

НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЕ УЧРЕЖДЕНИЕ
ИНСТИТУТ ЯДЕРНОЙ ФИЗИКИ им. Г.И.Будкера СО РАН

В.Р. Мамкин, А.Н. Селиванов

Разработка контроллера ввода-вывода
с поддержкой TCP/IP

ИЯФ 2001-55

НОВОСИБИРСК
2001

**Разработка контроллера ввода-вывода
с поддержкой TCP/IP**

В.Р. Мамкин, А.Н. Селиванов

Институт ядерной физики им. Г.И. Будкера
630090 Новосибирск, Россия

Аннотация

В статье описан модуль ввода-вывода, выполненный на базе микроконтроллера фирмы Motorola. Рассмотрены варианты возможных применений модуля, особенности аппаратного построения и архитектура программного обеспечения. Приведен пример использования модуля для вывода данных из однокоординатного рентгеновского детектора.

Development of the IO controller with TCP/IP support

V.R. Mamkin, A.N. Selivanov

Institute of Nuclear Physics, 630090, Novosibirsk

Abstract

The input-output module, based on Motorola's microcontroller, is described. Variants of potential applications, hardware features and software architecture are considered. Employment of the module is illustrated on the example of data acquisition from one-coordinate x-ray detector.

© *Институт ядерной физики
им. Г.И. Будкера СО РАН, Россия*

1 Введение

В настоящее время стандарт Ethernet является основной средой передачи данных в локальных компьютерных сетях. Достоинства Ethernet, такие как масштабируемость, высокая скорость передачи данных, наличие гальванической развязки, делают его привлекательным также для применения в АСУ ТП и научном приборостроении. Массовый выпуск различного сетевого оборудования в данном стандарте привел к снижению цен и появлению развитой элементной базы, которая может быть использована при разработке устройств передачи данных. Кроме того, в большинстве случаев имеется готовая сетевая инфраструктура, что позволяет без значительных затрат интегрировать прибор с интерфейсом Ethernet в существующую компьютерную сеть.

Кроме реализации физического уровня Ethernet, для эффективной передачи данных прибор с сетевым интерфейсом должен реализовывать пакетный, транспортный и, в отдельных случаях, прикладной уровни сетевой модели. Наиболее распространенным и перспективным стеком сетевых протоколов на сегодняшний день является TCP/IP. Для реализации его в отдельном приборе необходимы достаточно большие вычислительные ресурсы. Например, наиболее распространенная реализация TCP/IP в операционной системе Free BSD занимает 150-200 Кбайт кода и требует около 256 Кбайт оперативной памяти. В подавляющем большинстве случаев реализовать сетевой стек протоколов в устройстве, которое должно еще выполнять свои основные задачи, не представляется возможным. Выходом из этой ситуации может служить применение интеллектуального встраиваемого контроллера, который мог бы выполнять функции шлюза между средой Ethernet/TCP/IP и внутренними интерфейсами прибора.

Руководствуясь приведенными выше соображениями в ИЯФ был разработан модуль ввода-вывода IP302, который использован для реализации интерфейса передачи данных в однокоординатном рентгеновском детекторе ¹. Модуль обладает функциональной законченностью, имеет значительные вычислительные ресурсы при небольших габаритах, а так-

¹Детектор разработан в лаб. 3-12, Аульченко В.М., Жуланов В.В., Шехтман Л.И.

же открытый программный интерфейс. Все это позволяет использовать модуль также и в других приборах и установках.

2 Технические требования к модулю

Технические требования, которые ставились при разработке модуля, можно условно разделить на следующие пункты:

- набор интерфейсов, которые должен поддерживать модуль
- процессор
- системное программное обеспечение

Для облегчения интеграции модуля в целевую систему, он должен поддерживать аппаратные интерфейсы, используемые обычно в приборостроении для связи между функциональными блоками. Примерами таких интерфейсов являются: SPI, применяемый обычно для подключения микросхем ЦАП и АЦП; UART; параллельная шина, которая может служить для ввода-вывода во внешние по отношению к модулю регистры; регистровый ввод-вывод. Для подключения к сети передачи данных был выбран интерфейс 10 base T "витая пара", как наиболее дешевый в реализации.

При выборе процессора учитывались следующие факторы. Использование TCP/IP стека предполагает наличие в модуле операционной системы, что означает достаточно большой объем кода и оперативной памяти. Большинство развитых операционных систем, предназначенных для встраиваемых контроллеров (VxWorks, OS-9 и т.д.), ориентируются на 32-разрядную архитектуру. Поэтому выбранный процессор должен быть 32-х разрядным и, кроме того, для него должны иметься в наличии свободно распространяемые (freeware) средства разработки программного обеспечения. Наличие на кристалле встроенного контроллера Ethernet и богатой периферии однозначно определил выбор в пользу микроконтроллера фирмы Motorola.

Системное программное обеспечение модуля должно иметь следующие возможности: поддержку протоколов TCP/IP, эффективное управление процессорным временем, абстрагирование прикладного программного обеспечения от аппаратного уровня, взаимодействие между нитями исполнения. Всем этим требованиям удовлетворяет операционная система реального времени (OS RV). Для установки ОС RV, на модуль ввода-вывода должны быть разработаны архитектурно зависимые компоненты: драйверы консоли, интерфейса Ethernet, таймера и код инициализации системы.

3 Аппаратное построение модуля

Блок схема модуля IP302 приведена на рис. 1.

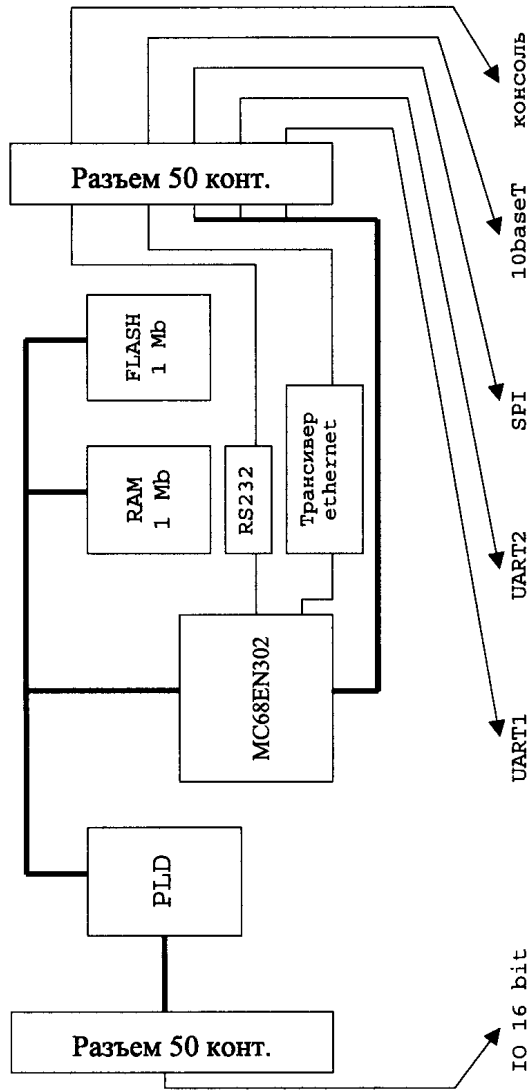


Рис. 1: Блок схема модуля

3.1 Процессор

Интегрированный микроконтроллер MC68EN302 фирмы Motorola имеет следующие характеристики:

Ядро	MC68000 совместимое 16 32-х разрядных регистров общего назначения Тактовая частота 25 Мгц
Шина	16 разрядов данных 24 разряда адреса
Блок системной интеграции	1 DMA контроллер 1 контроллер прерываний 2 таймера общего назначения 1 watchdog таймер DRAM контроллер
Коммуникационные возможности	3 универсальных многопротокольных контроллера с поддержкой асинхронных и синхронных протоколов (UART, HDLC, и т.д.) 1 SPI контроллер (мастер) 1 контроллер Ethernet

3.2 Память

Объем оперативной (SRAM) и постоянной перепрограммируемой (flash) памяти должен быть достаточным для запуска ОС РВ. На модуле установлено две микросхемы статической памяти 256К X 16, а также микросхема 512К X 16 flash памяти. Flash память может перепрограммироваться "на лету", средствами программного обеспечения модуля. Память используется программным обеспечением следующим образом:

адреса	назначение
0..0xffff	статическая память: вектора прерываний сегмент данных сегмент bss heap операционной системы стек системные таблицы
0x400000..0x40ffff	flash, сектор загрузочного монитора
0x410000..0x41ffff	flash, сектор параметров
0x420000..0x4ffff	flash, сектор кода

3.3 Интерфейсы ввода-вывода

Модуль имеет следующие интерфейсы ввода-вывода:

- Ethernet
- Консоль
- SPI
- 2 универсальных последовательных интерфейса
- регистровый ввод-вывод
- параллельная шина

Интерфейс Ethernet выведен за пределы модуля физическими уровнями и может быть подключен непосредственно к концентратору, коммутатору и другому сетевому оборудованию. Интерфейс построен на базе трансивера фирмы Level One и встроенного MAC контроллера Ethernet, входящего в состав процессора Motorola.

Консоль предназначена для начальной настройки модуля и контроля его работы. Консоль имеет уровни V.24, протокол UART и может подключаться без дополнительного согласования к последовательному порту компьютера, с которого осуществляется конфигурация модуля. Коммуникационные параметры консоли - скорость 38400, 8 bit, no parity, 1 stop.

Интерфейс SPI состоит из трех логических сигналов - вход данных, выход данных, сигнал синхронизации. Устройства, взаимодействующие по SPI интерфейсу могут быть двух видов - мастер (master) и подчиненный (slave). Мастер всегда является источником сигнала синхронизации и инициатором передачи данных. Модуль IP302 всегда является мастером. Скорость передачи данных устанавливается программно.

Универсальный последовательный интерфейс представляет собой группу логических сигналов: TXD, RXD, TXC, RXC, CD, RTS, CTS. Внешними по отношению к IP302 средствами эти сигналы могут преобразовываться в стандартные интерфейсы RS-232, RS-422, RS-485 и т.д. В зависимости от потребностей последовательный интерфейс может работать в асинхронном или синхронном режиме с внутренней или внешней синхронизацией. Протоколы канального уровня обеспечиваются коммуникационным контроллером в составе MC68EN302. Кроме последовательного интерфейса через указанные логические сигналы может осуществляться регистровый ввод-вывод.

Параллельная шина может служить для скоростного ввода-вывода данных из внешних устройств. Шина представляет собой группу логических сигналов: 16 битов данных, 5 битов адреса, 3 программируемых

строба. Для дополнительной гибкости шина управляется через программируемую логическую матрицу (PLD) фирмы Altera. В зависимости от конфигурации PLD может осуществлять следующие функции: стробируемый ввод/вывод данных во внешние регистры, организация FIFO между модулем IP302 и внешним процессором, организация DMA каналов.

3.4 Конструктивное исполнение

Модуль IP302 выполнен в формате модуля Industry Pack и имеет габариты 100x50x11(мм). Внешние сигналы выводятся через два 50-контактных разъема, один из которых занят параллельной шиной, через другой подключаются последовательные интерфейсы, консоль, SPI и Ethernet.

4 Программное обеспечение

Программное обеспечение модуля состоит из:

- Кросс-средств разработки ПО
- Загрузочного монитора
- Операционной системы реального времени
- Прикладных задач

4.1 Кросс-средства разработки ПО

В качестве инструментария для разработки кода были использованы свободно распространяемые кросс-средства под лицензией GNU. В состав этих средств входят бинарные утилиты и C компилятор. Особенность этого программного обеспечения в том, что оно поставляется в исходных текстах и построено по идеологии host - target. Под host платформой понимается платформа, на которой исполняется программное обеспечение, под target понимается целевая система, т.е. система для которой генерируется программный код. Host и target указываются на этапе построения кросс-средств из исходных текстов. В случае модуля IP302 target процессором является MC68000, host платформа - Linux или Win9x. Имеются следующие кросс-средства разработки ПО:

- ассемблер
- линкер
- утилиты для работы с объектными файлами (objcopy, objdump)
- C препроцессор
- C компилятор

Процесс разработки кода прикладной программы может выглядеть так:

1. Разработка в текстовом редакторе исходного текста модулей программы на С или ассемблере.
2. Компиляция исходных текстов С компилятором или ассемблером. На этом этапе получается несколько промежуточных объектных файлов.
3. Компоновка объектных файлов и системных библиотек с помощью линкера. На этом этапе получается один объектный файл, в котором содержится код прикладных программ и операционной системы.
4. Получение с помощью утилиты objcopy бинарного загружаемого образа.
5. Загрузка образа через Ethernet в оперативную память модуля или программирование его во flash память.
6. Запуск программы на исполнение. Результат работы может контролироваться через консоль.

4.2 Загрузочный монитор

После включения питания или перезапуска IP302 управление всегда получает загрузочный монитор. В зависимости от установленных параметров, монитор сразу после запуска может начать загрузку файла в оперативную память контроллера, выполнить переход по заранее установленному адресу или перейти в командный режим. Командный режим используется для конфигурации монитора и выполнения определенных сервисных функций. В этом режиме монитор выполняет вводимые пользователем через консоль команды и выводит на консоль результат их выполнения.

Монитор имеет следующие возможности:

- Загрузка файлов по протоколу TFTP из внешнего сервера
- Программирование и запуск программ во flash памяти
- Выполнение сервисных функций по команде пользователя (ARP запрос, ICMP echo)
- Редактирование параметров и настроек сети
- Копирование, заполнение, тестирование памяти

Параметры монитора сохраняются во flash памяти и могут использоваться операционной системой.

4.2.1 Варианты загрузки

Основная задача монитора - загрузка и запуск операционной системы. Источников загрузки может быть два: сетевой TFTP сервер или flash память. TFTP протокол был разработан специально для загрузки бездисковых устройств. Это транспортный протокол, базирующийся на UDP/IP. Обычно после включения, бездисковое устройство посылает TFTP серверу запрос на загрузку файла. Если файл найден, начинается копирование его в оперативную память устройства (в данном случае IP302). TFTP сервером может служить любая рабочая станция, подключенная к сети, с запущенной программой TFTP сервера. Для Linux, например, такая программа входит в состав стандартных утилит. В случае IP302 загрузка программ по TFTP протоколу служит для отладочных целей. Если программное обеспечение (ОС и прикладная задача) запрограммировано во flash памяти, монитор передает управление по заранее известному адресу во flash.

4.2.2 Параметры монитора

Параметры монитора сохраняются в отдельном секторе flash памяти и могут изменяться пользователем через консоль. Пользователю доступны следующие параметры:

enetAddr - ввод Ethernet адреса контроллера

myip - ввод IP адреса контроллера.

servip - ввод IP адреса сервера, с которого будет загружаться файл по TFTP протоколу.

gwip - ввод IP адреса сетевого шлюза.

mask - ввод сетевой маски

file - имя файла, загружаемого из TFTP сервера

loadptr - адрес области памяти, куда будет загружаться файл

jumpptr - адрес передачи управления по команде go или при автозапуске

bootstr - содержание командной строки, которая передается операционной системе

bootptr - адрес, куда копируется командная строка перед запуском ОС

4.2.3 Сервисные команды

Сервисные команды монитора иницируются пользователем в тестовых целях. Доступны следующие сервисные команды:

pings - монитор переходит в режим ICMP эхо сервера и отвечает на ping запросы от других станций в сети. Режим служит для проверки соединения локальной сети. Перед выполнением команды должны быть установлены сетевые настройки (IP адрес, маска, шлюз)

bootp - монитор запрашивает сетевые параметры у BOOTP сервера

arp - монитор делает ARP запрос и получает Ethernet адрес сервера по его IP адресу. Предварительно должен быть введен IP адрес сервера (параметр servip).

load - монитор загружает файл в оперативную память контроллера по TFTP протоколу. Адрес памяти задается параметром loadptr, сетевой адрес - параметром servip, имя файла - параметром file.

go - монитор передает управление по адресу

mdump - распечатка содержимого памяти

mfill - заполнение области памяти кодом

mtest - тестирование области памяти, содержимое при этом теряется

fflash - программирование flash памяти. Команда аналогична команде load, только данные загружаются не в оперативную память, а программируются во flash с адреса 0x420000. По этому же адресу должно передаваться управление командой go и при автозапуске. Длина запрограммированного участка во flash памяти определяется длиной файла.

4.3 Операционная система

В качестве системного программного обеспечения была выбрана операционная система реального времени RTEMS (Real-Time Executive for Multiprocessor Systems). Данная ОС является свободно распространяемой и поставляется в исходных текстах. RTEMS имеет следующие особенности:

- многозадачность с перепланировкой по событиям или таймеру
- распределение процессорного времени в соответствии с приоритетами задач
- межзадачное взаимодействие и синхронизация
- средства управления памятью
- поддержка драйверов внешних устройств
- гибкость конфигурации

4.3.1 Программный интерфейс

При работе с RTEMS программист оперирует со следующими объектами операционной системы:

Задачи. Можно создавать, удалять, останавливать задачи, менять их приоритет.

Прерывания. ОС дает возможность устанавливать обработчики прерываний, запрещать или разрешать прерывания.

Часы реального времени. С использованием средств операционной системы прикладная задача может получать или устанавливать показания часов реального времени.

Таймеры. Для вызова через заданное время определенной функции используются таймеры. В отличие от аппаратных таймеров, количество программных таймеров не лимитировано и задается при конфигурации ОС.

Семафоры используются для организации доступа нескольких задач к разделяемому ресурсу или для синхронизации задач.

Очереди сообщений могут использоваться для передачи информации между задачами, а также для синхронизации задач.

События также могут использоваться для синхронизации задач, но, в отличие от семафоров, события могут инициироваться обработчиками прерываний, т.е. аппаратным уровнем.

Сигналы. Каждая задача может установить обработчик сигнала, а также послать сигнал. Сигналы служат для обработки асинхронных программных событий.

Отделы представляют собой целостную область памяти, разделенную на буферы одинакового размера. Задачи могут запрашивать у отдела буфер и возвращать его.

Регионы представляют собой целостную область памяти, разделенную на блоки, размер которых может варьироваться. Задачи могут запрашивать и освобождать блоки памяти. В отличие от стандартной C кучи, если в регионе не хватает памяти, задача может быть приостановлена до тех пор, пока память не появится.

Драйверы устройств предназначены для абстрагирования прикладной задачи от аппаратуры. Задача может открывать, закрывать, писать и читать драйвер устройства.

Кроме того, операционная система предоставляет интерфейс сокетов TCP/IP. Подключение сетевых сервисов происходит путем компоновки

объектного кода прикладной программы и сетевой библиотеки. Программный интерфейс сокетов не отличается от принятого в BSD UNIX.

4.3.2 Пакет поддержки платы (BSP)

Во встраиваемых операционных системах архитектурно зависимую часть ядра принято выделять в отдельный программный модуль - BSP (Board Support Package). Для модуля IP302 и ОС RTEMS был разработан BSP, в состав которого входят:

- драйвер консоли
- драйвер интерфейса Ethernet
- поддержка таймера
- код инициализации ОС

4.4 Прикладные задачи

Есть два существенно отличных варианта применения модуля IP302. В первом случае IP302 выполняет только коммуникационные функции и выступает как шлюз между средами Ethernet /TCP/IP с одной стороны, и рассмотренными выше аппаратными интерфейсами, с другой стороны. При этом функции управления берет на себя система, в которую встраивается модуль. Во втором случае модуль рассматривается как интеллектуальный контроллер, в функции которого входит не только передача данных, но и функции управления, обработки данных и (возможно) их представление.

4.4.1 Шлюз

В случае, если модулем IP302 выполняются функции шлюза, прикладное программное обеспечение в общем случае состоит из сервера TCP, процессоров обработки данных, диспетчера и обработчиков событий (рис. 2).

Эти компоненты имеют следующее назначение. Сервер ожидает запросы на соединение от TCP/IP стека, после чего, получив от мастера (удаленного компьютера, который осуществляет функции управления) пакет-демультиплексор, определяет тип соединения и канал его обработки. Может быть два типа соединения: запрос на передачу данных и запрос на ожидание асинхронного события. После определения параметров запроса сервер передает номер сокета соответствующему программному

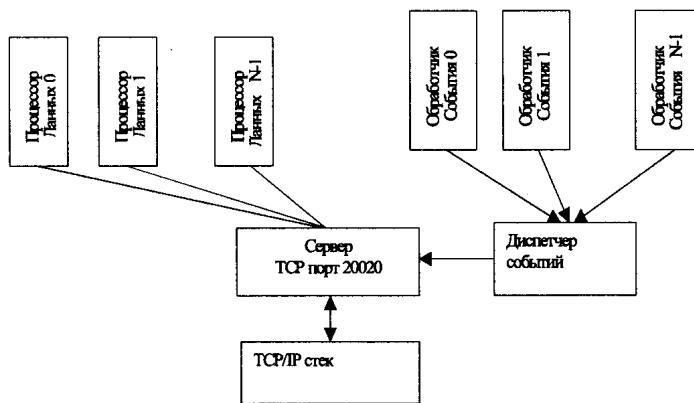


Рис. 2: Коммуникационный шлюз. Прикладные задачи.

модулю и "забывает" о соединении. Для передачи данных может быть организовано до 256 каналов, каждый из которых управляется своим процессором данных. Процессор данных (ПД) это программный модуль, соответствующий определенному физическому интерфейсу и реализующий определенный протокол по TCP каналу. ПД реализован в виде отдельной задачи (одновременно таких задач может быть не более 4), поэтому после обработки запроса, сервер не дожидается завершения передачи данных и готов к приему следующего запроса. ПД регистрируются у сервера при запуске программы. Для обработки асинхронных событий, например, прерывания от внешней аппаратуры, существует запрос на ожидание асинхронного события. Получив его, сервер сообщает номер сокета и маску желаемого события диспетчеру событий и "забывает" о соединении. Обработчики событий принимают прерывание непосредственно от аппаратуры и посылают программное сообщение о событии диспетчеру событий. Если диспетчер был инициализирован сервером и маска, которую имеет диспетчер, соответствует данному событию, диспетчер посылает сообщение о событии по TCP соединению. Как в случае работы с процессорами данных, так и в случае работы с асинхронными событиями, инициатива открытия и закрытия TCP соединения принадлежит мастеру (за исключением ошибочных ситуаций). В качестве шлюза между SPI и Ethernet модуль IP302 используется в однокоординатном рентгеновском детекторе.



Рис. 3: Представление данных через http.

4.4.2 Интеллектуальный контроллер

Системное программное обеспечение модуля IP302 позволяет использовать его для функций управления и представления данных. Благодаря наличию в составе ПО http сервера, уровень представления данных может быть реализован через www интерфейс. Достоинства такого подхода очевидны - для управления установкой используется стандартный интернет-навигатор, нет необходимости разрабатывать программное обеспечение на клиентской машине, возможно одновременное обслуживание нескольких клиентов. Блок-схема программного обеспечения приведена на рис. 3.

Виртуальная файловая система представляет собой набор html файлов, отображаемых http сервером по запросу удаленного клиента. html файлы состоят из таблиц, рисунков, текста и прочей отображаемой информации, которая формирует интерфейс пользователя. Обратная связь от клиента к модулю IP302 осуществляется пользователем путем заполнения html форм. http сервер декодирует формы и, в зависимости от их содержимого, изменяет содержимое базы данных параметров (БД). БД может содержать результаты измерений или переменные, используемые управляющей задачей.

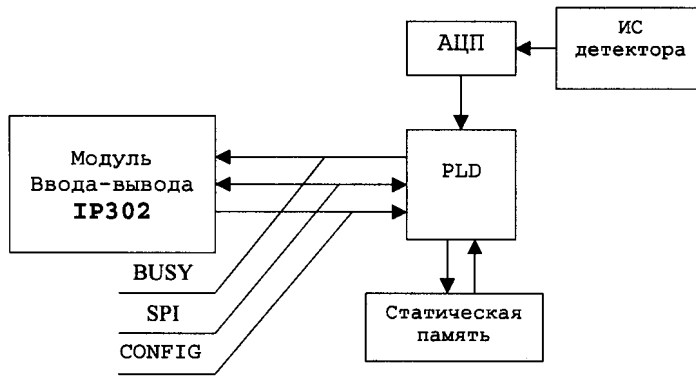


Рис. 4: Блок-схема электроники детектора.

5 Применение модуля IP302 в однокоординатном рентгеновском детекторе

Блок-схема электроники детектора приведена на рис. 4. Система считывания данных измерений состоит: заказной ИС детектора, АЦП, программируемой логической матрицы, статической памяти для сохранения результатов измерений, интерфейса взаимодействия с модулем IP302. Программируемая логическая матрица (PLD) может находиться в одном из двух состояний - состояние измерения или состояние взаимодействия по SPI. В состоянии измерения оцифрованные данные с АЦП считываются PLD и сохраняются в статической памяти. Это состояние сопровождается активным сигналом BUSY. В состоянии взаимодействия по SPI программируемая матрица принимает команды от модуля ввода-вывода. Команды могут двух видов - запустить измерение или считать данные измерений. Данные измерений, хранящиеся в статической памяти, считываются также через SPI интерфейс. Кроме взаимодействия по SPI и обработки сигнала BUSY модуль IP302 выполняет также процедуру начальной конфигурации PLD.

Структура прикладного программного обеспечения модуля IP302 соответствует показанной на рис. 2. Специально для рентгеновского детектора были разработаны следующие программные компоненты:

- Процессор SPI данных
- Процессор конфигурационных данных PLD Altera
- Обработчик сигнала BUSY

6 Заключение

Разработанный модуль IP302 был применен в качестве модуля ввода-вывода данных для однокоординатного рентгеновского детектора. Ряд особенностей модуля, таких как использование 32-х разрядного процессора, операционной системы, стандартных коммуникационных интерфейсов и стандартных протоколов передачи данных, позволяют применять модуль в других разработках в качестве интерфейса передачи данных или интеллектуального контроллера.

В.Р.Мамкин, А.Н.Селиванов

**Разработка контроллера ввода-вывода
с поддержкой TCP/IP**

V.R.Mamkin, A.N.Selivanov

Development of the IO controller with TCP/IP support

ИЯФ 2001-55

Ответственный за выпуск А.М. Кудрявцев

Работа поступила 31.07.2000 г.

Сдано в набор 1.08.2001 г.

Подписано в печать 2.08.2001 г.

Формат бумаги 60×90 1/16 Объем 1.4 печ.л., 1.1 уч.-изд.л.

Тираж 90 экз. Бесплатно. Заказ N 55

Обработано на IBM PC и отпечатано на
ротапринте ИЯФ им. Г.И. Будкера СО РАН

Новосибирск, 630090, пр. академика Лаврентьева, 11.