации распределенного управления не связывают с определенным типом МП. Проводимые в настоящее время проработки элементов системы на основе МП К580 позволяют проверить основные идеи, отредактировать ошибки и приобрести необходимый опыт.

Л и т е р а т у р а

1. Алексеев Н.Н. Препринт ИТЭФ, № II7, М., 1983.
2. Алексеев Н.Н. и др. В сб.: Труды девятого Всесоюзного совещания по ускорителям заряженных частиц, т. I, с. 264, Дубна, 1985.
3. Crowley-Milling M.C. Distributed digital control of accelerators. CERN LEP-DI/83-52.
4. Виноградов В.И. Информационно-вычислительные системы. М.: Энергоиздат, 1986.
5. Wirth N. Modula - 2, Bericht der Instituts für informatic, Technische Hochschule, Zürich, Dec. 1978.
6. Taylor R. and Wilson P. Process oriented Language meets demands of Distributed Processing, Electronics, Nov. 1982.
7. Вегнер П. Программирование на языке АДА. М.: Мир, 1983.