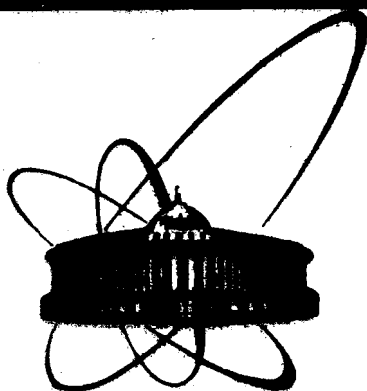


89-787



**СООБЩЕНИЯ
ОБЪЕДИНЕННОГО
ИНСТИТУТА
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ
ДУБНА**

Л 33

P10-89-787

М. Лебнер, А. И. Островной

**ОРГАНИЗАЦИЯ ПРОГРАММ СВЯЗИ
В РАСПРЕДЕЛЕННОЙ СИСТЕМЕ
АВТОМАТИЗАЦИИ ЭКСПЕРИМЕНТОВ
НА СПЕКТРОМЕТРЕ
ПОЛЯРИЗОВАННЫХ НЕЙТРОНОВ**

1989

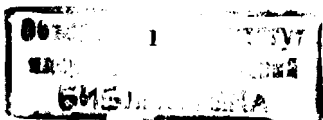
1. Введение

Развитие методики проведения современного физического эксперимента, усложнение экспериментальных установок и используемой электронной аппаратуры приводит к тому, что увеличивается поток регистрируемой экспериментальной информации, увеличивается количество регистрируемых и управляемых параметров, более сложными становятся объекты управления, причем зачастую они могут находиться на значительном расстоянии друг от друга. Во время проведения эксперимента и по его окончании требуется более сложная обработка данных. В достаточно сложных системах автоматизации экспериментов с помощью одной ЭВМ выполнение всех необходимых операций обеспечить не всегда возможно. Одним из методов комплексного решения задач автоматизации сложных установок является создание распределенных систем. В настоящее время их разработка составляет одно из быстро развивающихся направлений в области автоматизации ядерно-физических экспериментов.

В данной работе на примере распределенной системы автоматизации экспериментов для спектрометра поляризованных нейтронов СПН-1 [1] обсуждаются вопросы создания программного обеспечения связи. Из особенностей этой системы отметим следующие:

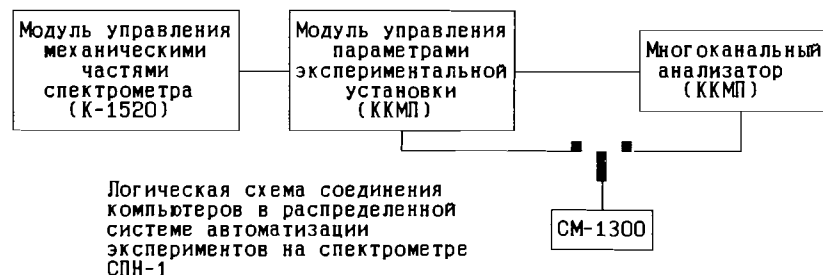
- 1) совокупность связанных компьютеров используется для решения одной задачи;
- 2) система делится на ряд взаимодействующих подсистем, каждая из которых выполняет определенный набор функций самостоятельно, а часть функций во взаимодействии с другими подсистемами;
- 3) управление имеет децентрализованный характер.

Такая организация системы автоматизации позволяет увеличить надежность работы наиболее сложных установок, сочетать возможность управления и работы подсистемы в автономном режиме и в качестве части одной большой системы. Это важно на различных этапах разработки или модификации системы, а также для наладки аппаратуры перед измерениями, контроля правильности ее работы во время эксперимента, иногда для автономной работы с определенной подсистемой. Помимо этого такая распределенная система может быть включена в качестве нижнего уровня более обширной иерархической системы.



2. Общая организация системы

Система автоматизации экспериментов на СПН-1 построена на базе соединенных линиями связи микро-ЭВМ СМ-1300, К-1520 [1], микропроцессорных контроллеров КАМАК - ККМП [2] и аппаратуры КАМАК. Логическая схема соединения компьютеров в системе представлена на рисунке.



Содержание прикладных операций, выполняемых в системе различными ЭВМ, описано в работе [1]. Отметим также, что каждая из подсистем, изображенная на схеме прямоугольником, выполняет вполне определенный набор прикладных операций, исходя из состава подключенной к ней аппаратуры и возложенных функций. Любая подсистема может работать автономно и в комплексе с остальными подсистемами. На уровне прикладного программирования можно предусмотреть их взаимодействие. Основная программа эксперимента может работать на одном из ККМП (обычно в многоканальном анализаторе, так как состав его аппаратуры позволяет выполнить основную функцию в эксперименте - накопление и сохранение спектрометрической информации). Автономная работа подсистем может осуществляться в режиме удаленного управления через линию связи либо управления от непосредственно подключенного терминала.

Для создания прикладных программ на ККМП для проведения конкретных экспериментов пользователю предоставляется интерпретируемый язык МСЛ [3]. Все операции для управления экспериментальным оборудованием и обеспечения взаимодействия подсистем реализованы в виде отдельных команд МСЛ.

Рассмотрим способ реализации операций, при выполнении которых используются линии связи, а также основные принципы организации и взаимодействия соответствующих программ.

3. Программы связи ККМП и К-1520

Микро-ЭВМ К-1520 соединена с ККМП через параллельный 16-разрядный интерфейс и работает в подчиненном режиме как достаточно сложное исполнительное устройство. Для К-1520 создана специальная программа, которая записана в ППЗУ и позволяет управлять механическими устройствами спектрометра. Команды управления эта программа в рабочем режиме принимает из линии связи от соответствующего ККМП. Каждая команда закодирована в виде пакета длиной 4 байта (размер пакета всегда фиксирован). Формат передаваемого пакета приведен в приложении 1. Ответ об удачном или неудачном выполнении заданной команды программа на К-1520 пересылает по линии связи обратно в виде пакета такой же структуры и размера. Если произошла ошибка в процессе выполнения, то в ответном пакете содержится код ошибки.

Операции управления механическими устройствами спектрометра предусматриваются в программе эксперимента, которая выполняется на ККМП модуля управления параметрами. Эти операции реализованы в виде интерпретируемых команд в системе МСЛ [3], под управлением которой работает ККМП. Выполнение команд может быть инициировано в интерактивном режиме или из интерпретируемой программы эксперимента. Список команд управления механикой спектрометра приведен в приложении 2. На время отладки программ или наладки аппаратуры к компьютеру К-1520 можно подключить терминал и управлять им в интерактивном режиме.

4. Программа связи двух ККМП

ККМП между собой соединены через дополнительный последовательный канал связи, который зарезервирован в ККМП [3] для связи с другими ЭВМ.

При выполнении операций связи двух ККМП ведущим всегда является ККМП в кресте многоканального анализатора. В память этого ККМП загружается основная программа эксперимента, в которой можно предусмотреть передачу задания другому ККМП на установку определенных физических параметров или условий на образце, а также возможность синхронизации программ, работающих в разных ККМП. Для целей коммуникации в оперативной памяти каждого из взаимодействующих ККМП выделены общие блоки данных, доступ к которым осуществляется посредством специальных инструкций

языка MCL (см. приложение 3), которые позволяют прочесть и записать блок данных. В программе MCL эти блоки представлены в виде массива чисел типа REAL длиной в 64 значения.

В одном ККМП выделено два таких блока, каждый из которых отвечает состоянию одного ККМП. Назначение первых двух ячеек в общих блоках данных отведено для служебного использования, а назначение остальных ячеек в этом блоке определяется пользователем при составлении программы эксперимента. Первая ячейка из числа зарезервированных для служебного пользования содержит код ошибки, а вторая - номер строки интерпретируемой программы на языке MCL, в результате выполнения которой произошла ошибка. С помощью этих двух ячеек пользуясь командами MCL программа в одном ККМП может прочесть общий блок данных из другого ККМП и определить, работает или нет в нем программа измерений. Таким способом можно обеспечить устойчивое взаимодействие двух программ.

Рассмотрим способ взаимодействия двух ККМП на примере. Во-первых, в каждом блоке пользователь (автор программы эксперимента) выделяет две ячейки для обеспечения синхронизации операций чтения и записи общих блоков в прикладной программе. Назовем их Flag1 (для записи) и Flag2 (для чтения). Во-вторых, для операций с общими блоками данных нужно выделить в прикладной программе массив размером в 64 числа (пусть он будет иметь имя R). Положение Flag1 и Flag2 в массиве должно быть фиксированным.

Для того, чтобы передать в модуль управления параметрами задание, в программе эксперимента в многоканальном анализаторе сначала необходимо предусмотреть запись нужных для выполнения задаваемой операции данных в выделенный массив R. Положение этих значений в массиве должны "знать" обе взаимодействующие программы. Затем нужно установить Flag1 (присвоить соответствующему элементу массива 1), записать массив R в общий блок данных командой WRREC R и передать этот блок с помощью команды SENDV в крест-исполнитель. После этого в цикле читать по команде RECVB общий блок данных из модуля управления параметрами эксперимента, передавать его в массив R командой RDREC, где он становится доступным для анализа прикладной программой и анализировать Flag1. Цикл выполнять до тех пор, пока Flag1 станет равным 0 (это будет означать, что задание принято и прочитано программой модуля управления параметрами). Соответственно программа в этом модуле должна читать общий блок данных до тех пор, пока там появится Flag1=1, и после того, как переданные данные прочитаны, обнулить его и выполнить заказанную операцию.

Аналогичным способом, учитывая, что один ККМП всегда пассивен, можно предусмотреть ответную передачу, но для синхронизации использовать другой флаг. Например, организовав в программе ККМП многоканального анализатора чтение общего блока данных из ККМП модуля управления параметрами (команда RECVB) в цикле, можно анализировать Flag2, и если он равен 1, то это означает, что переданы данные, которые требуют анализа и обработки. Такой метод позволяет реализовать на уровне прикладных программ другие схемы взаимодействия различных ККМП.

5. Программы связи ККМП и СМ-1300

Для связи с ККМП на СМ-1300 используется последовательный байт-ориентированный интерфейс типа DL-11. С помощью переключателя СМ-1300 может быть соединена либо с многоканальным анализатором, либо с модулем управления параметрами спектрометра, в которых для связи используется специальный блок КАМАК.

Особо следует отметить разнообразие функций, возложенных на эту связь [1]. Список прикладных операций, которые используют линию связи между ККМП и СМ-1300, приведен в приложении 4. Они реализованы в виде команд языка MCL и доступны пользователю как в интерактивном режиме, так и для использования в программе эксперимента. Рассмотрим особенности реализации этих операций.

Во-первых, программное обеспечение в ККМП и СМ-1300 работает в двух основных режимах. Один из них (режим удаленного управления) предполагает, что программа на СМ-1300 позволяет пользователю с терминала этой ЭВМ работать в интерактивном режиме с ККМП. В этом случае линия связи используется для бесформатной передачи данных байт за байтом. В другом режиме (режиме работы по протоколу) данные передаются в виде кадров, формат которых приведен в приложении 5. Каждый передаваемый кадр содержит тип и код функции. В нужных случаях кадр содержит необходимые параметры для выполнения заданной операции. При выполнении различных прикладных операций взаимодействующие ЭВМ в согласии с протоколом переходят из одного режима работы в другой автоматически, но выполнение любой операции начинается в режиме удаленного управления и в этом же режиме ЭВМ будут работать по окончании выполнения операции.

Во-вторых, все операции связи реализованы в программных модулях, реализующих прикладные операции. Это, с одной стороны, нарушает стройность построения программного обеспечения связи, но

с другой - является наиболее экономичным решением для описываемой системы. Следует добавить, что при реализации прикладных операций использованы одинаковые алгоритмы реализации функций связи, а где возможно - одни и те же программные модули.

Исполнение любой прикладной операции, использующей линию связи, всегда начинается с выполнения процедуры перевода СМ-1300 и ККМП в режим работы по протоколу путем послышки программой из ККМП специального служебного кода CTRL/T по линии связи. После чего программа на другой ЭВМ посылает запрос INI (см. приложение 5), ждет подтверждения правильно принятого запроса и правильно выполненной операции по этому запросу, а далее выполняется индивидуальная часть операции.

Пересылки кадров по линии связи во всех операциях реализуются двумя подпрограммами (SENDKADR, RECKADR), которые обеспечивают достоверную передачу и прием одного кадра с учетом вычисления и проверки контрольной суммы, повторных передач, если кадр передан с ошибкой или не получено подтверждение о его приеме за определенный интервал времени. Использование циклической нумерации передаваемых кадров позволяет "приемнику" обнаруживать правильно принятые, но передаваемые повторно кадры (такая ситуация может возникнуть, если подтверждение о правильно полученном кадре утеряно или передано с ошибкой).

В случае нарушения протокола или возникновения фатальной ошибки (ошибки, с которой не может "справиться" протокол) в процессе выполнения операции связи на терминалы ККМП и СМ-1300 выдаются сообщения об ошибке и весь комплекс переводится в режим удаленного управления.

Состав реализованных прикладных операций, использующих линию связи ККМП и СМ-1300, ясен из списка команд протокола, которые в большинстве носят прикладной характер (см. приложение 5). Рассмотрим алгоритм реализации некоторых из них. Например, команда загрузки программы в кодах микропроцессора из файла на диске СМ-1300 в оперативную память ККМП выполняется в следующей последовательности:

1. Выполняется процедура перевода программ в СМ-1300 и ККМП в режим работы по протоколу.

2. ККМП посылает запрос LOAD (в кадре указано и имя файла). Если файл найден, программа на СМ-1300 открывает его и посылает подтверждение на запрос LOAD (тип команды АСК и код команды LOAD, будем писать АСК+LOAD), в противном случае в ККМП посылается кадр АСК+LOAD и обрабатывается процедура выхода по ошибке.

3. Производится загрузка программы в кодах микропроцессора. Причем здесь уже не работает нормальный протокол связи, а имитируется загрузка программы в ККМП с перфоленты. В ККМП при этом используется модифицированный начальный загрузчик.

На подобной же основе (когда одна часть операции выполняется по классическому алгоритму связи, а другая - совершенно специфическая) реализована операция RDF (см. приложение 5). Другим способом (более общим) реализованы другие операции (WRS, WRF, WRA). Например, команда WRS выполняется в следующей последовательности:

1. Программы в ККМП и СМ-1300 переводятся в режим работы по протоколу.

2. ККМП передает кадр (тип команды - REQ, код команды - WRS) с именем файла.

3. Программа в СМ-1300, приняв кадр REQ+WRS открывает на запись файл с заданным именем и в ККМП посылается кадр АСК+WRS. В случае возникновения ошибки при выполнении операции открытия файла или ошибки в имени файла программа в СМ-1300 с помощью запроса MSG обеспечивает передачу сообщения об ошибке на ККМП, то же самое выводит на терминал СМ-1300 и с помощью запроса REJ+WRS прекращает выполнение операции WRS.

4. Если файл был открыт, то СМ-1300 посылает в ККМП запрос REQ+АССЕРТ, сообщая о готовности к приему очередного кадра данных, и принимает их в кадрах REQ+DATA. После успешного приема очередного кадра СМ-1300 снова посылает в ККМП запрос REQ+АССЕРТ. В случае возникновения ошибки записи на СМ-1300 передача данных прекращается после вывода на удаленной ЭВМ сообщения (см. п. 3), выполнения операции прекращения передачи данных (REJ+АССЕРТ) и окончания выполнения данной команды (СМ-1300 посылает запрос REQ+CLR).

5. Если операция полностью проходила успешно, то после того, как ККМП обнаружит, что информация, предназначенная для передачи, исчерпана, он посылает запрос REQ+CLR, который означает конец операции. СМ-1300, приняв этот запрос, посылает на ККМП АСК+CLR и сообщение для пользователя (с помощью запроса REQ+MSG). Сообщение выводится также и на терминал СМ-1300.

Несколько слов следует сказать о проектировании и реализации протокола связи в программном обеспечении ККМП и СМ-1300. Прежде всего отметим четкое разделение двух уровней протокола связи.

Первый - обеспечивает достоверную передачу кадра данных. При этом не анализируются команды и данные, передаваемые в кадре, а подтверждение о правильном приеме кадра посылается в виде нумерованного кадра с типом команды АСК и кодом 0. Если кадр принят с ошибкой, посылается нумерованный кадр с типом НАСК и кодом 0. Другой уровень протокола непосредственно связан с прикладными операциями и с помощью команд протокола позволяет реализовать необходимые функции на комплексе машин. На этом уровне для каждого посланного запроса предусмотрено ожидание и получение подтверждения о его правильной интерпретации и выполнении или подтверждения о приеме запроса и обнаружении ошибки при его интерпретации или выполнении.

При проектировании протокола связи мы руководствовались следующими требованиями:

1. Обеспечить надежность протокола. Заложенные в него алгоритмы должны обеспечить устойчивость программ ко всевозможным ошибкам как аппаратного, так и программного (нарушение протокола одной из взаимодействующих программ) характера.

2. Протокол не должен допускать "зависаний" системы, успешного или неуспешного завершения операции в неопределенном состоянии.

3. Обеспечить достоверность передачи данных и выполнения прикладных операций (результаты выполнения заданных операций должны быть в недвусмысленном виде сообщены пользователю независимо от места, с которого он управляет системой).

4. Четкое разделение протокола на уровни и строгая формулировка интерфейса между этими уровнями.

5. Обеспечить непротиворечивость протокола.

6. Протокол должен допускать возможность создания компактного программного обеспечения.

Реализация протокола на СМ-1300 выполнена на языке Паскаль на основе разработанной в ЛНФ методики программирования алгоритмов, выполняемых в реальном масштабе времени, в виде совокупности взаимодействующих процессов [4]. Применение этой методики существенно облегчило процесс отладки программ связи, так как она позволяет приостановить и продолжить выполнение процессов, контролировать их состояние. Отметим, что каждый процесс соответствует одной из команд протокола. На ККМП каждая команда протокола реализована в виде отдельного модуля, который может вызывать общие для всех подпрограммы. В процессе реализации и отладки программ связи весьма полезными оказались диаграммы

состояний, которые составлялись для каждой из взаимодействующих подсистем.

6. Заключение

Отметим, что, несмотря на использование разнородной аппаратуры, реализация программного обеспечения связи выполнена концептуально в одном стиле, использованы одни и те же алгоритмы и при создании программ, насколько возможно выдерживался систематический подход к проектированию и реализации программ. Описанная организация программ связи и принятый в разработке подход позволили реализовать эти программы в течение полугода.

Созданное программное обеспечение используется около двух лет и опыт эксплуатации показал его надежность и устойчивость к ошибкам. Особо следует отметить основные принципы проектирования и реализации программ связи СМ-1300 с ККМП, описанные в предыдущем параграфе (в нем подчеркнуты те решения, которые могут быть использованы в других разработках).

В заключение авторы выражают благодарность О. И. Елизарову за помощь при разработке программ и А. В. Алфименкову за полезные обсуждения.

Приложение 1. Формат кадра, передаваемого между К-1520 и ККМП

8 | n | f | c | x | x | x | x

n - номер выбранного сервомеханизма,
fc - код функции, а xxxx - код позиции,
в которую нужно установить указанную ось
гониометра. Объем пакета - 4 байта.

Приложение 2. Команды управления механическими устройствами спектрометра, при выполнении которых используется линия связи

INGON - инициализировать аппаратуру связи между К-1520 и ККМП.
AXIS NA, PA, E - установить указанную ось гониометра в заданное положение, NA - номер оси, PA - код положения, E-событие (см. описание языка MCL [3]), оно устанавливается после того, как выполнение команды движения будет закончено (достигнуто заданное положение или произошла ошибка).
STAX NA - остановить движущуюся ось.
GSTOP - остановить все оси гониометра.
POSIT NA, PA - читать текущее положение оси с номером NA и записать эту информацию в переменную PA.

Приложение 3. Команды для связи двух ККМП

INCRТ	- инициализация канала связи.
SENDВ	- передача общего блока данных из крейта многоканального анализатора в крейт управления параметрами.
RECVВ	- запрос на чтение общего блока данных из крейта управления параметрами и передачу его в крейт многоканального анализатора.
RDREC R,NC	- запись общего блока данных из крейта с номером NC в массив R.
WRREC R	- запись информации из массива R в общий блок данных.

Приложение 4. Команды, при выполнении которых используется связь ККМП и СМ-1300

LINE 1	- инициализация аппаратуры и программ связи в ККМП, установление для ККМП режима удаленного терминала.
LOCAL	- выход ККМП из режима удаленного терминала.
WRMCL NL,NLE	- запись программы на языке MCL, находящейся в оперативной памяти ККМП, на диск СМ-1300. Имя файла задается в ответ на соответствующий вопрос программы. NL и NLE - номера первой и последней строки куска программы, который будет записан на диск.
RDMCL	- чтение программы на языке MCL из файла на диске СМ-1300 и запись ее в программный буфер интерпретатора MCL в ККМП. Имя файла вводится пользователем в ответ на соответствующий вопрос.
WRSPEC	- запись спектра из оперативной памяти ККМП на диск СМ-1300. Имя файла вводится в ответ на вопрос.
WASPEC P1,P2,P3	- запись спектра с номером P1 (номер участка буферной памяти, выделенной для данного спектра). Здесь P2 - код типа спектра, P3 - порядковый номер измерения (в одно измерение входят несколько спектров). Имя файла для спектра программа на СМ-1300 генерирует автоматически (в имени кодируются номер участка буферной памяти, номер измерения и тип спектра).

Тип спектра зависит от того, в каких условиях проводились измерения с образцом. Допустимые коды типа спектра приведены в следующей таблице.

Код типа		Код типа	
10	Спектр образца	110	Контрольный спектр образца
11	Спектр фона образца	111	Контрольный спектр фона образца
20	Нормировочный спектр	120	Контрольный нормировочный спектр
21	Нормировочный спектр фона	121	Контрольный нормировочный спектр фона

Приложение 5. Формат кадра, передаваемого между двумя ККМП или между ККМП и СМ-1300

SOM	CMD	L	DATA	CRC
-----	-----	---	------	-----

SOM (Start Of Message) - метка начала кадра (один байт), CMD - код команды протокола (один байт), L - длина данных (два байта), DATA - данные объемом в L байтов, CRC - циклическая контрольная сумма (два байта).

Код команды состоит из трех частей: тип команды, циклический двоичный счетчик и код функции.

Тип команды занимает три старших разряда (с 5-го по 7-й) в байте кода команды. Используется всего 4 типа:

Мнемоническое название	Двоичный код	Назначение типа
REQ	001	Запрос на выполнение какой-либо прикладной операции (Request).
REJ	010	Прекратить выполнение операции (Reject).
ACK	011	Подтверждение правильно принятого кадра (Acknowledge).
NAK	100	Подтверждение: кадр принят, но с ошибкой (Noacknowledge).

Циклический двоичный счетчик (для него отведен 4-й разряд кода команды протокола связи) используется для циклической нумерации передаваемых по линии связи кадров. Он позволяет определить повторно переданные кадры, о которых было утеряно подтверждение ACK или NACK. Потери или искажение кадров в других случаях протокол позволяет обнаружить без помощи циклической нумерации кадров.

Для кода функции отведены разряды с 0-го по 3-й:

Мнемоническое название	Двоичный код	Назначение кода функции
	0000	Нет прикладной функции.
INI	0001	Перевод системы в режим работы по протоколу связи.
REM	0010	Перевод системы в режим удаленного терминала.
RDF	0011	Чтение MCL-программы с диска на CM-1300 и загрузка ее в память ККМП.
WRF	0100	Запись MCL-программы из памяти ККМП на диск CM-1300.
WRA	0101	Запись спектра из ККМП на диск CM-1300 в режиме автоматической генерации имен.
WRS	0110	Запись спектра из ККМП на диск CM-1300 (имя файла передается от ККМП в кадре данных).
DATA	0111	Передача кадра данных из буфера в ККМП на диск CM-1300 в файл, который заранее был открыт по запросу RDF, WRF, WRA, WRS или LOAD.
MSG	1000	Передача сообщений для вывода их на терминал пользователя.
LOAD	1001	Запрос на загрузку программы в кодах микропроцессора ККМП из файла на CM-1300 в память ККМП.
ACCEPT CLR	1010 1011	Подтверждение готовности принять данные. Конец операции.

Литература

1. Гюнтер З. и др. Спектрометр поляризованных нейтронов. Распределенная система автоматизации экспериментов на базе микро-ЭВМ и микропроцессорных контроллеров. Сообщение ОИЯИ, P10-88-455, Дубна, 1988.
2. Гюнтер З. и др. Микропроцессорный контроллер с внутренней памятью 32Кбайт в стандарте КАМАК. Сообщение ОИЯИ, 11-84-482, Дубна, 1984.
3. Гюнтер З. и др. Использование диалогового языка MCL при автоматизации автономного многоканального анализатора. XII Международный симпозиум по ядерной электронике (Дубна, 2-6 июля 1985 г.). ОИЯИ, D13-85-793, Дубна, 1985, с. 250-254.
4. Островной А.И. Методика программирования систем автоматизации экспериментов на языке Паскаль. Автоматрия, N 5, 1986.

Рукопись поступила в издательский отдел
22 ноября 1989 года.

Лебнер М., Островной А.И.

P10-89-787

Организация программ связи в распределенной системе автоматизации экспериментов на спектрометре поляризованных нейтронов

Обсуждаются вопросы, которые возникают при создании программного обеспечения связи для распределенных систем автоматизации, ориентированных на решения одной, хотя и достаточно сложной задачи. Проблема рассматривается на примере системы автоматизации экспериментов на спектрометре поляризованных нейтронов, которая состоит из ряда относительно самостоятельных, но взаимодействующих подсистем, и имеет децентрализованное управление. Система автоматизации построена на базе соединенных линиями связи микро-ЭВМ и микропроцессорных контроллеров КАМАК. Рассмотрены особенности реализации прикладных операций, при выполнении которых используются линии связи, а также основные принципы организации программ и протокола связи.

Работа выполнена в Лаборатории нейтронной физики ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна 1989

Перевод авторов

Loebner M., Ostrovnoj A.I.

P10-89-787

The Network Software In the Distributed System of Experiment Automation for the Polarized Neutron Spectrometer

The problems which arise in software development for distributed automation systems for a single but quite complicated task are discussed. The problem is illustrated by the system for experiments automation on the polarized neutron spectrometer. The system includes a set of relatively independent but interacting subsystems with decentralized control. The automation system is built on the basis of microcomputer and microprocessor CAMAC controllers of the crate connected by communication lines. The features of the software implementation of applied operations, where communication lines are used, as well as the main principles of the software and communication protocol organization are shown.

The investigation has been performed at the Laboratory of Neutron Physics, JINR.

Communication of the Joint Institute for Nuclear Research, Dubna 1989